Sensors and Networks - Project 1

Felix Auer, Elena Zueva, Xabier Gardeazabal

1) Build the hardware setup
2) Program a simple script to test the sensors
3) …

Setting up the hardware

We have built and tested the accelerometer following this simple schema and sample code: http://wiring.org.co/learning/basics/accelerometer.html

Accelerometer-Calibration

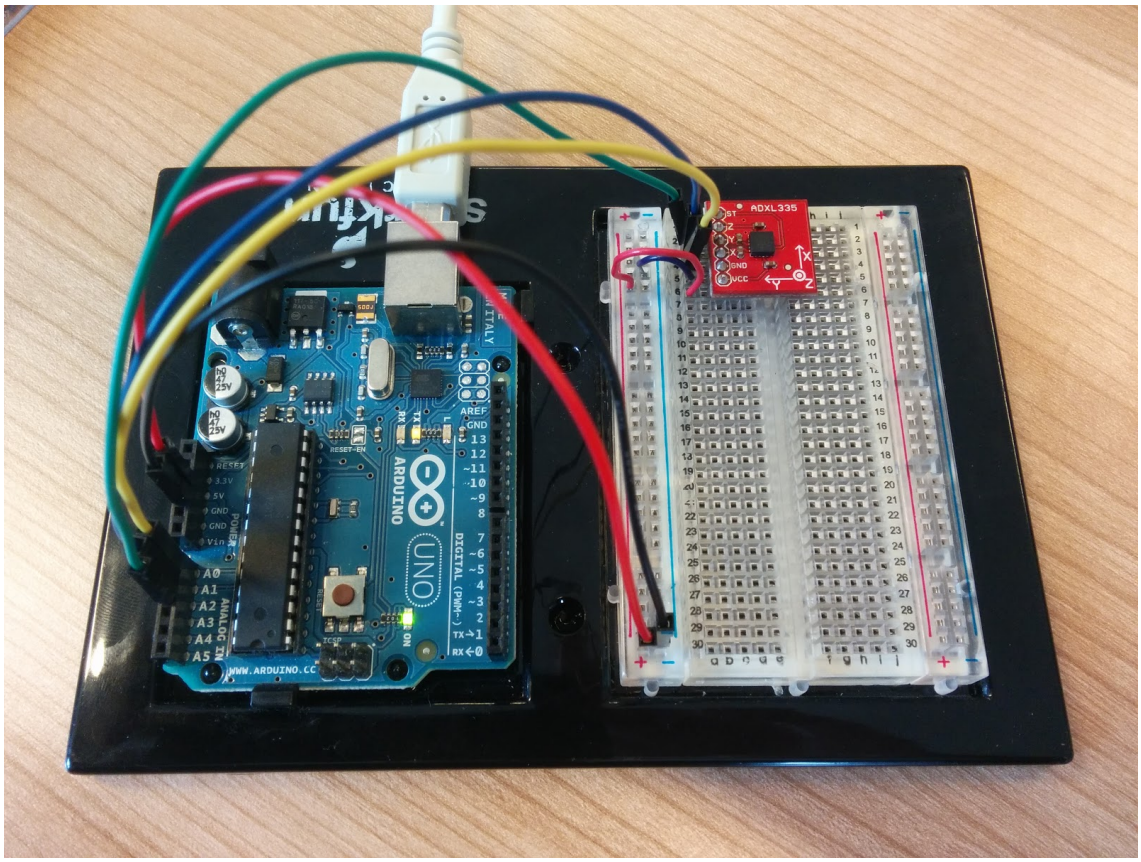
```
int x, y, z;

void setup()
{
  Serial.begin(9600);      // sets the serial port to 9600
}

void loop()
{
  x = analogRead(0);       // read analog input pin 0
  y = analogRead(1);       // read analog input pin 1
  z = analogRead(2);       // read analog input pin 1
  Serial.print("accelerations are x, y, z: ");
  Serial.print(x, DEC);    // print the acceleration in the X axis
  Serial.print(" ");       // prints a space between the numbers
  Serial.print(y, DEC);    // print the acceleration in the Y axis
  Serial.print(" ");       // prints a space between the numbers
  Serial.println(z, DEC);  // print the acceleration in the Z axis
  delay(100);              // wait 100ms for next reading
}
```

