PIC Microcontrollers

PIC stands for Peripheral Interface Controller given by Microchip Technology to identify its single-chip microcontrollers. These devices have been very successful in 8-bit microcontrollers. The main reason is that Microchip Technology has continuously upgraded the device architecture and added needed peripherals to the microcontroller to suit customers' requirements. The architectures of various PIC microcontrollers can be divided as follows.

Low - end PIC Architectures :

Microchip PIC microcontrollers are available in various types. When PIC microcontroller MCU was first available from General Instruments in early 1980's, the microcontroller consisted of a simple processor executing 12-bit wide instructions with basic I/O functions. These devices are known as low-end architectures. They have limited program memory and are meant for applications requiring simple interface functions and small program & data memories. Some of the low-end device numbers are

12C5XX , 16C5X , 16C505

Mid-range PIC Architectures

Mid-range PIC architectures are built by upgrading low-end architectures with more number of peripherals, more number of registers and more data/program memory. Some of the mid-range devices are 16C6X , 16C7X , 16F87X

Program memory type is indicated by an alphabet. C = EPROM , F = Flash , RC = Mask ROM

Popularity of the PIC microcontrollers is due to the following factors.

1. Speed: Harvard Architecture, RISC architecture, 1 instruction cycle = 4 clock cycles.
2. Instruction set simplicity: The instruction set consists of just 35 instructions (as opposed to 111 instructions for 8051).
3. Power-on-reset and brown-out reset. Brown-out-reset means when the power supply goes below a specified voltage (say 4V), it causes PIC to reset; hence malfunction is avoided.
   A watch dog timer (user programmable) resets the processor if the software/program ever malfunctions and deviates from its normal operation.
4. PIC microcontroller has four optional clock sources.
   - Low power crystal
   - Mid range crystal
   - High range crystal
   - RC oscillator (low cost).
5. Programmable timers and on-chip ADC.
6. Up to 12 independent interrupt sources.
7. Powerful output pin control (25 mA (max.) current sourcing capability per pin.)
8. EPROM/OTP/ROM/Flash memory option.
9. I/O port expansion capability.

CPU Architecture: The CPU uses Harvard architecture with separate Program and Variable (data) memory interface. This facilitates instruction fetch and the operation on data/accessing of variables simultaneously.

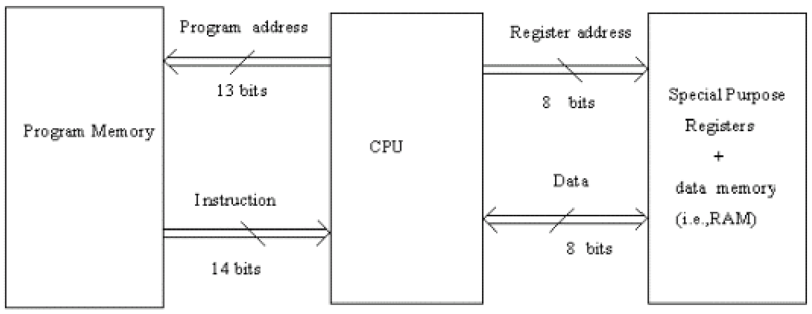


Figure 1 CPU Architecture of PIC microcontroller

PIC Memory Organisation:

PIC microcontroller has 13 bits of program memory address. Hence it can address up to 8k of program memory. The program counter is 13-bit. PIC 16C6X or 16C7X program memory is 2k or 4k. While addressing 2k of program memory, only 11- bits are required. Hence two most significant bits of the program counter are ignored. Similarly, while addressing 4k of memory, 12 bits are required. Hence the MSb of the program counter is ignored.

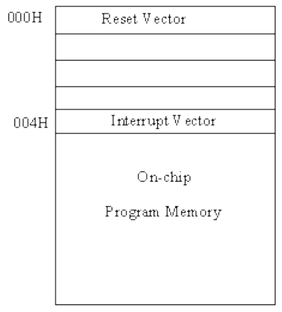


Figure 2 Program Memory map

The program memory map of PIC16C74A is shown in Fig.2.

On reset, the program counter is cleared and the program starts at 00H. Here a 'goto' instruction is required that takes the processor to the mainline program.

When a peripheral interrupt, that is enabled, is received, the processor goes to 004H. A suitable branching to the interrupt service routine (ISR) is written at 004H.

Data memory (Register Files):

Data Memory is also known as Register File. Register File consists of two components.

