**Arduino based voting machine**

Introduction

Design principle

Circuit description

1. Power supply
2. Arduino UNO
3. LCD
4. Switch

Conclusion

**Introduction**

This project is all about Simple & Smart Electronic Voting Machine Using Arduino. The basic idea of this project is to create an electronic voting machine that will help to eradicate defrauding of the manual voting systems and prior versions of electronic voting. The system is provided with n number of the switch where n is the number of a political party. Here the voter will be allowed to proceed for choosing their preferred candidate from the panel of buttons. The final vote is then displayed onto an LCD for the satisfaction of voters. In the end, the result can be automatically calculated by pressing the result button.

**Materials Used**

Arduino UNO Board

16x2 LCD Display

Potentiometer 10k

Push Button Switch

Connecting Wires

PCB

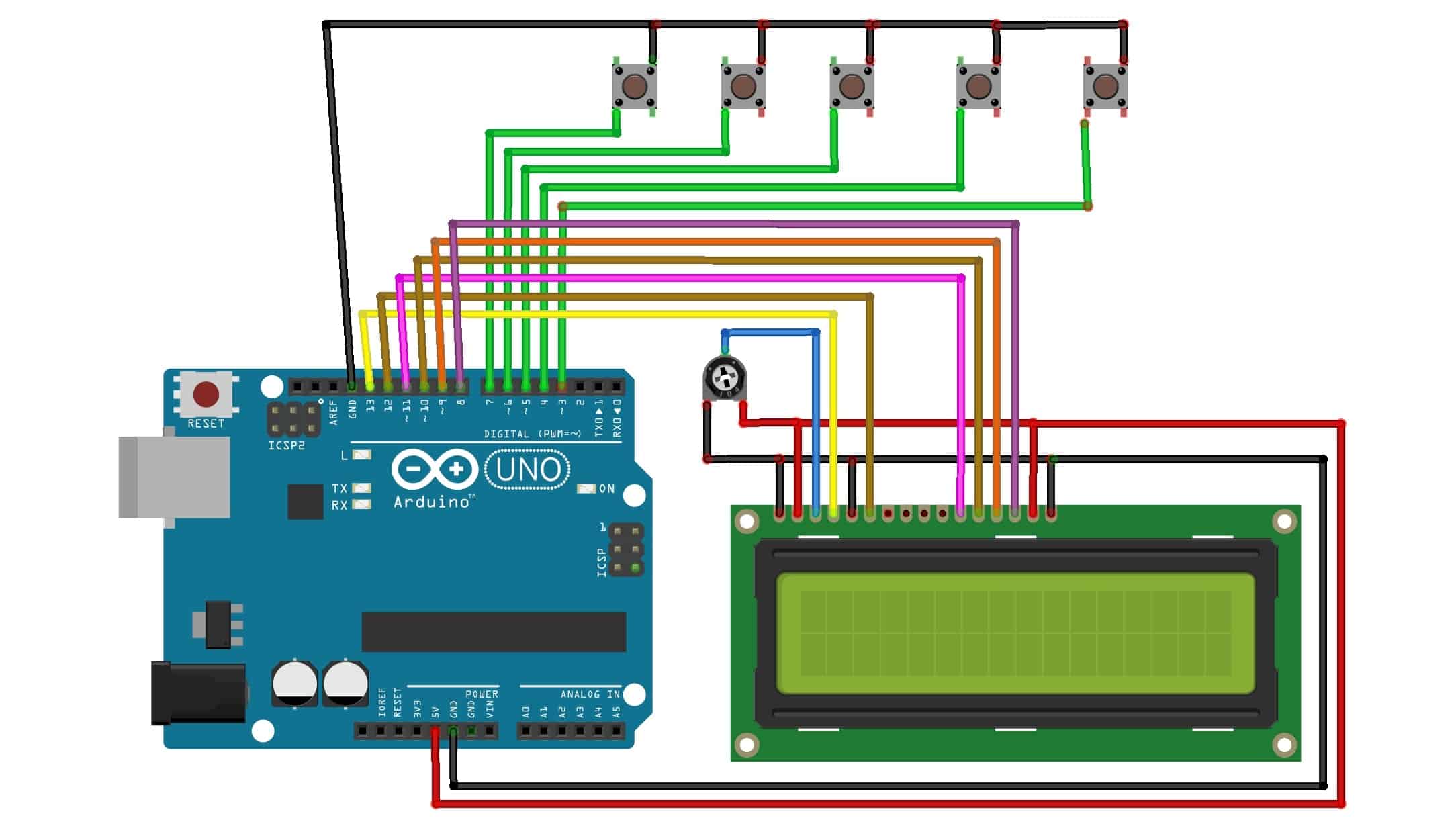
5V adaptor

**Working and design principle**

In this project, i.e Smart Electronic Voting Machine Using Arduino, we have used four pushbuttons for four different candidates who are taking part in the election. We can increase the number of the candidate as per requirement. When any voter press any of four buttons then respecting voting value will increment by one each time. After the whole voting process, the result button can be pressed to display the result.Here Arduino is the heart and brain of this system. Arduino controls the complete voting processes like reading button, incrementing vote value, generating a result, and sending vote and result in LCD Display.

Here we have added five buttons that are assigned for team A, team B, team C, team D, and the last button is used for calculating or displaying results.

**Circuit diagram**



**Circuit description**

**Power supply**

In this project the power supply required is very much precession and also requires different level of power supply. Basically the power supply used for the transmitter and receiver is arranged from a battery. Along with the battery the power supply requirement is +5Volts.

**Description**

The power supply designed for catering a fixed demand connected in this project. The basic requirement for designing a power supply is as follows,

