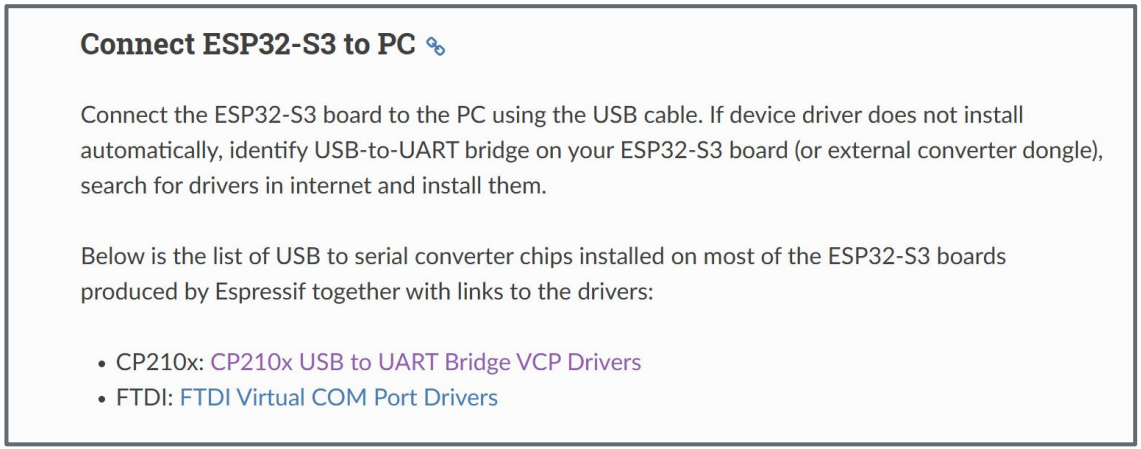
Workshop II

ESP32 Controlled Servos

SECTION I

Serial Connection

Drivers for Serial Connection



Connect ESP32-S3 to PC

Connect the ESP32-S3 board to the PC using the USB cable. If device driver does not install automatically, identify USB-to-UART bridge on your ESP32-S3 board (or external converter dongle), search for drivers in internet and install them.

Below is the list of USB to serial converter chips installed on most of the ESP32-S3 boards produced by Espressif together with links to the drivers:

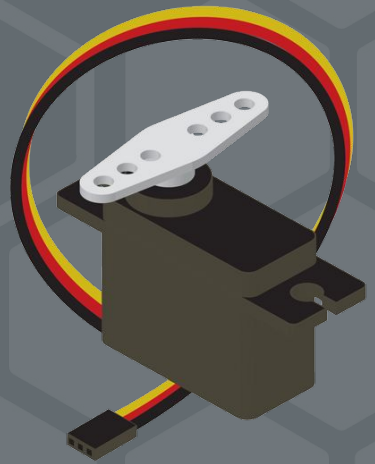
- CP210x: CP210x USB to UART Bridge VCP Drivers
- FTDI: FTDI Virtual COM Port Drivers

- If you are experiencing connectivity issues, try installing these drivers

https://docs.espressif.com/projects/esp-idf/en/latest/esp32s3/get-started/establish-serial-connection.html

SECTION I

Servo Motors

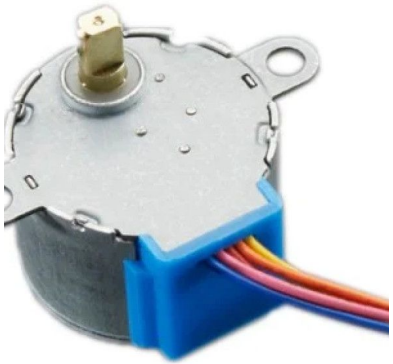


Other Motors

- Other motors
  - DC motor, Stepper motor
- Servo is utilized for precise control over location
- Control the position of a servo motor using a microcontroller or servo controller

