INTRODUCTION TO PIC MICROCONTROLLERS

Brief Introduction

- The term **PIC** stands for “Peripheral Interface Controller” .
- These 8-bit micro controllers have become very important now -a -days in industrial automation and embedded applications etc.

Brief History

- PIC MCU was developed by General Instruments in 1975. (Microelectronics division) (CP1600).
- In 1985, General Instruments converted their Microelectronics division to Microchip technology. (PIC1640 and PIC1650).
- In 1993, Microchip technology launched the 8-bit PIC 16C84 with EEPROM.
- In 1998, Improved version of PIC 16C84 with flash memory was evolved (PIC 16F84)

Some of the low-end device numbers are

- 12C5XX
- 16C5X
- 16C505

Mid range PIC architectures

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

Classification of PIC

The 8-bit PIC microcontroller is divided into following four categories on the basis of internal architecture:

1. Base Line PIC
2. Mid-Range PIC
3. Enhanced Mid-Range PIC
4. PIC18

Comparison for above four categories

	Base Line	Mid-Range	Enhanced Mid-Range	PIC18
No. of Pins	6-40	8-64	8-64	18-100
Program Memory	Up to 3 KB	Up to 14 KB	Up to 28 KB	Up to 128 KB
Data Memory	Up to 134 Bytes	Up to 368 Bytes	Up to 1.5 KB	Up to 4 KB
Instruction Length	12-bit	14-bit	14-bit	16-bit
No. of instruction set	33	35	49	83
Speed	5 MIPS*	5 MIPS	8 MIPS	Up to 16 MIPS
Feature	• Comparator • 8-bit ADC • Data Memory •Internal Oscillator	In addition of baseline · SPI · I^2C · UART · PWM · 10-bit ADC · OP-Amps	In addition of Mid-range · High Performance · Multiple communication peripherals	In addition of Enhanced Mid-range • CAN • LIN • USB • Ethernet • 12-bit ADC
Families	PIC10,PIC12, PIC16	PIC12, PIC16	PIC12F1XXX, PIC16F1XXX	PIC18

Major Features

All the major features of mid range PIC are divided into different categories:

- Core Features
- Architectural Features
- Peripheral Features

Core Features

Core pertains to the basic features that are required to make the device operate

- Device Oscillator (Crystal, RC)
- Reset logic (Master Clear, POR, BOR, WDT)
- CPU Operation (W, Status, PCL, PCLATH, FSR, INDF)
- ALU operation (8)
- Device Memory Map Organization
- Interrupt Logic
- Instruction set (35)
- Low Power Consumption (< 2mA @5V), 4-20MHz.

Device Oscillator

The internal oscillator circuit is used to generate the device clock. The device clock is required for the device to execute instructions and for the peripherals to function. Four device clock periods generate one internal instruction clock (TCY) cycle.

There are up to eight different modes which the oscillator may have :

- LP Low Frequency (Power) Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC External Resistor/Capacitor (same as EXTRC with CLKOUT)
- EXTRC External Resistor/Capacitor
- EXTRC External Resistor/Capacitor with CLKOUT
- INTRC Internal 4 MHz Resistor/Capacitor
- INTRC Internal 4 MHz Resistor/Capacitor with CLKOUT

Clock Types

RC oscillator

- Least expensive
- Can be used for non-critical frequency accuracy and stability
- Some devices have internal RC oscillator at 4 MHz

Crystal oscillator

- Most stable

External clock

- Provided by external digital system

Specific modes

- LP mode — frequencies between 32 kHz and 200 kHz
- XT mode — frequencies between 100 kHz and 4 MHz
- HS mode — frequencies between 8 MHz and 20 MHz

PIC Oscillator Connections

- There are 4 common oscillator modes that are available on most PIC micro devices.
- HS, XT, LP and RC. (High speed, Crystal, Low-power clocking, Internal RC)
- The HS mode stands for “High Speed” mode. It is designed to be used with crystals, and resonators with a frequency of 3 to 4 MHz or more. The important thing to remember about HS mode is that it provides the highest drive level available.

- The XT mode stands for “Crystal” mode and will produce a medium drive level. It is designed to be used with crystals and resonators of 1 to about 4 MHz.
- XT mode has moderate power consumption since its drive level is lower than that of HS mode, and because a lower clock speed is produced. Remember, as a rule: the faster the clock used, the more current the application will require

- The LP mode stands for "Low Power" mode.

  This mode is useful for circuits that require the lowest power possible. LP mode is engineered for 32.768kHz crystal operation, and it can function at any frequency below 200kHz. LP mode is most commonly used for 32.768kHz operation.

  LP mode will produce the slowest clock rate, and as a result, the lowest power consumption of all the modes.
- LP mode is ideal for timing sensitive applications since these same crystals are used as a time base in wrist-watches.

- The RC mode stands for “External RC” mode.
- It is important to note that this is an _EXTERNAL_ RC mode, (some PICmicro devices have an _INTERNAL_ RC mode).
- RC mode uses a resistor-capacitor network connected to the OSC1 pin. When the device is configured for external RC mode, these components are automatically driven to produce a frequency which will run the PICmicro MCU.

- RC mode is designed for very low-cost applications. The power consumed will vary due to the wide range of frequencies that can be created using this mode

- It is extremely important to note that RC mode will produce an inaccurate clock source. In some applications this won't matter, but applications that are timing sensitive should not be used with this mode. For this reason, RC mode is not recommended for timing sensitive applications or for RS-232 communication.

- The IntRC mode stands for “Internal RC” mode and functions much like the standard External RC mode.
- Unlike External RC mode, in IntRC mode the resistor and capacitor are already provided. Microcontrollers with this feature have an on-chip RC oscillator. Current designs run at approximately 4 MHz.

- IntRC mode is the least expensive oscillator available since no external components are needed. It is also useful, because devices in this mode can often use the OSC1 and OSC2 lines for general purpose I/O. This feature makes the PIC12CXXX and other 8-pin PICmicro devices very popular.

Reset logic

The reset logic is used to place the device into a known state.

- Power-on Reset (POR)
- MCLR reset during normal operation
- MCLR reset during SLEEP
- WDT reset during normal operation
- Brown-out Reset (BOR)

PIC Reset actions

- Reset is used to put the microcontroller into a known state. Normally when a PIC microcontroller is reset, execution starts from address 0 of the program memory. This is where the first executable user program resides. The reset action also initializes various SFR registers inside the Microcontroller.

- Reset mechanisms ensure that the CPU starts running when the appropriate operating conditions have been met, and can be used to restart the CPU in case of program failure.

- The actual reset to the CPU or the Chip_Reset, is generated by a flip-flop. This has two inputs, S (Set) and R (Reset).

- The CPU enters Reset mode when Chip_Reset goes low, which is caused by the S line going high. It stays there until the flip-flop is cleared, caused by the R line going high.

- The S input to the flip-flop goes high, via a three-input OR gate, if any of the following goes high:

- The Reset action normally occurs in various situations like
- Power On Reset
- Master Clear(MCLR’) Reset
- Watchdog Timer Reset
- Brown-out Reset (This feature is not available in PIC16C61/62/64/65 series. Only 74 series have this feature

- The reset action will set the program counterto the starting address of the program.This starting address is different for different members.For Ex:Starting address for PIC16C71 is 000H.whereas for PIC16C57 it is 7FFH.
- So,The first instruction to be executed will appear at the reset vector.

- The Brown-out reset occurs when the supply voltage falls below 4 volts.The device remains in the Brown-out reset state till the supply voltage is restored.The Brown-out reset also includes another feature .If Vdd is hit with a transient noise spike ,causing Vdd to drop below 4.0Volts for longer than 100 micro seconds the Brown-out reset circuit detects that and reset the chip.whereas in most of the other microcontrollers many erroneous executions takes place during this time. The PIC brown-out reset will also help if the power-on reset conditions are not met.

CPU Operation

The Central Processing Unit (CPU) is responsible for using the information in the program memory (instructions) to control the operation of the device. Many of these instructions operate on data memory. To operate on data memory, the Arithmetic Logical Unit (ALU) is required. In addition to performing arithmetical and logical operations, the ALU controls status bits (which are found in the STATUS register). The result of some instructions force status bits to a value depending on the state of the result.

The CPU controls the program memory address bus, the data memory address bus, and accesses to the stack.

