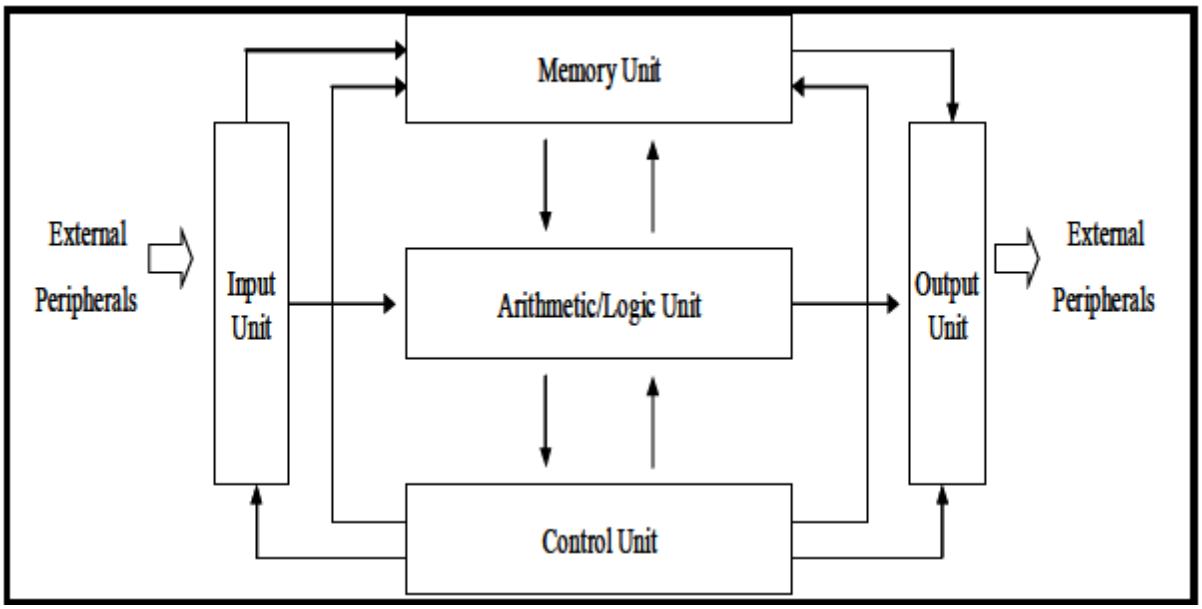
## Chapter 1

1 microprocessor .microcomputers, and microcontrollers (basic introduction and comparison)
2 Types of buses, address bus ,data bus and control bus
3 Harvard and Von-neumman architecture
4 8051 microcontroller: Architecture, Pin configuration, stack ,memory
5 Boolean processor, Power saving options
6 Derivatives of 8051 (8951 , 8952 8031 ,8751)

**1.**Microprocessor .microcomputers, and microcontrollers (basic introduction and comparison)

**A Block Diagram of Microcomputer.**

c




**1.Arithmetic Logic Unit (ALU)**
The arithmetic and logical unit (ALU) performs arithmetic operations such as addition, subtraction, multiplication, and or division, and logical operation such as AND, OR, NOT and XOR needed to carry out the instructions.
**2. Control Unit (CU)**
The control unit (CU) is responsible for fetching instructions from main memory and determining their type.
**3. Memory Unit (MU)**
The memory unit (MU) is used to store information such as number or character data.
The memory has the ability to hold this information for processing or for outputting at a later time. The memory unit is divided into primary storage memory and secondary storage memory. Typically, Primary storage memory is implemented with semiconductor memories: read-only memory (ROM) and random access read/write memory (RAM) . Secondary storage memory is

used for long-term storage of information that is not currently being used such as disk and CD ROM.

**Semiconductor Memories**

**Rom (Read-Only memory)**

By using ROM, the information is made nonvolatile; that is, the information is not lost if power is turned off. ROMs can be divided into:

1. Mask ROM

Mask ROMs cannot be changed or erased, internationally or otherwise. The data in a mask ROM are inserted during its manufacture.. The only way to change the program in a mask ROM is to replace the entire chip.

2. PROM

The PROM (Programmable) is like a mask ROM, except that it can be programmed once in the field by special device called PROM programmer.

3. EPROM

The EPROM (Erasable PROM) cannot only be field – programmed but also field erased. When the quartz window in an EPROM is expressed to storage ultraviolet light for 15-20 minutes, all the bits are sets to 1.

4. EEPROM

The EEPROM (Electrically Erasable PROM) or E2PROM can be erased by applying pulses to it instead of requiring it to be put in a special chamber for exposure to ultraviolet light . The new type of ROM called Flash memory is similar to EEPROM in configuration. Flash memory can be programmed on a circuit board by the use of ISP (In-System Programming).

**RAM (Random access Memory)**

By using RAM, the information is made volatile; that is, the information is lost if power is turned off. RAMs come in tow varieties: static and dynamic.

1. SRAM

SRAMs are constructed internally using circuits similar to the basic D latch. These memories have the property that their contents are retained as long as the power is kept on.

2. DRAM (Dynamic RAM)

DRAMs, in contrast, do not use latch – like circuits. Instead, a dynamic RAM is an array of tiny capacitors, each of which can be charged or discharged, allowing 0 and 1 to be stored. Because the electric charge tends to leak out, each bit in a dynamic RAM must be refreshed every few milliseconds to prevent the data from leaking away.

**4. Input unit (IU)**

The input unit (IU) is used to input the information to be processed from external input device such as a card reader, keyboard, or switch.

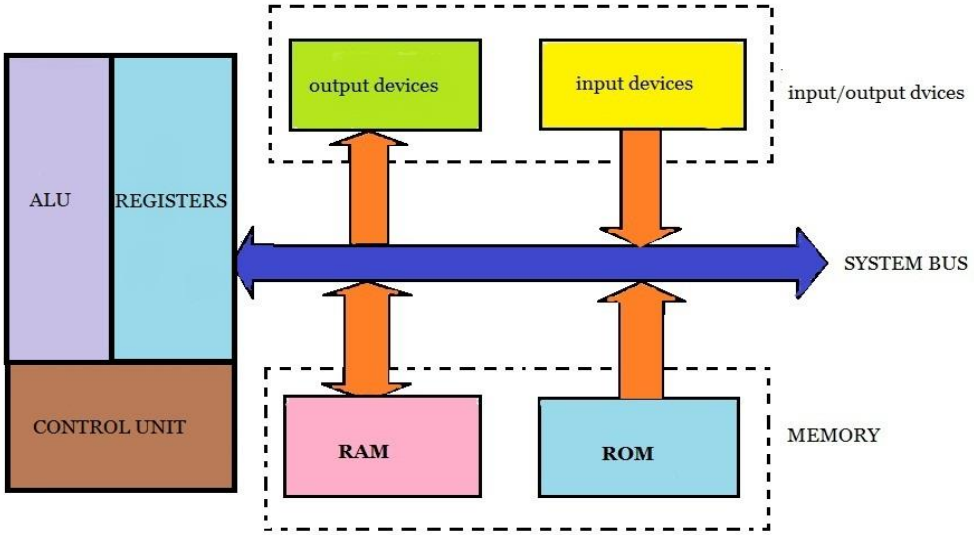
**5. Output Unit (OU)**

The output unit (OU) is used to output the processed results of computer to the external output devices such as a printer, monitor, 7- segment display, and LED.

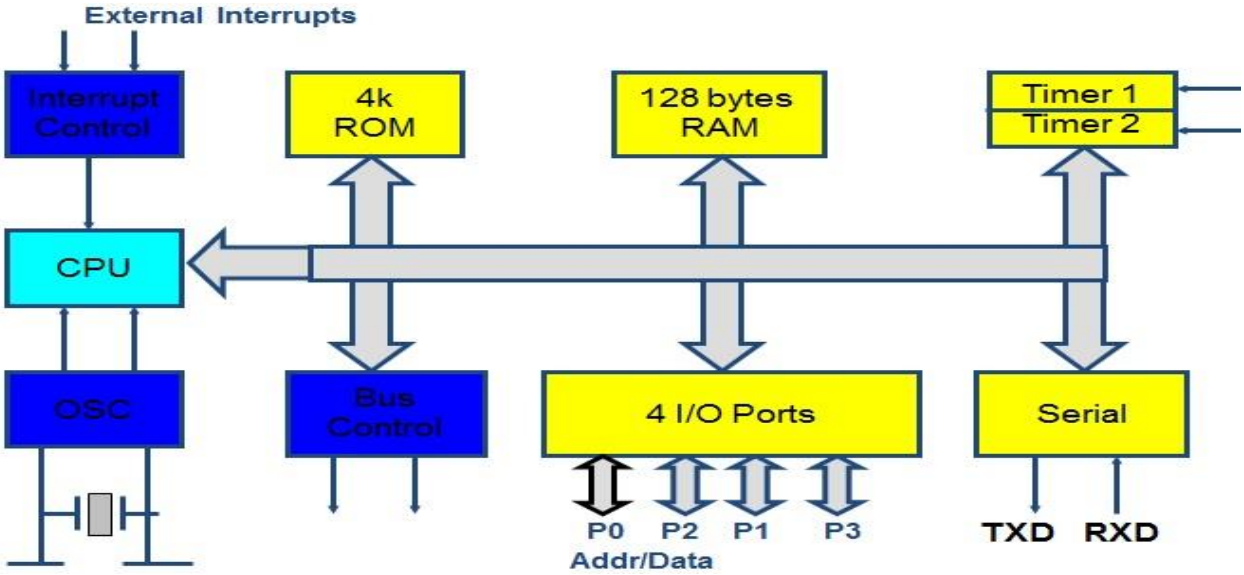
.

The central processing unit (CPU) is formed by combining the ALU and CU together. The CPU is the brain of the microcomputer.

