**ABSTRACT**

In the present day scenario, many times we see that the garbage bins or Dust bin are placed at public places in the cities are overflowing due to increase in the waste every day. It creates unhygienic conditions for people as well as ugliness to that place leaving foul smell. To avoid all such situations we are going to implement a project called IOT Based Smart waste management system.

In the proposed system, the garbage bins are assembled with sensor unit such as level sensors which is interfaced with PIC microcontroller. This sensor unit shows the current level of wastes, present in garbage bins. These values are uploaded in the server database using Bluetooth wireless communication. These values can be visualized by the corporation officials by viewing the particular webpage.

The main aim of this project is to reduce human resources and efforts along with the enhancement of a smart city vision. The “SMART WASTE MANAGEMENT APP” will be created, which manages the dynamic values of all garbage bins placed in the city. The people of particular location make use of this app, to check the level of garbage in their location. If the level indicates full, then they can switch to next garbage which is nearest to them. Hence this reduces the human effort.

As soon as the level of bin reaches the pre-specified setpoint, a message will be given to the waste collector of that location, using GSM module. Then he can check the status of waste collected using app, and the waste disposal takes place with less effort.

An LCD display will be provided in front of the bin, to indicate the status to the passerby. An LED will blink indicating that the level of bin crossed setpoint. This is mainly implemented for the purpose of understanding of lay-man. By adopting these measures, our city can become a “SMART CITY” within a short period of time. ***Keywords:*** *Zigbee wireless communication, PIC microcontroller, GSM module.*

**ii**

|  |  |  |
| --- | --- | --- |
|  | **TABLE OF CONTENTS** |  |
| **CHAPTER NO** | **TITLE** | **PAGE NO.** |
|  | **ACKNOWLEDGEMENT** | **i** |
|  | **ABSTRACT** | **ii** |
|  | **LIST OF TABLES** | **viii** |
|  | **LIST OF FIGURES** | **ix** |
|  | **LIST OF SYMBOLS** | **xi** |
|  | **LIST OF ABBREVIATIONS** | **xii** |
| **1** | **INTRODUCTION** | **01** |
|  | 1.1 WASTE MANAGEMENT | 01 |
|  | 1.2 AIM OF THE PROJECT | 02 |
|  | 1.3 OBJECTIVES OF THE PROJECT | 02 |
|  | 1.4 LITERATURE SURVEY | 02 |
|  | 1.5 ORGANISATION OF THE REPORT | 04 |
| **2** | **WASTE LEVEL DETECTION** | **06** |
|  | 2.1 DEVICES USED FOR WASTE LEVEL DETECTION | 06 |
|  | 2.1.1 Ultrasonic Sensor | 06 |
|  | 2.1.2 Infrared (IR) Sensor | 07 |
|  | 2.2 SPECIFICATIONS | 07 |
|  | 2.2.1 Materials | 07 |
|  | 2.2.2 Physical Product | 07 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 2.2.3 | Cable Colours | 07 |  |
|  | 2.2.4 | Environmental Range | 08 |  |
|  | 2.3 PROBLEM IDENTIFICATION | | 08 |  |
|  | 2.4 ISSUES OF THE EXISTING METHOD | | 08 |  |
| **3** | **PROPOSED METHOD** | | **09** |  |
|  | 3.1 PROPOSED METHADOLOGY | | 09 |  |
|  | 3.2 BLOCK DIAGRAM | | 09 |  |
|  | 3.3 BLOCK DIAGRAM EXPLANATION | | 10 |  |
|  | 3.3.1 Conditions | | 10 |  |
| **4** | **HARDWARE REQUIREMENTS** | | **12** |  |
|  | 4.1 POWER SUPPLY UNIT | | 12 |  |
|  | 4.1.1 | Working of Power Supply Unit | 12 |  |
|  | 4.1.2 | Characteristics of Power Supply Unit | 13 |  |
|  |  | 4.1.2.1 Load Regulation | 13 |  |
|  |  | 4.1.2.2 Minimum Load Resistance | 14 |  |
|  |  | 4.1.2.3 Source/Line Regulation | 14 |  |
|  |  | 4.1.2.4 Output Impedance | 14 |  |
|  |  | 4.1.2.5 Ripple Rejection | 14 |  |
|  | 4.2 PIC16F877A MICROCONTROLLER | | 15 |  |
|  | 4.2.1 | Features of PIC16F877A Microcontroller | 15 |  |
|  |  |  |
|  |  | 4.2.1.1 General Features | 15 |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | 4.2.1.2 Peripheral Features | 16 |  |
|  | 4.2.1.3 Special Features | 16 |  |
| 4.2.2 | Port Details | 17 |  |
| 4.2.3 Utilization of PIC16F877A Microcontroller | | 18 |  |
|  |  |
| 4.3 IR SENSORS | | 18 |  |
| 4.3.1 | IR Radiation Theory | 18 |  |
| 4.3.2 | Types of IR Sensors | 19 |  |
| 4.3.3 | Construction of IR Sensor | 20 |  |
| 4.3.4 | Working Principle | 20 |  |
| 4.4 GSM MODULE | | 21 |  |
| 4.4.1 | GSM Module Specifications | 22 |  |
| 4.4.2 | Firmware Support | 22 |  |
| 4.4.3 Working of GSM Module | | 23 |  |
| 4.5 LIQUID DRYSTAL DISPLAY | | 24 |  |
| 4.5.1 | Working of Liquid Crystal Display | 24 |  |
| 4.5.2 | Important Codes of LCD Display | 25 |  |
| 4.6 BUZZER | | 26 |  |
| 4.6.1 | Working Principle of Buzzer | 26 |  |
| 4.6.2 | Synchronization of Buzzer with PIC16F877A | 26 |  |
|  | Microcontroller |  |
|  |  |  |
| 4.7 RF MODULE | | 27 |  |
| 4.7.1 | Bluetooth Module | 27 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 4.7.2 | Features | 28 |  |
|  | 4.7.3 | Specifications | 28 |  |
|  | 4.7.4 | Working of Bluetooth Module | 28 |  |
| **5** | **SOFTWARE REQUIREMENTS** | | **30** |  |
|  | 5.1 INTRODUCTION TO LABVIEW | | 30 |  |
|  | 5.2 LABVIEW PROGRAM FOR WASTE MANAGEMENT | | 30 |  |
|  |  |  |
|  | 5.2.1 | VISA | 31 |  |
|  | 5.2.2 | Applications of VISA | 31 |  |
|  | 5.3 CREATION OF WASTE MONITORING | | 33 |  |
|  | APPLICATION SETUP IN LABVIEW | |  |
|  |  |  |
| **6** | **UTILIZATION OF WEB SERVER** | | **35** |  |
|  | 6.1 INTRODUCTION TO WEB SERVER USING CLOUD | | 35 |  |
|  |  |  |
|  | 6.1.1 | Benefits of Web Server | 35 |  |
|  | 6.2 CREATION OF WEB SERVER USING WEB | | 35 |  |
|  | PUBLISHING TOOL | |  |
|  |  |  |
| **7** | **DATA DASHBOARD MOBILE APPLICATION** | | **37** |  |
|  | 7.1 INTRODUCTION TO DATA DASHBOARD | | 37 |  |
|  | 7.2 SMART WASTE MANAGEMENT APP | | 37 |  |
| **8** | **FLOWCHART FOR WASTE MANAGEMENT** | | **39** |  |
|  | 8.1 FLOWCHART SETUP FOR SMART WASTE | | 39 |  |
|  | MANAGEMENT SYSTEM | |  |
|  |  |  |
| **9** | **HARDWARE SETUP** | | **41** |  |

|  |  |  |
| --- | --- | --- |
| **10** | **CONCLUSION AND OUTCOMES** | **43** |
|  | 10.1 CONCLUSION AND FUTURE SCOPE | 43 |
|  | 10.2 EXPECTED DELIVERABLES | 43 |
|  | 10.3 EXPECTED BENEFITS | 43 |
|  | 10.4 NOVELITY OF THE PROJECT | 44 |

**REFERENCES**

**APPENDICES**

**APPENDIX - I OVERALL CIRCUIT DIAGRAM**

**APPENDIX – II PIC16F877A MICROCONTROLLER**

**PROGRAM**

**APPENDIX - III PIC16F877A MICROCONTROLLER**

**DATASHEET**

|  |  |  |
| --- | --- | --- |
|  | **LIST OF TABLES** |  |
| **TABLE NO.** | **TITLE** | **PAGE NO.** |
| 1 | Port details of PIC16F877A microcontroller | 18 |
| 2 | Important codes of liquid crystal display | 25 |

**viii**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **LIST OF FIGURES** |  |  |
| **FIGURE NO.** | **TITLE** | **PAGE NO.** |  |
| 1 | Working of ultrasonic sensor | 6 |  |
| 2 | IR sensor | 7 |  |
| 3 | IOT based smart waste management system |  |  |
| using labVIEW | 9 |  |
|  |  |
| 4 | Block diagram of power supply unit | 13 |  |
| 5 | Circuit diagram for power supply unit | 15 |  |
| 6 | Pin details of PIC16F877A microcontroller | 17 |  |
| 7 | Pin diagram of IR sensor | 19 |  |
| 8 | Circuit diagram of IR sensor | 20 |  |
| 9 | GSM module | 22 |  |
| 10 | Interfacing PIC16F877A microcontroller with |  |  |
| GSM module | 23 |  |
|  |  |
| 11 | Interfacing PIC16F877A microcontroller with |  |  |
| Liquid crystal display | 24 |  |
|  |  |
| 12 | Pin diagram of buzzer | 26 |  |
| 13 | Pin diagram of buzzer | 27 |  |
| 14 | Interfacing of HC-05 Bluetooth module with |  |  |
| PIC16F877A microcontroller | 29 |  |
|  |  |
| 15 | LabVIEW software | 30 |  |
| 16 | Various functions of VISA | 31 |  |

**ix**

17

18

19

20

21

22

23

24

|  |  |
| --- | --- |
| Front panel indication for smart waste |  |
| management using labVIEW | 32 |
| Upper portion of web page for smart waste |  |
| management system | 36 |
| Lower portion of web page for smart waste |  |
| management system | 36 |
| Creation of Smart waste management app by |  |
| utilizing data dashboard mobile application | 38 |
| Flowchart for waste management system | 39 |
| Hardware setup of Smart waste management |  |
| system | 41 |
| LCD indication of waste level in dustbins | 42 |
| Overall circuit diagram of IOT based smart |  |
| waste management system | 46 |

**x**

**LIST OF SYMBOLS**

**SYMBOLS** **EXPANSION**

**μF** Micro Farad.

**A** Ampere.

**MHz** Mega Hertz.

**V** Volts.

**xi**

**LIST OF ABBREVIATIONS**

**ACRONYMS** **ABBREVIATIONS**

**GSM** Global System for Mobile communications.

**AT** Attention commands.

**RF** Radio Frequency.

**IR sensor** Infrared sensor.

**App** Application.

**IOT** Internet Of Things.

**MSWM** Municipal Solid Waste Management.

**LCD** Liquid Crystal Display.

**LED** Light Emitting Diode.

**RISC** Reduced Instruction Set Computer.

**IDE** Integrated Development Environment.

**NI** National Instruments.

Laboratory Virtual Instrumentation Engineering

**LabVIEW** Workbench

**VISA** Virtual Instrument Software Architecture.

**HTTP** Hyper Text Transfer Protocol.

**xii**

**CHAPTER 1**

**INTRODUCTION**

**CHAPTER 1**

**INTRODUCTION**

**1.1 WASTE MANAGEMENT**

**Waste management** or **Waste disposal** is all the activities and actionsrequired to manage waste from its inception to its final disposal. This includes amongst other things, collection, transport, treatment and disposal of waste together with monitoring and regulation. It also encompasses the legal and regulatory framework that relates to waste management encompassing guidance on recycling etc. The term normally relates to all kinds of waste, whether generated during the extraction of raw materials, the processing of raw materials into intermediate and final products, the consumption of final products, or other human activities, including municipal (residential, institutional, commercial), agricultural, and social (health care, household hazardous waste, sewage sludge). Waste management is intended to reduce adverse effects of waste on health, the environment or aesthetics. Waste management practices are not uniform among countries (developed and developing nations), regions (urban and rural area), and sectors (residential and industrial).

Due to the increase in population, waste management poses a great threat to all the countries in the world. At present, the amount of solid waste produced in urban India is 68.8 million tons per year. This amount is expected to be double by 2025. The complexity in managing the wastes and disposing them requires lot of efforts and it is also time consuming. This problem occurs in both rural and urban areas, but the rural people are worst affected, as in urban areas the problem can be rectified to an extent because of technological advancements. This is impossible in rural areas. As more number of wastes getting accumulated, these wastes starts to spill out of garbage bin and possesses great threat to the locality, especially to the children in that locality. To avoid this situation, an efficient waste management system is required. There are many technologies are used for waste collection as

1

well as for well managed recycling. In the proposed model, the waste levels are being sensed and it is transmitted to the waste collector and to the corporation office using RF module. The process of waste monitoring takes place here. The local residents know about the waste details, the bins nearest to their locality and the bin number using an application. Hence efficient waste management is made possible.

**1.2 AIM OF THE PROJECT**

To monitor the level of garbage prevailing in the city and to transmit the information to the garbage collection station and garbage collector and to reduce the dumping and overflowing of garbage.

**1.3 OBJECTIVES OF THE PROJECT**

∑ To detect the level of garbage, present in the city.

∑ Update the status regarding the garbage level detected, to the garbage collection unit.

∑ To eliminate the harmful effects caused by mosquitoes, insects etc.

∑ To reduce the overflowing of garbage wastes in the city by means of an application titled as **“SMART WASTE MANAGEMENT APP”.**

**1.4 LITERATURE SURVEY**

1. **Kanchan Mahajan, “Waste Bin Monitoring System Using Integrated Technologies”, International Journal of Innovative Research in Science, Engineering and Technology, July 2014, Vol.3, No.7.**

In this project, the ZigBee, GSM (Global System for Mobile Communication) and ARM7 is used to form the Integrated system to monitor the waste bins remotely. The sensors are placed in the common garbage bins placed at the public places. When the garbage reaches the level of the sensor, then that indication will be given to ARM 7 Controller. The controller will give indication to the driver of garbage

2

collection truck as to which garbage bin is completely filled and needs urgent attention. ARM 7 will give indication by sending SMS using GSM technology.

1. **M. Al-Maaded, N.K. Madi, A. Ramazan Kahraman, A. Hodzic, N.G. Ozerkan, “An Overview of Solid Waste Management and Plastic Recycling in Qatar”, Springer Journal of Polymers and the Environment, March 2012, Vol.20, No.1, pp.186-194.**

It is important to understand the societal concerns over the increased rate of resource consumption and waste production and therefore the policy makers have encouraged recycling and reuse strategies to reduce the demand for raw materials and to decrease the quantity of waste going to landfill.

1. **Islam M.S, Arebey M, Hannan M A, Basri H, “Overview for solid waste bin monitoring and collection system”, International Conference of Innovation Management and Technology Research , 2012, pp.258 – 262.**

This paper shows the introduction of a system combined with an integrated combination of Radio Frequency Identification, Global Position System, General Packet Radio Service, Geographic Information System and web camera that will solve the problem of solid waste They also analyzed the actual performance of the system.

1. **Narayan Sharma, “Smart Bin Implemented for Smart City”, International Journal of Scientific & Engineering Research, September 2015, Vol.6, No.9.**

This paper describes the application of a model known as “Smart Bin” in managing the waste collection system of an entire city. The network of sensors enabled smart bins connected through the cellular network generates a large amount of data, which is further analyzed and visualized at real time to gain insights about the status of waste around the city. This paper also aims at encouraging further research in the topic of waste management.

3

1. **Raghumani Singh C, Dey M, “Solid waste management of Thoubal Municipality, Manipur-a case study”, International Conference of Green Technology and Environmental Conservation, 2011, pp. 21 – 24.**

The objective of the study was to determine the characterization of the waste and the current system of management activities. The paper highlights an overview of the current municipal solid waste management (MSWM) system of Thoubal Municipality and it concludes with a few suggestions, which may be beneficial to the authorities to work towards further improvement of the current management systems.

**1.5 ORGANIZATION OF THE REPORT**

The dissertation contains seven chapters in total and is organized as under:

* **CHAPTER 1**

This chapter includes the introductory part of the report depicting the basic concepts, need and objective of Smart waste management system and narrates the detailed literature review.

* **CHAPTER 2**

This chapter describes the Importance and features of waste level detection and gives the problem faced by the existing methods used for waste management.

* **CHAPTER 3**

This chapter deals with the proposed methodology that is used in waste management along with the working of its block diagram.

* **CHAPTER 4**

This chapter deals with the complete hardware requirements used for smart waste management system.

* **CHAPTER 5**

This chapter deals with the software requirements in this project and describes the creation of LabVIEW program and waste monitoring application in LabVIEW.

4

* **CHAPTER 6**

This chapter deals with the introduction of cloud concepts and facilitates the utilization of web server using web publishing tool.

* **CHAPTER 7**

This chapter deals with the features and specifications of data dashboard mobile application and its utilization for creating **“SMART WASTE**

**MANAGEMENT APP”**

* **CHAPTER 8**

This chapter describes with the working of flowchart for waste management.

* **CHAPTER 9**

This chapter contains the output photographs of this project.

* **CHAPTER 10**

This chapter includes the conclusion, outcomes, novel and future works of the project.

* **CHAPTER 11**

This chapter contains the references for this project.

5

**CHAPTER 2**

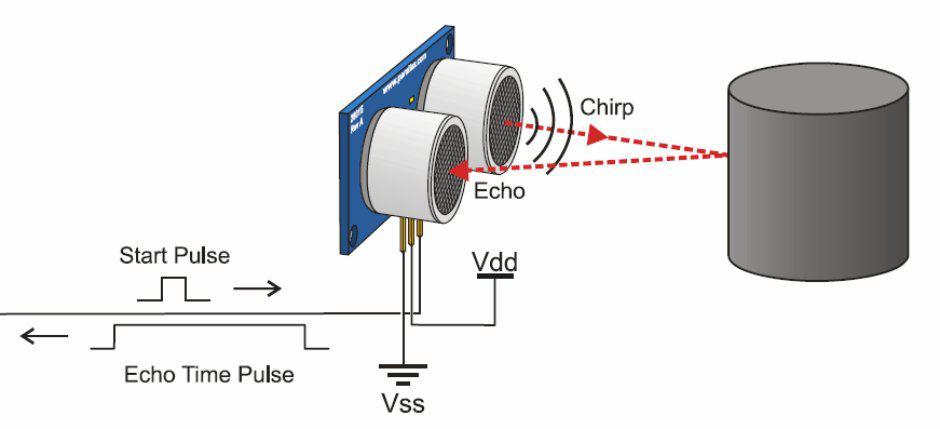
**WASTE LEVEL DETECTION**

**CHAPTER 2**

**WASTE LEVEL DETECTION**

**2.1 DEVICES USED FOR WASTE LEVEL DETECTION** The commonly used devices for waste level detection are, **2.1.1 Ultrasonic Sensor**

An Ultrasonic sensor is a device that can measure the distance to an object by using sound waves. It measures distance by sending out a sound wave at a specific frequency and listening for that sound wave to bounce back. By recording the elapsed time between the sound wave being generated and the sound wave bouncing back, it is possible to calculate the distance between the sonar sensor and the object. The working of ultrasonic sensor is shown in Fig.1.



**Fig.1. Working of ultrasonic sensor**

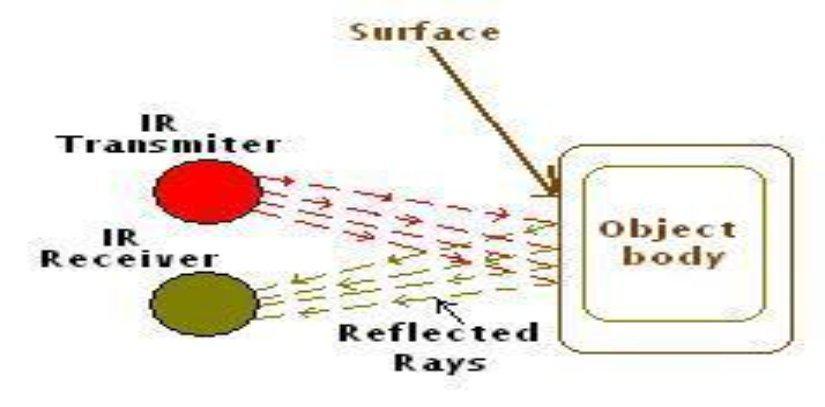
**Disadvantages of Ultrasonic Sensor**

In a single ultrasonic sensor, the distance of wastes only in particular portion of bin can only be monitored. So to monitor the remaining portions, more number of ultrasonic sensors needs to be placed. Hence the efficiency of waste level detection is reduced.

6

**2.1.2 Infrared (IR) Sensor**

An infrared sensor is an electronic device that emits and/or detects infrared radiation in order to sense some aspect of its surroundings. Infrared sensors can measure the heat of an object, as well as detect motion. Many of these types of sensors only measure infrared radiation, rather than emitting it, and thus are known as passive infrared (PIR) sensors. The working of IR sensor is shown in Fig.2.



**Fig.2. IR sensor**

**2.2 SPECIFICATIONS**

**2.2.1 Materials**

Probe Casing - PVC Premium Quality Extruded Tube.

Cable - PVC/PVC Multicore, Purpose Manufactured.

**2.2.2 Physical Product**

Dimensions mm - 32 mm Diameter x Specified Length.

Mounting - Via the Supplied Suspension Cleaning Bracket.

**2.2.3 Cable Colours**

Cores - 2, Connected to the Same Sensor.

Over sheath Diameter - 7.4 mm +/- 0.3 mm.

Bend Radius - Min Bend Radius 45 mm.

Conductor - 0.75 mm2, Multistrand, Tinned, Annealed Copper.

7

**2.2.4 Environmental Range**

Fahrenheit – 32 degree F to 149 degree F.

Centigrade – 0 degree C to 65degree C.

**2.3 PROBLEM IDENTIFICATION**

∑ A big challenge in urban and rural areas is solid waste management.

∑ Most often in the city, the dustbins are getting over flown and concern person don’t get the information within time, this leads to unsanitary condition in the surroundings.

∑ This leads to spreading of deadly diseases like malaria, typhoid etc. ∑ It may cause illness to human beings.

∑ Hence this leads to the bad look of the city which paves the way for air pollution.

∑ More traffic and Noise, due to vehicle transportation.

∑ Breathing problem exists due to the spreading of harmful gases like methane and carbon di oxide.

**2.4 ISSUES OF THE EXISTING METHOD**

∑ Time consuming. ∑ Less effective.

∑ Transportation of wastes in trucks becomes less effective, because of the lack of information.

∑ High costs.

∑ Unhygienic Environment and look of the city. ∑ Bad smell spreads.

∑ It may cause illness to human beings.

∑ More traffic and Noise, due to vehicle transportation.

8

**CHAPTER 3**

**PROPOSED METHOD**

**CHAPTER 3**

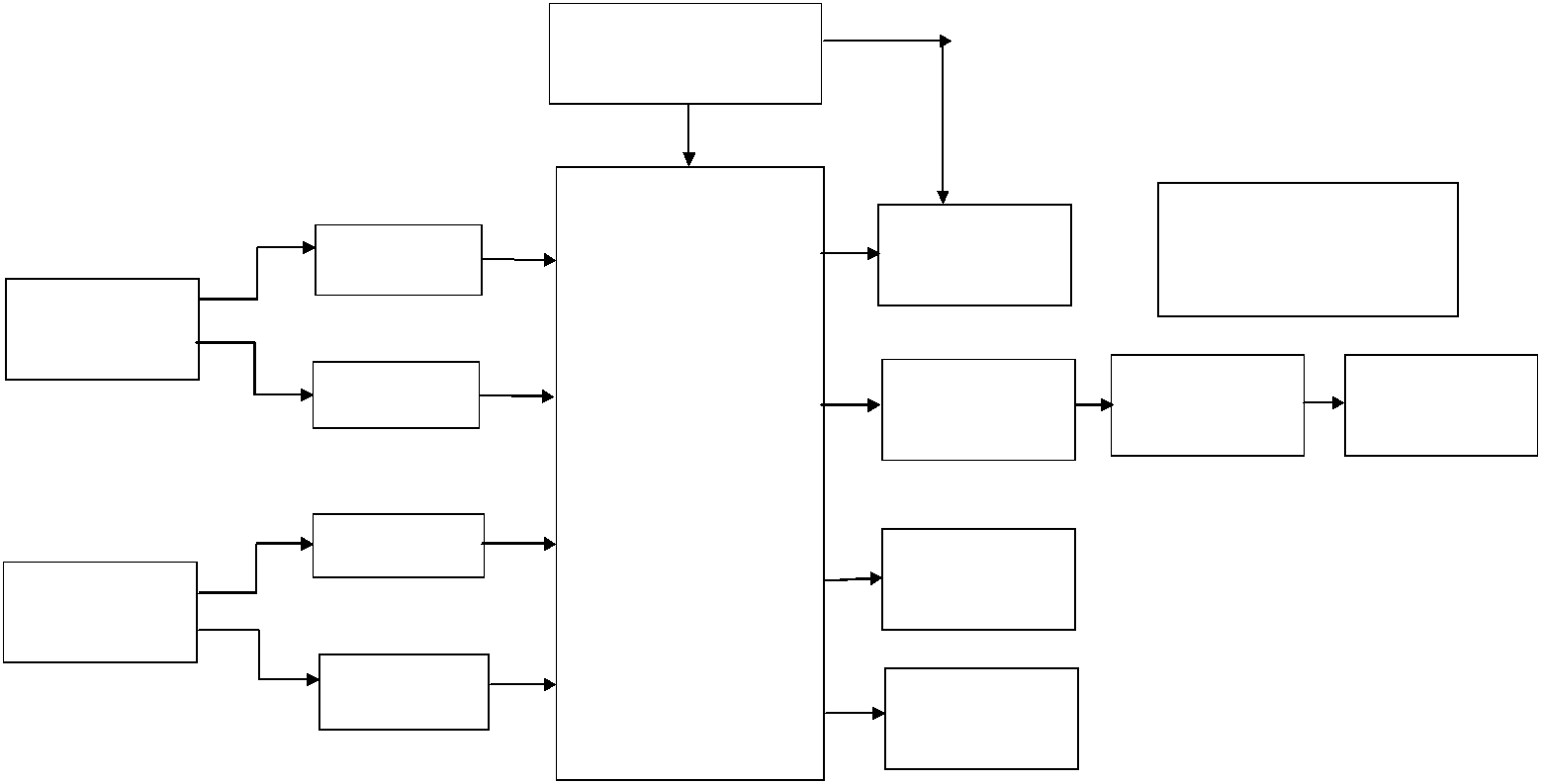
**PROPOSED METHOD**

**3.1 PROPOSED METHADOLOGY**

This chapter deals about various components utilized in this project. The working, operational features and specifications of the components used in this project are also explained here briefly.

