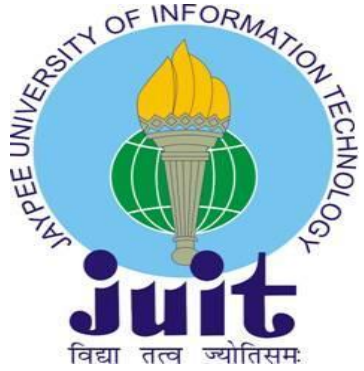
MICROCONTROLLER BASED SERIAL DATA TRANSFER TO PC

Submitted in partial fulfilment of the Degree of

Bachelor of Technology



May 2015

Enrolment. No.	**111009, 111016, 111108**
Name of Student	**AnubhavVerma, Kartikeya Bahl, Nirav Gupta**
Name of supervisor	**Dr. Pradeep Kumar**

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY,

WAKNAGHAT

CERTIFICATE

This is to Certify that project report entitled "Microcontroller based Serial Data Transfer to PC", submitted by AnubhavVerma(111009), Kartikeya Bahl(111016) andNirav Gupta(111108), in partial fulfillment for the award of degree of Bachelor of Technology in Electronics and Communication Engineering to Jaypee University of Information Technology, Waknaghat, Solan has been carried out under my supervision.

This work has not been submitted partially or fully to any other university or Institute for the award of this or any other degree or diploma.

Date: 26.05.15

Supervisor's Name: Dr. Pradeep Kumar
Designation : Associate Professor

ACKNOWLEDGEMENTS

"The successful completion of any task would be incomplete without the support of the people who made it all possible and whose constant guidance and encouragement secured us the success."
We feel proud and privileged in expressing our deep sense of gratitude to all those who have helped us in presenting this assignment. We express our sincere gratitude to Dr. Pradeep Kumar for his inspiration, constructive suggestions, mastermind analysis and affectionate guidance in our work. It would have been impossible for us to complete this project without his guidance.

Lastly we would like to add our deepest gratitude for the entire faculty of ECE Department at JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY from where we have learnt the basics of Embedded Systems which helped us a lot in completion of this project.

Date: 26.05.15

AnubhavVerma (111009)
Kartikeya Bahl (111016)
Nirav Gupta (111108)

TABLE OF CONTENTS

ABSTRACT

This project is based on Microcontroller based Serial Data transfer to PC. The Microcontroller used is 8051 and serial data transfer takes place through DB9 port. The software used to transmit data is Hercules and the data is displayed on an LCD screen which has been interfaced with the microcontroller.

LIST OF FIGURES

1. INTRODUCTION

Serial communication is the process of sending data, bit after bit in a timed sequence, over a communication channel or computer bus. Although a serial link may seem inferior to a parallel one, since it can transmit less data per clock cycle, it is often the case that serial links can be clocked considerably faster than parallel links in order to achieve a higher data rate.

Serial communication is used for all long-haul communication and most computer networks, where the cost of cable and synchronization difficulties make parallel communication impractical. Over the years, dozens of serial protocols have been crafted to meet particular needs of embedded systems. USB (universal serial bus), and Ethernet, are a couple of the well-known computing serial interfaces.

Serial interfaces can be sorted into one of two groups: synchronous or asynchronous. A *synchronous* serial interface always pairs its data line(s) with a clock signal, so all devices on a synchronous serial bus share a common clock. *Asynchronous* means that data is transferred without support from an external clock signal. This transmission method is perfect for minimizing the required wires and I/O pins.

2. OBJECTIVE

The basic aim of this project is to design and implement a circuit which is used to transfer data serially to a PC.

The design of such a circuit requires a lot of expertise and experience. The circuit uses an 8051 microcontroller to send and receive data serially from the computer. Error detection in the data sent and received will also be taken care of.

3. TECHNICAL DETAILS

The softwares used in our project are:

Hercules SETUP Utility

Hercules SETUP Utility is useful serial port terminal (RS-232 Terminal). It is a freeware and it includes many functions in one utility. It was the first mainframe emulator to incorporate 64 bit z/Architecture support.

C51 C Compiler

The Keil C51 C Compiler for the 8051 microcontroller is the most popular 8051 C compiler in the world. The C51 Compiler allows you to write 8051 microcontroller applications in C that, once compiled, have the efficiency and speed of assembly language.

4. METHODOLOGY

4.1) Circuit Design:

The process of designing a circuit starts with the design of a power supply through which the circuit can function efficiently and without any risks of burnout. Once the power supply is designed, we can move onto the design of the actual circuit which performs the needed task. In this case, that task is the transfer of data to and from a PC to the microcontroller serially. This is achieved by making the voltage levels of the RS232 port in the PC and the TTL level microcontroller compatible by using a MAX 232 line driver.

Here we discuss the entire circuitry involved in the design of this project.

4.1.1) Regulated Power Supply:

Power supplies are designed to convert high voltage AC mains to a suitable low voltage supply for electronic circuits and other devices.

