

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
імені ВОЛОДИМИРА ДАЛЯ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з дисципліни

«МІКРОПРОЦЕСОРНА ТЕХНІКА»

*(для здобувачів вищої освіти спеціальності 141 «Електроенергетика,
електротехніка та електромеханіка»)*

(Електронне видання)

ЗАТВЕРДЖЕНО
на засіданні кафедри
електричної інженерії
Протокол №7 від 17.02.2021 р.

Сєверодонецьк 2021

Методичні вказівки до виконання лабораторних робіт з дисципліни «Мікропроцесорна техніка» (для здобувачів вищої освіти спеціальності 141 «Електроенергетика, електротехніка та електромеханіка») (Електронне видання) / Уклад.: Г.С. Белоха, Є.С. Руднєв, Р.М. Брошко. – Сєверодонецьк: вид-во СНУ ім. В. Даля, 2021. – 53 с.

Методичне видання спрямоване на вивчення та засвоєння здобувачами вищої освіти програмування мікроконтролерів на навчальних стендах EV8031/AVR . Кожна лабораторна робота містить короткий теоретичний опис, мету, порядок виконання роботи, вимоги до її оформлення і контрольні питання.

Методичні матеріали розраховані на здобувачів ВО закладів вищої освіти.

Укладачі:

Г.С. Белоха, к.т.н.
Є.С. Руднєв, к.т.н., доц.
Р.М. Брошко, к.т.н.

Рецензент:

І.С. Шевченко, к.т.н, проф

ЗМІСТ

ВСТУП.....	4
ВИМОГИ ДО ВИКОНАННЯ ТА ОФОРМЛЕННЯ ЛАБОРАТОРНИХ РОБІТ	5
ЛАБОРАТОРНА РОБОТА №1. Вивчення навчального стенда EV8031/AVR.....	6
ЛАБОРАТОРНА РОБОТА №2. Система команд МК51.....	24
ЛАБОРАТОРНА РОБОТА №3. Відображення інформації світлодіодними індикаторами.....	30
ЛАБОРАТОРНА РОБОТА №4. Відображення інформації знаковинтезуючим індикатором.....	32
ЛАБОРАТОРНА РОБОТА №5. Відображення інформації на рідкокристалічному індикаторі.....	35
ЛАБОРАТОРНА РОБОТА №6. Формування аналогових сигналів на основі ЦАП.....	44
ЛАБОРАТОРНА РОБОТА №7. Аналого-цифрове перетворення в МПС.....	50
СПИСОК ЛІТЕРАТУРИ	52
Додаток А.....	53

ВСТУП

Навчальний стенд "EV8031 / AVR" - програмно-апаратний комплекс, орієнтований для застосування в навчальних цілях з дисципліни «Мікропроцесорна техніка», а також як засіб розробки програмного забезпечення для контролерів серії МК51 і контролерів архітектури AVR.

Для підготовки програмного забезпечення мікроконтролерів МК51 використовуються в основному мови "ASM-51", "C", для яких існують цілий ряд компіляторів, бібліотек стандартних підпрограм і програмних емуляторів, вироблених різними закордонними та вітчизняними фірмами.

2. ВИМОГИ ДО ВИКОНАННЯ ТА ОФОРМЛЕННЯ ЛАБОРАТОРНИХ РОБІТ

При підготовці до лабораторної роботи студент повинен підготувати бланк звіту, який наведено у додатку А в якому будуть міститися дані студента і назва і номер лабораторної роботи, а так же все те що представлено в розділі «Зміст звіту» для конкретної лабораторної роботи, крім програм та результатів їх виконання.

Проведення досліджень в лабораторній роботі зводиться до розроблення програм та налагодження їх, та отримання осцилограм різних сигналів досліджуваного інтерфейсу.

Результати роботи налагоджених програм індивідуальних завдань необхідно продемонструвати викладачеві. Після завершення лабораторної роботи, студенти повинні записати файли цих програм.

Оформлення програм проводиться відповідно до пункту «Зміст звіту» і вказівками викладача. Програми повинні мати коментарі, що відображають послідовність виконання алгоритму рішення завдання.

Звіт з лабораторної роботи повинен бути оформлений безпосередньо після її виконання і зданий на перевірку викладачеві. Захист звіту вимагає правильних відповідей на контрольні питання, наведені в кінці кожної лабораторної роботи.

Пропущені лабораторні роботи необхідно відпрацьовувати під час консультацій, не допускаючи накопичення боргів.

ЛАБОРАТОРНА РОБОТА №1

Вивчення навчального стенда EV8031/AVR

Мета роботи: вивчити структурну схему, функціональні можливості і порядок роботи зі стендом.

1. Теоретична частина

Технічні характеристики навчально - налагоджувального стенда EV8031/AVR:

1. Для цілей однокристальних мікроконтролери в корпусі DIP-40: AT89C51, AT89C52, AT90S8515 (ATmega8515);
2. Пам'ять програм – 16 КБайт; пам'ять даних – 16 КБайт;
3. Послідовна EEPROM пам'ять, 256 байт (AT24C02);
5. Два послідовних каналу передачі даних RS-232C;
6. Системний інтерфейс;
7. Інтерфейс розширення (16 ліній вихід, 8 ліній вхід / вихід, порт P1 OEBM);
8. Клавіатура 4x3;
9. Статична 4-х розрядна 7 - ми сегментна індикація;
10. Цифроаналоговий і аналого-цифровий перетворювач (плата розширення);
11. Генератор з фіксованою частотою генерації - близько 10 кГц, генератор із змінною частотою генерації від 0,9 кГц до 20 кГц (плата розширення);
12. Динамічна 4-х розрядна 7 - ми сегментна індикація (плата розширення);
13. Пристрій дискретного введення інформації: 2 кнопки;
14. Лінійка з 8 - ми світлодіодів;
15. Знакосинтезуючий світлодіодний індикатор 5 x 7, 1 шт. (Плата розширення).

1.1 Будова стенда

Структурна схема стенду EV8031 / AVR представлена на рис. 1.1.

Тут прийняті наступні позначення:

ОЭВМ - однокристальна ЭВМ (мікроконтролер);

ОЗУ 32К - пам'ять контролера (програм і даних);

RS-232C / 1, інтерфейсні роз'єми для зв'язку з ПК;

RS-232C / 2 - інтерфейсні роз'єми для зв'язку з іншим стендом;

PORTA, PORTB, PORTC - паралельні порти для організації введення / виведення даних з периферійними пристроями стенду.

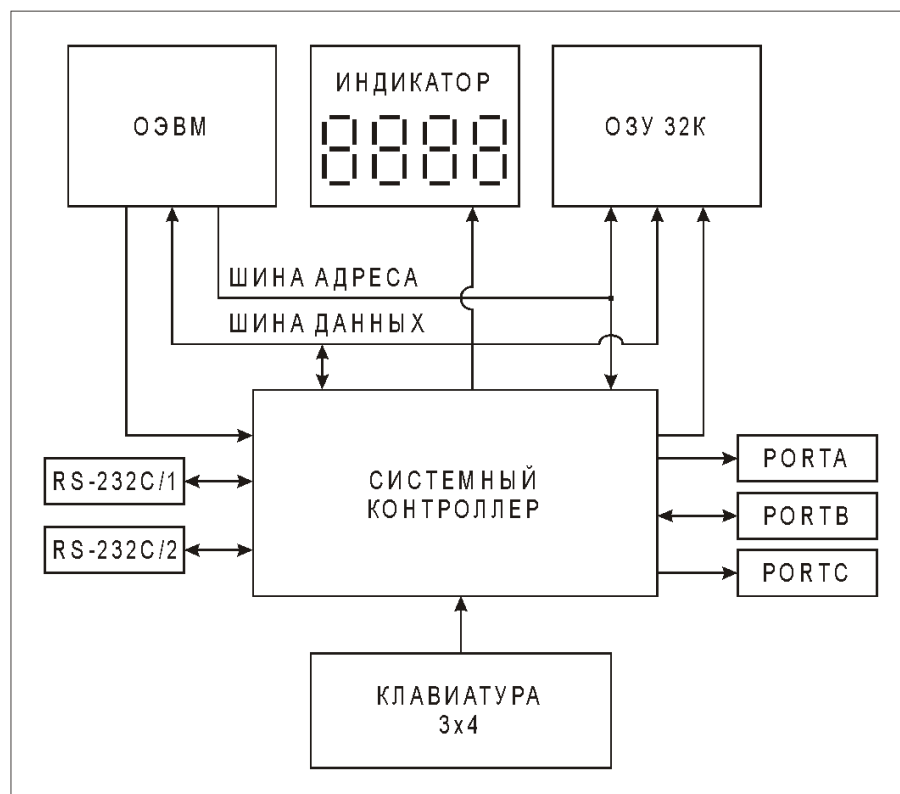


Рисунок 1.1 – Структурна схема стенду EV8031 / AVR

Вся логіка стенду реалізована на програмованій логічній інтегральній мікросхемі EPM7128STC100 (DD4). Системний контролер управляє режимами роботи, вироблення керуючих сигналів для ОЗУ, регістрів - засувов, статичної, динамічної, світлодіодним індикації, клавіатури.

Програма - завантажувач знаходиться в Flash - пам'яті мікроконтролера AT89C51, вона проводить ініціалізацію послідовного приймача ОЕВМ (DD1), перевіряє наявність і розмір пам'яті даних.

Елементи, що використовуються в стенді, наведені в табл. 1.1

Таблиця 1.1 - Перелік інтегральних мікросхем стенду

№ п/п	Позначення на схемі	Позначення (фірмове)	Короткий опис ІМС
1	DD1	AT89C51	Однокристалъна ЕВМ
2	DD2, DD7, DD9, DD11, DD17	74HC573N	8-розрядний регістр
3	DD3	62256	Стат. ОЗУ 32Кб
4	DD4	EPM7128STC100	Прогр. Лог. Схема
5	DD6, DD18	74HC04	6 КМОП інверторів
6	DD8, DD10	1489	Перетворювач рівня RS-232C
7	DD12	ADM485	Перетворювач рівня RS-485
8	DD14	AT29C02	ЭСПЗУ 2 Кбит
9	DD15	DS1621	Цифр. темп. датчик
10	DD16	DS1302	Годинник реального часу

Пам'ять ОЗУ об'ємом 32К ділиться на дві частини по 16К. Одна частина утворює пам'ять програм, а інша - пам'ять даних. У режимі завантаження вся пам'ять 32К відображається в адресний простір, як пам'ять даних.

Адресація (звернення) процесора до периферійних пристроїв стенду реалізовано як адресація до осередків пам'яті даних в адресному просторі від 8000H до FFFFH. Сигнали вибірки периферійних пристроїв формуються дешифратором адреси всередині мікросхеми системного контролера DD4. Розподіл адрес між периферійними пристроями УОС EV8031 / AVR приведено в табл. 1.2, де символом «х» позначено байдуже стан біта (0 чи 1).

При надходженні даних з послідовного порту персонального комп'ютера в послідовний порт (роз'єм X11) стенду, ОЕВМ записує їх в ОЗУ, відведений під пам'ять програм. Сигнали управління - PME, WR, RD, ALE, що формуються процесором і необхідні для звернення до пам'яті даних, надходять через системний контролер. Після прийняття останнього байта завантажувач формує сигнал запуску програми, із записом керуючого коду в системний контролер.

Кнопка SW2, необхідна для формування сигналу скидання на вході RESET процесора, тобто перекладу стенду в режим завантаження і очікування прийому даних з послідовного порту. Процесор готовий приймати дані в пам'ять даних.

Кнопка SW1, необхідна для перезапуску завантаженої з ПК програми, що знаходиться в пам'яті програм (DD3). При натисканні кнопки SW1, загоряється світлодіод HL9. При цьому можлива нова запис програми в стенд з персонального комп'ютера. При передачі даних з персонального комп'ютера в стенд, комп'ютер на лінії RI послідовного порту формує сигнал, який через системний контролер скидає процесор, також як і кнопка SW2.

Таблиця 1.2 – Карта портів введення / виводу даних стенду

Адреса	Тип циклу	B7	B6	B5	B4	B3	B2	B1	B0	Им'я
Периферійний порт										
8xx0	Запис	[Порт А]								PA_REG
8xx1	Запис	[Порт В]								PB_REG
8xx2	Запис	[Порт С]								PC_REG
8xx3	Запис	x	x	x	X	x	TRISC	x	x	TRIS
PKI										
8xx4	Запис	Регістр команд РК - індикатора								LCD_CMD
8xx5	Запис	Регістр даних РК - індикатора								LCD_DATA
Послідовний порт										
9xxx	Чтение	CTS	DSR	DCD	RI	KL3	KL2	KL1	KL0	US_REG
Cxx0	Запись	x	x	X	x	DTR	RTS	CFG1	CFG0	UC_REG
Індикатор і світлодіоди										
Axx0	Запис	[Регістр індикатора 0]								DISPLAY[0]
Axx1	Запис	[Регістр індикатора 1]								DISPLAY[1]
Axx2	Запис	<зарезервовано>								DISPLAY[2]
Axx3	Запис	<зарезервовано>								DISPLAY[3]
Axx4	Запис	DP3	DP2	DP1	DP0	BL3	BL2	BL1	BL0	DC_REG
Axx5	Запис	<зарезервовано>								EDC_REG
Axx6	Запис	LED7	LED6	LED5	LED4	LED3	LED2	LED1	LED0	LED_REG
Управління роботою										
Axx7	Запис	x	x	X	x	x	x	x	RUN	SYS_CTL
Сумісні регістри										
Bxx0	Запис	[Регістр індикатора 1]								DISPLAYB

На рис. 1.2 показано розташування основних органів управління і мікросхем плати ядра YOC EV8031 / AVR, які дозволяють освоїти основні прийоми програмування мікроконтролерних систем.

