CHAPTER (1) الباب الأول

مقدمة عن الميكروكونترلر

Introduction to Microcontrollers

1.1 Internal Organization of computers

1.1.1 The general components of computer

The internal working of every working computer can be broken into three parts as shown in Fig. 1-1:

1. CPU (central processing unit).
2. Memory
3. Input/output (I/O) devices.
4. Bus system.

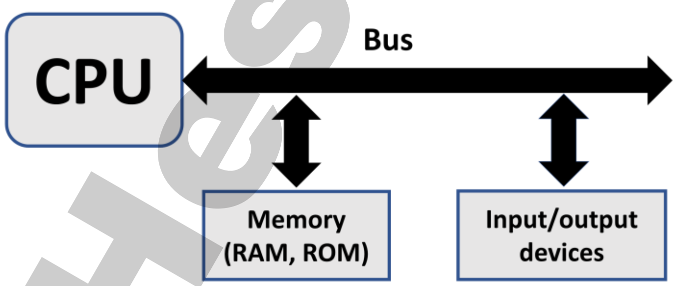


Fig. 1-1: Main components of computer systems.

The function of the CPU is to execute information stored in memory. The function of I/O devices (such as keyboard and video) is to provide a means of communicating with CPU. The CPU is connected through group of wires called a Bus. The bus inside a computer carries information from place to place just as a street bus carries people from one place to place. In every computer there are three types of buses (Fig. 1-2):

1. Address bus.
2. Data bus.
3. Control bus.

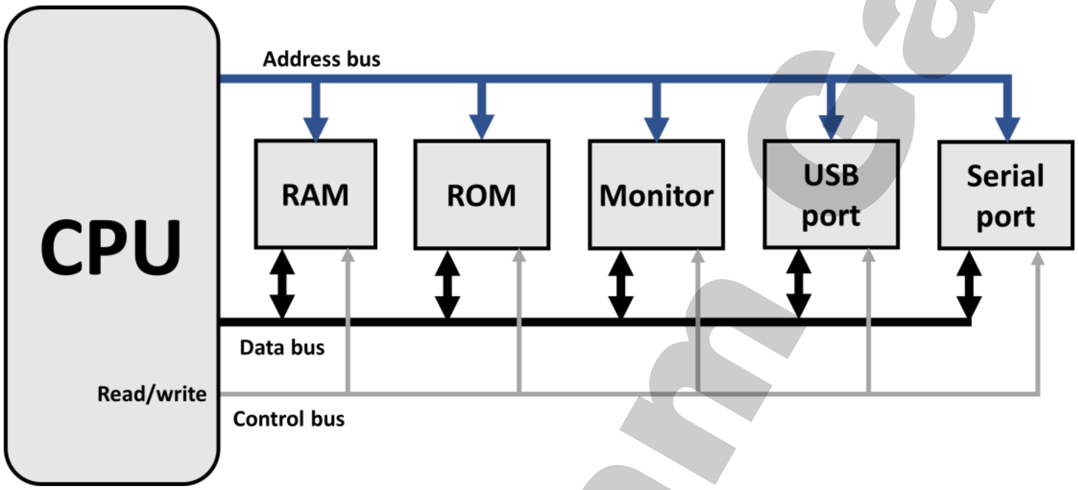


Fig. 1-2: Types of busses in computer system.

What is the purpose of each bus?

1. **Address bus:**
   - For a device (memory of or I/O) to be recognized (لكى يتم التعرف بها) by a CPU, it must be assigned to an address (يجب أن يكون لها عنوان) .
   - This address must be unique; no two devices are allowed to share the same address.
   - When the CPU put the required address on this bus, the device immediately responds to this address.

     عندما يضع الـ CPU عنوان الـ Device الذى يريد التحدث معه على الـ Bus فإن هذا الـ Device يجب أن يستجيب فوراً.

2. **Data bus:**
   - When the device responds to CPU, then the CPU begins to send or receive the data to or from the device via by using the data bus.

3. **Control bus:**

The control buses are used to provide read or write signals to the device to indicate if the CPU is asking for information or sending information.

يستخدم الـ Control bus لتوفير إشارة القراءة أو الكتابة من وإلى الـ Device للإشارة إلى هل أن الـ CPU يريد معلومه أو يريد إرسال معلومة إلى هذا الـ Device.

It is important to know that:

1. The more address buses available, the larger the number of devices that can be addressed. In other words, the number of address buses for a CPU determines the number of locations which it can communicate.
2. The address bus is a unidirectional bus, which means that the CPU uses the address bus only to send out addresses.

1. من المهم أن تعرف أنه بزيادة حجم الـ Address bus فإنه يمكن وضع العديد من الـ Devices. أو بالمعنى الأصح أن عدد الخطوط فى الـ Address bus يحدد عدد الأماكن التى يمكن أن يصل إليها الـ CPU.
2. الـ Address bus أحادى الأتجاه أى أنه يصدر فقط من الـ CPU ولا يمكن أن يستعمله أى Device أخر.

1.1.2- CPU and its relation to the Volatile and Non-Volatile storages

For the CPU to process information, the data must be stored in volatile (such as RAM) or non-volatile (such as hard-drive) storages. The function of RAM is to store temporary information that can change with time, such as the operating system and applications or even variables.

