P- 3385

SMART BILLING USING RFID

By

V.ARAVINDHAN Reg No: 0710107007

A.INFANT AUGUSTINE Reg No: 0710107036

N.KARTHICK SRINIVASAN Reg No: 0710107041

S.KARTHIK Reg No: 0710107043

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

This is to be certified that this project report "**SMART BILLING USING RFID**" is the bonafide work of "**V.ARAVINDHAN, A.INFANT AUGUSTINE, N.KARTHICK SRINIVASAN, S.KARTHIK**" who carried out the project work under my supervision.

SIGNATURE

Dr. (Mrs.) Rajeswari Mariappan

Head of the Department

Department of Electronics and Communication Engineering,
Kumaraguru College of Technology,
Coimbatore.

SIGNATURE

Ms. R. Latha

Project Guide

Department of Electronics and Communication Engineering,
Kumaraguru College of Technology,
Coimbatore.

The candidates with University Register Numbers **0710107007, 0710107036, 0710107041, 0710107043** are examined by us in the Project Viva Voce Examination held on 18-04-2011.

Internal Examiner

External Examiner

ABSTRACT

The main objective of this project is to perform automation on bills, reduce time consumption and enhance customer satisfaction in supermarkets and stores.

In the present world, where people are tightly packed with works they don't have time to wait in a long queue to purchase the grocery items and to wait for billing. Now-a-days, we don't have peoples who are ready to work for the salary less than 3k. In such cases, this project will be useful for doing the billing process automatically.

This project can be divided into **3** sections namely product section, trolley section and the billing counter section. The product section comprises of the RFID tag in the form of the sticker, which is sticked over all the products.

The trolley section comprises of the LCD module, Microcontroller, RFID Reader and Zigbee. The read card value is passed to the Zigbee via the RFID Reader. Embedded coding is fed into the microcontroller .The LCD module is used for display of the count value and the total price of the products in the trolley.

The billing section consists of a Zigbee and a PC. The Zigbee receives the value send by Zigbee in the trolley section. The PC finds a match for the value and returns back the price to Zigbee.

ACKNOWLEDGEMENT

We are extremely grateful to **Dr.J.Shanmugam, Ph.D., Director,** Kumaraguru College of Technology for having given us a golden opportunity to embark on this project.

We are extremely grateful to **Dr.Ramachandran, M.Tech., Ph.D., Principal**, Kumaraguru College of Technology for having given us a golden opportunity to embark on this project.

We are deeply obliged to **Dr.Rajeswari Mariappan, M.E., Ph.D., B.Tech.Ed., F.I.E, MISTE, Head of the Department** of Electronics& Communication for her concern and implication during the project course.

We extend our heartfelt thanks to our project coordinator **Ms.V.Jeyasri Arokiamary, M.E., Associate Professor** and our respectable Guide **Ms.R.Latha, M.E., Associate Professor** Department of Electronics and Communication for their helpful guidance and valuable support given to us throughout this project.

Our thanks to all **Teaching and non-teaching staffs** of our department for providing us the technical support for our project.

We also thank **our college Management, friends and family** who helped us to complete this project with flying colours.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER1

INTRODUCTION

1.1 MOTIVATION

In the present world, where people are tightly packed with works they don't have time to wait in a long queue to purchase the grocery items and to wait for billing. Now-a-days, people are not ready to work for the salary less than 3k. This project is to overcome the time consumption and labour power.

RFID is used instead of barcode because of its large range and easy accessibility. RFID tag is provided in each product as a sticker and RFID reader is placed in the trolley. Whenever the customer drops an item inside the trolley, the Reader captures tag value of the product and it is sent through Zigbee.

In the billing section (receiver), the Zigbee receives the tag value and searches for a match in the PC. The PC has the database created through Microsoft access. Then the price of the product is passed to the microcontroller in the trolley via Zigbee in the transmitter and receiver section and it is displayed in the LCD present in the trolley. The PC maintains separate bill for each trolley.

The Microcontroller is ATMEL AT89S52 is used, because of its high range reprogrammable flash memory, low power consumption. The customer can purchase the items by saying his trolley number itself. A default tag is hung near the reader to alert the PC that a product is to be removed; the count value of products is displayed automatically in the LCD as the product is being dropped into the trolley. This project finds a key role in shopping for supermarkets and huge departmental stores.

1.2 OVERVIEW

The block diagram of this project comprises of two major sections namely,

1. Transmitter section,

2. Receiver section

1.3. TRANSMITTER SECTION

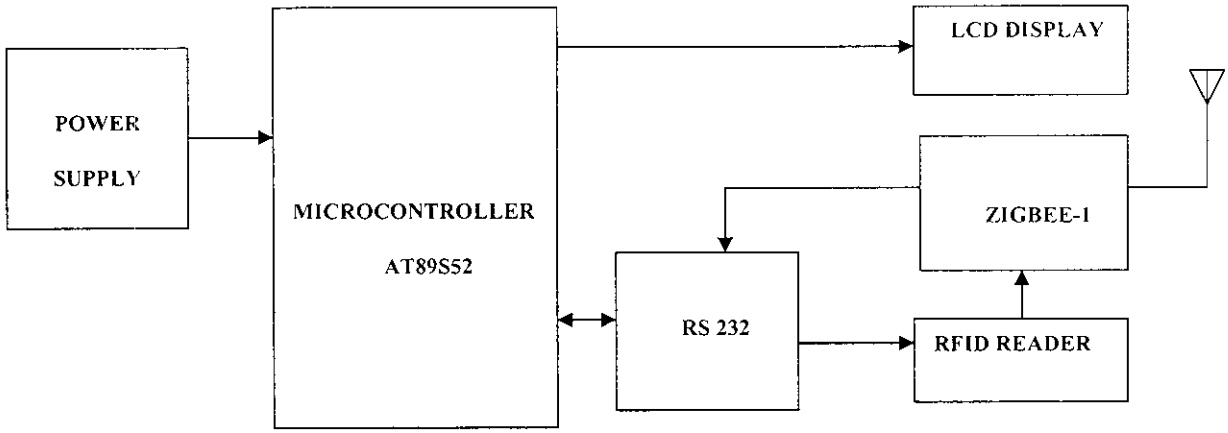


Fig 1.1 Block Diagram of Transmitter Section

1.4. RECEIVER SECTION

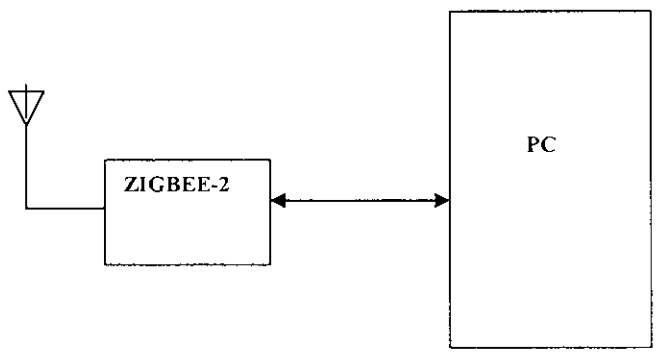


Fig 1.2 Block Diagram of Receiver Section

1.5. DESCRIPTION OF TRANSMITTER SECTION

The Transmitter Section of Smart billing machine is shown in fig 1.1.The **power supply** is the important unit required in the circuit as each and every block requires +**5V**. The power supply unit consists of a transformer, rectifiers, filters, regulators and optional filters.

Microcontroller employed in the Transmitter Section is **AT89S52.** It is known for its **in-system reprogrammable memory** and its **endurance of 1000 write/erase cycles.** It is a very powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

RS-232 is an important part in interfacing a peripheral device with microcontroller. The output of the microcontroller is usually in TTL format but the peripheral devices require a RS-232 input. Hence RS-232 serves the part of serial communication.

RFID Reader unit comprises of an in-built power supply with a serial port and a **PIC controller** to produce serial and Wigand output. The reader module supports the Wigand standard that gives the **Wigand encoded output**. This output comprises 3 bytes of data. It will be indicated as low pulse on data line, if it is a Data 1 signal and low pulse on the zero line, if it is a Data 0 signal.

Zigbee acts as a transceiver; it can send or receive a data byte. Here the Zigbee consists of serial receive and serial transmit buffer. Zigbee is a wireless network standard comes under **Personal Area Networks (PAN).**

1.6. DESCRIPTION OF RECEIVER SECTION

The Receiver Section of smart billing machine is shown in fig 1.2The receiver section also consists of **Zigbee**; its serial port is directly connected to the COM port of the PC.

PC section must have the Operating System of **Windows XP, Microsoft office tools** and **Visual Studio** being installed.

1.7. OVERALL CIRCUIT DIAGRAM

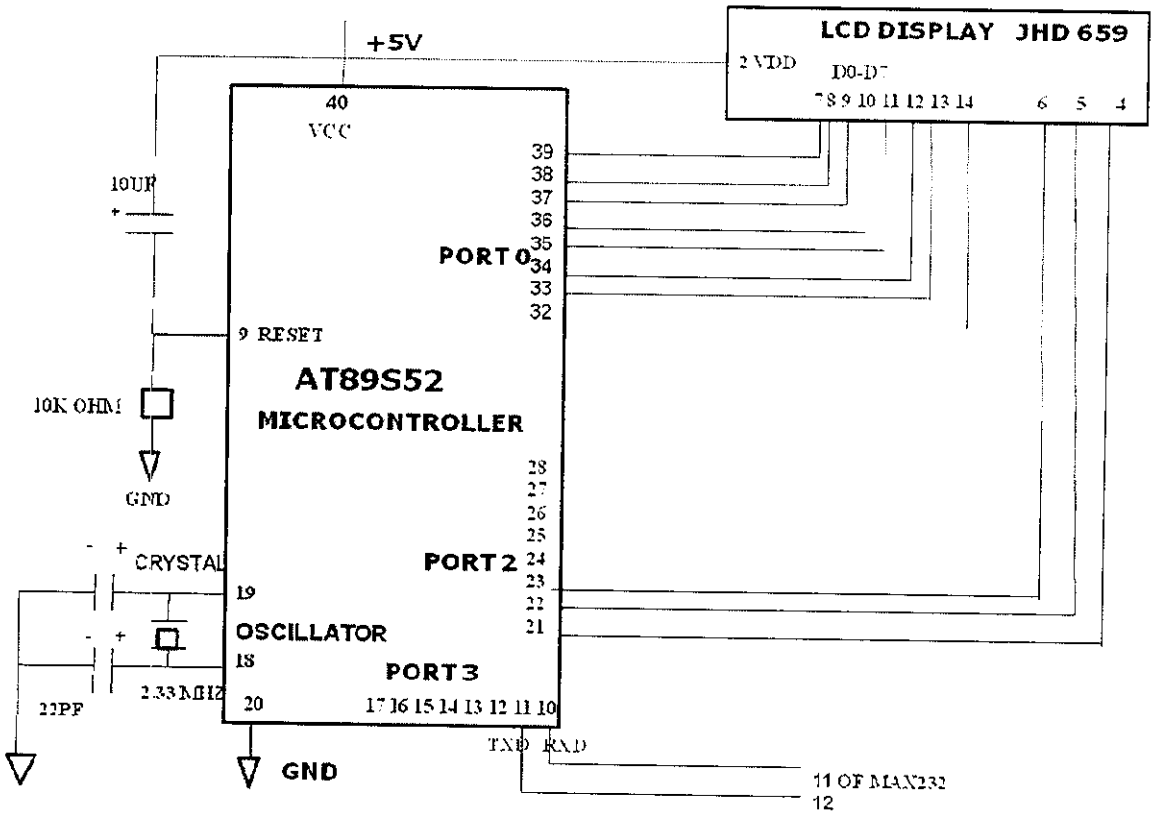


Fig 1.3 Circuit Diagram- Microcontroller and LCD (Transmitter Section-1)

DESCRIPTION

The Microcontroller and LCD display interface circuit diagram is shown in fig1.3. The pins of Microcontroller from 32-39 constitute the data lines for the LCD display and the lines 21-23 used as control for LCD section.

The RFID reader and zigbee1 UART is interfaced to Microcontroller through MAX 232 as shown in fig1.4. The TXD pin of RFID reader is connected to RXD pin of Zigbee1, the TXD pin of Zigbee1 is connected is connected to RXD pin of 9 pin adaptor and its TXD pin is connected to the RXD pin of RFID reader.

The fig1.5 shows the interface between the PC and Zigbee2. The TXD pin of Zigbee2 is connected to the PC's RXD pin and the TXD pin of PC is connected to the RXD pin of Zigbee2.

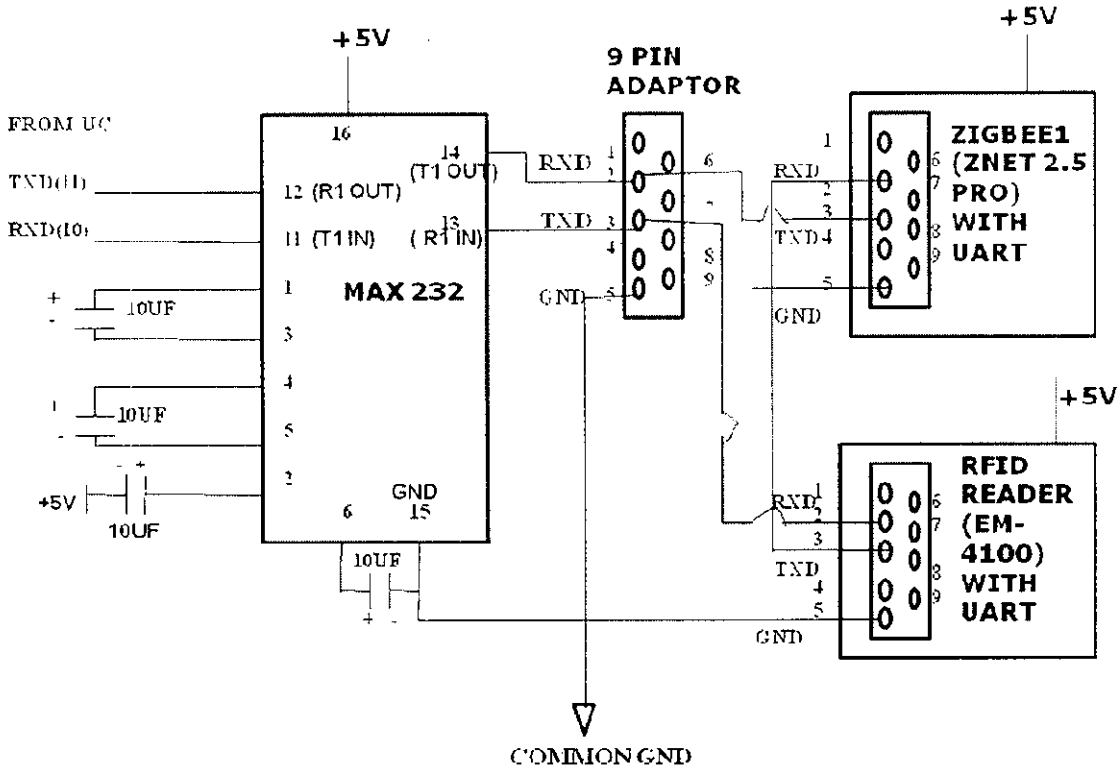


Fig 1.4 Circuit Diagram- MAX-232, RFID Reader and Zigbee1 (transmitter section-2)

