GENERAL DESCRIPTION

The MC3635 is an ultra-low power, low noise, integrated digital output 3-axis accelerometer with a feature set optimized for wearables and the Internet of Moving Things (IoMT) devices. With its low power modes and tiny footprint this accelerometer extends sensor battery life and reduces the space required for the sensor by up to 3X. Low noise and low power are inherent in the monolithic fabrication approach.

The EV3635A is a prebuilt circuit board with the MC3635 accelerometer sensor, with an internal sample rate from 14 to 1300 samples/second. It measures acceleration with a wide usage range, from +/-2g up to +/-16g. It offers 8-bit to 14-bit high-precision ADC output which is easy mounted onto a microcontroller platform, such as Arduino. The accelerometer communicates via I2C (or SPI) and gives out motion detection or sample acquisition conditions to trigger an interrupt toward a MCU.

Sensor data is easily read by simply connecting DVDD to 3.3V, GND to ground, and SPI (DOUT, SCK, SCL) pins to your Arduino SPI (MISO, MOSI, CLK) respectively. Download the MC3635 library from mCube's website onto the board, run the example sketch, and then sensor data shortly comes out in raw data count and SI unit accelerometer measurements.

MC3635 FEATURES

Range, Sampling & Power

- ±2,4,8,12 or 16g ranges
- 8, 10 or 12-bit resolution with FIFO
  - 14-bit single samples
- 14 - 1300 samples/sec
  - Sample trigger via internal oscillator, clock pin or software command
- Sniff and Wake Modes
  - 0.4 µA Sniff current @ 6Hz
- Ultra-Low Power with FIFO
  - 0.9 µA typical current @ 25Hz
  - 1.6 µA typical current @ 50Hz
  - 2.8 µA typical current @ 100Hz
  - 36 µA typical current @ 1300Hz

Simple System Integration

- I2C interface, up to 1 MHz
- SPI Interface, up to 4 MHz
- 1.6 × 1.6 × 0.94 mm 12-pin package
- Single-chip 3D silicon MEMS
- Low noise down to 2.3mgRMS

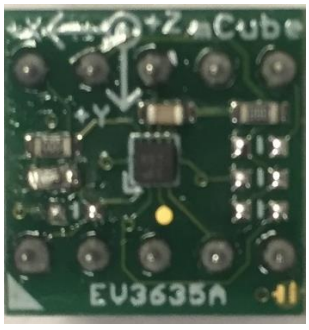
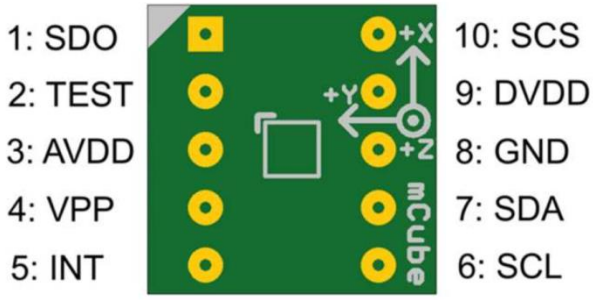


TABLE OF CONTENTS

1 GENERAL OPERATION

1.1 PINOUTS



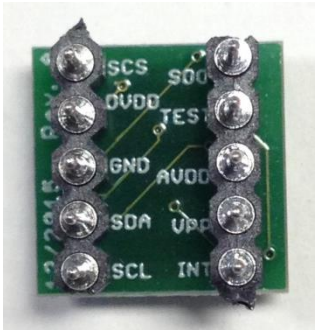


Diagram View from front	Labels on back of device (note: pins are flipped because view is of the back)

Note: for normal use, the current measurement jumper should we closed. Otherwise, it can be used to measure current. Please refer to Section 2.1 for more information on current measurement tests.

