

Harvard architecture refers to a memory structure where the processor is connected to 2 different memory banks via two sets of buses. This is to provide the processor with 2 distinct data paths, one for instruction & one for data. Through this scheme, the CPU can read both an instruction & data from the respective memory banks at the same time.

Microchip PIC16F877 Microcontroller

1. Introduction

This tutorial is to introduce the microcontroller technology the capabilities and the specifications of a commonly used Microcontroller Microchip PIC16F877 and describe the experiments conducted using the Development board Flash PIC development board which accommodates this microcontroller.

What is a microcontroller?

A microcontroller is a compact standalone computer, optimized for control applications. Entire processor, memory and the I/O interfaces are located on a single piece of silicon so, it takes less time to read and write to external devices.

Why are microcontrollers used?

Following are the reasons why microcontrollers are incorporated in control systems:

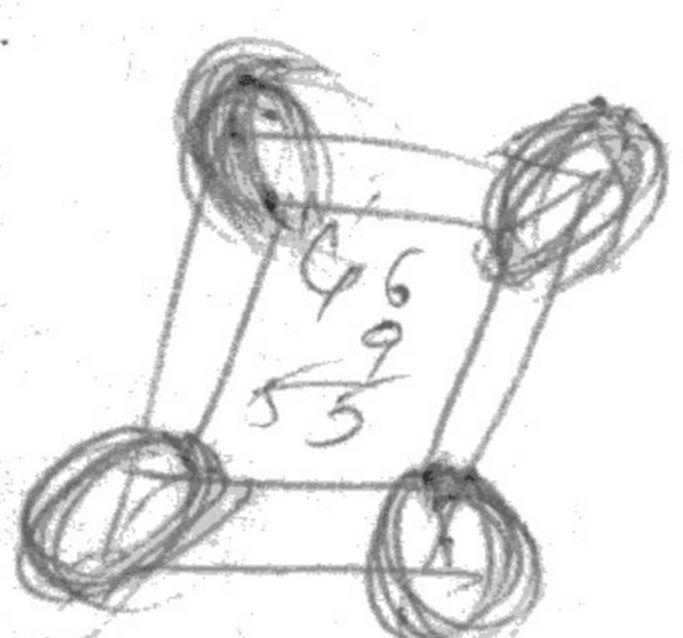
- a. *Cost*: Microcontrollers with the supplementary circuit components are much cheaper than a computer with an analog and digital I/O
- b. *Size and Weight*: Microcontrollers are compact and light compared to computers
- c. *Simple applications*: If the application requires very few number of I/O and the code is relatively small, which do not require extended amount of memory and a simple LCD display is sufficient as a user interface, a microcontroller would be suitable for this application.
- d. *Reliability*: Since the architecture is much simpler than a computer it is less likely to fail.
- e. *Speed*: All the components on the microcontroller are located on a single piece of silicon. Hence, the applications run much faster than it does on a computer.

What is PIC?

Peripheral Interface Controller from Microchip Technology Inc.USA.,

They are Reduced Instruction Set Computer (RISC) Configuration processors and uses Harvard machine architecture where separate memories are used for the program and data which are accessed via separate buses. In the PIC16F84, the program bus is 14 bits wide, whereas the data bus is 8 bits wide. In addition, the PIC family is based on a Reduced Instruction Set Computer (RISC) configuration which use fewer instructions than a Complex Instruction Set Computer (CISC). All the PIC devices use less than 60 instructions.

The PIC family are fully static devices, meaning that they preserve the contents of their registers when the clock frequency is reduced to zero. In PIC microcontrollers, each instruction takes four clock periods to execute. If a 1MHz clock frequency is used, the corresponding clock period is $1\mu\text{sec}$, so each instruction will take $4\mu\text{sec}$ —this time is called the *instruction cycle time* t_i .



The fastest devices in the PIC family can operate at clock frequencies up to 33MHz, with corresponding instruction cycle times of 121nsec.

Most instructions execute in one instruction cycle, but some require two cycles because they need to branch to some destination other than the next address in the PC. Instructions that need two cycles to execute are btfsc, btfss, call, decfsz, goto, incfsz, retfie, retlw and return.

