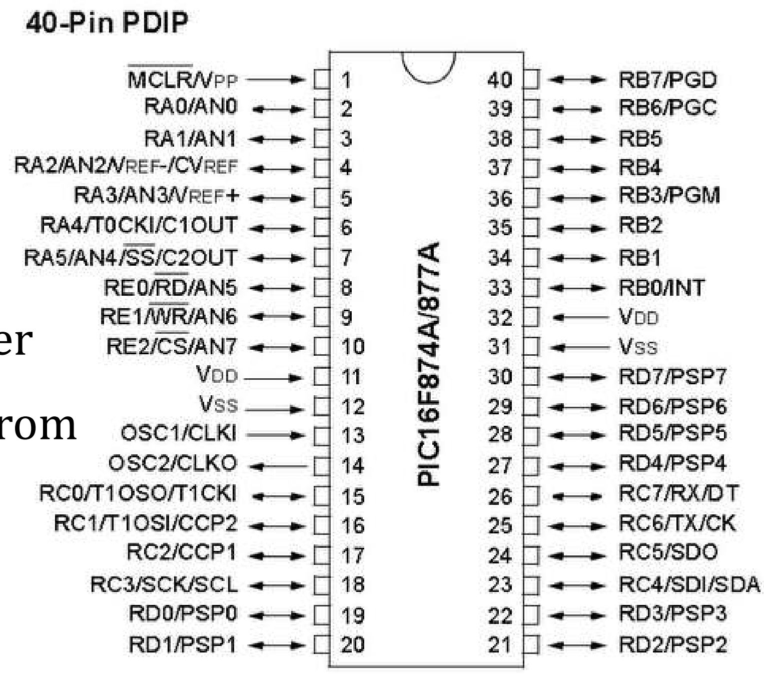
*Name :* **Mariam Elia Aziz**

# Embedded Systems Training

## Task 1

## [a] :




This the block diagram
Of PIC16F877A

♦ It is a **40-pin** PIC Microcontroller

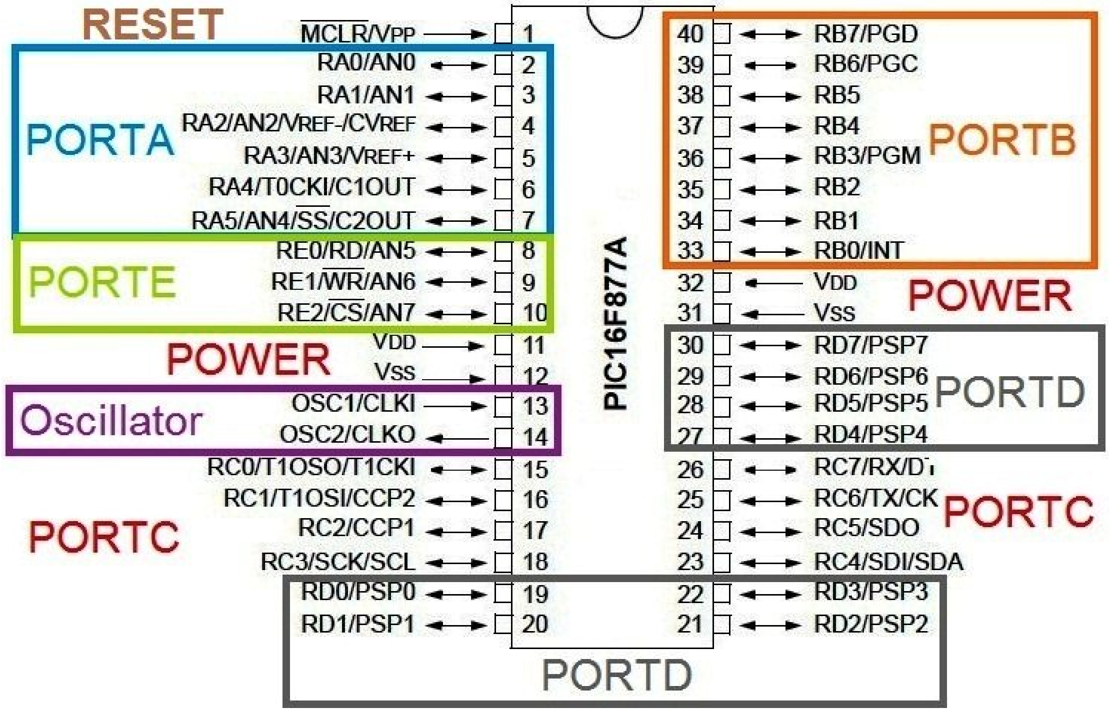
♦ It has **five Ports** on it, starting from
Port A , Port B , Port C ,
Port D and Port E

♦ It has three Timers in it,
two of which are 8-bit Timers
while 1 is of 16 Bit.

