**Objective**

* Using communication hardware to communicate with other devices

**Parts**

* Hook up the thermocouple circuit to your Arduino, program and demonstrate the program to your instructor.

**Questions**

1. How did you test the thermocouple in the lab?
   1. I tested the thermocouple by holding the sensor with my hand to make the temperature rise and then watched it fall.

1. Based on the calculation used to compute the temperature, what is the resolution in degrees Fahrenheit?
   1. The accuracy of the temperature in Fahrenheit is to the hundredths place (XX.XX). And the resolution should be 12 bits.

1. What is the range of temperatures that this sensor can measure? Realize that 0 from the chip is zero degrees Celsius, so what is the maximum value it can read?
   1. The range of temperatures the sensor can measure is found accordingly:
      1. We know that one bit represents .25 degrees in Celsius
         1. (bit 3 = 0 degrees, bit 4 = .25 degrees, bit 5 = .50 degrees, … , bit 14 = 3)
      2. As a result, we can simply do some algebra to obtain the range.
         1. 12 bits \* .25 = 3
         2. (1.8 \* 3) + 32 = 37.4
         3. 0 degrees to 37.4 degrees Fahrenheit = range

**Appendix A**

#include <SPI.h> //using the SPI libraries

int local; //int for just a local variable

unsigned long secondTimer; //unsigned long for the seconds timer

float Temperature; //float to hold the calculated temperature

float ThermoCouple; //float to use for the ThermoCouple

// function to read temperature from Max6675

float Read\_SPI\_Temperature()

{

SPI.beginTransaction(SPISettings(8000000, MSBFIRST, SPI\_MODE0)); //set up the SPI

PORTC &= ~0x04; //set the chip select low

int TempData = SPI.transfer16(0); Sets the temp data from transfer

PORTC |= 0x04; //sets chip select high

SPI.endTransaction(); //ends SPI transaction

TempData = (TempData / 8 ) & 0x0fff; //Masks tempData

Temperature = 0.25 \* (float) TempData; //Calculates temp from TempData

Temperature = 1.8\*Temperature + 32.0; //Converts to farenheit

return Temperature; //returns temperature variable

} // end of Read\_SPI\_Temperature

void setup()

{

//Setup Chip Select Pin and set as inactive.

pinMode( A2, OUTPUT); //Sets up A2 as OUTPUT

digitalWrite( A2, HIGH ); //Sets A2 to HIGH

SPI.begin(); //begins the SPI

SPI.setClockDivider( SPI\_CLOCK\_DIV2 ); //sets the clock divider up

SPI.setBitOrder( MSBFIRST ); //sets up the master/slave bit

SPI.setDataMode(SPI\_MODE0); //sets the SPI mode

Serial.begin( 9600); //setup serial port; baud rate to 9600

secondTimer = millis(); //sets timer to millis()

}

void loop()

{

//checks if time passed is 1 second

if( millis() - secondTimer >= 1000 )

{

ThermoCouple = Read\_SPI\_Temperature( ); // Read Temperature sensor

secondTimer += 1000; // increments secondTimer

Serial.print( "Temp = "); // report data to serial monitor

Serial.println( ThermoCouple ); // prints the ThermoCouple value

}

}