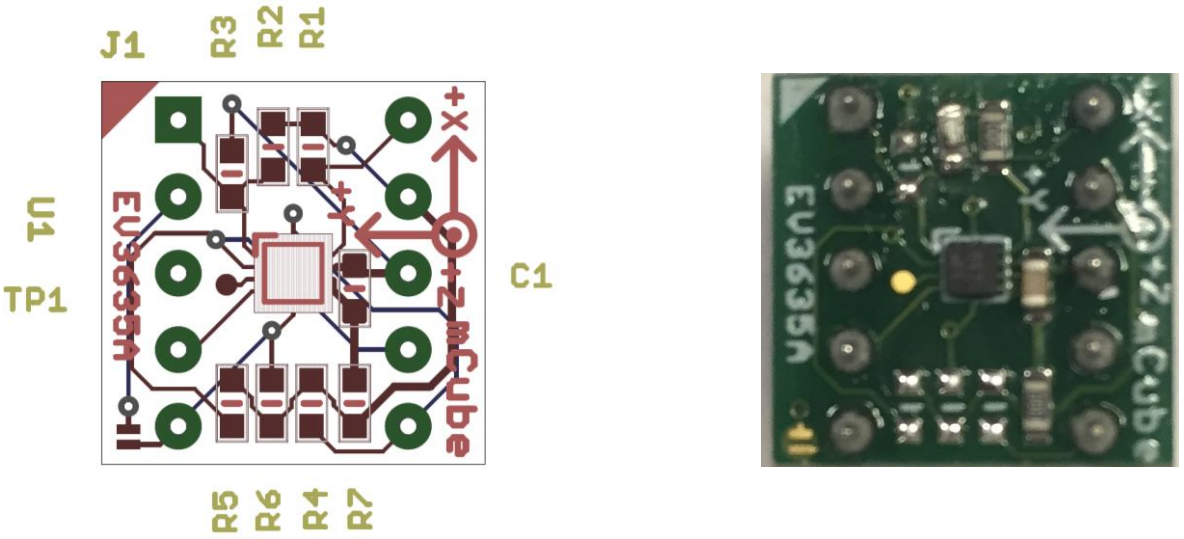
1.2 POWER PINS

- **DVDD** – 1.8V/3.3V Power Supply Input
- **GND** – Ground Pin for Power and Logic

For use with an Arduino, a small modification must be made to the Arduino for it to output at a 3.3V level. By default, Arduinos operate at 5V, which is higher than the maximum voltage rating for the MC3635 at 3.7V.

An excellent tutorial on modifying an Arduino for use at 3.3V is located at:

https://learn.adafruit.com/arduino-tips-tricks-and-techniques/3-3v-conversion



1.3 SPI PINS

To setup using the SPI protocol, connect the following pins. Note that factory preset configuration of the EV3635A evaluation board is for SPI; no additional rework is required to start communicating with the device. See Section 7 for more information.

- Connect the **SCL** (I2C clock pin) to your microcontroller's SPI clock line.
- Connect the **SDA** (I2C data pin) to your microcontroller's SPI MOSI line.
- Connect the **SDO** SPI pin to your microcontroller's SPI MISO line
- Connect the **CSN** SPI pin to your microcontroller's CSN SPI line

Use SPI Mode : CPOL = 1, CPHA = 1, MSB first.

1.4 I2C PINS

To setup using the I2C protocol, connect the following pins. Note that factory preset configuration of the EV3635A evaluation board is for SPI, not I2C. To reconfigure for I2C, rework is needed. See Section 7 for more information.

- Connect the **SCL** (I2C clock pin) to your microcontroller's I2C clock line.
- Connect the **SDA** (I2C data pin) to your microcontroller's I2C data line.

NOTE: DO NOT install more than one setup pull-up resistors per I2C bus.