Once we have verified that the sensors work, we have followed this article and have calibrated the accelerometer:
http://bildr.org/2011/04/sensing-orientation-with-the-adxl335-arduino/

With this test we could obtain the minimum and maximum value during standard-operation (just tilting - no shaking). These values could than be mapped to a tilting value between -90° and +90°.



Ultimately, we found a library to access the accelerometer-data in a more conveinient way.

Library: https://github.com/infomaniac50/ADXL335

LCD-Screen

Note: We have discovered that our first LCD display was defective, so it had to be replaced.

The LCD is attached to a digital IO-Pin and a software-serial port was set up, as the internal serial-port is used for the communication via USB.

We moved all the methods to display text on the LCD to its own class-file (LCDprinter.ino)

Our display: https://www.sparkfun.com/products/9393
Optional library: https://github.com/nemith/serLCD
(Instalation guide: http://www.arduino.cc/en/Guide/Libraries)

Diagnostic-LEDs

We have now wired some LEDs to light up and act as alarms when some variables reach certain values [see last wiring image]. There are two yellow LEDs that light up when either the sensed total acceleration (in all three axis) or temperature go over a predefined threshold. The other two red LEDs light up when it is sensed that either the accelerometer or the thermistor are malfunctioning (e.g. when the connection wires are cut or disconnected).

Temperature Sensor (Thermistor)

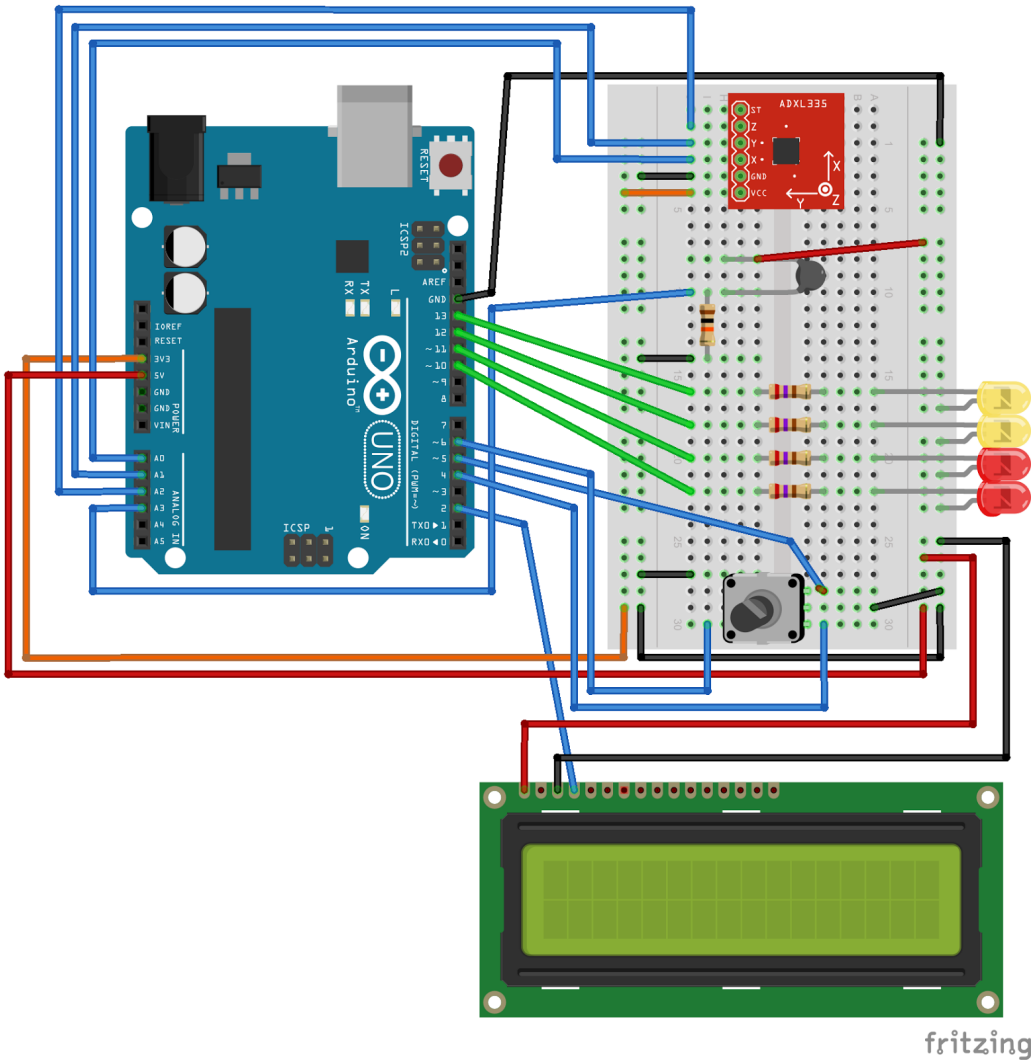
The temperature Sensor is realized via a thermistor, which is basically a resistor, which changes its resistance when the temperature changes.
The thermistor is attached to an analog-input (using a pullup resistor).

