



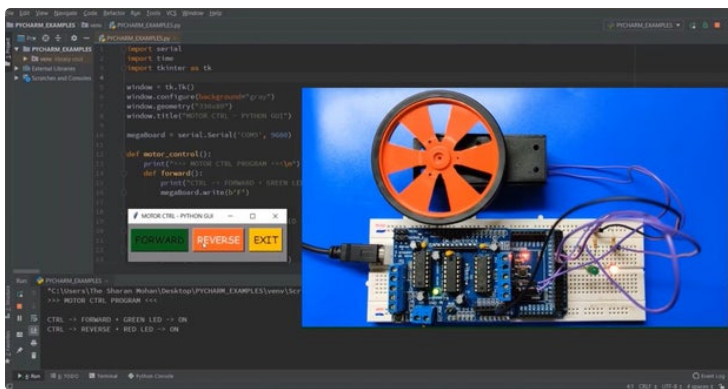
instructables

Python (pySerial) + Arduino + DC Motor



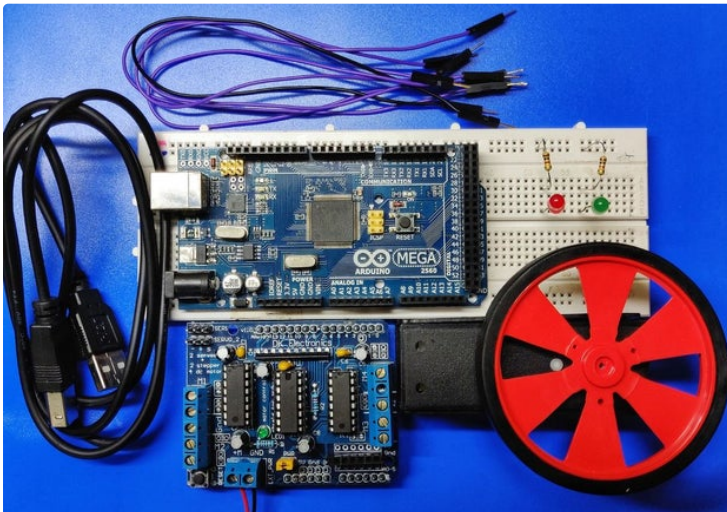
by thesharanmohan

This quick tutorial shows the simple operation of a DC motor using a Python GUI. To make Python to communicate with an Arduino board we will be using the pySerial package. pySerial is a Python library which provides support for serial connections over a variety of different devices.



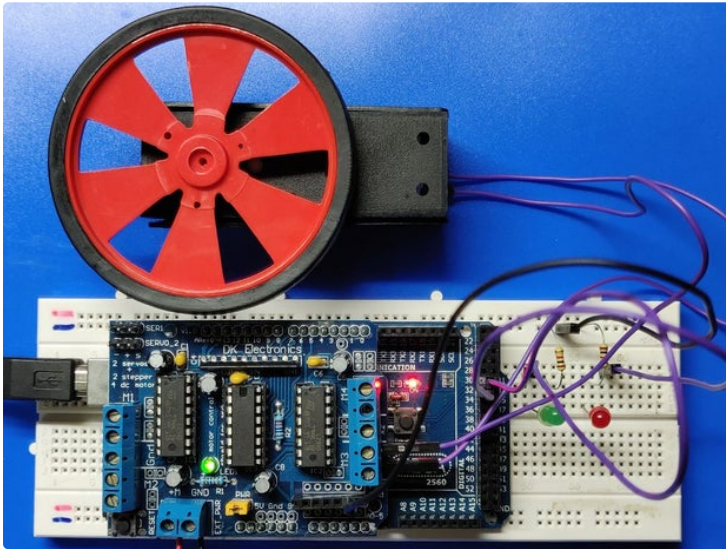
Step 1: Hardware

Adafruit motor shield, Arduino board (Mega), DC motor, 1k ohm resistors (2), LEDs (2), hook-up wires and breadboard.