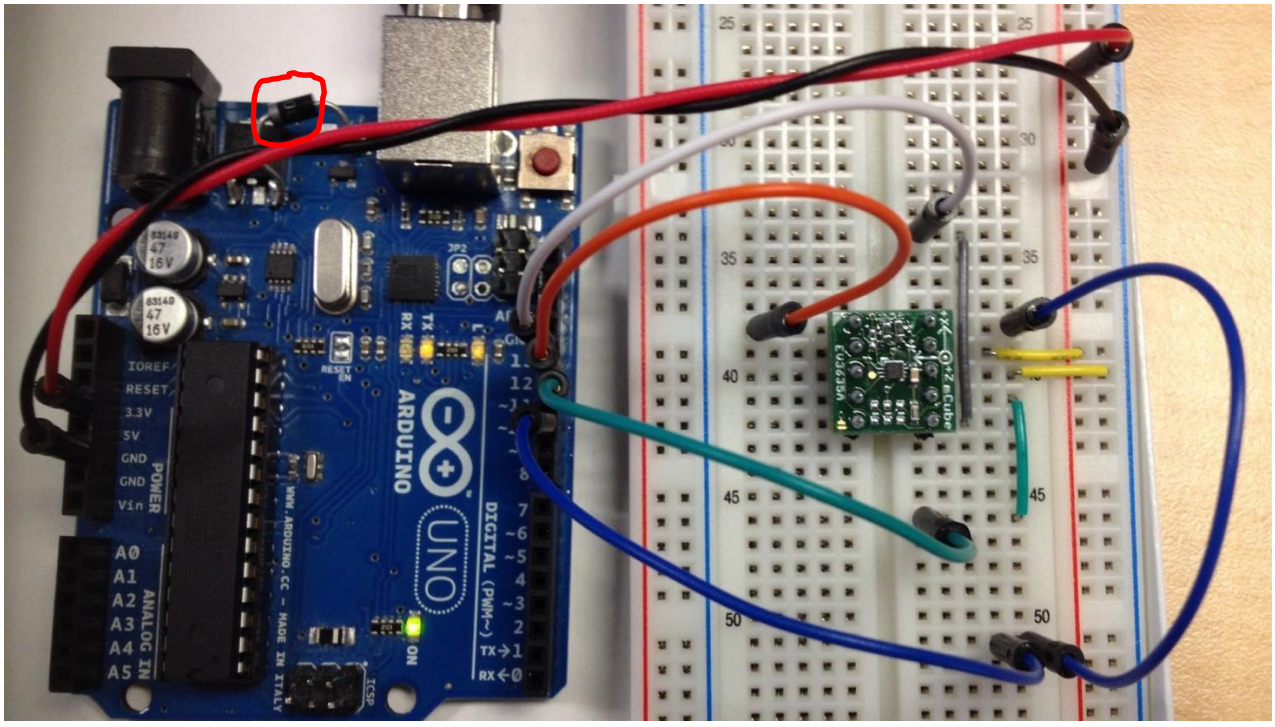
1.5 INTERRUPT PINS

INTn - HW interrupt signal pin. It will be driven by the chip when data is ready to read, or a motion event is detected by the accelerometer. (Not currently supported in the library for the interrupt pin, so please check the datasheet for the commands toward related registers).

2 ASSEMBLY AND TEST

You can easily wire this breakout to any microcontroller; we'll be using an Arduino. For another kind of microcontroller, just make sure it has compatible with SPI, then port the code.

- **DVDD** – 3.3V Power Supply Input (See Section 1.2 to modify Arduino to output at 3.3V instead of 5V. **WARNING**: attempting to power the part at 5V is likely to damage the part). See red circle in below figure.
- **GND** – Ground Pin for Power and Logic
- Connect the **SCL** (I2C clock pin) to your microcontroller's SPI clock line.
- Connect the **SDA** (I2C data pin) to your microcontroller's SPI MOSI line.
- Connect the **SDO** SPI pin to your microcontroller's SPI MISO line
- Connect the **CSN** SPI pin to your microcontroller's CSN SPI line



3 DEMO

3.1 DOWNLOAD THE DRIVER FROM GITHUB

To begin reading sensor data, you will need to download the MC3635_Library from MCube website. You can do that by visiting the website and navigating to the part (where the datasheet and other information is located), or by simply clicking this link to the attached URL to download the zip file.

http://www.mcubemems.com/wp-content/uploads/2016/08/Accelerometer_MC3635.zip

Rename the uncompressed folder **Accelerometer_MC3635** and check that the Accelerometer_MC3635 folder contains **MC3635.cpp** and **MC3635.h**

Place the Accelerometer_MC3635 library folder to your **Arduino_sketch_folder/libraries/** folder.