Programs are loaded from the hard drive into RAM to be processed by the CPU. The CPU cannot get the information from the disk directly because the disk is too slow. In other words, the CPU first seeks the information to be processed from RAM. If the data is not there, the CPU seeks it from the hard-drive a disk, and then it transfers the information to RAM. For this reason, RAM are sometimes referred to as *primary memory* and disks are called *secondary memory*.

1.1.3 Inside the CPU

A program code stored in memory provides instructions to the CPU to perform an action. The action can simply be adding data such or controlling a machine such as a robot. The function of the CPU is to fetch (إحضار) these instructions from memory and execute them. To perform the actions of fetch and execute, all CPUs are equipped (مجهز) with resources (بموارد) such as the following:

1. **Arithmetic and Logic Unit (ALU):**
   - The ALU section of the CPU is responsible for performing arithmetic functions such as add, subtract, multiply, and divide, and logic functions such as AND, OR, and NOT.
2. **Registers:**
   - The CPU uses registers to store information temporarily (بشكل مؤقت). The information could be two values to be processed, or the address of the value needed to be fetched from memory
   - Registers inside the CPU can be 8-bit, 16-bit, 32-bit, or even 64-bit registers, depending on the CPU.
3. **Program Counter:**
   - The function of the program counter is to point to the address of the next instruction to be executed.
4. **Instruction decoder:**

- The function of the instruction decoder is to interpret the instruction fetched into the CPU.

1.2 Microcontrollers and embedded processors

1.2.1 Microcontroller versus general-purpose microprocessor

1. The general-purpose microprocessor (Fig. 1-3):

- The microprocessor chip (such as the Pentium) cannot work alone because it doesn’t contain RAM, ROM and I/O ports on the microprocessor chip itself.
- To use the general-purpose microprocessor, we must add RAM, ROM, I/O ports externally to make them functional.
- Although the addition of external RAM, ROM, and I/O ports makes these systems much more expensive, they have the advantage of flexibility (المرونه) enabling the designer to decide on the amount of RAM, ROM, and I/0 ports needed to fit the task at hand.
- Such processors cannot be used in small size applications such as TV remote control, air conditioners, engine control systems… etc.

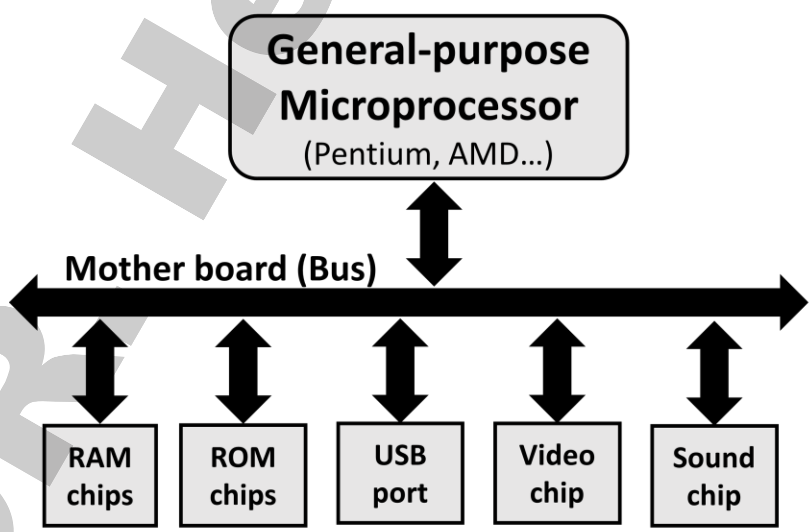


Fig. 1-3.

2. The microcontroller:

- The microcontroller is simply a computer in a single chip, which contains:
    a. Microprocessor (CPU)
    b. Fixed amount of memory (RAM and ROM).
    c. Input/output ports.
    d. Other peripherals (such as timers, Analog-to-digital ...)
- In other words, all of the previous components are embedded together in a single chip. Therefore, the designer cannot add any external memory, I/O or timers to it.
- The fixed amount of on-chip ROM, RAM, and number of I/0 ports in microcontrollers makes them ideal for many applications for example a TV remote control or small size robotics.

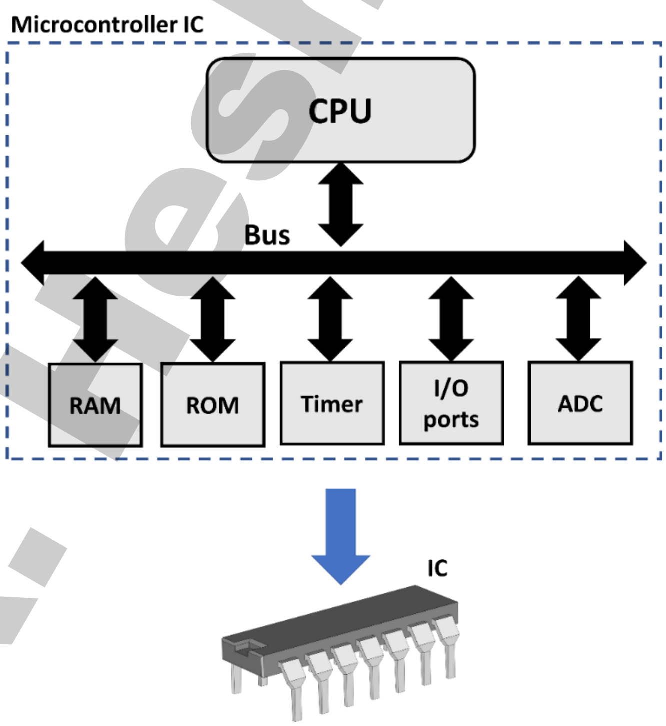


Fig. 1-4.

