

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ**

До захисту допускається
Завідувач кафедри
комп'ютерних наук та інженерії
Скарга-Бандурова І. С.

« ____ » _____ 2019 р.

**ДИПЛОМНИЙ ПРОЕКТ БАКАЛАВРА
ПОЯСНЮВАЛЬНА ЗАПИСКА**

НА ТЕМУ:

**«ТЕСТУВАННЯ РІС-КОНТРОЛЕРІВ З ЗАСТОСУВАННЯМ
СЕРЕДОВИЩА MP-LAB»**

Освітньо-кваліфікаційний рівень «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

Науковий керівник роботи: _____ Кардашук В. С.
(підпис) (ініціали, прізвище)

Консультант з охорони праці: _____ Критська Я. О.
(підпис) (ініціали, прізвище)

Студент: _____ Буряченко С. С.
(підпис) (ініціали, прізвище)

Група: _____ КІ-15 бд

Сєвєродонецьк – 2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет інформаційних технологій та електроніки
Кафедра комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень бакалавр
Спеціальність 123 «Комп'ютерна інженерія»

«ЗАТВЕРДЖУЮ»
Завідувач кафедри
комп'ютерних наук та інженерії
д.т.н, доц. Скарга-Бандурова І. С.

“ _____ ” _____ 2019 року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Буряченку Сергію Станіславовичу

(прізвище, ім'я, по-батькові)

1. Тема проекту (роботи): «Тестування PIC-контролерів з застосуванням середовища MP-Lab» затверджена наказом по університету № 83/15.15 від «13» травня 2019 р.

2. Строк здачі студентом закінченого проекту (роботи): 10.06.2019 р.

3. Вихідні дані проекту (роботи): матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити):

1. Огляд PIC-контролерів та інструментальних засобів реалізації програми тестування.

2. Розроблення програми тестування PIC-контролерів.

3. Компіляція розробленої програми.

4. Завантаження програми тестування в пам'ять PIC-контролера.

5. Охорона праці.

5. Перелік графічного матеріалу (з точною назвою обов'язкових креслень):

Електронні плакати

6. Консультанти роботи, з вказівкою розділів, що до них відносяться

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Основна частина	к.т.н, доц. Кардашук В. С.		
Охорона праці	ст. викл. Критська Я. О.		

7. Дата видачі завдання _____

Керівник _____ Кардашук В. С.
(підпис)

Завдання до виконання прийняв _____ Буряченко С. С.
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітки
1.	Отримання завдання, збір матеріалів	18.05.19- 24.05.19	
2.	Огляд літератури й обґрунтування необхідності розроблення	25.05.19–28.05.19	
3.	Огляд інструментальних засобів реалізації програми тестування PIC контролерів	29.05.19– 28.05.19	
4.	Створення проекту в програмному середовищі MPLAB IDE	28.05.19–31.05.19	
5.	Завантаження програми тестування в пам'ять мікроконтролера PIC	03.06.19– 04.06.19	
6.	Оформлення пояснювальної записки	05.06.19– 08.06.19	
7.	Підготовка та подання роботи до захисту	09.06.19– 10.06.19	

Студент

(підпис)

Науковий керівник

(підпис)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту: 82 с., 33 рис., 3 табл., 24 джерел посилань.

Мета роботи: розроблення діагностичної програми тестування внутрішніх вузлів та портів мікроконтролера PIC 16C84A.

З метою розроблення програми діагностування проведено огляд RISC-процесора із симетричною системою команд Розглянуті особливості гарвардської архітектури, що базується на концепції роздільних шин і областей пам'яті для даних і для команд.

Проведено огляд безкоштовного програмного пакету MP-LAB IDE фірми Microchip Technology Inc, що об'єднує в собі текстовий редактор для створення HEX-файлів та програмний емулятор МК різних типів.

На основі проведено огляду визначені задачі розроблення програми діагностування портів та внутрішніх вузлів МК PIC.

Візуалізація результатів тестування здійснюється виводом інформації в порт МК та запалюванням відповідних світлодіодів, що свідчить про працездатність вузлів та порту МК. Проведена компіляція розробленої програми. Розглянуті робота в симуляторі, основні засоби налаштування MPLAB IDE та асемблювання програми.

Ключові слова: *діагностична програма, мікроконтролер PIC, порт, система команд, компіляція, пам'ять EEPROM.*

Умови одержання роботи:

93406. м. Сєвєродонецьк, пр-кт Центральний, 59а, СНУ ім. В. Даля

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1 ОГЛЯД PIC КОНТРОЛЕРІВ ТА ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ ПРОГРАМИ ТЕСТУВАННЯ.....	8
1.1 Огляд мікроконтролерів PIC.....	8
1.2 Особливості архітектури PIC контролерів	10
1.3 Огляд інструментальних засобів реалізації програми тестування PIC контролерів	14
1.4 Висновки до розділу 1 та постановка задачі розроблення	15
РОЗДІЛ 2 РОЗРОБЛЕННЯ ПРОГРАМИ ТЕСТУВАННЯ PIC КОНТРОЛЕРІВ	17
2.1 Створення проекту в програмному середовищі MPLAB IDE	17
2.2 Розроблення програми тестування портів вводу-виводу МК PIC16F84	25
2.3 Компіляція розробленої програми	33
2.4 Робота в симуляторі та налаштування програми	34
2.5 Основні засоби налаштування MPLAB IDE	39
2.6 Асемблювання програми	42
2.7 Висновки до розділу 2	43
РОЗДІЛ 3 ЗАВАНТАЖЕННЯ ПРОГРАМИ ТЕСТУВАННЯ В ПАМ'ЯТЬ МІКРОКОНТРОЛЕРА PIC	44
3.1 Пам'ять команд мікроконтролерів PIC. Технологія FLASH– EEPROM	44
3.2 Регістри і оперативна пам'ять мікроконтролера PIC16C84	45
3.3 Читання даних EEPROM мікроконтролерів PIC. Пам'ять даних EEPROM	46
3.4 Робота з мікроконтролерами PIC в програмі Pony Prog	47
3.5 Команди Асемблера для PIC	52

3.6 Висновки до розділу 3	54
РОЗДІЛ 4 ОХОРОНА ПРАЦІ	55
4.1 Загальні питання з охорони праці	55
4.2 Правові та організаційні основи охорони праці	56
4.3 Організаційно-технічні заходи з безпеки праці	56
4.4 Аналіз стану умов праці та вимоги до приміщення	56
4.5 Вимоги до організації робочого місця	57
4.6 Навантаження та напруженість процесу праці	58
4.7 Аналіз небезпечних та шкідливих факторів при роботі на персональному комп'ютері	59
4.8 Пожежна безпека	60
4.9 Електробезпека	61
4.10 Мікроклімат	62
4.11 Освітлення робочого місця	63
4.12 Шум, вібрація та електромагнітне випромінювання	65
4.13 Розрахунок захисного заземлення	66
4.14 Висновки до розділу 4	70
ВИСНОВКИ	72
ПЕРЕЛІК ПОСИЛАНЬ	73
ДОДАТОК А – Презентація	76

ВСТУП

Подальше зростання ступеня інтеграції дозволило розмістити в кристалі мікросхеми вже не окремі прості вузли або фрагменти пристроїв, а цілі пристрої і навіть цілі обчислювальні вузли. Це привело до створення мікроконтролера (МК) – виробу мікроелектроніки і обчислювальної техніки принципово нового класу, здатного вести обробку і зберігання інформації в одному або декількох корпусах мікросхем [1].

Використання МК у виробках не лише приводить до підвищення техніко-економічних показників (вартості, надійності, споживаної потужності, габаритних розмірів), але і дозволяє скоротити час розробки виробів і робить такими, що їх модифікуються, адаптивними. Використання МК в системах керування забезпечує досягнення високих показників ефективності при низькій вартості.

МК є ефективним засобом автоматизації всіляких об'єктів і процесів.

Можна вважати що МК – це комп'ютер, що розмістився в одній мікросхемі. Звідси і його основні привабливі якості: малі габарити; високі продуктивність, надійність і здатність бути адаптованим для виконання самих різних завдань.

МК окрім центрального процесора (ЦП) містить пам'ять і багаточисельні пристрої вводу/виводу: аналого-цифрові перетворювачі, послідовні і паралельні канали передачі інформації, таймери реального часу, широко-імпульсні модулятори (ШИМ), генератори програмованих імпульсів і так далі.

Його основне призначення – використання в системах автоматичного керування, вбудованих в різні пристрої: банківські термінали, мобільні телефони, музичні центри, пральні машини, системи охоронної сигналізації і т.п.

В даний час існує більше 10000 різних МК, що розрізняються сферою застосування, параметрами, вбудованими в кристал периферійними вузлами. Випуском МК займається більше десятка виробників.

МК сімейства PIC (Peripheral Interface Controller) компанії Microchip поєднують усі передові технології МК: електрично програмовані користувачем EEPROM, мінімальне енергоспоживання, високу продуктивність, добре розвинуту RISC-архітектуру, функціональну закінченість і мінімальні розміри. Широка номенклатура виробів забезпечує використання МК у пристроях, призначених для різноманітних сфер застосування.

РОЗДІЛ 1

ОГЛЯД РІС КОНТРОЛЕРІВ ТА ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ ПРОГРАМИ ТЕСТУВАННЯ

1.1 Огляд мікроконтролерів РІС

Перші МК компанії Microchip PIC16C5x з'явилися наприкінці 1980-х років і завдяки своїй високій продуктивності і низькій вартості склали серйозну конкуренцію 8-розрядним МК із CISC-архітектурою, які виготовлялися на той час.

Висока швидкість виконання команд у РІС-контролерах досягається за рахунок використання двошинної гарвардської архітектури замість традиційної одношинної фон-нейманівської. Гарвардська архітектура базується на наборі регістрів з розділеними шинами й адресними просторами для команд і даних. Усі ресурси МК, такі як порти вводу/виводу, комірки пам'яті і таймер, являють собою фізично реалізовані апаратні регістри [2].

МК РІС містять RISC-процесор із симетричною системою команд, що дозволяє виконувати операції з будь-яким регістром, використовуючи довільний метод адресації. Користувач може зберігати результат операції в самому регістрі-акумуляторі або в другому регістрі, який використовується для операції.

В даний час компанія Microchip випускає п'ять основних сімейств 8-розрядних RISC МК, сумісних знизу нагору за програмним кодом.

PIC12CXXX – сімейство МК, що випускаються в мініатюрному 8-вивідному виконанні. Ці МК випускаються як з 12-розрядною (33 команди), так і з 14-розрядною (35 команд) системою команд. Містять вмонтований тактовий генератор, таймер/лічильник, сторожовий таймер, схему управління перериваннями. У складі сімейства є МК з убудованим 8-розрядним чотирьохканальним АЦП. Здатні працювати при напрузі живлення до 2,5 В.

PIC16C5X – базове сімейство МК з 12-розрядними командами (33 команди), що випускається в 18-, 20- і 28-вивідних корпусах. Являють собою прості недорогі МК з мінімальною периферією. Здатність працювати при малій напрузі живлення (до 2 В) робить їх зручними для застосування в переносних конструкціях. До складу сімейства входять МК підгрупи PIC16HV5XX, здатні працювати безпосередньо від батареї в діапазоні напруг живлення до 15 В.

PIC16CXXX – сімейство МК середнього рівня з 14-розрядними командами (35 команд). Найбільш численне сімейство, що поєднує МК з різноманітними периферійними пристроями, до складу яких входять аналогові компаратори, аналогово-цифрові перетворювачі, контролери послідовних інтерфейсів SPI, USART і I2C, таймери-лічильники, модулі захоплення/порівняння, широтно-імпульсні модулятори, сторожові таймери, супервізорні схеми і так далі.

PIC17CXXX – сімейство високопродуктивних МК з розширеною системою команд 16-розрядного формату (58 команд), що працюють на частоті до 33 МГц, з об'ємом пам'яті програм до 16 Кслів. Крім великої периферії, 16-рівневого апаратного стека і векторної система переривань, майже всі МК цього сімейства мають убудований апаратний множник 8x8, що виконує операцію множення за один машинний цикл. Є одними із самих швидкодіючих у класі 8-розрядних МК.

PIC18CXXX – сімейство високопродуктивних МК з розширеною системою команд 16-розрядного формату (75 команд) і убудованим 10-розрядним АЦП, які працюють на частоті до 40 МГц. Містять 31-рівневий апаратний стек, вмонтовану пам'ять команд до 32 Кслів і здатні адресувати до 4 Кбайт пам'яті даних і до 2 Мбайт зовнішньої пам'яті програм. Розширене RISC-ядро МК даного сімейства оптимізоване під використання нового Сі-компілятора.

Більшість PIC-контролерів випускаються з однократно програмованою пам'яттю програм, з можливістю внутрішньосхемного програмування або

масочним ПЗП. Для мети налагодження пропонуються більш дорогі версії з ультрафіолетовим стиранням і Flash-пам'яттю. Повний список модифікацій PIC-контролерів, що випускаються, включає порядку п'ятисот найменувань. Тому продукція компанії перекриває майже весь діапазон застосувань 8-розрядних МК.

Найбільш розповсюдженими сімействами PIC-контролерів є PIC16CXXX і PIC17CXXX.

Застосування PIC-контролерів доцільне в нескладних приладах з обмеженим струмом споживання (автономні пристрої, прилади з живленням від телефонної лінії і т.п.). Завдяки малій кількості компонентів, використовуваних при побудові таких приладів, їхні розміри зменшуються, а надійність збільшується.

1.2 Особливості архітектури PIC контролерів

Архітектура базується на концепції роздільних шин і областей пам'яті для даних і для команд (гарвардська архітектура). Спрощена структурна схема МК підгрупи PIC16F8X наведена на рис. 1.1 [3].

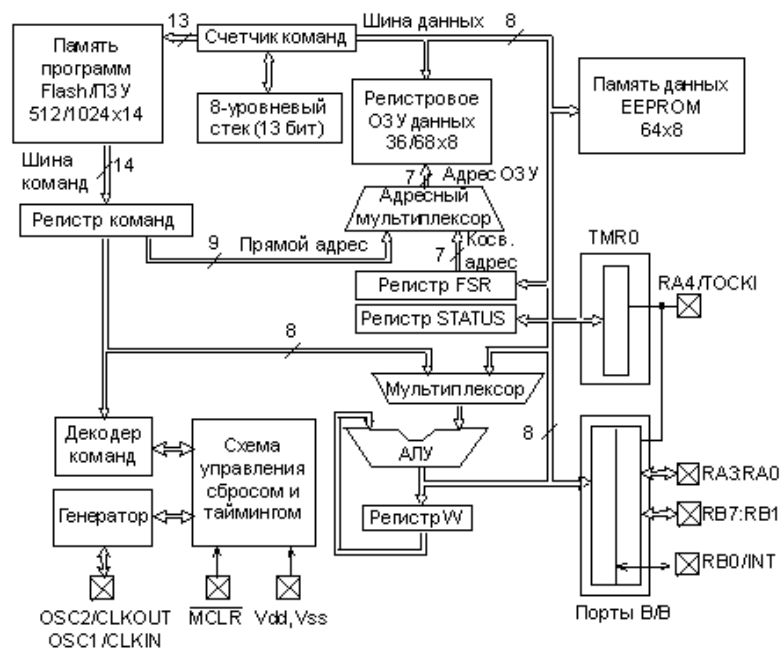


Рисунок 1.1 – Структурна схема МК підгрупи PIC16F8X.

Шина даних і пам'ять даних (ОЗП) – мають ширину 8 біт, а програмна шина і програмна пам'ять (ПЗП) мають ширину 14 біт. Така концепція забезпечує просту, але потужну систему команд, розроблену так, що бітові, байтові і реєстрові операції працюють з високою швидкістю і з перекриттям за часом вибірок команд і циклів виконання. 14-бітова ширина програмної пам'яті забезпечує вибірку 14-бітової команди в один цикл. Двоступінчастий конвеєр забезпечує одночасну вибірку і виконання команди. Усі команди виконуються за один цикл, крім команд переходів.

МК PIC16F83 і PIC16CR83 адресують 512x14 пам'яті програм, а PIC16F84 і PIC16CR84 - 1Кx14 пам'яті програм. Уся пам'ять програм є внутрішньою.

МК може прямо або побічно звертатися до реєстрів або пам'яті даних. Усі реєстри спеціальних функцій, включаючи лічильник команд, відображаються на пам'ять даних. Ортогональна (симетрична) система команд дозволяє виконувати будь-яку команду над будь-яким реєстром з використанням довільного методу адресації. Ортогональна архітектура і відсутність спеціальних винятків робить програмування МК групи PIC16F8X простим і ефективним.

Призначення виводів МК підгрупи PIC16F8X наведено в табл. 1.1.

Таблиця 1.1 – Призначення виводів МК підгрупи PIC16F8X.

Позначення	Тип	Буфер	Опис
OSC1/CLKIN	I	ТШ/ КМОН ²⁾	Вхід кристала генератора, RC-ланки або вхід зовнішнього тактового сигналу
OSC2/ CLKOUT	O	-	Вихід кристала генератора. У RC-режимі - вихід 1/4 частоти OSC1
/MCLR	I/P	ТШ	Сигнал ініціалізації/вихід програмуючої напруги. Ініціалізація низьким рівнем.
RA0	I/O	ТТЛ	PORTA - двонаправлений порт вводу/виводу RA4/TOSC1 може бути обраний як тактовий вихід таймера/лічильника TMR0. Вихід з відкритим стоком.
RA1	I/O	ТТЛ	
RA2	I/O	ТТЛ	
RA3	I/O	ТТЛ	
RA4 /TOSC1	I/O	ТШ	
RB0/INT	I/O	ТТЛ/ ТШ ¹⁾	PORTB - двонаправлений порт вводу/виводу. Може бути запрограмований у режимі внутрішніх активних навантажень на лінію живлення по усіх виводах. Вивід RB0/INT може бути обраний як зовнішній вхід переривання. Виводи RB4...RB7 можуть бути програмно надбудовані як входи переривання за зміною стану на кожному із входів. При програмуванні МК RB6 використовується як тактовий, а RB7 як вхід/вихід даних.
RB1	I/O	ТТЛ	
RB2	I/O	ТТЛ	
RB3	I/O	ТТЛ	
RB4	I/O	ТТЛ	
RB5	I/O	ТТЛ	
RB6	I/O	ТТЛ	
RB7	I/O	ТТЛ/ТШ 2) ТТЛ/ТШ 2)	
Vdd	P	-	Позитивна напруга живлення
Vss	P	-	Загальний провід (земля)

У таблиці використані наступні позначення: I - вхід; O - вихід; I/O - вхід/вихід; P - живлення; - - не використовується; ТТЛ - ТТЛ вхід; ТШ - вхід тригера Шмідта.

Примітки:

- 1) Буфер має вхід тригера Шмідта, коли конфігурується як вхід зовнішнього переривання.
- 2) Буфер має вхід тригера Шмідта, коли використовується в режимі послідовного програмування.
- 3) Буфер має вхід тригера Шмідта, коли конфігурується в режимі RC-генератора і КМОН-вхід в інших випадках.

