

# myDAQ USB Interface

Ver. 4 Dec 5<sup>th</sup> 2018 (120 slides)

# myDAQ USB Interface

## myDAQ Connections

### Analog Input:

2 channels, 200kS/s, 16-bit

### Analog Output:

2 channels, 200kS/s, 16-bit  
3.5mm stereo audio jacks

**Digital I/O:** 8 LVTTTL lines

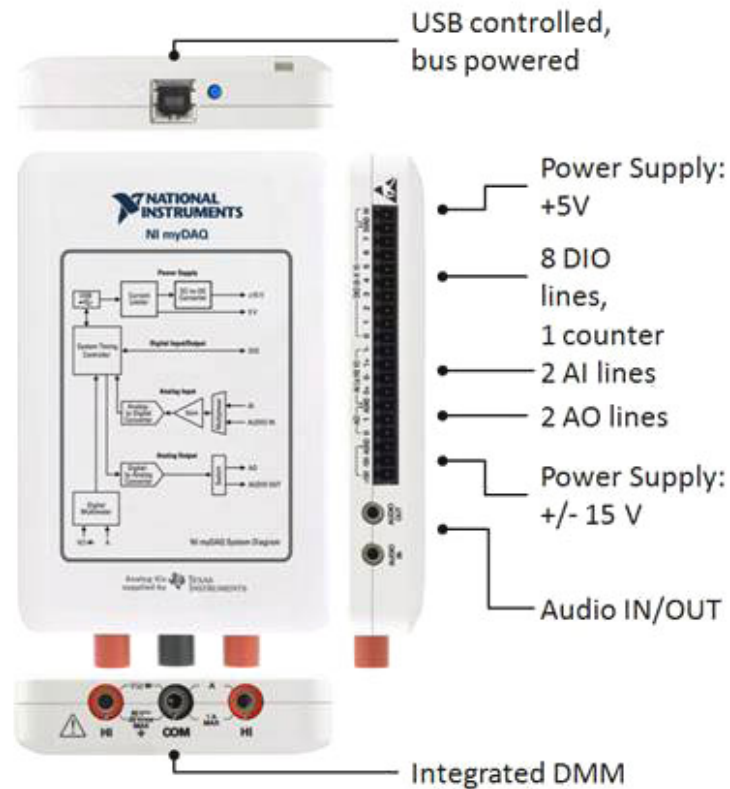
**Counter:** 1 counter/timer

**Integrated DMM:** V, A, Ohm

**Power Supply:** +5V, +/-15V

Screw term + mass term option

Bus Powered (USB) operation



How many digital I/O lines are on the myDAQ?

How many bits does the A/D use?

How many analog in and out are on the myDAQ?

What is the voltage resolution of the A/D converter?

Which pin is used to measure an input frequency signal \_\_\_\_\_ , which pin is used to produce a counter/timer PWM signal? \_\_\_\_\_

Which two myDAQ pins are used to measure a quadrature signal? \_\_\_\_\_

## MyDAQ Specs

### Digital I/O

|  |  |
|--|--|
| Number of lines .....                              | 8; DIO <0..7>  |
| Direction control .....                            | Each line individually programmable as input or output |
| Update mode .....                                  | Software-timed   |
| Pull-down resistor .....                           | 75 k $\Omega$  |
| Logic level .....                                  | 5 V compatible LVTTTL input; 3.3 V LVTTTL output       |
| V <sub>IH</sub> min .....                          | 2.0 V  |
| V <sub>IL</sub> max .....                          | 0.8 V  |
| Maximum output current per line <sup>2</sup> ..... | 4 mA   |

### Analog Input

|                             |  |
|-----------------------------|--|
| Number of channels .....    | 2 differential or 1 stereo audio input |
| ADC resolution .....        | 16 bits                                |
| Maximum sampling rate ..... | 200 kS/s                               |
| Timing accuracy .....       | 100 ppm of sample rate                 |
| Timing resolution .....     | 10 ns                                  |
| Range                       |  |
| Analog input .....          | $\pm 10$ V, $\pm 2$ V, DC-coupled      |



### General Purpose Counter/Timer

|   |                            |
|---|----------------------------|
| Number of counter/timers .....                          | 1                          |
| Resolution .....  | 32 bits                    |
| Internal base clocks .....                              | 100 MHz                    |
| Base clock accuracy .....                               | 100 ppm                    |
| Maximum counting and pulse generation update rate ..... | 1 MS/s                     |
| Default routing   |                            |
| CTR 0 SOURCE .....                                      | PF1 0 routed through DIO 0 |
| CTR 0 GATE .....  | PF1 1 routed through DIO 1 |
| CTR 0 AUX .....   | PF1 2 routed through DIO 2 |
| CTR 0 OUT .....   | PF1 3 routed through DIO 3 |
| FREQ OUT .....  | PF1 4 routed through DIO 4 |
| Data transfers .....                                    | Programmed I/O             |
| Update mode .....                                       | Software-timed             |

### Analog Output

|   |  |
|---|--|
| Number of channels .....                                  | 2 ground-referenced or 1 stereo audio output |
| DAC resolution .....                                      | 16 bits                                      |
| Maximum update rate .....                                 | 200 kS/s                                     |
| Range   |  |
| Analog output .....                                       | $\pm 10$ V, $\pm 2$ V, DC-coupled            |
| Audio output .....  | $\pm 2$ V, AC-coupled                        |
| Maximum output current (analog output) <sup>1</sup> ..... | 2 mA   |

# myDAQ Digital I/O specifications:

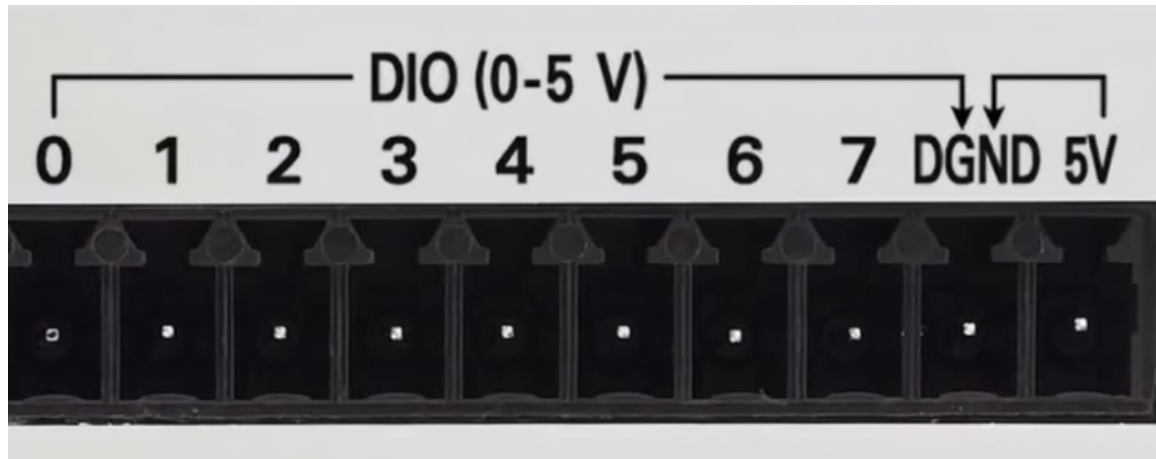
F2018

## Digital In

- 3.3 volts LVTTTL inputs
- 5.0 TTL volt tolerant

## Digital Out

- 3.3 volt LVTTTL
- Max. 4 mA drive



Explain how you can increase the current capacity of the myDAQ (4mA) output.

This may include adding additional circuitry such as a transistors, ULN2003A, MOSFETS. Be able to draw each.

Are the myDAQ inputs TTL tolerant?

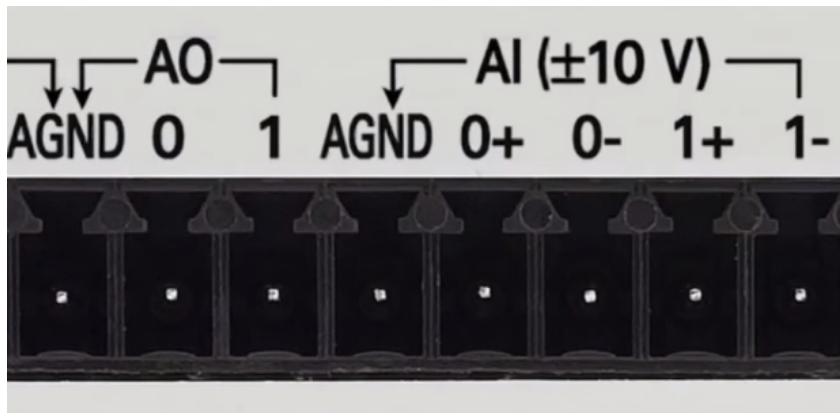
# myDAQ Analog I/O Specifications:

## Analog Out

- +/- 10 volts
- 2 mA drive
- 16 bit resolution
- 200 kilo samples/second

## Analog In

- +/- 10 volts
- 10 G ohms input impedance
- 16 bit resolution
- 200 kilo samples/second



When measuring a signal such as the voltage across the DC motor you must use differential voltage measurements.

In this case neither (0+) or (0-) connect to \_\_\_\_\_?

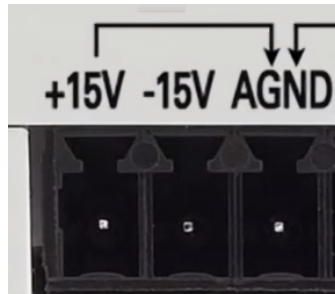
For single ended inputs, use a ground reference, the 0- and 1- inputs should connect to the ground reference.

# myDAQ Power Supply Specifications:

- +/- 15 volts
- 32 mA drive / supply
- +/- 14 volts at max. load
- + 5.0 volts
- 100 mA max. drive
- 4.0 volts(min) at maximum load

Note the current limitations of the supplies. Going over the current limits will cause the supply voltage level to decrease.

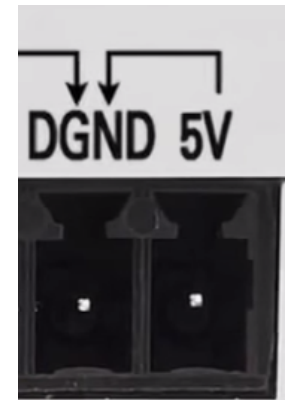
With a load applied to myDAQ supply the supply voltage will drop, at maximum load the supply will drop to a minimum of \_\_\_\_\_ volts.



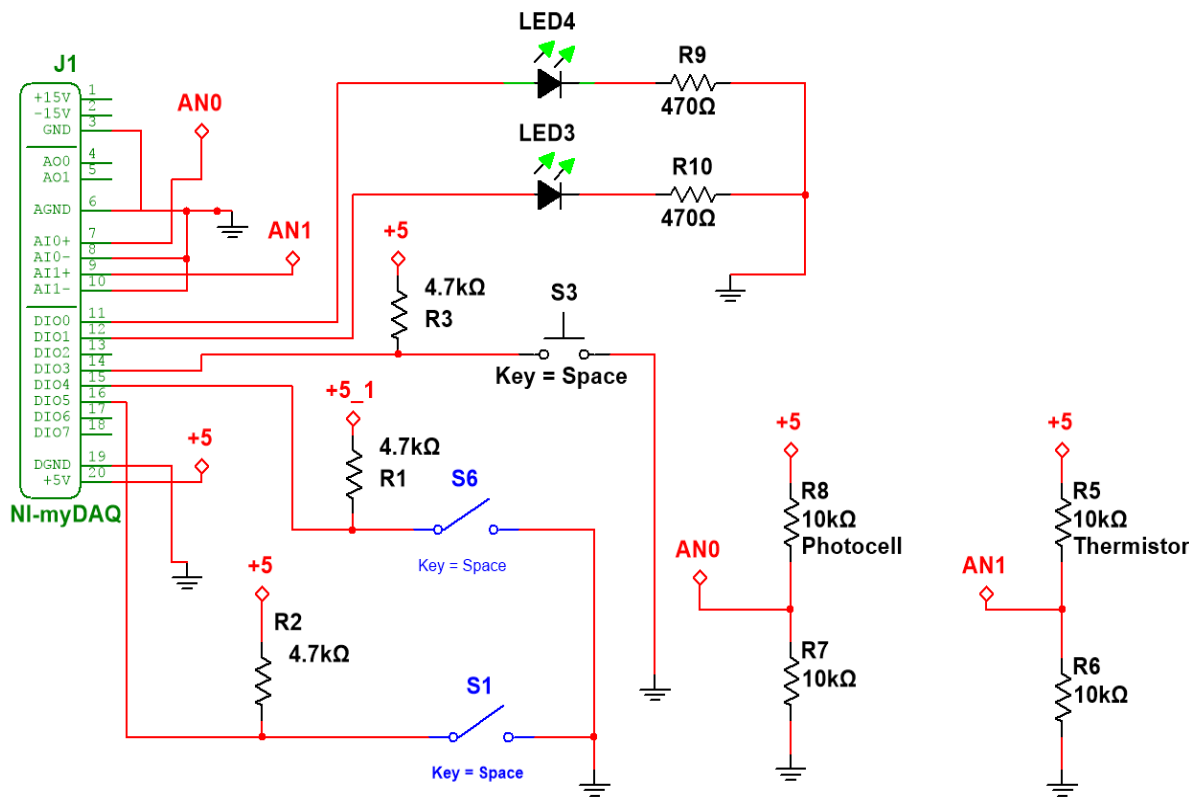
### Caution:

Do not connect another power supply to the myDAQ.

There is a 1.25 A internal fuse.



# myDAQ Basic I/O Circuit:

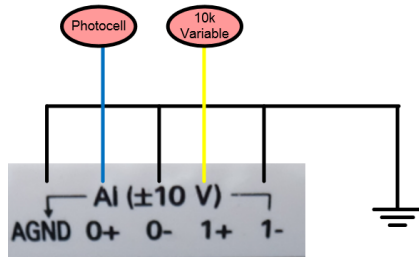


This is a typical schematic using Multisim. The schematic shows all the components, component values and their connection to ground, the supply, other parts of the circuit or to the myDAQ interface device.

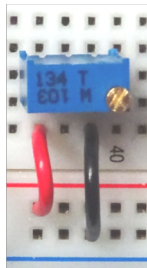
What is the voltage at AN0 when the photocell = 2.2k?

What is the voltage at AN1 when the thermistor = 7.1k?

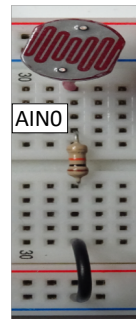
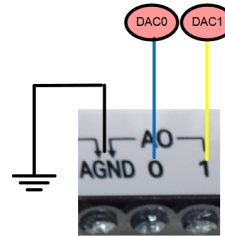
## myDAQ Analog I/O Wiring



The AGND, 0- and 1- must connect to the ground of your circuit when measuring signals that are referenced to ground.

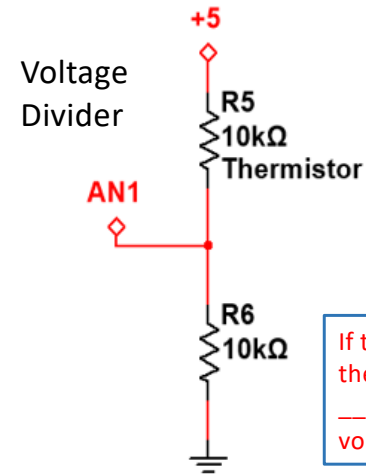


The 10k variable is used to provide a variable DC signal from 0 – 5 volts to the analog inputs. Connect the device so that turning it CW increases the voltage on the middle terminal.



Photocell

A photocell increases resistance in darkness and decreases when exposed to daylight. They typically range from about 200 ohms to 100,000 ohms and are non linear. When the circuit is wired as above the AINO voltage decreases in darkness.



If the temperature is increased the thermistor resistance will \_\_\_\_\_ and the divider voltage will \_\_\_\_\_?

A thermistor decreases resistance as it is heated and increases resistance when cooled. The thermistor used in the lab is:

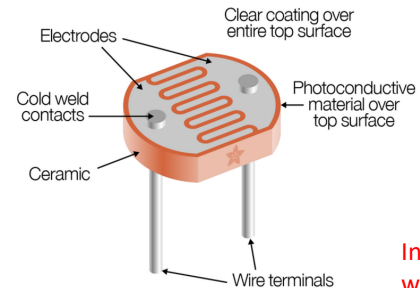
- 10k at 25 degC.
- 8k at 30 degC
- 12.4k at 20 degC.

As the temperature increases the voltage to AN1 increases.

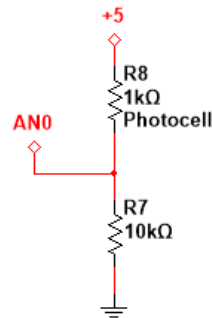
## myDAQ/Arduino Sensors – Photocell

<https://learn.adafruit.com/photocells/measuring-light>

| Illuminance          | Example   |
|----------------------|---|
| 0.002 lux            | Moonless clear night sky                                |
| 0.2 lux              | Design minimum for emergency lighting (AS2293).         |
| 0.27 - 1 lux         | Full moon on a clear night                              |
| 3.4 lux              | Dark limit of civil twilight under a clear sky          |
| 50 lux               | Family living room                                      |
| 80 lux               | Hallway/toilet  |
| 100 lux              | Very dark overcast day                                  |
| 300 - 500 lux        | Sunrise or sunset on a clear day. Well-lit office area. |
| 1,000 lux            | Overcast day; typical TV studio lighting                |
| 10,000 - 25,000 lux  | Full daylight (not direct sun)                          |
| 32,000 - 130,000 lux | Direct sunlight   |



In a dark room the AN0 voltage will be about \_\_\_\_\_ volts?



The photocell resistance decreases with higher visible light and increases in darkness.

Placed in a voltage divider circuit, as shown, the output voltage will increase when exposed to brighter light levels.

(Wikipedia) Illuminance is a measure of how much [luminous flux](#) is spread over a given area. One can think of luminous flux (measured in [lumens](#)) as a measure of the total "amount" of visible light present, and the illuminance as a measure of the intensity of illumination on a surface. A given amount of light will illuminate a surface more dimly if it is spread over a larger area, so illuminance (lux) is inversely proportional to area when the luminous flux (lumens) is held constant.

200 k ohms dark room  
2.0 k ohms office lighting  
200 ohms - daylight

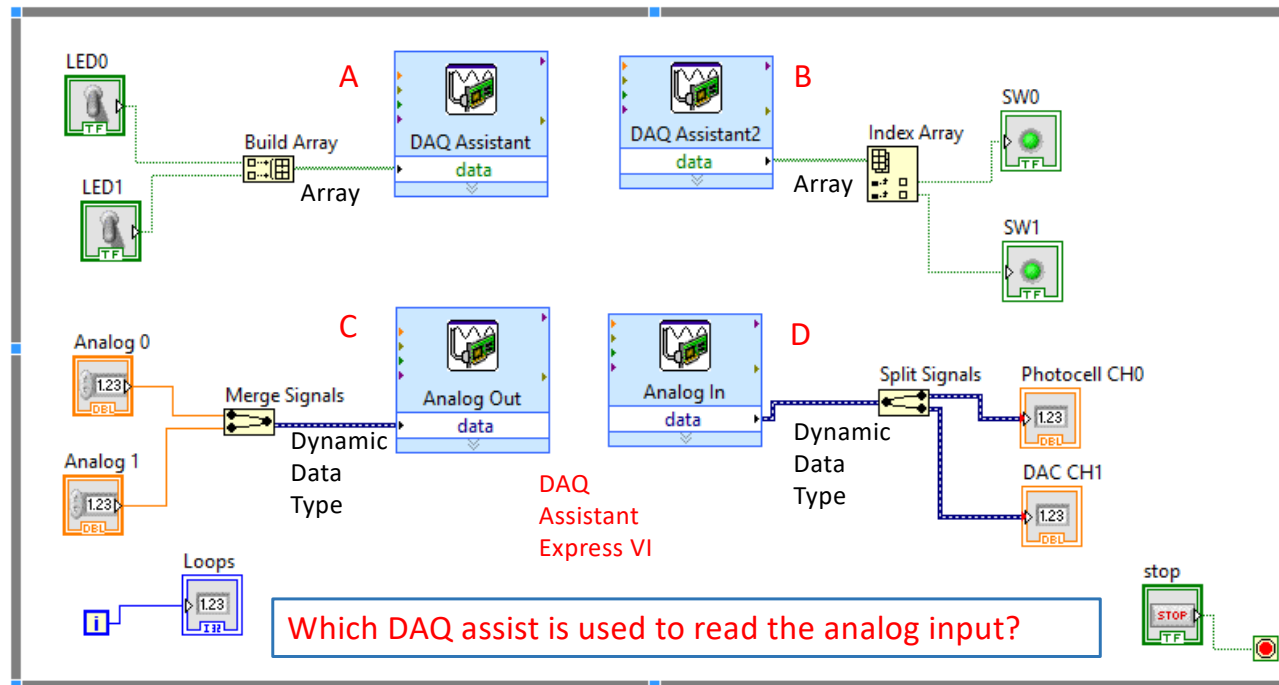
## LabVIEW block diagram used to interface to the myDAQ.

F2018

The DAQ Assistant is a configurable function that is used to allow signals from the myDAQ interface to be controlled or monitored. There are 4 possibilities.

- 1) Digital In to read the digital signals.
- 2) Digital Out to control the logic levels on an output pin.
- 3) Analog In to read the voltage on an analog input pin.
- 4) Analog out to control the voltage on a pin.

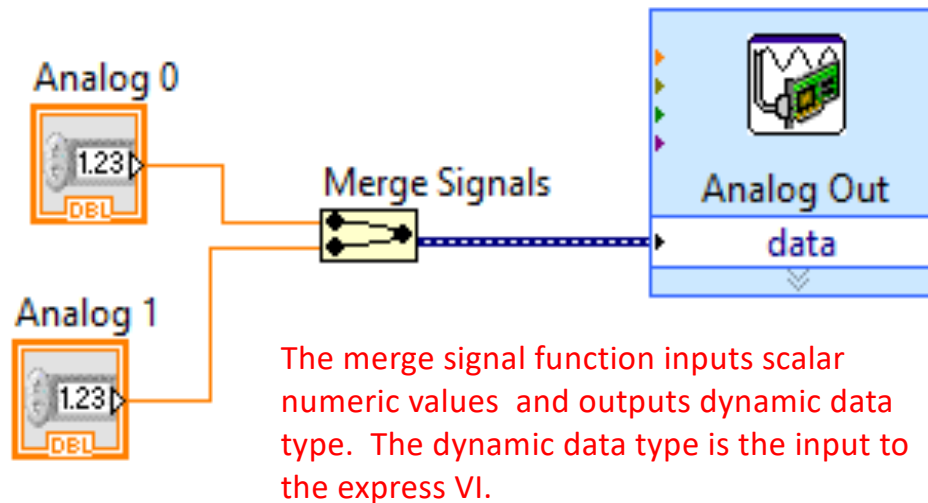
Which DAQ assist is used to control a digital output?



The blue wire is a \_\_\_\_\_ data type?

The orange wire is a \_\_\_\_\_ data type?

### myDAQ Express Analog Output



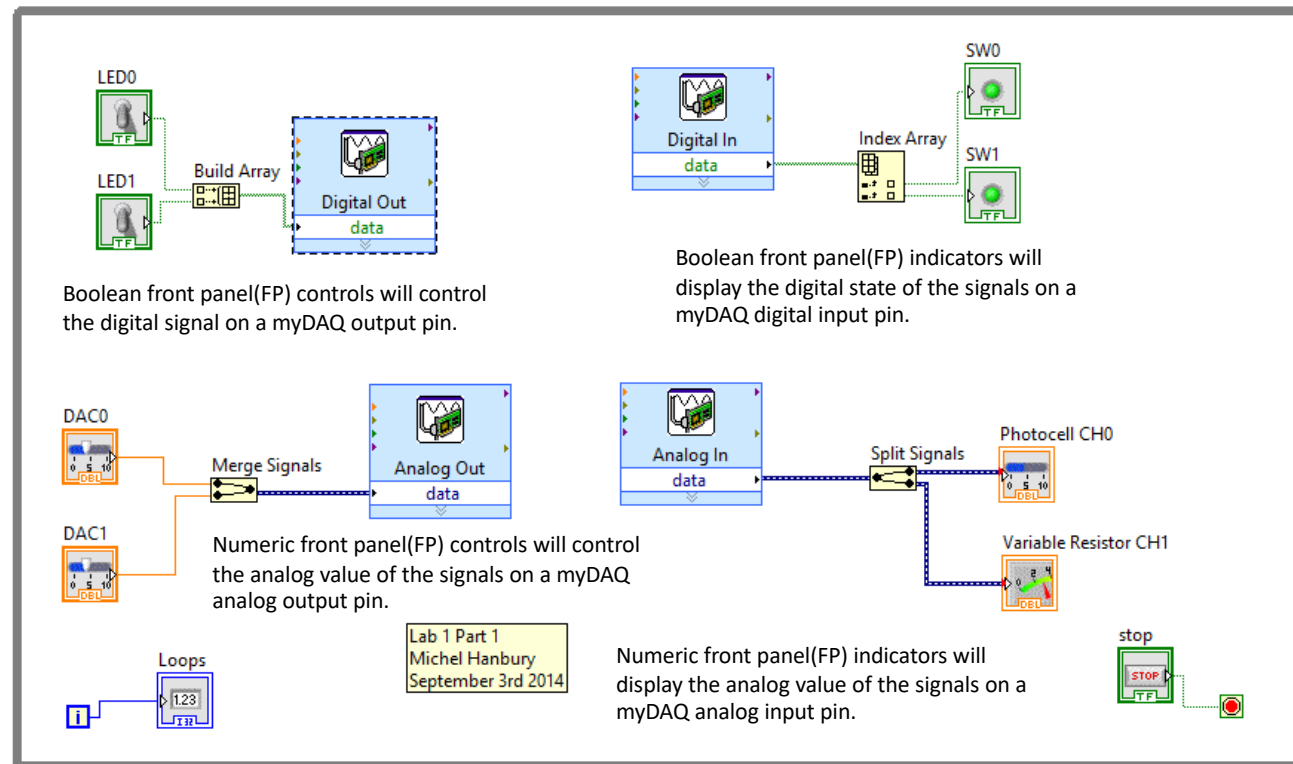
The following is the VI (virtual instrument) or block diagram used to control the analog voltage to the myDAQ analog out terminals. The data into and out from the DAQ assistant for analog data is a Dynamic data type. The merge signals function can combine many scalar data values into one dynamic data type signal. The express VI reads the signals in sequence. The upper value will be applied to the first output signal.

# LabVIEW Block Diagram (Program)

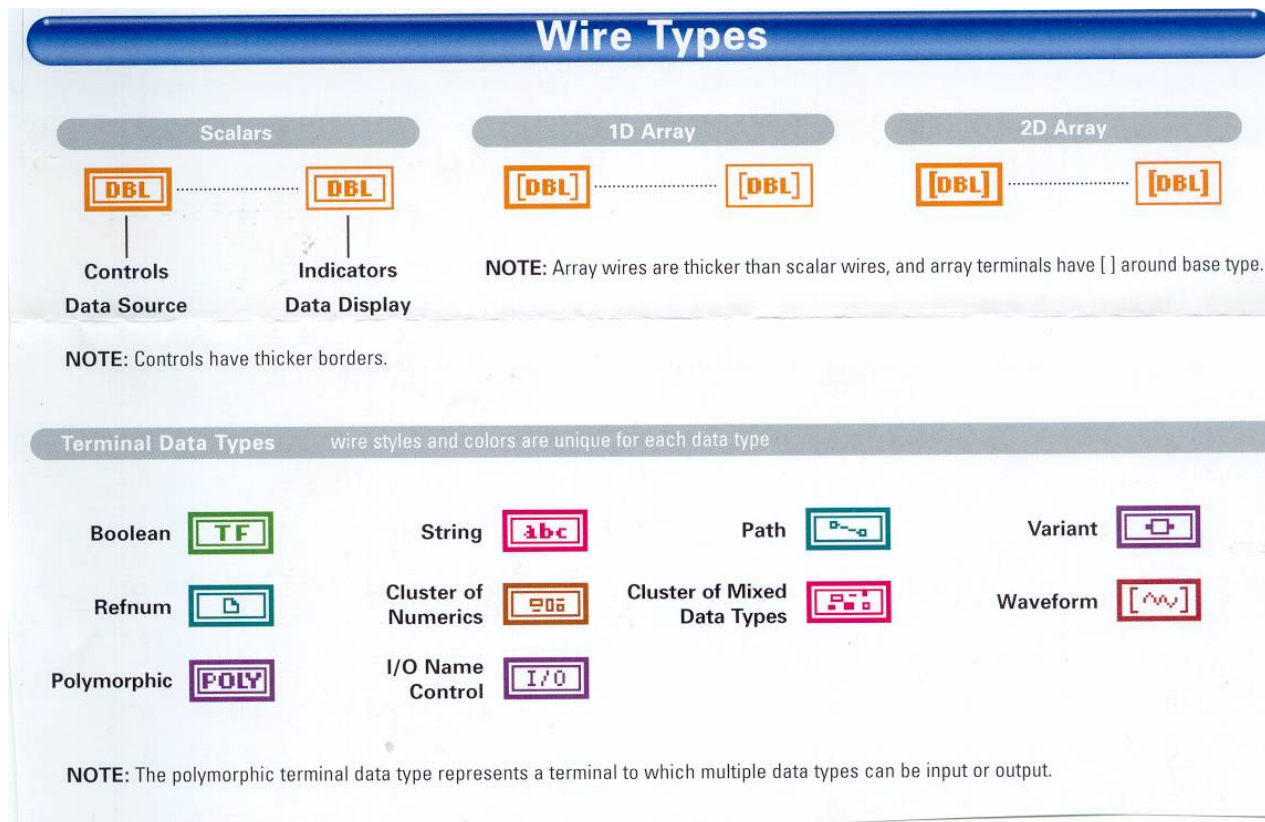
How many elements are in each array?

The blue wires are \_\_\_\_\_ data type?

The green wires are \_\_\_\_\_ data type?



# LabVIEW Wire Colours and Type:



What is the signed and unsigned range of a 8 bit integer?

What is the signed and unsigned range of a 16 bit integer?

What data type is each of the following colours.

Green?

Orange?

Blue?

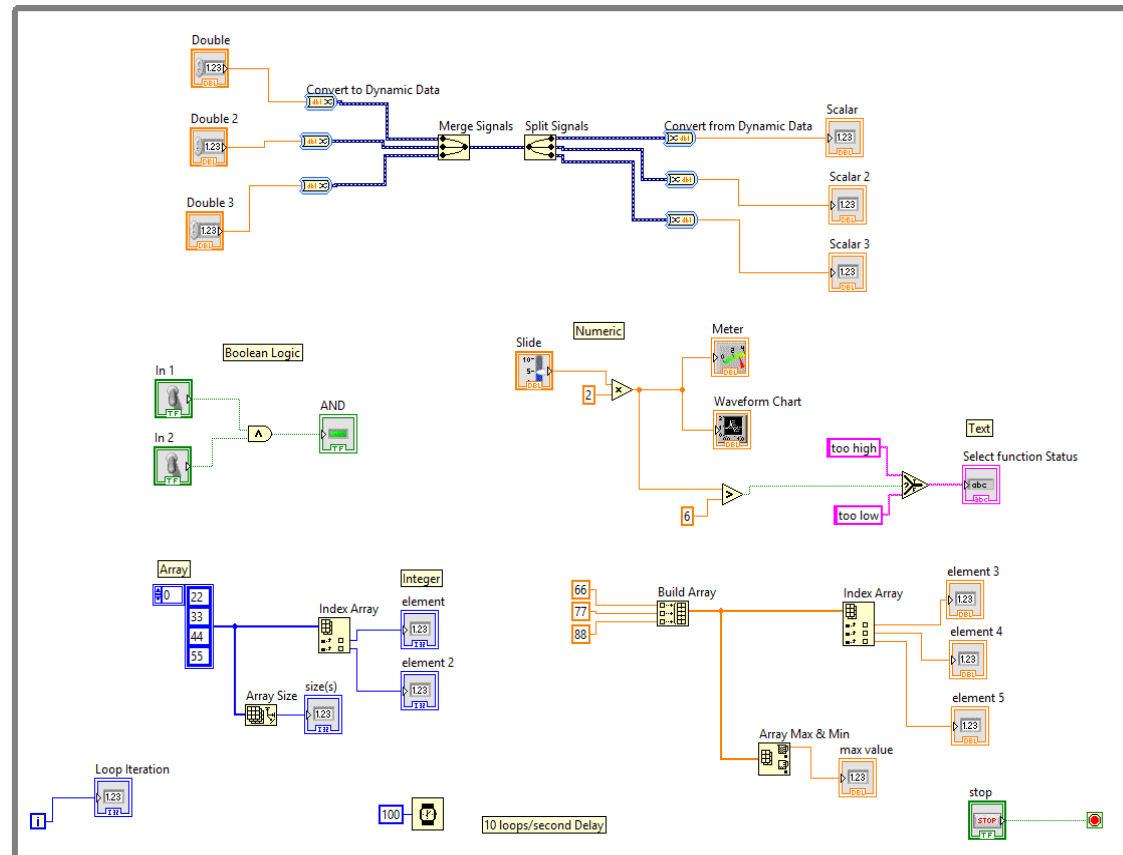
Pink?