1. The different voltage levels required for operating the devices. Here +5Volt required for operating microcontroller and other circuits etc.
2. The current requirement of each device or load must be added to estimate the final capacity of the power supply.

The power supply always specified with one or multiple voltage outputs along with a current capacity. As it is estimate the requirement of power is approximately as follows,

Out Put Voltage = +5Volt.

Capacity = 1000mA

The power supply is basically consisting of three sections as follows,

1. Step down section
2. Rectifier Section
3. Regulator section

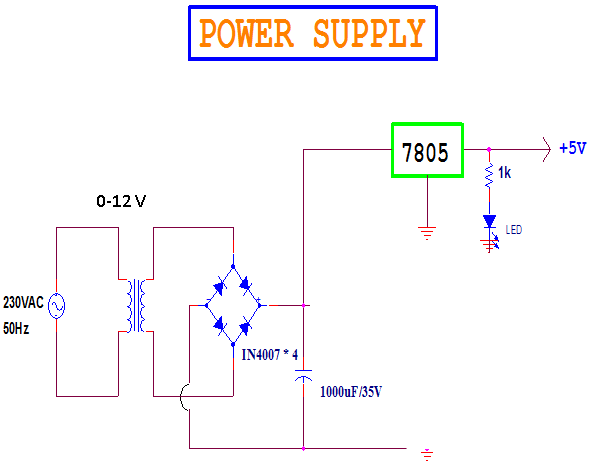
**Design principle:**

There are two methods for designing power supply, the average value method and peak value method. In case of small power supply peak value method is quit economical, for a particular value of DC output the input AC requirement is appreciably less. In this method the Dc output is approximately equal to Vm. The rectifier output is approximately charged to Vcc due to charging of the capacitor. The capacitance provides the backup during the discharge period. So, the value of the capacitor is calculated.

**Circuit connection: -** In this we are using Transformer (0-12) vac, 1Amp, IC 7805, diodes IN 4007, LED & resistors. Here 230V, 50 Hz ac signal is given as input to the primary of the transformer and the secondary of the transformer is given to the bridge rectification diode. The o/p of the diode is given as i/p to the IC regulator (7805 ) through capacitor (1000mf/35v). The o/p of the IC regulator is given to the LED through resistors.

**Circuit Explanations: -** When ac signal is given to the primary of the transformer, due to the magnetic effect of the coil magnetic flux is induced in the coil(primary) and transfer to the secondary coil of the transformer due to the transformer action.” Transformer is an electromechanical static device which transformer electrical energy from one coil to another without changing its frequency”. Here the diodes are connected in a bridge fashion. The secondary coil of the transformer is given to the bridge circuit for rectification purposes.