A power supply can be broken down into a series of blocks, each of which performs a particular function.

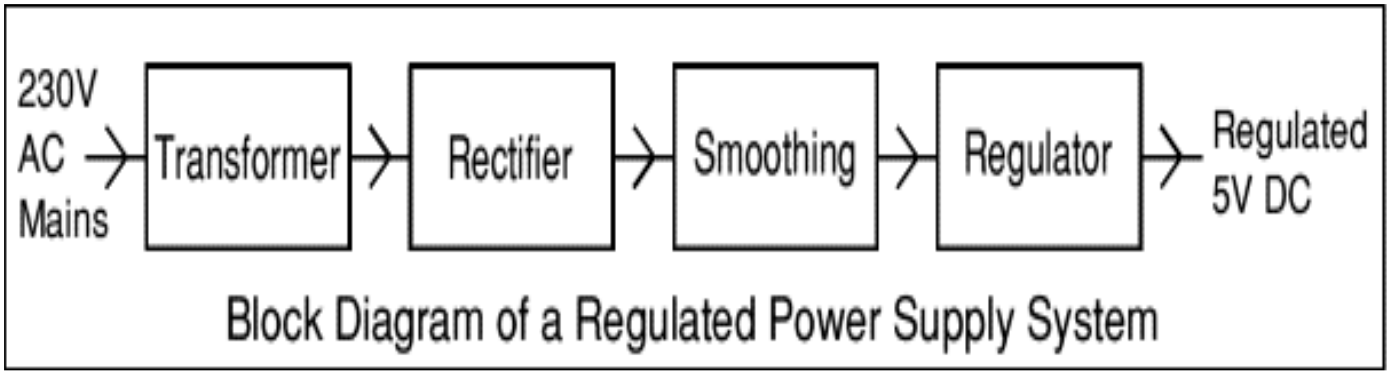


Figure 1: Block Diagram of a regulated Power Supply

- Transformer – steps down high voltage AC mains to low voltage AC.
- Rectifier – converts AC to DC, but the DC output is varying.
- Smoothing – smoothes the DC from varying greatly to a small ripple.
- Regulator – eliminates ripple by setting DC output to a fixed voltage.

TRANSFORMER:

Transformers convert AC electricity from one voltage to another with little loss of power. Transformers work only with AC and this is one of the reasons why mains electricity is AC. The two types of transformers

- Step-up transformers increase voltage
- Step-down transformers reduce voltage.

Most power supplies use a step-down transformer to reduce the dangerously high mains voltage (230V in UK) to a safer low voltage. Transformers waste very little power so the power out is (almost) equal to the power in. Note that as voltage is stepped down current is stepped up. The ratio of the number of turns on each coil, called the turn ratio, determines the ratio of the voltages. A step-down transformer has a large number of turns on its primary (input) coil which is connected to the high voltage mains supply, and a small number of turns on its secondary (output) coil to give a low output voltage.

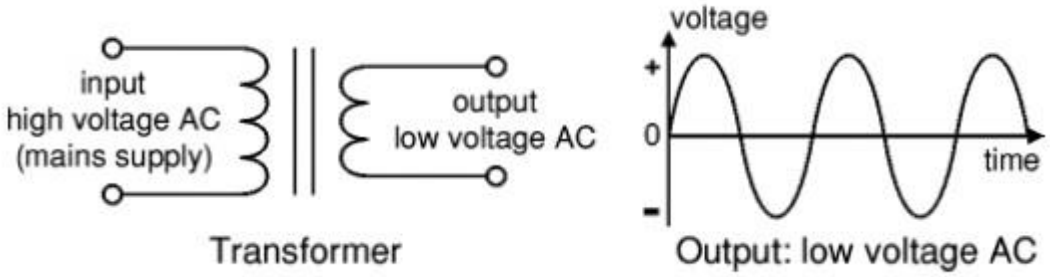


Figure 2: Transformer and its output voltage waveform

BRIDGE RECTIFIER:

A bridge rectifier can be made using four individual diodes, but it is also available in special packages containing the four diodes required. It is called a full-wave rectifier because it uses all AC wave (both positive and negative sections). 1.4V is used up in the bridge rectifier because each diode uses 0.7V when conducting and there are always two diodes conducting.

