МІНІСТЕРСТВО ОСВІТИ І НАУКИ,
МОЛОДІ ТА СПОРТУ УКРАЇНИ

В.М. РЯБЕНЬКИЙ, О.О. УШКАРЕНКО

СХЕМОТЕХНІКА
ЕЛЕКТРОННИХ ПРИСТРОЇВ ТА СИСТЕМ

Рекомендовано Міністерством освіти і науки України як підручник для студентів вищих навчальних закладів

У ШЕСТИ ТОМАХ

ТОМ 6

В.М. РЯБЕНЬКИЙ
О.О. УШКАРЕНКО

АПАРАТНО-ПРОГРАМНІ ЗАСОБИ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ

Миколаїв
“Іліон”
2013

УДК 621.3.085
ББК ____
Р 98

Р е ц е н з е н т и :
В. С. СМИРНОВ, доктор технічних наук, професор
М. Т. ФІСУН, доктор технічних наук, професор

Рекомендовано Міністерством освіти і науки України
як підручник для студентів вищих навчальних закладів
(лист № 1.4/Г8–Г–3007 від 31.12.2008 р.)

Рябенький В. М.
Р 98 Схемотехніка електронних пристроїв та систем : підручник : у 6 т. / В. М. Рябенький. – Миколаїв : Іліон, 2013 – .
ISBN 978-617-534-066-0
Т. 6 : Апаратно-програмні засоби відображення інформації / В. М. Рябенький, О. О. Ушкаренко. – 2013. – 464 с.
ISBN 123-456-789-012-0

У підручнику розглядаються системні принципи аналізу та проектування електронних систем.

Шостий том – «Апаратно-програмні засоби відображення інформації» – присвячено поясненню принципів роботи і особливостей управління різними типами індикаторів. Викладені фізичні основи роботи, конструкції, основні параметри і сфери застосування індикаторів та оптоелектронних приладів. Розглянуті питання управління індикаторами з використанням мікропроцесорної техніки. Наводяться приклади схем з мікропроцесорним управлінням різними засобами відображення інформації, а також коди програм для мікроконтроллерів сімейства AVR. Підручник призначений для студентів і фахівців електротехнічних спеціальностей.

УДК 621.3.085
ББК ____

ISBN 978-617-534-066-0
ISBN 123-456-789-012-0 (Т. 6)

© Рябенький В.М., 2013
© Ушкаренко О.О., 2013

ВСТУП

Більше 80% інформації людина сприймає органами зору. Тому візуальному сприйняттю інформації приділяється першочергова увага. Значні успіхи, досягнуті в області обчислювальної техніки і цифрових апаратих комплексів, що розробляються на її основі, стимулювали широкий фронт робіт зі створення електронних індикаторних пристроїв і систем.

Сфери застосування електронних індикаторів різноманітні і практично не обмежені. В даний час в апаратному забезпеченні різних електронних систем в якості елементів індикації широко використовуються напівпровідникові індикатори, рідкокристалічні панелі, газорозрядні, електролюмінесцентні індикатори, електронно-променеві трубки. Кожен з вказаних типів індикаторів, заснованих на використанні різних фізичних принципів, висуває певні вимоги до амплітуд управляючих напруг, виду струму, щільності розміщення на лицьових панелях приладів, зовнішньої освітленості. Вибір типу індикаторів диктується часто не лише оптимальним поєднанням вказаних технічних характеристик, але і вартістю, термінами розробки, станом серійного виробництва.

Сучасні індикатори повинні забезпечувати впевнене зчитування інформації в умовах сильної і слабкої освітленості, мати високу яскравість і контрастність, мати високу чутливість до управляючих сигналів, малу споживану потужність, високу довговічність і надійність.

Застосування мікропроцесорних засобів дозволяє значно скоротити об'єми схем управління елементами індикації і підвищити надійність індикаторних пристроїв і систем за рахунок реалізації алгоритму управління не апаратними, а програмними засобами.

Пропонований підручник дає загальні відомості і фактичний матеріал по технічним характеристикам індикаторів і мікросхемам управління ними, питанням конструювання електронних індикаторних пристроїв і систем. Приводиться велика кількість прикладів пристроїв, в яких використовуються напівпровідникові індикатори. Значна увага приділяється питанням управління рідкокристалічними індикаторами. Приводяться тексти програм для управляючих мікроконтроллерів. Розглянуті питання формування зображення з використанням телевізійних засобів відображення інформації.

Підручник призначений для студентів вищих учбових закладів, які займаються проектуванням електронних пристроїв, у складі яких є засоби відображення інформації. Книга буде корисною також для широкого кола фахівців в області електроніки, які бажають використовувати сучасні засоби відображення інформації в своїх проектах.

УМОВНІ СКОРОЧЕННЯ

АПЗВІ – апаратно-програмні засоби відображення інформації
АЦП – аналого-цифровий перетворювач
БЦІ – буквено-цифрові індикатори
ВАХ – вольтамперна характеристика
ВЛІ – вакуумний люмінесцентний індикатор
ВІР – вакуумний індикатор розжарювання
ГІМ – графічна інформаційна модель
ГРІ – газорозрядний індикатор
ГЕ – графічний елемент
ЗСІ – знакосинтезуючі індикатори
ІМ – інформаційна модель
ІП – інформаційне поле
ІС – інтегральна схема
РК – рідкі кристали
РКІ – рідкокристалічний індикатор
РКМ – рідкокристалічна матриця
ОЗП – оперативний запам’ятовуючий пристрій
НПІ – напівпровідниковий індикатор
ПЕП – п'єзоелектричний перетворювач
РБВ – розподілений брегівський відбивач
РЗЗ – резонатор з розподіленим зворотнім зв’язком
СВД – світловипромінювальний діод
СЛР – сигнальні лампи розжарювання
ЗВІ – засоби відображення інформації
СЕ – сонячні елементи
УЯС – ультраяскраві світлодіоди
ЦІ – цифровий індикатор
ШІ – шкальний індикатор
ШІМ – широтно-імпульсна модуляція
ЕІМ – елементи інформаційної моделі
ЕЛІ – електролюмінесцентний індикатор
ЕПТ – електронно-променева трубка
ЕВ – елемент відображення
ЕФІ – електрофоретичний індикатор
ЕХІ – електрохромний індикатор
LCD – рідкокристалічний дисплей
LED – світлодіод
WST – всесвітня система телетексту

1. КЛАСИФІКАЦІЯ ЗАСОБІВ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ. ОСНОВНІ ВИЗНАЧЕННЯ

1.1. Основні параметри індикаторів

Для того, щоб система або пристрій відображення інформації із застосуванням індикаторів працювала надійно і ефективно, розробнику необхідно знати повну характеристику індикатора, що застосовується. Система параметрів, яка найповніше описує всі властивості і особливості індикатора, включає в себе:

– параметри, що характеризують індикатор як елемент системи «оператор — індикатор» і які визначають якість відображення інформації і надійність її сприйняття;

– параметри, що характеризують індикатор як елемент електричного ланцюга;

– параметри, що характеризують можливість функціонування індикатора в умовах впливу різних експлуатаційних факторів (вібрації, ударів, температури і т. ін.);

– параметри, що характеризують надійність роботи.

До першої групи параметрів відносяться фотометричні і ергономічні параметри. Розглянемо основні фототометричні параметри індикаторів.

Світловий потік Φ_υ – це потужність світлової енергії, що оцінюється за дією на людське око, спектральна чуттєвість якого стандартизована. Світловий потік вимірюється в люменах [Лм];

Сила світла I_υ – характеризує просторову щільність світлового потоку в данному напрямку і визначається як відношення світлового потоку до $d\Phi_\upsilon$, що проходить всередині малого тілесного кута $d\omega$ у напрямку, що розглядається, до величини цього кута. Сила світла рівна:

$$I_v = d\Phi_v / d\omega ,$$

де $d\omega = dA / r^2$ – тілесний кут, dA – площа, обмежена тілесним кутом на поверхню сфери A радіуса r; сила світла вимірюється в Канделах;

Яскравість – визначається силою світла, яка випромінюється одиницею поверхні. Людське око сприймає дуже широкий діапазон яскравості – від 10^{-6} до 10^6 Кд/м^2. При вимірах рівня яскравості чуттєвість ока не залишається постійною. Як високий, так і низький рівні яскравості стомлюють спостерігача. Яскравість випромінюваного світла окремих елементів зображення повинна бути по можливості однакова, що не завжди вдається реалізувати. Тому в довідникових даних індикаторів зазвичай вказують припустиму нерівномірність випромінюваного світла елементів зображення. Яскравість елемента зображення B і його освітленість E пов’язані співвідношенням:

$$B = E\rho / \pi ,$$

де ρ – коефіцієнт відбиття поверхні.

В затемнених приміщеннях при низьких рівнях зовнішнього освітлння достатньо мати яскравість 30…70 Кд/м^2. Для зручного сприйняття рухливих зображень бажано мати яскравість не нижче 300 Кд/м^2.

Контраст – відношення різниці яскравостей зображення і фону до яскравості фону. Можливість виділення об'єкту на фоні визначається контрастною чутливістю. Вона найбільш висока до чорно-білих колірних поєднань. Яркісний контраст визначається різницею яскравостей (освітленостей) сусідніх ділянок поля зору, або різницею яскравостей сигналів, які спостерігаються послідовно в часі на одній і тій же ділянці поля. Оцінка яскравості зображення, що відтворюється, залежить від яскравості фону, що його оточує.

Яркісний контраст для знаків більш темних, ніж фон (прямий контраст), визначається співвідношенням:

$$K_{пр} = \frac{B_ф - B_0}{B_ф} \cdot 100\% ,$$

де $B_ф$ – яскравість фону; B_0 – яскравість знаку.

Для знаків светліших від фону, контраст яскравості (зворотній контраст) визначається як:

$$K_{пр} = \frac{B_0 - B_ф}{B_0} \cdot 100\% .$$

Символ, що відтворюється засобом відображення інформації, невиразний, якщо контраст знаку і фону менше значення, яке називається пороговим контрастом. Пороговий контраст для знаків з кутовим розміром 60 і яскравістю близько 100 Кд/м^2 складає 0,015...0,02. Значення $K \le 0{,}2$ відповідає малому, $0{,}2 < K < 0{,}5$ – середньому і $K \ge 0{,}5$ високому контрасту. Підвищити контраст можна створюючи штучний фон, або підсвічуючи фон.

Параметром пасивних індикаторів є контраст знаку і фону. Використовуються два визначення контрасту:

Абсолютний контраст

$$K_{абс} = B_{\max} / B_{\min} ,$$

де $B_{\max}$ і $B_{\min}$ – яскравість світлих і темних ділянок зображення відповідно.

Відносний контраст

$$K_{отн} = \frac{B_{\max} - B_{\min}}{B_{\max}} \cdot 100\% .$$

Оскільки око є нелінійним приймачем, відносний контраст більш повно відображає характер сприйняття. Абсолютний і відносний контрасти пов'язані:

$$K_{отн} = \frac{K_{абс} - 1}{K_{абс}} \cdot 100\% .$$

Кут огляду – параметр індикатора, що визначає зручність застосування його в пристроях відображення інформації. Кут огляду – це сукупність кутів спостереження, при яких забезпечується безпомилкове зчитування відтворюваної індикатором інформації. Під кутом спостереження розуміється кут між напрямом спостереження і нормаллю до площини знакомісця індикації. На величину кута спостереження впливають багато чинників: конструктивне виконання індикатора (плоский, циліндричний, торцевий, глибина знакомісця та ін.), рівень зовнішньої освітленості, розташування спостерігача, яскравості індикатора та ін. Для кожного поєднання цих чинників існують кути спостереження, при яких забезпечується безпомилкове зчитування інформації. Максимальна величина такого кута спостереження називається оптимальним кутом огляду. Для характеристики просторового сприйняття випромінювання використовується діаграма спрямованості випромінювання індикатора, що відображає кутовий розподіл інтенсивності випромінювання (рис. 1.1).

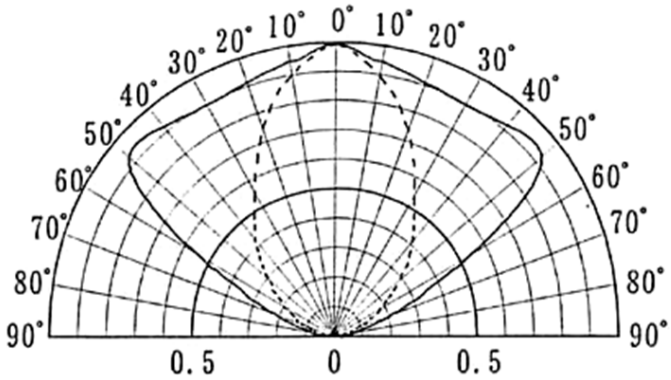


Рис. 1.1.

Колір – використовується для найкращого розрізнення сигналів, що відображаються. Колір характеризується трьома параметрами: яскравістю, тоном кольору і насиченістю. Яскравість кольору визначається величиною світлового потоку, що випромінюється в заданому напрямі одиницею площі поверхні в межах одиничного тілесного кута. Тон кольору – властивість кольору, що дозволяє визначити даний колір як червоний,

зелений і т.д. Він характеризується довжиною світлового випромінювання. Насиченість кольору – ступінь його свободи від домішки білого кольору. В індикаторах зазвичай не вдається досягти вузького спектру випромінювання. У однобарвних індикаторах бажано, щоб максимум спектру випромінювання збігався з максимальною чутливістю ока. Відтінки сірого (в діапазоні білий – чорний) носять назву ахроматичних кольорів. Найяскравішим ахроматичним кольором є білий, найтемнішим – чорний. При максимальному зниженні насиченості розрізнення тону (відношення до певного кольору спектру) відтінку стає неможливим [39].

До другої групи відносяться параметри, які характеризують індикатор як елемент електричного ланцюга, тобто електричні параметри і їх залежності від режиму застосування. Перелік електричних параметрів і їх визначення приведені в табл. 1.1.

Табл. 1.1.

Найменування параметра	Визначення
Постійний прямий струм елемента відображення $I_{пр}$, мА	Значення постійного струму, який протікає через випромінювач у прямому напрямку
Імпульсний прямий струм елемента відображення $I_{пр.і}$, мА	Найбільше миттєве значення прямого струму, який протікає через випромінювач
Середній прямий струм елемента відображення $I_{пр.ср}$, мА	Середнє за період значення прямого струму через випромінювач
Постійна (імпульсна) зворотня напруга на елементі відображення $U_{зв}$ ($U_{зв.і}$), В	Значення постійної напруги, прикладеної до випромінювача у зворотньому напрямку
Постійна (імпульсна) напруга пряма $U_{пр}$ ($U_{пр.і}$), В	Значення постійної (імпульсної) напруги на випромінювачі при проходженні через нього заданого постійного (імпульсного) прямого струму
Середня електрична потужність випромінювача, що розсіюється, $P_{ср}$, мВт	Середня за період потужність, що розсіюється випромінювачем при протіканні струму у прямому і зворотньому напрямках
Імпульсна потужність випромінювача, що розсіюється, $P_і$, мВт	Найбільше миттєве значення потужності, що розсіюється випромінювачем при подачі імпульсів із заданою тривалістю та скважністю

Постійний зворотній струм $I_{зв}$, мкА	Постійний струм у зворотньому напрямку
Тепловий опір перехід-корпус R_{nK}, °С/Вт	Відношення різниці температур переходу в контрольній точці на корпусі індикатора до потужності, що розсіюється приладом

Третьою групою є параметри, які характеризують стійкість індикаторів до дії різних експлуатаційних чинників (механічні параметри індикаторів). Перелік цих чинників приведений в таблиці 1.2.

Табл. 1.2.

Найменування параметра	Діапазон значень
Вібраційні навантаження: діапазон частот, Гц	1 – 5000
Прискорення g, м/с^2	10 – 20
Багатократні ударні навантаження з прискоренням g, м/с^2	75 – 150
Одиночні удари з прискоренням g, м/с^2	1500
Постійне прискорення g, м/с^2	500
Акустичні шуми в діапазоні 50 – 10000 Гц з рівнем звуку, дБ	160 – 170

Останньою групою параметрів, що характеризують індикатори, є параметри надійності. Під надійністю індикатора розуміється вірогідність того, що задача буде успішно виконана оператором при будь-якому поєднанні ергономічних параметрів в заданих режимах і умовах експлуатації протягом заданого часу.

Одним з основних параметрів є інтенсивність відмов. Найбільш достовірні результати за значенням інтенсивності відмов можна отримати при спеціальних випробуваннях на надійність.

Істотним чинником, який необхідно враховувати при проектуванні пристроїв і систем відображення інформації, є ступінь і характер деградації світлотехнічних параметрів в процесі тривалої експлуатації індикаторів.

На практиці при створенні того або іншого приладу із застосуванням індикаторів розробника цікавить, в якій мірі будуть змінюватися світлотехнічні параметри в процесі експлуатації. В даний час прийнято обговорювати в технічній документації на індикатор (активний) зміну сили світла, що дорівнює 50% від початкової. Це обумовлено можливостями оператора вловити зміну сили світла працюючого індикатора. На практиці, за наявними експериментальними матеріалами, зміна сили світла протягом тривалої експлуатації значно менша.

1.2. Фізичні ефекти, які використовуються в ЗВІ

Фізичні ефекти, придатні для використання в індикаторній техніці, виключно різноманітні. Нижче наведені основні з них.

1. Випромінювання світла вольфрамовою ниткою, яка розташована у вакуумі та розжарена пропусканням через неї електричного струму (індикатори накалювання).

2. Випромінювання світла електричним розрядом у газах (газоразрядні індикатори).

3. Передпробійна електролюмінесценція порошкових люмінофорів у змінному електричному полі (електролюмінесцентні індикатори).

4. Інжекційна люмінесценція монокристалічних напівпровідників з p-n переходами (напівпровідникові індикатори).

5. Випромінювання фотолюмінофорів, що нанесені на напівпровідникові випрмінювачі; можливі два останніх варіанта: антистоксовий люмінофор на інфрачервоному (ІЧ) випромінювачі та «звичайний» фотолюмінофор на випромінювачі синьо-фіолетового діапазону.

6. Електролюмінесценція тонких полікристалічних напівпровідникових плівок у постійному та змінному електричних полях (тонкоплівочні електролюмінесцентні індикатори).

7. Низьковольтна катодолюмінесценція (вакумні люмінесцентні індикатори).

8. Електрооптичні ефекти в рідких кристалах (рідкокристалічні індикатори).

9. Зміна забарвлення речовини при пропусканні через неї електричного струму (електрохромні індикатори).

10. Електрооптичні явища у сегнетоелектриках, що мають ефект подвійного заломлення променів (сегнетоелектричні індикатори).

11. Гальванічне осадження та розчинення тонкоплівочних металічних малюнків (електролітичні індикатори).

12. Переміщення заряджених колоїдних частинок під дією постійного електричного поля (електрофоретичні індикатори).

13. Різноманітні зворотні електро- і фотохімічні процеси (електрохімічні індикатори).

14. Зміна оптичних властивостей речовини при переході з рідкої фази в пароподібну при нагріванні електричним струмом (парорідинні індикатори).

Зіставлення цих ефектів дозволяє зробити ряд узагальнень:

– всі види індикаторів можна розділити на індикатори з активним і пасивним растрами. До першої групи належать прилади на основі світлогенераційних ефектів (1–7), прилади другої групи потребують зовнішнього підсвічування (8–14);

– у світлогенераційних індикаторах виділяються прилади з прямим (2, 3, 4, 6) і двоступінчатим (1, 5, 7) перетворенням електричної енергії в світлову;

– індикатори з пасивним растром можуть бути основані на зміні коефіцієнтів відбиття (8, 11, 12, 13. 14), пропускання (8, 12, 13), поглинання (9, 13) світла і на обертанні площини поляризації (8, 10);

– управління індикаторами може здійснюватися електричним струмом (1, 4, 5, 6, 14), напругою (2, 3, 6, 7, 8, 10, 12), зарядом (9, 11);

– в якості активних середовищ в індикаторах виступають метали (1, 11), монокристали (4, 5), тверді полікристалічні речовини (6, 9, 10), порошки (3, 7), рідини (8, 12), гази (2, 14);

– наибільше розповсюдження отримали напівпровідникові, газорозрядні та рідкокристалічні індикатори.

1.3. Структура спектра світла, що випромінюється

Спектральне сприйняття світла визначається кривою видимості (функцією відносної спектральної світлової ефективності стандартного спостерігача для денного зору) – $V(\lambda)$. Крива видимості має максимум при довжині хвилі 555 нм.

На рис. 1.6 представлена система CIE XYZ – це лінійна трьохкомпонентна кольорова модель, що заснована на результатах вимірювання людського ока. Комітет CIE (Commission Internationale de l'Eclairage) провів багато експериментів з великою кількістю людей, пропонуючи їм порівнювати різні кольори, а потім на основі отриманих даних в результаті цих експериментів побудував функції відповідності кольорів та універсальний кольоровий простір, в якому був представлений діапазон колорів, які бачить середньостатистична людина. Зовнішні криві лінії кордону – геометричне місце точок, що представляють кольори безперервного спектру.

Спектральне джерело з довжиною хвилі 555 нм і шириною спектру 1 нм має наступні координати кольору: X=0,337, Y=0,659 (рис. 1.6).

В ідеальному випадку джерело білого світла можна представити джерелом світла з прямокутним безперервним спектром в діапазоні від 360 нм до 830 нм з шириною 470 нм (рис. 1.6). Таке джерело має координати кольору X=0,333, Y=0,333 (рис. 1.6).

На рис. 1.2 показані: крива видимості $V(\lambda)$, прямокутний спектр E, прямокутний спектр W шириною 300 нм, спектр білого світла A і спектр білого кольору $D65$.

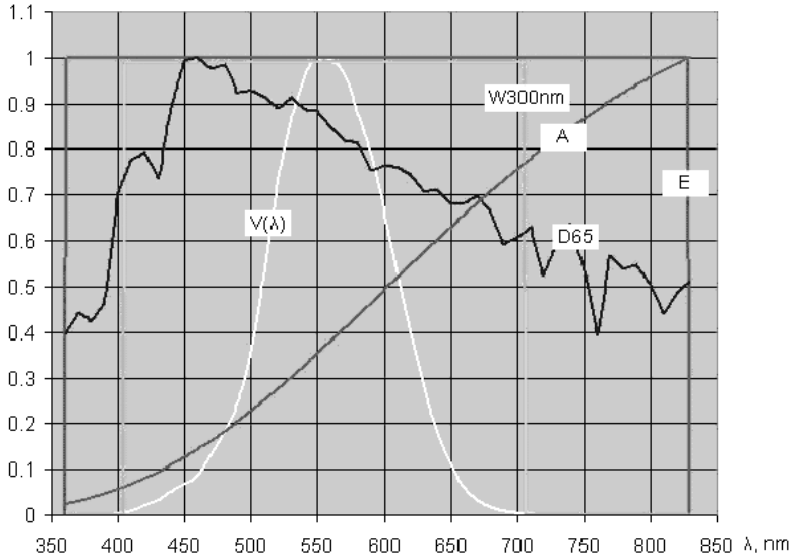
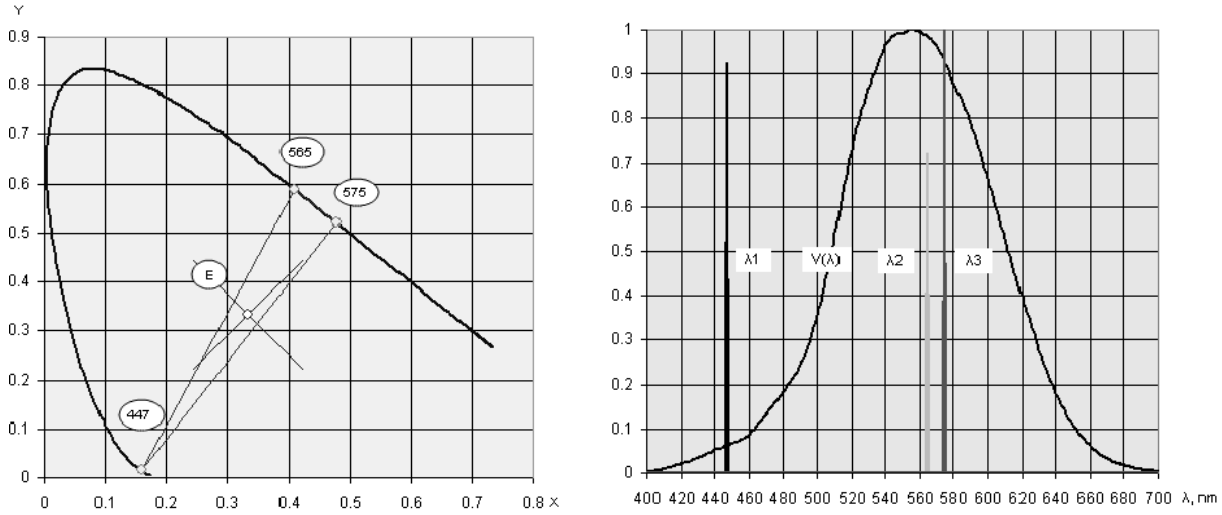


Рис. 1.2.

Біле світло можна отримати, комбінуючи довжину хвилі та інтенсивність трьох джерел світла (R – червоний, G – зелений, B – синій). Це становиться можливим через будову ока, що містить три групи спектрально-чутливих фоторецепторів, які реагують на певні ділянки спектру (400 – 500, 500 – 600 і 600 – 700 нм). Для отримання білого світла із заданими координатами кольору, необхідно щоб точка з цими координатами лежала всередині трикутника, утвореного трьома джерелами світла в стандартній колориметричній системі XYZ. На рис. 1.3 показані колірний трикутник (а), утворений трьома джерелами і спектри цих джерел (б).



а *б*

Рис. 1.3.

Для отримання білого світла можна обійтися двома джерелами світла замість трьох. Про це говорить близькість довжин хвиль другого і третього джерел. Для отримання двокомпонентного білого світла точка *E* повинна

лежати на лінії, що сполучає координати двох вибраних в XYZ системі кольорів. Таких ліній існує нескінченна множина. Варіюючи потужність F_e джерел, можна переміщувати результуючий колір по лінії, що сполучає ці джерела. При певному співвідношенні потужностей можна потрапити в точку *E*. На рис. 1.4 показана область існування двокомпонентного білого світла.

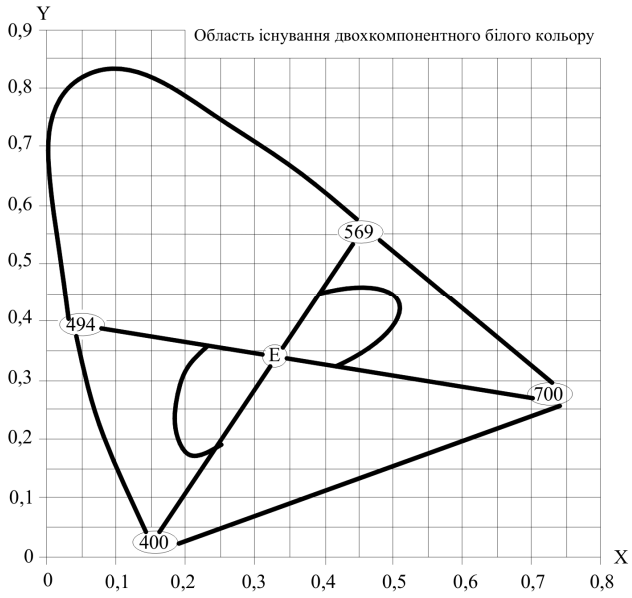


Рис. 1.4.

Граничні лінії, що проходять через точку *E*, відповідають джерелам світла 400 – 569 нм (синя лінія) і 494 – 700 нм (червона лінія).

Як приклад, на рис. 1.5 показані (у діапазоні 400 – 700 нм) спектри, що приведені в документації на повноколірний індикатор HSMF-C118 (Agilent). Діаграми спектрів завжди нормалізовані, тобто приводяться у відносних одиницях, де $F_e(\lambda_{peak}) = 1$.

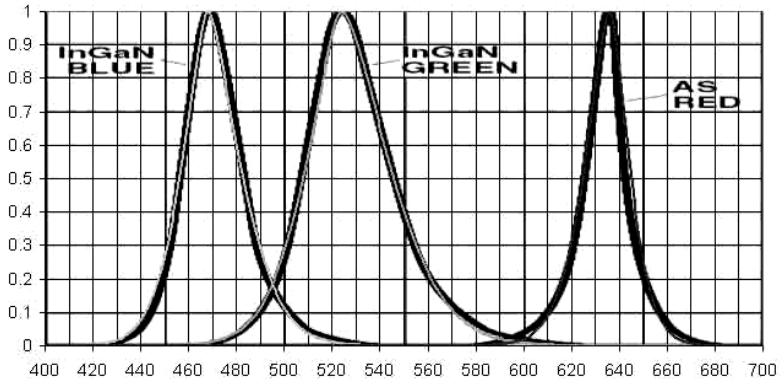


Рис. 1.5.

Повний енергетичний потік визначається площиною під кривою спектра. Для визначення площини потрібно проінтегрувати спектр:

$$Fe = \int_{\lambda=360\,нм}^{\lambda=830\,нм} Fe(\lambda)d\lambda\,.$$

При фіксованих значеннях довжин хвиль і ширини спектрів, балансування по білому кольору відбувається зміною потужності, яка подається на індикатор. Для розрахунку енергетичного балансу необхідно нормувати енергетичні потоки, а для розрахунку світлового балансу, відповідно, світлові потоки.

За спектром є можливість розраховувати колірні координати:

$$Fvx = \int_{\lambda=360\,нм}^{\lambda=830\,нм} Fe(\lambda)X(\lambda)d\lambda\,,\ Fvy = \int_{\lambda=360\,нм}^{\lambda=830\,нм} Fe(\lambda)Y(\lambda)d\lambda\,,\ Fvz = \int_{\lambda=360\,нм}^{\lambda=830\,нм} Fe(\lambda)Z(\lambda)d\lambda\,.$$

Після нормування можна визначити колірні координати в системі XYZ:

$$X = \frac{Fvx}{Fvx + Fvy + Fvz}\,,\ Y = \frac{Fvy}{Fvx + Fvy + Fvz}\,,\ Z = \frac{Fvz}{Fvx + Fvy + Fvz}\,.$$

При фіксированих значеннях довжин хвиль і ширини спектрів, балансування по білому кольору здійснюється зміною потужності, що подається на індикатор. Для розрахунку енергетичного балансу необхідно нормувати енергетичні потоки, а для розрахунку світлового балансу, відповідно, світлові потоки:

$$We = r \cdot Re + g \cdot Ge + b \cdot Be = \frac{Fer}{Fer + Feg + Feb} + \frac{Feg}{Fer + Feg + Feb} + \frac{Feb}{Fer + Feg + Feb} = 1\,,$$

де r, g, b – колірні коефіцієнти.

Якщо в індикаторі не вистачає потужності за будь-якою одною компонентою, то для формування білого світла необхідно знижувати потужність двох останніх компонент кольору.

1.3.1. Коло кольорів

Відкриття кольорового спектру і неспектральних кольорів відкрило можливість до представлення усього різноманіття кольорів певною сумішшю основних кольорів. Німецький математик Герман Грассман встановив, що при дотриманні певних умов будь-який колір можна представити як суму трьох лінійно незалежних кольорів.

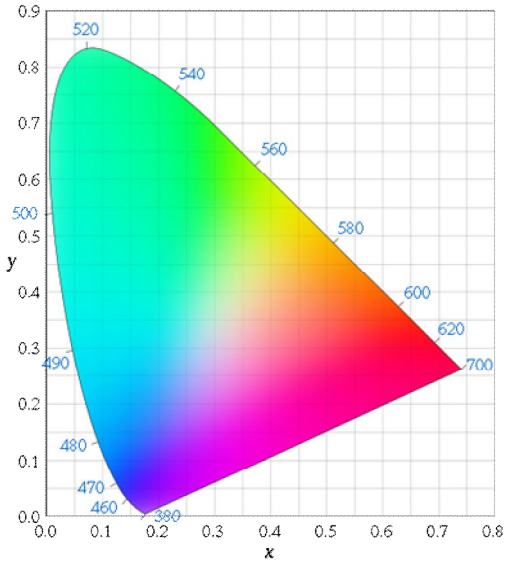


Рис. 1.6.



Рис. 1.7.

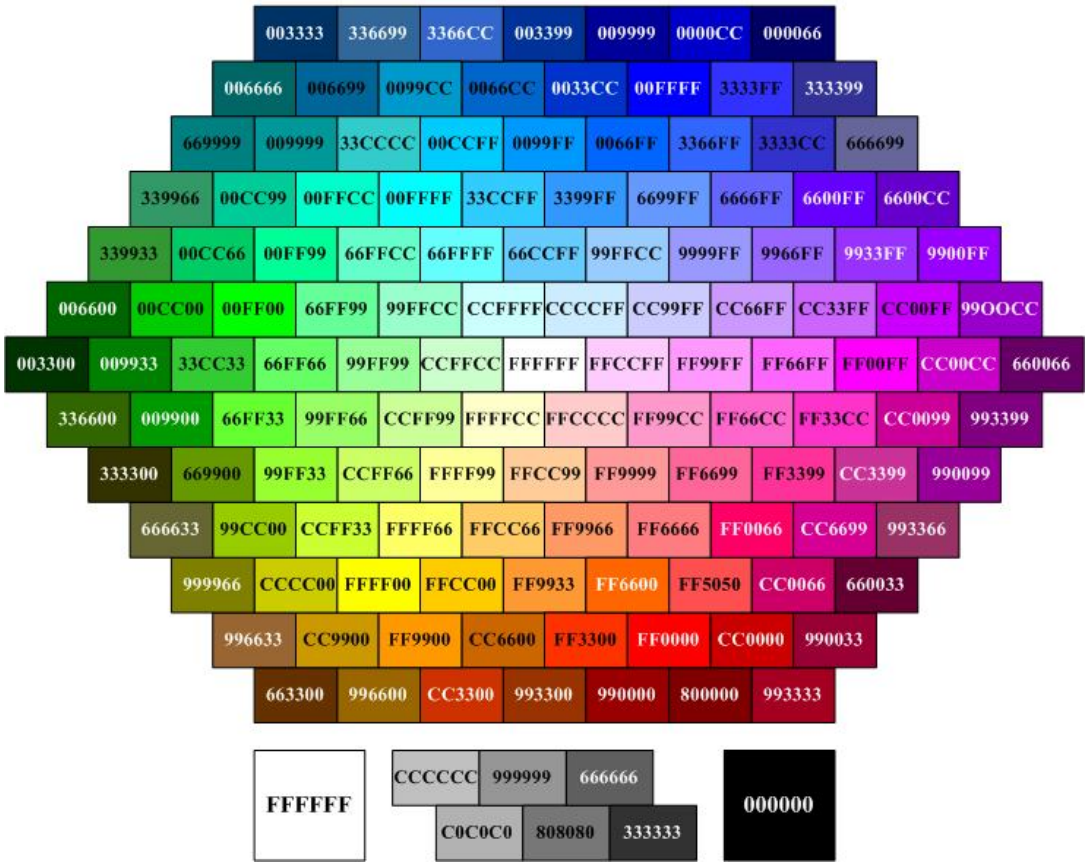
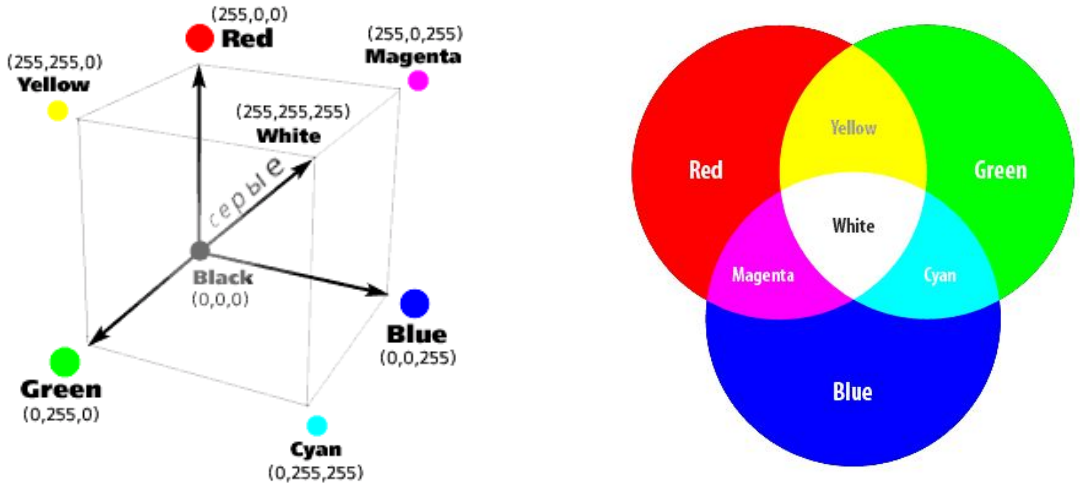
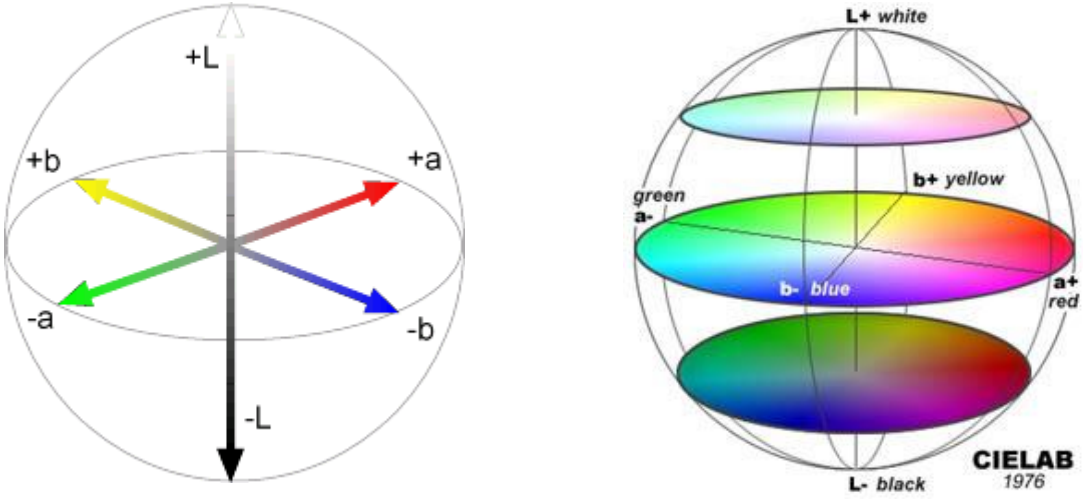


Рис. 1.8.



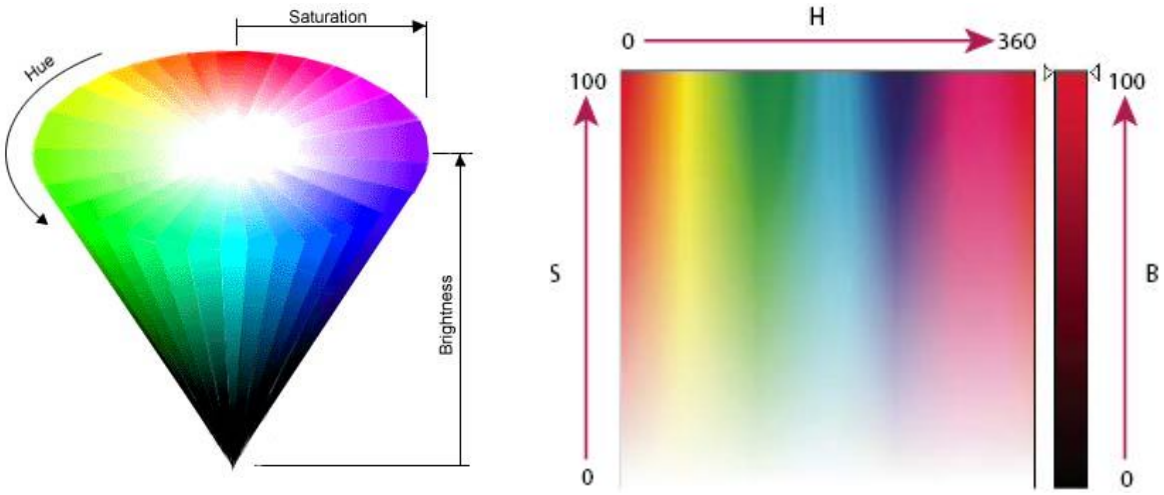
а *б*

Рис. 1.9.



а *б*

Рис. 1.10.



а *б*

Рис. 1.11.

В Англії основними кольорами довго вважали червоний, жовтий і синій, лише в 1860 р. Максвелл запровадив адитивну систему RGB (червоний, зелений, синій). У 1931 році асоціація CIE (фр. Commission Internationale de l'Eclairage) прийняла стандартну модель «універсального кольорового простору», що отримала назву XYZ, або «нормальної кольорової системи». Ця система наразі домінує в системах відтворення кольорів для моніторів і телевізорів як така, яку легко забезпечити трьома потоками світла. В 1951 р. Енді Мюллер запропонував субтрактивну систему CMYK (Синьо-зелений, пурпурний, жовтий, чорний), яка мала переваги у поліграфії та кольоровій фотографії.

У цифрових технологіях, як і в аналоговому телебаченні, застосовують RGB-системи. При цьому кожен компонент кольору представляється за допомогою цілих чисел у межах деякого діапазону, як правило, від 0 до ступеню числа 2 мінус один ($2N - 1$), що відповідає максимально можливій кількості значень, що може бути закодована певною кількістю бітів.

Найбільше поширення отримала технологія «truecolor», що передбачає 24-бітне кодування кольору. Інтенсивність кожного компоненту в цій системі кодується за допомогою 8 бітів і, таким чином, може приймати 256 значень – від 0 до 255. Загалом же «truecolor» дозволяє представити 16777216 (тобто 2^{24}) кольорів для кожного пікселя, що вважається достатнім для реалістичного зображення кольору.

Коло кольорів – абстрактний ілюстративний спосіб відображення безперервності переходів між кольорами. Сектори кола пофарбовані в різні кольорові тони, розміщені в порядку розташування спектральних кольорів, причому пурпуровий колір пов'язує крайні (червоний і фіолетовий) кольори. На рис. 1.7 представлене коло кольорів.

1.3.2. Оцифровування зображень

З появою графічних станцій у вигляді комп'ютера розпочався новий етап в освоєнні ПК – машини як засобу обробки графічної інформації. Адже комп'ютер здатний не тільки вирішувати обчислювальні завдання, але і показувати, візулізувати будь-які процеси на дисплеї. Графічний інтерфейс користувача став стандартом програмного забезпечення різних програмних областей. Будь-яка графіка представляється в аналоговій або дискретних формах.

Аналогова форма зображення – це мальовниче полотно, дискретне зображення – малюнок, виконаний на струменевому принтері, яке складаєтсья з множини різнобарвних маленьких крапок.

Зображення з аналогової форми (папір, кіноплівка) у цифрову (дискретну) форматується шляхом дискретизації – наприклад, шляхом сканування. Більш детально принципи роботи різних типів сканерів розглянуті в розділі 9.

Після дискретизації зображення виконується його кодування. Кодування – це перетворення символів однієї знакової системи в іншу. Всі комп'ютерні зображення розділяють на два основні типи: растрові і векторні.

В растровій графіці зображення в процесі кодування розбивається на маленькі крапки або пікселі (рис. 1.12). Кожному пікселю привласнюється код кольору. Інформація про кожну таку крапку містить комп'ютерна відеопам'ять. Векторна графіка створюється із примітивних об'єктів таких як лінія, крива, крапка, прямокутник, трикутник, коло тощо. Ці елементи (і їх обсяг) описуються за допомогою математичних формул.

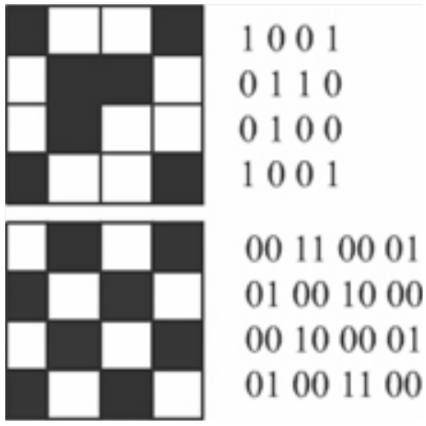


Рис. 1.12.

Отже, графічна інформація являє собою зображення, побудоване з конкретного числа крапок (пікселів) – це і є просторова дискретизація. Цей процес можна зрівняти із зображеннями, побудованими з мозаїки. У такому зображенні кожній мозаїці (точці зображення) привласнюється свій код кольору. Чим більше точок тим якісніше зображення. Чим менша за розміром крапка, тим більше роздільна здатність, а значить знову – вища якість зображення. Якість зображення залежить і від кількості кольорів. Чим більше несе в собі інформацію кожна точка про кількість кольорів, тим розширенішою буде палітра кольорів зображення.

При кодуванні кольору відбувається його розкладання на основні складові – їх три (червоний, синій, зелений). Змішуючи ці кольори виходять різні відтінки. У процесі дискретизації використовуються різні палітри.

Кожний колір розглядається як майбутній стан точки. Кількість кольорів N і кількість інформації I, зв'язані між собою і обчислюються по формулі: $N=2^I$. Такі вихідні дані необхідні для кодування кольору кожної окремо точки. Наприклад, у найпростішому випадку палітра кольорів складається із двох – чорного і білого. Значить кожна точка на екрані приймає або «чорний» або «білий» стан. По заданій формулі можна обчислити кількість інформації, необхідну для кодування кольору кожної

точки. Кількість біт, необхідних для кодування кольору пікселя – це глибина кольору. Кольорові зображення будуються із двійковим кодом кольору кожної точки. Ця інформація зберігається у відеопам'яті. Такі зображення змінюють глибини кольору – 8, 16, 24 і 32 біта.

Кольорова картинка отримується на екрані шляхом змішування трьох базових кольорів: червоного, зеленого та синього. Ця колірна модель називається RGB (по перших буквах англійських назв кольорів – Red, Green, Blue). Для одержання множини відтінків (рис. 1.9) основним (або базовим) кольорам задаються різні інтенсивності (яскравість, контрастність, насиченість).

Наприклад, при глибині кольору в 24 біта (по 8 біт на кожний колір або канал) виходить $N = 2^8 = 256$ рівнів інтенсивності, задані двійковими кодами – від 00000000 (мінімальної) до 11111111 (максимальної).

Розглянемо формування растрового зображення на екрані монітора, розміром 800×600 точок. Розраховуємо необхідний обсяг відеопам'яті для графічного режиму 800×600 і глибиною кольору в 24 біта. Усіх точок на екрані 480000 (800 множимо на 600). Обчислюємо необхідний обсяг відеопам'яті: 24 біт × 480000 = 11520000 біт = 1,37 Мб.

Аналогічно розраховуються обсяги відеопам'яті і для інших популярних графічних режимів – 1024×768, 1280×1024, 1440×900, 1600×1200 і т.д.

1.3.3. Кольорові моделі

Колірна модель – це модель конкретизованої класифікації гами світлових кольорів сприйнятних для людини, котра дає можливість класифікувати конкретний колір для подальшої можливості його відтворення [40].

RGB – адитивна колірна модель, що описує спосіб синтезу кольору, за якого червоне, зелене та синє світло накладаються разом, змішуючись у різноманітні кольори. У даній моделі колір кодується градаціями складових каналів (Red, Green, Blue). Тому для збільшення величини градації визначеного каналу зростає його інтенсивність під час синтезу [41].

Кількість градацій кожного каналу залежить від бітового значення RGB. Зазвичай використовують 24-х бітну модель, у якій визначається по 8 біт на кожен канал, і тому кількість градацій дорівнює 255, що дозволяє закодувати 255^3=16,5 млн. кольорів. Колірна модель RGB призначена сприймати, представляти та відображати зображення в електронних системах, таких як телебачення та комп'ютери [41].

Переваги моделі:

– апаратна близькість із монітором, сканером, проектором, іншими пристроями;

– велика кольорова гамма, близька до можливостей людського зору;
– доступність багатьох функцій обробки зображення (фільтрів) у програмах растрової графіки;
– невеликий (порівняно до моделі CMYK) обсяг, проте ширший спектр кольорів.

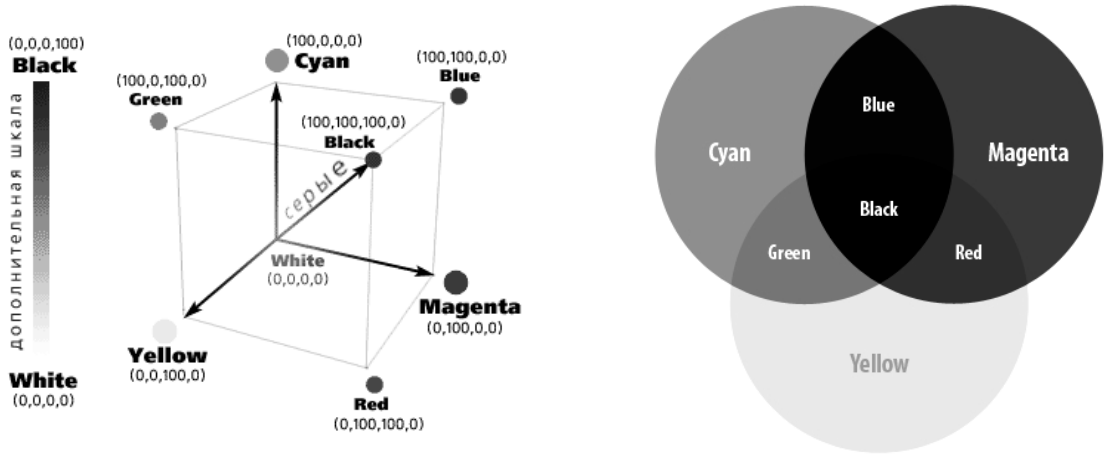
Модель RGB може бути представлена у вигляді кубу (рис. 1.9, а). На рис. 1.9, б, представлена схема змішування кольорів для моделі RGB.

CMYK (Cyan, Magenta, Yellow, Key color) – субтрактивна колірна модель, яка використовується у поліграфії, перш за все при багатофарбовому (повноколірному) друці. Вона застосовується у друкарських машинах і кольорових принтерах [42].

Оскільки модель CMYK застосовують в основному в поліграфії при кольоровому друці, а папір та інші друкарські матеріали є поверхнями, що відбивають світло, зручніше рахувати, яка кількість світла відбилася від тієї або іншої поверхні, ніж скільки поглинулося. Таким чином, якщо відняти з білого три первинні кольори (RGB), то отримаємо трійку доповнюючих кольорів CMY. Модель CMYK забезпечує менше колірне охоплення, ніж адитивна модель RGB [42].

Згідно з теорією кольору, практично будь-який відтінок можна отримати за допомогою лише трьох кольорів – наприклад Cyan, Magenta і Yellow. Зокрема чорний колір – змішуванням згаданих кольорів в рівній пропорції і з максимальною інтенсивністю. На практиці через недосконалості пігментації фарб стовідсоткове змішування цих трьох кольорів дає скоріше брудно-коричневий або брудно-сірий колір; тріадні фарби не дають тієї глибини і насиченості, яка досягається використанням чорної фарби. Оскільки чистота і насиченість чорного кольору є надзвичайно важливою в друкарському процесі, його було введено в цю колірну модель.

Кожен колір в CMYK описується сукупністю чотирьох чисел, які називають колірними координатами. Кожне з цих чисел є відсотком фарби даного кольору у складовій колірної комбінації. Приклад: для отримання темно-помаранчевого кольору слід змішати 30% фарби cyan, 45% фарби magenta, 80% фарби yellow і 5% кольори black. Цей колір можна записати таким чином: (30, 45, 80, 5). Іноді користуються іншим позначенням: C30M45Y80K5. Модель CMYK може бути представлена у вигляді кубу (рис. 1.13, а). На рис. 1.13, б, представлена схема змішування кольорів для моделі CMYK.



а *б*

Рис. 1.13.

Lab – система задання кольорів, що використовує як параметри світлосилу, відношення зеленого до червоного та відношення синього до жовтого (рис. 1.10). Ці три параметри утворюють тривимірний простір, точки якого відповідають певним кольорам.

Колірна модель Lab розроблялась як апаратно-незалежна, тобто вона задає кольори без врахування особливостей відтворення кольорів. Має три параметри для опису кольору: світлосилою L (англ. Lightness) і двома хроматичними параметрами. Перший (умовно позначений латинською буквою a) вказує на співвідношення зеленої і червоної складової кольору, другий (позначений буквою b) – співвідношення синьої і жовтої складової.

HSB – колірна модель, що використовується тільки для оформлення векторних і текстових об'єктів документа. Описує кольоровий простір, заснований на трьох характеристиках кольору: кольоровому тоні (Hue), насиченості (Saturation) і яскравості (Brightness). На рис. 1.11 представлені кольорові координати моделі HSB.

1.4. Класифікація засобів відображення інформації

Розрізняють два основні класи індикаторних елементів: світловипромінювальні та такі, що модулюють світло (світломодулюючі), тобто ті, що змінюють параметри середовища, через яке проходить світло.

Відповідно до цього на рис. 1.14 приведена класифікація основних типів елементів, що відрізняються за фізичними принципами, що використовуються.

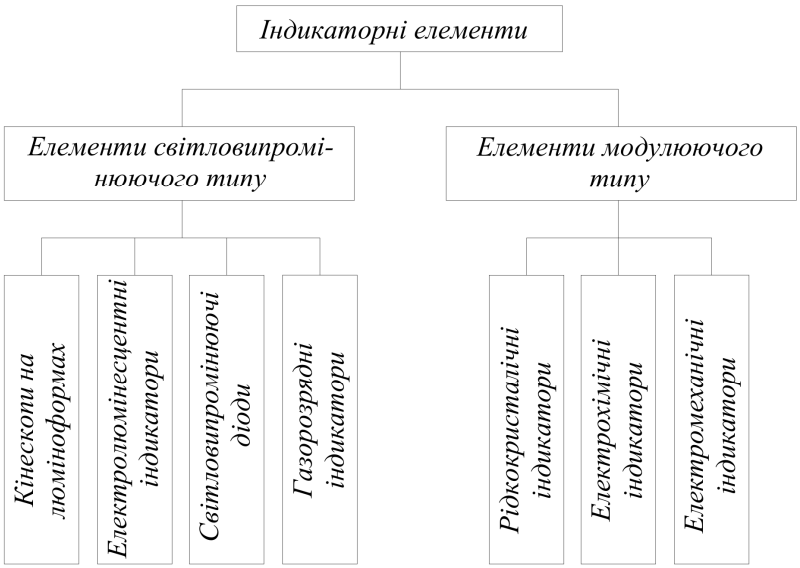


Рис. 1.14.

Індикатори можуть бути класифіковані за типом інформації, що відображається, за типом інформаційного поля і за способом управління.

Класифікація сучасних напівпровідникових індикаторів за вказаними класифікаційними ознаками приведена на рис. 1.15.

Одиничні індикатори складаються з одного елементу відображення і призначені в основному для представлення інформації у вигляді точки або іншої геометричної фігури.



Рис. 1.15.

Шкальні індикатори мають елементи відображення у вигляді правильних прямокутників і призначені для відображення інформації у вигляді рівнів, або значень величин. Окрему групу шкальних індикаторів складають так звані лінійні формувачі зображення у високонадійних оптоелектронних реєстраторах оперативної аерокосмічної інформації на фотоплівку.

Цифрові індикатори складаються з елементів відображення у вигляді сегментів і призначені для відображення цифрової інформації та окремих букв алфавіту. Буквено-цифрові індикатори призначені для відображення інформації у вигляді букв, цифр, різних знаків. Одиничні елементи відображення таких індикаторів згруповані по рядках і стовпцях.

Графічні (матричні) індикатори дозволяють збирати модулі з елементів екрану різного розміру без втрати кроку. Графічні індикатори призначені для відображення будь-якої інформації. Цифрові і буквено-цифрові індикатори бувають одно- та багаторозрядні. Під однорозрядним розуміється індикатор, що має одне знакомісце, тобто інформаційне поле індикатора або його частину, необхідну і достатню для відображення одного знаку. Багаторозрядний індикатор має декілька фіксованих знакомісць. Цифрові, буквено-цифрові, матричні і шкальні індикатори можуть бути без управління і з вбудованими схемами управління. Принципи управління перерахованими індикаторами, а також схемотехніка, будуть розглядатися в наступних розділах.

1.5. Ергономічні характеристики ЗВІ

Ефективність сприйняттня інформації залежить від наступного:

− типу символів;

− форми і кутових розмірів символів;

− рівня яскравості і контрастності між зображенням і фоном;

− кольорів умовних знаків, що відтворюються індикатором;

− рівня зовнішньої освітленості;

− величини кутів огляду і відстані до лицьових панелей.

Для оцінки якості відображення інформації використовують інтегральну характеристику, яка носить назву «читаність». Остання величина оцінюється такими показниками, як швидкість і точність розрізнення, розпізнавання і інтерпретації символів. Читаність елементів ЗВІ залежить від правильної розробки окремих її деталей.

Найважливішим є вибір алфавіту символів, що використовуються як коди: геометричних фігур, літер, цифр, символів. Форма символів впливає на швидкість і точність розпізнавання.

Товщина ліній символів залежить від освітленості і контрастності символів з фоном. При відтворенні білих цифр на чорному фоні товщину ліній рекомендується приймати рівною 1/10 висоти цифр. При відтворенні чорних цифр на білому фоні товщину ліній вибирають рівною 1/6 висоти

цифр. Ширина цифр повинна складати 2/3 висоти. Високі, вузькі цифри розпізнаються краще при слабкому освітленні. Розміри знаків повинні відповідати рекомендованій відстані до спостерігача. Залежно від відстані зорового сприйняття L висоту знаку h вибирають на основі наступної нерівності:

$$h \geq 2L \cdot tg\frac{\alpha}{2}.$$

Час зчитування інформації непостійний і залежить від кутових розмірів символів, світлових характеристик, складності знаків, що відображаються на індикаторі. Чим складніший символ, тим більше часу потрібно для його розпізнавання. Для оптимального розпізнавання знаків простої складності рекомендують приймати їх кутовий розмір $\alpha = 18' \pm 1'$; знаків середньої складності $\alpha = 21'$; для складних знаків $\alpha = 35'$. При використанні знаків, що мають розміри менше допустимих, час сприйняття і число помилок зростають.

Рекомендований рівень яскравості світіння в Кд/м^2 при зовнішній освітленості у 200 Лк наведено в таблиці 1.3.

Табл. 1.3.

Колір	Оптимальна яскравість, Кд/м^2	Максимальна яскравість, Кд/м^2
Білий	100	175
Червоний	20	45
Помаранчевий	65	110
Жовтий	70	120
Зелений	30	55
Голубий	35	80
Синій	10	25

Для найкращого розрізнення символів, що відображаються, застосовують кодування кольором. Оптимальним кольором вважається жовто-зелений, найбільш простим – білий. Ці кольори і рекомендуються для застосування в ЗВІ. Кодування кольором використовують також для відтворення сигналів небезпеки. Міру небезпеки позначають різним кольором: червоним – зупинка, припинення роботи устаткування, помаранчевим або жовтим – попередження.

1.6. Інформаційна модель та формування її елементів

У ЗВІ інформація представляється інформаційною моделлю (ІМ) – організованим відповідно до певної системи правил відображенням станів об'єкту управління, зовнішнього середовища і способів дії на них. Під ІМ розуміють сукупність сигналів, організованих відповідно до певної системи правил і створюючих відображення об'єкту, яким управляють. На

основі сприйняття ІМ в свідомості оператора формується образ стану керованого об'єкту, який називають концептуальною моделлю. Представлення фізичного стану однієї системи фізичним станом іншої називають кодуванням. У ІМ в закодованій формі представляється суть протікання реальних процесів, явищ, об'єктів. Кодування інформації в ІМ здійснюється за допомогою елементів інформаційної моделі (ЕІМ), в якості яких використовують букви, умовні знаки (символи), геометричні фігури, лінії, точки і т.д. Набір елементів ІМ, що використовуються, складає алфавіт інформаційної моделі. Число елементів, що створюють алфавіт, називають основою коду алфавіту N_a. До складу алфавіту можуть включатися і такі ознаки ІМ, як колір, градації яскравості, розмір, орієнтація та ін. Основа коду повного алфавіту, що включає всі кодові ознаки, не повинна перевищувати розумної межі 200-400. Подальше його збільшення ускладнює роботу оператора.

При представленні елементів алфавіта ІМ двійковим кодом число його разрядів n_a визначають з умови:

$$n_a \geq \left[\log_2 N_a\right],$$

де N_a – основа коду повного алфавіту інформаційної моделі;

Інформаційна ємність визначає кількість інформації, яка може бути одночасно представлена на інформаційному полі ЗВІ. Інформаційна ємність алфавітно-цифрових ЗВІ задається кількістю знаків в текстовому рядку $N_{зтс}$ і числом текстових рядків $N_{тс}$.

Інформаційну ємність графічних ЗВІ часто оцінюють сумарною довжиною ліній, або кількістю точок, що відтворюються. Частина простору, в межах якого відбувається формування інформаційної моделі, називається інформаційним полем (ІП).

Відношення ширини інформаційного поля B до його висоти H називається форматом інформаційного поля $K_ф$:

$$K_{ô} = \frac{B}{H}.$$

В залежності від алфавіта, що використовються, виділяють наступні основні типии інформаційних моделей:

− буквено-цифрові;

− графічні;

− напівтонові;

− комбіновані.

У буквено-цифрових моделях використовують букви, цифри, умовні знаки (символи), а властивості об'єкту або процесу, що відображається, представляють у вигляді буквеного тексту, цифрової комбінації, формул,

таблиць. При побудові буквено-цифрових ІМ все інформаційне поле розбивають на окремі знакомісця (рис. 1.16).

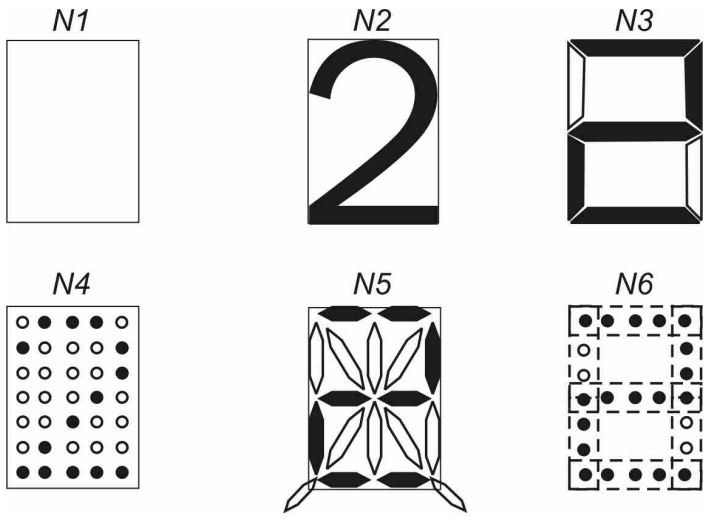


Рис. 1.16.

Найпростіший елемент інформаційної моделі, який може бути реалізований вибраним типом індикатора, називають елементом відображення (ЕВ). ЕВ характеризується формою, геометричними розмірами, яскравістю, часом післясвітіння, кольором і т.д.

Відповідно до елементів відображення, що використовуються, всі способи формування знаків можна розділити на три основні групи:

1) знакомоделюючий спосіб характеризується цілісним представленням знаку, при цьому форма елементу відображення збігається з контуром знаку, наприклад цифри 2 в знакомісці *N*2 (рис. 1.16);

2) знакосинтезуючий спосіб характеризується тим, що знаки формуються з більш простих елементів відображення. У знакомісці *N*3 на рис. 1.16 показаний синтез цифри 2 з сегментів.

Набір сегментів у знакомісці складає деяку узагальнену фігуру – поліграму. З 7-сегментної поліграми можна синтезувати всі арабські цифри і деякі букви. Розширення алфавіту досягається за рахунок збільшення числа сегментів в поліграмі. Наприклад, за допомогою 18-сегментної поліграми можна синтезувати букви російського і латинського алфавітів (знакомісце *N*5).

У знакомісці *N*5 показаний синтез тієї ж цифри 2 з точкових ЕВ, розміри яких набагато менше розмірів знаків, що синтезуються. В межах знакомісця точкові елементи відображення утворюють матрицю знаку. Число елементів відображення в матриці знаків вибирають, виходячи з вимоги безпомилкової і швидкої ідентифікації (розпізнавання) всіх знаків алфавіту. Так, наприклад, матриця 5×7 (5 стовпців і 7 рядків) є практично мінімально прийнятною для синтезу букв російського і латинського алфавітів і цифр. Для синтезу лише арабських цифр розмірність матриці можна зменшити до 3×5.

Елементи відображення можуть виконуватися у вигляді окремих конструктивних елементів, наприклад електричної лампи розжарювання, світлодіода, катода газоподібної лампи, виконаного у формі цифри або сегменту. Такі елементи відображення називають дискретними. В електронно-променевих приладах елементи відображення, що входять в знак, генеруються електронним променем в процесі відтворення зображення. Синтез знаків з отриманих таким чином елементів називають знакогенеруючим способом формування знаків.

Графічні інформаційні моделі (ГІМ) представляють кресленнями, діаграмами, схемами (структурними, функціональними, монтажними) і т.д. Основними елементами інформаційної моделі при побудові графічної ІМ є лінії, точки, двовимірні області. Найбільш універсальними елементами відображення (ЕВ), з яких формуються елементи ІМ є точкові елементи відображення. Кожен точковий ЕВ, що входить до формованої моделі, має бути заданий координатами x_i, y_i, що визначають його положення на інформаційному полі.

Око не помічає дискретного характеру зображення, якщо кутовий розмір ЕВ близький до граничного кута, під яким людина розрізняє дві роздільні точки. Зменшення розмірів ЕВ при збереженні розмірів інформаційного поля призводить до збільшення загального числа ЕВ і, відповідно, до технічного ускладнення засобів відображення інформації.

Напівтонові інформаційні моделі використовують широкий діапазон градацій яскравості, що дозволяє забезпечити наочний характер формованих зображень (фотографії, діапроєкції, кіно- і телевізійні зображення). Комбіновані інформаційні моделі складають з компонентів моделей різних класів.

Питання до розділу 1

1. Назвіть основні параметри індикаторів.
2. Дайте визначення поняттям «яскравість» та «контрастність» індикаторів.
3. Що таке кут огляду?
4. Якими параметрами характеризується індикатор як елемент електричного ланцюга?
5. Які фізичні ефекти використовуються в засобах відображення інформації?
6. Назвіть основні кольорові моделі?
7. Як кодуються кольори зображення?
8. Дайте класифікацію засобам відображення інформації.
9. Що таке коло кольорів?
10. Яким чином можна отримати білий колір?
11. Що таке інформаційна модель?
12. На що впливають ергономічні характеристики індикатора?

2. НАПІВПРОВІДНИКОВІ ІНДИКАТОРИ

Напівпровідникові індикатори є одним з видів знакосинтезуючих індикаторів (ЗСІ), під якими розуміють прилади, де інформація, що призначена для зорового сприйняття, відображається за допомогою одного, або сукупності дискретних елементів. Напівпровідникові індикатори є активними знакосинтезуючими індикаторами, в яких використовується явище інжекційної електролюмінесценції.

Високі технічні характеристики напівпровідникових індикаторів (НПІ) забезпечили їх успішне впровадження як елементів індикації в апаратурі, що використовується в приладах управління стаціонарним виробничо-технологічним устаткуванням, рухомими об'єктами, об'єктами побутового призначення та ін. Одною з переваг напівпровідникової технології індикаторів є можливість їх конструктивного виконання у вигляді уніфікованих модулів, що забезпечують можливість безшовного з’єднання. Крім того, модульність виконання індикаторів гарантує високу ремонтопридатність пристроїв відображення інформації. Іншою не менш важливою перевагою НПІ є сумісність рівнів управляючої напруги і струму, що споживається з напругою логічних рівнів і струмом мікросхем. Це дозволяє значно скоротити об'єми схем управління елементами індикації і підвищити надійність індикаторних пристроїв і систем за рахунок використання елементної бази, виконаної лише за напівпровідниковою технологією.

2.1. Класифікація напівпровідникових індикаторів

Напівпровідникові індикатори можуть бути класифіковані по вигляду інформації, що відображається, по вигляду інформаційного поля і за способом управління. Класифікація сучасних напівпровідникових індикаторів за вказаними класифікаційними ознаками приведена на рис. 2.1.

Одиничні індикатори складаються з одного елементу відображення і призначені в основному для представлення інформації у вигляді точки, або іншої геометричної фігури.

Шкальні індикатори мають елементи відображення у вигляді правильних прямокутників і призначені для відображення інформації у вигляді рівнів або значень величин.

Цифрові індикатори складаються, як правило, з елементів відображення у вигляді сегментів, і призначені для відображення цифрової інформації та окремих букв алфавіта. Буквенно-цифрові індикатори призначені для відображення інформації у вигляді букв, цифр, різних знаків. Одиничні елементи відображення таких індикаторів згруповані по рядкам і стовбцям.



Рис. 2.1.

Графічні (матричні) індикатори дозволяють збирати модулі з елементів екрана різного розміру без втрати кроку. Графічні індикатори призначені для відображення будь-якої інформації.

Цифрові і буквенно-цифрові індикатори бувають одно- і багаторозрядні. Під одноразрядним розуміють індикатор, що має одне знакомесце. Багаторозрядний індикатор має декілька фіксованих знакомісць. Цифрові, буквено-цифрові, матричні та шкальні індикатори можуть бути без управління та з вбудованими схемами управління.

2.2. Фізичні основи формування оптичного випромінювання

Напівпровідникові світловипромінюючі діоди – це клас твердотілих приладів, в яких електрична енергія безпосередньо перетворюється на світлову.

Генерація світлового випромінювання забезпечується двома шляхами: за допомогою нагрівання (теплове випромінювання) і за допомогою одного з видів люмінесценції (люмінесцентне випромінювання, яке є основою сучасних напівпровідникових світловипромінюючих діодів).

У будь-якому випадку процес випромінювання складається з двох етапів. На першому етапі має місце накопичення енергії, завдяки якому генеруються носії заряду і переходять в зону вільних рівнів. На другому етапі носії шляхом рекомбінації віддають свою енергію. Випромінювання квантів світла відбувається за рахунок переходу електрона на нижчий рівень – безпосередньо або через рекомбінаційну пастку. Тобто можливі різні переходи електронів: міжзонний перехід – перехід із зони вільних

рівнів у валентну зону; перехід із зони вільних рівнів на рівень домішок (рекомбінаційну пастку), яка знаходиться в забороненій зоні, а потім у валентну зону. Відповідно, рекомбінації в таких напівпровідниках відбуваються випромінюванням фотону, довжина хвилі якого визначається як:

$$\lambda = 1{,}23/W_3 \, . \qquad (2.1)$$

Формула (2.1) використовується для вибору матеріалу світловипромінюючих приладів за шириною забороненої зони. З неї виходить, що для роботи в діапазоні видимого випромінювання необхідно, щоб $1{,}5 \le W_3 \le 3$ еВ.

У прямозонних сучасних напівпровідниках (GaAs, GaAlAs, GaAs) вірогідність прямих оптичних переходів досить висока, і тому вони широко використовуються для виготовлення світловипромінюючих приладів.

У непрямозонних напівпровідниках (Ga) рекомбінація відбувається через центри рекомбінації в два етапи: спочатку в центрі рекомбінації проходить локалізація носіїв одного знаку, а потім його рекомбінація з вільними носіями іншого знаку. В якості центрів рекомбінації використовують, наприклад, атоми домішок Ga в гратках. Довжина хвилі випромінювання при непрямих переходах буде більшою, ніж у прямозонних. Слід зазначити, що поряд з випромінюванням квантів світла, має місце і процес самопоглинання. У прямозонних напівпровідниках рівень самопоглинання значно більший, ніж в непрямозонних. Незважаючи на цей недолік, перші знаходять значно ширше застосування.

У напівпровідникових приладах генерація оптичного випромінювання забезпечується за допомогою інжекційної люмінесценції, тобто для збудження електронів використовується енергія електричного поля, яке прикладається до *p*-*n*-переходу в прямому напрямі. При цьому в напівпровідникових структурах за рахунок інжекції створюється надлишкова концентрація носіїв. На рис. 2.2. показана область випромінювання

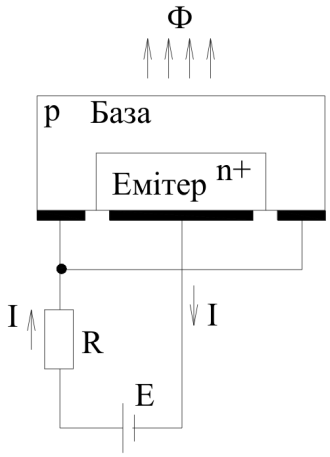


Рис. 2.2.

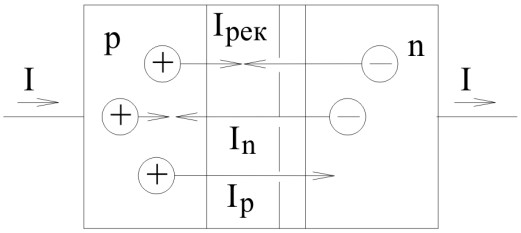


Рис. 2.3.

Випромінюваною є лише область по одну сторону від p-n переходу (переважно p-область) (рис. 2.2). У цій області має бути максимальна кількість інжекційних носіїв. Для виконання такої умови в *n*-області має бути значно більша концентрація основних носіїв, ніж в p-області.

Аналізуючи процеси в такій структурі, можна виділити наступні складові струму *I*, показані на рис. 2.3: рекомбінаційна складова $I_{\text{рек}}$, яка обумовлена рекомбінаціями в *p*-*n*-переході і не призводить до створення фотонів (за рахунок такої рекомбінації виділяються теплові кванти); електронна складова I_{n}, яка призводить до рекомбінацій в області бази і створення фотонів; діркова складова I_{p}, обумовлена переходом дірок з *p*-області в *n*-емітер, і також не створює фотони. Відношення I_{n} до сумарного (повного) струму

$$I = I_{\text{n}} + I_{\text{рек}} + I_{\text{p}}$$

називається коефіцієнтом інжекції, який оцінює теоретичну межу ефективності світловипромінювання:

$$\gamma = I_{\text{n}} / I \,,$$

Практично межа ефективності має ще меншу величину, оскільки в напівпровідниках існують механізми рекомбінацій без випромінювання,

які зменшують величину γ, і оцінюються коефіцієнтом, який називається внутрішнім квантовим виходом. В результаті світловипромінююча ефективність *p*-*n*-переходу визначається добутком γ·η.

Особливості світловипромінюючих властивостей гетероструктур полягають в наступному. По-перше, однобічна інжекція не пов'язана з рівнями концентрацій носіїв в емітерній і базовій областях, як це має місце в звичайних *p*-*n*-переходах. Тому вона зберігається до значних цільностей струму приладу, який дає можливість значно підвищити ефективність випромінювання в оптичному діапазоні. По-друге, оптичні властивості емітера і бази істотно відрізняються, тобто спектральна характеристика випромінювання бази зміщена у бік довгих хвиль по відношенню до спектральних характеристик поглинання широкозонного емітера. Тому випромінювання бази може виводитися через емітер без поглинання.

2.3. Типи світловипромінюючих діодів

Світловипромінюючі діоди, які випускаються промисловістю, за своєю конструкцією можуть бути розділені на наступні групи:

1) в металоскляному корпусі (рис. 2.4, а);
2) в конструкції з полімерною герметизацією на основі металоскляної ніжки, або рамочного утримувача (рис. 2.4, б);
3) безкорпусні (рис. 2.4, в).

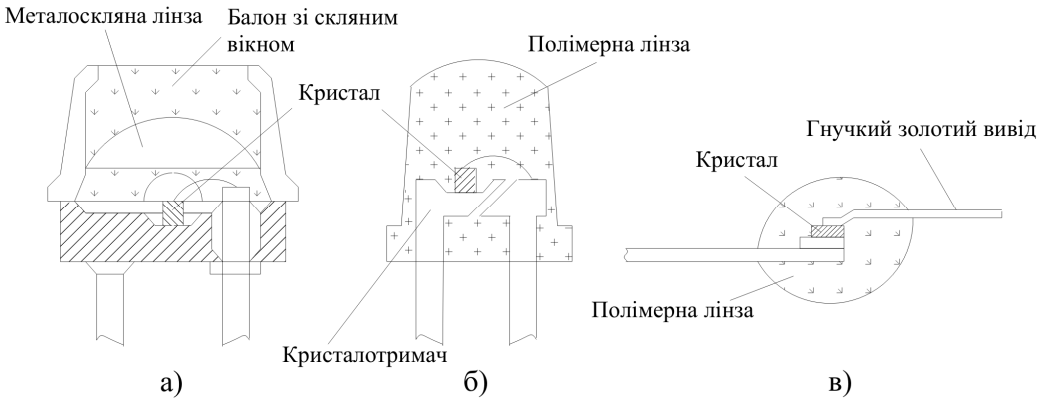


Рис. 2.4.

Перші з них характеризуються високою надійністю і стабільністю параметрів, механічною і кліматичною стійкістю. Другі мають такі переваги: за рахунок полімерного герметичного корпусу можливе забезпечення перерозподілу світлового потоку як у напрямі звуження діаграми спрямованості, так і у напрямі її розширення; полімерна герметизація збільшує зовнішній квантовий вихід випромінювання за рахунок збільшення кута повного внутрішнього віддзеркалення на межі кристал-полімер; велика стійкість до ударних і вібраційних навантажень; можливо отримати менші габарити; простіші в технології виготовлення, і

тому мають меншу вартість. Безкорпусні світлодіоди використовуються в герметичній апаратурі.

Наявність ізолюючого корпусу приводить до додаткового зниження ефективності випромінювання за рахунок поглинання в корпусі. Тому кожен корпус світлодіода характеризується оптичною ефективністю, або коефіцієнтом виходу світла $\eta_{ОПТ}$, а якість світлодіода визначається зовнішнім квантовим виходом:

$$\eta_С = \eta_{ОПТ} \cdot \gamma \cdot \eta.$$

2.3.1. Схеми включення одиничних світлодіодів

Найбільш поширеними застосуваннями напівпровідникових одиничних індикаторів є індикація стану інтегральних схем і апаратури в цілому, підсвічування написів і кнопок, створення шкал і табло, випромінювачі в оптронах. Візуальна індикація стану апаратури типу «включено – виключено» і створення індикаторних табло є найбільш частим використанням світлодіодів, де вони прийшли на зміну неоновим лампам і лампам розжарювання. Основною причиною такої заміни є здатність роботи світлодіодів при малих струмах і напрузі, сумісних з амплітудами логічних рівнів напруги мікросхем цифрової техніки. Така заміна ламп дозволила підвищити надійність приладів відображення інформації за рахунок використання в них в якості елементної бази лише виробів, виконаних за напівпровідниковою технологією. Крім того, лампи розжарювання, споживаючи значні потужності, виділяють велику кількість тепла, що призводить до руйнування патронів і утримувачів, скорочує термін служби ізоляційних матеріалів. Напівпровідникові індикатори споживають менше потужності в порівнянні з лампами розжарювання, практично не виділяють тепла, надійніші і довговічніші.

Розглянемо схеми включення світловипромінюючих діодів. На рис. 2.5 приведені схеми включення з одним СВД.

Для схеми (рис. 2.5, а) значення струму $I_{с.д}$, що протікає через СВД, залежить від опору $R_н$, напруги живлення $U_п$ і падіння напруги на СВД:

$$I_{ñ.ä} = (U_ï - U_{ñ.ä})/R_í .$$

Для схеми на рис. 2.6, б, вихідна напруга $U_{вх} \approx U_{вих}$, а струм

$$I_{с.д} = I_{вх}(1+\beta)\frac{R_{вх}}{r_{вх}}\frac{R_э + (R_н + r_д)}{R_э},$$

де $R_{вх}$, $r_{вх}$ – вхідний опір каскаду та транзистора відповідно; $r_д$ – динамічний опір СВД; β – коефіцієнт підсилення струму.

Якщо вважати, що $R_{вх} \approx r_{вх}$, а $r_д \ll R_н$, що легко забезпечити при живленні великою напругою, то отримаємо

$$I_{с.д} = I_{вх}(1+\beta)\frac{(R_э + R_н)}{R_э}.$$

На рис. 2.5 наведено схеми включення світлодіода.

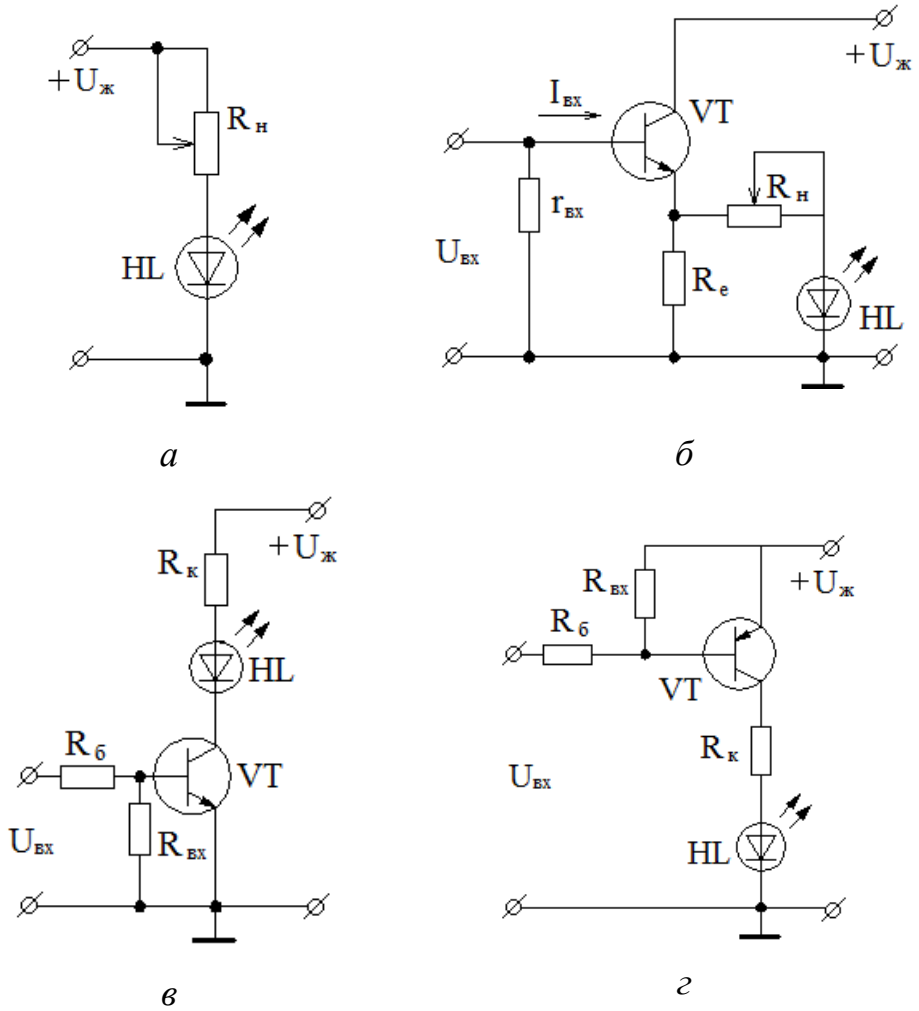


Рис. 2.5.

Для схеми на рис. 2.5, в, г, струм через СВД визначаэться виразом:

$$I_{с.д} = \frac{U_п - U_{с.д} - U_{КЕ\ відкр}}{R_н + r_д},$$

де $U_{КЕвідкр}$ – залишкова напруга на транзисторі у відкритому стані.

Якщо вважати, що $U_п \gg U_{с.д}$, $U_п \gg U_{КЕ}$ і $R_к \gg r_д$, то отримаємо:

$$I_{с.д} = U_п / R_н.$$

Струм бази транзистора

$$I_Б = (U_{вх} - U_{БЕ})/R_б .$$

Опір $R_б$ повинен забезпечувати струм насичення транзистора

$$I_Б = SI_К/\beta ,$$

де $S = I_Б/I_{Б\ гр}$ – коефіцієнт насичення; $I_{Б\ гр}$ – граничне значення струму бази.

Аналіз схем вмикання СВД показує, що схеми на рис. 2.5, в, г, мають вищу швидкодію і є генераторами струму, якщо не доводити їх до насичення. Схема на рис. 2.5, б, є генератором напруги, і струм через СВД визначається опором навантажувального резистора СВД.

Найбільшу швидкодію мають схеми на рис. 2.6, в, г. Їх відмінність полягає в тому, що в схемі на рис. 2.5, г, СВД підключений до загальної шини. Слід також відзначити, що значення струму через СВД визначається його внутрішнім опором. Для усунення цих нестабільностей необхідно підвищувати напругу живлення $U_п$, оскільки нестабільність, що вноситься транзистором і СВД, визначається виразом:

$$K_{іс} = \Delta U_н/U_п ,$$

де $\Delta U_н$ – нестабільність через зміни падіння напруги на транзисторі та СВД.

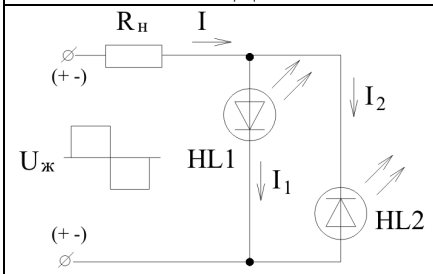
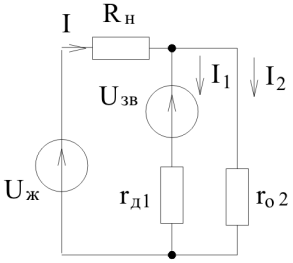
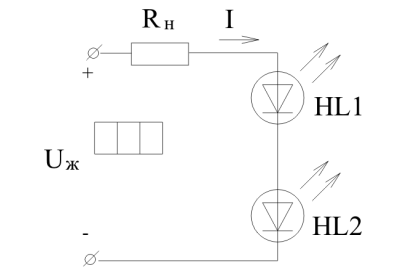
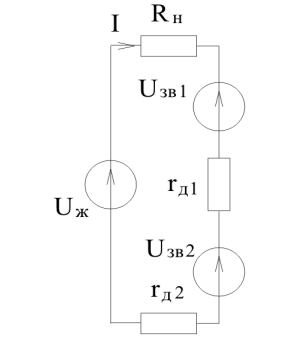
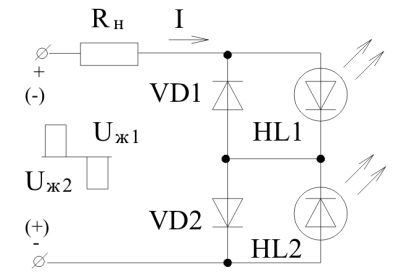
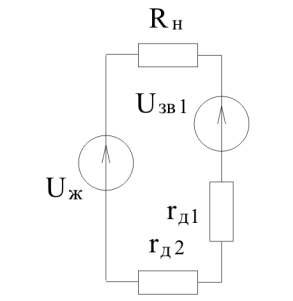
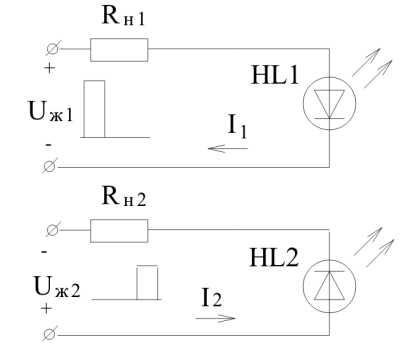
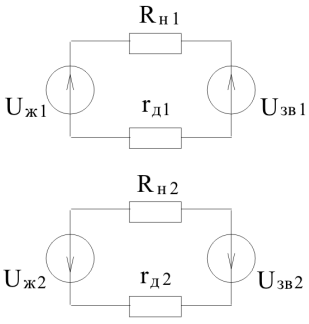
Таким чином, чим більше $U_п$, тим більшими мають бути опори $R_к$ і $R_н$, і менше позначатиметься вплив внутрішнього опору СВД.

При використанні СВД в двохвильових пристроях слід розрізняти чотири варіанти включення СВД опорного і вимірювального каналів: зустрічно-паралельне; послідовне; зустрічно-послідовне, роздільне (незалежне) включення.

У табл. 2.1 приведені варіанти вмикання СВД та їх схеми заміщення, складені з урахуванням наступних припущень. Для схем 1 і 2 вважаємо, що СВД U_1, U_2 мають ідентичні характеристики, і складаємо схему заміщення для одного з СВД. Розглянемо окремо кожен з варіантів вмикання.

При зустрічно-паралельному вмиканні СВД живлення здійснюється від генератора змінного струму. В один з напівперіодів включається перший СВД, в інший напівперіод – другий. Отже, якщо підібрані ідентичні СВД, то на точність вимірювання не впливає нестабільність живлячих імпульсів. Розрахункові співвідношення для цього варіанту включення СВД приведені в табл. 2.1. При цьому вважаємо, що СВД ідентичні і розрахунок виконано для одного з них.

Табл. 2.1.

Схема вмикання СВД	Схема заміщення	Розрахункові формули
		$I = \frac{U_{ж} - U_{зв} + I_2 r_{о2}}{r_{о2} + R_н} - \frac{U_{зв} + I_1 r_{д1}}{r_{о2} + r_{д1}},$ $R_н = \frac{U_{ж} - U_{зв} - r_{д1}(I_1 + I_2)}{I}$
		$I = \frac{U_{ж} - U_{зв1} - U_{зв2}}{R_н + r_{д1} + r_{д2}},$ $R_н = \frac{U_п - U_{зв1} - U_{зв2} - I(r_{д1} + r_{д2})}{I}$
		$I_1 = I_2,\ E_1 = E_2,$ $I_1 = \frac{U_{ж} - U_{зв1}}{R_н + r_{д1} + r_{д2}},$ $R_н = \frac{U_{ж} - U_{зв1} - I_1(r_{д1} + r_{д2})}{I}$
		$I_1 = \frac{U_{ж1} - U_{зв1}}{R_{н1} + r_{д1}}, I_2 = \frac{U_{ж2} - U_{зв2}}{R_{н2} + r_{д2}},$ $R_{н1} = \frac{U_{ж1} - U_{зв1} - I_1 r_{д1}}{I_1},$ $R_{н2} = \frac{U_{ж2} - U_{зв2} - I_2 r_{д2}}{I_2}$

Примітка: $U_ж$, $U_{ж1}$, $U_{ж2}$ – напруга джерел живлення; $R_{н1}$, $R_{н2}$ – опір навантажувальних резисторів; I_1, I_2 – струми через СВД; $r_д$ – динамічний опір СВД; $R_{зв1}$, $R_{зв2}$ – зворотні опори СВД; $U_{зв}$ – пряме падіння напруги на СВД.

Перевагою послідовного варіанту включення є простота схемної реалізації джерела живлення. Недоліком є необхідність застосування двох фотоприймачів, причому кожен з них повинен сприймати випромінювання лише одного СВД, тобто цей варіант не забезпечує часового розділення

потоків від кожного імпульсу випромінювача. Проте застосування комбінації фотодіод-фоторезистор дає можливість реалізувати операцію розділення сигналів від потоків обох СВД. Основна похибка послідовного включення СВД пов'язана з нестабільністю характеристик двох фотоприймачів, окремо чутливих до потоків першого і другого СВД. Ця схема може бути застосована при невисоких вимогах до пристроїв контролю (абсолютна похибка понад 3%).

Наступним варіантом включення СВД є зустрічно-послідовне включення. Для визначення струму, що протікає через СВД, зробимо припущення, що СВД мають ідентичні електричні параметри. Як і при першому варіанті включення, тут живлення здійснюється від генератора змінної напруги. Недоліком схеми є відсутність роздільного вирівнювання потоків випромінювання.

Останнім варіантом є роздільне незалежне включення СВД. В цьому випадку складають рівняння для обох ланцюгів з СВД окремо, і з них знаходять струми та опори навантаження.

Переваги останньої схеми включення – можливість незалежного живлення СВД, взаємного вирівнювання потоків випромінювання і часове розділення цих потоків.

Внаслідок низького диференціального опору світлодіодів їх не слід включати паралельно, оскільки незначна зміна $U_{\text{іп}}$ (напруга джерела живлення) і відмінність в диференціальному опорі може призвести до різкого зростання струму світлодіода з меншим $U_{\text{іп}}$ (пряме падіння напруги на світлодіоді), до різкого зростання його світловіддачі та розсіюваної потужності. Таким чином, при підключенні до джерела напруги $U_{\text{іп}}$ кожен світлодіод має бути захищений своїм резистором.

При необхідності підключення до одного джерела живлення світлодіодів різних кольорів (червоного, зеленого, жовтого), опори струмообмежувальних резисторів у зв'язку з відмінністю $U_{\text{пр}}$ розраховуються для світлодіодів кожного кольору окремо. Один з варіантів такого підключення представлений на рис. 2.6.

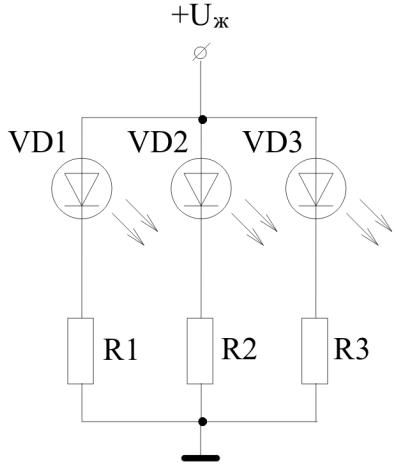


Рис. 2.6.

2.3.2. SMD-світлодіоди

Останніми роками все чіткіше прослідковується тенденція у бік мініатюризації РЕА. Світлодіоди для поверхневого монтажу надзвичайно потрібні виробникам компактної або портативної апаратури з батарейним живленням, де дуже висока щільність компоновки елементів і необхідне изьке споживання енергії.

Абревіатура SMD означає поверхневий монтаж деталей (surface montage details). Дана технологія (“3-in-1” – “3-в-1”) дозволяє виготовити в одному корпусі три світлодіоди (червоний, синій, зелений) і збільшити світловий потік світлодіода за рахунок наближення напівпровідника до теплопровідної поверхні. На рис. 2.7 представлений вигляд SMD-світлодіодів.

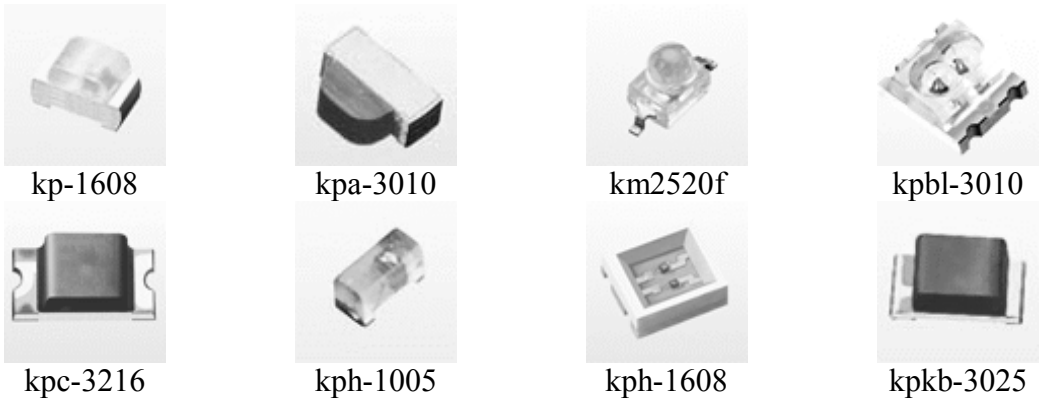


Рис. 2.7.

Світлодіоди виробляються синього, зеленого, червоного, жовтого, помаранчевого кольорів, а також їх різних відтінків. Існує також ряд двокольорових моделей. За силою світла і ефективностю світлодіоди розділяються на світлодіоди високої яскравості, високої яскравості з малим споживанням і стандартні. Діапазон яскравостей складає від 3,6 до 360 мКд залежно від моделі і кольору, а струм споживання від 2 до 20 мА. У табл. 2.2 приведені основні параметри деяких smd-світлодіодів.

Табл. 2.2.

Тип	Структура	Довжина хвилі λр, nm	Колір	Uпр ,В при Iпр=20 мА		Iv , мКд при Iпр=20 мА			Кут 2Θ 1/2 , град
				Мін	Макс	Мін	Тип	Макс	
0805B120C-B	InGaN	460-470	синій	3,0	3,6	150		250	120
0805G120C-B	InGaN	515-530	зелений	3,0	3,6	600		750	120
0805G120C-B	InGaN	500-510	синьо-зелений	3,0	3,6	400		600	120

0805W120C-B	InGaN	-	білий	3,0	3,6	600		750	120
0805R120C-B	AlGaInP	615-630	червоний	1,9	2,2	150		250	120
0805A120C-B	AlGaInP	600-610	помаранчовий	1,9	2,2	150		250	120
0805Y120C-B	AlGaInP	585-595	жовтий	1,9	2,2	150		250	120

Деякі компоненти мають матовий розсіювач. Більшість SMD-світлодіодів поставляються з прозорим розсіювачем. Розподіл випромінювання в півсфері таких світлодіодів близький до рівномірного розподілу (рис. 2.8). Найбільші моделі комплектуються рефлектором, завдяки якому різко підвищується інтенсивність випромінювання.

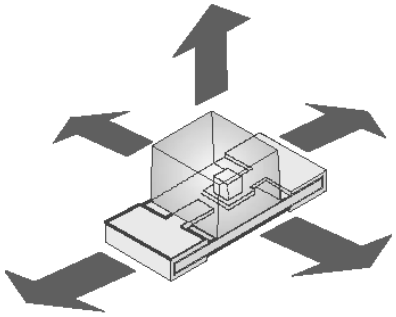


Рис. 2.8.

Також доступний ряд дуже популярних світлодіодів з бічним випромінюванням для підсвічування РКІ і зворотнього монтажу (зворотнього світіння).

2.3.3. Ультраяскраві світлодіоди

Область застосувань ультраяскравих світлодіодів може бути умовно розділена на дві широкі категорії, а саме з використанням прямого світла і освітлення. Пряме світло використовується для передачі інформації, наприклад в алфавітно-цифрових табло і повноколірних відеодисплеях, де світлодіоди формують пікселі дисплея. У сигнальних пристроях також використовується пряме світло. Як приклад, дорожні сигнали – світлофори і знаки, стоп-сигнали та індикатори повороту транспортних засобів. В освітленні світлодіод використовується щоб освітити поверхню, простір або об'єкт, замість того, щоб бути видимим безпосередньо.

Ультраяскраві світлодіоди (УЯС) випускає фірма Hewlett-Packard (HLMA-QL00, HMLP-Q106, HPWT-PL00 та ін.).

Основними характеристиками УЯС є:

– сила світла, яка в деяких моделях УЯС доходить до 4-9 тис. мКд (при $I_{пр}$ = 20 мА);

– пряме падіння напруги $U_{пр}$ (1,7 – 2,4 В);

– прямий максимальний струм $I_{макс}$ (до 70 мА);

– кут розсіювання світла (5-125 градусів).

На рис. 2.9 представлено зовнішній вигляд деяких ультраяскравих світлодіодів.



STAR LXHL-MB1C



STAR/C LXHL-MB1B



STAR/O LXHL-NB98



K2 LXK2-PH12-S00

Рис. 2.9.

В таблиці 2.3 наводяться основні параметри ультраяскравого світлодіода LXHL-MB1C.

Табл. 2.3.

Яскравість	16 Лм
Максимальний неперервний струм	350 мА
Пряма напруга	3,42 В
Довжина хвилі	470 нм
Тип	Star

Ультраяскраві світлодіоди мають переваги перед сигнальними лампами розжарювання (СЛР):

1) УЯС споживають набагато меншу потужність при однаковому світловому потоці, порівняно до СЛР (приблизно в 5 разів);

2) УЯС виділяють набагато менше тепла ніж СЛН, тому їх робоча температура набагато нижча, ніж в СЛР;

3) УЯС дозволяють отримати принаймні чотири вихідні кольори (червоний, янтарний, зелений, блакитний), а при комбінаціях УЯС кількість кольорів може бути достатньо великою. У СЛР доводиться ставити спеціальні кольорові ковпаки (світлофільтри);

4) надійність УЯС складає порядка 10 млн. годин безперервної роботи, в порівнянні з 5000 – 20000 годин в СЛР;

5) УЯС здатні витримувати без пошкоджень інтенсивні механічні удари і вібрацію (на відміну від СЛР);

6) УЯС витримують більше 1000 циклів нагрівання/охолодження від –40 ^{0}С до +100 ^{0}С без пошкоджень (СЛР – не більше 3-5 циклів);

7) розміри УЯС дозволяють створити пристрої індикації набагато менших розмірів, ніж пристрої індикації на СЛН;

8) УЯС допускають поступове регулювання яскравості світіння за допомогою потенціометра; більше того, одним потенціометром можна

регулювати яскравість світіння одночасно декількох УЯС; регулювання яскравості може також здійснюватися за допомогою широтно-імпульсної модуляції. В цьому випадку кількість одночасно регульованих УЯС взагалі не має значення;

9) оскільки УЯС працюють при менших струмах (20-70 мА), ніж СЛР (0,3-1 А), то пристрої управління і комутації УЯС простіші і дешевші, ніж такі ж пристрої для СЛР;

10) у момент ввімкнення СЛН мають набагато більший струм, ніж робочий. УЯС позбавлені цього недоліку.

2.3.4. Лазерні світлодіоди

Випромінювання лазера має вимушену природу. Воно утворюється коли падаючий фотон викликає перехід електронів з другого енергетичного рівня, на якому заздалегідь була сформована інверсна заселеність, на перший. При цьому виділяються фотони випромінювання, направленого в один бік з однаковими довжинами хвиль і поляризацією, тобто утворюється когерентне випромінювання за наявності відбиваючих торців або інших оптичних резонаторів. Вимушене випромінювання лазерних діодів дозволяє використовувати модуляцію за параметрами світлової хвилі, наприклад частотну. Крім того, вони характеризуються максимальною для напівпровідникових випромінювачів потужністю до декількох сотень міліват, мінімальною шириною спектру і дуже вузькою спрямованістю. На рис. 2.10 представлений зовнішній вигляд лазерних світлодіодів.

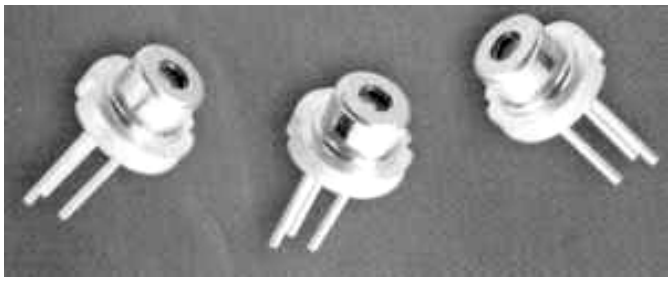
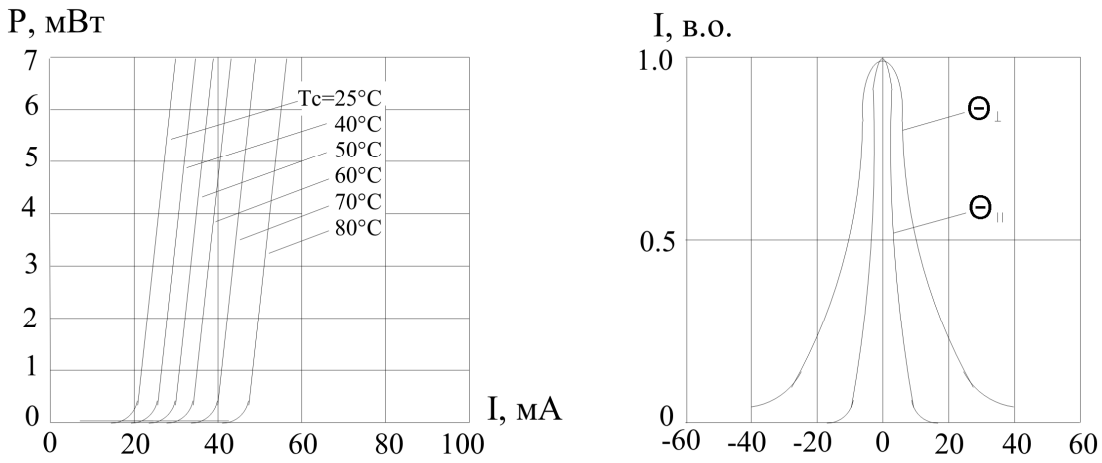


Рис. 2.10.

На рис. 2.11 представлено графіки потужності випрмінювання (а) і діаграма спрямованості лазерного світлодіода (б).



а *б*

Рис. 2.11.

Принциповою відмінністю лазерного діода від звичайного світлодіода є наявність в першому вбудованого оптичного резонатора, що дозволяє за умови перевищення струмом інжекції деякого порогового значення отримати режим індукованого випромінювання, яке характеризується високою мірою когерентності. Найчастіше як оптичний резонатор використовують: плоский резонатор Фабрі-Перо і його модифікації, включаючи складові і зовнішні резонатори, резонатори з розподіленим зворотним зв'язком (РЗЗ-резонатор) і з розподіленим бреговським відбивачем (РБВ-резонатор). Плоский резонатор утворюється зазвичай паралельно зрізаними торцями напівпровідника, а РЗЗ- і РБВ-резонатори – шляхом періодичної просторової модуляції параметрів структури, що впливають на умови поширення випромінювання.

Для лазерних діодів характерна температурна залежність порогового струму і диференціальної квантової ефективності. Із зростанням температури пороговий струм збільшується, а диференційна квантова ефективність зменшується. Зміна температури призводить також до зміни довжини хвилі випромінювання.

Оскільки лазерні діоди відрізняються складнішою конструкцією і більшими електричними навантаженнями, то вони поступаються в надійності, зручності експлуатації і вартості. Це визначає їх переважне застосування для здійснення передачі інформації на далекі відстані в магістральних лініях.

2.3.5. Багатокольорові світлодіоди

Світлодіоди є одними з небагатьох джерел світла, які дозволяють реалізувати керовану зміну кольору світіння. Існує декілька шляхів створення світлодіодів з керованим кольором світіння: двохперехідний однокристальний діод Ga з незалежним включенням переходів; одноперехідний двосмуговий однокристальний діод Ga; двохкристальний

біполярний діод з паралельним з'єднанням кристалів; двохкристальний діод з незалежним включенням кристалів; двохперехідний однокристальний діод, один з p-n-переходів якого випромінює червоне світіння, а інший – інфрачервоне випромінювання, яке за допомогою антистоксового люмінофора перетворюється на зелене світіння. Найбільший інтерес на теперішній час представляє двохперехідний однокристальний діод Ga. На рис. 2.12 представлені зовнішній вигляд і структура деяких багатокольорових світлодіодів.

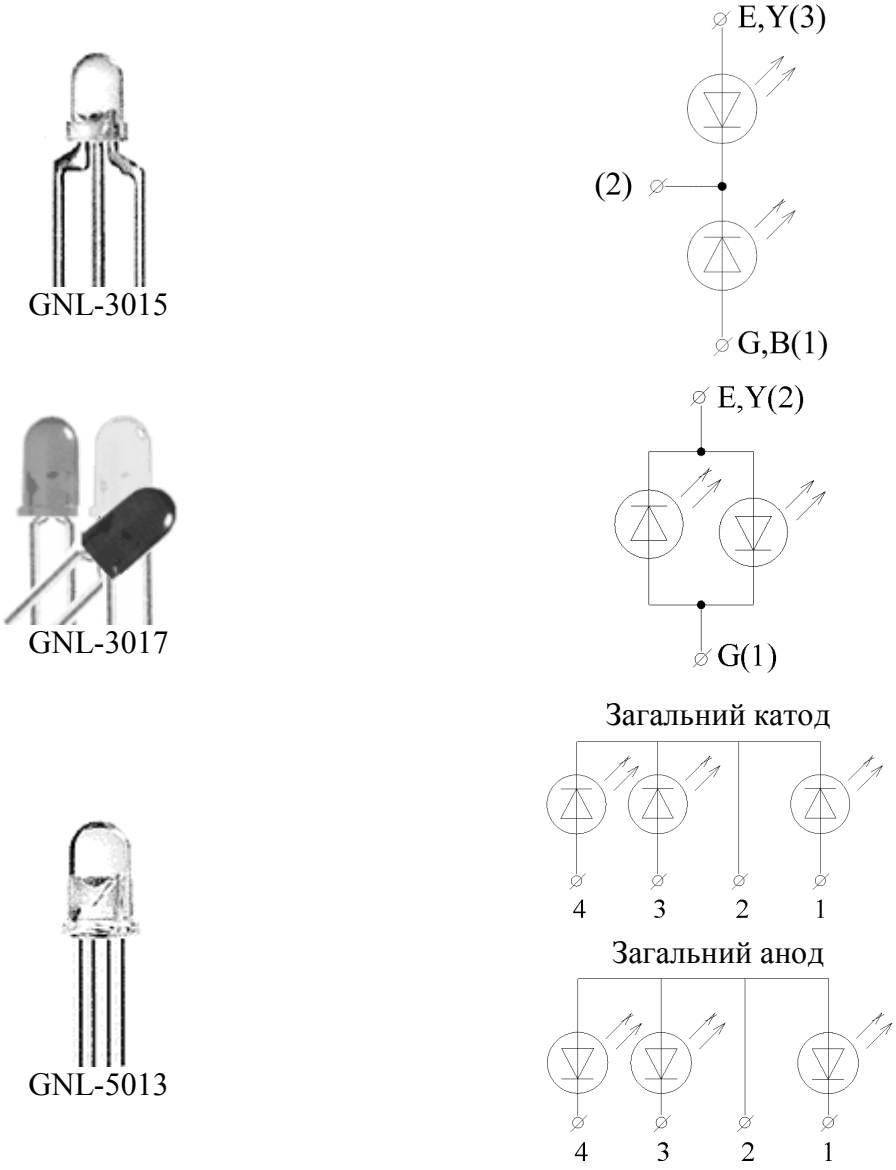


Рис. 2.12.

Як видно на рисунку, з'єднання світловипромінюючих діодів в одному корпусі може бути різним: із загальним катодом (GNL-3015), зустрічно-паралельне (GNL-3017), із загальним анодом (GNL-5013). У табл. 2.4 представлені параметри багатоколірних світлодіодів.

Табл. 2.4.

Максимальний постійний прямий струм	20 мА
Постійна зворотня напруга	5 В
Постійний зворотній струм (Vr–5B)	10 мкА
Максимальний прямий струм, tw=1мс, Q=1/20	150 мА
Діапазон робочих температур	-40...+80°C
Температура збереження	-40...+100°C
Умови пайки (5 сек, 1,6 мм від корпусу)	300±5°C

2.3.6. RGB-світлодіоди

Існує дві технології отримання білого світла за допомогою світлодіодів. Один підхід – змонтувати червоний, зелений і синій кристали в одному корпусі як можна ближче один до одного, і змішати їх світлові потоки в правильній пропорції для здобуття білого кольору. Проблемою такого підходу, ігноруючи технічні проблеми установки коректних рівнів світлових виходів, є вартість 3 кристалів. Проте, трикольорові світлодіоди популярні для застосівання в якості підсвічування LCD дисплеїв, оскільки користувач може встановити підсвічування будь-якого відтінку. На рис. 2.13 представлений зовнішній вигляд RGB-світлодіодів.



LF-5WAEMBGMBW



SxTR-300



GNL-5013

Рис. 2.13.

Як і для багатоколірних світлодіодів, в RGB-світлодіодах електричне з'єднання всередині одного корпусу може бути різним. У RGB-світлодіода LF-5WAEMBGMBW кожен з кристалів має окремі виводи анода і катода, які всередині корпусу не сполучені. У світлодіода GNL-5013 кристали мають загальний катод. Параметри RGB-світлодіодів аналогічні параметрам багатоколірних світлодіодів.

На рис. 2.14 представлена типова діаграма спрямованості RGB-світлодіода.

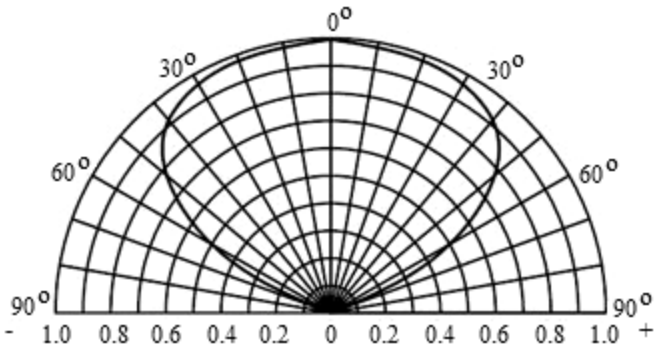


Рис. 2.14.

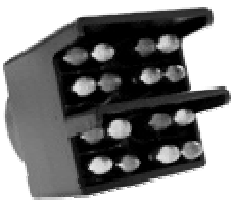
Компанія Kingbright випустила RGB серію високоефективних світлодіодів Chameleon з повним колірним рядом. Світлодіоди даної серії містять в одному корпусі три мініатюрні надяскраві кристали (червоний, синій і зелений) для відтворення різних кольорів видимого спектру, включаючи біле світло, при видимому тілесному куті 100 градусів. Чудові показники тепловіддачі, досягнуті за рахунок зміни провідної основи до 5,0×6,0 мм, дозволяють підвищити робочий струм кожного світлодіода до 50 мА. Можливе досягнення повнокольорового спектру за допомогою використання різного управляючого струму в кожному окремому кристалі світлодіода.

2.3.7. Світлодіодні кластери

Світлодіодні кластери представляють собою зв'язку світлодіодів в одному корпусі (рис. 2.15). Світлодіодні кластери бувають повноколірні (BL0104-21-35, BL0101-15-33, BL0102-14-34), що містять світлодіоди трьох основних кольорів (червоний, зелений, синій), а також двох- та одноколірні (BL0102-14-34, BL0307-50-44).



GNCL-R15A



GNCL-S15A



FYCL-S28A



FYCL-S15A

Рис. 2.15.

Внутрішнє з’єднання світлодіодів буває загальне (послідовно-паралельне, із загальним анодом) для одноколірних кластерів і роздільне для повноколірних світлодіодних кластерів.

На рис. 2.16 представлений зовнішній вигляд повноколірного світлодіодного кластера і схема розташування світлодіодів. Для управління кластером є роз'єм, розташований з тильної сторони.

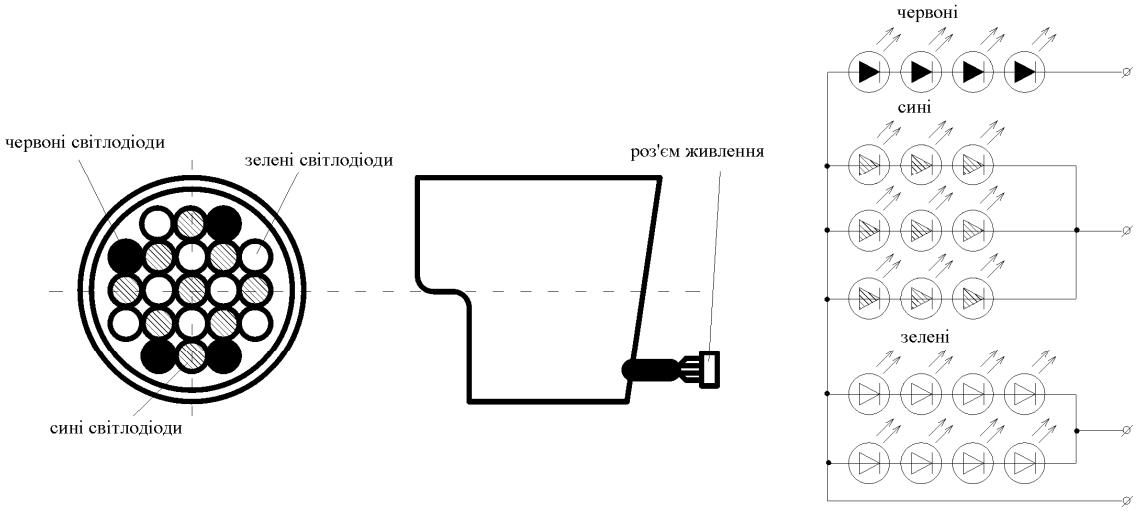


Рис. 2.16.

Геометричне розташування світлодіодів в кластері може бути різним. На рис. 2.17 представлені конструктивні модифікації одноколірних, багатоколірних і RGB-кластерів.

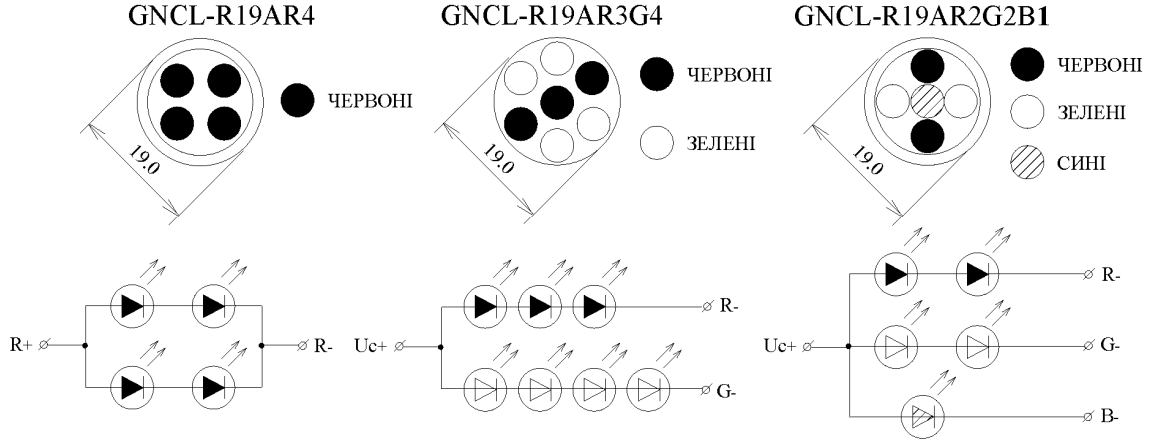


Рис. 2.17.

Існують збірки зі світлодіодних кластерів, які використовуються для побудови великих екранів. Конструктивно збірка зі світлодіодних кластерів є конструкцією, в якій розташування окремих елементів (кластерів) є геометрично симетричним. Це дозволяє формувати цілісне зображення, оскільки з'єднання декількох збірок виходить безшовним. На рис. 2.18 показана структура такої збірки.
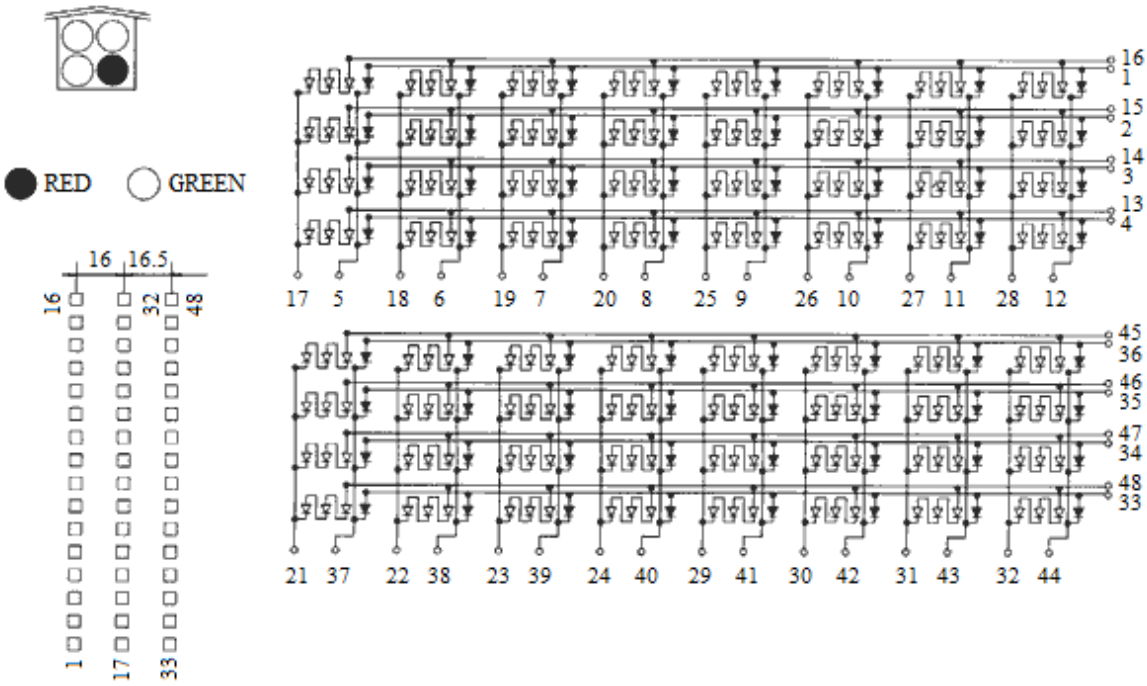


Рис. 2.18.

На рис. 2.19 представлено зовнішній вигляд кластерної збірки.

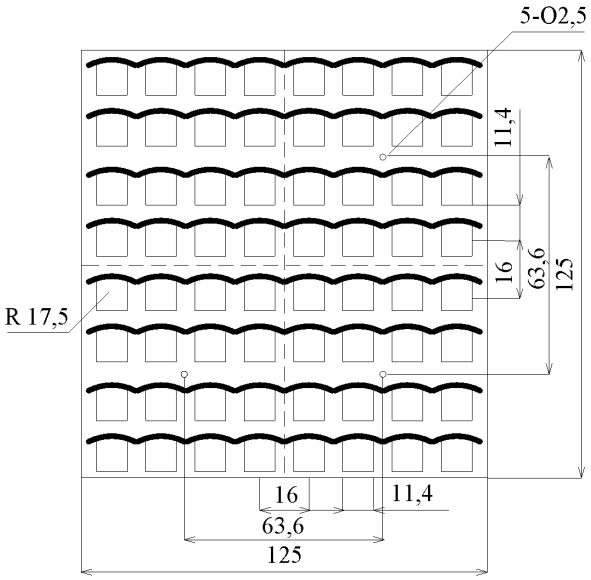


Рис. 2.19.

2.4. Режими живлення світлодіодів

Жівлення СВД здійснюється в різних режимах: постійним струмом, в імпульсному режимі і функціональному режимі (наприклад, живлення струмом, сформованим по експоненті). У табл. 2.5 приведені ці режими, їх часові діаграми і математичне представлення.

Режим живлення постійним струмом є найбільш простим, оскільки для його здійснення не потрібні спеціальні пристрої (генератори). СВД

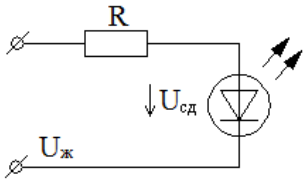
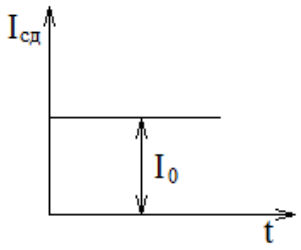
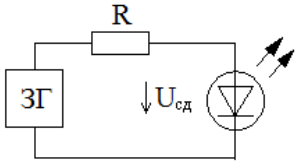
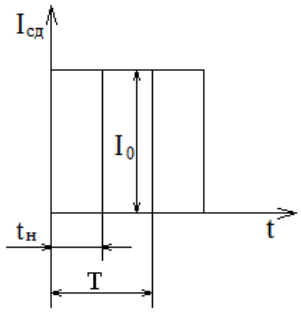
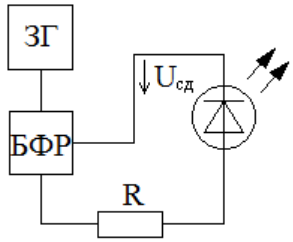
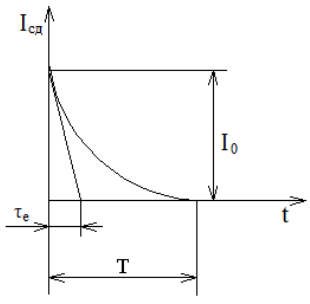
підключаються до джерела живлення через струмозадаючий регістр. Опір цього резистора визначається з виразу:

$$R = (U_{ж} - U_{с.д})/I_{н},$$

де $U_Ж$ – напруга джерела живлення; $U_{С.Д}$ – падіння напруги на СВД, значення якої від 1,0 до 2,2 В; $I_Н$ – номінальний струм СВД.

Цей режим живлення найбільш прийнятний для одноканальних оптоелектронних вимірювальних перетворювачів з модуляцією сигналу в оптичному каналі (наприклад, для підрахунку штучної продукції, виміри щільності тканини і т. п.).

Табл. 2.5.

Режим живлення	Схема включення	Часові діаграми	Математичне представлення сигналу
Постійни струм			$\Phi(t) = KI_0 = \Phi_0$
Імпульсний струм			$\Phi(t) = \begin{cases} 0 \ при\ t < 0, \\ \Phi_0 \ при\ 0 \le t \le t_u \\ 0 \ при\ t \ge t_u \end{cases}$
Експонен-ціальний			$\Phi(t) = \Phi_0 e^{-(t/\tau_э)}$

Примітка: ЗГ – задаючий генератор, БФР – блок функціональної розгортки.

Імпульсний режим живлення має ряд переваг в порівнянні з режимом живлення постійним струмом. Якщо СВД застосовуються в двохвильових пристроях, то цей режим забезпечує часове розділення потоків і, отже, застосовується один фотоприймач із спектральною характеристикою, чутливою до обох потоків. Спрощуються процес посилення і обробка сигналу в порівнянні з першим режимом. Зменшується енергоспоживання, що вкрай необхідне при розробці приладів з автономним живленням. Потужність розсіювання на СВД можна визначити з урахуванням лінійної апроксимації вольт-амперної характеристики за наступною формулою:

$$W_{рас} = I_{с.д} U_{с.д} + I^2_{с.д} r_д ,$$

де $I_{с.д}$ – струм через СВД; $r_д$ – динамічний опір СВД; $U_{сд}$ – напруга відсічки.

Ефективне значення струму:

$$I_{еф} = I_i \sqrt{Q} ,$$

де I_i – імпульсний струм; Q – скважність.

Звідси максимально припустиме значення струму:

$$I_{i\,max} = I_i \sqrt{Q} = I_i \sqrt{T/t_e} ,$$

де t_e – тривалість імпульса; T – тривалість паузи.

Отже, для здобуття необхідної потужності випромінювання необхідно зменшувати тривалість імпульа і збільшувати паузи між імпульсами. Проте слід зазначити, що тривалість імпульсів струму обмежена швидкодією фотоприймача. Якщо фотопровідність (фотострум) встигає встановитися за (3…4) $\tau_ф$, то

$$t_i \geq (3…4)\ \tau_ф,$$

де $\tau_ф$ – постійна часу фотоприймача.

Підбираючи необхідну скважність, можна забезпечувати досить високу потужність випромінювання, на один-два порядки перевищуючу номінальну потужність випромінювання СВД.

Третій режим живлення – функціональний. Його перевагою є інтенсифікація виконуваних математичних операцій, і в разі застосування в пристроях контролю забезпечується лінеаризація передавальної характеристики пристрою з одночасним спрощенням всього пристрою і підвищенням точності.

Збільшення потужності випромінювання СВД при експоненціальній розгортці можна досягти зменшенням сталої часу, тривалості експоненти і

збільшенням початкового максимального значення струму СВД. Проте зменшення сталої часу призводить до зниження чутливості і збільшення похибки виміру. Подальше збільшення потужності випромінювання можна досягти, застосувавши режим дискретного функціонального живлення.

2.5. Пристрої зі світлодіодами

Інфрачервоні світловипромінюючі діоди (СВД) можна використовувати в зовнішніх пристроях обчислювальної техніки, в системах перетворення кут-код, в комутаційних пристроях та ін. (рис. 2.20 – рис. 2.28). На рис. 2.20 представлена схема індикатора заданої температури, на рис. 2.21 – схема індикатора сигналу модуляції, на рис. 2.22 – схема включення світлодіода в імпульсному режимі, на рис. 2.23 – схема індикатора стану логічного елементу, на рис. 2.24 – схема індикатора налаштування на радіостанцію.

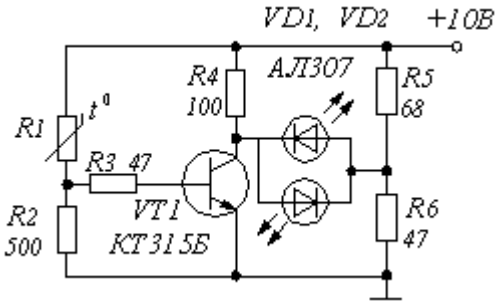


Рис. 2.20.

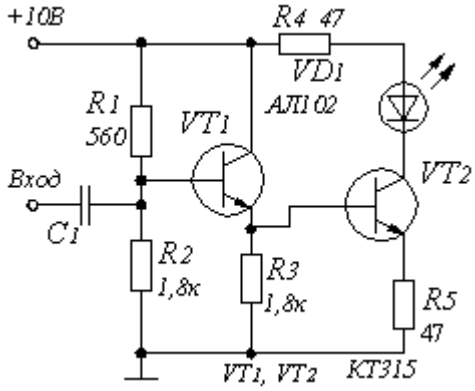


Рис. 2.21.

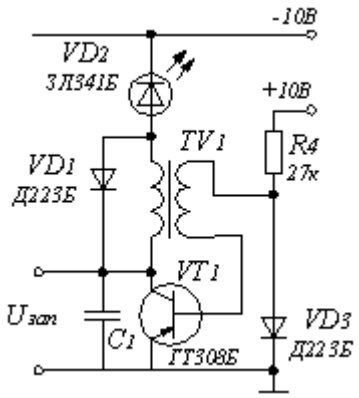


Рис. 2.22.

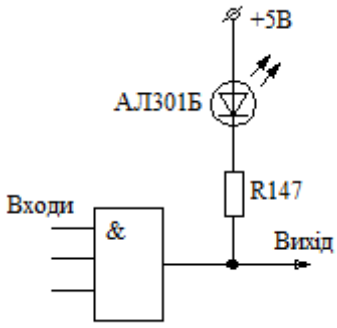


Рис. 2.23.

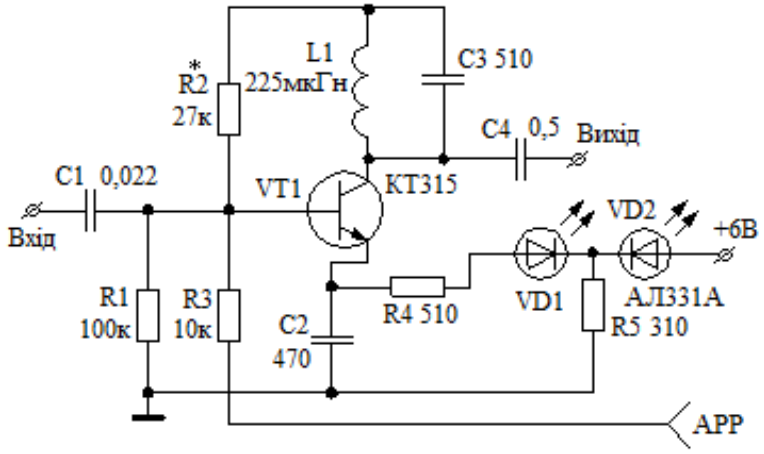


Рис.2.24.

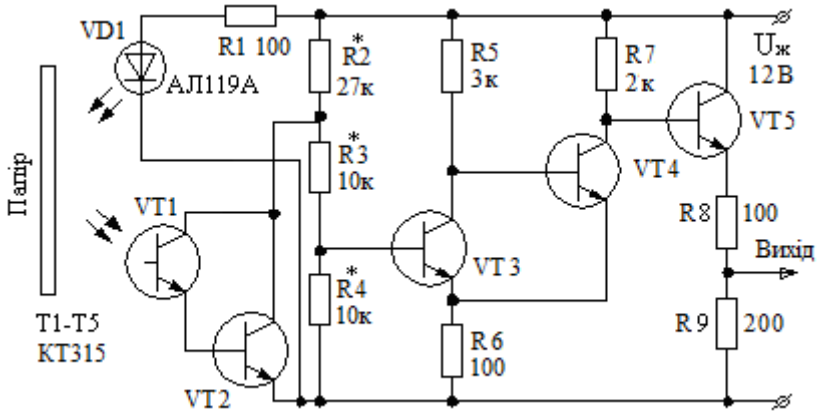


Рис. 2.25.

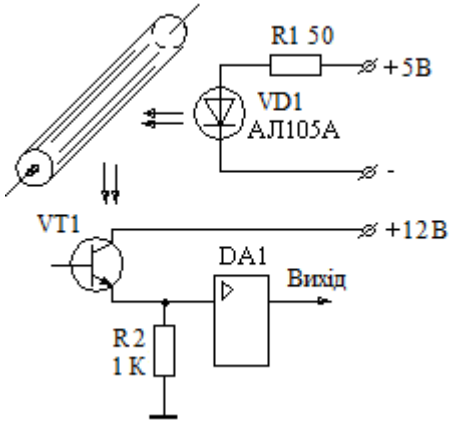


Рис. 2.26.

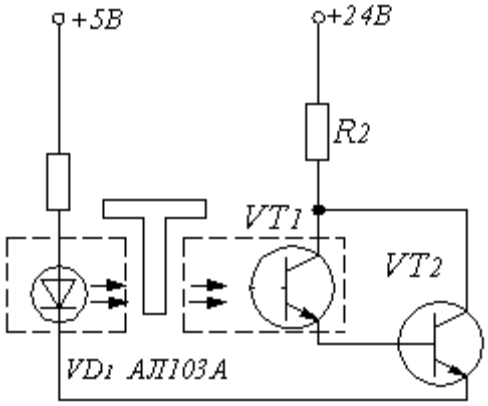


Рис. 2.27.

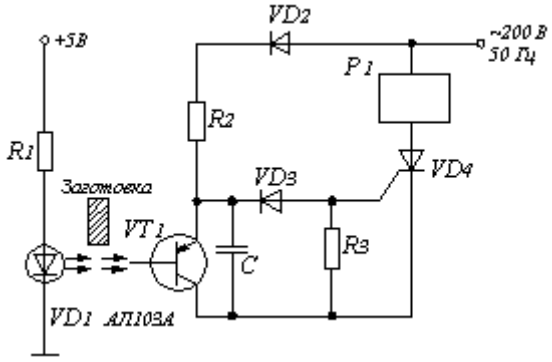


Рис. 2.28.

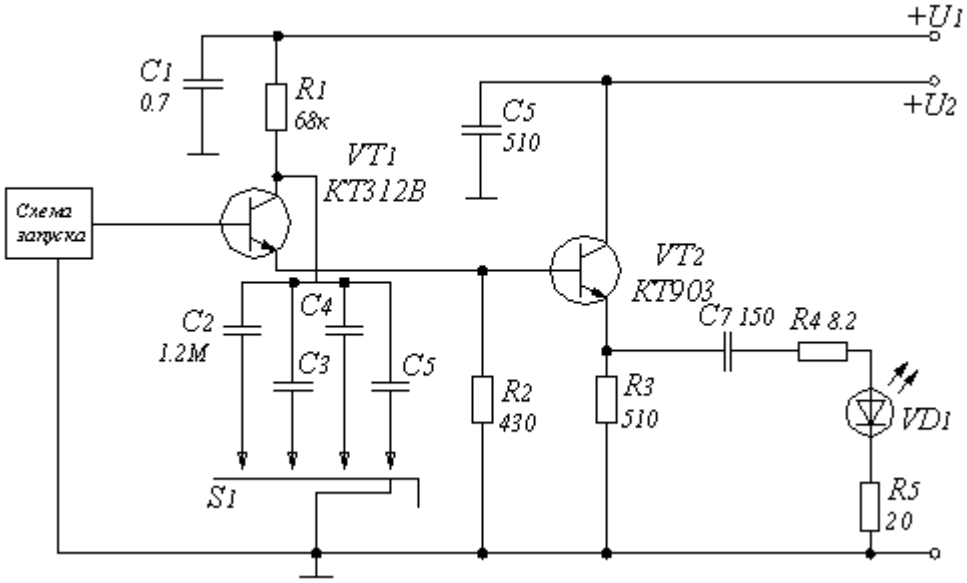


Рис.2.29.

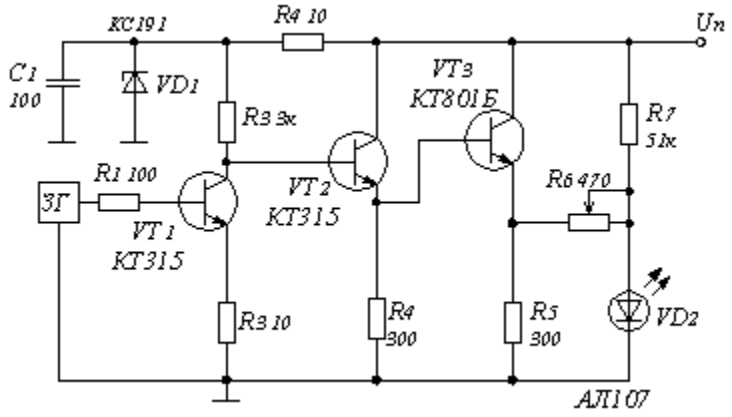


Рис. 2.30.

На рис. 2.25 представлена схема пристрою виявлення краю листа, на рис. 2.26 – схема оптоелектронного перетворювача кут-код, на рис. 2.27 – схема оптоелектронного клавішного перемикача конвейера, на рис. 2.28 – схема аварийної зупинки, на рис. 2.29 – схема джерела імпульсного живлення, на рис. 2.30 – схема імітатора лазерного випромінювання.

Більшість пристроїв використовує імпульсний режим живлення СВД, оскільки при цьому можна забезпечити необхідну потужність випромінювання, спростити процес підсилення і обробки сигналу. Отже, для живлення СВД необхідні генератори імпульсів, в яких забезпечуються необхідна стабільність амплітуди і частота повторення вихідних імпульсів. При невисоких вимогах джерело імпульсного живлення може бути реалізоване за схемою на рис. 2.29.

При використанні СВД в імітаторах лазерного випромінювання можна використовувати схему, приведену на рис. 2.30 [7]. Схема запуску може бути виконана у вигляді блокінг-генератора. Амплітуда імпульсу струму може встановлюватися від 2А до 10А зміною опору резистора R_6. Тривалість імпульсу може змінюватися в межах 50...500 нс перемиканням зарядних конденсаторів C_2-C_5 перемикачем S_1.

При високих вимогах джерела імпульсного живлення можуть бути виконані на основі компенсованих транзисторних ключів (рис. 2.31) [2]. Такі джерела мають високу стабільність вихідної імпульсної напруги ($5\cdot10^{-5}$). Керуючий сигнал через вхідний пристрій (VT_5, VT_6) поступає на бази транзисторів $VT_1 - VT_4$, які при цьому відкриваються. З колектора VT_4 на базу VT_1 поступає замикаюча напруга. Таке управління дозволяє усунути викиди на фронті вихідного імпульсу. Струм, що протікає через СВД, визначається ємністю конденсатора C. Його включення забезпечує малу залежність світлового потоку від внутрішнього опору СВД. Нестабільність комутованої напруги визначається в основному вибором стабілітронів і умовами їх роботи.

За допомогою розглянутих випромінювачів можуть проводитися дослідження імпульсних і перехідних характеристик фотоприймачів при різних номіналах елементів схеми.

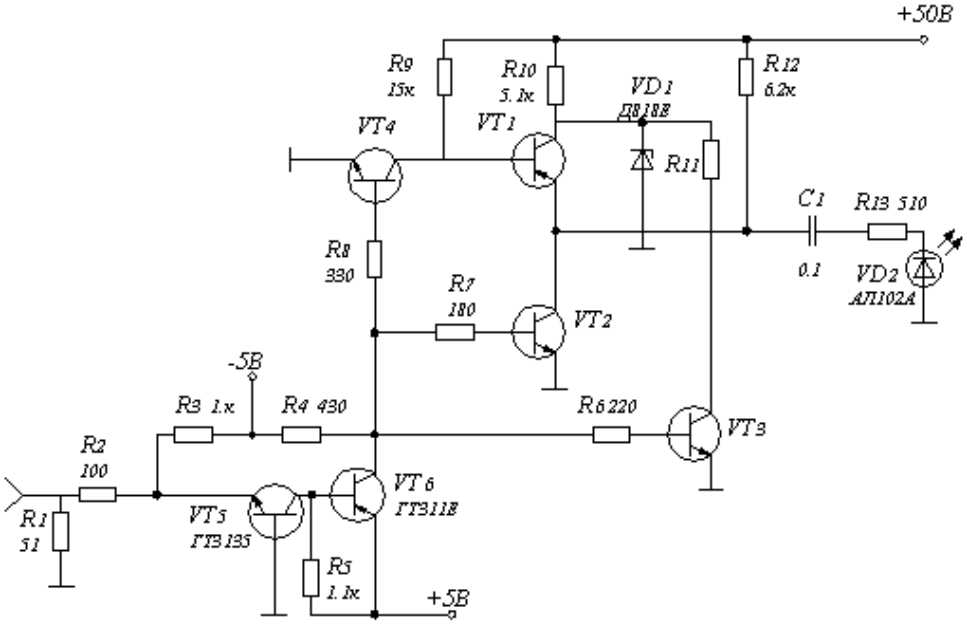


Рис. 2.31.

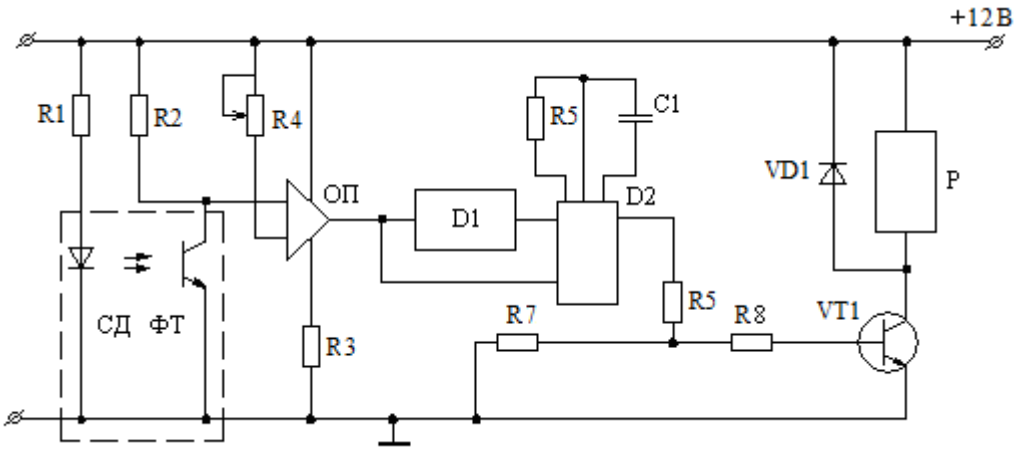


Рис. 2.32.

При підвищених вимогах до лінійності випромінювання можна використовувати параметричну лінеаризацію вихідної характеристики СВД [4].

Оптопару, СВД – фотоприймач, можна використовувати для контролю рівня. Прилад контролю рівня рідини містить дві пари елементів СВД – фототранзистор. Одна пара використовується для контролю верхнього рівня, інша – нижнього [7]. Електрична схема одного каналу вимірювача рівня приведена на рис. 2.32. Принцип дії приладу заснований на тому, що при зменшенні рівня рідини на фототранзистор *ФТ* падає випромінювання від СВД. Спрацьовує реле *Р* і включає відповідний блок системи. Так автоматично регулюється рівень рідини.

2.6. Багатосегментні індикатори

Найбільш поширеним виглядом напівпровідникових індикаторів (НПІ) слід вважати цифрові індикатори (ЦІ), оскільки переважна кількість інформації відтворюється в цифровому вигляді (годинник, калькулятори, вольтметри, термометри і т.д.). Конфігурація інформаційних полів цифрових індикаторів приведена на рис. 2.33.

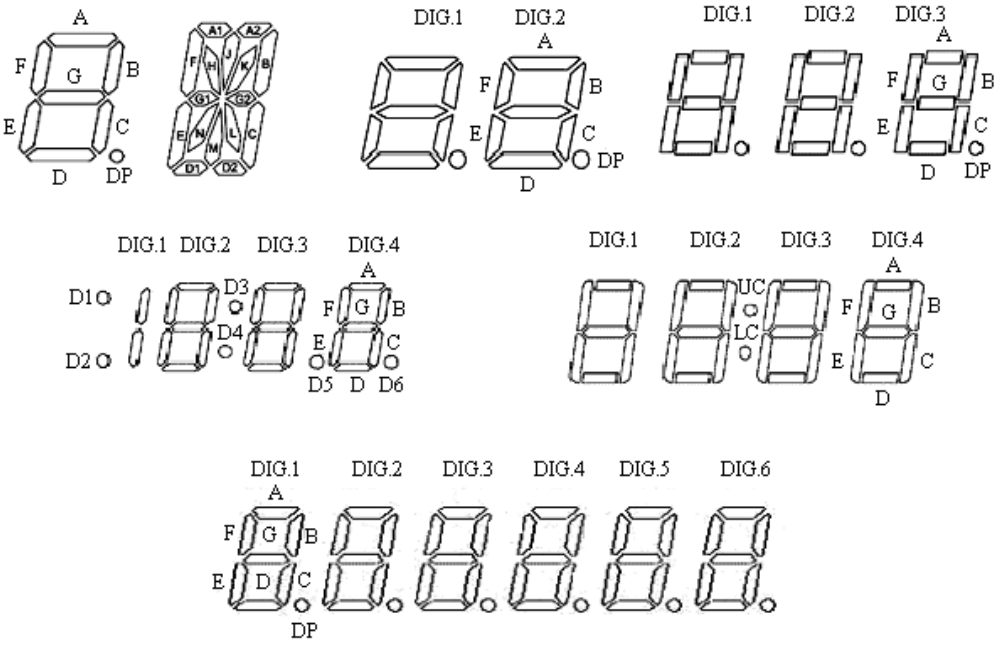
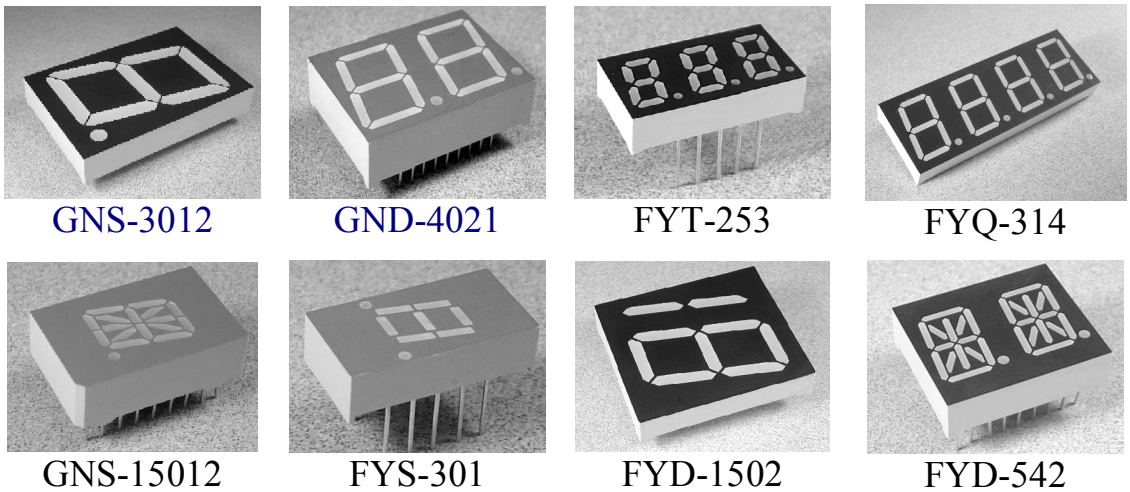


Рис. 2.33.

Всі цифрові індикатори за своїм конструктивним виконанням можна розділити на наступні групи: безкорпусні монолітні, монолітні в полімерній герметизації, гібридні з різними світлопровідниками, монолітні в склокерамічному корпусі.

Безкорпусні ЦІ – порівняно малочисельна група індикаторів. Конструктивно вони виконані на монолітному кристалі з випромінюючими елементами і контактними зонами для приєднання виводів. Прикладом такого ЦІ є індикатор для наручного годинника.

Найбільш поширеним типом ЦІ є прилади гібридної конструкції з використанням принципу розсіювання світла. Основу конструкції складає керамічний або текстолітовий тримач, на якому кріпляться за допомогою струмопровідного клею випромінюючі кристали. На рис. 2.34 представлені різновиди багатосегментних індикаторів.



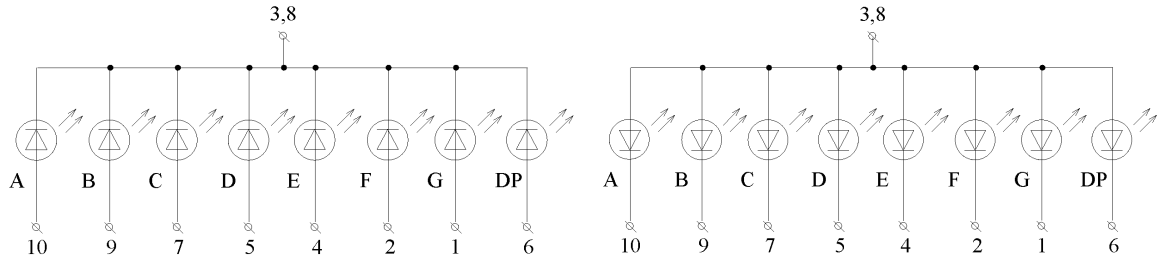
GNS-3012 GND-4021 FYT-253 FYQ-314

GNS-15012 FYS-301 FYD-1502 FYD-542

Рис. 2.34.

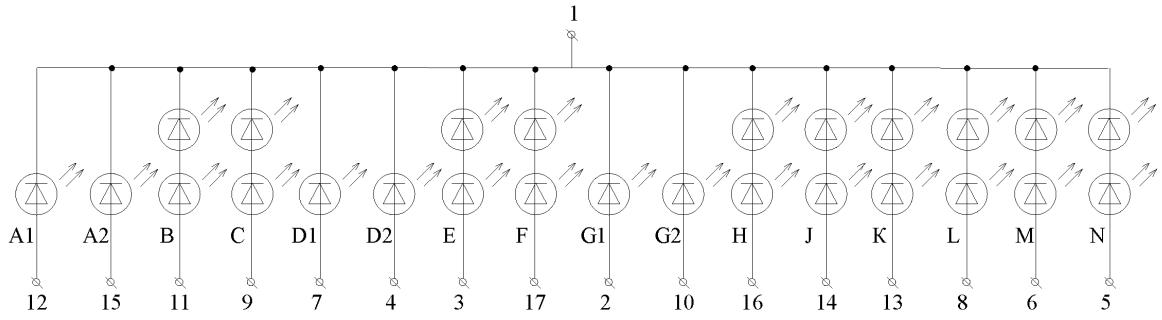
Буквено-цифрові індикатори (БЦІ) дозволяють відображати арабські і римські цифри, букви російського, латинського і грецького алфавітів, і ряд інших знаків і символів. На робочому полі може одночасно відображатися одне або декілька знакомісць. За кількістю елементів та їх взаємному розташуванню в межах одного поля розрізняють семисегментні, дев'ятисегментні, п'ятисегментні, 19-сегментні та інші. Аналіз великої кількості форматів шрифтів показав, що при зображенні цифр, яке оцінюється за п’ятибальною системою, значну перевагу мають індикатори девяти-, восьми- і семи сегментні.

З точки зору схемних рішень по управлінню індикаторами всі НПІ можуть бути представлені двома групами. До першої групи відносяться напівпровідникові індикатори в основному гібридної конструкції, що знайшли застосування в промислових приладах. Такі індикатори, як правило, мають великі габаритні розміри. Кожен з сегментів цього типу індикаторів має окремий світлодіод, а кожен зі світлодіодів – свій управляючий вхід. На рис. 2.35. показані схеми однорозрядних семисегментних світлодіодних цифрових індикаторів: а – GNS-3911 Ax (спільний катод); б – GNS-3911 Bx (спільний анод). На рис. 2.36. показані схеми однорозрядних 16-сегментних світлодіодних цифрових індикаторів: а – GNS-15012 Ax (спільний катод); б – GNS-15012 Bx (спільний анод).

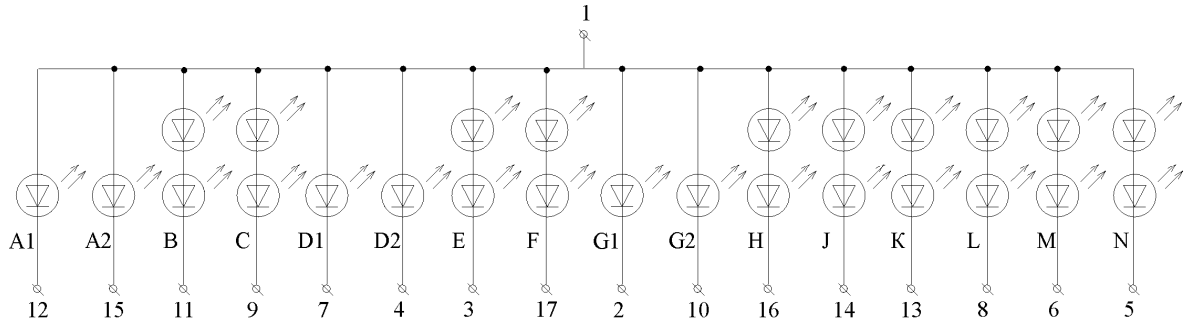


а *б*

Рис. 2.35.



а



б

Рис. 2.36.

Для відображення цифр десяткової системи счислення на індикаторі необхідно виконати перетворення двійкового коду в прямий код семисегментних індикаторів. Відповідно до розташування сегментів, створюємо таблицю істинності перетворювача (табл. 2.6), приписуючи значення логічної “1” тому світлодіоду, який засвічується при відображенні цифри, і логічного “0” – в протилежному випадку.

Табл. 2.6.

N	x_3	x_2	x_1	x_0	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

Використовуючи табл. 2.6, можемо записати логічні функції роботи кожного світлодіода:

$$a = \vee\, 0, 2, 3, 5, 6, 7, 8, 9 = \wedge 1, 4\,;$$

$$b = \vee\, 0, 1, 2, 3, 4, 7, 8, 9 = \wedge 5, 6\,;$$

$$c = \vee\, 0, 1, 3, 4, 5, 6, 7, 8, 9 = \wedge 2\,;$$

$$d = \vee\, 0, 2, 3, 5, 6, 8, 9 = \wedge 1, 4, 7\,;$$

$$e = \vee\, 0, 2, 6, 8 = \wedge 1, 3, 4, 5, 7, 9\,;$$

$$f = \vee\, 0, 4, 5, 6, 8, 9 = \wedge 1, 2, 3, 7\,;$$

$$g = \vee\, 2, 3, 4, 5, 6, 8, 9 = \wedge 0, 1, 7\,.$$

На рис. 2.35 та 2.36 сегменти індикаторів позначені заглавними літерами (A-H), а у формулах, що наведені вище, використані строкові літери. Це пов'язано з різницею стандартів, що використовуються при описанні подібних систем. У вітчизняній літератури використовуються маленькі літери. Разом з тим, більшість фірм, що виробляють індикатори, є закордонними, і в технічній документації на індикатори цих фірм використовуються заглавні літери.

Таблиця 2.6 відповідає логіці роботи дешифратора К564ІД5 (CD4056BE, HCC4056BD, TC4056BF). Умовне графічне позначення дешифратора К561ІД5 приведене на рис. 2.37.

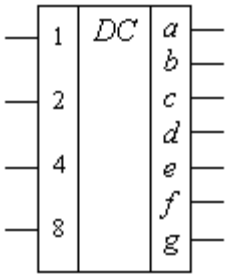
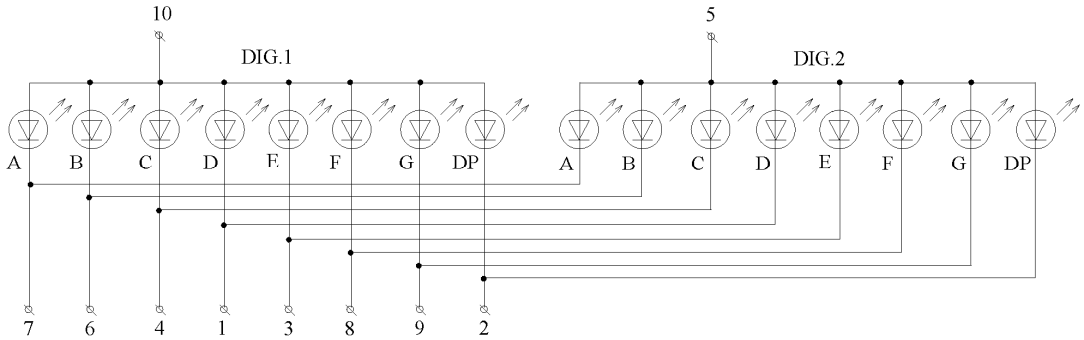


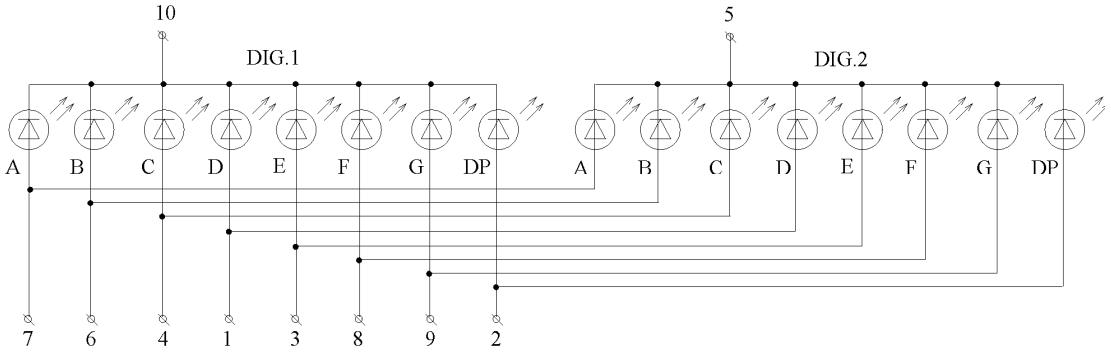
Рис. 2.37.

