UNIT – III: PIC 18FXXX Microcontroller Architecture

Q. 1 List Features of the PIC18 microcontroller

- It uses RISC architecture
- It has 8-bit CPU,16 bit instructions
- 2 MB program memory space
- 256 bytes to 1KB of data EEPROM
- Up to 3968 bytes of on-chip SRAM
- 4 KB to 128KB flash program memory
- 4 Timers/ Counters,(4 Timers)
- 5 ports A[6], B,C,D[8], E[4] =35 IO lines (USB)
- PWM, real-time interrupt, and watchdog timer
- Serial communication interfaces: SCI, SPI, I2C
- Supports for CCP Mode
- Background debug mode (BDM)
- 10-bit A/D converter
- Instruction pipelining
- Operates at up to 40 MHz crystal oscillator.
- Programmable Brown out reset module, Power On Reset(POR), Power-saving mode
-

Q. 2 Explain Criteria for Choosing Microcontroller

While selecting Microcontroller series for particular application following criteria has to be considered

1. Data Handling Capacity- Bits, Nibble Bytes, Words, Double words, Quad Words etc.
2. Speed --- Depends on Clock
3. Amount of RAM/ ROM/ EPROM/ Flash/ Static
4. Number of I/O pins, Timers – All SFRS
5. Power consumption – Based on the modes
6. Packaging – 40 PIN DIP,/ QFP/ other– Important – Space, assembly, Prototyping the end Product
7. Added features like ADC/ DAC/ CCP, Bus support like CAN, SPI, I2C, USB.
8. Watchdog timer, Timer modes, Data EEPROM etc.
9. Easy to upgrade --Higher Performances or low power operations
10. Availability of Software and Hardware Development Tools like
    1. Assembler
    2. Compilers – Code efficient C language
    3. Debuggers
    4. Emulators
    5. Technical Support
11. Wide Availability and Reliable Sources of Microcontrollers

Q. 3 compare various PIC series architectures.

All PIC series architectures can be compared according to following features,

Data Bus Width

Program Memory

Data Memory

No. of I/O Pins used for Interfacing

Enhanced Modes of Operation (CCP, PWM, ISP)