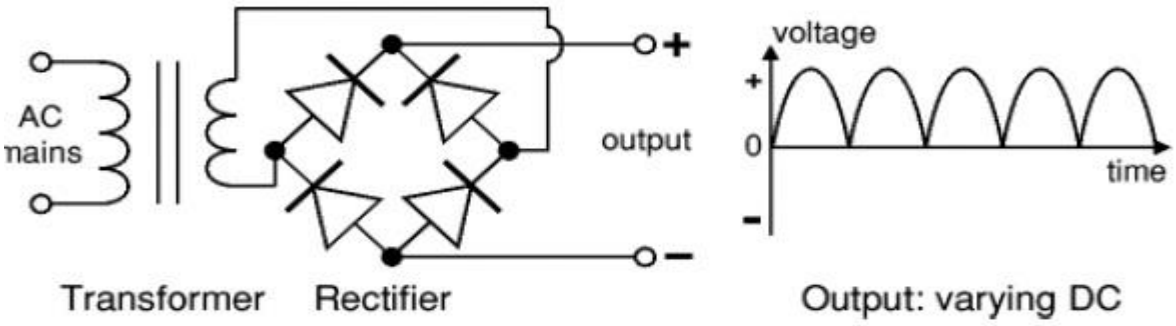


Figure 3: Rectifier output voltage waveform

Bridge rectifiers are rated by the maximum current they can pass and the maximum reverse voltage they can withstand (this must be at least three times the supply RMS voltage so the rectifier can withstand the peak voltages). In this alternate pairs of diodes conduct, changing over the connections so the alternating directions of AC are converted to the one direction of DC.

SMOOTHING:

Smoothing is performed by a large value electrolytic capacitor connected across the DC supply to act as a reservoir, supplying current to the output when the varying DC voltage from the rectifier is falling.

The diagram shows the unsmoothed varying DC (dotted line) and the smoothed DC (solid line). The capacitor charges quickly near the peak of the varying DC, and then discharges as it supplies current to the output.

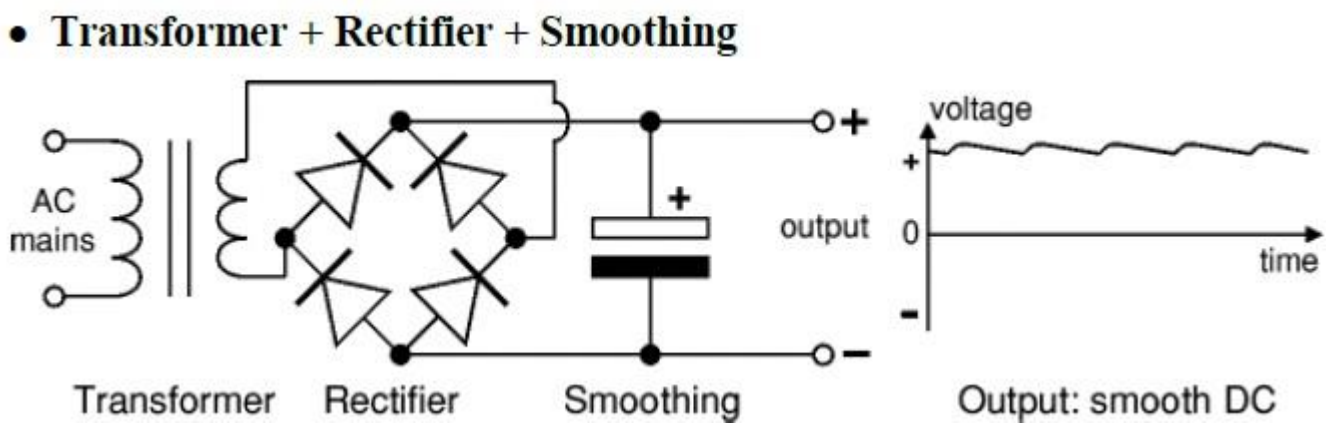


Figure 4: Smoothing output voltage waveform

Smoothing is not perfect due to the capacitor voltage falling a little as it discharges, giving a small ripple voltage. For many circuits a ripple which is 10% of the supply voltage is satisfactory. A larger capacitor will give fewer ripples. The capacitor value must be doubled when smoothing half-wave DC.

REGULATOR:

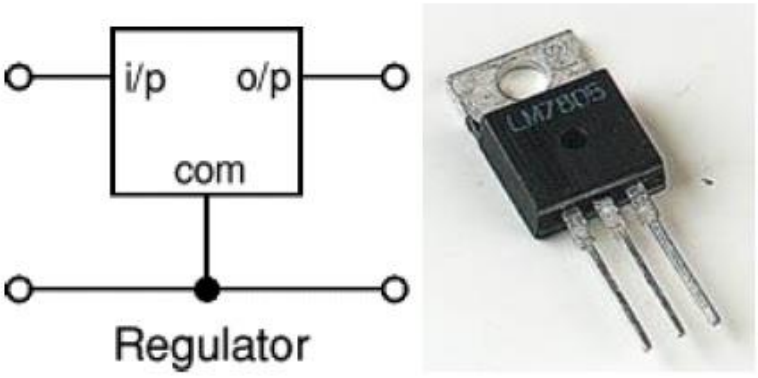


Figure 5: Regulator

Voltage regulator ICs are available with fixed (typically 5, 12 and 15V) or variable output voltages. They are also rated by the maximum current they can pass. Negative voltage regulators are available, mainly for use in dual supplies. Most regulators include some automatic protection from excessive current (‘overload protection') and overheating (‘thermal protection').