CPU REGISTERS

The CPU registers are

- Working Register (W)
- Status – Register
- FSR – File Select Register
- INDF
- PCLATH
- Program Counter
- PCL
- Eight Level Stack

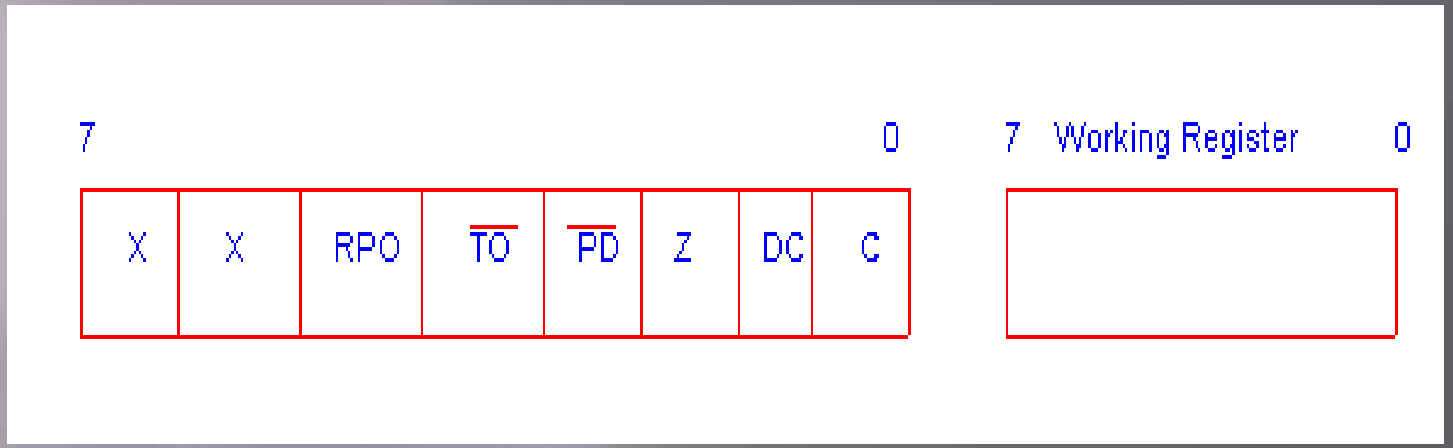
CPU Registers

- Working Register:

  Working Register is used by many instructions as the source of an operand. It also serves as the destination for the result of instruction execution and it is similar to accumulator in other controllers and processors. It is an 8-bit register.

- Status Register:

  It contains the arithmetic status of the ALU, the RESET status and the bank select bits for the data memory.



C: Carry/borrow bit
DC: Digit carry/borrow bit
Z: Zero bit
NOT_PD: Reset Status bit (Power-down mode bit)
NOT_TO: Reset Status bit (time- out bit)
RPO: Register bank Select
The bits 7 and 6 of Status Register are unused by 16c6x/7x.

The ‘C’ bit is set when two 8-bit operands are added together and a 9-bit result occurs. This 9-bit is placed in the carry bit.

- The DC or Digit carry bit indicates that a carry from the lower 4 bits occurred during an 8-bit addition.

- Example: 0011 1000
  0011 1000
  0111 0000

Here DC=1 as a result of the carry from the bit 3 to the bit 4 position.

- The Z or zero bits is affected by the execution of arithmetic or logic instructions.

- The reset status bits NOT_TO and NOT_PD are used in conjunction with PIC's sleep mode. The micro controller can put itself to sleep mode to save power during intervals when it has nothing to do. It can be reset by any of three kinds. Upon reset the CPU can check these two reset status bits to determine which kind of event resettled it and then respond accordingly.

- The Register bank select bit RPO is used to select either bank .

  When RPO=0, select Bank 0,

  RPO=1, select Bank 1.

- Example: bcf STATUS, RPO

  Select bank 0

- bsf STATUS, RPO

  Select bank 1.

- FSR – (File Select Register):

  It is the pointer used for indirect addressing. In the indirect addressing mode the 8-bit register file address is first written into FSR. It is a special purpose register that serves as an address pointer to any address through out the entire register file.

- INDF – (Indirect File):

  It is not a physical register .Addressing this INDF will cause indirect addressing. Any instruction using the INDF register actually access the register pointed to by the FSR.

- PCL:

  PCL is actually the lower 8-bits of the 13-bit program counter. It can be read like any other register.

- PCLATH (Program Counter Latch):

  The upper 3-bits of PCLATH remains zero and serves no purpose, it is only when PC2 is written to that PCLATH is automatically written into the PC at the same time.

Working Register:(W)

- **Temporary holding** register.
- Called as **Accumulator.**
- Cannot be access **Directly**
- **W** content moved to some other register and then it can be accessed.
- used by many instructions as **source of an operand**.
- **destination for the result** of instruction execution.
- It is an **8-bit** register