During the +ve cycle of the ac signal the diodes D2 & D4 conduct due to the forward bias of the diodes and diodes D1 & D3 does not conduct due to the reversed bias of the diodes. Similarly during the –ve cycle of the ac signal the diodes D1 & D3 conduct due to the forward bias of the diodes and the diodes D2 & D4 does not conduct due to reversed bias of the diodes. The output of the bridge rectifier is not a power dc along with rippled ac is also present. To overcome this effect, a capacitor is connected to the o/p of the diodes (D2 & D3). Which removes the unwanted ac signal and thus a pure dc is obtained. Here we need a fixed voltage, that’s for we are using IC regulators (7805).”Voltage regulation is a circuit that supplies a constant voltage regardless of changes in load current.” This IC’s are designed as fixed voltage regulators and with adequate heat sinking can deliver output current in excess of 1A. The o/p of the bridge rectifier is given as input to the IC regulator through capacitor with respect to GND and thus a fixed o/p is obtained. The o/p of the IC regulator (7805) is given to the LED for indication purpose through resistor. Due to the forward bias of the LED, the LED glows ON state, and the o/p are obtained from the pin no-3.



**Arduino UNO (Atmega328p)**

The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. Hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a boot loader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.



**Background**

The Arduino project started at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. At that time, the students used a BASIC Stamp microcontroller at a cost of $100, a considerable expense for many students. In 2003 Hernando Barragán created the development platform *Wiring* as a Master's thesis project at IDII, under the supervision of Massimo Banzi and Casey Reas, who are known for work on the Processing language. The project goal was to create simple, low-cost tools for creating digital projects by non-engineers. The Wiring platform consisted of a printed circuit board (PCB) with an ATmega168 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2003, Massimo Banzi, with David Mellis, another IDII student, and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked the project and renamed it *Arduino*. Early arduino boards used the FTDI USB-to-serial driver chip and an ATmega168. The Uno differed from all preceding boards by featuring the ATmega328P microcontroller and an ATmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

**Technical specifications**

Microcontroller : Microchip ATmega328P

Operating Voltage: 5 Volt

Input Voltage: 7 to 20 Volts

Digital I/O Pins: 14 (of which 6 provide PWM output)

Analog Input Pins: 6

DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA

Flash Memory: 32 KB of which 0.5 KB used by bootloader

SRAM: 2 KB

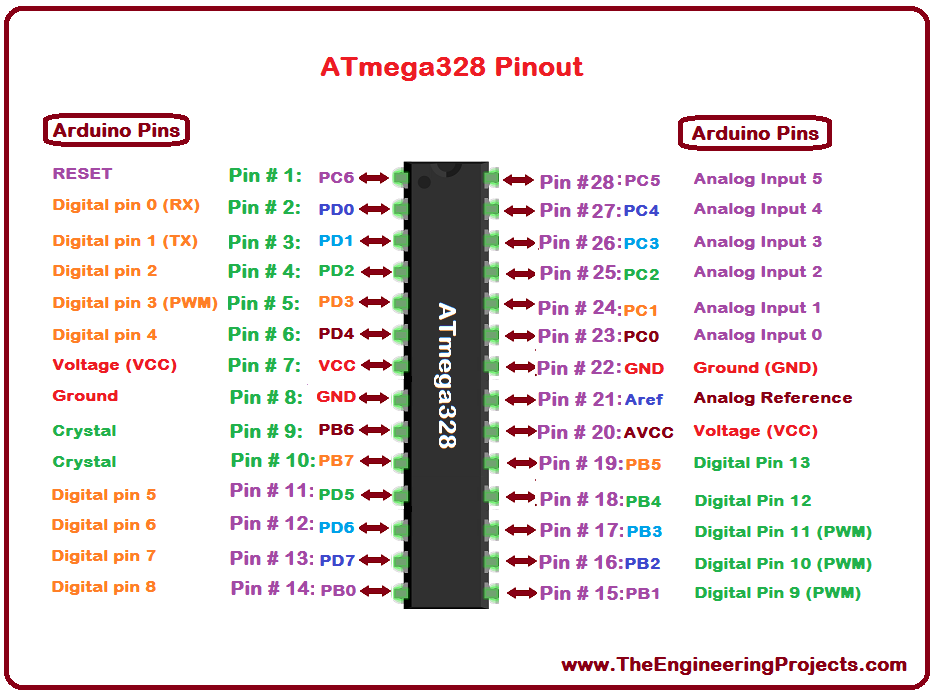
EEPROM: 1 KB

Clock Speed: 16 MHz

Length: 68.6 mm

Width: 53.4 mm

Weight: 25 g



**General Pin functions**

**LED**: There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

**VIN**: The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

**5V**: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.

**3V3**: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

**GND**: Ground pins.

**IOREF**: This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.

**Reset**: Typically used to add a reset button to shields which block the one on the board.

**Special Pin Functions**

Each of the 14 digital pins and 6 Analog pins on the Uno can be used as an input or output, using pin Mode(),digital Write(), and digital Read () functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference() function.