DC motor

Stepper motor

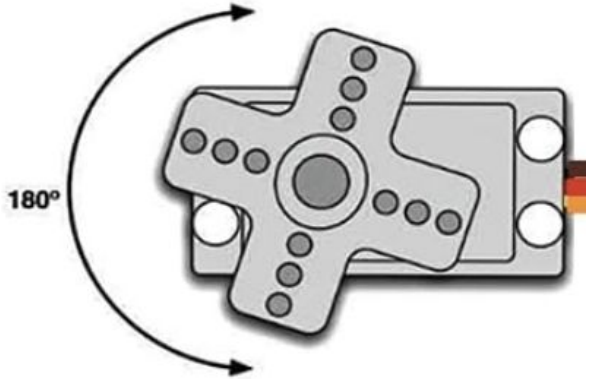


Introduction to Servo Motors

- Servo motors converts electrical signals into rotational motion
- Servos move to a specific angle given an input signal
    - Has a feedback control mechanism for precise position control
- Controller compares the motor's current position with the desired position and the motor continually adjusts its motion to reach and sustain the intended position or motion
- ~180 Degrees of rotation

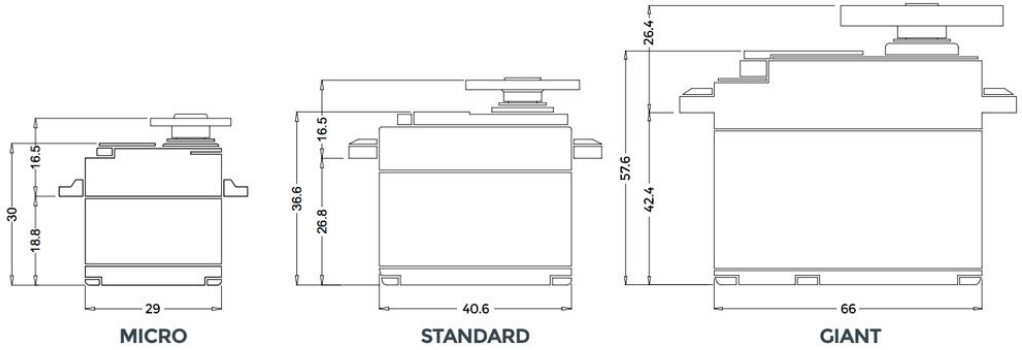
Introduction to Servo Motors

- **Closed Loop** type Servos typically rotate 180 degrees
  - Usually have an internal rotation limiting pin
  - Precise position control



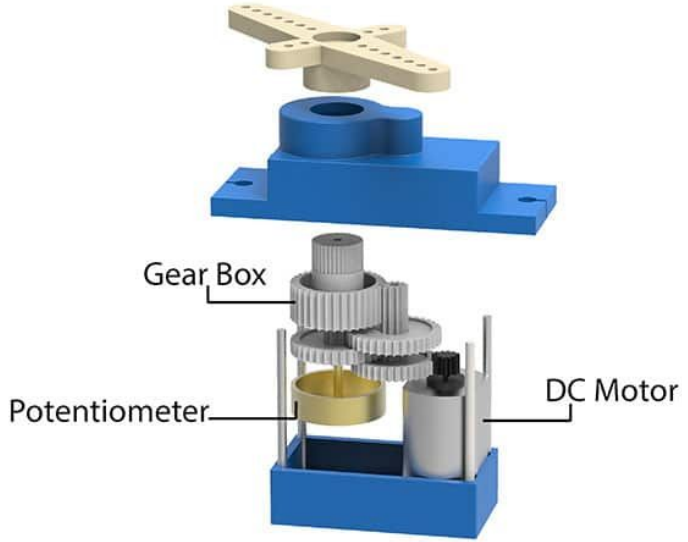
Sizes

- Micro-servo: small, lightweight
- Standard-servo: medium, more powerful
- Giant-servo: large, most powerful