ALU Operation

ALU sources

Special **WORKING** register W

Data register or immediate

ALU destination

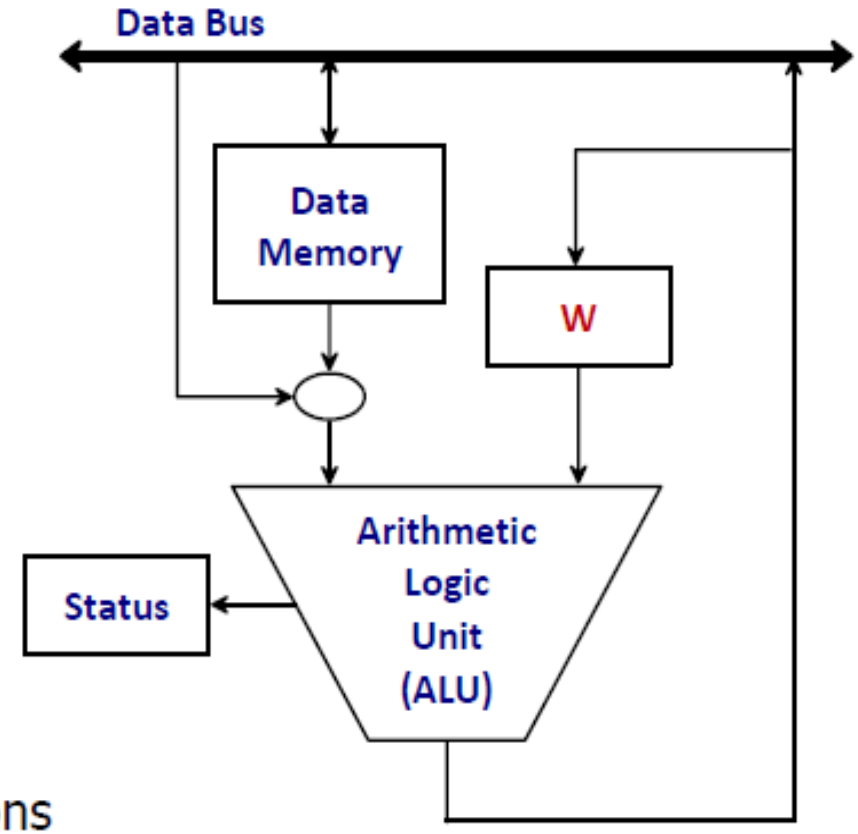
Data register or W

Transfer operations

Data register ↔ W

Status register

Flags produced by ALU operations



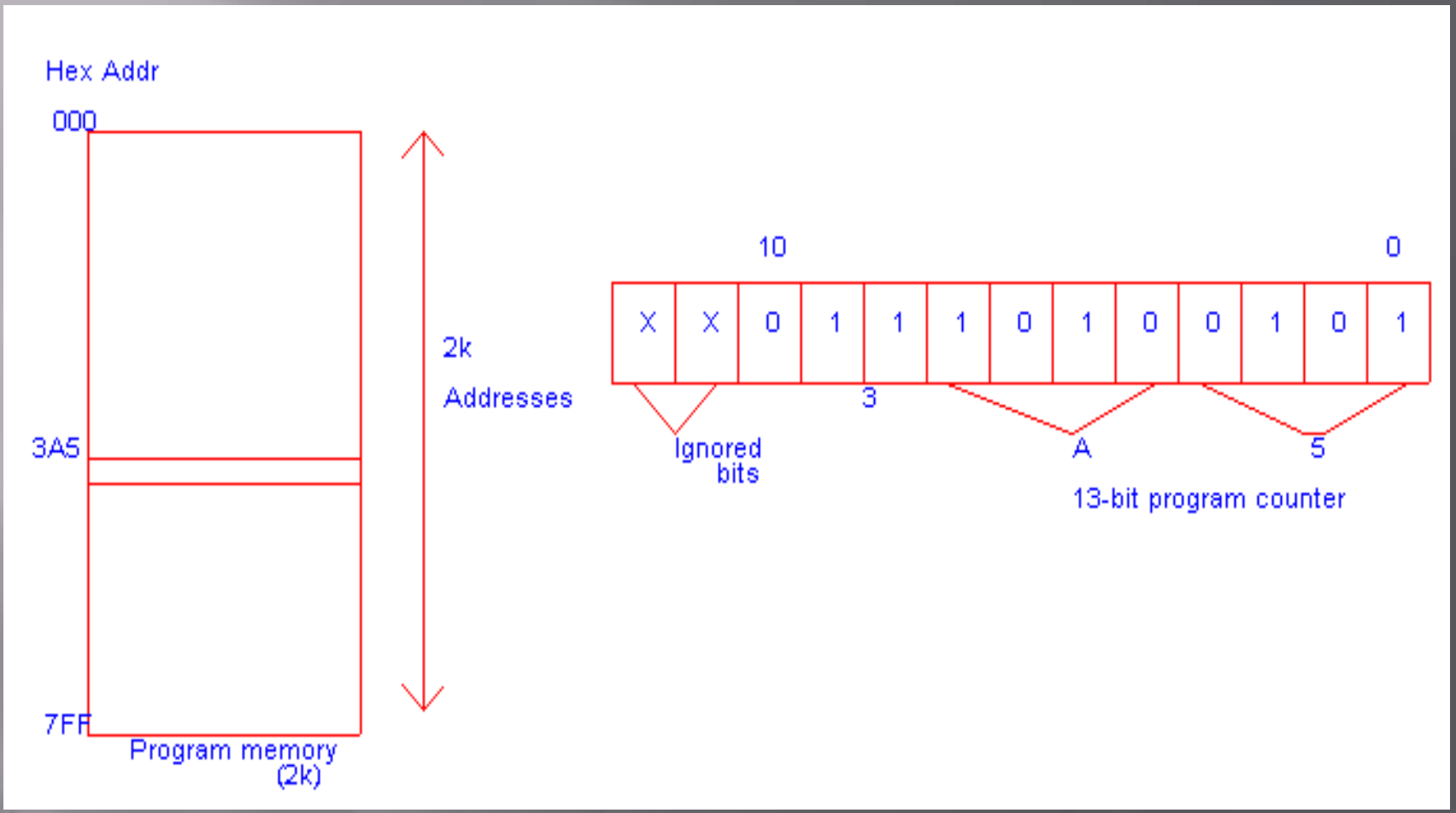
Memory organisation :

It has three memory blocks.

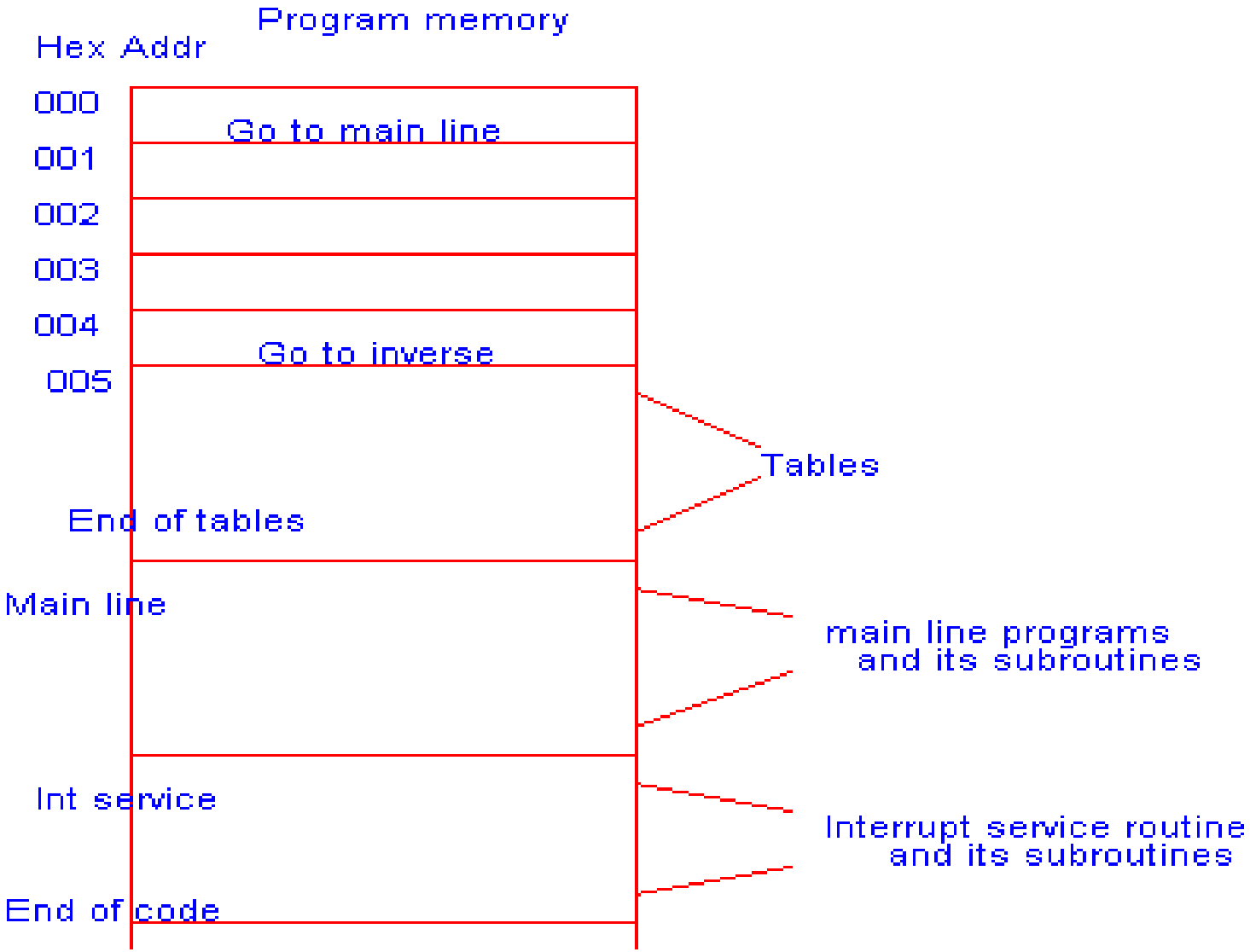
- Program memory
- Data memory
- Stack

Program Memory

- The 6x/7x family controllers have either 2k or 4k address of program memory. Normally a program memory of 2k addresses needs only a 11-bit program counter to access any address (211=2048=2k).
- A program memory of 4k address needs a 12-bit program counter. But this PIC family uses 13-bit program counter allowing the controllers to an 8k-program memory without changing the CPU structure.


Hex Addr
000
3A5
7FF
Program memory
(2k)
2k
Addresses
10
0
X X 0 1 1 1 0 1 0 0 1 0 1
3
Ignored
bits
A
5
13-bit program counter

- Two addresses in the program memory address space are treated in a special way by the CPU. The first address H’ 000’ being a go to mainline instruction the second special address, H’ 004’ being a ‘go to in service’ instruction can be assigned to this address to make the CPU to jump to the beginning of the Interrupt Service routine located elsewhere in the memory space.


Program memory
Hex Addr
000
Go to main line
001
002
003
004
Go to inverse
005
Tables
End of tables
Main line
main line programs
and its subroutines
Int service
Interrupt service routine
and its subroutines
End of code

- When we deal with tables, if any tables are created they are assigned to addresses in the range H’005 – H’0FF’. For most of the applications this space is sufficient.

