DESIGN AND CONSTRUCTIO OF QUIZ CONTROL SYSTEM

ALIYU JIBRIL MUSA
2005/22002EE

A THESIS SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING, FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA NIGER STATE.

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF BACHELOR IN ENGINEERING (B.ENG) ELECTRICAL AND COMPUTER ENGINEERING.

NOVEMBER, 2010

DEDICATION

This work is exclusively dedicated to the almighty ALLAH, the holy prophet Mohammed (S.A.W.), my parents and you who wish me the best in this world and the hereafter.

DECLARATION

I Aliyu Jibril Musa, declare that this work was done by me and has never been presented elsewhere for the award of degree. I also hereby relinquish the copyright to the Federal University of Technology, Minna.

ALIYU JIBRIL MUSA

(Student)

Dr. E. N. ONWUKA

(Supervisor)

Engr A.G. RAJI

(H.O.D. Elect & Comp Engineering)

(External Examiner)

ACKNOWLEDGEMENT

All praise is to ALLAH, Lord of the world, and Master of the Day of Judgment. He alone we worship and He alone we ask for help, who made me seen the end of this project by given me the best guidance throughout my work. I pray to ALLAH that his guidance should not only stop here, but continues to the end of time.

My project supervisor, Dr. Ms. E. N. Onwuka, I am particularly indebted to her whose commitment to this work at all stages gave me the best encouragement to keep going. Also, my profound gratitude goes to my parents who supported me in all forms. I also appreciate the effort of my assistant supervisor, Mallam Kudu Bagudu and other staff of the department for their support.

ABSTRACT

This project is a design and construction of Quiz Control System. The device was constructed basically using Microcontroller, Liquid Crystal Display (LCD), and Trigger Push Switches. Other components were also used in the construction which play important role in the device. The device is designed and constructed for ten contestants only; more contestants can be added by simply making further modification.

AT89C52 microcontroller used in this device is a programmable chip which monitors the whole process after properly programmed. It serves as a link of communication between the inputs (trigger push switches) and the output (LCD). The programming of the chip was achievable by studying its instructions set. The trigger push switches are the inputs unit used by the quizmaster and the contestants. The reset and start trigger switches are held by the quizmaster in monitoring the device while others are in the possession of the contestants. A 16x2 LCD displays the three fastest contestants to press the trigger switches within a given time whenever question is asked; no number is displayed after that time.

The device can work basically in two ways depending on how the quizmaster wants it to be, instructions given to all contestants before commencement.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1 Introduction to Quiz Control System

Choosing or picking out the fastest person from a group or competitors has been a serious task in this time of ours over the years. For that reason, in knowing the fastest out of a group or competitors, some factors have to be considered. These factors include: speed, accuracy, and time cautiousness. It is better to have a reliable device by which quiz can be conducted in other to prevent some of the problems encountered during quiz.

However, over time, various techniques have been used to get the fastest person in a quiz or competition. People are tested in quiz basically on timing and accuracy, for that reason, a reliable device is needed to do that.

Quiz, as we all know, is an important activity in schools and colleges and requires means by which people are to be chosen. This project will design and construct a device which is to be used to get the three best contestants, based on speed by the quiz master from ten contestants. This device is called "QUIZ CONTROL SYSTEM" In this device, each contestant or team has a trigger push switch in front of him [1]. The first contestant to finish immediately presses the push switch in front of him. The second contestant to finish also presses his push switch, and likewise the third contestant to finish do the same thing. The results of these three contestants will be recorded on the register of the MICROCONTROLER used by this mechanism [2]. The results of these contestants will be displayed as first, second, and third on the screen of the LIQUID CRYSTAL DISPLAY used [1].

The circuit has a reset trigger push used operated by the quiz master to cancel the whole process ready, for the next question [1].The whole process is monitored by the microcontroller which is programmed to register the activity carried out [2].

1.2 Aims and Objectives of this Project

Aims and objectives of this project are listed below:

1.2.1 Aims

The aims of this project can be seen as having:

- Reliable device used for quiz competition
- Improve students` abilities in solving questions very fast
- Eliminating the use of bell during quiz competition
- Accurate timing in quiz competition
- Answering of the same questions by contestants in a quiz competition

1.2.2 Objectives of this Project

The objectives of this project can be achieved through using:

- Programmable AT89C52 microcontroller
- LCD in displaying the three fastest contestants
- Push trigger switches as means of communication to the microcontroller
- A delay timing in the written programme

1.3 Methodology

During the process of carrying out the project work, the circuit diagram of each section of the construction was drawn, later on; all the sections were then combined together as a complete circuit of the device. Each of the components used was tested using digital multi-meter while others were examined by the inscriptions on them.

Furthermore, a temporary construction of the device was carried out on a project board, in order to see its functionality. Necessary adjustments and replacements of components were made on the board, so as to meet with the aim of the work. The microcontroller was then programmed using "C Language". The written programme was burnt on the microcontroller using universal programmer, and connected it along with the other components on the board using connecting wires.

Before permanent soldering of the components, microcontroller and other components used were tested on the project board in order to see their effectiveness, the results were recorded. Permanent soldering of the components was then carried out on a circuit board, after testing. The permanent soldered work was also tested after completion, and everything was cased using wood and screws.

1.4 Key Features of this Project

- Many number of units can be attached
- Easy installation
- No double pushes, no switch bouncing problems
- Results of the fastest three contestants display on the LCD
- No manual intervention
- Cheap

1.5 Scope of this Project

The QUIZ CONTROL SYSTEM, as the name implies, is basically for quiz. Although it can also be used for other purposes when carefully understood. Hence, the purpose(s) and application(s) of this device can be achievable with the aid of three basic different items which are: microcontroller, liquid crystal display, and trigger push switches.

In the design and construction of this device, AT89C52 microcontroller is programmed with the aid of its instructions set so as to accomplish its given tasks. This chip accepts any correct programme that is written on it. Firstly, its instructions set are studied in order to know the various functions of its port pins and how they are used; the written programme is the burnt on the microcontroller.

Also, before using the 16x2 LCD, each port on it will be studied in its data sheet so as to know how use it. This device, as it is known, is designed and constructed for ten contestants only. In that case, each of the contestants is having a push trigger switch in front of him, the reset and start push switches are held by the quizmaster which he uses in monitoring the device. The trigger push switches serve as the means of communication (input) with the microcontroller. The microcontroller processes any of the inputs and sends the result(s) to the LCD to display.

1.6 Sources of Materials used

It is important to mention the names and numbers of materials used in the design and construction of this device. These items are listed as shown below are below:

i. One AT89C52 microcontroller
ii. One 16x2 LCD
iii. Twelve trigger push switches
iv. One ON/OFF switch
v. One 9V battery
vi. Twelve 10kΩ resistors, one 470Ω resistor, and one 5kΩ variable resistor

vii. One 1μF capacitor, and two 33pF capacitors

viii. One 12MHz crystal oscillator

ix. One voltage regulator

x. Connecting wires, and

xi. Soldering lead

CHAPTER TWO

LITTERATURE REVIEW / THEORETICAL BACKGROUND

2.1 Theoretical Background

The QUIZ CONTROL SYSTEM is a modification of some works carried out by other people. The device is called QUIZ CONTROL SYSTEM because of the fact that it is mainly designed for quiz. This is implemented for the main purpose of solving the problems encountered during quiz. It is a simple circuit arrangement which consists of few components and devices in order to function properly.

The components and materials used in this work are very important; they make it possible for the process to be complete. Trigger push switches, microcontroller and liquid crystal display are the main components used to achieve the design of this device. Other important components are also included in this device, these are: capacitors, crystal oscillators, resistors, power source, and voltage regulator.

The quiz control system is simple in its operation. In the first place, the start push switch is pressed making the device ready for operation. At this time, the timing for the quiz has commenced. The microcontroller which is already programmed and the required timings are done on it in order for the contestants to solve a given problem within a given time [2]. A contestant immediately presses the push switch after completing solving his question, and the result is displayed on the LCD. The program written on the microcontroller only allows results of the fastest three contestants that press the switches to be displayed. The result of any other contestant who presses the push switch is not displayed after the first three contestants or after the given time [3].

The type of liquid crystal display used is 16x2 LCD Character, and which has special features [4]. AT89C52 microcontroller is the type used also in this design and construction work [5].

The block diagram of the circuit used is shown below:

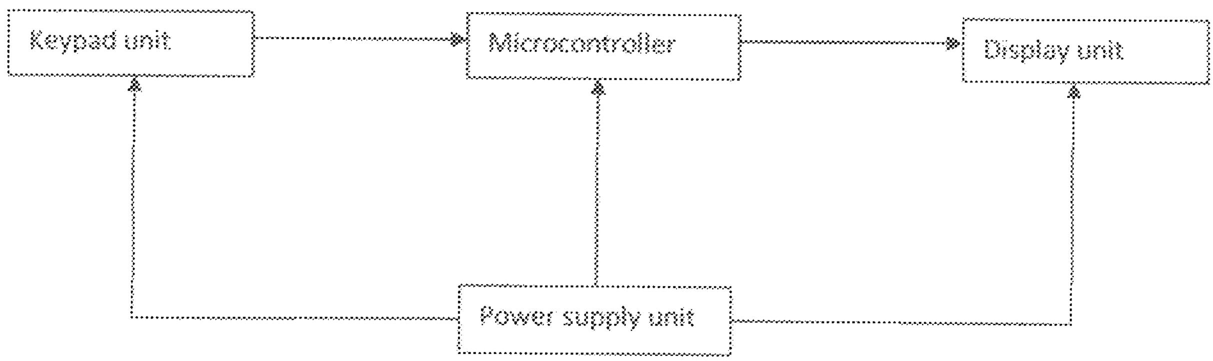


Fig. 2.1 Block Diagram of Quiz Control System

2.2 Historical Background

According to the ABC TELEVISION NETWORK of the UNITED STATES OF AMERICA (USA), the FASTEST FINGER device was first used in the year August 16, 1999. This was used on a television program called WHO WANTS TO BE A MILLIONAIRE [6]. Subsequently, other people started doing similar devices which are mostly used in quiz or competition.