1.2.3 Microcontrollers for embedded systems

What's embedded systems?

- An embedded system is controlled by its own internal microprocessor (or microcontroller).
- In an embedded system, the microcontroller's ROM is loaded with a program for specific functions needed for the system.
- A printer is an example of an embedded system because the processor inside it performs one task only (getting the data and printing it).
- In contrast, a general-purpose computer (such as PC) can be used for any number of applications such as word processor, video game player, network server, or Internet terminal.
- The reason a PC can perform many tasks is that it has an operating system that can be used to execute many applications at the simultaneously فى نفس الوقت.
- In an embedded system, only one application software is burned into ROM to perform a specific task.

1.2.4 Choosing a microcontroller

When buying a microcontroller, some criteria (المعايير) must be taken into account:

1. Speed, what is the highest speed the microcontroller support?
2. Packaging, does it come in a 40-pin DIP (dual inline package) or a QFP (quad flat package)?
3. Power consumption (إستهلاك الكهرباء). This is especially critical for battery-powered products.
4. The amount of RAM and ROM on the chip.

5. The number of I/O pins.
6. Cost per unit. For example, some microcontrollers cost 50 cents per unit when purchased 100,000 units at a time.
7. The availability of an assembler, C language compiler and technical support.
8. The number of internal peripherals (Timers, ADCs, …etc).

1.2.5 PIC microcontroller program memory types

In microcontrollers, the ROM is used to store programs and since we will use PIC18 microcontroller family in this course, we will discuss in brief the different types of ROM technologies.

1. UV-EEPROM version of the PIC microcontroller

- Some of the PIC microcontrollers use UV-EPROM for storing program code.
- To use these kinds of chips for development a ROM burner, as well as a UV-EPROM eraser to erase the contents of ROM must be used
- The window on the UV-EPROM chip allows the UV light to erase the ROM.
- The problem with the UV-EPROM is that it takes around 20 minutes to erase the chip before it can be re-programmed again.

2. Flash-EEPROM version of the PIC microcontroller

- The flash version uses the letter “F” in the part number to indicate that the on-chip ROM is a flash memory, for example PIC18F452 means that the user program code is stored in a flash memory inside the chip.
- The flash version is ideal for fast development because flash memory can be erased in seconds.

- To program flash memories we need a ROM burner that support flash memory. Also, this burner can be used to erase the content of the flash memory easily.

3. **OTP version of the PIC microcontroller**
   - While the flash version of the PIC microcontroller is user for product development, the OTP (one-time-programmable) versions are typically used for mass production because it is cheaper than flash memory.
   - The problem with the OTP is that we cannot reprogram it if we want to modify the program.
4. **Masked version of the PIC microcontroller**
   - Microchip company (الشركة المصنعة لشرائح البيك) provide a service in which we can send the program and they will burn the it into the PIC during the fabrication process of the chip.
   - This chip is commonly preferred to as masked PIC, which is one of the of IC fabrication.
   - Masked PIC is the cheapest of all types, if the unit numbers are high enough. This is because there is a minimum order for the masked version of the PIC microcontrollers.

1.3-Internal Structure of PIC18F452 Micro-controller

1.3.1 Program memory organization of PIC18

- Since all PIC18 devices have 21-bits program counter, they can address up to 2 Mbytes of address space.
- The program memory map is divided into two spaces:
  1. User-memory space:

Depends on the flash memory size inside the microcontroller.
(In PIC18F452 the size of this memory is 32KB).

2. Non-existed memory space:
This space is non-existed, which cannot use it for any purposes.

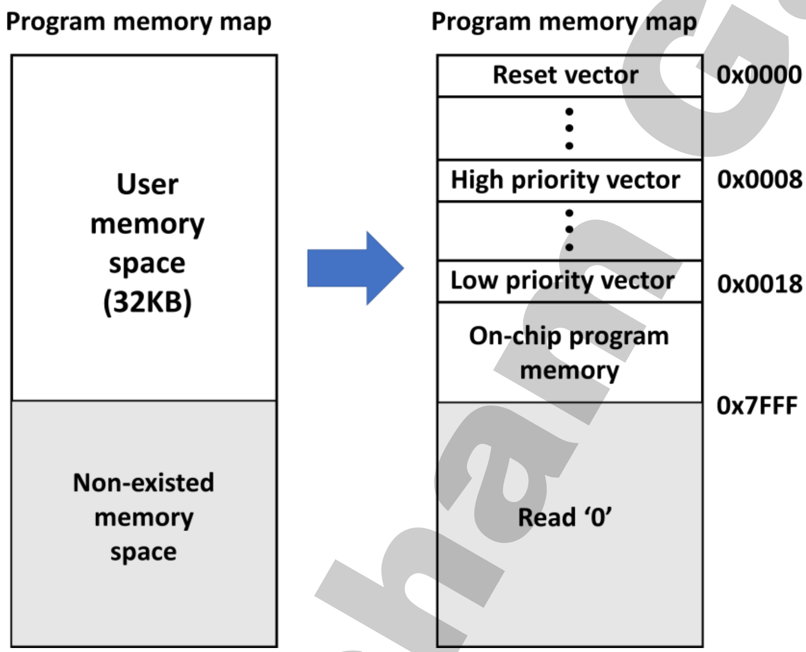
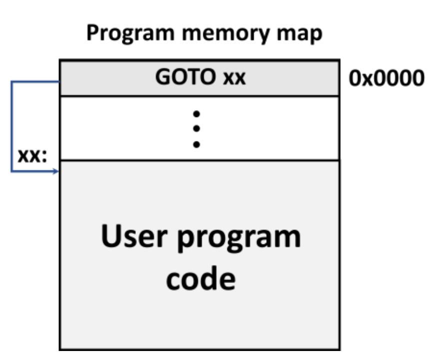


