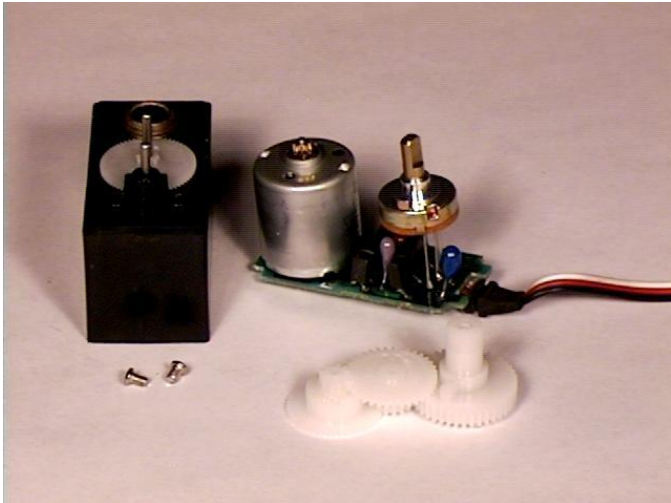
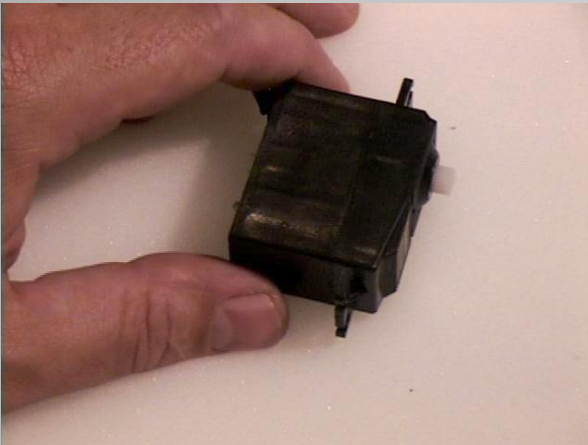


Introduction to Servos



The material presented is taken from a variety of sources including:
www.seattlerobotics.org/guide/servos.html,
www.baldor.com/pdf/manuals/1205-394.pdf
and Parallax educational materials

Overview

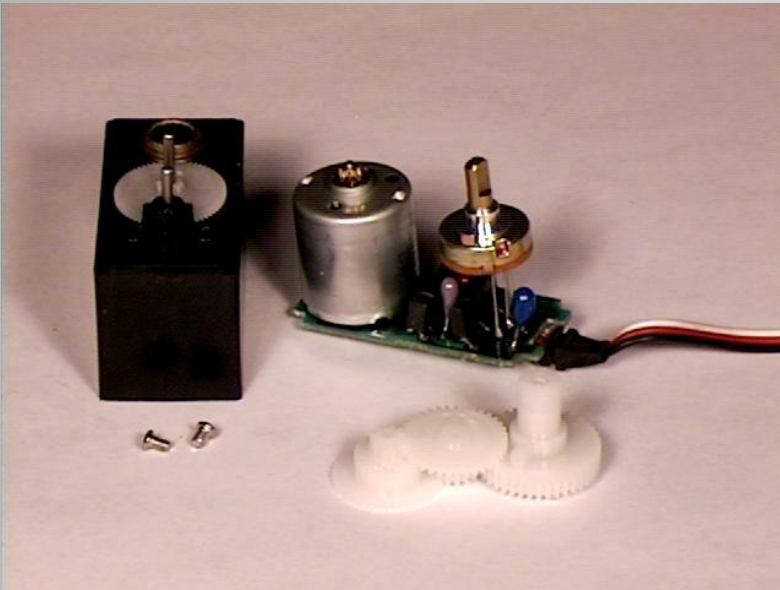


- Servo motors are used for angular positioning, such as in radio control airplanes.
- The output shaft of a servo does not rotate freely, but rather is made to seek a particular angular position under electronic control.