One of these devices called QUIZ PROJECT was designed and constructed by John Hewes, used in quiz. It consists of four 555 timer bistables, LEDs, trigger push switches, resistors and diodes. In his device, only four contestants are involved and one as the winner [1]. FASTEST-FINGER FIRST is another type of the aforementioned devices. It is used in quiz for only three contestants or teams, and one contestant as the winner The main components used are: 555 and 556 timers, LEDs [7]. FASTEST FINGER FIRST was also constructed by another person, used in quiz. In his device, an AT89C51 microcontroller was used. The microcontroller was programmed to do the selection. The device was designed for only seven contestants, and one as the winner at the end of the quiz [3].

Furthermore, EMBEDDED QUIZ MONITOR FOR BUZZER ROUND was also used as a device for monitoring quiz. It uses an AT89C51 microcontroller and a 16x2 LCD. It can only be used for four contestants and one winner at the end of the quiz. The microcontroller is programmed to enable recognise the first to press the button, and the result is displayed on the LCD [2].

2.3 Previous Works of Others

Some similar works to QUIZ CONTROL SYSTEM have been done by other people, most of which are called FASTEST FINGER [3, 8]. This device is mostly used in WHO WANTS TO BE A MILLIONAIRE, started in USA. The device is used in choosing a specific number of people from group contestants.

2.3.1 Fastest- Finger First

This project is used for a quiz with not more than 3 contestants (or teams). This device makes use of three 555 timer bistables, light emitting diodes, and a bleeper as its main components. Each contestant has a trigger push switch and a LED. When a trigger switch is pressed, it lights the corresponding LED, sounds the bleeper and prevent other switches from working, therefore showing which contestant was first to press his switch. A reset push switch (operated by the quizmaster) cancels the bleeper and switches off the LED so that the circuit is ready for the next question. In this system, only one contestant is chosen at the end of the quiz [7].

Also, if two events appear to occur almost simultaneously, it becomes quite difficult to decide which one occurred first. The 555 timer bistables are triggered or reset when their inputs are low. Their reset inputs are connected together (pin 2) through a 0.1μF capacitor so that only the initial press triggers the bistable; continuing to hold the switch closed will have no effect [7].

2.3.2 Multi-User Jeopardy Game

This project describes a "FASTEST FINGER FIRST" system cheap to build, has high noise immunity, reliable and also extensible. The unit can be built in multiple of2 users up to 8 users on a single PCB. Additional PCBs can be properly chained on to allow 10 to 16, 18 to 24 users etc. There is no design limit, though in practice signal degradation will kick in at some stage [8].

This mechanism ignores noise spikes and bounces on any button line. and completely blocks other users once the first user has pressed his button. In this device only one of the contestants is chosen at the end of the quiz. The main components used in this device are: diodes, buzzer, CMOS logic, LEDs etc. [8]

2.3.3 Fastest Finger First Using 89C51 Microcontroller

This device is useful for quiz, games, and dumb charades. It displays the player number along with a beep sound when a specific player presses the button before others. It is designed for only 7 players and one as the winner. It can be constructed easily using minimal hardware and software. When the microcontroller receives an input from any of the 7 push switches (S1 to S7), it disables the input from reset of the switches assigned to other players. A master RESET switch has to be pressed to enable all the inputs for the next round of quiz [3].

The MICROCONTROLLER continuously waits for the input from the button switch s0 at port 0. When the microcontroller detects closure of switch s0, it starts checking the inputs from other switches or players. On detecting an input, it checks which switch (corresponding to a player) has been pressed. The player number information relating to each input is already stored on the flash memory of the microcontroller. If a specific match is found, the program control shifts the corresponding data into output of port-1 buffer. The information is instantly displayed on LT542. During this time, if any other switch is also pressed, there will be a beep sound but the microcontroller will not recognise the request unless the RESET button (s0) is pressed once again [3].

2.3.4 Embedded Quiz Monitor for Buzzer Round Using 89C51 Microcontroller

In the BUZZER ROUND of quiz contests, the question is thrown open to all the teams. The person who knows the answer hits the buzzer first and then answers the question. Sometimes, two or more players hit the buzzer almost simultaneously and it is very difficult to detect which of them has pressed the buzzer first. On television shows where the whole event is recorded, the actions are replayed in slow motion to detect the first hit. Such slow motions are possible only where huge funds are available to conduct the show. For this reason, buzzer rounds are avoided for quiz contests held colleges.

This device is an ELECTRONIC QUIZ BUZZER that is affordable by colleges and even individuals. The device is useful for a 4-teams quiz contests and one winner at the end of the quiz. This system is sensible. The circuit can detect and record the first hit of the contestant among all the contestants by saving marks of all participants on the microcontroller register per QUIZ MASTER guidance. It uses 89C51 microcontroller, and 16x2 LCD is provided to alert the participants for the question and to display who is answering for the question. Finally, the quizmaster can close the contest and he can display the marks obtained by individual team and winner details [2].

CHAPTER THREE

DESIGN AND IMPLEMENTATION

By considering the explanation in the previous chapter, that is chapter two, it is important to analyse each section of the project work for proper design and implementation.

In that case, the design and construction were done by the following methods:

3.1 Design of Power Supply Unit

A power supply is a device that supplies electrical energy to one or more electric loads [9]. A basic power supply is frequently supplemented by additional circuitry to control the output voltage, improve output voltage regulation, lower power supply impedance seen by the load, or establish other desirable characteristic [10]. In the world of electronic devices and circuits, most of the electronic circuits require a dc source for their operations [11]. The dc power supply is responsible for the conversion of the standard 230V, 50Hz ac available at all wall outlets into a constant dc voltage. This is one of the most common electronic circuits that you will find. The dc voltage produced by power supply is used to supply power to all types of electronic circuits, such as radio set, television receivers, computers and laboratory equipment [12]. A power supply can by broken down into a series of blocks, each of which performs a particular function [13]. A typical dc power supply consists of five stages as shown in fig. 3.1. In that case, the ICs and other components used, run on a power supply of 5V and 12V, hence the supply must be regulated to prevent fluctuation in the voltage level for proper supply of electricity. A battery can also be used in as a source of the power supply.

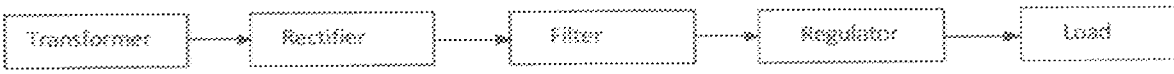


Fig. 3.1Typical dc power supply

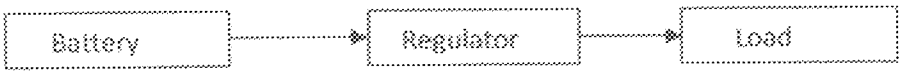


Fig. 3.2 Typical battery block diagram

These blocks can be briefly explained as follows:

(i.) **Transformer:** transformer plays an important role in electronic circuits, changes the voltage of an alternating current from one value to another. It is used to either step up or (mostly) step down the ac supply voltage to suit the requirement of the solid state electronic devices and circuits fed by the dc power supply [11].

(ii.) **Rectifier:** it is an electronic circuit which consists of one or more diodes to convert ac voltage into pulsating dc voltage. A rectifier is an electrical device that converts alternating current (AC), which periodically reverses direction, to direct current (DC), current that flows in only one direction, a process known as rectification [14]. Rectifiers can either be a half-wave rectifier or full-wave rectifier [11]. In this design a full-wave rectifier was employed in the conversion.

(iii.) **Filter:** the function of filter in this circuit element is to remove the fluctuations or pulsations (called ripples) present in the output voltage supplied by the rectifier. This is achieved by connecting a capacitor filter to the rectifier circuit in order to remove the pulsation [15].

(iv.) **Voltage Regulator:** the main function of voltage regulator is to keep the terminal voltage of the dc supply constant even when the ac input line voltage to the transformer or the load varies [14].

(v.) **Load:** the load block is usually a circuit for which the power supply is used in producing the dc voltage and load current [11].