Modes of Serial Communications (MSSP) – I2C and SPI

Timers used for Setting Delay

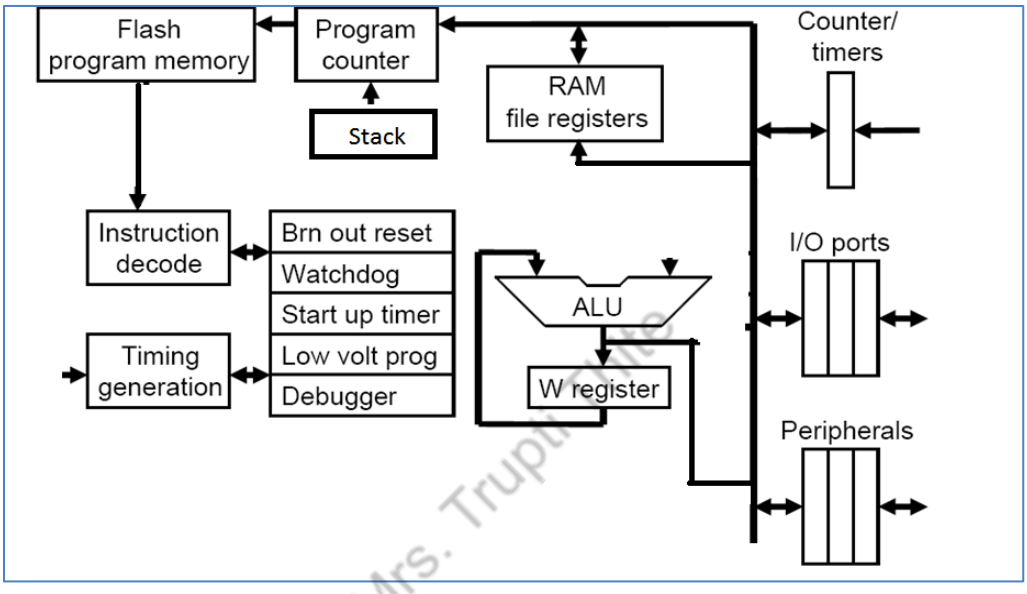
Most Important Special features

Processor	Code ROM	Data RAM	Data EEPROM	I/O Pins	ADC	Timers	PIN and Pack
PIC18F1220	4K (flash)	256	256	16	10-bit	4	18 DIP
PIC18F2420	16K (flash)	768	0	25	10-bit	4	28 DIP
PIC18F2220	4K (flash)	512	256	25	10-bit	4	28 DIP
PIC18F452	32K (flash)	1536	256	34	10-bit	4	40 DIP
PIC18F4520	32K (flash)	1536	256	36	10-bit	4	40 DIP
PIC18F458	32K (flash)	1536	256	33	10-bit	4	40 DIP
PIC18F4580	32K (flash)	1536	256	36	10-bit	4	40 DIP
PIC18F8722	128K (flash)	3936	1024	70	10-bit	5	80 TQFP
PIC18F4550	32K flash	2048	256	35	10 bit /13	4	40 DIP

Q 4. Draw and explain PIC 18F architecture

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, ADC etc.
Support for serial communication using Various Buses like SPI, I2C



MCU unit includes: Arithmetic Logic Unit (ALU), Registers, and Control Unit
Arithmetic Logic Unit (ALU)

- WREG (8)– working register (Accumulator)
- Status register that stores flags
- Instruction decoder – when the instruction is fetched it goes into the ID

The CPU fetches instructions from memory, decodes them, and passes them to the ALU for execution.

- The arithmetic logic unit (ALU) is responsible for adding, subtracting, shifting and performing logical operations.
  The ALU operates in conjunction with:
  - A general purpose register called W register