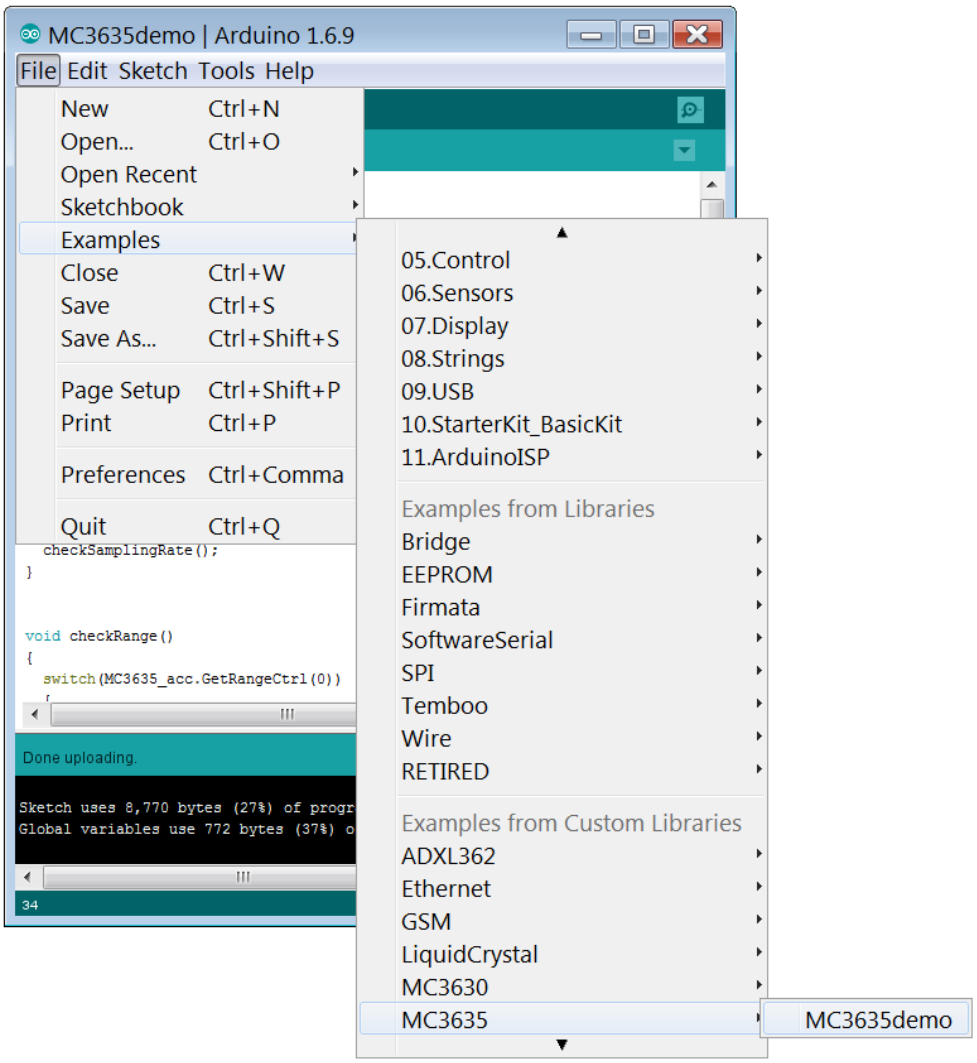
You may need to create the library subfolder if it is your first library. Then just restart the IDE.