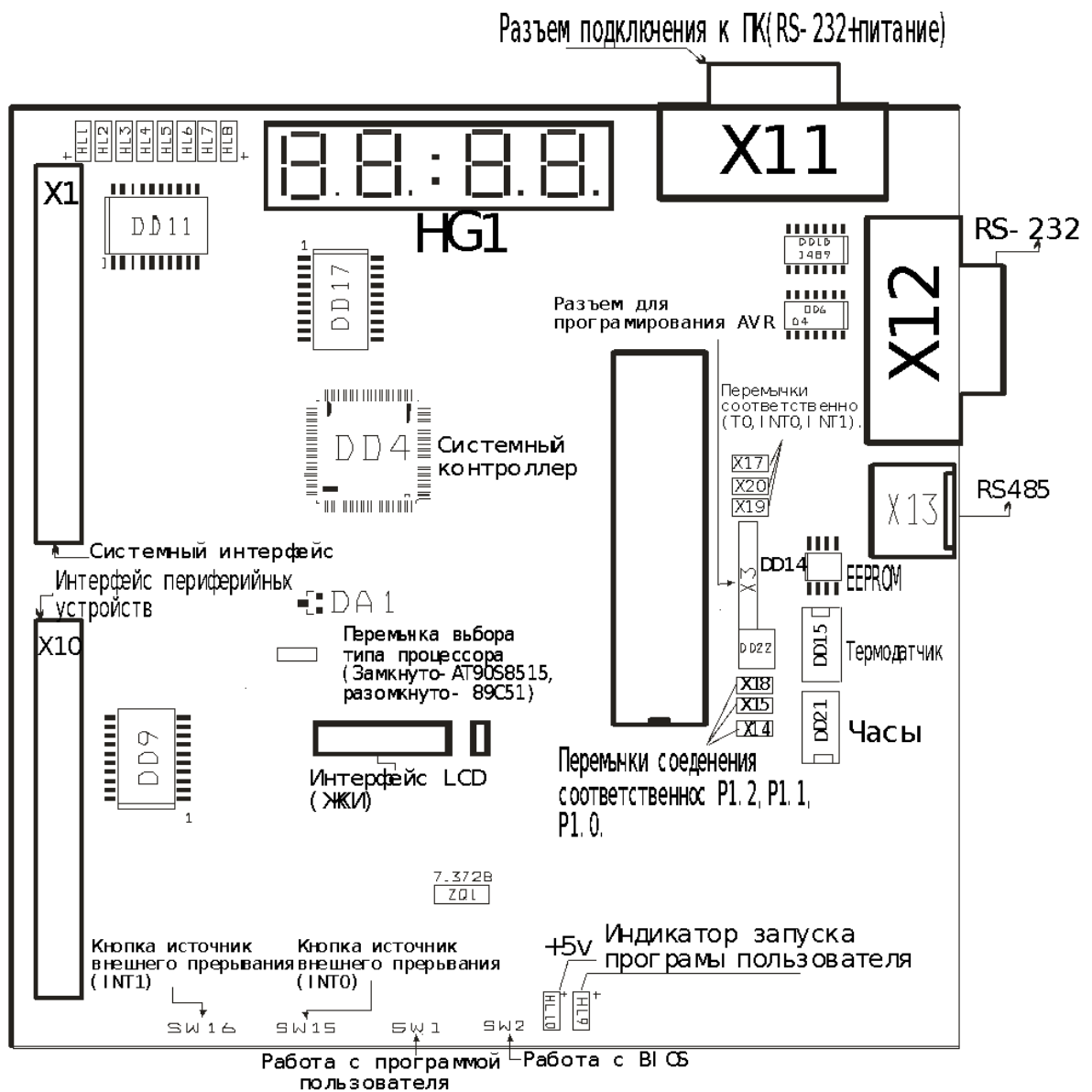


Рисунок 1.2 – Схема розташування елементів плати ядра стенду

Тут позначені:

X1 - Системний інтерфейс з повним адресним простором;

X10 - Інтерфейс розширення для підключення зовнішніх пристроїв з використанням паралельного інтерфейсу;

X11 - Інтерфейс послідовного порту COM1 для зв'язку стенду з РС;

X12 -Інтерфейс послідовного порту COM2 для зв'язку стенду з іншими пристроями мають стандартний порт RS232C;

X3 - Інтерфейс програмування AVR;

X14, X15 -Перемички підключення пристроїв шини I2C до процесора.

1.2 Периферійні пристрої плати ядра стенду

Розглянемо периферійні пристрої, розташовані на платі ядра EV8031 / AVR.

Послідовний приймач. Модуль послідовного зв'язку сформований на базі мікросхеми приймача 1489 передавача 74НС04, мультиплексор а каналу передачі (усередині системного контролера).Швидкість обміну по послідовному порту в режимі завантаження - 9600 біт/с (бод). Швидкість обміну по послідовному порту в налагоджують програму може бути змінена.

Вибір каналу послідовної передачі здійснюється сигналами CFG0, CFG1 за адресою 09001H. Установка цих бітів в "логічний нуль" включає порт 1 (на схемі X11). Цей порт має неповний набір сигналів (RxD, TxD, RI) і призначений для запису програми в стенд.

Програмна установка сигналів CFG0 в "0", а CFG1 в "1" формує вибірку додаткового каналу послідовної передачі даних (роз'єм X12). Додатковий послідовний канал має повний набір сигналів інтерфейсу RS-232C.

Статичний семисегментний індикатор. Чотирирозрядний семисегментний індикатор підключений до системного контролеру, який автоматично виконує декодування двійкового коду в код 7 - ми сегментного індикатора. Індикатор працює відразу після подачі живлення і при виконанні будь-якої програми. До складу індикатора входить два восьмирозрядних регістра, вміст яких і відображається на індикаторі. Вміст регістра з адресою 0A000h відображається на двох лівих розрядах, вміст регістра з адресою 0A001h (0B000h) - на двох правих розрядах в шістнадцятковій формі (дані адреси підписані безпосередньо під мікросхемою 7 - ми сегментного індикатора). Управління десятковими точками і гасінням розрядів індикатора здійснюється за допомогою регістра

DC_REG (адреса - 0A004h). Біти DP3 ... DP0 керують десятковими точками. Запис 1 в відповідний розряд включає десяткову точку. Біти BL3 ... BL0 керують гасінням розрядів індикатора. Запис 1 в ці біти викликає гасіння відповідного розряду індикатора.

Матрична клавіатура. Стан 4-х стовпців матриці клавіатури зчитується з осередку з базовою адресою 09000h - біти 3 ... 0. Відповідний стовпець вибирається нулем в розрядах адреси A2 ... A0. Тобто, адреса 09006h вибирає перший стовпець, адреса 09005h - другий стовпець, адреса 09003h - третій стовпець. Ознака натиснутою кнопки клавіатури зчитується як нуль у відповідному розряді.

Включення портів OEVB і EEPROM пам'яті. Лінії P1.0, P1.1 OEVB можуть бути відключені від внутрішньої периферії стану (шина I2C) перемичками X14, X15. На роз'єм інтерфейсу розширення ці сигнали приходять, минаючи перемички.

1.3 Плата розширення

Плата розширення, в комплексі з платою ядра УОС EV8031 / AVR, призначена для проведення лабораторних робіт, присвячених вивченню різноманітних аналого - цифрових пристроїв мікроконтролерних систем. Структурна схема плати розширення приведена на рис. 1.3, а її зовнішній вигляд - на рис. 1.4.

Тут позначені:

8888 - 4 розрядна динамічна індикація;

ІПУ - Інтерфейс периферійних пристроїв;

ЦАП - Цифро-аналоговий перетворювач;

СДІ - Світлодіодні індикатори;

ЗСИ - знакосінтезующих індикатор 5x7;

ДПЧ - Генератор змінної частоти імпульсної послідовності;

ІНТ - Кнопки запиту переривання;

ПН - Джерело вимірюваної напруги;

ШД - Шина даних..

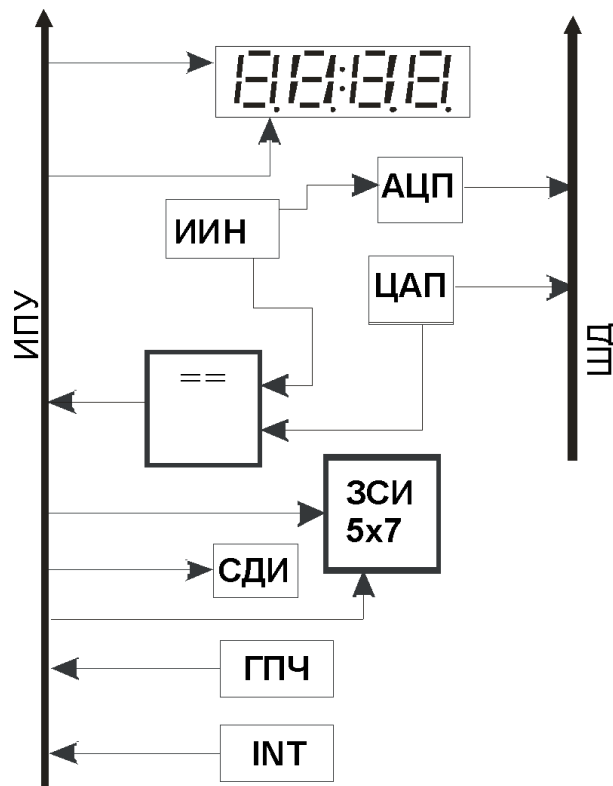


Рисунок 1.3 – Структурна схема плати розширення

Тут позначені:

HG1 - знакосінтезуюючих індикатор 5x7;

HL2 - 4-х розрядна динамічна індикація;

HL1 - світлодіодний індикатор спрацьовування компаратора;

J1 - перемичка підключення до роз'єму J2 виходу генератора з постійною частотою генерації (T0);

J2 –BNC роз'єм підключення зовнішніх контрольно-вимірювальних приладів;

J3 - перемичка підключення до роз'єму J2 виходу генератора із змінною частотою генерації (T1);

J4 - перемичка підключення до роз'єму J2 виходу ЦАП;

J5 - підключення до входу компаратора движка змінного резистора R19 (вихід джерела змінного сигналу);

J6 - підключення до входу компаратора виходу каскаду на фототранзистор VT3;
 J7 - динамік (speaker);
 J8 - підключення до входу динаміка (speaker) сигналу з лінії порту P1.0;
 J9 - підключення до входу динаміка (speaker) сигналу з виходу ЦАП;
 R19 - змінний резистор - джерело вхідного сигналу для АЦП;
 R4 - змінний резистор - змінює частоту проходження імпульсів генератора змінної частоти.

