

Año III, No.05 Enero-Junio 2015

ISSN: 2395-9029

PROYECTOS INSTITUCIONALES Y DE VINCULACIÓN



UANL

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN



FIME

FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA

MEASUREMENT AND RECORDING OF MAGNETIC FIELDS AT HIGH ALTITUDES

Ing. Pedro Antonio Limón Díaz
Centro de Investigación e Innovación en Ingeniería Aeronáutica
Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León
pedrolimond@gmail.com

MC. Sergei García Ballinas
Universidad Pablo Guardado Chávez
Lecturer
kanrak@hotmail.com

Mario Alberto García Ramírez, PhD
Centro de Investigación e Innovación en Ingeniería Aeronáutica
Facultad de Ingeniería Mecánica y Eléctrica
Universidad Autónoma de Nuevo León
Lecturer
seario@gmail.com

ABSTRACT.

When a land-air missile is fired to pursue a mobile target, quite a few variables play a key role in order to reach and neutralise the target in no-time. It is desired; nevertheless, this system requires an amount of time, fuel, real-time data analysis as well as a discriminator system, all these in order to reach the mobile target. This is why, in this work we present a study for a set of variables measured in situ such as: the wind speed, magnetic field, pressure and humidity to generate logs that can be useful for either civil and military applications. The system is designed to measure in situ a set of variables starting at 500 m up to 30 km. As a result of this study, all the information gathered can be fed into the missile main computer and with the info loaded, it can trace trajectories in which the efficiency of the engine, fuel and cargo reach the target. The low atmosphere system has the following set of instruments: a barometer, accelerometer, gyroscope and a magnetometer as the sensor array. We are using an Intel Galileo board as the motherboard. The information collected is saved within a SD high density memory card.

KEYWORDS: Magnetic field, Intel Galileo, GPUs, Sensors array,

INTRODUCTION.

Every now and then, the road that follows the magnetic field has evolved for several natural reasons. It changes according to the altitude and longitude [1].

According to such information, it is required to have a rather accurate information in a certain area. This is why, the hardware and the software has to be robust enough to process the information without suffer any sort of missfunction.

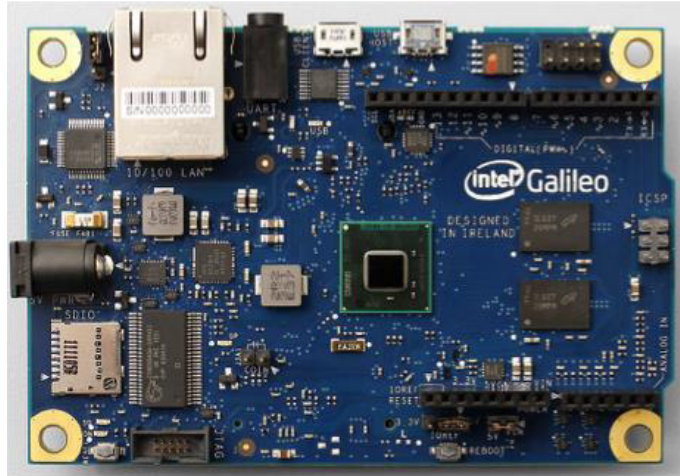


Figure 1.- Intel Galileo Board.

Figure 1 shows the top view of an Intel Galileo Board. This board provides a friendly programmable control for swift experiments. It is based on the Intel® Quark SoC X1000 Application Processor, a 32-bit Intel Pentium-class system on a chip. The board is compatible with Arduino-based environment software (including the communication protocol I2C) [2]; similarly, it describes that by using a micro SD memory it is possible to load an operative system (OS) and manipulate the inputs and outputs as required.

Ubilinux is an embedded free Linux distribution from Emutex. It is based on a Debian Wheezy distribution that is an extension of the OS Linux kernel. It is targeted at embedded devices that have limited memory and storage capabilities. The first version of the Ubilinux distribution was available as an installer image for Intel Quark. The installer package can be downloaded and copied to an external memory (this case a micro SD) from any explorer or any other standard file copying mechanism.

An inertial measurement unit (IMU) is an electronic device that measures and reports velocity, gravity forces and orientation through accelerometers, magnetometers and gyroscopes. For the experimentation of this work the IMU selected has to be commercial and it has to be communicated via I2C protocol and as consequence it requires to be maneuverable.