Hence, the complete circuit diagram of the power supply unit is as shown below

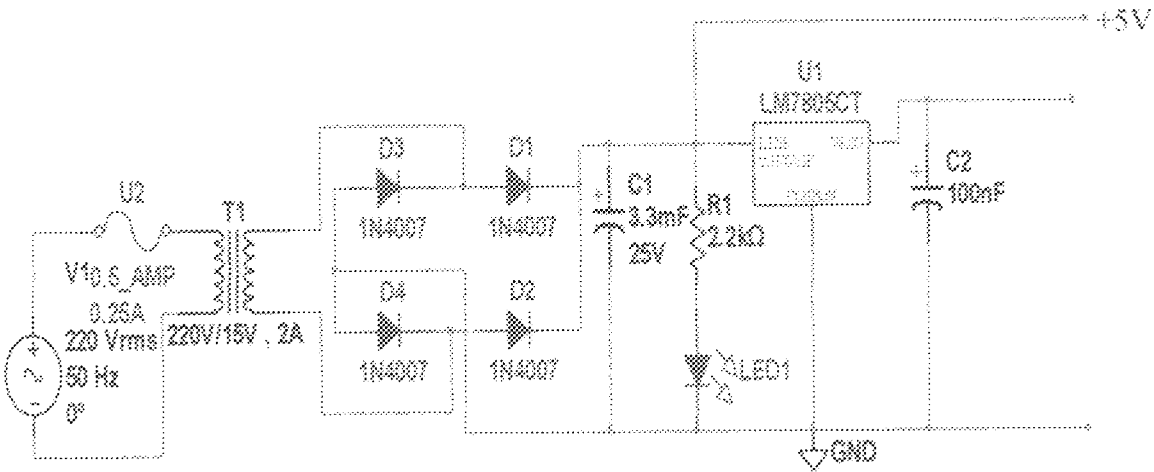
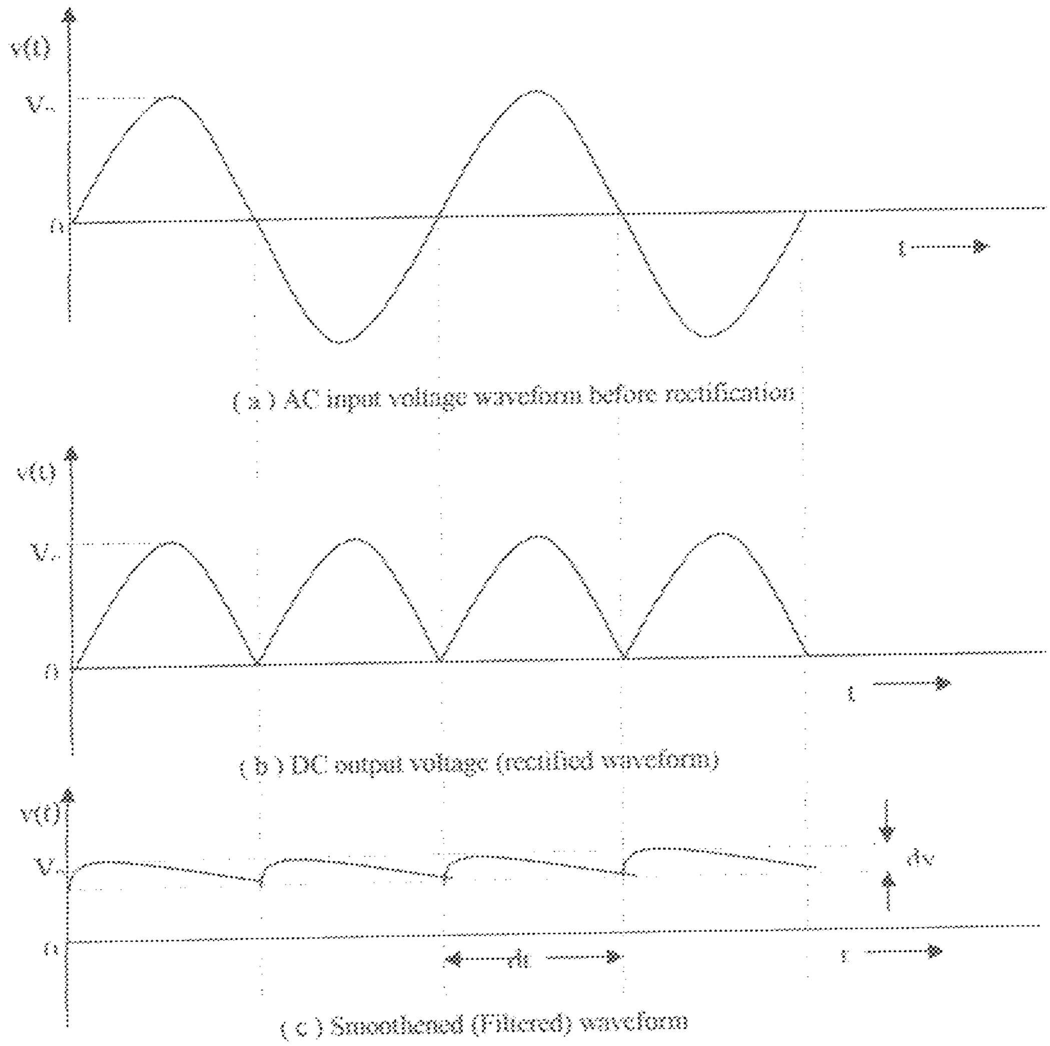


Fig. 3.3 Power Supply Unit Circuit Diagram

The main voltage of 220V is stepped down by a 220V/15V, 2A transformer. It is then rectified by full wave bridge diode rectifier. The waveform at this stage has no negative component, but a lot of ripples. Smoothing capacitors are needed to reduce the ripple to an acceptable level. The resulting ripple voltage (dv) can be calculated by considering the waveforms given in fig. 3.3.



(a) AC input voltage waveform before rectification

(b) DC output voltage (rectified waveform)

(c) Smoothened (Filtered) waveform

Fig. 3.3 Power supply waveforms

The load causes the capacitor to discharge between half cycles. If the load current stays constant, as it will for small ripple, then

$$I = C\frac{dV}{dt}$$

The frequency of the full wave signal is double the input frequency. This makes sense. A full wave output has twice as many cycles as the sine wave input has. The full wave rectifier inverts each negative half cycle, so that we get double the number of positive half cycles. This will result in doubling the frequency [16].

Therefore, the output frequency of the full wave rectifier is:

$f_{out} = 2f_{in}$

This implies that,

$$dt = \frac{1}{2f_{in}} = \frac{1}{(2*50)} = 0.01s$$

This is on the safe side, as the capacitor begins charging up in less than half a cycle.

The maximum current that can be drawn by the main circuit is determined by the voltage regulator following the filtering capacitor, the 7805.

The standard 7800 series can produce output current in excess of 1A when used with adequate heat sink [11]. Therefore, it can supply a maximum of 1A. This current will be drawn from the supply. Thus I_{load} = 1A (maximum). The value of C can then be calculated from

$$C = \frac{Idt}{dv}$$

But generally dv, which is the ripple voltage, is chosen to be 25% of V_p, where V_p is the peak voltage.

Therefore, $V_p = V_{rms}\sqrt{2}$

Where V_{rms} =15V, since the transformer of 220V/15V was used.

$$\Rightarrow V_p = 15*\sqrt{2} = 21.21V$$

For bridge rectifier $V_{P(out)} = V_{P(in)} - 1.4V$, since 0.7V dropped across a diode whenever it conducts. And it is only two diodes that will conduct at a time [16].

Therefore, $V_{P(out)} = 15*\sqrt{2} - 1.4 = 19.81V$

$$\Rightarrow dv = (\frac{25}{100})*19.81 = 4.95V$$

$$\text{Therefore, } C = \frac{(1*0.01)}{4.95} = 2.0202*10^{-3}F$$

$$= 2{,}020\mu F$$

So, the commercial value of 3,300µF, 25V was used in order to reduce the ripple to the nearest minimum. Then, the expected ripple voltage using this value of capacitor is

$$dv = \frac{(1*0.01)}{3{,}300*10^{-6}} = 3.03mV$$

This means that the output waveform goes from a peak value of 19.81V to (19.81-3.03*10^{-6}) =19.81V. It may be noted that the input voltage to the IC regulator must be at least 2V above the output voltage. This is required in order to maintain regulation.

Therefore, the peak value of 19.81V to 19.81V is acceptable since the output voltage is 5V. The ripple is neglected by the 7805 to a negligible value.

The average voltage going to 7805 is calculated by

$$V_P - 0.5dv = 19.81-(0.5*3.03*10^{-6}) = 19.81V$$

The output from the 7805 is 5V, at maximum current output of 1A. A heat sink is necessary and was screwed on to its back. The output remains constant in spite of input voltage variation.

The output capacitor 0.1µF, connected across the output acts as line filter to improve the transient response [12, 20].

For power indication, a light emitting diode (LED1) is connected from the positive supply line immediately after the capacitor to ground through a resistor. The resistor value is determined by the current carrying capacity of the diode. A typical red LED will drop 1.7V cathode to anode when forward biased (positive anode-to-cathode voltage) and will illuminate with 10 to 20mA flowing through it. Since the red LED is used as an indicator, then the required limiting resistor can be calculated as:

$$19.81 = V_d + I_d R = 1.7 + 10mA * R$$

$$\text{Therefore, } R = \frac{19.81}{10 * 10^{-3} A} = 1.981 k\Omega$$

The commercial value of 2.2kΩ was used in the design.

The diode and resistor served as a path to ground which the smoothing capacitor can discharge after the supply has been turned off. This prevents high voltages that might damage other parts of the circuit.

3.2 Design of the Keypads Unit (Trigger Push Switches)

This unit made provision for the selection of timing suitable for all participants in the quiz. This enables the quiz master to reset and start the device, and each of the participants to press the switch in his possession during the quiz.

