SENSORSHLD1-EVK-101

Usage Guide

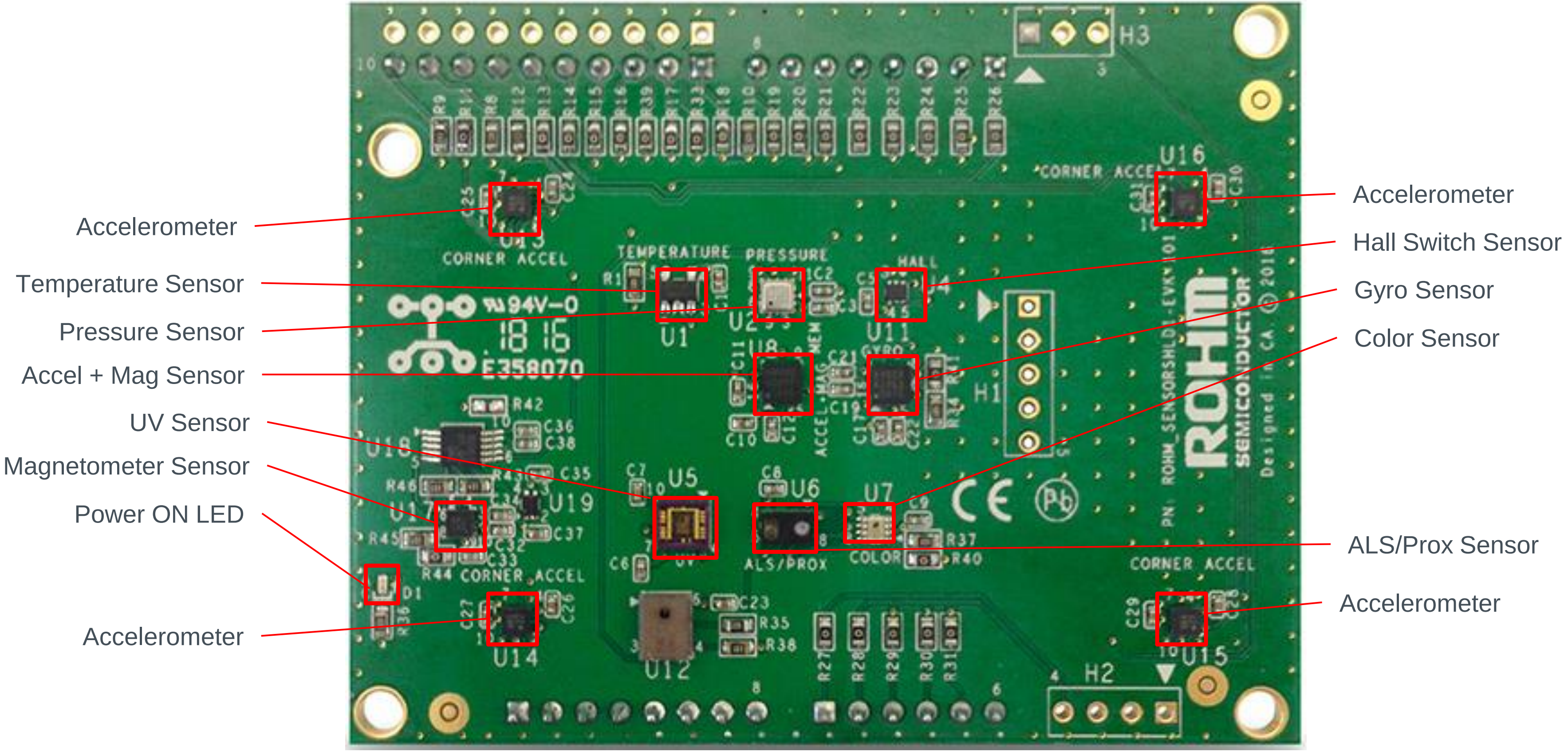
October 26, 2016

Kris Bahar
Applications Engineer

Good References

- ROHM General Page:
  - http://www.rohm.com/web/global/multi-sensor-shield
- Design Files and FW Examples
  - https://github.com/ROHMUSDC/ROHM_SensorPlatform_Multi-Sensor-Shield