4.1.2) Microcontroller 8051:

In our day to day life the role of micro-controllers has been immense. They are used in a variety of applications ranging from home appliances, FAX machines, Video games, Camera, Exercise equipment, Cellular phones musical Instruments to Computers, engine control, aeronautics, security systems and the list goes on.

Microcontroller versus Microprocessors

What is the difference between a microprocessor and microcontroller? The microprocessors (such as 8086, 80286, 68000 etc.) contain no RAM, no ROM and no I/O ports on the chip itself. For this reason they are referred as general- purpose microprocessors. A system designer using general-purpose microprocessor must add external RAM, ROM, I/O ports and timers to make them functional. Althoughthe addition of external RAM, ROM, and I/O ports make the system bulkier and much more expensive, they have the advantage of versatility such that the designer can decide on the amount of RAM, ROM and I/o ports needed to fit the task athand. This is the not the case with microcontrollers. A microcontroller has a CPU (a microprocessor) in addition to the fixed amount of RAM, ROM, I/O ports, and timers are all embedded together on the chip: therefore, the designer cannot add any external memory, I/O, or timer to it. The fixed amount of on chip RAM, ROM, and number of I/O ports in microcontrollers make them ideal for many applications in which cost and space are critical.

8051 Microcontroller

In 1981, Intel Corporation introduced an 8-bit microcontroller called the 8051. This microcontroller had 128 bytes of RAM, 4K bytes of on-chip ROM, two timers, one serial port, and four ports (8-bit) all on a single chip. The 8051 is an 8-bit processor, meaning the CPU can work on only 8- bit pieces to be processed by the CPU. The 8051 has a total of four I/O ports, each 8- bit wide. Although 8051
can have a maximum of 64K bytes of on-chip ROM, many manufacturers put only 4K bytes on the chip.

AT89C51 from ATMEL Corporation:

This popular 8051 chip has on-chip ROM in the form of flash memory. This is ideal for fast development since flash memory can be erased in seconds compared to twenty minutes or more needed for the earlier versions of the 8051. To use the AT89C51 to develop a microcontroller-based system requires a ROM burner that supports flash memory: However, a ROM eraser is not needed. The PROM burner does this erasing of flash itself and this is why a separate burner is not needed.

Hardware features

- 40 pin IC
- 4 Kbytes of Flash
- 128 Bytes of RAM
- 32 I/O lines
- Two16-Bit Timer/Counters
- Two-Level Interrupt Architecture
- Full Duplex Serial Port
- On Chip Oscillator and Clock Circuitry

Software features

- Bit Manipulations
- Single Instruction Manipulation
- Separate Program and Data Memory
- 4 Bank of Temporary Registers
- Direct, Indirect, Register and Relative Addressing

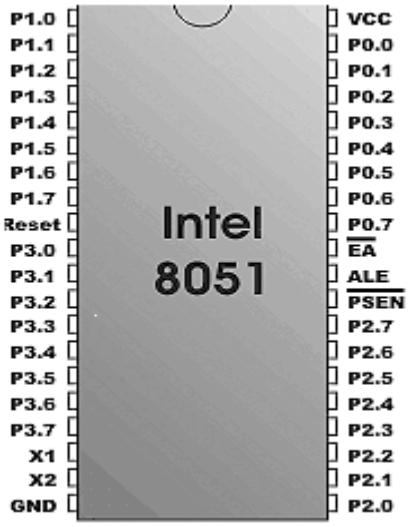


Figure 6: Pin Diagram of 8051 microcontroller

Pin description :

V_{cc}

Pin 40 provides supply voltage to the chip. The voltage source is +5V.

GND

Pin 20 is the ground.

Oscillator Characteristics

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which 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.

RST

Pin 9 is the reset pin. It is an input and is active high (normally low). Upon applying a high pulse to this pin, the microcontroller will reset and terminate all activities. This is often referred to as a power –on reset. Activating a power-on reset will cause all values in the registers to be lost. In order to RESET input to be effective, it must have a minimum duration of 2 machine cycles. In other words, the high pulse must be high for a minimum of 2 machine cycles before it is allowed to go low.

EA

It stands for 'external access'. All the 8051 family members come with on-chip ROM to store programs. In such cases, the EA pin is connected to the Vcc. For family members such as 8031 and 8032 in which there is no on-chip ROM, code is stored on an external ROM and is fetched by the 8031/32. Therefore for the 8031 the EA pin must be connected to ground to indicate that the code is stored externally.

PSEN

This is an output pin. PSEN stands for "program store enable." It is the read strobe to external program memory. When the microcontroller is executing from external memory, PSEN is activated twice each machine cycle.

ALE

ALE (Address latch enable) is an output pin and is active high. When connecting a microcontroller to external memory, port 0 provides both address and data. In other words the microcontroller multiplexes address and data through port 0 to save pins.

I/O port pins and their functions

The four ports P0, P1, P2, and P3 each use 8 pins, making them 8-bit ports. All the ports upon RESET are configured as output, ready to be used as output ports. To use any of these as input port, it must be programmed.