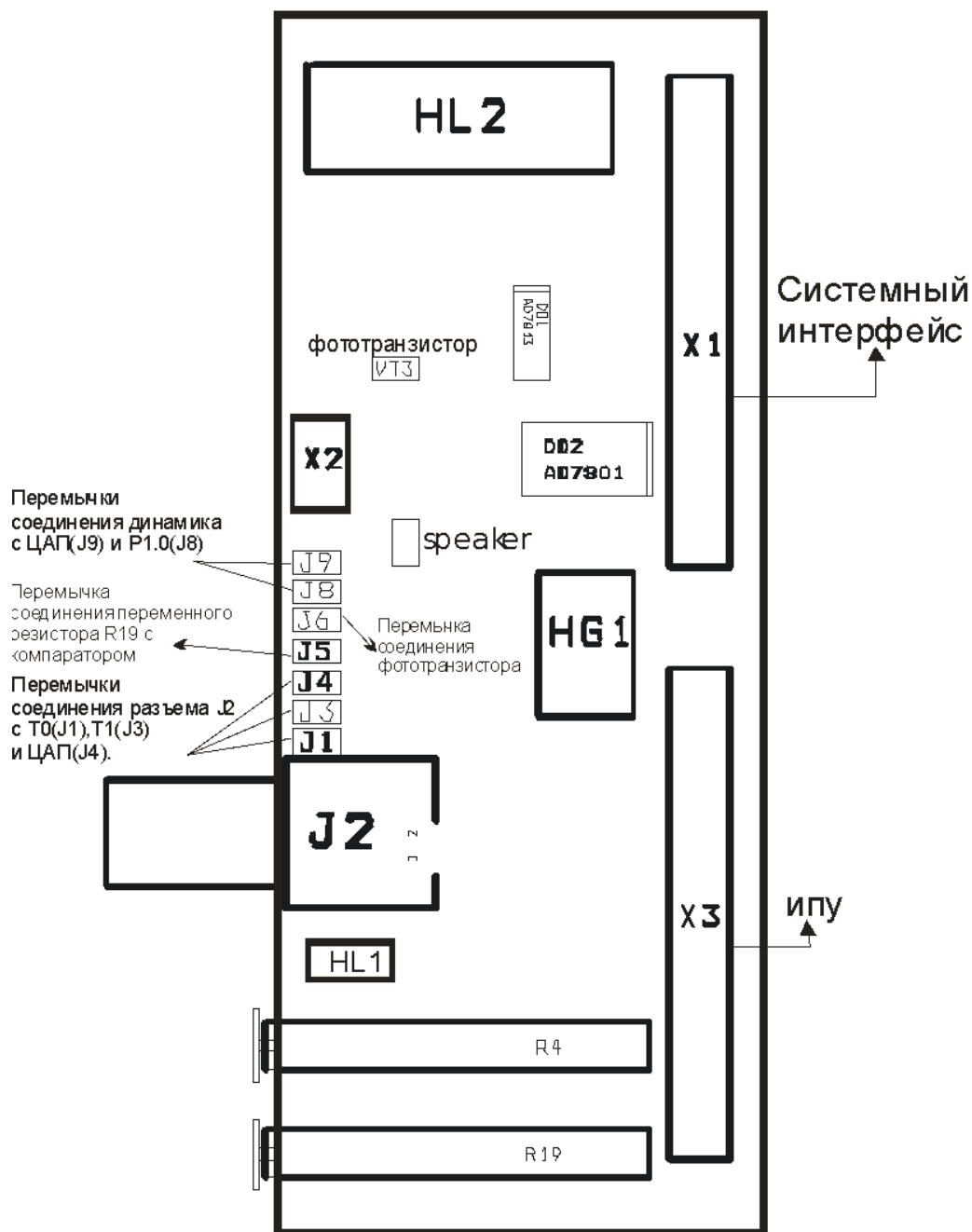


Рисунок 1.4 – Плата периферійних пристроїв стенду

Плати ядра і розширення стенду з'єднуються між собою за допомогою двох інтерфейсних роз'ємів: системного (X1) і периферійних пристроїв (X3).

За допомогою роз'єму X2, можна організувати подачу на вхід компаратора АЦП зовнішнього сигналу, а також зняти напругу живлення (+5 В і загальний) для зовнішніх пристроїв введення - виведення даних.

1.4 Периферійні пристрої плати розширення

Розглянемо периферійні пристрої плати розширення.

Цифроаналоговий перетворювач. ЦАП виконаний на мікросхемі AD7801 DD2 (8 - розрядний ЦАП). Вхідними сигналами для ЦАП є лінії AD7 - AD0. Його вихідний сигнал можна перемикає за допомогою перемичок на роз'єм BNC J2 (J4) або на вхід динаміка (speaker) (J9).

Аналого-цифровий перетворювач. АЦП може бути організований на базі мікросхем ЦАП AD7801 (DD2) і операційному підсилювачі LM358 (DA1.2), що використовуються як компаратора. Вхідним аналоговим сигналом для АЦП є сигнали, які можуть подаватися з: змінного резистора R19 (установка перемички J5); виходу каскаду на фототранзистор VT3 (установка перемички J6); роз'єму X2 (контакт 1). Лінії AD7-AD0 (див. Схему стенда рис. П1.3) використовуються для формування цифрового вхідного коду. На виході ЦАП формується напруга, пропорційне вхідного коду. Сигнал спрацювання компаратора (DA1.2) подається на лінію P1.7 ОЕВМ. Спрацьовування компаратора можна спостерігати по загоряння світлодіода HL1 (P1.7 = 0).

Генератори. Генератор із змінною частотою генерації в межах 1-50 кГц виконаний на елементах R1, R4, R5, R7, R10, R11, R15, R16, C3, VT1, DA1-1 (зміна частоти здійснюється за допомогою резистора R4), а генератор фіксованої частоти близько 10 кГц виконаний на елементах R19, R20, C16, DD18.1, DD18.2, DD18.3.

Вивід дискретної інформації. Вивід дискретної інформації здійснюється за допомогою 4-х розрядного 7 - ми сегментного індикатора HL2, включеного за схемою динамічної індикації. Управління динамічною індикацією здійснюється за допомогою дешифратора DD3, формує сигнали вибірки відповідного індикатора за адресою PC0, PC1 порту C. яку вказують дані надходять з порту PB (PB0 - PB6 надходять на сегменти A, B, C, D, E, F і G, а PB7 - на десяткову точку DP).

Зовнішній вид плати розширення для систем автоматичного управління представлений на рисунку 1.5.

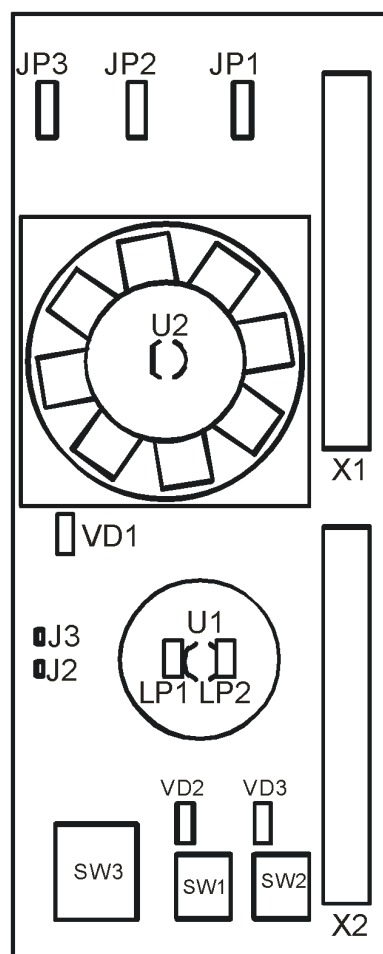


Рисунок 1.5 – Плата розширення для систем автоматичного управління

Схема розташування елементів плати розширення

JP1- Перемичка вибору виду регулювання. Вибирається регулювання частоти обертів двигуна (MOTOR), або регулювання температури (LAMP).

JP2- Перемикач вибору способу регулювання. Вибирається регулювання лінійне, змінюючи амплітуду вихідного сигналу з ЦАП (LINE), або регулювання ШІМ (широтно-імпульсна модуляція) (PWM).

JP3- Перемикач вибору об'єкта регулювання. Вибирається режим регулювання частоти обертання / температури (REG), або режим управління звуком (SOUND).

J2, J3- Перемикачі вибору способу управління звуком. 2 замкнута, 3 розімкнута - звуком керують безпосередньо в мікроконтролер. 2 розімкнута, 3 замкнута -Звук керують за допомогою ЦАП.

X1, X3 роз'єми підключення до основної плати стенду;

U1- Давач температури;

U2- Давач оборотів (На основі ефекту Холла);

LP1, LP2- нагрівальні елементи (Лампи розжарювання);

SW1, SW2- Дискретні кнопки;

SW3- багатооборотна перемикач;

VD1- індикатор обертання двигуна;

VD2, VD3- індикатори натискання кнопок SW1, SW2.

2. Порядок виконання роботи

2.1 Вивчити порядок підключення стенду до комп'ютера.

2.2. Розглянути блок - схему стенда. Керуючись схемами розташування елементів, на платі ядра і розширення візуально визначити місце розташування вузлів, які будуть часто використовуватися при виконанні лабораторних робіт:

- мікросхема мікроконтролера МК51 (DD1);
- статичний 4-х розрядний індикатор;
- динамічний 4-х розрядний індикатор;
- лінійка з 8 - ми світлодіодів;
- блок рідко - кристалічного індикатора (PKI);
- ІМС цифрового датчика температури (DD15);

- клавіатура;
- ІМС годин реального часу (DD16);
- кварцовий резонатор МК51;
- кнопки введення дискретної інформації (SW15, SW16) і вибору режиму роботи стенда (SW1, SW2);
- індикатори живлення і завантаження програми користувача;
- знакосінтезуючий індикатор;
- динамік;
- ІМС ЦАП (DD2);
- резистори регулювання рівня вхідного сигналу для АЦП (R19) і частоти ДПЧ (R4);
- перемички («джампери») J1 - J9 для комутації сигналів на платі розширення (записати якісь перемички встановлені).

2.3 Провести тестування і записати результати для кожного тесту.

Тестування стенду

При подачі напруги живлення на стенд, процесор автоматично визначає розмір пам'яті даних з відображення на індикаторі стенду HG1 числа ємності пам'яті в байтах (256 байт). Тим самим, проводиться тестування процесора, регістра засувки, дешифратора адреси, схеми скидання процесора).

Програма TEST1.HEX, яка входить в програмне забезпечення стенду, дозволяє перевірити канал послідовної передачі даних з персонального комп'ютера на стенд (мікросхема приймача даних), схеми перемикування і мультиплексора вибірки, дешифратора адреси і все розряди елементів статичної, динамічної і знакосінтезующих індикації.

Тестування плати розширення здійснюється за допомогою вбудованої в ПЗУ завантажувача програмою тестування.

Вхід в тестовий режим проводиться наступним чином: утримуючи будь-яку кнопку на клавіатурі, натиснути і відпустити кнопку скидання (SW2).

Вихід з режиму тестування проводиться шляхом натискання кнопки скидання або за кодом виходу.

Для виклику тесту необхідно ввести номер тесту і натиснути кнопку ("##") , а для виходу з поточного тесту натиснути будь-яку кнопку на клавіатурі.

Коди вбудованих тестів:

- "1" - Вимірювання частоти генератора з незмінною частотою генерації, кГц;
- "2" - Вимірювання частоти генератора із змінною частотою генерації, кГц;
- "3" - "Той, що біжить вогник" на світлодіодах;
- "4" - Послідовне засвічування сегментів семисегментних індикаторів;
- "5" - "Той, що біжить вогник" на матриці світлодіодів;
- "6" - Програма АЦП, відображає десятковий код ЦАП;
- "10" - Програма тестування мікросхеми годин реального часу DS1302, настройка годин за допомогою кнопок SW15 і SW16 розташованих на цифровий платі;
- "11" - Програма тестування інтегрального датчика температури DS1631;

2.4 Завантажити програми 1.ihx і 2.ihx в стенд через програму Eval.

2.5 Вивчити порядок набору програми в текстовому редакторі Kiel, її подальшого завантаження і виконання в УОС на наступному навчальному прикладі.

На статичному індикаторі з адресою B000h відобразити число 17h.

ORG 0

Start:

```
movDPTR,#0B000h    ; заносимо в регістр DPTR
                    ; адреса индикації
movA,#17h           ; заносимо в регістр A дані для відображення
movx @DPTR,A        ; выводимо на індикацію, адреса якої
                    ; знаходиться в регістрі DPTR, число 17
jmpStart            ; перехід на початок
END
```

Робота в текстовому редакторі Kiel

2.1.1 Створення проекту

1. Запустити Keil.
2. Ознайомитися з елементами управління. Середовище розробки Keil є типовим додатком Windows. Її вигляд з основними елементами управління зображений на рисунку 1.6.

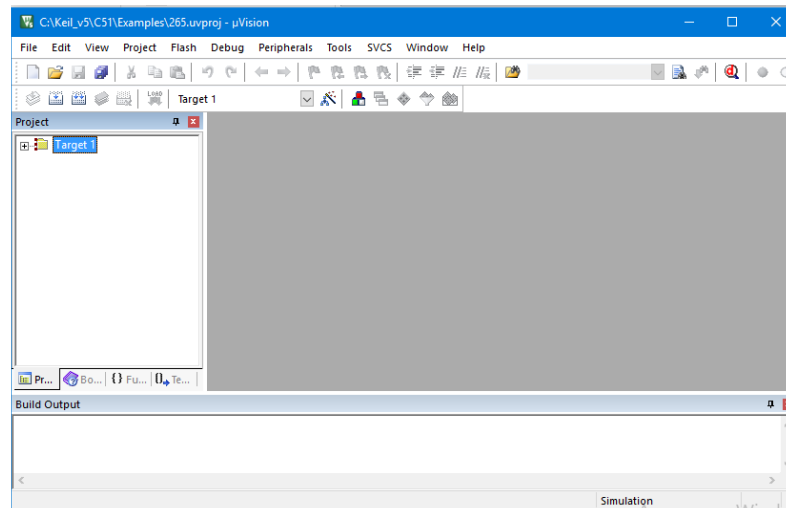


Рисунок 1.6 – Інтерфейс середовища Keil μ Vision

3. Створити новий проект (див. рис. 1.7).

