MONITORING AND CONTROLLING OF CONVEYOR BELT USING ZIGBEE

By

V.ARTHI — **Reg No: 0710107008**

P.DHIVYA PRABHA — **Reg No: 0710107023**

G.JEEVITHA — **Reg No: 0710107038**

of

KUMARAGURU COLLEGE OF TECHNOLOGY,
COIMBATORE - 641 049.
(An Autonomous Institution affiliated to Anna University of Technology, Coimbatore)

A PROJECT REPORT

Submitted to the

FACULTY OF ELECTRONICS AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements
for the award of the degree

of

BACHELOR OF ENGINEERING
in
ELECTRONICS AND COMMUNICATION ENGINEERING

APRIL 2011

BONAFIDE CERTIFICATE

Certified that this project report titled **"Monitoring And Controlling Of Conveyor Belt Using Zigbee"** is the bonafide work of **Ms.V.Arthi** [Reg. No. 0710107008], **Ms.P.Dhivya Prabha**[Reg.No. 0710107023] and **Ms.G.Jeevitha** [Reg. No. 0710107038] who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Project Guide

Mrs.S.Thilagam

Head of the Department

Dr. Rajeswari Mariappan

The candidate with Register No. 0710107008, 0710107023, 0710107038 is examined by us in the project viva-voce examination held on 18.4.2011

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First I would like to express my praise and gratitude to the Lord, who has showered his grace and blessing enabling me to complete this project in an excellent manner.

I express my sincere thanks to our beloved Director **Dr.J.Shanmugam**, Kumaraguru College of Technology, for his kind support and for providing necessary facilities to carry out the work.

I express my sincere thanks to our beloved Principal **Dr.S.Ramachandran**, Kumaraguru College of Technology, who encouraged me in each and every steps of the project work.

I would like to express my deep sense of gratitude to our HOD, the ever active **Dr.Rajeswari Mariappan**, Department of Electronics and Communication Engineering, for her valuable suggestions and encouragement which paved way for the successful completion of the project work.

In particular, I wish to thank with everlasting gratitude to the Project Coordinator **Mrs.V.Jeyasri Arokiamary,** Associate Professor, Department of Electronics and Communication Engineering, for her expert counseling and guidance to make this project to a great deal of success.

I am greatly privileged to express my heartfelt thanks to my project guide **Mrs.S.Thilagam,** Assitant Professor, Department of Electronics and Communication Engineering, throughout the course of this project work and I wish to convey my deep sense of gratitude to all the teaching and non-teaching staffs of ECE Department for their help and cooperation.

Finally, I thank my parents and my family members for giving me the moral support and abundant blessings in all of my activities and my dear friends who helped me to endure my difficult times with their unfailing support and warm wishes.

ABSTRACT

ABSTRACT:

'**MONITORING AND CONTROLLING OF CONVEYOR BELT USING ZIGBEE**' simplifies the access of controlling of conveyor belts at factory outlet for loading/unloading purpose. In factory outlet point labours are involved to load and unload the goods to/from the factory. While going for the labours, it is a time consuming and our project is to minimize the time consumption and employee cost by wireless control from a PC using low cost wireless sensor networks-Zigbee protocol.

The output of our project is based on belt conveyor. Conveyors are used as components in automated distribution and warehousing. In combination with computer controlled pallet handling equipment this allows for more efficient retail, wholesale, and manufacturing distribution. It is considered a labour saving system that allows large volumes to move rapidly through a process, allowing companies to ship or receive higher volumes with smaller storage space and with less labour expense. Material flowing over the belt may be weighed in transit using a belt weighter. Belts with regularly spaced partitions, known as elevator belts, are used for transporting loose materials up steep inclines Belts with regularly spaced partitions are used for transporting loose materials –can be controlled for ON /OFF, direction and speed.

CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

PIC	PERIPHERAL INTERFACE CONTROLLER
RS 232	RECOMMENDED STANDARD 232
PC	PERSONAL COMPUTER
CPU	CENTRAL PROCESSING UNIT
LCD	LIQUID CRYSTAL DISPLAY
UART	UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITTER
USB	UNIVERSAL SERIAL BUS
TTL	TRANSISTER TRANSISTER LOGIC

CHAPTER 1

INTRODUCTION

1.1 BRIEF OVERVIEW:

Our system has two sections – a transceiver (control section), a transceiver (conveyor section). The control section is connected to a PC, which receive and monitor & control the different parameters on the conveyor section.

The control section receives data from the conveyor section and displays it on the VB counter via Zigbee transceiver. The monitering station measures the variation in the temperature of the conveyor section. The conveyor belt can be functioned from wireless station which includes seven different functions such as start, slow, medium, fast, forward, reverse and stop.

1.2 EMBEDDED SYSTEMS

The embedded system is a dedicated system which is a combination of hardware and software. As the size of the processor goes smaller and cheaper more products have the processor embedded in the system. It is basically has a CPU, memory and control register. The memory is the storage device. But this processor cannot function independently because of the simple reason they don't have enough memory and they don't have direct control over the external device. The external devices that provide support to the processor are DMA channel, ROM, RAM etc to make the processor function. This complete set appears to be very big though they can perform vast tasks.

BLOCK DESCRIPTION

CHAPTER 2

BLOCK DESCRIPTION

2.1 BLOCK DIAGRAM:

2.1.1 CONTROLLER BLOCK:

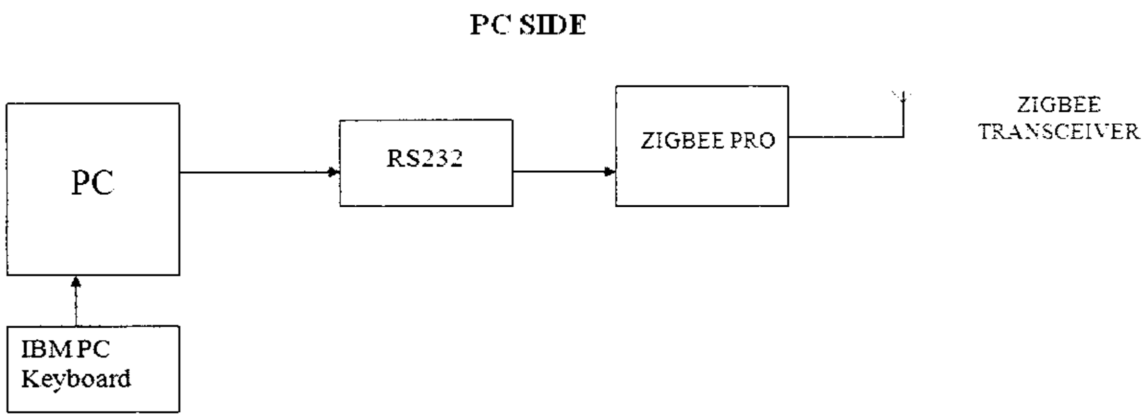


Fig 2.1 Block diagram of control section

2.1.2 CONVEYOR BLOCK:

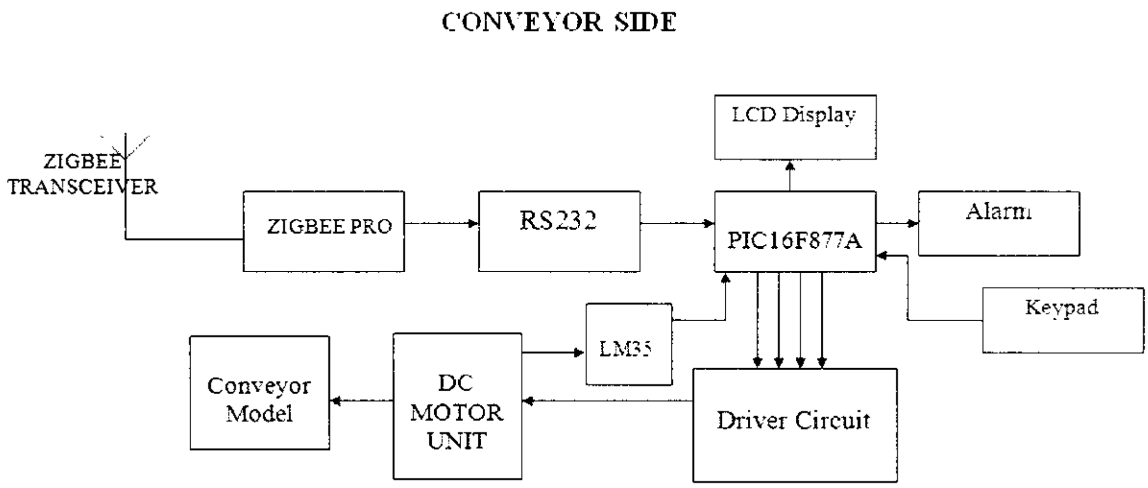


Fig 2.2 Block diagram of conveyor section

2.2 BLOCK DESCRIPTION:

The block diagram is bifurcated as follows

Control block:

It consists of a PC, RS232 cable, MAX-232 IC, Zigbee module and its driver circuit. The PC is configured in VB interface for displaying various parameters of the conveyor belt. The power ON/OFF of the conveyor can also be choosen from here. The control signals are being transmitted by using Zigbee to the conveyor.

Conveyor block:

This block contains the main hardware interfaces. A DC motor circuit along with its driver unit is connected to the PIC microcontroller. The motor unit drives the conveyor belt and our aim is to control the functionality of this motor unit. A temperature sensor is connected DC motor, PIC microcontroller converts these analog values to the digital values with the help of inbuilt ADC. A keypad, alarm and LCD display were used for user-interface. All these were connected to the microcontroller and values will be transmitted to the controller block via Zigbee and the further communications will be received from the controller block. A keypad is used to control the conveyor belt when zigbee undergoes failure.

2.3 CIRCUIT DIAGRAM AND EXPLANATION:

2.3.1 CIRCUIT DIAGRAM OF CONVEYOR SECTION:

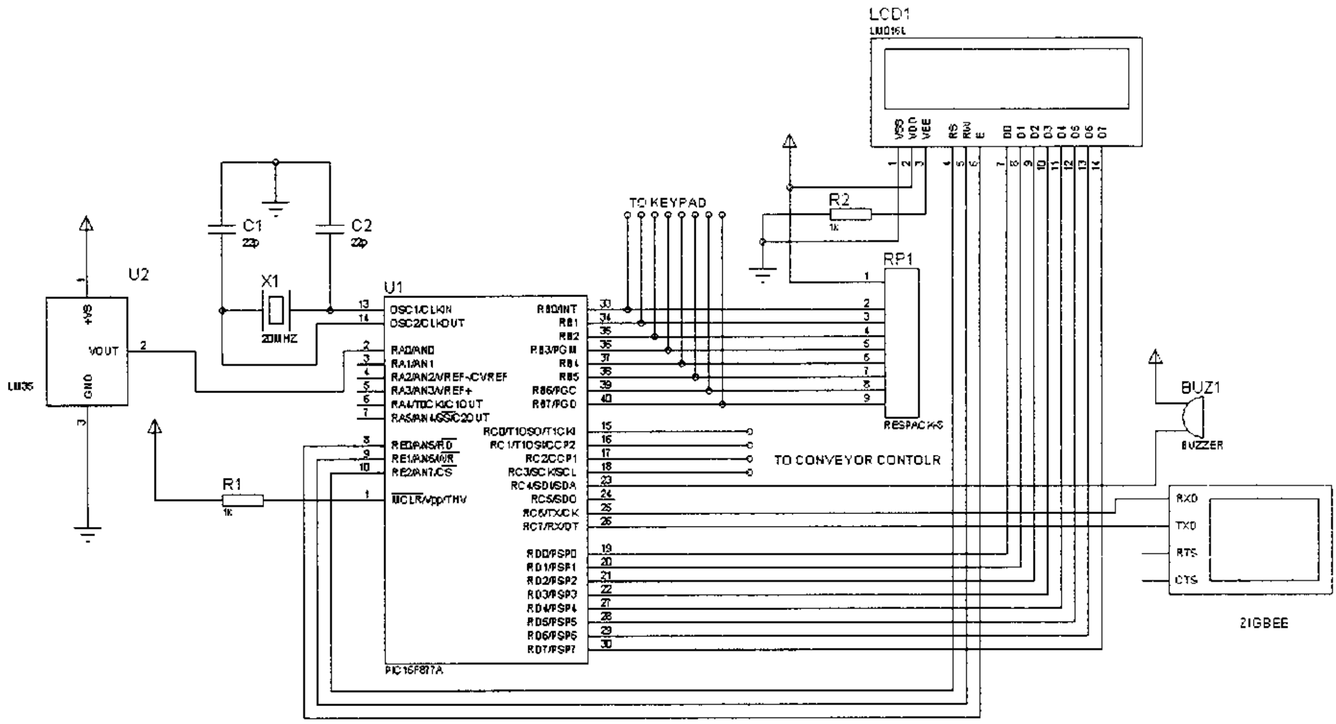


Fig 2.3 Circuit diagram of Conveyor section

The 230 V AC supply is stepped down to 12 V AC by step down transformer. The bridge rectifier converts the 12 V AC to 12 V DC. The filter capacitor filters the noise. Now the regulators 7805 and 7812 are used to supply a constant voltage of 5V and 12 V DC to various circuits. The relay circuit uses 12 V DC supply whereas the other circuits uses 5V DC supply. The LED is used to indicate the flow of constant current.