**3.2 BLOCK DIAGRAM**

The block diagram of Smart waste management system is shown in Fig.3.



POWER SUPPLY UNIT

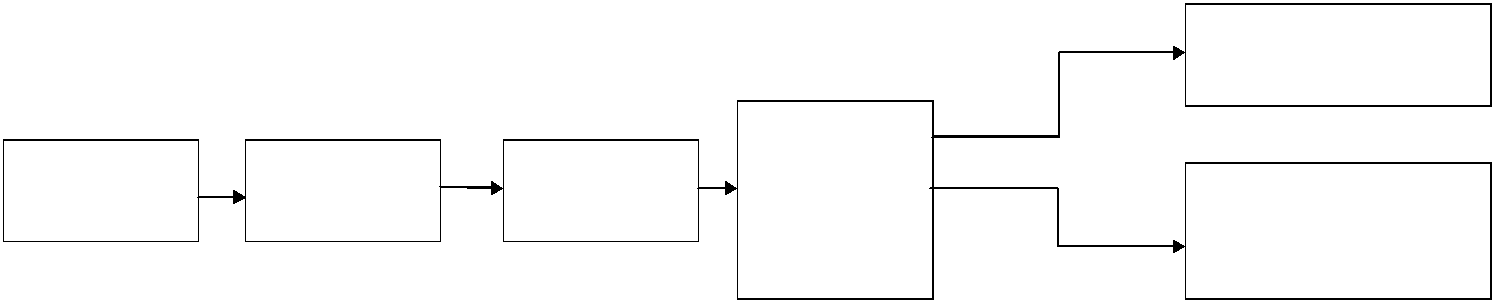
MESSAGE AND CALL

IR SENSOR 1 GSM MODULE  INTIMATION TO WASTE COLLECTOR

BIN NO 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IR SENSOR 2 | LEVEL | RF | WIRELESS |  |
|  | CONVERTER | TRANSMITTER | MEDIUM (AIR) |  |
|  | PIC16F877A |  |  |  |
|  | MICROCONTROLLER |  |  |  |
| IR SENSOR 3 | LCD DISPLAY |  |  |  |
|  |  |  |  |
| BIN NO 2 |  |  |  |  |
| IR SENSOR 4 | BUZZER |  |  |  |
|  |  |  |  |

**TRANSMITTER UNIT**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | WEB SERVER (FOR NON |  |
|  |  |  |  | ANDROID USERS) |  |
|  |  |  | PC LABVIEW |  |  |
| WIRELESS | RF RECEIVER | LEVEL | (IOT) (FOR | SMART WASTE |  |
| MEDIUM (AIR) |  | CONVERTER | CITY |  |
|  |  |  | CORPORATION | MANAGEMENT APP (FOR |  |
|  |  |  | OFFICE) | ANDROID USERS) |  |

**RECEIVER UNIT**

**Fig.3. IOT based smart waste management system using LabVIEW**

9

**3.3 BLOCK DIAGRAM EXPLANATION**

The proposed method consists of a power supply unit, bin sensors, PIC16F877A microcontroller, LCD display, Buzzer, GSM module, RF module. The microcontroller is powered by 5V DC supply. The project model consists of two bin setups. Each dustbin setup is equipped with two IR sensors. Totally four IR sensors are utilized in our project.

IR sensor1 and IR sensor2 is installed in bin setup1. IR sensor3 and IR sensor4 is installed in bin setup2. These IR sensors detect the waste level in the respective bins.

**3.3.1 Conditions**

∑ IR sensor1 detects 50% waste level in bin1 and IR sensor2 detects 100% waste level in bin1.

∑ IR sensor3 detects 50% waste level in bin2 and IR sensor2 detects 100%

waste level in bin2.

These digital signals are given as input to the PIC16F877a microcontroller. The program for waste management is written using MPLAB IDE software. This program is embedded into the microcontroller using PCKIT3 hardware setup. Based on the signals obtained from the respective IR sensors, the PIC16F877A microcontroller triggers signal to the LCD display. It displays the waste level for the respective bins.

The buzzer is powered by 5V supply. It produces a beep sound when both bins are indicated with full conditions. Simultaneously the waste level values are sent to the mobile numbers programmed in microcontroller using GSM module. The GSM module works with Attention (AT) commands. A message indication is sent to the mobile number of waste collector to collect the wastes from the bins immediately. This avoids the overflowing of wastes in garbage.

The same information is transmitted in wireless medium via Bluetooth transmitter. At the corporation office, the Bluetooth receiver is interfaced with LabVIEW software using Ethernet cable. A program is executed in LabVIEW

10

to monitor the waste level. This will be monitored and maintained by the corporation officials.

These values will be uploaded into the web server by means of cloud concepts. This is mainly done using web publishing tool. This is mainly utilized for the non android users. They can know about the bin number, waste level in bins and locality of bins from their homes.

An app known as “SMART WASTE MANAGEMENT APP” will be utilized using data dashboard mobile application. This is mainly utilized for the android users. They can know about the waste details from their homes and dispose the litters accordingly. Hence efficient smart waste management is possible through these methods.

11

**CHAPTER 4**

**HARWARE REQUIREMENTS**

**CHAPTER 4**

**HARDWARE REQUIREMENTS**

**4.1 POWER SUPPLY UNIT**

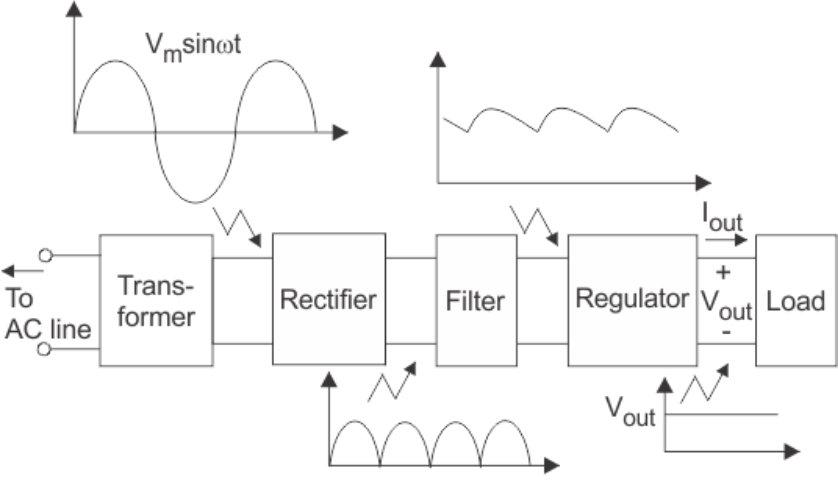
Power supply is an electronic circuit that is designed to provide a constant dc voltage of predetermined value across load terminals irrespective of ac mains fluctuations or load variations. The two main parts of a regulated power supply are a simple power supply and a voltage regulating device. The power supply output is given as input to the voltage regulating device that provides the final output. The voltage output of the power supply remains constant irrespective of large variations in the input AC voltage or output load current.

**4.1.1 Working of Power Supply Unit**

The input AC voltage (230 Volts, 5A), is supplied to a step down transformer. The output will be a stepped down 12V AC supply and 1Ampere. This AC voltage is then given to a bridge rectifier to produce a full-wave rectified 12V DC output. The filter capacitor output may have some voltage variations and ripples. Then it passes through the IC 7805 voltage regulator that produces a constant 5V DC output. This regulated DC voltage is then given to a voltage divider, which supplies the different DC voltages that may be needed for different electronic circuits. The Schematic block diagram of power supply unit is shown in Fig.4.

The potential divider is a single tapped resistor connected across the output terminals of the supply. The tapped resistor may consist of two or three resistors connected in series across the supply. A bleeder resistor may also be employed as a potential divider.

12



**Fig.4. Block diagram of power supply unit**

**4.1.2 Characteristics of Power Supply Unit**

The quality of the power supply is determined by various characteristics like load voltage, load current, voltage regulation, source regulation, output impedance, ripple rejection, and so on. Some of the characteristics are briefly explained below:

**4.1.2.1 Load Regulation**

The load regulation or load effect is the change in regulated output voltage when the load current changes from minimum to maximum value.

**Load regulation = Vno-load – Vfull-load.**

where,

Vno-load – Load Voltage at no load

Vfull-load – Load voltage at full load.

From the above equation we can understand that when Vno-load occurs the load resistance is infinite, that is, the out terminals are open circuited. Vfull-load

13

occurs when the load resistance is of the minimum value where voltage regulation is lost.

**% Load Regulation = [(Vno-load – Vfull-load)/Vfull-load] \* 100.**

**4.1.2.2 Minimum Load Resistance**

The load resistance at which a power supply delivers its full-load rated current at rated voltage is referred to as minimum load resistance.

**Minimum Load Resistance = Vfull-load/Ifull-load.**

The value of Ifull-load, full load current should never increase than that mentioned in the data sheet of the power supply.

**4.1.2.3 Source/Line Regulation**

In the block diagram, the input line voltage has a nominal value of 230 Volts but in practice, there are considerable variations in ac supply mains voltage. Since this ac supply mains voltage is the input to the ordinary power supply, the filtered output of the bridge rectifier is almost directly proportional to the ac mains voltage. The source regulation is defined as the change in regulated output voltage for a specified rage of line voltage.

**4.1.2.4 Output Impedance**

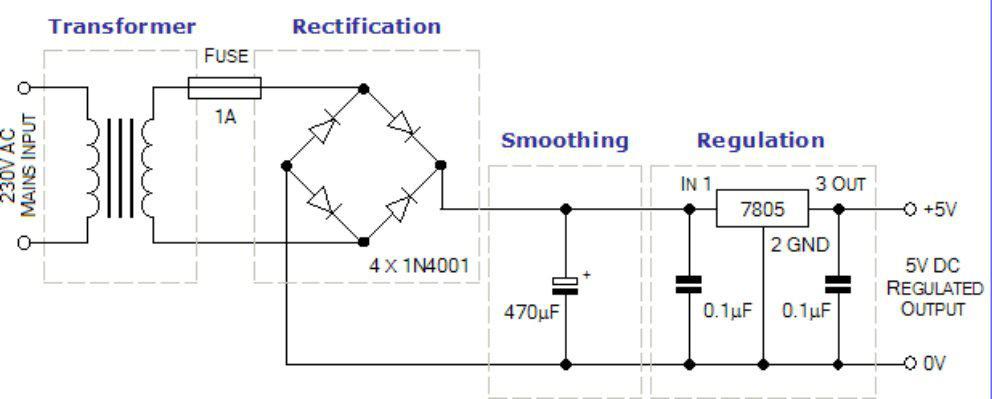
A regulated power supply is a very stiff dc voltage source. This means that the output resistance is very small. Even though the external load resistance is varied, almost no change is seen in the load voltage. An ideal voltage source has an output impedance of zero.

**4.1.2.5 Ripple Rejection**

Voltage regulators stabilize the output voltage against variations in input voltage. Ripple is equivalent to a periodic variation in the input voltage. Thus, a

14

voltage regulator attenuates the ripple that comes in with the unregulated input voltage. Since a voltage regulator uses negative feedback, the distortion is reduced by the same factor as the gain. The circuit diagram for power supply unit is shown in Fig.5.



**Fig.5. Circuit diagram for power supply unit**

**4.2 PIC16F877A MICROCONTROLLER**

PIC 16F877 is one of the most advanced microcontroller from Microchip. This controller is widely used for experimental and modern applications because of its low price, wide range of applications, high quality, and ease of availability. It is ideal for applications such as machine control applications, measurement devices, study purpose, and so on. The PIC 16F877 features all the components which modern microcontrollers normally have. The pin details of PIC16F877A microcontroller is shown in Fig.6.

**4.2.1 Features of PIC16F877A Microcontroller** The important features of PIC16F877A is given below, **4.2.1.1 General Features**

∑ High performance RISC CPU.

∑ Only 35 simple word instructions.

15

∑ All single cycle instructions except for program branches which are two cycles.

∑ Operating speed: clock input (200MHz), instruction cycle (200nS).

∑ Up to 368×8bit of RAM (data memory), 256×8 of EEPROM (data memory), 8k×14 of flash memory.

∑ Pin out compatible to PIC 16C74B, PIC 16C76, PIC 16C77. ∑ Eight level deep hardware stack.

∑ Interrupt capability (up to 14 sources).

**4.2.1.2 Peripheral Features**

∑ Timer 0: 8 bit timer/counter with pre-scalar. ∑ Timer 1:16 bit timer/counter with pre-scalar.

∑ Timer 2: 8 bit timer/counter with 8 bit period registers with pre-scalar and post-scalar.

∑ Two Capture (16bit/12.5nS), Compare (16 bit/200nS), Pulse Width Modules (10bit).

∑ 10bit multi-channel A/D converter

∑ Synchronous Serial Port (SSP) with SPI (master code) and I2C (master/slave).

∑ Universal Synchronous Asynchronous Receiver Transmitter (USART) with 9 bit addresses detection.

∑ Parallel Slave Port (PSP) 8 bit wide with external RD, WR and CS controls (40/46pin).

∑ Brown Out circuitry for Brown-Out Reset (BOR).

**4.2.1.3 Special Features**

∑ 100000 times erase/write cycle enhanced memory.

∑ 1000000 times erase/write cycle data EEPROM memory. ∑ Self programmable under software control.

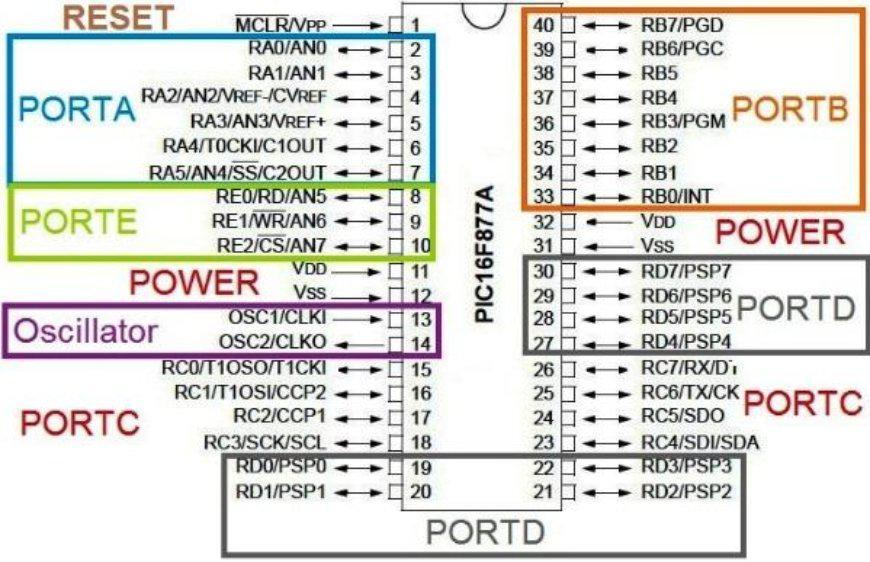
∑ In-circuit serial programming and in-circuit debugging capability.

16

∑ Single 5V,DC supply for circuit serial programming

∑ WDT with its own RC oscillator for reliable operation. ∑ Programmable code protection.

∑ Power saving sleep modes. ∑ Selectable oscillator options.



**Fig.6. Pin details of PIC16F877A microcontroller**

**4.2.2 Port Details**

PIC16F877 has 5 basic input/output ports. They are usually denoted by PORT A (R A), PORT B (RB), PORT C (RC), PORT D (RD), and PORT E (RE). These ports are used for input/ output interfacing. In this controller, “PORT A” is only 6 bits wide (RA-0 to RA-7), ”PORT B” , “PORT C”,”PORT D” are only 8 bits wide (RB-0 to RB-7,RC-0 to RC-7,RD-0 to RD-7), ”PORTE” has only 3 bit wide (RE-0 to RE-7). The port details of PIC16F877A microcontroller is shown in Table.1.

17

|  |  |  |  |
| --- | --- | --- | --- |
|  | **PORT-A** | RA-0 to RA-5 | 6 bit wide |
|  |  |  |  |
|  | **PORT-B** | RB-0 to RB-7 | 8 bit wide |
|  |  |  |  |
|  | **PORT-C** | RC-0 to RC-7 | 8 bit wide |
|  |  |  |  |
|  | **PORT-D** | RD-0 to RD-7 | 8 bit wide |
|  |  |  |  |
|  | **PORT-E** | RE-0 to RE-2 | 3 bit wide |
|  |  |  |  |

**Table.1. Port details of PIC16F877A microcontroller**

**4.2.3 Utilization of PIC16F877A Microcontroller**

The microcontroller controls the overall activities of the setup. The signal from the IR sensor1, IR sensor2 of bin1 and IR sensor3, IR sensor4 of bin2 is given as input to the microcontroller. The program for waste management is written using MPLAB IDE software. This program is embedded in microcontroller using PICKIT3 hardware. Based on the signals obtained from sensors, the microcontroller provides the output signals. The output from the microcontroller indicates the prespecified waste level status. This output will be given to the LCD display, Buzzer and Zigbee module for wireless communication.

**4.3 IR SENSORS**

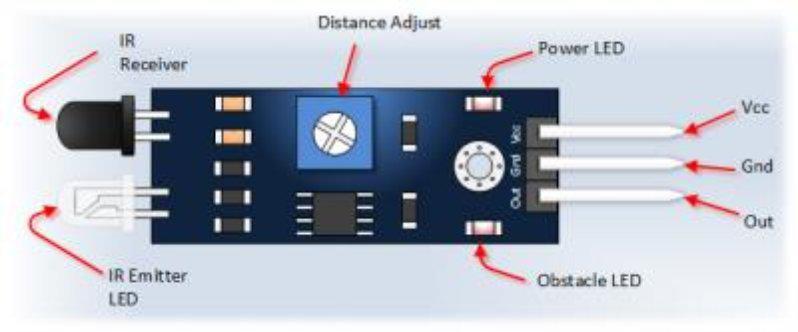
The disadvantage of ultrasonic sensor is overcome by IR sensors. The IR sensors are used to detect the waste levels in dustbins. It uses the Infrared (IR) ray to detect the waste obstacle in front of it.

**4.3.1 IR Radiation Theory**

Infrared waves are not visible to the human eye. In the electromagnetic spectrum, infrared radiation can be found between the visible and microwave regions. The infrared waves typically have wavelengths between 0.75 and 1000µm. The wavelength region which ranges from 0.75 to 3µm is known as

18

the near infrared regions. The region between 3 and 6µm is known as the mid-infrared and infrared radiation which has a wavelength greater higher than 6µm is known as far infrared. The pin diagram of IR sensor is shown in Fig.7. Infrared technology finds applications in many everyday products. Televisions use an infrared detector to interpret the signals sent from a remote control.



**Fig.7. Pin diagram of IR sensor**

**4.3.2 Types of IR Sensors**

Infrared sensors are broadly classified into two main types:

∑ **Thermal infrared sensors** – use infrared energy as heat. Their photosensitivity is independent of the wavelength being detected. Thermal detectors do not require cooling but do have slow response times and low detection capabilities.

∑ **Quantum infrared sensors** – provide higher detection performance andfaster response speed. Their photo sensitivity is dependent on wavelength. Quantum detectors have to be cooled in order to obtain accurate measurements.

19

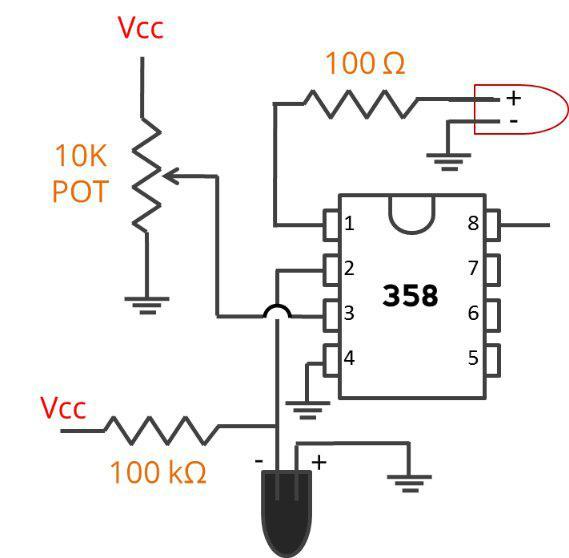
**4.3.3 Construction of IR Sensor**

An infrared sensor circuit is one of the basic and popular sensor module in an electronic device. This sensor is analogous to human’s visionary senses, which can be used to detect obstacles and it is one of the common applications in real time. The circuit diagram for IR sensor is shown in Fig.8. This circuit comprises of the following components,

∑ LM358 IC 2 IR transmitter and receiver pair ∑ Resistors of the range of kilo ohms.

∑ Variable resistors.

∑ LED (Light Emitting Diode).



**Fig.8. Circuit diagram of IR sensor**

**4.3.4 Working Principle**

The principle of an IR sensor working as an Object Detection Sensor can be explained using the following figure. An IR sensor consists of an IR LED and an IR Photodiode, together they are called as Photo Coupler or Opto Coupler. It consists of an IR LED, a photodiode, a potentiometer, an IC

20

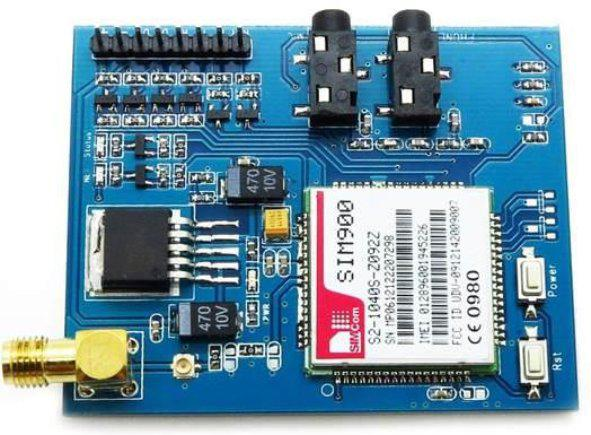
Operational amplifier and an LED. IR LED emits infrared light. The Photodiode detects the infrared light. An IC Op – Amp is used as a voltage comparator. The potentiometer is used to calibrate the output of the sensor according to the requirement. When the light emitted by the IR LED is incident on the photodiode after hitting an object, the resistance of the photodiode falls down from a huge value. One of the inputs of the op-amp is at threshold value set by the potentiometer. The other input to the op-amp is from the photodiode’s series resistor. When the incident radiation is more on the photodiode, the voltage drop across the series resistor will be high. In the IC, both the threshold voltage and the voltage across the series resistor are compared. If the voltage across the resistor series to photodiode is greater than that of the threshold voltage, the output of the IC operational amplifier is high.

**4.4 GSM MODULE**

A GSM modem is a specialized type of modem which accepts a SIM card, and operates over a subscription to a mobile operator, just like a mobile phone. From the mobile operator perspective, a GSM modem looks just like a mobile phone.

When a GSM modem is connected to a computer, this allows the computer to use the GSM modem to communicate over the mobile network. While these GSM modems are most frequently used to provide mobile internet connectivity, many of them can also be used for sending and receiving SMS and MMS messages. The complete hardware of GSM module is shown in Fig.9.

21



**Fig.9. GSM module**

**4.4.1 GSM Module Specifications**

∑ Dual-Band - 900/ 1800 MHz. ∑ GPRS multi-slot - class 10/8. ∑ GPRS mobile station - class B. ∑ Compliant to GSM phase 2/2+. ∑ Dimensions: 24x24x3mm.

∑ Weight: 3.4g.

∑ Control via AT commands. (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)

∑ SIM application toolkit.

∑ Supply voltage range: 12V.

∑ Low power consumption: 1.0mA (sleep mode).

∑ Operation temperature: -40°C to +85 °C.

∑ Fax: Group 3, Class 1.

**4.4.2 Firmware Support**

∑ FOTA. ∑ MMS.

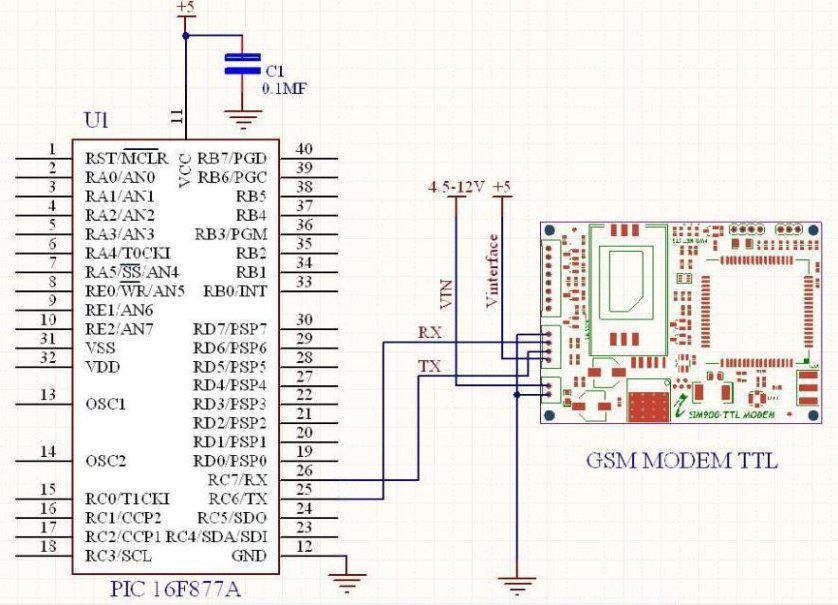
22

∑ Java.

