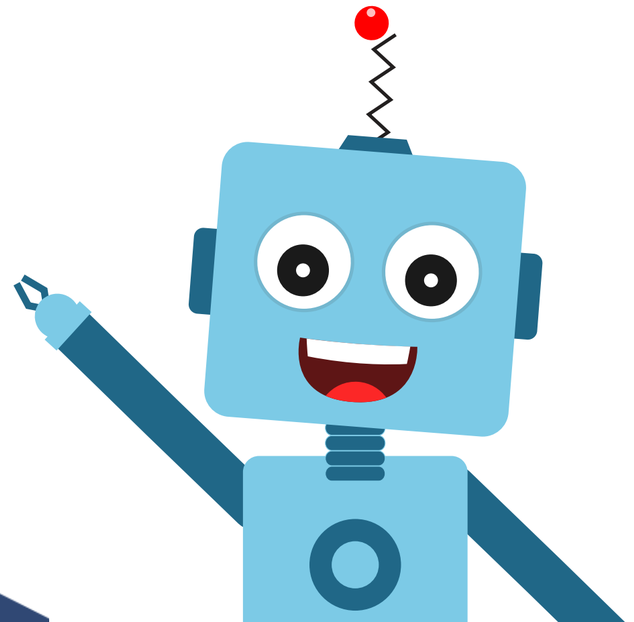
POWERED BY STEMpedia

Introduction to Actuators

Session 27



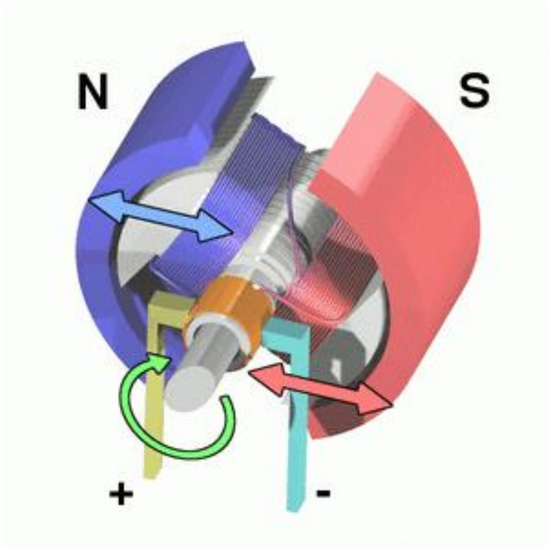
Actuators

- In simpler terms, an actuator is a device that actuates another device. To put it another way, it is a component in a machine that absorbs energy – which can come from air, electricity, or a liquid – and uses it to enable movement of something. The working of an actuator is analogous to how a human body works. They work in a machine to perform movement, similar to muscles in the body that transform energy obtained from food into motion such as arm movement.

Type of Actuators

Types

- **Electrical**: These actuators convert electricity into rotational motion, linear motion or both. These are mainly found in electric cars, robots etc.

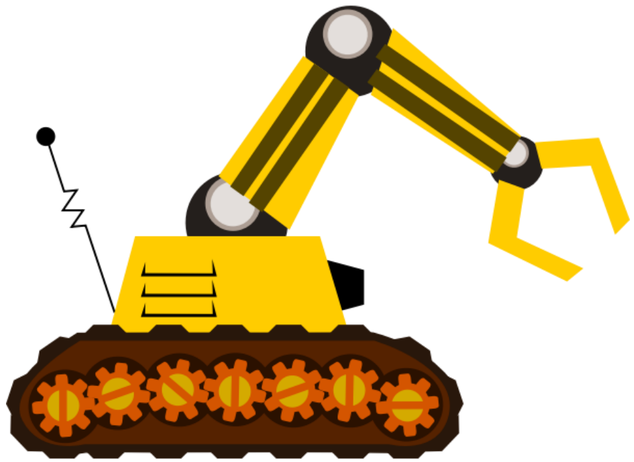


- **Mechanical**: These convert circular motion into linear (straight-line) motion. These are one of most widely used actuators, for instance: a pulley, gear etc.

Applications

POWERED BY STEMpedia

- The applications of actuators are endless. They are now a universal necessity in different industries including automation, automobiles, manufacturing, etc. It's inconceivable to fathom getting anything done in industrial automation or robotics without actuators. They are also utilized extensively in heavy construction equipment with high performance to carry out various risky and tedious tasks For example – bulldozers, cranes, transport carriers etc.
- The robots we'll be building will also have an electric actuator – a servo motor. We must learn about it first because we will be dealing with it later while making projects.