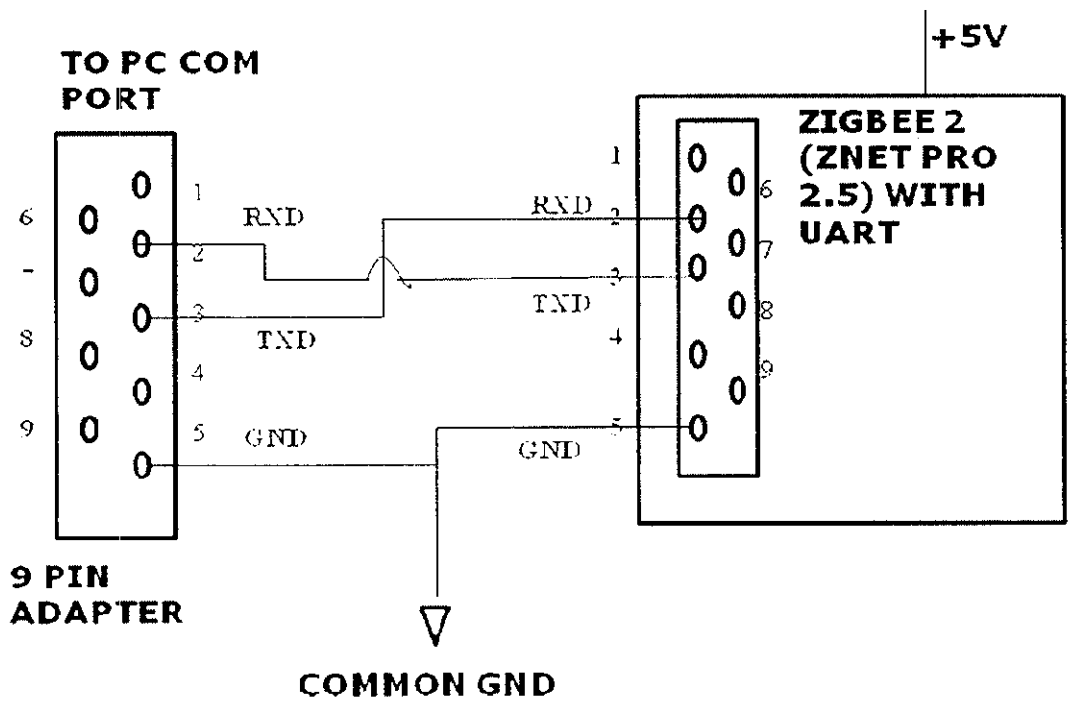


Fig 1.5 Circuit Diagram Zigbee2 and Pc (receiver section)

CHAPTER 2

MICROCONTROLLER

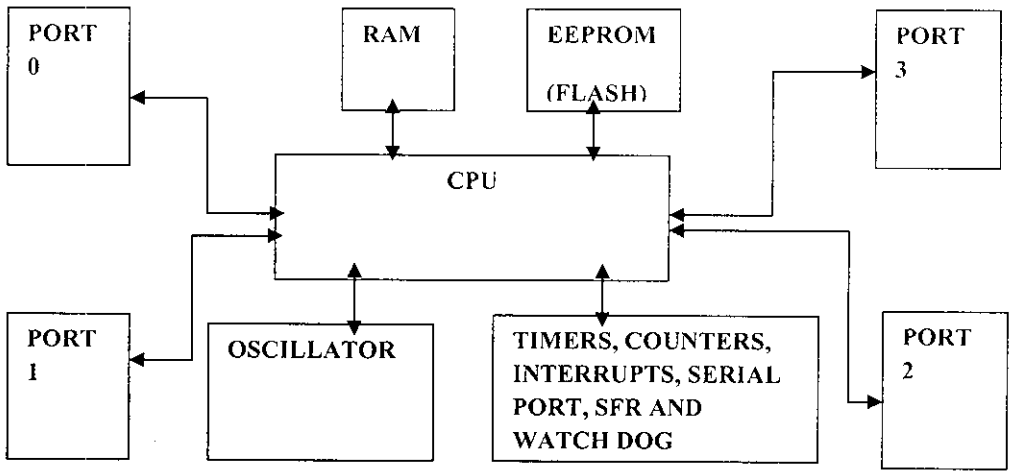


Fig 2.1 Block Diagram of Microcontroller

2.1 FEATURES

The Microcontroller used here in this project is **AT89S52**. It is Compatible with MCS-51™ Products and also has 8K Bytes of In-System Reprogrammable Flash Memory. The Endurance is of 1,000 Write/Erase Cycles; its Full Static Operation is from **0 Hz** to **24 MHz.** It has 256 x 8-bit Internal RAM with **32 Programmable I/O Lines**. It has Eight Interrupt Sources and Programmable Serial Channel.

2.2 DESCRIPTION

The **AT89C52** is a low-power, high-performance CMOS 8-bit microcomputer with **8K bytes of Flash programmable** and **erasable read only memory (PEROM)**. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional non-volatile memory programmer. AT89C52 is a powerful microcomputer which provides a **highly-flexible and cost-effective** solution to many embedded control applications.

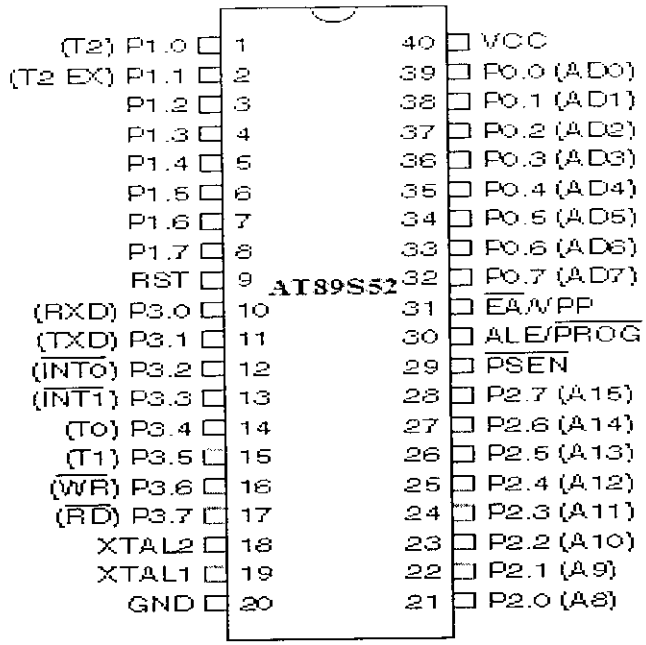


Fig 2.2 Pin Diagram of AT89S52 Microcontroller

2.3 PIN DESCRIPITION WITH PIN NUMBERS IN BRACES

With reference to the above pin diagram of AT89S52 in the **fig 3.2,** its description is given below

1. VCC (40)

It constitutes the Supply voltage. The voltage required is **+5V.**

2. GND (20)

This pin is the zero potential (**ground**).

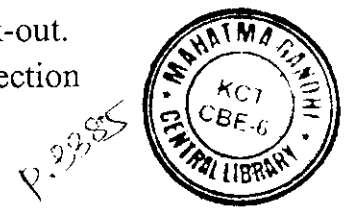
3. PORT 0(39-32)

Port 0 is an **8-bit open drain bi-directional I/O port**. As an Output port, each pin can sink eight TTL inputs. When 1's are written to port 0 pins, the pins can be used as high impedance inputs. Port 0 can also be configured to be the multiplexed low order address/ data bus during accesses to external programmable data memory. In this mode, P0 has internal pull-ups. Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. **External pull-ups** are required during program verification.

4. PORT 1(1-8)

Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups. Port 1 also receives the low-order address bytes during Flash programming and verification.

P1.0 T2 - External count input to Timer/ Counter 2, Clock-out.
P1.1 T2EX - Timer/Counter 2 capture/reload trigger and Direction Control



5. PORT 2 (21-28)

Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (IIL) because of the internal pull-ups.

In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that uses 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order address bits and some Control signals during Flash programming and verification.

6. PORT 3 (10-17)

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (IIL) because of the pull-ups.
Port 3 also receives some control signals for Flash programming and verification.

P3.0 - RXD (serial input port)
P3.1 - TXD (serial output port)
P3.2 - INT0 (external interrupt 0)
P3.3 - INT1 (external interrupt 1)
P3.4 - T0 (timer 0 external input)
P3.5 - T1 (timer 1 external input)
P3.6 - WR (external data memory write strobe)
P3.7 - RD (external data memory read strobe)

7. RESET (9)

Reset input. A high on this pin for two machine cycles while The oscillator is running resets the device.

8. ALE/PROG (30)

Address Latch Enable is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory. If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high.

9. PSEN (29)

Program Store Enable is the read strobe to external program Memory. When the AT89C52 is executing code from external program memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to external data memory.

10. EA/VPP (31)

External Access Enable. EA must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH.

Note: However, that if lock bit 1 is programmed, EA will be internally latched on reset. EA should be strapped to VCC for internal program executions. This pin also receives the 5-volt programming enable voltage (VPP) during Flash programming when 5-volt programming is selected.

11. XTAL1 (19)

Input to the inverting oscillator amplifier and input to the Internal clock operating circuit.

12. XTAL2 (18)

Output from the inverting oscillator amplifier.

2.4 OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed. Oscillator diagram is shown in figure

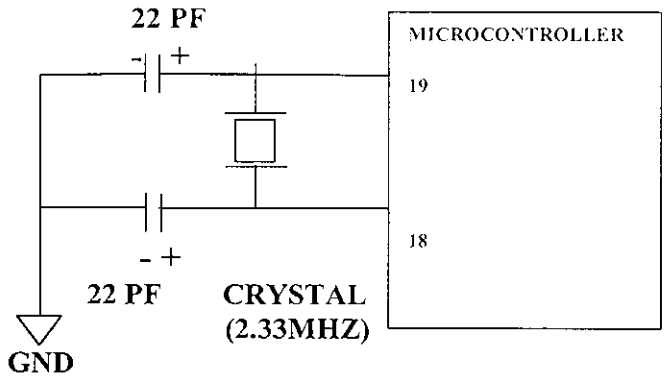


Fig 2.3 Figure of an Oscillator

2.5 MODES OF OPERATION

There are two modes of operation namely,

1. Idle mode
2. Power down mode

2.5. 1 Idle Mode

In idle mode, the CPU puts itself to sleep while all the on chip Peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset. Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control.

2.5. 2 Power-down Mode

In the power-down mode, the oscillator is stopped, and the Instruction that invokes power-down is the last instruction executed. The on-chip RAM and

Special Function Registers. Retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before VCC.

2.6 RESET FUNCTION

The reset is used to start-up or to restart the 80C51 microcontroller activities. It forces the 8051 in a known state by reinitializing all the internal registers needed to properly start the program execution. The reset must be kept active until all three of the following conditions are respected:

- The power supply must be in the specified range.
- The oscillator must reach a minimum oscillation level to ensure a good noise to signal ratio and a correct internal duty cycle generation.
- The reset pulse width duration must be at least two machine cycles. If one of the conditions is not respected the microcontroller will not start up properly.

2.7 DATA MEMORY

The AT89C52 implements 256 bytes of on-chip RAM. The upper 128 bytes occupy a parallel address space to the Special Function Registers. That means the upper 128 bytes have the same addresses as the SFR space but are physically separate from SFR space. When an instruction accesses an internal location above address 7FH, the address mode used in the instruction specifies whether the CPU accesses the upper 128 bytes of RAM or the SFR space.

Instructions that use direct addressing access SFR space. For example, the following direct addressing instruction accesses the SFR at location 0A0H (which is P2). Instructions that use indirect addressing access the upper 128 bytes of RAM. For example, the following indirect Addressing instruction, where R0 contains 0A0H, accesses .The data byte at address 0A0H, rather than P2 (whose Address is 0A0H).

2.8 UART (UNIVERSAL ASYNCHRONOUS RECEIVER AND TRANSMITTER)- SERIAL PORT

One of the microcontroller features making it so powerful is an integrated UART, better known as a serial port. It is a full-duplex port, thus being able to transmit and receive data simultaneously and at different baud rates. Without it, serial data send and receive would be an enormously complicated part of the program in which the pin state is constantly changed and checked at regular intervals.

When using UART, all the programmer has to do is to simply select serial port mode and baud rate. When it's done, serial data transmit is nothing but writing to the SBUF register, while data receive represents reading the same register. The microcontroller takes care of not making any error during data transmission. SBUF register format is shown in fig 2.4.

X	X	X	X	X	X	X	X
BIT 7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

Fig 2.4 Frame format of SBUF (Serial buffer)

Serial port must be configured prior to being used. In other words, it is necessary to determine how many bits is contained in one serial "word", baud rate and synchronization clock source. The whole process is in control of the bits of the SCON register (Serial Control) as shown in fig 2.5.

MSB **LSB**

SM0	SMI	SM2	REN	TB8	RB8	TI	RI

Fig 2.5 Frame format of SCON (Serial control)

SM0 - serial port mode it bit 0.
SM1 - serial port mode it bit 1.
SM2 - The multiprocessor mode.
REN - receive enable; it is made 1 to imply that device is

ready to receive.

TB8 - Used for optional transmit parity.

RB8 - used for optional receive parity.

TI - Transmit Interrupt Flag
Set when Byte in SBUF is completely transmitted.

RI - Receive Interrupt Flag
Set when a valid byte is received into SBUF.

2.9 TIMERS AND COUNTERS

TMOD selects the timers either T1 or T2. The first 4 bits belongs to Timer 1 and next 4 bits belongs to Timer 0. The frame format of TMOD is shown in fig 2.6.

MSB							LSB
GATE1	C/T1	T1M1	T1M0	GATE2	C/T0	T0M1	T0M0

Fig 2.6 Frame format of TMOD (Timer modifier)

GATE1 enables and disables Timer 1 by means of a signal brought the INT1 pin (P3.3):

- **1** - Timer 1 operates only if the INT1 bit is set.
- **0** - Timer 1 operates regardless of the logic state of the INT1 bit.

C/T1 selects pulses to be counted up by the timer/counter 1:

- **1** - Timer counts pulses brought to the T1 pin (P3.5).
- **0** - Timer counts pulses from internal oscillator.

T1M1, T1M0 These two bits select the operational mode of the Timer 1

T1M1	T1M0	Mode	Description
0	0	0	13-bit timer
0	1	1	16-bit timer
1	0	2	8-bit auto-reload
1	1	3	Split mode

Table 2.1 for different values of TIM1 &T1M0

GATE0 enables and disables Timer 1 using a signal brought to the INT0 pin (P3.2):

- **1** - Timer 0 operates only if the INT0 bit is set.
- **0** - Timer 0 operates regardless of the logic state of the INT0 bit.

C/T0 selects pulses to be counted up by the timer/counter 0:

- **1** - Timer counts pulses brought to the T0 pin (P3.4).
- **0** - Timer counts pulses from internal oscillator.

T0M1, T0M0 these two bits select the operational mode of the Timer0.

T0M1	T0M0	Mode	Description
0	0	0	13-bit timer
0	1	1	16-bit timer
1	0	2	8-bit auto-reload
1	1	3	Split mode

Table 2.2 for different values of T0M1 &T0M0

2.10 WATCHDOG TIMER

The Watchdog Timer is a timer connected to a completely separate RC oscillator within the microcontroller.

If the watchdog timer is enabled, every time it counts up to the program end, the microcontroller reset occurs and program execution starts from the first instruction. The point is to prevent this from happening by using a special command. The whole idea is based on the fact that every program is executed in several longer or shorter loops.

If instructions resetting the watchdog timer are set at the appropriate program locations, besides commands being regularly executed, then the operation of the watchdog timer will not affect the program execution.

If for any reason (usually electrical noise in industry), the program counter "gets stuck" at some memory location from which there is no return,

the watchdog will not be cleared, so the register's value being constantly incremented will reach the maximum limit and then reset occurs. The frame format of IE register is shown in fig 2.7.