An excellent tutorial on Arduino library installation is located at:

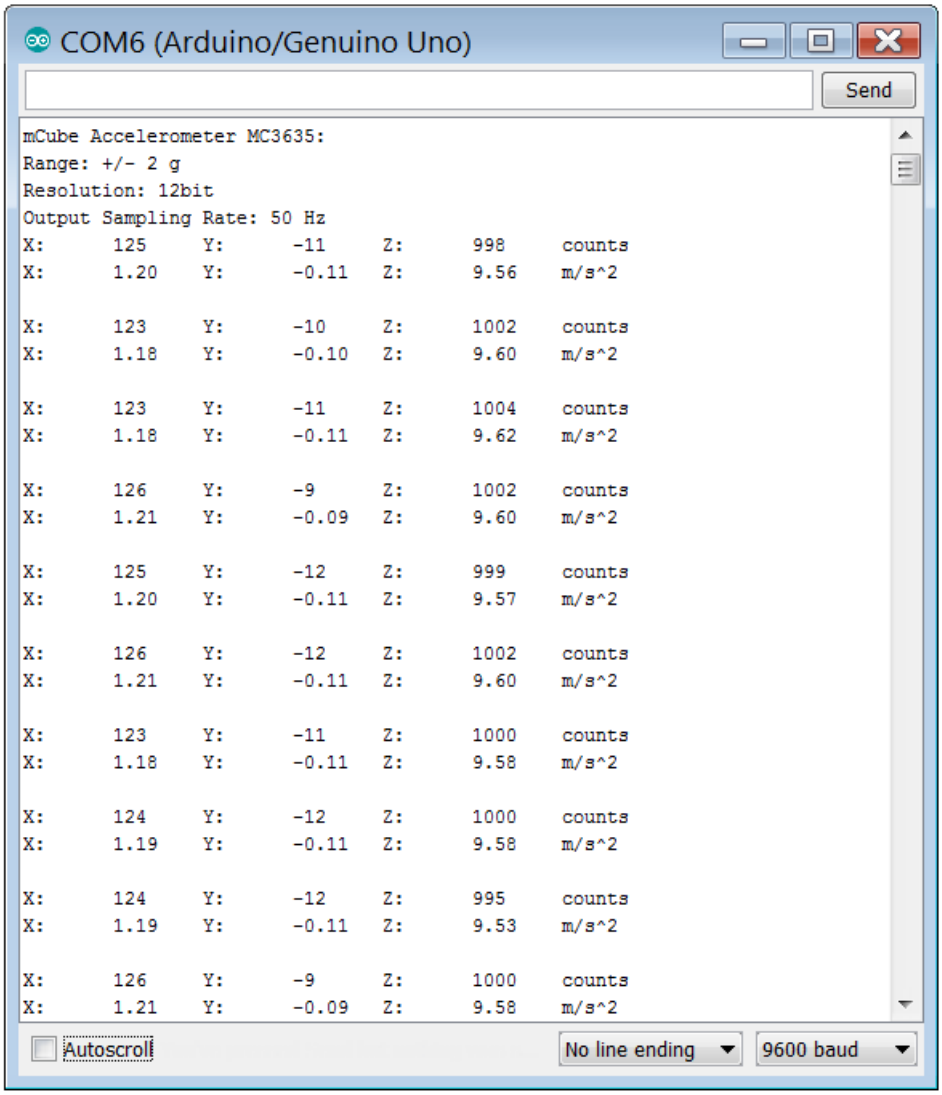
http://learn.adafruit.com/adafruit-all-about-arduino-libraries-install-use

3.2 LOAD THE DEMO

Open up File->Examples->MC3635-> MC3635demo and upload to your Arduino wired up to the sensor



Now open up the serial terminal window at 9600 baud rate speed to begin the test.



You will see the output from the serial terminal showing the current range scale and resolution of the sensor in the first three lines followed by two lines of output sensor data at some output data rate which depict “raw count" data for line 8: X: 123 Y: -10 Z: 1004 with 8G range, 14bit ADC resolution.

Line 9 indicates the SI units for measuring acceleration as X: 1.18 m/s^2 Y: -0.10 m/s^2 Z: 9.60 m/s^2.