# Block Diagram Example:

How many different variable types are shown in this block diagram? List each type.

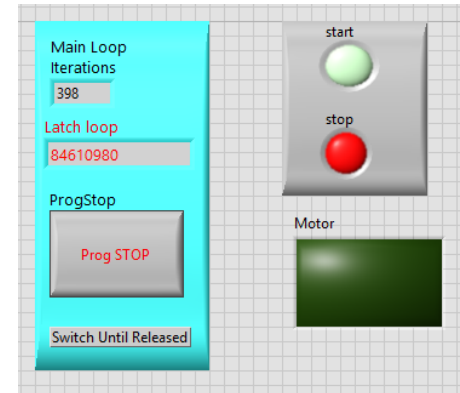
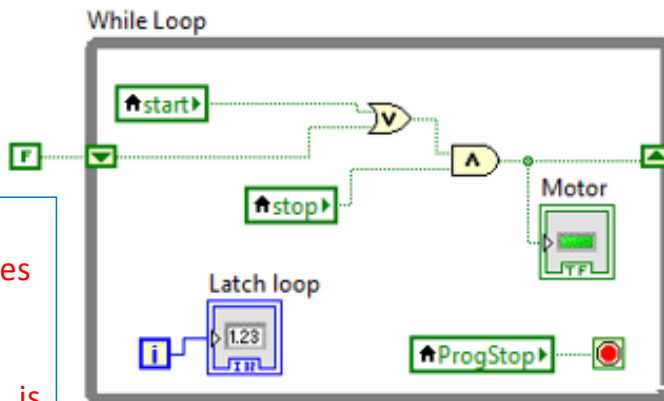
Can the AND gate be used for both Boolean and Integer values?

Determine the value of:  
Max value, element2,  
element 4?



## LabVIEW Shift Register Example

The up and down arrows on the edges of the While Loop indicate that a \_\_\_\_\_ is being used.



### Shift Register 2<sup>nd</sup> Example:

To create the shift register right click on the border of the while loop. Select Add Shift Register.

The motor signal will go true when start is true and remain that way because of the shift register until stop is true. Take some time to understand the shift register, try it out in the lab.

The Boolean constant "F" is true only on the first iteration of the while loop.

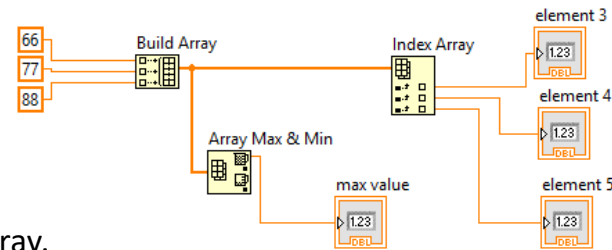
The initial value of Motor equals \_\_\_\_\_?

The false constant is only applied on the first pass through the \_\_\_\_\_?

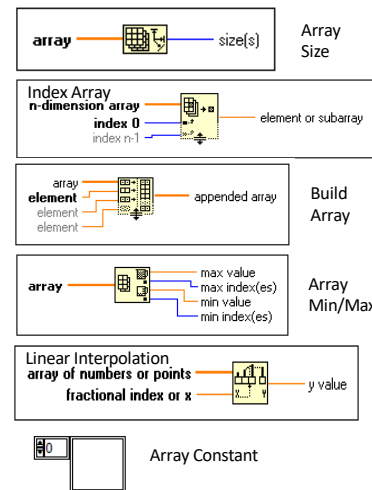
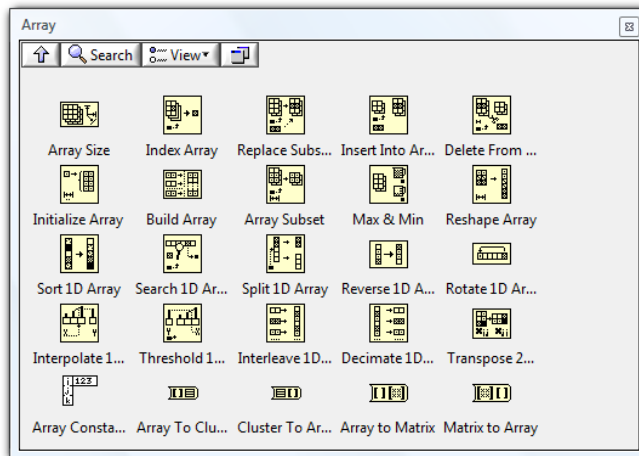
# Arrays Functions:

There are **array functions** to:

- Find the size of an array
- Find the sum of an array.
- Find the min and max value of an array.
- Find the index of the min and max value.
- Find the average of the array values.



What is the value of element 4? \_\_\_\_\_



# Arduino Mega Specs:

Each Arduino output pin can handle \_\_\_\_\_ mA?

The A/D input voltage range is 0 to \_\_\_\_\_ volts.

The A/D number of bits equals \_\_\_\_\_?

Specs (from <http://arduino.cc/en/Main/ArduinoBoardMega>):

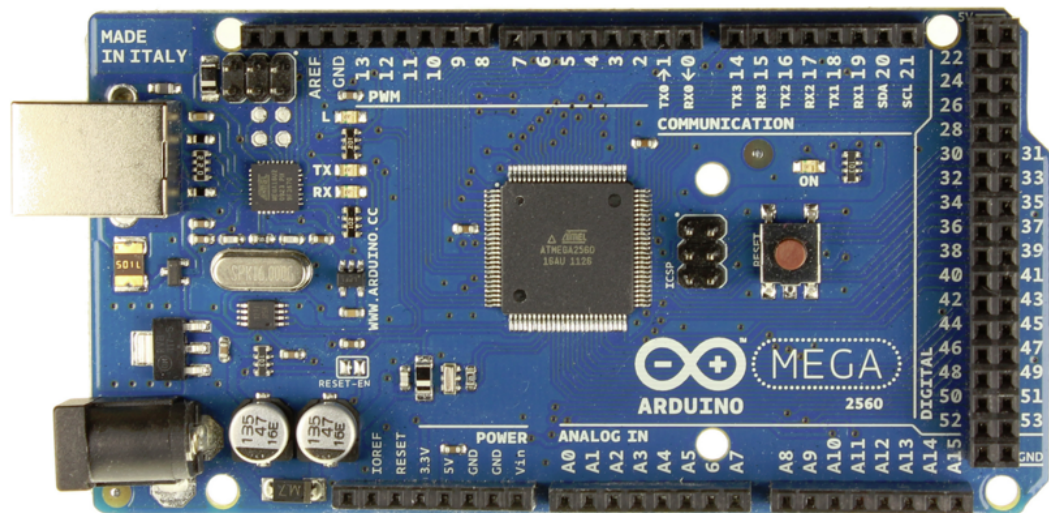
|                             |                                     |   |
|-----------------------------|-------------------------------------|---|
| Microcontroller             | ATmega2560                          | Indicates amount of memory, I/O pins and basic voltage and current specs. |
| Operating Voltage           | 5V                                  |   |
| Input Voltage (recommended) | 7-12V                               |   |
| Input Voltage (limits)      | 6-20V                               |   |
| Digital I/O Pins            | 54 (of which 14 provide PWM output) |   |
| Analog Input Pins           | 16                                  | 10 bit (0-5 volts)  |
| DC Current per I/O Pin      | 40 mA                               |   |
| DC Current for 3.3V Pin     | 50 mA                               |   |
| Flash Memory                | 256 KB                              | of which 8 KB used by bootloader  |
| SRAM                        | 8 KB                                |   |
| EEPROM                      | 4 KB                                |   |
| Clock Speed                 | 16 MHz                              | Flash – non volatile program storage. SRAM – variable storage.            |

Specs (from <http://arduino.cc/en/Main/ArduinoBoardMega>):

|                             |   |
|-----------------------------|---|
| Microcontroller             | ATmega2560                              |
| Operating Voltage           | 5V                                      |
| Input Voltage (recommended) | 7-12V                                   |
| Input Voltage (limits)      | 6-20V                                   |
| Digital I/O Pins            | 54 (of which 14 provide PWM output)     |
| Analog Input Pins           | 16                                      |
| DC Current per I/O Pin      | 40 mA                                   |
| DC Current for 3.3V Pin     | 50 mA                                   |
| Flash Memory                | 256 KB of which 8 KB used by bootloader |
| SRAM                        | 8 KB                                    |

- A/D channels are 10 bits

### Arduino Specs



Identify which pin can be used for:

Connection to ground.

Connection to Vcc

Connect an analog signal.

Can produce PWM.

Can create a hardware interrupt

## setup()

The setup() function is called once when your program starts. Use it to initialize pin modes, or begin serial. It must be included in a program even if there are no statements to run.

```
void setup()
{
  pinMode(pin, OUTPUT);    // sets the 'pin' as output
}
```

## loop()

After calling the setup() function, the loop() function does precisely what its name suggests, and loops consecutively, allowing the program to change, respond, and control the Arduino board.

```
void loop()
{
  digitalWrite(pin, HIGH); // turns 'pin' on
  delay(1000);             // pauses for one second
  digitalWrite(pin, LOW);  // turns 'pin' off
  delay(1000);             // pauses for one second
}
```

What is the  
frequency on the  
output pin?

```

// Convert the analog reading (which goes from 0 - 1023) to a voltage (0 to 5V):
float TempVolts = TempValue * (5.0 / 1023.0); // float or real data type

Start_State = digitalRead(Start_Sw); // read the input
Stop_State = digitalRead(Stop_Sw); // read the input

if (!Start_State) // Wait for the PB to be pressed, causing it to go low.
{
    digitalWrite(WhtLed, HIGH); // turn on led if push button is pressed
    digitalWrite(BrdLed, HIGH); // high turn on the led, pin is connected to anode.
}
/* || OR, && AND, ! NOT */

if (TempVolts > 1.0 || !Stop_State) // turn off led if voltage > 1.5 volts or Stop_Sw is pressed.
{
    digitalWrite(WhtLed, LOW);
    digitalWrite(BrdLed, LOW);
}

```

Does this program use polling or interrupts?

## Code that is run only once

```
void setup()
{
  Serial.begin(9600);           // configure the serial port for 9600 bits / second

  pinMode(Start_Sw, INPUT);    // define line input bit 4
  pinMode(Stop_Sw, INPUT);
  pinMode(WhtLed, OUTPUT);     // define line as output bit 5
  pinMode(BrdLed, OUTPUT);
}
```

How long will it take to transmit one bit serially using a rate of 9600 bits per second?

Other than bit rate what other parameters are required when specifying serial data parameters?

```

--
22 void loop() {
23     float Vsup = 4.61; // measured with meter
24     int Photocell = analogRead(A8);
25     int Pb_state1 = digitalRead(Pb1);
26     int Pb_state2 = digitalRead(Pb2);
27
28     float Vpc = Photocell * (Vsup / 1023.0); // Vin to A/D
29
30     float I = (Vsup - Vpc)/10000;
31
32     float Rpc = Vpc / I;
33
34     float ImA = 1000 * (Vsup - Vpc)/10000;
35

```

If  $V_{pc} = 3.7$  volts  
then  
 $R_{pc} = \underline{\hspace{2cm}}?$   
 $I_{mA} = \underline{\hspace{2cm}}?$

Calculations on data from photocell. The code converts the integer value from the A/D converter to a voltage and then to a resistance.

```

35
36  if (Pb_state1 == LOW)
37     digitalWrite (GrnLed, HIGH);
38  else
39     digitalWrite (GrnLed, LOW);
40
41  if (Pb_state2 == HIGH)
42     digitalWrite (RedLed, HIGH);
43  else
44     digitalWrite (RedLed, LOW);
45
46  if (Rpc > 2000.0) // Getting dark
47     digitalWrite (BoardLed, HIGH);
48  else
49     digitalWrite (BoardLed, LOW);
50

```

PB1 push button is normally HIGH using a pull up resistor. When the push button is pressed the pins short and a connection to ground is made.

Control output based on input conditions.

PB2 is normally LOW, a resistor is used to pull the signal to a low and a short on the pin ties it to +5.

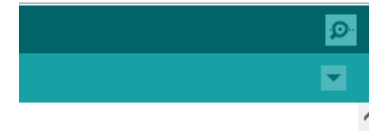
What is the difference between a "==" and "=" in a program?

Pin 13 L on the board.

```

50
51 Serial.print(" PhotoCell Integer = ");
52 Serial.print(Photocell);
53
54 Serial.print(" PhotoCell Volts = ");
55 Serial.print(Vpc, 4);
56
57 Serial.print(" PhotoCell Current mA = ");
58 Serial.print(ImA, 4);
59
60 Serial.print(" PhotoCell Resistance Ohms = ");
61 Serial.print(Rpc);
62
63 Serial.print(" Pb1 state = ");
64 Serial.print(Pb_state1);
65
66 Serial.print(" Pb2 state = ");
67 Serial.println(Pb_state2);
68 delay (50);
69 }
--

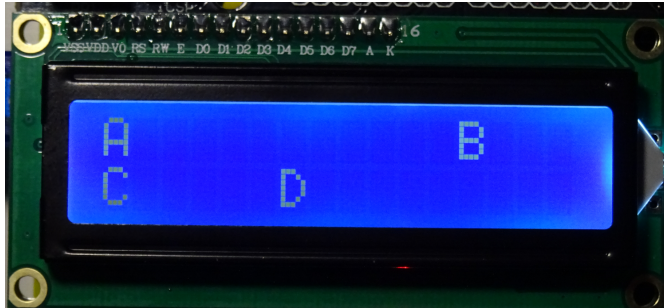
```



The Arduino Serial monitor can be used to output variable data to a computer screen.

The BPS (bits/second) baud rate must match the configuration in software. In this example the rate is 9600 bits per second.

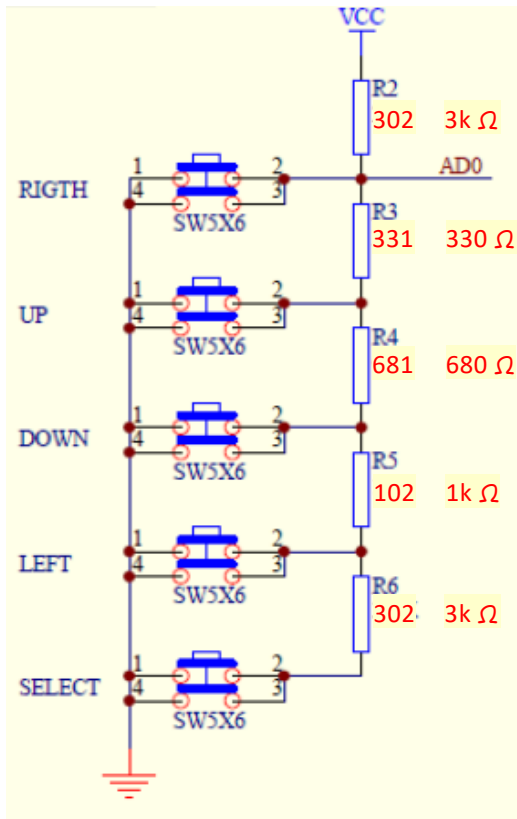
Serial.println moves to the start of the next line, then prints.



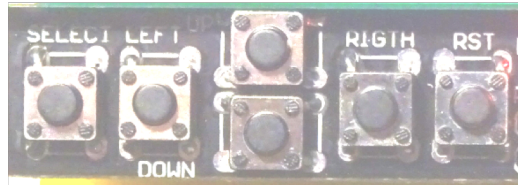
What is the position address of the most right column on the bottom row?

```
19 void loop() {
20
21     lcd.setCursor(0,0);
22     lcd.print("A");
23     lcd.setCursor(12,0);
24     lcd.print("B");
25     lcd.setCursor(0,1);
26     lcd.print("C");
27     lcd.setCursor(6,1);
28     lcd.print("D");
29 }
```

The LCD display is arranged as 2 rows and 16 columns. The numbering starts at 0 for the rows and columns. The first number represents the column the second the row. The top left character is at 0,0. For example 6,1 is the bottom row and the 7<sup>th</sup> position over.



## Arduino LCD / Shield Keypad

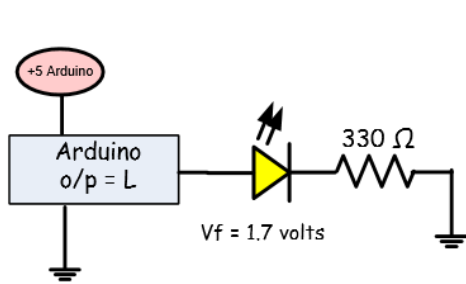


What is the voltage at AD0 when only the Left button is pressed?

The LCD display board also has a set of push button switches which are connected to a set of resistors used in a voltage divider. Pressing different switches will cause the voltage on Analog 0 (AD0) to change. The resistors shown are the actual values used on the board.

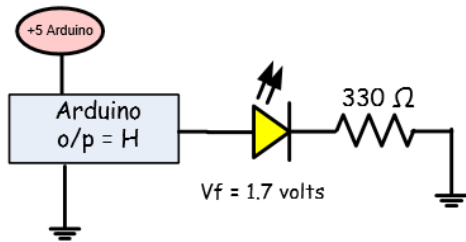
The numbers in red are values I found when testing my keypad. The range may vary by +/- 3.

Right – 0  
 Up – 99  
 Down – 257  
 Left – 411  
 Select 641

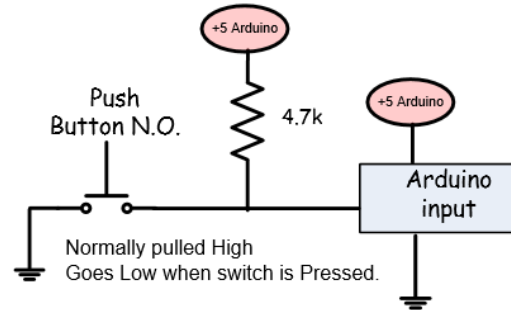


Why is the 330 ohm resistor required in the circuit?

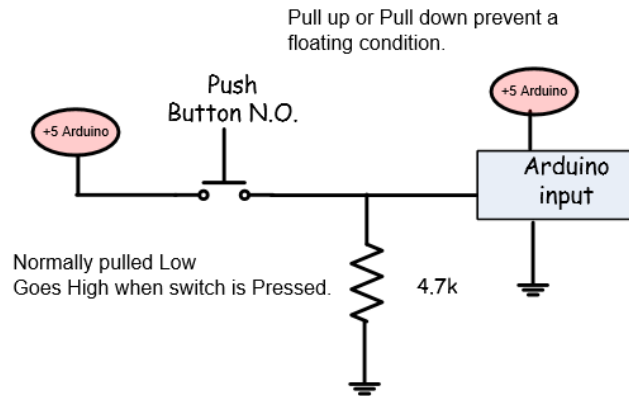
Led is reversed biased  
 Arduino output is about 0 volts.  
 0 volts across resistor.  
 No current flow.  
 Led Off.



Led is forward biased  
 Arduino output is about 5 volts.  
 3.3 volts across resistor.  $I=V/R$   
 $3.3 \text{ volts} / 330 \text{ ohms} = 10 \text{ mA}$   
 Led On.

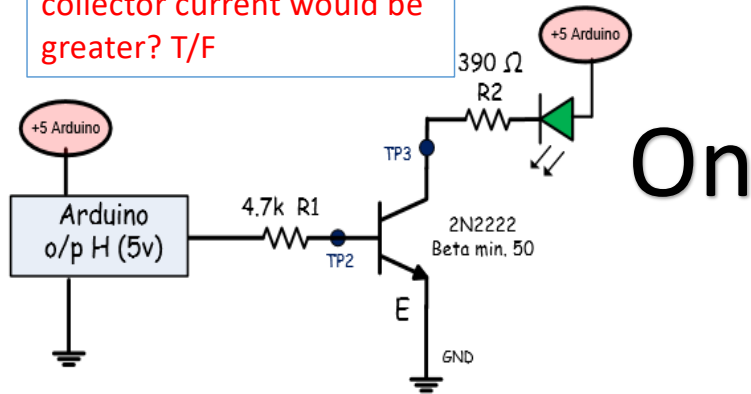


Why is a pull up resistor required in this circuit?

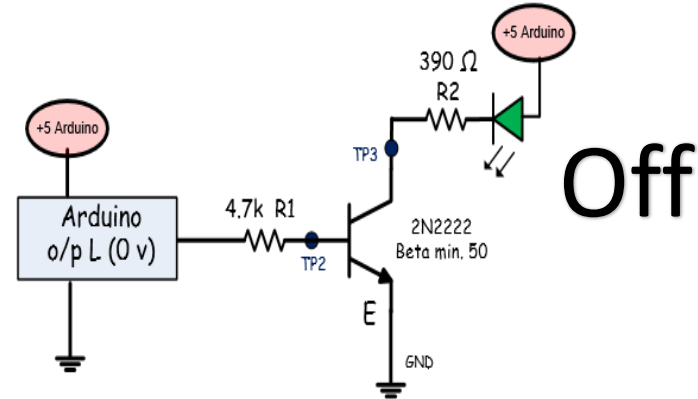


Pull up or Pull down prevent a floating condition.

If the base resistor were changed to 3.3k the collector current would be greater? T/F



- 1) The output from the Arduino is about 5.0 volts.
- 2) Current flows from the Arduino through the resistor, the base and the emitter diode.
- 3) The diode drop is 0.7 volts. The remaining 4.3 volts is across the 4.7k resistor.
- 4)  $4.3\text{v}/4.7\text{k} = 0.91\text{ mA}$  of base current.
- 5) The transistor is on acting like a closed switch.
- 6) The collector voltage equals about 0.1 volts.
- 7) The LED is forward biased with a drop of about 1.7 volts.
- 8)  $5.0\text{ volts} - 1.7\text{ volts}$  leaves about 3.3 volts across the 390 ohm resistor.
- 9)  $3.3\text{ volts}/390\text{ ohms} = 8.5\text{ mA}$ .



- 1) The output from the Arduino is about 0 volts.
- 2) No current flows through the base.
- 3) The emitter diode is reversed biased.
- 4) 0 mA of base current.
- 5) The transistor is off acting like an open switch.
- 6) The collector voltage equals about 3.5 volt. (Transistor 3.5megs, LED 1.5 Megs of resistance.)
- 7) The LED is reversed biased.
- 8) 0 volts across the 390 ohm resistor.
- 9)  $0\text{ volts}/390\text{ ohms} = 0\text{ mA}$ .

## NPN 2N2222 Transistor Switch - "ON"

□ Saturation - Transistor is turned on:  
The collector to emitter junction acts like a closed switch.

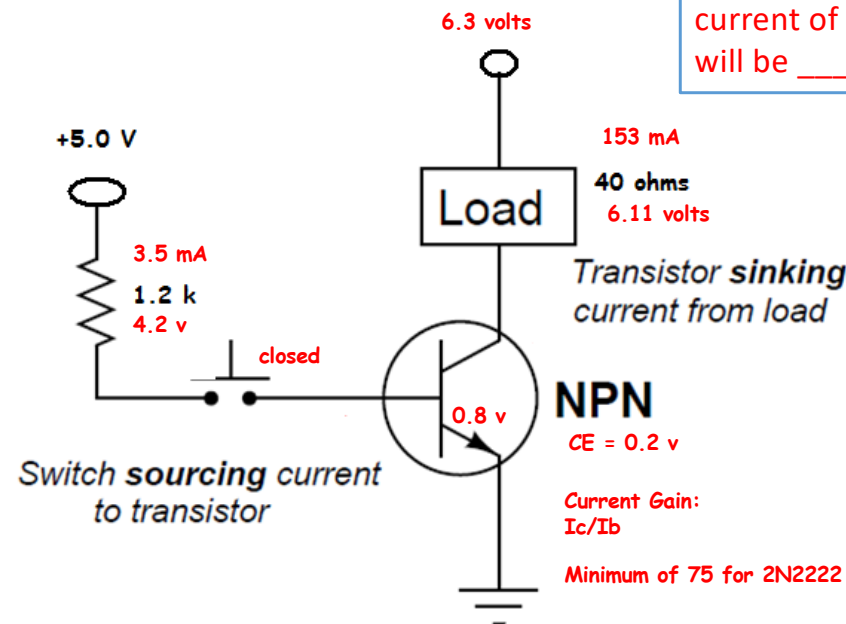
□ The base voltage must be greater than the emitter voltage by 0.8 volts.

□ The transistor has a gain of about 75.

□ Example: If the base current equals 3.5 mA and the gain = 75 mA the transistor will allow up to 263 mA of current to flow.

□ Transistors can only handle a certain amount of collector current specified in the data sheets.

□ The CE junction does not act like a perfect switch and may have a voltage across the CE junction of about 0.2 volts.

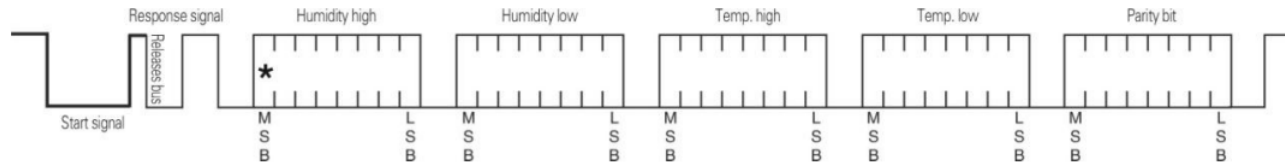


With a load of 40 ohms and a Beta = 75 the collector o/p current with a base current of 1.2 mA will be \_\_\_\_\_ mA?

## DHT22 Digital Temperature and Humidity Sensor (Serial Data Output)

### ◎ Single bus to send data definition

SDA For communication and synchronization between the microprocessor and the AM2302, single-bus data format, a transmission of 40 data, the high first-out. Specific communication timing shown in Figure 5, the communication format is depicted in Table 5.



**Pic5:** AM2302 Single-bus communication protocol

The DHT22 temperature and humidity sensor sends a digital pattern to the Arduino representing % relative humidity and temperature in degrees Celsius .

The first 16 bits transmitted represent the humidity, the second 16 the temperature and the last 8 bits are a checksum to verify the data.

$$\text{Relative Humidity (RH)} = \frac{(\text{Actual Vapor Density})}{(\text{Saturation Vapor Density})} \times 100\%$$

relative humidity

[Word Origin](#)

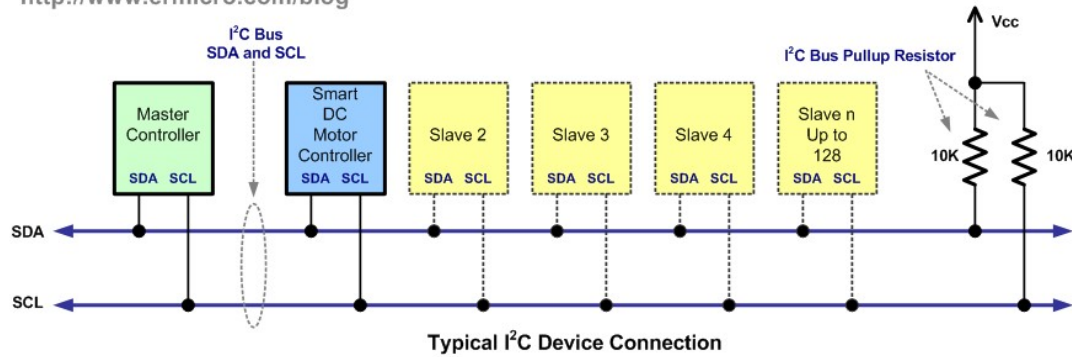
noun

1. the amount of water vapor in the air, expressed as a percentage of the maximum amount that the air could hold at the given temperature; the ratio of the actual water vapor pressure to the saturation vapor pressure.

**Abbreviation:** RH, rh.

The last byte is used for \_\_\_\_\_ checking?

<http://www.ermicro.com/blog>



Data on the I2C bus can be transferred in three modes:

- 1) Standard Mode: 100kbps.
- 2) Fast Mode: 400kbps
- 3) High Speed Mode: 3.4Mbps.

Only 2 device can be connected at once to the I2C bus? T/F

The I2C bus transmits data faster than the asynchronous serial I/O? T/F

The I2C uses Vcc, ground and 2 signal wires.

SDA – serial data

SCL – serial clock.

The bus is used to connect many devices to the I2C signals on the Arduino or other controllers such as a Raspberry PI low cost computer.



RaspberryPi

| Type     | Part #    | Notes                           | Address | Address | Address |
|----------|-----------|---------------------------------|---------|---------|---------|
| OLED     | SSD1306   | One addr pin                    | 0x3C    | 0x3D    |         |
| NFC/RFID | PN532     |                                 | 0x48    |         |         |
| Sensor   | TSL2561   | 1 tri-state addr pin            | 0x29    | 0x39    | 0x49    |
| Sensor   | BMP085    | Can use XCLR to select/deselect | 0x77    |         |         |
| Sensor   | ADXL345   | 1 address pin                   | 0x1D    | 0x53    |         |
| Sensor   | HMC5883L  |                                 | 0x1E    |         |         |
| Sensor   | BMA180    |                                 | 0x77    |         |         |
| Sensor   | MMA7455L  | Same as ADXL345                 | 0x1D    |         |         |
| RTC      | DS1307    | Same as DS3231                  | 0x68    |         |         |
| RTC      | DS3231    | Same as DS1307                  | 0x68    |         |         |
| DAC      | MCP4725A0 | 1 addr pin, 3 variants          | 0x60    | 0x61    |         |
| DAC      | MCP4725A1 | 1 addr pin, 3 variants          | 0x62    | 0x63    |         |
| DAC      | MCP4725A2 | 1 addr pin, 3 variants          | 0x64    | 0x65    |         |

Each I2C device has a specific 7 bit address to select the I2C device.

In the lab three I2C devices will be used.

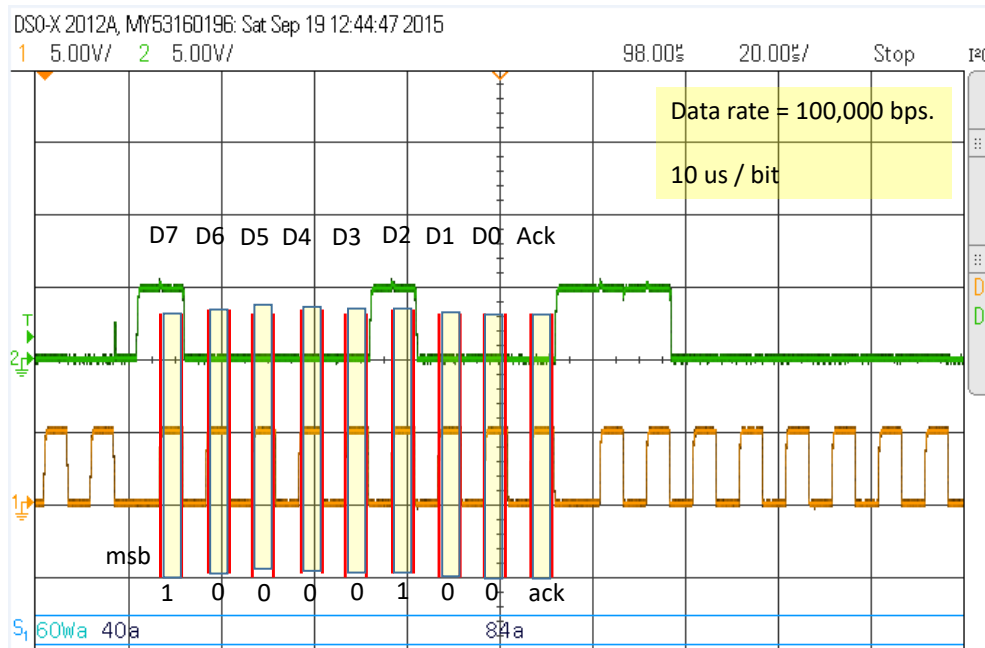
A MCP4725 12 bit DAC.

An ADXL345 Accelerometer.

A DS3231 real time clock.

There are many I2C devices that can be connected to the Arduino. Displays, pressure and temperature sensors, etc.

