

DM-MEGA2560PRO-EXT V1.0

User Manual

Rev. 1.00



Table of Contents

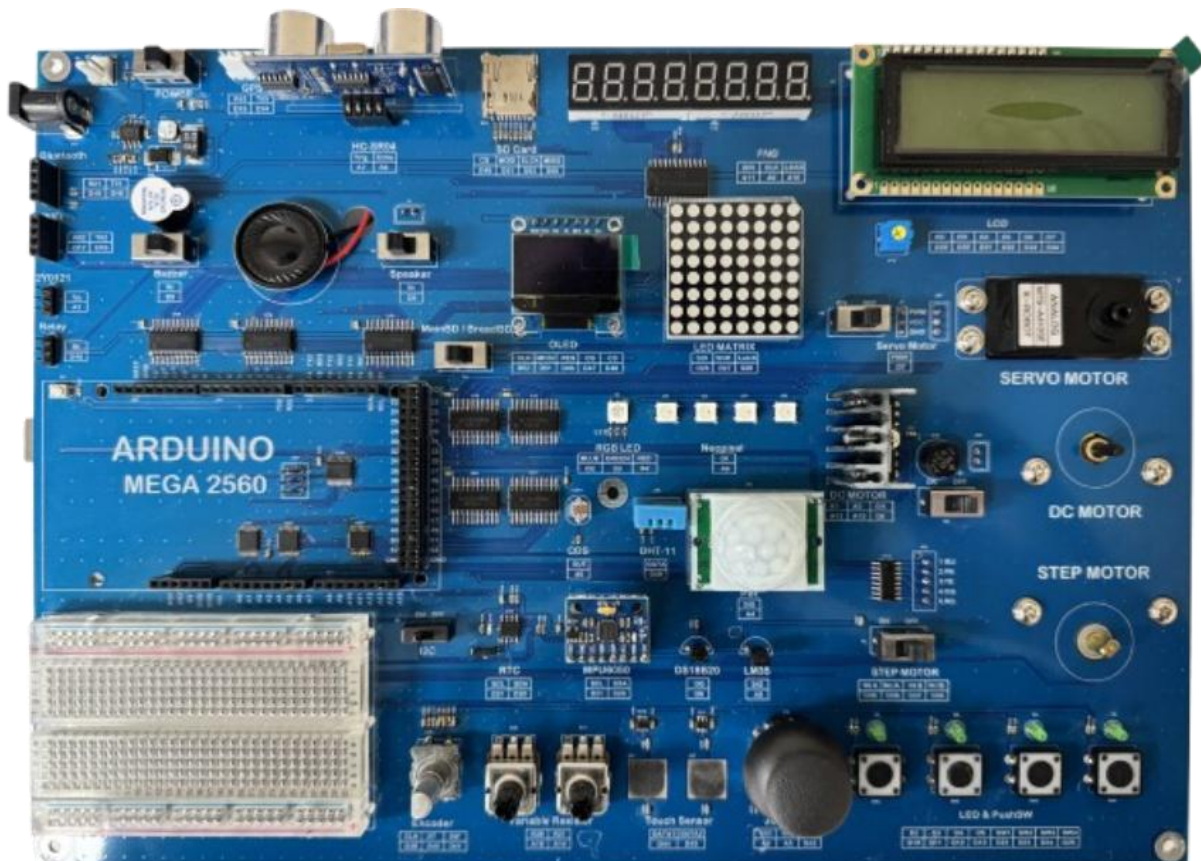
1. DM-MEGA2560PRO-EXT Introduction.....	3
2. Hardware Configuration	4
2.1. Display Section.....	4
2.2. Motor Section.....	8
2.3. Sensor Section.....	11
2.4. Various Switches	22
3. Pin Map	23
4. Upload Method.....	24
4.1. Uploading from the Sketch	24
5. Epilog	28
5.1. Product Inquiries and Acknowledgements	28
5.2. Technical Support Website	28
6. Revision	28

1. DM-MEGA2560PRO-EXT Introduction

This is a MEGA2560 expansion board used by connecting an Arduino MEGA2560 board.

After disconnecting it from the expansion board via switch operation while the MEGA2560 board remains connected,

you can connect pins to the MEGA2560 and build circuits directly using the built-in breadboard.



2. Hardware Configuration

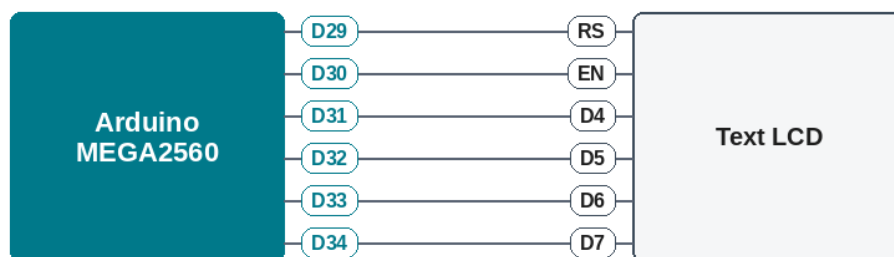
2.1. Display Section

LCD

A 2x16 LCD connected using the 4-bit (D4–D7) control method.

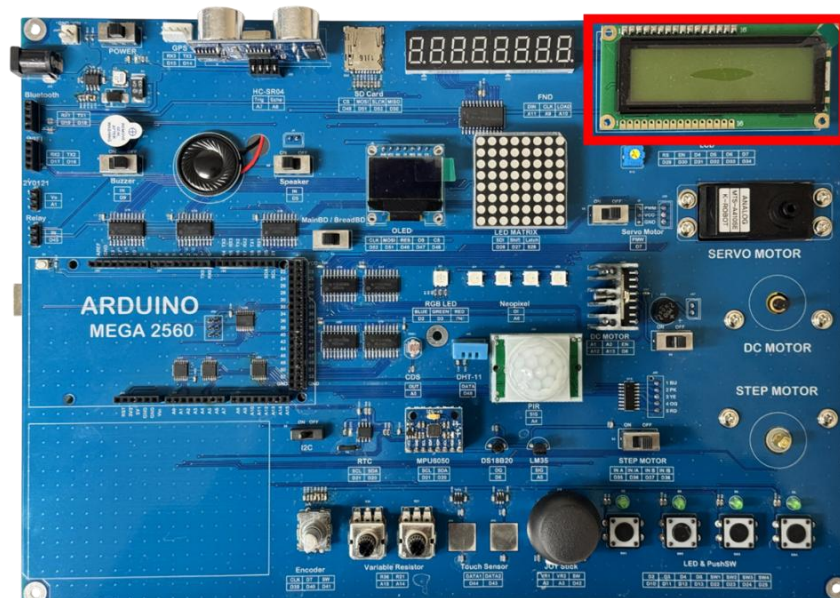
▪ Arduino Pin

RS=D29, EN=D30, D4=D31, D5=D32, D6=D33, D7=D34



▪ Example code

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(29, 30, 31, 32, 33, 34);
void setup() {
  lcd.begin(16, 2);
  lcd.print("Hello, World!");
}
void loop() {}
```

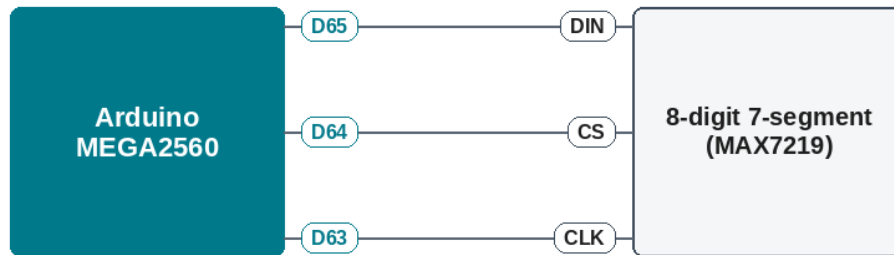


7-segment

Common cathode type, 4-digit 7-segment x2.

▪ Arduino Pin

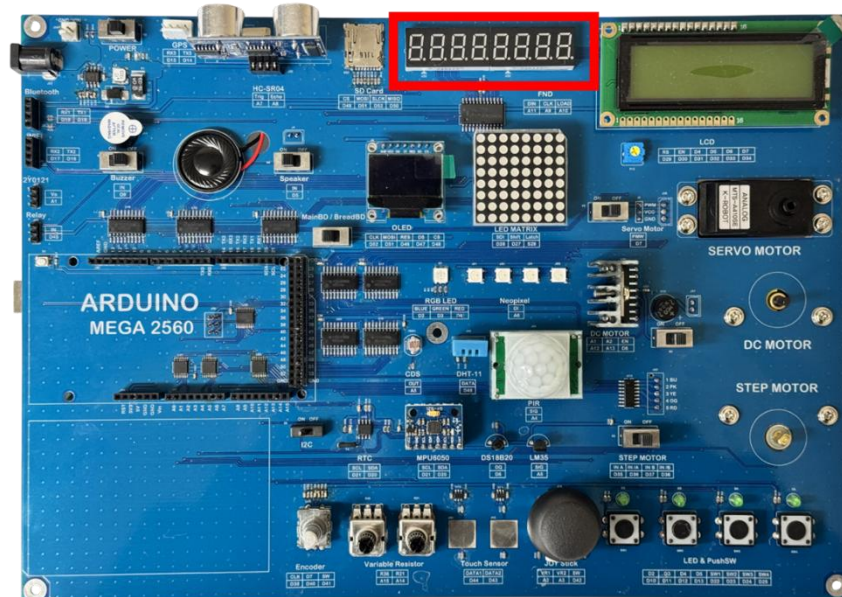
DIN=D65 (A11), CS=D64 (A10), CLK=D63 (A9) – MAX7219



▪ Example code

