Embedded Programming in C for 32-Bit PIC Microcontroller

40 Hours

Course Description

This course is designed to provide a comprehensive introduction to embedded programming using the C language for 32-bit PIC microcontrollers. Participants will gain hands-on experience with MPLAB X, the industry-standard integrated development environment (IDE) for PIC microcontrollers. The course covers essential topics such as creating projects, configuring clock speeds, utilizing header files, implementing delays, controlling displays (seven-segment, LCD, and dot matrix), communication protocols (I2C and EEPROM), interrupts, and real-time clock applications. The course also explores analog interfacing, sensor integration (DHT11 and DS3231), and generating square waves.

Audience

This course is suitable for electronics engineers, embedded systems enthusiasts, and anyone interested in learning the fundamentals of embedded programming for 32-bit PIC microcontrollers using the C language. Participants should have basic knowledge of electronics and programming concepts.

Pre-requisite Knowledge/Skills

- Basic understanding of electronics and digital logic
- Familiarity with C programming language basics

Course Objectives

- Develop a strong foundation in embedded programming principles.
- Understand the MPLAB X IDE and create projects for 32-bit PIC microcontrollers.
- Configure clock speeds and write configuration words for PIC microcontrollers.
- Implement header files and delays for efficient program execution.
- Control various display types, including seven-segment, LCD, and dot matrix displays.
- Explore communication protocols such as I2C and EEPROM interfacing.
- Master the concepts of interrupts and real-time clock applications.
- Interface with the analog world, including sensors like DHT11 and DS3231.
- Generate square waves and understand pulse width modulation (PWM).

Course Outline

Module 1: Creating Our First C Program

- What Is MPLAB X
- Creating a Project in MPLAB X
- The Initial Comments
- The Speed of the Clock
- Writing the Configuration Words
- Our First Program
- Setting the PORTS
- Exercise 1-1
- The Hexadecimal Number System
- The First Program Listing
- Analysis of Listing 1-1
- Simulating the Program in MPLAB X
- Solution to Exercise 1-1
- Summary

Module 2: Header Files and Delays

- What Are Header Files and Why We Use Them?
- Creating a Header File
- Using the Header File
- Slowing the PIC Down
- Creating a Delay
- The Variable Delay Subroutine
- What Is a Subroutine
- The Analysis of the Delay Subroutine
- Creating a 1ms Delay
- Good Programming Practice
- The Algorithm
- The Flowchart
- Our First Useful Program
- The Algorithm
- The Allocation List
- The Flowchart for the Program
- Creating the Project

- Analysis of Listing 2-3
- Downloading the Program to a Prototype Board
- Extending the Program to the Crossroads Traffic Lights
- The Algorithm
- Analysis of Listing 2-4
- Simulating the Program Within MPLAB X
- Summary

Module 3: The Seven-Segment Display

- Controlling a Seven-Segment Display
- The Seven-Segment Display
- Common Anode Seven-Segment Display
- Common Cathode Seven-Segment Display
- Controlling the Display with the PIC
- The Seven-Segment Display Program
- The Algorithm
- The Flowchart
- The Listing for the Seven-Segment Display Program
- Analysis of Listing 3-1
- Improving the Seven-Segment Display Program
- The Problem with the Program
- Arrays
- Using Pointers
- Analysis of Listing 3-2
- The Improved Program
- Exercise 3-1
- Solution to Exercise 3-1
- Summary

Module 4: The LCD

- The 1602 LCD
- Instruction or Command Mode
- Data Mode
- The Control Pins of the LCD
- The LCD Header File for PORTE
- Analysis of Listing 4-1
- Analysis of Listing 4-2
- Creating Your Own Symbols to Display on the LCD

- The Pixel Maps
- The 8-Bit Binary Values for the Four Special Characters
- The Program “Pixel to Matrix”
- The Special Character Program
- The Program Listing for the Special Character Program
- Analysis of Listing 4-3
- Summary

Module 5: The Dot Matrix Display

- The 8 by 8 Dot Matrix Board
- The Single Dot Matrix Display
- The Max7219 Driver IC
- Writing to the Max7219
- Analysis of Listing 5-1
- Creating the Data for Each Row in the Two-Dimensional Array
- Controlling Four 64-Bit Dot Matrix Boards
- Analysis of Listing 5-3
- Analysis of Listing 5-2
- A Program to Scroll Text on the Matrix Display
- Summary