The temperature is then calculated according to this tutorial:
http://computers.tutsplus.com/tutorials/how-to-read-temperatures-with-arduino--mac-53714
At the beginning we were obtaining a strange behavior in the temperature change readings, as the warmer it got, the lower the resistance got. We discovered that this was happening because the order of the 10k resistor and the thermistor was swapped in the voltage divider circuit.

Rotary Encoder:

The rotary encoder is attached to 3 digital IO-pins. Two for the rotation encoding, one for the internal push-button. For the encoding we used the library provided via this link:
https://github.com/0xPIT/encoder/tree/arduino

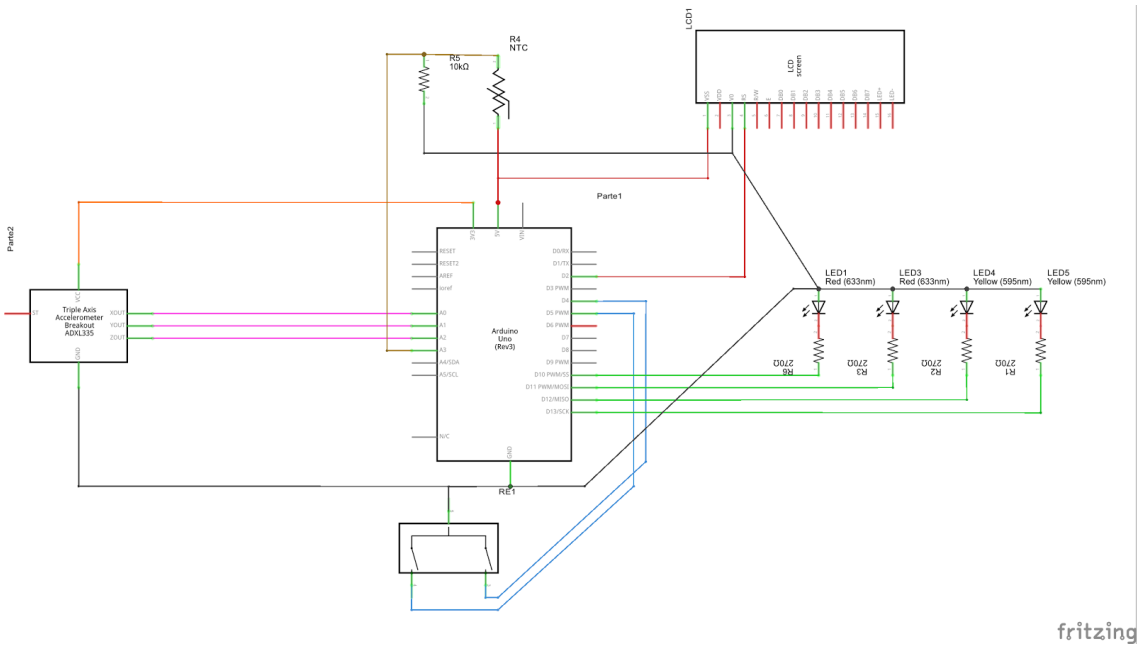
Final wiring schema



The image above shows the final wiring schema.

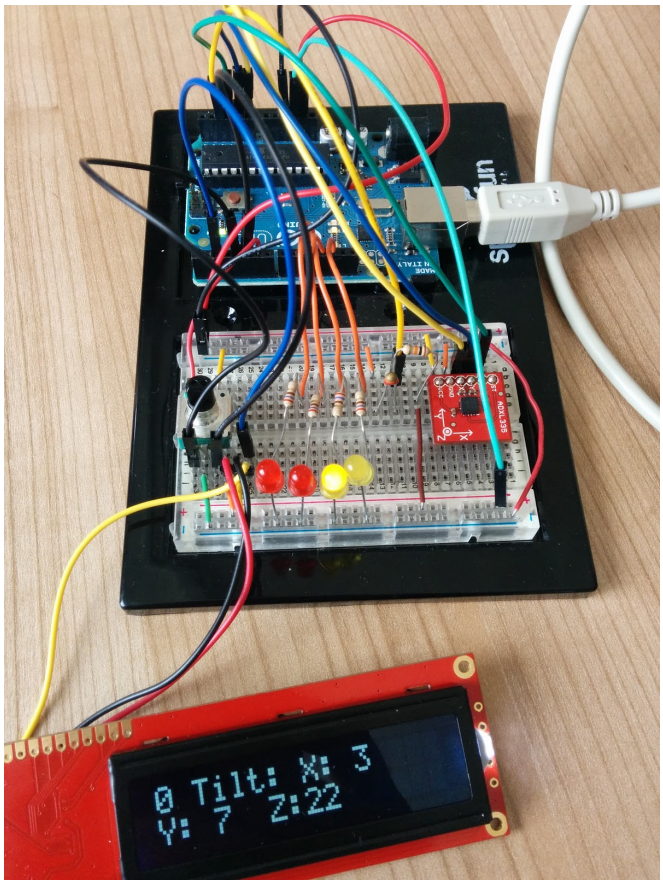
- Analog inputs:
  - A0-A2: 3-Axis Accelerometer
  - A3: Thermistor
- Digital inputs:
  - 10-13: Status LEDs
  - 4,5: Rotary encoder
  - 2: Software-serial-port for LCD-display
- All attached devices are grounded.
- 3,3V-power supply:
  - Thermistor
- 5V-power supply:
  - LCD-Display
  - Rotary-Encoder & Pushbutton
  - Thermistor

Circuit Diagram



User guide:

- Power the board by connecting it to the PC.
- Use the rotary encoder to navigate through the different data pages.
- Push the rotary encoder while on a certain data-page to reset the maximum and/or minimum values of that data-page.
- Change the tilt of the board or move it in any direction to see the tilt angles and the acceleration values change. Get a heat or cold emitting source to the board to see how the temperature changes.
- In the picture below, the red and yello LEDs from left to right:
    - 1st red LED: thermistor plausibility indicator (if lighted up, thermistor sensor is not working properly)
    - 2nd red LED: accelerometer plausibility indicator (if lighted up, accelerometer sensor is not working properly)
    - 1st yellow LED: temperature alarm (if lighted up, it indicates that the maximum temperature has been reached).
    - 1st yellow LED: accelerometer alarm (if lighted up, it indicates that the maximum acceleration (sum of all three axis) has been reached.


0 Tilt: X: 3
Y: 7 Z:22