Chapter 1
The PIC18 Microcontroller

Objective

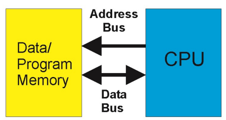
- Von Neumann and Harvard Architectures
- Compare the contrasts in microprocessors and microcontrollers.
- Explain the concept of embedded system.

Section 1.1
Von Neumann and Harvard Architectures

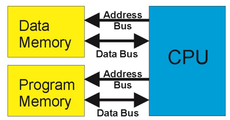
Harvard vs von Neumann Architecture

- von Neumann – uses the same address and data bus for code ROM space and data RAM space.
- Harvard – uses different address and data buses to access code ROM space and data RAM space.
- von Neumann architecture is slow because ROM and RAM cannot be accessed simultaneously.
- Implementing Harvard architecture is expensive.

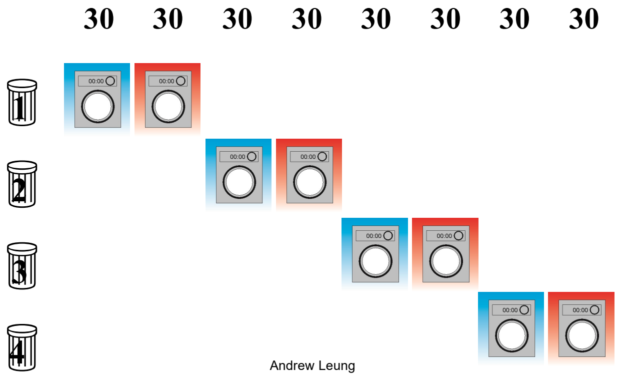
von Neumann



Harvard



Von Neumann Architecture is Slow

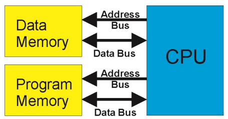


5

Advantage of Harvard Architecture
(easy for pipeline implementation)

6

Disadvantage of Harvard Architecture



- µP 64-bit data bus and 32-bit address bus needs about 200 wires.

7

Section 1.2
microprocessors and microcontrollers

8

General-purpose microprocessor

- CPU for Computers
- No RAM, ROM, I/O on CPU chip itself
- Example：Intel’s x86, Motorola’s 680x0

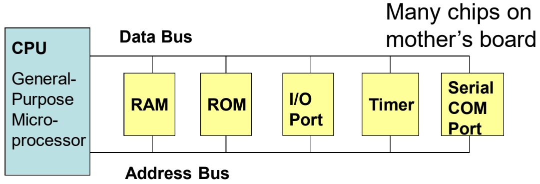
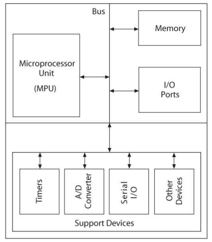


Figure 1-1 (a) General-Purpose Microprocessor System

Microcontrollers

- Embedded Systems
  - Operations managed behind the scenes by a microcontroller
- Microcontroller (MCU)
  - An integrated electronic computing device that includes three major components on a single chip
    - Microprocessor (MPU)
    - Memory
    - I/O (Input/Output) ports



Microcontrollers

- Support Devices (peripheral)

  a device attached to a computer
  - Timers (counting number of pulses)
  - A/D converter (Analog to Digital)
  - Serial I/O (example keyboard to main board)
- All components connected by common communication lines called the system bus.

Microcontroller

- A smaller computer
- On-chip RAM, ROM, I/O ports...
- Example：Motorola’s 6811, Intel’s 8051, Zilog’s Z8 and PIC 18

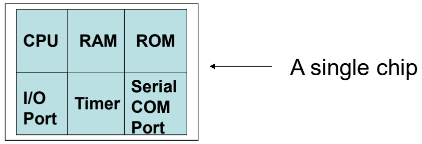


Figure 1-1 (b) Microcontroller

Microprocessor v.s. Microcontroller

Microprocessor
- CPU is stand-alone, RAM, ROM, I/O, timer are separate
- Designer can decide on the amount of ROM, RAM and I/O ports.
- General-purpose
- Expensive