Microchip characterises PIC microcontrollers according to their instruction word lengths. The low-end PICs, such as the eight pin 12C5XX series, have 12 bit word length instructions. The midrange PICs, such as the PIC16XXX, have 14 bit instructions and the high-end 17XXX PICs have 16 bit instructions. All PIC microcontrollers are, however, classified as eight bit microcontrollers as they all manipulate data in byte units on an eight bit wide data bus.

2. PIC16F877 Microcontroller

PIC16F877 is one of the most commonly used microcontrollers especially in automotive, industrial, appliances and consumer applications. In Figure –1, the block diagram of the PIC16F877 is illustrated.

The core features of PIC16F877 are:

- High performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input DC - 200 ns instruction cycle
- Up to 8K x 14 words of FLASH Program Memory, Up to 368 x 8 bytes of Data Memory (RAM), Up to 256 x 8 bytes of EEPROM Data Memory
- Pinout compatible to the PIC16C73B/74B/76/77
- Interrupt capability (up to 14 sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low power, high speed CMOS FLASH/EEPROM technology
- Fully static design
- In-Circuit Serial Programming. (ICSP) via two pins
- Single 5V In-Circuit Serial Programming capability
- In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA

10752
1052

- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
 - < 0.6 mA typical @ 3V, 4 MHz
 - 20 µA typical @ 3V, 32 kHz
 - < 1 µA typical standby current

The peripheral features of the PIC16F877 are:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during SLEEP via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM max. resolution is 10-bit.
- 10-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with SPI. (Master mode) and I2C. (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external RD, WR and CS controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)
- It has 5 ports:

Port A: 6 pins

Port B: 8 pins

Port C: 8 pins

Port D: 8 pins

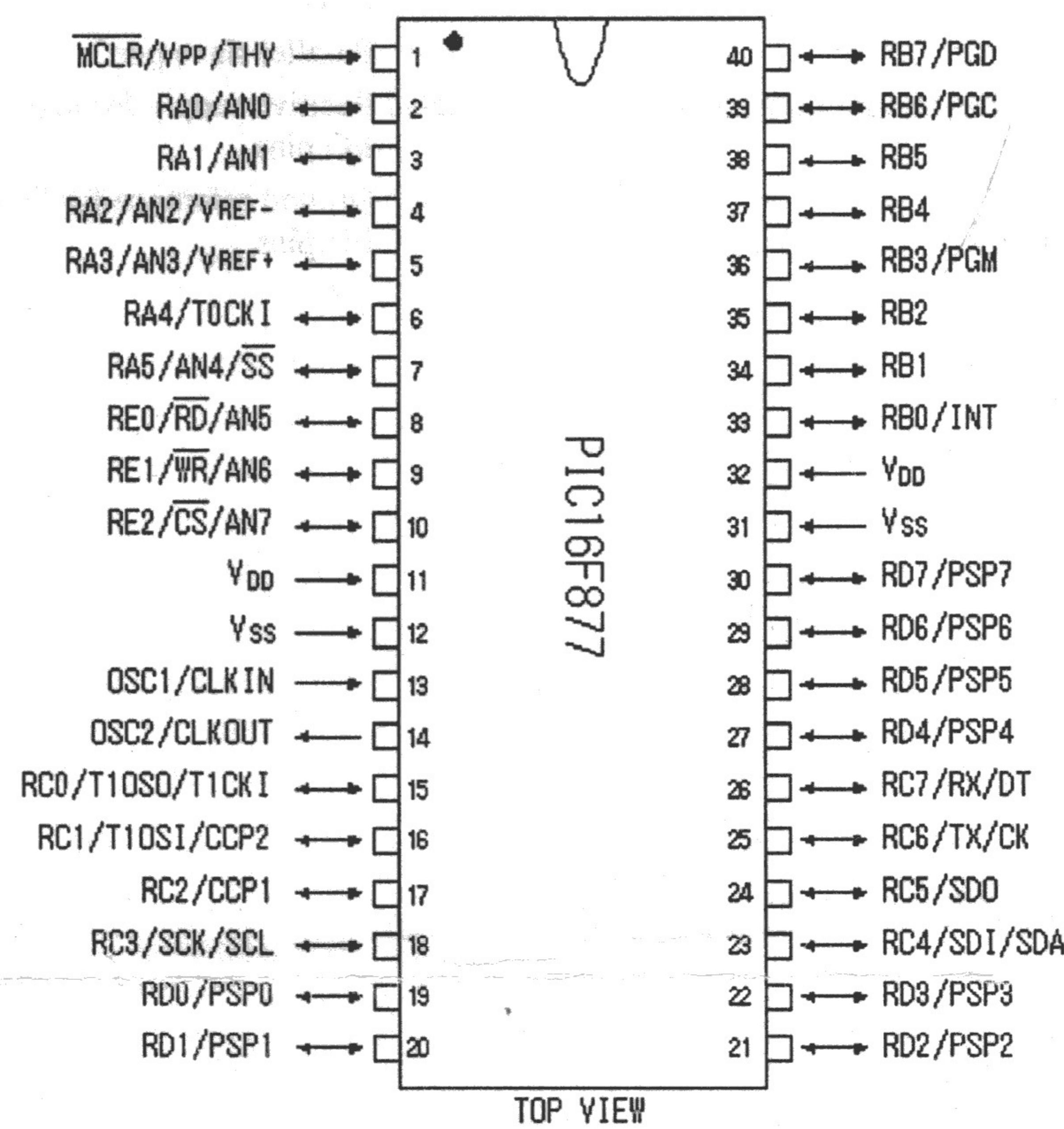
Port E: 3 pins

Low power operation and the sleep instruction

In portable battery powered equipment, it is desirable to minimise power consumption, especially during periods when the microcontroller is idle, waiting for an input or an interrupt to spur it into action. To conserve power while idle, the processor can be placed in a state of 'suspended animation' using the sleep instruction. The power consumption in sleep mode is very low. When the processor is in sleep mode, all activity is suspended, except for the watchdog timer. No execution of instructions is possible as the internal clock is stopped and the timer/counter module is disabled. Waking up the processor can only be achieved via a reset, a pin interrupt or a watchdog timer time-out. To ensure low power consumption, all unused inputs must be connected to one of the supply rails.

The Pinouts of the PIC16F877 is given in Figure 3.0-2

Hardware of the PIC16F877



TOP VIEW

RA0-5 : Input/Output port A
RB0-7 : Input/Output port B
RC0-7 : Input/Output port C
RD0-7 : Input/Output port D
RE0-2 : Input/Output port E
AN0-7 : Analog input port
RX : USART Asynchronous Receive
TX : USART Asynchronous Transmit
SCK : Synchronous serial clock input
SCL : Output for both SPI and I²C modes
DT : Synchronous Data
CK : Synchronous Clock
SDO : SPI Data Out (SPI mode)
SDI : SPI Data In (SPI mode)
SDA : Data I/O (I²C mode)

MCLR : Master Clear (Active low Reset)
Vpp : Programming voltage input
THV : High voltage test mode control
VREF^{+/−} : Reference voltage
SS : Slave select for the synchronous serial port
T0CKI : Clock input to Timer0
T1OSO : Timer1 oscillator output
T1OSI : Timer1 oscillator input
T1CKI : Clock input to Timer1
PGD : Serial programming data
PGC : Serial programming clock
PGM : Low voltage programing input
INT : External interrupt
RD : Read control for the parallel slave port
WR : Write control for the parallel

CCP1,2 : Capture In/Compare
Out/PWM Out

OSC1/CLKIN : Oscillator In/External
Clock In

OSC2/CLKOUT : Oscillator Out/Clock Out

slave port

CS : Select control for the parallel
slave

PSP0-7 : Parallel slave port

VDD : Positive supply for logic and
I/O pins

Vss : Ground reference for logic and
I/O pins

CCP - CCP is a special module designed for modulation & waveform generation application. Each of the CCP module contains 16 bit registers which works as :

- 16 bit capture reg.
- 16 bit compare reg
- PWM duty cycle reg.

CCP mode	Timer Resource
Capture	Timer 0
Compare	Timer 1
PWM	Timer 2

CCP Register 1 (CCPR1) is a 16 bit reg. composed of two 8-bit registers. CCPR1,L (low byte) & CCPR1,H (high byte). The CCP,CON reg. controls the operation of CCP1.

- In capture mode, CCPR1 captures the 16 bit value of Timer 0 (TMR0) reg. when an event occurs on pin CCP1.
- Event is described as one of the foll.
 - Every falling edge
 - or rising edge
 - Every 4th or "
 - Every 16th or "

- In compare mode, the 16 bit CCP1, reg. value is constantly compared against the TMR0, reg. pair value. When a match occurs, the CCP1 pin is

- driven high
 - driven low
 - remain unchanged.
 - In PWM mode, the CCL_x pin produces a PWM op.