What makes a Servo

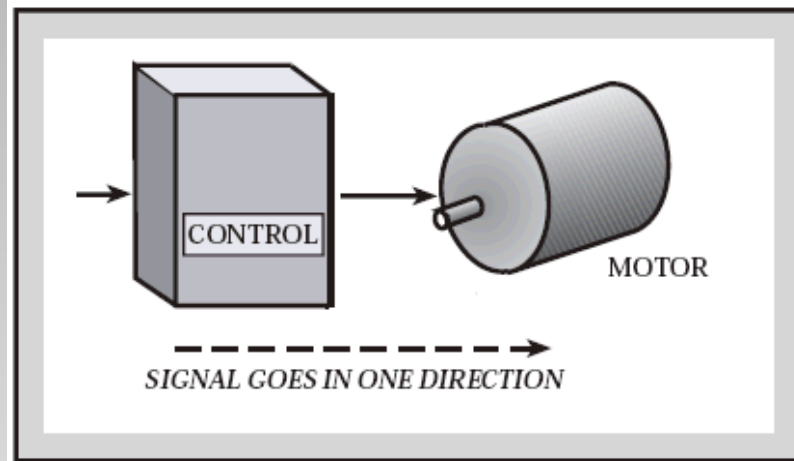
Servo motors are constructed out of basic DC motors, by adding:

- some gear reduction
- a position sensor for the motor shaft
- an electronic circuit that controls the motor's operation

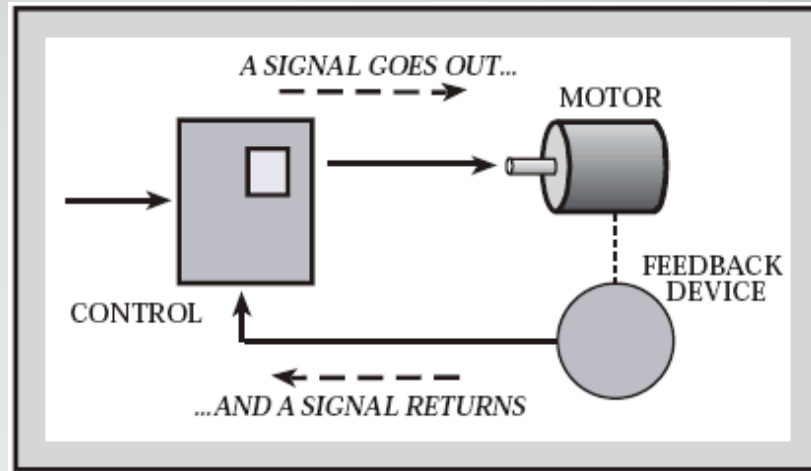


- The basic hobby servo has a 180:1 gear ratio. The motor is typically small.
- Typically, a potentiometer (variable resistor) measures the position of the output shaft at all times so the controller can accurately place and maintain its setting.