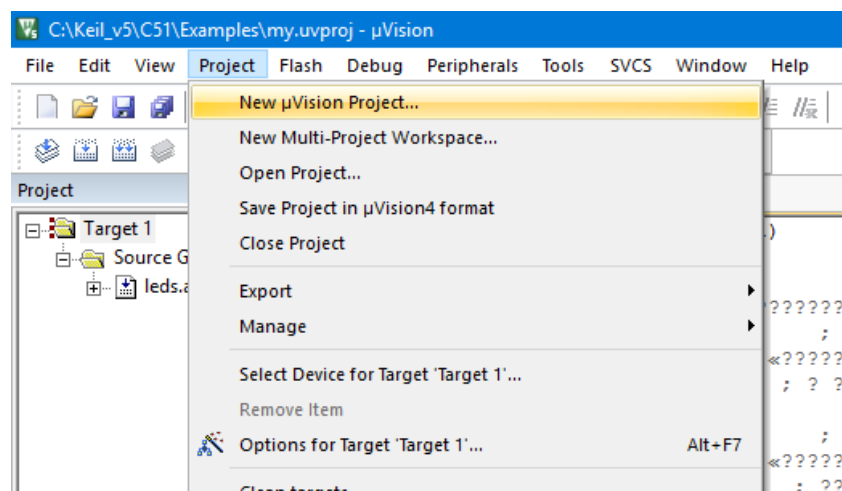


Рисунок 1.7 – Створення нового проекту

4. Зберегти проект на диску. Вибрати необхідний тип мікроконтролера (див. Рис.1.8) AtmelAT89S52.

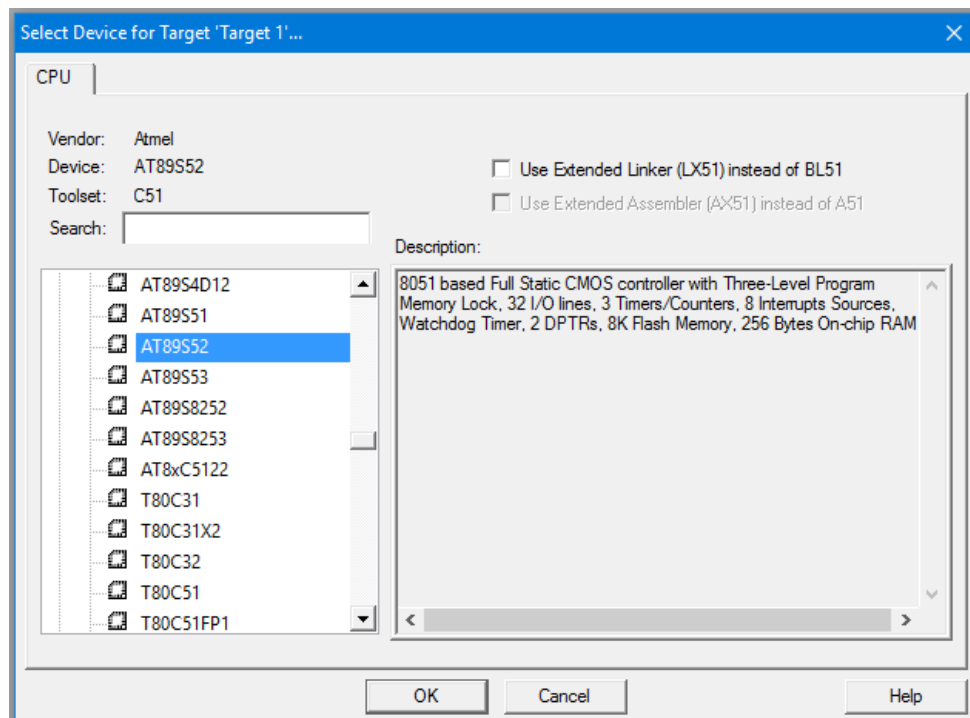


Рисунок 1.8 – Вибір типу мікроконтролера

;

5. Створити новий файл. Вибрати тип файлу і зберегти його в папці під назвою 1 (див. рис.1.9). З'явиться вікно з ім'ям 1.a51 (рис.1.10).

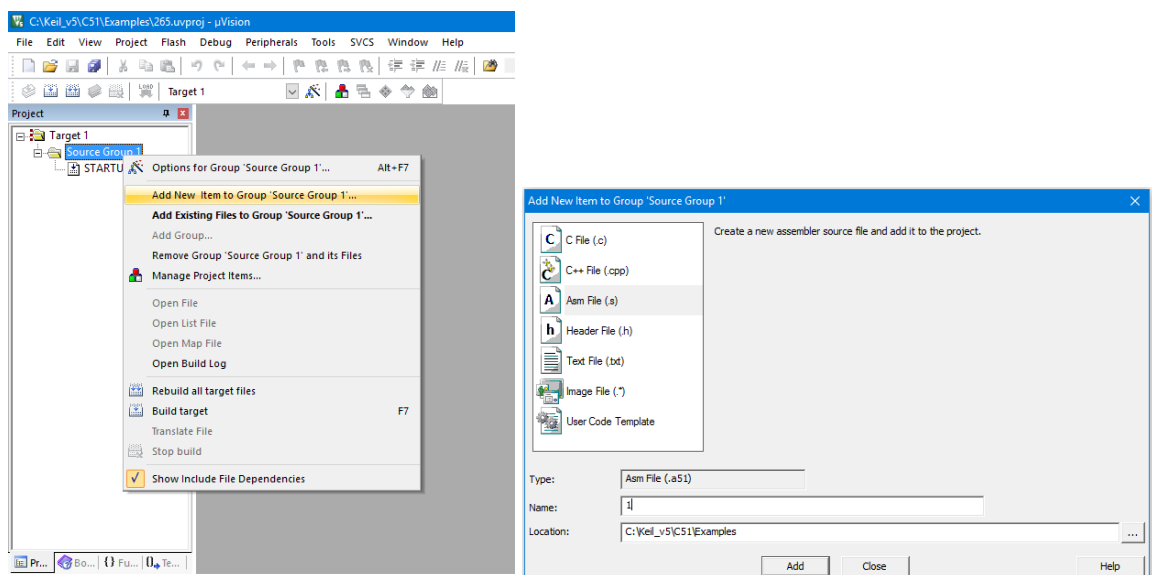


Рисунок 1.9 – Створення проекту

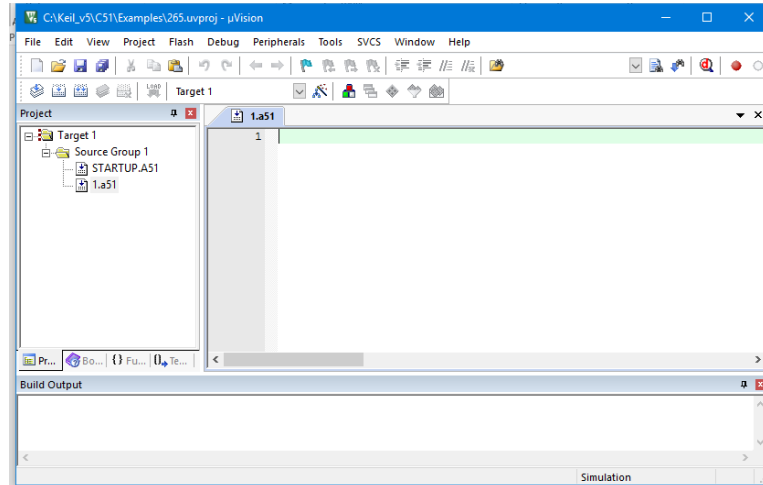


Рисунок 1.10 – Вікно проекту

6. Виконати настройку проекту:

1. Вибрати Project -> Options for target '...'.
2. У вкладці "Output" поставити "пташку" CreateHEXFile. (рис. 1.11).

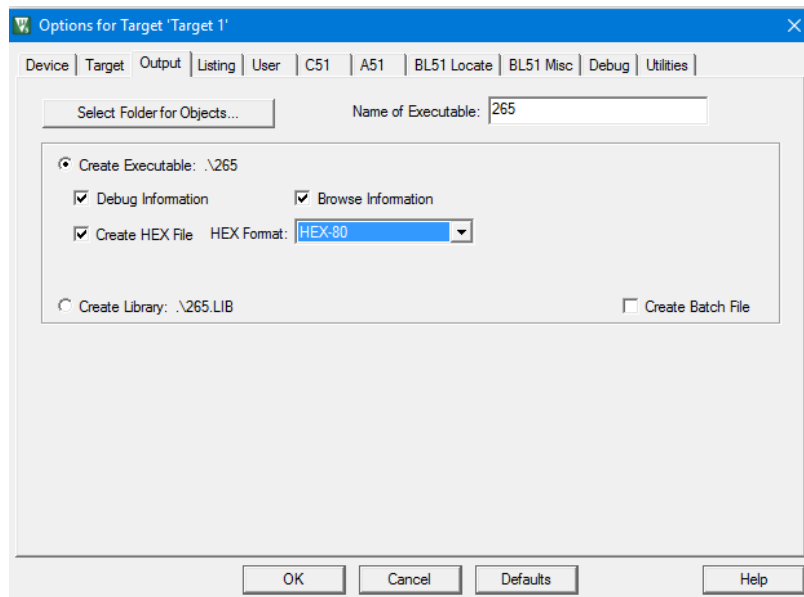


Рисунок 1.11 – Налаштування проекту

6. Скомпілювати програму.

У програму поміщається спеціальний рядок, в якій Асемблеру повідомляється ім'я файлу, що включається. Наприклад, рядки

\$INCLUDE(C:\...\MOD51),

де **C:\...\MOD51** - шлях,

MOD51 – ім'я вмикаємого файла.

Цей рядок необхідно включати в програму в тому випадку, коли транслятор Асемблера МК51 не сприймає ряду символів (наприклад, позначення регістра **B**, портів **P0**, **P1**, **P2**, **P3** і їх окремих битів).

3. Зміст звіту

- 3.1 Блок - схема стенду з коротким описом функцій навчального стенду.
- 3.2 Порти введення / виводу даних стенду з таблицею їх адрес (табл. 1.2).
- 3.3 Порядок виконання роботи.
- 3.4 Результати тестування стенду.
- 3.5 Роздруківки файлів навчальної програми з розширеннями asm і hex.
- 3.6 Висновки по роботі.

4. Контрольні питання

- 4.1 За блок - схемою навчального стенду коротко поясніть особливості його структурної схеми і організацію зв'язку з ПК і УВВ.
- 4.2 Для чого необхідний текстовий редактор Keil
- 4.3 Розкажіть про основні функції програмного забезпечення
- 4.4 Перерахуйте і поясніть основні етапи розробки та виконання на навчальному стенде прикладної програми.
- 4.5 З якою метою проводиться тестування?
- 4.8 Перерахуйте види індикаторів, представлених на схемі.
- 4.9 Периферійні пристрої плати ядра стенду
- 4.10 Плата розширення і її периферійні пристрої

ЛАБОРАТОРНА РОБОТА №2

Система команд МК51

Мета роботи: вивчити групи команд мікроконтролера, вивід результатів розрахунку на статичний індикатор

1. Теоретична частина

Для відображення цифрової інформації найбільшого поширення набули семисегментні індикатори, в яких зображення цифри складають з семи лінійних світлодіодних сегментів розташованих у вигляді цифри 8. На основі світлодіодів і семисегментних індикаторів будуються підсистеми відображення інформації. При побудові підсистем відображення інформації розрізняють два підходи - динамічна і статична схема побудови підсистеми індикації.

Статична індикація полягає в постійній підсвічуванні індикаторів HL1-n від одного джерела інформації рис.2.1.

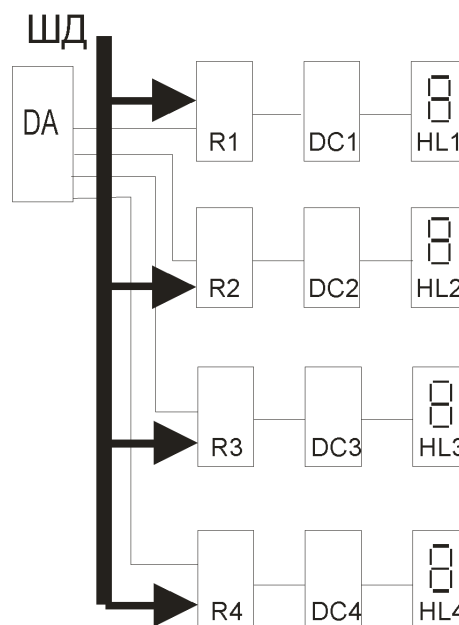


Рисунок 2.1 – Структурна схема статичної індикації

DA – дешифратор адреси, необхідний для вибірки відповідного регістру, R1-R4-регістри, в яких тимчасово зберігається значення коду числа для відображення

(відповідний регістр вибирається DA), DC1-DC4 - семисегментні дешифратори, що перетворюють двійковий код в семисегментний код, HL1-HL4 - семисегментний індикатори. ШД - шина даних, по ній здійснюється передача даних на індикацію.