```

#include "DigitLedDisplay.h"
#define DIN A11 // = D65
#define CS A10 // = D64
#define CLK A9 // = D63
DigitLedDisplay ld = DigitLedDisplay(DIN, CS, CLK);
void setup() {
  ld.setBright(15);
  ld.setDigitLimit(8);
}
void loop() {
  ld.printDigit(12345678);
  delay(2000);
  ld.clear();
}
  
```

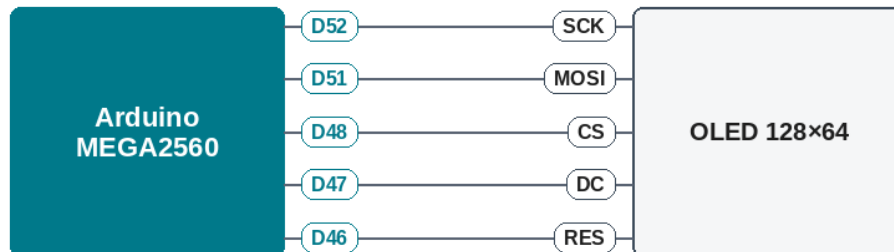


OLED

A 0.96-inch 128x64 OLED.

▪ Arduino Pin

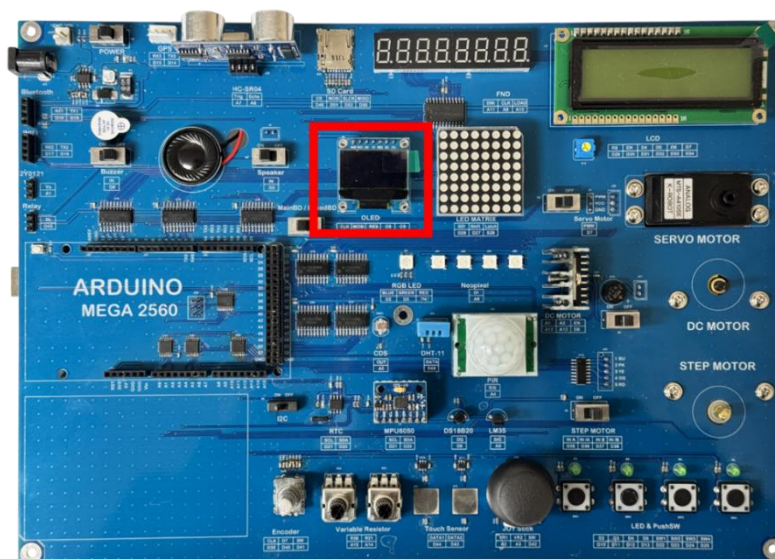
MOSI=D51, CLK=D52, DC=D47, RES=D46, CS=D48



▪ Example code

```

#include <SPI.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define OLED_MOSI 51
#define OLED_CLK 52
#define OLED_DC 47
#define OLED_RES 46
#define OLED_CS 48
Adafruit_SSD1306 display(128, 64,
    OLED_MOSI, OLED_CLK, OLED_DC, OLED_RES, OLED_CS);
void setup() {
    display.begin(SSD1306_SWITCHCAPVCC);
    display.clearDisplay();
    display.setTextSize(1); display.setTextColor(SSD1306_WHITE);
    display.setCursor(0, 0); display.print("Hello, OLED!");
    display.display();
}
void loop() {}
  
```

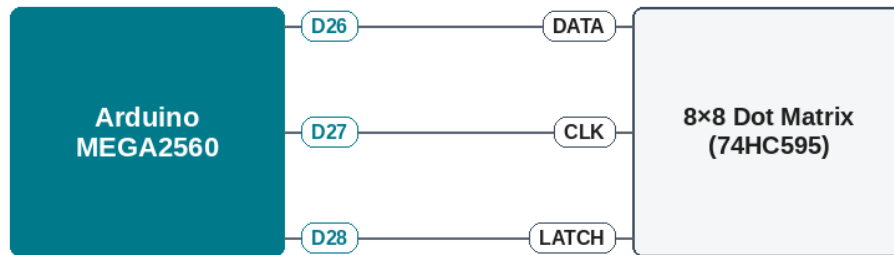


Dotmatrix

An 8x8 dot matrix.

▪ Arduino Pin

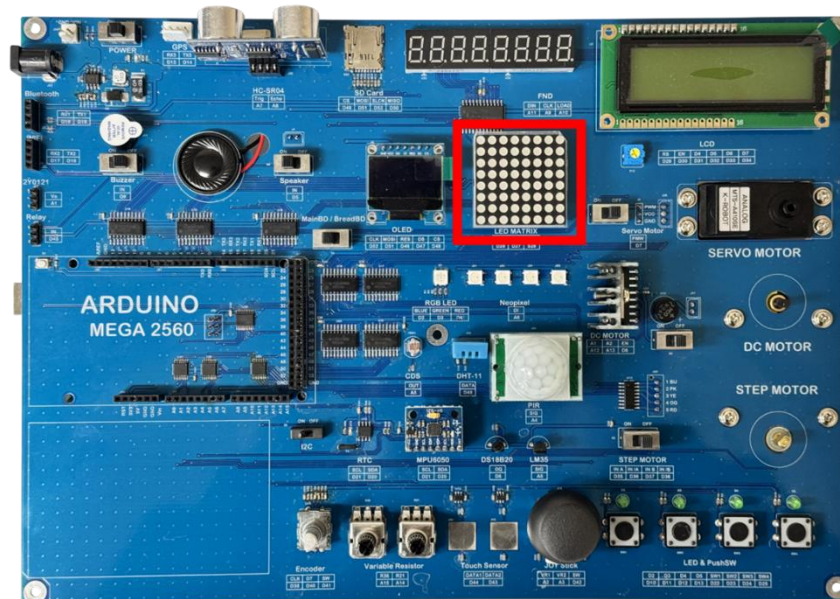
DATA=D26, CLK=D27, LATCH=D28 (74HC595)



▪ Example code