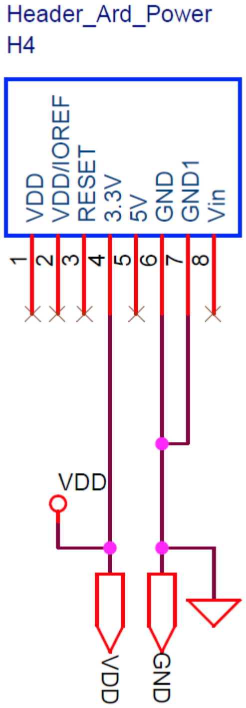
General Info: SENSORSHLD1-EVK-101

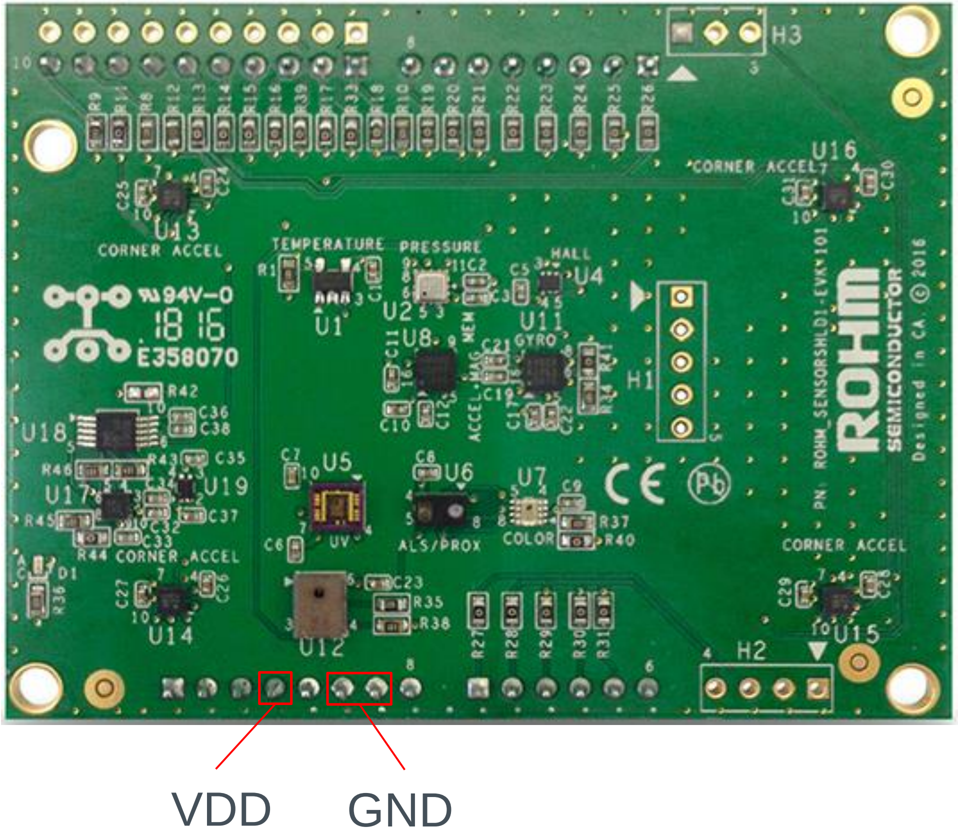


General Info: Board Power

- Power is routed to the board via H4 Header

- NOTE: Be sure that this is connected to 3.3V only; some sensors on board will break due to 5V





Setup: Arduino Recommended Rework

ROHM SEMICONDUCTOR

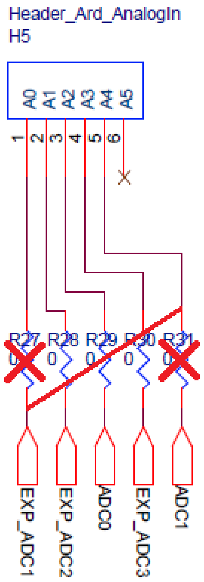
a. On the Arduino UNO board, the I2C pins connected to the top left header are routed to pins A4 and A5 on the bottom right connector.

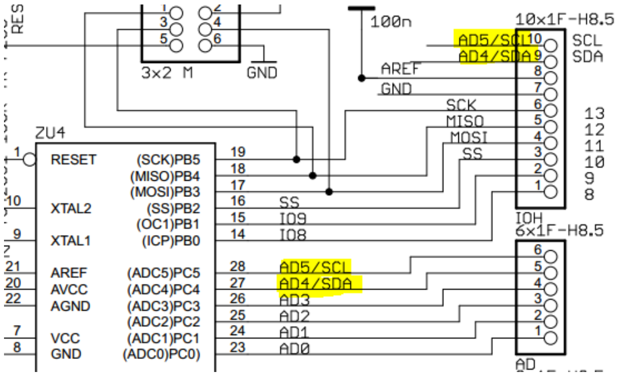
b. This conflicts with the UV sensor's ADC output.