In addition, some pins have specialized functions:

**Serial**: pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

**External Interrupts**: pins 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

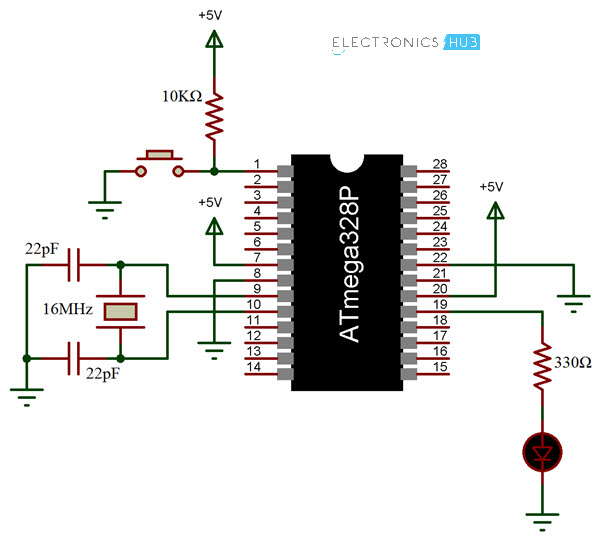
**PWM**(**P**ulse **W**idth **M**odulation) 3, 5, 6, 9, 10, and 11 Can provide 8-bit PWM output with the analog Write() function.

**SPI**(**S**erial **P**eripheral **I**nterface): 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

**TWI**(**T**wo **W**ire **I**nterface): A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

**AREF**(**A**nalog **Reference**): Reference voltage for the analog inputs.

The Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A SoftwareSerial library allows serial communication on any of the Uno's digital pins.



**Automatic (Software) Reset**

Rather than requiring a physical press of the reset button before an upload, the Arduino/Genuino Uno board is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened.

**LCD INTERFACE**

LCD panel consists of two patterned glass panels in which crystal is filled under vacuum. The thickness of glass varies according to end use. Most of the LCD modules have glass thickness in the range of 0.70 to 1.1mm

Normally these liquid crystal molecules are placed between glass plates to form a spiral staircase to twist the twist the light. Light entering the top plate twist 900 before entering the bottom plate. Hence the LCDs are also called as optical switches. These LCD cannot display any information directly. These act as an interface between electronics and electronics circuit to give a visual output.

**Technology: -** The liquid crystal display (LCD), as the name suggests is a technology based on the use of liquid crystal. It is a transparent material but after applying voltage it becomes opaque. This property is the fundamental operating principle of LCDs.

An LCD consists of two-glass panel with a cavity in between. The panels are sealed together. The inner surface of glass is coated with transparent material to form characters or symbols for display. The most common type of liquid crystal used is ‘nematis’. In this type of crystal the long rod type molecules are arranged in parallel. It changes the optical characteristics with change in direction by applying voltage to it.

There are two common type of LCDs which use this material. They are :

1. TN (Twisted nematic):-The twisted nematic field effect mode arranges the liquid crystal molecules by controlling their movement Witt electric voltage soaks twist them by 900 in the direction for their thickness. It controls the light passing through the polarized placed on the two plates of the LCD by controlling the movements of the liquid crystal molecules. Almost all the medium and small type segment LCD are these types. Hence this type is most common type used.
2. STN (Super twisted nematic) While the TN mode arranges the liquid crystal molecule by twisting them by 900 the STN effect mode arranges them by giving a still larger twist and provides a display by refringence effect of the liquid crystal. The LCD structure in STN mode is same as that in TN mode. But as it has a different arrangement of liquid crystal, and by bi-refringence effect of liquid crystal. The LCD structure in STN mode is same as that in TN mode but as it has a different arrangement of liquid crystal and bi-refringence effect there is a colour in display and also a background colour. In STN mode, a wide viewing angle is obtained. The STN mode also offers a high contrast display compare to the TN mode. This mode is widely used in large size full dot-matrix LCD modules. For colours it has multiple modes depending on the combination of the polarized and retardation film.