## B. Microprocessor




## C, General block diagram of Microcontroller

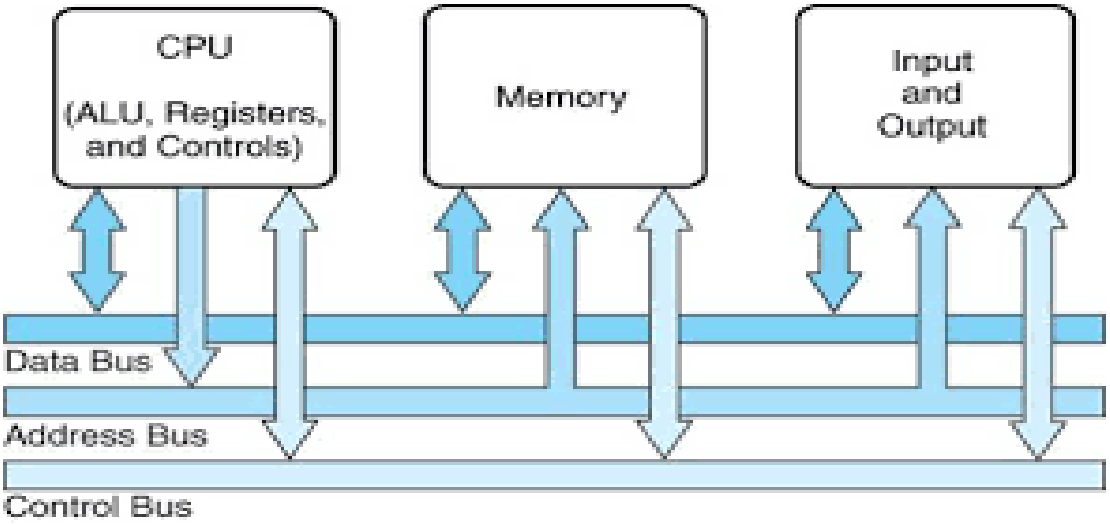


**Comparison of microprocessor ( 8085 ) and microcontroller (8051)**

| **Microcontroller** | **Microprocessor** |
|---|---|
| 1.Microcontroller having inbuilt RAM or ROM | 1. Do not have inbuilt RAM or ROM<br>. |
| 2.Have inbuilt timer | 2. Do not have timer |
| 3. Input and output ports are available. | 3. Input and output ports are not available, requires extra device like 8155 |
| 4. Inbuilt serial port. | **4.** . Do not have inbuilt serial port, requires 8250 device |
| 5. Separate memory to store program and data. | 5. Program and data are stored in same memory |
| 6. Many functions pins on the IC. | 6. Less multifunction pins on IC. |
| 7. Boolean operation directly possible. | 7. Boolean operation is not possible directly. |
| 8. It takes few instructions to read and write data from external memory. | 8. It take many instruction to read and write data from external memory |
| 9. Microcontrollers are designed to perform specific tasks. and the cost | 9.. Microprocessor find applications where tasks are unspecific like developing software, games, websites, photo editing, creating documents |
| 10. example, keyboards, mouse, washing machine, digicam, pendrive, remote, microwave, cars, bikes, telephone, mobiles, watches, etc. | 10. Application of microprocessor includes Desktop PC's, Laptops, notepads |
| 11. System design using microprocessor costly | 11. System design using microcontroller cheaper |

## 2. System Bus

A bus is a collection of wires used to transmit signals in parallel. According to the purpose, the buses of a microcomputer can be divided into three types: address bus, data bus, and control bus.




1. Address Bus
A external device is identified by its unique address. The address bus holds the address external device being accessed. The address bus is made up of a set of wires or lines; each line takes one bit of the address. The address bus is uni-directional . The width of the address bus is measured by the number of lines on the bus. A 3-bit address bus has 3 lines and can identify 2^3 devices.

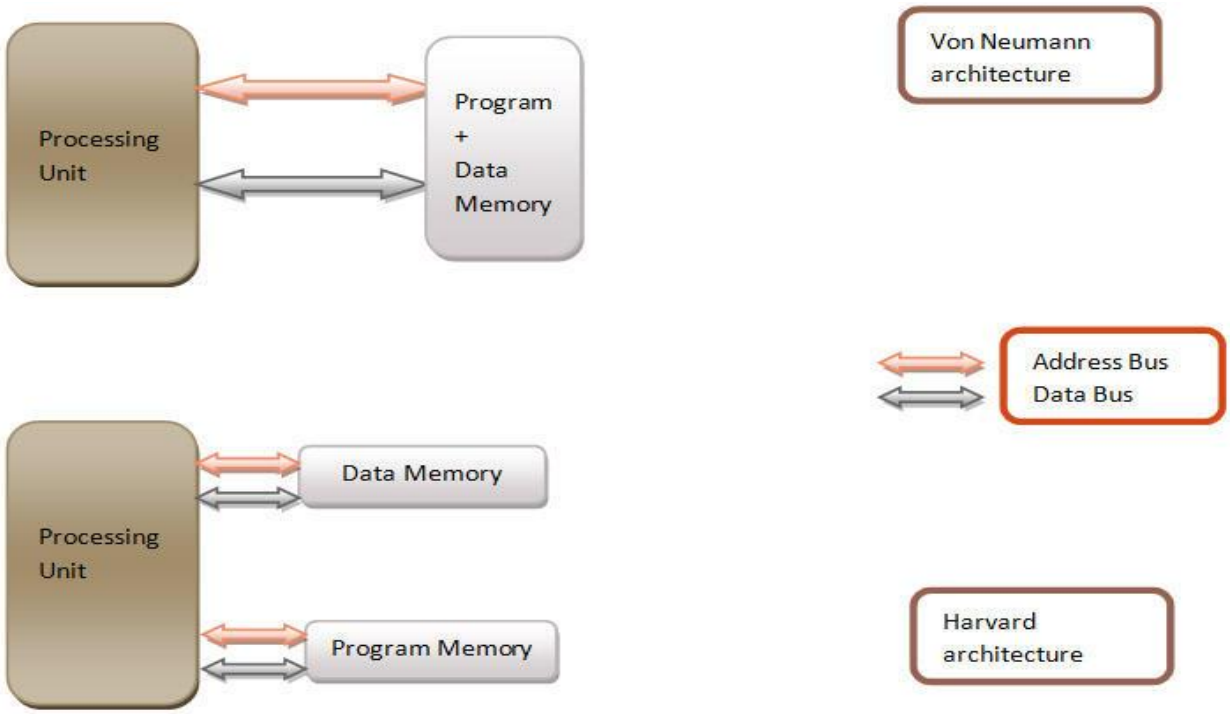
2. Data Bus
The data bus allows data to be transferred between the external devices and the processor. The data bus is bi-directional as data must go both ways. The data bus is made up of a set of wires or lines, each line takes one bit of the data. The width of the data bus usually matches the size of a memory location, processor registers are also usually the same size. The width of the data bus is measured by the number of lines on the bus. A 32-bit data bus has 32 lines and can transfer 32-bit of data in one go. The width of the data bus is a factor in system performance. The wider the data bus the more data that can be transferred in one go.
3. Control Bus
The lines on the control bus allow the processor to send or receive (transmit) control signals. The control bus is used to transmit the control signals such as read, write, and interrupt control signal. The control bus is *not really a bus* but a group of separate lines that each performs **individual** tasks.

**3. Harvard and Von-neumman architecture**

**Von Nueman architecture**

- It is named after the mathematician and early computer scientist John Von Neumann.
- The computer has single storage system(memory) for storing data as well as program to be executed.
- Processor needs two clock cycles to complete an instruction.Pipelining the instructions is not possible with this architecture.

- In the first clock cycle the processor gets the instruction from memory and decodes it. In the next clock cycle the required data is taken from memory. For each instruction this cycle repeats and hence needs two cycles to complete an instruction.
- This is a relatively older architecture and was replaced by Harvard architecture.




**Harvard architecture**

The name is originated from "Harvard Mark I" a relay based old computer.

The computer has two separate memories for storing data and program.

Processor can complete an instruction in one cycle if appropriate pipelining strategies are implemented.

In the first stage of pipeline the instruction to be executed can be taken from program memory. In the second stage of pipeline data is taken from the data memory using the decoded instruction or address.

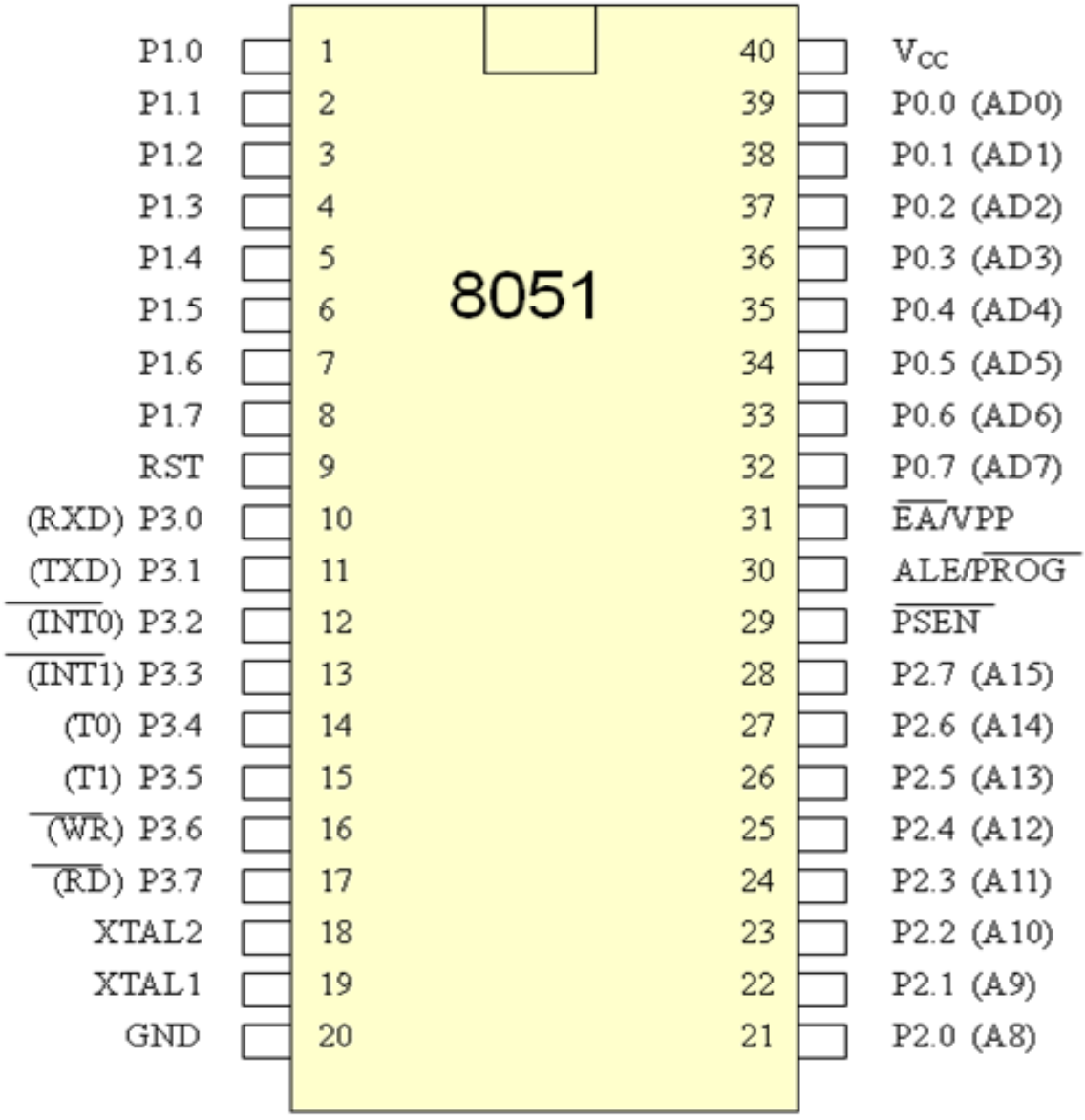
Most of the modern computing architectures are based on Harvard architecture. But the number of stages in the pipeline varies from system to system.

