Welcome!

Thank you for purchasing our *AZ-Delivery MG995 Micro Servomotor*. On the following pages, you will be introduced to how to use and set-up this handy device.

Have fun!

Table of Contents

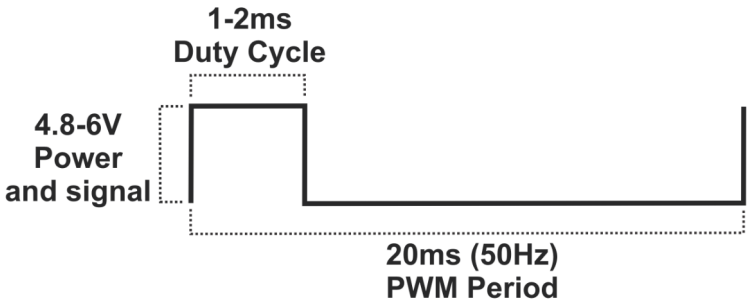
Introduction....3
Specifications....4
The pinout....5
How to set-up Arduino IDE....6
How to set-up the Raspberry Pi and Python....10
Connecting the servo motor with Atmega328p....11
Sketch examples....12
Connecting the module with Raspberry Pi....14
Libraries and tools for Python....15
Python script....16

Introduction

The MG995 is a small and light micro servo-motor that has aluminum metal gears, a coreless motor, a double ball bearing and a high power output.

Servo motors can be commanded to go to a specific position and so are the good choice when accurate positioning is needed, such as for turning the front wheels on an RC model for steering or pivoting a sensor to look around on a robotic vehicle.

Servos expect to see a pulse on their PWM pin every 20mSec. The pulse is active HIGH and the width of the pulse determines the position(angle) of the servos shaft. The pulse can vary between 1mSec and 2mSec. A 1mSec pulse positions the shaft at 0 degrees. A 1.5mSec pulse positions the shaft at 90 degrees (centered in its range). A 2mSec pulse positions the shaft at 180 degrees. Pulses with values between these can be used to position the shaft arbitrarily.



The MG995 has many applications such as Radio Controled Airplanes, Helicopters, Quadcopters, Smart robots and many others.

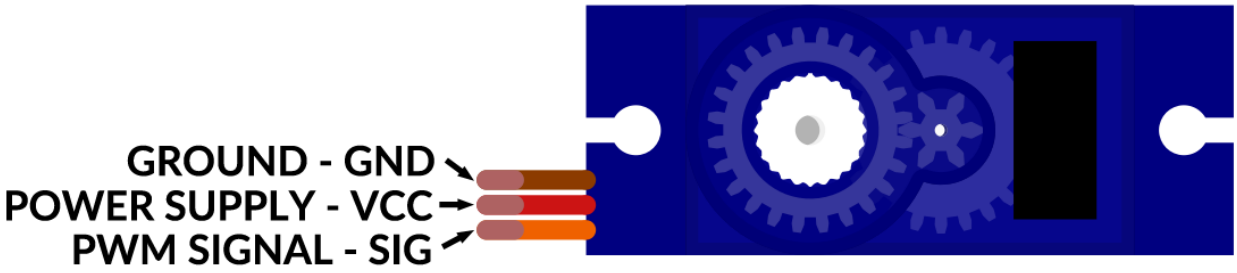
Specifications

Operating voltage	4.8V - 7.2V
Operating current (movement)	100-250mA
Operating current (stall)	700mA
Operating speed	0.2s/60° (4.8V), 0.16s/60° (6V)
Control interface	Analog
Stall torque	10kg/cm(4.8V), 12kg/cm(6V)
Dead band width	5µs
Maximum angle of rotation	120°
Operating temperature	0 - 55℃
Weight	55g
Dimensions	40x19x42mm (1.5x0.7x1.6in)

MG995 Metal Gear Servo Motor is a high-speed standard servo can rotate approximately 180 degrees (60 in each direction) used for airplane, helicopter, RC-cars and many RC model. Provides 10kg/cm at 4.8V, and 12kgcm at 6V.