МК містить 8-розрядний АЛП і робочий регістр W. АЛП є арифметичним модулем загального призначення і виконує арифметичні і логічні функції над умістом робочого регістра і кожного з регістрів контролера. АЛП може виконувати операції додавання, віднімання, зсуву і логічні операції. Якщо не зазначене інше, то арифметичні операції виконуються в додатковому двійковому коді.

У залежності від результату операції, АЛП може змінювати значення бітів регістра STATUS: C (Carry), DC (Digit carry) і Z (Zero).

Вхідна тактова частота, що надходить з виводу OSC1/CLKIN, ділиться усередині на чотири, і з неї формуються чотири циклічні тактові послідовності Q1, Q2, Q3 і Q4, які не перекриваються. Лічильник команд збільшується в такті Q1, команда зчитується з пам'яті програми і замикається в регістрі команд у такті Q4. Команда декодується і виконується протягом наступного циклу в тактах Q1...Q4. Схема тактування та виконання команди зображена на рис. 1.2.

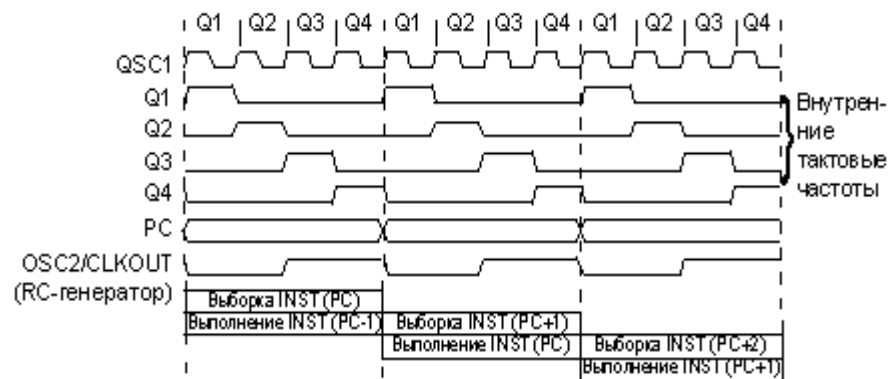


Рисунок 1.2 – Схема тактування і виконання команди

Цикл виконання команди складається з чотирьох тактів: Q1...Q4. Вибірка команди і її виконання поєднані в часі таким чином, що вибірка команди займає один цикл, а виконання - наступний цикл. Ефективний час виконання команди складає один цикл. Якщо команда змінює лічильник команд (наприклад, команда GOTO), то для її виконання буде потрібно два цикли, як показано на рис. 1.3.



Рисунок 1.3 – Вибірка команд

Цикл вибірки починається зі збільшення лічильника команд у такті Q1. У циклі виконання команди обрана команда замикається в регістр команд у такті Q1. Протягом тактів Q2, Q3 і Q4 відбувається декодування і виконання команди. У такті Q2 зчитується пам'ять даних (читання операнда), а запис відбувається в такті Q4.

1.3 Огляд інструментальних засобів реалізації програми тестування PIC контролерів

Для програмування МК сімейства PIC існує багато засобів розробки, проте, найбільш популярним, поза сумнівом, є програмний пакет MP-LAB IDE. Є ряд причин такої популярності – це безкоштовний програмний пакет, розроблений фірмою *Microchip Technology Inc*, який об'єднує в собі текстовий редактор для створення HEX-файлів та програмний емулятор МК різних типів [4].

З сайту компанії **Microchip Technology Inc** програма MPLAB доступна для скачування за адресою: <http://www.microchip.ru>

Середовище налаштування є програмним продуктом, що поширюється безкоштовно. На сайті компанії доступні оновлення програми, що викликані появою нових МК, а також поліпшенням її роботи. Установка програми проходить в стандартному для таких додатків режимі. Шлях установки програми: C:\Program Files\Microchip.

MP-LAB це інтегроване середовище проектування для PIC контролерів, тобто, набір програм, які дозволяють програмістові, з максимальним комфортом, складати, налагоджувати і оптимізувати текст програми, а також створювати (після цього) HEX - файл програми ("прошивку"). MP-LAB також підтримує всі типи PIC контролерів, за винятком самих новітніх розробок (їх можна довантажити) і істотно полегшує роботу програміста за рахунок поєднання декілька, необхідних для роботи програміста, функцій.

Виробники програматорів і апаратних засобів також приділяють увагу PIC-контролерам. Випускаються як спеціалізовані програматори, такі як PICPROG, що програмує майже весь спектр PIC МК, так і універсальні: UNIPRO і STEPX, які підтримують найбільш відомі версії PIC-контролерів.

1.4 Висновки до розділу 1 та постановка задачі розроблення

В першому розділі дипломного проєкту проведений огляд мікроконтролерів PIC фірми Microchip з метою розроблення програми діагностування портів та внутрішніх вузлів мікроконтролера. Зазначено, що МК PIC містять RISC-процесор із симетричною системою команд, що дозволяє виконувати операції з будь-яким регістром, використовуючи довільний метод адресації. Виборка наступної команди з пам'яті виконується одночасно з виконанням попередньою, що забезпечує високу швидкодію роботи МК. Розглянуті особливості архітектури, що базується на концепції роздільних шин і областей пам'яті для даних і для команд (гарвардська архітектура).

Проведено огляд безкоштовного програмного пакету MP-LAB IDE фірми *Microchip Technology Inc*, який об'єднує в собі текстовий редактор для створення HEX-файлів та програмний емулятор МК різних типів.

На основі проведено огляду визначені задачі розроблення програми діагностування портів та внутрішніх вузлів МК PIC.

Для розроблення програмного забезпечення обрано МК PIC16F84A фірми Microchip, що має наступні характеристики:

- тактова частота – 20МГц;
- продуктивність – 5 MIPS (млн. операцій в секунду);
- технологія пам'яті –FLASH;
- розмір пам'яті програм – 1750 байт;
- розмір пам'яті даних EEPROM – 64 байта;
- розмір оперативної пам'яті даних – 68 байт;
- 1 таймер на 8 біт;
- кількість ліній вводу/виводу – 13;
- температурний режим роботи – від – 40°С до +85°С.

Мікросхеми цієї серії відносяться до базового класу МК PIC. Для програмування на мові асемблера використовується 37 команд. Кожна команда представляє собою 14-розрядне слово, що містить поле коду операції OP CODE і поле операндів. Система команд включає в себе команди роботи з байтами, команди роботи з бітами, команди управління і операції з константами.

Всі команди виконуються протягом одного командного циклу, крім наступних двох випадків:

- перехід по перевірці умови, якщо результат перевірки умови – істина;
- зміна лічильника команд як результат виконання команди.

На рисунку 1.4 наведена нумерація контактів МК PIC16F84A.

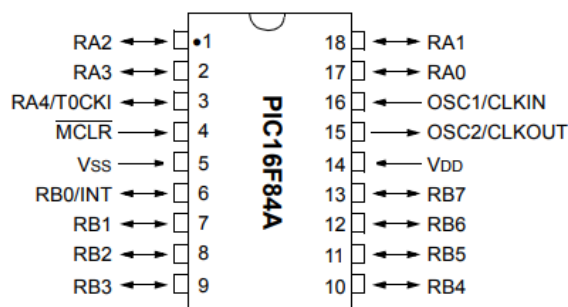


Рисунок 1.4 – Нумерація контактів МК PIC16F84A

РОЗДІЛ 2

РОЗРОБЛЕННЯ ПРОГРАМИ ТЕСТУВАННЯ PIC КОНТРОЛЕРІВ

2.1 Створення проекту в програмному середовищі MPLAB IDE

При першому запуску програмного середовища MPLAB налаштування міняти не треба. Розробники MPLAB встановили оптимальний варіант налаштування [5].

Після запуску даної програми отримуємо Діалогове вікно Майстра налаштування проекту (рис. 2.1).

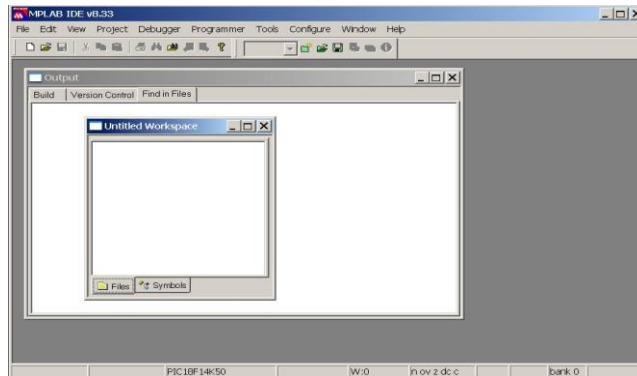


Рисунок 2.1 – Діалогове вікно Майстра налаштування проекту

Ліпше за все проект створити за допомогою Project Wizard. Для запуску помічника створення проекту потрібно вибрати Project>Project Wizard (рис. 2.2).

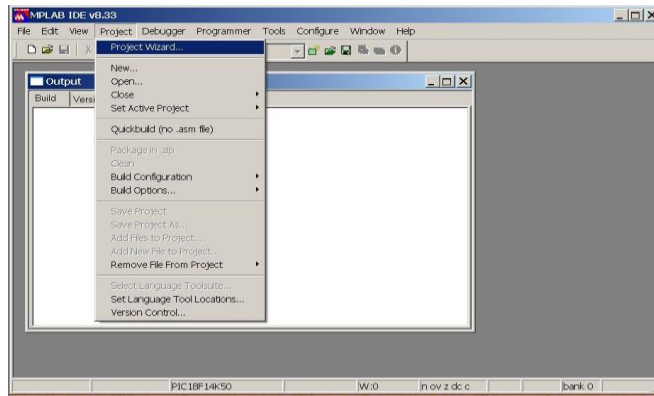


Рисунок 2.2 – Діалогове вікно виклику помічника створення проектів
Далі з'являється вікно створити новий проект. Для створення нового проекту необхідно вибрати Далі (Next) (рис. 2.3).

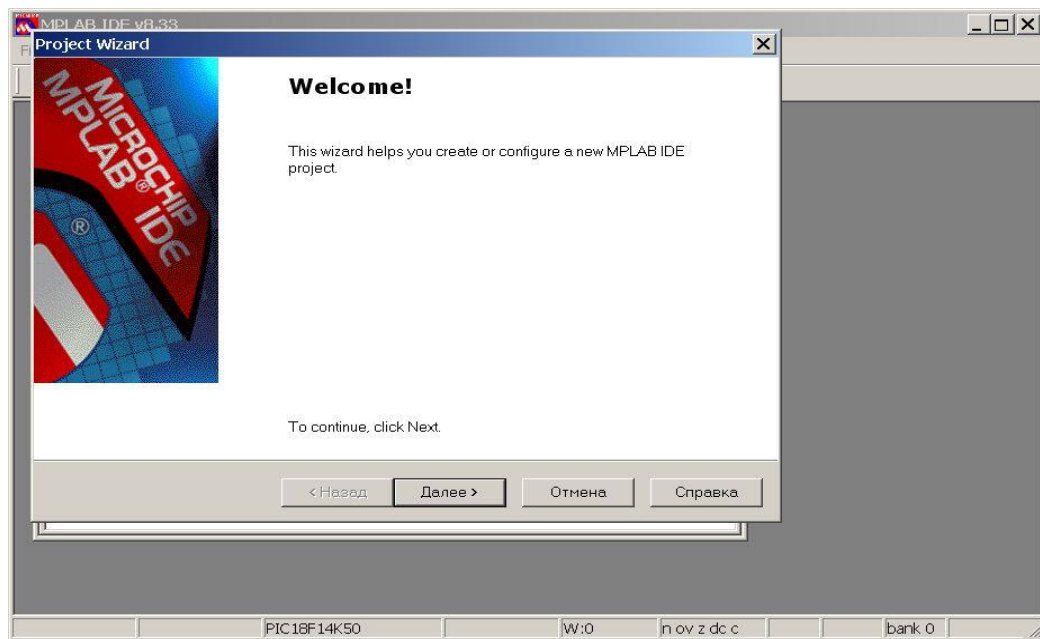


Рисунок 2.3 – Діалогове вікно запрошення помічника

У наступному вікні (Step One – перший крок) необхідно обрати тип МК (рис. 2.4). У списку Device обирають той МК, який встановлений в пристрої або з яким будемо працювати.

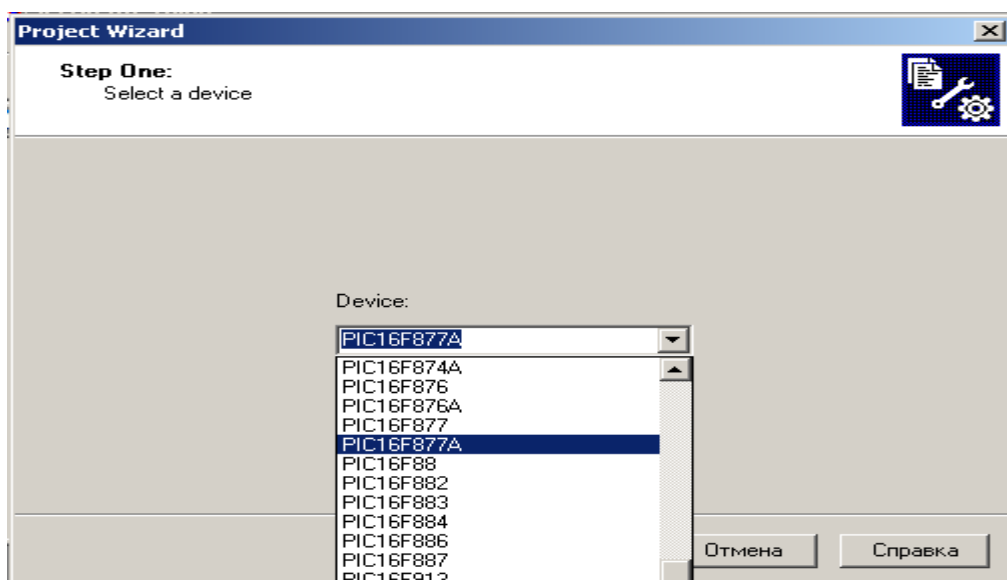


Рисунок 2.4 – Діалогове вікно вибору МК

Наступне вікно – вікно вибору мови програмування (Step Two). Для проекту, що створюється (рис. 2.5) вказується розташування набору програм у графі Active Toolsuite, у вікні Toolsuite Contents програми: MPASM, MPLINK, MPLIB.

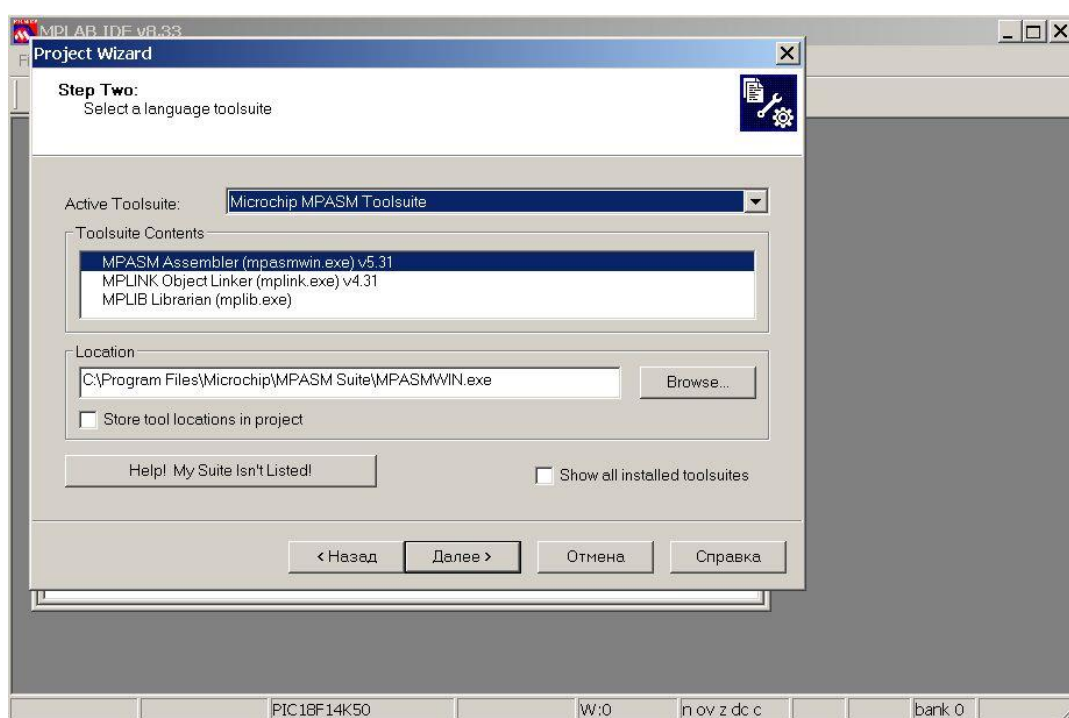


Рисунок 2.5 – Діалогове вікно вибору мови програмування

У цьому діалоговому вікні необхідно визначити мову програмування для MPLAB. При розробці програм для PIC використовуються Асемблер, що дозволяє максимально використовувати ресурси кристалу.

MPASM Assembler є інтегрованим програмним середовищем для розробки програмних кодів МК PIC усіх сімейств. MPASM може використовуватися для двох цілей:

- генерації виконуваної (абсолютного) коди, призначеної для запису в МК за допомогою програматора;
- генерації переміщеної об'єктної коди, яка потім буде пов'язана з іншими модулями, що асемблюють або компілюють.

Код, що виконується, є для MPASM вихідним кодом за замовчуванням. Якщо при асемблюванні не виявляється помилок, то генерується вихідний hex-файл, який може бути завантажений в МК за допомогою програматора.

MPLINK - програма – компоувальник, перетворить переміщені об'єктні коди у виконуваний бінарний код, прив'язаний до абсолютних адрес МК.

MPLIB бібліотечна утиліта яка дозволяє для зручності роботи згрупувати переміщені об'єкти в один файл або бібліотеку.

Після правильної вказівки програм натискаємо Next>.

