Lesson 9 ADC Detect Voltage

1. Working Principle

ADC is short for A/D Converter. In microcontroller applications, the input analog signal usually needs to be converted to a digital signal that can be recognized by the microcontroller, and the technology to convert continuously changing analog signals to digital signals is called A/D conversion technology.

When the analog signal is input into the control board, it is converted into a digital signal by ADC and then numerical analysis and processing is performed to calculate the voltage value.

The path to the source code of the program is 5. MaxArm Hardware Basic Learning/Arduino Development/Game Programs/ADC Detect Voltage/ Detecting_Voltage\Detecting_Voltage.ino

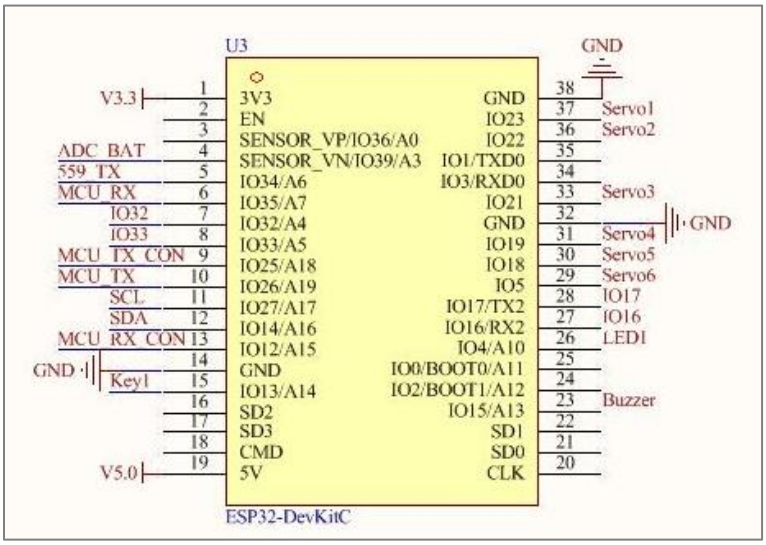
```
#define DetectingPin 39 // Detect pin

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  Serial.println("start...");
  delay(500);
}

void loop() {
  // put your main code here, to run repeatedly:
  // Read the pin value of the value, and converted by 12-bit ADC. The value range is between 0 and 4095.
  float ReadValue = analogRead(DetectingPin);
  // The full range of pin detection is 3.3V, but actually is 3.2V. Therefore, use partial voltage detection to detect circuit. The coefficient is 0.25
  // Total voltage=partial voltage/coefficient
  float VoltageValue = ((ReadValue / 4095) * 3.2) / 0.25; // Get the total voltage according to the formula.
  Serial.println(VoltageValue); // The serial port prints the total voltage.
  delay(2000);
}
```

Firstly, define ADC pin, and get the digital signal of voltage through analogRead() function, and then calculate the current voltage. Finally, print the voltage value in serial monitor.

According to the following the circuit diagram, IO39 on ESP32 expansion board has ADC function.



2. Preparation

2.1 Hardware

MaxArm robotic arm, power adapter, USB cable.

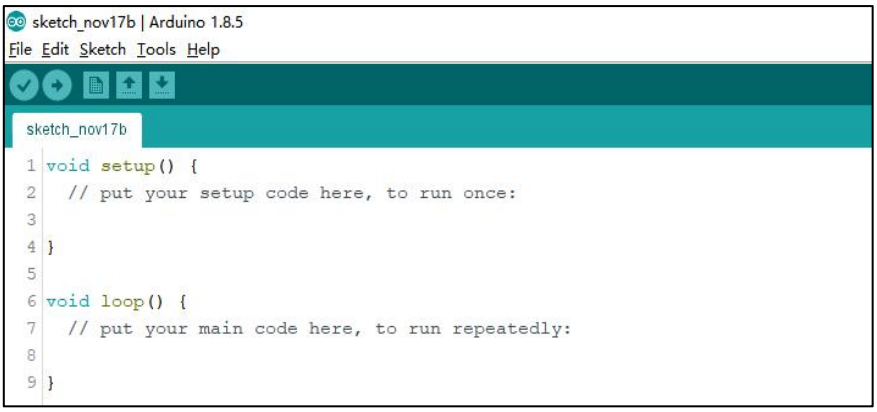
2.1 Software

Please connect MaxArm to the Arduino editor according to the tutorial in folder “4. MaxArm Underlying Program/Python Development/Lesson 1 Set Development Environment”.

