DESIGN AND IMPLEMENTATION OF kWhr METER USING EMBEDDED SYSTEMS

A PROJECT REPORT

Submitted by

J.MOHANAPRIYA	(71201105025)
M.SAGAYA SABINA	(71201105044)
G.YOGASRI	(71201105073)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRICAL AND ELECTRONICS ENGINEERING

Under the guidance of

Mr. V.Kandasamy

KUMARAGURU COLLEGE OF TECHNOLOGY,COIMBATORE

ANNA UNIVERSITY::CHENNAI-600 025

APRIL 2005

ANNA UNIVERSITY : CHENNAI-600 025

BONAFIDE CERTIFICATE

Certified that this project report " **DESIGN AND IMPLEMENTATAION OF kWhr METER USING EMBEDDED SYSTEMS**" is the bonafide work of

J.MOHANAPRIYA (71201105025)
M.SAGAYA SABINA (71201105044)
G.YOGASRI (71201105073)

who carried out the project work under my supervision.

SIGNATURE

Prof.K.Regupathy subramaniam,B.E.,M.Sc.
DEAN/EEE,
Kumaraguru college
of technology

SIGNATURE

Mr.V.Kandasamy,M.Tech,
Lecturer,EEE,
Kumaraguru college
of technology

CERTIFICATION OF EVALUATION

College : KUMARAGURU COLLEGE OF TECHNOLOGY

Branch : Electrical And Electronics Engineering

Semester : Eighth semester

SI.No.	Name of the students	Title of the project	Name of the supervisor with designation
01	J.MOHANAPRIYA	**“Design and implementation of kWhr meter using embedded systems”**	Mr.V.KANDASAMY LECTURER.
02	M.SAGAYA SABINA		
03	G.YOGASRI		

The report of the project work submitted by the above students in partial fulfillment for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering of Anna University were evaluated and confirmed to be report of the work done by the above students and then evaluated.

15/4/05

(INTERNAL EXAMINER) **(EXTERNAL EXAMINER)**

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people who gave us constant guidance and support.

We would like to express our deep sense of gratitude towards our project guide **Mr.V.Kandasamy,M.Tech** lecturer, for directing us towards a more application-oriented project and for his support and interest in all our endeavors.

We would like to thank our project coordinator **Prof.K.Rajan M.E** assistant professor for having channelized our thoughts towards this project.

We would like to thank our former head of the department **Dr.T.M.Kameswaran PhD** for his constant encouragement and support.

We also thank our Dean **Prof.Reghupathy Subramanian B.E, M.sc** for his constant guidance and support.

Further we would like to acknowledge the support shown by our beloved principal **Dr.K.K.Padmanabhan B.Sc(Engg), Ph.D.**

We also extend our heartiest thanks to all our staff members, non-technicians, without them we would have never attained this form.

SYNOPSIS

SYNOPSIS

The Kilowatt-hour meter cum cost indicator using embedded systems is a very reliable energy meter. It is designed so as to measure and display the number of units consumed and its corresponding cost. All houses, apartments, offices and industries are equipped with energy meter. At present, analog energy meters are very commonly used. In these energy meters, the number of units consumed for a certain period can be found by calculating the difference between readings at the start and end of the period in an energy meter. Our proposed energy meter digitally displays the number of units consumed and corresponding costs for it.

The Microcontroller we have used is PIC 16F877. It has only 35 single word instructions and thus easily programmable. It has an inbuilt ADC, which reduces the complexity of the circuit. In our project we have used EEPROM data memory.

The interfacing to the LCD display is done directly through the I/O ports of the PIC.We have also transmitted the data to the pc through serial communication.

DEDICATED
TO
OUR
BELOVED PARENTS
AND
FRIENDS

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Need For Our Project

Energy plays a significant role in development of nation. Today there is a trend towards tremendous growth and phenomenal modernization of electrical system throughout the world. Energy is infact power integrated over a period of time. Energy meters are essentially integrating instruments reading a continuous quantity and are quite different from other integrating instruments. They measure power over a period of time, integrating it into energy. Thus the amount of electrical energy supplied to the customers is metered.

Induction type meters are the most common form of AC energy meters used in everyday domestic and industrial installations. There are various disadvantages encountered in analog energy meters. Errors in conventional energy meters are mainly due to forces at the rotor bearings and in register mechanism. These errors are avoided in digital energy meters. In case of digital measurements the readings obtained are precise and accurate. Our proposed energy meter using embedded system displays the energy consumed over a period of time and its corresponding cost and thus manual calculations are eliminated.

In small industries, when the cost goes beyond the limit luxury equipment like fans, refrigerators and air conditioners can be turned off to reduce energy consumption.

1.2 ANALOG ENERGY METER

An energy meter is a device that is used to measure the energy consumed by the electrical equipment connected to the supply. The Induction Type Energy Meter is universally used in domestic and industrial AC circuits.

1.2.1 PRINCIPLE OF OPERATION

The main principle of operation of analog equipments mainly depends on magnetic field produced. In analog energy meters the eddy currents produced due to magnetic flux plays an important role in energy measurement. The single-phase energy meter consists of two coils namely current coil and pressure coil. The current coil is made up of few turns of thick wire and is connected in series with the load. The pressure coil consists of few turns of thin wire and is connected in parallel with the supply. An aluminum disc mounted on a spindle is placed between two coils. The two fluxes produced by the two coils interact with aluminium disc and produces eddy currents in the disc and hence the disc starts rotating. The speed of the disc is proportional to the energy consumed by the load.

1.2.2 DISADVANTAGES

Disadvantages present in analog meters are

- The lifetime of analog meters are very less.
- The readings obtained are not precise and accurate.
- The readings obtained can easily be tampered.
- Presence of mechanical moving parts.

1.3 ELECTRICAL ENERGY

The unit of energy is Joule, which is the work done when a force of 1N acts through a distance of 1m in the direction of the force.

Electrical energy		voltage in		current in		time in
in watt hour	=	volts	*	amps	*	hours

1.4 TARIFF

The rate at which electrical energy supplied to the customer is known as tariff .The various types of tariff are

- Simple tariff
- Flat rate tariff
- Block rate tariff
- Two port tariff
- Maximum demand tariff
- Power factor tariff
- Three port tariff

The TNEB tariff for most of the organizations like schools, colleges, hospitals, residential houses, hotels and shops falls in one of the first three tariffs.

METHODOLOGY

CHAPTER 2

METHODOLOGY

2.1 SCHEMATIC DIAGRAM

The schematic diagram of our project is shown below in figure 2.1.

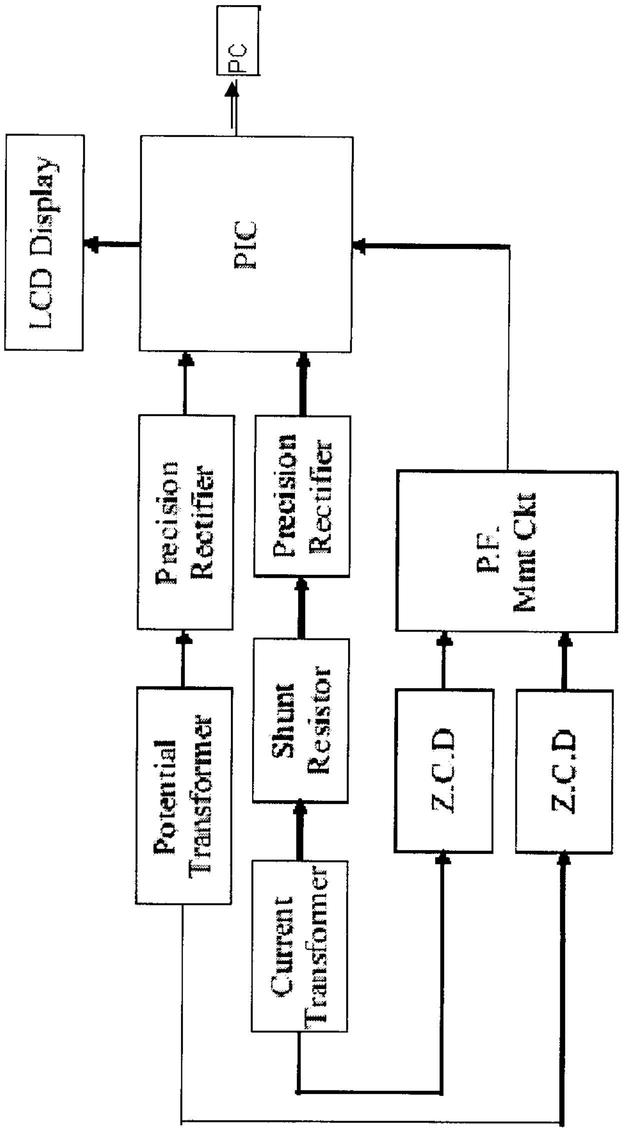


Fig 2.1 BLOCK DIAGRAM

The voltage and current signals are sensed and stepped down using potential transformer and current transformer respectively. The voltage and current signals are then given to precision rectifier to convert into an equivalent D.C signal. The current measurement circuit is similar to the voltage measurement circuit except for the shunt resistor included in the current measurement circuit.

The time lag between the voltage and the current signal is calculated using a zero crossing detector along with an XOR gate from which power factor can be calculated. The voltage signal, current signal and the power factor are fed to the PIC microcontroller.

The PIC 16F877 is used to calculate the energy and the cost for the number of units consumed. The voltage, current, powerfactor, energy and the cost are displayed using a LCD.The PIC is also interfaced to PC.

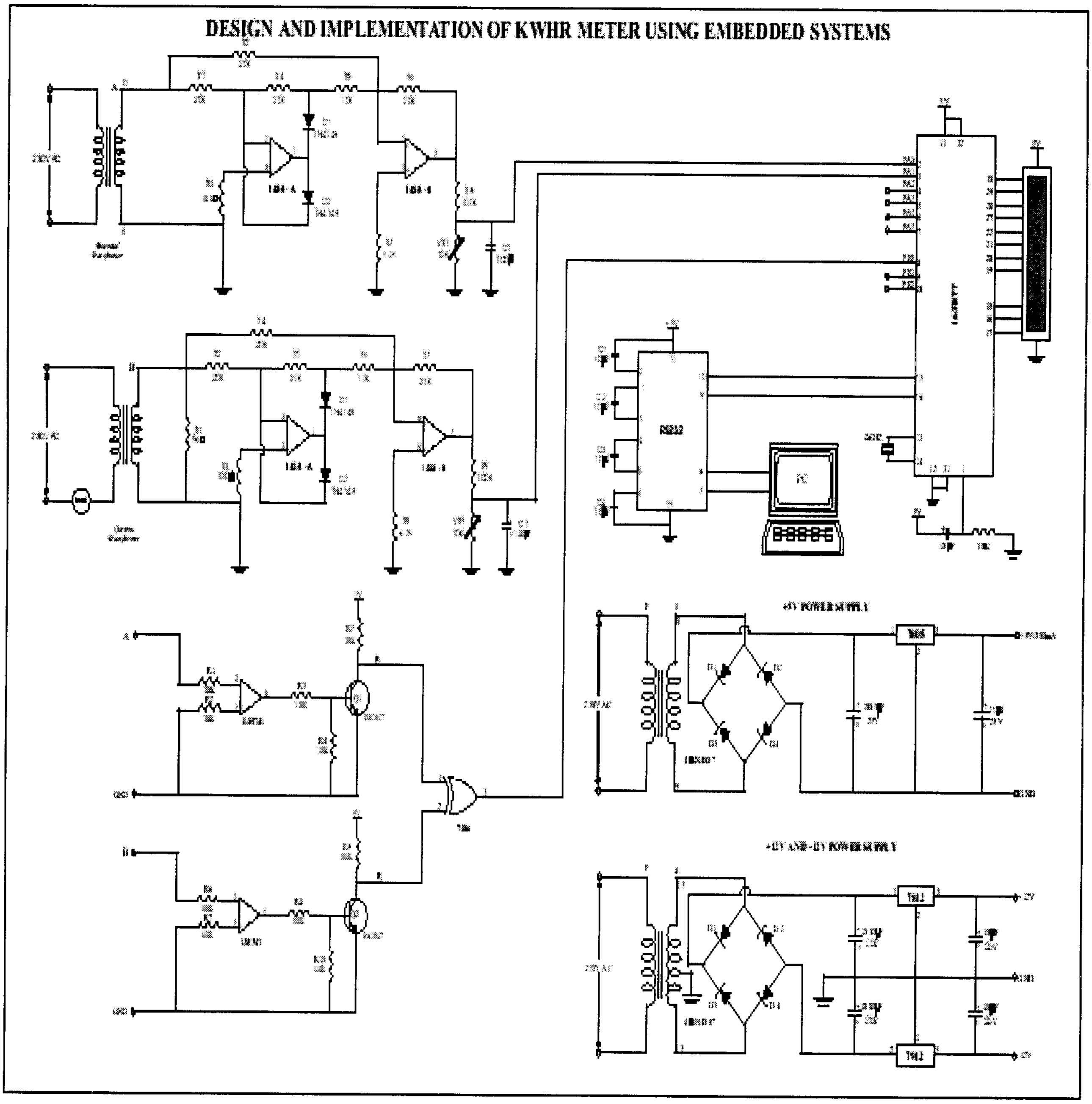

DESIGN AND IMPLEMENTATION OF KWHR METER USING EMBEDDED SYSTEMS
230V AC
RS232
PC
+5V POWER SUPPLY
+12V AND -12V POWER SUPPLY
7805
7812
7912
GND
A
B

Fig.2.2 OVERALL CIRCUIT DIAGRAM

HARDWARE

CHAPTER 3

HARDWARE

3.1 POWER SUPPLY UNIT

Since all the power circuits work only with low D.C. voltage we need a power supply unit to provide the appropriate voltage supply. The general power supply unit is shown in fig.3.1

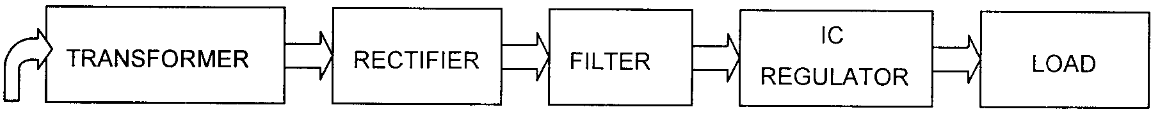
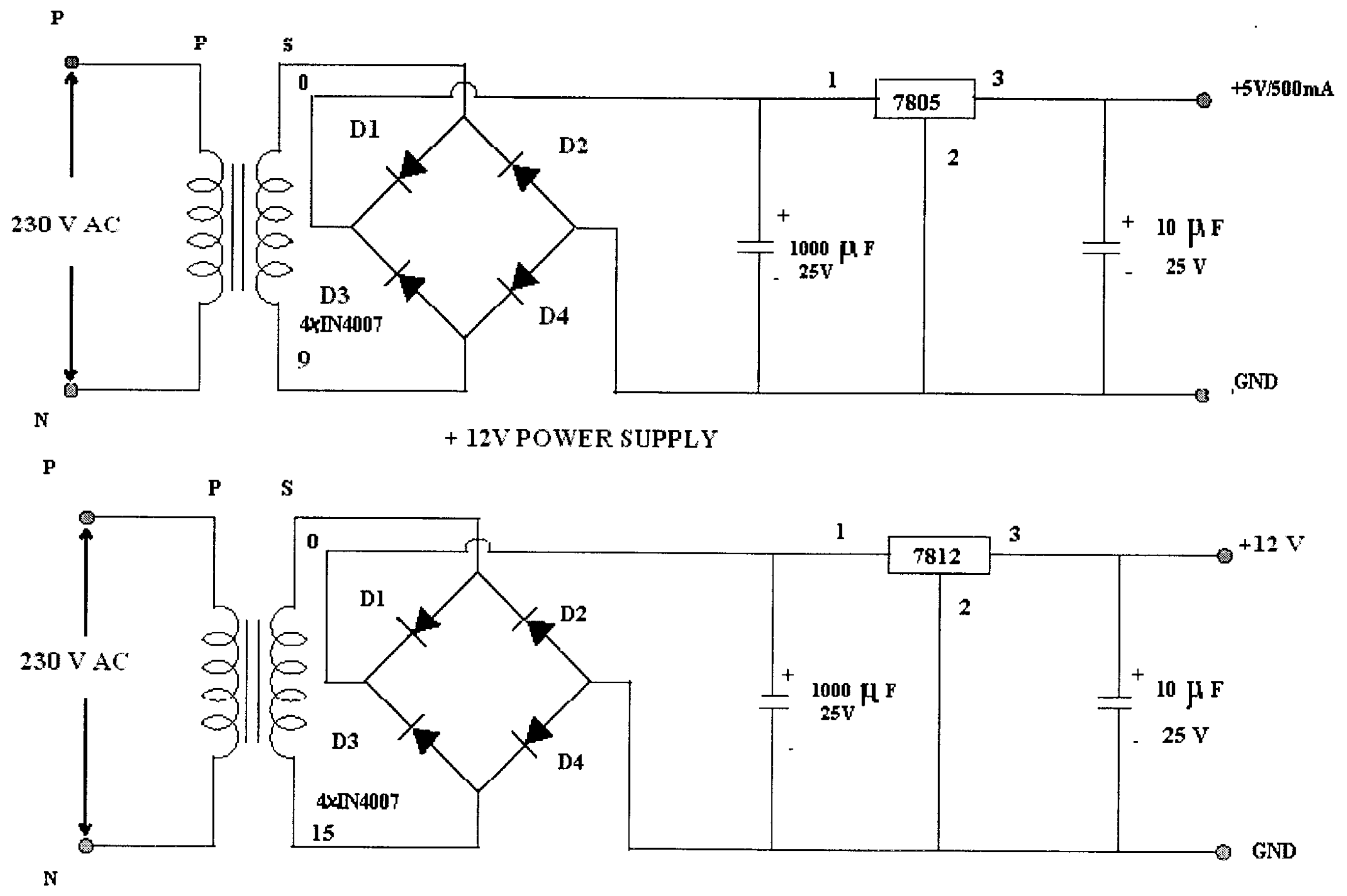


Fig 3.1 POWER SUPPLY UNIT

This unit consists of transformers, rectifiers, filters and regulators. The input AC voltage diode rectifier then provides a full wave rectified, typically 230Vrms is connected to a transformer to obtain the desired step-down AC voltage. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a DC voltage. This resulting DC voltage usually has some ripple or AC voltage variations. A regulator circuit can use this DC input to provide DC voltage that not only has less ripple voltage but also remains the same DC value even if the DC voltage varies, or when the load connected to the output DC voltage changes.