Microcontroller
- CPU, RAM, ROM, I/O and timer are all on a single chip
- Fix amount of on-chip ROM, RAM, I/O ports
- Single-purpose
- Cheap
- For applications in which cost, power and space are critical

13

Section 1.3 Embedded System

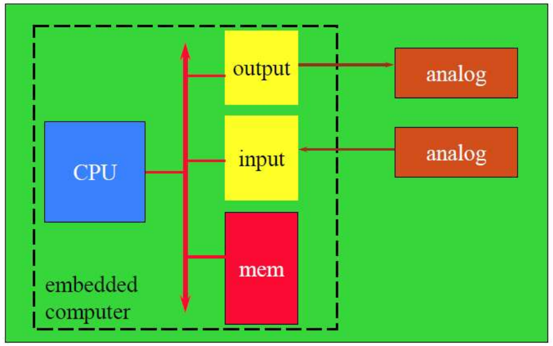
14

Embedded System

- Embedded system means the processor is embedded into that application.
- An embedded product uses a microprocessor or microcontroller to do specified tasks only.
- In an embedded system, there is only one application software that is typically burned into ROM.
- Some embedded products using microcontrollers. Example：printer, keyboard, video game player, door opener, copier, ABS, fax machine, camera, cellular phone, keyless entry, microwave.

15

Embedded System



16

Processors in Embedded Systems

- Which is your choice for an embedded product？
- microcontroller
  - cost down
  - embedded processor ＝ microcontroller
  - In future, an entire computer on a chip
  - high-end embedded systems may use microprocessors

Three criteria in Choosing a Microcontroller

1. meeting the computing needs of the task efficiently and cost effectively
   - speed, the amount of ROM and RAM, the number of I/O ports and timers, size, packaging, power consumption
   - easy to upgrade
   - cost per unit
2. availability of software development tools
   - assemblers, debuggers, C compilers, emulator, simulator, technical support
3. wide availability and reliable sources of the microcontrollers.

Basic Data Format (8-bit)

- Unsigned Integers: All eight bits (Bit7 to Bit0) represent the magnitude of a number
  - Range 00 to FF in Hex and 0 to 255 in decimal

Basic Data Format (8-bit)

- Signed Integers: Seven bits (Bit6 to Bit0) represent the magnitude of a number.
  - The eighth bit (Bit7) represents the sign of a number. The number is positive when Bit7 is zero and negative when Bit7 is one.
  - Positive numbers: 00 to 7F (0 to 127)
  - Negative numbers: 80 to FF (-1 to -128)
  - All negative numbers are represented in 2’s complement

Basic Data Format (8-bit)

- Binary Coded Decimal Numbers (BCD)
  - 8 bits of a number divided into groups of four, and each group represents a decimal digit from 0 to 9
  - Four-bit combinations from A through F in Hex are invalid in BCD numbers
    - Example: 0010 0101 represents the binary coding of the decimal number 25 which is different in value from 25_H.

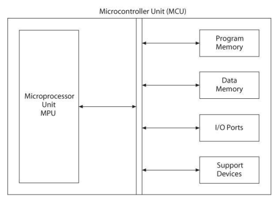
Basic Data Format (8-bit)

- American Standard Code for Information Interchange (ASCII)
  - Seven-bit alphanumeric code with 128 combinations (00 to 7F)
  - Represents English alphabet, decimal digits from 0 to 9, symbols, and commands

Section 1.4
Overview of PIC18

PIC18F Microcontroller Families

- PIC microcontrollers are designed using the Harvard Architecture which includes:
  - Microprocessor unit (MPU)
  - Program memory for instructions
  - Data memory for data
  - I/O ports
  - Support devices such as timers



Microprocessor Unit

- Includes Arithmetic Logic Unit (ALU), Registers, and Control Unit
  - Arithmetic Logic Unit (ALU)
    - Instruction decoder
      - 16-bit instructions
    - Status register that stores flags
      - 5-bits
    - WREG – working register
      - 8-bit accumulator