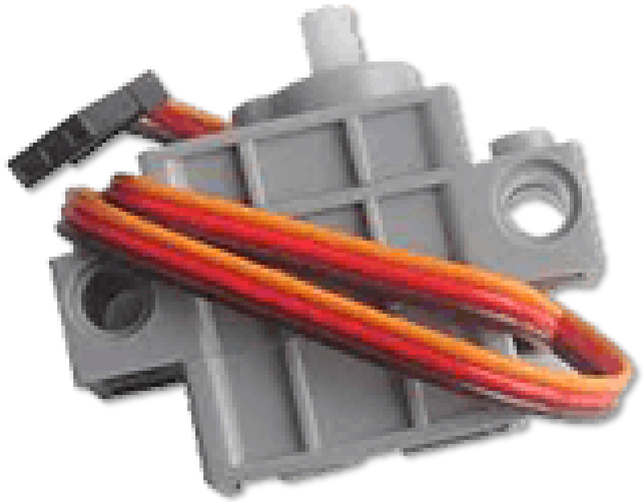


Servo Motor

What is a Servo Motor?

POWERED BY STEMpedia

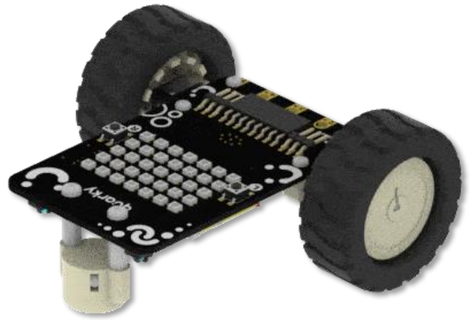
A servo motor is used to push, pull or rotate an object with great precision. Inside the motor casing, there is a motor (either DC motor or AC motor – alternating current motor), a potentiometer (a small electrical component whose resistance can be varied), and a *control circuit*.

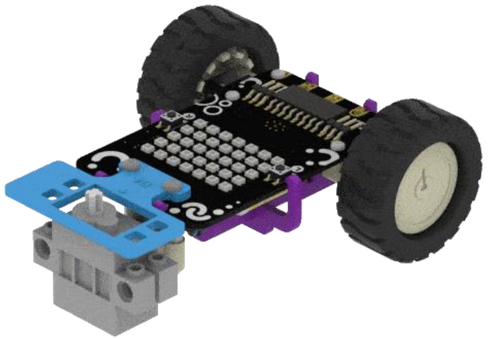


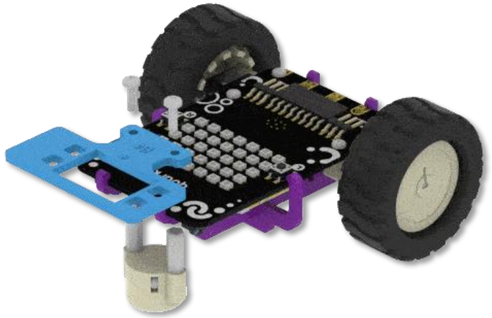
Let's Assemble!!

POWERED BY STEMpedia

- Begin by detaching the castor wheel and placing the **robot link** on the top of the Two-wheel drive robot.
- Now fasten both **robot link** and the **castor wheel** using bolts.
- Snap the **Servo motor** onto the robot link.
- The shaft points upwards and is aligned with the cross marked on the robot link.