У такій системі кожен індикатор HL1-n підключений через власний дешифратор DC1-n і регістр-засувку RG1-n до шини даних, вибірка регістрів RG1-n проводиться за допомогою селектора адреси CA. Апаратні витрати при такій організації складають n пар регістр + дешифратор при n десяткових розрядів індикатора.

Для індикації даних на статичному індикаторі HG1 потрібно записати їх у відповідні регістри - дешифратори RG - DC: DISPLAY [0] - адреса 0A000h (1, 2 знакомісце) або DISPLAY [1] - адреса 0B000h (3, 4 знакомісце).

Режим роботи статичного індикатора задається шляхом установки бітів регістра управління DC_REG за адресою 0A004h. Формат цього регістра дан в карті портів введення / виводу даних стенду (лб.р. №1), де DPi - біт управління десятковою крапкою розряду; BLi - біт управління гасінням розряду; i = 0, 1, 2, 3 - номер розряду (знакоместо). Установка в 0 бітів регістра управління DC_REG блокує індикацію десяткових точок (DP0 - DP3) і дозволяє роботу індикаторів (BL0 - BL3).

Основні команди :

Команди передачі даних

Назва команди	Мнемокод	Б	Ц	Операція
Пересилання в акумулятор з регістра (n = 0 - 7)	MOV A, Rn	1	1	(A) ← (Rn)
Завантаження в акумулятор константи	MOV A, #d	2	1	(A) ← #d
Пересилка в регістр з акумулятора	MOV Rn, A	1	1	(Rn) ← (A)
Пересилка в акумулятор байта з розширеної ВПД	MOVX A, @DPTR	1	2	(A) ← ((DPTR))
Пересилка в ВПД з акумулятора	MOVX @Ri, A	1	2	((Ri)) ← (A)
Пересилка в розширену ВПД з акумулятора	MOVX @DPTR, A	1	2	((DPTR)) ← (A)
Обмін акумулятора з регістром	XCH A, Rn	1	1	(A) ↔ (Rn)

-команди арифметичні

Назва команди	Мнемокод	Б	Ц	Операція
Додавання акумулятора з регістром (n = 0 - 7)	ADD A, Rn	1	1	$(A) \leftarrow (A) + (Rn)$
Віднімання з акумулятора регістра і позику	SUBB A, Rn	1	1	$(A) \leftarrow (A) - (C) - (Rn)$
інкремент акумулятора	INC A	1	1	$(A) \leftarrow (A) + 1$
інкремент регістра	INC Rn	1	1	$(Rn) \leftarrow (Rn) + 1$
декремент акумулятора	DEC A	1	1	$(A) \leftarrow (A) - 1$
декремент регістра	DEC Rn	1	1	$(Rn) \leftarrow (Rn) - 1$
Множення акумулятора на регістр B	MUL AB	1	4	$(B)(A) \leftarrow (A) \cdot (B)$
Розподіл акумулятора на регістр B	DIV AB	1	4	$(A), (B) \leftarrow (A) / (B)$

-логічні команди

Назва команди	Мнемокод	Б	Ц	Операція
Логічне І акумулятора і регістра	ANL A, Rn	1	1	$(A) \leftarrow (A) \wedge (Rn)$
Логічне АБО акумулятора і регістра	ORL A, Rn	1	1	$(A) \leftarrow (A) \vee (Rn)$
Виключне АБО акумулятора і регістра	XRL A, Rn	1	1	$(A) \leftarrow (A) \oplus (Rn)$
Скидання акумулятора	CLR A	1	1	$(A) \leftarrow 0$
Інверсія акумулятора	CPL A	1	1	$(A) \leftarrow (\bar{A})$
Зсув акумулятора вліво циклічно	RL A	1	1	$(A_{n+1}) \leftarrow (A_n),$ где $n = 0, \dots, 6,$ $(A_0) \leftarrow (A_7)$
Зсув акумулятора вліво через перенос (Арифметичний Зсув вліво)	RLC A	1	1	$(A_{n+1}) \leftarrow (A_n),$ где $n = 0, \dots, 6,$ $(A_0) \leftarrow (C),$ $(C) \leftarrow (A_7)$
Зсув акумулятора вправо циклічно	RR A	1	1	$(A_n) \leftarrow (A_{n+1}),$ где $n = 0, \dots, 6,$ $(A_7) \leftarrow (A_0)$
Зсув акумулятора вправо через перенос (Арифметичний Зсув вправо)	RRC A	1	1	$(A_n) \leftarrow (A_{n+1}),$ где $n = 0, \dots, 6,$ $(A_7) \leftarrow (C),$ $(C) \leftarrow (A_0)$
Обмін місцями тетрад в акумуляторі	SWAP A	1	1	$(A_{0-3}) \leftrightarrow (A_{4-7})$

-команди переходів

Команда DJNZ <байт>, <зміщення>

Команда "декремент і перехід, якщо дорівнює нулю" виконує віднімання "1" з вказаною комірки і здійснює розгалуження по обчислюваному адресою, якщо результат не дорівнює нулю. Початкове значення 00H перейде в 0FFH. Адреса переходу (розгалуження) обчислюється складанням значення зміщення (зі знаком), зазначеного в останньому байті команди, з вмістом лічильника команд, збільшеним на довжину команди DJNZ.

Команда CJNE <байт призначення>, <байт джерело >,< зсув>

Команда "порівняння і перехід, якщо не дорівнює" порівнює значення перших двох операндів і виконує розгалуження, якщо операнди не рівні. Адреса переходу (розгалуження) обчислюється за допомогою додавання значення (зі знаком), зазначеного в останньому байті команди, з вмістом лічильника команд після збільшення його на три.

Прапор переносу "C" встановлюється в "1", якщо значення цілого без знака <байта призначення> менше, ніж значення цілого без знака <байта джерела>, в іншому випадку перенесення скидається (якщо значення операндів рівні, прапор переносу скидається). Ця команда не впливає на операнди.

Операнди, що стоять в команді, забезпечують комбінації чотирьох режимів адресації:

- якщо байтом призначення є акумулятор:

- 1) прямий

- 2) безпосередній,

- якщо байтом призначення є будь-яка осередок ОЗУ з побічно-реєстрової адресацією:

- 3) безпосередній до реєстровому

- 4) безпосередній до побічно-реєстровому

В основі багатьох програм лежить використання підпрограми часової затримки TIME. Підпрограма вимагає 1, 2 або 3 реєстра (R1, R2, R3). Суть підпрограми TIME - виконання команд «цикл в циклі». Цикл організується за

допомогою команди DJNZ Rn, Met, яка наказує декремент (- 1) вмісту регістра Rn і перехід на мітку Met, якщо не нуль. Розглянемо підпрограму TIME для організації тривалих тимчасових затримок.

```
TIME:                ;підпрограма затримки
mov R1,#const        ; завантаження константою const регистр R1
C3:  mov R2,#0FFh     ; завантаження константою FFh (25510) R2
C2:  mov R3,#0FFh     ; завантаження константою FFh (25510) R3
C1:  djnz R3, C1       ; перший цикл (метка C1)
      djnz R2, C2       ; другий цикл (метка C2)
      djnz R1, C3       ; третій цикл (метка C3)
      ret              ;вихід з програми
```

Визначимо число машинних циклів, що витрачаються МК51 на виконання підпрограми TIME:

$$T_{\text{TIME}} \approx \text{const} \cdot 130818$$

Знаючи тривалість одного машинного циклу, отримаємо час виконання підпрограми (час затримки):

$$T_{\text{ЗАДЕРЖКИ}} = T_{\text{TIME}} T_{\text{МЦ}} = (\text{const}130818) \cdot 1,628 \cdot 10^{-6} \text{ (секунд)}.$$

Наприклад, при const = 5 отримаємо $T_{\text{ЗАТРИМКИ}} = 1,065$ секунди.

2. Порядок виконання роботи

2.1 Вивчити команди пересилання, обробки даних (арифметичні і логічні), команди переходів.

2,2 Набрати і виконати програму застосування команд обробки даних відповідно до свого варіанту.

3. Зміст звіту

3.1 Короткий перелік груп команд з описом їх призначення і особливостей їх виконання.

3.2 Порядок виконання роботи

3.3 Контрольні питання.

3.4 Програма.

3.5 Висновки по роботі.

4. Контрольні питання

4.1 Перерахуйте основні групи команд МК51.

4.2 Перерахуйте види адресації в системі команд МК51.

4.3 Команди пересилання даних.

4.4 Арифметичні команди

4.5 Команди перенесення.

4.6 Чому для відображення результату виконання команд множення і ділення потрібно реєстрова пара (реєстр А + регістр В)?

4.7 Назвіть основну особливість виконання команд обробки даних (арифметичних і логічних).

4.8 Регістри стану. Прапори.

ЛАБОРАТОРНА РОБОТА №3

Відображення інформації світлодіодними індикаторами

Мета роботи: вивчити способи організації та роботи цифрової індикації на основі світлодіодів.

1. Теоретична частина .

Часто потрібно індикація типу ТАК/НІ, вмикання/вимикання. Така інформація відображається за допомогою окремих світлодіодів. На рис. 3.1 показані схеми включення світлодіодів, де в якості ключів використовується логічний інвертор.

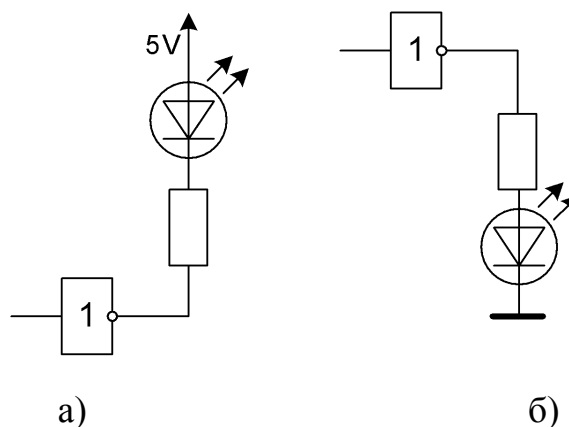


Рисунок 3.1 – Включення світлодіодних індикаторів

: а - з відкритим колектором; б - типова схема

У напівпровідникових світлодіодах використовується властивість р - n переходу випромінювати світло у видимій частині спектру при протіканні через нього прямого струму ($I_{пр} = 5 - 20 \text{ mA}$, $U_{пр} = 2 - 3 \text{ V}$). Баластний резистор включається послідовно з світлодіодом і обмежує величину його прямого струму $I_{пр}$.

У лабораторному стенді лінійка світлодіодів **HL1 - HL8**, доступ до яких здійснюється як до осередку зовнішнього ОЗУ за адресою 0A006h. Світлодіоди включаються записом логічних одиниць у відповідні розряди.

2. Порядок виконання роботи

2.1 Розробити програму для виконання індивідуального завдання відповідно до заданого варіантом. При розробці програми рекомендується використовувати підпрограми.

2.2 Ввести програму індивідуального завдання в персональний комп'ютер.

2.3 Завантажити програму в стенд. Переконавшись в правильному виконанні індивідуального завдання. При негативному результаті змінити програму і повторити її завантаження в стенд.

2.4 Роздрукувати лістинги правильно працюючої програми.

3. Зміст звіту

3.1 Текст програми

3.2 Розрахунок констант для підпрограми затримки.

3.3 Висновки по роботі.

4. Контрольні питання

4.1 Типи індикація

4.2 Організації світлодіодної індикації у стенді

4.3. Алгоритм написання програм

4.4. Вплив тривалості часу затримки на роботу програми.

Відображення інформації знаковинтезуючим індикатором

Мета роботи: вивчити способи організації та роботу цифрової індикації на основі знаковинтезуючих світлодіодних матриць.

1. Теоретична частина

Знакосинтезуюча індикація (ЗСІ) використовується для відображення алфавітно - цифрової інформації. ЗСІ виконуються у вигляді ІМС, що представляє собою матрицю світлодіодів розмірністю $m \times n$, де m - число рядків, n - число колонок матриці (рис. 4.1). Найчастіше ЗСІ мають розмірність 7×5 або 9×7 .