## 4. 8051 microcontroller: Architecture, Pin configuration, stack ,memory

### A. 8051 Features

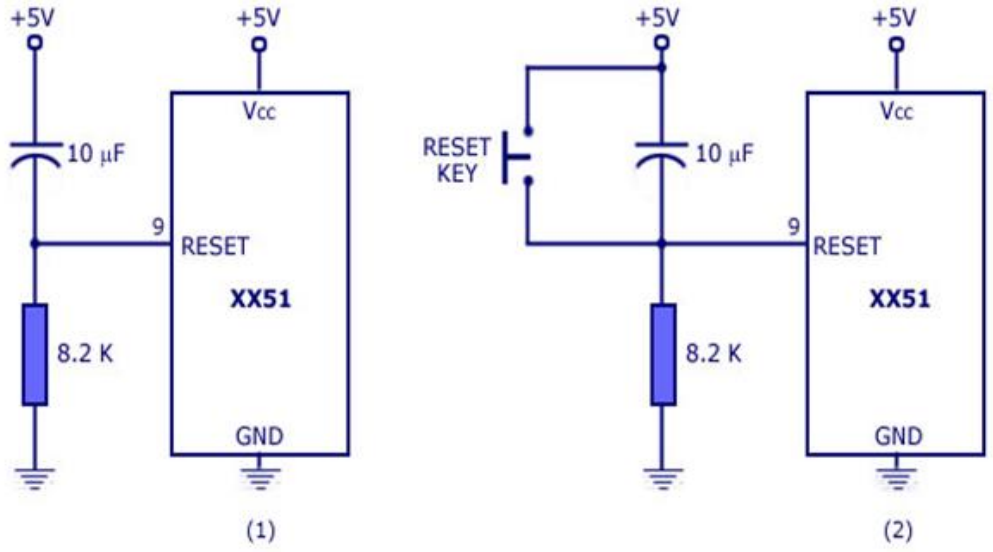
- It is an 8bit microcontroller.
- 8bit accumulator, 8bit Register and 8bit ALU.
- On chip RAM 128 bites (data memory).
- On chip ROM 4 Kbytes (program memory).
- Two 16bit counter/ timer.
- A 16 bit dptr(data pointer)
- 4 byte bi-directional input/ output port.
- Power saving mode (on some derivatives).
- 16bit address bus:-it can access 2^16 memory locations:-64kb (65536) each of RAM and ROM.
- It is an inclusion of Boolean processing system, have an ability to allow logic operations to be carried out on registers and RAM.
- 8bit data bus:-it can access 8bit of data in one operation.
- UART (this serial communication port makes chip to use simply as a serial communication interface).
- It has four separate Register set. (Each contains 8 Registers (R0 to R7)).

### B. 8051 Pin Description




**(1) Pins 1-8: P1.0-P1.7, I/O port 1**
a. An 8-bit bidirectional I/O port 1, P1.
b. With internal pullups.

**(2) Pin 9: RESET input**
a. The reset input of 8051. A high on this pin for two machine cycles while the oscillator is running resets the device. The 8051 executes from the program memory address 0000H.
b. This pin is used as the input of backup power supply. If the VCC is removed from main power supply and a +5V is connected to the reset input, the data in RAM will be remained.
c. Power-on reset circuit and RESET button are connected as shown in Fig. below




(1) Power-on Reset Circuit and (2) With Manual Reset Option

1. Power on Reset: Initially charging of capacitor makes RST High, When capacitor charges fully it blocks DC.
2. Manual Reset: Closing the switch momentarily will make RST High.

**(3) Pins 10-17: P3.0-P3.7, I/O port 3**

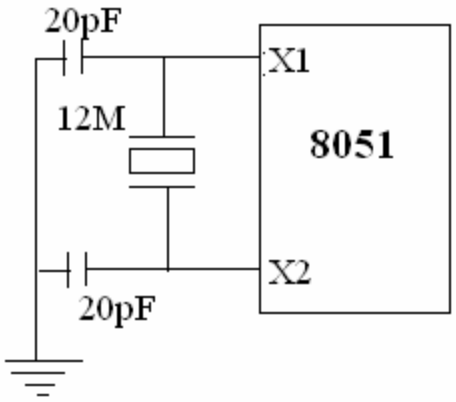
a. An 8-bit bidirectional I/O port
b. With internal pullups
c. P3 also serves the special features:

- P3.0 (RXD) Serial input port
- P3.1 (TXD) Serial output port
- P3.2 (INT0) External interrupt input 0
- P3.3 (INT1) External interrupt input 1
- P3.4 (T0) Timer/Counter 0 external input
- P3.5 (T1) Timer/Counter 1 external input
- P3.6 (WR) External data memory write strobe
- P3.7 (RD) External data memory read strobe

The CPU generates write and read signals, WR and RD, as needed during external data memory accesses (MOVX instruction).

**(4) Pins 18-19: X2, X1**

X1 is the input to the inverting oscillator amplifier and input to internal clock generator circuits. X2 is the output from the inverting oscillator amplifier. To generate the required clock signal, a crystal is connected across X1 and X2 pins, and two loading capacitors (20 pF) are required from X1 and X2 to ground. The connection of crystal oscillator is shown in Fig.below




**(5) Pin 20: Vss, ground, 0V reference**

**(6) Pins 21-28: P2.0-P2.7, I/O port 2**

a. An 8-bit bidirectional I/O port with internal pull-ups.
b. P2 emits the high-order address byte (A8-A15) during fetches from external program memory and during accesses to external data memory. In this application, it uses strong internal pull-ups when emitting 1s.

**(7) Pin 29: PSEN, Program Store Enable**

a. The program store enable is an output signal on pin 29. This signal is used for fetching instructions from external code memory. Program Store Enable is the read control signal to external program memory. When the 8051 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.
b. PSEN is not activated during fetches from internal program memory.

**(8) Pin 30: ALE/PROG, Address Latch Enable/programming pulse input**

a. Output pulse for latching the low byte address (A0-A7) during accesses to external memory.
b. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency. In programming 8751, this pin receives the programming pulses.

**(9) Pin 31: EA /Vpp, External Access Enable/Programming Supply Voltage**

a. This pin must be externally held low to enable the 8051 to fetch code from external program memory. For ROMless devices, such as the 8031, this pin should be wired to ground for using external ROM.
b. If EA is held high, the device executes from internal program memory unless the program counter (PC) contains an address greater than the maximum internal program memory space.
c. If the 8051 on-chip program memory is not used, this pin is directly wired to ground.

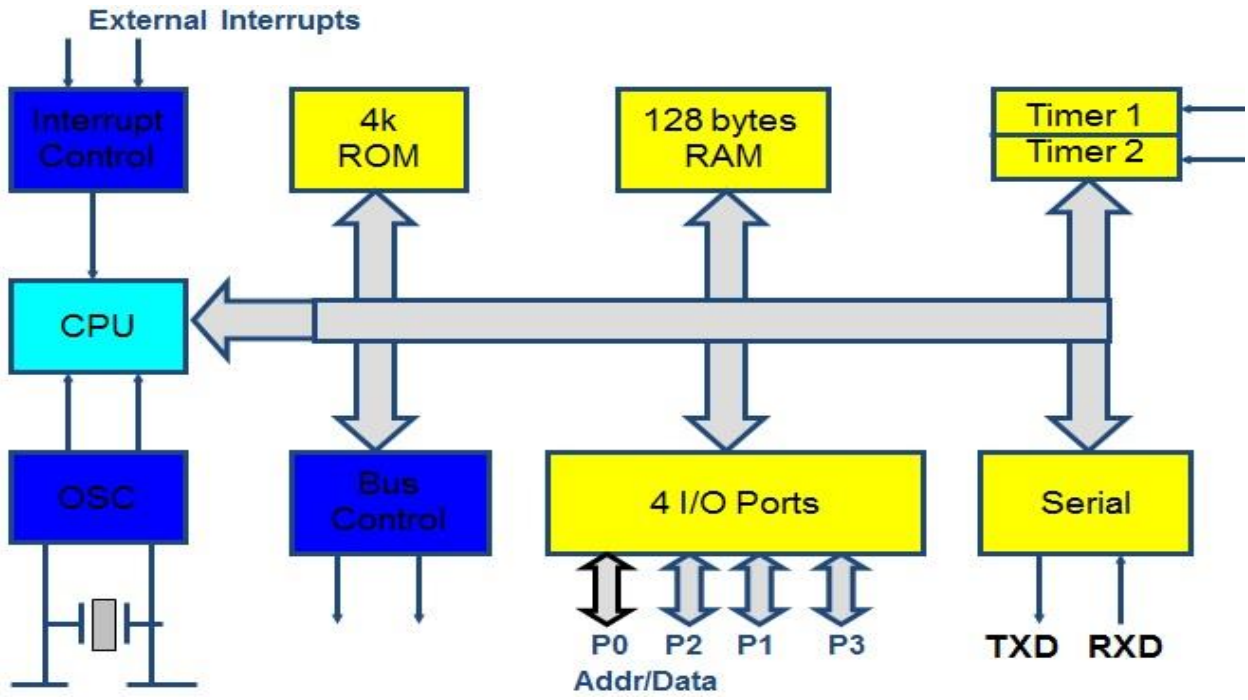
d. In programming mode, this pin must be connected to the programming power supply generally 12V.