c. Thus, to reroute this on our board, we suggest the following rework…

   i. Disconnect existing nets by removing R27, R31

   ii. Reconnect UV sensor ADC by connecting the top pad of R31 to the bottom pad of R27



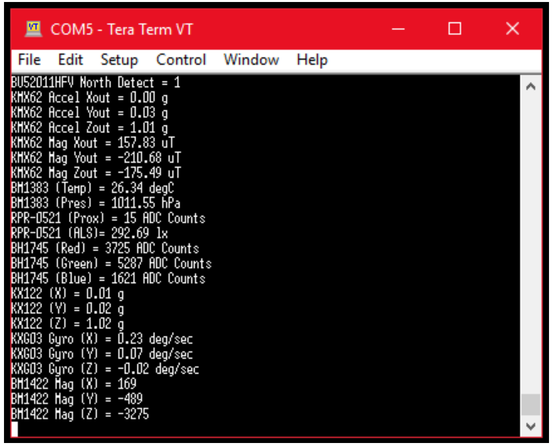


Arduino Uno R3 Schematic

ROHM Shield Rework Recommendation

Setup: Environment Preparation

- Needed: Install the Arduino Software IDE.
  - https://www.arduino.cc/en/Guide/Windows
- Recommended: Install TeraTerm
  - Recommendation since serial monitor is available within the Arduino IDE.
  - However, this includes good functions such as logging display of special ASCI character sets.
  - https://ttssh2.osdn.jp/index.html.en



Setup: Download Code and Open in Arduino IDE

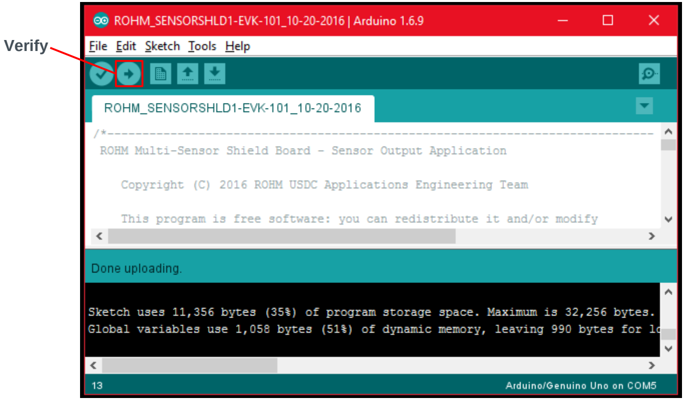
- Download the example sensor shield code found in the GitHub Repository
  - https://github.com/ROHMUSDC/ROHM_SensorPlatform_Multi-Sensor-Shield
    - …/Platform Code/Arduino_UNO_FirmwareExample/

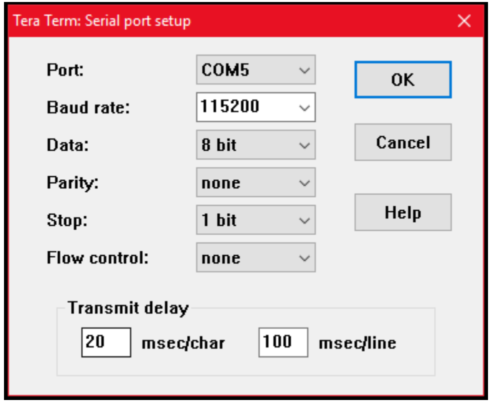
- Open the file named “ROHM_SENSORSHLD1-EVK-101_10-20-2016.ino” in the Arduino IDE

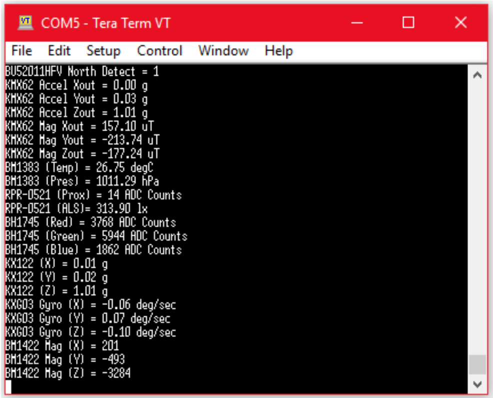
- Install the SoftI2CMaster Library.
  - We can not use the general “wire” library for standard I2C because we want to utilize the repeated start bit of the I2C interface protocol to properly read data back from our sensor devices
  - Instructions on how to install
    - http://playground.arduino.cc/Main/SoftwareI2CLibrary
  - Where to download the library itself
    - https://github.com/felias-fogg/SoftI2CMaster