PORT 0

Port 0 occupies a total of 8 pins (pins 32 to 39). It can be used for input or output. To use the pins of port 0 as both input and output ports, each pin must be connected externally to a 10K-ohm pull-up resistor. This is due to fact that port 0 is an open drain, unlike P1, P2 and P3. With external pull-up resistors connected upon reset, port 0 is configured as output port. In order to make port 0 an input port, the port must be programmed by writing 1 to all the bits of it. Port 0 is also designated as AD0-AD7, allowing it to be used for both data and address. When connecting a microcontroller to an external memory, port 0 provides both address and data. The microcontroller multiplexes address and data through port 0 to save pins. ALE indicates if P0 has address or data. When ALE=0, it provides data D0- D7, but when ALE=1 it has address A0-A7.

PORT 1

Port 1 occupies a total of 8 pins (pins 1 to 8). It can be used as input or output. In contrast to port 0, this port does not require pull-up resistors since it has already pull-up resistors internally. Upon reset, port 1 is configures as an output port. Similar to port 0, port 1 can be used as an input port by writing 1 to all its bits.

PORT 2

Port 2 occupies a total of 8 pins (pins 21 to 28). It can be used as input or output. Just like P1, port 2 does not need any pull-up resistors since it has pull-up resistors internally. Upon reset port 2 is configured as output port. To make port 2 as input port, it must be programmed as such by writing 1s to it.

PORT 3

Port 3 occupies a total of 8 pins (pins 10 to 17). It can be used as input or output. P3 does not need any pull-up resistors, the same as P1 and P2 did not. Although port 3 is configured as output port upon reset, this is not the way it is most commonly used. Port 3 has an additional function of providing some extremely important signals such as interrupts. Some of the alternate functions of P3 are listed below:

P3.0 RXD (Serial input)

P3.1 TXD (Serial output)

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 memory write strobe)

P3.7 RD (External memory read strobe)

MEMORY SPACE ALLOCATION

Internal ROM

The 89C51 has 4K bytes of on-chip ROM. This 4K bytes ROM memory has memory addresses of 0000 to 0FFFh. Program addresses higher than 0FFFh, which exceed the internal ROM capacity, will cause the microcontroller to automatically fetch code bytes from external memory. Code bytes can also be fetched exclusively from an external memory, addresses 0000h to FFFFh, by connecting the external access pin to ground. The program counter doesn't care where the code is: the circuit designer decides whether the code is found totally in internal ROM, totally in external ROM or in a combination of internal and external ROM.

Internal RAM

The 128 bytes of RAM inside the 8051 are assigned addresses 00 to 7Fh. These 128 bytes can be divided into three different groups as follows:

- A total of 32 bytes from locations 00 to 1Fh are set aside for register banks and the stack.
- A total of 16 bytes from locations 20h to 2Fh are set aside for bit addressable read/write memory and instructions.

A total of 80 bytes from locations 30h to 7Fh are used for read and write storage, or what is normally called a scratch pad. These 80 locations of RAM are widely used for the purpose of storing data and parameters by 8051 programmers.

SERIAL COMMUNICATION

Data Communication Concepts

Within a microcomputer data is transferred in parallel, because that is the fastest way to do it. For transferring data over long distances, however, parallel data transmission requires too many wires. Therefore, data to be sent long distances is usually converted from parallel form to serial form so

that it can be sent on a single wire or pair of wires. Serial data received from a distant source is converted to parallel form so that it can be easily transferred on the microcomputer buses.

Serial Interface

Basic concepts concerning the serial communication can be classified into
categories below:

- Interfacing requirements
- Transmission format
- Error check in data communication
- Standards in serial I/O

Interfacing Requirements:

The serial interface requirement is very much similar to parallel interface requirement. Computer identifies the peripheral through port address and enable if using the read and write signals. The primary difference between the parallel I/O and serial I/O is the number of lines used for data transfer. Parallel I/O requires the entire bus while the serial I/O requires only one or pair of data lines for communication.

Transmission Format:

Transmission format for communication is concerned with the issues such as synchronization, direction of data flow, speed, errors and medium of transmission. Serial data can be sent synchronously or asynchronously.