The power supply unit comprises of two circuits, one providing a +5V supply and the other providing +12V and -12V supply. The circuit diagrams for the same are shown in the Figure 3.2.



+ 5V POWER SUPPLY

+ 12V POWER SUPPLY

Fig 3.2 POWER SUPPLY UNIT

3.1.1 RECTIFIER

The DC level obtained from a sinusoidal input can be improved 100% using a process called full-wave rectification. It uses 4 diodes in a bridge configuration. From the basic bridge configuration we see that two diodes (say D2, D3) are conducting while the other two diodes (D1,D4) are in “off” state during the period t=0 to T/2. Accordingly for the negative half cycle of the input the conducting diodes are D1and D4.Thus the polarity across the load is the same.

3.1.2 FILTER

The filter circuit used here is the capacitor is connected at the rectifier output, and a DC is obtained across it. The filtered waveform is essentially a DC voltage with negligible ripples, while is ultimately fed to the load.

3.1.3 REGULATORS

The voltage regulator is a device, which maintains the output voltage constant irrespective of the change in supply variations, load variation and temperature changes. The fixed voltage regulator is a three terminal device which has an unregulated dc input voltage, Vi, applied to one input terminal, a regulated output dc voltage, Vo, from a second terminal, with the third terminal connected to ground .For a selected regulator, IC device specifications list a voltage range over which the input voltage can vary to maintain a regulated output voltage over a range of load current. The specifications also list the amount of output voltage change resulting from a change in load current or in input voltage.

Here we use two fixed voltage regulators namely LM7812, LM7805 and LM7912.The IC7812 is a +12V regulator, IC7912 is a -12V regulator and IC7805 is a +5V, regulator. The series 78 regulators provide regulated voltages from -5 to -24 V.

3.2 VOLTAGE & CURRENT MEASUREMENT

3.2.1 SPECIFICATIONS

a) **Potential Transformer ratings**

Primary:	230V AC
Secondary:	6V AC

b) **Current Transformer ratings**

Primary:	15A
Secondary:	5A

3.2.2 VOLTAGE MEASUREMENT

Voltage measurement circuit comprises of a potential transformer and precision rectifier. Potential transformer is used for sensing and stepping down the voltage signal. Precision rectifier converts the stepped down AC signal into an equivalent DC signal. The DC output voltage signal is then given to the zero crossing detector and the PIC microcontroller unit.

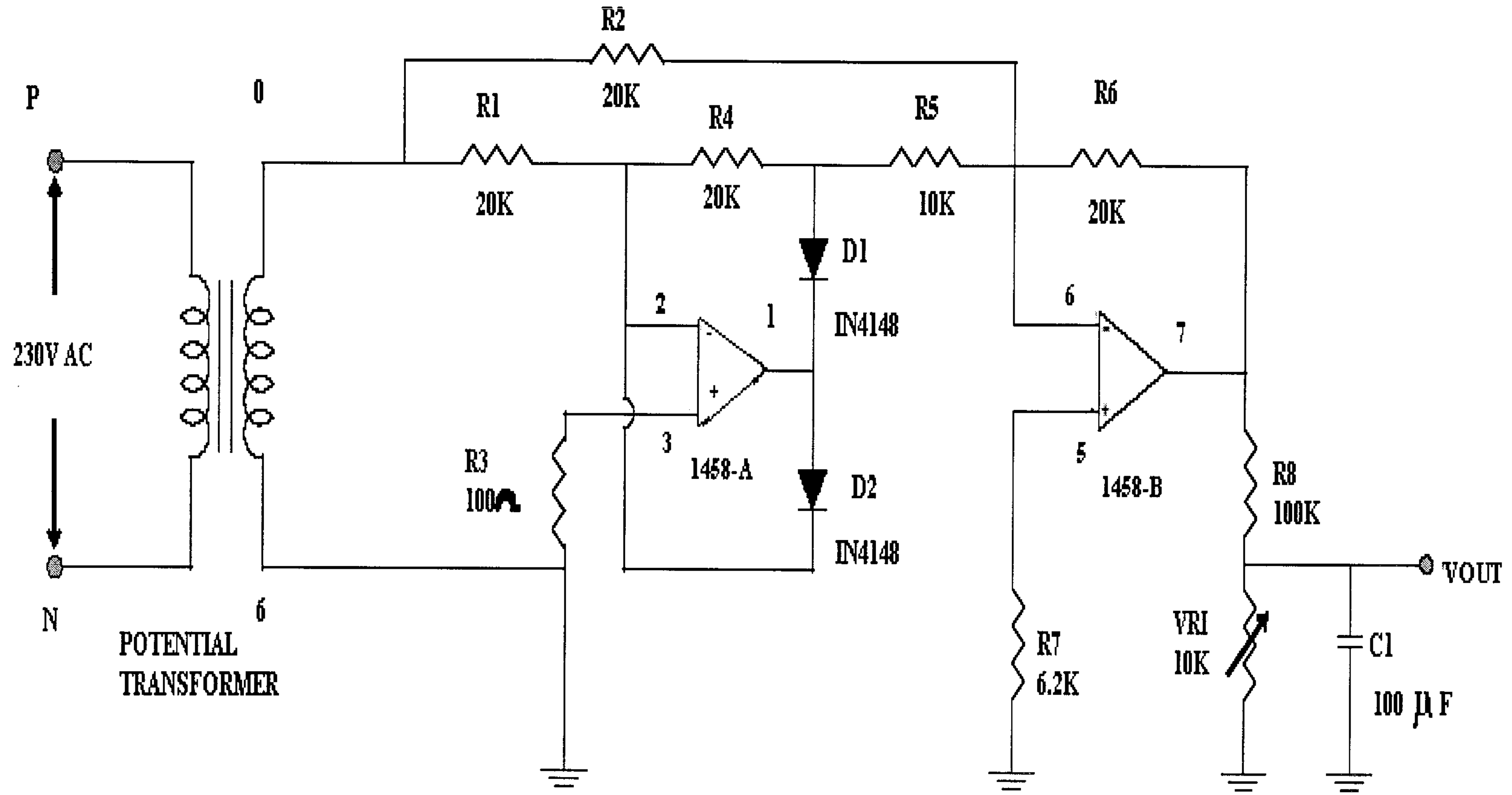


Fig 3.2.2 VOLTAGE MEASUREMENT CIRCUIT

3.2.3 ABSOLUTE VALUE CIRCUITS

An absolute-value circuit, or full-wave precision rectifier, can be implemented by summing the output of a half-wave rectifier and its input with the proper phase and amplitude relations. This circuit will be the starting point for a number of other absolute-value circuits, which have evolved from this basic form.

In this circuit, A1 is an inverting rectifier similar to the figure. The output from A1 is added to the original input signal in A2 (a summing mixer) with the signal amplitude and phase relations shown. Negative alterations of Ein feeds A2 through 20-kΩ resistor, and E1 feed A2 through a 10-kΩ resistor. The net effect of this scaling is that, for equal amplitudes of Ein and E1, E1will provide twice as much current into the summing point. This fact is used to advantage here, as the negative alteration of E1 produces twice the input current of that caused by the positive alternation of Ein.

This causes a current of precisely half the amplitude, which E1 alone would generate due to the subtraction of Ein. It is the equivalent of having E1 feed through a 20-kΩ input receiver and having Ein non-existent during this half cycle, and it results in a positive going output at A2. During negative alterations of Ein, E1 is absent and Ein produces the alternate positive output swing that, in summation, produces the desired full-wave rectified response. As before, operation with the opposite output polarity is possible by reversing D1 and D2.

3.2.4 CURRENT MEASUREMENT

The current measurement circuit consists of Current transformer of turns ratio 5:1.The Current Transformer's primary winding is connected in series with the line. The secondary of the Current Transformer is connected to a precision rectifier network. The shunt resistor of value 56 ohm is connected across the CT. Hence we have ac voltage proportional to the current flowing through the primary of the CT given as the input to the rectifier unit and by a similar procedure as seen in case of voltage measurement, we get pure DC at the output stage of the rectifier.The current measurement circuit is shown in figure 3.2.4.

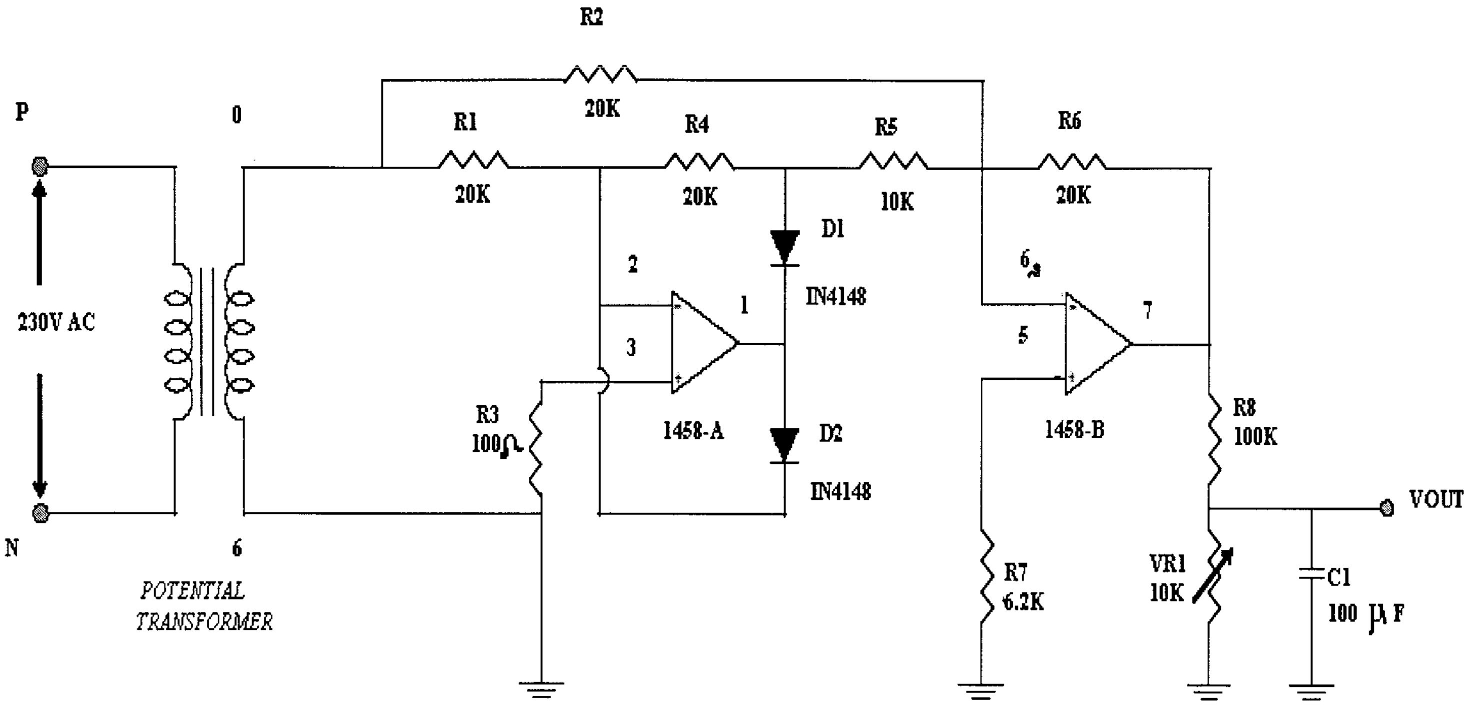


Fig 3.2.4 CURRENT MEASUREMENT CIRCUIT

3.3. POWER FACTOR MEASUREMENT

The power factor measurement circuit consists of PT, CT, zero crossing detector and finally an XOR gate. PT and CT are used to sense the voltage and current signals respectively. Zero crossing detector, converts the sine wave into square wave. The circuit is also called a sine to square wave generator.

The square pulse outputs of the ZCD circuits are given as an input to an XOR gate. The output of the gate depends upon the input given. The truth table of XOR gate is

Y= AB' + A'B.

TABLE: 3.3.1. TRUTH TABLE FOR XOR GATE

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

The XOR gate will produce the output when both the inputs to the gate are not in the same state. So the output of the gate is the indirect measure of phase angle between the voltage and current. A timer is incremented in millisecond, which is equivalent to corresponding phase angle in degrees. Leading or lagging power factor is determined based on the pulse at the output of voltage's Zero Crossing Detector .

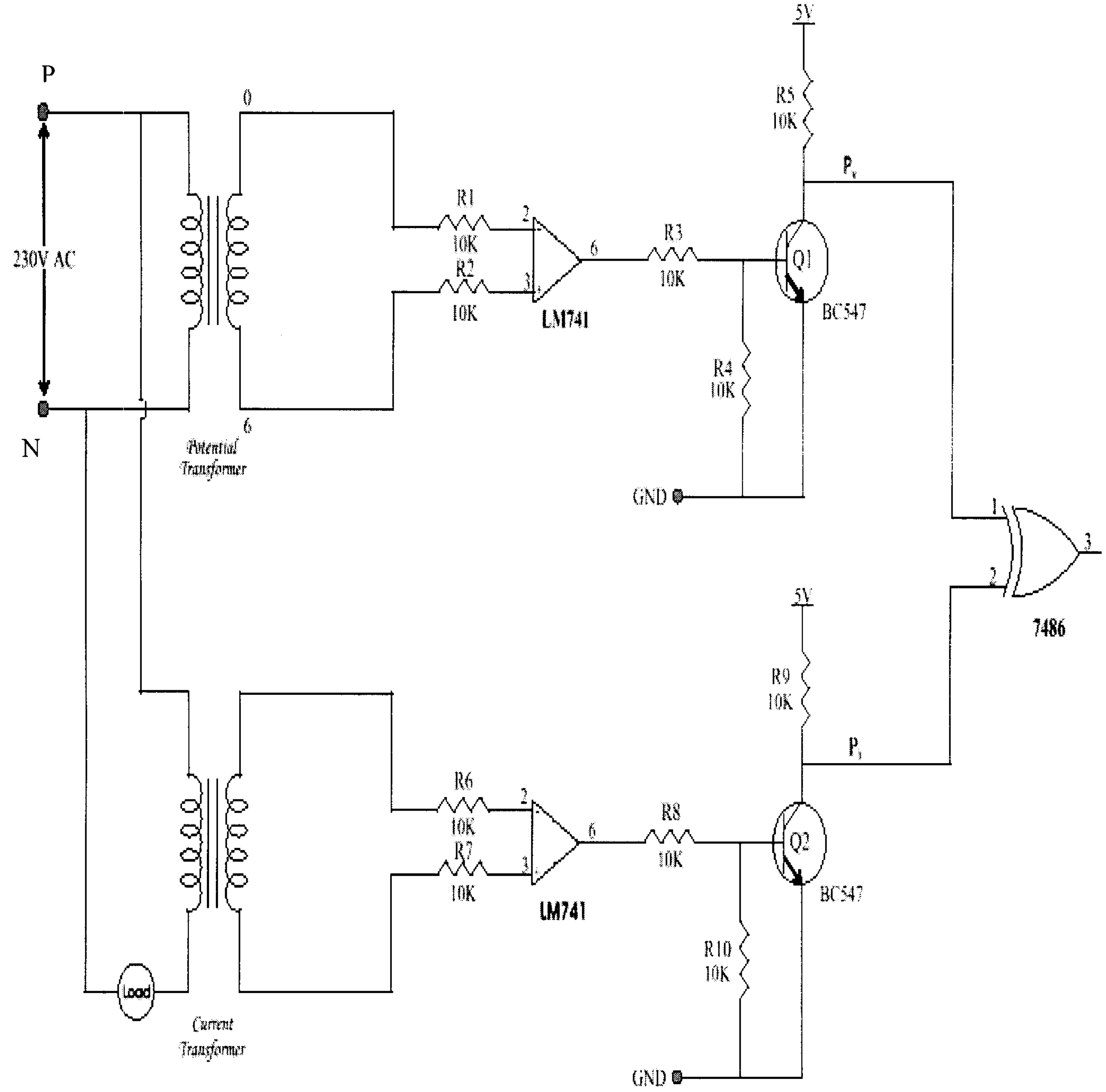


Fig 3.3.1 POWER FACTOR MEASUREMENT CIRCUIT

3.4 LCD DISPLAY

In our project, the readings obtained as shown on an LCD display. An LCD consists of two glass panels, with the liquid crystal material sand-witched between them. The inner surface of the glass plate is coated with transparent electrodes, which define the character, symbols or patterns to be displayed. Polymetric layers are present in between the electrodes and the light crystal, which makes the liquid crystal electrodes to maintain a defined oriented angle.