Module 6: Communication

- The 25LC256 EEPROM
- What Is an EEPROM Device
- Writing to the EEPROM
- Reading from the EEPROM
- The Connections of the EEPROM on the Explorer 16 Development Board
- The Algorithm for the EEPROM Program
- Analysis of Listing 6-1
- The Setting of the CKE and CKP Bits
- The Message “Ann Ward”
- Using the Parallel Master Port
- The Data for the PMMODE Control Register
- The PMCON Control Register
- The PMAEN Control Register
- Analysis of Listing 6-2
- Analysis of Listing 6-3
- Summary

Module 7: The I2C Communication

- The I2C Protocol
- I2C Communication Protocol
- Writing to the Slave
- The Start Bit
- The Address Bits and Control Bit in the First Byte
- The Acknowledgment Bit
- The Data Byte
- The Stop Bit
- Reading from the Slave
- The NACK or Not Acknowledgment Bit
- Analysis of the I2C Protocol Header File
- The I2C Expander Module
- The LCD2004
- The Connections of the I2C Expander
- Analysis of Listing 7-2
- The Program to Use the I2C Expander to Control the LCD2004
- Analysis of Listing 7-3
- Summary

Module 8: Interrupts

- Interrupts
- The Fetch and Execute Cycle
- The Program Counter or “PC”
- Single Vectored Interrupts
- Multivectored Interrupts
- How Does the PIC Keep Track of Where It Must Go and Where It Must Get Back To?
- The Stack and Its Main Use
- The Sources of Interrupts in a 32-Bit PIC
- Analysis of Listing 8-1
- Using More Than One Interrupt Source with Single Vector Mode
- Analysis of Listing 8-2
- Interrupt Priority
- Multivectored Interrupts with Priority
- Analysis of Listing 8-3
- Summary

Module 9: The Real-Time Clock

- The External Crystal 32.768kHz Oscillator
- The 24-Hour LCD Clock Program
- Analysis of Listing 9-1
- The TM1637 and the Four Seven-Segment Displays
- The TM1637 Driver IC
- Analysis of Listing 9-2
- Summary

Module 10: The Real-Time Clock and the DS3231

- The DS3231 RTC Module
- The Order the DS3231 Expects the Data
- The Program Algorithm
- Displaying the Temperature
- Binary Numbers
- Adding Binary Numbers
- Reading the Temperature from the TC72
- Examples of the Two's Complement Process
- Using the UART and a Terminal Software
- Using the Alarms of the DS3231
- The UART
- The UxMODE Control Register
- The UxSTA Register
- Using Tera Term
- Connecting the Devices to the PIC32
- Analysis of Listing 10-1
- Summary

Module 11: The RTCC Module

- The RTCC Module of the 32-Bit PIC
- BCD (Binary-Coded Decimal)
- Displaying the 32-Bit Value on the LCD
- RTCC Module Program
- Analysis of Listing 11-1
- Understanding Instruction on Line 504 RTCALRM = 0x8403;
- Summary

Module 12: The Real Analog World

- The Real-World Signals
- An Analog Signal
- The Digital World
- A Simple Voltmeter Program
- The Algorithm of the Voltmeter Program
- Analysis of Listing 12-1
- The ADC Process
- The Acquisition Time
- The 4–20mA Transducer
- Controlling Two Analog Inputs
- Analysis of Listing 12-2
- Summary

Module 13: The DHT11 Transducer

- The DHT11 Humidity and Temperature Sensor
- Communicating with the DHT11
- The Use of a Pull-Up Resistor
- The DHT11 Program Listing
- Analysis of Listing 13-1
- The Logical OR and AND Truth Tables
- A Design Procedure
- Know the Events You Want to Control
- Analysis of Listing 13-2
- Simulating the Program in MPLAB X
- Summary

Module 14: Creating a Square Wave

- Creating a Simple Square Wave
- Using the Output Compare Module (OCMP) of the PIC32
- Using the Logic Analyzer Within MPLAB X
- Creating Some Musical Notes
- Creating a PWM Square Wave with the OC1MP Module
- The Average of a Square Wave
- Analysis of Listing 14-4
- Varying the Brightness of a Lamp
- Analysis of Listing 14-5

- Summary