Nume student: Ștefan Drăguț
Mail: stefan.dragut@student.tuiasi.ro
Proiect: Acceleration Monitor

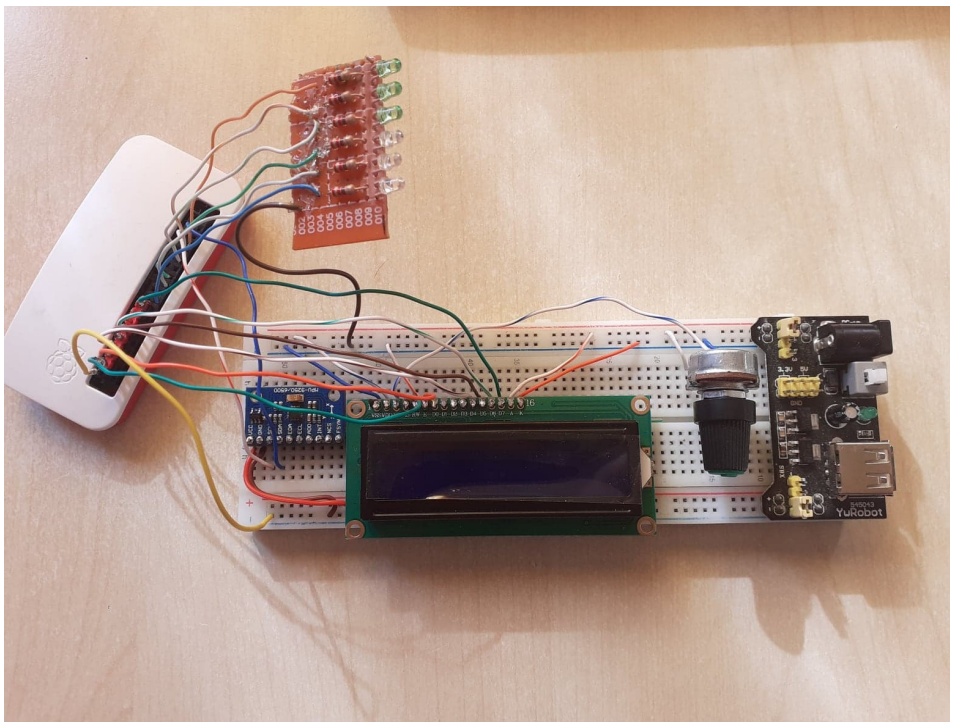
Acceleration Monitor

Video:https://www.youtube.com/watch?v=1WAdju9xkc4

Hackster:https://www.hackster.io/stefan-dragut/acceleration-monitor-de1f4c

Purpose

The project aims to create a module for measuring the acceleration force, displaying it on an LCD display, as well as illustrating the proportion of the force through a row of LEDs.



Operating mode

- With the help of the accelerometer, the acceleration on all three coordinate axes, x, y and z, is constantly read.
- The gravitational force is calculated using these accelerations, but is not displayed in real time on the LCD display, but once every 250 milliseconds, to ensure the readability of the text on the display. It is accompanied by forces acting on all three coordinate axes. On the z-axis, while the assembly stands still, it can be seen that the gravitational force is approximately equal to 1G, ie the gravitational acceleration of the Earth. The potentiometer is used to adjust the contrast of the display.
- For a real-time representation of the applied force, a module consisting of six LEDs was used, of which three green, two yellow and one red, which light up in proportion to the force measured by the accelerometer. Depending on the range of values in which the measured force is located, more and more LEDs will light up, until, from a force of 1.6G onwards, all the LEDs will be lit, up to the red one.
- When the red LED lights up, the program will write in a file the exact date and time at which the event occurred, as well as the gravitational force applied at that time.