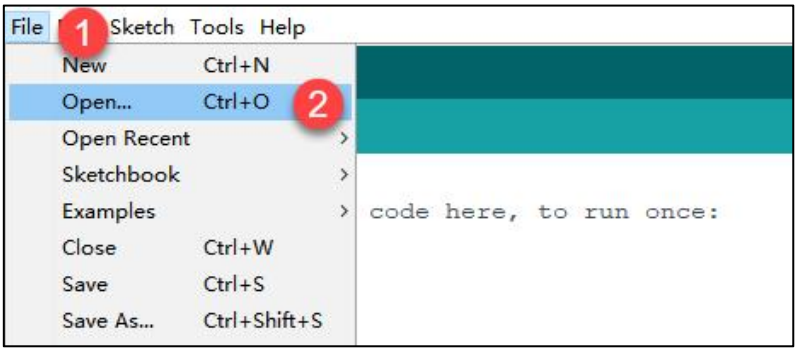
3. Program Download

1) Double click on 

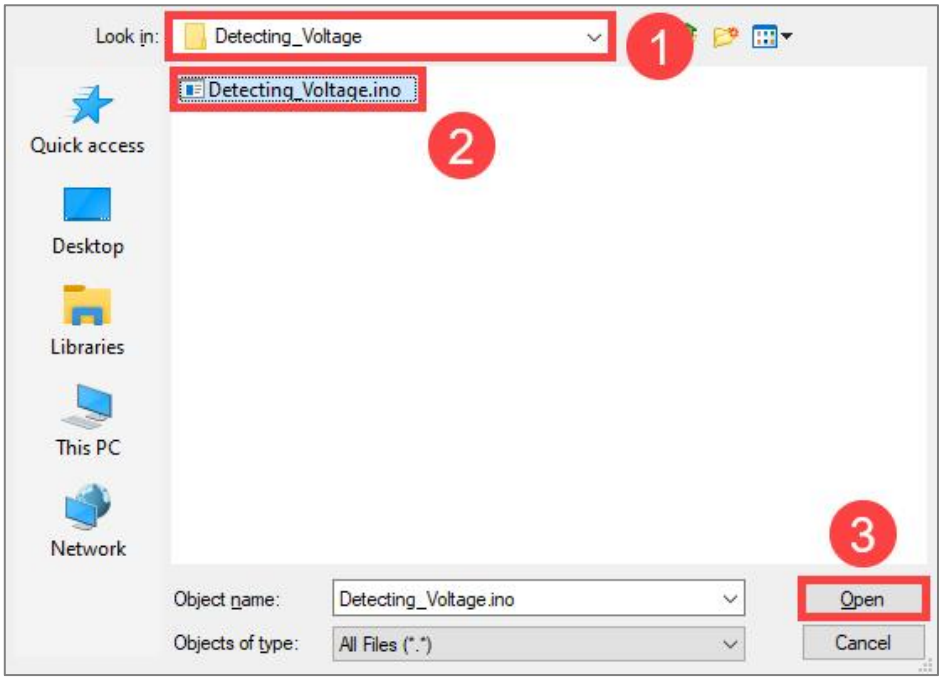
icon to open Arduino IDE.