Serial Transmission Methods

Serial Communication, like any data transfer, requires coordination between the sender and receiver. For example, when to start the transmission and when to end it, when one particular bit or byte ends and another begins, when the receiver's capacity has been exceeded, and so on. A protocol defines the specific methods of coordinating transmission between a sender and receiver.
Two serial transmission methods are used that correct serial bit errors.
The *first* one is ***synchronous*** communication, the sending and receiving ends of the communication are synchronized using a clock that precisely times the period separating each bit. By checking the clock the receiving end can determine if a bit is missing or if an extra bit (usually electrically induced) has been introduced in the stream.
Here is an example of this method of communication, lets say that on a conveyor belt a product is passing through a sensing device every 5 seconds, if the sensing device senses something in between the 5 second lap it assumes that whatever is passing is a foreign object of some sorts and sounds an alarm, if on the 5 second lap nothing goes by it assumes that the product is missing and sounds an alarm. One important aspect of this method is that if either end of the communication loses its clock signal, the communication is terminated.
The *alternative* method (used in PCs) is to add markers within the bit stream to help track each data bit. By introducing a start bit which indicates the start of a short data stream, the position of each bit can be determined by timing the bits at regular intervals, by sending start bits in front of each 8 bit streams, the two systems don't have to be synchronized by a clock signal, the only important issue is that both systems must be set at the same port speed. When the receiving end of the communication receives the start bit it starts a short term timer. By keeping streams short, there's not enough time for the timer to get out of sync. This method is known as ***asynchronous*** communication because the sending and receiving end of the communication are not precisely synchronized by the means of a signal line. By convention, the least significant bit of the word is sent first and the most significant it is sent last. When communicating the sender encodes the each word by adding a start bit in front and 1 or 2 stop bits at the end. Sometimes it will add a parity bit between the last bit of the word and the first stop bit, this used as a data integrity check. This is often referred to as a data frame. Five different parity bits can be used, the mark parity bit is always set at a logical 1, the space parity bit is always set at a logical 0, the even parity bit is set to logical 1 by counting the number of bits in the word and determining if the result is even, in the odd parity bit, the parity bit is set to logical 1 if the result is odd. The later two methods offer a means of detecting bit level transmission errors.

Bit Rates

Another important part of every asynchronous serial signal is the bit rate at which the data is transmitted. The rates at which the data is sent is based on the minimum speed of 300 bps (bits per second). Faster speeds are all based on the 300 bps rate, you merely double the preceding rate, so the rates are as follows, 600, 1200, 2400, 4800, 9600, 19200 and 38400 which is the fastest speed supported by today's BIOS's.

Asynchronous Serial Communication:

When a sender is connected to a receiver over an electrical connecting line, there is an initial state in which communication has not yet begun, called the idle or mark state. Because older electromechanical devices operate more reliably with current continually passing through them, the mark state employs a positive voltage level. Changing the state of the line by shifting the voltage to a negative value is called a space. Once this change has occurred, the receiver interprets a negative voltage level as a 0 bit, and a positive voltage level as a 1 bit. These transitions are shown in figure. The change from mark to space is known as the start bit, and this triggers the synchronization necessary for asynchronous serial transmission. The start bit delineates the beginning of the transmission unit defined as a character frame. The receiver then samples the voltage level at periodic intervals known as the bit time, to determine whether a 0-bit or a 1-bit is present on the line.

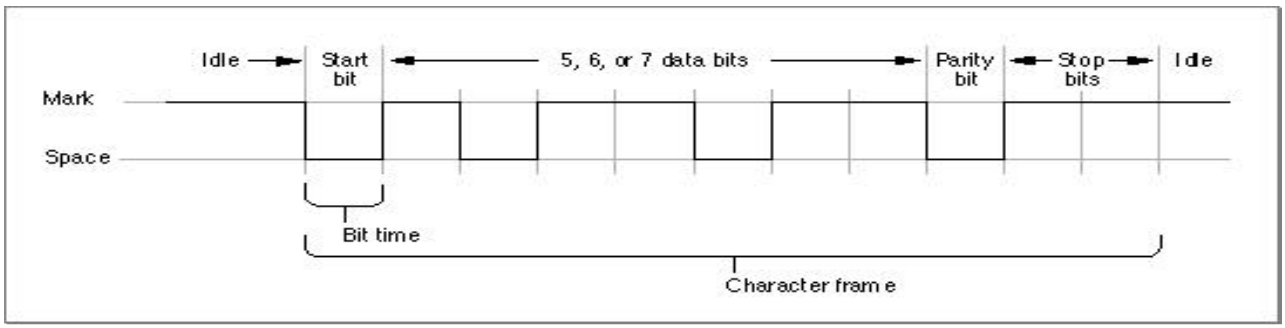


Figure 7: The format of serialized bits

The sampling rate must be agreed upon by sender and receiver prior to start of transmission in order for a successful transfer to occur. Common values for the sampling rate are 1200 baud and 2400 baud. In the case where one sampling interval can signal a single bit, a baud rate of 1200 results in a

transfer rate of 1200 bits per second (bps). Note that because modern protocols can express more than one bit value within the sampling interval, the baud rate and the data rate (bps) are not always identical.

After the data bits in the frame are sent, the sender can optionally transmit a paritybit for error-checking. There are various parity schemes, which the sender andreceiver must agree upon prior to transmission. In odd parity, a bit is sent so thatthe entire frame always contains an odd number of 1 bit. Conversely, in evenparity, the parity bit results in an even number of 1 bit. No parity means that noadditional bit is sent.