Polarisers are pasted outside the glass panels. These polarisers would rotate the light rays passing through them to a definite angle, in a particular direction. When the LCD is in off state, light rays are rotated by the two polarisers and the liquid crystal, such that the light rays come out of the LCD without any orientation, and hence the LCD appears transparent.

When sufficient voltage is applied to the electrodes, the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarisers, which would result in activating /highlighting the desired characters. The LCD's are lightweight with only a few millimeters thickness. Since the LCD's consume less power, they are compatible with low power electronic circuits, and can be powered for long durations. The LCD doesn't generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. The LCD's have long life and a wide operating temperature range.

MICROCONTROLLER UNIT

CHAPTER 4

MICROCONTROLLER UNIT

4.1 PIC 16F877 ARCHITECTURE

The microcontroller that we have used is PIC 16F877. The PIC series comprises of PIC16F873, PIC16F874, PIC16F876 and PIC16F877. PIC16F873 and PIC16F874 are 28-pin packages. In our project we have used the PIC16F877 for its outstanding features like:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- 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
- Processor read/write access to program memory
- Wide operating voltage range: 2.0V to 5.5V
- High Sink/Source Current: 25 mA

Pin Diagram

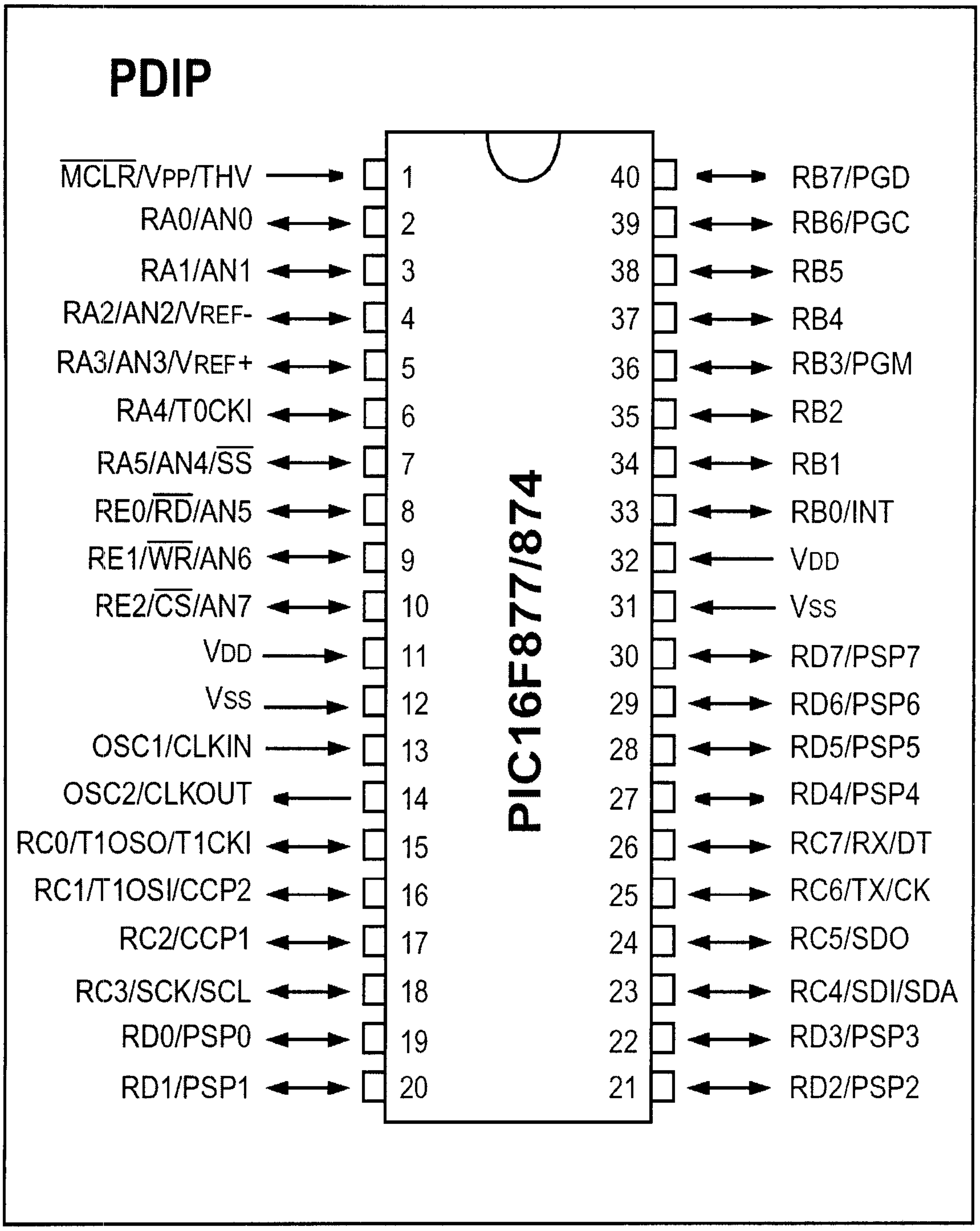
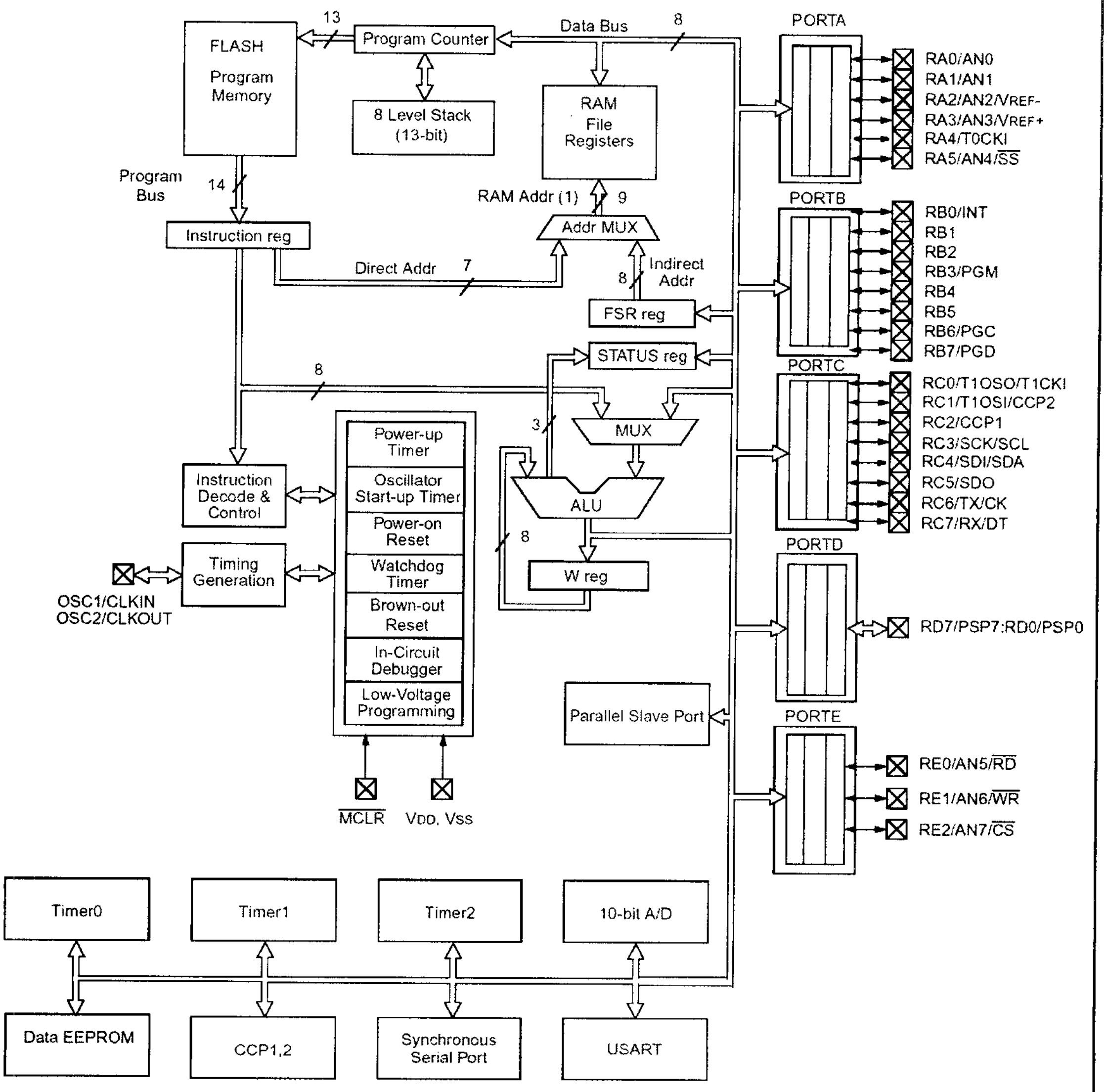


Fig 4.1: PIN DIAGRAM OF 16F877

ARCHITECTURE:

Device	Program FLASH	Data Memory	Data EEPROM
PIC16F874	4K	192 Bytes	128 Bytes
PIC16F877	8K	368 Bytes	256 Bytes



Note 1: Higher order bits are from the STATUS register.

Fig 4.2: ARCHITECTURE OF 16F877

4. 2 MEMORY ORGANIZATION:

There are three memory blocks in each of these PIC MCU's. The Program Memory and Data Memory have separate buses so that concurrent access can occur.

4.2.1 PROGRAM MEMORY ORGANIZATION:

The PIC16F87X devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16F877/876 devices have 8K x 14 words of FLASH program memory. Accessing a location above the physically implemented address will cause a wrap around. The reset vector is at 0000h and the interrupt vector is at 0004h.

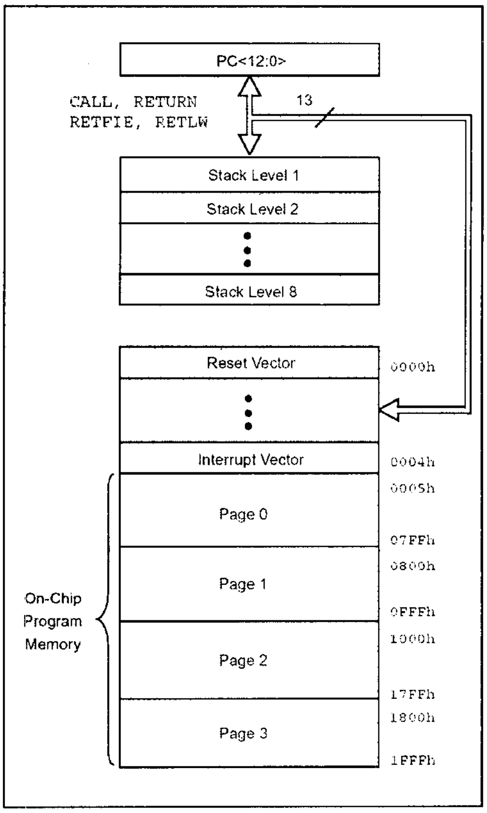


Fig.4.3 PROGRAM MEMORY ORGANISATION

4.2.2 DATA MEMORY ORGANIZATION

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers.

RP1:RP0	Bank
00	0
01	1
10	2
11	3

Table 4.1 BANK SELECTION

All implemented banks contain Special Function Registers. Some "high use" Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

The register file can be accessed either directly, or indirectly through the File Select Register, FSR.

The special function registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. The special function registers can be classified into 2 sets: Core (CPU) and peripheral. The registers present in SFR's are explained in brief.

The STATUS register contains the arithmetic status of the ALU, the RESET, status and bank select bits for data memory. The OPTION-REG register is a readable and writable register, which contains various control bits to configure the TMRO prescales/WDT postscaler, TMRO and the weak pull-ups on port B.

The INTCON register is a readable and writable register , which contains various enable and flag bits for the TMRO register overflow, RB port change and external RBO/INT pin interrupts.

The PIE1 register contains the individual enable bits for peripheral interrupts.

The PIR1 register contains the individual flag bits for peripheral interrupts.

The PIE2 register contains the individual enable bits for CCP2 peripheral interrupt, the SSP bus collision interrupt and EEPROM write operation interrupt's

The PIR2 register contains the individual flag bits for CCP2 peripheral interrupt, the SSP bus collision interrupt and EEPROM write operation interrupt's.

4.3 I/O PORTS

There are totally 6 ports present in PIC16F877. They are port A, port B, port C, port D, port E and parallel slave port.

PORTA AND THE TRISA REGISTER:

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

PORTB AND THE TRISB REGISTER:

PORTB is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

PORTC AND THE TRISC REGISTER:

PORTC is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISC. Setting a TRISC bit (=1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISC bit (=0) will make the corresponding PORTC pin an output (i.e., put the contents of the output latch on the selected pin).

PORTD AND TRISD REGISTERS:

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output. PORTD can be configured as an 8-bit wide microprocessor port (parallel slave port) by setting control bit PSPMODE (TRISE<4>). In this mode, the input buffers are TTL.

PORTE AND TRISE REGISTER:

PORTE has three pins, RE0/RD/AN5, RE1/WR/AN6 and RE2/CS/AN7, which are individually configurable as inputs or outputs.

4.4 ANALOG TO DIGITAL CONVERTER

The Analog-to-Digital (A/D) Converter module has eight inputs for the 40 device.The analog input charges a sample and hold capacitor. The output of the sample and hold capacitor is the input into the converter. The converter then generates a digital result of this analog level via successive approximation. The A/D conversion of the analog input signal results in a corresponding 10-bit digital number. The A/D module has high and low voltage reference input that is software selectable to some combination of VDD, VSS, RA2 or RA3.

The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode. To operate in sleep, the A/D clock must be derived from the A/D's internal RC oscillator. The A/D module has four registers.

These registers are:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register0 (ADCON0)
- A/D Control Register1 (ADCON1)

The ADCON0 register controls the operation of the A/D module. The port pins can be configured as analog inputs (RA3 can also be the voltage reference) or as digital I/O.

4.5 TIMERS

TIMER 0 MODULE

The timer 0 module timer/counter has the following features:

- 8-bit timer counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal and external clock select
- Interrupt on overflow from ffh to ooh
- Edge select for external clock

The TMRO interrupt is generated when the TMRO register overflows from ffh to ooh. This overflow sets bit TOIF (INTCON<2>). The interrupt can be masked by cleaning bit TOIE (INTCON<5>). Bit TOIF must be cleared in software by timer 0 module. Interrupt service routine before re-sending the interrupt. The TMRO interrupt cannot awaken the processor from sleep, since timer is shut-off during sleep.

SERIAL DATA COMMUNICATION

CHAPTER 5

SERIAL DATA COMMUNICATION

5.1 INTRODUCTION

Modem and other devices used to send serial data are often referred to as data communication equipment or DCE. The terminals or computers that are sending or receiving the data are referred to as data terminal equipment or DTE. In response to the need for signal and handshake standards between DTE and DCE, the Electronic Industries Association (EIA) developed standard RS-232c. This standard describes the functions of 25 signals and handshake pins for serial data transfer. It also describes the voltage levels, Impedance levels, rise and fall times, Maximum bit rates, and maximum capacitance for these signal lines. RS232-c specifies 25 signal pins and it specifies that the DTE connector should be a male, and the DCE connector should be a female.

TABLE: 5.1. RS232 PIN ASSIGNMENTS

Pin1	Received line signal detector (data carrier detect)
Pin2	Received data
Pin3	Transmit data
Pin4	Data terminal ready
Pin5	Signal ground
Pin6	Data set ready
Pin7	Request to send
Pin8	Clear to send
Pin9	Ring indicator

5.1. PIN DESCRIPTION

DTR (Data Terminal Ready)

When the terminal is turned on, after going through a self-test, it sends out signal DTR to indicate that it is ready for communication.

DSR (Data Set Ready)

When DCE is turned on and has gone through the self-test, it asserts DSR to indicate that it is ready to communicate.

RTS (Request to Send)

When DTE device such as has a byte to transmit, it asserts RTS to signal the modem that it has a byte of data transmit.

CTS (Clear To Send)

In response to RTS when the modem has room for storing the data it is to receive, it sends out signal CTS to the DTE (pc) to indicate that it can receive the data now.

DCD(Data Carrier Detect)

The modem assert signal DCD to inform the DTE (pc) that a valid carrier has been detected and that contact between it and other modem is established.

RI (Ring Indicator)

An output from the modem (DCE) and an input to a pc (DTE) indicates that the telephone is ringing. It goes on and off in synchronization with the ringing sound.

While signals DTR and DSR are used by the pc and modem respectively, to indicate that they are alive and well, it is RTS and CTS that actually control the flow of data. RTS and CTS are also referred to as hardware control flow signals.

This concludes the description of the most important pins of the RS232 handshake signals plus TxD, RxD and ground. Ground is also referred to as SG (signal ground).

5.2 CHARACTERISTICS OF MAX 232

5.2.1. ELECTRICAL CHARACTERISTICS OF MAX 232

Vcc = 6V.

V+ = 12V.

V- = 12V.

5.2.2. INPUT VOLTAGE

T1 in, T2 in = -0.3 to (Vcc + 0.3V)

R1 in, R2 in = +30V or –30V.

5.2.3. OUTPUT VOLTAGE

TI out, T2 out = ((V+) + 0.3V) to ((V-) + 0.3V).