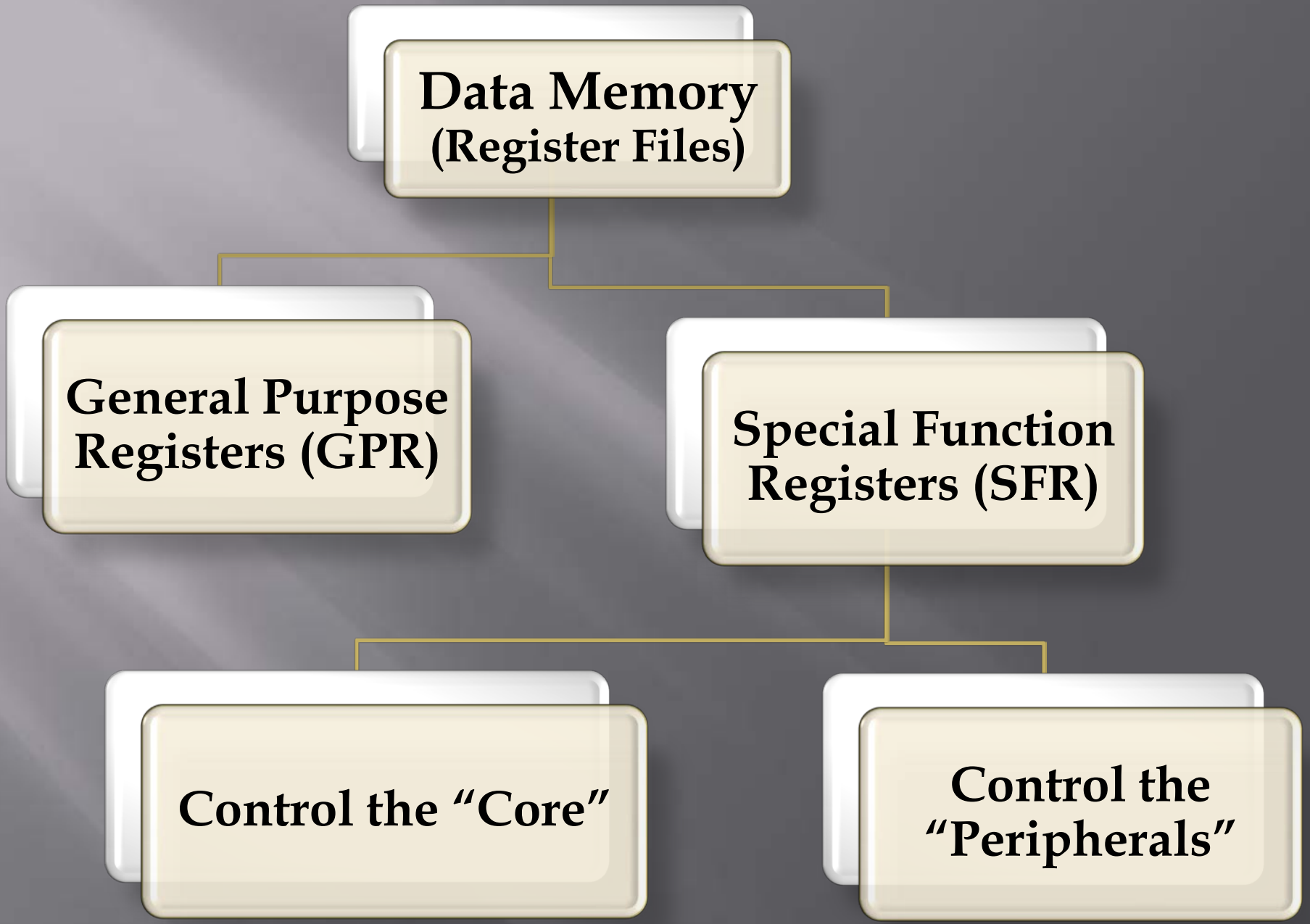
The main line program begins after the tables.

Data memory (Register Files):

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

- General purpose register file (RAM).
- 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 bank


Data Memory
(Register Files)
General Purpose
Registers (GPR)
Special Function
Registers (SFR)
Control the "Core"
Control the
"Peripherals"

Architectural Features

These features gives information regarding internal details and basic functionality of a controller:

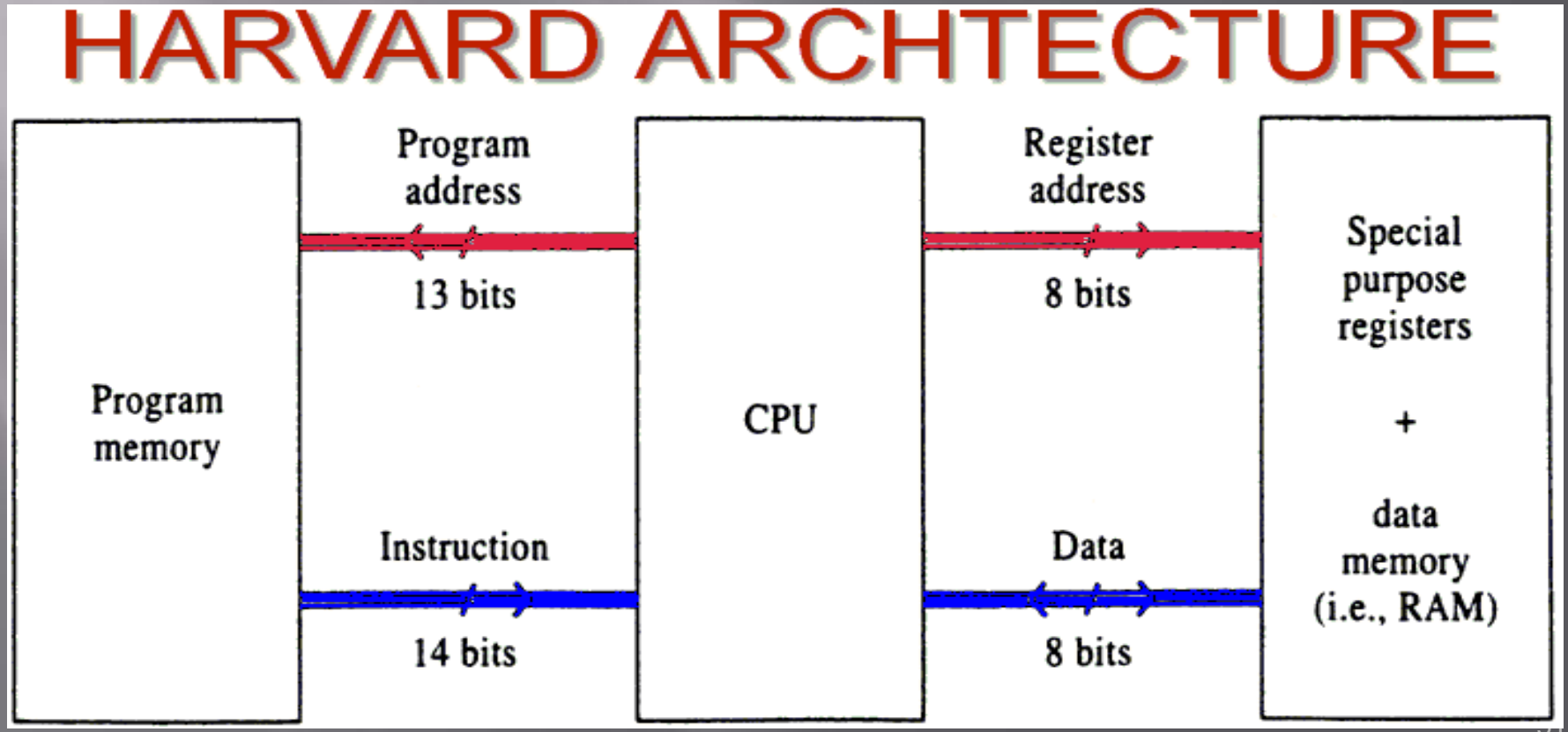
- Harvard Architecture
- Long word Instructions
- Single word Instructions
- Single Cycle Instructions
- Instructions Pipelining
- Reduced Instruction set (35)
- Register File Architecture
- Orthogonal Instruction set (symmetric)
- Eight-level deep hardware stack

ARCHITECTURE

- The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC 16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional Von Neumann architecture where program and data may be fetched from the same memory using the same bus.