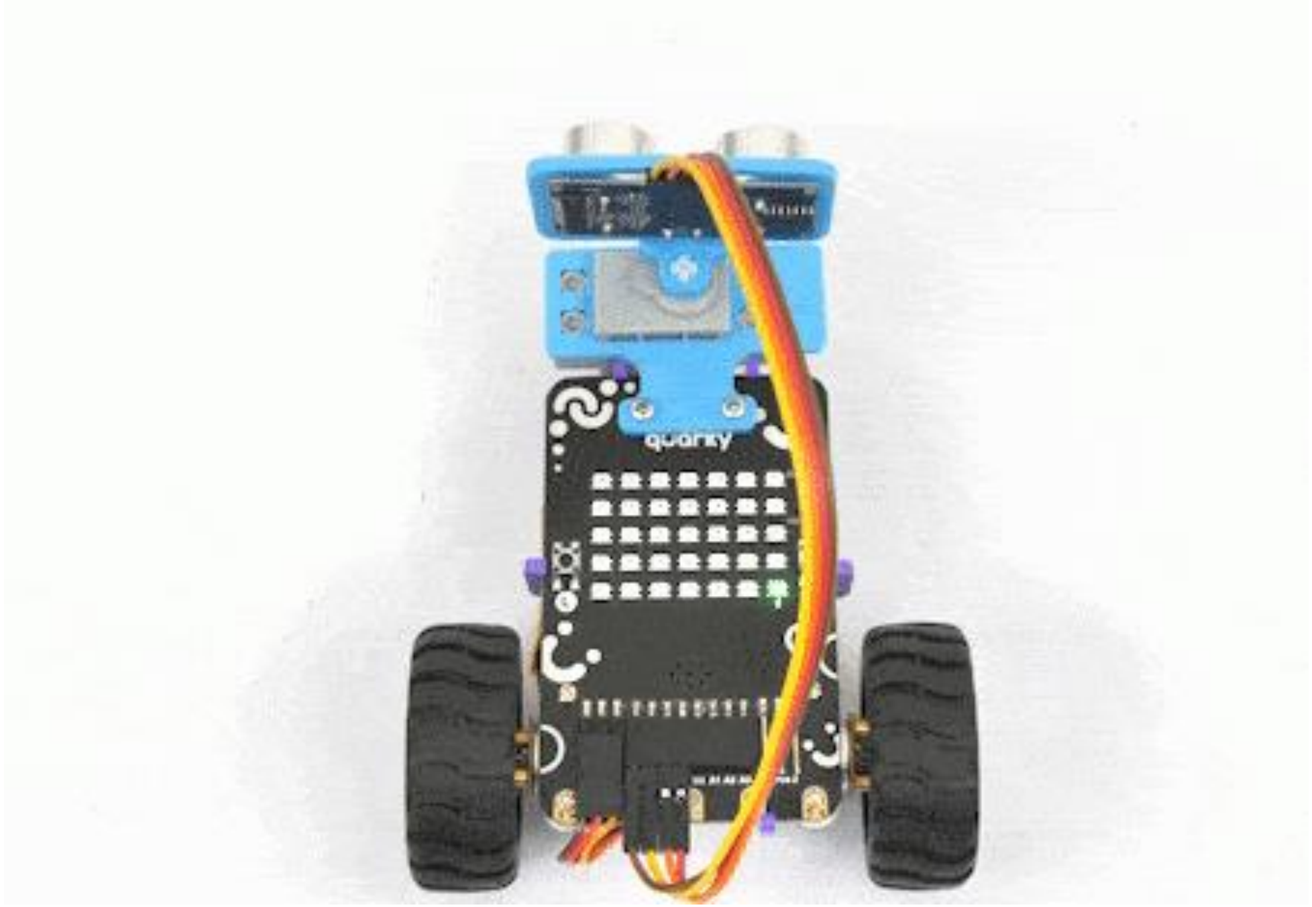


Activity: Servo Movement

Application of Robot

POWERED BY STEMpedia

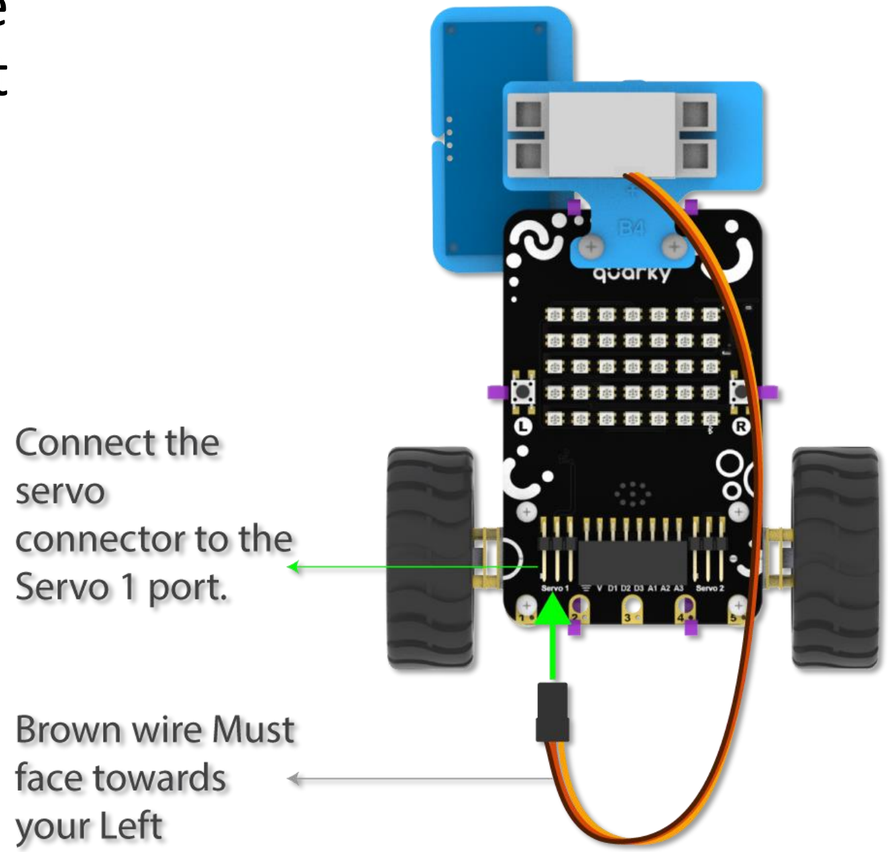
- The purpose of servo motor calibration is to rectify the angle of your servo motor.



Connecting servo to your Quarky

POWERED BY STEMpedia

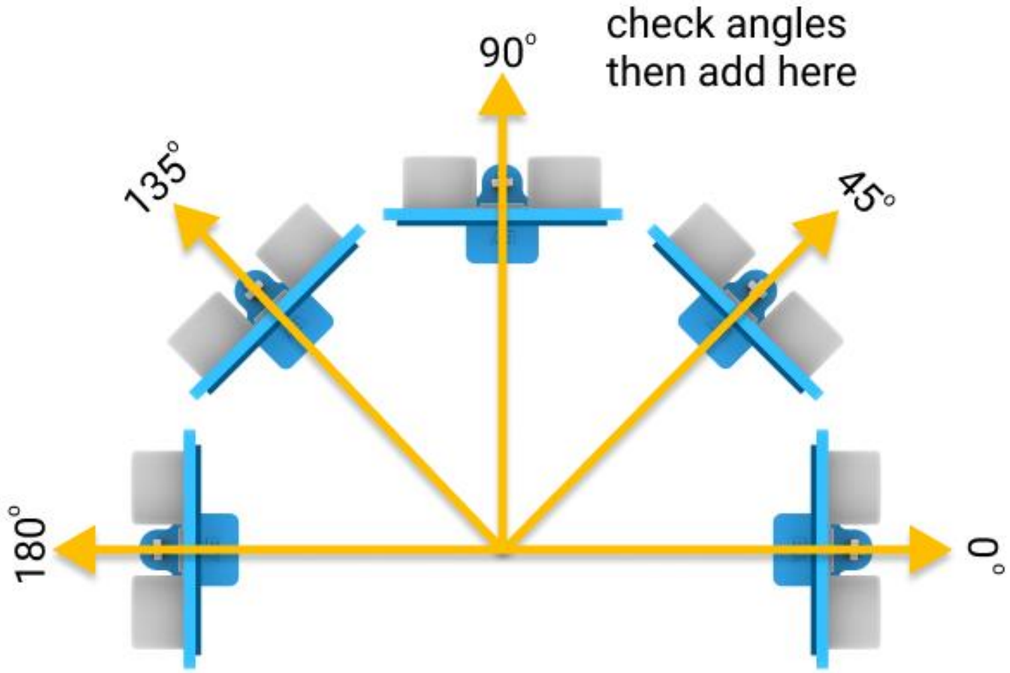
- Servo motor will be connected to Quarky servo connector. There are two servo ports on Quarky. Always make sure that brown wire is on your left side.



Angle Calibration

POWERED BY STEMpedia

- To do the calibration we have to be familiar with the servo angles,
- According to the servo motor's configuration, **90°** means facing *forward*. Therefore, **45°** is in the right, and **135°** in the left, which is nothing but **45°** in the left, *from* the forward position.