∑ Embedded AT.

**4.4.3 Working of GSM Module**

It is known as Global System for Mobile Communications. This module can be used to give call intimation, messages and MMS. This mainly works by AT commands. The GSM module used here is SIM900. The waste details from the microcontroller will be sent to the mobile number of the waste collector using this module. The GSM module is interfaced with LabVIEW software. The mobile number will be specified in the LabVIEW software. A sim card slot is present in it, which makes this communication possible. Hence immediate disposal of accumulated wastes is made possible. The interfacing of PIC16F877A microcontroller with GSM module is shown in Fig.10.



**Fig.10.Interfacing PIC16F877A microcontroller with GSM module**

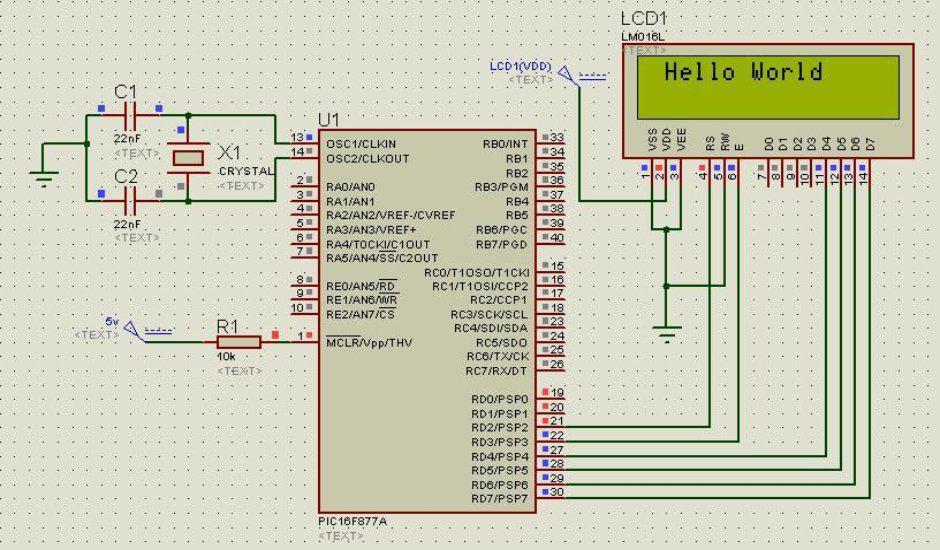
23

**4.5 LIQUID CRYSTAL DISPLAY**

An LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used in DIYs and circuits. The 16×2 translates a display 16 characters per line in 2 such lines.

**4.5.1 Working of Liquid Crystal Display**

It will be installed to the front portion of every dustbin. When the level of waste reaches IR sensor1 of bin1, 50% waste level detection will be displayed in the LCD display. When the level of waste reaches IR sensor2 of bin1, 100% waste level detection will be displayed in the LCD display. Similar procedure follows for bin2 also. So the people who are going to dispose the wastes in the bins become aware of this waste condition and dispose their litters accordingly. So the overflowing of wastes will be avoided. The interfacing of PIC16F877A microcontroller with Liquid Crystal Display is shown in Fig.11.



**Fig.11. Interfacing PIC16F877A microcontroller with Liquid crystal display**

24

**4.5.2 Important Codes of LCD Display**

The important codes of Liquid Crystal Display are shown in Table.2.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **S NO.** |  |  | **Hex Code** |  | **Command to LCD instruction Register** |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **1** |  |  | 01 |  | Clear display screen |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **2** |  | 02 | |  | Return home | |  |
|  |  |  |  |  | |  |  | |  |
|  |  | **3** |  |  | 04 |  | Decrement cursor (shift cursor to left) |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **4** |  | 06 | |  | Increment cursor (shift cursor to right) | |  |
|  |  |  |  |  | |  |  | |  |
|  |  | **5** |  |  | 05 |  | Shift display right |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **6** |  | 07 | |  | Shift display left | |  |
|  |  |  |  |  | |  |  | |  |
|  |  | **7** |  |  | 08 |  | Display off, cursor off |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **8** |  |  | 0A | | Display off, cursor on | |  |
|  |  |  |  |  |  | |  | |  |
|  |  | **9** |  |  | 0C |  | Display on, cursor off |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **10** |  |  | 0E | | Display on, cursor blinking | |  |
|  |  |  |  |  |  | |  | |  |
|  |  | **11** |  |  | 0F |  | Display on, cursor blinking |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **12** |  | 10 | |  | Shift cursor position to left | |  |
|  |  |  |  |  | |  |  | |  |
|  |  | **13** |  |  | 14 |  | Shift cursor position to right |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | **14** |  | 18 | |  | Shift the entire display to the left | |  |
|  |  |  |  |  |  | |  | |  |
|  |  | **15** |  |  | 1C |  | Shift the entire display to the right |  |  |
|  |  |  |  |  |  |  |  |  |  |

**Table.2. Important codes of liquid crystal display**

25

**4.6 BUZZER**

A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric.

**4.6.1 Working Principle of Buzzer**

The buzzer converts the electrical signal it receives into a vibration,which creates a buzzing sound. The higher the signal it receives, the more intense the vibration, and there will be louder sound.

**4.6.2 Synchronization of Buzzer with PIC16F877A Microcontroller**

It provides a beep sound when the level of wastes in the bin reaches 100% full condition. This will alert the people about the filled bin condition. So they will move to the next nearest bin to dispose the wastes. It has three pins. The first pin carries the filled status signal from the microcontroller. The second pin is connected to Vcc. The third pin is connected to ground. The pin diagram of buzzer is shown in Fig.12.



**Fig.12. Pin diagram of buzzer**

26

**4.7 RF MODULE**

The most commonly used RF modules are Bluetooth and Zigbee.

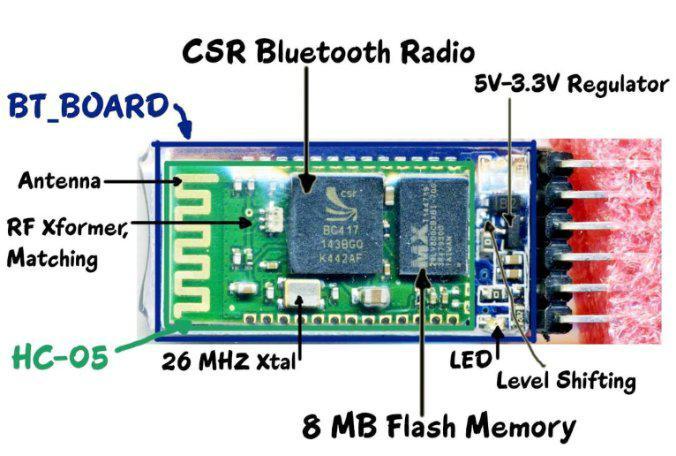
Bluetooth is used for short distance communication (approximately 10 meters).

Zigbee is used for long distance communication (approximately 100 meters). In case of complete hardware deployment, Zigbee is more comfortable. Since this is a prototype deployment, Bluetooth is sufficient in terms of cost and output.

**4.7.1 Bluetooth Module**

The complete hardware for HC-05 Bluetooth module is shown in Fig.13. HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm.



**Fig.13. HC-05 Bluetooth module**

27

**4.7.2 Features**

∑ Typical -80dBm sensitivity.

∑ Up to +4dBm RF transmit power.

∑ Low Power 1.8V Operation, 1.8 to 3.6V I/O. ∑ PIO control.

∑ UART interface with programmable baud rate. ∑ With integrated antenna.

∑ With edge connector.

**4.7.3 Specifications**

∑ Auto-connect to the last device on power as default. ∑ Permit pairing device to connect as default.

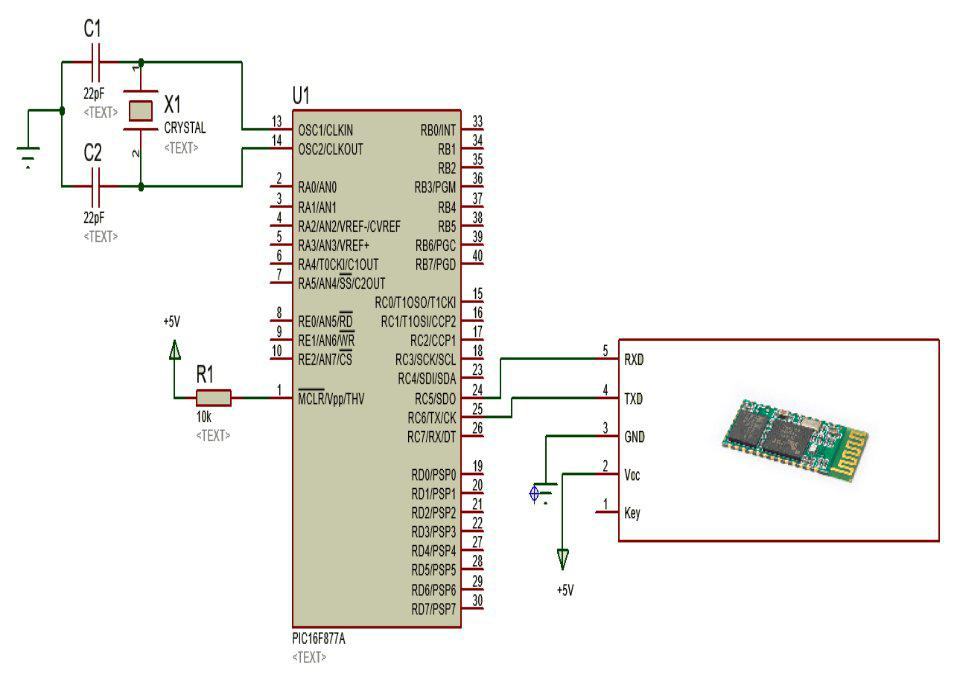
∑ Auto-pairing PINCODE:”0000” as default

∑ Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

**4.7.4 Working of Bluetooth Module**

The Bluetooth module contains a transmitter and a receiver. The data representing the waste details will be transmitted using Bluetooth transmitter into the wireless medium. The same data can be received in the corporation office using Bluetooth receiver (Dongle). This is mainly used for serial data transmission. The serial data is represented by 1, 2, 3 and 4. The data 1 represents the 50% waste level detection of bin1. The data 2 represents the 100% waste level detection of bin 1. The data 3 represents the 50% waste level detection of bin2. The data 4 represents the 100% waste level detection of bin2. These data will be transmitted into the wireless medium by means of Bluetooth module. The interfacing of HC-05 Bluetooth module with PIC16F877A microcontroller is shown in Fig.14.

28



**Fig.14. Interfacing of HC-05 Bluetooth module with PIC16F877A microcontroller**

29

**CHAPTER 5**

**SOFTWARE REQUIREMENTS**

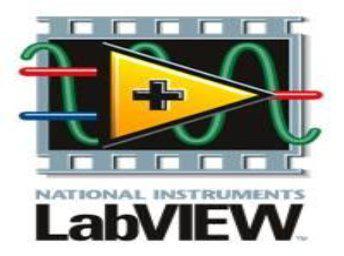
**CHAPTER 5**

**SOFTWARE REQUIREMENTS**

**5.1 INTRODUCTION TO LABVIEW**

Laboratory Virtual Instrument Engineering Workbench (LabVIEW) is a system-design platform and development environment for a visual programming language from National Instruments.

The graphical language is named "G"; not to be confused with G-code. Originally released for the Apple Macintosh in 1986, LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of operating systems (OSs), including Microsoft Windows, various versions of Unix, Linux, and macOS. The latest version of LabVIEW is 2016, released in August 2016. The logo of LabVIEW software is shown in Fig.15.



**Fig.15. LabVIEW software**

**5.2 LABVIEW PROGRAM FOR WASTE MANAGEMENT**

The waste management program is created using LabVIEW software. The serial data that is received using Bluetooth receiver (dongle) is interfaced with LabVIEW software using Ethernet cable. The serial data transmission is achieved in LabVIEW using VISA (Virtual Instrument Software Architecture).

30

**5.2.1 VISA**

The Virtual Instrument Software Architecture (VISA) is a standard for configuring, programming, and troubleshooting instrumentation systems comprising GPIB, VXI, PXI, Serial, Ethernet, and/or USB interfaces. VISA provides the programming interface between the hardware and development environments such as LabVIEW, LabWindows/CVI, and Measurement Studio for Microsoft Visual Studio. NI-VISA is the National Instruments implementation of the VISA I/O standard.

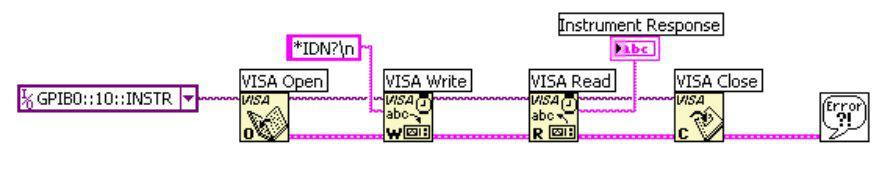
NI-VISA includes software libraries, interactive utilities such as NI I/O Trace and the VISA Interactive Control, and configuration programs through Measurement & Automation Explorer for all your development needs. NI-VISA is standard across the National Instruments product line.

**5.2.2 Applications of VISA**

The various functions of VISA are shown in Fig.16. A typical VISA application would go through the following steps.

1. Open a Session to a given Resource.
2. Do any configuration on the given resource (setting baud rates, termination character, etc...).
3. Perform writes and reads to the device.
4. Close the Session to the Resource.
5. Handle any errors that may have occurred.

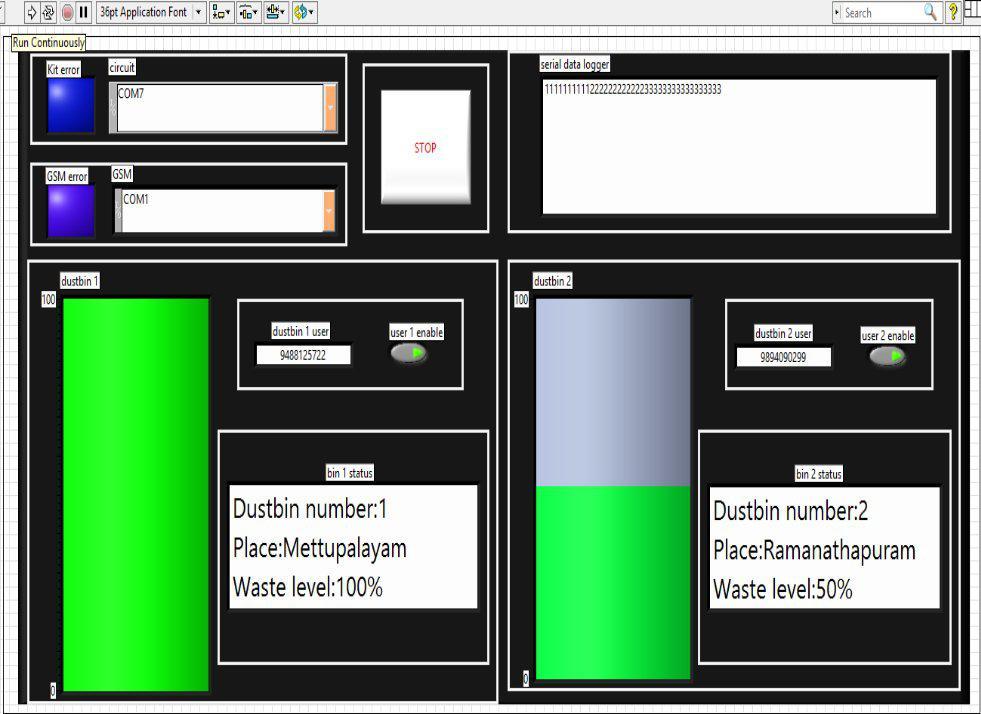
The following is a LabVIEW application that opens a session to a GPIB Instrument, and then queries the device for its response.



**Fig.16. Various functions of VISA**

31

This exact same format would be used in a text based language like C++ or Basic. This code would run on any operating system that supports LabVIEW and NI-VISA. The Front panel indication for smart waste management using LabVIEW is shown in Fig.17.



**Fig.17. Front panel indication for smart waste management using LabVIEW**

The serial data is received in LabVIEW using VISA serial port. Then by using compare operations, we are comparing the serial data with prespecified data and displaying the intimation message accordingly. The front panel indication of this message will be monitored by the officials in the corporation office. Time delay of five to ten seconds will be provided to get the confirmation about waste level.

32

The serial data logger will be programmed in LabVIEW to see the serial data that is received from starting time till end. The GSM module is interfaced with LabVIEW and it is programmed using Attention (AT) commands. The mobile numbers of waste collector and corporation official needs to be entered manually in LabVIEW. By clicking the enable button in front panel, the corresponding waste level and other details will be sent to the respective mobile numbers specified. So the wastes ion bins will be disposed at an early stage and hence efficient waste management is made possible.

**5.3 CREATION OF WASTE MONITORING APPLICATION SETUP IN LABVIEW**

There are three configuration steps to create an application installer setup that will run an executable program.

1. Add the executable to the project:
   1. Navigate to the Project Explorer window of the project.
   2. Right-click **My Computer**, and click **Add File**.
   3. Browse to the desired executable for launching after the installer finishes and click the **Add File** button.

The executable will now appear in the list under My Computer in the Project Explorer window.

1. Include the executable in the installer:
   1. Right-click **Build Specifications** in the Project Explorer window and select **New » Installer**.
   2. Select **Source Files** in the Category list on the left side.
   3. Expand **My Computer** in the Project View by Left-clicking the plus sign.

33

1. Left-click the executable that appears in the expanded list, and verify in the Destination View that the selected destination is the project folder.
2. Left-click the **blue arrow** that is between the Project View and the Destination View to transfer the executable to the destination project folder.

Do not exit out of the Installer Properties window yet.

1. Configure the installer to run the executable after installation is complete:
   1. Select **Advanced** in the Category List on the left side.
   2. In the Launch Executable after Installation portion of the advanced dialog window, there are two buttons on the right side of the

Executable file path display. Left-click the **AddExecutable** button.

(In **LabVIEW 2009** and later, tick the **Run executable at end ofinstaller** box)

3. Select the **executable** from inside the project list, and click **OK**.

The waste monitoring application will be stored in the destination location. Click on the file to run it. This is mainly done for the system where the LabVIEW software is not installed. The execution of this file doesn’t require the installation of LabVIEW software. These details will be monitored and maintained in the corporation office.

34

**CHAPTER 6**

**UTILIZATION OF WEB SERVER**

**CHAPTER 6**

**UTILIZATION OF WEB SERVER**

**6.1 INTRODUCTION TO WEB SERVER USING CLOUD**

A web server is a computer system that processes requests via HTTP, the basic network protocol used to distribute information on the World Wide Web. The term can refer to the entire system, or specifically to the software that accepts and supervises the HTTP requests.

The cloud servers work in the same way as physical servers but the

functions they provide can be very different. When opting for cloud hosting, clients are renting virtual server space rather than renting or purchasing physical servers. They are often paid for by the hour depending on the capacity required at any particular time.

**6.1.1 Benefits of Web Server**

∑ Flexibility and scalability. ∑ Cost-effectiveness

∑ Ease of set up. ∑ Reliability.

**6.2 CREATION OF WEB SERVER USING WEB PUBLISHING TOOL**

It is mainly utilized for waste management system. The steps to create web server using web publishing tool is given below,

* 1. In the front panel of waste management system program, go to **“Tools”** and click **“Web Publishing Tool”** option.

1. The web publishing window pops up. In the **“Vi name”** tab, select the waste management program.

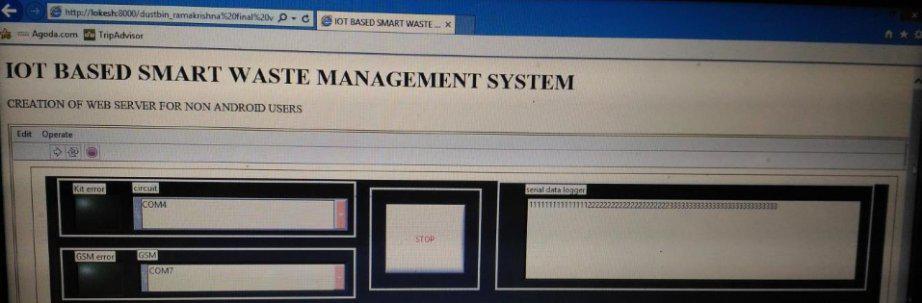
35

1. Since the program needs to be monitored, select the **“Viewing mode”** as **“Embedded”** and click the **“Start web server”** option.
2. The **“Preview in browser”** tab shows the waste details and its information in a miniaturised form and click next.
3. In the next window, we need to specify the Title of the web page as **“IOT**

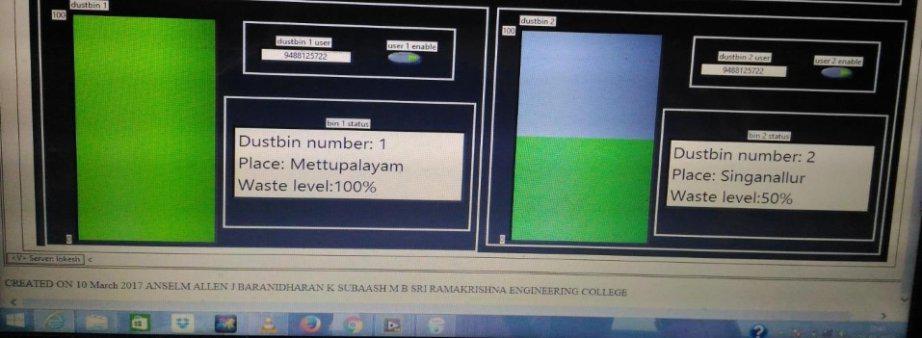
**BASED SMART WASTE MANAGEMENT SYSTEM FOR NON ANDROID USERS”** in the **“Documents title”** tab.

1. Then we need to specify the **“Header**” and **“Footer”** for the web page.
2. Enter the **“URL link”** in the browser and get the web page for smart waste management system using **“Web server”.**

This is mainly created for non android users to know the waste details and its information. The upper portion and lower portion of web page for smart waste management system is shown in Fig.18 and Fig.19.



**Fig.18. Upper portion of web page for smart waste management system**



**Fig.19.Lower portion of web page for smart waste management system**

36

**CHAPTER 7**

**DATA DASHBOARD MOBILE APPLICATION**

**CHAPTER 7**

**DATA DASHBOARD MOBILE APPLICATION**

**7.1 INTRODUCTION TO DATA DASHBOARD**

Data Dashboard allows you to create custom, portable views of National Instruments LabVIEW applications. Using this application, you can create dashboards to display the values of network-published shared variables and deployed LabVIEW Web services on indicators, such as charts, gauges, textboxes, and LEDs. Data Dashboard 2.2 is available for Apple iPad and Android tablets in the iTunes and Google Play Store.

**7.2 SMART WASTE MANAGEMENT APP**

This app can be created and maintained by using the data dashboard mobile application. This is mainly utilized for android users to get the exact status of waste level and its details from their homes. This app decreases the human effort and increases the human comfort.

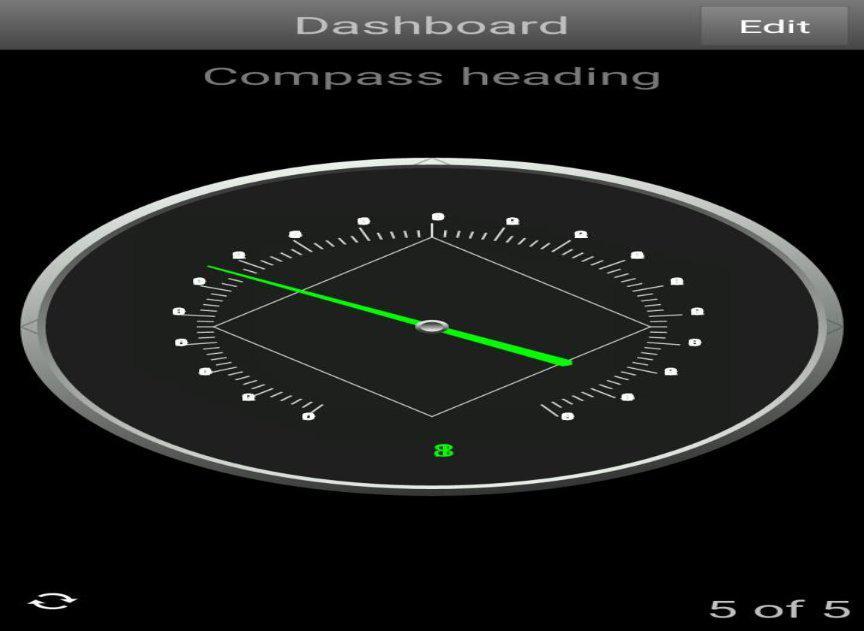
The following steps are involved in creating dashboard for mobile application,

1. Open Data Dashboard on your device. The main view is the dashboard carousel. We can use the dashboard carousel to manage your dashboards. To create a new dashboard tap the **"+"** sign at the bottom of the carousel and select **New Dashboard.**
2. The new dashboard will open. To place an indicator on the dashboard tap the **Controls and Indicators Icon** and select **Indicators**. This will open the indicators palette.
3. To place an indicator on the dashboard, tap the desired indicator and then tap where you want to place it on the page. For this example, we want to drag a **Chart** and a **Numeric** on to the dashboard.
4. For deploying shared variables, click on **File>>New Project** to create a new LabVIEW project.

37

5. Add a shared variable to the project. Right-click **My Computer** in the **Project Explorer** window and select **New>>Variable**.

1. Right-click the shared variable and select **Rename.** Rename the shared variable DDVariable1.
2. Add the variable to the block diagram of waste management program and connect the output of while loop to this variable and save the program.
3. After saving the program, a **“URL link”** will be provided.
4. After performing necessary operations, enter the **“URL link”** in the **“Newserver”** tab of data dashboard and click connect.
5. The browser will open and display the waste details and its location in the mobile phone. The Creation of Smart waste management app by utilizing data dashboard mobile application is shown in Fig.20.
6. Hence efficient waste management is made possible.