```

const int DATA=26, CLK=27, LATCH=28;
byte pat[8] = {0x18,0x3C,0x7E,0xFF,0xFF,0x7E,0x3C,0x18};
void setup() {
  pinMode(DATA, OUTPUT); pinMode(CLK, OUTPUT); pinMode(LATCH, OUTPUT);
}
void loop() {
  for (int r = 0; r < 8; r++) {
    digitalWrite(LATCH, LOW);
    shiftOut(DATA, CLK, MSBFIRST, ~(1 << r)); // row
    shiftOut(DATA, CLK, MSBFIRST, pat[r]);    // cols
    digitalWrite(LATCH, HIGH);
    delay(2);
  }
}
  
```

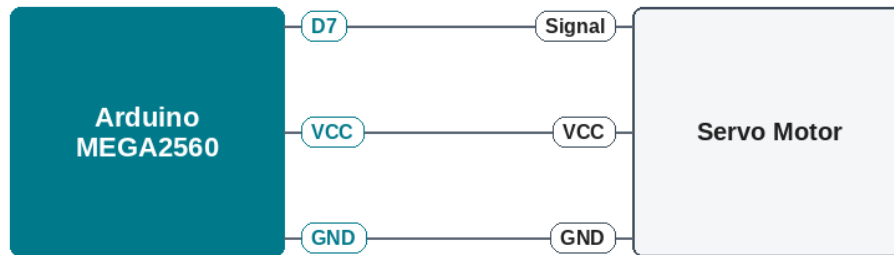


2.2. Motor Section

Servo Motor

▪ Arduino Pin

Signal=D7 (PWM)

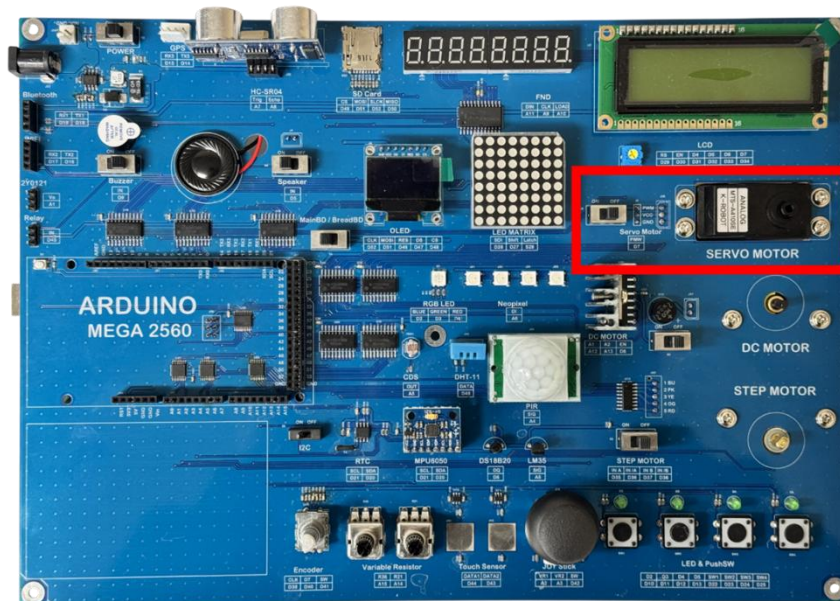


▪ Example code

```

#include <Servo.h>
Servo servo;
void setup() { servo.attach(7); }
void loop() {
    for (int a = 0; a <= 180; a++) { servo.write(a); delay(15); }
    for (int a = 180; a >= 0; a--) { servo.write(a); delay(15); }
}

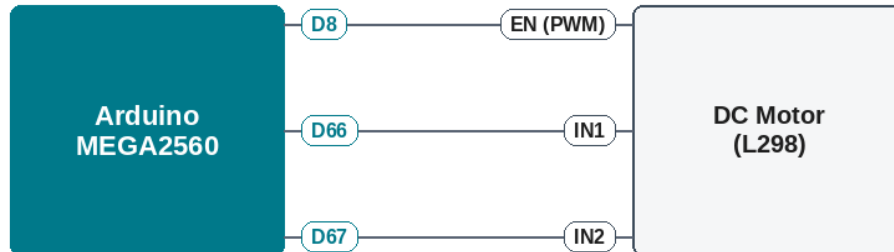
```



DC Motor

▪ Arduino Pin

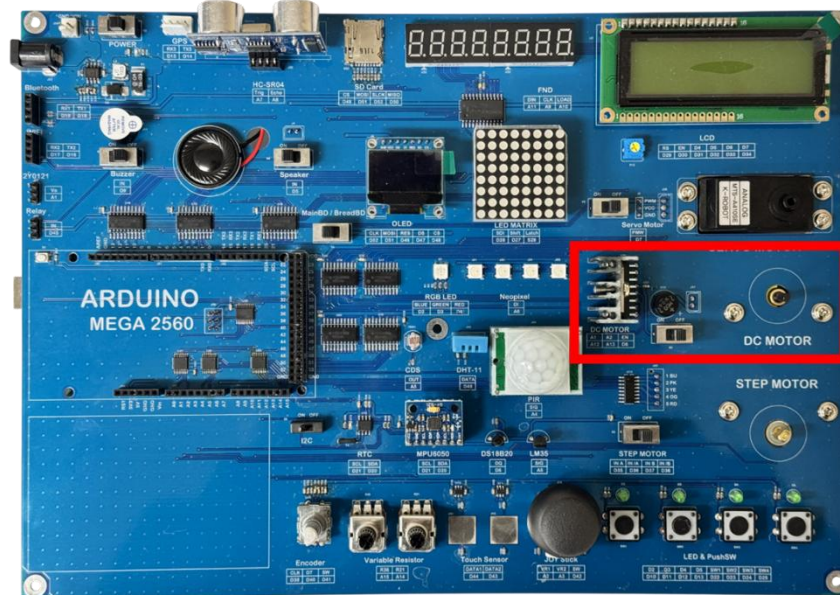
EN=D8 (PWM), IN1=D66, IN2=D67 (L298)



▪ Example code

```

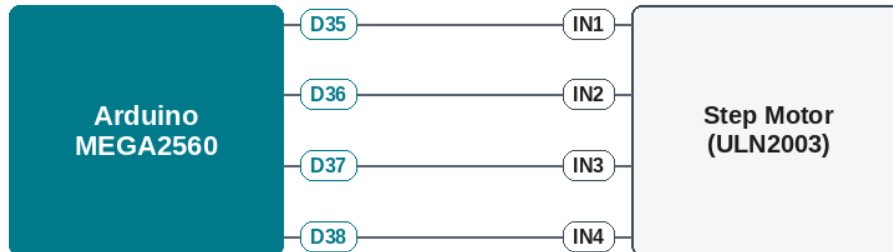
const int EN=8, IN1=66, IN2=67;
void setup() {
  pinMode(EN, OUTPUT); pinMode(IN1, OUTPUT); pinMode(IN2, OUTPUT);
}
void loop() {
  digitalWrite(IN1, HIGH); digitalWrite(IN2, LOW); // forward
  analogWrite(EN, 200); // speed
  delay(2000);
  digitalWrite(IN1, LOW); digitalWrite(IN2, HIGH); // reverse
  delay(2000);
}
  
```



Step Motor

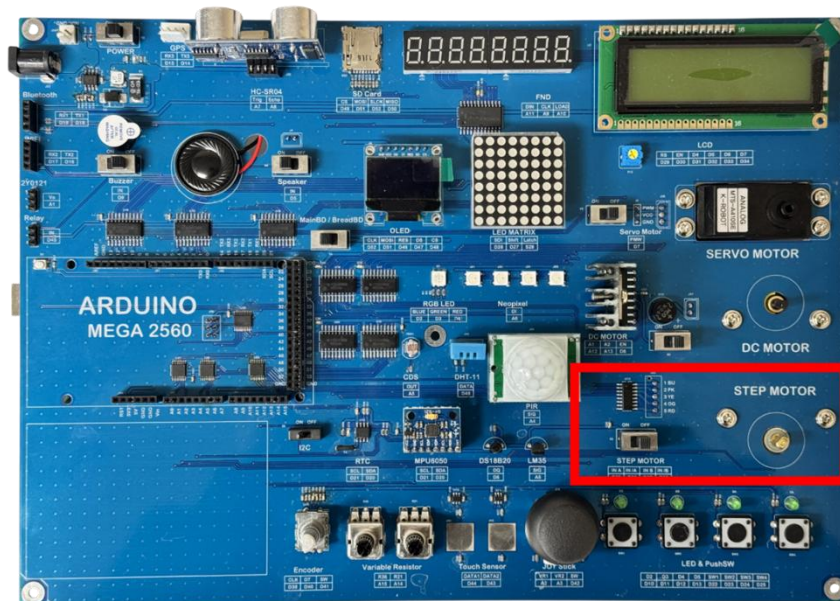
▪ Arduino Pin

IN1=D35, IN2=D36, IN3=D37, IN4=D38 (ULN2003)



▪ Example code