Microprocessor Unit

- Registers
  - Program Counter (PC)
    - 21-bit register that holds the program memory address while executing programs
  - Bank Select Register (BSR)
    - 4-bit register used in direct addressing the data memory
  - File Select Registers (FSRs)
    - 12-bit registers used as memory pointers in indirect addressing data memory
- Control unit
  - Provides timing and control signals to various Read and Write operations

PIC18F - Address Buses

- Address bus
  - 21-bit address bus for program memory addressing capacity: 2 MB of memory
  - 12-bit address bus for data memory addressing capacity: 4 KB of memory

Data Bus and Control Signals

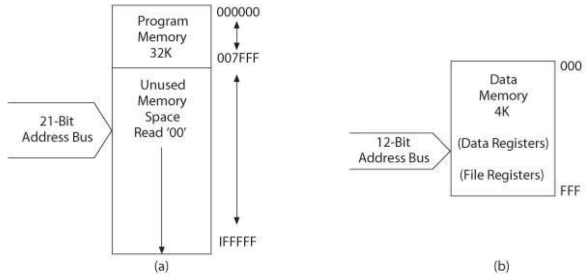
- Data bus
  - 16-bit instruction/data bus for program memory
  - 8-bit data bus for data memory
- Control signals
  - Read and Write

PIC18F452/4520 Memory

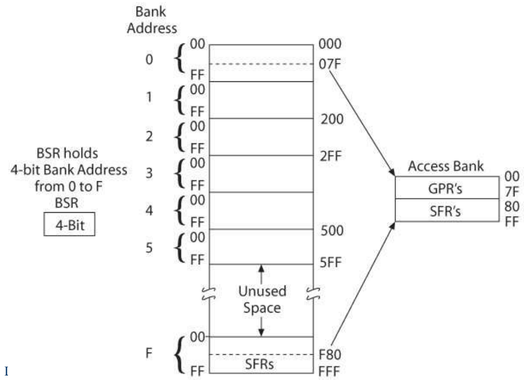
- Program Memory: 32 K
  - Address range: 000000 to $007FFF_H$
- Data Memory: 4 K
  - Address range: 000 to FFF_H
- Data EEPROM
  - Not part of the data memory space
  - Addressed through special function registers

PIC18F452/4520 Memory

- Program memory
- Data memory



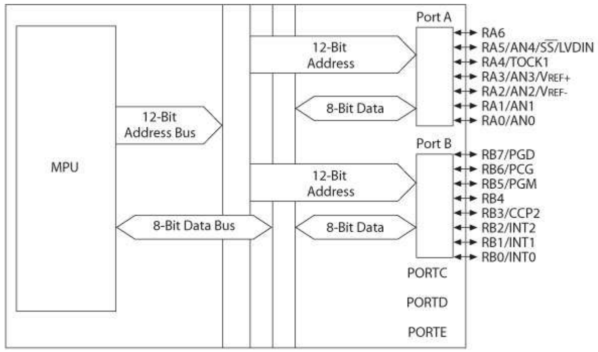
PIC18F452/4520 – Data Memory with Access Banks



PIC18F452 I/O Ports

- Five I/O ports
  - Digital IO is the most fundamental mode of connecting a MCU to external world. The interface is done using what is called a PORT.
  - The IO ports can be programmed as input or output.
  - PORT A through PORT E
  - Most I/O pins are multiplexed
  - Generally have eight I/O pins with a few exceptions
  - Addresses already assigned to these ports in the design stage
  - Each port is identified by its assigned SFR