R1 out, R2 out = -0.3V to (Vcc + 0.3V).

Power dissipation = 375 mW.

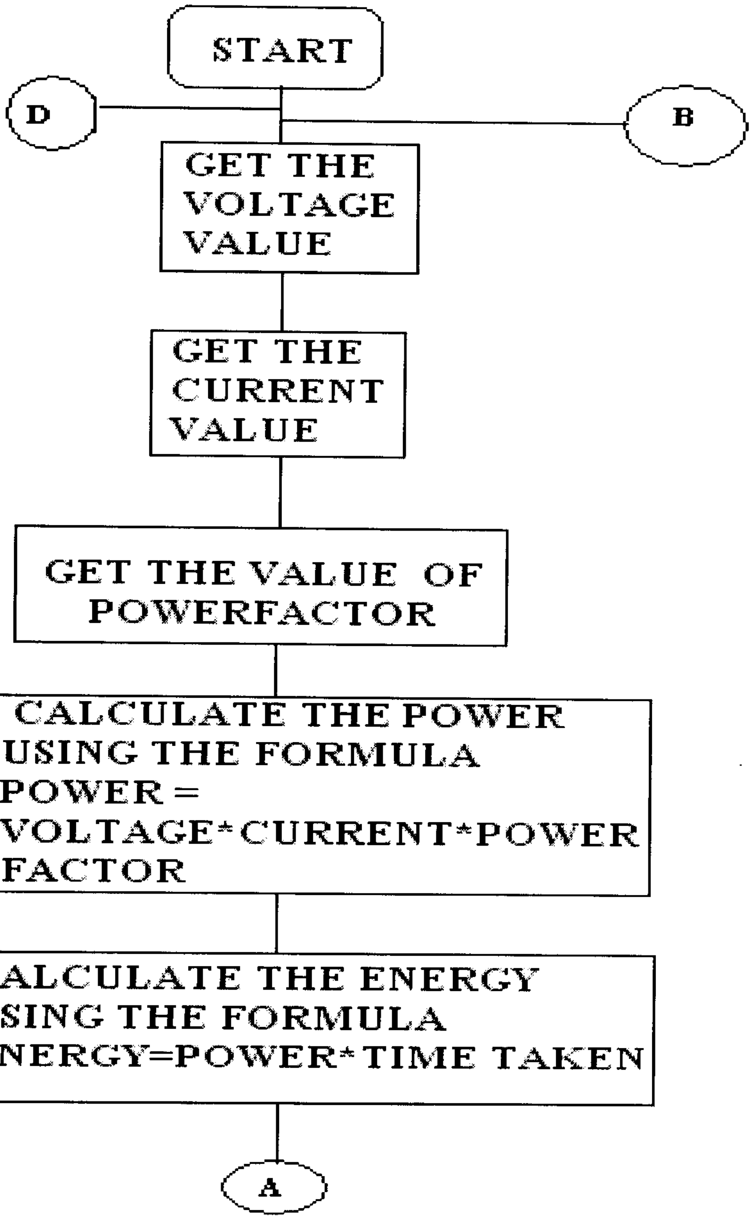
Output resistance = 300Ω.

SOFTWARE

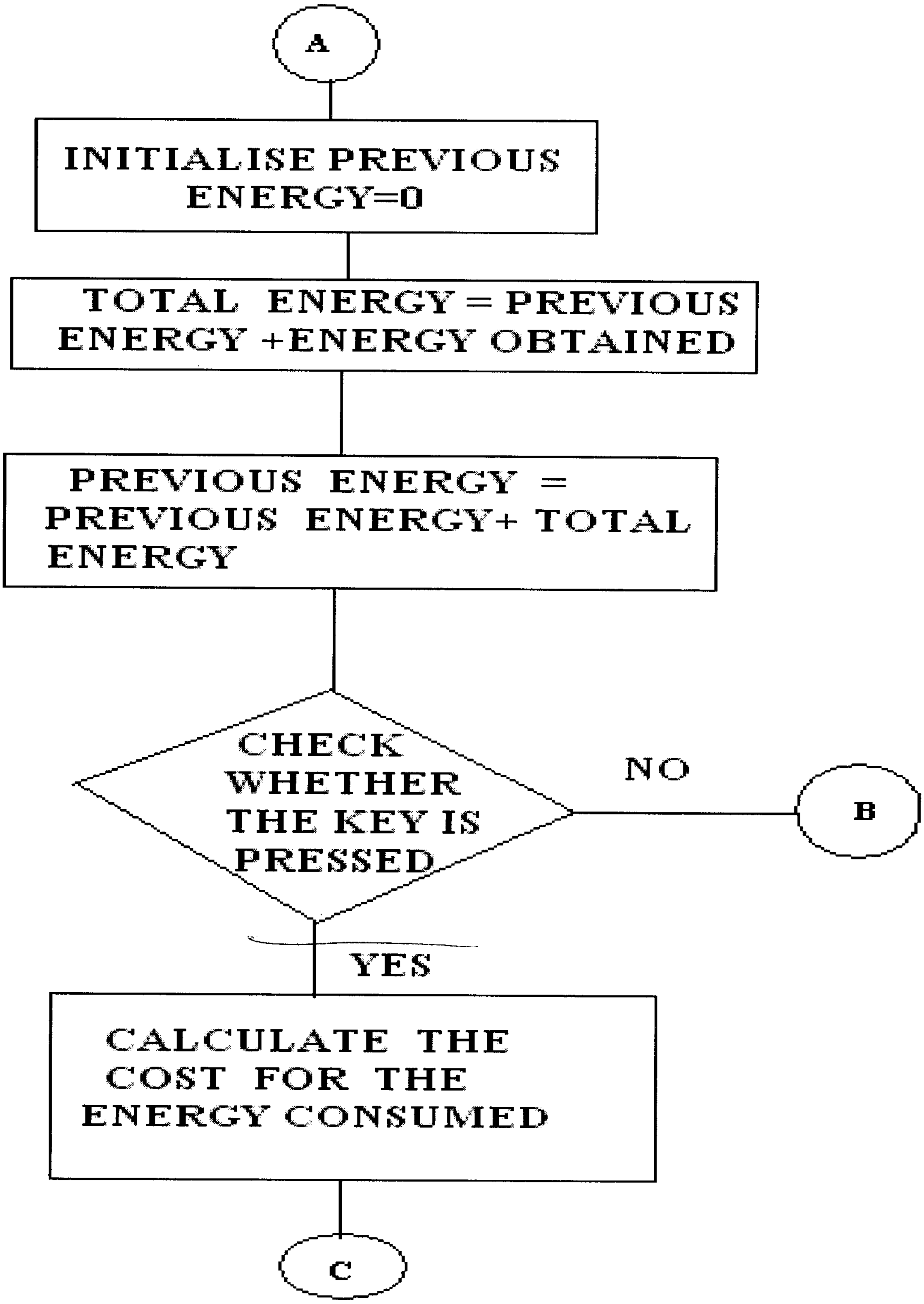
CHAPTER 6

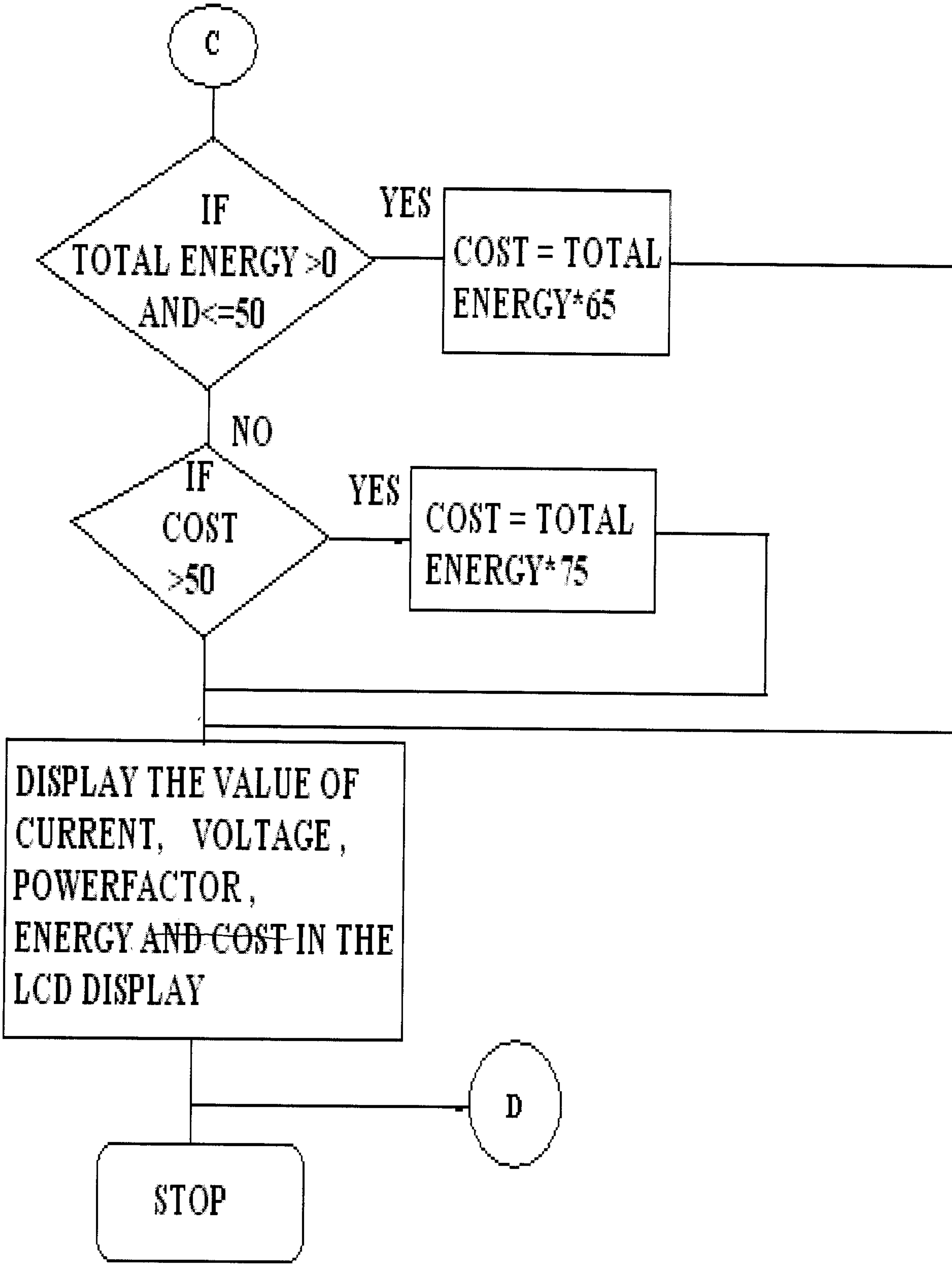
SOFTWARE

6.1 FLOW CHART:



START
D
B
GET THE VOLTAGE VALUE
GET THE CURRENT VALUE
GET THE VALUE OF POWERFACTOR
CALCULATE THE POWER USING THE FORMULA POWER = VOLTAGE*CURRENT*POWER FACTOR
CALCULATE THE ENERGY USING THE FORMULA ENERGY=POWER*TIME TAKEN
A


A
INITIALISE PREVIOUS ENERGY=0
TOTAL ENERGY = PREVIOUS ENERGY +ENERGY OBTAINED
PREVIOUS ENERGY = PREVIOUS ENERGY+ TOTAL ENERGY
CHECK WHETHER THE KEY IS PRESSED
NO
B
YES
CALCULATE THE COST FOR THE ENERGY CONSUMED
C


C
IF
TOTAL ENERGY >0
AND<=50
YES
COST = TOTAL
ENERGY*65
NO
IF
COST
>50
YES
COST = TOTAL
ENERGY*75
DISPLAY THE VALUE OF
CURRENT, VOLTAGE,
POWERFACTOR,
ENERGY ~~AND COST~~ IN THE
LCD DISPLAY
D
STOP