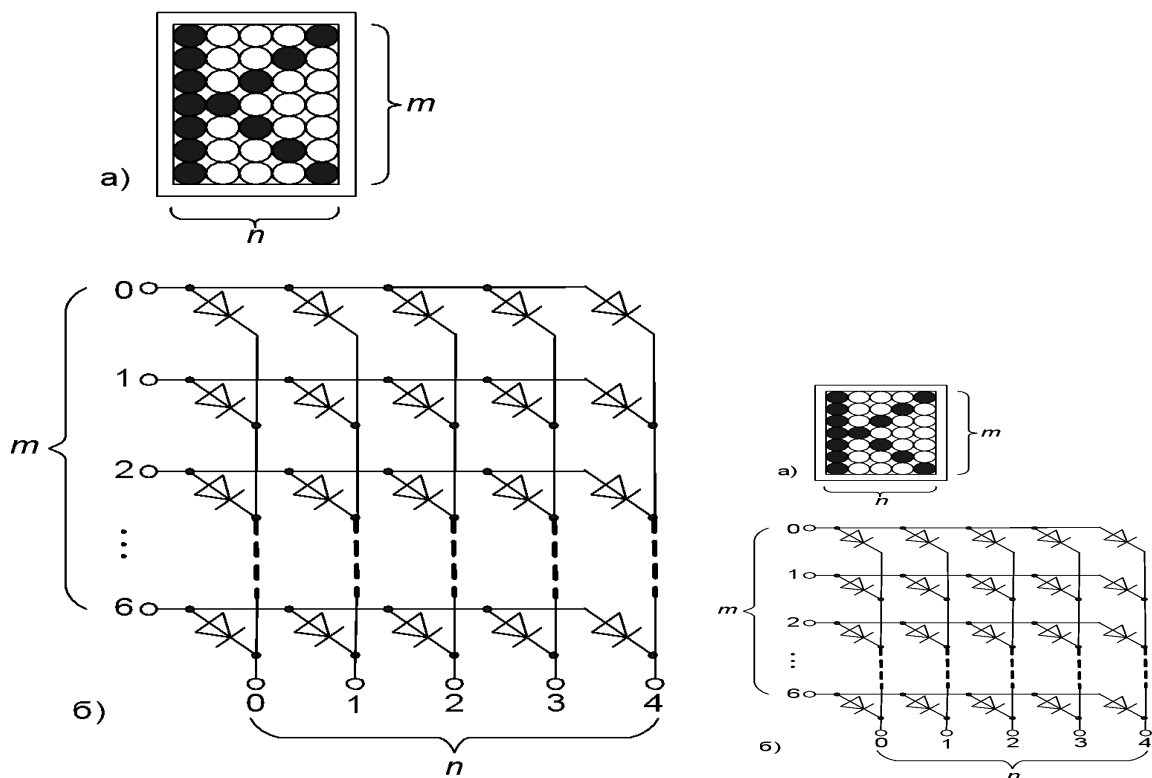


Рисунок 4.1 – Знакосинтезуючий індикатор:

а - зовнішній вигляд; б - схема

Для включення одного світлодіода матриці необхідно забезпечити прямий струм 10 - 15 мА при прямому падінні напруги в 2 - 2,5 В. Підключення ЗСІ до МК51 здійснюється за допомогою керованих схем формування струму колонок і рядків (рис.4.2).

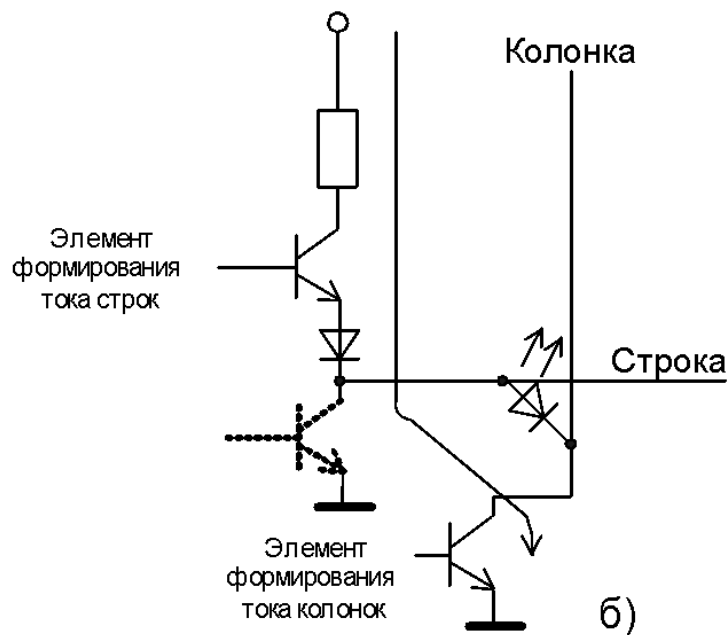
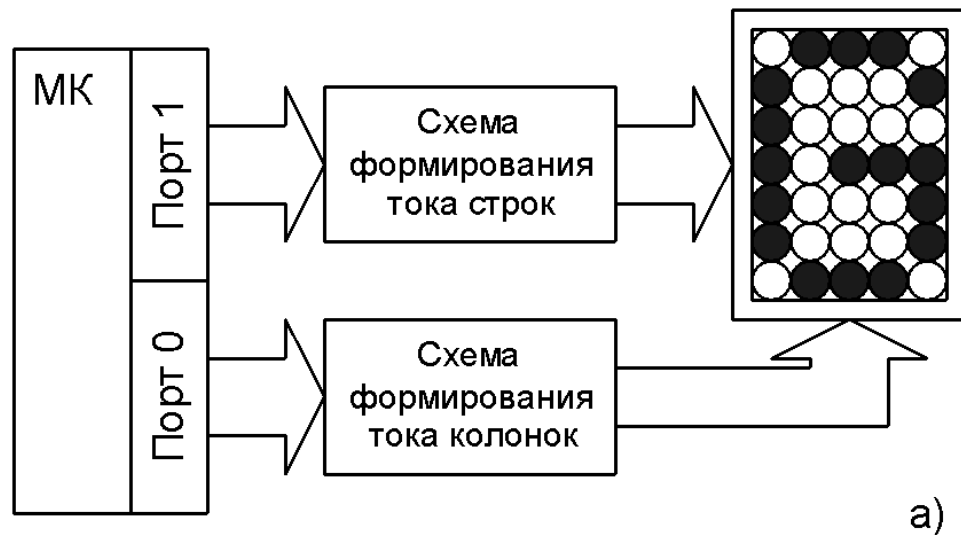


Рисунок 4.2 – Схема підключення ЗСІ до МК (а) і ланцюг протікання струму через світлодіод (б)

Для відображення символу на ЗСІ використовується принцип динамічної індикації по колонках. Байт індикації є кодом засвічення світлодіодів відповідної

колонки. Графічний образ символу набирається з послідовності п'яти байтів індикації шляхом перебору колонок.

Байт індикації записується в регістр порту 3 **PC_REG** за адресою 08002h: PC0 - біт 1 - го рядка, ..., PC6 - біт 7 - го рядка, PC7 = 0 (цей біт не впливає на роботу ЗСИ). Запис 0 в відповідний біт відповідає засвіченню даного світлодіода ЗСИ.

Вибір колонки ЗСИ організовується шляхом запису 1 в один з 5 - ти бітів PA0 - PA4 порту A **PA_REG** за адресою 08000h: 00001 - 1 - я колонка; ..., 10000 - 5 - я колонка («біжуча» 1). Біти PA5 - PA7 = 0 і не впливають на роботу ЗСИ.

2.Порядок виконання роботи

2.1 Розробити програму для виконання індивідуального завдання: вивести першу букву імені на ЗСИ

2.2 Ввести програму індивідуального завдання в персональний комп'ютер.

2.3 Завантажити програму в стенд. Переконавшись в правильному виконанні індивідуального завдання. При негативному результаті змінити програму і повторити її завантаження в стенд.

2.4 Роздрукувати лістинги правильно працюючої програми.

3. Зміст звіту

3.1 Короткі теоретичні відомості.

3.2 Текст програми.

3.4 Висновки по роботі.

4.Контрольні питання

4.1 Типи індикація

4.2 Організації знакосинтезуючої індикації у стенді

4.3. Особливості написання програми.

4.4. Вплив тривалості часу затримки на роботу програми.

ЛАБОРАТОРНА РОБОТА №5

Відображення інформації на рідкокристалічному індикаторі

Мета роботи: вивчити принцип організації і роботу РКІ - модуля.

1. Теоретична частина

Дисплеї, побудовані на основі світлодіодних індикаторів, мають ряд недоліків: відносно великий рівень споживаної потужності; низька контрастність зображення в умовах прямих сонячних променів.

РКІ - модулі - недороге і зручне рішення, що дозволяє заощадити час і ресурси при розробці МПС. Вони забезпечують відображення великого обсягу інформації при хорошій розрізненні і низькому енергоспоживанні.

Рідкими кристалами називають матеріал у вигляді довгих ланцюжків з дуже високою рухливістю. За рахунок цього в звичайному стані ці молекули розташовуються хаотично і рідкий кристал не прозорий. Якщо помістити рідкий кристал в електричне поле, то молекули орієнтуються щодо лінії напруженості поля і рідкий кристал стає прозорим.

РКІ - модулі мають вбудований контролер управління і поділяються на два класи: цифрові 7 - ми сегментні дисплеї і алфавітно - цифрові знакосінтезуючі. Перший клас індикаторів містить одну або кілька рядків, що складаються з 7 - ми сегментних розрядів, які по розташуванню сегментів подібні світлодіодним. У знакосінтезуючих РКІ кожен з розрядів являє собою матрицю точок, за допомогою якої можуть відображатися не тільки цифри, але і букви, символи. Ці РКІ - модулі дорожчі порівняно з 7 - ми сегментними, але зате вони і більш інформативні.

У навчальному стенді використовується РКІ - модуль **BC1004A** фірми **Bolymin**, основу якого складає контролер **KS0066**. Цей контролер є аналогом контролера **HD44780** фірми **Hitachi**, який фактично є промисловим стандартом і широко застосовується при виробництві алфавітно-цифрових РКІ - модулів.

Контролер **HD44780** управляє 2-мя рядками по 40 символів в кожній, при матриці символу 5 x 7 точок. У 4-х малих модулях, через проблеми розводки струмоведучих доріжок, перша і друга рядки цих модулів з'єднані з іншими рядками наступним чином: третій рядок є продовженням першого рядка, а четверта - другий. Контролер, за принципом динамічної індикації, циклічно виробляє оновлення інформації на РКІ. РКІ організований як матриця, що складається з 16-ти рядків (два рядки символів по 5 x 7 точок плюс символ курсору) по 200 сегментів в кожній (кожен рядок налічує 40 символів).

У контролера **HD44780** існує набір внутрішніх прапорів, що визначають режими роботи різних елементів контролера (табл. 5.1).

Таблиця 5.1 - Прапори, що керують роботою контролера **HD44780**

Прапор	Найменування прапора і його вплив на режим роботи контролера
I/D	режим зміщення лічильника адреси АС: 0 - зменшення, 1 - збільшення
S	прапор режиму зсуву вмісту екрана: 0 - зсув екрану не проводиться, 1 - після запису в DDRAM чергового коду екран зсувається в напрямку, який визначається прапором I / D : 0 - вправо, 1 - вліво. При зсуві не проводиться зміна вмісту DDRAM , змінюються тільки внутрішні показники розташування видимого початку рядка в DDRAM
S/C	прапор - команда, яка виробляє разом з прапором R / L операцію зсуву вмісту екрана (так само, як і в попередньому випадку, без змін в DDRAM) або курсора. Визначає об'єкт зсуву: 0 - зсувається курсор, 1 - зсувається екран
R/L	прапор - команда, яка виробляє разом з прапором S / C операцію зсуву екрану або курсору. Уточнює напрямок зсуву: 0 - вліво, 1 - вправо
D/L	прапор, що визначає розрядність шини даних: 0 - 4 розряду, 1 - 8 розрядів
N	режим розгортки зображення на РКІ: 0 - один рядок, 1 - два рядки
F	розмір матриці символів: 0 - 5 x 8 точок, 1 - 5 x 10 точок
D	наявність зображення: 0 - вимкнено, 1 - включено
C	курсор у вигляді підчерка: 0 - вимкнено, 1 - включений
B	курсор у вигляді мерехтливого знакоместа: 0 - вимкнено, 1 - включений

У табл. 5.2 наведені значення керуючих прапорів безпосередньо після подачі на РКІ - модуль напруги живлення. Перевизначення значень прапорів проводиться спеціальними командами, записуваними в регістр IR, при цьому комбінації старших бітів визначають групу прапорів або команду, а молодші містять власне прапори.

Таблиця 5.2 Значення керуючих прапорів після подачі живлення

Прапор	Режим роботи після включення живлення
I/D = 1:	режим збільшення лічильника на 1
S = 0:	без зсуву зображення
D/L = 1:	8-ми розрядна шина даних
N = 0:	режим розгортки одного рядка
F = 0:	символи з матрицею 5 x 8 точок
D = 0:	відображення вимкнено
C = 0:	курсор у вигляді підчерка вимкнений
B = 0:	курсор у вигляді мерехтливого знакоместа вимкнений

Список керівників комбінацій бітів регістра **IR** і виконувані ними команди наведені в табл. 4.3. Так як на момент включення РКІ - модуль нічого не відображає (прапор **D** = 0), то для того, щоб вивести будь-якої текст необхідно, як мінімум, включити відображення, встановивши прапор **D** = 1. Приклад послідовності команд для ініціалізації РКІ - модуля: 38h, 0Ch, 06h. Тут: команда 38h встановлює режим відображення 2-х рядків з матрицею 5 x 8 точок і роботу з 8-ми розрядної шиною даних; 0Ch включає відображення на екрані РКІ - модуля, без відображення курсорів; 06h встановлює режим автоматичного переміщення курсору зліва - направо після виведення кожного символу.