**Fig.20. Creation of Smart waste management app by utilizing data dashboard mobile application**

38

**CHAPTER 8**

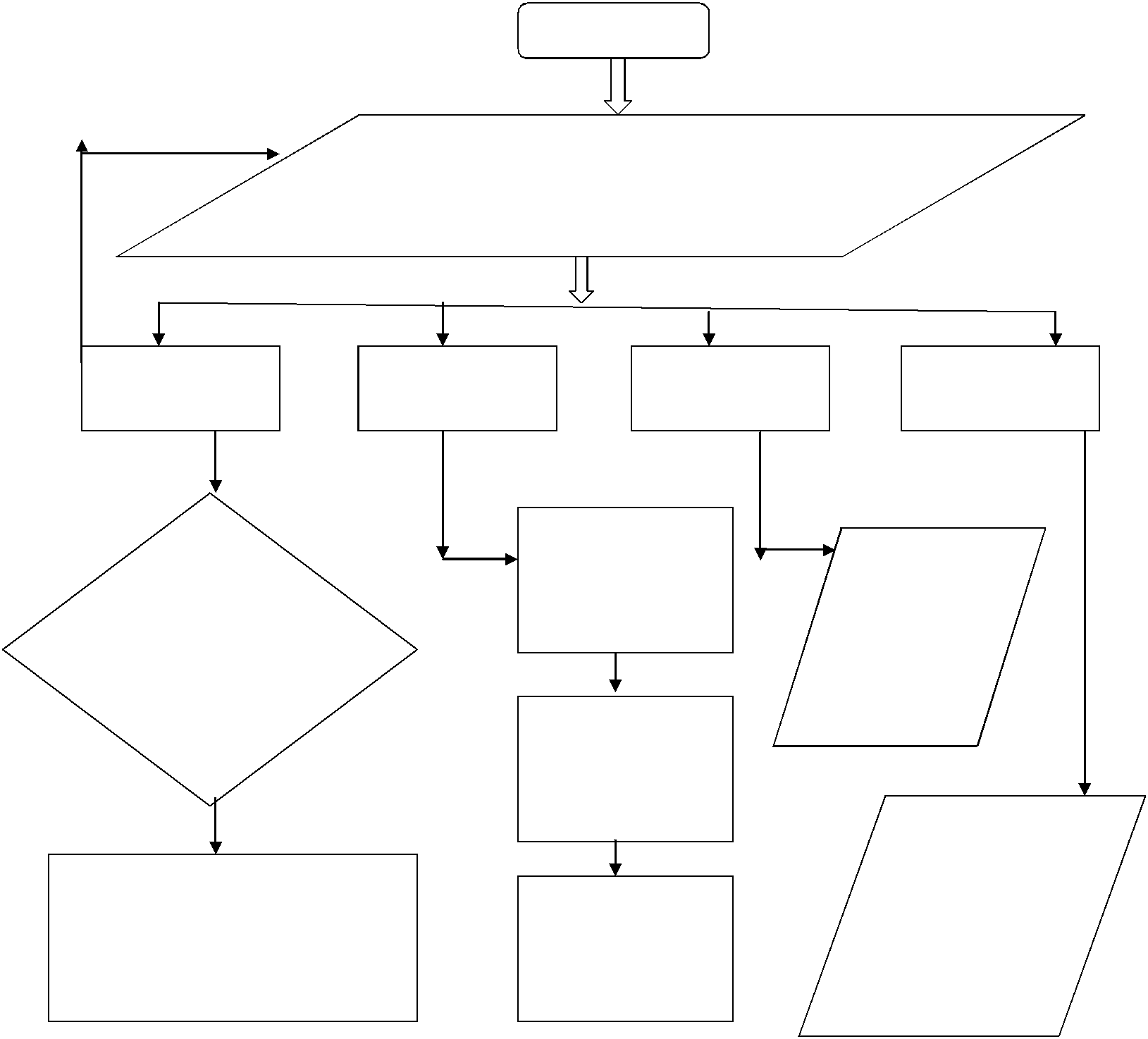
**FLOWCHART FOR WASTE MANAGEMENT**

**CHAPTER 8**

**FLOWCHART FOR WASTE MANAGEMENT**

**8.1 FLOWCHART SETUP FOR SMART WASTE MANAGEMENT SYSTEM**

The flowchart for waste management system is shown in Fig.21.



Start

**No**

IR sensors signals from garbage bins is given to

microcontroller

GSM module RF module LCD display

|  |  |  |
| --- | --- | --- |
|  | PC (LabVIEW) for |  |
| If the input waste | corporation |  |
| officials |  |
| level is equal to |  |
|  |  |
| prespecified |  |  |
| setpoint in |  |  |
| microcontroller | Web server for non |  |
|  |  |
|  | android users |  |

**Yes**

Send the waste levels to the

Mobile App for

mobiles numbers of the waste

android users

collector and corporation officials

Buzzer

Displays the waste level status

Produces beep sound when the bins are filled

**Fig.21. Flowchart for waste management system**

39

When the wastes start accumulating in the bins, the IR sensors searches signal for the corresponding level of waste detection. The corresponding signals from the IR sensors are given to the microcontroller unit. The output signal from the microcontroller corresponds to the waste level detected. Once the bins are completely filled with waste, the buzzer produces a beep sound. The status of waste level will be displayed in the Liquid Crystal Display.

Then the level signals are transmitted from the bin location using RF transmitter and the level signals are received in the corporation office using RF receiver. The RF receiver is interfaced with LabVIEW using Ethernet cable. The waste management program is installed in LabVIEW that will be monitored by the corporation officials. A web server will be utilized for non android users. A mobile application will be created for android users. If the level information crosses the prespecified setpoint in the microcontroller, a message will be given as intimation to the waste collector via GSM module.

40

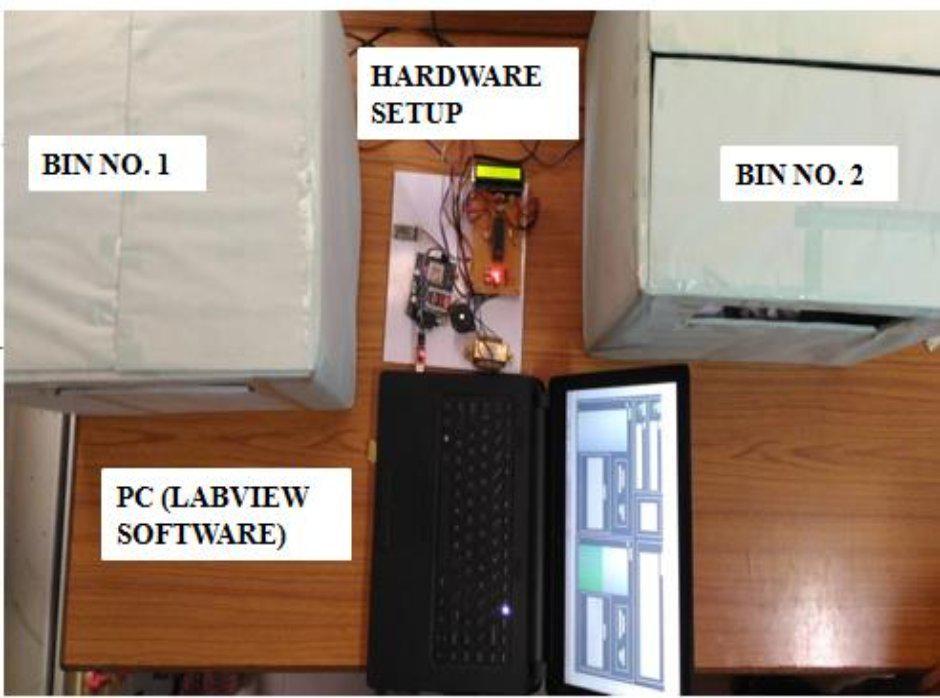
**CHAPTER 9**

**HARDWARE SETUP**

**CHAPTER 9**

**HARDWARE SETUP**

The Hardware setup of Smart waste management system is shown in Fig.22. The two bins are completely filled with waste. So the IR sensors that are installed in respective bins sense the waste levels and send the signal to the PIC16F877A microcontroller unit.

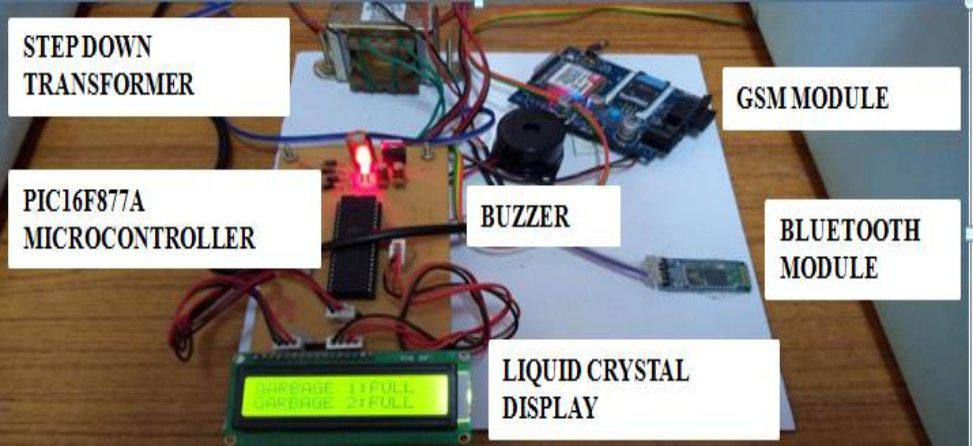


**Fig.22. Hardware setup of Smart waste management system**

Based on the conditions that are programmed in the microcontroller, it sends its output signal to the buzzer, Liquid Crystal Display, RF module and to the GSM module. The waste level status for the two bins will be displayed simultaneously in the LCD display.

41

This will be installed in front of the dustbins. Hence the local residents know about the waste conditions in the respective bins and dispose the litters accordingly. Hence overflowing of waste from the dustbins will be avoided. The LCD indication of waste level in dustbins is shown in Fig.23.



**Fig.23. LCD indication of waste level in dustbins**

42

**CHAPTER 10**

**CONCLUSION AND OUTCOMES**

**CHAPTER 10**

**CONCLUSION AND OUTCOMES**

**10.1 CONCLUSION AND FUTURE SCOPE**

This project addresses the environmental features like durability, affordability, prevention against damage and maintenance issues. This project can contribute a lot towards clean and hygienic environment, in building a smart city. Hence this system avoids overflowing of garbage and promises efficient waste management system.

The future scope is to develop an automated bot which can able to pick up waste in and around the bin, segregate them and put them in respective bins. Then a differentiation can be made between dry trash bin and wet trash bin which collects plastic dry waste and biodegradable waste respectively. To implement this methane and smell sensors can be used. This helps in distinguishing the waste at the source and hence reducing the requirement of manpower.

**10.2 EXPECTED DELIVERABLES**

∑ Accurate waste level detection is done.

∑ Determining the type of wastes are performed.

∑ Waste management program simulation in LabVIEW is done. ∑ Creation of web server for non android users is performed.

∑ Utilization of smart waste management app for android users is done.

**10.3 EXPECTED BENEFITS**

∑ Overflowing of garbage wastes is prevented.

∑ Prevents the spreading of harmful gases like methane.

43

∑ Avoids air pollution and simultaneous monitoring of wastes are done.

∑ By means of mobile application, the local residents know about the waste details and dispose the litters accordingly.

**10.4 NOVELITY OF THE PROJECT**

∑ This “SMART WASTE MANAGEMENT APP” utilizing data dashboard mobile application is user friendly.

∑ This system provides efficient waste management system for Smart city. ∑ This project provides way for “SWACHH BHARAT”.

44

**REFERENCES**

**REFERENCES**

1. Monika K.A., Nikitha Rao and Prapulla S.B., (2016), ‘Smart Dustbin - An Efficient Garbage Monitoring System’, International Journal of Engineering Science and Computing, Vol.6, No.6, pp.7113-7116.
2. Narayan Sharma, Nirman Singha, Tanmoy Dutta, (2015), ‘Smart Bin Implementation for Smart Cities’, International Journal of Scientific & Engineering Research, Vol.6, No.9, pp.787-789.
3. Twinkle Sinha, Mugesh Kumar, Saisharan.P, (2015), ‘Smart Dustbin’, International Journal of Industrial Electronics and Electrical Engineering, SRM University, India, Vol.3, No.5, pp.101-104.
4. Kanchan Mahajan, Chitode J.S., (2014), ‘Waste Bin Monitoring System Using Integrated Technologies’, International Journal of Innovative Research in Science, Engineering and Technology, Vol.3, No.3, pp. 14953-14956.
5. Islam M.S, Arebey M, Hannan M.A, Basri H, (2012), ‘Overview for solid waste bin monitoring and collection system’, Innovation Management and Technology Research , International Conference, Malacca, pp. 258-262.
6. Hassan M.N., Chong T.L., & Rahman M.M., (2012), ‘Solid Waste Management-What’s The Malaysian Position’, Seminar Waste to Energy, University Putra Malaysia.
7. Latifah A., Mohd A.A., & NurIlyana M., (2009), ‘Municipal solid waste management in Malaysia: Practices and challenges of Waste Management’, Vol. 29, pp.2902-2921.

Minimum of 20 references. Chapters of 50 pages

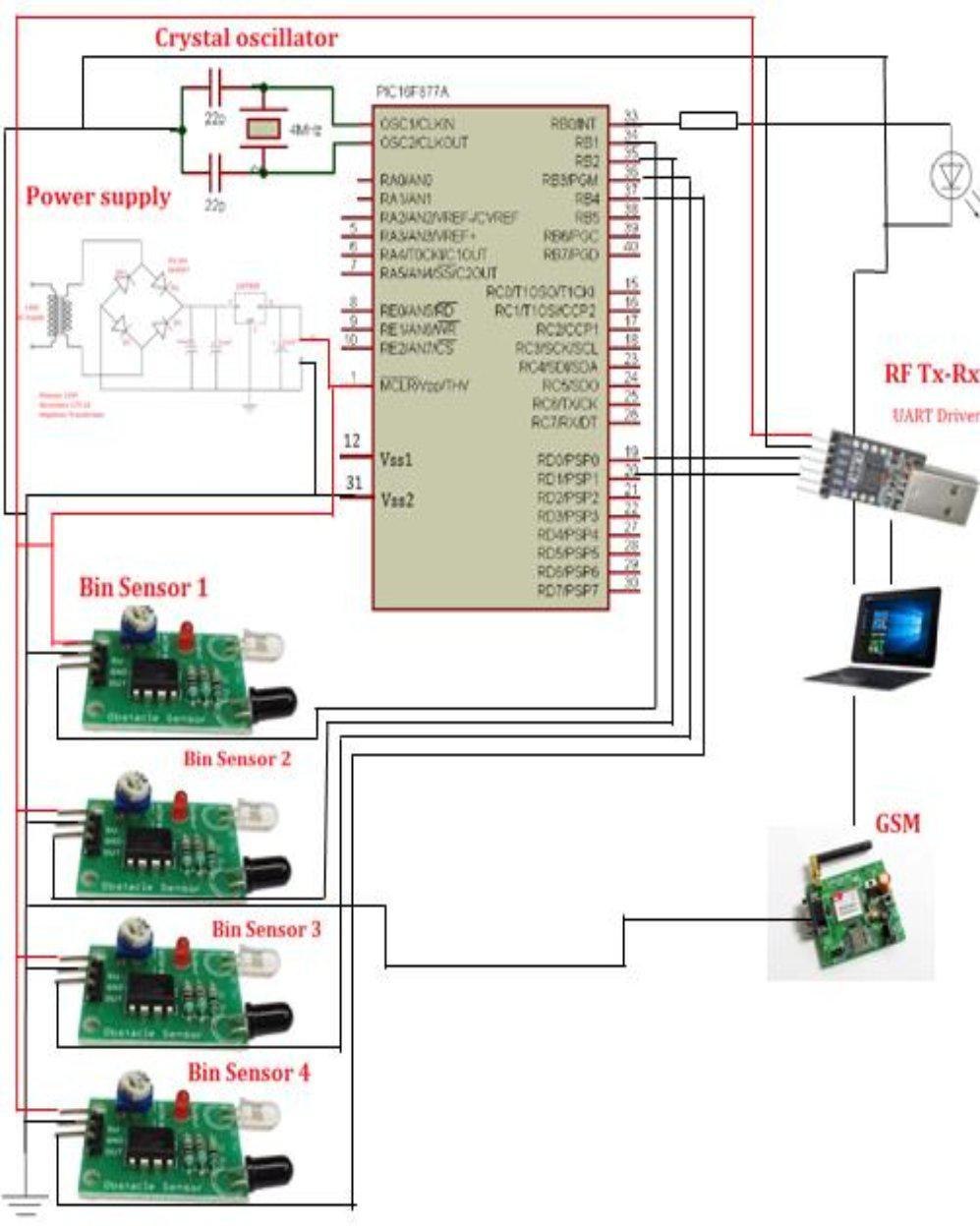
45

**APPENDIX - I**

**OVERALL CIRCUIT DIAGRAM**

**APPENDICES**

**APPENDIX – I OVERALL CIRCUIT DIAGRAM**



**Fig.24. Overall circuit diagram of IOT based smart waste management system**

46

**APPENDIX - II**

**PIC16F877A MICROCONTROLLER PROGRAM**

**APPENDIX – II PIC16F877A MICROCONTROLLER PROGRAM**

#include<pic.h>

/\*

* File: IOT BASED SMART WASTE MANAGEMENT SYSTEM \*/

#include <xc.h>

#pragma config FOSC = HS // Oscillator Selection bits (HS oscillator)

#pragma config WDTE = OFF // Watchdog Timer Enable bit (WDT disabled)

#pragma config PWRTE = OFF // Power-up Timer Enable bit (PWRT

disabled)

#pragma config BOREN = OFF // Brown-out Reset Enable bit (BOR enabled)

#pragma config LVP = OFF // Low-Voltage (Single-Supply) In-Circuit

Serial Programming Enable bit (RB3 is digital I/O, HV on MCLR must be used

for programming)

#pragma config CPD = OFF // Data EEPROM Memory Code Protection bit

(Data EEPROM code protection off)

#pragma config WRT = OFF // Flash Program Memory Write Enable bits

(Write protection off; all program memory may be written to by EECON

control)

#pragma config CP = ON // Flash Program Memory Code Protection bit

(All program memory code-protected)

#define rs RD0

#define rw RD1

#define en RD2

void delay(unsigned int y);

void adc2();

47

void adc1();

void lcd\_val();

void lcd\_com(unsigned char com);

void lcd\_data(unsigned char dat);

void lcd\_display(unsigned char \*data);

void lcd\_condis(const unsigned char \*da,unsigned char num); void Lcd\_enable();

void tx(unsigned char tx\_data);

void transmit(unsigned char \*data);

void timer1();

unsigned int count1=0,g\_detect1=0;

unsigned int count2=0,g\_detect2=0;

unsigned int count3=0,g\_detect3=0;

unsigned int count4=0,g\_detect4=0;

char a1=0,a2=0,a3=0,a4=0;

void interrupt isr()

{

if(TMR0IF==1)

{

TMR0IF=0;

if(RB1==1)//ir1

{

count1++;

if(count1>=30)

{

g\_detect1=1;

a1=1;

* tx('g'); tx('1'); tx('\_'); tx('h'); count1=0;

48

}

}

else

{

count1=0;

g\_detect1=0;

}

if(RB1==1&&RB2==1)//ir2

{

count2++;

if(count2>=30)

{

g\_detect2=1;

* tx('g'); tx('1'); tx('\_'); tx('f');

a2=1;

count2=0;

}

}

else

{

count2=0;

g\_detect2=0;

}

if(RB3==1)//ir3

{

count3++;

if(count3>=30)

{

g\_detect3=1;

49

//tx('g'); tx('2'); tx('\_'); tx('h');

a3=1;

count3=0;

}

}

else

{

count3=0;

g\_detect3=0;

}

if(RB4==1&&RB3==1)//ir4

{

count4++;

if(count4>=30)

{

a4=1;

g\_detect4=1;

* tx('g'); tx('2'); tx('\_'); tx('f'); count4=0;

}

}

else

{

count4=0;

g\_detect4=0;

}

}

}

void delay(unsigned int y)

50

{

while (y--);

}

void transmit(unsigned char \*data)

{ unsigned char d=0; while(data[d]!='\0')

{

while(!TXIF)

{

}

TXREG=data[d];d++;

}

}

void timer0()

{

OPTION\_REG = 0x87;

TMR0 = 61;

TMR0IF = 0;

TMR0IE = 1;

}

void tx(unsigned char tx\_data)

{

while(!TXIF)

{

}

TXREG=tx\_data;

}

void Lcd\_enable()

{

51

en=1;

delay(200);

en=0;

}

void lcd\_com(unsigned char com)

{

unsigned char temp;

PORTD=com&0xf0;

rs=0;

rw=0;

Lcd\_enable();

temp=com<<4;

PORTD=temp&0xf0;

rs=0;

rw=0;

Lcd\_enable();

}

void lcd\_data(unsigned char dat)

{

unsigned char val1;

PORTD=dat&0xf0;

rs=1;

rw=0;

Lcd\_enable();

val1=dat<<4;

PORTD=val1&0xf0;

rs=1;

rw=0;

Lcd\_enable();

52

}

void lcd\_init()

{

lcd\_com(0x02);

lcd\_com(0x2c);

lcd\_com(0x06);

lcd\_com(0x0c);

lcd\_com(0x01);

lcd\_com(0x80);

}

/\*void lcd\_condis(unsigned char \*da,unsigned char num)

{

unsigned char i;

for(i=0;i<num;i++)

{

lcd\_data(da[i]);

delay(300);

}

}\*/

void lcd\_display(unsigned char \*data)

{

unsigned char d=0;

while(data[d]!='\0')

{

lcd\_data(data[d]);

d++;

}

}

53

void main()

{

TRISD=0x00;

TRISB=0xff;

TRISC=0x8f;

PORTD=0x00;

PORTB=0x00;

PORTC=0x00;

TXEN=1;

SPEN=1;

CREN=1;

BRGH=1;

SPBRG=25;

GIE=1;

PEIE=1;

RCIE=1;

lcd\_init();

timer0();

while(1)

{

//delay(50000); delay(50000); delay(50000); /\*\*\*\*\*\* GARBAGE 1\*\*\*\*\*/ if(g\_detect2==1)

{

lcd\_com(0x80);

lcd\_display("GARBAGE 1:FULL ");

if(a2==1)

54

{

tx('1');

a2=0;

}

//tx('1');

//delay(50000);delay(50000);delay(50000);delay(50000);

}

else if(g\_detect1==1)

{

if(a1==1)

{

tx('2');

a1=0;

}

// tx('g'); tx('1'); tx('\_'); tx('h');

lcd\_com(0x80);

lcd\_display("GARBAGE 1:HALF ");

* tx('2');

//delay(50000);delay(50000);delay(50000);delay(50000);

}

else

{

lcd\_com(0x80);

lcd\_display("GARBAGE 1:.......");

}

if(g\_detect4==1)

{

55

if(a4==1)

{

tx('3');

a4=0;

}

* tx('g'); tx('2'); tx('\_'); tx('f'); lcd\_com(0xc0);

lcd\_display("GARBAGE 2:FULL ");

//tx('3');

// delay(50000);delay(50000);delay(50000);delay(50000);

}

else if(g\_detect3==1)

{

if(a3==1)

{

tx('4');

a3=0;

}

* tx('g'); tx('2'); tx('\_'); tx('h');

lcd\_com(0xc0);

lcd\_display("GARBAGE 2:HALF ");

//tx('4');

//delay(50000);delay(50000);delay(50000);delay(50000);

}

else

{

lcd\_com(0xc0);

lcd\_display("GARBAGE 2:........"); }}}

56

**APPENDIX - III**

**PIC16F877A MICROCONTROLLER DATASHEET**

**APPENDIX-3 PIC16F877A MICROCONTROLLER DATASHEET**



**PIC16F87XA**

**Data Sheet**

28/40/44-Pin Enhanced Flash

Microcontrollers



|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

57

**PIC16F87XA**



**28/40/44-Pin Enhanced Flash Microcontrollers**



**Devices Included in this Data Sheet:**

• PIC16F873A • PIC16F876A

• PIC16F874A • PIC16F877A

**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 other 28-pin or 40/44-pin

PIC16CXXX and PIC16FXXX microcontrollers

**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

Synchronous Serial Port (SSP) with SPI™ (Master mode) and I2C™ (Master/Slave)

Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection

Parallel Slave Port (PSP) – 8 bits wide with external RD, WR and CS controls (40/44-pin only)

Brown-out detection circuitry for Brown-out Reset (BOR)

|  |  |  |
| --- | --- | --- |
| **Program Memory** | **Data** |  |
|  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Device** |  |  |  |  | **# Single Word** | |  | **SRAM** | | **EEPROM** | **I/O** |  |
|  |  |  | **Bytes** | |  | **Instructions** | |  | **(Bytes)** | | **(Bytes)** |  |  |
|  | PIC16F873A |  | 7.2K | | 4096 | |  | 192 | |  | 128 | 22 |  |
|  | PIC16F874A |  | 7.2K | |  | 4096 |  | 192 | |  |  | 33 |  |
|  |  |  |  |  | 128 |  |
|  |  |  |  |  |  | |  |  | 368 |  |  |  |  |
|  | PIC16F876A |  | 14.3K |  |  | 8192 |  |  |  | 256 | 22 |  |
|  | PIC16F877A |  | 14.3K |  |  | 8192 |  |  | 368 |  | 256 | 33 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

2003 Microchip Technology Inc.

**Analog Features:**

10-bit, up to 8-channel Analog-to-Digital Converter (A/D)

Brown-out Reset (BOR)

Analog Comparator module with:

Two analog comparators

Programmable on-chip voltage reference (VREF) module

Programmable input multiplexing from device inputs and internal voltage reference