1. General purpose register file (same as RAM).
2. Special purpose register file (similar to SFR in 8051).

The special purpose register file consists of input/output ports and control registers. Addressing from 00H to FFH requires 8 bits of address. However, the instructions that use direct addressing modes in PIC to address these register files use 7 bits of instruction only. Therefore the register bank select (RP0) bit in the STATUS register is used to select one of the register banks.

In indirect addressing FSR register is used as a pointer to anywhere from 00H to FFH in the data memory.

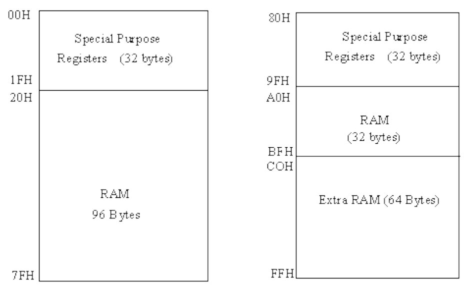


Figure 3 Data Memory map

Basic Architecture of PIC Microcontrollers

Specifications of some popular PIC microcontrollers are as follows:

Device	Program Memory (14bits)	Data RAM (bytes)	I/O Pins	ADC	Timers 8/16 bits	CCP (PWM)	USART SPI / I2C
16C74A	4K EPROM	192	33	8 bits x 8 channels	2/1	2	USART SPI / I^2C
16F877	8K Flash	368 (RAM) 256 (EEPROM)	33	10 bits x 8 channels	2/1	2	USART SPI / I^2C

Device	Interrupt Sources	Instruction Set
16C74A	12	35
16F877	15	35

PIC Microcontroller Clock

Most of the PIC microcontrollers can operate upto 20MHz. One instructions cycle (machine cycle) consists of four clock cycles.

Instructions that do not require modification of program counter content get executed in one instruction cycle.

Although the architectures of various midrange 8 - bit PIC microcontroller are not the same, the variation is mostly interns of addition of memory and peripherals. We will discuss here the architecture of a standard mid-range PIC microcontroller, 16C74A. Unless mentioned otherwise, the information given here is for a PIC 16C74A microcontroller Chip.

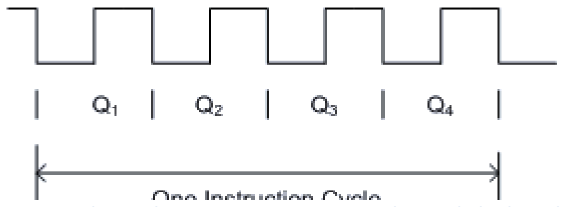


Figure 4 Relation between instruction cycles and clock cycles for PIC microcontrollers

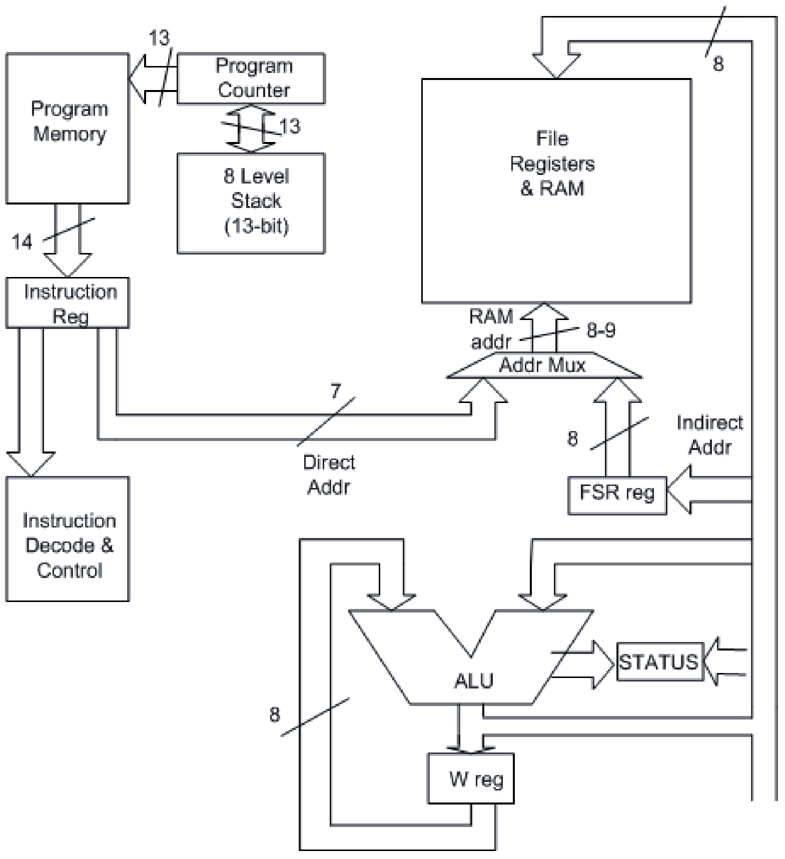


Figure 5 Basic Architecture of PIC 16C74A

Architecture of PIC16C74A

The basic architecture of PIC16C74A is shown in fig 17.2. The architecture consists of Program memory, file registers and RAM, ALU and CPU registers. It should be noted that the program Counter is 13 - bit and the program memory is organised as 14 - bit word. Hence the program Memory capacity is 8k x 14 bit. Each instruction of PIC 16C74A is 14 - bit long. The various CPU registers are discussed here.