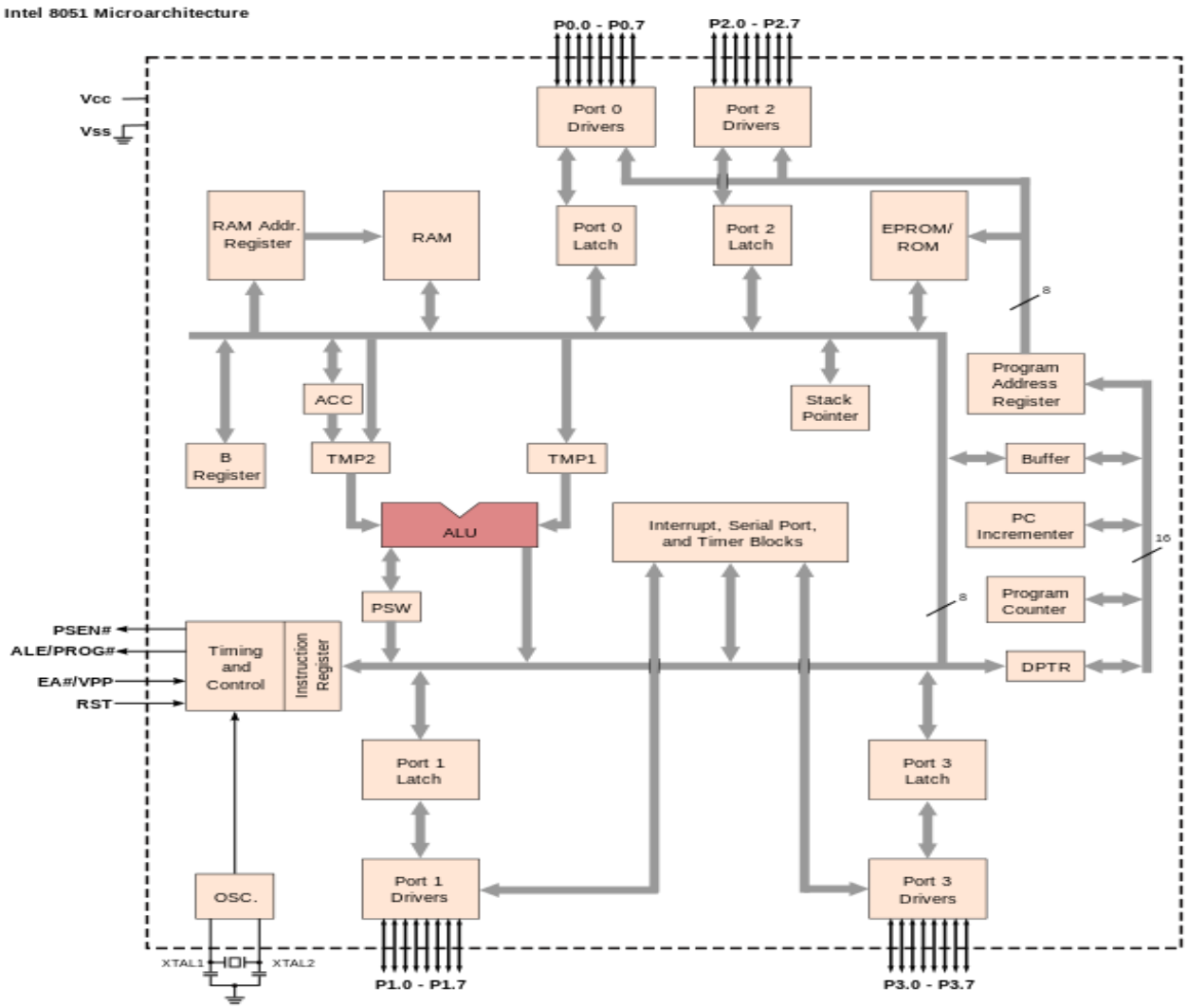
**(10) Pins 32-39: P0.0-P0.7, I/O port 0**

a. An 8-bit bidirectional I/O port, P0.
b. Port 0 is an open-drain configuration. When port 0 serves as an I/O port, an external pull-up resistor (generally 2.2 KΩ) is required for each line.
c. Port 0 pins that have 1s written to them float and can be used as high impedance inputs.
d. Port 0 is also the multiplexed low-order address (A0-A7) and data bus (D0-D7) during accesses to external program and data memory.

**(11) Pin 40: Vcc, positive power supply, +5V.**

**C. 8051 Architecture**






An 8051 microcontroller has the following 12 major components:

1. ALU (Arithmetic and Logic Unit)
2. PC (Program Counter)
3. Registers
4. Timing and control unit

5. Internal RAM and ROM
6. Four general purpose parallel input/output ports
7. Interrupt control logic with five sources of interrupt
8. Serial data communication
9. PSW (Program Status Word)
10. Data Pointer (DPTR)

11. Stack Pointer (SP)
12. Data and Address bus.

13. Timers and counters.

## C.1. ALU

All arithmetic and logical functions are carried out by the ALU.

Addition, subtraction with carry, and multiplication come under arithmetic operations.

Logical AND, OR and exclusive OR (XOR) come under logical operations.

## C.2. Program Counter (PC)

A program counter is a 16-bit register and it has no internal address. The basic function of program counter is to fetch from memory the address of the next instruction to be executed. The PC holds the address of the next instruction residing in memory and when a command is encountered, it produces that instruction. This way the PC increments automatically, holding the address of the next instruction.

## C.3. Registers

Registers are usually known as data storage devices. 8051 microcontroller has 2 registers, namely Register A and Register B. Register A serves as an accumulator while Register B functions as a general purpose register. These registers are used to store the output of mathematical and logical instructions.

The operations of addition, subtraction, multiplication and division are carried out by Register A. Register B is usually unused and comes into picture only when multiplication and division functions are carried out by Register A. Register A also involved in data transfers between the microcontroller and external memory.

8051 microcontroller also has 7 Special Function Registers (SFRs). They are:

1. Serial Port Data Buffer (SBUF)
2. Timer/Counter Control (TCON)
3. Timer/Counter Mode Control (TMOD)
4. Serial Port Control (SCON)
5. Power Control (PCON)
6. Interrupt Priority (IP)
7. Interrupt Enable Control (IE)

## C.4. Timing and control unit

Synchronization among internal operations can be achieved with the help of clock circuits which are responsible for generating clock pulses. During each clock pulse a particular operation will be carried out, thereby, assuring synchronization among operations. For the formation of an oscillator, we are provided with two pins XTAL1 and XTAL2 which are used for connecting a resonant network in 8051 microcontroller device. In addition to this, circuit also consists of four more pins. They are,
1. EA: External enable
2. ALE: Address latch enable
3. PSEN: Program store enable and
4. RST: Reset.
Quartz crystal is used to generate periodic clock pulses.

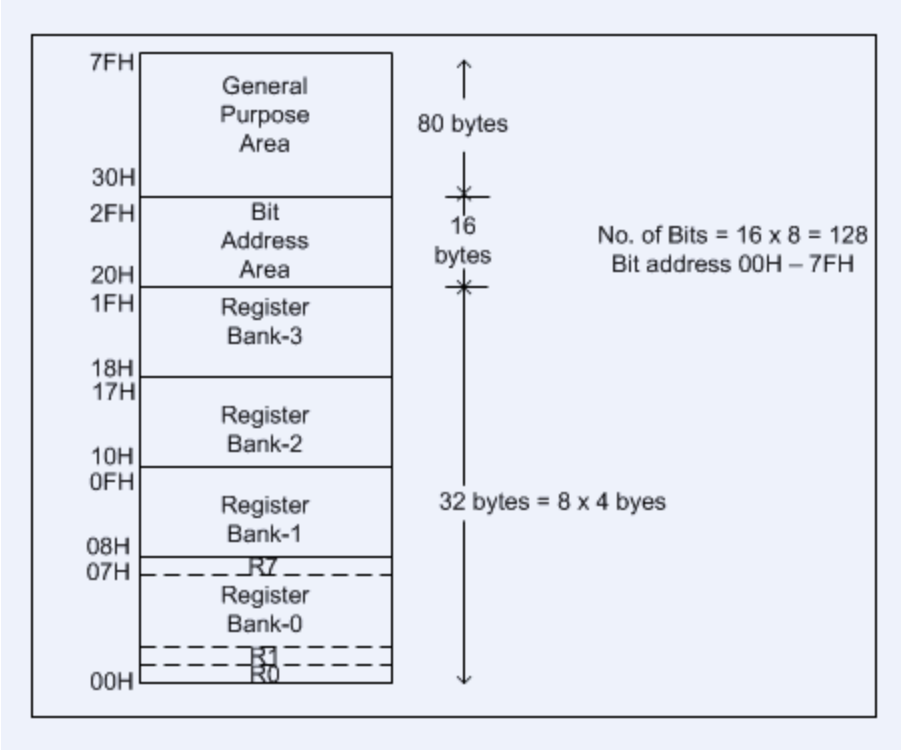
## C.5. Internal RAM and ROM

### ROM

A code of 4K memory is incorporated as on-chip ROM in 8051. The 8051 ROM is a non-volatile memory meaning that its contents cannot be altered and hence has a similar range of data and program memory, i.e, they can address program memory as well as a 64K separate block of data memory.