У вікні, що з'явилося (Step Three) натискаємо кнопку browse (рис. 2.6)

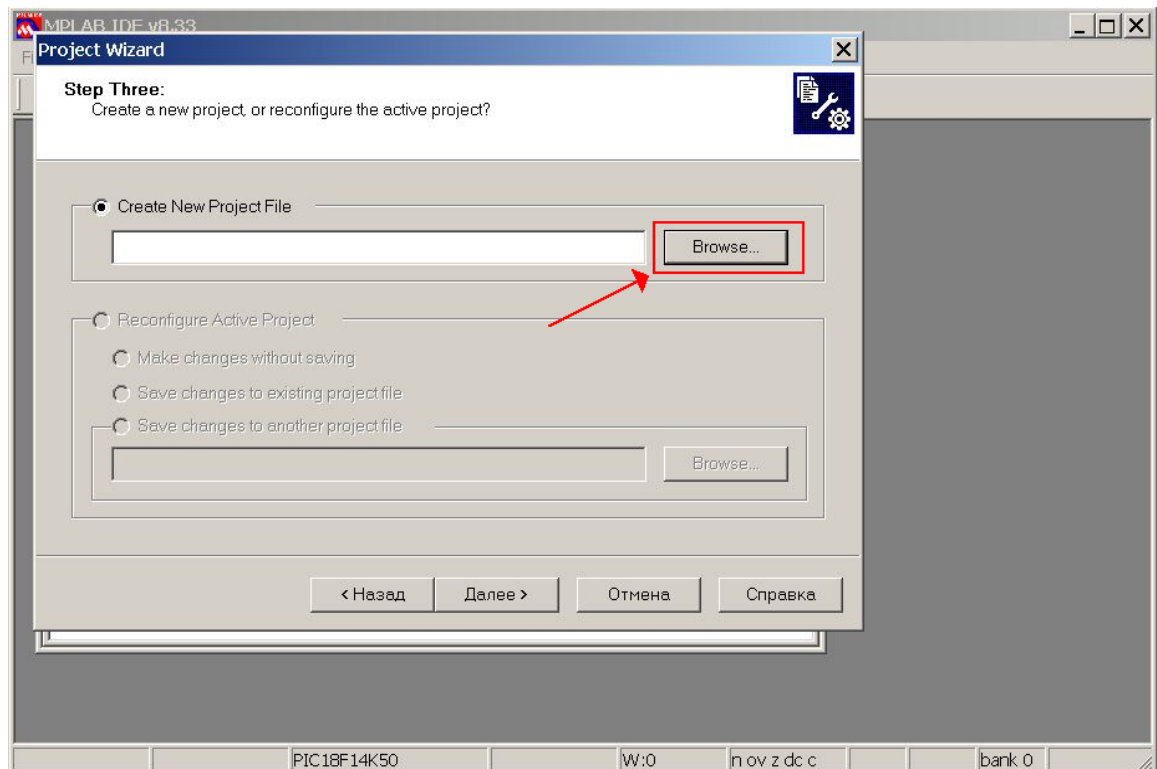


Рисунок 2.6 – Зміна розташування проекту на диску

Далі вказуємо місце і ім'я для нашого проекту в створеній заздалегідь теці на диску C або іншому логічному диску (рис. 2.7).

Не рекомендується використовувати робочий стіл або мережеві диски для зберігання файлів. Це пов'язано з тим, що, MPLAB IDE не працює з файлами, що містять російські або українські букви.

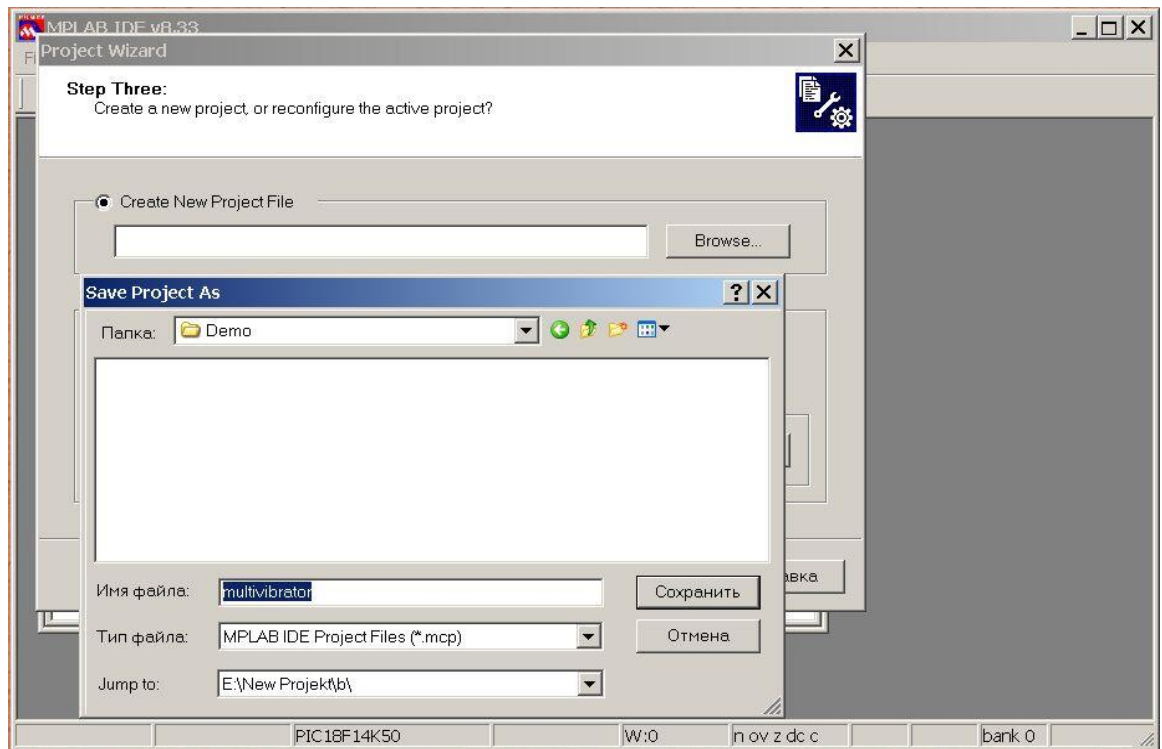


Рисунок 2.7 – Вибір місця розташування проекту

Після вибору місця збереження проекту та введення імені проекту натискаємо Next> і з'являється наступне вікно (рис. 2.8).

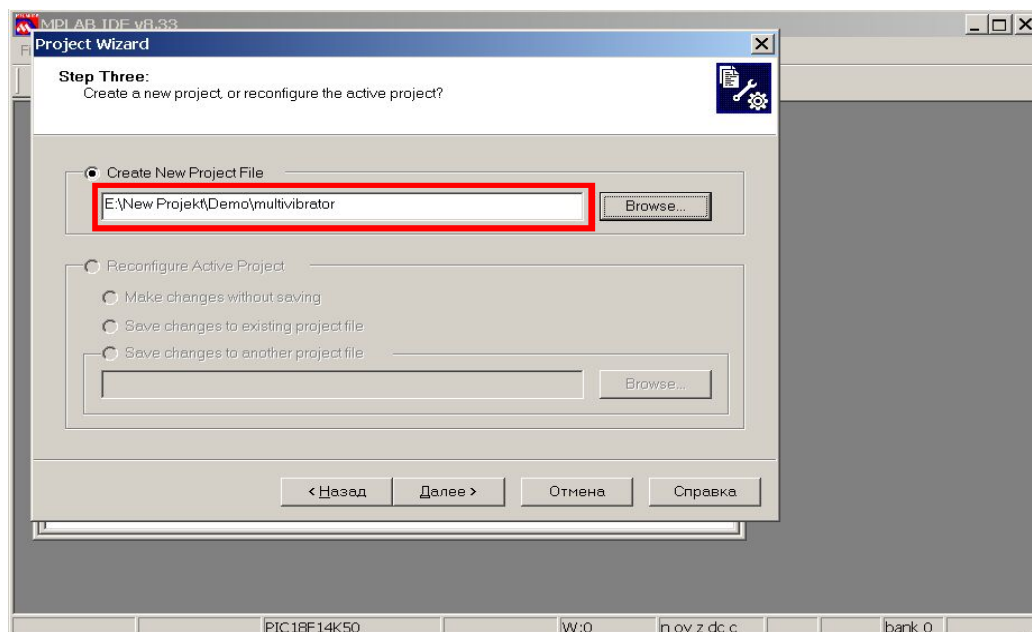


Рисунок 2.8 – Третій крок створення файлу проекту

Перед наступним кроком необхідно створити в тій же теці порожній файл з розширенням .asm однойменний проекту, це можна зробити в Провіднику.

На наступному кроці знаходимо створений файл, відзначаємо його, і додаємо до проекту (рис 2.9).

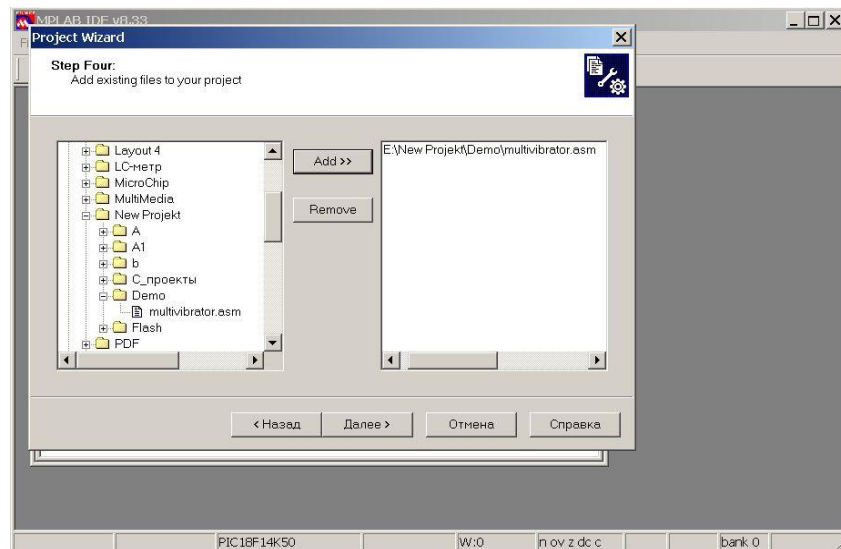


Рисунок 2.9 – Останній крок створення файлу проекту

Останнє вікно Project Wizard – це резюме проекту. Показуються вибраний МК, розташування програм компіляторів і ім'я файлу проекту(рис. 2.10).

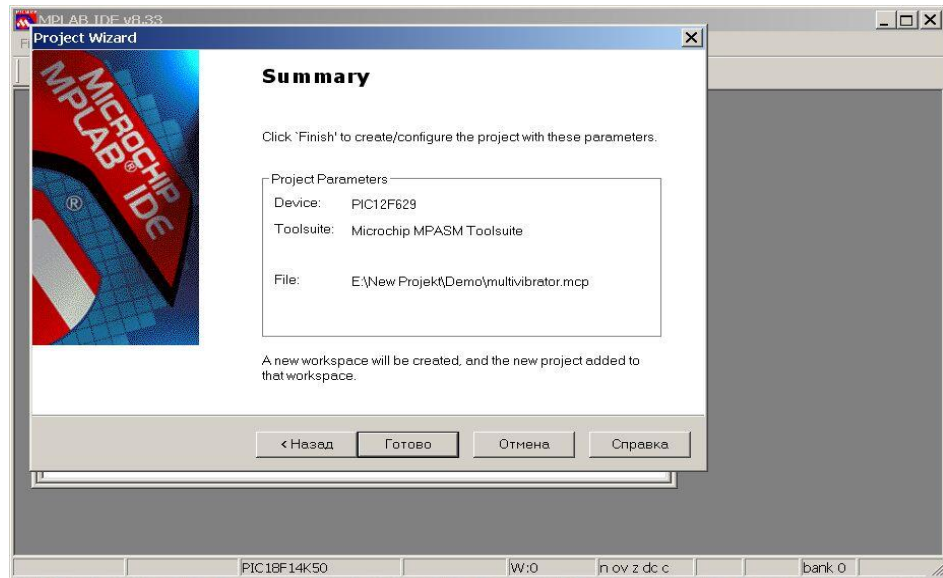


Рисунок 2.10 – Відомості про проект

Після натискання кнопки Finish, на екрані з'явиться вікно з назвою проекту (рисунок 2.11).

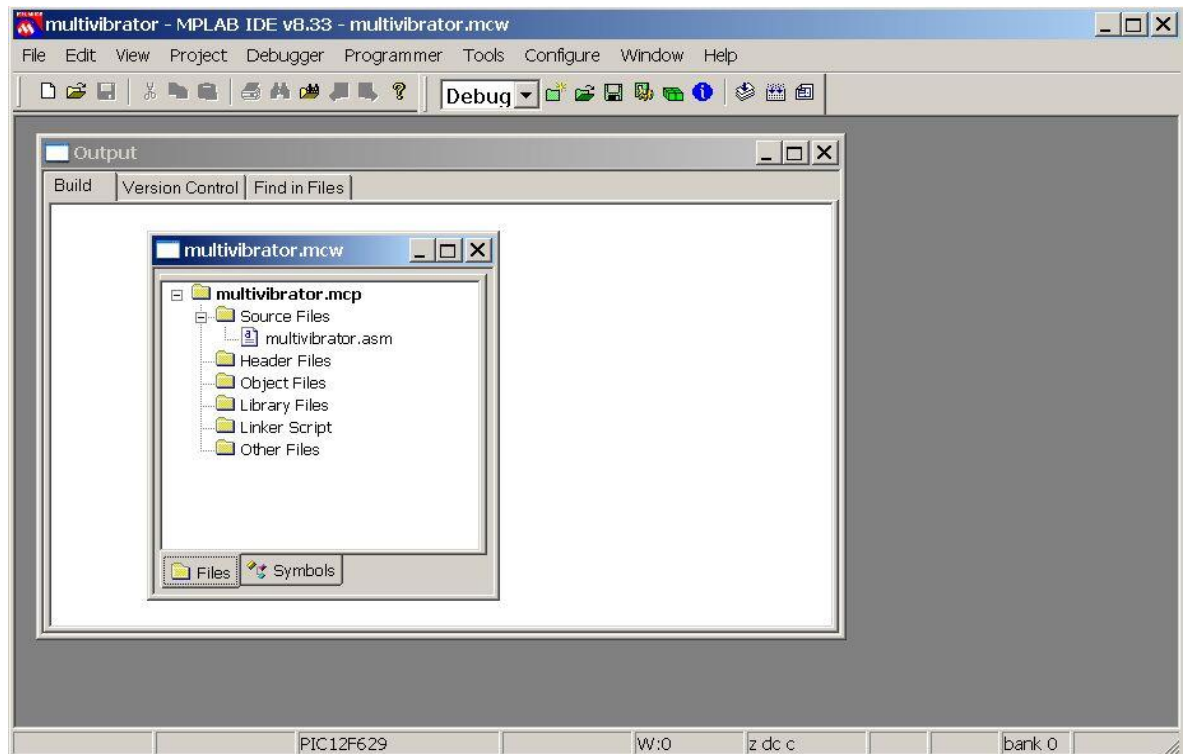


Рисунок 2.11 – Вікно створеного проекту

Тепер можна набирати текст програми на асемблері, або скопіювати сюди вже готовий текст програми та вносити у неї зміни.

2.2 Розроблення програми тестування портів вводу-виводу МК PIC16F84A

Програма для MPLAB IDE, що розроблена на Асемблері, складається з послідовності різних інструкцій, які вводяться у вікно проекту [6].

Діагностична програму «запалює» світлодіоди, що підключені до виводів МК.

Для реалізації програми необхідно звернутися до опису МК PIC16F84A, а також ознайомитися з системою команд МК PIC.

На рис. 2.12 наведена принципова схема пристрою індикації для демонстрації роботи основних команд PIC16F84A.

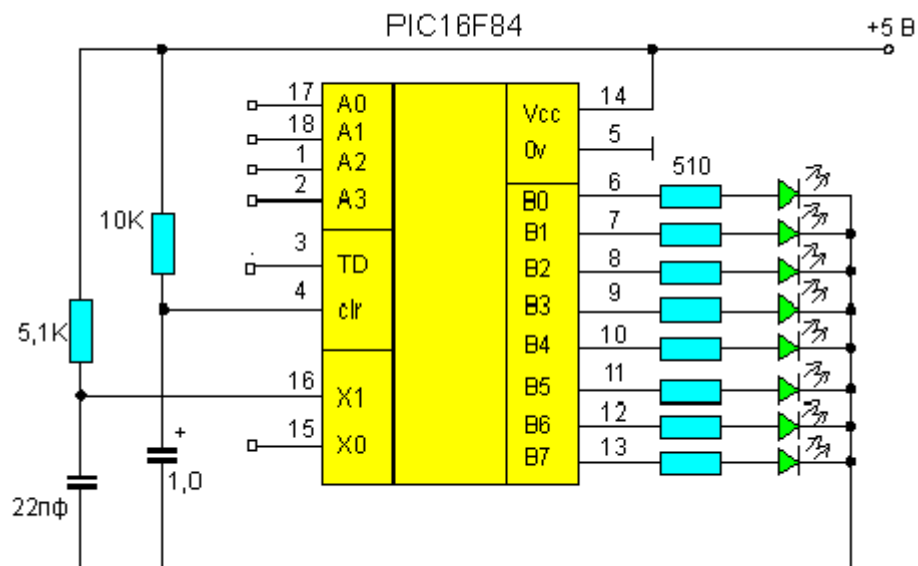


Рисунок 2.12 – Схема пристрою індикації на базі МК PIC16F84A

Приклад програми тестування портів МК PIC16F84 виглядає наступним чином:

```

LIST P=16C84;
    MOVLW 0
    TRIS 6
OPTION
    LOOP SLEEP
    INCF 6,F
    GOTO LOOP
END

```

На самому початку програми встановлюється тип МК **P=16F84**. За наявності цього коду компілятор асоціює наш код саме з цією моделлю МК, відстежуючи правильність його написання. Далі йде текст самої програми.

MOVLW 0 – це команда, 11 00xx kkkk kkkk її машинний код. Ця команда завантажує в регістр W бітову константу, у даному випадку це 0.

TRIS 6 – це команда, 00 0000 0110 0fff її машинний код. Ця команда завантажує вміст регістра W в регістр TRISA або TRISB при f = 5 або 6, відповідно.

OPTION – це команда, 00 0000 0110 0010 її машинний код. Вміст регістра W завантажується в регістр OPTION.

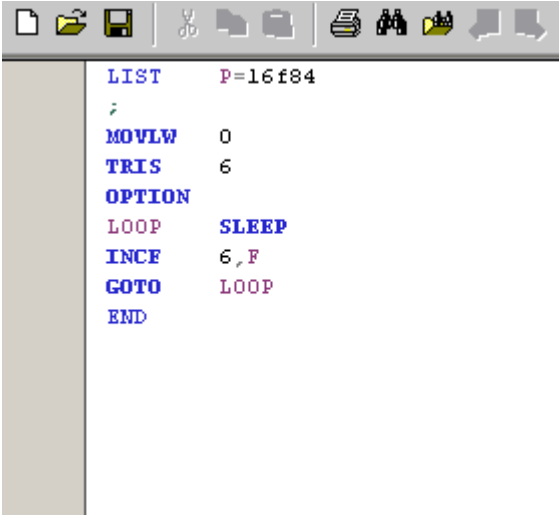
LOOP – це команда. Ця команда призначена для організації циклу. В цієї команди один операнд – ім'я мітки, на яку здійснюється перехід. **SLEEP** – це команда, 00 0000 0110 0011 її машинний код. Команда скидає сторожовий таймер WDT і попередній дільник. У регістрі стану встановлюється біт TE і скидається біт PD. Процесор переходить в режим SLEEP з вимкненим генератором.

INCF 6,F – це команда, 00 1010 dfff ffff її машинний код. Регістр f збільшується на 1. Якщо d = 0, результат зберігається в регістрі W. Якщо d = 1, результат зберігається в тому ж регістрі f.

