नीति आयोग
National Institution for Transforming India

ATAL INNOVATION MISSION

PHYSICAL COMPUTING

GETTING STARTED GUIDE - PART 3

I'LL MAKE UBIQUITY

I'll make ubiquity is the beginners guide developed and created to assist students to understand and work with technology.

It contains a step wise introduction into one of the technology platforms - Genuino 101.

We urge you to learn & understand how to work with one platform and then go out and explore further.

WHAT'S INSIDE?

1 Tutorials

- Getting Started
- Controlling LED
- Fairy Lights
- Push Button

2 Hands on

- Accelerometer
- Gyroscope

3 Exercise

- Bluetooth works
- Pedometer
- Visual Programming

HANDS ON

ACCELEROMETER

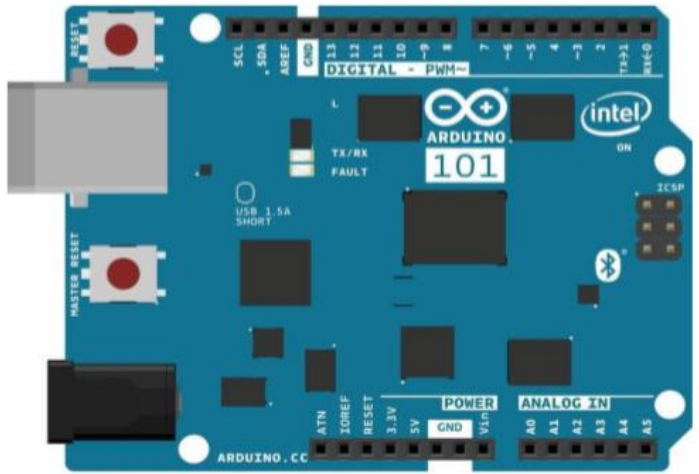
THE IDEA

The idea is to read the three axes of the accelerometer contained in the IMU (Inertial Measurement Unit) of the Genuino 101 board.

Each axis measures the acceleration within a range defined by a specific function and returns a raw value that needs to be con- verted to get a value in mg. The result of the conversion is print- ed on the Serial monitor as triplets of acceleration values (X, Y and Z).

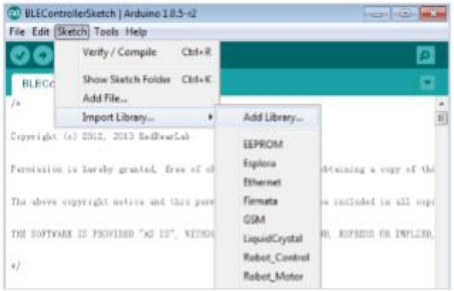
Hardware

No additional hardware is required apart from the Genuino board.



SOFTWARE

Libraries



CurieIMU.h is the library that gives access to all the parameters, features and readings of the IMU chip of the Genuino board.

This unit contains a three axes accelerometer and a three axes gyroscope. This library is part of the Genuino board core and it is loaded together with the core files for Arduino or Genuino 101. In this tutorial we read the raw accelerometer values.

Functions

float convertRawAcceleration(int aRaw) - transforms the raw data read from the accelerometer (aRaw) into a value expressed in mg (thousandths of g). The formula of the function must be adjusted to match the accelerometer range set with setAcceler-ometerRange.

CODE

```
/*Copyright (c) 2015 Intel Corporation.  All rights reserved.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser
General Public License as published by the Free Software Foundation; either version 2.1 of the License,
or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without
even the implied
warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PUR-
POSE.  See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth
Floor, Boston, MA
02110-1301  USA

*/
/*This sketch example demonstrates how the BMI160 on the Intel(R) Curie(TM) module can be used to
read accelerometer data */
```

code continue on next page

```
#include "CurieIMU.h"

void setup() {
  Serial.begin(9600); // initialize Serial communication while (!Serial);       // wait
  for the serial port to open

  // initialize device Serial.println("Initializing IMU device...");
  CurieIMU.begin();

  // Set the accelerometer range to 2G
  CurieIMU.setAccelerometerRange(2);
}

void loop() {                                          // raw accelerometer values
  int axRaw, ayRaw, azRaw; float ax, ay, az;

  // read raw accelerometer measurements from device
  CurieIMU.readAccelerometer(axRaw, ayRaw, azRaw);

  // convert the raw accelerometer data to G's ax =
  convertRawAcceleration(axRaw);
  ay = convertRawAcceleration(ayRaw); az =
  convertRawAcceleration(azRaw);

  // display tab-separated accelerometer x/y/z values Serial.print("a:\t");
  Serial.print(ax);              Serial.print("\t");
  Serial.print(ay);              Serial.print("\t");
  Serial.print(az); Serial.println();
}
```

code continue on next page

```
float convertRawAcceleration(int aRaw) {
  // since we are using 2G range
  // -2g maps to a raw value of -32768
  // +2g maps to a raw value of 32767

  float a = (aRaw * 2.0) / 32768.0; return a;
}
```