The microcontroller uses crystal oscillator which is used to trigger the circuit. The PIC16FF877A microcontroller is used to control the functions of conveyor belt depending upon the commands received from PC through through ZigBee . The LCD is used to indicate the status of conveyor belt and measured temperature. The change in functions of conveyor is indicated by buzzer ON for few seconds. The keypad is interfaced to control the conveyor when any failure occurs for ZigBee communication.

KEYPAD:

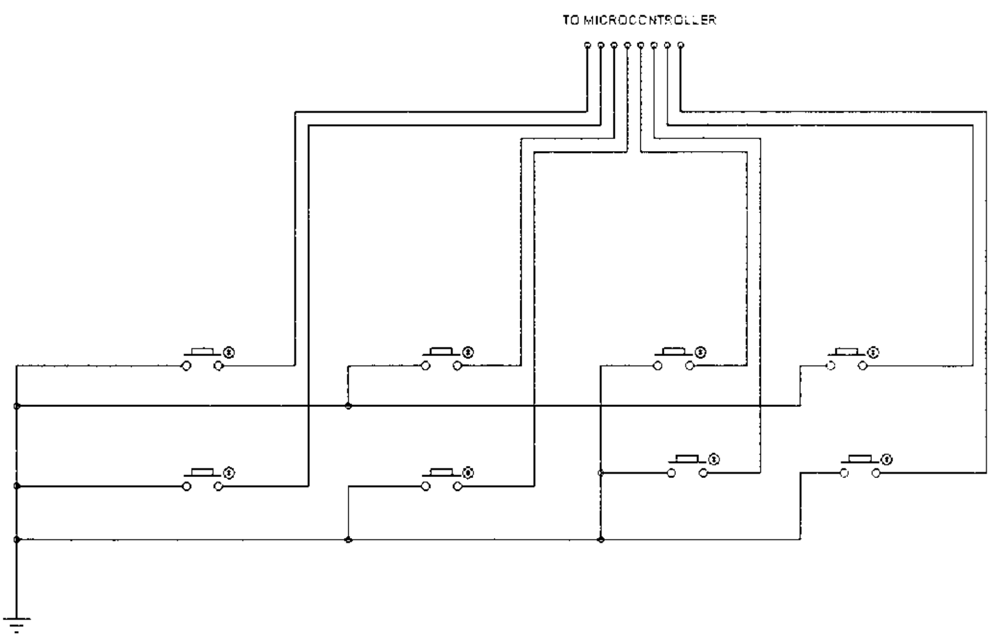


Fig 2.4 Circuit diagram of Keypad

A keypad is a device used to control the operation of our project event in the case of Zigbee transceiver failure in the Zigbee transmitter section.

A keypad controls the whole operation programmed by the user in the Bluetooth receiving section and it contains eight switches. Each switch contains separate function of our conveyor with user security protection.

ALARM:

The alarm in our project contains both buzzer and LED .It determines whether the command receive or not in the Xbee receiving section. The alarm circuit sounds a warning in the form of a continuous or intermittent buzzing or beeping sound.

LCD DISPLAY:

LCD's also are used as numerical indicators, especially in digital watches. LCD needs much smaller current than LED displays (microamperes compared with mill amperes) prolong battery life. Liquid crystals are organic (carbon) compounds, which exhibit both solid and liquid properties.

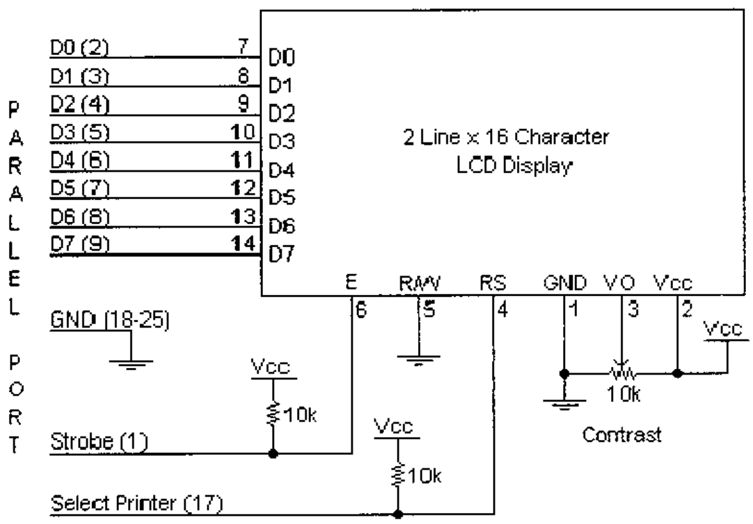


Fig no 2.5 Circuit diagram of LCD display

A 'cell' with transparent metallic conductors, called electrodes, on opposite daces, containing a liquid crystal, and on which light falls, goes 'dark' when a voltage is applied across the electrodes. The effect is due to molecular rearrangement within the liquid crystal. The LCD display used in our project consists of 2 rows. Each row consists of maximum 16 characters. So using maximum of 32 characters can be displayed.

2.3.2 CIRCUIT DIAGRAM OF PC SECTION:

In the computer controlled section, the data which send from controller of conveyor can be received and processed. The control of conveyor belt can be made through the keyboard of PC. The ZigBee transceiver can be interfaced to the PC using UART-RS 232 cable.

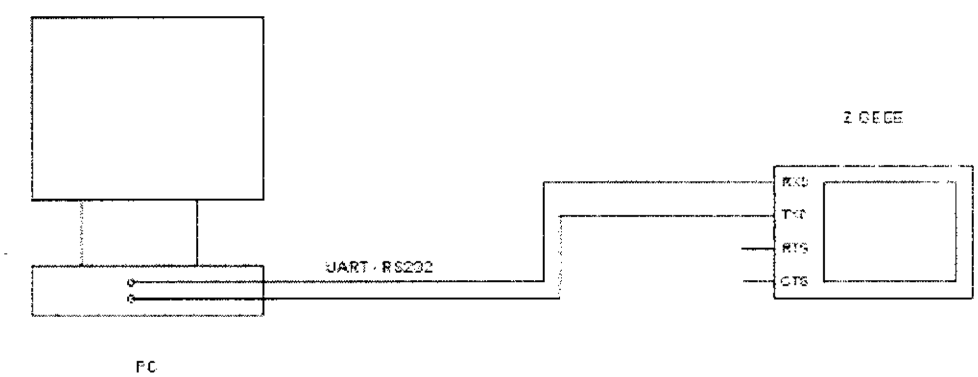


Fig no 2.6 Circuit diagram of PC section

ZIGBEE PRO

3.2 FUNCTIONAL CAPABILITY:

3.2.1 ZIGBEE STANDARD:

ZigBee was created to address the market need for a cost-effective, standards-based wireless networking solution that supports low data-rates, low power consumption, security and reliability. ZigBee uses the PHY and MAC layers defined by the IEEE 802.15.4 standard for short distance wireless communications. The ZigBee PRO network stack supports star and mesh network topologies, and provides robust communication. Developed on the IEEE802.15.4 standard, which operates in the globally unlicensed ISM 2.4GHz band, ZigBee PRO co-exists with Wi-Fi and Bluetooth. Multiple levels of security ensure that the network and data remain intact and secure.

3.2.2 ZIGBEE WIRELESS MODULE:

The board is configured with either an XBee-Pro transceiver that is wired to a SMA RF connector on the edge of the board. The difference between these two modules is the amount of power consumed (1mW vs. 60mW) and signal range. For the XBee-Pro configuration, the indoor and urban range is up to 300 feet (100m) and outdoor line-of-sight is up to 1 mile (1500m). For the XBee the urban range is 100 feet (30m) and outdoor range is 300 feet (100m). Selection of the antenna type and placement has a major impact on the range.

By default, the XBee modules are configured to support NonBeacon communications. NonBeacon systems operate with a peer-to-peer topology and are not dependent upon Master/Slave relationships. This provides fast synchronization times and fast cold starts times.

It can also be programmed to support NonBeacon mode as either a Coordinator or End Device. In a NonBeacon (w/Coordination) system, the Coordinator mode can be configured to use direct and indirect transmissions. It will also support Unicast or Broadcast communications.

The ZigBee module requires a serial asynchronous data stream for command and data. If the data cannot be sent out immediately (for example if it is currently receiving data) then the serial data will be stored in a buffer and sent when the RF channel is clear.

INTERRUPTS:

Interrupts are generated on error conditions and/or receive and transmit buffer status for each of the serial channels. Each of the interrupt outputs is wired to a jumper header and then to the PC/104 Bus connector. The PCM-ZigBee supports IRQ channels 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, and 15. If the GPS module is installed, then the 1 pulse per second output can be connected to an IRQ input.

SERIAL COMMUNICATIONS:

The XBee-PRO OEM RF Modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can communicate with any logic and voltage compatible UART; or through a level translator to any serial device (For example: Through a Max- Stream proprietary RS-232 or USB interface board). Devices that have a UART interface can connect directly to the pins of the RF module as shown in the figure below.

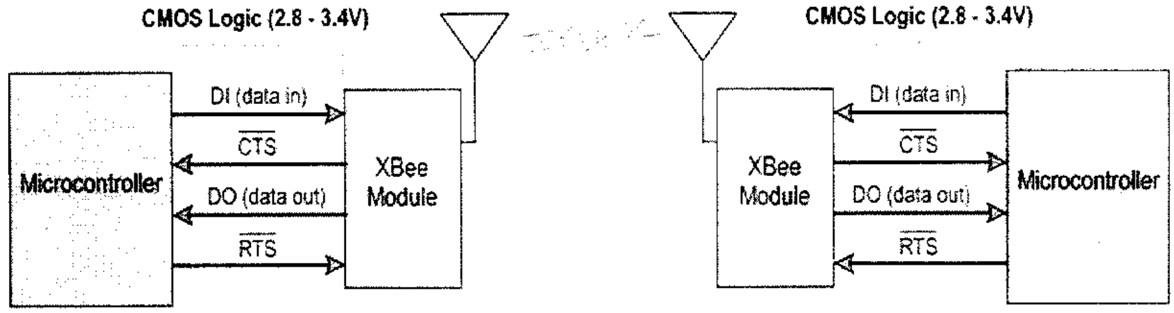


Figure 3.2 System Data Flow Diagram in a UART-interfaced environment

Data enters the module UART through the DI pin (pin 3) as an asynchronous serial signal. The signal should idle high when no data is being transmitted

ANTENNA OPTIONS:

The following antennas have been tested and approved for use with the embedded XBee-PRO RF Module:

• Dipole (2.1 dBi, Omni-directional, Articulated RPSMA, MaxStream part number A24-HABSM)

• Chip Antenna (-1.5 dBi)

• Attached Monopole Whip (1.5 dBi)

The RF modem encasement was designed to accommodate the RPSMA antenna option. This device has been designed to operate with the antennas having a maximum of 17.5 dB. The required antenna impedance is 50Ω.

EXTERNAL INTERFACE:

The development kit includes an RS-232 and a USB interface board. Both boards provide a direct connection to many serial devices and therefore provide access to the RF module registries. Parameters stored in the registry allow OEMs and integrators to customize the modules to suite the needs of their data radio systems. Since the module requires signals to enter at TTL voltages, one of the main functions of the interface board is to convert signals between TTL levels and RS- 232 and USB levels.

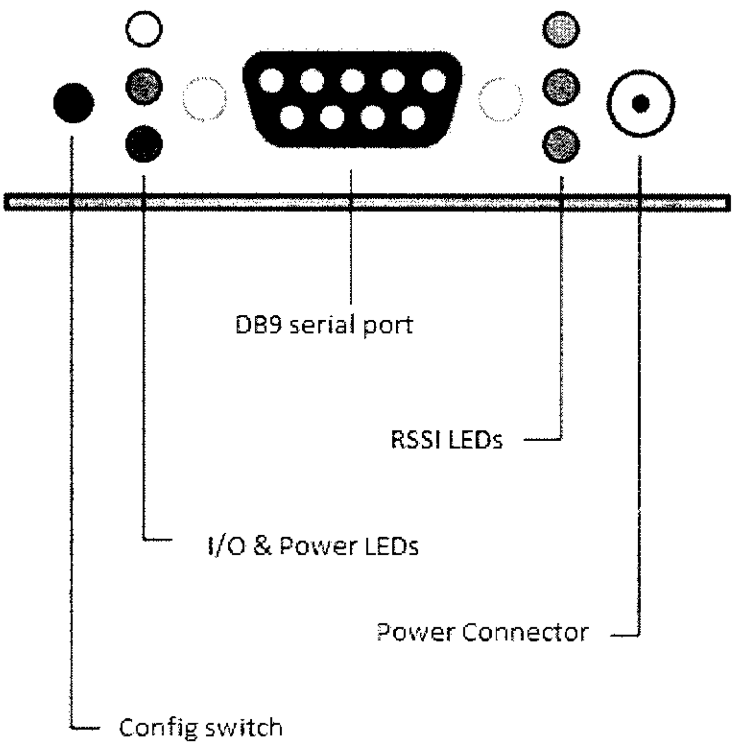


Figure 3.3 External interface-Front view

POWER SUPPLY:

A 3.3 volt regulator is on board to supply to the XBee-Pro module so that the PCM-ZigBee only requires +5V to operate.