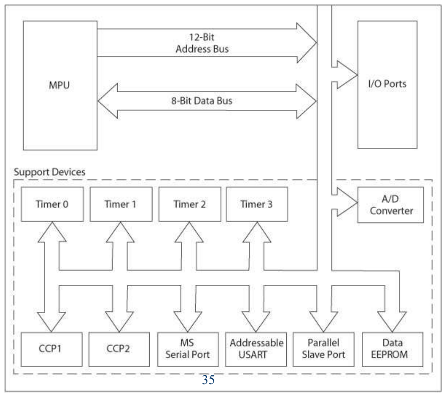
PIC18F452—I/O Ports A and B



MCU Support Devices

- Timers
  - Capture, Compare and PWM (CCP Modules)
- Serial Communications
  - Master Synchronous Serial Port (MSSP)
  - Addressable USART
- A/D converter
- Parallel Slave Port (PSP)
- Data EEPROM

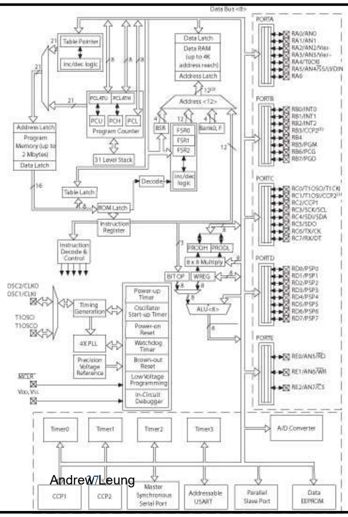
MCU Support Devices



PIC18F Special Features

- Sleep mode
- Watchdog timer (WDT)
- Code protection
- In-circuit serial programming
- In-circuit debugger

- PIC18F4X2 Architecture Block Diagram



List of Selected Microcontroller Families from Microchip

Part No.	Program OTP/Flash	EE PROM	RAM	Total Pins	I/O Pins	Analog ADC	Analog Comp.	Digital Timers/ WDT	Digital Serial I/O	CCP/ ECCP	Max Speed MHz	Instruction Size	Total Instructions
10F200	256x12 Flash		16	8	4			1-8 bit, 1-WDT			4	12-bit	33
10F220	256x12 Flash		16	8	4	2x8-bit	-	1-8 bit, 1-WDT			8	12-bit	33
12F510	1536x12 Flash		38	8	6	3x8-bit	1	1-8 bit 1-WDT			8	12-bit	33
16F506	1536x12 Flash		67	14	12	3x8-bit	2	1-8 bit 1-WDT			20	12-bit	33
16C55A	768x12 OTP		24	28	20			1-8 bit 1-WDT			40	12-bit	33
16CR58B	3072x12 ROM		73	18	12			1-8 bit 1-WDT			20	12-bit	33
12F683	2048x14 Flash	256	128	8	6	4x10-bit	1	1-16 bit, 2-8 bit, 1-WDT			20	14-bit	35
16F687	2048x14 Flash	256	128	20	18	12x10-bit	2	1-16 bit, 1-8 bit, 1-WDT	EU/I^{2}C/ SPI		20	14-bit	35
18F1230	2048x16 Enh Flash	128	256	18-28	16	4x10-bit	3	2-16 bit 1-WDT	EU		40	16-bit	77
18F4520	16384x16 Enh Flash	256	1536	40-44	36	13x10-bit	2	1-8 bit, 3-16 bit, 1-WDT	EU/ MI^{2}C /SPI	1/1	40	16-bit	77
18F6527	24576x16 Enh Flash	1024	3936	64	54	12x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	40	16-bit	77
18F8622	32768x16 Enh Flash	1024	3936	80	70	16x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	40	16-bit	77
18F96J60	32768x16 Flash		2048	100	72	16x10-bit	2	2-8 bit, 3-16 bit, 1-WDT	2EU/ 2 - MI^{2}C /SPI	2/3	42	16-bit	77
24FJ128GA-010	65536x16 Flash		8192	100-128	86	16x10-bit	2	5-16 bit, 1-WDT	2 –UART 2-I^{2}C/ SPI	5	32	16-bit	77