  - And f register that can be any location in data location

Control Unit:

- Provides timing and control signals to various Read and Write operations

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: 16-bit instruction/data bus for program memory and 8-bit data bus for data memory

I/O ports:

The series of PIC18 consists of five ports such as Port A, Port B, Port C, Port D & Port E. These ports are used to connect digital Input output Devices

Timers

PIC microcontroller has four timer/counters wherein the one 8-bit timer and the remaining timers have the choice to select 8 or 16-bit mode. Timers are used for generating accuracy actions, for example, creating specific time delays between two operations.

ADC

The main intention of this analog to digital converter is to convert analog voltage values to digital voltage values. The operation of the analog to digital converter is controlled by ADCON0 and ADCON1 special registers.

CCP module

The name CCP module stands for capture/compare/PWM where it works in three modes such as capture mode, compare mode and PWM mode

Serial Communication

Serial communication is the method of transferring data one bit at a time sequentially over a communication channel. PIC 18 has SPI, I2C and USART protocol for serial communication.

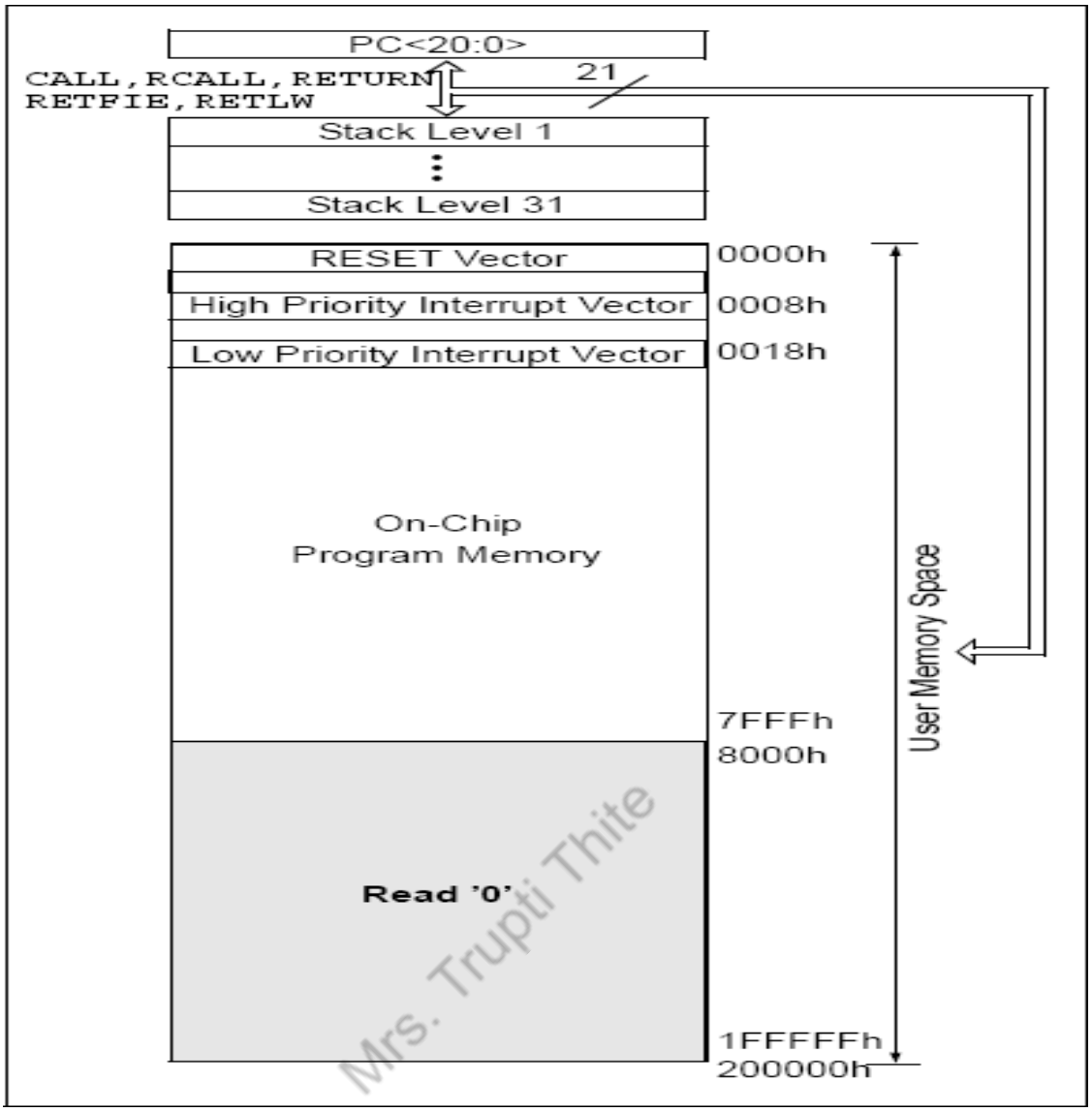
Q 5. Draw and Explain Program Memory map of PIC18Fxxx.