GOTO LOOP – це команда, 10 1kkk kkkk kkkk її машинний код. Здійснює перехід на наступне кільце числових перетворень.

END – кінець програми.

На рис. 2.13 наведено робоче вікно з текстом програми.



```

LIST      P=16f84
;
MOV LW    0
TRIS     6
OPTION
LOOP     SLEEP
INCF     6,F
GOTO     LOOP
END

```

Рисунок 2.13 – Робоче вікно з текстом програми

Числа в МК можна представляти в двійковій, вісімковій, десятковій та шістнадцятиричній формах. Наприклад, десяткове число 255 можна записати так: у двійковій – 0b11111111; у вісімковій – \$377 та в шістнадцятиричній – 0xff. Найбільш поширена – двійкова система числення.

Пристрій для виводу результатів виконання програми складається з 8 світлодіодів з резисторами, що обмежують струм і елементів, що задають частоту. Кожен контакт МК сімейства PIC може безпосередньо керувати світлодіодом без додаткових підсилювачів.

Почнемо опису базового коду, якому буде використаний в нашій програмі.

Коли пишуть код для проекту, секція заголовка (весь код до рядка з виразом `ORG 0`) повинна враховувати особливості застосування МК. У секції заголовка визначаються логічні імена для всіх ресурсів, що

використовуються у проекті – портів, бітових та байтових змінних і регістрів. Заголовок також встановлює порт вводу/виводу, так що все розряди портів А і В будуть встановлені як виходи після виконання наступних команд:

Лістинг програми ініціалізації портів на виведення інформації.

;Ініціалізація порту А

BCF STATUS,RP ;Вибір банка 0

CLRF CNTRLPORT ;Очистити регістр CNTRLPORT

MOVLW INITA ;Завантажити B'00000000' в регістр W

BSF STATUS,RP ;Вибір банка 1

MOVWF TRISA ;Всі розряди порту А встановити як виходи

;Ініціалізація порту В

BCF STATUS,RP ; Вибір банка 0

CLRF DATAPORT ;Очистити регістр DATAPORT

MOVLW INITB ;Завантажити B'00000000' в регістр W

BSF STATUS,RP ;Вибір банка 1

MOVWF TRISB ;Всі розряди порту В встановити як виходи

Коли включається живлення, PIC16F84 встановлює всі розряди портів А і В на введення і починає виконувати програму з адреси 000h.

Нижче представлений базовий код.

; Приклад базового коду для демонстраційної програми

; Секція заголовка

; опис операційних регістрів

TMR0 EQU 01h

PC EQU 02h

STATUS EQU 03h

FSR EQU 04h

; регістри вводу/виводу

CNTRLPORT EQU 05h

DATAPORT EQU 06h

; комірки оперативної пам'яті

```
SCRATCH EQU 0Ch
```

```
DIGIT EQU 0Dh
```

```
; біти регістра STATUS
```

```
C EQU 0h
```

```
DC EQU 1h
```

```
Z EQU 2h
```

```
PD EQU 3h
```

```
TO EQU 4h
```

```
RP EQU 5h
```

```
; керуючі регістри
```

```
TRISA EQU 85h
```

```
TRISB EQU 86h
```

```
; слова ініціалізації для портів вводу/виводу
```

```
INITA EQU B'00000000'
```

```
INITB EQU B'00000000'
```

```
; Робоча секція
```

```
; початок коду виконання
```

```
ORG 0
```

```
GOTO BEGIN
```

```
ORG 100h
```

```
BEGIN
```

```
; Ініціалізація порту A
```

```
BCF STATUS,RP ;Вибір банка 0
```

```
CLRF CNTRLPORT ;Очистити регістр CNTRLPORT
```

```
MOVLW INITA ;Завантажити B'00000000' в регістр W
```

```
BSF STATUS,RP ;Вибір банка 1
```

```
MOVWF TRISA ;Всі розряди порту A встановить як виходи
```

```
;Ініціалізація порту B
```

```
BCF STATUS,RP ;Вибір банка 0
```

```
CLRF DATAPORT ;Очистити регістр DATAPORT
```

```

MOVLW  INITB      ;Завантажити B'00000000' в регістр W
BSF     STATUS,RP ;Вибір банка 1
MOVWF   TRISB     ;Всі розряди порту В встановити як виходи
END

```

Перейдемо до опису програми.

Всі рядки, що починаються зі знака ";", – це коментарі.

Вираз TMR0 говорить асемблеру, що кожен раз, коли зустрінеться слово TMR0, необхідно підставити значення 01h (01 шістнадцятирічне). Слово "EQU" означає рівність.

Таким чином, ми присвоїли TMR0 значення 1h. Можна використовувати 01h кожен раз, коли необхідно адресувати регістр TMR0, але це буде значно складніше налагоджувати, оскільки необхідно буде пам'ятати, що 01h означає RTCC.

Використання символічних імен ліквідує двозначність і дозволяє полегшити читання вихідного тексту. Комірки ОП також можуть мати імена.

В програмі ніде не використано PC безпосередньо, хоча це ім'я і визначено. В цьому немає помилки – можна визначати імена і потім не використовувати їх, хоча, звичайно, не можна використовувати ім'я, якщо воно не було попередньо визначені.

Робота асемблера полягає в перевірці тексту на дотримання всіх правил, і користувач отримує повідомлення про помилки, якщо щось не буде відповідати правилам.

Можна не тільки іменувати регістри, а й окремі біти всередині регістрів.

Треба звернути увагу на секцію, задану регістром STATUS. Символу C присвоєно значення 0h, оскільки C або CARRY, це нульовий біт слова стану STATUS. Кожен раз, коли перевіряється біт CARRY (біт 0), необхідно користуватися попередньо визначеним символом "C". Кожен раз, коли ми захочемо звернутись до біту 2 або біту ZERO, необхідно використовувати

символ "Z" замість 02h. Можна визначити повну структуру бітів регістра, навіть якщо потім не всі з них буде використано.

Перед тим, як почати виконуваний код, необхідно задати вираз `ORG 0`. Це покажчик для асемблера, що код, наступний за цим виразом, починається з нульової адреси EEPROM. Вираз "ORG" використовується для розміщення сегментів коду по різним адресам в межах розміру EEPROM. Ще один вираз `ORG` знаходиться перед міткою `BEGIN`, що має адресу 100h, як задано виразом `ORG 100h`.

Виконуваний код повинен закінчуватися директивою `END`, що означає, що за цією директивою відсутні виконавчі команди.

При включенні живлення PIC16F84 переходить на адресу 000h. Перша інструкція, яка буде виконана МК, це команда `GOTO BEGIN`, яка передасть керування на адресу 100h і подальша робота продовжиться з цієї адреси.

`BEGIN` вибирається користувачем як ім'я мітки, якій асемблер використовує як адресу посилення. У процесі асемблер визначає розташування мітки `BEGIN` і запам'ятовує, що якщо це ім'я зустрінеться ще раз, замість нього буде підставлена адреса мітки.

Команди `CALL` і `GOTO` використовують мітки для посилень в початковому тексті. Команда `MOVLW INITA` завантажує в робочій реєстр `W` значення `INITA`. Це значення задано в заголовку і дорівнює `B'00000000`, тобто 00h. Символи `B'` означають, що дані задані в двійковому форматі. Можна було б написати в цьому ж місці `0` (десятковий) або `0h` (шістнадцятиричний) і отримати той же самий результат. Двійкове представлення зручніше використовувати в тих випадках, коли передбачаються операції з бітами в реєстрі.

Наступна команда `MOVWF TRISA` завантажує значення з робочого реєстра `W` в реєстр керування конфігурацією порту `A` `TRISA`. Значення `0` в розрядах цього реєстра визначає, що відповідний розряд порту `A` є виходом.

У нашому випадку все розряди порту `A` встановлюються виходами. Необхідно звернути увагу, що порт `A` має тільки 5 розрядів, і старші 3 біта

значення, що записуються в реєстр TRISA, також мають 5 розрядів, не використовуються. Якщо б треба було, наприклад, встановити молодший розряд порту А як вхід, ми б задали в секції опису реєстрів значення INITA рівне B'00000001 '. Якщо по ходу програми необхідно змінити призначення окремих розрядів портів, наприклад, при двунправленій передачі, то зручніше за все задати всі необхідні слова конфігурації в секції опису, як, наприклад для INITA і INITB.

Наступні дві команди MOVLW INITB і MOVWF TRISB визначають конфігурацію порту В. Можна зекономити і не писати команду MOVLW INITB, оскільки INITB також рівне 0h.

Команди BCF STATUS, RP і BSF STATUS, RP потрібні для перемикання між банками пам'яті. Справа в тому, що вся пам'ять даних МАК розбита на два банки. Банку 0 відповідають адреси 00h .. 7F, банку 1-8F .. FF. Вибір банку визначається станом біта 5 в реєстрі STATUS. Коли цей біт встановлений в 1, обраний банк 1, інакше – банк 0.

Таким чином, аналізуючи лістинг програми зазначимо наступне:

- за допомогою команди EQU асемблер отримав вказівку, які символічні імена необхідно використати;
- встановлено вектор скидання на адресу 000h;
- встановлено початкову адресу виконання програми з мітки BEGIN на адресі 100h.
- провели конфігурацію контактів портів А і В як виходи.

Для реалізації першої програми вистачить всього трьох команд:

MOVLW k

MOVWF f

GOTO k

Ці команди вже були використані в заголовку базового коду. Команда MOVLW завантажує константу в робочий реєстр W.

Наступна команда MOVWF пересилає байт з робочого реєстра W в заданий реєстр f. Команда GOTO передає керування на адресу k. Наступна

програма записує в робочій регістр *W* значення 01010101 і потім видає його вміст в порт *B*. Після запуску цієї програми запалюються 4 світлодіоди.

`MOVLW B'01010101 '`; завантажити 01010101 в регістр *W*

`MOVWF DATAPORT`; записати *W* в порт *B* (*DATAPORT*)

`GOTO $`; зациклитися назавжди

Директива асемблера "\$" означає поточне значення програмного лічильника (PC). Тому команда `GOTO $` означає перехід туди, де ми в даний момент перебуваємо. Такий цикл нескінченний, оскільки не існує способу (крім переривання) вийти з нього. Команда `GOTO $` часто застосовується для зупинки коду при налаштуванні.

2.3 Компіляція розробленої програми

Тепер написану програму необхідно скомпілювати [7]. Для цього у верхньому меню вікна програми необхідно вибрати опцію `Project -> Build All` або натиснути клавішу `Ctrl+F10` на клавіатурі або натиснути кнопку `Build All` (рис. 2.14).

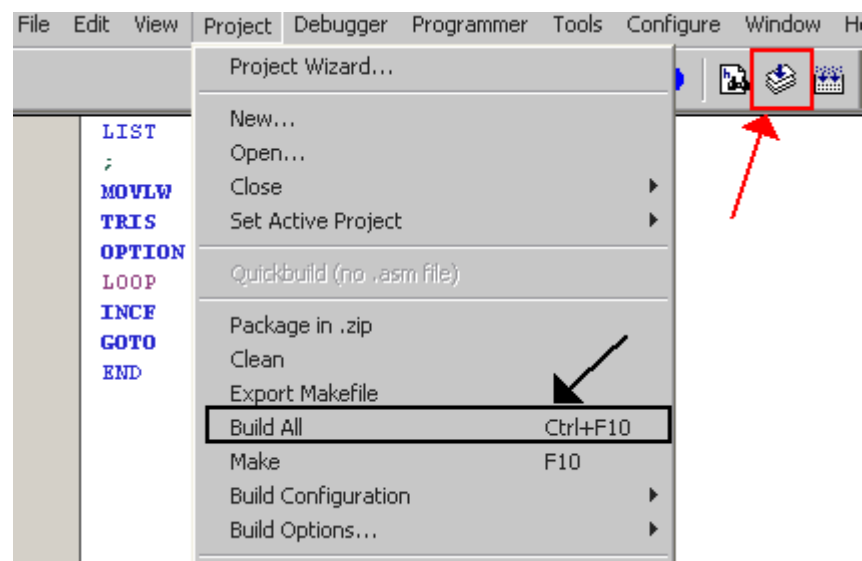


Рисунок 2.14 – Команда на початок компіляції програми

Після завершення програми компіляції внизу вікна програми з'явиться звіт про результат компіляції. Якщо помилок не виявлено, то в цьому вікні можна ознайомитися із статистикою роботи компілятора (рис. 2.15).

```

Build | Version Control | Find in Files
-----
Debug build of project 'C:\Program Files\Microchip\PIC\OH.mcp' started.
Preprocessor symbol '__DEBUG' is defined.

Clean: Deleting intermediary and output files.
Clean: Deleted file "C:\Program Files\Microchip\PIC\OH.mcs".
Clean: Done.
Executing: "C:\Program Files\Microchip\MPASM Suite\MPASMWIN.exe" /q /p16F84 "OH.asm" /I"OH.lst" /e"OH.err" /d__DEBU
Warning[205] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 1 : Found directive in column 1. (LIST)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 3 : Found opcode in column 1. (MOVLW)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 4 : Found opcode in column 1. (TRIS)
Warning[224] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 4 : Use of this instruction is not recommended.
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 5 : Found opcode in column 1. (OPTION)
Warning[224] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 5 : Use of this instruction is not recommended.
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 7 : Found opcode in column 1. (INCF)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 8 : Found opcode in column 1. (GOTO)
Warning[205] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 9 : Found directive in column 1. (END)
Executing: "C:\Program Files\Microchip\MPASM Suite\mplink.exe" /p16F84 "OH.o" /u__DEBUG /z__MPLAB_BUILD=1 /z__MPL
MPLINK 4.30.01, Linker
Copyright (c) 2009 Microchip Technology Inc.
Errors : 0

Loaded C:\Program Files\Microchip\PIC\OH.cof.

Debug build of project 'C:\Program Files\Microchip\PIC\OH.mcp' succeeded.
Preprocessor symbol '__DEBUG' is defined.

BUILD SUCCEEDED

```

Рисунок 2.15 – Звіт про роботу компілятора

2.4 Робота в симуляторі та налаштування програми

Якщо процес компіляції пройшов без помилок, то можна приступити до налаштування програми. Для цього треба визначитись яким інструментом буде проводитись перевірка роботи програми. Це може бути реальний пристрій з вибраним МК підключений через інтерфейс типа Icd2 до ПК або віртуальний пристрій імітований програмою MPLAB. Для використання симулятора MPLAB треба вибрати Debugger > MPLAB SIM (рис. 2.16).

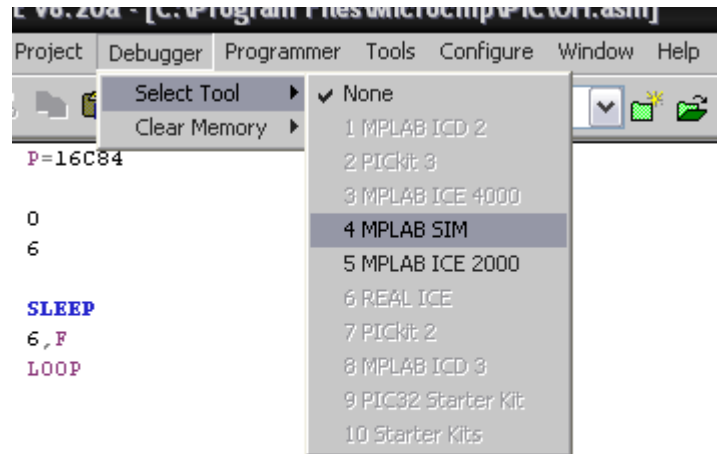


Рисунок 2.16 – Запуск програми симуляції МК

Після вибору симулятора MPLAB SIM з'являється додаткова панель інструментів та нова вкладка у вікні Output. На початку виконання програми емулятор встановить покажчик на першу команду програми (рис. 2.17).

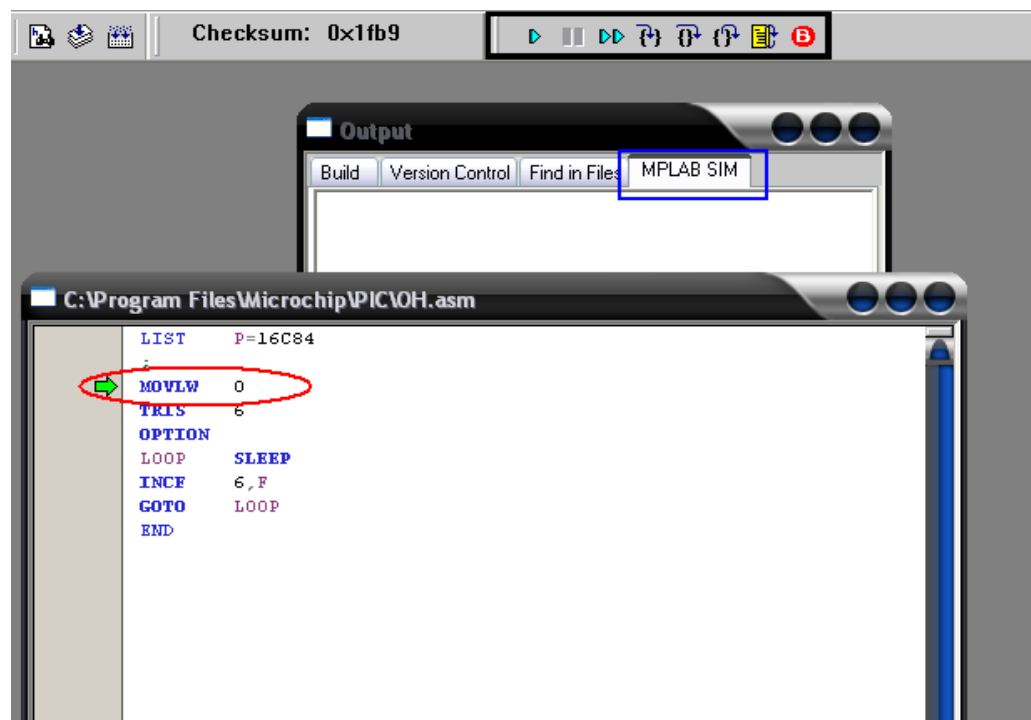


Рисунок 2.17 – Виконання першої команди програми

Запуск симуляції здійснюється натиском на блакитний трикутник. Якщо клацнути двічі мишкою на сірому фоні з ліва від рядка з'явиться червоний кружок – це "ТОЧКА ЗУПИНУ" (Breakpoint) – перед виконанням рядка коди перед якою стоїть "ТЗ" симулятор зупиниться і чекатиме вашої команди (рис. 2.18).