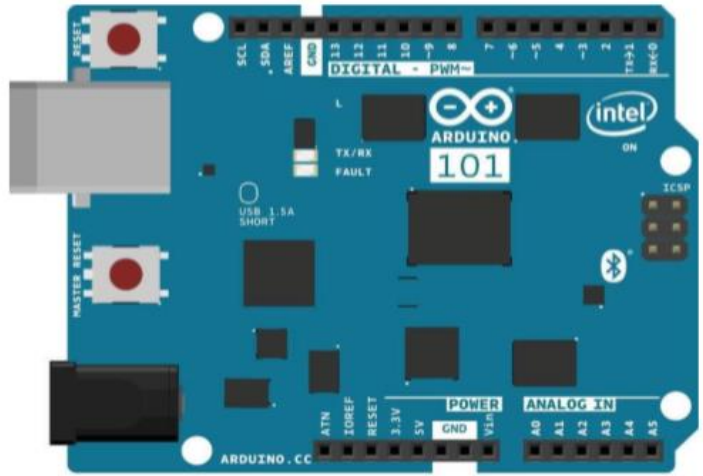
GYROSCOPE

THE IDEA

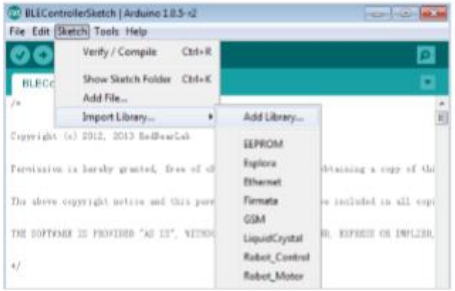
The idea is to read the gyroscope raw values and convert them into an angular velocity around each of the three axes. This information is useful to measure rotational movement around the three axes, something that acceleration can't measure if the movement is continuous.

Hardware

No additional hardware is required apart from the Genuino board.



SOFTWARE



Libraries

CurieIMU.h is the library that gives access to all the parameters, features and readings of the IMU chip of the Genuino board.

This unit contains a three axes accelerometer and a three axes gyroscope. This library is part of the Genuino board core and it is loaded together with the core files for Arduino or Genuino 101. In this tutorial we read the raw gyro values.

Functions

float convertRawGyro(int gRaw) - transforms the raw data read from the gyroscope (gRaw) into a value expressed in degrees per second (°/s). The formula of the function must be adjusted to match the gyroscope range set with setGyroRange.

CODE

```
/* Copyright (c) 2015 Intel Corporation.  All rights reserved.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU
Lesser General Public License as published by the Free Software Foundation; either version 2.1 of
the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY;
without even the implied
warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PUR-
POSE.  See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin
Street, Fifth Floor, Boston, MA
02110-1301  USA

*/
/* This sketch example demonstrates how the BMI160 on the Intel(R) Curie(TM) module can be
used to read accelerometer data*/
```