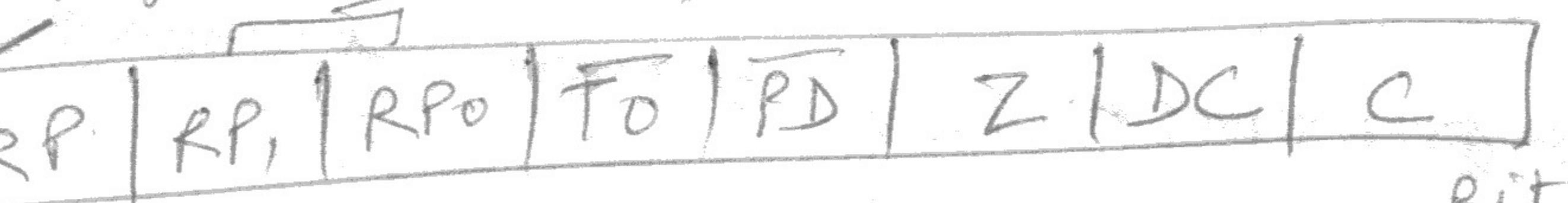
~~■ Three types of memory:~~

 - Program Memory -
 - Data Memory: -
 - Data EEPROM.

General Purpose Registers & Special Function

Registers:

- Registers:

 - Status Register —
 - Reg. Bank Select → Reg. Bank Select (Indirect Addressing)
(Direct Addressing)
 - 

Bit 0 Bit 7 Bit 16
 - $IRP = 0$; program will work with bank 0, 1
 - $IRP = 1$; " " " " " bank 2, 3.

RP, RP₀ Bank

0	1	2	3
1	0	1	2
0	1	1	3
1	0	0	1
0	1	1	2
1	1	1	3
1	1	1	1
1	1	1	1

Set Down Bit

(2)

- 1 - After power by the CLRWDI instruction or after power up.
- 0 - By execution of sleep instruction.

TO - Time out bit.

- 1 - After power up, CLRWDI instrucⁿ or SLEEP instruction.

0 - A WDT time out occurred.

→ PORT REGISTERS 5 I/O ports located in

Bank 0. Corresponding control registers.

($I = 1/P$)

TRISA, TRISB, TRISC, TRISD, TRISE. ($O = 8/P$)

port reg.(in Bank 0)

TRIS reg.(in Bank 1)

By default, TRIS is set in I/P mode.

SFRs - have dedicated system function

- PSP - It allows 8-bit asynchronous bidirectional data transfer b/w the PIC & external devices such as other mc or computers. Used for parallel communication.

