***PROJECT 2***

**2.0: Arduino - Changing and moving the sentences on the LCD panel (Coding and Hardware connection)**

This is a simple project that uses Liquid Crystal LCD with arduino uno. Liquid Crystal LCD connected to arduino uno board serves as Output viewer media. The output displayed varies depending on the given command. This project can try how to program arduino uno and Display Text and running Text on Liquid Crystal LCD.

**Hardware Required**

Arduino UNO

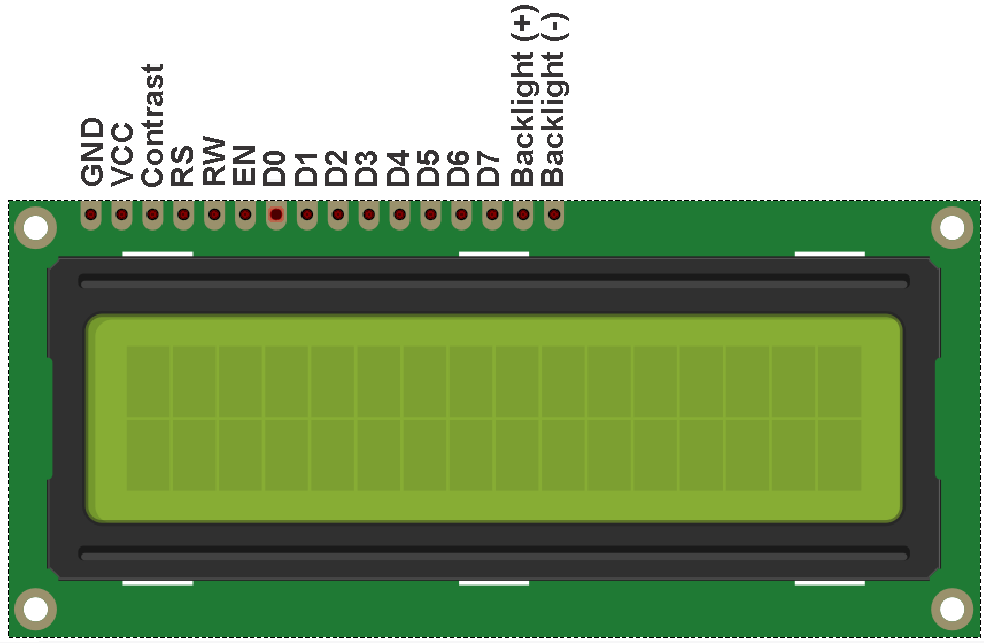
LCD ( 16 x 2 ) without driver

Jumper wires male to male

Breadboard

USB type A to B cable

**Schematic**



Connect the "GND" of the arduino uno board to "VSS" , "R\_W" and "K" LCD.

Connect the "5V" of the arduino uno board to "VDD" ,and "A" LCD.

Connect the "pin number 7" of the arduino uno board to "RS" LCD.

Connect the "pin number 6" of the arduino uno board to "E" LCD.

Connect the "pin number 5" of the arduino uno board to "DB(4)" LCD.

Connect the "pin number 4" of the arduino uno board to "DB(5)" LCD.

Connect the "pin number 3" of the arduino uno board to "DB(6)" LCD.

Connect the "pin number 2" of the arduino uno board to "DB(7)" LCD.

**Code**

#include <stdio.h>

#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins

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

void setup() {

// set up the LCD's number of rows and columns:

lcd.begin(16, 2);

// Print a message to the LCD.

lcd.print("Muhammad Ahmad Kamal"); // print to LCD screen

delay(1000);

}

void loop() {

// scroll 12 positions (string length) to the left

// to move it offscreen left:

for (int positionCounter = 0; positionCounter < 12; positionCounter++) {

// scroll one position left:

lcd.scrollDisplayLeft();

// wait a bit:

delay(300);

}

// scroll 28 positions (string length + display length) to the right

// to move it offscreen right: 12+16 = 28

for (int positionCounter = 0; positionCounter < 28; positionCounter++) {

// scroll one position right:

lcd.scrollDisplayRight();

// wait a bit:

delay(300);

}

// scroll 16 positions (display length + string length) to the left

// to move it back to center:

for (int positionCounter = 0; positionCounter < 16; positionCounter++) {

// scroll one position left:

lcd.scrollDisplayLeft();

// wait a bit:

delay(300);

}

// delay at the end of the full loop:

delay(1000);

}

**2.1: Arduino - Internet of Things (IOT) – to switch on/off LED and buzzer wirelessly using ESP8266 (Coding and Hardware connection)**

This example demonstrates the use of IOT to switch on/off LED and buzzer wirelessly using ESP8266. NodeMCU is a low-cost open source IoT platform. It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module.