EA	x	ET2	ES	ET1	EX1	ET0	EX0

MSB LSB

Fig 2.7 Frame format of IE Register (Interrupt Enable)

EA - global interrupt enable/disable:

0 - disables all interrupt requests.

1 - Enables all individual interrupt requests

ES - enables or disables serial interrupt:

0 - UART system cannot generate an interrupt.

1 - UART system enables an interrupt.

ET1 - bit enables or disables Timer 1 interrupt:

0 - Timer 1 cannot generate an interrupt.

1 - Timer 1 enables an interrupt.

EX1 - bit enables or disables external 1 interrupt:

0 - change of the pin INT0 logic state cannot generate an interrupt.

1 - Enables an external interrupt on the pin INT0 state change.

ET0 - bit enables or disables timer 0 interrupt:

0 - Timer 0 cannot generate an interrupt.

1 - Enables timer 0 interrupt.

EX0 - bit enables or disables external 0 interrupt:

- 0 - change of the INT1 pin logic state cannot generate an interrupt.

1 - Enables an external interrupt on the pin INT1 state change.

The AT89C52 has a total of **six** interrupt vectors:

1. **Two external interrupts (INT0 and INT1),**
2. **Three timer interrupts (Timers 0, 1, and 2) and**
3. **The serial port interrupts.**

CHAPTER 3

MAX 232

3.1 INTRODUCTION

The **MAX232** is a dual driver/receiver that includes a capacitive voltage generator to supply **TIA/EIA-232** voltage levels from a single 5V supply. Each receiver converts TIA/EIA-232-F inputs to 5V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V and can accept ±30V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232 levels.

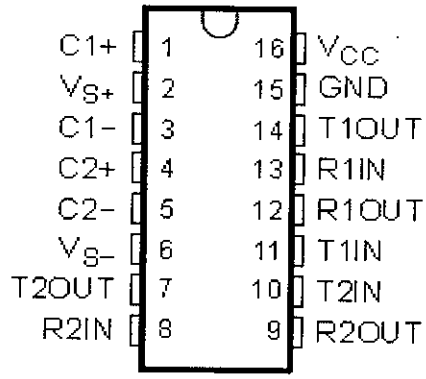


Fig 4.1 Pin Configuration of MAX 232

PIN NUMBER	DESCRIPTION
1	The positive terminal of capacitor C1 is connected.
2	A supply voltage of +5V
3	Negative terminal of C1
4	Positive terminal of C2
5	Negative terminal of C2
6	Negative supply of -5V
7	Driver 2 output(RS 232)
8	Receiver 2 input(RS 232)
9	Receiver 2 output(TTL)
10	Driver 2 input(TTL)

11	Driver 1 input(TTL)
12	Receiver 1 output(TTL)
13	Receiver 1 input(TTL)
14	Driver 1 output (RS 232)
15	ground
16	VCC (+5V)

Table 3.1 Pin Description of MAX 232

3.2 VOLTAGE LEVELS OF RS232

The RS-232 standard defines the voltage levels that correspond to logical one and logical zero levels. Valid signals are plus or minus 3 to 25 volts. The range near zero volts is not a valid RS-232 level; logic one is defined as a negative voltage, the signal condition is called marking, and has the functional significance of OFF. Logic zero is positive, the signal condition is spacing.

So a Logic Zero represented as **+3V to +25V** and Logic One represented as **-3V to -25V**. As shown in **fig 3.2** .

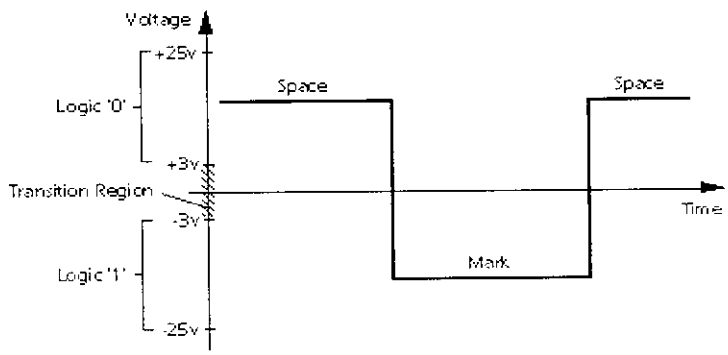


Fig 4.2 Figure of Voltage Levels of RS 232

3.3 RS 232 LEVEL CONVERTERS

Usually all the digital ICs work on TTL or CMOS voltage levels which cannot be used to communicate over RS 232 Protocol. So a Voltage or level converter is needed which can convert TTL to RS 232 and RS 232 to TTL Voltage levels. The most commonly used RS 232 level converter is MAX 232.

3.4 MAX 232 INTERFACING WITH MICROCONTROLLER, ZIGBEE AND RFID READER

To communicate over UART or USART, we just need three basic signals which are namely RXD (receive), TXD (transmit), GND (ground). So to interface MAX 232 with any Microcontroller (AVR, ARM, 8051, PIC etc.). We just need the basic signals. A simple schematic diagram of connections between a Microcontroller and MAX 232, Zigbee and RFID reader is shown in fig3.3.

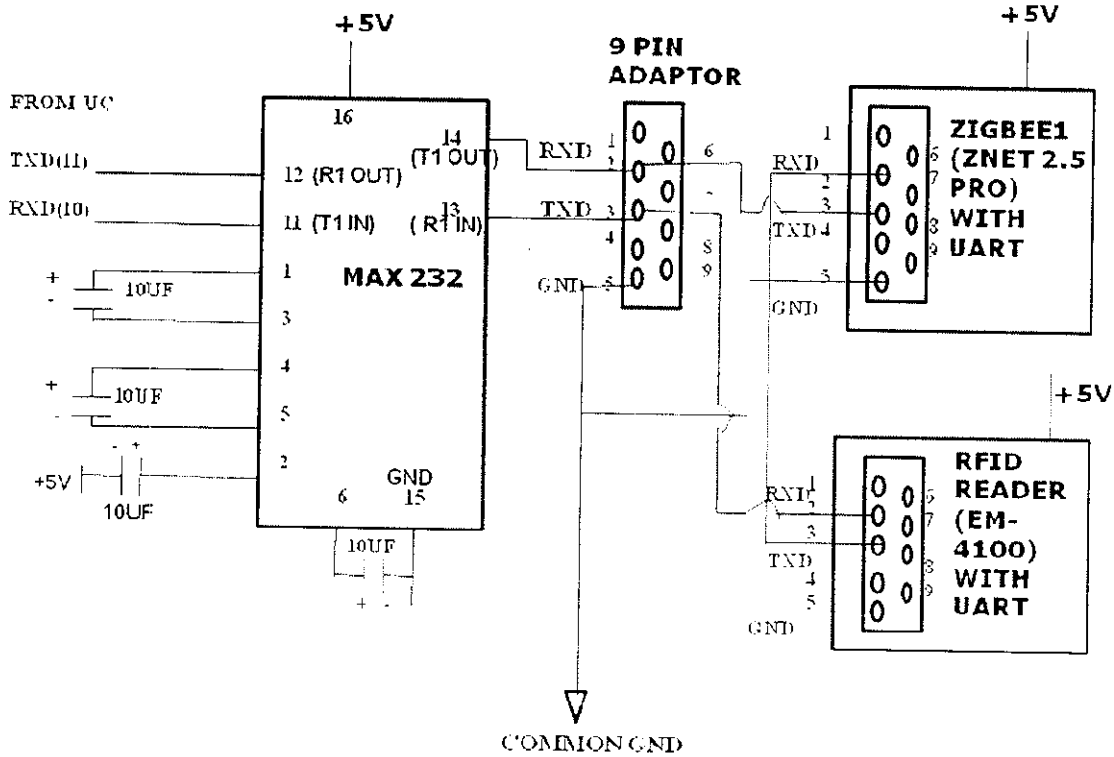


Fig 3.3 Interfacing Circuit of MAX 232 with Zigbee, Microcontroller and RFID Reader

3.5 RS232 TO TTL CONVERSION

- Burg Connecter for Supply Input and TTL O/p
- 9Pin D type male Connecter For Rs232 O/p

Description	Signal	9-pin DTE
Carrier Detect	CD	1
Receive Data	RD	2
Transmit Data	TD	3
Data Terminal Ready	DTR	4
Signal Ground	SG	5
Data Set Ready	DSR	6
Request to Send	RTS	7
Clear to Send	CTS	8
Ring Indicator	RI	9

Table 3.2 RS 232 with 9-pin Connector Pins

CHAPTER 4

LCD DISPLAY

4.1 INTRODUCTION

LCD's also are used as numerical indicators, especially in digital watches where their much smaller current needs than LED displays (microamperes compared with mill amperes) prolong battery life. Liquid crystals are organic (carbon) compounds, which exhibit both solid and liquid properties. 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 this project consists of 2 rows. Each row consists of maximum 16 characters. So using this display only maximum of 32 characters can be displayed. Fig4.1 shows Schematic Diagram of LCD Display.

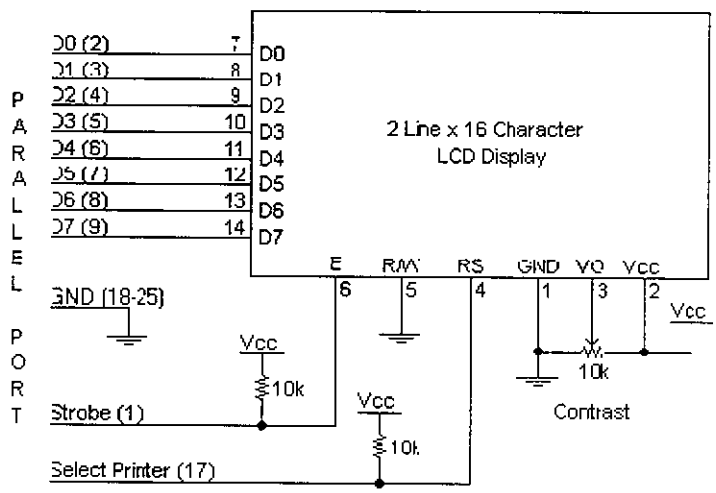


Fig 4.1 Schematic Diagram of LCD Display

4.2 CIRCUIT DESCRIPTION

- ✓ The LCD panel's **Enable (E)** of **pin (6)** and **Register Select (RS) of pin (4)** are connected to the Control Port **(port 2).**
- ✓ The Control Port is an open collector / open drain output. While all ports except **port 0** have internal pull-up resistors, hence we incorporate **the two 10K external pull up resistors**, the circuit is more portable for a

wider range of devices, some of which may have no internal pull up resistors.

- ✓ There is no need to make the Data bus into reverse direction. Hence hard wire the **R/W line pin (5)** of the LCD panel, into write mode. This will cause no bus conflicts on the data lines. As a result, read back the LCD's internal **Busy Flag** is not possible, which tells us if the LCD has accepted and finished processing the last instruction. This problem is overcome by inserting known delays into our program.
- ✓ The **10k Potentiometer** controls the **contrast** of the LCD panel through V0 of **pin (3). Pin (1) is ground.**
- ✓ The lines D0-D7 are called data lines they are connected with the port 0 through internal pull up resistors. They belong to the **pins (7-14) .pin (12) forms the supply VCC.** The schematic view of LCD display is shown in fig4.2.

4.3 FEATURES OF LCD DISPLAY

- ✓ 16 Characters x 2 Lines.
- ✓ 5 x 7 Dots with Cursor.
- ✓ Built in Controller.
- ✓ +5v Power Supply (Also Available for +3V).
- ✓ 1/16 Duty Circle.

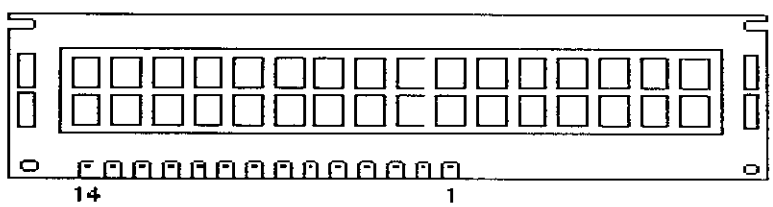


Fig 4.2 Schematic View of LCD Display

4.4 MAJOR STEPS IN LCD DISPLAY

Major task in LCD interfacing is the **initialization** sequence. In LCD initialization you have to send command bytes to LCD. Here you set the interface mode, display mode, address counter increment direction, set contrast of LCD, horizontal or vertical addressing mode, color format. This sequence is given in respective LCD driver datasheet. Studying the function set of LCD lets you know the definition of command bytes. It varies from one LCD to another. If you are able to initialize the LCD properly 90% of your job is done.

Next step after initialization is to send data bytes to required display data RAM memory location. Firstly set the address location using address set command byte and than send data bytes using the DDRAM write command. To address specific location in display data RAM one must have the knowledge of how the address counter is incremented.

4.5 COMMANDS AND INSTRUCTION SET

Only the instruction register (IR) and the data register (DR) of the LCD can be controlled by the MCU. Before starting the internal operation of the LCD, control information is temporarily stored into these registers to allow interfacing with various MCUs, which operate at different speeds, or various peripheral control devices. The internal operation of the LCD is determined by signals sent from the MCU. These signals, which include register selection, signal (RS), read/write signal (R/W), and the data bus (DB0 to DB7), The LCD commands are shown in table4.1.There are four categories of instructions that:

- Designate LCD functions, such as display format, data length, etc.
- Set internal RAM addresses
- Perform data transfer with internal RAM
- Perform miscellaneous functions

COMMANDS	D7	D6	D5	D4	D3	D2	D1	D0	HEXADECIMAL VALUE
CLEAR DISPLAY	0	0	0	0	0	0	0	1	01
DISPLAY & CURSOR HOME	0	0	0	0	0	0	1	X	02 or 03
CHARACTER ENTRY MODE	0	0	0	0	0	0	1/D	S	04 to 07
DISPLAY ON AND OFF	0	0	0	0	1	D	U	B	08 to 0F
DISPAY/ CURSOR SHIFT	0	0	0	1	D/C	R/L	X	X	10 to 1F
FUNCTION SET	0	0	1	8/4	2/1	10/7	X	X	20 to 3F
SET DDRAM ADDRESS	0	1	A	A	A	A	A	A	40 to 7F
SET DISPLAY ADDRESS	0	A	A	A	A	A	A	A	80 to FF

Table 5.1 LCD Commands Table

1/D - '**1**' corresponds to increment

'**0**' corresponds to decrement

S - '**1**' display on

'**0**' display off

U - '**1**' corresponds to cursor underline on

'**0**' cursor underline off

B -'**1**' corresponds to cursor blink on

'**0**' corresponds to cursor underline off

R/L -'**1**' corresponds to right shift

'**0**' corresponds to left shift

8/4 -'**1**' corresponds to 8bit interface

'**0**' corresponds to 4 bit interface

2/1 -'**1**' corresponds to 2 line mode.

'**0**' corresponds to 4 line mode.