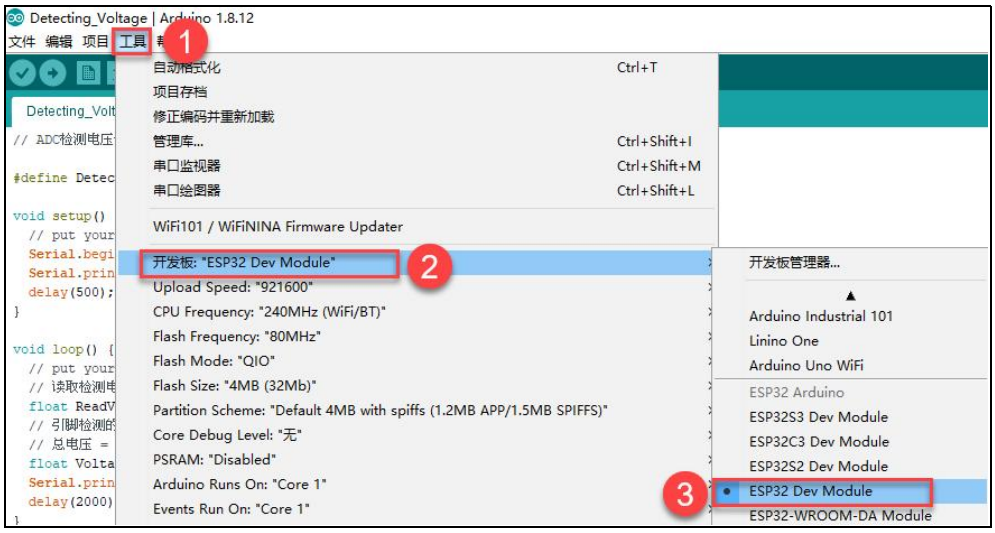
2) Click “File->Open” in turn.



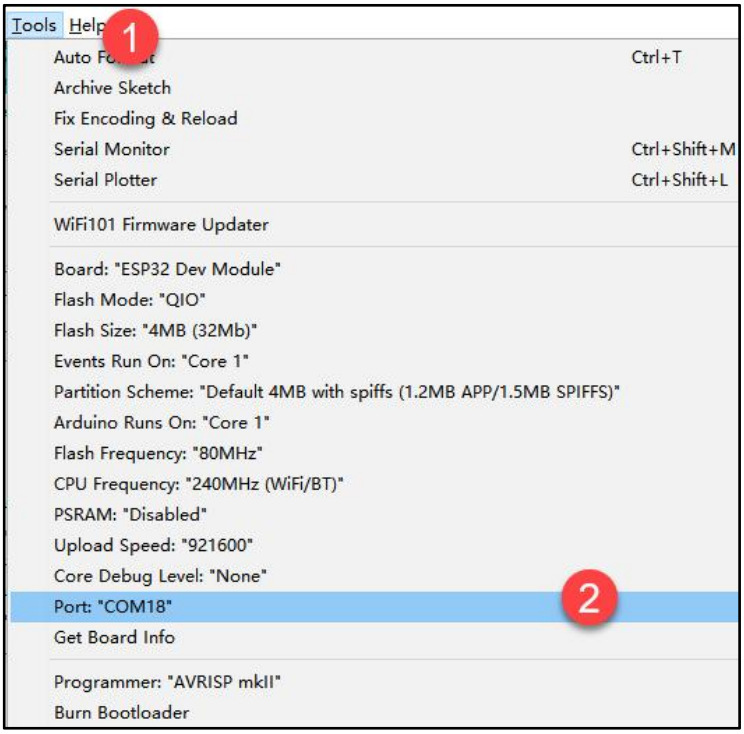
3) Select the program “**Detecting_Voltage.ino**” in the folder “5.MaxArm Hardware Basic Learning/Arduino Development/Game Programs/ADC Detect Voltage/ **Detecting_Voltage**”.



