

Parallax Standard Servo (#900-00005)

The Parallax Standard Servo provides 180° range of motion and position control to your project. Great for animatronics and robotics applications.

Features

- Holds any position between 0 and 180 degrees
- 38 oz-in torque at 6 VDC
- Accepts four mounting screws
- Easy to interface with any Parallax microcontroller or PWM-capable device
- Simple to control with the PULSOUT command in PBASIC
- High-precision gear made of POM (polyacetal) resin makes for smooth operation with no backlash
- Weighs only 1.55 oz (44 g)



Key Specifications

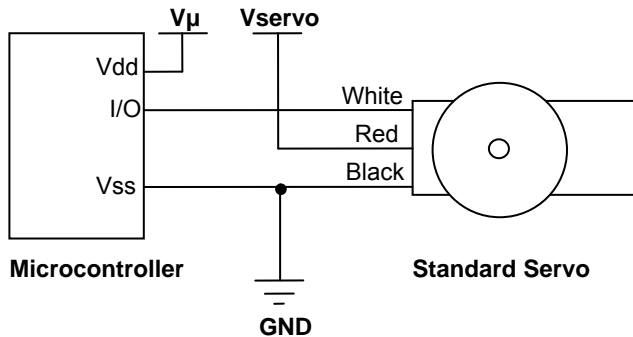
- Power requirements: 4 to 6 VDC*; Maximum current draw is 140 +/- 50 mA at 6 VDC when operating in no load conditions, 15 mA when in static state
- Communication: Pulse-width modulation, 0.75–2.25 ms high pulse, 20 ms intervals
- Dimensions approx 2.2 x 0.8 x 1.6 in (5.58 x 1.9 x 40.6 cm) excluding servo horn
- Operating temperature range: 14 to 122 °F (-10 to +50 °C)

*Power Requirement Notes

The manufacturer specifies 4-6 VDC for this servo. However, we find that this servo is tolerant of a 9 V battery for very brief periods of time when there is no load, as used in some activities in the Stamps in Class series of tutorials. (Slight jittering may be observed when batteries are fresh; this does not cause damage). Do not use this servo with an unregulated wall-mount supply, or a regulated wall mount supply exceeding 6 VDC.

Servo current draw can spike while under load. Be sure that your application's power supply and voltage regulator is prepared to supply adequate current for all servos used. Do not try to power this servo directly from a BASIC Stamp module's or any microcontroller's Vdd or Vin pins; do not connect the servo's Vss line directly to the BASIC Stamp module's or any microcontroller's Vss pin.

Quick-Start Circuit



V_{μ} = microcontroller voltage supply

V_{servo} = 4 to 6 VDC, regulated or battery

I/O = PWM TTL or CMOS output signal from microcontroller: 3.3 to 5 V, not to exceed $V_{servo} + 0.2$ V

Specifications

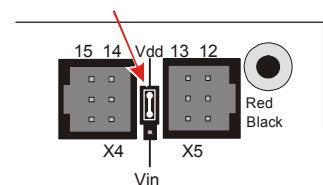
Pin	Name	Description	Minimum	Typical	Maximum	Units
1 (White)	Signal	Input; TTL or CMOS	3.3	5.0	$V_{servo} + 0.2$	V
2 (Red)	Vservo	Power Supply	4.0	5.0	6.0	V
3 (Black)	Vss	Ground		0		V

Power Precautions

- Do not use this servo with an unregulated wall-mount supply. Such power supplies may deliver variable voltage far above the stated voltage.
- Do not power this servo through the BASIC Stamp Module's Vdd pin.
- Servo current draw can spike while under peak load; be sure your application's regulator is prepared to supply adequate current for all servos used in combination.
- Some Stamps in Class tutorials, such as *What's a Microcontroller?* instruct the user to briefly power these servos with a 9 V battery when using a HomeWork Board and no load; this does not cause damage.

Board of Education Jumper Connection

When connecting the servo to the Board of Education Rev C's servo header, be sure the jumper is set to Vdd (regulated 5 VDC for this board) as shown in the figure below. Failure to place the jumper at this setting can cause damage your servo.



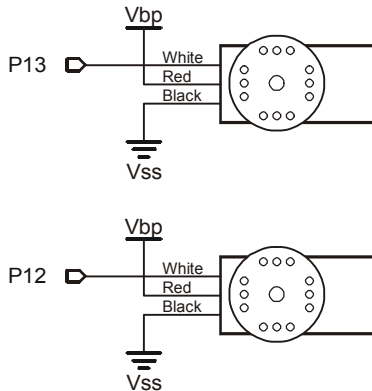
Using a Separate Power Supply on a HomeWork Board

The BASIC Stamp HomeWork Board uses a 9 V battery for a power supply. A servo can drain a fresh 9 V battery in under 20 minutes! Follow these directions to build two servo ports on the breadboard, and power them with a separate battery pack.