PIC architecture is a "Harvard" architecture

- The Harvard Architecture



HARVARD ARCHITECTURE

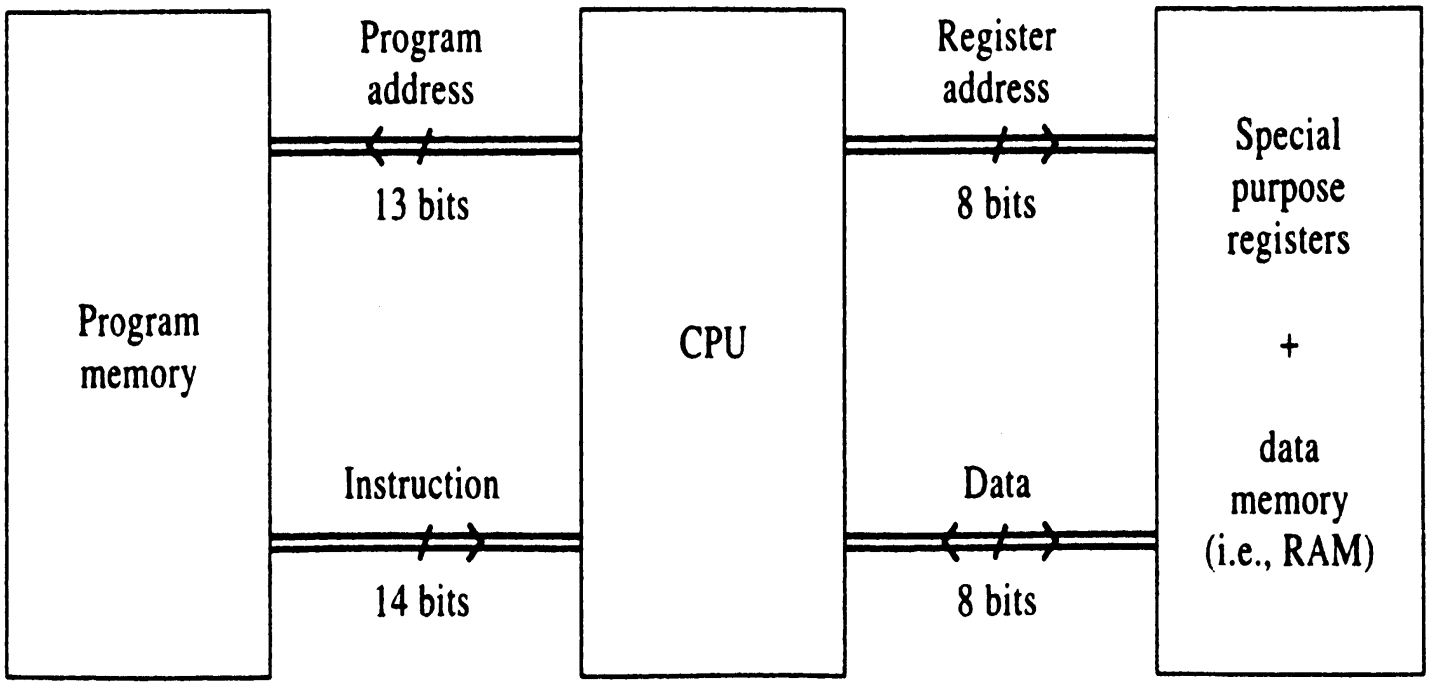


Figure 1 Harvard architecture.

- As the PIC 16c6x/7x family of micro-controllers uses Harvard Architecture it enables the devices exceptionally fast execution speed for a given clock rate. In the Harvard Architecture separate buses are used for Data and Instruction as shown in the diagram.

- Instructions are fetched from program memory using buses that are distinct from the buses used for accessing variables in data memory, I/O ports etc. Every instruction is coded as a single 14-bit word and fetched over a 14-bit wide bus.

- Separating program and data buses further allows instructions to be sized differently than 8-bit wide data words. Instruction op-codes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle

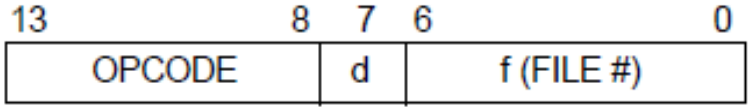
Specifications and Features:

- The PIC 16C61 addresses 1K x 14 of program memory.
- The PIC16C62/62A/R62/64 addresses 2K x 14 of program memory, and
- the PIC16C63/R63/65/65/65A/R65 devices address 4K x 14 of program memory.
- The PIC 16C66/67 address 8K x 14 program memory. All program memory is internal.
- The PIC16CXX can directly or indirectly address its register files or data memory.
- All special function registers including the program counter are mapped in the data memory.

Long Word Instructions (14 Bit)

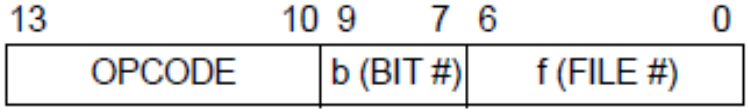
Figure 5-1: General Format for Instructions