### RAM

The 8051 microcontroller is composed of 128 bytes of internal RAM. This is a volatile memory since its contents will be lost if power is switched off. These 128 bytes of internal RAM are divided into 32 working registers which in turn constitute 4 register banks (Bank 0-Bank 3) with each bank consisting of 8 registers (R0 - R7). There are 128 addressable bits in the internal RAM.




## C.6. Four General Purpose Parallel Input/Output Ports

The 8051 microcontroller has four 8-bit input/output ports. These are:

PORT P0: When there is no external memory present, this port acts as a general purpose input/output port. In the presence of external memory, it functions as a multiplexed address and data bus. It performs a dual role.

PORT P1: This port is used for various interfacing activities. This 8-bit port is a normal I/O port i.e. it does not perform dual functions.

PORT P2: Similar to PORT P0, this port can be used as a general purpose port when there is no external memory but when external memory is present it works in conjunction with PORT PO as an address bus. This is an 8-bit port and performs dual functions.

PORT P3: PORT P3 behaves as a dedicated I/O port

## C.7. Interrupt Control

An event which is used to suspend or halt the normal program execution for a temporary period of time in order to serve the request of another program or hardware device is called an interrupt. An interrupt can either be an internal or external event which suspends the microcontroller for a while and thereby obstructs the sequential flow of a program. The interrupt mechanism keeps the

normal program execution in a "put on hold" mode and executes a subroutine program and after the subroutine is executed, it gets back to its normal program execution.
In 8051, 5 sources of interrupts are provided. They are:

a) 2 external interrupt sources connected through INT0 and INT1
b) 3 external interrupt sources- serial port interrupt, Timer Flag 0 and Timer Flag 1.

There are two registers associated with interrupts:
IP (Interrupt priority register)
The IP register is use to define the priority levels (high and low) of interrupt resources.
IE(Interrupt enable register)
The IE register is used to enable or disable the interrupt resources.

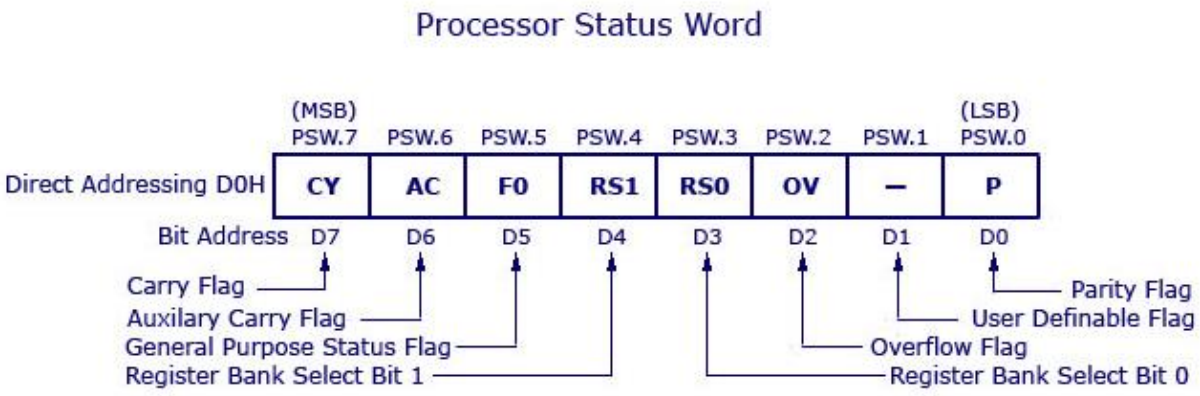
## C.8. Serial Data Communication

A method of establishing communication among computers is by transmitting and receiving data bits is a serial connection network. In 8051, the SBUF (Serial Port Data Buffer) register holds the data; the SCON (Serial Control) register manages the data communication and one bit in the PCON (Power Control) register manages the data transfer rates. Further, two pins - RXD and TXD, establish the serial network.

The SBUF register has 2 parts – one for storing the data to be transmitted and another for receiving data from outer sources. The first function is done using TXD pin and the second function is done using RXD pin.

## C.9. PSW (Program Status Word)

Program Status Word or PSW is a hardware register which is a memory location which holds a program's information and also monitors the status of the program this is currently being executed. This register reflects the status of the operation that is being carried out in the processor. The picture below shows PSW register and the way register banks are selected using PSW register bits – RS1 and RS0. PSW register is both bit and byte addressable. The physical address of PSW starts from D0H. The individual bits are then accessed using D1, D2 … D7. The various individual bits are explained below.




Generally, the instruction of the result of a program is stored in a single bit register called a 'flag'. The are7 flags in the PSW of 8051. Among these 7 flags, 4 are math flags and 3 are general purpose or user flags.

The 4 Math flags are:

- Carry (c)
- Auxiliary carry (AC)
- Overflow (OV)
- Parity (P)

The 3 General purpose flags or User flags are:

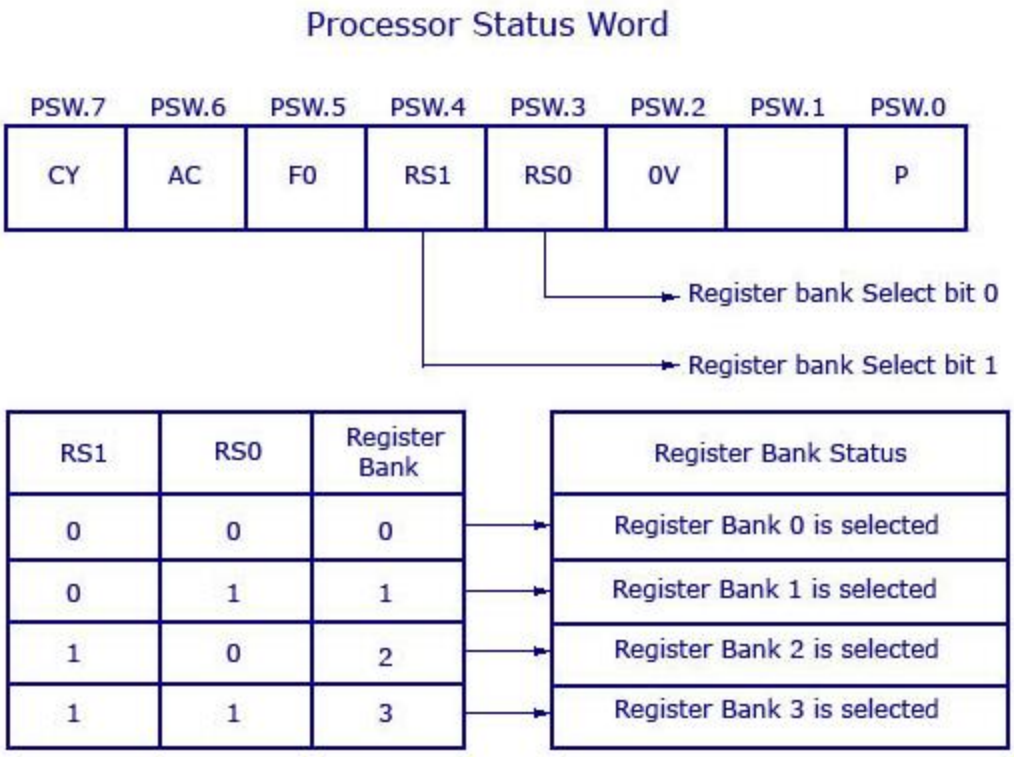
- FO
- GFO
- GF 1

CY: Carry flag. The CY is set if there is a carry-out or borrow-in for the MSB of the result during the execution of an arithmetic instruction. Otherwise, CY is reset.

AC: Auxiliary Carry flag. The AC is set if there is a carry-out from the low nibble into the high nibble or a borrow-in from the high nibble into the low nibble.Otherwise, AC is reset. This flag can be used in the adjustment of binary to BCD.

F0: Flag 0 is available to the user for general purposes.

RS0, RS1: Register bank select bits 0 and 1.




OV: Overflow flag. When OV is set, it indicates that the signed result is out of range. If the result is not out of range, OV remains reset. OV flag is set if there is a carry from bit 6 but not from bit 7 of an Arithmetic operation. It's also set if there is a carry from bit 7 (but not from bit 6) of Acc

P: Parity bit. The P flag is set if the result produce by the instruction has odd parity, that is, if it contains an odd number of bits at the 1 logic level. If parity is even, P flag is reset.

### C.10. Data Pointer (DPTR)

The data pointer or DPTR is a 16-bit register. It is made up of two 8-bit registers called DPH and DPL. Separate addresses are assigned to each of DPH and DPL. These 8-bit registers are used for the storing the memory addresses that can be used to access internal and external data/code.

### C.11. Stack Pointer (SP)

The stack pointer (SP) in 8051 is an 8-bit register. The main purpose of SP is to access the stack. As it has 8-bits it can take values in the range 00 H to FF H. Stack is a special area of data in memory. The SP acts as a pointer for an address those points to the top of the stack.

### C.12. Timers and Counters

The 8051 has two 16-bit timer/counter registers: timer 0and timer 1. The 8052 has these two plus one more: timer 2. All these can be configured to operate either as timers or event counters.