4 LIBRARY REFERENCE

4.1 CREATE MCUBE_MC3635 OBJECT

You can create the MCUBE_MC3635 object with:

```
MC3635 MC3635_acc = MC3635();
```

4.2 INITIALIZE AND CONFIGURE SENSOR

Initialize and configure the sensor with:

```
MC3635_acc.start();
```

4.3 SET RANGE

Set the accelerometer max range to ±2g, ±4g, ±8g or ±16g with:

```
MC3635_acc.SetRangeCtrl(MC3635_RANGE_2G);
MC3635_acc.SetRangeCtrl(MC3635_RANGE_4G);
MC3635_acc.SetRangeCtrl(MC3635_RANGE_8G);
MC3635_acc.SetRangeCtrl(MC3635_RANGE_16G);
```

4.4 READ RANGE

Read the current range with:

```
MC3635_acc.GetRangeCtrl();
```

This returns: 0 for ±2g, | 1 for ±4g, | 2 for ±8g | 3 for ±16g.

4.5 READ RESOLUTION

Read the current resolution with:

```
MC3635_acc.GetResolutionCtrl();
```

This returns: 0 for 6-bit | 1 for 7-bit | 2 for 8-bit | 3 for 10-bit | 4 for 12-bit | 5 for 14-bit

4.6 READ RAW COUNT DATA

Read the raw count data and SI unit measurement with:

```
MC3635_acc.readRawAccel();
```

5 DOWNLOADS

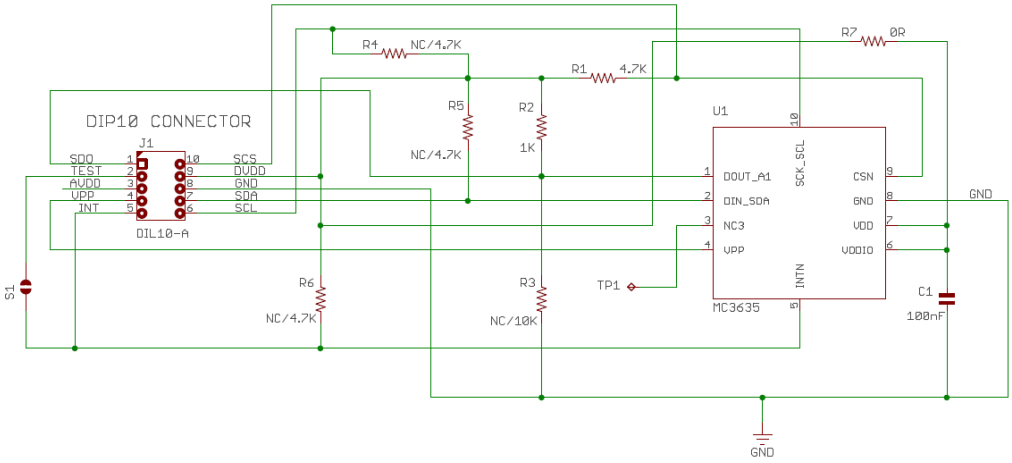
5.1 MC3635 ACCELEROMETER DATASHEET

http://www.mcubemems.com/product/MC3635-3-axis-accelerometer/

5.2 ALL OTHER MCUBE DOCUMENTATION

http://www.mcubemems.com/resources-support/resources/

6 SCHEMATICS



Above is a diagram showing setup for SPI protocol. This is the factory preset when receiving the part.

For other options, please refer to the following table:

Interface	R1	R3	R2
SPI @ 1Mhz (Factory default)	4.7KΩ to 100KΩ	DNI	4.7KΩ
I2C 0x6C or SPI @ 2MHz	4.7KΩ	DNI	1KΩ
I2C 0x4C (0x98)	4.7KΩ to 100KΩ	10KΩ	DNI
I2C 0x6C (0xD8)	4.7KΩ to 100KΩ	DNI	10KΩ