Byte-oriented file register operations



d = 0 for destination W
d = 1 for destination f
f = 7-bit file register address

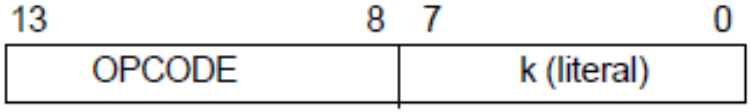
Bit-oriented file register operations



b = 3-bit bit address
f = 7-bit file register address

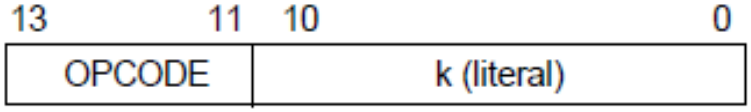
Literal and control operations

General



k = 8-bit immediate value

`CALL` and `GOTO` instructions only

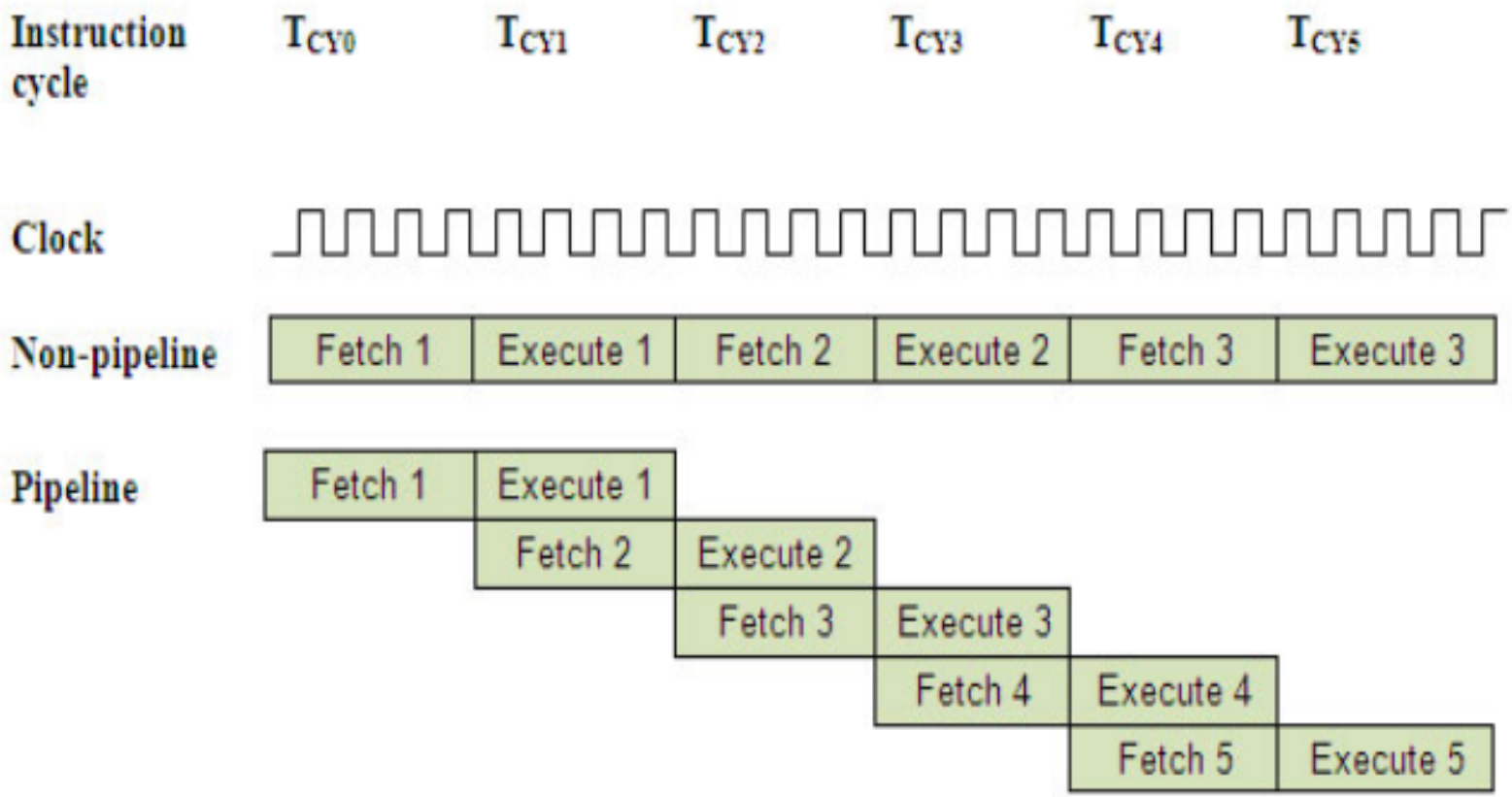


k = 11-bit immediate value

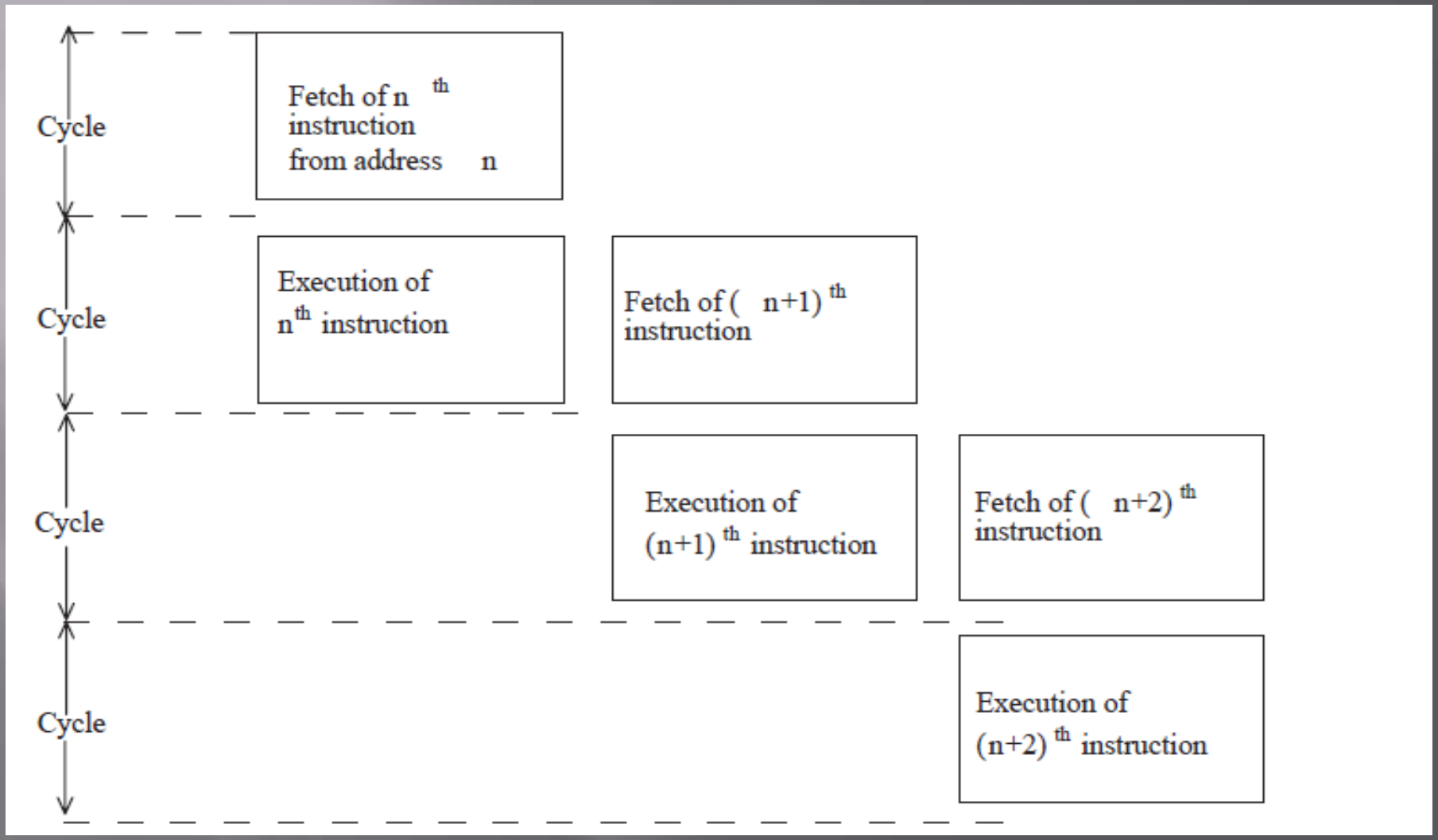
Single word/cycle Instructions

Mnemonics	Description	Instruction Cycles
bcf f, b	Clear bit b of register f	1
bsf f, b	Set bit b of register f	1
clrw	Clear working register W	1
clrf f	Clear f	1
movlw k	Move literal 'k' to W	1
movwf f	Move W to f	1
movf f, F(W)	Move f to F or W	1
swapf f, F(W)	Swap nibbles of f, putting result in F or W	1
andlw k	And literal value into W	1
andwf f, F(W)	And W with F and put the result in W or F	1
andwf f, F(W)	And W with F and put the result in W or F	1
iorlw k	inclusive-OR literal value into W	1
iorwf f, F(W)	inclusive-OR W with f and put the result in F or W	1
xorlw k	Exclusive-OR literal value into W	1
xorwf f, F(W)	Exclusive-OR W with f and put the result in F or W	1
addlw k	Add the literal value to W and store the result in W	1
addwf f, F(W)	Add W to f and store the result in F or W	1
sublw k	Subtract the literal value from W and store the result in W	1
subwf f, F(W)	Subtract f from W and store the result in F or W	1
rlf f, F(W)	Copy f into F or W; rotate F or W left through the carry bit	1
rrf f, F(W)	Copy f into F or W; rotate F or W right through the carry bit	1
btfsc f, b	Test 'b' bit of the register f and skip the next instruction if bit is clear	1 / 2
btfss f, b	Test 'b' bit of the register f and skip the next instruction if bit is set	1 / 2
decfsz f, F(W)	Decrement f and copy the result to F or W; skip the next instruction if the result is zero	1 / 2
incfsz f, F(W)	Increment f and copy the result to F or W; skip the next instruction if the result is zero	1 / 2
goto label	Go to the instruction with the label "label"	2
call label	Go to the subroutine "label", push the Program Counter in the stack	2
return	Return from the subroutine, POP the Program Counter from the stack	2
retlw k	Return from the subroutine, POP the Program Counter from the stack; put k in W	2
retie	Return from Interrupt Service Routine and re-enable interrupt	2
clrwdt	Clear Watch Dog Timer	1
sleep	Go into sleep/ stand by mode	1
nop	No operation	1