3.2.1 Calculation of Pull-up Resistors

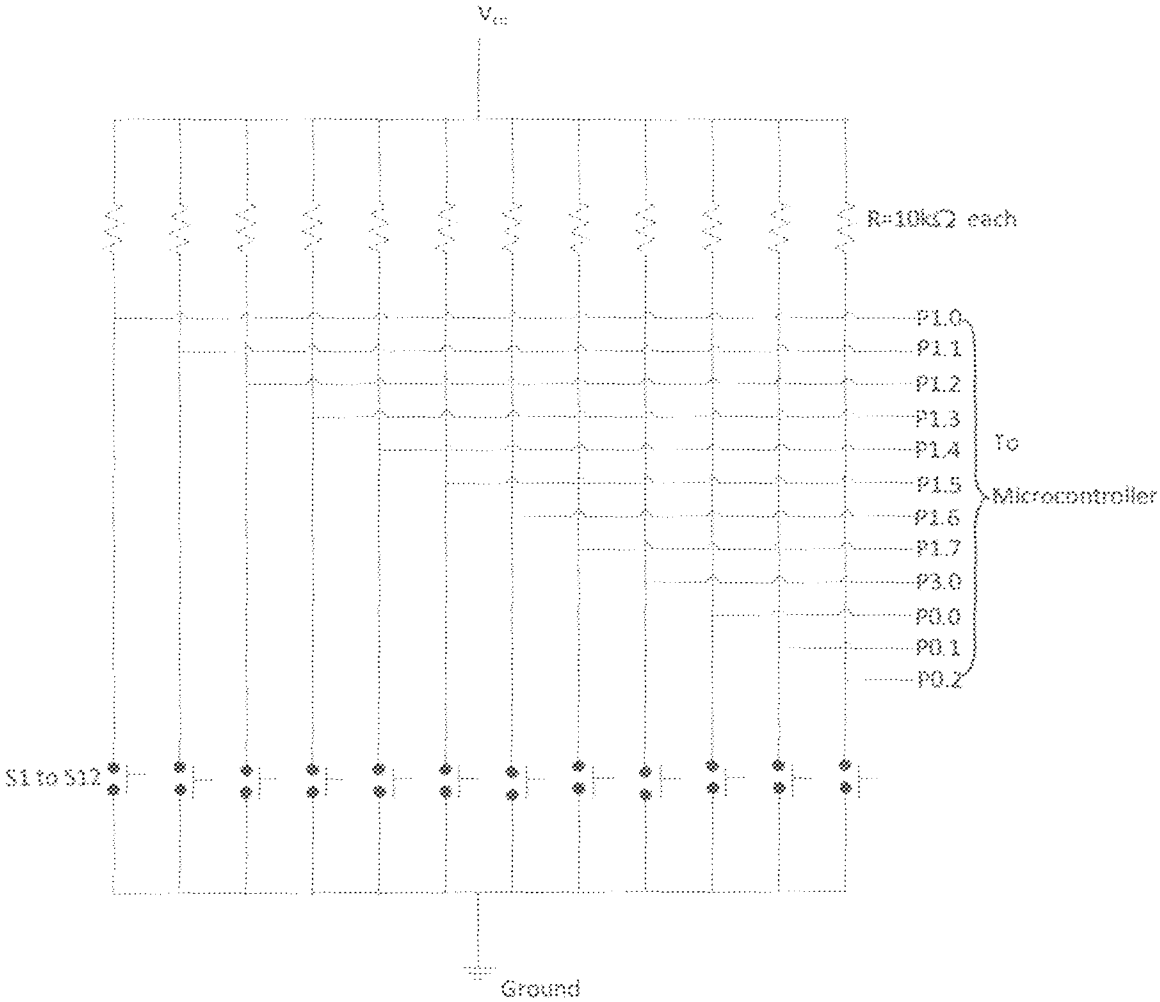


Fig. 3.4 Circuit diagram of keypad unit

Fig.3.4 shows the circuit diagram of the input trigger push switching unit. The circuit consists of eleven (12) push-to-on switches, S1 through S12. Each switch is connected from the pins of microcontroller to the ground. Every microcontroller has ability to sink or source current. The type of microcontroller used in this design is AT89S51.From the datasheet of AT89C52 [5], port1, 2 and 3 can sink current of about 1.6mA. Therefore, by taking the sinking current to be 0.5mA, the pull-up resistors can be calculated as follows:

From fig.3.4, V_{cc} = IR,

Where V_{cc} = +5V, I = 0.5mA, R = pull-up resistor.

This implies that, $R = \frac{Vcc}{I} = \frac{5V}{0.5mA} = 10k\,\Omega$

3.2.2 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 device is

manufactured using Atmel's high-density non-volatile memory technology and is compatible

with the industry-standard 80C51 and 80C52 instruction set and pin out. The on-chip flash

allows the program memory to be reprogrammed in-system or by a conventional non-volatile

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.

The connected pins to the switches are held HIGH by the pull-up resistors. The input/output ports of the AT microcontroller can be both read and written to (bidirectional). This implies that if any of the switches after being powered is pressed after the completion of any of the contestants the, the concerned pin of the microcontroller will be pulled down to ground which serves as the input to the microcontroller. When microcontroller senses a signal from that pin, it will start executing some specific instructions as directed by the software programming. The reset switch is always active at every point in time once the schedule has been chosen.

3.3 Design of Microcontroller Unit

One of the most important components used in this device design is the AT89C52 microcontroller which controls, coordinates and directs all the activities and behaviours of the design. Most control applications required extensive input/output and need to work with individual bits.The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8Kbytes of Flash programmable and erasable read only memory (PEROM). The on-chip Flash allows the program memory to be reprogrammed in-system or by conventional nonvolatile memory programmer [17]. The AT89C52 addresses both of these by having 32 input/output bit manipulation and bit checking.

The input from timing unit to the microcontroller automatically selects the corresponding timing programmed within the microcontroller chip. The output of this goes to the liquid crystal display (LCD). Fig.3.5 shows the unit circuit diagram, 1μF is connected from V_{cc} to the reset pin of microcontroller.

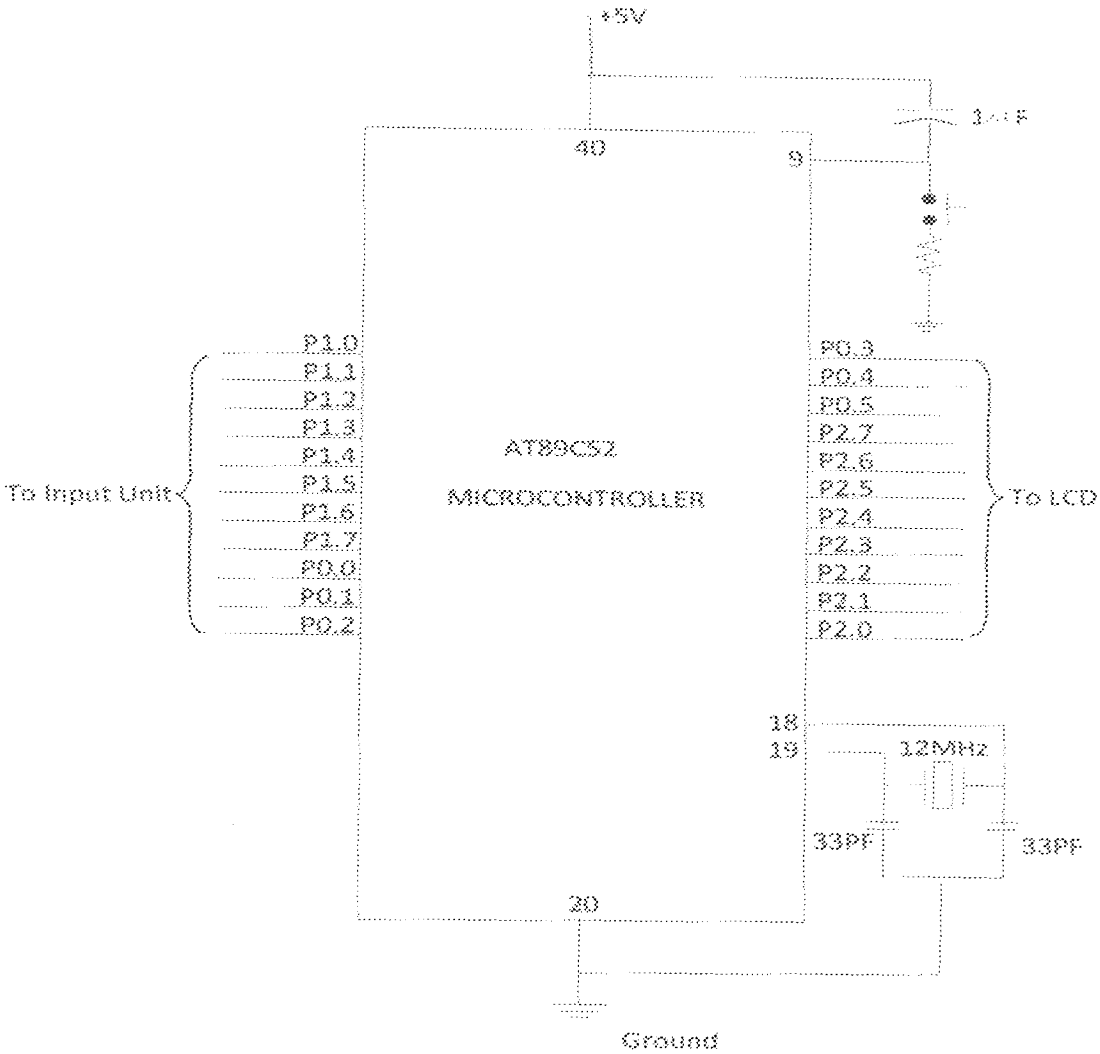


Fig. 3.5 Circuit diagram of Microcontroller unit

3.3.1 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

3.3.2 Oscillator Characteristic

A crystal oscillator is an electronic oscillator circuit that uses the mechanical resonance of a vibrating crystal of piezoelectric material to create an electrical signal with a very precise frequency [18]; it oscillates when excited [19],as shown in Fig. 3.5. Either a quartz crystal or ceramic resonator may be used. In this design, a quartz crystal was used.