```
#include <Stepper.h>
// 28BYJ-48: 2048 steps / revolution
Stepper stepper(2048, 35, 37, 36, 38);
void setup() { stepper.setSpeed(10); }
void loop() {
    stepper.step(2048); // one full revolution
    delay(1000);
}
```



2.3. Sensor Section

CDS

A CDS light sensor.

- **Arduino Pin**

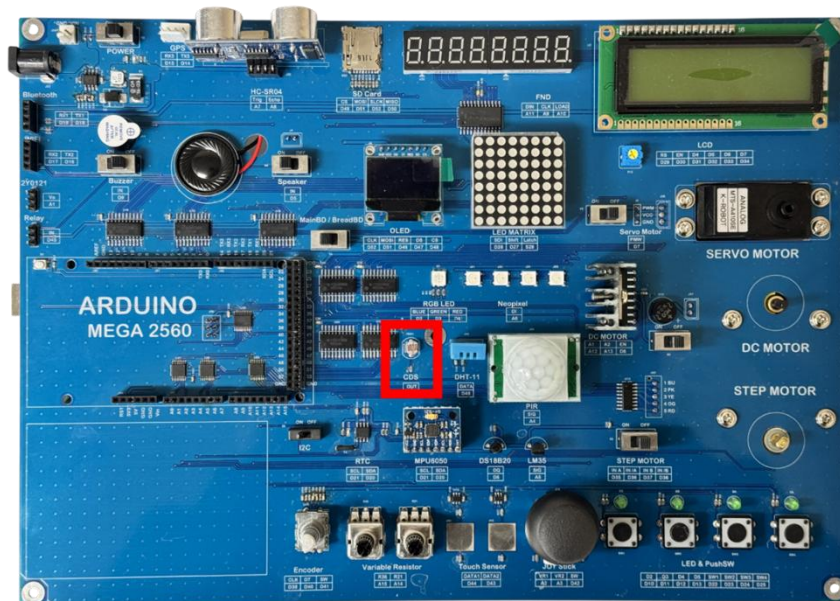
OUT=A5 (analog)



- **Example code**

```

void setup() { Serial.begin(9600); }
void loop() {
  int light = analogRead(A5); // 0 (bright) .. 1023 (dark)
  Serial.println(light);
  delay(500);
}
  
```



DHT-11

A digital temperature and humidity sensor.

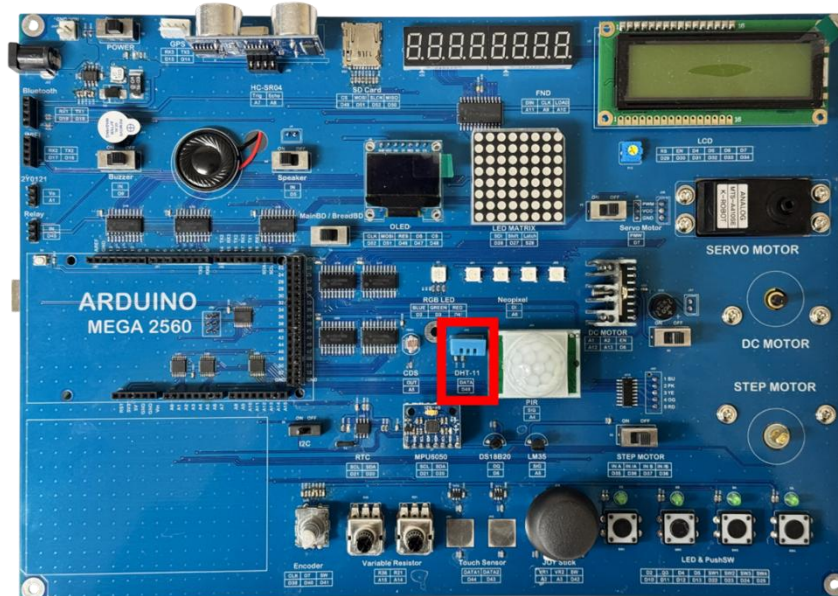
▪ Arduino Pin

DATA=D49



▪ Example code

```
#include "DHT.h"           // Rob Tillaart DHTlib
#define DHTPIN 49
DHT dht;
void setup() {
  Serial.begin(9600);
  dht.setup(DHTPIN);
}
void loop() {
  delay(2000);
  Serial.print("H="); Serial.print(dht.getHumidity());
  Serial.print(" T="); Serial.println(dht.getTemperature());
}
```



PIR

A PIR motion sensor.

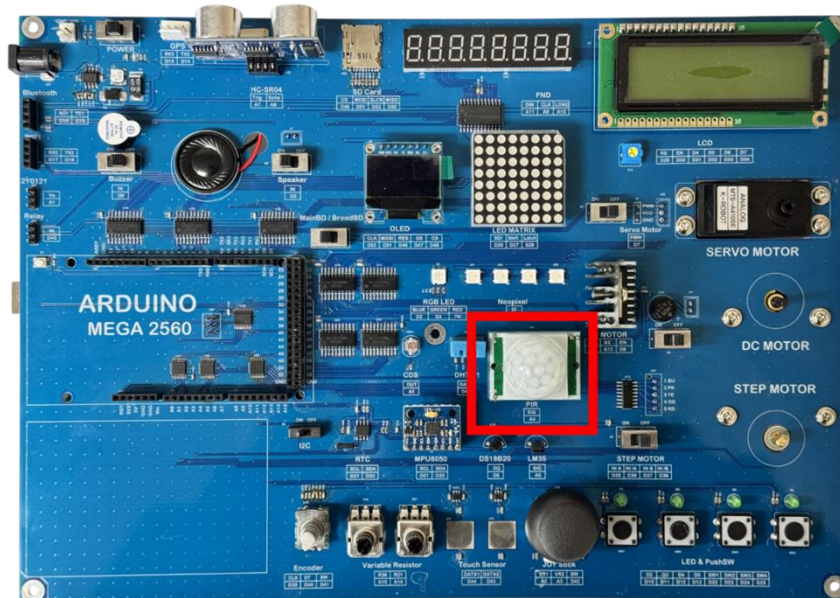
▪ Arduino Pin

OUT=A4 (digital)



▪ Example code

```
const int PIR = A4;
void setup() {
  pinMode(PIR, INPUT); Serial.begin(9600);
}
void loop() {
  if (digitalRead(PIR) == HIGH) Serial.println("Motion detected!");
  delay(200);
}
```

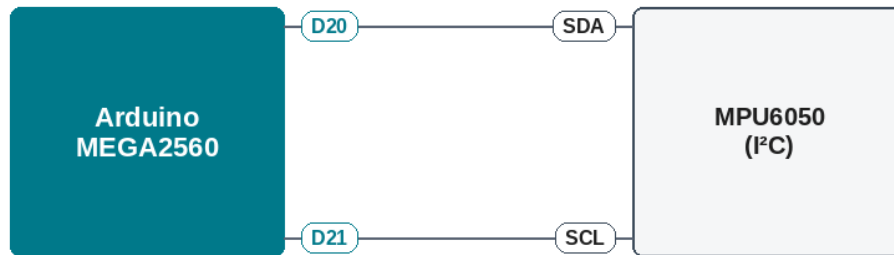


MPU6050

A 6-axis gyro/accelerometer sensor.

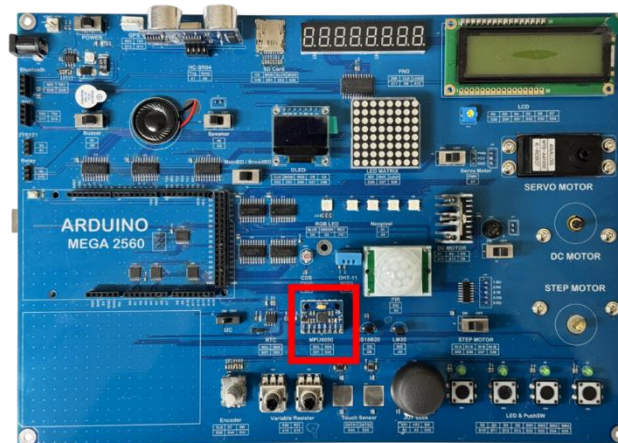
▪ Arduino Pin

SDA=D20, SCL=D21 (I²C, addr=0x69)



▪ Example code