open-loop



closed-loop



Feed-back loop

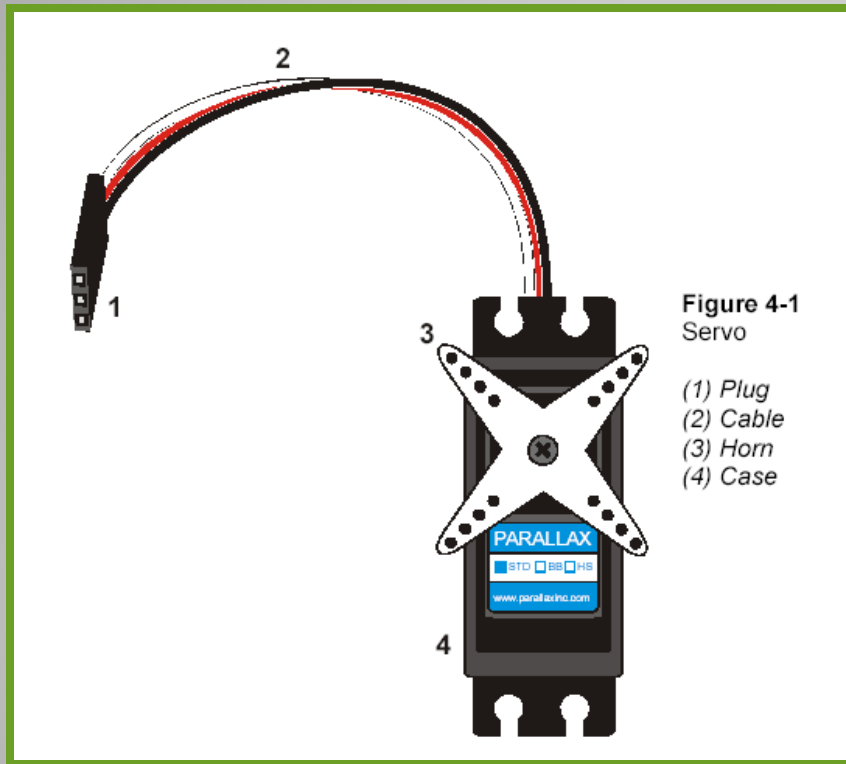
Control

- The servo responds to a 1 to 2ms pulse/signal 20ms apart.
- An external controller (such as the Basic Stamp) tells the servo which direction and how fast to move.
- This is referred to as Pulse Width Modulation (PWM)

Pulse Width and Position

- A control wire communicates the desired angular movement. The angle is determined by the duration of the pulse applied to the control wire.
- The servo expects to see a pulse every 20 milliseconds (.02 seconds). The length of the pulse will determine how far the motor turns
- If the pulse is **shorter than 1.5 ms**, then the motor will turn clockwise.
- If the pulse is **longer than 1.5ms**, the shaft turns **counter clockwise**.

Parallax Servo Connections

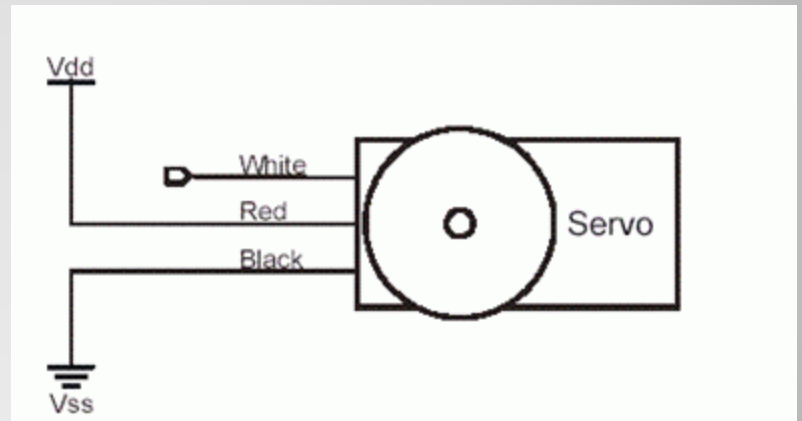


Servo Connector:

Black – Vss

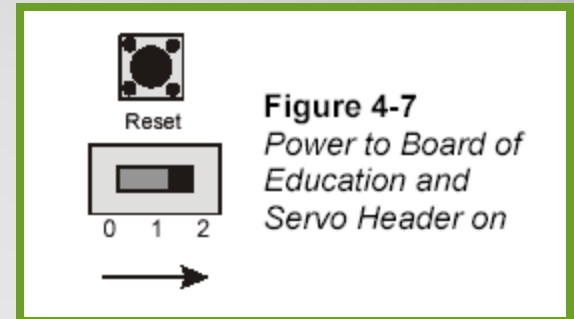
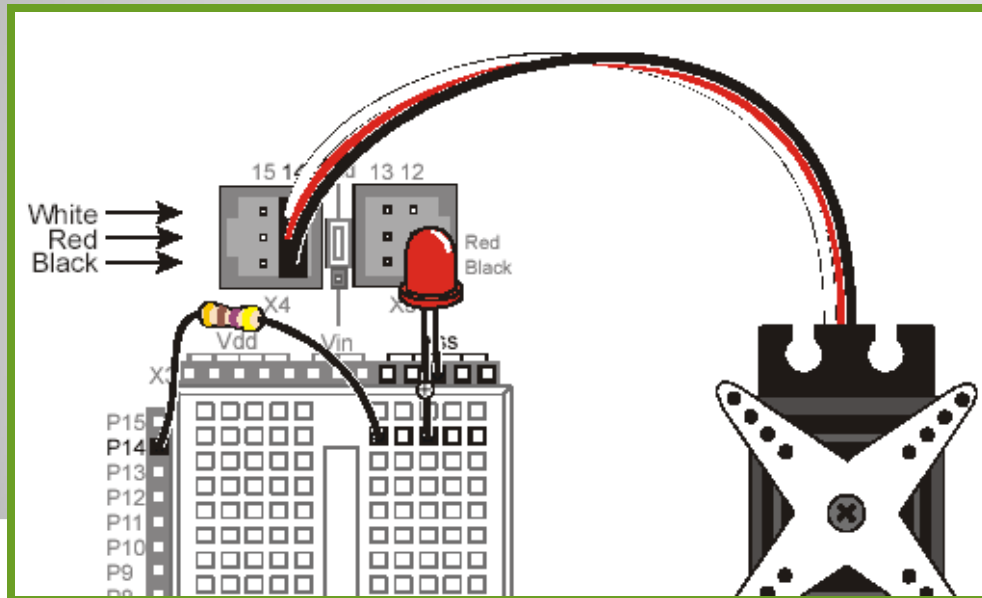
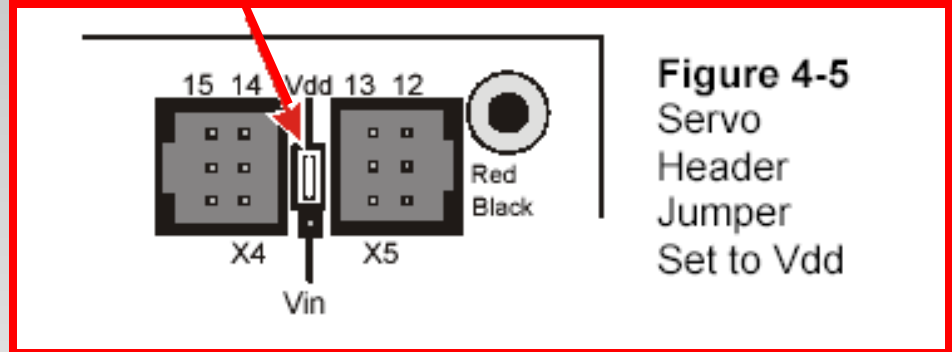
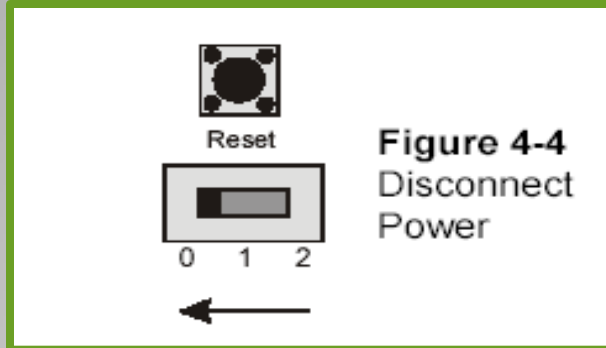
Red – Vdd or Vin