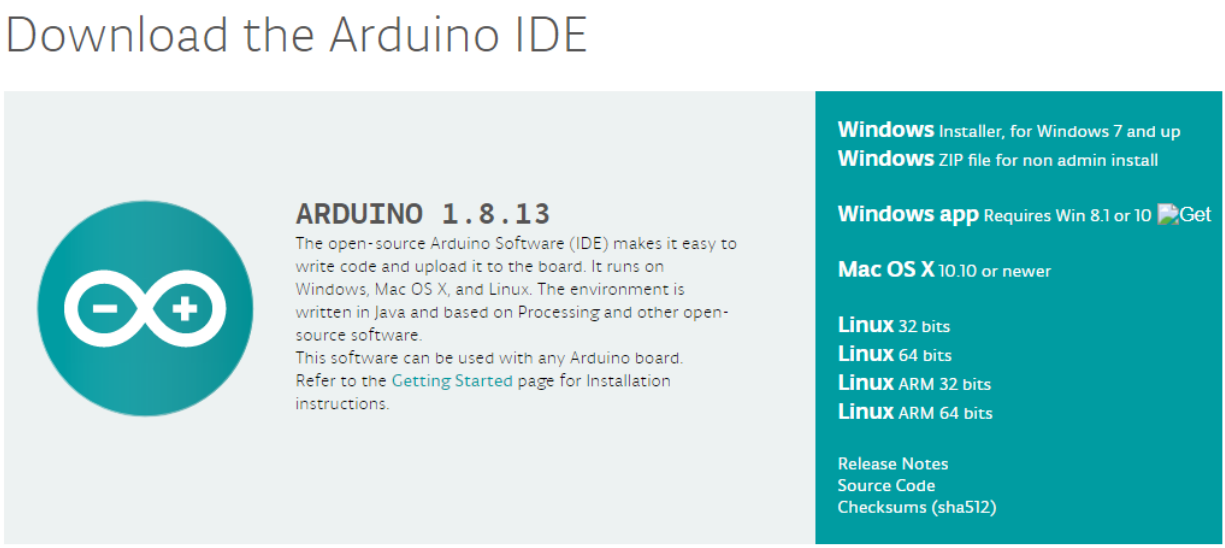
The pinout

The MG995 Micro Servomotor has three pins. The pinout is shown on the following image:



How to set-up Arduino IDE

If the Arduino IDE is not installed, follow the *link* and download the installation file for the operating system of choice. The Arduino IDE version used for this eBook is **1.8.13.**



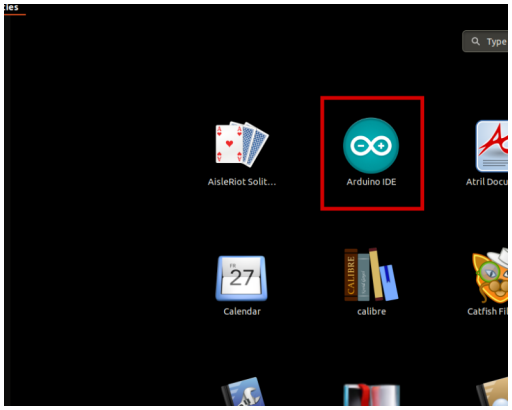
For *Windows* users, double click on the downloaded *.exe* file and follow the instructions in the installation window.

For *Linux* users, download a file with the extension *.tar.xz*, which has to be extracted. When it is extracted, go to the extracted directory and open the terminal in that directory. Two *.sh* scripts have to be executed, the first called *arduino-linux-setup.sh* and the second called *install.sh*.

To run the first script in the terminal, open the terminal in the extracted directory and run the following command:
sh arduino-linux-setup.sh user_name
user_name - is the name of a superuser in the Linux operating system. A password for the superuser has to be entered when the command is started. Wait for a few minutes for the script to complete everything.