## MCP4725 12 bit DAC Waveform



MCP4725 I2C DAC  
 Data = FFF

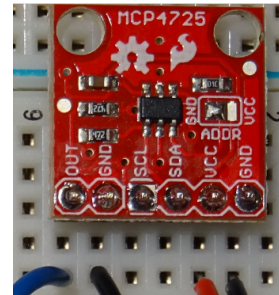
$FFF/FFF * 5.0 \text{ volts}$   
 = 5.0

Based on this scope screen capture; determine the approximate clock frequency.

Note: Horiz time ref = 20 us/div.

## MCP4725 12 bit DAC (Arduino sketch)

```
10 #include <Wire.h> // Include the Wire library to talk I2C
11
12 // This is the I2C Address of the MCP4725, by default (A0 pulled to GND).
13
14 #define MCP4725_ADDR 0x60 // address 0x60
15 // For devices with A0 pulled HIGH, use 0x61
16
17 void setup()
18 {
19     Wire.begin(); // start 2 wire IIC
20 }
21 void loop() // runs endlessly
22 {
23
24     Wire.beginTransmission(MCP4725_ADDR); // 12 bit DAC
25     Wire.write(0x40); // cmd to update the DAC
26     Wire.write(0x7A); // the 8 most significant bits...
27     Wire.write(0xD0); // the 4 least significant bits...
28     Wire.endTransmission();
29     delay(3000);
30 }
```



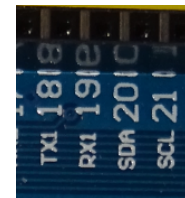
MCP4725 I2C DAC  
Data = 0x7AD  
First 12 bits of the two  
byte number

= 1965 decimal

1965/4096  
= 0.4797

0.4797 \* 5.0 volts  
= 2.4 volts.

The DAC I2C address  
equals?



The DAC uses \_\_\_\_\_  
bits?

If the 7A in the data  
changed to 4D the  
o/p voltage would  
equal?

# ASCII TABLE

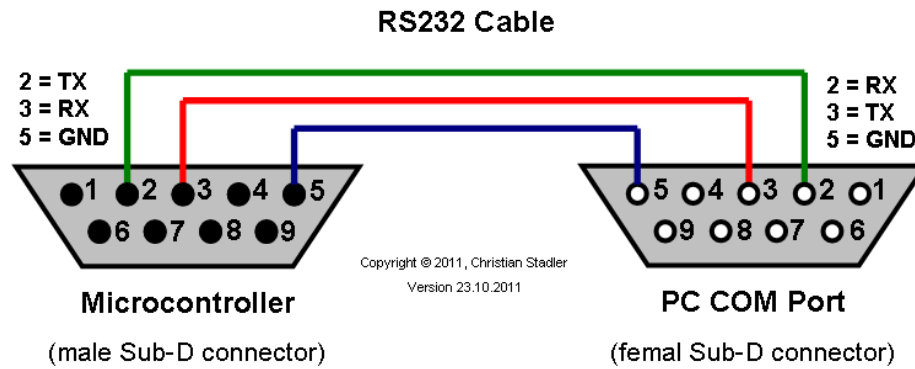
American Standard Code for Information Interchange

| Decimal | Hex | Char                   | Decimal | Hex | Char    | Decimal | Hex | Char | Decimal | Hex | Char  |
|---------|-----|------------------------|---------|-----|---------|---------|-----|------|---------|-----|-------|
| 0       | 0   | [NULL]                 | 32      | 20  | [SPACE] | 64      | 40  | @    | 96      | 60  | `     |
| 1       | 1   | [START OF HEADING]     | 33      | 21  | !       | 65      | 41  | A    | 97      | 61  | a     |
| 2       | 2   | [START OF TEXT]        | 34      | 22  | "       | 66      | 42  | B    | 98      | 62  | b     |
| 3       | 3   | [END OF TEXT]          | 35      | 23  | #       | 67      | 43  | C    | 99      | 63  | c     |
| 4       | 4   | [END OF TRANSMISSION]  | 36      | 24  | \$      | 68      | 44  | D    | 100     | 64  | d     |
| 5       | 5   | [ENQUIRY]              | 37      | 25  | %       | 69      | 45  | E    | 101     | 65  | e     |
| 6       | 6   | [ACKNOWLEDGE]          | 38      | 26  | &       | 70      | 46  | F    | 102     | 66  | f     |
| 7       | 7   | [BELL]                 | 39      | 27  | '       | 71      | 47  | G    | 103     | 67  | g     |
| 8       | 8   | [BACKSPACE]            | 40      | 28  | (       | 72      | 48  | H    | 104     | 68  | h     |
| 9       | 9   | [HORIZONTAL TAB]       | 41      | 29  | )       | 73      | 49  | I    | 105     | 69  | i     |
| 10      | A   | [LINE FEED]            | 42      | 2A  | *       | 74      | 4A  | J    | 106     | 6A  | j     |
| 11      | B   | [VERTICAL TAB]         | 43      | 2B  | +       | 75      | 4B  | K    | 107     | 6B  | k     |
| 12      | C   | [FORM FEED]            | 44      | 2C  | ,       | 76      | 4C  | L    | 108     | 6C  | l     |
| 13      | D   | [CARRIAGE RETURN]      | 45      | 2D  | -       | 77      | 4D  | M    | 109     | 6D  | m     |
| 14      | E   | [SHIFT OUT]            | 46      | 2E  | .       | 78      | 4E  | N    | 110     | 6E  | n     |
| 15      | F   | [SHIFT IN]             | 47      | 2F  | /       | 79      | 4F  | O    | 111     | 6F  | o     |
| 16      | 10  | [DATA LINK ESCAPE]     | 48      | 30  | 0       | 80      | 50  | P    | 112     | 70  | p     |
| 17      | 11  | [DEVICE CONTROL 1]     | 49      | 31  | 1       | 81      | 51  | Q    | 113     | 71  | q     |
| 18      | 12  | [DEVICE CONTROL 2]     | 50      | 32  | 2       | 82      | 52  | R    | 114     | 72  | r     |
| 19      | 13  | [DEVICE CONTROL 3]     | 51      | 33  | 3       | 83      | 53  | S    | 115     | 73  | s     |
| 20      | 14  | [DEVICE CONTROL 4]     | 52      | 34  | 4       | 84      | 54  | T    | 116     | 74  | t     |
| 21      | 15  | [NEGATIVE ACKNOWLEDGE] | 53      | 35  | 5       | 85      | 55  | U    | 117     | 75  | u     |
| 22      | 16  | [SYNCHRONOUS IDLE]     | 54      | 36  | 6       | 86      | 56  | V    | 118     | 76  | v     |
| 23      | 17  | [ENG OF TRANS. BLOCK]  | 55      | 37  | 7       | 87      | 57  | W    | 119     | 77  | w     |
| 24      | 18  | [CANCEL]               | 56      | 38  | 8       | 88      | 58  | X    | 120     | 78  | x     |
| 25      | 19  | [END OF MEDIUM]        | 57      | 39  | 9       | 89      | 59  | Y    | 121     | 79  | y     |
| 26      | 1A  | [SUBSTITUTE]           | 58      | 3A  | :       | 90      | 5A  | Z    | 122     | 7A  | z     |
| 27      | 1B  | [ESCAPE]               | 59      | 3B  | ;       | 91      | 5B  | [    | 123     | 7B  | {     |
| 28      | 1C  | [FILE SEPARATOR]       | 60      | 3C  | <       | 92      | 5C  | \    | 124     | 7C  |       |
| 29      | 1D  | [GROUP SEPARATOR]      | 61      | 3D  | =       | 93      | 5D  | ]    | 125     | 7D  | }     |
| 30      | 1E  | [RECORD SEPARATOR]     | 62      | 3E  | >       | 94      | 5E  | ^    | 126     | 7E  | ~     |
| 31      | 1F  | [UNIT SEPARATOR]       | 63      | 3F  | ?       | 95      | 5F  | _    | 127     | 7F  | [DEL] |

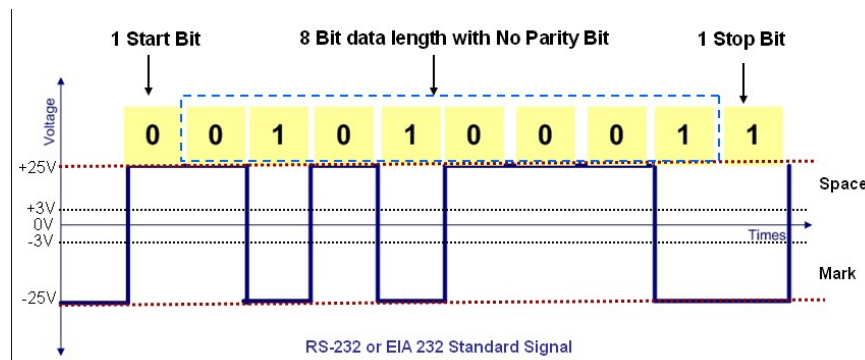
What is the ASCII code for the number "5" and the uppercase letter "j"?

The ASCII table shows the standard code used to transfer characters.

Examples: "1" = 0x31, "A" = 0x41 "U" = 0x55 or 01010101 in binary



An RS-232 serial connector has 9 pins. At least 3 pins are required to transmit and receive data

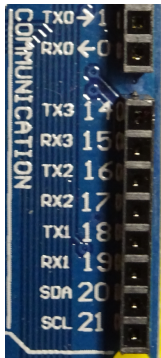


Do RS-232 levels go to a negative voltage?

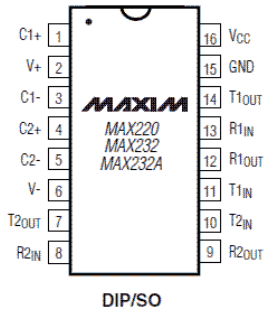
If the logic level of the TXD signal is 5.0 volts the RS-232 level equals \_\_\_\_\_ volts.

At TTL level the data begins with a low start bit, 8 data bits and a high stop bit.

To transmit at greater distance RS-232 levels are used. Typical values are +/- 12 volts.

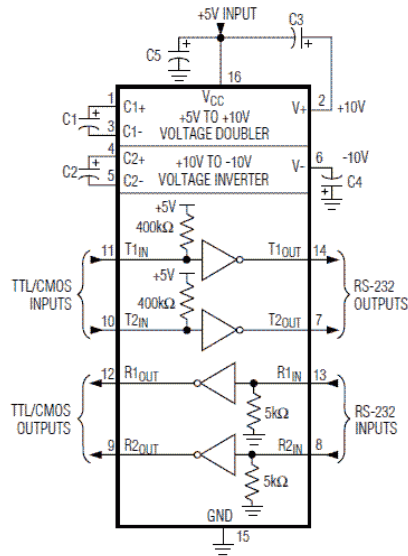


TOP VIEW



| CAPACITANCE (µF) |       |      |      |      |      |
|------------------|-------|------|------|------|------|
| DEVICE           | C1    | C2   | C3   | C4   | C5   |
| MAX220           | 0.047 | 0.33 | 0.33 | 0.33 | 0.33 |
| MAX232           | 1.0   | 1.0  | 1.0  | 1.0  | 1.0  |
| MAX232A          | 0.1   | 0.1  | 0.1  | 0.1  | 0.1  |

Diagrams continued in the full data sheet.



To convert from TTL levels to RS-232 levels it is common to use a level translator called a MAX232. This chip is manufactured by MAXIM.

The chip converts a logic high (5 volts) to (-12 volts)

A logic low (0 volts) is converted to (+12 volts DC)

The TX0 signal on the Arduino is at \_\_\_\_\_ levels. A \_\_\_\_\_ is required when interfacing to RS-232 levels.



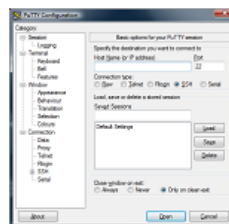
A typical USB to RS-232 converter.

| Baud   | Bit Width | Real Data Rate *<br>(1 start bit, 8 data bits, no parity, 1 stop bit) |
|--------|-----------|---|
| 4800   | 208.33us  | 480 Bytes/s 0.47 KBytes/s   |
| 9600   | 104.17us  | 960 Bytes/s 0.94 KBytes/s   |
| 19200  | 52.08us   | 1920 Bytes/s 1.88 KBytes/s  |
| 38400  | 26.04us   | 3840 Bytes/s 3.75 KBytes/s  |
| 57600  | 17.36us   | 5760 Bytes/s 5.63 KBytes/s  |
| 115200 | 8.68us    | 11520 Bytes/s 11.25 KBytes/s  |

Standard asynchronous serial bit rates.

At 19200 bits per second it will take \_\_\_\_\_ seconds to transmit 1000 bytes using 8 data, 1 stop and no parity?

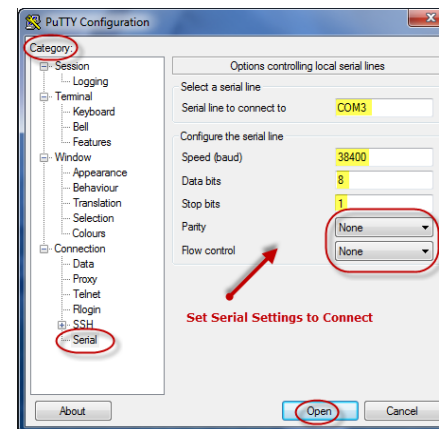
PuTTY is a common (free) serial terminal program for a PC or MAC computer. It can be used to receive and transmit serial data from the Arduino or myDAQ to a PC or MAC.



### Download PuTTY

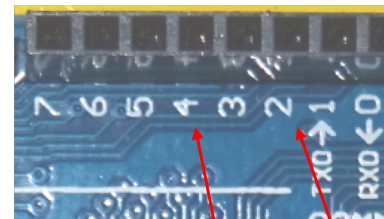
PuTTY is an SSH and telnet client, by a group of volunteers.

You can download PuTTY [here](http://www.putty.org).





```
1  /* Lab 2 Part 1 CAM8302E
2   * Modified by Michel Hanbury
3   * September 28th 2015 */
4
5  #define set_sw 4
6  #define clear_sw 5 // digital inputs
7  #define RedLed 2
8
9  void setup()
10 {
11   Serial.begin(9600); // set bit rate to 9600 bps
12
13   pinMode(set_sw, INPUT);
14   pinMode(clear_sw, INPUT);
15   pinMode(RedLed, OUTPUT); // on power up pins default to input
16
17 }
```



Pin 2  
Digital Output

Pin 4  
Digital input

### Arduino I/O setup

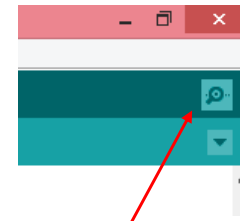
```

8 #include <DHT.h>
9
10 #define DHTPIN 22 // what pin we're connected to
11 #define DHTTYPE DHT22 // DHT 22 (AM2302)
12
13 // Initialize DHT sensor for normal 16mhz Arduino
14 DHT dht(DHTPIN, DHTTYPE);
15
16 void setup(){
17     Serial.begin(9600); // set bit rate to 9600 bps
18 }
19 void loop(){
20
21     float temperature = dht.readTemperature();
22     float humidity = dht.readHumidity();
23
24     Serial.print(temperature);
25     Serial.print(" deg C ");
26
27     Serial.print(humidity);
28     Serial.println(" % hum");
29
30     delay(2000); // wait 2 seconds
31 }
32

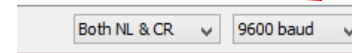
```

DHT Library

Library Functions

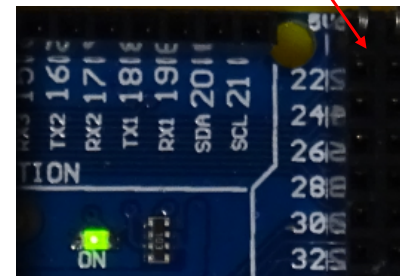


Serial Port Viewer



Baud must match setting in your program.

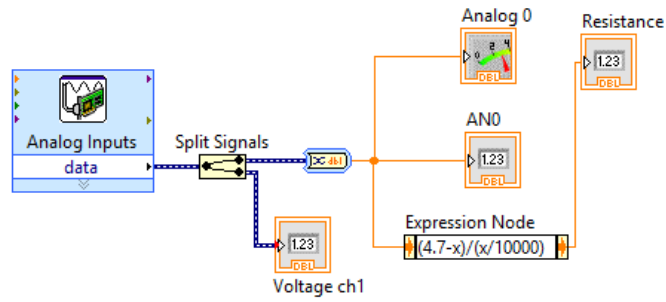
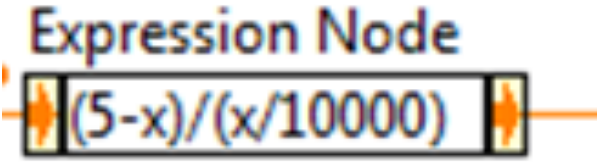
Sensor Pin (22)



# Thermistor Data and Conversion

|      |       |      |        |
|------|-------|------|--------|
| 10,0 | 19691 | 26,0 | 9577,7 |
| 11,0 | 18788 | 27,0 | 9175,6 |
| 12,0 | 17932 | 28,0 | 8792,6 |
| 13,0 | 17120 | 29,0 | 8427,7 |
| 14,0 | 16350 | 30,0 | 8080,0 |
| 15,0 | 15618 | 31,0 | 7748,5 |
| 16,0 | 14923 | 32,0 | 7432,4 |
| 17,0 | 14236 | 33,0 | 7131,0 |
| 18,0 | 13636 | 34,0 | 6843,4 |
| 19,0 | 13040 | 35,0 | 6569,0 |
| 20,0 | 12474 | 36,0 | 6307,0 |
| 21,0 | 11928 | 37,0 | 6057,0 |
| 22,0 | 11409 | 38,0 | 5818,1 |
| 23,0 | 10915 | 39,0 | 5590,0 |
| 24,0 | 10446 | 40,0 | 5372,0 |
| 25,0 | 10000 |      |        |

If the input to the expression is 2.9 volts, the output of the expression equals \_\_\_\_\_. The temperature equals \_\_\_\_\_ degrees Celsius?



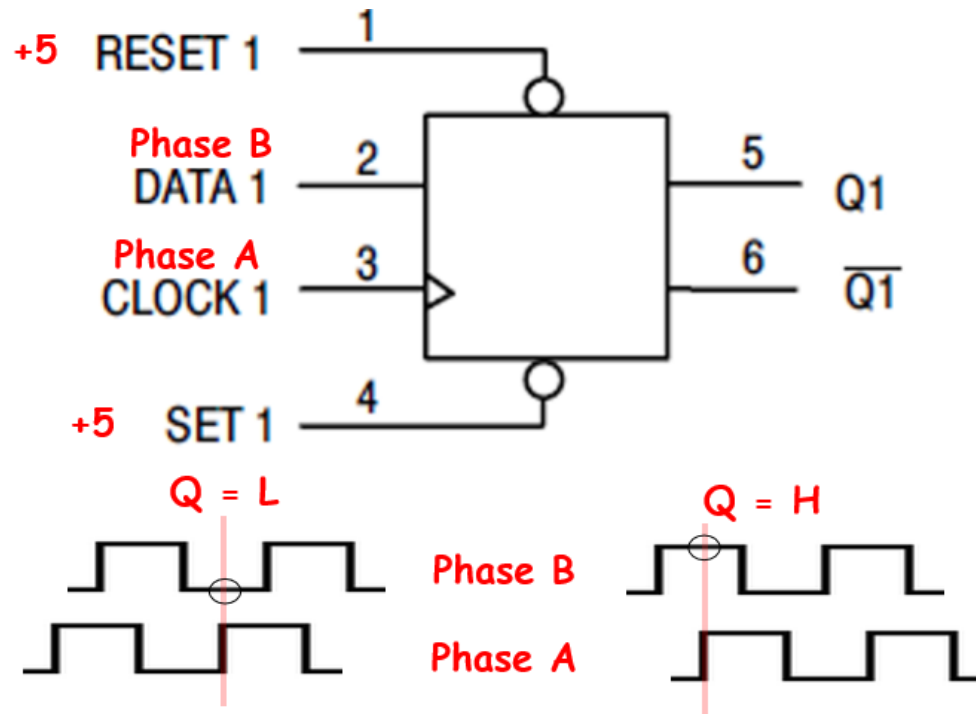


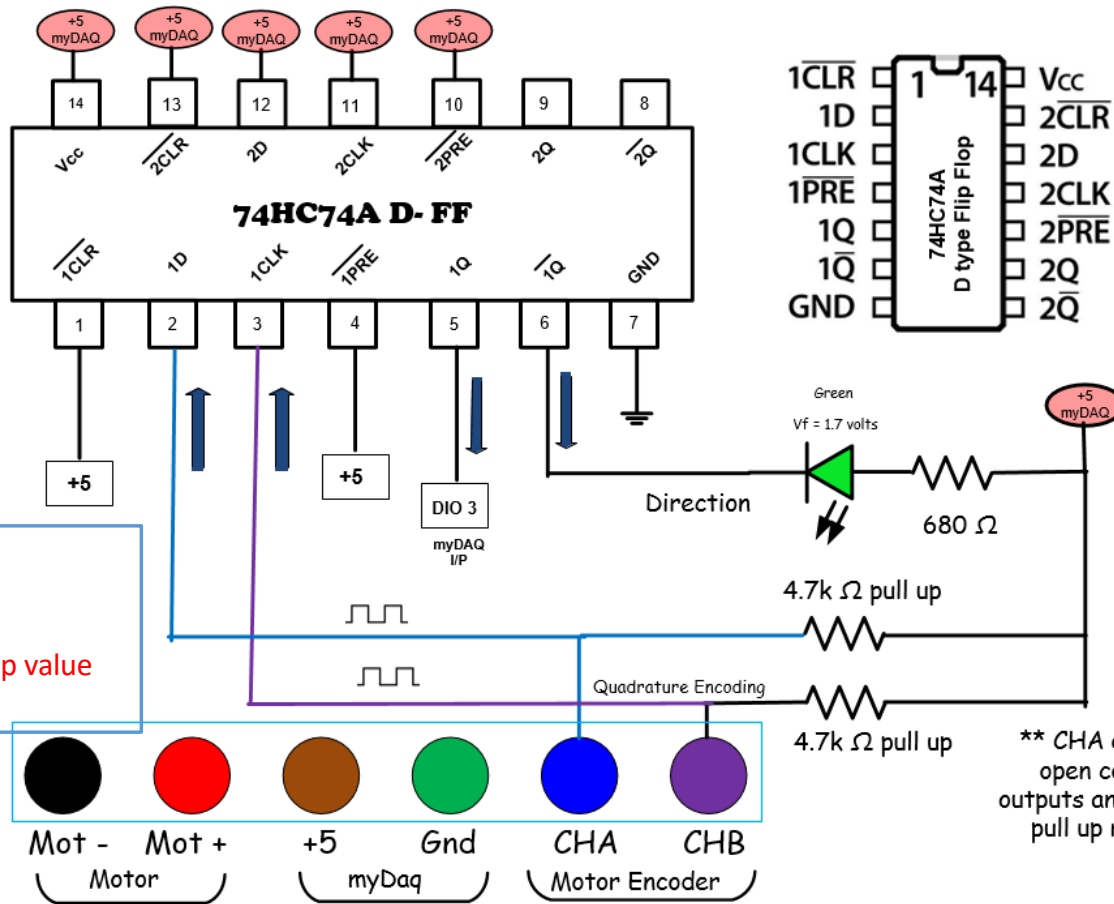




# Direction Detection Using Encoder

If pin 2 is high and the clock goes through a rising edge /Q1 would be a logic \_\_\_\_\_?

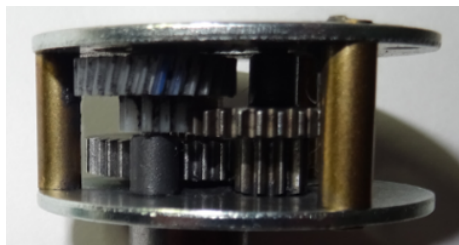
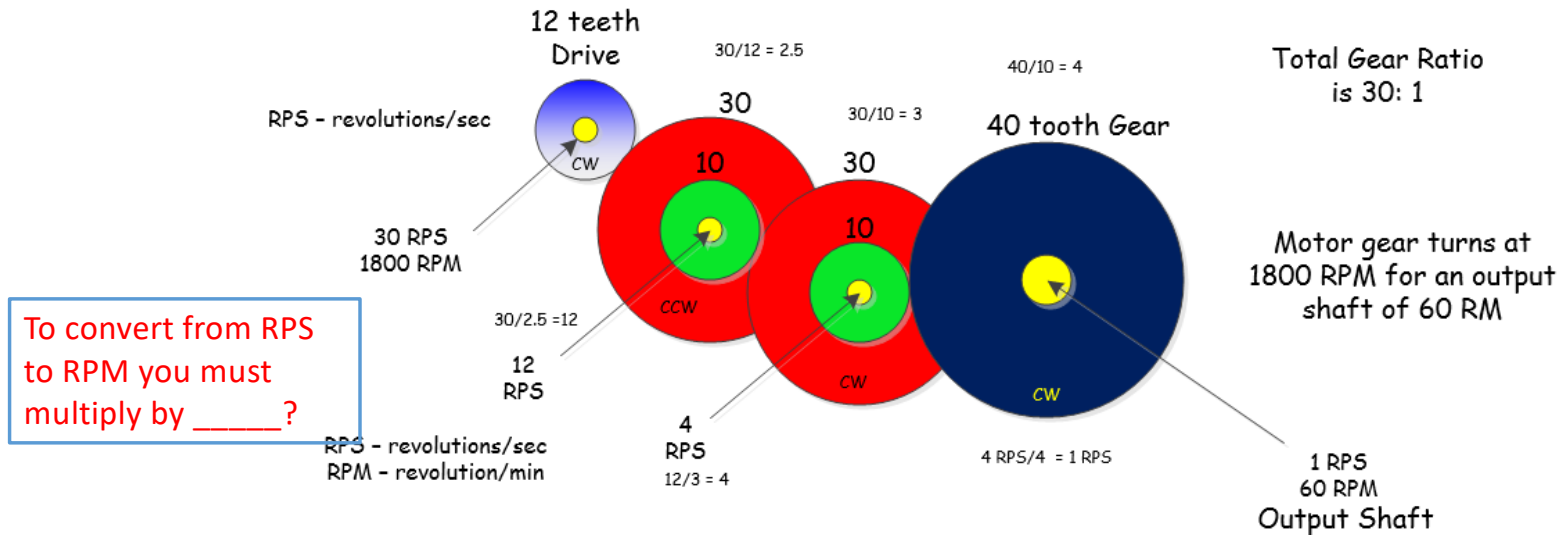




\*\* CHA and B are open collector outputs and require a pull up resistor.

Why are pull-up resistors required?  
 What is a common pull-up value range?

# Compound Gears and Drive Trains

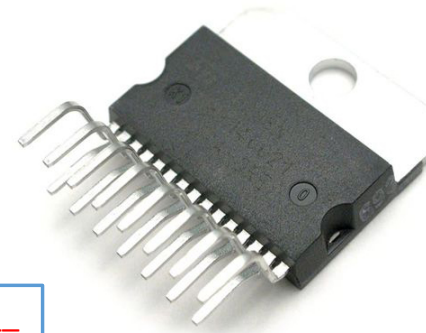


The gear set to the left is from the motor used in the lab. At 12 VDC the output shaft rotates at 103 RPM with a maximum torque of 0.127 Nm. The motor has a gear ratio of 30:1. If the output shaft is turning at 103 RPMs the motor is rotating at 3090 RPM or 51.5 RPS. The hall sensors will produce pulses at a rate of 154.5 Hz.

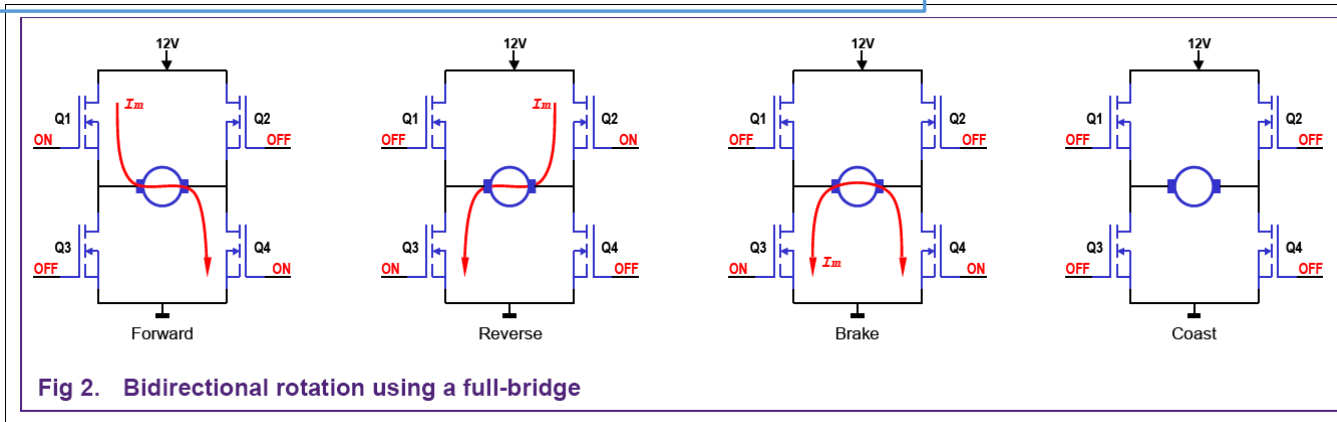
# DC Motor Control H-Bridge

## L293, L298 H-Bridge Operation:

A group of four transistors (MOSFETS) are used to control the direction of a DC motor. The circuit reverses which pins on the motor receive the positive and negative signals. Logic is built into the driver IC to prevent a short circuit.

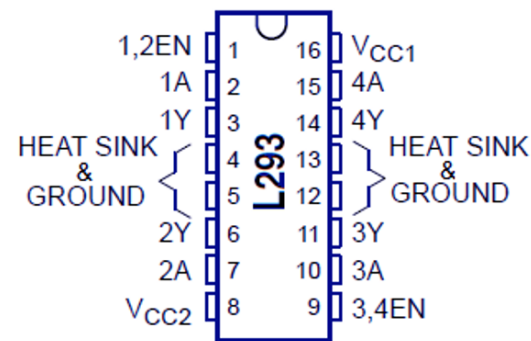


Motor braking occurs when both sides of the motor have the \_\_\_\_\_ voltage applied?



# Motor Controller Function Table

| EN (EIO0) Pin 1 | IN1 (EIO1) Pin 2 | IN2 (EIO2) Pin 7 | Motor Function            |
|-----------------|------------------|------------------|---------------------------|
|                 |                  |                  |                           |
| L               | X                | X                | Free Run Motor Stop       |
| H               | L                | L                | Fast Motor Stop (Braking) |
| H               | L                | H                | Clockwise                 |
| H               | H                | L                | Counter-ClockWise         |
| H               | H                | H                | Fast Motor Stop (Braking) |

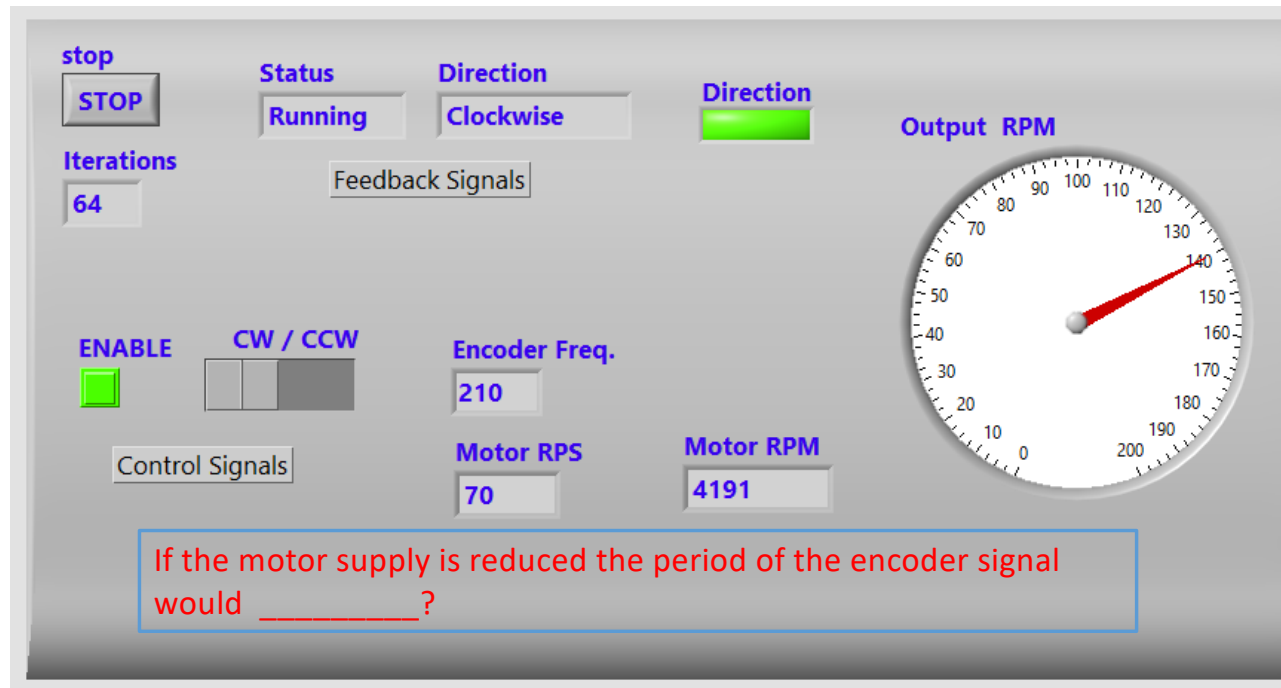


L293 pinout and function diagram

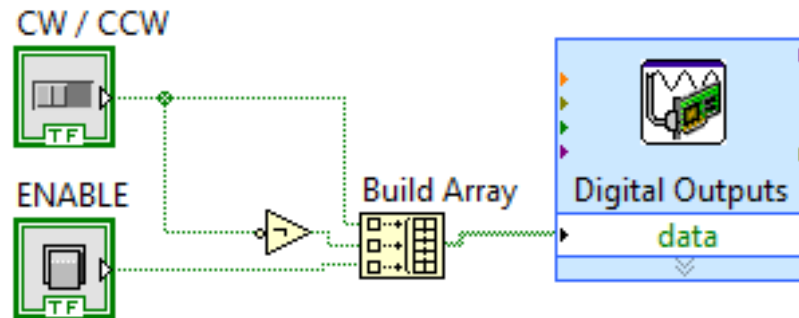
Motor braking only occurs when EN is a logic \_\_\_\_\_ and IN1 and IN2 are at the \_\_\_\_\_ logic level?