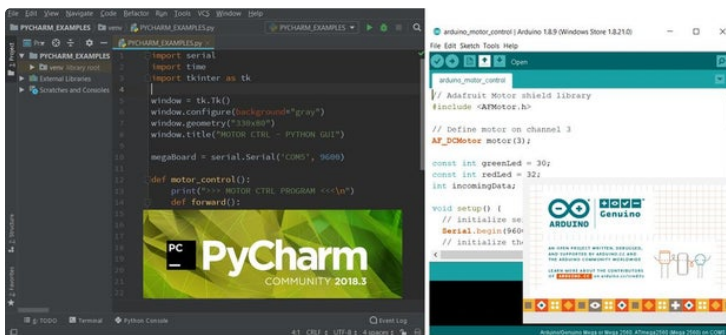


Step 2: Hardware Setup

In this setup,
green LED -> Pin 30 of Arduino board
red LED -> Pin 32 of Arduino board
DC Motor -> Channel 3 (M3) of motor shield

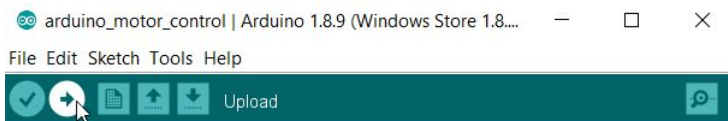


Step 3: Software - Arduino IDE, PyCharm IDE



Step 4: Arduino IDE

Connect the desired Arduino board to PC (in this case I'm using an Arduino Mega). Open Arduino IDE and select the appropriate COM port and board. The code below is uploaded to the Arduino board by clicking on the Upload button.



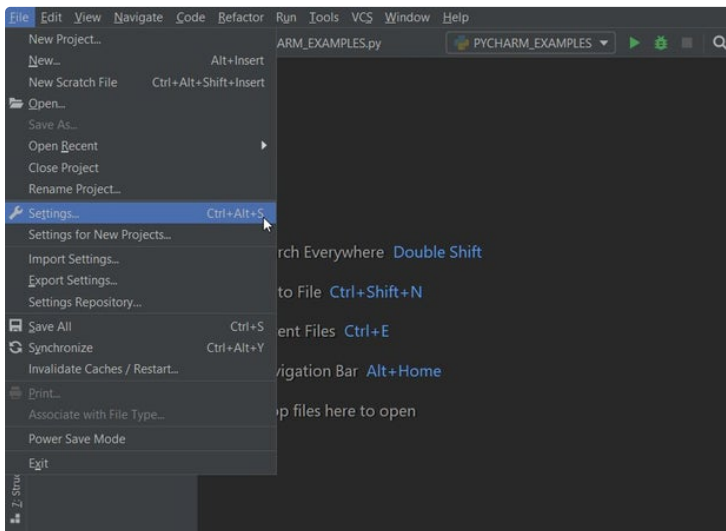
Step 5: Arduino IDE - Code Part 1

```
1 // Adafruit Motor shield library
2 #include <AFMotor.h>
3
4 // Define motor on channel 3
5 AF_DCMotor motor(3);
6
7 const int greenLed = 30;
8 const int redLed = 32;
9 int incomingData;
10
11 void setup() {
12   // initialize serial communication
13   Serial.begin(9600);
14   // initialize the LED pin as an output
15   pinMode(greenLed, OUTPUT);
16   pinMode(redLed, OUTPUT);
17
18   // speed - Valid values for 'speed' are between 0 and 255 with 0 being off and 255 as full throttle
19   motor.setSpeed(200);
20   // RELEASE - Stop the motor. This removes power from the motor and is equivalent to setSpeed(0)
21   motor.run(RELEASE);
22 }
23
```

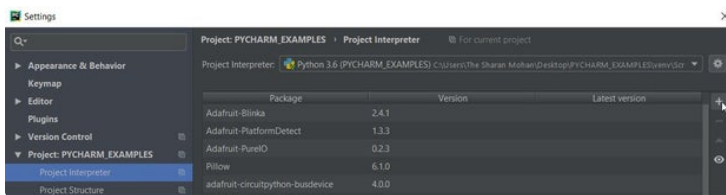
Step 6: Arduino IDE - Code Part 2

```
24 void loop() {
25   // see if there's incoming serial data
26   if (Serial.available() > 0) {
27
28     incomingData = Serial.read();
29
30     if (incomingData == 'F') {
31       motor.run(FORWARD);
32       digitalWrite(greenLed, HIGH);
33       digitalWrite(redLed, LOW);
34     }
35
36     if (incomingData == 'R') {
37       motor.run(BACKWARD);
38       digitalWrite(greenLed, LOW);
39       digitalWrite(redLed, HIGH);
40     }
41
42     if (incomingData == 'Q') {
43       motor.run(RELEASE);
44       digitalWrite(greenLed, LOW);
45       digitalWrite(redLed, LOW);
46     }
47   }
48 }
```

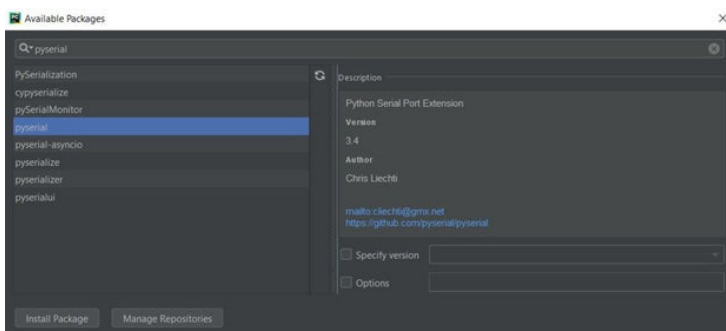
Step 7: Open PyCharm IDE and Click on File -> Settings.