Setup: Flash Board and View Output

- In order to flash the board, click the “verify” button to compile your code.
- Once this completes and returns with no errors, click the “upload” button
- The dialog box will return with “Done uploading” when complete



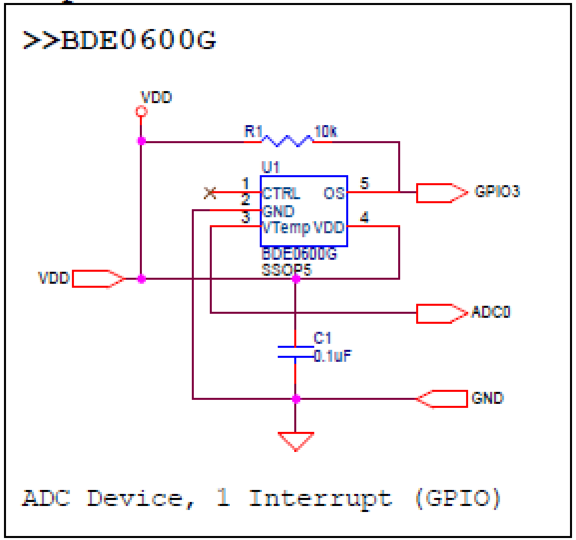




- To view the sensor output, open the serial monitor or Tera Term
- If using Tera Term, configure the Tera Term Serial Port to the above settings (Setup -> Serial Port)
- Then, open a new connection (Alt-N), select the appropriate COM port, and click OK
- Once completed, the connection will open and sensor data will stream to the console

Sensor Explanation: BDE0600G Temp Sensor

Temperature Sensor

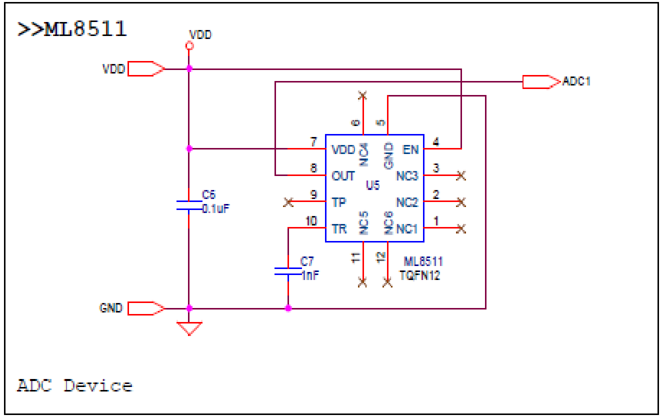


- Typical Use Cases:
  - Temperature Detection
  - Battery Management
  - Heat Management
  - Thermostat Interrupt Detection

- Datasheet:
  - http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/temperature/bdexxx0g-e.pdf
- Communication Interface:
  - Analog Output
  - Thermostat Interrupt (55°C to 115°C in 5°C steps)
- How to Convert:
  - Known Values
    1. Temperature Sensitivity = -10.68mV/degC
    2. Temp Known Point = 1.753V @ 30 degC
  - Calculation
    1. ADC_Voltage = (sensorValue / 670) * 3.3V
    2. ADC_Voltage = sensorValue * 0.004925
    3. Temperature (in deg C) = (ADC_Voltage - 1.753)/(-0.01068) + 30

Sensor Explanation: ML8511 UV Sensor

UV Sensor

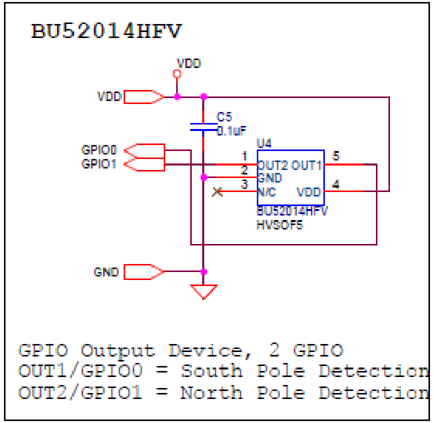


- Datasheet:
  - http://media.digikey.com/pdf/Data%20Sheets/Rohm%20PDFs/ML8511_3-8-13.pdf
- Communication Interface:
  - Analog Output
- What does this measure?
  - UV intensity in mW/cm^2
- How to Convert:
  - Known Values
    1. UV Sensitivity = 0.129 $V/(mW/cm^2)$
    2. Temp Known Point = 2.2V @ $10mW/cm^2$
  - Calculation
    1. ADC_Voltage = (sensorValue / 670) * 3.3V
    2. ADC_Voltage = sensorValue * 0.004925
    3. UV Intensity (in mW/cm^2) = (ADC_Voltage - 2.2)/(0.129) + 10