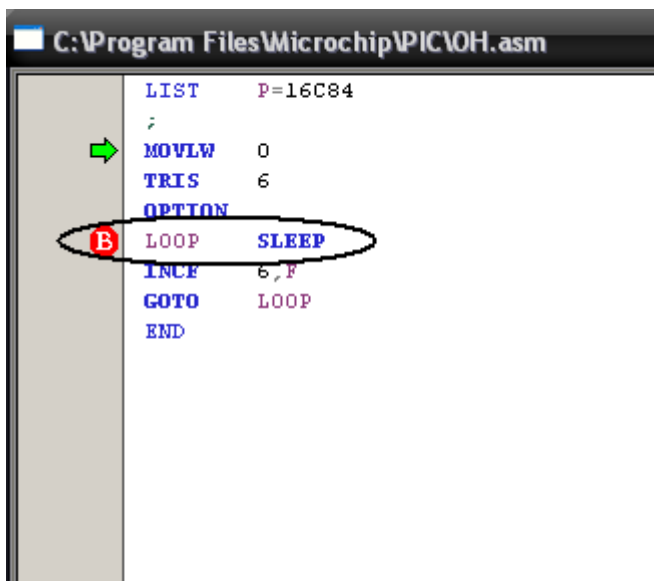


Рисунок 2.18 – Виконання програми в покроковому режимі

Для подальшого виконання програми необхідно знову натиснути блакитний трикутник.

Для того щоб подивитись як програма виконується в нутрі PIC та що в ній відбувається по ходу відробітку програми треба вдатися до іншого методу симуляції. Щоб проглянути вміст пам'яті програми треба відкрити Program Memory. Для цього виберіть View > Program Memory (рис. 2.19).

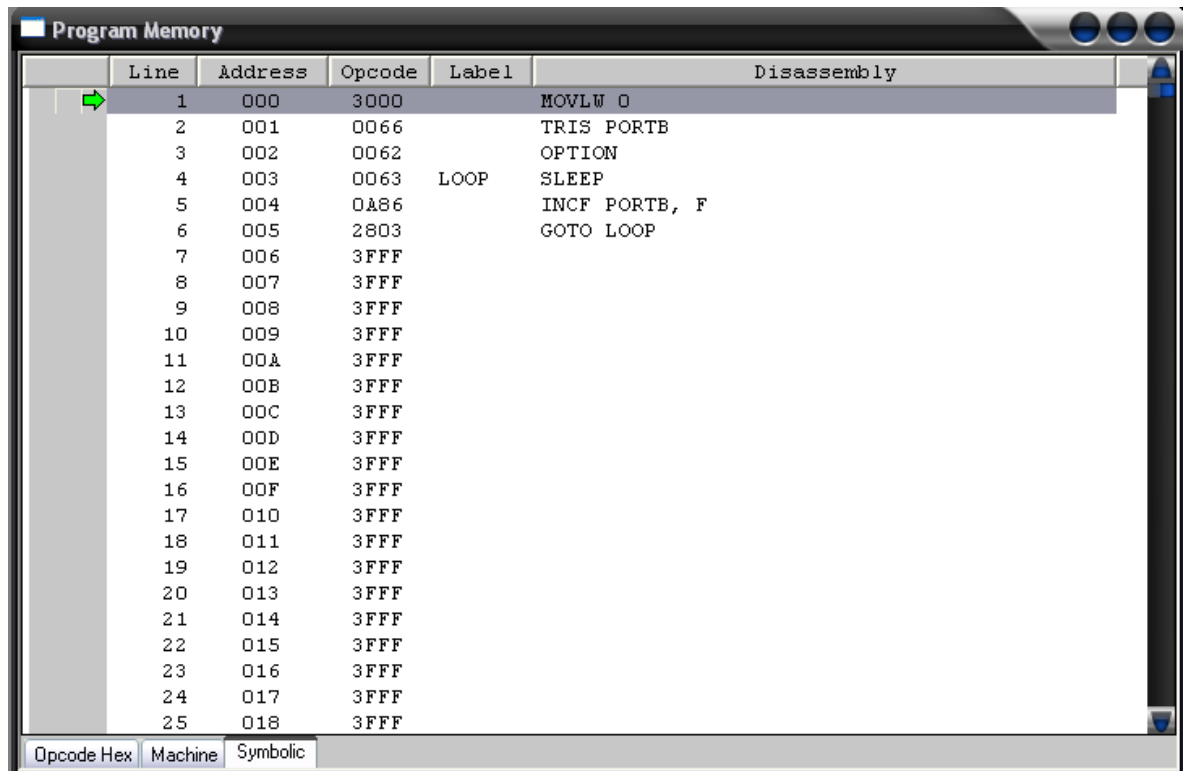


Рисунок 2.19 – Вікно Program Memory

У цьому вікні:

- 1-й стовпець: номери елементів пам'яті програм, в 10-чній системі числення.
- 2-й стовпець: адреси елементів пам'яті програм, в 16-річній системі числення.
- 3-й стовпець: бар'єр пам'яті.
- 4-й і 5-й стовпці: власне кажучи, сама програма без зайвих коментарів і позначок.

У вікні Program Memory, можна визначити, яку саме кількість елементів пам'яті програм займає програма (в даному випадку, 6) і який об'єм пам'яті програм. В даному випадку 1024 комірки.

Для відкриття вікна File Register виберіть View > File Register (рис. 2.20).

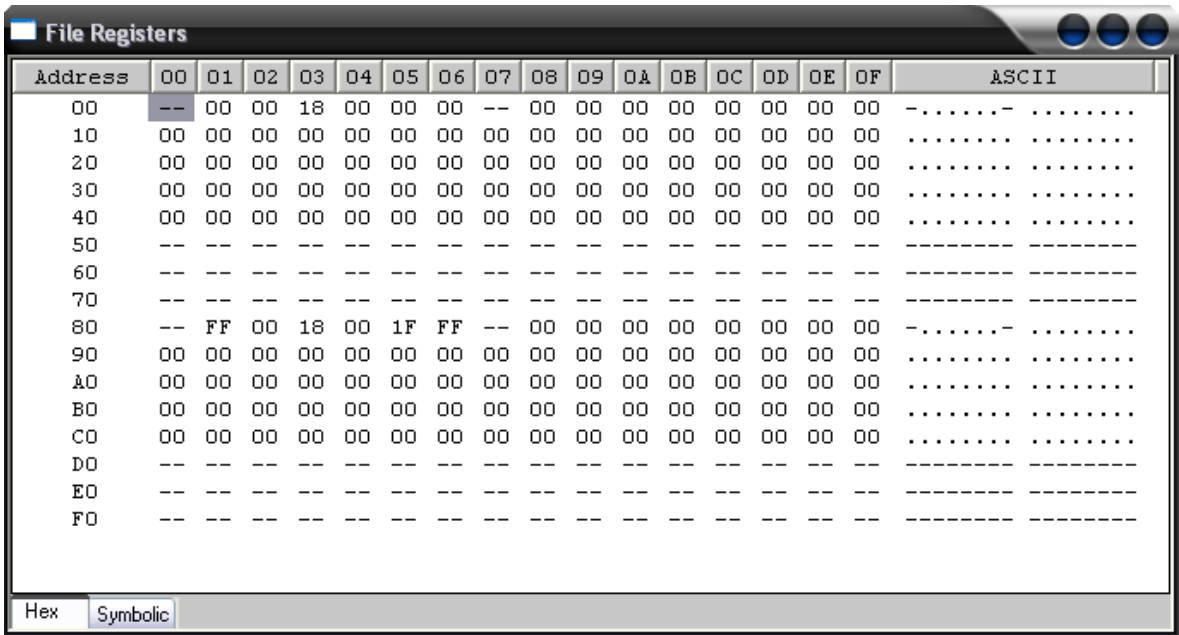
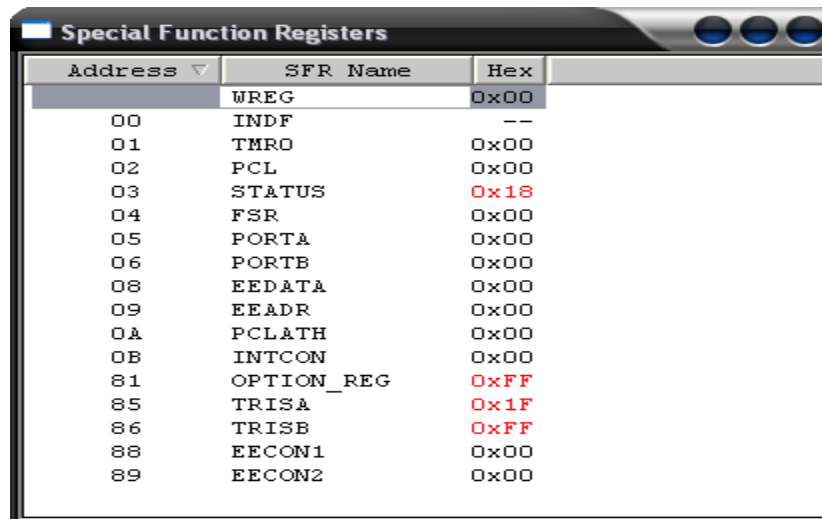


Рисунок 2.20 – Вікно File Register

File Register це область оперативної пам'яті. У вікні File Register, можливо побачити "числову начинку" будь-якого з регістрів області оперативної пам'яті. Причому, за станом на момент завершення виконання будь-якої команди програми. Усі регістри, що знаходяться в області оперативної пам'яті, 8-бітові, і тому вони працюють в числовому діапазоні від 00h до FFh (від .0 до .255).

Для відкриття вікна Special Function Register виберіть View > Special Function Register (рис.к 2.21).



Address	SFR Name	Hex
00	WREG	0x00
01	INDF	--
02	TMRO	0x00
03	PCL	0x00
04	STATUS	0x18
05	FSR	0x00
06	PORTA	0x00
07	PORTB	0x00
08	EEDATA	0x00
09	EEADR	0x00
0A	PCLATH	0x00
0B	INTCON	0x00
81	OPTION_REG	0xFF
85	TRISA	0x1F
86	TRISB	0xFF
88	EECON1	0x00
89	EECON2	0x00

Рисунок 2.21 – Вікно Special Function Register

Якщо в попередньому вікні відображувався вміст всіх регістрів області оперативної пам'яті, то в цьому вікні лише вміст регістрів спеціального призначення, в трьох системах числення. Вікно Special Function Register можна використовувати як конвертор систем числення.

2.5 Основні засоби налаштування MP-LAB IDE

У пункті меню Debugger є наступні підпункти:

- Clear Program Memory – Очистити пам'ять програм;
- Run – запуск виконання в безперервному режимі (з використанням емулятора в реальному часі, з використанням симулятора з максимальною можливою швидкістю);
- Animate – запуск в безперервному режимі, але з можливістю спостерігати зміну значень регістрів і пам'яті в процесі. Аналогічно постійному натисненню на кнопку покрокового виконання
- Halt – зупинити виконання програми.;
- Step Into – виконати одну команду і зупинитися;
- Step Over – працює так само як і Step Into, але не опускається в

підпрограми (виконуються в безперервному режимі);

- Step Out – використовується для запуску швидкого автоматичного налаштування, якщо поточна позиція команди знаходиться у програмі, що викликається;

- Reset – меню скидання;

- Breakpoints. – Конфігурація точок зупинки (переривання). Можна встановлювати до 16 крапок. Можна встановити досягши адреси або області і інших випадків. Якщо зупинка у вибраній точці не відбувається, то перевірте чи встановлений відповідний прапор у вікні конфігурації (Options>development Mode). Ще точку останову можна встановити клацнувши правої кнопки миші на потрібному рядку вихідного файлу і вибравши Break Point(s), але ці установки скидаються при кожному перекомпілюванні проекту;

- Stopwatch – вікно перегляду тимчасових параметрів. Показує тактову частоту (виставляється у вікні конфігурації МК) і час минулий з початку виконання програми або з моменту обнуління в тактах і мікросекундах. Дуже зручно при налаштуванні програми, коли використовуються лічильники, таймери, переривання і все, зв'язане з часом;

- Complex Breakpoints – викликає симулятор Complex Breakpoints;

На рисунку 2.22 наведено симулятор Complex Breakpoints.

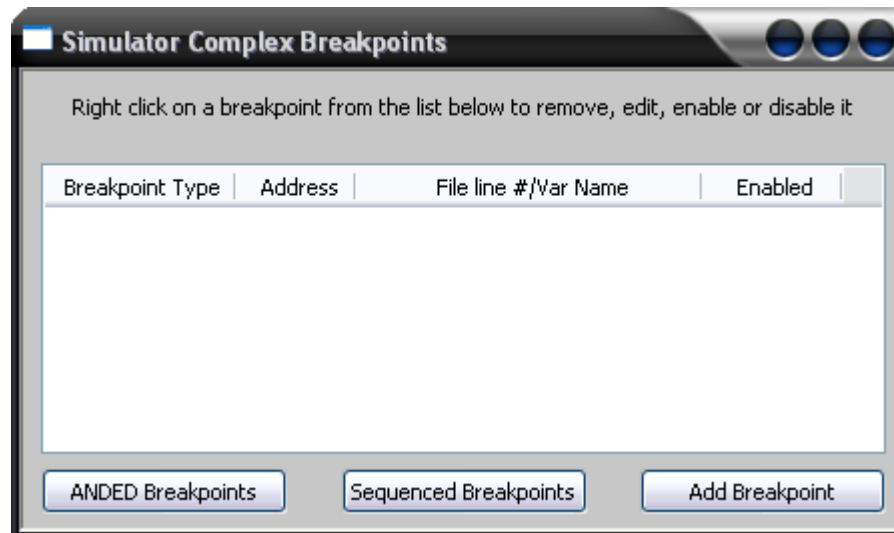


Рисунок 2.22 – Симулятор Complex Breakpoints

– Stimulus – Дозволяє імітувати зовнішні сигнали на вхідних виводах (портах) МК. MPLAB не дозволяє змінювати стан портів як інших регістрів. З одного боку це викликає деякі незручності, але з іншою є досить потужні засоби моделювання вхідних сигналів: Asynchronous Stimulus Dialog – опція дозволяє встановлювати постійні логічні рівні на вхідних виводах Pin Stimulus Files – підключення спеціального текстового файлу (розширення .sti) що описує зміну зовнішніх сигналів на вхідних виводах МК. Register Stimulus Files – підключення текстового файлу (розширення .reg), що описує зміну регістру (8 біт). Зазвичай використовується при налаштуванні аналого-цифрового перетворення. Clock Stimulus – опція дозволяє імітувати періодичні сигнали;

- Profile – виводь профіль симулятора на екран або очищає його;
- Refresh PM – оновлення пам'яті програми;
- Settings – налаштування поточного проекту.

2.6 Асемблювання програми

Для асемблювання використано макроасемблер MPASM, який містить всі необхідні для налаштування можливості. MPASM входить в пакет програм Microchip MPLAB фірми Microchip Technology [8].

Для виконання необхідно запустити файл Mpsasmwin.exe. У полях Radix, Warning Level, Hex Output, Macro Expansion обрати Default. У Generation Files дозволите Error File і List File. У рядку Processor встановити 16F84, Tab Size – 8, Case Sensitive – включений. У рядку Source File Name за допомогою кнопки Browse обрати файл, який повинен бути асембльований. За замовчуванням файлу вихідного тексту .ASM.

В результаті роботи асемблера створюються наступні файли:

- * HEX – об'єктний файл
- * LST – файл лістингу
- * ERR – файл помилок і попереджень
- * COD

Об'єктний файл створюється у 16-річному форматі і містить код, який повинен бути записаний у мікросхему. Файл лістингу містить повний лістинг програми разом з кодом завантаження. У файл помилок і попереджень записуються всі помилки і попередження, що виникають в процесі асемблювання.

Після обробки програми асемблер повинен видати повідомлення "Assembly Successful", що означає, що помилок не було знайдено. Файл помилок в цьому випадку не створюється. Якщо асемблер видав які не будь повідомлення про помилки, або не створилися файли EXAMPLE.HEX, EXAMPLE.LST і EXAMPLE.COD, необхідно перевірити синтаксис написаної програми.

2.7 Висновки до розділу 2

У другому розділі дипломного проекту розглянуто питання створення проекту в програмному середовищі MPLAB IDE. Розроблена програми тестування портів вводу-виводу МК PIC16F84A. Візуалізація результатів тестування здійснюється виводом інформації в порт МК та запалюванням відповідних світлодіодів, що свідчить про працездатність вузлів та порту МК. Проведена компіляція розробленої програми. Розглянуті робота в симуляторі, основні засоби налаштування MPLAB IDE та асемблювання програми.

РОЗДІЛ 3

ЗАВАНТАЖЕННЯ ПРОГРАМИ ТЕСТУВАННЯ В ПАМ'ЯТЬ МІКРОКОНТРОЛЕРА PIC

Кінцевим етапом даної роботи є «прошивка» МК. Сам МК не сприймає команди асемблера які пишуть програмісти, а сприймає тільки машинні коди. Е безліч програм які перетворюють готову програму на файл з набором потрібних кодів які можна завантажувати безпосередньо до пам'яті МК.

3.1 Пам'ять команд мікроконтролерів PIC. Технологія FLASH–EEPROM

На даний час всі МК PIC оснащуються електрично перепрограмованою flash – EPROM пам'яттю програм.

Такі перші представники однокристальних МК як 8048 і 8051 компанії Intel можна було придбати або з пам'яттю типа MROM (програмно–маскована ROM), або з вбудованою пам'яттю EPROM. Вони могли працювати також і із зовнішньою пам'яттю EPROM. Моделі з пам'яттю типа MROM мали один недолік: їх міг програмувати лише завод, що виготовляв такі моделі в рамках виробничого процесу, а витерти дані було неможливо.

МК з пам'яттю EPROM мали в корпусі віконце з кварцовим склом для витирання інформації. Щоб перепрограмувати такий PIC МК, необхідно було повітити деякий час у це віконце кварцовою лампою.

МК сімейства PIC з вбудованою флеш–пам'яттю EPROM економлять не лише місце на платі електронного пристрою, але також представляють в розпорядження користувача всі контакти вводу/виводу МК.