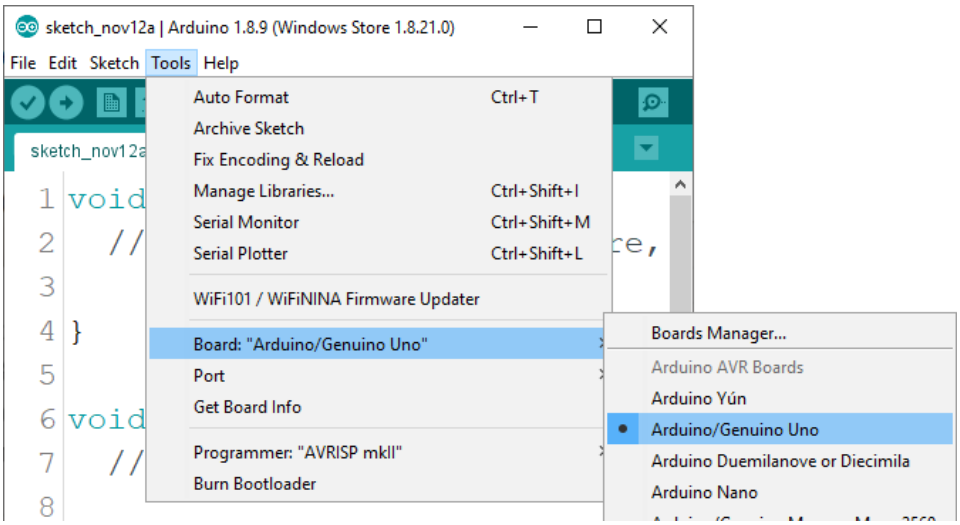
The second script called *install.sh*, has to be used after the installation of the first script. Run the following command in the terminal (extracted directory): **sh install.sh**

After the installation of these scripts, go to the *All Apps*, where the *Arduino IDE* is installed.



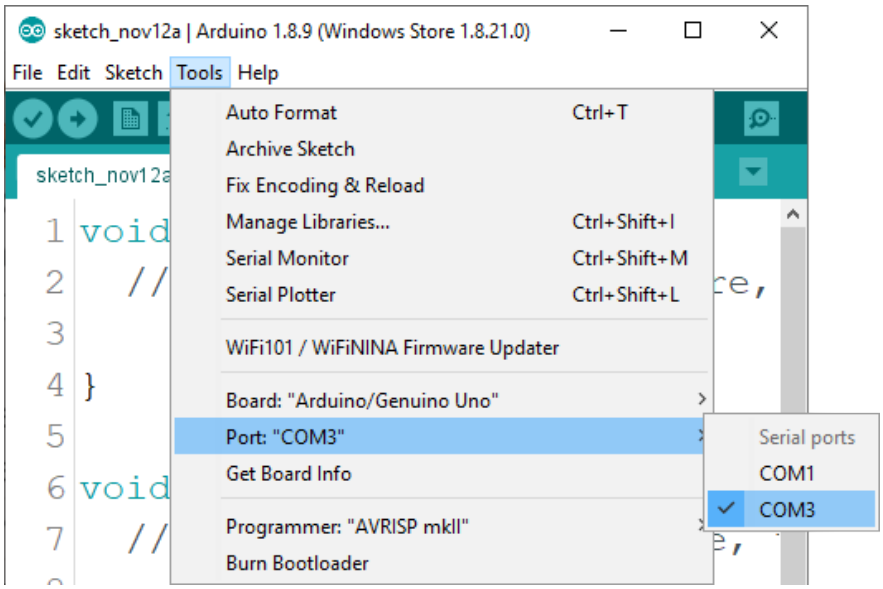
Almost all operating systems come with a text editor preinstalled (for example, *Windows* comes with *Notepad*, *Linux Ubuntu* comes with *Gedit*, *Linux Raspbian* comes with *Leafpad*, etc.). All of these text editors are perfectly fine for the purpose of the eBook.

Next thing is to check if your PC can detect an Atmega328p board. Open freshly installed Arduino IDE, and go to:
Tools > Board > {your board name here}
{your board name here} should be the *Arduino/Genuino Uno*, as it can be seen on the following image:



The port to which the Atmega328p board is connected has to be selected. Go to: *Tools > Port > {port name goes here}*
and when the Atmega328p board is connected to the USB port, the port name can be seen in the drop-down menu on the previous image.

If the Arduino IDE is used on Windows, port names are as follows:



For *Linux* users, for example port name is */dev/ttyUSBx*, where *x* represents integer number between *0* and *9*.