До другої групи цифрових індикаторів відносяться індикатори монолітної конструкції. Вони виготовляються методом дифузії напівпровідникових переходів для декількох індикаторів на одну підложку, тобто в одному корпусі такого приладу розміщено декілька індикаторів (чотири – шість і більше).

Як правило, такі індикатори мають загальні управляючі входи для однойменних сегментів декількох цифр (рис. 2.38, а – GND-4021Ax; б – GND-4021Bx). Монолітні індикатори застосовуються в основному в індикаторах приладів індивідуального користування, зокрема в наручних годинниках, калькуляторах, переносних приладах.

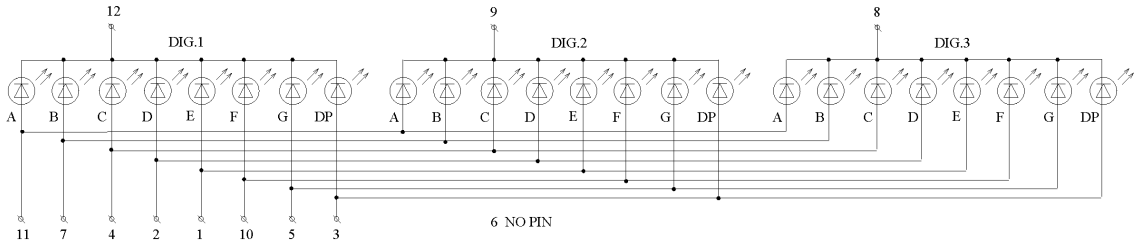


а

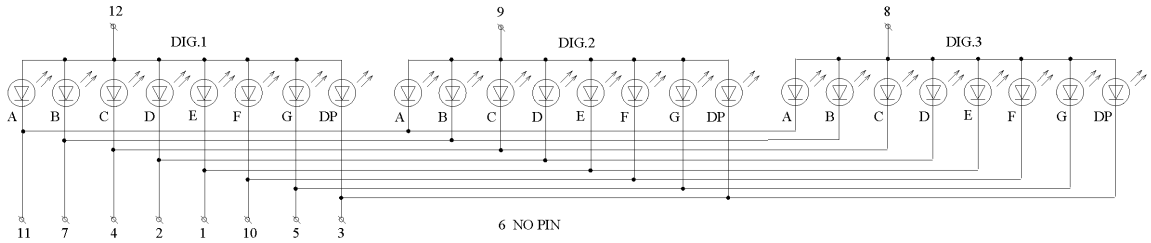


б

Рис. 2.38.



а



б

Рис. 2.39.

На рис. 2.40 представлена схема двоколірного одноразрядного світлодіодного цифрового дисплею, а на рис. 2.41 – двоколірні одноразрядні світлодіодні цифрові дисплеї: а – GNS-40011 AEG (спільний катод); б – GNS-40011 BEG (спільний анод).

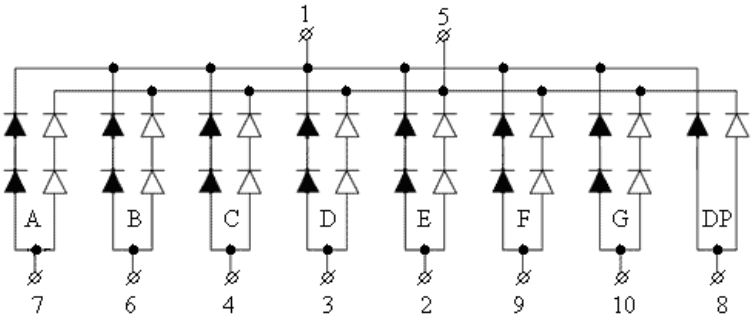
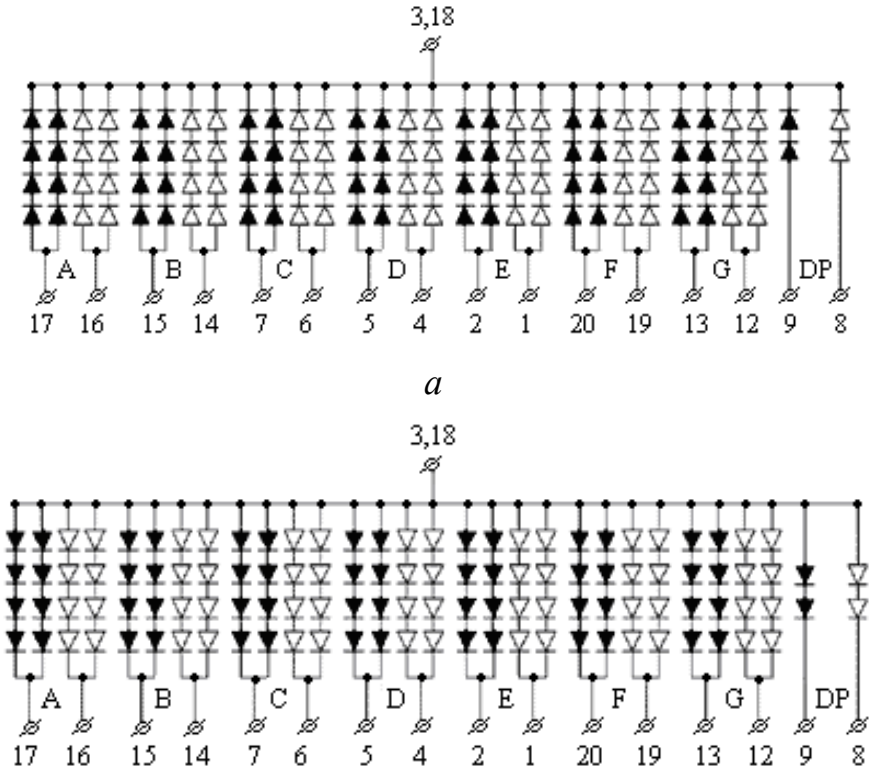


Рис. 2.40.



а

б

Рис. 2.41.

Існує два методи управління цифровими індикаторами: управління в статичному режимі (або в режимі постійного струму) і в мультиплексному режимі (режимі послідовного стробування цифр).

2.6.1. Статичне управління

Статичний режим рекомендується використовувати для управління напівпровідниковими цифровими індикаторами в пристроях відображення інформації з малою інформативною ємністю. Для цього використовують індикатори першої групи, що мають окремий управляючий вивід для кожного елементу індикації (сегменту).

Приклад 2.1. Розробити схему і написати програму для мікроконтролера, яка підраховує кількість натискань на кнопку і відображає значенняна на двох семисегментних індикаторах (загальний катод).

На рис. 2.42 представлена схема пристрою підрахунку кількості натискань на кнопку. В пристрої використовується мікроконтролер AT90S2313.

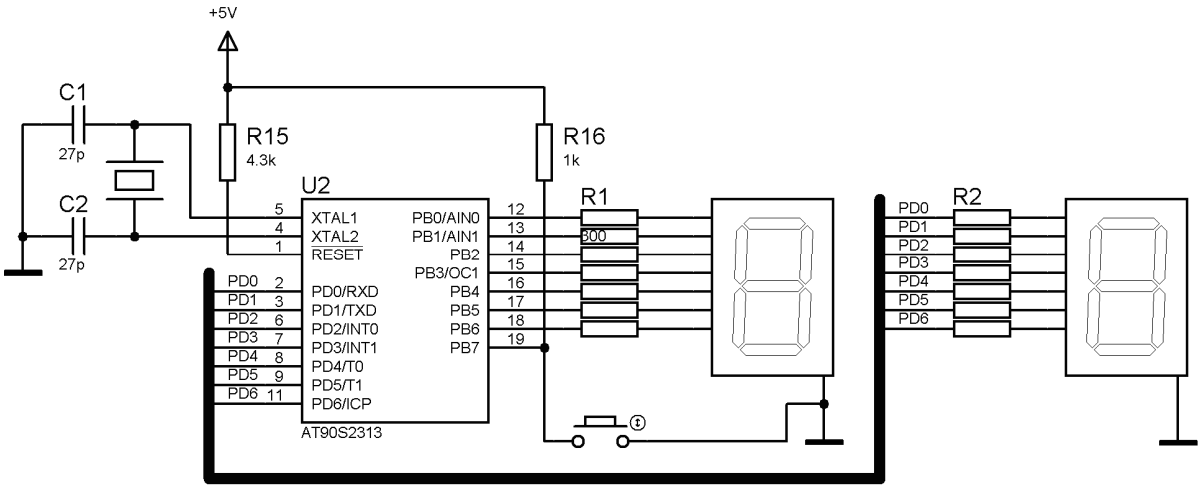


Рис.2.42.

Для роботи мікроконтролера необхідно підключити кварцевий резонатор до виводів XTAL1 і XTAL2, а також через резистор опором 4,3 кОм з'єднати вивід RESET з шиною живлення. Індикатори підключаються до портів мікроконтролера через опори 300 Ом. Для того, щоб уникнути появи помилкових сигналів, вивід PB7 мікроконтролера, до якого приєднана кнопка, підтягнутий до шини живлення за допомогою резистора опором 1 кОм.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>          // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)            // функція формування затримки
{
 int c1, c2;                  // додаткові змінні
 for(c1=0; c1<ms; c1++)       // у циклі виконується пуста операція
  for(c2=0; c2<970; c2++) NOP();
}
void main()                   // головна функція програми
{
// оголошення масиву з кодами цифр для 7-сегментного індикатора
 char segment[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
 0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
 char cnt=0;                  // змінна – лічильник
 DDRB=0b01111111;             // налаштування 7 біта порта B на вихід
 DDRD=0b01111111;             // налаштування 7 біта порта D на вихід
 while(1)                     // вічний цикл
 {
  PORTB=segment[cnt/10];   // виведення десяток на перший індикатор
  PORTD=segment[cnt%10];// виведення одиниць на другий індикатор
  if(((PINB&0x80) && (cnt<99))==0) cnt++;
  // при натисканні на кнопку значення лічильника збільшується
  if(cnt>99) cnt=0;      // якщо значення лічильника більше 99, відбувається обнуління
```

```
 delay(100);            // затримка 100 мс
 }
}
```

Коди цифр для семисегментного індикатора зберігаються в масиві. Для відображення потрібної цифри на індикаторі в порт виводиться значення елементу масиву, індекс якого відповідає цифрі, що виводиться. При досягненні лічильником значення 99, відбувається його онулення.

Наступний приклад демонструє принципи управління індикаторами зі спільним анодом.

Приклад 2.2. Розробити схему і написати програму для мікроконтролера, яка підраховує кількість натискань на кнопку і відображає значення на двох семисегментних індикаторах (загальний анод).

Схема на рис. 2.43 повністю аналогічна розглянутій вище, за винятком того, что семисегментні індикатори з загальним анодом підключені до шини живлення, а катоди окремих сегментів – до виводів мікроконтролера через резистори опором 300 Ом.

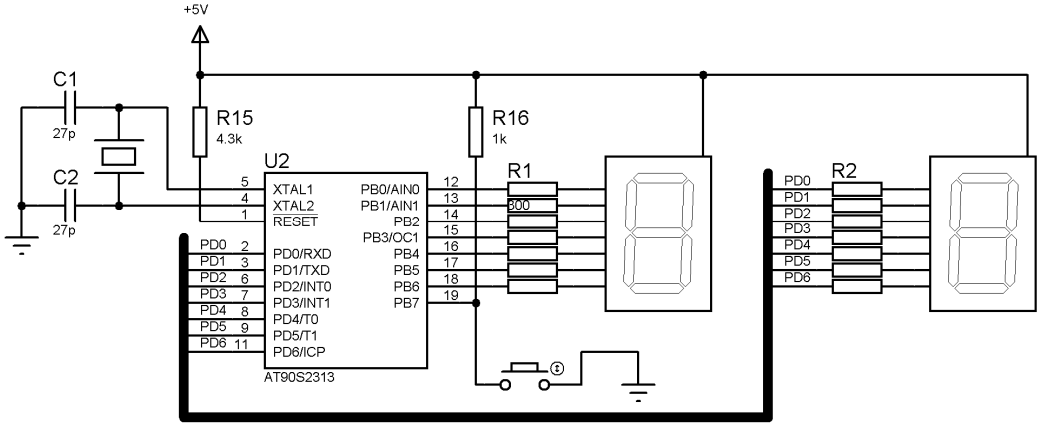


Рис. 2.43.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>          // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)            // функція формування затримки
{
 int c1, c2;                  // додаткові змінні
 for(c1=0; c1<ms; c1++)       // у циклі виконується пуста операція
  for(c2=0; c2<970; c2++) NOP();
}
void main()                   // головна функція програми
{
// оголошення масиву з кодами цифр для 7-сегментного індикатора
 char segment[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
 0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
 char cnt=0;                  // змінна – лічильник
```

```
 DDRB=0b01111111;          // налаштування 7 біту порта B на вихід
 DDRD=0b01111111;          // налаштування 7 біту порта D на вихід
 while(1)                  // вічний цикл
 {
  // виведення десятих на перший індикатор (з інверсією коду)
  PORTB=~segment[cnt/10];
  // виведення одиниць на другий індикатор (з інверсією коду)
  PORTD=~segment[cnt%10];
  if(((PINB&0x80) && (cnt<99))==0) cnt++;
  // при натисненні на кнопку значення лічильника збільшується
  if(cnt>99) cnt=0;     // якщо значення лічильника більше 99, відбувається онулення
  delay(100);           // затримка 100 мс
 }
}
```

Відмінність цієї програми від попередньої полягає в тому, що при виведенні кодів 7-сегментних індикаторів в порти мікроконтролера вони інвертуються, оскільки при підключенні індикатора із загальним анодом сегмент світиться, якщо відповідний катод сполучений із землею. Іншими словами, щоб запалити сегмент необхідно на виводі мікроконтролера встановити рівень лог. 0. Саме схему із загальним анодом рекомендується використовувати при підключенні індикаторів до мікроконтролера.

Приклад 2.3. Розробити принципову схему годинника і написати відповідну програму для мікроконтролера. На чотирьох 7-сегментних індикаторах виводити значення годин і хвилин, а за допомогою 8 світлодіодів виводити значення секунд у двійковому коді. Передбачити можливість установки часу.

На рис. 2.44 представлена принципова схема годинника. Для спрощення управління семисегментними індикаторами використовується паралельні регістри. При подачі на вхід LE (Load Enable) сигналу високого рівня, в регістр записуються дані, що присутні на його входах, і зберігаються до наступного перезапису. Таке схемотехнічнері шення дозволило використовувати загальну шину для підключення до неї індикаторів і принцип статичної індикації. Для вибору відповідного регістра використовується дешифратор двійкового коду в десятковий. Входи дешифратора сполучені з виводами PA0-PA2 мікроконтролера і двійковий код на цих входах відповідає тому індикатору, в регістр якого відбувається запис даних. У схемі використані семисегментні індикатори із загальним катодом. Для установки часу використовуються дві кнопки, підтягнуті до лінії живлення через резистори номіналом 1 кОм. При натисканні на першу кнопку змінюються години, а при натисненні на другу кнопку змінюються хвилини. Опори струмообмежувальних резисторів, через які підключені світлодіодні індикатори, дорівнюють 300 Ом.

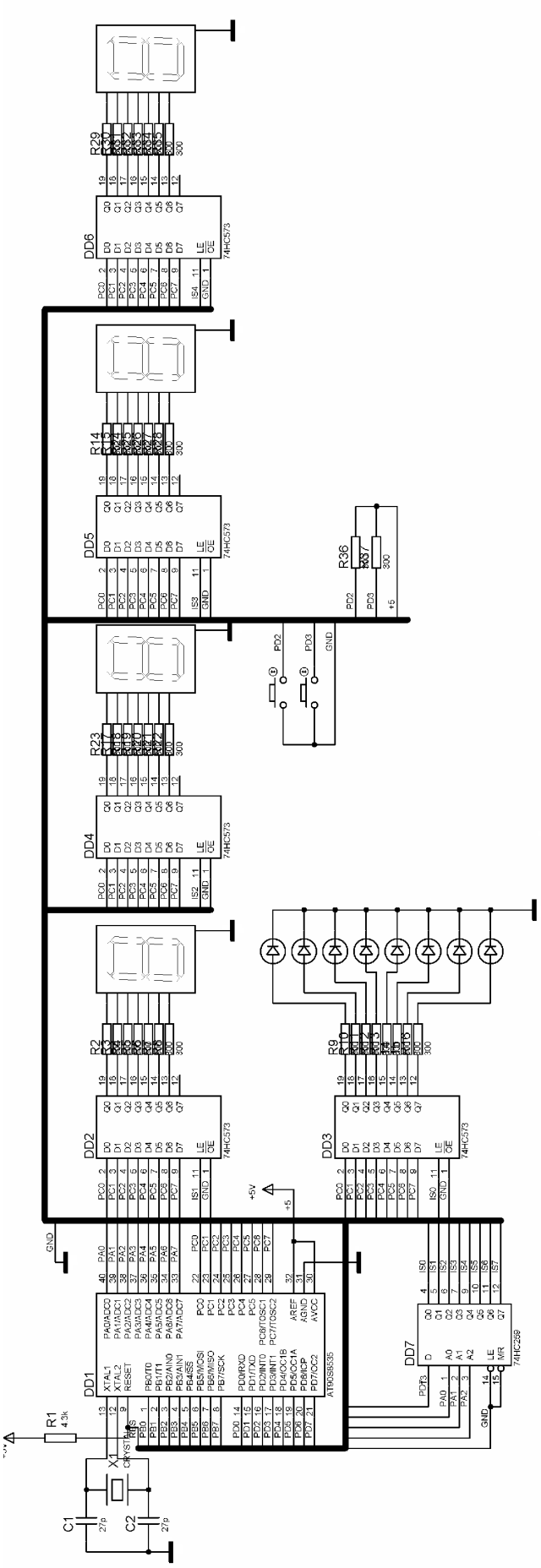


Рис. 2.44.

Текст програми для мікроконтроллера представлено нижче.

```
#include <io8535v.h>          // підключення заголовочних файлів
#include <macros.h>

void init_cpu(void);          // оголошення функції управління портами
void delay(int ms);           // оголошення функції затримки
void showtime(void);          // оголошення функції виведення часу на індикатори

 char sec=0, min=0, hour=0; // змінні для зберігання секунд, хвилин і годин
 // масив з кодами для 7-сегментного індикатора
 char seg[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
 0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
 // обробник зовнішнього переривання INT0
#pragma interrupt_handler int0_isr:2
void int0_isr(void)
{
 if(hour<23) hour++;          // якшо значення годин менше 23, інкремент змінної
 else hour=0;                 // інакше – онулення змінної
 showtime();                  // виклик функції виведення значення часу на індикатори
}
// обробник зовнішнього переривання INT1
#pragma interrupt_handler int1_isr:3
void int1_isr(void)
{
 if(min<59) min++;            // якшо значення хвилин менше 59, інкремент змінної
 else min=0;                  // інакше – онулення змінної
 showtime();                  // виведення значення часу на індикатори
}

void init_cpu()               // функція ініціалізації мікроконтролера
{
 // налаштування старшого і трьох молодших виводів порту A на вихід
 DDRA=0b10000111;  PORTA=0;   // виведення значення 0 в порт A
 DDRB=0;                      // порт B налаштований на вхід
 PORTB=0;                     // виведення значення 0 в порт B
 DDRC=0xFF;                   // налаштування порта C на вихід
 PORTC=0;                     // виведення значення 0 в порт C
 DDRD=0x80;                   // налаштування старшого біта порту D на вихід
 PORTD=0x80;                  // установка лог.1 на виході PD7
 MCUCR = 0x0A;                // переривання виникає за зрізом імпульса
 GIMSK = 0xC0;                // дозвіл обробки зовнішніх переривань
 SEI();                       // дозвіл обробки переривань
}

void delay(int ms)            // опис функції затримки
{
 int c, c1;                   // додаткові змінні
 for(c=0;c<ms;c++)            // у циклі виконується пуста операція
```

```
 {
  for(c1=0;c1<970;c1++) NOP();
 }
}

void showtime()              // функція виведення часу на індикатори
{
  PORTA=0;                   // в порт А виводиться значення 0
  PORTC=sec;                 // виведення на світлодіодні індикатори значення секунд
  PORTA=1;                   // в порт А виводиться значення 1
  PORTC=seg[hour/10];        // виведення десяток годин
  PORTA=2;                   // в порт А виводиться значення 2
  PORTC=seg[hour%10];        // виведення одиниць годин
  PORTA=3;                   // в порт А виводиться значення 3
  PORTC=seg[min/10];         // виведення десяток хвилин
  PORTA=4;                   // в порт А виводиться значення 4
  PORTC=seg[min%10];         // виведення одиниць хвилин
}

void main()                  // головна функція програми
{
 init_cpu();                 // виклик функції ініціалізації мікроконтролера
 while(1)                    // вічний чикл
 {
  showtime();                // виведення значення часу на індикатори
  if(sec<59) sec++;          // якщо значення секунд менше 59, інкремент змінної
  else                       // в протилежному випадку
  {
   sec=0;                    // онулення лічильника секунд
   if(min<59) min++;         // якщо значення хвилин менше 59, інкремент змінної
   else                      // інакше
   {
        min=0;               // онулення лічильника хвилин
        if(hour<23) hour++;  // якщо значення годин менше 23, інкремент змінної
        else hour=0;         // інакше онулення змінної
   }
  }
   delay(1000);              // затримка 1 секунда (1000 мс)
 }
}
```

У функції init_cpu() виконується налаштування портів мікроконтролера. У функції showtime() реалізований алгоритм виведення часу на індикатори. Можливість зміни часу реалізована з використанням механізму переривань. У програмі не відбувається постійний опит стану кнопок. При їх натисканні викликаються відповідні обробники зовнішніх переривань, в яких відбуваються зміни значень змінних. Функція delay() формує часову затримку. Як параметр в цю функцію передається значення затримки в мілісекундах.

У наступному прикладі показано, яким чином створити біжучий рядок на семисегментних індикаторах.

Приклад 2.4. Написати програму для мікроконтролера, яка створює ефект біжучого рядка на 4-х семисегментних індикаторах. Виводити на індикатори слово «HELP». Використовувати схему з попереднього прикладу (рис. 2.44).

```
#include <io8535v.h>          // підключення заголовочних файлів
#include <macros.h>
void init_cpu()               // функція ініціалізації мікроконтролера
 {
 DDRA=0b00000111;             // налаштування трьох молодших виводів порту А на вихід
 DDRC=0b11111111;             // налаштування порту С на вихід
 DDRD=0b10000000;             // налаштування виводу PD7 на вихід
 PORTD=0b10000000;            // установка на виводі PD7 сигналу високого рівня
 }

void delay (int ms)           // функція затримки
 {
 int c1,c2;                   // оголошення додаткових змінних
 for (c1=0;c1<ms;c1++)
 for (c2=0;c2<970;c2++)  NOP(); // у циклі виконується «пуста» операція
 }
void main()                   // головна функція програми
 {
 char arr4[7]={0,0,0,118,121,56,115};      // в масивах зберігаються коди букв «HELP»
 char arr3[7]={0,0,118,121,56,115,0};
 char arr2[7]={0,118,121,56,115,0,0};
 char arr1[7]={118,121,56,115,0,0,0};
 char i;                      // додаткова змінна
 init_cpu();                  // виклик функції ініціалізації мікроконтролера
 while(1)                     // вічний цикл
 {
 for (i=0;i<7;i++)            // у циклі в порт виводяться значення елементів масивів
{
 PORTA=1;                     // в порт А виводиться значення 1
 PORTC=arr4[i];               // виведення в порт С елементу масиву arr4
 PORTA=2;                     // в порт А виводиться значення 2
 PORTC=arr3[i];               // виведення в порт С елементу масиву arr3
 PORTA=3;                     // в порт А виводиться значення 3
 PORTC=arr2[i];               // виведення в порт С елементу масиву arr2
 PORTA=4;                     // в порт А виводиться значення 4
 PORTC=arr1[i];               // виведення в порт С елементу масиву arr1
 delay (500);                 // затримка 500 мс
}
}
}
```

У масивах arr1-arr4 зберігаються коди букв H, E, L і P, які будуть виводитись на семисегментні індикатори. У кожному із масивів коди букв зміщені на один елемент. Таким чином, при виводі в циклі на індикатори елементів відповідних масивів, буде створюватися ефект біжучого рядка.

Приклад 2.5. Написати програму, яка створює світловий ефект «хвиля» на 7-сегментних індикаторах. Використовувати схему з попереднього прикладу (рис. 2.44).

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>          // підключення заголовочних файлів
#include <macros.h>
void init_cpu()               // функція ініціалізації мікроконтролера
 {
 DDRA=0b00000111;             // налаштування трьох молодших виводів порту A на вихід
 DDRC=0b11111111;             // налаштування порту C на вихід
 DDRD=0b10000000;             // налаштування виводу PD7 на вихід
 PORTD=0b10000000;            // установка на виводі PD7 сигналу високого рівня
 }
void delay (int ms)           // функція затримки
 {
 int c1,c2;                   // оголошення додаткових змінних
 for (c1=0;c1<ms;c1++)
 for (c2=0;c2<970;c2++)  NOP(); // у циклі виконується «пуста» операція
 }
void main()                   // головна функція програми
{
int i;                        // додаткова змінна
char arr1[6]={0x01,0x02,0x04,0x08,0x10,0x20};
char arr2[6]={0x02,0x04,0x08,0x10,0x20,0x01};
char arr3[6]={0x04,0x08,0x10,0x20,0x01,0x02};
char arr4[6]={0x08,0x10,0x20,0x01,0x02,0x04};
init_cpu();                   // виклик функції ініціалізації мікроконтролера
while(1)                      // вічний цикл
{
for(i=0;i<6;i++)              // у циклі в порт C виводяться елементи масивів
{
PORTA=1;                      // в порт A виводиться значення 1
PORTC=arr1[i];                // виведення в порт C елементу масиву arr1
PORTA=2;                      // в порт A виводиться значення 2
PORTC=arr2[i];                // виведення в порт C елементу масиву arr2
PORTA=3;                      // в порт A виводиться значення 3
PORTC=arr3[i];                // виведення в порт C елементу масиву arr3
PORTA=4;                      // в порт A виводиться значення 4
PORTC=arr4[i];                // виведення в порт C елементу масиву arr4
delay(50);                    // затримка 50 мс
}
}
}
```

За аналогією з попереднім прикладом, в масивах зберігаються коди знаків, що відображуються. Оскількі семисегментний індикатор представляє собою орієнтовані в просторі світлодіоди, то шляхом маніпуляцій над бітами слів, які виводяться в порт, можна досягти різних світлових ефектів.

Приклад 2.6. Написати програму для мікроконтролера AT90S8535, яка виконує зчитування температури з датчика DS1621 по інтерфейсу I^2C і виводить значення температури на 2 семисегментних індикатора.

На рис. 2.45 представлена принципова схема електронного термометра.

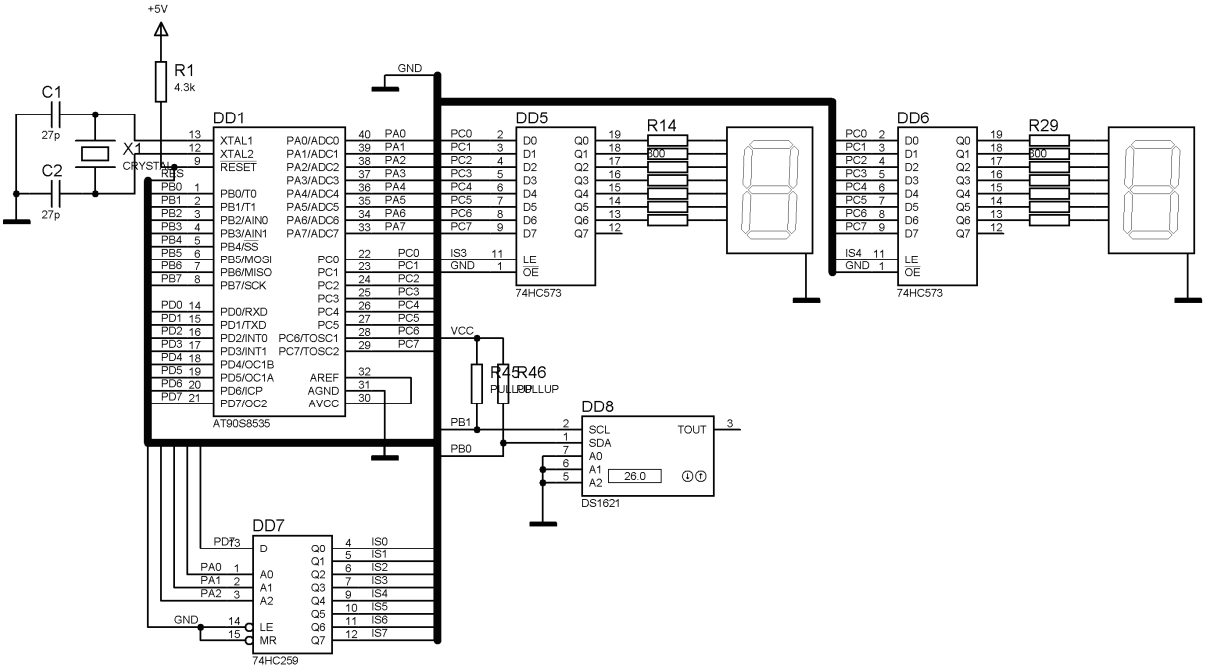


Рис. 2.45.

Лінії SDA і SCL підтягнуті за допомогою резисторів до лінії живлення. Для використання статичної індикації в схемі є два паралельні регістри, які адресуються дешифратором DD7. Подібне рішення схемотехніки дозволяє заощадити виводи мікроконтролера і спростити програмну реалізацію алгоритму виведення значення температури на індикатори.

Текст програми для мікроконтролера, створеної з використанням компілятора ImageCraft, представлено нижче.

```
#include <io8535v.h>                    // підключення заголовочних файлів
#include <macros.h>
#define SETSDA() PORTB|=0b00000001      // уст. сигналу SDA
#define SETSCL() PORTB|=0b00000010      // уст. сигналу SCL
#define RSTSDA() PORTB&=~0b00000001     // скидання сигналу SDA
#define RSTSCL() PORTB&=~0b00000010     // скидання сигналу SCL
    // функція ініціалізації контролера
```

```
void init_cpu()
{
        DDRA=0x07;              // 3 молодших біта порту A налаштовані на вихід
        DDRC=0xFF;              // порт C налаштований на вихід
        DDRD=0x80;              // налаштування виводу 7 порту D на вихід
        PORTD=0x80;             // установка 1 на виводі 7 порту D
}
        // функція затримки; параметр – затримка в мс
void delay_ms(int ms)
{
        int c1, c2;             // оголошення додаткових змінних
        for(c1=0;c1<ms;c1++)    // у циклі виконується «пуста» операція
        for(c2=0;c2<970;c2++) NOP();
}
        // функція затримки; параметр – кількість тактів затримки
void delay_us(int st)
{
        int c1;                 // оголошення додаткових змінних
        for(c1=0;c1<st;c1++) NOP();  // у циклі виконується «пуста» операція
}
        // функція перевірки «зайнятості» шини
char i2c_isbusy()
        {
        char st;                // додаткова змінна
        DDRB=0x00;              // налаштування порту B на вхід
        PORTB=0x00;             // відключення підтягуючих резисторів
        if((PINB&0x03)!=0x03) st=1;// перевірка стану ліній
        else st=0;              // функція повертає 0 якщо шина вільна
        return st;              // інакше повертається значення 1
}
        // функція формування сигналу початку передачі даних
void i2c_start()
{
        DDRB=0x03;              // налаштування 2 молодших біт порту B на вихід
        SETSDA();               // установка сигналу SDA
        SETSCL();               // установка сигналу SCL
        delay_us(2);            // затримка
        RSTSDA();               // скидання сигналу SDA
        delay_us(2);            // затримка
        RSTSCL();               // скидання сигналу SCL
        delay_us(2);            // затримка
}
        // функція формування сигналу завершення обміну даними
void i2c_stop()
{
        DDRB=0x03;              // налаштування 2 молодших біт порту B на вихід
        RSTSDA();               // скидання сигналу SDA
        SETSCL();               // установка сигналу SCL
        delay_us(2);            // затримка
        SETSDA();               // установка сигналу SDA
```

```
        delay_us(2);                    // затримка
}
        // функція передачі 1 байту; параметр – байт, що передається
unsigned char i2c_sendbyte(char d)
{
 unsigned char i=8,acknowledge=0;// додаткові змінні
  do {                                  // у циклі виконується передча 8 бит
   if((d & 0x80)!=0) SETSDA();          // установка сигналу SDA якщо
                                        // старший біт дорівнює 1
        else RSTSDA();                  // інакше скидання сигналу SDA
   d =d << 1;                           // зсув байта, що передається, на 1 розряд ліворуч
   delay_us(2);                         // формування затримки
                                        // формування тактового імпульса
   SETSCL();                            // установка сигналу SCL
        delay_us(2);                    // затримка
        RSTSCL();                       // скидання сигналу SCL
        delay_us(2);                    // затримка
  }
  while (--i);                          // цикл виконується 8 разів
        // формування тактового імпульса для біта підтвердження
  DDRB=0x02;                            // переведення лінії SDA в третій стан
  RSTSDA();                             // скидання сигалу SDA
  SETSCL();                             // установка сигналу SCL
  delay_us(2);                          // затримка
  if ((PINB&0x01)==0) acknowledge=1; // якщо на лінії SCL низький
                                        // рівень, біт підтвердження прийнято
  RSTSCL();                             // скидання сигналу SCL
  delay_us(2);                          // затримка
  DDRB=0x03;                            // налаштування 2 молодших біт порту B на вихід
  return acknowledge;                   // якщо біт підтвердження прийнято, повертається 1
}
        // функція зчитування одного байта
unsigned char i2c_readbyte(unsigned char acknowledge)
{
  unsigned char i=8,i2c_data=0;         // додаткові змінні
  char d;
  DDRB=0x02;                            // переведення лінії SDA в третій стан
  do {
    i2c_data<<=1;                       // зсув на 1 розряд ліворуч
    SETSCL();                           // установка сигналу SCL
        delay_us(2);                    // затримка
        d=PINB;                         // зчитування стану порту B
        d=d&0x01;                       // виділення 0 біта
    i2c_data|=d;                        // копіювання 0 біта в змінну
    RSTSCL();                           // скидання сигналу SCL
        delay_us(2);                    // затримка
  }
  while (--i);                          // цикл виконується 8 разів
  if (acknowledge)                      // якщо необхідний біт підтвердження
        {
        DDRB=0x03;                      // налаштування 2-х молодших біт на вивід
```

```
        RSTSDA();                   // скидання сигналу SDA
        }
        // формування тактового імпульса
  SETSCL();                         // установка сигналу SCL
  delay_us(2);                      // затримка
  RSTSCL();                         // скидання сигналу SCL
        DDRB=0x02;                  // переведення лінії SDA в третій стан
 return i2c_data;                   // функція повертає прийнятий байт
}

int main ()                         // головна функція
{
        int c1=0,c2=0;              // оголошення додаткових змінних
        int t1, t2;
char seg[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
0b01100110, 0b01101101, 0b01111101, 0b00000111, 0b01111111,
0b01101111};                        // дані з кодами символів для 7-сегм. індикатора
init_cpu();                         // виклик функції ініціалізації контролера
while(1)                            // вічний цикл
{
while (i2c_isbusy()!=0;             // чекаємо доки лінія не звільниться
DDRB=0x03;                          // налаштування 2-х молодших біт порту B на вихід
i2c_start();                        // функція формування сигналу початку передачі
i2c_sendbyte(0b10010000);           // передача байта 0x90 (адреса)
!i2c_sendbyte(0xEE);                // передача байта 0xEE (команда)
i2c_stop();                         // формування сигналу закінчення передачі
i2c_start();                        // початок передачі даних
i2c_sendbyte(0b10010000);           // передача байта 0x90 (адреса)
i2c_sendbyte(0xAA);                 // передача байта 0xAA
i2c_stop();                         // звершення передачі даних
i2c_start();                        // повторний старт
i2c_sendbyte(0b10010001);           // передача команди зчитування даних
c1=i2c_readbyte(1);                 // зчитування старшого байта температури
c2=i2c_readbyte(0);                 // зчитування молодшего байта температури
i2c_stop();                         // звільнення лінії I2C
t1=c1/10;                           // формування значення температури (десятки)
t2=c1-t1*10;                        // формування значення температури (одиниці)
PORTA=4;                            // ввімкнення лівого індикатора
PORTC=seg[t2];                      // виведення значення температури на індикатор
PORTA=3;                            // ввімкнення правого індикатора
PORTC=seg[t1];
delay_ms(1000);                     // затримка 1 с
}
}
```

На рис. 2.46 представлена діаграма роботи інтерфейса I^2C. Тактовий сигнал SCL формує мікроконтролер на виводі PB1. На лінії даних SDA (вивід PB0) встановлюються рівні лог.1 і 0 відповідно до байта, що передається. Після 8 тактових імпульсів, на 9 відбувається перевірка біта

підтвердження, який формує ведений пристрій (датчик температури DS1620).

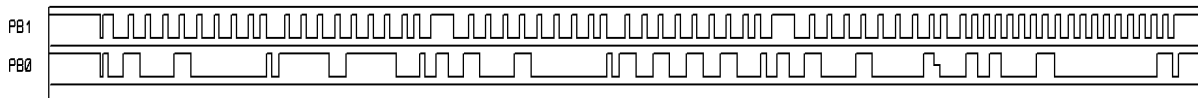


Рис. 2.46.

Наступний приклад демонструє спосіб управління семисегментними індикаторами, при якому використовується мінімальна кількість виводів мікроконтролера.

Приклад 2.7. Розробити пристрій підрахунку натискань на кнопку. Кількість десяткових розрядів лічильника дорівнює 4 (лічильник від 0 до 9999).

На рис. 2.47 представлена принципова схема лічильника. Для зменшення кількості виводів мікроконтроллера, необхідних для управління чотирма семисегментними індикаторами, використовуються регістри зсуву. Дані, що виводяться на індикатори, передаються від мікроконтролера в послідовному форматі, для чого потрібно всього 2 виводи – тактовий та інформаційний. Оновлення інформації на індикаторах відбувається за 32 такти. Оскільки оновлення інформації виконується з високою частотою, користувач не спостерігає мерехтіння індикаторів. Подібне рішення дозволяє використовувати мікроконтролери з малим числом виходів. Виходи кожного з регістрів зсуву приєднані до окремих індикаторів.

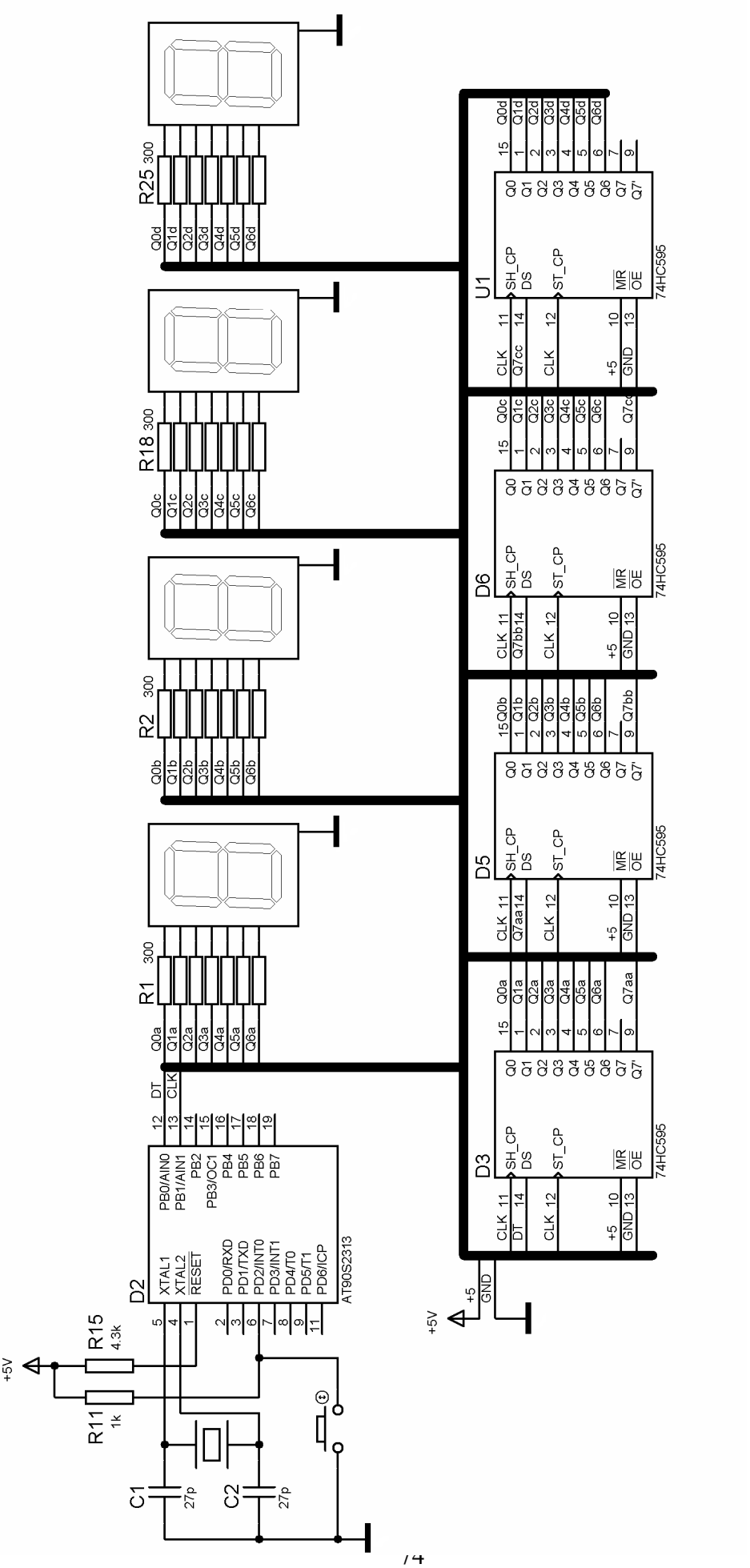

D2
AT90S2313
XTAL1
XTAL2
RESET
PB0/AIN0
PB1/AIN1
PB2
PB3/OC1
PB4
PB5
PB6
PB7
PD0/RXD
PD1/TXD
PD2/INT0
PD3/INT1
PD4/T0
PD5/T1
PD6/ICP
DT
CLK
R15 4.3k
R11 1k
C1 27p
C2 27p
+5V
R1 300
R2 300
R18 300
R25 300
D3
D5
D6
U1
74HC595
SH_CP
ST_CP
MR
OE
DS
CLK
GND
+5
Q0a
Q1a
Q2a
Q3a
Q4a
Q5a
Q6a
Q7aa
Q0b
Q1b
Q2b
Q3b
Q4b
Q5b
Q6b
Q7bb
Q0c
Q1c
Q2c
Q3c
Q4c
Q5c
Q6c
Q7cc
Q0d
Q1d
Q2d
Q3d
Q4d
Q5d
Q6d

Рис. 2.47.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>                  // підключення заголовочних файлів
#include <macros.h>
int cnt;                              // змінна – лічильник

#pragma interrupt_handler int0_isr:2 // обробник зовнішнього переривання
void int0_isr(void)
{
 // якщо значення лічильника менше 9999, відбувається його інкремент
 if(cnt<9999) cnt++;
 else cnt=0;                          // інакше – обнуління лічильника
}
void delay(int ms)                    // функція часової затримки
{
 int c1, c2;                          // оголошення додаткових змінних
 for(c1=0; c1<ms; c1++)               // в циклі виконується «пуста» операція
  for(c2=0; c2<970; c2++) NOP();
}
void main()                           // головна функція програми
{
 char d[4]={0,0,0,0};                 // оголошення масиву из 4-х елементів
 char i, j, tmp;                      // оголошення додаткових змінних
 // оголошення масиву з кодами для 7-сегментного індикатора
 char segment[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
 DDRB=0b00000011;                     // налаштування виводів PB0 і PB1 на вихід
 DDRD=0;                              // налаштування порту D на вихід
 MCUCR = 0x02;                        // обробка переривання за спадом імпульса
 GIMSK = 0x40;                        // дозвіл обробки зовнішнього переривання
 cnt=0;                               // онулення лічильника
 SEI();                               // дозвіл обробки переривань
 while(1)                             // вічний цикл
 {
  d[3]=segment[cnt/1000];             // в 0 елемент масиву записуються тисячі
  d[2]=segment[(cnt%1000)/100];       // в 1 елемент масиву записуються сотні
  d[1]=segment[((cnt%1000)%100)/10]; // в 2 елемент масиву записуються десятки
  d[0]=segment[((cnt%1000)%100)%10]; // в 3 елемент масиву записуються одиниці
PORTB|=0x02;                          // установка на виводі PB1 сигналу лог.1.
  PORTB&=~0x02;                       // установка на виводі PB1 сигналу лог.0.
  for(j=0;j<4;j++)                    // у циклі відбувається зсув значень
  {                                   // для передачі їх в послідовному форматі
  for(i=0;i<8;i++)
  {
   tmp=d[j]>>(7-i);                   // зсув значення елемента масиву
   tmp&=0x01;                         // виділення молодшего розряду
   PORTB=tmp;                         // виведення значення в порт B
   PORTB|=0x02;                       // установка на виводу PB1 сигналу лог.1
   delay(1);                          // затримка 1 мс
   PORTB&=~0x02;                      // установка на виводі PB1 сигналу лог.0
```

```
  delay(1);                    // затримка 1 мс
 }}
  PORTB|=0x02;                 // установка на виводі PB1 сигналу лог.1.
  PORTB&=~0x02;                // установка на виводі PB1 сигналу лог.0.
  delay(200);                  // затримка 200 мс
}}
```

У програмі результатом виконання оператора «%» є залишок від ділення. Елементами масиву d[] будуть значення тисяч, сотень, десятків і одиниць. У циклі відбувається зсув кожного з елементів масиву, і молодший біт встановлюється на виводі PB0. Після цього на виводі PB1 формується тактовий імпульс, по фронту якого відбувається запис значення в регістр зсуву. Після 32 тактів на виводах регістрів встановлюються значення, відповідні тим, що відображується.

На рис. 2.48 представлена часова діаграмма сигналів, що формуються мікроконтролером.

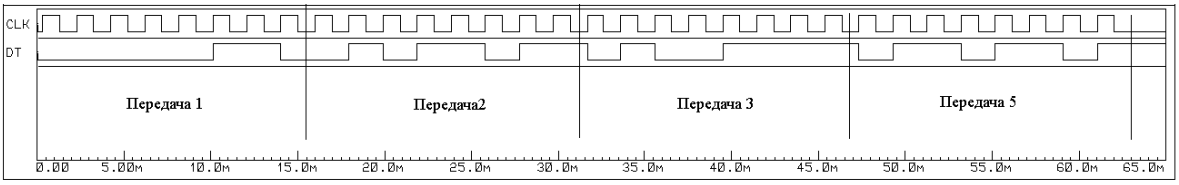


Рис. 2.48.

2.6.2. Мультиплексне управління

Мультиплексний режим рекомендується застосовувати для управління напівпровідниковими цифровими індикаторами з великою інформаційною ємністю. У таких пристроях використовуються в основному індикатори другої групи – індикатори монолітної конструкції, що мають управляючі виводи для однойменних сегментів декількох цифр, розміщених в одному корпусі. Мультиплексний режим управління напівпровідниковими цифровими індикаторами дозволяє знизити енергоспоживання індикаторів при збереженні їх характеристик яскравості.

Приклад 2.8. Розробити схему лічильника натискань на кнопку, в якій використовується динамічна індикація. Написати програму для мікроконтролера.

На рис. 2.49 представлена принципова схема управління цифровими індикаторами в мультиплексному режимі. У схемі використаний двохрозрядний семисегментний індикатор зі спільним катодом. Оскільки сумарний струм загального катода індикатора перевищує максимально допустиме значення струму, що формується виводом мікроконтролера, в схемі використані біполярні транзистори, які працюють в ключовому режимі. При відкритому транзисторі VT1 відображається цифра на першому індикаторі, а при відкритому транзисторі VT2 – на другому.

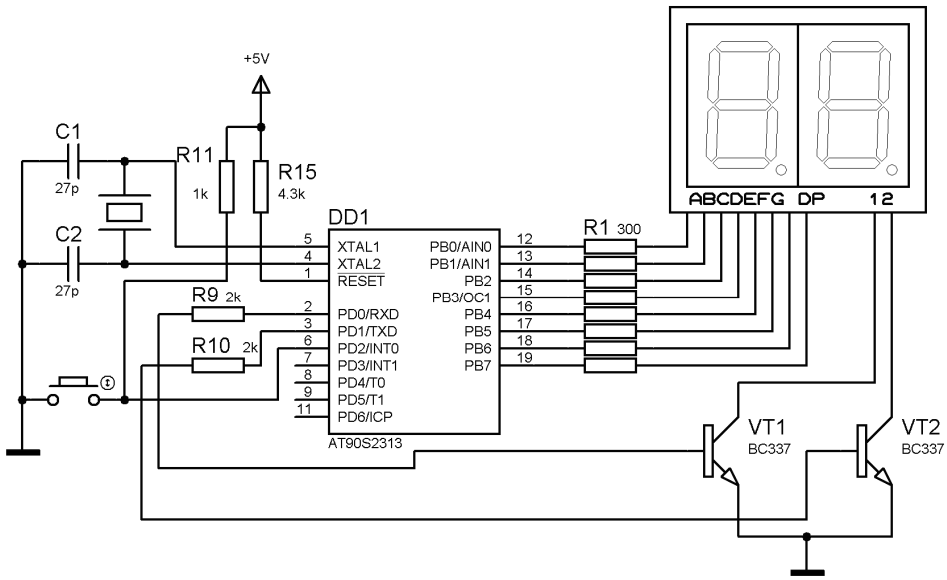


Рис. 2.49.

У запропонованій схемі виводи однойменних сегментів всіх цифр сполучені паралельно і підключені до відповідних виходів мікроконтролера. Транзистори VT1 і VT2 виконують почергове підключення загального виводу кожного з сегментів до землі.

У першому такті мікроконтролер видає на порт B інформаційний код для відображення першої цифри. У другому такті мікроконтролер вимикає загальний вивід першої цифри, формуючи сигнал низького рівня на виводі PD0, тим самим замикаючи транзистор VT1, і підключає загальний вивід другої цифри шляхом відкриття транзистора VT2. Мікроконтролер по другому імпульсу встановлює на порту B код для відображення другої цифри, відключивши код першої. Потім цикл повторюється.

Час протікання струму через елемент, що світиться, обернено пропорційний до кількості цифр в керованому наборі. Отже, значення середнього прямого струму сегментів і яскравість їх світіння також скорочуються.

Текст програми представлено нижче.

```
#include <io2313v.h>                    // підключення заголовочних файлів
#include <macros.h>
char cnt;                               // змінна – лічильник
#pragma interrupt_handler int0_isr:2// обробник зовнішнього переривання
void int0_isr(void)
{
  // якщо значення лічильника менше 99, відбувається його інкремент
  if(cnt<99) cnt++;
  else cnt=0;                           // в протилежному випадку лічильник онуляється
}
void delay(int ms)                      // функція затримки
{
```

```
 int c1, c2;                          // додаткові змінні
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();      // у циклі виконується «пуста» операція
}
void main()                           // головна функція програми
{
        // оголошення масиву з кодами для 7-сегментного індикатора
 char segment[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};
 DDRB=0b01111111;                     // налаштування виводів PB0-PB6 на вихід
 DDRD=0b00000011;                     // налаштування виводів PD0 і PD1 на вихід
 MCUCR = 0x02;                        // переривання виникає по спаду сигналу
 GIMSK = 0x40;                        // дозвіл обробки переривання INT0
 cnt=0;                               // онулення лічильника
 SEI();                               // дозвіл обробки переривань
 while(1)                             // вічний цикл
 {
  PORTD=1;                            // вмикання першого індикатора
  PORTB=~segment[cnt/10];             // виведення значення десятих на індикатор
  delay(20);                          // затримка 20 мс
  PORTD=2;                            // вмикання другого індикатора
  PORTB=~segment[cnt%10];             // виведення значення одиниць на індикатор
  delay(20);                          // затримка 20 мс
 }
}
```

На рис. 2.50 приводиться часова діаграма, що пояснює роботу схеми і принцип динамічної індикації. Як видно з діаграми, на базах транзисторів сигнал формується в протифазі, що виключає виведення на обидва сегменти однакових значень.

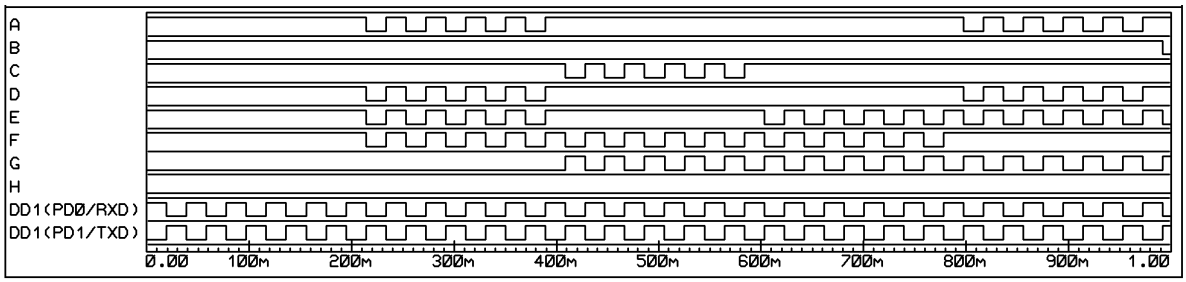


Рис. 2.50.

У наступному прикладі демонструється використання принципу динамічної індикації для багаторозрядних семисегментних індикаторів.

Приклад 2.9. Розробити схему і написати програму виведення числа 2981 на 7-сегментні індикатори з використанням динамічної індикації.

На рис. 2.51 представлена принципова схема пристрою. В цілому, принцип роботи схеми не відрізняється від розглянутої в попередньому прикладі. Транзистори виконують почергове підключення загального виводу кожного з сегментів до землі. Мікроконтролер видає на порт B

інформаційні коди для відображення цифр і перемикає стани виходами PD0-PD3 так, щоб кожна цифра відображалася на заданому сегменті. Оновлення інформації відбувається з частотою 25 Гц, тому користувач не помічає мерехтіння цифр. При необхідності частоту оновлення інформації можна підвищити.

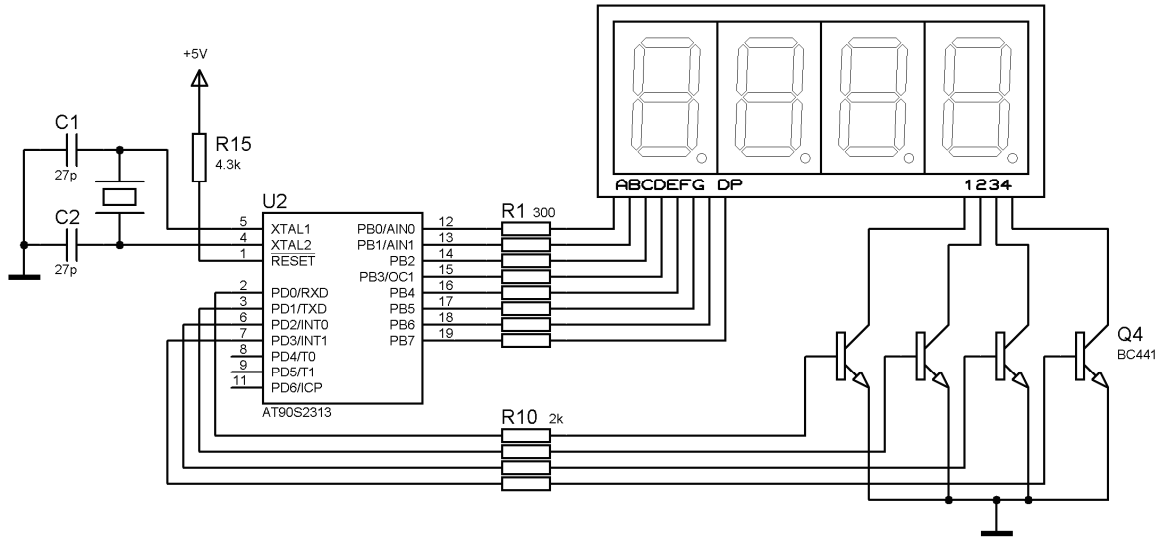


Рис. 2.51.

Текст програми представлено нижче.

```
#include <io2313v.h>                  // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                    // функція часової затримки
{
 int c1, c2;                          // оголошення додаткових змінних
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();      // у циклі виконується «пуста» операція
}
void main()                           // головна функція програми
{
 DDRB=0b11111111;                     // налаштування порту B на вихід
 // налаштування чотирьох молодших виводів порту D на вихід
 DDRD=0b00001111;
 while(1)                             // вічний цикл
 {
  PORTD=1;            // виведення значення 1 в порт D (включення першого індикатора)
  PORTB=0b01011011;                   // вивід на індикатор значення 2
  delay(10);                          // затримка 10 мс
  PORTD=2;             // виведення значення 2 в порт D (включення другого індикатора)
  PORTB=0b01101111;                   // виведення на індикатор значення 9
  delay(10);                          // затримка 10 мс
  PORTD=4;            // виведення значення 4 в порт D (включення третього індикатора)
  PORTB=0b01111111;                   // виведення на індикатор значення 8
  delay(10);                          // затримка 10 мс
```

```
 // виведення значення 8 в порт D (включення четвертого індикатора)
 // виведення на індикатор значення 1
 PORTD=8;
 PORTB=0b00000110;
 delay(10);                          // затримка 10 мс
 }
}
```

У програмі не відбуваються зміни числа, що виводиться на індикатори (2981). Кожна з цифр на індикаторі оновлюється через 40 мс.

На практиці найчастіше клавіатура та індикація об'єднуються в одну загальну схему. Наступний приклад демонструє, яким чином виконується опит стану кнопок і виведення значень на 4-розрядний індикатор. При цьому використовується принцип динамічної індикації.

Приклад 2.10. При натисканні кнопок клавіатури, номер натисненої кнопки виводиться на 7-сегментному індикаторі. У поточній позиції виводу горить десяткова точка. Використовувати принцип динамічної індикації при виведенні значень на індикатор із загальним катодом. На рис. 2.52 представлена принципова схема пристрою.

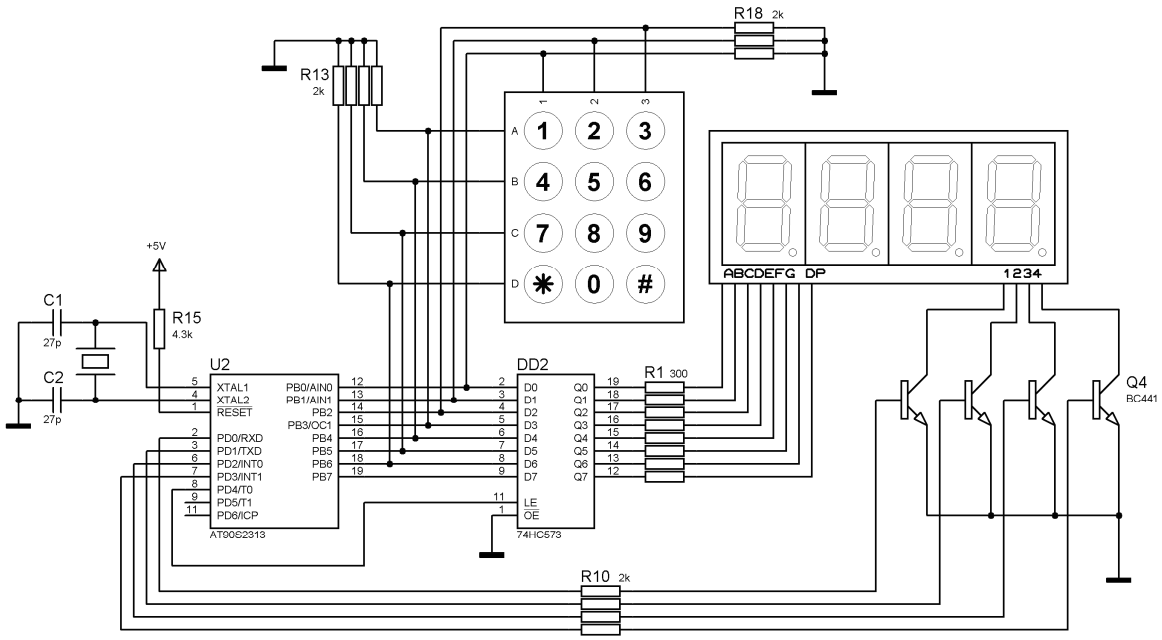


Рис. 2.52.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>                // підключення заголовочних файлів
#include <macros.h>
char prev[4]={0,0,0,0};             // оголошення масиву з 4-х елементів
char index;                         // оголошення додаткової змінної
void delay(int ms)                  // функця затримки
{
 int c1, c2;                        // оголошення додаткових змінних
```

```
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();   // у циклі виконується «пуста» операція
}

char scan_keyboard()               // функція сканування клавіатури
{
 char col=0, row=0;                // додаткові змінні
 char key=13;                      // оголошення змінної з ініціалізацією
 PORTD=0;                          // виведення значення 0 в порт D
 PORTB=0;                          // виведення значення 0 в порт B
 DDRB=0b00000111;                  // настройка виводів PB0-PB2 на вихід
 PORTB=0b00000111;                 // установка на виводах PB0-PB2 сигналу лог.1.
 delay(2);                         // затримка 2 мс
 row=PINB;                         // зчитування значення з порту B
 row=row>>3;                       // зсув значення на 3 розряди праворуч
 row=row&0x0F;                     // виділення 4-х молодших біт
 DDRB=0b01111000;                  // налаштування виводів PB6-PB3 на вихід
 PORTB=0b01111000;                 // установка на виводах PB6-PB3 сигналу лог.1.
 delay(2);                         // затримка 2 мс
 col=PINB;                         // зчитування значення з порту B
 col=col&0x07;                     // виділення 3-х молодших біт
 if(row==1) key=1;           // визначення ряду, в якому знаходиться натиснута кнопка
 else if(row==2) key=4;
 else if(row==4) key=7;
 else if(row==8) key=10;
 if(col==2) key=key+1;// визначення колонки, в якій знаходиться натиснута клавіша
 else if(col==4) key=key+2;
 PORTB=0;                          // виведення значення 0 в порт B
 DDRB=0xFF;                        // налаштування порту B на вихід
 PORTD|=0x10;                      // установка сигналу лог.1 на виводі PD4
 return key-1;                     // повернення номеру натиснутої кнопки
}

void main()                        // головна функція програми
{
 char c=0;                         // оголошення додаткових змінних
 char button;
 // оголошення масиву з кодами натиснутих клавіш (0-9, A, B).
 char seg[12]={0b00000110, 0b01011011, 0b01001111, 0b01100110,
0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111, 0b01110111,
0b00111111, 0b01111100};
 index=0;                          // онулення змінної
 DDRB=0b11111111;                  // налаштування порта B на вихід
 DDRD=0b00011111;            // налаштування 5 молодших виводів порту D на вихід
 PORTD=0;                          // виведення значення 0 в порт D
 while(1)                          // вічний цикл
 {
  if(c>50)                         // якщо значення змінної з більш 50
  {
   c=0;                            // онулення змінної
   button=scan_keyboard();         // сканування клавіатури
```

```
        if(button<12)                  // якщо натиснута яка-небудь клавіша
   {
    prev[index]=button;                // запис в масив номеру натиснутої кнопки
          if(index<3) index++;         // якщо значення змунної менше 3 – інкремент
          else index=0;                // інакше онулення змінної
   }
  }
  else c++;                            // збільшення значення змінної на 1
  PORTD|=1;                            // установка на виході PD0 значення лог.1.
  PORTB=seg[prev[0]];                  // виведення в порт B елемента масиву
  if(index==0) PORTB|=0x80;            // якщо значення індекса дорівнює 0, PB7=1
  else PORTB&=~0x80;                   // інакше PB7=0
  delay(5);                            // затримка 5 мс
  PORTD&=~1;                           // установка на виводі PD0 низького рівня сигналу
  PORTD|=2;                            // установка на виводі PD1 високого рівня сигналу
  PORTB=seg[prev[1]];                  // виведення в порт B елемента масиву
  if(index==1) PORTB|=0x80;            // якщо значення індекса дорівнює 1, PB7=1
  else PORTB&=~0x80;                   // інакше PB7=0
  delay(5);                            // затримка 5 мс
  PORTD&=~2;                           // установка на виводі PD1 низького рівня сигналу
  PORTD|=4;                            // установка на виводі PD2 високого рівня сигналу
  PORTB=seg[prev[2]];                  // виведення в порт B елемента масиву
  if(index==2) PORTB|=0x80;            // якщо значення індекса дорівнює 1, PB7=1
  else PORTB&=~0x80;                   // інакше PB7=0
  delay(5);                            // затримка 5 мс
  PORTD&=~4;                           // установка на виводі PD2 низького рівня сигналу
  PORTD|=8;                            // установка на виводі PD3 високого рівня сигналу
  PORTB=seg[prev[3]];                  // виведення в порт B елемента масиву
  if(index==3) PORTB|=0x80;            // якщо значення індекса дорівнює 1, PB7=1
  else PORTB&=~0x80;                   // інакше PB7=0
  delay(5);                            // затримка 5 мс
  PORTD&=~8;                           // установка на виводі PD3 низького рівня сигналу
 }
}
```

Опитування клавіатури (функція scan_keyboard()) відбувається в проміжок часу між змінами розрядів в процесі динамічної індикації. У масив prev[] записуються коди натиснутих кнопок. При записі чергового значення змінна index збільшується на 1. Таким чином, при кожному новому натисканні на кнопку елементи масиву prev[] оновлюються і виводяться на чотирирозрядний індикатор. У програмі відстежується значення змінної index і встановлюється або відключається старший біт порту B, який управляє станом десяткової точки в кожному розряді.

2.7. Матричні індикатори

При необхідності індикації великої кількості станів на обмеженій площі лицьової панелі одного приладу, або при створенні інформаційних табло, загальний об'єм устаткування для управління індикаторами може бути значно знижений за рахунок використання їх матричного вмикання.

На рис. 2.53 показані конфігурації інформаційних полів матричних індикаторів, а на рис. 2.54 – зовнішній вигляд матричних індикаторів.

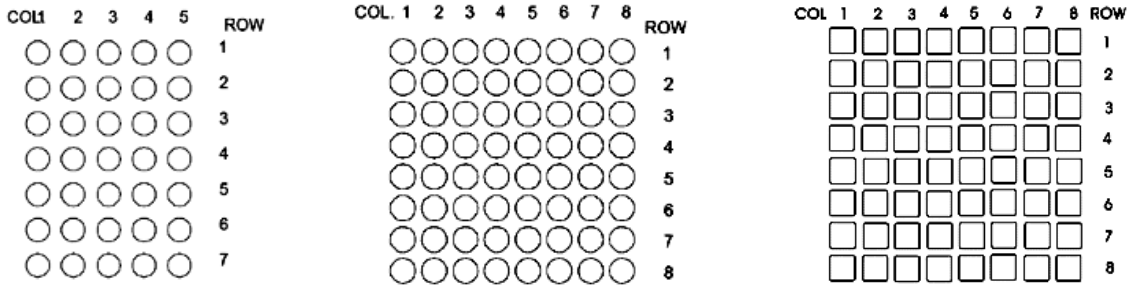
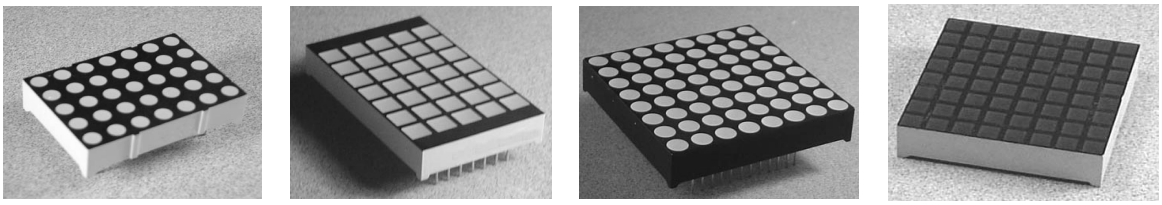


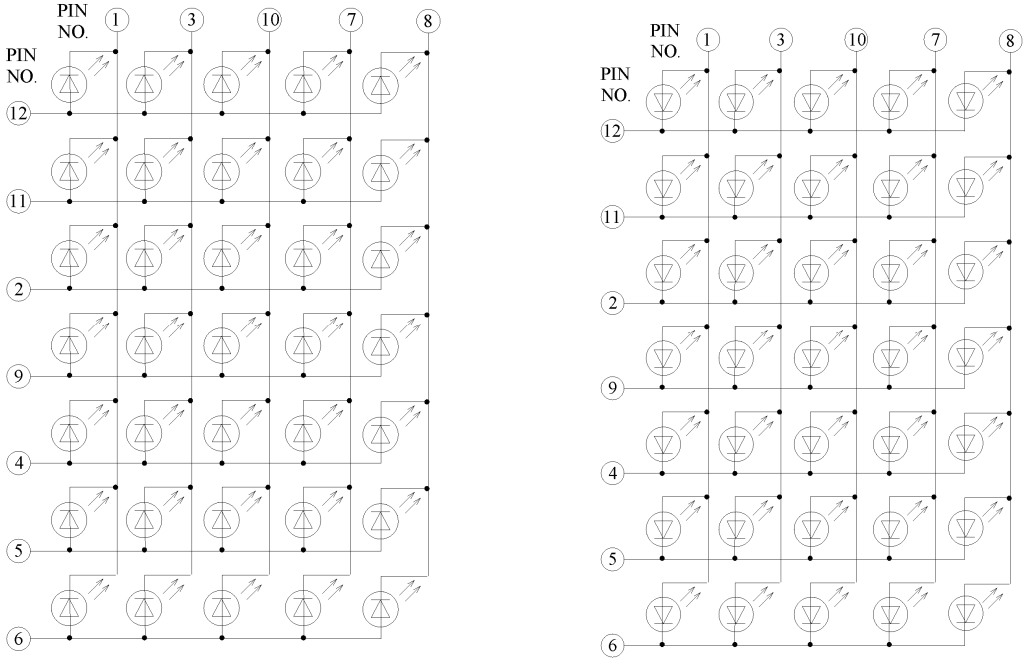
Рис. 2.53.



FYM-7571 FYM-12591 FYM-12882 FYM-12883

Рис. 2.54.

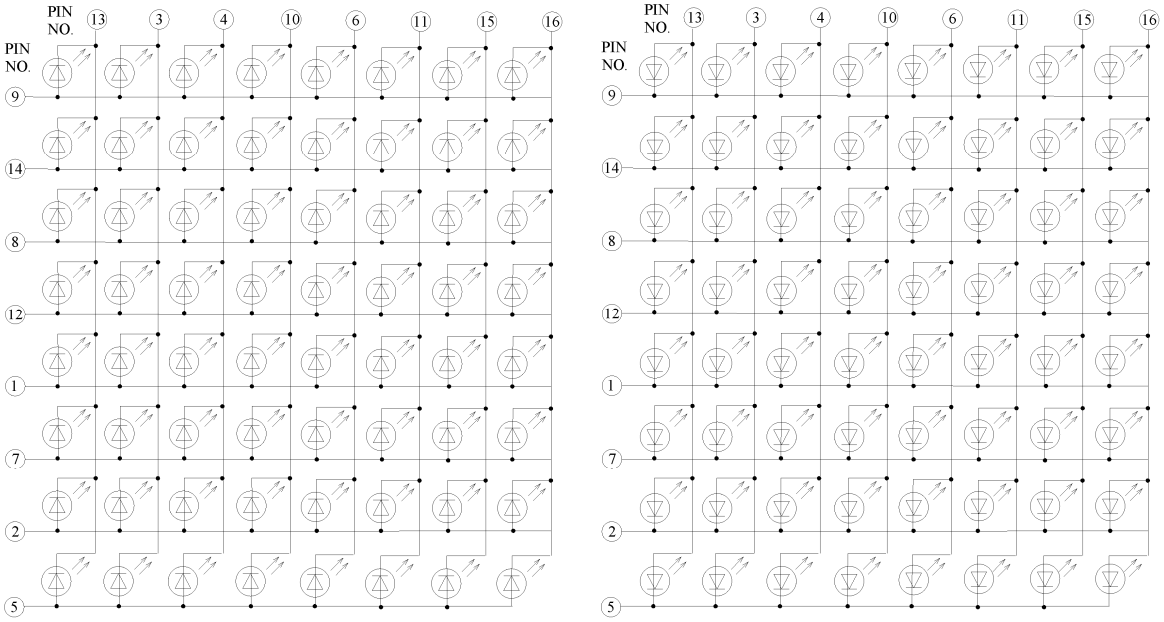
На рис. 2.55 і рис. 2.56 показано два основні режими управління індикаторами, об'єднаних в матрицю, в якій аноди одиничних індикаторів об'єднані по горизонталі (в рядки), а катоди – по вертикалі (у стовпці) (а), і навпаки, аноди одиничних індикаторів об'єднані по вертикалі (у стовпці), а катоди – по горизонталі (в рядки) (б).



а *б*

Рис. 2.55.

Найбільш простий режим управління – управління одиничним індикатором постійним струмом. Такий режим дозволяє забезпечити позиційне управління, тобто таке управління, при якому із всіх елементів матриці може бути включений лише один. При подачі управляючих сигналів на вибрані клеми по горизонталі та по вертикалі може бути включений лише один індикатор. При необхідності включення іншого індикатора раніше збуджений елемент матриці має бути погашений.



а *б*

Рис. 2.56.

Наступний приклад демонструє принципи управління одиничним індикатором в межах матриці.

Приклад 2.11. Розробити схему і написати програму для мікроконтролера, яка реалізує наступний алгоритм: при натисканні на кнопки «U», «D», «L», «R» в межах інформаційного поля матричного індикатора переміщується засвічений елемент.

На рис. 2.57 представлена принципова схема пристрою.

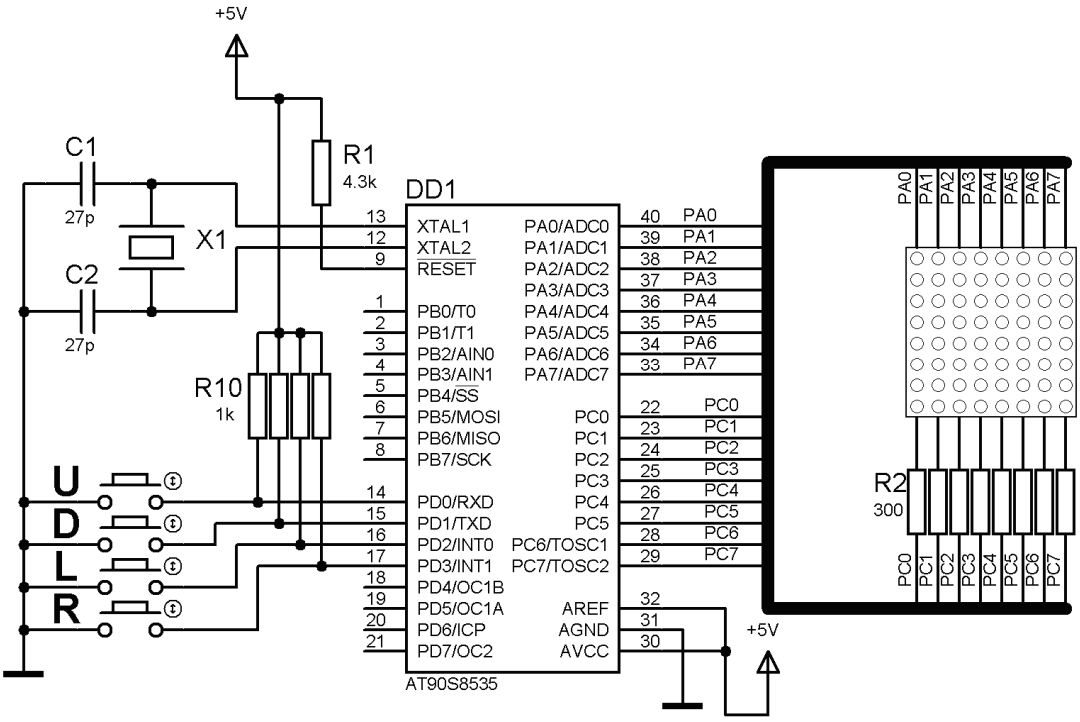


Рис. 2.57.

Оскільки виконується управління лише одним елементом матриці (світлодіодом), тому в схемі не використані підсилювальні елементи (транзистори) з огляду на те, що виводи мікроконтролера здатні сформувати необхідний струм. Виводи мікроконтролера, до яких приєднані кнопки, підтягнуті через резистори до шини живлення для уникнення появи на них невизначеного рівня. При натисканні на кнопку, на відповідному вході з'явиться сигнал низького рівня, що відстежується програмою.

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>                  // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                    // функція часової затримки
{
 int c1, c2;                          // оголошення додаткових змінних
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();      // у циклі виконується «пуста» операція
}
void main()                           // головна функція програми
{
 char x=0,y=0;                        // змінні для зберігання координат
 DDRA=0xFF;                           // налаштування порту A на вихід
 DDRB=0;                              // порт B налаштований на вхід
 DDRC=0xFF;                           // налаштування порту C на вихід
 DDRD=0;                              // порт D налаштований на вхід
 while(1)                             // вічний цикл
 {
```

```
  PORTA=~(1<<y);                  // виведення 0 в позицію з координатою «y»
  PORTC=(1<<x);                   // виведення 1 в позицію з координатою «x»
  delay(200);                     // затримка 200 мс
  if(((PIND&0x01)==0) && (y>0)) y--;     // при натисканні на кнопку «U» декремент
  if(((PIND&0x02)==0) && (y<7)) y++;     // при натисканн на кнопку «D» інкремент
  if(((PIND&0x04)==0) && (x>0)) x--;     // при натисканн на кнопку «L» декремент
  if(((PIND&0x08)==0) && (x<7)) x++;     // при натисканні на кнопку «R» інкремент
}}
```

Змінні *x* і *y* є координатами засвіченого сегменту матриці і значення цих змінних змінюються в межах від 0 до 7. Їх зміна (збільшення або зменшення) відбувається при натисканні на відповідні кнопки. Потім в порти A і C виводиться значення 1, зсунуте на *x* і *y* розрядів відповідно.

З метою визначення об'єму устаткування для управління деякою однаковою кількістю одиничних індикаторів при роздільному і матричному управліннях, слід розташувати їх в комплектні групи по *m* рядків і *n* стовпців. При роздільному управлінні буде потрібно *m·n* комутаторів живлення і *m·n* струмообмежувальних резисторів. При матричному з'єднанні буде потрібно *m*+*n* комутаторів і лише *m* або *n* струмообмежувальних резисторів, при скороченні кількості під'єднуючих провідників з *m·n*+1 до *m*+*n*.

Для управління 64 індикаторами приведеної вище схеми (рис. 2.57) потрібно 64 перемикачі живлення і 64 струмообмежувальних резистори. При матричному (8×8) підключенні для управління необхідно шістнадцять перемикачів живлення і вісім струмообмежувальних резисторів при майже чотирикратному скороченні кількості адресних шин. Специфіка режиму управління постійним струмом дозволяє при розрахунку струмообмежувального резистора виходити із значення струму $I_{пр}$, що протікає через один світлодіод. На рис. 2.58 представлена структура двоколірного матричного індикатора, а на рис. 2.59 – повноколірного (RGB).

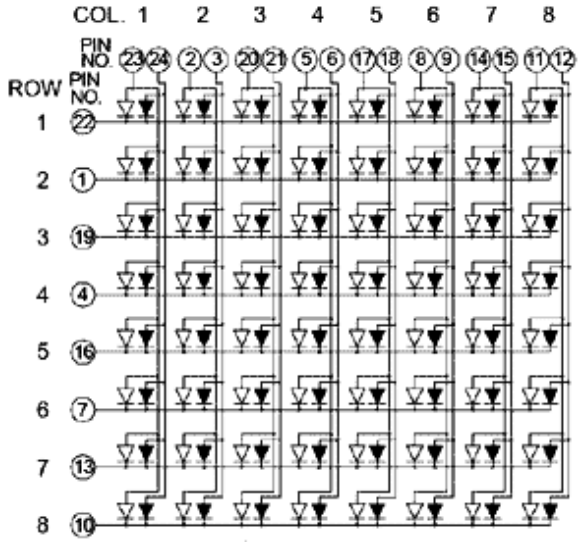


Рис. 2.58.

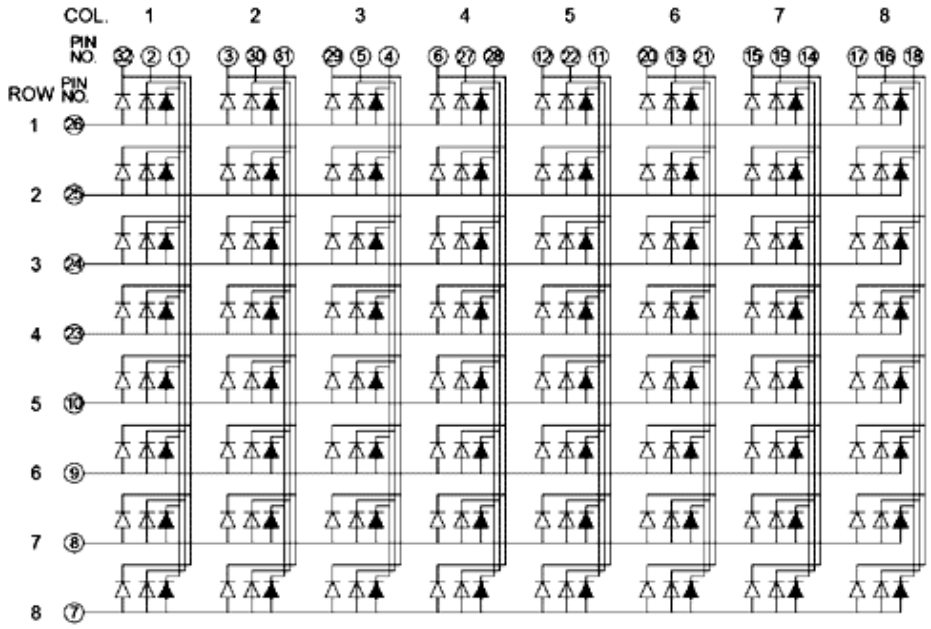


Рис. 2.59.

Ідентичність світіння елементів матриці різних кольорів може бути забезпечена розрахунковим або графічним способом. При графічній побудові за характеристикою яскравості одиничного індикатора по вибраній для всієї матриці яскравості, визначається значення прямого струму, а по вольт-амперній характеристиці визначаються значення струмообмежувальних резисторів.

Матричність структури букво-цифрових індикаторів дозволяє здійснити виведення на індикацію знаку лише в режимі стробування, при цьому кожен символ має бути утворений з п'яти або семи піднаборів залежно від способу адресації (для індикатора 5×7). Сама структура матриці $m \cdot n$ передбачає два способи адресації: стробування по рядках і по стовпцях. На рис. 2.60 представлена поетапность формування букви «Б» способом стробування по рядках.

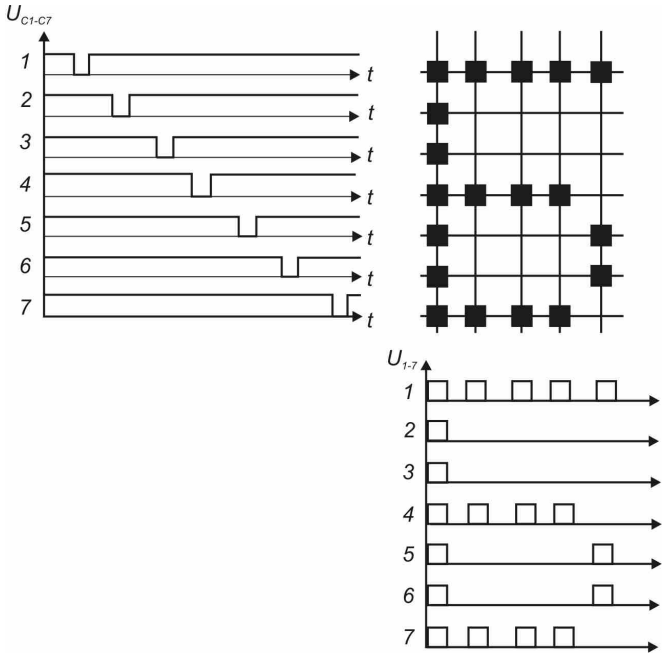


Рис. 2.60.

При стробуванні по рядках інформація на збудження подається по лініях стовпців при подачі дозволяючого строба на відповідний рядок. Цей процес повторюється для кожного рядка. Таким чином, інформація про символ має бути розкладена на сім п'ятирозрядних кодових слова. Інформація повинна подаватися на стовпці індикатора паралельними п'ятирозрядними кодами. Стробування рядків виконується послідовно.

Для розгляду циклів формування знаків на 35-елементній матриці приймемо, що аноди світловипромінюючих діодів об'єднані по стовпцях, а катоди – по рядках, тобто для загоряння елементу матриці (світлодіода) необхідний високий логічний рівень напруги подавати на виводи стовпців, низький логічний рівень – на виводи рядків.

Процес формування символу «Б» способом стробування по рядках протікає таким чином. Інформація U_1 про необхідність загоряння всіх елементів першого рядка (високий логічний рівень напруги) подається на входи всіх стовпців одночасно з сигналом стробування U_{c1} низького логічного рівня на вхід першого рядка. При цьому засвічуються всі світлодіоди першого рядка. Після закінчення часу експонування сигнали U_1 і U_{c1} знімаються. На входи стовпців подаються сигнали для загоряння елементів другого рядка (в даному випадку високий логічний рівень подається на вхід першого стовпця, на входи останніх – низький логічний рівень). При цьому подається на вхід другого рядка стробуючий сигнал низького логічного рівня U_{c2}. Засвічується лише перший світлодіод другого рядка. Загоряння світлодіодів останніх рядків відбувається аналогічним чином. Загоряння кожного рядка з частотою не менше 100 Гц забезпечує відображення символу «Б» без мерехтінь.

При стробуванні по стовпцях інформація на збудження елементу матриці подається по лініях рядків при подачі дозволяючого строба на

відповідний стовпець. Цей процес повторюється для кожного стовпця, тобто інформація має бути розкладена на п'ять семирозрядних кодових слова. Інформація повинна подаватися на рядки індикатора паралельними семирозрядними кодами. Стробування стовпців здійснюється послідовно. На рис. 2.61 представлена поетапність формування букви «Б» способом стробування по стовпцях.

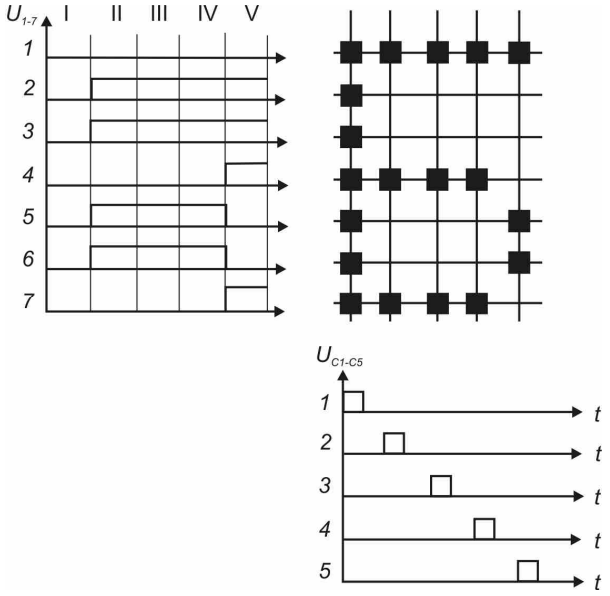


Рис. 2.61.

Формування відбувається таким чином. На вхід всіх рядків одночасно подається інформація $U_{1\text{-}7}$ про необхідність світіння світлодіодів (в разі індикації символу «Б» подаються на всі рядки низькі логічні рівні напруги). Одночасно на вхід першого стовпця подається сигнал стробування U_{c1} (сигнал високого логічного рівня), забезпечуючи світіння всіх семи елементів матриці першого стовпця. Після закінчення часу експонування інформаційні сигнали і сигнали стробування знімаються. У другому циклі роботи на вхід всіх рядків подається інформація про висвічуванняння елементів (в разі символу «Б» – на входи першого, четвертого і сьомого рядків подається сигнал низького логічного рівня), на останні входи – високий логічний рівень. На вхід другого стовпця подається стробуючий сигнал U_{c2}, висвічуючи світлодіоди першого, четвертого і сьомого рядків. Висвічування останніх елементів відбувається аналогічно. При частоті відновлення інформації на кожному з рядків не нижче 100 Гц зображення символу «Б» відображається без мерехтінь. При тривалій роботі оператора з дисплеєм, що працює в мультиплексному режимі, мерехтіння дратує і викликає стомлення ока. Мерехтіння обумовлене здатністю ока нижче за деяку частоту зміни яскравості уловлювати ці зміни. Вище за цю частоту мерехтіння не спостерігається. При нормальній освітленості частота мерехтіння менша

40 Гц непомітна операторові. При високих рівнях яскравості ця частота може бути вища.

Режим стробування забезпечує підключення кожного стовпця на деякий час, обернено пропорційний до кількості стробованих ліній діодів. При цьому падає яскравість світіння індикатора. Для збереження яскравості світіння імпульсний струм через кожен зі світлодіодів необхідно збільшити в кількість разів, що відповідає кількості стробованих ліній.

Наступні приклади демонструють принципи управління матричними індикаторами зі стробуванням по стовпцях і по рядках.

Приклад 2.12. Розробити принципову схему і написати програму для мікроконтролера, яка виводить на матричному індикаторі символ «А». Стробування виконується по стовпцях.

Принципова схема представлена на рис. 2.62. Оскільки максимальний сумарний струм одного стовпця матричного індикатора складає 70 мА, і мікроконтролер на своїх виводах не може сформувати такий струм, в схемі використовуються транзистори. Низький рівень на базі транзистора включає відповідний стовпець. В кожний момент часу відкритий лише один транзистор.

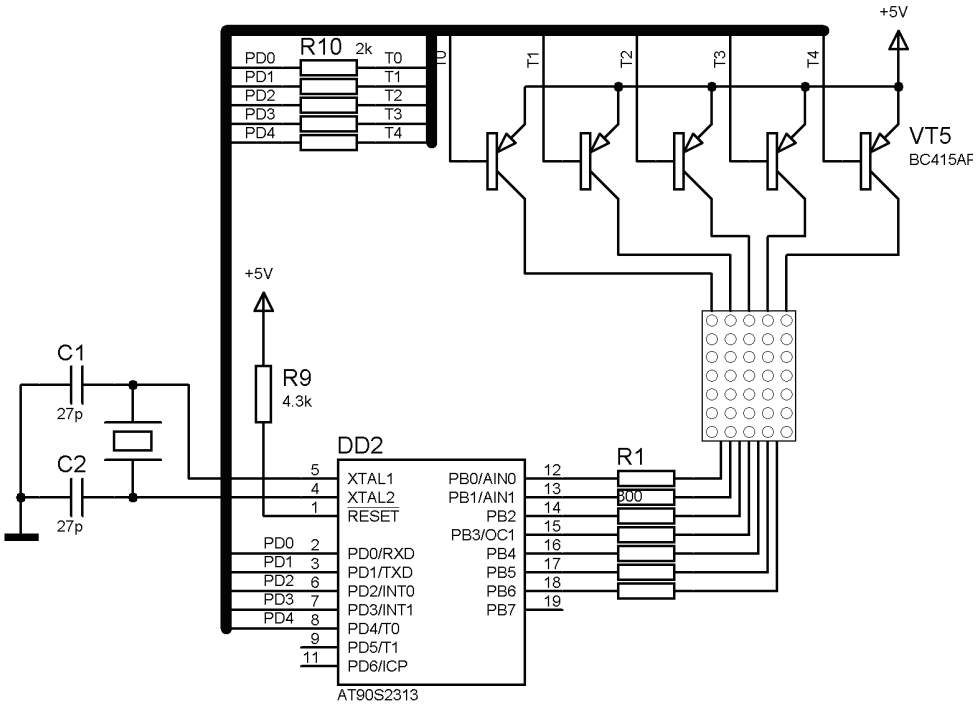


Рис. 2.62.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>                    // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                      // функція затримки
{
 int c1, c2;                            // оголошення додаткових змінних
```

```
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();   // у циклі виконується «пуста» операція
}
void main()                        // головна функція програми
{
 char c=0;                         // оголошення додоткової  змінної
 DDRB=0b01111111;                  // налаштування порту B на вихід
 DDRD=0b00011111;                  // налаштування виводів PD4-PD0 на вихід
 while(1)                          // вічний цикл
 {
 PORTD=0b00011110;
 PORTB=~0b01111100;
 delay(4);                         // затримка 4 мс
 PORTD=0b00011101;
 PORTB=~0b00010010;
 delay(4);                         // затримка 4 мс
 PORTD=0b00011011;
 PORTB=~0b00010001;
 delay(4);                         // затримка 4 мс
 PORTD=0b00010111;
 PORTB=~0b00010010;
 delay(4);                         // затримка 4 мс
 PORTD=0b00001111;
 PORTB=~0b01111100;
 delay(4);                         // затримка 4 мс
  }
}
```

На рис. 2.63 представлена часова діаграма роботи пристрою.

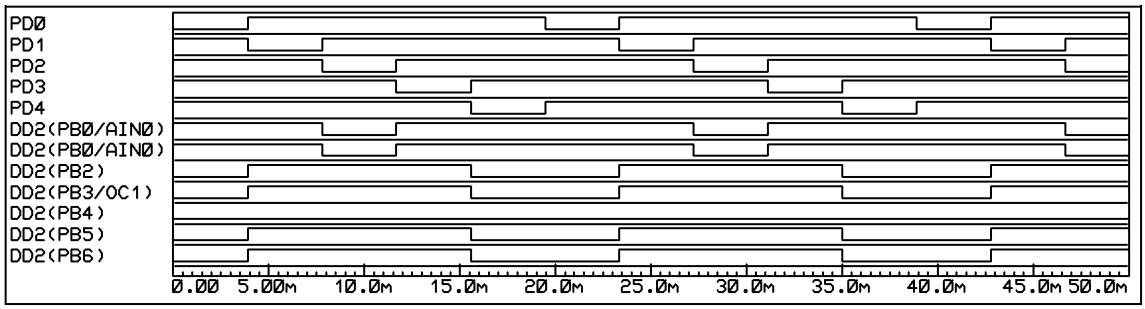


Рис. 2.63.

З часової діаграми видно, що в кожному такті лише на одному виводі порту D присутній низький логічний рівень, який управляє включенням відповідного стовпця. Відповідно до включеного стовпця, на порт B виводиться код, який формує символ, що відображається.

Приклад 2.13. Розробити принципову схему і написати програму для мікроконтролера, яка виводить на матричному індикаторі символ «Б». Стробування виконується по рядках.

Принципова схема представлена на рис. 2.64. Оскільки максимальний сумарний струм одного стовпця матричного індикатора складає 50 мА, і мікроконтролер на своїх виводах не може сформувати такий струм, в схемі використовуються транзистори. Низький рівень на базі транзистора вмикає відповідний рядок. В кожен момент часу відкритий лише один транзистор.

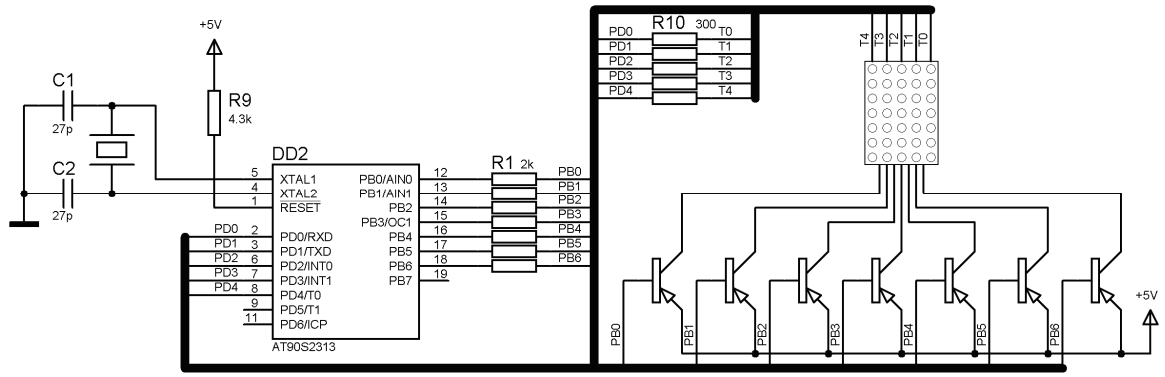


Рис. 2.64.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>                    // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                      // функція затримки
{
 int c1, c2;                            // оголошення додаткових змінних
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();        // в циклі виконується «пуста» операця
}
void main()                             // головна функція програми
{
 char i=0;
 // в масиві зберігаються коди для формування символа
 char symbol[7]={0b00000000, 0b00001111, 0b00001111, 0b00000001,
0b00001110, 0b00001110, 0b00000001};
 DDRB=0b01111111;                       // налаштування порту В на вихід
 PORTB=0b01111111;                      // установка сигналу лог.1 на виводах порту В
 DDRD=0b00011111;                       // налаштування виводів PD4-PD0 на вихід
 while(1)                               // вічний цикл
 {
  for(i=0;i<7;i++)              // у циклі почергово виводяться в порт елементи масиву
  {
   PORTD=symbol[i];
   PORTB=~(1<<i);                       // на виводі порту В встановлюється сигнал лог.1.
   delay(3);                            // затримка 3 мс
  }
 }
```

}

На рис. 2.65 представлена часова діаграма роботи пристрою

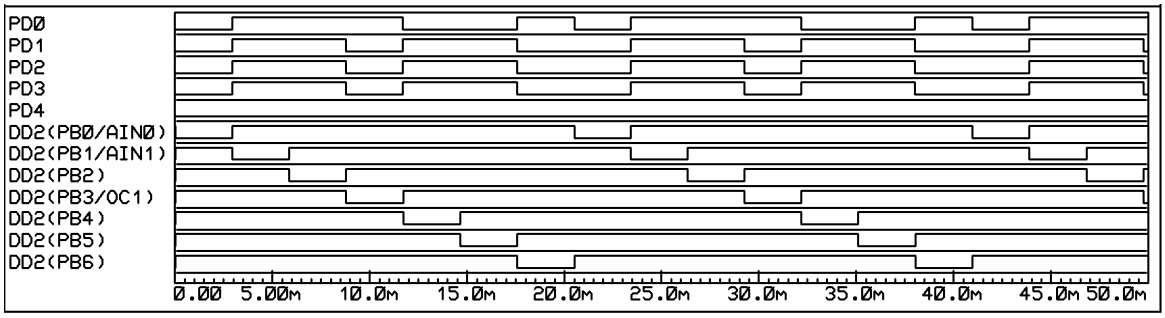


Рис. 2.65.

З часової діаграми видно, що в кожному такті лише на одному виводі порту B присутній низький логічний рівень, який управляє включенням відповідного рядка. Відповідно до включеної позиції, на порт D виводиться код, який формує символ, що відображається.

2.8. Системи адресації

При необхідності одночасної індикації декількох одиничних індикаторів у складі матриці, або відображення букво-цифрової інформації, використовується другий, більш загальний тип управління матрицею – мультиплексний, який дозволяє вмикати будь-яку комбінацію елементів матриці.

При мультиплексному управлінні інформація одночасно подається або на всі *m* рядків при послідовному підключенні стовпців, або одночасно на всі *n* стовпців при послідовному підключенні всіх *m* рядків. Вказані методи індикації отримали відповідні найменування – метод стробування по стовпцях і метод стробування по рядках. В залежності від кількості рядків і стовпців час протікання струму через кожен елемент матриці скорочується в *n* або *m* разів з відповідним зниженням яскравості світіння. Для підвищення яскравості світіння необхідно збільшувати середній струм через кожен одиничний індикатор у складі матриці.

Напівпровідникові індикатори витримують значні пікові струми, що і дозволяє забезпечувати мультиплексне управління матричними індикаторами без погіршення характеристик яскравості.

При частоті відновлення інформації більше 100 Гц, світіння всіх ввімкнених елементів сприйматиметься так, ніби вони управляються постійним струмом.

Приклад 2.14. Розробити принципову схему пристрою відображення інформації, який виводить на двох матричних індикаторах символи «UA». Написати програму для мікроконтролера.

Принципова схема представлена на рис. 2.66.

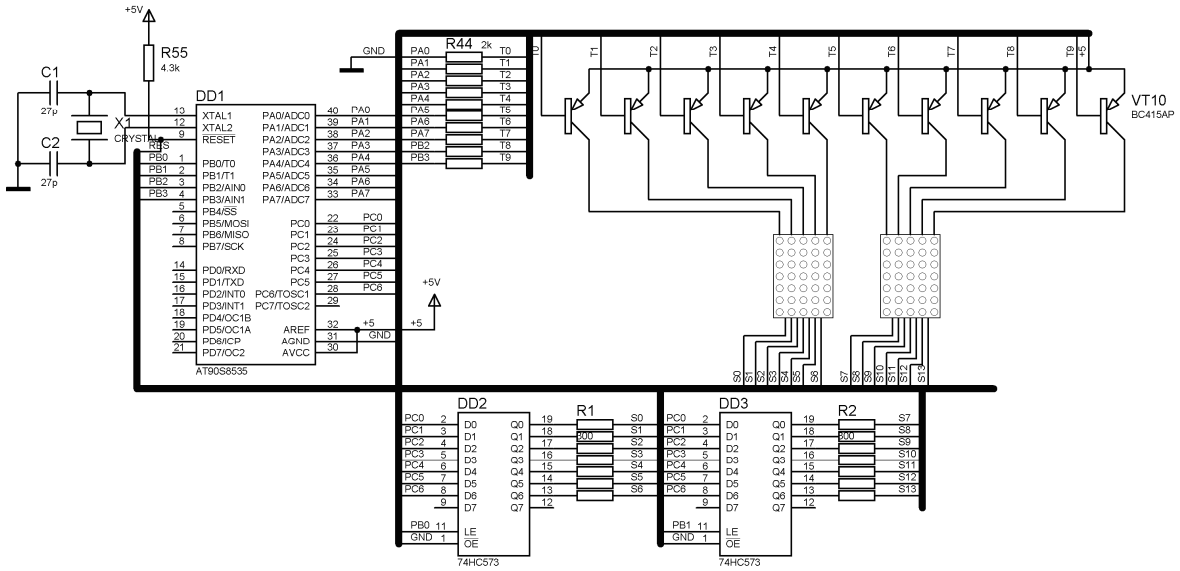


Рис. 2.66.

Текст програми з коментарями представлено нижче.

```
#include <io8535v.h>                // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                  // функція затримки
{
 int c1, c2;                        // оголошення додаткових змінних
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();    // в циклі виконується «пуста» операція
}
void init_cpu()                     // функція ініціалізації мікроконтролера
{
 DDRA=0xFF;                         // налаштування порту A на вихід
 DDRB=0x0F;                         // налаштування 4-х молодших біт порту B на вихід
 DDRC=0x7F;                         // налаштування виводів порту C на вихід
 DDRD=0x7F;                         // налаштування виводів порту D на вихід
}
void main()                         // головна функція програми
{
// в масиві зберігаються коди букв для виведення на індикатори
 char ua[10]={0b00111111, 0b01000000, 0b01000000, 0b01000000, 0b00111111,
 0b01111100, 0b00001010, 0b00001001, 0b00001010, 0b01111100};
 char i;                            // додаткова змінна
 init_cpu();                        // виклик функції ініціалізації мікроконтролера
 while(1)                           // вічний цикл
 {
  PORTB=0b00001101;                 // установка на виводі PB1 лог.0.
  for(i=0;i<5;i++)                  // у циклі виводиться символ «U»
  {
  PORTC=~ua[i];
  PORTA=~(1<<i);
  delay(2);                         // затримка 2 мс
  }
```

```
 PORTB=0b00001110;              // установка на виводі PB0 лог.0
 for(i=5;i<8;i++)               // у циклі виводиться символ «А»
 {
 PORTC=~ua[i];
 PORTA=~(1<<i);
 delay(2);                      // затримка 2 мс
 }
 PORTA=0xFF;                    // налаштування порту А на вихід
 for(i=0;i<2;i++)
 {
  PORTC=~ua[i+8];
  PORTB&=~(1<<(i+2));
  delay(2);                     // затримка 2 мс
  PORTB|=(1<<(i+2));
 }
}
}
```

У масиві ua[10] зберігаються коди символів, що відображуються. Кожен елемент масиву виводиться в порт С синхронно з включенням певного стовпця (низький рівень на виводах PA0-PA7 і PB0-PB4). Зображення на індикаторах оновлюється з частотою, близькою 50 Гц, тому мерехтіння зображення не помітне для спостерігача.

2.9. Шкальні індикатори

Шкальні індикатори мають елементи відображення у вигляді правильних прямокутників і призначені для відображення інформації у вигляді рівнів або значень величин.

Сучасні шкальні індикатори (ШІ) за своїм конструктивним виконанням можна розділити на індикатори в безкорпусному виконані, з полімерною герметизацією без світлопровідника, зі світлопровідником, в герметичних склокерамічних корпусах.

Безкорпусні ШІ можна вмонтовувати у великі інформаційні масиви, здійснювати різні принципи організації зчитування інформації. Конструктивно всі вони виконані на одному кристалі, на якому сформовані випромінюючі елементи з розміром поверхні від 30×30 до 50×50 мкм. На кристалі є контактні місця, необхідні для з'єднання ШІ зі схемою.

Шкальні індикатори зі світлопровідником – найчисленніший вид ШІ, що швидко розвивається. Для збільшення контрастності в ряді шкал, лицьова сторона світлопровідника забарвлюється в чорний колір. Для зменшення взаємного засвічування сегментів бічні поверхні його покриваються нікелем. Металізація зменшує взаємне засвічування, але разом з тим трохи зменшує силу світла.

Всі елементи приладу (тримач, кристал, з'єднання і кришка зі світлопровідником) герметизуються в нероз'ємну монолітну конструкцію оптично прозорим розсіюючим компаундом. Для збільшення розсіювання

світла ряд конструкцій лицьової сторони приладу покритий циліндричними мікролінзами. Індикатори випускаються різних кольорів світіння. Всі ШІ цього вигляду дозволяють здійснювати безшовне з'єднання.

На рис. 2.67 представлені види інформаційних полів шкальних індикаторів, а на рис. 2.68 – прямокутна світодіодна лінійка: а –двоколірна GNA-R102510 BEG; б – одноколірна GNA-R102510 Zx

.

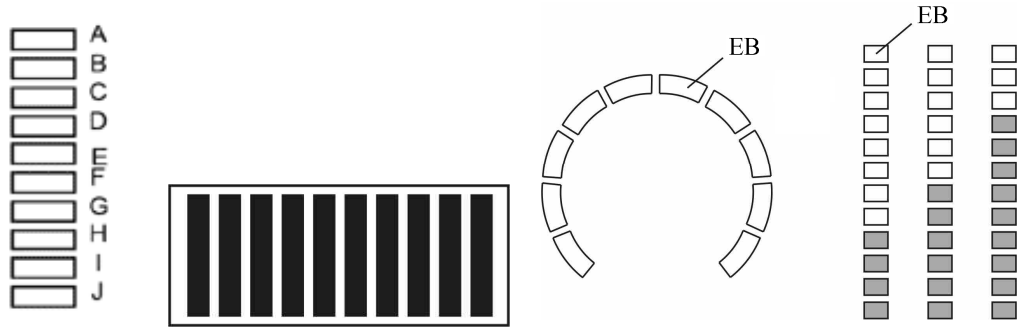


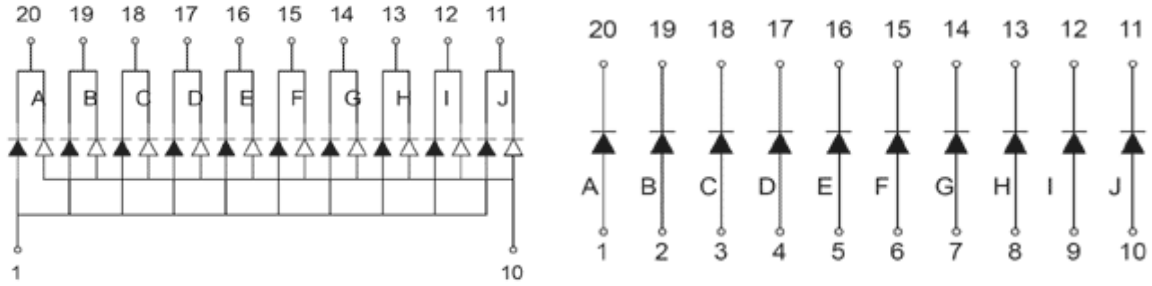
Рис. 2.67.

Лінійні шкальні індикатори призначені для відображення в аналоговій формі безперервно змінної інформації. Аналогова форма представлення інформації на шкальних індикаторах має ряд переваг:

– можливість візуального спостереження тенденцій зміни величини, що спостерігається;

– висока наочність відносних змін параметрів, виходу їх значень за межі допусків (при використанні багатоколірних шкальних індикаторів);

– забезпечення сприйняття оператором великої кількості інформації при найменшому рівні стомлюваності.



а *б*

Рис. 2.68.

Приклад 2.15. Розробити пристрій відображення рівня напруги на шкальному індикаторі. При обертанні ручки змінного резистора на світлодіодах відображається стовпчик, що світиться, висота якого пропорційна напрузі на вході PA0 (канал 0 аналого-цифрового перетворювача).

На рис. 2.69 представлена принципова схема пристрою.

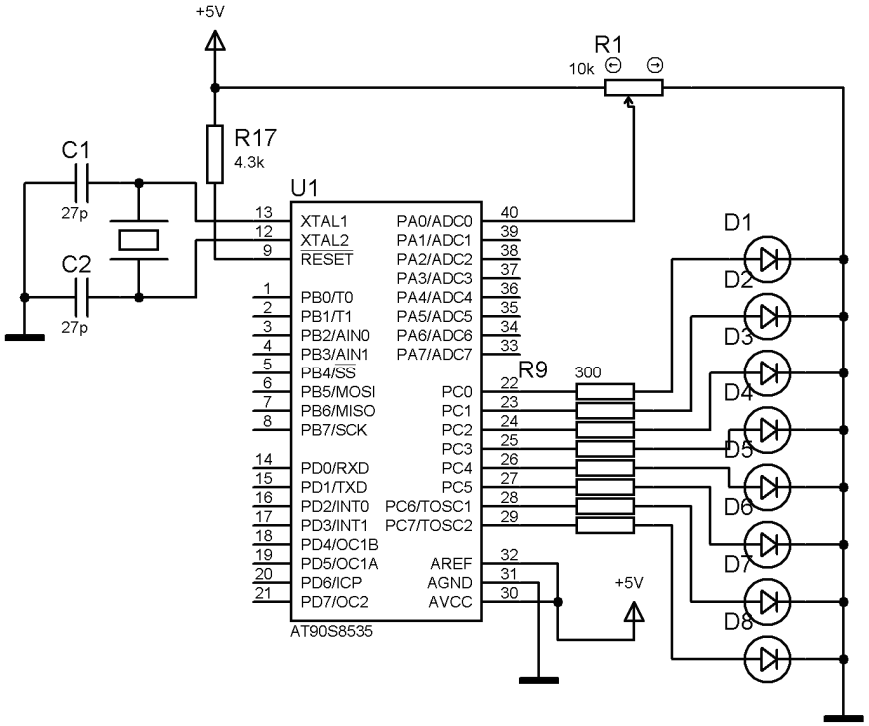


Рис. 2.69.

Текст програми для мікроконтролера представлений нижче.

```
#include <io8535v.h>                // підключення заголовочних файлів
#include <macros.h>
void init_cpu()                     // функція ініціалізації мікроконтролера
{
DDRA=0;                             // порт А налаштований на вхід
DDRC=0xFF;                          // порт С налаштований на вихід
ADCSR=0b10000110;                   // налаштування АЦП
}
unsigned int read_adc(char mux)     // функція перетворення АЦП
{
char dh,dl;                         // додаткові змінні
int res;
ADMUX=mux;                          // вибір каналу АЦП
ADCSR|=(1<<ADSC);                   // початок перетворення
while(!(ADCSR&(1<<ADIF)));          // очікування завершення перетворення
dl=ADCL;                            // зчитування отриманих значень в змінні
dh=ADCH;
res=(dh<<8)+dl;                     // формування 10-бітного результату
return res;
}

void delay(int ms)                  // функція затримки
{
        int C1,C2;                  // додаткові змінні
```

```
        for(C1=0;C1<ms;C1++)
        for(C2=0;C2<970;C2++) NOP();
}
void main()                          // головна функція програми
{
char arr[9]={0x00,0x01,0x03,0x07,0x0F,0x1F,0x3F,0x7F,0xFF}; // масив з кодами
int d;                               // додаткова змінна
init_cpu();                          // виклик функції ініціалізації
while(1)                             // вічний цикл
{
d=read_adc(0);                       // виклик функції перетворення АЦП
PORTC=arr[d/115];                    // виведення значення в порт
delay(100);                          // затримка 100 мс
}
}
```

У масиві зберігаються значення, при виведенні яких в порт С буде запалюватись кількість світлодіодів, відповідних елементу масиву, що виводиться в порт. Функція read_adc(0) виконує аналого-цифрове перетворення і повертає значення, пропорційне величині напруги на вході PA0. Далі, це значення масштабується так, щоб при максимальному значенні напруги на вході АЦП, рівній 5 В, результат не перевищував значення 8 – максимальний індекс в масиві.

Опори в схемі, що обмежують струм через світлодіоди, можуть бути визначені з наступного співвідношення:

$$R = \frac{U_{вих} - U_{сд}}{I_{пр}},$$

де $U_{вих}$ – вихідна напруга на виводі мікроконтролера у включеному стані (5В), $U_{сд}$ – пряме падіння напруги на світлодіоді (2В) при постійному струмі $I_{пр}$, $I_{пр}$ – прямой струм через світлодіод (10 мА). В схемі номінали резисторів в ланцюгу світлодіодів при вказаних значеннях напруги і струму складають 300 Ом.

Максимально допустимий вихідний струм кожного виводу мікроконтролера не повинен перевищувати 20 мА, за умови, що сумарний струм одного порту не перевищує 80 мА. Таким чином, кожен порт мікроконтролера може управляти 8 світлодіодами, якщо прямий струм кожного з них не перевищує 10 мА, або 4 світлодіодами при прямому струмі 20 мА.

Приклад 2.16. Розробити пристрій, який виконує наступну функцію: при натисканні на верхню кнопку стовпчик збільшується на одне ділення, при натисканні на нижню кнопку – зменшується.

Принципова схема пристрою представлена на рис. 2.70.

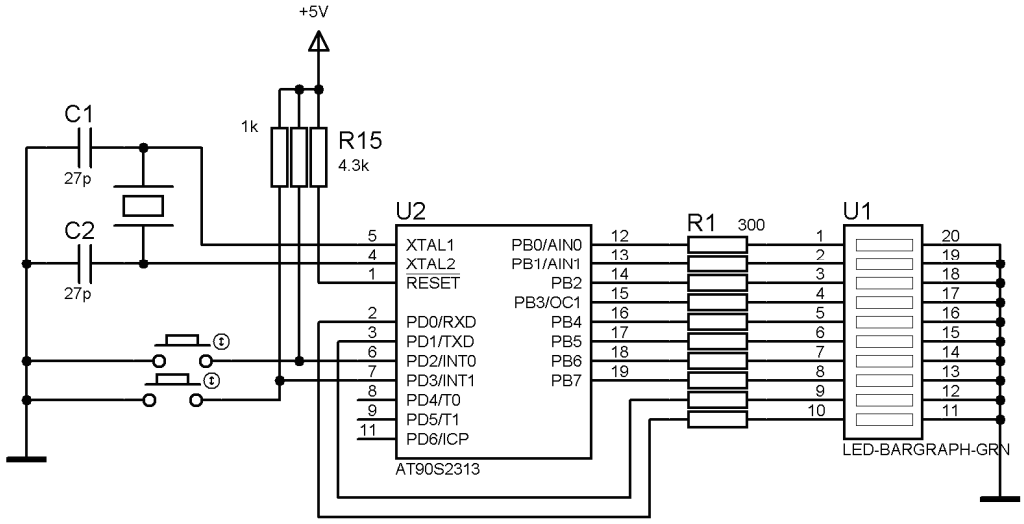


Рис. 2.70.

Текст програми для мікроконтролера представлено нижче.

```
#include<io2313v.h>                  // підключення заголовочних файлів
#include<macros.h>
void init_cpu()                      // функція ініціалізації мікроконтролера
{
 DDRB=0b11111111;                    // налаштування порту B на вихід
 DDRD=0b00000011;                    // налаштування двох виводів порту D на вихід
 }
void delay (int ms)                  // функція затримки
 {
 int c1,c2;
 for (c1=0;c1<ms;c1++)
 for (c2=0;c2<970;c2++)
 NOP();
 }
void main()                          // головна функція програми
 {
 char i=0;                           // додаткова змінна
 int arr[11]={0x0000, 0x0001, 0x0003, 0x8003, 0xC003, 0xE003, 0xF003,
 0xF803, 0xFC03, 0xFE03, 0xFF03}; // масив з кодами для виводу на ШІ
 init_cpu();                         // виклик функції ініціалізації мікроконтролера
 while(1)                            // вічний цикл
 {
        if(((PIND&0x04)==0) && (i<10))   // якщо натиснута верхня кнопка
        {
         i++;                        // інкремент змінної
         delay(20);                  // затримка 20 мс
        }
        else if(((PIND&0x08)==0) && (i>0))        // якщо натиснута нижня кнопка
        {
         i--;                        // декремент змінної
         delay(20);                  // затримка 20 мс
        }
```

```
  PORTD=(char)arr[i];              // виведення в порт D елементу масива
  PORTB=arr[i]>>8;                 // виведення в порт B елементу масива
  delay(100);                      // затримка 100 мс
}
}
```

Приклад 2.17. Розробити пристрій, в якому при натисканні на верхню кнопку стовпчик збільшується на одне ділення, при натисканні на нижню кнопку – зменшується. Виконати нарощування шкальних індикаторів.

Принципова схема пристрою представлена на рис. 2.71. Оскільки в схемі використовуються два шкальні індикатори, в кожному з яких по 10 світлодіодів, для управління цими індикаторами було б потрібно 20 виводів мікроконтролера. Проте, використовуючи регістри зсуву можливе скорочення кількості необхідних виводів до 2. Передача коду, що відображується, виконується в послідовному форматі. Протягом 20 тактів відбувається заповнення всіх регістрів зсуву. Оскільки розрядність регістрів зсуву дорівнює 8, то для управління індикаторами було потрібно три регістри. Вихід попереднього регістра зсуву з'єднується з входом наступного, таким чином всі регістри сполучені послідовно і представляють собою один 24-х розрядний регістр зсуву. Виходи регістрів через струмообмежувальні опори з'єднуються з анодами сегментів шкальних індикаторів.

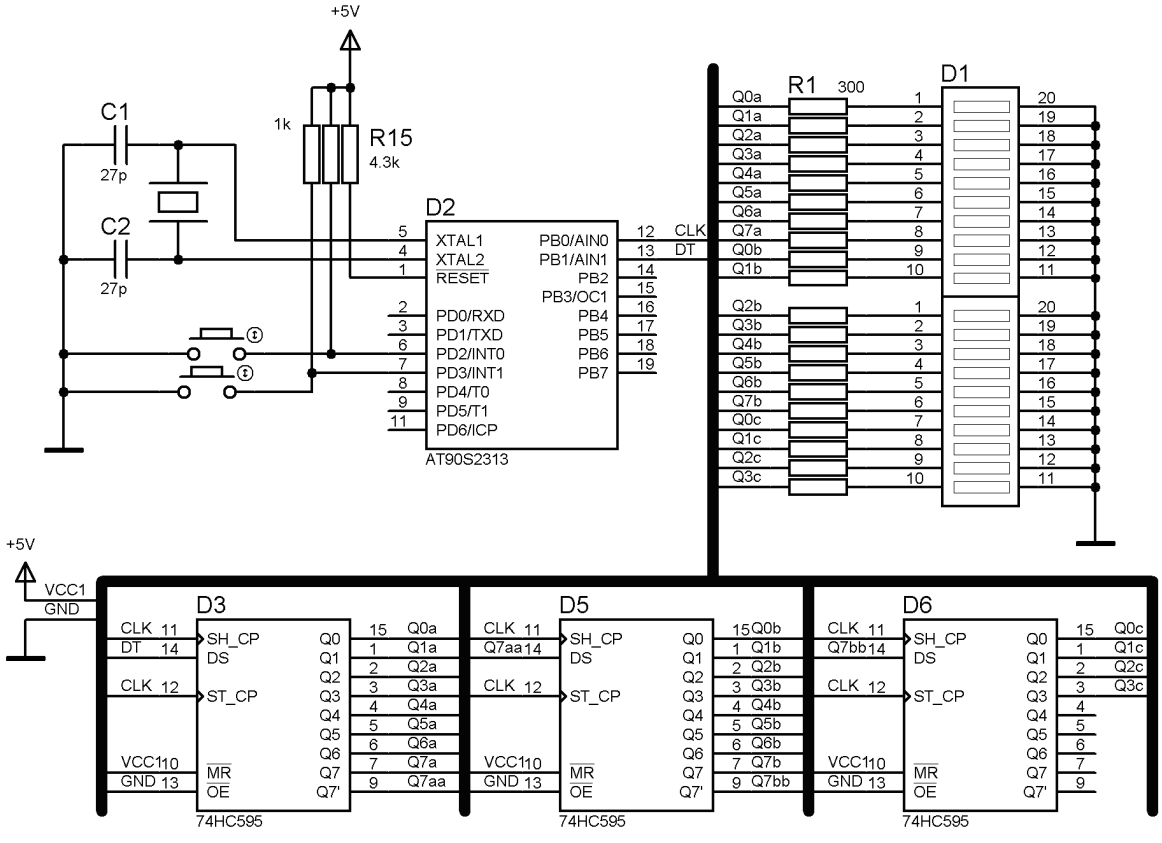


Рис. 2.71.

Текст програми для мікроконтролера представлено нижче.

```
#include<io2313v.h>                  // підключення заголовочних файлів
#include<macros.h>
void init_cpu()                      // функція ініціалізації мікроконтролера
 {
 DDRB=0b00000011;                    // виводи PB0 і PB1 налаштовані на вихід
 DDRD=0;                             // порт D налаштований на вхід
 }

void delay (int ms)                  // функція затримки
 {
 int c1,c2;
 for (c1=0;c1<ms;c1++)
 for (c2=0;c2<970;c2++)
 NOP();
 }

void main()                          // головна функція програми
 {
 char i=0, j;                        // оголошення додаткових змінних
 init_cpu();                         // виклик функції ініціалізації
 while(1)                            // вічний цикл
 {
        if(((PIND&0x04)==0) && (i<20)) // якщо натиснута верхня кнопка
        {
         i++;                        // інкремент значення змінної
         delay(20);                  // затримка 20 мс
        }
        else if(((PIND&0x08)==0) && (i>0)) // якщо натиснута нижня кнопка
        {
         i--;                        // декремент значення змінної
         delay(20);                  // затримка 20 мс
        }
         for(j=0;j<=20;j++)
         {
          PORTB&=~0x02;              // скидання сигналу на виводі PB1
          PORTB|=0x01;               // установка сигналу високого рівня на виводі PB0
          PORTB=0;                   // виведення значення 0 в порт B
         }
        for(j=0; j<i; j++)
        {
          PORTB|=0x02;               // установка сигналу високого рівня на виводі PB1
          PORTB|=0x01;               // установка сигналу високого рівня на виводі PB0
          PORTB=0;                   // виведення значення 0 в порт B
        }
        for(j=0; j<=20-i; j++)
        {
          PORTB&=~0x02;              // скидання сигналу на виводі PB1
          PORTB|=0x01;               // установка сигналу високого рівня на виводі PB0
```

```
        PORTB=0;                    // виведення значення 0 в порт B
      }
      delay(100);                   // затримка 100 мс
      }
}
```

Оновлення даних, що виводяться на шкальні індикатори, відбувається з високою швидкістю і користувач цього не помічає. Тому на індикаторах відображується стовпчик без мерехтіння.

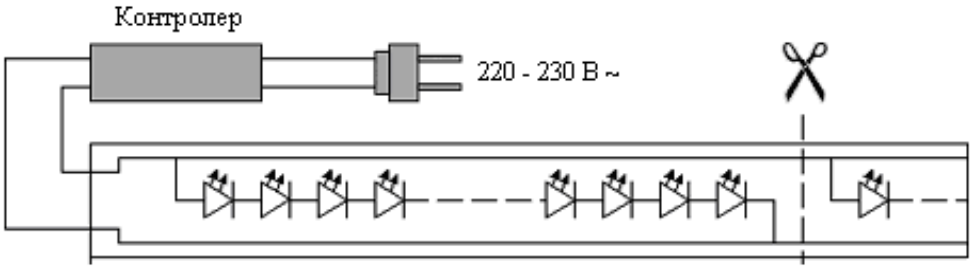
2.10. Світлодіодні дюралайти

Світлодіодні трубки (дюралайти), зовнішній вигляд яких показаний на рис. 2.72, призначені для заміни неонових ламп в зовнішніх і внутрішніх світлових конструкціях. Відрізняються від неонових трубок низьким енергоспоживанням, великим терміном служби і можливістю повторного використання після демонтажу конструкції.

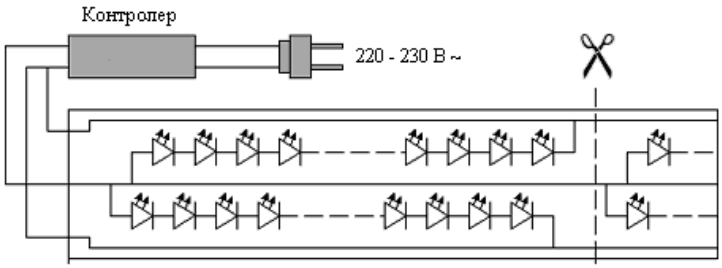
Світлодіодні трубки виготовляються з білого дифузного, забарвленого в колір світіння або прозорого пластика D-подібного перерізу. Елементом монохромних трубок (GN-30STB24-48xx), що світиться, є 48 світлодіодів підвищеної яскравості. Повноколірний варіант (GN-30STB24-114RGB) виконаний з 114 світлодіодів.



Рис. 2.72.



а



б

Рис. 2.73.

Будова дюралайта представлена на рис. 2.73: а – 36 світлодіодів на метр; б – 72 світлодіоди на метр. Споживана потужність дюралайтів складає 2 Вт/метр (EMR-MC 2Вт-100М-230В). Довжина може складати 100 м. Відстань між світлодіодами в трубці порядку 28 мм. Дюралайти випускаються з різними кольорами світіння: червоний, жовтий, зелений, синій, білий.

2.11. Світлодіодні лампи

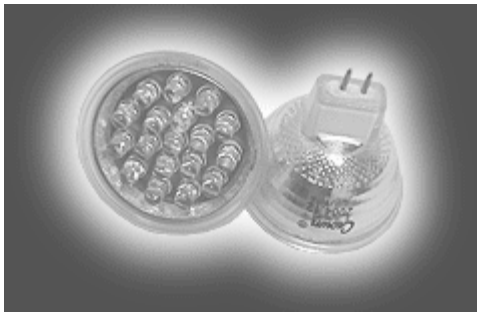
На відміну від ламп розжарювання, світлодіодні лампи відрізняються меншим споживанням електроенергії, високими показниками надійності, високою мірою ергономічності. На рис. 2.74 представлений зовнішній вигляд деяких світлодіодних ламп. Лампи повністю сумісні з патронами для кріплення ламп розжарювання, а також можуть мати роз'єм у вигляді вилки. Світлодіодні лампи випускаються різних кольорів світіння і можуть використовуватися як для освітлення приміщень, так і в декоративних цілях.



GNL-MR16



GNL-PAR-38



SAIW18BAT-H



SAIW20BMT-H

Рис. 2.74.

Світлодіодна лампа GNL-MR16 має наступні характеристики:

- діаметр 50 мм, висота 38 мм;
- 21 світлодіод;
- час роботи – 50 000 годин;
- напруга – 12 В AC/DC, 24 В AC/DC;
- діапазон робочих температур –20...+40°С;
- температура зберігання –40...+85°С;
- енергозберігаюча технологія;
- 6 кольорів + інфрачервоні;
- корпус – пластмаса.

В табл. 2.7 представлені основні характеристики світлодіодних ламп.

Табл. 2.7.

Колір	білий
Робоча температура	–20...+40°С
Температура зберігання	–40...+85°С
Робоча частота	50/60 Гц
Кут огляду	50±5°
Напруга	АС 240 В
Вхідний струм	23±4 мА
Вихідна потужність	1,6±0,3 Вт
Світловий потік	30±6 Лм

Більше десяти років ведуться роботи по заміні ламп розжарювання на світлодіоди в панелях контрольно-вимірювальних приладів і вимикачах, розташованих в кабіні автомобіля. Найбільш обгрунтованим напрямом заміни ламп розжарювання на світлодіодні лампи є виготовлення всіх зовнішніх червоних і жовтих ліхтарів автомобіля. Спеціальні дослідження підтверджують, що використання світлодіодних ламп в повторювачах стоп-сигналу автомобіля – це об'єктивна необхідність. Час включення до повної світлової потужності лампи з ниткою розжарювання складає 200-300 мсек, а в повторювачів стоп-сигналу на світлодіодних лампах 50-70 нс. При русі по автостраді зі швидкістю 100 км/год за рахунок різниці в другому випадку водій отримує додаткові 5-6 метрів для екстреного гальмування або маневру. Враховуючи ці результати з 1996 року на всіх нових автомобілях в Європі повинні встановлюватися повторювачі стоп-сигналу на світлодіодних лампах.

2.12. Способи управління яскравістю індикаторів

Особливістю використання напівпровідникового індикатора, як і будь-якого активного (світловипромінюючого) індикатора, є залежність якості сприйняття інформації від рівня зовнішньої освітленості. Зокрема, в приміщеннях з розсіяним освітленням, індикатор в номінальних режимах роботи дасть дискомфорт зчитування інформації через надзвичайно

високий контраст яскравості; у приміщеннях з високими рівнями зовнішньої освітленості (від 10000 до 100000 Лк) контрасту яскравості для впевненого зчитування інформації навіть при максимумі світловіддачі без вживання спеціальних заходів буде недостатньо. Для пристроїв, що працюють в широкому діапазоні зовнішньої освітленості, необхідно вирішувати обидві ці задачі.

Зниження контрасту при роботі в приміщеннях з низьким рівнем зовнішньої освітленості досягається шляхом регулювання яскравості світіння індикаторів.

2.12.1. Параметричне управління

Регулювання яскравості можливе за рахунок зміни живлення напівпровідникових індикаторів, і відповідно, амплітуди струму, що проходить через світлодіоди. Регулюючим елементом може служити змінний резистор, винесений на лицьову панель приладу (рис. 2.75).

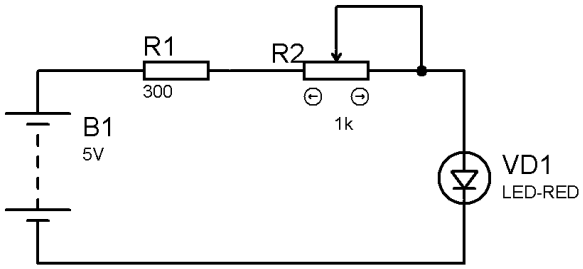


Рис. 2.75.

При малих значеннях струму, що протікає через елемент, спостерігається значний розкид яскравості їх світіння. При зниженні до певного мінімуму струму, що протікає через світлодіоди, розкид яскравості значно збільшується. На рис. 2.76 представлена залежність яскравості світіння світлодіода від прямого струму.

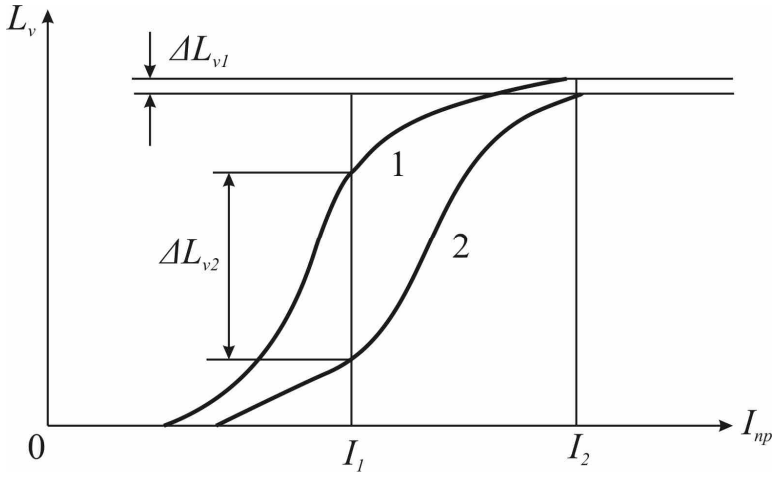


Рис. 2.76.

При невисоких рівнях яскравостей зорове сприйняття нерівномірності світіння сприйматиметься сильніше. Регулювання яскравості світіння індикаторів методом зміни напруги на нижніх межах регулювання може створити дискомфорт при зчитуванні інформації через нерівномірності світіння світлодіодів.

Тому спосіб регулювання яскравості світіння індикаторів зниженням напруги живлення, прийнятний для приладів, що розміщуються в приміщеннях з постійним середнім і високим рівнем зовнішньої освітленості, неприйнятний для пристроїв відображення інформації, що розміщуються в приміщеннях і на об'єктах з широким діапазоном яскравостей зовнішніх освітлень.

2.12.2. Часове управління

Широтно-імпульсний метод (ШІМ) управління яскравістю заснований на скороченні часу протікання струму через світлодіоди індикаторів. При цьому знижується значення середнього прямого струм через світлодіоди, отже, знижується і яскравість їх світіння.

Приклад 2.18. Написати програму управління яскравістю світлодіода. Протягом 1 секунди яскравість світлодіода плавно знижується до мінімуму. На рис. 2.77 представлена принципова схема пристрою. Світлодіод підключений катодом до виводу PB0 мікроконтролера, а анодом – до лінії живлення.

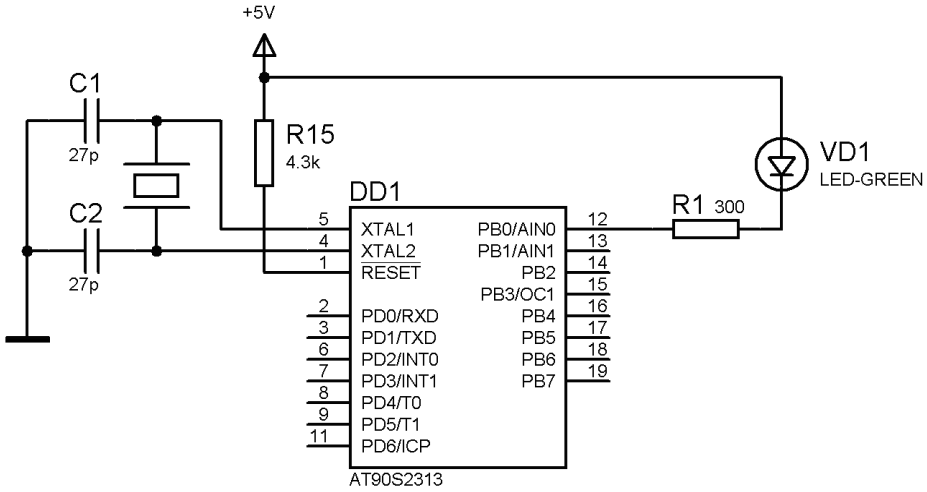


Рис. 2.77.

Текст програми для мікроконтролера представлено нижче.

```
#include <io2313v.h>                  // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                    // функція затримки
{
 int c1, c2;                          // додаткові змінні
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<150; c2++) NOP();      // пуста операція
```

```
}
void main()                          // головна функція програми
{
 char i;                             // оголошення змінних
 char k=10;
 DDRB=0x01;                          // налаштування виводу PB0 на вихід
 while(1)                            // вічний цикл
 {
  for(i=0;i<7;i++)
  {
  PORTB=1;                           // виведення значення 1 в порт B
  delay(k);                          // затримка k мілісекунд
  PORTB=0;                           // виведення значення 0 в порт B
  delay(100-k);                      // затримка 100-k мілісекунд
  }
  if(k<90) k=k+10;    // якщо значення змінної k менше 90, до нього додається 10
  else k=10;          // інакше змінній k присвоюється значення 10
 }
}
```

На рис. 2.78 представлена епюра напруги, що пояснює роботу цієї схеми і алгоритм програми. При зміні скважності імпульсів, зміниться середній струм, що протікає через світлодіод.

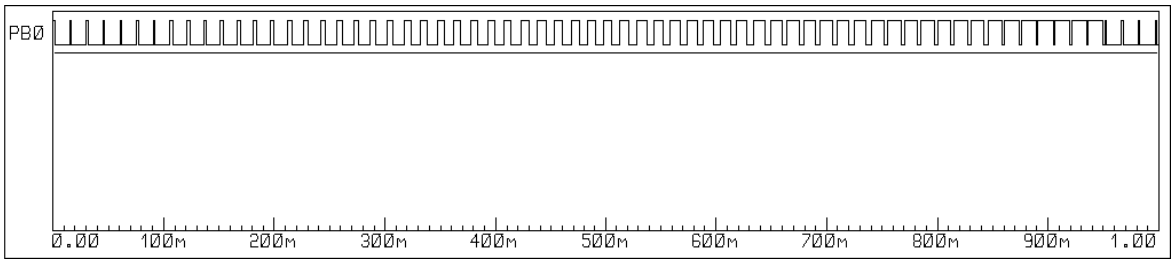


Рис. 2.78.

Зниження середнього прямого струму через сегменти викликає зниження світловіддачі напівпровідникового матеріалу індикатора, тобто регулювання яскравості індикатора.

Приклад 2.19. Написати програму для мікроконтролера, яка забезпечує плавне запалення і згасання світлодіодів через один. Принципова схема представлена на рис. 2.79. Аноди світлодіодів підключені до лінії живлення 5В, а катоди через струмообмежувальні резистори до виводів мікроконтролера. Таке схемотехнічне рішення використане з тієї причини, що мікроконтролер на своїх виводах може прийняти більший струм, ніж віддати. Таким чином, не відбувається перевантаження вихідних буферів мікроконтролера.

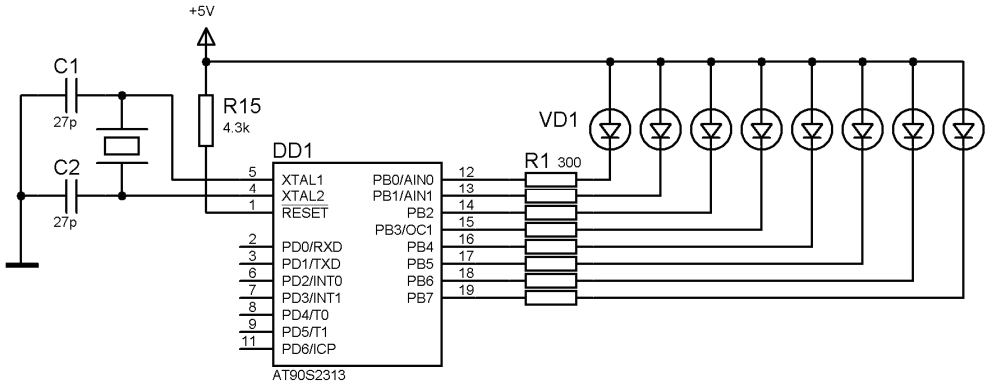


Рис. 2.79.

Текст програми для мікроконтролера представлено нижче

```
#include <io2313v.h>                // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                  // функція затримки
{
 int t1,t2;                         // додаткові змінні
 for(t1=0;t1<ms;t1++)
 for(t2=0;t2<100;t2++) NOP();       // у циклі виконується «пуста» операція
}
void main()                         // головна функція програми
{
 char C=1;                          // оголошення змінної
 DDRB=0xFF;                         // налаштування порту B на вихід
 while(1)                           // вічний цикл
 {
 for(C=1;C<100;C++)                 // плавне загоряння/згасання світлодіодів через один
 {
        PORTB=0b01010101;           // виведення значеня 0x55 в порт B
        delay(C);                   // затримка C мілісекунд
        PORTB=0b10101010;           // виведення значення 0xAA в порт B
        delay(100-C);               // затримка 100-C мілісекунд
        }
for(C=1;C<100;C++)
        {
        PORTB=0b01010101;           // виведення значення 0x55 в порт B
        delay(100-C);               // затримка 100-C мілісекунд
        PORTB=0b10101010;           // виведення значення 0xAA в порт B
        delay(C);                   // затримка C мілісекунд
 }
 }
}
```

На рис. 2.80 представлена часова діаграма роботи пристрою. Світлодіоди, підключені до виводів PB0, PB2, PB4 і PB6 плавно згасають, а

світлодіоди, підключені до виводів PB1, PB3, PB5 і PB7 плавно запалюються протягом 500 мс. Потім цикл повторюється знову.

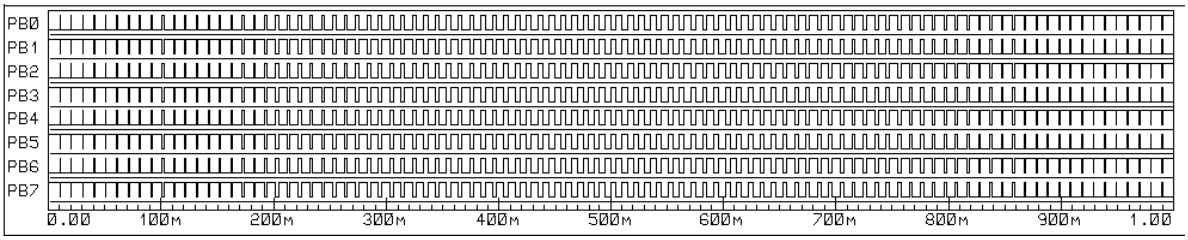


Рис. 2.80.

Схеми регулювання яскравості індикаторів з використанням генераторів широтно-модульованих імпульсів можуть бути різними. Проте будь-які варіанти такої схеми регулювання яскравості можуть використовуватися лише при обмеженій кількості індикаторів, оскільки одночасне включення-виключення великої кількості індикаторів викликає значні зміни струму джерела живлення. Боротьба з такими перешкодами в мікросхемній частині приладу часто викликає значні труднощі. Наприклад, найбільший струм індикатора ЗЛС324Б1 складає 140 мА і при одночасному регулюванні яскравості 15 індикаторів зміна струму складе більше 2А. Це необхідно враховувати при проектуванні джерела живлення і розводці провідників, або друкованого монтажу. Крім того, такий індикатор не можна безпосередньо під'єднати до виводів мікроконтролера, тому необхідно використовувати буферні елементи або транзисторні ключі.

2.13. Драйвери напівпровідникових індикаторів

Напівпровідникові індикатори є струмовими приладами, для їх нормального функціонування необхідно стабілізувати прямий струм через кожен елемент. Стабілізація інтенсивності випромінювання світлодіодних індикаторів має велике значення. Від інтенсивності випромінювання залежить надійність і безпомилковість зчитування інформації з індикаторів, якість зображення, максимальний кут огляду. Основними причинами зміни яскравості світлодіодних індикаторів є зміна температури, деградація напівпровідникових матеріалів, що призводить до зміни опору, і як наслідок, зміні струму через світлодіод. Відомі наступні методи стабілізації інтенсивності випромінювання:

– термостатування;

– ізоляція індикатора від дії дестабілізуючих факторів;

– введення в ланцюг живлення світлодіодів термочутливих елементів із зворотною температурною залежністю;

– вибір оптимальних значень опорів в ланцюзі живлення індикатора;

– стабілізація робочого режиму шляхом введення зворотного зв'язку

по термозалежному параметру індикатора;

– введення зворотного зв'язку по оптичному каналу.

Основною причиною зміни яскравості напівпровідникового індикатора є зміна струму через *p*-*n*-перехід. Таким чином, стабілізувавши струм через індикатор можна зменшити нестабільність випромінювання.

Перші два способи дозволяють зменшити нестабільність випромінювання, проте вони відрізняються складністю та інерційністю, а також мають обмежені можливості, оскільки вимагають вживання додаткових заходів для ослаблення нестабільності через розкид параметрів індикаторів і їх зміною внаслідок старіння.

Із збільшенням температури відбувається зменшення падіння напруги на напівпровідниковому індикаторі (світлодіоді). При паралельному включенні опору і світлодіода, розподіл струму відбувається у бік ланцюга зі світлодіодом. При оптимальному виборі опору вдається стабілізувати інтенсивність випромінювання. Проте цей метод вимагає великих витрат потужності і не усуває часову нестабільність.

Для вирішення задачі стабілізації інтенсивності випромінювання, а також живлення світлодіодів, які неможна безпосередньо підключати до цифрових мікросхем і мікроконтролерів зважаючи на великий струм споживання, рекомендується використовувати спеціальні світлодіодні драйвери.

Промисловістю випускається широка номенклатура мікросхем – драйверів світлодіодів: одноканальні і багатоканальні без шини управління і з шиною управління, драйвери RGB – світлодіодних кластерів, драйвери для світлодіодних екранів. Деякі з мікросхем–драйверів мають вбудовані інтерфейси, що дозволяють організувати їх в мережу, а також засоби діагностики індикаторів.

У цьому параграфі представлені основні види, характеристики і функції драйверів фірми Macroblock, як одного з кращих, в даний час виробників драйверів і найбільш ефективних з точки зору ціни. Фірма спеціалізується лише на драйверах для світлодіодів, і, відповідно, концентрує свої ресурси лише на драйверах.

Одним з найбільш простих є драйвер MBI 1801 – драйвер підвищеної потужності без шини управління. Одноканальний світлодіодний драйвер MBI 1801 застосовується для управління потужними світлодіодами, в автомобільному освітленні, для зовнішної реклами, в декоративному освітленні, для підсвічування рідкокристалічних панелей, в копіювальній техніці, в сканерах. Виходи постійного струму інваріантні до зміни напруги на навантаженні. Підстроювання вихідного струму здійснюється за допомогою зовнішнього резистора.

В драйвері MBI 1801 реалізовані технології:

– PrecisionDrive™ – поліпшення вихідних характеристик і точності задання струму, як між каналами однієї мікросхеми, так і між окремими корпусами мікросхем;

– All-Ways-ON™ – швидке включення-виключення світлодіодів одним сигналом, без зняття живлення;
– управління одним виводом (OE);
– програмована яскравість ШІМ (PWM) 0-100%;
– захист від перегрівання. Вбудований сенсор температури відключає драйвер при нагріванні більше 100-150 °C.
На рис. 2.81 представлена структурна схема драйвера.

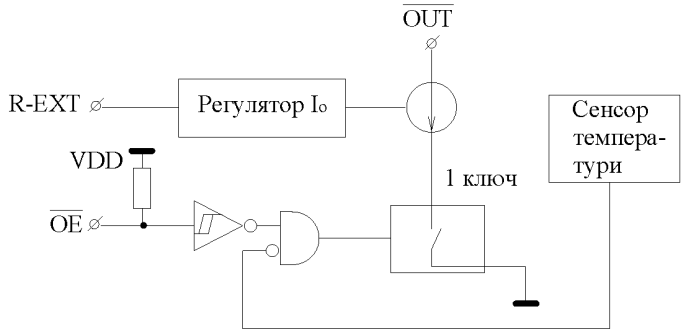
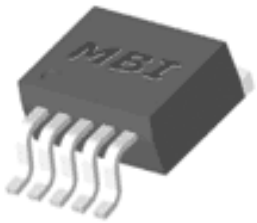


Рис. 2.81.

Драйвер має 1 канал управління, вихідний струм якого регулюється в межах від 50 до 1200 мА. Напруга живлення складає 5В, мінімальна тривалість імпульсу 1 мкс. На рис. 2.82 представлений зовнішній вигляд мікросхеми і схема розташування виводів.



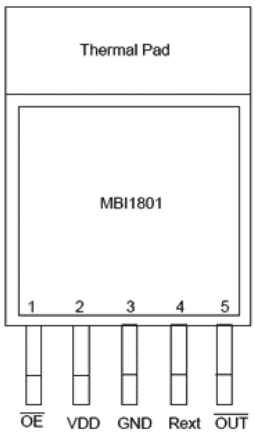


Рис. 2.82.

В табл. 2.8 приводится опис виводів мікросхеми.

Табл. 2.8.

Номер контакту – Позначення	**Функція**
1 – OE	Дозвіл виходу
2 – VDD	Живлення
3 – GND	Заземлення
4 – R-EXT	Програмування струму (зовнішній резистор)
5 – OUT	Вихід

В драйвері MBI 1802 реализовані технології:

– PrecisionDrive™ – поліпшення вихідних характеристик і точності задання струму, як між каналами однієї мікросхеми, так і між окремими корпусами мікросхем;

– All-Ways-ON™ – швидке включення-виключення світлодіодів одним сигналом, без зняття живлення;

– управління одним виводом (OE);

– програмована яскравість ШІМ (PWM) 0-100%;

– захист від перегрівання.

– об'єднання виходів;

– економічний режим. Скидання 75% потужності одним сигналом.

На рис. 2.83 представлена структурна схема двохканального драйвера.

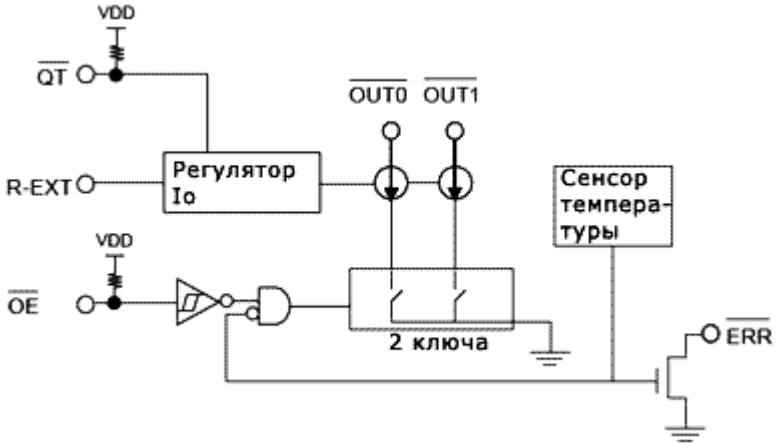


Рис. 2.83.

Точність формуання струму між каналами лежить в межах 3%. Точність формування струму між мікросхемами лежить в межах 6%. Вихідний струм кожного з каналів лежить в межах від 40 до 360 мА. На рис. 2.84 представлена схема розташування виводів.

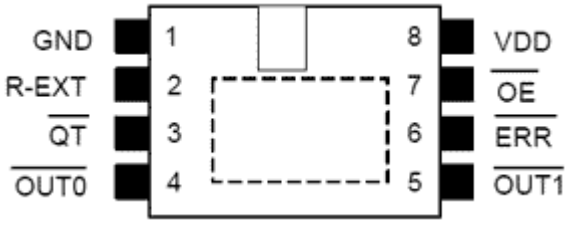


Рис. 2.84.

В табл. 2.9 приводится опис виводів мікросхеми.

Табл. 2.9.

Номер контакту – Позначення	Функція
1 – GND	Заземлення
2 – R-EXT	Програмування струму (зовнішній резистор)
3 – QT	Установка економічного режиму
4,5 – OUT 0-1	Вихід
6 – ERR	Помилка при перевищені температури 150°C
7 – OE	Дозвіл виходу
8 – VDD	Живлення

На рис. 2.85 представлена структурна схема чотирьохканального драйвера MBI 1804.

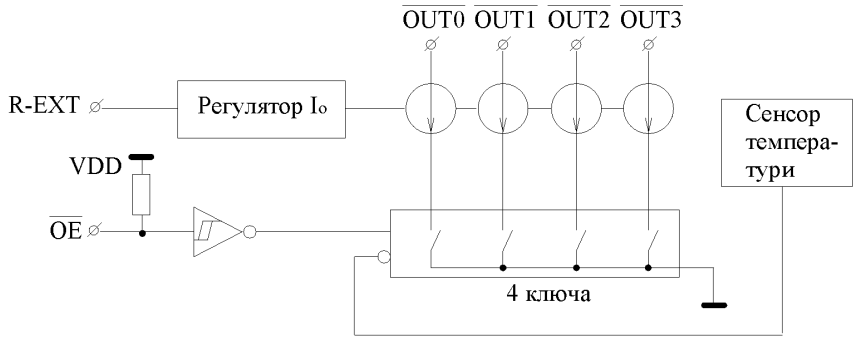


Рис. 2.85.

Вихідний струм кожного з каналів лежить в межах від 40 до 240 мА. Характеристики аналогічні мікросхемам MBI 1801 і MBI 1802.

Технологія Precisiondrive™ використовує новітню технологію поліпшення вихідних характеристик і точності задання струму, як між каналами однієї мікросхеми, так і між окремими корпусами мікросхем. Основні характеристики даної технології показані на рис. 2.86.

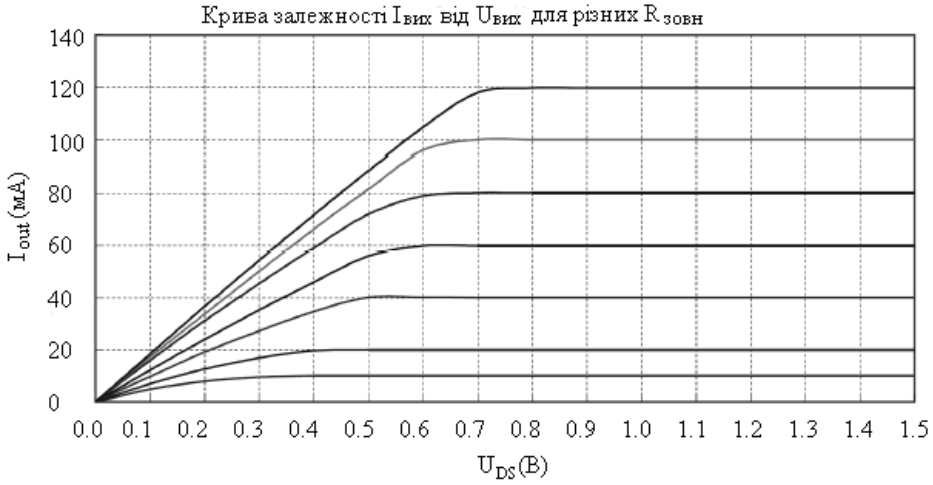


Рис. 2.86.

Відмінними особливостями цієї технології є:

– точність задання струму ±3% між каналами одної мікросхеми;
– точність задання струму ±6% між різними мікросхемами;
– зміна струму виводу, при зміні живлячого струму і температури довкілля не перевищує 1%;
– зміна струму світлодіодів, при зміні прямої напруги, підтримується з точністю 0,1% на вольт.

У розглянутих драйверах для регулювання вихідного струму використовується зовнішній резистор. Випускаються також мікросхеми з шиною управління, що дозволяє виконати їх сполучення з мікроконтролером і виконувати програмне налаштування вихідного струму і, відповідно, яскравості напівпровідникового індикатора.

Технологія Share-I-O™ дозволяє без додаткових виводів, використовуючи стандартну шину (рис. 2.87), забезпечувати такі додаткові функції як діагностика несправності світлодіодів з їх локалізацією, управління яскравістю світлодіодів (зміна струму). На рис. 2.88 показаний принцип діагностики несправних світлодіодів: а – визначення обривів; б – визначення замикань

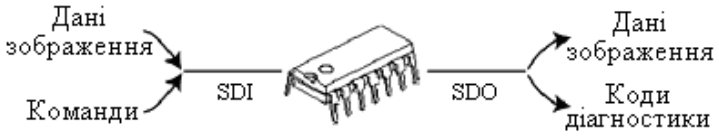
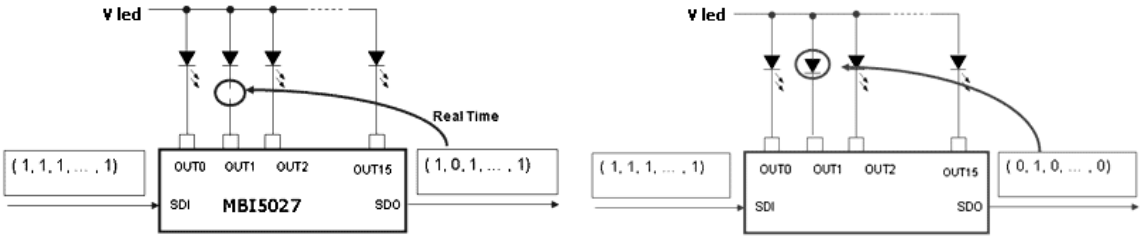


Рис. 2.87.



а *б*

Рис. 2.88.

Мікросхема MBI 5171 є 8-канальним драйвером світлодіодів. У ній також застосована технологія Precision Drive TM для поліпшення її вихідних характеристик. Крім того, в ній використовується ідея технології Share I-O для того, щоб розширити її функціональність для визначення помилок в світлодіодному навантаженні в режимі реального часу. На рис. 2.89 представлена структурна схема чотирьохканального драйвера MBI 5171.