The three control pins determine the motor direction, clockwise or counter-clockwise. The control signals also control whether the motor brakes or free runs to a stop. If the motor control IC is enabled and both inputs are at the same level the motor will stop quickly (brake). If the IC is disabled the motor free runs to a stop.

# myDAQ Motor Speed LabVIEW Front Panel



## L293 Control Signals



This section of the VI controls the direction of the motor and enables and disables the driver.

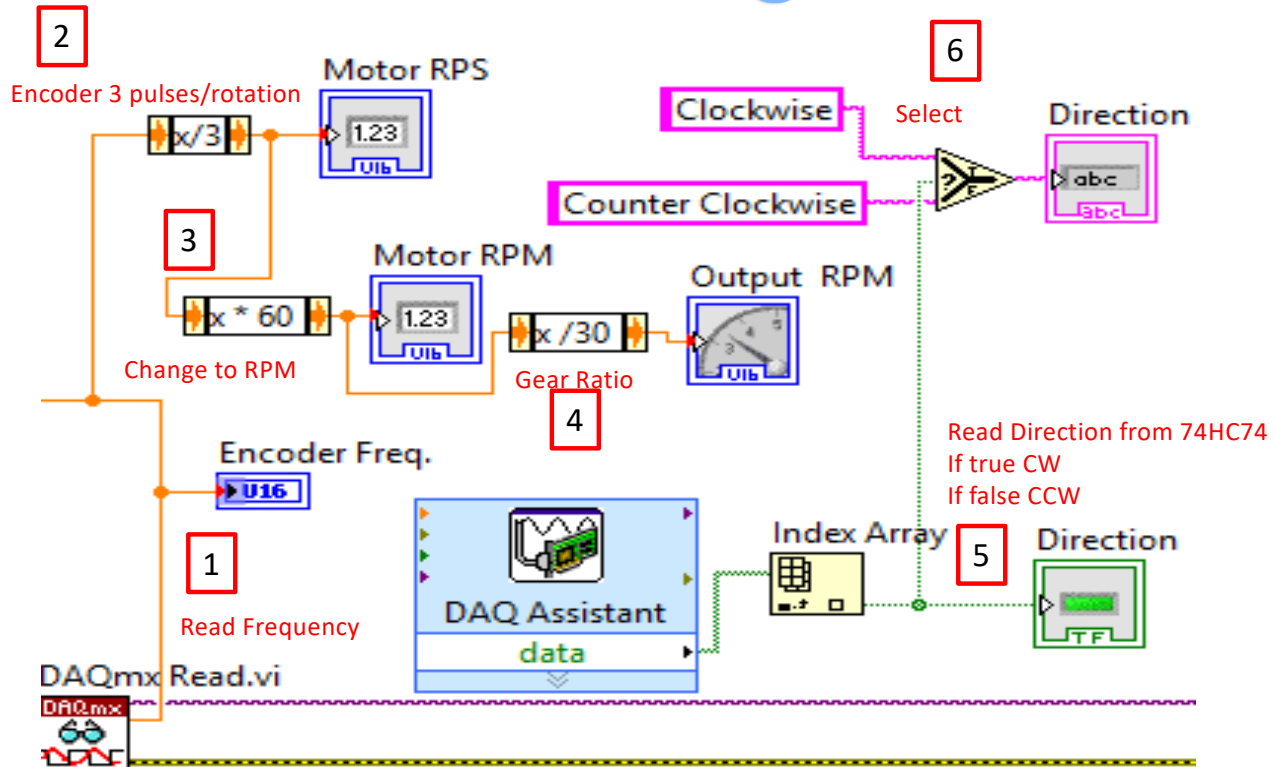
The signals connect to IN1 and IN2 on the L293 motor controller.

### Channel Settings

|                | Order | Physical Channel   |
|----------------|-------|--------------------|
| Digital Output |       |                    |
| In1            | 0     | myDAQ1/port0/line5 |
| in2            | 1     | myDAQ1/port0/line6 |
| EN             | 2     | myDAQ1/port0/line7 |

| EN (EIO0) Pin 1 | IN1 (EIO1) Pin 2 | IN2 (EIO2) Pin 7 | Motor Function            |
|-----------------|------------------|------------------|---------------------------|
| L               | X                | X                | Free Run Motor Stop       |
| H               | L                | L                | Fast Motor Stop (Braking) |
| H               | L                | H                | Clockwise                 |
| H               | H                | L                | Counter-ClockWise         |
| H               | H                | H                | Fast Motor Stop (Braking) |

# Block Diagram



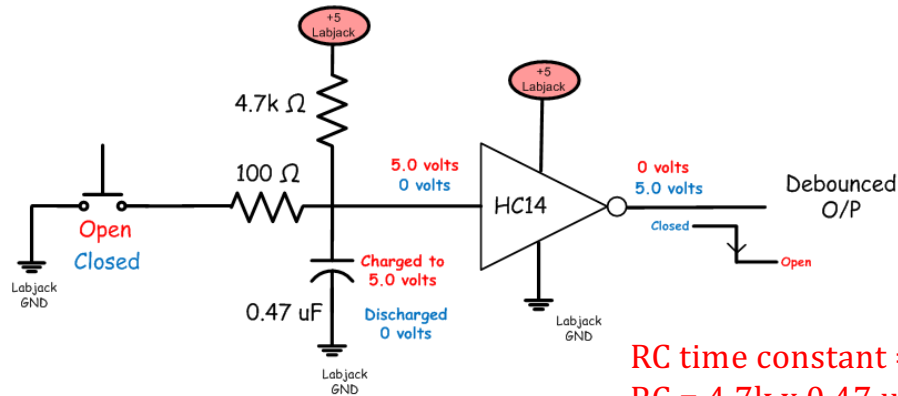
The output of the DAQmxRead.vi is the \_\_\_\_\_?

# Mechanical Switch Debouncing



How long will it take to fully charge the capacitor when the switch is opened?

When the PB is pressed and then released a \_\_\_\_\_ edge is created at the HC14 output?



When the P.B. switch closes the capacitor discharges quickly through the 100 ohm resistor the cap and then to ground.

When the switch opens the capacitor charges slowly through the 4.7 kΩ resistor.

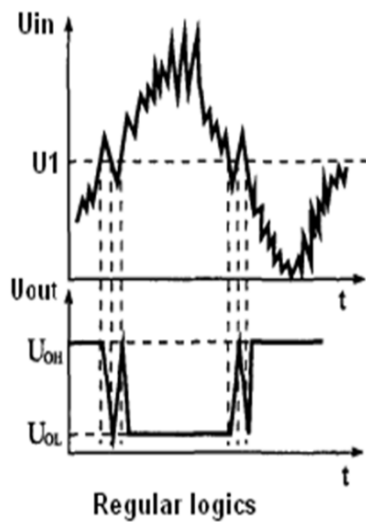
While the switch is bouncing it will take much longer to charge than it will to discharge.

The bouncing voltage tends to occur between 0 and a voltage which is less than the upper threshold or from above the lower threshold to 5.0 volts.

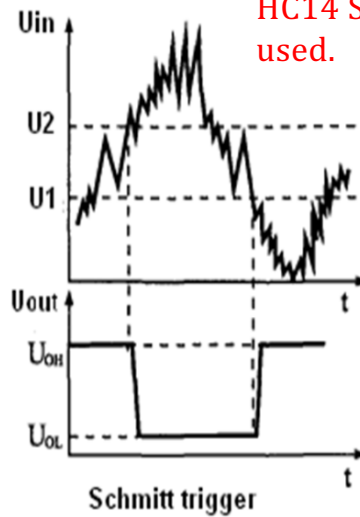
# Schmitt Trigger Operation

An HC14 has  
 \_\_\_\_\_  
 switching points  
 and \_\_\_\_\_ and  
 a \_\_\_\_\_?

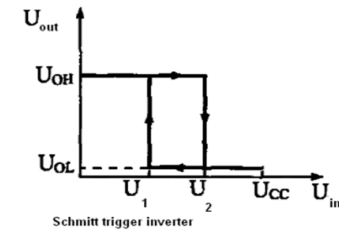
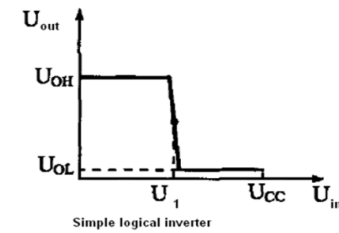
When de-bouncing or wave shaping an  
 HC14 Schmitt trigger inverter must be  
 used.



HC04 (bad)  
 Multiple edges.



HC14 (good)  
 Single edges.



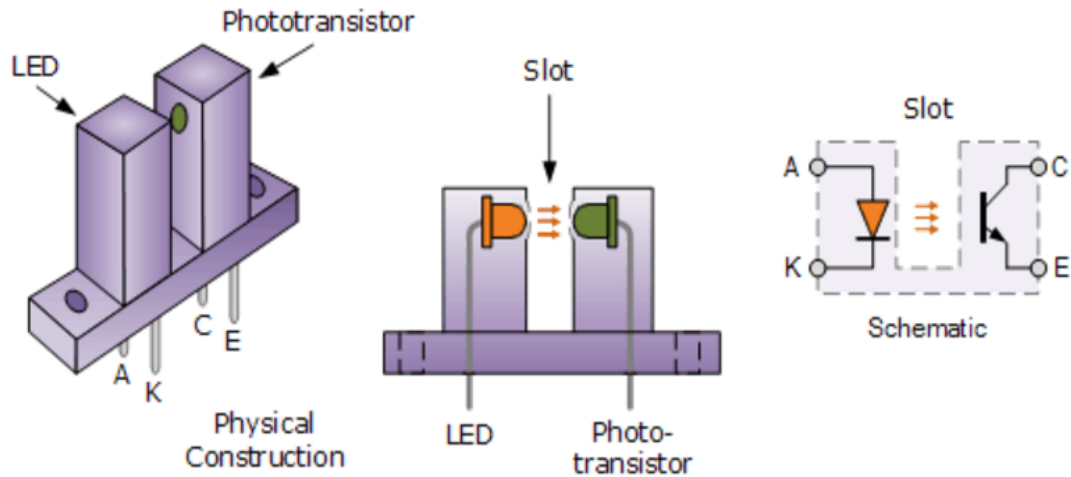
The Schmitt trigger converts the noisy  
 or slow changing signals to vertical  
 single edges.

# Slotted Optical Switch



## Slotted Optical Switch

Most optical sensors use \_\_\_\_\_ leds?

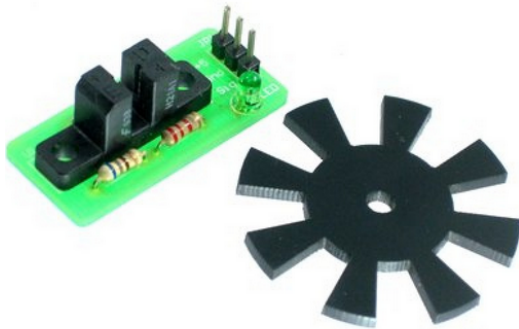


A DC voltage is generally used to drive a light emitting diode (LED) which converts the input signal into infrared light energy. This light is reflected and collected by the phototransistor on the other side of the isolation gap and converted back into an output signal.

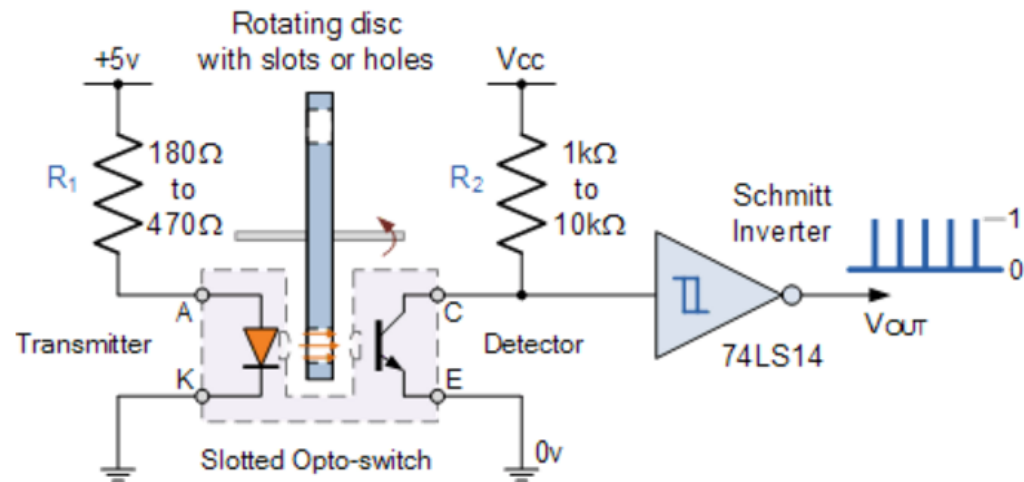
For normal opto-switches, the forward voltage drop of the LED is about 1.2 to 1.6 volts at a normal input current of 5 to 20 milliamperes. This gives a series resistor value of between 180 and 470 $\Omega$ 's.

# Slotted Optical Switch

When the slot is blocked and then unblocked a \_\_\_\_\_ edge is created?



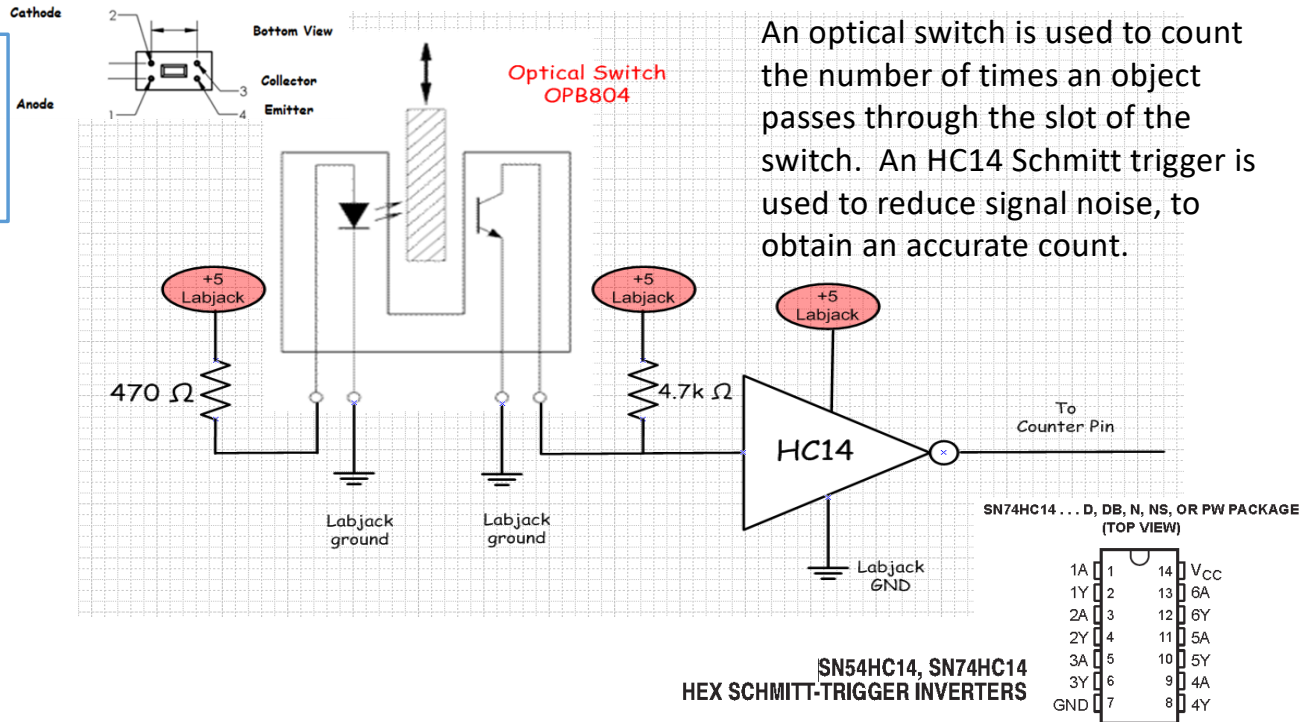
## Slotted Opto-switch Circuit



# Optical Switch Conditioning Using a Schmitt Trigger Inverter

Is the IR LED  
always ON?

Yes/No



An optical switch is used to count the number of times an object passes through the slot of the switch. An HC14 Schmitt trigger is used to reduce signal noise, to obtain an accurate count.

## DAQ Assistant Pulse Out

Configuration    Triggering    Advanced Timing

Channel Settings

+    X       Details >>    ^

Pulse Output

PulseOutput

Click the Add Channels button (+) to add more channels to the task.

Pulse Output Setup

Settings

Signal Output Range

High Time    1m    Scaled Units

Low Time    9m    Seconds

Idle State    Initial Delay (s)

Low    0

Output Terminal:

PFI3

Timing Settings

Generation Mode    Samples to Write    Rate (Hz)

Continuous Pulses    100    1k

Pulse output can be configured to produce a time low or time high of any given value. The pulse out can be used to PWM (pulse width modulate ) loads such as heater motors and lamps.

This configuration will produce a signal with a frequency of \_\_\_\_\_ ms. and it will last for \_\_\_\_\_ second.

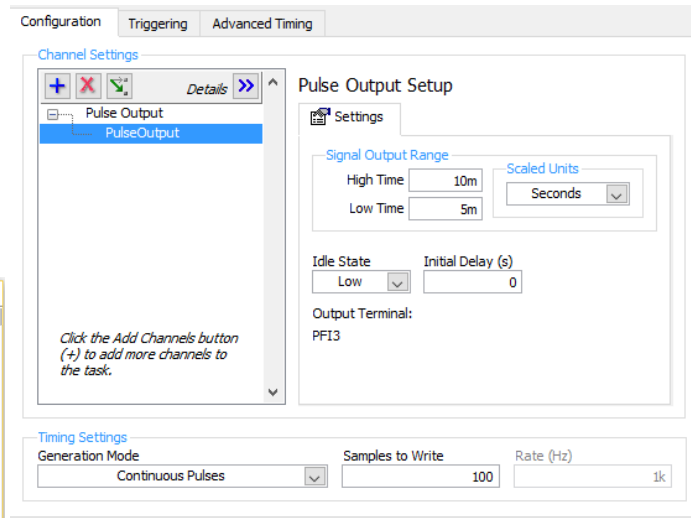
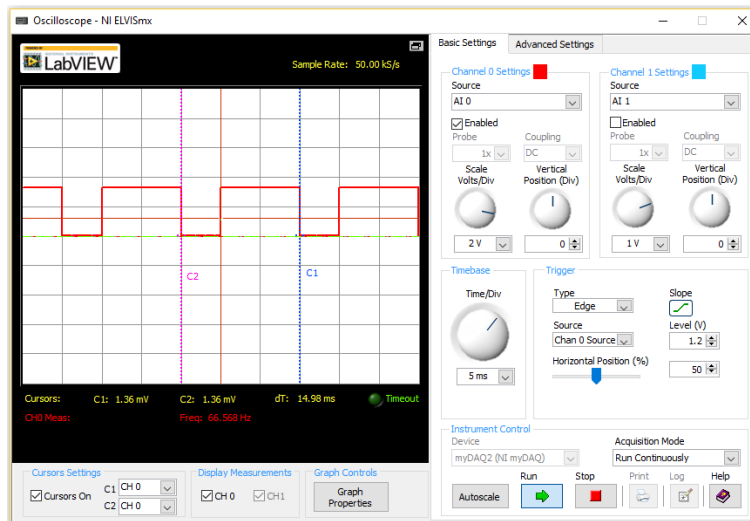
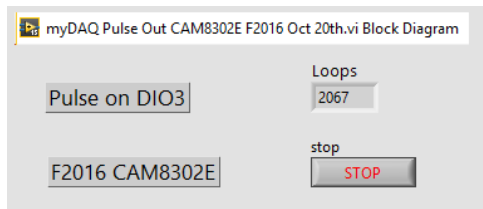
The output signal is created on terminal DIO3.

The %DC = \_\_\_\_\_

The signals or LVTTTL.

# DAQ Assistant Pulse Out

Volts/div = 2, time/div = 5 ms (myDAQ scope)

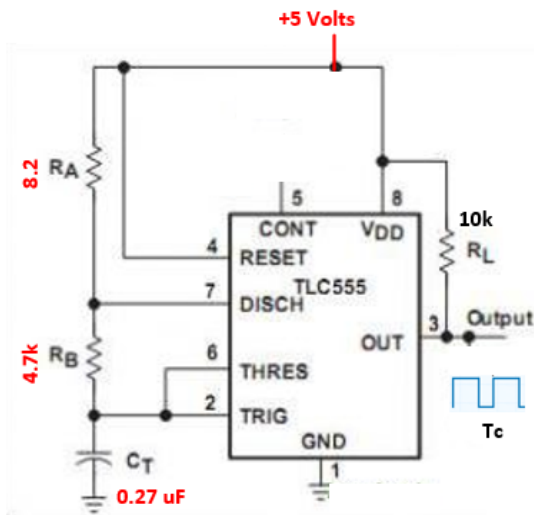


Frequency is 66.7 Hz.

Time high = 10 ms

Time Low = 5 ms.

# TLC555 CMOS Astable Timer

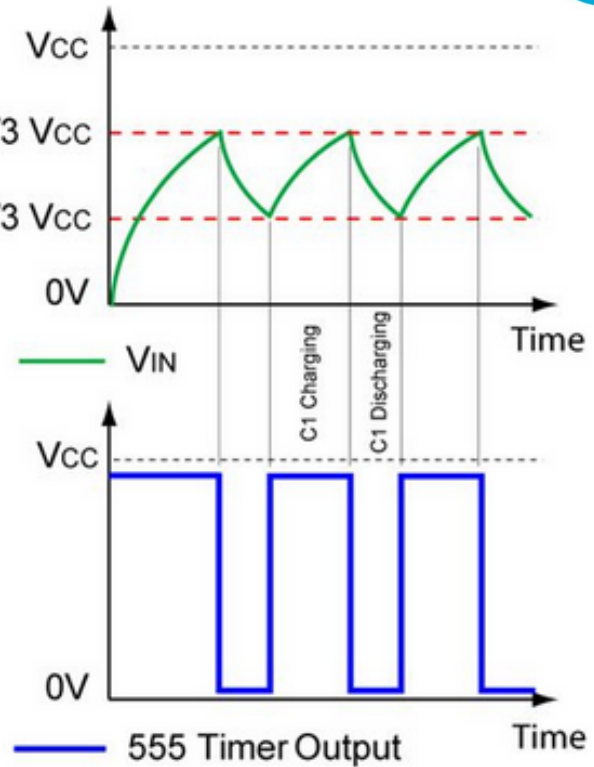


$$t_{c(H)} = C_T (R_A + R_B) \ln 2 \quad (\ln 2 = 0.693)$$

$$t_{c(L)} = C_T R_B \ln 2$$

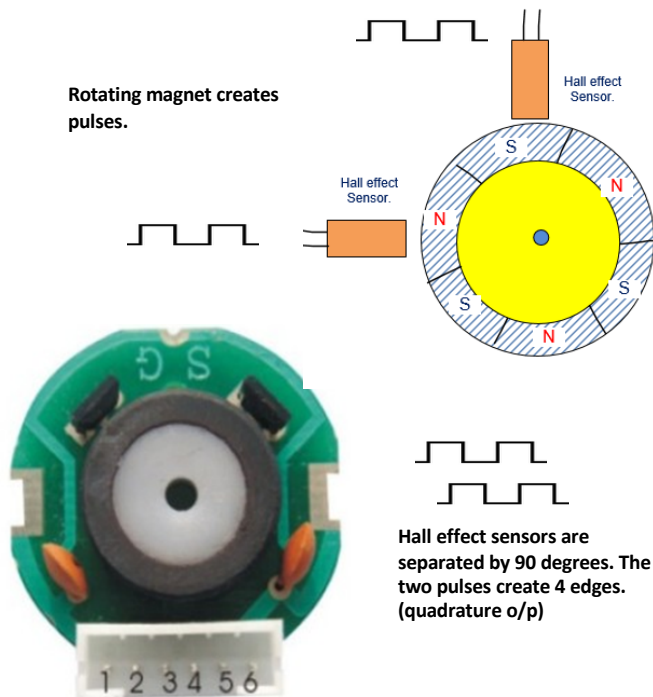
Using the given values determine:

- Period
- Frequency
- Time high
- Time low
- % duty cycle



The capacitor charges through  $R_A + R_B$  and discharges through  $R_B$  only.

# Direction Detection Using Encoder



Changing the gear ratio to 40:1 would decrease the \_\_\_\_\_?

# 74HC74 – D-Type Flip Flop

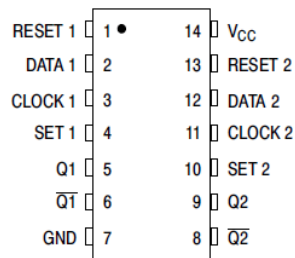


This device changes the output on a rising edge of the clock?

T/F?

## 74HC74

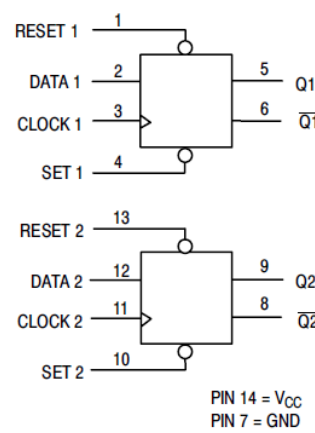
### PIN ASSIGNMENT



### FUNCTION TABLE

| Inputs |       |       |      | Outputs   |           |
|--------|-------|-------|------|-----------|-----------|
| Set    | Reset | Clock | Data | Q         | Q̄        |
| L      | H     | X     | X    | H         | L         |
| H      | L     | X     | X    | L         | H         |
| L      | L     | X     | X    | H*        | H*        |
| H      | H     | ↗     | H    | H         | L         |
| H      | H     | ↘     | L    | L         | H         |
| H      | H     | L     | X    | No Change | No Change |
| H      | H     | H     | X    | No Change | No Change |
| H      | H     | ↔     | X    | No Change | No Change |

### LOGIC DIAGRAM



One phase of the encoder will attach to the clock input the other phase to the data input.

When turning in one direction the data will be low when the clock goes through a rising edge.

In the opposite direction the data will be high when the clock goes through a positive edge,

## DC Motor Types

Brushed DC



### **Advantages**

- Cheapest and simplest motor
- Speed linear to applied voltage
- Simple motor control

### **Disadvantages**

- High maintenance
- Low life-span (due to physical wear on brushes)

Brushless DC



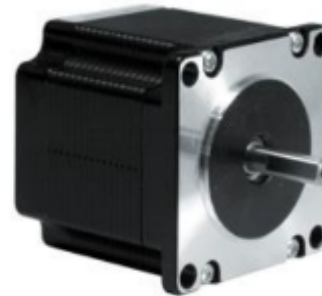
### **Advantages**

- High efficiency
- Little to no maintenance
- Long life span
- High output power per frame size

### **Disadvantages**

- More complicated motor control
- Large initial costs

Stepper



### **Advantages**

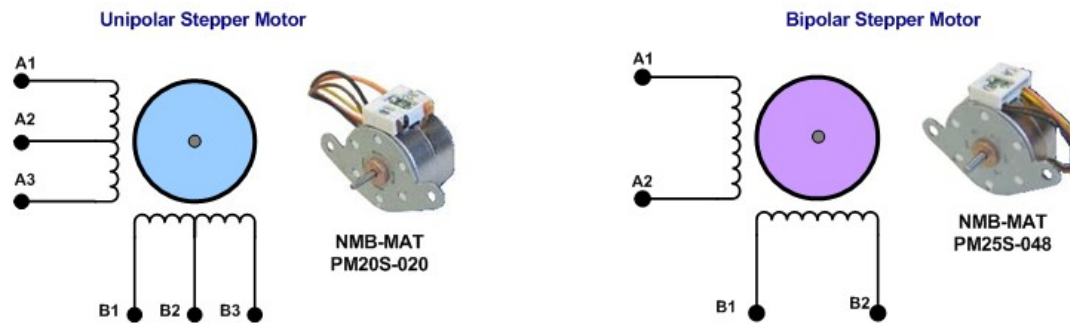
- Accurate position control
- Excellent low speed torque
- Long life

### **Disadvantages**

- Low efficiency
- Prone to resonances, noise, and torque ripple
- Cannot accelerate loads rapidly

Brushless motors are more \_\_\_\_\_ than brushed motors.

# Bipolar/Unipolar stepper motors.



The Unipolar and Bipolar Stepper Motor Windings

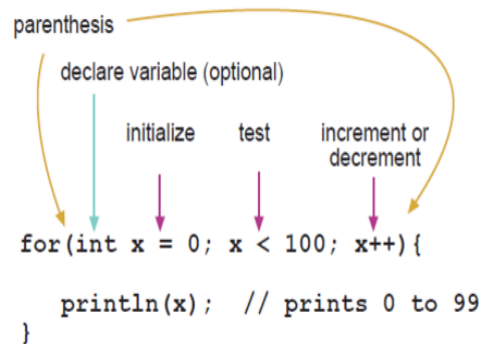
Unipolar stepper motors require a simpler drive circuit, bipolar stepper motors have higher torque capabilities.

Unipolar – current flow in only one direction

Bipolar – current flow in two directions

Bipolar motors always have \_\_\_\_\_ wires. Unipolar motors can have \_\_\_\_\_ or \_\_\_\_\_?

## For Loop – Used in Stepper Program



The diagram shows a for loop with four annotations: 'parenthesis' with a curved arrow pointing to the entire loop header; 'declare variable (optional)' with a blue arrow pointing to 'int x = 0'; 'initialize' with a purple arrow pointing to 'x = 0'; 'test' with a purple arrow pointing to 'x < 100'; and 'increment or decrement' with a purple arrow pointing to 'x++'. A yellow arrow also points from the 'increment or decrement' annotation back to the 'test' annotation, indicating the loop's flow.

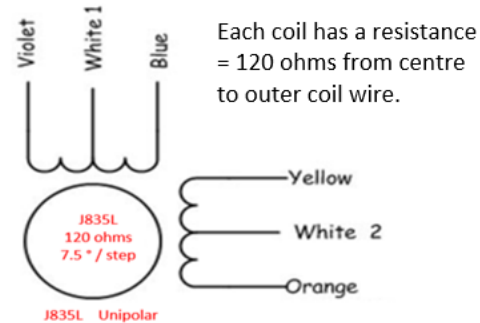
```
for(int x = 0; x < 100; x++){  
    println(x); // prints 0 to 99  
}
```

The statement in this loop is  
executed \_\_\_\_\_ times?

The initialization happens first and exactly once. Each time through the loop, the condition is tested; if it's true, the statement block, and the increment is executed, then the condition is tested again. When the condition becomes false, the loop ends.

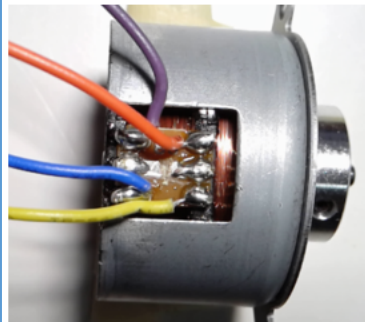
# Unipolar to Bipolar Stepper Modification Motor

Original Unipolar stepper – the white wire is cut off and the PCB trace has been cut.