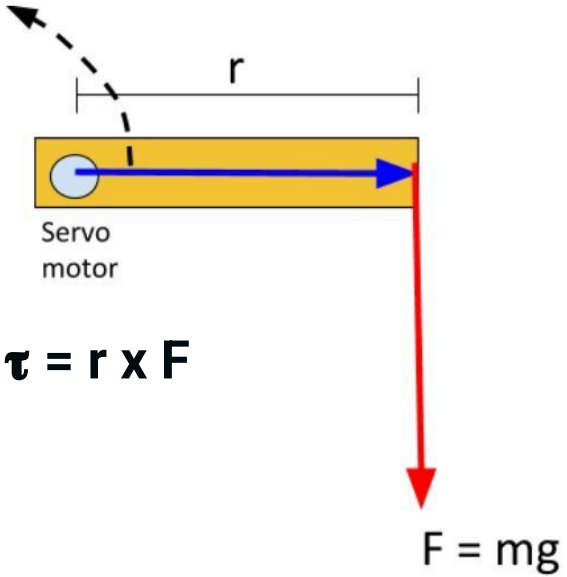
Inside the Servo Motor

- DC motor
  - Provides rotational motion
- Gearbox
  - Decrease rotational speed and increase output torque
- Position Potentiometer
  - Variable resistor that rotates with servo
- Control Circuit & Position Sensor
  - Compares input voltage to potentiometer voltage
  - Adjusts the power to the motor to reach intended position until the voltage difference between potentiometer and PWM signal is 0



Torque

- A servo's torque is the rotational force it can produce
  - Common Units: kg-cm (kilogram-centimeters), Nm (Newton-meters), lb-in (pound-inches)



Servo Motor Torque

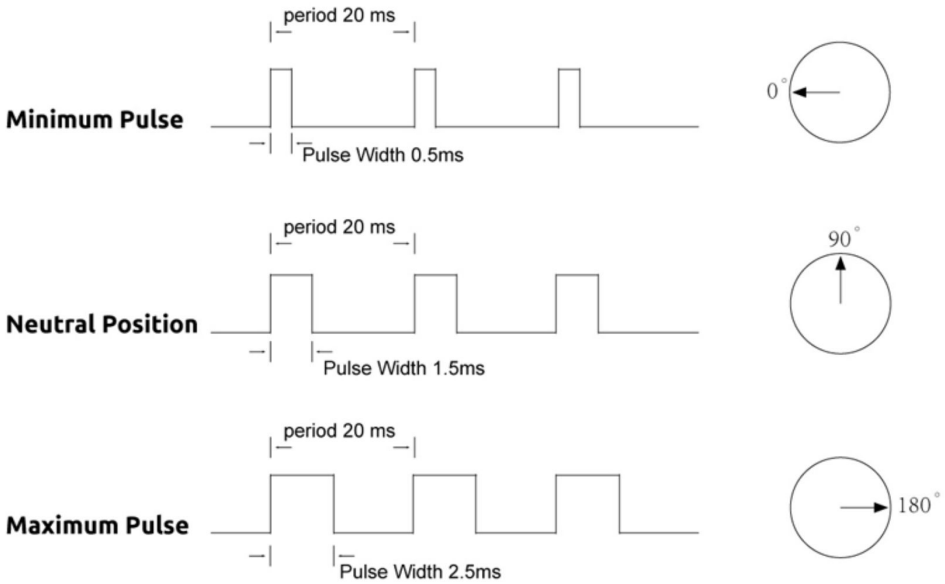
- Servos output a torque to the lever
  - This is how much force the servo can apply to a lever
- Stall torque spec: the maximum weight the servo can pull at zero speed
  - Example: SG90 Servo stall torque: 1 kg-cm
- Rated torque spec: continuous torque the servo can handle without exceeding limits