CPU registers (registers commonly used by the CPU)

W, the working register, is used by many instructions as the source of an operand. This is similar to accumulator in 8051. It may also serve as the destination for the result of the instruction execution. It is an 8 - bit register.

7
0
W, Working register

Figure 6 W register

STATUS Register

The STATUS register is a 8-bit register that stores the status of the processor. This also stores carry, zero and digit carry bits.

STATUS - address 03H, 83H

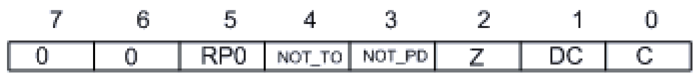


Figure 7 STATUS register

C = Carry bit

DC = Digit carry (same as auxiliary carry)

Z = Zero bit

NOT_TO and NOT_PD - Used in conjunction with PIC's sleep mode

RP0- Register bank select bit used in conjunction with direct addressing mode.

FSR Register (File Selection Register, address = 04H, 84H)

FSR is an 8-bit register used as data memory address pointer. This is used in indirect addressing mode.

INDF Register (INDirect through FSR, address = 00H, 80H)

INDF is not a physical register. Accessing INDF access is the location pointed to by FSR in indirect addressing mode.

PCL Register (Program Counter Low Byte, address = 02H, 82H)

PCL is actually the lower 8-bits of the 13-bit program counter. This is a both readable and writable register.

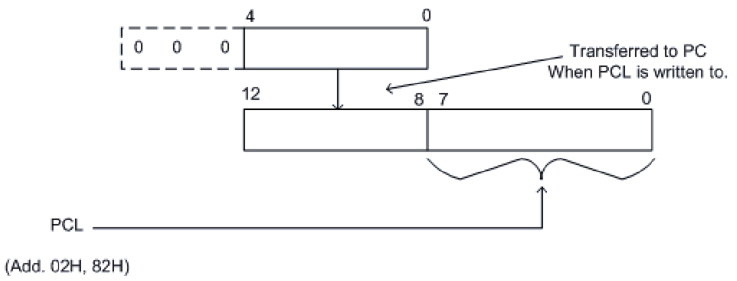
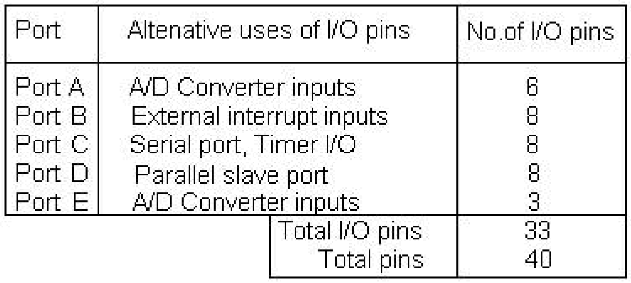


Figure 8 Schematic of how PCL is loaded from PCLATH

PCLATH Register (Program Counter Latch, address = 0AH, 8AH)

PCLATH is a 8-bit register which can be used to decide the upper 5bits of the program counter. PCLATH is not the upper 5bits of the program counter. PCLATH can be read from or written to without affecting the program counter. The upper 3bits of PCLATH remain zero and they serve no purpose. When PCL is written to, the lower 5bits of PCLATH are automatically loaded to the upper 5bits of the program counter, as shown in the figure.

Program Counter Stack
An independent 8-level stack is used for the program counter. As the program counter is 13bit, the stack is organized as 8x13bit registers. When an interrupt occurs, the program counter is pushed onto the stack. When the interrupt is being serviced, other interrupts remain disabled. Hence, other 7 registers of the stack can be used for subroutine calls within an interrupt service routine or within the mainline program.
Register File Map
It can be noted that some of the special purpose registers are available both in Bank-0 and Bank-1. These registers have the same value in both banks. Changing the register content in one bank automatically changes its content in the other bank.
Port Structure and Pin Configuration of PIC 16C74A
As mentioned earlier, there is a large variety of PIC microcontrollers. However, the midrange architectures are widely used. Our discussion will mainly confine to PIC16C74A whose architecture has most of the required features of a mid-range PIC microcontroller. Study of any other mid-range PIC microcontroller will not cause much variation from the basic architecture of PIC 16C74A ..



