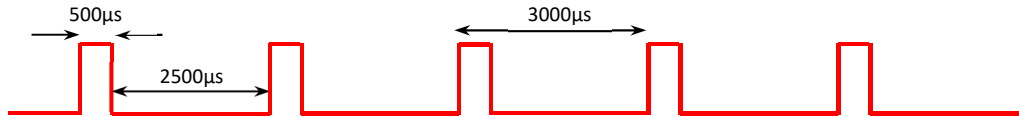


# Chapter 18 Written Problems

Use this sheet to record your answers to the following Chapter 18 Challenge Problems.

18-1. Explain the primary difference between modified and unmodified servos.

18-2. Examine the pulse train shown in Figure 18.29. Give the pulse width of the pulses and the period of the square wave.



**Figure 18.29.** A train of pulses. What is the pulse width and the period of the wave? What is the time between a rising edge and a falling edge? What is the time between subsequent leading edges?

Pulse width = \_\_\_\_\_

Period,  $T$  = \_\_\_\_\_

18-3. In Figure 18.29 above, what is the time between a falling edge and its subsequent rising edge? What is the time between consecutive leading edges?

18-4. Programmers can control the speed and direction of a servomotor by varying the width of the pulses of a square wave sent to the servo. What is this technique commonly called?

18-5. Fill-in the blanks: Servomotors generally respond to pulse widths within the range of \_\_\_\_\_µs and \_\_\_\_\_µs.

Between each pulse, a delay of \_\_\_\_\_ms is required. That is, how much time should be between the trailing edge of a pulse and the leading edge of the subsequent pulse?

18-6. The code below is meant to control a servomotor connected to pin 2 on the Teensy 3.2 microcontroller but it contains **one mistake**. Do **not** run this code with your microcontroller – instead, **identify** and **correct** the mistake. Assume that the servo pin has been properly placed in **OUTPUT** mode.

```
void loop() {
  digitalWrite(2, HIGH);
  delayMicroseconds(100);
  digitalWrite(2, LOW);
  delay(20);
}
```

18-7. The code below is meant to control a servomotor connected to pin 2 on the Teensy 3.2 microcontroller but it contains **one mistake**. Do **not** run this code with your microcontroller – instead, **identify** and **correct** the mistake. Assume that the servo pin has been properly placed in **OUTPUT** mode.

```
void loop() {
  digitalWrite(2, HIGH);
  delayMicroseconds(1000);
  digitalWrite(2, LOW);
  delay(2);
}
```

- 18-8. The code below is meant to control a servomotor connected to pin 2 on the Teensy 3.2 microcontroller but it contains **one mistake**. Do **not** run this code with your microcontroller – instead, **identify** and **correct** the mistake. Assume that the servo pin has been properly placed in **OUTPUT** mode.

```
void loop() {  
  digitalWrite(2, HIGH);  
  delayMicroseconds(3000);  
  digitalWrite(2, LOW);  
  delay(20);  
}
```

- 18-9. Your friend has written the following code, which was intended to spin the spline of a modified servomotor in a clockwise direction. While the code compiles just fine, but the servo, which is plugged into pin A6 on the PRT3 motherboard, does not rotate. The physical connections are fine, and all the switches and shorting blocks on the PRT3 motherboard are in the correct orientation. Explain why the spline is not spinning. Assume that the servo pin has been properly placed in **OUTPUT** mode.

```
void loop() {  
  digitalWrite(A6, HIGH);  
  delayMicroseconds(1500);  
  digitalWrite(A6, LOW);  
  delay(20);  
}
```

- 18-10. Imagine that you connect a modified servo from your kit to pin 4 on the PRT3 motherboard, and that the pin has been properly placed in **OUTPUT** mode. Fill in the blanks of the following code with numeric values that could be used to spin the servo **rapidly** in a **clockwise** direction.

```
void loop() {  
  
  digitalWrite(4, _____ );  
  
  delayMicroseconds( _____ );  
  
  digitalWrite(4, _____ );  
  
  delay(_____);  
  
}
```

- 18-11. Imagine that you connect a modified servo from your kit to pin 4 on the PRT3 motherboard, and that the pin has been properly placed in **OUTPUT** mode. Fill in the blanks of the following code with numeric values that could be used to spin the servo **slowly** in a **counterclockwise** direction.

```
void loop() {  
  
  digitalWrite(4, _____ );  
  
  delayMicroseconds( _____ );  
  
  digitalWrite(4, _____ );  
  
  delay(_____);  
  
}
```

- 18-12. Imagine that you connect a modified servo from your kit to pin 4 on the PRT3 motherboard, and that the pin has been properly placed in **OUTPUT** mode. Fill in the blanks of the following code with numeric values that could be used to hold the servo **motionless**.

```
void loop() {  
  
    digitalWrite(4, _____ );  
  
    delayMicroseconds( _____ );  
  
    digitalWrite(4, _____ );  
  
    delay(_____);  
  
}
```