Practical applicability

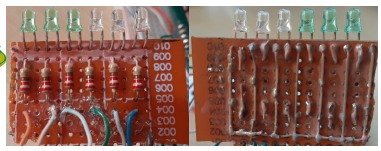
The idea of the module can be put into practice by installing it to monitor the transport of an

important package, which is fragile and could be damaged or destroyed if hit. These events can be seen in the “log.txt” file in the module memory.

```
pi@raspberrypi:~ $ cat log.txt
High force detected on << Sat Apr 25 22:07:12 2020 >> with the GForce: 2.1G
High force detected on << Sat Apr 25 22:07:14 2020 >> with the GForce: 2.07G
High force detected on << Sat Apr 25 22:07:16 2020 >> with the GForce: 2.21G
High force detected on << Sat Apr 25 22:08:26 2020 >> with the GForce: 2.29G
High force detected on << Sat Apr 25 22:08:29 2020 >> with the GForce: 2.01G
High force detected on << Sat Apr 25 22:08:30 2020 >> with the GForce: 1.7G
High force detected on << Sat Apr 25 22:08:31 2020 >> with the GForce: 1.84G
High force detected on << Sat Apr 25 22:38:10 2020 >> with the GForce: 2.16G
High force detected on << Sat Apr 25 22:38:12 2020 >> with the GForce: 2.11G
High force detected on << Sat Apr 25 22:38:13 2020 >> with the GForce: 2.08G
High force detected on << Sat Apr 25 22:39:11 2020 >> with the GForce: 1.74G
High force detected on << Sat Apr 25 22:39:15 2020 >> with the GForce: 2.33G
High force detected on << Sun Apr 26 22:22:31 2020 >> with the GForce: 1.71G
High force detected on << Sun Apr 26 22:22:33 2020 >> with the GForce: 1.74G
High force detected on << Sun Apr 26 22:22:43 2020 >> with the GForce: 1.9G
High force detected on << Sun Apr 26 22:23:14 2020 >> with the GForce: 1.73G
High force detected on << Sun Apr 26 22:23:16 2020 >> with the GForce: 1.64G
High force detected on << Sun Apr 26 22:23:53 2020 >> with the GForce: 2.07G
High force detected on << Sun Apr 26 22:24:06 2020 >> with the GForce: 1.82G
High force detected on << Sun Apr 26 22:24:06 2020 >> with the GForce: 1.94G
High force detected on << Sun Apr 26 22:24:07 2020 >> with the GForce: 1.63G
High force detected on << Sun Apr 26 22:24:07 2020 >> with the GForce: 2.18G
High force detected on << Sun Apr 26 22:24:25 2020 >> with the GForce: 1.81G
High force detected on << Sun Apr 26 22:24:26 2020 >> with the GForce: 2.37G
pi@raspberrypi:~ $
```