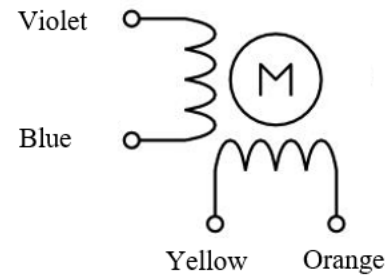


Assume a bipolar motor. With a 10 volts supply and two coils energized the power supply will draw \_\_\_\_\_ mA while the motor is stopped.

While the motor is turning the current will \_\_\_\_\_.

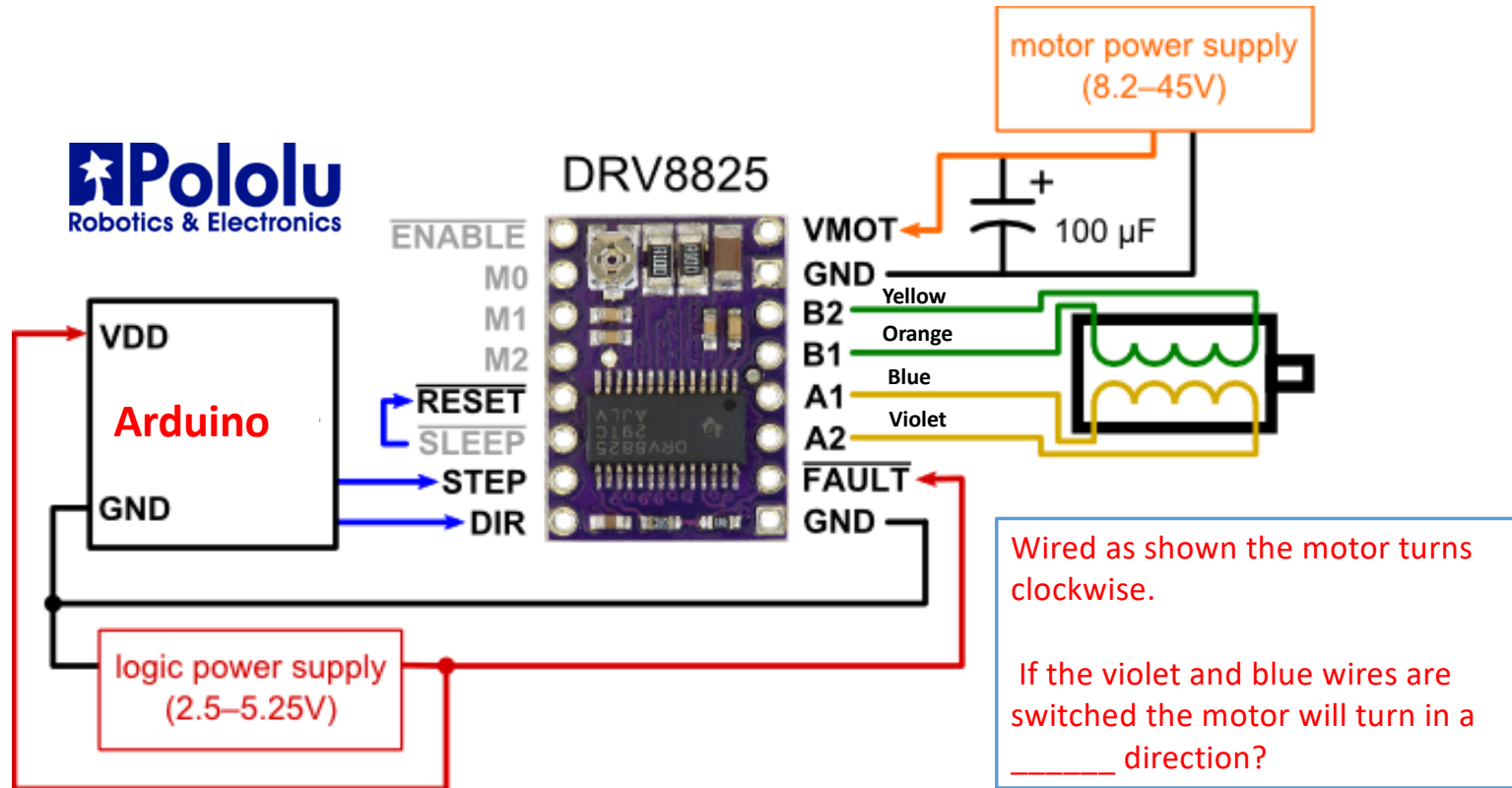


Modified Unipolar --> Bipolar



Modified -- each coil has a resistance = 240 ohms.

## Bipolar DC Stepper Motor Controller

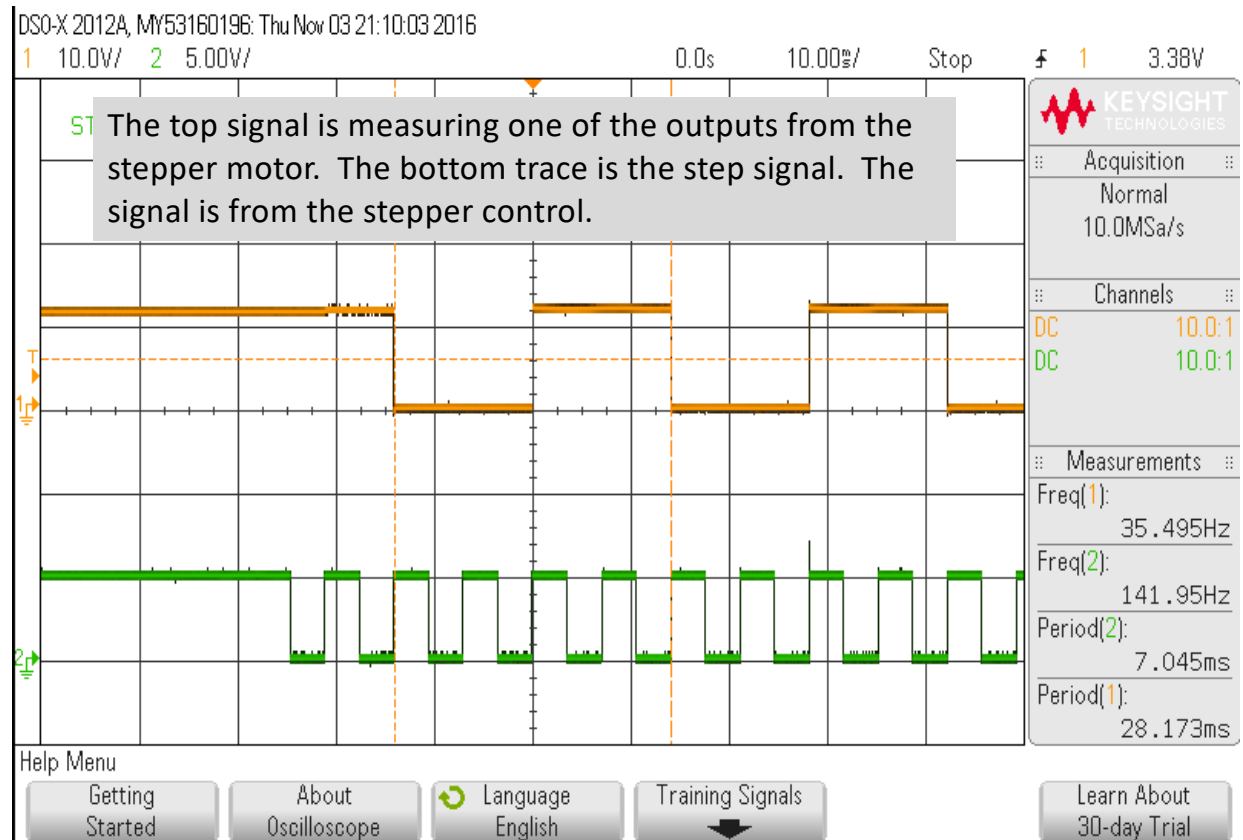


The frequency of the coil signal is \_\_\_\_\_ less than the step signal?

The frequency of the step signal is \_\_\_\_\_ more than the coil signal?

The stepper motor advance by its step angle on the \_\_\_\_\_ edge of the step signal?

If the step frequency = 4 kHz the coil frequency will equal \_\_\_\_\_ kHz?



### Stepper\_FWD\_REV\_DRV8825\_F2016

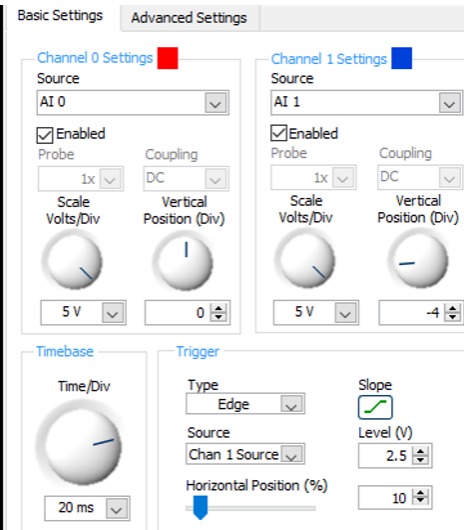
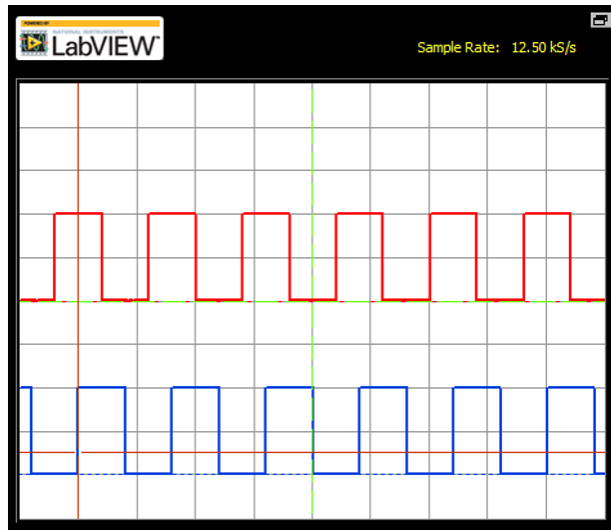
```
1 /* Stepper Motor Control using a
2 * Pololu DRV8825 Controller
3 * October 31st 2016
4 * Michel Hanbury from CAM8302E
5 */
6 void setup() {
7   pinMode(2, OUTPUT); // step pulse
8   pinMode(3, OUTPUT); // direction
9 }
10
11 void loop()
12 {
13   digitalWrite(3, LOW); //direction
14
15   for (int x=0; x < 200; x++)
16   {
17     digitalWrite(2, LOW);
18     delayMicroseconds(400);
19     delay(2);
20
```

Assume a bipolar stepper with a step angle of 7.5 degrees/step?

```
21   digitalWrite(2, HIGH);
22   delayMicroseconds(400);
23   delay(2);
24 }
25 delay(1000);
26
27
28   digitalWrite(3, LOW); //direction
29
30   for (int x=0; x < 200; x++)
31   {
32     digitalWrite(2, LOW);
33     delayMicroseconds(400);
34     delay(2);
35
36     digitalWrite(2, HIGH);
37     delayMicroseconds(400);
38     delay(2);
39   }
40   delay(1000);
41 }
42
```

How long with this motor turn before changing direction?

How many revolutions are made in each direction?



The scope is triggered on channel \_\_\_\_?

The scope is triggered on a \_\_\_\_\_ edge?

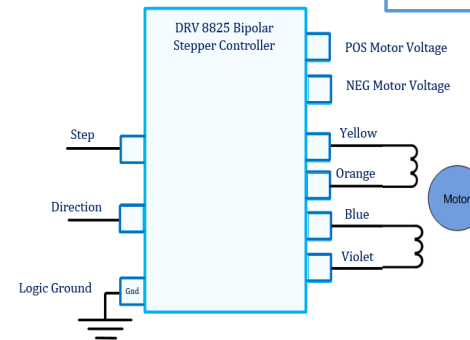
The scope is triggered at the \_\_\_\_\_ horizontal position?

Channel 1 vertical position = \_\_\_\_\_ and the ground is \_\_\_\_\_ division higher than the base?

The trigger level = \_\_\_\_ volts.

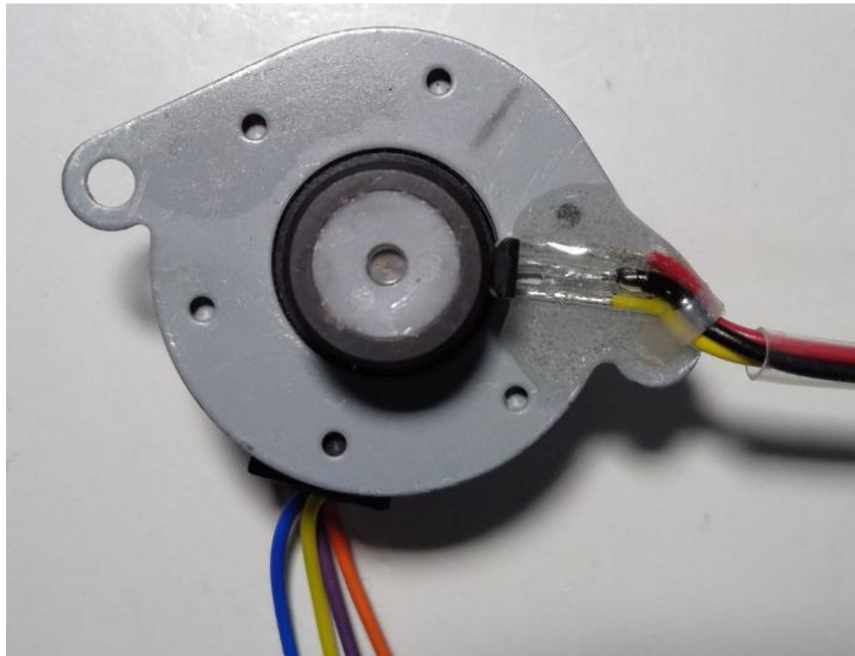
The two traces above are measuring the signal on the driver outputs connected to the yellow wire and the blue wire.

The two signals are 90 degrees out of phase. When the motor is turning in one direction the signal is leading by 90 degrees, in the opposite direction the signal is lagging by 90 degrees to the other coil.



Modified Stepper motor with Hall Effect Sensor and magnet.

If the motor is turning at 150 RPM the period of the hall effect sensor o/p will equal \_\_\_\_\_?



240 ohms / coils

7.5 degrees / step

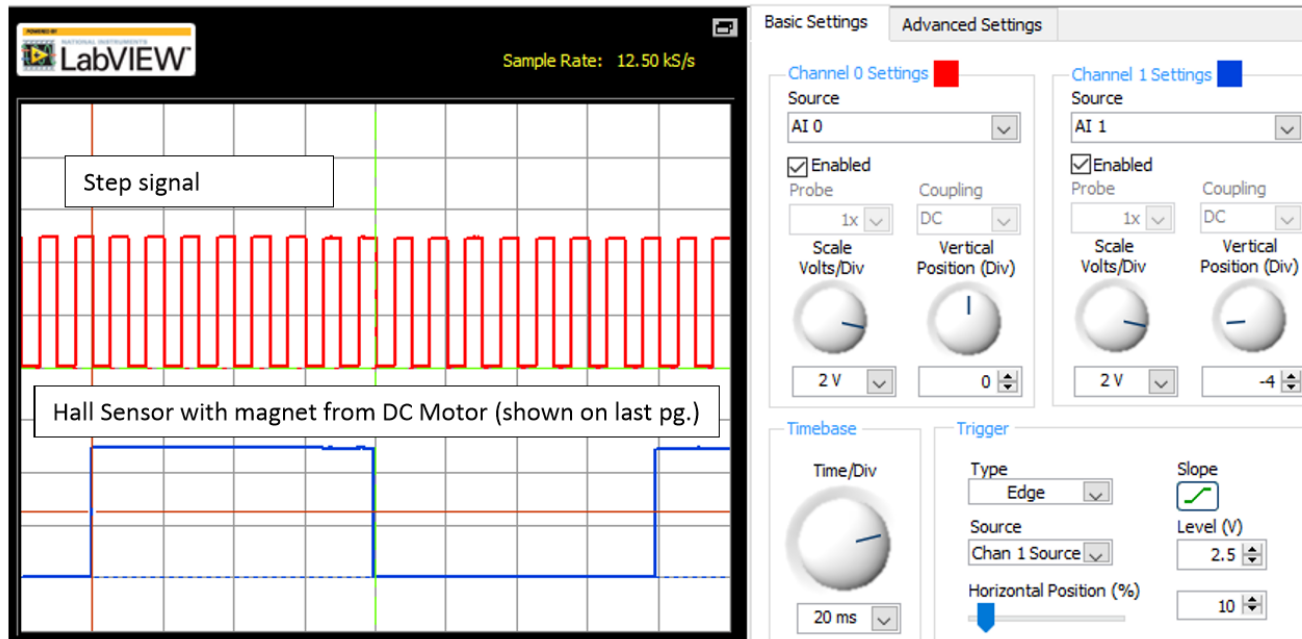
Bi-polar

A Hall Effect sensor is attached near the magnet.

The magnet is the same one used on the DC motor with three NS magnets that will generate 3 pulses per revolution.

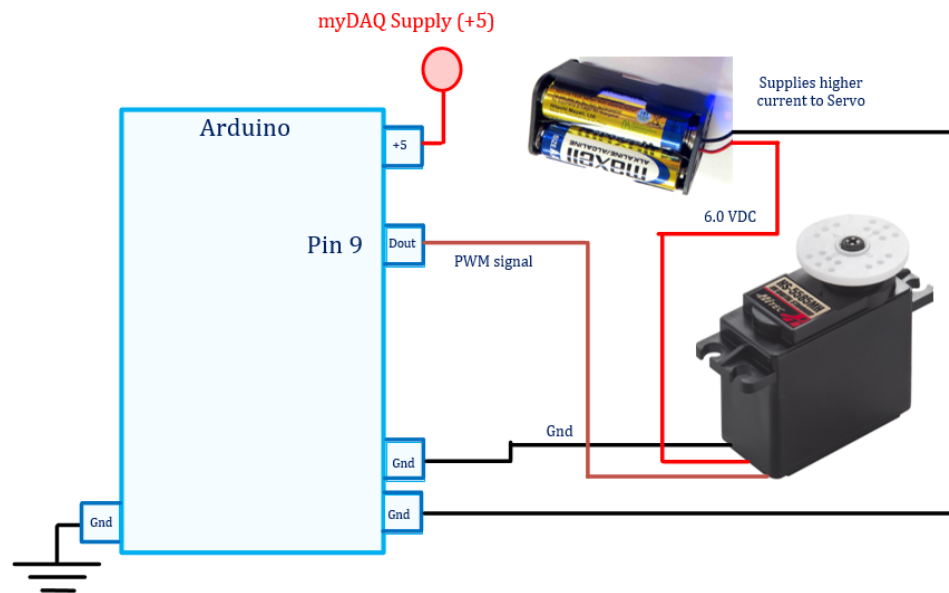
The Hall sensor requires a 5 volt supply and a 10 k pull up resistor attached to the open collector output of the Hall sensor.

The Hall Effect sensor, and the magnet were added to measure the RPMs of the stepper motor. The magnet has three N/S poles that generate three pulses per revolution.



The motor takes 480 ms. per revolution,  $1/480 \text{ ms.} = 2.08$  revolutions per second or about 120 RPM.

What is the voltage level of the step signal and the coil signal with respect to ground?



o hardware  
figuration.

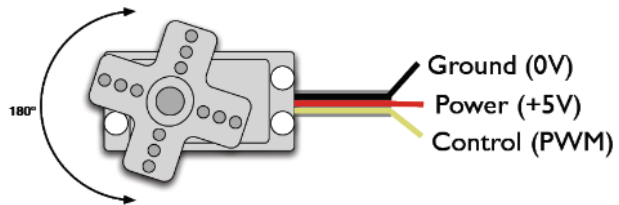
The period of the servo  
motor PWM control  
signal = \_\_\_\_ ms.

The position of the shaft  
is changed by altering the  
time \_\_\_\_ of the PWM  
signal from 1 to 2 ms.

**Servo Motor Connections to Arduino:** Connect the power for the servo to a 4.5 to 6.5 volt battery or DC power supply. The signal controlling the servo motor shaft position is controlled using an Arduino PWM output.

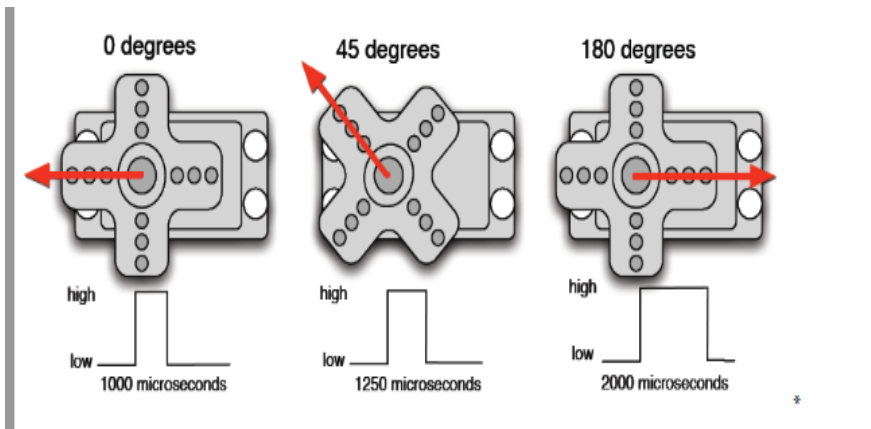
### Make the servo movement

Of course you need to connect the servo motor. Now get your servo motor, and you can find three pin hole. They are:



As we know, we are going to send a pulse to make the servo move. The pulse should be ranges from 1 to 2 milliseconds.

At 1.5 ms. high the servo output shaft is in the \_\_\_\_\_ position?

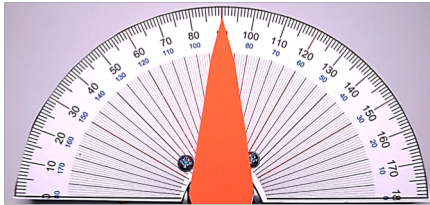


servo\_motor\_test\_with\_slider\_Nov\_11\_2016

```
1 // Controlling a servo position using a potentiometer (variable resistor)
2 // by Michal Rinott <http://people.interaction-ivrea.it/m.rinott>
3 // Modified by Michel Hanbury
4 // For CAM8302E Nov. 12, 2016
5
6 #include <Servo.h> // add servo library
7
8 Servo myservo; // create servo object to control a servo
9
10 int slider = 0; // analog pin used to connect the potentiometer
11 int Analog_In; // variable to read the value from the analog pin
12 int Servo_Angle;
13 void setup()
14 {
15   Serial.begin(9600); // 9600 bits/second to display variable data
16   myservo.attach(9); // attaches the servo on pin 9 to the servo object
17 }
18
19 void loop()
20 {
21
22   Analog_In = analogRead(slider); // reads the value of the potentiometer (value between 0 and 1023)
23   Servo_Angle = map(Analog_In, 0, 1023, 0, 180); // scale it to use it with the servo (value between 0 and 180)
24   // adjusted for min of 0 deg to 180 deg.
25   myservo.write(Servo_Angle); // sets the servo position according to the scaled value
26
27   Serial.print("Analog In ");
28   Serial.print(Analog_In);
29   Serial.print(" Servo Angle ");
30   Serial.println(Servo_Angle);
31
32   delay(15); // waits for the servo to get there
33 }
```

Servo Motor  
test program.

If A/D 0 = 2.1 volts then Analog\_In =  
\_\_\_\_\_ and Servo\_Angle = \_\_\_\_\_?



Servo motor at the middle neutral position. This is normally about 1.5 milliseconds.

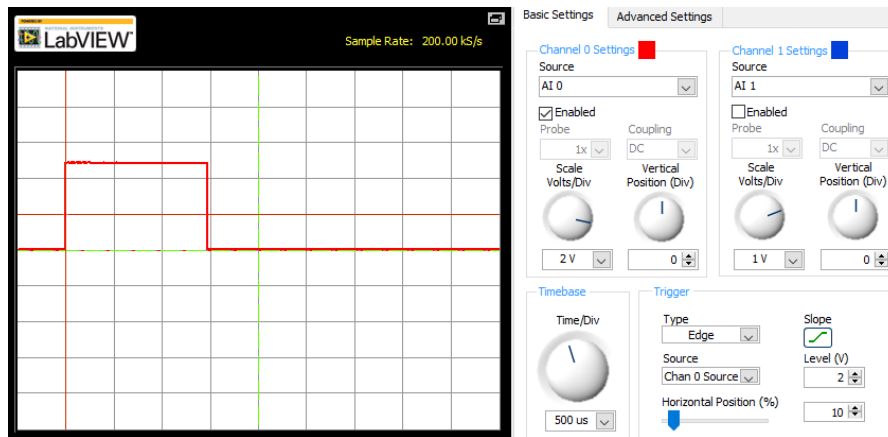
This screen captures shows a time high of  $\sim 1.5$  ms.

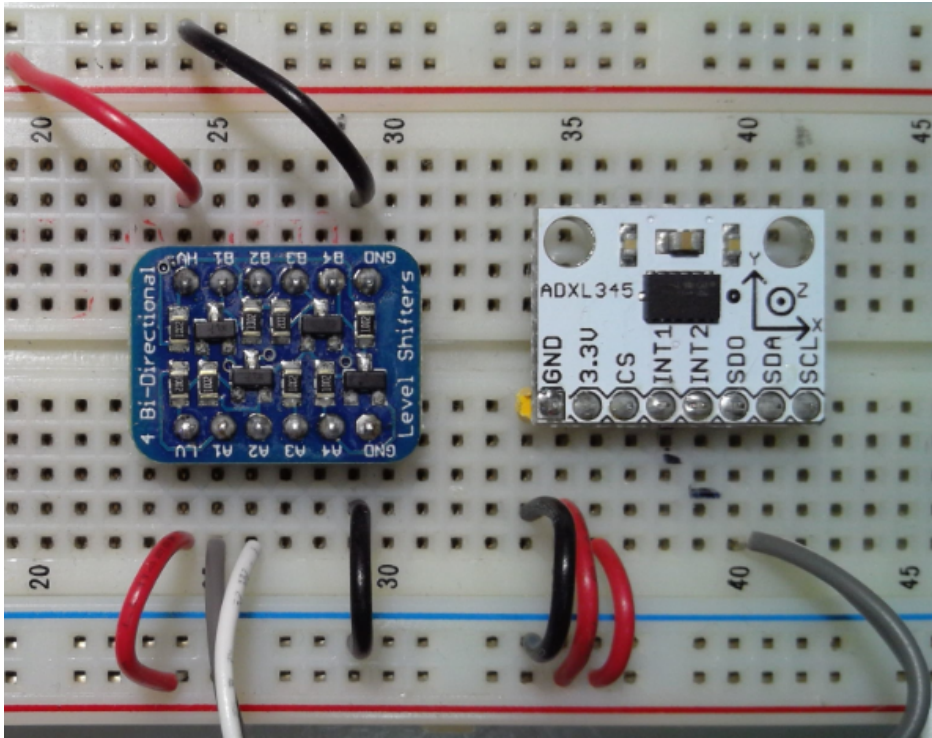
### Program Output

```

Analog In 515   Servo Angle 90
Analog In 515   Servo Angle 90
Analog In 515   Servo Angle 90
  
```

The analog input voltage equals?





This level converter converts \_\_\_\_\_ to \_\_\_\_\_ levels and vice versa.

When interfacing the accelerometer to the Arduino a 3.3 volt to TTL level converter is required. This board converts 3.3 levels to TTL and vice versa.

# Interrupts vs Polling

```
21 void setup() {  
22  
23   attachInterrupt(0, slow, RISING); // on pin 2  
24   attachInterrupt(1, fast, RISING); // on pin 3  
25
```

A rising edge on pin 2 will execute the slow() routine?

T/F?

```
101 void fast(){           // Interrupt service routine  
102   delayTime = 10;  
103 }  
104  
105 void slow(){           // Interrupt service routine  
106   delayTime = 20;  
107 }
```

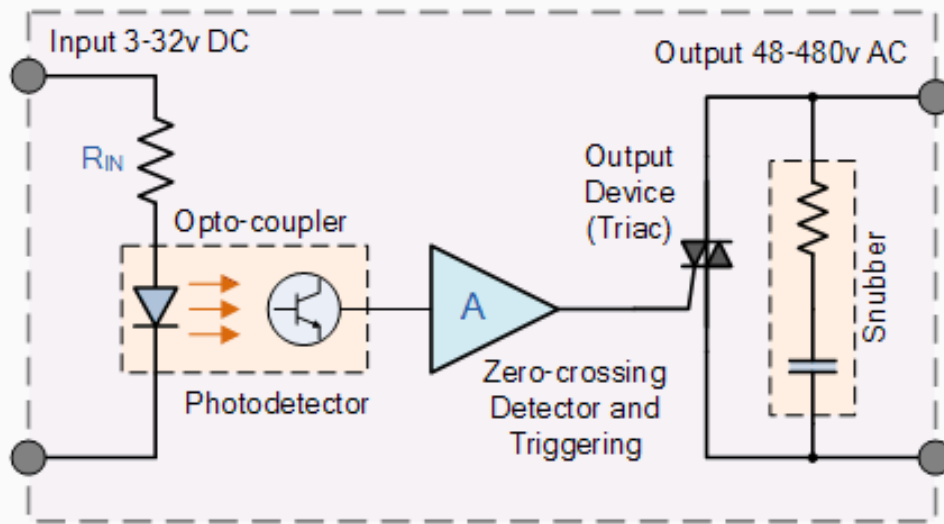
Interrupts can be configured for rising or falling edges?

T/F?

Lines 21-24 configure the interrupt. The program is telling the Arduino that when a rising edge occurs on pin 2 (interrupt 0) then stop what the program is currently doing, execute the “slow” routine then return to where it was before the interrupt.

# Inside a Solid State Relay (OAC5) Black

## Solid State Relay



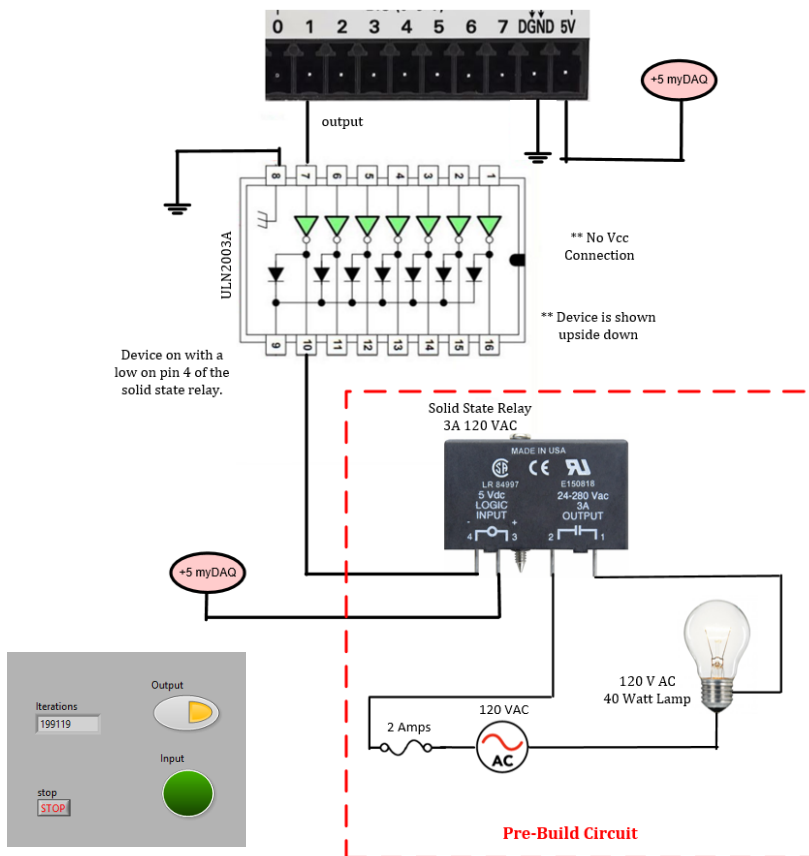
 ElectronicsTutorials

A triac has \_\_\_\_\_ terminals?

The output is \_\_\_\_\_ isolated from the input?

The snubber is used to \_\_\_\_\_ the internal circuit.

The solid state relay includes opto-isolation, a zero crossing detector, a triac and a snubber circuit to suppress transients from inductive loads.



Circuit is required in lab 8.