3.3 ZIGBEE PIN SIGNALS:

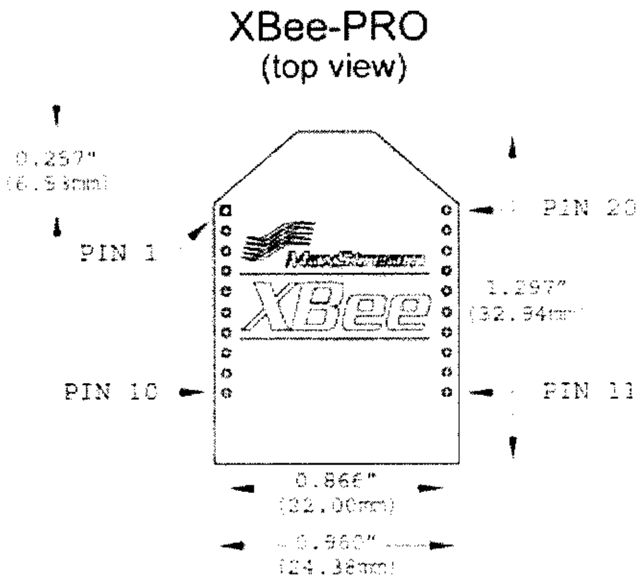


Figure 3.4 Mechanical drawings of ZigBee Pro

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0

Table no.1 Pin description of ZigBee transceiver

DESIGN NOTES:

- Minimum connections: VCC, GND, DOUT & DIN
- Minimum connections for updating firmware: VCC, GND, DIN, DOUT, RTS & DTR
- Signal Direction is specified with respect to the module
- Module includes a 50k Ω pull-up resistor attached to **RESET**
- Several of the input pull-ups can be configured using the PR command
- Unused pins should be left disconnected

3.4 ZIGBEE TRANSCEIVER:

DIAGRAM:

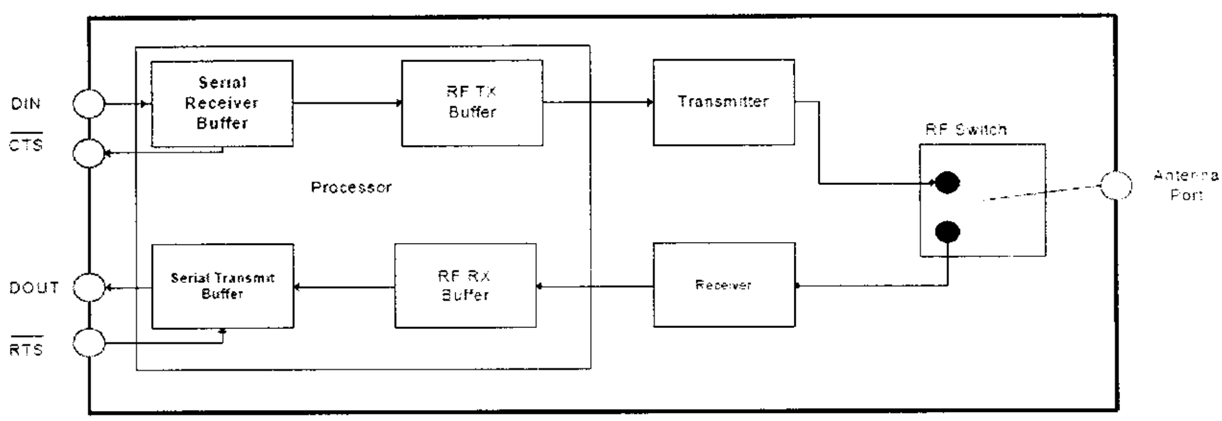


Fig 3.5 Block Diagram of ZigBee Transceiver

POWER SUPPLY

CHAPTER 4

POWER SUPPLY

4.1 INTRODUCTION:

The present chapter introduces the operation of power supply circuits built using filters, rectifiers and then voltage regulators. Starting with a AC voltage, a steady DC voltage is obtained by rectifying the AC voltage, then filtering to a DC level, and finally regulating to obtain a desired fixed DC voltage.

4.2 BLOCK DIAGRAM:

The ac voltage, typically 220V rms, is connected to a transformer, which steps that ac voltage down to the level of the desired dc output. A diode rectifier then provides a full-wave rectified voltage that is initially filtered by a simple capacitor filter to produce a dc voltage.

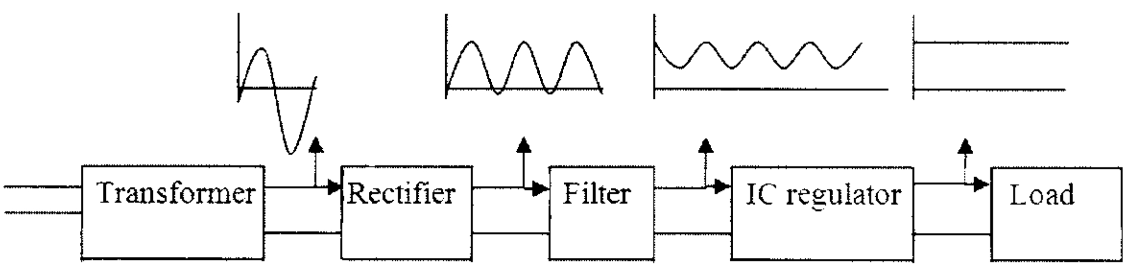


Fig 4.1-Block Diagram of power supply

4.3 WORKING PRINCIPLE:

4.3.1 TRANSFORMER:

The potential transformer will step down the power supply voltage (0-230V) to (0-6V) level. Then the secondary of the potential transformer will be connected to the precision rectifier, which is connected with the help of op-amp. The advantages of using precision rectifier, it will give peak voltage output as DC, rest of the circuits will give only RMS output.

4.3.2 BRIDGE RECTIFIER:

When four diodes are connected as shown in figure, the circuit is called as bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network, and the output is taken from the remaining two corners.

Let us assume that the transformer is working properly and there is positive potential, at point A and a negative potential at point B. The positive potential at point A will forward bias D3 and reverse bias D4.

The negative potential at point B will forward bias D1 and reverse bias D2. At this time D3 and D1 are forward biased and will allow current flow to pass through them; D4 and D2 are reverse biased and will block current flow.

The path for current flow is from point B through D!, up through RL, through D3, through the secondary of the transformer back to point B. This path is indicated by solid arrows. Wave forms (1) and (2) can be observed across D1 and D3.

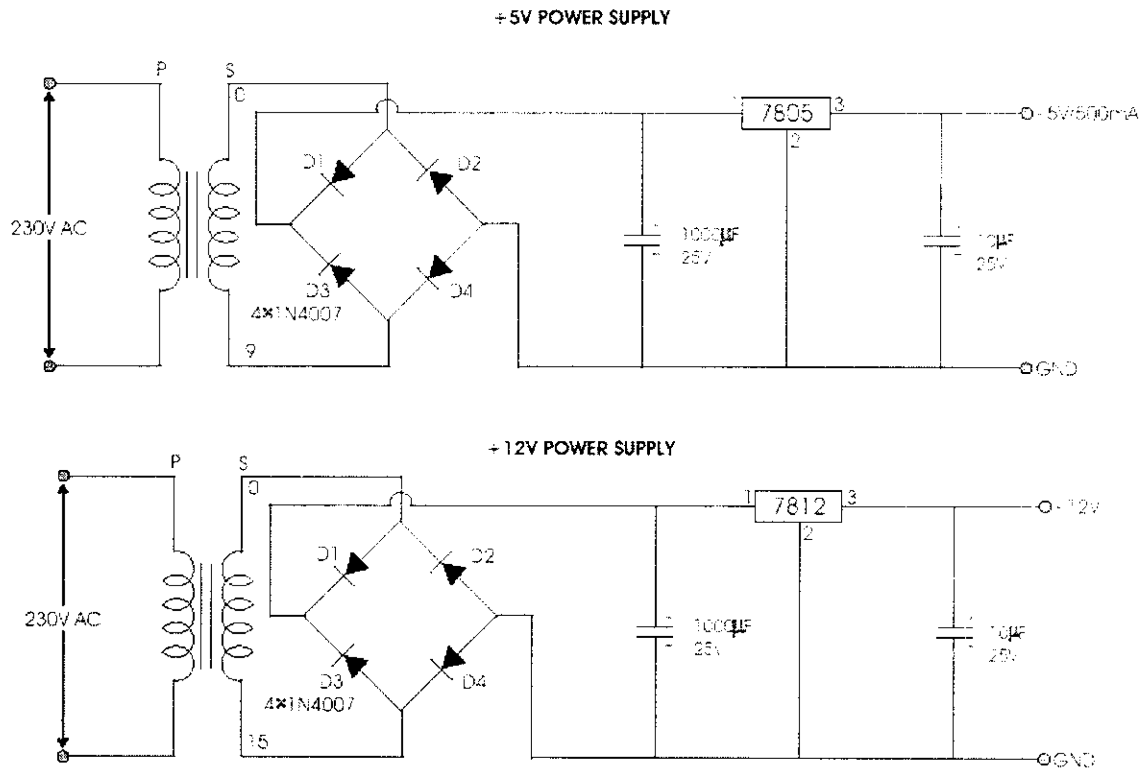


Fig 4.2-Circuit Diagram of power supply

One-half cycle later the polarity across secondary of the transformer forward biasing D2 and D4 and reverse biasing D1 and D3. Current flow will now be flow from point a through D4, up through Rl, through D2, through the secondary of T1, and back to point A. This path is indicated by the broken arrows. Waveforms (3) and (4) can be observed across D2 and D4. The current floe through RL is always in same direction. In flowing through RL, this current develops a voltage corresponding to that shown waveform (5).since current flows through the load (RL) during both half cycles of applied voltage, this bridge rectifier is a full wave rectifier.

One advantage of a bridge rectifier over a conventional full-wave rectifier is that with a given transformer the bridge rectifier produces a voltage output that is nearly twice that of the conventional full-wave circuit.

This may be shown by assigning values to some of the components shown in views A and B. Assume that the same transformer is used in both the circuits. This peak voltage is developed in between X and Y is 1000 volts in both circuits. In the conventional full-wave circuit shown- in view A, the peak voltage that can be rectified at any instant is 500 volts. Since only one diode can conduct at any instant, the maximum voltage that can be rectified at any instant is 500 volts

The maximum voltage that appears across the load resistor is nearly- but exceeds-500 volts, as result of the small voltage drop across the diode. In the bridge rectifier shown in view B, the maximum voltage that can be rectified is the full secondary voltage, which is 1000 volts. Therefore, the peak output voltage across the load resistor is nearly 1000 volts. With both circuits using the same transformer, the bridge rectifier circuit produces a higher output voltage than the conventional full-wave rectifier circuit.

4.3.3 FILTERS:

Capacitors are used as filters in the power supply unit. Shunting the load with the capacitor., affects filtering. The action of the system depends upon the fact the capacitor stores energy during the conduction period and delivers this energy to the load during the inverse or non-conducting period. In this way, time during which the current passes through the load prolonged and ripple is considerably reduced.

4.3.4 IC VOLTAGE REGULATORS:

Voltage regulators comprise a class widely used ICs. Regulator IC unit contain the circuitry for reference source, comparator amplifier, control device and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative , or an adjustably set voltage. The regulators can be selected for operation with load currents from hundreds of milli amperes, corresponding to power ratings from milli watts to ten of watts.

RELAY

CHAPTER 5

RELAY UNIT

5.1 INTRODUCTION:

Relays are components, which allow a low power circuit to switch a relatively high current ON and OFF, or to control signals that must be electrically isolated from the controlling circuit itself. The DC motor operates at 12 V through relay.

5.2 RELAYS WORK:

Relays are composed of a coil of wire around a steel core, a switch, and a spring that hold one or more contacts.

When an electrical flows through the coil it becomes energized, acting like an electromagnet. The refuse field opens the contacts, and closes the circuit. When electrical current stops flowing, the opposite occurs.

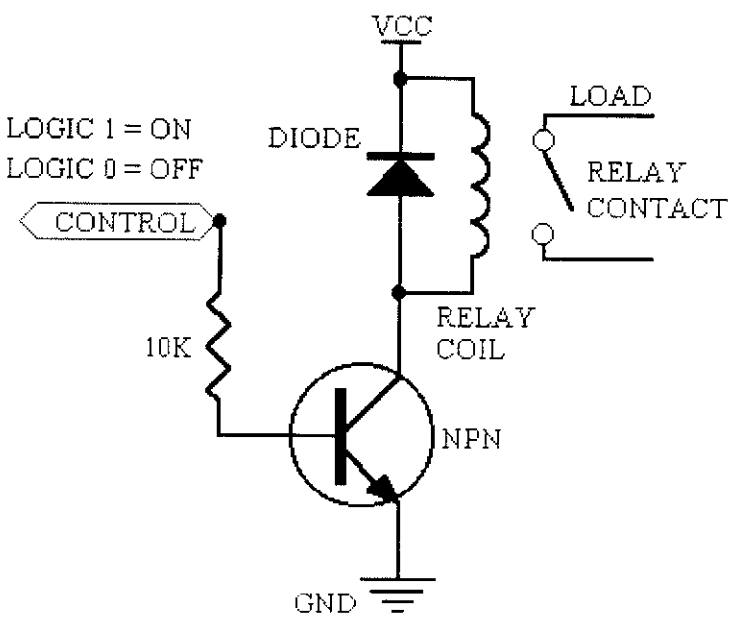


Figure 5.1 circuit diagram of relay

5.3 DRIVER CIRCUIT:

The circuit is designed to control the motor in the forward and reverse direction. The circuit consists of a relay. The relay ON and OFF is controlled by the switch transistor.