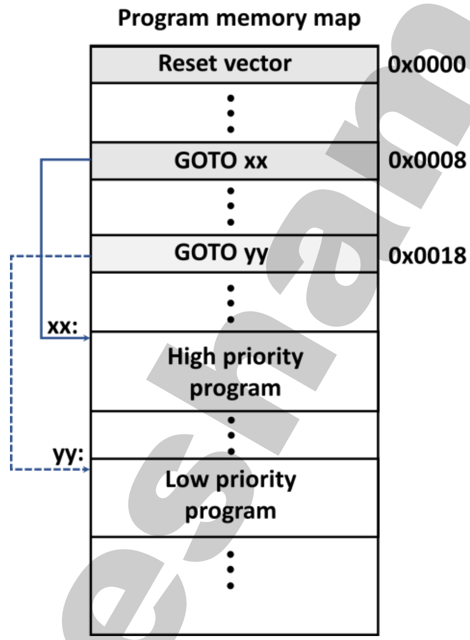
Fig. 1-5: Program memory organization of PIC18F452.

What’s RESET, High priority interrupt and Low priority interrupt vectors?

- When the CPU inside the microcontroller reset, it starts from address 0x0000. This address contains a GOTO instruction to the first address of the program code.



- If a high priority interrupt is occurred, the CPU automatically go to address 0x0008. This address contains a GOTO instruction to force the CPU to go to a specific memory location which contains the interrupt program that must be executed if a high priority signal is happened.
- The same thing for low priority interrupt vector, except that it existed at location 0x0018. This address also contains a GOTO instruction to force the CPU to go to the address of the low priority interrupt code.



1.1.2 Data memory organization of PIC18 family

- The data memory in general consists of 16 banks. Each bank has 256 bytes of data. But only 6 banks are used in PIC18F452, why???
- Because PIC18F452 has only 1536 bytes of data memory, the number of banks will be:

$$\text{No. of banks} = \frac{1536\,\text{bytes}}{256} = 6\ \text{banks}$$

- Switching between banks is done automatically when using high level language such as C or BASIC.
- The layout (خريطة) of all registers available in a microcontroller called Register File Map (RFM) as shown in Fig. 1-6.
- This map is extremely useful when programming the device, especially when using assembly language (لغة التجميع).

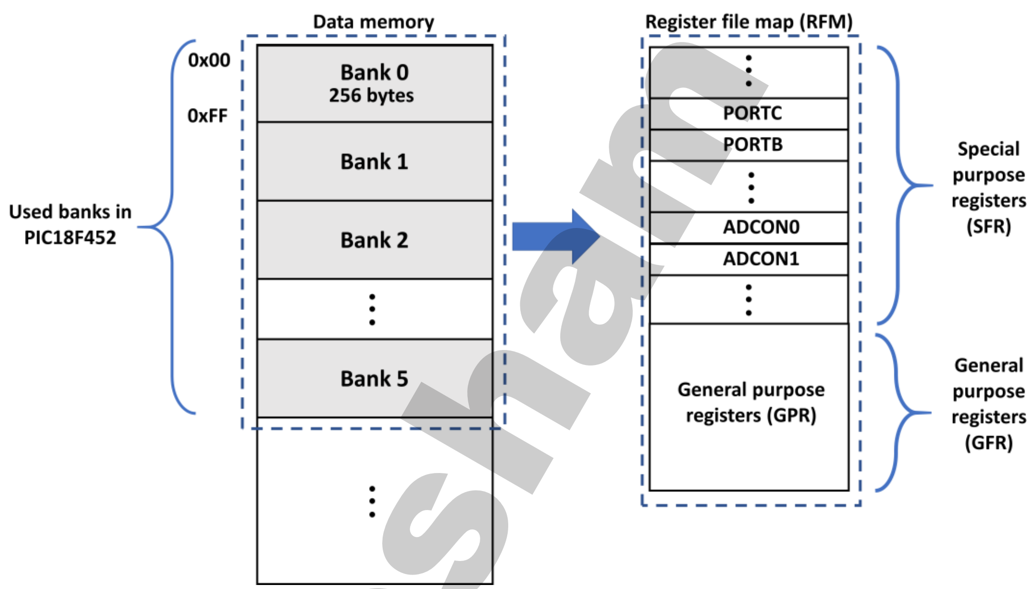


Fig. 1-6.

- The RFM is divided into two groups:

1. **General Purpose Registers (GPR)**

   Used to store the variables of the user program code.

2. **Special Purpose Registers (SFR)**

   Used to control the internal peripherals of the device.

   Ex. TRISA, TRISB, PORTA, ADCON0….

1.3.1 PIC18F452 Internal Structure

Unlike personal computers (PCs), the architecture of PIC microcontroller is based on something called “Harvard architecture”, meaning that the program and data memories are separated as shown in the figure. Each memory has its own bus (address and data).