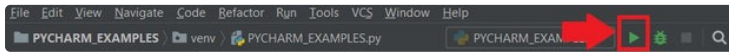
Step 8: Under Project, Select Project Interpreter and Click on the “+” Icon.



Step 9: In the Search Bar, Type Pyserial and Click on Install Package.



Step 10: The Python Code Below Is Run on PyCharm IDE.



Step 11: Python Code - Part 1

NOTE: Make sure the same COM port number is used in the Python code.

Tkinter: <https://docs.python.org/3/library/tkinter.html#tkinter-modules>

References:

pySerial: <https://pyserial.readthedocs.io/en/latest/shortintro.html>

```
1  import serial
2  import time
3  import tkinter as tk
4
5  window = tk.Tk()
6  window.configure(background="gray")
7  window.geometry("330x80")
8  window.title("MOTOR CTRL - PYTHON GUI")
9
10 megaBoard = serial.Serial('COM5', 9600)
11
12 def motor_control():
13     print(">>> MOTOR CTRL PROGRAM <<<\n")
14     def forward():
15         print("CTRL -> FORWARD + GREEN LED -> ON")
16         megaBoard.write(b'F')
17
18     def reverse():
19         print("CTRL -> REVERSE + RED LED -> ON")
20         megaBoard.write(b'R')
```

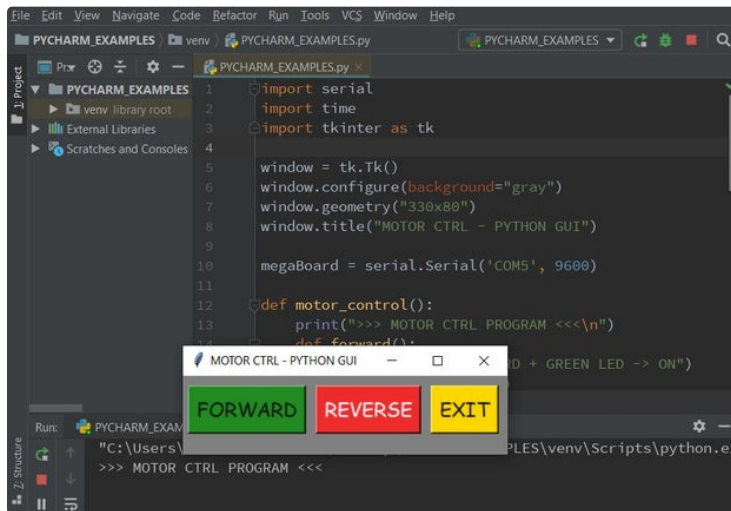
Step 12: Python Code - Part 2

```
21
22 def quit():
23     print("\n** END OF PROGRAM **")
24     megaBoard.write(b'Q')
25     megaBoard.close()
26     window.destroy()
27
28 b1 = tk.Button(window, text="FORWARD", command=forward, bg="forest green", fg="gray7", font=("Comic Sans MS", 15))
29
30 b2 = tk.Button(window, text="REVERSE", command=reverse, bg="firebrick2", fg="ghost white", font=("Comic Sans MS", 15))
31
32 b3 = tk.Button(window, text="EXIT", command=quit, bg="gold", fg="gray7", font=("Comic Sans MS", 15))
33
34 b1.grid(row=1, column=0, padx=5, pady=10)
35 b2.grid(row=1, column=1, padx=5, pady=10)
36 b3.grid(row=1, column=2, padx=5, pady=10)
37
38 window.mainloop()
39
40 time.sleep(2)
41 motor_control()
```

Step 13: Final

A simple GUI opens with 3 buttons – **FORWARD**, **REVERSE** and **EXIT**.

Depending on the motor connection wiring, the motor runs in the desired direction with the click of the FORWARD or REVERSE button. The EXIT button closes the serial port and ends the program execution.



Step 14: Video

<https://www.youtube.com/watch?v=9xUVIkRJ1rU>



Thank you for sharing how to do this :)