From the datasheet of AT89C52 [5], it is noted that

C1, C2 = 30 pF ± 10 pF for Crystals

= 40 pF ± 10 pF for Ceramic Resonators

Therefore, since crystal was used, 33pF was chosen for both C1 and C2 as shown in fig. 3.5

3.4 Design of the display unit

The liquid crystal display is the device responsible for displaying the outputs of the contestants. It is a thin, flat electronic visual display that uses light modulating properties of liquid crystals (LCs) [1]. The 16 x 2 liquid crystal display used in this design consists of 16 columns and 2 rows screen resolution properties (pixels) [4].

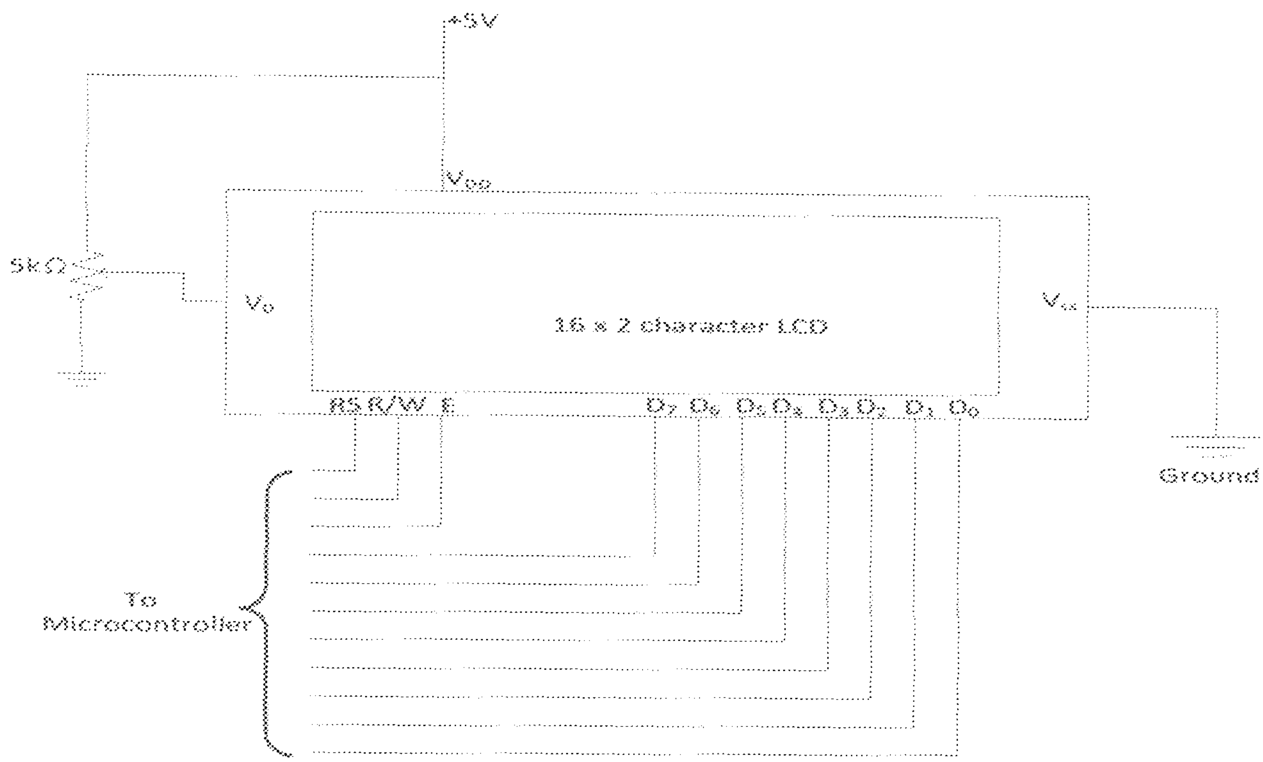


Fig 3.6 Circuit diagram of 16x2 LCD

Table 3.1 Interface Pin Function

Pin No.	Symbol	Level	Description
1	V_{ss}	0V	Ground
2	V_{DD}	5.0V	Supply Voltage for Logic
3	VO	Variable	Operating voltage for LCD
4	RS	H/L	H: DATA, L; Instruction code
5	R/W	H/L	H: Read(MPU-Module) L: Write(MPU-Module)
6	E	H, H-L	Chip enable signal
7	DBO	H/L	Data bit 0
8	DB1	H/L	Data bit 1
9	DB2	H/L	Data bit 2
10	DB3	H/L	Data bit 3
11	DB4	H/L	Data bit 4
12	DB5	H/L	Data bit 5
13	DB6	H/L	Data bit 6
14	DB7	H/L	Data bit 7
15	A	-	LED +
16	K	-	LED -