- Each PIC18F member has a 21-bit program counter PC (21 bit). It is used by CPU to point to the address of next instruction to be executed from ROM.
- A 21-bit program counter is capable of addressing the 2Mbyte program memory space

1FFFFF=221=2M

- User memory space on the PIC18 microcontroller is 00000H to 7FFFH. Accessing a nonexistent memory location (8000H to 1FFFFFH) will cause a read of all 0s.
- The reset vector where the program starts after a reset is at address 0000H. Addresses 0008H and 0018H are reserved for the vectors of high-priority and low-priority interrupts, respectively, and interrupt service routines must be written to start at one of these locations.
- The PIC18F microcontroller has a 31-entry stack that is used to hold the return addresses for subroutine calls and interrupt processing. The stack is controlled by a 5-bit stack pointer, which is initialized to 00000 after a reset.
- PIC18F452 each have 32 Kbytes of FLASH memory. This means that it can store up to 16K of single word instructions Total of word data size= 32K*16

The program memory map is shown below



Q 6. Draw and Explain Data Memory map of PIC18Fxxx

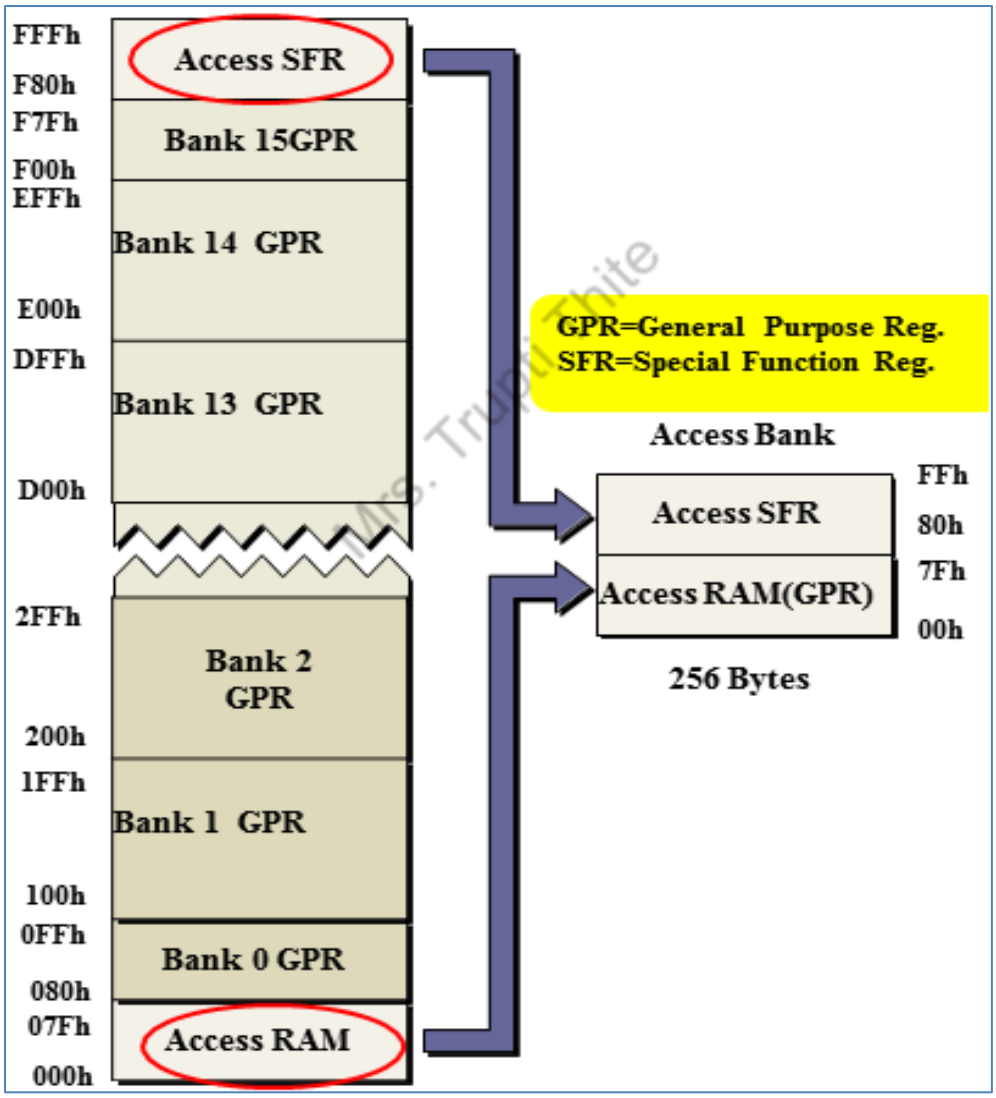
- Data Memory is also called as File Register
- The data memory address bus is 12 bits, with the capability of addressing up to 4 KB.
- The memory in general consists of 16 banks, each of 256 bytes.
- Bank switching is done automatically when using a high-level language compiler, and thus the user need not worry about selecting memory banks during programming.

- There are two types of registers in the data memory, the **General Purpose Register (GPRs**) and the **Special function Registers (SFRs).**
- General purpose registers (GPRs) are used to hold dynamic data during program execution.
- SFR has dedicated functions Such as ALU Status, Timers, Serial communication. ADC, I/O Ports etc.

- Bank Select Register (BSR) is used to select the bank.

- PIC18 implements the **access bank** to reduce the problem caused by bank switching.
- It is the default bank when we power up the PIC18
- Access bank is formed using 128 bytes from GPR 00h to 7Fh and 128 bytes from SFR F80h to FFFh

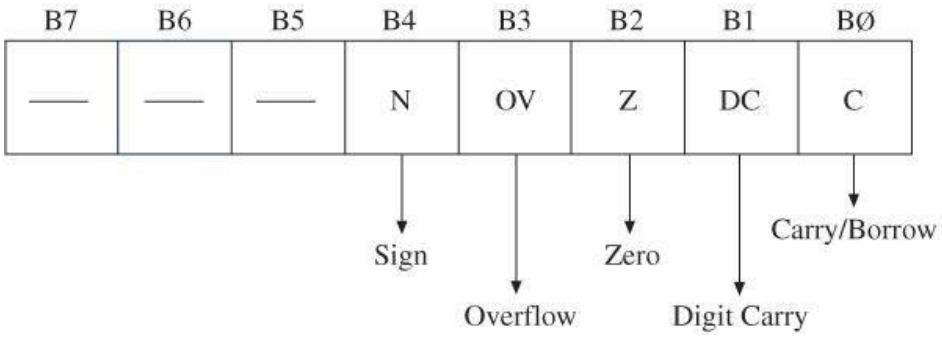
- The data memory map of the 18F microcontroller is shown below.



Q 7. Explain working register (WREG) & status register in PIC18.