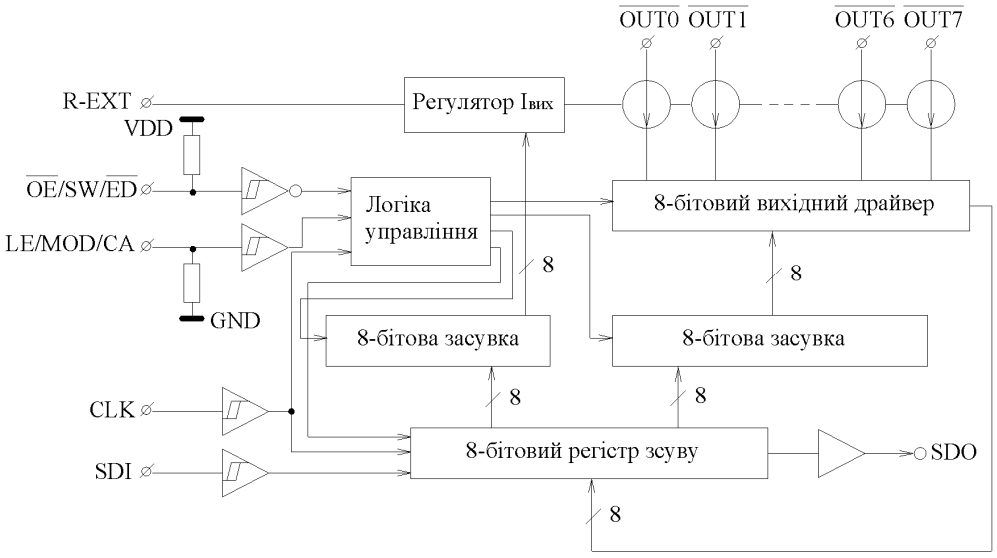


Рис. 2.89.

Мікросхема містить 8-бітовий регістр зсуву і 8-бітову вихідний фіксатор, які перетворюють послідовні вхідні дані в паралельний формат на виході. У конструкції вихідних каскадів є 8 керованих струмових портів, які забезпечують рівномірне і стабільне відведення струмів для управління світлодіодами в широкому діапазоні відхилень вихідної напруги. Споживачі можуть регулювати вихідний струм від 5 до 120 мА зовнішнім резистором R_{ext}, який забезпечує гнучке управління інтенсивністю світіння світлодіодів. Гарантована максимальна напруга на вихідних портах, яка витримує мікросхема, дорфінює 17 В. Високая тактова частота (до 25 МГц) задовольняє вимогам систем з великим об'ємом даних.

У мікросхемі є 2 робочих режими і 3 фази: фаза нормального режиму, перехідна фаза перемикання режиму і фаза особливого режиму. Сигнал на багатофункціональному виводі може бути керованим. Коли на виводі з'являється короткий імпульс, рівний тривалості синхроімпульса, мікросхема переходить у фазу перемикання режиму. У цей момент рівень напруги на ньому використовується для визначення наступного режиму, в який готується перейти мікросхема.

У фазі нормального режиму послідовні дані могли б передаватися через виводи SDI мікросхеми, зсуватися в регістрі зсуву і виходити через вивід SDO. Вивід LE/MOD/CA може фіксувати послідовні дані в регістрі зсуву для вихідного фіксатора. Вивід $\overline{OE}/SW/\overline{ED}$ відкриває вихідні драйвери для відбору струму.

У фазі особливого режиму сигнал низького рівня $\overline{OE}/SW/\overline{ED}$ може відкривати вихідні канали і визначати стан вихідного струму, щоб визначити, чи досить високий рівень вихідного струму. Стан визначеної помилки потім поступить у 8-бітовий регістр зсуву і просувається на вихід

паралельно з синхросигналом CLK. Потім контролер системи зчитує стан і визначає, чи правильно горять світлодіоди. З іншого боку, у фазі особливого режиму мікросхема дозволяє також регулювати рівень вихідного струму шляхом установки програмованого коду конфігурації в поточному часі. Код поступає через вивід SDI. Позитивний імпульс на виводі LE/MOD/CA фіксує код регістра зсуву не у вихідному фіксаторі, а у вбудованому 8-бітовому фіксаторі конфігурації. Код змінює напругу на виводі R_{ext} і управляє регулятором вихідного струму. Точне регулювання здійснюється зміною підсилення в діапазоні від 1/12 до 127/128 в 256 кроків. Таким чином, розкид вихідних струмів між мікросхемами може бути вирівняний в межах менше 1% і це дозволяє здійснювати в кольорових світлодіодних панелях баланс білого.

При використанні в СВД-дісплеях мікросхема забезпечує мінімальний розкид струмів як між каналами, так і між мікросхемами. Це досягається завдяки тому, що при $I_{OUT} \leq 100$ мА максимальна відмінність величин струмів різних каналів менше 1%, а в однотипних мікросхем – менше 6 %. Вихідний струм залишається постійним незалежно від змін вхідної напруги (V_f). Вихідний струм ($I_{OUT,\ TARGET}$) в області насичення рівномірний.

Мікросхема підсилює опорний струм I_{ref} , встановлений зовнішнім резистором R_{EXT} до величини I_{OUT} в кожному порті. Для розрахунку цільового вихідного струму використовуються наступні формули:

$$V_{R\text{-}EXT} = 1{,}25 \cdot \text{Volt} \times \text{VG},$$
$$I_{ref} = V_{R\text{-}EXT}/R_{EXT},$$
$$I_{OUT,\ TARGET} = I_{ref} \times 15 \times 3^{(CM-1)},$$

де R_{EXT} – опір зовнішнього резистора, підключеного до виводу REXT, а V_{R_EXT} – напруга на ньому. Струм управляється програмованим підсиленням напруги, що визначається кодом конфігурації. Множник струму CM визначає величини відношення $I_{OUT,\ TARGET}/I_{ref}$ рівними 15 або 5.

$$V_{R\text{-}EXT} = 1{,}25 \cdot \text{Volt} \times 127/128 = 1{,}24 \cdot \text{Volt},$$
$$I_{OUT,\ TARGET} = (1{,}24 \cdot \text{Volt}/R_{EXT}) \times 15.$$

Таким чином, величина струму по замовчуванню дорівнює 50 мА при 372 Ом, і 25 мА при 744 Ом.

Залежність $I_{OUT,\ TARGET}$ від R_{EXT} показана на рис. 2.90.

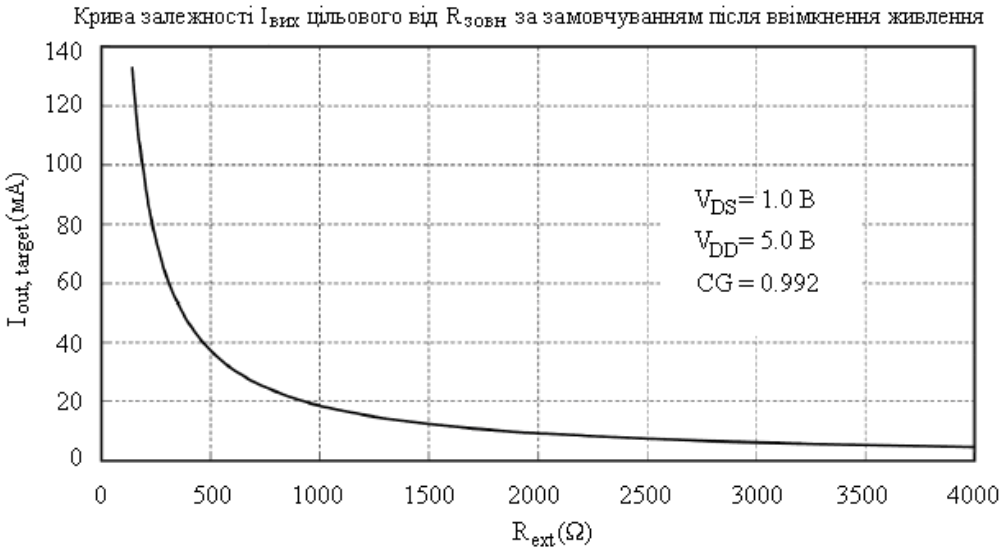
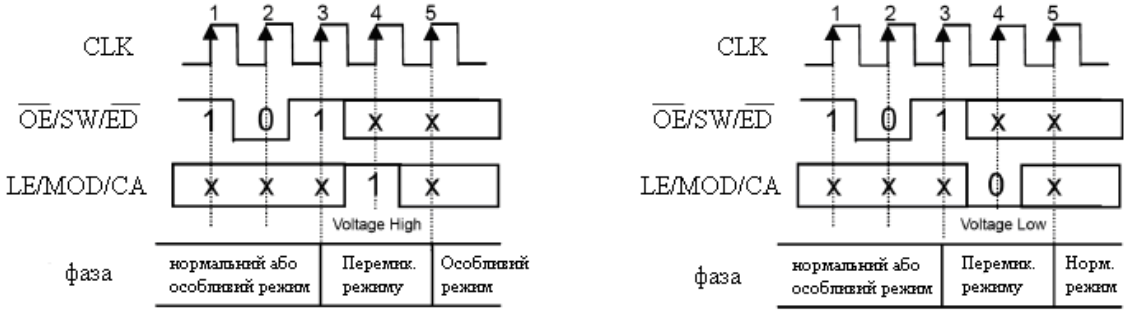


Рис. 2.90.

Для розширення функціональних можливостей виводів мікросхеми з метою забезпечення можливості визначення помилок в навантаженні і програмування струмів в особливому режимі поряд із звичайною роботою в нормальному режимі, в мікросхемі застосована методика Share-I-O™. Для переходу з режиму в режим використовується управляючий сигнал на виводі $\overline{OE}/SW/\overline{ED}$. Коли на ньому з'являється сигнал, рівний по тривалості одному синхроімпульсу, мікросхема перейде у фазу перемикання режиму, тривалість якої дорівнює двом періодам синхроімпульса. Після включення живлення режимом по замовчуванню є нормальний режим. На рис. 2.91 показані часові діаграми перемикання режимів роботи мікросхеми (а – особливий режим; б – нормальний режим).



а б

Рис. 2.91.

Як показано на рисунках, при появі на виводі $\overline{OE}/SW/\overline{ED}$ короткого імпульсу «101», рівного одному періоду синхроімпульса, мікросхема переходить у фазу перемикання режиму. На 4-му передньому фронті синхроімпульса, якщо на виводі LE/MOD/CA високий рівень, мікросхема переходить в особливий режим. Інакше вона переходить в нормальний

режим. Слід зазначити, що між 3 і 5 передніми фронтами синхроімпульса мікросхема не може зафіксувати жодних даних. Його рівень використовується лише для визначення режиму, що перемикається. Однак короткий імпульс на $\overline{OE}/SW/\overline{ED}$ може відімкнути вихідні порти. Під час перемикання режиму послідовні дані можуть передаватися через SDI і просуватися на вивід SDO. Послідовність сигналів для перемикання режимів часто можна використовувати для того, щоб переконатися, в якому режимі працює мікросхема. Значення «1» і «0» сприймаються за переднім фронтом синхросигналу. Символ «X» означає, що рівень сигналу не впливає на механізм перемикання режиму.

Мікросхема в особливому режимі здійснює визначення помилки і завантажує коди стану помилки навантаження в регістр зсуву, а також відкриває вихідні порти для відведення струму. На рис. 2.92 приведена часова діаграма для визначення помилки. Вказані «0» і «1» сприймаються за переднім фронтом кожного синхроімпульса.

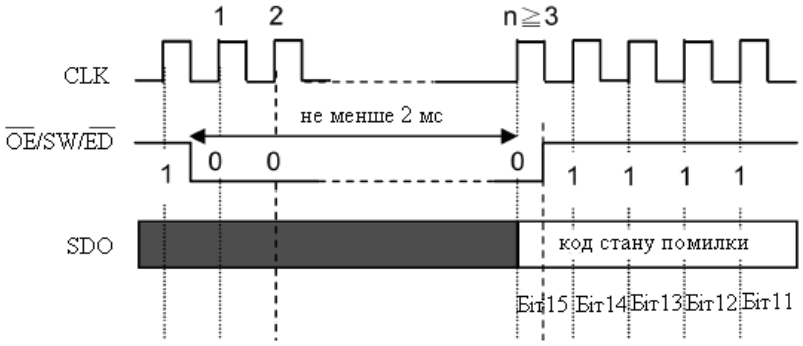


Рис. 2.92.

При низькому рівні на виводі $\overline{OE}/SW/\overline{ED}$ мають бути прийняті щонайменше три «0». Як тільки надійде другий «0», в регістр зсуву замість послідовних даних з SDI почнуть поступати 8-и бітові паралельні коди стану помилки зі схеми визначника помилок. Зазвичай коди стану помилки починають правильно формуватися через 2 мс після заднього фронту імпульсу на $\overline{OE}/SW/\overline{ED}$. Поява 3-х і більшої кількості нулів призводить до запам'ятовування коду стану певної помилки в регістрі зсуву. Таким чином, коли на виводі $\overline{OE}/SW/\overline{ED}$ низький рівень, послідовні дані через вивід SDI мікросхеми проходити не можуть. Але під час переходу рівня з низького на високий, джерелом вхідних даних для регістра зсуву знову стає вивід SDI. В цей же час вихідні порти замкнуті і визначення помилки завершене. Після цього коди стану помилки, що зберігаються в регістрі зсуву, можуть виводитися біт за бітом в такт з синхросигналом, і в мікросхему можуть поступати нові послідовні дані через вивід SDI. Обмеженням є неможливість в особливому режимі одноразової передачі послідовних даних і визначенні стану помилки світлодіодного навантаження.

Активний високий рівень на вході LE/MOD/CA фіксує послідовні дані регістра зсуву не у вихідному регістрі, а в регістрі конфігурації (рис. 2.93). Зафіксовані послідовні дані є кодом конфігурації. Коди зберігаються в пам'яті до відключення живлення або до перезапису регістра конфігурації.

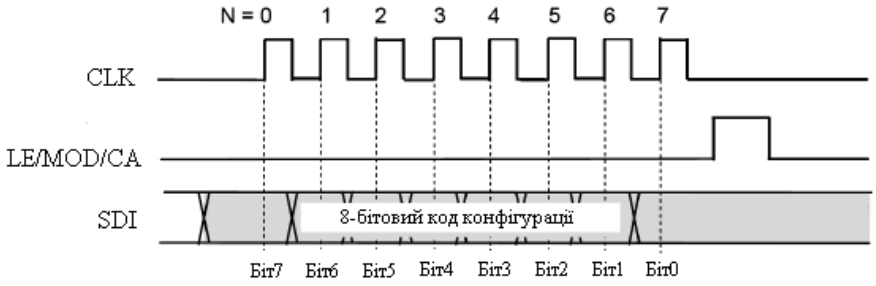


Рис. 2.93.

Користувач повинен звертати увагу на послідовність визначення помилки і регулювання струму, щоб уникнути перезапис коду конфігурації кодом стану помилки.

У табл. 2.10 приведений опис бітів коду конфігурації в регістрі конфігурації.

Табл. 2.10.

	Опис бітів 8-бітового коду конфігурації							
	Біт 0	Біт 1	Біт 2	Біт 3	Біт 4	Біт 5	Біт 6	Біт 7
Значення	CM	HC	CC0	CC1	CC2	CC3	CC4	CC5
За замовчуванням	1	1	1	1	1	1	1	1

Спочатку на виводі SDI мікросхеми поступає біт 7. Біти 1-7 {HC, CC[0:5]} визначають підсилення напруги (VG), яка впливає на напругу на виводі R-EXT і опорний струм I_{ref}, що протікає через зовнішній резистор на виводі R-EXT. Біт 0 – біт-помножувач струму, що визначає співвідношення $I_{OUT,TARGET}/I_{ref}$. Будь-яка комбінація VG і CM дає підсилення струму (CG).

Залежність між {HC, CC[0:5]} наступна:

$$VG=(1+HC)\times(1+D/64)/4,$$
$$D=CC0\times2^5+CC1\times2^4+CC2\times2^3+CC4\times2^1+CC5\times2^0,$$

де HC – 1 або 0, а D – двійкове значення CC[0:5]. Таким чином, VG може розглядатися як число з плаваючою точкою з однобітовою експонентою H и 6-бітовою мантисою CC[0:5]. {HC, CC[0:5]} ділить програмоване підсилення напруги VG на 128 кроків і 2 піддіапазони (рис. 2.94):

– низьковольтний піддіапазон (HC=0): VG = 1/4 ~ 127/256, лінійно розділений на 64 кроки;

– високовольтний піддіапазон (HC=1): VG = 1/2 ~ 127/128, лінійно розділений на 64 кроки.

На рис. 2.94 представлена залежність підсилення струму CG від коду конфігурації

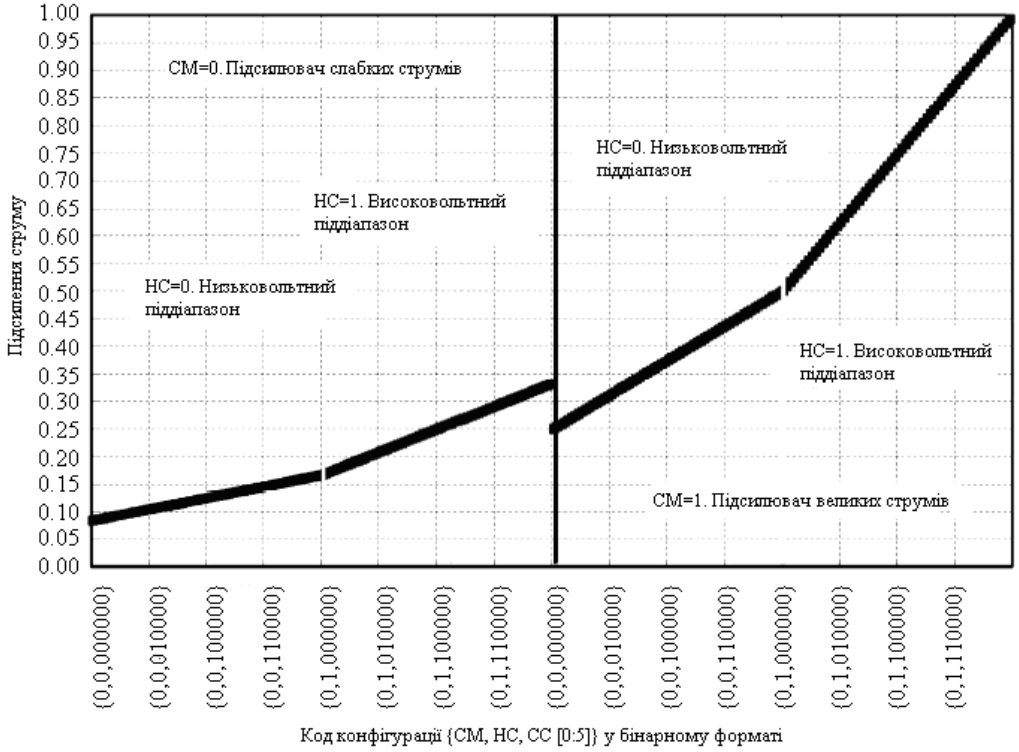


Рис. 2.94.

На рис. 2.95 і рис. 2.96 представлені часові діаграми для регулювання струму і діагностування світлодіодів відповідно.

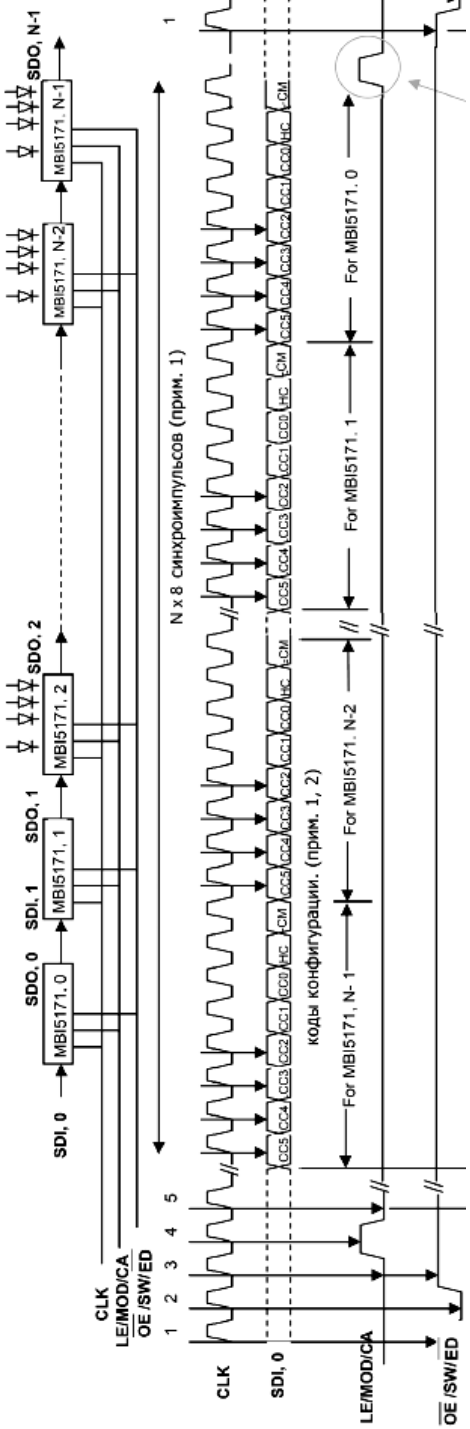


Рис. 2.95. Часова діаграма для регулювання струму

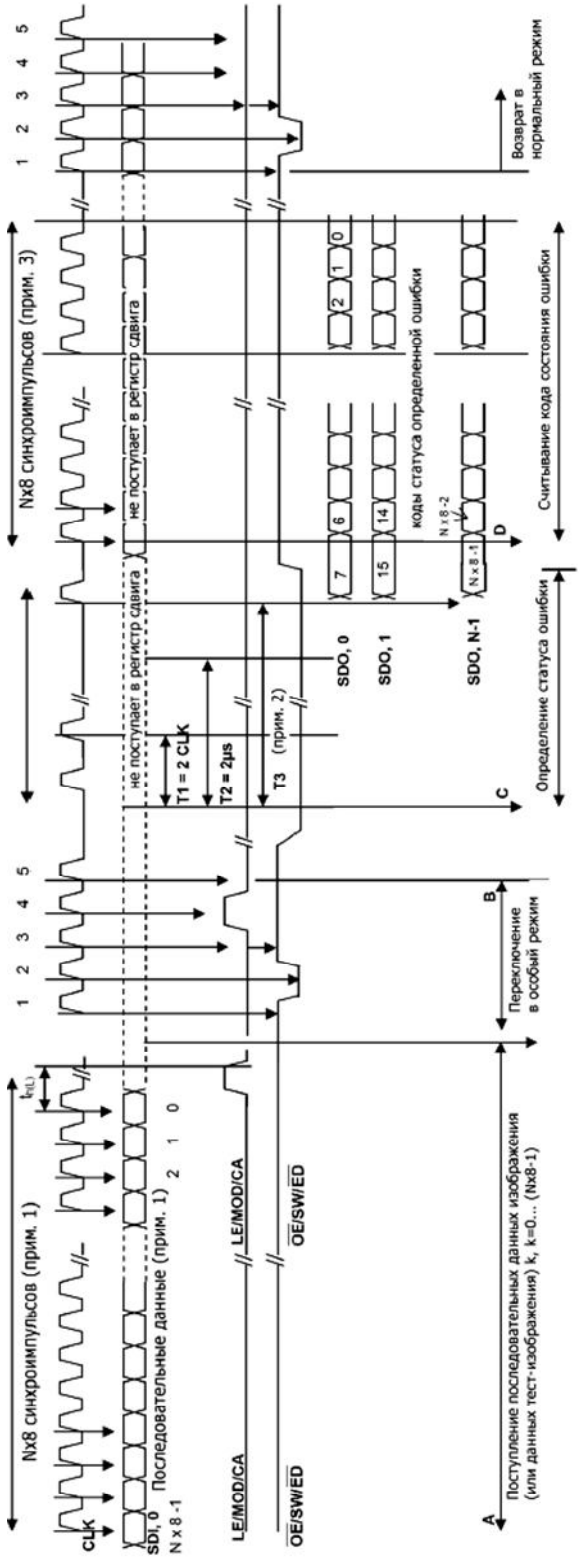


Рис. 2.96.

Трьохканальний світлодіодний драйвер для RGB кластерів MBI 6010 застосовується для управління світлодіодами в декоративному освітленні, архітектурному освітленні, освітленні для розважальних установ, світлодіодних виробах для прикраси міст, ландшафтному освітленні, дорожніх знаках. На рис. 2.97 представлена структурна схема трьохканального драйвера, а на рис. 2.98 – часова діаграма керування драйвером.

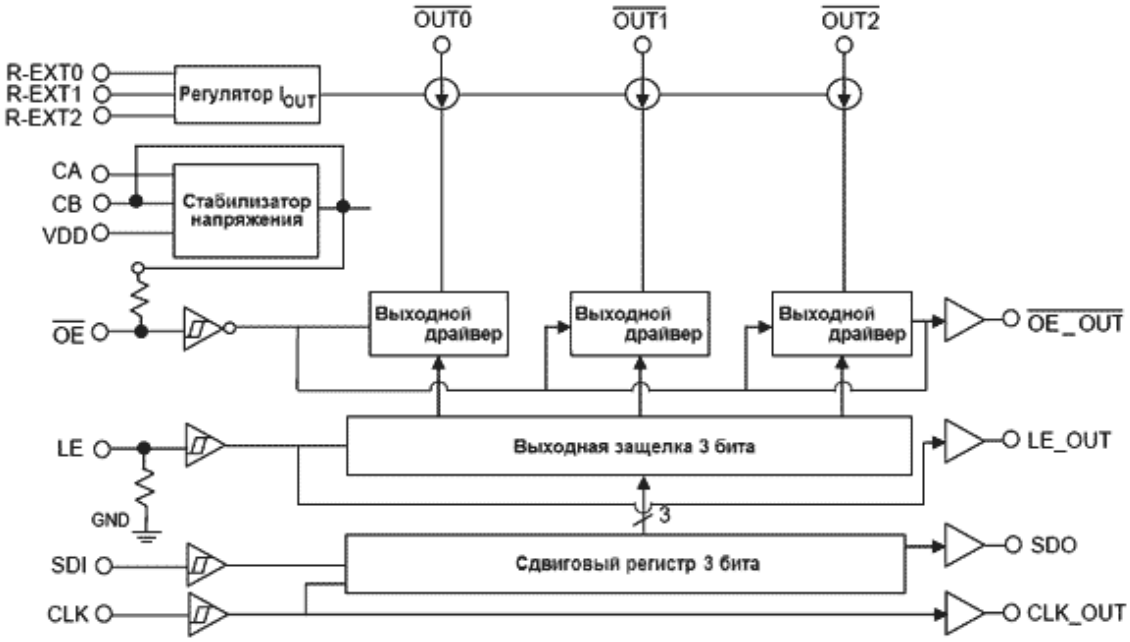


Рис. 2.97.

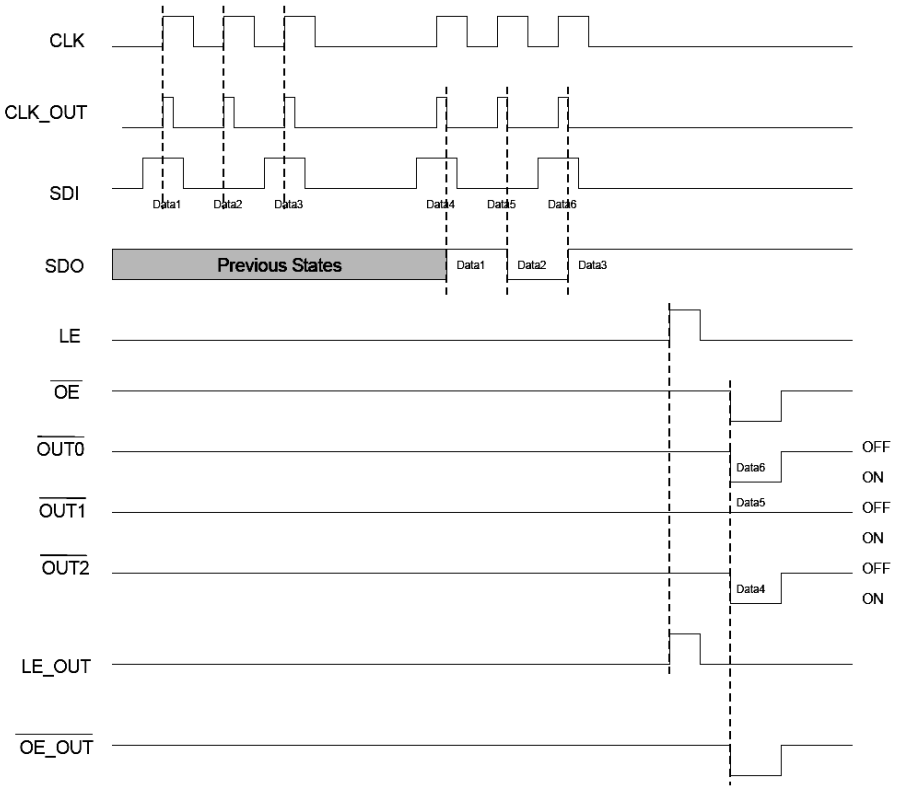


Рис. 2.98.

Табл. 2.11.

	TI	Toshiba	SiTI	STM	Macroblock
Типи мікросхем	TLC5921	TB62726, TB62725, TB62706, TB62705	ST2221C, ST2221A, DM135, DM132	STP16C596, STP08C596	MBI5026, MBI5168
Точність по струму - канал-канал - мікросхема-мікросхема	± 4% ± 10%	± 4% ± 12%	± 4% ± 10%	± 4% ± 10%	± 3% ± 6%
Діагностика несправностей світлодіодів	TLC5921/ 5905 - по мікросхемі - багатократне сканування	TB62727 - по мікросхемі - багатократне сканування	DM133	–	MBI5027, MBI5169 - індивідуально по світлодіоду - в режимі реального часу
Баланс білого (управління яскравістю)	TLC5902/ 5905 5-біт (32 кроки)	TB62727 6-біт (64 кроки) Індивідуально	DM133	–	MBI5028, MBI5170 7-біт (128 кроків) глобально
Кількість виводів	TLC5921: 32 TLC5902: 64 TLC5905: 100	TB62727: 30 TB62718: 64	28	–	MBI5027/5028 : 24

2.14. Принцип роботи біжучого рядка

Біжучі рядки на світлодіодах, як елемент інформаційного носія, можуть використовуватися як на фасаді будівель, так і всередині. Це дуже сучасний рекламний носій, що дозволяє людині отримувати цікаву інформацію у вигляді букво-цифрової послідовності, а також графіки.

Управління будь-яким світлодіодним рядком полягає в передачі спеціальної управляючої послідовності в ПЗП рядка. Управляюча послідовність – це спеціальний протокол, який дозволяє в певному порядку виводити час, температуру (тиск, вологість) та іншу інформацію на рядок, при цьому можуть використовуватися різні спецефекти (наприклад, вибігання по букві, поява тексту зверху внизу і так далі). Формування управляючої послідовності здійснюється за допомогою програмного забезпечення. Виникає також необхідність сполучення світлодіодного рядка з якимось іншим устаткуванням, що не є комп'ютером, наприклад мікроконтролером. При цьому принципи управління біжучим рядком

залишаються однаковими. Передача інформаційної послідовності може здійснюватися через порти RS232, LPT, USB та ін. Використання багатоколірних світлодіодів дозволяє створювати кольорові зображення (рис. 2.99, рис. 2.100).



Рис. 2.99.



Рис. 2.100.

Для побудови засобів відображення інформації типу “біжучий рядок”, використовуються регістри зсуву.

Пояснимо принцип формування символів. Передбачимо, що необхідно сформувати і вивести на біжучий рядок стилізований напис R1 (рис. 2.101).

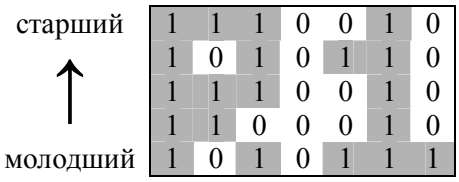


Рис. 2.101.

Для цього необхідно виконати таку послідовність дій:

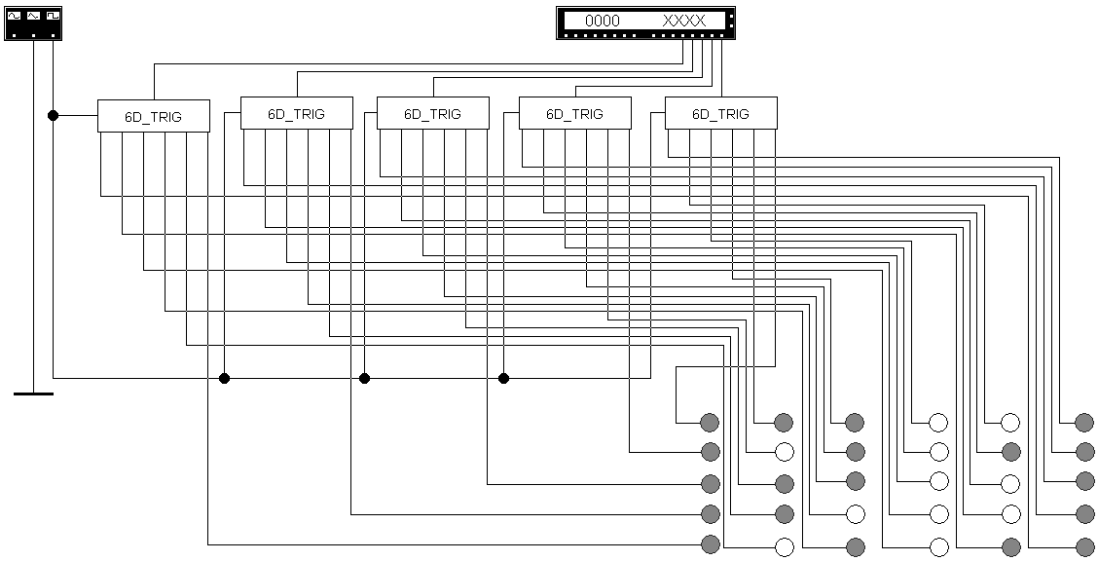
– на аркуші паперу вписати необхідні символи в матрицю, висота якої дорівнює кількості світлодіодів в біжучому рядку по вертикалі (замалювати потрібні клітинки);

– привласнити клітинкам, які виявилися зафарбованими, значення “1”, а незафарбованим – значення “0”;

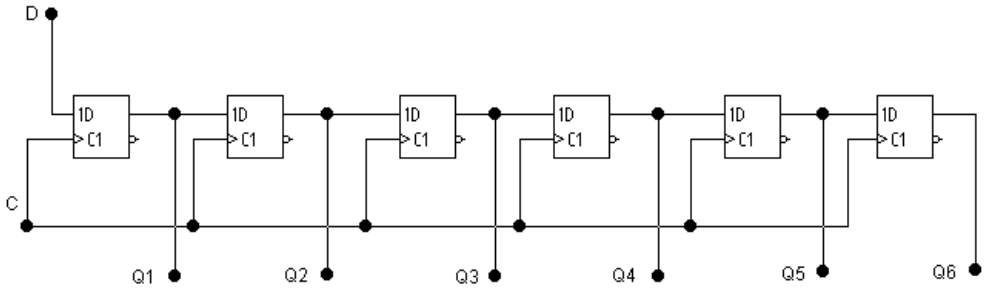
– нанести розрядну сітку;

– отримані числа розміщуються в масиві (в програмі для мікроконтролера або комп'ютера) для того, щоб він формував на своїх виходах в паралельному форматі потрібний двійковий код.

На рис. 2.102 представлена схема біжучого рядка, яка створена в Workbench. Для формування потрібної послідовності чисел використовується блок Word Generator (рис. 2.103).



а



б

Рис. 2.102.

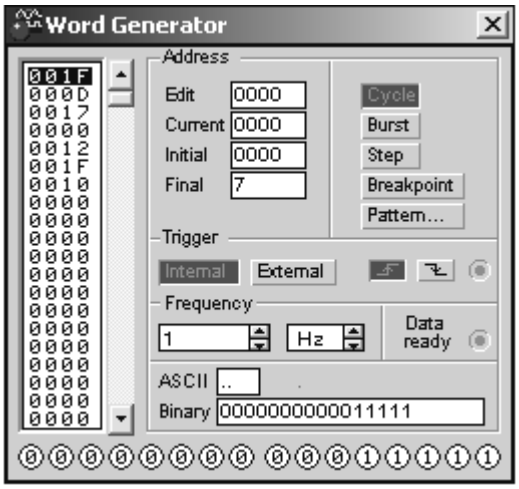


Рис. 2.103.

Приклад 2.20. Розробити пристрій «Біжучий рядок». На рис. 2.104 представлена принципова схема біжучого рядка. Порт С мікроконтролера сполучений з входом паралельного регістра DD4. До виходів паралельного регістра DD3 через струмообмежувальні резистори, підключена лінійка світлодіодів, а також входи регістра DD4. До виходів паралельного регістра DD4 також підключена лінійка світлодіодів, і входи регістра DD5, і т.д. Таким чином, всі паралельні регістри сполучені послідовно, і до виходу кожного з них підключені світлодіоди. При кожному наступному такті відбувається запис нового значення в регістри і створюється ефект руху.

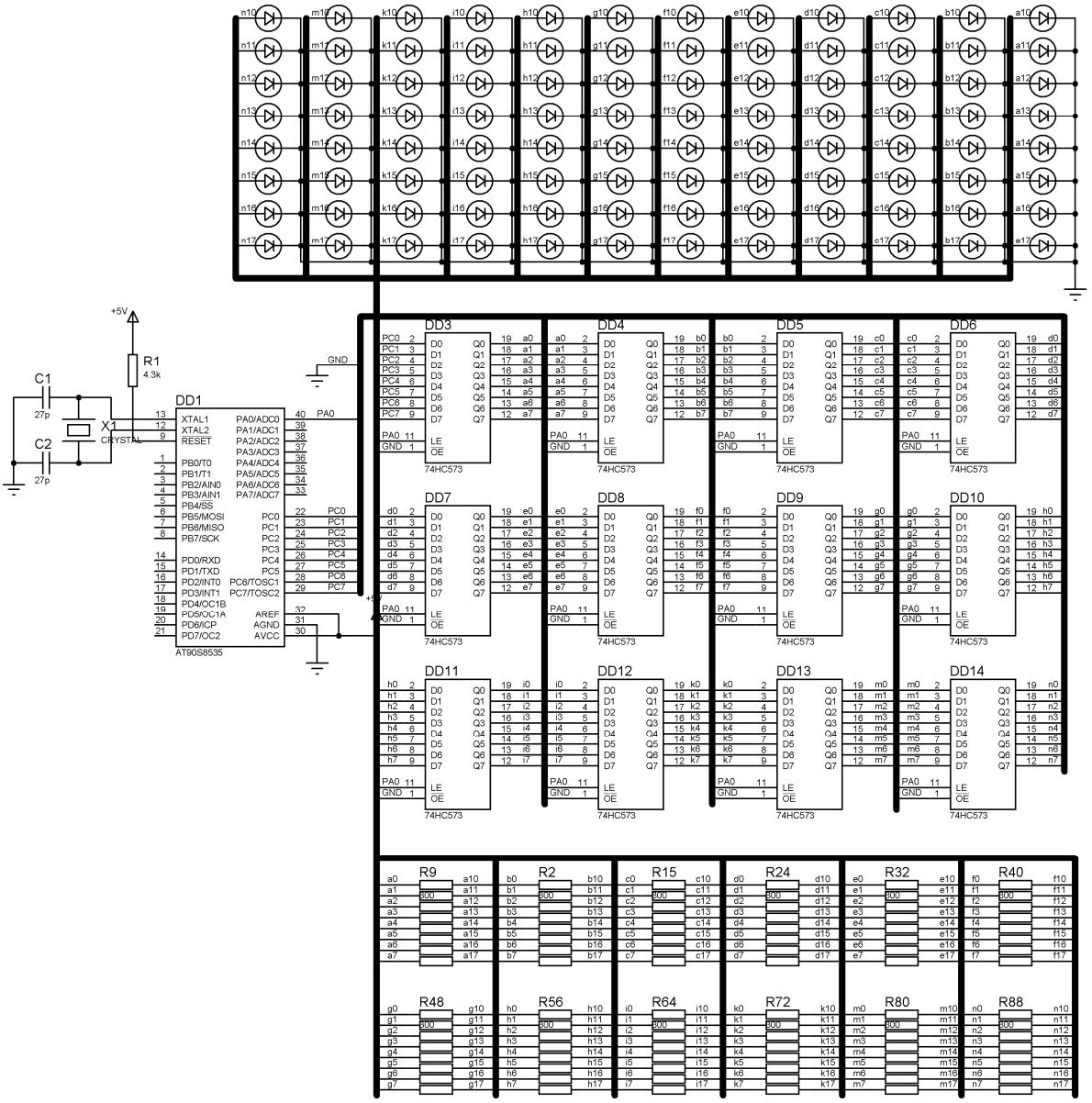


Рис. 2.104.

Табл. 2.12.

1	1	1		1	1	1		1	1	1		0	1	0
0	1	0		1	0	0		1	0	0		1	0	1
0	1	0		1	0	0		1	0	0		1	0	0
0	1	0		1	1	1		1	1	1		1	0	0
0	1	0		1	0	0		1	0	0		1	0	0
0	1	0		1	0	0		1	0	0		1	0	0
0	1	0		1	0	0		1	0	0		1	0	1
0	1	0		1	1	1		1	1	1		0	1	0

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>                // підключення заголовочних файлів
#include <macros.h>
void delay(int ms)                   // функція затримки
```

```
{
 int c1, c2;                          // додаткові змінні
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();      // у циклі виконується «пуста» операція
}
void main()                           // головна функція програми
{
 char i;                              // додаткова змінна
 char text[12]={0b00000001, 0b11111111, 0b00000001, 0b11111111,
0b10001001, 0b10001001, 0b11111111, 0b10001001, 0b10001001,
0b01111110, 0b10000001, 0b01000010};  // в масиві зберігаються коди символів
 DDRA=0x01;                           // вивід PA0 налаштований на вихід
 DDRC=0xFF;                           // порт C налаштований на вихід
 while(1)                             // вічний цикл
 {
  for(i=0;i<12;i++)                   // у циклі виводяться в порт елементи масиву
  {
  PORTC=text[i];                      // виведення елементу масиву в порт
  delay(10);                          // часова затримка
  PORTA=1;                            // установка на виводі PA0 сигналу лог.1
  delay(10);                          // часова затримка
  PORTA=0;                            // установка на виводі PA0 сигналу лог.0
  delay(200);                         // затримка 200 мс
  }
 for(i=0;i<12;i++)
  {
  PORTC=0;                            // виведення в порт C значення 0 для очищення табло
  PORTA=1;                            // установка на виході PA0 сигналу лог.1
  delay(10);                          // часова затримка
  PORTA=0;                            // виведення в порт A значення 0
  delay(200);                         // затримка 200 мс
  }
 delay(1000);                         // затримка 1 с
 }
}
```

На рис. 2.105 представлена часова діаграма роботи біжучого рядка.

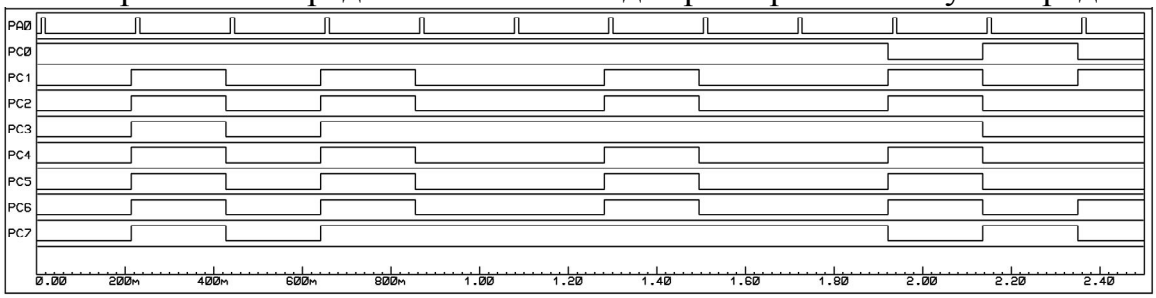


Рис. 2.105.

На рис. 2.106 представлений результат виконання програми. На світлодіодних індикаторах відображається рядок «TEEC».

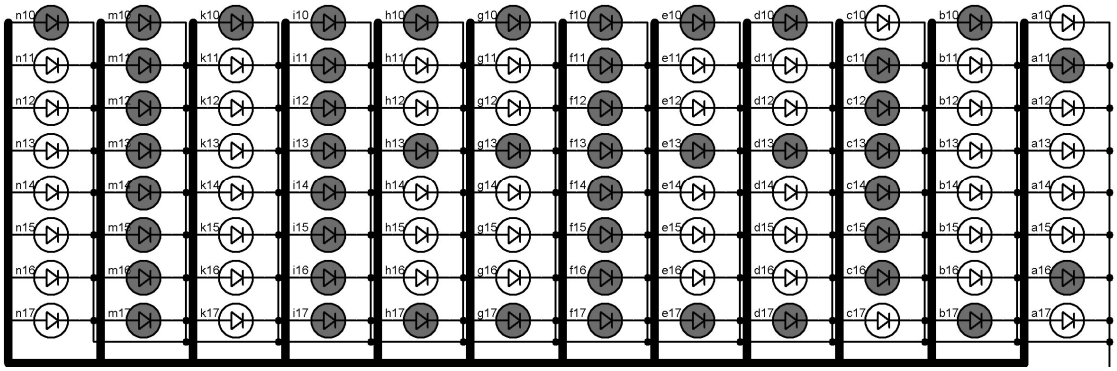


Рис. 2.106.

Приклад 2.21. Написати програму управління біжучим рядком через LPT-порт для комп'ютера.

Операційна система Windows має ряд програм, які не дозволяють безпосередньо звертатися до апаратних ресурсів, зокрема до LPT-порту. Спроба виконання програми, в якій є звернення до якого-небудь регістра порта, викличе помилку під час виконання програми. Для подолання цієї проблеми необхідно встановити спеціальний драйвер, а в написаній програмі цей драйвер підключити.

Існує цілий ряд спеціальних драйверів, встановивши які розробник зможе отримати доступ до апаратних ресурсів комп'ютера. Одним з найбільш популярних драйверів є giveio.sys, який можна знайти в Internet.

1. Для установки драйвера необхідно мати права адміністратора і виконати наступні дії: скопіювати файл giveio.sys у директорію c:\windows\system32\drivers. Запустити програму Loaddrv.exe, вигляд якої представлений на рис. 2.107, і в полі Full pathname of driver вказати шлях до файлу драйвера. Після цього слід натиснути на кнопку Install.

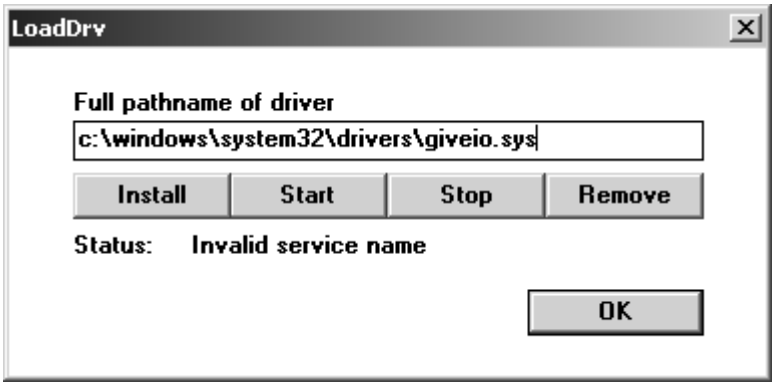


Рис. 2.107.

Після установки драйвера в нижній частині діалогового вікна з'явиться відповідне повідомлення. Потім слід перезавантажити комп'ютер.

Якщо драйвер встановити не вдалося, з'явиться повідомлення про причину помилки.

Тепер драйвер необхідно запустити. Для цього використовується диспетчер пристроїв (рис. 2.108). Для того, щоб побачити повний список встановлених драйверів, необхідно в меню «Вид» встановити прапорець «Показувати приховані пристрої». Встановлений драйвер можна знайти у вузлі «Драйверы устройств не Plug and Play».

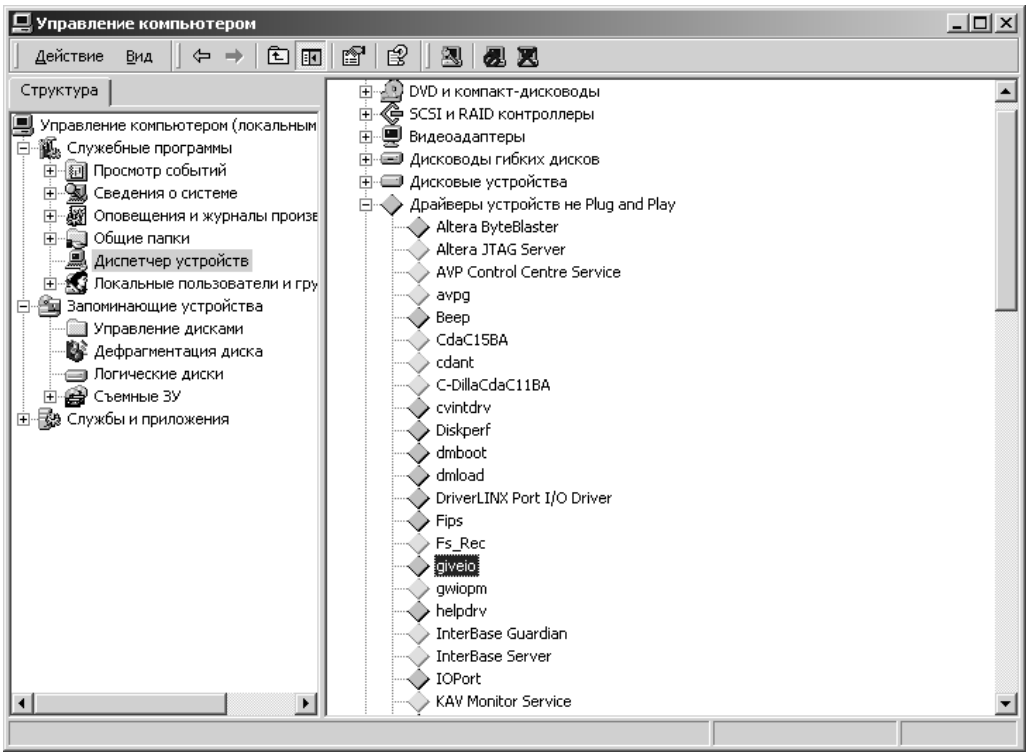


Рис. 2.108.

Далі, виділивши вузол «giveio», в контекстному меню вибрати пункт «Властивості». З'явиться діалогове вікно, показане на рис. 2.109. На закладці «Драйвер» слід натиснути на кнопку «Запустити», а в списку «Автозавантаження» вибрати пункт «Автомат». В такому разі драйвер автоматично запускатиметься при завантаженні операційної системи.

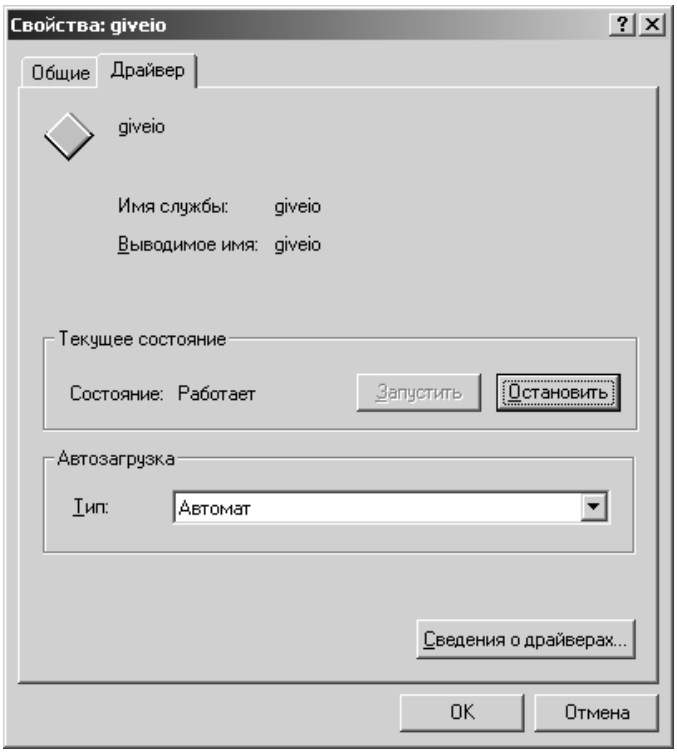


Рис. 2.109.

2. Запускаємо Visual Studio і створюємо новий проект з назвою «Stroka» на базі діалогового вікна. Редагуємо форму так, щоб вона мала вигляд, представлений на рис. 2.110.

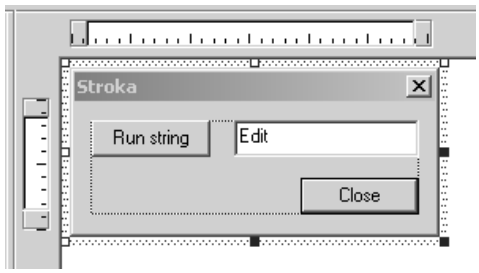


Рис. 2.110.

До текстового поля прив'язуємо змінну з ім'ям «m_edit». Також створюємо обробник події натискання на кнопку «Run string».

3. Підключаємо встановлений драйвер. Для цього у функцію OninitDialog, яка з'являється перед відображенням діалогового вікна, додаємо наступний код:

```
HANDLE h;
h=CreateFile("\\\\.\\giveio", GENERIC_READ, 0, NULL, OPEN_EXISTING,
FILE_ATTRIBUTE_NORMAL, NULL);
if(h==INVALID_HANDLE_VALUE)
{
        MessageBox("Giveio Error");
        return false;
```

```
        }
        CloseHandle(h);
```

4. Редагуємо обробник натиснення на кнопку. Функція має наступний вигляд:

```
void CStrokaDlg::OnButton2()
{
char currfont[91][6]={{0,0,0,0,0,0},{0,6,95,6,0,0},{0,0,0,0,0,0},
{0,0,0,0,0,0},{36,74,255,82,36,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,
0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,160,96,0,0},{0,0,0,0,0,0},{0,0,96,96,0,0},{0,0,0,0,0,
0},{62,65,65,65,62},{0,4,2,127,0,0},{66,97,81,73,70,0},{34,65,73,73,54,0},{56,36,34,127,6
4,0},{47,73,73,73,49,0},{62,73,73,73,50,0},{65,33,17,9,7,0},{54,73,73,73,54,0},{38,73,73,7
3,62,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0
,0,0,0,0},{124,18,17,18,124,0},{127,73,73,73,54,0},{62,65,65,65,34,0},{127,65,65,65,28,0},
{127,73,73,73,73,0},{127,9,9,9,1,0},{62,65,65,81,50,0},{127,8,8,8,127,0},{129,129,127,129
,129,0},{48,64,64,64,63,0},{127,8,20,34,65,0},{127,64,64,64,64,0},{127,2,4,2,127,0},{127,
4,8,16,127,0},{62,65,65,65,62,0},{127,9,9,9,6,0},{62,65,73,81,62,0},{127,9,25,41,70,0},{38
,73,73,73,50,0},{1,1,127,1,1,0},{63,64,64,64,63,0},{31,32,64,32,31,0},{127,64,48,64,127,0}
,{99,20,8,20,99,0},{3,4,120,4,3,0},{105,89,73,77,75,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,
0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{56,68,68,36,120,0},{127,72,72,72,48,0},{56,68,
68,68,40,0},{56,68,68,36,127,0},{56,84,84,84,88,0},{8,126,9,1,6,0},{28,162,162,162,124,0}
,{127,8,8,8,112,0},{0,68,125,68,0,0},{0,32,64,64,61,0},{127,16,40,68,0,0},{0,127,64,0,0,0},
{124,4,124,4,120,0},{124,4,4,4,120,0},{56,68,68,68,56,0},{252,36,36,36,24,0},{24,36,36,36
,252,0},{124,4,4,4,8,0},{72,84,84,84,36,0},{4,63,68,64,32,0},{60,64,64,64,124,0},{28,32,64
,32,28,0},{60,64,56,64,60,0},{68,40,16,40,68,0},{28,160,160,160,124,0},{68,116,84,92,68,0
}};
   CString s;
   m_edit.GetWindowText(s);
   for(int i=0;i<s.GetLength();i++)
   {
      for(int j=0;j<6;j++)
      {
         _outp(0x378, ~currfont[s.GetAt(i)][j]);
         _outp(0x37A, 1);
         _outp(0x37A, 0);
         Sleep(20);
      }
   }
   _outp(0x378,255);
   for(i=0;i<100;i++)
   {
         _outp(0x37A, 1);
         _outp(0x37A, 0);
         Sleep(20);
   }
}
```

Після компіляції проекту з’явится діалогове віккно, показане на рис. 2.111.



Рис. 2.111.

В текстовому полі потрібно ввести текст и натиснути на кнопку «Run string». На біжучому рядку з’явится рухомий текст (рис.2.112).

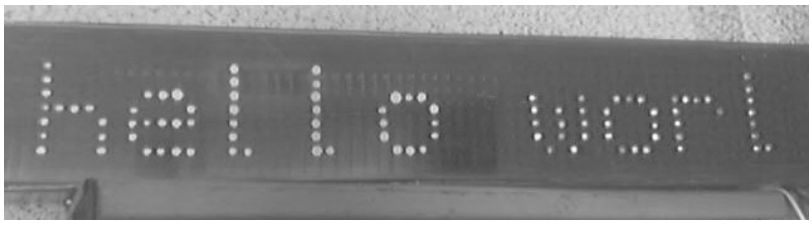
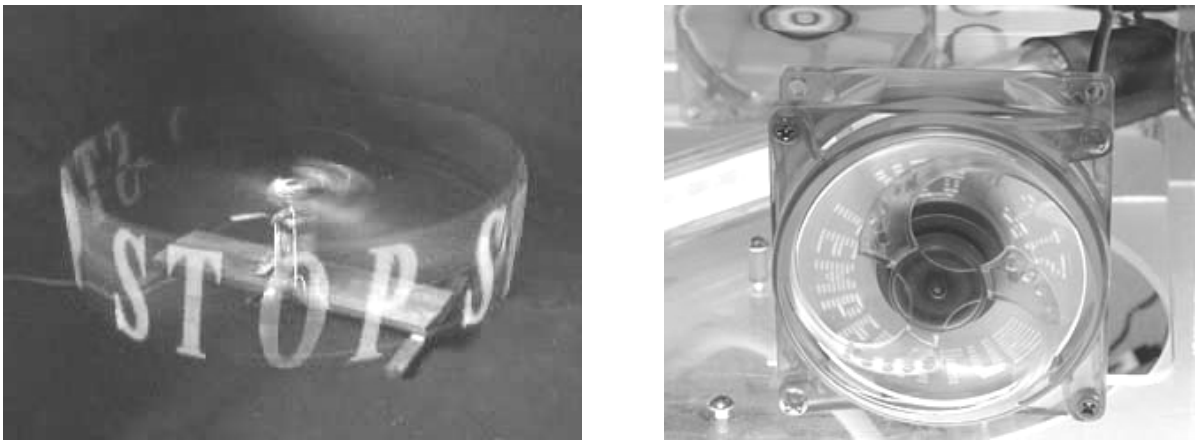


Рис. 2.112.

2.15. Системи відображення інформації з механічною розгорткою

У наш час переважна більшість пристроїв світлодіодної індикації з механічною розгорткою знайшли своє застосування в рекламних і сувенірних цілях. Переважна більшість всіх розробок працюють за принципом стаціонарного й стабільного розташування привідного двигуна, а також розраховані на стабільну швидкість обертання лінійки світлодіодів. Існує декілька типів систем з механічною розгорткою. Це системи горизонтального, вертикального, обертального та маятникового типів. Такі системи однакові за принципом відображення інформації та різні за напрямком і типом обертальної системи. На рис. 2.113, а, представлене світлодіодне табло з механічною розгорткою. На рис. 2.113, б, представлений екран з механічною розгорткою, побудований на базі вентилятора. На екрані можна відображати, наприклад, температуру процесора, на якому він встановлений, або будь-яку іншу інформацію.

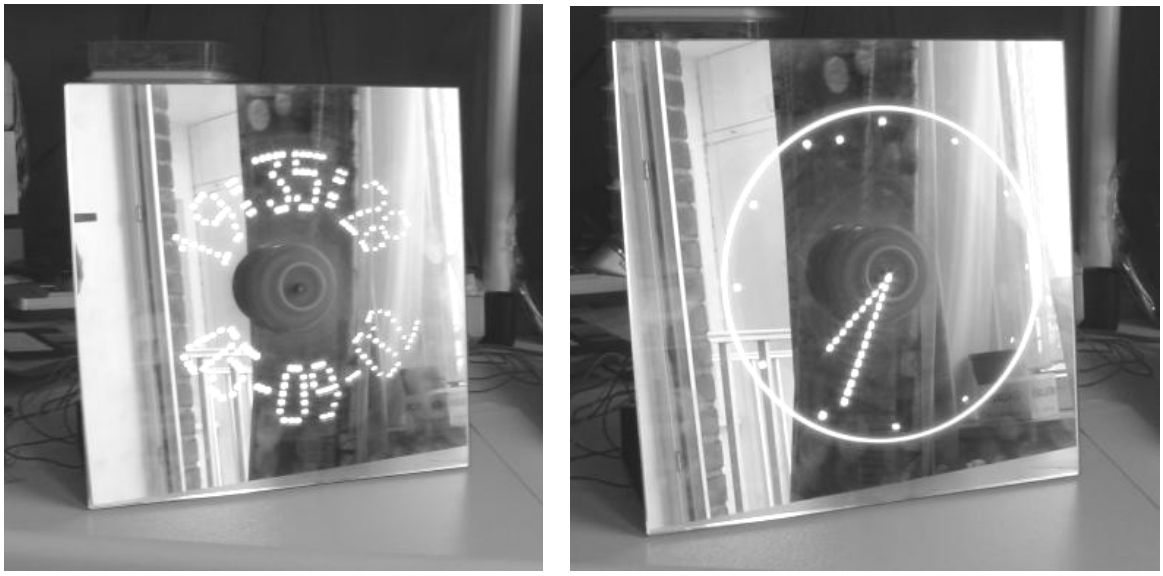


а *б*

Рис. 2.113.

Дуже популярними є світлодіодні годинники з механічною розгорткою. Екран цих годинників представляє собою важіль із установленими на ньому світлодіодами, який швидко обертається та формує зображення.

Важіль обертається із частотою близько 1500 об/хв, і діоди загоряються та вимикаються на строго визначений час. Оскільки важіль обертається з великою швидкістю, то він майже непомітний. У кожному положенні важеля світлодіоди загоряються у певній комбінації, що дозволяє формувати графічну й текстову інформацію (рис. 2.114, а). В залежності від форми важеля дисплей може бути у вигляді циліндра або диска. Прямий важіль дозволяє імітувати годинники зі стрілками (рис. 2.114, б).



а *б*

Рис. 2.114.

Системи, які використовують принцип механічної розгортки, застосовуються в більшості для того, щоб відображати статичну інформацію на елементах систем, що рухаються з великою швидкістю.

На кафедрі ТЕЕС НУК було розроблено систему відображення інформації з механічною розгорткою. Автором розробки є Бабак В.І. Структурна схема рухомої частини системи з механічною розгорткою представлена на рис. 2.115.

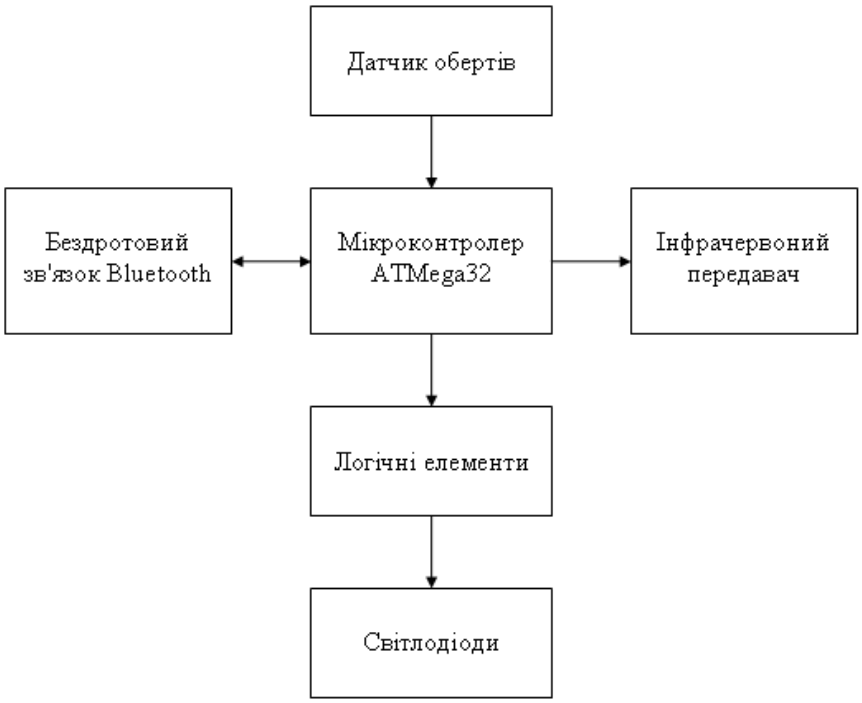


Рис. 2.115.

Мікроконтролер виконує функції збору та обробки інформації, яка потрапляє з датчика обертів, блоку бездротового зв’язку Bluetooth. Мікроконтролер формує необхідні імпульси для керуванням логічними елементами, які в свою чергу керують світлодіодами. Датчик обертів диску повинен забезпечити достатню інформативність даних як при низьких, так і при високих обертах диску. Це необхідно для того, щоб мати змогу розраховувати момент часу, коли необхідно запалювати світлодіоди. Система передачі та прийому даних об’єднані в одному пристрої – Bluetooth-модемі, який виступає в якості мосту між мікроконтролером та COM-портом комп’ютера.

Система відображення інформації має 36 RGB-світлодіодів, кожен з яких може випромінювати три кольори (червоний, зелений та синій). З цього виходить, що для керування ними знадобитися 108 виводів. Мікроконтролер ATMega32 має лише 32 лінії вводу/виводу. Тому в схемі використані додаткові зовнішні елементи – регістри.

На рис. 2.116 зображена схема рухомої частини механічної розгортки.

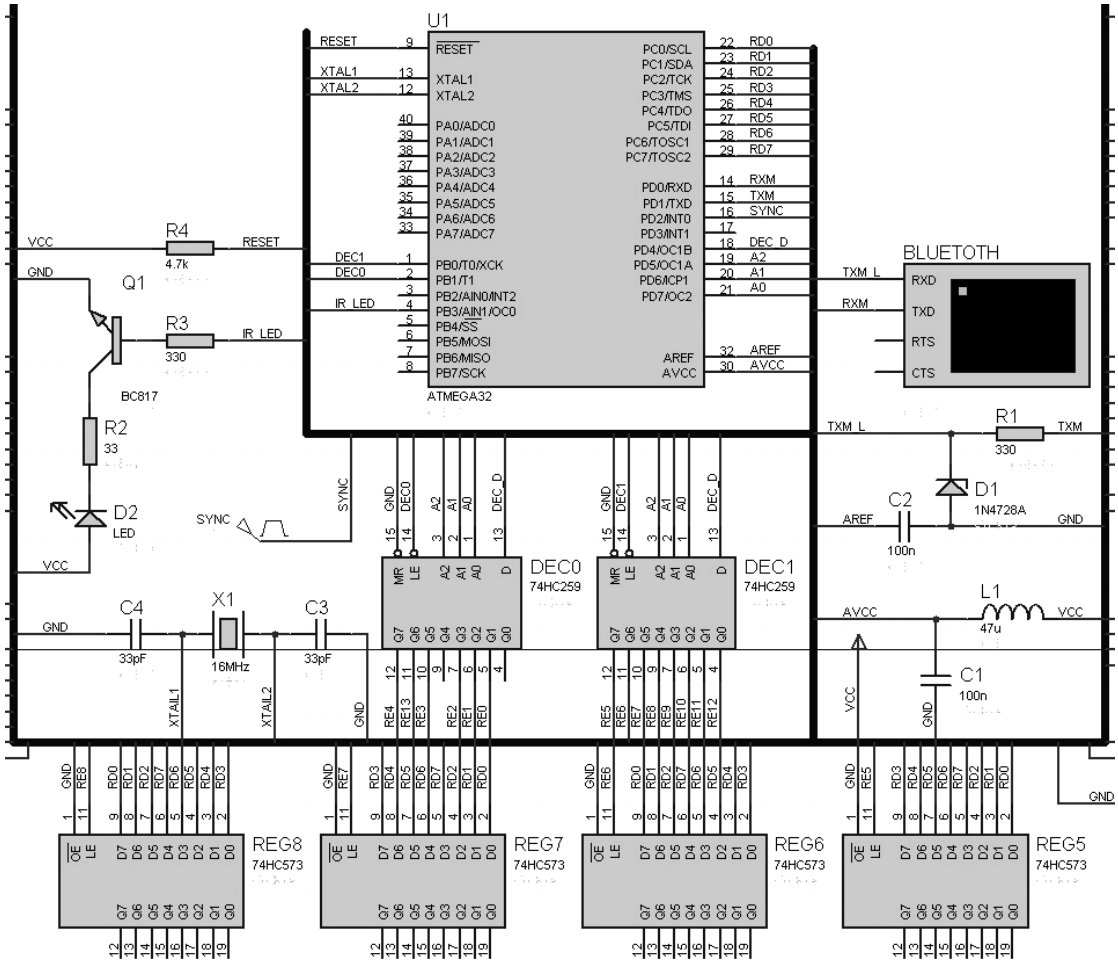


Рис. 2.116.

Дешифратори DEC0 та DEC1 демультиплексують данні, що дозволяє заощадити 6 виводів мікроконтролера. REGX – REGY – регістри, в які записуються данні з порту мікроконтролера тільки тоді, коли на їхньому вході «LE» присутній сигнал логічної одиниці. Для керування регістрами необхідно всього (8+*n*) ліній вводу/виводу, де *n* – це кількість регістрів. Транзистор Q1 виступає в якості підсилювача струму, для подальшого живлення інфрачервоного світлодіода D2, який призначений для передачі модульованого сигналу на плату керування двигуном. До регістрів підключені світлодіоди з опорами.

На рис. 2.117 приведена схема модуля управління двигуном.

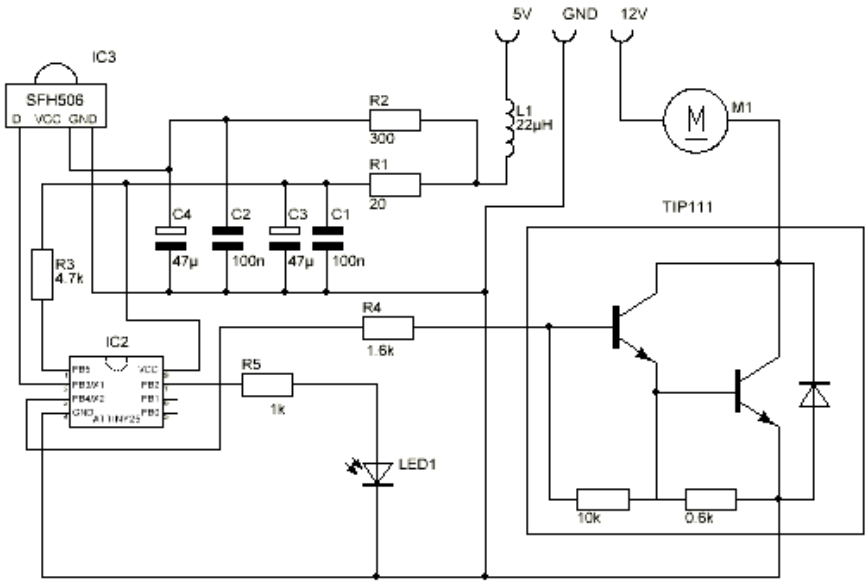


Рис. 2.117.

Приймач SFH506 демодулює сигнал протоколу RC5, який подається на вивід PB5 мікроконтролеру ATTiny45. Мікроконтролер ATTiny45 оброблює дані, прийняті з SFH506, та формує необхідну скважність імпульсів для регулювання швидкості двигуна.

В результаті моделювання була отримана осцилограма сигналів (рис. 2.118), що формує зображення.

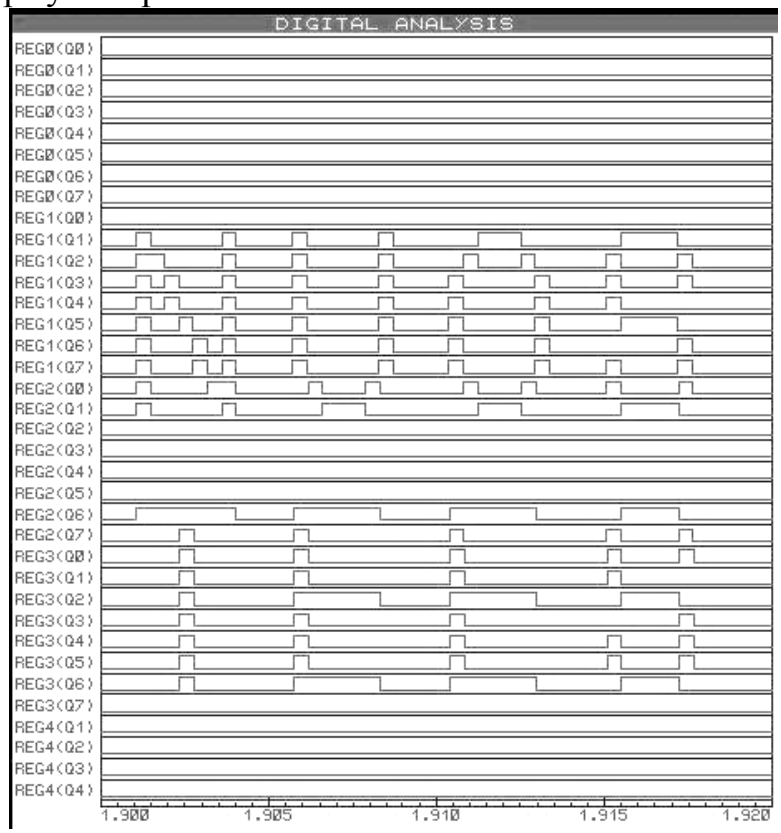


Рис. 2.118.

Вигляд друкованої плати представлений на рис. 2.119.

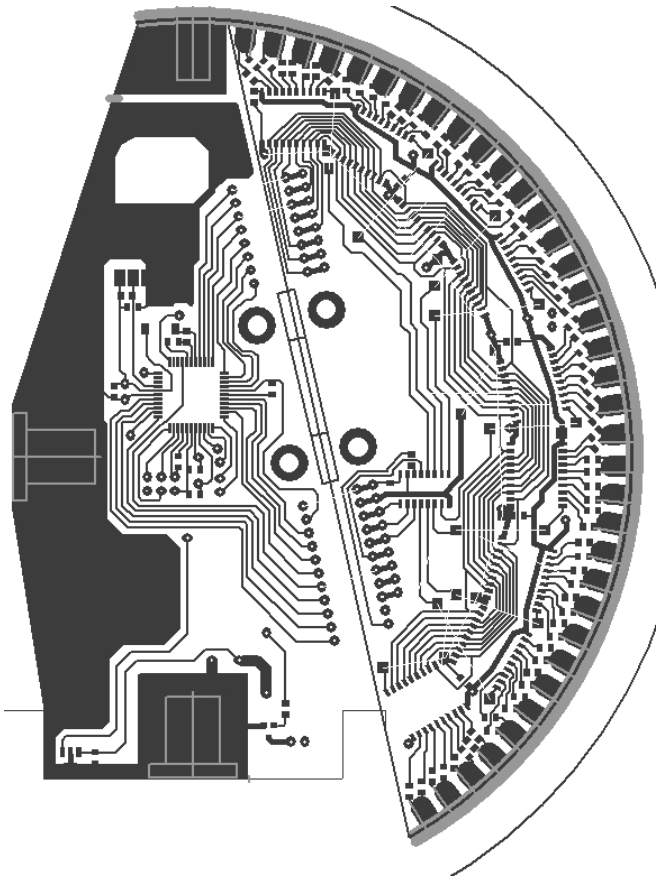


Рис. 2.119.

Виготовлена плата рухомої частини механічної розгортки зображена на рис. 2.120.

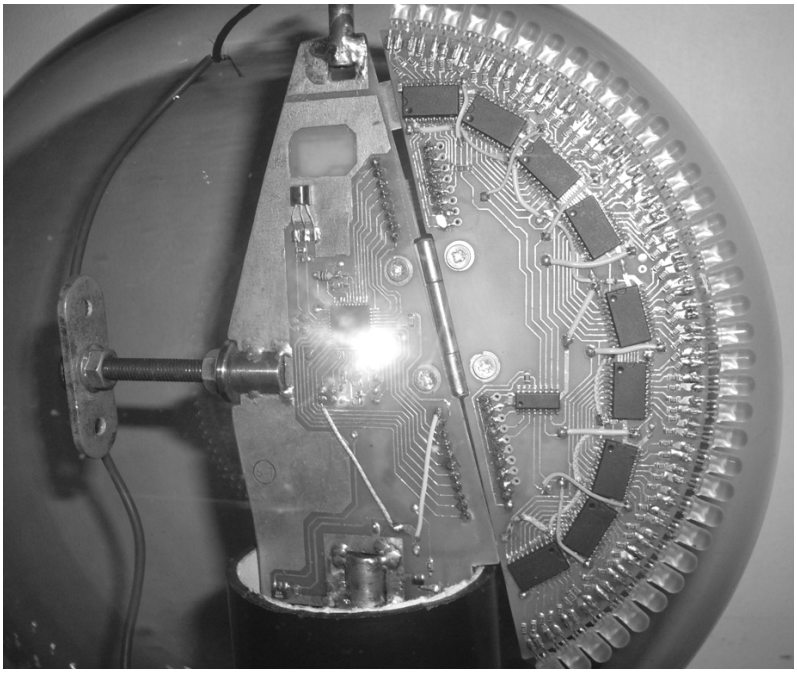
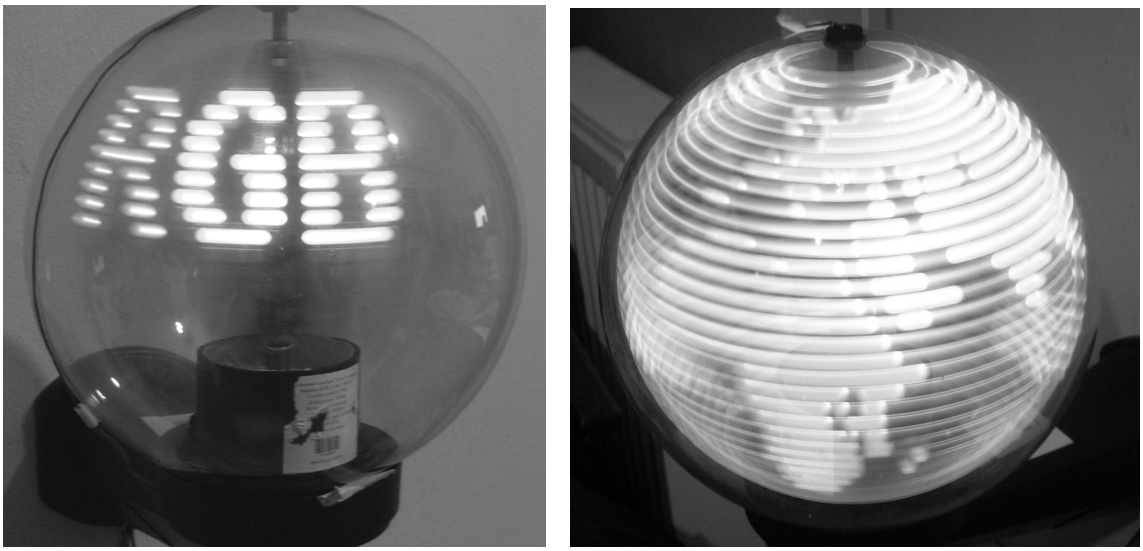


Рис. 2.120.

Результати тестування мікропроцесорної системи курування механічною розгорткою наведені на рис. 2.121.



а *б*

Рис. 2.121.

Програмне забезпечення даного проекту можна розподілити на три основні частини, які представляють собою три окремі програми.

Перша частина – це програма для персонального комп'ютера, в функції якої входить установка режиму роботи (ефекти/текст/зображення), завантаження тексту в EEPROM мікроконтролера, керування двигуном.

Друга частина – програма для мікроконтролерного пристрою, який керує двигуном.

Третя частина – це програма для мікроконтролера, що безпосередньо знаходиться на рухомій частині пристрою і відповідає за формування зображення на диску під час механічної розгортки. Також цей мікроконтролер забезпечує формування даних по протоколу RC5, які потрібні для керування двигуном. Під час роботи пристрою програма відслідковує появу сигналу з датчика холла, що відповідає початку оберту диска.

Питання до розділу 2

1. Види та характеристики світлодіодів.
2. Основні схеми включення світлодіодів.
3. Опишіть принципи динамічної індикації.
4. Одиничні та багатоелементні напівпровідникові знакосинтезуючі індикатори.
5. Використання знакосинтезуючих індикаторів в пристроях.
6. Різновиди матричних індикаторів.

7. Способи керування яскравістю світлодіодів та індикаторів.
8. Яким чином здійснюється виведення на індикатори дійсних чисел?
9. Принцип програмного регулювання яскравості семисегментних індикаторів.
10. Особливості управління семисегментними індикаторами при їх підключенні на загальну шину.
11. У чому полягає відмінність в управлінні семисегментними індикаторами з загальним анодом і загальним катодом?
12. Яким чином можна керувати сегментами індикатора, якщо номінальний струм перевищує максимально допустимий струм виводу мікроконтролера?
13. Розрахунок часу регенерації для динамічного методу відображення.
14. Обгрунтування необхідності застосування різних методів відображення.
15. Схемотехнічні рішення для побудови схем відображення інформації.
16. Схеми включення одиничних індикаторів.
17. Опишіть принцип роботи біжучого рядка.

Задачі до розділу 2

1. Реалізувати ефект «гусениця» на світлодіодних індикаторах.
2. Створити ефект «хвиля» на семисегментних індикаторах.
3. Реалізувати біжучий рядок на семисегментних індикаторах. Виводити наступний текст: 123456789AbCdEF.
4. Відображати по черзі на матричному індикаторі числа від 0 до 9, дублювати ці числа на семисегментному індикаторі.
5. Реалізувати біжучий рядок на матричному індикаторі. Виводити своє ім'я.
6. Написати програму, яка послідовно запалює всі елементи матричного індикатора, а потім також послідовно, один за іншим їх гасить.
7. Реалізувати таку задачу: по контуру матричного індикатора рухається змійка (довжина 5 точок).
8. Реалізувати 5 світлових ефектів: 3 на світлодіодах, по 1 на семисегментному та матричному індикаторі. Вид ефекту вибирається натисканням відповідних кнопок.

3. БУДОВА І ПРИНЦИПИ РОБОТИ РІЗНИХ ТИПІВ ІНДИКАТОРІВ

3.1. Електролюмінесцентні індикатори

Явище передпробійної електролюмінесценції в даний час застосовується в плівкових джерелах світла: порошкових і тонкоплівкових, які керуються постійною та змінною напругами відповідно. Перші промислові зразки електролюмінесцентних індикаторів (ЕЛІ) на основі порошкоподібних люмінофорів були отримані в кінці 50-х років. Вони застосовуються до теперішнього часу в джерелах світла різного призначення як електролюмінесцентні панелі: світильники, світлові вказівники, алфавітно-цифрові індикатори, мнемосхеми та ін.

3.1.1. ЕЛІ постійного струму

Порошковий світловипромінюючий елемент є багатошаровою структурою на прозорій основі: скляній пластині або полімерній плівці. Джерело випромінювання складається з прозорого електроду на основі, люмінесцентного шару і другого непрозорого електроду. Випромінювання виходить через прозорий електрод і основу. Люмінесцентний шар складається з порошкоподібного люмінофора з органічним або неорганічним сполучником. Товщина люмінесцентного шару складає 30-100 мкм, вміст сполучника для випромінювачів, що збуджуються змінною напругою, складає 30-50%, а для випромінювачів, що збуджуються постійним струмом, складає 5-25%. Ці випромінювачі відрізняються також розміром частинок люмінесцентних порошкоподібних матеріалів: люмінофори, що збуджуються змінною напругою, мають середній розмір частинок від 1 до 10 мкм, а люмінофори для постійного струму більш дрібнозернисті – від 0,2 до 2 мкм. На рис. 3.1 представлена конструкція порошкового електролюмінесцентного індикатора.

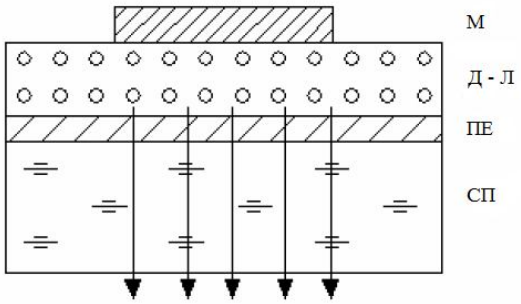


Рис. 3.1.

На рисунку позначено: М – електрод; Д-Л – люминесцентний шар з діелектричним сполучником; ПЕ – прозорий електрод; СП – скляна підложка.

В основному в якості люмінофорів використовуються порошки сульфіду цинку і кадмію, а також сульфоселенідні з'єднання. Останніми роками розроблені люмінесцентні матеріали на основі сульфідів кальцію і

стронцію. Як активатори використовуються мідь, марганець, галій, срібло, свинець та інші речовини, які вводяться в люмінофори в кількості від одиниць до сотих доль відсотка. Присутність міді в цих люмінофорах є обов'язковою, оскільки вона утворює окрему фазу (сульфід міді) на поверхні зерен. Для утворення окремої фази мідь вводять в кількостях, що перевищують її межу розчинності в матеріалі люмінофора. Контакт цих фаз призводить до появи на поверхні зерен люмінофора гетеропереходів, обумовлюючих концентрацію електричного поля в поверхневих бар'єрах. Легування міддю приводить до збільшення яскравості випромінювання і підвищення ефективності електролюмінесценції.

Колір світіння люмінофорів визначається типом і кількістю домішок, що вводяться. Промисловістю випускаються люмінесцентні матеріали, які випромінюють у всьому видимому діапазоні. Найбільш високі значення яскравості отримані для люмінофорів на основі сульфіду цинку, легованого марганцем, з жовтим кольором світіння. Перспективними вважаються люмінофори на основі сульфідів лужноземельних металів, легованих галогенідами рідкоземельних елементів. Відповідні електролюмінесцентні склади позначаються ЕЛ, ЕЛС-455, 510, 670 та ін., де число показує довжину хвилі в нм, що відповідає максимальній інтенсивності випромінювання.

В якості сполучника в люмінесцентному шарі використовуються діелектричні лаки на основі епоксидних смол (ЕП-96), цианетилцелюлози, цианетилового ефіру полівінілового спирту (ВС-530) та ін., а також склоемалевий сполучник (цинковоборосилікатне скло).

Світлотехнічні характеристики випромінювачів залежать від конструктивно-технологічних і експлуатаційних чинників. До конструктивно-технологічних відносять тип люмінофора, розмір зерен порошку, тип сполучника, товщину і вміст люмінесцентного шару та ін. До експлуатаційних чинників відносять амплітуду і частоту збуджуючої напруги, температуру, вологість середовища та ін.

Характеристика вольт-яскравості електролюмінесцентних індикаторів зазвичай апроксимується степеневою залежністю від напруги і частоти.

Збудження світла в порошкових випромінювачах виробляється напругою величиною 50-300 В, що змінюється з частотою 50-5000 Гц. Яскравість випромінювання електролюмінесцентних панелей складає від 5 до 500 кд/м^2. Інтервал робочих температур лежить в межах від –60 до 50 ^{0}С. Подальше зростання температури призводить до зниження яскравості світіння. Вітчизняною промисловістю випускаються електролюмінесцентні індикатори типу ІТЕЛ, ІЕМ, МЕЛ та інші, що використовуються як джерела світла, цифрових індикаторів, мнемонічних і графічних малоформатних екранів. Основною перевагою порошкових електролюмінесцентних індикаторів є можливість отримання великих площ світіння і твердотіла конструкція. До недоліків порошкових

електролюмінесцентних індикаторів відносять обмежені значення яскравості і нетривалий термін служби, який багато в чому обумовлений дифузією міді та інших домішок. Незважаючи на дослідження деградаційних явищ в порошкоподібних світловипромінювальних структурах, довговічність приладів не перевищує декількох тисяч годин.

3.1.2. Тонкоплівкові ЕЛІ

Дослідження тонкоплівкових джерел випромінювання було розпочато дещо пізніше, ніж вивчення порошкових люмінофорів. Довгий час плівки розглядалися в основному як зручний об'єкт для дослідження фізики електролюмінесценції. Інтерес до тонкоплівкових електролюмінесцентних індикаторів різко зріс після того, як в 1974 р. дослідниками японської фірми "Sharp" був представлений перший тонкоплівковий плоский екран, що має високі функціональні характеристики. Успіхи в технології отримання тонких плівок, а також розробка структур, в яких шар люмінофора завтовшки близько 1 мкм розташовано між двома тоншими шарами діелектрика, дозволили створити випромінювачі, що збуджуються змінною напругою, які мають вищу яскравість і термін служби в порівнянні з порошковими.

Тонкоплівкові електролюмінесцентні структури типу МДЛДМ (прозорий електрод – діелектрик – люмінофор – діелектрик-метал) зазвичай складаються з п'яти послідовно нанесених на скляну пластину шарів з використанням методів тонкоплівкової технології. Така структура представлена на рис. 3.2.

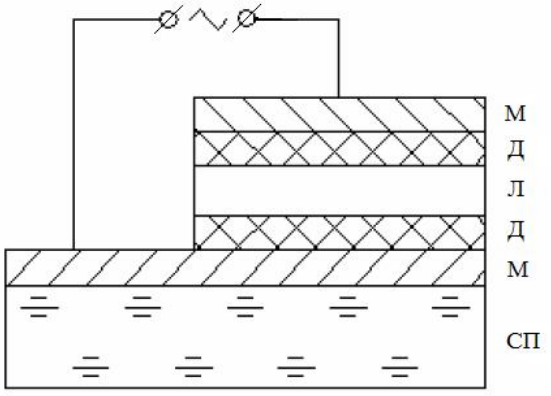


Рис. 3.2.

На рисунку позначено: М – електрод; Д – діелектрична плівка; Л – люмінесцентний шар; СП – скляна пластина.

Товщина плівок провідників складає 0,1-0,2 мкм, діелектричних плівок – 0,2-0,5 мкм, плівок люмінофора – 0,5-1,5 мкм. Окрім перерахованих плівок до складу електролюмінесцентного індикатора можуть входити додаткові шари, наприклад, світлопоглинаючого контрастного матеріалу. Крім того, кожен з функціональних шарів може бути неоднорідним, наприклад, складатися з декількох плівок. Прозорим

може бути і другий електрод, тоді контрастний шар розміщується за межами тонкоплівкової структури і може поєднуватися за своїми функціями із захисним покриттям. В якості основи і одночасно першого діелектрика може бути використана керамічна пластина, тоді верхній електрод має бути прозорим.

В якості люмінесцентних шарів зазвичай використовують сульфід цинку, легований марганцем або фторидами рідкоземельних елементів. Перспективними люмінофорами показали себе сульфіди стронцію і кальцію, леговані фторидами рідкоземельних металів. Всі використані люмінофори є широкозонними напівпровідниками з високим питомим опором, в яких легуючі домішки утворюють центри світіння, що мало впливають на електричні властивості матеріалів люмінофора. Кристалічна структура люмінесцентних плівок залежить від їх хімічного складу і умов отримання. В більшості випадків ці плівки є полікристалічними із стовпчастою формою кристалітів, поздовжні розміри яких порядку товщини плівки, а поперечні в залежності від температури основи лежать в межах від сотих до десятих частин мікрометра. В якості основи при отриманні структур використовуються пластини безлужного термостійкого скла товщиною 2-3 мм з нанесеним шаром прозорого електроду, для створення якого застосовують сильно леговані оксиди індія або стануму, що є виродженими широкозонними напівпровідниками. Для здобуття непрозорих електродів використовують алюміній, індій, станум, тугоплавкі метали. Інколи для збільшення провідності тонких електродів в матричних панелях, на прозорі електроди наносять вузькі смужки металів з більшою електропровідністю. Для створення діелектричних плівок в електролюмінесцентних структурах застосовують оксиди кремнію, алюмінію, ітрію та рідкоземельних металів, нітрид кремнію та ін., їх композиції, сегнетоелектричні матеріали. Оскільки ці плівки працюють в умовах сильних електричних полів, до їх електричних параметрів висувають високі вимоги. Діелектричні шари, що використовуються, повинні мати високі електричні і технологічні властивості, стабільність і відтворюваність. Наявність двох діелектричних шарів, що відокремлюють плівку люмінофора від електродів, перетворює даний пристрій на електролюмінесцентний конденсатор, що визначає необхідність використовувати змінну напругу для збудження електролюмінесценції. Для МДЛДМ-структур з малою щільністю вільних носіїв заряду в люмінофорі прикладена напруга розподіляється між плівками люмінофора і діелектриків відповідно до значень їх геометричних ємностей. Таким чином, чим більша ємність діелектриків, тим більша частина загальної напруги падає на шарі люмінофора. Головна роль діелектричних шарів полягає в обмеженні заряду, що проходить через люмінофор в робочих режимах. Необхідність застосування таких шарів обумовлена фізичною природою процесу перенесення заряду в люмінесцентній плівці. Електролюмінесценція в тонких шарах сульфіду цинку та інших

матеріалах пов'язана з електричним пробоєм напівпровідника. При малих електричних полях матеріал має дуже низьку провідність, але при деякій пороговій напруженості поля, яка складає від $5 \cdot 10^5$ до $2 \cdot 10^6$ В/см, одночасно спостерігається прискорення зростання струму і виникнення люмінесценції. Невеликі неоднорідності полікристалічних люмінесцентних плівок можуть призвести до того, що після прикладення високої напруги виникнуть області зі швидким наростанням струму, які увійдуть до режиму постійного короткого замикання, і струм може досягти величин, достатніх для руйнування структури. При збудженні пульсуючою або змінною напругою структур з ізолюючими шарами, через люмінесцентний шар проходить лише обмежений заряд. В цьому випадку нагрівання плівки люмінофора обмежується до прийнятного рівня і катастрофічного руйнування не відбувається. Електричний пробій люмінесцентного шару носить зворотній характер і супроводжується електролюмінесценцією. Інша важлива роль діелектричних шарів полягає у формуванні поляризаційного заряду на межі розділу люмінофор – діелектрик. За мірою накопичення зарядів на протилежних сторонах люмінесцентного шару, створене ними поляризаційне поле зменшує напруженість електричного поля в люмінофорі і пробій припиняється. Поляризаційне поле підсилює дію наступного імпульсу напруги протилежної полярності. При подачі на електролюмінесцентний індикатор імпульсу напруги протилежної полярності поляризаційне поле додається до зовнішнього поля, тому збудження світіння в цих індикаторах виконується змінною напругою.

Основною характеристикою електролюмінесцентних джерел світла є залежність яскравості випромінювання від прикладеної напруги. На рис. 3.3 представлені типові характеристики тонкоплівкових електролюмінесцентних структур на основі сульфіду цинку, легованого марганцем, виміряні для різних частот змінної напруги. Ці залежності є лінійними при малій напрузі і частотах, з підвищенням цих параметрів графіки залежності є сублінійними. Загальноприйнято зображати графіки характеристик в напівлогарифмічному масштабі, тобто по осі абсцис напругу позначають в лінійному масштабі, а по осі ординат показують яскравість в логарифмічних координатах. Використання логарифмічного масштабу для яскравості випромінювання пов'язане з суб'єктивністю сприйняття її величини органами зору людини. Збільшення яскравості джерела світла на порядок людина сприймає як збільшення в 2 рази.

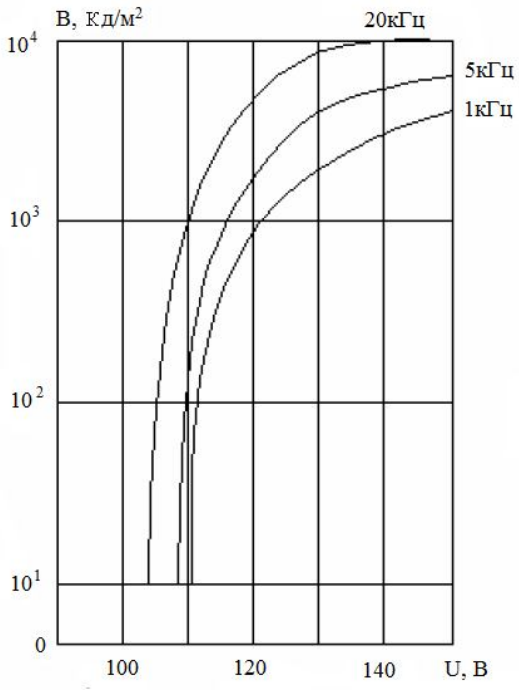


Рис. 3.3.

Характеристика тонкоплівкових випромінювачів має пороговий характер. Зростання яскравості відбувається коли напруга перевищує деяке значення. Величина порогової напруги визначається, в першу чергу, властивостями матеріалу люмінофора, а саме – пороговою напруженістю електричного поля, а також величиною і співвідношенням ємностей люмінесцентного і діелектричного шарів. Значення порогової напруженості електричного поля в плівках люмінофора, ізольованих від електродів, визначається механізмом генерації вільних носіїв заряду і складає приблизно 105 В/см.

Для забезпечення надійності функціонування і збільшення довговічності тонкоплівкових електролюмінесцентних індикаторів необхідна їх ретельна герметизація. Герметизація конструкцій індикаторних пристроїв здійснюється найчастіше наступними способами. У першому варіанті використовуються вакуумні скляні корпуси, заповнені інертним газом або діелектричною рідиною. У другому варіанті шар герметика наноситься безпосередньо на поверхню тонкоплівкової структури. Другий варіант герметизації має безперечні переваги, що полягають в простішому технологічному процесі герметизації, можливості одночасного використання герметика в якості контрастного шару, кращих масогабаритних показниках герметизованого індикатора, стійкості до механічних дій. Для герметизації тонкоплівкових електролюмінесцентних індикаторів застосовуються різні неорганічні і органічні матеріали, найчастіше кремнійорганічні, смоли, лаки, клеї, масла і компаунди.

Область поля індикатора, що світиться, визначається розмірами, формою і взаємним розташуванням верхніх і нижніх електродів. Світіння люмінофора виникає в місцях перетину цих електродів в площині поверхні

індикаторів. Залежно від форми і розташування світловипромінюючих елементів розрізняють мнемонічні, знакові, шкальні і матричні індикатори.

Мнемонічні індикаторні пристрої показують фігури, що світяться, простої або складної форми. Знакові індикатори відображають різні букви, цифри та інші символи. Шкальні індикатори зазвичай використовуються у вимірювальних приладах для безперервної реєстрації рівня сигналу. Матричні індикаторні пристрої є найбільш складними і дозволяють відображати найрізноманітнішу інформацію. Найбільш високі вимоги висуваються до матричних індикаторів великого формату або екранів, призначених для відображення динамічної інформації на статистичному фоні.

На практиці застосовують частоти більше 4 кГц. На низьких частотах (менше 0,4 кГц) ємність, опір (активний і повний) значно збільшуються. Починаючи з 5 кГц подальше збільшення частоти не викликає помітної зміни електричних параметрів: $C_{елі}$, $R_{елі}$, $Z_{елі}$. Тангенс кута діелектричних втрат tgδ у діапазоні частот 0,1÷10 кГц майже не змінюється. Подальше підвищення частоти викликає зростання tgδ. Споживана потужність визначається втратами в діелектрику та електролюмінофорі. Приблизний розрахунок споживаної потужності оцінюється наступним виразом:

$$P=0{,}125\cdot 10^{-300/U}\cdot F,$$

де U – напруга збудження, В; F – частота, кГц; P – потужність, мкВт/см^2.

Із зростанням частоти і амплітуди збуджуючої напруги збільшується споживана потужність. Активна складова споживаної потужності ЕЛІ невелика, і в розрахунках не враховується. Наприклад, при напрузі U=220В і частоті f=0,4 кГц активна складова потужності приблизно дорівнює 30 мВт/см^2. Але саме активна складова потужності перетвориться в світлове випромінювання. Розрахунок джерел живлення для ЕЛІ ведеться за величиною реактивної складової. Для ЕЛІ між струмом і напругою характерна лінійна залежність, зі зростанням частоти величина споживаного струму збільшується. При конструюванні схем управління сегмент ЕЛІ розглядають у вигляді ємності з втратами, величина якої при напрузі, приблизно рівною 220В, складає менше 200 пФ/см^2. Повна та спрощена еквівалентні схеми ЕЛІ представлені відповідно на рис. 3.4, а, б.

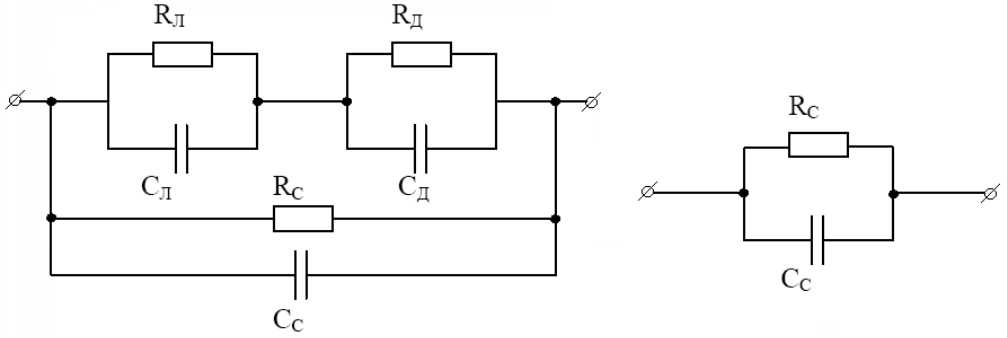
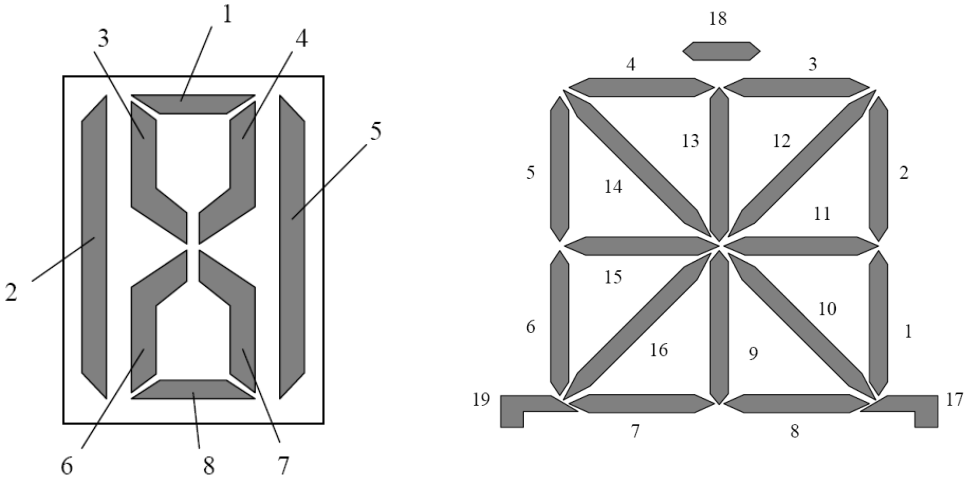


Рис. 3.4.

Величини $R_л$ і $C_л$ означають загальний опір і сумарну ємність часток люмінофора; $R_д$ і $C_д$ відповідають загальному опору і ємності прошарків діелектрика між часточками; $R_с$ і $C_с$ – опір і ємність наскрізних ділянок діелектрика відповідно (опір і ємність сегмента).

3.1.3. Сегментні ЕЛІ

Символи формуються з окремих елементів – сегментів, кількість яких може бути різною (рис. 3.5, а – 8-сегментний, б – 19-сегментний).



а *б*

Рис.3.5.

Найбільшого практичного поширення набув семисегментний індикатор, що синтезує арабські цифри.

Простий матричний екран (рис. 3.6) представляє собою шар люмінофора між двома системами вузьких паралельних шин електродів, одна з яких є прозорою.

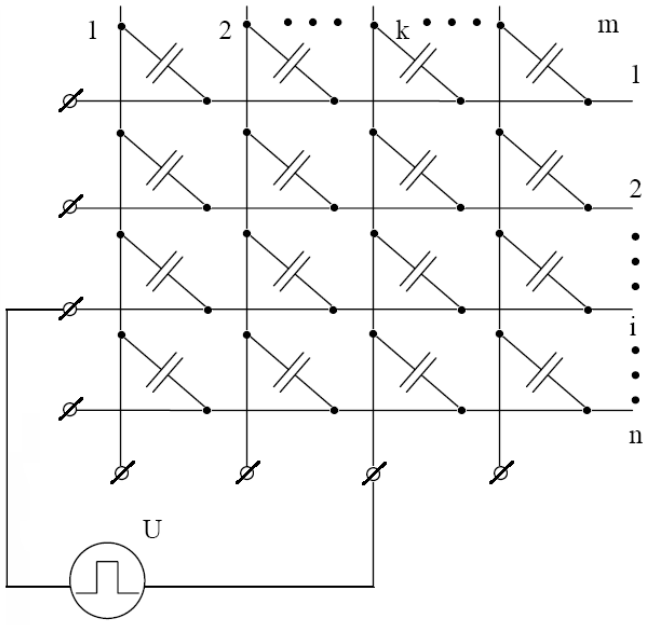


Рис. 3.6.

У місцях перетину електродів утворюються електролюмінесцентні конденсатори, що складають прямокутний растр з елементарних комірок. Для збудження комірки матричного екрану подається напруга на ті електроди, між якими вона розташована.

При прикладенні до обраної комірки Z_{ik} напруги збудження, до необраних (m–1)-м комірок, що приєднані до i-го рядка, прикладено напругу:

$$U_{ряд}=U(n-1)/(m+n-1),$$

а до необраних (n–1)-м комірок, що приєднані до k-го стовпцю, прикладено напругу:

$$U_{ст}=U(m-1)/(m+n-1).$$

до (n–1)·(m–1) комірок прикладено напругу:

$$U_{ост}=U/(m+n-1).$$

Ці напруги викликають паразитне підсвічування необраних комірок і як наслідок – зниження контрастності. Збільшення контрастності досягається ускладненням конструкції, наприклад застосуванням електролюмінесцентних підсилювачів зображень і додаткового шару нелінійного опору.

Існують багатоколірні електролюмінесцентні індикатори. Електролюмінофори різних кольорів наносяться на прозорі гребінчасті елементи так, що кожен з них утворює рядковий растр. Зверху наноситься шар діелектрика і металеві електроди. Світіння спостерігається у вигляді паралельних ліній. При великій щільності рядків створюється враження рівномірного світіння індикатора. Еквівалентна схема растрового індикатора представлена на рис. 3.7.

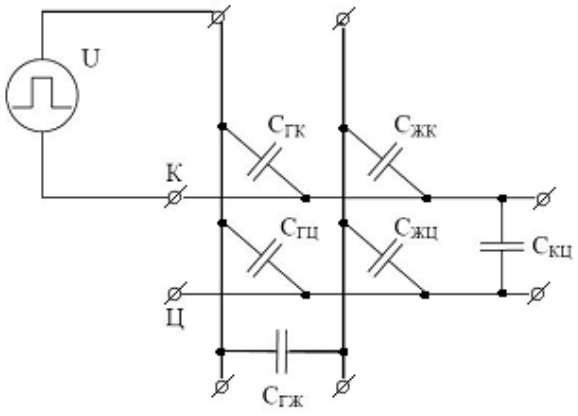


Рис. 3.7.

Ємності $C_{бк}$ (наприклад, блакитний контур), $C_{бц}$ (блакитний центр), $C_{жк}$ (жовтий контур), $C_{жц}$ (жовтий центр) – робочі, ємності $C_{бж}$ (між блакитним і жовтим растром), $C_{кц}$ (між електродами контура і центру) – паразитні.

Для збудження ЕЛІ необхідно комутувати змінні напруги з великими амплітудами, що лежать в межах 250 – 350 В. Із напівпровідникових приладів для цієї мети можна використовувати тиристори, польові транзистори і оптрони. Проте істотним недоліком тиристорів є великі струми утримання. Тому, для управління ЕЛІ зручно використовувати оптрони, які дозволяють комутувати порівняно високу напругу при малих робочих струмах, забезпечуючи гальванічну розв'язку цифрової частини схеми управління від високовольтного джерела живлення індикатора. Принципова схема пристрою формувача для одного елементу індикатора зображена на рис. 3.8.

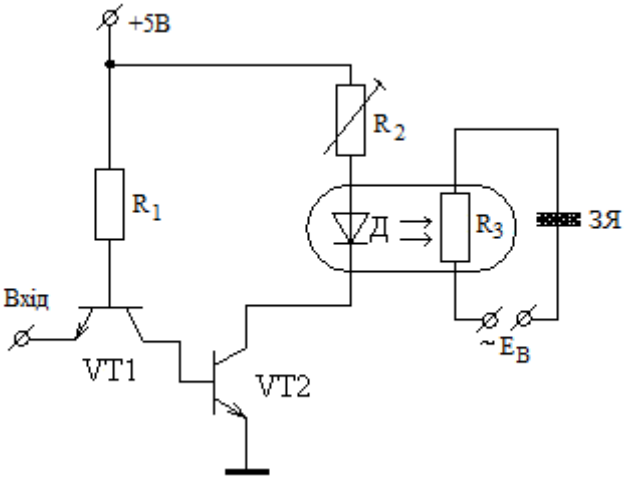


Рис. 3.8.

Вибір фоторезистора в якості комутуючого елемента обумовлений не лише хорошим взаємним узгодженням його характеристик з ЕЛІ, але і тим, що завдяки інерційності перемикання індикатора відбувається плавно, що підвищує термін служби.

3.2. Вакуумні люмінесцентні індикатори

Вакуумні люмінесцентні індикатори (ВЛІ) випускаються в циліндричних і плоских балонах. Перші бувають як однорозрядними, так і багаторозрядними, другі – лише багаторозрядними. На рис. 3.9 представлені зображення вакуумних люмінесцентних індикаторів.

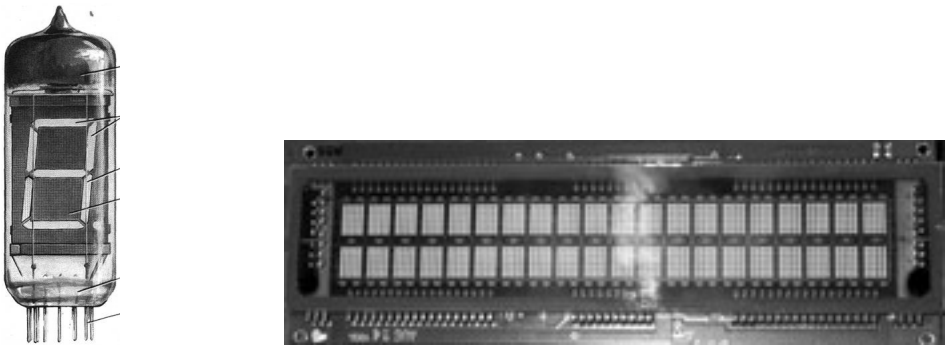


Рис. 3.9.

Принцип дії ВЛІ заснований на використанні явища люмінесценції, що виникає в катодолюмінофорах при збудженні їх електронним пучком. У ВЛІ має місце низьковольтна люмінесценція.

Катодолюмінесценція виникає при досягненні електронами визначеної енергії eU_L, де U_L – потенціал початку катодолюмінесценції. Є ряд речовин, в яких потенціал початку катодолюмінесценції складає одиниці вольт.

Люмінофор для ВЛІ має задовольняти ряд потреб:

1. Ширина забороненої зони dW – не більше 3 – 4 еВ. Інакше умовний квантовий вихід стає дуже малим.

2. Висока електропровідність. Згідно з оцінками, опір шару не повинен перевищувати одиниць кОм.

3. Низький потенціал початку катодолюмінесценції. Навіть при малому опорі шару люмінофора він виявляється непридатним для використання у ВЛІ, якщо U_L = 10–12 В.

4. Низька світловіддача. При тривалому бомбардуванні люмінофора яскравість його світіння змінюється, причому в цьому процесі можна виділити три етапи: початкова зміна, етап стабільної яскравості та етап вираженого старіння.

Перший етап викликаний встановленням стаціонарного стану поверхні люмінофора. Критерієм тривалості другого етапу є зниження яскравості до 50–70% від початкового значення. Яскравість світіння на цьому етапі зменшується у зв'язку з дією різних хімічних процесів в люмінофорі.

Основа однорозрядного ВЛІ – скляна або керамічна основа, на якій закріплені всі інші деталі індикатора (рис. 3.10). У поглибленнях основи, виконаних у вигляді сегментів, знаходиться провідний шар, сполучений з контактами. Кожен сегмент має окремий вивід. Провідні шари сегментів повністю вкриті люмінофором. На передній стороні основи, у напрямі зчитування, встановлюється плоский металевий електрод. Отвори в цьому електроді розташовані навпроти відповідних сегментів, покритих люмінофором. На невеликій відстані від екрануючого електроду розташована управляюча сітка. У свою чергу, на малій відстані від

площини сітки, приблизно паралельно вісі лампи, розташований прямонакальний оксидний катод. Вся ця система міститься в циліндричній скляній колбі, яка всередини покрита прозорим провідним шаром. На рис. 3.10 представлена конструкція вакуумного люмінесцентного індикатора.

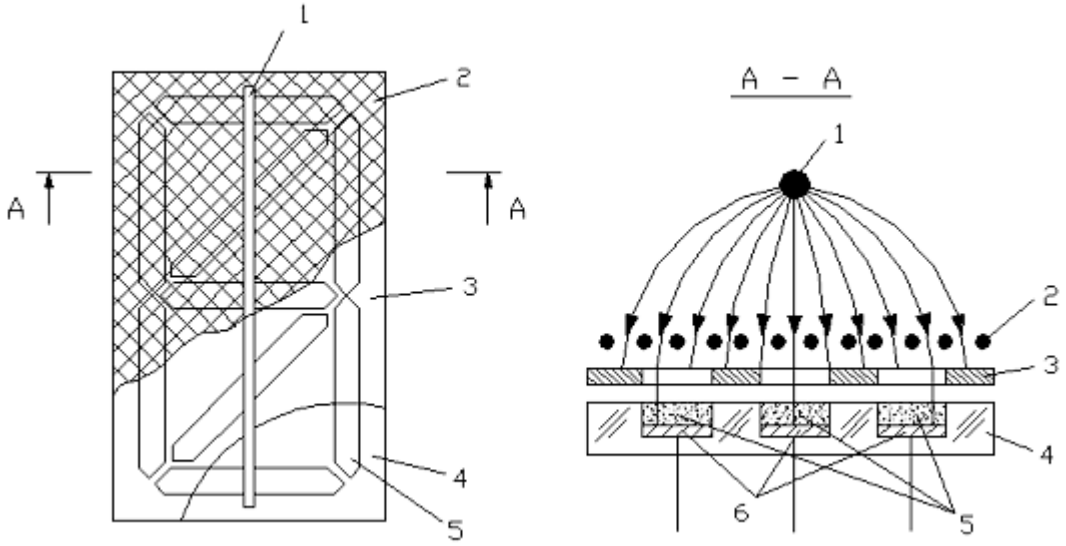


Рис.3.10.

На рисунку позначено: 1 – катод; 2 – сітка; 3 – екран; 4 –основа; 5 – люмінофор; 6 – анод

У вихідному стані для надійного запирання електронного струму і запобігання небажаного світіння люмінофора, до сітки прикладається негативна напруга зміщення – декілька вольт відносно катода.

При позитивній напрузі на управляючій сітці електрони прискорюються у напрямку анодних сегментів. Задача управляючої сітки полягає ще в тому, щоб забезпечувати більш рівномірний розподіл щільності потоку електронів на поверхні анода індикатора. Екрануючий електрод має той самий потенціал, що і управляюча сітка. Електрони потрапляють на сегменти, які мають в даний момент позитивний потенціал; виникає низьковольтна катодолюмінесценція – нанесений на сегмент (анод) люмінофор починає світитися. Яскравість світіння, в залежності від застосованого люмінофора, досягає значень 300...700 кд/м^2 і більше.

Розвитком циліндричного ВЛІ стала конструкція індикатора в плоскому балоні, устрій якого показаний на рис. 3.11.

Окрім 7-сегментних плоских ВЛІ розроблені також 14-сегментні індикатори, знакомісце якого виконане у вигляді точкової матриці 5×7 або 7×12 елементів, матричні, аналогові і цифро-аналогові.

Перші два типи індикаторів забезпечують представлення всіх букв, цифр і великої кількості символів. Матричні ВЛІ складаються з великої кількості світловипромінюючих елементів. Такий індикатор дозволяє відображати буквено-цифрові повідомлення, графіки, і навіть нескладні рухомі зображення.

Зазвичай в матричному індикаторі одна сітка покриває один стовпець світловипромінюючих елементів. Управління індикатором здійснюється по сіткових ланцюгах. При роботі яскравість світіння не постійна за площею, а знижується по краях, оскільки на них потрапляє менше електронів, ніж на центральну частину елементу. У цьому проявляється вплив сусідніх сіток, що мають негативний потенціал. З метою усунення цього недоліку розроблена вдосконалена конструкція матричного ВЛІ. У ньому кожна сітка покриває 2 стовпці випромінюючих елементів (рис. 3.11). Управління здійснюється як по сіткових, так і по анодних ланцюгах.

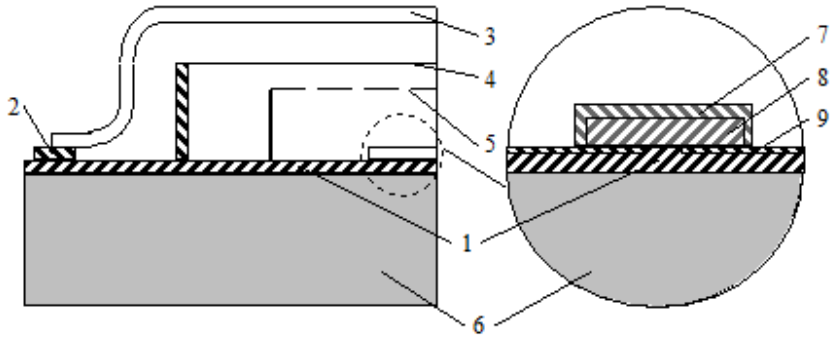


Рис. 3.11.

На рисунку позначено: 1 – провідний шар; 2 – герметик; 3– лицьове скло; 4 – катод; 5 – сітка; 6 – скляна пластина; 7 – шар люмінофору; 8 – провідний шар; 9 – шар діелектрика.

Така структура особливо успішно застосовується при високій зовнішній освітленості індикатора. Позитивна управляюча напруга подається на дві сполучені сітки і два розташованих під ними анода. В результаті яскравість світіння елементів виявляється рівномірною. Інші переваги цієї конструкції полягають в тому, що число управляючих сіток зменшене на половину і забезпечується велика яскравість за рахунок одночасного випромінювання світла двома стовпцями елементів.

Деякі ВЛІ мають дискретний анод, що складається з великого числа окремих елементів, розташованих уздовж прямої лінії, або по колу. За допомогою світлофільтрів можна отримати колір від синього до червоного. Яскравість цих кольорів є достатньою, якщо яскравість вихідного світіння складає приблизно 1000 Кд/дм^2. Використанням люмінофорів різних кольорів світіння (табл. 3.1) також можна отримати багатокольорові індикатори. Можна створити ВЛІ з різними кольорами світіння. Багатоколірний індикатор реалізують за рахунок конструктивних змін і специфічних способів управління.

Табл. 3.1.

Колір світіння	Слад люмінофора	Довжина хвилі, мкм
Синій	ZnS : Ag+In_2O_3	0,450
Синьо-зелений	ZnO : Zn	0,510
Зелений	(Zn, Cd)S : Ag	0,525
	ZnS : Cu	0,530
Лимонний	ZnS : Au, Al+In_2O_3	0,550
Жовтий	ZnS : Mn+ In_2O_3	0,585
Червоний	(Zn, Cd)S : Ag+ In_2O_3	0,626

Двохколірний індикатор можна отримати, якщо подвоїти число сегментів і покрити їх люмінофорами вибраних кольорів (рис. 3.4, а). При цьому збільшується число виводів, а символ при зміні кольору зміщується. Управління здійснюється по анодному ланцюгу.

Багатоколірний індикатор з сітковим управлінням дещо відрізняється розташуванням світловипромінюючих елементів (рис. 3.12, б). Його конструкція складніша, оскільки окрім загальної сітки в прилад ще вводяться сітки, відповідні світловипромінюючим елементам кожного з кольорів. Зміною потенціалів сіток можна міняти колір світіння. Аналогічними способами створюються триколірні індикатори, особливо якщо одночасно необхідно забезпечити високу роздільну здатність. Колір світіння ряду люмінофорів залежить від анодної напруги, що дозволяє створити багатоколірні ВЛІ простої конструкції з електричним перемиканням кольорів.



а *б*

Рис. 3.12.

Вибір режимів роботи ВЛІ визначається в основному вимогою до яскравості світіння індикатора і допустимими значеннями струмів, напруги, тривалості імпульсів, що подаються на анод і сітку.

В даний час випускається значна кількість типів ВЛІ: однорозрядні, багаторозрядні, сегментні в циліндричних і плоских балонах, матричні та

ін. На рис. 3.13 показана конструкція вакуумного люмінесцентного багаторозрядного індикатора (1 – катодна нитка; 2 – основа; 3 – анод; 4 – шина однойменних сегментів; 5 – сітка знакомісця; 6 – виводи з корпуса).

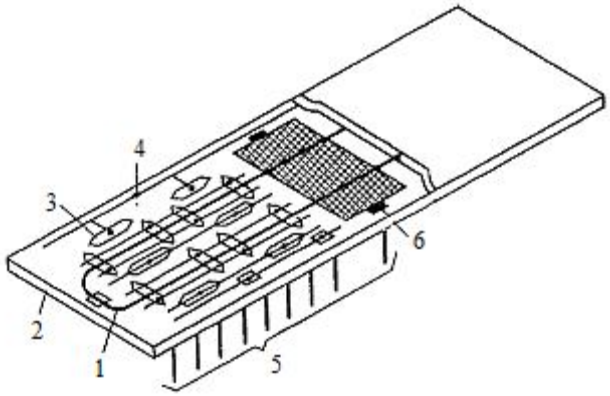


Рис. 3.13.

Більшість індикаторів випускаються разом зі схемою управління і живлення. Споживана потужність у 10-розрядного ВЛІ 1,33 Вт, в 40-розрядного 2,66 Вт, висота знаків – від 5 до 15 мм, середній час напрацювання на відмову 105 годин, ударна міцність 100*g*. На основі матричних ВЛІ також розроблені і випускаються модулі.

Розвитком цих пристроїв є ряд універсальних модулів з індикаторами різних розмірів, найбільші з яких мають 256×256 елементів, що забезпечує індикацію на площі 167,5×167,5 мм^2. Спосіб управління такими індикаторами хоча і передбачає збільшення числа виводів, управляючих напруг, він одночасно дозволяє підвищити яскравість до 700 Кд/м^2 і світловіддачу люмінофора до 5,2 Лм/Вт.

3.2.1. Статичне управління ВЛІ

При статичному способі управління збуджуючі сигнали подаються на необхідні для отримання заданого зображення аноди-сегменти, і все зображення знаку формується одночасно. Принципова схема підключення ВЛІ до джерела двійкових даних за допомогою мікросхеми серії К161 представлена на рис. 3.14.

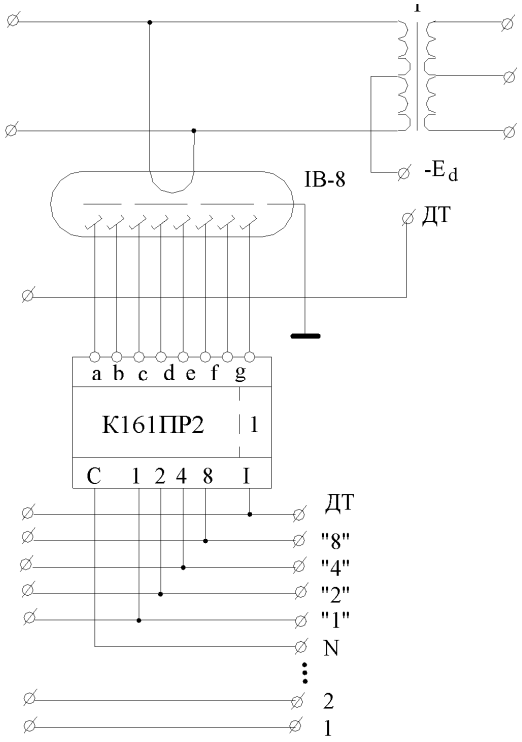


Рис. 3.14.

Двійкові числа, що надходять від джерела інформації, записують в регістр-фіксатор по сигналу дозволу запису. Після зняття сигналу дозволу запису дані зберігаються. При необхідності висвічування десяткової точки на вхід «ДТ» подають сигнал. Індикатор управляється напругою $U_a = U_c = 20$ В при анодних струмах 0,8 мА, що дозволяє застосувати мікросхему К161ПР2. Наявність в цій мікросхемі регістра-фіксатора дозволяє організувати статичний режим індикації багаторозрядного індикатора від загальної шини даних. Запис інформації в конкретне знакомісце здійснюється за допомогою стробування відповідного входу «дозвіл інформації».

При розробці схеми управління ВЛІ необхідно враховувати, що живлення ланцюга розжарення здійснюється змінним струмом, а для вирівнювання напруги між анодом і катодом негативний полюс джерела живлення приєднують до середньої точки вторинної обмотки трансформатора, як це показано на принциповій схемі.

3.2.2. Мультиплексне управління ВЛІ

Формування зображення мультиплексним (динамічним) способом полягає в тому, що протягом кожного моменту часу формується не повне зображення, а його окремі елементи.

Розрізняють три способи мультиплексного управління з часовою розгорткою:

– за сітками індикаторів;

– за анодами-сегментами індикаторів;

– за знаками.

При використанні управління за сітками (рис. 3.15), знаки по черзі синтезуються на кожному знакомісці.

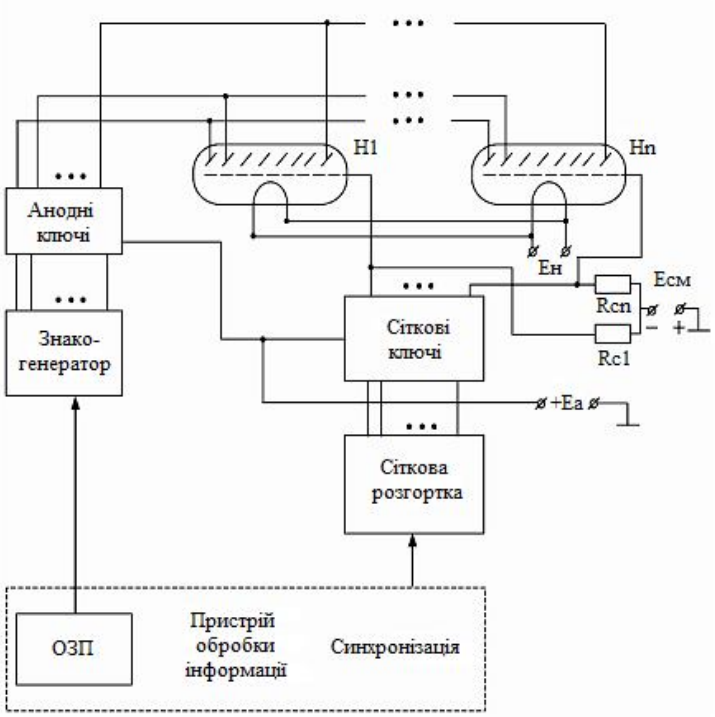


Рис. 3.15.

Аноди-сегменти збуджуються зі скважністю Q, що визначається кількістю знакомісць. Середня яскравість світіння анодів в Q разів менша миттєвої. При використанні управління за анодами-сегментами індикаторів (рис. 3.16) напруга збудження подається на однойменні аноди-сегменти, які беруть участь у формуванні знаків, що відображуються, а позитивна напруга на сітки окремих знакомісць подається в моменти анодної розгортки, які відповідають цифрі, що синтезується, в даному знакомісці. Середня яскравість світіння анодів нижче миттєвої в n разів (n – кількість сегментів в одному розряді).

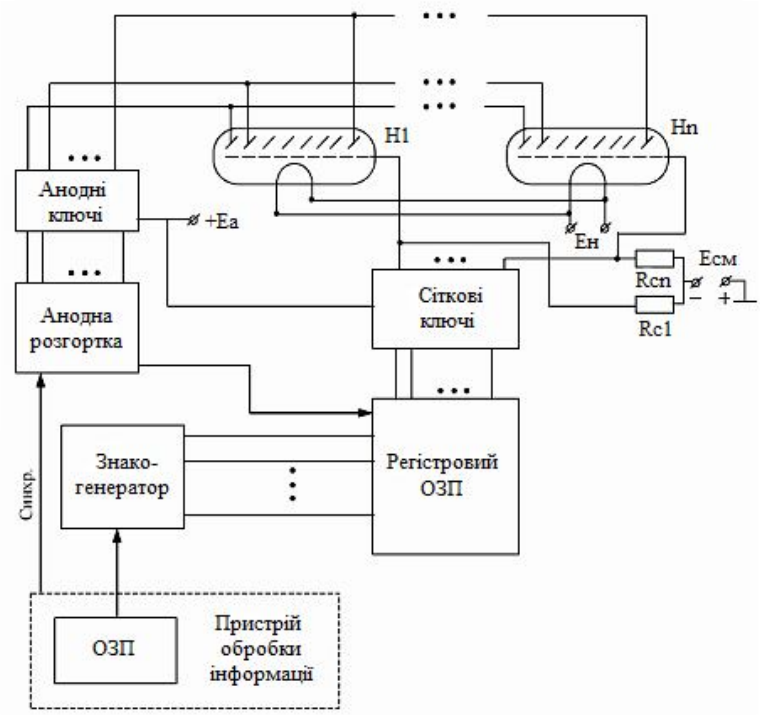


Рис. 3.16.

Зазвичай ВЛІ відображають обмежену кількість знаків, наприклад цифри від 0 до 9. У цих випадках можна використовувати розгортку за знаками (рис. 3.17), при якій на паралельно включені аноди-сегменти всіх знакомісць по черзі подається напруга, що відповідає кожному з 10 знаків, а позитивна напруга подається на сітку того знакомісця, на якому в даний час повинен відображатися відповідний символ.

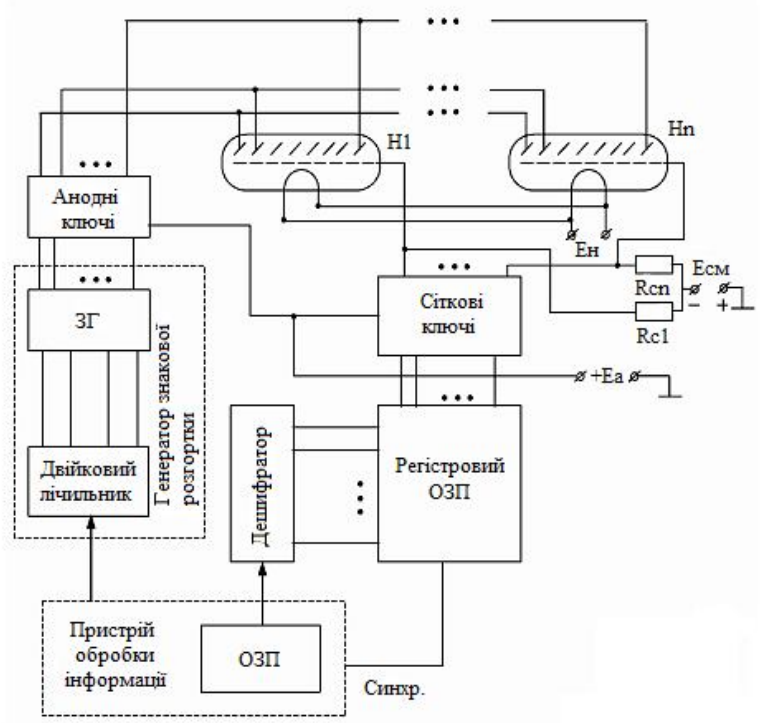


Рис. 3.17.

Для однорідного світіння анодів-сегментів на всіх знакомісцях індикатора при будь-якому способі мультиплексного управління необхідно забезпечити рівність скважностей на кожному з анодів, що беруть участь у формуванні знаку, який відображається.

3.3. Вакуумні індикатори розжарювання

Принцип дії вакуумних індикаторів розжарювання (ВІР) заснований на використанні явища теплового випромінювання твердих тіл, що нагріваються струмом до температури декілька тисяч градусів.

Під тепловим розуміють електромагнітне випромінювання, що виникає внаслідок перетворення теплової енергії на світлове випромінювання і таке, що не супроводжується жодними іншими змінами в тілі, окрім його теплового стану. Температура є основним параметром теплового випромінювання. Принциповим недоліком температурного випромінювання з точки зору його використання в техніці є низька економічність. Це пояснюється тим, що у видимій частині спектру при робочих температурах, які реально досягти, інтенсивність випромінювання нагрітих тіл мала. Тому лише дуже мала частина енергії, що випромінюється температурним джерелом, сприймається оком, тоді як інша частина енергії сприймається у вигляді тепла. ККД таких індикаторів складає одиниці відсотків.

Індикатор конструктивно є вакуумним балоном, всередині якого розташовані елементи відображення у вигляді вольфрамових спіралей (рис. 3.18).

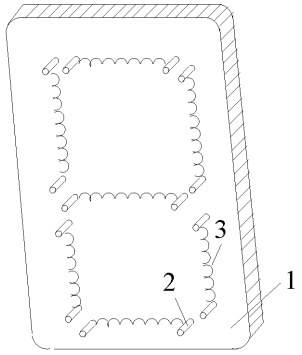


Рис. 3.18.

На рисунку позначено: 1 – основа; 2 – опорні виводи; 3 – спіраль розжарювання.

Робоча температура ниток не перевищує 1250 ^{0}C, чим забезпечується висока довговічність. Спіральна форма нитки і невелика відстань між опорними виводами запобігають помітному провисанню при нагріванні. Для збільшення контрасту основа має чорний колір.

Збудження індикаторів відбувається подачею постійної, змінної або імпульсної напруги між загальним виводом і окремими виводами сегментів. Нагрів елементу відображення відбувається не миттєво, отже

час реакції при живленні постійним струмом складає 0,2–0,25 с, проте, подаючи підвищену напругу на нитку (до 15В) можна здійснювати їх адресацію за 10 мс.

Вакуумні індикатори розжарювання зручні для пристроїв, які повинні працювати при високому рівні навколишньої освітленості. Вольт-амперна характеристика ВІР приведена на рис. 3.19.

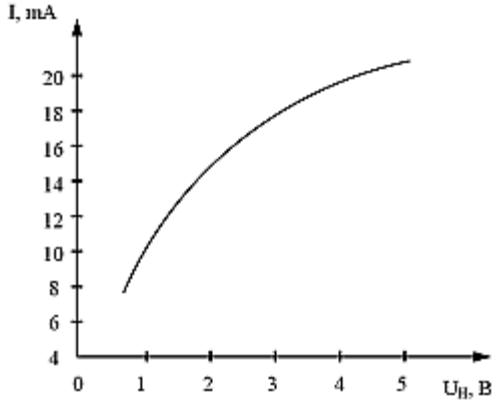


Рис. 3.19.

Аналіз вольт-амперної характеристики ВІР показує, що дані індикатори працюють при порівняно великих (у десятки міліампер) струмах. Необхідно також враховувати, що в початковий момент через малий опір холодної нитки можливий кидок струму (до двохкратного значення), що продовжується декілька десятих доль секунди. Номінальна робоча напруга ВІР зазвичай лежить в межах 3,5 В. При максимальному вихідному випромінюванні споживаний струм дорівнює 12....15 мА/сегмент. Споживана потужність типового 7-сегментного НПІ дуже близька до вказаного значення, а багато індикаторів високої яскравості споживають ще більшу потужність. Кут огляду ВІР, як правило, вище 90^0 (120^0). За нормальних робочих умов і номінальної напруги середній термін служби ВІР складає 10^5 годин. Зменшення напруги розжарювання забезпечує набагато більший термін служби. Наприклад, ВІР з робочою напругою 5 В, експлуатований при U=4 В, має приблизно 14-кратний виграш за терміном служби при роботі індикатора. При напрузі вище номінальної середній термін служби різко скорочується. За ударними і вібраційними навантаженнями ВІР більш вразливий, ніж індикатори інших типів.

Поряд із спіральною ниткою використовують плівкові нитки розжарення. При струмі 23 мА і напрузі 2,6 В, яскравість світіння цієї плівки досягає 1700 Кд/м^2.

Сегментні ВІР мають від 4 до 10 сегментів. Застосовуючи різні варіанти їх підключення, можна відображати цифри від 0 до 9, більшість букв російського і деякі букви латинського алфавіту, знаки і т.п. Для поліпшення умов зчитування рекомендується застосовувати нейтральні або кольорові світлофільтри.

У табл. 3.2 приведені параметри типових буквено-цифрових індикаторів розжарювання. При живленні імпульсною або змінною напругою, для запобігання виникнення резонансу рекомендується використовувати частоти в діапазоні 0-150 Гц, або вище 500 Гц.

Табл. 3.2.

Тип індикатора	Яскравість, Кд/м^2	Розмір знаку, мм	Кут огляду, град	Номінальна напруга, В	Струм розжарювання, мА
ІВ-9	1700/3000	6×12	120	4,5	19,5
ІВ-13	7000/10000	15,4×23	120	7,5	36

На рис. 3.20 приведена схема управління однорозрядним індикатором типу ІВ-9.

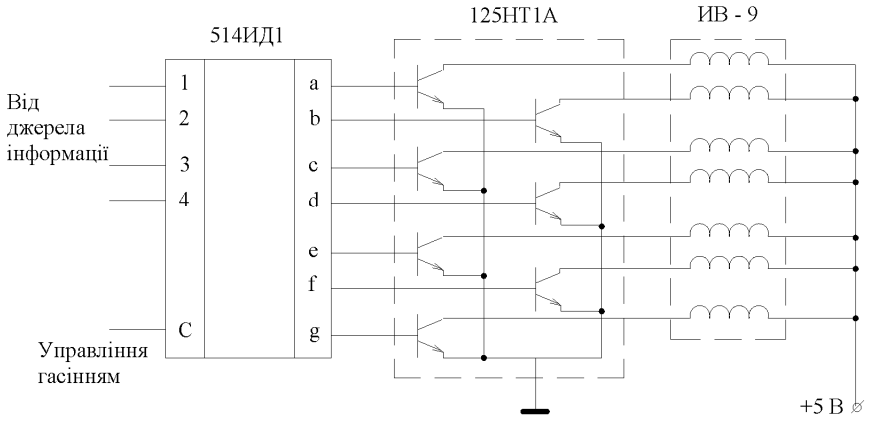


Рис. 3.20.

У цій схемі в якості знакогенератора використаний дешифратор К514ІД1. Вихідний струм даної мікросхеми достатній для включення транзисторної збірки типу 125НТ1А. Транзисторна збірка допускає вихідний струм до 40 мА. При живленні схеми джерелом 5 В і з врахуванням падіння напруги на транзисторній збірці, яка дорівнює приблизно 0,7 В, напруга на індикаторі складе 4,3 В.

3.4. Газорозрядні індикатори

Схема газоразрядної комірки і графік розподілу інтенсивностей представлені на рис. 3.21.

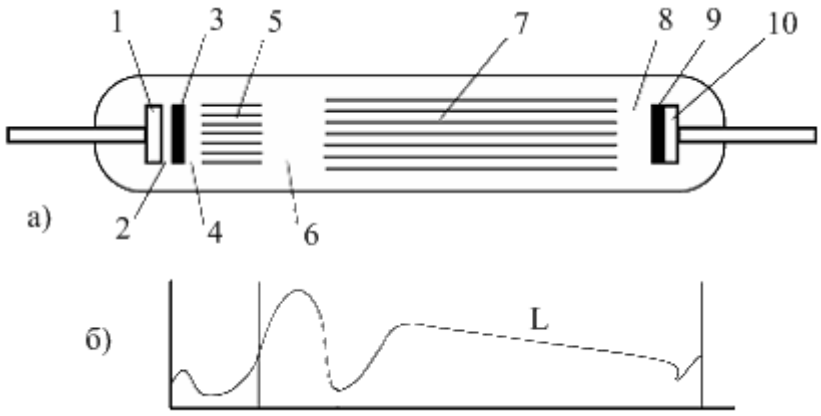


Рис. 3.21.

На рисунку позначені: 1 – катод; 2 – астоновий темний простір; 3 – катодний шар; 4 – катодний темний простір; 5 – негативне світіння; 6 – фарадеєвий темний простір; 7 – позитивний стовп; 8 – анодний темний простір; 9 – анодне світіння; 10 – анод. З рисунку видно, що випромінює світло не весь газорозрядний проміжок, а лише окремі його частини. Ці області, що світяться, чергуються з областями, в яких світіння або взагалі відсутнє, або дуже слабке.

Як і в твердому тілі, під час переходу електрона з одного енергетичного рівня на інший газ випромінює світло. Проте в газі електрони відділяються від атомів. При втраті одного електрона більшістю атомів газу говорять, що газ іонізований. У крайньому випадку, кожен атом може втратити один електрон, проте світіння спостерігається лише тоді, коли концентрація електронів та іонів перевищує 1% і газ переходить в плазму. У цьому стані випромінювання відбувається найінтенсивніше. Хоча термін «плазма» вперше застосований у зв'язку з газорозрядними індикаторами змінного струму, він однаково відноситься до приладів постійного струму. Два даних типи приладів розрізняються лише характером запалення і світіння. Для іонізації газу необхідно прикласти енергію, достатню для виведення електрона з поля іона. Ця енергія називається потенціалом іонізації.

Помаранчеве світіння характерне для більшості плазмових індикаторів, оскільки в них зазвичай використовують заповнення Ne (неон), хоча в окремих випадках додають домішки і деяких інших газів. При зіткненні електронів з неіонізованими атомами звільняються інші електрони, що і сприяє розвитку процесу іонізації. Електрони, що знаходяться в метастабільних станах, можуть теж отримати велику енергію завдяки довгому часу існування в цьому стані, і підвищують вірогідність іонізації. Вона може початися при напрузі, набагато меншій від напруги нормальної іонізації газу. В результаті цих процесів стан плазми підтримується доти, доки вистачає енергії. Типова вольт-амперна характеристика газового розряду приведена на рис. 3.22.

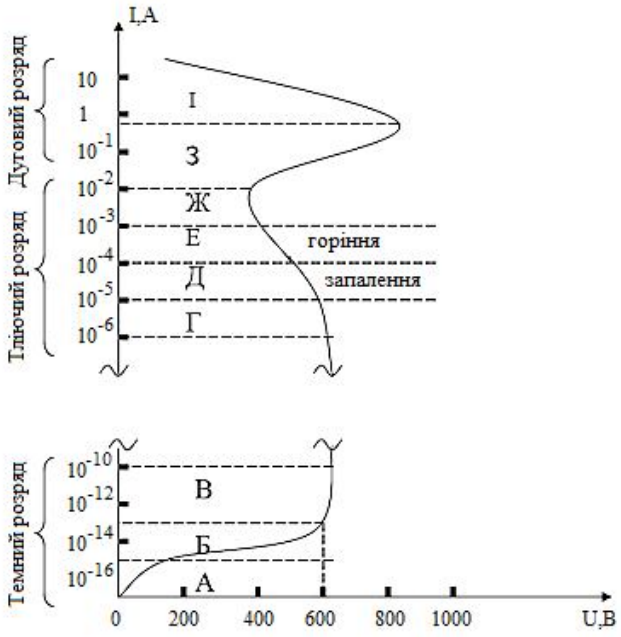


Рис. 3.22.

На рисунку використані наступні позначення: А – початкова ділянка (сплеск напруги); Б – струм насичення; В – розряд Таунсенду; Г – струм самостійного розряду; Д – корона; Е – переднормальна ділянка; Ж – ділянка нормального світіння (тліючий розряд); З – ділянка аномального світіння; І – дуговий розряд.

Ділянку Ж (ділянка тліючого розряду) зазвичай і використовують в різних індикаторах. Щоб тліючий розряд не переходив в дуговий, послідовно з газорозрядним елементом включають баластний резистор R_a.

Порівняно висока робоча напруга (сотні вольт) газорозрядних індикаторів зумовлює застосування цих приладів в апаратурі з мережним живленням. Основною випромінюючою областю розряду є негативне світіння. Світловіддача випромінювання в цій області невелика – близько 0,1 Лм/Вт. При прикладення до газорозрядної комірки напруги $U>U_З$ (напруга запалення) самостійний розряд починається з деякою затримкою. Час затримки складається із статичного часу запізнення і часу формування. Статичний час запізнення – це час між моментом прикладення напруги і початком протікання струму. Час формування дорівнює часу наростання струму від первинного значення до значення струму пробою. Якщо не прийняти спеціальні заходи, то час запізнення виявляється досить великим і має місце значна нестабільність цього часу при неперіодичному запаленні розряду. Повний час запізнення реального приладу складає одиниці мікросекунд.

Довговічність приладів з газовим розрядом обмежується, як правило, розпиленням матеріалу катода, що призводить до збільшення струму витоку і коротких замикань між електродами, зменшення виходу світла з індикатора, зменшення тиску газу.

Найбільш прості по конструкції і принципу дії знакові (цифрові) індикатори містять безліч катодів, оточених загальним анодом. Електродна структура знакомоделюючого цифрового індикатора, показана на рис. 3.23, містить набір з десяти катодів, кожен з яких має форму цифри, оточеної з усіх боків анодним електродом (1 – торцева частина анода (прозора); 2 – катод; 3 – бічна частина анода (суцільна)).

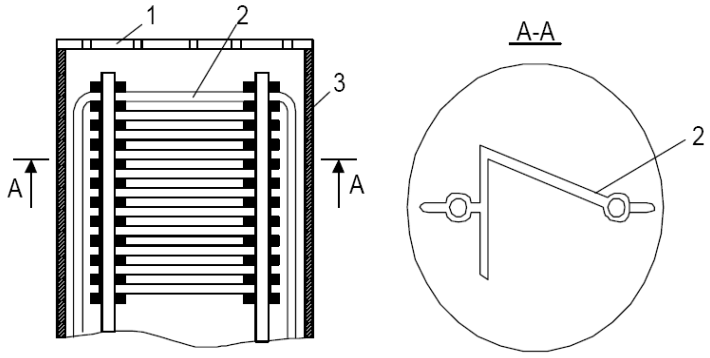


Рис. 3.23.

Для виведення випромінювання верхня частина анода виконана у вигляді оптично прозорої сітки. Оскільки тиск газу в приладі складає декілька тисяч паскаль, то світіння тліючого розряду утворює тонку (завтовшки в десяті долі міліметра) область, що щільно оточує катод. У зв'язку з цим область світіння має форму досить близьку до контура катода, тобто цифри, що відображається. Катоди виконуються у вигляді цифр від 0 до 9, знаків «+», «–», букв «Ом», «В», «А» та інших символів. Висота цифр 8–30 мм.

Переваги газорозрядних знакових індикаторів зводяться до наступного: великі розміри знаку, постійна готовність до роботи, мала споживана потужність, широкий сектор огляду, відсутність залежності яскравості знаку від кута огляду і низька вартість. До появи вакуумних люмінесцентних і напівпровідникових індикаторів вони були найбільш поширеними індикаторними приладами в обчислювальній і вимірювальній техніці. Недоліками цих індикаторів є необхідність комутування високої управляючої напруги і екранування одних символів іншими, а також зменшення кута огляду внаслідок великої товщини катодного пакету і обмежене число знаків, що використовуються.

3.5. Електрохромні індикатори

Під терміном «електрохромізм» розуміється широке коло явищ, при яких будь-яка зміна кольору речовини викликається прикладеним електричним полем. На рис. 3.24 представлене схематичне зображення рідинного (а) і твердотілих (б, в) EXI: 1 – скляні пластини; 2 – електроди; 3 – аморфний WO_3; 4 – прокладки; 5 – розчин електроліту; 6 – кристалічний WO_3: 7 – твердий електроліт; 8 – шар $MgWO_3$; 9 – шар діелектрика.

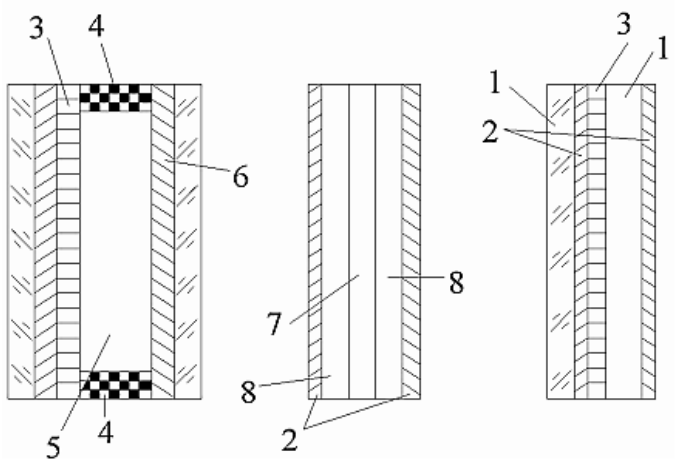


Рис. 3.24.

У першому випадку інтенсивність світла модулюється зміною коефіцієнта пропускання твердої неорганічної плівки окису вольфраму, товщина якої близько 1 мкм. Електрохромні індикатори з тонкими плівками конструктивно розділяються на три типи відповідно до поєднань компонентів, що використовуються: шар аморфного електроліту (рідинні); шар кристалічного WO_3 – твердий електроліт; шар аморфного WO_3 – тонка плівка діелектрика (твердотілі). Електролітом є суміш сірчаної кислоти, органічного спирту і пігменту. Спирт додають для розведення кислоти, а пігмент – для отримання білого фону. Товщина рідини декілька міліметрів.

Основні конструктивні елементи ЕХІ – електроди, електроліт, прокладки, передня і задня панелі. Колба повинна мати таку конструкцію, яка могла б протистояти значним температурним змінам (від –20 до +70 °С) без порушення герметичності. Матеріалом передньої панелі є скло, задньої – скло або нержавіюча сталь, прокладки – скло, тефлон.

У роботі ЕХІ можна виділити три етапи: запис, запам’ятовування і стирання. При записі за допомогою електрохімічного відновлення утворюється кольоровий шар; запам’ятовування є консервацією утвореного кольорового шару без витрати енергії; стирання пов'язане з усуненням кольору електрохімічним окисленням. Запис, як і стирання, може бути отриманий двома способами. Перший полягає в подачі напруги на робочі електроди так, що необхідна реакція окислення (відновлення) відбувається із значною швидкістю. В цьому випадку напруга має бути постійною, а струм змінюватися в часі. Другий спосіб полягає в підтримці постійного струму між робочим електродом і протиелектродом протягом запису (стирання). Тоді напруга між електродами залежатиме від часу.

У будь-якого типу ЕХІ на електроді, що контактує з шаром при подачі негативної напруги 1,0 – 1,5 В, утворюються центри забарвлення, кількість яких пропорційна протікаючому струму, і плівка WO_3 набуває блакитного кольору. Колір зберігається десятки годин після зняття напруги. Забарвлена плівка має певний потенціал, тобто індикатор такого типу володіє як оптичною, так і електричною пам'яттю. Знебварвлення

відбувається при зміні полярності прикладеної напруги. При знебварвленні і фарбуванні через шар проходить великий імпульс струму, в інший час струм не значний.

Електрохромні індикатори на основі іридієвих оксидних плівок мають малий час реакції (50 мс), великий термін служби (10^7 циклів) і здатність працювати при низьких температурах.

В даний час описано декілька конструкцій багатоколірних ЕХІ. Типовий багатоколірний рідинний індикатор працює таким чином. У вихідному стані розчин безбарвний, при прикладанні напруги 0,95 В на робочому електроді утворюється плівка блакитного кольору. Збільшення напруги до 1,5 В змінює колір плівки на пурпурно-червоний. Спектр поглинання суміші добре узгоджується зі спектрами поглинання використаних матеріалів. Оскільки час фарбування цих матеріалів різний і залежить від прикладеної напруги, колір плівки можна міняти лише зміною напруги. Розчин обезбарвлюється при подачі на ЕХІ напруги 1,5 В протилежної полярності.

Таким чином, ЕХІ мають ряд переваг: великий кут огляду і контраст, що не залежить від кута зору, мала споживана потужність і низька робоча напруга, широкий діапазон робочих температур. В цих індикаторах невеликий вибір можливих кольорів, значний часи реакції і релаксації, малий термін служби. Вирішення задачі збільшення терміну служби дозволить успішно застосовувати ЕХІ в різних областях для відображення статичної інформації або інформації, що повільно змінюється.

3.6. Електрофоретичні індикатори

Електрофорезом називається напрямлений рух колоїдних частинок або мікроіонів під дією зовнішнього електричного поля. Дія електрофоретичного індикатора (ЕФІ), що призводить до зміни коефіцієнта віддзеркалення і кольору робочої речовини індикатора, заснована на русі зважених заряджених пігментних частинок в забарвленій безводній рідині під дією електричного поля.

Комірка ЕФІ складається з тонкого (25–100 мкм) шару колоїдної суспензії, введеної у вигляді прошарку між двома скляними пластинами, на внутрішні поверхні яких нанесені прозорі електроди. На рис. 3.25 зображено ЕФІ, коли на два сегментні електроди подані відносно заземленого прозорого електроду напруги різної полярності (1 – прозорий електрод; 2 – сегмент; 3 – основа; 4 – заряджені часточки пігмента; 5 – забарвлена рідина; 6 – ущільнення; 7 – скляна пластина). При цьому, часточки суспензії осідають в одному випадку на сегменті, а в іншому – на прозорому електроді.

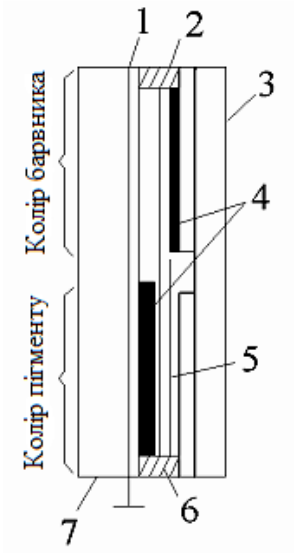


Рис. 3.25.

Оскільки рідина і пігмент мають різні кольори, то відбувається зміна кольору в площі сегментів. У першому випадку спостерігається колір рідини, а в другому – колір пігменту. Шари пігменту залишаються на електродах і після зняття напруги через дію поверхневих сил. Це обумовлює наявність внутрішньої пам'яті індикатора. При прикладенні напруги зворотної полярності, частинки переміщуються до протилежного електроду. Відповідним підбором пігменту і рідини можна отримати великий набір поєднань кольорів.

Завдяки розсіюючим властивостям, що містяться в робочій речовині ЕФІ, просторовий розподіл відбитого світла для обох типів індикаторів є майже рівномірним (ламбертовським).

Споживання потужності в ЕФІ обумовлене проходженням струму при переміщенні частинок пігменту і струмом спокою, що йде після припинення цього імпульсу. Друга складова може бути виключена зняттям прикладеної напруги після закінчення часу, необхідного для переміщення частинок, і забезпеченням ефекту пам'яті для підтримки бажаного зображення. В елементі завтовшки 50 мкм імпульс напруги амплітудою 30В переміщує частинки приблизно за 100 мс при щільності струму 1 мкА/см^2. Це призводить до споживання енергії 6 мкДж/см^2.

Головні причини виходу з ладу ЕФІ – групування, міграція, витоки і електрохімічні ефекти (наприклад, гідроліз, відбілювання фарбника). В даний час ресурс простих ЕФІ перевищує 10^8 циклів, складніших (цифрові, матричні) – $5 \cdot 10^7$ (10^8 циклів при частоті перемикання 1 Гц складають 4 місяці безперервної роботи). В даний час розрішаюча здатність складає 5 мм$^{-1}$ і може бути збільшена до 10–15 мм$^{-1}$.

Питання до розділу 3

1. Будова вакуумних люмінесцентних індикаторів.
2. Принцип роботи вакуумних люмінесцентних індикаторів.
3. Керування вакуумними люмінесцентними індикаторами.
4. Особливості ЕЛІ постійного струму.
5. Наведіть еквівалентну схему ЕЛІ.
6. Чим визначається вибір режимів роботи ВЛІ?
7. Поясніть різницю між статичним та мультиплексним керуванням ВЛІ.
8. Будова газорозрядних індикаторів.
9. Принцип роботи електрохромних індикаторів.
10. Будова електрофоретичних індикаторів.
11. Від чого залежить колір світіння тонкоплівкових люмінесцентних індикаторів?
12. Дайте порівняння різних типів індикаторів. Визначте переваги та недоліки кожного з них.

4. РІДКОКРИСТАЛІЧНІ ІНДИКАТОРИ

Терміном рідкий кристал визначається метафаза між твердим станом й ізотропним рідким станом, при цьому метафаза зберігає фундаментальні властивості, притаманні двом станам матерії. Рідкі кристали, з одного боку, мають властивості ізотропної рідини, з іншого боку, зберігають певний порядок в розташуванні молекул як кристал. Більшість рідких кристалів утворюються стрижневими молекулами.

В даний час рідкокристалічні індикатори є найбільш поширеним типом індикаторів. Сфери застосування РКІ-індікаторів і дисплеїв включають:

- вимірювальні прилади і обладнання;
- портативна побутова електроніка;
- прилади сбору інформації;
- навігаційне обладнання і системи визначення координат;
- моніторинг електронних міток у виробничо-складській логістиці;
- портативні телефони й пейджери;
- дисплеї в кабіні пілота авіасудна;
- персональні цифрові асистенти;
- електронні книги;
- електронні інформаційні табло;
- торгові автомати;
- медичне обладнання.

4.1. Принцип роботи і конструктивні модифікації

Рідкокристалічні індикатори (РКІ) управляють віддзеркаленням і пропусканням світла для створення зображень цифр, букв, символів й інших зображень. На відміну від світлодіодів, рідкокристалічні індикатори не випромінюють світло.

Основу РКІ складають рідкі кристали (РК), молекули яких впорядковані пошарово певним чином між двома скляними пластинами. У кожному шарі сигароподібні молекули РК вишиковуються в одному напрямі, їх вісі стають паралельними (рис. 4.1, а). Скляні пластини мають спеціальне покриття, таке, що спрямованість молекул в двох крайніх шарах перпендикулярна. Орієнтація кожного шару РК плавно змінюється від верхнього до нижнього шару, формуючи спіраль (рис. 4.1, б). Ця спіраль "скручує" поляризацію світла по мірі його проходження через дисплей.



а *б*

Рис. 4.1.

Під дією електричного поля молекули РК переорієнтовуються паралельно полю. Цей процес називається твіст-нематичним польовим ефектом. При такій орієнтації поляризація світла не скручується при проходженні через шар РК. Якщо передній поляризатор орієнтований перпендикулярно задньому, світло пройде через ввімкнений дисплей, але заблокується заднім поляризатором. В цьому випадку РКІ діє як заслінка світла.

Відображення різних символів досягається вибірковим травленням провідної поверхні, заздалегідь створеної на склі. Не витравлені області стають символами, а витравлені – фоном дисплея.

4.1.1. Рефлективні РКІ

Рефлективні РКІ використовують режим відображення з темними символами на світлому фоні (позитивне зображення). В індикаторі з позитивним зображенням передній і задній поляризатори знаходяться в протифазі, або перехресно поляризовані на 90°. Якщо сегмент "вимкнений", зовнішнє світло йде наступним шляхом: проходить через вертикальний поляризатор, через прозорий електрод сегменту, через РК молекули, які скручують його на 90°, далі через прозорий загальний електрод, горизонтальний поляризатор, і потрапляє на рефлектор, який відбиває світло назад тим же шляхом (рис. 4.2).

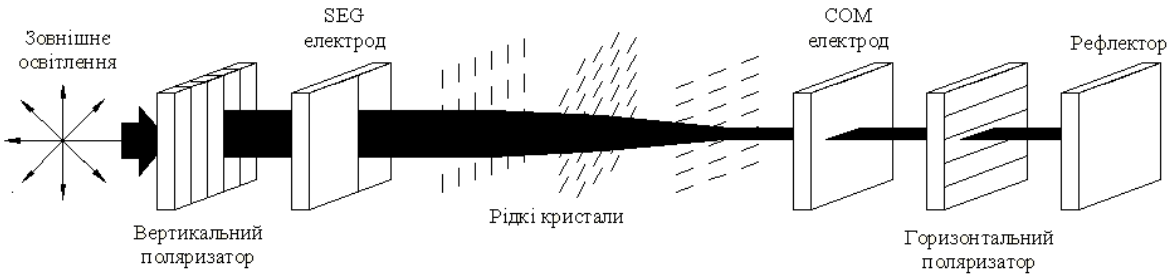


Рис. 4.2.

Якщо сегмент "включений", зовнішнє світло не змінює своєї поляризації при проходженніі через шар рідких кристалів. Таким чином, поляризація світла протилежна до заднього поляризатора, що не дає світлу

пройти до рефлектора. Оскільки світло не відбивається, отримується темний сегмент (рис. 4.3).

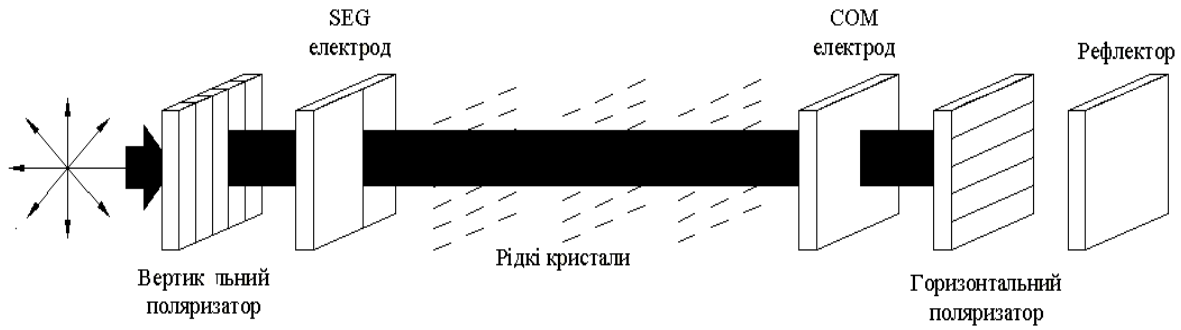


Рис. 4.3.

Рефлективні індикатори дуже яскраві, з відмінним контрастом, мають широкий кут огляду. Вони потребують хорошого зовнішнього освітлення і не використовують штучного заднього підсвічування.

4.1.2. Трансмісивні РКІ

Трансмісивні РКІ не відбивають світло. Навпаки, вони створюють зображення, управляючи світлом штучного джерела освітлення, розташованого позаду індикатора. У трансмісивних індикаторах передній і задній поляризатори знаходяться "у фазі" один з одним (паралельні). У вимкненому сегменті поляризоване світло скручується на 90° молекулами РК і опиняється в протифазі з переднім поляризатором. Поляризатор блокує світло, створюючи темний сегмент (рис. 4.4).

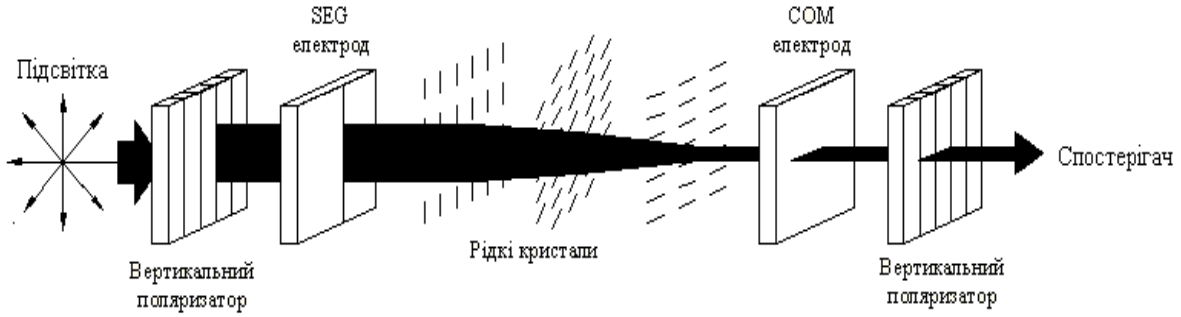


Рис. 4.4.

Якщо сегмент включений (рис. 4.5), світло не скручується, опиняючись у фазі з переднім поляризатором, і проходить через нього, створюючи світловий малюнок. Таким чином, трансмісивний дисплей створює світле зображення на темному фоні (негативне зображення).

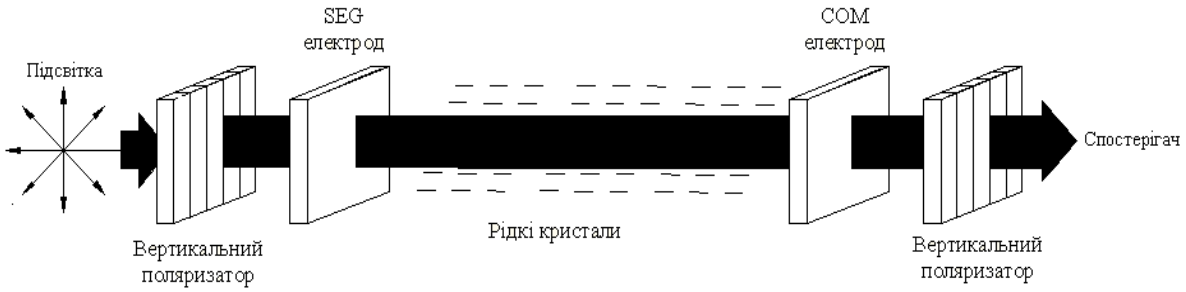


Рис. 4.5.

Трансмісивні індикатори повинні мати заднє підсвічування, щоб гарантувати рівномірне світіння сегментів. Вони хороші для використання в умовах слабкого освітлення. В умовах прямого сонячного світла підсвічування не може здолати сонячних променів, і зображення не помітне.

4.1.3. Трансрефлективні РКІ

Трансрефлективні індикатори використовують білий або срібний напівпрозорий матеріал, який відбиває частину зовнішнього світла, а також пропускає світло заднього підсвічування. Оскільки ці індикатори як відбивають, так і пропускають світло, вони можуть використовуватися в широкому діапазоні яскравостей освітлення. Прикладом можуть служити індикатори мобільних телефонів – вони читаються як при яскравому освітленні, так і в повній темряві. Трансфлективні дисплеї мають нижчу контрастність в порівнянні з рефлективними, оскільки частина світла проходіт крізь відбивач.

4.1.4. Бістабільні РКІ

Бістабільні технології засновані на використанні природної (фізичної) пам'яті РК-матеріалу в дисплеї без застосування активних додаткових елементів. У бістабільних РК-дисплеях зображення, будучи відтвореним один раз (записаним), може зберігатися, не вимагаючи регенерації або подачі будь-яких сигналів для підтримки ефекту пам'яті. Завдяки цій властивості можна значно зменшити потужність споживання РК-дисплея. Основні особливості бістабільних РКІ полягають в наступному:

- зображення зберігається при вимкненому живленні;
- діапазон температур: –40...+100 ^{0}C;
- контрастність до 20:1 – пікселі подібні мікродзеркалам і відсутній поляризатор;
- не потрібне підсвічування;
- кути огляду: максимальні (180×180 град.) – рефлективна поверхня при відсутності поляризатора;

- відмінна читабельність навіть при прямому сонячному світлі і на будь-яких кутах, завдяки віддзеркаленню до 70% падаючого на них світла;
- нульове споживання струму в статичному стані завдяки бістабільності пікселів, і значно нижче, ніж в звичайних РК-дісплеїв при оновленні інформації;
- низька вартість.

Такі екрани відносяться до класу статичних дисплеїв, де енергія витрачається лише на перше виведення зображення, аж до наступної зміни зображення. Звичайно, існує термін, при якому зображення плавно зникне з екрану, але він дуже тривалий.

Технологія заснована на бістабільних РК-структурах – мікроструктурах, здатних протягом тривалого часу знаходитися в одному з двох стійких станів навіть за відсутності зовнішнього електричного поля. В одному з цих станів комірка на основі рідкого кристала пропускає світло, а в іншому – ні. Для перемикання комірки з одного стану в інший застосовується зовнішнє електричне поле, що виникає при подачі напруги на пару управляючих електродів. Об'єднавши безліч бістабільних комірок в двовимірний масив, можна створити дисплей з дуже високою дозволяючою здатністю.

РК-дисплей на основі бістабільності BiNem (рис. 4.6) має дві скляні підкладки з ортогональною системою електродів. Технологія BiNem використовує стандартний нематичний матеріал. У конструкції може використовуватися як два поляризатори, так і один. Важливим параметром є товщина робочого проміжку – близько 2 мкм. Це перша характерна особливість даної технології.

Необхідність отримання рівномірних малих проміжків по всій площі екрану є характерною межею для всіх бістабільних дисплеїв. Такий проміжок з високою рівномірністю за площею реалізувати складно. Основний дефект, який часто виникає при реалізації малих проміжків, – коротке замикання між електродами верхньої і нижньої підкладок дисплея.

Наявність малого проміжку необхідна, щоб реалізувати певні фізичні властивості молекул РК-матеріалу. Чим менше проміжок, тим меншим стає крок спіралі РК-молекул, а розміри проміжку стають співрозмірні з розмірами спіралі РК-молекул. При таких малих проміжках поведінка молекул нематичного РК-матеріалу дещо відрізняється від поведінки молекул в стандартній твіст-нематичній комірці (проміжок 20–25 мкм). Друга характерна особливість даної технології – використання орієнтуючих покриттів, що забезпечують слабке зчеплення спіралі молекул РК-матеріалу.

Орієнтуючі шари призначені для того, щоб створити різні умови для зв'язку безлічі мікропружинок РК-матеріалу з верхньою і нижньою підкладками. Одним із способів задання вектора орієнтації є створення

мікрорельєфу, утвореного канавками з різною щільністю або глибиною. Мікрорельєф може бути заданий селективним травлінням матеріалу. Площини граничних витків пружин мають малий кут по відношенню до поверхні підкладки і укладаються в канавки шару орієнтуючого покриття. Один з кінців пружини має по відношенню до другого кінця сильніше зчеплення з підкладкою. Орієнтація канавок на обох підкладках різна – напрям канавок на нижній підкладці зміщена на 90° відносно верхньої.

При дії електричного поля молекули, утворюючі пружину, розвертаються вздовж поля, пружина руйнується і вироджується в диполь. Після зняття поля молекули під дією внутрішніх сил, що визначаються просторовою топологією, знову прагнуть повернутися до стану пружини. Цей динамічний процес проходить на фоні іншого процесу – взаємодії кінців формованих пружин з силами, що виникають на границі з орієнтуючим покриттям. Відмінність у величині цих сил для верхньої і нижньої підкладки дає можливість реалізувати ефект бістабільності. За певних умов кінці пружин, утворених РК-молекулами, орієнтовані на обох підкладках точно у відповідності до векторів, заданих орієнтуючими покриттями. Після зняття поля система може зайняти два стійкі стани, які володіють різними оптичними властивостями. Ця несиметричність забезпечує особливе перетікання перехідних процесів в процесі запису.

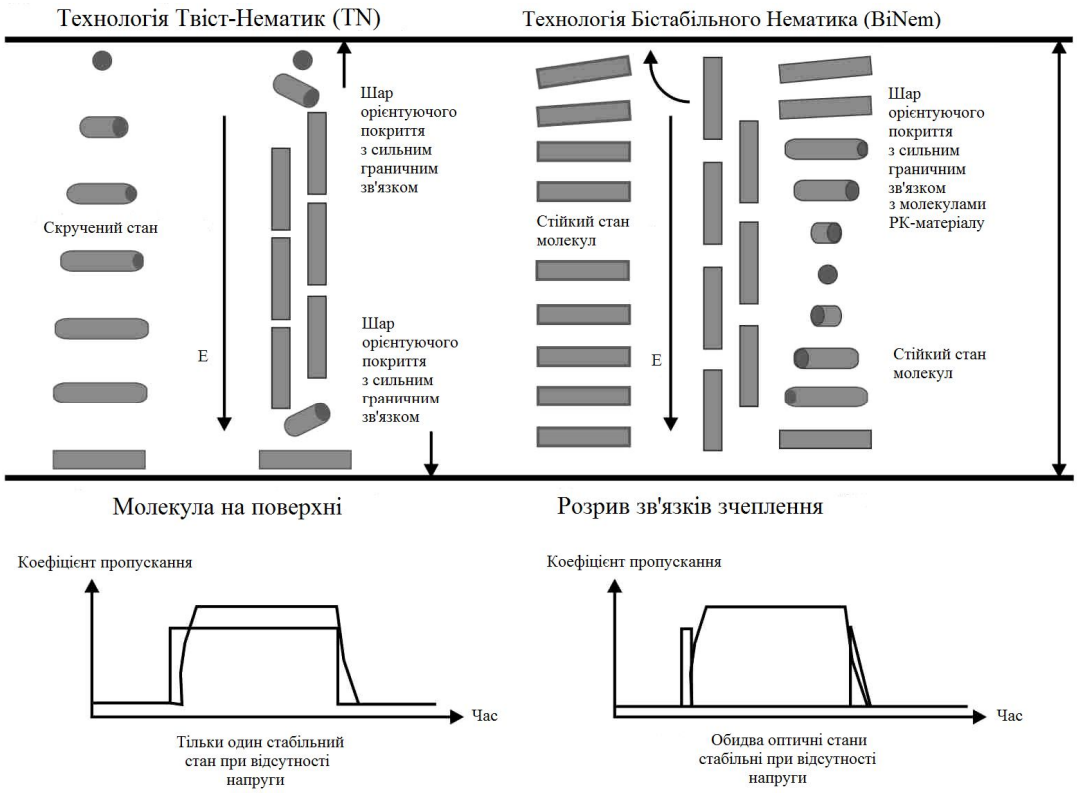


Рис. 4.6.

Комірка BiNem має два стійкі оптичні стани – перший стан відповідає нескрученій орієнтації молекул РК-матеріалу (U – untwisted) у комірці зображення, а друге – скрученою на 180° (T – twisted).

РК-комірка може мати і проміжний гомеотропний стан, що досягається при подачі електричного імпульсу амплітудою вище за деяку порогову E_c. У гомеотропному стані диполі молекул РК-матеріалу орієнтуються уздовж електричного поля між електродами (рис. 4.7).

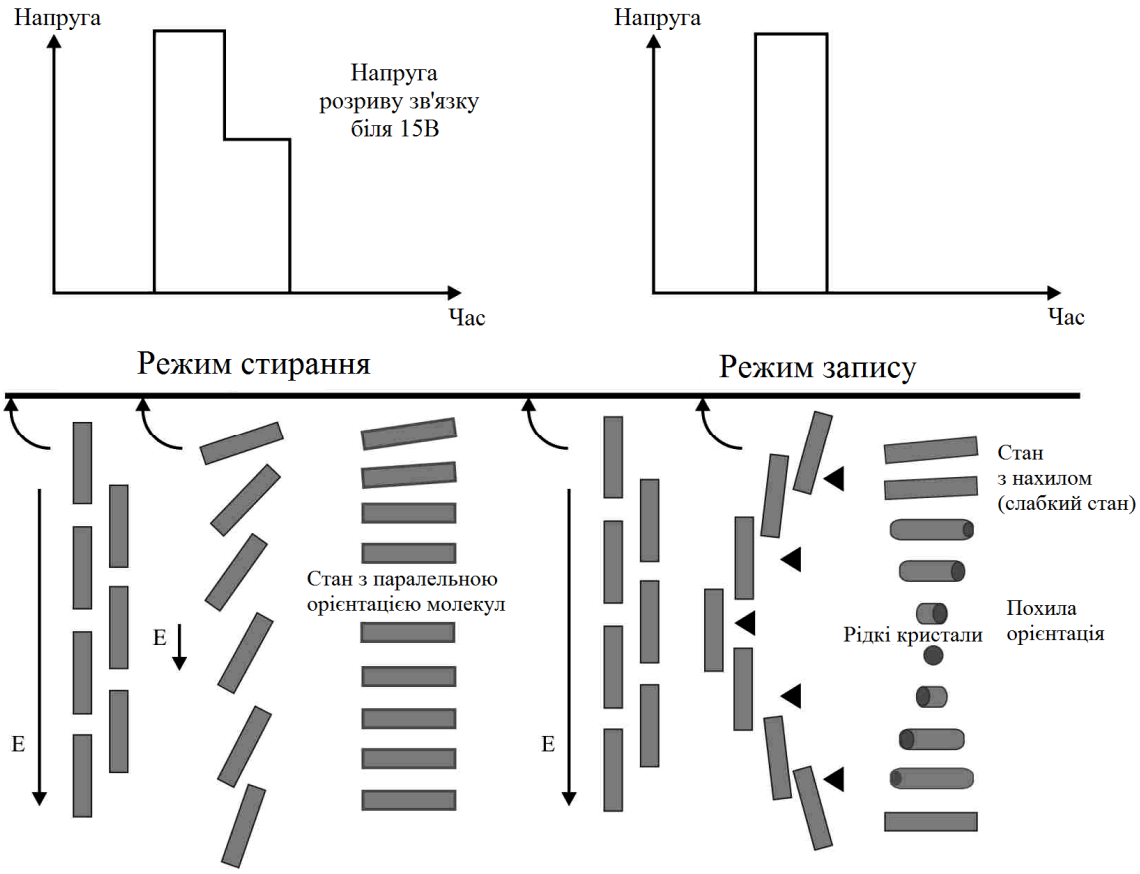


Рис. 4.7.

При знятті електричного поля молекули взаємодіють з поверхнево активними силами, заданими спеціальними орієнтуючими шарами на верхній і нижній підкладці дисплея. Для даної технології важливе те, що сила взаємодії молекул на границі з орієнтуючими покриттями верхньої і нижньої підкладки різна.

Практично, будучи одного разу записаною, структура зображення може залишатися стабільною при відсутності електричного поля протягом декількох місяців. Лише дуже сильні удари, величина яких знаходиться вже на межі механічного руйнування матеріалу підкладки, можуть призвести до втрати або порушення записаного зображення. Проте, це не призведе до необоротних наслідків, і знову записане зображення матиме таку саму якість, як і до удару.

4.1.5. Позитивні і негативні РКІ

Залежно від орієнтації поляризатора, РКІ може відображати позитивне або негативне зображення. У дисплеї з позитивним зображенням передній і задній поляризатор перпендикулярні один одному, так що виключені сегменти і фон пропускають світло зі зміненою поляризацією, а включені сгменти перешкоджають проходженню світла. В результаті – темні символи на світлому фоні.

У дисплеї з негативним зображенням поляризатори паралельні, і перешкоджають проходженню світла з поверненою поляризацією, так що виключені символи і фон темні, а включені – світлі.

Рефлективний РКІ має відбивач (рефлектор) за заднім поляризатором, який відбиває світло, яке пройшло через викючені сегменти і фон. У негативних рефлективних дисплеях світло відбивається через включені сегменти. Трансмісивні дисплеї використовують ті ж принципи, але фон, або сегменти, стають яскравішими за рахунок використання заднього підсвічування.

Щоб вибрати оптимальний режим для конкретного випадку необхідно розглянути типові умови освітлення індикатора (табл. 4.1).

Табл. 4.1.

Режим відображення	Зображення	Застосування	Пряме сонячне світло	Офісне освітлення	Слабке світло	Дуже слабке світло
Рефлективний позитивний	Темні сегменти на світлому фоні	Без підсвічування. Забезпечує кращий фронтальний контраст і стабільність.	Відмінно	Дуже добре	Погано	Дуже погано
Трансфлективний позитивний	Темні сегменти на сірому фоні	Може освітлюватись відбитим зовнішнім світлоом або підсвічуванням.	Відмінно (без підсвічування)	Добре (без підсвічування)	Добре (підсвічування)	Дуже добре (підсвічування)
Трансфлективний негативний	Світло-сірі сегменти на темному фоні	Потрібне яскраве освітлення або підсвічування Часто використовується з кольоровим трансфлектором (напівпрозорий відбивач).	Добре (без підсвічування)	Добре (без підсвічування)	Добре (підсвічування)	Дуже добре (підсвічування)
Трансмісивний позитивний	Темні сегменти на підсвічуваному фоні	Розроблено для поганих умов освітлення, можливе використання при зовнішньому освітленні.	Добре (без підсвічування)	Добре підсвічування)	Дуже добре (підсвічування)	Відмінно (підсвічування)
Трансмісивний негативний	Подсвічені сегменти на темному фоні	Не може бути використаний без підсвічування.	Погано (підсвічування)	Добре підсвічування)	Дуже добре (підсвічування)	Відмінно (підсвічування)

4.2. Параметри рідкокристалічних індикаторів

При виборі РКІ слід визначити, як спостерігач дивитиметься на індикатор: чи буде він сидіти або стояти, під яким кутом розташований дисплей, яка потрібна ширина кута огляду. Річ у тому, що контрастність зображення на індикаторі залежить від відносного розташування дисплея і спостерігача.

Кут огляду. Звичайний напрям зору описується аналогічно циферблату годинника. Якщо спостерігач дивиться зверху, це називається 12 годин, знизу – 6 годин, справа – 3 години, зліва – 9 годин. Критичні кути зору (нахилу індикатора) залежать від напряму огляду і можуть бути проілюстровані ізоконтрастними кривими на графіку в полярній системі координат (рис. 4.8).

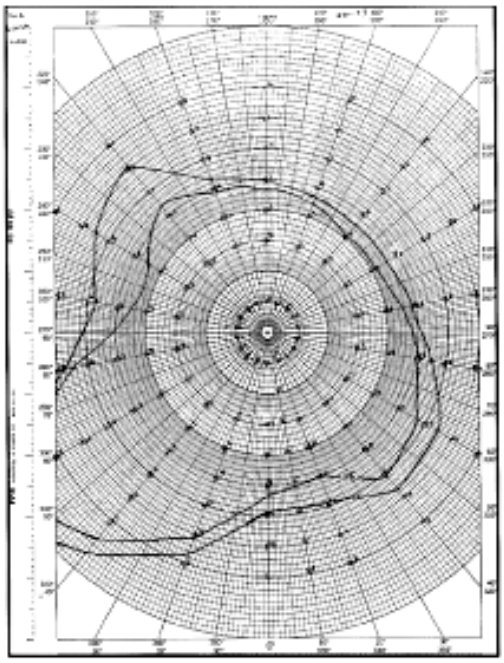


Рис. 4.8.

Кут огляду залежить також від товщини шару РК. Більшість РКІ виготовляються за другим класом з товщиною від 6 до 8 мікрон. Перший клас має товщину від 3 до 4 мікрон. Найбільш широкий кут огляду (до 165°) досягається при 4-х мікронній технології. При цьому також зменшується час спрацьовування РКІ.

Час спрацьовування. РКІ зазвичай має час спрацьовування 20 мс при 20°С, а кращі моделі – до 40 мкс. Стандартний РКІ може відображати сигнал частотою до 10 Гц, якщо потрібно; неозброєним оком важко відстежити дані з такою частотою.

Температура використання. Аналіз температурного діапазону дуже важливий при описі РКІ. Всі РК матеріали мають строго визначену верхню межу робочої температури, або ізотропічну межу. Вище за цю межу

молекули РК приймають довільну орієнтацію. Ізотропічні умови роблять позитивне зображення повністю темним, а негативне – прозорим. Ізотропічна температура називається температурою нематично-ізотропічного переходу.

РКІ можуть відновлюватися після короткої дії ізотропічної температури, хоча температури більше 110°С руйнують внутрішнє покриття індикатора.

Нижня межа температурного діапазону РКІ не так добре визначена, як верхня. При низьких температурах час спрацьовування індикатора збільшується, оскільки сповільнюється рух молекул і зростає в'язкість РК речовини.

При дуже низьких температурах РК речовина переходить в твердий, або кристалічний стан. Ця температура називається температурою кристалічно-нематичного переходу. Проте, РК матеріал сприймає температури нижче за цю межу, фактично повертаючи кристали речовини. Ефект низьких температур зазвичай обратимий. РКІ, опущений в рідкий азот, повертається в нормальний стан після короткого періоду нагрівання.

Нагрівачі. За межами температурного діапазону може знадобитись температурна компенсація. Індикатори з інтегральними нагрівачами можуть працювати при температурах до –55°С. Нагрівачі вимагають температурно-керованого джерела живлення. При користуванні нагрівачами час відгуку індикатора при низьких температурах залишається таким же, як і при 0°С. Збільшення потужності нагрівача зменшує час нагріву. Зазвичай потрібна потужність між 2 і 3 ватами на квадратний дюйм поверхні індикатора.

Варіанти підсвічування. РКІ-модулі можуть оснащуються заднім підсвічуванням, що розміщується між РКІ і монтажною платою. У табл. 4.2 представлені варіанти підсвічування РКІ.

Табл. 4.2.

Властивість	Світлодіодний	Лампами розжарювання	Електролюмінесцентний
Яскравість	Середня	Висока	Мала – Середня
Колір	Червоний – Янтарний – Зелений	Білий	Білий
Розмір	Малий	Малий – Средній	Тонкий
Кріплення	SMD – Радіальний	Радіальний – Осьовий	Осьовий
Напруга	5 Вольт	1,5 В – 28 В	45 В – 100 В
Струм при 5 В (на кв.дюйм)	10 – 30 мА	20 мА	1 – 10 мА
Температура	Теплий	Гарячий	Холодний
Вартість (на кв. дюйм)	0,10 – 1,00 долл.	0,10 – 0,80 долл.	0,50 – 2,00 долл.
Розповсюдження світла	Напрямлене	Сферичне	Ламбертовське
Ударостійкість	Відмінна	Низька	Відмінна
Строк служби (годин)	100000	150 – 10000	500 – 15000

Підсвічування може бути реалізоване декількома способами: за допомогою електролюмінесцентної панелі, що є тонкою плівкою, і випромінює світло при прикладенні змінного струму підвищеної напруги порядка 100...150 В; люмінесцентною лампою з холодним катодом, що також працює при підвищеній напрузі, випромінювання якої рівномірно розподіляється по всій площі РКІ за допомогою відбивача або плоского світлопровідника. Третій варіант – підсвічування на основі світлодіодної матриці.

Перші два способи підсвічування забезпечують високу яскравість і можуть мати білий тон світіння при відносно низькому споживанні, але вимагають наявності джерела підвищеної напруги, що викликає деякі труднощі при створенні апаратури з автономним живленням. Навпаки, світлодіодне підсвічування не вимагає високовольтного джерела, і при використанні нескладного джерела струму дозволить виробляти живлення від джерела з напругою 5 В. Крім того, світлодіодне підсвічування має значно більший (у десятки разів) час напрацювання, а також лише воно допустиме до експлуатації в розширеному діапазоні температур.

4.3. Багатосегменті й матричні РКІ

Частини РКІ, що працюють як заслінки, вмикаючись і вимикаючись для формування зображень, називаються сегментами. Символи створюються з одного або декількох сегментів. Кожен сегмент може бути адресований індивідуально. Таким чином, проходження світла управляється електрично, вмикаючи і вимикаючи необхідні сегменти. У неактивній частині дисплея спрямованість молекул залишається спіральною, формуючи фон. Включені сегменти складають символи, що контрастують з фоном.

Сегменти створюються прозорими електродами з оксидів індія і олова, нанесеними на скло РКІ. Цифри від 0 до 9 і деякі букви можуть відображатися на семисегментному індикаторі. Шістнадцятисегментний індикатор може відображати цифри, всі латинські і майже всі російські букви (окрім Й, Ц, Щ). Для того, щоб символи були більш натуральнішими, використовують матричні індикатори. З їх допомогою можна також відображати невеликі зображення. Кількість сегментів індикатора впливає на метод управління ним. На рис. 4.9 представлені семисегментний, шістнадцятисегментний і матричний дисплеї.

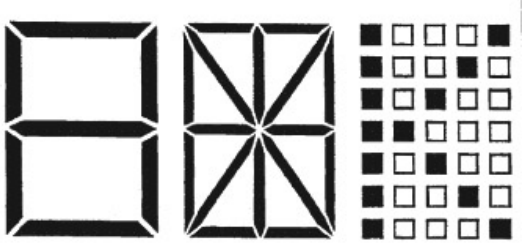


Рис. 4.9

РКІ може відображати невеликі картинки, або іконки. На рис. 4.10 представлений вигляд сегментного рідкокристалічного індикатора.



Рис. 4.10.

Матричні рідкокристалічні дисплеї є одними з основних засобів виведення інформації для сучасних цифрових систем. В порівнянні зі своїми попередниками – знаковими рідкокристалічними індикаторами – матричні РКІ мають набагато більші можливості виводу. При цьому вони зберігають головну перевагу перших – низьке енергоспоживання, завдяки чому можуть використовуватися в портативній апаратурі.

З точки зору управляючого інтерфейсу РКІ діляться на два великі класи – з контролерами і без контролерів. Перші зазвичай застосовуються в системах, що володіють не дуже великими обчислювальними ресурсами. Наявність контроллера повністю звільняє програміста від операції регенерації зображення – йому необхідно лише переслати інформацію в ОЗП дисплея. Для цього використовується простий 8-розрядний паралельний інтерфейс.

Дисплеї без контролерів простіші по структурі, але для управління вимагають великих обчислювальних ресурсів.

Приклад 4.1. Розробити схему годинника з виведенням часу на рідкокристалічний індикатор. Написати програму для мікроконтролера. Передбачити можливість установки часу.

На рис. 4.11 представлена принципова схема годинника. Рідкокристалічний модуль має виводи для управління кожним сегментом знакомісць. Низький рівень сигналу на відповідному вході запалює сегмент, а високий – гасить. Оскільки рідкокристалічний модуль містить велику кількість виводів, більше ніж мікроконтролер, то для управління індикатором використовуються паралельні регістри. Для запису даних в регістр необхідно сформувати високий рівень сигналу на виходах PA0-PA3 відповідно. Кожен регістр управляє окремим знакомісцем індикатора. У схемі використовуються дві кнопки для установки часу – годин і хвилин. Вони приєднані до входів зовнішніх переривань мікроконтролера. Переривання виникають по спаду сигналу. Для уникнення перешкод ці виводи підтягнуті до лінії живлення через резистори номіналом 1 кОм.

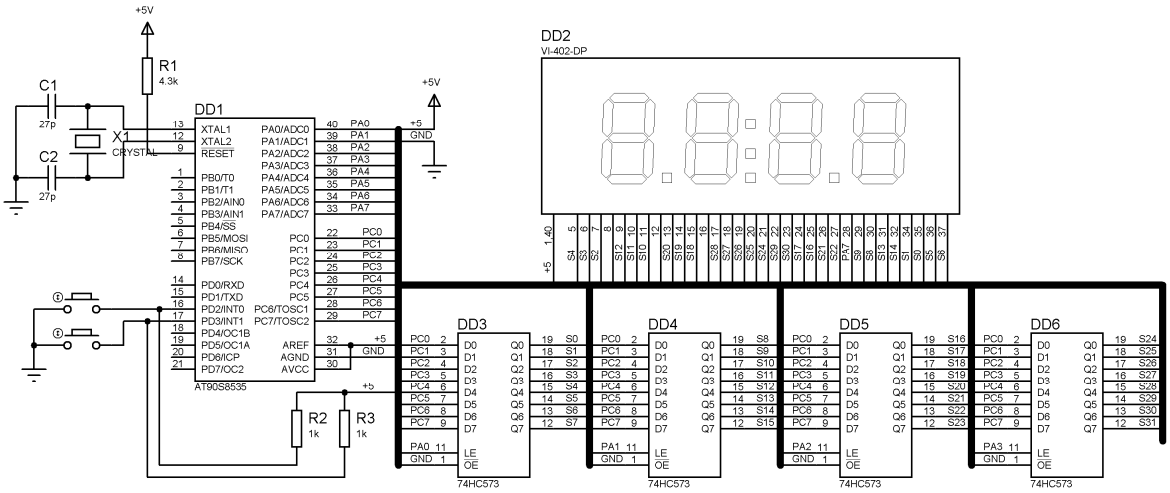


Рис. 4.11.

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>          // підключення заголовочних файлів
#include <macros.h>

void init_cpu(void);          // оголошення функції управління портами
void delay(int ms);           // оголошення функції затримки
void showtime(void);          // оголошення функції виведення часу на екран РКІ

 char sec=0, min=0, hour=0; // оголошення змінних
 // в масиві зберігаються коди цифр для 7-сегментного індикатора
 char seg[10]={0b00111111, 0b00000110, 0b01011011, 0b01001111,
0b01100110,0b01101101, 0b01111101, 0b00000111, 0b01111111, 0b01101111};

 #pragma interrupt_handler int0_isr:2        // обробник зовнішнього переривання 0
void int0_isr(void)
{
 if(hour<23) hour++;          // якщо значення змінної менше 23, відбувається інкремент
 else hour=0;                 // інакше відбувається обнуління змінної
 showtime();                  // виклик функції виведення часу на екран РКІ
}

#pragma interrupt_handler int1_isr:3 // обробник зовнішнього переривання 1
void int1_isr(void)
{
 if(min<59) min++;            // якщо значення змінної меньше 59, відбувається інкремент
 else min=0;                  // інакше відбувається обнуління змінної
 showtime();                  // виклик функції виведення часу на екран РКІ
}

void init_cpu()               // функція ініціалізації мікроконтролера
{
 DDRA=0b10001111;             // налаштування виводів PA7, PA3-PA0 на вихід
 PORTA=0;                     // виведення значення 0 в порт A
```

```
 DDRC=0xFF;                  // налаштування порту С на вихід
 PORTC=0;                    // виведення значення 0 в порт С
 MCUCR = 0x0A;               // преривання виникають при появі зрізу сигналу
 GIMSK = 0xC0;               // дозвіл обробки зовнішніх переривань
 SEI();                      // дозвіл обробки переривань
}

void delay(int ms)           // реалізація функції затримки
{
 int c, c1;                  // оголошення додаткових змінних
 for(c=0;c<ms;c++)
 {
  for(c1=0;c1<970;c1++) NOP();     // в циклі виконується «пуста» операція
 }
}
void showtime()              // функція виведення часу на екран РКІ
{
  PORTA=0b00000001;          // виведення значення 1 в порт А
  PORTC=~seg[hour/10];       // виведення значення годин (десяті)
  PORTA=0b00000010;
  PORTC=~seg[hour%10];       // виведення значення годин (одиниці)
  PORTA=0b00000100;
  PORTC=~seg[min/10];        // виведення значення хвилин
  PORTA=0b00001000;
  PORTC=~seg[min%10];        // виведення значення годин (одиниці)
}
void main()                  // головна функція програми
{
 init_cpu();                 // виклик функції ініціалізації мікроконтролера
 while(1)                    // вічний цикл
 {
  showtime();                // визов функції виведення часу на екран РКІ
  if(sec<59) sec++;          // якщо значення секунд менше 59, інкремент змінної
  else                       // в протилежному випадку
  {
   sec=0;                    // обнуління змінної
   if(min<59) min++;         // якщо значення хвилин менше 59, інкремент змінної
   else                      // в протилежному випадку
   {
        min=0;               // обнуління змінної
        if(hour<23) hour++;  // якщо значення годин менше 23, інкремент змінної
        else hour=0;         // в протилежному випадку обнуління змінної
   }}
  PORTA^=0b10000000;         // зміна стану сигналу на виводі PA7
  delay(500);                // затримка 500 мс
  PORTA^=0b10000000;         // зміна стану сигналу на виводі PA7
  delay(500);                // затримка 500 мс
 }
}
```