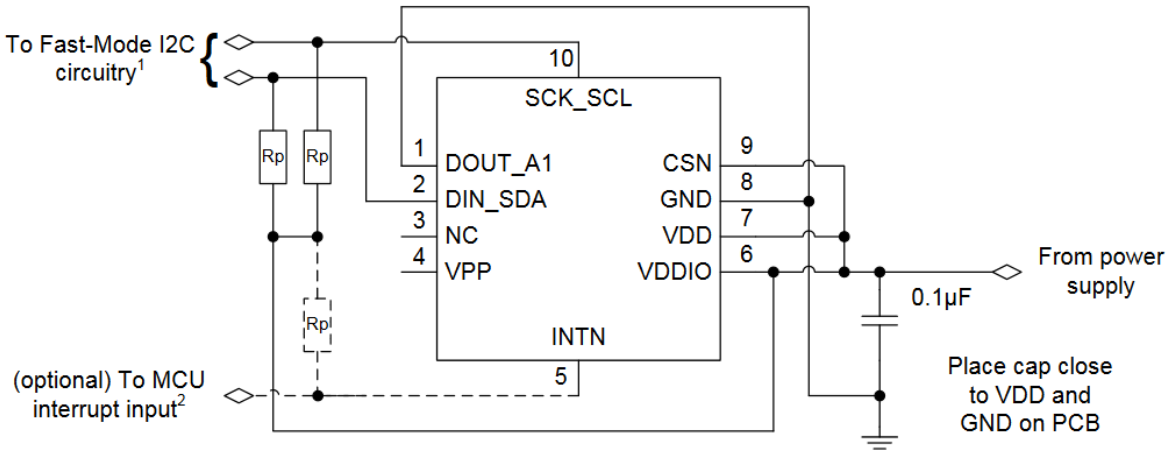
DNI = Do not install

R6: If using the sensor interrupt signal as open-drain, then install pull-up resistor ~4.7KΩ into R6 (not installed by default).

R4, R5: If using I2C and I2C pull-up resistors are needed for your application then install ~4.7KΩ resistors into R4 and R5 (not installed by factory default). NOTE: DO NOT install more than one set up pull-up resistors per I2C bus.

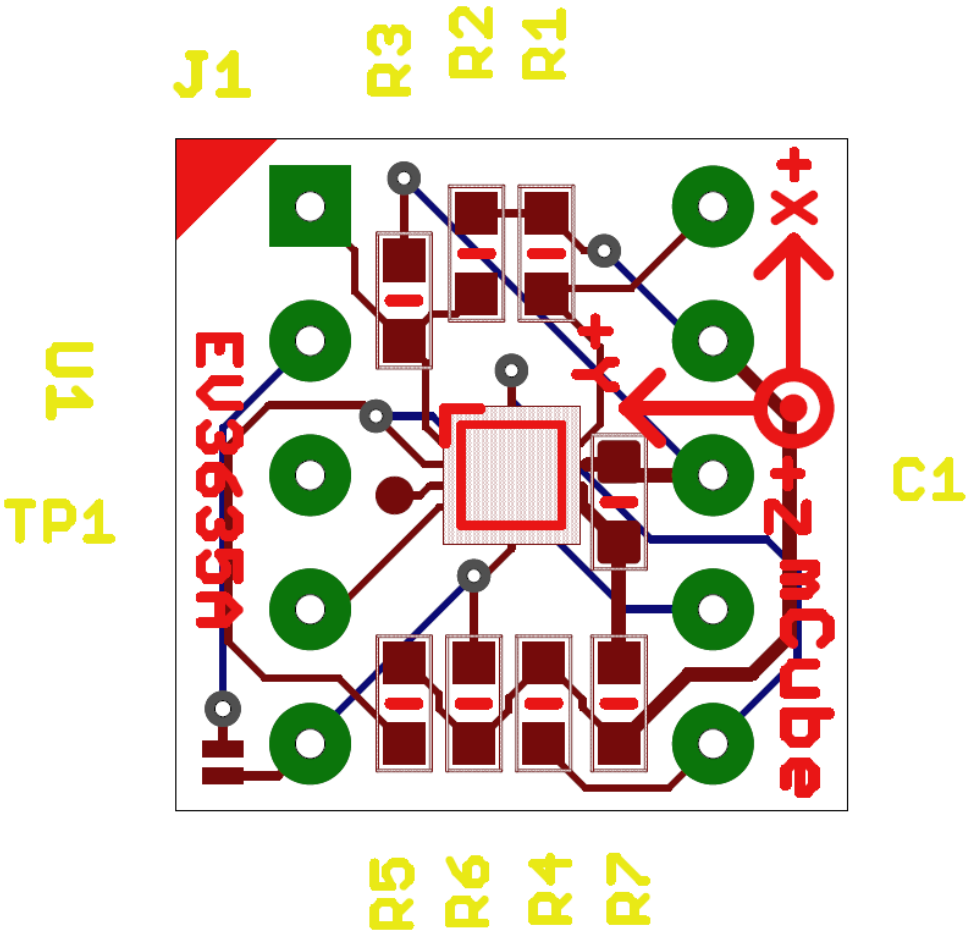
R7: The current drawn the sensor can be measured by putting an ammeter in place of R7.