Python is a programming language of high level, interpreted and multipurpose. Python can be used across multiple platforms and operating systems, including popular highlights such as Windows, Mac OS and Linux. It is a powerful, flexible and it has a rather clear and concise language syntax. Python also has the advantage of being fast due to it interprets and not compiles [3].

DEVELOPMENT.

The Intel Galileo Board is connected and tested to verify the correct operation of the inputs and outputs by using the Arduino environment as shown elsewhere [4]. Once the entire system is checked and confirmed, a Ubilinux image is downloaded [5] and copied in to a micro SD and inserted within the micro SD slot on the board (8 Gb of capacity is recommended). To access the Linux environment in the screens it is required to download and install the software Putty, described elsewhere [4]. The communication between the board and the PC will be serial as consequence.

After the users load the OS, it is plugged to Internet by the Ethernet Port to update the necessary components. To develop the algorithms and coding them into Python and running them in Ubilinux, it is the most time-consuming process. The best way to have a communication with an external sensor is via SPI and I2C. The advantage of the I2C communication over the SPI is the capability of having many devices connected at the same time and transferring it at the same time all sort of data. To check that it is already installed within the board, it is required to introduce the line code displayed in (1) and as response, the screen depicted in Fig. 2 must be displayed. Although the I2C is installed, it does not mean it is available [4]. The enabling set of flags is always introducing at the beginning of the program.

```
i2cdetect -r -y 0 (1)
```

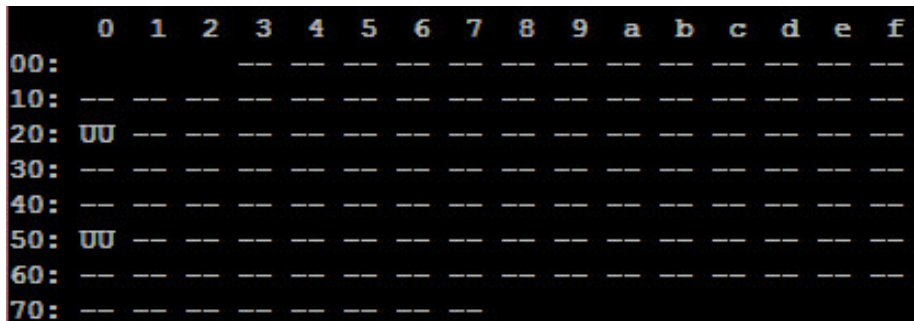


Figure 2.- Shows that the I2C communication is installed. Each direction means one sensor.

The IMU sensor used is a GY-81 (**Fig. 3**); it includes a BMP08 barometric sensor; ITG3205 gyroscope chip; BMA180 accelerometer chip; and HMC5883L magnetometer. Once connected the chip and to enable the I2C in a short routine, the code (1) is introduced and now you can see the address in hexadecimal number of each sensor, Fig. 4. The magnetometer is in 0x1e, the accelerometer in 0x40; the gyro in 0x68; and the barometer in 0x77.

```

Norte (327.20, 28.50) Este (57.20, 28.50)
Sur (147.20, 28.50) Oeste (237.20, 28.50)
Frente (217.10, 23.90) Arriba (307.10, 23.90)
Atras (37.10, 23.90) Abajo (127.10, 23.90)
Temperatura: 24.10 C
Presion: 95761.00 Pa
Altura: 511.60 m
Cabeceo= 0.16 Alabeo= -0.15 Guinhada= 0.05
X= -0.15 g Y= 0.04 g Z= 1.99 g
Heading (1.20, 43.50) Downing (0.10, 47.90)
Cabeceo= 0.16 Alabeo= -0.08 Guinhada= -0.02
X= -0.16 g Y= 0.04 g Z= 1.99 g
Heading (0.20, 16.50) Downing (359.10, 21.90)
Cabeceo= 0.09 Alabeo= -0.01 Guinhada= 0.12
X= -0.16 g Y= 0.04 g Z= 1.99 g
Heading (0.20, 16.50) Downing (359.10, 21.90)
Cabeceo= 0.16 Alabeo= 0.13 Guinhada= 0.05
X= -0.16 g Y= 0.03 g Z= 2.00 g
Heading (1.20, 53.50) Downing (359.10, 21.90)
Cabeceo= 0.09 Alabeo= -0.08 Guinhada= 0.12
X= -0.16 g Y= 0.03 g Z= 2.00 g
Heading (0.20, 26.50) Downing (359.10, 54.90)
Cabeceo= -0.05 Alabeo= -0.01 Guinhada= -0.02
X= -0.16 g Y= 0.03 g Z= 2.00 g
Heading (1.20, 53.50) Downing (359.10, 21.90)
Cabeceo= 0.16 Alabeo= 0.06 Guinhada= 0.05
X= -0.17 g Y= 0.03 g Z= 1.99 g
Heading (0.20, 26.50) Downing (359.10, 36.90)
Cabeceo= 0.16 Alabeo= -0.01 Guinhada= -0.02
X= -0.17 g Y= 0.03 g Z= 1.99 g
Heading (0.20, 26.50) Downing (359.10, 54.90)
    
```