Hardware Required

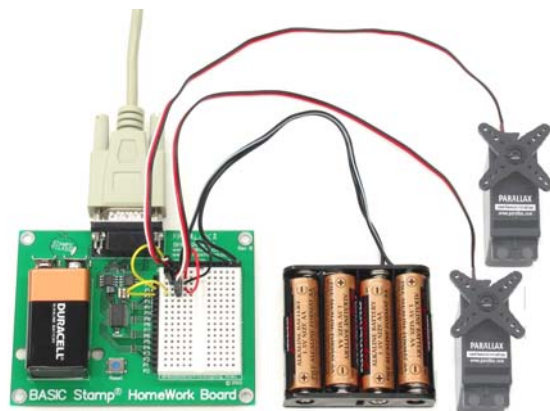
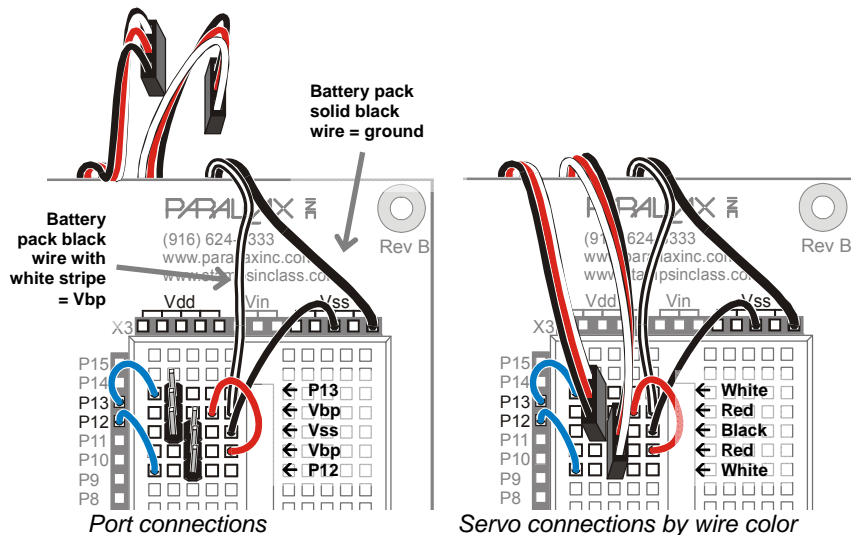
- (1) BASIC Stamp HomeWork Board (serial #555-28158 or USB #555-28188)
- (1) Battery pack with tinned leads (Parallax #753-00001)
- (2) Parallax standard servos (#900-00005)
- (2) 3-pin male-male headers (Parallax #451-00303)
- (4) Jumper wires (10-pack: Parallax #800-00016)
- (4) 1.5 V AA batteries

- ✓ Disconnect the 9 V battery from the board, and do not put the AA batteries in their holder yet.
- ✓ Build the servo ports shown by the schematic and wiring diagram below.
- ✓ Double-check to make sure the black wire with the white stripe is connected to Vbp, the solid black wire is connected to Vss, and that all the connections for P13, Vbp, Vss, Vbp (another one), and P12 all exactly match the wiring diagram.
- ✓ Connect the servo plugs to the male headers on the right side of the wiring diagram.
- ✓ Connect the 9 V battery, and insert the AA batteries into their holder.



Vbp stands for Voltage battery pack.

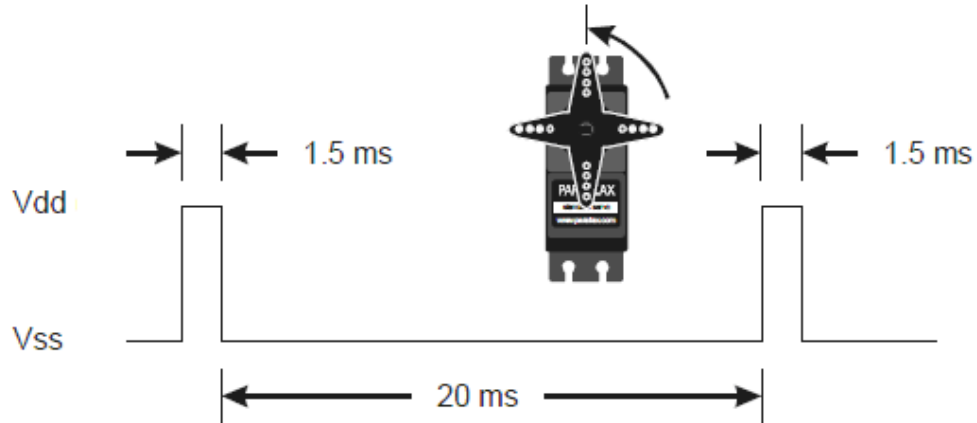
It refers to the 6 VDC supplied by the four 1.5 V batteries. This is brought directly to the breadboard to power the servos. BASIC Stamp is still powered by the 9 V battery.