How Servos Motors are Controlled

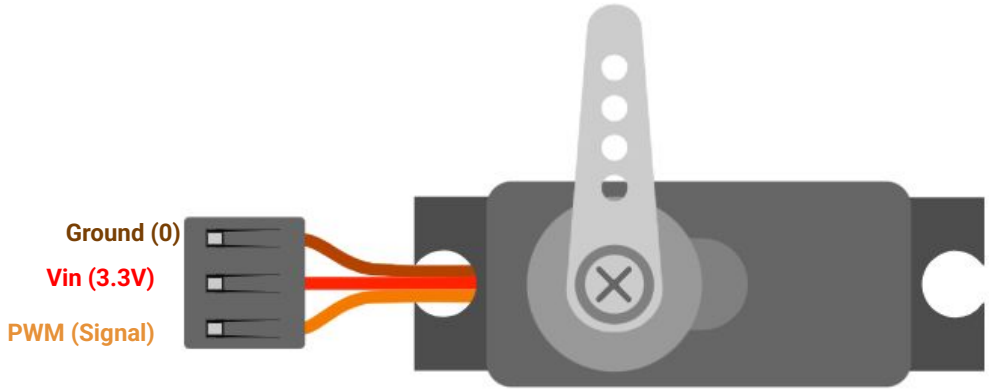
- Microcontroller (esp32) or Servo Motor Tester
- Controlled via Pulse Width Modulation (PWM) signal
- PWM signal usually ~50 Hz, a period of 20 ms
- The pulse width is varied within the signal period
  - Short pulse width moves shaft towards 0 degrees
  - Longer pulse moves shaft towards 180 degrees
    - 1ms pulse → 0° position
    - 1.5ms pulse → 90° (middle position)
    - 2ms pulse → 180° position
- The pulse is continually applied to the control lead and the internal circuit self adjusts the servo until the desired position is locked into place

Servo Positioning via PWM

- The width of the pulse determines the servo's position
- Internal positioning feedback

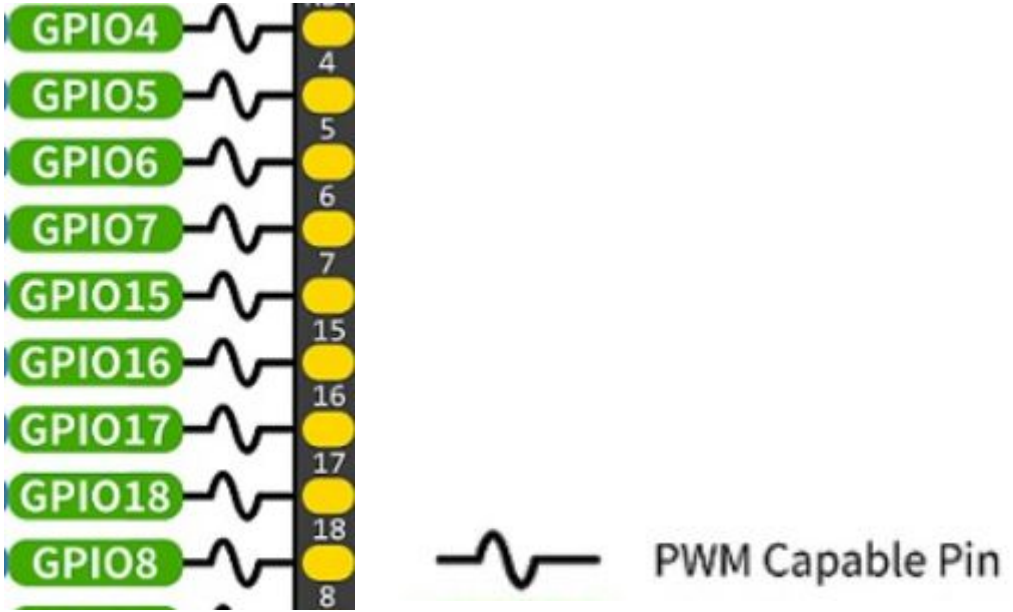


Connecting Servo Motor to ESP32



ESP32 PWM Capable Pins

- Recommended pins for ESP32S3
  - Pin 4 (GPIO4), Pin 5 (GPIO5), Pin 6 (GPIO6), Pin 7 (GPIO7), Pin 8 (GPIO8), or any other PWM capable pin



Example Applications

- Analog Clock (minute and hours hands)
- Robotic arm control
- Moving robots
- Automation (animatronics)
- Remote Controlled Cars (Steering axle)
- Active Aero Wing on car