### Energy consumption: - LCD normally requires very little energy to operate typically 5μA to 25μA at five volts (per square inch) for a display. In addition, auxiliary lighting will require supplementary energy. ALL LCD require a pure AC drive voltage. Inadvertent DC voltage, such as DC component in an AC signal, can significantly reduce the life of LCDs and must be limited to 50mv DC.

**Direct drive: -** Direct drive, static or simplex drive, means that each segment of the LCD has an independent connection to the driver. Direct drive LCD has the highest contrast over the widest temperature ranges. They are widely used in outdoor application. Direct drives typically requires drive frequency between 30HZ and 60HZ frequency. Frequency below 30HZ will flicker the display. While frequency above 60HZ will excessive current draw in the circuit. This is very important for battery mode operation. If voltage frequency across the limit then LCD ’Off’ segments can be come in adherently energized. This partial activation of segment is known as cross talk or ghosting.

LCD is available in a verity of model having one to four rows of 8 to 20 characters each. A display with two rows of 16 characters is used for this example project. Almost all aspects of the design can be used with other model of LCD, since the internal structure of the various LCD models are almost same, differencing only in the number of driver chips used. The display module is powered from a 5 V supply.

Connecting an LCD to a micro controller is very simple, requiring either bit or an 8-bit bus. A 4-bit interface saves I/O pins but requires that the command and data be split into 4-bit pieces, which are sent one after the other. Thus the saving in I/O lines comes at the price of more complicated software. To simplify understanding of the software the example uses a 8-bit interface. Three control lines are required in addition to the data line.

The voltage at the V0 pin adjusts the contrast of the display. Normally this voltage is provided by an adjustable voltage divider. The control line E (Enable) enables or disables the display. When the display is enabling it monitors the value of the other two control lines and interprets the data lines accordingly. When the display is disabled it ignores the status of the other two control lines and places its data line drivers in a high impedance state (tri-state). The data bus can then be used for other purpose. The control line R/W (Read /Write) determines weather data is read from or written to the LCD. Finally, the RS (Register select) line distinguishes between commands and display characters.

#### LCD Controller device characteristics

The HD44780 contains 80 bytes of internal RAM called Data display RAM (DDR) that is used for presentation of characters in the LCD. The size of the DDR is independent of the LCD configuration (number of rows and character). For an LCD having two rows of characters, the leftmost character in the first row is assigned to address 0 of the DDR. Each following character position in the first display row is assigned to the next following address in turn until the 40th character location reached, which is assigned to DDR address 27 hex. The character locations in the second row of the display are assigned to DDR address 40 hex through 67 hex. If for example a character is to be written to the third position from the left in the second row of the display, it must be written to DDR, so that for example with a display, it must be written to DDR location 42hex.If the LCD module displays fewer than 40 characters per row, then these are mapped into a ‘window’ within the DDR, so that for example with a display of 16 characters per line only 16 of the 40 available DDR locations per line can be displayed at one time. The HD44780 supports commands to move this window to the left or to the right to allow various regions of the DDR to be displayed.

In additions to the DDR the HD44780 has a character genator ROM (CGROM) and a character-generator RAM (CGRAM).The CG ROM contains the dot-matrix patterns for the standard(fixed)character set,whilethe CGRAM allows the user to program additional character. Either eight4 X 7-point or four 5 X 10-point characters may be stored on the **CGRAM.**

###### **THE PRINCIPLES OF LCD TECHNOLOGY**

### In this section, we will explain everything ranging from the properties of liquid crystal molecules to the basic principle of display technology by using TN type liquid crystals as an example.

## The parallel arrangement of liquid crystal molecules along grooves

### When coming into contact with grooved surface in a fixed direction, liquid crystal molecules line up parallelly along the grooves.

Natural state

## When liquid crystals are sandwiched between upper and lower plates, they line-up with grooves pointing in directions 'a' and 'b,' respectively

|  |  |
| --- | --- |
| Description: p43-2 | The molecules along the upper plate point in direction 'a' and those along the lower plate in direction 'b,' thus forcing the liquid crystals into a twisted structural arrangement./ (figure shows a 90-degree twist) (TN type liquid crystal) |

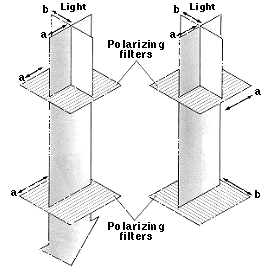
## Light travels through the spacing of the molecular arrangement

The light also "twists" as it passes through the twisted liquid crystals

## Molecules rearrange themselves when voltage is applied

When voltage is applied to the liquid crystal structure, the twisted light passes straight through.

