

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

I. RESUMEN

A continuación se podrá evidenciar los procedimientos con los cuales se realizan las practicas respectivas del segundo corte, las cuales fueron dadas en clase; así también se podrán evidenciar los resultados de estas practicas, se analizaran de manera detallada, y las gráficas que se han obtenido mediante las pruebas de calibración de los módulos IMU (unidad de medición inercial) se verán contrastadas con las que nos otorgo el ingeniero como ejemplo en clase, La Mems la cual se encargara de adquirir los datos y traducir los cambios en los voltajes a un valor que pueda ser leíble; al igual se comentara el método de comunicación VNC, en el cual se encargara de establecer la comunicación entre la tarjeta Raspberry pi3 y un computador, este documento fue realizado en latex y se subirá mediante un repositorio git, en este caso, GitHub, con la plataforma Git Bash, la cual implementa lenguaje Linux(tipo C).

Palabras claves—Git, latex, Raspberry, giroscopio, acelerómetro, programación, calibración, IMU.

II. INTRODUCCIÓN

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

- Raspberry Pi
- Python
- Repositorios Git
- Mems
- IMU
- VNC

III. OBJETIVOS

De acuerdo a lo mencionado, el taller tiene como objetivo:

- Visualizar y comparar la toma de datos
- Realizar la calibración de la IMU
- Integrar el conocimiento adquirido para trabajar con la tarjeta Raspberry Pi
- Generar una conexión efectiva con la Raspberry Pi
- Implementar el repositorio Git

IV. 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 coste 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.[1] 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.[2]

- 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.[3]
- 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.[4]
- 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.[5]

- 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.[6]
- VNC son las siglas en inglés de Virtual Network Computing (Computación Virtual en Red). VNC es un programa de software libre basado en una estructura cliente-servidor que permite observar las acciones del ordenador servidor remotamente a través de un ordenador cliente. VNC no impone restricciones en el sistema operativo del ordenador servidor con respecto al del cliente: es posible compartir la pantalla de una máquina con cualquier sistema operativo que admita VNC conectándose desde otro ordenador o dispositivo que disponga de un cliente VNC portado. En la enseñanza, VNC sirve para que el profesor comparta su pantalla con los alumnos, por ejemplo en un laboratorio. También puede usarse para que un técnico ayude a un usuario inexperto, el técnico ve remotamente el problema del que informa el usuario. [7]

V. 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.[8]
	Raspberry Pi	Es un concepto de un ordenador desnudo donde se intenta dejar de la forma básica así retirar la mayoría de elementos posibles pero que permitan un funcionamiento básico de un ordenador.[1]

VI. MARCO PROCEDIMENTAL

■ STM32L433

En la parte practica al tener la conexión de los Jumpers con la IMU y la tarjeta de desarrollo STM32L433,[8] 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.

Aclaramos que en esta practica se realiza un circuito como se observa en la figura 1 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.

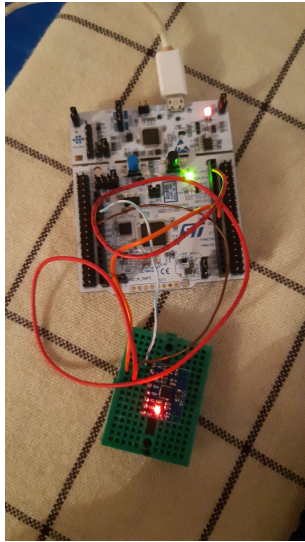


Figura 1. Circuito

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. 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.

- Sin Calibrar

En el desarrollo obtenemos las gráficas no calibradas de aceleración y velocidad angular tal y como se observa en la figura 2 y la figura 3, en donde tenemos el valor de cada uno de los ejes de medida.