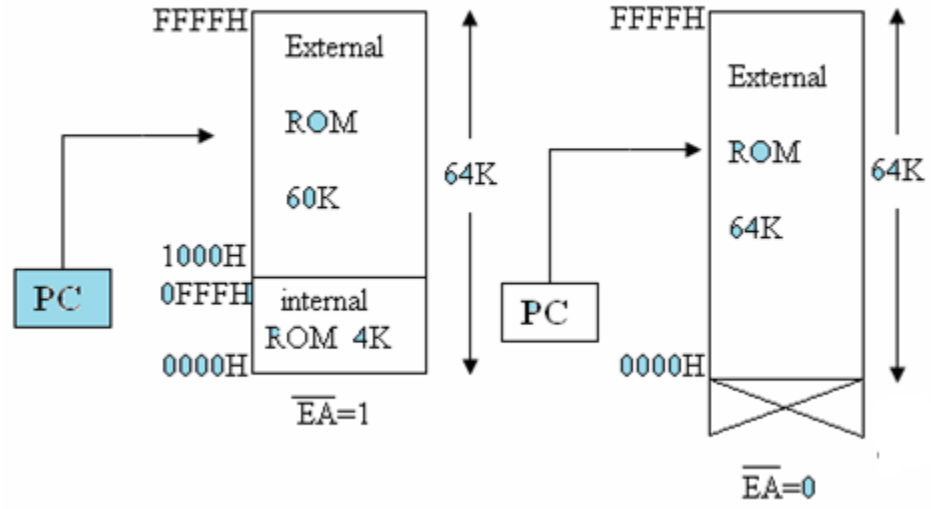
The registers associated with Timer operation are:

TH1, TL1 (Timer/counter 1 register high low bytes): The two 8-bit registers are used to store the count value of timer counter 1.
TH0, TL0 (Timer/counter 0 register ): The two 8-bit registers are used to store the count value of timer/counter 0.
TMOD (Timer/counter mode control register): It is used to define the operating modes of timer/ counters. Timer or counter selection.
TCON (Timer /counter control register): It contains the overflow flags and run control bits of timer counters, and the edge flag and type control bits of interrupts.

## D. Memory organization

### D.1 Program Memory

Program memory is used to store instruction code. The program memory is mask ROM or PROM in the Intel 8051, EPROM in Intel 8751, EEPROM in AtmelAT89C51/52, and Flash in Philips 8051.
The 8051 contains 4K-byte internal program memory, 0000H-0FFFH, and is expandable up to 60K bytes of external program memory. Program memory address is determined by the contents of program counter (PC). The map of 8051 program memory is shown in Fig. below.



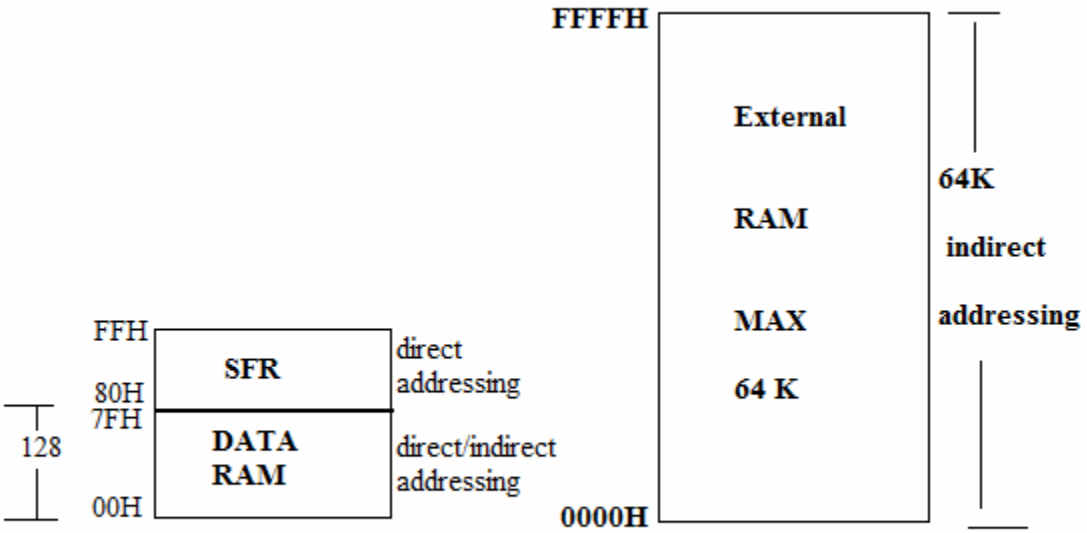

**8051 program memory map**

If both internal and external ROMs are used, EA pin should be wired to high. When the contents of program counter are less than 0FFFH, the CPU fetches the code stored in internal ROM; otherwise, the CPU fetches the code stored in external ROM.

If EA pin is wired to ground, all program fetches are directed to external ROM. The ROMless devices such as 8031 must have this pin externally strapped to ground potential to enable them to execute properly. Mask ROM devices can also be used as ROMless devices 8051 if external ROM is added.

**D.2 Data Memory**

The 8051 contains 128 bytes of on-chip (internal) RAM, locations 00H-7FH, accessible by direct and indirect addressing. This block of internal RAM is used for data, stack, and registers. In fact,. A block of 128 bytes, 80H-FFH, is used as the Special Function Registers (SFRs) such as accumulator, stack pointer, timer registers, etc. This block is accessed by direct addressing. The data memory is expandable up to 64K bytes of external data memory. The 8051 treats internal RAM and external RAM as two physically separate blocks and accesses them by different instructions. The internal RAM is accessed by the MOV instruction, whereas the external RAM is accessed by the MOVX instruction. The maximum RAM space of 8051 system is equal to 64K plus 128 bytes. Fig. below shows the 8051 data memory map.

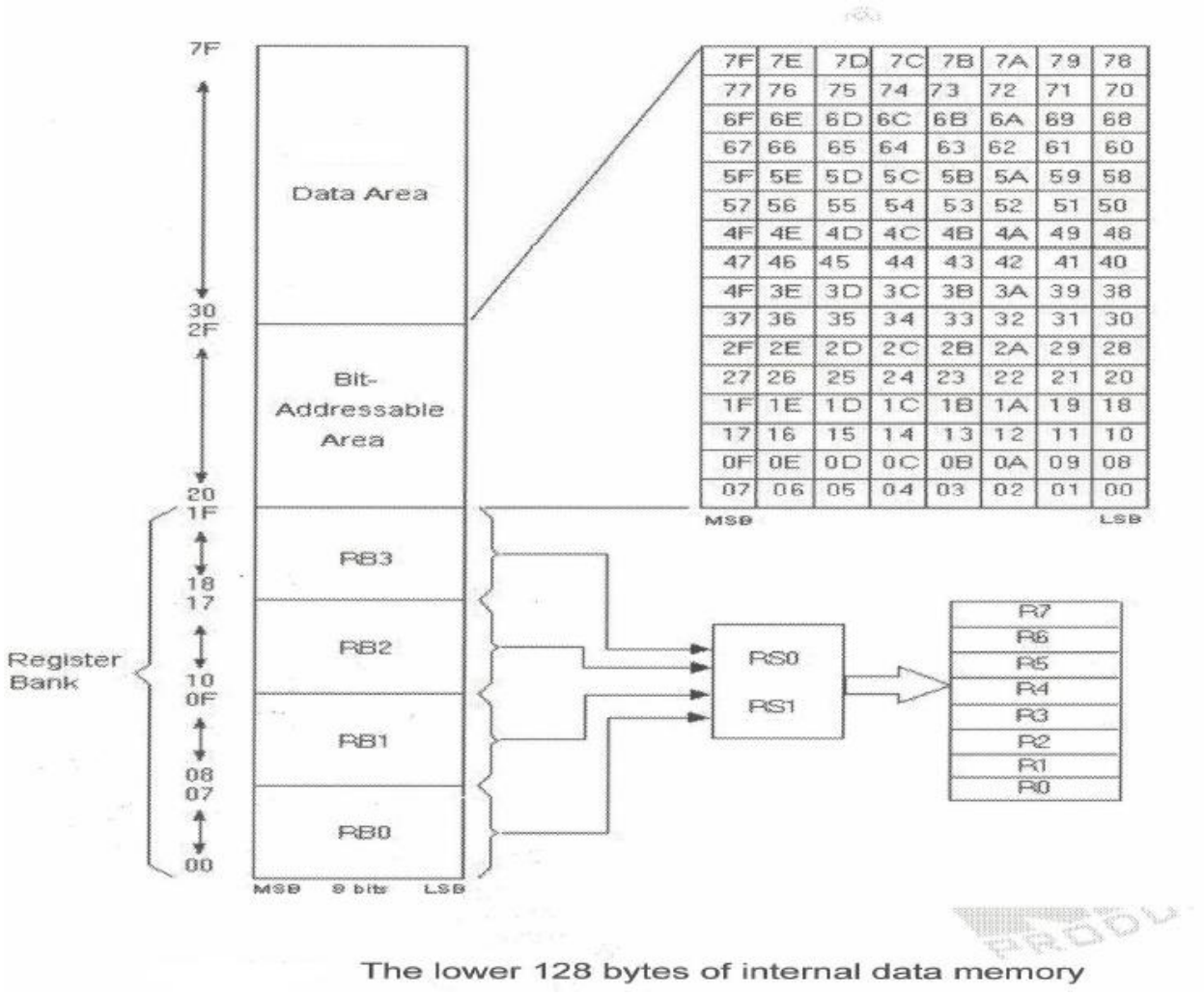



**8051 data memory map**

8051 data memory is divided into three spaces:
(1) Locations 00H-7FH internal data memory
(2) Locations 80H-FFH SFR space
(3) 64K bytes of external data memory

**Internal Data Memory**

8051 internal data memory is divided into three areas as shown in. Fig. below




The lower 128 bytes of internal data memory

**1. Locations 00H-1FH: Register banks 0-3**

The lowest 32 bytes of RAM are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. These 4 register banks are referred to as RB0 through RB3. Two bits, RS0 and RS1, in the program status word (PSW) register select which register bank is in use as shown in Table below. This allows more efficient use of code space, since register instructions are shorter than the instructions that use direct addressing.