- **Working Register (W):** It is 8 bit Temporary holding register. Called as Accumulator.
  - ✓ It Cannot be access Directly
  - ✓ It is used by many instructions as source of an operand.
  - ✓ It is used as Destination for the result of instruction execution

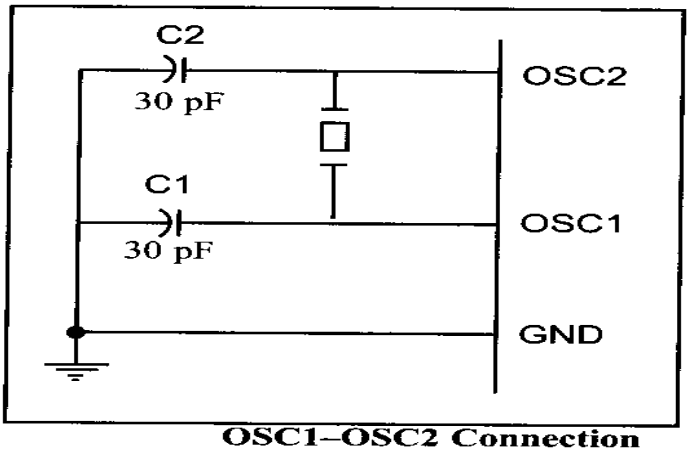
- **status register:** It is an 8-bit register referred as flag registers and all are conditional



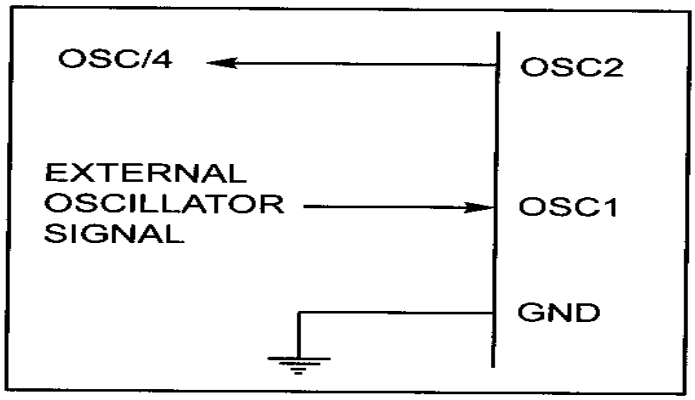
- C (Carry Flag)
  - Set when an addition or subtraction generates a carry out of B7 Bits (unsigned Nos.)
- DC (Digit Carry Flag) (Half Carry)
  - Set when carry generated from B3 to B4 in an BCD arithmetic operation (Addition or Subtraction)
- Z (Zero Flag)
  - It reflects the result of arithmetic or logic operations. If result =0, Z=1 and if result ≠0, Z=0
- OV (Overflow Flag)
  - Set when result of an operation of signed numbers goes beyond 7-bits- used to represent the errors { carry from B6 to B7 but no carry out of B7[C=0];Carry out from B7[C=1]}
- N (Negative Flag)
  - Binary representation of signed No. It is the reflection of the result of an arithmetic/logic Operation if Result B7=0 , No. is Positive, if Result B7=1 , No. is negative

Q 7. What are various oscillator options? How they can be selected using config register

Essentially, there are three clock sources for The PIC microcontroller to operate in different modes of operation:
Primary oscillators: The primary oscillators include the External Crystal and Resonator modes
Secondary oscillators: The secondary oscillators are those external sources not connected to the OSC1 or OSC2 pins.
Internal oscillator block: In addition to being a primary clock source, the internal oscillator block is available as a power-managed mode clock source.

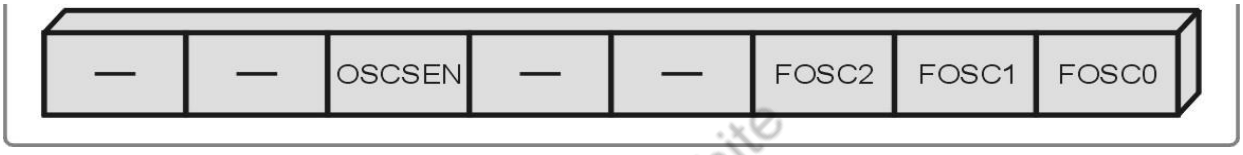


OSC1–OSC2 Connection to Crystal Oscillator



OSC Connection to an External Clock Source

CONFIG1H Register is used to set the clock oscillator for smooth function in PIC with internal or external crystal for low power consumption



bit 5 **$\overline{\text{OSCSEN}}$:** Oscillator System Clock Switch Enable bit

1 = Oscillator system clock switch option is disabled (main oscillator is source)