6.2 PROGRAM

```
#include<pic.h>       //V,I & PF in single chip PIC
#include<math.h>


static bit  pb1 @((unsigned)&PORTB*8+1);
static bit  pb2 @((unsigned)&PORTB*8+2);
static bit  pb4 @((unsigned)&PORTB*8+4);
static bit  pb5 @((unsigned)&PORTB*8+5);

static bit input @((unsigned)&PORTE*8+0);

#define  DISCTR    ((pb4=0),(pb2=1),(pb1=0),(pb5=0),(pb5=1))
#define DISDAT    ((pb4=0), (pb2=0), (pb1=1),(pb5=0),(pb5=1))

static bit   key2  @((unsigned) &PORTC*8+0);

void lcd_init();
void lcd_rd ();
void lcd_wr();
void lcd_disp(unsigned char);
void lcd_condis(const unsigned char*,unsigned int);

void hex_dec(unsigned char);
void hex_dec_curr(unsigned int);
void hex_dec_pf(int);
void hex_dec_pow(unsigned int);
void hex_dec_cost(unsigned int);
void hex_dec_ap(unsigned int);
void delay(unsigned int);

void ser_txn(unsigned char);
void ser_contxn(const unsigned char*,unsigned int);

void eeprom_wr(unsigned char,unsigned char);
char eeprom_rd(unsigned char);

unsigned char h,hr,t,o,ptt,pth,ph,pt,po,i;
unsigned char tkw,q,s,l,hi;
unsigned char dh1,dh2,b,count,pf;
unsigned int p,ene,kw,cost,enel;
unsigned int pttr,pthr,phr;

float x,y;
bit fac,m,k1,k2;

void interrupt  timer1(void)
{
 if(TMR1IF==1 && fac==1)    //for power factor cal
 {
  TMR1ON=0;
  TMR1IF=0;
  TRISC=0xf0;

  count++;
  TMR1H=0xff;
  TMR1L=0x9b;
  TMR1ON=1;
 }

 else if(TMR1IF==1 && fac==0)    //for time cal
 {
  TMR1ON=0;
  b++;
  if(b==15)       //for 1 sec
  {
   b=0;
   s++;
   if(s==10)      //for 10 sec
   {
    s=0;
    m=1;
   }
  }
  TMR1IF=0;
  TMR1ON=1;
 }
}

main()
 {
```

```
ADCON1=0x02;
TRISE=0x07;
TRISB=0x00;
count=0;
TRISD=0;
TRISC=0xff;
lcd_init();
lcd_condis("POWER MONITORING",16);
delay(10000);

eeprom_rd(0x00);
ptt=q;
eeprom_rd(0x01);          //read ENERGY from EEPROM
pth=q;
eeprom_rd(0x02);
ph=q;
eeprom_rd(0x03);
pt=q;
eeprom_rd(0x04);
po=q;
enel=(ptt*10000+pth*1000+ph*100+pt*10+po);

eeprom_rd(0x08);
ptt=q;
eeprom_rd(0x09);          //read ENERGY from EEPROM
pth=q;
eeprom_rd(0x0a);
```

```
ph=q;
eeprom_rd(0x0b);
pt=q;
eeprom_rd(0x0c);
po=q;
tkw=(ptt*10000+pth*1000+ph*100+pt*10+po);

GIE=1;
PEIE=1;
TMR1IE=1;
TMR1L=0x00;
TMR1H=0x00;
PORTD=0x01;
lcd_rd();

while(1)
{
 ADCON0=0x00;
 ADON=1;
 delay(255);
 ADCON0=0x05;
 while(ADCON0!=0x01);
 dh1=ADRESH;

 PORTD=0x80;
 lcd_rd();
 lcd_condis("V :",3);
```

```
hex_dec(dh1);

ser_contxn("Voltage:",8);
ser_txn(h+0x30);
ser_txn(t+0x30);
ser_txn(o+0x30);
ser_contxn("      ",6);

ADON=0;
ADCON0=0x08;
ADON=1;
delay(255);
ADCON0=0x0d;
while(ADCON0!=0x09);
dh2=ADRESH;

PORTD=0x87;
lcd_rd();
lcd_condis(" I:",3);
hex_dec_curr(dh2);

ser_contxn("Current:",8);
ser_txn(h+0x30);
ser_txn(t+0x30);
ser_txn('.');
ser_txn(o+0x30);
ser_contxn("      ",6);
```

```
GIE=1;
PEIE=1;
TMR1IE=1;
TMR1ON=0;
hi=TMR1H;
l=TMR1L;
TMR1H=0xff;
TMR1L=0x9b;
fac=1;
while(input==0);
TMR1ON=1;
while(input==1);
fac=0;

TMR1L=l;
TMR1H=hi;

x=count*1.8;
x=x*3.14;
x=x/180;
y=cos(x);
y=y*100;
pf=y;

PORTD=0xc0;
lcd_rd();
```

```
lcd_condis("PF:",3);
hex_dec_pf(pf);
count=0;

ser_contxn("PF:",3);
ser_txn(h+0x30);
ser_txn('.');
ser_txn(t+0x30);
ser_txn(o+0x30);
ser_contxn("      ",6);

p=dh1*dh2;
p=p/100;
p=p*pf;

if(m==1)
{
 m=0;
 p=p/360;
 ene=enel+p;
 enel=ene;

 PORTD=0xc7;
 lcd_rd();
 lcd_condis(" E:",3);
 hex_dec_pow(ene);
```

```
ser_contxn(" Energy:",8);
ser_txn(ptt+0x30);
ser_txn(pth+0x30);
ser_txn(ph+0x30);
ser_txn(pt+0x30);
ser_txn('.');
ser_txn(po+0x30);
ser_contxn("      ",6);

eeprom_wr(0x00,ptt);
eeprom_wr(0x01,pth);
eeprom_wr(0x02,ph);
eeprom_wr(0x03,pt);
eeprom_wr(0x04,po);

if(ene>=10000)
{
 kw=ene/10000;
 tkw=tkw+kw;
 enel=ene%10000;

 PORTD=0xc7;
 lcd_rd();
 lcd_condis(" E:",3);
 hex_dec_pow(ene);

 hex_dec_ap(tkw);
```

```
  eeprom_wr(0x08,ptt);
  eeprom_wr(0x09,pth);
  eeprom_wr(0x0a,ph);
  eeprom_wr(0x0b,pt);
  eeprom_wr(0x0c,po);
 }
}
if(key2==0 && k2==0) k2=1;
if(key2==1 && k2==1)
{
 k2=0;
 if(tkw>0 && tkw<=50 )
 {
  PORTD=0x01;
  lcd_rd();
  PORTD=0x80;
  lcd_rd();
  lcd_condis("UNITS:",6);
  hex_dec_ap(tkw);
  lcd_disp(ptt+0x30);
  lcd_disp(pth+0x30);
  lcd_disp(ph+0x30);
  lcd_disp(pt+0x30);
  lcd_disp(po+0x30);

 cost=tkw*65;
 PORTD=0xc0;
 lcd_rd();
 hex_dec_cost(cost);

 ser_contxn(" Cost:",6);
 ser_txn(ptt+0x30);
 ser_txn(pth+0x30);
 ser_txn(ph+0x30);
 ser_txn('.');
 ser_txn(pt+0x30);
 ser_txn(po+0x30);
 ser_contxn("      ",6);

 delay(50000);
 delay(50000);
 delay(50000);
 delay(50000);
 PORTD=0x01;
 lcd_rd();
}
else if(tkw>50)
{
 PORTD=0x01;
 lcd_rd();
 PORTD=0x80;
 lcd_rd();
 lcd_condis("UNITS:",6);
 hex_dec_ap(tkw);
 lcd_disp(ptt+0x30);
 lcd_disp(pth+0x30);
 lcd_disp(ph+0x30);
 lcd_disp(pt+0x30);
 lcd_disp(po+0x30);

 cost=tkw*75;
 PORTD=0xc0;
 lcd_rd();
 hex_dec_cost(cost);

 ser_contxn(" Cost:",6);
 ser_txn(ptt+0x30);
 ser_txn(pth+0x30);
 ser_txn(ph+0x30);
 ser_txn('.');
 ser_txn(pt+0x30);
 ser_txn(po+0x30);
 ser_contxn("      ",6);

 delay(50000);
 delay(50000);
 delay(50000);
 delay(50000);
 PORTD=0x01;
 lcd_rd();
}
```

```
   }
  }
 }

void lcd_init()
{
 TRISD=0;
 PORTD=0x38;
 lcd_rd();
 PORTD=0x0c;
 lcd_rd();
 PORTD=0x06;
 lcd_rd();
 PORTD=0x01;
 lcd_rd();
 PORTD=0x80;
 lcd_rd();
}

void lcd_disp(unsigned char lr)
{
 PORTD=lr;
 lcd_wr();
}

void lcd_condis(const unsigned char *word,unsigned int n)
{
```

```
for(i=0;i<n;i++)
{
 PORTD=word[i];
 lcd_wr();
}
}

void lcd_rd()
{
 DISDAT;                    //send data to lcd data lines
 PORTD=0x04;                   //to select rs,rw=0;en=1
 DISCTR;                    //send data to ctrl lines
 delay(125);
 PORTD=0x00;                   //t0 select rs,re,en=0
 DISCTR;                    //send data to ctrl lines
}

void lcd_wr()
{
 DISDAT;                    //send data to lcd data lines
 PORTD=0x05;                   //to select rs=1,rw =0,en=1;
 DISCTR;                    //send data to ctrl lines
 delay(125);
 PORTD=0x01;                   //to select rs=1,rw=0,en=0;
 DISCTR;                    //send data to ctrl lines
}
```

```
void hex_dec(unsigned char val)
{
 h=val/100;
 hr=val%100;
 t=hr/10;
 o=hr%10;

 lcd_disp(h+0x30);
 lcd_disp(t+0x30);
 lcd_disp(o+0x30);
}

void hex_dec_curr(unsigned int vai)
{
 h=vai/100;
 hr=vai%100;
 t=hr/10;
 o=hr%10;

 lcd_disp(h+0x30);
 lcd_disp(t+0x30);
 lcd_disp('.');
 lcd_disp(o+0x30);
}

void hex_dec_cost(unsigned int pv)
{
```

```
//  unsigned int pttr,pthr,phr;

 ptt=pv/10000;
 pttr=pv%10000;
 pth=pttr/1000;
 pthr=pttr%1000;
 ph=pthr/100;
 phr=pthr%100;
 pt=phr/10;
 po=phr%10;

 lcd_disp(ptt+0x30);
 lcd_disp(pth+0x30);
 lcd_disp(ph+0x30);
 lcd_disp('.');
 lcd_disp(pt+0x30);
 lcd_disp(po+0x30);
}

void hex_dec_pow(unsigned int pv)
{
 unsigned int pttr,pthr,phr;

ptt=pv/10000;
pttr=pv%10000;
pth=pttr/1000;
pthr=pttr%1000;
```

```
 ph=pthr/100;
 phr=pthr%100;
 pt=phr/10;
 po=phr%10;

 lcd_disp(ptt+0x30);
 lcd_disp(pth+0x30);
 lcd_disp(ph+0x30);
 lcd_disp(pt+0x30);
 lcd_disp('.');
 lcd_disp(po+0x30);
}

void hex_dec_ap(unsigned int pv)
{
 ptt=pv/10000;
 pttr=pv%10000;
 pth=pttr/1000;
 pthr=pttr%1000;
 ph=pthr/100;
 phr=pthr%100;
 pt=phr/10;
 po=phr%10;
}

void hex_dec_pf(int pfval)
{
 h=pfval/100;
 hr=pfval%100;
 t=hr/10;
 o=hr%10;

 lcd_disp(h+0x30);
 lcd_disp('.');
 lcd_disp(t+0x30);
 lcd_disp(o+0x30);
}

void eeprom_wr(unsigned char addr,unsigned char wlr)
{
 GIE=0;          //desable all interrupts
 EEPGD=0;          //access EEPROM data memory
 EEADR=addr;       //select memory location to store setpoint
 EEDATA=wlr;
 WREN=1;
 EECON2=0x55;      //for enable write cycle
 EECON2=0xaa;      //      ,,
 WR=1;           //start write operation
 while(!EEIF);    //wait until write is finished
 EEIF=0;
 WREN=0;
 GIE=1;
}
```

```
char eeprom_rd(unsigned char radd)
{
 EEADR=radd;      //select memory location to store setpoint
 EEPGD=0;         //access EEPROM data memory
 RD=1;
 q=EEDATA;        //read the setpoint
 return(q);
}

void ser_txn(unsigned char te)
{
  SPBRG=51;        //for 1.2 kb baud rate
  BRGH=0;          //for low baud rate
  SYNC=0;          //asynchronous mode
  SPEN=1;          //Enable the serial port
//  TX9=1;         //enable the ninth data bit
  TXEN=1;          //enable TXion
  TXREG=te;
  delay(2500);
  TXIF=0;
}

void ser_contxn(const unsigned char *dat,unsigned int m)
{
unsigned int j;

for(j=0;j<m;j++)
```

```
 {
  TXREG=dat[j];
  delay(1500);
 }
}


void delay(unsigned int del)
{
 while(del--);
}
```

CONCLUSION

CHAPTER 7

CONCLUSION

Thus the system calculates the energy utilization of a consumer over a certain period and its corresponding cost is calculated according to the tariff imposed. The system also monitors the electrical parameters like voltage, current, power factor, power, energy and also the actual cost of consumption bypassing the need for manual calculation. The system is interfaced with the personal computer and the data's are transferred to the computer through serial communication.

7.1 FUTURE ENHANCEMENT

1. ATTACHING A PRINTER FOR BILLING:

A printer can be attached to enable direct printing of the electrical parameters like voltage, current, power, energy, no of units consumed, cost and even the meter number and the EB division.

2. CUTTING THE LOADS AUTOMATICALLY:

Various luxury electrical appliances which consume high power can be cut off at predetermined rates with the help of relays interfaced the micro controller.

APPENDIX

PIC16F87X

28/40-pin 8-Bit CMOS FLASH Microcontrollers

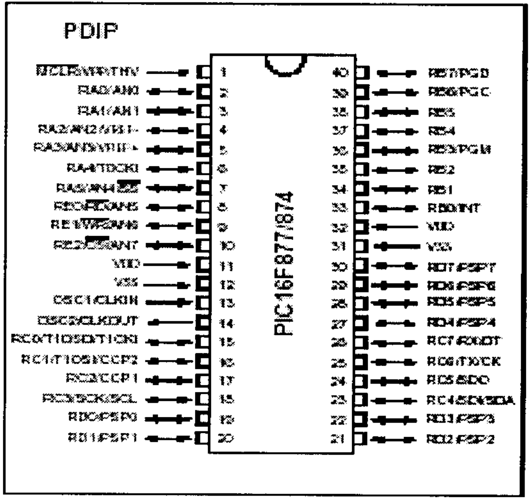
Devices Included in this Data Sheet:

- PIC16F873
- PIC16F874
- PIC16F876
- PIC16F877

Microcontroller Core Features:

- 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
- Commercial and Industrial temperature ranges
- Low-power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

Pin Diagram



Peripheral Features:

- 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 I^2C™ (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)

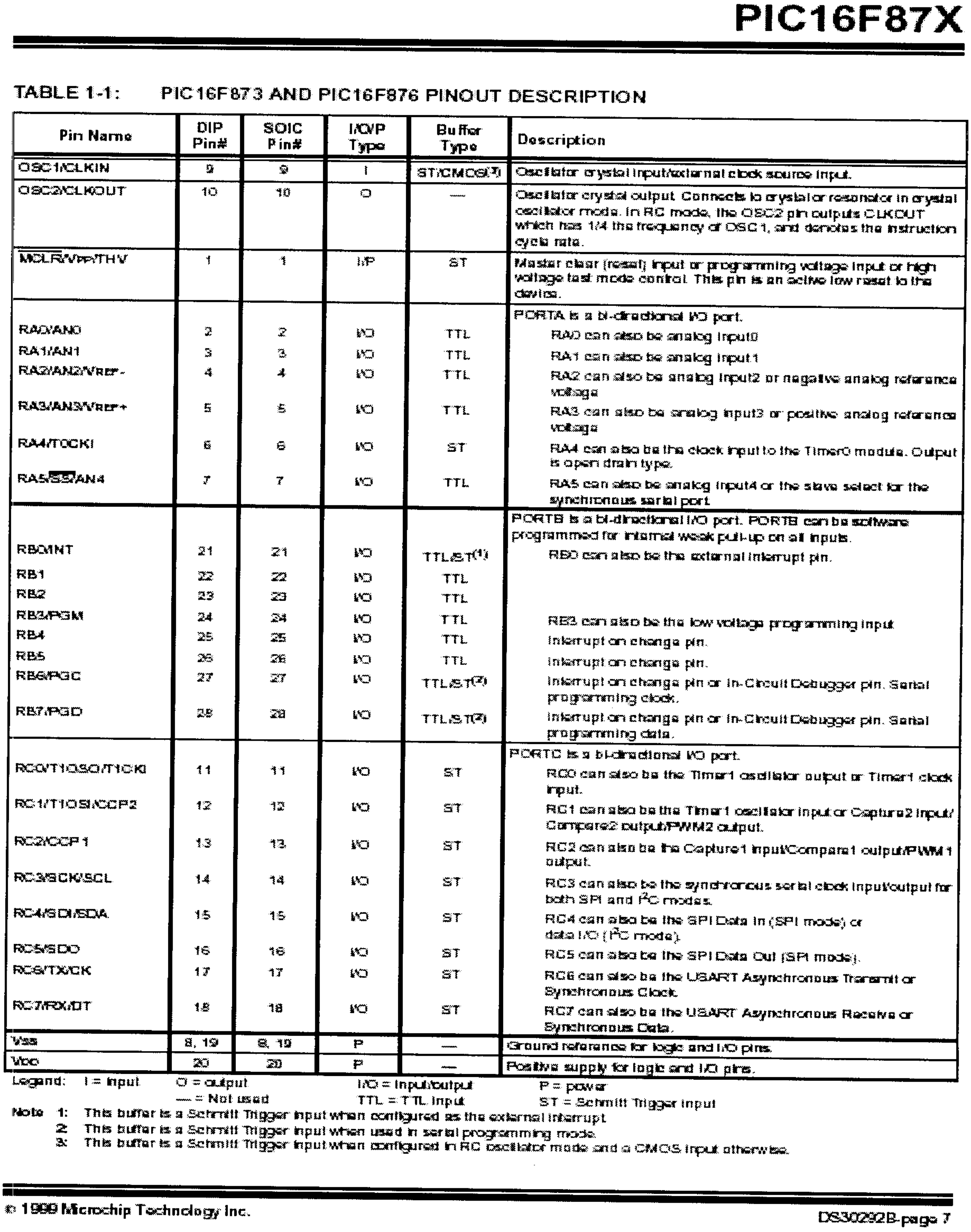


TABLE 1-1: PIC16F873 AND PIC16F876 PINOUT DESCRIPTION

Pin Name	DIP Pin#	SOIC Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	9	I	ST/CMOS[3]	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	10	10	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{\text{MCLR}}$/V_{PP}/THV	1	1	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
					PORTA is a bi-directional I/O port.
RA0/AN0	2	2	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	3	I/O	TTL	RA1 can also be analog input1
RA2/AN2/V_{REF-}	4	4	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/V_{REF+}	5	5	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	6	I/O	ST	RA4 can also be the clock input to the Timer0 module. Output is open drain type.
RA5/$\overline{\text{SS}}$/AN4	7	7	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
					PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	21	21	I/O	TTL/ST[1]	RB0 can also be the external interrupt pin.
RB1	22	22	I/O	TTL	
RB2	23	23	I/O	TTL	
RB3/PGM	24	24	I/O	TTL	RB3 can also be the low voltage programming input.
RB4	25	25	I/O	TTL	Interrupt on change pin.
RB5	26	26	I/O	TTL	Interrupt on change pin.
RB6/PGC	27	27	I/O	TTL/ST[2]	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	28	28	I/O	TTL/ST[2]	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
					PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	11	11	I/O	ST	RC0 can also be the Timer1 oscillator output or Timer1 clock input.
RC1/T1OSI/CCP2	12	12	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	13	13	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	14	14	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	15	15	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	16	16	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	17	17	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	18	18	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.
V_{SS}	8, 19	8, 19	P	—	Ground reference for logic and I/O pins.
V_{DD}	20	20	P	—	Positive supply for logic and I/O pins.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS$^{(4)}$	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{\text{MCLR}}$/VPP/THV	1	2	18	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	3	19	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	4	20	I/O	TTL	RA1 can also be analog input1
RA2/AN2/V_{REF-}	4	5	21	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/V_{REF+}	5	6	22	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	7	23	I/O	ST	RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type.
RA5/$\overline{\text{SS}}$/AN4	7	8	24	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	I/O	TTL/ST$^{(1)}$	RB0 can also be the external interrupt pin.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	RB3 can also be the low voltage programming input
RB4	37	41	14	I/O	TTL	Interrupt on change pin.
RB5	38	42	15	I/O	TTL	Interrupt on change pin.
RB6/PGC	39	43	16	I/O	TTL/ST$^{(2)}$	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	40	44	17	I/O	TTL/ST$^{(2)}$	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
						PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^{2}C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^{2}C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

TABLE 1-2: PIC16F874 AND PIC16F877 PINOUT DESCRIPTION

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS(4)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{\text{MCLR}}$/V_{PP}/THV	1	2	18	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	3	19	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	4	20	I/O	TTL	RA1 can also be analog input1
RA2/AN2/V_{REF-}	4	5	21	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/V_{REF+}	5	6	22	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	7	23	I/O	ST	RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type.
RA5/$\overline{\text{SS}}$/AN4	7	8	24	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	I/O	TTL/ST(1)	RB0 can also be the external interrupt pin.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	RB3 can also be the low voltage programming input
RB4	37	41	14	I/O	TTL	Interrupt on change pin.
RB5	38	42	15	I/O	TTL	Interrupt on change pin.
RB6/PGC	39	43	16	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	40	44	17	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
						PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

FIGURE 2-3: PIC16F877/876 REGISTER FILE MAP

							File Address
Indirect addr.(*)	00h	Indirect addr.(*)	80h	Indirect addr.(*)	100h	Indirect addr.(*)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	▨	105h	▨	185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h	▨	107h	▨	187h
PORTD (1)	08h	TRISD (1)	88h	▨	108h	▨	188h
PORTE (1)	09h	TRISE (1)	89h	▨	109h	▨	189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATA	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADR	10Dh	EECON2	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved(2)	18Eh
TMR1H	0Fh	▨	8Fh	EEADRH	10Fh	Reserved(2)	18Fh
T1CON	10h	▨	90h	General Purpose Register 16 Bytes	110h	General Purpose Register 16 Bytes	190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	▨	95h		115h		195h
CCPR1H	16h	▨	96h		116h		196h
CCP1CON	17h	▨	97h		117h		197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah	▨	9Ah		11Ah		19Ah
CCPR2L	1Bh	▨	9Bh		11Bh		19Bh
CCPR2H	1Ch	▨	9Ch		11Ch		19Ch
CCP2CON	1Dh	▨	9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h
			EFh		16Fh		1EFh
		accesses 70h-7Fh	F0h	accesses 70h-7Fh	170h	accesses 70h - 7Fh	1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

▨ Unimplemented data memory locations, read as '0'.
* Not a physical register.