6 Instruction set for the PIC microcontroller

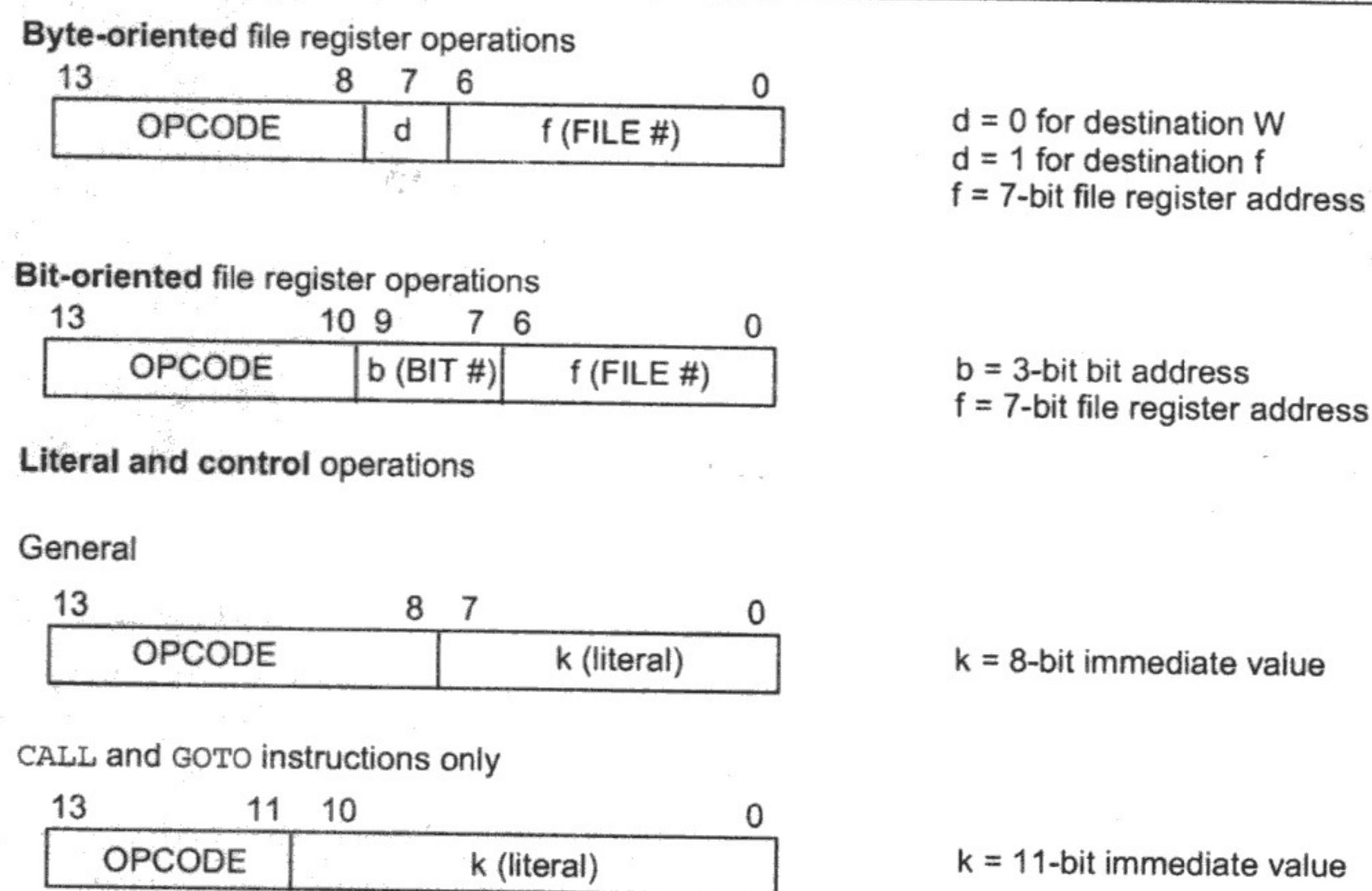


Figure 4: Format of instructions

Mnemonic, Operands	Description		Status Affected
Byte-oriented file register operations			
ADDWF f,d	Add W and f	C,DC,Z	
ANDWF f,d	AND W with f	Z	
CLRF f	Clear f	Z	
CLRW -	Clear W	Z	
COMF f,d	Complement f	Z	
DECFSZ f,d	Decrement f, Skip if 0	Z	
INCF f,d	Increment f	Z	
INCFSZ f,d	Increment f, Skip if 0	Z	
IORWF f,d	Inclusive OR W with f	Z	
MOVF f,d	Move f	Z	
MOVWF d	Move W to f	Z	
NOP -	No operation		
RLF f,d	Rotate Left f through Carry	C	
RRF f,d	Rotate Right f through Carry	C	
SUBWF f,d	Subtract W from f	C,DC,Z	
SWAPF f,d	Swap nibbles in f		
XORWF f,d	Exclusive OR W with f	Z	
Bit-oriented file register operations			
BCF f,b	Bit Clear f		
BSF f,b	Bit Set f		
BTFSC f,b	Bit Test f, Skip if Clear		
BTFSS f,b	Bit Test f, Skip if Set		
Literal and Control operations			
ADDLW k	Add literal and W	C,DC,Z	
ANDLW k	AND literal with W	Z	
CALL k	Call subroutine		
CLRWDT -	Clear watchdog timer	TO, PD	
GOTO k	Goto address		
IORLW k	Inclusive OR literal with W	Z	
MOVLW k	Move literal to W		
RETFIE -	Return from interrupt		
RETLW k	Return with literal in W		
RETURN -	Return from subroutine		
SLEEP -	Clear watchdog timer	TO, PD	
SUBLW k	Subtract W from literal	C,DC,Z	
XORLW k	Exclusive OR literal with W	Z	

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
d	Destination select: d = 0, Store result in W d = 1, Store result in file register f default is d = 1