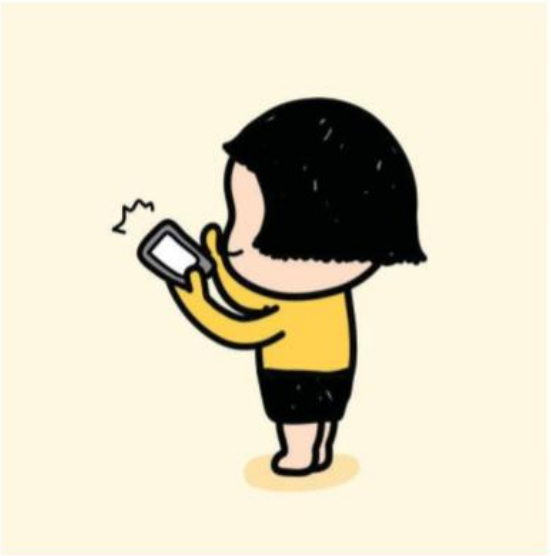
code continue on next page

```
#include "CurieIMU.h"

void setup() {
  Serial.begin(9600); // initialize Serial communication
  while (!Serial);       // wait for the serial port to open

  // initialize device Serial.println("Initializing
  IMU device..."); CurieIMU.begin();

  // Set the accelerometer range to 250 degrees/second
  CurieIMU.setGyroRange(250);
}

void loop() {
  int gxRaw, gyRaw, gzRaw;              // raw gyro values
  float gx, gy, gz;

  // read raw gyro measurements from device
  CurieIMU.readGyro(gxRaw, gyRaw, gzRaw);

  // convert the raw gyro data to degrees/second gx
  = convertRawGyro(gxRaw);
  gy = convertRawGyro(gyRaw); gz
  = convertRawGyro(gzRaw);

  // display tab-separated gyro x/y/z values
  Serial.print("g:\t");
  Serial.print(gx);
  Serial.print("\t");
  Serial.print(gy);
  Serial.print("\t");
  Serial.print(gz);
  Serial.println();
}
```

code continue on next page

```
float convertRawGyro(int gRaw) {
  // since we are using 250 degrees/seconds range
  // -250 maps to a raw value of -32768
  // +250 maps to a raw value of 32767

  float g = (gRaw * 250.0) / 32768.0; return g;
}
```

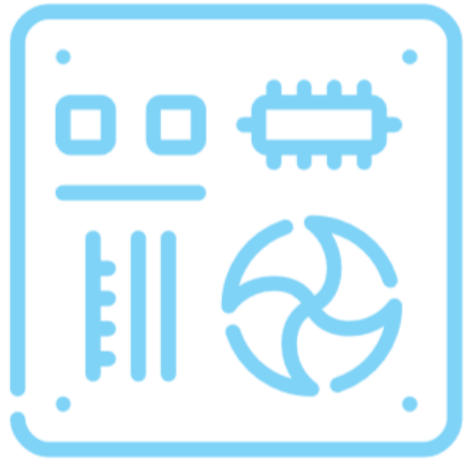
EXERCISES



BLUETOOTH WORKS

OVERVIEW

One of the coolest things about micro-controllers is their ability to communicate externally, in a sense making them Internet of Things devices. The Genuino 101 board has an inbuilt Bluetooth module, let's explore how to connect it and what you can do!



Sketch

There are many preloaded program examples or sketches within the Arduino IDE this is just one you can expand upon.

Click on File >> Examples >> CurieImu

>> CurieBLEHeartRateMonitor.

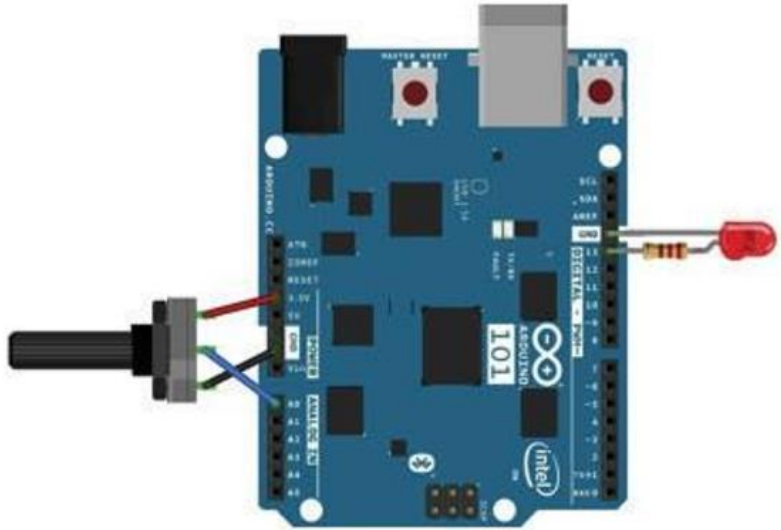
Hardware

You will need:

1. 1x LED
2. 1x Resistor
3. 1x Potentiometer
4. 1x Genuino board

Wiring up

Because we are working with a physical computer, we some- times need to connect physical elements. Wire up your Genuino 101 board as in the picture below (Access this site to be taken to the larger image bit.ly/GenuinoBluetooth).