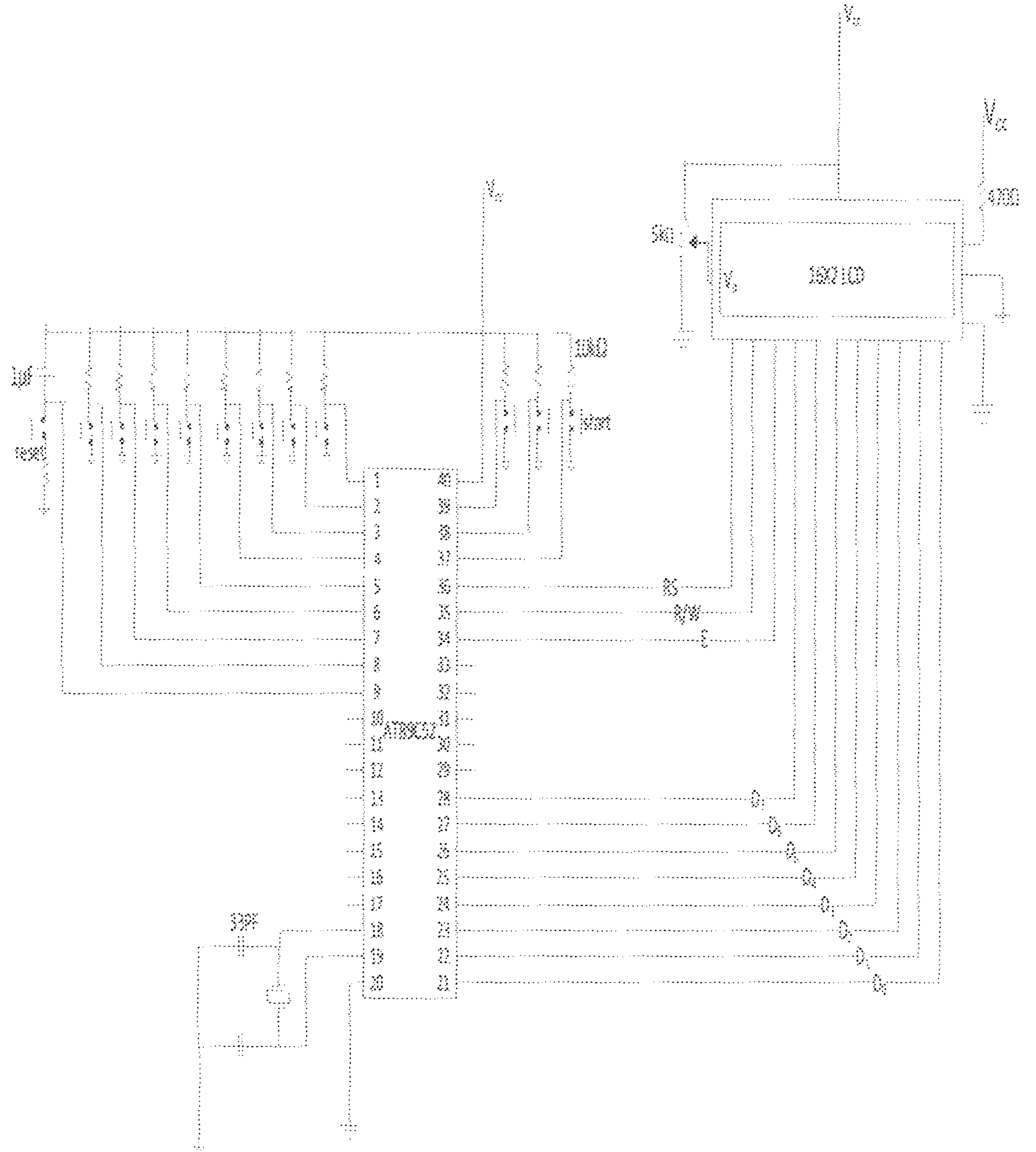


Figure 3.7 Complete circuit diagram of quiz control system

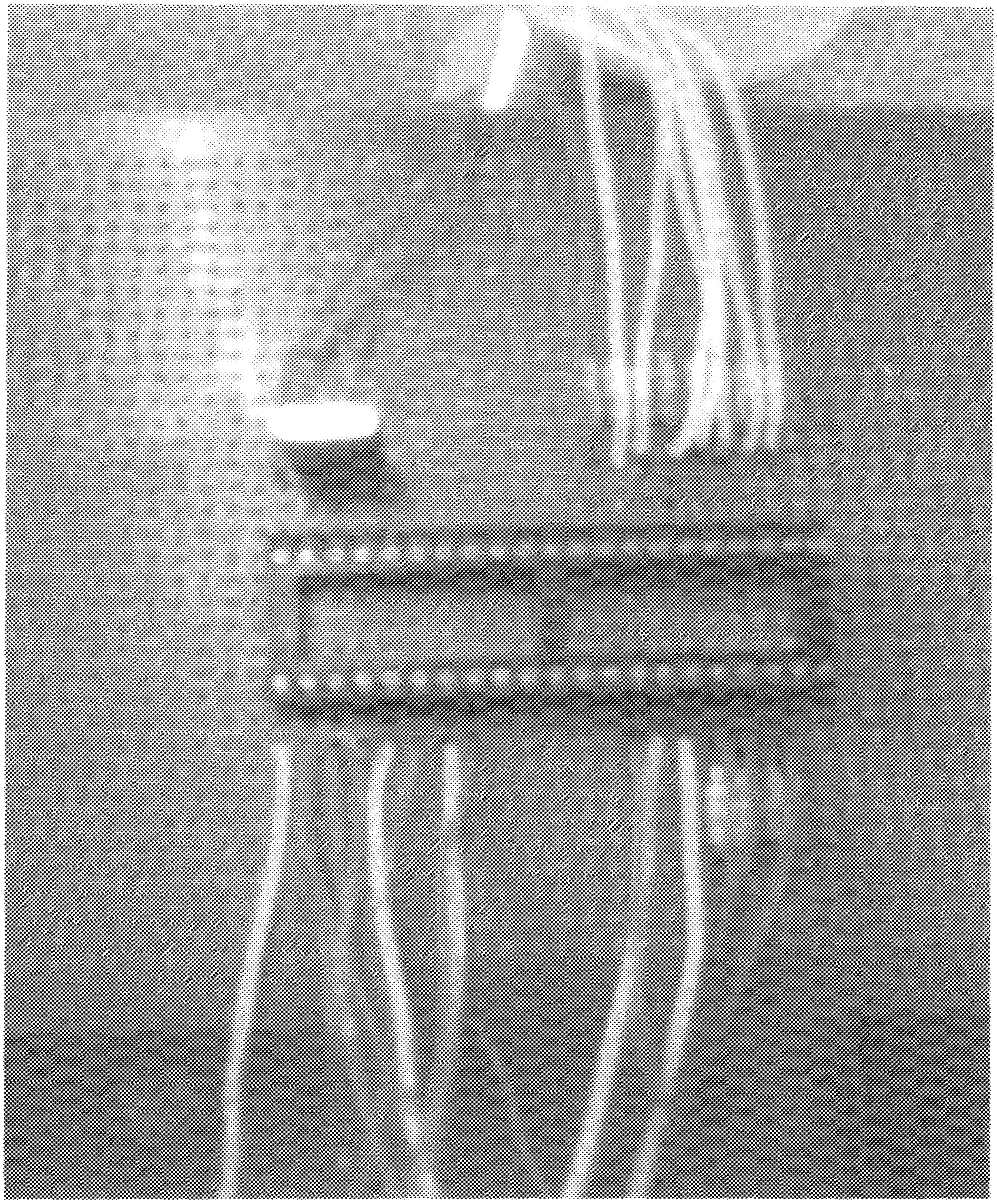


Figure 3.8 Construction of microcontroller socket

CHAPTER FOUR

TESTS, RESULTS, AND DISCUSSION

4.1 Tests

In the process of construction of this device, various forms of tests were done on the components used for the built. Most of the tests carried out were done using multi-meter while others were examined by the inscription on them in order to obtain the exact values needed for the work.

Pins of the programmable AT89C52 microcontroller integrated circuit (IC) before being programmed were properly checked to ensure that they are straight so as not to be rejected by the programmer. Firstly, the source codes were compiled on a "µvision 3" and test ran. During the process of compilation, proper concentration was given to the codes, to avoid any logic error. The generated hex-file was then transferred to the chip (microcontroller) with aid of a programmer.

A 9v dc source battery was connected to a voltage regulator for proper voltage needed by the circuit. Test on the voltage regulator was done; the actual voltage needed was obtained and noted. Each of the push switches was also tested by pressing and releasing them. The voltage on each switch was observed and recorded when the port pins of the microcontroller were normally high and active.

The proper contrast of the LCD was tested and set with the aid of variable resistor used in the circuit, this is necessary in obtaining the best output of data on the LCD screen. Before the permanent soldering of the system, the microcontroller and the other components used were connected on a "project board" and tested, making sure they function properly. The connections were done using connecting wires. Each stage of the construction was tested to see its effectiveness and necessary adjustments were made.

4.2 Results obtained

The following results were obtained and some tabulated as shown below during testing of the device, by careful observation.

Table 4.1 Results obtained during testing

S/N	Resistor (Ω)	Opened push switch (v)	Closed push switch (v)
1	10	4.92	0.00
2	10	4.92	0.00
3	10	4.92	0.00
4	10	4.92	0.00
5	10	4.92	0.00
6	10	4.92	0.00
7	10	4.92	0.00
8	10	4.92	0.00
9	10	4.92	0.00
10	10	4.92	0.00

Battery used: 9v

Voltage require: 5v

Measured voltage on regulator: 4.92v

Other components tested are: 470Ω resistor, crystal oscillator, 10µF10Ω capacitor, 33PF capacitor, and 5kΩ variable resistor.

4.3 Discussion of the Results

Various deductions could be made when the results obtained in section 4.2 are being looked at. The explanations of the results are thus:

The voltage measured on each of the push switches when opened was 4.92v, but when closed the measured voltage obtained was 0v on each switch. The port pins of the microcontroller work when they are active low, that is "0" triggers the pins. Initially, the pins were normally high; pressing any of them makes it "active low". That is why the voltage measured on the push switches when closed was 0v.

The voltage required by the circuit was 5v, obtained by the used of voltage regulator from a battery of 9v. The voltage regulator is used in getting required for the circuit because the microcontroller and the LCD operate at that voltage (5v). In getting a better contrast of the LCD, the variable resistor was being adjusted. It was observed that at some instances when adjusting of the variable resistor, the LCD was dull, and bright at some instances. The best adjustment was achieved when the texts on the LCD were properly seen. The 470Ω resistor helped in reducing the current flow through the LED of the LCD. When a resistor value higher than 470Ω was used, the LED did not display.

It was also observed that when the crystal oscillator was removed from the circuit, the microcontroller could not work. That was because there was no clock signal for the microcontroller to operate, crystal oscillator generates clock signal (frequency) for microcontrollers. The microcontroller could not reset when the 10μF10v capacitor was removed.