```
#include <Wire.h>
const int MPU = 0x69;
void setup() {
  Wire.begin();
  Wire.beginTransmission(MPU);
  Wire.write(0x6B); Wire.write(0); // wake up
  Wire.endTransmission(true);
  Serial.begin(9600);
}
void loop() {
  Wire.beginTransmission(MPU);
  Wire.write(0x3B); // ACCEL_XOUT_H
  Wire.endTransmission(false);
  Wire.requestFrom(MPU, 6, true);
  int16_t ax = Wire.read() << 8 | Wire.read();
  int16_t ay = Wire.read() << 8 | Wire.read();
  int16_t az = Wire.read() << 8 | Wire.read();
  Serial.print(ax); Serial.print(' ');
  Serial.print(ay); Serial.print(' ');
  Serial.println(az);
  delay(200);
}
```



DS18B20

A digital temperature sensor.

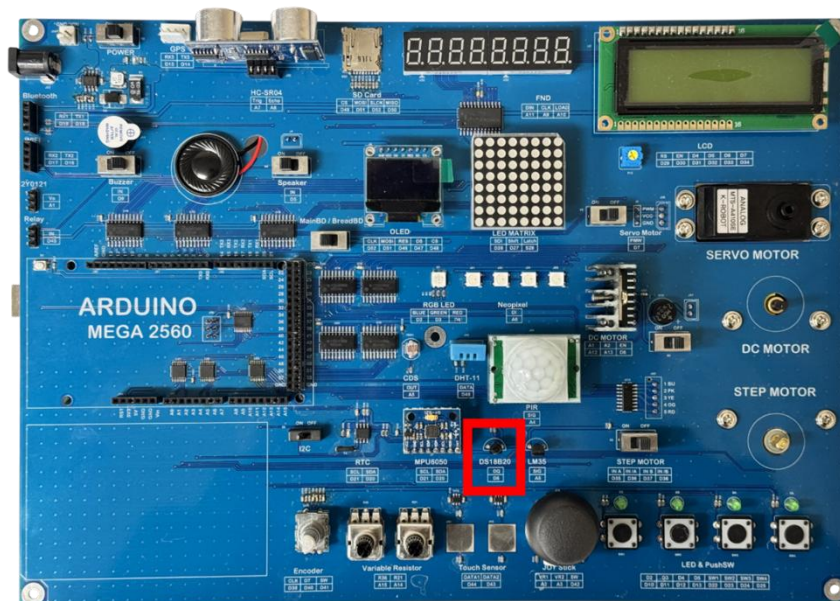
▪ Arduino Pin

DATA=D6 (1-Wire)



▪ Example code

```
#include <OneWire.h>
#include <DallasTemperature.h>
OneWire wire(6);
DallasTemperature sensor(&wire);
void setup() { Serial.begin(9600); sensor.begin(); }
void loop() {
  sensor.requestTemperatures();
  Serial.println(sensor.getTempCByIndex(0));
  delay(1000);
}
```



LM35

An analog temperature sensor.

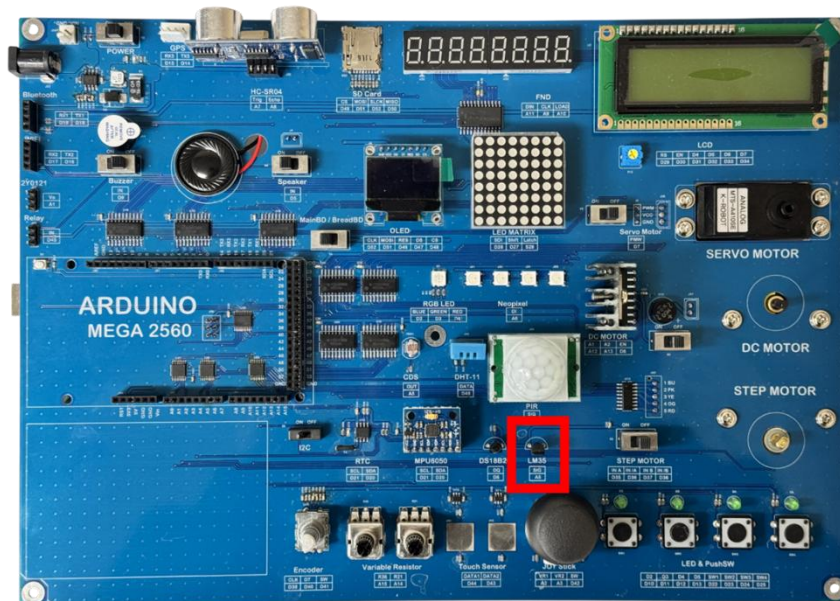
- **Arduino Pin**

OUT=A0 (analog)



- **Example code**

```
void setup() { Serial.begin(9600); }
void loop() {
  float v    = analogRead(A0) * (5.0 / 1023.0);
  float tempC = (v - 0.5) * 100.0; // 10 mV / °C, 500 mV at 0°C
  Serial.println(tempC);
  delay(500);
}
```



HC-SR04

An ultrasonic distance sensor.

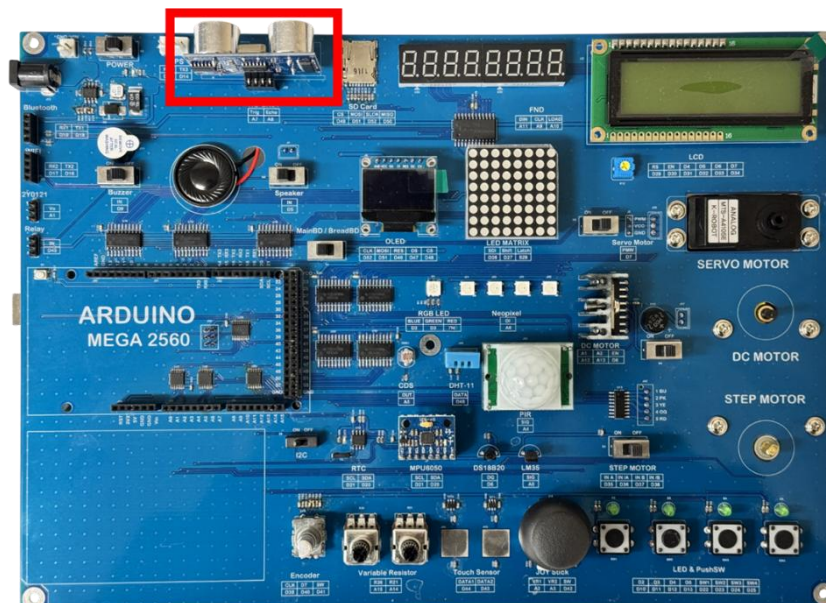
▪ Arduino Pin

TRIG=A7, ECHO=A8



▪ Example code

```
#define TRIG A7
#define ECHO A8
void setup() {
  Serial.begin(9600);
  pinMode(TRIG, OUTPUT); pinMode(ECHO, INPUT);
}
void loop() {
  digitalWrite(TRIG, LOW); delayMicroseconds(2);
  digitalWrite(TRIG, HIGH); delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  long t = pulseIn(ECHO, HIGH);
  long cm = t * 17 / 1000; // (t * 343 m/s) / 2 / 10000
  Serial.print("Distance: "); Serial.print(cm); Serial.println(" cm");
  delay(1000);
}
```



Potentiometer

A potentiometer whose resistance value can be changed by turning the dial.

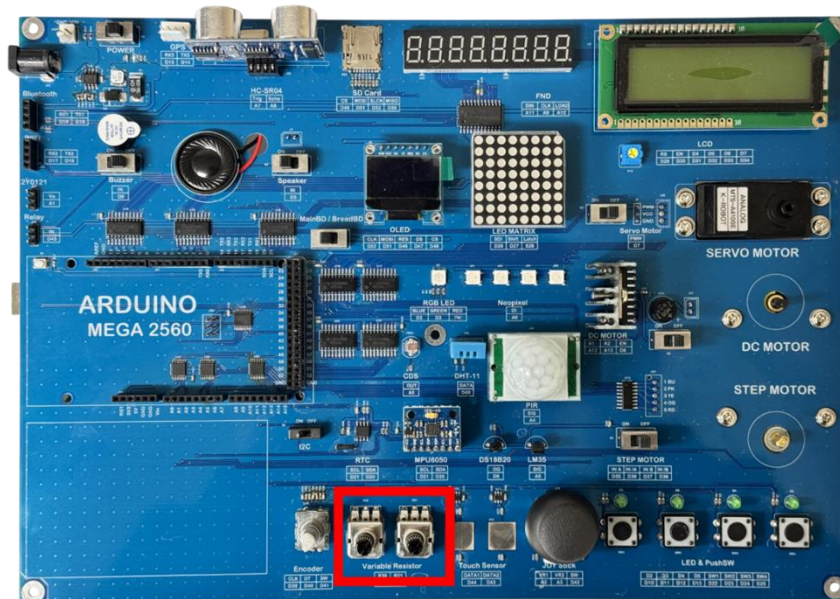
▪ Arduino Pin

Wiper=A14 (also A15 available)



▪ Example code