- Typical Use Cases:
  - Outdoor Detection
  - UV Index Approximation

Sensor Explanation: BU52014 Hall Sensor

Hall Sensor



- Typical Use Cases:
  - Contactless Switch
  - Tablet/Phone Case

- Datasheet:
  - http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/hall/bu52004gul-e.pdf
- Communication Interface:
  - Digital Logic Output (H/L)
- What does this measure?
  - Magnetic Flux Presence
- How to Interpret:
  - OUT1/GPIO0 is tied to South Pole Detection Pin
    - Pin Output LOW on South Detect
  - OUT2/GPIO1 is tied to North Pole Detection Pin
    - Pin Output LOW on North Detect

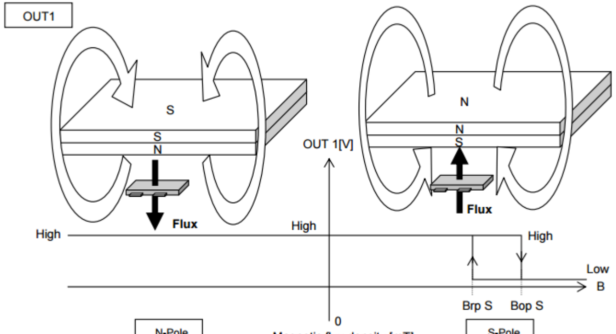


Fig.23 S-Pole Detection

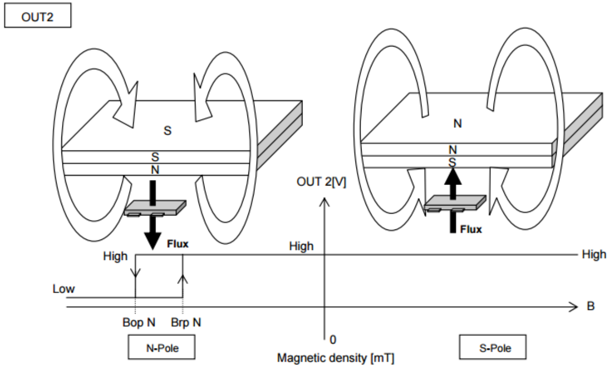
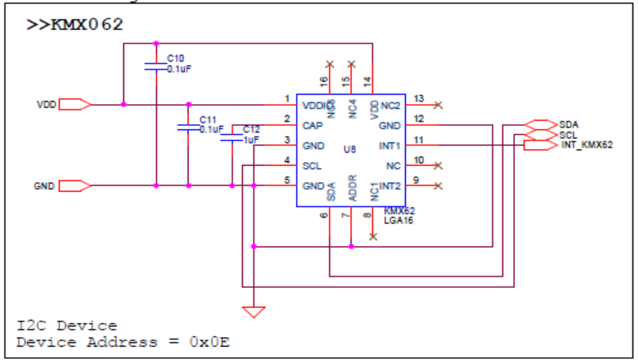