Comparator outputs are externally accessible

**Special Microcontroller Features:**

100,000 erase/write cycle Enhanced Flash program memory typical

1,000,000 erase/write cycle Data EEPROM memory typical

Data EEPROM Retention > 40 years Self-reprogrammable under software control

In-Circuit Serial Programming™ (ICSP™) via two pins

Single-supply 5V In-Circuit Serial Programming

Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation

Programmable code protection

Power saving Sleep mode

Selectable oscillator options

In-Circuit Debug (ICD) via two pins

**CMOS Technology:**

Low-power, high-speed Flash/EEPROM technology

Fully static design

Wide operating voltage range (2.0V to 5.5V)

Commercial and Industrial temperature ranges

Low-power consumption

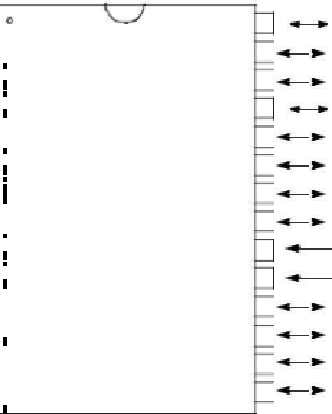
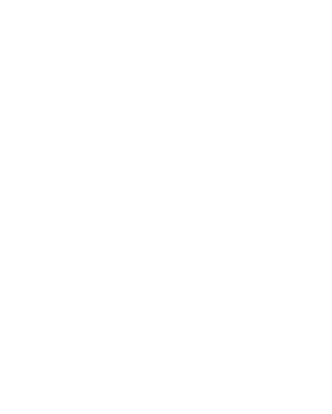
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **MSSP** | | | | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| **10-bit** | | **CCP** | |  |  |  |  |  |  |  |  | **Timers** | |  |  |  |
|  |  |  | **Master** | |  | **USART** | | **Comparators** |  |  |
| **A/D (ch)** | | **(PWM)** | | **SPI** | |  | **I2C** | |  |  |  | **8/16-bit** | |  |  |  |
| 5 |  | 2 |  | Yes | |  | Yes | |  | Yes | | 2/1 |  | 2 |  |  |
|  |  |  |  | Yes | |  | Yes | |  | Yes | |  |  |  |  |  |
| 8 |  | 2 |  |  |  | 2/1 |  | 2 |  |  |
| 5 |  | 2 |  | Yes | |  | Yes | |  | Yes | | 2/1 |  | 2 |  |  |
| 8 |  | 2 |  | Yes | |  | Yes | |  | Yes | | 2/1 |  | 2 |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | DS39582B | | |  |  |
| 58 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**PIC16F87XA**



**Pin Diagrams**

**28-Pin PDIP, SOIC, SSOP**



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  | 1 | 28 |  | RB7/PGD |  |
|  |  |  | MCLR/VPP |  |  | |  |  | 27 |  | RB6/PGC |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  | RA0/AN0 |  |  |  |  |  |  | 26 |  | RB5 |  |
|  | RA1/AN1 | | |  |  |  |  |  | 3 |  |  |
| RA2/AN2/VREF-/CVREF | | | |  |  |  |  |  | 4 | 25 |  | RB4 |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  | 24 |  | RB3/PGM |  |
|  |  |  | RA3/AN3/VREF+ |  |  |  |  |  |  | 23 |  | RB2 |  |
| RA4/T0CKI/C1OUT | | | |  |  |  |  |  | 6 |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | 22 |  | RB1 |  |
| RA5/AN4/SS/C2OUT | | | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | 7 | 21 |  | RB0/INT |  |
| OSC1/CLKI | | | |  | | |  | | | 20 |  | VDD |  |
|  | |  |  |  | 9 |  |  |
|  |  |  |  |  |  |
| OSC2/CLKO | | | |  | | |  |  | 10 | 19 |  | VSS |  |
|  |  |  |
|  |  |  |  |  |  |
| RC0/T1OSO/T1CKI | | | |  |  |  |  |  | 11 | 18 |  | RC7/RX/DT |  |
|  |  |  |  |  | 17 |  | RC6/TX/CK |  |
|  |  | RC1/T1OSI/CCP2 | |  |  |  | |  | 12 |  |  |
|  | RC2/CCP1 | | |  | | | | | 13 | 16 |  | RC5/SDO |  |
|  |  |  |  |  |  |  |  |
| RC3/SCK/SCL | | | |  | | | | | 14 | 15 |  | RC4/SDI/SDA |  |
|  |  |  |  |  |  |  |



**28-Pin QFN**

RA2/AN2/VREF-/CVREF

RA3/AN3/VREF+

RA4/T0CKI/C1OUT

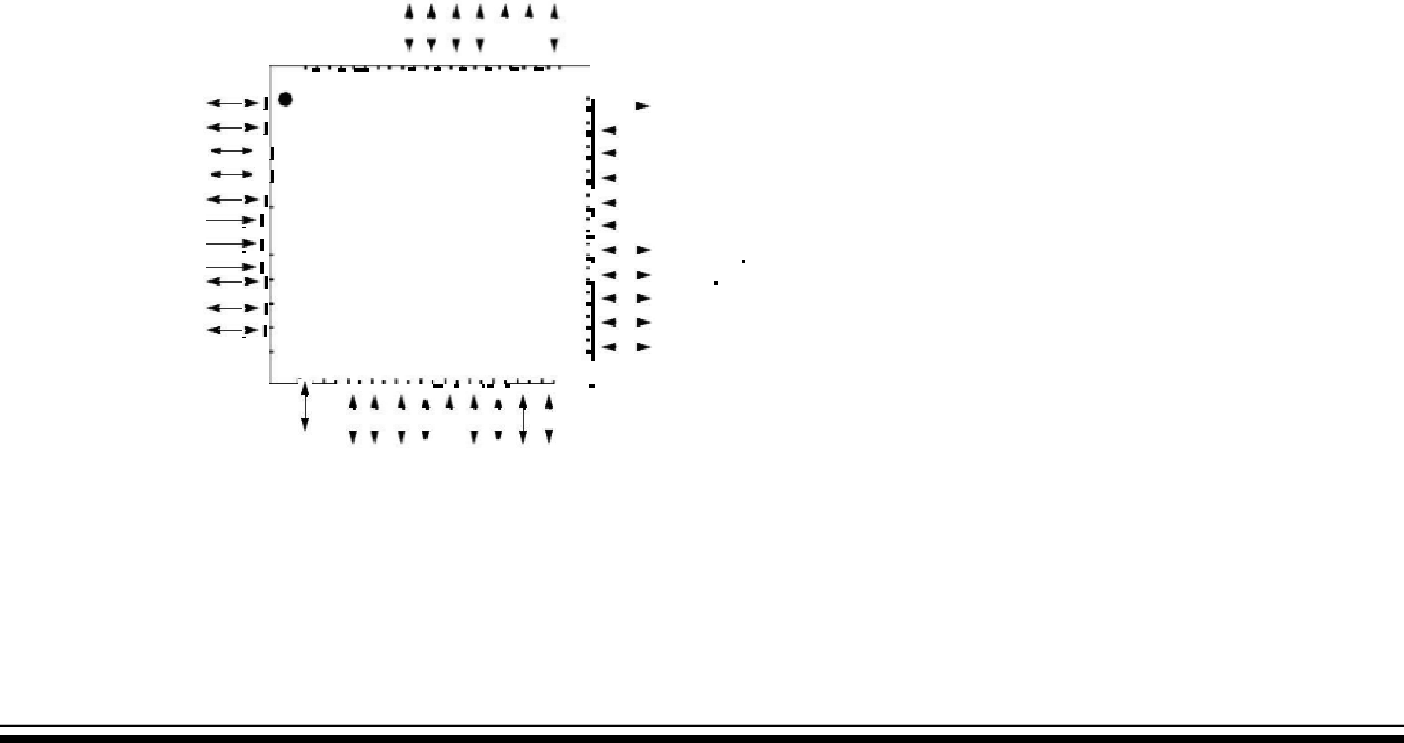
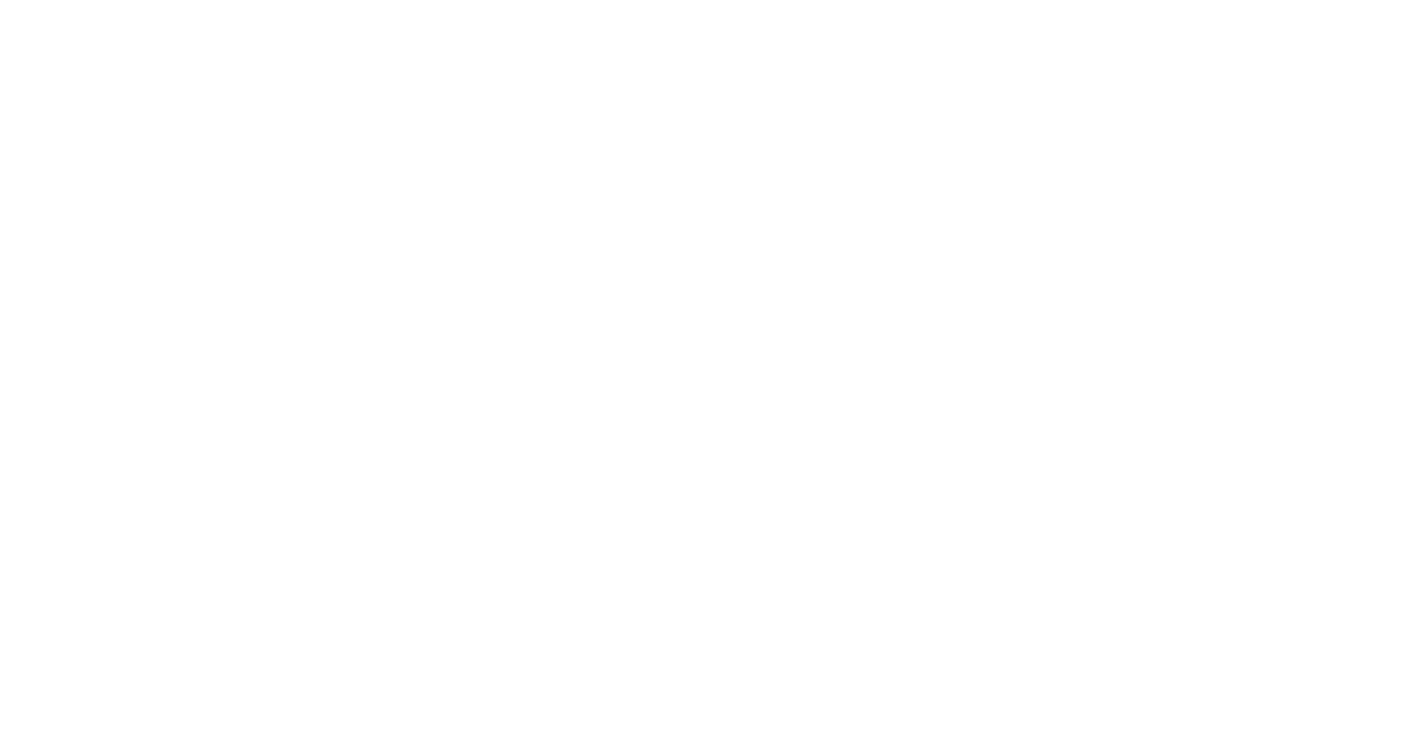
RA5/AN4/SS/C2OUT

VSS

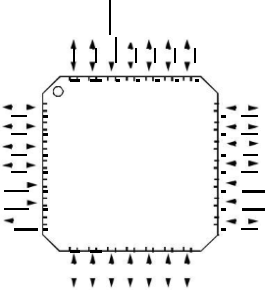
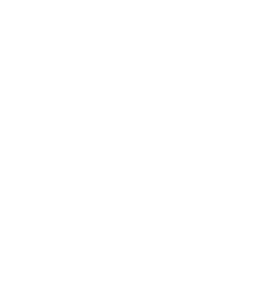
OSC1/CLKI

OSC2/CLKO

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **44-Pin QFN** |  |  |  |  |  |  | RC6/TX/CKRC5/SDO | | | | RC4/SDI/SDA | | |  |  |  | RD2/PSP2RD1/PSP1RD0/PSP0RC3/SCK/SCLRC2/CCP1RC1/T1OSI/CCP2RC0/T1OSO/T1CKI | | | | | | | | | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | |  | |  |  |  | |  | | | | | | | | | |  | | | | | |  | | |  | | | | | | |  | | |  | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  | | | | | | | | | | | | | | | | | |  |
|  | | | | | | |  |  |  |  | | |  | | |  |  | | |  | |  | |  | | |  | | |  |  | |  | |  | | |  |  |  |  | |  | | | | | | | | | | | | | | |  |
|  |  |  |  | 44 | | | | 43 | | | 4241403938373635 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 34 | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RC7/RX/DT |  |  |  |  |  |  |  |  |  |  |  |  | OSC2/CLKO | | | | | | | | |  |
| 1 | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 33 | | | |  |  |  |  |  |  |
| RD4/PSP4 |  |  |  | 2 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 32 | | | |  |  |  |  |  | OSC1/CLKI | | | | | | | | |  |
| RD5/PSP5 |  |  |  | 3 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 31 | | | |  |  |  |  |  | VSS | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RD6/PSP6 |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 30 | | | |  |  |  |  |  | VSS | | | | | | | | |  |
| RD7/PSP7 |  |  |  | 5 | |  |  |  |  |  | **PIC16F874A** | | | | | | | | | | | | | | | | | | | | | | | | | | |  |  | 29 | | | |  |  |  |  |  | VDD | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | VDD | | | | | | | | |  |
| VSS |  |  | 6 |  |  |  |  |  |  |  | **PIC16F877A** | | | | | | | | | | | | | | | | | | | | | | | | | | |  |  | 28 | | | |  |  |  |  |  |  |
| VDD |  |  | 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RE2/ | | | CS | | /AN7 | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
| VDD |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 27 | | | | |  |  |  |  | RE1/ | | WR | |  | | /AN6 | | |  |
| RB0/INT |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  | | | |  | |  |  |  |
|  |  |  | 10 | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 25 | | | |  |  |  |  |  | RE0/RD/AN5 | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RB1 |  | |  |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 24 | | | |  |  |  |  |  | RA5/AN4/SS/C2OUT | | | | | | | | |  |
| RB2 |  |  |  | 11 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 23 |  |  |  |  |  | RA4/T0CKI/C1OUT | | | | | | | | |  |
| RB3/PGM | | | 12 | | | |  |  | 4 |  | 5 | |  | 6 | | |  |  |  | 7 | |  |  | 8 | | | |  | 9 | | |  | 0 | | |  | 21 | |  | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 22 | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | | |  | | | | |  |  | | | |  | | |  | |  | | | |  | | | |  | | RA2/AN2/V -/CVREFREF | |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | NCRB4RB5RB6/PGC | | | | | | | | | | | RB7/PGD | | | | |  | | MCLR/VPPRA0/AN0RA1/AN1 | | | | | | | | | | | | RA3/AN3/V +REF | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



|  |  |
| --- | --- |
| RA1/NRA0/AN0MCLR/P | RB7/PGRB6/GRB5RB4 |



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |
| 28 | 2726 | | 25 | | 242322 | | | | 21 |  |
| 1 |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  | 20 |  |

|  |  |  |
| --- | --- | --- |
| 43 | **PIC16F873A** | 1819 |
| 5 | **PIC16F876A** | 17 |

1. 16
2. 15

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | 11 | | 12 | | 13 | | 14 | |
|  | |  |  |  |  |  |  |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  |  | |  |
| RC0/T1OSO/T1CKI | | RC1/T1OSI/CCP2RC2/CCP1 | | | RC3/SCK/SCL | | RC4/SDI/SDA | | RC5/SDORC6/TX/CK | | | |

RB3/PGM

RB2

RB1

RB0/INT

VDD

VSS

RC7/RX/DT

DS39582B-page 2 2003 Microchip Technology Inc.

59

**PIC16F87XA**



**Pin Diagrams (Continued)**

**40-Pin PDIP**



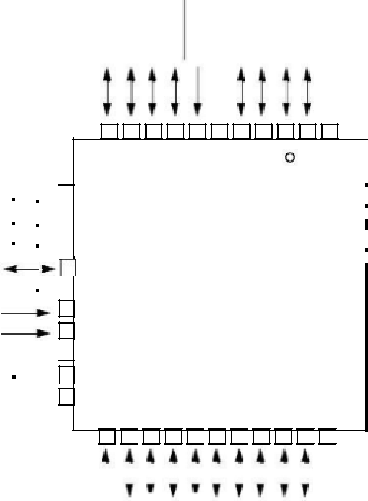
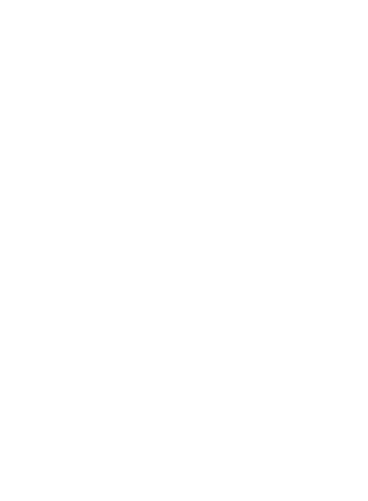
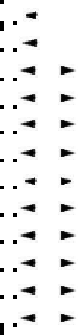
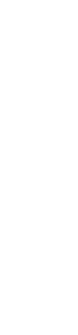
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | MCLR/VPP | | | | | | | |  | |  |  | |  |  | 1 |  |
|  |  | RA0/AN0 | | | | | | | |  |  |  |  |  | |  | 2 |  |
|  |  | RA1/AN1 | | | | | | | |  |  |  |  |  | |  | 3 |  |
| RA2/AN2/VREF-/CVREF | | | | | | | | | |  |  |  |  |  | |  | 4 |  |
| RA3/AN3/VREF+ | | | | | | | | | |  |  |  |  |  | |  | 5 |  |
| RA4/T0CKI/C1OUT | | | | | | | | | |  |  |  |  | |  |  | 6 |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  | |  |  | 7 |  |
| RA5/AN4/SS/C2OUT | | | | | | | | | |  |  |
|  | | |  | |  |  |  |  |  |  | |  | | | | | |  |
| RE0/RD/AN5 | | | | | | | | | |  |  |  |  | |  |  | 8 |  |
|  | | |  | | |  |  |  |  |  |  |  |  | |  |  | 9 |  |
| RE1/WR/AN6 | | | | | | | | | |  |  |
|  | | | | | |  |  |  |  |  | |  | | | | | |  |
| RE2/CS/AN7 | | | | | | | | | |  |  | |  |  | |  | 10 |  |
|  |  |  |  |  |  |  | VDD |  | |  | |  | |  | |  | 11 |  |
|  |  |  |  |  |  |  | VSS | | |  | | | | | |  | 12 |  |
| OSC1/CLKI | | | | | | | | |  |  | | |  |  | |  | 13 |  |
| OSC2/CLKO | | | | | | | | | |  |  | |  |  | |  | 14 |  |
| RC0/T1OSO/T1CKI | | | | | | | | | |  |  | |  |  | |  | 15 |  |
| RC1/T1OSI/CCP2 | | | | | | | | | |  |  | |  |  | |  | 16 |  |
|  |  | RC2/CCP1 | | | | | | | |  |  | |  |  | |  | 17 |  |
| RC3/SCK/SCL | | | | | | | | | |  |  | |  |  | |  | 18 |  |
|  |  | RD0/PSP0 | | | | | | | |  |  | |  |  | |  | 19 |  |
|  |  | RD1/PSP1 | | | | | | | |  |  | |  | |  |  | 20 |  |



|  |
| --- |
| **7A 87 4A/ 87 6F C1 PI** |

1.  RB7/PGD
2.  RB6/PGC
3.  RB5
4.  RB4
5.  RB3/PGM
6.  RB2
7.  RB1
8.  RB0/INT

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 32 |  |  |  |  |  |  | VDD |  |  |  |
|  |  |  |  |
| 31 |  |  |  |  |  |  | VSS |  |  |  |
|  |  |  |  |  |  |  |  |  |
| 30 |  |  |  |  |  |  | RD7/PSP7 |  |  |  |
| 29 |  |  |  |  |  |  | RD6/PSP6 |  |  |  |
| 28 |  |  |  |  |  |  | RD5/PSP5 | NCRB4RB5 | RB6/PGCRB7/PGDNCMCLR/VPPRA0/AN0RA1/AN1RA2/AN2/V-REF/CVREFRA3/AN3/V+REF |  |
| 27 |  |  |  |  |  |  | RD4/PSP4 |  |
| 26 |  |  |  |  |  |  | RC7/RX/DT |  |
|  |  |  |  |  |  |  |  |  |
| 25 |  |  |  |  |  |  | RC6/TX/CK |  |  |  |
| 24 |  |  |  |  |  |  | RC5/SDO |  |  |  |
| 23 |  |  |  |  |  |  | RC4/SDI/SDA |  |  |  |
| 22 |  |  |  |  |  |  | RD3/PSP3 |  |  |  |
| 21 |  |  |  |  |  |  | RD2/PSP2 |  |  |  |



**44-Pin PLCC**

RA4/T0CKI/C1OUT 

RA5/AN4/SS/C2OUT

 RE0/RD/AN5

 RE1/WR/AN6

RE2/CS/AN7 

VDD

VSS

OSC1/CLKI 

OSC2/CLKO 

RC0/T1OSO/T1CK1 

NC

|  |  |  |  |
| --- | --- | --- | --- |
| 65432144434241 7 | 8 | 40 |  |
| 9 |  |  |
|  |  |  |

10

11 **PIC16F874A**

12 **PIC16F877A**

13

14

15

16

17

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1891 | 20212 | 322 | 526272 | 82 |

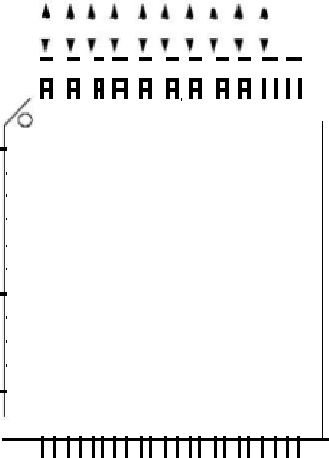
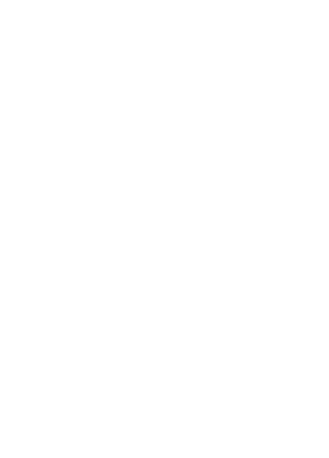
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 39 |  |  |  |  | RB3/PGM |  |
|  |  |  |
| 37 |  |  |  |  | RB |  |
|  |  |  |  |  |
| 36 |  |  |  |  | RB0/INT |  |
|  |  |  |  |
| 35 |  |  |  |  | VDD |  |
| 34 |  |  |  |  | VSS |  |



1.  RD7/PSP7
2.  RD6/PSP6
3.  RD5/PSP5
4.  RD4/PSP4
5.  RC7/RX/DT

**44-Pin TQFP**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RC6/TX/CKRC5/SDO | | | |  |  |  | RC4/SDI/SDARD3/PSP3RD2/PSP2 | | |  |  |  | RD0/PSP0 | |  |  | RC2/CCP12 | | |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | NC | RC1/T1OSI/CCP2RC2/CCP1 | | |  | RC3/SCK/SCLRD0/PSP0RD1/PSP | | RD2/PSP2RD3/PSP3 | |  | RC4/SDI/SDARC5/SDORC6/TX/CK NC | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RC7/RX/DT | | 44 | | | | | |  | 43 | |  |
|  |  |  | 1 | |  |  |  |  |  |
| RD4/PSP4 | |  |  |  | | 2 |  |  |  |  |  |
| RD5/PSP5 | |  |  |  | | 3 |  |  |  |  |  |
| RD6/PSP6 | |  |  |  | | 4 |  |  |  |  |  |
|  |  | |  |  |  |
| RD7/PSP7 | |  |  |  | | 5 |  |  |  |  |  |
| VSS |  |  |  |  | | 6 |  |  |  |  |  |
| VDD |  |  |  |  | 7 | |  |  |  |  |  |
| RB0/INT | |  |  |  | | 8 |  |  |  |  |  |
|  |  | |  |  |  |
| RB1 | |  |  |  | | 9 |  |  |  |  |  |
| RB2 | |  |  |  | | 10 |  |  |  |  |  |
| RB3/PGM | |  |  |  |  | 11 |  |  |  |  |  |
|  |  | 12 | | | | | | |  |  |  |
|  |  |  |  |  |  |  |  |  | 13 | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4241 | 40 | 39 | 38 | 3736 |

**PIC16F874A**

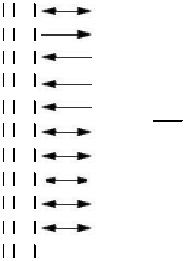
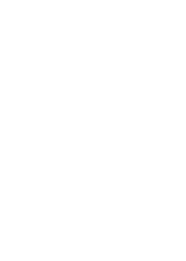
**PIC16F877A**

|  |  |
| --- | --- |
| 14151617 | 181920 |



|  |
| --- |
| 35 |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 34 | | |  | NC | | | | | | |  |
| 33 | | |  |  |
| 32 | | |  | RC0/T1OSO/T1CKI | | | | | | |  |
|  |  |
| 31 | | |  | OSC2/CLKO | | | | | | |  |
|  |  |
| 30 | | |  | OSC1/CLKI | | | | | | |  |
|  |  |
| 29 | | |  | VSS | | | | | | |  |
|  |  |
| 28 | | |  | VDD | | | | | | |  |
|  |  |
|  |  |
| 27 | | |  | RE2/ | | CS/ | AN7 | | | |  |
|  |  |
| 26 | | |  | RE1/ | WR | | | /AN6 | | |  |
|  |  |
|  |  |
|  |  | | |  |
| 25 | | |  | RE0/RD/A | | | | | N5 |  |  |
|  |  |
|  |  |  |
| 24 | | |  | RA5/AN4/SS/C2OUT | | | | | | |  |
|  |  |
| 23 | | |  | RA4/T0CKI/C1OUT | | | | | | |  |
|  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| 2122 | | | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| NCNCRB4RB5RB6/PGCRB7/PGD |  | MCLR/VPP |  | RA0/AN0RA1/AN1 | RA2/AN2/V-/CVRA3/AN3/V+REFREFREF |  |
|  |  |
|  |  |  |
|  |  |  |  |  |  |



|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

60

**DEVICE OVERVIEW**

This document contains device specific information about the following devices:

PIC16F873A

PIC16F874A

PIC16F876A

PIC16F877A

PIC16F873A/876A devices are available only in 28-pin packages, while PIC16F874A/877A devices are avail-able in 40-pin and 44-pin packages. All devices in the PIC16F87XA family share common architecture with the following differences:

The PIC16F873A and PIC16F874A have one-half of the total on-chip memory of the PIC16F876A and PIC16F877A

The 28-pin devices have three I/O ports, while the 40/44-pin devices have five

The 28-pin devices have fourteen interrupts, while the 40/44-pin devices have fifteen

The 28-pin devices have five A/D input channels, while the 40/44-pin devices have eight

The Parallel Slave Port is implemented only on the 40/44-pin devices

The available features are summarized in Table 1-1. Block diagrams of the PIC16F873A/876A and PIC16F874A/877A devices are provided in Figure 1-1 and Figure 1-2, respectively. The pinouts for these device families are listed in Table 1-2 and Table 1-3.