Для програмування МК PIC потрібен програматор. Програматор можна купити вже готовий або зробити самому вибравши 1 з великої кількості схем пропонованих в Інтернеті. А можливо зробити програматор EXTRA – PIC наведений на рисунку 3.1

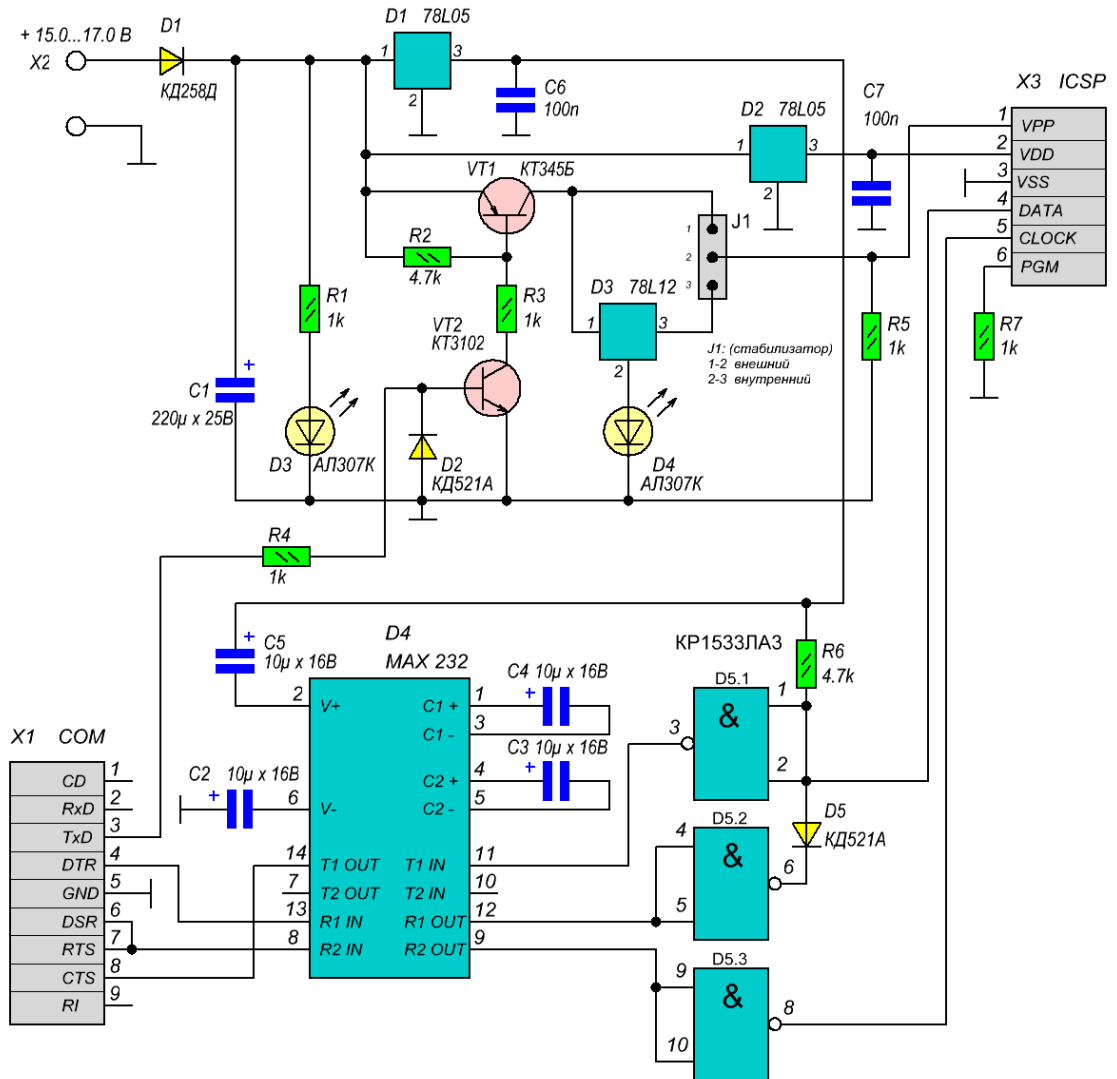


Рисунок 3.1 – Схема програматора EXTRA – PIC

3.2 Регістри і оперативна пам'ять мікроконтролера PIC16c84

Область ОП організована як 128 x 8. До комірок ОП можна адресуватися прямо або побічно, через регістр показчик FSR (04h). Це також відноситься і до EEPROM пам'яті даних – констант.

У регістрі статусу (03h) є біти вибору сторінок, які дозволяють звертатися до чотирьох сторінок майбутніх модифікацій цього кристала. Проте для PIC16C84 пам'ять даних існує лише до адреси 02fh. Перші 12 адрес використовуються для розміщення регістрів спеціального

призначення. Регістри з адресами 0ch-2fh можуть бути використані, як регістри загального призначення, які є статичним ОЗУ. Деякі регістри спеціального призначення продубльовані на обох сторінках, а деякі розташовані на сторінці 1 окремо. Коли встановлена сторінка 1, те звернення до адрес 8ch-afh фактично адресує сторінку 0. До регістрів можна адресуватися прямо або побічно. У обох випадках можна адресувати до 512 регістрів.

3.3 Читання даних EEPROM мікроконтролерів PIC. Пам'ять даних EEPROM

Пам'ять даних EEPROM дозволяє прочитати і записати байт інформації. При записі байта автоматично стирається попереднє значення і записуються нові дані (стирання перед записом). Всі ці операції виробляє вбудований автомат запису EEPROM. Вміст елементів цієї пам'яті зберігається при виключенні живлення. Кристал Pic16c84 має пам'ять даних 64x8 EEPROM біт, яка дозволяє запис і читання під час нормальної роботи (у всьому діапазоні живлячої напруги). Ця пам'ять не належить області регістрів ОЗУ. Доступ до неї здійснюється через два регістри: EEDATA <08h>, який містить в собі восьмибітові дані для читання/запису і EEADR <09h>, який містить в собі адресу вічка до якої йде звернення. Додатково є два керівників регістра: Eecon1 <88h> і Eecon2 <89h>.

При читання даних з пам'яті EEPROM необхідно записати необхідну адресу в EEADR регістр і потім встановити біт RD Eecon1<0> в одиницю. Дані з'являться в наступному командному циклі в регістрі EEDATA і можуть бути прочитані. Дані в регістрі EEDATA замикаються.

При записі в пам'ять EEPROM, необхідно спочатку записати необхідну адресу в EEADR регістр і дані в EEDATA регістр. Потім виконати спеціальну послідовність команд, що виробляє безпосередній запис.

Під час виконання цієї ділянки програми, всі переривання мають бути заборонені для точного виконання часової діаграми. Час запису – приблизно 10мс. Фактичний час запису змінюватиметься залежно від напруги, температури і індивідуальних властивостей кристала. В кінці запису біт WR автоматично обнуляється, а прапор завершення запису EEIF, він же запит на переривання, встановлюється. Для запобігання випадковим записам в пам'ять даних передбачений спеціальний біт WREN в регістрі Eecon1. Рекомендується тримати біт WREN вимкненим, окрім тих випадків, коли потрібно відновити пам'ять даних. Більш того, кодові сегменти, які встановлюють біт WREN і ті, які виконують запис повинні зберігатися на різних адресах, щоб уникнути випадкового виконання їх обох при збої програми.

3.4 Робота з мікроконтролерами PIC в програмі Pony Prog

Pony Prog – це найпростіший і надійний програматор. Він може працювати як незалежно так і спільно з ICProg [9].

У верхній, правій частині вікна програми, на одній лінії, Ви побачите два рядки: вибору вигляду і типу МК (або мікросхеми пам'яті). Вони забезпечені кнопками, клацнувши по яких, у випадному списку, можна проглянути ті види і типів МК (і мікросхем пам'яті), які підтримує програма Pony Prog (рисунок 3.2).

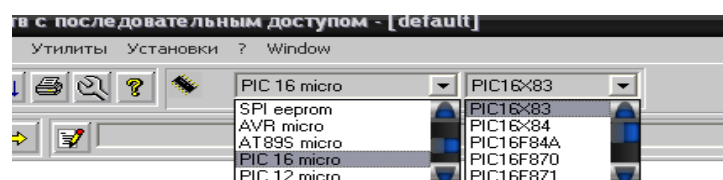


Рисунок 3.2 – Вибір виду та типу МК або схеми

Після вибору потрібного пристрою слід перейти до процедури налаштування програми. У головному меню програми, клацніть по кнопці Установки. У відкритшомуся списку, що складається з двох пунктів, клацніть по пункту Налаштування устаткування. Розкриється вікно Налаштування плати програматора, в якому єдине що потрібно зробити поставити крапку в кружечку з номером того СОМ порту, до якого підключена апаратна частина програматора. Останнє за умовчанням (з випадного списку має бути вибрана рядок Si Prog API, а в списку Вибір полярності сигналів керування, всі квадратики мають бути без галочок).

Далі слід провести калібрування. Для цього ще раз потрібно клацнути по кнопці Установки і з випадного списку, вибрати пункт . Програма спочатку запитає дозволу на запуск калібрування. Після підтвердження калібрування через певний час, відкриється віконце Notice з написом Калібрування завершено.

У PICax є два види незалежної пам'яті: пам'ять програм і пам'ять даних. Запис в них можна виробляти багато разів. У пам'ять даних записуються дані, які необхідно зберегти після виключення живлення, з метою їх використання після наступного включення живлення. Цей об'єм невеликий для PIC16F84A усього 64 байти.

Об'єм пам'яті програм значно більше. Пам'ять програм PIC16C84A дозволяє розмістити в ній до 1024 команд (максимум). Саме у пам'ять програм і записується програма. Але спочатку потрібно hex-файл перевести в машинні коди програмою Pony Prog. Для цього відкриваємо .HEX файл натиснувши Файл > Відкрити файл з даними. Після відкриття .HEX файлу у вікні програми Pony Prog з'явиться те що призначене для самого МК (рис. 3.3).

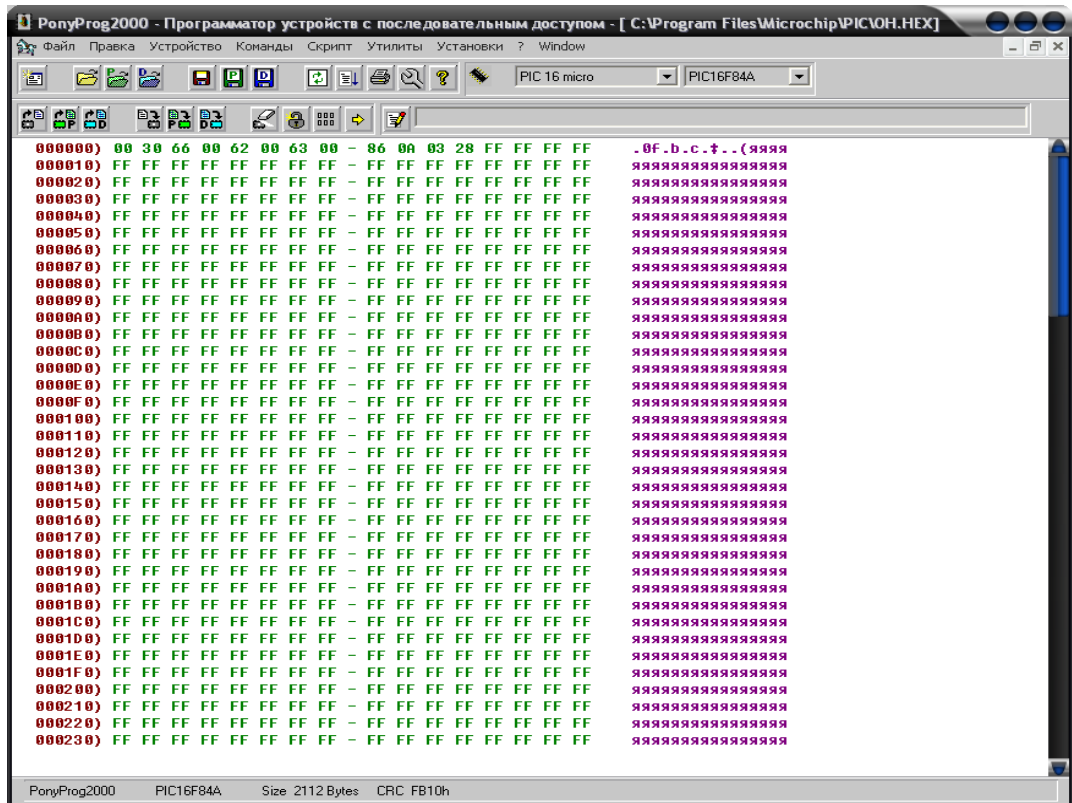


Рисунок 3.3 – Вікно програматора Pony Prog

Для початку запису програми в МК потрібно або клацнути по кнопці Записати пристрій або Команди > Записати все(рис. 3.4).

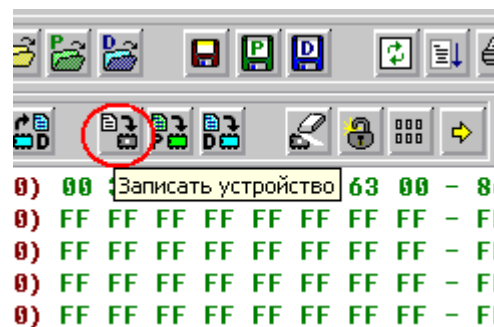


Рисунок 3.4 – Початок запису програми

Після закінчення процесу запису і перевірки з'явиться повідомлення про успішний запис. Якщо видається повідомлення про помилку запису, клацніть по піктограмі з намальованим на ній замком, і у вікні бітів конфігурації, що

відкрилося, поглянете, чи є галочка в маленькому віконці з назвою CP (біт захисту) (рис. 3.5).

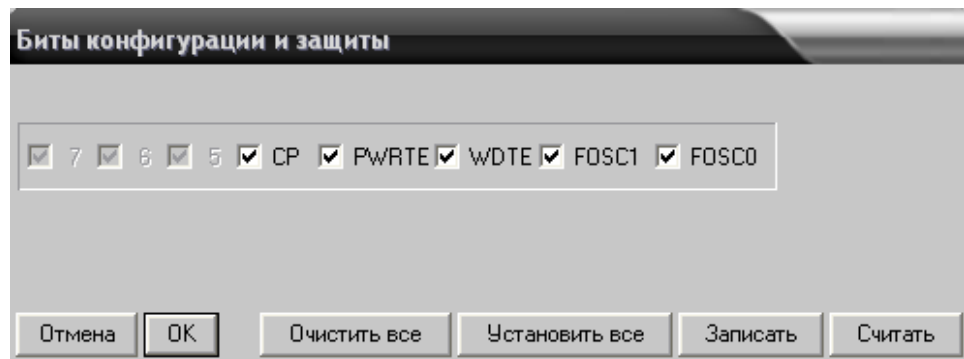


Рисунок 3.5 – Вікно конфігурації та захисту

Якщо вона встановлена, то спочатку запишіть, а потім вважайте встановлені програмою біти конфігурації (міняти їх або вносити які-небудь зміни не потрібно). Після причитування переконаєтеся, що лічене те ж саме, що записане. У інших випадках, це робити не потрібно.

Використовуючи апаратну частину програматора Pony Prog, можна "поєднати" її з новішою програмою, яка називається ICProg105. Це дуже зручно, оскільки використовуючи одну і ту ж апаратну частину програматора, можна працювати в одній з двох програм Pony Prog и ICProg105.

Даний програматор це одна з останніх версій. Він не вимагає інсталяції, досить розпакувати архів, і програма готова до роботи. ICProg105 має в своєму складі дуже велику базу МК, що підтримуються, а так само FLASH і EEPROM (рис. 3.6).

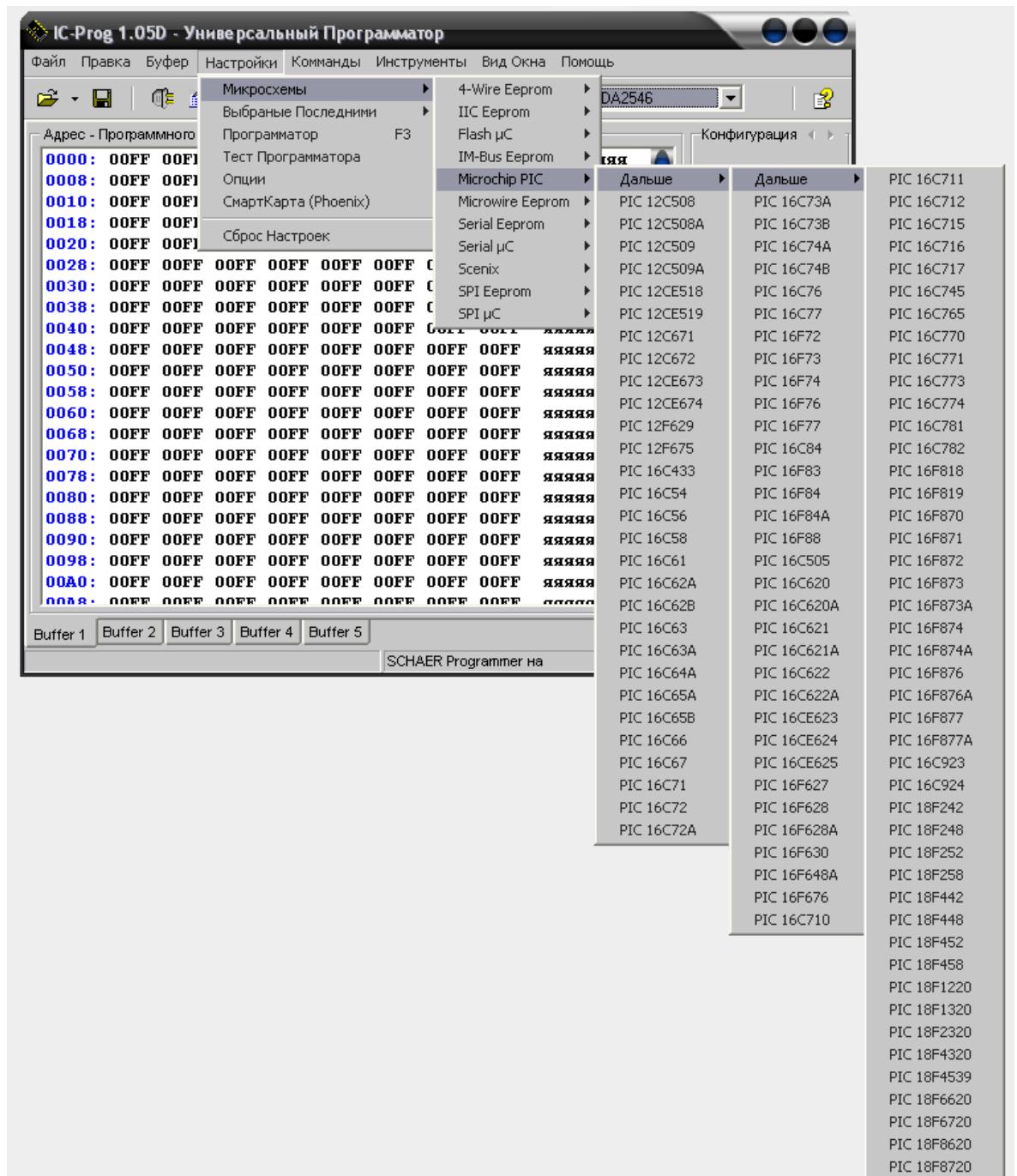


Рисунок 3.6 – Интерфейс і список пристроїв, що підтримуються

Програма ICProg105 працює так само і з фізичними програматорами, підтримуючими LPT порти.

3.5 Команди Асемблера для PIC

Мнемонічні команди які (за рідким виключенням) відповідають інструкціям процесора обчислювальної системи. Трансляція програми у виконуваний машинний код виробляється асемблером – програмою-транслятором, яка і дала мові асемблера його назву (таблиця 3.1).