0 = Oscillator system clock switch option is enabled (oscillator switching is enabled)

bit 2-0 **FOSC2:FOSC0:** Oscillator Selection bits

111 = RC oscillator w/OSC2 configured as RA6

110 = HS oscillator with PLL enabled/clock frequency = (4 x Fosc)

101 = EC oscillator w/OSC2 configured as RA6

100 = EC oscillator w/OSC2 configured as divide-by-4 clock output

011 = RC oscillator

010 = HS oscillator

001 = XT oscillator

000 = LP oscillator

The Explanation of above different oscillator options is given in below table

LP	Low Power
XT	Crystal
HS	High Speed
RC	Resistor/Capacitor
EC, OSC2 as Clock Out	External Clock
EC, OSC2 as RA6	External Clock
HS-PLL Enabled	High Speed Phase Lock Loop
EC, OSC2 as RA6	External Clock

Q 8. Explain BOD and Power Down modes of PIC18Fxxx.

BOD/BOR

Mostly all microcontrollers have built in Brown-out detection (BOD) or Brown out reset(BOR) circuit, which monitors supply voltage level during operation. BOD circuit is nothing more than comparator, which compares supply voltage to a fixed trigger level.

CONFIG2L Register is used for This Purpose

Used to provide stable voltage and clock frequency during reset function, PWRT provides fixed delay for stabilizing the voltage source at required value.

When power source provided to Microcontroller fluctuates, it causes CPU malfunction. This BOR provision allows to set minimum voltage and CPU will go into reset state and stops all activities if supply voltage falls below set voltage.

—	—	VREGEN	BORV1(1)	BORV0(1)	BOREN1(2)	BOREN1(2)	$\overline{\text{PWRTEN}}$(2)
bit 7							bit 0

Brown-out Reset	
BOR = ON	Enabled
BOR = OFF	Disabled

Power-up Timer	
PWRT = ON	Enabled
PWRT = OFF	Disabled

Brown-out Voltage	
BORV = 45	4.5 V
BORV = 42	4.2 V
BORV = 27	2.7 V
BORV = 20	2.0 V

Power Down modes/ of power-managed modes of PIC18Fxxx

There are three categories of power-managed modes:

• Run modes

In the Run modes, clocks to both the core and peripherals are active

• Idle modes

The Idle modes allow the controller's CPU to be selectively shut down while the peripherals continue to operate. Selecting a particular idle mode allows users to further manage power consumption.

• Sleep mode

Sleep mode does not use a clock source

- Selecting a power-managed mode requires two decisions: if the CPU is to be clocked or not and the Selection of a clock source
- Switching from one power-managed mode to another begins by loading the OSCCON register.
- The SCS1:SCS0 bits select the clock source and determine which Run or Idle mode is to be used

---

Following Table explains how particular power down mode will be selected

Mode	OSCCON<7,1:0>		Module Clocking		Available Clock and Oscillator Source
	IDLEN(1)	SCS1:SCS0	CPU	Peripherals	
Sleep	0	N/A	Off	Off	None – all clocks are disabled
PRI_RUN	N/A	00	Clocked	Clocked	Primary – all oscillator modes. This is the normal full-power execution mode.
SEC_RUN	N/A	01	Clocked	Clocked	Secondary – Timer1 oscillator
RC_RUN	N/A	1x	Clocked	Clocked	Internal oscillator block(2)
PRI_IDLE	1	00	Off	Clocked	Primary – all oscillator modes
SEC_IDLE	1	01	Off	Clocked	Secondary – Timer1 oscillator
RC_IDLE	1	1x	Off	Clocked	Internal oscillator block(2)

Q 9. Give Details about Watch Dog Timers

Watch dog timer is Hardware used in PIC microcontroller .This Timer force Microcontroller into reset state when system is hang or out of control due to execution of incorrect code