```
void setup() { Serial.begin(9600); }  
void loop() {  
  int v = analogRead(A14);    // 0 .. 1023  
  Serial.println(v);  
  delay(200);  
}
```

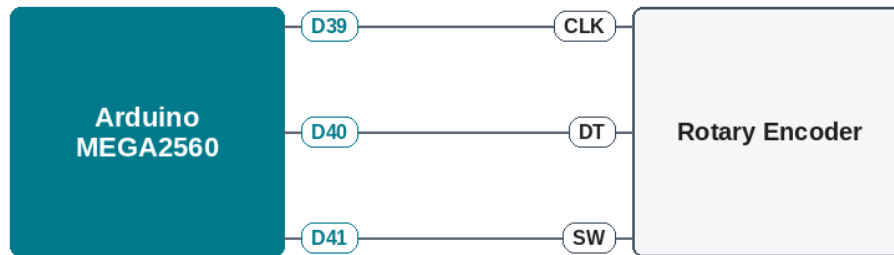


Encoder

A rotary encoder switch that also has a push-switch function.

▪ Arduino Pin

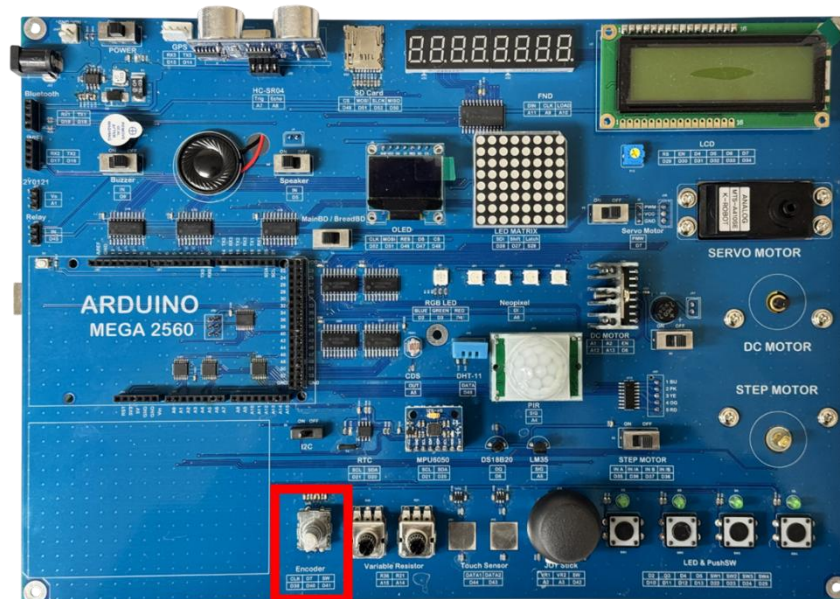
CLK=D39, DT=D40, SW=D41



▪ Example code

```

const int CLK=39, DT=40, SW=41;
int pos = 0, last;
void setup() {
  pinMode(CLK, INPUT); pinMode(DT, INPUT); pinMode(SW, INPUT_PULLUP);
  last = digitalRead(CLK);
  Serial.begin(9600);
}
void loop() {
  int c = digitalRead(CLK);
  if (c != last && c == LOW)
    pos += (digitalRead(DT) != c) ? 1 : -1;
  last = c;
  Serial.println(pos);
}
  
```



Buzzer

A device that provides sound-signal notifications.

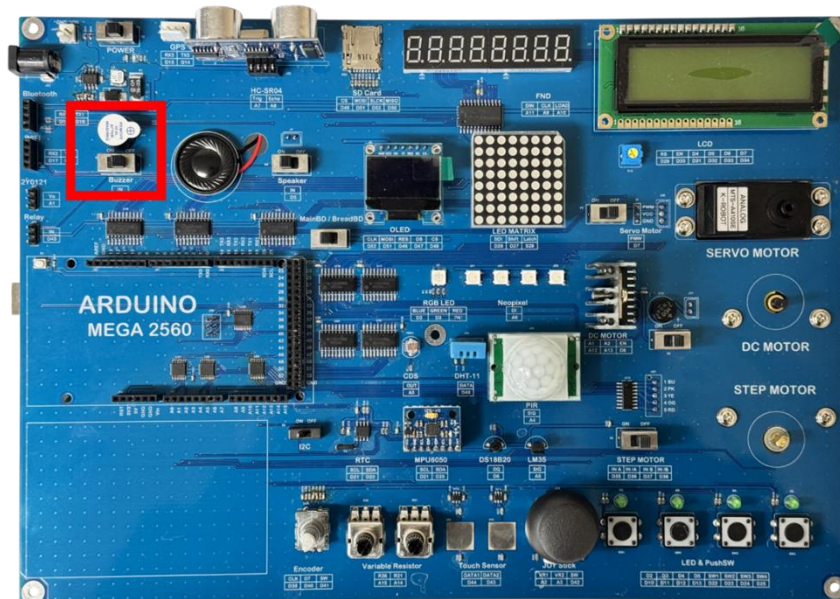
▪ Arduino Pin

IN=D9



▪ Example code

```
const int BUZZ = 9;
void setup() { pinMode(BUZZ, OUTPUT); }
void loop() {
  digitalWrite(BUZZ, HIGH); delay(300);
  digitalWrite(BUZZ, LOW); delay(300);
}
```



Speaker

A device that outputs electrical audio signals as sound.

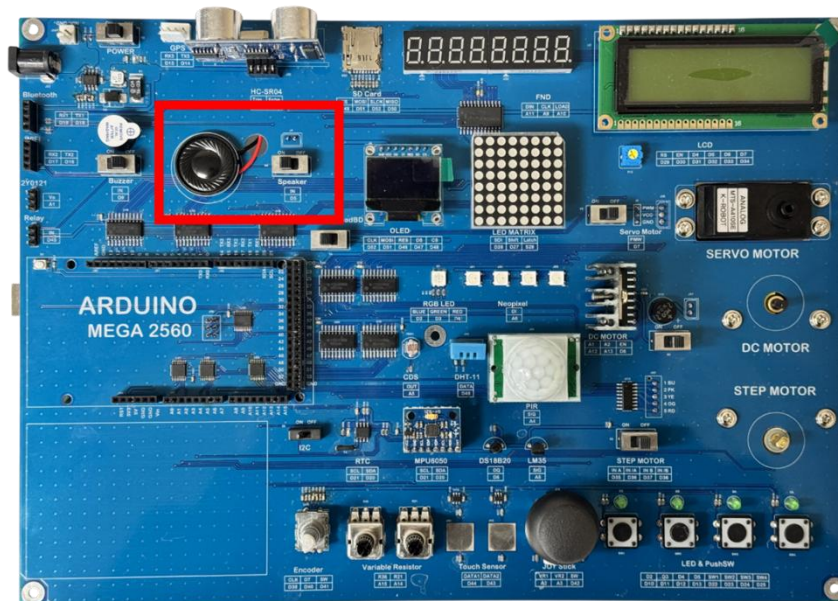
▪ Arduino Pin

Signal=D5

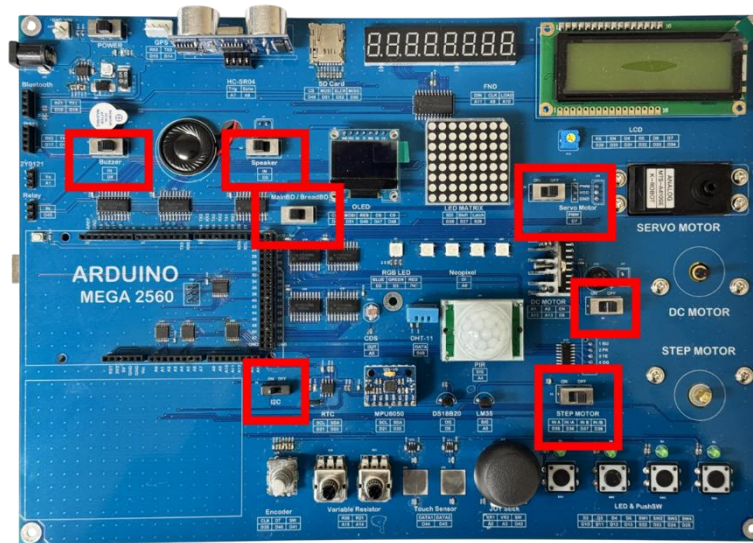


▪ Example code