Additional information may be found in the PICmicro® Mid-Range Reference Manual (DS33023), which may be obtained from your local Microchip Sales Represen-tative or downloaded from the Microchip web site. The Reference Manual should be considered a complemen-tary document to this data sheet and is highly recom-mended reading for a better understanding of the device architecture and operation of the peripheral modules.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TABLE 1-1:** | **PIC16F87XA DEVICE FEATURES** | |  |  |  |
|  | |  |  |  |  |
| **Key Features** | | **PIC16F873A** | **PIC16F874A** | **PIC16F876A** | **PIC16F877A** |
|  | |  |  |  |  |
|  | |  |  |  |  |
| Operating Frequency | | DC – 20 MHz | DC – 20 MHz | DC – 20 MHz | DC – 20 MHz |
|  | |  |  |  |  |
| Resets (and Delays) | | POR, BOR | POR, BOR | POR, BOR | POR, BOR |
|  |  | (PWRT, OST) | (PWRT, OST) | (PWRT, OST) | (PWRT, OST) |
|  | |  |  |  |  |
| Flash Program Memory | | 4K | 4K | 8K | 8K |
| (14-bit words) |  |  |  |  |  |
|  | |  |  |  |  |
| Data Memory (bytes) | | 192 | 192 | 368 | 368 |
|  | |  |  |  |  |
| EEPROM Data Memory (bytes) | | 128 | 128 | 256 | 256 |
|  |  |  |  |  |  |
| Interrupts |  | 14 | 15 | 14 | 15 |
|  |  |  |  |  |  |
| I/O Ports |  | Ports A, B, C | Ports A, B, C, D, E | Ports A, B, C | Ports A, B, C, D, E |
|  |  |  |  |  |  |
| Timers |  | 3 | 3 | 3 | 3 |
|  | |  |  |  |  |
| Capture/Compare/PWM modules | | 2 | 2 | 2 | 2 |
|  | |  |  |  |  |
| Serial Communications | | MSSP, USART | MSSP, USART | MSSP, USART | MSSP, USART |
|  | |  |  |  |  |
| Parallel Communications | | — | PSP | — | PSP |
|  | |  |  |  |  |
| 10-bit Analog-to-Digital Module | | 5 input channels | 8 input channels | 5 input channels | 8 input channels |
|  | |  |  |  |  |
| Analog Comparators | | 2 | 2 | 2 | 2 |
|  |  |  |  |  |  |
| Instruction Set |  | 35 Instructions | 35 Instructions | 35 Instructions | 35 Instructions |
|  |  |  |  |  |  |
| Packages |  | 28-pin PDIP | 40-pin PDIP | 28-pin PDIP | 40-pin PDIP |
|  |  | 28-pin SOIC | 44-pin PLCC | 28-pin SOIC | 44-pin PLCC |
|  |  | 28-pin SSOP | 44-pin TQFP | 28-pin SSOP | 44-pin TQFP |
|  |  | 28-pin QFN | 44-pin QFN | 28-pin QFN | 44-pin QFN |
|  |  |  |  |  |  |



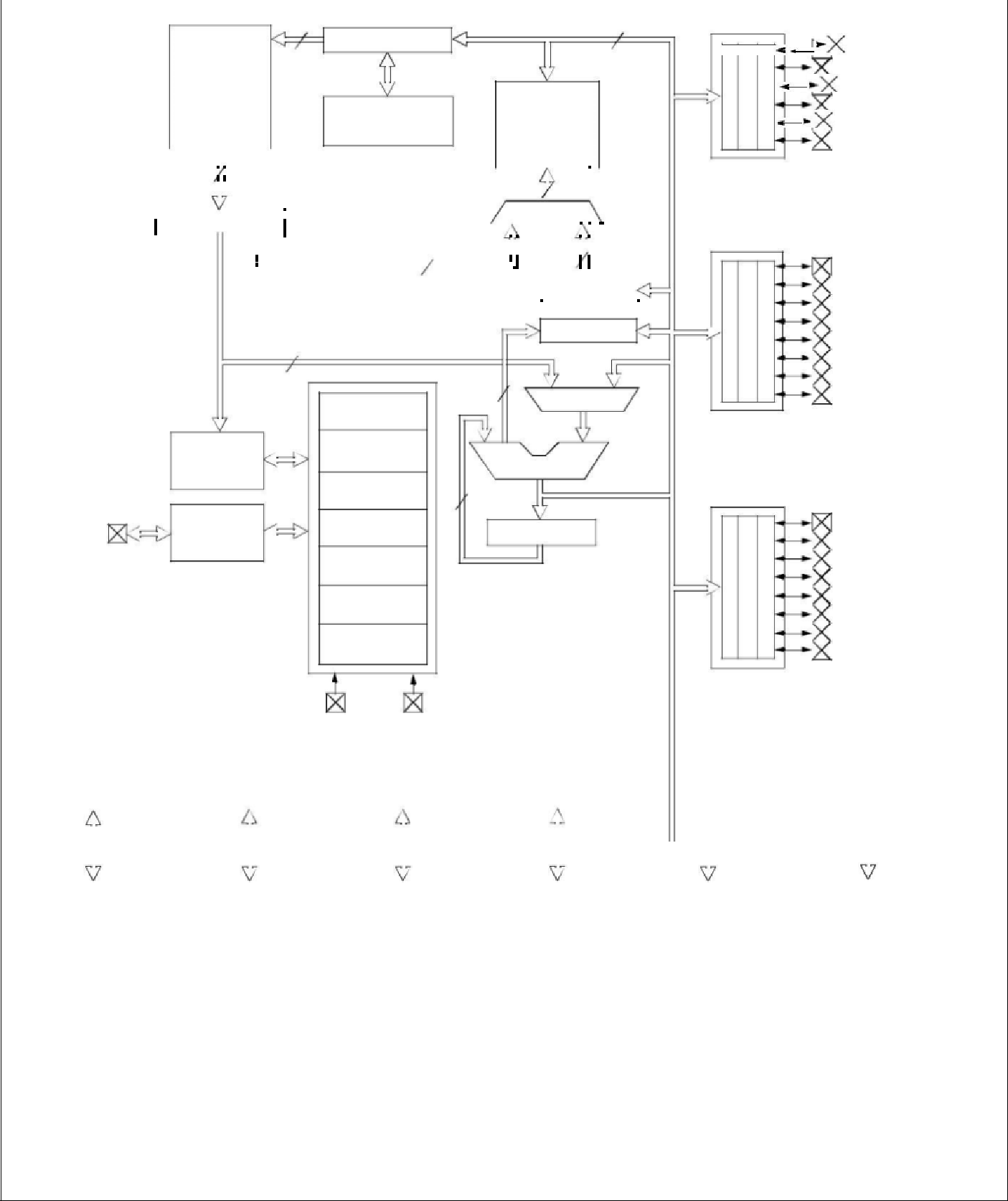
|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

61

**PIC16F87XA**



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FIGURE 1-1:** | | | | **PIC16F873A/876A BLOCK DIAGRAM** | | | | | | | | | | | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | 13 | | | | | | | | | | Program Counter | | | |  |  |  |  | Data Bus | | | | | | 8 | | | | |  |  | PORTA | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RA0/AN0 | | | |  |
|  |  |  |  |  | Flash | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RA1/AN1 | | | |  |
|  |  |  |  | Program | | | | | | | | | |  |  |  |  |  |  |  |  |  | RAM | | | |  |  |  |  |  |  |  |  | RA2/AN2/VREF-/CVREF | | | |  |
|  |  |  |  |  | Memory | | | | | | | | | 8 Level Stack | | | |  |  |  |  |  |  |  |  |  |  |  |  |  | RA3/AN3/VREF+ | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | File | | | |  |  |  |  |  |  |  |  | RA4/T0CKI/C1OUT | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | (13-bit) | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Registers | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RA5/AN4/SS/C2OUT | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | Program | | 14 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  | RAM Addr**(1)** | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | Bus | |  |  |  |  |  |  |  |  |  |  |  |  |  | 9 |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Addr MUX | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Instruction reg | | | | | | |  |  |  | 7 | | | |  |  |  |  |  |  |  |  |  | PORTB | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | Direct Addr | | | |  |  |  |  |  |  |  |  |  | 8 |  | Indirect | | | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Addr | | | |  | RB0/INT | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | FSR reg | | | | | | | |  |  | RB1 | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RB2 | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Status reg | | | | | | | | |  | RB3/PGM | | | |  |
|  |  |  |  | 8 | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RB4 | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RB5 | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RB6/PGC | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | Power-up | | | | 3 | | |  |  |  |  |  |  | MUX | | | | | | |  | RB7/PGD | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Timer | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Instruction | | | | | | | | | | Oscillator | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Start-up Timer | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Decode & | | | | | | | | | |  |  |  |  |  | ALU | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | Control | | | | | | | | | | Power-on | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Reset | | | 8 | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | PORTC | | | |  |
|  |  |  |  | Timing | | | | | | | | | | Watchdog | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC0/T1OSO/T1CKI | | | |  |
|  |  |  |  | Generation | | | | | | | | | |  | Timer | | |  |  |  |  |  | W reg | | | |  |  |  |  |  |  |  |  | RC1/T1OSI/CCP2 | | | |  |
|  | OSC1/CLKI | | |  |  |  |  |  |  |  |  |  |  | Brown-out | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC2/CCP1 | | | |  |
|  | OSC2/CLKO | | |  |  |  |  |  |  |  |  |  |  |  | Reset | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC3/SCK/SCL | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | In-Circuit | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC4/SDI/SDA | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | Debugger | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC5/SDO | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | Low-Voltage | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC6/TX/CK | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | Programming | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RC7/RX/DT | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  | MCLR | | | VDD, VSS | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | | |  | |  |  |  |  |  |  |  |  |  |  | | |  |  | | | |  |  | |  |  |  |  |  |  |
|  | Timer0 | |  |  |  |  | Timer1 | | | | |  | |  |  | Timer2 | |  |  |  |  |  | 10-bit A/D | | | | | | | |  | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data EEPROM |  | CCP1,2 |  | Synchronous |  | USART |  | Comparator |  | Voltage |  |
|  |  | Serial Port |  |  |  | Reference |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Device** | **Program Flash** | **Data Memory** | **Data EEPROM** |
|  |  |  |  |  |
|  |  |  |  |  |
|  | PIC16F873A | 4K words | 192 Bytes | 128 Bytes |
|  |  |  |  |  |
|  | PIC16F876A | 8K words | 368 Bytes | 256 Bytes |
|  |  |  |  |  |

**Note 1:** Higher order bits are from the Status register.



DS39582B 2003 Microchip Technology Inc.

62

**PIC16F87XA**



**FIGURE 1-2:** **PIC16F874A/877A BLOCK DIAGRAM**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 13 | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  | Data Bus | | | | | | | | | | 8 | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Program Counter | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Flash | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Program | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Memory | | | | | | | | | | | | | | 8 Level Stack | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RAM | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | File | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | (13-bit) | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Registers | | | | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Program | | | 14 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | RAM Addr**(1)** | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |
| Bus | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | 9 | |  |  |  |  |  |
|  |  |  | Instruction | reg | | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Direct Addr | | | 7 |  |  |  |  |  |  |  |  |  |  | 8 | | | | | | |  | Indirect | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Addr | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | FSR reg |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | | |  |  |
|  |  |  |  | 8 | | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Status reg | | | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Power-up | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | MUX | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Timer | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | |  | | | |  | | | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | |  | | | |  | | | |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Instruction | | | | | | | | | | | | | | Oscillator | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Decode & | | | | | | | | | |  |  |  | | Start-up Timer | | |  |  |  |  |  |  |  |  |  |  |  | ALU | | | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Control | | | | | |  | | | | | | |  |  | | |  |  | 8 | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | | | | | | | | Power-on | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | | | | | |  | | | | | | |  |  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Timing | | | | | |  | | | |  |  |  | | Watchdog | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Generation | | | |  | |  | | | | | | |  | Timer | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| OSC2/CLKO | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Reset | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | In-Circuit | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Debugger | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



Low-Voltage

Programming

MCLR VDD, VSS

PORTA

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  | RA0/AN0 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RA1/AN1 | | |  |
|  |  |  |  |  |  |  |  |  |  | RA2/AN2/VREF-/CVREF | | |  |
|  |  |  |  |  |  |  |  |  |  | RA3/AN3/VREF+ | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RA4/T0CKI/C1OUT | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RA5/AN4/SS/C2OUT | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB0/INT | | |  |
|  | PORTB | | | | | | | | |  |
|  | RB1 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB2 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB3/PGM | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB4 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB5 | | |  |
|  |  |  |  |  |  |  |  |  |  | RB6/PGC | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RB7/PGD | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | | |  | RC0/T1OSO/T1CKI | | |  |
|  | PORTC | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  | RC1/T1OSI/CCP2 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RC2/CCP1 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RC3/SCK/SCL | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RC4/SDI/SDA | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RC5/SDO | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RC6/TX/CK | | |  |
|  |  |  |  |  |  |  |  |  |  | RC7/RX/DT | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  | | | |  |  | | |  |  |  |  |  |
|  | PORTD | | | | | | | | | RD0/PSP0 | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RD1/PSP1 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD2/PSP2 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD3/PSP3 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD4/PSP4 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD5/PSP5 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD6/PSP6 | | |  |
|  |  |  |  |  |  |  |  |  |  | RD7/PSP7 | | |  |
| PORTE | | | | | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | RE0/RD/AN5 | | |  |
|  |  |  |  |  |  |  |  |  |  | RE1/WR/AN6 | | |  |
|  |  |  |  |  |  |  |  |  |  | RE2/CS/AN7 | | |  |



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Timer0 | | |  | Timer1 | | | |  | Timer2 | | |  |  | 10-bit A/D | | | |  |  |  |  |  | Parallel | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | Slave Port | | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | |  |  |
| Data EEPROM | | |  | CCP1,2 | | | |  | Synchronous | | |  |  | USART | | |  |  | Comparator | | | | |  |  |  | Voltage | | |  |  |
|  |  | Serial Port | | |  |  |  |  |  | |  | Reference | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  | | |  |  |  |  |  |  |  |  | | |  |  |  |  |  | | | |  |  |
|  |  |  | |  |  |  |  |  | | | |  |  |  |  |  |  | | | | |  |  |  |  |  | | | |  |  |
|  |  | **Device** | |  |  |  |  | **Program Flash** | | | |  |  |  |  |  | **Data Memory** | | | | |  |  |  |  | **Data EEPROM** | | | |  |  |
|  |  |  | |  |  |  |  |  |  | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | |  |  |
|  |  |  | |  |  |  |  |  |  | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | |  |  |
|  |  | PIC16F874A | |  |  |  |  |  | 4K words | | |  |  |  |  |  | 192 Bytes | | | | |  |  |  |  |  | 128 Bytes | | |  |  |
|  |  |  | |  |  |  |  |  |  | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | |  |  |
|  |  | PIC16F877A | |  |  |  |  |  | 8K words | | |  |  |  |  |  | 368 Bytes | | | | |  |  |  |  |  | 256 Bytes | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



**Note 1:** Higher order bits are from the Status register.



|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

63

**PIC16F87XA**



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TABLE 1-2:** | | | | | | | | | | **PIC16F873A/876A PINOUT DESCRIPTION** | | | | | | | | | | |  |
|  |  |  | **Pin Name** | | | | | | |  | **PDIP, SOIC,** | | **QFN** |  | **I/O/P** | **Buffer** | **Description** | | | |  |
|  |  |  |  | **SSOP Pin#** | | **Pin#** |  | **Type** | **Type** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | OSC1/CLKI | | | | | | |  |  |  |  | 9 | 6 |  |  | ST/CMOS | Oscillator crystal or external clock input. | | |  |  |
|  |  |  | OSC1 | | | | |  |  |  |  |  |  |  | I |  | Oscillator crystal input or external clock source input. ST | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | buffer when configured in RC mode; otherwise CMOS. | | |  |  |
|  |  |  | CLKI | | | | |  |  |  |  |  |  |  | I |  | External clock source input. Always associated with pin | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | function OSC1 (see OSC1/CLKI, OSC2/CLKO pins). | | |  |  |
|  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | OSC2/CLKO | | | | | | | | |  |  | 10 | 7 |  |  | — | Oscillator crystal or clock output. | | |  |  |
|  |  |  | OSC2 | | | | |  |  |  |  |  |  |  | O |  | Oscillator crystal output. Connects to crystal or resonator | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | in Crystal Oscillator mode. | | |  |  |
|  |  |  | CLKO | | | | |  |  |  |  |  |  |  | O |  | In RC mode, OSC2 pin outputs CLKO, which has 1/4 the | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | frequency of OSC1 and denotes the instruction cycle rate. | | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |
|  |  | |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  | | |  |
|  | MCLR/VPP | | | | | | |  |  |  |  | 1 | 26 |  |  | ST | Master Clear (input) or programming voltage (output). | | | |  |
|  |  |  | MCLR | | | |  |  |  |  |  |  |  |  | I |  | Master Clear (Reset) input. This pin is an active low Reset | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | to the device. | | | |  |
|  |  |  | VPP | | | | |  |  |  |  |  |  |  | P |  | Programming voltage input. | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | PORTA is a bidirectional I/O port. | | | |  |
|  | RA0/AN0 | | | | | | |  |  |  |  | 2 | 27 |  |  | TTL |  |  |  |  |  |
|  |  |  | RA0 | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O. | | | |  |
|  |  |  | AN0 | | | | |  |  |  |  |  |  |  | I |  | Analog input 0. | | | |  |
|  | RA1/AN1 | | | | | | |  |  |  |  | 3 | 28 |  |  | TTL |  |  |  |  |  |
|  |  |  | RA1 | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O. | | | |  |
|  |  |  | AN1 | | | | |  |  |  |  |  |  |  | I |  | Analog input 1. | | | |  |
|  | RA2/AN2/VREF-/ | | | | | | | | |  |  | 4 | 1 |  |  | TTL |  |  |  |  |  |
|  | CVREF | | | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O. | | | |  |
|  |  |  | RA2 | | | | |  |  |  |  |  |  |  | I |  | Analog input 2. | | | |  |
|  |  |  | AN2 | | | | |  |  |  |  |  |  |  | I |  | A/D reference voltage (Low) input. | | | |  |
|  |  |  | VREF- | | | | |  |  |  |  |  |  |  | O |  | Comparator VREF output. | | | |  |
|  |  |  | CVREF | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | RA3/AN3/VREF+ | | | | | | | | |  |  | 5 | 2 |  |  | TTL |  |  |  |  |  |
|  |  |  | RA3 | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O. | | | |  |
|  |  |  | AN3 | | | | |  |  |  |  |  |  |  | I |  | Analog input 3. | | | |  |
|  |  |  | VREF+ | | | | |  |  |  |  |  |  |  | I |  | A/D reference voltage (High) input. | | | |  |
|  | RA4/T0CKI/C1OUT | | | | | | | | | |  | 6 | 3 |  |  | ST |  |  |  |  |  |
|  |  |  | RA4 | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O – Open-drain when configured as output. | | | |  |
|  |  |  | T0CKI | | | | |  |  |  |  |  |  |  | I |  | Timer0 external clock input. | | | |  |
|  |  |  | C1OUT | | | | |  |  |  |  |  |  |  | O |  | Comparator 1 output. | | | |  |
|  |  | |  | |  |  | | |  | |  |  |  |  |  |  |  |  |  |  |  |
|  | RA5/AN4/SS/C2OUT | | | | | | | | | |  | 7 | 4 |  |  | TTL |  |  |  |  |  |
|  |  |  | RA5 | | | | |  |  |  |  |  |  |  | I/O |  | Digital I/O. | | | |  |
|  |  |  | AN4 | | | | |  |  |  |  |  |  |  | I |  | Analog input 4. | | | |  |
|  |  |  | SS | |  | | |  |  |  |  |  |  |  | I |  | SPI slave select input. | | | |  |
|  |  |  | C2OUT | | | | |  |  |  |  |  |  |  | O |  | Comparator 2 output. | | | |  |
|  |  | | | | | | |  | | | |  | | |  | |  |  | | |  |
|  | **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. | | | | | | | | | | | | |  |

This buffer is a Schmitt Trigger input when used in Serial Programming mode.

This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.



DS39582B 2003 Microchip Technology Inc.

64

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  | **PIC16F87XA** | |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **TABLE 1-2:** | **PIC16F873A/876A PINOUT DESCRIPTION (CONTINUED)** | | | | | | |  |  |
| **Pin Name** |  | **PDIP, SOIC,** | **QFN** | **I/O/P** | **Buffer** | **Description** | |  |  |
|  | **SSOP Pin#** | **Pin#** | **Type** | **Type** |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | PORTB is a bidirectional I/O port. PORTB can be software |  |  |  |
|  |  |  |  |  | TTL/ST**(1)** | programmed for internal weak pull-ups on all inputs. | |  |  |
| RB0/INT |  | 21 | 18 |  |  |  |  |  |
| RB0 |  |  |  | I/O |  | Digital I/O. | |  |  |
| INT |  |  |  | I |  | External interrupt. | |  |  |
| RB1 |  | 22 | 19 | I/O | TTL | Digital I/O. | |  |  |
| RB2 |  | 23 | 20 | I/O | TTL | Digital I/O. | |  |  |
| RB3/PGM |  | 24 | 21 |  | TTL |  |  |  |  |
| RB3 |  |  |  | I/O |  | Digital I/O. | |  |  |
| PGM |  |  |  | I |  | Low-voltage (single-supply) ICSP programming enable pin. | |  |  |
| RB4 |  | 25 | 22 | I/O | TTL | Digital I/O. | |  |  |
| RB5 |  | 26 | 23 | I/O | TTL | Digital I/O. | |  |  |
| RB6/PGC |  | 27 | 24 |  | TTL/ST**(2)** |  |  |  |  |
| RB6 |  |  |  | I/O |  | Digital I/O. | |  |  |
| PGC |  |  |  | I | TTL/ST**(2)** | In-circuit debugger and ICSP programming clock. | |  |  |
| RB7/PGD |  | 28 | 25 |  |  |  |  |  |
| RB7 |  |  |  | I/O |  | Digital I/O. | |  |  |
| PGD |  |  |  | I/O |  | In-circuit debugger and ICSP programming data. | |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | PORTC is a bidirectional I/O port. | |  |  |
| RC0/T1OSO/T1CKI | | 11 | 8 |  | ST |  |  |  |  |
| RC0 |  |  |  | I/O |  | Digital I/O. | |  |  |
| T1OSO |  |  |  | O |  | Timer1 oscillator output. | |  |  |
| T1CKI |  |  |  | I |  | Timer1 external clock input. | |  |  |
| RC1/T1OSI/CCP2 |  | 12 | 9 |  | ST |  |  |  |  |
| RC1 |  |  |  | I/O |  | Digital I/O. | |  |  |
| T1OSI |  |  |  | I |  | Timer1 oscillator input. | |  |  |
| CCP2 |  |  |  | I/O |  | Capture2 input, Compare2 output, PWM2 output. | |  |  |
| RC2/CCP1 |  | 13 | 10 |  | ST |  |  |  |  |
| RC2 |  |  |  | I/O |  | Digital I/O. | |  |  |
| CCP1 |  |  |  | I/O |  | Capture1 input, Compare1 output, PWM1 output. | |  |  |
| RC3/SCK/SCL |  | 14 | 11 |  | ST |  |  |  |  |
| RC3 |  |  |  | I/O |  | Digital I/O. | |  |  |
| SCK |  |  |  | I/O |  | Synchronous serial clock input/output for SPI mode. | |  |  |
| SCL |  |  |  | I/O |  | Synchronous serial clock input/output for I2C mode. | |  |  |
| RC4/SDI/SDA |  | 15 | 12 |  | ST |  |  |  |  |
| RC4 |  |  |  | I/O |  | Digital I/O. | |  |  |
| SDI |  |  |  | I |  | SPI data in. | |  |  |
| SDA |  |  |  | I/O |  | I2C data I/O. | |  |  |
| RC5/SDO |  | 16 | 13 |  | ST |  |  |  |  |
| RC5 |  |  |  | I/O |  | Digital I/O. | |  |  |
| SDO |  |  |  | O |  | SPI data out. | |  |  |
| RC6/TX/CK |  | 17 | 14 |  | ST |  |  |  |  |
| RC6 |  |  |  | I/O |  | Digital I/O. | |  |  |
| TX |  |  |  | O |  | USART asynchronous transmit. | |  |  |
| CK |  |  |  | I/O |  | USART1 synchronous clock. | |  |  |
| RC7/RX/DT |  | 18 | 15 |  | ST |  |  |  |  |
| RC7 |  |  |  | I/O |  | Digital I/O. | |  |  |
| RX |  |  |  | I |  | USART asynchronous receive. | |  |  |
| DT |  |  |  | I/O |  | USART synchronous data. | |  |  |
|  |  |  |  |  |  |  |  |  |  |
| VSS |  | 8, 19 | 5, 6 | P | — | Ground reference for logic and I/O pins. | |  |  |
|  |  |  |  |  |  |  |  |  |  |
| VDD |  | 20 | 17 | 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. | | | |
|  | This buffer is a Schmitt Trigger input when used in Serial Programming mode. | | | |
|  | This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise. | | | |
|  | | |  |  |
|  | | |  |  |
| 2003 Microchip Technology Inc. | | |  | DS39582B |