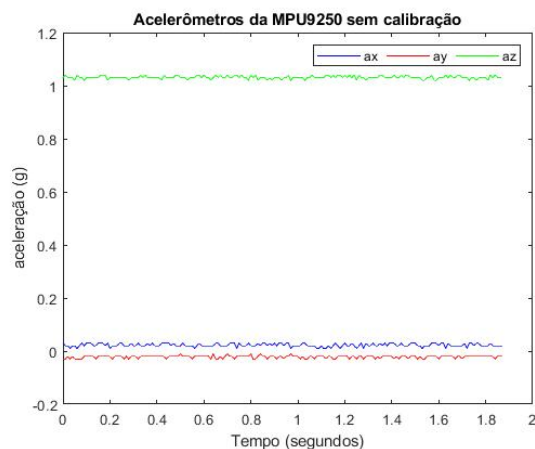


Figura 2. Gráfica de Acelerómetro Sin Calibrar

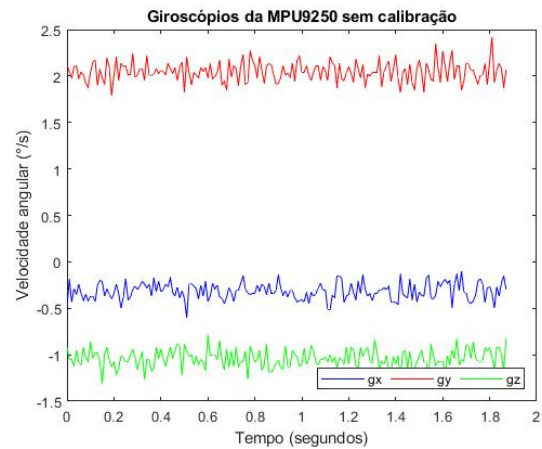


Figura 3. Gráfica del Giroscopio Sin Calibrar

- Calibrado

Se toman los datos de calibración para adicionar los offset en el nuevo código a cargar para obtener las gráficas calibradas de aceleración y velocidad angular tal y como se observa en la figura 4 y la figura 5, en donde tenemos el valor de cada uno de los ejes de medida.

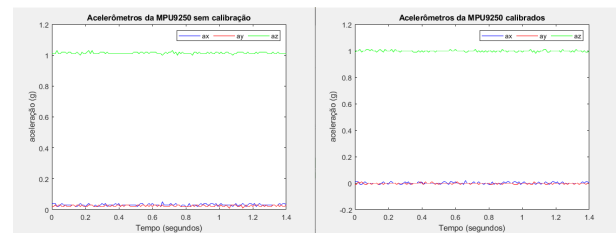


Figura 4. Gráfica de Acelerómetro Calibrada

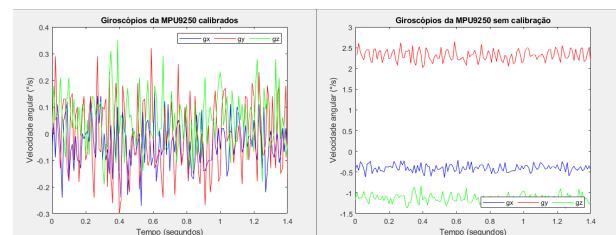


Figura 5. Gráfica del Giroscopio Calibrada

- Raspberry Pi

Inicialmente para poder trabajar en la tarjeta se debe instalar el sistema operativo con el que trabaja la misma, el cual es raspbian, teniendo en cuenta que este se encuentra gratuito [9]

se debe de realizar la configuración de la tarjeta y configurar la IP a trabajar, con esto se procede a configurar la IP en el computador.

Implementamos un protocolo VNC para poder observar las acciones que queríamos ejecutar en nuestra tarjeta

raspberry y así poder hacer un uso adecuado de la misma.
[10] Teniendo en cuenta que en el momento que se usa por primera vez esta herramienta se debe de buscar la tarjeta con la dirección IP asignada.

Otro de lo dato que se considera importante es el usuario y contraseña que se tiene por defecto, que vendrían siendo:

- Usuario: pi
- Contraseña: raspberry

VII. RESULTADOS

Dado el Marco Procedimental y las gráficas obtenidas:

- Las gráficas de la aceleración y la velocidad angular esta dada en un intervalo de 0 a 2 segundos
- Se calibraron los resultados de la IMU MPU9250, y obtuvieron mayor precisión en los datos arrojados, donde se observa en la Figura 4 que cada uno de sus ejes están mayor posicionadas.
- Se realiza un proceso de calibración y los resultados obtenidos fueron con una mejor calidad y precisión en cada uno de sus Ejes.

VIII. 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 e igual forma frente a la Raspberry.

- Los encapsulados o paquetes como los MEMS 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.
- Es muy importante realizar la calibración de nuestros componentes para garantizar la toma de datos correcta.
- Podemos integrar con distintas herramientas la conexión de la tarjeta Raspberry.
- 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, "Raspberry Pi Foundation - About Us," 2016. [Online]. Available: <https://www.raspberrypi.org/about/>
- [2] —, "Buy a Raspberry Pi," 2018. [Online]. Available: <https://www.raspberrypi.org/products/>
- [3] 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>
- [4] 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>

- [5] Saberes y Ciencias, "Sistemas Micro Electromecánicos - Saberes y Ciencias | Saberes y Ciencias." [Online]. Available: <http://saberesyciencias.com.mx/2013/02/02/sistemas-micro-electromecanicos/>
- [6] J. O. L. D. V. CALERO, "Adquisición de datos IMU en un sistema embebido," Ph.D. dissertation, UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL, 2015.
- [7] J. G. Hoyos and A. Puertas, "Montaje de un clúster raspberry para procesamiento paralelo basado en programación funcional," *CICIC 2017 - Séptima Conferencia Iberoamericana de Complejidad, Informatica y Cibernética, Memorias*, no. Cicic, pp. 142–147, 2017.
- [8] U.-l.-p. A. Cortex, M. C. U. Fpu, and U. S. B. Fs, "STM32L433xx," no. May, 2018.
- [9] R. P. Foundation, "Raspbian," 2019-09-26.
- [10] U. Gangoiti, "Control remoto multiplataforma basado en software libre : automatización de sesión VNC integrando gestión de incidencias e inventario Multiplatform remote control based on free software : Automating VNC session integrating incident management and inventor," *Control*, pp. 193–197.