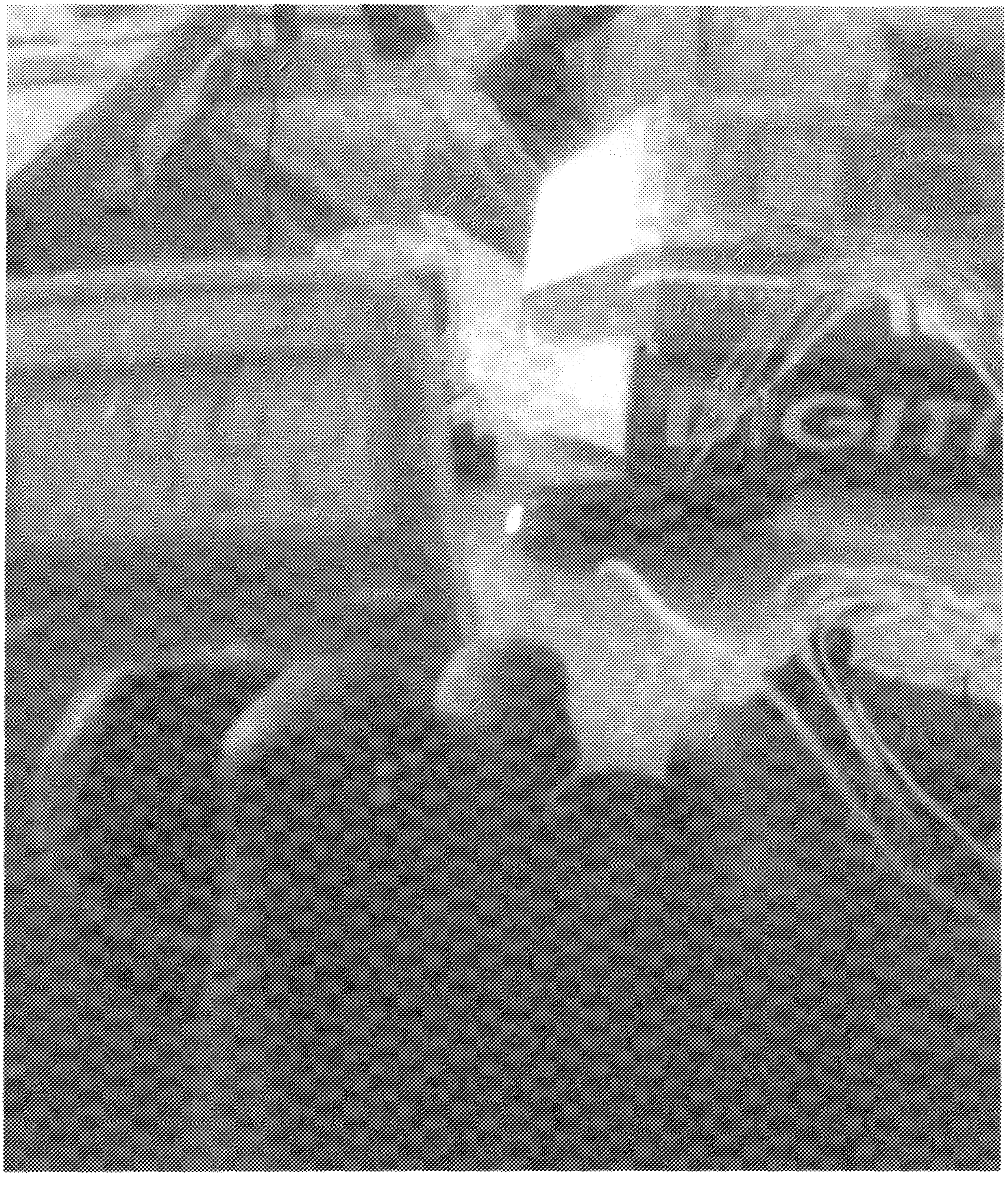


Figure 4.1 Voltage measured on opened trigger switches

Figure 4.2 measured voltage on closed trigger switches

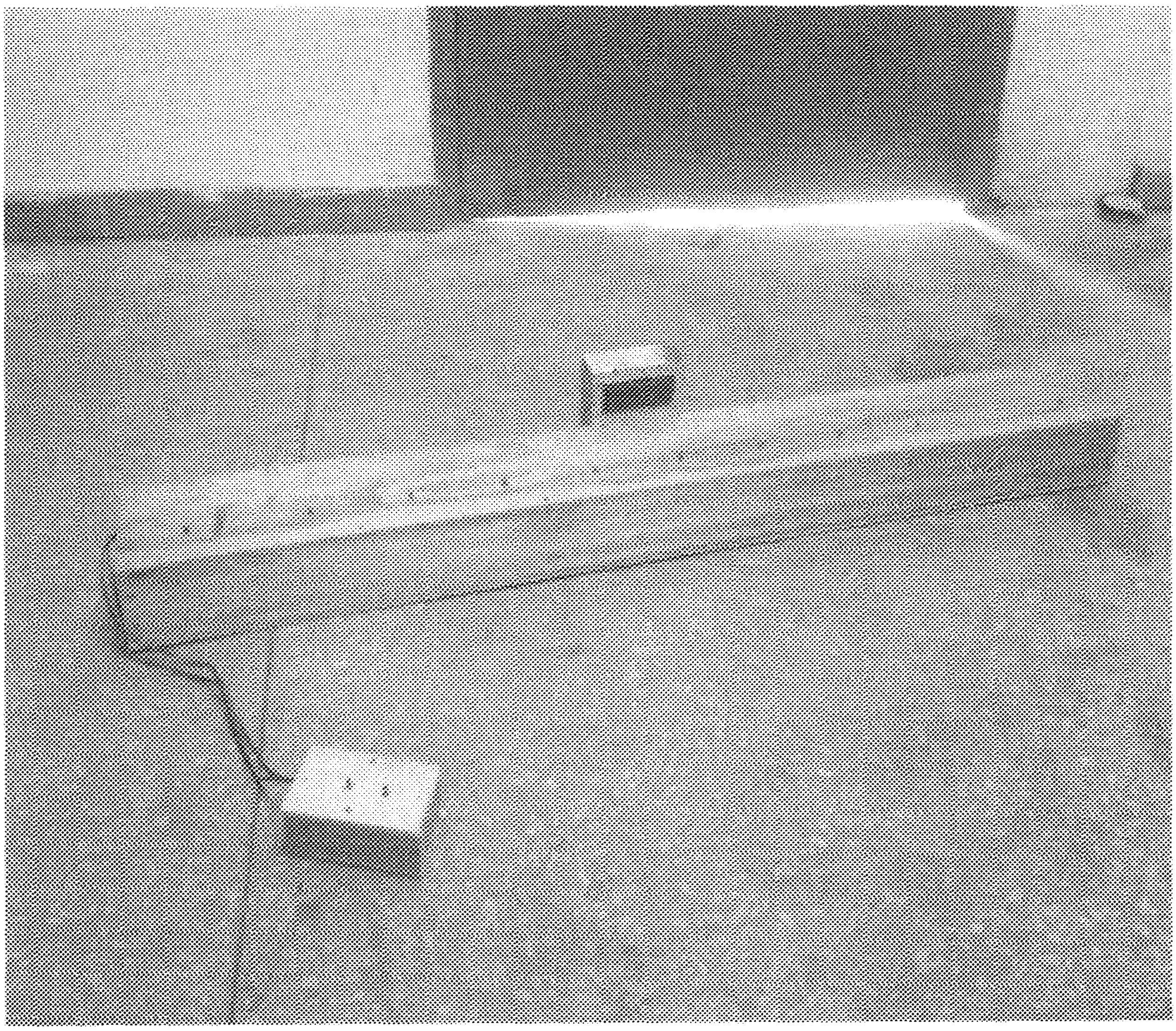


Figure 4.3 Device of Quiz Control System

4.3.1 Short Comings / Limitations

After the completion of this work, some limitations were observed and noted as shown below:

i. The device can only be used for ten contestants.

ii. It is not a wireless device.

iii. It can only be to access speed of contestants, but not accuracy.

iv. Capacity and of the LCD used was small which could only accept few data

4.3.2 Possible Remedy

Below are some of the suggestions made at the end of this in order to attain an optimum functionality and better output for the device, which are:

i. More number of contestants could be added by including more trigger push switches to the device.

ii. Wireless can be used.

iii. Large capacity of display system can be used in place of the small LCD.

CHAPTER FIVE

CONCLUSIONS

5.1 Conclusions

The basic design of the Quiz Control System which can be used by schools and colleges is a device which other functions can be added to. This is used in getting the three fastest contestants in a quiz by simply pressing the trigger push switches, their various positions are then displayed on the screen of the LCD as: 1st, 2nd, and 3rd. The Quizmaster has the start and reset push switches with him, he uses them to monitor the device. The reset push switch initializes the device while the start push switch makes it ready for use.

With the aid of the obtained results, it can be concluded at this point that the aim of the project work has been practically and theoretically achieved.

5.2 Possible Improvements

Other things can be added to this device in order to increase its performance, dynamism, and functionality.

i. By including a buzzer so as to keep the contestants ready.

ii. By using a rechargeable battery source instead of the non-rechargeable one, to maintain longer source of power.

5.3 Recommendations

Based on the features of this, the device can be used in the following recommended areas as:

i. It can be used mainly in primary and secondary schools during quiz.

ii. It can be used in a competition.

iii. It can also be used in choosing employees in work places.

REFERENCES

[1] www.kpsec.freeuk.com/project/quiz.htm, pp.1.

[2]www.wineyardtechnologies.com/tools/assets/wk206.pdf, pp.1.

[3] saitronics.org/saitronics/pdf/FastFingerFirst.pdf, pp.1.

[4] www.datasheetcatalog.com, pp1.

[5] www.embeddedi.co.th/downloads/sheets/ic/AT89C52.pdf, pp.1.

[6] enwikipedia.org/.../Who_Wants_to_Be_a_Millionaire_(US_game_show), pp1.

[7] www.indiastudychannel.com/.../1141_FASTEST_FINGER_FIRST.aspx_India, pp.1.

[8] www.desmith.net/NMds/Electronics/Jeopardy.html, pp.1.

[9] enwikipedia.org/wiki/Power_supply, pp.1.

[10] Electronic Principles and Circuits: An Introduction to Electronics for the Technician. William E. Long and Paul L. Evans, Foothill College Los Atlos Hills, California, pp.46-50.

[11] B.L Theraja and A.K Theraja, a Textbook Of Electrical Technology, S. Chand and Company Limited, Ram Nagar, New Delhi-110055, 2003, pp.332-336.

[12] Thomas L. Floyd, Electronic Fundamentals: Circuits Deices and Applications, 7th Edition. Published by Pearson Education, Upper Saddle River, New Jersey, 2007, pp.180-184.

[13] www.kpsec.freeuk.com/powersup.htm, pp.1.

[14] enwikipedia.org/wiki/Rectifier, pp.1.

[15] Electricity and Electronics by Howard H. Gerrish, Williams E . Dogger, Jr. Director Technology for all American Projects, pp.149-151.

[16] Malyino, A.P "Malyino Electronic Principles" 6th Edition, Published by Tala Mcgray-Hill Publishing Co. Ltd, Patel Nagar, New Delhi 11008, 1999, pp.221-223.

[17] www.atmel.com/Atmel/AT89C52.shtml, pp.1.

[18] enwikipedia.org/wiki/crystal_oscillator, pp.1.

[19] An Introduction to Modern Electronics by Williams L. Faissler, pp.207-209.

[20] Electronic Principles Physics, Models, and Circuits by Paul E. Gray and Cambell L. Searle, pp.216-217.