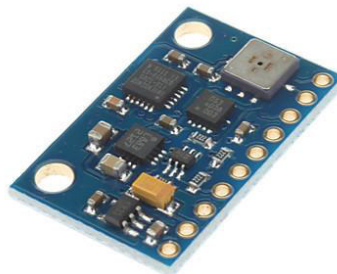


Figure 3.- GY-81 IMU.

The barometer program developed gives altitude, atmospheric pressure and altitude. The gyroscope's gives data of the rotation in the axes (roll, yaw and pitch). The accelerometer's provides the information of the movement. The magnetometer's works as a compass, first takes the references from where the North is located. To delete the interference the offset is implemented, it is considered that the average of 10 samples is enough to build an offset reference.

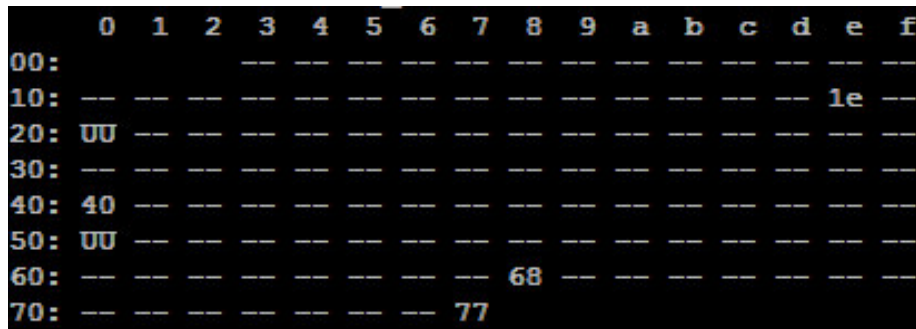


Fig 4.- I2C communication enable and showing the address of each sensor.

The barometer program developed gives altitude, atmospheric pressure and altitude. The gyroscope's gives data of the rotation in the axes (roll, yaw and pitch). The accelerometer's provides the information of the movement. The magnetometer's works as a compass, first takes the references from where the North is located. To delete the interference the offset is implemented, it is considered that the average of 10 samples is enough to build an offset reference.

The way of recollecting the data of each sensor is simple, first enable the I2C, access to the memory of each sensor one by one, do some math to the data and transform the data from a hexadecimal number to decimal. Once the four routines have been tested and added into the main program. While this unique program is running, it is doing two things, displaying the result and saving that result in a txt document. Figure 5 shows the display of the entire program.

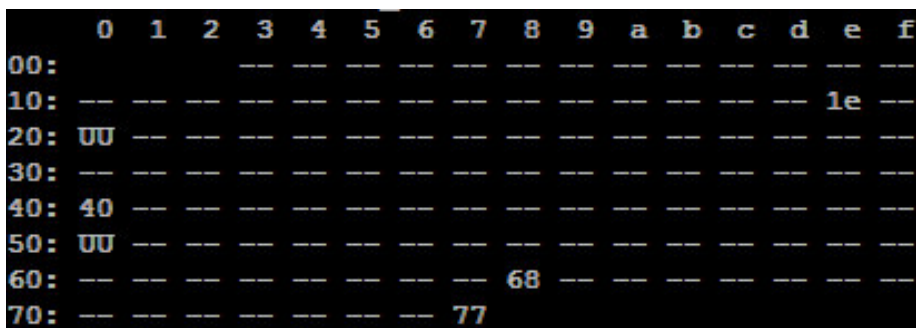


Figure 5.- The results displayed in the screen.