Register bank selection

| RS1 | RS0 | Register Bank |
|---|---|---|
| 0 | 0 | RB0 |
| 0 | 1 | RB1 |
| 1 | 0 | RB2 |
| 1 | 1 | RB3 |

Any register name in register instructions is actually an address of the 32-byte RAM as shown in Table below. If RB2 is selected, for example, then the R3 register represents the address 13H. After reset, the register bank 0, RB0, is default.

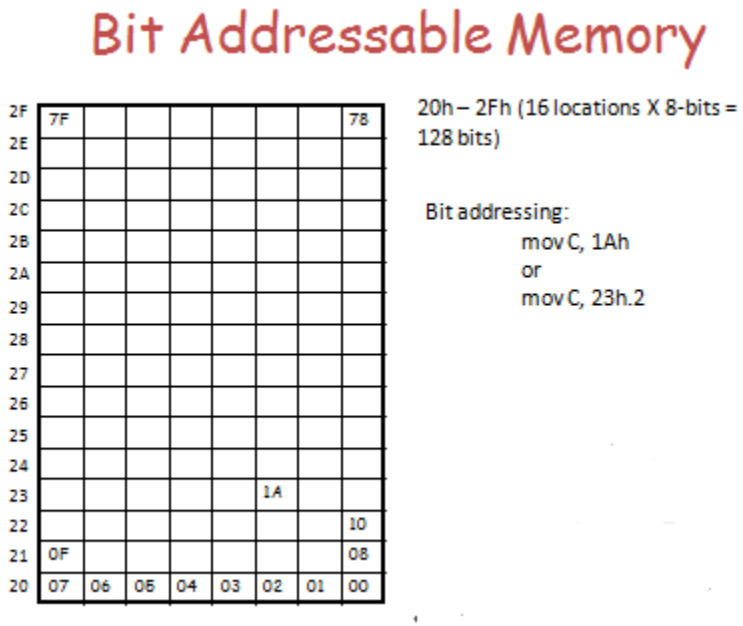
Register addresses in different register banks

| Reg. name \ Reg. bank | RBb0 | RB1 | RB2 | RB3 |
|---|---|---|---|---|
| R0 | 00H | 08H | 10H | 18H |
| R1 | 01H | 09H | 11H | 19H |
| R2 | 02H | 0AH | 12H | 1AH |
| R3 | 03H | 0BH | 13H | 1BH |
| R4 | 04H | 0CH | 14H | 1CH |
| R5 | 05H | 0DH | 15H | 1DH |
| R6 | 06H | 0EH | 16H | 1EH |
| R7 | 07H | 0FH | 17H | 1FH |

**2. Locations 20H-2FH: Bit addressable area**

Sixteen bytes have been assigned for this area, 20H-2FH. Each of the 16 bytes in this segment can be addressed as a byte. Each one of the 128 bits of this area can also be directly addressed (00H-7FH).

The bits can be referred to in two ways both of which are acceptable by the 8051. One way is to refer to their addresses, 00H to 7FH. The other way is with reference to bytes 20H to 2FH.



**3. Locations 30H-7FH: Scratch pad area**

The bytes 30H through 7FH are available to the user as data RAM. However, if the stack pointer has been initialized to this area, enough number of bytes should be left aside to prevent SP data destruction. The first-in-last-out (FILO) stack is used to save the contents of certain registers or some their main program parameters after the switching to a subroutine.

**4. Special Function Registers (SFRs)**

Special function registers (SFRs) include the I/O port latches, timers, peripheral controls, etc These registers uses the address from 80H to FFH and can only be accessed by direct addressing.

| Byte address | Bit address | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 98 | 9F | 9E | 9D | 9C | 9B | 9A | 99 | 98 | SCON |
| | | | | | | | | | |
| 90 | 97 | 96 | 95 | 94 | 93 | 92 | 91 | 90 | P1 |
| | | | | | | | | | |
| 8D | not bit addressable | | | | | | | | TH1 |
| 8C | not bit addressable | | | | | | | | TH0 |
| 8B | not bit addressable | | | | | | | | TL1 |
| 8A | not bit addressable | | | | | | | | TL0 |
| 89 | not bit addressable | | | | | | | | TMOD |
| 88 | 8F | 8E | 8D | 8C | 8B | 8A | 89 | 88 | TCON |
| 87 | not bit addressable | | | | | | | | PCON |
| | | | | | | | | | |
| 83 | not bit addressable | | | | | | | | DPH |
| 82 | not bit addressable | | | | | | | | DPL |
| 81 | not bit addressable | | | | | | | | SP |
| 80 | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 | P0 |

| Byte address | Bit address | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| FF | | | | | | | | | |
| F0 | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 | B |
| | | | | | | | | | |
| E0 | E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 | ACC |
| | | | | | | | | | |
| D0 | D7 | D6 | D5 | D4 | D3 | D2 | – | D0 | PSW |
| | | | | | | | | | |
| B8 | – | – | – | BC | BB | BA | B9 | B8 | IP |
| | | | | | | | | | |
| B0 | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | P3 |
| | | | | | | | | | |
| A8 | AF | – | – | AC | AB | AA | A9 | A8 | IE |
| | | | | | | | | | |
| A0 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | P2 |
| | | | | | | | | | |
| 99 | not bit addressable | | | | | | | | SBUF |

**5. External Data Memory**

The 8051 can address up to 64K bytes of external data memory. Similar to the external ROM accesses, port 0 in this case serves as a multiplexed address/data bus to the external RAM. The CPU generates RD and WR signals as needed during external RAM accesses. During the time that low-order address signals (A0-A7) are valid onP0, the signal ALE clocks this byte into the address latch 74373. Meanwhile, P2 emits the high-order address signals (A8-A12). The external data memory is indirectly addressed with the MOVX instruction.

## E. Stack Memory of 8051

The stack is a section of RAM used by the CPU to store information temporarily. This information could be data or an address. The CPU needs this storage area since there are only a limited number of registers.

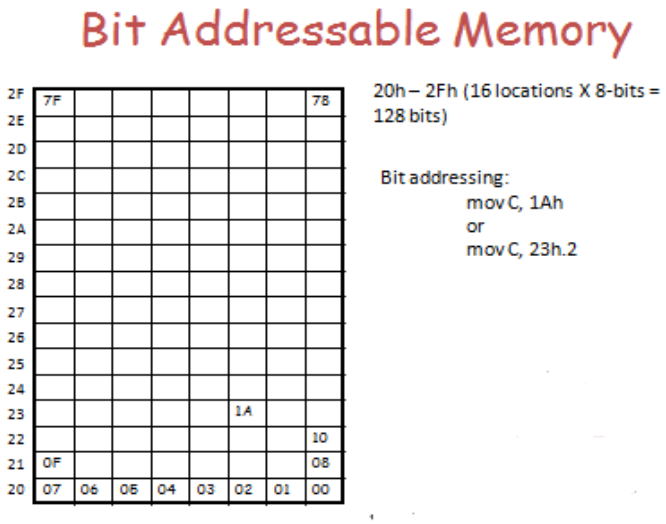
The register used to access the stack is called the SP (stack pointer) register. The stack pointer in the 8051 is only 8 bits wide, which means that it can take values of 00 to FFH. When the 8051 is powered up, the SP register contains value 07. This means that RAM location 08 is the first location used for the stack by the 8051. The content of SP can be changed using MOV SP, #8 bit data instruction. The storing of a CPU register in the stack is called a PUSH, and pulling the

contents off the stack back into a CPU register is called a POP. In other words, a register is pushed onto the stack to save it and popped off the stack to retrieve it.
The locations 08 to IF in the 8051 RAM can be used for the stack. This is because locations 20 – 2FH of RAM are reserved for bit-addressable memory and must not be used by the stack. If in a given program we need more than 24 bytes (08 to 1FH = 24 bytes) of stack, we can change the SP to point to RAM locations 30 – 7FH. This is done with the instruction "MOV SP, #xx".

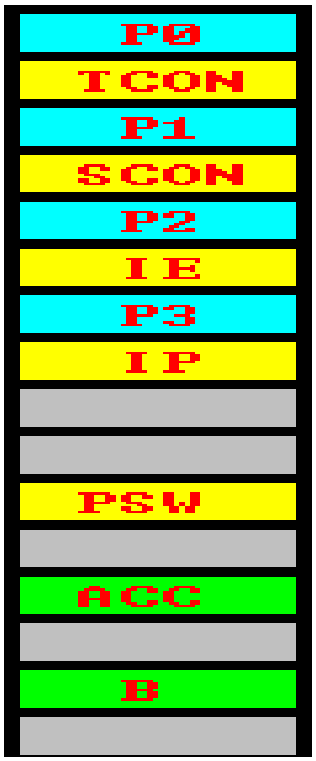
## 5. Special Features of 8051, Boolean Processor and Power saving options

### A. Boolean Processor

The 8051 processor contains a complete Boolean processor for single-bit operations. The internal RAM contains 128 addressable bits, and the SFR space supports up to 128 other addressable bits. All port lines are bit-addressable, and each can be treated as a separate single-bit port. The instructions that access these bits are not only conditional branches move, set, clear, complement, OR, and AND instructions. Such bit operations—one of the most powerful features of the MCS-51TM family of microcontrollers are not easily obtained in other architectures with byte-oriented operations.
All bit accesses use direct addressing with bit addresses 00H-7FH in the lower 128 locations, and bit addresses 80H–FFH in the SFR space, Those in the lower 128 locations at byte addresses 20H-2FH are numbered sequentially from bit 0 of address 20H (bit 00H) to bit 7 of address 2FH (bit 7FH). Bits may be set or cleared in a single instruction. Single-bit control is common for many I/0 devices, including output to relays, motors, solenoids, status LEDs, buzzers, alarms, loudspeakers, or input from a variety of switches or status indicators. If an alarm is connected to Port 1 bit 7, for example, it might be turned on by setting the port bit.