APPENDIX A

1.1 USER'S MANUAL

1.1.1 Usage

I. Turn ON the toggle switch of the device in order to power it

II. Press the RESET TRIGGER switch to initialize the device

III. Press the RESET TRIGGER switch to initialize the device

IV. Press the START TRIGGER switch to enable it ready for use

V. If there is any interruption during the quiz, press the RESET PUSH TRIGGER switch

1.1.2 Troubleshooting of the device

I. Check if BATTERY is well connected

II. Check if the SWITCH is at "ON" position

III. Check if the cables are not broken

IV. Check if the cables are still firmly SOLDERD on the board

V. Check if the BATTERY is not working by replacing it with a new one

VI. Adjust the variable resistor if the LCD is dull

APPENDIX B

1.3 SOURCE CODE

```
#include <reg52.h>
#define lcd_port P2
#define LCD_en  0x80
#define LCD_rs  0x20
//sbit LCD_D7 = P2^7;                    //LCD D7/Busy Flag
//sbit LCD_rw = P0^2;                    //LCD Read/Write
//sbit LCD_en = P0^3;                    //LCD Enable
sbit start  = P0^0;
sbit con1   = P0^1;
sbit con2   = P0^2;
sbit con3   = P1^0;
sbit con4   = P1^1;
sbit con5   = P1^2;
sbit con6   = P1^3;
sbit con7   = P1^4;
sbit con8   = P1^5;
sbit con9   = P1^6;
sbit con10  = P1^7;
unsigned int con1_attempt = 0;
unsigned int con2_attempt = 0;
unsigned int con3_attempt = 0;
unsigned int con4_attempt = 0;
unsigned int con5_attempt = 0;
unsigned int con6_attempt = 0;
unsigned int con7_attempt = 0;
unsigned int con8_attempt = 0;
unsigned int con9_attempt = 0;
unsigned int con10_attempt = 0;
unsigned int winner = 0;
void send_string(unsigned char*);
void lcd_cmd(unsigned char);


int wait_second()
  {
        unsigned int y;
              for(y = 0; y < 33000; y++)
                 {
                            if(con1 == 0 && winner < 3 && con1_attempt == 0)
                                    {
```

```
                con1_attempt++;
                        winner++;
                        if(winner == 1)
                          {
                            send_string("Result:");
                                    lcd_cmd(0x88);
                                    send_string("1 =con1");
                          }
                        else if(winner == 2)
                          {
                            //send_string("Result:");
                                    lcd_cmd(0xC0);
                                    send_string("2 =con1");
                          }
                        else if(winner == 3)
                          {
                            //send_string("Result:");
                                    lcd_cmd(0xC8);
                                    send_string("3 =con1");
                                    break;

                          }



        }
else if(con2 == 0 && winner < 3 && con2_attempt == 0)
  {
                con2_attempt++;
                        winner++;
                        if(winner == 1)
                          {
                            send_string("Result:");
                                    lcd_cmd(0x88);
                            send_string("1 =con2");
                          }
                        else if(winner == 2)
                          {
                            //send_string("Result:");
                                    lcd_cmd(0xC0);
                                    send_string("2 =con2");
                          }
                        else if(winner == 3)
                          {
                            lcd_cmd(0xC8);
                                    send_string("3 =con2");
```

```
                //break;
            }


    }
    else if(con3 == 0 && winner < 3 && con3_attempt == 0)
     {
        con3_attempt++;
            winner++;
            if(winner == 1)
             {
               send_string("Result:");
                    lcd_cmd(0x88);
                    send_string("1 =con3");
             }
            else if(winner == 2)
             {
               lcd_cmd(0xC0);
                    send_string("2 =con3");
             }
            else if(winner == 3)
             {
               lcd_cmd(0xC8);
                    send_string("3 =con3");
                    //break;
             }


    }
    else if(con4 == 0 && winner < 3 && con4_attempt == 0)
     {
        con4_attempt++;
            winner++;
            if(winner == 1)
             {
               send_string("Result:");
                    lcd_cmd(0x88);
                    send_string("1 =con4");
             }
            else if(winner == 2)
             {
               lcd_cmd(0xC0);
                    send_string("2 =con4");
             }
            else if(winner == 3)
```

```
                    {
                      lcd_cmd(0xC8);
                              send_string("3 =con4");
                              //break;
                    }


          }
      else if(con5 == 0 && winner < 3 && con5_attempt == 0)
        {
              con5_attempt++;
                   winner++;
                   if(winner == 1)
                    {
                      send_string("Result:");
                          lcd_cmd(0x88);
                          send_string("1 =con5");
                    }
                   else if(winner == 2)
                    {
                      lcd_cmd(0xC0);
                          send_string("2 =con5");
                    }
                   else if(winner == 3)
                    {
                      lcd_cmd(0xC8);
                          send_string("3 =con5");
                          //break;
                    }


          }
      else if(con6 == 0 && winner < 3 && con6_attempt == 0)
        {
              con6_attempt++;
                   winner++;
                   if(winner == 1)
                    {
                      send_string("Result:");
                          lcd_cmd(0x88);
                          send_string("1 =con6");
                    }
                   else if(winner == 2)
                    {
                      lcd_cmd(0xC0);
```

```
                send_string("2 =con6");
            }
            else if(winner == 3)
            {
              lcd_cmd(0xC8);
                send_string("3 =con6");
                //break;
            }


    }
    else if(con7 == 0 && winner < 3 && con7_attempt == 0)
    {
        con7_attempt++;
            winner++;
            if(winner == 1)
            {
              send_string("Result:");
                lcd_cmd(0x88);
                send_string("1 =con7");
            }
            else if(winner == 2)
            {
              lcd_cmd(0xC0);
                send_string("2 =con7");
            }
            else if(winner == 3)
            {
              lcd_cmd(0xC8);
                send_string("3 =con7");
                //break;
            }


    }
    else if(con8 == 0 && winner < 3 && con8_attempt == 0)
    {
        con8_attempt++;
            winner++;
            if(winner == 1)
            {
              send_string("Result:");
                lcd_cmd(0x88);
                send_string("1 =con8");
            }
```

```
                    else if(winner == 2)
                     {
                       lcd_cmd(0xC0);
                              send_string("2 =con8");
                     }
                    else if(winner == 3)
                     {
                       lcd_cmd(0xC8);
                              send_string("3 =con8");
                              //break;
                     }


            }
        else if(con9 == 0 && winner < 3 && con9_attempt == 0)
          {
                   con9_attempt++;
                        winner++;
                        if(winner == 1)
                         {
                           send_string("Result:");
                                  lcd_cmd(0x88);
                                  send_string("1 =con9");
                         }
                        else if(winner == 2)
                         {
                           lcd_cmd(0xC0);
                                  send_string("2 =con9");
                         }
                        else if(winner == 3)
                         {
                           lcd_cmd(0xC8);
                                  send_string("3 =con9");
                                  //break;
                         }


            }
        else if(con10 == 0 && winner < 3 && con10_attempt == 0)
          {
                   con10_attempt++;
                        winner++;
                        if(winner == 1)
                         {
                           send_string("Result:");
```

```
                lcd_cmd(0x88);
                send_string("1 =con10");

            }
            else if(winner == 2)
            {
                lcd_cmd(0xC0);
                    send_string("2 =con10");

            }
            else if(winner == 3)
            {
                lcd_cmd(0xC8);
                    send_string("3 =con10");
                    //break;

            }

        }
    }
    return(winner);

}
void LCD_busy()
{
    unsigned char i,j;
    for(i=0;i<50;i++)       //A simple for loop for delay
      for(j=0;j<255;j++);
}
void reset_LCD()
  {
      lcd_port = 0xFF;
    LCD_busy();
    lcd_port = 0x03+LCD_en;
    lcd_port = 0x03;
    LCD_busy();
    lcd_port = 0x03+LCD_en;
    lcd_port = 0x03;
    LCD_busy();
    lcd_port = 0x03+LCD_en;
    lcd_port = 0x03;
    LCD_busy();
    lcd_port = 0x02+LCD_en;
    lcd_port = 0x02;
    LCD_busy();
  }
/* ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

```
*** Send Command to LCD **********************************

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~*/
void lcd_cmd(unsigned char ch)
  {
     lcd_port = ((ch >> 4)&0x0F)|LCD_en;
    lcd_port = ((ch >> 4)&0x0F);

    lcd_port = (ch &0x0F)|LCD_en;
    lcd_port = (ch &0x0F);
            LCD_busy();
            LCD_busy();
    }
/*-------------------------------------------------------- ----

---- Initializing the LCD ------------------------------

------------------------------------------------------------*/
void LCD_init()
  {
            reset_LCD();      // Call LCD reset
    lcd_cmd(0x28);      // 4-bit mode - 2 line - 5x7 font.
    lcd_cmd(0x0C);      // Display no cursor - no blink.
    lcd_cmd(0x06);      // Automatic Increment - No Display shift.
    lcd_cmd(0x80);      // Address DDRAM with 0 offset 80h.
    }
/* .....................................................

***** Send Data to LCD ***************************

.......................................................*/

void send_data(unsigned char info)
  {
     lcd_port = (((info >> 4)&0x0F)|LCD_en|LCD_rs);
    lcd_port = (((info >> 4)&0x0F)|LCD_rs);

    lcd_port = ((info&0x0F)|LCD_en|LCD_rs);
    lcd_port = ((info&0x0F)|LCD_rs);

    LCD_busy();
            LCD_busy();
    }
/* ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

```

$$$$$$$$$$$    Send String to LCD      &&&&&&&&&&&&&&&&&&&&&&&&&&&&&

^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
void send_string(unsigned char *string)
{
      while(*string)
                  send_data(*string++);
}


void wait_a_second()
  {
         unsigned int n;
                  for(n = 0; n < 33000; n++);
      }
main()
  {
        unsigned int x,n;
                  LCD_init();
                  P1 = 0xFF;
                  con1 = 1;
                  con2 = 1;
                  start = 1;
                  lcd_cmd(0x01);
                  lcd_cmd(0x83);
                  send_string("Quiz Control");
                  lcd_cmd(0xC5);
                  send_string("System");
                  for(x = 0; x < 10; x++)
                       wait_a_second();
                  lcd_cmd(0x01);
                  lcd_cmd(0x80);


                   send_string("Press Start...");
                      while(start == 1);
                      lcd_cmd(0x01);
                      for(x = 0; x < 120; x++)
                      {
                                      wait_second();
                                      if(winner == 3)
                                        break;
                             }
                         for(n = 0; n < 5; n++)
```

```
                        wait_a_second();
                lcd_cmd(0x01);
                        //lcd_cmd(0x07);
                send_string("QUIZ END");


            return 0;
}
```