

Electiva Profesional - Primer Corte

1st Manuel Marquéz

Ingeniería Mecatrónica

Universidad ECCI

Bogotá, Colombia

manuelj.marquezl@ecci.edu.co

2st Julian Cantillo

Ingeniería Mecatrónica

Universidad ECCI

Bogotá, Colombia

julianh.cantillo@ecci.edu.co

3st Angel Trujillo

Ingeniería Mecatrónica

Universidad ECCI

Bogotá, Colombia

angeld.trujillo@ecci.edu.co

4st Ricardo Erez

Ingeniería Mecatrónica

Universidad ECCI

Bogotá, Colombia

ricardol.perezg@ecci.edu.co

I. RESUMEN

En este documento se desarrolla los conocimientos vistos en clase con ayuda de herramientas de desarrollo y practicas continuas de programación.

Palabras claves—Git, latex, overleaf, giroscopio, acelerómetro, programación.

II. INTRODUCCIÓN

El taller esta enfocado en cada uno de los conocimientos adquiridos en el primer corte de la asignatura Electiva Profesional en la Universidad ECCI, entre estos conocimientos encontramos:

- Raspberry Pi
- Python
- git
- Mems
- IMU

III. MARCO CONCEPTUAL

De acuerdo a lo temas ya mencionados tenemos en cuenta que:

- Raspberry Pi Foundation, Para tener una idea de que es una Raspberry Pi, debemos conocer sus inicios, Raspberry Pi Foundation fue creada en el Reino Unido sobre los años 2011 en la universidad de Cambridge, por la necesidad de crear una herramienta de enseñanza informática a bajo costo y de acceso a todo el mundo; es un concepto de un ordenador desnudo donde se intenta dejar de la forma más básica así retirar la mayoría de elementos posibles pero que permitan un funcionamiento básico de un ordenador. (Raspberry Pi Foundation, 2016) Esta fundación es de fin benéfico y uno de sus fundadores es Eben Upton extrabajador de Broadcom el cual es uno de los responsables de la arquitectura de las tarjetas Raspberry Pi. Una de las primeras tarjetas prototipo basado en ARM fue elaborada en un encapsulado de un tamaño similar a una memoria USB. Tenía un puerto USB en un extremo y un puerto HDMI en el otro.[1]
- Python, Es un lenguaje de programación de propósitos generales, fue creado a finales de 1980 por Guido van Rossum en el Centro para las Matemáticas y la Informática el cual debe a un lenguaje interpretado debido a que debe usar un compilador para ser transferido a lenguaje de máquina, dicho lenguaje es de una fácil sintaxis y



permite una mejor interpretación; Soporta orientación a objetos, programación imperativa, programación funcional es compatible con varias plataformas. como Linux, Windows y Mac.[2]

- Git es un sistema de almacenamiento y modificación en línea que analiza los cambios de los proyectos como una línea de tiempo como si fueran fotogramas; esta aplicación de Git piensa en sus datos más como una serie de instantáneas de un sistema de archivos en miniatura. Por tal cada vez que se confirma o guarda el estado del proyecto, Git básicamente toma una fotografía del estado del archivo actual según como se vean los archivos en el momento este almacena una referencia a esa instantánea. Con esto si Git no observa cambios en el archivo no lo guarda con esto es una forma eficiente de mantener sus proyectos y caso de requerirlo volver a un estado según la línea de tiempo.[3]
- Los sistemas Mems son Micro-electro-mecánicos, estos dispositivos son de pequeñas dimensiones los cuales están integrados como su nombre lo indica por elementos electrónico mecánicos y eléctricos generalmente de forma de encapsulados. Los cuales tienen distintas funciones en algunos casos interacciones con el medio comunicaciones.[4]
- Una IMU es un dispositivo formado por diferentes componentes como un acelerómetro, que medirá los cambios de la fuerza registrada producidos por el movimiento del dispositivo y cuenta con un giroscopio, el cual registrará la variación de la posición de los ejes representados por la IMU. Estos datos que serán proporcionados por la IMU, son de gran importancia para muchas disciplinas que están estrechamente relacionadas con el mundo de la topografía y la geomática, desde conocer la posición de la cámara de un avión cuando se realiza un vuelo fotogramétrico, hasta para la asistencia y control de los modernos y avanzados drones topográficos, civiles o militares y que se están convirtiendo en una herramienta muy común en nuestra sociedad. Del mismo modo, el software desarrollado para la gestión del sensor IMU proporcionará una herramienta muy útil cuyo objetivo, no es otro que ayudar a la obtención, gestión y manipulación de los datos inerciales que una IMU es capaz de ofrecernos.[5]