The program memory size of PIC18F452 is 32KB and its type is flash EEPROM, which is a non-volatile flash memory that can be erase and reprogrammed by using a simple programmer. The data memory size of this PIC is 1536 Bytes, which is high compared to that of PIC16F877A (368 Bytes).

The Input/output pins in this microcontroller is divided into five ports (PORTA, PORTB, PORTC, PORTD and PORTE). The pins in each port are bi-directional meaning that they can be controlled by user to be either input our output. These ports are fully controlled by the CPU through registers called PORT and TRIS registers.

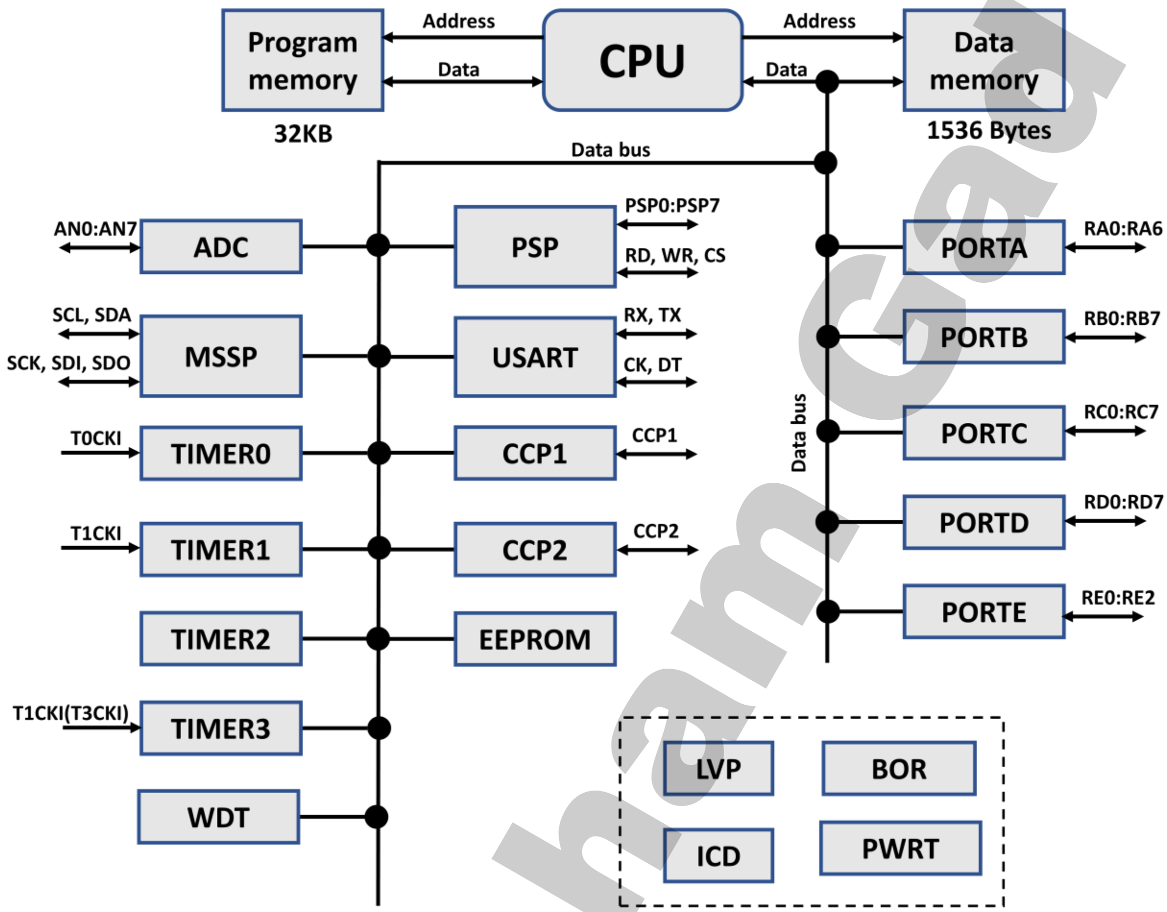


Fig. 1-7.

In addition to the input/output ports, PIC18F452 has some peripherals that can be controlled by the CPU as follows:

1. **Analog-to-Digital (ADC) module:**

   Which can measure up to eight analog sensors (AN0:AN7) with 10 bits resolution.

2. **Four timer modules (TIMER0 to TIMER3)**

   The timers are basically a counter but it can be used as a timer to generate precise timing signals defined by the user program code. The counters inside the timers can be also used for counting external pulses from pins (T0CKI and T1CKI), which is extremely useful in motor control applications.

3. **EEPROM module:**

The Electrically Erasable Programmable Read Only Memory (EEPROM) module is very useful to store important non-volatile **data** that must be kept in case of power failure such as, air conditioner temperature, security lock number…etc. The size of EEPROM memory in this PIC is 256 bytes which is the same as PIC16F877A.

4. **Parallel slave port (PSP) module:**

This module can be used to connect the microcontroller to other devices such as PCs or external memories or even to another microcontroller through the pins (PSP0:PSP7) and ($\overline{RD}, \overline{WR}, \overline{CS}$).

5. **Universal Synchronous/Asynchronous Receiver Transmitter (USART) module:**

This module is used to connect the microcontroller to other devices and components either in asynchronous (Full duplex via RX, TX) or in synchronous (Half duplex via CK, DT) modes. Full duplex means that sending and receiving can be done in the same time. In contrast, half duplex mode permits sending or receiving but not both at the same time in which we will discuss later in this course.