RESULTS.

In this part of the work the results of the test are shown. Tests were done in a controlled environment. Those show the rotation and the move of the IMU. The magnetometer data first takes the references doing an offset.

Figure 6 shows the data read by the barometer. This data is updated every 10 seconds and compared to the previous one. The experiment was defined in to lift and drop the sensor at intervals. While the atmospheric pressure registered the rises and drops; the altimeter was not precisely calibrated in here.

During the excesize, it was expected that the Z-axis of the accelerometer shows the most significant changes, it was like that until the chip was rotated to the X-axis and the peak of the axis increase significantly. After returning to the original orientation the changes fell back in the Z-axis as shown in Fig. 7.

When the peak in the X-axis of the accelerometer appears, the same happens in the X-axis of the gyroscope (Fig. 8), what changes are the values but the effect is similar. Meanwhile, the magnetometer (Fig. 9) is taking the data and shows where it is seeing. The reference to where the north is when it takes the value of 130 degrees.

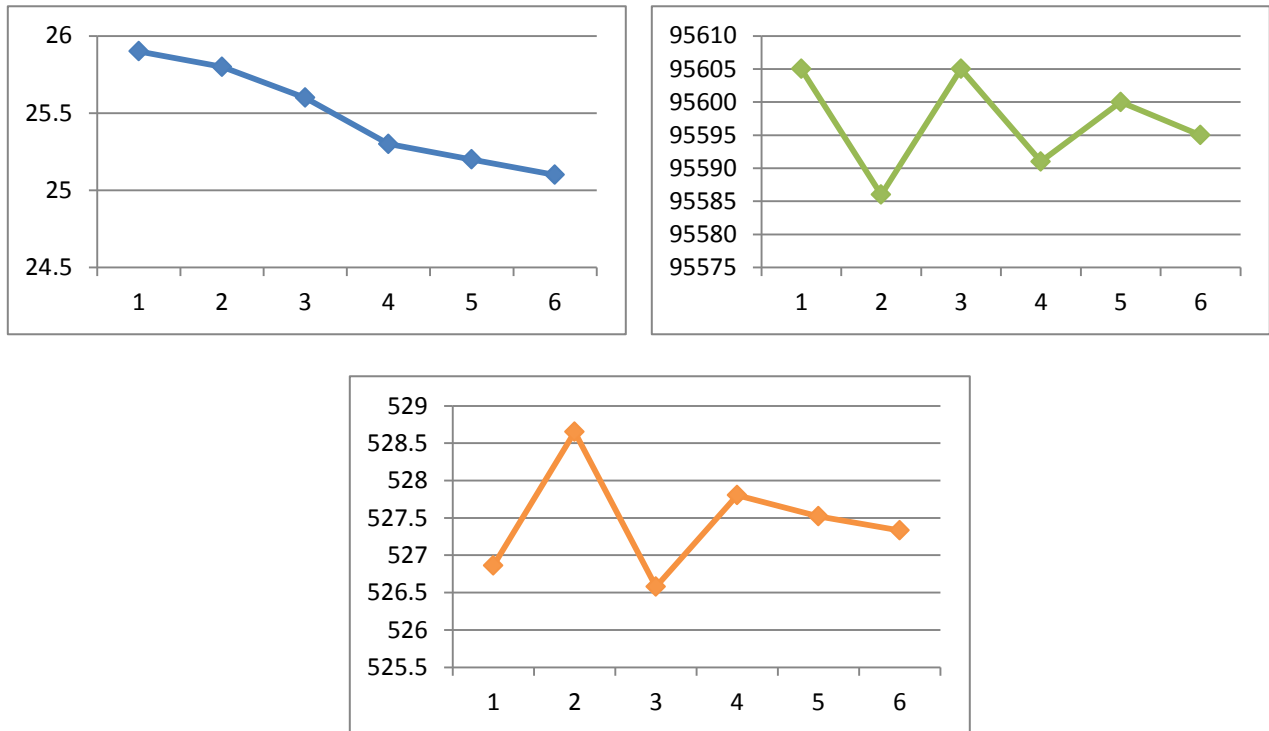


Fig 6. - a) the blue indicates the temperature. B) the green indicates the pressure inside the room
c) the orange is the altitude.