10/7 -'**0**' corresponds to 5*10 dot frame format

'**1**' corresponds to 5*7 display format

CHAPTER 5

RFID READER

5.1 Introduction to RFID Technology

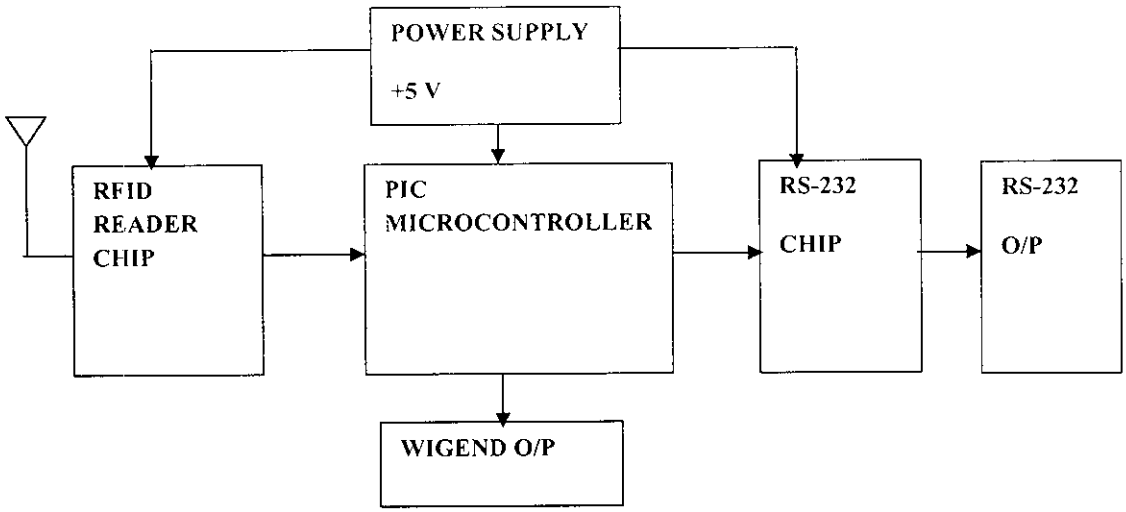


Fig 5.1 Block Diagram of RFID Reader

Radio-frequency identification (RFID) is a technology that uses communication via radio waves to exchange data between a reader and an electronic tag attached to an object, for the purpose of identification and tracking. Some tags can be read from several meters away and beyond the line of sight of the reader. The application of bulk reading enables an almost parallel reading of tags. Radio-frequency identification involves **interrogators** (also known as **readers**), and **tags** (also known as **labels**).Most RFID tags contain at least two parts. One is an integrated circuit for storing and processing information, modulating and demodulating a radio-frequency (RF) signal, and other specialized functions. The other is an antenna for receiving and transmitting the signal. The block diagram of RFID reader is shown in fig 5.1.

5.2 RFID TAG TYPES

There are basically three types of RFID tags based on their power source, they are namely

1. Passive tag

2. Active tags

3. Battery assisted passive

- Passive tags have no power source and require an external electromagnetic field to initiate a signal transmission.
- Active tags contain a battery and can transmit signals once an external source ('Interrogator') has been successfully identified.
- Battery assisted passive (BAP) RFID tags, have an external source to wake up but have significant higher forward link capability providing greater range.

5.3 IMPORTANCE OF RFID TECHNOLOGY IN THIS PROJECT

Long checkout lines at the grocery store are one of the biggest complaints about the shopping experience. Soon, these lines could disappear when the ubiquitous Universal Product Code (UPC) bar code is replaced by **smart labels**, also called **radio frequency identification** (RFID) tags. RFID tags are intelligent bar codes that can talk to a networked system to track every product that you put in your shopping cart.

5.4 RFID FREQUENCIES

Radio waves are the carriers of data between the reader and tags. The approach generally adopted for RFID communication is to allocate frequencies depending on application. The frequencies used cover a wide spectrum.

These specified bands are

Very long wave 9 - 135 kHz
Short wave 13.56 MHz
UHF 400-1200 MHz
Microwave 2.45 and 5.8 GHz

The allocation of frequencies is regulated by government agencies, requiring care in considering RFID applications in different countries. Efforts at standardization should avert these problems. The many varied applications will work their best at different frequencies. The most common uses of low

frequency systems are in security access, asset tracking and animal identification. They generally have short reading ranges and lower system costs. High-frequency systems are used for such applications as railroad car tracking and automated toll collection. The power level of the interrogator and the power available within the tag to respond will determine the reading range that can be achieved in an RFID system. Environmental conditions, particularly at the higher frequencies, can also influence the range of communication.

5.5 HARDWARE USED

The RFID Reader is has 3 stages

- RFID Reader Chip
- PIC Microcontroller to decode the data into Serial o/p and Wigand O/p
- Rs232 to convert signal into TTL to RS232

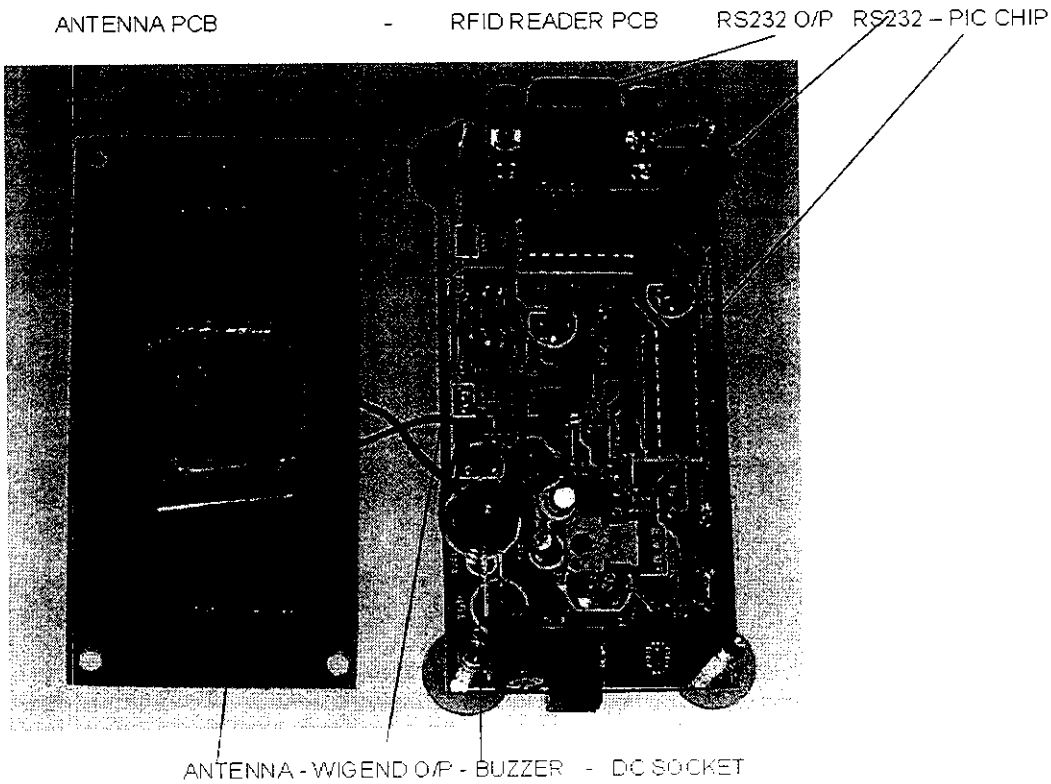


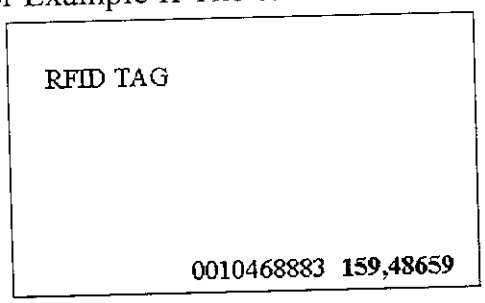
Fig 5.2 Schematic View of RFID Reader

The schematic view of the RFID reader used is shown in fig 5.2.

5.6 DATA TRANSMISSION IN ASCII STANDARD

Data read from the tag is Manchester encoded. The Manchester encoded data is decoded to ASCII standard. Decoded data is sent to the UART serial interface for wired communication with the host systems. ASCII data format is shown below:

For Example If The card Shown Above is placed on the Reader,



UART Data Will be 159 48659

Wigand O/p Will be 1 1001 1111 1011 1110 0001 0011 0

PARITY FACILITY CODE(8 BITS) CUSTOMER ID (16 BITS) PARITY

5.7 DATA TRANSMISSION IN WIGAND26 STANDARD

The reader module supports the Wigand standard that gives the Wigand encoded output. This output comprises 3 bytes of data. It will be indicated as low pulse on data line if it is a Data 1 signal and low pulse on the zero line if it is a Data 0 sign. This timing pattern falls within the Wigand guidelines as prescribed by the SIA's Access.

Control Standard Protocol for the 26-bit Wigand Reader Interface (a Pulse Width time between 20 μS and 100 μS, and a Pulse Interval time between 200 μSand 20 mS). The Wigand protocol diagram is shown in fig 5.3.

The Data 1 and Data 0 signals are held at logic high level (above the Voh level) until the reader is ready to send a data stream. The reader places the data

as asynchronous low-going (negative) pulses (below the Vol level) on the Data 1 or Data 0 lines to transmit the data stream to the access control panel (The Data 1 and Data 0 pulses do not overlap or occur simultaneously.

The composition of the open existing industry standard 26- bit Wigand format contains 8 bits for the facility code field and 16 bits for the ID number field. Mathematically these 8 facility code bits allow a total of 256 (0 to 255) facility codes, while the 16 ID number bits allow a total of only 65,536 (0 to 65,535) individual ID's within each facility code.

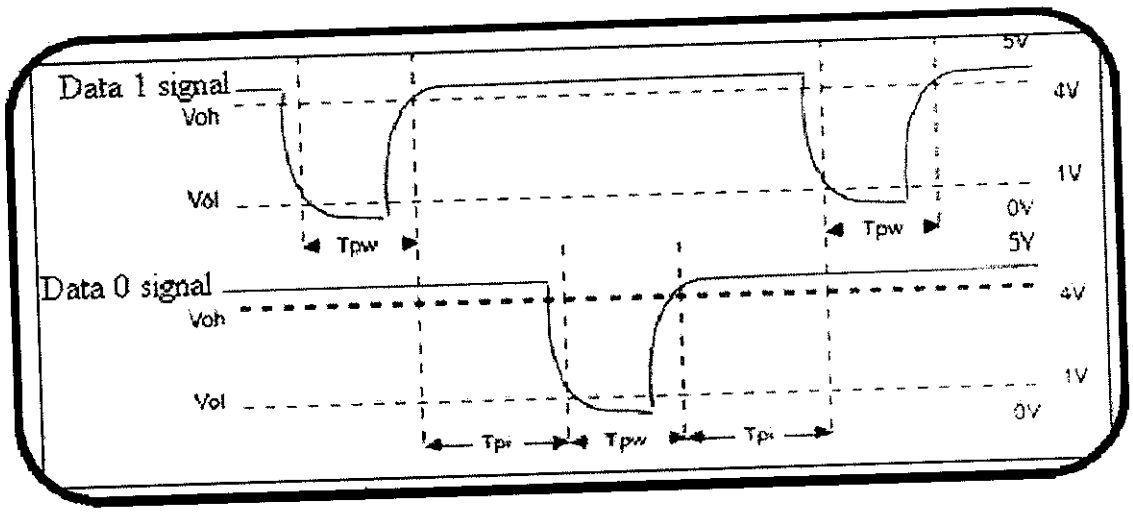


Fig 5.3 Wigand Protocol Diagram

CHAPTER 6

ZIGBEE

6.1 INTRODUCTION

Zigbee is a specification for a suite of high level communication protocols using small, low-power digital radios based on the **IEEE 802.15.4-2003** standard for **Low-Rate Wireless Personal Area Networks (LR-WPANs)**, such as wireless light switches with lamps, electrical meters with in-home-displays, consumer electronics equipment via short-range radio needing low rates of data transfer. The technology defined by the Zigbee specification is intended to be simpler and less expensive than other WPANs, such as Bluetooth. Zigbee is targeted at radio-frequency (RF) applications that require a low data rate, long battery life, and secure networking.

6.2 FEATURES

HIGH PERFORMANCE, LOW COST

Indoor/Urban : **up to 300' (100 m)**

Outdoor line-of-sight: **up to 1 mile (1.6 km)**

Transmit Power Output: **100 mW (20 dBm) EIRP**

Receiver Sensitivity: **-102 dBm**

RF Data Rate: **250,000 bps**

Advanced networking & security

Retries and Acknowledgements

DSSS (Direct Sequence Spread Spectrum) Each direct sequence channel has over **65,000** unique network addresses available

Point-to-point, point-to-multipoint and peer-to-peer topologies supported , Self-routing, self-healing and fault-tolerant mesh networking low.

6.3 ZIGBEE NETWORK FORMATION

Zigbee networks are called **personal area networks (PAN)**. Each network contains a 16-bit identifier called **a PAN ID**.

Zigbee defines three different device types –

1. Coordinator (**C**),
2. Router (**R**), and
3. End device (**E**).

The topology of the zigbee network is shown in fig 6.1.

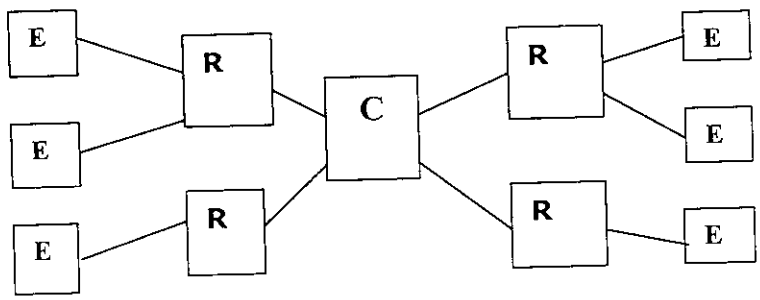


Fig 6.1 Zigbee Network Topology

6.3. 1 CO ORDINATOR (C)

Coordinator is responsible for selecting the channel and PAN ID. The coordinator starts a new PAN. Once it has started a PAN, the coordinator can allow routers and end devices to join the PAN. The coordinator can transmit and receive RF data transmissions, and it can assist in routing data through the mesh network. Coordinators are not intended to be battery-powered devices. Since the coordinator must be able to allow joins and/or route data, it should be mains powered.

6.3.2 ROUTER (R)

A **router** must join a Zigbee PAN before it can operate. After joining a PAN, the router can allow other routers and end devices to join the PAN. The router can also transmit and receive RF data transmissions, and it can route data packets through the network. Since routers can allow joins and participate in routing data, routers cannot sleep and should be mains powered.

6.3.3 END DEVICE

An **end device** must join a Zigbee PAN, similar to a router. The end device, however, cannot allow other devices to join the PAN, nor can it assist in routing data through the network. An end device can transmit or receive RF data transmissions. End devices are intended to be battery powered devices. Since the end device may sleep, the router or coordinator that allows the end device to join must collect all data packets intended for the end device, and buffer them until the end device wakes and is able to receive them. The router or coordinator that allowed the end device to join and that manages RF data on behalf of the end device is known as the end device's parent. The end device is considered a child of its parent. Table 7.1 shows the pin descriptions of XBEE.

PIN	NAME	DESCRIPTION
1	VCC	Power supply
2	DOUT	UART data out
3	DIN/CONFIG	UART data in
4	DIO12	Digital i/o 12
5	RESET	Module reset
6	PWM0	PWM Output 0
7	PWM	Digital i/o 11
8	RESERVED	No connection
9	DTR	Digital i/o 8
10	GND	Ground
11	DIO4	digital i/o 4
12	CTS	Clear to send
13	ON	Module status indicator
14	RESERVED	No connection
15	DIO5	Digital i/o 5
16	RTS	Request to send
17	AD3/DIO3	Analog input 3/ digital i/o 3

18	AD2/DIO2	Analog input 2/ digital i/o 2
19	AD1/DIO1	Analog input 1/ digital i/o 1
20	AD0/DIO0	Analog input 0/ digital i/o 0

Table 6.1 Pin Description of XBEE

6.4 SERIAL DATA

Data enters the module UART through the DIN (pin 3) as an asynchronous serial signal. The signal should idle high when no data is being transmitted. Each data byte consists of a start bit (low), 8 data bits (least significant bit first) and a stop bit (high). The following figure illustrates the serial bit pattern of data passing through the module. Example The module UART performs tasks, such as timing and parity checking, that are needed for data communications. Serial communications depend on the two UARTs to be configured with compatible settings (baud rate, parity, start bits, stop bits, data bits). The transmission of data (0x1F) is shown in fig 6.2.