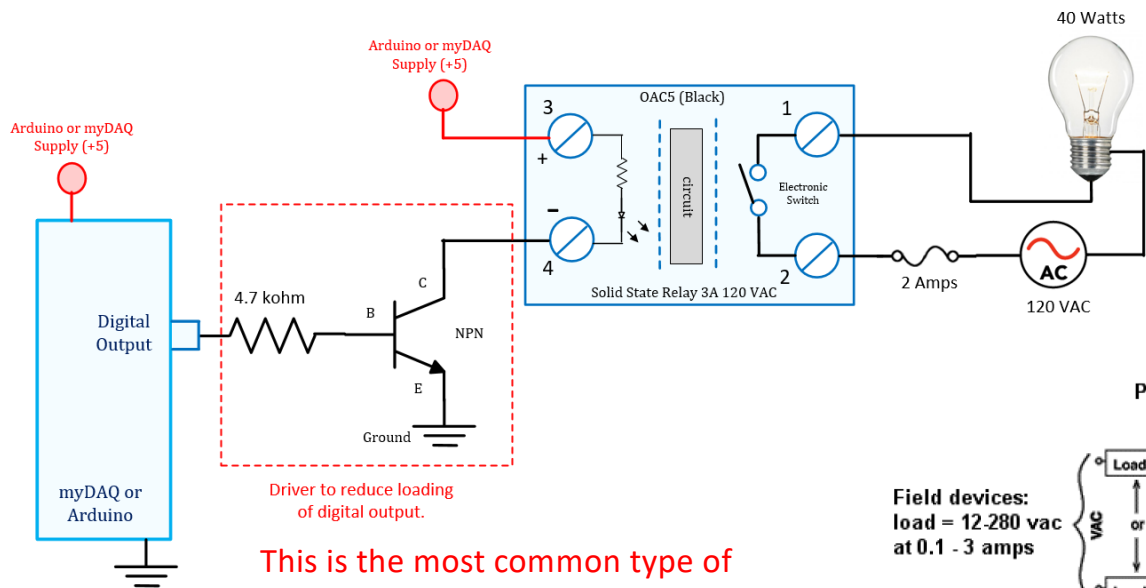
The ULN2003A drivers is added to the digital output on the myDAQ. The specifications for the myDAQ indicate a maximum current of 4 mA. The SSR has a load of about 15 mA when active.

The ULN2003A inputs draw less then 1 mA when controlling the SSR.

What is the voltage on pin 10 when pin 7 is 3.3 volts?

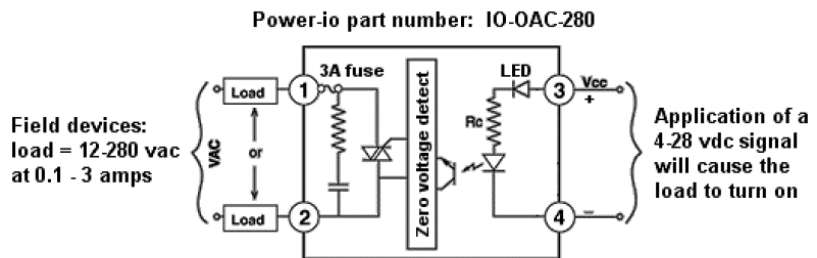
Could this SSR handle a 200 watt lamp at 120 volts? How much current would flow through the SSR?

OAC5 – Controls an AC signal using logic levels. Current flowing through the LED energized a photo transistor. A triac is energized using a zero voltage detection circuit. The zero detection circuits used to energize the device only while the current is low, this helps to reduce spikes when turning the device on or off.

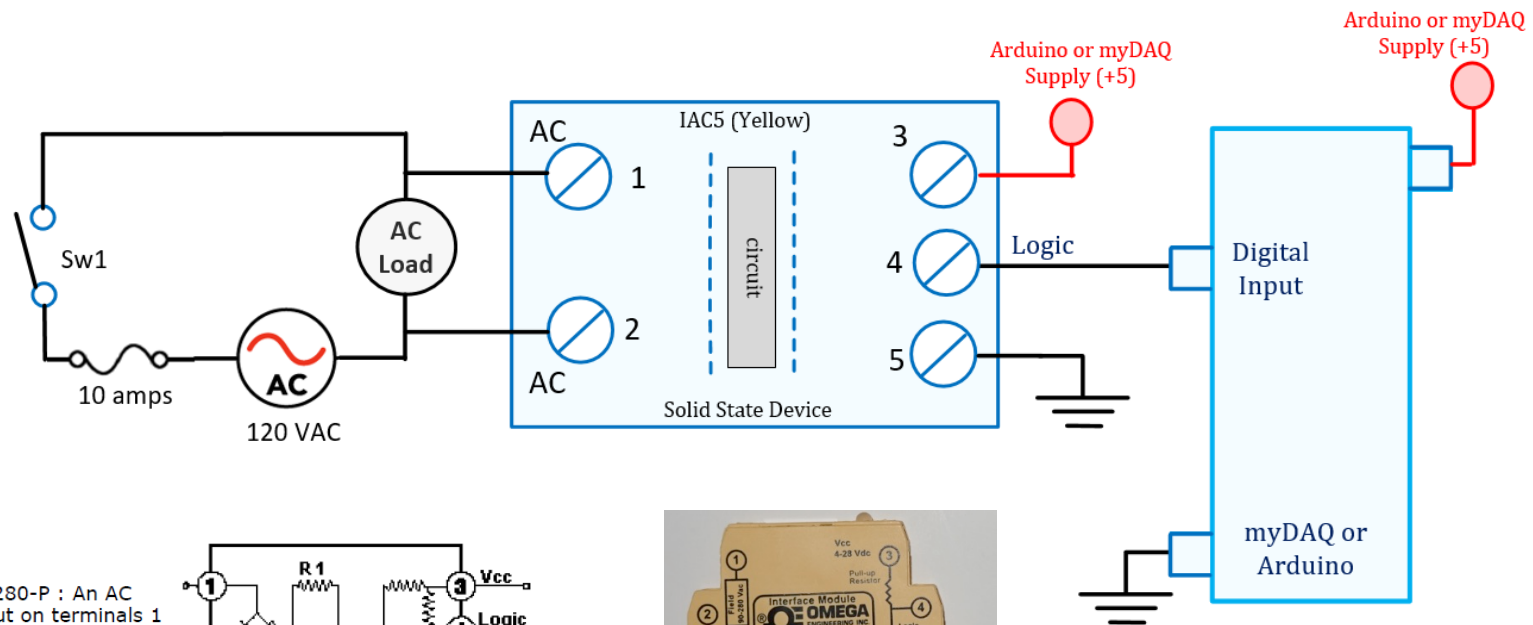


This is the most common type of SSR? T/F?

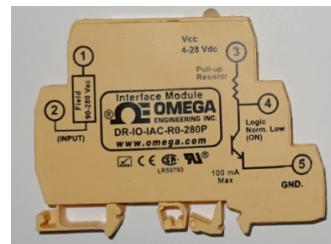
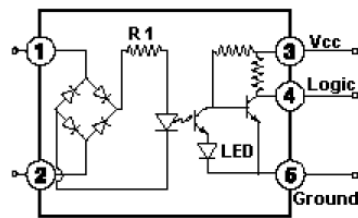
Why is the transistor required on the Arduino?



IAC5 – Measures an AC input voltage and converts it to a logic level. The AC is converted to DC using a bridge rectifier. The DC current activates an led which then turns on an LED and transistor causing the logic level to change.

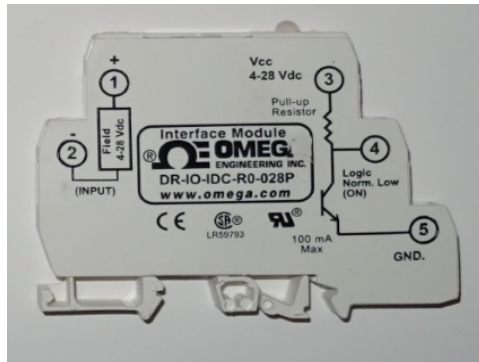


IO-IAC-280-P : An AC field input on terminals 1 and 2 = the voltage of terminal 4 will change from ground (terminal 5) to vcc (terminal 3)

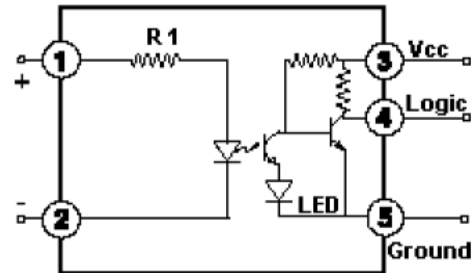


What is the purpose of the bridge rectifier?

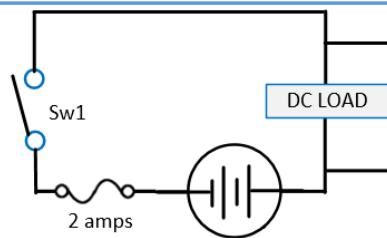
IDC5 – Measures an DC input voltage and converts it to a logic level.



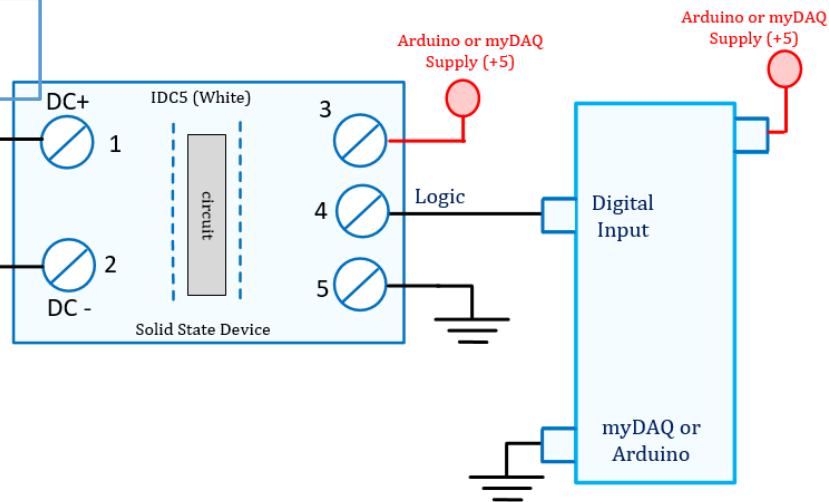
IO-IDC-028-P : A DC field input on terminals 1 and 2 = the voltage of terminal 4 will change from ground (terminal 5) to vcc (terminal 3)



This device can be used to interface an Arduino input to a PLC output? T/F

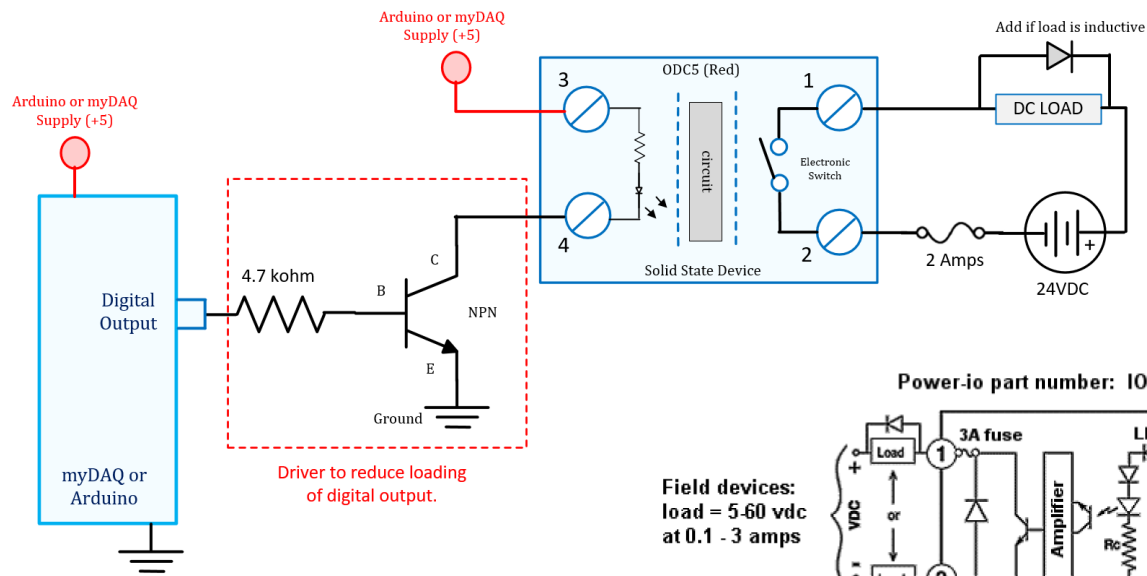


How much power is dissipated by the load when the supply is 24 volts and the load is 40 ohms?

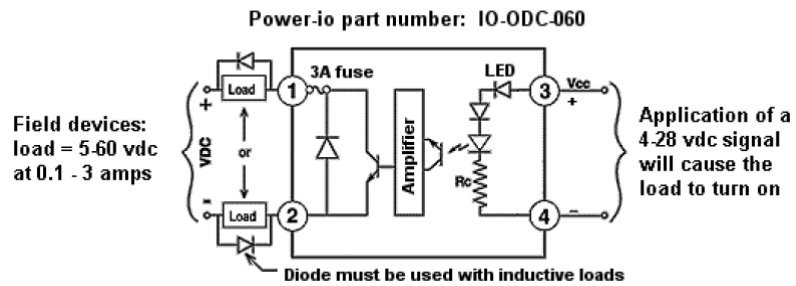
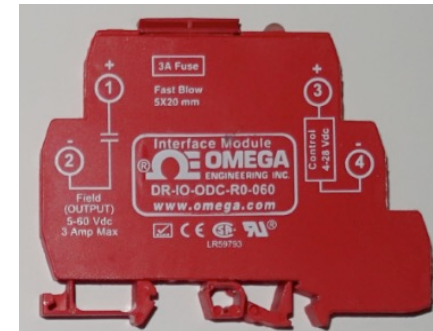


# ODC5 – Controls A DC voltage using logic levels.

This device can be used to interface an Arduino output to a PLC input? T/F



When are diodes required at the output?



# Transducer Specs



| Technical Data                 |  |
|--------------------------------|--|
| <b>Configuration:</b>          |  |
| Configuration                  | DIP switches   |
| <b>Input:</b>                  |  |
| Input signal                   | 0 ... 20 mA, 4 ... 20 mA,<br>0 ... 5 V, 0 ... 10 V, 2 ... 10 V, 1 ... 5 V<br>(calibrated switchable) |
| Input resistance               | $\leq 50 \Omega$ (In = mA)<br>$\geq 100 \text{ k}\Omega$ (In = V)                                    |
| <b>Output:</b>                 |  |
| Output signal                  | 0 ... 20 mA, 4 ... 20 mA,<br>0 ... 5 V, 0 ... 10 V, 2 ... 10 V, 1 ... 5 V<br>(calibrated switchable) |
| Load impedance                 | 600 $\Omega$ (Out = mA)<br>2 k $\Omega$ (Out = V)  |
| <b>General specifications:</b> |  |
| Voltage supply $V_s$           | 24V DC   |
| Supply voltage range           | 16.8 V ... 31.2 V  |
| Current consumption at 24 V DC | $\leq 25$ mA   |
| Max. operating frequency       | 100 Hz / 5 kHz<br>(switchable via DIP switch)  |
| Response time ( $T_{10-90}$ )  | < 3.5 ms / < 100 $\mu$ s   |
| Transmission error             | $\leq 0.1$ % of the full scale value   |
| Temperature coefficient        | $\leq 0.01$ % /K   |
| Zero/span adjustment           | $\pm 3$ % of upper range value   |

A transducer converts one signal form to another.

Examples:

Resistance to voltage: 0-100 ohms to 0- 10 volts.

Voltage to current: 0-10 to 4-20 mA.

Current to Voltage: 4 to 20 mA to 0 to 5 volts.

What power supply voltage is required for this device?

List 5 common instrumentation signals ranges?

| Dip Switch S1 (2-fold) |             | Dip Switch S2 (6-fold) |   |   |   |           | Max. Operating Frequency |   |        |
|------------------------|-------------|------------------------|---|---|---|-----------|--------------------------|---|--------|
| 1                      | 2           | Output Signal          |   |   |   |           | 6                        |   |        |
| •                      | 0 ... 20 mA |                        |   |   |   |           | 0 ... 20 mA              | • | 5 kHz  |
|                        |             |                        |   |   |   | •         | 4 ... 20 mA              |   | 100 Hz |
|                        |             | •                      | • |   |   |           | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   |           | 0 ... 5 V                |   |        |
| •                      | 4 ... 20 mA |                        |   |   |   | •         | 0 ... 20 mA              |   |        |
|                        |             |                        |   |   |   |           | 4 ... 20 mA              |   |        |
|                        |             | •                      | • |   |   | •         | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   | •         | 0 ... 5 V                |   |        |
| •                      | 0 ... 10 V  |                        |   |   |   | •         | 0 ... 20 mA              |   |        |
|                        |             |                        |   |   |   |           | 4 ... 20 mA              |   |        |
|                        |             | •                      | • |   |   |           | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   |           | 0 ... 5 V                |   |        |
| •                      | 2 ... 10 V  |                        |   |   |   | •         | 0 ... 20 mA              |   |        |
|                        |             |                        |   |   |   |           | 4 ... 20 mA              |   |        |
|                        |             | •                      | • |   |   | •         | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   | •         | 0 ... 5 V                |   |        |
|                        | 0 ... 5 V   |                        |   |   |   |           | 0 ... 20 mA              |   |        |
|                        |             |                        |   |   |   | •         | 4 ... 20 mA              |   |        |
|                        |             | •                      | • |   |   |           | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   |           | 0 ... 5 V                |   |        |
|                        | 1 ... 5 V   |                        |   |   |   | •         | 0 ... 20 mA              |   |        |
|                        |             |                        |   |   |   |           | 4 ... 20 mA              |   |        |
|                        |             | •                      | • |   |   | •         | 0 ... 10 V               |   |        |
|                        |             | •                      | • |   | • |           | 2 ... 10 V               |   |        |
|                        |             | •                      | • | • |   | •         | 0 ... 5 V                |   |        |
|                        |             | •                      | • |   |   | 1 ... 5 V |                          |   |        |



**JUMPFLEX® Transducers**  
Isolation amplifier, with zero/span adjustment

A set of jumpers are used to select the input and output signal range.  
Examples:

- 0-20 mA
- 4-20 mA
- 0 – 10 VDC
- 2-10 VDC
- 0-5 VDC
- 1-5 VDC

Circle the setting to select an input of 4-20 mA and an output of 2-10 volts?

# Omega T/C Transducer

This converter is powered by 24 VDC.

The input can be a type “K”, “T” or “J” thermocouple selected using jumpers.

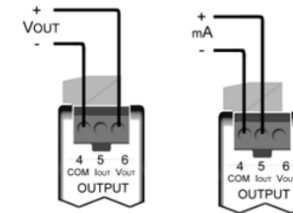
The device includes cold junction compensation.

The output is jumper selectable for 4 to 20 mA or 0 to 10 volts.

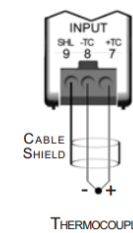
The device is calibrated using zero and span adjustments.



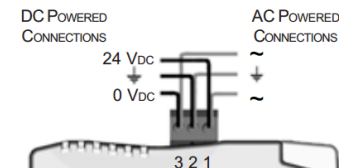
Output Connections



Input Connections



The \_\_\_\_\_ wire on a thermocouple is the negative signal?



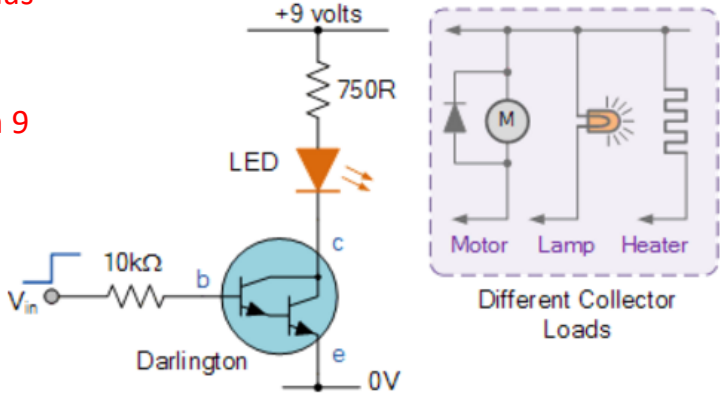
The advantage of using an arrangement such as this, is that the switching transistor is much more sensitive as only a tiny base current is required to switch a much larger load current as the typical gain of a Darlington configuration can be over 1,000 whereas normally a single transistor stage produces a gain of about 50 to 200.

Then we can see that a darlington pair with a gain of 1,000:1, could switch an output current of 1 ampere in the collector-emitter circuit with an input base current of just 1mA. This then makes darlington transistors ideal for interfacing with relays, lamps and motors to low power microcontroller, computer or logic controllers as shown.

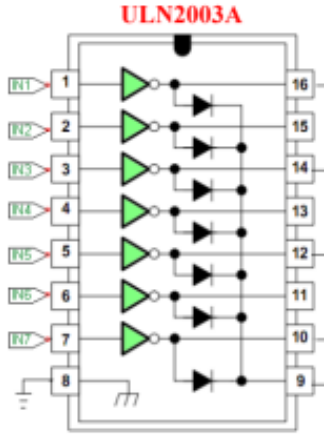
### Darlington Transistor Applications

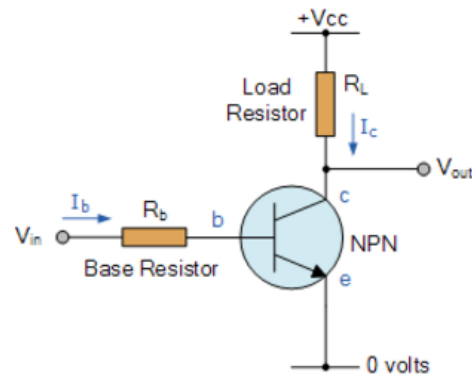
The Darlington pair transistor has a very \_\_\_\_\_ Beta?

When using inductive loads pin 9 must connect to the \_\_\_\_\_ ?



CAM8302E F2018 Exam Review





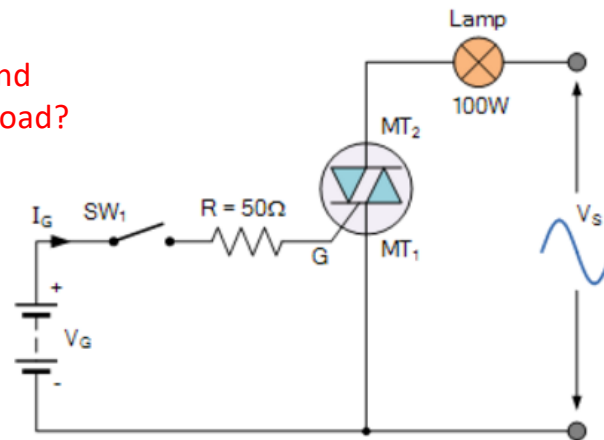
When turned on the voltage between the CE junction is very \_\_\_\_\_ ?

When the base of the NPN transistor is grounded (0 volts) and no base current,  $I_b$  flows, no current flows from the emitter to the collector and the transistor is therefore switched “OFF”. If the base is forward biased by more than 0.7 volts, a current will flow from the emitter to the collector and the transistor is said to be switched “ON”. When operated in these two modes, the transistor operates as a switch.

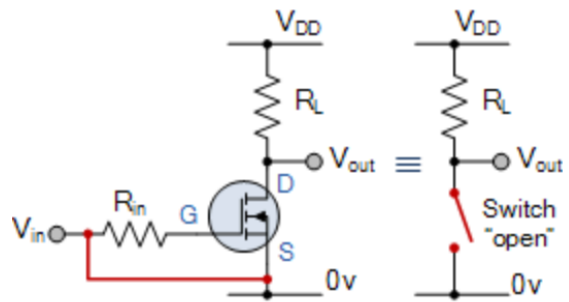
The problem here is that the transistors base needs to be switched between zero and some large, positive value for the transistor to become saturated at which point an increased base current,  $I_b$  flows into the device resulting in collector current  $I_c$  becoming large while  $V_{ce}$  is small. Then we can see that a small current on the base can control a much larger current flowing between the collector and the emitter.

## Triac Switching Circuit

Triacs have \_\_\_\_\_ leads and usually control an \_\_\_\_\_ load?



The circuit above shows a simple DC triggered triac power switching circuit. With switch  $SW_1$  open, no current flows into the Gate of the triac and the lamp is therefore “OFF”. When  $SW_1$  is closed, Gate current is applied to the triac from the battery supply  $V_G$  via resistor  $R$  and the triac is driven into full conduction acting like a closed switch and full power is drawn by the lamp from the sinusoidal supply.



## MOSFET as a Switch

We saw previously, that the N-channel, Enhancement-mode MOSFET (e-MOSFET) operates using a positive input voltage and has an extremely high input resistance (almost infinite) making it possible to interface with nearly any logic gate or driver capable of producing a positive output.

The MOSFET source connects to \_\_\_\_\_?

To “turn ON” the MOSFET the gate voltage must be \_\_\_\_\_ than the trigger voltage?

The MOSFET on resistance is very \_\_\_\_\_ the off resistance is very \_\_\_\_\_?

## The DC Stepper Motor

Like the DC motor above, **Stepper Motors** are also electromechanical actuators that convert a pulsed digital input signal into a discrete (incremental) mechanical movement are used widely in industrial control applications. A stepper motor is a type of synchronous brushless motor in that it does not have an armature with a commutator and carbon brushes but has a rotor made up of many, some types have hundreds of permanent magnetic teeth and a stator with individual windings.

As its name implies, the stepper motor does not rotate in a continuous fashion like a conventional DC motor but moves in discrete “Steps” or “Increments”, with the angle of each rotational movement or step dependant upon the number of stator poles and rotor teeth the stepper motor has.

Because of their discrete step operation, stepper motors can easily be rotated a finite fraction of a rotation at a time, such as 1.8, 3.6, 7.5 degrees etc. So for example, lets assume that a stepper motor completes one full revolution ( $360^\circ$ ) in exactly 100 steps.

Then the step angle for the motor is given as  $360 \text{ degrees} / 100 \text{ steps} = 3.6 \text{ degrees per step}$ . This value is commonly known as the stepper motors **Step Angle**.



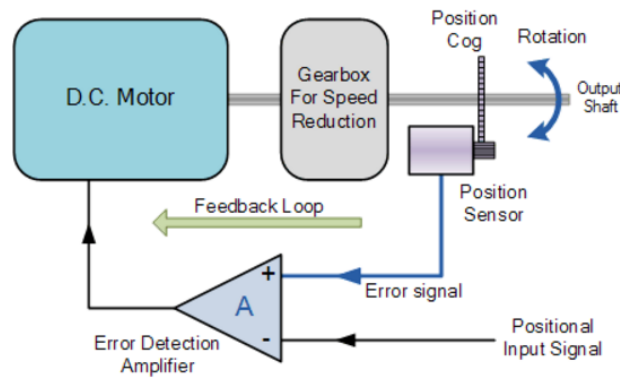
Stepper Motor

A stepper motor with a step angle of 1.8 degrees per step has \_\_\_\_\_ steps per revolution?

A stepper motor has 48 steps per revolution, each step angle equals \_\_\_\_\_ degrees.

A 1.8 degrees/step stepper motor is operated using 1/32 micro stepping. How long will it take to rotate one revolution a 30 us/step? \_\_\_\_\_

DC Servo Motor Block Diagram



A servo motor consists of a DC motor, reduction gearbox, positional feedback device and some form of error correction. The speed or position is controlled in relation to a positional input signal or reference signal applied to the device.

The error detection amplifier looks at this input signal and compares it with the feedback signal from the motors output shaft and determines if the motor output shaft is in an error condition and, if so, the controller makes appropriate corrections either speeding up the motor or slowing it down. This response to the positional feedback device means that the servo motor operates within a “Closed Loop System”.



RC Servo Motor  
CAM8302E F2018 Exam Review

What is the typical period of the PWM signal of a typical servo motor? \_\_\_\_\_ ms.

What is difference between an incremental optical encoder and an absolute optical encoder?

Which encoder type is used in the DC motors used in the lab? \_\_\_\_\_

Which type is used in industrial robots? \_\_\_\_\_

## The “Brushless” DC Motor

The brushless DC motor (BDCM) is very similar to a permanent magnet DC motor, but does not have any brushes to replace or wear out due to commutator sparking. Therefore, little heat is generated in the rotor increasing the motors life. The design of the brushless motor eliminates the need for brushes by using a more complex drive circuit where the rotor magnetic field is a permanent magnet which is always in synchronisation with the stator field allows for a more precise speed and torque control.

Then the construction of a brushless DC motor is very similar to the AC motor making it a true synchronous motor but one disadvantage is that it is more expensive than an equivalent “brushed” motor design.

The control of the brushless DC motors is very different from the normal brushed DC motor, in that it this type of motor incorporates some means to detect the rotors angular position (or magnetic poles) required to produce the feedback signals required to control the semiconductor switching devices. The most common position/pole sensor is the “Hall Effect Sensor”, but some motors also use optical sensors.

Using Hall effect sensors, the polarity of the electromagnets is switched by the motor control drive circuitry. Then the motor can be easily synchronized to a digital clock signal, providing precise speed control. Brushless DC motors can be constructed to have, an external permanent magnet rotor and an internal electromagnet stator or an internal permanent magnet rotor and an external electromagnet stator.

Advantages of the **Brushless DC Motor** compared to its “brushed” cousin is higher efficiencies, high reliability, low electrical noise, good speed control and more importantly, no brushes or commutator to wear out producing a much higher speed. However their disadvantage is that they are more expensive and more complicated to control.

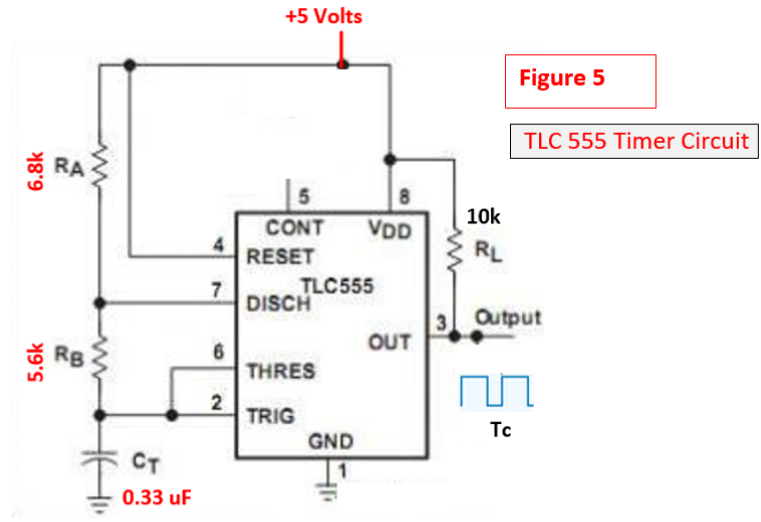
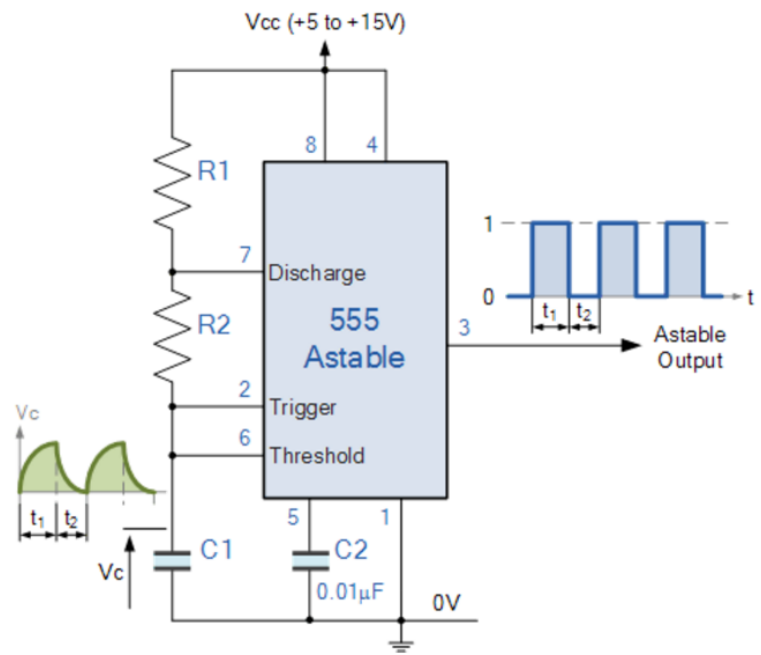
Brushless motors usually last \_\_\_\_\_ because they have no \_\_\_\_\_ ?