Fig.24 N-Pole Detection

Sensor Explanation: KMX62 Accel + Mag Sensor

Accel + Mag MEM Sensors

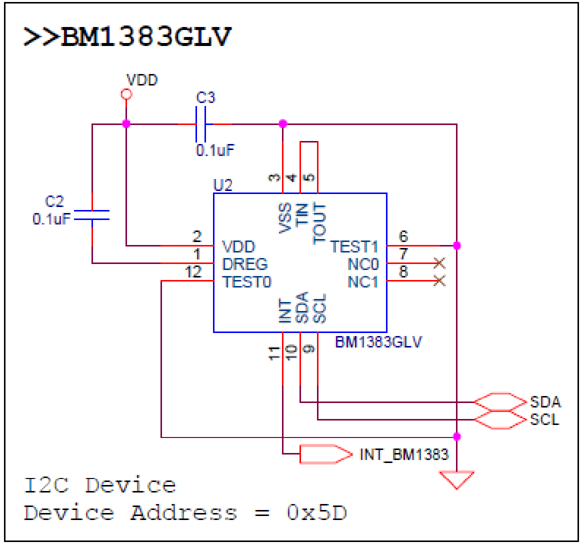


- Typical Use Cases:
  - Low Power Gyro
  - Compass Orientation

- Datasheet:
  - http://kionixfs.kionix.com/en/datasheet/KMX62-1031%20Specifications%20Rev%203.0.pdf
- Communication Interface:
  - I2C
- What does this measure?
  - Acceleration, 3 Axis, 14bit, ±16g
  - Magnetic Flux, 3 Axis, 14bit, ±1200µT
- How to Interpret
  - Perform 12 Bytes I2C Data Read
  - Format 12 Bytes of Data into the six 14bit raw sensor output (Accel X/Y/Z, Mag X/Y/Z)
  - Convert Using Appropriate Scheme
    - ACCEL = Axis_ValueInG = MEMS_Accel_axis / 2048
    - MAG = Axis_ValueInuT = MEMS_Mag_axis / 0.146

Sensor Explanation: BM1383 Pressure Sensor

Pressure Sensor



- Typical Use Cases:
  - Height/Altitude Detection

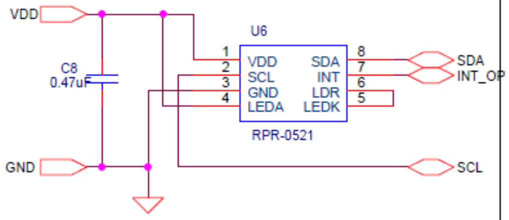
- Datasheet:
  - http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/pressure/bm1383glv-e.pdf
- Communication Interface:
  - I2C
- What does this Measure?
  - Barometric Pressure, 16bit + 5bit decimal, 300hPa to 1100hPa
- How to Convert:
  - Perform 3 Bytes I2C Data Read
  - Format 3 Bytes of Data into the pressure output conversion formula
    - Pressure Value[hPa] = { PRESS_OUT[15:8], PRESS_OUT[7:0], PRESS_OUT_XL[5:0] } / 2048

Sensor Explanation: RPR-0521 ALS+PROX Sensor

ROHM SEMICONDUCTOR

ALS/Proximity Sensor

>>RPR-0521



I2C Device
Device Address = 0x38
INT - Open Drain

- Typical Use Cases:
  - Mobile Phones (brightness optimization and ear detect)
  - Contactless Switch/Presence Detection

- Datasheet:
  - http://www.rohm.com/web/global/datasheet/RPR-0521RS
- Communication Interface:
  - I2C
- What does this measure?
  - Ambient Light, 0 to 43000 Lx
  - Proximity Detection, within 100mm maximum
- How to Convert:
  - Perform 6 Bytes I2C Data Read (4 for ALS, 2 for Prox)
  - Format 4 Bytes of Data into 16bit values for DATA0 (Visible Light) and DATA1 (Infrared Light)
  - Format 2 Bytes of Data into 16bit value for Proximity ADC Counts
  - Perform Conversion of Raw Data to Lx (see next slide)

ALS Sensor Conversion Equation

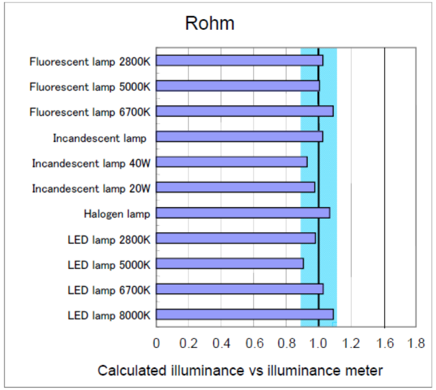
Q: Why does this equation look so complex?

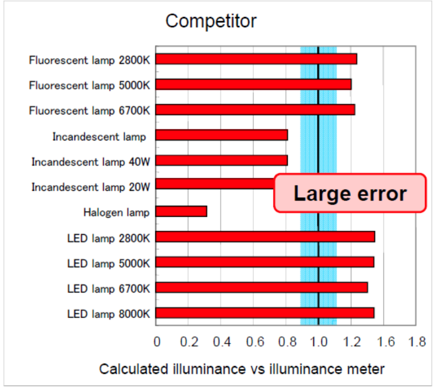
A: This ALS sensor utilizes 2 photodiodes, one for visible light and one for ambient light.