**Hardware Required**

ESP8266WiFi (Node MCU)

LED

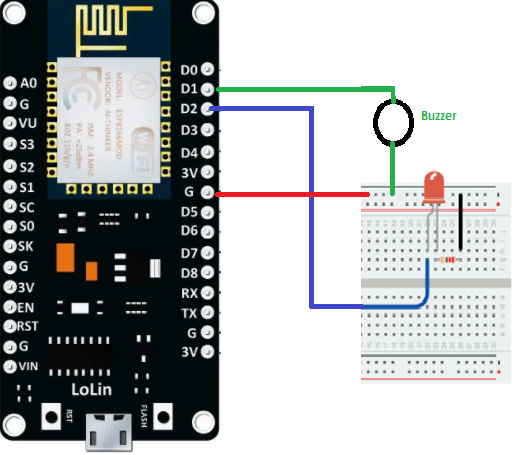
220 ohm resistor

buzzer

hook-up wires

breadboard

**Schematic**

****

**Code**

#include <ESP8266WiFi.h>

const char\* ssid = "Hazwani@unifi";

const char\* password = "0139755567";

int ledPin = D2;

int buzzerPin = D1;

WiFiServer server(80);

void setup() {

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,LOW);

pinMode(buzzerPin,OUTPUT);

digitalWrite(buzzerPin,LOW);

Serial.begin(115200);

Serial.println();

Serial.print("Wifi connecting to ");

Serial.println( ssid );

WiFi.begin(ssid,password);

Serial.println();

Serial.print("Connecting");

while( WiFi.status() != WL\_CONNECTED ){

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("Wifi Connected Success!");

Serial.print("NodeMCU IP Address : ");

Serial.println(WiFi.localIP() );

server.begin();

Serial.println("NodeMCU Server started");

// Print the IP address

Serial.print("Use this URL to connect: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/");

}

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}

// Wait until the client sends some data

Serial.println("Selamat datang ke kelas Embedded System");

while(!client.available()){

delay(1);

}

// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();

// Match the request

int value = LOW;

if (request.indexOf("/LED=ON") != -1) {

digitalWrite(ledPin, HIGH);

value = HIGH;

}

if (request.indexOf("/LED=OFF") != -1) {

digitalWrite(ledPin, LOW);

value = LOW;

}

int value2 = LOW;

if (request.indexOf("/BUZZER=ON") != -1) {

tone(buzzerPin, 1000); // Send 1KHz sound signal...

value2 = HIGH;

}

if (request.indexOf("/BUZZER=OFF") != -1) {

digitalWrite(buzzerPin, LOW);

value2 = LOW;

}

// Set ledPin according to the request

//digitalWrite(ledPin, value);

// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.print("Selamat Datang Ke Kelas Embedded System.</p> ");

client.print(" Keadaan semasa LED: ");

if(value == HIGH) {

client.print("Menyala");

} else {

client.print("Padam");

}

client.println("<br><br>");

client.println("<a href=\"/LED=ON\"\"><button>Turn On </button></a>");

client.println("<a href=\"/LED=OFF\"\"><button>Turn Off </button></a><br />");

client.println("</html>");

client.print(" Keadaan semasa Buzzer: ");

if(value2 == HIGH) {

client.print("Berbunyi");

} else {

client.print("Senyap");

}

client.println("<br><br>");

client.println("<a href=\"/BUZZER=ON\"\"><button>Turn On </button></a>");

client.println("<a href=\"/BUZZER=OFF\"\"><button>Turn Off </button></a><br />");

client.println("</html>");

delay(1);

Serial.println("Client disonnected");

Serial.println("");

}

**2.2: Arduino - Internet of Things (IOT) – to switch on/off LCD wirelessly using ESP8266 (Coding and Hardware connection)**

This example demonstrates the use of IOT to switch on/off LCD and buzzer wirelessly using ESP8266. NodeMCU is a low-cost open source IoT platform. It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module.

**Hardware Required**

ESP8266WiFi (Node MCU)