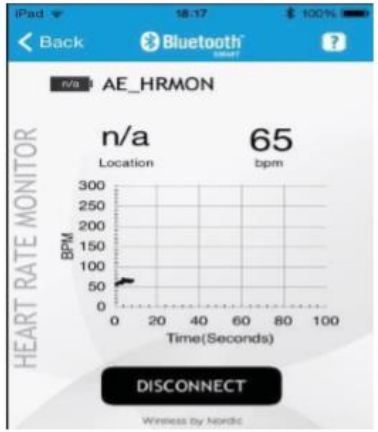
Upload sketch

Upload this sketch to the board by clicking on the upload button. You will need to wait for the green line to fill completely to know your sketch has been uploaded

DOWNLOAD APP

Visit the Apple or Google Play store and download the nRF Tool- box for BLE App then:

1. Open the App.
2. Select the 'heart' icon.
3. Click 'Connect.'
4. Select 'heart rate sketch'



The red LED should now be on, identifying that you have a Bluetooth connection. Now twist the potentiometer and see what happens! Magic! Then think of the possibilities! A mood sensor, boredom ranker, 'The worm' in a classroom and so much more!

You can also try:

1. Add a battery pack to make it mobile
2. Think of other uses and variables you could measure
3. Try connecting other analogue sensors and see what happens

PEDOMETER

OVERVIEW

Fitness tracking technology, wearables and smart watches are all the rage, but how do they work? This guide will show you how to create your own using the Genuino 101 board, and then the possibilities are infinite!

Sketch

There are many preloaded program examples or sketches within the Arduino IDE this is just one you can expand upon.

Click on File >> Examples >> CurieImu

>> StepCount

Tips and tricks

Make sure you have selected Tools >> Board >> Genuino 101 and that a COM port is selected Tools >> Port (select the port corresponding to your Genuino 101 board - it should look like “COM* (Genuino 101)”.

Upload Sketch

Upload this sketch to the board by clicking on the upload button. You will need to wait for the green line to fill completely to know your sketch has been uploaded

Watch Your Steps

Now open the serial monitor Tools >> Serial Monitor to see the number of steps taken, there will be a slight delay.

Also Try

1. Add a battery pack to make it mobile.
2. Add an LCD screen to display the steps in real time.
3. Connect the Pedometer using Bluetooth to your phone.

VISUAL PROGRAMMING

OVERVIEW

For many people making the transition from visual programming languages like Scratch to text based programming languages like Arduino* can be challenging! Ardublock a visual program builder for Arduino might be just the tool you need to bridge the void!

Quick Check

1. You will need an Arduino board such as a Genuino 101 board
2. You will need to have installed the Arduino IDE

Download Ardublock

1. Visit http://bit.ly/Ardublock and download the latest version of Ardublock
2. Open Arduino and click on File >> Preferences and open the 'Sketchbook Location' by clicking on Browse

Install Ardublock

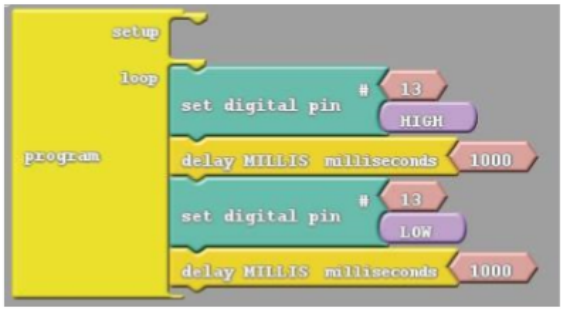
1. Click on Arduino
2. Create a folder called tools. (all lower case)
3. Within the tools folder create a new folder called ArduBlockTool. (case sensitive)
4. Within the ArduBlockTool folder create a new folder called tool.
5. Paste the Ardublock file (Ardublock-xxxxx.jar) you have down- loaded into the final folder (tool) you created.

Starting Ardublock

1. Open Arduino
2. Click on Tools >> Ardublock

   Now you are set to create your first Sketch. Let's make the board blink!

3. Click on "Control" and drag the "Program" block as the base for your code.
4. Using the blocks in the "Control" and "Pin" sections replicate the block structure to your left.



Uploading Your Code

1. Click on Upload (The Arduino IDE should open)
2. Make sure your Arduino board is connected and both your Board and Port is selected in the “tools” menu.
3. Click “upload” and you should be blinking!

THANK YOU