```
const int SPK = 5;
int notes[] = { 262, 294, 330, 349, 392, 440, 494, 523 };
void setup() {}
void loop() {
  for (int i = 0; i < 8; i++) { tone(SPK, notes[i], 300); delay(350); }
  delay(1000);
}
```



2.4. Various Switches



S3 – Step motor On/Off

A switch that turns the step motor power On/Off.

S5 – DC motor On/Off

A switch that turns the DC motor power On/Off.

S8 – Servo Motor On/Off

A switch that turns the servo motor power On/Off.

S6 – Speaker On/Off

A switch that turns the speaker power On/Off.

S7 – Buzzer On/Off

A switch that turns the buzzer power On/Off.

S4 – Connect/disconnect the MEGA2560 board to/from the main board (when using the breadboard)

A switch that turns the connection between the attached MEGA2560 board and the main board On/Off.

S9 – I2C On/Off — connect/disconnect the MEGA2560 main-board I2C pins

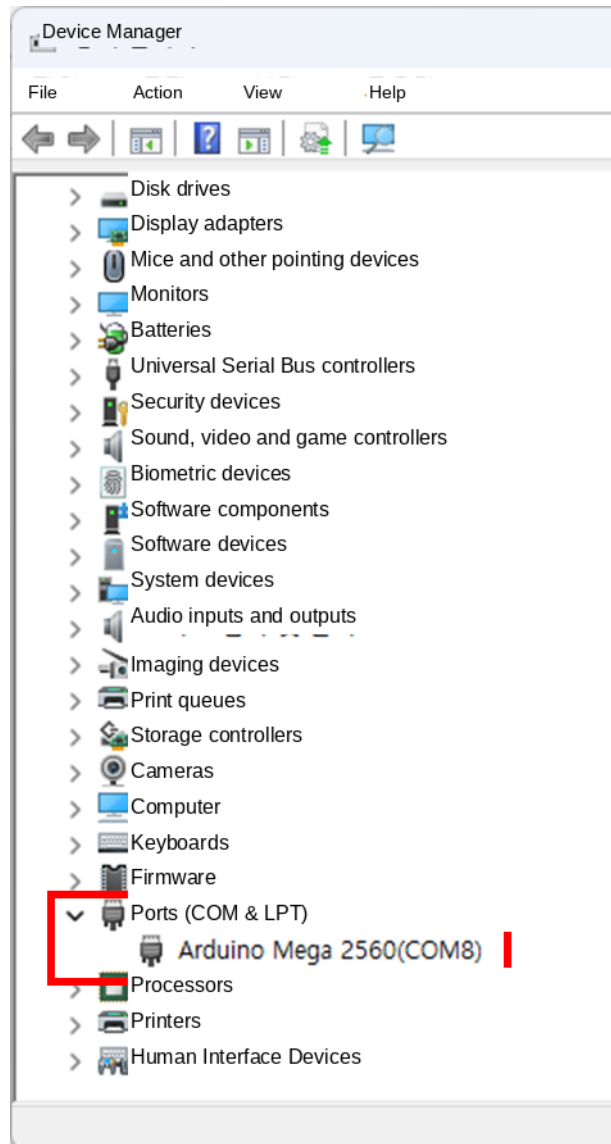
A switch that turns the I2C communication pin connection On/Off.

Pin Map

I/O Pin		Description	I/O Pin		Description
0	PE0	Program Upload (Serial)	35	PC2	Step Motor
1	PE1	Program Upload (Serial)	36	PC1	Step Motor
2	PE4	RGB LED	37	PC0	Step Motor
3	PE5	RGB LED	38	PD7	Step Motor
4	PG5	RGB LED	39	PG2	Rotary Encoder
5	PE3	Speaker	40	PG1	Rotary Encoder
6	PH3	DS18B20 Temperature Sensor	41	PG0	Rotary Encoder
7	PH4	Servo Motor	42	PL7	Joystick Button
8	PH5	DC Motor	43	PL6	Touch Sensor
9	PH6	Active Buzzer	44	PL5	Touch Sensor
10	PB4	LED	45	PL4	Relay
11	PB5	LED	46	PL3	SPI - OLED
12	PB6	LED	47	PL2	SPI - OLED
13	PB7	LED	48	PL1	SPI - OLED
14	PJ1	GPS	49	PL0	DHT11 Temp/Humidity Sensor
15	PJ0	GPS	50	PB3	SPI - MISO
16	PH1	UART - WiFi	51	PB2	SPI - MOSI
17	PH0	UART - WiFi	52	PB1	SPI - SCK
18	PD3	UART - Bluetooth	53	PB0	SPI - SD Card
19	PD2	UART - Bluetooth	54	PF0	TMP36 Temperature Sensor
20	PD1	I2C - SDA	55	PF1	Infrared Distance Sensor
21	PD0	I2C - SCL	56	PF2	Joystick
22	PA0	Button (Pull-down)	57	PF3	Joystick
23	PA1	Button (Pull-down)	58	PF4	PIR Sensor
24	PA2	Button (Pull-up)	59	PF5	Illuminance Sensor
25	PA3	Button (Pull-up)	60	PF6	NeoPixel