- 18-13. Examine the following code, which is used to control one of your modified servos. Assume that the servo pin has been properly placed in **OUTPUT** mode. Give the direction (clockwise or counterclockwise) and the speed (fast or slow) of the servo. If the servo will not spin, state so.

```
void loop() {  
    digitalWrite(9, HIGH);  
    delayMicroseconds(2000);  
    digitalWrite(9, LOW);  
    delay(20);  
}
```

- 18-14. Examine the following code, which is used to control one of your modified servos. Assume that the servo pin has been properly placed in **OUTPUT** mode. Give the direction (clockwise or counterclockwise) and the speed (fast or slow) of the servo. If the servo will not spin, state so.

```
void loop() {  
    digitalWrite(9, HIGH);  
    delayMicroseconds(1000);  
    digitalWrite(9, LOW);  
    delay(20);  
}
```

- 18-15. Examine the following code, which is used to control one of your modified servos. Assume that the servo pin has been properly placed in **OUTPUT** mode. Give the direction (clockwise or counterclockwise) and the speed (fast or slow) of the servo. If the servo will not spin, state so.

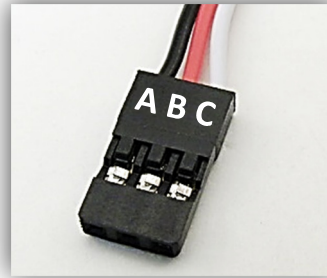
```
void loop() {  
    digitalWrite(9, HIGH);  
    delayMicroseconds(1500);  
    digitalWrite(9, LOW);  
    delay(20);  
}
```

- 18-16. Examine the following code, which is used to control one of your modified servos. Assume that the servo pin has been properly placed in **OUTPUT** mode. Give the direction (clockwise or counterclockwise) and the speed (fast or slow) of the servo. If the servo will not spin, state so.

```
void loop() {  
    digitalWrite(9, HIGH);  
    delayMicroseconds(1700);  
    digitalWrite(9, LOW);  
    delay(20);  
}
```

18-17. Below is an image of a 3-conductor cable. Identify the purpose for each of the wires **A**, **B**, and **C**. That is, identify which wire is power, signal, and ground.

- A. \_\_\_\_\_
- B. \_\_\_\_\_
- C. \_\_\_\_\_



18-18. Determine which image(s) below show the proper way to connect a servomotor to the PRT3 motherboard. (There may be more than one.) If a cable is connected improperly, explain why it is wrong.

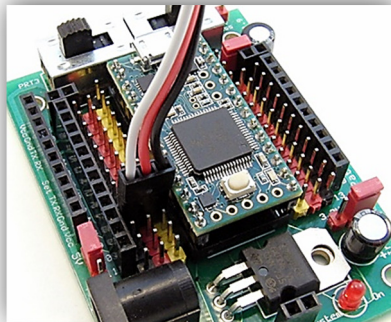


Figure A.

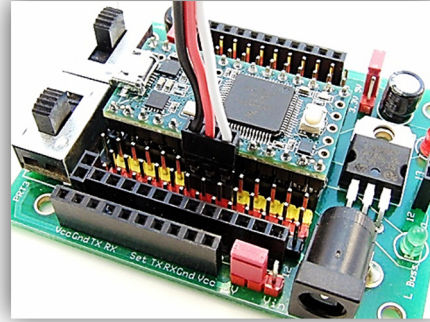


Figure B.

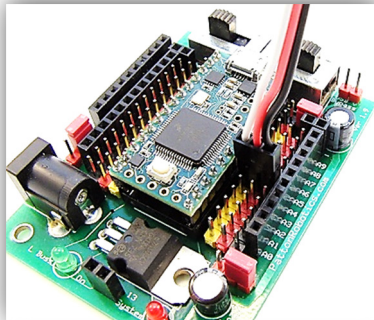


Figure C.

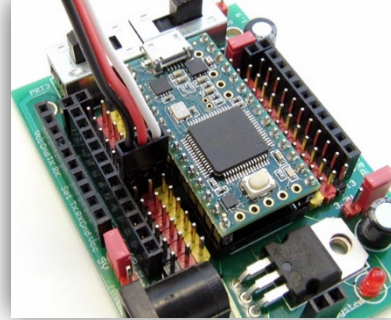


Figure D.

18-19. The initial position of the spline of an **unmodified** Hitec HS-311 servo is at the 1400- $\mu$ s position, as shown in the image below. If a string of 1500- $\mu$ s pulses is sent to the servo in the figure, which of the following best describes the action of the spline? Assume the pulses are properly created with a 20-ms delay in between each pulse, and that the servo pin has been properly placed in **OUTPUT** mode.

- A. The spline will not move.
- B. The spline will rotate clockwise in a continuous fashion.
- C. The spline will rotate counterclockwise in a continuous fashion.
- D. The spline will begin to rotate clockwise but will eventually stop when it reaches the 1500- $\mu$ s position.
- E. The spline will begin to rotate counterclockwise but will eventually stop when it reaches the 1500- $\mu$ s position.
- F. The spline will begin to rotate clockwise but will eventually stop when it reaches the 2000- $\mu$ s position.
- G. The spline will begin to rotate counterclockwise but will eventually stop when it reaches the 1000- $\mu$ s position.