Коди цифр для відображення на РКІ зберігаються в масиві seg в прямому коді. Оскільки для запалення сегментів необхідно подати на відповідний вхід РКІ низький рівень, тому перед виведенням елементів масиву в порт виконується операція інвертування «~».

На рис. 4.12 представлено результат виконання програми.

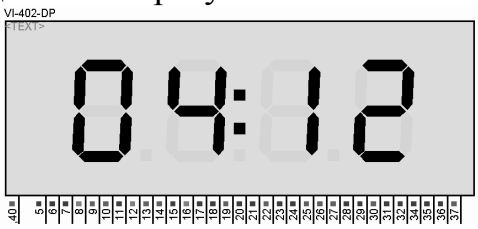


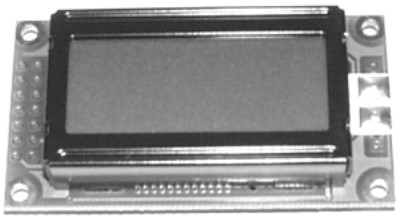
Рис. 4.12.

4.4. Алфавітно-цифрові РКІ

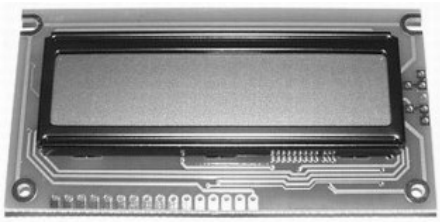
Алфавітно-цифрові РКІ-модулі є недорогим і зручним рішенням при розробці нових виробів, при цьому забезпечують відображення великого об'єму інформації при низькому енергоспоживанні. Можливість оснащення РКІ-модулів заднім підсвічуванням дозволяє експлуатувати їх в умовах із зниженою або нульовою освітленістю, а виконання з розширеним діапазоном температур (–20°С...+70°С) в складних експлуатаційних умовах.

Існує декілька різних стандартних форматів РКІ-модулів (символів × рядків): 8×2, 16×1, 16×2, 16×4, 20×1, 20×2, 20×4, 24×2, 40×2, 40×4. Зустрічаються і менш поширені формати: 8×1, 12×2, 32×2 та інші. Модуль може мати будь-яку кількість символів від 1 до 80, хоча в деяких комбінаціях програмна адресація символів може виявитися не дуже зручною.

В рамках одного формату можуть вироблятися РКІ-модулі декількох конструктивів, що відрізняються як габаритами РКІ (і, як наслідок, розмірами символів), так і розмірами плати і посадки. На рис. 4.13 представлені деякі моделі алфавітно-цифрових РКІ.



WH0802A (8 символів на 2 рядки)



WH1602A (16 символів на 2 рядки)



WH1602D (16 символів на 2 рядки)



WH2002L (20 символів на 2 рядки)



WH2004L (20 символів на 4 рядки)



MT-10S1-1YG (10 символів на 1 рядок)

Рис. 4.13.

В рамках одного конструктиву РКІ-модуль може мати ще ряд модифікацій. Зокрема, можуть застосовуватися декілька типів РКІ, що відрізняються кольором фону і кольором символів, а також по використаних РК-матеріалах і структурі: TN, STN і FSTN типу. РКІ STN і FSTN типу мають вищу вартість, але одночасно мають підвищену контрастність і вдвічі більший максимальний кут огляду, причому РКІ FSTN типу мають кращі характеристики, ніж STN.

При виробництві алфавітно-цифрових РКІ-модулів широко застосовується контролер HD44780 фірми Hitachi. Контроллер HD44780 потенційно може управляти 2-ма рядками по 40 символів в кожному (для модулів з 4-ма рядками по 40 символів використовуються два однотипні контролери), при матриці символу 5×7 точок. Контролер також підтримує символи з матрицею 5×10 точок, але останніми роками РКІ-модулі з такою матрицею практично не зустрічаються.



Рис. 4.14.

Спрощена структурна схема контролера приведена на рис. 4.14. Модуль містить регістр даних (DR), регістр команд (IR), відеопам'ять (DDRAM), ОЗП знакогенератора (CGRAM), лічильник адреси пам'яті (AC), прапор зайнятості контролера.

Інші елементи не є об'єктом прямої взаємодії з управляючою програмою – вони беруть участь в процесі регенерації зображення на РКІ: знакогенератор, формувач курсора, регістри зсуву і драйвери.

Управління контролером відбувається за допомогою інтерфейсу управляючої системи. Основними об'єктами взаємодії є регістри DR і IR. Вибір регістра, що адресується, здійснюється лінією RS, якщо RS = 0 – адресується регістр команд (IR), якщо RS = 1 – адресується регістр даних (DR).

Дані через регістр DR, залежно від поточного режиму, можуть розташовуватися (або зчитуватися) у відеопам'ять (DDRAM), або в ОЗП знакогенератора (CGRAM) за поточною адресою, що вказується лічильником адреси (ЛА). Інформація, що потрапляє в регістр IR, інтерпретується пристроєм виконання команд як управляюча послідовність. Зчитування регістра IR повертає в 7-ми молодших розрядах поточне значення лічильника ЛА, а в старшому розряді прапор зайнятості (BF).

4.4.1. Сполучення РКІ з мікроконтролером

Для підключення РКІ-модуля до управляючої системи використовується паралельна синхронна шина DB0...DB7, яка може бути 8-и або 4-х бітною (вибирається програмно), лінія вибору операції зчитування/запису R/W, лінія вибору регістра RS і лінія синхронізації E. Крім ліній управляючої шини є дві лінії для подачі напруги живлення 5В – GND (V_{SS}) і V_{CC}, а також лінія для управління контрастністю – V_{EE}.

Назви ліній шини є стандартними, але існує безліч різних варіантів розташування контактів в кожного конкретного конструктиву РКІ-модуля. Стандартним варіантом розташування контактів є дворядне 14-и контактне поле (табл. 4.3), розташоване вертикально в лівій частині модуля, а також співпадаюче з ним дворядне 16-ти контактне поле, що містить додаткову пару контактів з підключеними до неї виводами живлення для підсвічування. Для отримання достовірної інформації необхідно скористатися відповідною довідковою літературою виробника модуля.

Табл. 4.3.

N	Назва виводу	Опис
1	V_{SS}	(–) Живлення, 0 В.
2	V_{DD}	(+) Живлення, +5 В.
3	V_{EE} (V_0)	Напруга зміщення, якою регулюється контрастність
4	RS	Вхід. Високий рівень – дані, низький рівень – команди

5	R/W	Вхід. Високий рівень – читання, низький рівень – запис
6	E	Вхід. Строб, який супроводжує сигнали на шині "команда/дані"
7	DB_0	Шина даних (8-бітний режим)
8	DB_1	Шина даних (8-бітний режим)
9	DB_2	Шина даних (8-бітний режим)
10	DB_3	Шина даних (8-бітний режим)
11	DB_4	Шина даних (8-и і 4-х бітний режими, молодший біт)
12	DB_5	Шина даних (8-и і 4-х бітний режими)
13	DB_6	Шина даних (8-и і 4-х бітний режими)
14	DB_7	Шина даних (8-и і 4-х бітний режими, старший біт)

На початковому етапі необхідно подати живлення на РКІ-модуль і добитися від нього ознак працездатності. Схема включення модуля, показана на рис. 4.15.

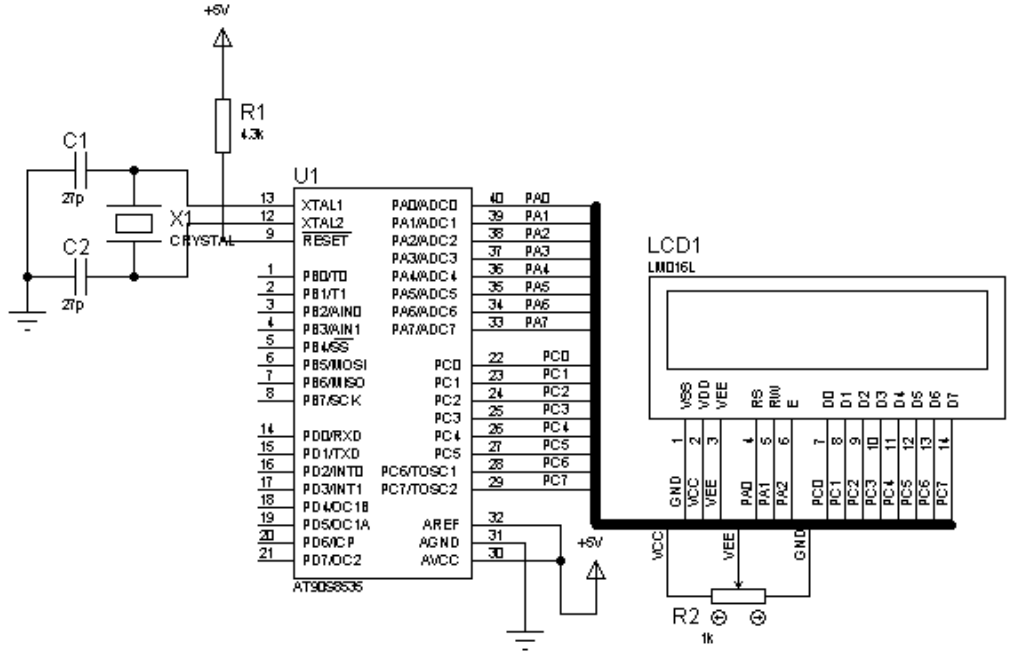


Рис. 4.15.

Змінний резистор R2 дозволяє плавно змінювати напругу живлення драйвера РКІ, що призводить до зміни кута повороту рідких кристалів. Цим резистором можна відрегулювати фактичну контрастність при деякому переважаючому куті спостереження (знизу-догори або згори донизу).

Після закінчення циклу внутрішньої ініціалізації модуль включається в режим розгортки одного верхнього рядка. При зміні напруги на виводі V_{EE} сегменти цього рядка повинні міняти свій стан від прозорого до непрозорого, що свідчить про правильне підключення живлення модуля і працездатності контролера й драйверів РКІ.

Послідовність дій, які необхідно виконувати управляючій системі при здійсненні операцій запису і зчитування для 8-ми розрядної шини демонструє часова діаграма, показана на рис. 4.16.

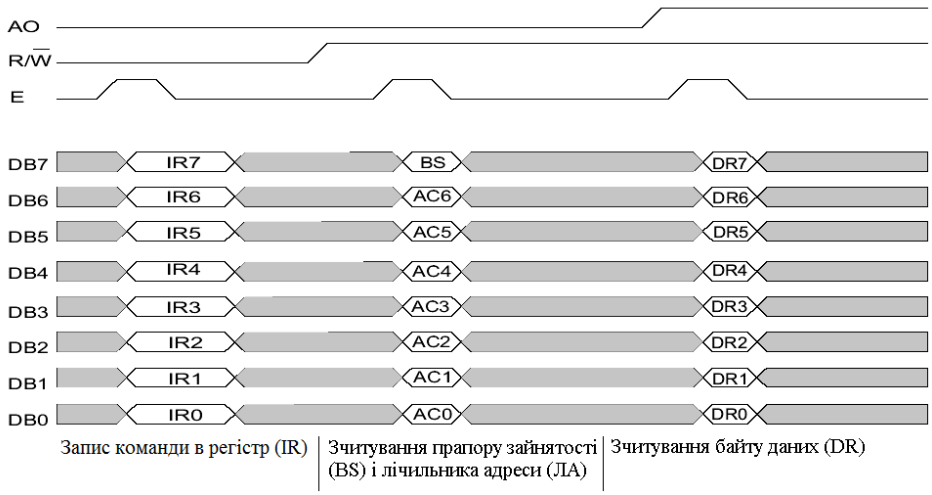


Рис. 4.16.

Операції запису для 8-ми разрядної шини полягають в наступному:
1. Встановити значення лінії RS.
2. Вивести значення байта даних на лінії шини DB0...DB7.
3. Встановити лінію E = 1.
4. Встановити лінію E = 0.
5. Встановити лінії шини DB0...DB7 = HI.

Для зчитування значення з модуля виконуються наступні операції:
1. Встановити значення лінії RS.
2. Встановити лінію R/W = 1.
3. Встановити лінію E = 1.
4. Зчитати значення байта даних з ліній шини DB0...DB7.
5. Встановити лінію E = 0.
6. Встановити лінію R/W = 0.

Діаграма обміну даними з використанням 4-х бітного інтерфейсу представлена на рис. 4.17.

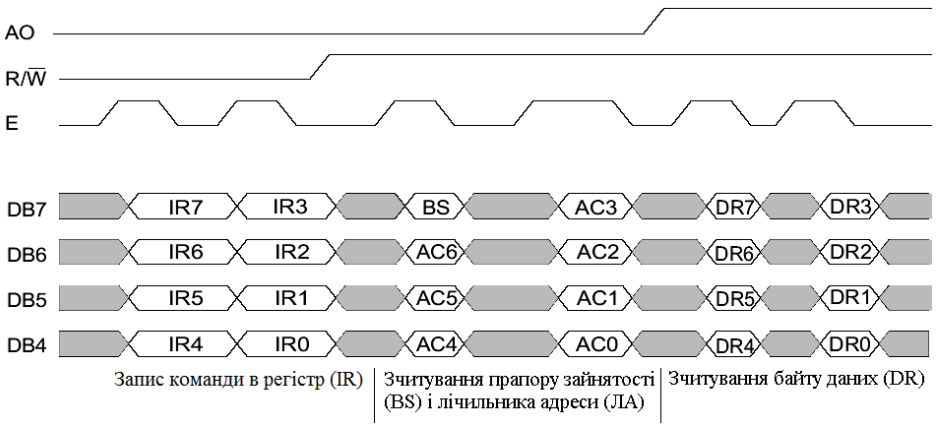


Рис. 4.17.

На рис. 4.18 представлена схема сполучення РКІ з мікроконтролером по 4-х бітній шині.

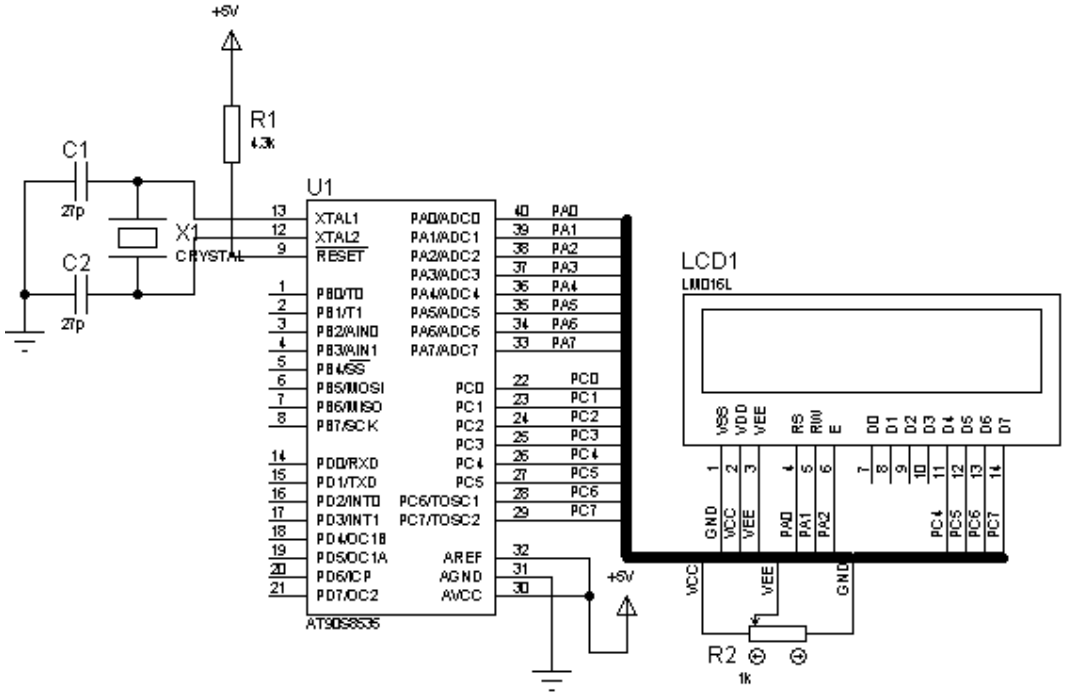


Рис.4.18.

Операція запису для 4-х розрядної шини:

1. Встановити значення лінії RS.

2. Вивести значення старшої тетради байта даних на лінії шини DB4...DB7.

3. Встановити лінію E = 1.

4. Встановити лінію E = 0.

5. Вивести значення молодшої тетради байта даних на лінії шини DB4...DB7.

6. Встановити лінію E = 1.

7. Встановити лінію E = 0.

8. Встановити лінії шини DB4...DB7 = HI.

Операція зчитування для 4-х розрядної шини:

1. Встановити значення лінії RS.
2. Встановити лінію R/W = 1.
3. Встановити лінію E = 1.
4. Зчитати значення старшої тетради байта даних на лініях шини DB4...DB7.
5. Встановити лінію E = 0.
6. Встановити лінію E = 1.
7. Зчитати значення молодшої тетради байта даних на лініях шини DB4...DB7.
8. Встановити лінію E = 0.
9. Встановити лінію R/W = 0.

У кожному циклі обміну даними необхідно передавати всі 8 біт – двічі по 4 біта. Передача старших 4-х біт без подальшої передачі молодших 4-х біт не допускається.

При виконанні операцій запису і зчитування необхідно ретельно контролювати мінімальні значення часових інтервалів, щоб вони завжди знаходилися в області допустимих значень.

Описані вище операції запису/зчитування байта є базовими для здійснення обміну з РКІ-модулем.

4.4.2. Система команд РКІ на основі контролера HD44780

Програмне управління рідкокристалічним індикатором здійснюється за допомогою системи команд, що наведена в табл. 4.4.

Табл. 4.4.

Код										Описання команди	Час виконання команди (f_{osc}= 250 кГц)
RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
0	0	0	0	0	0	0	0	0	1	Очистити дисплей і встановити курсор в нульовую позицію (адреса 0)	до 1,64 мс
0	0	0	0	0	0	0	0	1	*	Встановити курсор в нульовую позицію (адреса 0). Встановити дисплей відносно буфера DDRAM в початкову позицію. Вміст DDRAM при цьому не змінюється.	от 40 мкс до 1,6 мс
0	0	0	0	0	0	0	1	I/D	S	Встановити напрям зсуву курсора праворуч (I/D=1) або ліворуч (I/D=0) при записі/зчитуванні наступного коду в DDRAM. Дозволити (S=1) зсув дисплея разом із зсувом курсора.	40 мкс

0	0	0	0	0	0	1	D	C	B	Увімкнути(D=1)/вимкнути (D=0) дисплей. Запалити (C=1)/погасити (C=0) курсор. Зображення курсора зробити мігаючим (B=1).	40 мкс
0	0	0	0	0	1	S/C	R/L	*	*	Перемістити курсор (S/C=0) або зсунути дисплей (S/C=1) праворуч (R/L=1), або ліворуч (R/L=0).	40 мкс
0	0	0	0	1	DL	N	F	*	*	Встановити розрядність шини даних 4 біта (DL=0) або 8 біт (DL=1), кількість рядків дисплея - один (N=0) або два (N=1), шрифт - 5×7 точок (F=0) або 5×10 точок (F=1).	40 мкс
0	0	0	1	A_{CG}						Встановлення адреси CGRAM. Після цієї команди дані будуть записуватися/зчитуватись в/із CGRAM.	40 мкс
0	0	1	A_{DD}							Встановлення адреси DDRAM. Після цієї команди дані будуть записуватися/зчитуватись в/із DDRAM.	40 мкс
0	1	BF	AC							Зчитування стану busy-прапору (BF) та лічильника адреси.	1 мкс
1	0	Дані								Запис даних в DDRAM або CGRAM.	40 мкс
1	1	Дані								Зчитування даних з DDRAM или CGRAM.	40 мкс

DDRAM – Display data RAM – ОЗП ASCII-кодів символів, що відображаються

CGRAM – Character generator RAM – ОЗП знакогенератора

* – означає будь-яке значення

Модуль містить ОЗП для зберігання даних (DDRAM), що виводяться на екран. Розподіл адрес модуля представлено на рис. 4.19.

№ Знакомісця		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
АДРЕСА	1-й рядок	0h	1h	2h	3h	4h	5h	6h	7h	8h	9h	0Ah	0Bh	0Ch	0Dh	0Eh	0Fh
	2-й рядок	40h	41h	42h	43h	44h	45h	46h	47h	48h	49h	4Ah	4Bh	4Ch	4Dh	4Eh	4Fh

Рис. 4.19.

Відеопам'ять, що має загальний об'єм 80 байтів, призначена для зберігання кодів символів, що відображуються на РКІ. Відеопам'ять організована в два рядки по 40 символів в кожному. Ця прив'язка є жорсткою і не підлягає зміні.

Приклад 4.2. Написати програму для виведення статичного тексту на алфавітно-цифровий рідкокристалічний індикатор по 8-бітовій шині. Принципова схема представлена на рис. 4.15.

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>                    // підключення заголовочних файлів
 #include <macros.h>                    // бібліотека макросів
 #include <string.h>                    // бібліотека функцій для роботи з рядками
 #define RS() PORTA|=0b00000001         // встановити сигнал команда/дані
 #define NRS() PORTA&=~0b00000001       // скинути сигнал команда/дані
 #define RW() PORTA|=0b00000010         // встановити сигнал зчитування/запис
 #define NRW() PORTA&=~0b00000010       // скинути сигнал зчитування/запис
 #define E() PORTA|=0b00000100          // встановити сигнал стробу
 #define NE() PORTA&=~0b00000100        // скинути сигнал стробу
 #define LCD PORTC                      // РКІ модуль підключений до порту С
 #define DDLCD DDRC                     // псевдонім регістру напрямку даних
// оголошення прототипів функцій
 void init_cpu();            // функція ініціалізації мікроконтролера
 void pause();               // функція часової затримки
 void init_lcd();            // функція ініціалізації РКІ
 void write();               // функція, при виклику якої формується тактовий сигнал
 // реализація функцій
 void init_cpu()             // функція ініціалізації мікроконтролера
 {
 DDRA=0b00000111;            // налаштування трьох молодших біт порту А на вивід
 DDLCD=0x00;                 // порт, до якого підключений РКІ, переводиться
                             // в стан високого імпеданса
 NRS();                      // скидання сигналу команда/дані
 NRW();                      // скидання сигналу читання/запис
 NE();                       // скидання тактового сигналу
 }
 void pause()                // функція формування часової затримки
 {
 int tmp;                    // додаткова змінна
 for(tmp=0;tmp<350;tmp=tmp+1) NOP();   // у циклі контролер виконує пусту операцію
 }
 void init_lcd()             // функція ініціалізації РКІ
 {
 DDLCD=0xFF;                 // налаштування порту на виведення даних
 LCD=0x38;                   // виведення команди 0x38 на порт контроллера
 write();                    // запис команди в модуль
 LCD=0x0C;                   // виведення команди 0x0C на порт контролера
 write();                    // запис команди в модуль
 LCD=0x06;                   // виведення команди 0x06 на порт контролера
 write();                    // запис команди в модуль
 LCD=0x02;                   // виведення команди 0x02 на порт контролера
 write();                    // запис команди в модуль
  LCD=0x01;                  // виведення значения 0 на порт контролера
 write();
 }
```

```
void write()                    // функція, яка видає тактовий сигнал
{
 E();                           // установка тактового сигналу
 pause();                       // часова затримка
 NE();                          // скидання тактового сигналу
 pause();                       // часова задержка
}
void main()                     // основна програма
{
char text1[]="Hello students!";       // оголошення змінних,
char text2[]="How are you?"; // інформація, яка буде виводитися на екран РКІ
int index;                      // змінна для використання у циклі
init_cpu();                     // виклик функції ініціалізації мікроконтролера
init_lcd();                     // виклик функції ініціалізації РКІ модуля
DDLCD=0xFF;                     // налаштування порту на виведення даних
RS();                  // установка в 1 сигналу команда/дані; в РКІ передаються дані
for(index=0; index<strlen(text1); index++) // у циклі виконується запис даних в РКІ
{
 LCD=text1[index];              // установка ASCII-коду символу на порту
 write();               // запис символу в РКІ; курсор зсовується автоматично
}
NRS();                 // скидання сигналу команда/дані; в РКІ записуються команди
LCD=0;                 // установка значення 0 на порту
DDLCD=0x00;            // переведення порту в стан високого імпеданса
while(1);              // зациклювання програми
}
```

На рис. 4.20 представлено результат виконання програми.

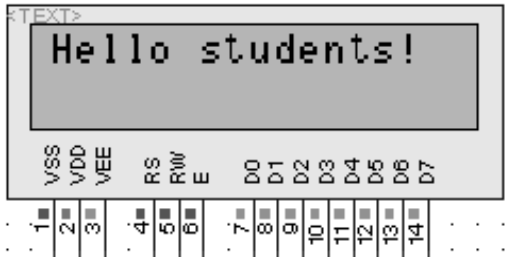


Рис. 4.20.

Приклад 4.3. Написати програму для мікроконтролера, яка виводить на рідкокристалічному індикаторі біжучий рядок. Обмін даними з РКІ-модулем виконується по 4-х бітній шині.

```
#include <io8535v.h>                // підключення заголовочних файлів
#include <macros.h>                 // бібліотека макросів
#include <string.h>                 // бібліотека функцій для роботи з рядками
#define RS() PORTA|=0b00000001          // сигнал команда/дані
#define NRS() PORTA&=~0b00000001
#define RW() PORTA|=0b00000010          // сигнал зчитування/запис
#define NRW() PORTA&=~0b00000010
#define E() PORTA|=0b00000100           // сигнал стробування
```

```
#define NE() PORTA&=~0b00000100
#define LCD PORTC
#define DDLCD DDRC                  // псевдонім регістру напрямку даних
 #define display 0x10               // довжина видимої частини дисплея
// оголошення прототипів функцій
 void init_cpu();                   // функція ініціалізації мікроконтролера
 void pause();                      // часова затримка
 void init_lcd();                   // функція ініціалізації РКІ
 void write();                      // функція що формує тактовий сигнал
 void delay(int ms);                // функція затримки
 void less_bit(int in);             // функція передачі даних по 4-х бітній шині
 // реализація функцій
 void less_bit(int in)
 {
 int second, first;                  //старші і молодші біти відповідно
 second=in;
 first=in<<4;                       // зсув на 4 розряди ліворуч
 LCD=second;                        // передача молодшої тетради
 write();                           // формування тактового імпульса
 LCD=first;                         // передача старшої тетради
 write();                           // формування тактового імпульса
 }
 void init_cpu()                    // функція ініціалізації мікроконтролера
 {
 DDRA=0b00000111;          // налаштування 3 молодших виводів порту А на вихід
 NRS();                             // скидання сигналу RS
 NRW();                             // скидання сигналу RW
 NE();                              // скидання сигналу E
 }
 void delay(int ms)                 // функція затримки
 {
 int temp,temp1;                    // додаткові змінні
 for(temp=0;temp<ms;temp++)         // у циклі виконується «пуста» операція
 for(temp1=0;temp1<970;temp1++) NOP();
 }
 void pause()                       // функція затримки
 {
 int temp;
 for(temp=0;temp<1000;temp++) NOP();
 }
 void init_lcd()                    // функція ініціалізації РКІ
 {
 DDLCD=0xFF;                        // налаштування порту С на вихід
 LCD=0b00100000;                    // команда: кількість рядків 1, ширина шини 4 біта,
                                    // розмір шрифта 5x7
write();
less_bit(0x0c);// включити дисплей, вимкнути курсор, вимкнути мерехтіння курсора
less_bit (0x07);
 }
 void write()                       // функція, яка формує тактовий сигнал
 {
```

```
 E();                                  // установка тактового сигналу
 pause();                              // часова затримка
 NE();                                 // скидання тактового сигналу
 pause();                              // часова затримка
 }
 void main()                           // головна функція програми
{
 char text[]="Kathedra TEES ";         // масив з текстом, який відобразиться на РКІ
 char i;
 init_cpu();                           // виклик функції ініціалізації мікроконтролера
 init_lcd();                           // виклик функції ініціалізації РКІ
 while(1)                              // вічний цикл
 {
 NRS();                                // передаються команди
 less_bit (0x80+display);              // зсув курсора
 RS();                                 // передаються дані
for(i=0; i<(strlen(text)+display);i++)// передача масиву на РКІ
 {
 if(i<strlen(text)){
 less_bit (text[i]);
 delay(200);                           // затримка 200 мс
 }
 else{
 less_bit (' ');
 delay(200);                           // затримка 200 мс
 }
 }
 NRS();                                // скидання сигналу RS
 less_bit (0x01);
 less_bit (0x02);
 } }
```

Функція less_bit виконує запис в РКІ-модуль спочатку старшої, а потім молодшої тетради символу, що виводиться. У цій функції використовується операція зсуву. Для того, щоб індикатор встиг відпрацювати команди, після кожної операції запису даних формується часова затримка. Якщо цього не виконати, на екрані РКІ можуть не відображуватися деякі символи. На рис. 4.21 показаний результат виконання програми.

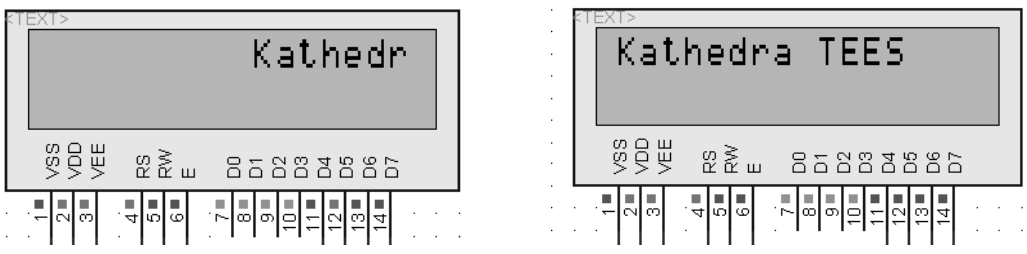


Рис. 4.21.

4.4.3. Принцип формування символів

Модуль дозволяє задати зображення 8 додаткових символів знакогенератора. Якщо мікроконтролер передає в контролер РКІ дані, що представляють собою ASCII-коди символів, які відображуються, то вони записуються в буфер даних (DDRAM). Окрім DDRAM, контролер РКІ містить ще один блок пам'яті – знакогенератор. Основна його частина є ПЗП (CGROM), в якій знаходяться зображення символів, і її не можна змінити. Друга частина, в якій задаються зображення символів для 16-ти перших кодів таблиці знакогенератора, є ОЗП (CGRAM), який можна перепрограмувати. Є можливість задати зображення 8 символів, які відповідають кодам 0(8), 1(9), 2(10)...7(15). Для кожного з восьми перепрограмованих символів в CGRAM відводиться по 8 комірок пам'яті, кожна з яких відповідає одному рядку точок в зображенні символу. Таким чином, перепрограмована частина знакогенератора містить 64 байти пам'яті (8×8). Приклад кодування CGRAM для однієї букви наведений на рис. 4.22.

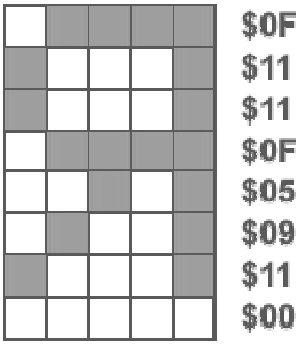


Рис. 4.22.

Для перепрограмування доступні 8 символів з матрицею 5×7 крапок і 4 з матрицею 5×10 крапок (у цьому режимі символи адресуються кодами DDRAM через один). Для кодування матриці використовуються 8 горизонтально розташованих байт, які несуть інформацію про малюнок (1 означає, що сегмент буде включеним).

Одним із способів представлення значення деякої величини – виведення значення параметра у вигляді шкали. Кількість сегментів, які будуть включені, пропорційно значенню параметра (по аналогії з шкальними індикаторами). Для цього необхідно сформувати елемент зображення шкали, наприклад у вигляді замальованого прямокутника. Приклад, наведений нижче, демонструє основні операції при роботі з модулем для представлення даних різними способами. У прикладі використовується сформований символ.

Приклад 4.4. Написати програму, яка виводить на верхньому рядку екрану РКІ лічильник (0-15). В нижньому рядку РКІ реалізувати шкальний

індикатор. Кількість включених сегментів повинна дорівнювати значенню лічильника.

Текст програми з коментарями представлено нижче.

```
#include <io8535v.h>          // підключення заголовочних файіов
 #include <macros.h>
 #include <string.h>
 #include <stdlib.h>
 // визначення псевдонімов портів і управляючих
 #define RS() PORTA|=0b00000001
 #define NRS() PORTA&=~0b00000001
 #define RW() PORTA|=0b00000010
 #define NRW() PORTA&=~0b00000010
 #define E() PORTA|=0b00000100
 #define NE() PORTA&=~0b00000100
 #define LCD PORTC
 #define DDLCD DDRC
// оголошення прототипів функцій
 void init_cpu();             // функція ініціалізації мікроконтролера
 void pause();                // функція затримки
 void init_lcd();             // функція ініціалізації РК-індикатора
 void write();                // функція запису команд/даних в РКІ
 void delay_ms(int ms);       // функція затримки в мс
 // реализация функций
 void init_cpu()              // функція ініціалізації мікроконтролера
 {
 DDRA=0b00000111;             // налаштування трьох старших біт порту А на вивід
 DDLCD=0x00;                  // переведення порту, до якого підключений РКІ, у стан
                              // високого імпедансу
 NRS();                       // скидання сигналу RS (команда/дані)
 NRW();                       // скидання сигналу RW (зчитування/запис)
 NE();                        // скидання тактового сигналу E
 }
 void pause()                 // функція затримки
{
 int tmp;                     // додаткова змінна для використання у циклі
 for(tmp=0;tmp<450;tmp=tmp+1) NOP(); // контролер виконує пусті операції
 }
 void init_lcd()              // функція ініціалізації РКІ
 {
 DDLCD=0xFF;                  // налаштування порту, до якого підключений РКІ, на вихід
 NRS();                       // скидання сигналу команда/дані – передаються команди
 NRW();                       // скидання сигналу запис/зчитування – виконується запис
 LCD=0x38;                    // виведення команди 0x38 в порт контролера
 write();                     // запис команди в модуль
 LCD=0x0C;                    // виведення команди 0x0C в порт контролера
 write();                     // запис команди в модуль
 LCD=0x06;                    // виведення команди 0x06 в порт контролера
 write();                     // запис команди в модуль
 LCD=0x02;                    // виведення команди 0x02 в порт контролера
 write();                     // запис команди в модуль
```

```
 LCD=0x00;                  // виведення значення 0 в порт контролера
 DDLCD=0x00;                // переведення порту в стан високого імпеданса
 }
 void write()               // функція формування тактового сигналу
{
  E();                      // установка тактового сигналу
  pause();                  // часова затримка
  NE();                     // скидання тактового сигналу
  pause();                  // часова затримка
 }
void delay_ms(int ms)       // функція часової затримки: параметр – затримки в мс
 {
        char i, j;          // змінні для використання в циклах
        int k;
        for(k=0;k<ms;k++)   // у циклі мікроконтролер не виконує ніяких операцій
        {
         for(i=0;i<20;i++)
         for(j=0;j<71;j++);
        }
 }
void main()                 // основна програма
 {
 int index, i;              // змінні для використання в циклах
 char text[15];             // рядок довжиною 15 символів
 char text1[]="Value is ";  // рядок з початковою ініціалізацією
 init_cpu();                // виклик функції ініціалізації мікроконтролера
 init_lcd();                // виклик функції ініціалізації РКІ
 DDLCD=0xFF;                // налаштування порту, до якого підключений РКІ, на вихід
 LCD=0x40;                  // установка адреси CGRAM
 write();                   // запис команди в РКІ
 RS();                      // установка сигналу RS – передаються дані
 for(i=0;i<8;i=i+1)         // у циклі записується 8 байт в CGRAM
 {
  LCD=0xFF; // виведення значення в порт – створюється символ чорного прямокутника
  write();                  // запис даних в CGRAM
 }
 NRS();                     // скидання сигналу RS – передаються команди
 LCD=0x80;                  // установка адреси DDRAM
 write();                   // запис команди в РКІ
 while(1)                   // вічний цикл
 {
 for(index=0; index<16;index=index+1)     // цикл виконується 16 раз
 {
  itoa(text,index+1,10);    // перетворення числа index+1 в текстовий рядок text
  LCD=0x01;                 // установка команди очищення дисплея. Курсор в позиції 0
  write();                  // запис команди в порт
  RS();                     // установка сигналу RS – передаються дані
  for(i=0;i<sizeof(text1)-1;i=i+1)     // у циклі передається рядок в РКІ
  {
   LCD=text1[i];            // виведення ASCII-коду символа в порт
   write();                 // запис даних в рідкокристалічний модуль
```

```
 }
 if(index<10)                  // якщо одна цифра (0-9)
 {
  LCD=text[0];                 // виведення ASCII-коду 0-го елементу масива в порт
  write();                     // запис даних в порт
 }
 else                          // якщо 2 цифри (10-15)
 {
  LCD=text[0];                 // виведення ASCII-коду 0-го елементу масива в порт
  write();                     // запис даних в порт
  LCD=text[1];                 // виведення ASCII-коду 1-го елементу масива в порт
  write();                     // запис даних в порт
 }
 NRS();                        // скидання сигналу RS – передаються команди
 LCD=0b10101000;               // команда установки адреси DDRAM в позицію 40
 write();                      // запис команди в РКІ
 RS();                         // установка сигналу RS – передаються дані
 for(i=0;i<=index;i=i+1)       // у циклі виводиться index символів з нульової комірки
CGRAM
  {
   LCD=0;                      // виведення значення 0 в порт
   write();                    // запис даних в РКІ
  }
 NRS();                        // скидання сигналу RS – передаються команди
 delay_ms(500);                // затримка 500 мс
}
NRS();                         // скидання сигналу RS – передаються команди
} }
```

У програмі за допомогою директив #define операціям з портами привласнені відповідні імена. Це зручно, оскільки програмісту немає необхідності запам'ятовувати назви виводів мікроконтролера, які сполучені з управляючими входами РКІ. Крім того, це прискорює процес написання програми, а також дозволяє швидко змінити програму в разі зміни принципової схеми. В цьому випадку досить поміняти назви портів і номери виводів. В даному прикладі використовується знакогенератор. У пам'ять CGRAM записані значення, які формують чорний прямокутник, – один сегмент шкального індикатора, які виводяться в другому рядку екрану. Результат виконання програми представлений на рис. 4.23.

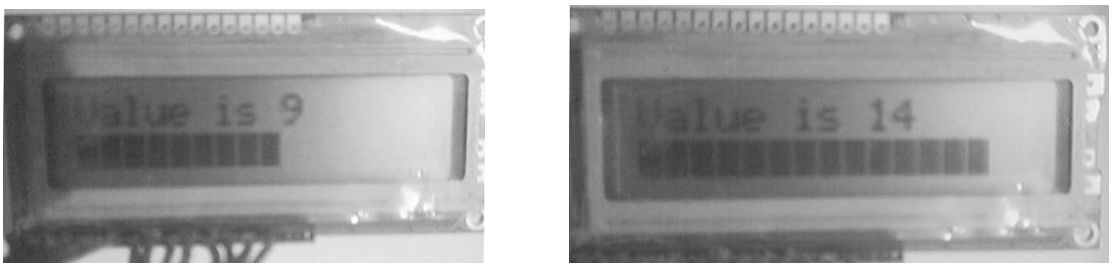


Рис. 4.23.

При вимірюванні значення якого-небудь параметра, перед виведенням на РКІ отримані дані заздалегідь необхідно обробити. У рідкокристалічний модуль необхідно передавати ASCI-коді тих символів, які, власне, і відображатимуться на екрані. Наприклад, для виведення на РКІ числа „1234” необхідно в контролер передати ASCII-коди цифр 1, 2, 3 і 4. Тому, для того, щоб вивести число на індикатор, його потрібно перетворити в текстовий рядок. Для цього можна використовувати функції зі стандартної бібліотеки <stdlib.h>. В табл. 4.5 розглянуті деякі функції з цієї бібліотеки.

Табл. 4.5.

Функція	Описання
char *ftoa(**float** f, **int** *status)	Перетворює число з плаваючою точкою в рядок символів. Функція повертає вказівник на буфер розміром 15 байт. Змінна status містить код виконання функції. Значення 0 означає вдале виконання. Якщо значення цієї змінної дорівнює _FTOA_TOO_LARGE або _FTOA_TOO_SMALL, це означає, що передане число f виходить за рамки діапазону.
void itoa(**char** *buf, **int** value, **int** base)	Перетворює цілочисельне значення value в рядок символів. Рядок повертається в масиві, вказівник на який – змінна buf. Змінна base – основа системи числення, в якій буде представлено перетворене число (2 – двійкова, 10 – десяткова, 16 – шістнадцяткова).
void ltoa(**char** *buf, **long** value, **int** base)	Перетворює значення value типу long в рядок символів. Рядок повертається в масиві, вказівник на який – змінна buf. Змінна base – основа системи числення, в якій буде представлено перетворене число.
int atoi(**char** *s)	Функція перетворює рядок символів s в число типу **int**. Функція повертає 0 якщо виникає помилка.
double atof(**const char** *s)	Функція перетворює рядок символів s в число типу **double**. Функція повертає 0 якщо виникає помилка.
long atol(**char** *s)	Функція перетворює рядок символів s в число типу **long**. Функція повертає 0 якщо виникає помилка.

Приклад 4.5. Написати програму для мікроконтролера, яка виводить на рідкокристалічному індикаторі біжучий рядок. Обмін даними з РКІ-модулем виконується по 8-бітній шині.

Текст програми для мікроконтролера представлено нижче.

```
#include<io8535v.h>                 // підключення заголовочних файлів
 #include<macros.h>                 // бібліотека макросів
 #define RS() PORTA|=0x01           // встановити сигнал команда/дані
 #define NRS() PORTA&=~0x01         // скинути сигнал команда/дані
 #define RW() PORTA|=0x02           // встановити сигнал зчитування/запис
 #define NRW() PORTA&=~0x02         // скинути сигнал зчитування/запис
 #define E() PORTA|=0x04            // встановити сигнал стробу (біт 7 портуА)
 #define NE() PORTA&=~0x04          // скинути сигнал стробу
 // оголошення прототипів функцій
 void init_cpu(void);               // функція ініціалізації контролера
 void pause();                      // часова затримка
 void init_lcd(void);               // функция ініціалізації РКІ
 void delay(int ms);
 // реалізація функцій
 void init_cpu()                    // функція ініціалізації контролера
 {
 DDRA=0b00000111;                   // налаштування трьох мол. біт порту А на вихід
 DDRC=0xFF;
 NRS();                             // скидання сигналу команда/дані
 NRW();                             // скидання сигналу зчитування/запис
 NE();                              // скидання тактового сигналу
 }
 void pause()                       // функція часової затримки
 {
 int t;                             // часова змінна
 for (t=0;t<800;t++)NOP();          // у циклі виконується пуста операція
 }
 void write()                       // функція, яка формує тактовий сигнал
 {
 E();                               // установка тактового сигналу
 pause();                           // часова затримка
 NE();                              // скидання тактового сигналу
 pause();                           // часова затримка
 }

 void init_lcd()                    // функція ініціалізації РКІ
 {
 PORTC=0b00111000;                  // команда – вст. розрядність шини 8 біт,
                                    // 2 рядки, шрифт 5×7 точок
 write();                           // запис команди в модуль
 PORTC=0b00001100;                  // команда «включити дисплей»,
                                    // погасити курсор
 write();
 PORTC=0x06;                        // команда дозволу зсуву курсора праворуч
                                    // заборонити зсув дисплея з курсором
 write();
 PORTC=0x01;                        // команда очистити дисплей і встановити
                                    // курсор в нульову позицію
```

```
write();
}
void delay(int ms)                  // функція затримки
{
int t1,t2;                          // додаткові змінні
for (t1=0;t1<ms;t1++)               // у циклі виконується «пуста» операція
for (t2=0;t2<970;t2++)NOP();
}
void main()                         // основна програма
{
char text1[]="Running string";      // оголошення змінних
int i=0;                            // змінна для використання в циклі
init_cpu();                         // виклик функції ініціалізації контролера
init_lcd();                         // виклик функції ініціалізації РКІ -модуля
RS();                               // установка в 1 сигнала команда/дані;
                                    // в РКІ передаються дані
pause();                            // затримка
for(i=0;i<15;i++)                   // у циклі виконується запис даних в РКІ
{
PORTC=text1[i];                     // установка ASCII-коду символу на порту С
write();                            // запис символу в РКІ
}
NRS();                              // скидання сигналу команда/дані;
                                    // в РКІ записуються команди
while(1)
{
PORTC=0b00011100;                   // команда зсуву дисплея на одну позицію
write();                            // формування тактового сигналу
delay(300);                         // затримка, регулюється швидкість руху рядка
}
}
```

У програмі функція init_cpu() виконує ініціалізацію мікроконтролера. У цій функції налаштовуються 3 молодших біта порту А і порт С на вихід, а також встановлюються сигнали низького рівня на виводах PA0-PA2. Функція init_lcd() виконує ініціалізацію РКІ-модуля: включення, установка 8-розрядної шини, режим відображення курсора, очищення дисплея і установка курсора в початкову позицію. Для створення ефекту «біжучого рядка» в циклі виконується зсув екрану праворуч. На рис. 4.24 представлений результат виконання програми.

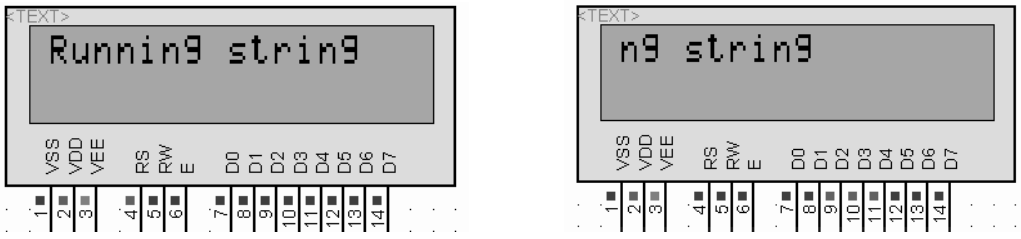


Рис. 4.24.

Пример 4.6. Написати програму, яка виводить на екрані РКІ поточний час.

Текст програми представлено нижче.

```
#include<io8535v.h>                  // підключення заголовочних файлів
#include<macros.h>
#include<stdlib.h>
#define RS() PORTA|=0x01
#define NRS() PORTA&=~0x01
#define NRW() PORTA&=~0x02
#define E() PORTA|=0x04
#define NE() PORTA&=~0x04
void init_cpu()                      // функція ініціалізації мікроконтролера
  {
  DDRA=0x07;                         // налаштування трьох мол. біт порту А на вихід
  DDRC=0xFF;                         // налаштування порту С на вихід
  NRS();                             // скидання сигналу RS
  NRW();                             // скидання сигналу RW
  NE();                              // скидання сигналу E
  }
void pause()                         // функція затримки
  {
  int t;
  for (t=0;t<500;t++)
  NOP();
  }
void write()                         // функція формування тактового імпульса
  {
  E();                               // установка сигналу E
  pause();                           // пауза
  NE();                              // скидання сигналу E
  pause();                           // пауза
  }
void init_lcd()                      // функція ініціалізації РКІ
  {
  PORTC=0b00111000;                  // виведення команди в порт
  write();                           // запис команди в РКІ
  PORTC=0b00001100;
  write();
  }
void delay(int ms)                   // функція затримки
  {
  int t1,t2;                         // оголошення додаткових змінних
  for(t1=0;t1<ms;t1++)               // у циклі виконується «пуста» операція
  for(t2=0;t2<970;t2++)  NOP();
  }
void main()                          // головна функція програми
  {
  char mas[7];                       // масив, що зберігає перетворене значення
  double time=1257550;               // початкове значення (поточний час)
  int t1,t2,t3,t4,t5,t6,t7=0;
```

```
int f1,f2,f3,f4,f5,f6,f7=0;
int i=0;
init_cpu();                    // виклик функції ініціалізації мікроконтролера
init_lcd();                    // виклик функції ініціалізації РКІ
ltoa(mas,time,10);             // функція, що перетворює цілочисельне
                               // значення time в рядку символів mas
while(1)                       // вічний цикл
  {
  f7=mas[6]+t7;                // значення останньої цифри (милісекунди)
  f6=mas[5]+t6;
  f5=mas[4]+t5;
  f4=mas[3]+t4;
  f3=mas[2]+t3;
  f2=mas[1]+t2;
  f1=mas[0]+t1;
    t7++;                      // початковий лічильник (милісекунди)
  // милісекунди:
  if(f7==0x3A)                 // якщо значення попередньої цифри більше 9
  {
  t7=0;                        // обнуління початкового лічильника
  f7=0x30;                     // обнуління попередньої цифри
  mas[6]=0x30;                 // обнуління елементу масива
t6++;                          // лічильник обнулінь
}
// аналогічно для всіх останніх цифр
if(f6==0x3A)
{
t6=0;
f6=0x30;
mas[5]=0x30;
t5++;
}
if(f5==0x36)
{
t5=0;
f5=0x30;
mas[4]=0x30;
t4++;
}
if(f4==0x3A)
{
t4=0;
f4=0x30;
mas[3]=0x30;
t3++;
}
if(f3==0x36)
{
t3=0;
f3=0x30;
mas[2]=0x30;
```

```
t2++;
}
if(f2==0x3A)
{
t2=0;
f2=0x30;
mas[1]=0x30;
t1++;
}
if((f2==0x34)& (mas[0]==0x32))          // якщо час 24:00:00 – обнуління
{
t2=0;
f2=0x30;
mas[1]=0x30;
t1++;
}
if(f1==0x33)
{
t1=0;
f1=0x30;
mas[0]=0x30;
}
// виведення цифр на індикатор в послідовності (f1,f2:f3,f4:f5,f6:f7)
NRS();
pause();
PORTC=128;              // установка курсора в початкову позицію верхнього рядка
write();
RS();
pause();
PORTC=f1;               // виведення цифри (f1)
write();
NRS();
pause();
PORTC=129;
write();
RS();
pause();
PORTC=f2;               // виведення цифри (f2)
write();
NRS();
pause();
PORTC=130;
write();
RS();
pause();
PORTC=0x3A;
write();
delay(20);              // затримка для початкового лічильника
NRS();
pause();
PORTC=0x01;             // оновлення індикатора
```

```
  write();
  }
}
```

У змінній time зберігається початковий час. Для виділення годин, хвилин, секунд і мілісекунд використовується масив. У програмі відстежується значення годинника. Якщо значення перевищує 23, то відбувається онулення годинника. Аналогічно, якщо значення хвилин і секунд перевищує 59, відбувається онулення цих значень і збільшення значення годин на 1. Оскільки РКІ відображає символи за переданими ASCII-кодами, то для того, щоб вивести цифри на екран РКІ, необхідно передати в модуль не безпосереднє значення цифри, а її ASCII-код. У програмі не передбачена установка часу. При необхідності програма може бути легко змінена. На рис. 4.25 представлений результат виконання програми.

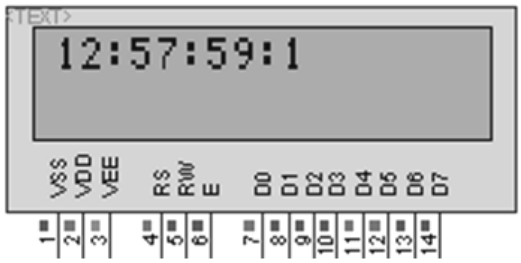


Рис. 4.25.

Приклад 4.7. Розробити принципову схему термометра на базі датчика DS18B20 і написати програму для мікроконтролера. Значення температури виводити на алфавітно-цифровому рідкокристалічному індикаторі.

Широко поширена мікросхема цифрового термометра DS18B20 забезпечує вимірювання температури в діапазоні –55..+125°С. Дозволяюча здатність температури може бути від 9 до 12 розрядів (налаштовується). Інформація із/в DS18B20 передається по однопровідному інтерфейсу. DS18B20 допускає напругу живлення від +3 до +5,5 В. У режимі очікування споживаний струм близький до нуля (менш 1мкА), а під час перетворення температури він складає приблизно 1 мА. Процес перетворення триває максимум 750 мс.

Кожен датчик містить унікальний 48-бітовий серійний номер, записаний при виробництві, тому на одну шину можуть бути підключені декілька датчиків. Це дозволяє розмістити датчики в різних місцях і збирати інформацію по простому 2-х провідному кабелю. Окрім серійного номера, в ПЗП міститься код сімейства (для DS18B20 це 10h) і контрольна сума. Також DS18B20 має температурний датчик, регістр конфігурації, енергозалежні температурні сигнальні тригери TH і TL, проміжний ОЗП об'ємом 8 байт (рис. 4.26).

Байтами TH і TL є температурні пороги, з якими порівнюються 8 біт кожного виміряного значення температури (молодший біт відкидається). За допомогою спеціальної команди можна організувати сигналізацію виходу температури за межі цих порогів. Якщо така функція не потрібна, байти TH і TL можна використовувати для зберігання будь-яких даних користувача.

Зчитування значення виміряної температури, а також передача команди початку перетворення й інших команд виконується за допомогою 1-провідного інтерфейсу (1-Wire). На рис. 4.26 представлена струтура пам'яті DS18B20.

Проміжний ОЗП

Молодший байт температури	**0**
Старший байт температури	**1**
TH/байт користувача 1	**2**
TL/байт користувача 2	**3**
Зарезервовано	**4**
Зарезервовано	**5**
COUNT_REMAIN	**6**
COUNT_PER_C	**7**

Рис. 4.26.

На рис. 4.27 показана апаратна конфігурація інтерфейсної частини DS18B20. У кожного 1-провідного пристрою до шини підключений вхід приймача і вихід передавача з відкритим стоком. Відкритий сток дозволяє підключати до шини багато пристроїв, забезпечуючи логіку «монтажне або». Генератор струму 5 мкА забезпечує на вході 1- провідного пристрою низький логічний рівень, коли шина не підключена. Оскільки лінія тактового сигналу відсутня, обмін є асинхронним. Це означає, що в процесі обміну потрібно досить точно витримувати необхідні часові співвідношення.

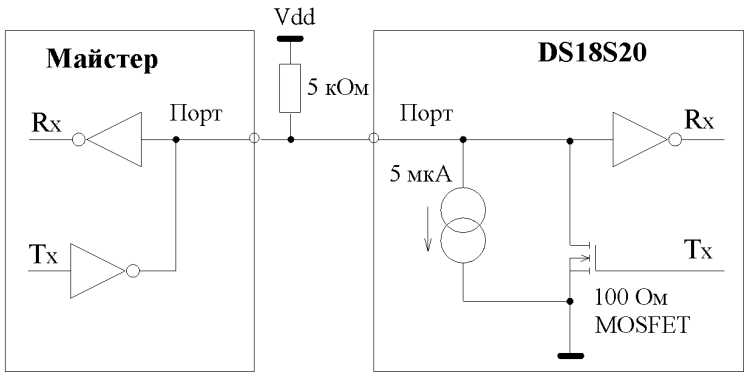


Рис. 4.27.

1-провідна шина оперує з TTL-рівнями, тобто логічна одиниця представлена рівнем напруги близько 5 В, а логічний нуль – напругою 0 В. У початковому стані на лінії присутній рівень логічної одиниці, який забезпечується підтягуючим резистором номіналом 5 кОм.

Ведучий пристрій посилає імпульс скидання (reset pulse) – сигнал низького рівня тривалістю не менше 480 мкс. За імпульсом скидання відбувається відповідь підлеглого пристрою (presence pulse) – сигнал низького рівня тривалістю 60 – 240 мкс, який генерується через 15 – 60 мкс після завершення імпульсу скидання.

Відповідь підлеглого пристрою дає ведучому пристрою зрозуміти, що на шині присутній термометр, і він готовий до обміну. Після того, як ведучий виявив відповідь, він може передати термометру одну з команд. Передача ведеться шляхом формування спеціальних часових інтервалів (time slots). Кожен часовий інтервал служить для передачі одного біта. Першим передається молодший біт. Інтервал починається імпульсом низького рівня, тривалість якого лежить в межах 1 – 15 мкс. Оскільки перехід з одиниці в нуль менш чутливий до ємності шини (він формується відкритим транзистором, тоді як перехід з нуля в одиницю формується підтягуючим резистором), саме цей перехід використовують 1-провідні пристрої для синхронізації з майстром. У підлеглому пристрої запускається схема часової затримки, яка визначає момент зчитування даних. Номінальне значення затримки дорівнює 30 мкс, проте, воно може коливатися в межах 15 – 60 мкс. За імпульсом низького рівня передається біт даних. Він повинен утримуватися ведучим на шині протягом 60 – 120 мкс від початку інтервалу. Часовий інтервал завершується переведенням шини в стан високого рівня на час не менше 1 мкс. Слід відзначити, що обмеження на цей час зверху не накладається. Аналогічним чином формуються часові інтервали для всіх бітів, що передаються.

Першою командою, яку повинен передати майстер для DS18B20 після ініціалізації, є одна з команд функцій ПЗП. Всього DS18B20 має 5 команд функцій ПЗП:

– **Read ROM [33h].** Ця команда дозволяє прочитати вміст ПЗП. У відповідь на цю команду DS18B20 передає 8-бітний код сімейства (10h), потім 48-бітний серійний номер, а потім 8-бітну CRC для перевірки правильності прийнятої інформації.

– **Match ROM [55h].** Ця команда дозволяє адресувати на шині конкретний термометр. Після цієї команди ведучий пристрій повинен передати потрібний 64-бітний код, і лише той термометр, який має такий код, відповість на наступний імпульс скидання.

– **Skip ROM [CCh].** Ця команда дозволяє пропустити процедуру порівняння серійного номера, тим самим заощадити час в системах, де на шині є всього один пристрій.

– **Search ROM [F0h].** Ця досить складна у використанні команда, дозволяє визначити серійні номери всіх термометрів, що присутні на шині.

– **Alarm Search [ECh].** Ця команда аналогічна попередній, але відповідати будуть лише ті термометри, в яких результат останнього виміру температури виходить за попередньо встановлені межі TH і TL.

Оскільки в прикладі на шині всього один пристрій, найбільш підходячою функцією є функція Skip ROM. Окрім неї ще може бути корисною функція Read ROM, яка дозволяє ідентифікувати підключений на шину пристрій за його кодом сімейства і серійним номером.

Після обробки однієї з команд функцій ПЗП, DS18B20 здатний сприймати ще декілька команд:

– **Write Scratchpad [4Eh]**. Ця команда дозволяє записати дані в проміжний ОЗП DS18B20.

– **Read Scratchpad [Beh].** Ця команда дозволяє зчитувати дані з проміжного ОЗП.

– **Copy Scratchpad [48h].** Ця команда копіює байти TH і TL з проміжного ОЗП в енергонезалежну пам'ять. Ця операція потребує біля 10 мс.

– **Convert T [44h]**. Ця команда запускає процес перетворення температури.

– **Recall E2 [B8h]**. Ця команда діє звототнім чином по відношенню до команди Copy Scratchpad, тобто вона дозволяє зчитати байти TH и TL з енергонезалежної пам’яті в проміжний ОЗП. При включенні живлення ця команда виконується автоматично.

– **Read Power Supply [B4h].** Ця команда дозволяє перевірити, чи використовує DS18B20 паразитне живлення. Справа в тому, що DS18B20 можна підключати всього за допомогою двох провідників, в такому випадку для живлення використовується лінія даних.

При використанні DS18B20 тільки для вимірювання температури потрібні всього дві з цих команд: Convert T і Read Scratchpad. Послідовність дій при вимірюванні температури повинна бути наступною:

– Надсилаємо імпульс скидання і приймаємо відповідь термометра.

- Надсилаємо команду Skip ROM [CCh].
- Надсилаємо команду Convert T [44h].
- Формуємо затримку мінімум 750 мс.
- Надсилаємо імпульс скидання і приймаємо відповідь термометра.
- Надсилаємо команду Skip ROM [CCh].
- Надсилаємо команду Read Scratchpad [BEh].
- Зчитуємо дані з проміжного ОЗП (8 байт) та CRC.
- Перевіряємо CRC, і якщо дані зчитані вірно, обчислюємо температуру.

На рис. 4.28. представлена схема підключення датчика температуры DS18B20 до мікроконтролера AT90S8535.

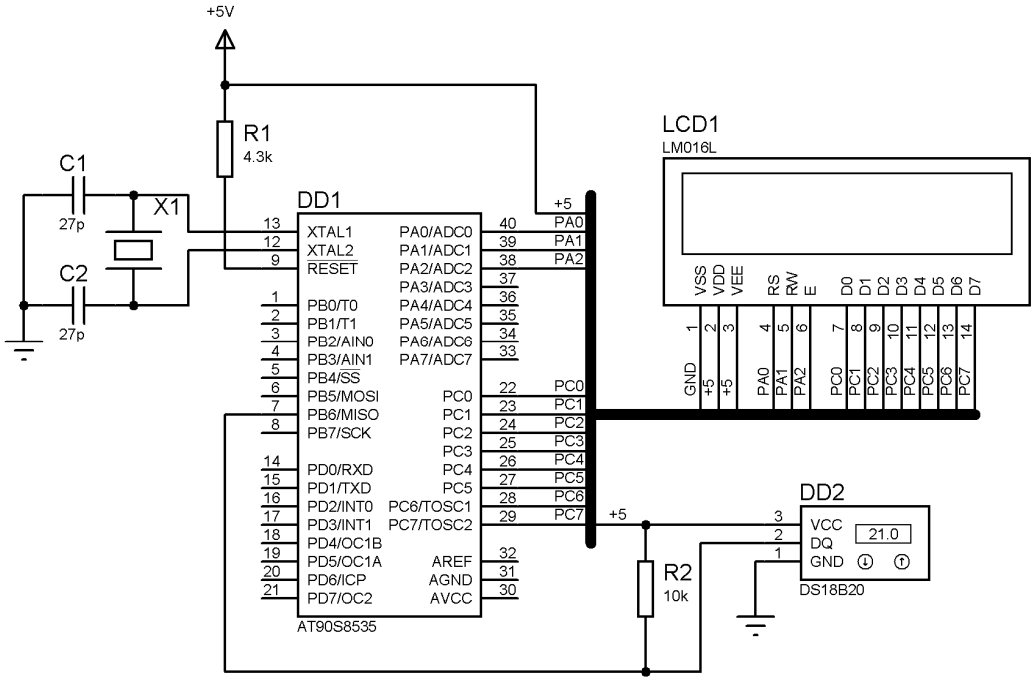


Рис. 4.28.

Запускаємо середовище розробки CodeVision і створюємо новий проект, обравши з меню File команду New. У вікні, що з'явилося (рис. 4.29) слід відмітити позицію Project.

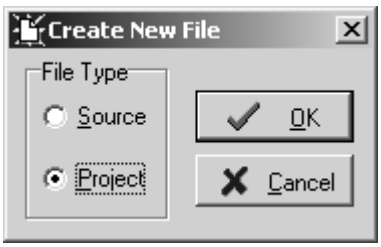


Рис .4.29.

У вікні, з пропозицією використання майстра проекту (CodeWizard), що з'явилося, натиснути на кнопку «NO». Далі потрібно ввести назву проекту (наприклад TestWire). У вікні налаштувань проекту, що з'явилося встановити значення, як показано на рис. 4.30.

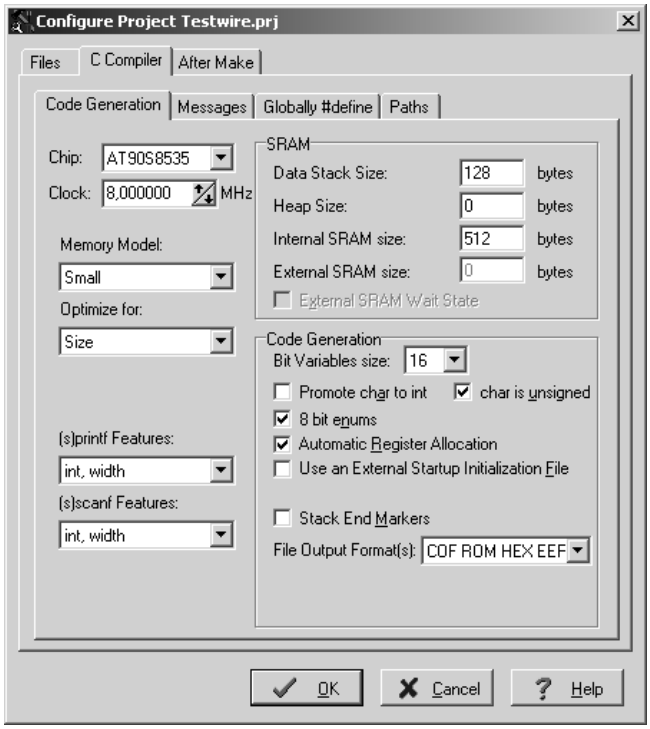


Рис. 4.30.

Додаємо в проект файл с вихідним кодом. Для цього в меню File потрібно обрати команду New. Програма для мікроконтролера AT90S8535, створена в компіляторі CodeVision, представлена нижче.

```
#asm                                  // вказання виводів, до яких підключається датчик
   .equ __w1_port=0x18
   .equ __w1_bit=6                          // лінія даних на виводі 6 порту B
#endasm
#include "1wire.h"                           // підключення заголовочних файлів
#include "90s8535.h"
#include "delay.h"
#include "math.h"
#include "string.h"

 unsigned char delay;                        // затримка опитування датчика

 #define RS() PORTA|=0b00000001              // встановити сигнал команда/дані
 #define NRS() PORTA&=~0b00000001            // скинути сигнал команда/дані
 #define RW() PORTA|=0b00000010              // встановити сигнал зчитування/запис
 #define NRW() PORTA&=~0b00000010            // скинути сигнал зчитування/запис
 #define E() PORTA|=0b00000100               // встановити сигнал стробу
 #define NE() PORTA&=~0b00000100             // скинути сигнал стробу
```

```
// объявление прототипов функций
void main(void);                        // основна функція програми
void temp(void);                        // опитування датчика температури
void set_config();                      // змінює байт конфігурації датчика
void init_lcd();
void write();
void init_cpu(void);                    // ініціалізація мікроконтролера
// функція опитування датчика температури
void temp(void)
{
        unsigned char buf[9];           // буфер зчитування даних
        unsigned char i;                // додаткові змінні
        int t=0;
        char h,l;
        w1_init();                      // виклик функції ініціалізації датчика
        w1_write(0xCC);                 // пропуск ПЗП
        w1_write(0x44);                 // перетворення температури
        delay_ms(delay);                // очікування кінця перетворення температури
        w1_init();                      // ініціалізація
        w1_write(0xCC);                 // пропуск ПЗП
        w1_write(0xBE);                 // зчитування блокнотної пам'яті
        for (i=0; i<9; i++)             // зчитати дев'ять байт
        {
                buf[i] = w1_read();     // запис зчитаного байта в буфер
        }
        if((buf[1]&0x80)!=0)
        {
         buf[1]=~buf[1];
         buf[0]=~buf[0]+1;
         RS();
         PORTC='-';
         write();
        }
        else
        {
         RS();
         PORTC='+';
         write();
        }
        t=buf[1];
        t=t<<8;
        t=t+buf[0];
        t=t>>4;
        h=t/10;
        l=t%10;
        PORTC=h+48;
        write();
        PORTC=l+48;
        write();
        PORTC='C';
        write();
```

```
}
// функція установки байта конфігурації
void set_config()
{
        w1_init();                  // повторювати поки датчик не відповість
        w1_write(0xCC);             // пропуск ПЗП
        w1_write(0x4E);             // запис в блокнотну пам'ять

        w1_write(0x00);             // запис Твис
        w1_write(0x00);             // запис Тниз
        w1_write(0b00011111);       // запис байта конфігурації
}
// функция ініціалізації мікроконтролера
 void init_lcd()                    // функція ініціалізації РКІ
 {
 DDRC=0xFF;                         // налаштування порту на виведення даних
 PORTC=0x38;                        // виведення команди 0x38 в порт контролера
 write();                           // запис команди в модуль
 PORTC=0x0C;                        // виведення команди 0x0C в порт контролера
 write();                           // запис команди в модуль
 PORTC=0x06;                        // виведення команди 0x06 в порт контролера
 write();                           // запис команди в модуль
 PORTC=0x02;                        // виведення команди 0x02 в порт контролера
 write();                           // запис команди в модуль
 PORTC=0x01;                        // виведення значення 0 на порт контролера
 write();
 }

 void write()                       // функція, яка формує тактовий сигнал
 {
  E();                              // встановлення тактового сигналу
 delay_us(200);                     // часова затримка
 NE();                              // скидання тактового сигналу
 delay_us(200);                     // часова затримка
 }

void init_cpu(void)
{
      DDRA=0x07;
      DDRC=0xFF;
      DDRB = 0b00010100;            // підключення DS18B20: 6 – DQ, 4 – UCC
      PORTB = 0;                    // вивести одиницю на PORTB.4
      DDRD=0xFF;
      NRS();
      NRW();
      NE();
      delay=100;                    // затримка
}

void main(void)                     // головна функція
{
```

```
    char text1[]="Temperature:";
    char index=0;
    init_cpu();                          // виклик функції ініціалізації мікроконтролера
    init_lcd();
      set_config();
      while(1)                           // вічний цикл
      {
       delay_ms(1000);
       NRS();
       PORTC=0b00000001;
       write();
       RS();   // встановлення в 1 сигналу команда/дані; в РКІ передаються дані
       delay_ms(10);
      for(index=0; index<strlen(text1); index++) // в циклі виконується запис даних
      {
        PORTC=text1[index];        // установка ASCII-коду символу на порті
        write();                   // запис символу в РКІ; курсор зміщується автоматично
       }
      temp();                      // виклик функції перетворення температури
      }
}
```

Зберігши файл на диск з ім'ям Testwire.c, проект можна компілювати. Для цього в меню Project вибрати команду Compile, або натиснути клавішу F9. У каталозі, де розташовані файли проекту, буде створений файл для прошивки мікроконтролера Testwire.hex. Після запуску схеми на моделювання в Proteus, на екрані РКІ з'явиться значення температури (рис. 4.31).

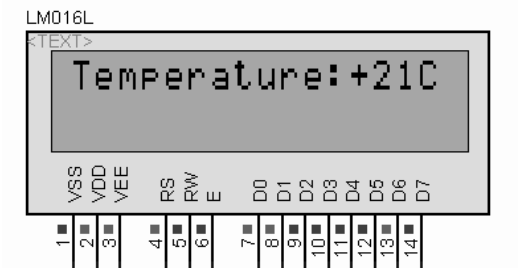


Рис. 4.31.

4.4.4. Програмування і управління багаторядковими РКІ

Модуль MTC-S16400X (4 рядки по 16 символів) за допомогою 16-контактного роз'єму (табл .4.6) обмінюється інформацією з управляючим мікроконтролером (у нашому випадку з AVR). AVR-мікроконтролер посилає в РКІ команди (табл. 4.4), управляючи режимами його роботи, і ASCII-коди символів, що виводяться. У свою чергу, РКІ може посилати мікроконтролеру за його запитом інформацію про свій стан, і дані зі своїх внутрішніх блоків пам'яті.

Табл. 4.6.

№	Назва виводу	Опис
1	V_{SS}	(–) Живленя, 0 В.
2	V_{DD}	(+) Живленя, +5 В.
3	V_0	Напруга зсуву, яка управляє контрастністю
4	RS	Вхід. Високий рівень – дані; низький – команди
5	R/W	Вхід. Високий рівень – зчитування, низький рівень – запис
6	E	Вхід. Строб, який супроводжує сигнали на шині "команда/дані"
7	DB_0	Шина "команда/дані"
8	DB_1	
9	DB_2	
10	DB_3	
11	DB_4	
12	DB_5	
13	DB_6	
14	DB_7	
15	BKL1	Живленя для підсвічування. 100 В/400 Гц для електролюмінесцентної, або 4,2 В/220÷330 мА для світлодіодного підсвічування. Модуль може працювати без підсвічування.
16	BKL2	

Три виводи 14-контактного роз'єму призначено для подачі напруги живлення і напруги зсуву, яка управляє контрастністю дисплея. З 11 виводів, які залишилися, 8 (DB0 – DB7) використовуються для організації мультиплексованої шини "команда/дані", і на 3 виводи (RS, R/W, E) мікроконтролер формує управляючі сигнали. За допомогою сигналу на лінії RS мікропроцесор повідомляє контролер індикатора про те, що саме передається по шині: команда або дані. Сигнал на лінії E є стробом, який супроводжує сигнали на шині "команда/дані". Запис інформації в РКІ відбувається по зрізу цього сигналу. Потенціал на управляючому виводі R/W задає напрям передачі даних: запис в RAM індикатора (R/W =0) або зчитування (R/W=1). На рис. 4.32 представлена схема з’єднання багаторядкового алфавітно-цифрового РКІ з мікроконтролером.

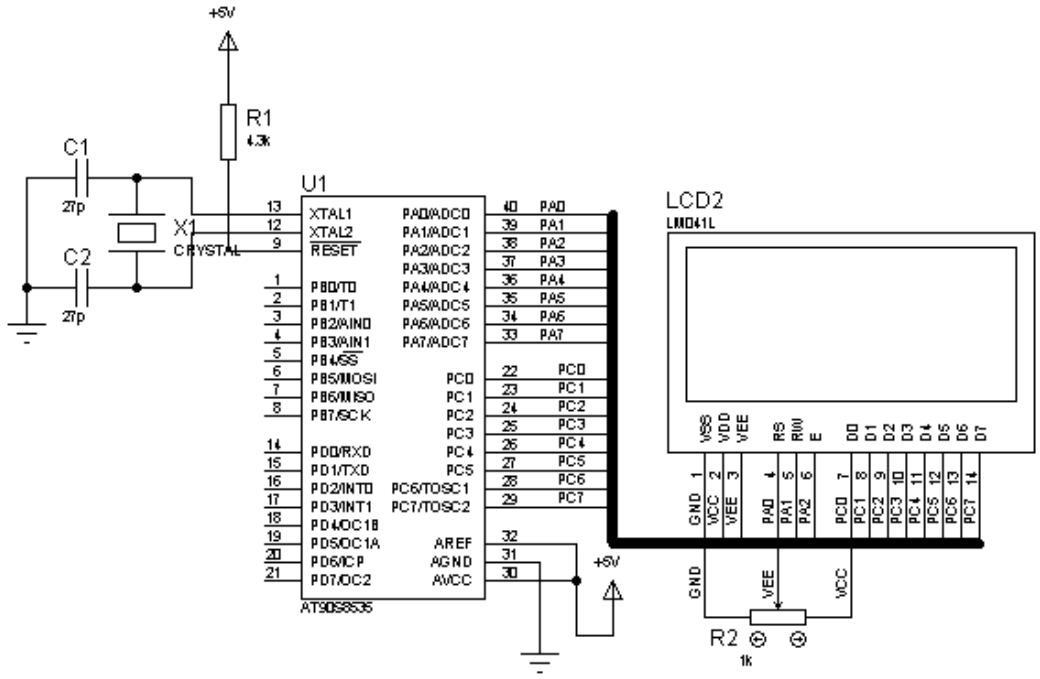


Рис. 4.32.

Для випадку, коли мікроконтролер має обмежену кількість ліній вводу/виводу, передбачений другий варіант підключення РКІ з використанням 4-х розрядної шини "команда/дані". При цьому кожен байт даних передається по лініях DB4 – DB7 послідовно двома тетрадами, починаючи зі старшої. Контролер РКІ, після прийому байта команди або байта даних, потребує деякого часу (близько 50 мкс) для обробки отриманої інформації, впродовж якого мікроконтролер не повинен виконувати нових передач. Команда очищення дисплея (Clear Display) виконується приблизно за 1,6 мс.

Система команд цього модуля не відрізняється від системи 2-рядкового 16-символьного алфавітно-цифрового модуля. Це пояснюється тим, що в обох модулях використаний контролер HD44780 (або сумісний).

В табл. 4.7 приводиться структура адресного простору модуля.

Табл. 4.7.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F

Модуль має 4 строчки по 16 символів в кожній. Як видно з таблиці, після останньої адреси комірки в першому рядку (0x0F), наступну адреса має перша комірка в третьому рядку (0x10). Це означає, що при виведенні

даних на індикатор, перші 16 символів відображатимуться у верхньому рядку, наступні 16 символів відображатимуться на 3 рядку. Остання комірка в третьому рядку має адресу 0x1F, а перша комірка в 2 рядку має адресу 0x40. Таким чином, є розрив адресного простору. Символи, починаючи з 33, не відображатимуться в 2 строчці. Для цього потрібно виконати команду „Установка адреси DDRAM", де в позиціях адреси вказати потрібне значення (табл. 4.2). Така організація пам'яті створює деякі незручності при програмуванні виводу інформації.

Окрім DDRAM, контролер РКІ містить ще один блок пам'яті – знакогенератор. Основна його частина є ПЗП (CGROM), в якій знаходяться зображення символів, і її не можна змінити. Друга частина, в якій задаються зображення символів для 16-ти перших кодів таблиці знакогенератора, є ОЗП (CGRAM), який можна перепрограмувати. Є можливість задати зображення 8 символів, які відповідають кодам 0(8), 1(9), 2(10)...7(15). Для кожного з восьми перепрограмованих символів в CGRAM відводиться по 8 комірок пам'яті, кожна з яких відповідає одному рядку точок в зображенні символу. Таким чином, перепрограмована частина знакогенератора містить 64 байти пам'яті (8×8).
Принцип кодування CGRAM демонструється в табл. 4.8.

Табл. 4.8.

7	6	5	4	3	2	1	0
ст.				мол.			
*	*	*	0	1	1	0	0
			1	0	0	1	0
			0	0	1	0	0
			0	1	0	0	0
			1	1	1	1	0
			0	0	0	0	0
			0	0	0	0	0
			0	0	0	0	0
*	*	*	1	1	1	1	1
			1	0	0	0	1
			1	0	1	0	1
			1	0	1	1	1
			1	0	1	0	1
			1	0	0	0	1
			1	1	1	1	1
			0	0	0	0	0

Розглянемо послідовність команд, за допомогою яких виконується ініціалізація рідкокристалічного індикатора: 0x38, 0x0C, 0x06, 0x01. Ці команди виконують наступні функції (табл. 4.2):

$0x38_{16}=00111000_2$
DL=1 – розрядність шини даних 8 біт
N=1 – кількість рядків дисплея рівна 2
F=0 – шрифт 5×7 точок
$0x0C_{16}=00001100_2$
D=1 – включити дисплей
C=0 – погасити курсор
B=0 – зображення курсора не робити мігаючим
$0x06_{16}=00000110_2$
I/D=1 – встановити напрям зсуву курсора праворуч при записі даних
S=0 – заборонити зсув дисплея разом з курсором
$0x02_{16}=00000010_2$ – встановити курсор в нульову позицію (адреса 0).

Приклад 4.8. Написати програму, яка виводить на екран РКІ текстову інформацію. Реалізувати функцію, яка виводить текст у вказаному рядку, починаючи з вказаної позиції.

```
#include <io8535v.h>          // підключення заголовочних файлів
#include <macros.h>           // бібліотека макросів
#include <string.h>           // бібліотека функцій для роботи з рядками

#define RS() PORTA|=0b00000001          // встановити сигнал команда/дані
#define NRS() PORTA&=~0b00000001        // скинути сигнал команда/дані
#define RW() PORTA|=0b00000010          // встановити сигнал зчитування/запис
#define NRW() PORTA&=~0b00000010        // скинути сигнал зчитування/запис
#define E() PORTA|=0b00000100           // встановити тактовий сигнал
#define NE() PORTA&=~0b00000100         // скинути тактовий сигнал
#define LCD PORTC                 // РКІ модуль підключений до порту С контролера
#define DDLCD DDRC                // даємо назву регістра напрямку даних
// оголошення прототипів функцій
void init_cpu();              // функція ініціалізації контролера
void pause();                 // часова затримка
void init_lcd();              // функція ініціалізації РКІ
void write();                 // функція, при виклику якої формується тактовий сигнал
void out_text(char* text, char row, char pos); // функція виведення текстового рядка
// реализація функцій
void init_cpu()               // функція ініціалізації контролера
{
DDRA=0b00000111;              // налаштування трьох старших біт порту А на вихід
DDLCD=0x00;                   // порт переводиться в стан високого імпеданса
NRS();                        // скидання сигналу команда/дані
NRW();                        // скидання сигналу зчитування/запис
NE();                         // скидання тактового сигналу
}
void pause()                  // функція часової затримки
 {
 int tmp;                     // часова змінна
```

```
 for(tmp=0;tmp<1000;tmp=tmp+1) NOP(); // у циклі контролер виконує пусту операцію
 }
 void init_lcd()              // функція ініціалізації РКІ
 {
 DDLCD=0xFF;                  // налаштування порту на виведення даних
 LCD=0x38;                    // виведення команди 0x38 в порт контролера
 write();                     // запис команди в модуль
 LCD=0x0C;                    // виведення команди 0x0C в порт контролера
 write();                     // запис команди в модуль
 LCD=0x06;                    // виведення команди 0x06 в порт контролера
 write();                     // запис команди в модуль
 LCD=0x02;                    // виведення команди 0x02 ва порт контролера
 write();                     // запис команди в модуль
 LCD=0x00;                    // виведення значення 0 в порт контролера
 DDLCD=0x00;                  // переведення порту в стан високого імпеданса
 }
 void write()                 // функція, яка видає тактовий сигнал
 {
  E();                        // установка тактового сигналу
  pause();                    // часова затримка
  NE();                       // скидання тактового сигналу
  pause();                    // часова затримка
 }
 void out_text(char* text, char row, char pos)      // функція виведення текстового рядка
на екран РКІ
 {
   char addr, index;          // додаткові змінні
   DDLCD=0xFF;                // налаштування порту, до якого підключений РКІ, на вихід
   switch(row)                // конструкція вибору
   {
         case 0:              // якщо row=0 (перший рядок)
         addr=pos;            // установка адреси з урахуванням початкової позиції (0)
         break;               // вихід з конструкції
         case 1:              // якщо row=1 (другий рядок)
         addr=0x40+pos;       // установка адреси початку другого рядка 0x40
         break;               // вихід з конструкції
         case 2:              // якщо row=2 (третій рядок)
         addr=0x10+pos;       // установка адреси з урахуванням початкової позиції (0x10)
         break;               // вихід з конструкції
         case 3:              // якщо row=3 (четвертий рядок)
         addr=0x50+pos;       // установка адреси з урахуванням початкової позиції (0x50)
         break;               // вихід з конструкції
         default:             // якщо жодна з альтернатив не підходить
         addr=0;              // установка адреси в 0
         break;               // вихід з конструкції
   }
   NRS();                     // скидання сигналу RS – передаються команди
   addr|=0x80;                // формування команди адреси
   LCD=addr;                  // установка команди на шині даних
   write();                   // запис команди в РКІ
   RS();                      // установка сигналу RS – передаються дані
```

```
  for(index=0; index<strlen(text);index++)  // у циклі відбувається передача даних в РКІ
  {
        LCD=(text[index]);   // установка ASCII-коду символа на шину даних
        write();             // запис даних в модуль
  }
  NRS();                     // скидання сигналу RS
  DDLCD=0x00;                // переведення порту, до якого підключений РКІ,
                             // у високоімпедансний стан
}
 void main()                 // основна програма
{
char* text1="First row";     // оголошенння змінних
char* text2="Second row";    // інформація, яка буде виводитися в другому рядку РКІ
char* text3="Third row";     // інформація, яка буде виводитися в третьому рядку РКІ
char* text4="Fourth row";    // інформація, яка буде виводитися в четвертому рядку РКІ
init_cpu();                  // виклик функції ініціалізації контролера
init_lcd();                  // виклик функції ініціалізації РК-модуля
out_text(text1,0,0);         // виведення тексту в першому рядку з позиції 0
out_text(text2,1,1);         // виведення тексту в другому рядку з позиції 1
out_text(text3,2,2);         // виведення тексту в третьому рядку з позиції 2
out_text(text4,3,3);         // виведення тексту в четвертому рядку з позиції 3
DDLCD=0;                     // переведення порту в високоімпедансний стан
while(1);                    // зациклення контролера
}
```

Після виконання програми на екрані РКІ з’явиться зображення, як показано на рис. 4.33.

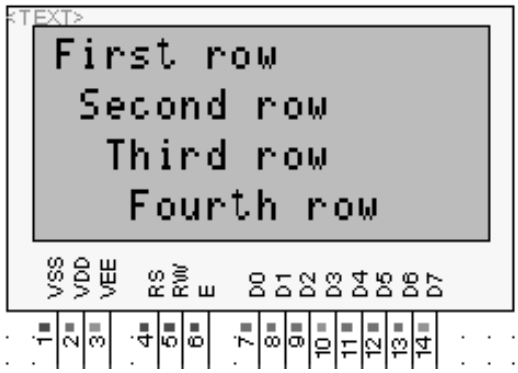


Рис. 4.33.

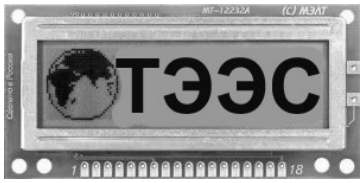
4.5. Графічні РКІ

Рідкокристалічний модуль MTG-S12864X складається з контролера управління та рідкокристалічної панелі. Кожній точці, яка світиться на РКІ, відповідає „1” в ОЗП модуля. Зовнішній вигляд модуля MTG-S12864X представлений на рис. 4.34.

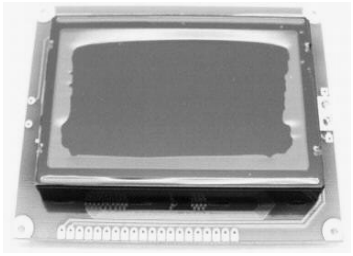


Рис.4.34.

Також випускається широкий набір графічних рідкокристалічних модулів, які відрізняються конструктивним оформленням, розмірами, розрішаючою здатністю (рис. 4.35).



MT-12232A



WG12864A



YMS160160-01



YMS240128-01

Рис. 4.35.

4.5.1. Сполучення РКІ з мікроконтролером

Модуль дозволяє приймати команди по шині DB7-DB0, записувати дані в ОЗП по 8-розрядній шині даних (DB7-DB0). Для початкової ініціалізації модуля потрібно подати сигнал RES. При цьому модуль виконує команди установки Display Start Line в 0, Display Off. Після деактивації сигналу RES необхідно дочекатися скидання біт BUSY і RESET в регістрі статусу. Після цього модуль нормально функціонує. Опис виводів модуля MTG-S12864BFGHSGY представлений в табл. 4.9.

Табл. 4.9.

№ виводу	Назва виводу	Напрям даних	Опис
1	V_{dd}	–	Напруга живлення
2	V_{ss}	–	Земля
3	V_{ee}	–	Управління контрастністю
4-11	DB0-DB7	I/O	Шина даних з 3 станами
12	CS1	I	Вибір контролера лівої половинки дисплея. Активний рівень – низький
13	CS2	I	Вибір контролера правої половинки дисплея. Активний рівень – низький
14	RES	I	Встановлення сигналу в низький рівень ініціалізує графічний модуль
15	R/W	I	R/W=1 – дані можуть бути зчитані з модуля в контролер; R/W=0 – дані можуть бути записані в модуль
16	D/I	I	D/I=1 – дані на DB7-DB0 – дані для відображення D/I=0 – дані на DB7-DB0 – команда
17	E	I	Сигнал тактування. Якщо R/W=0, дані записуються по зрізу сигналу E. Якщо R/W=1, дані зчитуються, коли E приймає високий рівень
18	FGND	–	Земля
19	BKL_A	–	Живлення для підсвічування
20	BKL_K	–	

На рис. 4.36 представлена типова схема з’єднання графічного РКІ MTG-12864 з мікроконтролером AT90S8535.

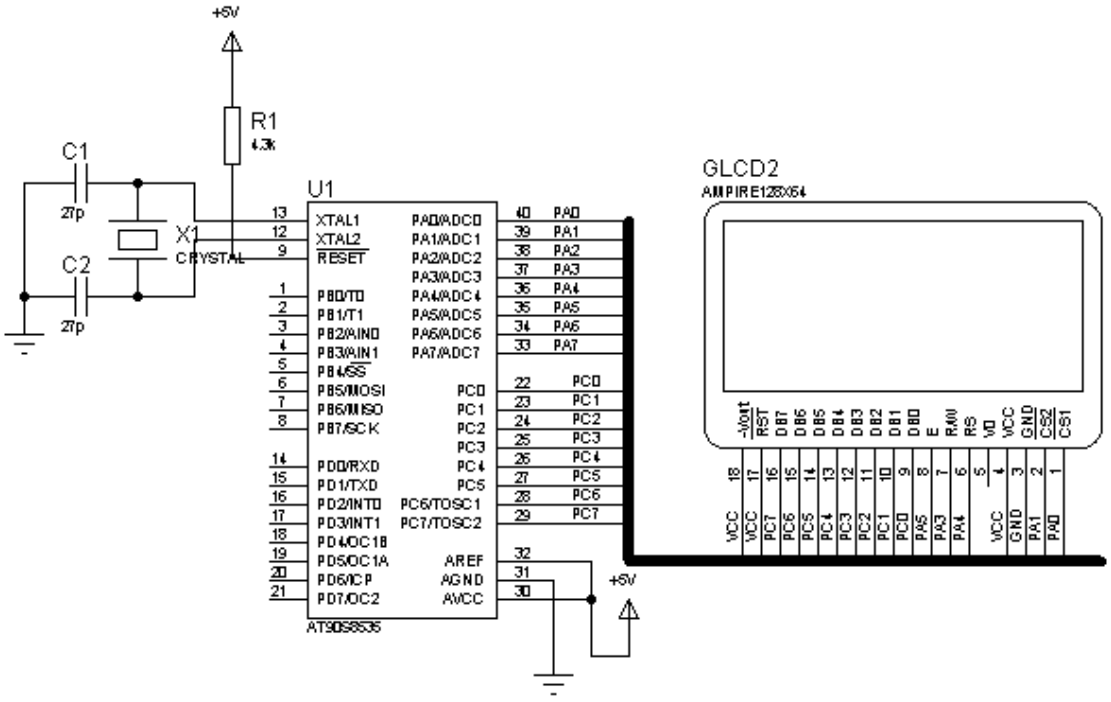


Рис. 4.36.

4.5.2. Структура адресного простору модуля MTG-12864

Модуль містить ОЗП для зберігання даних, які виводяться на РКІ, розміром 64×64×2 біт (по 64×64 біт на кристал). Для вибору потрібного кристалу використовуються сигнали CS1 і CS2. Далі, вся інформація відноситься до кожного кристалу. ОЗП розбитий на 8 сторінок розмірами 64×8 біт кожна. Кожній точці, яка світиться на РКІ, відповідає лог.1 у комірці ОЗП модуля. Відповідність між адресами в ОЗП і точками, які відображуються на РКІ, представлено на рис. 4.37.

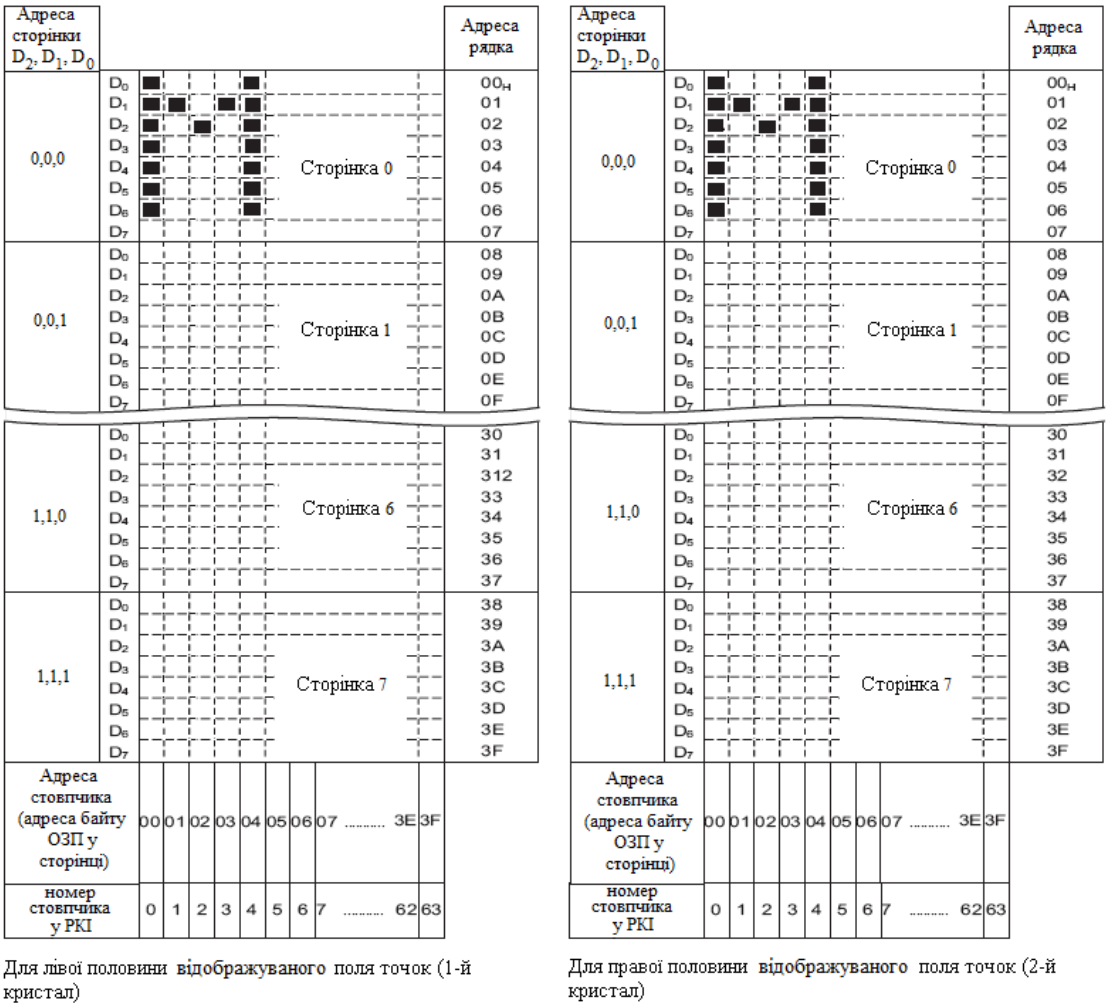


Рис. 4.37.

4.5.3. Система команд модуля MTG-12864

Зчитування (запис) даних із (в) модуль виконується по сторінках (64×8 біт або 64×1 байт). Кожна сторінка представлена як 64 байти. Для зчитування або запису даних за заданою адресою необхідно заздалегідь встановити сторінку ОЗП і вибрати стовбчик в межах сторінки ОЗП. Це виконується командами „Set Page” і „Set Address” відповідно (табл. 4.10). Після цього можна записати, або прочитати байт даних. Модуль підтримує безперервну послідовність операцій зчитування і запису: після зчитування (запису) одного байта лічильник стовбчика автоматично збільшується на 1, і модуль готовий до нової операції зчитування (запису) за наступною адресою без попередньої установки сторінки ОЗП і адреси стовпчика. Лічильник стовпчика рахує в межах однієї сторінки. При досягненні лічильником адреси 63, наступним буде значення 0 і т. д.

Табл. 4.10.

Команда	RD/WR	A0	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Функція
Display ON/OFF	0	0	0	0	1	1	1	1	1	0/1	Вмикає або вимикає РКІ, незалежно від даних у екранному ОЗП й внутрішнього стану "1"- ввімкнути дисплей "0"- вимкнути дисплей
Display START Line	0	0	1	1	Display START Line (0...63)						Визначає строку ОЗП, яка буде відображатися у верхній строчці РКІ (Стартова строка РКІ)
Set Page	0	0	1	0	1	1	1	Page (0...7)			Встановлює сторінку ОЗП (стр. 0...7)
Set Address	0	0	0	1	Column address (0...63)						Визначає адресу стовпця
Status Read	1	0	BUSY	0	ON/OFF	RESET	0	0	0	0	Зчитування режиму стану BUSY 1 Модуль зайнятий внутрішньою обробкою 0 Модуль готовий до роботи з зовнішнім МП ON/OFF 1 РКІ ввімкнено 0 РКІ вимкнено RESET 1 Стан скидання 0 Нормальний стан
Write Display Data	0	1	Write Data								Запис данних до ОЗП модуля
Read Display Data	1	1	Read Data								Зчитування данних з ОЗП модуля

Ці команди вибирають ОЗП за допомогою раніше заданої адреси, після чого адреса стовпця інкремінується

Модуль підтримує команду "Display START Line", яка встановлює номер верхнього рядка, що відображається. Номер може бути в інтервалі від 0 до 63, що відповідає інтервалу від першого рядка нульової сторінки ОЗП до останнього рядка сьомої сторінки ОЗП. Відразу за сьомою сторінкою відображується знову нульова сторінка ОЗП. Це дозволяє реалізувати плавний зсув інформації на РКІ по вертікалі шляхом зміни першого рядка, що відображається.

4.5.4. Формування текстової інформації

На відміну від алфавітно-цифрових рідкокристалічних модулів, в яких для виведення тексту достатньо було записати ASCII-коди символів в пам'ять, в графічних модулях така можливість відсутня. Тому завдання по відображенню інформації (як текстової, так і графічної) вирішується програмним шляхом. Кожна сторінка пам'яті має 8 біт (точок) у висоту. При виведенні текстової інформації необхідно витримати проміжки між рядками для зручності сприйняття тексту (хоч і не обов'язково). У алфавітно-цифрових РКІ найчастіше розміри символу складають 5×7 точок. Розмір знакомісця дорівнює 6×8 точок. Нижній рядок і стовпчик праворуч завжди погашені. Це зроблено для формування відстані між символами. Для відображення одного знакомісця вказаних розмірів, необхідно записати в ОЗП модуля 6 байт даних, заздалегідь вказавши адресу сторінки і стовпця, де потрібно вивести символ (рис. 4.38). Всі дані,

які утворюють шрифт, зручно оформити у вигляді двовимірного масиву, наприклад font[256][6]. У цьому масиві зберігатиметься графічна інформація про 256 символів ASCII, кожен з яких представлений 6 байтами. Управління виведенням текстової інформації значно полегшується, якщо індекс елементу масиву збігається з ASCII-кодом символу. У такому разі досить звернутися до елементу масиву за кодом символу з таблиці ASCII, і розробник отримує доступ до 6 байт даних, які власне і сформують графічне зображення при виведенні їх на індикатор.

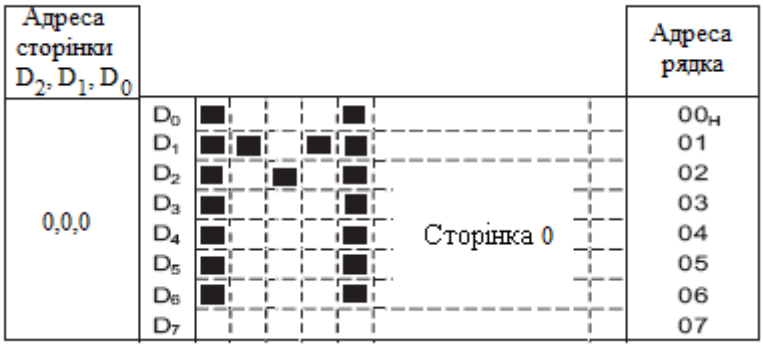


Рис. 4.38.

4.5.5. Формування графічної інформації

Окрім тексту можна виводити на екран графічного РКІ різні зображення. Принцип формування зображення залишається таким самим, як і при виведенні текстової інформації. Необхідно сформувати масив даних в пам'яті контролера (Flash, EEPROM), а потім в циклі записати ці дані в індикатор. Звичайно, потрібно стежити за значенням стовпця, в який виводяться дані у поточний момент, і досягши значення 63, змінювати номер сторінки пам'яті. Крім того, можливе програмне формування графіки.

Приклад 4.9. Написати програму, яка виводить в правій частині екрану фотографію, а в лівій – текстову інформацію. Принципова схема пристрою представлена на рис. 4.36.

Текст програми представлено нижче.

```
#include <io8535v.h>              // підключення заголовочних файлів
#include <macros.h>               // підключення бібліотеки макросів
#include <string.h>               // бібліотека функцій для роботи з рядками
#define CS1() PORTA|=0x01         // установка сигналу управління
                                  // лівою половиною індикатора
#define NCS1() PORTA&=~0x01       // скидання сигналу управління
                                  // лівою половиною індикатора
#define CS2() PORTA|=0x02         // установка сигналу управління
                                  // правою половиною індикатора
#define NCS2() PORTA&=~0x02       // скидання сигналу управління
                                  // правою половиною індикатора
```

```
#define RES() PORTA|=0x04          // установка сигналу скидання RES
#define NRES() PORTA&=~0x04        // скидання сигналу RES
#define RW() PORTA|=0x08           // установка сигналу RW – зчитування/запис
#define NRW() PORTA&=~0x08         // скидання сигналу RW – зчитування/запис
#define DI() PORTA|=0x10           // установка сигналу DI – команда/дані
#define NDI() PORTA&=~0x10         // скидання сигналу DI – команда/дані
#define RS() DI()                  // установка сигналу RS – команда/дані
#define NRS() NDI()                // скидання сигналу RS – команда/дані
#define E() PORTA|=0x20            // установка сигналу дозволу запису в РКІ
#define NE() PORTA&=~0x20          // скидання сигналу дозволу запису в РКІ
// в двомірному масиві зберігається інформація про шрифт
const char currfont[91][6]={{0,0,0,0,0,0},{0,6,95,6,0,0},{0,0,0,0,0,
0},{0,0,0,0,0,0},{36,74,255,82,36,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{
0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,160,96,0,0},{0,0,0,0,0,0},{0,0,96,96,0,0},{0,0,0,
0,0,0},{62,65,65,65,62},{0,4,2,127,0,0},{66,97,81,73,70,0},{34,65,73,73,54,0},{56,36,34,12
7,64,0},{47,73,73,73,49,0},{62,73,73,73,50,0},{65,33,17,9,7,0},{54,73,73,73,54,0},{38,73,7
3,73,62,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{
0,0,0,0,0,0},{124,18,17,18,124,0},{127,73,73,73,54,0},{62,65,65,65,34,0},{127,65,65,65,28,
0},{127,73,73,73,73,0},{127,9,9,9,1,0},{62,65,65,81,50,0},{127,8,8,8,127,0},{129,129,127,
129,129,0},{48,64,64,64,63,0},{127,8,20,34,65,0},{127,64,64,64,64,0},{127,2,4,2,127,0},{1
27,4,8,16,127,0},{62,65,65,65,62,0},{127,9,9,9,6,0},{62,65,73,81,62,0},{127,9,25,41,70,0},
{38,73,73,73,50,0},{1,1,127,1,1,0},{63,64,64,64,63,0},{31,32,64,32,31,0},{127,64,48,64,12
7,0},{99,20,8,20,99,0},{3,4,120,4,3,0},{105,89,73,77,75,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0
,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{56,68,68,36,120,0},{127,72,72,72,48,0},{56
,68,68,68,40,0},{56,68,68,36,127,0},{56,84,84,84,88,0},{8,126,9,1,6,0},{28,162,162,162,12
4,0},{127,8,8,8,112,0},{0,68,125,68,0,0},{0,32,64,64,61,0},{127,16,40,68,0,0},{0,127,64,0,
0,0},{124,4,124,4,120,0},{124,4,4,4,120,0},{56,68,68,68,56,0},{252,36,36,36,24,0},{24,36,
36,36,252,0},{124,4,4,4,8,0},{72,84,84,84,36,0},{4,63,68,64,32,0},{60,64,64,64,124,0},{28,
32,64,32,28,0},{60,64,56,64,60,0},{68,40,16,40,68,0},{28,160,160,160,124,0},{68,116,84,9
2,68,0}};
// в масиві, який розташований у Flash-пам’яті, зберігається інформація про
// зображення (фотографія)
const char image[512]={0,0,0,24,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
64,128,128,160,224,224,224,216,176,247,239,14,13,13,27,19,7,15,15,31,31,51,32,0,0,0,0,0,0,
1,2,0,0,1,2,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,192,192,128,64,132,133,139,127,255,255
,255,255,255,127,123,115,118,96,98,68,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,3,134,142,22,12,28,29,25,27,19,7,135,135,135,7,7,7,7,6,26,28,12,140,28,1
42,138,200,204,232,248,248,248,248,240,240,240,224,224,224,240,224,224,64,128,128,192,
192,0,128,192,128,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,132,198,199,99,99,7,6,14,159,255,255,255
,255,255,254,252,252,248,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,2
55,255,255,255,255,254,248,251,223,255,7,7,15,15,6,0,0,0,0,0,0,224,224,192,128,224,224,0,
0,0,0,1,1,0,192,228,252,254,255,231,255,255,255,255,255,255,127,127,63,63,191,251,243,24
3,243,227,227,199,199,7,15,15,223,255,255,255,255,255,255,255,255,255,63,7,1,1,4,0,0,12,1
24,112,0,6,134,140,143,183,63,115,121,252,124,126,254,124,112,56,140,135,95,63,127,255,
1,1,1,1,1,0,48,56,62,31,159,207,231,103,179,243,217,249,224,244,248,252,248,243,255,255,
255,255,255,255,127,1,128,192,224,104,0,32,0,0,0,1,6,14,12,31,31,31,15,6,30,62,240,222,22
3,60,62,127,127,127,92,28,8,0,0,0,0,0,0,56,48,48,0,4,15,14,6,135,195,195,199,255,255,255,2
55,255,255,255,255,255,63,31,15,7,129,208,254,255,255,246,250,240,240,248,254,254,240,0,0,0,
0,16,24,120,123,127,63,127,255,39,6,4,0,0,0,0,0,0,128,240,112,184,136,144,216,152,224,240
,176,32,195,7,143,15,143,31,159,143,207,199,131,1,1,128,8,28,8,65,97,89,239,255,255,255,2
55,255,255,255,255,254,255};
```

```
 // оголошення прототипів функцій
void init_device(void);          // функція ініціалізації контролера
void write(void);                // функція запису команди або даних в РКІ
void clear_lcd(void);            // функція очищення екрану індикатора
void clear_row(char row, char half);          // функція очищення окремого рядка;
// передаються номер рядка (0-7) і номер кристалу (1-лівий, 2 – правий, 0 – обидва)
void init_lcd(void);             // функція ініціалізації індикатора
void delay_ms(int);              // функція затримки; параметр – затримка в мс
void hi_z(void);                 // функція переведення виводів в стан високого імпеданса
void delay_us(int);              // функція затримки; параметр – затримка в мкс
void out_string(char *str, char row, char stpos);   // функція виведення текстового рядка
void draw_flash(char half);  // функція, яка виводить на індикатор дані з масиву image
// реалізація функцій
void init_device(void)           // функція ініціалізації контролера
{
 PORTA=0x00;                     // виведення значення 0 в порт A
 DDRA=0b00111111;                // налаштування шести розрядів порту A на вихід
 PORTB=0x00;                     // виведення значення 0 в порт B
 PORTC=0x00;                     // виведення значення 0 в порт C
 DDRC=0xFF;                      // налаштування виводів порту C на вихід
 PORTD=0x00;                     // виведення значення 0 в порт D
 DDRD=0x08;                      // налаштування виводів PD3, PD4, PD5 на вихід
 }
 void write(void)                // функція запису команд/даних в індикатор
 {
 char t;                         // оголошення додаткової змінної
 E();                            // установка тактового сигналу
 for(t=0;t<6;t++);               // затримка
 NE();                           // скидання тактового сигналу
 for(t=0;t<6;t++);               // затримка
}
 void hi_z(void)                 // функція переведення виводів в стан високого імпеданса
 {
 NCS1();                         // скидання сигналу CS1
 NCS2();                         // скидання сигналу CS2
 NRS();                          // скидання сигналу RS
 PORTC=0;                        // виведення в порт значення 0
 DDRC=0;                         // переведення порту C в стан високого імпеданса
 }

 void clear_lcd(void)            // функція очищення екрану індикатора
 {
 char i;                         // додаткова змінна для використання в циклі
 for(i=0;i<8;i++)                // цикл виконується 8 разів
 clear_row(i,0);                 // виклик функції очищення рядків обох кристалів дисплея
(параметр 0)
 DDRC=0xFF;                      // налаштування порту C на виведення даних
 NRS();                          // скидання сигналу RS – передаються команди
 PORTC=0b11000000;               // виклик команди Display Start Line
 write();                        // запис команди в індикатор
 hi_z();                         // переведення порту в стан високого імпеданса
```

```
}

void clear_row(char row, char half) // функція очищення окремого рядка індикатора
{
char i;                     // додаткова змінна для використання в циклі
PORTC=0;                    // виведення в порт С значення 0
DDRC=0xFF;                  // налаштування порту С на виведення даних
NRS();                      // скидання сигнала RS – передаються команди
NRW();                      // скидання сигналу RW
if(half==1)                 // якщо вибрана ліва половинка екрану
{
NCS1();                     // скидання сигналу CS1
CS2();                      // установка сигналу CS2
}
else if(half==2)            // якщо вибрана права половинка екрану
{
CS1();                      // установка сигналу CS1
NCS2();                     // скидання сигналу CS2
}
else                        // якщо вибрані обидві половинки екрану
{
NCS1();                     // скидання сигналу CS1
NCS2();                     // скидання сигналу CS2
}
row&=0x07;                  // виділення трьох молодших біт змінної row
row|=0b10111000;            // формування команди Set Page
PORTC=row;                  // виведення даних на порт С
write();                    // запис команди в індикатор
PORTC=0b01000000;           // передача команди Set address
write();                    // запис команди в модуль
RS();                       // установка сигналу RS – передаються дані
PORTC=0;                    // установка значення 0 на порт С
for(i=0;i<64;i++)           // цикл виконується 64 рази
{
 write();                   // запис 0 в пам’ять індикатора - очищення рядка
}
hi_z();                     // переведення порту в високоімпедансний стан
}

void init_lcd(void)         // функція ініціалізації РКІ
{
        char i;             // додаткова змінна
        NRES();             // скидання сигналу RES
        delay_us(500);      // часова затримка 500 мкс
        RES();              // установка сигналу RES
        delay_us(500);      // часова затримка 500 мкс
        NCS1();             // скидання сигналу CS1
        NCS2();             // скидання сигналу CS2
        NRW();              // скидання сигналу R/W
        NDI();              // скидання сигналу D/I (RS) – передаються команди
        PORTC=0b00111111;       // виклик команди Display ON
```

```
        write();                    // запись команди в індикатор
        PORTC=0b11000000;           // виклик команди Display StartLine
        write();                    // запись команди в індикатор
        PORTC=0b10111000;           // виклик команди Set Page
        write();                    // запис команд в модуль
        PORTC=0b01000000;           // виклик команди Set Address
        write();                    // запис команди в індикатор
        hi_z();                     // переведення порту в стан високого імпеданса
}
void delay_ms(int ms)               // функція затримки; параметр – затримка в мс
 {
  char i,j;                         // додаткові змінні
  int k;
  for(k=0;k<ms;k++)                 // у циклі контролер виконує «пусті» операції
        {
         for(i=0;i<20;i++)
         for(j=0;j<75;j++);
        }
}
void delay_us(int us)               // функція затримки; параметр – затримка в мкс
 {
 int k;                             // додаткова змінна
 for(k=0;k<=us;k++)                 // у циклі контролер виконує «пусті» операції
        {
                NOP();              // пуста операція
        }
}
void out_string(char * str, char row, char stpos)   // функція виведення текстового рядка
 {
 char len=strlen(str);          // визначення довжини, яка виводиться текстового рядка
 char i,j;                      // змінні для роботи алгоритма
 char curpos=stpos;             // запам'ятовування стартової позиції
 clear_row(row,0);              // виклик функції очищення рядка
 PORTC=0;                       // виведення значення 0 в порт С
 DDRC=0xFF;                     // налаштування порту С на вивід
 NRS();                         // скидання сигналу RS – передаються команди
 delay_us(2);                   // затримка 2 мкс
 NCS1();                        // скидання сигналу CS1
 NCS2();                        // скидання сигналу CS2
 row&=0x07;                     // виділення трьох молодших біт змінної row
 row=row|=0b10111000;           // формування команди Set Page
 PORTC=row;                     // виведення команди в порт
 write();                       // запис команди в порт
 CS2();                         // установка сигналу CS2
 stpos&=0b00111111;             // виділення 6 молодших біт значення стартової позиції
 stpos|=0b01000000;             // формування команди Set Address
 PORTC=stpos;                   // виведення команди в порт
 write();                       // запис команди в порт
 RS();                          // установка сигналу RS – передаються дані
 delay_us(2);                   // затримка 2 мс
 for(i=0;i<len;i++)             // цикл виконується стількои разів, скільки букв в рядку
```

```
{
 for(j=0;j<6;j++)              // виводяться дані про символи з двомірного масиву
 {
  PORTC=currfont[str[i]-32][j];      // установка значення j-елемента масиву на порту C
  write();                     // запис даних в РКІ
  curpos++;                    // збільшення на 1 внутрішнього лічильника
  if(curpos==64)               // якщо досягнуто край половинки екрану
  {
        CS1();                 // скидання сигналу CS1
        NCS2();                // установка сигналу CS2
        NRS();                 // скидання сигналу RS – передаються команди
        PORTC=0b01000000;          // установка команди Set Address
        write();               // запис команди в РКІ
        RS();                  // установка сигналу RS – передаються дані
  }
 }
}
hi_z();                        // переведення порту в стан високого імпеданса
}

void draw_flash(char half)  // функція виведення на екран даних
{
char i=0;                      // додаткова змінна
char page=0;                   // змінна – номер сторінки пам’яті
int a=0;                       // додаткова змінна
char data;
DDRC=0xFF;                     // налаштування порту C на вивід
if(half==1)                    // якщо значення змінної half дорівнює 1
{
CS2();                         // установка сигналу CS2
NCS1();                        // скидання сигналу CS1
}
else if(half==2)               // якщо значення змінної half дорівнює 2
{
NCS2();                        // скидання сигналу CS1
CS1();                         // установка сигналу CS2
}
else                           // якщо значення змінної half дорівнює 0
{
NCS2();                        // скидання сигналу CS2
NCS1();                        // скидання сигналу CS1
}
for(page=0;page<8;page++)// у циклі виконується запис даних в кожну сторінку пам’яті
{
NRS();                         // скидання сигналу RS – передається команда
PORTC=0b11000000;              // виведення команди Display Start Line
write();                       // запис команди в індикатор
PORTC=0b10111000|page; // виведення команди Set Page
write();                       // запис команди в индикатор
PORTC=0b01000000;              // виведення команди Set Address
write();                       // запис команди в індикатор
```

```
 for(i=0;i<64;i++)              // цикл повторюється 64 рази
 {
  RS();                         // установка сигналу RS – запис даних
  PORTC=~image[a];              // установка даних на порту C
  write();                      // запис даних в індикатор
  a++;                          // інкремент змінної
 }
 }
 hi_z();                        // переведення порту в стан високого імпеданса
 }

 void main(void)
 {
 char name[]="Photo on";    // оголошення змінних, в яких зберігається текстова
інформація
 char sname[]="Graphic";    // кожна змінна – одномірний масив
 char surname[]="Display"; // елементи масиву – окремі букви
 char ed1[]="Magister of";  // розмір кожного з масивів дорівнює кількості букв у слові
 char ed2[]="electronics";
 char vuz[]="TEES 2011";
 init_device();                 // виклик функції ініціалізації контролера
 init_lcd();                    // виклик функції ініціалізації РКІ
 clear_lcd();                   // очищення екрану РКІ
 out_string(name,0,0);          // виклик функції виведення тексту
 out_string(sname,1,0);         // виклик функції виведення тексту
 out_string(surname,2,0);       // виклик функції виведення тексту
 out_string(ed1,4,0);  // виклик функції виведення тексту; 4 рядок починаючи з позиції 0
 out_string(ed2,5,0); // виклик функції виведення тексту; 5 рядок починаючи з позиції 0
 out_string(vuz,7,0); // виклик функції виведення тексту; 7 рядок починаючи з позиції 0
 draw_flash(2);         // виклик функції виведення зображення у правій частині екрану
 while(1);              // зациклення мікроконтролера
 }
```

Фотографія, яка відображається на екрані РКІ, зберігається у flash-пам'яті мікроконтролера. Розмір фотографії дорівнює 8×64=512 байт. Також в пам'яті мікроконтролера зберігається інформація про шрифт. Для одного символу потрібно 6 байт. З метою економії пам'яті можна зберігати в пам'яті лише ті символи, які відображатимуться. Формування на екрані РКІ фотографії відбувається таким чином. У циклі послідовно на екран виводяться значення з масиву. Коли значення індексу досягає 63, відбувається зміна адреси сторінки пам'яті, в яку записуються дані. У межах однієї сторінки немає необхідності установки зсуву по горизонталі, оскільки контролер РКІ автоматично збільшує значення лічильника зсуву на 1 після запису чергового значення.

Після компіляції проекту і програмування мікроконтролера, на екрані графічного РКІ з'явиться текстова інформація і зображення, як показано на рис. 4.39.



Рис. 4.39.

Приклад 4.10. Написати програму, яка обчислює взаємну кореляційну функцію прямокутного і трикутного імпульсів, і виводить графік функції на екрані РКІ. Принципова схема пристрою представлена на рис. 4.36.

Текст програми представлено нижче.