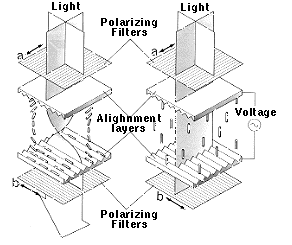
## Blocking light with two polarizing filters

When voltage is applied to a combination of two polarizing filters and twisted liquid crystal, it becomes a LCD display.  
  


### Light passes when two polarizing filters are arranged with polarizing axes as shown above, left. Light is blocked when two polarizing filters are arranged with polarizing axes as shown above, right.

## TN type LCDs

A combination of polarizing filters and twisted liquid crystal creates a liquid crystal display.



When two polarizing filters are arranged along perpendicular polarizing axes, light entering from above is re-directed 90 degrees along the helix arrangement of the liquid crystal molecules so that it passes through the lower When voltage is applied, the liquid crystal molecules straighten out of their helix pattern and stop redirecting the angle of the light, thereby preventing light from passing through the lower filter. This figure depicts the principle behind typical twisted nematic (TN) liquid crystal displays. In a TN type LCD, liquid crystals in which the molecules form a 90-degree twisted helix, are sandwiched between two polarizing filters. When no voltage is applied, light passes; when voltage is applied, light is blocked and the screen appears black. In other words, the voltage acts as a trigger causing the liquid crystals.

**Switch**

In electrical engineering, a switch is an electrical component that can disconnect or connect the conducting path in an electrical circuit, interrupting the electric current or diverting it from one conductor to another. The most common type of switch is an electromechanical device consisting of one or more sets of movable electrical contacts connected to external circuits. When a pair of contacts is touching current can pass between them, while when the contacts are separated no current can flow. Switches are made in many different configurations; they may have multiple sets of contacts controlled by the same knob or actuator, and the contacts may operate simultaneously, sequentially, or alternately. A switch may be operated manually, for example, a light switch or a keyboard button, or may function as a sensing element to sense the position of a machine part, liquid level, pressure, or temperature, such as a thermostat. Many specialized forms exist, such as the toggle switch, rotary switch, mercury switch, push-button switch, reversing switch, relay, and circuit breaker. A common use is control of lighting, where multiple switches may be wired into one circuit to allow convenient control of light fixtures. Switches in high-powered circuits must have special construction to prevent destructive arcing when they are opened.



The most familiar form of switch is a manually operated electromechanical device with one or more sets of electrical contacts, which are connected to external circuits. Each set of contacts can be in one of two states: either "closed" meaning the contacts are touching and electricity can flow between them, or "open", meaning the contacts are separated and the switch is nonconducting. The mechanism actuating the transition between these two states (open or closed) is usually (there are other types of actions) either an "alternate action" (flip the switch for continuous "on" or "off") or "momentary" (push for "on" and release for "off") type.

A switch may be directly manipulated by a human as a control signal to a system, such as a computer keyboard button, or to control power flow in a circuit, such as a light switch. Automatically operated switches can be used to control the motions of machines, for example, to indicate that a garage door has reached its full open position or that a machine tool is in a position to accept another workpiece. Switches may be operated by process variables such as pressure, temperature, flow, current, voltage, and force, acting as sensors in a process and used to automatically control a system. For example, a thermostat is a temperature-operated switch used to control a heating process. A switch that is operated by another electrical circuit is called a relay. Large switches may be remotely operated by a motor drive mechanism. Some switches are used to isolate electric power from a system, providing a visible point of isolation that can be padlocked if necessary to prevent accidental operation of a machine during maintenance, or to prevent electric shock.

An ideal switch would have no voltage drop when closed, and would have no limits on voltage or current rating. It would have zero rise time and fall time during state changes, and would change state without "bouncing" between on and off positions. Practical switches fall short of this ideal; as the result of roughness and oxide films, they exhibit contact resistance, limits on the current and voltage they can handle, finite switching time, etc. The ideal switch is often used in circuit analysis as it greatly simplifies the system of equations to be solved, but this can lead to a less accurate solution. Theoretical treatment of the effects of non-ideal properties is required in the design of large networks of switches, as for example used in telephone exchanges.