To signify the end of the character frame, the sender places the line back to themark state (positive voltage) for a minimum specified time interval. This intervalhas one of several possible values: 1 bit time, 2 bit times, or 1-1/2 bit times. Thissignal is known as the stop bit, and returns the transmission line back to idle status.

Standard in Serial I/O

The serial I/O technique is commonly used to interface terminals, printers etc. a standard is normally defined by a professional organization (such as IEEE). Astandard may include such items as assignment of pin positions for signals, voltagelevels, speed of data transfer, length of cable and mechanical specifications. Whendata are transmitted as voltage, the commonly used standard is known as RS232C.It is defined as reference to data terminal equipment (DTE) and data communication equipment (DCE). The rate of transmission is RS232C is restrictedto a maximum of 20k baud and a distance of 50 feet.

8051 Serial Communication Programming

Baud Rate in The 8051

The 8051 transfers and receives data serially at many different baud rates. Thebaud rate in the 8051 is programmable.

Frequency of XTAL = 11.0592 MHZ

Machine cycle frequency = 11.0592/12= 921.6 KHZ.

The 8051's serial communication UART circuitry divides the machine cycle frequency of 921.6 kHz by 32 before it is used by timer1 to set the baud rate. Result is 28,800 HZ. This value is used to find the timer 1 value to set the baudrate. When timer 1 is used to set the baud rate it must be programmed in mode 2.

Counter/Timer Programming

The 8051 has two timers/counters. They can be used either as timers to generate atime delay or as counters to count events happening outside the microcontroller.These timers are, timer 0 and timer 1.both are 16 bits wide, and each 16 bit timer isaccessed as two separate registers of low byte and high byte.

TMOD (Timer Mode) Register (20h)

Both timers 0 & 1 use the same register, called TMOD,to set the various timeroperation modes.TMOD is an 8-bit register in which the lower 4 bits are set asidefor timer 0 and the upper 4 bits are set aside for timer 1. In each case, the lower 2bits are used to set the timer mode and upper 2 bits to specify the operation.

SBUF Register

SBUF is an 8 bit register used solely for special communication in the 8051.for abyte of data to be transferred via the TxD line; it must be placed in the SBUFregister. Similarly, SBUF holds the byte of data when it is received by the 8051'sRxD line.

SCON (Serial Control) Register (50 H)

The SCON register is an 8 bit register used to program the start bit, stop bit, and data bits of data framing, among other things.

4.1.3) Display Unit (Liquid Crystalline Display)

Figure 8: 2x16 LCD Display Unit

The LCD, which is used as a display in the system, is LMB162A. The main features of this LCD are: 16 X 2 display, intelligent LCD, used for alphanumeric characters & based on ASCII codes. This LCD contains 16 pins, in which 8 pins are used as 8-bit data I/O, which are extended ASCII. Three pins are used as control lines these are Read/Write pin, Enable pin and Register select pin. Two pins are used for Backlight and LCD voltage, another two pins are for Backlight & LCD ground and one pin is used for contrast change.

Interfacing of micro controller with LCD display

Different LCD execute instructions at different rates and to avoid problems later on (such as if the LCD is changed to a slower unit). Before sending commands or data to the LCD module, the Module must be initialized. Once the initialization is complete, the LCD can be written to with data or instructions as required. Each character to display is written like the control bytes, except that the "RS" line is set. During initialization, by setting the "S/C" bit during the "Move Cursor/Shift Display" command, after each character is sent to the LCD, the cursor built into the LCD will increment to the next position (either right or left).

4.1.4) TTL to RS232 Line Driver Module

Serial Interface

Basic concepts concerning the serial communication can be classified into

categories below:

- Interfacing requirements
- Transmission format
- Error check in data communication
- Standards in serial I/O

Interfacing Requirements

Computer identifies the peripheral through port address andenable if using the read and write signals. The primary difference betweenthe parallel I/O and serial I/O is the number of lines used for data transfer.Parallel I/O requires the entire bus while the serial I/O requires only one or pair of data lines for communication.

Transmission Format

Transmission format for communication is concerned with the issues such assynchronization, direction of data flow, speed, errors and medium oftransmission. Serial data can be sent synchronously or asynchronously.

Synchronous Data Transmission

For synchronous data transmission data is sent in blocks at a constant rate.The start and end of the block are identified with specific bytes or bitpatterns.

Asynchronous Data Transmission

For asynchronous transmission each data character has a bit which identifiesits start and 1 or 2 bits, which identifies its end. Since each character isindividually identified, characters can be sent at any time (asynchronously),in the same way that a person types on a keyboard.

Asynchronous Data Format

Error Check In Data Communication

During transmission, various types of errors can occur. These errors need tobe checked, therefore, additional information for error checking is sentduring transmission the receiver can check the received data against the errorcheck information, and if the error is detected, the receiver can request thereretransmission of that data segment.

Standard in Serial I/O

The serial I/O technique is commonly used to interface terminals, printersetc. a standard is normally defined by a professional organization (such asIEEE). A standard may include such items as assignment of pin positions forsignals, voltage levels, speed of data transfer, length of cable and mechanicalspecifications.