The relay is nothing, but an electro magnetic switch device which consists of three pins. They are common, Normally Closed and Normally Opened. The common pin of relay is connected to positive and negative terminal of the motor through snubber circuit.

The relay is connected in the collector terminal of the transistor. When high pulse signals is given to base of the transistor, the transistor conduct and shorts the collector and emitter terminal and zero signals are given to base of the transistor so the relay is turned OFF state. When low pulse is given to the base of the transistor, the transistor conduct and the relay is turned ON.

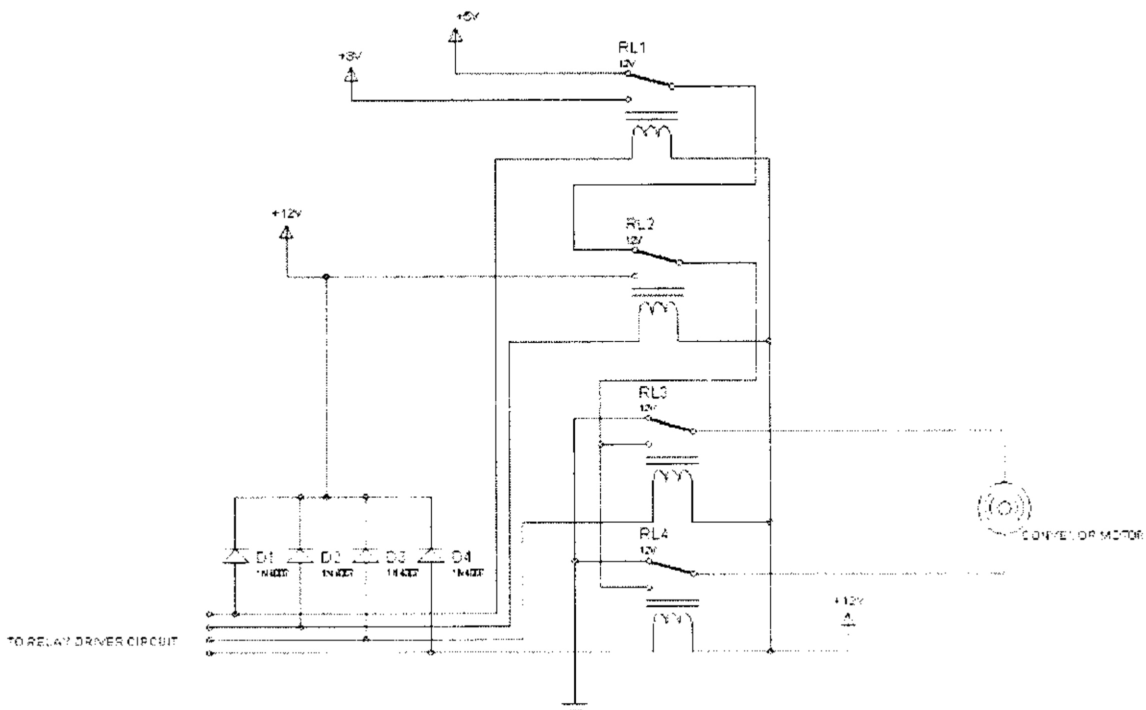


Figure 5.2 Circuit diagram for Relay

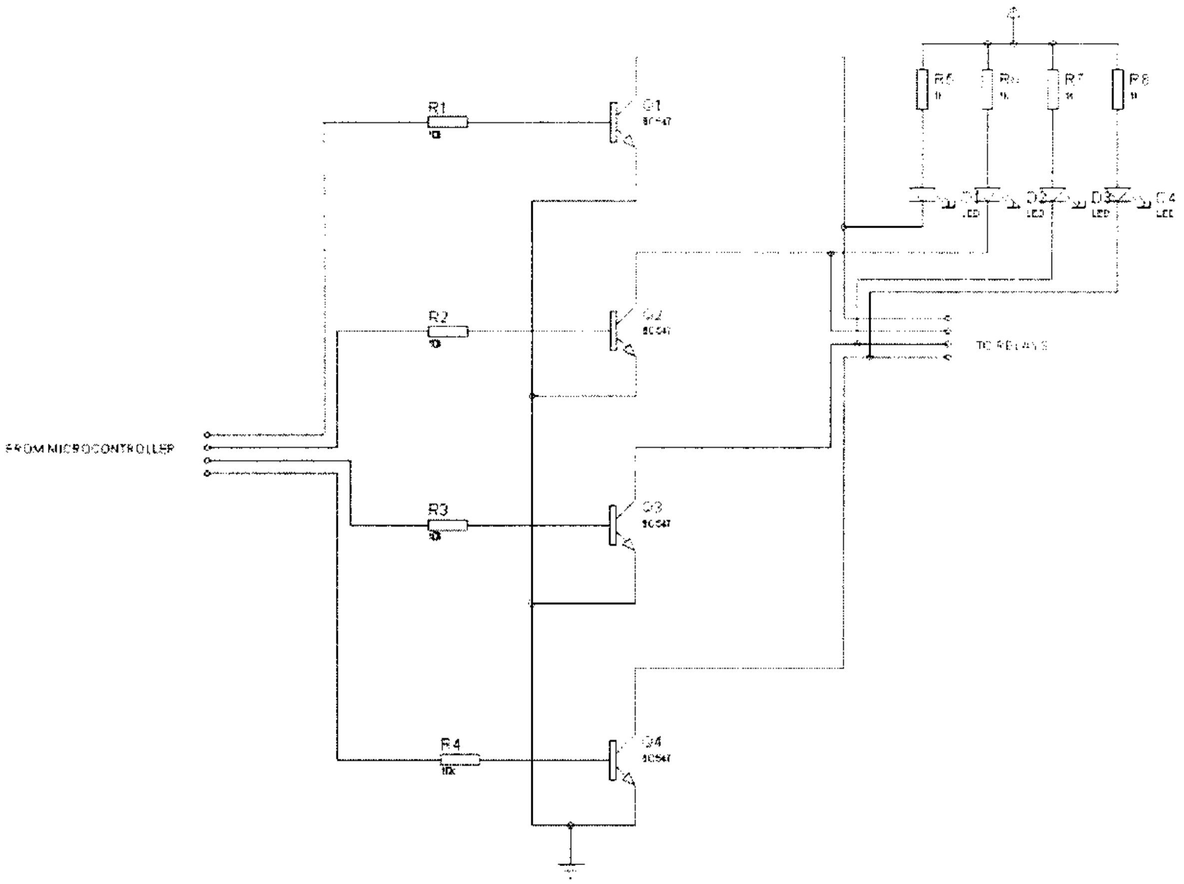


Figure 5.3 Circuit diagram of relay driver

Circuit symbol for a relay:

The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs (chips) cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil.

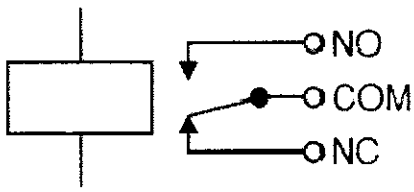


Figure no 5.2 Circuit symbol for a relay

The relay's switch connections are usually labelled COM, NC and NO

- **COM** = Common, always connect to this, it is the moving part of the switch.
- **NC** = Normally Closed, COM is connected to this when the relay coil is **off**.
- **NO** = Normally Open, COM is connected to this when the relay coil is **on**.

Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits, the link is magnetic and mechanical.

MICROCONTROLLER

CHAPTER 6

MICROCONTROLLER

6.1 INTRODUCTION:

Microcontroller helps in every automation and controlling applications.

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)

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

FEATURES OF PIC MICROCONTROLLER:

- The PIC microcontroller family meets the specifications contained in the Microchip Data Sheet.

- Microchip believes that its family of PIC microcontrollers is one of the most secure products of its kind on the market today.

- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PIC microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.

- Microchip is willing to work with the customer who is concerned about the integrity of their code.

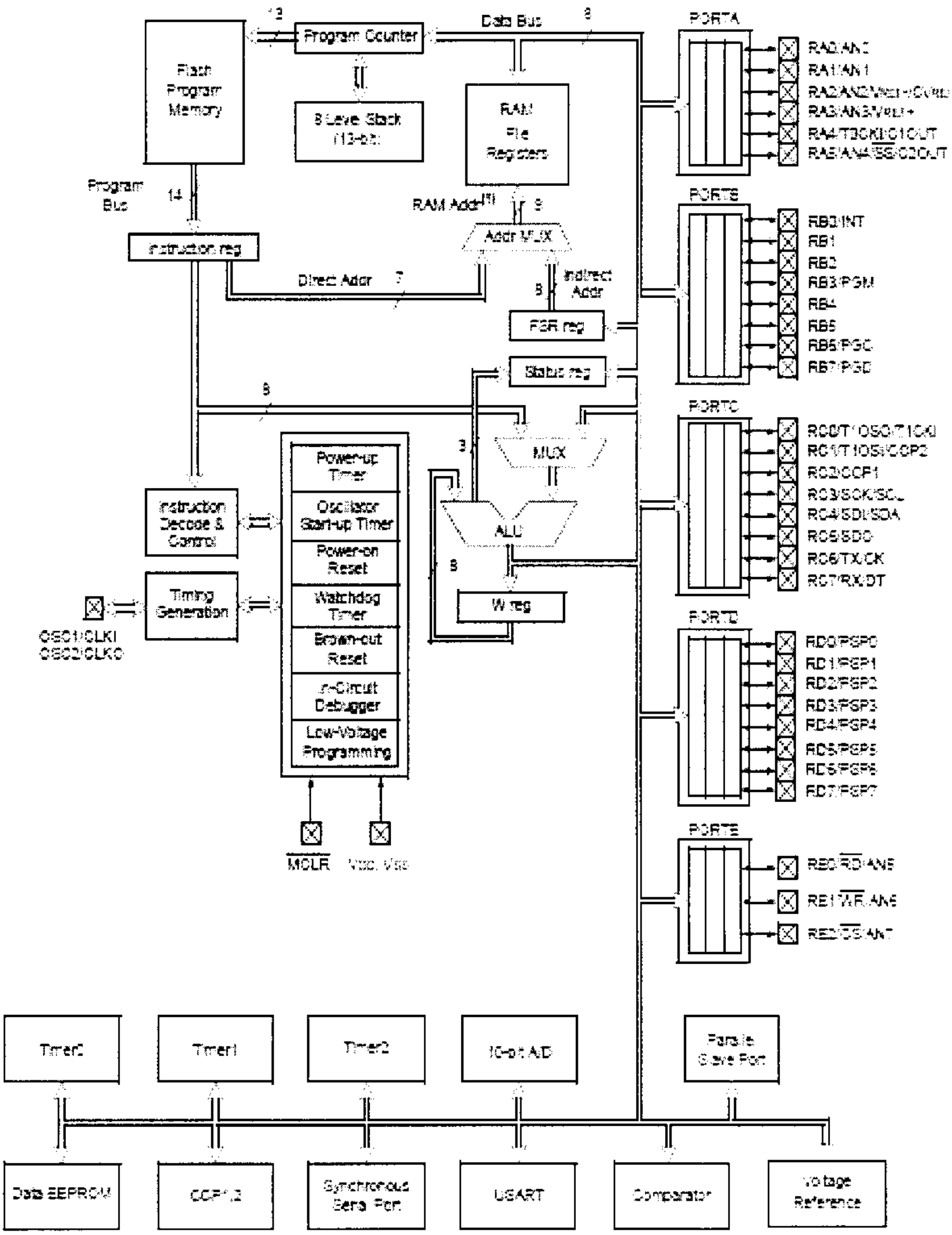
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable".

- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

6.2 ARCHICTECTURE:

PIC16F87XA

BLOCK DIAGRAM



6.3 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.

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.