Note 1: These registers are not implemented on 28-pin devices.

2: These registers are reserved, maintain these registers clear.

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 2-1.

The Special Function Registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in the peripheral feature section.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 0											
00h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(4)	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
04h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h(5)	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h(5)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(3)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	(6)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON	0000 00-0	0000 00-0

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

TABLE 2-1: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 2											
100h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
101h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
102h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
103h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
104h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
10Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Ch	EEDATA	EEPROM data register								xxxx xxxx	uuuu uuuu
10Dh	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM data register high byte						xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM address register high byte					xxxx xxxx	uuuu uuuu
Bank 3											
180h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
183h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
184h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
18Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM control register2 (not a physical register)								---- ----	---- ----
18Eh	—	Reserved maintain clear								0000 0000	0000 0000
18Fh	—	Reserved maintain clear								0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved.
Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

2.2.2.5 PIR1 REGISTER

The PIR1 register contains the individual flag bits for the peripheral interrupts.

> **Note:** Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt bits are clear prior to enabling an interrupt.

REGISTER 2-5: PIR1 REGISTER (ADDRESS 0Ch)

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF[1]	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

R = Readable bit
W = Writable bit
- n= Value at POR reset

bit 7: PSPIF[1]: Parallel Slave Port Read/Write Interrupt Flag bit
1 = A read or a write operation has taken place (must be cleared in software)
0 = No read or write has occurred

bit 6: ADIF: A/D Converter Interrupt Flag bit
1 = An A/D conversion completed
0 = The A/D conversion is not complete

bit 5: RCIF: USART Receive Interrupt Flag bit
1 = The USART receive buffer is full
0 = The USART receive buffer is empty

bit 4: TXIF: USART Transmit Interrupt Flag bit
1 = The USART transmit buffer is empty
0 = The USART transmit buffer is full

bit 7: SSPIF: Synchronous Serial Port (SSP) Interrupt Flag
1 = The SSP interrupt condition has occurred, and must be cleared in software before returning from the interrupt service routine. The conditions that will set this bit are:
SPI
A transmission/reception has taken place.
I^2C Slave
A transmission/reception has taken place.
I^2C Master
A transmission/reception has taken place.
The initiated start condition was completed by the SSP module.
The initiated stop condition was completed by the SSP module.
The initiated restart condition was completed by the SSP module.
The initiated acknowledge condition was completed by the SSP module.
A start condition occurred while the SSP module was idle (Multimaster system).
A stop condition occurred while the SSP module was idle (Multimaster system).
0 = No SSP interrupt condition has occurred.

bit 2: CCP1IF: CCP1 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused in this mode

bit 1: TMR2IF: TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0: TMR1IF: TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

Note 1: PSPIF is reserved on 28-pin devices; always maintain this bit clear.

3.0 I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

3.1 PORTA and the TRISA Register

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, the value is modified and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other PORTA pins have TTL input levels and full CMOS output drivers.

Other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

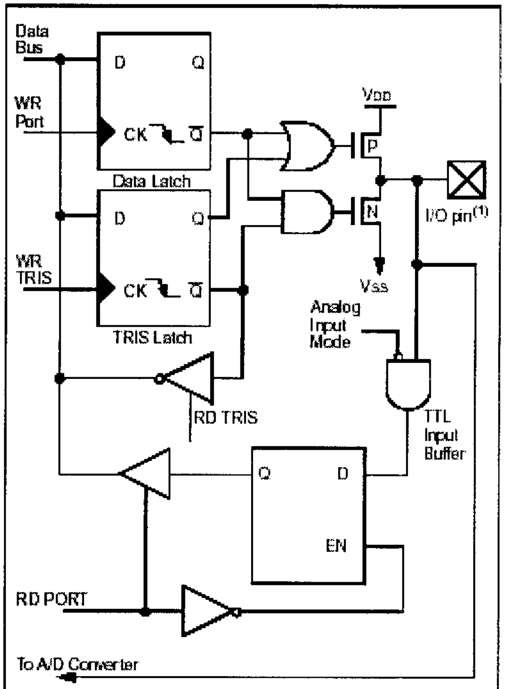
Note:	On a Power-on Reset, these pins are configured as analog inputs and read as '0'.

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 3-1: INITIALIZING PORTA

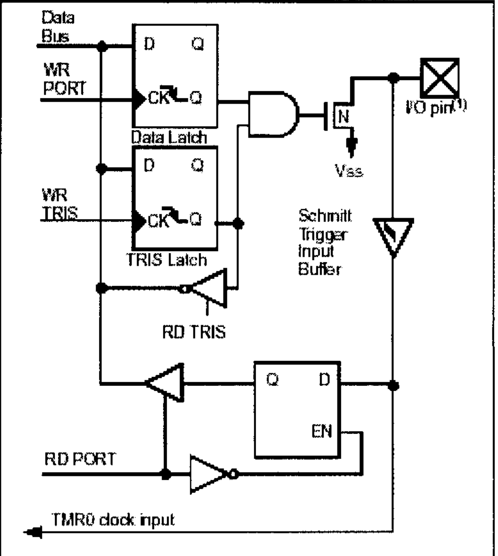
```
BCF     STATUS, RP0    ;
BCF     STATUS, RP1    ; Bank0
CLRF    PORTA          ; Initialize PORTA by
                       ; clearing output
                       ; data latches
BSF     STATUS, RP0    ; Select Bank 1
MOVLW   0x06           ; Configure all pins
MOVWF   ADCON1         ; as digital inputs
MOVLW   0xCF           ; Value used to
                       ; initialize data
                       ; direction
MOVWF   TRISA          ; Set RA<3:0> as inputs
                       ; RA<5:4> as outputs
                       ; TRISA<7:6> are always
                       ; read as '0'.
```

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS



Note 1: I/O pins have protection diodes to VDD and VSS.

FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



Note 1: I/O pin has protection diodes to VSS only.

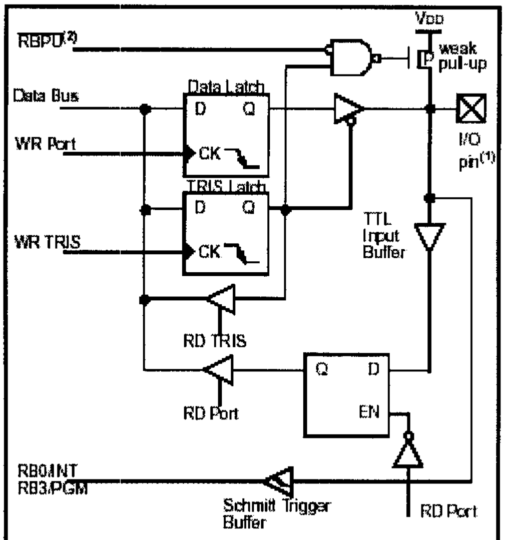
3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

Three pins of PORTB are multiplexed with the Low Voltage Programming function; RB3/PGM, RB6/PGC and RB7/PGD. The alternate functions of these pins are described in the Special Features Section.

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBPU}}$ (OPTION_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.

b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

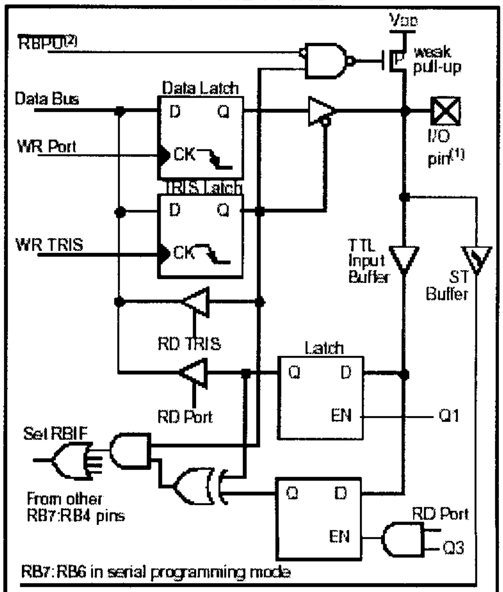
The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

This interrupt on mismatch feature, together with software configureable pull-ups on these four pins, allow easy interface to a keypad and make it possible for wake-up on key-depression. Refer to the Embedded Control Handbook, *"Implementing Wake-Up on Key Stroke"* (AN552).

RB0/INT is an external interrupt input pin and is configured using the INTEDG bit (OPTION_REG<6>).

RB0/INT is discussed in detail in Section 12.10.1.

FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS



Note 1: I/O pins have diode protection to VDD and VSS.
2: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

Note: When using Low Voltage ICSP Programming (LVP) and the pull-ups on PORTB are enabled, bit 3 in the TRISB register must be cleared to disable the pull-up on RB3 and ensure the proper operation of the device.

3.5 PORTE and TRISE Register

This section is not applicable to the PIC16F873 or PIC16F876.

PORTE has three pins, RE0/$\overline{\text{RD}}$/AN5, RE1/$\overline{\text{WR}}$/AN6 and RE2/$\overline{\text{CS}}$/AN7, which are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers.

I/O PORTE becomes control inputs for the microprocessor port when bit PSPMODE (TRISE<4>) is set. In this mode, the user must make sure that the TRISE<2:0> bits are set (pins are configured as digital inputs). Ensure ADCON1 is configured for digital I/O. In this mode, the input buffers are TTL.

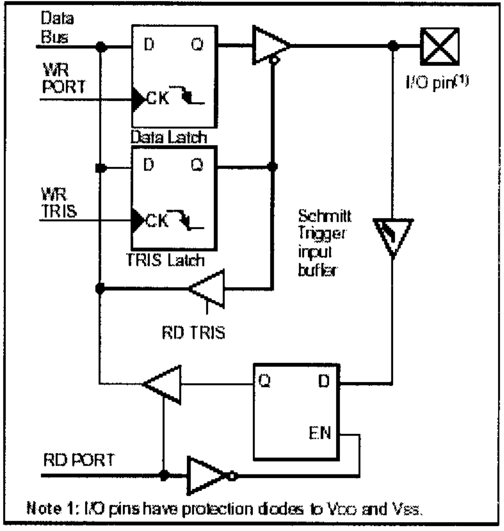
Register 3-1 shows the TRISE register, which also controls the parallel slave port operation.

PORTE pins are multiplexed with analog inputs. When selected as an analog input, these pins will read as '0's.

TRISE controls the direction of the RE pins, even when they are being used as analog inputs. The user must make sure to keep the pins configured as inputs when using them as analog inputs.

Note: On a Power-on Reset, these pins are configured as analog inputs.

FIGURE 3-8: PORTE BLOCK DIAGRAM (IN I/O PORT MODE)



Note 1: I/O pins have protection diodes to VDD and VSS.

REGISTER 3-1: TRISE REGISTER (ADDRESS 89h)

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1
IBF	OBF	IBOV	PSPMODE	—	bit2	bit1	bit0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n= Value at POR reset

Parallel Slave Port Status/Control Bits

bit 7 : **IBF**: Input Buffer Full Status bit
1 = A word has been received and is waiting to be read by the CPU
0 = No word has been received

bit 6: **OBF**: Output Buffer Full Status bit
1 = The output buffer still holds a previously written word
0 = The output buffer has been read

bit 5: **IBOV**: Input Buffer Overflow Detect bit (in microprocessor mode)
1 = A write occurred when a previously input word has not been read (must be cleared in software)
0 = No overflow occurred

bit 4: **PSPMODE**: Parallel Slave Port Mode Select bit
1 = Parallel slave port mode
0 = General purpose I/O mode

bit 3: **Unimplemented**: Read as '0'

PORTE Data Direction Bits

bit 2: **Bit2**: Direction Control bit for pin RE2/$\overline{\text{CS}}$/AN7
1 = Input
0 = Output

bit 1: **Bit1**: Direction Control bit for pin RE1/$\overline{\text{WR}}$/AN6
1 = Input
0 = Output

bit 0: **Bit0**: Direction Control bit for pin RE0/$\overline{\text{RD}}$/AN5
1 = Input
0 = Output

4.0 DATA EEPROM AND FLASH PROGRAM MEMORY

The Data EEPROM and FLASH Program Memory are readable and writable during normal operation over the entire VDD range. A bulk erase operation may not be issued from user code (which includes removing code protection). The data memory is not directly mapped in the register file space. Instead it is indirectly addressed through the Special Function Registers (SFR).

There are six SFRs used to read and write the program and data EEPROM memory. These registers are:

- EECON1
- EECON2
- EEDATA
- EEDATH
- EEADR
- EEADRH

The EEPROM data memory allows byte read and write. When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and EEADR holds the address of the EEPROM location being accessed. The registers EEDATH and EEADRH are not used for data EEPROM access. These devices have up to 256 bytes of data EEPROM with an address range from 0h to FFh.

The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature, as well as from chip-to-chip. Please refer to the specifications for exact limits.

The program memory allows word reads and writes. Program memory access allows for checksum calculation and calibration table storage. A byte or word write automatically erases the location and writes the new data (erase before write). Writing to program memory will cease operation until the write is complete. The program memory cannot be accessed during the write, therefore code cannot execute. During the write operation, the oscillator continues to clock the peripherals, and therefore they continue to operate. Interrupt events will be detected and essentially "queued" until the write is completed. When the write completes, the next instruction in the pipeline is executed and the branch to the interrupt vector address will occur.

When interfacing to the program memory block, the EEDATH:EEDATA registers form a two byte word, which holds the 14-bit data for read/write. The EEADRH:EEADR registers form a two byte word, which holds the 13-bit address of the EEPROM location being accessed. These devices can have up to 8K words of program EEPROM with an address range from 0h to 3FFFh. The unused upper bits in both the EEDATH and EEDATA registers all read as '0's'.

The value written to program memory does not need to be a valid instruction. Therefore, up to 14-bit numbers can be stored in memory for use as calibration parameters, serial numbers, packed 7-bit ASCII, etc. Executing a program memory location containing data that forms an invalid instruction results in a NOP.

4.1 EEADR

The address registers can address up to a maximum of 256 bytes of data EEPROM or up to a maximum of 8K words of program FLASH.

When selecting a program address value, the MSByte of the address is written to the EEADRH register and the LSByte is written to the EEADR register. When selecting a data address value, only the LSByte of the address is written to the EEADR register.

On the PIC16F873/874 devices with 128 bytes of EEPROM, the MSbit of the EEADR must always be cleared to prevent inadvertent access to the wrong location. This also applies to the program memory. The upper MSbits of EEADRH must always be clear.

4.2 EECON1 and EECON2 Registers

EECON1 is the control register for memory accesses.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the memory write sequence.

Control bit EEPGD determines if the access will be a program or a data memory access. When clear, any subsequent operations will operate on the data memory. When set, any subsequent operations will operate on the program memory.

Control bits RD and WR initiate read and write operations, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at the completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental or premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a $\overline{MCLR}$ reset or a WDT time-out reset during normal operation. In these situations, following reset, the user can check the WRERR bit and rewrite the location. The value of the data and address registers and the EEPGD bit remains unchanged.

Interrupt flag bit EEIF, in the PIR2 register, is set when write is complete. It must be cleared in software.

4.8 Protection Against Spurious Write

4.8.1 EEPROM DATA MEMORY

There are conditions when the device may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built-in. On power-up, the WREN bit is cleared. Also, the Power-up Timer (72 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during brown-out, power glitch, or software malfunction.

4.8.2 PROGRAM FLASH MEMORY

To protect against spurious writes to FLASH program memory, the WRT bit in the configuration word may be programmed to '0' to prevent writes. The write initiate sequence must also be followed. WRT and the configuration word cannot be programmed by user code, only through the use of an external programmer.

4.9 Operation during Code Protect

Each reprogrammable memory block has its own code protect mechanism. External Read and Write operations are disabled if either of these mechanisms are enabled.

4.9.1 DATA EEPROM MEMORY

The microcontroller itself can both read and write to the internal Data EEPROM, regardless of the state of the code protect configuration bit.

4.9.2 PROGRAM FLASH MEMORY

The microcontroller can read and execute instructions out of the internal FLASH program memory, regardless of the state of the code protect configuration bits. However the WRT configuration bit and the code protect bits have different effects on writing to program memory. Table 4-1 shows the various configurations and status of reads and writes. To erase the WRT or code protection bits in the configuration word requires that the device be fully erased.

TABLE 4-1: READ/WRITE STATE OF INTERNAL FLASH PROGRAM MEMORY

Configuration Bits			Memory Location	Internal Read	Internal Write	ICSP Read	ICSP Write
CP1	CP0	WRT					
0	0	x	All program memory	Yes	No	No	No
0	1	0	Unprotected areas	Yes	No	Yes	No
0	1	0	Protected areas	Yes	No	No	No
0	1	1	Unprotected areas	Yes	Yes	Yes	No
0	1	1	Protected areas	Yes	No	No	No
1	0	0	Unprotected areas	Yes	No	Yes	No
1	0	0	Protected areas	Yes	No	No	No
1	0	1	Unprotected areas	Yes	Yes	Yes	No
1	0	1	Protected areas	Yes	No	No	No
1	1	0	All program memory	Yes	No	Yes	Yes
1	1	1	All program memory	Yes	Yes	Yes	Yes

TABLE 4-2: REGISTERS ASSOCIATED WITH DATA EEPROM/PROGRAM FLASH

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Dh	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM address high					xxxx xxxx	uuuu uuuu
10Ch	EEDATA	EEPROM data resister								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM data resister high						xxxx xxxx	uuuu uuuu
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM control resister2 (not a physical resister)									
8Dh	PIE2	—	(1)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
0Dh	PIR2	—	(1)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0

Legend: x = unknown, u = unchanged, r = reserved, - = unimplemented read as '0'. Shaded cells are not used during FLASH/EEPROM access.