IV. MARCO TEÓRICO

En la practica desarrollada se tienen distintas herramientas y componentes como los son:

Cuadro I
FUNCIONES

Imagen	Nombre	Descripción
	MPU6050 - IMU	Es un dispositivo formado por diferentes componentes como un acelerómetro, un giroscopio, registrará la variación de la posición de los ejes representados por la IMU
	Jumpers	Es un conductor eléctrico (generalmente cobre) o conjunto de ellos, generalmente recubierto de un material aislante o protector, si bien también se usa el nombre de cable para transmisores de luz (cable de fibra óptica) o esfuerzo mecánico (cable mecánico).
	Tarjeta de desarrollo Mbed - STM32 L433	La placa STM32 Núcleo proporciona una forma asequible y flexible para que los usuarios prueben nuevos conceptos y construyan prototipos con el microcontrolador STM32, eligiendo entre las diversas combinaciones de rendimiento, consumo de energía y características.[6]

V. MARCO PROCEDIMENTAL

En la parte practica al tener la conexión de los Jumpers con la IMU y la tarjeta de desarrollo STM32L433,[6] a partir del código suministrado en la clase de Electiva Profesional en la Universidad ECCI: Se ingresa en la plataforma de Mbed para compilar y de este modo poder transferir el archivo a la tarjeta tal como se observa en la figura 1

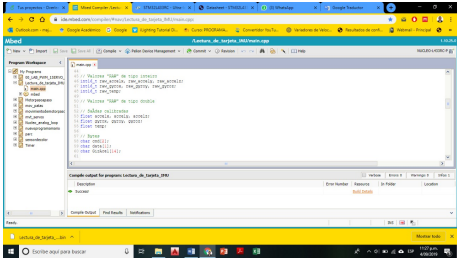


Figura 1. Plataforma Mbed

Aclaremos que en esta practica se realiza un circuito como se observa en la figura 2 donde se compone como bien lo dice el Marco Teórico de un MPU6050, la Tarjeta de Desarrollo STM y los Jumpers para la respectiva conexión.