**Power switching**

When a switch is designed to switch significant power, the transitional state of the switch as well as the ability to withstand continuous operating currents must be considered. When a switch is in the on state, its resistance is near zero and very little power is dropped in the contacts; when a switch is in the off state, its resistance is extremely high and even less power is dropped in the contacts. However, when the switch is flicked, the resistance must pass through a state where a quarter of the load's rated power (or worse if the load is not purely resistive) is briefly dropped in the switch.

For this reason, power switches intended to interrupt a load current have spring mechanisms to make sure the transition between on and off is as short as possible regardless of the speed at which the user moves the rocker.

Power switches usually come in two types. A momentary on‑off switch (such as on a laser pointer) usually takes the form of a button and only closes the circuit when the button is depressed. A regular on‑off switch (such as on a flashlight) has a constant on-off feature. Dual-action switches incorporate both of these features.

**Conclusion**

This project is developed and tested in the laboratory and found to operating satisfactory in the test conditions. The accuracy observed is quite high.

**Code**

#include<LiquidCrystal.h>

LiquidCrystal lcd(2,3,4,5,6,7);

#define S1 A0

#define S2 A1

#define S3 A2

#define S4 A3

#define S5 A4

int vote1=0;

int vote2=0;

int vote3=0;

int vote4=0;

void setup()

{

pinMode(S1, INPUT);

pinMode(S2,INPUT);

pinMode(S3,INPUT);

pinMode(S4,INPUT);

pinMode(S5,INPUT);

lcd.begin(16, 2);

lcd.print(" Electronic ");

lcd.setCursor(0,1);

lcd.print(" Voting Machine ");

delay(4000);

digitalWrite(S1, HIGH);

digitalWrite(S2, HIGH);

digitalWrite(S3, HIGH);

digitalWrite(S4, HIGH);

digitalWrite(S5, HIGH);

lcd.clear();

lcd.setCursor(1,0);

lcd.print("BJP");

lcd.setCursor(5,0);

lcd.print("CON");

lcd.setCursor(9,0);

lcd.print("AAP");

lcd.setCursor(13,0);

lcd.print("BJD");

}

void loop()

{

lcd.setCursor(1,0);

lcd.print("BJP");

lcd.setCursor(1,1);

lcd.print(vote1);

lcd.setCursor(5,0);

lcd.print("CON");

lcd.setCursor(5,1);

lcd.print(vote2);

lcd.setCursor(9,0);

lcd.print("AAP");

lcd.setCursor(9,1);

lcd.print(vote3);

lcd.setCursor(13,0);

lcd.print("BJD");

lcd.setCursor(13,1);

lcd.print(vote4);

if(digitalRead(S1)==0)

vote1++;

while(digitalRead(S1)==0);

if(digitalRead(S2)==0)

vote2++;

while(digitalRead(S2)==0);

if(digitalRead(S3)==0)

vote3++;

while(digitalRead(S3)==0);

if(digitalRead(S4)==0)

vote4++;

while(digitalRead(S4)==0);

if(digitalRead(S5)==0)

{

int vote=vote1+vote2+vote3+vote4;

if(vote)

{

if((vote1 > vote2 && vote1 > vote3 && vote1 > vote4))

{

lcd.clear();

lcd.print("BJP is Winner");

delay(3000);

lcd.clear();

}

else if((vote2 > vote1 && vote2 > vote3 && vote2 > vote4))

{

lcd.clear();

lcd.print("CON is Winner");

delay(3000);

lcd.clear();

}

else if((vote3 > vote1 && vote3 > vote2 && vote3 > vote4))

{

lcd.clear();

lcd.print("AAP is Winner");

delay(3000);

lcd.clear();

}

else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)

{

lcd.setCursor(0,0);

lcd.clear();

lcd.print("BJD is Winner");

delay(3000);

lcd.clear();

}

else if(vote4 > vote1 && vote4 > vote2 && vote4 > vote3)

{

lcd.setCursor(0,0);

lcd.clear();

lcd.print("D is Winner");

delay(3000);

lcd.clear();

}

else

{

lcd.clear();

lcd.print(" Tie Up Or ");

lcd.setCursor(0,1);

lcd.print(" No Result ");

delay(3000);

lcd.clear();

}

}

else

{

lcd.clear();

lcd.print("No Voting....");

delay(3000);

lcd.clear();

}

vote1=0;vote2=0;vote3=0;vote4=0,vote=0;

lcd.clear();

}

}