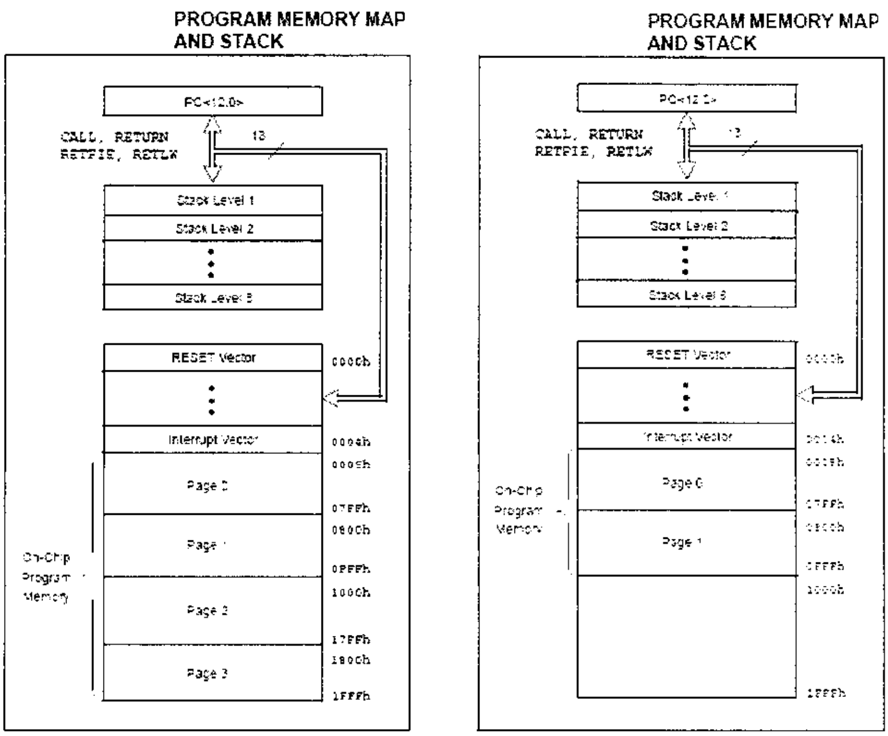


Fig no. 6.2 Memory Organization of Microcontroller

6.4 PIN CONNECTION:

- Port C – to zigbee transceiver, conveyor machine and buzzer
- Port B – to keypad incase zigbee fails
- Port D – as data lines to LCD
- Port A – to temperature sensor and for future extension
- Port E – as control lines for LCD

6.5 PIN DIAGRAM:

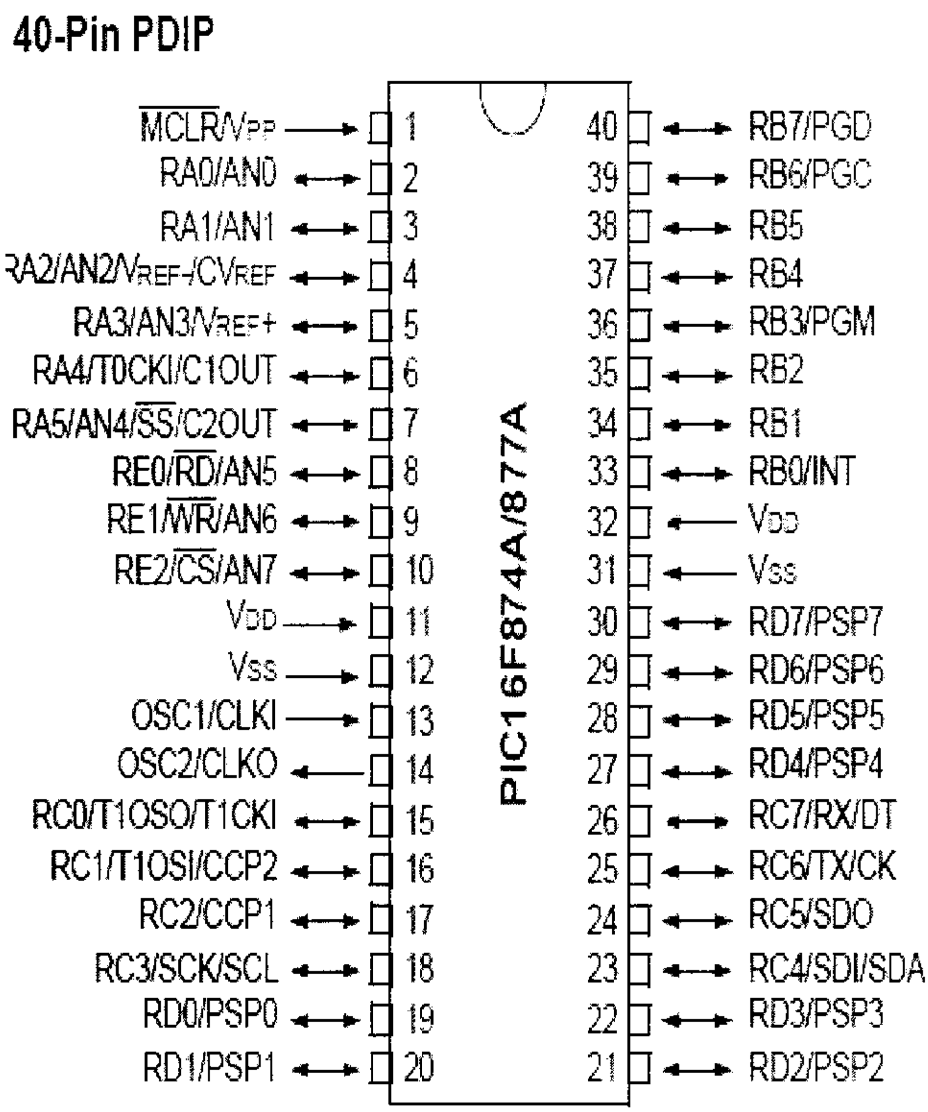


Figure 6.3 Pin diagram of PIC16F877A

TRENDS AND DEVELOPMENTS IN MICCROCONTROLLER:

The manner in which the use of microcontrollers is shaping our lives is breathtaking. Today, this versatile device can be found in variety of control applications. CVTs, VCDs, CD players, microwave ovens, and automotive engine systems are some of these.

Microcontroller unit (MCU) uses the microprocessor as its central processing unit (CPU) and incorporates memory, timing reference, I/O peripherals, etc on the same chip. Limited computational capabilities and enhanced I/O are special features.

The microcontroller is the most essential IC for continuous process based applications in industries like chemical, refinery, pharmaceutical automobile, steel, and electrical, employing programmable logic systems (DCS). PLC and DCS thrive on the programmability of an MCU. There are many MUC manufactures. To understand and apply general concepts, it is necessary to study one type in detail. This specific knowledge can be used to understand similar features of other MCU.

Microcontroller devices have many similarities. When you look at the differences, they are not so great either. Most common and popular MCU are considered to be matured and well- established products, which have their individual adherents and devotees. There are a number of variants within each family to satisfy most memory, I/O, data conversion, and timings of end user applications.

The microcontroller is designed to operate on application-oriented sensor data-for example, temperature and pressure of a blast furnace in an industrial process that is fed through its serial or operated on under the control of software and stored in ROM. Appropriate signals are fed via output ports to control external devices and systems.

Incoming data can be from a variety of external devices sensors. Priority for receiving and operating on the data can be established using the interrupt-control circuitry. The microcontroller operates with external circuitry to perform control oriented tasks, using a control program in ROM. The instruction set is simpler than that of a microprocessor, since most of its instructions will move code and data from internal memory to ALU (arithmetic logic unit).

The use of many input/output pins allows data to be moved between the microcontroller and external devices, often as single bits. Operation on single bits, such as logical operations, is unique to microcontrollers; microprocessors operate generally on bytes or larger data groups. The microcontroller is designed to allow programming for single task operations, which is ideal for applications requiring large product volumes. The microcontroller includes registers (not only used by its CPU but by timer), interrupt control units, port latches, etc.

6.6 PROGRAMS:

To program microcontrollers, the trend is to use the C language, due to its efficiency and ease of use relative to the assembly language makes most things possible using the least amount of memory (code and date) and time, and offers increased productivity.

Typically, a software developer can write more codes to do more things using C than when using assembly. This is important because at least half the cost of developing an MCU application is in paying people to develop the software.

Since C source code is standardized and portable, many people know how to program in C. It can be written anywhere and then complied for the target processor of choice. Writing microcontroller software often requires knowledge of bits, registers, etc. C is considered to be a good language for real time applications, as it has more or less compactness and special features of the high level languages of portability. Also flow control is more flexible and easy to use. There is still a place for assembly as many applications (such as microwave ovens, coffee machines and VCRs) need small and cheap microcontrollers in order to keep the price reasonable. These applications use MCUs with limited memory resources. To get the best of both worlds, programmers use both C and assembly languages when speed is critical, C has a provision to allow assembly language code to be embedded within it. An application code with time-critical sections in assembly and the rest in C is one option.

SERIAL INTERFACE

CHAPTER 7

SERIAL INTERFACE – RS 232

7.1 INTRODUCTION:

Serial data is any data that is sent one bit at a time using a single electrical signal. In contrast, parallel data is sent 8,16, 32, or even 64 bits at a time using a single line for each bit. Data that is sent without the use of a master clock is said to be asynchronous serial data.

Several communications standards exists for the transfer of asynchronous serial data. Common PC's transfer data using the EIARS-232C (also known as V.28 or V.24). Updated versions of this standards imclude RS-232D and EIA/TIA-232E, but most literature still refers to the RS-232 or RS-232 standard.

The baud rate for a serial connection is the number of bits that are transmitted per second. It is specified in bits/second or baud.

VOLTAGE LEVELS IN RS 232:

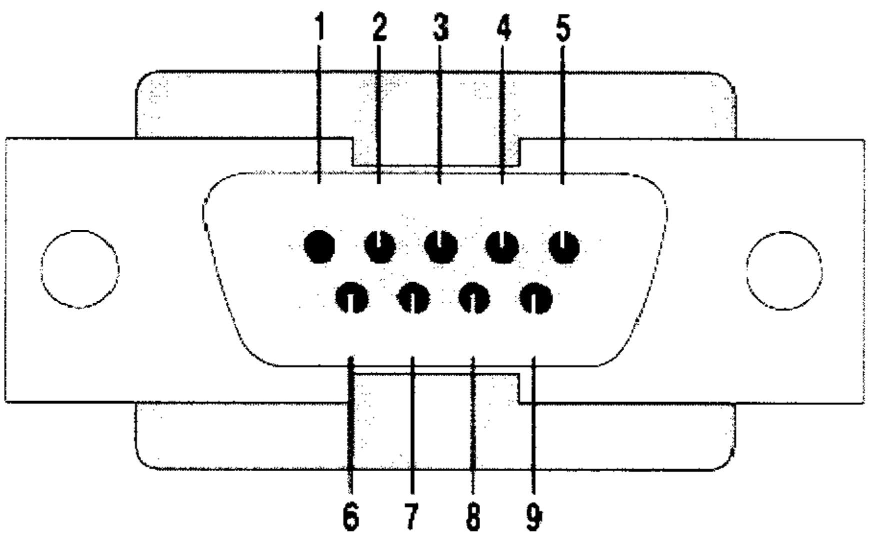
1. Logic high (1) represented as -3 to -25
2. Logic low (0) represented as +3 to +25
3. -3 to +3V not defined

7.2 RS 232 INTERFACE:

1. RS 232 was introduced in 1960, and is currently the most widely used communication protocol. It is simple, inexpensive to implement, and though relatively slow it is more than adequate for most simple serial communication devices such as keyboards and mice.
2. 232 is a single- ended data transmission system, which means that it uses a single wire for data transmission.
3. Since useful communication is generally two way, a two wire system is employed, one to transmit and one to receive.

RS 232 PINS:

The table provides the pins and the labels for RS 232 cable, commonly referred to as DB-25 connector.



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

Figure 7.1 RS 232 labels

The standards for RS 232 and similar interfaces usually restrict RS 232 to 20Kbps or less and line length of 15m or less. RS 232 is fair more robust than the traditional limits of 20kbps over a 15m line would imply RS 232 as perfectly adequate at speed upto 200kbps.

FEATURES OF RS 232:

The essential features of RS 232 is the signals are carried as signal voltages referred to a common earth and print data to a common earth pin, data is transmitted and received on pins 2 &3 respectively. Data Set Ready (DSR) is an indication for the data set i.e., the modem DSU/CSU i.e., it is on. Similarly, DTR indicates to the data set that the DTE is on. Data carrier (DCD) indicates that the carrier for data transmit is on. Pins 4 & 5 carry the RTS and CTS constantly go throughout the communication session. However when the DTE is connected to a turn carrier on the modem ON & OFF. On a multipoint linear, it is imperative that only one station is transmitting at a line when a station wants to transmit, it raises RTS the modem turns on carrier.

The truth table for RS 232

Signal >+3V=0

Signal <-3V=1

The output signal level usually swings between +12V and -12V. The dead area between +3V and -3V is designated to absorb line noise.

SERIAL PORT OF PC:

IBM PC/compatible computers based on x86(8086, 286,386,486 and pentium) microprocessors normally have two COM (read:COMMUNICATION) ports. Both COM ports have RS 232 type connectors. Many PCs use one each of DB-25 and DB-9 RS 232 connectors. The COM ports are designated as COM 1 , COM 2 etc.. One can utilize COM X port of a PC for serial communication experiments, where X designates a particular port. The RS 232 standard is not TTL compatible, therefore, it requires the line driver such as MAX 232 chip to convert the RS 232 voltage levels to TTL levels and vice versa.

7.3 MAX 232:

MAX 232 is primary used for people building electronics with an RS 232 interface. Serial RS-232communications works with voltages (-15V ... -3V for high) and +3V ... +15V for low) which are not compatible with normal computer logic voltages. To receive serial data from an RS-232 interface the voltages has to be reduced, and low and high voltage level inverted. In the other direction (sending data from some logic over RS-232) the logic voltage has to be “bumped up”, and a negative voltages has to be generated, too.

The MAX232 has two sets of the line drivers for transferring and receiving data. The line drivers used for TxD are called T1 and T2, while the line drivers for RxD designated as R1 and R2. in many applications only one of each is used. For example T1 and R1 are used together for TxD and RxD of the PIC and second set is left unused. Notice in MAX232 that the T1 line driver has a designated to T1 in a T1 out on pin numbers 11 and 14 respectively.