♦ It supports many communication protocols like:
1) Serial Protocol.
2) Parallel Protocol.
3) I2C Protocol.

♦ It supports both hardware pin interrupts and timer interrupts.

Here is the Segmentation of the ports of the PIC :




● Port A **(Pin2 to Pin7):**

Port A includes 6 Pins **<u>from Pin2 to Pin7</u>** like RA0 to RA5.
→ RA0 pin is the first Pin label of Port A.

- Pin2 (RA0/AN0): This is the first pin of the port-A. so this pin is used as an analog pin and it is built in ADC.
- Pin3 (RA1/AN1): This is the analog i/p1.
- Pin4 (RA2/AN2/Vref-): This pin acts as the analog i/p2 or it acts as an analog -ve reference voltage.
- Pin5 (RA3/AN3/Vref+): This pin acts as the analog i/p3 or it acts as the analog +ve reference voltage.
- Pin6 (RA0/T0CKI): This pin acts as the CLK input pin to timer0, so the output type is open drain.
- Pin7 (RA5/SS/AN4): This pin is the analog i/p4. This pin acts as the slave selects for the synchronous serial port within this controller.

## ● Port B **(Pin33 to Pin40):**

Port B has a total of 8 Pins **from Pin 33 to Pin 40** like RB0 – RB7.
From all these Port B pins, RB0 is utilized as the ***external interrupt*** pin
& also the change on the pins from RB4 to RB7 .

## ● Port C **(Pin15 to Pin18 & Pin23 to Pin26):**

Port C includes a total of 8 Pins that are not connected together.
The ***first 4 Pins are placed at Pin 15 to Pin 18*** whereas
the remaining pins are placed at Pin 23 to Pin 26.

## ● Port D **(Pin19 to Pin22 & Pin27 to Pin30):**

Port D includes 8 Pins which are also not connected together.
In this port, the ***first four pins are placed at Pin19 to Pin22*** whereas the remaining pins are placed at Pin27 – Pin30.
This port is a bidirectional I/O port & acts as the parallel slave port when the microprocessor bus needs to be interfaced.

## ● Port E **(Pin8 to Pin10):**

Port E includes 3 Pins **from Pin8 to Pin10** like RE0 – RE2
where the 'RE0' pin is the first pin label for Port E.

***Other 7 Pins:***

● **Pin 1 (MCLR):** This is the ***master clear pin***, so this pin resets the PIC microcontroller. Once this MCLR pin is active low, then it must be constantly given 5V & if 0V is given then the microcontroller will reset.

●**Pin 11 & 32 (VDD):** These are the ***positive supply pins*** for both the I/O & logic pins. So these two pins must be connected to 5Volts.

●**Pin 12 & 31 (VSS):** These two pins are the ***GND reference*** for both the I/O & logic pins. So these pins must be connected to zero potential.

●**Pin13 (OSC1/CLKIN):** This is the input pin of the oscillator or the external CLK i/p pin.

●**Pin14 (OSC2/CLKOUT):** This is the o/p pin of the oscillator.
A *crystal resonator* is simply connected between two pins like pins 13 & 14 for ***providing external CLK*** to the microcontroller.

---

## [b] :

## PIC16F877 Internal Block Diagram

• The basic architecture of PIC16F877 consists of :
Program memory, Instruction registers , status and control , ALU , program counter , Flash program memory , Instruction decoder .

### →Memory of the PIC16F877A

• divided into 3 types of memories:

1. Program Memory
2. Data Memory
3. Data EEPROM

The program memory and data memory have separate buses so that concurrent access can occur .

## Program Memory Organization

The PIC16F87XA devices have a 13-bit program counter capable of addressing an 8K word x 14 bit program memory space.
The PIC16F877A devices have ***8K words x 14 bits*** of Flash program memory

## Instruction registers

A register is a place inside the PIC which used to read or write the data/program. The memory of the PIC is divided into a series of registers. Each of the registers has its own address and memory locations.
the registers in PIC are classified into **two categories** :

- **General Purpose Registers (GPR) :**

is a small amount of storage that can be accessible more quickly than any other memory. These register files can be accessed either directly, or indirectly, through the File Select Register (FSR).

- **Special Function Registers (SFR)**

The special function registers are also memory registers which is used for special dedicated functions.
They are registers ***used by the CPU and peripheral modules*** for controlling the desired operation of the device , and they are implemented as *static RAM.*

# Instruction decoder

The Instruction Decoder block is responsible for decoding the instruction that is fetched from the program memory in the PIC16F877A microcontroller.
It is a part of the CPU and plays a critical role in the execution of instructions.
It takes the instruction code from the Instruction Register and **decodes it into the corresponding control signals** that will control the operation of various blocks within the microcontroller.

