17 Servo

A servo motor — usually called a *servo* for short — combines a DC motor, gearbox, potentiometer, and controller electronics to provide relatively precise angular position control of a rotating shaft. Servos provide motion control for robot arms, rotating sensor scanners, and other actuators. Servos have been used for many years in remote-controlled airplanes, cars, and boats to manipulate control surfaces (aileron, elevator, rudder) and steering. Figure 17.1 pictures the GWS S03N STD servo included in the NI myRIO Mechatronics Kit; the kit also includes the GWS S35+ XF continuous-rotation sensor that can serve as the basis of a robot drive train.

Learning Objectives: After completing the activities in this chapter you will be able to:

1. Discuss the main components of a servo feedback control system: pulse-style command input, controller, DC motor, gearbox, and potentiometer as an angular position sensor,
2. Create a pulse-width modulated (PWM) signal to command the servo spline to a desired angle,
3. Null any nonideal offsets in the angular position, and
4. Explain the fundamental difference between a standard servo and a servo intended for continuous operation.

Figure 17.1: NI myRIO Mechatronics Kit servo.

17.1 Component Demonstration

Follow these steps to demonstrate correct operation of the servo.

Select these parts from the NI myRIO Mechatronics Kit:

- Servo, GWS S03N STD, `http://gwsus.com/english/product/servo/standard.htm`
- Jumper wires, M-F (3×)

Build the interface circuit: Refer to the schematic diagram shown in Figure 17.2 on the next page; the servo requires three connections to NI myRIO MXP Connector B (see Figure A.1 on page 233):

1. Vcc (red) → B/+5V (pin 1)
2. Ground (black) → B/GND (pin 6)
3. Command signal (white) → B/PWM0 (pin 27)

Make certain that you are using the correct servo (GWS S03N STD); the NI myRIO Mechatronics Kit includes a continuous-rotation servo with similar appearance.

Run the demonstration VI:

- Download `http://www.ni.com/academic/myrio/project-guide-vis.zip` if you have not done so previously and unpack the contents to a convenient location,
- Open the project `Servo demo.lvproj` contained in the subfolder `Servo demo`,
- Expand the hierarchy button (a plus sign) for the myRIO item and then open `Main.vi` by double-clicking,
- Confirm that NI myRIO is connected to your computer, and
- Run the VI either by clicking the Run button on the toolbar or by pressing Ctrl+R.

Expect to see a "Deployment Process" window showing how the project compiles and deploys (downloads) to NI myRIO before the VI starts running.

NOTE: You may wish to select the "Close on successful completion" option to make the VI start automatically.

Expected results: The demo VI includes a pointer slide control to adjust the servo angle. Move the slider and confirm that the servo shaft turns in response. You may also use the Page-Up and Page-Down keys to move the slider. Which slider values (positive or negative) correspond to clockwise motion?

The servo includes a parts kit that includes a variety of "servo arms," also known as "servo horns" (see Figure 17.3 on page 72). Attach the two-arm servo horn to the servo "spline" (the gear-shaped rotating shaft of the servo) so that you can more easily see the servo rotation angles.

The slider is calibrated in "percent full scale" (%FS). Estimate the servo angle at 100%FS and then at −100%FS. Use the direct entry box at the top of the slider to apply step changes, e.g., +100%FS to −100%FS; how quickly does the servo rotate between these two angles?

The default limits on the slider allow a 2× "over travel" factor. At what %FS values does the servo reach its rotation limits?

The servo command input is a variable-width pulse with 1.0 ms at −100%FS and doubling to 2.0ms at +100%FS; the midpoint pulse width 1.5 ms — called the neutral-position pulse width — corresponds to 0%FS. The pulse must be repeated at a sufficiently fast rate and yet not too fast. Try adjusting the freq [Hz] control to a lower frequency (say, 10 Hz) and to a higher frequency (say, 200 Hz), each time moving the position slider to command different angles. Note the two indicators under this control that indicate the pulse width and "duty cycle" (percent of time that the pulse is active) of the signal sent to the servo command input. What happens to the servo angle at these frequency extremes? Experiment to determine a range of frequencies that yields satisfactory control of the servo angle.