Note 1: These bits are reserved; always maintain these bits clear.

5.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 5-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

Additional information on the Timer0 module is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

Timer mode is selected by clearing bit T0CS (OPTION_REG<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

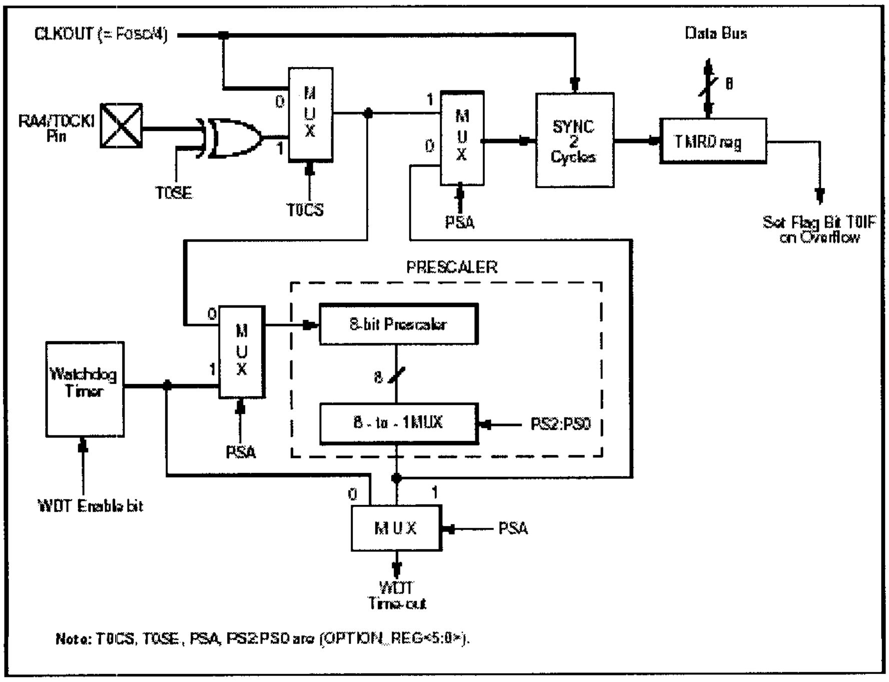
Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In counter mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 5.2.

The prescaler is mutually exclusively shared between the Timer0 module and the watchdog timer. The prescaler is not readable or writable. Section 5.3 details the operation of the prescaler.

5.1 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP.

FIGURE 5-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note: T0CS, T0SE, PSA, PS2:PS0 are (OPTION_REG<5:0>).

5.2 Using Timer0 with an External Clock

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

5.3 Prescaler

There is only one prescaler available, which is mutually exclusively shared between the Timer0 module and the watchdog timer. A prescaler assignment for the Timer0 module means that there is no prescaler for the watchdog timer, and vice-versa. This prescaler is not readable or writable (see Figure 5-1).

The PSA and PS2:PS0 bits (OPTION_REG<3:0>) determine the prescaler assignment and prescale ratio.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. `CLRF 1`, `MOVWF 1`, `BSF 1,x`....etc.) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

Note:	Writing to TMR0, when the prescaler is assigned to Timer0, will clear the prescaler count, but will not change the prescaler assignment.

REGISTER 5-1: OPTION_REG REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: $\overline{\text{RBPU}}$

bit 6: INTEDG

bit 5: T0CS: TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

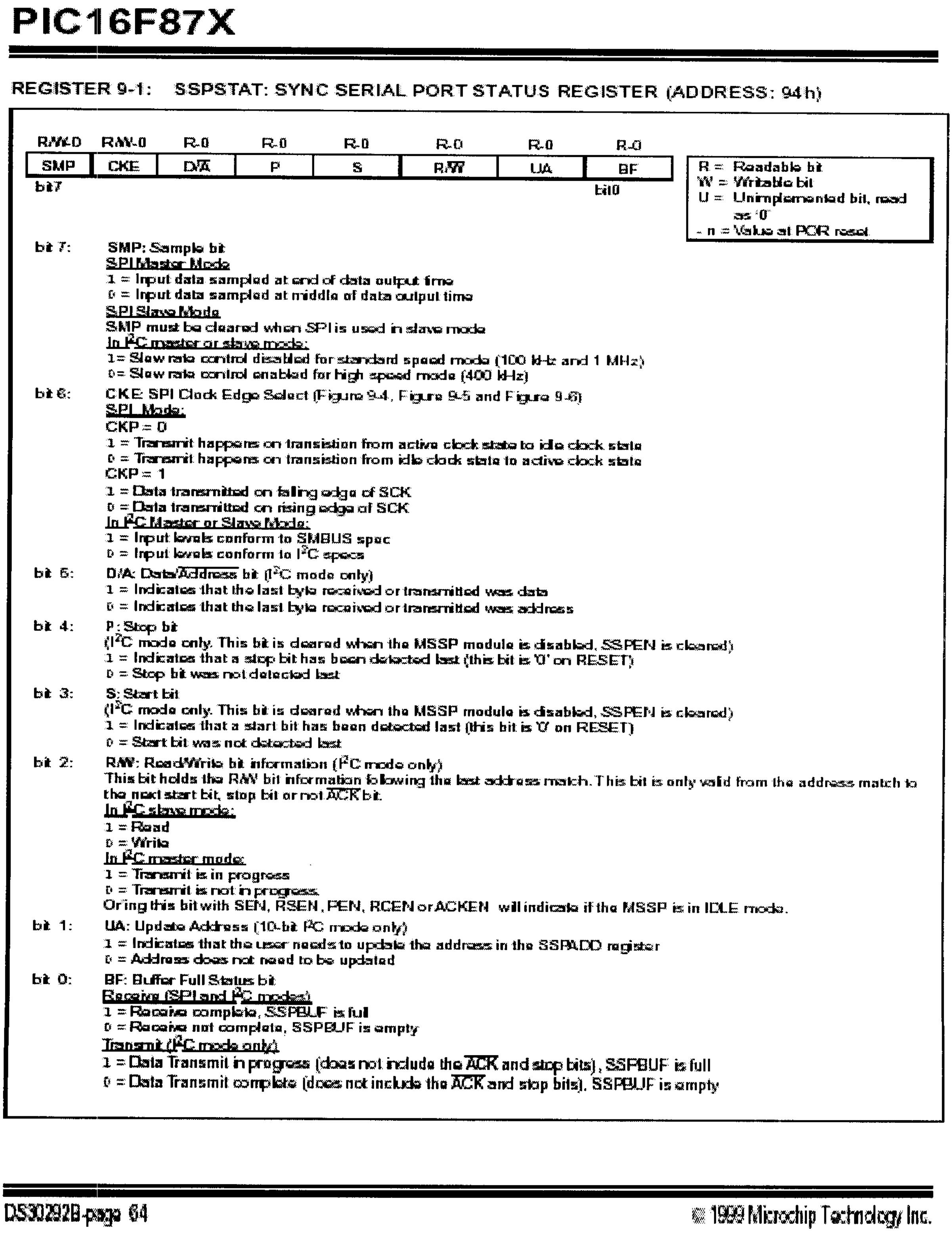
bit 4: T0SE: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3: PSA: Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0: PS2:PS0: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note:	To avoid an unintended device RESET, the instruction sequence shown in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.



REGISTER 9-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS: 94h)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: SMP: Sample bit
SPI Master Mode
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave Mode
SMP must be cleared when SPI is used in slave mode
In I^{2}C master or slave mode:
1= Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)
0= Slew rate control enabled for high speed mode (400 kHz)

bit 6: CKE: SPI Clock Edge Select (Figure 9-4, Figure 9-5 and Figure 9-6)
SPI Mode:
CKP = 0
1 = Transmit happens on transistion from active clock state to idle clock state
0 = Transmit happens on transistion from idle clock state to active clock state
CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
In I^{2}C Master or Slave Mode:
1 = Input levels conform to SMBUS spec
0 = Input levels conform to I^{2}C specs

bit 5: D/A: Data/$\overline{\text{Address}}$ bit (I^{2}C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4: P: Stop bit
(I^{2}C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared)
1 = Indicates that a stop bit has been detected last (this bit is '0' on RESET)
0 = Stop bit was not detected last

bit 3: S: Start bit
(I^{2}C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared)
1 = Indicates that a start bit has been detected last (this bit is '0' on RESET)
0 = Start bit was not detected last

bit 2: R/W: Read/Write bit information (I^{2}C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next start bit, stop bit or not $\overline{\text{ACK}}$ bit.
In I^{2}C slave mode:
1 = Read
0 = Write
In I^{2}C master mode:
1 = Transmit is in progress
0 = Transmit is not in progress.
Oring this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSP is in IDLE mode.

bit 1: UA: Update Address (10-bit I^{2}C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0: BF: Buffer Full Status bit
Receive (SPI and I^{2}C modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^{2}C mode only)
1 = Data Transmit in progress (does not include the $\overline{\text{ACK}}$ and stop bits), SSPBUF is full
0 = Data Transmit complete (does not include the $\overline{\text{ACK}}$ and stop bits), SSPBUF is empty

REGISTER 9-2: SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

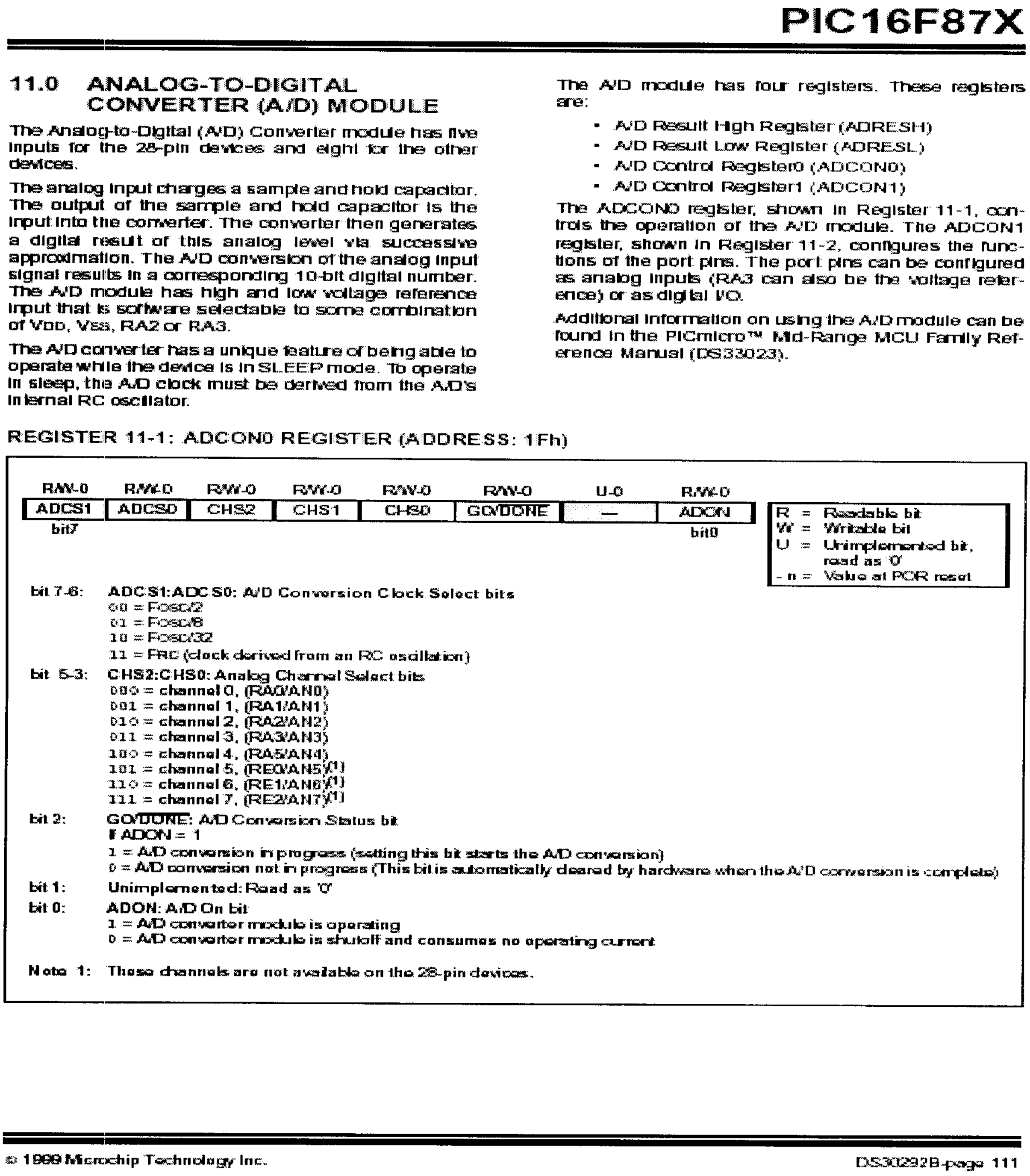
bit 7: WCOL: Write Collision Detect bit
<u>Master Mode:</u>
1 = A write to SSPBUF was attempted while the I^{2}C conditions were not valid
0 = No collision
<u>Slave Mode:</u>
1 = SSPBUF register is written while still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6: SSPOV: Receive Overflow Indicator bit
<u>In SPI mode</u>
1 = A new byte is received while SSPBUF holds previous data. Data in SSPSR is lost on overflow. . In slave mode the user must read the SSPBUF, even if only transmitting data, to avoid overflows. In master mode the overflow bit is not set since each operation is initiated by writing to the SSPBUF register. (Must be cleared in software).
0 = No overflow
<u>In I^{2}C mode</u>
1 = A byte is received while the SSPBUF is holding the previous byte. SSPOV is a "don't care" in transmit mode. (Must be cleared in software).
0 = No overflow

bit 5: SSPEN: Synchronous Serial Port Enable bit
<u>In SPI mode</u>, when enabled, these pins must be properly configured as input or output.
1 = Enables serial port and configures SCK, SDO, SDI, and $\overline{SS}$ as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins
<u>In I^{2}C mode</u>, when enabled, these pins must be properly configured as input or output
1 = Enables the serial port and configures the SDA and SCL pins as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins

bit 4: CKP: Clock Polarity Select bit
<u>In SPI mode</u>
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
<u>In I^{2}C slave mode</u>, SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch) (Used to ensure data setup time)
<u>In I^{2}C master mode</u>
Unused in this mode

bit 3-0: SSPM3:SSPM0: Synchronous Serial Port Mode Select bits
0000 = SPI master mode, clock = Fosc/4
0001 = SPI master mode, clock = Fosc/16
0010 = SPI master mode, clock = Fosc/64
0011 = SPI master mode, clock = TMR2 output/2
0100 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin
0110 = I^{2}C slave mode, 7-bit address
0111 = I^{2}C slave mode, 10-bit address
1000 = I^{2}C master mode, clock = Fosc / (4 * (SSPADD+1))
1011 = I^{2}C firmware controlled master mode (slave idle)
1110 = I^{2}C firmware controlled master mode, 7-bit address with start and stop bit interrupts enabled
1111 = I^{2}C firmware controlled master mode, 10-bit address with start and stop bit interrupts enabled.
1001, 1010, 1100, 1101 = reserved



11.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The Analog-to-Digital (A/D) Converter module has five inputs for the 28-pin devices and eight for the other devices.

The analog input charges a sample and hold capacitor. The output of the sample and hold capacitor is the input into the converter. The converter then generates a digital result of this analog level via successive approximation. The A/D conversion of the analog input signal results in a corresponding 10-bit digital number. The A/D module has high and low voltage reference input that is software selectable to some combination of VDD, VSS, RA2 or RA3.

The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode. To operate in sleep, the A/D clock must be derived from the A/D's internal RC oscillator.

The A/D module has four registers. These registers are:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register0 (ADCON0)
- A/D Control Register1 (ADCON1)

The ADCON0 register, shown in Register 11-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 11-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be the voltage reference) or as digital I/O.