```
#include <io8535v.h>                    // підключення заголовочних файлів
#include <macros.h>
#include <math.h>
#define CS1() PORTA|=0x01
#define NCS1() PORTA&=~0x01
#define CS2() PORTA|=0x02
#define NCS2() PORTA&=~0x02
#define RES() PORTA|=0x04
#define NRES() PORTA&=~0x04
#define RW() PORTA|=0x08
#define NRW() PORTA&=~0x08
#define DI() PORTA|=0x10
#define NDI() PORTA&=~0x10
#define RS() DI()
#define NRS() NDI()
#define E() PORTA|=0x20
#define NE() PORTA&=~0x20
#define SET_PIXEL 0
#define CLR_PIXEL 1
#define MASK16 0b00001111
#define MASK32 0b00011111
#define MASK64 0b00111111
void delay_us(int us)                   // функція часової затримки
{
        int k;
        CLI();                          // заборона переривань
        for(k=0;k<=us;k++) NOP();
}
```

```
void write(void)                        // функція запису в РКІ
{
 char t;
 E();                                   // установка сигналу E
 for(t=0;t<6;t++);
 NE();                                  // скидання сигналу E
 for(t=0;t<6;t++);
}
void hi_z(void)                         // функція переведення порту в третій стан
{
 NCS1();                                // скидання сигналу CS1
 NCS2();                                // скидання сигналу CS2
 NRS();                                 // скидання сигналу RS
 PORTC=0;
 DDRC=0;                                // налаштування порту C на вхід
}
void clear_row(char row, char half)     // функція очищення рядка
{
 char i;
 PORTC=0;
 DDRC=0xFF;                             // налаштування порту C на вихід
 NRS();                                 // скидання сигналу RS – передача команд
 NRW();
 if(half==1)                            // якщо вибрана ліва половина екрану
 {
 NCS1();                                // скидання сигналу CS1
 CS2();                                 // установка сигналу CS2
 }
 else if(half==2)                       // якщо вибрана права половина екрану
 {
 CS1();                                 // установка сигналу CS1
 NCS2();                                // скидання сигналу CS2
 }
 else                                   // якщо вибрані обидві половинки екрану
 {
 NCS1();                                // скидання сигналу CS1
 NCS2();                                // скидання сигналу CS2
 }
 row&=0x07;                             // виділення 3-х молодших біт
 row|=0b10111000;                       // формування команди
 PORTC=row;                             // передача команди в РКІ-модуль
 write();                               // запис команди
 PORTC=0b01000000;
 write();
 RS();                                  // установка сигналу RS
 PORTC=0;                               // виведення в порт C значення 0
 for(i=0;i<64;i++) write();             // запис значення 0 в РКІ – очищення рядка
 hi_z();                                // переведення порту в третій стан
}
void init_lcd(void)                     // функція ініціалізації РКІ
{
```

```
 NRES();                                  // скидання сигналу RES
 delay_us(500);                           // пауза 500 мкс
 RES();                                   // установка сигналу RES
 delay_us(500);                           // пауза 500 мкс
 NCS1();                                  // скидання сигналу CS1
 NCS2();                                  // скидання сигналу CS2
 NRW();                                   // скидання сигналу RW
 NDI();                                   // скидання сигналу DI
 PORTC=0b00111111;                        // виведення команди в порт
 write();                                 // запис команди в РКІ-модуль
 PORTC=0b11000000;
 write();
 PORTC=0b10111000;
 write();
 PORTC=0b01000000;
 write();
 hi_z();                                  // переведення порту в третій стан
}
void clear_lcd(void)                      // функція очищення екрану РКІ
{
 char i;
 for(i=0;i<8;i++)                         // у циклі очищається кожен рядок РКІ
 clear_row(i,0);
 DDRC=0xFF;                               // налаштування порту С на вихід
 NRS();                                   // скидання сигналу RS
 PORTC=0b11000000;                        // виведення команди в порт
 write();                                 // запис команди
 hi_z();                                  // переведення порту в третій стан
}
void out_line(char vali, char coli, char spage, char mask) // функція виводу лінії
{
 char tmp=0;                              // оголошення змінних
 char page;
 char k=0;
 char val;
 char col;
 col=coli;
 val=vali;
 page=spage;
 DDRC=0xFF;                               // налаштування порту С на вихід
 PORTC=0;
 val=val&mask;                            // виділення декілька біт
 col=col&0b01111111;                      // 7 біт для обліку 127 стовпців
 if(col>63)                               // якщо номер стовпця більше 63
 {
  CS1();                                  // включається ліва половина екрану
  NCS2();                                 // виключається права половина екрану
  col=col-64;                             // зсув номера стовпця на 65
 }
 else
 {
```

```
 NCS1();                          // включення лівої половинки екрану
 CS2();                           // виключення правої половинки екрану
 }
 while(val>8)
 {
 NRS();                           // передаються команди в РКІ
 PORTC=0b10111000|page;           // формування команди установки сторінки
 write();                         // запис команди в РКІ
 page--;                          // зменшення номеру сторінки на 1
 PORTC=0b01000000|col;            // формування команди установки стовпця
 write();                         // запис команди в РКІ
 RS();                            // далі передаються дані
 PORTC=0xFF;                      // виведення в порт значення 255
 write();                         // запис значення в модуль
 val=val-8;
 }
 for(k=0;k<val;k++)
 {
  tmp=tmp>>1;
  tmp|=0x80;
 }
 NRS();                           // передаються команди
 PORTC=0b10111000|page;           // установка адреси сторінки
 write();
 PORTC=0b01000000|col;            // установка адреси стовпця
 write();
 RS();                            // передаються дані
 PORTC=tmp;
 write();                         // запис даних в модуль
 while(page>0)                    // доки не досягнута 0 сторінка пам'яті РКІ
  {
  NRS();                          // предаються команди
  page--;
  PORTC=0b10111000|page;          // установка номеру сторінки пам'яті
  write();
  PORTC=0b01000000|col;           // установка номеру стовпця
  write();
  RS();                           // передаються дані
  PORTC=0;                        // виведення значення 0 в порт С
  write();                        // запис даних в РКІ-модуль
  }
  hi_z();                         // переведення порту в третій стан
}
void main(void)                   // головна функція програми
{
char s1[32], s2[32];              // масиви для зберігання вибірок
char n,m;
double corr[64];                  // масив для зберігання функції кореляції
char rcorr[64];
for(n=0;n<32;n++)                 //  формування сигналів
```

```
{
 s1[n]=32;                              // прямокутний імпульс
 s2[n]=n;                               // трикутний імпульс
 corr[n]=0;                             // очищення масиву
 corr[n+32]=0;
}
 DDRA=0xFF;                             // налаштування порту A на вихід
 DDRC=0xFF;                             // налаштування порту C на вихід
 PORTD=0x00;                            // виведення 0 в порт D
 DDRD=0x38;
 init_lcd();                            // виклик функції ініціалізації РКІ
 clear_lcd();                           // виклик функції очищення РКІ
        for(n=0;n<32;n++)               // обчислення функції кореляції
        {
                for(m=0;m<=n;m++)
                {
                 corr[n]=corr[n]+s1[m]*s2[31-n+m];
                }
                corr[n]=corr[n]/256;
        }
        for(n=0;n<32;n++)
        {
                for(m=n;m<32;m++)
                {
                 corr[n+32]=corr[n+32]+s1[m]*s2[m-n];
                }
                corr[n+32]=corr[n+32]/256;
        }
        for(n=0;n<64;n++) rcorr[n]=(char)corr[n];
for(n=0;n<64;n=n+1) out_line(rcorr[n],n*2,7,MASK64);  // виведення результату на РКІ
while(1);                               // вічний цикл
}
```

Для відображення графіка в програмі реалізована функція out_line(), яка виводить на екран вертикальну лінію у вказаній позиції. Ініціалізація РКІ відбувається шляхом формування сигналу низького рівня на вході RES протягом 500 мс. Після цього в модуль передається команда вмикання екрану, установка початкової сторінки і початкової позиції курсора. Функція write() формує тактовий імпульс, по фронту якого відбувається запис команд або даних в модуль. Низький рівень сигналу на вході RS означає, що передаються команди, а високий рівень сигналу означає, що передаються дані. Як і раніше, для зручності написання програми операціям з бітами порту призначені відповідні псевдоніми з використанням директиви #define. На рис. 4.40 представлений результат виконання програми. На екрані РКІ відображається функція взаємної кореляції прямокутного і трикутного імпульсів.

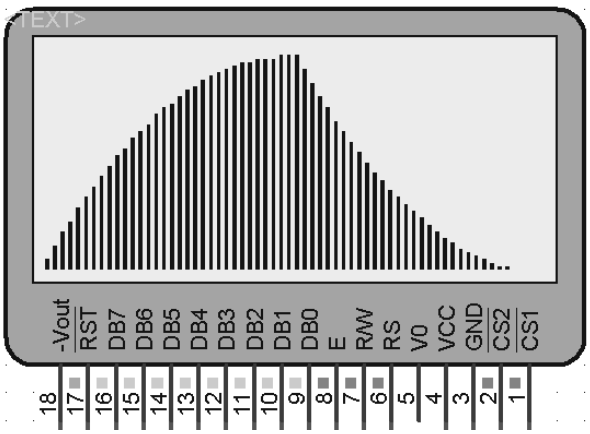


Рис. 4.40.

Приклад 4.7. Вивести на екран РКІ олимпійські кільця. Текст програми представлений нижче.

```
#include <io8535v.h>                    // підключення заголовочних файлів
#include <string.h>
#include <macros.h>
#include <math.h>
#define CS1() PORTA|=0x01
#define NCS1() PORTA&=~0x01
#define CS2() PORTA|=0x02
#define NCS2() PORTA&=~0x02
#define RES() PORTA|=0x04
#define NRES() PORTA&=~0x04
#define RW() PORTA|=0x08
#define NRW() PORTA&=~0x08
#define DI() PORTA|=0x10
#define NDI() PORTA&=~0x10
#define RS() DI()
#define NRS() NDI()
#define E() PORTA|=0x20
#define NE() PORTA&=~0x20
#define MASK16 0b00001111
#define MASK32 0b00011111
#define MASK64 0b00111111
#define SET_PIXEL 0
#define CLR_PIXEL 1
#define CIRCLE_DRAW SET_PIXEL
#define CIRCLE_CLEAR CLR_PIXEL
#define _DRAW_CIRCLE_

int addr;                               // оголошення глобальних змінних
char flag;
char row;
                                        // оголошення прототипів функцій
```

```
void init_device(void);
void write(void);
void clear_lcd(void);
void clear_row(char row, char half);
void init_lcd(void);
void delay_ms(int);
void hi_z(void);
void delay_us(int);
void out_line(char val, char col, char spage, char mask);
void setpixel(char x, char y, char clear);
void settoprow(char row);
void draw_circle(char x0, char y0, char rad, char rop);

void init_device(void)                  // функція ініціалізації мікроконтролера
{
 PORTA=0x00;                            // в порт А записується 0
 DDRA=0b00111111;                       // налаштування порту А на вихід
 PORTB=0x00;
 DDRB=0x00;                             // порт В налаштований на вхід
 PORTC=0x00;
 DDRC=0xFF;                             // налаштування порта С на вихід
 PORTD=0x00;
 DDRD=0x38;
 addr=0;                                // обнуління змінних
 row=0;
}
void write(void)                        // функція запису команди в РКІ
{
 char t;
 E();                                   // установка сигналу Е
 for(t=0;t<6;t++);                      // затримка
 NE();                                  // скидання сигналу Е
 for(t=0;t<6;t++);                      // затримка
}
void hi_z(void)                         // функція переведення порту в третій стан
{
 NCS1();                                // скидання сигналів CS1 і CS2
 NCS2();
 NRS();
 PORTC=0;                               // в порт С виводиться 0
 DDRC=0;                                // порт С налаштований на вхід
}

void clear_row(char row, char half)     // функція очищення рядка
{
 char i;
 PORTC=0;
 DDRC=0xFF;                             // налаштування порту С на вихід
 NRS();                                 // скидання сигналу RS – передача команд
 NRW();
 if(half==1)                            // якщо вибрано ліву половину екрану
```

```
 {
 NCS1();                              // скидання сигналу CS1
 CS2();                               // установка сигналу CS2
 }
 else if(half==2)                     // якщо вибрано праву половину екрану
 {
 CS1();                               // установка сигналу CS1
 NCS2();                              // скидання сигналу CS2
 }
 else                                 // якщо вибрані обидві половинки екрану
 {
 NCS1();                              // скидання сигналу CS1
 NCS2();                              // скидання сигналу CS2
 }
 row&=0x07;                           // виділення 3-х молодших біт
 row|=0b10111000;                     // формування команди
 PORTC=row;                           // передача команди в РКІ-модуль
 write();                             // запис команди
 write();
 RS();                                // установка сигналу RS
 PORTC=0;                             // виведення в порт С значення 0
 for(i=0;i<64;i++) write();           // запис значення 0 в РКІ – очищення рядка
 hi_z();                              // переведення порту в третій стан
}

void init_lcd(void)                   // функція ініціалізації РКІ
{
 NRES();                              // скидання сигналу RES
 delay_us(500);                       // пауза 500 мкс
 RES();                               // установка сигналу RES
 delay_us(500);                       // пауза 500 мкс
 NCS1();                              // скидання сигналу CS1
 NCS2();                              // скидання сигналу CS2
 NRW();                               // скидання сигналу RW
 NDI();                               // скидання сигналу DI
 PORTC=0b00111111;                    // виведення команди в порт
 write();                             // запис команди в РКІ-модуль
 PORTC=0b11000000;
 write();
 PORTC=0b10111000;
 write();
 PORTC=0b01000000;
 write();
 hi_z();                              // переведення порту в третій стан
}
void delay_ms(int ms)                 // функція часової затримки
{
 char i,j;
 int k;
 CLI();                               // заборона обробки переривань
 for(k=0;k<ms;k++)
```

```
 {
  for(i=0;i<20;i++)                    // у циклі виконується пуста операція
  for(j=0;j<71;j++);
 }
 SEI();                                // дозвіл обробки переривань
}

void delay_us(int us)                  // функція часової затримки
{
 char i;
 int k;
 CLI();                                // заборона обробки переривань
 for(k=0;k<=us;k++)
 {
        NOP();                         // пуста операція
 }
 SEI();                                // дозвіл обробки переривань
}
void setpixel(char x, char y, char clear)  // функція виведення точки на екран РКІ
{
 char page=0;                          // оголошення змінних
 char pos=0;
 char i=0;
 char pix=0;
 char t;
 x=x&0b01111111;                       // виділення 7 біт за допомогою маски
 y=y&0b00111111;
 NRS();                                // скидання сигналу RS
 NRW();                                // скидання сигналу RW
 DDRC=0xFF;                            // налаштування порту С на вихід
  if(x>63)                             // якщо координата х більше 63
 {
  CS1();                               // виключення лівої половинки екрану
  NCS2();                              // включення правої половинки екрану
  x=x-64;
 }
 else                                  // інакше
 {
  NCS1();                              // включення лівої половинки екрану
  CS2();                               // виключення правої половинки екрану
 }
 page=y>>3;                            // формування адреси сторінки пам'яті
 pos=y-page*8;
 i=1<<pos;
 PORTC=0b11000000;                     // виведення команди в порт С
 write();                              // запис команди в РКІ-модуль
 PORTC=0b10111000|page;                // виведення команди установки сторінки
 write();                              // запис команди в модуль
 PORTC=0b01000000|x;                   // виведення команди установки стовпця
 write();                              // запис команди в модуль
 PORTC=0x00;                           // виведення 0 в порт С
```

```
 DDRC=0x00;                                  // налаштування порту С на вхід
 RW();                                       // установка сигналу зчитування
 RS();                                       // далі передаються дані
 E();                                        // установка сигналу Е
 for(t=0;t<3;t++);                           // пауза
 pix=PINC;                                   // зчитування даних з порту С
 NE();                                       // скидання сигналу Е
 NRS();                                      // скидання сигналу RS
 NRW();                                      // скидання сигналу RW
 PORTC=0b10111000|page;                      // установка адреси сторінки
 write();                                    // запис команди в модуль
 PORTC=0b01000000|x;                         // установка номеру стовпця
 write();                                    // запис команди в модуль
 PORTC=0x00;                                 // виведення значення 0 в порт С
 DDRC=0x00;                                  // налаштування порту С на вхід
 RW();                                       // установка сигналу RW
 RS();                                       // установка сигналу RS
 E();                                        // установка сигналу Е
 for(t=0;t<3;t++);                           // пауза
 pix=PINC;                                   // зчитування даних з РКІ
 NE();                                       // скидання сигналу Е
 NRS();                                      // скидання сигналу RS
 NRW();                                      // скидання сигналу RW
 DDRC=0xFF;                                  // налаштування порту С на вихід
 PORTC=0b10111000|page;                      // установка сторінки пам'яті
 write();                                    // запис команди в модуль
 PORTC=0b01000000|x;                         // установка номеру стовпця
 write();                                    // запис команди в модуль
 RS();                                       // передаються дані
 if(clear==SET_PIXEL) pix|=i;                // установка біта в змінній
 else if(clear==CLR_PIXEL) pix&=(~i);        // скидання біта в змінній
 PORTC=pix;                                  // виведення значення в порт
 write();                                    // запис значення в РКІ-модуль
 hi_z();                                     // переведення порту в третій стан
}
void settoprow(char row)                     // функція установки номеру першого рядка
{
 NCS1();                                     // включення обох половинок екрану
 NCS2();
 NRW();                                      // далі відбувається запис команд
 NRS();
 DDRC=0xFF;                                  // налаштування порту С на вихід
 row=row&0b00111111;                         // формування команди
 PORTC=0b11000000|row;                       // виведення команди в порт
 write();                                    // запис команди в модуль
 hi_z();                                     // переведення порту в третій стан
}
void draw_circle(char x0, char y0, char rad, char rop) // функція виводу кола
{
        char i;
        char r;
```

```
        int sr=rad*rad;
        for(i=0;i<=rad;i++)
                {
                r=sqrt(sr-i*i);                 // обчислення радіусу
                setpixel(x0-i,y0-r,rop);        // по координатам малюються точки кола
                setpixel(x0+i,y0-r,rop);
                setpixel(x0-i,y0+r,rop);
                setpixel(x0+i,y0+r,rop);
                }
}
void main(void)                                 // головна функція програми
{
 float f;                                       // оголошення додаткових змінних
 float rad;
 float pi=3.14;
 char r=0;
 init_device();                                 // виклик функції ініціалізації МК
 init_lcd();                                    // виклик функції ініціалізації РКІ
 clear_lcd();                                   // очищення екрану РКІ
 while(1)
 {
        draw_circle(42,21,15,CIRCLE_DRAW);      // малювання кілець
        draw_circle(64,21,15,CIRCLE_DRAW);
        draw_circle(86,21,15,CIRCLE_DRAW);
        draw_circle(52,40,15,CIRCLE_DRAW);
        draw_circle(74,40,15,CIRCLE_DRAW);
        while(1)                                        // вічний цикл
        {
         for(r=0;r<63;r++)
        {
         settoprow(r);                          // у циклі відбувається прокрутка зображення
         delay_ms(100);                         // затримка 100 мс
        }}}
}
```

У наведеному прикладі формування графіки виконується програмно. Алгоритм малювання кола оформлений у вигляді функції draw_circle(), параметрами якої є координати центру кола, а також радіус. В свою чергу, у функції draw_circle() використовується більш низькорівнева функція setpixel(), яка запалює або гасить точку у вказаній позиції екрану.

Для прокрутки зображення по вертикалі використовується функція settoprow(), параметром якої є номер першого рядка, що відображується. У цій функції виконується передача команди Display Start Line в РКІ-модуль (табл. 4.10).

На рис. 4.41 представлено результат виконання програми.

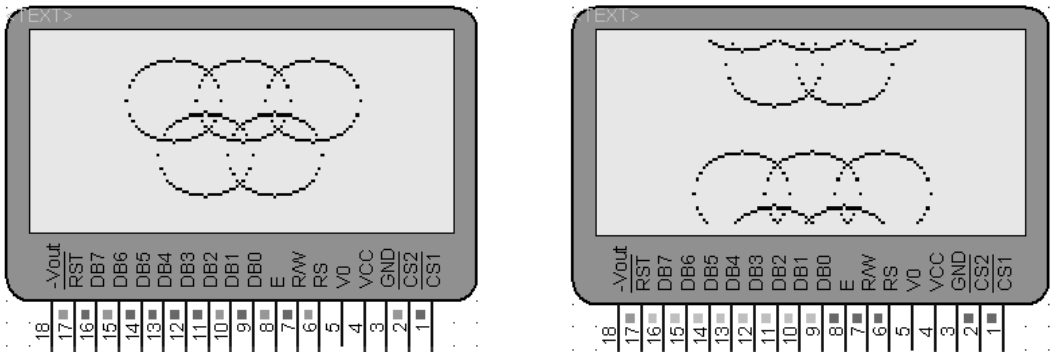


Рис. 4.41.

4.5.6. Управління РК-індикатором LM234

Рідкокристалічний модуль LM234 складається з чотирьох мікросхем контролера HD44102 і РК панелі. Модуль дозволяє відображати на графічному полі розмірами 100×80 точок. Кожній точці, що світиться на РК панелі, відповідає 1 в ОЗП контролера. Габаритні розміри дисплея 140×95×19. На рис. 4.42 показаний зовнішній вигляд модуля.

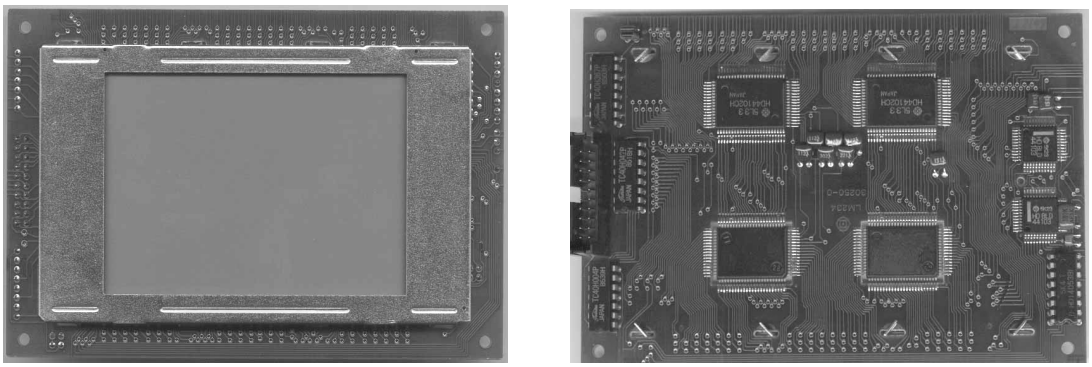


Рис. 4.42.

Даний модуль є РК-панеллю, що обслуговується чотирма однотипними контролерами HD44102 фірми Hitachi. Кожен контролер обслуговує свою область екрану розмірами 50×40 точок. На платі дисплея також розташовано два контролери рядків HD44102, дешифратор 74HC42 (повний функціональний аналог КР555ІД6) і буферні елементи. Дешифратор призначений для вибору одного з кристалів HD44102, на його входи (CS1, CS2, CS3) подається двійково-десятковий код, відповідний одній з мікросхем контролера. Входи R/W – зчитування/запис, D/I – дані/команди та EN – вибір мікросхеми, сполучені паралельно. Кожен контролер має чотири сторінки пам'яті, які відповідають даним на індикаторі. Таким чином, виведення даних на дисплей зводиться до почергового програмування мікросхем контролерів. Оскільки швидкодія даних контролерів дастатньо низька, необхідно в процесі роботи контролювати прапорець зайнятості BUSY FLAG, і не здійснювати обміну

інформацією з контролером доки даний прапор не буде скинутий. Інший, простіший спосіб, полягає в програмному формуванні затримок між пердачами команд/даних в модуль.

До основних операцій при роботі з модулем належать:

- прийом команд з шини даних DB0-DB7;
- запис даних в ОЗП по 8-ми розрядній шині даних DB0-DB7;
- зчитування статусу стану по шині даних DB0-DB7;
- управління контрастністю здійснюється зовнішнім резистором.

Призначення виводів роз'єму підключення індикатора приведено в табл. 4.11.

Табл. 4.11.

№	Познач	Призначення	№	Познач.	Призначення
1	Gnd	Загальний контакт	11	R/W	Зчитування/Запис
2	+5В	Живлення модуля	12	D/I	Дані/команди
3	CT	Керування контрастністю	13	DB0	Шина даних 0-й розряд
4		Не викор.	14	DB1	Шина даних 1-й розряд
5		Не викор.	15	DB2	Шина даних 2-й розряд
6	CS1	Вибір кристалу	16	DB3	Шина даних 3-й розряд
7	CS2	Вибір кристалу	17	DB4	Шина даних 4-й розряд
8	CS3	Вибір кристалу	18	DB5	Шина даних 5-й розряд
9		Не викор.	19	DB6	Шина даних 6-й розряд
10	EN	Загальний дозвіл	20	DB7	Шина даних 7-й розряд

Для програмування індикатора необхідно виконати наступну послідовність дій: записати команду Display ON, вибрати один з чотирьох контролерів за допомогою входів CS1, CS2, CS3, вказати сторінку ОЗП (тобто потрібний рядок), вказати координату X (зсув по горизонталі), і записати байт інформації. Система команд для управління модулем приводиться в табл. 4.12.

Табл. 4.12.

Команда	R/W	D/I	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Display On Display Off	0 0	0 0	0 0	0 0	1 1	1 1	1 1	0 0	0 0	1 0
Set X/Y Adress	0	0	X-adress	X-adress	Y-adress	Y-adress	Y-adress	Y-adress	Y-adress	Y-adress
Read Data	1	1								
Write Data	0	1								
Display start page	0 0 0 0	0 0 0 0	0 0 1 1	0 1 0 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	0 0 0 0
Up set Down set	0 0	0 0	0 0	0 0	1 1	1 1	1 1	0 0	1 1	1 0
Status read	1	0	Busy flag	Up/ Down	Off/On	Reset	0	0	0	0

Принцип організації пам’яті РКІ-модуля показаний на рис. 4.43.

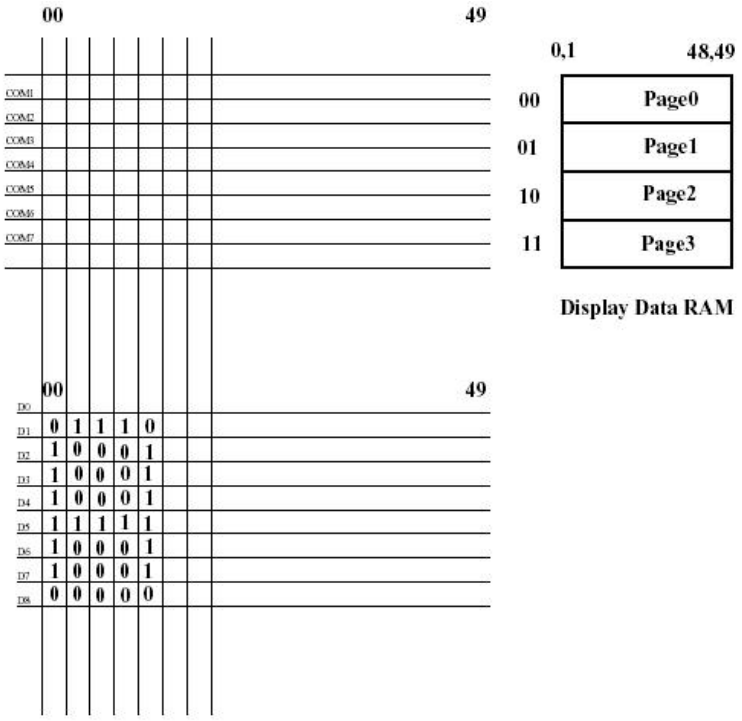


Рис. 4.43.

Кожен з 4-х контролерів, який обслуговує певну область екрану, має 4 сторінки пам'яті, висота кожної з яких дорівнює 10 пікселям, а довжина 50 пікселям. За допомогою команди Set X/Y Address є можливість вибрати необхідну сторінку пам'яті (тобто рядок), а також встановити зсув по горизонталі (координата Y). Після цього передавати в модуль байти даних, які, власне, і формують зображення.

Приклад 4.12. Розробити програму, яка виводить 4 рядки текстової інформації на рідкокристалічний модуль.

Текст програми з коментарями приводиться нижче.

```
#include <io8535v.h>                // підключення заголовочних файлів
#include <macros.h>
#define RW() PORTD|=0x08            // установка сигналу зчитування/запис
#define NRW() PORTD&=~0x08          // скидання сигналу зчитування/запис
#define DI() PORTD|=0x10            // установка сигналу дані/команди
#define NDI() PORTD&=~0x10          // скидання сигналу дані/команди
#define RS() DI()                   // установка сигналу дані/команди
#define NRS() NDI()                 // скидання сигналу дані/команди
#define E() PORTD|=0x20             // установка тактового сигналу
#define NE() PORTD&=~0x20           // скидання тактового сигналу
#define CS0() PORTA=0               // вибір першого кристалу – верхній лівий
#define CS1() PORTA=0x20            // вибір другого кристалу – верхній правий
#define CS2() PORTA=0x40            // вибір третього кристалу – нижній лівий
#define CS3() PORTA=0x60            // вибір четвертого кристалу – нижній правий
// у двовимірному масиві містяться дані для формування символів – шрифт
```

```
// передається ASCII-код символу в якості індекса масиву
const char
currfont[91][6]={{0,0,0,0,0,0},{0,6,95,6,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{36,74,255,82,36,0}
,{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0
},{0,0,160,96,0,0},{0,0,0,0,0,0},{0,0,96,96,0,0},{0,0,0,0,0,0},{62,65,65,65,62},{0,4,2,127,0,
0},{66,97,81,73,70,0},{34,65,73,73,54,0},{56,36,34,127,64,0},{47,73,73,73,49,0},{62,73,73
,73,50,0},{65,33,17,9,7,0},{54,73,73,73,54,0},{38,73,73,73,62,0},{0,0,0,0,0,0},{0,0,0,0,0,0},
{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{124,18,17,18,124,0},{12
7,73,73,73,54,0},{62,65,65,65,34,0},{127,65,65,65,28,0},{127,73,73,73,73,0},{127,9,9,9,1,0
},{62,65,65,81,50,0},{127,8,8,8,127,0},{129,129,127,129,129,0},{48,64,64,64,63,0},{127,8,
20,34,65,0},{127,64,64,64,64,0},{127,2,4,2,127,0},{127,4,8,16,127,0},{62,65,65,65,62,0},{
127,9,9,9,6,0},{62,65,73,81,62,0},{127,9,25,41,70,0},{38,73,73,73,50,0},{1,1,127,1,1,0},{6
3,64,64,64,63,0},{31,32,64,32,31,0},{127,64,48,64,127,0},{99,20,8,20,99,0},{3,4,120,4,3,0}
,{105,89,73,77,75,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0,0,0,0,0},{0,0
,0,0,0,0},{56,68,68,36,120,0},{127,72,72,72,48,0},{56,68,68,68,40,0},{56,68,68,36,127,0},{
56,84,84,84,88,0},{8,126,9,1,6,0},{28,162,162,162,124,0},{127,8,8,8,112,0},{0,68,125,68,0,
0},{0,32,64,64,61,0},{127,16,40,68,0,0},{0,127,64,0,0,0},{124,4,124,4,120,0},{124,4,4,4,12
0,0},{56,68,68,68,56,0},{252,36,36,36,24,0},{24,36,36,36,252,0},{124,4,4,4,8,0},{72,84,84,
84,36,0},{4,63,68,64,32,0},{60,64,64,64,124,0},{28,32,64,32,28,0},{60,64,56,64,60,0},{68,
40,16,40,68,0},{28,160,160,160,124,0},{68,116,84,92,68,0}};
// оголошення прототипів функцій
void init_device();                    // функція ініціалізації контролера
void display_on();                     // функція включення дисплея
void delay_us(int us);                 // функція затримки; параметр – затримка в мкс
void delay_ms(int ms);                 // функція затримки; параметр – затримка в мс
void write_cmd(char c);                // функція запису команди в РК-модуль
void write_data(char c);               // функція запису даних в РК-модуль
void out_char(unsigned char c);        // функція виведення символа на індикатор
void out_text(char* c,  unsigned char size, char line);     // функція виведення рядка
void clear_lcd();                      // функція очищення індикатора
// реализація функцій
void init_device()                     // ініціалізація контролера
{
 PORTA=0;                              // виведення значення 0 в порт A
 DDRA=0b11100000;                      // налаштування трьох біт порту A на вихід
 PORTC=0;                              // виведення значення 0 в порт C
 DDRC=0;                               // порт C налаштований на вихід
 PORTD=0;                              // установка значення 0 на порту D
 DDRD=0xFF;                            // налаштування порту D на вивід
}
void init_lcd()                        // функція ініціалізації РКІ
{
 CS0();                                // вибір першого кристалу
 write_cmd(0b00111001);                // запис команди Display ON
 write_cmd(0);                         // запис команди Set X/Y Address
 write_cmd(0b00111011);                // запис команди Up Set
 write_cmd(0b00111110);                // запис команди Display Start Page;
 clear_lcd();                          // виклик функції очищення екрану індикатора
 CS1();                                // вибір другого кристалу
 write_cmd(0b00111001);                // запис команди Display ON
 write_cmd(0b00000000);                // запис команди Set X/Y Address
```

```
 write_cmd(0b00111011);          // запис команди Up Set
 write_cmd(0b00111110);          // запис команди Display Start Page;
 clear_lcd();                    // виклик функції очищення екрану індикатора
 CS2();                          // вибір третього кристалу
 write_cmd(0b00111001);          // запис команди Display ON
 write_cmd(0b00000000);          // запис команди Set X/Y Address
 write_cmd(0b00111011);          // запис команди Up Set
 write_cmd(0b00111110);          // запис команди Display Start Page;
 clear_lcd();                    // виклик функції очищення екрану індикатора
 CS3();                          // вибір четвертого кристалу
 write_cmd(0b00111001);          // запис команди Display ON
 write_cmd(0b00000000);          // запис команди Set X/Y Address
 write_cmd(0b00111011);          // запис команди Up Set
 write_cmd(0b00111110);          // запис команди Display Start Page;
 clear_lcd();                    // виклик функції очищення екрану індикатора
}
void delay_us(int us)        // функція затримки; параметр – величина затримки в мкс
{
        int k;                  // додаткова змінна
        for(k=0;k<=us;k=k+1)    // цикл виконується us разів
        {
        NOP();                  // пуста операція
        NOP();                  // пуста операція
        }
}
void delay_ms(int ms)        // функція затримки; параметр – величина затримки в мс
{
        char i,j;               // додаткові змінні
        for(i=0;i<ms;i++)       // цикл, виконується ms разів
        for(j=0;j<970;j++) NOP(); // у циклі виконується пуста операція 970 разів
}
void write_cmd(char c)          // функція запису команди в РК-модуль
{
  NRS();            // скидання сигналу команди/дані – передаються команди
  NRW();            // скидання сигналу зчитування/запис – команди передаються в РКІ
  NE();                         // скидання тактового сигналу
  PORTC=c;                      // виведення команди в порт С
  E();                          // установка тактового сигналу
  delay_us(10);                 // невелика пауза
  NE();                         // скидання тактового сигналу
  delay_us(10);                 // затримка
}
void write_data(char c)         // функція запису байта даних в РК-модуль
{
 RS();              // установка сигналу команди/дані – передаються дані
 NRW();             // скидання сигналу запис/зчитування – дані записуються в РКІ
 NE();                          // скидання тактового сигналу
 PORTC=c;                       // виведення байта даних в порт С
 E();                           // установка тактового сигналу
 delay_us(10);                  // затримка
 NE();                          // скидання тактового сигналу
```

```
  delay_us(10);                    // затримка
}
void out_char(unsigned char c)     // функція виводу символу на екран РКІ
{
 char b=0;                         // додаткова змінна
 for(b=0;b<5;b=b+1)   write_data(currfont[c][b]);
// у циклі в індикатор передається 6 байт даних, які формують зображення символу
}
// функція виводу рядка в задану позицію
void out_text(char* c, unsigned char size, char line)
{
 char b=0;                         // додаткові змінні
 char b1=0;
 char ln=line<<6;                  // формування команди Set X/Y Address
 write_cmd(ln);                    // передача команди Set X/Y Address в модуль
 for(b=0;b<size;b=b+1)             // в циклі виводиться по черзі кожен символ рядка
   for(b1=0;b1<5;b1=b1+1)  write_data(currfont[c[b]][b1]>>1);    // записуються дані,
                                   // які формують текстовий рядок на індикаторі
}
void clear_lcd()                   // функція очищення екрана індикатора
{
 char b;                           // додаткова змінна
 write_cmd(0);                     // передача команди Set X/Y Address – перший рядок
 for(b=0;b<50;b++)  write_data(0); // у циклі 50 разів записується 0
 write_cmd(0b01000000);            // передача команди Set X/Y Address – другий рядок
 for(b=0;b<50;b++)  write_data(0); // у циклі 50 разів записується 0
 write_cmd(0b10000000);            // передача команди Set X/Y Address – третій рядок
 for(b=0;b<50;b++)  write_data(0); // у циклі 50 разів записується 0
 write_cmd(0b11000000);            // передача команди Set X/Y Address
for(b=0;b<50;b++)  write_data(0);  // у циклі 50 разів записується 0
 write_cmd(0b00000000);            // передача команди Set X/Y Address – перший рядок
}
void main()                        // головна функція програми
{
 init_device();                    // виклик функції ініціалізації контролера
 init_lcd();                       // виклик функції ініціалізації індикатора
 DDRC=0xFF;                        // налаштування порту С на вихід
 CS0();                            // вибір першого кристалу
 out_text("THIS IS A ",10,0);
// запис текстового рядка; на кожен кристал можна вивести 10 символів.
// Потім необхідно включити другой кристал, і вивести ще 10 символів.
// Всього в одному рядку вміщується 20 символів. Кількість рядків дорівнює 8.
 CS1();                            // вибір другого кристалу
 out_text("FIRST ROW",9,0);        // виведення рядка
 CS0();                            // вибір першого кристалу
 out_text("THIS IS A ",10,2);      // запис текстового рядка
 CS1();                            // вибір другого кристалу
 out_text("THIRD ROW",9,2);        // запис текстового рядка
 CS2();                            // вибір третього кристалу
 out_text("THIS IS A ",10,0);      // запис текстового рядка
 CS3();                            // вибір четвертого кристалу
```

```
out_text("FIFTH ROW",9,0);          // запис текстового рядка
CS2();                              // вибір третього кристалу
out_text("THIS IS A ",10,3);        // запис текстового рядка
CS3();                              // вибір четвертого кристалу
out_text("LAST ROW",8,3);           // запис текстового рядка
PORTC=0;                            // виведення значення 0 в порт C
PORTA=0;                            // виведення значення 0 в порт A
DDRC=0;                             // переведення порта C в третій стан
DDRA=0;                             // переведення порту D в третій стан
while(1);
}
```

Інформація про шрифти зберігається в двовимірному масиві currfont. Для зберігання одного символу в пам'яті мікроконтролера потрібно 6 байт. Масив організований таким чином, що для доступу до графічних даних певного символу в якості першого індексу виступає ascii-код цього символу. Функція init_lcd() виконує ініціалізацію РКІ. У цій функції в кожен з 4-х контролерів передаються команди включення дисплея, установки сторінки пам'яті й положення курсора в межах сторінки. Функція clear_lcd() виконує очищення екрану шляхом запису значення 0 у всі комірки пам'яті. Функція out_char() виконує виведення одного символу. У цій функції, в циклі, виводяться в порт значення елементу масиву currfont, які відповідають ascii-коду символу, що передається у функцію як параметр.

В результаті виконання програми на екрані рідкокристалічного модуля з'явиться 4 рядки, як показано на рис. 4.44.

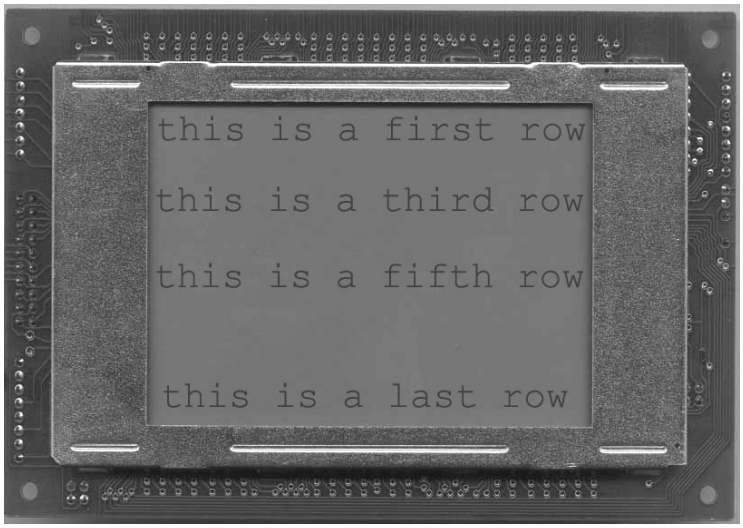


Рис. 4.44.

4.6. Управління індикаторами з інтерфейсом RS232

Компанія Milford випускає рідкокристалічні модулі, які управляються по інтерфейсу RS232. Використання послідовного інтерфейсу дозволяє зменшити кількість необхідних виводів мікроконтролера. Фактично, для роботи рідкокристалічного модуля потрібна лінія живлення, загальний провідник і одна інформаційна лінія.

На рис.4.45 представлена схема сполучення рідкокристалічного індикатора й мікроконтролера з використанням послідовного інтерфейсу.

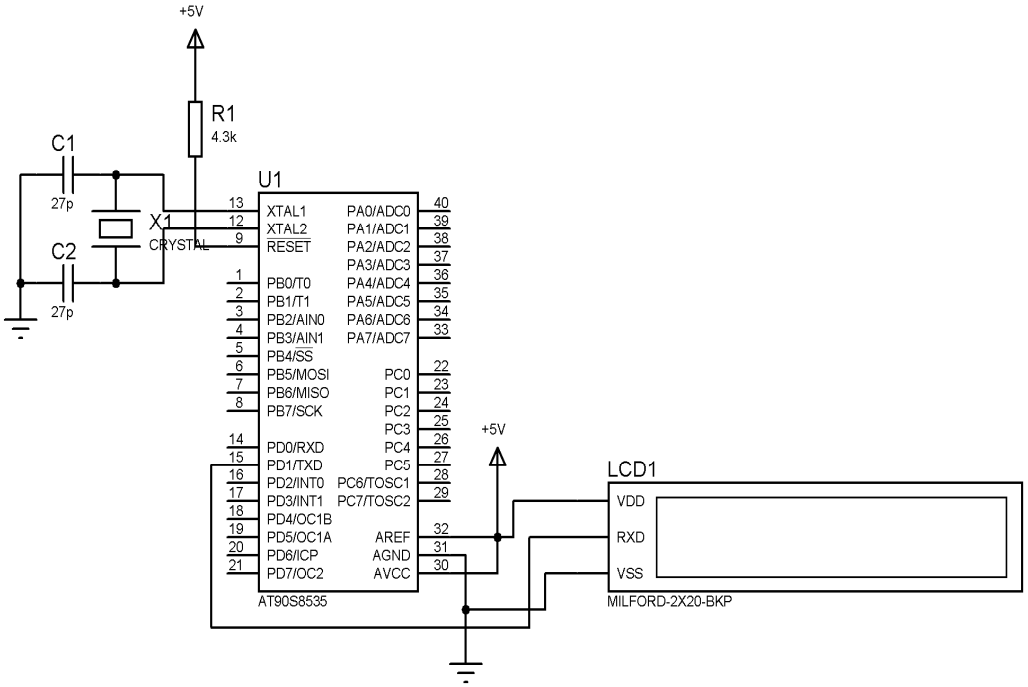


Рис. 4.45.

Приклад 4.13. Написати програму для мікроконтролера, яка у верхньому рядку РКІ виводить статичний текст, а в нижньому рядку – значення лічильника. Управління РКІ виконується по послідовному інтерфейсу.

Текст програми для мікроконтролера представлено нижче.

```
#include <io8535v.h>                    // підключення заголовочних файлів
#include <macros.h>
#include <string.h>
void delay(int ms)                      // функція часової затримки
{
 int c1, c2;                            // додаткові змінні
 for(c1=0; c1<ms; c1++)
  for(c2=0; c2<970; c2++) NOP();        // у циклі виконується «пуста» операція
}
void init_cpu()                         // функція ініціалізації мікроконтролера
{
        UCR=0b00011000;                 // налаштування UART
        UBRR=51;                        // швидкість передачі 9600 біт/с
}
```

```
void main()                                  // головна функція програми
{
char t;
int counter=0;
char text[]="LCD Display MILFORD";           // рядок для виведення на екран РКІ
char text1[]="Counter value: ";
init_cpu();                                  // виклик функції ініціалізації РКІ
delay(500);                                  // затримка 500 мс
while(1)                                     // вічний цикл
{
UDR=0xFE;                                    // префікс команди: передаються команди
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
UDR=0x0D;                                    // команда відображення курсора
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
for (t=0;t<strlen(text);t++)                 // у циклі передається по буквам рядок
{
UDR=text[t];                                 // передача елемента масиву
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
}
for(counter=0; counter<10000; counter++)     // у циклі лічільник інкрементується
{
UDR=0xFE;                                    // префікс команди: передаються команди
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
UDR=0xC0;                                    // переведення курсора на 2 рядок
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
for (t=0;t<strlen(text1);t++)
{
UDR=text1[t];                                // виведення тексту
while(!(USR&(1<<UDRE)));                     // очікування завершення передачі
}
 UDR=counter/1000+48;                        // виведення тисяч
 while(!(USR&(1<<UDRE)));                    // очікування завершення передачі
 UDR=(counter%1000)/100+48;                  // виведення сотень
 while(!(USR&(1<<UDRE)));                    // очікування завершення передачі
 UDR=((counter%1000)%100)/10+48;             // виведення десятків
 while(!(USR&(1<<UDRE)));                    // очікування завершення передачі
 UDR=((counter%1000)%100)%10+48;             // виведення одиниць
 while(!(USR&(1<<UDRE)));                    // очікування завершення передачі
 delay(500);                                 // пауза 500 мс
}}}
```

У функції init_cpu() виконується налаштування UART мікроконтролера. Швидкість обміну даними дорівнює 9600 біт/с. Для виведення статичного тексту в РКІ відправляються ASCII-коді символів, що виводяться. Після цього передається команда переведення курсора на другий рядок 0xC0. Для того, щоб вказати, що наступним байтом є команда, в модуль необхідно відправити префікс команди – число 0xFE. Для виведення числа на екран РКІ його необхідно заздалегідь перетворити в символи.

На рис. 4.46 показаний результат виконання програми.



Рис.4.46.

4.7. Моделювання роботи РКІ

Існують програми, за допомогою яких можна досліджувати алгоритми управління алфавітно-цифровими рідкокристалічними індикаторами. Однією з програм, яка створена для модлювання роботи дворядкового РКІ, є ***DjLCDSIM.*** Цей стимулятор можна знайти в мережі Internet за адресою: www.geocities.com. Стимулятор створений у вигляді HTML-сторінки, вигляд якої представлений на рис. 4.47.

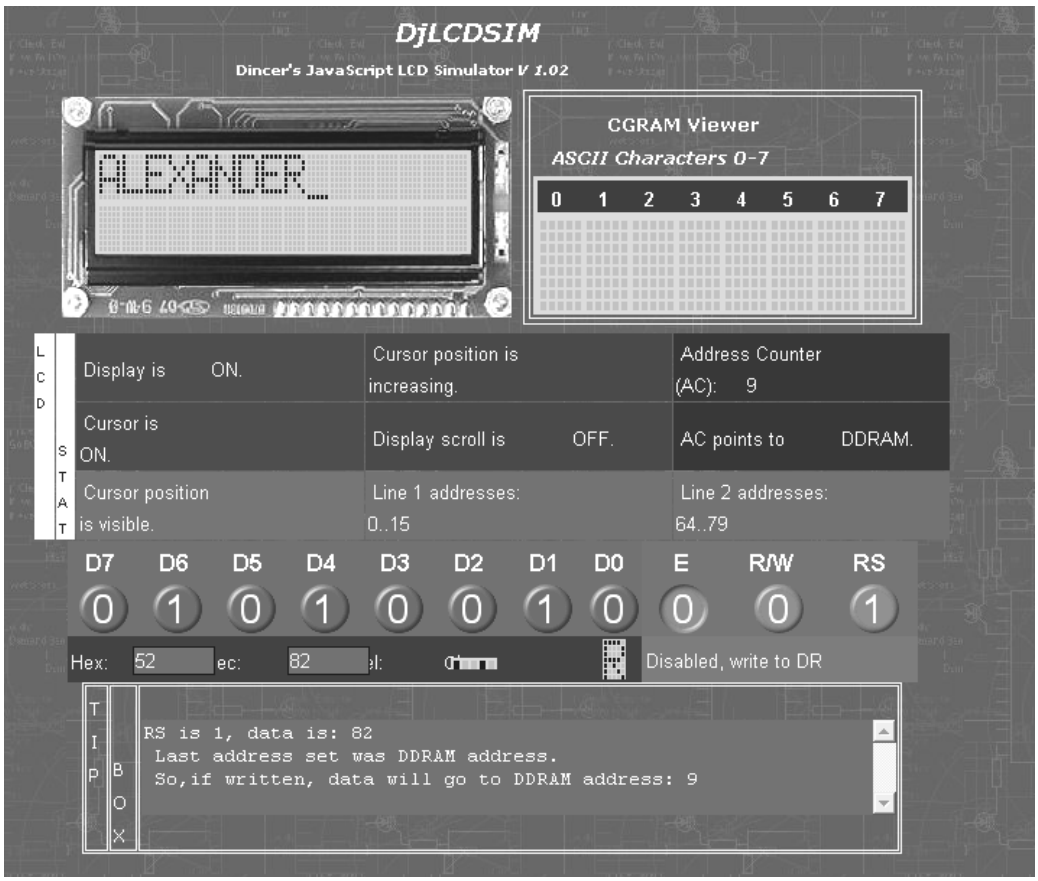


Рис. 4.47.

За допомогою цього симулятора можна вивчати різні режими роботи модуля. У нижній частині розташовані кнопки, за допомогою яких встановлюється рівень сигналів управління (E, R/W, RS), і шина даних (D0-D7). У текстовому полі з'являються підказки про команду, яка

встановлена на шині. Також можна побачити значення біт регістра статусу: включений або виключений дисплей, значення лічильника адреси, режим відображення курсора і напрям його зсуву, чи включена прокрутка дисплея та інші. Також відображається вміст пам'яті CGRAM.

На рис. 4.48 представлено вигляд симулятора графічних РКІ.

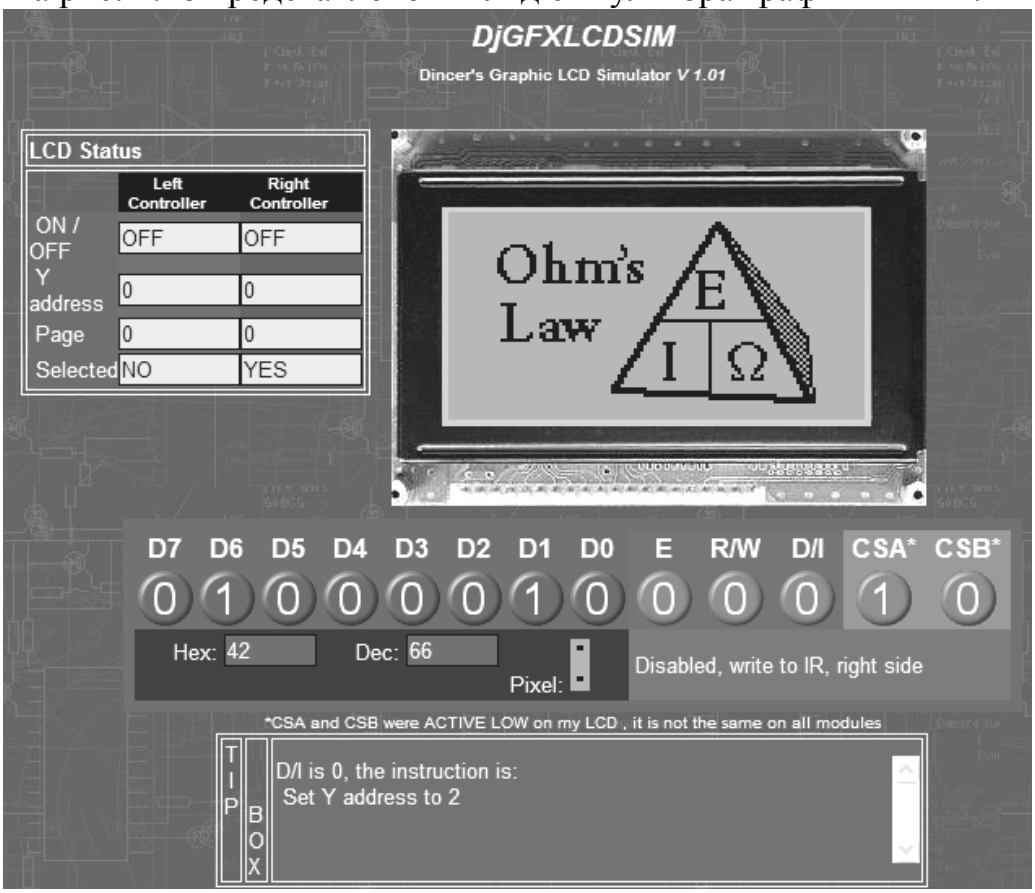


Рис. 4.48.

За допомогою цього симулятора можна вивчати різні режими роботи графічного модуля. У нижній частині розташовані кнопки, за допомогою яких встановлюється рівень сигналів управління (E, R/W, RS, CSA, CSB) і шина даних (D0-D7). Ці кнопки дублюють відповідні виводи безпосередньо на самому індикаторі. У текстовому полі з'являються підказки про команду, яка встановлена на шині. Також можна побачити, яка половинка індикатора вибрана, поточне значення сторінки пам'яті і позицію курсора по горизонталі в межах сторінки. Послідовно формуючи команди й дані на шині даних, і записуючи їх в індикатор, на стилізованому зображенні індикатора можна побачити результат виконання переданих команд.

Загалом, використання подібного симулятора дозволяє значно прискорити налаштуваня програми, зрозуміти принципи управління РКІ, а також досліджувати роботу індикатора без необхідності програмування контролера.

4.8. Формування кольорового зображення

Для виробництва кольорових дисплеїв в даний час широко використовуються РКІ на основі TFT (тонкоплівкові транзистори). В основі структури TFT-панелі містяться рідкі кристали, два поляризатори і дві скляні пластини: верхня підкладка кольорового фільтру і нижня підкладка масиву TFT. Рідкокристалічна речовина розташовується між цими скляними пластинами. Регулювання світлового потоку здійснюється шляхом зміни величини вхідної напруги, що подається на РК. Тим самим змінюється розташування і орієнтація РК-молекул, що призводить до відповідної зміни величини світлового потоку, що проходить через них. При виготовленні такої панелі за допомогою високоточних технологій фотолітографії, на скляну підкладку наноситься візерунок для послідовного покрокового перенесення зображень множини електродів РКІ (рис. 4.49). Кількість транзисторів на склі TFT дорівнює кількості підпікселів дисплея, при цьому генерацію кольору забезпечує скло кольорового фільтру. Рух рідких кристалів викликається появою різниці потенціалів між електродами, що знаходяться на склі TFT і склі кольорового фільтру, і саме цей рух приводить до генерації кольору і зміни яскравості РКІ.

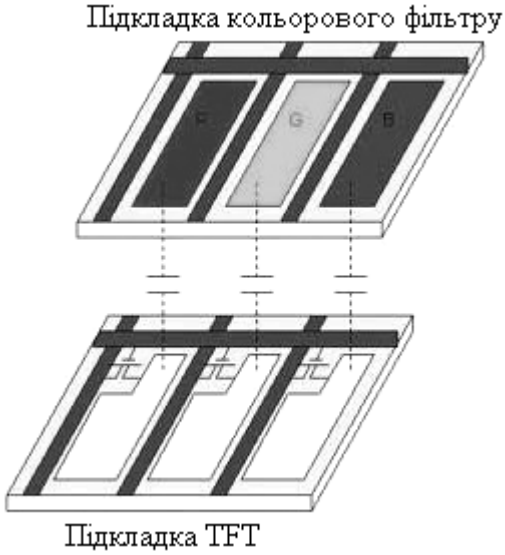


Рис. 4.49.

На рис. 4.50 показаний принцип управління коміркою РК-матриці. В межах одного вибраного періоду часу перемикач замикається, і на РК подається вхідна напруга, що призводить до зміни орієнтації рідкокристалічних молекул. Після вимкнення перемикача в ємності C_{lc} (еквівалентна ємкість РК-речовіни) зберігається деякий заряд, що зменшується з часом. Для збільшення тривалості зберігання заряду паралельно C_{lc} додається запам'ятовуючий конденсатор C_{st}. Оскільки фактично управління рідкими кристалами відбувається змінною напругою, для активації РК напруга подається лише при замкнному перемикачі, після

чого він негайно розмикається. У ряді випадків напруга на РК падатиме через втрати. Для запобігання цього і використовується додатковий конденсатор C_{st}, компенсуючий втрати заряду.

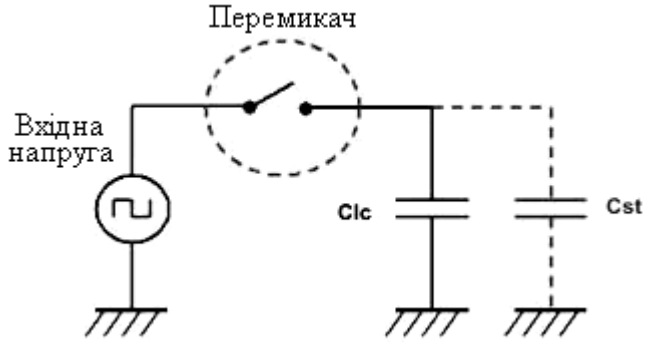


Рис. 4.50.

У TFT-панелі тонкоплівковий транзистор виконує функцію розглянутого перемикача. Вивід затвора TFT підключено до лінії сканування, вивід витоку сполучено з лінією даних, а вивід стоку з C_{lc} і C_{st} (рис. 4.51). Коли затвір активізований (вибраний на лінії сканування), канал TFT відкривається, і дані про зображення записуються в C_{lc} та C_{st}. Якщо затвір не вибраний, TFT закритий.

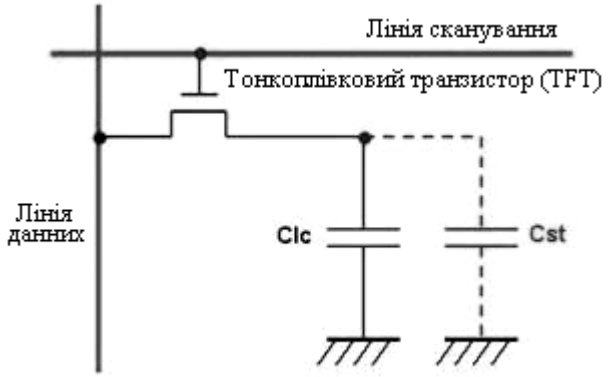


Рис. 4.51.

Формування кольору в TFT-панелі відбувається при проходженні світлового потоку через кольоровий фільтр, інтегрований у верхнє кольорове скло. Кожен окремий піксель зображення формується при змішуванні базових кольорових елементів RGB. Якщо червоний, зелений і блакитний елементи пікселя вибрані в рівній пропорції, буде сформоване біле світло. Шляхом регулювання співвідношення світлопропускання цих трьох елементів отримують необхідну кількість різноманітних кольорів.

4.9. Рідкокристалічні монітори

Екраном LCD-монітора є масив маленьких сегментів (пікселів), якими можна керувати для відображення інформації. LCD-монітор має декілька шарів, де ключову роль відіграють дві панелі, зроблені з вільного

від натрію і дуже чистого скляного матеріалу, які містять тонкий шар рідких кристалів між собою.

Дозволяюча здатність монітора відповідає максимальній фізичній дозволяючій здатності CRT-моніторів, тобто фізичній кількості пікселів в матриці з рідких кристалів. Саме при такій дозволяючій здатності LCD-монітор відтворює зображення найкраще. Ця дозволяюча здатність визначається розміром пікселів, який в LCD-монітора фіксований. Наприклад, якщо LCD-монітор має дозволяючу здатність 1024×768, то це означає, що на кожній з 768 ліній розташовано 1024 електродів (пікселів). При цьому є можливість використовувати і нижчу дозволяючу здатність. Для цього є два способи. Перший називається "Centering" (центрування). Суть методу в тому, що для відображення зображення використовується лише та кількість пікселів, яка необхідна для формування зображення. В результаті зображення виходить не у весь екран, а лише в середині. Всі пікселі, які не використовуються, залишаються чорними, тобто навколо зображення утворюється широка чорна рамка. Другий метод називається "Expansion" (розтягування). Суть його в тому, що при відтворенні зображення з нижчою дозволяючою здатністю використовуються всі пікселі, тобто зображення займає весь екран. Проте, через те, що зображення розтягується на весь екран, виникають невеликі спотворення, і погіршується різкість.

Ще один параметр, який потрібно відмітити, це контрастність. Контрастність LCD-монітора визначається відношенням яскравостей між найяскравішим білим і найтемнішим чорним кольором. Хорошим контрастним співвідношенням вважається 120:1, що забезпечує відтворення живих насичених кольорів. Контрастне співвідношення 300:1 і вище використовується тоді, коли потрібне точне відображення чорно-білих напівтонів.

Суттєвим недоліком РКІ-моніторів є кут огляду: навіть останні моделі часто не дозволяють нормально бачити зображення, якщо користувач дивиться на монітор під кутом. Втім ця проблема починає відходити в минуле: якщо ще недавно типовий кут огляду для рідкокристалічного монітора був близько 45 градусів, то сьогодні є моделі з кутом огляду до 170 градусів.

Варто відзначити і таку особливість частини LCD-моніторів, як можливість повороту самого екрану на 90°, з одночасним автоматичним розворотом зображення. В результаті, наприклад, аркуш формату А4 можна повністю умістити на екрані без необхідності використання вертикального прокручування, щоб побачити весь текст на сторінці.

До переваг LCD-моніторів можна віднести те, що вони дійсно плоскі в буквальному розумінні цього слова, а створюване на їх екранах зображення відрізняється чіткістю і насиченістю кольорів. Відсутність спотворень на екрані, що властиві традиційним CRT-моніторам.

Споживана і розсіювана потужність в LCD-моніторів істотно нижча, ніж в CRT-моніторів.

Ще однією перевагою LCD моніторів є їх малі розміри в глибину: якщо в звичайного монітора глибина приблизно дорівнює діагоналі екрану, то рідкокристалічні монітори досить тонкі, їх товщина складає 5-10 см.

Питання до розділу 4

1. Принцип дії рідкокристалічних індикаторів.
2. Конструктивні модифікації рідкокристалічних індикаторів.
3. Способи керування рідкокристалічними індикаторами.
4. Якими параметрами характеризуються рідкокристалічні індикатори?
5. Порівняйте рідкокристалічні індикатори з іншими типами індикаторів.
6. Назвіть методи керування рідкокристалічними індикаторами.
7. Застосування алфавітно-цифрових та графічних індикаторів.
8. Підключення рідкокристалічних модулів до контролера. Схема.
9. Призначення сигналів керування модуля типу MTG12864.
10. Система команд та програмування графічного модуля MTG12864.
11. Організація пам’яті в рідкокристалічному модулі MTG12864.
12. Призначення сигналів керування модулем HD44780.
13. Система команд та програмування контролера HD44780.
14. Організація пам’яті в контролері HD44780.
15. Опишіть особливості виводу текстової та графічної інформації на рідкокристалічний модуль.
16. Опишіть принцип створення рухомого зображення на рідкокристалічний модулі.
17. Системи команд рідкокристалічний модулів. Структура команд.

Задачі до розділу 4

1. Написати програму, що створює ефект «біжучий рядок». Текст відображається у верхній частині екрану і рухається справа наліво. У нижній частині екрану відображається статичний текст.
2. Написати функцію, яка виводить заданий текст у вказану позицію екрану.
3. Написати функцію, яка змінює режим відображення курсора.
4. Написати програму, яка виводить статичний текст по 4-х бітній шині.
5. Розробити меню. Навігація по меню здійснюється за допомогою тактових кнопок.
6. Написати програму – годинник. В якості початкового значення часу прийняти поточний час. Час має бути представлено у форматі: ГГ: ХХ: СС: МС і виводитися у верхньому рядку екрану.

7. Написати функцію, яка перетворює число типу int в рядок символів (за власним алгоритмом). Порівняти швидкість виконання стандартної функції з бібліотеки (itoa (...)) з розробленою.
8. Створити власні 8 символів і вивести їх в нижній частині екрану.
9. Розробити вольтметр. У верхньому рядку РКІ відображати напругу на першому каналі АЦП, в нижньому рядку відображати значення напруги на другому каналі АЦП. Змінювати рівень напруги можна за допомогою змінних резисторів.
10. Розробити український або російський шрифт для графічного РКІ. Вивести інформацію про себе з використанням розробленого шрифту.
11. Написати функцію, яка інвертує зображення на екрані графічного РКІ.
12. Написати функцію, яка очищає зазначену область екрану. Параметри функції – координати верхнього лівого і нижнього правого кутів.
13. Написати програму що виводить на екран РКІ вашу фотографію.

5. ПРИНЦИПИ ПОБУДОВИ ПРОЕКЦІЙНИХ СИСТЕМ

Сучасні проекційні телевізори за принципом їх дії можна розділити на дві великі групи: з фронтальною проекцією і з рір-прекцією (rear-projection). За типом джерела світлового випромінювання вони поділяються на моделі з проекційними кінескопами і проектори з РК-матрицею. Обидва типи проекторів мають свої переваги і недоліки. Моделі з фронтальною проекцією можуть забезпечувати розгортку зображення на екрани великих розмірів, але мають обмежену яскравість і складні в розгортанні і налаштуванні (зокрема, може бути потрібне проведення додаткових операцій по зведенню на екрані зображень растрів первинних кольорів). Для того, щоб збільшити яскравість зображення, використовуються спеціальні екрани з напрямленим віддзеркаленням світла. При цьому відбувається концентрація (підсилення) світіння екрану при його спостереженні з певного напряму, проте кут оптимального перегляду зображення обмежений.

В даний час досить широко поширені РК-проектори фронтального типу, які є одними з найпоширеніших. Вони використовуються при проведенні конференцій і презентацій, оскільки легко сполучаються з персональними комп'ютерами (у тому числі і ноутбуками), і забезпечують високоякісне відображення комп'ютерної інформації на екранах розміром 1-2 м. Сьогодні вони практично повністю витіснили традиційні діапроектори.

5.1. Джерела світла в проекційних системах

У сучасних проекторах для отримання інтенсивного потоку світла використовують ефективні галогенні, метало-галідні або ксенонові дугові лампи. Галогенні лампи використовуються в проекторах невеликої потужності і мають термін служби 50 – 100 годин (через спад яскравості більш ніж на 50% за рахунок запилення внутрішньої поверхні скла лампи). Метало-галідні лампи використовуються в проекторах середньої і високої потужності. Характерний термін служби ламп складає 1000 – 2000 годин. За 200 годин яскравість падає на 5%, за 1000 – на 20%. У найбільш потужних проекторах використовуються ксенонові дугові лампи, що мають ресурс більше 1000 годин, і дають найбільш природний колір.

Для формування світлового потоку використовується параболічний рефлектор і коліматор (конденсор). У результаті на модулятор спрямовується паралельний пучок світла (рис. 5.1).

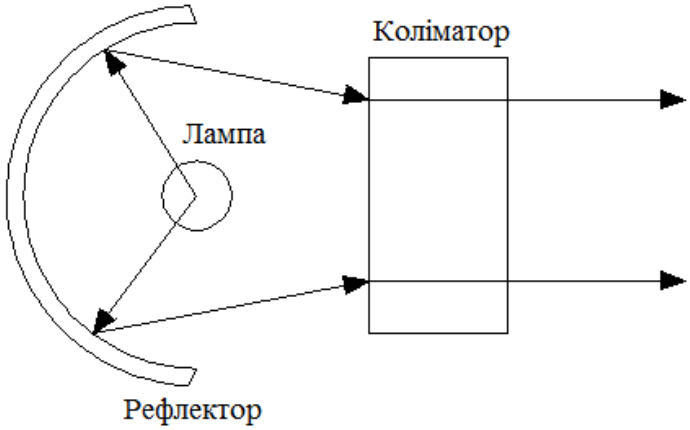


Рис. 5.1.

5.2. Основні типи модуляторів

Модулятор визначає найважливіші параметри проектора, такі як дозволяюча здатність, число градацій яскравості, швидкодія та інші.

5.2.1. Кінескопні проектори

На рис. 5.2 приведена базова схема побудови телевізора (дисплея) із зворотною проекцією на основі проекційних кінескопів. Джерелом зображення в цих телевізорах і дисплеях служать три проекційні кінескопи (трубки) первинних кольорів з розміром екрану від 3 до 7 дюймів.

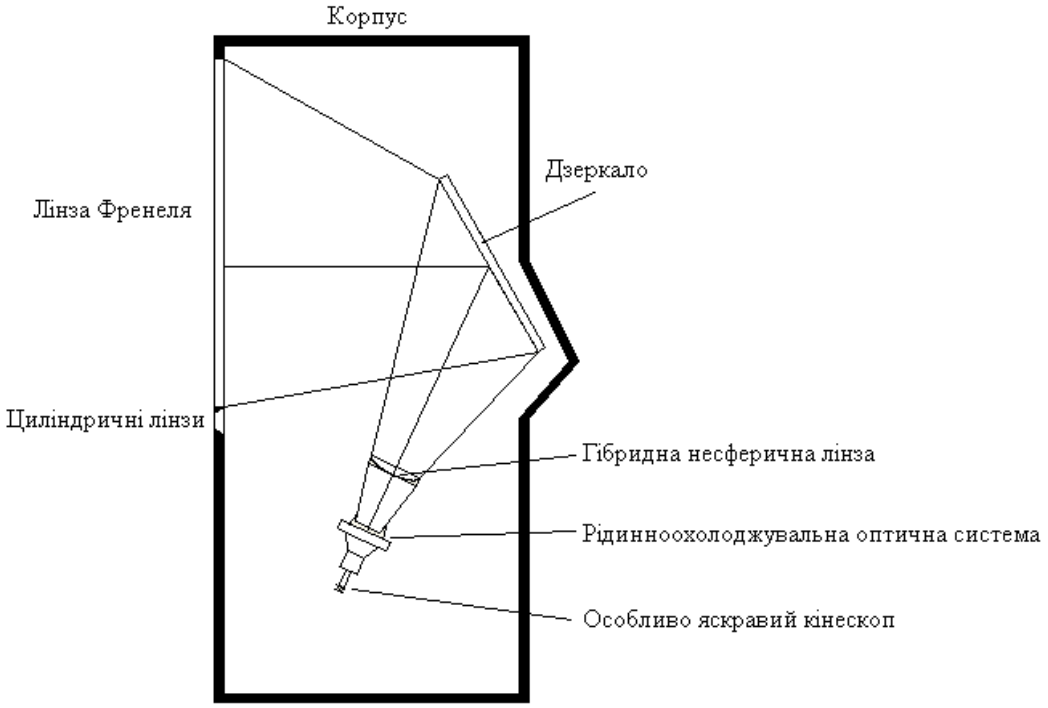


Рис. 5.2.

Оскільки в проекційних монохромних кінескопах первинних кольорів немає тіньової маски і використовуються дуже високі значення

анодної напруги (30-35 кВ), вдається забезпечити дуже високу яскравість їх світіння (понад 10 тис. Кд/м^2).

Для забезпечення високої чіткості зображення на екрані телевізора і зменшення фокусної відстані проекційного об'єктиву використовується спеціальна асферична оптика в об'єктиві. Для здешевлення проекторів застосовуються пластикові лінзи, і лише в дорогих моделях – скляні. Для зменшення розмірів корпусу проектора використовується система відбивних дзеркал. Відбиті від дзеркала світлові промені проектуються на екран проектора, на якому і відтворюється телевізійне зображення.

Конструкція сучасних проекційних кінескопів досить складна. Вона не відрізняється від звичайного чорно-білого кінескопа, за винятком типу люмінофора, що забезпечує світіння одним з основних кольорів (відповідно в проектор встановлюються кінескопи червоного, зеленого і синього кольору). Для стиковки з проекційним об'єктивом передня поверхня проекційного кінескопа виконана плоскою. Проекційний об'єктив з асферичною оптикою герметично кріпиться до корпусу проекційного кінескопа і заповнюється спеціальною охолоджувальною рідиною. Рідинна лінза, що утворюється при цьому, є частиною проекційного об'єктиву, забезпечуючи при цьому ефективне охолодження поверхні кінескопа.

5.2.2. Проектори з рідкокристалічним модулятором

Найбільш поширеним модулятором для проекторів є рідкокристалічна матриця (РКМ). Принцип дії РКМ полягає в модуляції світла, що проходить через рідкі кристали (РК), за рахунок зміни орієнтації або інших властивостей РК. На рис. 5.3 схематично показаний принцип дії РК-комірки і структура РК матриці.

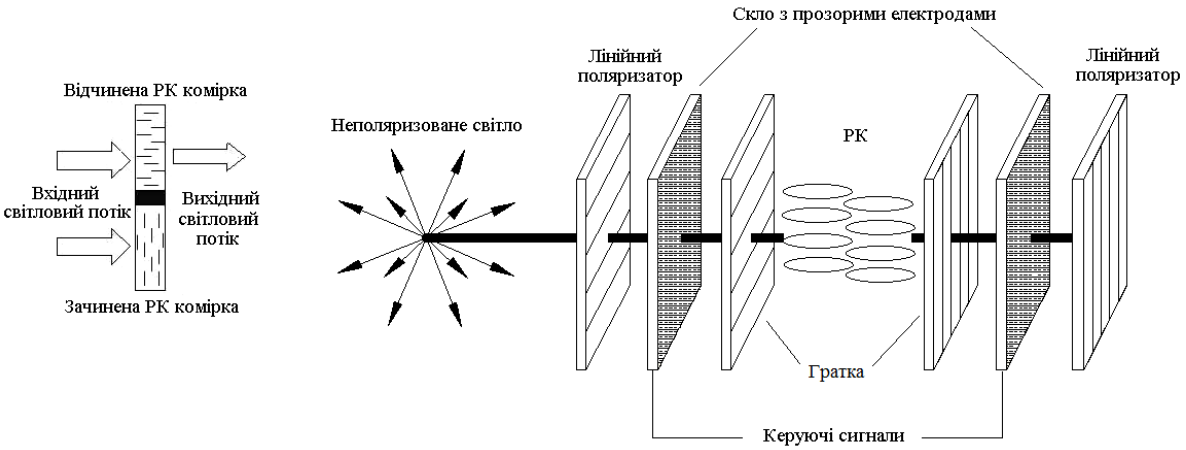


Рис. 5.3.

Характерні розміри РК матриці для проекційних систем складають від 20 до 150 мм по діагоналі. Інформаційна ємність лежить в межах від 640×480 до 1280×1024 і більше елементів зображення. Існують два різновиди РК матриць: на базі аморфного і на базі полікристалічного

кремнію. Для підвищення швидкодії і поліпшення електричних характеристик для управління РК-матрицями використовують тонкоплівкові транзистори (TFT – Thin Film Transistor).

5.2.3. Проектори на базі DMD (Digital Mirror Device)-матриць

Останнім часом з'явилося декілька нових типів модуляторів. Так, на зміну РК-матрицям з фотоелектронним шаром з'явилися розроблені фірмами JVC і Hughes D-ILA матриці (Direct Drive Image Light Amplifier – електрично керований підсилювач яскравості зображення). D-ILA матриця працює на віддзеркалення, для чого в ній використовуються дзеркальні катоди.

Також з'явився цілий ряд MEMS (Micro Electromechanical System – мікроелектромеханічні системи) матриць, заснованих на електрично керованій зміні кута нахилу, або формі мікродзеркал. Такі системи мають дуже високу швидкодію (час перемикання мікродзеркала близько 20 наносекунд).

Найбільшого поширення набули DMD (Digital Mirror Device – мікродзеркальний пристрій) матриці, розроблені фірмою Texas Instrument. Розмір кожного дзеркала 16×16 мікрон, інтервал 1 мікрон (рис. 5.4).

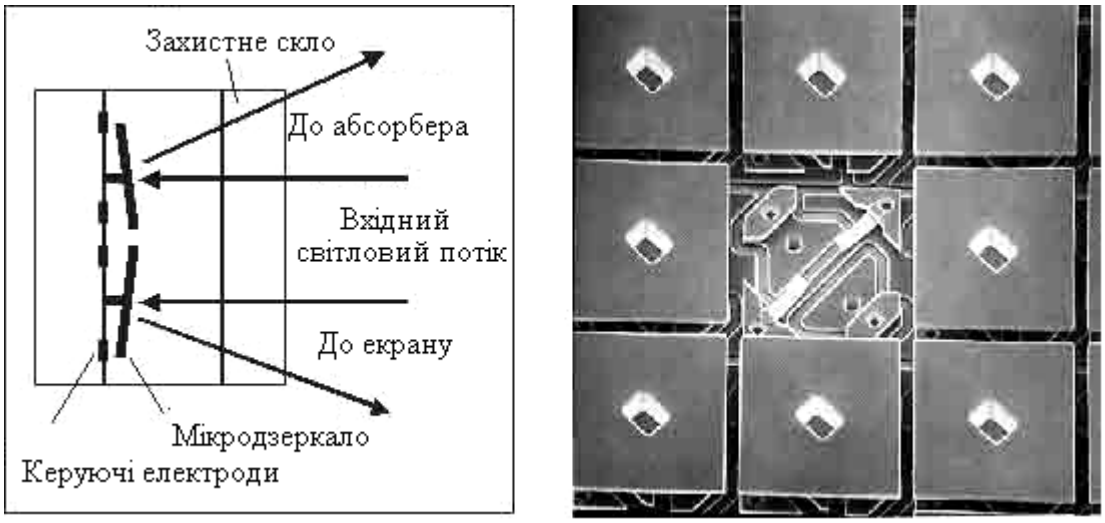


Рис. 5.4.

DMD матриці мають дуже великий термін служби, не менше 30 років безперервної роботи. Проектори на базі DMD матриць називаються DLP (Digital Light Processing – цифрова обробка світла) проекторами.

Ще одним різновидом MEMS є TMA (Thin film Micromirror Array – масив тонкоплівкових мікродзеркал) матриці, розроблені фірмою Daewoo. TMA відрізняються від DMD тим, що в них відхилення дзеркал відбувається не електростатичним, а п'єзоелектричним способом. Характерний розмір мікродзеркала – 97×97 мікрон, проміжок 3 мікрони.

Перевагою TMA матриць є можливість здобуття напівтонів в межах одного звернення до комірки.

До іншого типу MEMS пристроїв відносяться GLV матриці (Grating Light Valve – мікромеханічна фазова гратка). На рис. 5.5 показаний зовнішній вигляд і принцип дії такої матриці.

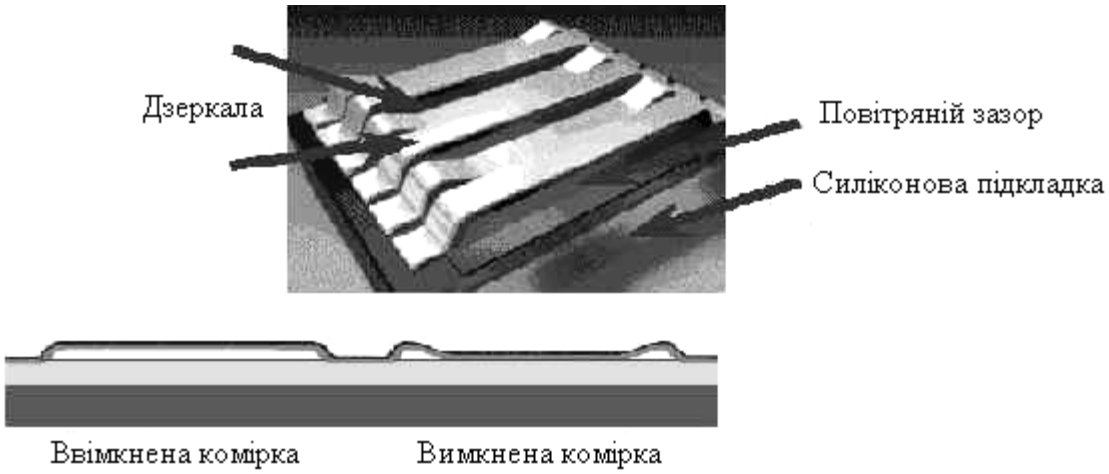


Рис. 5.5.

5.2.4. Лазерні проектори

На рис. 5.6 показана схема пристрою відображення інформації з використанням лазера.

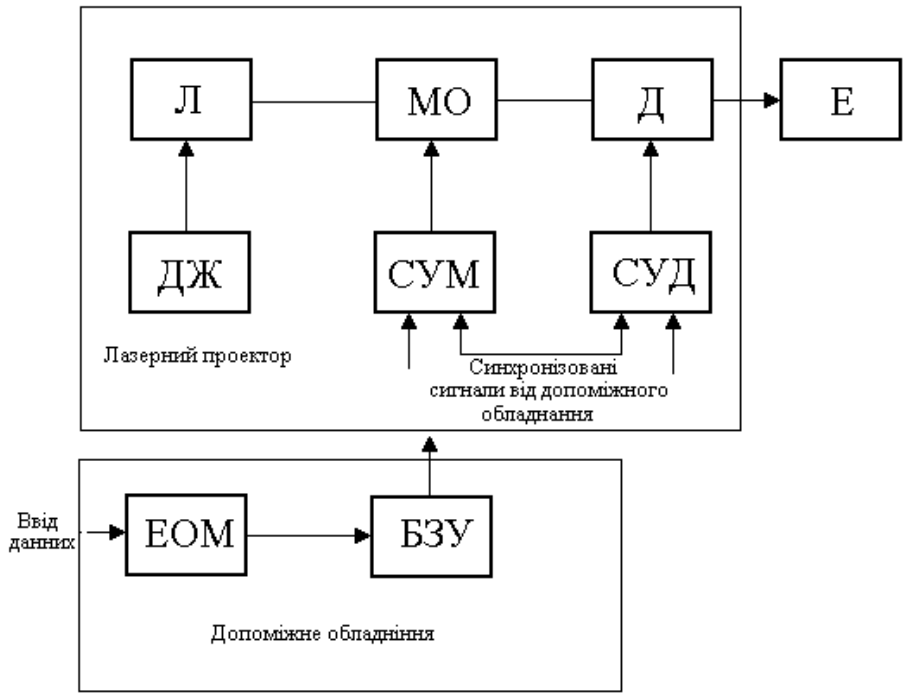


Рис. 5.6.

Лазер Л, оптичний модулятор МО, дефлектор Д, схема управління модулятором СУМ, схема управління дефлектором СУД і джерело

живлення ДЖ утворюють лазерний проектор. Відображається інформація на екрані Е. Допоміжне устаткування, в яке входить ЕОМ і буферно-перетворюючий запам'ятовуючий пристрій БЗУ, лазерний проектор і екран забезпечують управління процесом відображення інформації, а також довгострокове і короткострокове її зберігання.

При розробці пристроїв відображення на лазерах використовуються наступні методи: візуальна лазерна індикація, при якій на екран спрямовується власне світло лазера; індикація з активним екраном, коли промінь лазера застосовується лише для управління світловим випромінюванням деякого активного матеріалу екрана; лазерно-променевий світловий клапан – промінь лазера забезпечує місцеве управління оптичними параметрами деякого матеріалу (його коефіцієнтом віддзеркалення, або коефіцієнтом пропускання), а окреме джерело звичайного типу дає світло для проекції на екран; лазерний генератор зображення з безпосередньою дією на об'ємний резонатор.

При відображенні інформації використовують спосіб “послідовної видачі”, при якому промінь лазера послідовно обходить всі точки поверхні екрану, а також спосіб “вибіркового відображення”, при якому промінь лазера спрямовується лише на ті елементи екрану, в які вводиться інформація.

Модулятор світла призначений для накладання інформації, що змінюється в часі, на випромінювання лазера шляхом зміни в часі його яскравості. Якщо зміни інформації синхронізовані з переміщенням променя дефлектора, то інформація перетворюється на зображення.

До основних характеристик модулятора відносять ширину смуги частот, характеристики світлопропускання і відтворення напівтонів, контрастні характеристики, розсіювану потужність, лінійність і вимоги до модулюючого сигналу. Потрібна смуга частот модулятора залежить від необхідної якості зображення і способу відображення.

Дефлектори, що здійснюють управління променем, засновані на різних способах відхилення променя: механічному, рефракційному, дифракційному, двійкового електрооптичного управління положенням променя та ін.

Механічний спосіб реалізується за допомогою застосування двох багатогранних призм, що обертаються, або дзеркала з досить високим коефіцієнтом віддзеркалення, яке переміщується по горизонталі і вертикалі п'єзоелектричним і гальванометричним приводами. Спосіб забезпечує відносно великі робочі кути відхилення і досить високий оптичний коефіцієнт корисної дії. Швидкодія таких пристроїв мала, тому їх можна використовувати лише при режимі послідовної видачі. Крім того, їм властиві нестабільність, жорсткі допуски на елементи, труднощі синхронізації.

Рефракційний спосіб реалізує відому оптичну властивість – відхилення світлового променя внаслідок заломлення (рефракції) на

границі двох прозорих середовищ. В цьому випадку застосовують електрооптичну призму або ультразвукову комірку рефракції.

Дифракційний спосіб може бути використаний, якщо діаметр падаючого світлового пучка істотно більше довжини ультразвукової хвилі, коли виникає дифракція світла (при растровій розгортці). Він забезпечує малі робочі кути (до декількох градусів) і низьку ефективність відхилення.

Спосіб когерентної оптичної фазової гратки заснований на властивості випромінювання лазера, що характеризується високою мірою часової і просторової когерентності. Ця властивість використовується для відхилення лазерного променя за рахунок розділення його на безліч паралельних променів і зміни відносних фаз між сусідніми променями в ближній зоні поля. Цей спосіб вимагає високої стабільності, як джерела світла, так і дефлектора, і має ряд інших обмежень.

Спосіб двійкового електрооптичного управління світловим променем заснований на використанні властивості подвійного заломлення променя в деяких речовинах. У таких речовинах звичайний неполяризований промінь світла розщіплюється на два промені. Один з променів називається звичайним, а інший – незвичайним. Ці промені лінійно поляризовані, причому площина їх поляризації взаємно ортогональна. Якщо світло, що падає на речовину з подвійним променезаломленням (по нормалі), повністю лінійно поляризоване і його площина поляризації збігається з площиною поляризації звичайного променя, то світло проходить не відхиляючись. Якщо падаюче світло лінійно поляризоване в площині незвичайного променя, вихідний промінь стає зміщеним відносно точки виходу звичайного променя. Величина такого зсуву пропорційна товщині кристалу з подвійним променезаломленням.

Технологія LDT (Laser Display Technologie), або лазерна технологія, виникла не так давно, а серійне виробництво освоєне лише в 2000 році. Виробництвом таких проекторів займається німецька компанія Laser Technologies AG. Для створення зображення використовується лазер. Три промені – зеленого, синього і червоного кольору, модулюються по амплітуді відповідно до відеосигналу, що подається. Потім, за допомогою спеціальної системи напівпрозорих дзеркал, три складові змішуються в один потік. Цей промінь подається по оптоволоконному кабелю на проекційний пристрій, який включає систему фокусування і оптомеханічну систему розгортки. На екрані зображення створюється по рядках: по вертикалі променем управляє дзеркало, а по горизонталі – колесо з 25 дзеркалами. Промінь, рухаючись зверху донизу, встигає промальовувати рядки зліва направо. Оскільки промінь встигає за секунду повністю перемальовувати екран 50 разів, око сприймає зображення як єдине ціле і не помічає мерехтіння. На рис. 5.7 представлена схема, що пояснює принцип роботи лазерного проектора.

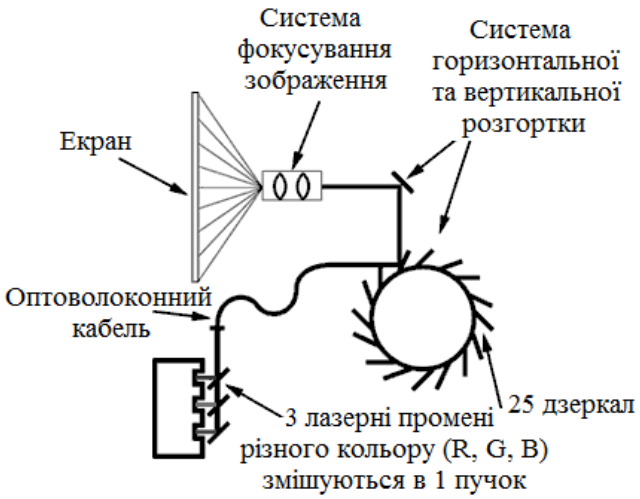


Рис. 5.7.

За допомогою лазерного проектора можна проектувати зображення на поверхню площею декілька сотень квадратних метрів, причому це не обов'язково має бути звичний екран – це можуть бути стіни будівель або якісь інші криві поверхні. Лазерний промінь в будь-якій точці створює різке, насичене і при цьому яскраве і контрастне зображення. Хоча для забарвлення кожного з променів застосовуються спеціальні кристали, що змінюють довжину хвилі і, відповідно, колір, досягти виключно правильного втілення кольорів непросто. Тривалість служби таких проекторів можна зв'язати з тривалістю "життя" лазера, а вона в 3-5 разів перевищує показники ламп в інших видах проекторів через кращий в порівнянні з ними ККД. Виробляються лазерні проектори у достатньо обмежених кількостях, а їх вартість досягає декількох десятків тисяч доларів. Застосування проекторів, заснованих на лазерній технології, може бути виправдане при організації крупних світлових шоу, проектування комп'ютерної графіки, космічного моделювання, в центрах управління, тренажерах, системах віртуальної реальності, крупних конференціях.

5.3. Формування кольорового зображення

Для отримання кольорових зображень може використовуватися один з декількох способів:

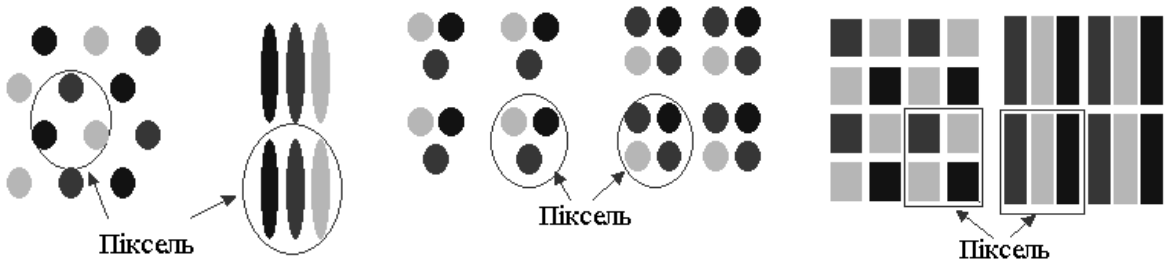
– з розподіленням кольорів у просторі;

– з розподіленням кольорів у часі;

– із суміщенням кольорів у часі і просторі.

Розподіл кольорів у просторі найбільш поширений спосіб отримання кольорів. В цьому випадку елемент зображення (піксель) складається з декількох елементів різних кольорів (як правило, червоний, зелений і синій). При спостереженні з певної дистанції ці елементи зливаються, і їх можна спостерігати як єдиний піксель довільного кольору. При

просторовому розділенні кольорів ефективна яскравість складає від 20 до 30% від максимальної. На рис. 5.8 показані типові структури пікселів (а – ЕПТ піксель; б – світлодіодний піксель; в – РК піксель).



а *б* *в*

Рис. 5.8.

За типом формування світлового потоку трьох основних кольорів РК-проекційні телевізори можна розділити на дві великі групи: проектори з однією потрійною РК-панеллю і проектори з трьома простими РК-панелями. У проекторах першого типу в якості управляючого світловим потоком елементу використовується одна кольорова РК-панель з трьома типами кольорових комірок – RGB.

На рис. 5.9 показана структура РК-матриці з просторовим розподіленням кольорів.

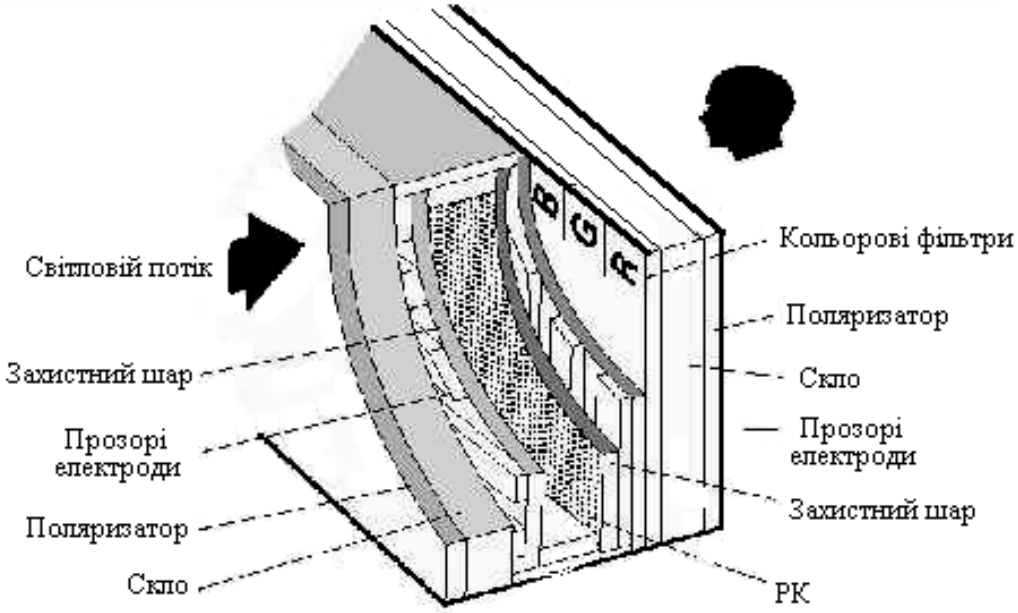


Рис. 5.9.

Принцип роботи такого проектора достатньо простий. Світловий потік від джерела світла (ксенонова або галогенна лампа) потрапляє на поверхню модулятора світла кольорової РК-панелі. Проходячи через RGB-фільтри і відповідні комірки РК-панелі, світлові промені первинних

кольорів модулюються за яскравістю, і через лінзу і об'єктив оптичної системи проектуються на екран.

В проекційних системах використовується розподіл кольорів в часі. При цьому матриця-модулятор по черзі освітлюється джерелом світла червоного, зеленого і синього кольорів. Для здобуття різних кольорів найчастіше використовується диск, що обертається, з секторами різного кольору. При такому способі від модулятора вимагається підвищена (у три рази) швидкодія, що досить просто реалізується в MEMS матрицях, і ускладнене в РК матрицях. Часове розділення кольорів знижує максимальну яскравість в три рази. Схема DMD проектора з часовим розподілом кольорів приведена на рис. 5.10.

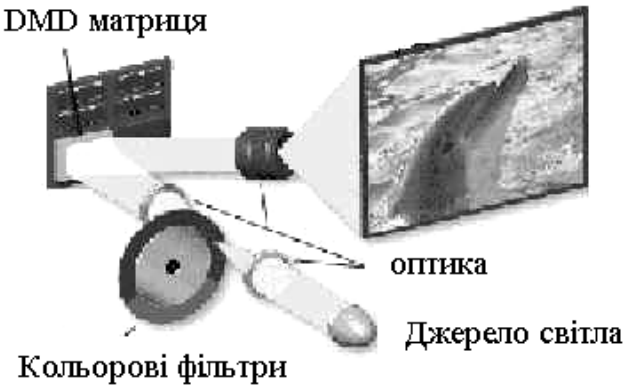


Рис. 5.10.

Поєднання кольорів у часі і просторі є найбільш ефективним, але вимагає використання трьох модуляторів. Застосовується в проекційних системах на базі ЕПТ і в якісних матричних проекційних системах. Цей принцип поєднання дозволяє отримати найбільш високу ефективність, але вимагає якісного поєднання вихідних пучків світла від кожного з трьох модуляторів. Відповідно, системи з використанням цього принципу є найбільш дорогими.

Світловий потік від джерела світла заздалегідь розділяється на три потоки первинних кольорів дихроїчними дзеркалами. У деяких варіантах РК-проекторів цього типу для розділення променів замість дихроїчних дзеркал використовуються дифракційні гратки, що дозволяє дещо збільшити значення світлового потоку проектора за рахунок виключення втрат світлової енергії. На рис. 5.11 показана схема формування кольорів для трьох РК.

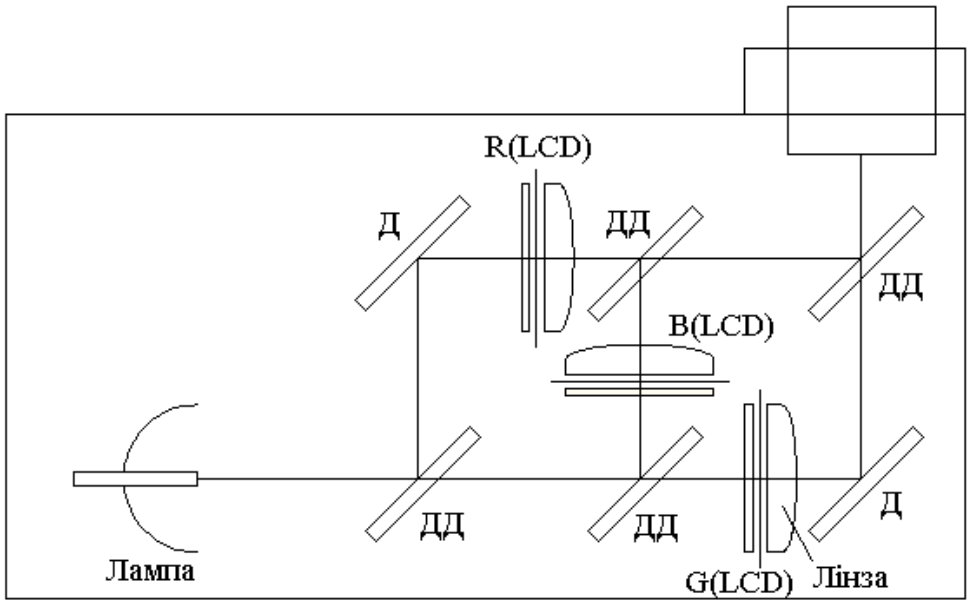


Рис. 5.11.

На малюнку позначені: Д – дзеркало, ДД – дихроїчне дзеркало. Світловий потік від джерела світла за допомогою системи дихроїчних дзеркал розділяється на три кольорові потоки первинних кольорів RGB і прямує на три прості РК-панелі. Світлові потоки первинних кольорів, що проходять через них, модулюються за яскравістю і знову складаються системою дзеркал в один загальний світловий потік, який поступає на вхід проекційного об'єктиву, і потім проектується на екран. Недоліком РК-проекторів такого типу є очевидна складність конструкції.

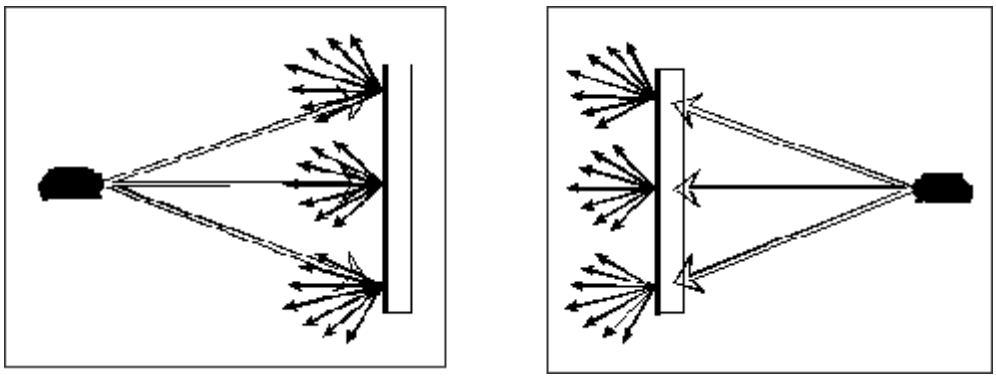
За рахунок практично повного усунення втрат світлової енергії на світлофільтрах (втратами в дихроїчних дзеркалах можна знехтувати) зображення на екрані відрізняється високою яскравістю і насиченістю кольорів. Яскравість в основному визначається світловим потоком джерела світла і досконалістю екрану проектора.

Ще одним досить популярним різновидом РК-проекторів з простими РК-панелями є проектори на основі додаючої (композитної) призми. У ньому для розділення вихідного світлового потоку на 3 потоки первинних кольорів також використовуються дихроїчні дзеркала, а для додавання променів використана призма. Проектори з таким типом оптичної системи вдається зробити досить компактними, оскільки вони дозволяють використовувати короткофокусні об'єктиви з лінзами малого діаметру, що є їх очевидною перевагою. До недоліків даного типу проектора слід віднести високі вимоги до якості виготовлення призми, яка значно дорожча за дихроїчні дзеркала. Крім того, на поверхню РК-панелей наноситься штрихове поглинаюче покриття типу «чорної матриці» для усунення їх засвічення, що також ускладнює виготовлення проектора.

5.4. Оптичні властивості екранів з фронтальною і rear-проекціями

Джерело світла, світломодулюючі матриці, оптика, супутні деталі і вузли утворюють функціонально закінчений пристрій – проектор. Зображення, що формується проектором, може проектуватися на екран певних розмірів і з певними оптичними властивостями. Цілий ряд важливих параметрів зображення, таких, як яскравість, контраст, діаграма спрямованості та інші або визначаються, або істотно залежать від оптичних властивостей екрану.

Основною функцією екрану є розсіювання світлового потоку, що надходить. Це розсіювання може здійснюватися при віддзеркаленні, або при пропусканні світла. Відповідно, існують два типи екранів: для фронтальної проекції, що працюють на віддзеркалення, і для рір-прекції, що працюють на пропускання. Схема роботи цих двох типів екранів показана на рис. 5.12, а, б.



а *б*

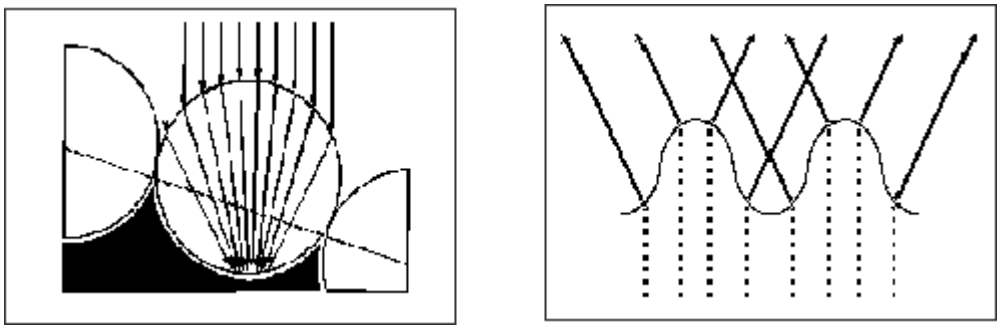
Рис. 5.12.

Розподіл світлового потоку в просторі характеризує діаграма спрямованості екрану. У найпростішому випадку екран має рівномірний розподіл світла в межах 180 градусів (ламбертовське або дифузне розсіювання). При цьому коефіцієнт підсилення екрану, що визначається як відношення світлового потоку на нормалі екрану до середнього світлового потоку, дорівнює одиниці. Поверхня таких екранів покривається спеціальними речовинами з великою дифузією. Використовуючи комбіноване покриття з високодифузних і відбиваючих речовин, можна до певної міри формувати діаграму спрямованості та коефіцієнт посилення.

Для рір-проєкційних екранів використовуються напівпрозорі плівки, скло з нанесеним плівковим покриттям. Формування діаграми спрямованості служить для підвищення яскравості зображення на екрані. Наприклад, якщо спостереження на екран відбувається не у всьому 180-градусному секторі, а лише в 90-градусному секторі по горизонталі і 45-градусному секторі по вертикалі, то використовуючи екран з

горизонтальним підсиленням 2 і вертикальним підсиленням 4, можна підняти яскравість на нормалі екрану у 8 разів.

Простим способом формування заданої діаграми спрямованості є використання вігнутих екранів із заданою геометрією поверхні. Але це не завжди прийнятно. Для формування діаграми спрямованості з різними коефіцієнтами підсилення у вертикальному і горизонтальному напрямах, використовуються екрани з профільованою поверхнею, екрани, покриті мікроскопічними скляними кульками діаметром в декілька десятків мікрон і екрани з циліндричними лінзами. На рис. 5.13 показані поверхні екранів зі сферичними (а) і циліндричними (б) лінзами.



а *б*

Рис. 5.13.

При використанні в екранах методів оптичного формування діаграми спрямованості, можуть бути отримані досить вузькі діаграми з великим коефіцієнтом підсилення. У найдосконаліших екранах додатково до циліндричних лінз застосовуються лінзи Френеля, що дозволяють не лише формувати задані діаграми спрямованості, але і частково корегувати нерівномірність яскравості на межах екрану.

Для екранів фронтальної проекції існують проблеми із зовнішнім контрастом, оскільки і світло від проектора, і сторонні засвічування йдуть з одного і того ж напряму, і немає можливості відрізнити перше від другого. Для того, щоб яскравість зображення була як можна вищою, екран повинен дифузно відбивать якомога більше світла. Для підвищення контрасту необхідно підвищувати корисну освітленість екрану і знижувати сторонні засвічування (затемнення приміщення в цілому або зони розташування екрану).

У екранів рір-прекції світло від проектора і зовнішнє засвічування поступають на екран з різних сторін. Спрощено можна розглядати екран, що працює на просвічування, складеним з дифузного шару, що пропускає корисне зображення, і нейтрально сірого фільтру. Тоді світло від проектора, проходячи через цей фільтр, затухатиме в k разів, а зовнішнє світло пройде через фільтр ослабнувши в k разів і, відбившись від дифузного шару, назад пройде через фільтр та ослабне ще в k разів, всього в k^2 разів. При підвищенні контрасту знижується яскравість зображення.

Через більш високу складність екрани для рір-прекції, як правило, істотно менші за площею, ніж екрани для фронтальної проекції.

Підвищити контраст можна шляхом використання поляризованих джерел освітлення. Для цього потрібно використовувати поляризаційні фільтри як на екрані, так і на джерелах світла, але це достатньо дорого.

5.5. Похибки оптичних систем

У проекторах широко використовуються різні оптичні системи. Між джерелом світла і модулятором відбувається формування світлового потоку заданої геометрії і розподіл кольору. Між модулятором і екраном також використовуються досить складні оптичні системи. Оскільки ці системи маніпулюють вже з модульованим світловим потоком, що несе інформацію, то саме вони в істотній мірі визначають якість зображення. Для того, щоб зрозуміти, яку дію чинить вихідна оптична система на якість зображення, розглянемо похибки оптичних систем.

Астигматизм. Проявляється у викривленні площини зображення. При проекції на плоский екран зображення від центру до периферії стає все більш і більш розмитим. Це пов'язано з тим, що проектоване зображення представляє собою не площину, перпендикулярну до оптичної осі, а деяку вигнуту поверхню. Астигматизм оптичної системи виправляється підбіром конструктивних елементів системи, радіусів поверхонь, показників заломлення і відстаней між елементами оптичної системи.

Дисторсія. Прояв дисторсії полягає в подушкоподібних або бочкоподібних спотвореннях зображення, і пов'язане з тим, що збільшення оптичної системи залежить від кута між падаючим світлом і оптичною віссю системи. Явище дисторсії також може бути мінімізоване застосуванням спеціальних оптичних заходів.

Хроматична аберація. Цей вид спотворень пов'язаний з тим, що показник заломлення скла оптичної системи залежить від довжини світлової хвилі (явище дисперсії), отже, світлові промені з різною довжиною хвиль фокусуватимуться не в одній площині, що призведе до появи на зображенні кольорових кіл. Для вирівнювання фокусних відстаней використовуються спеціальні ахроматичні лінзи, що складаються з комбінації двоопуклих і розсіюючих лінз, виготовлених з різних оптичних матеріалів.

Коефіцієнти пропускання. Навіть при проходженні світла через абсолютно прозоре скло на двох оптичних переходах втрачається близько 8% світлового потоку. Це не дуже багато, але для складних оптичних систем, де число оптичних переходів може досягати декількох десятків, сумарні втрати можуть бути досить великі. Для зниження втрат в оптичних системах використовуються спеціальне тонкоплівкове покриття (просвітлення оптичних поверхонь). Тришарове просвітлення може зменшити втрати з 4 до 0,5% на кожен оптичний перехід.

Нерівномірність яскравості. У реальних оптичних системах яскравість на границях зображення істотно (у 2 – 3 рази) нижча, ніж в центрі зображення. Цей спад яскравості тим більший, чим коротша фокусна відстань оптичної системи. Частково спад яскравості може бути скоригований в самій оптичній системі. Проте радикальнішими способами вирівнювання яскравості є електронна корекція яскравості в модуляторі і використання екранів з особливими оптичними властивостями.

Механічні і теплові похибки. Похибки в юстуванні елементів оптичної системи можуть призводити до появи вищерозглянутих оптичних спотворень, а для проекторів з поєднанням кольорів – до порушення зведення кольорів. Оптична система проектора пропускає через себе досить потужний світловий потік, і тому є термонавантаженою системою. Внаслідок різних коефіцієнтів теплового розширення також можлива поява різного виду оптичних спотворень.

Як правило, для проекторів пропонується цілий ряд об'єктивів з різними фокусними відстанями і різною світлосилою.

Проектування оптичної системи, що усуває оптичні спотворення різного виду, є досить складним і навіть суперечливим завданням. Як правило, доводиться йти на певні компроміси, у результаті жертвуючи якістю зображення. Звідси ж витікає, що сучасні оптичні системи достатньо складні і, отже, дорогі.

5.6. Огляд відеопроекторів різних фірм-виробників

В табл. 5.1 приведені основні показники ряду проекторів.

Табл. 5.1.

Проектор	Тип модулятора	Контраст	Джерело світла (тип лампи)	Світловий потік, Лм	Потужність лампи, Вт
Digital Projection Power 15sx	Три DMD	500:1	Дугова ксенонова	12000	
Electrohome VistaULTRA12K	Три DMD	400:1	Дугова ксенонова	12000	1800
BARCO Elm R12	Три DMD	400:1	Дугова ксенонова	11500	
BARCO Elm G10	Три DMD	400:1	Дугова ксенонова	9500	

NEC MultiSync HiVid 6500	DMD		Дугова ксенонова	6500	1600
BARCOVision 9200	Три РК	340:1	Метало-галідна	5000	1500
Daewoo 5000	ТМА		Дугова ксенонова	5000	1000
Proxima Pro Av9400	Три РК		Метало-галідна	2300	400
Dukane 9000 ImagePro	Три D-ILA	250:1	Дугова ксенонова	1000	400

Питання до розділу 5

1. Які джерела світла використовуються в проекційних системах?
2. Що використовується в якості модуляторів світла?
3. Яке призначення модулятора світла?
4. Структура кінескопного проектора.
5. Яким чином формується кольорове зображення?
6. Які існують похибки оптичних систем?
7. Яким чином можна покращити якість зображення?
8. Що таке піксель?
9. Чи впливають оптичні властивості екрану на якість хображення?
10. Наведіть структурну схему лазерного проектора.
11. Поясніть принцип роботи лазерного проектора.

6. ПРИНЦИПИ ПОБУДОВИ СЕНСОРНИХ ПАНЕЛЕЙ

Використання сенсорних екранів в якості комп'ютерного інтерфейсу стає сьогодні поширеним явищем. Популярність останніх обумовлена простотою інтерфейсу користувача, що досягається з їх допомогою. Інтерфейс на основі сенсорного екрану дозволяє виключити застосування великоформатних клавіатур, а також забезпечує зручну контекстну систему меню, призначену для користувача.

Сьогодні розробникам електронної апаратури доступні як готові сенсорні екрани на основі графічних рідкокристалічних індикаторах (РКІ) і сенсорних панелей зі своїм контролером, так і окремі сенсорні панелі (рис. 6.1).



Рис. 6.1.

Проте, ціна готових сенсорних екранів досить висока, до того ж обмежена номенклатура пристроїв даного типу не завжди дозволяє обрати відповідний по конструкції тип сенсорного екрану. Замовлення на проведення розробки сенсорного екрану із заданими параметрами – дуже дороге задоволення, яке навряд окупиться при малих обсягах випуску. У цій ситуації слід правильно оцінити можливості й вартість сенсорного екрану в порівнянні з конкуруючим стандартним і перевіреним рішенням – інтерфейсом на основі РКІ з фіксованими знакомісцями та функціональною клавіатурою на 4-16 клавіш.

Є ряд виробників, які поставляють стандартні накладні резистивні сенсорні панелі. В цьому випадку для реалізації інтерфейсу сенсорного екрану необхідно розробити свій контролер, який забезпечуватиме підтримку функцій виявлення торкання та визначення координат точки дотику. До сфер застосування сенсорних панелей відносяться:

– легка промисловість;
– електроенергетика;
– телекомунікації;
– системи безпеки;
– медичне устаткування.

6.1. Типи сенсорних панелей

У сенсорних панелях можуть бути використані різні фізичні ефекти, що забезпечують сенсорну чутливість. Домінують переважно дві технології, засновані на резистивній і ємнісній чутливості. Є інші способи, що також використовуються в практичних конструкціях сенсорних панелей, засновані на використанні ІЧ-сканування (матриця ІЧ-світлодіодів і фотоприймачів), ультразвукового сканування й електромагнітних ефектів.

Інфрачервоний сенсорний екран від компанії Atouch (рис. 6.2) має оригінальний дизайн і інтерфейс. Цей сенсорний екран може бути застосований до різних видів моніторів і плазмових панелей.



Рис. 6.2.

Уздовж меж сенсорного екрану, що застосовує в своїй роботі принцип інфрачервоних (ІЧ) хвиль, встановлюються спеціальні випромінюючі елементи, що генерують напрямлені уздовж поверхні екрану світлові хвилі інфрачервоного діапазону, що розподіляються в його робочому просторі на зразок координатної сітки (рис. 6.3). З другого боку екрану змонтовані чутливі до ІЧ світла елементи, які приймають хвилю і перетворюють її в електричний сигнал. Якщо один з інфрачервоних променів перекривається стороннім предметом, який потрапив в зону дії променів, тоді промінь перестає поступати на приймальний елемент, що одразу ж фіксується мікропроцесорним контролером. Таким чином обчислюється координата дотику.

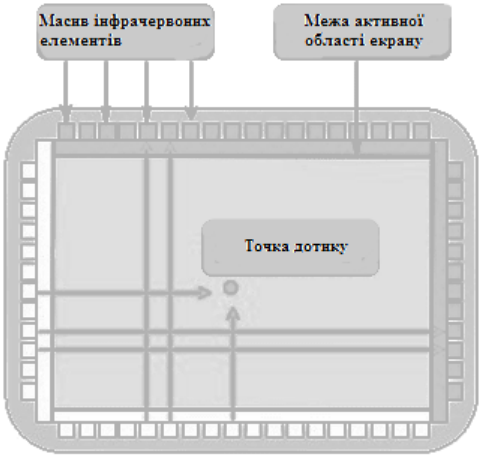


Рис. 6.3.

До переваг інфрачервоних сенсорних панелей можна віднести:

- витримують жорсткі умови;
- налаштовуються на зміну умов освітлення, включаючи пряме сонячне світло;
- стабільні, не втрачають калібровку;
- зносостійкі;
- захищені від забруднюючих речовин;
- можливий вибір інтерфейсу підключення;
- можливий вибір типу фільтра і основи для захисту системи та оптимальної оптики.

Екран на технології поверхнево акустичних хвиль представляє собою скляну панель з п’єзоелектричними перетворювачами (ПЕП), що знаходяться у кутках (рис. 6.4).

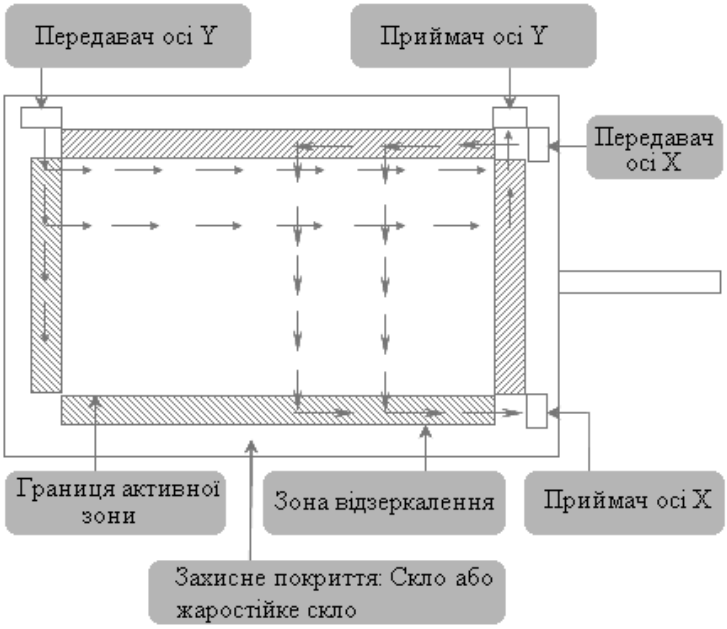


Рис. 6.4.

По краях панелі знаходяться рефлектори та приймаючі датчики. Принцип дії такого екрану полягає в наступному. Спеціальний контроллер формує високочастотний електричний сигнал і посилає його на ПЕП. ПЕП перетворює цей сигнал у хвилю, а рефлектори відбивають її. Ці відбиті хвилі приймаються відповідними датчиками і посилаються на ПЕП. Інші ПЕП приймають відбиті хвилі й перетворюють їх в електричний сигнал, який потім аналізується за допомогою контролера. При дотику до екрану відбувається зміна в хвилях, що поширюються, і це фіксується приймаючими датчиками. Панель реагує на дотик предметом, здатним поглинути хвилю.

Проте, в порівнянні зі всіма іншими типами, резистивні і ємнісні панелі забезпечують набагато меншу вартість і кращу дозволяючу здатність.

Використання резистивної технології має деякі переваги в порівнянні з ємнісною:

- прийнятна ціна;
- екран спрацьовує від натискання пальцем, рукою в рукавичці, або пером;
- висока чуттєвість.

Слід зазначити, що і в інших сенсорних технологій є свої переваги. Наприклад, дискретний сенсорний екран, заснований на ІЧ-скануванні, дозволяє отримати фіксацію координат торкання до дотику (і без дотику). Такий тип сенсорної панелі використовується в деяких моделях логічних аналізаторів Hewlett-Packard.

Ємнісні сенсорні екрани широко використовуються в різних ігрових і торгівельних терміналах, в терміналах довідкових систем аеропорту і вокзалів. Вони найбільш стійкі проти вандалізму, оскільки чутлива поверхня та екран відокремлені від користувача міцним прозорим захисним склом.

Найбільше розповсюдження в сенсорних екранах портативних пристроїв має резистивна технологія. Далі розглядаються конструкції і схемотехніка сенсорних панелей, що використовують саме цей тип сенсорної технології.

6.2. Дискретні і аналогові типи сенсорних панелей

В найпростішому варіанті сенсорну панель можна організувати як матричну клавіатуру (рис. 6.5).

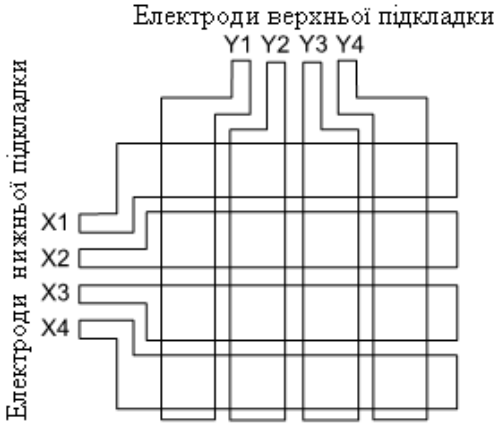


Рис. 6.5.

Топологія електродів верхньої і нижньої підкладки в цьому випадку аналогічна топології плівкової клавіатури. Такі сенсорні панелі можна використовувати як пристрій вводу інформації в поєднанні як з матричними символьними РКІ, так і з графічними РКІ. На екрані РКІ можна організувати систему меню, де користувачу досить лише вибрати потрібну із запропонованих опцій у вікні.

Сканування сенсорної клавіатури здійснюється так само, як і сканування звичайної клавіатури з матричним розташуванням контактів. В такому типі сенсорних панелей окрім однієї переваги – простоти, є і недоліки: велика кількість виводів і обмежені можливості при створенні призначеного для користувача інтерфейсу. До того ж, для отримання малюнку електродів потрібна операція шовкографії. Проте, в простих пристроях, така сенсорна панель може забезпечити зручний інтерфейс користувача.

Показану на рис. 6.6 топологію сенсорної панелі можна віднести до проміжного типу – дискретно-аналогового.

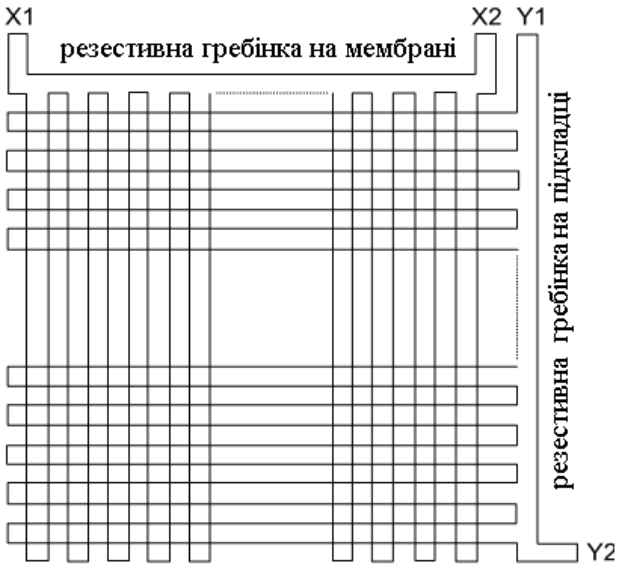


Рис. 6.6.

По суті, це аналогова сенсорна панель, оскільки для здобуття координати точки дотику потрібне використання АЦП. В даної топології лише одна перевага: її використання дозволяє знизити вимогу до однорідності опору провідної плівки. Недолік полягає в тому, що необхідно проводити фотополітографію і травлення малюнка електродів для кожної підкладки. В даний час такий тип практично не використовується, оскільки є постачальники готових скляних підкладок, а також плівок з нанесеним шаром, що мають високу лінійність опору. До таких фірм відноситься, наприклад, Asahi Glass Co, яка є найбільшим в світі постачальником скляних підкладок для більшості відомих виробників РКІ. Asahi Glass є засновником широко відомого виробника РК-дісплеїв – фірми Ohtrex. При виготовленні сучасних аналогових резистивних панелей не потрібне використання операцій фотолітографії і травлення по провідному шару.

6.3. Конструкція і технологія резистивних сенсорних панелей

Резистивна технологія історично стала першим механізмом, що дозволив створити прості та надійні сенсорні екрани. Сенсорна резистивна панель складається з двох склеєних плоскопаралельних прозорих підкладок, на внутрішній поверхні яких нанесене прозоре провідне покриття (плівка In_2O_3 – двоокис індія). В якості нижньої підкладки використовується скляна пластина товщиною 0,8÷2,5 мм, а як верхня – майларова плівка товщиною близько 100 мкм, або тонке скло (рис. 6.7).



Рис.6.7.

Із зовнішнього боку, оберненого до користувача, верхня підкладка покрита шаром антивідблиску і шаром, що забезпечує стійкість сенсорної панелі до подряпин, а також до дії різних хімічних речовин. Робочий зазор, що ізолює провідні шари підкладок один від одного, створюється за допомогою прозорих спейсерів, що мають, як правило, сферичну форму. Як матеріал для спейсерів використовується скло (калібровані скляні кульки), або спеціальний пластик. Спейсери можуть бути нанесені за допомогою різних технологій, зокрема шовкографії. Відомий інший спосіб створення ізолюючого зазору – за допомогою селективного травлення стовпців-спейсерів на внутрішній поверхні скляної підкладки.

Спейсери мають бути нанесені на поверхню нижньої підкладки рівномірно, з кроком 1...20 мм. Топологічно спейсери утворюють сітчастий

малюнок. В ідеальному випадку вони не повинні бути помітними. Лише при розгляданні поверхні сенсорного екрану під певним кутом може бути ледве помітний сітчастий малюнок. Величина зазору, як і крок сітки спейсерів, може змінюватися в широких межах, і вибирається конкретним використанням сенсорної панелі. Механічно верхня плівка еквівалентна системі дискретних мікробалок, утворених сегментами плівки, що спираються на спейсери. При натисканні на верхню плівку відбувається локальна деформація мікробалки і замикання двох провідних шарів. Сенсорна чутливість – мінімальний тиск натискання, при якому відбувається замикання шарів. Чутливість залежить від величини зазору між підкладками, товщини верхньої плівки і кроку спейсерів. Резистивний сенсорний екран може бути оснащений контролерами з RS-232 та USB-інтерфейсами.

6.4. Типи топологічних схем аналогових сенсорних панелей

Існують дві топологічні схеми, за якими реалізуються аналогові сенсорні панелі: чотирьохпровідна і п'ятипровідна.

Ці два типи панелей розрізняються не лише кількістю провідників для підключення та конструкцією, а також алгоритмом отримання координат. На рис. 6.8 показані конструкції цих типів сенсорних резистивних панелей.

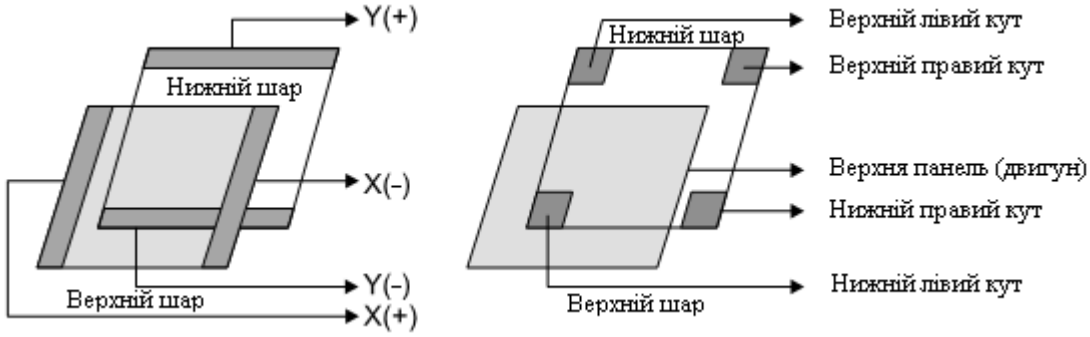


Рис. 6.8.

Резистивний сенсорний екран має багатошарову структуру, що складається з двох провідних поверхонь, розділених спеціальним ізолюючим складом, розподіленим по всій площі активної області екрану. Зовнішній шар виконаний з тонкого прозорого пластика, і його внутрішня провідна поверхня поєднується з провідним шаром основної скляної пластини, яка грає роль каркасу конструкції, завдяки чому відбувається зміна опору всієї системи. Ця зміна фіксується мікропроцесором контролера, який передає координати точки дотику керуючій програмі комп'ютера, яка, у свою чергу, перетворює їх в стандартний сигнал, що виникає при натисканні клавішею миші в будь-якій зоні робочого вікна стандартної програми операційної системи.

6.4.1. Чотирьохпровідна схема

У чотирипровідній конструкції (рис. 6.9) використовуються дві рівнозначні провідні поверхні: одна – визначає положення координати на вісі *X*, інша – по вісі *Y*. Сканування відбувається за дві фази. У першій фазі до електродів провідного шару пластини *X* прикладається тестова напруга (наприклад, до електроду *–X* підключений потенціал *Gnd*, а до *+X* – потенціал V_{ref}). Провідна поверхня підкладки *Y* в даній фазі служить для зняття потенціалу, утвореного інтегральним резистивним дільником в точці дотику. Таким чином, виводи *–X* і *+X* у даній фазі є виходами, і до них підключено джерело опорної напруги, а виводи *–Y* і *+Y* є вхідними і підключаються до входу АЦП. В процесі аналого-цифрового перетворення в даній фазі отримуємо величину, пропорційну координаті по вісі *X*. Відповідно, в другій фазі виконується симетрична процедура, і отримується друга координата по вісі *Y*.

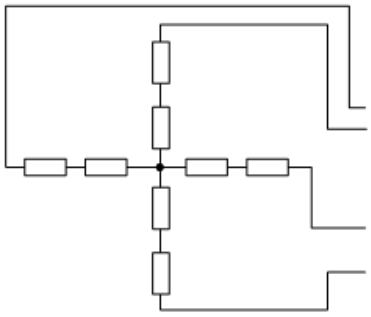


Рис. 6.9.

Для отримання рівномірного розподілу потенціалів уздовж провідної плівки, до торців якої прикладається опорна напруга, використовуються шунтуючі металеві торцеві смужки (рис. 6.10). Смужки виконані зі срібла і мають сферичну форму. Це зроблено для того, щоб компенсувати вплив крайового ефекту в плівковому інтегральному резисторі. Сферична форма електродів дозволяє поліпшити лінійність розподілу потенціалу в резистивному шарі поблизу меж плівки.



Рис. 6.10.

В більшості випадків це дозволяє відмовитися від необхідності програмної корекції координат резистивного шару.

Основні характеристики сенсорних резистивних панелей:

- лінійність провідного покриття підкладок панелі – до 97%;
- прозорість панелі (коефіцієнт пропускання на довжині хвилі 550 нм) – 82…87%;
- питомий опір провідних покриттів – 325…500 Ом/м для верхньої плівки і 400…600 Ом/м для скляної підкладки;
- величина прогину при натисканні (Actuation Travel) – 25…75 мкм;
- зносостійкість (гарантоване число натискань) – до 30 млн. (у Bergquist);
- сенсорна чуттєвість (мінімальне зусилля при натисканні) – 15…500 гр.;
- гарантійный строк експлуатації – до 5 років (у панелей фірми Bergquist).

6.4.2. П’ятипровідна схема

У цій схемі чотири виводи завжди є входами, до яких прикладаються тестові напруги (рис. 6.11), а п'ятий завжди є виходом, з якого виконується зняття потенціалу, що відповідає координатам точки замикання при торканні. Чотири виводи з'єднуються з чотирма кутовими електродами провідної плівки на нижній скляній підкладці, а п'ятий вивід має контакт з провідною плівкою на верхній підкладці сенсорної панелі. При дотику до зовнішнього шару, виконаного з тонкого прозорого пластика, його внутрішня провідна поверхня поєднується з провідним шаром основної скляної пластини, що грає роль каркасу конструкції, завдяки чому відбувається зміна опору всієї системи. Верхня плівка при замиканні є еквівалентом повзунка в змінному резисторі, утвореному нижньою резистивною плівкою. Вимірювання відбувається в дві фази. У першій фазі потенціал *Gnd* прикладений до електродів двох суміжних кутів плівки, а опорна напруга U_{ref} – до останніх двох електродів. В ідеальному випадку бажано отримати лінійний розподіл потенціалів уздовж поверхні плівки від електродів U_{ref} до електродів *Gnd*. Однак в даній схемі набагато сильніше, ніж в чотирипровідній, проявляються граничні ефекти.

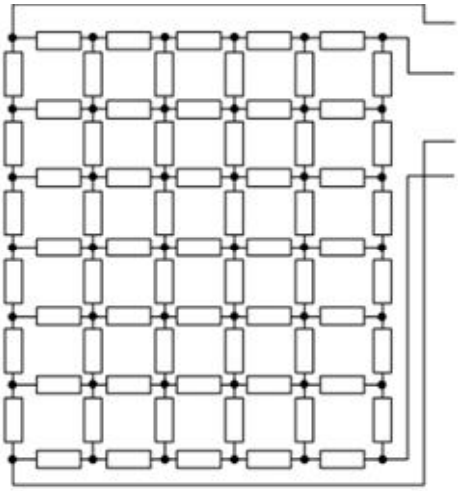


Рис. 6.11.

В результаті, при використанні п'ятипровідної схеми, потрібна програмна корекція координат, яка заснована на табличному методі. Таблиця зашита в окремій мікросхемі EEPROM.

6.5. Драйвери сенсорних панелей

Інтегральна схема (IC) MXB7846 (рис. 6.12) є стандартним, 4-х провідним контролером екранної панелі з сенсорним введенням. IC містить 12-ти розрядний аналого-цифровий перетворювач (АЦП) з синхронним послідовним інтерфейсом і комутаторами, що мають низький опір відкритого каналу, призначеними для управління резистивними сенсорними панелями.

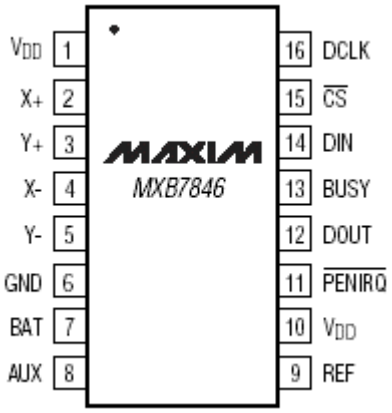


Рис. 6.12.

IC MXB7846 використовує вбудоване +2,5 В, або зовнішнє джерело опорної напруги. IC MXB7846 здатна працювати в режимах абсолютних і логарифмічних вимірів. Крім того, IC має вбудований сенсор температури, канал моніторингу батареї живлення і володіє функцією проведення виміру тиску в точці дотику сенсорної панелі без використання зовнішніх елементів.

Функціональна схема драйвера сенсорної панелі представлена на рис. 6.13.

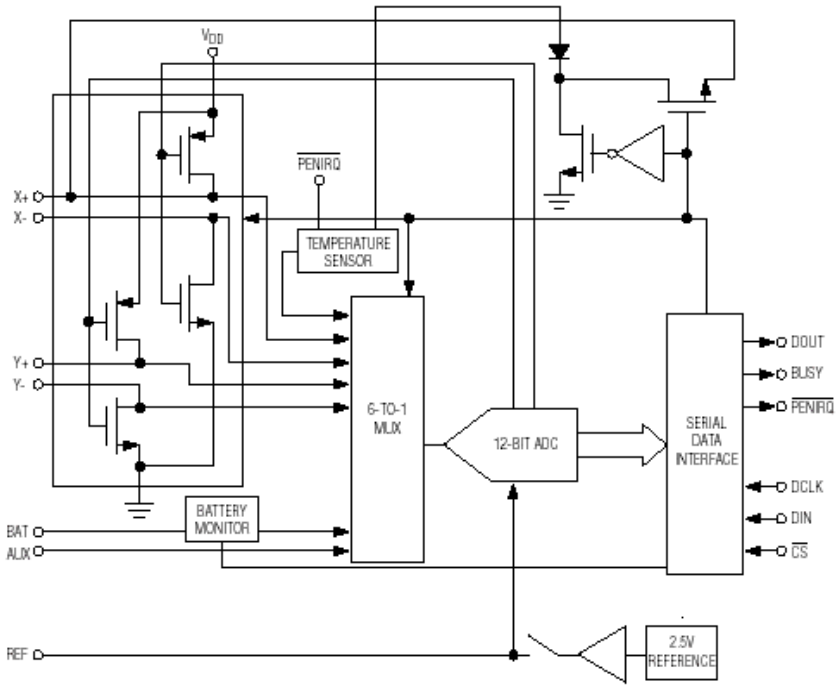


Рис. 6.13.

IC MXB7846 має один допоміжний аналоговий вхід. Всі аналогові входи мають захист від електростатичних розрядів (ESD), який виключає необхідність використання зовнішніх елементів безпеки.

Типова схема включення мікросхеми представлена на рис. 6.14.

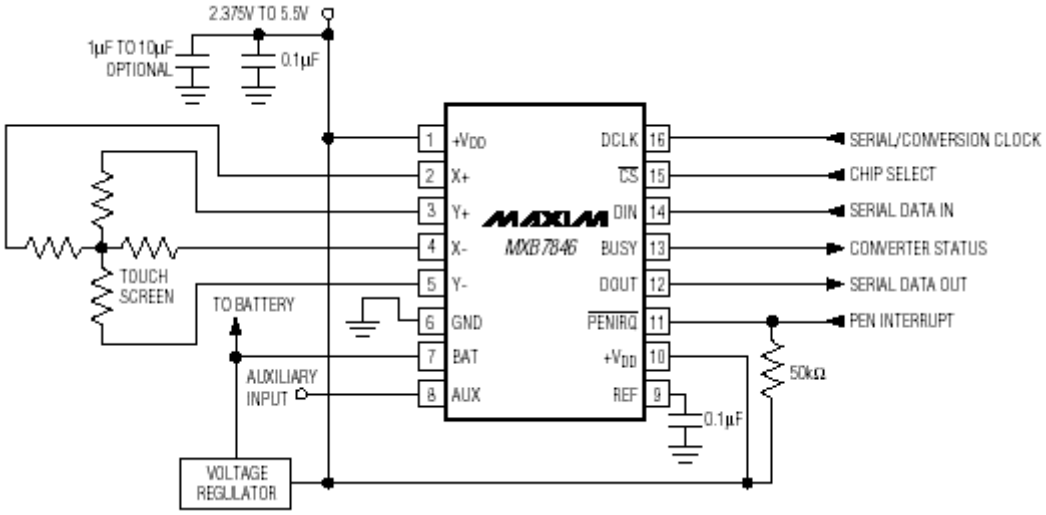


Рис. 6.14.

Питання до розділу 6

1. Область застосування сенсорних панелей.
2. Типи сенсорних панелей. Переваги та недоліки кожної з них.
3. Дискретні та аналогові сенсорні панелі.
4. Конструкція резистивних сенсорних панелей.
5. Алгоритм визначення точки дотику в аналогових сенсорних панелях.
6. Типи топологічних схем аналогових сенсорних панелей.
7. Для сого використовуються драйвери сенсорних панелей.
8. Намалюйте схему сполучення 4-провідної сенсорної панелі з мікроконтролером.

7. ЕКРАНИ

Екрани стали невід'ємною частиною великих міст. Екрани розташовані в місцях великого скупчення людей: автомагістралі, станції метро, торгівельні комплекси і спортивні споруди, – все це є в кожному місті. Ще одна перевага великих екранів – це можливість розміщення великої кількості відеороликів, які змінюватимуть один одного.

7.1. Принципи побудови світлодіодних екранів

Основним елементом світлодіодного екрану є піксель (pixel) – елементарна точка або структурна одиниця зображення (рис. 7.1). Фізичний піксель складається зі світлодіодів основних кольорів (червоний, зелений і синій). Піксель може бути кольоровим або монохромним, мати різні розміри. При цьому незалежно від того, скільки діодів використовується в пікселі – один або декілька, піксель несе в собі інформацію про одну елементарну точку зображення. Пікселі розміщені у вузлах прямокутної сітки екрану, тому можна говорити про розмір пікселя і про крок пікселів в полі екрану. Залежно від задач, що вирішуються, і конструктивних особливостей екрану, співвідношення розміру пікселя до кроку може складати (0,5-0,9):1.

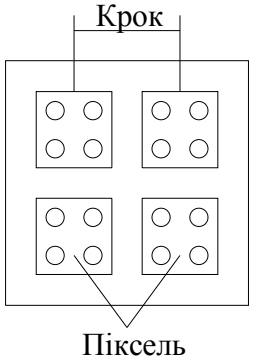
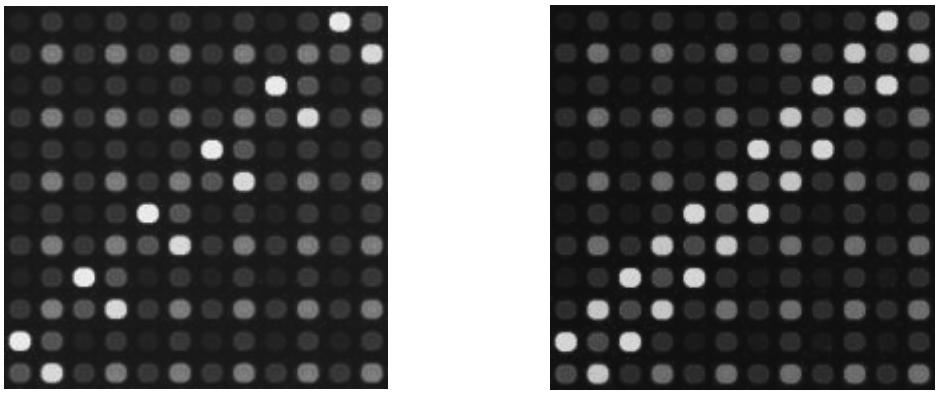


Рис. 7.1.

Віртуальний піксель формується програмно-апаратним способом з набору світлодіодів, що належать сусіднім «фізичним» пікселям (рис. 7.2, а). Використання «віртуального» пікселя дозволяє збільшити дозволяючу здатність екрану в динамічному режимі, тобто при демонстрації рухомих зображень, і, як правило, поліпшити якість зображення. «Віртуальний» і «фізичний» пікселі демонструються в різних кадрах і, за рахунок інерційності зорової системи людини, виникає відчуття подвоєння дозволяючої здатності екрану. Слід зазначити, що технологія «віртуального пікселя» в класичному вигляді застосовується лише в матричних світлодіодних екранах з пікселями, складеними з 4-х світлодіодів, зі світлодіодами розташованими на рівних відстанях один від одного, і не застосовується в екранах, що мають структуру пікселя з 3-х

світлодіодів або з smd-світлодіода «3 в 1». Приклад використання технології віртуального пікселя показаний на рис. 7.2, б.



а *б*

Рис. 7.2.

Незважаючи не деякі переваги, технологія віртуального пікселя дозволяє поліпшити якість інформації, що виводиться на екран, лише в деяких випадках – сюжетні картинки з плавними переходами яскравості, або відеоінформація. В більшості випадків, особливо це стосується зображень з різкими кольоровими переходами, на екрані можуть виникати кольорові артефакти, більш того, частина інформації може взагалі втрачатися.

Крок пікселя (pitch) – відстань між центрами (або початком) двох сусідніх пікселів. Крок пікселя вибирається виходячи з відстані спостереження і необхідної дозволяючої здатності екрану. Іншими словами, крок пікселя визначає, з якої мінімальної відстані глядач може комфортно сприймати зображення на екрані, не помічаючи піксельної структури екрану. Людина зі стовідсотковим зором не розрізняє дві точки на екрані, якщо відстань між ними складає менше однієї кутової хвилини (1/60 градуса). Так, для кроку пікселя 10 мм ця відстань складає приблизно 20 м; для пікселя 30 мм вона складає приблизно 50 м. При великих відстанях спостереження, крок пікселя може бути збільшений. Крім того, існує поняття «відстані комфортного спостереження». Як правило, вона вибирається в межах 3-8 розмірів діагоналі екрану і більше. Проте, навіть при вдвічі меншій відстані спостереження людина може дивитися на екран тривалий час, не відчуваючи стану дискомфорту (наприклад розсіюючий екран в кінотеатрі). У таблиці 7.1 приводяться дані про вибір відстані спостереження в залежності від розміру кроку пікселя екрану і від розмірів екрану.

Табл. 7.1.

Крок пікселя, мм	3	6	10	15	20	25	30	35	40
Мінімальна відстань до екрана, м	5	10	17	25	34	43	52	60	70

Крок пікселя безпосередньо задає дозволяючу здатність світлодіодного екрану, а отже і якість інформації, що відображається.

Розрішаюча здатність (resolution) – це здатність світлодіодного екрану відтворювати дрібні деталі інформації, що відображаються (відео або графіки). Ця характеристика є однією з найбільш важливих для світлодіодних екранів. дозволяюча здатність виражається в кількості адресованих точок, або пікселів на екрані, по двох координатах: X і Y. Існує стандартний ряд дозволяючих здатностей, що відповідають формату зображення 4:3 – 160×120; 256×192; 320×240; 640×480; 800×600; і т.д. Відповідно, чим вища дозволяюча здатність екрану, тим більш детальним буде зображення, що виводиться на нього. На практиці для демонстрації рекламних відеороликів дозволяюча здатність має бути не менше 256×192.

Світлодіодний модуль – структурна одиниця, з яких збирається екран необхідного розміру. Світлодіодний модуль, як правило, комплектується своїм елементом управління і блоком живлення. Модулі для формування екранів можуть мати різні лінійні розміри, характеризуватися різними розмірами фізичних пікселів, їх складом, і т.д. Виконуються як для зовнішнього використання (outdoor виконання), так і для використання всередині приміщень (indoor виконання). На рис. 7.3 показана структура світлодіодного екрану, що складається зі світлодіодних модулів.

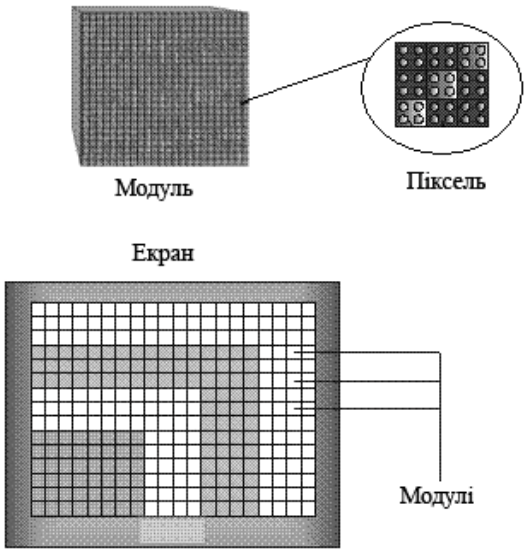


Рис. 7.3.

До переваг світлодіодних екранів належать:
– висока яскравість і контрастність зображення;
– висока якість повнокольорового зображення;
– відображення динамічної інформації;
– можливість прямої відеотрансляції;
– модульна конструкція екранів;

– відносно невеликий розмір по глибині (200-300 мм);
– можливість "нарощування" екранів;
– висока надійність і довговічність (теоретично до 100000 годин);
– низьке енергоспоживання (у порівнянні з іншими технологіями);
– можливість об'єднання екранів в мережу і управління ними з єдиного центру.

Завдяки можливості об'єднання екранів в мережу, можливо безпосередньо, з центрального комп'ютера, передавати управляючому комп'ютеру кожного екрану відеоінформацію. На рис. 7.4 представлена блок-схема системи управління світлодіодним екраном.

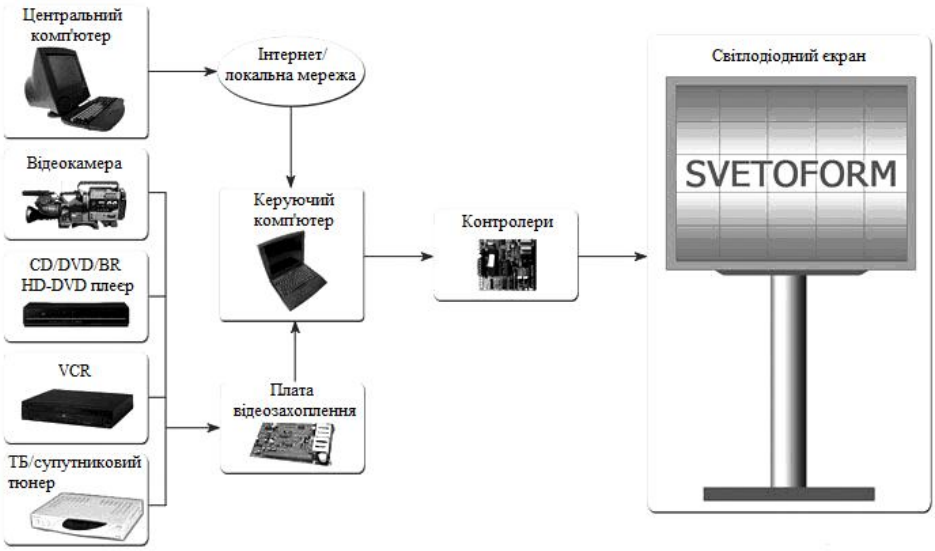


Рис. 7.4.

При роботі зі світлодіодним екраном передача інформації і управління світлодіодними модулями здійснюється контролерами, які приєднані до управляючого комп'ютера. У комп'ютері встановлюється плата відеопроцесора (або плата відеозахоплення), на вхід якої можуть подаватися відеосигнали від різних джерел – звичайний телевізійний сигнал, сигнал від відеомагнітофона, DVD-плеєра, відеокамери, іншого комп'ютера і т.д. Крім того, на жорсткий диск управляючого комп'ютера можуть бути записані заздалегідь підготовлені сюжети, які можна відтворювати за визначеним розкладом. З появою великої кількості відеоекранів виникла необхідність об'єднання світлодіодних екранів, що стоять окремо, в мережу, з єдиним центром управління. Управління мережею відеоекранів здійснюється з центрального комп'ютера по каналах зв'язку (модем/радіомодем, виділена лінія і т.д.) з комп'ютерами управління кожного окремого екрану. Слід зазначити також, що важливою характеристикою відеоекрану є співвідношення його сторін 4:3 (формат зображення), оскільки для прямої трансляції відео необхідно дотримуватися цього співвідношення, інакше зображення буде або обрізане, або відтворюватиметься на частині екрану. Якщо світлодіодний

екран використовується лише для показу заздалегідь підготовлених роликів, то співвідношення сторін не має значення і може бути будь-яким.

Основні світлотехнічні параметри світлодіодних екранів:

– лінійний розмір (ширина, висота);

– яскравість (кд/м^2);

– контрастність;

– розрішаюча здатність;

– крок пікселя, розмір пікселя;

– склад пікселя (кількість світлодіодів, з яких складається піксель);

– кількість світлових відтінків (градацій яскравості);

– кут огляду;

– частота регенерації зображення (зміна кадрів).

Яскравість – величина, що характеризує світлове випромінювання ділянки поверхні джерела світла в заданому напрямі. Яскравість вимірюється в канделах на квадратний метр. Для екранів характеристика яскравості дається для білого кольору. Для застосування в приміщенні вважається достатньою яскравість в 1000 Кд/м^2. Для застосування на вулиці (у сонячний день) необхідна яскравість не менше 5000 Кд/м^2.

Контрастність – відношення максимальної яскравості світіння екрану при відображенні інформації до яскравості тієї ж ділянки екрану без інформації, що відображається. Зазвичай контрастність вказується для умов роботи в абсолютно темному приміщенні. Типові значення контрастності для світлодіодних екранів складають від 500:1 до 1000:1. Проте, за умови зовнішнього освітлення, контрастність істотно знижується і визначається коефіцієнтом віддзеркалення лицьової поверхні екрану. З метою підвищення контрастності зображення на передній площині присутні сонцезахисна гратка.

Кут огляду екрану зазвичай визначається кутом огляду (діаграмою спрямованості) окремого світлодіода і залежить від типу світлодіода. У тому випадку, коли на лицьовій поверхні екрану використовуються сонцезахисні елементи, кут огляду по вертикалі зменшується.

Кількість кольорових відтінків, що відтворюється екраном, визначає якість кольорової палітри екрану. Повнокольорове зображення (TRUE COLOR) досягається за наявності 256 відтінків кожного основного кольору системи RGB (червоного, зеленого і синього), що відповідає 16,78 млн. відтінків для 3-х кольорового пікселя. У простому випадку піксель екрану складається з трьох світлодіодів з різним кольором світіння (зазвичай червоний, зелений, синій). Для великих пікселів використовуються від 4 світлодіодів (два червоних, зелений і синій) і більше. Кількість і тип окремих світлодіодів зазвичай вибирають з урахуванням максимального наближення до балансу білого кольору.

Частота регенерації зображення повинна забезпечувати максимально можливий в даних умовах час роботи з екраном без відчуття дискомфорту. В даний час це значення складає не менше 100 Гц.

До перерахованих світлотехнічних параметрів світлодіодних екранів слід додати так звану нерівномірність яскравості зображення, яка має місце при використанні даної технології відображення інформації, і пов'язана, в першу чергу, з якістю виготовлення світлодіодних модулів. Якщо для проекційних систем нерівномірність яскравості виражається, в основному, в спаді яскравості на границях системи, то для світлодіодних екранів на перше місце виступає піксельна нерівномірність яскравості. Це пов'язано з тим, що інформаційне поле екрану складається з окремих світлодіодів, в яких завжди існують технологічні розкиди за силою світла. Виробники світлодіодів розділяють діоди на ранги, в межах кожного з яких сила світла діодів відрізняється не більше ніж на 15 – 30%. Другим джерелом нерівномірності яскравості може стати неоднакова орієнтація світлодіодів в полі екрану, що приводить до зсуву діаграми спрямованості.

Надійність екрану визначається надійністю світлодіодів і схеми управління. Провідні виробники заявляють термін служби світлодіодів при експлуатації їх в номінальному режимі до 100 000 годин.

Великі екрани на основі світлодіодних модулів дозволяють відображати комп'ютерну графіку і анімацію, транслювати відео матеріали і ТВ-програми, а також передавати будь-яку текстову інформацію.

7.2. Принципи побудови великих плазмових екранів

Розвиток персональних комп'ютерів стимулював не лише істотне поліпшення параметрів пристроїв відображення інформації на базі електронно-променевих трубок, але і дав новий імпульс розвитку іншим дисплейним технологіям. Дисплеї, виконані за альтернативними технологіями, під загальною назвою плоскопанельні дисплеї (далі FPD - Flat Panel Displays) не лише стали успішно конкурувати з електронно-променевими трубками в області комп'ютерних моніторів, але і почали тіснити їх у традиційній телевізійній області.

Великі кольорові плазмові модульні екрани є достатньо привабливими для цілей колективного спостереження в приміщеннях. Проводячи порівняння плазмових екранів зі світлодіодними екранами і відеостінами можна відзначити наступне:

– плазмові екрани мають істотно (у 5 – 7 разів) нижчу ціну і вищу якість зображення в порівнянні зі світлодіодними екранами, поступаючись їм в яскравості;

– плазмові екрани мають на порядок меншу товщину, менші експлуатаційні витрати і вищу якість зображення в порівнянні з відеостінами, але поступаються їм в дозволяючій здатності.

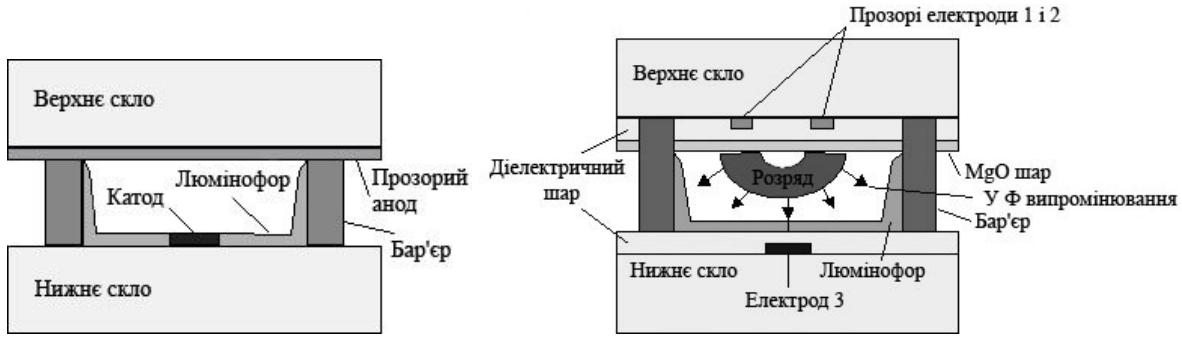
Пласкопанельні дисплеї за своєю структурою належать до класу матричних індикаторів, в яких світловипромінюючі, світловідбиваючі або

світлопропускаючі комірки утворюють прямокутну матрицю. Управління комірками здійснюється за допомогою системи перпендикулярно розташованих електродів. У простому випадку (для двоелектродних систем) для обслуговування матриці з $M \times N$ комірок потрібно $M+N$ управляючих входів. Для отримання кольорового зображення три або чотири комірки групуються в елемент зображення – піксель.

При подачі на вертикальний і горизонтальний електроди напруги, що перевищує напругу займання, у комірці виникає газовий розряд, обмежений бар'єрами. Ультрафіолетове (УФ) випромінювання газового розряду збуджує фотолюмінофор, що випромінює світло червоного, зеленого або синього кольору. Люмінофор зазвичай наноситься на дно і бічні стінки комірки, а інколи – тонким напівпрозорим шаром на стелю комірки. Верхня система електродів робиться прозорою або досить вузькою для того, щоб не перешкоджати виходу світла. Три або чотири комірки з різними кольорами світіння утворюють квадратний піксель.

В комірках змінного струму використовується складніша комірка, в якій електроди відокремлені від газового простору діелектричними шарами. За наявності додаткових електродів поява розряду у комірці відбувається за допомогою електродів 1 і 3, а підтримка горіння так званого планарного розряду – між електродами 1 і 2.

На рис. 7.5, а, показано схематичне зображення простої комірки постійного струму, а на рис. 7.5, б – структура комірки змінного струму.



а б

Рис. 7.5.

Плазмові екрани (рис. 7.6) будуються на базі плазмових панелей постійного струму. Панеллю є два скла, між якими розміщені керамічні гратки, утворюючи прямокутні матриці комірок. У керамічні гратки вмонтовані вертикальні і горизонтальні електроди. По периметру панелі розташований герметизуючий шов. Простір між склом заповнений сумішшю інертних газів. При прикладенні до двох перпендикулярних електродів управляючої дії, у комірці виникає плазмовий розряд, випромінюючий ультрафіолет. Цей ультрафіолет збуджує розташований на дні кожної комірки люмінофор з червоним, синім або зеленим кольором

світіння. Конструкція панелей дозволяє стикувати їх одна з одною, з мінімальними порушеннями цілосності зображення.

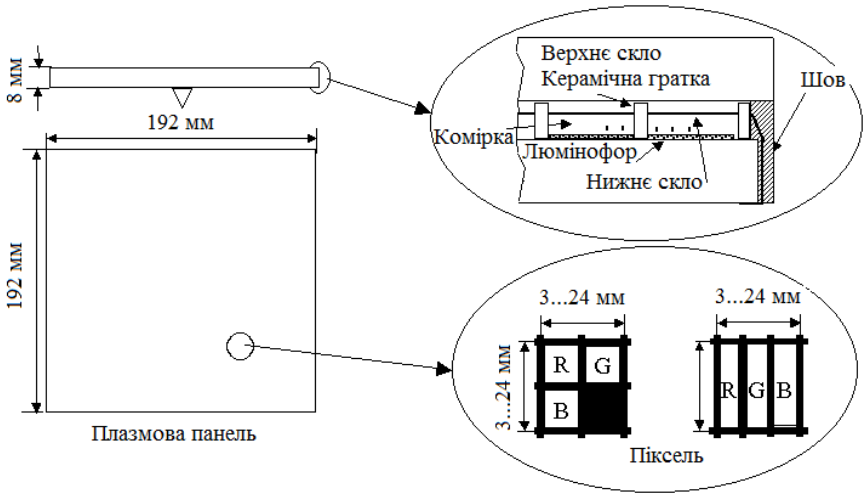


Рис. 7.6.

При прикладенні до електродів напруги в області катодного шару утворюється значний об'ємний розряд, що призводить до істотного перерозподілу потенціалу вздовж розряду. У полі цього заряду прискорюються електрони, які іонізують газ. Втративши енергію ці, а також вторинні електрони, дрейфують до анода. В межах темного простору вони знову набирають енергію, достатню для іонізації атомів, внаслідок чого утворюється позитивний стовп. На рис. 7.7 умовно показаний зовнішній вигляд і розподіл параметрів уздовж розряду.

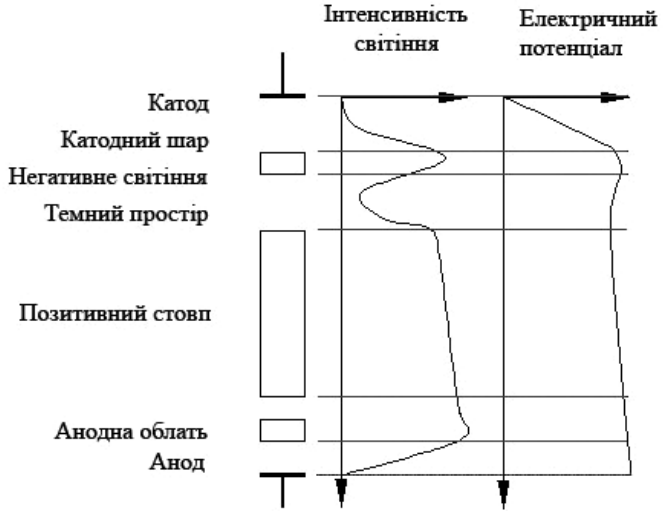


Рис. 7.7.

У більшості плазмових дисплеїв використовується ультрафіолетове випромінювання, що виходить з області негативного світіння.

Робочим середовищем плазмових дисплеїв є інертні гази. Для зниження потенціалу займання використовуються так звані пенінговські

суміші, з добавкою до основного газу домішок іншого газу, потенціал іонізації якого нижчий від енергії збудження метастабільного рівня основного газу. Найбільш поширеним газовим наповненням плазмових дисплеїв є суміш інертних газів на основі гелію або неону з додаванням ксенону.

Панелі, що випускаються, можуть мати різний розмір пікселя і, відповідно, різне число пікселів в панелі:

- 64×64 пікселя розміром 3 мм;
- 48×48 пікселя розміром 4 мм;
- 32×32 пікселя розміром 6 мм;
- 24×24 пікселя розміром 8 мм;
- 16×16 пікселя розміром 12 мм;
- 12×12 пікселя розміром 16 мм;
- 8×8 пикселів размером 24 мм.

Великі екрани будуються з окремих функціонально-закінчених одиниць – модулів. До складу модуля входить 6 панелей, модульно-панельний контролер, джерело живлення і конструктив. На рис. 7.8 показана функціональна схема модуля.

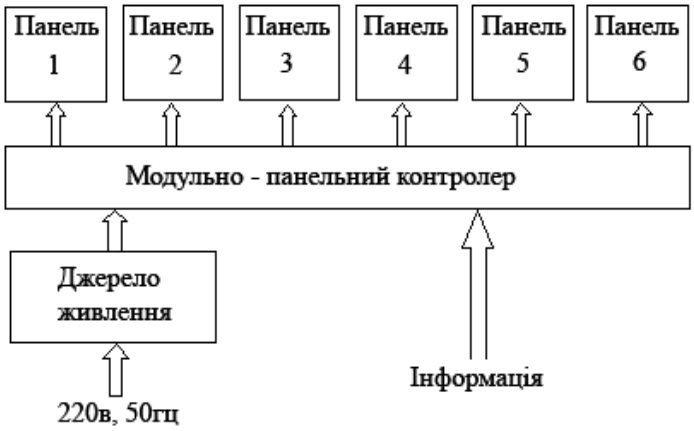


Рис. 7.8.

Модульний принцип побудови екранів дозволяє практично необмежено нарощувати їх розміри й інформаційну ємність по горизонталі і вертикалі. Загальні параметри модулів зведені в табл. 7.2.

Табл. 7.2.

Максимальна споживана потужність, Вт	< 330
Средня споживана потужність, Вт	< 180
Строк служби (при 8-годинній роботі за добу), років	7
Кут спостереження по вертикалі і горизонталі,	160^0
Температура навколишнього середовища, ^{0}C	+5 ...+45
Вага, кг	< 10

Для того, щоб зображення на модульному екрані виглядало цілісним, панелі для модульних екранів повинні забезпечувати стиковку одна з одною з мінімальною втратою кроку пікселів. На рис. 7.9 показаний стик двох таких панелей, в яких використані об'ємні електроди.

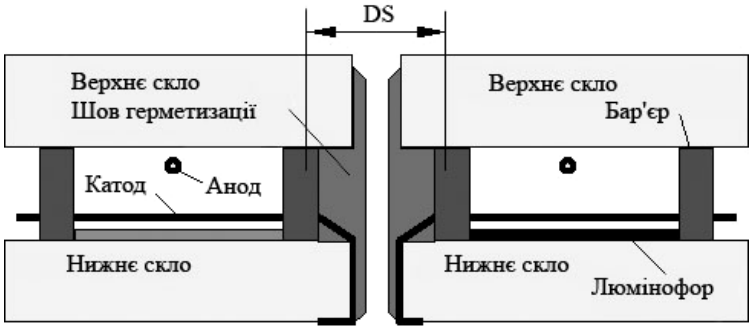


Рис. 7.9.

Подвійний розмір шва DS складається з наступних компонентів: товщина бар'єру, товщина двох швів герметизації, технологічний проміжок між панелями. Розмір DS складає близько 3 мм. Використовуючи оптичне маскування швів, можна досягти практично їх повного візуального зникнення. Але все одно є наявність швів, що проявляються на зображенні у вигляді регулярної сітки.

Управління екраном відбувається від комп'ютера. Для зв'язку комп'ютера з екраном використовується високошвидкісний цифровий інтерфейс DVI. Відстань від комп'ютера до екрану може складати від 5 до 10 метрів з використанням відео кабеля, або до 500 метрів з використанням оптичного кабеля.

Найважливішою характеристикою плазмових екранів є ефективність, яка характеризує міру перетворення електричної енергії в світлову. Теоретична (100%) ефективність складає 683 Лм/Вт. У кращих зразках газорозрядних ламп ефективність досягає 100 Лм/Вт. Сучасні плазмові панелі знаходяться на рівні 0,5 – 1,0 Лм/Вт, тому досягнення ефективності навіть 5 Лм/Вт вже є дуже серйозним завданням. Розподіл потужності в 42-дюймовій плазмовій панелі показаний на рис. 7.10.

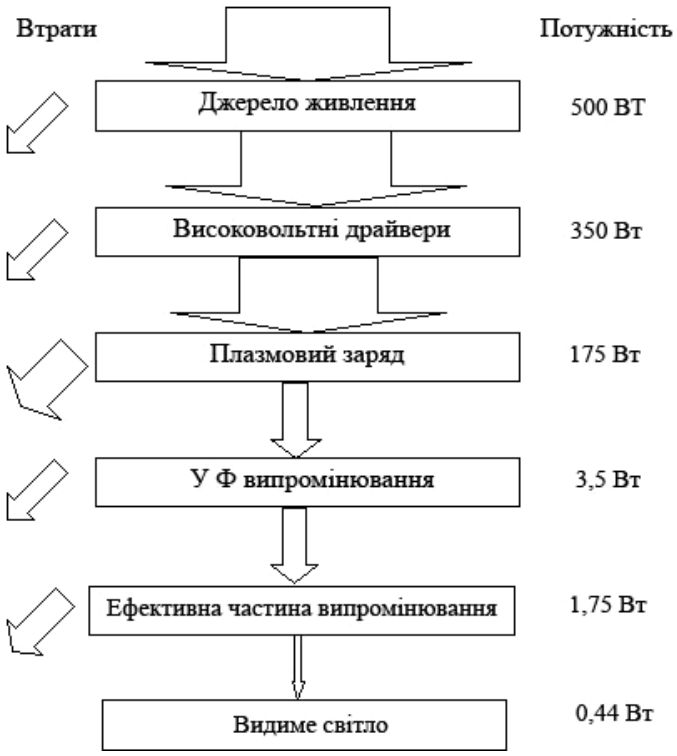


Рис. 7.10.

На ефективність плазмових панелей впливає безліч чинників, серед яких наступні:

- високовольтні драйвери;
- джерела живлення;
- резистивні енергетичні втрати;
- ємнісні енергетичні втрати;
- електроди;
- люмінофори;
- скло.

У сучасних кольорових екранах для отримання кольору використовуються фотолюмінофори, що перетворюють ультрафіолетове випромінювання газового розряду у видиме випромінювання червоного, зеленого і синього кольорів.

Квантова ефективність фотолюмінофора визначається тим, скільки фотонів видимого світла збуджує один фотон ультрафіолетового спектру. Для звичайних фотолюмінофорів квантовий вихід близький до одиниці, тобто один фотон ультрафіолетового спектру викликає випромінювання одного фотона видимого світла. Спектральна ефективність люмінофора визначається співвідношенням довжин хвиль збудження і випромінювання. Наприклад, для зеленого люмінофора при збудженні 147 нм і випромінюванні 570 нм ефективність складе 0,258.

Енергетична ефективність випромінювання люмінофором видимого світла характеризує втрати в люмінофорному шарі. Типові значення енергетичної ефективності складають від 0,1 до 0,3.

Засвічення люмінофора ультрафіолетовим випромінюванням залежить від структури газового розряду і від області нанесення люмінофорів. Для підвищення світлового потоку необхідно збільшувати об'єм плазми, що генерує ультрафіолет, збільшувати сумарну площу люмінофора у комірці і наближати джерело випромінювання до поверхні люмінофора.

Кольорова (спектральна) чистота люмінофора визначається шириною спектру випромінювання. Типове значення ширини спектру для синього фотолюмінофора складає 100 нм, для зеленого – 150 нм і для червоного – 200 нм. Фотолюмінофори, що використовуються в сучасних плазмових екранах генерують видиме світло з наступними довжинами хвиль: R=680 нм, G=570 нм, B=460 нм.

Кольоровий баланс по білому кольору для інтенсивностей основних (R, G, B) кольорів визначається наступним кольоровим рівнянням:

$$W = 0{,}3R + 0{,}59G + 0{,}11B.$$

Зазвичай відносна інтенсивність люмінофорів не збігається з коефіцієнтами даного рівняння. Тому для балансування по білому доводиться або знижувати концентрацію люмінофора, або корегувати яскравість по кольорах за допомогою управляючих дій. Обидва способи здійснюють балансування по білому кольору шляхом зниження яскравості. Перспективнішим способом є перерозподіл площі RGB комірок. За рахунок такого перерозподілу можна підвищити світловий потік на 20–30%. Але зміна площі комірки веде до зміни її електричних характеристик, що може потребувати досить складних схемотехнічних корекцій. Ще одним перспективним варіантом є використання четвертого кольору, що, за різними оцінками, дозволить підвищити світловий потік на 30–50%.

Оскільки ККД плазмових екранів як джерела світла складає десяті долі відсотка, то практично вся потужність, що підводиться до панелі, перетворюється в тепло. Підвищення робочої температури призводить до різкого зниження терміну служби. Тому підвищення яскравості повинно здійснюватись за рахунок підвищення не потужності, а ефективності.

На відміну від CRT-екранів, де використовується поелементна розгортка, в плазмових екранах застосовують порядкову розгортку, тобто одночасне відображення всіх елементів одного рядка. У простому випадку середня яскравість зворотньо пропорційна кількості рядків. Тому вимога підвищення яскравості суперечить вимозі збільшення інформаційної ємності. Підвищення яскравості за рахунок збільшення робочих струмів призводить до падіння терміну служби панелі через затемнення внутрішньої поверхні верхнього скла продуктами розпилення катодів.

Тому, в сучасних плазмових екранах використовується ефект внутрішньої пам'яті комірок. В результаті з'являється можливість одночасної ініціалізації більш ніж одного рядка панелі, що призводить до багатократного зниження миттєвої яскравості. Використання внутрішньої пам'яті комірок дозволяє здійснити "м'яке" підвищення яскравості не за рахунок підвищення миттєвої яскравості, а за рахунок збільшення сумарного часу горіння комірки в кадрі.

Контраст, так само як і яскравість, належить до важливих параметрів екранів. Є два джерела засвічування панелі, які знижують контраст. Перше джерело – це світло, що випромінюється панеллю, яке не пов'язане із зображенням, наприклад, паразитне підсвічування світлом з допоміжних комірок. При яскравості цього засвічування B_{min} і максимальній яскравості B_{max} власний контраст C_o (контраст для темного приміщення) визначиться як:

$$C_o=B_{max}/B_{min}.$$

Друге джерело засвічування – відбите люмінофорами та іншими структурами панелі світло. В цьому випадку контраст вимірюється для деякого значення зовнішньої освітленості B_e. Зовнішній контраст (контраст при денному світлі) визначається виразом:

$$C_e=(B_{max}+B_e)/(B_{min}+B_e).$$

Для підвищення власного контрасту окрім збільшення максимальної яскравості, існує ще один шлях – вдосконалення конструкції екранів, спрямоване на зниження паразитного засвічування. Для підвищення зовнішнього контрасту існують наступні шляхи:

- Збільшення максимальної яскравості панелі. Цей шлях вимагає великих зусиль, оскільки підвищення яскравості можливе лише за рахунок підвищення ефективності.
- Зменшення коефіцієнтів віддзеркалення зовнішнього світла люмінофорами й іншими внутрішніми структурами панелі. Цей спосіб досить простий, але, на жаль, таке зменшення призводить до втрат світлового потоку, що генерується у комірці. Іншим варіантом є пігментація люмінофорів, тобто додавання в люмінофорні пасти неорганічних фарбників, що поглинають всі частини видимого спектру, окрім тієї частини, яка генерується даним люмінофором.
- Використання нейтральних світлофільтрів, засноване на тому факті, що при коефіцієнті пропускання такого фільтру K, світло, що випромінюється панеллю, буде ослаблене в K разів, а зовнішнє світло, пройшовши через світлофільтр туди і назад – в K^2 разів. Приблизно такий же ефект дає створення на лицьовій поверхні панелі мікрорельєфу, що розсіює зовнішнє світло.

- Використання спектральних світлофільтрів – перспективний шлях підвищення зовнішнього контрасту. Підвищення контрасту засноване на тому, що для кожної комірки використовується світлофільтр зі смугою пропускання, що відповідає кольору, який випромінюється даною коміркою.

В існуючих екранах використовується три типи внутрішньої пам'яті: ємнісна (для панелей змінного струму), резисторна і динамічна (для панелей постійного струму).

Принцип використання внутрішньої пам'яті комірок базується на наступних факторах:

- У PDP існує одночасний доступ до всіх комірок одного рядка. Спроба одночасного запалення двох комірок, об'єднаних по анодах, призведе до того, що внаслідок неминучого розкиду параметрів комірка, яка спалахне першою, перешкодить запаленню другої. Наявність пам'яті робить електричні характеристики комірок незалежними, що дозволяє одночасно запалювати декілька комірок по одному аноду.
- Для того, щоб в стовпцях панелі запалювалися не всі, а лише потрібні комірки, в екранах з внутрішньою пам'яттю до операцій запису і гасіння додається операція підтримки. Якщо до комірок прикласти напругу підтримки, яка вища за напругу горіння, але нижча за звичайну напругу запалення, то знову запаляться лише ті комірки, які "пам'ятають" про те, що вони були активізовані при записі.

В даний час розробка і виробництво плазмових екранів зосереджені в Японії і Південно-східній Азії. Такі фірми, як Fujitsu, Mitsubishi, NEC, Pioneer, Hitachi, Matsushita, Samsung, LG, Orion, і ряд інших вже випускають такі екрани в промислових кількостях, як для комп'ютерних дисплеїв, так і для настінних телевізорів. У табл. 7.3 приведені параметри плазмових екранів фірм Fujitsu і Pioneer.

Табл. 7.3.

Виробник	Fujitsu 42"	Pioneer 40"
Размір екрану, мм	920×518	806×604
Формат	16:9	4:3
Інформаційна ємність	852×480	640×480
Крок пікселів, мм	1,08×1,08	1,26×1,26
Кількість кольорів, млн.	16,7	16,7
Яскравість, Кд/м^2	300	400
Контраст	70:1	150:1

Кут спостереження, градусів	160	160
Габарити дисплея, мм	985×590×75	946×740×100
Маса, кг	18	30

На рис. 7.11 представлені фотографії плазмових екранів (а – модульний плазмовий екран; б – плазмовий телевізор фірми NEC) [22].



а *б*

Рис. 7.11.

7.3. Електролюмінесцентні дисплеї

Явище електролюмінесценції – це процес безпосереднього перетворення електричної енергії в світло. В електролюмінесцентних дисплеях світло генерується за допомогою ударного збудження світловипрмінюючих активаторів електронами з високою енергією в тонкоплівкових структурах. Світловипромінюючі дисплеї мають найбільш просту структуру, оскільки не вимагають зовнішнього джерела випромінювання і складної оптики. Такі дисплеї належать до пристроїв відображення інформації на основі тонкоплівкових електролюмінісцентних структур з сильними електричними полями (Thin Film Electroluminescent – TFEL). На рис. 7.12 приведена структура TFEL-індикатора.

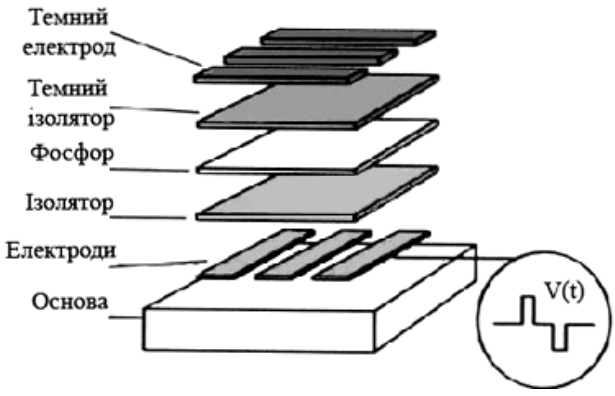


Рис. 7.12.

Центральний шар є тонкою плівкою фосфору, який випромінює світло при дії сильного електричного поля з напруженістю 1,5 МВ/см. Через таке сильне електричне поле будь-які дефекти в тонкоплівковій структурі, здатні створити коротке замикання, можуть послужити причиною розсіювання руйнівної кількості енергії, якщо фосфор безпосередньо з'єднається з електродами. Тому на кожній із сторін фосфорного шару необхідні ізолятори для того, щоб сформувати надійну структуру пристрою. Ізолятори обмежують максимальний струм заряду і розряду ємностей. Електроди знизу і зверху пристрою завершують основну ємнісну структуру. Один набір з цих електродів має бути прозорим для того, щоб мати можливість спостерігати випромінюване світло.

Контрастність покращується шляхом спеціальної обробки скла, що зменшує віддзеркалення падаючого ззовні світла. На рис. 7.13 представлений поперечний розріз індикаторної панелі.

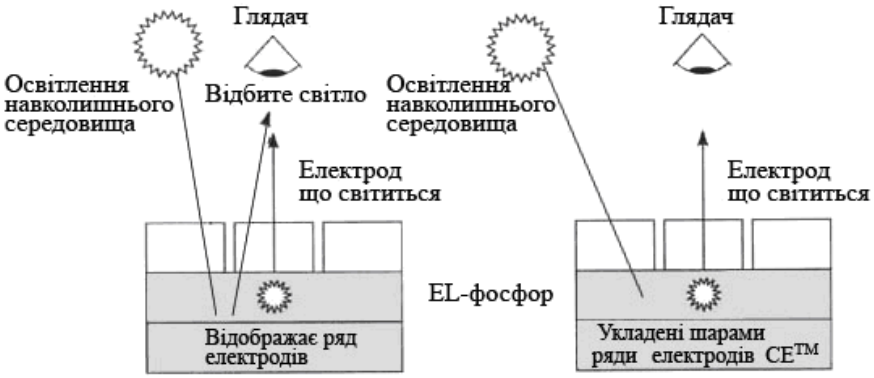


Рис. 7.13.

Спеціальна обробка нижнього шару електродів зменшує його здатність віддзеркалення і створює темний фон для всіх пікселів, що не світяться. В результаті виходить чітке та ясне зображення, що добре читається при будь-якому зовнішньому освітленні й видиме з будь-якої точки фронтального простору.

Світловипромінюючі електролюмінесцентні дисплеї мають просту структуру, оскільки не вимагають зовнішнього джерела випромінювання і додаткової оптики.

Електролюмінесцентні дисплеї мають чудові експлуатаційні характеристики:

- широкий кут огляду (більше 160°);
- яскравість до 150 Кд/м^2;
- швидке безінерційне відображення;
- розширений діапазон робочих температур (–40...85°С);

– високий показник надійності, середній час безвідмовної роботи до 100000 годин;
– компактна і надійна конструкція.

Основні недоліки електролюмінесцентних дисплеїв пов'язані з високою робочою напругою (від 170 до 210 В) і з необхідністю ізоляції високовольтних сигналів від невибраних елементів відображення та логіки, а також із застосуванням дорогої управляючої електроніки.

7.4. Рідкокристалічні екрани

Рідкокристалічні екрани (РКЕ) є одними з основних засобів виведення інформації для сучасних цифрових систем. З точки зору управляючого інтерфейсу РКЕ діляться на два великі класи – з контролерами і без контролерів. Перші застосовуються в системах, що мають не дуже великі обчислювальні ресурси. Наявність контролера повністю звільняє програміста від операції регенерації зображення – йому необхідно лише переслати інформацію в ОЗП дисплея. Для цього використовується простий 8-розрядний паралельний інтерфейс.

Дисплеї без контролерів простіші за структурою, але для управління вимагають набагато більших обчислювальних ресурсів – вони застосовуються у складі потужних обчислювальних систем. Багато сучасних 16- і 32-розрядних мікроконтролерів мають вбудовані інтерфейси РК панелей, що істотно полегшує задачу управління. За відсутності контролера всі часові діаграми розгортки необхідно генерувати програмно.

Структура дисплея приведена на рис. 7.14.

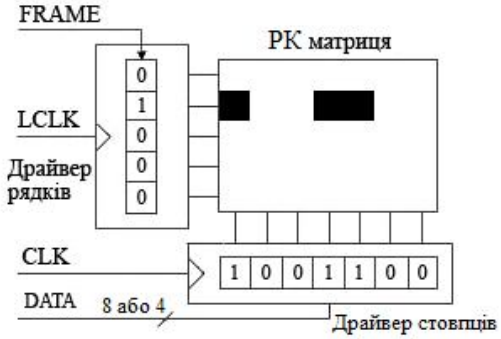


Рис. 7.14.

Для управління РК панеллю використовуються дві мікросхеми – драйвер рядків і драйвер стовпців. Ці мікросхеми встановлені на платі РКЕ. Кількість управляючих виводів мікросхем завжди не менша числа точок на дисплеї по вертикалі і по горизонталі відповідно (у разі, коли РКЕ має великі розміри, застосовується каскадне з'єднання драйверів). Мікросхеми мають в своєму складі регістри, що управляють напругою на виводах. Одиниця в регістрі означає включення рядка/стовпця, нуль – виключення. Точка на екрані засвічується в тому випадку, якщо вона стоїть на перетині включених рядка і стовпця.

Процес розгортки РК панелі відбувається таким чином. Спочатку відбувається заповнення регістра стовпців. Дані передаються по 8- або 4-розрядній шині і тактуються сигналом CLK (очевидно, в разі 8-розрядної шини заповнення рядка відбудеться за кількість тактів, що дорівнює кількості точок, поділених на 8). Далі подається імпульс рядкової синхронізації LCLK. При цьому відбувається прокрутка регістра зсуву в драйвері рядків на один біт. У цей регістр під час передачі першого рядка установкою сигналу FRAME вводиться 1, а при передачі подальших рядків вводяться 0. Таким чином, прокрутка регістра на один біт означає включення наступного (нижнього) рядка.

Для оцінки обчислювальної потужності, необхідної для розгортки РКЕ без контролера, підрахуємо частоту отримання даних. Припустимо, що є панель з дозвляючою здатністю 320×240 точок і 8-бітним інтерфейсом, яку передбачається розгортати з частотою кадрів 60 Гц. Тоді частота сигналу CLK буде дорівнювати:

$$f_{CLK} = 320 / 8 \times 240 \times 60 = 576 \text{ кГц}$$

Існують дисплеї з подвійною розгорткою (Dual Scan), структура яких дозволяє досягти якісного зображення при меншій частоті кадрів. РК панель розділена на дві половини. Запис одиниці в регістр рядкового драйвера включає відразу два рядки – один у верхній, один в нижній частині. З 8 біт даних, що передаються РКЕ, 4 йдуть на верхню, 4 – на нижню половину. Таким чином, на екрані одночасно засвічуються два рядки, що при тій самій частоті розгортки дає менше мерехтіння зображення. Типова структура такого дисплея приведена на рис. 7.15

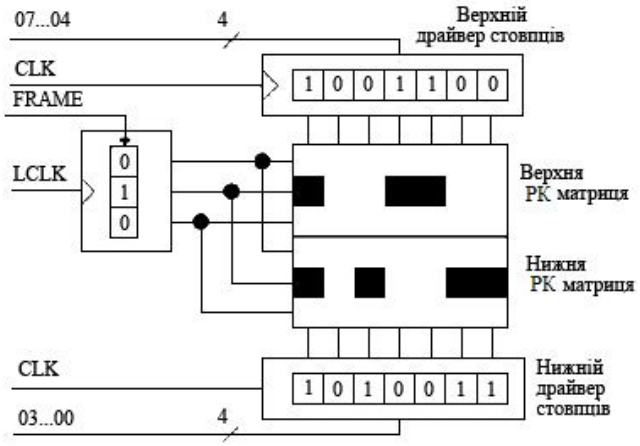


Рис. 7.15.

Більшість контролерів РК дисплеїв мають в своєму складі регістр команд і регістр даних. Обмін з контролером зводиться до запису/зчитування цих регістрів. Контролери розділяються на алфавітно-цифрові і графічні.

7.5. Технологія гнучких дисплеїв

Розвиток ринку мобільних пристроїв викликав активну розробку нових компонентів, матеріалів, технологій, які забезпечують зменшення розмірів пристроїв, споживаної потужності, а також ціни. Гнучкі електронні пристрої є новітнім продуктом сучасних технологій.

Пристрої відображення інформації, що належать до класу гнучких дисплеїв, можна умовно розділити на декілька категорій:

– плоскі і тонкі дисплеї, в яких корисна властивість гнучкості є побічною і реалізована для досягнення малої ваги, товщини і вартості. Такі дисплеї стійкі до механічного впливу;

– дисплеї, які в процесі експлуатації піддаються короткочасним деформаціям робочої поверхні без погіршення якості зображення. Кут вигину незначний і складає декілька градусів, а радіус вигину складає 60...100 мм;

– дисплеї, в яких екрани мають статичну криволінійну поверхню. Дана поверхня армується один раз при виготовленні дисплея. В процесі експлуатації форма екрану вже не змінюється. У категорії даних дисплеїв можуть бути використані різні дисплейні технології, у тому числі модуляційні (РК-дісплеї) і емісійні (OLED);

– прилади, конструкція яких дозволяє згортати їх в рулон. Електронний пристрій з таким екраном виглядає як два з'єднаних між собою циліндри. Рулонна технологія гнучких дисплеїв може бути використана і для оперативного розгортання екранів великого розміру.

Кожній категорії гнучких дисплеїв відповідають певні різновиди дисплейних технологій. Щоб забезпечити гнучкість екрану дисплея, необхідно використовувати гнучкі і тонкі підкладки.

Для застосування в гнучких дисплеях можна розглядати наступні дисплейні технології:

- РК-дисплеї на основі твістнематичних матеріалів;
- РК-дисплеї на основі холестеричних РК матеріалів;
- бістабільні РК-дисплеї на ефекті ZBD;
- РК-дисплеї на основі капсульованого РК матеріалу;
- E-book (E-ink і Gyricon);
- електрофоретичні, електродепозитні та електрохромні дисплеї;
- світловипромінюючі OLED/PLED технології.

Традиційна твістнематична технологія з пасивною і активною адресацією може бути використана для перших двох категорій гнучких дисплеїв, оскільки вона критична до зміни товщини робочого проміжку РК комірок. Бістабільна технологія і технологія OLED можуть застосовуватися у всіх категоріях гнучких дисплеїв.

Зараз активно розвивається нова технологія E-ink – електроні чорнила. Готові дисплеї, створені на основі даної технології, називаються E-paper – електронний папір. На прозорій підкладці розташовуються

мільйони двобарвних електрично заряджених кульок (рис. 7.16) розміром 30...100 мкм. При діаметрі кульок 30 мкм може бути отримана розрішаюча здатність 300 dpi (dot per inch – точок на дюйм), а при діаметрі 100 мкм – 100 dpi. Половина кульок забарвлені в контрастні кольори, наприклад, білий і чорний, або білий і червоний. Кульки знаходяться у в'язкій рідині – силіконовому еластомірі.

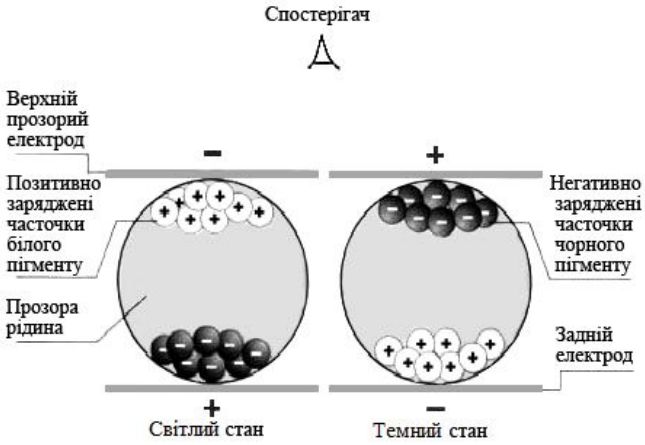


Рис. 7.16.

На рис. 7.17 показана конструкція і принцип роботи дисплея SmartPaper.

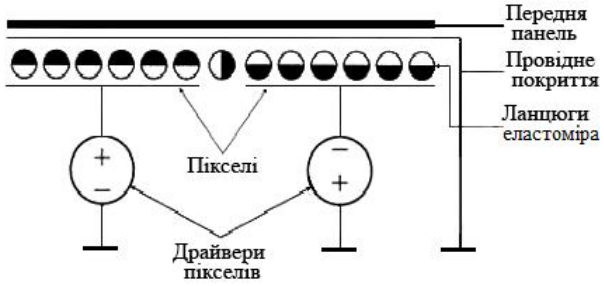


Рис. 7.17.

Прикладена напруга змушує кульку обертатися в електричному полі однією із забарвлених сторін до електроду на передній підкладці. Рівні управляючих напруг досить високі – близько 100 В. На рис. 7.18 представлена структура схеми управління.



Рис. 7.18.

Технології Gyricon і E-ink мають властивість пам'яті (бістабільність), тобто зображення на екрані дисплея може зберігатися необмежено довго без прикладення напруги. На відміну від технології фірми Xerox, в дисплеях E-ink застосовуються крихітні сфери – мікрокапсули, що мають прозору оболонку, заповнену нейтральними частинками блакитного пігменту і негативно зарядженими мікрочастинками білого кольору (рис. 7.19).

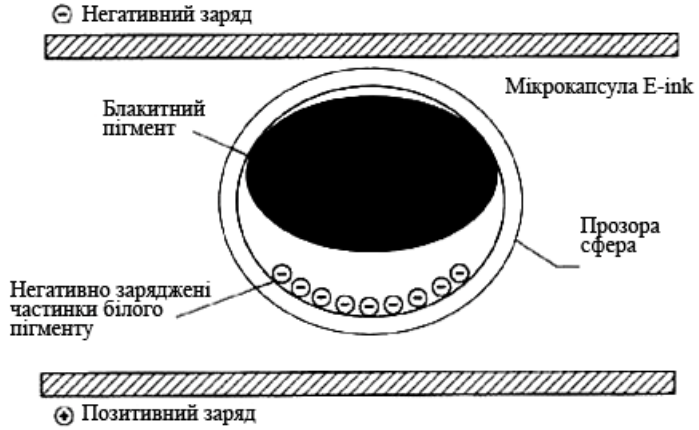


Рис. 7.19.

Плівкові провідники мають достатню гнучкість, проте і в них потрібне створення багатошарової структури, щоб забезпечити довговічність і стійкість до багатократних згинів. Для створення перехідних з'єднань між різними компонентами дисплейної системи використовується перевірена технологія анізотропних провідних плівок. Сучасний рівень даної технології дозволяє отримувати надійне з'єднання з мінімальною відстанню між сусідніми контактами 10 мкм. Зразок гнучкого дисплея показаний на рис. 7.20. Загальна товщина гібридної конструкції досягає 0,9...1 мм.



Рис. 7.20.

Технології гнучких дисплеїв дуже перспективні для збірки мобільних комунікаційних пристроїв і кишенькових комп'ютерів наступного покоління. Планується її використання для адресації

надвеликих набірних гнучких екранів, а також для проведення розробок спеціальних мікромодулів і датчиків.

7.6. Стереоскопічні системи

В даний час для створення умов об'ємного сприйняття в системах віртуальної реальності використовуються найрізноманітніші підходи, з яких найбільший розвиток і поширення отримав підхід, що використовує стереоскопію, через його простоту.

В основу стереоскопічного принципу бінокулярної візуалізації покладений той факт, що зображення на сітківці ока є двовимірним. Якщо «пред'явити» очам два зображення деякої 3D-сцени, отримані з двох точок, відповідних розташуванню оптичних центрів очей, то центральний відділ зорового аналізатора повинен зрештою створити у спостерігача ілюзію спостереження 3D-сцени. Однією з основних технологічних проблем стереоскопії є створення умов для рознесення зображень стереопари по різних оптичних каналах, призначених відповідно для лівого і правого ока, – сепарація полів стереопари. Для цієї мети застосовуються різні оптичні схеми, з яких найбільш поширеними є, просторова, часова, поляризаційна, а також кольорова і схема Пульфріха.

Перераховані схеми, окрім просторової, об'єднує одна особливість – для спостереження стереокартинки потрібні різні окуляри, тип яких залежить від способу сепарації. Схема з просторовою сепарацією відрізняється від останніх тим, що поля стереопари розташовуються на фізично різних носіях зображення, а для візуалізації необхідна додаткова лінзова або дзеркально-лінзова система. У всіх останніх схемах для виведення зображень спільно використовується один носій. Останніми роками все більшого поширення набувають і безокулярні системи, в яких оптичні пучки, відповідні полям стереопари, розділяються в просторі таким чином, що досягають лівого і правого очей.

Найвідомішим пристроєм, в якому використовується просторова сепарація, є стереоскоп, принцип дії якого був покладений в основу всіх шлем-дисплейних систем. Зображення, що створюють стереопару (поля стереопари) розташовуються на двох різних носіях, або на одному носії, але розділеному на дві половинки. Принциповим є те, що зображення не перекриваються і можуть незалежно підводитися до очей по різних оптичних каналах.

Основний недолік цієї схеми до останнього часу полягав в тому, що через великі геометричні розміри носіїв зображення (дисплеїв) відстань між геометричними центрами зображень істотно перевищувала міжзіничну відстань. Це приводило до необхідності застосування дзеркальних систем для зведення оптичних осей. В даний час, завдяки розвитку LCD/OLED - технологій з'явилася можливість мініатюризації дисплеїв і заміни шлем-дисплейних конструктивів на окулярні оправи.

У системах з часовою сепарацією на один і той самий носій (екран дисплея) послідовно виводяться праве і ліве зображення стереопари. Синхронно з виведенням зображень послідовно відкриваються і перекриваються праве і ліве поля зору. Для перекриття використовуються високошвидкісні оптичні затвори, найчастіше рідкокристалічні, розміщені в окулярній оправі. Для синхронізації застосовується спеціальна електронна або електронно-оптична схема.

При електронній синхронізації використовуються сигнали кадрової розгортки. Відеокарта повинна мати відповідну підсистему синхронізації та інтерфейс для підключення затворів. При електронно-оптичній синхронізації зображення стереопари повинні містити спеціальні мітки. Ці мітки розташовуються на периферії полів стереопари і призначені для розпізнавання фотодатчиками, пов'язаними із затворами. Таким чином, відеокарта може бути звичайною, а підсистема синхронізації автономною.

Основний недолік систем з часовою сепарацією полягає в їх складності, оскільки потрібні додаткові пристрої й синхронізація їх роботи. Окрім цього монітор, на який послідовно виводяться ліве і праве зображення, що взаємно перекриваються, повинен мати спеціальні характеристики, зокрема, досить високу частоту кадрової розгортки і низьку інерційність післясвітіння кінескопа.

Найчастіше для аудиторної стереовізуалізациі на великих екранах використовуються системи з поляризаційною сепарацією. Використовуються два проектори, забезпечені поляризаційними, орієнтованими ортогонально фільтрами. Обидва зображення одночасно проектуються на екран зі спеціального матеріалу, що володіє мінімальною мірою деполяризації. Для прямої і зворотної проекції використовуються різні матеріали. Використовуються окуляри з поляризаційними фільтрами, орієнтованими паралельно відповідним фільтрам проекторів, внаслідок чого кожне око отримує призначене для нього зображення. Перевагами поляризаційних стереосистем є можливість спостереження стереозображення одночасно великим числом глядачів, а також відносна простота і дешевизна виготовлення великої кількості поляризаційних стереоокулярів.

У системах з кольоровою сепарацією використовується особливість сприйняття додаткових кольорів зоровим аналізатором людини. В даний час стандартом є пара «червоний-синьозелений» («red-cyan»). Фільтрація призводить до маскування одного з кольорів пари. Завдяки цій особливості, на одному і тому ж носієві можна з перекриттям розмістити два зображення, створюючі стереопару, але забарвлені в додаткові кольори. Стереозображення потрібно розглядати через кольорові фільтри, що пропускають світлові пучки додаткових кольорів, до різних очей (червоно-сині або червоно-зелені окуляри). Перевагою цієї схеми є

гранична простота пристрою спостереження стереопари. Недоліком є те, що при спостереженні виникає відчуття своєрідної кольорової гамми.

Ефект Пульфріха пов'язаний із запізненням створення зоровим аналізатором образу 3D-сцени в одному з каналів. Запізнення відбувається якщо перед одним оком розміщується затемнюючий світлофільтр, а друге око сприймає сцену без затемнення. Таким чином, якщо сцена рухається відносно спостерігача в такому напрямі, що око із затемнюючим фільтром бачить сцену раніше, ніж око без такого фільтру, єдиний образ відтворюється за двома зображеннями, які за певних умов можна вважати стереопарою.

В даний час найбільш поширеними є два варіанти автостереоскопічних систем – лінзо-растрові і апертурно-растрові. Якщо створити зображення з вертикальних растрових смужок, що чергуються, на кожній з них буде розміщений фрагмент зображення, призначений «своєму оку», і крім цього перекрити кожному оку «чуже» зображення струнами апертурної гратки, то очі побачать «свої» поля стереопари. При цьому, кожне зображення буде «проріджене» чорними вертикальними смужками. Апертурна гратка (задній бар'єр паралаксу) використовуються для створення вертикальних тіней від плоскої освітлювальної лампи на растрі РК-екрану. Таким чином, вона розміщується між прозорим РК-екраном і плоскою освітлювальною лампою. Стереодисплей з бар'єром паралаксу може функціонувати в двох режимах – моно і стерео. У стереорежимі (рис. 7.21) бар'єр паралаксу, що є ще одним додатковим РК-екраном з вертикальними смужками, активований. При цьому смужки непрозорі і створюють тіні від лампи в потрібних місцях основного екрану.

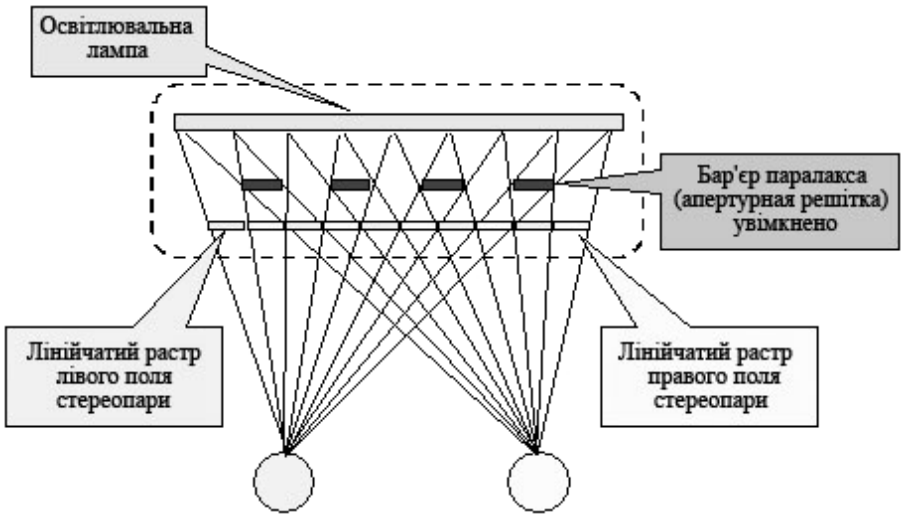


Рис. 7.21.

У монорежимі задній РК-екран повністю прозорий (тіні на основному екрані відсутні), і спостерігається моноскопічне зображення (рис. 7.22).

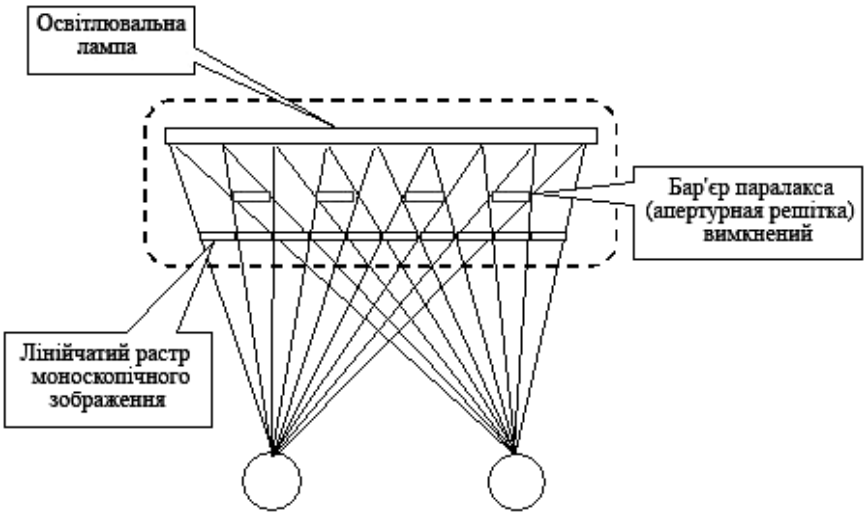


Рис. 7.22.

7.7. Системи з 3D-розгорткою

Використання голографії для 3D-візуалізації має великі перспективи і одночасно великі перешкоди у вигляді труднощів технологічного характеру. Зокрема, необхідно мати роздільну здатність носія зображення, сумірну з довжиною хвилі; для візуалізації динамічних сцен в голограми необхідно передавати величезні обсяги інформації. Тому голографічні системи для 3D-візуалізації застосовуються досить рідко (для статичних сцен).

Принцип візуалізації в системах з 3D-розгорткою заснований на побудові зображення 3D-об'єкту лазерним променем на рухливому носієві. В якості рухливих носіїв можуть використовуватися найрізноманітніші об'єкти – мембрани, що коливаються, дзеркала, поверхні, що обертаються, рідини і гази, що містять дрібні частки, що розсіюють світло (пара, дим, аерозолі, рідини).

Незважаючи на те, що системи з 3D-розгорткою набагато простіші від голографічних з точки зору реалізації, в даний час вони є, в основному, експериментальними лабораторними зразками, і не набули масового поширення.

Питання до розділу 7

1. Принципи побудови світлодіодних екранів.
2. Що є основним елементом екрану?
3. В чому полягає різниця між фізичним та віртуальним пікселем?
4. Назвіть переваги світлодіодних екранів.
5. Основні світлотехнічні параметри світлодіодних екранів.
6. Будова плазмових екранів.
7. В чому полягає перевага модульного принципу побудови великих екранів?
8. Назвіть основні технології побудови гнучких дисплеїв.
9. Поясніть принцип роботи стереоскопічних систем.

8. ТЕЛЕВІЗІЙНІ ЗАСОБИ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ

8.1. Будова електронно-променевої трубки

Електронно-променеві трубки (ЕПТ) є найбільш поширеним і важливим пристроєм в системах відображення інформації.

Фізичний принцип роботи ЕПТ заснований на створенні керованого сфокусованого пучка електронів, що падає на покритий люмінофором екран і викликає світіння його окремих ділянок.

Найбільш широке використання в сучасних дисплеях знайшли монохромні і кольорові ЕПТ. Електронно-променеві трубки поділяються на:

– ЕПТ з електростатичним фокусуванням і електростатичним відхиленням електронного променя;

– ЕПТ з магнітним фокусуванням і магнітним відхиленням променя;

– ЕПТ з електростатичним фокусуванням і магнітним відхиленням електронного променя (комбіновані).

8.1.1. Монохромні ЕПТ

На рис. 8.1 показано будову монохромної електронно-променевої трубки з електростатичним управлінням.

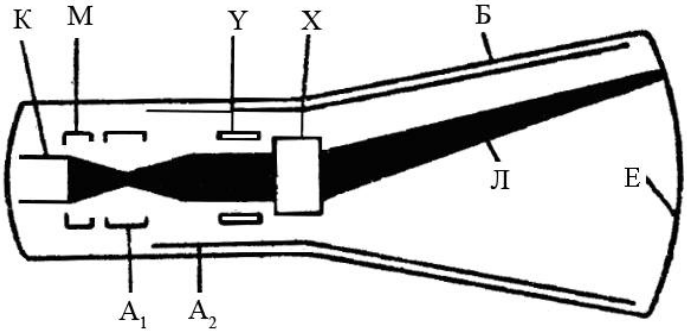


Рис. 8.1.

На рисунку позначені: Б – вакуумний балон; К – катод; М – модулятор; A_1 та A_2 – перший і другий аноди (другий анод виконують у вигляді шару аквадагу, що покриває внутрішню поверхню ЕПТ в частині, що приєднується до екрану; X – горизонтальні відхиляючі пластини; Y – вертикальні відхиляючі пластини; Е – екран.

На рис. 8.2 представлений устрій монохромної електронно-променевої трубки з магнітним управлінням.

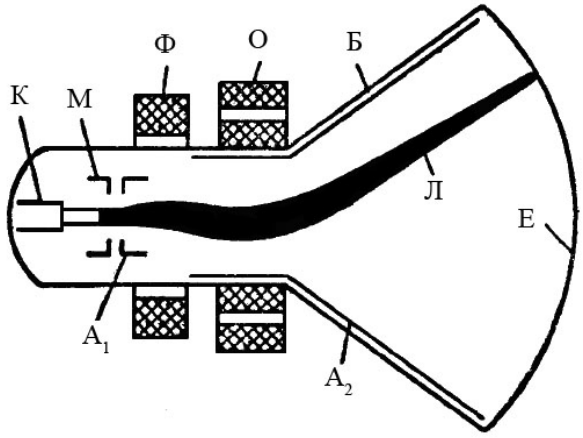


Рис. 8.2.

На рисунку позначені: Ф – фокусуюча котушка; О – відхиляюча котушка; Л – електронний потік; Б – вакуумний балон; К – катод; М – модулятор; A_1 та A_2 – перший і другий аноди.

Електронний потік, що випромінюється катодом, після фокусування в електронно-оптичній системі (A_1 і Ф) в ЕПТ з електростатичним управлінням, або фокусування за допомогою фокусуючої котушки (Ф) в ЕПТ з електромагнітним управлінням, набуває форми електронного променя, і під дією сигналів, прикладених до X і Y пластин, або до відхиляючих котушок, послідовно спрямовується в задані ділянки люмінесцентного екрану для створення зображення.

Конструктивно ЕПТ представляє собою вакуумний балон Б – вузький скляний циліндр з розширенням на кінці до прямокутного або круглого екрану, на внутрішню стінку якого нанесений шар люмінофора. У задній частині трубки розташовується електрона гармата, до складу якого входять оксидний катод К, управляючий електрод, модулятор (циліндр з торцевим отвором і з негативним потенціалом відносно катода) для формування електронного променя з потоку електронів і управління площиною променя (яскравістю), перший і другий аноди A_1 та A_2. До першого анода A_1 прикладений позитивний потенціал в декілька сотень вольт, що прискорює електронний промінь. До другого анода A_2 прикладена висока напруга в декілька десятків кіловольт, тому електрони набувають дуже велику швидкість.

Електронно - променева гармата дозволяє отримати сфокусований промінь електронів великої швидкості. Після електронної гармати електронний промінь потрапляє в електричне поле, яке створюється відхиляючою системою електродів з двох пар взаємно перпендикулярних пластин X та Y. Відхиляюча система керує положенням електронного променя в просторі. Пара пластин X та Y утворюють плоскі конденсатори. Горизонтальні пластини Y відхиляють промінь у вертикальній площині, а вертикальні пластини X – в горизонтальній площині.

Перевагою ЕПТ з електростатичним відхиленням є висока швидкість роботи. ЕПТ з магнітним управлінням застосовуються в телебаченні (кінескопи), а також в радіолокаційних системах і в засобах відображення інформації. До складу електронного прожектора входить катод К, модулятор М і анод $А_1$. На вакуумну трубку зовні одягають фокусуючу котушку Ф і котушки О. Останні використовуються для відхилення променя по горизонталі та вертикалі. ЕПТ з магнітним управлінням конструктивно простіші і, крім того, забезпечують краще фокусування електронного променя.

8.1.2. Кольорові ЕПТ

Для отримання будь-якого кольору в ЕПТ використовують принцип змішування (принцип Гельмгольца) трьох основних кольорів – червоного, зеленого і синього (RGB) в певних пропорціях:

$$C_1 = R_1(R) + G_1(G) + B_1(B),$$

де G_1, R_1, B_1 – вагові коефіцієнти основних кольорів.

У кольорових ЕПТ екран має дискретну структуру. При цьому люмінофори трьох основних кольорів наносяться або у вигляді дуже маленьких плям круглої форми, об'єднаних в тріади, або у вигляді вертикальних смуг, що чергуються. Для розділення електронних променів за трьома основними кольорами застосовують маску.

На рис. 8.3 показані основні елементи конструкції кінескопа з дельтовидним розташуванням електронних гармат.

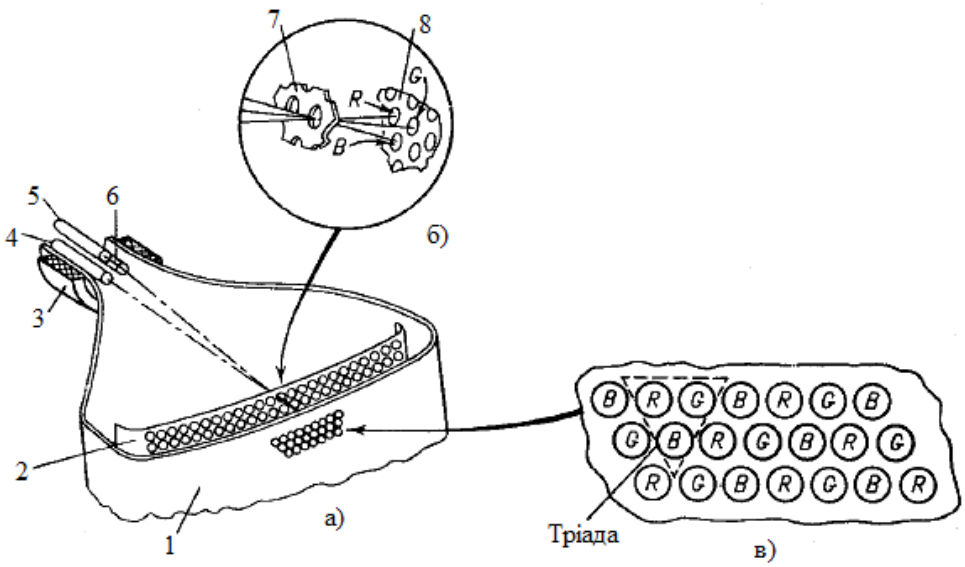


Рис. 8.3.

На рисунку позначені: 1 – екран; 2 – тіньова маска; 3 – відхиляюча система; 4 – зелена гармата; 5 – синя гармата; 6 – червона гармата; 7 – отвори в масці; 8 – люмінофорні елементи.

Кожна з електронних гармат не відрізняється від гармат, які використовуються в монохромних ЕПТ і здійснює генерацію, фокусування і прискорення променя. Всередині ЕПТ гармати зорієнтовані таким чином, що їх промені лежать в одній площині під деяким кутом один до одного, і проходячи через будь-який з отворів в масці, потрапляють кожен на смугу люмінофора лише свого кольору. Відхилення всіх трьох променів здійснюється за допомогою однієї котушки у відхиляючій системі. З відстані, на якій видно зображення, дозволяюча здатність ока така, що не дозволяє розрізнити дрібні розміри люмінофорних елементів. В результаті, світіння елементів люмінофорної тріади, що виникає в процесі електронного бомбардування, сприймається людським оком як сумарне. Колір залежить від пропорцій основних кольорів і може бути будь-яким в області видимого спектру. Пропорції можна міняти, управляючи напругою модулятора незалежно в кожній з гармат.

8.2. Види масок, що використовуються в ЕПТ

Електронний промінь, призначений для червоних люмінофорних елементів, не повинен впливати на люмінофор зеленого або синього кольору. Щоб досягти такої ефекту використовується спеціальна маска.

Shadow Mask (тіньова маска, дельтовидна технологія). У даній технології кольоровий елемент складається з трьох зерен (рис. 8.4), розташованих у вершинах правильного трикутника. Різкість зображення визначається відстанню між геометричними центрами сусідніх елементів.

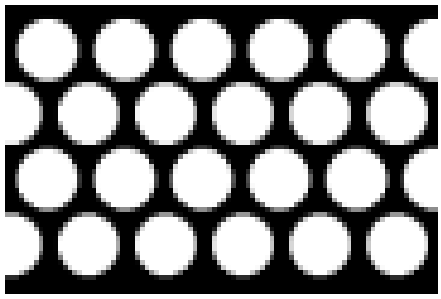


Рис. 8.4.

Основна перевага дельтовидної технології: точна чітка картинка, діагональні лінії не мають спотворень. Основним же недоліком вважається велика відстань між зернами, тому такі трубки мають не дуже насичений колір.

Тіньова маска (shadow mask) – це найпоширеніший тип масок для CRT-моніторів. Тіньова маска складається з металевої сітки перед частиною скляної трубки з люмінофорним шаром. Як правило, більшість

сучасних тіньових масок виготовляють з інвару (invar, сплав заліза і нікелю). Отвори в металевій сітці працюють як приціл (хоча і не точний), саме цим забезпечується те, що електронний промінь потрапляє лише на необхідні люмінофорні елементи, і лише в певних областях.

Мінімальна відстань між люмінофорними елементами однакового кольору називається dot pitch (крок точки), і є важливим параметром якості зображення. Крок точки зазвичай вимірюється в міліметрах (мм). Чим менше значення кроку точки, тим вища якість відтвореного на моніторі зображення.

Slot Mask (щілинна маска). Щілинна маска (slot mask) – це технологія, яка широко застосовується компанією NEC під назвою "CromaClear". В даному випадку люмінофорні елементи розташовані у вертикальних еліптичних комірках (рис. 8.5), а маска зроблена з вертикальних ліній. Фактично, вертикальні смуги розділені на еліптичні, або прямокутні комірки, які містять групи з трьох люмінофорних елементів трьох основних кольорів. Мінімальна відстань між двома комірками називається slot pitch (щілинний крок). Чим менше значення slot pitch, тим вища якість зображення на моніторі.

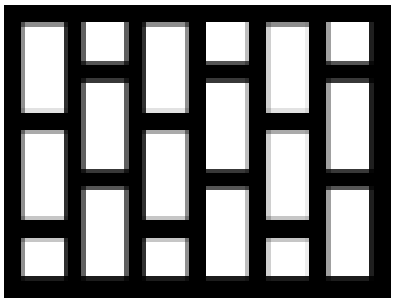


Рис. 8.5.

Переваги щілинної маски: більш насичений колір, ніж при використанні дельтовидної технології, за рахунок більшої площі люмінофора, що світиться, проте така маска декілька програє в чіткості ліній.

Щілинна маска використовується, окрім моніторів від NEC (де комірки еліптичні), в моніторах Panasonic з трубкою PureFlat (раніше називалися PanaFlat). LG використовує плоску щілинну трубку Flatron з кроком 0,24 в своїх моніторах. Відмітимо, що в плоских трубках Infinite Flat Tube (серія DynaFlat) від Samsung використовується не щілинна маска, а звичайна тіньова.

Aperture Grill (апертурна гратка). Апертурна гратка (aperture grill) – це тип маски, який використовується різними виробниками в своїх технологіях для виробництва кінескопів, що носять різні назви, але що мають однакову суть, наприклад, технологія Trinitron від Sony, або Diamondtron від Mitsubishi. Це рішення не включає металеву гратку з

отворами, як у випадку з тіньовою маскою, а має гратку з вертикальних ліній (рис. 8.6). Замість точок з люмінофорнимі елементами трьох основних кольорів, апертурна гратка містить серію ниток, що складаються з люмінофорних елементів, побудованих у вигляді вертикальних смуг трьох основних кольорів. Така система забезпечує високу контрастність зображення і хорошу насиченість кольорів, що разом забезпечує високу якість моніторів з трубками на основі цієї технології. Маска, що використовується в трубках фірми Sony (Mitsubishi, ViewSonic), є тонкою фольгою, на якій розташовані тонкі вертикальні лінії. Вона тримається на горизонтальному (одном в 15", двох в 17", трьох і більше в 21") дроті, тінь від якої користувач може бачить на екрані у вигляді тонкої смуги (або смуг), що пересікають по діагоналі екран. Цей дріт застосовується для гасіння коливань і називається damper wire. Його добре видно, особливо при світлому фоні зображення на моніторі. Деяким користувачам ці лінії принципово не подобаються, інші ж, навпаки, задоволені і використовують їх як горизонтальну лінійку.



Рис. 8.6.

Як і в будь-якій іншій технології, в апертурної гратки є свої переваги і недоліки. Основна перевага цієї технології: прекрасні за насиченістю кольори; екран таких моніторів менш опуклий, ніж при застосуванні тіньових масок і щілинних масок, оскільки в принципі не має причин бути викривленим по вертикалі. Проте, сучасні технології виготовлення плоских трубок, дозволяють і при вживанні тіньової або щілинної маски досягти плоского екрану. Основний же недолік – ступінчастість і нечіткість діагональних ліній, тому проектувальникам не подобаються монітори із застосуванням апертурної гратки.

8.3. Типи спотворень, що виникають в ЕПТ

Залежно від типу маски, якості управляючої електроніки, налаштувань, на екрані телевізорів і моніторів, в яких використовуються електронно-променеві трубки, можливі появи різного роду спотворень.

Перевірка фокусування: дуже важливо, щоб електронні гармати були правильно сфокусовані, як в центрі екрану, так і по кутах (рис. 8.7). Саме місця в кутах екрану є проблемними. Букви темного тексту, виведеного на світлому фоні в центрі і в кутах екрану, мають бути чіткими і добре читатися, а на краях екрану пікселі не повинні розмазуватися або

двоїтися. Дуже добре всі недоліки видно на строчкових буквах "e" та "m", в ідеальному випадку, вони повинні добре читатися в будь-якому місці екрану.

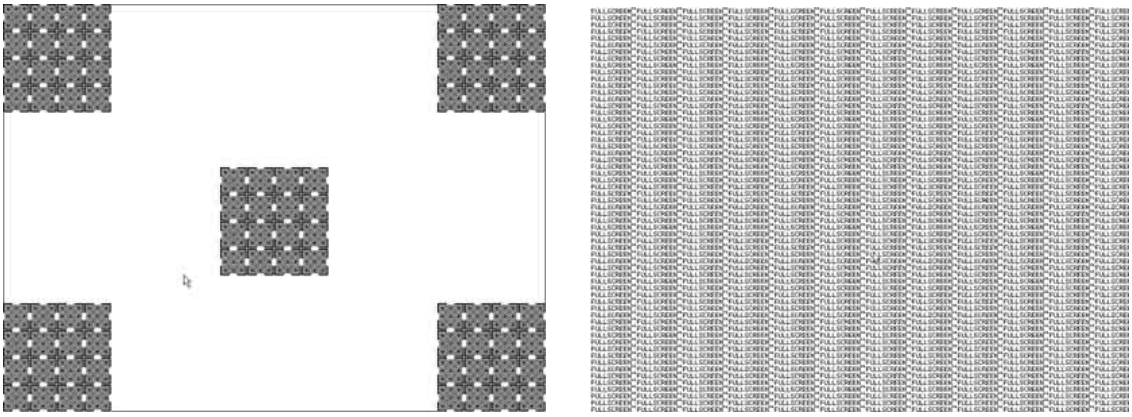


Рис. 8.7.

Білі лінії, що відображаються на чорному фоні (рис. 8.8), повинні залишатися білими уздовж країв екрану. Якщо на лінії з'являються смуги іншого кольору, в цьому випадку відтворення на даному екрані дрібних об'єктів, таких, як символи або лінії, може бути посереднім. При цьому, навіть якщо кольорові смуги присутні, то монітор все одно може відповідати специфікаціям виробника. Якщо кольорові смуги кожного разу виявляються по-різному і в різних місцях, то, швидше за все, монітор не відповідає специфікаціям, проте, прояв кольорових смуг на межах екрану властивий більшості моніторів.

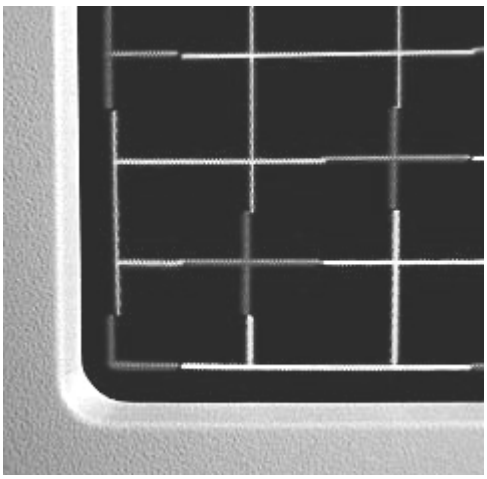


Рис. 8.8.

Перевірка подушки. Якщо межі зображення відхиляються від прямої лінії, значить, у монітора присутнє спотворення у вигляді подушки, або бочки. Бочкоподібне спотворення є наслідком неправильного (надмірного)

використання корекції подушкоподібності, тобто краї зображення опуклі назовні. Якщо в моніторі передбачена можливість корекції подушкоподібності (pincushion), то можна спробувати виправити положення. Якщо такої можливості немає, або якщо корегування не допомогло, то на екрані монітора будуть присутні геометричні спотворення, інколи досить значні. Варто відмітити, що зміна дозводяючої здатності, або частоти регенерації, можуть впливати на наявність подушкоподібних спотворень: вони або можуть зовсім зникнути, або посилитися.

Геометричні спотворення. Якщо розміри вікна змінюються в різних частинах екрану, значить, присутнє геометричне спотворення, яке можливо, не можна виправити, особливо якщо в моніторі не передбачені змінні параметри налаштування геометрії в достатній кількості (рис. 8.9).

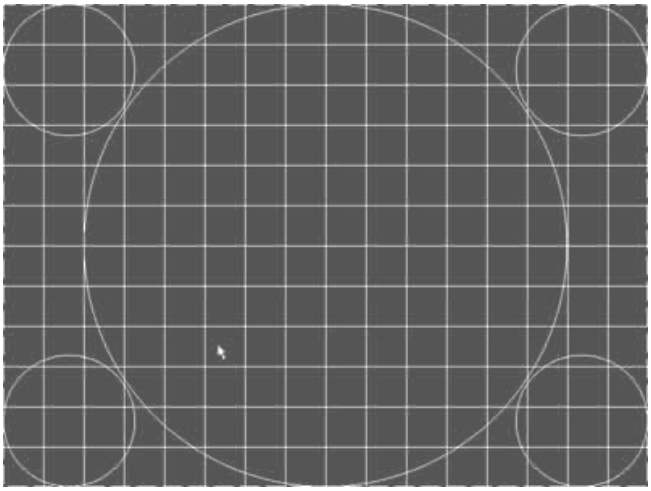


Рис. 8.9.

Передача кольорів. Відобразивши на екрані чистий червоний, зелений і синій кольори, слід поглянути на те, як ці кольори відображаються на екрані, і якщо колір відображається неправильно, то в монітора погана передача кольорів.

Рівномірність засвічення. При виводі на екран повністю білого зображення яскравість має бути рівномірною по всій площі, і не повинно бути помітно жодних явних кольорових, або темних плям.

Муар. Муар, або комбінаційне спотворення, проявляється на фоні або навколо об'єктів у вигляді контурів ліній, хвиль, і т.п. Муар – це явище природної інтерференції, яке характерне для всіх CRT-моніторів. Муар залежить від розрішаючої здатності і розміру монітора, і краще всього помітний саме при високих значеннях розрішаючої здатності на моніторах з добре сфокусованими променями. Якщо видно муар, значить монітор добре сфокусировано, але це неприємно. Якщо муару взагалі ніколи не немає, це означає, що в монітора погане фокусування. В деяких моніторах передбачено регулювання муару, що дозволяє зробити його непомітним.

8.4. Программні засоби тестування моніторів

Для тестування якості зображення, що виводиться на екран монітора, можна використовувати спеціальні утиліти, найвідоміша з яких Nokia Monitor Test від відомого виробника моніторів. На рис. 8.10 представлено головне вікно програми для тестування якості зображення, що виводиться на екран.

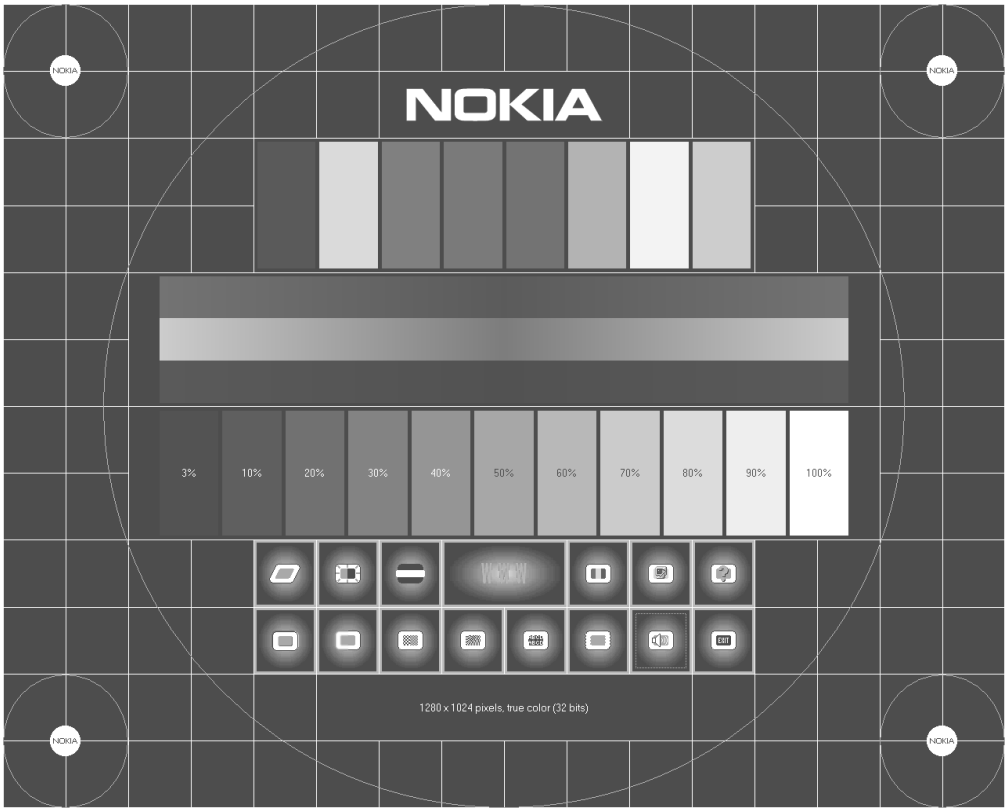


Рис. 8.10.

Програма дозволяє перевірити монітор на наявність спотворень. У нижній частині вікна програми розташовані кнопки, натиснення на які призводить до появи тестового зображення, на якому найсильніше виявляються спотворення. Програма дозволяє перевірити монітор на наявність наступних дефектів:

– геометричні спотворення;
– яскравість і контрастність;
– передача кольорів;
– фокусування;
– розрішаюча здатність;
– муар;
– подушка;
– рівномірність засвічування.

Програма безкоштовна і доступна в мережі Internet.

8.5. Принципи організації розгорток в ЗВІ

ЗВІ на основі ЕПТ поділяють на знакомоделюючі та знакогенеруючі. Знакомоделюючі ЗВІ в площині екрану ЕПТ формують електронний промінь, перетин якого має конфігурацію вибраного знаку. Такі системи не набули широкого практичного поширення.

У переважній більшості випадків використовують знакогенеруючі ЗВІ, в яких перетин електронного променя набуває форми кола малого діаметру. У кожний момент часу промінь займає певне положення на екрані. Повне зображення формується послідовно в часі відтворенням всіх його елементів. Послідовне в часі поелементне відтворення називається розгорткою зображення. Зображення, отримане в процесі розгортки сукупністю окремих елементів, називається кадром.

За принципом організації розгортки зображення, методи формування знаків діляться на растрові й функціональні. При растрових методах закон розгортки, а отже, і траєкторія переміщення променя по екрану, не залежать від формованої інформаційної моделі. Формування зображення здійснюється модуляцією яскравості променя при його проходженні відповідних елементів інформаційної моделі. При функціональних методах траєкторія переміщення променя збігається з контурами елементів інформаційної моделі, що відображаються, тобто закон розгортки в цьому випадку визначається інформацією, яка відображається. Прийнята розгортка визначає структуру кадру, яку називають растром.

Більшого поширення набув телевізійний растр, який формують двома способами: за допомогою прогресивної і черезрядкової розгорток.

На рис. 8.11 показаний телевізійний растр, утворений лінійною прогресивною розгорткою, при якій повний растр утворюється за один період кадрової розгортки T_k (1, 2 – зворотній хід строкової та кадрової розгорток відповідно).

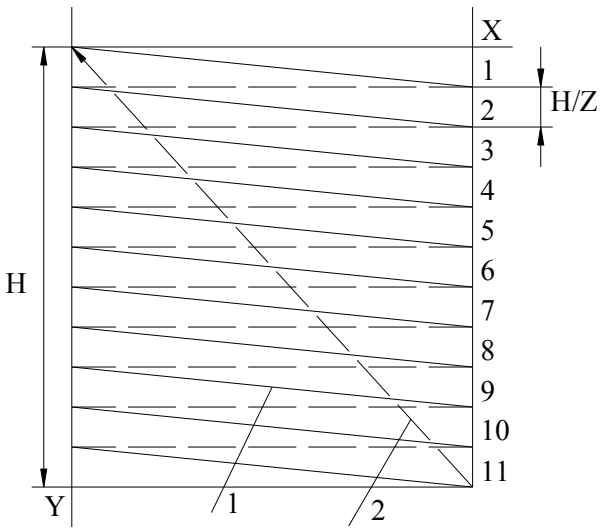


Рис. 8.11.

Строкова і кадрова розгортки здійснюються за допомогою сигналів U_x та U_y, що відхиляють промінь, – напруг для електростатичної відхиляючої системи, або струмів I_x та I_y для магнітної. Відхиляючі сигнали формуються генераторами строкової й кадрової розгорток.

Частота кадрової розгортки $f_k=1/T_k$ для ЕПТ з малим часом післясвітіння має бути більше критичної частоти мерехтіння. Зазвичай частоту f_k обирають рівною частоті мережі змінного струму, виключаючи цим ефект переміщення по екрану створюваної нею перешкоди 50 Гц. Частоту f_z і період T_z строкової розгортки ($f_z=1/T_z$) вибирають з умови:

$$f_z = Z \cdot f_k$$

де Z – число телевізійних рядків в кадрі, що визначає розрішаючу здатність ЗВІ по вертикалі. В телевізійному стандарті прийнято Z=625.

Період строкової розгортки T (рис. 8.12) включає час прямого ходу променя по рядку T_{zn}, і час зворотнього ходу T_{zo}. Зображення формується за час прямого ходу строкової розгортки.

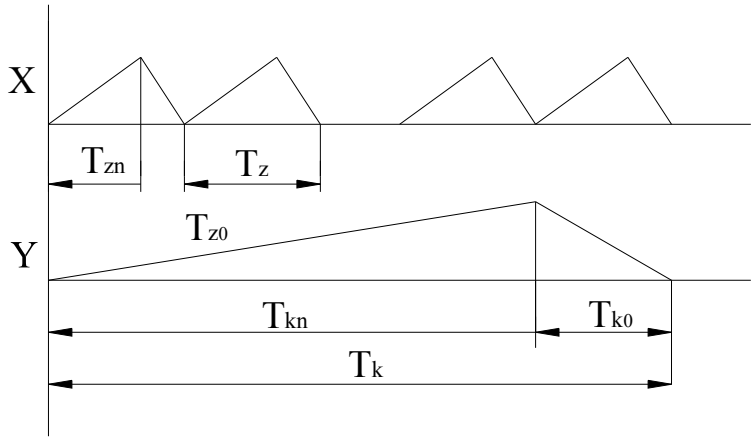


Рис. 8.12.

На рис. 8.13 показаний телевізійний растр, утворений черезстроковою розгорткою, яка передбачає формування одного кадру зображення з двох полів, що передаються послідовно (1 – парні рядки; 2 – непарні рядки).

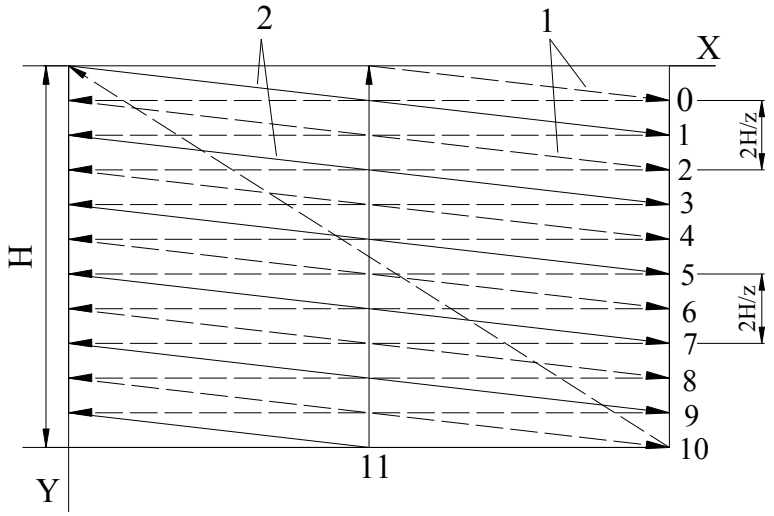


Рис. 8.13.

У першому полі прокреслюються непарні, а в другому – парні рядки растру (останні на малюнку показані штрих-пунктирними лініями). При черезстроковій розгортці кадр передають в два етапи: спочатку лише непарні, а потім парні рядки.

Зменшення частоти кадрів в два рази, в порівнянні з прогресивною розгорткою, при тому ж числі телевізійних рядків в кадрі призводить до двократного зменшення частоти строкової розгортки і необхідної смуги пропускання відеопідсилювача. Черезстрокова розгортка використовується в більшості телевізорів.

Для прогресивної розгортки частота строкової розгортки при формуванні телевізійного растру з числом рядків в кадрі Z=625 дорівнює 31250 Гц, а для черезстрокової розгортки – 15625 Гц.

8.6. Алфавітно-цифрові дисплеї растрового типу

Персональні комп'ютери оснащуються виключно растровими дисплеями. В кожному персональному комп'ютері є відеосистема або відеоадаптер, призначений для формування зображення, що виводиться на екран монітора. Це спеціалізовані схеми для генерування електричних сигналів, які управляють монітором. На платі відеоадаптера (або відеокарті) встановлені мікросхеми ПЗП знакогенератора, програмованого контролера (пристрої управління) ЕПТ, відеопам'яті (відеобуфера) та ін. Контролер призначений для генерування сигналів горизонтальної і вертикальної синхронізації, формування на екрані курсора. В текстовому режимі роботи відеопам'яті екран розділяють на окремі символьні позиції – знакомісця (character positions або character boxes), в кожній з яких виводиться один символ. Позиція знакомісця визначається двома координатами: номер текстового рядка (координата Y) і номер текстового стовпця (координата X). Початок координат (0,0) знаходиться у верхньому лівому кутку робочої області екрану.

У відеобуфері кожному знакомісцю відповідає два байти. Байт з парною адресою містить код символу, тобто він визначає що саме виводиться в даній символьній позиції. Сусідній байт з непарною адресою містить атрибути, що визначають, як формується зображення символу (яскравий, мерехтливий і т.д.). Байти, що містять коди символів і атрибути, розміщуються у відеобуфері послідовно. Адаптер зчитує їх за допомогою знакогенератора і перетворює у відеосигнал.

Структура растрових аналого-цифрових дисплеїв (АЦД) визначається в першу чергу необхідністю перетворення інформації, що зберігається в автономній пам'яті, в імпульсний відеосигнал по кожному рядку растру, синхронно з рухом променя по екрану. Регістр зсуву призначений для зберігання і перетворення команд в процесі їх виконання. На рис. 8.19 показані основні вузли АЦД та їх взаємозв'язки.

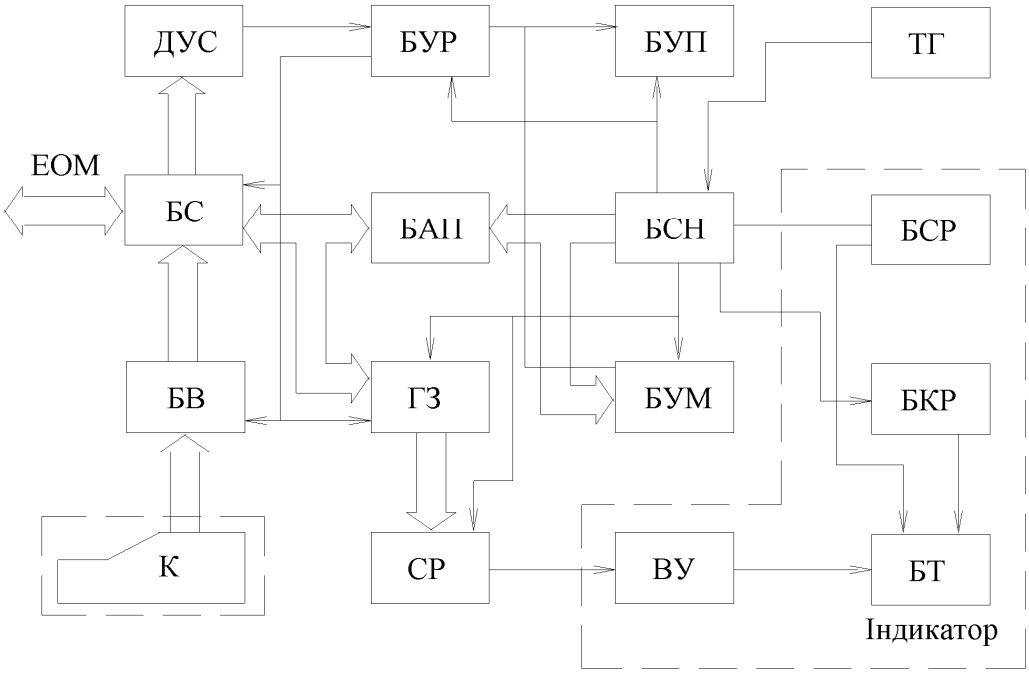


Рис. 8.14.

Найважливішим блоком АЦД є відеобуфер – блок автономної пам'яті. В ньому запам'ятовуються коди символів, що передаються з клавіатури або з ЕОМ через блок вводу (БВ) і блок сполучення (БС). За час горизонтальної розгортки електронного променя коди всіх символів текстового рядка, що відображається в даний момент, з відеобуфера послідовно поступають на вхід знакогенератора (ГЗ). Він здійснює перетворення вхідної інформації (коду символа і поточного номера рядка знакомісця) в паралельний код, одиниці в розрядах якого несуть інформацію про виведені на екрані в даному рядку знакомісця точки. Знакогенератор будується на базі ПЗП або набору логічних схем, здійснюючи таким чином комбінаційний спосіб перетворення інформації. Код з виходу знакогенератора надходить до регістра зсуву (ЗР), де

здійснюється його перетворення в послідовність імпульсів, керуючих через відеопідсилювач інтенсивністю світіння електронного променя. Узгодження роботи пам'яті, знакогенератора, регістра зсуву з розгорткою здійснюється блоком синхронізації (БС), який є набором схем, що працюють від загального тактового генератора, побудованого з використанням кварцевої стабілізації частоти. Частота сигналу, що формується тактовим гнератором (ТГ), визначається вибраною відстанню між суміжними точками на рядку растру, і швидкістю руху променя. У блоці синхронізації (БСН) здійснюється перерахунок кількості текстових знаків і рядків, кількості точок і рядків розкладання символу, формуються сигнали синхронізації строкової (БРР) і кадрової розгортки (БКР).

8.7. Графічні ЗВІ телевізійного типу

Графічні ЗВІ призначені для формування графічної інформаційної моделі (ГІМ). Вони широко застосовуються як термінальні пристрої ЕОМ для виведення результатів обробки інформації у формі графіків, схем, діаграм, гістограм, в системах автоматизованого проектування інтегральних схем, радіоелектронній та іншій апаратурі, в системах управління виробництвом і технологічними процесами.

Графічні інформаційні моделі можуть бути статичними і динамічними. Статична модель не змінюється при сприйнятті її оператором. У динамічній моделі частина, або все зображення, змінюється в процесі сприйняття оператором. Формування ГІМ полягає в синтезі моделі з окремих елементів відображення (ЕВ). ЕВ, що використовуються, визначають спосіб апроксимації ГІМ, і, відповідно спосіб кодування інформаційної моделі (ІМ) і структуру ЗВІ.

За типом елементів відображення (ЕВ), що використовуються, розрізняють ЗВІ з точковими ЕВ, які називають повнографічними, і ЗВІ зі збільшеними графічними елементами – графемами, які називають ЗВІ з обмеженою графікою, або квазіграфічними. Графічні ЗВІ зі збільшеними графічними елементами використовують обмежений набір графем, що формуються в межах графічного знакомісця. Графеми кодуються подібно до знаків. При цьому в інформаційне слово вводиться додатковий розряд, що дозволяє відрізняти код знаків від коду графічних елементів.

Перетворювач коду графічної інформаційної моделі містить ПЗП, який зберігає інформацію про графіку всіх графем, що входять в алфавіт, який визначається характером ІМ, які відображається, і необхідною точністю її апроксимації. Так, для відтворення різних графіків, в алфавіт можуть входити відрізки прямих і кривих другого порядку. Для відображення діаграм, географічних карт, алфавіт має бути доповнений двовимірними елементами. Для кусочно-лінійної апроксимації контурних графічних моделей широко використовується набір відрізків прямих, що проходять через точки, розташовані по периметру знакомісця (рис. 8.15).

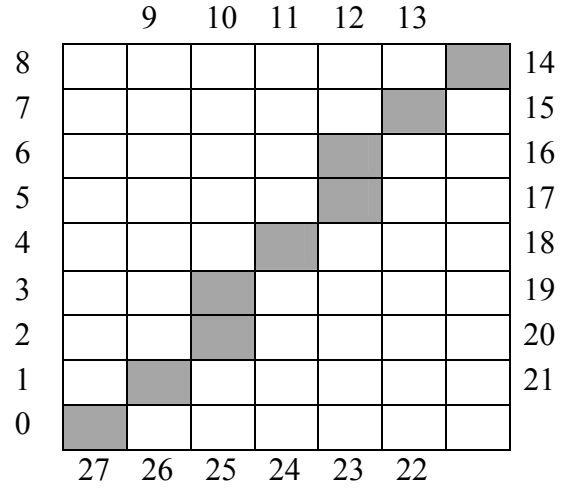


Рис. 8.15.

Для представлення ГІМ сукупністю графем, транслююча програма ЕОМ забезпечує попереднє розбиття зображення на окремі фрагменти, що за форматом і положенням відповідають знакомісцям. Потім відбувається ідентифікація кожного фрагмента з найбільш відповідною графемою, і видається в ЗВІ її код.

Недоліком методу графічних елементів є низька точність апроксимації ГІМ, обумовлена обмеженим набором ГЕ та їх фіксованим положенням на інформаційному полі. Цей недолік частково усувається шляхом організації незалежних символьних і графічних знакомісць. До недоліку методу збільшених графічних елементів слід також віднести необхідність спеціальних транслюючих програм при виведенні графічної інформації з ЕОМ.

8.8. Виведення зображення на екран телевізора

Мікроконтроллер практично без зовнішніх додаткових елементів може виводити на телевізор 16 рядків по 32 символи. Для цього зручно використовувати SPI-інтерфейс для швидкого послідовного виведення зображення. Проте, за допомогою інтерфейсу SPI не можна виводити біти безперервно – між байтами існує розрив (мінімум пів біта). Крім того, SPI після передачі байта залишає вихід в останньому стані. Тому використовується ще одна мікросхема "Виключаюче АБО". В такому разі можна виводити парні байти через інтерфейс SPI, а непарні – через вивід порту. Останній біт SPI компенсується інвертуванням стану сигналу на виводі порту. На рис. 8.16 показана принципова схема пристрою.

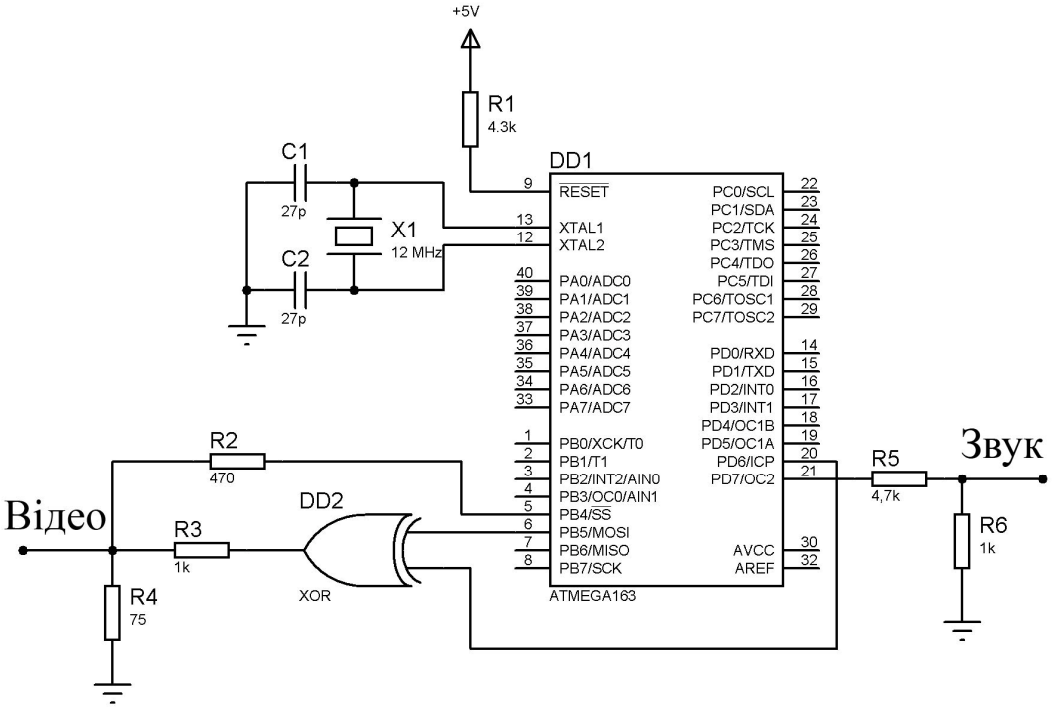


Рис. 8.16.

Процесор працює із збільшеною тактовою частотою (12 МГц). Мікросхему "виключаюче АБО" бажано використовувати імпортну, серії Fast. У такому разі не буде «хвостів» і голок у відеосигналі. Виходи "Відео" і "Звук" під'єднуються до телевізора через стандартні входи типу "тюльпан".

Програма генерує відео з кадровою частотою 50 Гц, 312,5 рядків в кадрі. Весь порт D зайнятий, незважаючи на те, що використовується лише один вивід. У програмі не бажано використовувати інші переривання, оскільки вони можуть заблокувати або затримати переривання від таймера 1, який формує відеосигнал. Текст програми для мікроконтролера представлений нижче. Програма розроблена в компіляторі AVR-GCC 3.3.