DB-9

The female DB-9 connector is typically used as the "plug" that goes into atypical PC. The male DB-9 connector is the connector that you are more likely to see for serialcommunications on a "generic" PC.

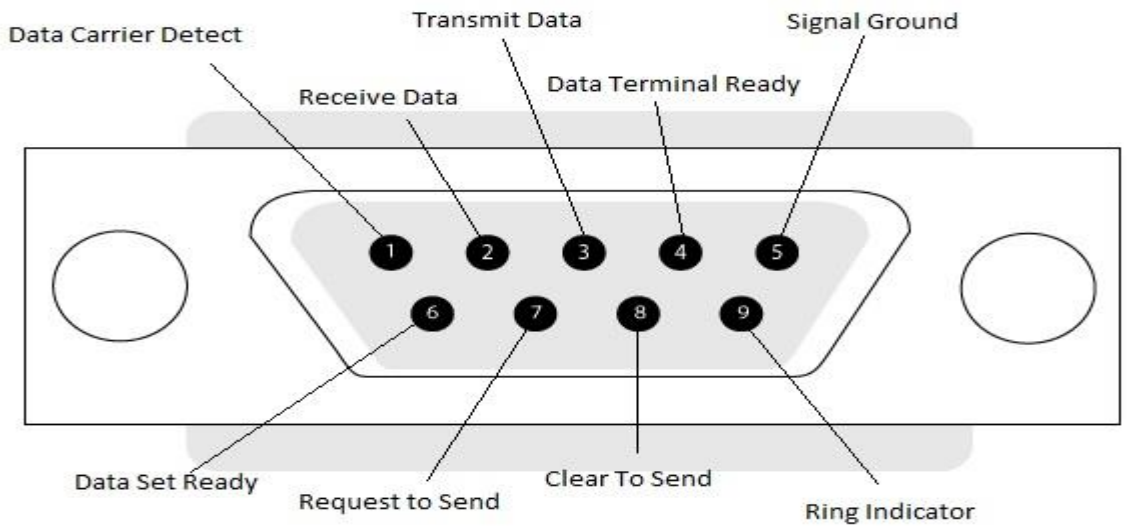


Figure 9: DB-9 Pin out diagram

TTL to RS232 Line-Driver (MAX 232) (communication interface)

This chip is used when interfacing micro controller with PC to check theBaud rate and changes the voltage level because micro controller is TTLcompatible whereas PC is CMOS compatible. The MAX 232 IC contains the necessary drivers and receivers, to adapt the RS- 232 signal voltage levels to TTL logic.

RS-232 Drivers

The typical driver output voltage swing is ±8V when loaded with a nominal 5kΩ RS-232 receiver and VCC = +5V. Input thresholds are both TTL and CMOS compatible.

5. C- CODE

The following code was programmed into the 8051 Microcontroller:

```
#include<lcdrout.h>          //header file for lcd
#include<serial.h>           //header file for serial comm
void main()
    {
    unsigned char rcvd_char[40];
    lcd_init();              //for lcd initialize
    lcd_cmd1(0x80);              //command for upper line first character
    lcd_puts("    Data    ");
    lcd_cmd1(0xc0);              //for lower line
    lcd_puts(" Communication  ");
    init_serial(9600);           //initialize serial comm at baud rate of 9600
    secdelay(2);             //wait for 3sec
    index=0;
    lcd_cmd1(0x01) ;
    while(1)
        {
        rcvd_char[index]=rec_data();
        buzz=0;
        ms_delay(40);
        buzz=1;
        if(rcvd_char[index]==13)
            {
            lcd_cmd1(0x01);
            lcd_cmd1(0x8f);
            lcd_puts(rcvd_char);
            ms_delay(800);
            while(RI==0)
                {
                lcd_cmd1(0x18);
```

```
                ms_delay(300);
                }
            RI=0;
            index=0;
            lcd_cmd1(0x01) ;
            }
        index++;
        }
}
```

6. RESULT

The hardware was successfully developed according to the needs of the project. The source code was also successfully burnt onto the Microcontroller unit. The operation of the project was checked and the errors were corrected, resulting in the desired working of the project.

7. CONCLUSION

Serial data transfer is more suitable for long distance communication purposes as it requires less number of wires and hence the circuit is less bulky. The voltage levels of the microcontroller and the PC are incompatible and to interface them, a line driver module is required in order to match the two voltage levels. Errors in the data can be decoded through Hercules software as well as the LCD unit.

8. REFERENCES

- Subrata Ghoshal-8051 Microcontroller- Internals, Instructions, Programming & Interfacing- Pearson Education India
- DC Kulshreshtha- Basic Electrical Engineering- Tata McGraw-Hill Education
- Barry B. Bray- The Intel Microprocessors- Prentice Hall
- http://www.engineersgarage.com/
- http://www.circuits4u.com/
- http://www.dkdynamics.com/