This ratio based piecewise function allows us to properly reject high IR based output and generate a good lux output based on the actual visible light spectrum

```
RPR0521_ALS_DataRatio = (float) RPR0521_ALS_D1_RAWOUT /
(float)RPR0521_ALS_D0_RAWOUT;

if(RPR0521_ALS_DataRatio < 0.595){
	RPR0521_ALS_OUT = (1.682*(float) RPR0521_ALS_D0_RAWOUT - 1.877*(float)
	RPR0521_ALS_D1_RAWOUT);
}
else if(RPR0521_ALS_DataRatio < 1.015){
	RPR0521_ALS_OUT = (0.644*(float) RPR0521_ALS_D0_RAWOUT - 0.132*(float)
	RPR0521_ALS_D1_RAWOUT);
}
else if(RPR0521_ALS_DataRatio < 1.352){
	RPR0521_ALS_OUT = (0.756*(float) RPR0521_ALS_D0_RAWOUT - 0.243*(float)
	RPR0521_ALS_D1_RAWOUT);
}
else if(RPR0521_ALS_DataRatio < 3.053){
	RPR0521_ALS_OUT = (0.766*(float) RPR0521_ALS_D0_RAWOUT - 0.25*(float)
	RPR0521_ALS_D1_RAWOUT);
}
else{
	RPR0521_ALS_OUT = 0;
}
```

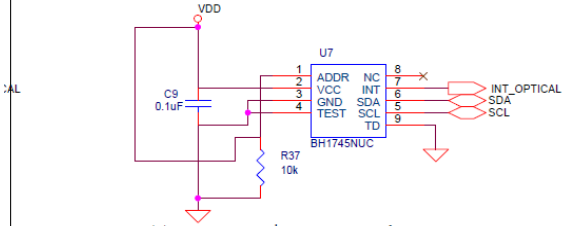




Sensor Explanation: BH1745NUC Color Sensor

Color Sensor

>>BH1745NUC



**One Int Pin - To Ard HDR

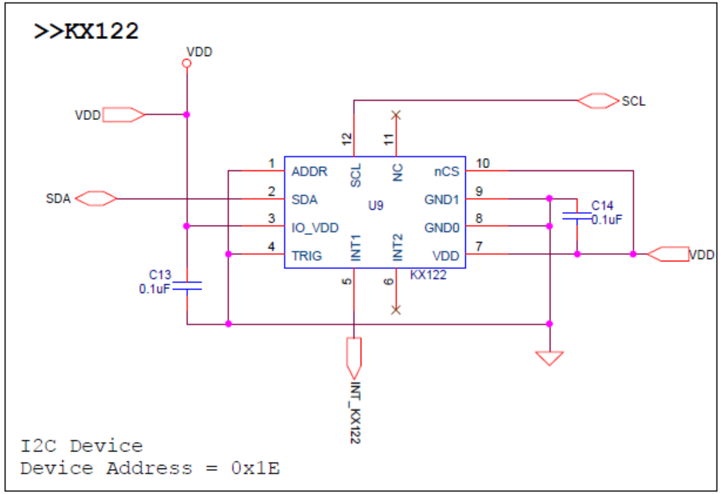
I2C Device
Use Jumper to select Addr Pin
Device Addr = 0x39
INT Open Drain

- Typical Use Cases:
  - Color Balancing (Lighting)
  - Industrial Machinery (Defect Detection)

- Datasheet:
  - http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/light/bh1745nuc-e.pdf
- Communication Interface:
  - I2C
- What does this measure:
  - Ambient Light, 0.005 to 40k lx
  - Individual photo diode ADC output for Red, Green, and Blue output
- How to Convert:
  - Perform 6 Bytes I2C Data Read (2 Bytes for each color: Red, Green, Blue)
  - Format 2 Bytes of Data into 16bit values for RED, GREEN, and BLUE intensity values

Sensor Explanation: KX122 Accelerometer Sensor

Accel MEM Sensors

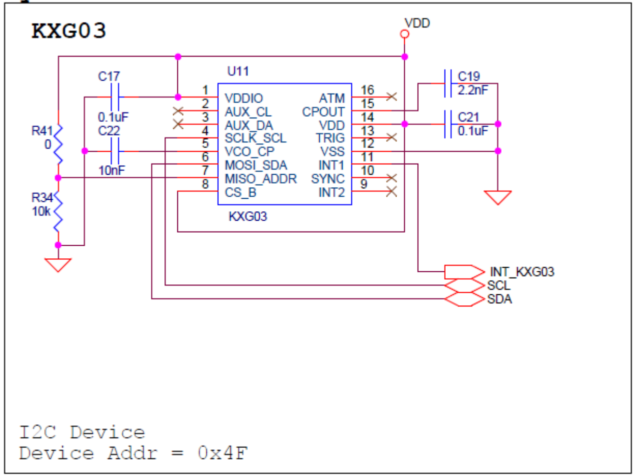


- Typical Use Cases:
  - Built-in Features (Tap Detection, Activity Detection, Free Fall engine, etc.)
  - Low Power Wearable Devices (0.9uA standby, 10uA low resolution mode)