Additional information on using the A/D module can be found in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

REGISTER 11-1: ADCON0 REGISTER (ADDRESS: 1Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	—	ADON
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **ADCS1:ADCS0: A/D Conversion Clock Select bits**
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = FRC (clock derived from an RC oscillation)

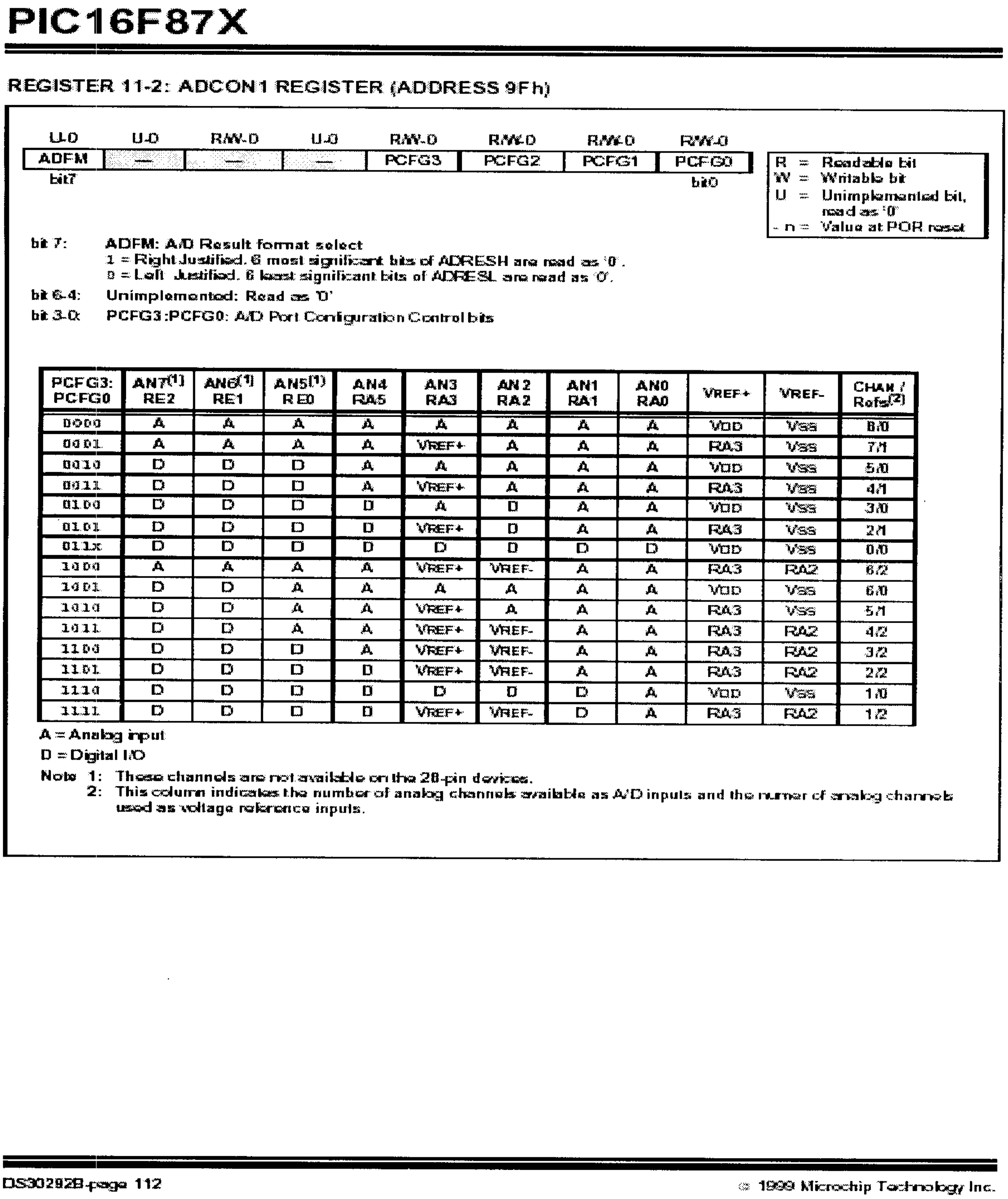
bit 5-3: **CHS2:CHS0: Analog Channel Select bits**
000 = channel 0, (RA0/AN0)
001 = channel 1, (RA1/AN1)
010 = channel 2, (RA2/AN2)
011 = channel 3, (RA3/AN3)
100 = channel 4, (RA5/AN4)
101 = channel 5, (RE0/AN5)(1)
110 = channel 6, (RE1/AN6)(1)
111 = channel 7, (RE2/AN7)(1)

bit 2: **GO/$\overline{\text{DONE}}$: A/D Conversion Status bit**
If ADON = 1
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete)

bit 1: **Unimplemented: Read as '0'**

bit 0: **ADON: A/D On bit**
1 = A/D converter module is operating
0 = A/D converter module is shutoff and consumes no operating current

Note 1: These channels are not available on the 28-pin devices.



PIC16F87X

REGISTER 11-2: ADCON1 REGISTER (ADDRESS 9Fh)

U-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **ADFM**: A/D Result format select
1 = Right Justified. 6 most significant bits of ADRESH are read as '0'.
0 = Left Justified. 6 least significant bits of ADRESL are read as '0'.

bit 6-4: **Unimplemented**: Read as '0'

bit 3-0: **PCFG3:PCFG0**: A/D Port Configuration Control bits

PCFG3: PCFG0	AN7[1] RE2	AN6[1] RE1	AN5[1] RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN / Refs[2]
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

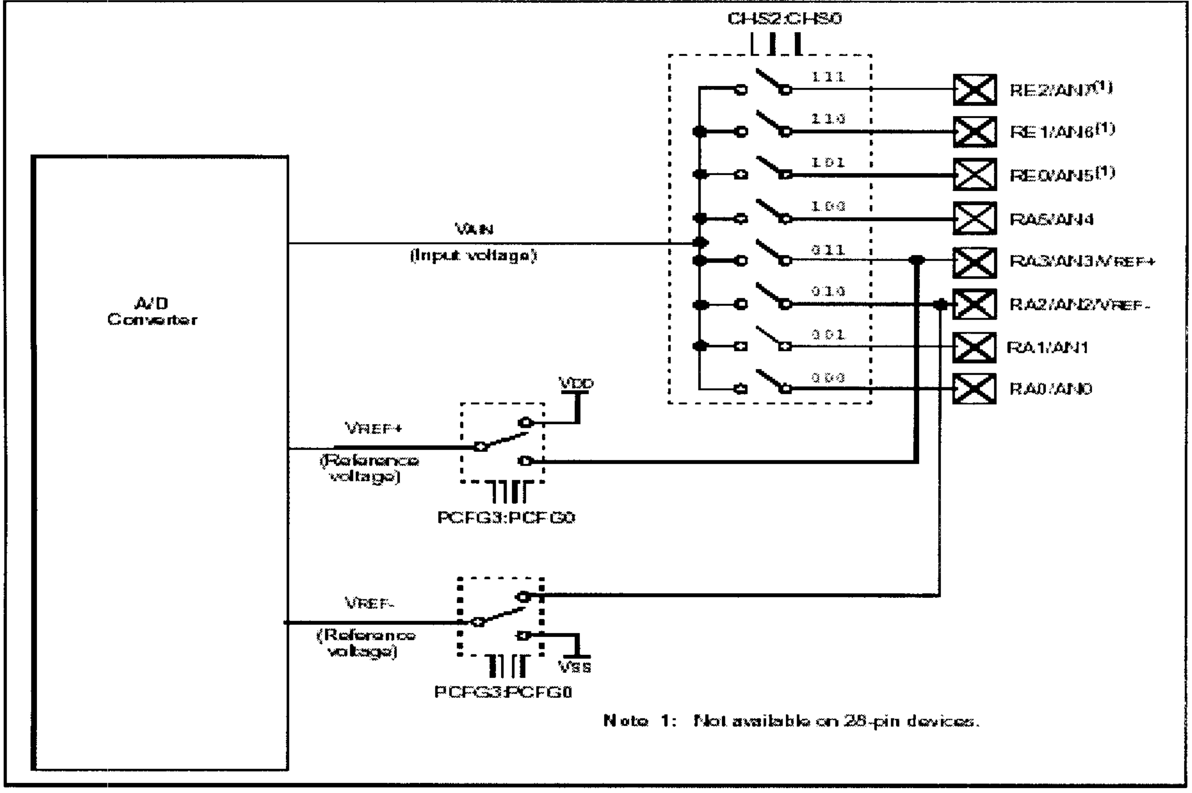
A = Analog input
D = Digital I/O

Note 1: These channels are not available on the 28-pin devices.
2: This column indicates the number of analog channels available as A/D inputs and the numer of analog channels used as voltage reference inputs.

DS30292B-page 112

© 1999 Microchip Technology Inc.

FIGURE 11-1: A/D BLOCK DIAGRAM



Note 1: Not available on 28-pin devices.

11.1 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 11-2. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), Figure 11-2. The maximum recommended impedance for analog sources is 10 kΩ. As the impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (changed), this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time, Equation 11-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

To calculate the minimum acquisition time, TACQ, see the PICmicro™ Mid-Range Reference Manual (DS33023).

12.10 Interrupts

The PIC16F87X family has up to 14 sources of interrupt. The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

> **Note:** Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set regardless of the status of the GIE bit. The GIE bit is cleared on reset.

The "return from interrupt" instruction, RETFIE, exits the interrupt routine, as well as sets the GIE bit, which re-enables interrupts.

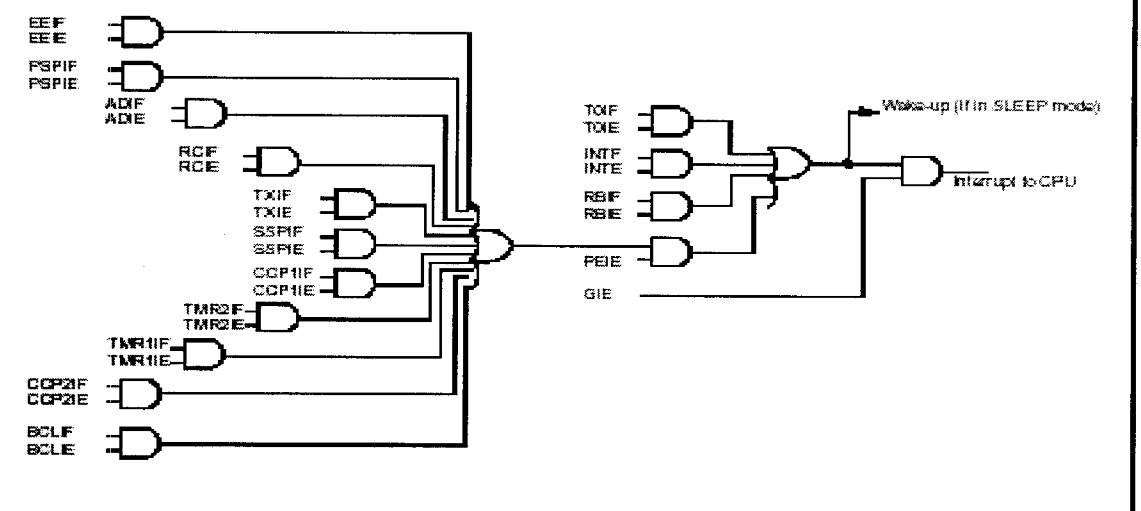
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the special function registers, PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers, PIE1 and PIE2, and the peripheral interrupt enable bit is contained in special function register INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit

FIGURE 12-9: INTERRUPT LOGIC



The following table shows which devices have which interrupts.

Device	T0IF	INTF	RBIF	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	EEIF	BCLIF	CCP2IF
PIC16F876/873	Yes	Yes	Yes	-	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PIC16F877/874	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

National Semiconductor

May 2000

LM78XX
Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expanded to make the LM78XX series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

Features

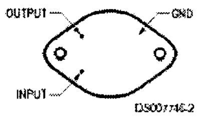
- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

LM7805C	5V
LM7812C	12V
LM7815C	15V

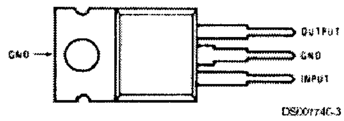
Connection Diagrams

Metal Can Package
TO-3 (K)
Aluminum



Bottom View
Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



Top View
Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings (Note 3)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Input Voltage	
(V_O = 5V, 12V and 15V)	35V
Internal Power Dissipation (Note 1)	Internally Limited
Operating Temperature Range (T_A)	0°C to +70°C
Maximum Junction Temperature	
(K Package)	150°C
(T Package)	150°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 sec.)	
TO-3 Package K	300°C
TO-220 Package T	230°C

Electrical Characteristics LM78XXC (Note 2)

0°C ≤ T_J ≤ 125°C unless otherwise noted.

Output Voltage				5V			12V			15V			Units
Input Voltage (unless otherwise noted)				10V			19V			23V			
Symbol	Parameter	Conditions		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
V_O	Output Voltage	Tj = 25°C, 5 mA ≤ I_O ≤ 1A		4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V
		P_D ≤ 15W, 5 mA ≤ I_O ≤ 1A		4.75		5.25	11.4		12.6	14.25		15.75	V
		$V_{MIN} \le V_{IN} \le V_{MAX}$		(7.5 ≤ V_{IN} ≤ 20)			(14.5 ≤ V_{IN} ≤ 27)			(17.5 ≤ V_{IN} ≤ 30)			V
ΔV_O	Line Regulation	I_O = 500 mA	Tj = 25°C		3	50		4	120		4	150	mV
			ΔV_{IN}	(7 ≤ V_{IN} ≤ 25)			14.5 ≤ V_{IN} ≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			50			120			150	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 20)			(15 ≤ V_{IN} ≤ 27)			(18.5 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 1A	Tj = 25°C			50			120			150	mV
			ΔV_{IN}	(7.5 ≤ V_{IN} ≤ 20)			(14.6 ≤ V_{IN} ≤ 27)			(17.7 ≤ V_{IN} ≤ 30)			V
			0°C ≤ Tj ≤ +125°C			25			60			75	mV
			ΔV_{IN}	(8 ≤ V_{IN} ≤ 12)			(16 ≤ V_{IN} ≤ 22)			(20 ≤ V_{IN} ≤ 26)			V
ΔV_O	Load Regulation	Tj = 25°C	5 mA ≤ I_O ≤ 1.5A		10	50		12	120		12	150	mV
			250 mA ≤ I_O ≤ 750 mA			25			60			75	mV
		5 mA ≤ I_O ≤ 1A, 0°C ≤ Tj ≤ +125°C				50			120			150	mV
I_Q	Quiescent Current	I_O ≤ 1A	Tj = 25°C			8			8			8	mA
			0°C ≤ Tj ≤ +125°C			8.5			8.5			8.5	mA
ΔI_Q	Quiescent Current Change	5 mA ≤ I_O ≤ 1A				0.5			0.5			0.5	mA
		Tj = 25°C, I_O ≤ 1A				1.0			1.0			1.0	mA
		$V_{MIN} \le V_{IN} \le V_{MAX}$		(7.5 ≤ V_{IN} ≤ 20)			(14.8 ≤ V_{IN}≤ 27)			(17.9 ≤ V_{IN} ≤ 30)			V
		I_O ≤ 500 mA, 0°C ≤ Tj ≤ +125°C				1.0			1.0			1.0	mA
		$V_{MIN} \le V_{IN} \le V_{MAX}$		(7 ≤ V_{IN} ≤ 25)			(14.5 ≤ V_{IN}≤ 30)			(17.5 ≤ V_{IN} ≤ 30)			V
V_N	Output Noise Voltage	T_A =25°C, 10 Hz ≤ f ≤ 100 kHz			40			75			90		µV
$\frac{\Delta V_{IN}}{\Delta V_{OUT}}$	Ripple Rejection	f = 120 Hz	I_O ≤ 1A, Tj = 25°C	62	80		55	72		54	70		dB
			or I_O ≤ 500 mA 0°C ≤ Tj ≤ +125°C	62			55			54			dB
		$V_{MIN} \le V_{IN} \le V_{MAX}$		(8 ≤ V_{IN} ≤ 18)			(15 ≤ V_{IN} ≤ 25)			(18.5 ≤ V_{IN} ≤ 28.5)			V
R_O	Dropout Voltage	Tj = 25°C, I_{OUT} = 1A			2.0			2.0			2.0		V
	Output Resistance	f = 1 kHz			8			18			19		mΩ

FAIRCHILD
SEMICONDUCTOR®

www.fairchildsemi.com

KA78XX/KA78XXA

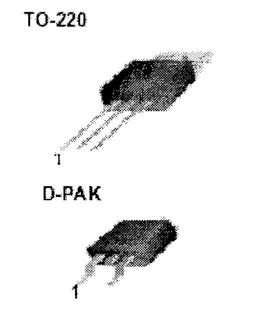
3-Terminal 1A Positive Voltage Regulator

Features

- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

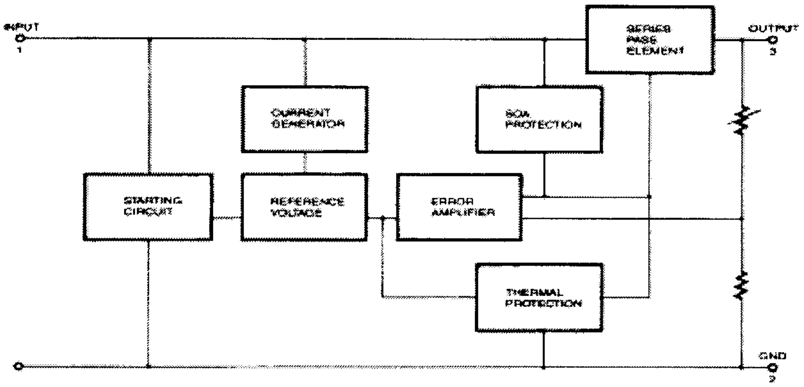
Description

The KA78XX/KA78XXA series of three-terminal positive regulator are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



1. Input 2. GND 3. Output

Internal Block Digram



Rev. 1.0.0

REFERENCES

BOOKS:

1. Electrical and Electronics Measurement and Instrumentation by A.K.Shawney-Dhanpat Rai and Sons
2. Linear Integrated Circuits by D. Roy Choudhury & Shail Jain
3. Power Electronics by Muhammad H. Rashid
4. Power Electronics by Bhimbra

WEB ADDRESSES:

www.microchip.com

www.enercon.com

www.icmasters.com