```
// файл types.h
#ifndef _TYPES_H
#define _TYPES_H 1
#define FALSE 0
#define TRUE(!FALSE)
typedef unsigned char bool;
typedef unsigned char u8;
typedef   signed char s8;
typedef unsigned int  u16;
typedef   signed int  s16;
typedef unsigned long u32;
typedef   signed long s32;
typedef union
{
        unsigned char byte;
```

```
            struct
           {
                    unsigned char b0 : 1 __attribute__ ((packed));
                    unsigned char b1 : 1 __attribute__ ((packed));
                    unsigned char b2 : 1 __attribute__ ((packed));
                    unsigned char b3 : 1 __attribute__ ((packed));
                    unsigned char b4 : 1 __attribute__ ((packed));
                    unsigned char b5 : 1 __attribute__ ((packed));
                    unsigned char b6 : 1 __attribute__ ((packed));
                    unsigned char b7 : 1 __attribute__ ((packed));
            }bit;
            } flags;

        #define nop() ({asm("nop");})
        #define sei() ({asm("sei");})
        #define cli() ({asm("cli");})
        #endif

// файл video.h

#include <types.h>
#include <font.h>

// оголошення змінних
volatile u8     video_buffer[512];   // відеобуфер
volatile u8     video_mode;          // поточний стан дескриптора переривань
volatile u8*    video_pointer;       // вказівник на поточний рядок
volatile u8     video_line;          // номер поточного стовпця
volatile u8     video_row;           // номер поточного рядка
volatile u8     video_cnt;           // лічильник кадрів
u16 cur_index;                       // положення курсора
void clrscr(void)                    // функція очищення екрану
{
        for (cur_index=0;cur_index<512;cur_index++)
        video_buffer[cur_index]=' ';
        cur_index=0;
}
void printc(u8 num)                  // функція виведення символа
{
        video_buffer[cur_index++]=num;
        if (cur_index==512) cur_index=0;
}
void print_hex(u8 num)               // функція виведення 16-бітного значення
{
        printc(num>>4);
        printc(num&0x0F);
}
// функція виводу рядка в координатах (x,y)
 void print_xys(u8 x, u8 y, const char * str)
{
        cur_index=y*32+x;
```

```
        while(PRG_RDB(str)) printc(PRG_RDB(str++));
}
void gotoxy(u8 x, u8 y)                 // функція встановлення курсора в позицію (x,y)
{
        cur_index=y*32+x;
}
// сигнал частотою = 46875 / тривалість імпульса
void beep (char pulse_time) { OCR2=pulse_time; }
```

// Файл font.h

```
const u8 char_set[4096] __attribute__((section(".tbl")))=
{
0xC7,0xE3,0xC7,0xC7,0xE3,0x1,0xE3,0x1,0xC7,0xC7,0xC7,0x7,0xC7,0x7,0x1,0x1,0xC7,0x11,0x83,0x1,0x11
,0x1F,0x39,0x31,0xC7,0x7,0xC7,0x7,0xC7,0x1,0x11,0x11,0xFF,0xFF,0xFF,0xFF,0xE7,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xE7,0xFF,0xC7,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0,0x11,0x11,0x11,0x1,0xFF,0xFF,0xFF,0xFF,0x11
,0xF1,0x1F,0xFF,0xC7,0xC7,0xF1,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0xDF,0xF0,0x7,0xF8,0x3F,0xFF,0xFF,0x3F,0xF9,0x55,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFB,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0x83,0xE3,0x83,0x83,0xE3,0x1,0xC3,0x1,0x83,0x83,0x83,0x3,0x83,0x3,0x1,0x1,0x83,0x11,
0x83,0x1,0x11,0x1F,0x39,0x31,0x83,0x3,0x83,0x3,0x83,0x1,0x11,0x11,0xFF,0xFF,0xFF,0xFF,0xE7,0x8F,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xC3,0xF
F,0xE7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xE7,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xE7,0xFF,0xFF,0x0,0x11,0x11,0x11,0x1,0xFF,0xFF,0x
FF,0xFF,0x11,0xE1,0xF,0xFF,0xC7,0x83,0xF1,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF
,0xFF,0xFF,0xFF,0xFF,0x0,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xF3,0xE4,0x
80,0x7F,0xFF,0xFF,0x7F,0xFF,0xFF,0xFC,0x1,0xEF,0xE0,0x7,0xE0,0xF,0xFF,0xFF,0x1F,0xF1,0xAA,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFB,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x1,0xC3,0x1,0x1,0xE3,0x1,0x87,0x1,0x1,0x1,0x1,0x1,0x1,0x1,0x1,0x1,0
x11,0x83,0x1,0x11,0x1F,0x11,0x11,0x1,0x1,0x1,0x1,0x1,0x1,0x1,0x11,0x11,0xFF,0xE7,0x99,0xC9,0xC3,0x27,0x
C7,0xE7,0xF3,0xCF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xC3,0xF3,0xC3,0xC3,0xCF,0x81,0xE3,0x81,0xC3,0x
C3,0xFF,0xFF,0xF9,0xFF,0x9F,0xC3,0x81,0xE7,0x83,0xC3,0x87,0x81,0x81,0xC3,0x99,0xC3,0xF9,0x99,0x9F
,0x9C,0x9C,0xC3,0x83,0xC3,0x83,0xC3,0x81,0x99,0x99,0x9C,0x99,0x99,0x81,0xC3,0x9F,0xC3,0x99,0xFF,0
xF3,0xFF,0x9F,0xFF,0xF9,0xFF,0xE1,0xFF,0x9F,0xE7,0xF3,0x9F,0x87,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xCF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF3,0xE7,0xCF,0x8E,0x0,0x11,0x11,0x11,0x1,0xFF,0xFF,0xFF,0
xFF,0x11,0xC3,0x87,0xFF,0xC7,0x1,0xF1,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0x0,0x0,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xF3,0xE4,0xF3,0x
E1,0xF0,0xFF,0xBF,0xFF,0xFF,0xF8,0x1,0xF7,0xC0,0xF,0xC3,0x87,0xFF,0xFF,0x9F,0xF3,0x55,0xE7,0x81,0
x83,0x81,0xC1,0x81,0x24,0xC3,0x9C,0x94,0x99,0xC1,0x9C,0x9C,0xC3,0x80,0x83,0xC3,0x81,0x99,0xE7,0x9
9,0x99,0x99,0x94,0x94,0x1F,0x9C,0x9F,0xC3,0x31,0xC1,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0x11,0x3,0x11,0x11,0xC3,0x1F,0xF,0xF1,0x11,0x11,0x11,0x11,0x11,0x11,0x1F,0x1F,0
x11,0x11,0xC7,0xF1,0x11,0x1F,0x11,0x11,0x11,0x11,0x11,0x11,0xC7,0x11,0x11,0xFF,0xC3,0x99,0xC9
,0x99,0x25,0x93,0xE7,0xE7,0xE7,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0x99,0xE3,0x99,0x99,0xCF,0x9F,0xE7,0
xF9,0x99,0x99,0xFF,0xFF,0xF3,0xFF,0xCF,0x99,0x3C,0xC3,0x99,0x99,0x93,0x9F,0x9F,0x99,0x99,0xE7,0xF
9,0x99,0x9F,0x9C,0x9C,0x99,0x99,0x99,0x99,0x99,0xE7,0x99,0x99,0x9C,0x99,0x99,0xF9,0xCF,0x9F,0xF3,0
xFF,0xFF,0xFF,0xFF,0x9F,0xFF,0xF9,0xFF,0xCF,0xFF,0x9F,0xFF,0xFF,0x9F,0xE7,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xCF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xE7,0xE7,0xE7,0x24,0x0,0x11,0x11,0x11,0xE3,0xFF,
0xFF,0xFF,0xFF,0x33,0xC7,0xC7,0xFF,0xC7,0x11,0xE3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
```

xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xC7,0xF9,0
xCF,0xF3,0xCC,0xE6,0x7F,0xDF,0xFF,0xFF,0xF0,0x3,0xF7,0xC0,0x1F,0x8F,0xE3,0xFF,0xFF,0x8F,0xE3,0x
AA,0xC3,0x9D,0x99,0x9D,0x99,0x9F,0x24,0x99,0x9C,0x9C,0x99,0x99,0x9C,0x9C,0x99,0x9C,0x99,0x99,0xE
7,0x99,0x81,0x99,0x99,0x99,0x94,0x94,0x1F,0x9C,0x9F,0x99,0x24,0x99,0xFF,0xFD,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xE7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF7,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x11,0x3,0x11,0x11,0xC7,0x7,0x1F,0xF1,0x11,0x11,0x11,0x11,0x11,0x
11,0x1F,0x1F,0x11,0x11,0xC7,0xF1,0x11,0x1F,0x11,0x11,0x11,0x11,0x11,0x11,0xC7,0x11,0x11,0xFF,0
xC3,0x99,0x80,0x9F,0x89,0x93,0xE7,0xE7,0xE7,0xC9,0xE7,0xFF,0xFF,0xFF,0xF3,0x91,0x83,0x99,0x99,0xC
9,0x9F,0xCF,0xF3,0x99,0x99,0xE3,0xE3,0xE7,0xFF,0xE7,0x99,0x3C,0x99,0x99,0x99,0x99,0x9F,0x9F,0x99,0
x99,0xE7,0xF9,0x93,0x9F,0x88,0x8C,0x99,0x99,0x99,0x99,0x9F,0xE7,0x99,0x99,0x9C,0xCB,0x99,0xF9,0xC
F,0xCF,0xF3,0xFF,0xFF,0xFF,0xC3,0x83,0xC3,0xC1,0xC3,0xCF,0xC1,0x83,0x87,0xC3,0x99,0xE7,0x81,0x83
,0xC3,0x83,0xC1,0x99,0xC1,0x81,0x99,0x99,0x9C,0x99,0x99,0x81,0xE7,0xE7,0xE7,0x71,0x0,0x11,0x11,0x1
1,0xE3,0xFF,0xFF,0xFF,0xC7,0xFF,0x87,0xC3,0x1,0xC7,0x11,0xE3,0x39,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0x
C7,0xF9,0xCC,0xF3,0x9E,0x4F,0x3F,0xDF,0xFF,0xFF,0xF0,0x7,0xFB,0x80,0x1F,0x9F,0xF3,0xFF,0xFF,0xC
3,0x87,0x55,0x99,0x9F,0x99,0x9F,0x99,0x9F,0x81,0xB9,0x9C,0x9C,0x93,0x99,0x88,0x9C,0x99,0x9C,0x99,0
x99,0xE7,0x99,0x24,0xCB,0x99,0x99,0x94,0x94,0x9F,0x9C,0x9F,0xB9,0x24,0x99,0xC3,0xC1,0x83,0x81,0xC
1,0xC3,0xB6,0xC3,0x99,0x99,0x9D,0xF1,0x9C,0x99,0xC3,0x81,0x83,0xC3,0x81,0x99,0xC1,0x99,0x99,0x99,
0x94,0x94,0x8F,0x9C,0x9F,0xC3,0x91,0xC1,0x11,0x3,0xF1,0xF1,0x8F,0x3,0x7,0xE1,0x11,0x11,0x11,0x11,0
x1F,0x11,0x1F,0x1F,0x1F,0x11,0xC7,0xF1,0x1,0x1F,0x1,0x1,0x11,0x11,0x11,0x11,0x1F,0xC7,0x11,0x11,0xF
F,0xC3,0xFF,0xC9,0xCF,0xF3,0xC7,0xFF,0xCF,0xF3,0xE3,0xE7,0xFF,0xFF,0xFF,0xF3,0x91,0xF3,0xF9,0xF
9,0xC9,0x9F,0x83,0xF3,0x89,0x99,0xE3,0xE3,0xCF,0x81,0xF3,0xF3,0x30,0x99,0x99,0x9F,0x99,0x9F,0x9F,0
x9F,0x99,0xE7,0xF9,0x93,0x9F,0x80,0x84,0x99,0x99,0x99,0x99,0xCF,0xE7,0x99,0x99,0x94,0xE7,0x99,0xF3,
0xCF,0xCF,0xF3,0xFF,0xFF,0xFF,0xF9,0x99,0x99,0x99,0x99,0xCF,0x99,0x99,0xE7,0xF3,0x99,0xE7,0x94,0x
99,0x99,0x99,0x99,0x91,0x9F,0xCF,0x99,0x99,0x94,0x99,0x99,0xF9,0xE7,0xE7,0xE7,0xFF,0x0,0x11,0x1,0x1
,0xE3,0xFF,0xFF,0xFF,0xC7,0xFF,0x8F,0xE3,0x1,0xC7,0xF1,0xE3,0x11,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xC7,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xC7,0xC
7,0xFC,0x9C,0xF3,0x80,0x4F,0xFF,0xEF,0xFF,0xFF,0xE0,0x7,0xFD,0x60,0x3F,0x1F,0xF1,0xFF,0xFF,0xE0,
0xF,0xAA,0x99,0x83,0x99,0x9F,0x99,0x9F,0xC3,0xE3,0x98,0x98,0x93,0x99,0x80,0x9C,0x99,0x9C,0x99,0x9
F,0xE7,0x99,0x24,0xE7,0x99,0x99,0x94,0x94,0x9F,0x9C,0x9F,0xF9,0x4,0x99,0xF9,0x9F,0x99,0x9F,0xC9,0x
99,0x94,0x99,0x91,0x91,0x99,0xE1,0x88,0x99,0x99,0x99,0x99,0x99,0xE7,0x99,0x94,0x99,0x99,0x99,0x94,0x
94,0x8F,0x9C,0x9F,0x99,0x94,0x99,0x11,0xE3,0xF1,0xC1,0xF,0x1,0x3,0xE3,0x1,0x11,0x11,0x1,0x1F,0x11,0
x3,0x3,0x1F,0x1,0xC7,0xF1,0x3,0x1F,0x1,0x1,0x11,0x11,0x11,0x11,0x7,0xC7,0x11,0x11,0xFF,0xE7,0xFF,0x
C9,0xE7,0xE7,0x9F,0xFF,0xCF,0xF3,0x80,0x81,0xFF,0x81,0xFF,0xE7,0x99,0xF3,0xF3,0xE3,0xC9,0x83,0x99
,0xE7,0xC3,0x99,0xFF,0xFF,0x9F,0xFF,0xF9,0xE7,0x24,0x99,0x83,0x9F,0x99,0x83,0x83,0x9F,0x81,0xE7,0x
F9,0x87,0x9F,0x94,0x90,0x99,0x83,0x99,0x83,0xE7,0xE7,0x99,0x99,0x94,0xE7,0xC3,0xE7,0xCF,0xE7,0xF3,
0xFF,0xFF,0xFF,0xF9,0x99,0x9F,0x99,0x99,0x81,0x99,0x99,0xE7,0xF3,0x93,0xE7,0x94,0x99,0x99,0x99,0x9
9,0x8F,0x9F,0xCF,0x99,0x99,0x94,0xC3,0x99,0xF3,0xCF,0xE7,0xF3,0xFF,0x0,0x29,0x83,0x83,0xC7,0xFF,0x
FF,0x1,0x1,0xFF,0x8F,0xE3,0x1,0xC7,0xF1,0xC7,0x83,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xC7,0x7,0xC0,0xE0,0xF,0x0,0x7,0x0,0xC0,0x0,0xFC,0x9C,0xF3,0x9F,0xCF,
0x3F,0xF7,0xFF,0xFF,0xC0,0xF,0xFD,0x70,0x7F,0x3F,0xF9,0xFF,0xFF,0xF8,0x3F,0x55,0x99,0x99,0x83,0x9
F,0x99,0x83,0x81,0xE3,0x90,0x90,0x87,0x99,0x94,0x80,0x99,0x9C,0x83,0x9F,0xE7,0x99,0x24,0xE7,0x99,0x
99,0x94,0x94,0x83,0x84,0x81,0xE1,0x4,0xC1,0xF9,0x83,0x99,0x9F,0xC9,0x99,0xC1,0xF9,0x91,0x91,0x93,0x
E9,0x88,0x99,0x99,0x99,0x99,0x9F,0xE7,0x99,0x94,0xC3,0x99,0x99,0x94,0x94,0xCF,0x9C,0x9F,0xF9,0x84,
0x99,0x11,0xE3,0xE1,0xC3,0x13,0x11,0x1,0xC3,0x83,0x1,0x11,0x3,0x1F,0x11,0x3,0x3,0x11,0x1,0xC7,0xF1,
0x7,0x1F,0x1,0x1,0x11,0x11,0x11,0x1,0x83,0xC7,0x11,0x11,0xFF,0xE7,0xFF,0xC9,0xF3,0xCF,0x90,0xFF,0x
CF,0xF3,0xE3,0xE7,0xFF,0xFF,0xFF,0xE7,0x89,0xF3,0xE7,0xF9,0x99,0xF9,0x99,0xE7,0x91,0xC1,0xFF,0xF
F,0xCF,0x81,0xF3,0xE7,0x24,0x81,0x99,0x9F,0x99,0x9F,0x9F,0x91,0x99,0xE7,0xF9,0x93,0x9F,0x94,0x98,0x
99,0x9F,0x99,0x93,0xF3,0xE7,0x99,0x99,0x94,0xD3,0xE7,0xCF,0xCF,0xE7,0xF3,0xFF,0xFF,0xFF,0xC1,0x9
9,0x9F,0x99,0x81,0xCF,0x99,0x99,0xE7,0xF3,0x87,0xE7,0x94,0x99,0x99,0x99,0x99,0x9F,0xC3,0xCF,0x99,0
x99,0x94,0xE7,0x99,0xE7,0x9F,0xE7,0xF9,0xFF,0x0,0x29,0xC7,0x83,0xC7,0xFF,0xFF,0x1,0x1,0xFF,0x8F,0x
E3,0xFF,0xC7,0xE3,0xC7,0xC7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0
x0,0x0,0xC7,0x7,0xC0,0xC0,0x7,0x0,0x7,0x0,0xC0,0x0,0xFE,0x3C,0xF3,0xCC,0xE6,0x7F,0xF7,0xFF,0xFF,0
xC0,0x1F,0xFE,0xF0,0x7F,0x3F,0xF9,0xFF,0xFF,0xFF,0xFF,0xAA,0x81,0x99,0x99,0x9F,0x99,0x9F,0x24,0x
F9,0x84,0x84,0x93,0x99,0x94,0x9C,0x99,0x9C,0x9F,0x9F,0xE7,0xC1,0x24,0xD3,0x99,0xC1,0x94,0x94,0x99,
0x90,0x98,0xF9,0x4,0xE1,0xC1,0x99,0x83,0x9F,0xC9,0x81,0xE3,0xE3,0x81,0x81,0x87,0xE9,0x94,0x81,0x99
,0x99,0x99,0x9F,0xE7,0x99,0x94,0xE7,0x99,0x99,0x94,0x94,0xC1,0x84,0x83,0xE1,0x84,0xC1,0x11,0xE3,0x
C3,0xC1,0x13,0xF1,0x11,0xC7,0x1,0x81,0x1,0x1,0x1F,0x11,0x3,0x3,0x11,0x1,0xC7,0xF1,0x3,0x1F,0x1,0x1,
0x11,0x1,0x11,0x3,0xC1,0xC7,0x11,0x11,0xFF,0xFF,0xFF,0x80,0xF9,0x91,0x99,0xFF,0xCF,0xF3,0xC9,0xE7
,0xFF,0xFF,0xFF,0xCF,0x89,0xF3,0xCF,0x99,0x80,0xF9,0x99,0xCF,0x99,0xF3,0xFF,0xFF,0xE7,0xFF,0xE7,0
xFF,0x30,0x99,0x99,0x99,0x99,0x9F,0x9F,0x99,0x99,0xE7,0x99,0x93,0x9F,0x9C,0x9C,0x99,0x9F,0x99,0x99,

0xF9,0xE7,0x99,0x99,0xC9,0x99,0xE7,0x9F,0xCF,0xF3,0xF3,0xFF,0xFF,0xFF,0x99,0x99,0x9F,0x99,0x9F,0x
CF,0x99,0x99,0xE7,0xF3,0x93,0xE7,0x94,0x99,0x99,0x99,0x99,0x9F,0xF9,0xCF,0x99,0x99,0x94,0xC3,0x99,
0xCF,0xCF,0xE7,0xF3,0xFF,0x0,0x1,0x83,0xC7,0xC7,0xFF,0xFF,0x1,0x1,0xFF,0x8F,0xE3,0x1,0xC7,0xC7,0
xC7,0x83,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xC7,0xF,0xE0,0
xC0,0x7,0x0,0x7,0x0,0xC0,0x0,0xFE,0x3C,0xF3,0xE1,0xF0,0xFF,0xFB,0xFF,0xFF,0x80,0x1F,0xFE,0xF8,0xF
F,0x3F,0xF9,0xF8,0x3F,0xFF,0xFF,0x55,0x99,0x99,0x99,0x9F,0x99,0x9F,0x24,0xB9,0x8C,0x8C,0x93,0x99,0
x9C,0x9C,0x99,0x9C,0x9F,0x99,0xE7,0xF9,0x81,0xDB,0x99,0xF9,0x94,0x94,0x99,0x90,0x98,0xB9,0x24,0xC
9,0x99,0x99,0x99,0x9F,0xC9,0x9F,0xC1,0xF9,0x89,0x89,0x93,0xC9,0x94,0x99,0x99,0x99,0x99,0x9F,0xE7,0x
99,0x94,0xC3,0x99,0xC1,0x94,0x94,0xCC,0x90,0x99,0xF9,0x84,0xE1,0x11,0xE3,0x87,0xF1,0x1,0xF1,0x11,0
x87,0x11,0xC1,0x1,0x11,0x1F,0x11,0x1F,0x1F,0x11,0x11,0xC7,0xF1,0x1,0x1F,0x29,0x1,0x11,0x3,0x11,0x7,
0xF1,0xC7,0x11,0x83,0xFF,0xE7,0xFF,0xC9,0x99,0xA4,0x99,0xFF,0xCF,0xF3,0xFF,0xFF,0xE3,0xFF,0xE3,0
xCF,0x99,0xF3,0x9F,0x99,0xF9,0xF3,0x99,0xCF,0x99,0xE7,0xE3,0xE3,0xF3,0xFF,0xCF,0xE7,0x3F,0x99,0x9
9,0x99,0x93,0x9F,0x9F,0x99,0x99,0xE7,0x99,0x99,0x9F,0x9C,0x9C,0x99,0x9F,0x99,0x99,0x99,0xE7,0x99,0x
C3,0xC9,0x99,0xE7,0x9F,0xCF,0xF3,0xF3,0xFF,0xFF,0xFF,0x99,0x99,0x99,0x99,0x9F,0xCF,0x99,0x99,0xE7
,0xF3,0x99,0xE7,0x94,0x99,0x99,0x99,0x99,0x9F,0xF9,0xCF,0x99,0xC3,0xC9,0x99,0x99,0x9F,0xE7,0xE7,0x
E7,0xFF,0x0,0x1,0x1,0xC7,0x8F,0xFF,0xFF,0xFF,0xC7,0xFF,0x8F,0xE3,0x1,0xC7,0xC7,0x8F,0x11,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0
xC7,0xFF,0xC7,0xC7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFD,0xFF,0xFF,0x0,0x3F,0xFF,0xFD,0xFF,0x1F,0x
F1,0xE0,0xF,0xFF,0xFF,0xAA,0x99,0x99,0x99,0x9F,0x99,0x9F,0x24,0x99,0x9C,0x9C,0x99,0x99,0x9C,0x9C,
0x99,0x9C,0x9F,0x99,0xE7,0x99,0xE7,0x99,0x99,0xF9,0x94,0x94,0x99,0x90,0x98,0x99,0x24,0x99,0x99,0x99
,0x99,0x9F,0xC9,0x9F,0x94,0x99,0x89,0x89,0x99,0xD9,0x9C,0x99,0x99,0x99,0x99,0x99,0xE7,0xC1,0xC1,0x
99,0x99,0xF9,0x94,0x94,0xCC,0x90,0x99,0x99,0x94,0xC9,0x11,0xE3,0xF,0x11,0x1,0x11,0x11,0x87,0x11,0xF
1,0x1,0x11,0x11,0x11,0x1F,0x1F,0x11,0x11,0xC7,0x11,0x11,0x1F,0x29,0x11,0x11,0x7,0x1,0x3,0x11,0xC7,0x
11,0x83,0xFF,0xE7,0xFF,0xC9,0xC3,0xE4,0xC4,0xFF,0xE7,0xE7,0xFF,0xFF,0xE3,0xFF,0xE3,0x9F,0xC3,0x
F3,0x81,0xC3,0xF9,0x87,0xC3,0xCF,0xC3,0xC7,0xE3,0xE3,0xF9,0xFF,0x9F,0xE7,0x80,0x99,0x83,0xC3,0x8
7,0x81,0x9F,0xC1,0x99,0xC3,0xC3,0x99,0x81,0x9C,0x9C,0xC3,0x9F,0xC3,0x99,0xC3,0xE7,0xC3,0xE7,0xC9
,0x99,0xE7,0x81,0xCF,0xF9,0xF3,0xFF,0xFF,0xFF,0xC1,0x83,0xC3,0xC1,0xC3,0xCF,0xC1,0x99,0x81,0xF3,
0x99,0x81,0x9C,0x99,0xC3,0x83,0xC1,0x9F,0x83,0xE1,0xC1,0xE7,0xC9,0x99,0xC1,0x81,0xE7,0xE7,0xE7,0
xFF,0x0,0x1,0x11,0xC7,0x8F,0xFF,0xFF,0xFF,0xC7,0xFF,0x87,0xC3,0x1,0xC7,0xC7,0x8F,0x39,0xFF,0xFF,0
xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,
0xFF,0xC7,0xC7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFD,0xFF,0xFF,0x0,0x7F,0xFF,0xFD,0xFF,0x9F,0xF3,0
xC3,0x87,0xFF,0xFF,0x55,0x99,0x83,0x83,0x9F,0x0,0x81,0x24,0xC3,0x9C,0x9C,0x99,0x39,0x9C,0x9C,0xC3
,0x9C,0x9F,0xC3,0xE7,0xC3,0xE7,0x99,0x80,0xF9,0x80,0x80,0x83,0x84,0x81,0xC3,0x31,0x99,0xC4,0xC3,0x
83,0x9F,0x80,0xC1,0xB6,0xC3,0x99,0x99,0x9D,0x99,0x9C,0x99,0xC3,0x99,0x83,0xC3,0xE7,0xF9,0xF7,0x99
,0x80,0xF9,0x80,0x80,0xC1,0x84,0x83,0xC3,0x91,0x99,0x11,0xE3,0x1F,0x11,0x1,0x11,0x11,0x8F,0x11,0x11
,0x11,0x11,0x11,0x11,0x1F,0x1F,0x1,0x11,0xC7,0x11,0x11,0x1F,0x11,0x11,0x11,0x1F,0x1,0x11,0x11,0xC7,0
x11,0x83,0xFF,0xFF,0xFF,0xFF,0xE7,0xF1,0xFF,0xFF,0xE7,0xE7,0xFF,0xFF,0xF3,0xFF,0xFF,0x9F,0xFF,0x
FF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0x
FF,0xFF,0xFF,0xFF,0xCF,0xF9,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xFF,0xFF,0x
F3,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xFF,0xE7,0xE7,0xE
7,0xFF,0x0,0x1,0x11,0xC7,0x8F,0xFF,0xC7,0xFF,0xFF,0xFF,0xC7,0xC7,0xFF,0xFF,0xFF,0x8F,0xFF,0xFF,0
xFF,0xFF,0xFF,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0x
C7,0xFF,0xC7,0xC7,0xC0,0x0,0x0,0x0,0x0,0x0,0x1F,0xFE,0xFF,0xFE,0x0,0x7F,0xFF,0xFF,0xFF,0x8F,0xE3,0x8
F,0xE3,0xFF,0xFF,0xAA,0xFF,0xFF,0xFF,0xFF,0x3C,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFC,0xFF,0xFF,0xFC,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0x9C,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0xFF,0xFF,0xF9,0xF7
,0xFF,0xFC,0xFF,0xFF,0xFC,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x1,0xE3,0x1,0x1,0xE3,0x1,0x1,0x8F,0x1,0x
1,0x11,0x1,0x1,0x1,0x1,0x1F,0x1,0x11,0x83,0x1,0x11,0x1,0x11,0x11,0x1,0x1F,0x3,0x11,0x1,0xC7,0x1,0xC7,
0xFF,0xFF,0xFF,0xFF,0xE7,0xFF,0xFF,0xFF,0xF3,0xCF,0xFF,0xFF,0xE7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xE7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xCF,0xFF,0xF3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xFF,0xFF,0xF3,0xFF,0x
FF,0xFF,0xFF,0xFF,0x9F,0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF9,0xFF,0xF3,0xE7,0xCF,0xFF,0x0
,0x83,0x11,0xC7,0x1,0xC7,0xC7,0xFF,0xFF,0xFF,0xC3,0x87,0xFF,0xC7,0xC7,0x1F,0xFF,0xFF,0xFF,0xFF,0
x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,
0xFF,0xC7,0xC7,0xC0,0x0,0x0,0x0,0x0,0x0,0x1F,0xFF,0x7F,0xFC,0x0,0xFF,0xFF,0xFF,0xFF,0xC3,0x87,0x9F,0xF3,0xFF
,0xFF,0x55,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0xFF,0xFF,0x99,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x83,0xE3,0x1,0x83,0xE3,0x83,0x83,0x8F,0x83,0x83,0x11,
0x3,0x83,0x3,0x1,0x1F,0x81,0x11,0x83,0x83,0x11,0x1,0x11,0x19,0x83,0x1F,0x81,0x11,0x83,0xC7,0x83,0xC

```
7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xC3,0xFF,0xC3,0xFF,0x0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xC3,0xFF,0xFF,0xC7,0xFF,
0xFF,0xFF,0xFF,0xFF,0x9F,0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xC3,0xFF,0xFF,0xE7,0xFF,0xFF,0
x0,0x93,0x11,0xC7,0x1,0xC7,0xC7,0xFF,0xFF,0xFF,0xE1,0xF,0xFF,0xC7,0xC7,0x1F,0xFF,0xFF,0xFF,0x0,0x
0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xFF,0xC
7,0xC7,0xE0,0x0,0x0,0x0,0x0,0x0,0x3F,0xFF,0x7F,0xFC,0x1,0xFF,0x0,0x0,0x0,0x0,0xE0,0xF,0x1F,0xF1,0xFF,0xFF,0
xAA,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x9F,0xFF,0xFF,0xC3,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xC7,0xE3,0x1,0xC7,0xE3,0xC7,0xC7,0x8F,0xC7,0xC7,0x11,0x
7,0xC7,0x7,0x1,0x1F,0xC9,0x11,0x83,0xC7,0x11,0x1,0x11,0x19,0xC7,0x1F,0xC1,0x11,0xC7,0xC7,0xC7,0xC
7,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xF
F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,
0xFF,0xFF,0xFF,0xFF,0x9F,0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
x0,0x93,0x11,0xC7,0x1,0xC7,0xCF,0xFF,0xFF,0xFF,0xF1,0x1F,0xFF,0xC7,0xC7,0x1F,0xFF,0xFF,0x0,0x0,0x
0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0xC7,0xFF,0xC
7,0xC7,0xF0,0x0,0x0,0x0,0x0,0x0,0x7F,0xFF,0xBF,0xF8,0x1,0xFF,0x0,0x0,0x0,0xF8,0x3F,0x3F,0xF9,0xFF,0xFF,
0x55,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0x0,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x0
,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0x0,0xFF,0xC7,0xFF,0xFF,0xC7,0xC7,0xC7,0xC
7,0xFF,0xC7,0xC7,0xF8,0x0,0x0,0x0,0x0,0x0,0x7F,0xFF,0xDF,0xF0,0x3,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0x3F,
0xF9,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0
xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF};
```

// файл main.c

```
#include <avr/io.h>
#include <avr/pgmspace.h>
#include <avr/signal.h>
#include "types.h"
#include "video.h"

void Init (void);
void Init (void)                // функція ініціалізації
{
        DDRB=0x30; //  налаштування виводів порту B на вихід (вихід синхронізації)
        DDRD=0xC0;// налаштування 2-х виводів порту D на вихід (звук і відео)
        SPCR=0x50;              // SPI в режимі master
        SPSR=0x01;              // дільник частоти роботи SPI дорівнює 2
        TCCR1B=0x01;            // запуск таймера 1
        TIMSK=0x04;             // дозволити переривання за переповненням таймера 1
        TIFR=0x04;              // очистити прапор переривання таймера 1
        TCCR2=0x1E;             // перемикання і очищення при співпадінні, дільник=256
```

```
        OCR2=0x00;          // в регістр порівняння записано 0 – немає звуку
}

void main(void)
{
        u8 i=0;             // додаткова змінна
        Init();             // виклик функції ініціалізації
        sei();              // дозвіл обробки переривань
        clrscr();           // очищення екрану
        // виведення на екран заголовку
        print_xys(0,0,PSTR("\xAB\xAC\xAD\xAE\xAF\xB0"));
        print_xys(0,1,PSTR("\xB1\xB2\xB3\xB4\xB5              Font Test"));
        print_xys(0,2,PSTR("_\xB6\xB7\xB8_____________________________"));
        do { printc(i); i++; } while(i!=0);     // виведення всих букв алфавіта
        // зображення прямокутника
        printc(0xA4);                           // верхній лівий кут
        for (i=0;i<30;i++) printc(0xA0);        // верхня лінія
        printc(0xA5);                           // правий верхній кут
        gotoxy(0,12); printc(0xA1);             // лінія ліворуч
        gotoxy(0,13); printc(0xA1);             // лінія ліворуч
        gotoxy(0,14); printc(0xA1);             // лінія ліворуч
        gotoxy(31,12); printc(0xA1);            // лінія праворуч
        gotoxy(31,13); printc(0xA1);            // лінія праворуч
        gotoxy(31,14); printc(0xA1);            // лінія праворуч
        print_xys(0,15,PSTR("\xA3\xA0\xA0\xA0(c) Kathedra TEES
        2008\xA0\xA0\xA0\xA0\xA2"));     // нижня лінія
        // головний цикл, зображення кіл
        u8  x=1,  y=1;                  // координати кола
        s8 dx=1, dy=1;
        while(1)
        {
                // через один кадр (25 Hz)
                while(video_cnt==0) {nop();} video_cnt=0; // очікування наступного кадра
                while(video_cnt==0) {nop();} video_cnt=0; // очікування наступного кадра
                // вимкнути звук
                beep(0);
                if (x==1)  { dx=1; beep(10); }          // сигнал високої частоти
                if (x==29) { dx=-1; beep(20); }         // сигнал низької частоти
                if (y==0)  dy=1;
                if (y==4)  dy=-1;
                // стерти кола
                gotoxy(x,(y>>1)+12);                    // встановити нові координати
                printc(' ');
                printc(' ');
                if(y&1)
                {
                        gotoxy(x,(y>>1)+13);
                        printc(' ');
                        printc(' ');
                }
                // Оновлення координат
```

```
            x+=dx;
            y+=dy;