65

**PIC16F87XA**



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TABLE 1-3:** | | | | | | | | | **PIC16F874A/877A PINOUT DESCRIPTION** | | | | | | |  |  |
|  |  |  |  | **Pin Name** | | | | |  | **PDIP** | **PLCC** | **TQFP** | **QFN** | **I/O/P** | **Buffer** | **Description** |  |
|  |  |  |  |  | **Pin#** | **Pin#** | **Pin#** | **Pin#** | **Type** | **Type** |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | OSC1/CLKI | | | | | | | |  | 13 | 14 | 30 | 32 |  | ST/CMOS | Oscillator crystal or external clock input. |  |
|  |  |  | OSC1 | | | | | |  |  |  |  |  | I |  | Oscillator crystal input or external clock source |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | input. ST buffer when configured in RC mode; |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | otherwise CMOS. |  |
|  |  |  | CLKI | | | | | |  |  |  |  |  | I |  | External clock source input. Always associated |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | with pin function OSC1 (see OSC1/CLKI, |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | OSC2/CLKO pins). |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | OSC2/CLKO | | | | | | | |  | 14 | 15 | 31 | 33 |  | — | Oscillator crystal or clock output. |  |
|  |  |  | OSC2 | | | | | |  |  |  |  |  | O |  | Oscillator crystal output. |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Connects to crystal or resonator in Crystal |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | Oscillator mode. |  |
|  |  |  | CLKO | | | | | |  |  |  |  |  | O |  | In RC mode, OSC2 pin outputs CLKO, which |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | has 1/4 the frequency of OSC1 and denotes the |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | instruction cycle rate. |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | MCLR/VPP | | | | | | | |  | 1 | 2 | 18 | 18 |  | ST | Master Clear (input) or programming voltage (output). |  |
|  |  |  | MCLR | | | |  | |  |  |  |  |  | I |  | Master Clear (Reset) input. This pin is an active |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | low Reset to the device. |  |
|  |  |  | VPP | | | | | |  |  |  |  |  | P |  | Programming voltage input. |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | PORTA is a bidirectional I/O port. |  |
|  | RA0/AN0 | | | | | | | |  | 2 | 3 | 19 | 19 |  | TTL |  |  |
|  |  |  | RA0 | | | | | |  |  |  |  |  | I/O |  | Digital I/O. |  |
|  |  |  | AN0 | | | | | |  |  |  |  |  | I |  | Analog input 0. |  |
|  | RA1/AN1 | | | | | | | |  | 3 | 4 | 20 | 20 |  | TTL |  |  |
|  |  |  | RA1 | | | | | |  |  |  |  |  | I/O |  | Digital I/O. |  |
|  |  |  | AN1 | | | | | |  |  |  |  |  | I |  | Analog input 1. |  |
|  | RA2/AN2/VREF-/CVREF | | | | | | | | | 4 | 5 | 21 | 21 |  | TTL |  |  |
|  |  |  | RA2 | | | | | |  |  |  |  |  | I/O |  | Digital I/O. |  |
|  |  |  | AN2 | | | | | |  |  |  |  |  | I |  | Analog input 2. |  |
|  |  |  | VREF- | | | | | |  |  |  |  |  | I |  | A/D reference voltage (Low) input. |  |
|  |  |  | CVREF | | | | | |  |  |  |  |  | O |  | Comparator VREF output. |  |
|  | RA3/AN3/VREF+ | | | | | | | |  | 5 | 6 | 22 | 22 |  | TTL |  |  |
|  |  |  | RA3 | | | | | |  |  |  |  |  | I/O |  | Digital I/O. |  |
|  |  |  | AN3 | | | | | |  |  |  |  |  | I |  | Analog input 3. |  |
|  |  |  | VREF+ | | | | | |  |  |  |  |  | I |  | A/D reference voltage (High) input. |  |
|  | RA4/T0CKI/C1OUT | | | | | | | |  | 6 | 7 | 23 | 23 |  | ST |  |  |
|  |  |  | RA4 | | | | | |  |  |  |  |  | I/O |  | Digital I/O – Open-drain when configured as |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | output. |  |
|  |  |  | T0CKI | | | | | |  |  |  |  |  | I |  | Timer0 external clock input. |  |
|  |  |  | C1OUT | | | | | |  |  |  |  |  | O |  | Comparator 1 output. |  |
|  |  | | | |  |  | |  | |  |  |  |  |  |  |  |  |
|  | RA5/AN4/SS/C2OUT | | | | | | | | | 7 | 8 | 24 | 24 |  | TTL |  |  |
|  |  |  | RA5 | | | | | |  |  |  |  |  | I/O |  | Digital I/O. |  |
|  |  |  | AN4 | | | | | |  |  |  |  |  | I |  | Analog input 4. |  |
|  |  |  | SS | |  | | | |  |  |  |  |  | I |  | SPI slave select input. |  |
|  |  |  | C2OUT | | | | | |  |  |  |  |  | O |  | Comparator 2 output. |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**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.

This buffer is a Schmitt Trigger input when used in Serial Programming mode.

This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.



DS39582B 2003 Microchip Technology Inc.

66

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | **PIC16F87XA** | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **TABLE 1-3:** | **PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)** | | | | | | | | |  |
| **Pin Name** |  | **PDIP** | **PLCC** | **TQFP** | **QFN** | **I/O/P** | **Buffer** | **Description** | |  |
|  | **Pin#** | **Pin#** | **Pin#** | **Pin#** | **Type** | **Type** |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | PORTB is a bidirectional I/O port. PORTB can be | |  |
|  |  |  |  |  |  |  |  | software programmed for internal weak pull-up on all | |  |
|  |  |  |  |  |  |  | TTL/ST**(1)** | inputs. | |  |
| RB0/INT |  | 33 | 36 | 8 | 9 |  |  |  |  |
| RB0 |  |  |  |  |  | I/O |  | Digital I/O. | |  |
| INT |  |  |  |  |  | I |  | External interrupt. | |  |
| RB1 |  | 34 | 37 | 9 | 10 | I/O | TTL | Digital I/O. | |  |
| RB2 |  | 35 | 38 | 10 | 11 | I/O | TTL | Digital I/O. | |  |
| RB3/PGM |  | 36 | 39 | 11 | 12 |  | TTL |  |  |  |
| RB3 |  |  |  |  |  | I/O |  | Digital I/O. | |  |
| PGM |  |  |  |  |  | I |  | Low-voltage ICSP programming enable pin. | |  |
| RB4 |  | 37 | 41 | 14 | 14 | I/O | TTL | Digital I/O. | |  |
| RB5 |  | 38 | 42 | 15 | 15 | I/O | TTL | Digital I/O. | |  |
| RB6/PGC |  | 39 | 43 | 16 | 16 |  | TTL/ST**(2)** |  |  |  |
| RB6 |  |  |  |  |  | I/O |  | Digital I/O. | |  |
| PGC |  |  |  |  |  | I | TTL/ST**(2)** | In-circuit debugger and ICSP programming clock. | |  |
| RB7/PGD |  | 40 | 44 | 17 | 17 |  |  |  |  |
| RB7 |  |  |  |  |  | I/O |  | Digital I/O. | |  |
| PGD |  |  |  |  |  | I/O |  | In-circuit debugger and ICSP programming 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 the external interrupt.

This buffer is a Schmitt Trigger input when used in Serial Programming mode.

This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.



|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

67

**PIC16F87XA**



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TABLE 1-3:** | **PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)** | | | | | | | |  |
| **Pin Name** |  | **PDIP** | **PLCC** | **TQFP** | **QFN** | **I/O/P** | **Buffer** | **Description** |  |
|  | **Pin#** | **Pin#** | **Pin#** | **Pin#** | **Type** | **Type** |  |
|  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | PORTC is a bidirectional I/O port. |  |
| RC0/T1OSO/T1CKI | | 15 | 16 | 32 | 34 |  | ST |  |  |
| RC0 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| T1OSO |  |  |  |  |  | O |  | Timer1 oscillator output. |  |
| T1CKI |  |  |  |  |  | I |  | Timer1 external clock input. |  |
| RC1/T1OSI/CCP2 |  | 16 | 18 | 35 | 35 |  | ST |  |  |
| RC1 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| T1OSI |  |  |  |  |  | I |  | Timer1 oscillator input. |  |
| CCP2 |  |  |  |  |  | I/O |  | Capture2 input, Compare2 output, PWM2 output. |  |
| RC2/CCP1 |  | 17 | 19 | 36 | 36 |  | ST |  |  |
| RC2 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| CCP1 |  |  |  |  |  | I/O |  | Capture1 input, Compare1 output, PWM1 output. |  |
| RC3/SCK/SCL |  | 18 | 20 | 37 | 37 |  | ST |  |  |
| RC3 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| SCK |  |  |  |  |  | I/O |  | Synchronous serial clock input/output for SPI |  |
|  |  |  |  |  |  |  |  | mode. |  |
| SCL |  |  |  |  |  | I/O |  | Synchronous serial clock input/output for I C |  |
|  |  |  |  |  |  |  |  | mode. |  |
| RC4/SDI/SDA |  | 23 | 25 | 42 | 42 |  | ST |  |  |
| RC4 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| SDI |  |  |  |  |  | I |  | SPI data in. |  |
| SDA |  |  |  |  |  | I/O |  | I2C data I/O. |  |
| RC5/SDO |  | 24 | 26 | 43 | 43 |  | ST |  |  |
| RC5 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| SDO |  |  |  |  |  | O |  | SPI data out. |  |
| RC6/TX/CK |  | 25 | 27 | 44 | 44 |  | ST |  |  |
| RC6 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| TX |  |  |  |  |  | O |  | USART asynchronous transmit. |  |
| CK |  |  |  |  |  | I/O |  | USART1 synchronous clock. |  |
| RC7/RX/DT |  | 26 | 29 | 1 | 1 |  | ST |  |  |
| RC7 |  |  |  |  |  | I/O |  | Digital I/O. |  |
| RX |  |  |  |  |  | I |  | USART asynchronous receive. |  |
| DT |  |  |  |  |  | I/O |  | USART 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 the external interrupt.

This buffer is a Schmitt Trigger input when used in Serial Programming mode.

This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.



DS39582B 2003 Microchip Technology Inc.

68

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **PIC16F87XA** | |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |  |  | |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |  |  | |  |  |  |
| **TABLE 1-3:** | | | | | | | | **PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)** | | | | | | | | | |  |
|  |  | **Pin Name** | | | | | |  | **PDIP** | **PLCC** | **TQFP** | **QFN** | **I/O/P** | **Buffer** |  | **Description** | |  |
|  |  |  | **Pin#** | **Pin#** | **Pin#** | **Pin#** | **Type** | **Type** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | PORTD is a bidirectional I/O port or Parallel Slave | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | ST/TTL**(3)** |  | Port when interfacing to a microprocessor bus. | |  |
| RD0/PSP0 | | | | | | |  |  | 19 | 21 | 38 | 38 |  |  |  |  |  |
|  | RD0 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP0 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD1/PSP1 | | | | | | |  |  | 20 | 22 | 39 | 39 |  |  |  |  |  |
|  | RD1 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP1 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD2/PSP2 | | | | | | |  |  | 21 | 23 | 40 | 40 |  |  |  |  |  |
|  | RD2 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP2 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD3/PSP3 | | | | | | |  |  | 22 | 24 | 41 | 41 |  |  |  |  |  |
|  | RD3 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP3 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD4/PSP4 | | | | | | |  |  | 27 | 30 | 2 | 2 |  |  |  |  |  |
|  | RD4 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP4 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD5/PSP5 | | | | | | |  |  | 28 | 31 | 3 | 3 |  |  |  |  |  |
|  | RD5 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP5 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD6/PSP6 | | | | | | |  |  | 29 | 32 | 4 | 4 |  |  |  |  |  |
|  | RD6 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP6 | | | | | |  |  |  |  |  |  | I/O | ST/TTL**(3)** |  | Parallel Slave Port data. | |  |
| RD7/PSP7 | | | | | | |  |  | 30 | 33 | 5 | 5 |  |  |  |  |  |
|  | RD7 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | PSP7 | | | | | |  |  |  |  |  |  | I/O |  |  | Parallel Slave Port data. | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  | ST/TTL**(3)** |  | PORTE is a bidirectional I/O port. | |  |
|  |  |  |  |  |  | | |  |  |  | 25 |  |  |  |  |  |  |
| RE0/RD/AN5 | | | | | | | |  | 8 | 9 | 25 |  |  |  |  |  |
|  | RE0 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  |  | |  |  | | |  |  |  |  |  |  |  |  |  |  | |  |
|  | RD | | | | | |  |  |  |  |  |  | I |  |  | Read control for Parallel Slave Port. | |  |
|  | AN5 | | | | | |  |  |  |  |  |  | I | ST/TTL**(3)** |  | Analog input 5. | |  |
|  | |  | |  |  | | |  |  |  |  |  |  |  |  |  |  |
| RE1/WR/AN6 | | | | | | | |  | 9 | 10 | 26 | 26 |  |  |  |  |  |
|  | RE1 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  |  | |  |  | | |  |  |  |  |  |  |  |  |  |  | |  |
|  | WR | | | | | |  |  |  |  |  |  | I |  |  | Write control for Parallel Slave Port. | |  |
|  | AN6 | | | | | |  |  |  |  |  |  | I | ST/TTL**(3)** |  | Analog input 6. | |  |
|  | |  | |  |  | | |  |  |  |  |  |  |  |  |  |  |
| RE2/CS/AN7 | | | | | | | |  | 10 | 11 | 27 | 27 |  |  |  |  |  |
|  | RE2 | | | | | |  |  |  |  |  |  | I/O |  |  | Digital I/O. | |  |
|  | CS | | |  | | |  |  |  |  |  |  | I |  |  | Chip select control for Parallel Slave Port. | |  |
|  | AN7 | | | | | |  |  |  |  |  |  | I |  |  | Analog input 7. | |  |
|  | | | | | | |  |  |  |  |  |  |  |  |  |  | |  |
| VSS | | | | | | |  |  | 12, 31 | 13, 34 | 6, 29 | 6, 30, | P | — |  | Ground reference for logic and I/O pins. | |  |
|  |  |  |  |  |  |  |  |  |  |  |  | 31 |  |  |  |  |  |  |
|  | | | | | | |  |  |  |  |  |  |  |  |  |  | |  |
| VDD | | | | | | |  |  | 11, 32 | 12, 35 | 7, 28 | 7, 8, | P | — |  | Positive supply for logic and I/O pins. | |  |
|  |  |  |  |  |  |  |  |  |  |  |  | 28, 29 |  |  |  |  |  |  |
|  | | | | | | |  |  |  |  |  |  |  |  |  |  | |  |
| NC | | | | | | |  |  | — | 1, 17, | 12,13, | 13 | — | — |  | These pins are not internally connected. These pins | |  |
|  |  |  |  |  |  |  |  |  |  | 28, 40 | 33, 34 |  |  |  |  | should be left unconnected. | |  |
|  | | | | | | |  | |  | |  |  | | |  | | |  |
| **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. | | | | | | | | | | |  |
|  |  |  |  |  |  |  | This buffer is a Schmitt Trigger input when used in Serial Programming mode. | | | | | | | | | | |  |
|  |  |  |  |  |  |  | This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise. | | | | | | | | | | |  |



|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

69

**PIC16F87XA**



**MEMORY ORGANIZATION**

There are three memory blocks in each of the PIC16F87XA devices. The program memory and data memory have separate buses so that concurrent access can occur and is detailed in this section. The EEPROM data memory block is detailed in **Section 3.0**

**“Data EEPROM and Flash Program Memory”**.Additional information on device memory may be

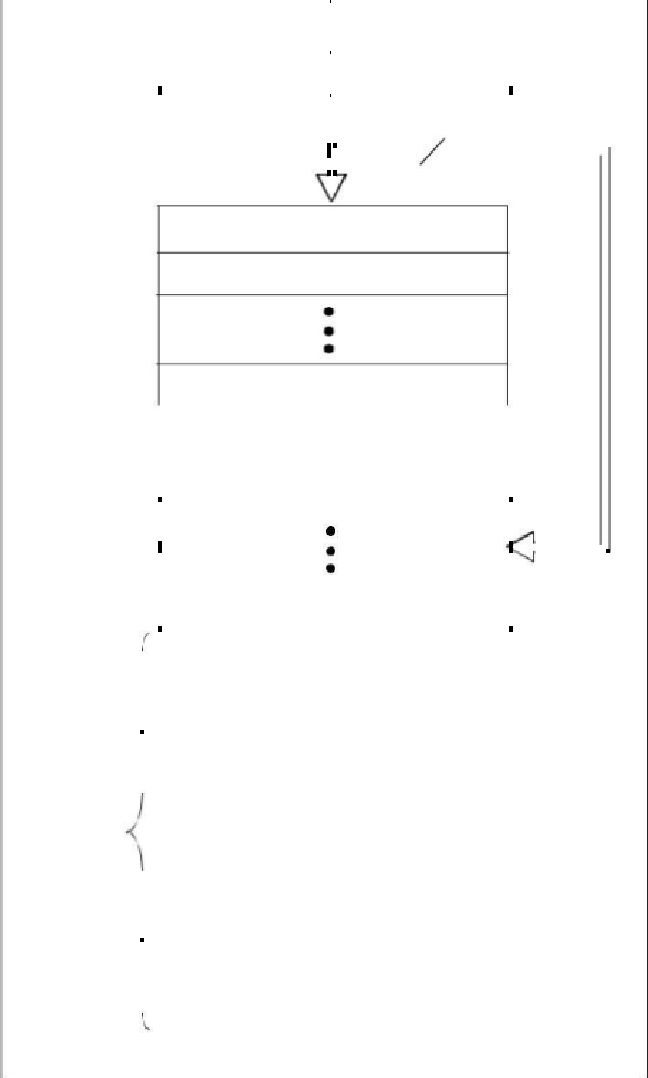
found in the PICmicro® Mid-Range MCU Family Reference Manual (DS33023).

**Program Memory Organization**

The PIC16F87XA devices have a 13-bit program counter capable of addressing an 8K word x 14 bit program memory space. The PIC16F876A/877A devices have 8K words x 14 bits of Flash program memory, while PIC16F873A/874A devices have 4K words x 14 bits. Accessing a location above the physically implemented address will cause a wraparound.

The Reset vector is at 0000h and the interrupt vector is at 0004h.

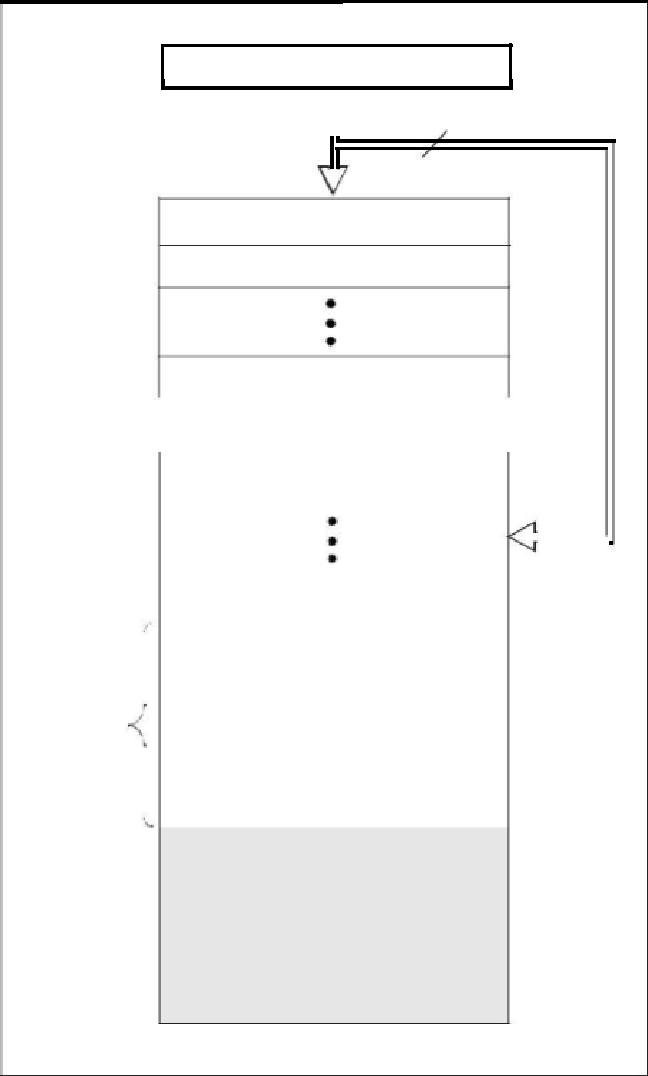
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FIGURE 2-1:** | | | **PIC16F876A/877A** | | | | | | | | | | | |  |
|  |  |  | **PROGRAM MEMORY MAP** | | | | | | | | | | | |  |
|  |  |  | **AND STACK** | | | | | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |
|  |  |  | PC<12:0> | | | | |  |  |  |  |  |  |  |  |
|  |  |  | |  | |  |  |  |  |  |  |  |  |  |  |
| CALL, RETURN | | | |  |  |  |  | 13 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| RETFIE, RETLW | | | |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | Stack Level 1 | | | | |  |  |  |  |  |  |  |  |
|  |  |  | Stack Level 2 | | | | |  |  |  |  |  |  |  |  |
|  |  |  | Stack Level 8 | | | | |  |  |  |  |  |  |  |  |
|  |  |  |  | | | | |  |  |  | | |  | |  |
|  |  |  |  | | | | |  |  |  | | |  | |  |
|  |  |  | Reset Vector | | | | |  | 0000h | | | | | |  |
|  |  |  |  | | | | |  |  |  | | |  | |  |
|  |  |  |  | | | | |  |  |  | |  |  | |  |
|  |  |  |  | | | | |  |  |  | | |  | |  |
|  |  |  |  | | | | |  |  | | | | | |  |
|  |  |  | Interrupt Vector | | | | |  | 0004h | | | | | |  |
|  |  |  |  |  |  |  |  |  | 0005h | | | | | |  |
|  |  |  | Page 0 | | | | |  |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | 07FFh | | | | | |  |
|  |  |  |  |  |  |  |  |  | 0800h | | | | | |  |
| On-Chip |  |  | Page 1 | | | | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | 0FFFh | | | | | |  |
| Program | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Memory | |  | Page 2 | | | | |  | 1000h | | | | | |  |
|  |  |  |  | 17FFh | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  | Page 3 | | | | |  | 1800h | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  | 1FFFh | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |



**FIGURE 2-2:** **PIC16F873A/874A**

**PROGRAM MEMORY MAP**

**AND STACK**



PC<12:0>

CALL, RETURN  13

RETFIE, RETLW

Stack Level 1

Stack Level 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | Stack Level 8 |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | Reset Vector | 0000h | |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | |  |  |
|  |  |  |  | Interrupt Vector | 0004h | |  |  |
|  |  |  |  |  | 0005h | |  |  |
| On-Chip | |  |  | Page 0 |  |  |
|  |  |  |  |
|  |  |  |  |  |  |
|  |  |  | 07FFh | |  |  |
| Program | | | |  |  |  |
|  |  |  |  |  |
| Memory |  | |  | Page 1 | 0800h | |  |  |
|  | |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | 0FFFh | |  |  |
|  |  |  |  |  | 1000h | |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | 1FFFh | |  |  |

|  |  |
| --- | --- |
| 2003 Microchip Technology Inc. | DS39582B |

70

**PIC16F87XA**



**FIGURE 2-3:** **PIC16F876A/877A REGISTER FILE MAP**

|  |  |  |
| --- | --- | --- |
|  | **File** |  |
| **Address** | |  |
|  |  |  |
| Indirect addr.**(\*)** | 00h |  |
| TMR0 | 01h |  |
|  |  |  |
| PCL | 02h |  |
| STATUS | 03h |  |
| FSR | 04h |  |
| PORTA | 05h |  |
| PORTB | 06h |  |
| PORTC | 07h |  |
| PORTD | 08h |  |
| PORTE | 09h |  |
| PCLATH | 0Ah |  |
| INTCON | 0Bh |  |
| PIR1 | 0Ch |  |
| PIR2 | 0Dh |  |
|  | 0Eh |  |
| TMR1L |  |
|  |  |  |
| TMR1H | 0Fh |  |
|  | 10h |  |
| T1CON |  |
|  |  |  |
| TMR2 | 11h |  |
|  |  |  |
| T2CON | 12h |  |
|  |  |  |
| SSPBUF | 13h |  |
|  |  |  |
| SSPCON | 14h |  |
|  |  |  |
| CCPR1L | 15h |  |
|  |  |  |
| CCPR1H | 16h |  |
|  |  |  |
| CCP1CON | 17h |  |
|  | 18h |  |
| RCSTA |  |
| TXREG | 19h |  |
| RCREG | 1Ah |  |
| CCPR2L | 1Bh |  |
| CCPR2H | 1Ch |  |
| CCP2CON | 1Dh |  |
| ADRESH | 1Eh |  |
| ADCON0 | 1Fh |  |
|  | 20h |  |
|  |  |
| General |  |  |
| Purpose |  |  |
| Register |  |  |
| 96 Bytes |  |  |
|  | 7Fh |  |
| Bank 0 |  |
|  |  |



|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **File** |  |
|  | **Address** | |  |
|  |  |  |  |
|  | Indirect addr.**(\*)** | 80h |  |
|  | OPTION\_REG | 81h |  |
|  | PCL | 82h |  |
|  | STATUS | 83h |  |
|  | FSR | 84h |  |
|  | TRISA | 85h |  |
|  | TRISB | 86h |  |
|  | TRISC | 87h |  |
|  | TRISD | 88h |  |
|  | TRISE | 89h |  |
|  | PCLATH | 8Ah |  |
|  | INTCON | 8Bh |  |
|  | PIE1 | 8Ch |  |
|  | PIE2 | 8Dh |  |
|  | PCON | 8Eh |  |
|  |  | 8Fh |  |
|  |  | 90h |  |
|  | SSPCON2 | 91h |  |
|  | PR2 | 92h |  |
|  | SSPADD | 93h |  |
|  | SSPSTAT | 94h |  |
|  |  | 95h |  |
|  |  | 96h |  |
|  |  | 97h |  |
|  | TXSTA | 98h |  |
|  | SPBRG | 99h |  |
|  |  | 9Ah |  |
|  |  | 9Bh |  |
|  | CMCON | 9Ch |  |
|  | CVRCON | 9Dh |  |
|  | ADRESL | 9Eh |  |
|  | ADCON1 | 9Fh |  |
|  |  | A0h |  |
|  | General |  |  |
|  | Purpose |  |  |
|  | Register |  |  |
|  | 80 Bytes |  |  |
|  |  | EFh |  |
|  | accesses | F0h |  |
|  |  |  |
|  | 70h-7Fh |  |  |
|  |  | FFh |  |
|  | Bank 1 |  |
|  |  |  |