Brushless DC Outrunner motor



ESC – Electronic Speed Control (PWM)

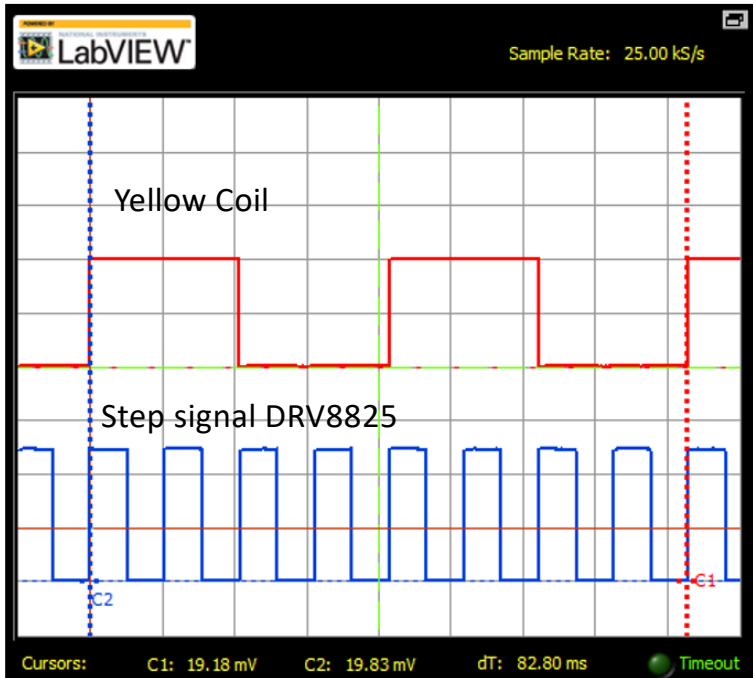


**Figure 5**  
TLC 555 Timer Circuit

$$t_{c(H)} \approx C_T (R_A + R_B) \ln 2 \quad (\ln 2 = 0.693)$$

$$t_{c(L)} \approx C_T R_B \ln 2$$

Calculate the period, frequency, % duty cycle, time high and time low of the 555 timer above.



Basic Settings    Advanced Settings

**Channel 0 Settings** ■

Source: AI 0

Enabled

Probe: 1x    Coupling: DC

Scale Volts/Div: 5 V    Vertical Position (Div): 0

**Channel 1 Settings** ■

Source: AI 1

Enabled

Probe: 1x    Coupling: DC

Scale Volts/Div: 2 V    Vertical Position (Div): -4

**Timebase**

Time/Div: 10 ms

**Trigger**

Type: Edge

Slope:  Rising

Source: Chan 1 Source

Level (V): 2

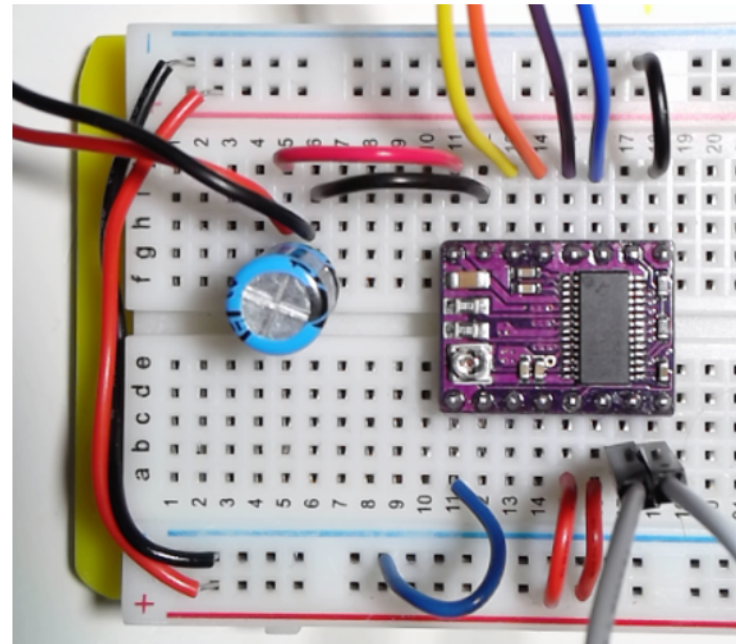
Horizontal Position (%): 10

\*\* Assume the motor is 7.5 degrees/step.  
 What is the speed of the stepper motor in RPS?  
 The scope is triggered on a rising edge T/F?

The scope trigger level should be set to about \_\_\_\_\_ in this example?

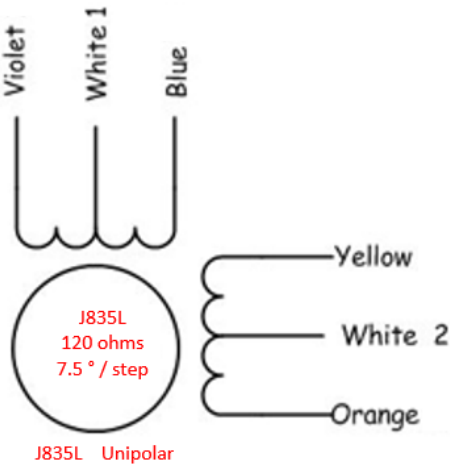
## Stepper\_FWD\_Test\_Drv8825\_F2016\_tone1 \$

```
1 /* Stepper Motor Control using a
2 * Pololu DRV8825 Stepper Controller
3 * November 30th 2016
4 * Michel Hanbury for CAM8302E
5 */
6 void setup() {
7   pinMode(2, OUTPUT); // step pulse
8   pinMode(3, OUTPUT); // direction
9 }
10
11 void loop()
12 {
13   digitalWrite(3, HIGH); //direction
14
15   tone(2,96); // 1/96 Hertz = 10.42 ms.
16
17   // tone(pin,freq,duration in ms)
18
19   // range 31 Hertz to 65,535 Hertz -- integer //
20   // square wave //
```



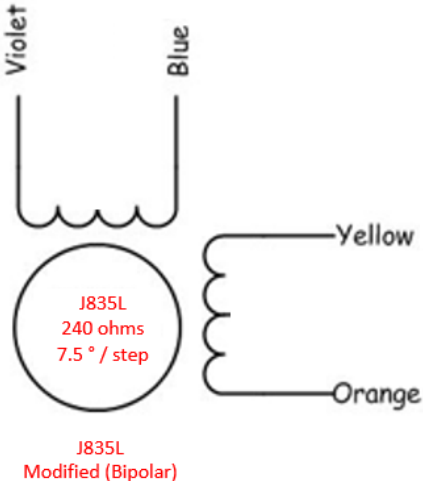
List two ways the motor direction can be changed, one software and one hardware?

Unipolar Motor 120 ohms between the centre tap wire and an outside lead.  
 White1 to blue = 120 ohms.



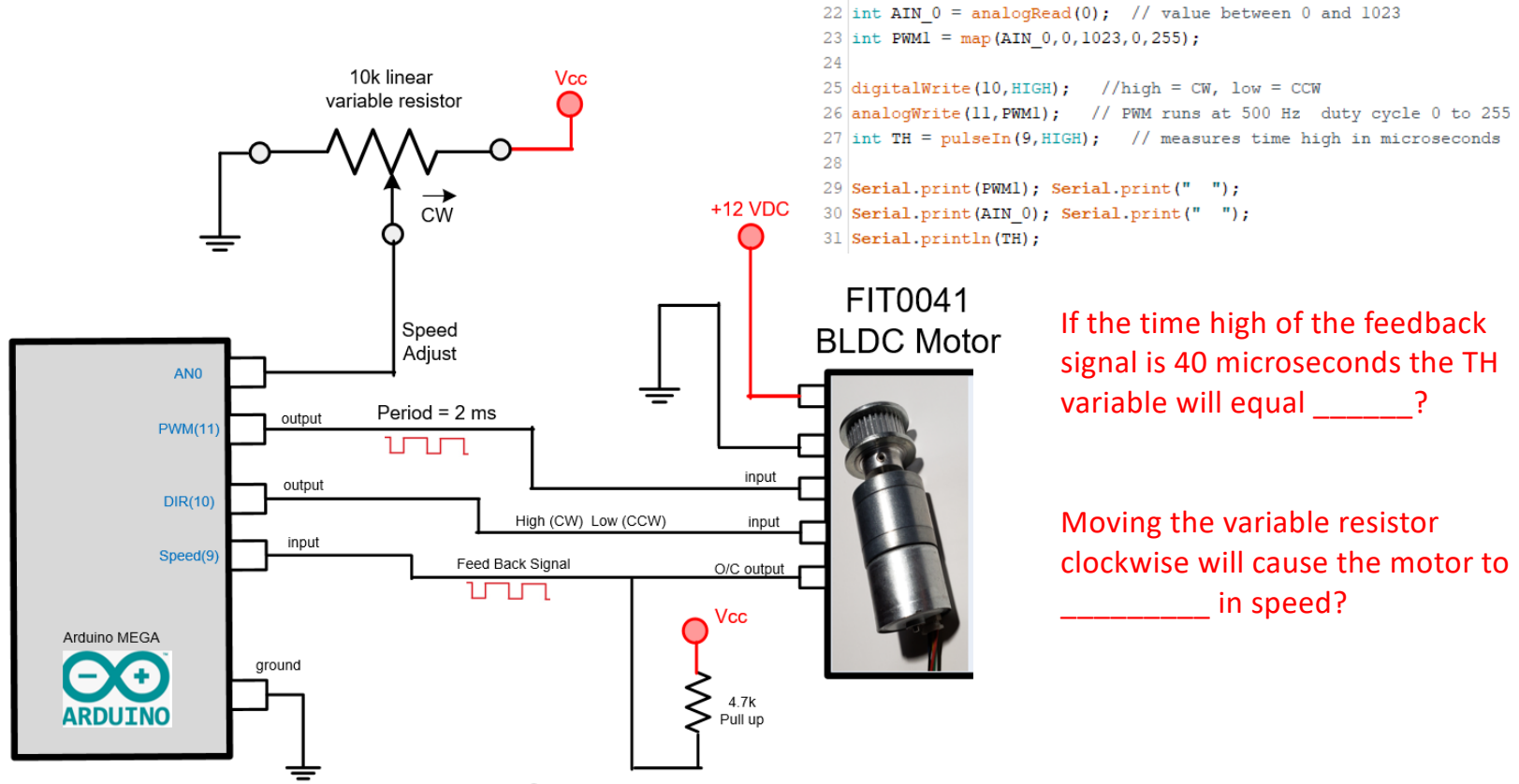
How much current flows through white and violet with a 10 volt supply to the stepper controller?

Bipolar stepper motor – this is the modified stepper the centre tap white wires were cut. The bipolar motor can be used with the DRV8825 bipolar motor and will have a greater maximum torque.



How much current flows through blue and violet with a 10 volt supply to the motor controller?

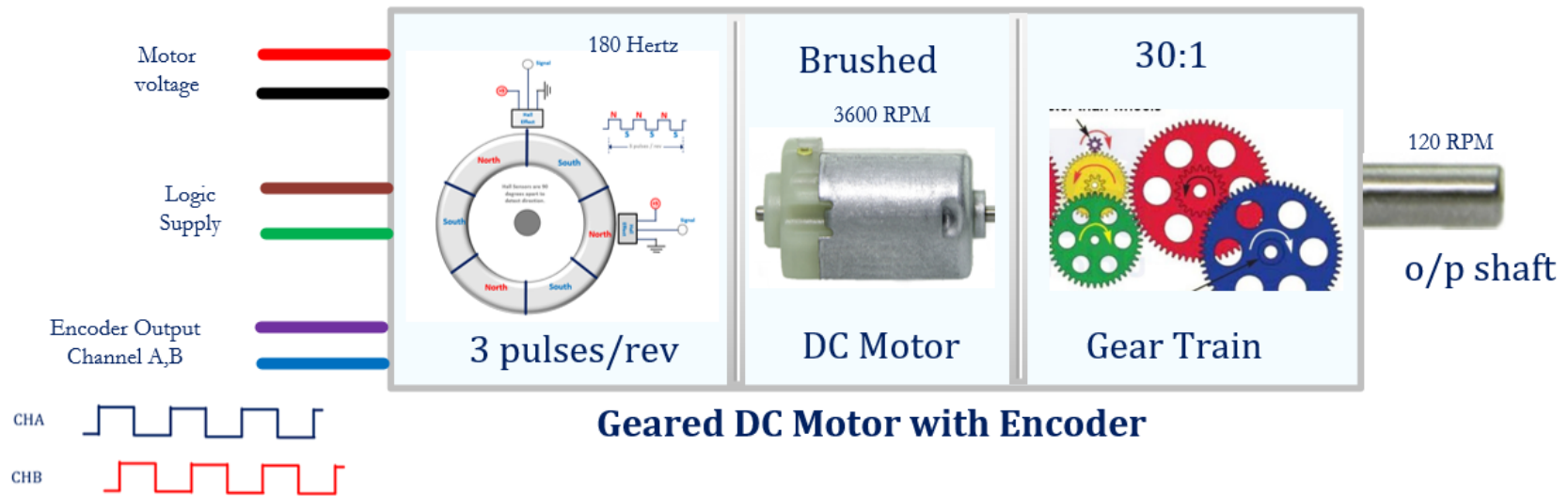
# Brushless DC Motor (BLDC)



If the time high of the feedback signal is 40 microseconds the TH variable will equal \_\_\_\_\_?

Moving the variable resistor clockwise will cause the motor to \_\_\_\_\_ in speed?

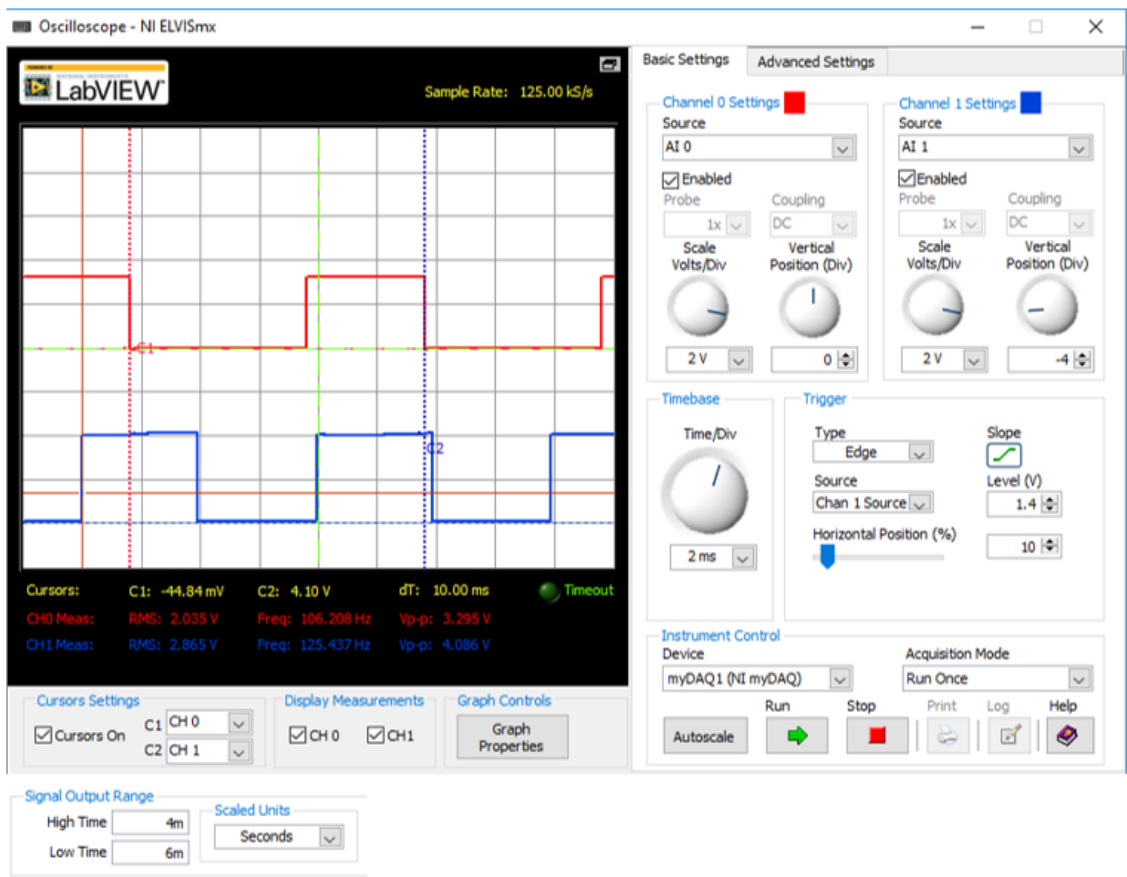
If the encoder frequency equals 1000 Hertz  
the output shaft RPM equals \_\_\_\_\_?



The gears are used in this application to \_\_\_\_\_ the \_\_\_\_\_ on the output shaft?

```
22 int AIN_0 = analogRead(0); // value between 0 and 1023
23 int PWM1 = map(AIN_0,0,1023,0,255);
24
25 digitalWrite(10,HIGH); //high = CW, low = CCW
26 analogWrite(11,PWM1); // PWM runs at 500 Hz duty cycle 0 to 255
27 int TH = pulseIn(9,HIGH); // measures time high in microseconds
28
29 Serial.print(PWM1); Serial.print(" ");
30 Serial.print(AIN_0); Serial.print(" ");
31 Serial.println(TH);
```

If the analog input equals 3.0 volts, the PWM variable will equal \_\_\_\_\_?



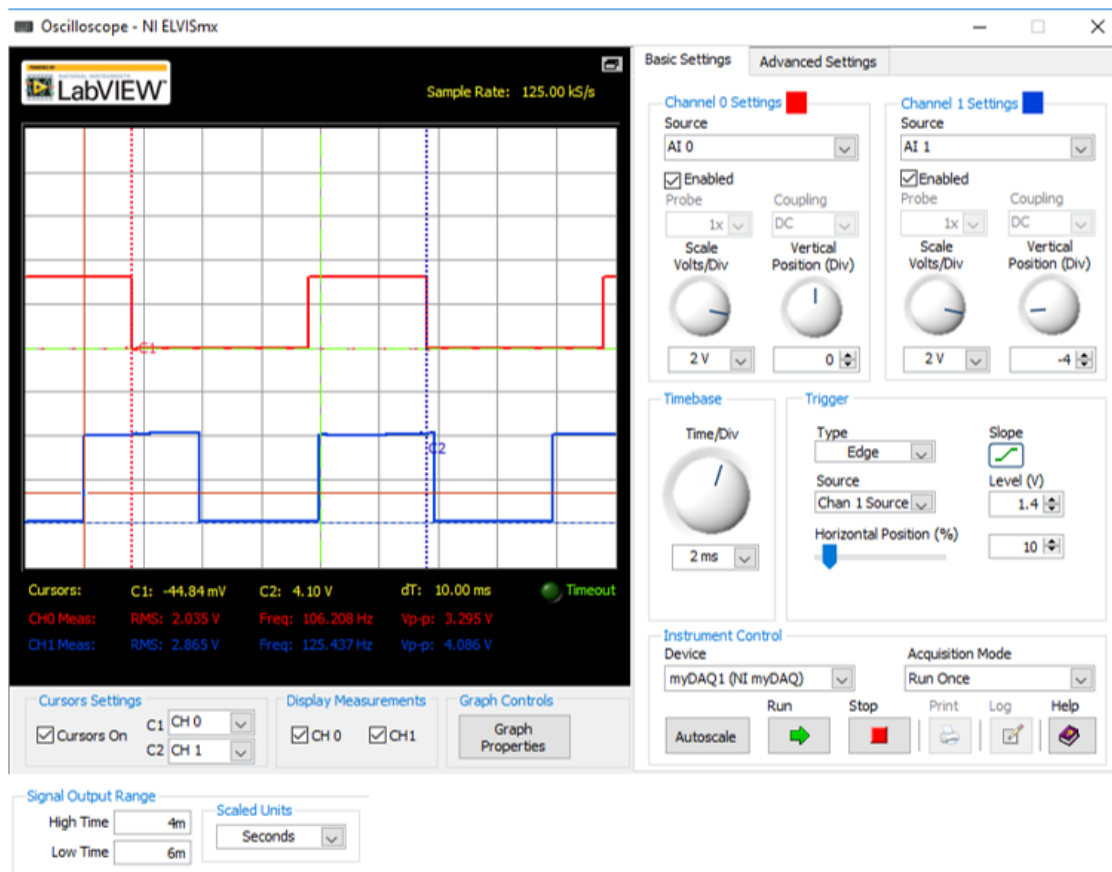
F2018

This is a screen capture from lab 6. Which trace is the PWM signal, which trace is the encoder signal?

The encoder signal is from one of the two channels from the DC motor. The motor produces 3 encoder pulses for each revolution of the DC motor. The motor has a gear ratio of 30:1.

How fast is the output shaft turning in RPMs \_\_\_\_\_?

F2018



This is a screen capture from lab 6. One trace is the PWM signal the other is the motor encoder signal.

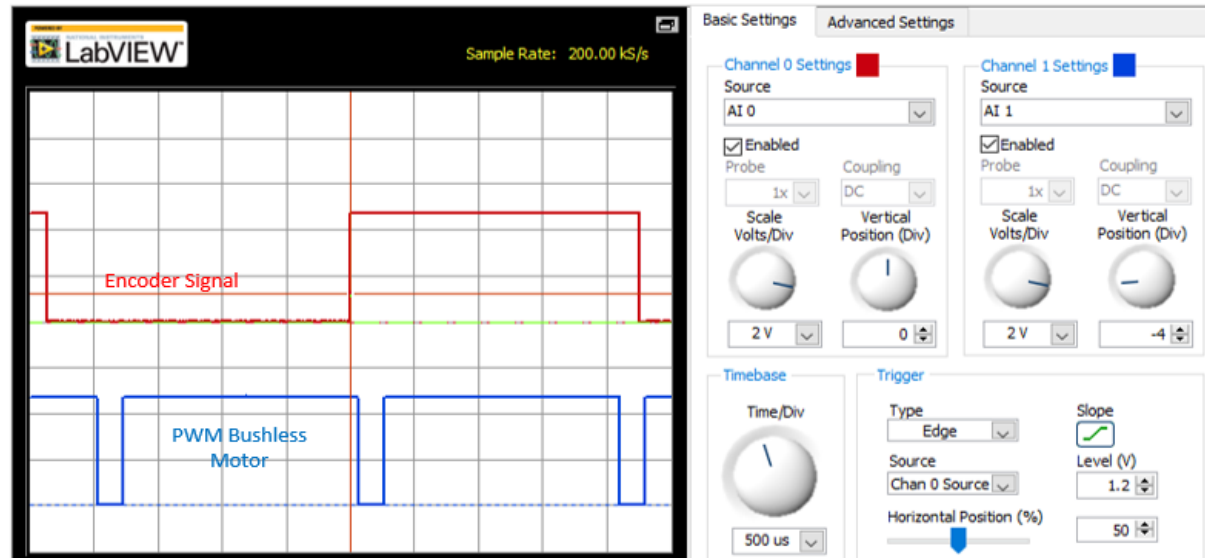
Which trace is would you measure to determine motor speed PWM or encoder signal?

Explain why?

Is the PWM signal synchronized with the encoder signal?

If the duty cycle of the PWM signal is increased the period of the encoder signal will \_\_\_\_\_?

F2018



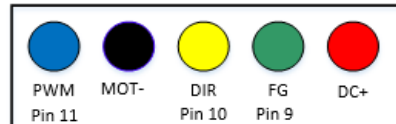
Based on the information in the screen capture the DC brushless motor is turning at \_\_\_\_\_ RPM.

Based on the information in the screen capture the PWM signal has a duty cycle of \_\_\_\_\_% and a frequency of \_\_\_\_\_ Hz.

```
22 int AIN_0 = analogRead(0); // value between 0 and 1023
23 int PWM1 = map(AIN_0, 0, 1023, 0, 255);
24
25 digitalWrite(10, HIGH); //high = CW, low = CCW
26 analogWrite(11, PWM1); // PWM runs at 500 Hz duty cycle 0 to 255
27 int TH = pulseIn(9, HIGH); // measures time high in microseconds
28
29 Serial.print(PWM1); Serial.print(" ");
30 Serial.print(AIN_0); Serial.print(" ");
31 Serial.println(TH);
```

A/D voltage = 4.27 (Supply = 4.74)

DC Brushless Motor FIT4401  
45:1 gear ratio (12 VDC 75 mA)  
6 pulses per revolution



-> 230 925 2270

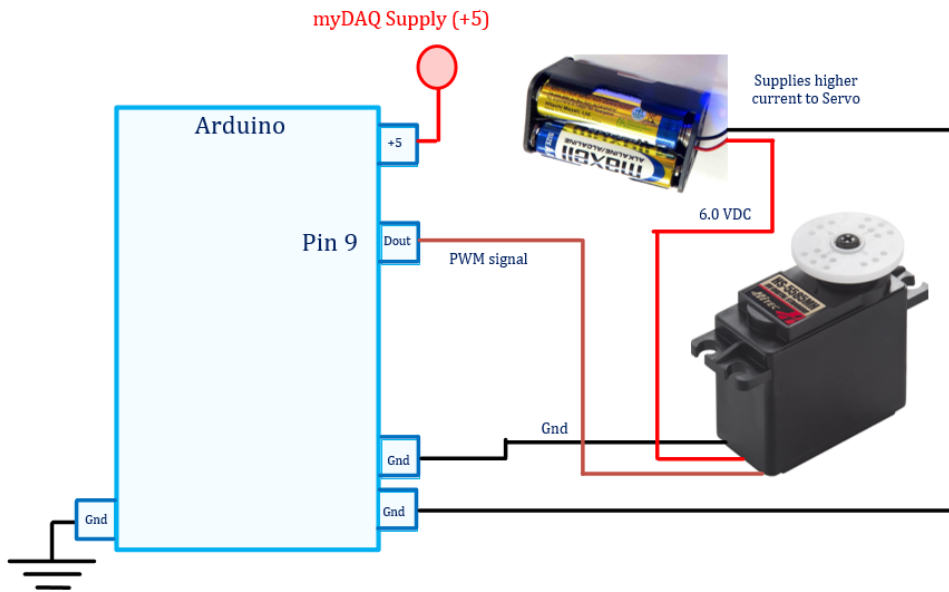
# Absolute Encoder



What is the benefit of using an absolute encoder compared to an incremental encoder?

The absolute encoder knows the current position on startup of the system. The encoder uses a pattern on the rotating disk and optics to determine position. A quadrature encoder is an incremental encoder.

CAM8302E F2018 Exam Review



Servo hardware configuration.

What is the typical period of a servo motor control signal?

The time high of the control signal is about \_\_\_\_\_ ms when the servo is in the centre position.

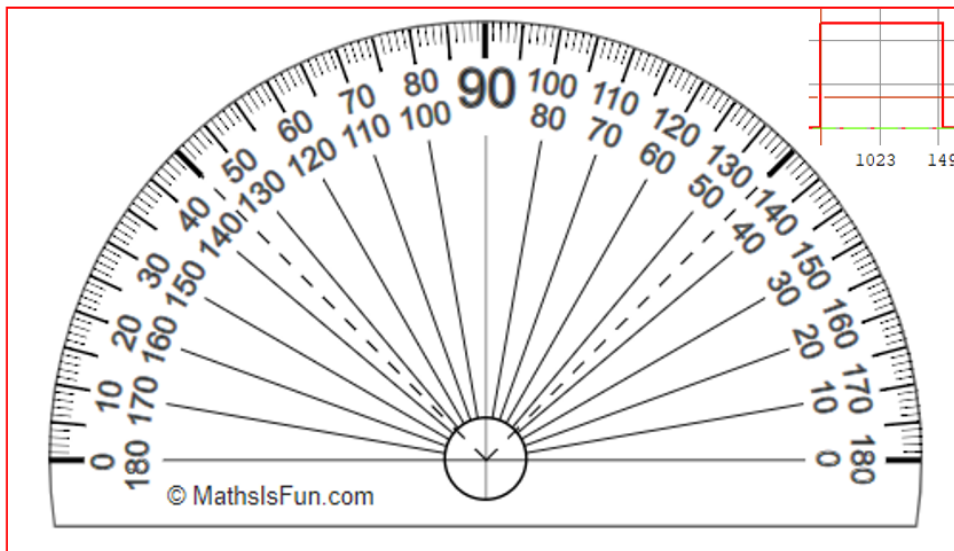
**Servo Motor Connections to Arduino:** Connect the power for the servo to a 4.5 to 6.5 volt battery or DC power supply. The signal controlling the servo motor shaft position is controlled using an Arduino PWM output.

```

23 Analog_IN = analogRead(potpin);           // reads the value of the potentiometer (value between 0 and 1023)
24 Servo_Angle = map(Analog_IN, 0, 1024, 35, 150); // scale it to use it with the servo (value between 0 and 180)
25 myservo.write(Servo_Angle);
26
27 | Serial.print(Analog_IN);
28 | Serial.print(" ");
29 | Serial.println(Servo_Angle);

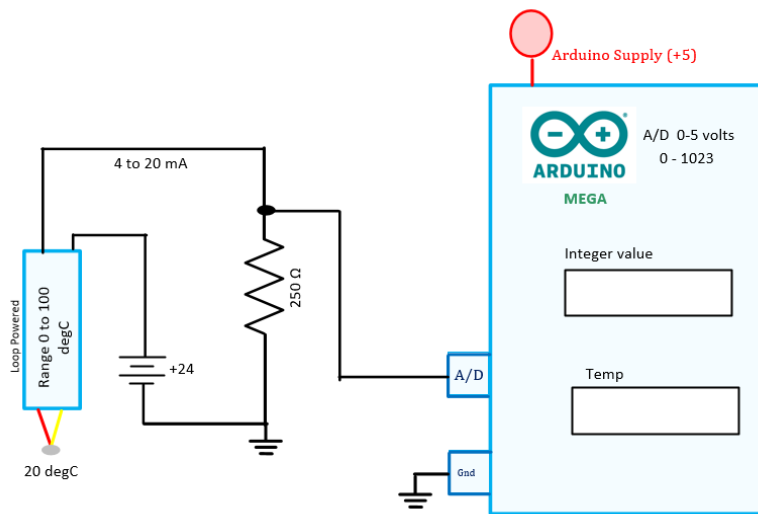
```

Analog\_In = 1023  
Servo\_Angle = 149

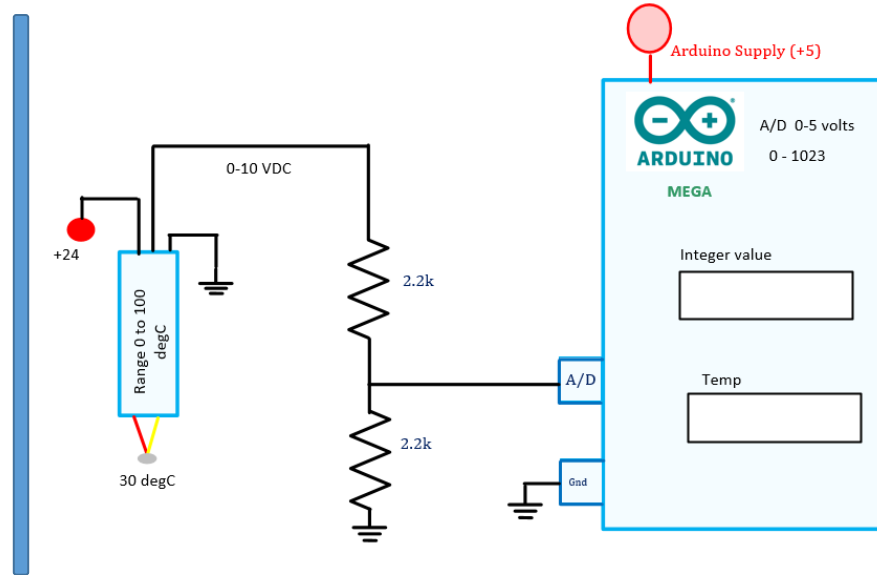


Based on the information in the screen capture the Servo Motor will rotate about \_\_\_\_\_ from the full CCW position.