Fig 7.- Value of the 3 axis of the accelerometer. Blue line is the X-axis, red line is Y-axis and green line Z-axis.

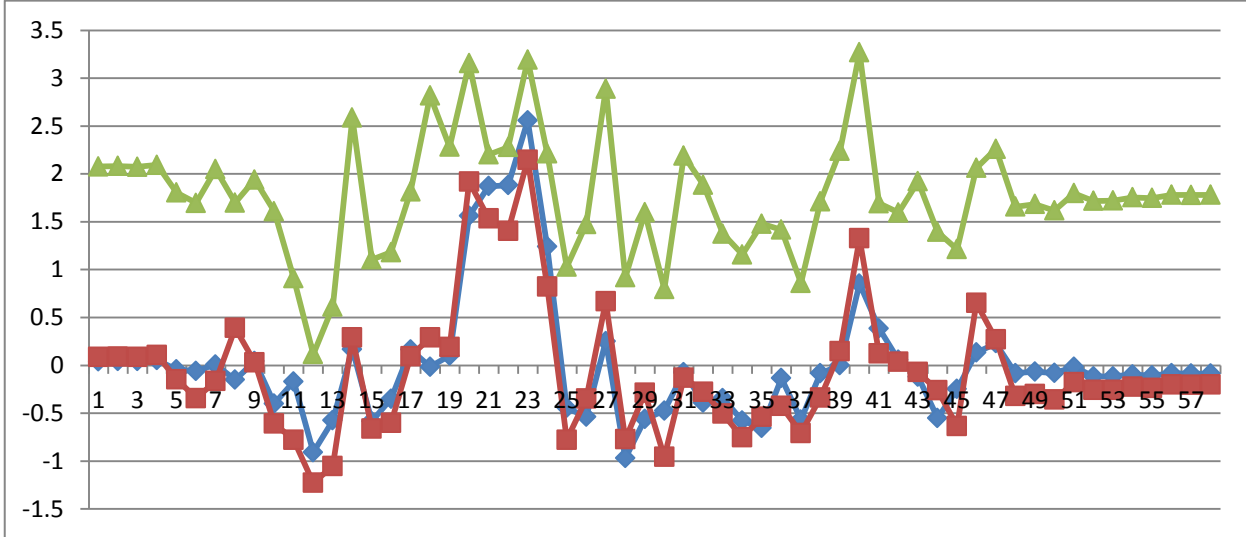
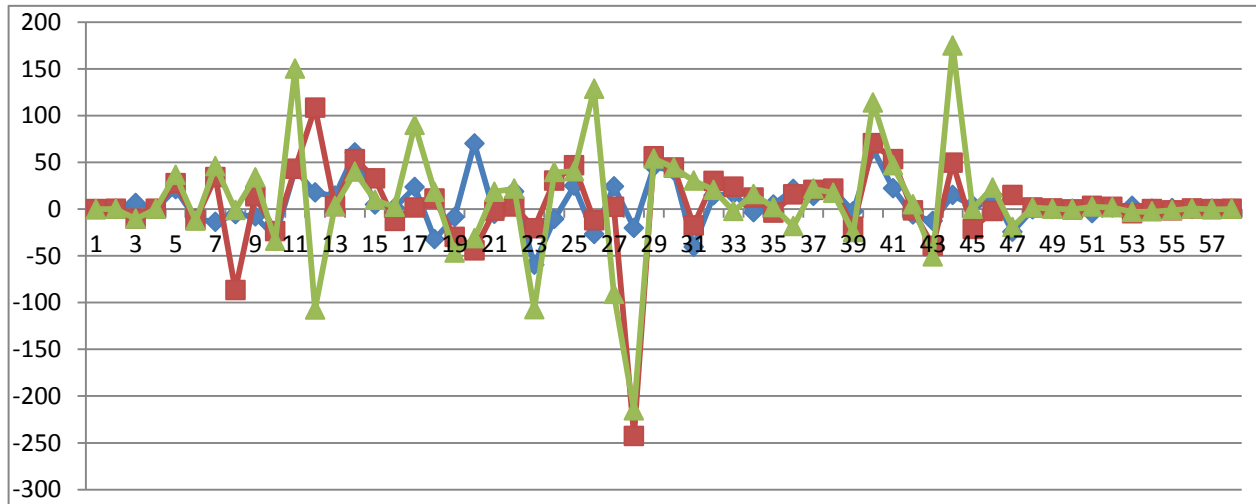