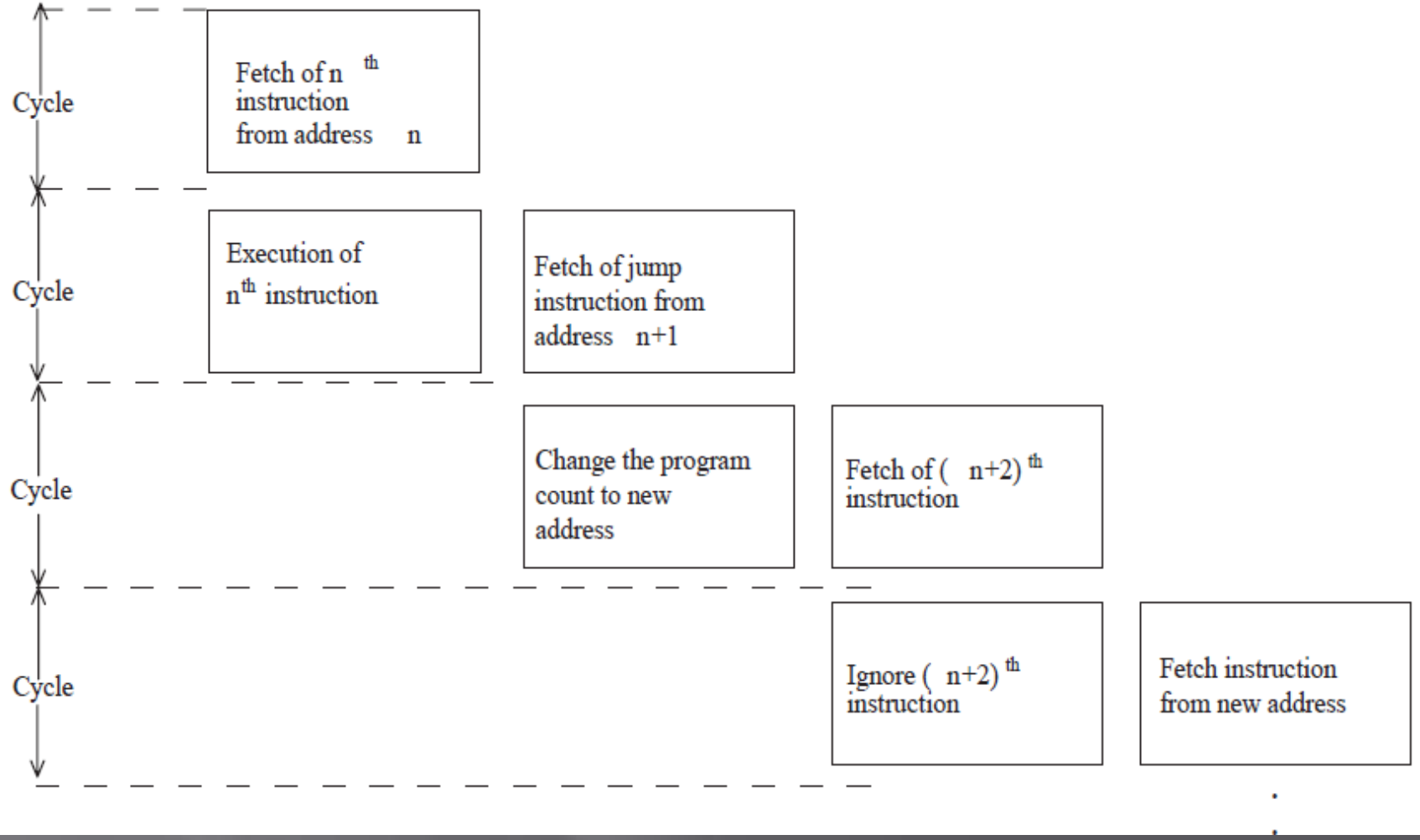
Instruction Pipelining (Sequential)



Instruction Pipelining (Sequential)



Instruction Pipelining (Branch Control)



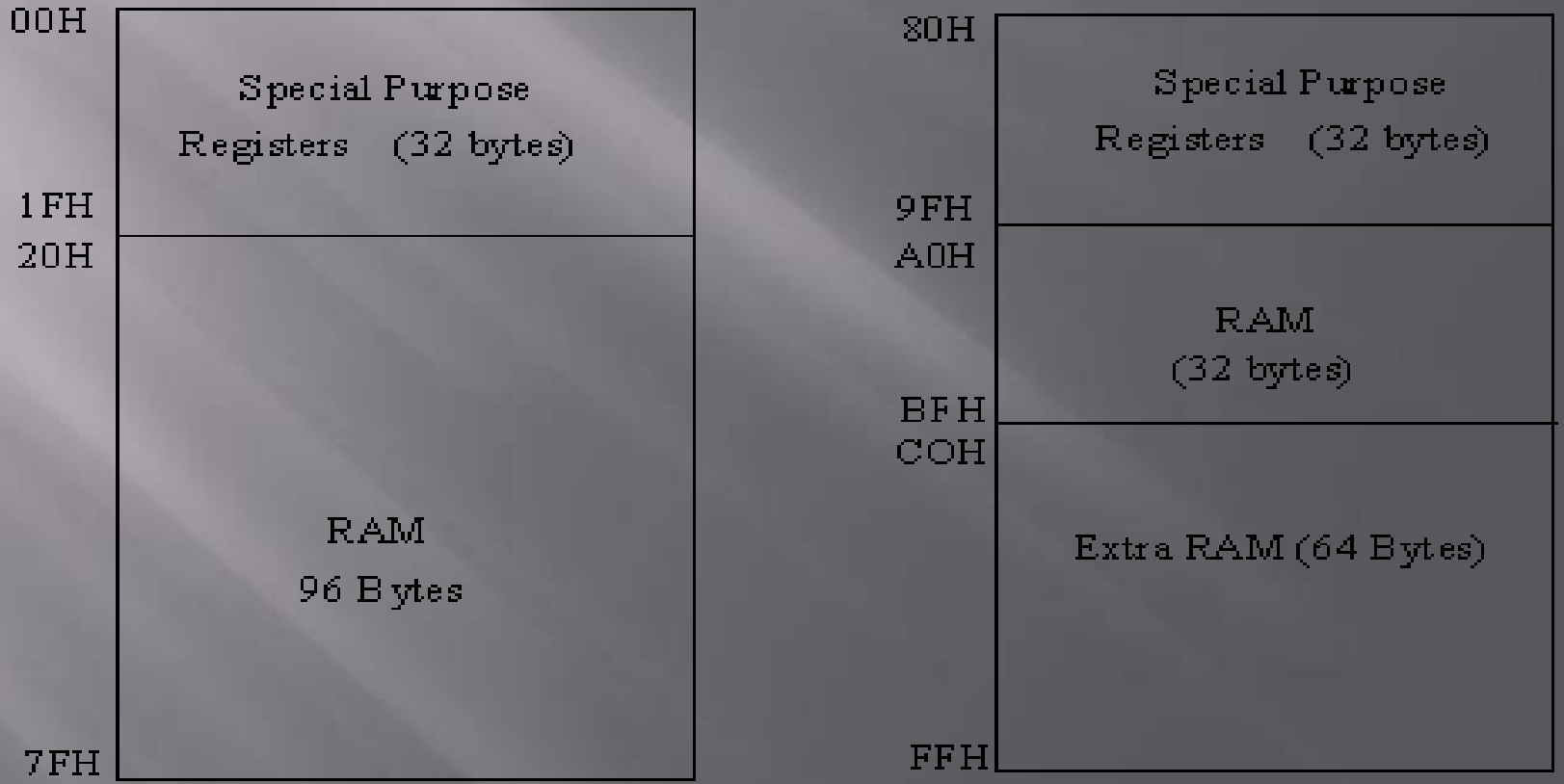
Register File Architecture:

- Data Memory is also known as Register File.

Register File consists of two components.

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

Data Memory Map



Register File Map

	Bank 0	Bank 1	
00	INDF	INDF	80
01	TMRO	OPTION	81
02	PCL	PCL	82
03	STATUS	STATUS	83
04	FSR	FSR	84
05	PORTA	TRISA	85
06	PORTB	TRISB	86
07	PORTC	TRISC	87
08	PORTD	TRISD	88
09	PORTE	TRISE	89
0A	PCLATH	PCLATH	8A
0B	INTCON	INTCON	8B
0C . . . 1F			9F
07F			0FF

Peripheral Features

These features add power to microprocessor or microcontroller which includes ease in interfacing to external world and internal tasks:

- General purpose I/O
- T0, T1,T2
- CCP- Capture, Compare and PWM
- SSP – Synchronous serial port
- USART- Serial Communication Interface
- Comparators
- ADCs- 8 or 10 bits
- LCD Drivers