6. **Master Synchronous Serial Port (MSSP) module:**

This module supports the using of synchronous communication protocols such as I2C and SPI. These two protocols are very handy when dealing with sensors, shift registers and real time clock ICs.

7. **Capture-Compare-PWM (CCP2) and (CCP1) modules:**

The CCP modules are very useful in motor control applications, in which we can use them to control motor's speed by using pulse-width-modulation (PWM). Also, we can measure the motor speed by using capture and compare modules in which we will discuss them later in this book.

8. **Watchdog timer (WDT):**

   This timer is used to automatically reset the CPU in case of power or program failures.

In addition to the previous peripherals, there are five additional modules inside the chip, which are not controlled by the CPU (i.e. not connected to the CPU bus):

1. **Low volt programming (LVP) module:**

   Allows the PIC microcontroller to be programmed using operating voltage (V_{DD}) of the device.

2. **Brown ON Reset circuit:**

   Resets the microcontroller if the supply voltage on V_{DD} is below a certain value.

3. **Power-up Timer (PWRT):**

   Makes the PIC in reset state for a certain time when powering up the chip until stabilizing the power supply.

4. **In-Circuit Debugger (ICD):**

   Enable the user to control the CPU of the microcontroller in real time while running (e.g. Halting or running or step-by-step).

1.3.1 PIC18F452 Pin configuration

Pin	Function
V_{DD}, V_{ss}	Power supply pins (5v, Gnd) (Note: this PIC has two VDDs and two Vss) the user can feel free to use any of them).
OSC1, OSC2	External oscillator pins (for clock input)
$\overline{MCLR}$	Master clear pin (Reset input)
RA0:RA6	Bi-directional I/O PORTA pins
RB0:RB7	Bi-directional I/O PORTB pins
RC0:RC7	Bi-directional I/O PORTC pins
RD0:RD7	Bi-directional I/O PORTD pins
RE0:RE2	Bi-directional I/O PORTE pins
CCP1, CCP2	Capture-compare-PWM pins (Used for motor control applications)
AN0:AN7	Analog input pins
TX, RX	Full duplex Serial communication pins
CK, DT	Clock and data line for half duplex serial communication
T0CKI	Timer0 clock input pin
T1CKI	Timer1 clock input pin
INT0:INT2	External interrupt pin
PSP0:PSP7	Parallel slave port pins (Used to connect the microcontroller to other devices such as PC or another microcontroller)
$\overline{RD}$, $\overline{WR}$, $\overline{CS}$	Read, write and chip select pins (Used with PSP pins)
SCK	Serial clock pin (For SPI interface)

SDI	Serial data input pin (For SPI interface)
SDO	Serial data output pin (For SPI interface)
SCL	Serial clock line pin (For I2C interface)
SDA	Serial data line pin (For I2C interface)

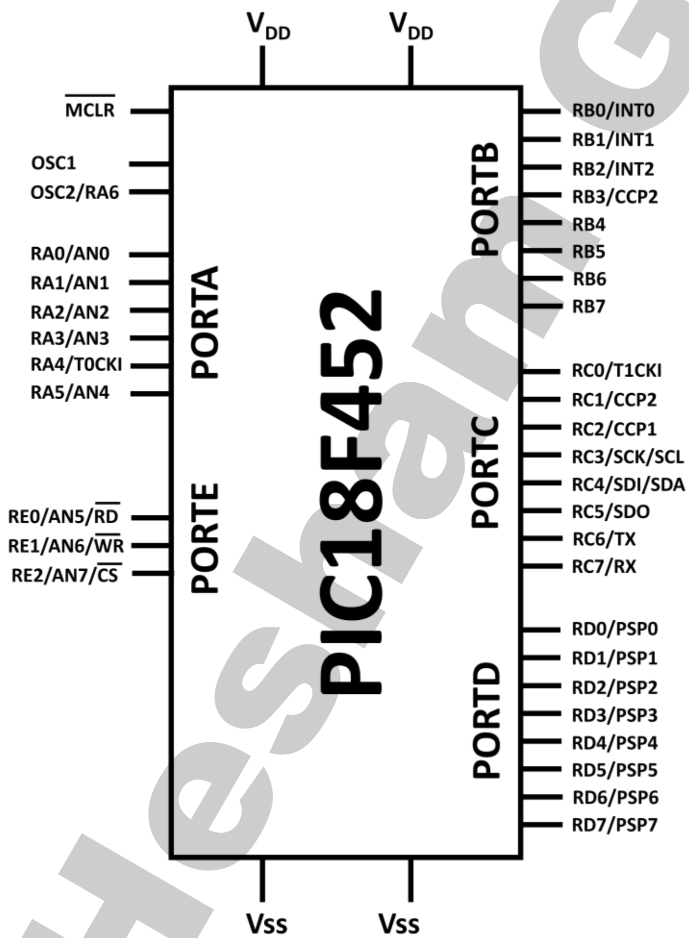


Fig. 1-8: Simplified layout of PIC18F452.