Port	Altenative uses of I/O pins	No.of I/O pins
Port A	A/D Converter inputs	6
Port B	External interrupt inputs	8
Port C	Serial port, Timer I/O	8
Port D	Parallel slave port	8
Port E	A/D Converter inputs	3
	Total I/O pins	33
	Total pins	40

PIC 16C74A has 5 I/O Ports. Each port is a bidirectional I/O port. In addition, they have the following alternate functions.
In addition to I/O pins, there is a Master clear pin (MCLR) which is equivalent to reset in 8051. However, unlike 8051, MCLR should be pulled low to reset the micro controller. Since PIC16C74Ahas inherent power-on reset, no special connection is required with MCLR pin to reset the micro controller on power-on.
There are two V_{DD} pins and two V_{SS} pins. There are two pins (OSC1 and OSC2) for connecting the crystal oscillator/ RC oscillator. Hence the total number of pins with a 16C74A is 33+7=40. This IC is commonly available in a dual-in-pin (DIP) package.

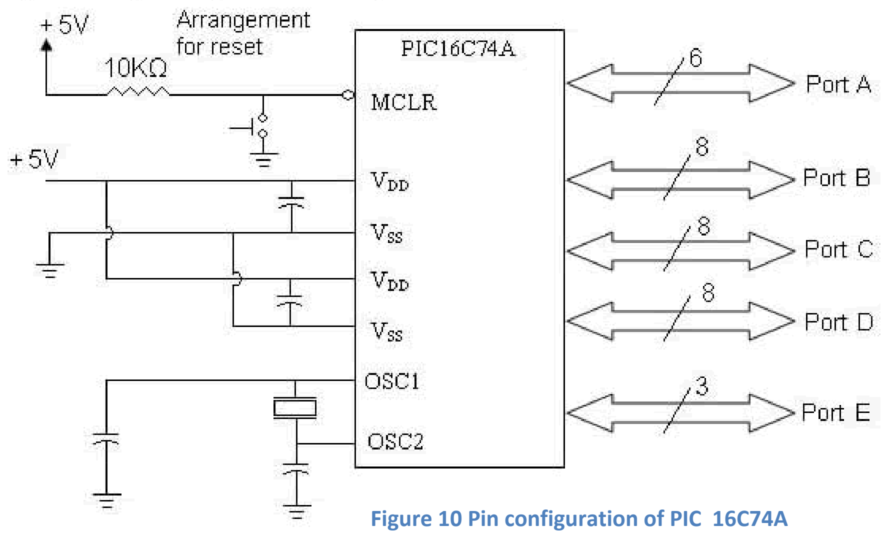


Figure 10 Pin configuration of PIC 16C74A

The PIC architecture is characterized by the following features:

- Separate code and data spaces (Harvard architecture) for devices other than PIC32, which has a Von Neumann architecture.
- A small number of fixed length instructions
- Most instructions are single cycle execution (2 clock cycles), with one delay cycle on branches and skips
- One accumulator (W0), the use of which (as source operand) is implied (i.e. is not encoded in the opcode)
- All RAM locations function as registers as both source and/or destination of math and other functions.[2]
- A hardware stack for storing return addresses
- A fairly small amount of addressable data space (typically 256 bytes), extended through banking
- Data space mapped CPU, port, and peripheral registers
- The program counter is also mapped into the data space and writable (this is used to implement indirect jumps).

Limits
The PIC architectures have several limits:

- Only one accumulator
- A small instruction set
- Operations and registers are not orthogonal; some instructions can address RAM and/or immediate constants, while others can only use the accumulator
- Memory must be directly referenced in arithmetic and logic operations, although indirect addressing is available via 2 additional registers
- Register-bank switching is required to access the entire RAM of many devices

The following limitations have been addressed in the PIC18, but still apply to earlier cores:

- Conditional skip instructions are used instead of conditional jump instructions used by most other architectures
- Indexed addressing mode is very rudimentary

- Stack:
  - The hardware call stack is so small that program structure must often be flattened
  - The hardware call stack is not addressable, so pre-emptive task switching cannot be implemented
  - Software-implemented stacks are not efficient, so it is difficult to generate reentrant code and support local variables
- Program memory is not directly addressable, and thus space-inefficient and/or time-consuming to access. (This is true of most Harvard architecture microcontrollers.)