HomeWork Board with servo ports built on the breadboard, with a separate battery pack power supply for the servos.

Communication Protocol

The Parallax Standard Servo is controlled through pulse width modulation, where the position of the servo shaft is dependent on the duration of the pulse. In order to hold its position, the servo needs to receive a pulse every 20 ms. Below is a sample timing diagram for the center position of the Parallax Standard Servo.



BASIC Stamp 2 Programming Example

PBASIC has a PULSOUT command that sets the I/O pin to an output and sends a pulse of a specified duration.

PULSOUT *Pin, Duration*

The example shown below for a BASIC Stamp 2 causes a servo connected to BASIC Stamp I/O pin 12 to turn to and hold its center position for approximately 5 seconds.

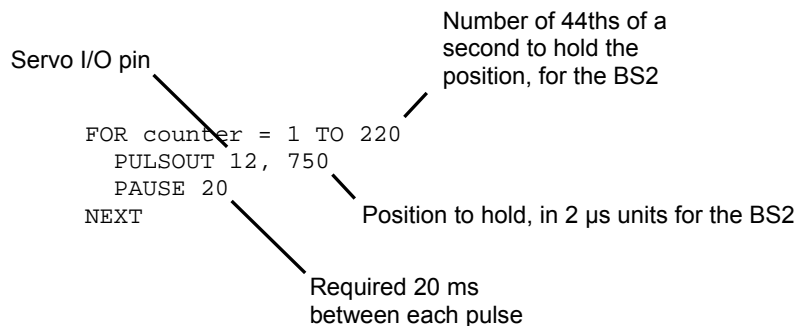
```
' CenterStdServo.bs2
' {$STAMP BS2}
' {$PBASIC 2.5}

counter VAR Word

FOR counter = 1 TO 220

    PULSOUT 12, 750
    PAUSE 20

NEXT
```



For detailed explanations using the BASIC Stamp 2, see *What's a Microcontroller?* Chapter 4, available for free download from www.parallax.com/go/WAM.

Different BASIC Stamp modules use different units for the PULSOUT command's *Duration* argument. When adapting BS2 code to another BASIC Stamp model, you may need to make adjustments. See the article "Adapt BS2 Code to Other Models" in the PBASIC Language section of the BASIC Stamp Editor Help.

The table below lists the PULSOUT ranges for each BASIC Stamp model.

BASIC Stamp Module	0.75 ms	1.5 ms (center)	2.25 ms
BS1	75	150	225
BS2, BS2e, BS2pe	375	750	1125
BS2sx, BS2px, BS2p24/40	938	1875	2813

Propeller P8X32A Programming Example

Servo control with the Propeller chip is simplified by using a cog's counter modules. The code below causes a servo connected to I/O pin P0 to turn to and hold the 90° position. For more information about counter modules and PWM with the Propeller, see Chapter 7 in the Propeller Education Kit Labs: Fundamentals text, which is included as a PDF in the Propeller Tool IDE Help.

```
{ { CenterParallaxServo.spin
For centering Parallax Continuous Rotation Servo
or holding Parallax Standard Servo at 90° position.
Sends a 1.5 ms pulse approx every 20 ms }}

CON
_clkmode = xtall + pll16x           ' System clock → 80 MHz
_xinfreq = 5_000_000              ' Using 5 MHz external crystal oscillator
servoPin = 0                       ' Servo signal to this I/O pin-change if needed

PUB CenterServo | tInc, tc, tHa, t

ctra[30..26] := %00100             ' Configure Counter A to NCO
ctra[8..0]   := servoPin

frqa := 1
dira[servoPin]~~

' Set up cycle and high times
tInc := clkfreq/1_000_000
tc    := tInc * 21_500
tHa   := tInc * 1500
t     := cnt                       ' Mark counter time

repeat                             ' Repeat PWM signal
  phsa := -tHa                     ' Set up the pulse
  t += tc                           ' Calculate next cycle repeat
  waitcnt(t)                        ' Wait for next cycle
```

Revision History

Version 2.1: corrected specifications for torque. Updated PULSOUT range table. Added Using a Separate Power Supply on a HomeWork Board section beginning on page 2.