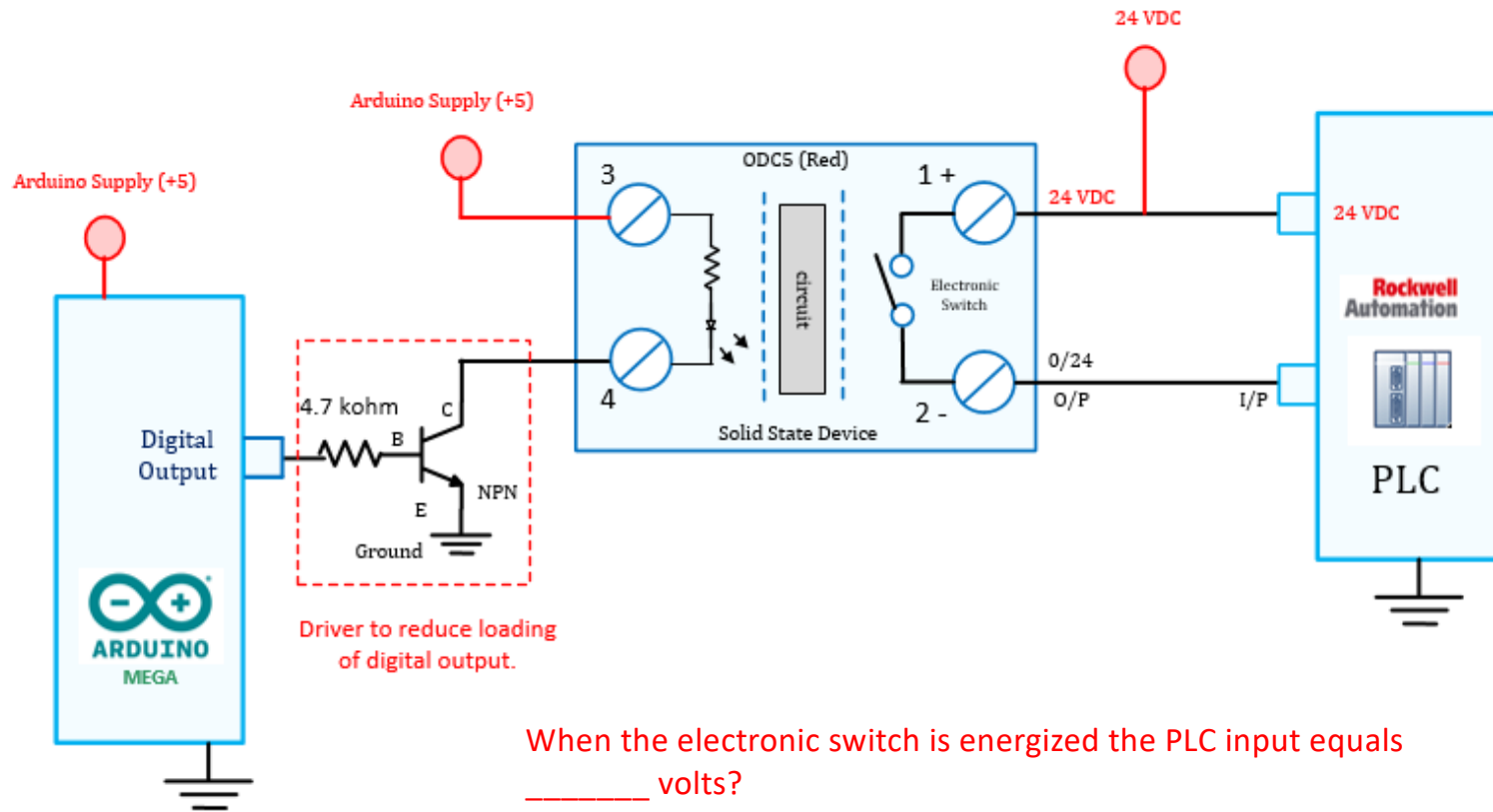
Based on the information in the screen capture the Servo Motor control signal will have a PWM duty cycle of \_\_\_\_\_% and a period of \_\_\_\_\_ Hz.

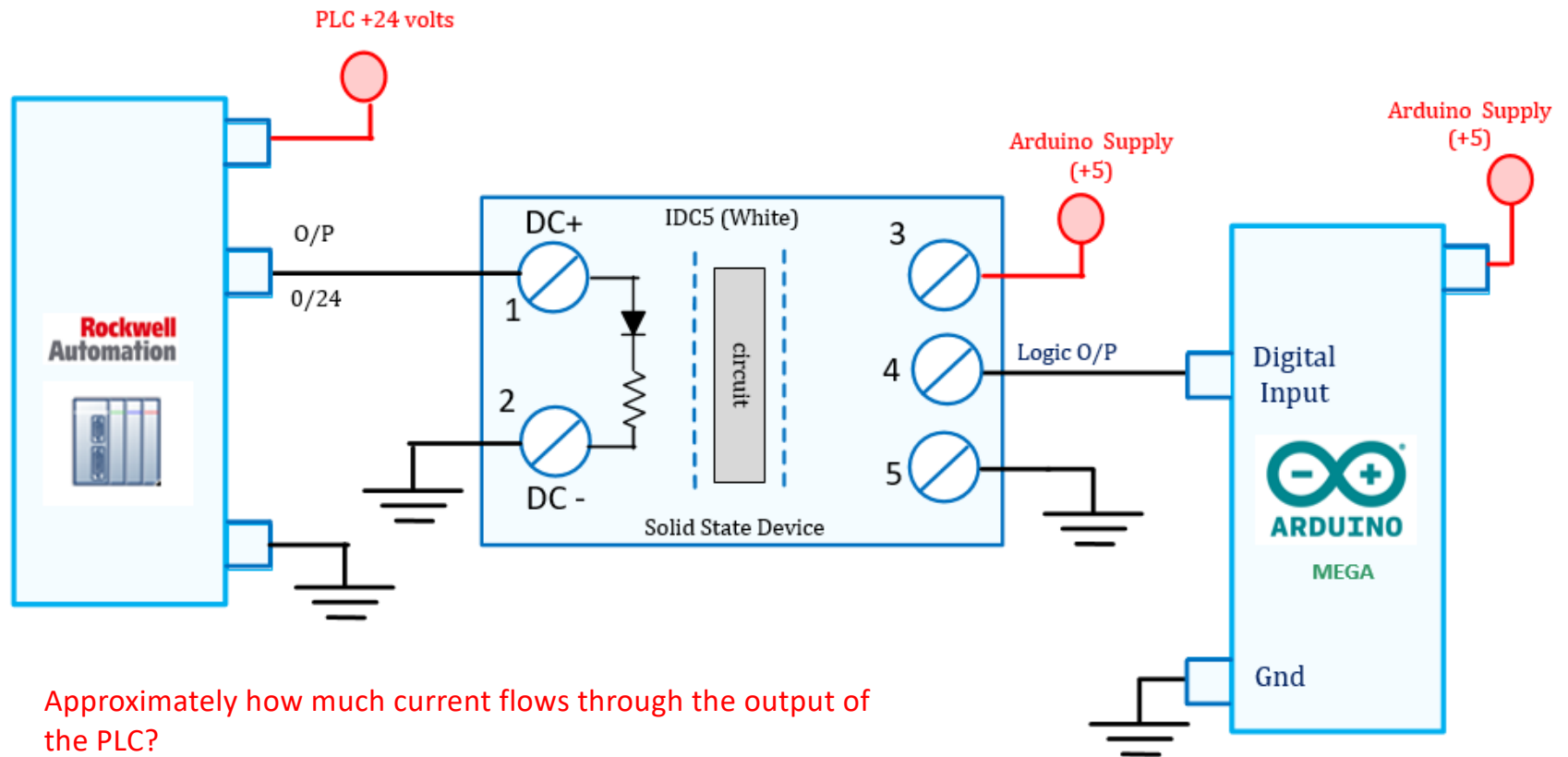


Determine the Arduino integer value when the temperature equals 20 degC.

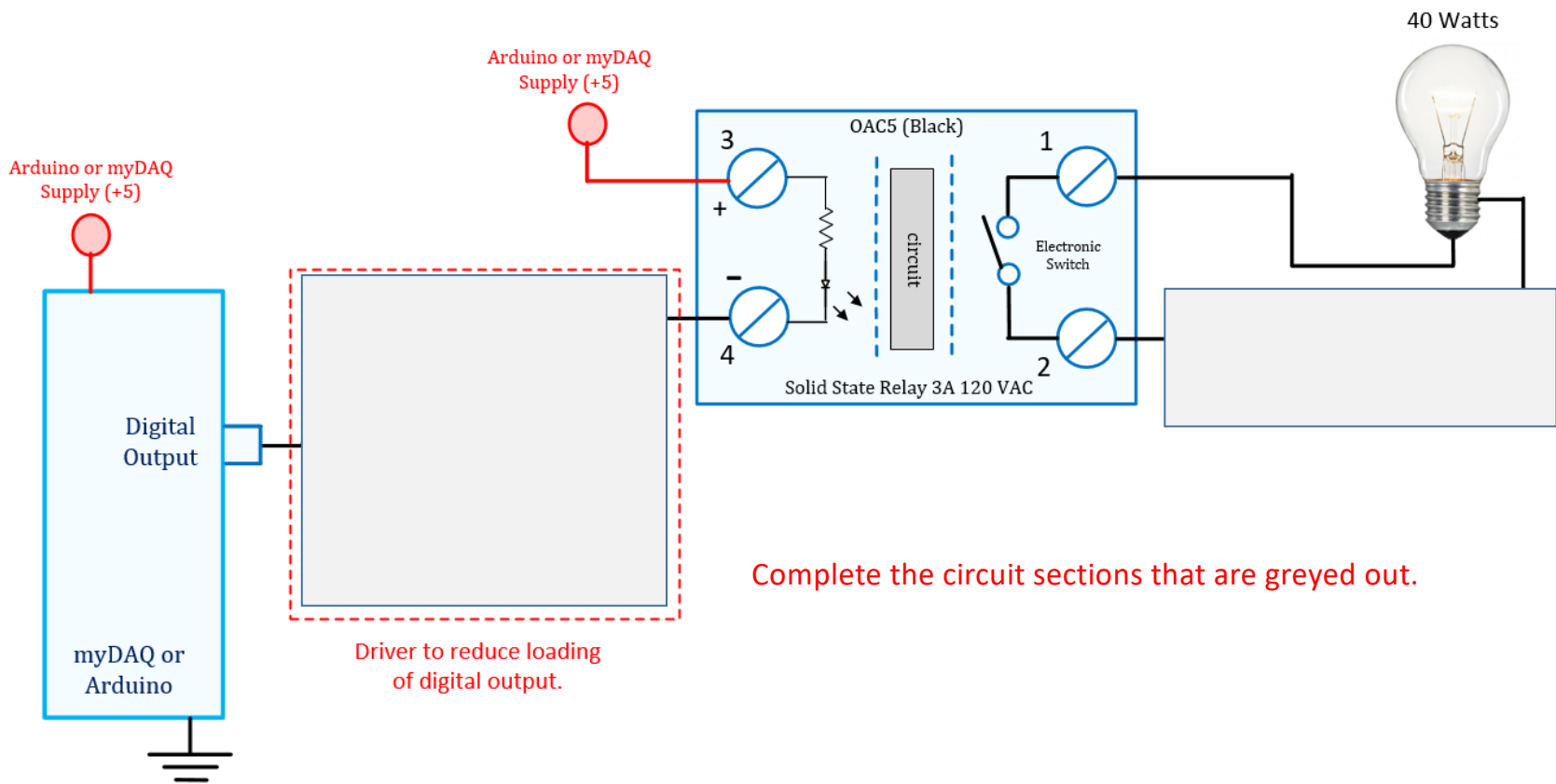


Determine the Arduino integer value when the temperature equals 30 degC.



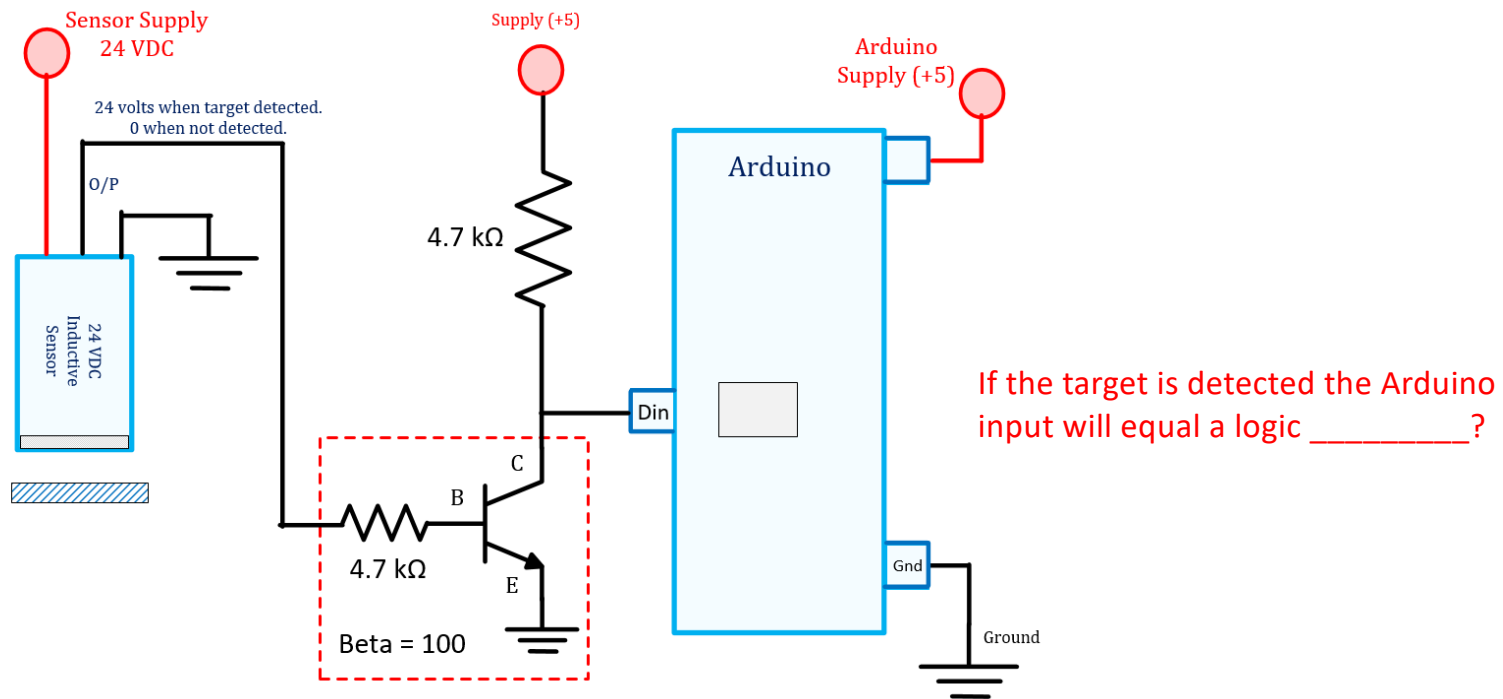


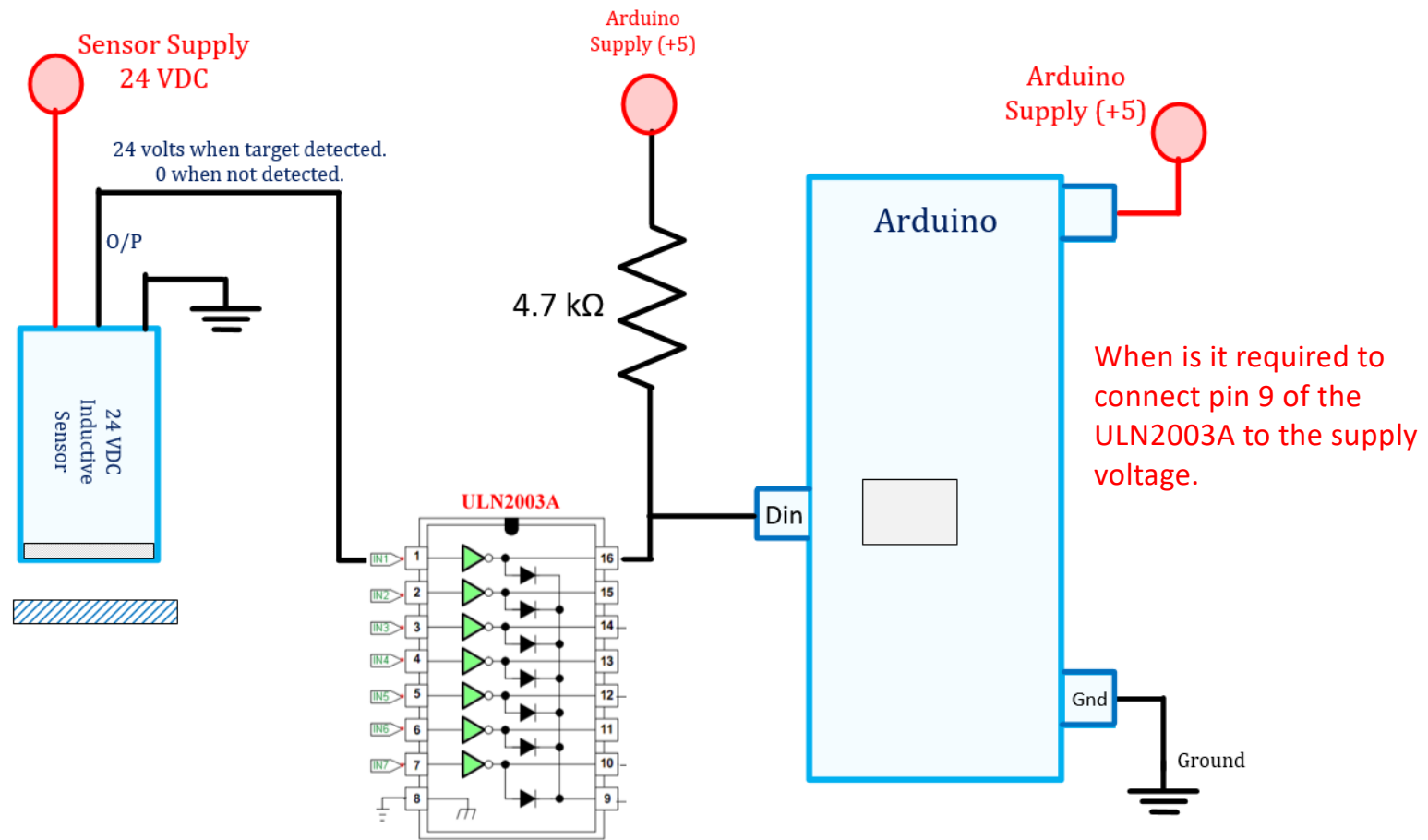
Approximately how much current flows through the output of the PLC?



Complete the circuit sections that are greyed out.

# Inductive Sensor





# Type "K" (yellow) Thermocouple

TABLE 9 Type K Thermocouple — thermoelectric voltage as a function of temperature (°C); reference junctions at 0 °C

K<sup>o</sup>C

| °C                                   | 0      | 1      | 2      | 3      | 4      | 5      | 6      | 7      | 8      | 9      | 10     | °C  |
|--------------------------------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|-----|
| Thermoelectric Voltage in Millivolts |        |        |        |        |        |        |        |        |        |        |        |     |
| -30                                  | -1.156 | -1.194 | -1.231 | -1.268 | -1.305 | -1.343 | -1.380 | -1.417 | -1.453 | -1.490 | -1.527 | -30 |
| -20                                  | -0.778 | -0.816 | -0.854 | -0.892 | -0.930 | -0.968 | -1.006 | -1.043 | -1.081 | -1.119 | -1.156 | -20 |
| -10                                  | -0.392 | -0.431 | -0.470 | -0.508 | -0.547 | -0.586 | -0.624 | -0.663 | -0.701 | -0.739 | -0.778 | -10 |
| 0                                    | 0.000  | -0.039 | -0.079 | -0.118 | -0.157 | -0.197 | -0.236 | -0.275 | -0.314 | -0.353 | -0.392 | 0   |
| 0                                    | 0.000  | 0.039  | 0.079  | 0.119  | 0.158  | 0.198  | 0.238  | 0.277  | 0.317  | 0.357  | 0.397  | 0   |
| 10                                   | 0.397  | 0.437  | 0.477  | 0.517  | 0.557  | 0.597  | 0.637  | 0.677  | 0.718  | 0.758  | 0.798  | 10  |
| 20                                   | 0.798  | 0.838  | 0.879  | 0.919  | 0.960  | 1.000  | 1.041  | 1.081  | 1.122  | 1.163  | 1.203  | 20  |
| 30                                   | 1.203  | 1.244  | 1.285  | 1.326  | 1.366  | 1.407  | 1.448  | 1.489  | 1.530  | 1.571  | 1.612  | 30  |
| 40                                   | 1.612  | 1.653  | 1.694  | 1.735  | 1.776  | 1.817  | 1.858  | 1.899  | 1.941  | 1.982  | 2.023  | 40  |
| 50                                   | 2.023  | 2.064  | 2.106  | 2.147  | 2.188  | 2.230  | 2.271  | 2.312  | 2.354  | 2.395  | 2.436  | 50  |
| 60                                   | 2.436  | 2.478  | 2.519  | 2.561  | 2.602  | 2.644  | 2.685  | 2.727  | 2.768  | 2.810  | 2.851  | 60  |
| 70                                   | 2.851  | 2.893  | 2.934  | 2.976  | 3.017  | 3.059  | 3.100  | 3.142  | 3.184  | 3.225  | 3.267  | 70  |
| 80                                   | 3.267  | 3.308  | 3.350  | 3.391  | 3.433  | 3.474  | 3.516  | 3.557  | 3.599  | 3.640  | 3.682  | 80  |
| 90                                   | 3.682  | 3.723  | 3.765  | 3.806  | 3.848  | 3.889  | 3.931  | 3.972  | 4.013  | 4.055  | 4.096  | 90  |

If the CJ is at 22 degC and the HJ is 40 degC the T/C voltage equals \_\_\_\_\_ mV?

About 40 uVolts /degree Celsius Type K

# AD623 Instrumentation Amplifier

The AD623 is an instrumentation amplifier. The gain of the amplifier is easily configured using one resistor. An output offset can also be easily added with a reference input pin. The device can be used to amplify signals from load cells, pressure sensors and thermocouple. The device has rail to rail output meaning the output signal can go from min supply to maximum supply.

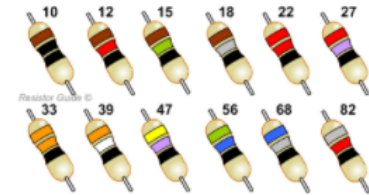
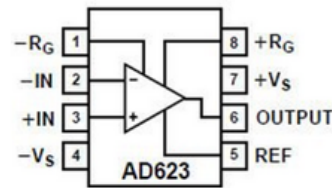


Table I. Required Values of Gain Resistors

| Desired Gain | 1% Std Table Value of $R_G$ , $\Omega$ | Calculated Gain Using 1% Resistors |
|--------------|--|------------------------------------|
| 2            | 100 k                                  | 2                                  |
| 5            | 24.9 k                                 | 5.02                               |
| 10           | 11 k                                   | 10.09                              |
| 20           | 5.23 k                                 | 20.12                              |
| 33           | 3.09 k                                 | 33.36                              |
| 40           | 2.55 k                                 | 40.21                              |
| 50           | 2.05 k                                 | 49.78                              |
| 65           | 1.58 k                                 | 64.29                              |
| 100          | 1.02 k                                 | 99.04                              |
| 200          | 499                                    | 201.4                              |
| 500          | 200                                    | 501                                |
| 1000         | 100                                    | 1001                               |

AD623 Data Sheet info



If the gain resistor equals 1.8k the gain equals \_\_\_\_\_?

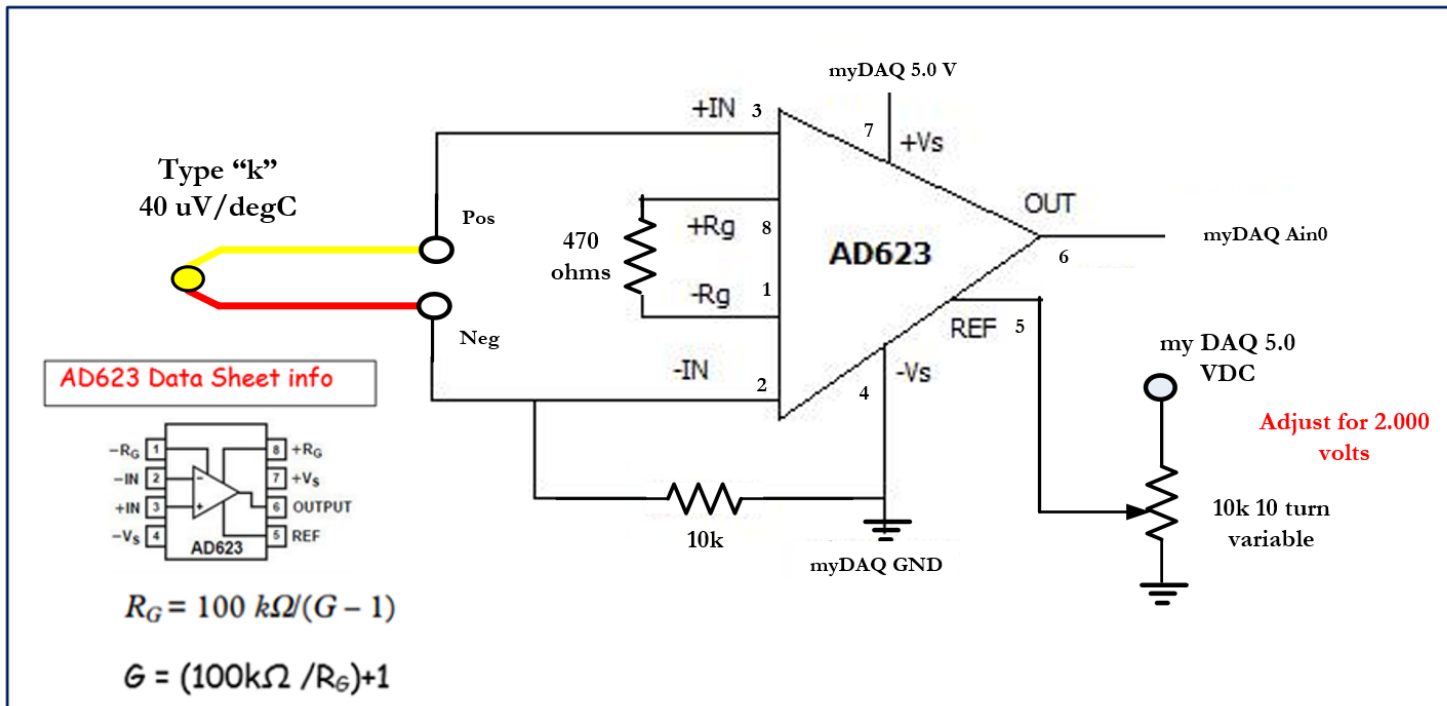
$$R_G = 100 \text{ k}\Omega / (G - 1)$$

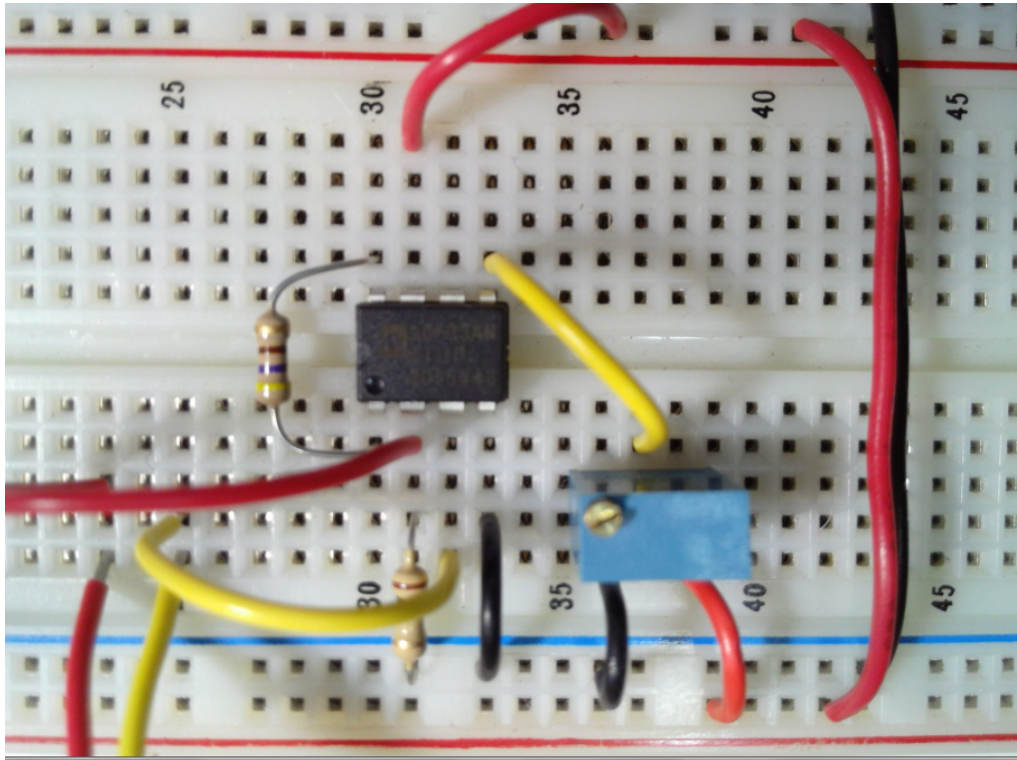
$$G = (100\text{k}\Omega / R_G) + 1$$

Determine the gain of the AD623 instrumentation amplifier when using a 470 ohm resistor?

If the resistance from the wiper to ground equals 4k, the reference voltage equals \_\_\_\_\_ volts.

If the CJ = 22 degC and the HJ is 39 degC the output of the amplifier equals \_\_\_\_\_ volts. Answer to +/- 2mV.





What is the value of the gain resistor?

What is gain value?

What is the purpose of the variable resistor?

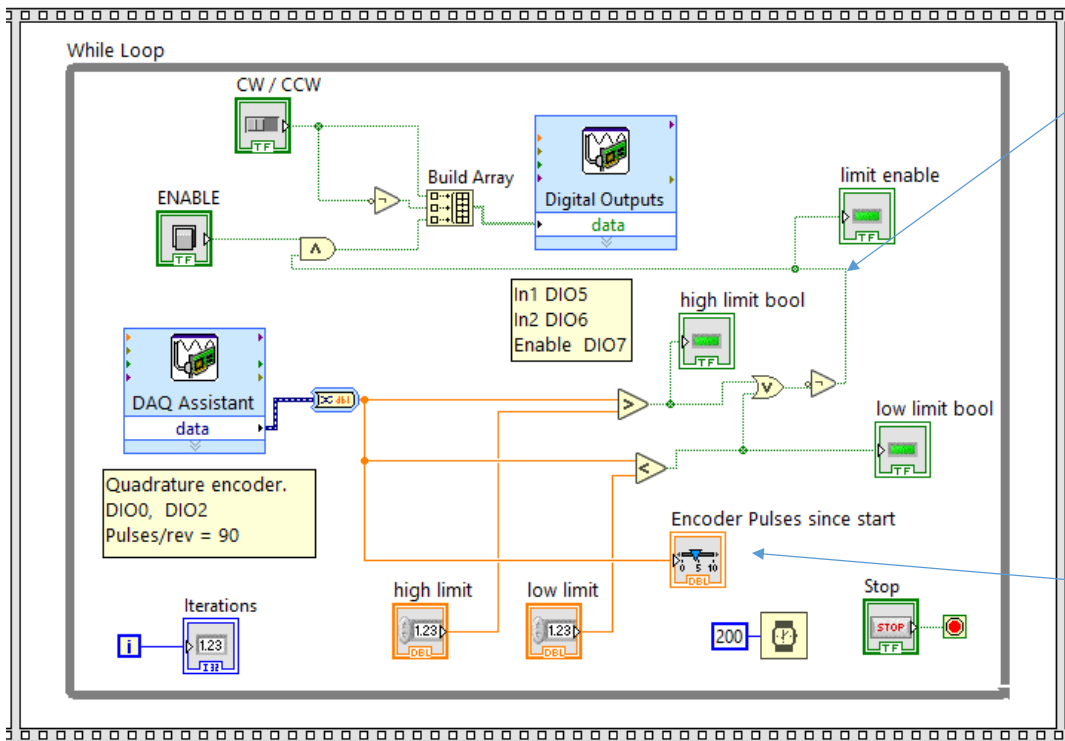


## Interrupt Program for Stepper

```
5 #define direction 4
6 #define step 5
7 #define steptime1 20 // micro seconds 10.4 high, 10.4 low
8 #define steptime2 0 // milli seconds period = 20.8ms * 48 = 1 second/rev.
9
10 void setup() {
11   pinMode(step, OUTPUT); // step pulse
12   pinMode(direction, OUTPUT); // direction
13   attachInterrupt(digitalPinToInterrupt(2),direction1, FALLING); // from optical
14   attachInterrupt(digitalPinToInterrupt(3),direction2, RISING); // from push button
15   digitalWrite(direction, LOW); //start in one direction
16 }
-- .. ..
```

What are the two benefits of interrupts over polling? \_\_\_\_\_, \_\_\_\_\_

# Quadrature Encoding



This signal will be equal to a logic \_\_\_\_\_ after the motor has turned \_\_\_\_\_ revolutions.

\*\* Assume the default front panel.

After turning 2 revolutions the "Encoder Pulses since start" will equal \_\_\_\_\_?