White – Signal



Servo on BOE Rev. C

Adjust jumper to connect to Vin



Programming Servos

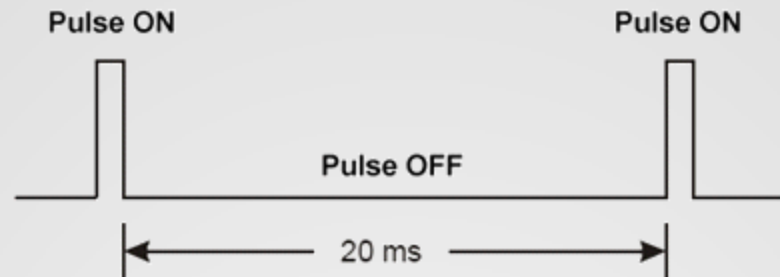
Recall that the servo is controlled by bursts of signals **spaced 20mS** apart. A high signal can last between **1mS to 2mS**.

The PULSOUT instruction is used to send the signals:
PULSOUT *pin, duration*

pin: Defines which I/O pin to use.

duration: defines how long the pulse should last, but it is NOT in mS.

The servo-motors are activated by sending them "pulses" of electricity at varying frequencies:



PBASIC PULSOUT command

The PULSOUT duration is in 2 microsecond (μS) increments.

$1 \mu\text{S} = .000001$ seconds.

$1 \text{ mS} = 1000\mu\text{S}$

To send a 1.5ms pulse which will center/stop the servo:

- Convert 1.5ms to 1500 μS
- Divide 1500 by 2 = 750
- Then use **PULSOUT 13, 750** in your code.
- This will send a pulse that lasts $750 \times 2 \mu\text{S} = 1500 \mu\text{S}$ or 1.5 mS on pin 13.

Moving Your Boe-Bot Forward

The left servo is connected to port 13

The right servo is connected to port 12

To move Forward:

Pulsout 13, 850	Lft must turn CCW (750 -850)
Pulsout 12, 650	Rt must turn CW (650 - 750)

850 = 1.7ms pulse = full speed CCW (1700 μ s /2)

750 = 1.5ms pulse = center & stop(1700 μ s /2)

650 = 1.3ms pulse = full speed CW (1300 μ s /2)

Notice that a tiny change in pulse width has huge effects on the servo.

You can slow the servo down by using 780 for left and 720 for right. I added and subtracted 30 from 750. To move forward, both values must be the same absolute distance from 1.5 (750).

Moving Your Boe-Bot Backwrds

The left servo is connected to port 13

The right servo is connected to port 12

To move Backwards:

Pulsout 13, 650

Lft must turn CW (650 - 750)

Pulsout 12, 850

Rt must turn CCW (750 - 850)

850 = 1.7ms pulse = full speed CCW (1700 μ s /2)

750 = 1.5ms pulse = center & stop(1700 μ s /2)

650 = 1.3ms pulse = full speed CW (1300 μ s /2)

Turning Your Boe-Bot

The left servo is connected to port 13
The right servo is connected to port 12

For Left turns:

Pulsout 13, 650

Lft must turn CW (650 - 750)

Pulsout 12, 650

Rt must turn CW

For Right turns:

Pulsout 13, 850

Lft must turn CCW (750 - 850)

Pulsout 12, 850

Rt must turn CCW

Summary For Motion Control

Summary of Servo PulsOut Commands

Left (P13)	Right (P12)	Resulting Motion
850	650	Forward Full Speed
850	750	Forward Right Turn
750	650	Forward Left Turn
650	850	Backward Full Speed
650	750	Backward Right Turn
750	850	Backward Left Turn
650	650	Counter-clockwise Pivot
750	750	No Motion
850	850	Clockwise Pivot

Example For One Servo

```
counter VAR Word
DEBUG "Counterclockwise 10 o'clock", CR
FOR counter = 1 TO 150
  PULSOUT 14, 1000
  PAUSE 20
NEXT

DEBUG "Clockwise 2 o'clock", CR
FOR counter = 1 TO 150
  PULSOUT 14, 500
  PAUSE 20
NEXT

DEBUG "Center 12 o'clock", CR
FOR counter = 1 TO 150
  PULSOUT 14, 750
  PAUSE 20
NEXT

DEBUG "All done."
END
```