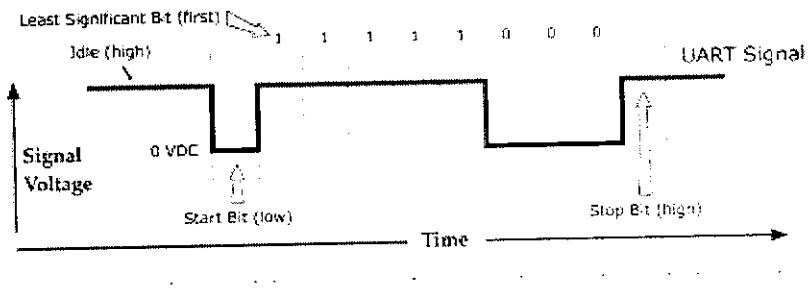


Fig 6.2 Figure showing Transmission of 0x1F (hexadecimal value)

Data Format is 8-N-1 (bits - parity - # of stop bits)

6.5 SERIAL BUFFERS

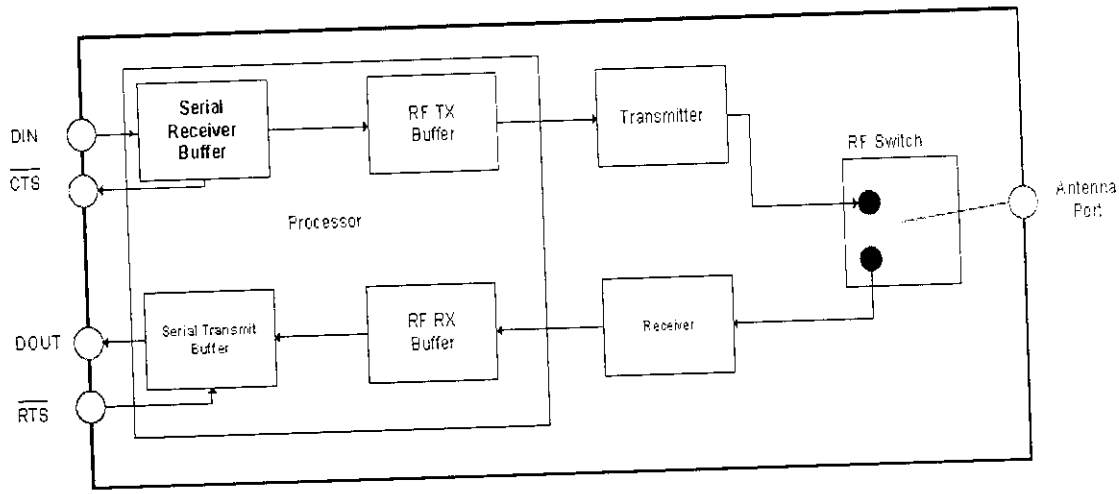


Fig 6.3 Figure of Serial Buffer

The serial buffer of Zigbee is shown in fig 6.3.

6.5.1 SERIAL RECEIVE BUFFER

When serial data enters the RF module through the DIN Pin (pin 3), the data is stored in the serial receive buffer until it can be processed. Under certain conditions, the module may not be able to process data in the serial receive buffer immediately. If large amounts of serial data are sent to the module, CTS flow control may be required to avoid overflowing the serial receive buffer.

Cases in which the serial receive buffer may become full and possibly overflow:

1. If the module is receiving a continuous stream of RF data, the data in the serial receive buffer will not be transmitted until the module is no longer receiving RF data.

2. If the module is transmitting an RF data packet, the module may need to discover the destination address or establish a route to the destination. After transmitting the data, the module may need to retransmit the data if an acknowledgment is not received, or if the transmission is a broadcast. These issues could delay the processing of data in the serial receive buffer.

6.5.2 SERIAL TRANSMIT BUFFER

When RF data is received, the data is moved into the serial transmit buffer and sent out the UART. If the serial transmit buffer becomes full enough such that all data in a received RF packet won't fit in the serial transmit buffer, the entire RF data packet is dropped.

Cases in which the serial transmit buffer may become full resulting in dropped RF packets

1. If the RF data rate is set higher than the interface data rate of the module, the module could receive data faster than it can send the data to the host.
2. If the host does not allow the module to transmit data out from the serial transmit buffer because of being held off by hardware flow control

6.6 OPERATING MODES

The operating modes of the Zigbee protocol are,

1. Idle mode
2. Transmit mode
3. Receive mode
4. Sleep mode
5. Command mode

6.6.1 IDLE MODE

When not receiving or transmitting data, the RF module is in Idle Mode. During Idle Mode, the RF module is also checking for valid RF data. The module shifts into the other modes of operation under the following conditions:

- Transmit Mode (Serial data in the serial receive buffer is ready to be packetized)
- Receive Mode (Valid RF data is received through the antenna)

- Sleep Mode (End Devices only)
- Command Mode (Command Mode Sequence is issued)

6.6.2. TRANSMIT MODE

When serial data is received and is ready for packetization, the RF module will exit Idle Mode and attempt to transmit the data. The destination address determines which node(s) will receive the data. Prior to transmitting the data, the module ensures that a 16-bit network address and route to the destination node have been established. If the destination 16-bit network address is not known, network address discovery will take place. If a route is not known, route discovery will take place for the purpose of establishing a route to the destination node. If a module with a matching network address is not discovered, the packet is discarded. The data will be transmitted once a route is established. If route discovery fails to establish a route, the packet will be discarded.

6.6.3. RECEIVE MODE

If a valid RF packet is received, the data is transferred to the serial transmit buffer.

6.6.4. SLEEP MODE

Sleep modes allow the RF module to enter states of low power consumption when not in use. The zigbee OEM RF modules support both pin sleep (sleep mode entered on pin transition) and cyclic sleep (module sleeps for a fixed time).

6.6.5. COMMAND MODE

To modify or read RF Module parameters, the module must first enter into Command Mode - a state in which incoming serial characters are interpreted as commands

Default AT Command Mode Sequence (for transition to Command Mode):

1. No characters sent for one second [GT (Guard Times) parameter = 0x3E8].

2. Input three plus characters ("+++") within one second [CC (Command Sequence Character) parameter = 0x2B.]

6.7 ZIGBEE DEVICE ADDRESSING

The 802.15.4 protocol upon which the Zigbee protocol is built specifies two address types

1.16-bit network addresses

2.64-bit Addresses

6.7.1 16-BIT NETWORK ADDRESSES

A 16-bit network address is assigned to a node when the node joins a network. The network address is unique to each node in the network. However, network addresses are not static - it can change. The following two conditions will cause a node to receive a new network address

1. If an end device cannot communicate with its parent it may need to leave the network and rejoin to find a new parent.

2. If the device type changes from router to end device, or vice-versa, the device will leave the network and rejoin as the new device type. Zigbee requires that data be sent to the 16-bit network address of the destination device. This requires that the 16-bit address be discovered before transmitting data.

6.7.2 64-BIT ADDRESSES

Each node contains a unique 64-bit address. The 64-bit address uniquely identifies a node and is permanent.

CHAPTER 7

POWER SUPPLY

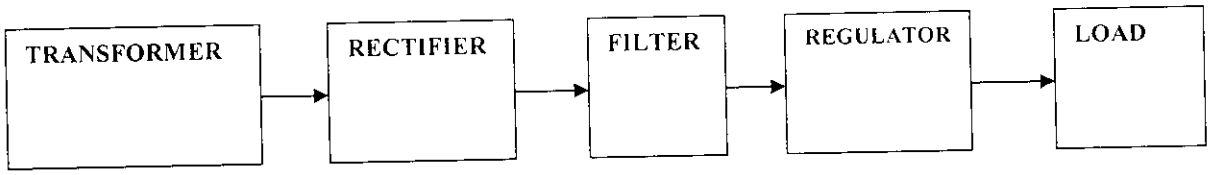


Fig 7.1 Block Diagram of Power Supply

7.1 INTRODUCTION

The block diagram of power supply is shown in fig7.1.Power supply circuits are built using Transformer, Filters, Rectifiers, and Voltage regulators. AC voltage given as an input to the transformer is converted to a steady dc voltage at the regulator output. The high AC voltage is converted to a lower level AC voltage using transformer. The low level AC voltage is converted to a DC voltage by the rectifier. The DC signal is then converted into a pure DC signal using filters. The DC voltage now obtained is not a steady voltage. Hence we employ a regulator to get a constant DC supply.

7.2 TRANSFORMER

The transformer employed here is 230V to 12V step down transformer. Transformer is a unit that converts a voltage level higher or lower than its original level. Usually step up transformers are very huge, they are used in power stations. Here require a lower voltage than its own input hence used a step down transformer.

Transformer comprises of a primary and secondary windings. The turn's ratio is the ratio of winding of secondary to the primary.

7.3 RECTIFIERS

Rectifier is a unit that converts a **AC signal into a DC signal**. The type of rectifier used here is a **bridge rectifier**. It is used because it has some advantages over the full wave rectifier.

Bridge rectifier consists of four diodes connected back to back, of which two diodes work in the positive half cycle and other two diodes works in negative half cycle. Hence we get a unidirectional DC voltage of both the half cycles.

Rectifier used here is a diode namely **IN 4007**. The diode consists of a positive terminal anode and a negative terminal cathode.

7.4 FILTERS

A **filter** is a unit that removes the unwanted AC ripples in the DC signal, thereby producing DC signals without ripples. The filters used popularly are,

1. Capacitor
2. Inductor
3. LC Filter

Of the above filters we use **capacitor** here. The capacitor here removes the ripples in DC voltage produced due to intervention of AC voltage. The capacitor used here is **1000 UF**. The use of the value of this capacitor depends on the input current. For a precautionary purpose we use a additional filter of **1UF** at the load end to save the power supply circuit from vibrations produced by the load.

7.5 REGULATORS

Regulator is a unit that produces a constant DC voltage. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. Although the internal construction of the IC is somewhat different from that described for discrete voltage regulator circuits, the external operation is much the same. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustably set voltage.

Regulator used is **IC 7805**. The 78XX series comes under the class of regulators. Here the requirement is 5V hence IC 7805 is used. The

representation of 7805 IC regulator and the detailed circuit diagram of power supply are shown in fig7.2 and fig7.3 respectively.

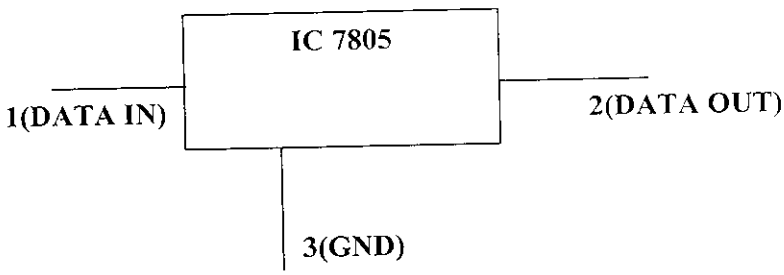


Fig 7.2 Representation of Regulator

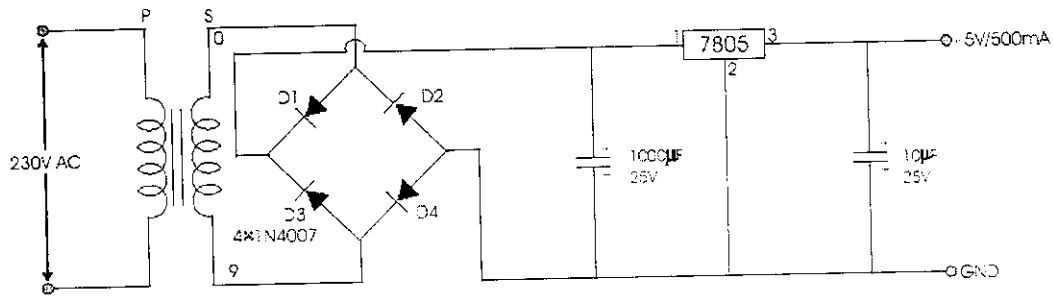


Fig 7.3 Circuit Diagram of Power Supply

CHAPTER 8

PROGRAMMING SECTION

The software tools used in this project are

1. EAGLE 6.0
2. Keil UVision 3
3. Topwin
4. Microsoft Visual Basic 6.0

8.1 EAGLE 6.0

EAGLE (Easily Applicable Graphical Layout Editor) is an ECAD program produced by Cad soft in Germany. EAGLE provides a schematic editor, for designing circuit diagrams and a tightly integrated PCB layout editor, which automatically starts off with all of the components required by the schematic. Components are manually arranged on the board, with the help of coloured lines showing the eventual connections between pins that are required by the schematic.

8.2 KEIL UVISION 3

Keil UVision 3 software is used to convert the embedded C program into a machine understandable HEX file.

8.3 TOPWIN

Topwin software is used to burn the converted HEX file into the microcontroller using Universal Programmer kit. Topwin, a type of software developed for TOP series programmers, adapts to the TOP hardware products of a new generation. TopWin supports multi-window operation, namely, it can connect multiple programmers on a computer to write device without any interference.