- Datasheet:
  - http://kionixfs.kionix.com/en/datasheet/KX122-1037%20Specifications%20Rev%204.0.pdf
- Communication Interface:
  - I2C
- What does this measure:
  - Acceleration, 3 Axis, 16bit, ±8g
  - Built-in features like Tap/Double Tap Detection, Activity Detection, Free Fall Detection, etc.
- How to Convert:
  - Perform 6 Byte I2C Data Read
  - Format 6 Bytes of Data into the six 16bit raw sensor output (Accel X/Y/Z)
  - Convert Using Appropriate Scheme
    - ACCEL = Axis_ValueInG = MEMS_Accel_axis / 16384

Sensor Explanation: KXG03 Accel + Gyro Sensor

Gyro

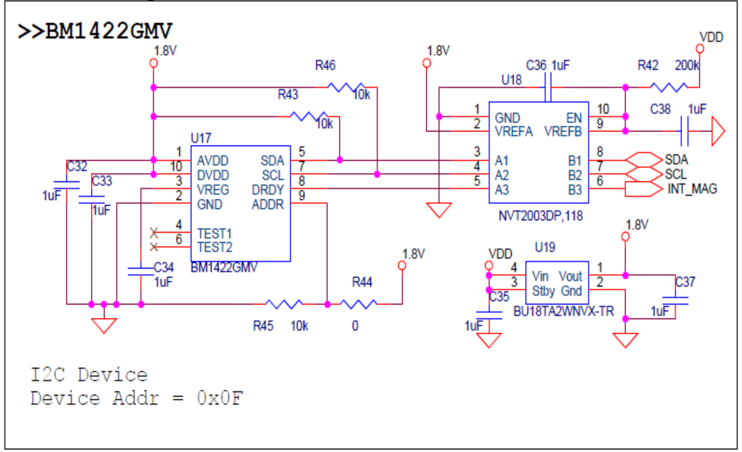


- Typical Use Cases:
  - 6 Axis Motion Sensor (Sensor Fusion via Complimentary Filtering or Kalman Filter)
  - 9 Axis Motion Sensor by adding Magnetometer Sensor
  - Motion Controller (Wii Controller), or Motion Feedback (Quadcopter)

- Datasheet:
  - http://kionixfs.kionix.com/en/datasheet/KXG03%20Specifications%20Rev%201.0.pdf
- Communication Interface:
  - I2C
- What does this measure?
  - Acceleration, 3 Axis, 16bit, ±16g
  - Angular Velocity, 3 Axis, 16bit, ±2048°/sec
- How to convert?
  - Initialize Gyro: take readings when Gyro is not moving; save this value as Gyro Offset (RAW value OK)
  - Perform 12 Byte I2C Data Read
  - Format 12 Bytes into the six 16bit raw sensor output (Accel X/Y/Z, Gyro X/Y/Z)
  - Convert Using Appropriate Scheme
    - ACCEL = Axis_ValueInG = MEMS_Accel_axis / 16384 (±2g)
    - GYRO = Axis_ValueinDegPerSec = MEMS_Gyro_axis * 0.007813

Sensor Explanation: BM1422 Magnetometer Sensor

3-Axis Magnetometer



- Typical Use Cases:
  - 9 Axis Motion Sensor (by adding Accel+Gyro Sensor)
  - E-compass
  - Contactless Current Sensor

- Datasheet:
  - https://github.com/ROHMUSDC/ROHM_Sensor Platform_Multi-Sensor-Shield/blob/master/Documentation/Sensor%20Datasheets/ROHM_MAG-IND_BM1422GMV.pdf
- Communication Interface:
  - I2C
- What does this measure?
  - Magnetic Flux, 3 Axis, 16bit, ±1200µT, 0.042µT/LSB
- How to Convert:
  - Perform 6 Byte I2C Data Read
  - Format 6 Bytes of Data into the three 16bit raw sensor output (Mag X/Y/Z)
  - Convert Using Appropriate Scheme
    - MAG = Axis_ValueInuT = Mag_axis_RawOut * 0.042

ROHM Semiconductor U.S.A., LLC