The following is a schematic detailing how to set up the accelerometer for I2C protocol. Set R7 = 0. R1 =0, R4 = 4.7k, R5 = 4.7k. The physical location of the resistor is in the diagram in Section 7.



NOTE[1]: Rp are typically 4.7kΩ pullup resistors to VDDIO, per I2C specification. When VDDIO is powered down, DIN_SDA and SCK_SCL will be driven low by internal ESD diodes.
NOTE[2]: Attach typical 4.7kΩ pullup resistor if INTN is defined as open-drain.

7 ASSEMBLY DRAWING



8 REVISION HISTORY

Date	Revision	Description
2016-08	APS-0xx-00xxv1.0	First release.

9 LEGAL

1. M-CUBE reserves the right to make corrections, modifications, enhancements, improvements and other changes to its products and to this document at any time and discontinue any product without notice. The information contained in this document has been carefully checked and is believed to be accurate. However, M-CUBE shall assume no responsibilities for inaccuracies and make no commitment to update or to keep current the information contained in this document.

2. M-CUBE products are designed only for commercial and normal industrial applications and are not suitable for other purposes, such as: medical life support equipment; nuclear facilities; critical care equipment; military / aerospace; automotive; security or any other applications, the failure of which could lead to death, personal injury or environmental or property damage. Use of the products in unsuitable applications are at the customer's own risk and expense.

3. M-CUBE shall assume no liability for incidental, consequential or special damages or injury that may result from misapplication or improper use of operation of the product.

4. No license, express or implied, by estoppel or otherwise, to any intellectual property rights of M-CUBE or any third party is granted under this document.

5. M-CUBE makes no warranty or representation of non-infringement of intellectual property rights of any third party with respect to the products. M-CUBE specifically excludes any liability to the customers or any third party regarding infringement of any intellectual property rights, including the patent, copyright, trademark or trade secret rights of any third party, relating to any combination, machine, or process in which the M-CUBE products are used.

6. Examples of use described herein are provided solely to guide use of M-CUBE products and merely indicate targeted characteristics, performance and applications of products. M-CUBE shall assume no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein

7. Information described in this document including parameters, application circuits and its constants and calculation formulas, programs and control procedures are provided for the purpose of explaining typical operation and usage. "Typical" parameters that may be provided in M-CUBE data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters including "Typicals," must be validated for each customer application by customer's technical experts. In no event shall the information described be regarded as a guarantee of conditions or characteristics of the products. Therefore, the customer should evaluate the design sufficiently as whole system under the consideration of various external or environmental conditions and determine their application at the customer's own risk. M-CUBE shall assume no responsibility or liability for claims, damages, costs and expenses caused by the customer or any third party, owing to the use of the above information.

is a trademark of M-CUBE, Inc.

M-CUBE and the M-CUBE logo are trademarks of M-CUBE, Inc.,

All other product or service names are the property of their respective owners.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

mCube:
EV3635A