Disconnect the servo and replace it with the continuous-rotation servo (GWS GWS S35+ XF); this servo uses a slightly different connector: red = +5 V, brown = ground, and yellow = command input. Investigate the behavior of this servo and compare it to your earlier results.

Click the Stop button or press the escape key to stop the VI and to reset NI myRIO.

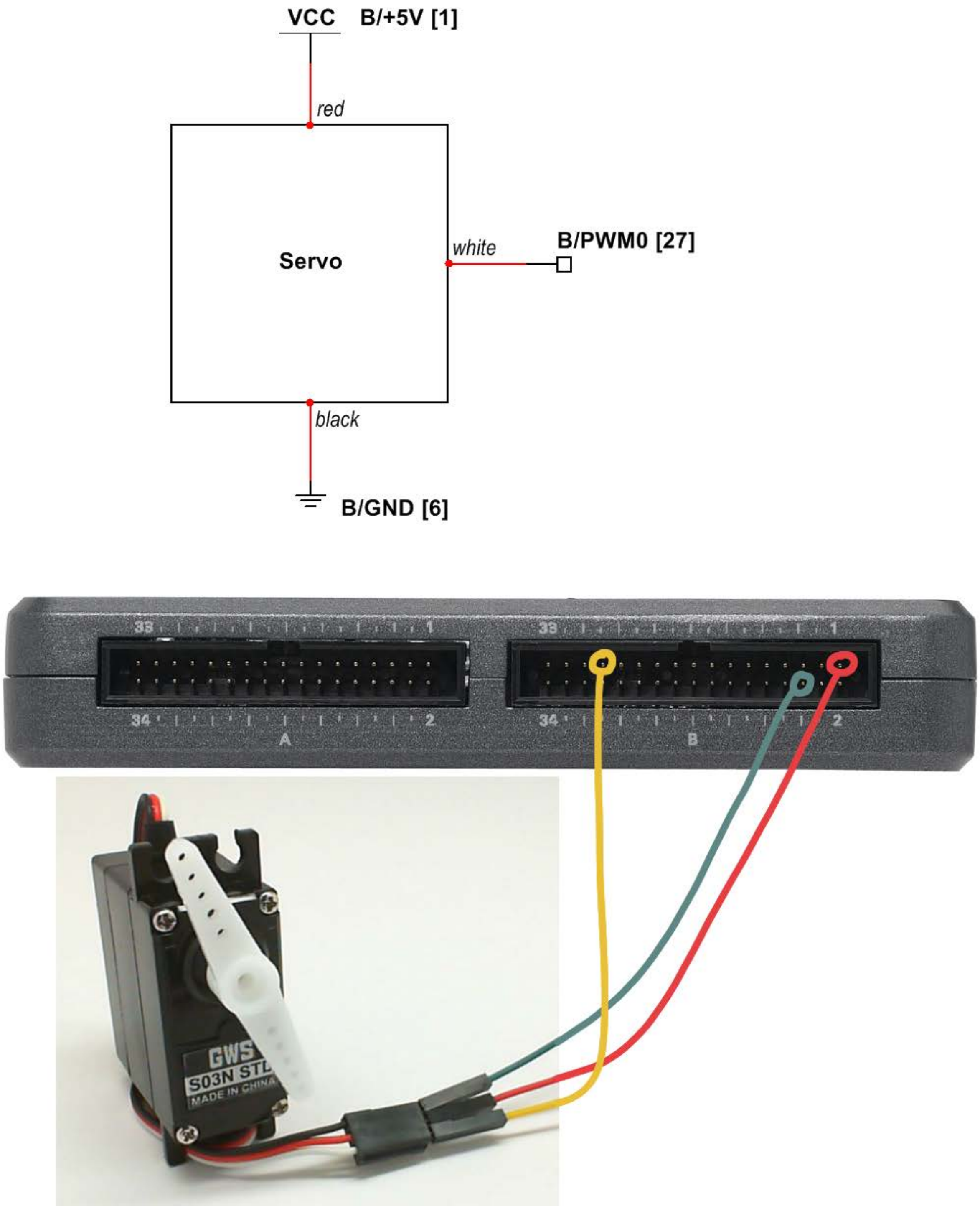

Figure 17.2: Demonstration setup for the GWS S03N STD servo connected to NI myRIO MXP Connector B.