More on Servo Motors

https://youtu.be/1WnGv-DPexc?si=uqaCo8-n--3iBQST

Servos Explained

https://www.sparkfun.com/servos

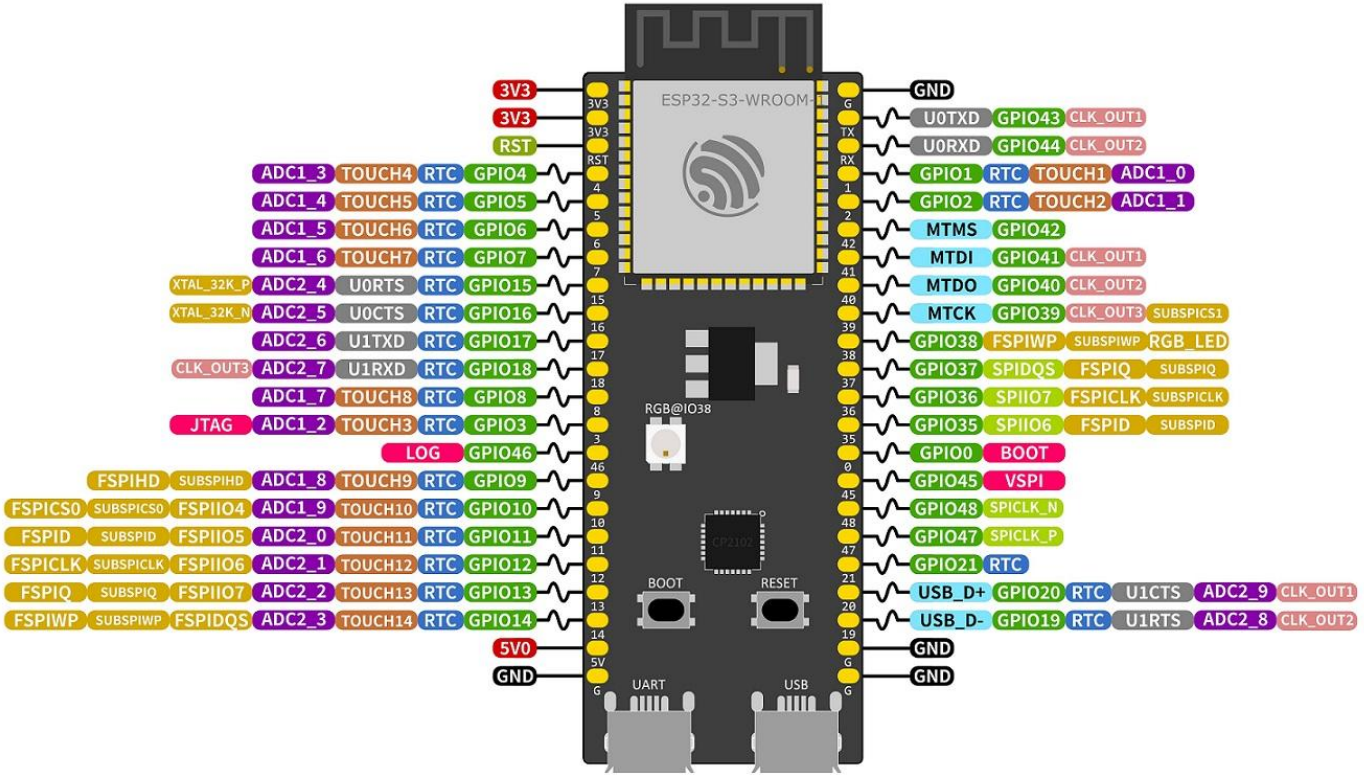
SECTION II

Setup Circuit

ESP32 Pinout

ESP32-S3-DevKitC-1

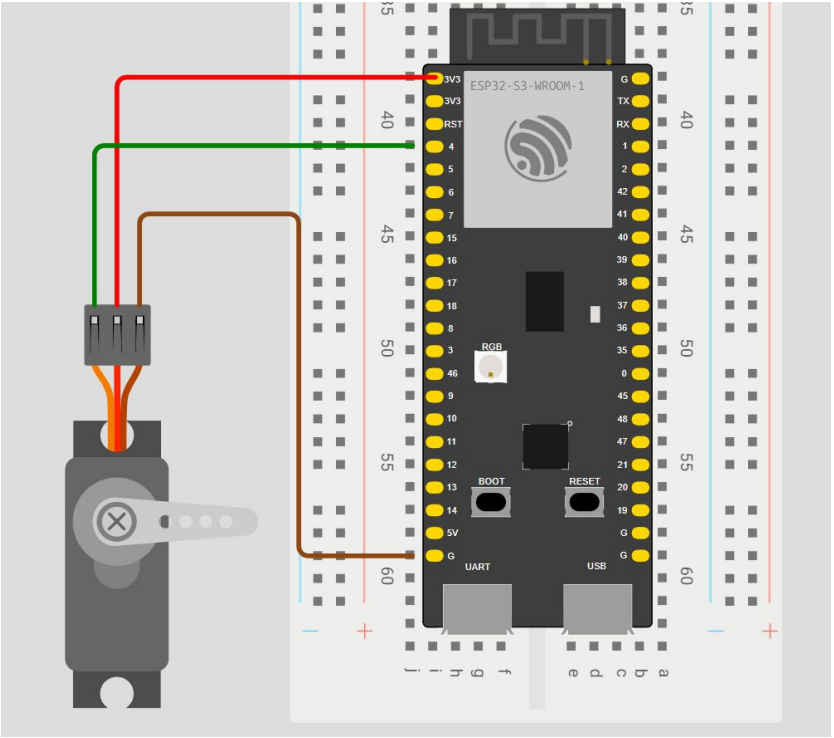
ESPRESSIF



[Pinout link]