No	Name	Purpose	Signal Voltage
1	C1+	+connector for capacitor C1	Capacitor should stand at least 16V
2	V+	Output of voltage pump	+10V
3	C1-	-connector for capacitor C1	Capacitor should stand at least 16V
4	C2+	+connector for capacitor C2	Capacitor should stand at least 16V
5	C2-	-connector for capacitor C2	
6	V-	Output of voltage pump/inverter	-10V
7	T2out	Driver 2 output	RS232
8	R2in	Receiver 2 input	RS232
9	R2out	Receiver 2 output	TTL
10	T2in	Driver 2 output	TTL
11	T1in	Driver 1 input	TTL
12	R1out	Receiver 1 output	TTL
13	R1in	Receiver 1 input	RS-232
14	T1out	Driver 1 output	RS-232
15	GND	Ground	0V
16	Vcc	Power supply	+5V

Table no. 2 Pin description of MAX 232

The T1 in pin is the TTL side is connected to TxD of the microcontroller, while T1 out is the RS232 side that is connected to the RxD pin of the RS232 DB connector. The R1 line driver has a designation of R1 in R1 out pin numbers 13 and 12 respectively.

The R1 in pin 13 is the RS232 DB connected to the TxD pin of the RS232 DB connector and R1 out(pin 12) in the TTL side that is connected MAX232 requires 4 capacitors ranging from 1 to 22micro farad. The most widely used value for this capacitor is 22microfarad.

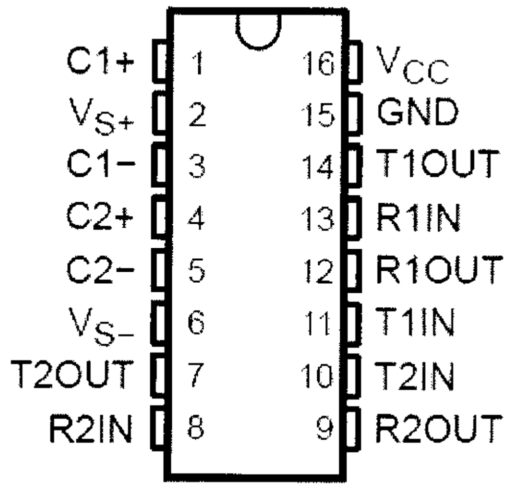


Figure no 7.2 Pin diagram of MAX 232

Features:

- Operate from single +5V power supply
- Low-power receive mode in shut down
- Multiple drivers and receivers
- 3-state drivers and Receiver outputs
- Open line detection

7.4 MAX 232 INTERFACING:

The data from microcontroller is made available to the serial port. The encoded RS232 level signal from pin 3 is given to pin 13 of MAX232 IC. The TTL level signal is obtained across pin 12 of the IC and is directly given to the microcontroller. Thus with this signal level conversion, the data is now ready to be transmitted.

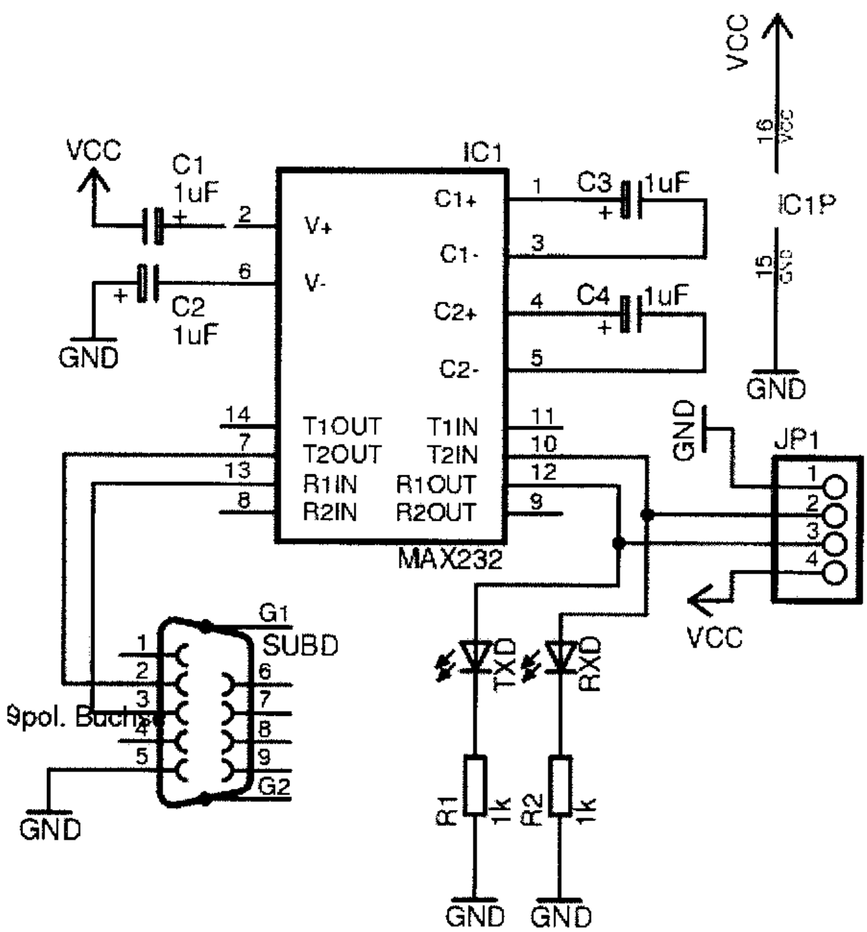
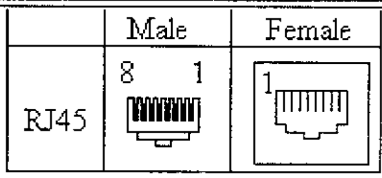


Fig no 7.3 Interfacing of MAX 232

The Electronics Industries Association (EIA) standard RS-232-C[1] as of 1969 defines:

- Functions of each circuit in the interface connector.

- Electrical signal characteristics such as voltage levels, signaling rate, timing and slew-rate of signals, voltage withstand level, short-circuit behavior, and maximum load capacitance.
- Interface mechanical characteristics, pluggable connectors and pin identification.
- Standard subsets of interface circuits for selected telecom applications.



Pin No.	Signal Description	Abbr.	DTE	DCE
1	DCE Ready, Ring Indicator	DSR/RI	←	→
2	Received Line Signal Detector	DCD	←	→
3	DTE Ready	DTR	→	←
4	Signal Ground	SG		
5	Received Data	RxD	←	→
6	Transmitted Data	TxD	→	←
7	Clear To Send	CTS	←	→
8	Request To Send	RTS	→	←

Table no. 3 Signal description of RS232 cable

Function Tables

EACH DRIVER

INPUT TIN	OUTPUT TOUT
L	H
H	L

H = high level, L = low level

EACH RECEIVER

INPUT RIN	OUTPUT ROUT
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)

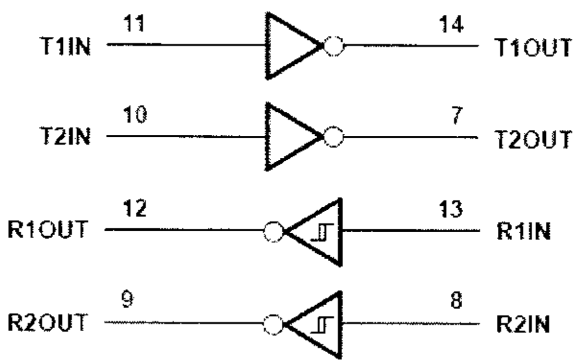


Table no. 4 Function table of RS232 cable

CHAPTER 8

GEAR MOTOR

8.1 INTRODUCTION:

Gear motors are complete motive force systems consisting of an electric motor and a reduction gear train integrated into one easy-to-mount and -configure package. This greatly reduces the complexity and cost of designing and constructing power tools, machines and appliances calling for high torque at relatively low shaft speed or RPM. Gear motors allow the use of economical low-horsepower motors to provide great motive force at low speed such as in lifts, winches, medical tables, jacks and robotics. They can be large enough to lift a building or small enough to drive a tiny clock.

8.2 OPERATING PRINCIPLE:

Most synchronous AC electric motors have output ranges of from 1,200 to 3,600 revolutions per minute. They also have both normal speed and stall-speed torque specifications. The reduction gear trains used in gear motors are designed to reduce the output speed while increasing the torque. The increase in torque is inversely proportional to the reduction in speed. Reduction gearing allows small electric motors to move large driven loads, although more slowly than larger electric motors. Reduction gears consist of a small gear driving a larger gear. There may be several sets of these reduction gear sets in a reduction gear box.

Speed Reduction:

Sometimes the goal of using a gear motor is to reduce the rotating shaft speed of a motor in the device being driven, such as in a small electric clock where the tiny synchronous motor may be spinning at 1,200 rpm but is reduced to one rpm to drive the second hand, and further reduced in the clock mechanism to drive the minute and hour hands. Here the amount of driving force is irrelevant as long as it is sufficient to overcome the frictional effects of the clock mechanism.

Torque Multiplication:

Another goal achievable with a gear motor is to use a small motor to generate a very large force albeit at a low speed. These applications include the lifting mechanisms on hospital beds, power recliners, and heavy machine lifts where the great force at low speed is the goal.

Motor Varieties:

Most industrial gear motors are AC-powered, fixed-speed devices, although there are fixed-gear-ratio, variable-speed motors that provide a greater degree of control. DC gear motors are used primarily in automotive applications such as power winches on trucks, windshield wiper motors and power seat or power window motors.

Many Applications:

What power can openers, garage door openers, stair lifts, rotisserie motors, timer cycle knobs on washing machines, power drills, cake mixers and electromechanical clocks have in common is that they all use various integrations of gear motors to derive a large force from a relatively small electric motor at a manageable speed. In industry, gear motor applications in jacks, cranes, lifts, clamping, robotics, conveyance and mixing are too numerous to count.

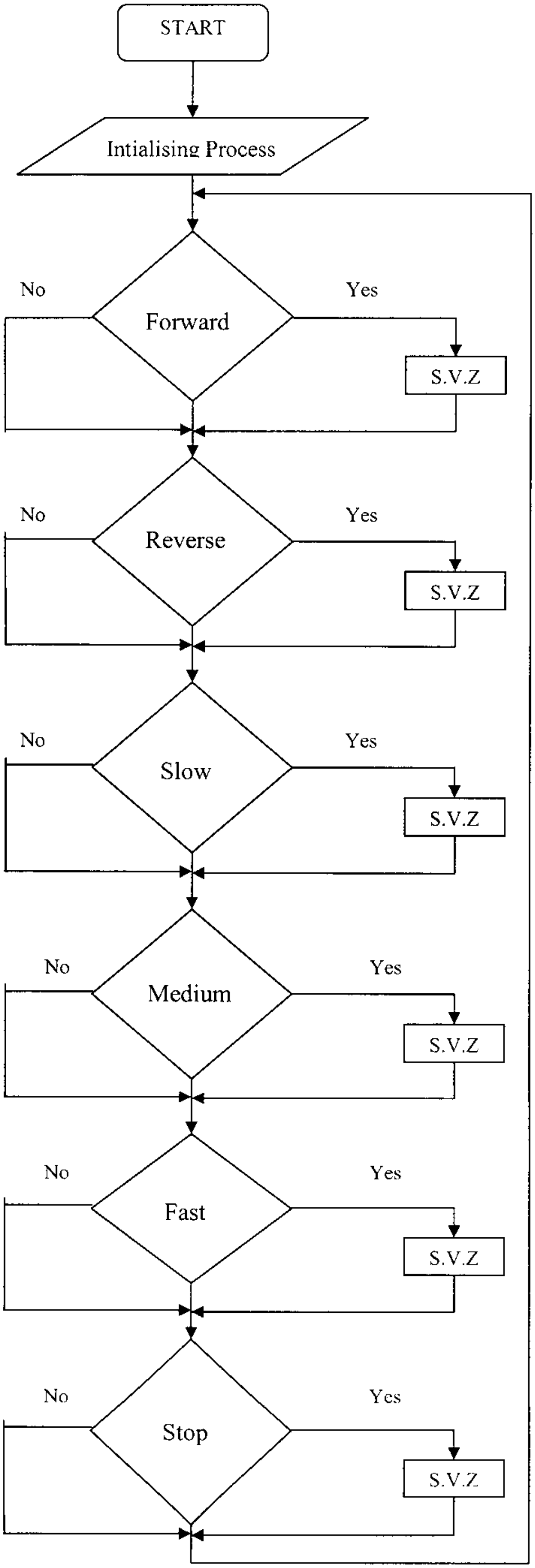
8.3 FEATURES:

GEAR MOTOR - HELICAL GEAR

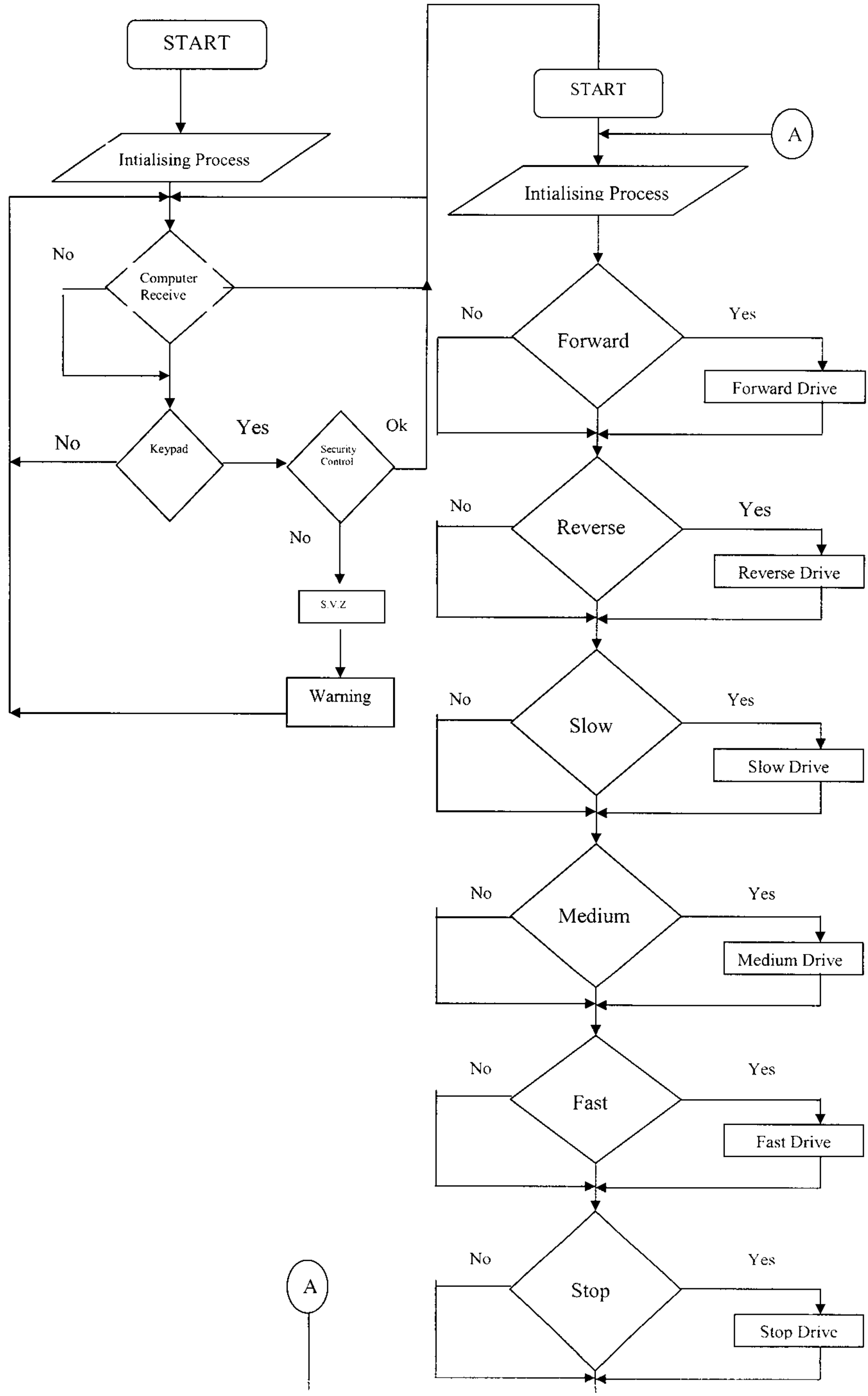
- 10RPM T0 1000RPM 12V DC geared motors for robotics applications
- Very easy to use and available in standard size.
- Nut and threads on shaft to easily connect Wheel
- 10RPM 12V DC motors with Gearbox
- 6mm shaft diameter with internal hole
- 125gm weight
- Same size motor available in various rpm
- 5kgcm torque
- No-load current = 60 mA(Max), Load current = 300 mA(Max)

FLOW CHART:

TRANSMITTING SECTION:



CONVEYOR SECTION:



CONCLUSION

CHAPTER 9

CONCLUSION

This project has a transmitting and receiving station. The conveyor belts with regularly spaced partitions works as per the command given in the PC, likewise the functionmality of whole length of conveyor belts are computer controlled which reduces the labour charge and reduces time consumption. Using more sophiscated technologies such as GSM, this project canbe enhanced to provide more convience for the user.

FUTURE ENHANCEMENT

CHAPTER 10

FUTURE ENHANCEMENT

- GSM can be used for radio communication as this has a long range of distance covered
- Java can be used as a programming language
- In case of any power failure in the transceiver unit, an additional buffer can be placed to carryout the work without any interruption

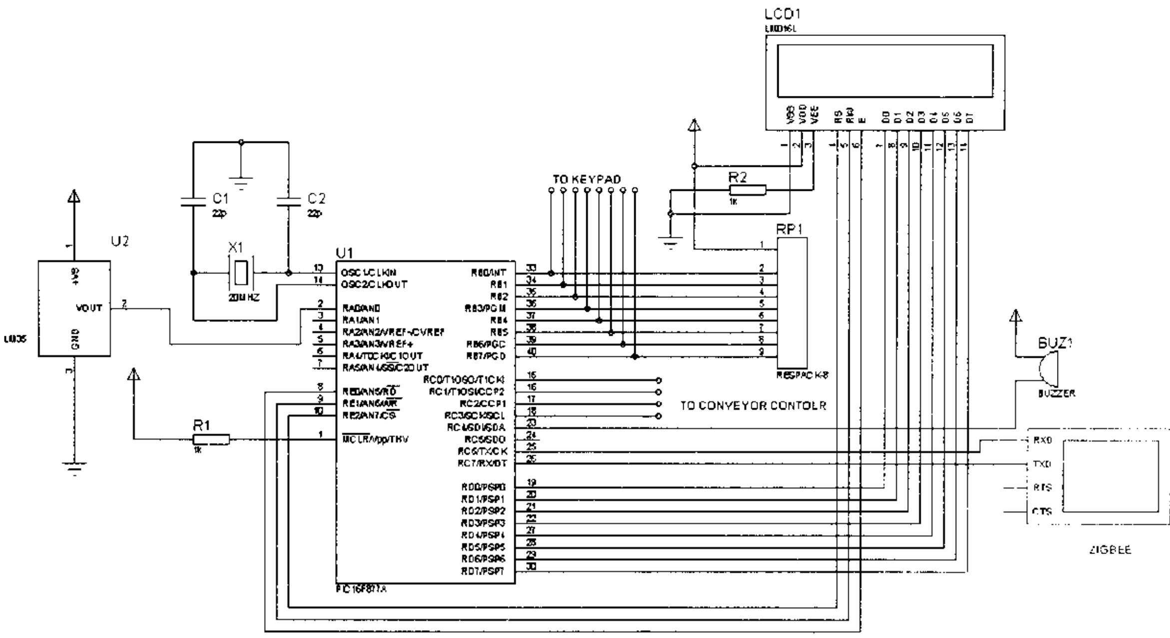
APPENDICES

APPENDICES

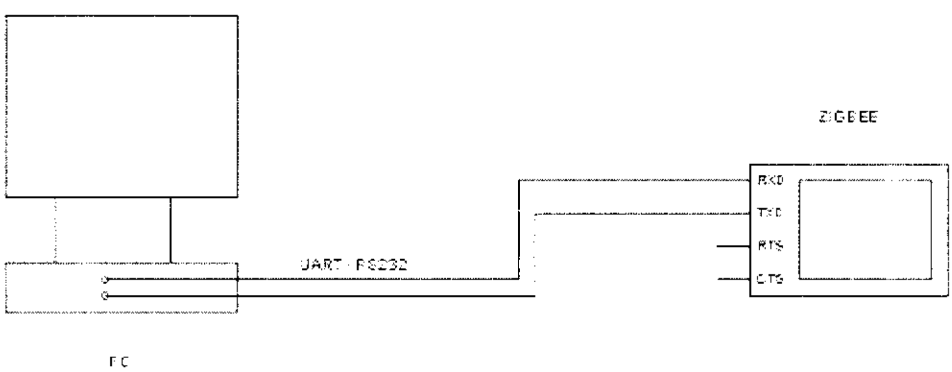
APPENDIX 1

COMPLETE CIRCUIT DIAGRAM:

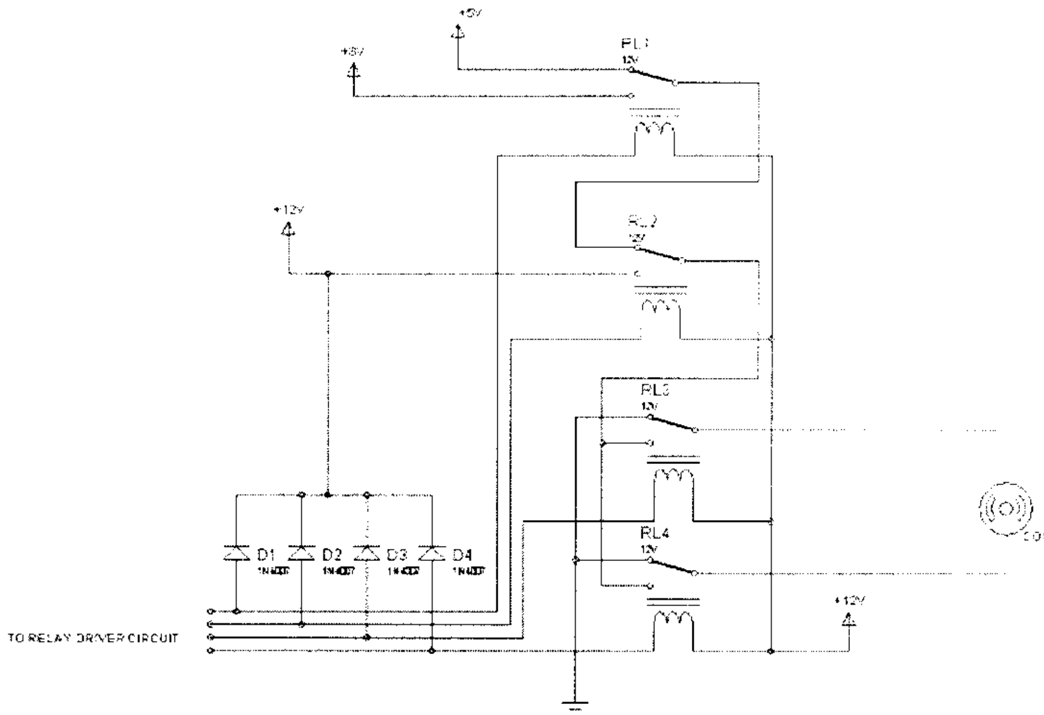
CONVEYOR SECTION:



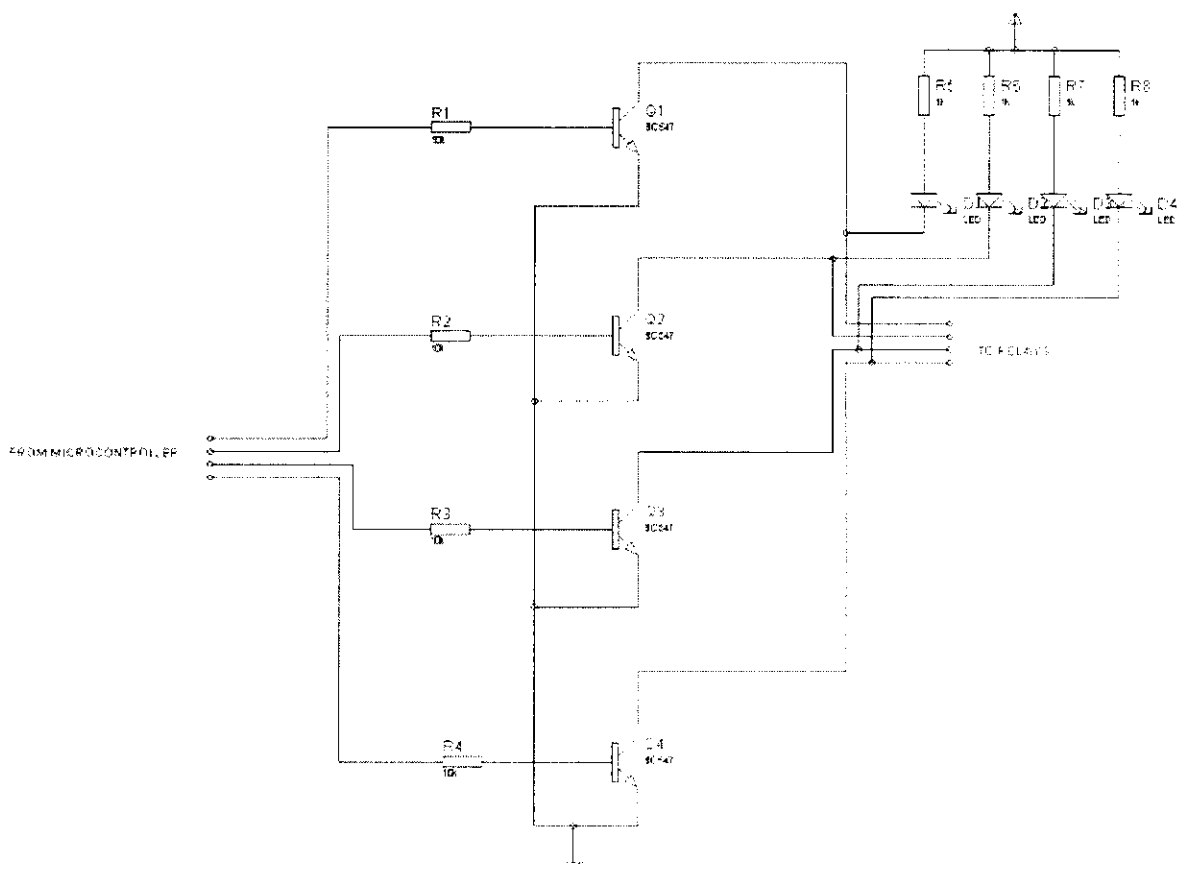
PC SECTION:



RELAY:

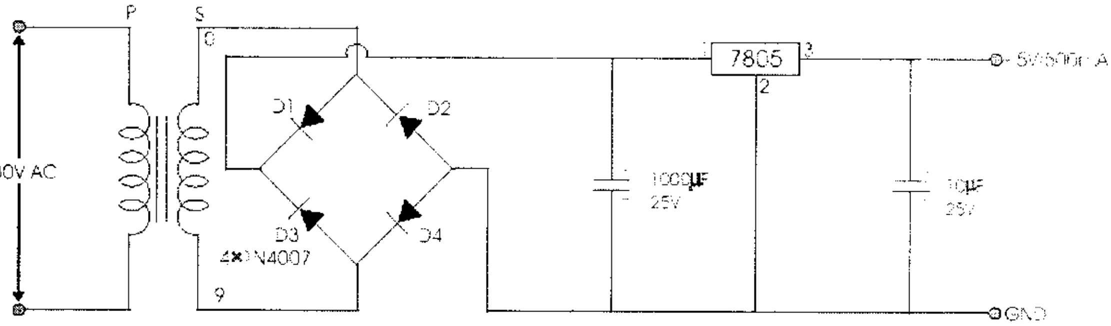


RELAY DRIVER:

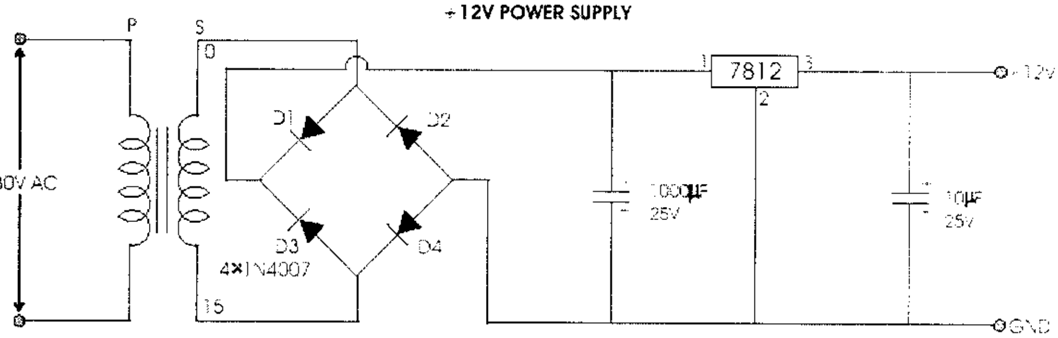


POWER SUPPLY:

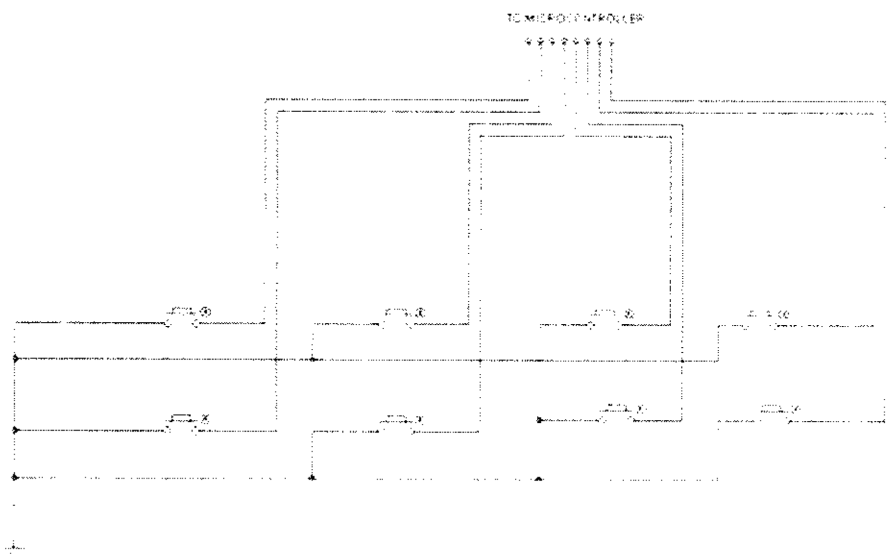
+5V POWER SUPPLY



+12V POWER SUPPLY



KEYPAD:



APPENDIX 2

MICROCHIP

PIC16F87XA

28/40/44-Pin Enhanced Flash Microcontrollers

Devices Included in this Data Sheet:

- PIC16F873A
- PIC16F874A
- PIC16F876A
- 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 I^{2}C™ (Master/Slave)
- Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) – 8 bits wide with external $\overline{RD}$, $\overline{WR}$ and $\overline{CS}$ controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

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 (V_{REF}) 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

Device	Program Memory		Data SRAM (Bytes)	EEPROM (Bytes)	I/O	10-bit A/D (ch)	CCP (PWM)	MSSP		USART	Timers 8/16-bit	Comparators
	Bytes	# Single Word Instructions						SPI	Master I^{2}C			
PIC16F873A	7.2K	4096	192	128	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F874A	7.2K	4096	192	128	33	8	2	Yes	Yes	Yes	2/1	2
PIC16F876A	14.3K	8192	368	256	22	5	2	Yes	Yes	Yes	2/1	2
PIC16F877A	14.3K	8192	368	256	33	8	2	Yes	Yes	Yes	2/1	2

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKI	13	14	30	32		ST/CMOS[4]	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.
$\overline{\text{MCLR}}$/VPP	1	2	18	18		ST	Master Clear (input) or programming voltage (output).
$\overline{\text{MCLR}}$					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/$\overline{\text{SS}}$/C2OUT	7	8	24	24		TTL	
RA5					I/O		Digital I/O.
AN4					I		Analog input 4.
$\overline{\text{SS}}$					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.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

Pin Name	PDIP Pin#	PLCC Pin#	TQFP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
							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^{2}C mode.
RC4/SDI/SDA	23	25	42	42		ST	
RC4					I/O		Digital I/O.
SDI					I		SPI data in.
SDA					I/O		I^{2}C 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.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

TABLE 1-3: PIC16F874A/877A PINOUT DESCRIPTION (CONTINUED)

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

3.0 DATA EEPROM AND FLASH PROGRAM MEMORY

The data EEPROM and Flash program memory is readable and writable during normal operation (over the full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers. There are six SFRs used to read and write this memory:

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

When interfacing to the data memory block, EEDATA holds the 8-bit data for read/write and EEADR holds the address of the EEPROM location being accessed. These devices have 128 or 256 bytes of data EEPROM (depending on the device), with an address range from 00h to FFh. On devices with 128 bytes, addresses from 80h to FFh are unimplemented and will wraparound to the beginning of data EEPROM memory. When writing to unimplemented locations, the on-chip charge pump will be turned off.

When interfacing the program memory block, the EEDATA and EEDATH registers form a two-byte word that holds the 14-bit data for read/write and the EEADR and EEADRH registers form a two-byte word that holds the 13-bit address of the program memory location being accessed. These devices have 4 or 8K words of program Flash, with an address range from 0000h to 0FFFh for the PIC16F873A/874A and 0000h to 1FFFh for the PIC16F876A/877A. Addresses above the range of the respective device will wraparound to the beginning of program memory.

The EEPROM data memory allows single-byte read and write. The Flash program memory allows single-word reads and four-word block writes. Program memory write operations automatically perform an erase-before-write on blocks of four words. A byte write in data EEPROM memory automatically erases the location and writes the new data (erase-before-write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump, rated to operate over the voltage range of the device for byte or word operations.

When the device is code-protected, the CPU may continue to read and write the data EEPROM memory. Depending on the settings of the write-protect bits, the device may or may not be able to write certain blocks of the program memory; however, reads of the program memory are allowed. When code-protected, the device programmer can no longer access data or program memory; this does NOT inhibit internal reads or writes.

3.1 EEADR and EEADRH

The EEADRH:EEADR register pair can address up to a maximum of 256 bytes of data EEPROM or up to a maximum of 8K words of program EEPROM. When selecting a data address value, only the LSByte of the address is written to the EEADR register. When selecting a program address value, the MSByte of the address is written to the EEADRH register and the LSByte is written to the EEADR register.

If the device contains less memory than the full address reach of the address register pair, the Most Significant bits of the registers are not implemented. For example, if the device has 128 bytes of data EEPROM, the Most Significant bit of EEADR is not implemented on access to data EEPROM.

3.2 EECON1 and EECON2 Registers

EECON1 is the control register for memory accesses.

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

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

The WREN bit, when set, will allow a write or erase operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write (or erase) operation is interrupted by a $\overline{MCLR}$ or a WDT Time-out Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit and rewrite the location. The data and address will be unchanged in the EEDATA and EEADR registers.

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

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

Note:	The self-programming mechanism for Flash program memory has been changed. On previous PIC16F87X devices, Flash programming was done in single-word erase/write cycles. The newer PIC18F87XA devices use a four-word erase/write cycle. See **Section 3.6 "Writing to Flash Program Memory"** for more information.

4.0 I/O PORTS

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

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

4.1 PORTA and the TRISA Register

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

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

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

Other PORTA pins are multiplexed with analog inputs and the analog VREF input for both the A/D converters and the comparators. The operation of each pin is selected by clearing/setting the appropriate control bits in the ADCON1 and/or CMCON registers.

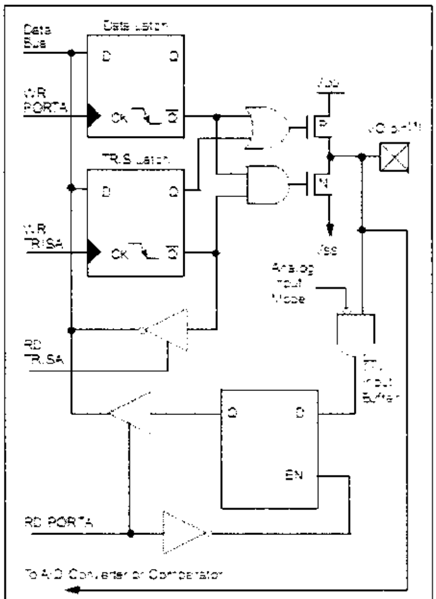
Note:	On a Power-on Reset, these pins are configured as analog inputs and read as '0'. The comparators are in the off (digital) state.

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

EXAMPLE 4-1: INITIALIZING PORTA

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

FIGURE 4-1: BLOCK DIAGRAM OF RA3:RA0 PINS



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

7.0 TIMER2 MODULE

Timer2 is an 8-bit timer with a prescaler and a postscaler. It can be used as the PWM time base for the PWM mode of the CCP module(s). The TMR2 register is readable and writable and is cleared on any device Reset.

The input clock (FOSC/4) has a prescale option of 1:1, 1:4 or 1:16, selected by control bits T2CKPS1:T2CKPS0 (T2CON<1:0>).

The Timer2 module has an 8-bit period register, PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is initialized to FFh upon Reset.

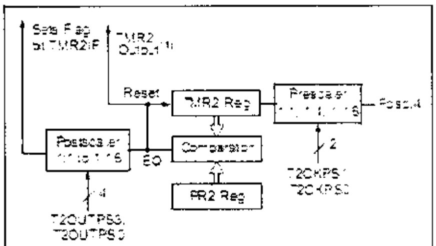
The match output of TMR2 goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling inclusive) to generate a TMR2 interrupt (latched in flag bit, TMR2IF (PIR1<1>)).

Timer2 can be shut-off by clearing control bit, TMR2ON (T2CON<2>), to minimize power consumption.

Register 7-1 shows the Timer2 Control register.

Additional information on timer modules is available in the PICmicro® Mid-Range MCU Family Reference Manual (DS33023).

FIGURE 7-1: TIMER2 BLOCK DIAGRAM



Note 1: TMR2 register output can be software selected by the SSP module as a baud clock.

REGISTER 7-1: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

bit 7 **Unimplemented:** Read as '0'

bit 6-3 **TOUTPS3:TOUTPS0:** Timer2 Output Postscale Select bits
0000 = 1:1 postscale
0001 = 1:2 postscale
0010 = 1:3 postscale
•
•
•
1111 = 1:16 postscale

bit 2 **TMR2ON:** Timer2 On bit
1 = Timer2 is on
0 = Timer2 is off

bit 1-0 **T2CKPS1:T2CKPS0:** Timer2 Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

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

CHAPTER 11

BIBLIOGRAPHY

- Electronics for u (technical reference manual).
- Chuck Hellebuyck (2008), ‘ Beginner’s Guide to Embedded C Programming using the PIC microcontroller and the HITECH PICC- Lite C Compiler’
- “Basic Microcontrollers” by Jan Axelon 1994
- http://www.artist-embedded.org/artist/
- www.discovercircuit.com
- www.google.com
- www.microchip.com