26	PA4	74595 - LED Matrix	61	PF7	Ultrasonic Distance Sensor
27	PA5	74595 - LED Matrix	62	PK0	Ultrasonic Distance Sensor
28	PA6	74595 - LED Matrix	63	PK1	8-digit 7-segment (MAX7219)
29	PA7	Text LCD	64	PK2	8-digit 7-segment (MAX7219)
30	PC7	Text LCD	65	PK3	8-digit 7-segment (MAX7219)
31	PC6	Text LCD	66	PK4	DC Motor
32	PC5	Text LCD	67	PK5	DC Motor
33	PC4	Text LCD	68	PK6	Potentiometer 1
34	PC3	Text LCD	69	PK7	Potentiometer 2

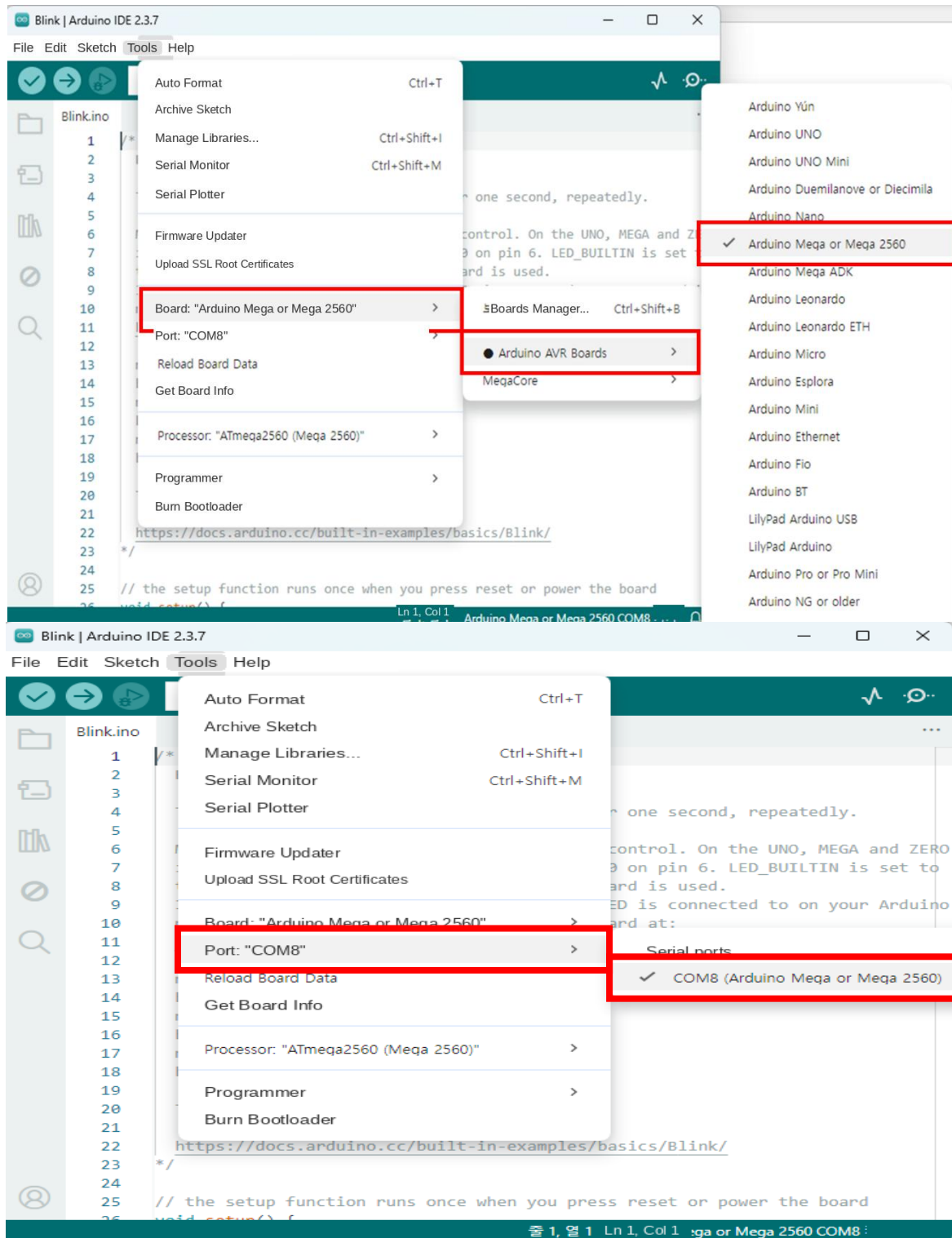
3. Upload Method

3.1. Uploading from the Sketch

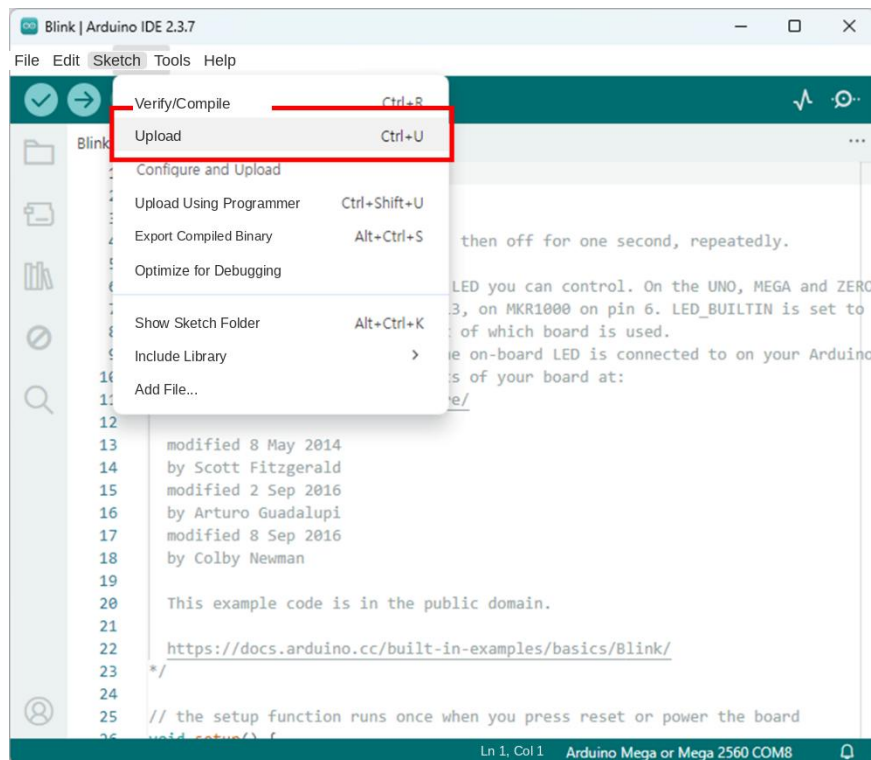
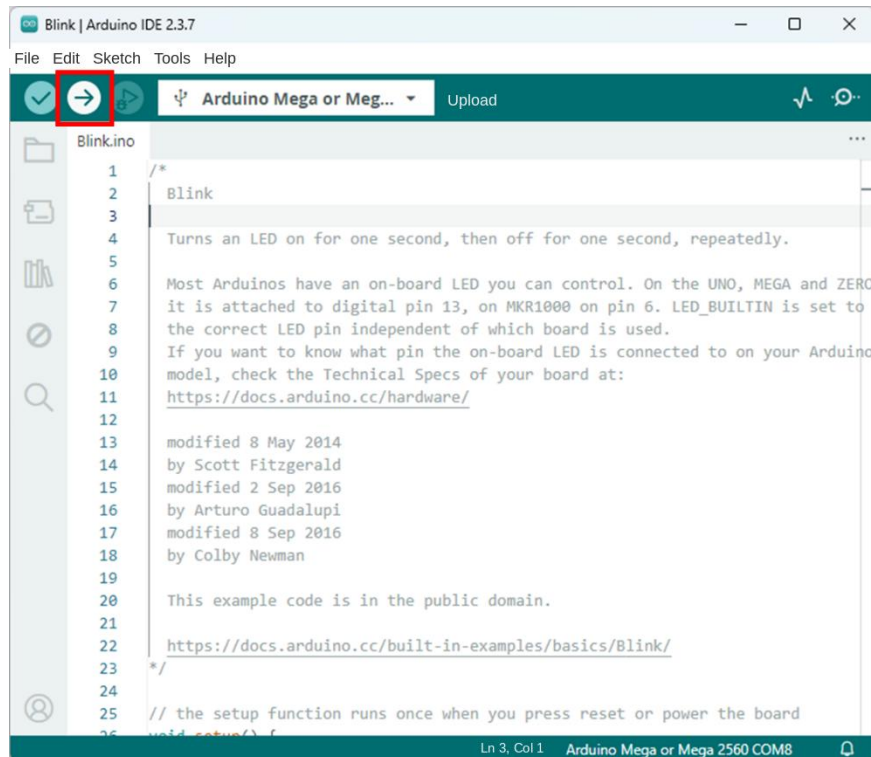


Connect the MEGA2560 board to the PC using a USB cable, and check the port number in Device Manager.

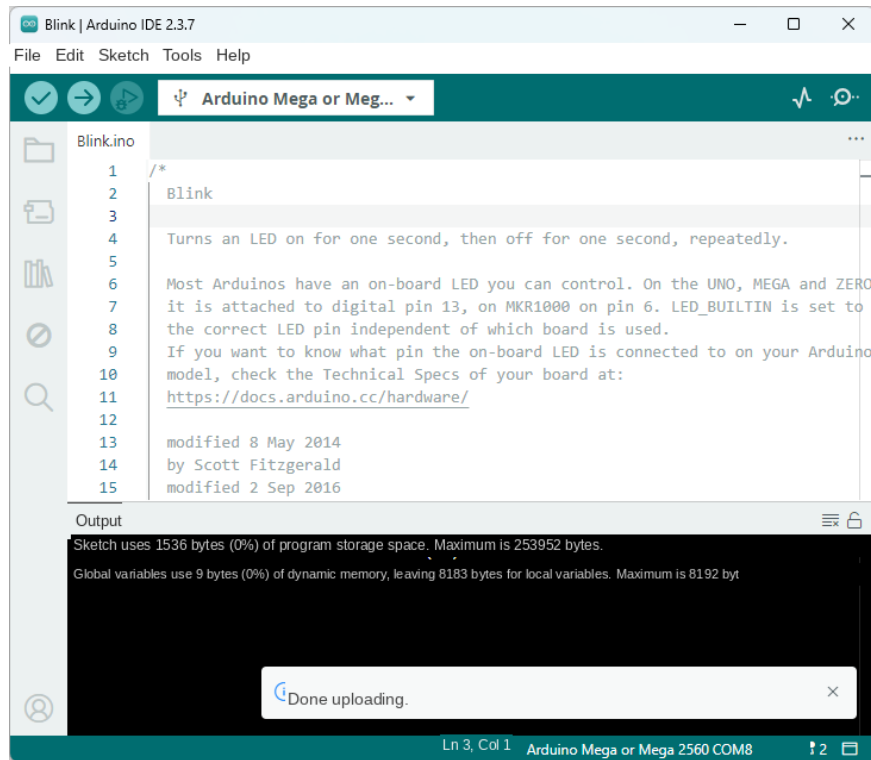
(In the photo above, the port number is COM8.)



In the Arduino IDE top menu, under Tools, set the board to Arduino Mega2560 and the port to COM8.



Then press the Upload button, or click Upload under the Sketch top menu.



Upload completed

4. Epilog

4.1. Product Inquiries and Acknowledgements

Thank you for purchasing a product from NEWTC Co., Ltd.

To improve the convenience of users who use the Arduino At Heart expansion board, we are always working hard on its development.

To study this material, use the kit examples and tutorials, or refer to the website's tutorials, download library, and other resources.

4.2. Technical Support Website

Technical support website: <http://www.newtc.co.kr> On the technical support website, lectures in various fields such as AVR, ARM, RENESAS, FPGA, electronics, and robot building are continually updated, and the download library is updated with various files and programs, so please refer to them. If you have any questions about a product, please do not hesitate to leave them in the Q&A section of the Customer Support menu on the website. For development-related inquiries, please use E-mail(newtc@newtc.co.kr). Thank you.

5. Revision

Rev.	Date	Description
1.00	27. 05, 2026	Release version completed