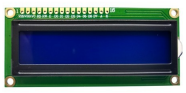
Components

1. Raspberry Pi Zero W 5. 6 LEDs custom module (3 Green, 2 Yellow, 1 Red)

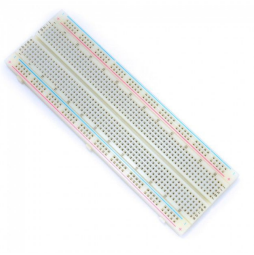



2. Accelerometer MPU9250 - 3 axis 6. Breadboard power supply 3.3-5V

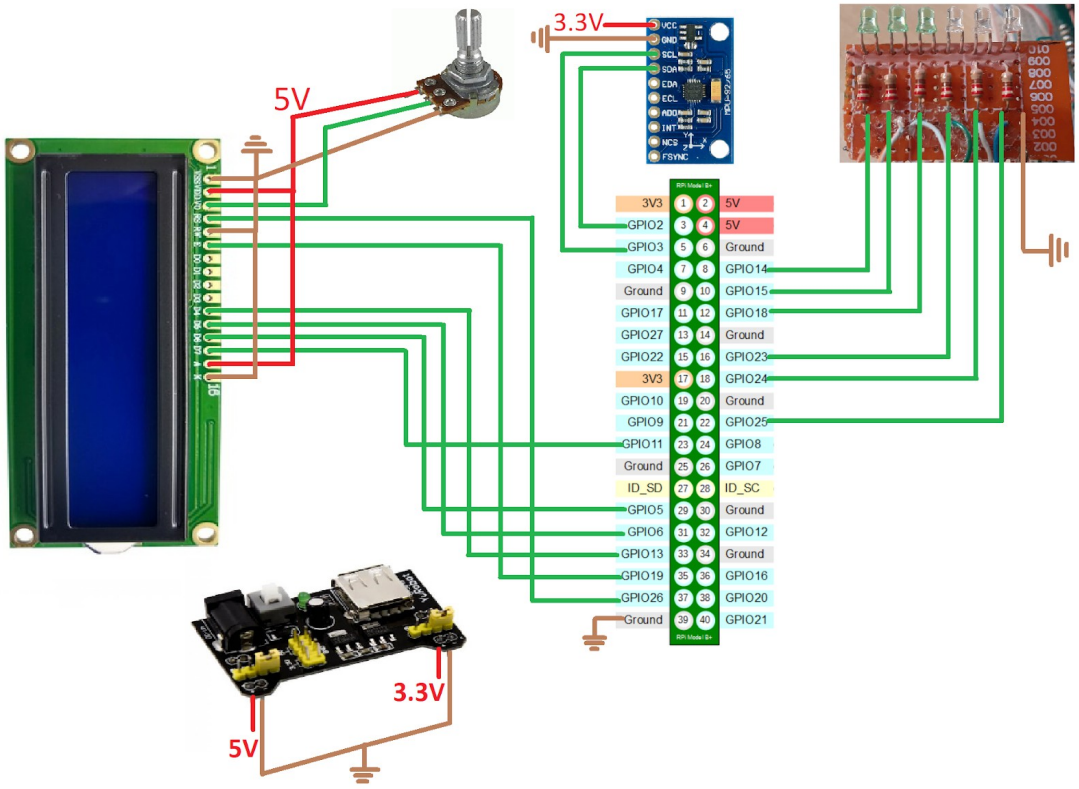

3. LCD Display 16x2 7. 10K ohm potentiometer



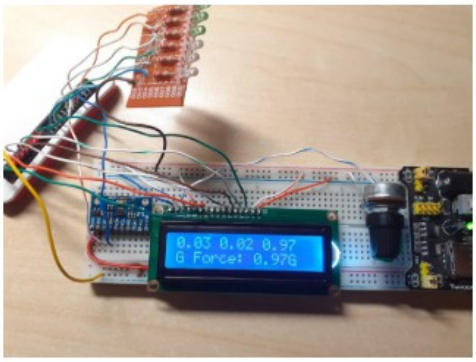
4. Breadboard

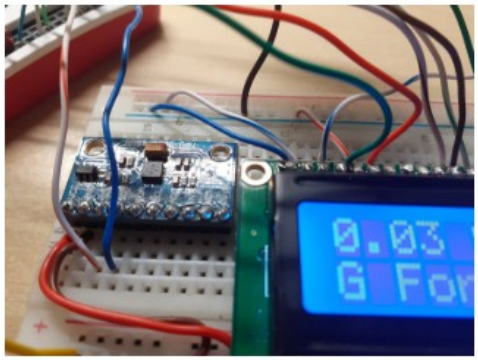


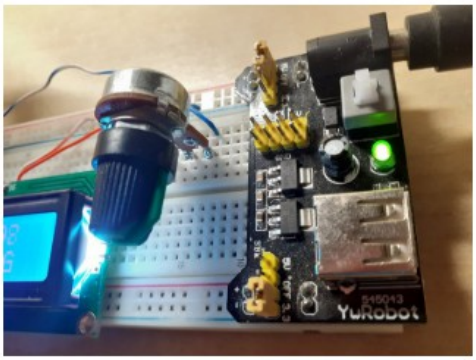
Electrical schematic

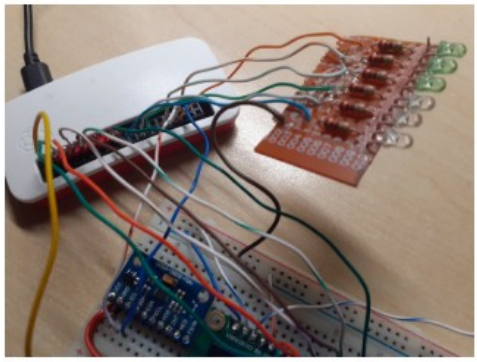


Project build images









Code

```
#!/usr/bin/python
# Example using a character LCD connected to a Raspberry Pi or BeagleBone Black.
import FaBo9Axis_MPU9250
import time
import sys
import math
import Adafruit_CharLCD as LCD
import RPi.GPIO as GPIO


mpu9250 = FaBo9Axis_MPU9250.MPU9250()


# Raspberry Pi pin configuration:
lcd_rs        = 26
lcd_en        = 19
lcd_d4        = 13
lcd_d5        = 6
lcd_d6        = 5
lcd_d7        = 11
lcd_backlight = 15
```

```
# Define LCD column and row size for 16x2 LCD.
lcd_columns = 16
lcd_rows    = 2


# Initialize the LCD using the pins above.
lcd = LCD.Adafruit_CharLCD(lcd_rs, lcd_en, lcd_d4, lcd_d5, lcd_d6, lcd_d7,
                lcd_columns, lcd_rows, lcd_backlight)
# LEDS
led_1 = 14
GPIO.setup(led_1,GPIO.OUT)
led_2 = 15
GPIO.setup(led_2,GPIO.OUT)
led_3 = 18
GPIO.setup(led_3,GPIO.OUT)
led_4 = 23
GPIO.setup(led_4,GPIO.OUT)
led_5 = 24
GPIO.setup(led_5,GPIO.OUT)
led_6 = 25
GPIO.setup(led_6,GPIO.OUT)


GPIO.output(led_1,GPIO.LOW)
GPIO.output(led_2,GPIO.LOW)
GPIO.output(led_3,GPIO.LOW)
GPIO.output(led_4,GPIO.LOW)
GPIO.output(led_5,GPIO.LOW)
GPIO.output(led_6,GPIO.LOW)


f = open("log.txt", "a")


# Print a two line message
lcd.clear()
lcd.message('   SISTEME CU\nMICROPROCESOARE')
time.sleep(4)
lcd.clear()
lcd.message('  ACCELERATION\n     MONITOR')
time.sleep(4)
lcd.clear()
lcd.message('  DRAGUT STEFAN\n   GRUPA 1306A')
time.sleep(4)


t2 = 0
try:
   while True:
      accel = mpu9250.readAccel()


      g = math.sqrt(accel['x']*accel['x'] + accel['y']*accel['y'] + accel['z']*accel['z'])