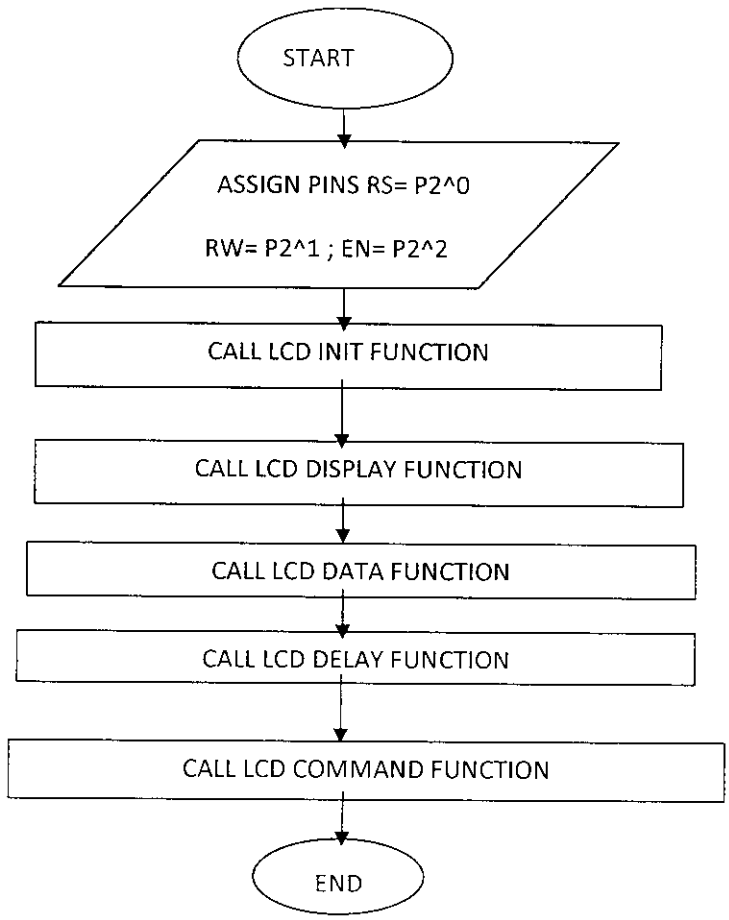
8.4 MICROSOFT VISUAL BASIC 6.0

Visual Basic (VB) is the third-generation event-driven programming language and integrated development environment (IDE) from Microsoft for its COM programming model. Visual Basic was derived from BASIC and enables the rapid application development (RAD) of graphical user interface (GUI)

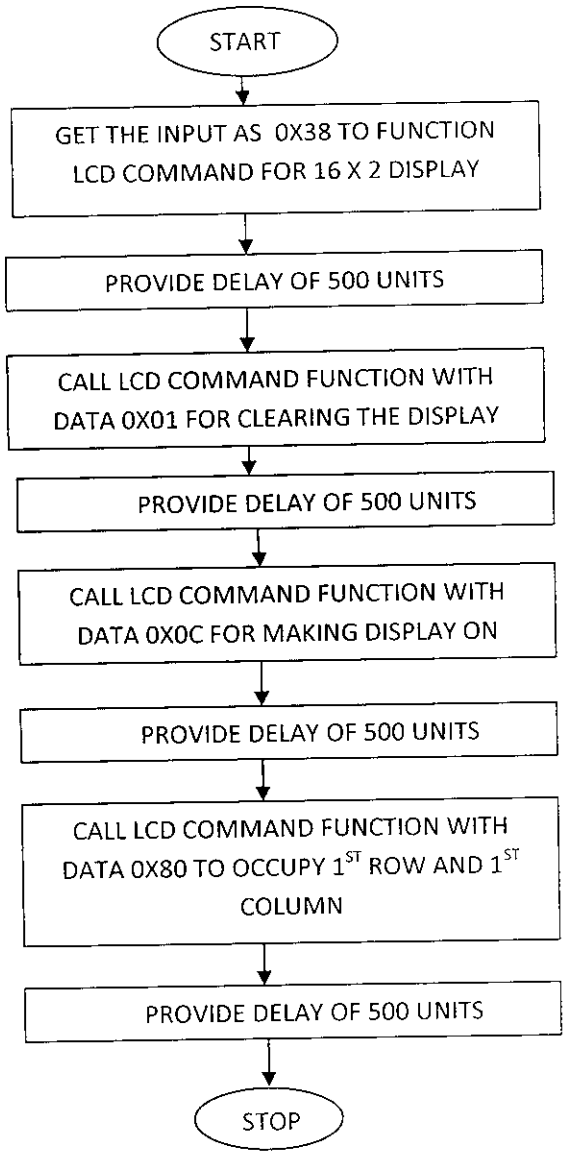
applications, access to databases using Data Access Objects. Moreover visual basic has advanced controls over the hardware.

8.5 PROGRAM FLOW OF LCD HEADER FILE

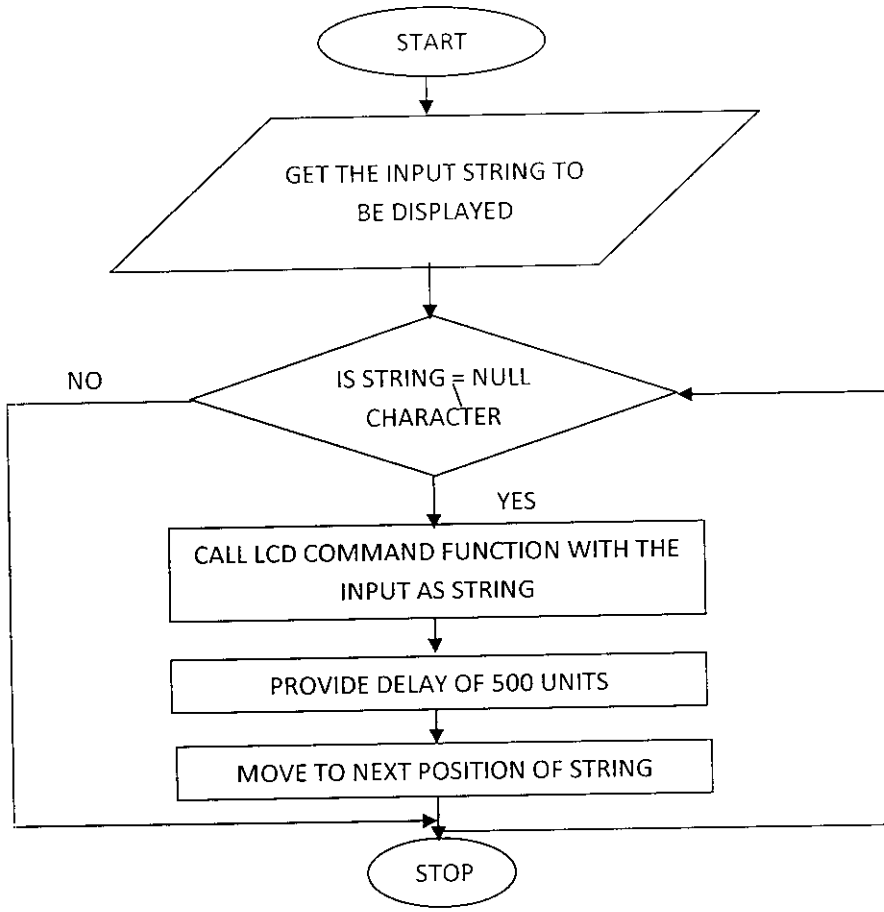
8.5.1 OVERALL FLOW



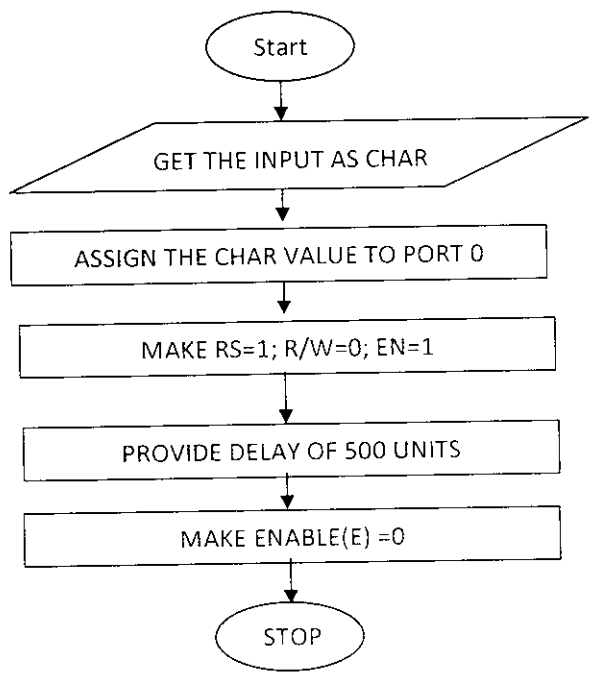
8.5.2 LCD INITIALIZE FUNCTION



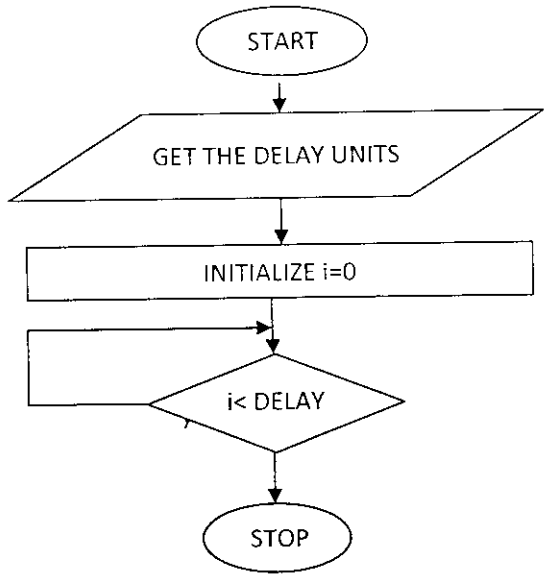
8.5.3 LCD DISPLAY FUNCTION:



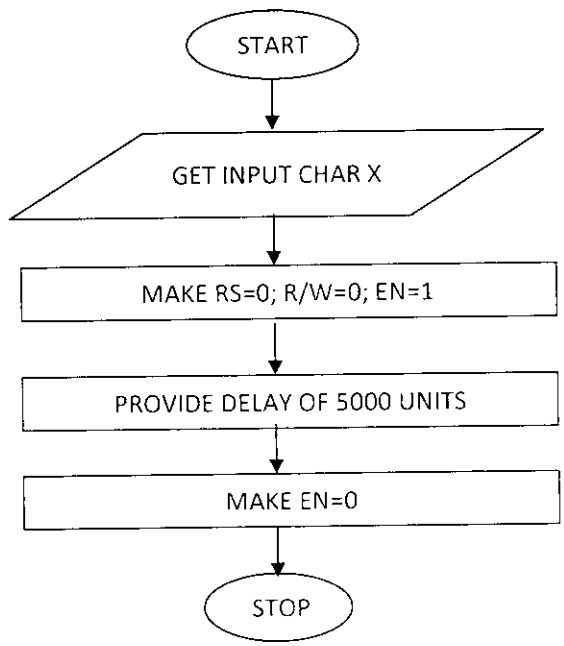
8.5.4 LCD DATA FUNCTION



8.5.5 LCD DELAY FUNCTION:

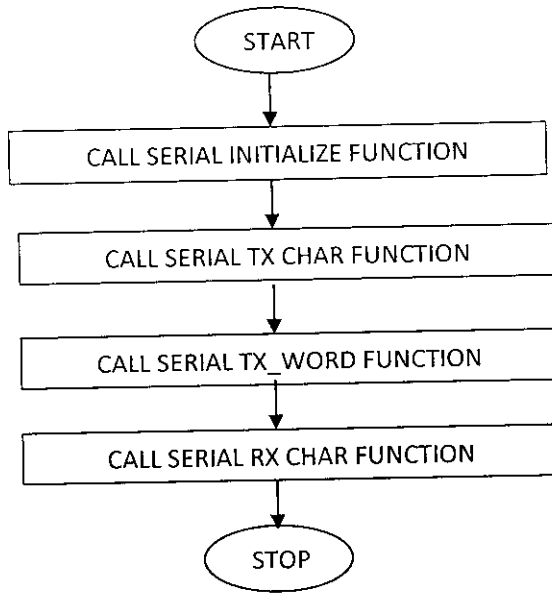


8.5.6 LCD COMMAND FUNCTION:

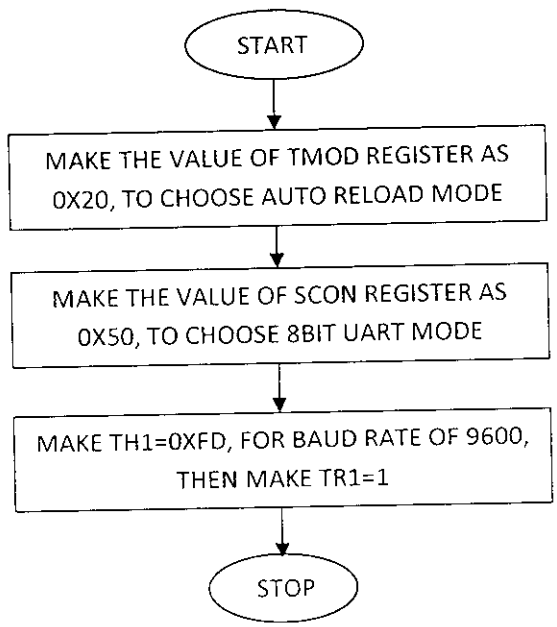


8.6 SERIAL COMMUNICATION HEADER

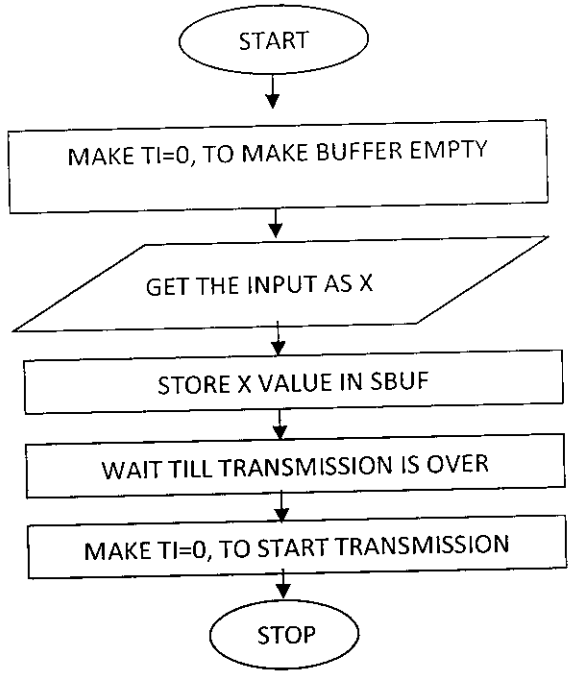
8.6.1 OVERFLOW



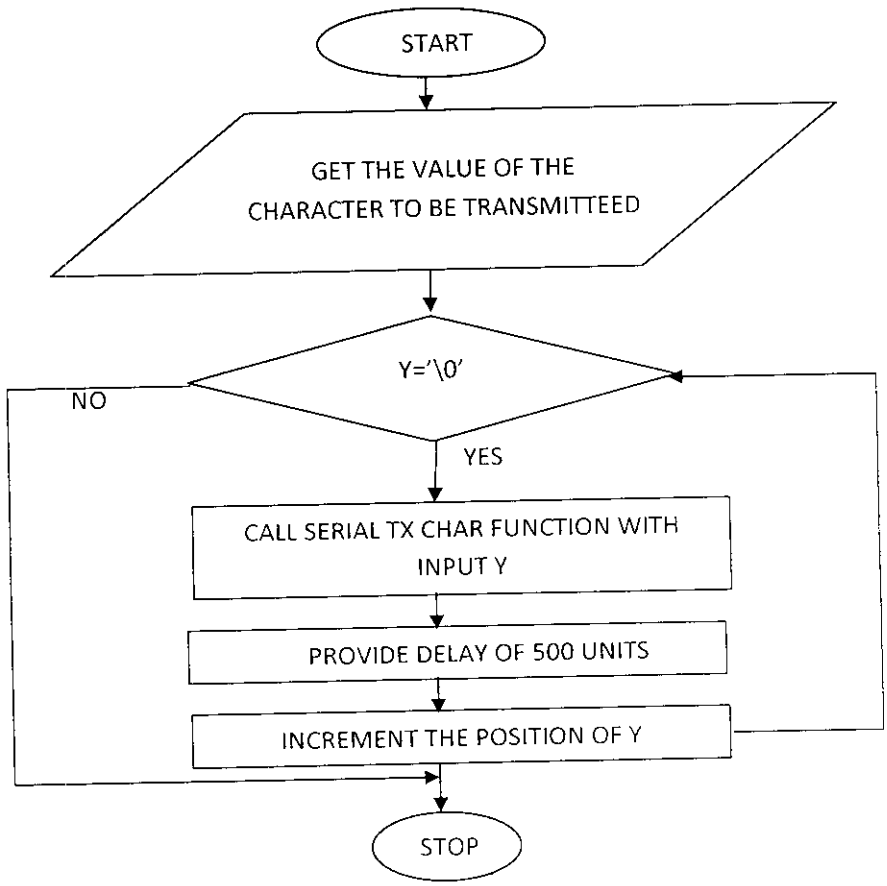
8.6.2 SERIAL INITIALIZE FUNCTION:



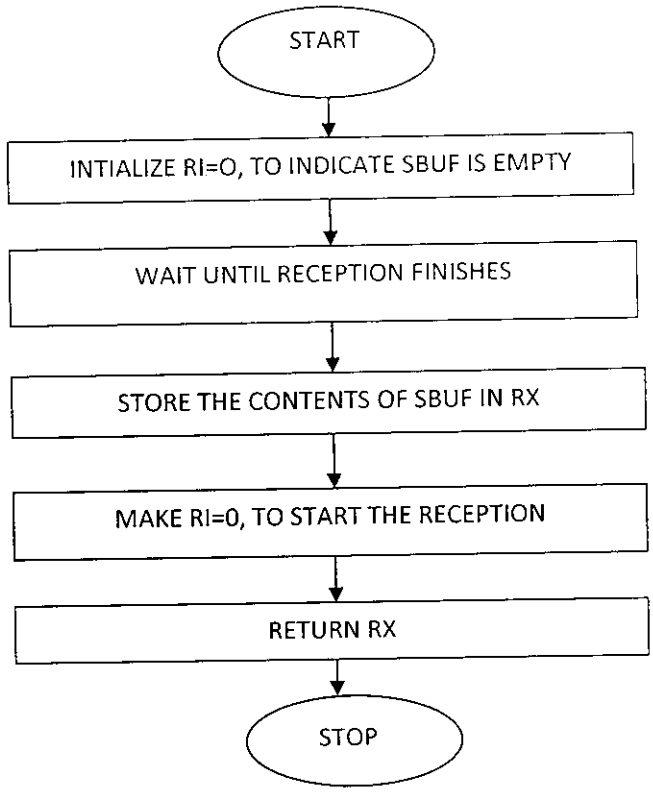
8.6.3 SERIAL TX CHAR FUNCTION



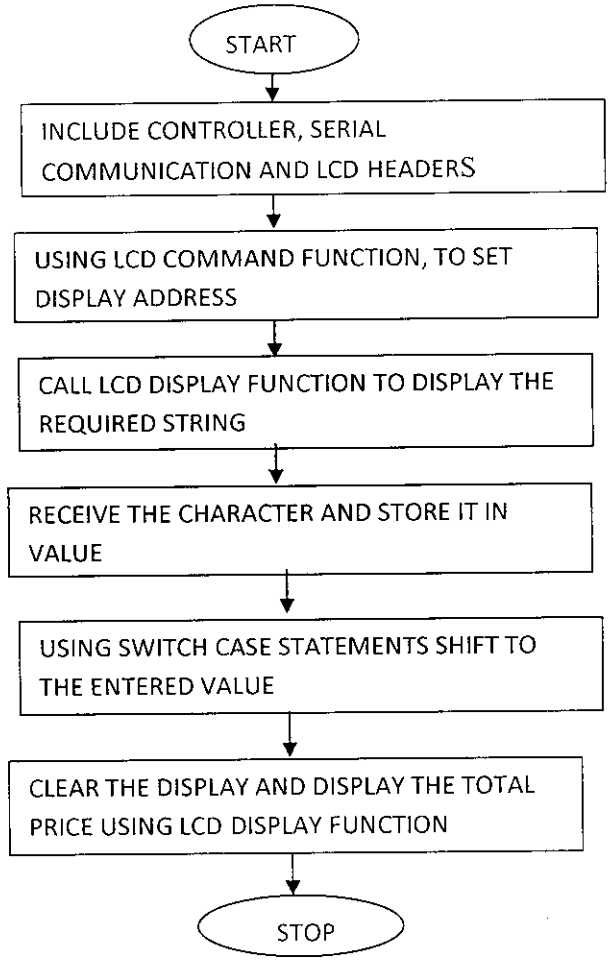
8.6.4 SERIAL TX WORD FUNCTION



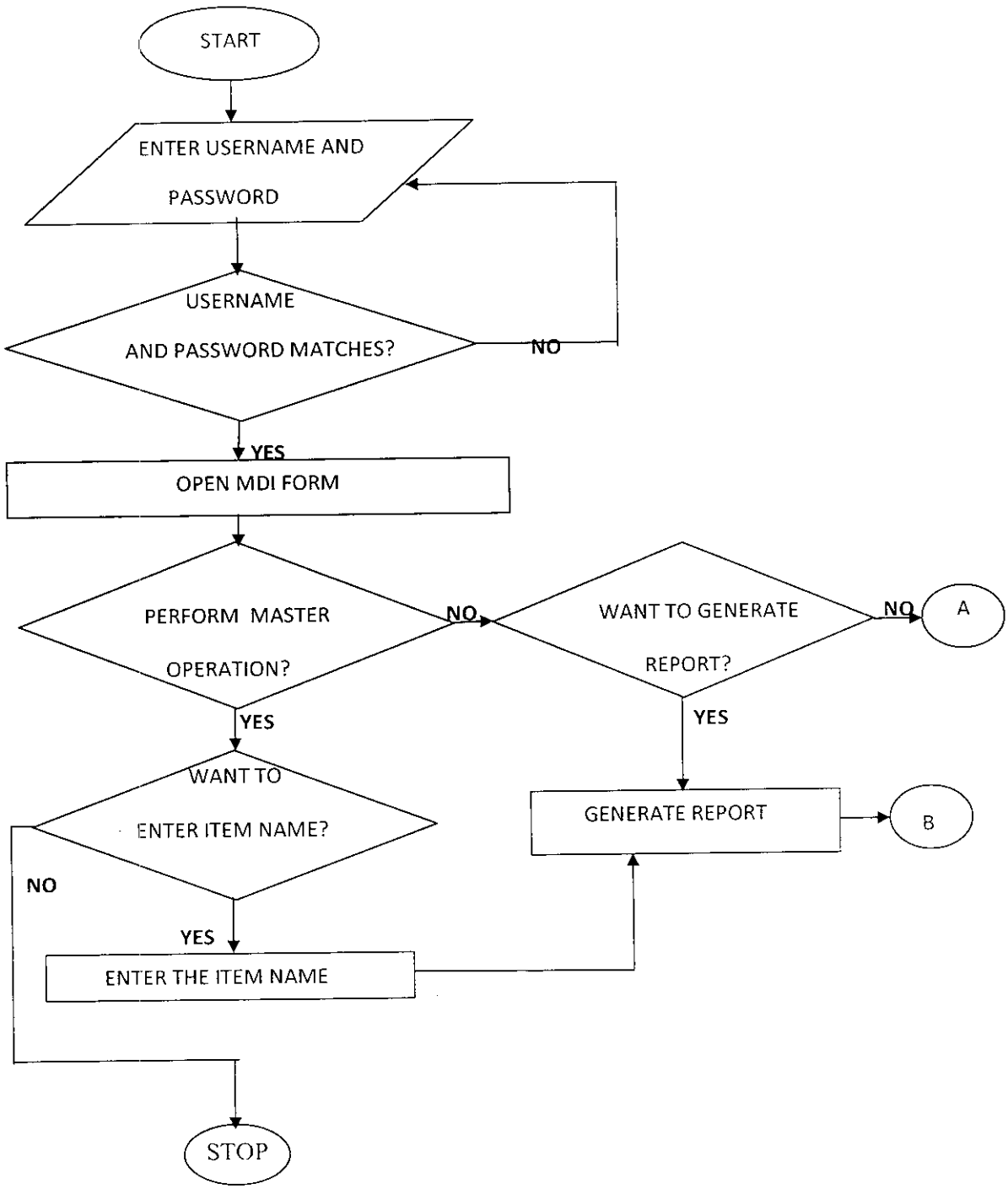
8.6.5 SERIAL RX CHAR FUNCTION

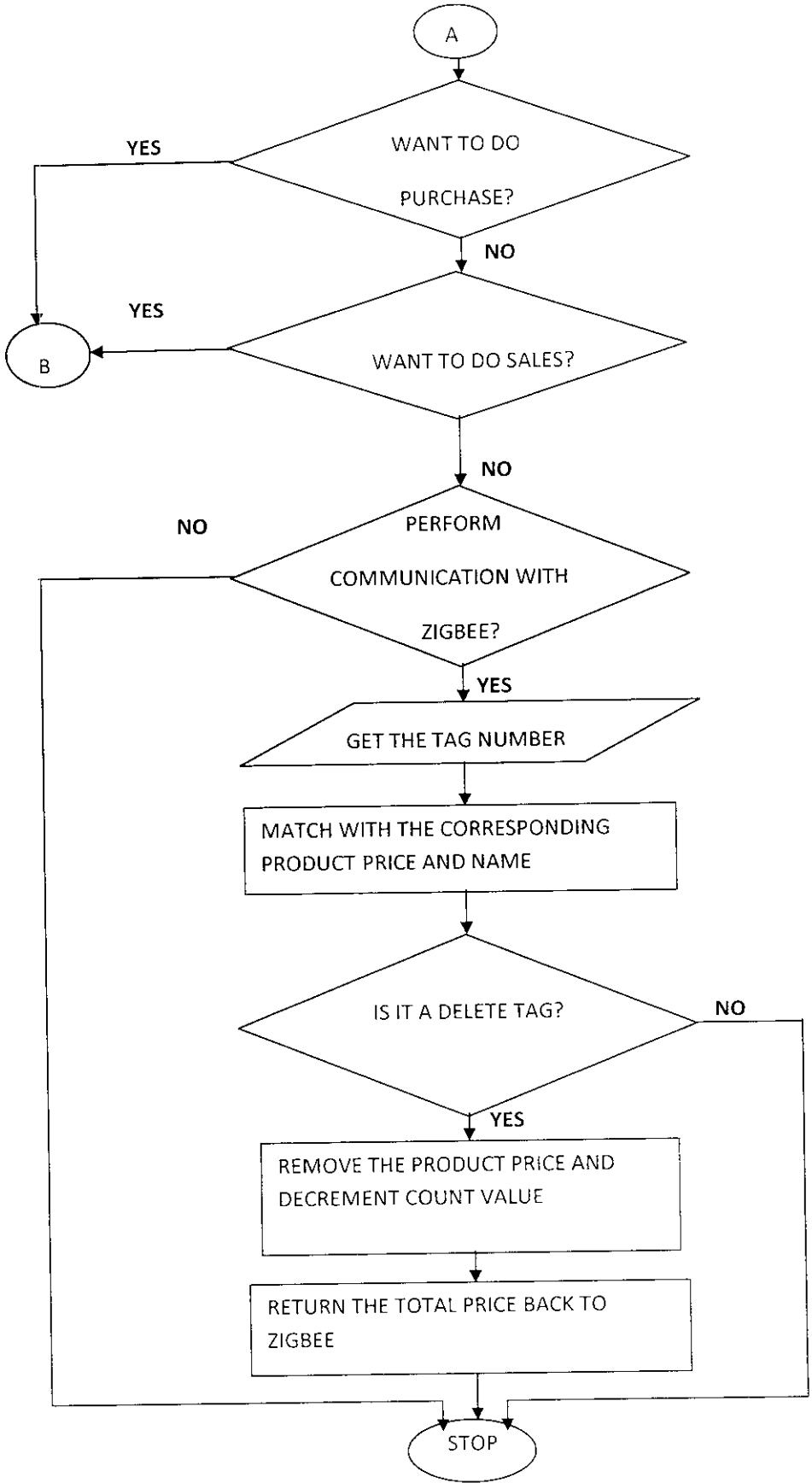


8.7 MAIN FUNCTION



8.8 VB PROGRAMMING FLOW




A
WANT TO DO PURCHASE?
YES
NO
B
YES
WANT TO DO SALES?
NO
NO
PERFORM COMMUNICATION WITH ZIGBEE?
YES
GET THE TAG NUMBER
MATCH WITH THE CORRESPONDING PRODUCT PRICE AND NAME
IS IT A DELETE TAG?
NO
YES
REMOVE THE PRODUCT PRICE AND DECREMENT COUNT VALUE
RETURN THE TOTAL PRICE BACK TO ZIGBEE
STOP

CHAPTER 9

FUTURE ENHANCEMENT AND CONCLUSION

FUTURE ENHANCEMENT

- In future this project can be implemented to supermarket in real time working environment.
- Then this project can be extended to a PAN (Personal Area Network) with a centralized common control.

CONCLUSION

This project finds time efficient for billing in counters. The labour work is reduced. Avoids crowding at counters. Helps the customer to purchase economically. A thought ahead with future RFID tags on products in super markets. IR counters which are used for used for counting of products is also completely replaced by RFID technique.

APPENDICES

DATASHEETS

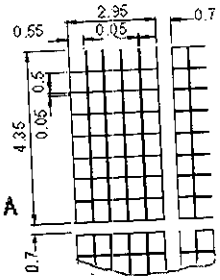
1. LCD DISPLAY (JHD 549)

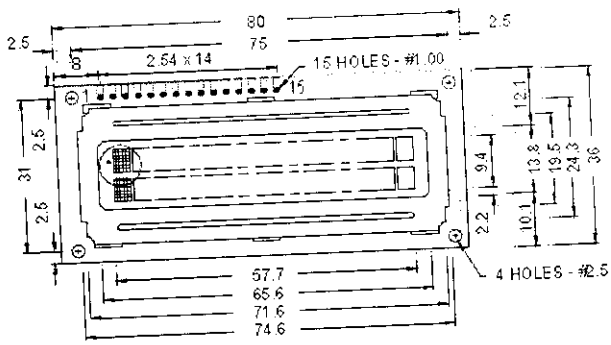
Alphanumeric LCD Module

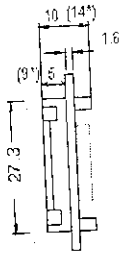
ODM-16216

16 Character 2 line

EXTERNAL DIMENSIONS







DISPLAY PATTERN DIMENSIONS

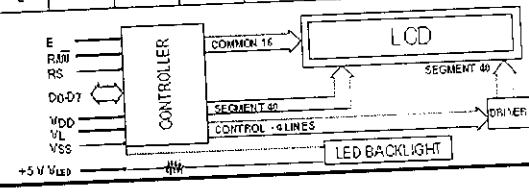
Specification marked * pertains to Backlight Version

GENERAL TOLERANCE : +/- 0.5

UNIT : mm

BLOCK DIAGRAM

PIN NO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
SYMBOL	VSS	VDD	VL	RS	R/W	E	D0	D1	D2	D3	D4	D5	D6	D7	+VLED



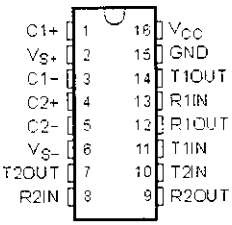
2. MAX 232

MAX232, MAX232I
DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047L – FEBRUARY 1989 – REVISED MARCH 2004

- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Operates From a Single 5-V Power Supply With 1.0-μF Charge-Pump Capacitors
- Operates Up To 120 kbit/s
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- ESD Protection Exceeds JESD 22
  - 2000-V Human-Body Model (A114-A)
- Upgrade With Improved ESD (15-kV HBM) and 0.1-μF Charge-Pump Capacitors is Available With the MAX202
- Applications
  - TIA/EIA-232-F, Battery-Powered Systems, Terminals, Modems, and Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply TIA/EIA-232-F voltage levels from a single 5-V supply. Each receiver converts TIA/EIA-232-F inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V, a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into TIA/EIA-232-F levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T_A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube of 25	MAX232N	MAX232N
	SOIC (D)	Tube of 40	MAX232D	MAX232
		Reel of 2500	MAX232DR	
	SOIC (DW)	Tube of 40	MAX232DW	MAX232
		Reel of 2000	MAX232DWR	
	SOP (NS)	Reel of 2000	MAX232NSR	MAX232
−40°C to 85°C	PDIP (N)	Tube of 25	MAX232IN	MAX232IN
	SOIC (D)	Tube of 40	MAX232ID	MAX232I
		Reel of 2500	MAX232IDR	
	SOIC (DW)	Tube of 40	MAX232IDW	MAX232I
		Reel of 2000	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package

3. ZIGBEE (XBEE PRO ZNET 2.5)

1. Overview

The XBee/XBee-PRO ZNet 2.5 OEM (formerly known as Series 2 and Series 2 PRO) RF Modules were engineered to operate within the ZigBee protocol and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between remote devices.