Activity:Servo movement

Activity:Servo movement

POWERED BY STEMpedia

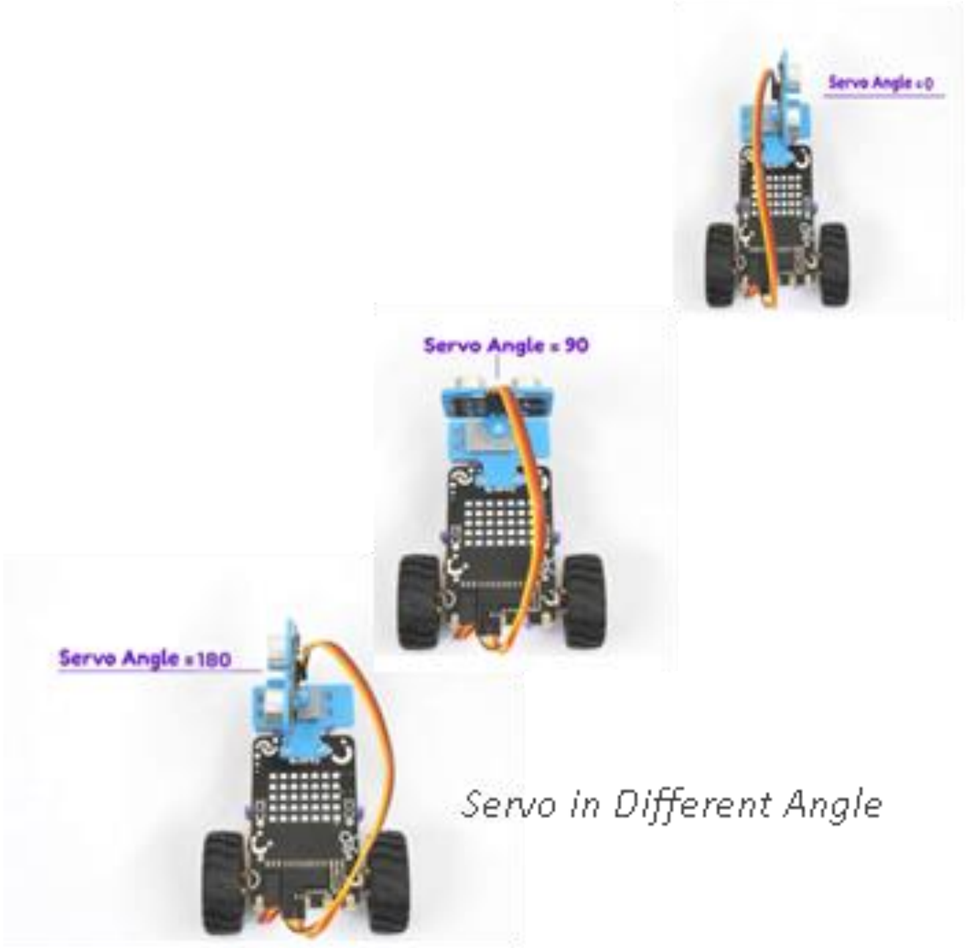
- Continuing with the next activity, we'll rotate the servo motor's shaft at different angle such as 0, 90, and 180 degrees.
- Attach the Ultrasonic Assembly to the servo shaft.
- we want to **run our code continuously**, hence we **use a while loop** such that the code keeps running continuously, until we stop it.
- Now, for adding a time delay of some second after forward motion, we will be using function.
- We set the initial parameter Angle to 0.
- Now, we will add a for loop, to apply filter on each and every face detected by the camera. For this, we use the **range (start, stop, step)** function in python, where we choose,
- We will start by setting the Angle to 0, then increasing it by 10 degrees until it reaches 180.
- Now we will code for decrease the Angle by 10 degrees.
- Now, our complete, function would look like this:
- Press Run Button to run the code.

Activity:Servo movement

POWERED BY STEMpedia

```
sprite = Sprite('Tobi')
quarky = Quarky()
import time
Angle = 0
while True:
  for i in range(0, 18):
    Angle += 10
    quarky.moveservo("Servo 1", Angle)
    time.sleep(0.01)

  for i in range(0, 18):
    Angle += -10
    quarky.moveservo("Servo 1", Angle)
    time.sleep(0.01)
```



Servo in Different Angle

POWERED BY
STEMpedia

www.ai.thestempedia.com