Таблиця 3.1 – Мнемоніка асемблера

Мнемоніка	Опис
Байт – орієнтовані команди	
ADDWF f, d	Скласти вміст регістрів W і f.
ANDWF f, d	Виконується по бітне “І” вмісту регістрів W і f.
CLRF f	Очистити вміст регістра f і встановити прапор Z.
CLRW	Очистити вміст регістра W і встановити прапор Z.
COMF f, d	Інвертувати всі біти в регістрі f.
DECF f, d	Декремент вмісту регістра f.
DECFSZ f, d	Декремент вмісту регістра f з галуженням.
INCF f, d	Інкремент вмісту регістра f.
INCFSZ f, d	Інкремент вмісту регістра f з галуженням.
IORWF f,d	По бітне “АБО” вмісту регістрів W і f.
MOVF f,d	Вміст регістра f пересилається в регістр адресата.
MOVWF f	Переслати вміст W в f.
NOP	Немає операції.
RLF f, d	Виконується циклічне зрушення вліво вмісту регістра f через біт C регістра STATUS.
RRF f, d	Виконується циклічне зрушення управо вмісту регістра f через біт C регістра STATUS.
SUBWF f, d	Відняти вміст регістра W з вмісту регістра f.
SWAPF f, d	Поміняти місцями старший і молодший пів байти регістра f.

XORWF f,d	Що по бітне "Виключає "АБО" вмісту регістрів W і f.
Біт - орієнтовані команди (b-від 0 до 7)	
BCF f, b	Встановити в 0 біт b регістра f.
BSF f, b	Встановити в 1 біт b регістра f.
BTFSC f, b	Якщо біт b в регістрі f =1, то виконується наступна інструкція Якщо біт b в регістрі f =0, то наступна інструкція не виконується (пропускається, замість неї виконується "віртуальний" NOP), а команда виконується за 2 м.ц.
BTFSS f, b	Якщо біт b в регістрі f=0, виконується наступна інструкція Якщо біт b в регістрі f=1, то наступна інструкція не виконується (пропускається, замість неї виконується "віртуальний" NOP), а команда виконується за 2 м.ц.
Команди операцій з константами (до – від 0 до 255)	
ADDLW k	Вміст регістра W складається з 8 – розрядною константою k.
SUBLW k	Відняти вміст регістра W з 8 – розрядної константи k.
MOVLW k	Переслати константу k в регістр W.
ANDLW k	Виконується по бітне "І" вмісту регістра W і 8 – розрядної константи k.
IORLW k	Виконується по бітне "АБО" вмісту регістра W і 8 – розрядної константи k.
XORLW k	Виконується по бітне "Виключає АБО" вмісту регістра W і 8 – розрядної константи k.
Команди керування	
CALL	Виконати умовний перехід.
GOTO k	Виконати безумовний перехід.
RETURN	Повернення з підпрограми.
RETLW k	Повернення з підпрограми зі встановленою константою.
RETFIE	Повернення з підпрограми обробки переривань.
CLRWDT	Скидання WDT і перед дільника.

SLEEP	Перехід в режим зниженого енергоспоживання (у “сплячий режим”).
OPTION	Переслати вміст регістра W в регістр OPTION.
TRIS	Переслати вміст регістра W в регістр TRIS.

3.6 Висновки до розділу 3

У третьому розділі дипломного проекту розглянуті принцип роботи з пам'яттю команд мікроконтролерів PIC, технологією FLASH–EEPROM, регістрами, оперативною пам'яттю мікроконтролера PIC16C84, читання даних EEPROM мікроконтролерів PIC, робота з мікроконтролерами PIC в програмі Pony Prog та команди Асемблера для МК PIC.

4 ОХОРОНА ПРАЦІ

4.1 Загальні питання з охорони праці

В Законі України «Про охорону праці» [10] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі.

4.2 Правові та організаційні основи охорони праці

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно Закону України «Про охорону праці» [10] до статті 3 законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" та інших правових актів.

Обов'язки працівників щодо додержання вимог нормативно-правових актів з охорони праці, відповідальність робітників всіх категорій за порушення вимог Закону України «Про охорону праці» щодо охорони праці [10] (ст. 44) та структура організації/виробництв системи управління охорони праці визначені безпосередньо «Інструкцією на робоче місце № 1», та

іншими затвердженими власними нормативними актами з питань охорони праці (правилами, нормами, регламентами, положеннями, стандартами, інструкціями та іншими документами, обов'язковими до виконання), тобто тих, що діють на підприємстві/організації, і визначені в НПАОП 0.00-6.03-93 «Порядок опрацювання та затвердження власником нормативних актів про охорону праці, що діють на підприємстві»[11].

4.3 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до НПАОП 0.00-4.12-05 «Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці» [12].

Також впроваджені організаційні заходи з пожежної безпеки - навчання і перевірку знань відповідно до НПАОП 0.00-8.24-05 «Перелік робіт з підвищеною небезпекою» [13].

4.4 Аналіз стану умов праці та вимоги до приміщення

Робота над створенням дипломного проекту проходитиме в приміщенні відповідної установи (компанії, підприємстві тощо). Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером. Геометричні розміри приміщення зазначені в таблиці 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, м ²	25
Об'єм, м ³	75

Згідно з ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень [14] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам. Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації відповідно НАПБ.А.01.001-2014 «Правила пожежної безпеки в Україні» [24].

4.5 Вимоги до організації робочого місця

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця згідно ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [15] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність в таблиці 4.2.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ □ 800

Екран монітору знаходиться на відстані 0.8 м, клавіатура має можливість регулювання кута нахилу 5-15°. Отже, за всіма параметрами робоче місце відповідає ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [15].

Приміщення кабінету знаходиться на другому поверсі трьох поверхової будівлі і має об'єм 78 м³, площу — 18 м². У цьому кабінеті обладнано три місця праці, з яких два укомплектовані ПК.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум на робочому місці знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет має бути оснащений переносним вуглекислотним вогнегасником ВВК-5.

4.6 Навантаження та напруженість процесу праці

Під час виконання робіт використовують ПК та периферійні пристрої, що призводить до навантаження на окремі системи організму.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

При роботі наявні психофізіологічні небезпечні та шкідливі фактори: фізичне перевантаження, розумове перенапруження та ін.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви для розробників програм тривалістю 15 хв через кожен годину роботи.

4.7 Аналіз небезпечних та шкідливих факторів при роботі на персональному комп'ютері

Роботу, пов'язану з електронно-обчислювальними машинами (далі - ПК) з відео дисплейними терміналами (далі - ВДТ), у тому числі на тих, які мають робочі місця, обладнані ПК з ВДТ і периферійними пристроями (далі - ПП), виконують із забезпеченням виконання НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями [16], які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ПК з ВДТ і ПП.

Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема ПК та периферійні пристрої.

Робочі місця мають відповідати вимогам ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин», ПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» та ДБН В.2.5-28:2018 «Природне і штучне освітлення [15-17].

Це передбачає, що визначена виробнича діяльність пов'язана з наявністю певної кількості небезпечних та/або шкідливих виробничих

факторів згідно Закону України «Про забезпечення санітарного та епідемічного благополуччя населення» та НПАОП 0.00-8.24-05 «Перелік робіт з підвищеною небезпекою» [18-19].

Робота ПК та периферійних пристроїв супроводжує виділення багатьох хімічних речовин, зокрема озону, оксидів нітрогену та аерозолів (високодисперсних частинок тонера).

4.8 Пожежна безпека

Пожежна безпека при застосуванні ПК забезпечується системою запобігання пожежі, протипожежного захисту та організаційно-технічними заходами

Згідно ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою [20], таке приміщення, площею 25 м², відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння. Відповідно до норм первинних засобів пожежогасіння пропонується використовувати вогнегасник порошковий ВП-2 в кількості 1 шт.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходиться пожежонебезпечні речовини і матеріали відповідно до ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» [20] відносяться до пожежонебезпечної зони класу П-Іа.

Це обумовлено тим, що в приміщенні знаходяться тверді горючі та важкозаймісті речовини та матеріали. Приміщенню, у якому розташоване робоче місце, присвоюється II ступень вогнестійкості.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол згідно ДСТУ 4462.0.02.2005 «Охорона природи. Комплекс стандартів у сфері поводження з відходами. Загальні вимоги» [21].

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза, що фільтрує, з коробкою марки «В» із сірою відміткою забарвлення – захист від неорганічних газів (хлор, фтор, бром, сірководень, сірковуглець, хлорціан, галогени), а цей фільтр не захистить від СО (тобто від чадного газу).

4.9 Електробезпека

На робочому місці периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту повинні відповідати НПАОП 40.1-1.01-97 «Правила безпечної експлуатації електроустановок» [23].

Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування виконана як окрема групова трипровідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

4.10 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [14] і наведені в таблиці 4.3.

Таблиця 4.3 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	Температура С ⁰	Відносна вологість %	Швидкість руху повітря, м/с
Холодна	легка-1 а	22 - 24	40 – 60	0,1
Тепла	легка-1 а	23 - 25	40 – 60	0,1

Дане приміщення обладнане системами опалення, кондиціонування повітря або припливно-витяжною вентиляцією. У приміщенні на робочому місці забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [14].

Рівні позитивних і негативних іонів у повітрі також мають їм відповідати.

Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі користувача, з метою його провітрювання. Контроль параметрів мікроклімату в холодний і теплий період року здійснюється не менше 3-х разів на зміну (на початку, середині, в кінці).

4.11 Освітлення робочого місця

У проекті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин), спектральний складом випромінюваного світла, близький до сонячного. При експлуатації ПК виконується зорова робота IV в розряді точності (середня точність). При цьому нормована освітленість на робочому місці (Ен) рівна 200 лк. Джерелом природного освітлення є сонячне світло.

У приміщенні, де розташовані ПК передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 «Природне і штучне освітлення» [17]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН В.2.5-28:2018 «Природне і штучне освітлення» [17] і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (4.1)$$

де S_b – площа віконних прорізів, м²;

S_n – площа підлоги, м².

$$S_n = a \cdot b = 5 \cdot 5 = 25 \text{ м}^2$$

$$S_{\text{вік}} = 1/8 \cdot 25 = 3,125 \text{ м}^2$$

Приймаємо 2 вікна площею $S = 1,6 \text{ м}^2$ кожне.

Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби передбачається обладнати приміщення, довжина якого складає 5 м, ширина 5 м, світильниками ЛПО2П, оснащеними лампами типа ЛБ (дві по 80 Вт) з світловим потоком 5400 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників N виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, м²; $S = 25 \text{ м}^2$;

Z – поправочний коефіцієнт світильника (для стандартних світильників $Z = 1.1 - 1.3$) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575;

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм.

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 25 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} = 1,99$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

Потужність електроосвітлювальної установки з урахуванням місцевого освітлення визначається за формулою:

$$N = \frac{n \cdot W + (0,1 \div 0,2) \cdot n \cdot W}{1000}, \text{ кВт} \quad (4.3)$$

де n – розрахункова кількість ламп для освітлення даного приміщення;

W – потужність однієї лампи, Вт;

$(0,1 \div 0,2)$ – додаткова потужність для ламп місцевого освітлення, Вт

$$N = \frac{3 \cdot 160 + 0,2 \cdot 3 \cdot 160}{1000} = 0,576 \text{ кВт}$$

4.12 Шум, вібрація та електромагнітне випромінювання

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів, а також зовнішніми чинниками, коливається у межах 50–65 дБА відповідно ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку» [22]. Шум такої інтенсивності на тлі високого ступеня напруженості праці негативно впливає на функціональний стан користувачів.

Для зниження шуму на шляху його поширення передбачається розміщення в приміщенні штучних поглиначів. Для зниження рівня шуму стелю або стіни вище 1.5 - 1.7 метра від підлоги повинні облицьовуватися звукопоглинальним матеріалом з максимальним коефіцієнтом звукопоглинання в області частот 63-8000 Гц. Додатковим звукопоглинанням в КВТ можуть бути фіранки, підвішені в складку на відстані 15-20 см. Від огорожі, виконані з щільної, важкої тканини. У приміщенні з ПК коректований рівень звукової потужності не перевищує 45 дБ.

Вібрація на робочому місці в приміщенні, що розглядається, відповідає нормам ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку» [22].

Допустимий рівень вібрацій на робочому місці: - для 1 ступеня шкідливості до 3 дБ; - для 2-3 - 1-6 дБ; - для 3 - більше 6 дБ.

4.13 Розрахунок захисного заземлення

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом НПАОП 40.1-1.01-97. Правила безпечної експлуатації електроустановок [23], приміщення в якому проводяться всі роботи відносяться до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4...0,99. Взаємну екрануючу дію горизонтального заземлювача

(з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η .

Визначимо необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (4.4)$$

де $R_{пр.з.}$ – опір природних заземлювачів; R_d – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (А.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Приблизне значення питомого опору глини приймаємо $\rho = 40$ Ом·м (табличне значення).

Розрахунковий питомий опір ґрунту, $\rho_{розр.}$, Ом·м, визначається відповідно для вертикальних заземлювачів $\rho_{розр.в}$, і горизонтальних $\rho_{розр.г}$, Ом·м за формулою:

$$\rho_{розр.} = \psi \cdot \rho, \quad (4.5)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів I кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{розр.в} = 1,7$ і горизонтальних $\rho_{розр.г} = 5,5$ Ом·м.

$$\rho_{розр.в} = 1,7 \cdot 40 = 68 \text{ Ом} \cdot \text{м}$$

$$\rho_{\text{розр.г}} = 5,5 \cdot 40 = 220 \text{ Ом}\cdot\text{м}$$

Розрахуємо опір розтікання струму вертикального заземлювача R_B , Ом, за (А.5).

$$R_B = \frac{\rho_{\text{розр.в}}}{2 \cdot \pi \cdot l_B} \cdot \left(\ln \frac{2 \cdot l_B}{d_{\text{ст}}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_B}{4 \cdot t - l_B} \right), \quad (4.6)$$

де l_B – довжина вертикального заземлювача (для труб - 2–3 м; $l_B=3$ м);

$d_{\text{ст}}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{\text{ст}}=0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (3.6):

$$t = h_B + \frac{l_B}{2}, \quad (4.7)$$

де h_B – глибина закладання вертикальних заземлювачів (0,8 м);

$$\text{тоді } t = 0,8 + \frac{3}{2} = 2,3 \text{ м}$$

$$R_B = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

Визначаємо теоретичну кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_B :

$$n = \frac{2 \cdot R_B}{R_d} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.8)$$

Визначаємо коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_B = 0,57$ (табличне значення).

Визначаємо необхідну кількість вертикальних заземлювачів з урахуванням коефіцієнта використання n_B , шт:

$$n_B = \frac{2 \cdot R_B}{R_d \cdot \eta_B} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.9)$$

Визначаємо довжину з'єднувальної стрічки горизонтального заземлювача l_c , м:

$$l_c = 1,05 \cdot L_B \cdot (n_B - 1), \quad (4.10)$$

де L_B – відстань між вертикальними заземлювачами, (прийняти за $L_B = 3$ м);

n_B – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м}$$

Визначаємо опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки) R_Γ , Ом:

$$R_\Gamma = \frac{\rho_{\text{розр.}\Gamma}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h_\Gamma}, \quad (4.11)$$

де $d_{\text{см}}$ – еквівалентний діаметр смуги шириною b , $d_{\text{см}} = 0,95b$, $b = 0,15$ м;

h_{Γ} – глибина закладання горизонтальних заземлювачів (0,5 м);

l_c - довжина з'єднувальної стрічки горизонтального заземлювача l_c , м

$$R_{\Gamma} = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1 \text{ Ом}$$

Визначаємо коефіцієнт використання горизонтального заземлювача η_c . відповідно до необхідної кількості вертикальних заземлювачів n_B .

Коефіцієнт використання з'єднувальної смуги $\eta_c=0,3$ (табличне значення).

Розраховуємо результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_B \cdot R_{\Gamma}}{R_B \cdot \eta_c + R_{\Gamma} \cdot n_B \cdot \eta_B} \leq R_d. \quad (4.12)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_d$$

4.14 Висновки до розділу 4

В даному розділі розроблені рекомендації з охорони праці, техніки безпеки при роботі на комп'ютері. Проведений аналіз умов праці, вплив шкідливих та небезпечних чинників на здоров'я людини. Визначено параметри і характеристики приміщення. Приведені рекомендації щодо організації робочого місця, електробезпеки та пожежної безпеки.

Наведені розміри приміщення та значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці.

ВИСНОВКИ

В дипломній роботі виконано розроблення програми тестування внутрішніх вузлів МК PIC16C84A з використанням програмно-інструментального середовища налаштування MP-LAB IDE.

В розробленій програмі реалізовані наступні функції PIC контролера PIC16F84:

- тестування портів вводу-виводу;
- тестування виконання команд.

Для реалізації дипломного проекту використані наступні продукти фірми Microchip:

- програмно-інструментальне середовище налаштування MP-LAB IDE;
- макроасемблер MPASM з пакету програм Microchip MPLAB;

Для програмування PIC контролера PIC16F84 використана програма Pony Prog (<http://www.lancos.com/prog.html>).

В дипломному проекті розроблені заходи щодо охорони праці в умовах виробництва.

Результати дипломної роботи можуть бути використанні в навчальному процесі кафедри комп'ютерних наук та інженерії при вивченні дисципліни «Цифрова схемотехніка».

ПЕРЕЛІК ПОСИЛАНЬ

1. Заец Н.И. Радиолобительские конструкции на PIC-микроконтроллерах.: Пер. с англ. – К.: МК– Пресс, 2005. – 192 с. , ил.
2. Кристиан Тавернье. PIC-микроконтроллеры. Практика применения / Тавернье Кристиан.– М.: ИП РадиоСофт, 2003. – 273с.
3. Предко М. PIC-микроконтроллеры: архитектура и программирование. –М: ДКМ Пресс, 2010. – 266 с.
4. Магда Ю.С. Микроконтроллеры PIC: архитектура и программирование. –М: ДМК Пресс, 2009. – 240 с.
5. Ульрих В. А. Микроконтроллеры PIC16X7XX. СОЛОН-Пресс: 2005. - 320 с.
6. Анна и Манфред Кениг. Полное руководство по PIC – микроконтроллерам. Перевод с немецкого МК-Пресс, 2007. – 256 с: ил.
7. Предко М. Руководство по микроконтроллерам. Том. II / Предко М. – М.: Постмаркет, 2001. – 488с.
8. С. М. Рюмик. 1000 и одна микроконтроллерная схема. Выпуск 1. – М: Додэка-XXI, 2010. – 356 с: ил. +CD.
9. Барри Брей. Применение микроконтроллеров PIC18. – К: КОРОНА-ВЕК, МК-Пресс, 2008. – с. 576.
10. Закон України «Про охорону праці». Вводиться в дію Постановою ВР № 2695-ХІІ від 14.10.92, ВВР, 1992, № 49, ст.669. - Режим доступу: www.URL: https://zakon.rada.gov.ua/laws/show/2694-12.
11. НПАОП 0.00-6.03-93 «Порядок опрацювання та затвердження власником нормативних актів про охорону праці, що діють на підприємстві». Режим доступу: https://dnaop.com/html/43271/doc-ДНАОП_0.00-6.03-93.
12. НПАОП 0.00-4.12-05. «Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці». Режим доступу: <https://zakon3.rada.gov.ua/laws/show/z0231-05>.