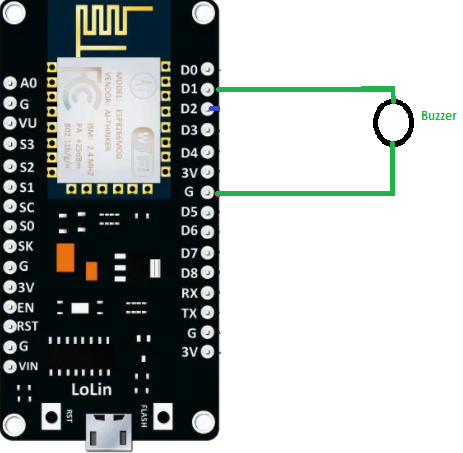
LCD ( 16 x 2 ) without driver

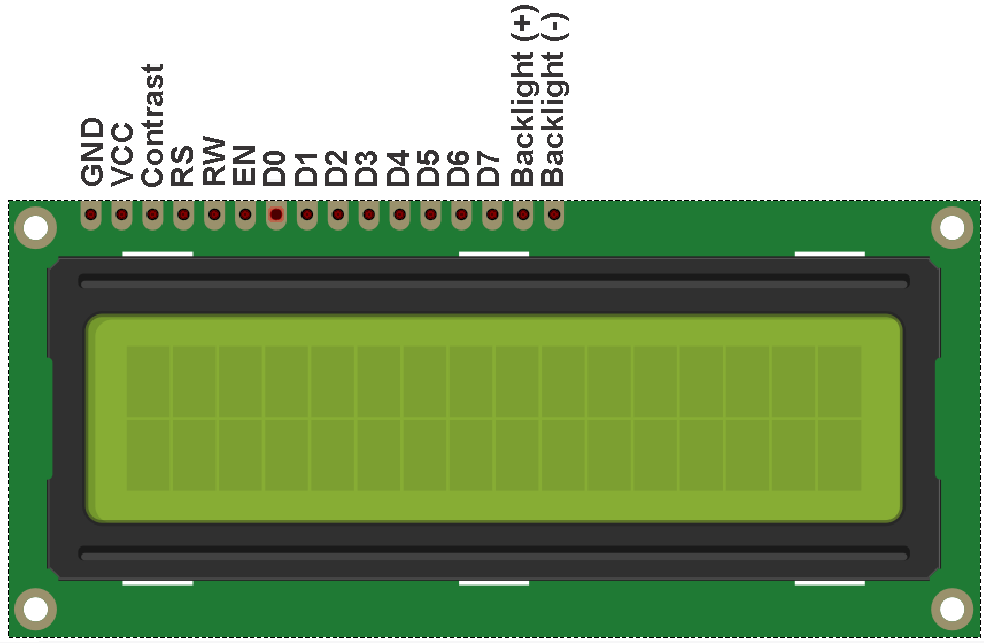
buzzer

hook-up wires

breadboard

**Schematic**

****



Connect the "GND" of the ESP8266 board to "VSS" , "R\_W" and "K" LCD.

Connect the "3V" of the ESP8266 board to "VDD" ,and "A" LCD.

Connect the "pin number 7" of the ESP8266 board to "RS" LCD.

Connect the "pin number 6" of the ESP8266 board to "E" LCD.

Connect the "pin number 5" of the ESP8266 board to "DB(4)" LCD.

Connect the "pin number 4" of the ESP8266 board to "DB(5)" LCD.

Connect the "pin number 3" of the ESP8266 board to "DB(6)" LCD.

Connect the "pin number 2" of the ESP8266 board to "DB(7)" LCD.

**Code**

//https://www.youtube.com/watch?v=MgOPZKpubBY

#include <ESP8266WiFi.h>

#include <LiquidCrystal.h>

const char\* ssid = "Hazwani@unifi";

const char\* password = "0139755567";

const int buzzerPin = D1;

WiFiServer server(80);

LiquidCrystal lcd(D7, D6, D5, D4, D3, D2);

void setup() {

pinMode(buzzerPin,OUTPUT);

digitalWrite(buzzerPin,LOW);

Serial.begin(115200);

Serial.println();

Serial.print("Wifi connecting to ");

Serial.println( ssid );

WiFi.begin(ssid,password);

Serial.println();

Serial.print("Connecting");

while( WiFi.status() != WL\_CONNECTED ){

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("Wifi Connected Success!");

Serial.print("NodeMCU IP Address : ");

Serial.println(WiFi.localIP() );

server.begin();

Serial.println("NodeMCU Server started");

// Print the IP address

Serial.print("Use this URL to connect: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/");

}

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}

// Wait until the client sends some data

Serial.println("Selamat datang ke kelas Embedded System");

while(!client.available()){

delay(1);

}

// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();

// Match the request

int value = LOW;

if (request.indexOf("/LCD=ON") != -1) {

lcd.begin(16, 2);

lcd.clear();

delay(2000);

lcd.setCursor(0,0);

lcd.print("Muhammad");

value = HIGH;

}

if (request.indexOf("/LCD=OFF") != -1) {

lcd.clear();

value = LOW;

}

int value2 = LOW;

if (request.indexOf("/BUZZER=ON") != -1) {

tone(buzzerPin, 1000); // Send 1KHz sound signal...

value2 = HIGH;

}

if (request.indexOf("/BUZZER=OFF") != -1) {

digitalWrite(buzzerPin, LOW);

value2 = LOW;

}

// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.print("Selamat Datang Ke Kelas Embedded System.</p> ");

client.print(" Keadaan semasa LCD: ");

if(value == HIGH) {

client.print("LCD ON");

} else {

client.print("LCD OFF");

}

client.println("<br><br>");

client.println("<a href=\"/LCD=ON\"\"><button>Turn On </button></a>");

client.println("<a href=\"/LCD=OFF\"\"><button>Turn Off </button></a><br />");

client.println("</html>");

client.print(" Keadaan semasa Buzzer: ");

if(value2 == HIGH) {

client.print("Berbunyi");

} else {

client.print("Senyap");

}

client.println("<br><br>");

client.println("<a href=\"/BUZZER=ON\"\"><button>Turn On </button></a>");

client.println("<a href=\"/BUZZER=OFF\"\"><button>Turn Off </button></a><br />");

client.println("</html>");

delay(1);

Serial.println("Client disonnected");

Serial.println("");

}