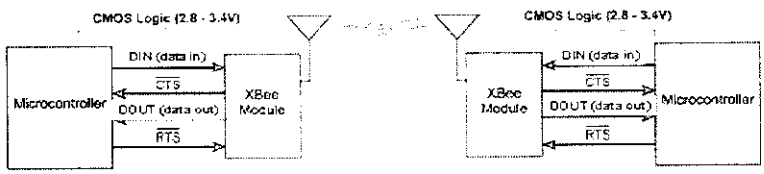
The modules operate within the ISM 2.4 GHz frequency band and are compatible with the following:

- XBee RS-232 Adapter
- XBee RS-232 PH (Power Harvester) Adapter
- XBee RS-485 Adapter
- XBee Analog I/O Adapter
- XBee Digital I/O Adapter
- XBee Sensor Adapter
- XBee USB Adapter
- XStick
- ConnectPort X Gateways
- XBee Wall Router.

2.1.1. UART Data Flow

Devices that have a UART interface can connect directly to the pins of the RF module as shown in the figure below.

Figure 2-01. System Data Flow Diagram in a UART-interfaced environment
(Low-asserted signals distinguished with horizontal line over signal name.)

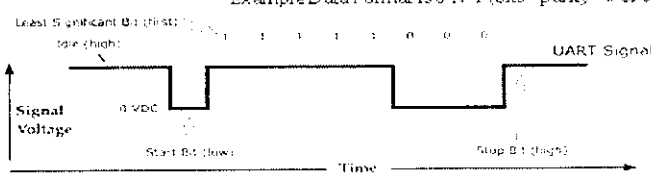


Serial Data

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

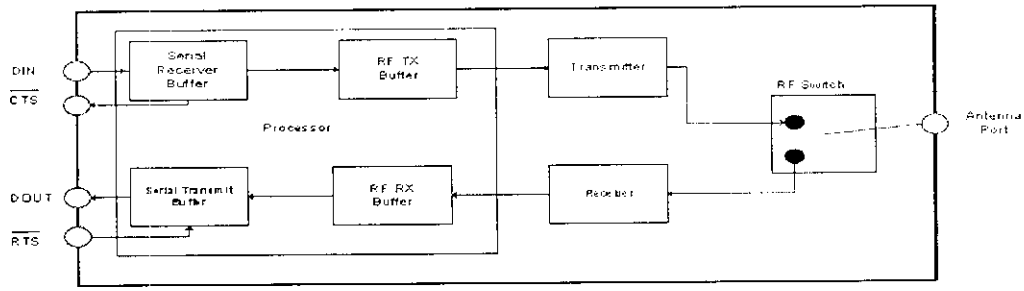
Each data byte consists of a start bit (low), 8 data bits (least significant bit first) and a stop bit (high). The following figure illustrates the serial bit pattern of data passing through the module.

Figure 2-02. UART data packet 0x1F (decimal number "31") as transmitted through the RF module
Example Data Format is 8-N-1 (bits - parity - # of stop bits)



The module UART performs tasks, such as timing and parity checking, that are needed for data communications. Serial communications depend on the two UARTs to be configured with compatible settings (baud rate, parity, start bits, stop bits, data bits).

Figure 2-03. Internal Data Flow Diagram



Serial Receive Buffer

When serial data enters the RF module through the DIN Pin (pin 3), the data is stored in the serial receive buffer until it can be processed. Under certain conditions, the module may not be able to process data in the serial receive buffer immediately. If large amounts of serial data are sent to the module, $\overline{CTS}$ flow control may be required to avoid overflowing the serial receive buffer.

Cases in which the serial receive buffer may become full and possibly overflow:

1. If the module is receiving a continuous stream of RF data, the data in the serial receive buffer will not be transmitted until the module is no longer receiving RF data.

2. If the module is transmitting an RF data packet, the module may need to discover the destination address or establish a route to the destination. After transmitting the data, the module may need to retransmit the data if an acknowledgment is not received, or if the transmission is a broadcast. These issues could delay the processing of data in the serial receive buffer.

Serial Transmit Buffer

When RF data is received, the data is moved into the serial transmit buffer and sent out the UART. If the serial transmit buffer becomes full enough such that all data in a received RF packet won't fit in the serial transmit buffer, the entire RF data packet is dropped.

Cases in which the serial transmit buffer may become full resulting in dropped RF packets

1. If the RF data rate is set higher than the interface data rate of the module, the module could receive data faster than it can send the data to the host.

2. If the host does not allow the module to transmit data out from the serial transmit buffer because of being held off by hardware flow control.

1.1. Key Features

High Performance, Low Cost

- Indoor/Urban: up to 300' (100 m)
- Outdoor line-of-sight: up to 1 mile (1.6 km)
- Transmit Power Output: 100 mW (20 dBm) EIRP
- Receiver Sensitivity: -102 dBm

RF Data Rate: 250,000 bps

Advanced Networking & Security

Retries and Acknowledgements

DSSS (Direct Sequence Spread Spectrum)

Each direct sequence channel has over 65,000 unique network addresses available

Point-to-point, point-to-multipoint and peer-to-peer topologies supported

Self-routing, self-healing and fault-tolerant mesh networking

Low Power

XBee PRO ZNet 2.5

- TX Current: 295 mA (@3.3 V)
- RX Current: 45 mA (@3.3 V)
- Power-down Current: < 1 µA @ 25°C

Easy-to-Use

No configuration necessary for out-of box RF communications

AT and API Command Modes for configuring module parameters

Small form factor

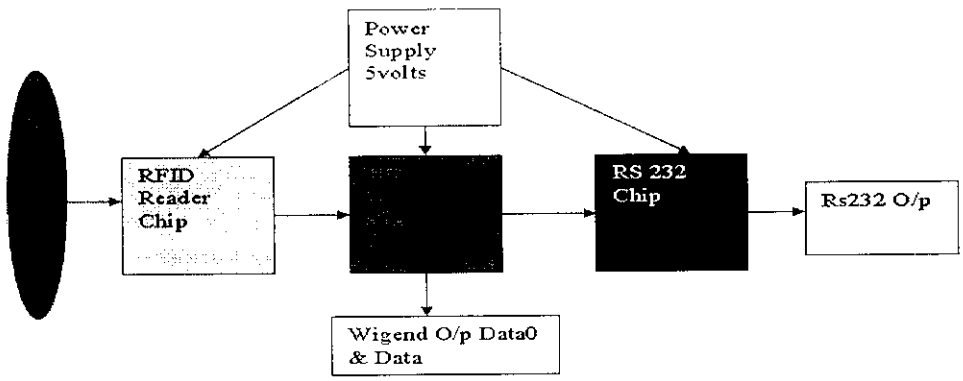
Extensive command set

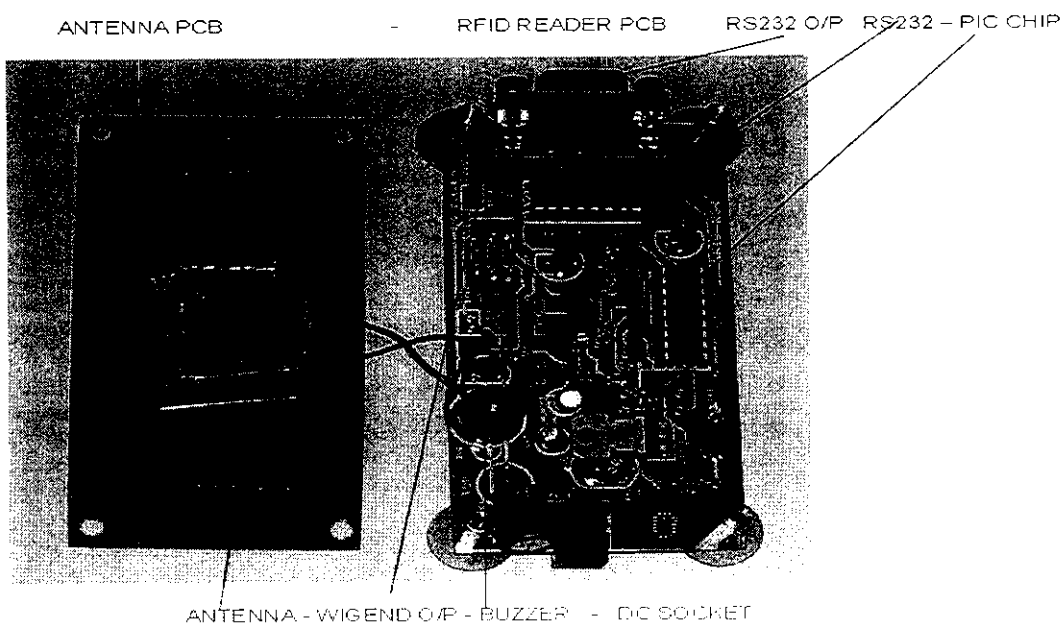
Free X-CTU Software (Testing and configuration software)

Free & Unlimited Technical Support

4. RFID READER

BLOCK DIAGRAM OF RFID READER NSK EDK 125KHZ





Features

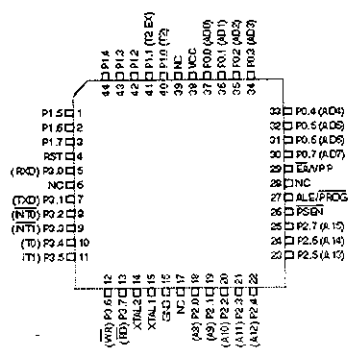
- Compatible with MCS-51™ Products
- 8K Bytes of In-System Reprogrammable Flash Memory
- Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

Description

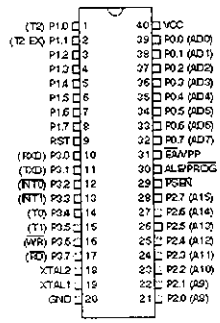
The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 and 80C52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

Pin Configurations

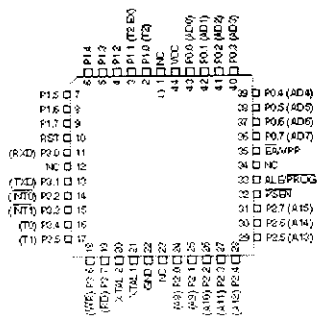
PQFP/TQFP



PDIP



PLCC

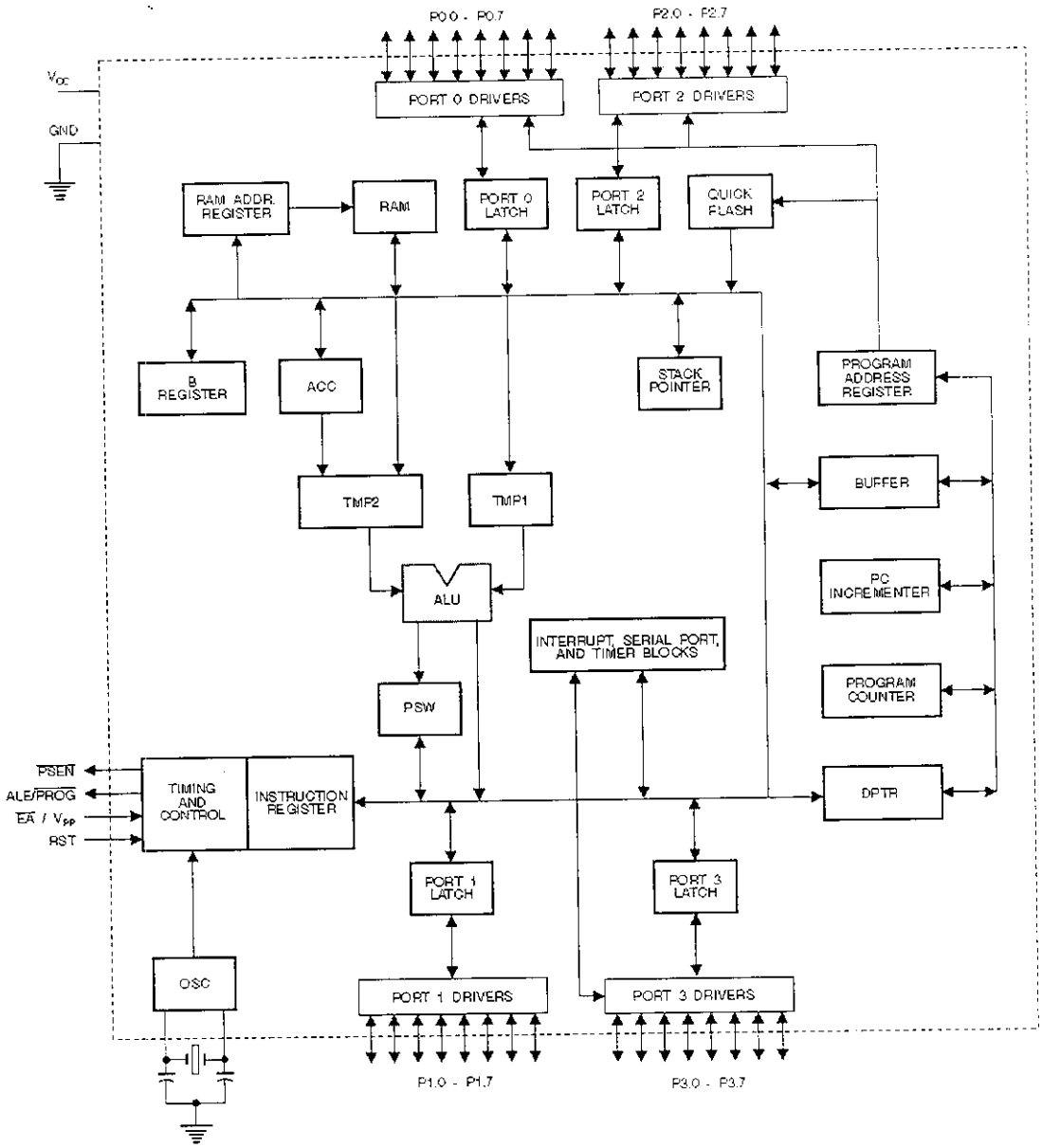


8-bit Microcontroller with 8K Bytes Flash

AT89C52

Rev. 0313H-(0/00)

Block Diagram



REFERENCES

1 Artist. http://www.artist-embedded.org/overview/.

2 Balarin, F. et al. 1997. Polis: A Design Environment for Control-Dominated Embedded Systems. Kluwer, Boston, MA.

3 Balarin, F. et al. 1999. Synthesis of software programs for embedded control application. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

4 Felice Balarin , Yosinori Watanabe , Harry Hsieh , Luciano Lavagno , Claudio Passerone , Alberto Sangiovanni-Vincentelli, Metropolis: An Integrated Electronic System Design Environment, Computer, v.36 n.4, p.45-52, April 2003 [doi>10.1109/MC.2003.1193228].

5 Massimo Baleani , Frank Gennari , Yunjian Jiang , Yatish Patel , Robert K. Brayton , Alberto Sangiovanni-Vincentelli, HW/SW partitioning and code generation of embedded control applications on a reconfigurable architecture platform, Proceedings of the tenth international symposium on Hardware/software codesign, May 06-08, 2002, Estes Park, Colorado [doi>10.1145/774789.774820].

6 Gérard Berry , Georges Gonthier, The ESTEREL synchronous programming language: design, semantics, implementation, Science of Computer Programming, v.19 n.2, p.87-152, Nov. 1992 [doi>10.1016/0167-6423(92)90005-V].

7 www.alldatasheets.com .

8 www.visualbasic.freetutes.com .