13. НПАОП 0.00-8.24-05 «Перелік робіт з підвищеною небезпекою». Режим доступу: - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0232-05>.
14. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень». - Режим доступу: <https://zakon.rada.gov.ua/rada/show/va042282-99>.
15. ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин». - Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0007282-98>.
16. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0508-18>.
17. ДБН В.2.5-28:2018 «Природне і штучне освітлення». - Режим доступу: [http://online.budstandart.com/ua/catalog/klassifikator-minregionstroya/09._dbn__\(derzhavnii_23018/V.2.5-28-2018+79885-detail.html](http://online.budstandart.com/ua/catalog/klassifikator-minregionstroya/09._dbn__(derzhavnii_23018/V.2.5-28-2018+79885-detail.html).
18. Закон України «Про забезпечення санітарного та епідемічного благополуччя населення». - Режим доступу: <https://zakon.rada.gov.ua/laws/show/4004-12>.
19. НПАОП 0.00-8.24-05 «Перелік робіт з підвищеною небезпекою». - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0232-05>.
20. ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою». - Режим доступу: http://online.budstandart.com/ua/catalog/klassifikator-minregionstroya/00._klasyfikatsiia_23686/v._tekhniichnii_norm_224/v.1_zahaln_otekhniich_234/v.1.1_zakhyst_viid_n_235/V.1.1-36-2016+70714-detail.html.
21. ДСТУ 4462.0.02:2005 «Охорона природи. Комплекс стандартів у сфері поводження з відходами. Загальні вимоги». - Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0351609-07>.

22. ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку». - Режим доступу: <https://zakon2.rada.gov.ua/rada/show/va037282-99>.

23. НПАОП 40.1-1.01-97. Правила безпечної експлуатації електроустановок. - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0011-98>.

24. НАПБ.А.01.001-2014 «Правила пожежної безпеки в Україні». - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0252-15>.

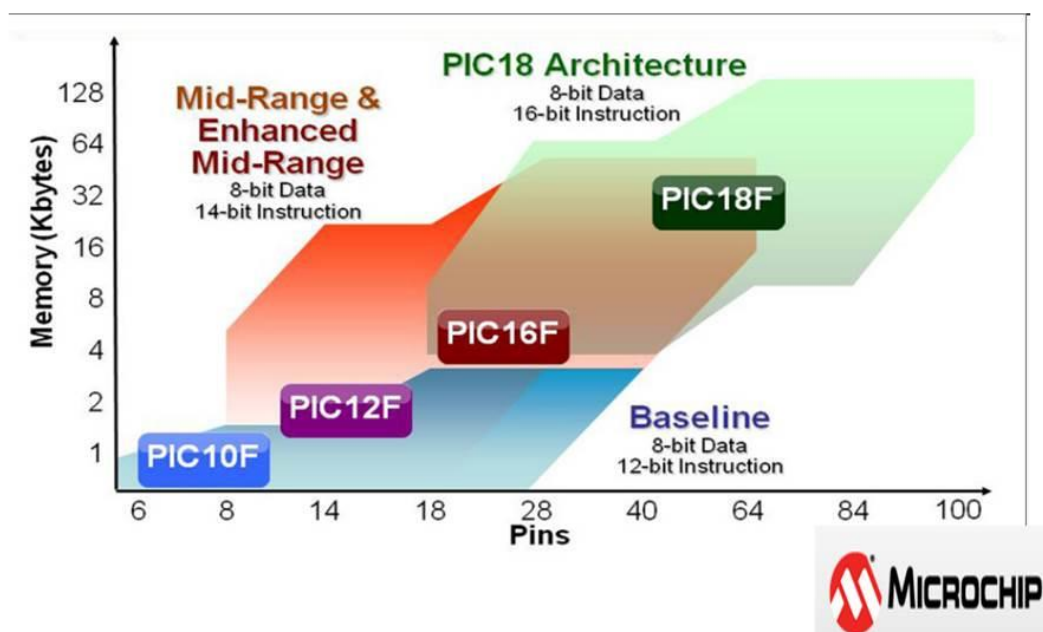
ПРЕЗЕНТАЦІЯ

Актуальність теми та мета дипломного проекту

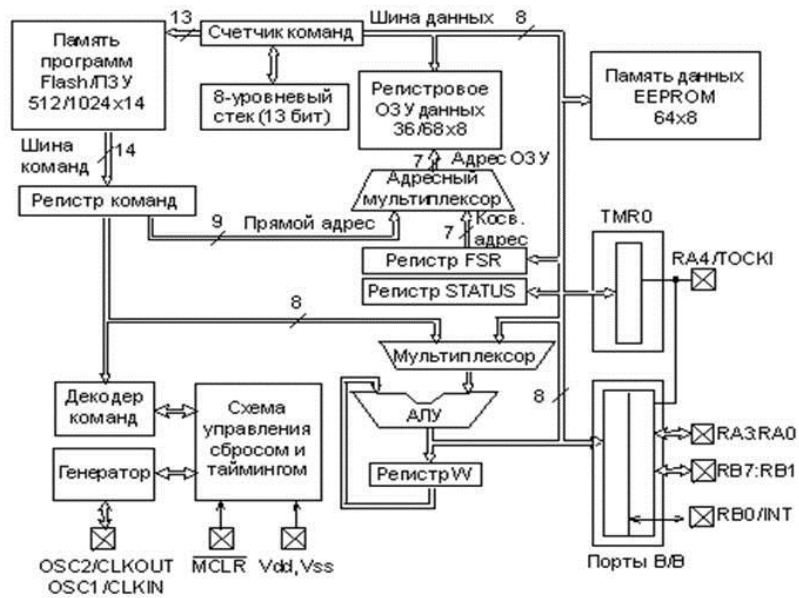
- Використання мікроконтролерів не лише приводить до підвищення техніко-економічних показників (вартості, надійності, споживаної потужності, габаритних розмірів), але і дозволяє скоротити час розроблення виробів. Використання мікроконтролерів в системах керування забезпечує досягнення високих показників ефективності при низькій вартості.
- **Мета дипломного проекту** - розроблення програми тестування PIC контролерів з застосуванням програмного середовища MPLAB IDE.
- **Для виконання поставленої задачі в дипломній роботі необхідно:**
- розробити програму тестування PIC контролера PIC16F84 :
 - портів вводу-виводу;
 - виконання команд.

2

Характеристики мікроконтролерів PIC фірми Microchip



Структурна схема МК підгрупи PIC16F8X



Діалогове вікно вибору мови програмування в середовищі MPLAB

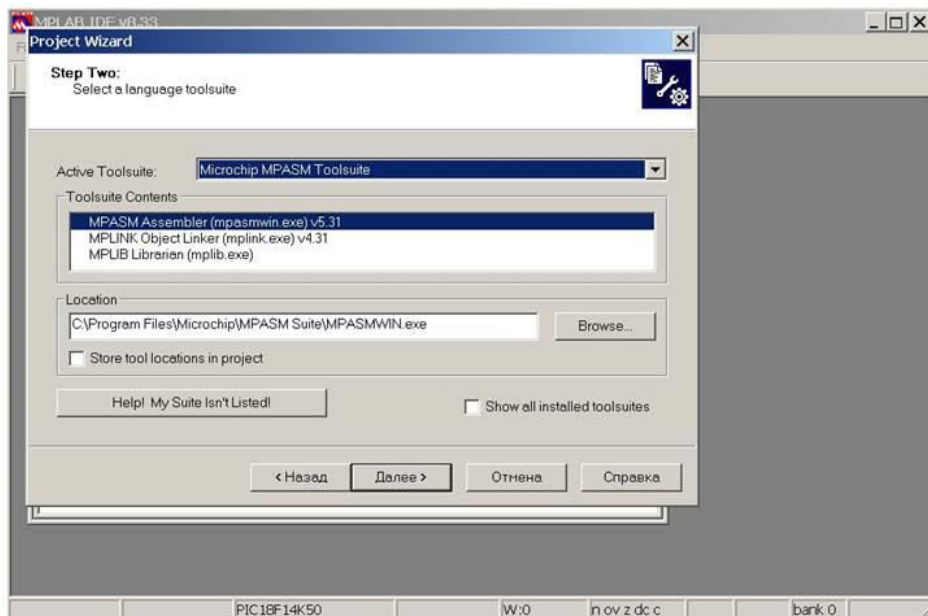
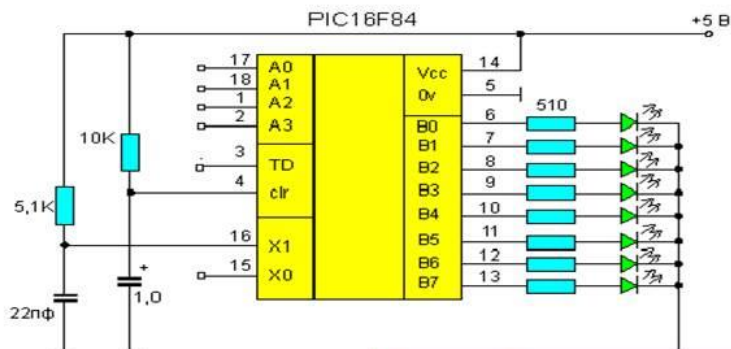


Схема пристрою індикації на базі МК PIC16F84



Parameter Name	Value
Program Memory Type	Flash
Program Memory (KB)	1.75
CPU Speed (MIPS)	5
RAM Bytes	68
Data EEPROM (bytes)	64
Timers	1 x 8-bit
Temperature Range (C)	-40 to 85
Operating Voltage Range (V)	2 to 6
Pin Count	18

Робоче вікно з фрагментом тексту програми

```

LIST      P=16f84
;
MOVLW    0
TRIS     6
OPTION
LOOP     SLEEP
INCF     6,F
GOTO     LOOP
END

```

Базовий код для демонстраційної програми

```

; Секція заголовка
; опис операційних регістрів
TMR0      EQU   01h
PC        EQU   02h
STATUS    EQU   03h
FSR       EQU   04h

; регістри вводу/виводу
CNTRLPORT EQU   05h
DATAPORT  EQU   06h

; комірки оперативної пам'яті
SCRATCH   EQU   0Ch
DIGIT     EQU   0Dh

; біти регістра STATUS
C         EQU   0h
DC        EQU   1h

```

Базовий код для демонстраційної програми (продовження)

```

; керуючі регістри
TRISA     EQU   85h
TRISB     EQU   86h

; слова ініціалізації для портів вводу/виводу
INITA     EQU   B'00000000'
INITB     EQU   B'00000000'

; Робоча секція
; початок коду виконання
    ORG    0
    GOTO   BEGIN
    ORG   100h
BEGIN

```


Фрагмент лістингу програми ініціалізації портів на виведення інформації

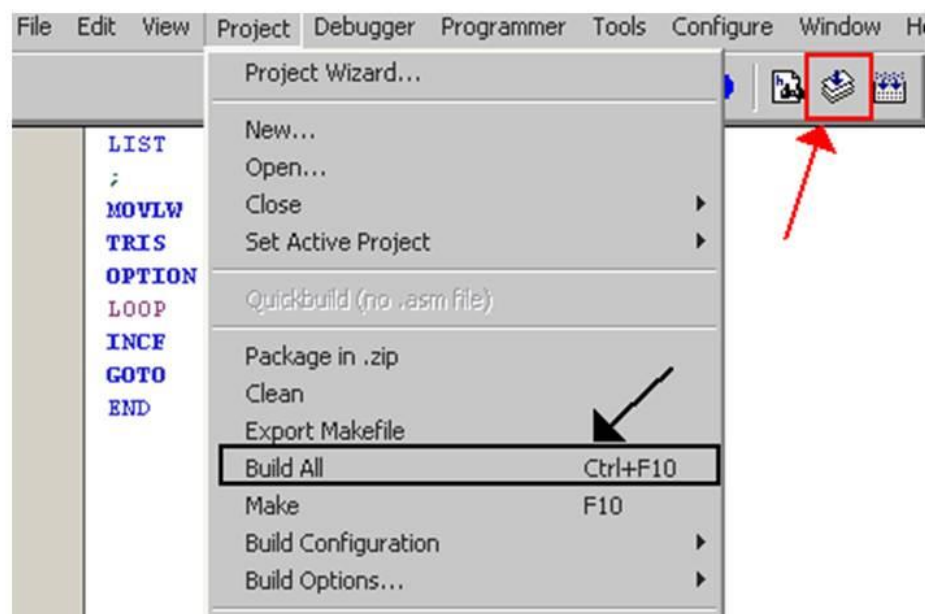
```

;Ініціалізація порту А
BCF     STATUS,RP      ;Вибір банка 0
CLRF    CNTRLPORT      ;Очистити регістр CNTRLPORT
MOVLW   INITA          ;Завантажити В'00000000' в регістр W
BSF     STATUS,RP      ;Вибір банка 1
MOVWF   TRISA          ;Всі розряди порту А встановити як виходи

;Ініціалізація порту В
BCF     STATUS,RP      ; Вибір банка 0
CLRF    DATAPORT       ;Очистити регістр DATAPORT
MOVLW   INITB          ;Завантажити В'00000000' в регістр W
BSF     STATUS,RP      ;Вибір банка 1
MOVWF   TRISB          ;Всі розряди порту В встановити як виходи

```

Компіляція розробленої програми



```

Build | Version Control | Find in Files
-----
Debug build of project 'C:\Program Files\Microchip\PIC\OH.mcp' started.
Preprocessor symbol '__DEBUG' is defined.

-----
Clean: Deleting intermediary and output files.
Clean: Deleted file "C:\Program Files\Microchip\PIC\OH.mcs".
Clean: Done.
Executing: "C:\Program Files\Microchip\MPASM Suite\MPASMWIN.exe" /q /p16F84 "OH.asm" /I"OH.lst" /e"OH.err" /d__DEBL
Warning[205] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 1 : Found directive in column 1. (LIST)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 3 : Found opcode in column 1. (MOVLW)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 4 : Found opcode in column 1. (TRIS)
Warning[224] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 4 : Use of this instruction is not recommended.
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 5 : Found opcode in column 1. (OPTION)
Warning[224] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 5 : Use of this instruction is not recommended.
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 7 : Found opcode in column 1. (INCF)
Warning[203] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 8 : Found opcode in column 1. (GOTO)
Warning[205] C:\PROGRAM FILES\MICROCHIP\PIC\OH.ASM 9 : Found directive in column 1. (END)
Executing: "C:\Program Files\Microchip\MPASM Suite\mplink.exe" /p16F84 "OH.o" /u__DEBUG /z__MPLAB_BUILD=1 /z__MPL
MPLINK 4.30.01, Linker
Copyright (c) 2009 Microchip Technology Inc.
Errors : 0

Loaded C:\Program Files\Microchip\PIC\OH.cof.

Debug build of project 'C:\Program Files\Microchip\PIC\OH.mcp' succeeded.
Preprocessor symbol '__DEBUG' is defined.

-----
BUILD SUCCEEDED

```

Запуск програми моделювання МК

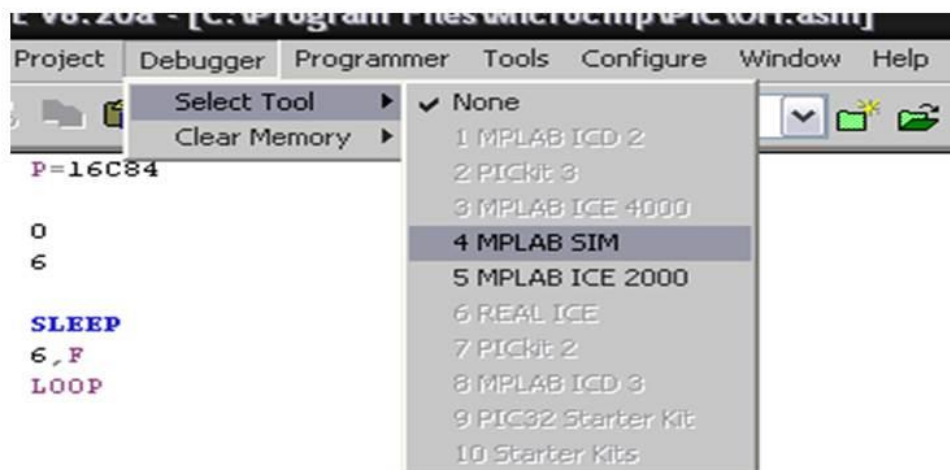
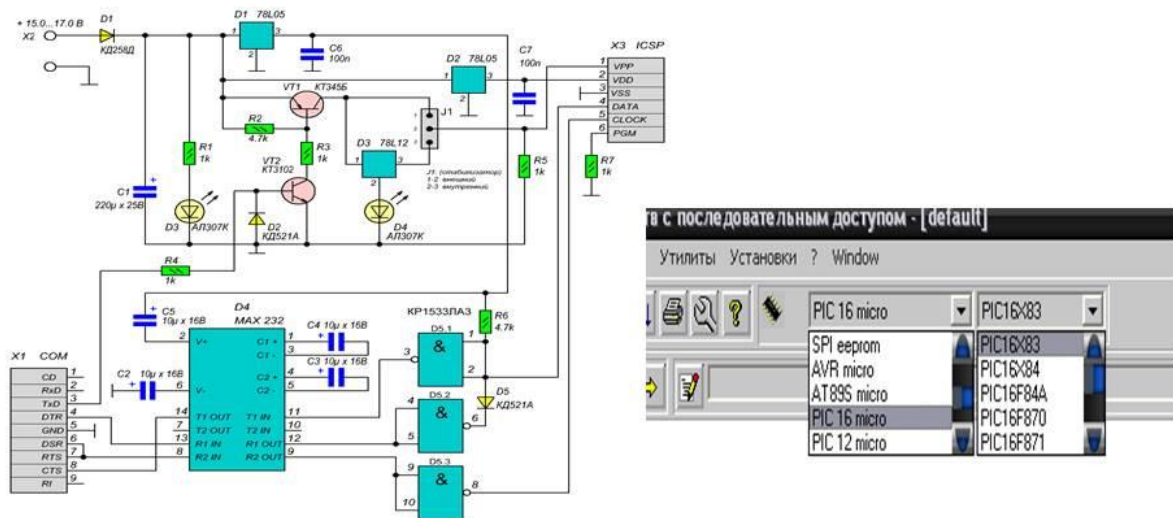


Схема проматора EXTRA – PIC та застосування програми Pony Prog



Висновки

- В дипломній роботі виконано розроблення програми тестування внутрішніх вузлів PIC контролерів з використанням програмно-інструментального середовища налаштування MPLAB.
- В розробленій програмі реалізовані наступні функції PIC контролера PIC16F84:
 - тестування портів вводу-виводу;
 - тестування виконання команд.
- Для реалізації дипломного проекту були використані наступні продукти фірми Microchip:
 - програмно-інструментальне середовище налаштування MPLAB IDE;
 - макроасемблер MPASM з пакету програм Microchip MPLAB;
- Для програмування PIC контролера PIC16F84 була використана програма Pony Prog (<http://www.lancos.com/prog.html>).
- В дипломному проекті розроблені заходи щодо охорони праці в умовах виробництва.