Abbreviations: 1) ADC: Analog-Digital Converter, 2) AUSART: Addressable USART, 3) CCP: Capture/Compare/PWM, 4) ECCP: Enhanced CCP, 5) EU: Enhanced USART, 6))Enh Flash: Enhanced Flash, 7) I^{2}C: Inter-integrated Circuit Bus, 8) MI^{2}C/SPI: Master I^{2}C /SPI, 9) OTP: One-Time Programmable, 10) SPI: Serial Peripheral Interface, 11) USART: Universal Synchronous/Asynchronous Receiver/Transmitter, 11) WDT: Watchdog Timer

PIC18F Instructions and Assembly Language

- Has 77 instructions
  - Earlier PIC family of microcontrollers have either 33 or 35 instructions
- In PIC18F instruction set, all instructions are 16-bit word length except four instructions that are 32-bit length

Instruction Description and Illustrations

- Copy (Move) 8-bit number (Literal) into W register
  - Mnemonics: MOVLW 8-bit
  - Binary format:

  0000 1110 XXXX XXXX (any 8-bit number)
- Copy (Move) contents of W register into PORTC (File)
  - Mnemonics: MOVWF PORTC, a
    - ('a' indicates that PORTC is in the Access Bank)
  - Binary format:

  0110 1110 1000 0010 (82_H is PORTC address)

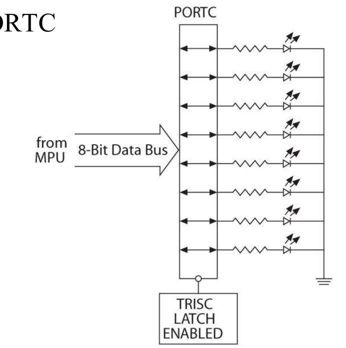
Illustrative Program: Displaying a Byte at an I/O Port

- Problem statement:
  - Write instructions to light up alternate LEDs at PORTC
- Hardware:
  - PORTC
    - bidirectional (input or output) port; should be setup as output port for display
  - Logic 1 will turn on an LED

41

Illustration

- Interfacing LEDs to PORTC



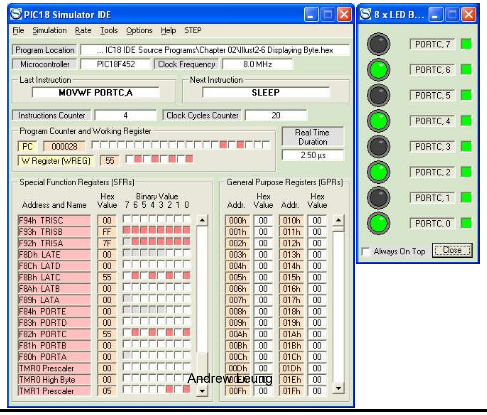
42

Illustration

- Program (software)
  - Logic 0 to TRISC sets up PORTC as an output port
  - Byte 55_H turns on alternate LEDs
    - MOVLW 00 ;Load W register with 0
    - MOVWF TRISC,0 ;Set up PORTC as output
    - MOVLW 0x55 ;Byte 55_H to turn on LEDS
    - MOVWF PORTC,0 ;Turn on LEDs

43

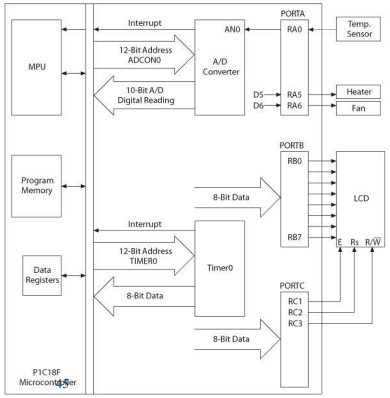
PIC18 Simulator IDE



44

Embedded System

Microcontroller-based
Time and
Temperature
System



Andrew Leung 45

45

You are able to

- Compare and contrast microprocessors and microcontrollers
- Describe the advantages of microcontrollers for some applications
- Explain the concept of embedded systems
- Discuss criteria to consider in choosing a microcontroller
- Explain the variations of speed, packaging, memory, and cost per unit and how these affect choosing a microcontroller
- Some basic feature of PIC18

Andrew Leung 46

46