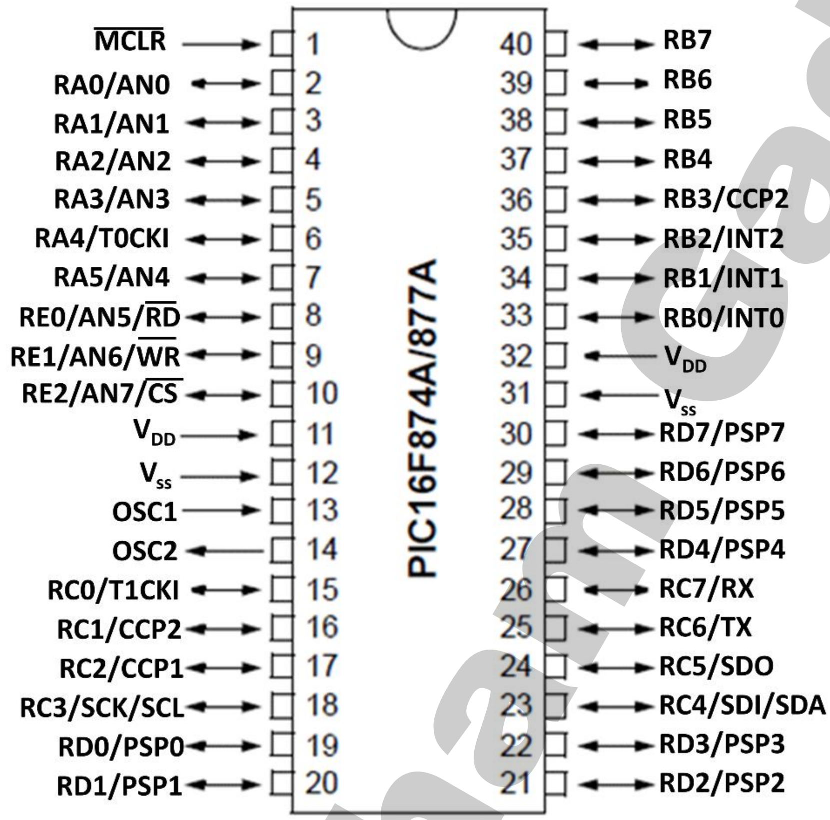


Fig. 1-9: Pin diagram of PIC18F452.

1.4 Minimum PIC18F452 configuration

Before burning the code into the microcontroller's ROM, we need three important components to make it alive. If one of them is missing, it will not work at all. The three components are:

1. The power supply circuit
2. Reset circuit.
3. Oscillator circuit.

This section will demonstrate each of these components in brief.

قبل تحميل البرنامج على ذاكرة الـ ROM الخاصة بالـ Microcontroller، فإننا نحتاج إلى مجموعة من المكونات التى تجعل الشريحة بها حياة. حيث أنه إذا لم يتواجد أياً من أحداهما فلن تعمل الشريحة مطلقاً. وهذه المكونات هى:

1. دائرة الـ Power supply.
2. دائرة إعادة تشغيل الـ Microcontroller أو الـ Reset circuit.
3. دائرة النبضات أو Oscillator circuit.

1.4.1 Power supply Circuit

Figure (1-10) shows a simple low-cost regulated power supply circuit. The power supply circuit can be supplied by 12V or 9V DC from either a battery or an AC source through a transformer and a full wave rectifier circuit. The main component of the circuit is an IC called (LM7805), which is used to reduce the input voltage from 12V or 9V to a regulated 5V DC with an output current up to 1 Amp. The capacitors are used to remove the ripples and the noise from the main power supply.

The output voltage (5V) from the power supply circuit can be used to supply any component such as microcontrollers, LCDs … etc. To operate the microcontroller, the 5V and GND of the power supply must be connected to V_{DD} and V_{ss} pins respectively as shown in Fig. 1-10.

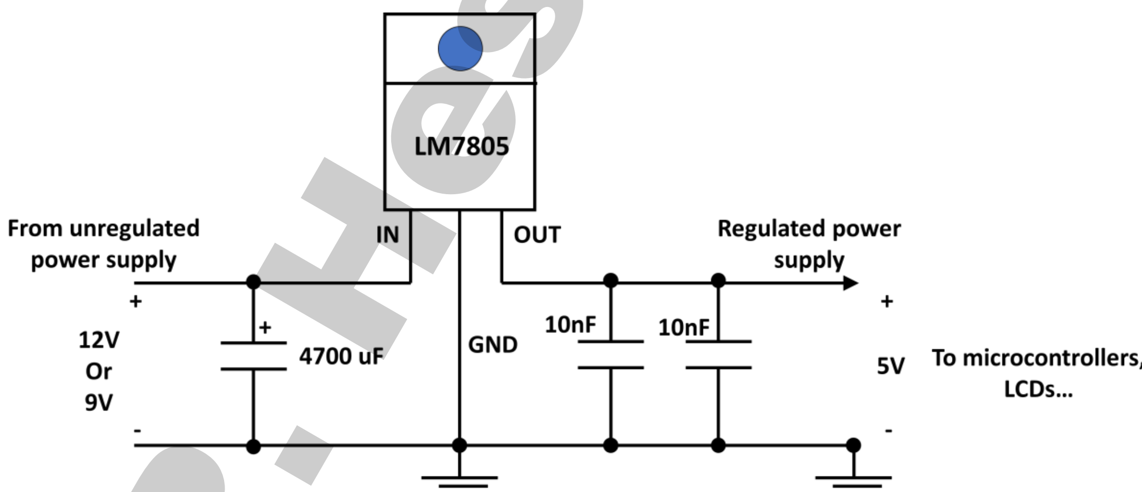


Fig. 1-10: Power supply circuit.