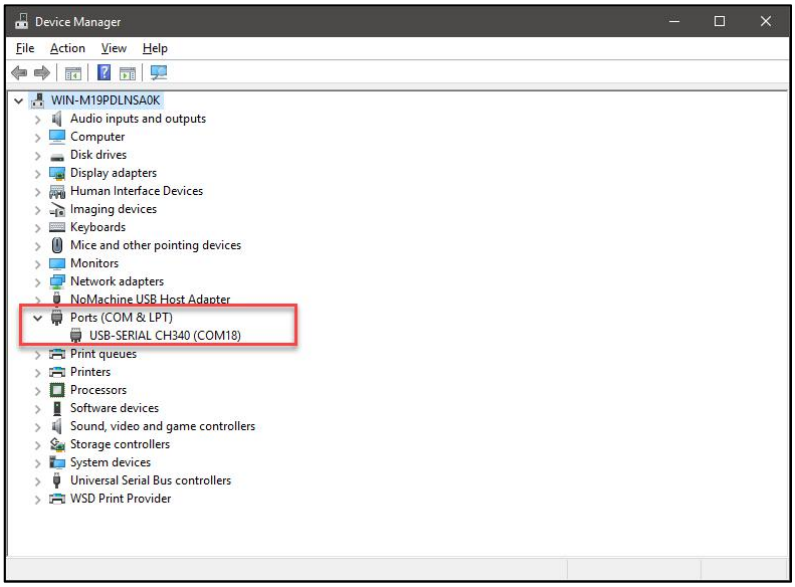
4) Check the board model. Click “Tools->Board” and select “ESP 32 Dev Module”. (If the model of development board has been configured when setting the development environment, you can skip this step.)



5) Select the corresponding port of ESP32 controller in “Tools->Port”. (Here take the port “COM5” as example. Please select the port based on your computer. If COM1 appears, please do not select because it is the system communication port but not the actual port of the development port.)

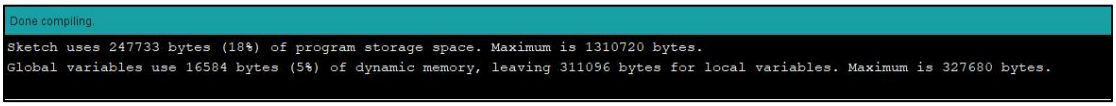


6) If you're not sure about the port number, please open the "This PC" and click "Properties->Device Manger" in turns to check the corresponding port number (the device is with CH340).

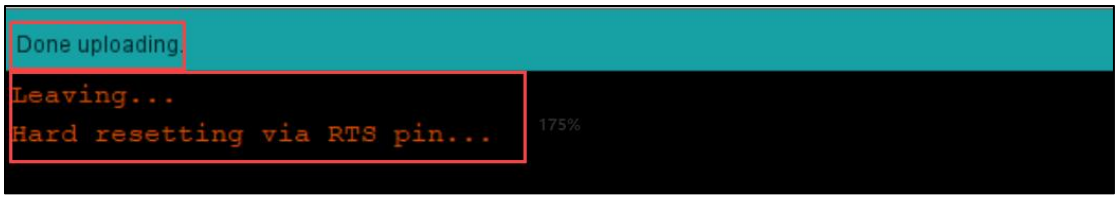


7) After selecting, confirm the board "ESP32 Dev Module" in the lower right corner and the port number "COM5" (it is an example here, please refer to the actual situation).

8) Then click on  icon to verify the program. If no error, the status area will display "Compiling->Compile complete" in turn. After compiling, the information such as the current used bytes, and occupied program storage space will be displayed.

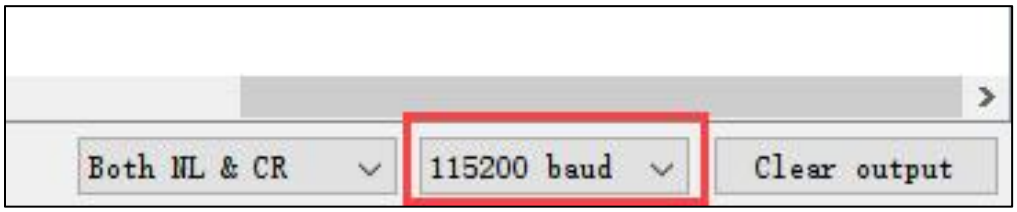


9) After compiling, click on  icon to upload the program to the development board. The status area will display "Compiling->Uploading->Complete" in turn. After uploading, the status area will stop printing the uploading information.



10) Then click on the serial monitor icon  in the upper right corner.

11) Select “115200 baud rate” in the pop-up window.



4. Project Outcome

The serial monitor will constantly print the input voltage of the robotic arm.