CONFIG2H register is Used for Watch dog Timer

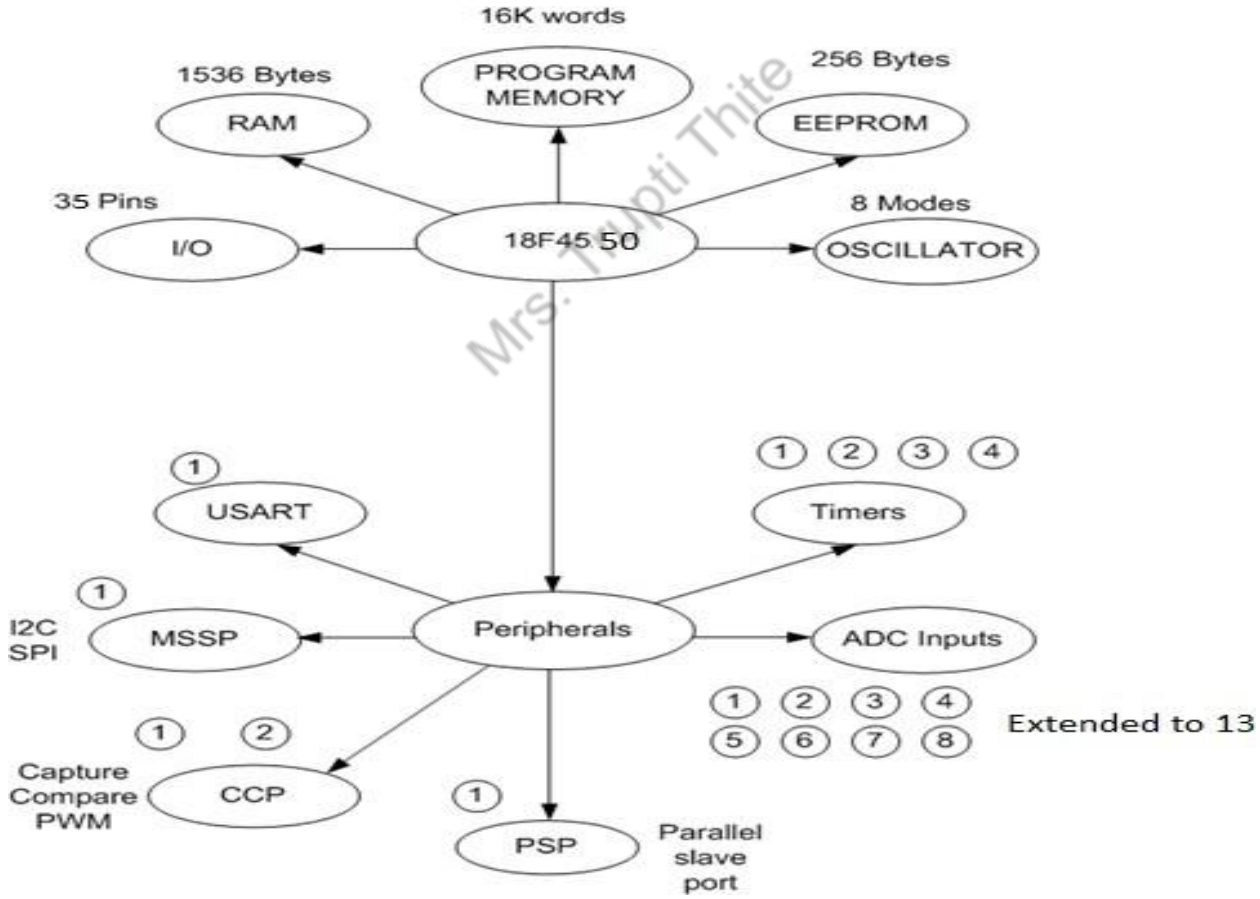
Watchdog Timer	
WDT = ON	Enabled
WDT = OFF	Disabled

This timer is used to prevent a system from going into infinite loop due to software bug

Q 9. Give Brief summary of Peripheral support

The PIC 18FXXXX has the following peripherals:

- Data ports:
  A (7-Bits)
  B, C and D (8-Bits)
  E (4- bits)
- Counter/Timer modules.
- Modules 0,2 (8-Bits)
  Modules 1,3 (16-Bits)
- CCP Modules.
- I2C/SPI serial port.
- USART port.
- ADC 10-bits 13 Channel
- EEPROM 256 Bytes



Q 10. PIC Instruction set (In question paper any instruction from below instruction can be asked

In Instruction Following notations are used

- F: File Register (or RAM)
- L: literal=8-bit constant
- W:Working Register

In Operand Following notations are used

- F: File Register (or RAM)
- a:Access Bank
- d:destination operand
- n: Memory address
- b:Bit

Move and Load Instructions

MOVLW	8-bit	;Load an 8-bit literal in WREG
MOVLW	0 x F2	
MOVWF	F, a	;Copy WREG in File (Data) Reg. ; If a = 0, F is in Access Bank ;If a = 1, Bank is specified by BSR
MOVWF	0x25, 0	;Copy W in Data Reg.25H
MOVFF	fs, fd	;Copy from one Data Reg. to ;another Data Reg.
MOVFF	0x20,0x30	;Copy Data Reg. 20 into Reg.30

Arithmetic Instructions

ADDWFC	F, d, a	;Add WREG to File Reg. with ;Carry and save result in W or F
SUBLW	8-bit	;Subtract WREG from literal
SUBWF	F, d, a	;Subtract WREG from File Reg.
SUBWFB	F, d, a	;Subtract WREG from File Reg. ;with Borrow
INCF	F, d, a	;Increment File Reg.
DECF	F, d, a	;Decrement File Reg.
COMF	F, d, a	;Complement File Reg.
NEGF	F, a	;Take 2's Complement-File Reg.