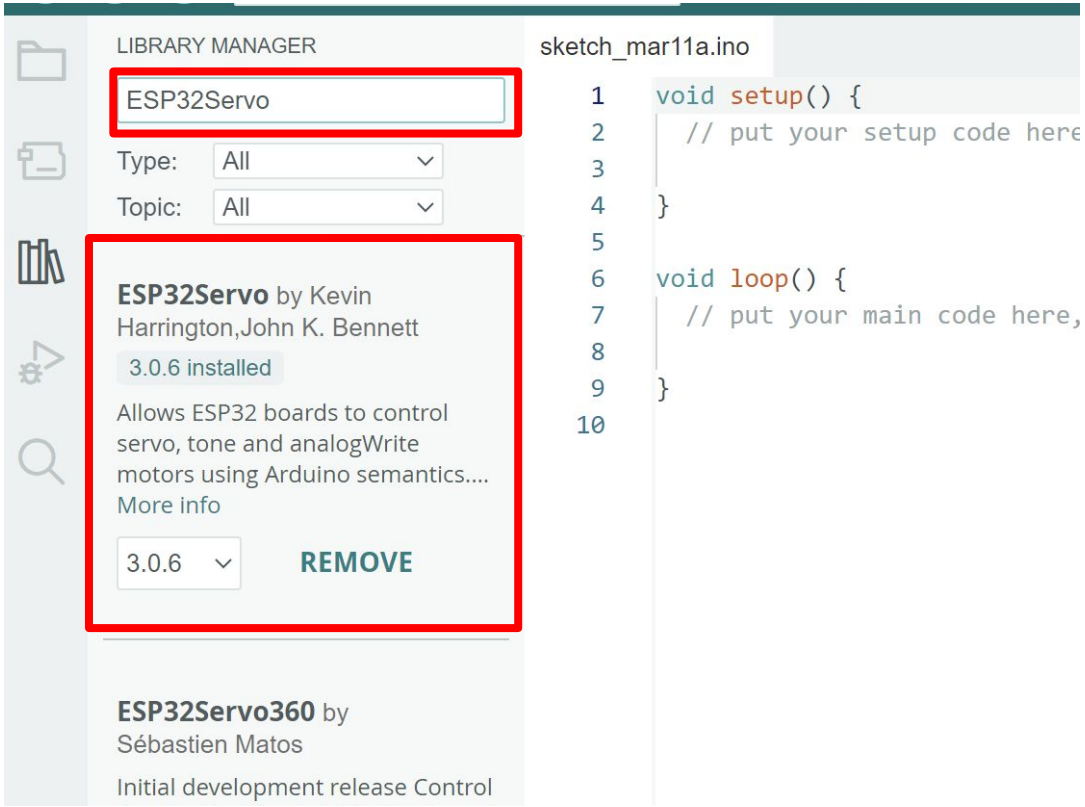
Circuit Schematic 1

- ESP32 connected to the SG90 servo



Servo Motor Library Installation

- Search for “ESP32Servo”
  - Kevin Harrington



Circuit 1 Program Code

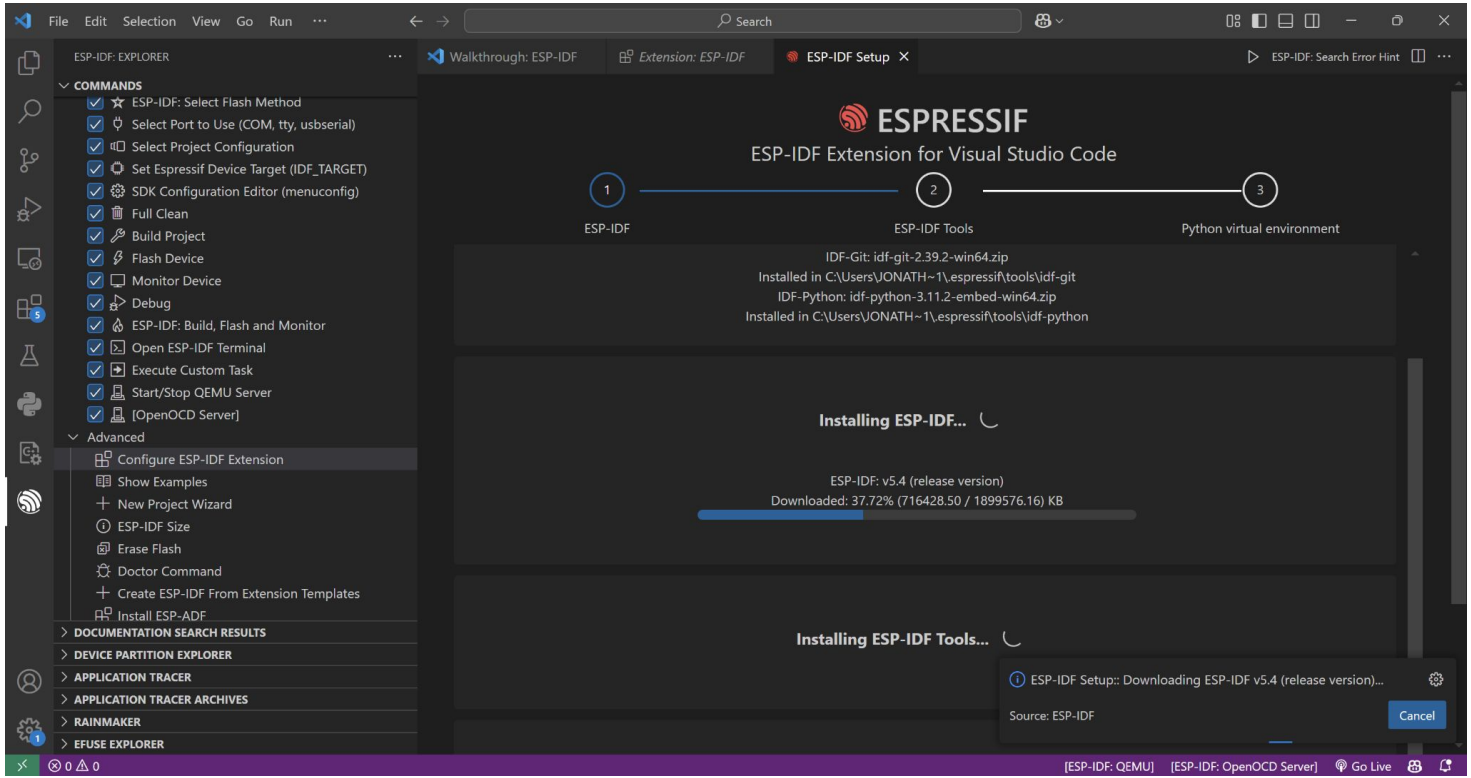
- A program to demonstrate rotating the servo across its full range of motion

```
#include <ESP32Servo.h>

Servo myServo;       // declare a Servo object
int servoPin = 4;    // define which pin controls the servo
int servoAngle = 0; // variable to define the servo starting angle
int speed = 20;      // define the delay to determine the speed of rotation

void setup() {
  Serial.begin(115200);
  myServo.attach(servoPin);
}

void loop() {
  if (Serial.available()) {            // true if we received a message via serial monitor
    servoAngle = Serial.parseInt();    // returns an int read in from Serial monitor
  }
  delay(speed);
  for (int posDegrees = servoAngle ; posDegrees <= 180 ; posDegrees++) {
    myServo.write(posDegrees);
    Serial.println(posDegrees);
    delay(speed);
  }
  for (int posDegrees = 180 ; posDegrees >= servoAngle ; posDegrees--) {
    myServo.write(posDegrees);
    Serial.println(posDegrees);
    delay(speed);
  }
}
```