The output of the Instruction Decoder block *is a set of control signals* that are used to execute the instruction. These control signals are used to select the appropriate register or memory location, enable the ALU, and perform other operations required by the instruction.

# ALU

The Arithmetic Logic Unit (ALU) is a key component of the PIC16F877A microcontroller's central processing unit (CPU) and is responsible for performing arithmetic and logical operations on binary data.

The ALU block takes input data from registers or memory locations, performs the specified operation, and produces the result in the destination register or memory location.
The result can then be used by the CPU for further processing.

## Flash program memory

The Flash Program Memory block in PIC16F877A is a ***non-volatile memory block*** that stores the program code that the microcontroller executes during operation.
It is a key component of the microcontroller and is responsible for holding the firmware or software that controls the behavior of the system.

It can be programmed by the user using a programmer device and can be reprogrammed multiple times →This allows for the code to be updated or modified as per the requirements of the application.

## program counter

Program Counter (PC) is ***13 bit*** and capable of addressing an ***8K word x 14 bit program memory space.***

→ It keeps track of the program execution by holding the address of the current instruction.

→ It is automatically incremented to the next instruction during the current instruction execution.

→ It also plays a crucial role in the execution of branch instructions, which allow the program to jump to a different memory location in the program memory.

## Status and Control

→ It is a register that contains the status and control flags including the Carry Flag, Zero Flag, and Overflow Flag and these are used by the CPU during the execution of instructions.

→ It is a critical component of the microcontroller's CPU and enables the CPU to communicate with the peripherals and control the execution of instructions.

→ The Status and Control register also contains the Global Interrupt Enable bit, which enables or disables all interrupts in the microcontroller.

## [C] :

**Pin RA4** is multiplexed with the Timer0 module clock
input to become the RA4/T0CKI pin.
The RA4/T0CKI pin is a Schmitt Trigger input and an open-drain output.

All other PORTA pins have TTL input levels and
full CMOS output drivers.
***<u>Other PORTA pins</u>*** are multiplexed with **analog inputs**
and the analog VREF input for both the A/D converters
and the comparators.
*** The operation of each pin is
selected by clearing/setting the appropriate control bits
in the **ADCON1 and/or CMCON registers**

On a Power-on Reset, these pins are configured as analog inputs and ***read as '0'.***

In the internal structure of the other pins of Port A there is two transistors ( PMOS , NMOS )
PMOS transistor gives output 5 v when the input is 1
and gives output 0 v when the input is 0

while NMOS transistor gives output 0 v when the input is 1
and gives output 5 v when the input is 0

in pin RA4 there is only NMOS transistor only

so when we want to apply flasher on this pin we **have to apply it by sink mode not the source one.**

## [D] :

**ATMega328p** and **PIC16F877A** are both popular microcontrollers used in a wide range of applications. Here is a comparison of their characteristics:

### Architecture:

The ATMega328p→ is based on the AVR architecture, which is known for its efficient use of power

while the PIC16F877A →is based on the PIC architecture , which is known for its versatility.

### Clock Speed:

The ATMega328p →has a clock speed of up to 20 MHz,

while the PIC16F877A →has a clock speed of up to 20 MHz.

### Memory:

The ATMega328p →has *32 KB of flash memory* for program storage, *2 KB of SRAM* for data storage, and *1 KB of EEPROM* for non-volatile data storage.

The PIC16F877A→ has *14 KB of flash memory*, *368 bytes of RAM*, and *256 bytes of EEPROM.*

### Power Consumption:

The ATMega328p →is known for its low power consumption, with a typical current consumption of 1.5 mA at 1 MHz and 1.8V.

The PIC16F877A→ has a typical current consumption of 15 mA at 4 MHz and 5V.

**Cost:** The ATMega328p is generally less expensive than the PIC16F877A, making it a popular choice for hobbyists and low-cost applications.

**Pin Count :**

The ATMega328p and PIC16F877A have a significant difference in the number of pins.
The ATMega328p →has **28 pins,**
while the PIC16F877A →has **40 pins**.

---

The choice between ATMega328P and PIC16F877A depends on the specific requirements of the embedded system.

Here are two examples of applications where **ATMega328P is a better choice than PIC16F877A:**

**Wearable Devices:** ATMega328P is a better choice than PIC16F877A for wearable devices because they often have strict power requirements and need to be small and lightweight.

and ATMega328P is known for its low power consumption and has a smaller form factor than the PIC16F877A, making it ideal for wearable devices.

**Home Automation:** ATMega328P is a better choice than PIC16F877A for home automation systems because they often require integration with various sensors and actuators, which can be programmed using the Arduino development environment.

and ATMega328P is compatible with the popular Arduino development environment, which has a large community and many libraries for interfacing with sensors and actuators. This makes it easier to develop home automation systems using ATMega328P compared to PIC16F877A.