      t1 = time.time()


      if t1-t2 > 0.25:
           lcd.clear()
           lcd.message(str(round(accel['x'],2)) +  ' ' + str(round(accel['y'],2)) + ' ' +
str(round(accel['z'],2))+ '\n' + 'G Force: ' + str(round(g,2)) + 'G')
```

```
            t2 = t1


      if abs(g)<=1:
            GPIO.output(led_1,GPIO.LOW)
            GPIO.output(led_2,GPIO.LOW)
            GPIO.output(led_3,GPIO.LOW)
            GPIO.output(led_4,GPIO.LOW)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.6:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.HIGH)
            GPIO.output(led_4,GPIO.HIGH)
            GPIO.output(led_5,GPIO.HIGH)
            GPIO.output(led_6,GPIO.HIGH)
            text = 'High force detected on << ' + time.asctime(time.localtime(time.time())) + ' >>
with the GForce: ' + str(round(g,2)) + 'G\r\n'
            f.write(text)
      elif abs(g)>1.5:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.HIGH)
            GPIO.output(led_4,GPIO.HIGH)
            GPIO.output(led_5,GPIO.HIGH)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.4:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.HIGH)
            GPIO.output(led_4,GPIO.HIGH)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.3:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.HIGH)
            GPIO.output(led_4,GPIO.LOW)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.2:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.LOW)
```

```
            GPIO.output(led_4,GPIO.LOW)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.1:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.HIGH)
            GPIO.output(led_3,GPIO.LOW)
            GPIO.output(led_4,GPIO.LOW)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      elif abs(g)>1.0:
            GPIO.output(led_1,GPIO.HIGH)
            GPIO.output(led_2,GPIO.LOW)
            GPIO.output(led_3,GPIO.LOW)
            GPIO.output(led_4,GPIO.LOW)
            GPIO.output(led_5,GPIO.LOW)
            GPIO.output(led_6,GPIO.LOW)
      time.sleep(0.025)

except KeyboardInterrupt:
   f.close()
   sys.exit()
```