**File**

**Address**

Indirect addr.**(\*)** 100h

TMR0 101h

PCL 102h STATUS 103h

FSR 104h 105h

PORTB 106h

107h

108h

109h

PCLATH 10Ah

INTCON 10Bh

EEDATA 10Ch

EEADR 10Dh

EEDATH 10Eh

EEADRH 10Fh

110h

111h

112h

113h

114h

115h

116h

General 117h

Purpose

Register 118h

16 Bytes 119h

11Ah

11Bh

11Ch

11Dh

11Eh

11Fh

120h

General

Purpose

Register

80 Bytes

16Fh

accesses  170h

70h-7Fh

17Fh

Bank 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **File** |  |
|  |  | **Address** | | |  |
|  |  |  |  |  |  |
|  | Indirect addr.**(\*)** |  |  | 180h |  |
|  | OPTION\_REG |  |  | 181h |  |
|  | PCL |  |  | 182h |  |
|  | STATUS |  |  | 183h |  |
|  | FSR |  |  | 184h |  |
|  |  |  |  | 185h |  |
|  | TRISB |  |  | 186h |  |
|  |  |  |  | 187h |  |
|  |  |  |  | 188h |  |
|  |  |  |  | 189h |  |
|  | PCLATH |  |  | 18Ah |  |
|  | INTCON |  |  | 18Bh |  |
|  | EECON1 |  |  | 18Ch |  |
|  | EECON2 |  |  | 18Dh |  |
|  | Reserved |  |  | 18Eh |  |
|  | Reserved |  |  | 18Fh |  |
|  |  |  |  | 190h |  |
|  |  |  |  | 191h |  |
|  |  |  |  | 192h |  |
|  |  |  |  | 193h |  |
|  |  |  |  | 194h |  |
|  |  |  |  | 195h |  |
|  | General |  |  | 196h |  |
|  |  |  | 197h |  |
|  | Purpose |  |  |  |
|  |  |  |  |  |
|  | Register |  |  | 198h |  |
|  | 16 Bytes |  |  | 199h |  |
|  |  |  |  | 19Ah |  |
|  |  |  |  | 19Bh |  |
|  |  |  |  | 19Ch |  |
|  |  |  |  | 19Dh |  |
|  |  |  |  | 19Eh |  |
|  |  |  |  | 19Fh |  |
|  |  |  |  | 1A0h |  |
|  |  |  |  |  |
|  | General |  |  |  |  |
|  | Purpose |  |  |  |  |
|  | Register |  |  |  |  |
|  | 80 Bytes |  |  |  |  |
|  |  |  |  | 1EFh |  |
|  | accesses |  |  | 1F0h |  |
|  |  |  |  |  |
|  | 70h - 7Fh |  |  |  |  |
|  |  |  |  | 1FFh |  |
|  | Bank 3 |  |  |  |
|  |  |  |  |  |

Unimplemented data memory locations, read as ‘0’.

Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F876A.

These registers are reserved; maintain these registers clear.



2003 Microchip Technology Inc. 71 DS39582B

**PIC16F87XA**



**FIGURE 2-4:** **PIC16F873A/874A REGISTER FILE MAP**

|  |  |  |
| --- | --- | --- |
|  | **File** |  |
| **Address** | |  |
|  |  |  |
| Indirect addr.**(\*)** | 00h |  |
| TMR0 | 01h |  |
| PCL | 02h |  |
| STATUS | 03h |  |
| FSR | 04h |  |
| PORTA | 05h |  |
| PORTB | 06h |  |
| PORTC | 07h |  |
| PORTD | 08h |  |
| PORTE | 09h |  |
| PCLATH | 0Ah |  |
| INTCON | 0Bh |  |
| PIR1 | 0Ch |  |
| PIR2 | 0Dh |  |
|  | 0Eh |  |
| TMR1L |  |
|  |  |  |
| TMR1H | 0Fh |  |
|  |  |  |
| T1CON | 10h |  |
|  |  |  |
| TMR2 | 11h |  |
|  |  |  |
| T2CON | 12h |  |
|  |  |  |
| SSPBUF | 13h |  |
|  |  |  |
| SSPCON | 14h |  |
|  |  |  |
| CCPR1L | 15h |  |
|  |  |  |
| CCPR1H | 16h |  |
|  |  |  |
| CCP1CON | 17h |  |
|  | 18h |  |
| RCSTA |  |
| TXREG | 19h |  |
| RCREG | 1Ah |  |
| CCPR2L | 1Bh |  |
| CCPR2H | 1Ch |  |
| CCP2CON | 1Dh |  |
| ADRESH | 1Eh |  |
| ADCON0 | 1Fh |  |
|  | 20h |  |
|  |  |
| General |  |  |
| Purpose |  |  |
| Register |  |  |
| 96 Bytes |  |  |
|  | 7Fh |  |
| Bank 0 |  |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **File** |  |
|  | **Address** | |  |
|  |  |  |  |
|  | Indirect addr.**(\*)** | 80h |  |
|  | OPTION\_REG | 81h |  |
|  | PCL | 82h |  |
|  | STATUS | 83h |  |
|  | FSR | 84h |  |
|  | TRISA | 85h |  |
|  | TRISB | 86h |  |
|  | TRISC | 87h |  |
|  | TRISD | 88h |  |
|  | TRISE | 89h |  |
|  | PCLATH | 8Ah |  |
|  | INTCON | 8Bh |  |
|  | PIE1 | 8Ch |  |
|  | PIE2 | 8Dh |  |
|  | PCON | 8Eh |  |
|  |  | 8Fh |  |
|  |  | 90h |  |
|  | SSPCON2 | 91h |  |
|  | PR2 | 92h |  |
|  | SSPADD | 93h |  |
|  | SSPSTAT | 94h |  |
|  |  | 95h |  |
|  |  | 96h |  |
|  |  | 97h |  |
|  | TXSTA | 98h |  |
|  | SPBRG | 99h |  |
|  |  | 9Ah |  |
|  |  | 9Bh |  |
|  | CMCON | 9Ch |  |
|  | CVRCON | 9Dh |  |
|  | ADRESL | 9Eh |  |
|  | ADCON1 | 9Fh |  |
|  |  | A0h |  |
|  |  |  |
|  | General |  |  |
|  | Purpose |  |  |
|  | Register |  |  |
|  | 96 Bytes |  |  |
|  |  | FFh |  |
|  | Bank 1 |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  | **File** |  |
|  | **Address** |  |
|  |  |  |
| Indirect addr.**(\*)** | 100h |  |
| TMR0 | 101h |  |
| PCL | 102h |  |
|  | 103h |  |
| STATUS |  |
| FSR | 104h |  |
|  | 105h |  |
| PORTB | 106h |  |
|  | 107h |  |
|  | 108h |  |
|  | 109h |  |
| PCLATH | 10Ah |  |
| INTCON | 10Bh |  |
|  | 10Ch |  |
| EEDATA |  |
| EEADR | 10Dh |  |
|  | 10Eh |  |
| EEDATH |  |
| EEADRH | 10Fh |  |
|  | 110h |  |
|  |  |

120h

accesses

20h-7Fh

16Fh

170h

17Fh

Bank 2

**File**

**Address**

Indirect addr.**(\*)** 180h OPTION\_REG 181h

PCL 182h

STATUS 183h

FSR 184h  185h

TRISB 186h  187h  188h  189h PCLATH 18AhINTCON 18Bh EECON1 18Ch EECON2 18Dh

Reserved 18Eh

Reserved 18Fh

190h

1A0h

accesses

A0h - FFh

1EFh

1F0h

1FFh

Bank 3

Unimplemented data memory locations, read as ‘0’.



Not a physical register.

**Note 1:** These registers are not implemented on the PIC16F873A.

These registers are reserved; maintain these registers clear.



DS39582B 72 2003 Microchip Technology Inc.

**PIC16F87XA**



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 features 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:** | | **Details** | |  |
|  |  |  |  |  | **POR, BOR** | | **on page:** | |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Bank 0** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 00h**(3** |  | INDF |  | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | | | | | | | | | | | | | | |  |  |  |  |  |
|  |  | 0000 0000 | | 31, 150 |  |  |
| 01h |  | TMR0 |  | Timer0 Module Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 55, 150 |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 02h**(3)** |  | PCL |  | Program Counter (PC) Least Significant Byte | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | 0000 | 0000 | 30, 150 |  |  |
| 03h |  | STATUS |  | IRP | | RP1 |  |  | RP0 | |  |  | TO | |  | PD | |  |  | Z | | DC | | C | | 0001 | 1xxx | 22, 150 |  |  |
| 04h**(3)** |  | FSR |  | Indirect Data Memory Address Pointer | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 31, 150 |  |  |
| 05h |  | PORTA |  | — |  | — |  |  | PORTA Data Latch when written: PORTA pins when read | | | | | | | | | | | | | | |  |  | --0x 0000 | | 43, 150 |  |  |
| 06h |  | PORTB |  | PORTB Data Latch when written: PORTB pins when read | | | | | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | 45, 150 |  |  |
|  |  |  |  |  | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 07h |  | PORTC |  | PORTC Data Latch when written: PORTC pins when read | | | | | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | 47, 150 |  |  |
|  |  |  |  |  | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 08h**(4)** |  | PORTD |  | PORTD Data Latch when written: PORTD pins when read | | | | | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | 48, 150 |  |  |
| 09h**(4** |  | PORTE |  | — |  | — |  |  | — |  |  |  | — | |  | — | |  | RE2 | | | RE1 | | RE0 | | ---- | -xxx | 49, 150 |  |  |
| 0Ah**(1,3)** |  | PCLATH |  | — |  | — |  |  | — |  | Write Buffer for the upper 5 bits of the Program Counter | | | | | | | | | | | | | | | ---0 0000 | | 30, 150 |  |  |
| 0Bh**(3)** |  | INTCON |  | GIE | | PEIE |  |  | TMR0IE | | INTE | | | | RBIE | | |  | TMR0IF | | | INTF | | RBIF | | 0000 | 000x | 24, 150 |  |  |
| 0Ch |  | PIR1 |  | PSPIF**(3** | | ADIF |  |  | RCIF | | TXIF | | | | SSPIF | | |  | CCP1IF | | | TMR2IF | | TMR1IF | | 0000 0000 | | 26, 150 |  |  |
| 0Dh |  | PIR2 |  | — |  | CMIF |  |  | — |  | EEIF | | | | BCLIF | | |  |  | — | | — |  | CCP2IF | | -0-0 0--0 | | 28, 150 |  |  |
| 0Eh |  | TMR1L |  | Holding Register for the Least Significant Byte of the 16-bit TMR1 Register | | | | | | | | | | | | | | | | | | | |  |  | xxxx xxxx | | 60, 150 |  |  |
|  |  |  |  |  | | | | | | | | | |  | | |  | | | |  | | |  |  |  |  |  |  |  |
| 0Fh |  | TMR1H |  | Holding Register for the Most Significant Byte of the 16-bit TMR1 Register | | | | | | | | | | | | | | | | | | | |  |  | xxxx xxxx | | 60, 150 |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |
| 10h |  | T1CON |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | T1SYN | |  |  |  |  |  |  |  |  |  |  |
|  |  | — |  | — |  |  | T1CKPS1 | | T1CKPS0 | | | | T1OSCEN | | |  |  | C | | TMR1CS | | TMR1ON | | --00 0000 | | 57, 150 |  |  |
| 11h |  | TMR2 |  | Timer2 Module Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0000 0000 | | 62, 150 |  |  |
|  |  |  |  |  | |  | | | | |  | | |  |  | |  |  | | |  |  | | |  |  |  |  |  |  |
| 12h |  | T2CON |  | — |  | TOUTPS3 | | | TOUTPS2 | | TOUTPS1 | | | | TOUTPS0 | | | TMR2ON | | | | T2CKPS1 | | T2CKPS0 | | -000 0000 | | 61, 150 |  |  |
| 13h |  | SSPBUF |  | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | 79, 150 |  |  |
|  |  |  |  |  | |  |  |  |  | |  | | |  |  | |  |  |  | |  |  | |  |  |  |  |  |  |  |
| 14h |  | SSPCON |  | WCOL | | SSPOV |  |  | SSPEN | | CKP | | | | SSPM3 | | |  | SSPM2 | | | SSPM1 | | SSPM0 | | 0000 0000 | | 82, 82, |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 150 |  |  |
|  |  |  |  |  | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 15h |  | CCPR1L |  | Capture/Compare/PWM Register 1 (LSB) | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 63, 150 |  |  |
| 16h |  | CCPR1H |  | Capture/Compare/PWM Register 1 (MSB) | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 63, 150 |  |  |
|  |  |  |  |  | |  |  |  |  | |  | | |  |  | |  |  | | |  |  | |  |  |  |  |  |  |  |
| 17h |  | CCP1CON |  | — |  | — |  |  | CCP1X | | CCP1Y | | | | CCP1M3 | | | CCP1M2 | | | | CCP1M1 | | CCP1M0 | | --00 0000 | | 64, 150 |  |  |
| 18h |  | RCSTA |  | SPEN | | RX9 |  |  | SREN | | CREN | | | | ADDEN | | |  | FERR | | | OERR | | RX9D | | 0000 000x | | 112, 150 |  |  |
|  |  |  |  |  | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| 19h |  | TXREG |  | USART Transmit Data Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0000 0000 | | 118, 150 |  |  |
|  |  |  |  |  | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1Ah |  | RCREG |  | USART Receive Data Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0000 0000 | | 118, 150 |  |  |
|  |  |  |  |  | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1Bh |  | CCPR2L |  | Capture/Compare/PWM Register 2 (LSB) | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 63, 150 |  |  |
|  |  |  |  |  | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1Ch |  | CCPR2H |  | Capture/Compare/PWM Register 2 (MSB) | | | | | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 63, 150 |  |  |
|  |  |  |  |  | |  |  |  |  | |  | | |  |  | |  |  | | |  |  | |  |  |  |  |  |  |  |
| 1Dh |  | CCP2CON |  | — |  | — |  |  | CCP2X | | CCP2Y | | | | CCP2M3 | | | CCP2M2 | | | | CCP2M1 | | CCP2M0 | | --00 0000 | | 64, 150 |  |  |
| 1Eh |  | ADRESH |  | A/D Result Register High Byte | | | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | 133, 150 |  |  |
|  |  |  |  |  | |  |  |  |  | |  | | |  |  | |  |  | | |  |  | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1Fh |  | ADCON0 |  | ADCS1 | | ADCS0 |  |  | CHS2 | | CHS1 | | | | CHS0 | | | GO/DONE | | | | — |  | ADON | | 0000 00-0 | | 127, 150 |  |  |
|  |  |  |  | |  |  |  |  |  |  |  |  |  |  | |  |  | | |  | |  |  |  |  |  |  |  |  |  |
| **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>, whosecontents are transferred to the upper byte of the program counter.

Bits PSPIE and PSPIF are reserved on PIC16F873A/876A devices; always maintain these bits clear.

These registers can be addressed from any bank.

PORTD, PORTE, TRISD and TRISE are not implemented on PIC16F873A/876A devices, read as ‘0’.

Bit 4 of EEADRH implemented only on the PIC16F876A/877A devices.



2003 Microchip Technology Inc. 73 DS39582B

**PIC16F87XA**



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | | | **Details** | |  |  |
|  |  |  |  | **POR, BOR** | | | **on page:** | |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Bank 2** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 100h**(3** | INDF |  | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | | | | | | | | | | | 0000 0000 | |  | 31, 150 |  |  |  |
| 101h | TMR0 |  | Timer0 Module Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | | 55, 150 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 102h**(3)** | PCL |  | Program Counter’s (PC) Least Significant Byte | | | | | | | | | |  |  |  |  |  |  |  |  | 0000 | 0000 | | 30, 150 |  |  |  |
| 103h | STATUS |  |  |  | IRP | RP1 | | RP0 | |  | TO | |  | PD | Z | | DC | |  | C | 0001 | 1xxx | | 22, 150 |  |  |  |
| 104h | FSR |  | Indirect Data Memory Address Pointer | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | | 31, 150 |  |  |  |
| 105h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 106h | PORTB |  | PORTB Data Latch when written: PORTB pins when read | | | | | | | | | | | |  |  |  |  |  |  | xxxx xxxx | | | 45, 150 |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |
| 107h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 108h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 109h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 10Ah**(1,3)** | PCLATH |  |  |  | — | — |  | — |  | Write Buffer for the upper 5 bits of the Program Counter | | | | | | | | | | | ---0 0000 | |  | 30, 150 |  |  |  |
| 10Bh**(3** | INTCON |  |  |  | GIE | PEIE | | TMR0IE | | INTE | | | RBIE | | TMR0IF | | INTF | |  | RBIF | 0000 | 000x | | 24, 150 |  |  |  |
| 10Ch | EEDATA |  | EEPROM Data Register Low Byte | | | | | | |  |  |  |  |  |  |  |  |  |  |  | xxxx xxxx | | | 39, 151 |  |  |  |
|  |  |  |  | |  | | |  | | | |  |  |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 10Dh | EEADR |  | EEPROM Address Register Low Byte | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | | 39, 151 |  |  |  |
|  |  |  |  |  |  |  | |  | | | |  | | |  |  |  |  |  |  |  |  | |  |  |  |  |
| 10Eh | EEDATH |  |  |  | — | — |  | EEPROM Data Register High Byte | | | | | | |  |  |  |  |  |  | --xx xxxx | | | 39, 151 |  |  |  |
| 10Fh | EEADRH |  |  |  | — | — |  | — |  | —**(5)** | |  | EEPROM Address Register High Byte | | | | | | |  | ---- | xxxx | | 39, 151 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Bank 3** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | |  | | |  | | | | | | | | | | | | |  |  |  |  |  |  |  |
| 180h**(3)** | INDF |  | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | | | | | | | | | | | 0000 0000 | |  | 31, 150 |  |  |  |
| 181h | OPTION\_REG | |  |  | RBPU | INTEDG | | T0CS | | T0SE | | | PSA | | PS2 | | PS1 | |  | PS0 | 1111 1111 | |  | 23, 150 |  |  |  |
|  |  |  |  | |  | | |  | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 182h**(3)** | PCL |  | Program Counter (PC) Least Significant Byte | | | | | | | | | |  |  |  |  |  |  |  |  | 0000 0000 | |  | 30, 150 |  |  |  |
| 183h | STATUS |  |  |  | IRP | RP1 | | RP0 | |  | TO | |  | PD | Z | | DC | |  | C | 0001 | 1xxx | | 22, 150 |  |  |  |
| 184h**(3** | FSR |  | Indirect Data Memory Address Pointer | | | | | | | | | |  |  |  |  |  |  |  |  | xxxx xxxx | | | 31, 150 |  |  |  |
| 185h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 186h | TRISB |  | PORTB Data Direction Register | | | | | | |  |  |  |  |  |  |  |  |  |  |  | 1111 1111 | |  | 45, 150 |  |  |  |
|  |  |  |  | |  | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  | |  |  |
| 187h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 188h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 189h | — |  | Unimplemented | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  | — | |  | — |  |  |  |
| 18Ah**(1,3)** | PCLATH |  |  |  | — | — |  | — |  | Write Buffer for the upper 5 bits of the Program Counter | | | | | | | | | | | ---0 0000 | |  | 30, 150 |  |  |  |
| 18Bh**(3** | INTCON |  |  |  | GIE | PEIE | | TMR0IE | | INTE | | | RBIE | | TMR0IF | | INTF | |  | RBIF | 0000 000x | | | 24, 150 |  |  |  |
| 18Ch | EECON1 |  | EEPGD | | | — |  | — |  |  | — | | WRERR | | WREN | | WR | |  | RD | x--- | x000 | | 34, 151 |  |  |  |
| 18Dh | EECON2 |  | EEPROM Control Register 2 (not a physical register) | | | | | | | | | | | |  |  |  |  |  |  | ---- | ---- |  | 39, 151 |  |  |  |
|  |  |  |  | |  | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |  |  |
| 18Eh | — |  | Reserved; maintain clear | | | | | |  |  |  |  |  |  |  |  |  |  |  |  | 0000 0000 | |  | — |  |  |  |
| 18Fh | — |  | Reserved; maintain clear | | | | | |  |  |  |  |  |  |  |  |  |  |  |  | 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>, whosecontents are transferred to the upper byte of the program counter.

Bits PSPIE and PSPIF are reserved on PIC16F873A/876A devices; always maintain these bits clear.

These registers can be addressed from any bank.

PORTD, PORTE, TRISD and TRISE are not implemented on PIC16F873A/876A devices, read as ‘0’.

Bit 4 of EEADRH implemented only on the PIC16F876A/877A devices.



2003 Microchip Technology Inc 74 DS39582B

**PIC16F87XA**



Status Register

The Status register contains the arithmetic status of the ALU, the Reset status and the bank select bits for data memory.

The Status register can be the destination for any instruction, as with any other register. If the Status reg-ister is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is dis-abled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable, therefore, the result of an instruction with the Status register as destination may be different than intended.

For example, CLRF STATUS, will clear the upper three bits and set the Z bit. This leaves the Status register as 000u u1uu (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter theStatus register because these instructions do not affect the Z, C or DC bits from the Status register. For other instructions not affecting any status bits, see **Section 15.0 “Instruction Set Summary”**.

**Note:** The C and DC bits operate as a borrow and digit borrow bit, respectively, in sub-traction. See the SUBLW and SUBWF instructions for examples.

**REGISTER 2-1:** **STATUS REGISTER (ADDRESS 03h, 83h, 103h, 183h)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| R/W-0 | R/W-0 | | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
|  |  |  |  |  |  |  |  |  |
| IRP | RP1 |  | RP0 | TO | PD | Z | DC | C |
|  |  |  |  |  |  |  |  |  |
| bit 7 |  |  |  |  |  |  |  | bit 0 |

bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)

1 = Bank 2, 3 (100h-1FFh)

0 = Bank 0, 1 (00h-FFh)

bit 6-5 **RP1:RP0**: Register Bank Select bits (used for direct addressing)

1. = Bank 3 (180h-1FFh)
2. = Bank 2 (100h-17Fh)
3. = Bank 1 (80h-FFh)
4. = Bank 0 (00h-7Fh) Each bank is 128 bytes.

bit 4 **TO**: Time-out bit

1 = After power-up, CLRWDT instruction or SLEEP instruction

0 = A WDT time-out occurred

bit 3 **PD**: Power-down bit

1 = After power-up or by the CLRWDT instruction

0 = By execution of the SLEEP instruction

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| bit 2 | **Z**: Zero bit | | | | | | | |  |  |  |  |
|  | 1 = The result of an arithmetic or logic operation is zero | | | | | | | | | |  |  |
|  | 0 = The result of an arithmetic or logic operation is not zero | | | | | | | | | |  |  |
|  |  | | | |  | | |  | bit (ADDWF, ADDLW,SUBLW,SUBWF instructions) | |  |  |
| bit 1 | **DC**: Digit carry/borrow | | | | | | | |  |  |
|  |  |  |  |  |  |  |  |  | |  |  |  |
|  | (for borrow, the polarity is reversed) | | | | | | | | |  |  |  |
|  | 1 = A carry-out from the 4th low order bit of the result occurred 0 | | | | | | | | | |  |  |
|  | = No carry-out from the 4th low order bit of the result | | | | | | | | | |  |  |
| bit 0 |  | |  | |  |  |  |  | | |  |  |
| **C**: Carry/borrow | | | | | | bit (ADDWF, ADDLW,SUBLW,SUBWF instructions) | | | |  |  |
|  | 1 = A carry-out from the Most Significant bit of the result occurred 0 | | | | | | | | | |  |  |
|  | = No carry-out from the Most Significant bit of the result occurred | | | | | | | | | |  |  |
|  |  | | | |  | | |  | | | |  |
|  | **Note:** For | | | | | borrow, | | | the polarity is reversed. A subtraction is executed by adding the two’s | | |  |
|  |  |  |  | complement of the second operand. For rotate (RRF, RLF) instructions, this bit is | | | | | | | |  |
|  |  |  |  | loaded with either the high, or low order bit of the source register. | | | | | | | |  |
|  |  | | | | | | | |  |  |  |  |
|  | **Legend:** | | | | | | | |  |  |  |  |
|  | R = Readable bit | | | | | | | | W = Writable bit | U = Unimplemented bit, read as ‘0’ | |  |
|  | - n = Value at POR | | | | | | | | ‘1’ = Bit is set | ‘0’ = Bit is cleared | x = Bit is unknown |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |



DS39582B 75 2003 Microchip Technology Inc.