MULLW	8-bit	;Multiply 8-bit and WREG ;Save result in PRODH-PRODL
MULWF	F, a	;Multiply WREG and File Reg. ;Save result in PRODH-PRODL
DAW		;Decimal adjust WREG for BCD ;Operations

Logic Instructions

ANDLW	8-bit	;AND literal with WREG
ANDWF	F, d, a	;AND WREG with File Reg. and ;save result in WREG/ File Reg.
IORLW	8-bit	;Inclusive OR literal with WREG
IORWF	F, d, a	;Inclusive OR WREG with File Reg. ;and save result in WREG/File Reg.
XORLW	8-bit	;Exclusive OR literal with WREG
XORWF	F, d, a	;Exclusive OR WREG with File Reg. ;and save result in WREG/File Reg.

Branch Instructions

- BC n ;Branch if C flag = 1 within + or – 64 Words
- BNC n ;Branch if C flag = 0 within + or – 64 Words
- BZ n ;Branch if Z flag = 1 within + or – 64 Words
- BNZ n ;Branch if Z flag = 0 within + or – 64 Words
- BN n ;Branch if N flag = 1 within + or – 64 Words
- BNN n ;Branch if N flag = 0 within + or – 64 Words
- BOV n ;Branch if OV flag = 1 within + or – 64 Words
- BNOV n ;Branch if OV flag = 0 within + or – 64 Words
- GOTO Address: Branch to 20-bit address unconditionally

Call and Return Instructions

- RCALL nn ;Call subroutine within +or – 512 words
- CALL 20-bit, s ;Call subroutine
  ;If s = 1, save W, STATUS, and BSR
- RETURN, s ;Return subroutine
- ;If s = 1, retrieve W, STATUS, and BSR
- RETFIE, s ;Return from interrupt
- ;If s = 1, retrieve W, STATUS, and BSR

Bit Manipulation Instructions

- BCF F, b, a ;Clear bit b of file register. b = 0 to 7
- BSF F, b, a ;Set bit b of file register. b = 0 to 7
- BTG F, b, a ;Toggle bit b of file register. b = 0 to 7

- RLCF F, d, a ;Rotate bits left in file register through
  ; carry and save in W or F register
- RLNCF F, d, a ;Rotate bits left in file register
  ; and save in W or F register
- RRCF F, d, a ;Rotate bits right in file register through
- ; carry and save in W or F register
- RRNCF F, d, a ;Rotate bits right in file register
  ; and save in W or F register

Test and Skip Instructions

- BTFSC F, b, a ;Test bit b in file register and skip the
  ;next instruction if bit is cleared (b =0)
- BTFSS F, b, a ;Test bit b in file register and skip the
  ;next instruction if bit is set (b =1)
- CPFSEQ F, a ;Compare F with W, skip if F = W
- CPFSGT F, a ;Compare F with W, skip if F > W
- CPFSLT F, a ;Compare F with W, skip if F < W
- TSTFSZ F, a ;Test F; skip if F = 0

Increment/Decrement and Skip Next Instruction

- DECFSZ F, b, a ;Decrement file register and skip the ;next instruction if F = 0
- DECFSNZ F, b, a ;Decrement file register and skip the ;next instruction if F ≠ 0
- INCFSZ F, b, a ;Increment file register and skip the ;next instruction if F = 0
- INCFSNZ F, b, a ;Increment file register and skip the ;next instruction if F ≠ 0

Machine Control Instructions

- CLRWDT ;Clear Watchdog Timer
  - Helps recover from software malfunction
  - Uses its own free-running on-chip RC oscillator
  - WDT is cleared by CLRWDT instruction
- RESET ;Reset all registers and flags
  - When voltage < a particular threshold, the device is held in reset
  - Prevents erratic or unexpected operation
- SLEEP ;Go into standby mode
- NOP ;No operation