SECTION III

Program the Servo

ESP-IDF Extension Installation



https://docs.espressif.com/projects/vscode-esp-idf-extension/en/latest/installation.html#installation

SECTION IV

LCD Screen

Circuit Schematic with LCD Screen

- Insert

Program code with LCD Screen

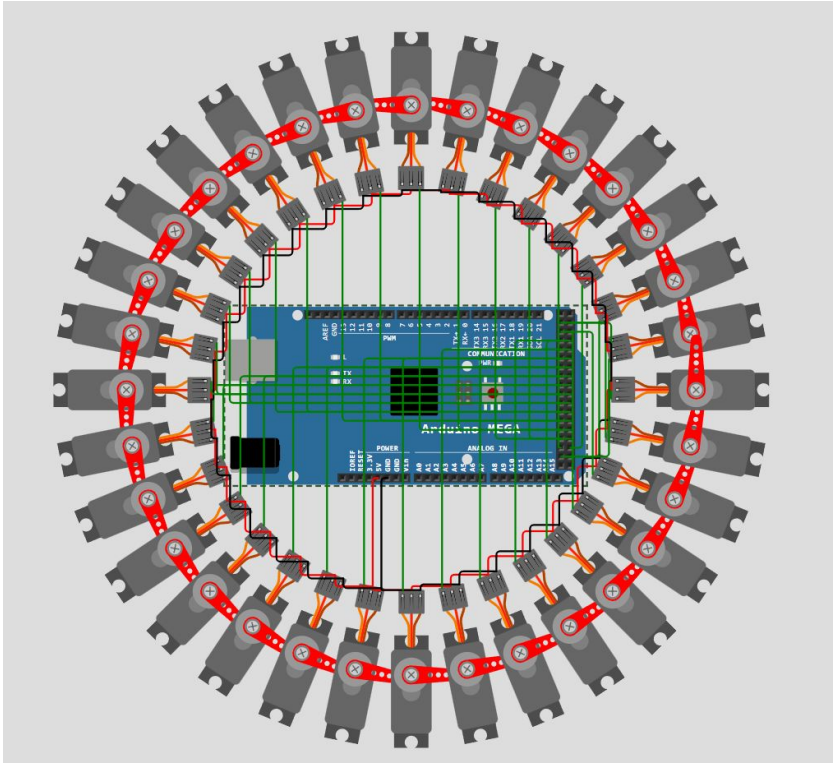
- Insert code for LCD screen

SECTION V

Potential Applications

Servo Array

- Programs servo objects to create interesting displays of motion
  - Random angle
  - Simultaneous sweeping
  - Rotating wave
  - Compass pointer



https://wokwi.com/projects/305336312628511297