Таблиця 5.3 - Команди РКІ - модуля (код для регістра **IR**)

Назва команди	Двійковий код операції								Н – код	Операція	Т _{ком} , мс
	D7	D6	D5	D4	D3	D2	D1	D0			
Очищення дисплея	0	0	0	0	0	0	0	1	01	У всі осередки заноситься код 20h (пропуск), зрушений текст / курсор встановлюються в початкову позицію	1,2 7
Повернення тексту / курсора в початкову позицію	0	0	0	0	0	0	1	x	02 - 03	Зрушений текст / курсор повертаються в початкову позицію	
Установка напрямку зсуву курсору, включення зміщення тексту	0	0	0	0	0	1	I/D	S	04 - 07	I/D = 1 - курсор при введенні рухається вправо, I / D = 0 - вліво; S = 1 - зміщення тексту включено, S = 0 - вимкнено	0,0 4
Включення / вимикання дисплея / курсора, установка типу курсора	0	0	0	0	1	D	C	B	08 - 0F	D = 1 – дисплей включений; C = 1 - курсор включений; B = 1 - миготіння курсору включено	
Зсув курсора / тексту	0	0	0	1	S/C	R/L	x	x	10 - 1F	S/C = 1/0 – змістити текст / курсор; R / L = 1/0 - змістити вправо / вліво	
Установка параметрів модуля	0	0	1	DL	N	F	x	x	20 - 3F	DL = 1/0 – розрядність шини даних 8/4 біта; N = 1/0 - рядків на індикаторі 2/1; F = 1/0 - матриця 5x10 / 5x7 точок	
Установка адреси пам'яті знакогенератора	0	1	AG	AG	AG	AG	AG	AG	40 - 7F	Установка адреси поточної комірки пам'яті знакогенератора CGRAM	
Установка адреси пам'яті даних	1	AD	AD	AD	AD	AD	AD	AD	80 - FF	Установка адреси поточної комірки пам'яті даних DDRAM	

Примітка: х - довільне (0 або 1) стан.

Контролер **HD44780** підтримує як операції запису, так і операції читання. Читання регістра **DR** призводить до завантаження вмісту **DDRAM** або **CGRAM**, в залежності від поточного режиму, при цьому курсор зміщується на одну позицію, як і під час запису. Читання регістра **IR** дає нам 8 значущих розрядів: в 7 - ми молодших міститься поточне значення лічильника **AC** (7 розрядів, якщо адресується **DDRAM**, або 6 - якщо **CGRAM**), а в старшому - прапор зайнятості **BF**. Цей прапор має значення 1 коли контролер зайнятий і 0 - коли вільний. Необхідно враховувати, що більшість операцій, виконуваних контролером, займають чимало часу. Для більшої частини команд їх час виконання **ТКОМ** становить близько 40 мкс, а для деяких - доходить до одиниць мілісекунд. Тому цикл очікування зняття прапора **BF** повинен обов'язково бути присутнім в програмах драйвера **PKI** - модуля і передувати здійсненню будь-якої операції (природно, крім операції перевірки прапора **BF**).

Вивід на екран символу проводиться записом його коду в регістр **DR**. При цьому символ розміщується в **DDRAM** за поточним адресою, що вказується **AC**, а значення **AC** збільшується або зменшується на 1. Щоб зробити переустановку курсору на потрібну позицію, необхідно присвоїти **AC** відповідне значення (див. Табл. 5.4).

Таблиця 5.4 - Адресація символів (h - код) **PKI** - модуля

Номер розряду	0	1	2	3	4	5	6	7	8	9
Номер рядка										
1	80	81	82	83	84	85	86	87	88	89
2	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
3	8A	8B	8C	8D	8E	8F	90	91	92	93
4	CA	CB	CC	CD	CE	CF	D0	D1	D2	D3

Набір символів PKI - модуля. Необхідно враховувати, що контролери, що встановлюються на PKI - модулі, можуть мати різні набори символів, причому це може залежати як від виробника контролера, так і від модифікації даної конкретної моделі.

На рисунку 5.1 наведено набір символів цього контролера.

СТАРША ЦИФРА

М О Л О Д Ш А Ц И Ф Р А																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				Ø	Q	P	`	Р			Б	Ю	Ч	.	Д	У
1			!	1	A	Q	a	ч			Г	Я	Ш	і	Ц	У
2			"	2	B	R	b	р			Ё	б	ъ	и	Щ	У
3			#	3	C	S	c	s			Ж	В	Ы	и	Д	У
4			\$	4	D	T	d	t			Э	Г	Ь	Ъ	Ф	И
5			%	5	E	U	e	u			И	ё	э	ж	Ц	У
6			&	6	F	V	f	v			Й	ж	Ю	Ъ	Щ	У
7			'	7	G	W	g	w			Л	З	Я	І	'	Е
8			<	8	H	X	h	x			П	И	«	И	"	Е
9			>	9	I	Y	i	y			У	й	»	↑	~	Е
A			*	:	J	Z	j	z			Ф	К	„	↓	é	Е
B			+	;	K	[	k	и			Ч	л	”	и	с	Е
C			,	<	L	ф	l	и			Ш	м	и	и	й	У
D			-	=	M	]	m	и			Ь	н	и	и	и	и
E			.	>	N	^	n	е			Ы	п	и	и	и	и
F			/	?	O	_	o	е			Э	т	и	и	и	и

Рисунок 5.1 – Набір символів контролера.

Примітка:

- код 20h - «пробіл»;
- коди 00h - 0Fh - символи, визначені користувачем.

З допустимих для розміщення в **DDRAM** кодів, символи з кодами 00h ... 07h (і їх дублікат з кодами 08h ... 0Fh) мають спеціальне призначення - це переобумовленої символи, графічне зображення яких може призначити сам споживач, записавши відповідну інформацію в області **CGRAM** . У режимі роботи з матрицею 5 x 7 точок для програмування доступні 8 переобумовлених символів. Для кодування матриці використовуються горизонтально "укладені" байти, п'ять молодших бітів, яких несуть інформацію про рисунок (причому 1 означає, що сегмент буде включений), 4-й розряд кожного з 8-ми байтів матриці визначає ліву колонку символу, а 0-й - праву. Старші три біта не використовуються, так само, як і старший байт, зі складових повну область матриці символу (8 байтів). Матриця програмованих символів допускає використання повної висоти рядка (8 рядків для режиму 5 x 7), тобто можна розміщувати точки в області підкреслює курсору).

Щоб визначити власний символ, необхідно встановити лічильник **AC** на адресу початку матриці потрібний символ в **CGRAM** - 00h, 08h, 10h і т.д. - провести перезапис всіх байтів матриці, починаючи з верхнього рядка. Після цього, записавши в **DDRAM** код запрограмованого символу: 00h, 01h, 02h, на екрані у відповідному місці буде відображатися перевизначення символ.

Ініціалізація PKI - модуля. Виробник контролера рекомендує виконувати наступну послідовність дій для ініціалізації. Витримати паузу не менше 15 мс між встановленням робочої напруги живлення (> 4,5 В) і виконанням будь-яких операцій з контролером. Першою операцією виконати команду, що вибирає розрядність шини (це повинна бути команда 30h незалежно від того, який розрядності інтерфейс ви збираєтеся використовувати в подальшому), причому перед виконанням цієї операції не перевіряти значення прапора **BF**. Далі знову витримати паузу не менше 4,1 мс і повторити команду вибору розрядності шини, причому перед подачею команди знову не виробляти перевірку прапора **BF**. Наступним кроком необхідно знову витримати паузу, на цей раз 100 мкс, і в третій раз повторити команду встановлення розрядності шини, знову без перевірки **BF**. Ці три операції є не започатковано і покликані вивести контролер в початковий

режим роботи (тобто перевести в режим роботи з 8-ми розрядної шиною) з будь-якого стану.

Потім, без витримування пауз, але з перевіркою прапора **BF**, виконується ініціалізація режимів роботи РКІ - модуля. Формується ініціалізуюча послідовність установки прапорів (табл. 5.1), відповідно до команд табл. 5.3 (що містить, в тому числі команду вибору необхідної розрядності шини).

Програмування РКІ - модуля стенду. Відзначимо, що при роботі з РКІ - модулем УОС ВС 1004А попередня ініціалізація (пов'язана з включенням живлення) вже виконана системним контролером. На екран модуля виводиться назва фірми - виробника УОС - «OpenSystem». Тому програми для РКІ - модуля можна починати з команд, що визначають режими його роботи.

Адреси регістрів, які керують роботою РКІ - модуля:

- регістр команд **IR**: 08004h;
- регістр даних **DR**: 08005h.

2. Порядок виконання роботи

2.1 Вивчити принцип роботи РКІ - модуля і намалювати функціональну схему лабораторної роботи.

2.2 Розробити алгоритм і програму для виконання завдання, яке полягає в організації відображення на екрані РКІ - модуля такої інформації (для кожного студента):

ім'я

Прізвище

Число. Місяць. Рік народження.

Допускається скорочення, якщо інформація перевищує розмір рядка (більше 10 символів).

2.4 Ввести програму завдання в персональний комп'ютер.

2.5 Завантажити програму в стенд. Переконалися в правильному виконанні завдання. При негативному результаті змінити програму і повторити її завантаження.

2.6 Роздрукувати лістинги правильно працюючої програми.

3. Зміст звіту

3.1 Теоретична частина.

3.2 Формат зображення інформації завдання на екрані РКІ і послідовність кодів символів.

3.3 Тексти файлів програми завдання.

3.4 Висновки по роботі.

Контрольні питання

4.1 РКІ, пристрій принцип дії.

4.2 Ініціалізація РКІ.

4.3. Набір символів РКІ - модуля.

4.4. Програмування РКІ - модуля стенду.

ЛАБОРАТОРНА РОБОТА №6

Формування аналогових сигналів на основі ЦАП

Мета роботи: вивчити особливості формування аналогових сигналів в МПС на основі ЦАП.

1. Теоретична частина

Цифро - аналогове перетворення. Цифро-аналогові перетворювачі (ЦАП) виробляють напруга або струм, прямо пропорційні вхідного коду. ЦАП застосовується в МПС для формування вихідних аналогових сигналів.

Застосовуються в основному два методи перетворення:

- 1) підсумовування однієї еталонної величини вагою в один квант;
- 2) підсумовування еталонних величин, ваги яких залежать від номера розряду.

У підсумовуванні беруть участь тільки ті еталонні величини, для яких у відповідному розряді вхідного двійкового коду встановлена одиниця.

Робота ЦАП, при вхідному двійковому паралельному коді, може бути описана наступним виразом

$$X = \frac{P}{2^L - 1} (a_0 \cdot 2^0 + a_1 \cdot 2^1 + a_2 \cdot 2^2 + \dots + a_{L-1} \cdot 2^{L-1})$$

де **X** – вихідна аналогова величина (струм або напруга);

a_i – значення відповідного двійкового розряду, $i = 0, 1, \dots, L - 1$;

P - опорний сигнал (струм або напруга);

L – розрядність вхідного двійкового коду.

Зазвичай, в ЦАП з опорного напруги $U_{оп}$ формуються еталонні величини, відповідні вазі розряду вхідного двійкового коду, які потім сумуються для одиничних значень розрядів і формуються дискретні значення вихідної аналогової величини.

ЦАП класифікують за рядом ознак:

- 1) за способом формування вихідної напруги: з підсумовуванням напруг, розподілом напружень і підсумовуванням струмів;
- 2) у зв'язку зі вихідного сигналу: з струмовим виходом або виходом по напрузі;
- 3) по полярності вихідної напруги: однополярні і двополярного.

Для перетворення вхідного двійкового коду в аналоговий сигнал зазвичай з опорного напруги $U_{оп}$ формуються струми, пропорційні ваг розрядів коду, і потім сумуються ті з них, які відповідають ненульовим розрядам коду. Для цього в інтегральних ЦАП використовуються структури з:

- зваженими резисторами в ланцюгах емітерів;
- зваженими резисторами в ланцюгах навантаження;
- матрицею R-2R в ланцюгах емітерів транзисторів - джерел струмів;
- вихідний матрицею R-2R.

Параметри ЦАП. Всі параметри ЦАП можуть бути розділені на дві групи: статичні і динамічні.

До статичних параметрів належать: статична передавальна характеристика, число розрядів L вхідного двійкового коду, опорна напруга $U_{оп}$, роздільна здатність (квант) $h = U_{оп} / (2^L - 1)$, номінальний вихідний струм, напруга зсуву нуля, похибка повної шкали, похибка лінійності, диференціальна нелінійність. Похибки ЦАП можуть бути виражені у відсотках, відносних одиницях, а також в частках кванта.