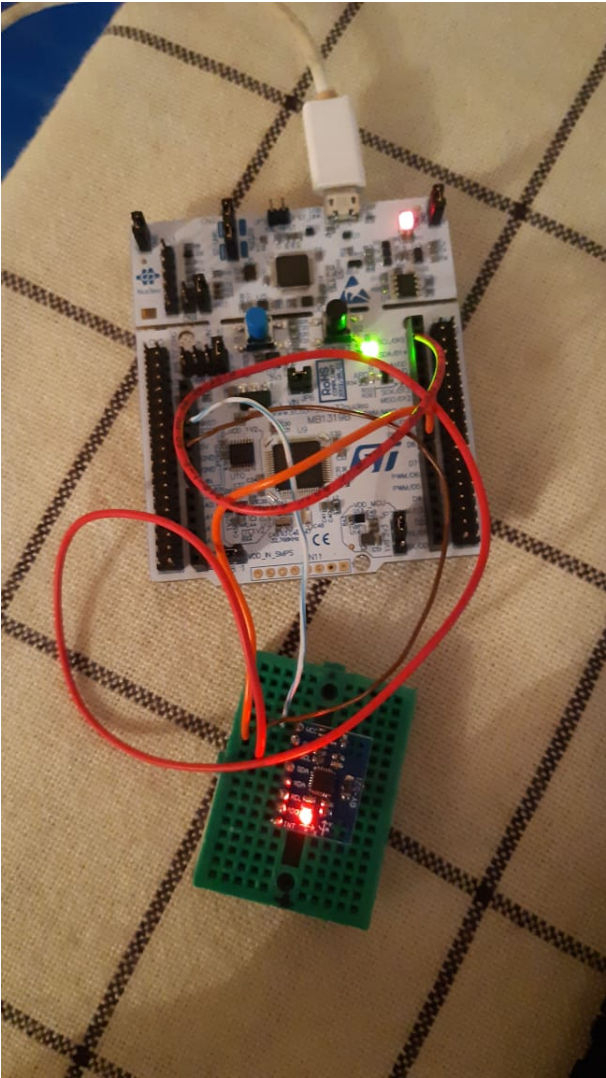


Figura 2. Circuito

Al tener el archivo compilado en la Tarjeta de Desarrollo se procede con la herramienta CoolTerm para evidenciar las lecturas de datos que se presentan de la IMU. Como se observa en la figura 3

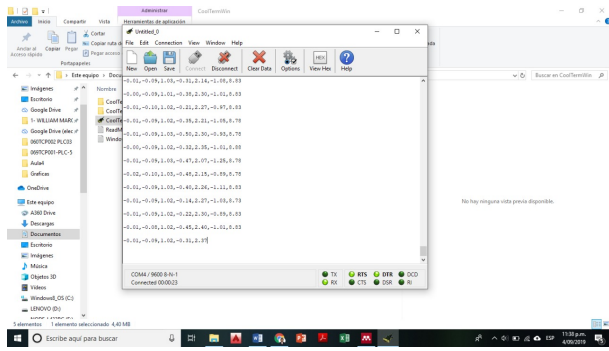


Figura 3. CoolTerm

Luego con la ayuda de la herramienta de Matlab se carga el código suministrado para la ejecución del programa y visualizar las curvas dadas por el IMU como se observa en la figura 4

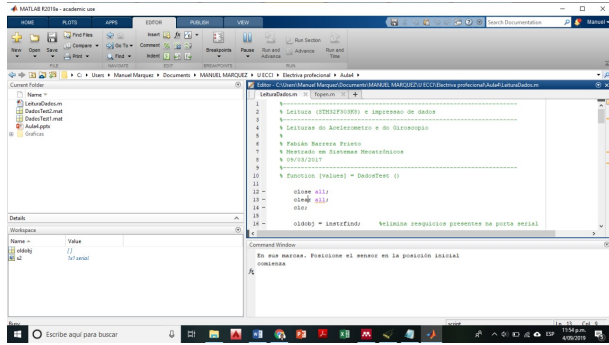


Figura 4. MatLab

En el desarrollo obtenemos las gráficas de aceleración y velocidad angular tal y como se observa en la figura 6 y la figura 7, en donde tenemos el valor de cada uno de los ejes de medida.