PIC 16C6X/7X CONTROLLERS

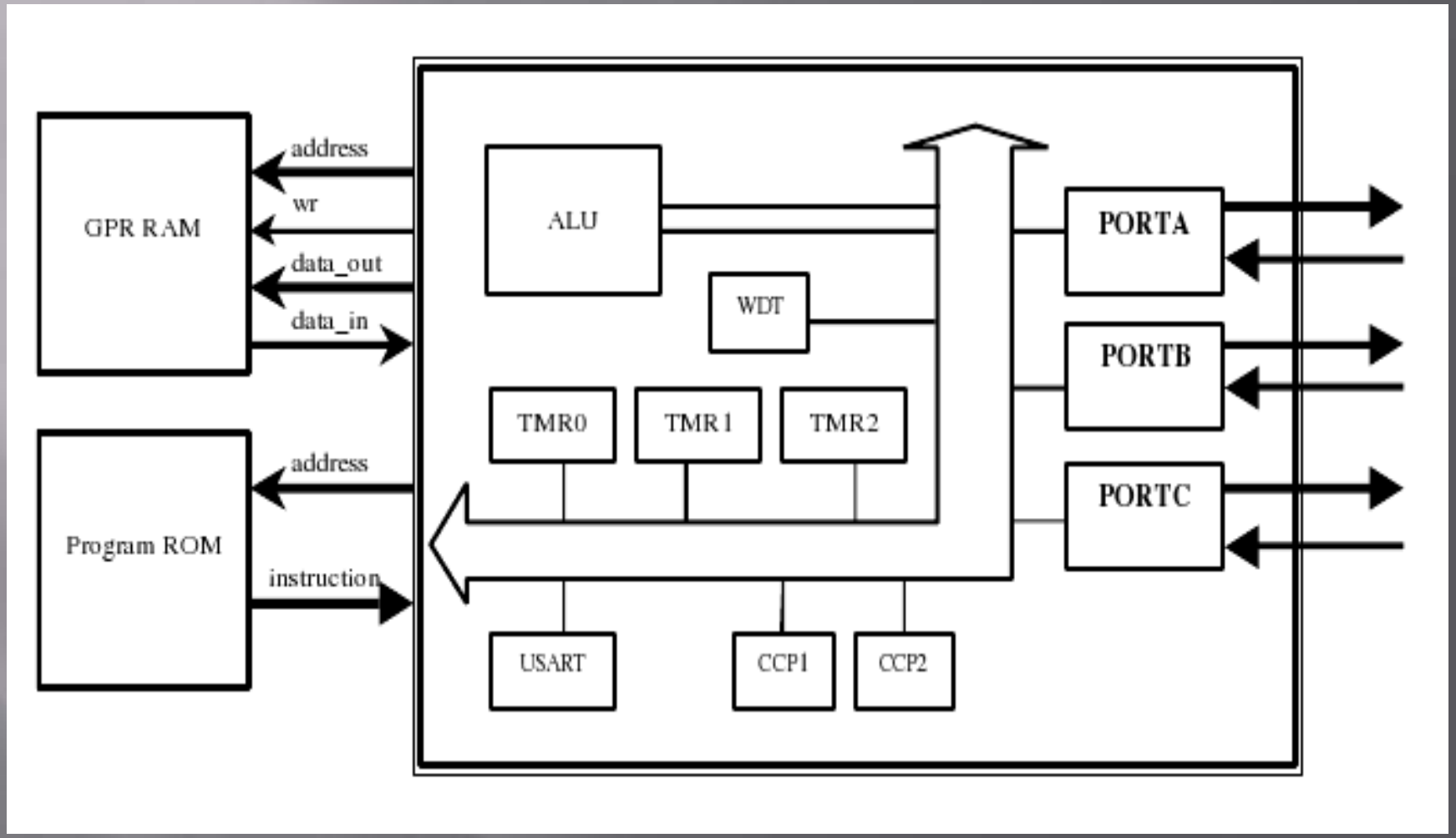
PERIPHERAL FEATURES

- The PIC16CXX/17CXX is a family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers.
- There are Three Timers : Timer 0, Timer 1, Timer 2
- Timer 0: 8-bit timer/counter with 8-bit prescaler
- Timer 1: 16-bit timer/counter with prescaler can be incremented during sleep via external crystal/clock
- Timer 2: 8-bit timer/counter with 8-bit period register, pre-scaler and post-scaler

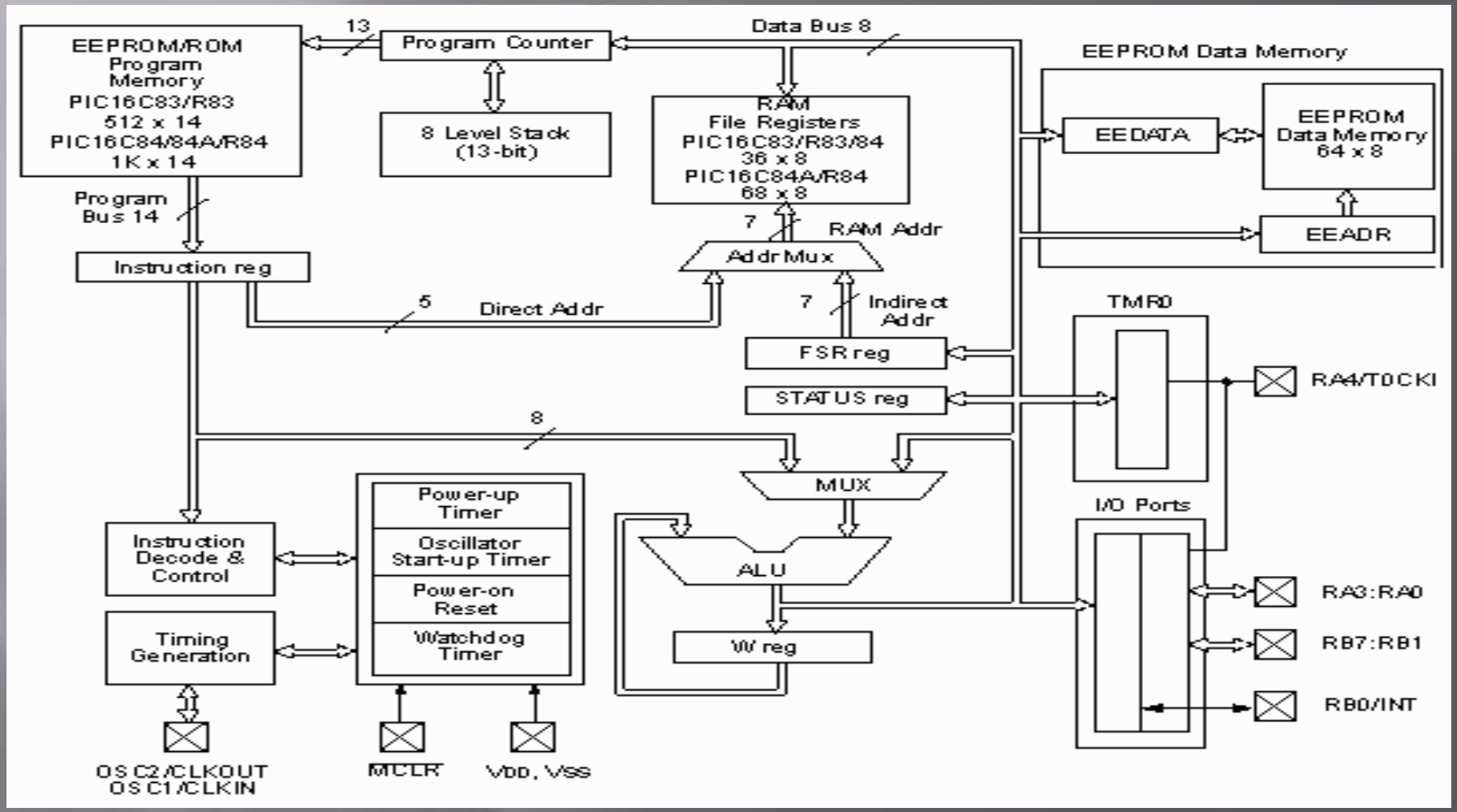
- Capture/Compare/PWM (CCP) module(s)
- Capture is 16-bit, max resolution is 12.5 ns, Compare is 16-bit, max resolution is 200ns, PWM max resolution is 10-bit
- Synchronous Serial Port (SSP) with SPI™ and I2C
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI)

- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls
- Brown-out detection circuitry for Brown-out Reset (BOR)
- PIC16CXX microcontroller family has enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources.

Block Diagram-16C6X



General Block Diagram



Typical Applications

- Motor control speed
- D-to-A Conversion
- Stepper motor, micro-step controller
- Timing control and event sequencing
- Magnetic/pneumatic actuator intelligent control
- LED/Lamp intelligent driver
- PWM of AC power
- Switching power supply controller

Limitations

The PIC architectures have several limitations:

- Only a single accumulator
- A small instruction set
- 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