Динамічні параметри ЦАП визначаються зі зміни (стрибкоподібного, від коду все «0» до коду все «1») вхідного двійкового коду: час встановлення вихідного сигналу $T_{уст}$; швидкість наростання вихідної напруги..

У складі УОС є мікросхема восьмирозрядного ЦАП AD7801 (DD2 плати розширення) з паралельним інтерфейсом. Основні параметри ЦАП наведені в табл. 6.1.

Таблиця 6.1 - Основні параметри ЦАП AD7801 (відповідно до схеми його включення)

№ п/п	Параметр	Значення параметра	Одиниці виміру	Примітка
Статичні параметри				
1	Напруга живлення, V_{CC}	5	В	
2	Амплітуда вихідної напруги, V_{OUT}	5	В	$V_{CC} = V_{DD} = \text{REFIN}$
3	Вихідний опір, $R_{ВЫХ}$	40	Ом	
4	Вихідний струм, $I_{ВЫХ}$	14	mA	Максимальне значення
5	Зміщення нуля	3	ЕМР	При + 25 °С
6	Похибка лінійності	± 1	ЕМР	
7	Похибка повної шкали	- 0,75	ЕМР	
8	Диференціальна нелінійність	± 1	ЕМР	
9	Дрейф нуля	100	$\mu\text{В}/^{\circ}\text{С}$	Типове значення
Динамічні параметри				
10	Швидкість наростання вихідної напруги	7,5	В/ $\mu\text{С}$	Типове значення
11	Максимальний час встановлення $T_{уст}$	2	$\mu\text{С}$	

Для спостереження вихідного сигналу ЦАП необхідно підключити осцилограф до ВНС роз'єму і замкнути перемичку J4 (при розімкнутих перемичках J1 і J3).

Функціональна схема лабораторної роботи дана на рис. 6.1.

Доступ до ЦАП здійснюється як до осередку зовнішньої пам'яті даних за адресою 0F000h. Зауважимо, що сигнал на виході ЦАП встановлюється не миттєво, а протягом деякого часу $T_{уст}$. Цей факт необхідно враховувати при розробці прикладних програм для ЦАП.

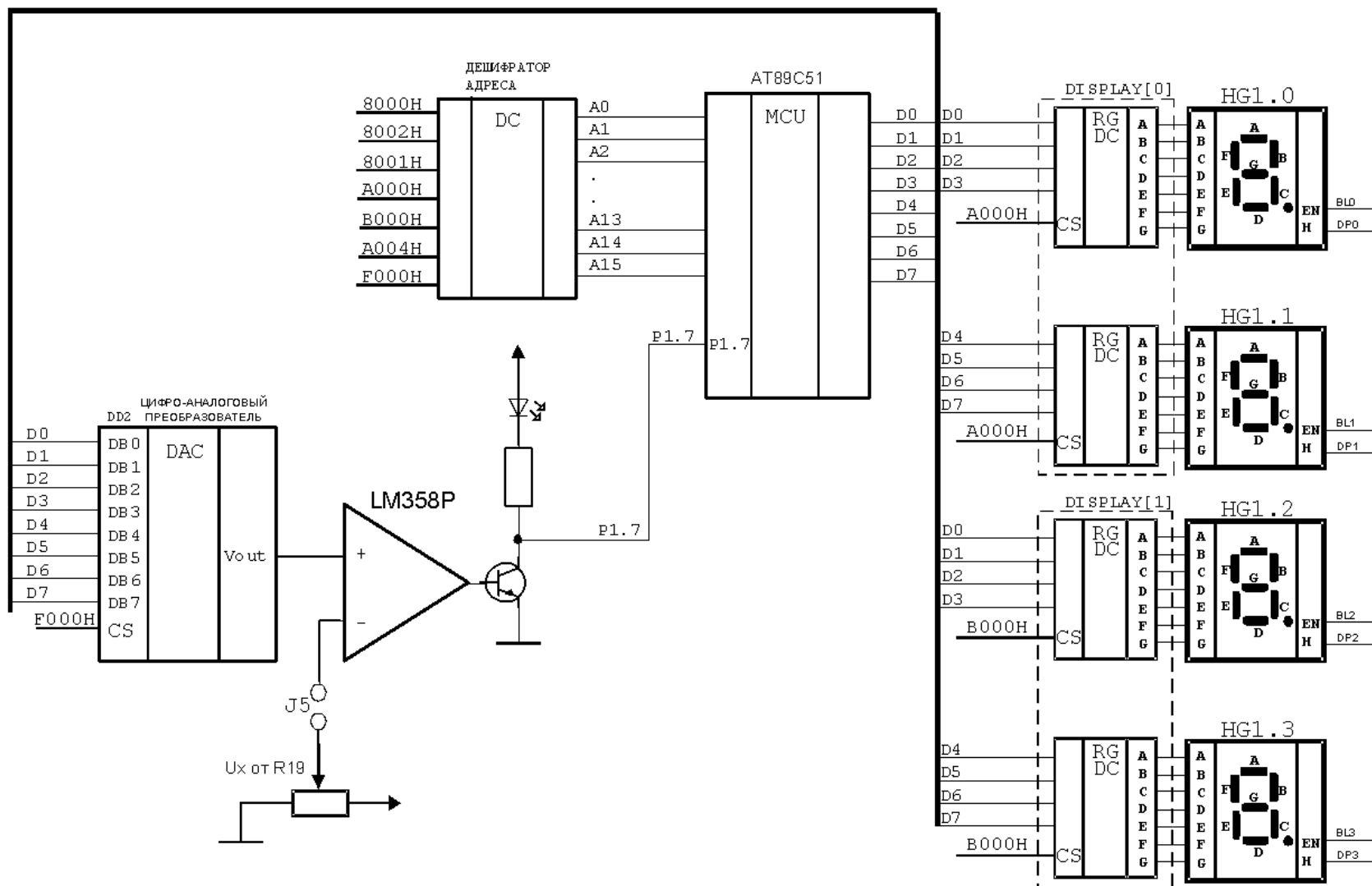


Рисунок 6.1 – Функціональна схема лабораторних робіт № 6 і №7

Приклад. Сформувати на виході ЦАП пилкоподібний сигнал

```

ORG 0
BEGIN:
movA,#0FFh      ; Записати в А максимальне (початкова) значення коду
M:
    INC A        ; Формування чергового значення коду
movDPTR,#0F000h ; Запис коду в
movx @DPTR,A    ; регістр даних ЦАП
    CALL TIME    ; Затримка на час формування
                  ; вихідного сигналу ЦАП
    CJNE A,#0FFh,M ; Якщо код не досяг максимуму (255 або 0FFh),
                  ; то продовжити формування "пили"
JMP BEGIN      ; Повторити виконання програми
; підпрограма затримки
TIME:
    movR2,#004h  ; Затримка
    djnzR2, $    ; на 21  $\mu$ C
    RET          ; Повернення з підпрограми
END

```

Використовуючи табличний спосіб завдання функції та процедуру тимчасової затримки **TIME** (лабораторна робота №2), можна в МПС реалізувати формування практично будь-якого виду керуючого аналогового сигналу.

2. Порядок виконання роботи

2.1 Вивчити теоретичні відомості і намалювати функціональну схему лабораторної роботи.

2.2 Набрати і виконати програму прикладу.

2.3 Встановити відповідну перемичку і, користуючись осцилографом, зняти осцилограму пилкоподібного сигналу на виході ЦАП. Записати значення кванта за часом ΔT , періоду T і амплітуди $U_{\text{вих}}$ цього періодичного сигналу.

2.4 Розробити алгоритм і програму для виконання індивідуального завдання (видається викладачем). Розрахувати константи для кванта за часом ΔT і таблиці значень заданої функції.

2.5 Ввести програму індивідуального завдання в персональний комп'ютер.

2.6 Завантажити програму в стенд. Переконатися в правильному виконанні індивідуального завдання. При негативному результаті змінити програму і повторити її завантаження в УОС.

2.7 Роздрукувати лістинги правильно працюючої програми.

3. Зміст звіту

3.1 Коротко описати теоретичні положення лабораторної роботи.

3.2 Тексти файлів програм прикладу і вимірювання часу $T_{\text{уст}}$ ЦАП.

3.3 Рисунки осцилограм пилкоподібної напруги.

3.4 Тексти файлів програми індивідуального завдання та осцилограми.

3.5 Висновки по роботі.

4. Контрольні питання

4.1 Перерахуйте основні статичні параметри ЦАП.

4.2 Назвіть основні динамічні параметри ЦАП.

4.3 Що таке квантування за рівнем, за часом, що дозволяє здатність ЦАП?

4.4 Що таке «напруга зсуву нуля» ЦАП?

4.5 Поясніть, чому за допомогою ЦАП в МПС може бути реалізована будь-яка функція управління?

ЛАБОРАТОРНА РОБОТА №7

Аналого-цифрове перетворення в МПС

Мета роботи: вивчити особливості організації перетворення аналогових сигналів в двійковий код в МПС.

1. Теоретична частина

Аналого-цифрове перетворення. Аналогово-цифрові перетворювачі (АЦП) застосовуються в МПС для введення даних з аналогових датчиків. В МПС часто АЦП виконується на базі ЦАП. Схема цього АЦП наведена на рис. 7.1. Вхідний код з порту **P0** перетворюється в аналоговий сигнал за допомогою ЦАП. Напруга з виходу ЦАП V_{out} надходить на один з входів компаратора **К**. На інший вхід, якого подається вимірювана напруга U_x . У момент рівності напруга компаратор **К** формує сигнал **FC**, який свідчить про закінчення циклу перетворення.

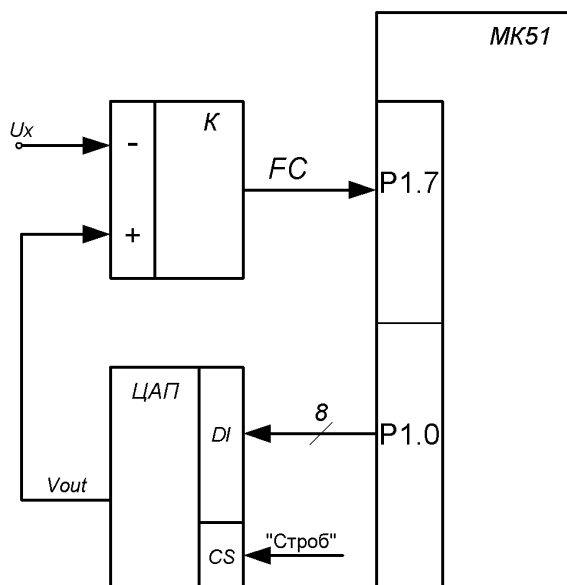


Рисунок 7.1 - Схема АЦП на основі ЦАП

АЦП побудований на мікросхемі восьмирозрядного ЦАП AD7801 і операційному підсилювачі LM358 як компаратор. Стан компаратора можна зчитувати з лінії порту P1.7 МК51. Закінчення циклу перетворення ($P1.7 = 0$)

сигналізується свіченням світлодіода, підключеного до виходу компаратора. Доступ до ЦАП здійснюється як до осередку зовнішньої пам'яті даних за адресою 0F000h. При установці перемички J5, вимірювана напруга U_x формується змінним резистором R19. Без цієї перемички, джерелом вимірюваної напруги U_x може бути зовнішній сигнал, який підключається до клемника X2 плати розширення.

Процес перетворення напруга U_x в двійковий код можна спостерігати на екрані осцилографа, підключеного до BCD гнізда (J2) плати розширення, якщо встановити перемичку J4 (перемички J1, J3 необхідно відключити).

2. Порядок виконання роботи

2.1 Вивчити теоретичні відомості і намалювати функціональну схему лабораторної роботи.

2.2 Набрати і виконати програму.

2.3 Встановити відповідні перемички і, користуючись осцилографом, зняти осцилограму процесу перетворення на виході ЦАП.

2.4 Роздрукувати лістинги правильно працюючої програми.

3. Зміст звіту

3.1 Коротко описати теоретичні положення лабораторної роботи.

3.2 Лістинг програми.

3.3 Висновки по роботі.

4. Контрольні питання

4.1 Перерахуйте основні статичні та динамічні параметри АЦП.

4.2 Нарисуйте осцилограми процесу поразрядного зрівноважування і поясніть особливості формування коду АЦП.

4.3 Нарисуйте осцилограми роботи інтегруючого АЦП і поясніть особливості формування коду АЦП.

СПИСОК ЛІТЕРАТУРИ

1. Каспер Э. Программирование на языке Ассемблера для микроконтроллеров семейства i8051. – М.: Горячая линия – Телеком, 2004. – 191 с.
2. Схемотехніка електронних систем: У 3 кн. Кн. 3. Мікропроцесори та мікроконтролери: Підручник/ В.І. Бойко, А.М. Гурій, В.Я. Жуйков та ін. – 2-ге вид. – К.: Вища шк., 2004. – 399 с.

Додаток А

Каф. ЕІ	Лабораторна робота №	група
СНУ ім. В.Даля	Назва роботи	Прізвище Ім'я