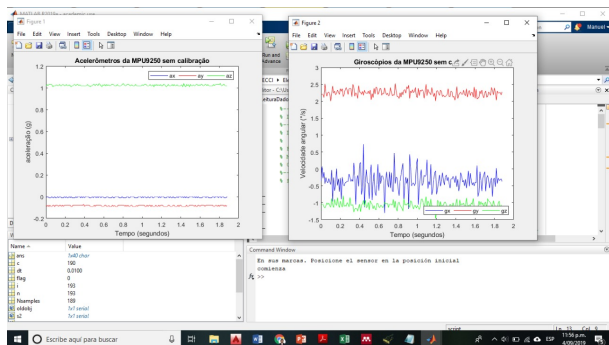


Figura 5. Matlab - Resultado

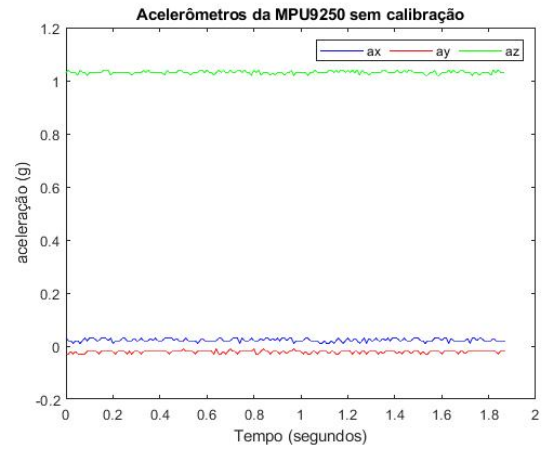


Figura 6. Gráfica de Acelerómetro

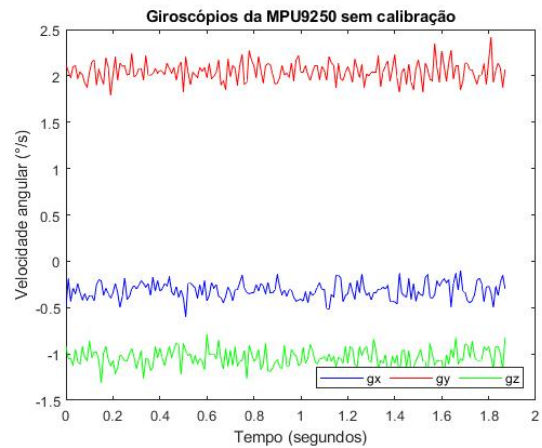


Figura 7. Gráfica del Giroscopio

VI. RESULTADOS

Dado el Marco Procedimental y las graficas obtenidad que se visualizan en la figura 6 y la figura 7

- Las graficas de la aceleracion y la velocidad angular esta dada en un intervalo de 0 a 2 segundos
- Segundo análisis
- Último análisis

VII. CONCLUSIONES

En el ejercicio de practica se observa la facilidad y la compatibilidad de las plataformas que en su interacción de trabajo, las cuales permiten visualización de los datos e interacción con el elemento IMU, en este caso herramientas Matlab y Embed.

- Los encapsulados o paquetes como los MENS facilitan el aprendizaje así como manejar y controlar la información suministrada por la IMU.
- Al generar una forma visual, en este caso gráficas se puede observar las aceleraciones y velocidades del elemento, forjando bases para generar controles y aplicación en campo.

- Identificar y conocer los componentes de sistemas de control tanto en software como en hardware, abren un ápice de curiosidad y permiten que el estudiante indague, de forma didáctica pueda plantear y experimentar la integración de todos estos elementos en aplicaciones que puedan suplir necesidades del entorno en cuestión.

REFERENCIAS

- [1] Raspberry Pi Foundation, "Buy a Raspberry Pi," 2018. [Online]. Available: <https://www.raspberrypi.org/products/>
- [2] Abellán Miguel, "¿Qué es Python? - Curso de iniciación a la programación con Python en Raspberry Pi," 2015. [Online]. Available: <https://www.programoergosum.com/cursos-online/raspberry-pi/244-iniciacion-a-python-en-raspberry-pi/que-es-python>
- [3] Git, "Una breve historia de Git," 2015. [Online]. Available: <https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git> <https://git-scm.com/book/es/v1/Empezando-Una-breve-historia-de-Git>
- [4] Saberes y Ciencias, "Sistemas Micro Electromecánicos - Saberes y Ciencias | Saberes y Ciencias." [Online]. Available: <http://saberesyciencias.com.mx/2013/02/02/sistemas-micro-electromecanicos/>
- [5] J. O. L. D. V. CALERO, "Adquisición de datos IMU en un sistema embebido," Ph.D. dissertation, UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL, 2015.
- [6] U.-l.-p. A. Cortex, M. C. U. Fpu, and U. S. B. Fs, "STM32L433xx," no. May, 2018.