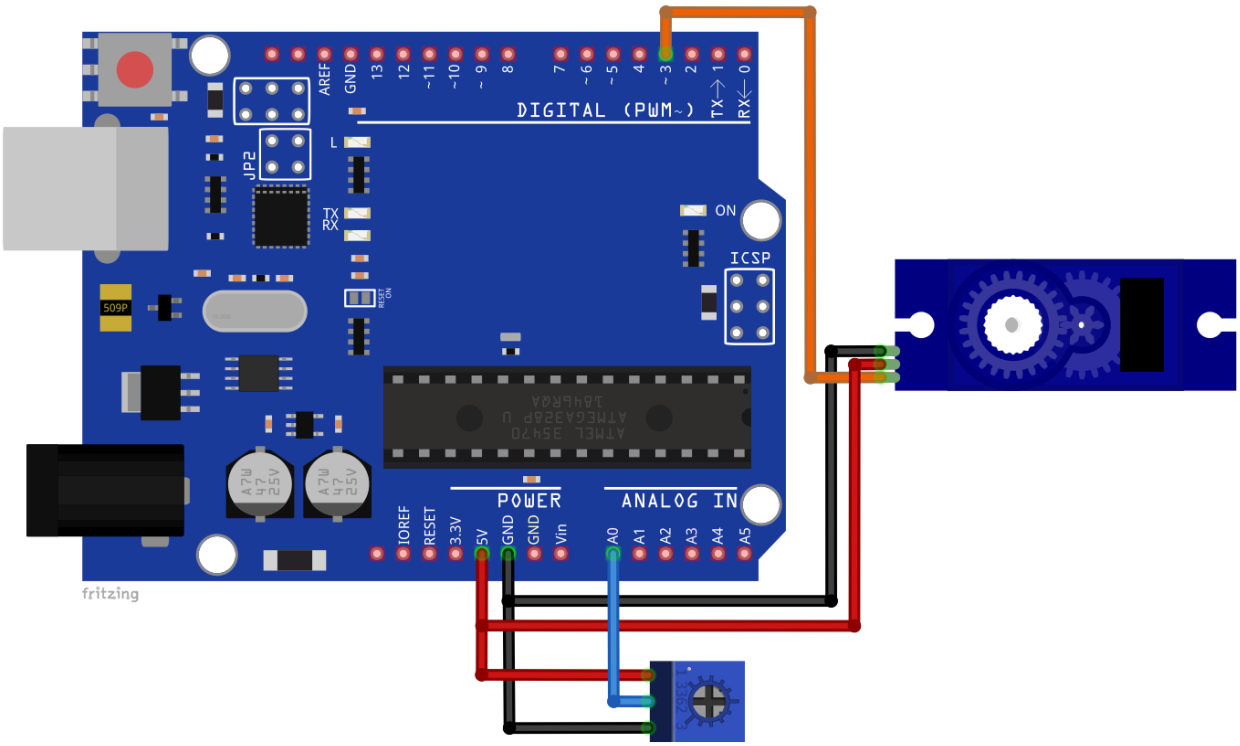
How to set-up the Raspberry Pi and Python

For the Raspberry Pi, first the operating system has to be installed, then everything has to be set-up so that it can be used in the *Headless* mode. The *Headless* mode enables remote connection to the Raspberry Pi, without the need for a *PC* screen Monitor, mouse or keyboard. The only things that are used in this mode are the Raspberry Pi itself, power supply and internet connection. All of this is explained minutely in the free eBook: *Raspberry Pi Quick Startup Guide*

The *Raspbian* operating system comes with *Python* preinstalled.

Connecting the servo motor with Atmega328p

Connect the servo motor with the Atmega328p as shown on the following image:



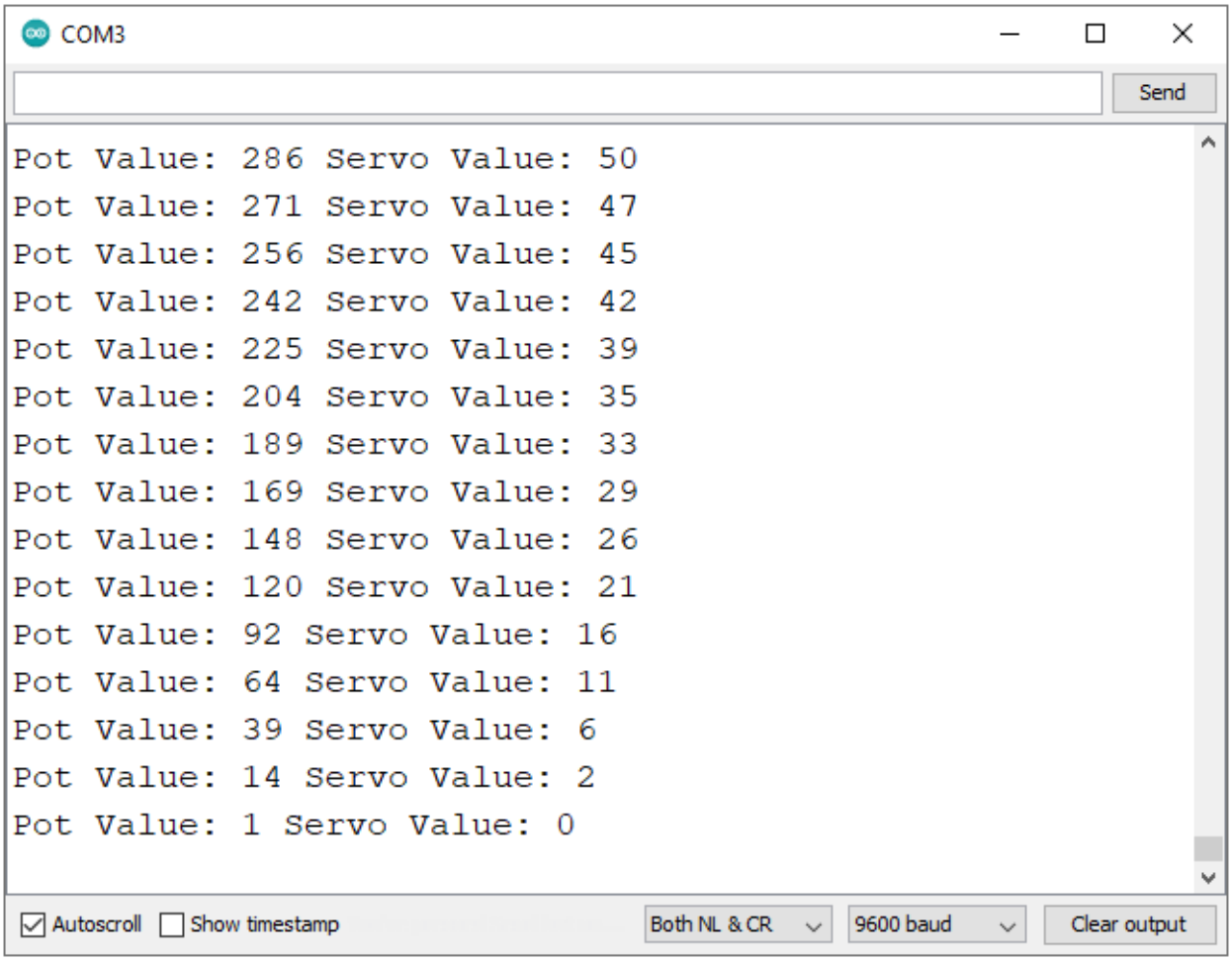
Motor pin	Mc pin	Wire color
GND	GND	**Black wire**
VCC	5V	**Red wire**
SIG	D3	**Orange wire**
Potentiometer pin	**Mc pin**	
P1	5V	**Red wire**
P2 (Center pin)	A0	**Blue wire**
P3	GND	**Black wire**

Use a potentiometer on pin A0 to command a servo attached to pin 3 to move to a specific position. The Servo `MIN_VALUE` and `MAX_VALUE` in the sketch can be adjusted to avoid hitting the servo stops.