Fig 8.- Value of the 3 axis of the gyroscope. Blue line is the X-axis (roll), red line is Y-axis (pitch) and green line Z-axis (yaw).



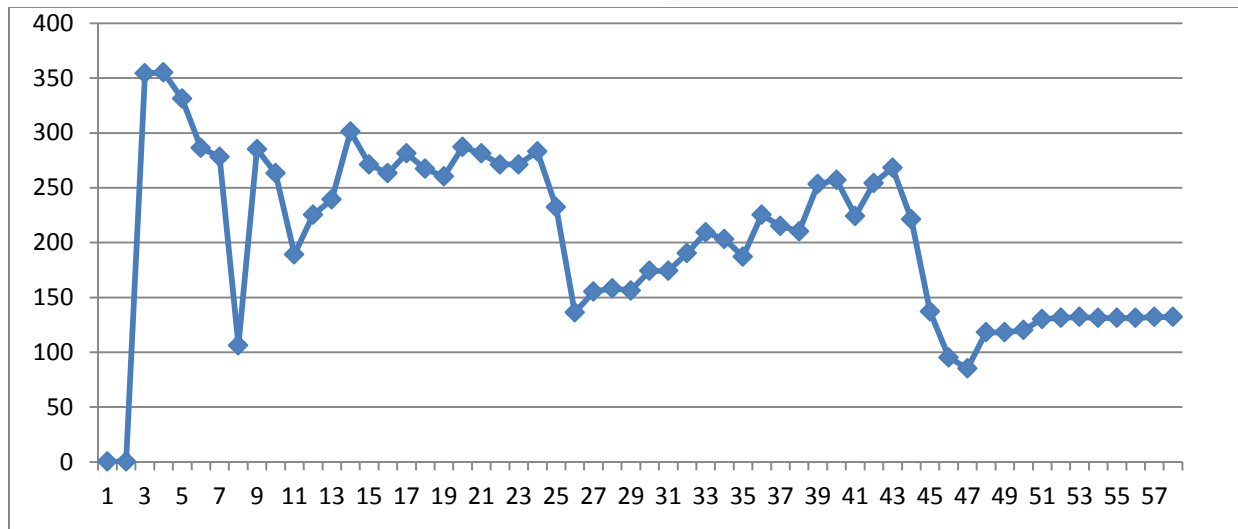


Fig 9.- Value of the gyroscope (compass). The reference for the north is 130 degrees.

CONCLUSION

The experiments performed were a success. Some effects that were not contemplated in here were the changes of temperature, the wind and some metallic stuff near the device that affects all sensors, especially the magnetometer.

The next step that could take place is the way to find how to send the device to the atmosphere, above the 30 kilometers and the way to supply the source of power with a battery and some other electronic stuff. In this manner, el circuit will depend only from the kind of charge of the battery.

BIBLIOGRAFÍA.

[1] Ronald T. Merril; Michel W. McElhinny; Phillip L. Mcfadden. The Magnetic Field of the Earth. Academic Press Internacional Geophysics Series; Volume 63. 1996.
 [2] Intel® Galileo Boar User Guide, March 2014, Order Number 330237-001US
 [3] Arturo Fernández Montoro; Python al Descubierto; RC Libros, 2012.
 [4] Manoel Carlos Roman; Intel Galileo and Intel Galileo Gen 2, API Features and Arduino Projects for Linux Programmers; Apress Open, 2014; ISBN 978-1-4302-6839-0.
 [5] <http://www.emutexlabs.com/ubilinx>