1.4.2- Reset circuit

The MCLR pin should be tied to the supply voltage directly (5V) or through a 4.7K resistor as shown in Fig. 1-11. If the voltage on the MCLR pin is dropped to 0v the microcontroller will reset immediately. Thus, we can use a RESET button to restart the microcontroller as shown in Fig. 1-12.

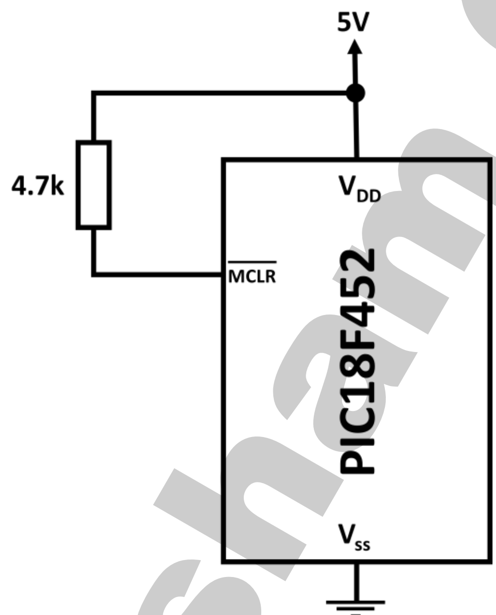


Fig. 1-11: Reset circuit.

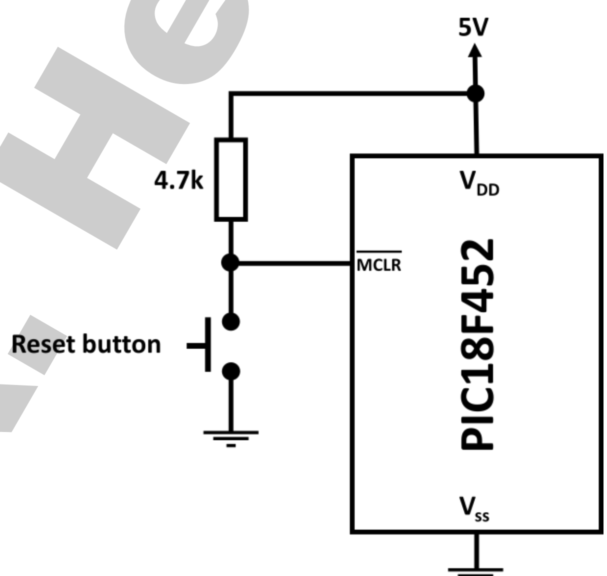


Fig. 1-12: Reset circuit with a restart button.

1.4.3- Oscillator circuit

Any microcontroller in general need a clock source to run. The PIC18F452 provides four modes of operation. However, we will discuss only three modes:

1. **Low-power crystal (LP) mode**

   For frequencies up to 200 kHZ.

2. **Crystal oscillator (XT) mode**

   For frequencies up to 4.0 MHZ.

3. **High-speed crystal oscillator (HS)**

   For frequencies up to 25.0 MHZ.

4. **High-speed crystal with PLL (HSPLL).**

   For frequencies up to 40.0 MHZ.

Figure 1-13 shows the crystal oscillator circuit, which is connected to the OSC1 and OSC2. The designer can use other circuit, but this circuit is preferred when timing accuracy is important. The capacitor values depend on the mode of the crystal and the selected frequency. Table 1-1 gives the recommended values. For example, for a 4MHz crystal frequency, we will use 15pF capacitors. Higher capacitance increases the oscillator stability but also increases the start-up time.

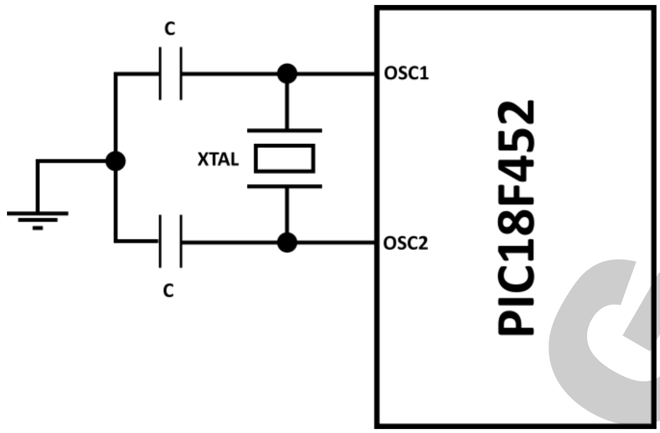


Fig. 1-13: Crystal oscillator circuit.

However, due to the electromagnetic interferences, the designer cannot use higher frequencies of crystal oscillators. A Phase Locked Loop (PLL) circuit is provided inside the PIC18F452 that can be enabled to multiply the clock frequency from the crystal oscillator circuit by 4. Thus, for a crystal clock frequency of 10MHz, the internal operation frequency will be multiplied to 40MHz. The PLL mode is enabled when the oscillator configuration bits are programmed for HSPLL mode.

Mode	Frequency	C1,C2 (pF)
LP	32 KHz	33
	200 KHz	15
XT	200 KHz	22-68
	1.0 MHz	15
	4.0 MHz	15
HS	4.0 MHz	15
	8.0 MHz	15-33
	20.0 MHz	15-33
	25.0 MHz	15-33