Sketch examples

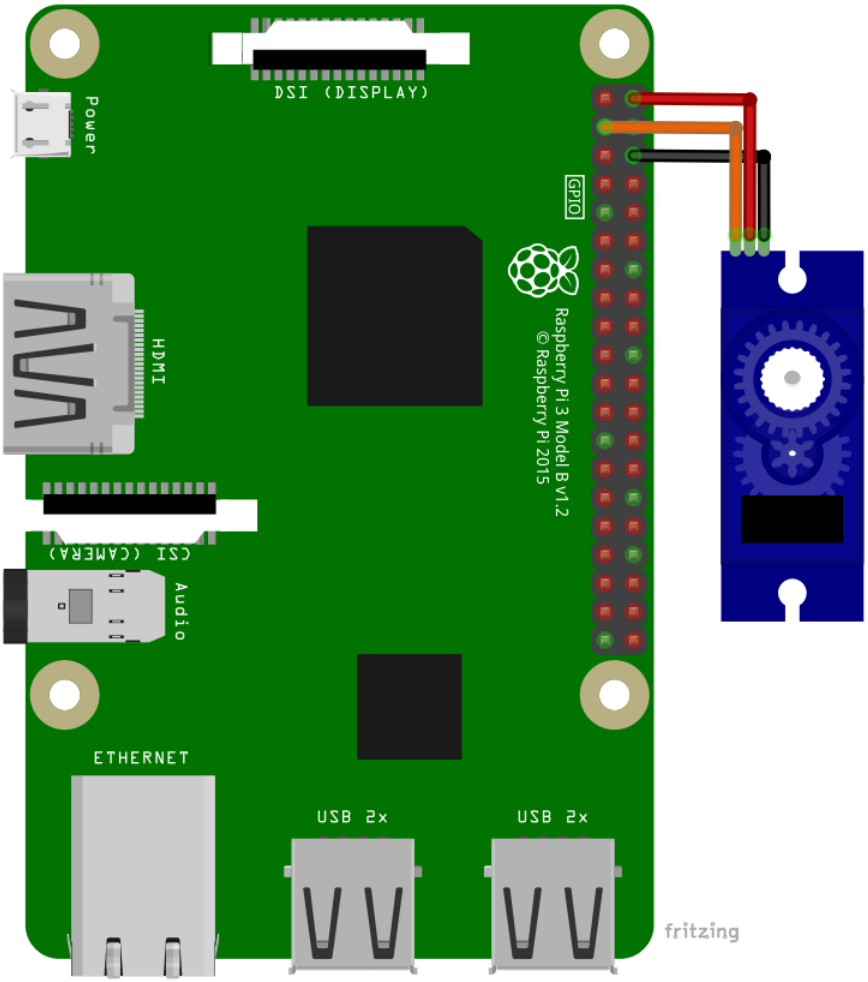
```
#include "Servo.h"
#define SERVO_PIN 3
#define POT_PIN A0
#define MIN_VALUE 0   // Minimum Servo position
#define MAX_VALUE 180 // Maximum Servo position

Servo servo;
int value_pot = 0;
int value_servo = 0;
int value_servo_old = 0;

void setup()
{
  servo.attach(SERVO_PIN);
  Serial.begin (9600);
}

void loop()
{
  value_pot = analogRead(POT_PIN);
  value_servo = map(value_pot, 0, 1023, MIN_VALUE, MAX_VALUE);
  if (value_servo != value_servo_old) {
    servo.write(value_servo);
    Serial.print("Pot Value: ");
    Serial.print(value_pot);
    Serial.print(" Servo Value: ");
    Serial.println(value_servo);
    value_servo_old = value_servo;
    delay(25);
  }
}
```

Upload the sketch to the Atmega328p and run the Serial Monitor (*Tools* > *Serial Monitor*). The result should look like as on the following image:



Connecting the module with Raspberry Pi

Connect the servomotor with the Raspberry Pi as shown on the following image:



Motor pin	Raspberry Pi pin	Physical pin	Wire color
VCC	5V	2	**Red wire**
SIG	GPIO2	3	**Orange wire**
GND	GND	6	**Black wire**

Libraries and tools for Python

To use the module with the Raspberry Pi, the library *RPi.GPIO* has to be installed. If the library is already installed, running the installation command only updates the library to a newer version.

To install the library, open the terminal and run the following commands, one by one:

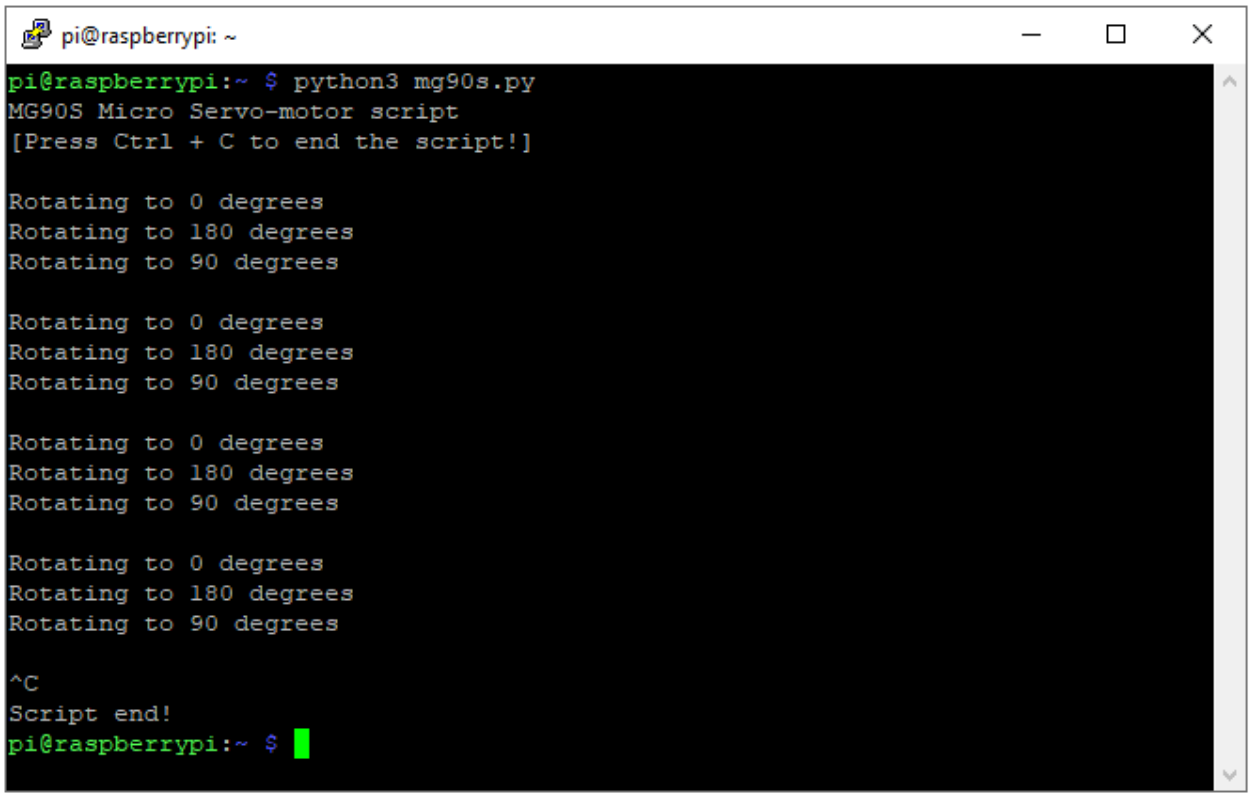
```
sudo apt-get update && sudo apt-get upgrade
sudo apt-get install python3-rpi.gpio
```

Python script

```
import RPi.GPIO as GPIO
import time

servo_pin = 2
GPIO.setmode(GPIO.BCM)
GPIO.setup(servo_pin,GPIO.OUT)

pwm = GPIO.PWM(servo_pin,50) # 50 Hz (20 ms PWM period)

pwm.start(7) # start PWM by rotating to 90 degrees

print('MG995 Micro Servo-motor script')
print('[Press Ctrl + C to end the script!]\n')

try:
  while True:
    pwm.ChangeDutyCycle(2.0)
    print('Rotating to 0 degrees')
    time.sleep(2)
    pwm.ChangeDutyCycle(12.0)
    print('Rotating to 180 degrees')
    time.sleep(2)
    pwm.ChangeDutyCycle(7.0)
    print('Rotating to 90 degrees\n')
    time.sleep(2)
    pwm.ChangeDutyCycle(0)

except KeyboardInterrupt:
    print('\nScript end!')
    pwm.stop()
    GPIO.cleanup()
```

Save the script by the name *mg995.py*. To run the script open the terminal in the directory where the script is saved and run the following command: **python3 mg995.py**

The result should look like as on the following image:



To stop the script press ‘CTRL + C’ on the keyboard.

Now it is the time to learn and make your own projects. You can do that with the help of many example scripts and other tutorials, which can be found on the Internet.

If you are looking for the high quality microelectronics and accessories, AZ-Delivery Vertriebs GmbH is the right company to get them from. You will be provided with numerous application examples, full installation guides, eBooks, libraries and assistance from our technical experts.

https://az-delivery.de

Have Fun!

Impressum

https://az-delivery.de/pages/about-us