        // Зображення кіл в новій позиції
            if(y&1)
            {
                gotoxy(x,(y>>1)+12);
                printc(0xBB);
                printc(0xBC);
                gotoxy(x,(y>>1)+13);
                printc(0xBD);
                printc(0xBE);
            }
            else
            {
                gotoxy(x,(y>>1)+12);
                printc(0xB9);
                printc(0xBA);
            }
    }
}
```

На екрані телевізора буде вивидитися зображення, показане на рис. 8.17.



Рис. 8.17.

8.9. Телетекст

Телетекст – це інформаційна система для масового користування, що забезпечує передачу власникам телевізорів найрізноманітнішої інформації разом зі звичайними телепередачами. В даний час використовується в якості світового стандарту WST (World System Teletext – всесвітня система телетексту). Інформація, що передається за стандартом WST, може бути текстовою або графічною. Вона формується на телецентрі у вигляді сторінок, пронумерованих від 100 до 899, і згрупованих у журнали. Кожен з них присвячений певній темі.

Для отримання інформації з системи телетексту (ТХТ) власник телевізора повинен налаштувати його на програму, яка веде таку передачу, і, перемкнувши на прийом сигналів ТХТ, викликати сторінку 100 з переліком журналів. Потім, вибравши і викликавши потрібний журнал, переглянути його зміст, і вивести бажаний розділ на екран телевізора. Таким є загальний алгоритм отримання інформації в системі ТХТ.

Існує декілька варіантів його реалізації. Це режим LIST, що вимагає виконання всіх вказаних кроків. Є режими зі спрощеною процедурою: FAST (швидкий, зручний телетекст), FLOF (Full Level One Features – одна функція для всіх рівнів, що у вільному перекладі означає – виклик всіх сторінок однією кнопкою), TOP (Table of Pages – список сторінок). У деяких регіонах використовують мало поширені системи Antiope, Safari, Spanish Teletext.

Основні відмінності між цими режимами полягають в характері зв'язку між сторінками, і в способі їх пошуку.

У режимах LIST і TOP такого зв'язку немає, сторінки самостійні і викликаються за їхніми номерами. Але якщо викликана одна із сторінок багатосторінкового розділу, разом з нею виводиться на екран повідомлення про наявність продовження і число сторінок в ньому (вони іменуються підсторінками). Різниця між режимами LIST і TOP полягає в тому, що в режимі LIST для виклику сторінки потрібно набрати її номер на пульті дистанційного управління (ПДУ), а в режимі TOP використовується меню (перелік сторінок на екрані), на якому встановлюють курсор (управляється з ПДУ) навпроти рядка з назвою потрібного журналу, розділу.

У режимі FLOF вся інформація згрупована по чотирьох темах, а ПДУ має чотири кольорові кнопки для їх виклику. При натисканні однієї з них, на екран послідовно виводяться одна за одною всі сторінки теми. Зміну сторінок можна зупинити для аналізу, а потім продовжити.

У режимі FAST перегляд сторінок організоване інакше. На першій сторінці кожного журналу, окрім списку розділів і номерів сторінок, є чотири кольорові поля з номерами сторінок. Кожному полю відповідає кнопка такого ж кольору на ПДУ. При її натисканні викликається (без набору номеру) сторінка, номер якої був вказаний на вибраному полі. На цій сторінці також є поля, але з іншими номерами. Діючи таким чином, можна за декілька кроків вийти в потрібний розділ і на потрібну сторінку.

Разом з тим, в будь-якому режимі кожна сторінка може бути вибрана способом, що застосовується в режимі LIST – шляхом набору її номеру.

Незважаючи на велику кількість режимів реалізації процесу пошуку інформації, кожен телецентр може використовувати лише два способи: LIST, і один зі швидких режимів (FAST, FLOF, TOP). В той же час, на приймальній стороні має бути забезпечена можливість багаторежимної роботи для прийому повідомлень від будь-якого телецентру.

Сторінка TXT стандарту WST складається з 25 рядків по 40 символів в рядку. Перший рядок – заголовок сторінки. У рядках 2-25 розміщена інформація TXT, а в режимах FAST і FLOF рядок 25 служить рядком статусу.

Заголовок містить номер сторінки власника; номер і найменування сторінки *W*, що передається телецентром у поточний момент; дату і час передачі; число і номери підсторінок. У рядку статусу відображуються кольорові поля з назвами тем (режим FLOF) або номерами сторінок (режим FAST).

Будь-який рядок передається серією з 45 байтів. Байти 1-3 – синхронізуючі. Байти 4, 5 є адресою рядка: номер журналу і номер рядка в сторінці.

Байти 6-45 заголовку використовують таким чином: у 6, 7 записаний номер сторінки *N*; у 8-11 – дата і час; у 12-45 – номер і назва сторінки *W*, а також символьна інформація, що виводиться в заголовку (день тижня і т.п.). Ці ж байти в інших рядках містять символьну інформацію про текст, який передається. Для підвищення перешкодостійкості восьмому біту кожного байта присвоюється значення, що забезпечує непарну кількість одиниць в байті. Адреса рядка захищена побітно.

Інформація TXT, підготовлена спеціальною службою телецентру до передачі, в цифровій формі зберігається в банку даних, з якого вона циклічно витягується і посторінково вводиться в телевізійний відеосигнал (ПЦТВ). Передача сторінок відбувається під час кадрових імпульсів, що гасять (КГІ).

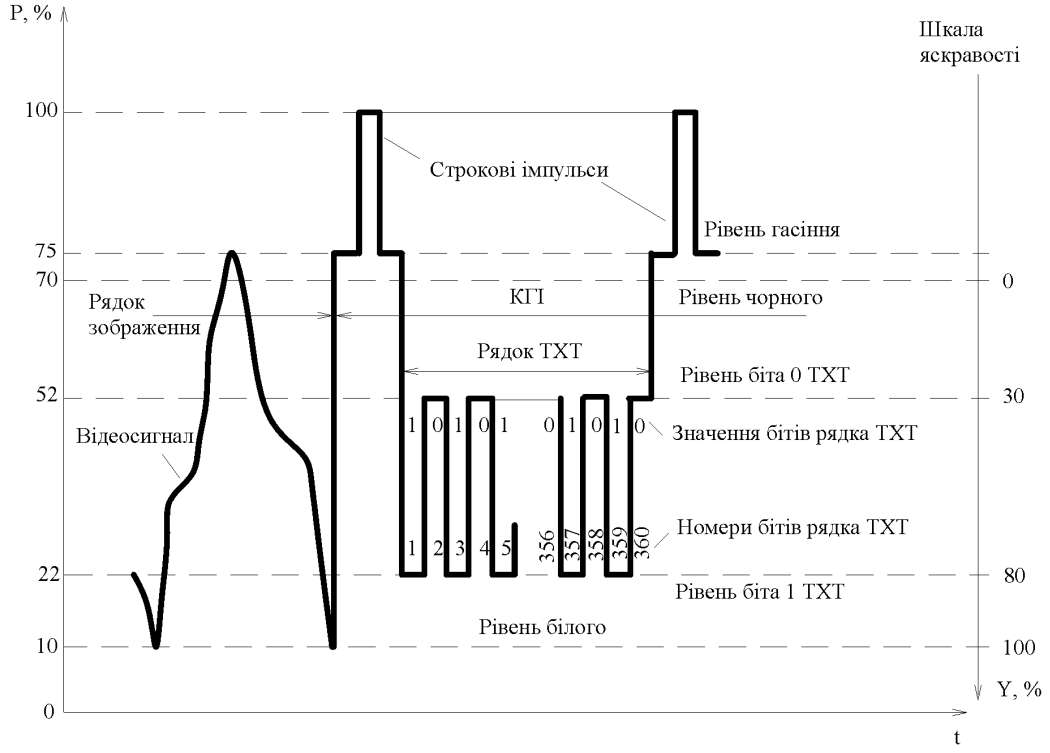


Рис. 8.18.

КГІ першого напівкадру (поля) ПЦТВ займає інтервал з 623-го рядка попереднього поля по 23-й рядок першого поля, а другого поля – з 311-ого по 335-й рядки. Частина з них вже зайнята зрівнюючими строковими імпульсами, сигналами кольорової синхронізації системи SECAM і телевізійними тестовими сигналами. Вільними в кожному кадрі є лише 12 рядків з номерами 6, 16-18, 22, 23, 318, 319, 329-332. У них і розміщують сигнали ТХТ.

На рис. 8.18 показана осцилограма ПЦТВ при передачі КГІ, і положення в ньому сигналів ТХТ. На ній розміщено дві вертикальні вісі: рівня миттєвої потужності *P*, що випромінюється передавачем, і рівня сигналу яскравості *Y*, що відповідає цій потужності. Оскільки вітчизняне тлвізійне мовлення ведеться з використанням негативної модуляції, нульові значення на цих вісях розташовані на різних рівнях, а вісі направлені в різні боки.

Рядок ТХТ передають в інтервалі між двома строковими імпульсами, що гасять. Цей інтервал дорівнює 52 мкс, і за цей час повинно бути передано 45 байт (360 біт) інформації. Отже, швидкість їх передачі має бути не нижче 6,923 Мбіт/с. У стандарті WST прийнято, що серія бітів рядка ТХТ передається сигналами прямокутної форми з тривалістю імпульсів і пауз 0,144144 мкс. Біту із значенням 1 відповідає сигнал з рівнем 80% яскравості ПЦТВ, а біту 0 – 30% яскравості. Ці сигнали займають смугу частот 4...10 МГц, що виходить за межі спектру ПЦТВ, обмеженого в різних системах мовлення частотою 5...6 МГц. Щоб ввести їх в спектр ПЦТВ, піднесучу сигналів телетексту зсувають на частоту 3,46875 МГц (гармоніка 222 строкової частоти), причому верхню бічну смугу придушують.

При використанні одного телевізійного рядка в кожному напівкадрі для передачі сигналів ТХТ, пропускна спроможність за стандартом WST дорівнює двом рядкам ТХТ за кадр, або 0,5 с на сторінку.

Для формування тексту, що виводиться разом із зображенням, може використовуватися мікросхема STV5730. До основних функцій цієї мікросхеми можна віднести:

– виведення 11 рядків по 28 символів на екран телевізора (308 символів);

–можливість створення 128 власних символи, які зберігаються в пам’яті мікросхеми;

–формування внутрішнього сигналу синхронизації, або витягнутого з відеосигналу;

– установка одного з 8 кольорів символів, фону, режиму мерехтіння символів;

– установка атрибутів рядків (зсув по горизонталі і вертикалі, включення/виключення фону, границь символів, масштабування);

– установка атрибутів сторінок (зсув по горизонталі і вертикалі, колір фону, частота мерехтіння);

– три режими работи: змішаний (текст формується на зображенні, що виводиться), повносторінковий режим (текст виводиться на фоні, внутрішнє формування синхросигналу) і режим, в якому текст накладений на відео.

На рис. 8.19 представлена структурна схема підключення системи телетексту.

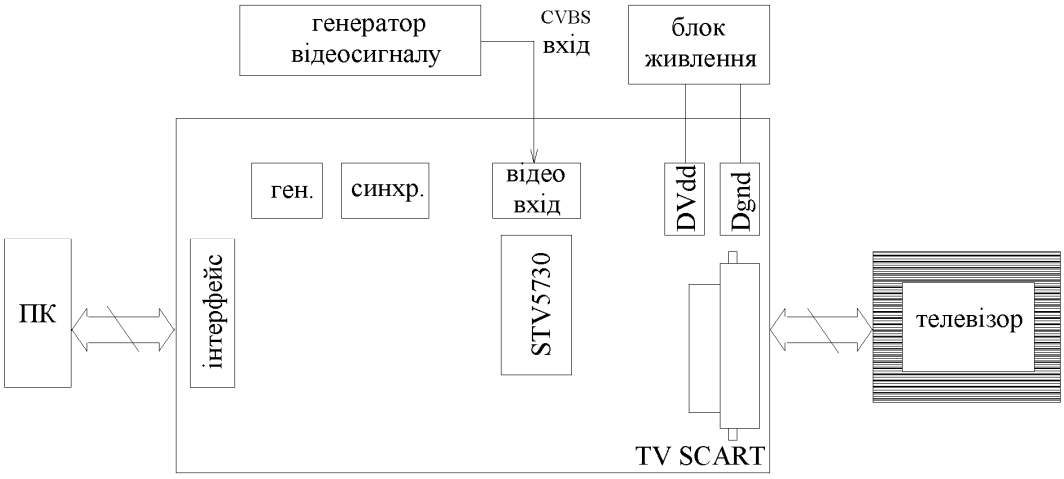


Рис. 8.19.

Мікросхема STV5730 містить декілька 12-разрядних регістрів управління режимом відображення телетексту. Далі приводиться опис цих регистрів.

Регістри установки атрибутів рядків (з 0 по 10) мають адреси від 0000 1100 0000 до 0000 1100 1010 для кожного рядка відповідно.

11	10	9	8	7	6	5	4	3	2	1	0
Зсув по горизонталі				RE	FBE	Зсув по вертикалі					

Зсув по вертикалі: старший значущий біт є знаковим (MSB, 0 – позитивний, 1 – негативний). Якщо значення лежить в межах від 0 до 17, то нові рядки додаються зверху, тобто текст рухається зверху вниз. Якщо значення від'ємні (від 0 до –17), то вказана кількість рядків видаляється.

FBE: якщо значення цього біта дорівнює 0, то символи відображаються без межі, інакше кожен символ має межу.

RE: якщо значення цього біта дорівнює 0, рядок тексту не відображається, інакше символи відображуються на фоні.

Зсув по горизонталі: рядок, що відображається, зсунутий праворуч на вказану кількість точок.

Регістр масштабування має адресу 0000 1100 1100:

11	10	9	8	7	6	5	4	3	2	1	0
VZb1	VZb0	HZb1	HZb0	VZm1	VZm0	HZm1	HZm0	VZt1	VZt0	HZt1	HZt0

HZt[1:0] – масштабний коефіцієнт по вертикалі верхнього (0) рядка:
00 – 1 піксель на 1 точку символа;
01 – 2 пікселя на 1 точку символа;
10 – 3 пікселя на 1 точку символа;
11 – 4 пікселя на 1 точку символа;
VZt[1:0] – масштабний коефіцієнт по горизонталі верхнього (0) рядка:
00 – 1 піксель на 1 точку символа;
01 – 2 пікселя на 1 точку символа;
10 – 3 пікселя на 1 точку символа;
11 – 4 пікселя на 1 точку символа;
HZm[1:0] – аналогічно HZt[1:0], але для рядків з 1 по 9;
VZm[1:0] – аналогічно VZt[1:0], але для рядків з 1 по 9;
HZb[1:0] – масштабний коефіцієнт по вертикалі нижнього (10) рядка;
VZb[1:0] – масштабний коефіцієнт по горизонталі нижнього (0) рядка.

Регістр управління кольором, розташований за адресою 0000 1100 1101:

11	10	9	8	7	6	5	4	3	2	1	0
R	G	B	R	G	B	//	//	//	R	G	B
SBC			FBC			//			BCC		

SBC – колір фону екрану;
FBC – колір межі символів;
BCC – колір фону символів.

Регістр управління растром, розташований за адресою 0000 1100 1110:

11	10	9	8	7	6	5	4	3	2	1	0
C11	C10	C9	C8	C7	C6	C5	C4	C3	C2	C1	C0

C11 – значення 0 вмикає повносторінковий режим відео, 1 – нормальний повносторінковий режим;
C10 – значення 0 – рівні яскравості генеруються всередині мікросхеми, 1 – встановлюються на виводах LESCREEN і LECHAR;
C9 – значення 0 вимикає кодування кольору символів, 1 – вмикає режим кодування кольору символів;
C8 – значення 0 вимикає режим NTSC, 60 Гц, значення 1 вмикає режим PAL/SECAM, 50 Гц;
C7 – при значенні біта, рівному 0, композитний сигнал синхронізації береться з виводу CSYNC, а при значенні біта, рівному 1 – формується мікросхемою;
C[6:5] – біти встанавлюють режим мерехтіння;
C4 – встановлюється період мерехтіння: 1 – 0,5 с, 0 – 1 с;

С3 – при значенні біта, рівному 0, колір символів встанавлюється в регістрі управління кольором (CBACKG і CBORD), а при значенні 1 параметри налаштовуються за допомогою регістра атрибутів символів;
С2 – значення 0 вимикає екран, 1 – вмикає;
С1 – значення біта, рівне 0, вмикає режим управління фоном символів за допомогою біта BE, 1 – фон символів вимкнено;
С0 – значення біта, рівне 0, вмикає змішаний режим відображення, 1 – режим повних сторінок.

Регістр положення, розташований за адресою 0000 1100 1111, використовується для оптимального розташування тексту на екрані телевізора:

11	10	9	8	7	6	5	4	3	2	1	0
Позиція по вертикалі						Позиція по горизонталі					

Значення по горизонталі лежать в межах від 6 до 63, а по вертикалі від 1 до 63.

Регістр пам'яті: для модифікації ОЗП необхідно вказати номер рядка (від 0 до 10 – 11 рядків), в якому відображатимуться символи (значення від 0 до 0x0A). Потім йде номер позиції (від 0 до 28 – 28 символів в рядку). Новий символ має бути визначений посилкою 12-бітового слова:

11	10	9	8	7	6	5	4	3	2	1	0
BE	R	G	B	BK	Код символа						

BE – біт управляє режимом мерехтіння символів (0 – викл., 1 вкл.);

RGB – колір символу: 000 – чорний, 001 – синій, 010 – зелений, 011 – бірюзовий, 100 – червоний, 101 – фіолетовий, 110 – жовтий, 111 – білий;

BK – біт управляє режимом відображення фону (0 – викл., 1 вкл.);

Код символу – значення від 0 до 127, яким відповідають символи в ПЗП.

На рис. 8.20 представлена принципова схема включення мікросхеми телетексту.

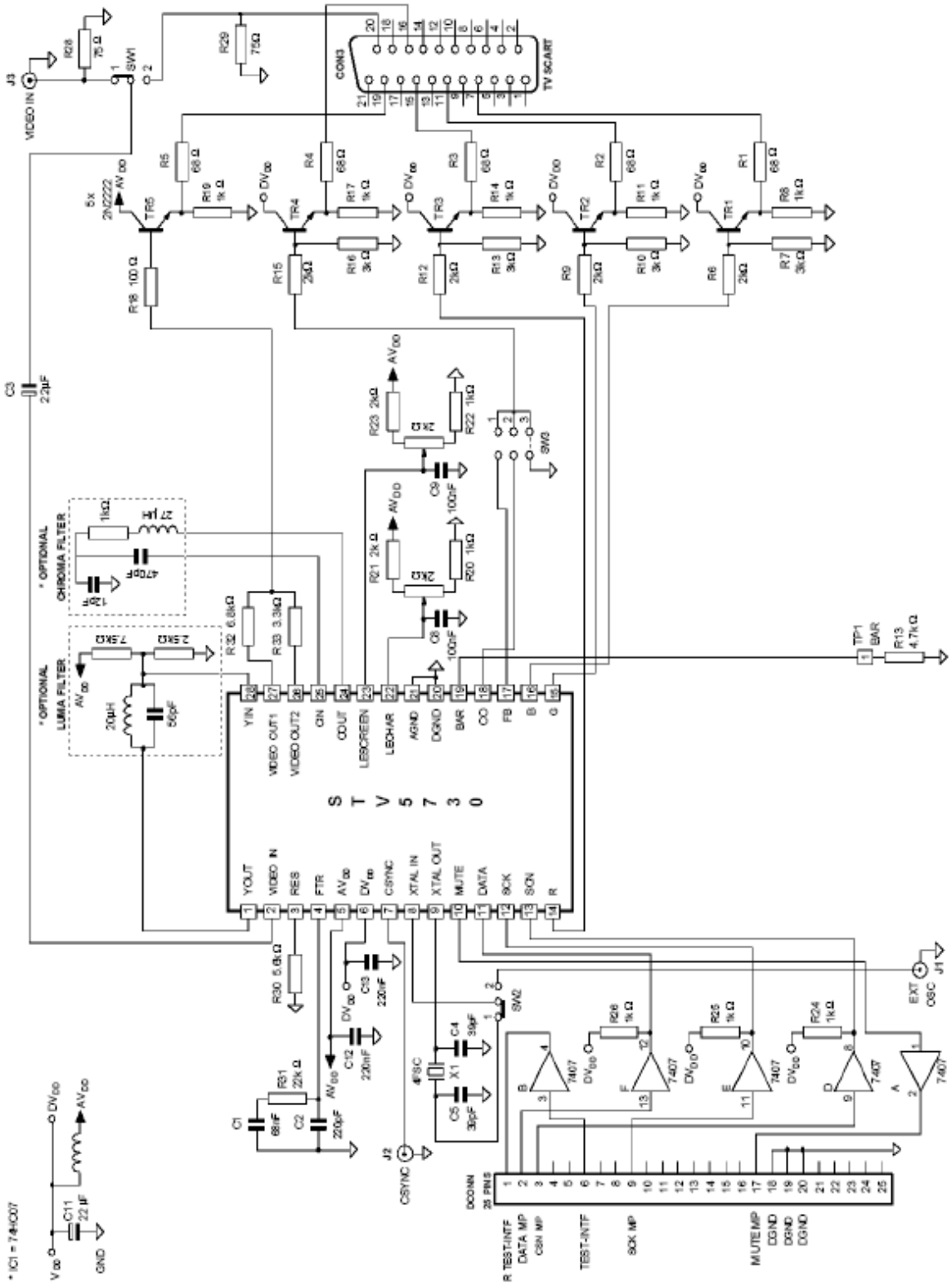


Рис. 8.20.

На рис. 8.21 представлені зображення символів, які зберігаються в ПЗП мікросхеми.

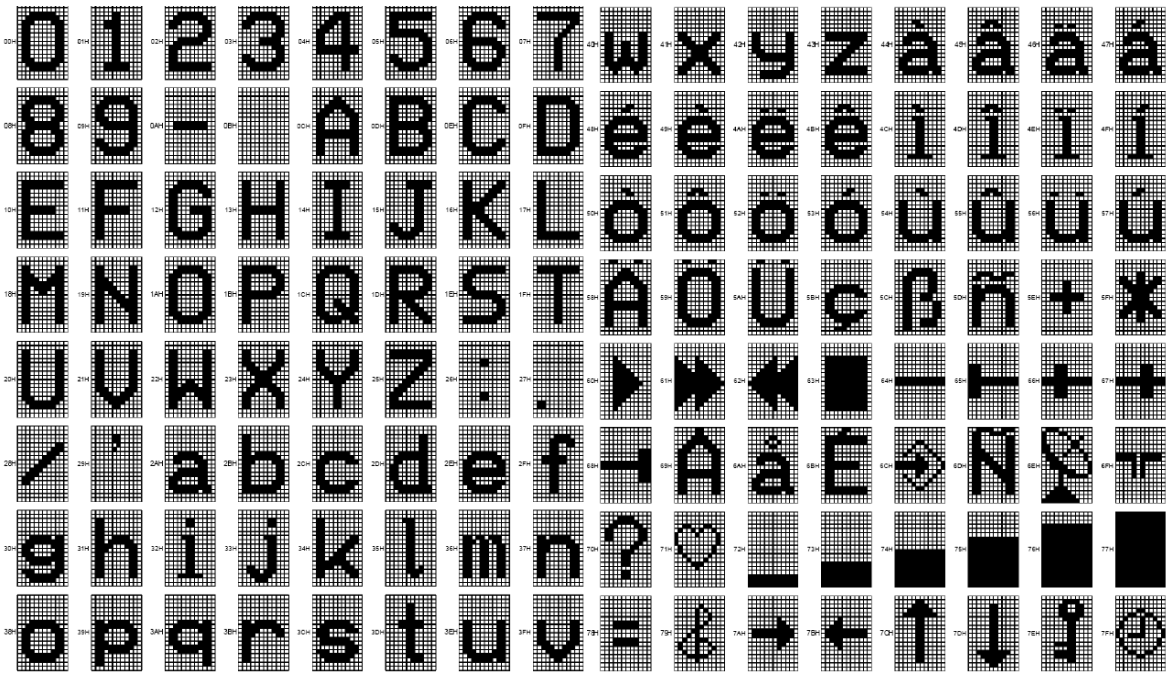


Рис. 8.21.

8.10. Опис порту VGA

Відеографічний матричний адаптер VGA, розроблений в 1988 р., дозволяє реалізувати 640×480 точок в графічному режимі. Кількість бітової площини визначає, скільки біт відводиться для зберігання атрибутивної ознаки пікселя. Якщо відеопам'ять має одну бітову площину, то такий дисплей може працювати лише в монохромному режимі (піксель може бути або яскравим, або темним). За наявності двох бітових площин у відеопам'яті може зберігатися 2^2=4 значення, що визначають, як повинен виглядати піксель на екрані (при кольоровому моніторі – чотири кольори, один з яких має код 00 – чорний (тобто фактично за допомогою двох бітових площин можна управляти RGB-монітором). При восьми бітових площинах атрибут пікселя забезпечує кодування 2^8=256 кольорів – такий адаптер ефективно застосовувати лише для аналогових RGB-моніторів, в яких між відеопам'яттю і управляючими кольором електродами ЕПТ розташовується цифро-аналоговий перетворювач (ЦАП) (Digital to Analog Converter, DAC). У ЦАП з відеопам’яті подається код кольору. З ЦАП в ЕПТ видається аналоговий сигнал (код кольору перетворюється у величину напруги на управляючому електроді).

Інколи між відеопам'яттю і ЦАП розташовуються регістри палітри (RAM DAC). Атрибутивна ознака кожного пікселя в цьому випадку позначає номер регістра палітри, в якому зберігається код кольору даного пікселя. При виборі відповідного регістра палітри код кольору, що знаходиться в ньому, передається в ЦАП, і управляє світінням пікселя.

Фізично відеопам'ять може мати лінійну структуру. Розбиття її на відеоплощини в цьому випадку може здійснюватися програмним шляхом –

за допомогою драйвера дисплея. Тому є можливість одну і ту ж відеопам'ять використовувати для різної роздільної здатності екрану (змінюючи довжину бітової площини), і для різної кількості кольорів, що відтворюються на екрані. Тому при фіксованому об'ємі пам'яті можна збільшити роздільну здатність (але при цьому скоротиться кількість відтворених кольорів), або збільшити кількість відтворених кольорів (зменшивши, відповідно, роздільну здатність екрану). Якщо ж відеоплощини реалізовані апаратно, перемикання режимів (мод екрану) може виконуватися драйвером дисплея.

Для відтворення динамічних (рухомих, анімаційних) зображень відеопам'ять доводиться ділити на сторінки, які по черзі виводяться на екран при кожній регенерації (поки одна сторінка виводиться на екран, друга заповнюється черговим кадром).

У всіх адаптерах частина відеопам'яті відводиться під знакогенератор, в якому записані коди форми символів, що виводяться на екран. В деяких випадках у відеопам'яті доводиться зберігати декілька знакогенераторів, наприклад з національними шрифтами.

На рис. 8.22 показано зовнішній вигляд роз’єму VGA.

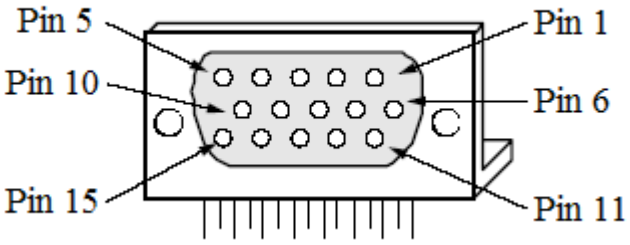


Рис. 8.22.

Для формування зображення на екрані монітора необхідно подавати на порт мінімум 5 сигналів: рівень червоного, рівень зеленого, рівень синього, сигнали строкової і кадрової розгорток. На рис. 8.23 показано призначення виводів порту VGA.

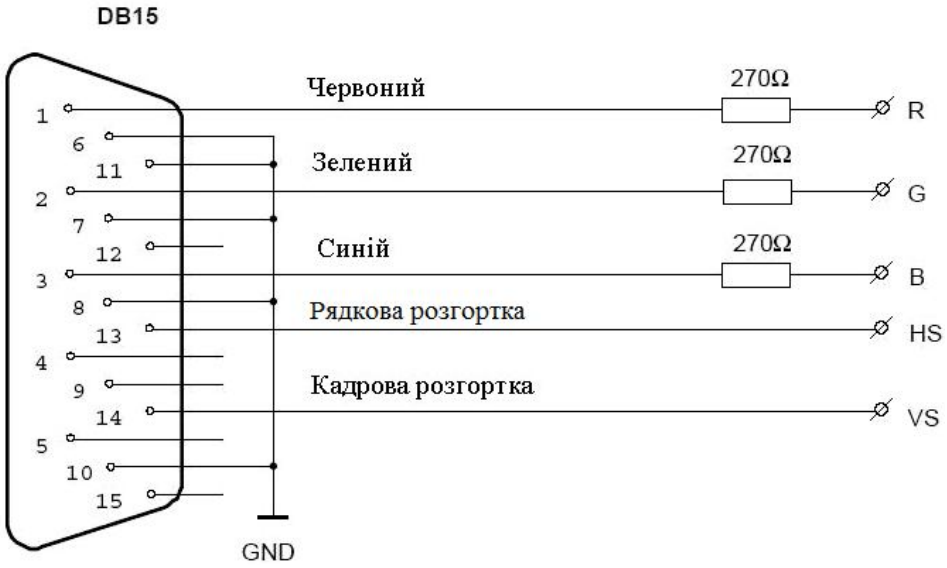


Рис. 8.23.

У кожній лінії управління кольором є резистор для забезпечення управління 3-бітовим кольором. Сигнали кольорів, що поступають на порт, змінюються в межах від 0 до 0,7 В, що забезпечується кабелем VGA. Сигнали розгорток HS і VS мають рівні ТТЛ. Управляючи сигналами R, G і B можна сформувати один з 8 кольорів (табл. 8.1). В розділі 1 приводиться коло кольорів, яке пояснює принцип кодування інформації про колір та насиченість.

Табл. 8.1.

Червоний (R)	Зелений(G)	Синій(B)	Результуючий колір
0	0	0	Чорний (Black)
0	0	1	Синій (Blue)
0	1	0	Зелений (Green)
0	1	1	Бірюзовий (Cyan)
1	0	0	Червоний (Red)
1	0	1	Фіолетовий (Magenta)
1	1	0	Жовтий (Yellow)
1	1	1	Білий (White)

Осцилограма сигналів управління ходом променя CRT-монітора показана на рис. 8.24, а сигнал синхронізації на рис. 8.25.

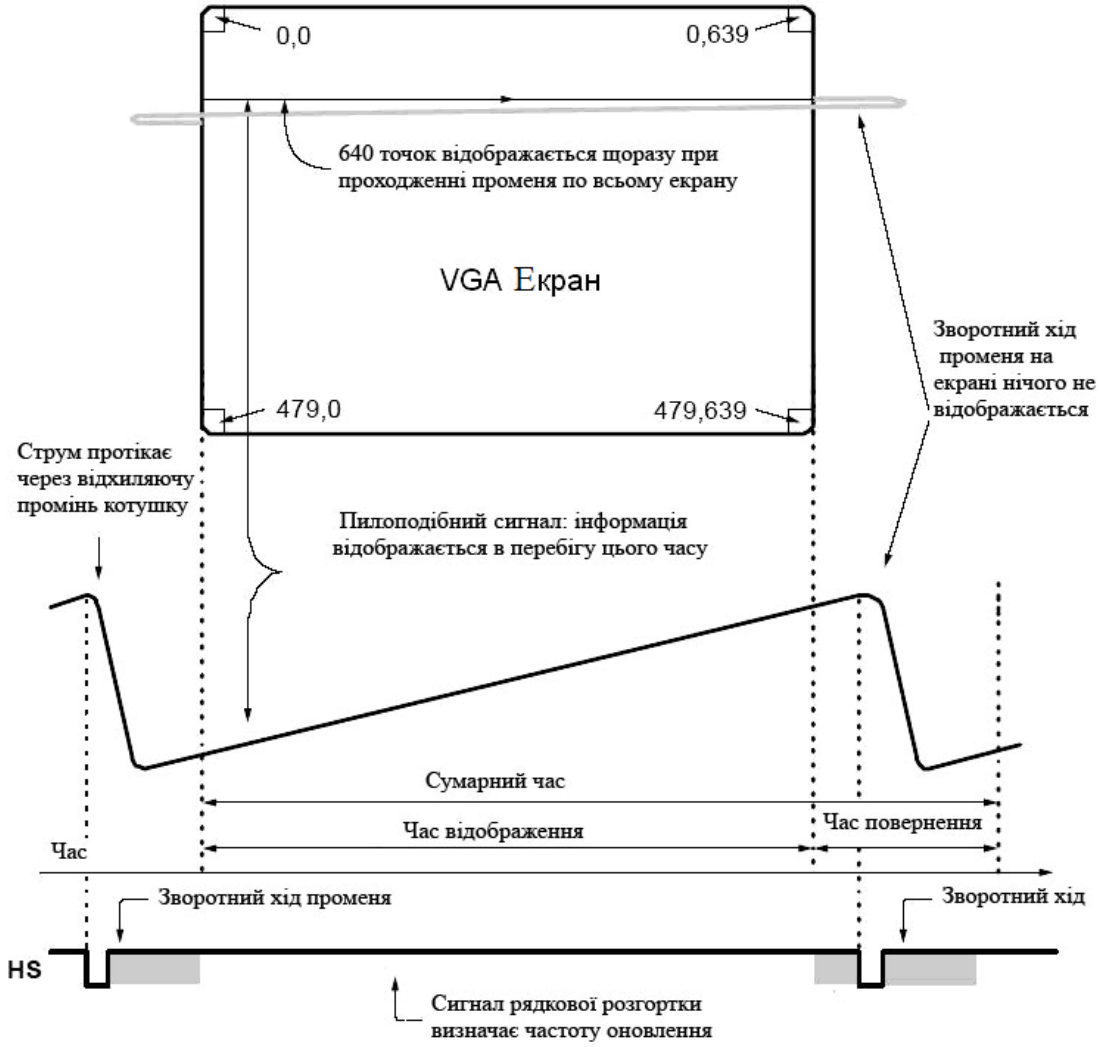


Рис. 8.24.

Як видно з рисунку, контролер VGA формує сигнали строкової і кадрової розгорток. Сигнал VS визначає частоту оновлення зображення на екрані. У табл. 8.2 приводяться часові характеристики сигналів при частоті оновлення зображення 60 Гц.

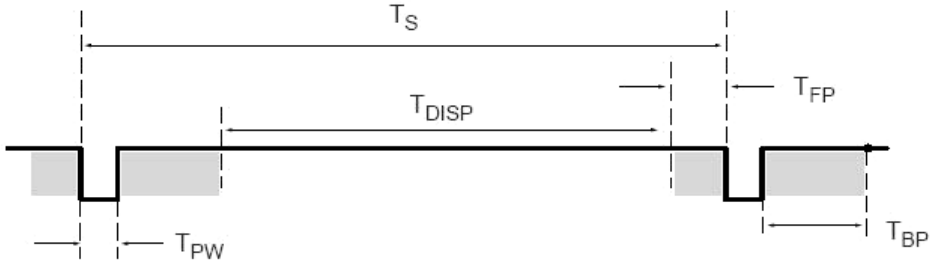


Рис. 8.25.

Табл. 8.2.

Позначення	Параметр	Кадрова розгортка			Строкова розгортка	
		Час	Такти	Лінії	Час	Такти
T_S	Імпульс синхронізації	16,7 мс	416,8	521	32 мкс	800
T_{DISP}	Час відображення	15,36 мс	384,0	480	25,6 мкс	640
T_{PW}	Тривалість імпульса	64 мкс	1,6	2	3,84 мкс	96
T_{FP}	Передній поріг	320 мкс	8,0	10	640 нс	16
T_{BP}	Задній поріг	928 мкс	23,2	29	1,92 мкс	48

Як правило, лічильник синхронізується елементами управління горизонтальної синхронізації. Розшифровані значення лічильника генерують сигнал HS. Цей лічильник відстежує позицію поточного пікселя в даній колонці. Окремий лічильник відстежує вертикальну синхронізацію. Лічильник вертикальної синхронізації збільшується з кожним імпульсом HS. Цей лічильник відстежує поточну дисплейну колонку. Ці два лічильники, що безперервно працюють, формують адресу пам'яті відеобуфера.

8.11. Системи телебачення SECAM, PAL, NTSC

Необхідність звести в один канал сигнали синхронізації, інформацію про колір і яскравість, а також сигнали розпізнавання, призвела до створення складного за структурою стандарту відеосигналу. Жирною лінією на рис. 8.26 виділений сигнал чорно-білого телебачення (ТБ), а частотні насадки (піднесучі) дозволяють передавати інформацію про колір і приналежність до конкретної системи мовлення. Одночасно передається сигнал синхронізації розгорток.

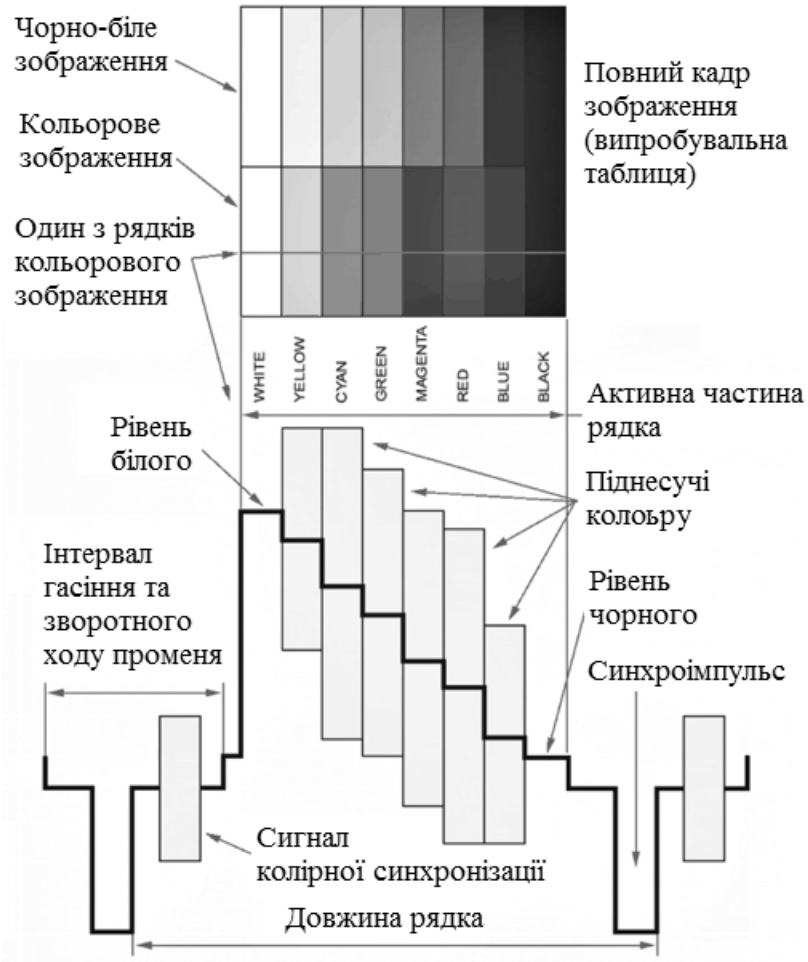


Рис. 8.26.

Після отримання різнецевих колірних сигналів, вони перетворюються в один сигнал в передавальному центрі. Яким чином кодувати сигнали колірності, було вирішено в різних країнах різними способами, і це призвело до появи трьох основних, несумісних один з одним стандартів.

Для роботи телевізійного приймача необхідне джерело сигналів кадрової синхронізації, яке вказує момент початку кадру в сигналі. На початкових стадіях проектування було вирішено використовувати частоту живлячої мережі в якості такого джерела з двох основних причин. По-перше, при використанні раніше створених блоків живлення телевізорів могла виникнути проблема «рухомої смуги» на зображенні в разі неточного збігу частоти кадрів і живлячої мережі. По-друге, телестудії мали б великі проблеми з мерехтінням при створенні телевізійних програм.

Подальші різновиди систем з'явилися з початком кольорового мовлення. Велика частина країн з мережею 60 Гц використовує систему кольорового телебачення NTSC, розроблену в Сполучених Штатах.

Незабаром за NTSC з'явилася її модифікація, яка була названа PAL. Вона прийнята в більшості «50-герцових» країн, у тому числі в Західній

Європі (окрім Франції), а також в деяких «60-герцових» країнах (наприклад, Бразилії).

В кінці 60-х років у Франції багато в чому з політичних причин (захист внутрішніх виробників) була розроблена система SECAM. Вона була широко розповсюджена в східно-європейському блоці країн, в основному для того, щоб мати несумісність із західними передачами. Частота кадрів в SECAM складає 50 Гц (за винятком деяких її варіацій).

Система кольорового телебачення NTSC була розроблена в 1953 році в США Національним комітетом з телевізійних стандартів (National Television Standards Committee). NTSC прийнята як стандартна система телевізійного мовлення також в Канаді, Японії і ряді країн американського континенту. В якості сигналів для передачі інформації про колір в системі NTSC прийняті колірнорізнецеві сигнали. Передача цих сигналів здійснюється в спектрі сигналу яскравості на одній колірній піднесучій.

Окрім експлуатаційних недоліків, пов'язаних зі складним принципом передачі та розділення сигналів колірності – квадратурною модуляцією і синхронним детектуванням, необхідно вказати на велику схильність системи NTSC до спотворень типу «диференційна фаза» і «диференційне підсилення». Перше призводить до спотворень колірного тону, який змінюється залежно від миттєвого значення сигналу яскравості. Друге через нелінійності амплітудних характеристик призводить до спотворень насиченості.

Система PAL (Phase Alternation Line – рядок із змінною фазою), розроблена у ФРН, в своїй основі містить всі ідеї американської NTSC. Особливість PAL полягає в оригінальному способі усунення фазових спотворень, властивих системі NTSC.

У системі PAL фаза піднесучої одного колірнорізнецевого сигналу від рядка до рядка змінюється на 180 градусів. Крім того, в приймачі використовується лінія затримки на час одного рядка (64 мкс). Тобто є два сигнали колірності з відносною затримкою на один рядок. Зміна фази від рядка до рядка на 180° призводить до того, що фазові помилки, однакові по величині, мають різні знаки. Складання напруги на вході лінії затримки з перевернутою напругою на її виході усуває помилку фази.

При очевидних перевагах головним недоліком системи PAL є істотне ускладнення приймача за рахунок введення в його схему додаткових вузлів для затримки сигналу колірності на час одного рядка і періодичної зміни фази колірнорізнецевого сигналу. Слід також відзначити, що спотворення типу «диференційне підсилення» в PAL не компенсуються.

У 1958 р. французький інженер Анрі де Франс винайшов нову систему, що називається SECAM (Sequential Couleur Avec Memoire), в якій був відсутній основний недолік NTSC – спотворення колірного тону, що викликаються нелінійністю частотних, фазових і амплітудних характеристик вузлів телевізійного тракту. У SECAM інформація про

колірний тон не визначається фазовими співвідношеннями сигналів колірності. У перших варіантах (система «Анрі де Франс») інформація про колірний тон передавалася амплітудною модуляцією піднесучої. У більш вдосконаленій системі SECAM колірна інформація передається за допомогою частотної модуляції піднесучої кольору.

Колірнорізнецеві сигнали в SECAM передаються по черзі: протягом одного рядка – сигнал R–Y, протягом наступної – B–Y і т.д. Колірна інформація як для R–Y, так і для B–Y «знімається» через рядок. При цьому передбачається, що в пропущених рядках колірна інформація ідентична сусіднім. Іншими словами, для сигналів колірності повний кадр містить вдвічі меншу кількість рядків, що призводить до відповідного збільшення розмірів забарвлених дрібних деталей по вертикалі. Візуальна чіткість по вертикалі при цьому не знижується, оскільки дрібніші деталі передаються сигналом яскравості Y з повним числом рядків розгортки.

Таким чином, при почерговій (через рядок) передачі сигналів колірності в приймачі в результаті використання елементу пам'яті (лінії затримки) утворюються три вихідні сигнали колірності. Тому дану систему часто називають послідовно-одночасною (або по-французьки Sequential a memoire – послідовна з пам'яттю).

8.12. Порівняння систем SECAM, NTSC і PAL

При порівнянні систем кольорового телебачення зазвичай враховують наступні якісні і техніко-економічні показники:

1. чутливість до спотворень;
2. якість кольорового зображення;
3. сумісність с чорно-білим ТБ;
4. оцінка особливостей систем;
5. можливість і особливості відеозапису.

Виходячи з цих показників, стисло порівняємо існуючі системи.

1. Нерівномірність частотних і фазових характеристик тракту передачі в тій області частот, де розташовані складові спектру сигналу колірності, призводить в системі NTSC до спотворення зображення. Ці спотворення виявляються на екрані на межах ділянок, що різко відрізняються за кольором. З цієї причини до частотних і фазових характеристик різних елементів апаратури в системі NTSC пред'являються досить жорсткі вимоги. Сказане повною мірою відноситься також до системи PAL. Використання в системі SECAM частотної модуляції для передачі кольорових сигналів дозволяє не пред'являти до рівномірності частотних характеристик жорсткіших вимог, ніж для систем чорно-білого телебачення. Всі спотворення сигналу колірності, що виникають через нерівномірності частотних характеристик, усуваються в амплітудних обмежувачах приймача. В цьому відношенні система SECAM має значні переваги перед системою NTSC і PAL. І хоча спотворення градацій в сигналі яскравості при цьому залишаються, проте вони помітні не більше,

ніж в чорно-білому телебаченні. Система PAL відносно спотворень типу «диференційне посилення» не має переваг перед NTSC, оскільки і в ній застосовується той самий метод модуляції піднесучої.

2. Потенційно найвищу якість зображення забезпечує NTSC. При цьому її основним мінусом залишається знижена вертикальна роздільна здатність (всього 525 рядків) і непристосованість передачі на великі відстані по радіорелейних лініях.

У реальних же умовах телеприйому, коли ефір наповнений перешкодами, а віддаленість телецентру лише сприяє зростанню шумів, пріоритет буде на користь SECAM – внаслідок того, що колірні сигнали передаються по черзі в різний час, перехресні спотворення майже відсутні. Для передачі сигналів SECAM можуть застосовуватися звичайні радіорелейні лінії.

Для звичайного користувача в умовах достатньої потужності телевізійного сигналу і мінімумі перешкод відмінностей в якісних показниках зображення на екранах телевізійних приймачів NTSC, PAL і SECAM майже не спостерігається.

3. В світі практично не ведуться чорно-білі передачі (навіть старі чорно-білі фільми транслюються з сигналами колірності), а кількість чорно-білих телевізорів, що випускаються, швидко скорочується. Більш важливою причиною сьогоднішньої несумісності є існуючий парк кольорових ТБ-приймачів, пристосованих здебільше для роботи в одному із стандартів.

Сигнали колірності утворюють на екрані чорно-білого приймача перешкоду у вигляді дрібної сітки. У NTSC найменше помітна її дія, оскільки при передачі чорно-білих ділянок зображення в NTSC сигнали колірності відсутні взагалі.

У SECAM через застосування частотної модуляції піднесуча сигналів колірності не може бути повністю пригнічена. Для усунення заважаючої дії сигналів колірності в системі SECAM застосовується комутація фази піднесучої.

У PAL так само, як і в системі SECAM, застосовується комутація фази піднесучої. Проте, цей захід не забезпечує повного усунення перешкод, і в результаті система PAL по цьому показнику стоїть нижче SECAM.

4**.** На прикладі переваг і недоліків розглянемо технічні особливості стандартів.

NTSC/525

Переваги: Більш висока частота кадрів – використання частоти кадрів 30 Гц (насправді 29,97 Гц) призводить до зменшення помітності мерехтіння зображення. Висока точність редагування кольору – можливо редагувати будь-які 4 поля без впливу на колір. Менш помітні шуми на зображенні, досягається краще відношення сигнал/шум, ніж в Pal/625.

Недоліки: Менше число рядків розгортки – знижена вертикальна чіткість, помітніша рядкова структура на екранах з великою діагоналлю. Більш виражені муар, інтерференція і перехресні спотворення – це відбувається через більші вірогідності взаємодії з монохромним сигналом зображення на нижчій частоті піднесучої. Зміна відтінку – варіації фази колірної піднесучої викликають зсув у відображенні кольорів, примушуючи оснащувати приймачі регулюванням відтінку (Tint). Нижча по відношенню до PAL контрастність – значення гами складає 2,2, тоді як в PAL/625 вона дорівнює 2,8.

PAL/625

Переваги: Більш детальна картинка – більше число рядків розгортки, а також більш широка смуга сигналу яскравості. Стійкість відтінків – завдяки інверсії фази піднесучої на кожному подальшому рядку, будь-яке фазове спотворення буде послаблене. Більш вищий рівень контрасту – значення гамми 2,8 проти 2,2 в NTSC /525.

Недоліки: Більш помітне мерехтіння через нижчу частоту кадрів (25 кадрів/сек.). Більш помітні шуми – вимога підвищеної частоти піднесучої призводить до погіршення відношення сигнал/шум в PAL/625 в порівнянні з NTSC/525. Втрата точності редагування кольору – через чергування фази колірного сигналу редагування може бути здійснене з точністю ±4 кадри (8 полів). Зниження колірної насиченості при незмінному відтінку – точність кольорів досягається за допомогою втрати інформації про різницю фаз сигналів відтінку і насиченості.

SECAM/625

Переваги: Стійкість відтінку і постійність насиченості. Більша вертикальна роздільна здатність – в SECAM використовується більша кількість рядків розгортки, ніж в NTSC/525.

Недоліки: Більш помітніше мерехтіння. Неможливе змішування двох синхронних сигналів кольору SECAM – більшість ТБ-студій в SECAM-країнах працюють в PAL і переводять передачі в SECAM лише для мовлення. Крім того, домашнє устаткування S-VHS, Hi8 записує в PAL і лише при програванні відбувається перекодування в SECAM. Регулярні шумові структури на зображенні – частотна модуляція призводить до появи регулярних шумових структур навіть на не кольорових об'єктах. Несумісність між різними версіями SECAM – деякі з варіантів SECAM (ефір і відео) несумісні один з одним.

5. Можливість відеозапису підтримують всі базові системи. Найбільш поширеною системою для відеозапису є PAL.

Останнім часом все більш активно розвивається телебачення високої чіткості і цифрове мовлення. Для першого ще актуальні розглянуті системи колірності. Для цифрового ТБ основоутворюючі кольори (R, G, B) вже не потребують аналогових систем кодування, що призводять до погіршення якості. Цифровий кодер здатний без втрат передати повний спектр кольорового ТБ-сигналу. На приймальному кінці відбувається

цифрове декодування первинних сигналів, які минуючи системи перетворення, потрапляють прямо на гармати кінескопа. Такому зображенню не властиві спотворення, взаємовплив RGB-каналів, а також роздвоювання.

8.13. Цифрове телебачення

DVB Project (Digital Video Broadcasting Project – Проект цифрового відеомовлення) – це організація, що займається розробкою стандартів в області цифрового телебачення для Європи. В основі стандартів DVB лежить стандарт кодування рухомих зображень і звукового супроводу MPEG-2. В стандартах DVB окрім методів кодування і параметрів транспортного потоку, визначені методи перешкодостійкого кодування, канального кодування, модуляції несучих частот, передачі додаткової інформації, захисту інформації від несанкціонованого доступу (умовного доступу до програм) та інші аспекти цифрового телевізійного мовлення. При цьому для всіх видів телевізійного мовлення загальною основою є використання методів MPEG-2 для стиснення зображення і звуку, формування транспортного потоку на основі транспортного потоку MPEG -2, кодування за допомогою кодів Ріда-Соломона. Тим самим досягається значна міра уніфікації.

Кабельні канали телевізійного мовлення знаходяться в найбільш сприятливих умовах з точки зору забезпечення перешкодостійкості, оскільки вони захищені від дії атмосферних та індустріальних перешкод, і в них є можливість підтримувати досить високе значення відношення сигнал/шум (не менше 30 дБ) за рахунок використання проміжних підсилювачів. Тому в кабельному телебаченні, як правило, не використовується внутрішнє кодування за допомогою згортальних кодів. Основне завдання в кабельному телевізійному мовленні – забезпечити найбільш ефективне використання смуги частот, щоб передавати по наявних кабельних мережах максимальну кількість телепередач. З цією метою в кабельних мережах відповідно до стандарту DVB-C використовується багатопозиційна квадратурна амплітудна маніпуляція (КАМ). В даний час застосовуються 16-, 32- і 64-позиційна КАМ. У перспективі передбачається застосування 256-позиційної КАМ. Швидкості передачі двійкових символів, що отримуються при різних кількостях позицій КАМ, приведені в табл. 8.3. Повна швидкість передачі канальних символів (другий стовпець) приблизно однакова для трьох варіантів КАМ. Повна швидкість передачі двійкових символів (третій стовпець) отримується множенням швидкості передачі канальних символів на кількість бітів в символі і виявляється вищим за швидкість передачі корисних даних за рахунок введення додаткових байтів при перешкодостійкому кодуванні Ріда-Соломона.

Табл. 8.3.

Вид модуляції	Швидкість передачі, МСимв/с	Повна швидкість передачі, Мбіт/с	Швидкість передачі корисних даних, Мбіт/с
16-КАМ	6,89	27,56	25,2
32-КАМ	6,92	34,60	31,9
64-КАМ	6,84	41,04	38,9

У передавальній частині системи кабельного цифрового телебачення виконуються наступні операції:

1. Перетворення аналогового телевізійного сигналу і сигналів звукового супроводу в цифрову форму.
2. Стиснення цифрового телевізійного сигналу і сигналів звукового супроводу відповідно до стандарту MPEG-2 і формування відповідних елементарних потоків.
3. Формування транспортного потоку відповідно до стандарту DVB. У транспортний потік включаються елементарні потоки декількох телепередач і різні дані. Транспортні пакети об'єднуються по 8. У першому пакеті кожної вісімки синхрогруппа інвертується, тобто замість шістнадцяткового числа 0x47 передається 0xB8. Це необхідно для циклічної синхронізації в приймачі.
4. Скремблювання для захисту програми від несанкціонованого доступу і для отримання більш рівномірного розподілу потужності по ширині смуги каналу зв'язку.
5. Кодування з використанням кодів Ріда-Соломона. При цьому довжина транспортних пакетів зростає з 188 до 204.
6. Змішування для захисту від пакетних помилок.
7. Перетворення байтів, що передаються, в символи КАМ. Наприклад, при використанні 64-позиційної 64-КАМ кожні 3 байти перетворюються в 4 шестибітових символи.
8. Диференціальне кодування двох старших бітів для підвищення перешкодостійкості.
9. Перетворення символів в імпульси напруги, що подається потім на модулятори квадратурних складових *I* та *Q*. Щоб обмежити спектр модульованого сигналу, фронти і зрізи імпульсів згладжуються ФНЧ.
10. Модуляція на проміжній частоті.
11. Перенесення модульованого сигналу на несучу частоту каналу і підсилення потужності.

У приймачі повинні виконуватися відповідні зворотні операції. Структурна схема приймального тракту кабельного цифрового телебачення показана на рис. 8.27.

Вхідний сигнал з кабельної лінії поступає на тюнер, в якому, як в звичайному телевізорі, виділяється потрібний канал, і сигнал цього каналу переноситься на проміжну частоту. Потім в перетворювачі частоти виконується додаткове пониження частоти. Для цього

використовується несуча частота, відновлена в демодуляторі КАМ або в самому блоці перетворювача частоти. Смуга частот сигналу після пониження частоти зазвичай складає 3 ... 11 МГц.

Далі сигнал проходе керований підсилювач, коефіцієнт підсилення якого визначається сигналом автоматичного регулятора рівня (АРР) з демодулятора. Розмах сигналу при цьому узгоджується з діапазоном вхідної напруги АЦП. Для прийому сигналів 16-КАМ, 32-КАМ і 64-КАМ досить АЦП, що має 8 двійкових розрядів, а для прийому 256-КАМ необхідно 9 двійкових розрядів. Дискретизація виконується з частотою канальних символів, тобто близько 7 МГц. Тактові імпульси (ТІ) формуються в демодуляторі.

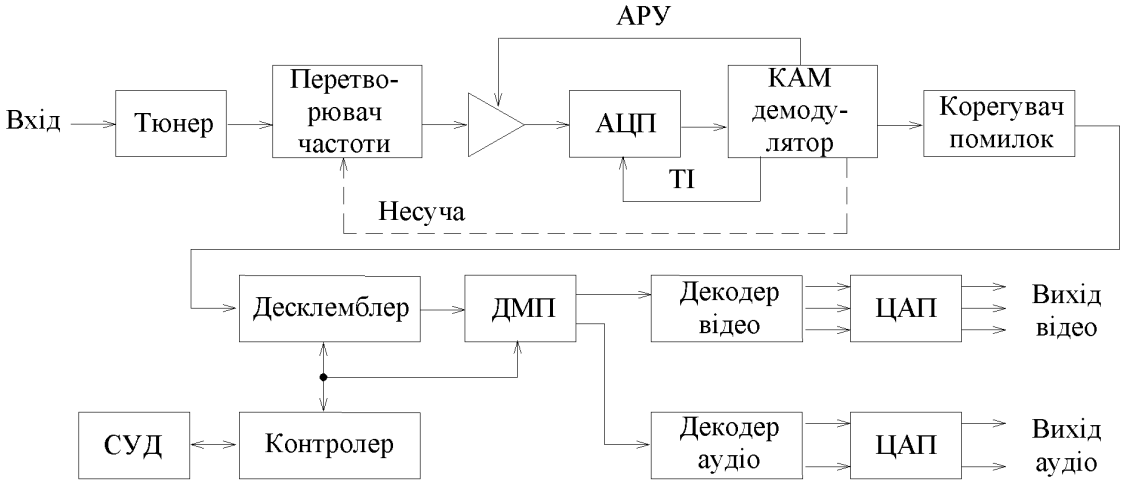


Рис. 8.27.

Цифровий сигнал, що поступає на демодулятор, перетворюється в сигнали квадратурних складових *I* та *Q*, по яких відновлюються канальні символи. Потім з цих символів формуються байти вихідного потоку даних. Наприклад, в разі 64-КАМ з 4 шестибітових канальних символів формуються 3 вихідних байта. У демодуляторі відновлюються частота несучої й тактові імпульси канальних символів, які використовуються як в самому демодуляторі, так і в передуючих блоках.

Потім потік даних поступає в блок корекції помилок (коректор помилок), в якому виконуються відновлення даних і декодування коду Ріда-Соломона. В результаті відновлюються розміри транспортних пакетів (по 188 байтів) і вихідний порядок байтів в пакетах.

Далі потік даних поступає на дескремблер. Дескремблірований транспортний потік приходить на демультиплексор (ДМП), в якому з транспортного потоку вибираються пакети, що відносяться до вибраної програми, і формуються елементарні потоки відео, звукового супроводу і даних. Управління дескремблером і демультиплексором здійснює контроллер. Демультиплексор вибирає з транспортного потоку пакети з

таблицями програм і передає їх на контроллер, який забезпечує відображення інформації про програми в транспортному потоці на екрані. По командах користувача вибирається одна з програм, і дані про ідентифікатори пакетів, що відносяться до цієї програми, передаються в демультиплексор, щоб надалі вибирати ці пакети. Допоміжні дані до вибраної програми (субтитри і т.п.) поступають з демультиплексора на контроллер, який передає їх далі на блок відтворення додаткової інформації.

Контролер пов'язаний також з системою умовного доступу (СУД), яка включає засоби, що забезпечують доступ користувача до тих платних програм, за які він заплатив. Дані для дескремблювання програм передаються у відповідних пакетах транспортного потоку. Для дескремблювання безкоштовних програм і пакетів з інформацією загального користування додаткові дані не потрібні.

Таким чином, в приймачі цифрового телебачення вибір програми здійснюється в два етапи. Спочатку вибирається канал ТБ-мовлення, в якому передається транспортний потік, що містить декілька програм. Потім вибирається одна з програм в цьому транспортному потоці.

Елементарні потоки відео і звуку з демультиплексора поступають на відповідні декодери MPEG-2. На виходах декодера відео отримуються яркісний та колірнорізнецевий сигнали в цифровій формі. Ці сигнали поступають на ЦАП (блок ЦАП на рис. 8.27, містить 3 ЦАП), де перетворюються в аналогову форму, а потім поступають на аналогові схеми управління яскравістю і контрастністю, підсилення і, нарешті, на катоди кінескопа. На виходах декодера аудіо отримуються сигнали каналів звукового супроводу, які перетворюються в аналогову форму і поступають на блоки відтворення звуку.

8.14. Основні вузли і блоки пристроїв цифрового телебачення

Цифровий телевізійний прийом в даний час зазвичай здійснюється або інтегрованим декодером, або приставкою до телевізора. Кодування вихідного сигналу – це MPEG-2 компресія, не дивлячись на те, що в залежності від форматів мовлення модуляційні схеми, канальне кодування і т.п. в різних країнах різні. Узагальнена структурна схема інтегрованого приймача-декодера IRD представлена на рис. 8.28.

Перш за все, треба виділити і підсилити тюнером високочастотний радіосигнал, налаштуватися на потрібний канал і перетворити радіочастоту в більш низьку – проміжну частоту (ПЧ). В разі прийому зі супутника, первинне перетворення з пониженням частоти несучої здійснюється малошумливим перетворювачем LNC, або малошумливим блоком LBN, що знаходиться в голівці антени.

Далі сигнал ПЧ демодулюється у вхідному блоці і перетворюється в сигнал основної смуги. Цей процес може бути аналоговим або цифровим. В разі використання QPSK-модуляції (для супутникової

передачі) або N-QAM-модуляції (для кабельної) демодулятором вхідного блоку виділяються два демодульованих сигнали *I* та *Q*, а при використанні OFDM-модуляції (для наземної передачі) – один багаторівневий вихід.

Слідом за демодуляцією у вхідному блоці йде Вітербі-декодування – процес, зворотний внутрішньому кодуванню. При його виконанні виділяється один цифровий потік, який піддається процедурі деперемежування для отримання правильного порядку у часі. З метою виділення транспортних 188-бітових пакетів потік потім підлягає процедурі декодування Ріда-Соломона. Після цього етапу сигнал приймає вид транспортного потоку.

Потім сигнал має бути демультиплексований, тобто роздений на окремі складові складеного інформаційного потоку, отриманого мультиплексором.

Для демультиплексування необхідно виконати так звану PID-фільтрацію, яка включає в себе:

– встановлення синхронізму роботи декодера. Для цього система автоматично знаходить перший байт в заголовку транспортного пакету;

– пошук транспортних пакетів з ідентифікатором PID=0 для побудови таблиці з'єднання програм PAT, а також пошук пакетів з ідентифікатором PID=1 для побудови при необхідності таблиці умовного доступу CAT;

– використання PAT для побудови таблиць планів програм PMT, а також таблиць CAT для надання глядачеві інформації про доступні для нього програми в мультиплексованому потоці.

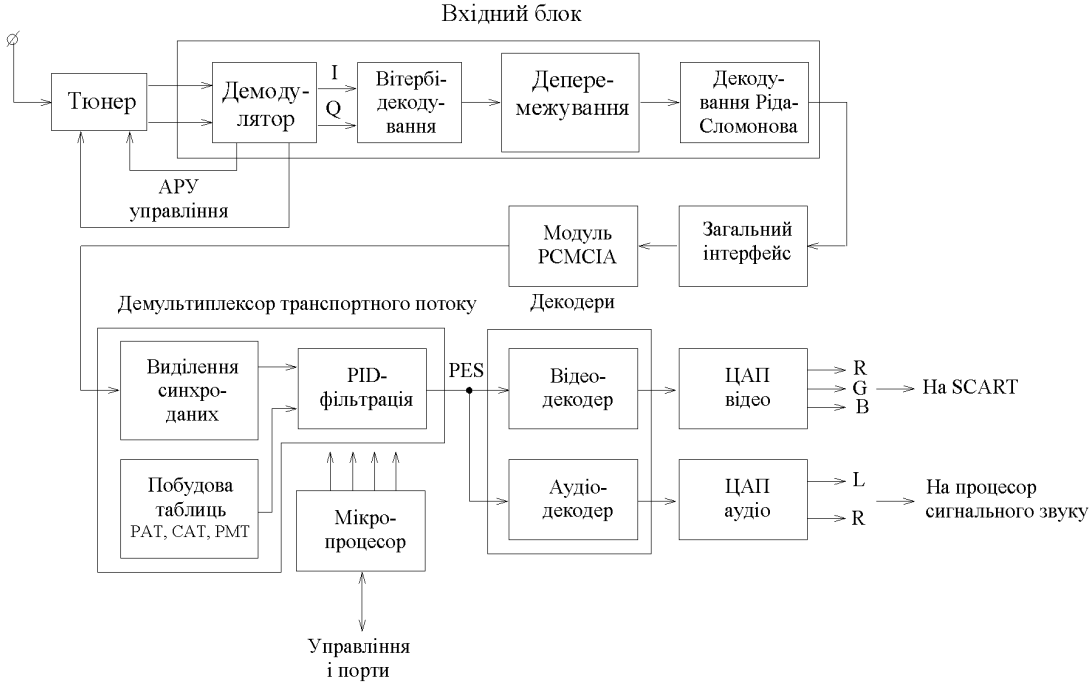


Рис. 8.28.

Глядач може вибрати необхідну програму, подавши відповідну команду з ПДУ, сигнал з якого обробляється мікропроцесором. Отримавши команду, приймач «впізнає» відповідно до цієї команди і інформації з таблиць PAT та PMT ідентифікаційні PID-номери транспортних пакетів, необхідних для реконструкції пакетних елементарних потоків, складаючих задану глядачем програму.

Таким чином приймач починає демультиплексувати транспортний потік, відновлюючи вихідні пакетні елементарні аудіо- і відеопотоки PES, які подаються на аудіо- і відеодекодерах для MPEG-декодування аудіо- і відеосигналів, внаслідок чого виходять цифрові потоки даних, що піддаються зворотньому перетворенню в низькочастотні аналогові відео- і аудіосигнали. Відеосигнали на виході ЦАП – це або сигнали основних кольорів R, G, B, або декодовані сигнали систем PAL, NTSC або SECAM, що поступають на відеопроцесор.

Сигнали звуку на виході ЦАП – це сигнали лівого і правого каналів, що поступають на процесор обробки звуку, де відбуваються регулювання гучності, балансу, тембрів ВЧ та НЧ і т.д.

На рис. 8.29 показаний варіант узагальненої структурної схеми приставки цифрового телевізійного декодера приймача, яка може бути підключена до звичайного телевізора.

Канальний декодер включає в себе цифровий демодулятор і вузол корекції помилок. Декодер приймає цифрові модульовані ВЧ сигнали і складає з них цифровий транспортний потік, що містить пакети мультиплексированних програм. Цифровий транспортний потік поступає на демультиплексор, який розпізнає кожен пакет по програмному ідентифікатору, що знаходиться в ньому, і перекомпоновує дані з метою створення пакету вибраної програми.

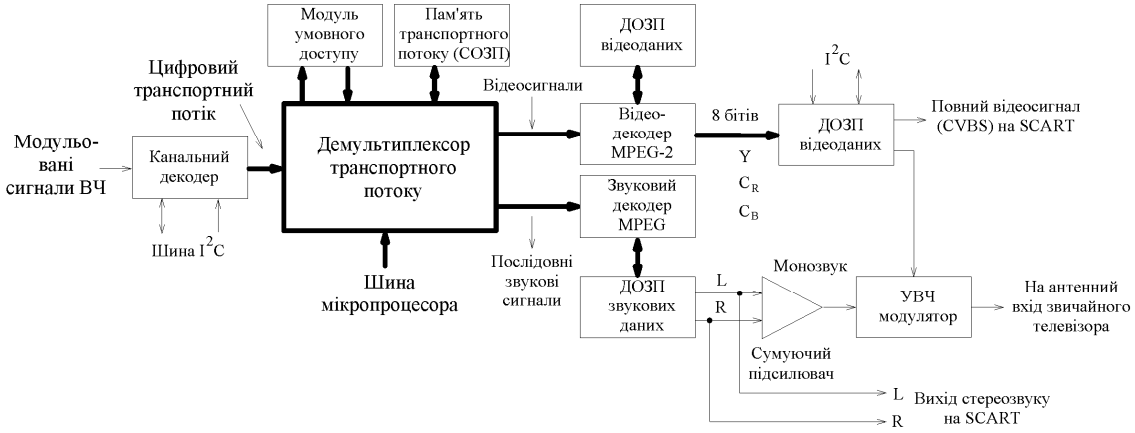


Рис. 8.29.

Якщо пакет скремблюється, тобто цифровий сигнал перетворюється, внаслідок чого руйнуються кореляційні зв'язки між символами вихідного сигналу, змінюються його статичні властивості, і він набуває властивостей випадкового сигналу, то він поступає в модуль

умовного доступу, призначений для визначення можливості глядачем користуватися вибраною програмою. Якщо це недоступно, то обробка сигналів припиняється. Якщо це робити можна, то сигнал повертається в демультиплексор, звідки два набори цифрових сигналів подаються на відповідні декодери: один набір – однобайтові (8 бітів) відеосигнали, інший – послідовні сигнали звуку. Для зберігання і подальшої видачі відео- і звукових сигналів використовується швидка пам'ять транспортного потоку ємністю 8К×8.

Відеодекодер стандарту MPEG-2 перетворює відеосигнали в сигнали яскравості Y і колірності CR і CB. Велика пам'ять відеоданих в мікросхемі ОЗП необхідна для одночасного зберігання інформації про декілька кадрів.

Цифрові сигнали яскравості і колірності передаються в кодер PAL, який перетворює їх в повний аналоговий відеосигнал CVBS, що поступає потім в модулятор.

Звуковий канал містить декодер MPEG, який декодує звуковий сигнал за тими ж правилами, що і при його кодуванні в передавачі. Звуковий декодер формує лівий (L) і правий (R) аналогові звукові сигнали. Мікросхема ОЗП звукових сигналів необхідна для їх зберігання в пам'яті, і створює затримку для забезпечення синхронізації звуку і зображення. Затримка необхідна, оскільки обробка відеосигналів виконується довше, ніж обробка звукових сигналів.

Звукові сигнали обох каналів поступають на підсилювач, де створюється сигнал монофонічного звуку, який подається на модулятор. Сигнал з модулятора може бути поданий на антенний вхід звичайного телевізора.

Для управління системою використовується мікроконтроллер (рис. 8.30), який виконує всю необхідну програмну обробку. Зв'язок мікроконтроллера зі всіма іншими вузлами приймального пристрою здійснюється по одній або декількох цифрових шинах управління як послідовним, так і паралельним способом.

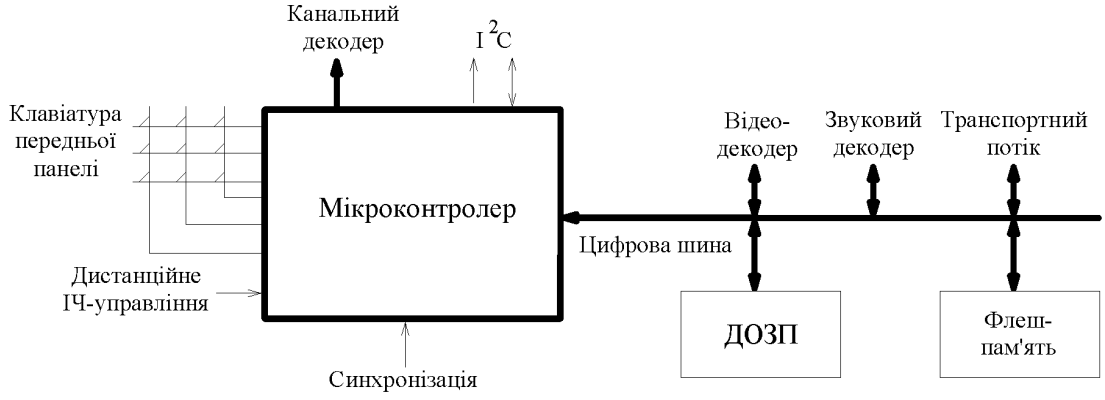


Рис. 8.30.

Проте частіше для управління використовується послідовний спосіб передачі даних, а для передачі даних корисних сигналів цифрових пристроїв – паралельний.

Зазвичай в телевізійних схемах зустрічаються три різні типи системних шин з послідовною передачею інформації:

– шина IM (IntermetaM-Bus);

– шина Томсона (M3L);

– шина I^2C.

У конкретному телевізорі може використовуватися дві і більше системних шини, а також можливі комбінації різних системних шин. На рис. 8.31 представлена спрощена структурна схема включення вузлів телевізора за допомогою цифрової шини I^2C.

Вимірювання рівнів сигналів в системній шині при пошуку несправності в телевізорі часто швидко ведуть до її локалізації. В першу чергу перевірці підлягає напруга живлення шини, оскільки навіть малі його відхилення від номінального значення можуть призвести до припинення роботи телевізора. Особливу увагу необхідно звертати на наявність напруги живлення, особливо у випадках, коли з'являється нерегулярна несправність. Пульсації напруги можуть складати декілька мілівольт, і в сумнівних випадках перевірку напруги живлення треба проводити за допомогою осцилографа.

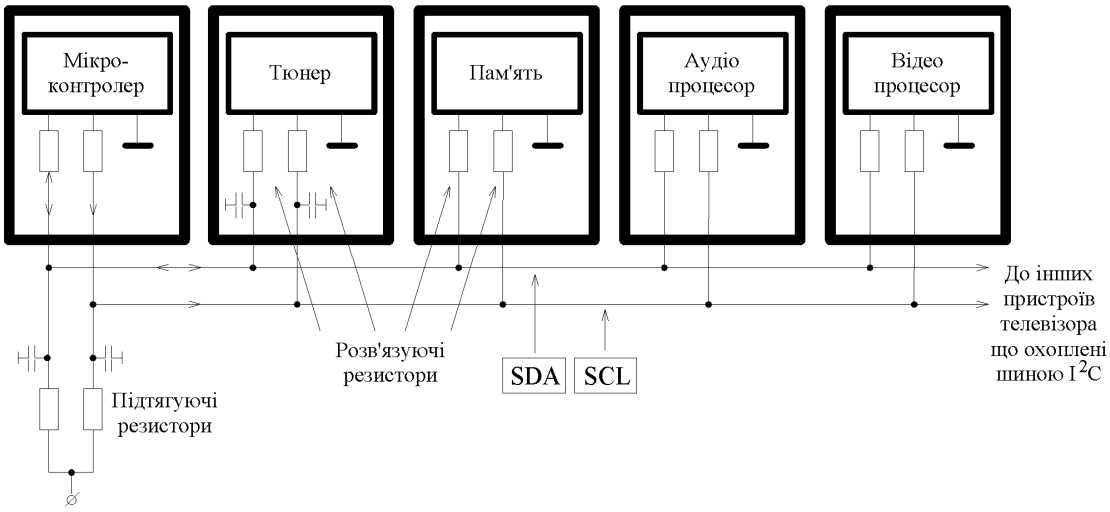


Рис. 8.31.

Вимірювання напруги до і після підтягуючих резисторів вже дає інформацію про вірогідні несправності в пристроях, підключених до системної шини, при великому падінні напруги на них. А вимірювання на входах вузлів або мікросхем (після відповідних розв'язуючих резисторів) можуть дати відповідь про несправність функціонального вузла.

Мікроконтролер системи управління декодує ІЧ сигнал від ПДУ і сигнали від клавіатури передньої панелі телевізора. Він управляє

канальним декодером, включаючи селектор каналів (тюнер) і цифровий демодулятор, відео і звуковим декодерами, декодером PAL і транспортним потоком. З мікроконтролером по цифровій шині зв'язані ОЗП і флеш-пам'ять.

Канальний декодер (рис. 8.32), що називається також зовнішнім інтерфейсом, є одним з основних частин цифрової приставки до телевізора. Селектор змішує вхідний модульований ВЧ сигнал з сигналом гетеродина (ГУН), внаслідок чого виходить модульовані сигнали *I* та *Q*.

Для передачі на цифровий демодулятор ПЧ треба перетворити їх в цифрову форму за допомогою АЦП. Частота вибірки повинна перевищувати частоту символів в сигналі, що передається, принаймні в 2 рази.

Цифровий демодулятор представляє собою мікросхему процесора цифрової обробки сигналів, який може бути демодулятором сигналів супутникового прийому або демодулятором COFDM сигналів наземного прийому. Демодулятор управляється і програмується системним мікроконтролером по цифровій шині I^2C. Він оцінює потужність вхідного сигналу і посилає сигнал АРУ на селектор.

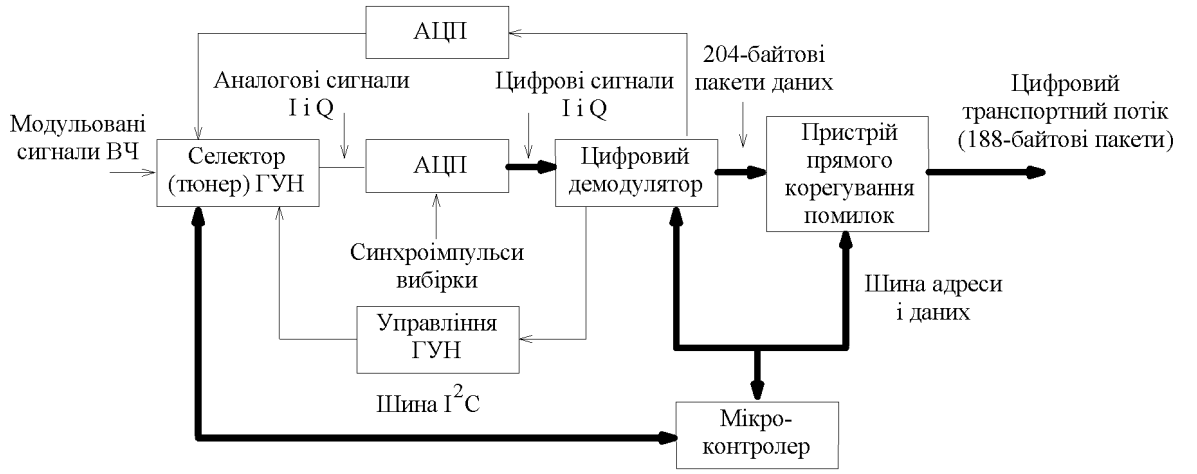


Рис. 8.32.

Демодулятор через схему управління ГУН управляє гетеродином селектора. Вихідні дані цифрового демодулятора складаються з 204-байтових пакетів даних (звукових для відео або обслуговування програм). Потім транспортний потік прямує на пристрій прямої корекції помилок. Цей пристрій використовує контрольні байти корегуючого коду для прийняття рішення залежно від отриманих даних. Він визначає, чи містять отримані пакети даних які-небудь помилки. Якщо це так, той пристрій намагається скорегувати їх. При невдалій спробі пристрій корекції помітить пакети, які містять помилки, і вони не піддаватимуться подальшій обробці.

В кінці цього процесу формується транспортний потік, який містить 188-байтові мультиплексовані пакети, що відносяться до чотирьох або п'яти різних телепередач. Перед відновленням зображення пакети, що відносяться до вибраної програми, мають бути розпізнані і розміщені в необхідному порядку, що і відбувається в транспортному демультиплексорі.

Транспортний демультиплексор (рис. 8.33) приймає на свій вхід 8-бітовий транспортний потік, що містить відео, звукову або службову інформацію, що відноситься до чотирьох або п'яти різних програм, представлений у вигляді 188-байтових пакетів. Кожен пакет містить слово синхронізації для входження в синхронізм, ідентифікатор пакету, опорний сигнал синхронізації програми і біти відео, звукових і службових даних. Якщо використовується скремблювання (шифрування), вхідні дані передаються на модуль умовного доступу по 8-бітовій шині CAM DATAOUT, якщо доступ дозволений, то дані повертаються по 8-бітовій шині CAM DATAIN. Доступом управляє мікроконтролер, який може звертатися до інтерфейсу кредитної карти, щоб встановити можливість глядача користуватися підпискою на вибрану програму.

Отже, основним завданням транспортного демультиплексора є ідентифікація пакетів, що відносяться до вибраної програми, і використання їх міток часу для синхронізації вибраних пакетів між собою. Потім демультиплексор фільтрує вибрані пакети і передає відеодані на відеодекодер MPEG-2 по шині розрядністю 8 біт.

Звукові пакети перетворюються демультиплексором в послідовність звукових даних, які подаються на звуковий декодер MPEG-2. Цією операцією управляє мікроконтролер, який окрім іншого, перевіряє службові пакети програм і направляє відповідні інструкції по управлінню і обробці на демультиплексор.

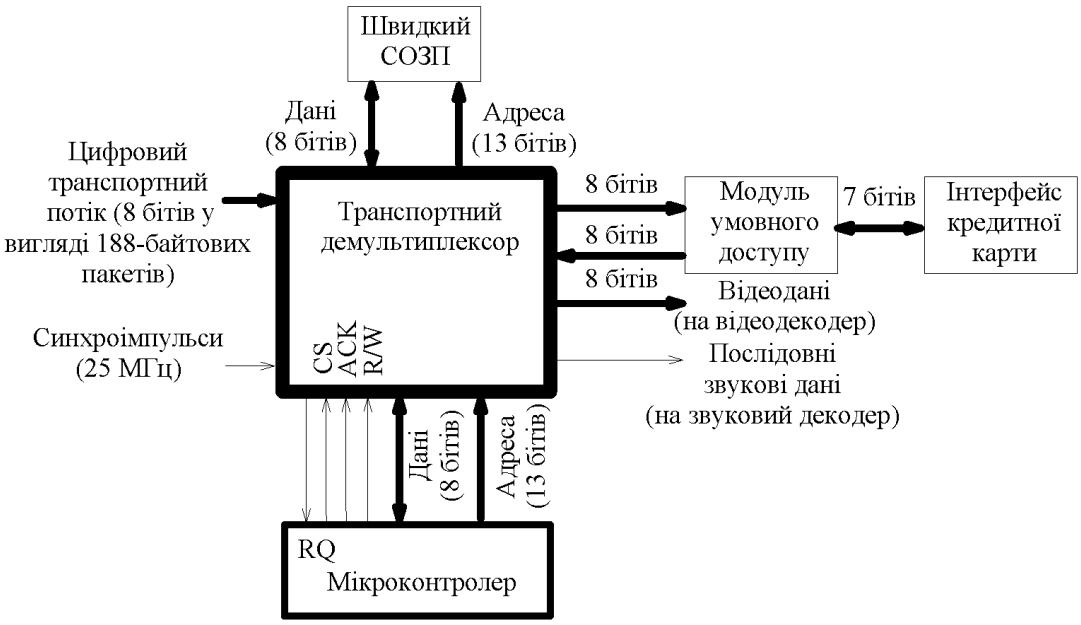


Рис. 8.33.

Для зберігання вибраних відео- і звукових пакетів і подальшій їх видачі, використовується швидкий ОЗП (час доступу 20...25 нс). Для цієї мети використовуються спеціальні шини адреси (13 бітів) і даних (8 бітів) ОЗП мікроконтролера. Крім того, використовуються декілька ліній управління: зчитування/запис R/W, підтвердження ASK і вибір кристалу CS, а також одна або більше ліній запиту переривання IRQ. Демультиплексор використовує запити переривання зокрема для того, щоб, наприклад, проінформувати про переповнення буфера ОЗП. Системна синхронізація транспортного демультиплексора складає 25 МГц, в той час як програмна опорна синхронізація потоку даних складає 27 МГц.

Відеодекодер MPEG-2 (рис. 8.34) призначений для відновлення вихідних сигналів. Процес включає розпаковування даних, реконструювання зображення за кадрами і відтворення вихідних складових яскравості Y і колірності Cr і Cв для кожного кадру. Щоб реконструювати зображення, треба одночасно зберігати кадри, що передаються, і провести необхідні порівняння між ними для відновлення повних кадрів. Тому необхідна пам'ять великого об'єму.

Після відновлення зображення дані пікселів, пов'язані з яскравістю і колірністю, передаються на кодер PAL по 8-розрядній мультиплексованій шині даних (мультиплексованний CVBS). Кодер PAL формує стандартний телевізійний сигнал (625 рядків в кадрі й 25 кадрів в секунду). Початок кожного рядка розгортки визначається сигналом HSYNC, що поступає з кодера на відеодекодер. Непарні і парні поля вказуються сигналами ODDE і EVEN відповідно. Повний відеосигнал CVBS подається на модулятор.

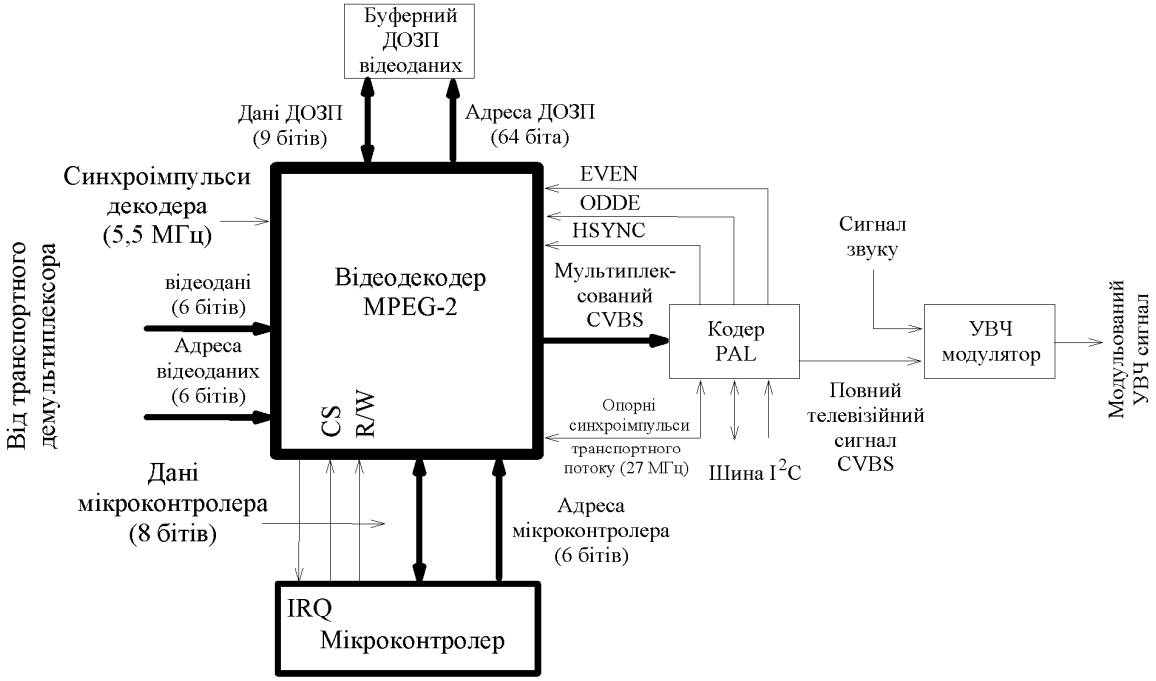


Рис. 8.34.

Для синхронізації процесу збору даних відеодекодер і кодер PAL тактуються опорними програмними синхроімпульсами транспортного потоку частотою 27 МГц.

Кодер PAL програмується і управляється мікроконтролером по 8-розрядній шині даних і 6-розрядній адресній шині з використанням декількох управляючих сигналів, у тому числі запису/зчитування R/W і вибору кристалу CS.

Для запиту переривання відеодекодеру MPEG-2 надається один сигнал IRQ. Кодер PAL управляється також від мікроконтролера по цифровій шині I^2C, по якій мікроконтролер встановлює робочі параметри кодера.

Звуковий декодер MPEG-2 (рис. 8.35) – це мікросхема, яка отримує послідовні цифрові звукові сигнали від транспортного демультиплексора і виконує необхідну обробку сигналів, формуючи два (лівий і правий) послідовні звукові канали з імпульсно-кодовою модуляцією (ІКМ). Звуковий декодер може формувати сигнали при трьох різних частотах вибірки 32; 44,1 і 48 кГц.

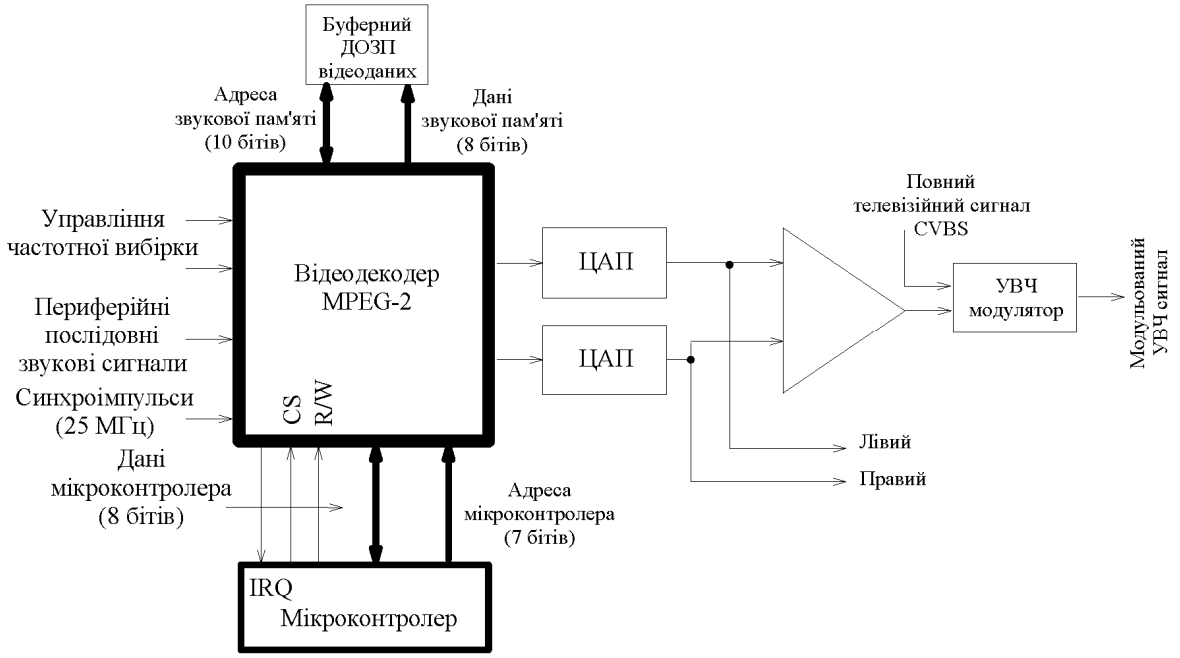


Рис. 8.35.

Інформацію про фактичну частоту вибірки, що використовується передавачем, надає транспортний демультиплексор, який витягує її з транспортного потоку, що поступає.

Зберігання об'єму звукових даних здійснює ОЗП, який забезпечує також затримку сигналу. Декодер звуку управляється і програмується мікроконтролером по 7-розряжній адресній шині, 8-розрядній шині даних і управляючим лініям R/W і CS. За допомогою запиту переривання IRQ, що надається, звуковий декодер може інформувати мікроконтролер про деякі виникаючі збої і запрошувати переривання.

Схема модулятора (рис. 8.36) складається з безпосередньо модулятора з синтезатором частот ФАПЧ і підсилювача петлі. У склад модулятора входять:

– синтезатор частот ФАПЧ, що управляється по цифровій шині I^2C;

– амплітудний модулятор;

– звуковий генератор для звукової піднесучої;

– схема відновлення постійної складової відеосигналу;

– змішувач з регулятором індексу модуляції.

Налаштування модулятора здійснюється постійною напругою 0...24 В, отриманою в синтезаторі частот ФАПЧ. Налаштування відбувається за допомогою варикапа.

Звуковий сигнал використовується для ЧМ-модуляції сигналу звукової піднесучої. Потім модульована звукова піднесуча складається з повним відеосигналом після відновлення в ньому постійної складової і отриманий таким чином сигнал використовується для амплітудної модуляції обраної несучої. Модульована несуча поступає в підсилювач петлі, який змішує цей сигнал з сигналом, що поступає на антенний вхід.

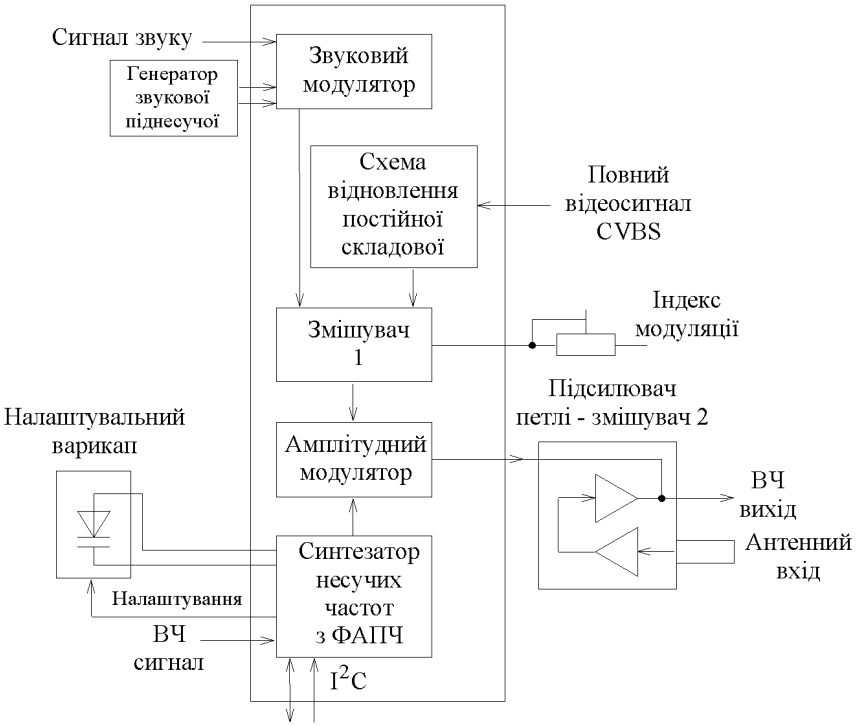


Рис. 8.36.

Коли цифровий декодер не працює в якості приставки, модулятор з синтезом відключається по команді від мікроконтролера управління, що надходить по цифровій шині I^2C. Підсилювач петлі продовжує працювати, тому сигнал від антенного входу через підсилювачі проходить на ВЧ вихід.

На закінчення розділу приводимо спрощену структурну схему телевізора (рис. 8.37), відео- і аудіо сигнали в якому обробляються в цифровому вигляді.

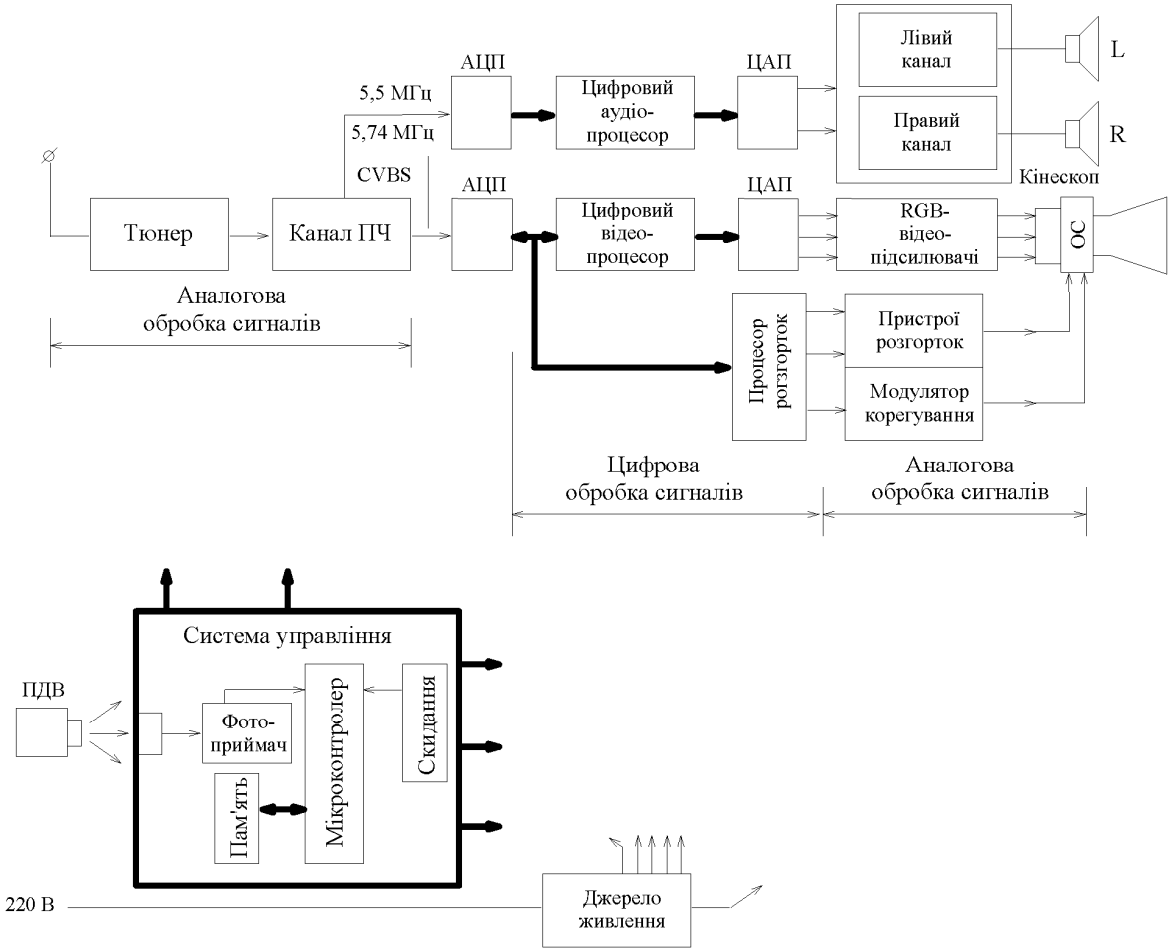


Рис. 8.37.

Питання до розділу 8

1. Формування телевізійного зображення.
2. Основні види спотворення телевізійного зображення.
3. Система NTSC.
4. Система PAL.
5. Система SECAM.
6. Електронно-променеві трубки.
7. Чорно-білі кінескопи. Будова та принцип роботи.
8. Кольорові кінескопи. Будова та принцип роботи.
9. Класифікація, основні параметри та норми телевізорів.
10. Засоби відображення інформації програмно-растрового типу.
11. Принципи побудови знакогенераторів в телевізійних ЗВІ.
12. Принцип роботи та будова CRT-моніторів.
13. Плазмові та FED-монітори.
14. Назвіть критерії вибору монітору.
15. Основні параметри моніторів. Енергозберігаючі функції моніторів.

9. ЕЛЕМЕНТИ ТА ПРИСТРОЇ ОПТОЕЛЕКТРОНІКИ

У цьому розділі розглядаються основні оптоелектронні елементи та пристрої: джерела і приймачі оптичного випромінювання, оптрони та оптоелектронні мікросхеми, індикаторні пристрої, волоконно-оптичні системи. Викладені фізичні основи роботи, конструкції і технології виготовлення, основні параметри і сфери застосування оптоелектронних приладів.

9.1. Основні характеристики та параметри фотоприймачів

Фотоприймач (приймач оптичного випромінювання) – це прилад, в якому під дією оптичного випромінювання відбуваються зміни, що дозволяють виявити і виміряти характеристики випромінювання. В основі роботи напівпровідникових фотоприймачів лежить використання явища внутрішнього фотоефекту, але можуть бути використані й інші ефекти [36].

Спектральна характеристика чутливості відображає реакцію приймача на вплив випромінювання з різною довжиною хвилі, і визначає спектральну сферу застосування приладу.

Енергетична (світлова) характеристика відображає залежність виходу приладу від інтенсивності збуджуючого потоку випромінювання (ампер-ватна, вольт-ватна, люкс-амперна). Енергетичною характеристикою називають також залежність інтегральної або спектральної чутливості приймача від інтенсивності опромінення.

Порогові характеристики – показують здатність фотоприймача реєструвати випромінювання малої інтенсивності. В основному вони визначаються рівнем власних шумів приладу – флуктуацій струму при відсутності опромінення або при немодульованому світловому потоці.

Вольт-амперна характеристика відображає залежність струму фотоприймача від прикладеної напруги.

Частотні характеристики описують залежність чутливості від частоти модуляції випромінювання або тривалості імпульсів, і характеризують інерційність приладу.

Температурні характеристики визначають залежність параметрів приладу від температури довкілля. Робочий температурний інтервал вказують в паспорті приладу, при цьому вказують значення основних параметрів в крайніх точках інтервалу.

До основних параметрів фотоприймачів відносять робочу і максимально припустиму напругу, потужність розсіювання, тепловий опір.

Чутливість фотоприймачів описується струмовою або вольтовою чутливістю. Струмова чутливість (А/Лм або А/Вт) визначає значення фотоструму, що створюється одиничним потоком випромінювання: $S_I = I_Ф/Ф$.

Вольтова чутливість характеризує значення фотонапруги, віднесене до одиниці потоку випромінювання (В/Лм або В/Вт):

$$S_V = V_Ф/Ф.$$

У загальному випадку залежності $I_Ф(Ф)$ і $V_Ф(Ф)$ є нелінійними, тобто чутливість може залежати від величини потоку випромінювання.

Короткохвильова (довгохвильова) межа спектральної чутливості визначається найменшою (найбільшою) довжиною хвилі монохроматичного випромінювання, при якій монохроматична чутливість приладу дорівнює 0,1 від її максимального значення.

Динамічний діапазон лінійності (у децибелах) характеризує область значень світлового потоку $Ф$ (від $Ф_{min}$ до $Ф_{max}$), в якій енергетична (світлова) характеристика є лінійною: $Δ=10·lg(Ф_{max}/Ф_{min})$. Інерційність фотоприймача характеризують постійні часу наростання $τ_н$ і спаду $τ_{сп}$ фотовідгуку для імпульсу випромінювання. Зростання і спад фотоструму описується експоненціальними залежностями:

$$J_Ф = J_{max}(1 – \exp(–t/τ_н)),\ J_Ф = J_{max}\exp(–t/τ_{сп}).$$

Сталі часу визначають граничні значення робочої частоти модуляції світлового потоку, при якому ще не помітно зменшення фотовідгуку. Зазвичай $τ_н ‹ τ_{сп}$, але ці величини мають однаковий порядок. При синусоїдальній модуляції світлового потоку швидкодію приладів характеризують граничною частотою, на якій фотовідгук зменшується до рівня 0,7 стаціонарного значення.

Струми у фотоприймачі за відсутності і наявності освітлення зазнають випадкових відхилень $ΔJ$ від середнього значення. Відхилення мають різні знаки, тому середнє значення відхилення $ΔJ_{сер}=0$. Міра випадкового відхилення (флуктуації) – середнє значення квадрату відхилення $ΔJ^2_{сер}$. Фотострум, що відповідає світловому сигналу, спостерігається на фоні цих відхилень (шумів), що визначають найменший світловий сигнал, який може бути зафіксований.

Шуми можуть мати різне походження. Тепловий шум (Джонсона-Найквіста) обумовлений хаотичністю теплового руху електронів, концентрація їх в різних ділянках напівпровідників змінюється з часом. Дробовий шум обумовлений відхиленням від середнього значення числа електронів, що проходять через прилад при постійній зовнішній напрузі. Причинами його виникнення можуть служити випадкові процеси генерації, рекомбінації, дифузії та ін. Існують й інші види шумів: фотонний, фоновий, технологічний та інші.

Оптична інформація в оптоелектронних пристроях практично зводиться до двох видів: дискретні сигнали (у часі, просторі і в спектральній області), і світлові образи.

9.1.1. Фоторезистори

Механізм поглинання світла, що призводить до появи вільних носіїв заряду в напівпровіднику, називають фотоактивним. Оскільки при цьому змінюється провідність, а отже, внутрішній опір напівпровідника, вказане явище було назване фоторезистивним ефектом. Основне застосування фоторезистивний ефект знаходить в світлочутливих напівпровідникових приладах – фоторезисторах [32].

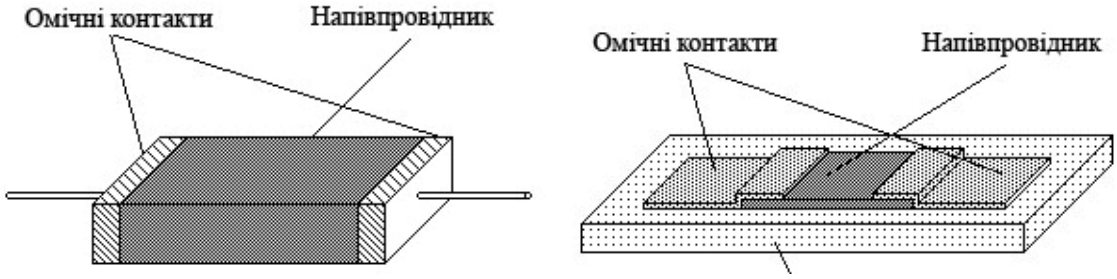
Фоторезистори застосовуються в різних системах, призначених для контролю і вимірювання швидкостей руху об'єктів, температури, управління різними механізмами, для визначення якісного складу твердих, рідких і газоподібних середовищ, включення і виключення різних пристроїв. При цьому у багатьох випадках фоторезистивний спосіб отримання й обробки інформації дає явну перевагу в порівнянні з іншими способами. Використання фоторезисторів показує їх високу надійність і широкі можливості. Вони мають високу чутливість, досить малу інерційність, незначні габарити, довговічні в роботі, забезпечують безконтактні виміри і контроль. При їх застосуванні досягається однобічність зв'язку між джерелом сигналу інформації – випромінювачем і приймачем.

Фоторезистор являє собою дискретний напівпровідниковий резистор, омічний опір якого визначається мірою освітленості. Під впливом світлового потоку електричний опір шару змінюється у декілька разів (у деяких типів фотоопорів він зменшується на два-три порядки). Залежно від використаного шару напівпровідникового матеріалу фотоопори поділяються на сернисто-свинцеві, сернисто-кадмієві, сернисто-вісмутові та полікристалічні селено–кадмієві. Фоторезистори мають високу чутливість, стабільність, економічні та надійні в експлуатації. В цілому ряді випадків вони з успіхом замінюють вакуумні й газонаповнені фотоелементи.

Напівпровідникові фоторезистори працюють в ланцюгах як постійного, так і змінного струму. Технічними умовами допускається також використання фоторезисторів в імпульсних режимах, при середній потужності розсіювання, яка не перевищує максимально припустиме значення. Фоторезистори можуть працювати при великих інтенсивностях світла, за умови не перевищення граничного значення фотоструму і потужності розсіювання.

Світлочутливий елемент в деяких типах фоторезисторів виконаний у вигляді круглої або прямокутної таблетки, спресованої з порошкоподібного сульфіду або селеніду кадмію, в інших він являє собою тонкий шар напівпровідника, нанесеного на скляну основу. В обох

випадках з напівпровідниковим матеріалом сполучені два металеві виводи. В залежності від призначення фоторезистори мають різне конструктивне оформлення. Інколи це просто пластина напівпровідника на скляній основі з омічними виводами, в інших випадках фоторезистор має пластиковий корпус з жорсткими виводами. Конструкція монокристалічного та плівкового фоторезисторів показана на рис. 9.1, а, б, відповідно.



а *б*

Рис. 9.1.

Основним елементом фоторезистора є в першому випадку монокристал, а в другому – тонка плівка напівпровідникового матеріалу. Якщо фоторезистор включений послідовно з джерелом напруги (рис. 9.2) і не освітлений, то в його ланцюзі наявний темновий струм:

$$I_т = E / (R_т + R_н),$$

де E – Е.Р.С. джерела живлення; $R_т$ – величина електричного опору фоторезистора в темряві, яка називається темновим опором; $R_н$ – опір навантаження.

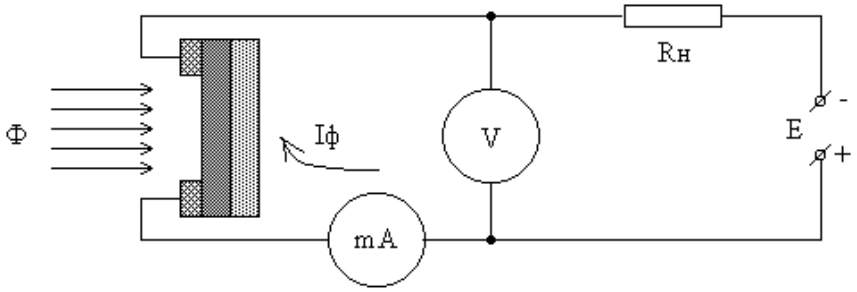


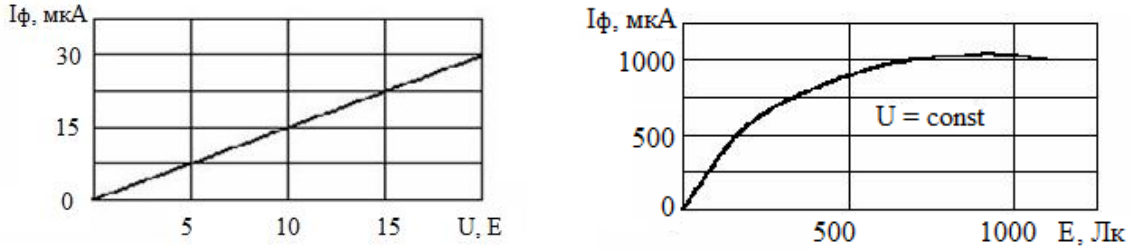
Рис. 9.2.

До основних характеристик фоторезисторів відносяться:

– вольтамперна, яка характеризує залежність фотоструму (при постійному світловому потоці $Ф$), або темнового струму від прикладеної напруги (рис. 9.3, а). Закон Ома порушується лише при високих напругах на фоторезисторі. Ця характеристика лінійна в досить широких межах. Для деяких типів фоторезисторів, при напругах менше робочої, спостерігається нелінійність.

– світлова (люксамперна), яка характеризує залежність фотоструму від падаючого світлового потоку постійного спектрального складу.

Напівпровідникові фоторезистори мають нелінійну люксамперну характеристику (рис. 9.3, б). Найбільша чутливість отримується при малих освітленостях. Це дозволяє використовувати фоторезистори для виміру дуже малих інтенсивностей випромінювання. При збільшенні освітленості світловий струм зростає приблизно пропорційно квадратному кореню з освітленості. Нахил люксамперної характеристики залежить від прикладеної до фоторезистора напруги.

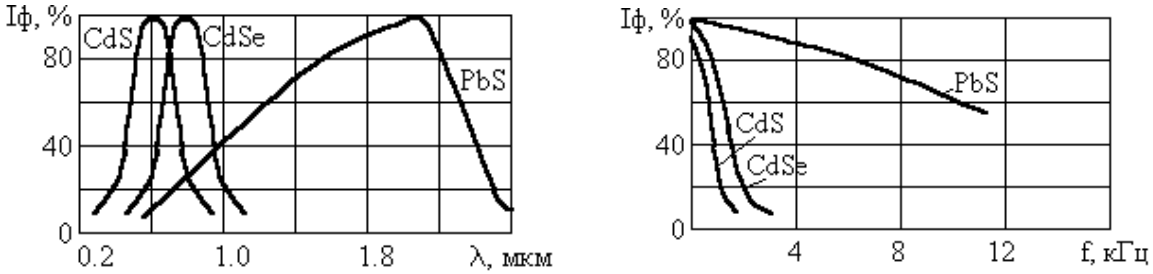


а *б*

Рис. 9.3.

– спектральна, яка характеризує чутливість фоторезистора при дії на нього потоку випромінювання постійної потужності визначеної довжини хвилі. Спектральна характеристика визначається матеріалом, який використовується для виготовлення світлочутливого елементу. Характеристики (рис. 9.4, а), фоторезистора з сірчано-кадмієвим світлочутливим елементом мають максимальну чутливість у видимій частині спектру. Фоторезистори, виконані на основі селеністого кадмію, найбільш чутливі до червоної та інфрачервоної частини спектру, а сірчано-свинцеві фоторезистори мають максимум чутливості в інфрачервоній області спектру

– частотна, яка характеризує чутливість фоторезистора при дії на нього світлового потоку, що змінюється з певною частотою. Наявність інерційності у фоторезисторів призводить до того, що величина їх фотоструму залежить від частоти модуляції падаючого на них світлового потоку – із збільшенням частоти світлового потоку фотострум зменшується (рис. 9.4, б). Інерційність обмежує можливості застосування фоторезисторів при роботі зі змінними світловими потоками високої частоти.



а *б*

Рис. 9.4.

Як видно з характеристики, величина сигналу, що знімається з фоторезистора, зменшується із збільшенням частоти модуляції світлового потоку.

До основних параметрів фоторезисторів відносяться:

– робоча напруга $U_р$ – постійна напруга, прикладена до фоторезистора, при якій забезпечуються номінальні параметри при тривалій його роботі в заданих експлуатаційних умовах. При роботі в імпульсному режимі в сірчано-кадмієвих і селеніcто-кадмієвих фоторезисторів максимально припустима напруга може в 2-3 рази перевищувати робочу;

– максимально припустима напруга фоторезистора U_{max} – максимальне значення постійної напруги, прикладеної до фоторезистора, при якій відхилення його параметрів від номінальних значень не перевищує вказаних меж при тривалій роботі в заданих експлуатаційних умовах;

– темновий опір $R_т$ – опір фоторезистора при відсутності падаючого на нього випромінювання в діапазоні його спектральної чутливості;

– світловий опір $R_с$ – опір фоторезистора, виміряний через деякий інтервал часу після початку дії випромінювання, що створює на ньому освітленість заданого значення;

– кратність зміни опору K_R – відношення темнового опору фоторезистора до опору при визначеному рівні освітленості (світловому опорі). Це один з найважливіших параметрів, що характеризує чутливість фоторезистора. Із збільшенням освітленості кратність зростає по лінійному закону, із зменшенням – знижується;

– припустима потужність розсіювання – потужність, при якій можлива тривала експлуатація фоторезистора при температурі +20 °С в оточуючому середовищі без небезпеки появи нзворотніх змін в світлочутливому шарі;

– загальний струм фоторезистора – струм, який складається із темнового струму і фотоструму;

– фотострум – струм, що протікає через фоторезистор при вказаній напрузі на ньому, обумовлений лише дією потоку випромінювання із заданим спектральним розподілом;

– питома чутливість – відношення фотоструму до добутку величини падаючого на фоторезистор світлового потоку, на прикладену до нього напругу, мкА/(Лм·В):

$$K_0 = I_ф / (Ф \cdot U),$$

де $I_ф$ – фотострум, рівний різниці струмів, що протікають по фоторезистору в темряві при визначеній (200 Лк) освітленності, мкА; $Ф$ –

падаючий світловий потік, Лм; U – напруга, прикладена до фоторезистора, В.

– інтегральна чутливість – добуток питомої чутливості на граничну робочу напругу:

$$S_{\text{инт}} = K_0 U_{\max}.$$

стала часу t_ϕ – час, протягом якого фотострум змінюється на 63%, тобто в e разів. Стала часу характеризує інерційність приладу і впливає на вигляд його частотної характеристики.

Завдяки високій чутливості, простоті та різноманітності конструкцій, фоторезистори мають широке застосування в радіоелектроніці. Основними сферами застосування є: системи фотоелектричної автоматики і телемеханіки, вимірювальні прилади, експонометричні прилади та фотодатчики, системи телеконтролю, а також апаратура промислового, медичного та оборонного призначення.

На рис. 9.5 представлена принципова схема пристрою автоматичного включення та виключення світла.

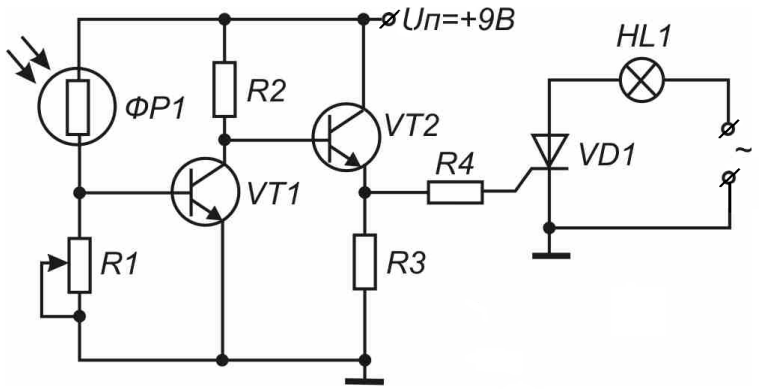


Рис. 9.5.

У випадку, коли інтенсивність природного світла достатньо велика, опір збудженого фоторезистора низький. Тому струм бази транзистора VT1 достатній, щоб утримувати його у відкритому стані. При відкритому VT1 транзистор VT2 замкнутий і, отже, замкнутий тиристор VD1 – лампа освітлення не горить. Коли природна освітленість почне знижуватися, опір фоторезистора ФР1 збільшується, а струм бази транзистора VT1 зменшується. При певному значенні освітленості транзистор VT1 закриється, а VT2 і VD1 відкриються і, отже, включиться лампа освітлення. Момент включення лампи може регулюватися змінним резистором R7.

На схемі рис. 9.6 фоторезистор ФР1 керує включенням реле Р1, яке замикає ланцюг виконавчого механізму. Отримуючи на вхідному вікні світловий імпульс, фоторезистор різко змінює власний опір, який знижується на три порядки в порівнянні з темновим опором. В цьому

випадку транзистори VT1 і VT2 закриваються, і спрацьовує реле P1 з нормально-розімкненими контактами.

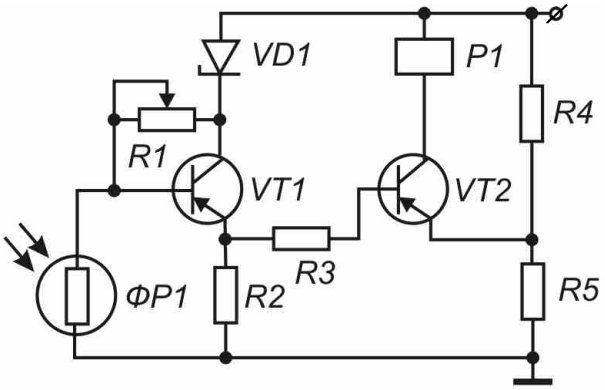


Рис. 9.6.

Основною сферою застосування фоторезисторів є автоматика, де вони в деяких випадках з успіхом замінюють вакуумні і газонаповнені фотоелементи. Володіючи підвищеною припустимою потужністю розсіювання в порівнянні з деякими типами фотоелементів, фоторезистори дозволяють створювати прості і надійні фотореле без підсилювачів струму. У вимірювальній техніці фоторезистори застосовуються для вимірювання високих температур, для регулювання температури в різних технологічних процесах, захисту персоналу від входу в небезпечні зони, контролю за запиленістю і задимленістю найрізноманітніших об'єктів, автоматичних вимикачах вуличного освітлення.

9.1.2. Фотодіоди

Фотодіод – приймач оптичного випромінювання (рис. 9.7), який перетворює світло, що потрапило на його фоточутливу область, в електричний заряд за рахунок процесів в *p*-*n*-переході.

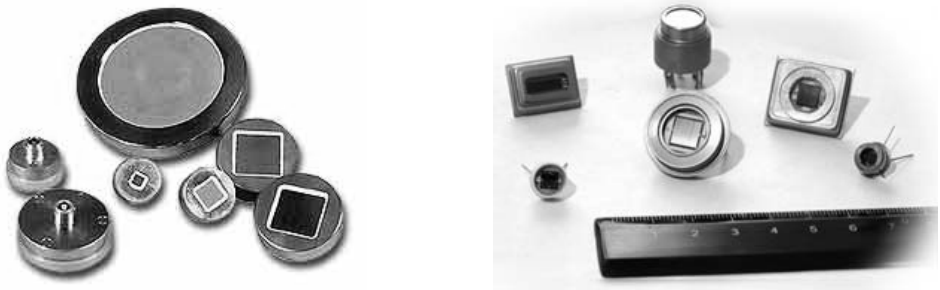


Рис. 9.7.

При дії квантів випромінювання в базі відбувається генерація вільних носіїв, які спрямовуються до кордону *p*-*n*-переходу. Ширина бази (*n*-область) робиться такою, щоб дірки не встигали рекомбінувати до переходу в *p*-область. Струм фотодіода визначається струмом неосновних

носіїв – дрейфовим струмом. Швидкодія фотодіода визначається швидкістю розділення носіїв полем *p-n*-переходу і ємністю *p-n*-переходу $C_{p\text{-}n}$. Фотодіод може працювати у двох режимах:

– фотогальванічний – без зовнішньої напруги;

– фотодіодний – із зовнішньою зворотною напругою.

Основні параметри фотодіодів:

– чутливість – відображає зміну електричного стану на виході фотодіоду при подачі на вхід одиничного оптичного сигналу. Кількісно чутливість вимірюється відношенням зміни електричної характеристики, що знімається на виході фотоприймача, до світлового потоку або потоку випромінювання, що його викликав. Струмова чутливість по світловому потоку визначається виразами:

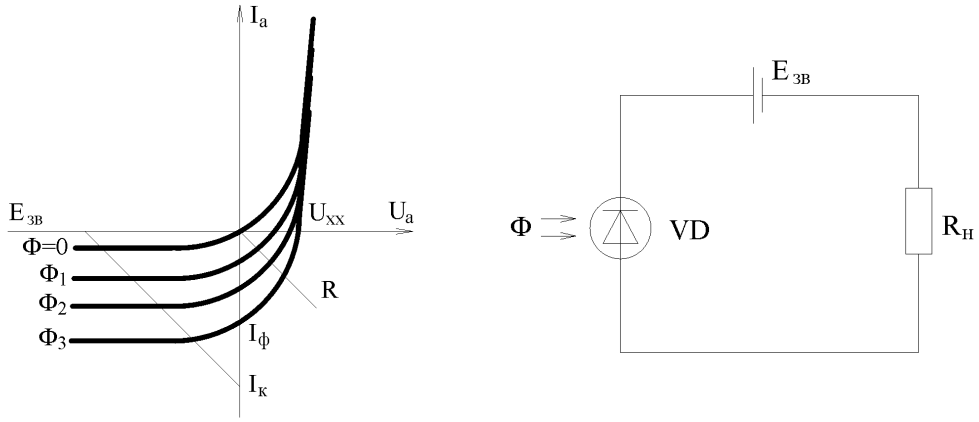
$$S_{i,\Phi_v} = \frac{I_\Phi}{\Phi_v} \quad , \quad S_{i,E_v} = \frac{I_\Phi}{E_v} \quad .$$

Вольтова чутливість по енергетичному потоку:

$$S_{u,\Phi_e} = \frac{U_\Phi}{\Phi_e} \quad , \quad S_{i,E_e} = \frac{U_\Phi}{E_e} \quad .$$

– шуми – окрім корисного сигналу на виході фотодіода з'являється хаотичний сигнал з випадковою амплітудою і спектром – шум фотодіода. Він не дозволяє реєструвати скільки завгодно малі корисні сигнали. Шум фотодіода складається з шумів напівпровідникового матеріалу і фотонного шуму;

– вольтамперна характеристика (ВАХ) – залежність вихідної напруги від вхідного струму ($U_a = f(I_a)$), представлена на рис. 9.8. Схема включення фотодіода представлена на рис. 9.8, б.



а *б*

Рис. 9.8.

У першому квадранті характеристика фотодіода повністю відповідає характеристиці випрямляючого діода. У третьому квадранті маємо фотодіодний режим (режим фотоперетворювача), який забезпечується при прикладенні зворотної напруги до фотодіода за схемою, яка приводиться на рис. 9.8, а. Величина фотоструму при такому режимі роботи майже не залежить від величини зворотної напруги і повністю задається інтенсивністю світлового потоку, тобто фотодіод при роботі в третьому квадранті може розглядатися як джерело струму, кероване світловим потоком. Робоча точка фотодіода при заданій величині $R_{\text{Н}}$ визначається відомими з електротехніки методами розрахунку електричних ланцюгів з нелінійними елементами. Четвертий квадрант ВАХ світлодіода відповідає фотогальванічному режиму, або режиму фотогенератора. Фотодіоди, які працюють в режимі фотогенератора, використовуються як перетворювачі світлового потоку в сонячних батареях.

– спектральні характеристики – залежність фотоструму від довжини хвилі падаючого світла на фотодіод. Короткохвильова границя чутливості залежить від товщини бази і від швидкості поверхневої рекомбінації. Положення максимуму в спектральній характеристиці фотодіода сильно залежить від міри зростання коефіцієнта поглинання;

– світлові характеристики – залежність фотоструму від освітленості, відповідає прямій пропорційності фотоструму від освітленості. Це обумовлено тим, що товщина бази фотодіода значно менше дифузійної довжини неосновних носіїв заряду. Тобто практично всі неосновні носії заряду, що виникли в базі, беруть участь в створенні фотоструму;

– стала часу – це час, протягом якого фотострум фотодіода змінюється після освітлення або після затемнення фотодіода в *e* разів (63%) по відношенню до сталого значення;

– темновий опір – опір фотодіода при відсутності освітлення;

– інерційність.

На рис. 9.9 представлена схема включення фотодіода.

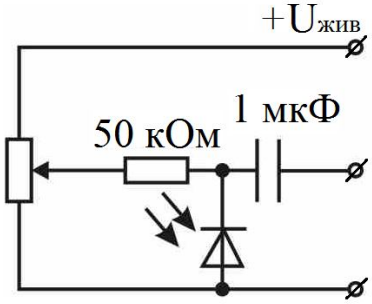


Рис. 9.9.

Змінюючи опір змінного резистора в приведеній схемі, можна протестувати і вибрати оптимальний режим роботи фотодіода. Змінюючи опір резистора від мінімального до максимального, можна підібрати

найкращий режим зсуву на фотодіоді. Замкнувши рухливий контакт на землю, схема працюватиме у фотогальванічному режимі.

Опір в 50 кОм не дасть пошкодити фотодіод, а за змінною складовою він виявляється включеним паралельно з навантаженням (менше 5 кОм), і корисний сигнал практично не послаблюється. Конденсатор позбавляє від постійної складової. Якщо приймається імпульсний сигнал, то від постійної складової, яка змінюється в залежності від фонового освітлення, краще позбавиться відразу.

Ще одна стандартна схема включення фотодіоду показана на рис. 9.10.

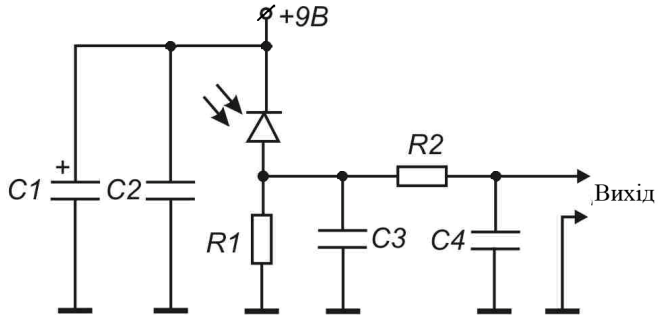


Рис. 9.10.

У приведеній схемі для зменшення впливу шумів і перешкод додані буферні конденсатори в ланцюги живлення, накопичувальний конденсатор С3 та інтегруючий ланцюг R2C4 на виході. С1 – електролітичний конденсатор великої ємності 100 мкФ, С2 – швидкий керамічний 0,1 мкФ, С3, С4 – керамічні по 100 пФ, R1 – 8 кОм, R2 – 5,6 кОм.

В табл. 9.1 представлені основні параметри деяких фотодіодів.

Табл. 9.1.

Тип	Розмір фоточутливого елемента, мм	Спектр $\Delta\lambda$, мкм	Робоча напруга, В	Темновий струм, мкА, не більше	Габарити
ФД-3К	1,13	0,5...1,1	15	0,5	11×11×1,7
ФД-8К	2×2	0,5...1,12	20	1,0	3,87×12,5
ФД-101КП	0,5	0,5...1,05	10	0,005	3,87×12,5
ФД-11К	2,5	0,5... 1,15	10	0,2	8,2×5,5
ФД-20-31К	1,4	0,47...1,17	20	0,1	7,2×5
ФД-21-КП	1,55	0,4...1,1	10	0,017	3,87×12,5
ФД-24К	10	0,47... 1,12	27	2,5	19,6×6,5
ФД-25К	1,9×1,9	0,4... 1,1	20	1,0	3,87×12,5

Фотодіоди з p-i-n-структурою. Для підвищення швидкодії роботи фотодіодів широко використовуються структури *p-i-n*-типу (рис. 9.11, а). Їх особливість полягає в тому, що між структурами напівпровідників *p*- і *n*-типів з високою концентрацією основних носіїв, розміщується напівпровідник з власною концентрацією. Його опір в 10^6 – 10^7 разів перевищує опір напівпровідників з домішками. Така особливість призводить до того, що прикладена зворотня напруга розподіляється по всій довжині *i*-структури, яка забезпечує можливість створення рівномірного і сильного електричного поля по всій її довжині. Наявність сильного поля E дає можливість забезпечення значного прискорення носіїв, підвищення їх швидкості $v = \mu E$, які з'являються в структурі внаслідок опромінення і, відповідно, зменшення часу пробігу $t_П$ через область:

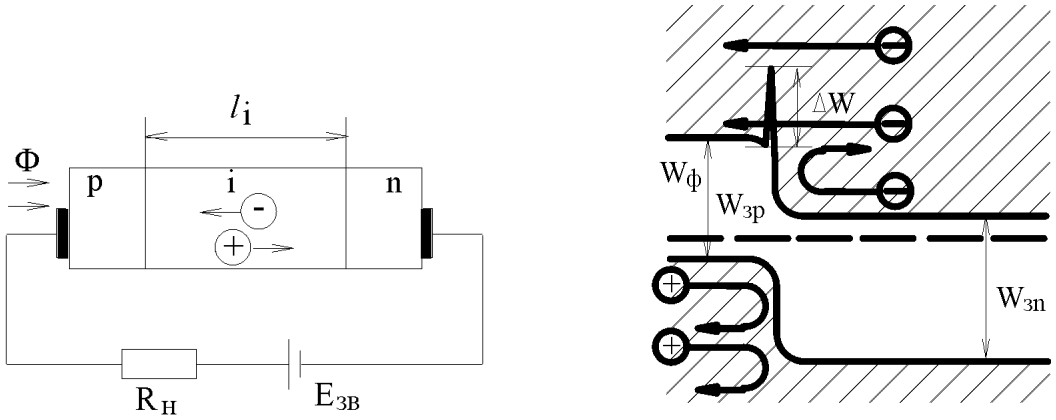
$$t_{П} = l_i / \mu E .$$

Значна величина l_i дає можливість істотно зменшити величину бар'єрній ємності і, відповідно, постійну часу $C_Б \cdot r_Б$, оскільки висока концентрація носіїв в області бази також дозволяє зменшити опір бази. Робочий частотний діапазон таких фотодіодів досягає меж до 10^9 Гц.

Фотодіоди з *p-i-n*-структурою мають такі переваги:

- поєднання високої чутливості і високої швидкодії;
- можливість забезпечення високої інтегральної чутливості в довгохвильовій області спектру при збільшенні ширини *i*-області;
- мала величина бар'єрної ємності;
- малі робочі напруги у фотодіодному режимі, що дає можливість забезпечити їх електричну сумісність з інтегральними схемами.

До недоліків *p-i-n*-структури слід віднести вимогу високої чистоти *i*-бази і погану технологічну сумісність з тонкими легованими пластинами інтегральних схем.



а *б*

Рис. 9.11.

Фотодіоди Шотки. Особливості структур напівпровідник-метал, які створюють контакт Шотки, були розглянуті вище. З точки зору використання у фотодіодах слід виділити такі їх особливості.

По-перше, у фотодіоді з бар'єром Шотки з'являється можливість поглинання квантів випромінювання з енергією, що менша за ширину забороненої зони, в металі контакту. При цьому, якщо енергія кванту випромінювання більше висоти потенційного бар'єру, збуджені електрони з металу можуть перейти в напівпровідник через потенційний бар'єр. В результаті довгохвильова границя спектральної характеристики фотодіоду Шотки зміщується в область більш високих частот.

По-друге, у фотодіоді Шотки із зростанням енергії квантів область поглинання випромінювання зміщується в область об'ємного заряду, де існує поле, яке розділяє фотоносії. У фотодіоді з *p*-*n*-переходом при малій глибині поглинання, фотострум практично дорівнює нулю. Спектральні характеристики фотодіодів Шотки значно ширші, ніж діодів з *p*-*n*-переходом.

Окрім вказаних, фотодіоди Шотки мають ще ряд позитивних властивостей:

- малий опір бази, яка забезпечує величину сталої часу $C_Б \cdot r_Б$ на рівні 10^{-12} с, тому інерційність цих приладів визначається лише часом переходу фотоносіїв через область об'ємного заряду ($10^{-10} – 10^{-11}$ с);
- поєднання високої швидкодії і високої чутливості;
- простота створення фоточутливих структур на різноманітних напівпровідниках і, як наслідок, можливість управління висотою потенційного бар'єру Шотки.

Фотодіоди з гетероструктурою. Гетероструктури можуть виготовлятися як на основі однотипних напівпровідників з різною шириною забороненої зони, так і на основі різнотипних. При створенні гетероструктур має місце перерозподіл носіїв заряду, що приводить до появи контактної різниці потенціалів і відповідного вирівнювання рівня Фермі. Особливістю енергетичних діаграм гетеропереходів є те, що на границі розділу структур створюється розрив енергії ΔW, наявність якого призводить до того, що потенційні бар'єри для дірок і електронів є різними.

На рис. 9.11, б, зображена енергетична діаграма гетероструктури з вузькою забороненою зоною W_{3p} напівпровідника *p*-типу (бази) і широкою забороненою зоною W_{3n} напівпровідника *n*-типу (емітера). Внаслідок цього в рівноважному стані має місце рух носіїв лише одного типу, а при прикладенні прямої напруги – відповідно, однобічна інжекція – рух електронів з широкозонного шару у вузькозонний.

Гетероструктура з широкозонним емітером і вузькозонною базою називається одинарною. Широке використання знаходять також подвійні гетероструктури, в яких розміщується допоміжний замикаючий широкозонний шар того ж типу провідності, як і база. У подвійній

гетероструктурі другий потенційний бар'єр перешкоджає виходу електронів з області бази, в якій накопичуються інжектовані електрони. Збиткова концентрація носіїв в активній зоні і однобічна інжекція різко підвищують її швидкодію.

Використання подвійної гетероструктури забезпечує локалізацію інжектованих носіїв заряду в базі при зменшенні її ширини буквально до декількох мікрон. Це і дозволяє підвищити швидкодію подвійних гетероструктур.

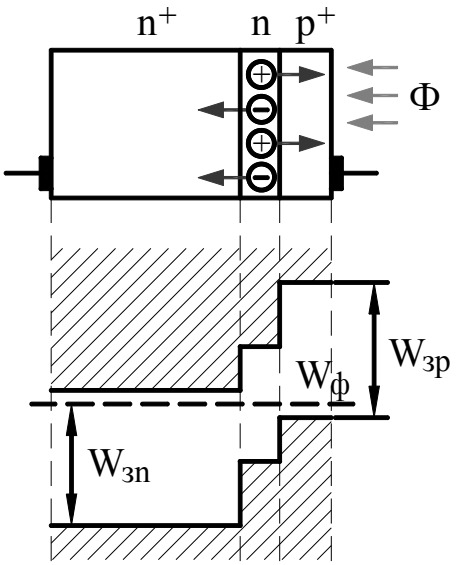


Рис. 9.12.

Принцип дії і зонна діаграма гетерофотодіоду зображені на рис. 9.12. Шар p^+ (GaAlAs) виконує функцію вікна, яке пропускає промені, що поглинаються в середній *n*-області.

Різниця в ширині заборонених зон по обидві сторони від гетеропереходу складає близько 0,4 еВ. Генеровані в *n*-області під дією оптичного випромінювання дірки переносяться в p^+-область. Товщина активної *n*-області вибирається такою, щоб забезпечити поглинання всього випромінювання.

Гетерофотодіоди мають високу чутливість і високу швидкодію при малих робочих напругах. Гетероструктури дозволяють також, підібравши відповідні пари напівпровідників, працювати практично в будь-якій частині оптичного діапазону хвиль. Така перевага обумовлена тим, що в гетероструктурі робоча довжина хвилі визначається різницею між шириною заборонених зон і не пов'язана із спектральною характеристикою глибини поглинання. Можлива величина фото ЕРС в гетерофотодіодах досягає 0,8-1,1 В (це в 2-3 рази вище, ніж у кремнієвих фотодіодів).

Лавинні фотодіоди. Використання лавинних фотодіодів є одним зі шляхів створення швидкодіючих фотоприймачів з високою чутливістю. При високому рівні електричного поля в активній зоні фотодіода енергія, яку отримують електрони і дірки в цьому полі, перевищує енергію

створення електронно-діркових пар. Тому, під дією квантів світла, починається лавинний процес розмноження носіїв. Посилення первинного фотоструму в лавинному фотодіоді визначається коефіцієнтом лавинного множення:

$$M = I_{\Phi}/I_{\Phi 0},$$

де I_{Φ} – струм на виході фотодіода з врахуванням перемножування; $I_{\Phi 0}$ – струм за відсутності перемножування. Коефіцієнт лавинного перемножування є коефіцієнтом посилення фотоструму.

Коефіцієнт перемножування залежить від напруги на переході:

$$M = 1/(1 - (U/U_{\text{ПР}})^m),$$

де $U_{\text{ПР}}$ – напруга лавинного пробою; U – напруга на p-n-переході; m = 1,5 ÷ 2 для кремнія p-типу і m = 3,4 ÷ 4 для кремнія n-типу. Тоді сімейство ВАХ лавинного фотодіода можуть бути математично записане у вигляді:

$$I_{\Phi} = I_{\Phi 0}/(1 - (U/U_{\text{ПР}})^m),$$

графічна ілюстрація яких приведена на рис. 9.13.

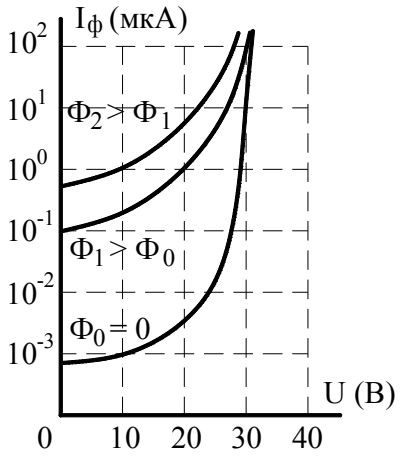


Рис. 9.13.

Лавинні фотодіоди знаходять використання для виявлення слабких оптичних сигналів. Більш широке їх використання пов'язане з труднощами. У передпробійному режимі коефіцієнт M сильно залежить від напруги (типові значення напруги пробою складає 30-100 В). Це призводить до значних втрат енергії у фотодіоді. По-друге, розвиток пробою відбувається не одночасно по всій площі p-n-переходу, а в окремих „мікроплазмах", що викликає додаткову нестабільність і підвищує шуми. Перераховані недоліки, у поєднанні із значним розкидом параметрів окремих типів, обмежує вживання лавинних фотодіодів.

Напівпровідникові пристрої мають інерційність, оскільки збільшення концентрації неосновних носіїв біля *p-n*-переходу при освітленні фотодіода та зменшенні цієї концентрації при її затемненні відбуваються не миттєво, а по експоненті з деякою постійною часу $\tau_Ф$. По такому самому закону змінюється і струм фотодіода. На рис. 9.14 представлена часова діаграма струму фотодіода при дії на нього прямокутного імпульсу світла [43].

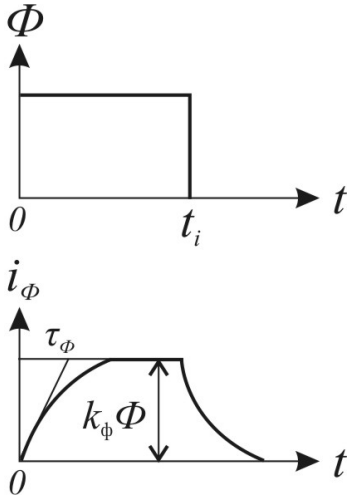


Рис. 9.14.

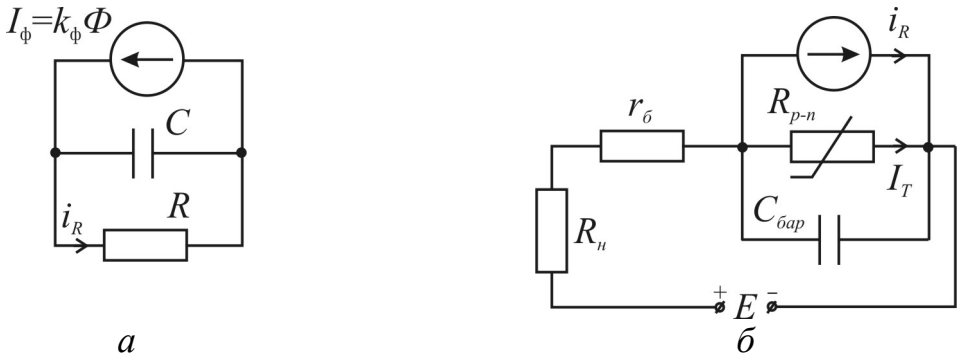
На рис. 9.15 приведені схеми заміщення фотодіода, в яких за допомогою RC-ланки з постійною часу $\tau_Ф$ враховані його інерційні властивості [43]. При використанні схеми спочатку потрібно знайти струм в опорі R другорядного кола (рис. 9.15, а):

$$i_R = k_Ф Ф(1 - e^{-t/\tau_Ф})$$

при появі, і

$$i'_R = k_Ф Ф(1 - e^{-(t-t_i)/\tau_Ф})$$

при зникненні світлового імпульса, а потім використовувати цей струм як струм генератора безпосередньо в еквівалентній схемі фотодіода (рис. 9.15, б).



а *б*

Рис. 9.15.

Еквівалентна схема фототранзистора принципово не відрізняється від схеми фотодіода (рис. 9.15). Тільки $I_Ф=\beta k_Ф Ф$, і постійна часу його другорядного кола (рис. 9.15, а) приблизно в β разів більша, ніж у фотодіода [43].

9.1.3. Фототранзистори

Фототранзистор – оптоелектронний напівпровідниковий прилад, варіант біполярного транзистора. Відрізняється від класичного варіанту тим, що область бази доступна для світлового опромінення, за рахунок чого з'являється можливість управляти підсиленням електричного струму за допомогою оптичного випромінювання.

Біполярний фототранзистор – напівпровідниковий прилад з двома *p*-*n*-переходами – призначений для перетворення світлового потоку в електричний струм. При освітленні фототранзистора в його базі генеруються електронно-діркові пари. Неосновні носії зарядів переходять в область колектора, і частково в область емітера. При цьому потенціали емітера і колектора відносно бази змінюються. Емітерний перехід зміщується в прямому напрямі, і навіть невелика зміну його потенціалу викликає велику зміна струму колектора, тобто фототранзистор є підсилювачем. Струм колектора освітленого фототранзистора виявляється достатньо великим – відношення світлового потоку до темнового велике (декілька сотень). Фототранзистори мають значно більшу, ніж фотодіоди, чутливість – порядка сотень міліампер на люмен. Біполярний фототранзистор подібний до звичайного біполярного транзистора, між виводами колектора і бази якого включений фотодіод. Таким чином, струм фотодіода є струмом фототранзистора і створює підсилений в *n* разів струм в ланцюзі колектора. Якщо на фототранзистор подається лише електричний сигнал, його параметри майже не відрізняються від параметрів звичайного транзистора.

Фототранзистор можна включати за схемами з вільним колектором, з вільною базою і з вільним емітером. На фототранзистор можна подавати оптичні і електричні сигнали. Без вхідного електричного сигналу, який зазвичай необхідний для зсуву, що компенсує перешкоди, фототранзистор працює як фотодіод з високою інтегральною чутливістю, невеликою граничною частотою і великим темновим струмом. Фототранзистори доцільно використовувати для реєстрації великих світлових сигналів; при реєстрації малих світлових сигналів слід подати позитивний зсув на базу. Застосовують два варіанти включення фототранзисторів: діодне – з використанням лише двох виводів (емітера і колектора) і транзисторне – з використанням трьох виводів, коли на вхід подають не лише світловий, але й електричний сигнали. Фототранзистори використовуються в якості фотоприймачів і транзисторних оптопарах. Фототранзистори мають низьку швидкодію у порівнянні з фотодіодами. Тому більш широке використання знаходять пари фотодіод – біполярний транзистор.

9.1.4. Фототиристори

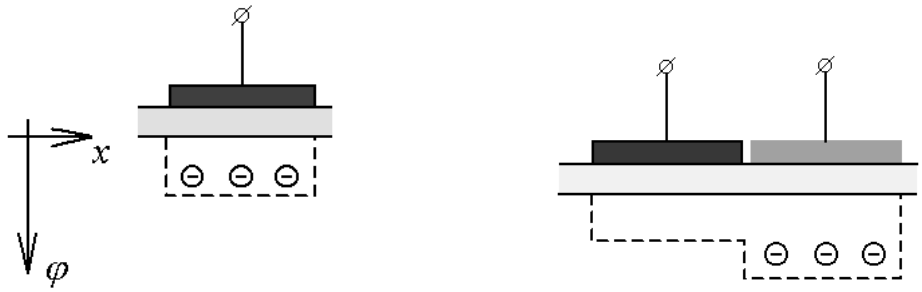
Фототиристор – оптоелектронний напівпровідниковий прилад, що має структуру, схожу зі структурою звичайного тиристора, але відрізняється від останнього тим, що вмикається не напругою, а світлом, яке падає на структуру тиристора. Цей прилад застосовується в керованих світлом випрямлячах і найбільш ефективний в управлінні значними струмами при високій напрузі. Швидкість відгуку на світло менше 1 мкс. Фототиристори зазвичай виготовляють з кремнію, і спектральна характеристика у них така ж, як і в інших кремнієвих фоточучливих елементів.

Різновидом фототиристора є оптотиристор, в якому джерело світла – світлодіод з арсеніду галію – інтегрований в одному світлонепроникному корпусі з кремнієвою тиристорною структурою.

9.1.5. Фоточутливі прилади із зарядовим зв'язком

Прилади із зарядовим зв'язком (ПЗЗ) відносяться до виробів функціональної електроніки, тобто їх не можна представити як сукупність транзисторів або конденсаторів. На рис. 9.16, а, показаний МОН-конденсатор який отримується з МОН-транзистора, якщо прибрати з нього стік і витік, тобто просто електрод, відокремлений від кремнію шаром діелектрика. Коли електричне поле, що створюється затвором, проникаючи в кремній крізь діелектрик, відштовхує рухливі дірки, виникає збіднена область – деякий об'єм кремнію, вільний від основних носіїв. Електрони, якщо вони яким-небудь чином (наприклад, в результаті фотогенерації) виявляться поблизу, притягнуться до затвору і будуть нкопичуватися на межі розподілу оксид-кремній безпосередньо під затвором, тобто будуть в потенційній ямі. При цьому електрони по мірі накопичення в ямі частково нейтралізують електричне поле, що створюється в напівпровіднику затвором, і врешті-решт можуть повністю його компенсувати, так що все електричне поле падатиме лише на діелектрику, і все повернеться у вихідний стан, за тим виключенням, що на межі розподілу утворюється тонкий шар електронів.

Нехай тепер поряд із затвором розташований ще один, і на нього теж поданий позитивний потенціал (рис. 9.16, б). Якщо затвори розташовані досить близько, їх потенційні ями об'єднуються, і електрони, що знаходяться в одній потенційній ямі, переміщаються в сусідню, якщо її потенціал вищий. Якщо є ланцюг затворів, то можна, подаючи на них відповідну керуючу напругу, передавати локалізований зарядовий пакет уздовж такої структури.



а *б*

Рис. 9.16.

Чудова властивість ПЗЗ – властивість самосканування – полягає в тому, що для управління ланцюгом затворів будь-якої довжини досить всього трьох тактових шин. Дійсно, для передачі зарядових пакетів необхідно і достатньо трьох електродів: одного передаючого, одного приймаючого і одного ізолюючого, який розділяє пари тих, що приймають і передають один від одного, причому однойменні електроди таких трійок можуть бути з'єднані один з одним в єдину тактову шину, що вимагає лише одного зовнішнього виводу (рис. 9.17).

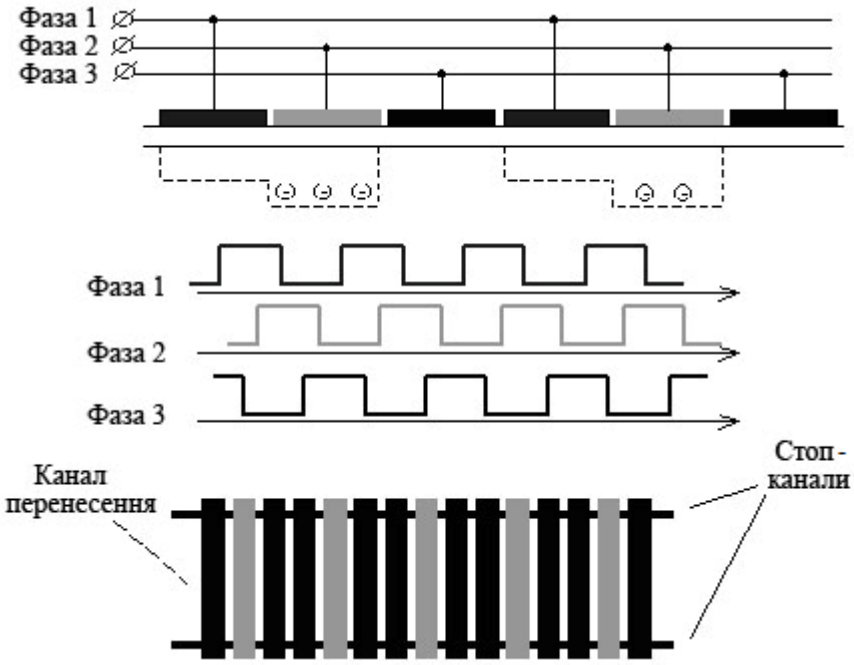


Рис. 9.17.

Для локалізації зарядових пакетів в поперечному напрямі формуються так звані стоп-канали – вузькі смужки з підвищеною концентрацією основної легуючої домішки, що йдуть уздовж каналу переносу. На повну передачу заряду з однієї ями в іншу потрібний час, так що при високій тактовій частоті (для ТВ стандарту вона складає в регістрі зчитування 7-13 МГц залежно від числа елементів по горизонталі) цього

часу може і не вистачити. Величина, що показує, яка частина зарядового пакету передалася в наступний елемент ПЗЗ, називається ефективністю переносу *e*. Часто користуються і пов'язаною з нею величиною неефективності $h = 1-e$.

Ступінь досконалості кристалічної решітки в сучасних матеріалах досить висока, і нині ефективність переносу в ПЗЗ з прихованим каналом досягає в кращих приладах величин 99,9999% (або $h = 10^{-6}$ на перенос, тобто після тисячі переносів спотворення від неефективності складають 0,1%.

На рис. 9.18 представлений вигляд стандартного телевізійного сигналу чорно-білого телебачення.

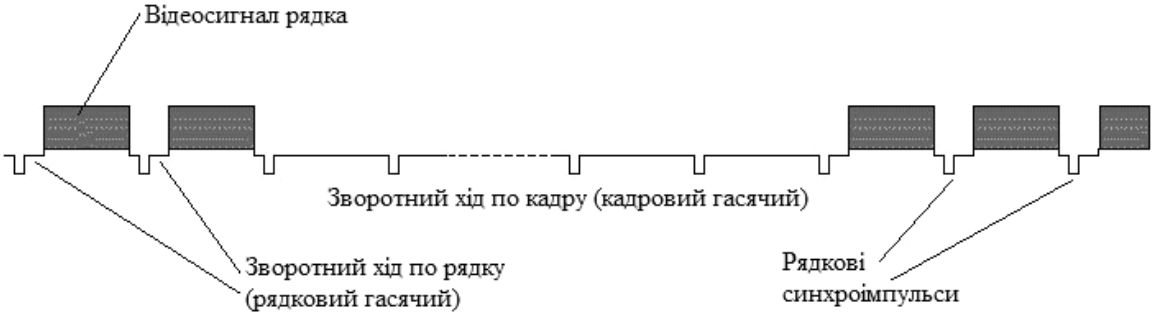


Рис. 9.18.

Він містить відеосигнали окремих рядків, розділених інтервалом зворотнього ходу по рядку (рядковий інтервал, що гасить), який необхідний для того, щоб електронний промінь як в кінескопі, так і в передаючій камері, встиг повернутися на початок наступного рядка. Під час цього інтервалу подається і рядковий синхроімпульс (він формується не самим датчиком зображення, а замішується з сигналом електронними схемами камери). Рівень синхроімпульсів прийнятий за 0, рівень чорного у відеосигналі складає 0,33 В, рівень, що гасить, дорівнює 0,3 В (30 мВ різниці утворюють захисний інтервал), максимальний рівень відеосигналу (рівень білого) дорівнює 1,0 В. Коли передані сигнали всіх рядків одного поля, починається формування кадрового інтервалу, що гасить. Рядкові синхроімпульси в цей час продовжують формуватися, щоб не збивати схеми рядкової розгортки кінескопу (у реальності їх частота на короткий час, що дорівнює рівний 2,5 тривалості рядка, подвоюється, а полярність інвертується, щоб позначити кадровий синхроімпульс), а відеосигнал не формується. Потім, після закінчення кадрового імпульсу що гасить, починається прямий хід по кадру для наступного поля. По прийнятому, наприклад, в Європі стандарту період рядкової розгортки складає 64 мкс, тривалість прямого ходу по рядку – 52 мкс, тривалість зворотнього ходу по рядку – 12 мкс, а тривалість кадрового імпульсу що гасить – 25 рядків. При цьому в кожному полі є 312,5 рядків, з яких 287,5, – активні, які мають відеосигнал (пів рядка виникає через те, що повне число рядків в кадрі для черезрядкової розгортки непарне – 625).

Найпростіший варіант ПЗЗ-матріці зображений на рис. 9.19. У ній можна виділити два вертикальні регістри зсуву ПЗЗ, які створюють секцію накопичення і секцію зберігання з рівним числом рядків (кожен рядок секції створений однією трійкою електродів), горизонтальний регістр зсуву і вихідний пристрій.

Протягом часу прямого ходу по кадру секція накопичення стоїть, тобто на неї подається незмінна напруга, що формує потенційні ями лише під одним електродом кожної трійки, скажімо, під електродом першої фази (VS1), причому потенційні ями утворюються у всіх елементах всіх рядків секції. По горизонталі окремі комірки накопичення відокремлені стоп-каналами. Зображення, що проектується на секцію накопичення, викликає фотогенерацію – утворення електронно-діркових пар. При цьому фотогенеровані електрони залишаються в потенційній ямі, дірки ж, відповідно, підуть в подложку або уздовж поверхні в стоп-канали. Таким чином, під дією світла у комірках накопичується зарядовий рельєф, тобто в кожній комірці збирається заряд, пропорційний її освітленості і часу накопичення.

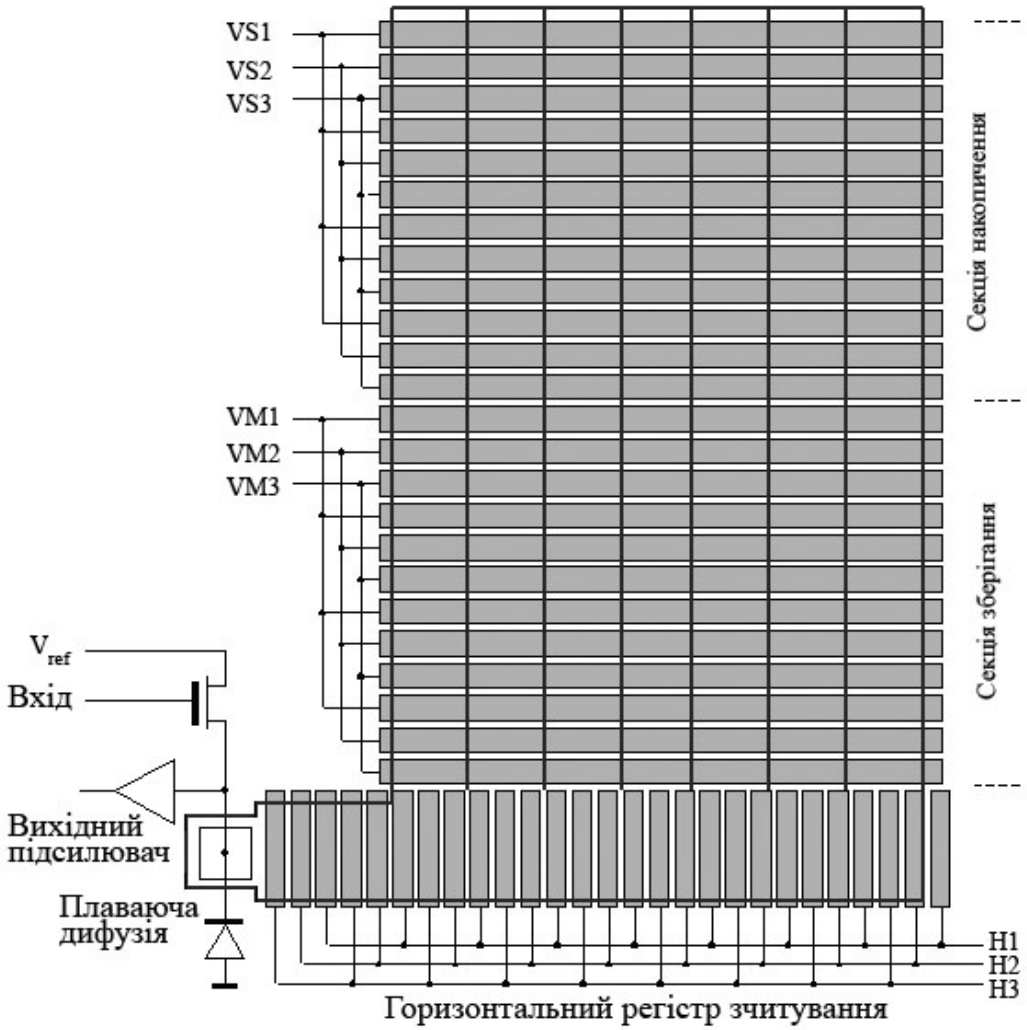


Рис. 9.19.

Після закінчення прямого ходу по кадру на обидві секції подаються тактові імпульси, що викликають синхронне перенесення заряду, при цьому важливо, що обидві секції утворюють безперервний регістр зсуву. Після кількості тактів, що дорівнює кількості рядків в кожній секції, весь накопичений зарядовий рельєф повністю переміститься в секцію пам'яті, яка закрита від світла, а секція накопичення буде очищена від заряду. Це перенесення секції в секцію відбувається досить швидко (фактично воно займає малу частину часу зворотнього ходу по кадру). Тепер, під час наступного циклу накопичення (це наступне поле кадрової розгортки), секція накопичення накопичує наступний кадр зображення, а з секції пам'яті заряди порядково, під час зворотного ходу по рядку, передаються в горизонтальний регістр (кожен елемент регістра має зарядовий зв'язок з відповідним стовпцем секції пам'яті, і за один раз передається один рядок), і потім виводяться у вихідний пристрій регістру за час прямого ходу по рядку, формуючи відеосигнал.

9.2. Принципи роботи та різновиди оптичних зчитувачів (сканерів)

Пристрій, що виконує перетворення документа або зображення в цифрову форму, називається сканером. Сканери подібні до пристроїв копіювання, лише замість друку копії сканер передає оцифровані дані в комп'ютер. Найбільш популярні останнім часом настільні (планшетні) сканери (рис. 9.20). Як і в інших типах сканерів, в них використовується відбитий від оригінала промінь. Розрізняють дві технології сканування – CCD (Charge-Coupled Device, прилад із зарядовим зв'язком – ПЗЗ) і CIS (Contact Image Sensor).

CCD. У цих моделях промінь проходить більш довгий шлях після і навіть до сканування, оскільки для сканування кольорових зображень він проходить через світлофільтри для розкладання на червону, зелену і блакитну складові. Промінь світла падає на оригінал, відбивається від нього і через систему дзеркал потрапляє на світлочутливі діоди, де перетворюється в електричний сигнал. Цей сигнал поступає на аналого-цифровий перетворювач, де конвертується в сигнал, що є пікселями оригіналу (чорні, білі, відтінки сірого або кольорові). Ця цифрова інформація передається в комп'ютер для подальшої обробки.

CIS. У цих сканерах використовується інший тип приймального елементу, що називається CIS (Contact Image Sensor). Цей елемент складається з лінійки датчиків, що безпосередньо сприймають світловий потік від оригіналу, причому лінійка має ширину, рівну ширині робочої області сканера, а оптична система – дзеркала, призма, об'єктив – повністю відсутні.

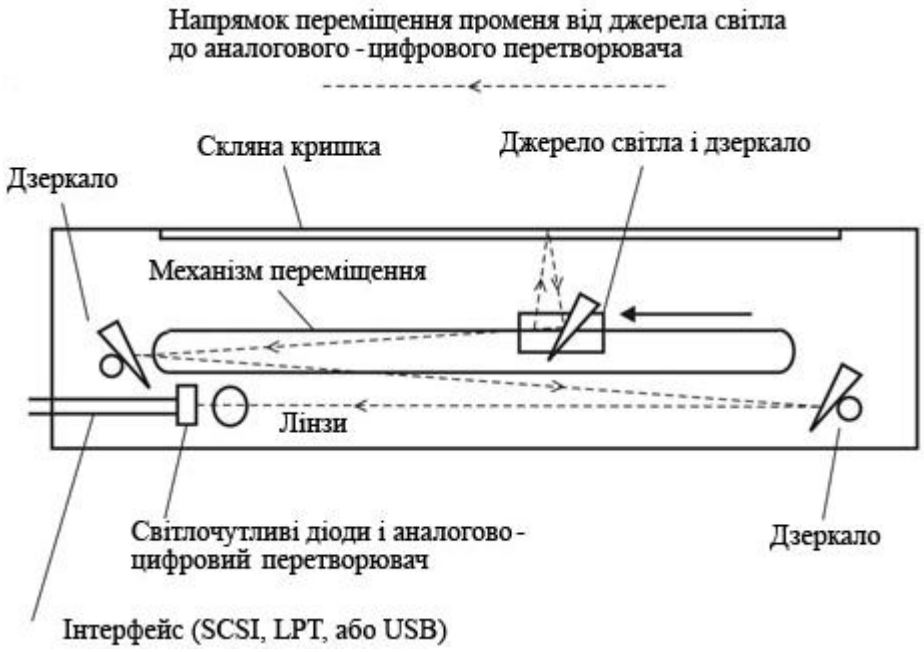


Рис. 9.20.

В табл. 9.2 приводится порівняння сканерів.

Табл. 9.2.

Charge-Coupled Device(CCD)	Contact Image Sensor (CIS)
Краща глибина різкості Глибина різкості CCD сканерів в 10 разів більша (±3 мм), ніж у CIS сканерів (±0,3мм). Це означає, що з CCD-сканером 3-мірні об'єкти або навіть книги та журнали будуть відскановані з хорошою різкістю, але при скануванні CIS сканером зображення часто буде розмитим і нерізким.	**Менші розміри і вага** Відсутність оптичної системи та зеркал дозволяє CIS-сканерам мати менші товщину та вагу, ніж їхні конкуренти з CCD-елементом.
Краща чутливість до відтінків CCD-сканери розрізняють рівні відтінків ±20%, тоді як CIS сканери визначають відмінності у відтінках тільки ±40%. Для користувача це означає, що передача деталей відтінків буде краща у CCD-сканерів.	**Зменшення витрат на виробництво** CIS-елементи замінюють цілий набір компонентів сканера, зменшуючи вартість виробництва
Триваліший термін служби сканера CCD сканери забезпечують стабільно високу якість сканування протягом більше 10 тис. годин.	

Дозволяюча здатність В даний час існують професійні CCD-сканери з розрішаючою здатністю 3000 точок на дюйм. В технології CIS дозволяюча здатність в даний час обмежена 1200 dpi (dot per inch – точок на дюйм).	
Добре розвинена технологія Протягом багатьох років були продані мільйони сканерів і факсів з CCD елементами. Хоча CIS елементи для факсів існують вже багато років, лише близько половини виробників факсів перейшли на них, незважаючи на низьку ціну.	

Переваги настільних сканерів:

– Можливість сканувати практично будь-який оригінал. Настільні сканери, як і копіювальні апарати, можуть сканувати оригінали різного розміру – від мініатюр до документів різних форматів, а також книг. При установці додаткового модуля з'являється можливість сканування прозорих плівок, негативів і слайдів. Більшість цих модулів призначена для сканування слайдів шириною 35 мм.

– Висока дозволяюча здатність. У настільних сканерах завжди використовуються два типи дозволяючої здатності – оптична й інтерпольована. Оптична дозвлляюча здатність описує можливості апаратної (оптичної) частини сканера. Для збільшення чіткості деталей оригіналу застосовуються спеціальні програмні алгоритми, які забезпечує драйвер сканера. Ця друга дозволяюча здатність називається інтерпольованою. Зазвичай воно збільшує максимальний дозвіл сканера до 4-х разів. Наприклад, якщо оптична дозволяюча здатність сканера 600 dpi, то максимальна інтерпольована – 2400 dpi. Оскільки ця інтерпольована дозволяюча здатність забезпечується програмними методами, при його використанні якість сканованого оригіналу може бути незадовільною. Але практично всі моделі сканерів забезпечують прийнятну якість при інтерпольованій дозволяючій здатності. Не можна застосовувати інтерполяцію при скануванні слайдів 35 мм.

Недоліки настільних сканерів:

– Великі розміри. Настільний сканер формату А4 має розміри як мінімум 210×297 мм і займає значну частину робочого простору.

– Обмеження на прозорі оригінали. Практично всі настільні сканери середнього і вищого рівня комплектуються модулем для сканування прозорих плівок або слайдів. Проте прийнятна якість досягається лише при скануванні оригіналів великих розмірів.

9.3. Сонячні батареї

Сонячні елементи (СЕ) виготовляються з матеріалів, які безпосередньо перетворюють сонячне світло в електричну енергію. В даний час більша частина СЕ виготовляється з кремнію. Зовнішній вигляд сонячної батареї представлений на рис. 9.21, а її будова на рис. 9.22.

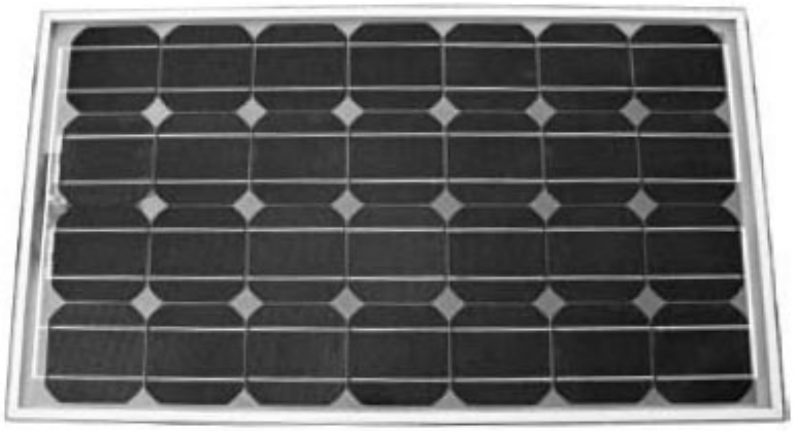


Рис. 9.21.

СЕ може бути наступних типів: монокристалічний, полікристалічний і аморфний. Відмінність між цими формами в тому, як організовані атоми кремнію в кристалі. Різні СЕ мають різний ККД перетворення енергії світла. Моно- і полікристалічні елементи мають майже однаковий ККД, який вище, ніж в СЕ, виготовлених з аморфного кремнію.

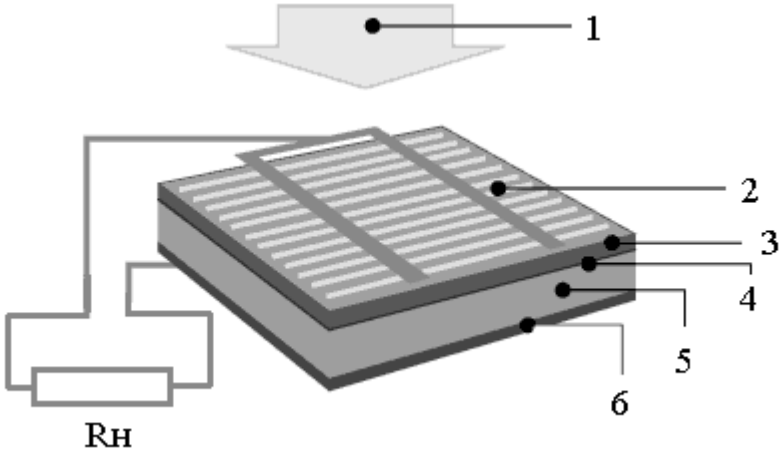


Рис. 9.22.

На рисунку позначені: 1 – світлові промені, 2 – передні контакти, 3 – негативний шар, 4 – відбиваючий шар, 5 – позитивний шар, 6 – задні контакти.

Перш за все, в СЕ є задній контакт і 2 шари кремнію різної провідності. Зверху є сітка з металевих контактів і просвітлююче покриття антивідблиску, яке дає СЕ характерний синій відтінок.

СЕ виробляє струм коли освітлюється світлом. В залежності від інтенсивності світла, сонячний елемент виробляє більший або менший струм. Для порівняння СЕ і модулів необхідно знати номінальну потужність елементу або модуля $W_{\text{р}}$.

Кристалічний кремнієвий СЕ з розмірами 10×10 см має пікову потужність приблизно $1{,}5W_{\text{р}}$. Більшість панелей з площею 1 квадратний метр мають номінальну потужність близько 100 Вт. Основні технічні характеристики сонячних елементів представлені в табл. 9.3.

Табл. 9.3.

Тип модуля	Ном. потужність, Вт	ККД, %	Тип	Напуга холостого ходу, U_{oc}, В	Струм короткого замикання, I_{sc}, А	Напруга в точці U_{mp}, В	Струм в точці I_{mp}, А	Розмір, мм	Повна площа, м^2
PSM4-120	120	11,3 до 13,2	цілий 72шт.	43,2	3,3	34	3,1	1600×800×38	1,28
PSM4-135	135	13,2 до 14,6	цілий 72шт.	43,2	3,8	34	3,7	1600×800×38	1,28
PSM4-150	150	≥14.6	цілий 72шт.	43,2	4,4	34	4,2	1600×800×38	1,28
PSM5-60	60	11,3 до 13,2	цілий 36шт.	21,6	3,3	17,5	3,1	1220×550×38	0,67
PSM5-67	67	13,2 до 14,6	цілий 36шт.	21,6	3,8	17,5	3,7	1220×550×38	0,67

9.4. Основні параметри оптронів

Оптронами називають такі оптоелектронні прилади, в яких є джерело і приймач випромінювання (світловипромінювач і фотоприймач) з тим або іншим виглядом оптичного і електричного зв'язку між ними, конструктивно зв'язані один з одним [31, 32].

Принцип дії оптронів будь-якого виду заснований на наступному. У випромінювачі енергія електричного сигналу перетворюється в світлову, у фотоприймачі навпаки, світловий сигнал викликає електричний відгук.

По мірі складності структурної схеми серед виробів оптронної техніки виділяють дві групи приладів. Оптопара (говорять також “елементарний оптрон”) представляє собою оптоелектронний напівпровідниковий прилад, що складається з випромінюючого і фотоприймального елементів, між якими є оптичний зв'язок, що забезпечує електричну ізоляцію між входом і виходом (рис. 9.23). Оптоелектронна інтегральна мікросхема представляє собою мікросхему, що складається з однієї або декількох оптопар і електрично сполучених з ними одного або декількох узгоджуючих, або підсилювальних пристроїв.

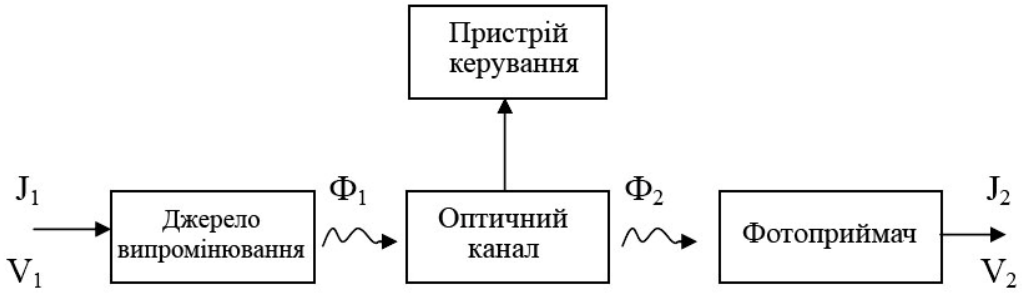


Рис. 9.23.

Таким чином, в електронному ланцюзі такий прилад виконує функцію елементу зв'язку, в якому в той же час здійснена електрична (гальванічна) розв'язка входу і виходу.

Переваги цих приладів базуються на загальному оптоелектронному принципі використання електрично нейтральних фотонів для перенесення інформації. Основні з них наступні:

– можливість забезпечення ідеальної електричної (гальванічної) розв'язки між входом і виходом; для оптронів не існує яких-небудь принципових фізичних або конструктивних обмежень по досягненню скільки завгодно високих напруг і опорів розв'язки, і скільки завгодно малої прохідної ємності;

– можливість реалізації безконтактного оптичного управління електронними об'єктами і обумовлені цим різноманітність і гнучкість конструкторських рішень управляючих ланцюгів;

– однонапрямленість поширення інформації по оптичному каналу, відсутність зворотньої реакції приймача на випромінювач;

– широка частотна смуга пропускання оптрона, відсутність обмеження з боку низьких частот (що властиве імпульсним трансформаторам); можливість передачі по оптронному ланцюгу як імпульсного сигналу, так і постійної складової;

– можливість управління вихідним сигналом оптрона шляхом впливу (у тому числі і неелектричного) на матеріал оптичного каналу, і витікаюча звідси можливість створення різноманітних датчиків, а також різноманітних приладів для передачі інформації;

– можливість створення функціональних мікроелектронних пристроїв з фотоприймачами, характеристики яких при освітленні змінюються за складним заданим законом;

– несприйнятливість оптичних каналів зв'язку до впливу електромагнітних полів, що в разі "довгих" оптронів (з протяжним волоконно-оптичним світлопровідником між випромінювачем і приймачем) обумовлює їх захищеність від перешкод втрати інформації, а також виключає взаємні наведення;

– фізична і конструктивно-технологічна сумісність з іншими напівпровідниковими і мікроелектронними приладами.

Оптронам властиві й певні недоліки:

– значна споживана потужність, обумовлена необхідністю подвійного перетворення енергії і невисокий ККД цих переходів;

– підвищена чутливість параметрів і характеристик до впливу підвищеної температури і проникаючої радіації;

– помітна часова деградація (погіршення) параметрів;

– відносно високий рівень власних шумів, обумовлений, як і два попередні недоліки, особливостями фізики світлодіодів;

– складність реалізації зворотних зв'язків, що викликано електричною роз'єднаністю вхідного і вихідного ланцюгів;

– конструктивно-технологічна недосконаліть, пов'язана з використанням гібридної непланарної технології (з необхідністю об'єднання в одному приладі декількох окремих кристалів із різних напівпровідників, що розташовуються в різних площинах).

На рис. 9.24 представлені конструкції оптопар.

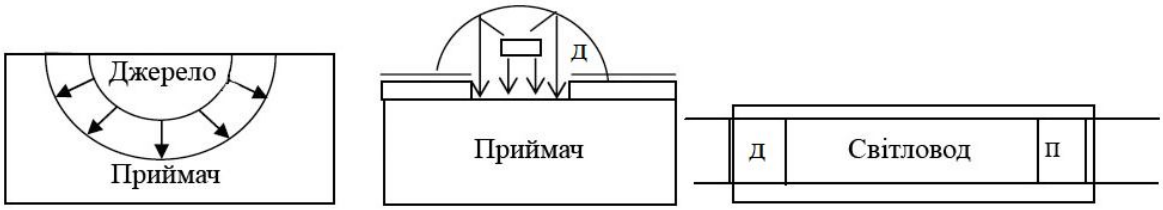


Рис. 9.24.

В якості елементів гальванічної розв'язки оптрони застосовуються: для зв'язку блоків апаратури, між якими є значна різниця потенціалів; для захисту вхідних ланцюгів вимірювальних пристроїв від перешкод і наведень, і т.д.

Інша важлива область застосування оптронів – оптичне безконтактне управління високовольтними ланцюгами. Запуск потужних тиристорів, симісторів, управління електромеханічними релейними пристроями.

Елементарний оптрон представляє собою чотириполюсний прилад, тому має 3 основних характеристики – вхідну, передавальну і вихідну. Вхідна характеристика є ВАХ випромінювача. Вихідна – відповідна характеристика фотоприймача (при заданому струмі на вході оптрона). Передавальна характеристика – залежність струму на виході J_2 від струму на вході J_1 (у загальному випадку нелінійна). Статичний коефіцієнт передачі по струму $K_J = J_2/J_1$. Якщо темновий струм $J_т$ значний, тоді $K_J = (J_2-J_T)/J_1$. Для більшості оптронів K_J є паспортним параметром (від 0,5% для діодних до 100% для транзисторних). Сумарну швидкодію оптрона часто характеризують часом перемикання $t_п=t_1+t_2$, t_1 и t_2 – час наростання і спаду сигналу на виході оптрона. Час $t_п$ неоднаковий у різних типів оптронів, залежить від режимів роботи і лежить в межах від 10^{-9} до 10^{-1} с. Швидкодія також може характеризуватися граничною частотою, $f_{гр}$ – 5 кГц ÷ 10 МГц.

Параметри ізоляції: максимально припустима напруга між входом і виходом $U_{із}$, статична або пікова (при роботі із змінними сигналами). Високий опір ізоляції – $R_{із} \sim 10^{12}$ Ом – виключає зворотний зв'язок по струму. Проте, наявність прохідної ємності $C_{пр}$ (між входом і виходом оптрону) обумовлює зв'язок по змінному струму, тобто стрибок напруги ΔU_2 на виході за час Δt приводить до появи ємнісного струму $J \sim C_{пр} \cdot \Delta U_2/\Delta t$. Значення прохідної ємності зазвичай складає ~ 1 пФ.

Універсальність оптронів як елементів гальванічної розв'язки і безконтактного управління, різноманітність і унікальність багатьох інших функцій є причиною того, що сферами використання цих приладів стали обчислювальна техніка, автоматика, апаратура зв'язку і радіотехнічна апаратура, автоматизовані системи управління, вимірювальна техніка, системи контролю і регулювання, медична електроніка, пристрої візуального відображення інформації.

9.4.1. Резисторні оптрони

Резисторні оптопари принципово відрізняються від всіх інших видів оптопар фізичними і конструктивно-технологічними особливостями, а також складом і значеннями параметрів. В основі принципу дії фоторезистора лежить ефект фотопровідності, тобто зміни опору напівпровідника при його освітленні. У резисторних оптронах в якості приймачів оптичного випромінювання використовуються фоторезистори на основі плівок CdS і CdSe. Особливістю фоторезисторів є порівняно велика інерційність ($t_п$=10^{-1}с), тому джерелами випромінювання в цих оптронах можуть бути лампи розжарювання. Залежність світлового опору від вхідного струму, залежність вихідної ВАХ від температури і залежність коефіцієнта передачі по струму від частоти показані на рис. 9.25.

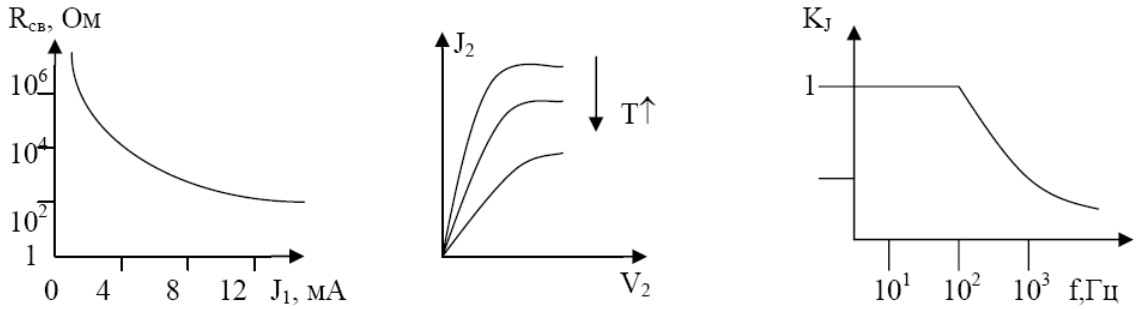


Рис. 9.25.

Випромінювач оптопари представляє собою електролюмінесцентний конденсатор або світлодіод, приймач – фоторезистор на основі селенистого кадмію.

9.4.2. Діодні оптопари

Зазвичай діодні оптопари виготовляються на основі кремнієвих *p-i-n*-фотодіодів та арсенідгалієвих випромінюючих діодів. Значення статичного коефіцієнта передачі по струму діодного оптрона визначається перетворенням вхідного струму J_1 у випромінювання Φ_1, передачею світлового потоку Φ_1 до приймача Φ_2,і перетворенням випромінювання Φ_2 у вихідний струм J_2.

Квантовий вихід випромінюючого діода $\eta_{сд}$ є сталою величиною в широкому діапазоні вхідних струмів, а квантовий вихід фотодіода $\eta_{фд}$ також стала величина в широкому діапазоні вихідних струмів. Вираз для коефіцієнта передачі по струму має вигляд:

$$K_J=\eta_{фд}\cdot\eta_{сд}\cdot k.$$

Тут коефіцієнт $k=\Phi_2/\Phi_1$ характеризує ефективність передачі випромінювання від джерела до приймача. Таким чином, значення коефіцієнта передачі по струму визначається величинами квантових виходів джерела і приймача. Для приймача $\eta_{фд}\approx100\%$, для світлодіода $\eta_{сд}\leq10\%$, тому величина коефіцієнта передачі по струму K_J декілька відсотків. Мала величина K_J діодних оптопар обумовлює відносно низькі значення вихідного струму, $J_2 \leq$ 1-2 мА. Залежність K_J від температури визначається температурною залежністю характеристик випромінювача.

Швидкодія діодних оптопар визначається швидкодією світлодіода і фотодіода. Але із-за малих вихідних струмів оптрони вмикають на велике навантаження, тому істотним стає час перезарядки $t_п=R_н\cdot C_2$, де $R_н$ – опір навантаження, C_2 – вихідна ємність оптрону. Зазвичай $R_н$ = 2-20 кОм, C_2 = 50 нФ, і час перезарядки $t_п$ може досягати 0,1-10 мкс. Залежності коефіцієнта передачі по струму від вхідного струму, вихідної напруги і температури показані на рис. 9.26.

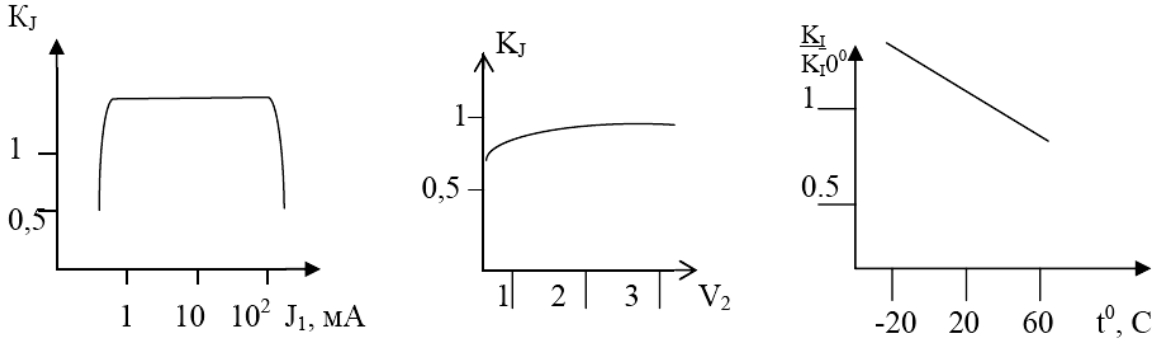


Рис. 9.26.

Оптрони можуть працювати у вентильному режимі, коли фотодіод служить джерелом живлення, тоді K_J = 3-4%, а ККД має значення 1%. Оптичний канал може бути виконаний у вигляді світловоду завдовжки до

30-100 мм, що забезпечує високий опір ізоляції і малу прохідну ємність ($U_{із} \approx$ 20-50 кВ, $C_{пр} \approx$ 0,01 нФ).

9.4.3. Транзисторні оптрони

Транзисторні оптопари рядом своїх властивостей вигідно відрізняються від інших видів оптронів. Це перш за все схемотехнічна гнучкість, яка проявляється в тому, що колекторним струмом можна управляти як по ланцюгу світлодіода (оптично), так і по ланцюгу бази (електрично), а також в тому, що вихідний ланцюг може працювати як в лінійному, так і в ключовому режимі. Механізм внутрішнього підсилення забезпечує отримання великих значень коефіцієнта передачі по струму K_i, так що подальші каскади підсилення не завжди необхідні. При цьому інерційність оптопари не дуже велика і для багатьох випадків цілком припустима. На рис. 9.27 представлені деякі транзисторні оптопари.

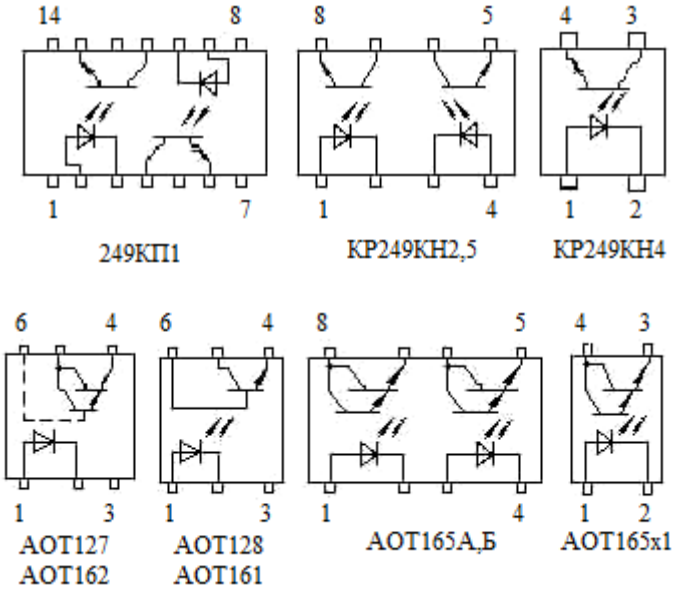


Рис. 9.27.

Вихідні струми фототранзисторів значно вищі ніж у фотодіодів, що робить їх придатними для комутації широкого кола електричних ланцюгів.

У табл. 9.4 приведені характеристики деяких транзисторних оптопар.

Табл. 9.4.

Найменування	$I_{вх}$ мА	K_i хв., %	$U_{ком}$ В	$I_{вих}$, мА	$t_{зд}$. макс., мкс	$U_{вих}$ ост. хв	$U_{із}$ В	Корпус
АОТ127А АОТ127Б АОТ127В	5	1400	30 30 15	70 15 15	10/100	1,5	1500	DIP-6
АОТ128А АОТ128Б АОТ128В АОТ128Г АОТ128Д	10	100 25	50 30 30 15 50	2,5 10 5 5 10	4	0,4	1500	DIP-6

АОТ161А АОТ161Б АОТ161А1 АОТ161Б1	10	100 25 100 25	50	10 2,5 10 2,5	5	0,3	6000 3000	DIP-6
АОТ162А АОТ162Б АОТ162В АОТ162А 1 АОТ162Б1 АОТ162В 1	5 1 5 1	1400	60 30 70 60 30 70	70 15 15 70 15 15	8/100	1,5	6000 3000	DIP-6
249КП1 249КП1А 249КП1С К249КП1	10 10 3,5 10	50 50 25 50	30	2 0,875 2	4	0,4	100	DIP-14

На рис. 9.28 представлена схема з використанням оптотранзистора, яка виконує функції індикатора.

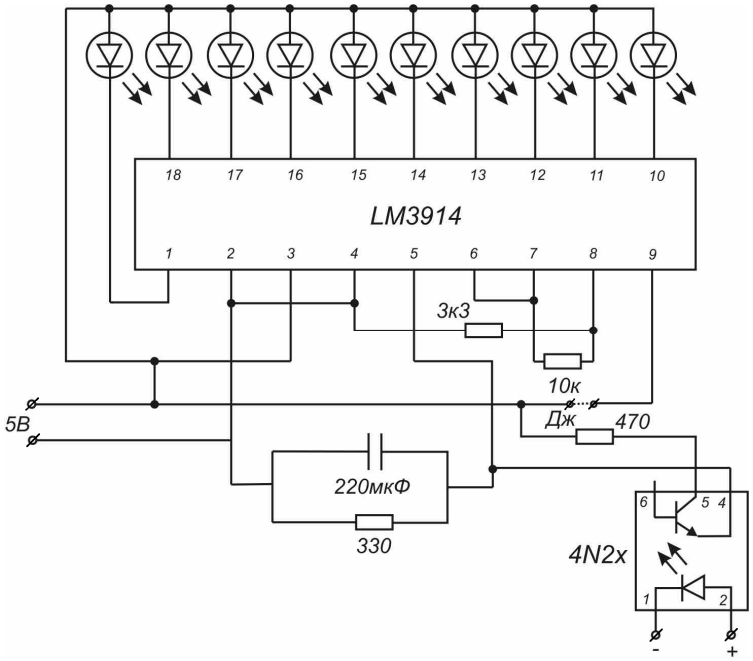


Рис. 9.28.

Перевага такої схеми в тому, що сигнал не використовується схемою безпосередньо, а використовується зміна опору емітерно-колекторного переходу оптотранзистора в оптопарі. Джампер (jumper) на схемі визначає режим роботи світлодіодної шкали. Якщо він замкнений, то світлодіоди спалахують один за одним справа на ліво, якщо розімкнений, то горітиме завжди лише один світлодіод. Мікросхема LM3914 керує світлодіодами і представляє собою драйвер шкального індикатора (led bargraph driver chip). Приведена схема може бути використана як своєрідний індикатор завантаження вінчестера. До виводів світлодіода оптопари (ніжки 1 і 2) підключаються виводи на материнській платі комп'ютера HDD Led. Якщо переплутати полярність на цьому світлодіоді, нічого не станеться, просто світлодіоди не будуть горіти.

9.4.4. Тиристорні оптрони

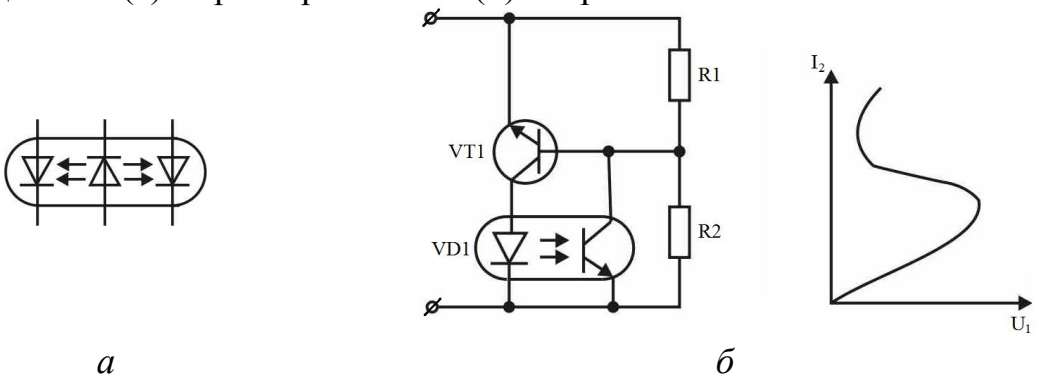
Тиристорні оптрони використовуються в якості ключів для комутації високовольтних ланцюгів радіоелектронного (РЕ) і електротехнічного (ЕТ) призначення (РЕ – U_2 = 50-600 В, J_2 = 0,1-10 А, ЕТ – U_2 = 100-1300 В, J_2 = 5-350 А). Тиристорні оптрони дозволяють здійснювати управління значними потужностями в навантаженні, а за входом сумісні з інтегральними мікросхемами.

Існує велика кількість груп оптронів, що розрізняються за значенням вихідних напруг, струмів та швидкодії. Оскільки тиристорні оптрони працюють у ключовому режимі, коефіцієнт передачі по струму K_J не використовується як параметр. Оптрони характеризуються номінальним значенням вхідного струму J_1, при якому відкривається фототиристор, і максимально припустимим вхідним струмом перешкоди (максимальний J_1, при якому фоториристор ще не включився). Номінальне значення струму $J_{1ном}$ = 20-200мА, максимально припустиме значення перешкоди складає близько 1мА.

9.4.5. Різновиди оптронів (диференціальні, регенеративні)

Диференціальні оптрони – в них один випромінювач впливає на два ідентичні фотоприймачі. Подібність вхідних характеристик двох каналів дозволяє використовувати їх для передачі декількох аналогових сигналів: перший канал для безпосередньої передачі, другий канал для ланцюга зворотного зв'язку – корегування температурних, деградаційних та інших змін потужності випромінювання.

Регенеративні оптрони відрізняються наявністю зворотнього зв'язку між випромінюючими і фотоприймальними елементами. Регенеративні оптрони використовуються як бістабільні елементи, заданому значенню напруги U_1 відповідає два значення вихідного струму J_2, тобто вони мають S-подібну ВАХ. Дані оптрони використовуються як перемикачі, підсилювачі, генератори. На рис. 9.29 представлені зображення диференційного (а) та регнеративного (б) оптронів.



а *б*

Рис. 9.29.

Широко застосовуються також оптрони з відкритим оптичним каналом, в яких до зазору між випромінювачем і приймачем є доступ

ззовні. Вони використовуються в безконтактній дистанційній техніці як індикатори положення об'єктів, для контролю стану їх поверхні та ін. Виділяють два типи оптронів з відкритим оптичним каналом:

1. оптопереривачі, які реагують на потрапляння в оптичний канал непрозорих предметів і переривають (або змінюють) світловий потік, який падає на приймач;

2. відбивні оптрони, які реєструють світловий потік, відбитий від досліджуваної поверхні.

Наявність повітряного зазору в оптичному каналі призводить до того, що коефіцієнт передачі по струму K_J малий, його значення залежить від властивостей поверхні та відстані до неї (декілька міліметрів). Для цих оптронів необхідно усунути вплив стороннього освітлення, і забезпечити точну просторову орієнтацію випромінювача та фотоприймача. Задача полегшується при використанні волоконних світлопроводів.

9.5. Пристрої на світловипромінюючих діодах та оптронах

Перспективні напрями розвитку і застосування оптронної техніки в значній мірі визначилися. Оптрони і оптронні мікросхеми ефективно застосовуються для передачі інформації між пристроями, що не мають замкнутих електричних зв'язків. Традиційно сильними залишаються позиції оптоелектронних приладів в техніці отримання і відображення інформації. Самостійне значення в цьому напрямі мають оптронні датчики, призначені для контролю процесів і об'єктів, досить різних за природою і призначенню. Помітно прогресує функціональна оптронна мікросхемотехніка, орієнтована на виконання різноманітних операцій, пов'язаних з перетворенням, накопиченням і зберіганням інформації. Ефективною і корисною виявляється заміна громіздких, недовговічних і нетехнологічних (з позицій мікроелектроніки) електромеханічних виробів (трансформаторів, потенціометрів, реле) оптоелектронними приладами і пристроями. Досить специфічним, але у багатьох випадках виправданим і корисним, є використання оптронних елементів в енергетичних цілях.

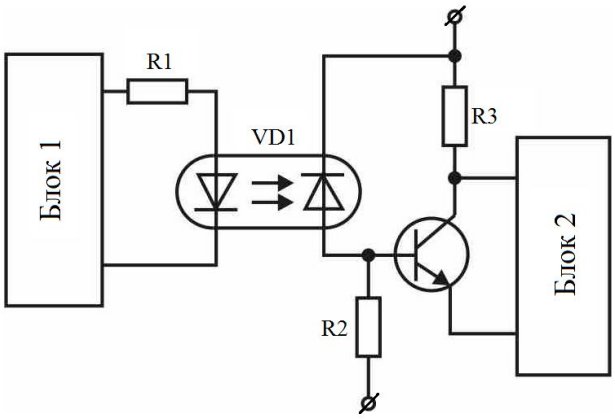


Рис. 9.30.

При передачі інформації оптрони використовуються як елементи зв'язку, і, як правило, не несуть самостійного функціонального навантаження. Їх застосування дозволяє здійснити досить ефективну гальванічну розв'язку пристроїв управління і навантаження (рис. 9.30), діючих в різних електричних умовах і режимах. З введенням оптронів різко підвищується перешкодостійкість каналів зв'язку; практично усуваються "паразитні" взаємодії по ланцюгах землі та живлення.

Інтерес представляє також раціональне і надійне узгодження цифрових інтегральних пристроїв з різнорідною елементною базою (ТТЛ, $І^2Л$, КМОН і т.п).

Схема узгодження елементу транзисторно-транзисторної логіки (ТТЛ) з інтегральним пристроєм на МДН-транзисторах побудована на транзисторному оптроні (рис. 9.31).

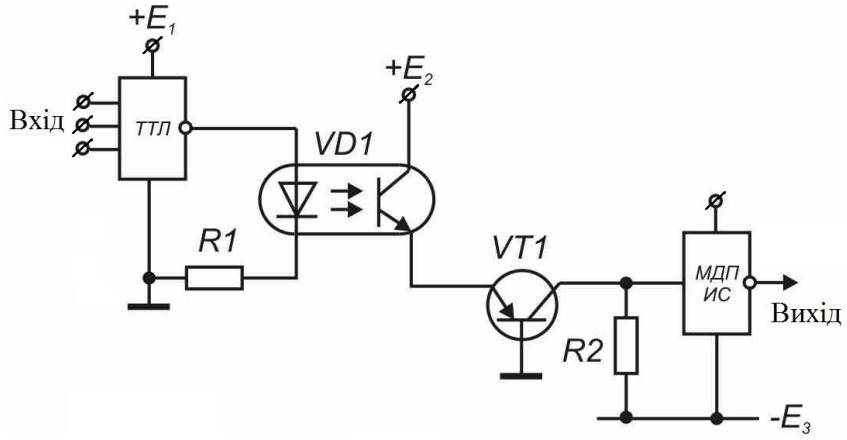


Рис. 9.31.

У приведеному варіанті: $E_1 = E_2 = 5$ В, $E_3 = 15$ В, $R_1 = 820$ Ом, $R_2 = 24$ кОм – світлодіод оптрона збуджується струмом (5 мА), достатнім для насичення транзистора і впевненого управління пристроєм на МДН-транзисторах.

Активно використовуються оптичні зв'язки в телефонних пристроях і системах. За допомогою оптронів технічно нескладними засобами вдається підключати до телефонних ліній мікроелектронні пристрої, призначені для виклику, індикації, контролю та інших цілей.

Введення оптичних зв'язків в електронну вимірювальну апаратуру, окрім корисної у багатьох відношеннях гальванічної розв'язки досліджуваного об'єкту від вимірювального приладу, дозволяє також різко зменшити вплив перешкод, діючих по ланцюгах заземлення і живлення.

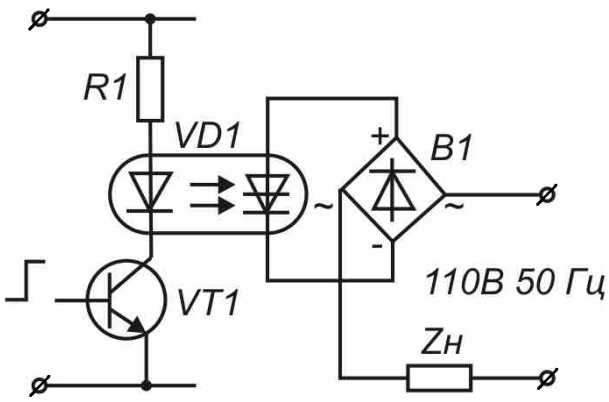


Рис. 9.32.

Значний інтерес представляють можливості та досвід використання оптоелектронних приладів і пристроїв в біомедичній апаратурі. Оптрони дозволяють надійно ізолювати хворого від дії високих напруг, що є, наприклад, в електрокардіографічних приладах.

Безконтактне управління потужними, високовольтними ланцюгами по оптичних каналах досить зручне і безпечне в складних технічних режимах, характерних для багатьох пристроїв і комплексів промислової електроніки. У цій області сильні позиції займають тиристорні оптрони (рис. 9.32).

Оптрони й оптронні мікросхеми займають міцні позиції в безконтактній дистанційній техніці оперативного отримання і точного відображення інформації про характеристики і властивості досить різних (за природою і призначенням) процесів та об'єктів. Унікальними можливостями в цьому плані володіють оптрони з відкритими оптичними каналами. Серед них оптоелектронні переривники, що реагують на перетин оптичного каналу непрозорими об'єктами (рис. 9.33), і відбивні оптрони, в яких дія світловипромінювачів на фотоприймачі цілком пов'язана з відбиттям випромінюваного потоку від зовнішніх об'єктів.

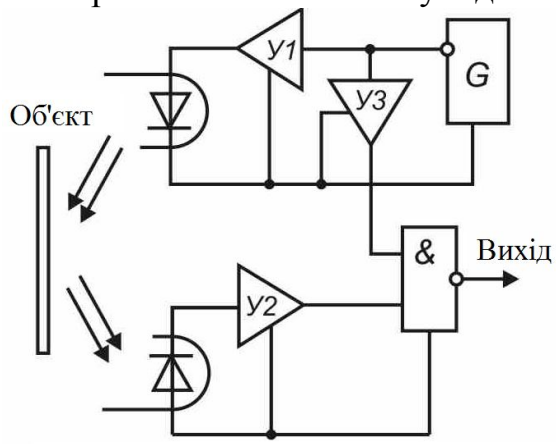


Рис. 9.33.

Сфера застосування оптронів з відкритими оптичними каналами обширна і різноманітна. Вже в 60-і роки оптрони подібного типу ефективно використовувалися для реєстрації предметів і об'єктів. При такій реєстрації, характерній в першу чергу для пристроїв автоматичного контролю і підрахунку об'єктів, а також для виявлення та індикації різного роду дефектів і відмов, важливо чітко визначити місцезнаходження об'єкту або відобразити факт його існування. Функції реєстрації оптрони виконують надійно і оперативно.

Потужність випромінювання, що генерується світлодіодом, і рівень фотоструму, що виникає в лінійних ланцюгах з фотоприймачами, прямо пропорційні струму електричної провідності випромінювача. Таким чином, по оптичних (безконтактних, дистанційних) каналах можна отримати цілком визначену інформацію про процеси в електричних ланцюгах, гальванічно пов'язаних з випромінювачем. Особливо ефективним виявляється використання світловипромінювачів оптронів в якості датчиків електричних змін у високовольтних ланцюгах. Чітка інформація про подібні зміни важлива для оперативного захисту джерел і споживачів енергії від електричних перевантажень.

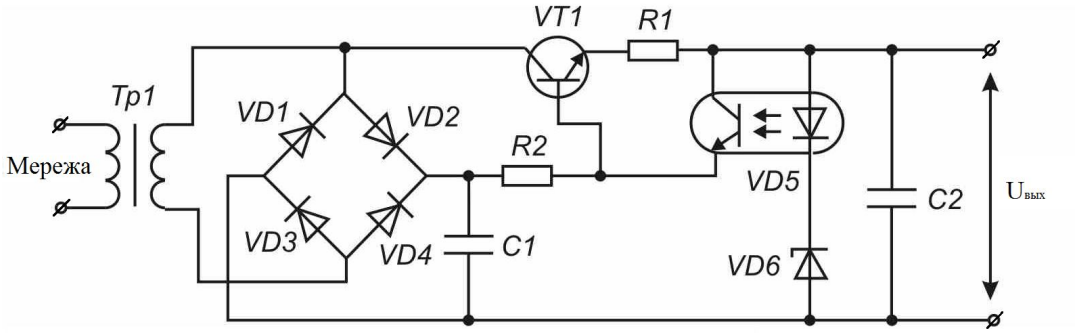


Рис. 9.34.

Оптрони успішно діють у високовольтних стабілізаторах напруги, де вони створюють оптичні канали від'ємних зворотних зв'язків. Даний стабілізатор (рис. 9.34) відноситься до пристрою послідовного типу, причому регулюючим елементом є біполярний транзистор, а кремнієвий стабілітрон діє як джерело опорної (еталонної) напруги. Порівнюючим елементом служить світлодіод.

Якщо вихідна напруга в схемі рис. 9.34 зростає, то збільшується і струм провідності світлодіода. Фототранзистор оптрону впливає на транзистор, зменшуюи можливу нестабільність вихідної напруги.

У комплексі технічних рішень, орієнтованих на підвищення ефективності і якості пристроїв автоматики, радіотехніки, електрозв'язку, промислової і побутової електроніки, доцільною і корисною мірою є заміна електромеханічних виробів (трансформаторів, реле, потенціометрів, реостатів, кнопкових перемикачів) компактнішими, довговічнішими, швидкодіючими аналогами.

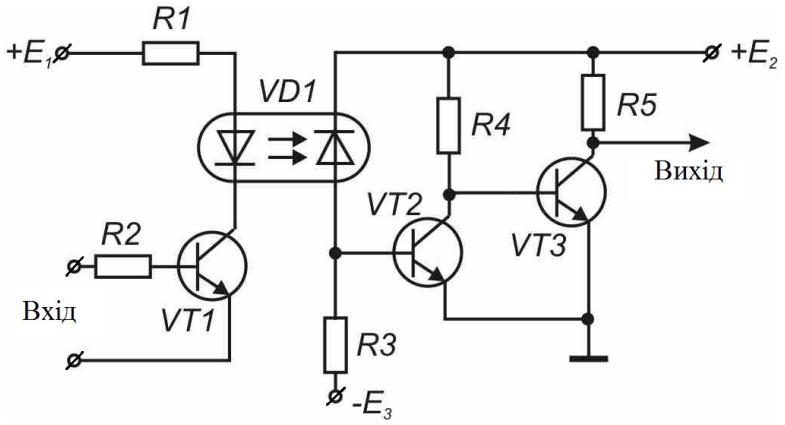


Рис. 9.35.

Провідна роль в цьому напрямі відводиться оптоелектронним приладам і пристроям. Річ у тому, що досить важливі технічні переваги трансформаторів та електромагнітних реле (гальванічна розв'язка ланцюгів управління і навантаження, надійне функціонування в потужних, високовольтних системах) властиві й оптронам. В той же час оптоелектронні вироби істотно перевершують електромагнітні аналоги по надійності, довговічності, перехідним і частотним характеристикам. Управління компактними і швидкодіючими оптоелектронними трансформаторами, перемикачами, реле, здійснюється за допомогою інтегральних мікросхем цифрової техніки без спеціальних засобів електричного узгодження. Приклад заміни імпульсного трансформатора наведений на рис. 9.35.

9.5.1. Пристрої індикації, перемикання та захисту

Поява світловипромінюючих діодів (СВД) дозволило створити малогабаритні безконтактні пристрої перемикання та захисту від перевантажень і короткого замикання. Під час контролю технологічних процесів інколи виникає необхідність безконтактного дистанційного перемикання (наприклад, у вибухонебезпечних середовищах). Для цього можна застосовувати дистанційні вимикачі на основі СВД – фотоприймач з відкритим оптичним каналом.

На рис. 9.36 приведені структурні схеми дистанційних вимикачів, які складаються з випромінювача (СВД), фотоприймача (рис. 9.36, а, б) і світловодів (рис. 9.36, в, г).

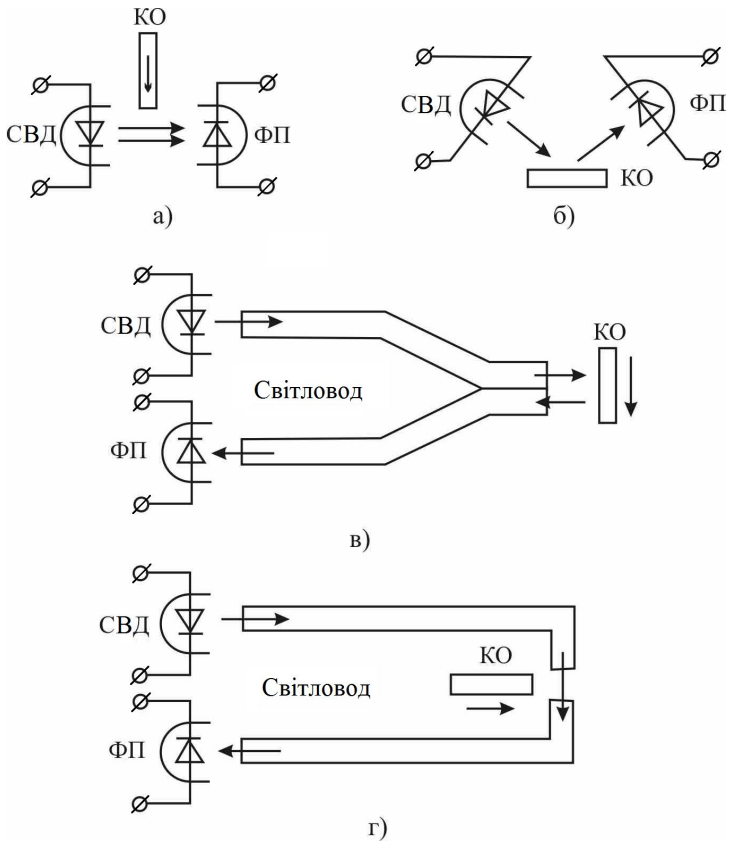


Рис. 9.36.

Принцип дії цих пристроїв заснований на тому, що контрольований об'єкт КО перериває світловий потік, або потрапляючи між випромінювачем і фотоприймачем, відбиває світловий потік, який далі падає на фотоприймач і перетворюється у фотоелектричний сигнал. Використання перемикачів зі світлодіодами доцільне в тих випадках, коли по умові технологічних процесів випромінювачі та фотоприймачі неможливо розташовувати безпосередньо біля контрольованого об'єкту, наприклад при високих температурах, в агресивних середовищах і вибухонебезпечних приміщеннях.

На рис. 9.37 приведена схема, що є прикладом використання СВД в індикаторах напруги акумуляторів. До складу індикаторів входить дільник 1 контрольованої напруги джерела 2, що складається з трьох послідовно сполучених резисторів. Середній резистор 3 дільника ввімкнений між входами логічного елементу 4 типу «НІ» і генератора 5, причому вихід елементу 4 підключений до входу іншого генератора 6, а виходи генераторів сполучені з входами логічного елементу 7 типу «І-НІ», до виходу якого підключений світловипромінюючий діод.

Керовані генератори 5 і 6 виконані так, що при появі на вході лог. 1 вони запускаються і починають видавати імпульсні сигнали частотою 4 і 1 Гц відповідно, а при надходженні 0 видають 1.

Живлення логічних елементів і генераторів може здійснюватися від контрольованого акумулятора. Для цього в індикаторі можуть бути

використані мікросхеми серії К511 або К561, що мають діапазони напруг живлення 10,8…22 і 3…15 В відповідно.

Припустимо, що дільник напруги 1 налаштований так, що на його виводах А і Б потенціали перевищують вхідну порогову напругу перемикання логічних елементів при перевищенні напруги акумулятора 11 і 14 В відповідно. При цьому індикатор працює таким чином. Якщо в точках А і Б дільника 0, то генератор 6 збуджений, а генератор 5 зупинений і 1 з його виходу дозволяє проходження імпульсів генератора 5 через елемент 7 на СВД, який видає імпульсний світловий сигнал зі зниженою частотою 1 Гц, що сигналізує про зменшення контрольованої напруги нижче рівня 11 В.

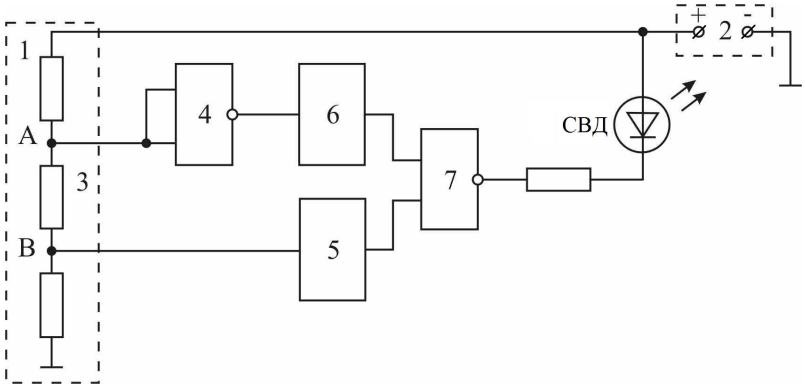


Рис. 9.37.

Якщо в точках А і В відповідно 1 і 0, то генератори 5 і 6 зупинені, і сигнали 1 з їхніх виходів відмикають логічний елемент 7, при цьому СВД включений безперервно, що свідчить про те, що контрольована напруга знаходиться в діапазоні 11…14 В.

Якщо в точках А і Б дільника 1 встановлена 1, то генератор 5 збуджений, а генератор 6 зупинений і СВД видає імпульсний світловий сигнал з підвищеною частотою 4 Гц, що сигналізує про перевищення контрольованої напруги рівня 14 В.

Змінюючи опори резистора дільника 1, можна встановлювати необхідні пороги контрольованих напруг. Таким чином, пристрій дозволяє вконувати трьохрівневу індикацію напруги акумулятора за допомогою одного елементу сигналізації за рахунок використання його станів «Імпульси з малою частотою» (1 Гц), «Ввімкнено» і «Імпульси з великою частотою» (4 Гц).

9.5.2. Пристрої контролю та вимірювання на світловипромінюючих діодах

Такі переваги СВД, як швидкодія, значна монохроматичність спектральних характеристик, малі габарити, висока надійність, 100% модуляція світлового струму, що протікає через СВД, та інші, дозволяють

розробляти різноманітні пристрої для контролю і вимірювання параметрів широкого класу речовин і матеріалів.

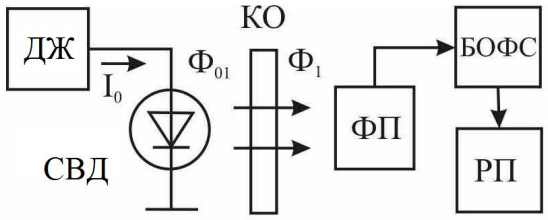
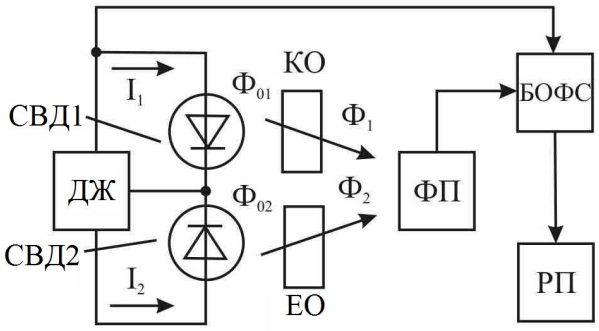
Оптоелектронні пристрої можна класифікувати за наступними основними ознаками: вигляду вимірюваної фізичної величини; кількості використаних ділянок спектру; способу перетворення вимірюваного параметру у фотоелектричний сигнал; способу обробки фотоелектричного сигналу.

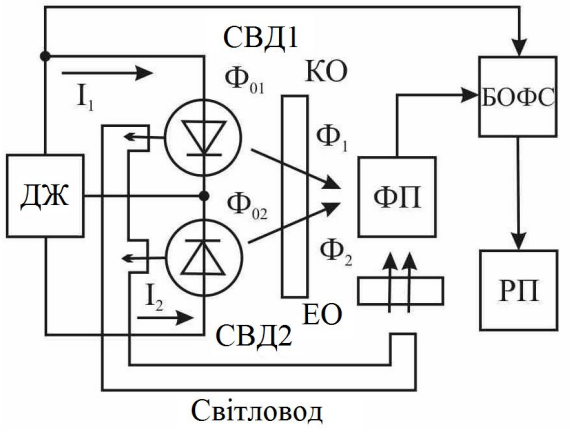
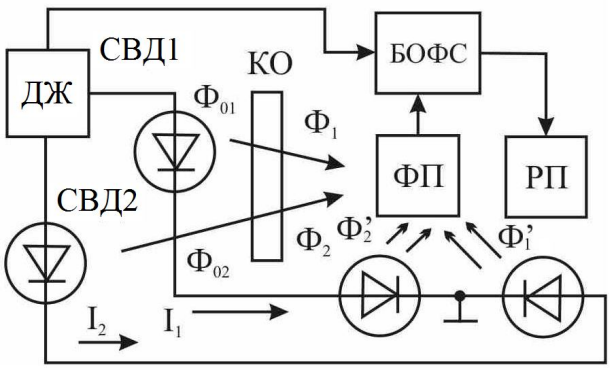
За видом фізичної величини, що вимірюється, оптоелектронні вимірювальні пристрої (ОВП) розділяються на фотометри та аналізатори складу речовин і матеріалів.

За числом ділянок спектру, що використовуються, ОВП розділяються на хвильові, двохвильові та багатохвильові, а по числу каналів (числу використаних СВД, випромінюючих на одній довжині хвилі) – на одноканальні, двоканальні і багатоканальні пристрої.

Поняття «хвиля» і «канал» дуже важливі для структурного аналізу, тому для розрізнення цих понять введемо позначення: Х – хвиля, К – канал, Б – багатоканальний, багатохвильовий, а цифрами перед буквами позначається кількість довжин хвиль і каналів. Наприклад: 2Х1К – двохвильова одноканальна структурна схема. У табл. 9.5 приведені основні (базові) структурні схеми пристрою на СВД згідно приведеної класифікації.

Табл. 9.5.

Позначення	Структурна схема
1Х1К	
1Х2К	

2Х2К	
2Х2К	

На рисунках позначені: ДЖ – джерело живлення СВД; СВД – світловипромінюючий діод; ФП – фотоприймач; БОФС – блок обробки фотоелектричного сигналу; РП – реєструючий пристрій; Φ_{o1} и Φ_{o2} – потоки випромінювання на опорній і вимірювальній довжинах хвиль, на випромінюваних СВД, I_1, I_2 – струми світловипромінюючих діодів; Φ_1 і Φ_2 – потоки, що пройшли через контрольований об'єкт (КО) та еталонний об'єкт (ЕО).

Принцип перетворення контрольованого параметру у фотоелектричний сигнал заснований на наступному. Контрольований об'єкт опромінюють потоком випромінювання (для одинхвильових, одноканальних структурних схем), або двома потоками на різних довжинах хвиль, одна з яких лежить на смузі поглинання контрольованим параметром, інша – поза смугою, приймають долю інтенсивності випромінювання, що пройшло або було відбито від поверхні об'єкту, перетворюють в електричний сигнал, пропорційний значенню контрольованого параметру.

Таким чином, для побудови пристроїв контролю і вимірювання СВД перш за все необхідні відомості про оптичні властивості контрольованого об'єкту. За спектральними характеристиками і в залежності від виду вимірювання вибирають відповідний смугам поглинання СВД і необхідний

режим його роботи. Для забезпечення максимальної вибірковості смуги поглинання контрольованого параметра повинні максимально перекриватися спектральними характеристиками СВД. Далі підбирають фотоприймач і узгоджують його з пристроями підсилення і обробки сигналу.

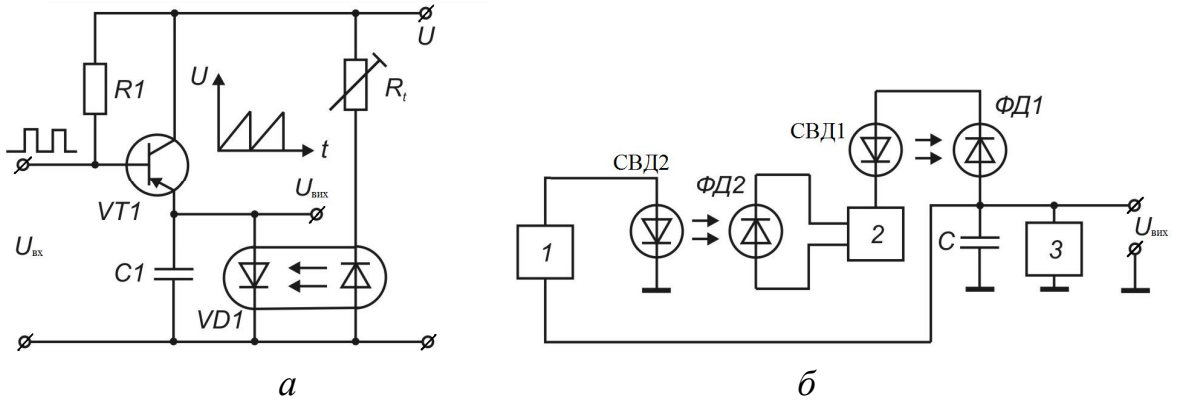
9.5.3. Імпульсні та обчислювальні пристрої

Використання СВД в імпульсних пристроях дозволяє значно спростити різні схеми автоматики. Широко відомі оптоелектронні пари СВД – фоторезистори, або СВД – фотодіод, або СВД – фототранзистор вже самі по собі можуть бути використані як елементарні ключові схеми. У пристроях, де використовуються звичайні напівпровідникові діоди, зручно замінити їх СВД. При цьому можна здійснити додатковий оптичний зв'язок в схемі або індикацію.

На основі елементів СВД – фотоприймач можна створювати генератори і формувачі імпульсів різної форми. Найчастіше використовуються генератори пилоподібної напруги. Отримати імпульси пилоподібної напруги можна за допомогою RC – ланцюга, але при цьому важко забезпечити задану лінійність. Лінеаризацію вихідної пилоподібної напруги можна здійснити параметричним методом і за допомогою зворотних зв'язків.

Параметрична лінеаризація здійснюється за рахунок розряду (або заряду) конденсатора через струмостабілізуючий двополюсник, в якості якого можна використовувати фотодіод.

На рис. 9.38, а, представлена схема генератора пилоподібної напруги з параметричною стабілізацією через струмостабілізуючий фотодіод, оптично пов'язаний з СВД. Основним недоліком цієї схеми є низька економічність і нелінійність сформованої пилоподібної напруги.



а *б*

Рис. 9.38.

Лінійність можна значно підвищити, використовуючи негативний зворотний зв'язок (НЗЗ). На рис. 9.38, б, наведена схема генератора пилоподібної напруги з НЗЗ. Струм в ланцюзі основного

світловипромінюючого діода СВД, підключеного до виходу підсилювача 2, можна представити у вигляді суми струмів:

$$I_{с.п.} = I_{п.с.} + I_{фд.п.}, \tag{9.1}$$

де $I_{п.с.}$ – постійна складова струму СВД; $I_{фд.п.}$ – струм додаткового фотодіоду.

Враховуючи, що додатковий фотодіод оптично пов'язаний з додатковим світловипромінюючим діодом СД, який підключений через підсилювач потужності до виходу генератора, попереднє рівняння можна записати у вигляді:

$$I_{с.д1.} = I_{с.д} + mu_{вих.},$$

де m – коефіцієнт перетворення вихідної напруги в струм СВД.
З врахуванням рівняння (9.1)

$$K_1 = \frac{1}{R_y} + \frac{1}{R_{пр}} + C_{с.п2.},$$

де K_1 – коефіцієнт передачі оптопари; R_y и $R_{пр.}$ – еквівалентні опори керованого ключа 3, отримаємо напругу на конденсаторі C:

$$u_c = \frac{K + C + U_a}{C},$$

де U_a – напруга живлення; C – ємність конденсатора.

Використовуючи зворотні зв'язки з фотодіодними і фоторезисторами, можна отримати широкий клас степеневих функцій.

Основними елементами автоколивальних імпульсних схем є конденсатори, за допомогою яких задається необхідна частота повторення імпульсів. Для отримання імпульсів низької частоти необхідні великі ємності. Збільшення ємності призводить до збільшення габаритів генератора. Можна отримати періодичну послідовність прямокутних імпульсів, використовуючи пари СВД – фотоприймач з ввімкненням в частотнозадаючий ланцюг.

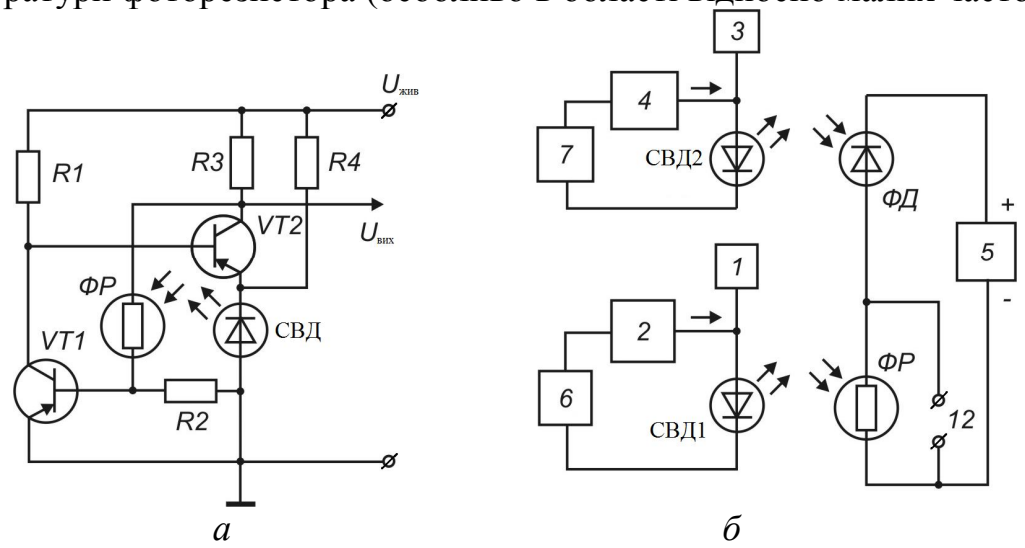
На рис. 9.39, а, приведена схема безреактивного оптоелектронного мультивібратора. При підключенні генератора до джерела живлення транзистор VT1 закритий, оскільки темновий опір фоторезистора ФР великий, а транзистор VT2 відкритий. Через СВД, включений в емітерний ланцюг транзистора VT2, протікає струм, що викликає світіння СВД. Останній оптично пов'язаний з фоторезистором, і тому опір фоторезистора знижується з деякою постійною часу, в залежності від інерційних

властивостей фоторезистора і рівня випромінювання до тих пір, поки струм, що протікає через фоторезистор, не буде достатній для відмикання транзистора VT1. Оскільки напруга на колекторі відкритого транзистора VT2 невелика, то для відмикання VT1 необхідне значне зменшення опору фоторезистора. Це призводить до зниження нижньої границі частотного діапазону.

При відкриванні транзистора VT1 негативна напруга на його колекторі зменшується, і при деякому його значенні починається процес замикання, а VT1 відмикається до насичення.

При включенні СВД опір фоторезистора зростає до моменту, поки струм зсуву, що задається фоторезистором, не виявиться недостатнім для насичення транзистора VT1. Транзистор VT1 закривається, а VT2 відкривається, потім цикл повторюється.

Слід зазначити, що частота імпульсів, що генеруються, залежить від температури фоторезистора (особливо в області відносно малих частот).



а *б*

Рис. 9.39.

Великі можливості для побудови різних нелінійних перетворювачів і їх моделей дає використання пари СВД – фотоприймач. Схема такого пристрою приведена на рис. 9.39, б. У її склад входить: світловипромінюючий діод СВД, вихідний фоторезистор ФР, джерело постійного зсуву 1, перетворювач 2 напруги в струм, фотодіод ФД, додаткове джерело постійного зсуву 3, додатковий перетворювач 4 напруги і струму, додатковий світловипромінюючий діод СВД2 і джерело живлення 5.

Світловипромінюючий діод СВД2, електрично сполучений з перетворювачем 4 напруги в струм, і фотодіод ФД є першою оптопарою; СВД1, сполучений з перетворювачем напруги в струм, і фоторезистор ФР є другою оптопарою. Обидва СВД підключені до джерел постійного зсуву, і джерел живлення 6 і 7. Фотодіод і фоторезистор включені послідовно і приєднані до джерела живлення 5. Робота пристроїв заснована на лінійній

залежності струму фотодіода від падаючого потоку світла незалежно від напруги живлення.

9.6. Оптичні запам'ятовуючі пристрої (CD-ROM, DVD-ROM)

Компакт-диск – це диск діаметром 120 мм, виготовляється з полімерів і покритий металевою плівкою, зазвичай сплавом на основі алюмінію (рис. 9.40). Інформація зчитується саме з цієї металевої плівки, яку додатково покривають шаром прозорого полімеру, який захищає дані від механічного пошкодження. Зчитування інформації з диску відбувається за рахунок реєстрації змін інтенсивності відбитого від алюмінієвого шару випромінювання малопотужного лазера. Приймач (фотодатчик) визначає, чи відбився промінь лазера від гладкої поверхні, або був розсіяний (поглинутий). Розсіювання або поглинання променя відбувається в тих місцях, де в процесі запису на диск були нанесені поглиблення. Сильне відбиття відбувається в тих областях диску, де таких поглиблень немає. Фотодатчик сприймає лазерний промінь, відбитий від поверхні диска, потім ці сигнали поступають в мікропроцесор зчитуючого пристрою, там вони перетворюються в двійкові дані. Глибина кожного поглиблення на диску дорівнює 0,12 мкм, ширина – 0,60 мкм. Поглиблення розташовані у вигляді спіральної доріжки, відстань між сусідніми витками якої складає 1,60 мкм, що відповідає щільності 16 тисяч витків на дюйм, або 625 витків на мм. Довжина штрихів уздовж доріжки запису може коливатися від 0,9 мкм до 3,3 мкм. Доріжка починається на деякій відстані від центрального отвору диска і закінчується приблизно в 5 мм від краю диска.

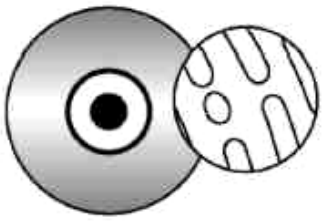


Рис. 9.40.

Якщо на компакт-диску (звуковому або інформаційному) необхідно відшукати місце запису певних даних, то його координати попередньо зчитуються зі змісту диска, після чого зчитувальний пристрій переміщується до потрібного витка спіралі і чекає появи визначеної послідовності бітів.

Дані на компакт-дисках записуються з використанням технології CLV (Constant Linear Velocity – запис з постійною лінійною швидкістю). Це означає, що запис і відтворення даних з компакт-диска відбуваються з постійною лінійною швидкістю переміщення доріжки відносно

зчитуючого пристрою. Іншими словами, при зчитуванні інформації з внутрішніх доріжок диск повинен обертатися швидше, а при зчитуванні із зовнішніх – повільніше. Цей спосіб застосовується тому, що спочатку компакт-диски були призначені для відтворення звуку, при якому була потрібна постійна швидкість зчитування даних. У зв'язку з цим спіраль компакт-диска розбивається на блоки (сектори), частота яких при запису і відтворенні складає 75 блоків в секунду. Це означає, що при повному часі зчитування, що дорівнює 74 хвилини, на диску розташовується 333 тис. блоків (секторів).

Типовий пристрій CD-приводу приведено на рис. 9.41.

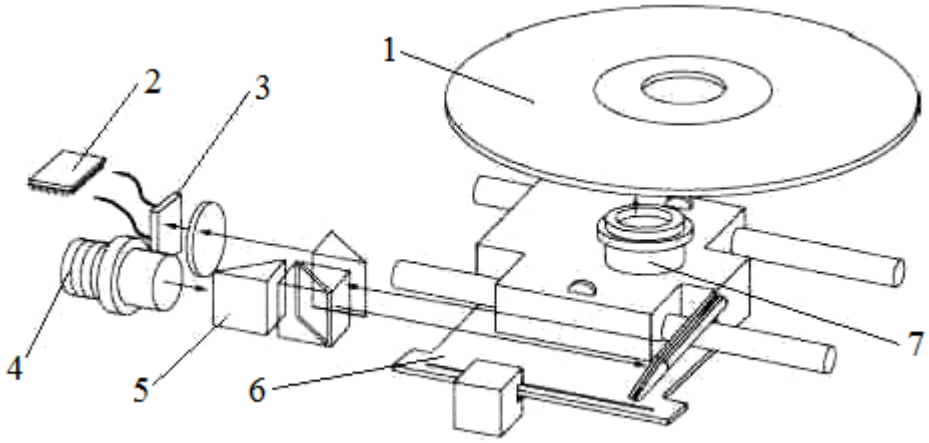


Рис. 9.41.

Напівпровідниковий лазер (4) генерує малопотужний лазерний промінь, який потрапляє на відбиваюче дзеркало. Двигун, керований мікропроцесором, зміщує рухливу каретку (6) з відбиваючим дзеркалом і фокусуючою лінзою (7) до потрібної доріжки компакт-диску (1). Промінь лазера фокусується на поверхні диска за допомогою лінзи, а потім лінза фокусує відбитий від поверхні диска промінь. Цей промінь за допомогою оптичної системи (5) подається на фотоприймач (3), який перетворює прийняті світлові імпульси в електричні, які потім відповідним чином розшифровуються контролером (2) і передаються в комп'ютер у вигляді цифрових даних.

Типова проблема при роботі приводу – пристрій розкручує диск, але прочитати нічого не може. Відповідно, проблема цілком ймовірно полягає в оптиці пристрою. Типові рекомендації такі: слід розібрати пристрій і переконатися в тому, що фокусуюча лінза, яка розташована під диском, не забруднена. В разі забруднення її потрібно вкрай обережно протерти. Також може бути забруднена поверхня втулки, яка обертає диск. Її теж необхідно очистити. Нерідко після цього працездатність пристрою повністю відновлюється.

Неможливість зчитування даних з дисків, особливо для старого приводу, може пояснюватися тим, що напівпровідниковий лазер "постарів" і не дає нормального пучка світла. У такому разі можна спробувати виправити ситуацію, піднявши струм в напівпровідниковому лазері. На лазерному діоді зазвичай є крихітний змінний резистор з. Покрутивши його можна збільшити струм в лазерному діоді. Якщо при обертанні в один бік, привід не починає краще читати, то потрібно спробувати змінити напрям обертання. Нерідко такий метод допомагає, проте не варто встановлювати дуже великий струм – це призводить до додаткового навантаження на лазерний діод.

Існують пристрої, які дозволяють здійснювати запис на лазерні диски. Такі пристрої називають CD-RW Drive, а сам диск, на який здійснюється запис, називається просто CD-R. Чистий CD-R не має на своїй поверхні жодного поглиблення, він представляє собою чисту поверхню. При записі на цій поверхні лазерний промінь записуючого пристрою випалює поглиблення, виконуючи таким чином запис інформації. В якості CD не застосовують алюмінієві диски, матеріал підкладки – золото. Зрозуміло, що записавши інформацію на такий диск, її не можна стерти або змінити, оскільки при записі шар золота незворотньо змінюється. Пристрій CD-RW Drive здатний також зчитувати інформацію, записану як на золотих, так і на звичайних алюмінієвих дисках. Швидкість таких пристроїв маркірують зазвичай двома числами, наприклад 4x/24x: запис здійснюється на 4x швидкості, зчитування – на 24x.

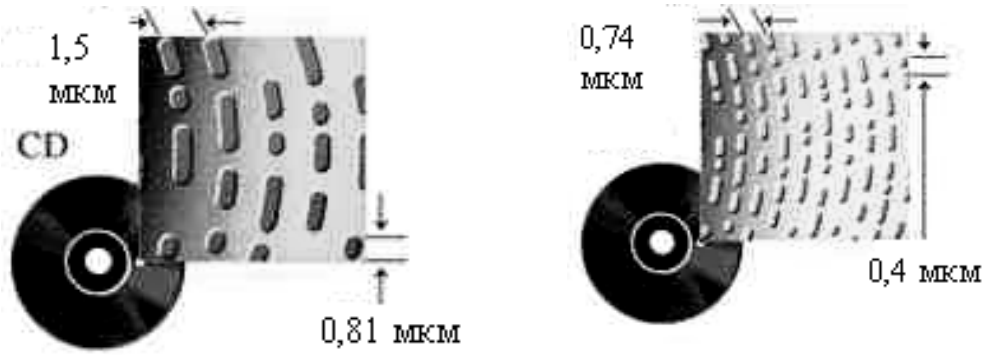
Окрім одноразово записуваних CD-R існують і багаторазово перезаписувані диски CD-RW (CD Rewritable). Диск CD-RW має активний шар із сплаву Ag-In-Sb-Te (срібло-індій-сурма-телур). Такий шар при нагріванні до 500-700 ^{0}C розплавляється, втрачаючи відбиваючі властивості, таким чином можна здійснити запис інформації. При нагріванні до температури близько 200 ^{0}C відбиваючі властивості шару відновлюються, таким чином запис на такий носій можна здійснювати багато разів.

Проте не всі CDROM Drive читатимуть CD-RW диски. Річ у тому, що стандарт CDROM вимагає, щоб відбиваюча здатність поверхні CD була не нижче 70%, а відбиваюча здатність поглиблень – не більше 28%. Відбиваюча здатність CD-RW – близько 25%, але різниця між відбиваючою здатністю поверхні диска і поглиблень досить велика.

Цифровий універсальний диск, що називається DVD (Digital Versatile Disc), значно збільшує об'єм пам'яті. Головна проблема сучасної технології CD-ROM полягає в тому, що вона жорстко обмежена об'ємом пам'яті диску. Диск CD-ROM (рис. 9.42, а) може містити максимум 650 Мбайт даних. Відповідно до стандарту DVD-диск (рис. 9.42, б) є однобічним, одношаровим і містить 4,7 Гбайт інформації. Новий диск має такий самий діаметр, як сучасні компакт-диски, проте він в два рази тонший (0,6 мм).

Для збільшення ємності DVD-диску можна змінювати такі параметри:

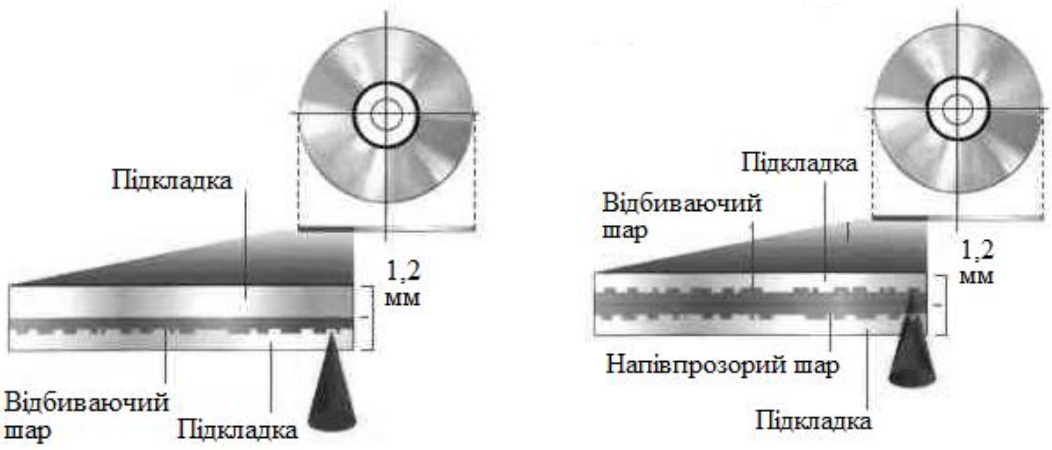
– зменшувати довжину штриха від 0,972 до 0,4 мкм);
– зменшувати відстань між доріжками (від 1,6 до 0,74 мкм);
– збільшувати область даних (від 86 до 87,6 см^2);
– підвищувати ефективність коду корегування помилок.



а *б*

Рис. 9.42.

Більшість дисків DVD мають ємність 4,7 Гб. Застосування схем подвоєння щільності та їх комбінування дозволяє отримати диски більшої ємності: від 8,5 Гб і 9,4 Гб до 17 Гб (рис. 9.43, а – одношаровий, б – двошаровий DVD-диски).



а *б*

Рис. 9.43.

Однобічний одношаровий диск – це найпростіша структура DVD диску. На такому диску можна розмістити до 4,7 Гб даних. Ця ємність в 7 разів більше ємності звичайного CD-диску. Однобічний двошаровий диск має два шари даних, один з яких напівпрозорий. Обидва шари зчитуються з одного боку і на такому диску можна розмістити 8,5 Гб даних, тобто на 3,5 Гб більше, ніж на одношаровому однобічному диску.

Всі перераховані пристрої підключаються абсолютно аналогічно жорсткому диску. Вони використовують інтерфейс IDE або SATA, підтримують ті ж режими PIO 1, 2, 3, 4, DMA 33, 66, 100. Для підключення всіх перерахованих пристроїв застосовують 40 контактні (40 або 80 жильні) кабелі, підтримується по два пристрої на кабель, при цьому необхідно встановити перемички на кожному пристрої, що підключається, в положення Master і Slave. На один кабель можна підключати жорсткі диски і перераховані пристрої оптичного зберігання даних.

9.7. Розповсюдження світла в світловодах

У волоконно-оптичних лініях передача інформації здійснюється за допомогою оптичних сигналів по волоконно-оптичних лініях з малими втратами. До складу ліній входять також джерело випромінювання, що перетворює електричні сигнали в оптичні, і фотоприймач, що здійснює зворотнє перетворення. Вони знаходять широке застосування в системах зв'язку і передачі інформації, а також у волоконно-оптичних датчиках [35].

Основним елементом волоконно-оптичної системи є світловод – тонка гнучка нитка, виготовлена з прозорого матеріалу таким чином, що її центральна частина оптично більш щільна, ніж периферійні області. Тому світловий промінь, що вводиться у волокно, може поширюватися уздовж його вісі, і пройти з малим загасанням значну відстань. У найпростішому випадку оптичне волокно представляє собою тонку нитку круглого перетину з прозорого в заданій спектральній області матеріалу (скло, кварц, різноманітні полімери). Скляні нитки зі зменшенням товщини до 0,1-0,15 мм стають гнучкими і не зламуються навіть при вигинах з радіусом 2-3 мм.

При побудові мереж використовуються багатожильні кабелі (рис. 9.44).

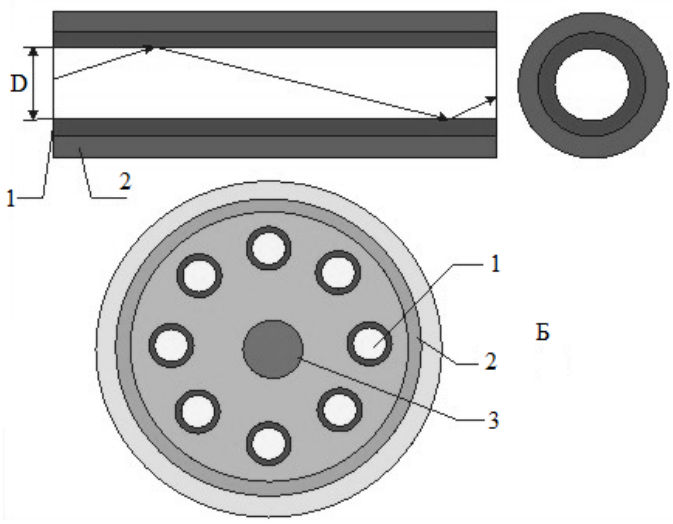


Рис. 9.44.

Світло (довжина хвилі 1350 нм або 1500 нм) вводиться в оптоволокно (діаметром d<100мкм) за допомогою світловипромінюючого діода або напівпровідникового лазера. Центральне волокно 1 покривається шаром, коефіцієнт заломлення якого менше, ніж в центральному ядрі (стрілками умовно показаний хід променів світла у волокні). Для забезпечення механічної міцності ззовні волокно покривається полімерним шаром 2. Кабель може містити багато волокон, наприклад 8. В центрі кабелю міститься сталевий трос 3, який використовується при прокладанні кабелю. Із зовнішнього боку кабель захищається сталевою оболонкою 2 і герметизується еластичним полімерним покриттям.

Існує декілька типів оптичних волокон, що володіють різними властивостями. Вони відрізняються один від одного залежністю коефіцієнта заломлення від радіусу центрального волокна. На рис. 9.42 показано три різновиди волокна (А, Б і В). Буквами А і Б помічений мультимодовий тип волокон. Тип Б має меншу дисперсію часу поширення, і з цієї причини вносить менші спотворення форми сигналу. Встановлено, що надаючи світловим імпульсам певну форму (зворотний гіперболічний косинус), дисперсійні ефекти можна повністю виключити. При цьому з'являється можливість передавати імпульси на відстань в тисячі кілометрів без спотворення їх форми. При сучасних технологіях необхідно використовувати повторювачі через кожні 30 км. У спрощеному вигляді можна вважати, що мода – це одна з можливих траекторій, по якій може поширюватися світло у волокні. Чим більше мод, тим більше дисперсійне спотворення сформи сигналу. Одномодове волокно дозволяє отримати смугу пропускання в діапазоні 50-100 ГГц/км.

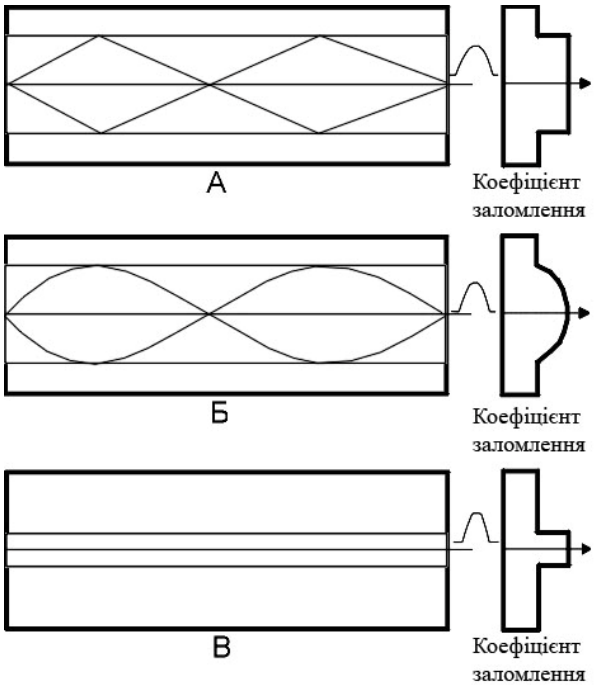


Рис. 9.45.

Кількість мод, що має волокно, певною мірою визначає його інформаційну ємність. Модова дисперсія призводить до розпливання імпульсів і їх накладання один на одного. Дисперсія залежить від діаметру центральної частини волокна і довжини хвилі світла. Число мод *n* для волокна типу А дорівнює:

$$N = \frac{2\pi^2 d^2 A^2}{\lambda^2},$$

де *d* – діаметр центральної частини (ядра), *A* – чисельна апертура волокна, λ – довжина хвилі. Волокно з діаметром центральної частини волокна 50 мкм підтримує 1000 мод. Для волокна типу Б (рис. 9.45) значення *n* у два рази менше.

Очевидно, що чим більша довжина хвилі, тим менша кількість мод і менше спотворення сигналу. Це, зокрема, є причиною роботи в довгохвильовому інфрачервоному діапазоні. Але навіть для однієї і тієї ж моди різні довжини хвиль поширюються по волокну з різною швидкістю. Волокно із згладженим профілем показника заломлення має дисперсію 1 нсек/км і менше. Це, зокрема, пов'язано з тим, що світло в перефірійних областях волокна з більшою довжиною траєкторії рухається швидше (адже там менший коефіцієнт заломлення). Одномодовий режим реалізується тоді, коли довжина хвилі світла стає зрівняною з діаметром ядра волокна. Довжина хвилі, при якій волокно стає одномодовим, називається пороговою. Волокно з діаметром 50 мкм може підтримувати до 1000 мод. На відміну від багатомодового волокна, в одномодовому випромінювання присутнє не лише всередині ядра.

Загасанням зазвичай називається ослаблення сигналу по мірі його проходження по волокну. Воно вимірюється в децибелах на кілометр і варіюється від 300 дБ/км для пластикових волокон до 0,21 дБ/км – для одномодових волокон. Смуга пропускання волокна визначається дисперсією. Приблизно смугу пропускання одномодового волокна можна оцінити згідно з формулою [35]:

$$BW = 0{,}187/(Disp \cdot SW \cdot L),$$

де Disp – дисперсія на робочій довжині хвилі; SW – ширина спектру джерела; L – довжина волокна;

Якщо діаметр джерела світла не відповідає діаметру ядра волокна, то втрати світла, пов'язані з геометричною неузгодженістю можуть бути охарактеризовані наступною формулою:

$$\text{Втрати}_{\text{діам}} = 10\log_{10}(\text{Діаметр}_{\text{волокна}}/\text{Діаметр}_{\text{джерела}})^2.$$

Втрат немає, коли волокно має діаметр більший за діаметр джерела світла. Якщо числова апертура джерела більше апертури волокна, то втрати світла складуть:

$$\text{Втрати}_{\text{діам}} = 10\log_{10}(A_{\text{волокна}}/A_{\text{джерела}})^2.$$

Окрім дисперсії швидкодія оптичного каналу обмежується шумами. Шуми мають дві складові: дробовий і тепловий шум. Дробовий шум визначається співвідношенням:

$$i_{sn}^2 = 2eiB,$$

де e – заряд електрона, i – середній струм, що протікає через приймач, В – ширина смуги пропускання приймача. Типове значення дробового шуму складає 25 нА при температурі 25 ^{0}С. Тепловий шум характеризується співвідношенням:

$$i_{sn}^2 = (4kTB)/R_L$$

де k – стала Больцмана, T – температура за шкалою Кельвіна, B – ширина смуги пропускання приймача, R_L – опір навантаження. При смузі в 10 МГц і температурі 298 ^{0}К ця складова шуму дорівнює 18 нА. Однією із складових теплового шуму є темновий струм, який зростає на 10% при зростанні температури на 1 градус.

Джерела випромінювання, інжектованого у волокно, мають обмежену смугу частот. Так світловипромінюючі діоди випромінюють світло з шириною смуги 35 нм, а лазери 2-3 нм (лазери мають вужчу діаграму спрямованості, ніж діоди). Характеристики світлодіодів та інжекційних лазерних діодів приведені в табл. 9.6.

Табл. 9.6.

Параметри	Світлодіод	Інжекційні лазерні діоди
Вихідна потужність	0,5 – 11,5 мВт	3 – 10 мВт
Час наростання	1 – 20 нс	1 – 2 нс
Діапазон струму зсуву	5- - 150 мА	100 – 500 мА

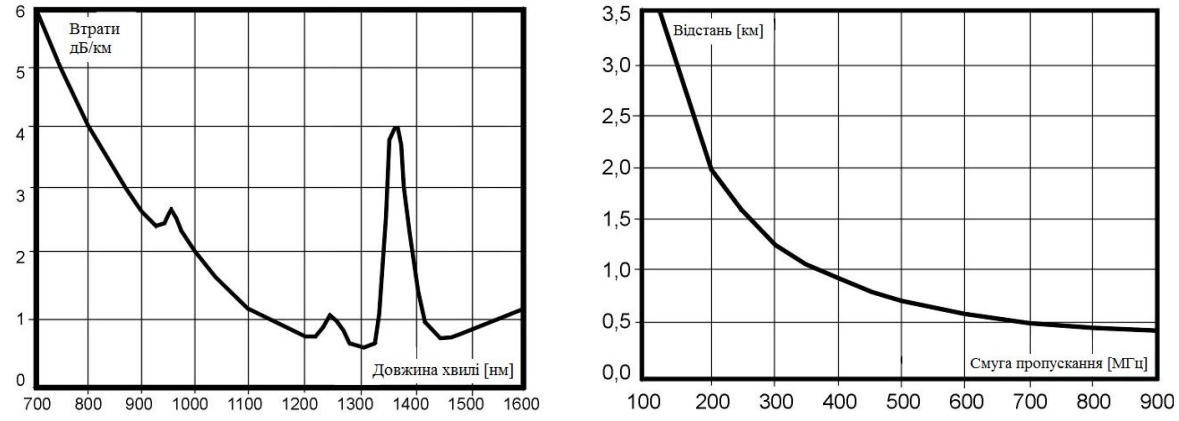
В табл. 9.7 приведені характеристики оптичних приймачів.

Табл. 9.7.

Параметри	pin	Лавинний фотодіод	Фототранзистор	Фотоприймач Дарлінгтона
Чутливість	0,5 мкА/мкВт	15 мкА/мкВт	35 мкА/мкВт	180 мкА/мкВт
Час наростання	1 нс	2 нс	2 мкс	40 мкс
Напруга зсуву	10 В	100 В	10 В	10 В

Поглинання світла у волокні відбувається з кількох причин. Поглинання у склі волокна зменшується з частотою, тоді як втрати через розсіювання на дефектах скла (релеєвське розсіювання) із збільшенням частоти зростає. При згинанні волокна поглинання збільшується. З цієї причини слід уникати малих радіусів вигину. В результаті втрати світла у волокні зазвичай лежать в діапазоні (2-5) дБ/км, для довжин хвиль 0,8 – 1,8 мкм.

Залежність поглинання світла у волокні від довжини хвилі показана на рис. 9.46, а. Залежність смуги пропускання волокна від його довжини приведена на рис. 9.46, б.



а *б*

Рис. 9.46.

З рис. 9.46, а, видно, що мінімуми поглинання розташований на 1300÷1500 нм, що і використовується для цілей телекомунікацій. При довжині хвилі 1300 нм дисперсія швидкостей поширення різних довжин хвиль мінімальна. Діапазон близько 850 нм характеризується високим поглинанням, але він привабливий тим, що лазери і фотоприймачі можуть бути виготовлені з одного матеріалу (арсеніду галію).

Типові характеристики оптичних волокон приведені в табл. 9.8.

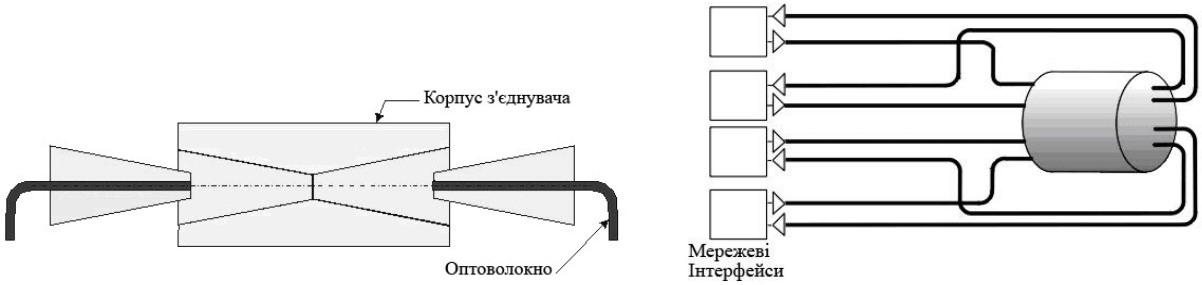
Табл. 9.8.

Тип волокна	Діаметр ядра [мкм]	Діаметр кледінгу [мкм]	A	Затухання [дБ/км]			Смуга пропускання [МГц/км]
Довжина хвилі				850	1300	1550	
Одномодове	9,3 8,1	125 125	0,13 0,17		0,4 0,5	0,3 0,25	5000 для 850 нм
Зі згладженим індексом	50 62,5 85	125 125 125	0,2 0,275 0,26	2,4 3,0 2,8	0,6 0,7 0,7	0,5 0,3 0,4	600 для 850 нм; 1500 для 1300 нм
Ступінчатий індекс	200	380	0,27	6,0			6 при 850 нм

9.8. Реалізація волоконно-оптичних ліній зв'язку

З'єднувачі для оптичних волокон мають конструкцію, показану на рис. 9.47, а, і виготовляються з кераміки. Втрати світла в з'єднувачі складають 10-20%. Для порівняння, зварка волокон приводить до втрат не більше 1-2%. Існує також техніка механічного зрощування волокон, яка характеризується втратами близько 10%. Оптичні аттенюатори для оптимального узгодження динамічного діапазону оптичного сигналу і інтервалу чутливості вхідного пристрою предсатвляють собою тонкі металеві шайби, які збільшують зазор між волокном кабелю і приймачем.

З використанням оптичних волокон можна створювати не лише кільцеві структури. Можлива побудова фрагменту мережі, по характеру зв'язків еквівалентного кабельному сегменту. Схема такого фрагмента мережі представлена на рис. 9.47, б.



а *б*

Рис. 9.47.

Базовим елементом цієї мережі є прозорий циліндр, на один з торців якого підключаються вихідні волокна всіх передавачів інтерфейсів пристроїв, що складають мережу. Сигнал з іншого кінця через волокна поступає на вхід фотоприймачів інтерфейсів. Таким чином сигнал, переданий одним з інтерфейсів, поступає на вхід всіх останніх інтерфейсів, підключених до цієї мережі.

Останнім часом помітного здешевлення оптичних каналів вдалося досягти за рахунок мультиплексування з діленням за довжиною хвилі. За рахунок цієї техніки вдалося в 16-160 разів збільшити широкосмуговість каналу з розрахунку на одне волокно. Схема мультиплексування показана на рис. 9.48. На вході каналу сигнали за допомогою призми об'єднуються в одне загальне волокно. На виході за допомогою аналогічної призми ці сигнали розділяються. Число волокон на вході і виході може досягати 32 і більше (замість призм останнім часом використовуються мініатюрні дзеркала, де застосовується 2D-розгортка по довжині хвилі). Розробка технології отримання особливо чистого матеріалу волокон дозволила розширити смугу пропускання одномодового волокна до 100 нм. Смуга одного каналу може лежати в діапазоні від 2 до 0,2 нм. Ця технологія в самий найближчий час розширить швидкість передачі даних по одному волокну з 1 до 10 Тбіт/с.



Рис. 9.48.

Призначення передавального модуля полягає в перетворенні вхідної інформації у вигляді електричних сигналів в оптичні сигнали, узгоджені з каналом передачі. «Центром» передавального модуля є випромінювач – саме в ньому відбувається оптоелектронне перетворення. Основні випромінювачі ВОЛЗ – напівпровідникові інжекційні лазери на основі з'єднань GaAlAs.

Приймальний модуль призначений для зворотнього перетворення оптичного сигналу, що поступає з каналу передачі (світловода), в електричний і його відновлення до початкового вигляду; через оптичний узгоджуючий елемент (зазвичай оптичний з'єднувач, а інколи і фокусуюча лінза) випромінювання поступає на фотоприймач, в якості якого практично завжди використовуються фотодіоди: лавинні і з *p-i-n*-структурою. Для спектрального діапазону $\Delta\lambda$ =0,8-0,9 мкм це кремнієві фотодіоди, для діапазону $\Delta\lambda$ = 1,3-1,6 мкм – фотодіоди на основі германію і, головним чином, на основі з'єднань InGaAs или InGaAsP.

Окрім передавальних і приймальних модулів в лініях дальнього зв’язку необхідні також ретранслятори, в даному випадку активні пристрої з оптичним входом і виходом, в яких здійснюється регенерація (відновлення) оптичних сигналів по мірі їх затухання при проходженні по світловоду. Практично ретранслятор представляє собою об'єднання приймального і передавального модулів, тобто оптоелектронний пристрій з подвійним перетворенням енергії (світло – електрика – світло).

Переваги волоконно-оптичних ліній зв'язку полягають в наступному:

1. Мале лінійне затухання і спотворення сигналу.

2. Надвисока пропускна здатність.

3. «Електрогерметичність». Волоконний світловод не чутливий до зовнішніх електромагнітних завад і сам практично не випромінює в навколишнє середовище. Цим забезпечується перешкодозахищеність ВОЛЗ і захист передачі інформації.

4. Малі габарити та маса.

5. Експлуатаційні переваги. Оптоелектронні принципи перетворення і передачі інформації у ВОЛЗ, використання кварцу в якості середовища для передачі інформації, обумовлюють наявність електричної розв'язки між входом і виходом лінії; однонапрямленість потоку інформації, відсутність зворотної реакції приймача на передавач.

Сфери практичного застосування ВОЛЗ надзвичайно широкі. В залежності від довжини каналу зв’язку вони умовно діляться на об'єктові, міські (міжміські, зонові) і магістральні, міжконтинентальні.

Передачу сигналів у ВОЛЗ здійснюють як в аналоговій, так і в цифровій формі. Широкосмуговість оптичних волокон робить переважною передачу інформації в цифровій формі, як більш точну і надійну, менш чутливу до шумів і спотворень.

Питання до розділу 9

1. Що таке фотоприймач?
2. Що відображає спектральна характеристика чутливості?
3. Що відображає енергетична характеристика чутливості?
4. Область використання фоторезисторів.
5. Назвіть основні параметри фоторезисторів.
6. Схема включення фотодіода.
7. Різновиди фотодіодів та їх особливості.
8. Призначення фототранзисторів та фототиристорів.
9. Принцип роботи приладів із зарядовим зв'язком.
10. Поясніть принцип роботи сканерів.
11. Структура та принцип роботи сонячних батарей.
12. Поясніть необхідність використання оптронів в електронних системах.
13. Різновиди оптронів.
14. Намалюйте кілька схем з використанням оптронів.
15. Структурні схеми дистанційних перемикачів.
16. Основні структурні схеми пристроїв контролю та вимірювання.
17. Типи оптичних волокон.
18. Типові характеристики оптичних волокон.
19. Особливості побудови волоконно-оптичних ліній зв’язку.

ДОДАТОК А. ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРІВ НА МОВІ СІ

Історично першою мовою програмування була машинна мова. Вона була дуже незручною для користувача, тому що потрібно було оперувати машинними кодами у двійковому або шіснадцятковому представленні. При вдосконаленні апаратної частини з'явилася інша, більш орієнтована для роботи з користувачем мова – асемблер. Для написання програм в асемблері використовуються мнемонічні коди машинної мови. Асемблер є досить незручним з погляду читабельності при написанні великих програм, для налагодження й пошуку помилок. Однак асемблер активно використовується для написання програм під мікроконтролери у зв'язку з компактністю коду та швидкістю його виконання. Як у програмуванні для комп'ютера, так і для програмування контролерів, на заміну асемблеру приходять мови програмування високого рівня. У цей час існує досить велике число компіляторів для роботи з мікроконтролерами, що реалізують різні мови програмування, такі як Basic, Pascal, Сі. Використання мови Сі є перспективним і дуже зручним для користувача. Переваги мови Сі у порівнянні з асемблером:

- легкість вивчення мови для початківців;
- простота написання програм;
- легкість читання;
- структурованість коду;
- можливість написання власних бібліотек;
- простота налагодження й пошуку помилок.

До недоліків можна віднести те, що код буде не оптимальним.

Існує велика кількість компіляторів мови Сі для AVR мікроконтролерів, які розроблялися для роботи з мовами високого рівня. Деякі з них є платними, інші безкоштовними або умовно безкоштовними. Основним критерієм при виборі того або іншого компілятора є наявність додаткових заголовних файлів для полегшення роботи з периферією та зовнішніми пристроями, а також умови поширення.

Компілятор мови Сі ImageCraft

Середовище розробки ImageCraft представляє собою компілятор мови Сі, графічну оболонку та автоматичний генератор програм, орієнтований на роботу із сімейством мікроконтролерів AVR фірми Atmel. Зовнішній вигляд вікна програми показаний на рис. А.1.

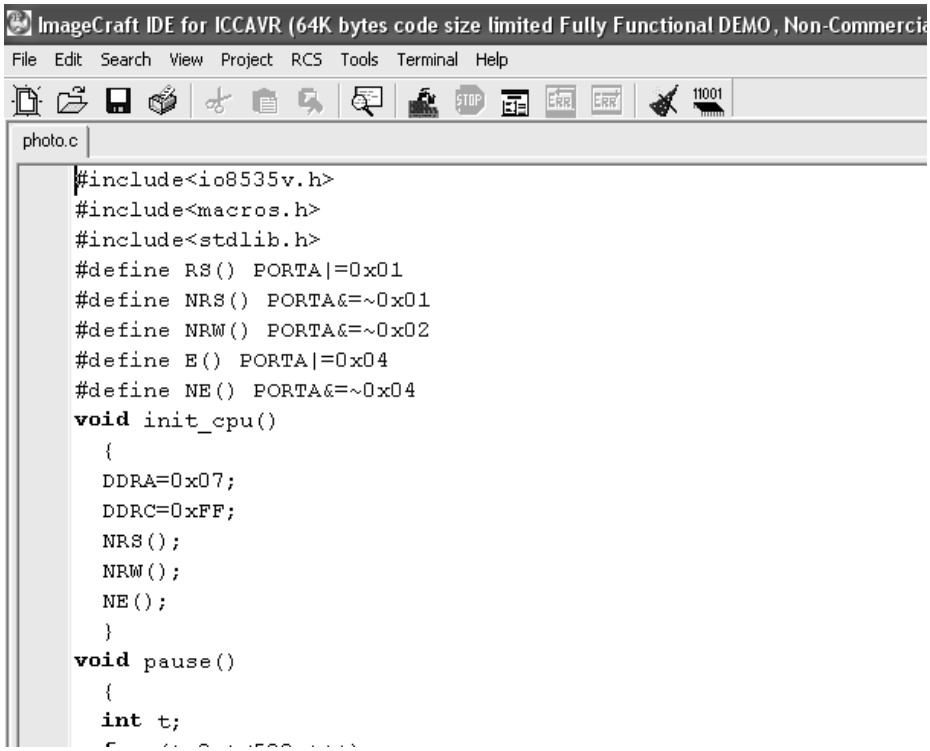


Рис. А.1.

Програма представляє собою 32-розрядний додаток для роботи в операційних системах Windows 2000, XP.

Компілятор містить у собі майже всі елементи, що відповідають стандарту ANSI. Також у компілятор включені додаткові можливості, орієнтовані на використання архітектурних особливостей мікроконтролерів цього сімейства.

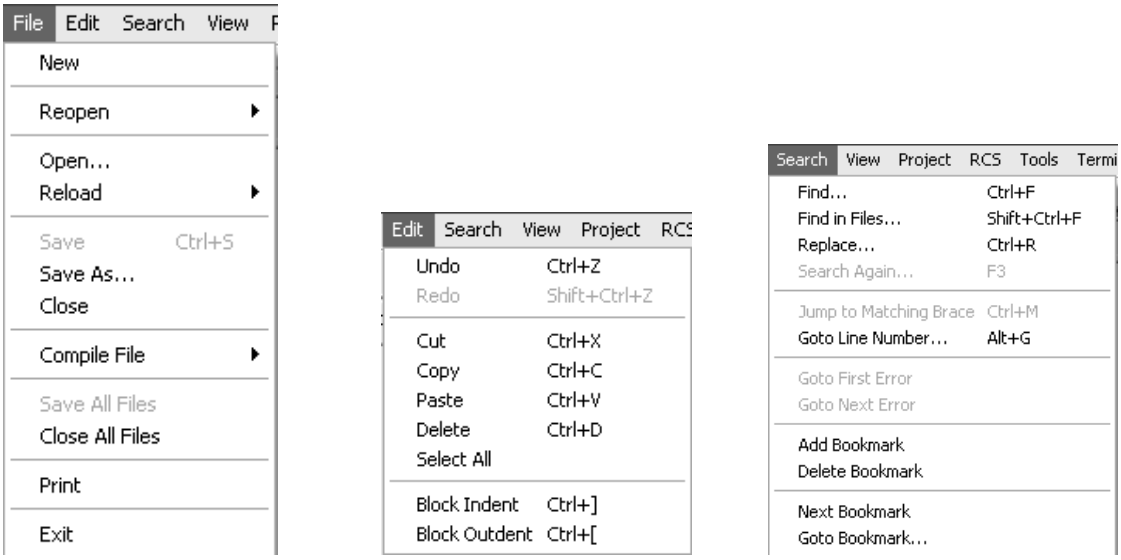
Об'єктні файли COFF дозволяють здійснювати налагодження програм з переглядом вмісту змінних. Для цього слід застосовувати вільно розповсюджуваний фірмою Atmel (www.atmel.com) відлагоджувач AVR Studio.

В ImageCraft є автоматичний генератор програм, що дозволяє протягом лічених хвилин одержати готовий код для наступних функцій:

- налаштування доступу до зовнішньої пам'яті;
- визначення джерела переривання Reset;
- ініціалізація портів введення/виведення;
- ініціалізація зовнішніх переривань;
- ініціалізація таймерів/лічильників;
- ініціалізація сторожового таймера;
- ініціалізація UART;
- ініціалізація аналогового компаратора;
- ініціалізація вбудованого АЦП;
- ініціалізація інтерфейсу SPI.

Середовище ImageCraft містить у собі програмне забезпечення для роботи з платою STK200. Після компіляції вихідної програми мовою Сі, отриманий код може бути безпосередньо запрограмований у мікроконтролер. Цей програматор використовує всього чотири сигнали: MOSI, MISO, SCK, RESET. Відомо багато найпростіших програматорів, що з'єднуються з послідовним або паралельним портом IBM сумісного персонального комп'ютера.

Для запуску компілятора ImageCraft вибираємо – меню "ПУСК" → "Програми" → "ImageCraft Development Tools" → "ICCV7 for AVR". Основне вікно містить стандартні елементи Windows-вікон – "Головне меню", "Панель інструментів", "Рядок стану". Основна робоча область розділена на три частини: "Редактор", "Вікно навігації" та "Вікно повідомлень". При першому запуску програми, всі три області порожні, але при наступних запусках програми можливий інший їхній стан. "Основне меню" містить кнопки "File", "Edit", "Search", "View", "Project", "RCS", "Tools", "Terminal" та "Help".



а *б* *в*

Рис. А.2.

Меню "File" (рис. А.2, а) містить наступні підменю:
"New" –використовується для створення нового файлу;
"Reopen" – для відкриття останніх файлів, що використовувалися;
"Open" – відкрити файл;
"Save" – збереження;
"Save as..." – зберегти як (можливо в іншому місці або з іншим ім'ям);
"Close" – закриття поточного файлу;
"Compile File" – компілювати файли;
"Save All Files" – зберегти всі файли;
"Close All Files" – закрити всі файли;
"Print" – друкування поточного файлу;

"Exit" – вихід із програми.

Меню "Edit" (рис. А.2, б) містить команди для роботи з текстом:

"Undo" – скасувати останню дію;

"Redo" – повернути скасовану дію;

"Cut", "Copy", "Paste", "Delete" – вирізати, копіювати, вставити, видалити виділений фрагмент тексту;

"Select All" – виділити все;

"Block Indent" / "Block Outdent" – зрушити виділений фрагмент праворуч/ліворуч.

Меню "Search" (рис. А.2, в) містить команди для роботи з текстом, пошуку заміни й переходу:

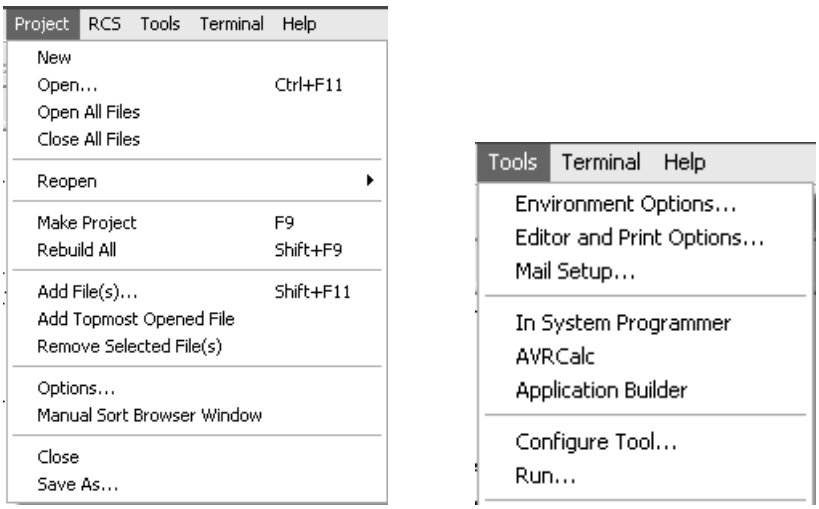
"Find" – знайти текст у поточному файлі;

"Find in Files" – знайти текст у всіх файлах проекту;

"Replace" – замінити фрагмент тексту;

"Go to Line Number" – перейти на рядок із зазначеним номером;

"Add Bookmark" / "Delete Bookmark" – додати / видалити закладку.



а *б*

Рис. А.3.

Меню "Project" (рис. А.3, а) дозволяє робити різні дії із проектом:

"New" – створити новий проект;

"Open" – відкрити існуючий проект;

"Reopen" – відкрити один з останніх проектів;

"Make project" – компілювати проект;

"Rebuild All" – перекомпілювати проект;

"Add Files" – додати файли в проект;

"Options" – налаштування властивостей проекту;

"Close" – закрити проект,

"Save As..." – зберегти проект.

Меню "Tools" (рис. А.3, б) містить інструменти та різні утілити для полегшення роботи з компілятором:

"Application Builder" – майстер, що допомагає налаштувати проект під конкретний пристрій, забезпечує генерацію коду для початкової ініціалізації пристрою;

"AVR Calc" – спеціальний калькулятор, що дозволяє переводити числа з однієї системи счислення в іншу, а також розраховувати значення дільників частоти для таймерів/лічильників, UART;

"In System Programmer" – вбудований програматор контролерів для програмування через SPI-інтерфейс;

"Configure Tool" – налаштування меню, дозволяє додавати в дане меню зовнішні програми для запуску (наприклад альтернативний програматор);

Меню "Terminal" – вікно термінала для роботи з UART, дозволяє налаштувати COM-порт.

Меню "Help" – допомога, містить розділи допомоги.

Панель інструментів забезпечує швидкий доступ до елементів меню, що найбільш часто використовуються. Панель інструментів містить наступні кнопки, розташування й розміри яких не змінюються.

 – створити новий файл;

 – відкрити файл;

 – зберегти файл (Ctrl+S);

 – друк файлу;

 – вирізати виділений фрагмент тексту (Ctrl+X);

 – копіювати виділений фрагмент тексту (Ctrl+C);

 – вставити (Ctrl+V);

 – знайти (Ctrl+F);

 – побудувати проект (Shift+F9);

 – налаштування проекту;

 – перейти до місця першої помилки;

 – знайти наступну помилку;

 – запустити майстер Application Builder для створення нового файлу;

 – запустити вбудований програматор.

Основне вікно програми складається із трьох дочірніх вікон: "Редактор", "Вікно проекту", "Вікно стану". У вікні "Редактора" відображається текст файлу, що редагується. У "Вікні проекту" відображається у вигляді дерева список включених у проект файлів (заголовних і файлів з кодом програми), список всіх функцій, список глобальних змінних, файл із позначками (Notes). Всі попередження й помилки з посиланнями на рядок, у якому вони виникають, з'являються у "Вікні стану". Між всіма елементами можна перемикатися за допомогою миші.

Як і більшість Сі-компіляторів, ImageCraft працює не з окремим Сі-файлом, а з проектом. Для створення нового проекту вибираємо Project → New, і у діалоговому вікні, що відрилося, вибираємо проект. Бажано кожен новий проект зберігати в окремому каталозі, тому що компілятор створює багато файлів. Після створення проекту, необхідно створити файл, у якому буде написана програма, а потім додати цей файл у проект. Для цього з меню File вибирається команда New. Потім бажано відразу файл зберегти, вибравши команду File → Save As... У діалоговому вікні слід ввести назву файлу, і обов'язково вказати розширення файлу «.c». Після цього натисканням правої кнопки миші на назві проекту у вікні проекту викликати контекстне меню, з якого вибрати команду «Add File(s)». Після цього можна писати програму.

Майстер Application Builder (рис. А.4) використовується для швидкого та зручного налаштування периферії та зовнішніх підключених пристроїв. Майстер дозволяє вибрати тип процесора (Target CPU), тактову частоту (Xtal Speed), режим роботи портів вводу-виводу (Ports), налаштувати роботу аналогового компаратора та АЦП (Analog), роботу SPI, налаштувати зовнішні переривання (External interrupts), роботу таймерів-лічильників (Timers), протокол UART. Зовнішній вигляд вікна залежить від типу обраного контролера та периферії, що використовується.

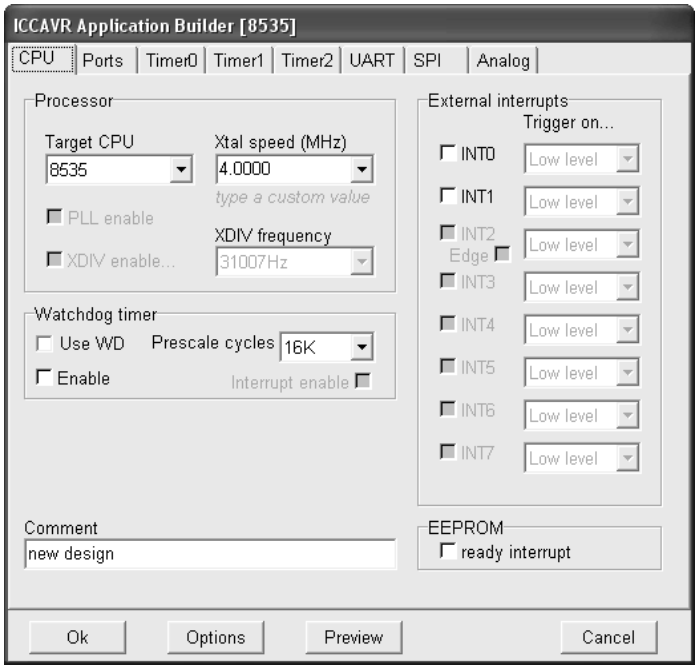


Рис. А.4.

Для перегляду коду програми можна натиснути на кнопку Preview. При використанні переривань, Application Builder автоматично генерує код прототипів функцій обробки переривань.

Перед написанням програми, властивості кожного проекту необхідно налаштувати. Для цього вибираємо Project → Options. З'явиться діалогове вікно, показане на рис. А.5. На першій вкладці (Paths) вказуються шляхи до бібліотек, включених в проект, для побудови мульти-файлового проекту; на третій вкладці (Target) вибираємо тип контролера, розмір стека. Налаштування на другій і четвертій закладках можна залишити без змін.

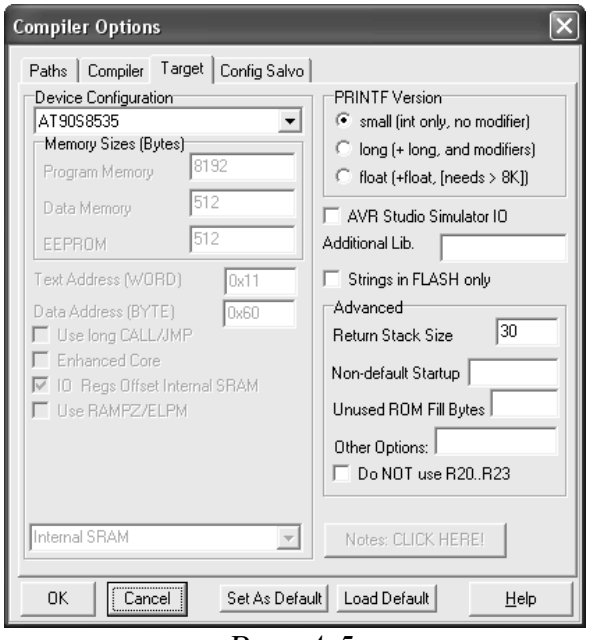


Рис. А.5.

Синтаксис мови

Синтаксис мови AVR Сі компілятора ImageCraft практично повністю відповідає міжнародному стандарту ANSI-C, але оскільки він призначений для роботи з мікроконтролерами, то має деякі особливості.

Коментарі в програмі

Всі символи, розташовані між /* та */, ігноруються компілятором, і цим можна вільно користуватися, щоб зробити програму більш зрозумілою. Коментар можна розташовувати в будь-якому місці, де можуть стояти символи пробілу, табуляції або символ нового рядка. Так само підтримуються коментарі стандарту мови С++, наприклад:

```
/* Це коментарі */
// Це коментарі
```

Зарезервовані слова

Компілятор резервує деякі слова, наприклад, такі як break, double, for, необхідні для керування програмою, що представляють собою ідентифікатори типів, деякі стандартні функції. У цей список входять як зарезервовані слова стандарту ANSI, так і специфічні слова, що відносяться тільки до мікроконтролерів, такі як eeprom, flash. Список всіх зарезервованих слів можна знайти в розділах допомоги.

Типи даних (ідентифікатори)

Імена складаються з букв латинського алфавіту (A...Z; a...z) і цифр (0...9); першим символом повинна бути буква. Символ підкреслення "_" вважається буквою; його іноді зручно використовувати, щоб поліпшити сприйняття довгих імен змінних. Не слід починати імена змінних з підкреслення, тому що багато змінних бібліотечних функцій починаються саме із цього знака. Великі (прописні) і малі (рядкові) букви розрізняються, так що x та X – це два різних ідентифікатори.

Для імен змінних значимими є 31 символ. Бажано давати змінним осмислені імена відповідно до їх призначення, причому такі, щоб їх було важко сплутати один з одним.

Для компілятора ImageCraft існує також список змінних, зарезервованих за периферією контролера, аналогічних відповідним регістрам асемблера. Список зарезервованих слів залежить від типу контролера й описується в його заголовному файлі.

Типи даних

Компілятор ImageCraft працює з наступними типами даних (табл.А.1):

Табл. А.1.

Тип	Розмір (Біт)	Діапазон
bit	1	0 or 1
char	8	-128 to 127
unsigned char	8	0 to 255
signed char	8	-128 to 127
int	16	-32768 to 32767
short int	16	-32768 to 32767
unsigned int	16	0 to 65535
signed int	16	-32768 to 32767
long int	32	-2147483648 to 2147483647
unsigned long int	32	0 to 4294967295
signed long int	32	-2147483648 to 2147483647
float	32	±1.175e-38 to ±3.402e38
double	32	±1.175e-38 to ±3.402e38

Константи

Константи можуть бути записані в десятковій формі (наприклад 123456), шістнадцятковій з використанням префікса 0x (наприклад 0x123abc або 0x123ABC), або у двійковій з використанням префікса 0b (наприклад 0b1010010).

Символьна константа є ціле, записане у вигляді символу, обрамленого одиночними лапками, наприклад 'k'. Строкові константи повинні записуватися в обрамленні подвійних лапок, наприклад "Hello world". Якщо текстовий рядок записується в лапках у вигляді параметра функції, то вона сприймається у вигляді строкової константи й міститься у flash пам'яті.

```
/* Ця функція показує рядок, розміщений SRAM */
void display_ram(char *s)
 {

 }

/ * Ця функція показує рядок, розміщений FLASH */
void display_flash(char flash *s)
{

}

void main(void)
{
/* Цей код працювати не буде, тому що функція адресує рядок, розміщений в
SRAM, а рядок "Hello world" - константа, що знаходиться у FLASH */
display_ram("Hello world");
```

```
/* Цей код буде працювати тому що функція адресує рядок, розміщений у FLASH
*/
display_flash("Hello world");
}
```

Приклад А.1. Написати програму, що виводить в порт (наприклад, PORTD) задане число $170_{10}=10101010_2=AA_{16}$.

Створюємо новий проект із назвою First. Для цього вибираємо Project → New і зберігаємо його в каталозі First з назвою First.prj. Майстром Application Builder користуватися не будемо, створюємо новий файл File → New і зберігаємо його в каталозі First з назвою first.c, додаємо сі-файл до проекту та вибираємо тип контролера, що використовується. У редакторі пишемо код програми:

```
#include <io8535v.h>          //підключаємо заголовний файл контролера
void main ()                  // основна функція
 {
  DDRD=0xFF;                  //установка напрямку роботи порту D на вихід
  PORTD=0b10101010;           //вивід на порт комбінації
 }
```

Перший рядок в програмі – підключення заголовного файлу, у якому описана структура та периферія контролера. Другий рядок – оголошення основної функції програми: DDRD – налаштування всіх виводів порту D на вихід; PORTD – вивід комбінації 10101010 на порт.

Змінні

Однією з основних переваг Сі компілятора, у порівнянні з асемблером, є використання у вихідному коді програми змінних. При цьому користувачеві немає необхідності стежити за розподілом регістрів загального призначення, FLASH та EEPROM пам'яті. Компілятор сам "стежить" за розподілом пам'яті, рятуючи програміста від цього завдання. Але при цьому можуть виникати проблеми, пов'язані з оптимальністю розподілу пам'яті та швидкодією програм, особливо при приведенні типів або обчисленнях із плаваючою крапкою.

Всі змінні повинні бути оголошені раніше, ніж будуть використовуватися. Оголошення передбачає вказівку типу і список з однієї або декількох змінних цього типу, наприклад:

```
int i, j, k;
char c;
```

При своєму оголошенні змінній може бути привласнене початкове значення, наприклад:

```
int i=0;
```

На відміну від стандарту ANSI, всі змінні повинні бути оголошені на початку блоку, у якому будуть використовуватися. Наприклад:

```
void main ()              // основна функція
{
  int i;                  // правильне оголошення
  DDRD=0xFF;              // установка напрямку роботи порту D на вихід
  int j, k;               // неправильне оголошення
  PORTD=0b10101010; // вивід на порт комбінації
}
```

Програмні змінні можуть бути глобальними (доступні для всіх функцій всієї програми) і локальними (доступні тільки всередині тіла функцій, де вони оголошені). Глобальні змінні оголошуються перед всіма функціями. Глобальним змінним при оголошенні автоматично встановлюється значення 0. Локальні змінні автоматично не ініціюються при виклику функцій.

До будь-якої змінної в оголошенні може бути застосований оператор const для вказівки того, що її значення далі не буде змінюватися.

```
const double e = 2.71828182845905;
const char msg[] = "Error number:";
```

Змінні можуть групуватися в масиви. Масив може мати розмір до 8 вимірів. Перший елемент масиву має індекс 0. Елементам глобально оголошених масивів автоматично привласнюється значення 0.

Стосовно до масиву ключове слово const вказує на те, що жоден з його елементів не буде змінюватися. Вказівку const можна також застосовувати до аргументу-масиву, щоб повідомити, що функція не змінює цей масив:

```
int strlen(const char[]);
```

При спробі змінити змінну, позначену ключовим словом const, компілятор повідомляє про помилку: "незмінне значення" (unmodifiable value).

Задання адреси змінних в області SRAM пам'яті

При використанні в програмі асемблера, часто доводиться знати адресу змінної, що розташована в області SRAM пам'яті. При кожній зміні програми, адреса змінної може змінюватися, тому можна жорстко вказати компілятору розташувати змінну по певній адресі.

Глобальні змінні можуть бути оголошені в області SRAM під час написання програми з використанням директиви `#pragma`. Наприклад:

```
#pragma abs_address:0x1000
unsigned LCD_control_register;
#pragma end_abs_address
```

Організація SRAM-пам'яті

Однією з основних переваг Сі-компіляторів у порівнянні з асемблером, є автоматичний розподіл регістрів, SRAM, FLASH та EEPROM пам'яті. При цьому різні типи змінних оголошуються в різних частинах пам'яті контролера. Оскільки контролер має досить обмежені ресурси, необхідно знати, як розподіляються змінні. Використання SRAM пам'яті контролера зазначене на рис. А.6.



Рис. А.6.

Working Registers (робочі регістри**)** – область пам'яті, що містить 32 8-бітних регістра загального призначення.

Компілятор використовує наступні регістри: R0, R1, R15, R22, R23, R24, R25, R26, R27, R28, R29, R30 та R31.

Також, деякі з регістрів від R2 до R14 можуть бути зарезервовані компілятором для глобальних змінних. Інші регістри з тих, що не використовуються у цьому діапазоні, резервуються для глобальних змінних типу char та int, а також для глобальних вказівників.

Регістри з R16 по R21 резервуються для локальних char та int змінних.

I/O Registers (регістри вводу/виводу) – діапазон, що містить 64 адреси для функцій периферійних пристроїв, таких як регістри керування напрямком роботи порту, регістри таймерів-лічильників та інші функції вводу-виводу. Можливо вільно використовувати ці регістри в асемблерних вставках.

Data Stack area (стек даних) використовується для динамічного зберігання локальних змінних, передачі параметрів функцій і зберігання

регістрів R0, R1, R15, R22, R23, R24, R25, R26, R27, R30, R31 та SREG під час обробки переривань.

Доступ до вказівника стека відбувається за допомогою Y-регістра.

При запуску вказівник стека ініціюється значенням 5Fh (FFh для деяких контролерів) + Розмір стека (Data Stack Size).

При налаштуванні компілятора, на вкладці Project → Options → Target → Return Stack Size, необхідно вказати достатній розмір стека, такий, щоб він не перекривав діапазон регістрів вводу-виводу при виконанні програми.

Global Variables (глобальні змінні) – область пам'яті, що використовується для статичного зберігання глобальних змінних. Розмір цієї області обчислюється як сума розмірів всіх оголошених глобальних змінних.

Hardware Stack (апаратний стек) – діапазон, що використовується для зберігання адреси повернення функцій.

Регістр вказівника стека (SP) використовується як вказівник стека та ініціюється при запуску зі значенням _HEAP_START_ – 1.

При виконанні програми апаратний стек (Hardware Stack) зміщується донизу, в область глобальних змінних (Global Variables).

HEAP ("купа") – вільна область пам'яті, розташована між апаратним стеком та SRAM. Ця пам'ять використовується для виділення функцій зі стандартної бібліотеки (Standard Library), таких як: malloc, calloc, realloc та free.

Список регістрів та адреси SRAM для ATmega603/103 наведений у табл. А.2.

Табл. А.2.

ПОЗНАЧЕННЯ	АДРЕСА SRAM	ФУНКЦІЯ
SREG	0x3F (0x5F)	Регістр статусу (Status REGister)
SPH	0x3E (0x5E)	Старший байт вказівника стека (Stack Pointer High)
SPL	0x3D (0x5D)	Молодший байт вказівника стека (Stack Pointer Low)
XDIV	0x3C (0x5C)	Регістр керування розподілом тактової частоти (XTAL Divide Control Register)
RAMPZ	0x3B 0x5B)	Регістр вибору сторінки Z RAM (RAM Page Z Select Register)
EICR	0x3A (0x5A)	Регістр керування зовнішніми перериваннями (External Interrupt Control Register)
EIMSK	0x39 (0x59)	Регістр масок зовнішніх переривань (External Interrupt MaSK register)
EIFR	0x38 (0x58)	Регістр прапорів зовнішніх переривань (External Interrupt Flag Register)
TIMSK	0x37 (0x57) Counter	Регістр масок переривань по таймерах/лічильниках (Timer/Interrupt MaSK register)
TIFR	0x36 (0x56)	Регістр прапорів переривань по таймерах/лічильниках (Timer/ Counter Interrupt Flag register)
MCUCR	0x35 (0x55)	Регістр керування MCU (MCU General Control Register)

MCUSR	0x34 (0x54)	Регістр статусу MCU (MCU Status Register)
TCCR0	0x33 (0x53)	Регістр керування таймером/лічильником 0 (Timer/Counter0 Control Register)
TCNT0	0x32 (0x52)	Таймер/лічильник0 (Timer/Counter0 (8-bit))
OCR0	0x31 (0x51)	Регістр порівняння виходу таймера/лічильника 0 (Timer/Counter0 Output Compare Register)
ASSR	0x30 (0x50)	Регістр статусу асинхронного режиму (Asynchronous Mode Status Register)
TCCR1A	0x2F (0x4F)	Керуючий регістр А таймера/лічильника 1 (Timer/Counter1 Control Register A)
TCCR1B	0x2E (0x4E)	Керуючий регістр В таймера/лічильника 1 (Timer/Counter1 Control Register A)
TCNT1H	0x2D (0x4D)	Старший байт таймера/лічильника 1 (Timer/Counter1 High Byte)
TCNT1L	0x2C (0x4C)	Молодший байт таймера/лічильника 1 (Timer/Counter1 Low Byte)
OCR1AH	0x2B (0x4B)	Старший байт регістра А порівняння виходу таймера/лічильника 1 (Timer/Counter1 Output Compare Register A High Byte)
OCR1AL	0x2A (0x4A)	Молодший байт регістра А порівняння виходу таймера/ лічильника 1 (Timer/Counter1 Output Compare Register A Low Byte)
OCR1BH	0x29 (0x49)	Старший байт регістра В порівняння виходу таймера/лічильника 1 (Timer/Counter1 Output Compare Register B High Byte)
OCR1BL	0x28 (0x48)	Молодший байт регістра В порівняння виходу таймера/лічильника 1 (Timer/Counter1 Output Compare Register B Low Byte)
ICR1H	0x27 (0x47)	Старший байт регістра захоплення таймера/лічильника 1 (Timer/Counter1 Input Capture Register High Byte)
ICR1L	0x26 (0x46)	Молодший байт регістра захоплення таймера/лічильника 1 (Timer/Counter1 Input Capture Register Low Byte)
TCCR2	0x25 (0x45)	Регістр керування таймером/лічильником 2 (Timer/Counter2 Control Register)
TCNT2	0x24 (0x44)	Таймер/лічильник 2 (Timer/Counter2 (8-bit))
OCR2	0x23 (0x43)	Регістр порівняння виходу таймера/лічильника 2 (Timer/Counter2 Output Compare Register)
WDTCR	0x21 (0x41)	Регістр керування сторожовим таймером (Watchdog Timer Control Register)
EEARH	0x1F (0x3F)	Старший байт регістра адреси EEPROM (EEPROM Address Register High)
EEARL	0x1E (0x3E)	Молодший байт регістра адреси EEPROM (EERPOM Address Register Low)
EEDR	0x1D (0x3D)	Регістр даних EEPROM (EEPROM Data Register)
EECR	0x1C (0x3C)	Регістр керування EEPROM (EEPROM Control Register)
PORTA	0x1B (0x3B)	Регістр даних порту А (Data Register, Port A)
DDRA	0x1A (0x3A)	Регістр напрямку даних порту А (Data Direction Register, Port A)
PINA	0x19 (0x39)	Виводи входів порту А (Input Pins, Port A)
PORTB	0x18 (0x38)	Регістр даних порту В (Data Register, Port B)
DDRB	0x17 (0x37)	Регістр напрямку даних порту В (Data Direction Register, Port B)
PINB	0x16 (0x36)	Виводи входів порту В (Input Pins, Port B)
PORTC	0x15 (0x35)	Регістр даних порту С (Data Register, Port C)
PORTD	0x12 (0x32)	Регістр даних порту D (Data Register, Port D)
DDRD	0x11 (0x31)	Регістр напрямку даних порту D (Data Direction Register, Port D)

PIND	0x10 (0x30)	Виводи входів порту D (Input Pins, Port D)
SPDR	0x0F (0x2F)	Регістр даних SPI I/O (SPI I/O Data Register)
SPSR	0x0E (0x2E)	Регістр статусу SPI (SPI Status Register)
SPCR	0x0D (0x2D)	Регістр керування SPI (SPI Control Register)
UDR	0x0C (0x2C)	Регістр даних UART I/O (UART I/O Data Register)
USR	0x0B (0x2B)	Регістр статусу UART (UART Status Register)
UCR	0x0A (0x2A)	Регістр керування UART (UART Control Register)
UBRR	0x09 (0x29)	Регістр керування швидкістю UART (UART Baud Rate Register)
ACSR	0x08 (0x28)	Регістр статусу й керування аналогового компаратора (Analog Comparator Control and Status Register)
ADMUX	0x07 (0x27)	Регістр вибору мультиплексора ADC (ADC Multi plexer Select Register)
ADCSR	0x06 (0x26)	Регістр статусу й керування ADC (ADC Control and Status Register)
ADCH	0x05 (0x25)	Старший байт регістра даних ADC (ADC Data Register High)
ADCL	0x04 (0x24)	Молодший байт регістра даних ADC (ADC Data Register Low)
PORTE	0x03 (0x23)	Регістр даних порту E (Data Register, Port E)
DDRE	0x02 (0x22)	Регістр напрямку даних порту E (Data Direction Register, Port E)
PINE	0x01 (0x21)	Виводи входів порту E (Input Pins, Port E)
PINF	0x00 (0x20)	Виводи входів порту F (Input Pins, Port F)

Математичні операції

Для обробки інформації, що надходить, мікроконтролер повинен виконувати різні математичні перетворення над даними. У табл. А.3 наведені основні математичні операнди компілятора. Якщо контролер не має можливості апаратного виконання функції, то функція виконується програмно.

Табл. А.3.

Операнд	Значення	Результат
+	Додавання	
-	Віднімання	
*	Множення	
/	Ділення	float
%	Ділення по модулю	
>>	Побітовий зсув праворуч	
<<	Побітовий зсув ліворуч	
|	Логічне (побітове) АБО	
&	Логічне (побітове) І	

Функції

Мова Сі заснована на понятті функції. Функція – це частина програми, що зберігається в певній області пам'яті і яку можна викликати необмежене число раз.

Будь-яка програма – це сукупність визначень змінних і функцій. Зв'язки між функціями здійснюються через аргументи, що повертають значення, й зовнішні змінні. У вихідному файлі функції можуть

розташовуватися в будь-якому порядку; вихідну програму можна розбивати на будь-яке число файлів, але так, щоб жодна з функцій не виявилася розрізаною.

Інструкція return реалізує механізм повернення результату від функції, що викликається, в основну програму. За ключовим словом `return` може розташовуватися будь-який вираз або змінна: `return` *вираз*;

Якщо буде потрібно, вираз буде приведено до типу даних, що повертає функція. Часто вираз беруть у дужки, але вони не обов'язкові.

Вираз в return може бути відсутнім, і тоді взагалі ніяке значення не буде повернуто в основну програму з функції. Керування повертається без результуючого значення також і у тому випадку, коли обчислення досягли "кінця" (тобто останньої закриваючої фігурної дужки функції).

Для використання функцій, необхідно використовувати прототипи. Це оголошення включає інформацію про параметри функцій.

Наприклад:

```
int alfa(char par1, int par2, long par3);
```

Дійсне визначення функції повинне бути описане в іншому місці, наприклад:

```
int alfa(char par1, int par2, long par3)
{
/* Тіло функції */
}
```

Параметри у функції передаються через стек (Data Stack). Функції повертають свої значення в регістри R30, R31, R22 та R23 (спочатку молодші розряди –LSB, потім старші – MSB).

Для створення функції, що обробляє переривання, використовується специфікатор #pragma interrupt_handler, після якого вказується джерело переривання (вектор). Функції, що обробляють переривання не повертають і не приймають ніяких змінних, тому мають тип **`void`**. Для кожного контролера існує список доступних переривань, описаних у його заголовному файлі. У табл. А.4 наведений список векторів переривань для контролера At90S8535.

Табл. А.4.

Вектор	**Адреса**	**Значення**
EXT_INT0	0x02	Зовнішнє int0
EXT_INT1	0x04	Зовнішнє int1
TIM2_COMP	0x12	Збіг при порівнянні таймера-лічильника 2
TIM2_OVF	0x	Переповнення таймера-лічильника 2
TIM1_CAPT	0x14	Захват таймера-лічильника 1
TIM1_COMPA	0x18	Збіг А при порівнянні таймера-лічильника 1

TIM1_COMPB	0x1A	Збіг B при порівнянні таймера-лічильника 1
TIM1_OVF	0x0C	Переповнення таймера-лічильника 1
TIM0_OVF	0x20	Переповнення таймера-лічильника 0
SPI_STC	0x22	Завершення пересилання SPI
UART_RXC	0x24	Прийом символу UART
UART_DRE	0x26	Спустошення регістра даних UART
UART_TXC	0x28	Передача символу UART
ADC_INT	0x2A	Завершення перетворення АЦП
EE_RDY	0x2C	Готовність АЦП
ANA_COMP	0x2E	Спрацьовування аналогового компаратора

Наприклад, обробка зовнішнього переривання int0:

```
#pragma interrupt_handler int0_isr:2
void int0_isr(void)
{
 //external interrupt on INT0
}
```

Використання асемблера в Сі програмах

Часто при написанні програм на Сі виникає необхідність використати асемблерні коди. Це може бути пов'язане з необхідністю оптимізації швидкості роботи або розміром програми, або можливо існує код, що виконує завдання і написаний на асемблері, і його можна просто вставити в текст програми. Асемблерний код може бути вставлений за допомогою команди:

```
#asm("asm_code");
```

Заголовочні файли

Для роботи з периферією, зовнішніми пристроями та для полегшення керування контролером, використаються заголовні файли. Заголовні файли можна включати в будь-якому порядку. Для включення заголовного файлу, використовується директива компілятора:

```
#include <заголовний файл>
```

Рядок #include не повинний бути всередині зовнішнього оголошення або визначення, і повинний зустрітися раніше, ніж що-небудь із заголовного файлу буде використано.

Для компілятора ImageCraft у заголовних файлах описуються тільки прототипи функцій, а сама реалізація функцій перебуває у файлах в каталозі lib, описана на асемблері.

ImageCraft містить наступні заголовні файли для роботи із зовнішніми пристроями:

Табл. А.5.

Бібліотека	Опис	Заголовний файл
Character Type Functions	Функції для роботи із символами	ctype.h
Standard C Input/Output Functions	Стандартні функції вводу-виводу	stdio.h
Standard Library Functions	Бібліотека стандартних функцій	stdlib.h,
Mathematical Functions	Математичні функції	math.h
String Functions	Функції для роботи з рядками	string.h

Опис функцій

Розглянемо деякі функції з тих, що найбільш часто використовуються. Заголовний файл «ctype.h» призначений для роботи з окремими символами, використовується при роботі з рядками, при обробці керуючих байтів пристроїв. Файл «ctype.h» містить наступні функції:

unsigned char isalnum(**char** c) – повертає 1 якщо змінна с алфавітно-цифровий символ;

unsigned char isalpha(**char** c) – повертає 1 якщо змінна с буквений символ;

unsigned char isascii(**char** c) – повертає 1 якщо змінна с ASCII символ (0..127);

unsigned char iscntrl(**char** c) – повертає 1 якщо змінна с керуючий символ(0..31 або 127);

unsigned char isdigit(**char** c) – повертає 1 якщо змінна с десяткова цифра (від 0 до 9);

unsigned char islower(**char** c) – повертає 1 якщо змінна с буквений символ нижнього регістра;

unsigned char isprint(**char** c) – повертає 1 якщо змінна с друкований символ (32..127);

unsigned char ispunct(**char** c) – повертає 1 якщо змінна с символ пунктуації (всі, крім керуючих і буквено-цифрових);

unsigned char isspace(**char** c) – повертає 1 якщо змінна с символ пробілу (пробіл, новий рядок, табулятор);

unsigned char isupper(**char** c) – повертає 1 якщо змінна с буквений символ у верхньому регістрі;

unsigned char isxdigit(**char** c) – повертає 1 якщо змінна с шістнадцяткове число;

char toascii(**char** c) – повертає ASCII еквівалент символу с;

unsigned char toint(**char** c) – інтерпретує с як шістнадцяткову цифру й повертає **unsigned char** від 0 до 15;

char tolower(**char** c) – повертає символ у нижньому регістрі, якщо с у верхньому регістрі, інакше – с;

char toupper(**char** c) – повертає символ у верхньому регістрі, якщо с у нижньому регістрі, інакше – с.

Заголовочний файл «stdio.h» містить функції вводу-виводу. Оскільки контролери не мають стандартного потоку вводу-виводу, функції заголовного файлу оптимізовані для роботи з обмеженими ресурсами. Файл «stdio.h» містить наступні функції:

char getchar(**void**) – повертає символ отриманий з UART, використовуючи послідовне опитування;

void putchar(**char** c) – передає символ в UART, використовуючи послідовне опитування;

void puts(**char** *str) – виводить (в UART), використовуючи putchar, символьний рядок str (null terminated), розташований в SRAM пам'яті, включаючи символ нового рядка '\n';

void putsf(**char** *str) – виводить (в UART), використовуючи putchar, символьний рядок str (null terminated), розташований у FLASH пам'яті, включаючи символ нового рядка '\n';

void printf(**char** *fmtstr [arg1, arg2, ...]) – виводить (в UART) форматований текст, використовуючи putchar, так само як він форматований у рядку fmtstr;

signed char sscanf(char *str, char flash *fmtstr [arg1 address, arg2 address, ...]) – ця функція аналогічна scanf, за винятком того, що форматуючий текст вводиться в рядку str, розташованому в SRAM.

Заголовний файл «math.h» містить опис більш ніж двадцяти математичних функцій. Тут наведені найбільш часто вживані. Кожна з функцій має один або два аргументи типу float і повертає результат також типу float:

sin(x) – повертає значення синуса *x*, *x* – у радіанах;
cos(x) – повертає значення косинуса *x*, *x* – у радіанах;
atan2(y,x) – повертає значення арктангенса відношення *y*/*x* у радіанах;
exp(x) – експонентна функція *e* у ступені *x*;
log(x) – натуральний логарифм *x* (*x*>0);
log10(x) – десятковий логарифм *x* (*x*>0);
pow(x,y) – *x* у ступені *y*;
sqrt(x) – корінь квадратний *x* (*x* > 0);
fabs(x) – абсолютне значення (модуль) *x*.

Компілятор ImageCraft містить також інші заголовні файли, які не розглянуті в даному розділі, тому що застосовуються досить рідко для специфічних пристроїв. Всі заголовні файли докладно описані в розділах вбудованої довідки.

Приклад А.2. Написати програму, що виводить на порт D інформацію, яка надходить на порт A мікроконтролера з інверсією.

Текст програми представлений нижче.

```
#include "io8535v.h"
 void main()
 {
 DDRD=0xff;                  // налаштування порту D на вихід
 DDRA=0x00;                  // налаштування порту A на вхід
 while(1)
  {
  PORTD=~PINA;               // вивід на порт D інформації з порту A
  }
}
```

У першому рядку програми виконується підключення заголовочного файлу, у якому зберігається інформація про адресний простір мікроконтролера і його ресурсів. Робота програми починається з функції main. У цій функції виконується налаштування порту D на вихід шляхом запису в регістр напрямку даних DDRD значення 0xFF. Порт A налаштовується на вхід. У циклі виконується зчитування значення з виводів порту A та передача цього значення в порт D.

Приклад А.3. Написати програму, що виводить на порт D добуток молодшого й старшого напівбайтів, що подаються на порт B.

```
#include <io8535v.h>
void main(void)              // головна функція програми
 {
  int i,j,rez;               // оголошення змінних
  PORTB=0;                   // вивід значення 0 в порт B
  DDRB=0;                    // налаштування порту B на вхід
  PORTD=0;                   // вивід значення 0 в порт D
  DDRD=0xff;                 // налаштування порту D на вихід
  while (1)                  // вічний цикл
     {
      i=PINB;                // запис значення з порту B у змінну
      j=i;                   // записати значення i в змінну j
      i=i&0b00001111;        // виділення 4-х молодших біт
      j=j&0b11110000;        // виділення 4-х старших біт
           j=j>>4;           // зсув 4-х старших біт у молодші
      rez=i*j;               // обчислення добутку
      PORTD=rez;             // вивід результату в порт
     };
 }
```

Приклад А.4. Програма для роботи з вбудованим АЦП AT90S8535 без переривань.

```
// В даному прикладі описані дві функції для роботи з АЦП:
// void InitADC(void) – ініціалізація АЦП
// int ReadADC(unsigned char) – зчитування значення
// напруги на заданому вході

 #include <io8535v.h>
  void InitADC(void)         // функція ініціалізації АЦП
 {
  ADMUX = 0;                 // вибрати вхід номер 0
  ADCSR = 0x0;               // включити АЦП і запустити перше
                             // "порожнє" перетворення
 }
 int ReadADC(unsigned char channel)
 {
  int i;
  ADMUX = channel;           // вибрати номер входу
  ADCSR |= 0x40;             // почати перетворення
  while (!(ADCSR & 0x10)); // перевірка завершення перетворення
                             // перетворення
  ADCSR |= 0x10;             // очищення біта "Перетворення
                             // завершене" шляхом запису в нього 1
  i = ADCL;                  // зчитування молодших 8 біт
  i=i+(int)ADCH<<8 ;         // зчитування старших 2 біт, множення
                             // їх на 256 і додавання з молодшим байтом
  return i;                  // повернення результату
 }
```

```
void main(void)              // головна функція програми
 {
 unsigned int temp;          // додаткова змінна
 InitADC();                  // ініціалізація АЦП
 temp=ReadADC(0);            // виміряти напругу на вході АЦП
 }
```

Завдяки вбудованим модулям підтримки широкого спектра периферійних пристроїв, процес створення програм істотно спрощується, що робить доступним застосування AVR мікроконтролерів навіть починаючим розробникам і студентам.

Вбудований автоматичний генератор програмного коду істотно підвищує продуктивність праці розробника та значно полегшує переведення програм з одного типу мікроконтролера на інший, оскільки звільняє програміста від турбот про внутрішню архітектуру процесора та адресації регістрів.

ДОДАТОК Б. ОПИС РОБОТИ З СИСТЕМОЮ МОДЕЛЮВАННЯ PROTEUS

Існує велика кількість програм для моделювання електронних пристроїв, які відрізняються між собою складністю інтерфейсу, наявністю бібліотек різних компонентів, точністю результатів моделювання. Одним з найпоширеніших пакетів є Electronic WorkBench, що має зручний інтерфейс, велику кількість бібліотек елементів. Однак істотне обмеження полягає в неможливості моделювання схем, в яких є мікропроцесори та програмовані логічні матриці. Аналогічне можна сказати й про систему MicroCap. Також істотним недоліком цих програм є неможливість створення друкованої плати для схеми, що моделюється. Система автоматизованого проектування PCAD призначена для розробки схем і друкованих плат, однак немає засобів моделювання.

САПР Proteus (рис. Б.1) дозволяє моделювати та налагоджувати складні пристрої, у яких може міститися декілька мікроконтролерів одночасно, причому різних сімейств. У бібліотеці є мікроконтролери AVR, 8051, Microchip PIC10, PIC16, PIC18, Philips ARM7, Motorola MC68HC11. Для моделювання мікроконтролера у вікні його параметрів потрібно вказати шлях до файлу прошивки формату «.hex», що може бути створений будь-яким компілятором (IAR, CodeVision, ICC, WinAVR), а також вказати тактову частоту. Крім того, є велика кількість різноманітних віртуальних приладів: осцилограф, логічний аналізатор, лічильник імпульсів, вимірник частоти, відлагоджувачі шин SPI та I^2C, термінал, генератор складних сигналів та інше. У генераторі сигналів можна вказати шлях до WAVE-файлу, інформація з якого буде використана під час моделювання. На виході цього блоку буде напруга, що відповідає даним у файлі. За допомогою цих інструментів можна одержати всю інформацію про роботу схеми. Дані й результати роботи віртуально створеного пристрою можуть бути збережені на комп'ютері тв доступні в цифровому вигляді для подальшого аналізу в пакетах Excel або MATLAB.

Істотною перевагою Proteus є можливість створення власних компонентів. Для цього на сайті виробника треба знайти SPICE-модель і додати її в потрібний корпус. SPICE-модель можна взяти з бібліотек інших пакетів моделювання. Наприклад, у системі PacDesigner фірми Lattice Semiconductors, що призначена для створення систем на базі програмованих аналогових інтегральних схем (ПАІС), є можливість створення SPICE-моделі розробленої системи на ПАІС. З підвищенням складності схеми, що моделюється, зростає різниця між часом моделювання та реальним часом.

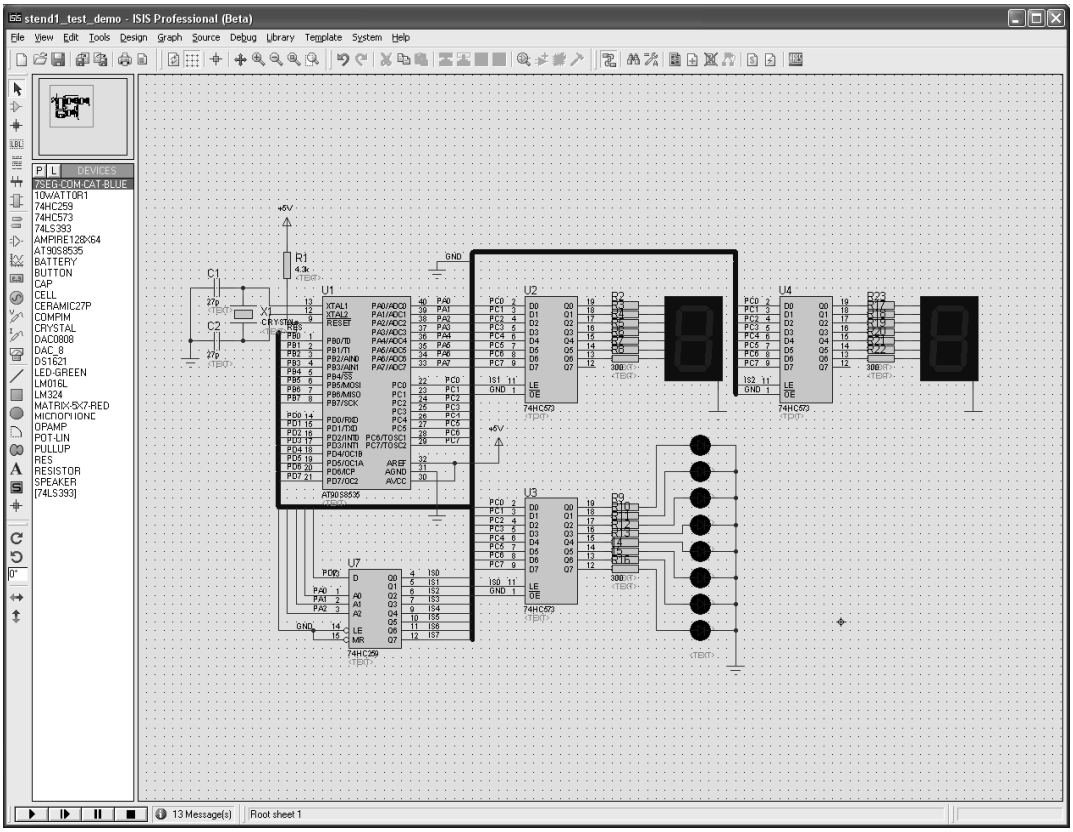


Рис. Б.1.

У лівій верхній області екрана можна побачити міні макет сторінки, а трохи нижче панель DEVICES – компоненти проекту (рис. Б.2). У цьому вікні відображаються всі елементи, що використовуються в схемі.



Рис. Б.2.

Кнопка з буквою "P" відкриває форму пошуку компонента в бібліотеках Proteus для додавання в схему. Кнопка з буквою "L" відкриває менеджер бібліотек – з його допомогою можна підключати нові бібліотеки компонентів.

У бібліотеці Proteus є моделі електромеханічних пристроїв: електродвигуни із квадратурним енкодером зворотного зв'язку, сервоприводи, крокові двигуни, безщітковий трьохобмотковий

електродвигун. У цих моделях можна встановлювати інерційність якоря з підключеними до нього механізмами, навантаження, опір й індуктивність пристрою, номінальна напруга живлення та швидкість обертів без навантаження. Моделі інтерактивних кнопок, перемикачів, ламп, світлодіодів, рідкокристалічних індикаторів, сенсорних панелей, змінних резисторів, дозволяють змінювати їхні параметри прямо в процесі роботи пристрою, що моделюється.

Для додавання компонента в схему необхідно натиснути на кнопку з буквою "P" . Відкриється діалогове вікно пошуку й вибору компонентів "Pick Devices", показане на рис. Б.3.

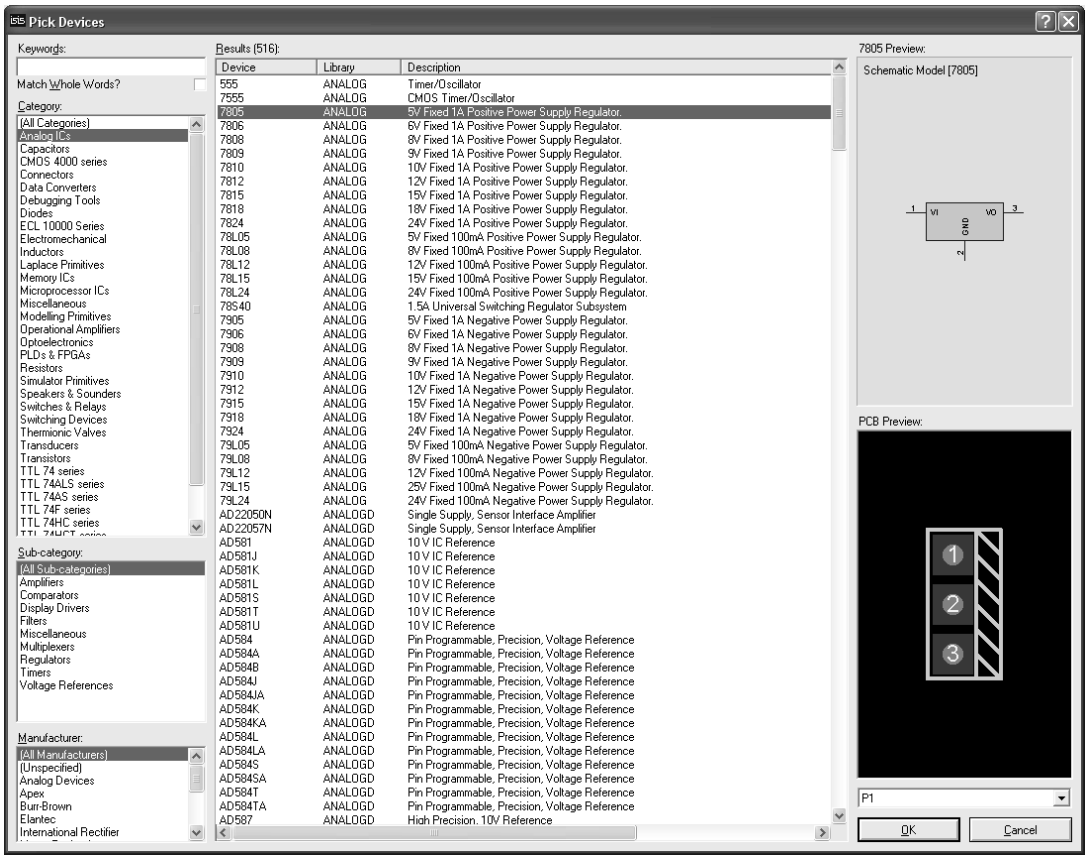


Рис. Б.3.

У полі "Keywords" можна ввести назву потрібного компонента, і якщо такий елемент є, то в правій частині діалогового вікна з'явиться його схематичне зображення, а також зображення посадкового місця для друкованої плати.

У полі "Category" відображається список груп електронних компонентів по їхньому призначенню: аналогові мікросхеми, перетворювачі даних, діоди, транзистори, елементи оптоелектроніки, мікропроцесори, лампи, реле й перемикачі, та інші. Виділивши окрему категорію, у поле "Results" з'явиться весь список доступних компонентів. У полі "Sub-category" можна виділити окремий клас елементів, що дозволить швидше знайти необхідний елемент. Наприклад, для категорії «Діоди» є

підкатегорії: мостові випрямлячі, діоди Шотки, стабілітрони, варикапи, тунельні діоди, випрямні діоди.

Помістити компонент на схему можна, натиснувши на кнопку "ОК", а потім перемістити курсор в потрібне місце на аркуші схеми і натиснути ліву кнопку миші. Компонент появиться на схемі.

Компонент на схему можна додати й так: виділити, скопіювати та вставити. Тільки в цьому випадку потрібно вручну привласнити йому порядковий номер замість знаку питання «?».

Виділити компонент або будь-який елемент схеми можна, натиснувши по ньому правою кнопкою миші. При цьому він стане червоного кольору й одночасно будуть виділені всі провідники, що підходять до компоненту – вони виділяться червоними кольорами до першого вузла. Скасувати виділення всіх виділених компонентів можна натисканням правої кнопки миші в порожньому місці схеми. Убрати компонент або будь-який елемент зі схеми можна двома натисканнями правої кнопки миші по ньому.

У табл. Б.1 приводиться опис кнопок панелі інструментів.

Табл. Б.1.

	L1	Відкрити панель DEVICES – компоненти проекту та пошук нових
	L2	Поставити вузол з'єднання провідників вручну
	L3	Дати назву провіднику – однойменні провідники з'єднані
	L4	Додати текст у довільне місце схеми
	L5	Прокласти шину
	L6	Створити підсхему
	L7	Натискання по компоненту відразу відкриває вікно редагування його властивостей
	L8	TERMINALS – живлення, земля, між блочні з'єднання, виводи
	L9	Додати вивід до створюваного компонента
	L10	Графічне відображення, збереження й аналіз результатів моделювання
	L11	Запис у файл і відтворення даних
	L12	Генератори напруг, струмів і вивід їх з файлів даних
	L13	Вказати точку виміру напруги на провіднику
	L14	Вказати точку виміру струму на провіднику
	L15	Virtual Instruments – вимірювальні прилади
	L16	Прокладка провідників на схемі, провідники можна називати за допомогою L3

Змінювати зображення виділеного компонента на схемі дозволяють кнопки:



– Повернути компонент на 90 градусів по годинній стрілці

– Повернути компонент на 90 градусів проти годинної стрілки

– Поле для введення кута повороту

– Дзеркальний переворот відносно вертикалі

– Дзеркальний переворот відносно горизонталі

Для керування процесом моделювання використовується панель, розташована в нижній частині головного вікна програми (рис. Б.4.).



Рис. Б.4.

Ця панель має 4 кнопки:

– "Пуск" – запуск симуляції або продовження припиненої симуляції;

– "Крок" – виконати мінімальний крок по програмі мікроконтролера (одна інструкція на асемблері). Цією кнопкою також можна почати симуляцію;

– "Пауза" – пауза симуляції. Можна продовжити кнопками "Пуск" або "Крок";

– "Стоп" – зупинка симуляції. Після цього симуляція почнеться спочатку кнопками "Пуск" або "Крок".

Створіть схему, показану на рис. Б.5.

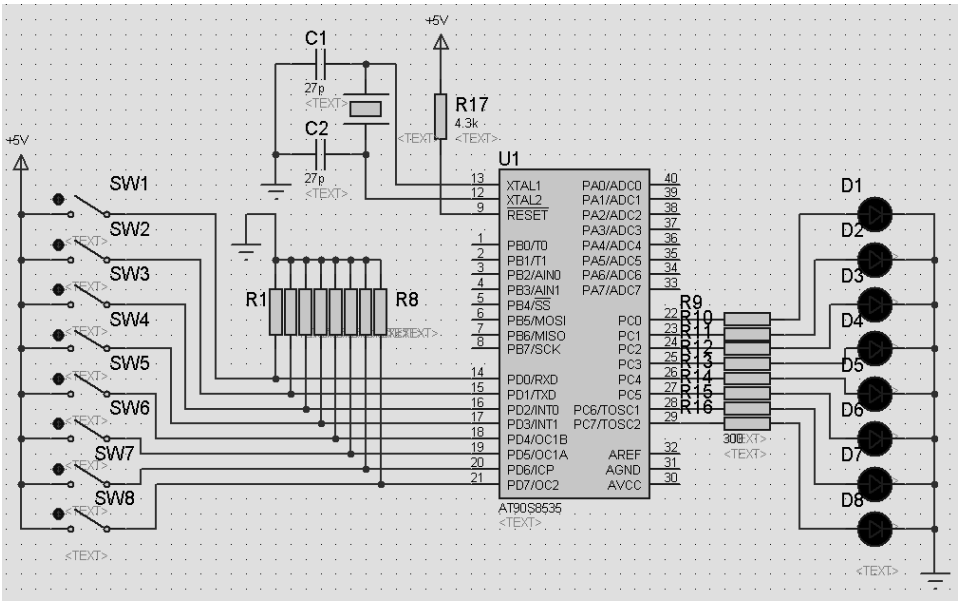


Рис. Б.5.

Виділить мікроконтролер натиснувши по ньому правою кнопкою миші. Мікроконтролер і підключені до нього провідники підсвітяться червоними кольорами. Відкрийте панель редагування властивостей компонента (Edit Component) натиснувши по виділеному мікроконтролеру лівою кнопкою миші. У діалоговому вікні, що з'явилося (рис. Б.6) натискання по кнопці "Hidden Pins" (сховані виводи) відкриє додаткове меню, у якому буде показано, як за замовчуванням називаються вузли (провідники) схеми до яких підключені живлення VCC і загальний провідник GND. Зміна цих назв може знадобитися при живленні мікроконтролера або інших компонентів різними напругами або від різних джерел.

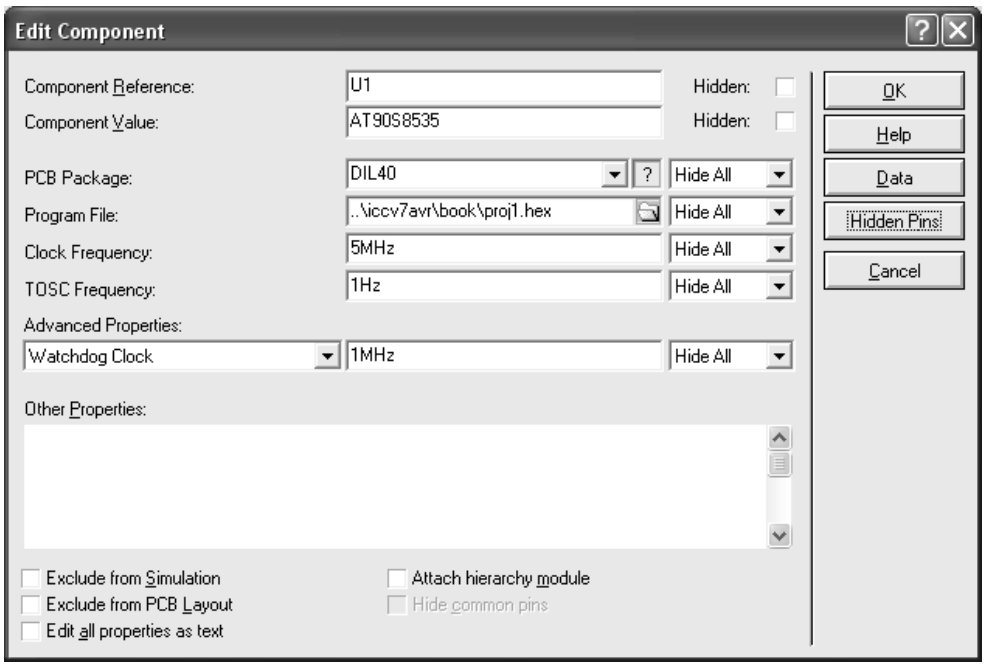


Рис. Б.6.

У полі "Program File" потрібно вказати шлях до файлу прошивки мікроконтролера:

«.cof» – файл якщо буде виконуватися налагодження програми, що написана на мові Сі;

«.hex» – файл прошивки.

У компіляторі ImageCraft необхідно створити новий проект, і відкомпілювати програму, текст якої представлений нижче:

```
#include "io8535v.h"        // підключення заголовочного файлу
void main()                 // початок програми
{
 char data;                 // оголошення змінної
 DDRC=0b11111111;           // налаштування порту С на вихід
```

```
 while(1)                    // вічний цикл
{
 data=PIND;                  // зчитування значення з порту D
 data=data+23;               // до зчитаного значення додається 23
 PORTC=data;                 // результат виводиться в порт C
 }
}
```

Наведена програма зчитує значення з порту D мікроконтролера, додає до цього значення число 23 і виводить результат у порт C.

Після компіляції проекту створюється файл із розширенням «.hex», шлях до якого необхідно вказати в діалоговому вікні налаштування властивостей мікроконтролера.

Також необхідно вказати частоту, на якій буде працювати мікроконтролер (поле Clock frequency). У наведеному прикладі частота дорівнює 5 МГц.

Щоб побачити всі налаштування компонента в панелі редагування властивостей, потрібно поставити галочку навпроти мітки «Edit All Properties As Text» (рис. Б.7).

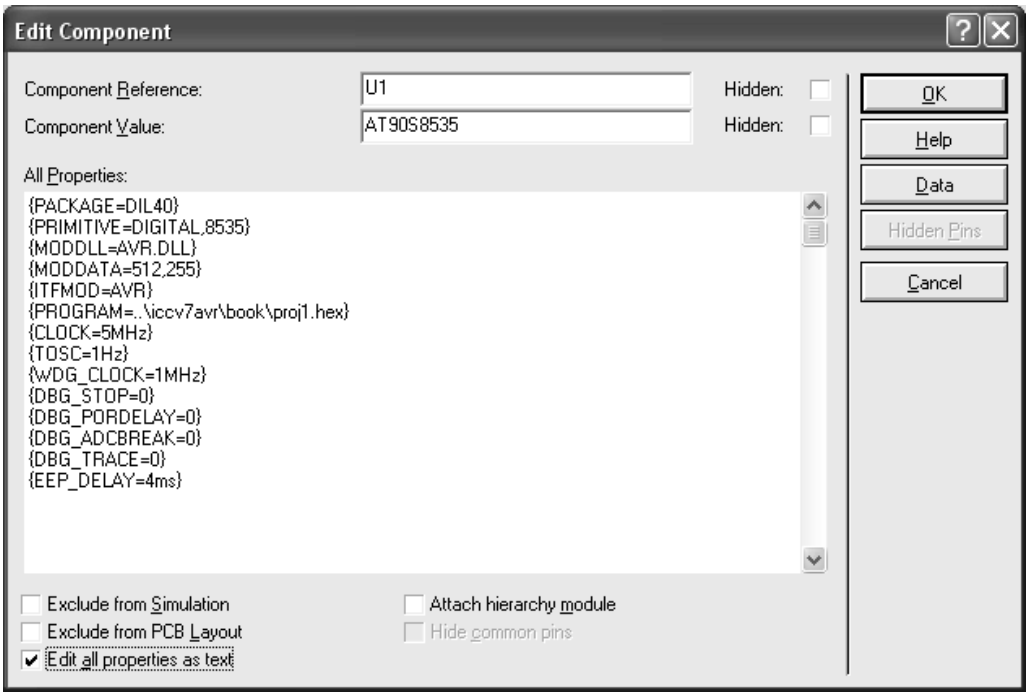


Рис. Б.7.

Запуск процесу моделювання виконується натисканням на кнопку "Пуск", що розташована на панелі керування процесом моделювання. Після цього на світлодіодах можна спостерігати стан виводів мікроконтролера. При натисканні на перемикачі, стан виводів буде змінюватися. У процесі моделювання біля кожного виводу мікроконтролера з'являється кольорова крапка. Якщо на ніжці мікросхеми

встановлений рівень логічного 0, крапка має синій колір. Якщо ж на виводі присутній сигнал високого рівня, то крапка буде червоного кольору.

В Proteus є можливість покрокового виконання програми мікроконтролера в режимі налагодження. Для запуску режиму налагодження з меню Debug (рис. Б.8) потрібно вибрати команду Start/Restart Debugging.

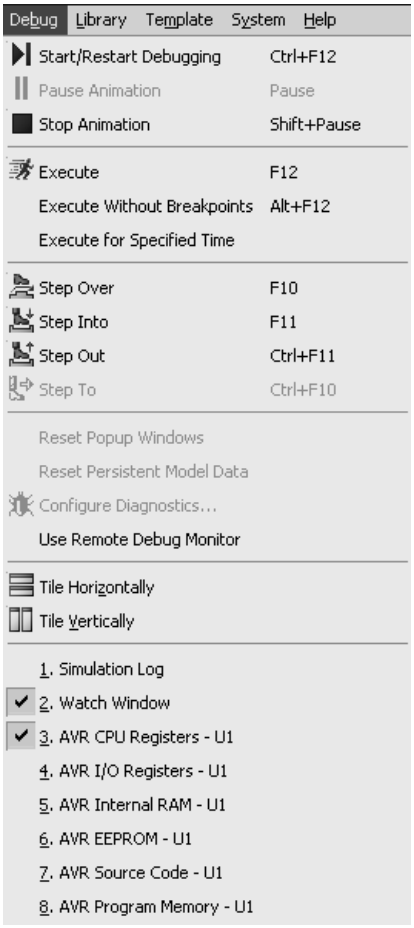


Рис. Б.8.

У нижній частині цього меню розташовується перелік вікон, у яких можна побачити інформацію про стан регістрів. На рис. Б.9 показане вікно спостереження за станами регістрів введення/виведення мікроконтролера.

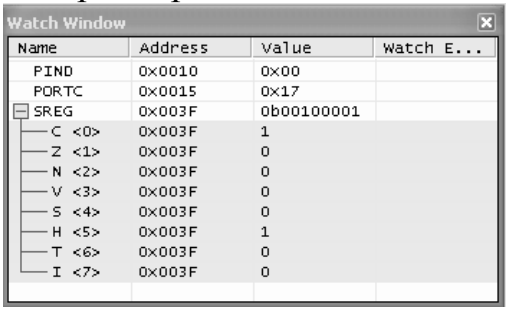


Рис. Б.9.

Для додавання регістра у вікно для спостереження за його станом, необхідно викликати контекстне меню в цьому вікні (рис. Б.10) шляхом натискання правої кнопки миші. Потім вибрати команду «Add Items By Name». У діалоговому вікні, що з'явилося, подвійне натискання лівої кнопки миші на назві регістра призведе до додавання його у вікно спостереження.

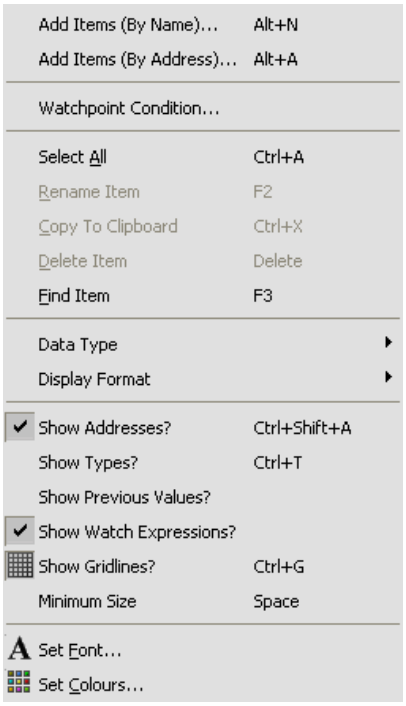


Рис. Б.10.

На рис. Б.11 представлене вікно станів внутрішніх регістрів мікроконтролера. У процесі покрокового налагодження стан регістрів буде змінюватися відповідно до алгоритму роботи програми, і ці зміни можна побачити в цьому вікні.

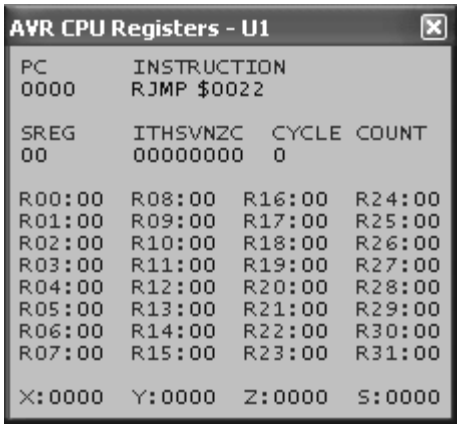


Рис. Б.11.

При натисканні клавіш F10 або F11 виконується перехід до наступної команди програми.

Перевагою пакета Proteus є те, що немає необхідності вносити зміни в розроблені програми для мікроконтролера при їх переносі на реальний пристрій. Система моделювання Proteus дозволяє редагувати та досліджувати програми без особливих матеріальних витрат, додаткових приладів і лабораторних стендів. Найбільша перевага – це додаткові елементи, які дозволяють моделювати роботу мікроконтролера зі світлодіодними індикаторами, рідкокристалічними дисплеями, клавіатурою, кроковим двигуном, датчиками з різними інтерфейсами, програмованою логікою, іншими мікроконтролерами ті зовнішніми пристроями.

ПЕРЕЛІК ЛІТЕРАТУРИ

1. Пароль Н. В. Знакосинтезирующие индикаторы и их применение: Справочник / Н. В. Пароль, С. А. Кайдалов. — М. : Радио и связь, 1989. — 128 с. : ил.
2. Мухитдинов М. Светоизлучающие диоды и их применение / М. Мухитдинов, Э. С. Мусаев. — М. : Радио и связь, 1988. — 80 с.
3. Вуколов Н. И. Знакосинтезирующие индикаторы. Справочник / Н. И. Вуколов, А. Н. Михайлов. — М. : Радио и связь, 1987. — 576 с.
4. Ермаков О. Н., Сушков В. П. Полупроводниковые знакосинтезирующие индикаторы / О. Н. Ермаков, В. П. Сушков. — М. : Радио и связь, 1990. — 238 с.
5. Седов С. А. Индивидуальные видеосредства. Справочное пособие / С. А. Седов. — К. : Наукова думка, 1990. — 752 с. : ил.
6. Самохвалов М. К. Конструкции и технология тонкопленочных электролюминесцентных индикаторов / М. К. Самохвалов. Ульяновск, УлГТУ, 1997. — 56 с.
7. Васерин Н. Н. и др. Применение полупроводниковых индикаторов / Н. Н. Васерин, Н. К. Дадерко, Г. А. Прокофьев; Под ред. Е. С. Липина. — М. : Энергоатомиздат 1991. — 200 с. : ил.
8. Бойко В. И. и др. Схемотехника электронных систем. Микропроцессоры и микроконтроллеры / В. И. Бойко. — СПб. : БХВ-Петербург, 2004. — 464 с.
9. Вайда З. Современная видеозапись: Перевод с венг. / З. Вайда. — М. : Радио и связь. 1987. — 304 с.
10. Самохвалов М. К. Элементы и устройства оптоэлектроники: Учебное пособие / М. К. Самохвалов. Ульяновск: УлГТУ, 2003. — 125 с.
11. Коган Л. М. Полупроводниковые светоизлучающие диоды / Л. М. Коган. — М. 1983.
12. Моделирование сложных систем и виртуальная реальность. Сборник «Вопросы кибернетики / Под ред. Ю. М. Баяковского и А. Н. Томилина. — М. : Изд-во РАН, 1995. №181. — 212 с.
13. "Электроныые компоненты" №7 2003г.
14. "Электронные компоненты" №6 2003г.
15. «Схемотехника» №7(9) июль 2001 с.20
16. http://www.espotec.ru/articles/lcd/art_lcd.htm
17. http://www.sibsutis.ru/~mavr/digital/LCD.htm
18. www.melt.aha.ru Жидкокристаллический модуль MTG128-64BFGHSGY
19. Сопряжение AVR-микроконтроллеров и ЖКИ // Режим доступа : www.almel.ru

20. Shinomiya T., Kuwnbatu M., et. al. A 40-in (1-m) diagonal direct-view TFT-LCD by seamless-connection technique // J. of the Society for information display. — 1997. — Vol. 5, N 4. — P. 363-365.
21. Дедов В. П. Архитектура модульных плазменных экранов / В. П. Дедов, А. А. Михаилов // В сб. "Устройства отображения информации на газоразрядных индикаторах в системах принятия решений" (Тезисы докладов межотраслевых научных конференций, совещаний, семинаров). — ВИМИ, 1988. — С. 6-9.
22. Buzcih T. Recent advances in PALC technology / T. Buzcih. The 18-th international display research conference "Asia display'98", — 1998.
23. Оптоэлектронные преобразователи и устройства отображения информации: Сб. науч. тр. — М. : МАИ, 1983. — 83 с.
24. Садовников В. И. Бесконтактный оптоэлектронный переключатель / В. И. Садовников // Оптоэлектронные преобразователи и устройства отображения информации. — М. : МАИ, 1983. — С. 22-27.
25. Оптоэлектронные преобразователи на основе управляемых светодиодных структур / В. И. Бусурин, В. Ф. Лярский, В. И. Садовников, Н. П. Удалов. — М. : Радио и связь, 1984. — 72 с.
26. Носов Ю. Р. Оптроны и их применение / Ю. Р. Носов, А. С. Сидоров. — М. : Радио и связь, 1981. — 280 с.
27. Птачек М. Цифровое телевидение: Теория и техника. / М. Птачек. Пер. с чеш. под ред. Л. С. Виленчика. — М. : Радио и связь, 1992. — 528 с.
28. Хохлов Б. Н. Декодирующие устройства цветных телевизоров / Б. Н. Хохлов. — М. : Радио и связь, 1998. — 512 с.
29. Новаковский С. В. Новые системы телевидения. Цифровые методы обработки видеосигналов / С. В. Новаковский, А. В. Котельников. — М. : Радио и связь, 1992. — 88 с.
30. Справочные материалы фирмы Philips (www.semiconductors.philips.com).
31. Носов Ю. В. Оптоэлектроника / Ю. В. Носов. — М.: Радио и связь, 1989. — 360 с.
32. Верещагин И. К. Введение в оптоэлектронику / И. К. Верещагин, Л. А. Косяченко, С. М. Кокин. — М. : Высшая школа, 1991. — 189с.
33. Мухин Ю. А. Приборы и устройства полупроводниковой электроники / Ю. А. Мухин. — М. : МЭИ, 1996. — 234 с.
34. Самохвалов М. К., Оптоэлектроника. Сборник лабораторных работ / М. К. Самохвалов. Ульяновск, УлГТУ, 1998. — 39 с.
35. Иванов А. Б. Волоконная оптика : компоненты, системы передачи, измерения / А. Б. Иванов. — М. : САЙРУС СИСТЕМС, 1999. — 671 с.
36. Пароль Н. В. Фоточувствительные приборы и их применение / Н. В. Пароль, С. А. Кайдалов. — М. : Радио и связь, 1991. — 112 с.
37. Раушенбах Г. Справочник по проектированию солнечных батарей / Г. Раушенбах. — М. : Энергоатомиздат, 1983. — 360 с.

38. Вуколов Н. И., Знакосинтезирующие индикаторы. Справочник / Н. И. Вуколов, А. Н. Михайлов. — М. : Радио и связь, 1987. — 576 с.
39. http://uk.wikipedia.org/wiki/Колір
40. http://uk.wikipedia.org/wiki/Коло_кольорів
41. http://uk.wikipedia.org/wiki/RGB
42. http://uk.wikipedia.org/wiki/CMYK
43. Бочаров Л. Н. Эквивалентные схемы и параметры полупроводниковых приборов / Л. Н. Бочаров. — М. : «Энергия», 1973. — 96 с.: ил.

ЗМІСТ

Вступ 3
Умовні скорочення 4
1. КЛАСИФІКАЦІЯ ЗАСОБІВ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ. ОСНОВНІ ВИЗНАЧЕННЯ 5
1.1. Основні параметри індикаторів 5
1.2. Фізичні ефекти, що використовуються в ЗВІ 10
1.3. Структура спектра випромінюваного світла 11
1.3.1. Коло кольорів 14
1.3.2. Оцифровування зображень 17
1.3.3. Кольорові моделі 19
1.4. Класифікація засобів відображення інформації 21
1.5. Ергономічні характеристики ЗВІ 23
1.6. Інформаційна модель і формування її елементів 24
Питання до розділу 1 27

2. НАПІВПРОВІДНИКОВІ ІНДИКАТОРИ 28
2.1. Класифікація напівпровідникових індикаторів 28
2.2. Фізичні основи формування оптичного випромінювання 29
2.3. Типи світловипромінювальних діодів 32
2.3.1. Одиничні світлодіоди 33
2.3.2. SMD-світлодіоди 38
2.3.3. Ультраяскраві світлодіоди 39
2.3.4. Лазерні світлодіоди 41
2.3.5. Багатокольорові світлодіоди 42
2.3.6. RGB-світлодіоди 44
2.3.7. Світлодіодні кластери 45
2.4. Режими живлення і схеми включення світлодіодів 47
2.5. Пристрої зі світлодіодами 50
2.6. Багатосегментні індикатори 54
2.6.1. Статичне управління 60
2.6.2. Мультиплексне управління 76
2.7. Матричні індикатори 82
2.8. Системи адресації 93
2.9. Шкальні індикатори 95
2.10. Світлодіодні дюралайти 102
2.11. Світлодіодні лампи 103
2.12. Способи управління яскравістю індикаторів 104
2.12.1. Параметричне управління 105
2.12.2. Часове управління 106
2.13. Драйвери напівпровідникових індикаторів 109
2.14. Принцип роботи біжучого рядка 123
2.15. Системи відображення інформації з механічною розгорткою 133

Питання до розділу 2 139
Задачі до розділу 2 140

3. БУДОВА І ПРИНЦИПИ РОБОТИ РІЗНИХ ТИПІВ ІНДИКАТОРІВ 141
3.1. Електролюмінесцентні індикатори 141
3.1.1. ЕЛІ постійного струму 141
3.1.2. Тонкоплівкові ЕЛІ 143
3.1.3. Сегментні ЕЛІ 148
3.2. Вакуумні люмінесцентні індикатори 150
3.2.1. Статичне управління ВЛІ 155
3.2.2. Мультиплексне управління ВЛІ 156
3.3. Вакуумні індикатори розрарювання 159
3.4. Газорозрядні індикатори 161
3.5. Електрохромні індикатори 164
3.6. Електрофоретичні індикатори 166
Питання до розділу 3 168

4. РІДКОКРИСТАЛІЧНІ ІНДИКАТОРИ 169
4.1. Принцип роботи і конструктивні модифікації 169
4.1.1. Рефлективні РКІ 170
4.1.2. Трансмісивні РКІ 171
4.1.3. Трансрефлективні РКІ 172
4.1.4. Бістабільні РКІ 172
4.1.5. Позитивні і негативні РКІ 176
4.2. Параметри рідкокристалічних індикаторів 177
4.3. Багатосегментні і матричні РКІ 179
4.4. Алфавітно-цифрові РКІ 183
4.4.1. Сполучення РКІ з мікроконтролером 185
4.4.2. Система команд РКІ на основі контролера HD44780 189
4.4.3. Принцип формування символів 195
4.4.4. Програмування і управління багаторядковими РКІ 213
4.5. Графічні РКІ 219
4.5.1. Сполучення РКІ з мікроконтролером 220
4.5.2. Структура адресного простору модуля MTG-12864 222
4.5.3. Система команд модуля MTG-12864 223
4.5.4. Формування текстової інформації 224
4.5.5. Формування графічної інформації 225
4.5.6. Управління РК-індикатором LM234 243
4.6. Управління індикаторами з інтерфейсом RS232 250
4.7. Моделювання роботи РКІ 252
4.8. Формування кольорового зображення 254
4.9. Рідкокристалічні монітори 255
Питання до розділу 4 257

Задачі до розділу 4 257

5. ПРИНЦИПИ ПОБУДОВИ ПРОЕКЦІЙНИХ СИСТЕМ 259
5.1. Джерела світла в проекційних системах 259
5.2. Основні типи модуляторів 260
5.2.1. Кінескопні проектори 260
5.2.2. Проектори з рідкокристалічним модулятором 261
5.2.3. Проектори на базі DMD-матриць 262
5.2.4. Лазерні проектори 263
5.3. Формування кольорового зображення 266
5.4. Оптичні властивості екранів з фронтальною та rear–проекціями 270
5.5. Похибки оптичних систем 272
5.6. Огляд відеопроекторів різних фірм-виробників 273
Питання до розділу 5 274

6. ПРИНЦИПИ ПОБУДОВИ СЕНСОРНИХ ПАНЕЛЕЙ 275
6.1. Типи сенсорних панелей 276
6.2. Дискретні і аналогові типи сенсорних панелей 278
6.3. Конструкція і технологія резистивних сенсорних панелей 280
6.4. Типи топологічних схем аналогових сенсорних панелей 281
6.4.1. Чотирьохпровідна схема 282
6.4.2. П’ятипровідна схема 283
6.5. Драйвери сенсорних панелей 284
Питання до розділу 6 286

7. ЕКРАНИ 287
7.1. Принципи побудови світлодіодних екранів 287
7.2. Принципи побудови великих плазменних екранів 292
7.3. Електролюмінесцентні дисплеї 301
7.4. Рідкокристалічні екрани 303
7.5. Технологія гнучких дисплеїв 305
7.6. Стереоскопічні системи 308
7.7. Системи з 3D-розгорткою 311
Питання до розділу 7 312

8. ТЕЛЕВІЗІЙНІ ЗАСОБИ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ 313
8.1. Будова електронно-променевої трубки 313
8.1.1. Монохромні ЕПТ 313
8.1.2. Кольорові ЕПТ 315
8.2. Види масок, що використовуютьсяе в ЕПТ 316
8.3. Типи спотворень, що виникають в ЕПТ 318
8.4. Програмні засоби тестування моніторів 321
8.5. Принципи організації розгорток в ЗВІ 322

8.6. Алфавітно-цифрові дисплеї растрового типу 324
8.7. Графічні ЗВІ телевізійного типу 326
8.8. Виведення зображення на екран телевізора 327
8.9. Телетекст 335
8.10. Опис порту VGA 343
8.11. Системи телебачення SECAM, PAL, NTSC 347
8.12. Порівняння систем SECAM, NTSC і PAL 350
8.13. Цифрове телебачення 353
8.14. Основні вузли і блоки пристроїв цифрового телебачення 356
Питання до розділу 8 367

9. ЕЛЕМЕНТИ І ПРИСТРОЇ ОПТОЕЛЕКТРОНІКИ 368
9.1. Основні характеристики і параметри фотоприймачів 368
9.1.1. Фоторезистори 370
9.1.2. Фотодіоди 375
9.1.3. Фототранзистори 384
9.1.4. Фототиристори 385
9.1.5. Фоточутливі прилади із зарядовим зв'язком 385
9.2. Принципи роботи і різновиди оптичних зчитувачів (сканерів) 389
9.3. Сонячні батареї 392
9.4. Основні параметри оптронів 393
9.4.1. Резисторні оптрони 396
9.4.2. Діодні оптопари 397
9.4.3. Транзисторні оптрони 398
9.4.4. Тиристорні оптрони 400
9.4.5. Різновиди оптронів (диференціальні, регенеративні) 400
9.5. Пристрої на світловипромінювальних діодах та оптронах 401
9.5.1. Пристрої індикації, перемикання і захисту 405
9.5.2. Пристрої контролю та вимірювання на світловипромінюючих діодах 405
9.5.3. Імпульсні та обчислювальні пристрої 410
9.6. Оптичні запам'ятовуючі пристрої (CD-ROM, DVD-ROM) 413
9.7. Розповсюдження світла у світлодіодах 417
9.8. Реализація волоконно-оптичних ліній зв'язку 422
Питання до розділу 9 424

Додаток А. Програмування мікроконтролерів AVR на мові С 425
Додаток Б. Опис роботи з системою моделювання Proteus 447

ПЕРЕЛІК ЛІТЕРАТУРИ 457

Навчальне видання

РЯБЕНЬКИЙ
Володимир Михайлович

СХЕМОТЕХНІКА ЕЛЕКТРОННИХ ПРИСТРОЇВ ТА СИСТЕМ

Підручник

У ШЕСТИ ТОМАХ

ТОМ 6

РЯБЕНЬКИЙ Володимир Михайлович
УШКАРЕНКО Олександр Олегович

АПАРАТНО-ПРОГРАМНІ ЗАСОБИ ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ

Рисунки – *Я. Дубовик*
Коректура та комп'ютерна верстка – *О. Матвєєва*

Формат 60x84$^{1}/_{16}$. *Ум. друк. арк.* 26,9. *Тираж* 300 пр. *Зам*. № 534-067.

В И Д А В Е Ц Ь І В И Г О Т О В Л Ю В А Ч

Товариство з обмеженою відповідальністю фірма «Іліон».
54001, м. Миколаїв, вул. Садова, 1.
Свідоцтво суб'єкта видавничої справи ДК № 1506 від 25.09.2003 р.