Compass Sensor Module

Setup

1. Connect one end of the cable into either Molex connectors on the sensor
2. Connect the other end of the cable to the Arduino board:
   - RED: 5V
   - WHITE: I2C SDA (pin A4 on Uno; pin 20 on Mega)
   - BLACK: GND
   - GREY: I2C SCL (pin A5 on Uno; pin 21 on Mega)

Example Sketch

```
// OSEPP Compass Sensor Example Sketch
// by OSEPP <http://www.osepp.com>

// This sketch demonstrates interactions with the Compass Sensor

#include <Wire.h>

// Sensor address (non-configurable)
const uint8_t sensorAddr = 0x1E;

// One-time setup
void setup()
{
   // Start the serial port for output
   Serial.begin(9600);

   // Join the I2C bus as master
   Wire.begin();

   // Configure the compass to default values (see datasheet for details)
   WriteByte(sensorAddr, 0x0, 0x10);
   WriteByte(sensorAddr, 0x1, 0x20);

   // Set compass to continuous-measurement mode (default is single shot)
   WriteByte(sensorAddr, 0x2, 0x0);
}

// Main program loop
void loop()
{
   uint8_t x_msb;   // X-axis most significant byte
   uint8_t x_lsb;   // X-axis least significant byte
   uint8_t y_msb;   // Y-axis most significant byte
   uint8_t y_lsb;   // Y-axis least significant byte
   uint8_t z_msb;   // Z-axis most significant byte
   uint8_t z_lsb;   // Z-axis least significant byte

   int x;
   int y;
   int z;

   // Get the value from the sensor
   if ((ReadByte(sensorAddr, 0x3, &x_msb) == 0) &&
       (ReadByte(sensorAddr, 0x4, &x_lsb) == 0) &&
       (ReadByte(sensorAddr, 0x5, &y_msb) == 0) &&
       (ReadByte(sensorAddr, 0x6, &y_lsb) == 0) &&
       (ReadByte(sensorAddr, 0x7, &z_msb) == 0) &&
       (ReadByte(sensorAddr, 0x8, &z_lsb) == 0))
   {
      x = x_msb << 8 | x_lsb;
      y = y_msb << 8 | y_lsb;
      z = z_msb << 8 | z_lsb;

      Serial.print("X: ");
```

```
      Serial.println(x);
      Serial.print("Y: ");
      Serial.println(y);
      Serial.print("Z: ");
      Serial.println(z);
   }
   else
   {
      Serial.println("Failed to read from sensor");
   }

   // Run again in 1 s (1000 ms)
   delay(1000);
}

// Read a byte on the i2c interface
int ReadByte(uint8_t addr, uint8_t reg, uint8_t *data)
{
   // Do an i2c write to set the register that we want to read from
   Wire.beginTransmission(addr);
   Wire.write(reg);
   Wire.endTransmission();

   // Read a byte from the device
   Wire.requestFrom(addr, (uint8_t)1);
   if (Wire.available())
   {
      *data = Wire.read();
   }
   else
   {
      // Read nothing back
      return -1;
   }

   return 0;
}

// Write a byte on the i2c interface
void WriteByte(uint8_t addr, uint8_t reg, byte data)
{
   // Begin the write sequence
   Wire.beginTransmission(addr);

   // First byte is to set the register pointer
   Wire.write(reg);

   // Write the data byte
   Wire.write(data);

   // End the write sequence; bytes are actually transmitted now
   Wire.endTransmission();
}
```