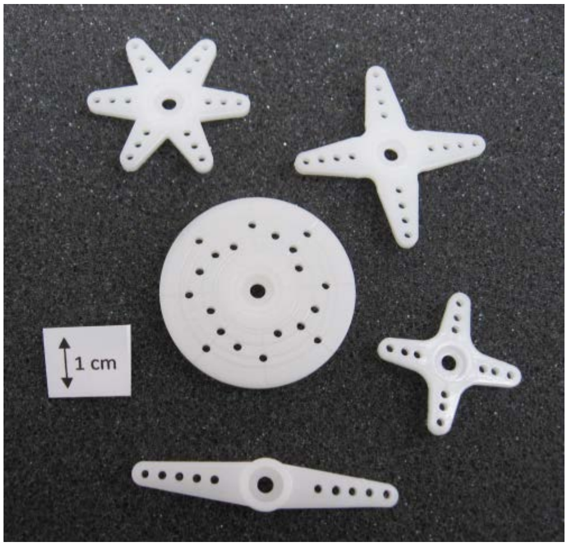

Figure 17.3: Servo control horns included with the GWS S03N STD.

Troubleshooting tips: Not seeing the expected results? Confirm the following points:

- Glowing power indicator LED on NI myRIO,
- Black Run button on the toolbar signifying that the VI is in run mode,
- Correct MXP connector terminals — ensure that you are using Connector B and that you have the correct pin connections, and
- Correct servo connections — double-check your connections, and ensure that you have connected the red line to the +5-volt supply, the black line to ground, and the white line to the PWM0 output.

17.2 Interface Theory

Interface circuit: The servo adjusts its shaft angle according to the *command input*, a periodic pulse that varies in width between 1.0 and 2.0 ms. A pulse width centered between these two limits (1.5 ms) commands the servo to its neutral (center) position. The servo requires the 5-volt power supply and a single connection to one of the pulse-width modulation (PWM) outputs available on the NI myRIO.

Study the video *Servo Interfacing Theory* (`youtu.be/DOu5AvSDP2E`, 7:18) to learn about servo applications, the internal components of the servo (controller, motor drive, gearbox, and potentiometer), the principle of operation of the servo's feedback control system, and the technique used to make a continuous-rotation servo.

LabVIEW programming: Study the video *PWM Express VI* (`youtu.be/mVN9jfwXleI`, 2:41) to learn how to use the PWM VIs to open a channel to a PWM output and to set the pulse width and pulse repetition rate.

17.3 Basic Modifications

Study the video *Servo Demo Walk-Through* (`youtu.be/QXHe0DFbUdc`, 4:23) to learn the design principles of `Servo demo`, and then try making these modifications to the block diagram of `Main.vi`:

1. Add the necessary computation to calibrate the pointer slide control in degrees of rotation. Use the same technique as the null offset code, but now with a multiplicative scale factor (be sure to initialize the feedback node to 1). Test your code as follows:
   - Null the offset,
   - Set the servo angle to zero and note the position of the servo horn arm,
   - Adjust the servo angle until the arm rotates +90 degrees,
   - Click and then release your "scale" button, and then
   - Enter 90 degrees into the point slide control direct entry box and confirm that the arm rotation is exactly 90 degrees from center.

2. Make the servo follow an *angular position trajectory*, i.e., a sequence of angles stored in an array:
   - Replace the while-loop with a for-loop (right-click on the for-loop and select the Conditional Terminal option),
   - Create an array using the Sine Pattern generator found in the Signal Processing | Sig Generation subpalette; select an amplitude for the sine pattern that will move the servo horn arm throughout its possible range of travel,
   - Change the pointer slide control to an indicator, and
   - Connect the Sine Pattern output through the for-loop frame to the position control wire.

17.4 Integrated Project Ideas

Now that you know how to use the servo consider integrating it with other devices to create a complete system, for example:

- *Steer By Wire* (44)
- *Hotel Room Safe Controller* (49)
- *Scanning Sensor* (51)
- *NTP Clock* (43)

17.5 For More Information

- *Actuators and Servos* by Society of Robots ∼ Many practical details on servos: `http://www.societyofrobots.com/actuators_servos.shtml`
- *Servo Control* by PC Control Learning Zone ∼ Another good tutorial on servos: `http://www.pc-control.co.uk/servo_control.htm`