**Bit Addressable SFRs**



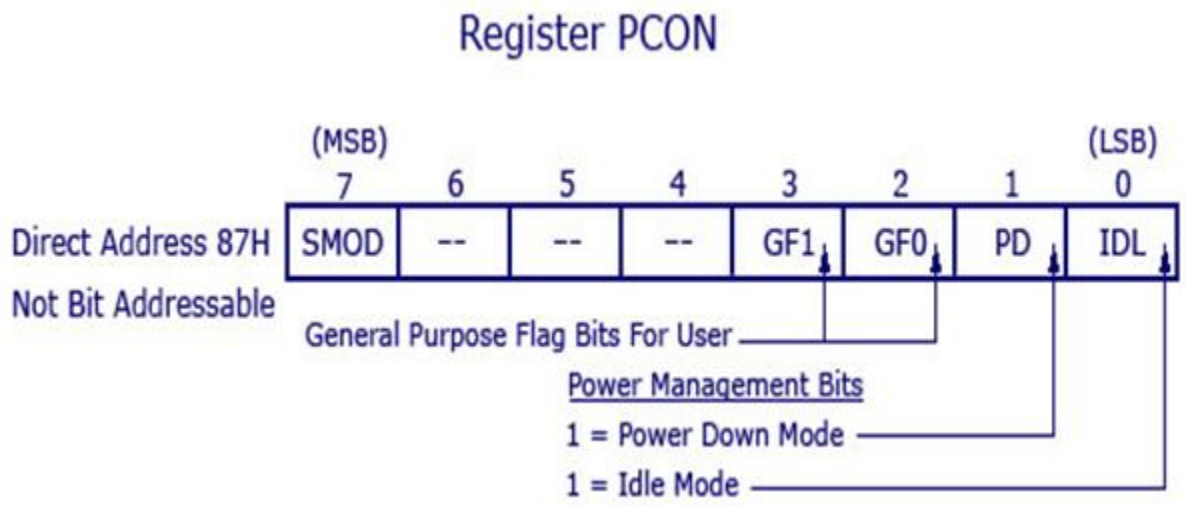

List of bit addressable Instructions in 8051:

| | |
|---|---|
| CLR C | Clears the carry flag |
| CLR bit | Clears the direct bit |
| SETB C | Sets the carry flag |
| SETB bit | Sets the direct bit |
| CPL C | Complements the carry flag |
| CPL bit | Complements the direct bit |
| ANL C,bit | AND direct bit to the carry flag |
| ANL C,/bit | AND complements of direct bit to the carry flag |
| ORL C,bit | OR direct bit to the carry flag |
| ORL C,/bit | OR complements of direct bit to the carry flag |
| MOV C,bit | Moves the direct bit to the carry flag |
| MOV bit,C | Moves the carry flag to the direct bit |

For Example ANL C, 20h will AND the bit content of carry flag and the content of the bit address 20H and store the result in carry flag.

### B. Power saving options of 8051

The 8051 series µCs offer idle and power-down modes, both of which can be invoked by setting appropriate control bits from the code. For instance, a 5V, 12-MHz Atmel AT89C2051 consumes approximately 9 mA in active mode, 1.8 mA in idle mode, and only 12 µA in power-down mode.
PCON register is used to enter into power saving options.




| Symbol | Position | Name & Significance |
|---|---|---|
| SMOD | PCON.7 | Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD = 1, the baud rate is doubled when the serial port is used in modes 1, 2, or 3. |
| - | PCON.6 | Not implemented |
| - | PCON.5 | Not implemented |
| - | PCON.4 | Not implemented |
| GF1 | PCON.3 | General-purpose flag bit. |
| GF0 | PCON.2 | General-purpose flag bit. |
| PD | PCON.1 | Power Down bit. Setting this bit activates the Power Down operation in the 8051BH. (Available only in CHMOS). |

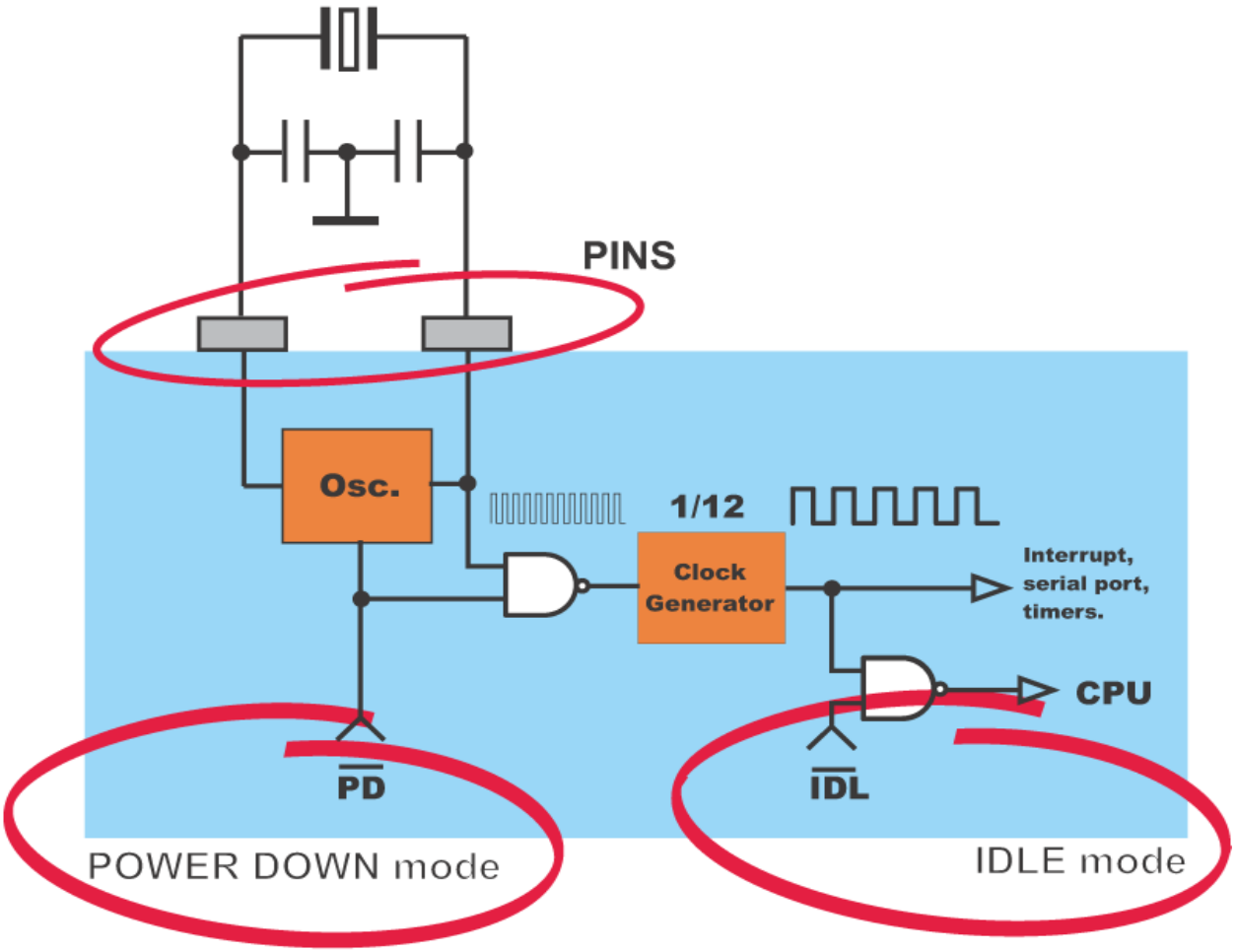
| | | |
|---|---|---|
| IDL | PCON.0 | Idle Mode bit. Setting this bit activates Idle Mode operation in the 8051BH. (Available only in CHMOS). |

**Power Down mode:**

- By setting the PD bit of the PCON register from within the program, the microcontroller is set to Power down mode.
- Thus turning off its internal oscillator and reduces power consumption enormously.
- The microcontroller can operate using only 2V power supply in power- down mode, while total power consumption is less than 40uA.
- The only way to get the micro controller back to normal mode is by reset.
- While the microcontroller is in Power Down mode, the state of all SFR registers and I/O ports remains unchanged.
- By setting it back into the normal mode, the contents of the SFR register is lost, but the content of internal RAM is saved.
- Reset signal must be long enough, approximately 10mS, to enable stable operation of the quartz oscillator.

**Idle Mode**

- Upon the IDL bit of the PCON register is set, the microcontroller turns off the greatest power consumer- CPU unit while peripheral units such as serial port, timers and interrupt system continue operating normally consuming below 6.5mA.
- In Idle mode, the state of all registers and I/O ports remains unchanged.
- In order to exit the Idle mode and make the microcontroller operate normally, it is necessary to enable and execute any interrupt or reset.
- Ways to exit Idle Mode:
- Activation of any enabled interrupt will clear PCON.0 bit and hence the Idle Mode is exited.
- A hardware reset exits the idle mode. The CPU starts from the instruction following the instruction that invoked the ‘Idle' mode.




**6. Derivatives of 8051:- 8031, 8751,8952, 89V51RD2**

| FEATURE | 8051 | 89s51 | 89s52 | 8031 | 8751 | 89v51RD2 |
|---|---|---|---|---|---|---|
| ROM(bytes) | 4K (mask ROM) | 4K (Flash) | 8K (Flash) | 0 | 4K (UV-EPROM) | 64 kB (Flash) |
| RAM(bytes) | 128 | 128 | 256 | 128 | 128 | 1k |
| Timers (16 bit) | 2 | 2 | 3 | 2 | 2 | 3 |

| | | | | | | |
|---|---|---|---|---|---|---|
| I/O pins | 32 | 32 | 32 | 32 | 32 | 32 pins with three high-current Port 1 pins (16 mA each) |
| ROM(bytes) | 4K (mask ROM) | 4K (Flash) | 8K (Flash) | 0 | 4K (UV-EPROM) | 64 kB (Flash) |
| RAM(bytes) | 128 | 128 | 256 | 128 | 128 | 1k |
| Timers (16 bit) | 2 | 2 | 3 | 2 | 2 | 3 |
| I/O pins | 32 | 32 | 32 | 32 | 32 | 32 pins with three high-current Port 1 pins (16 mA each) |