

Adquisición de datos con IMU, funcionalidad general de MEMS y repositorio Git

Nicolas Gomez Sierra
Ingeniería Mecatrónica
Universidad ECCI
Bogotá, Colombia
nicolas.gomezs@ecc.edu.co

I. RESUMEN

En el presente informe, se explora el funcionamiento de un dispositivo microelectromecánico (MEMS) que, sometido a condiciones de prueba y enlazado a una tarjeta embebida, es capaz de proporcionar información exacta de posición y movimiento efectuado. De manera general, se busca conocer su configuración, estructura y control, además de la manera correcta de adquirir datos confiables en la experimentación.

Palabras claves—IMUs, MEMS, Git, Repositorio, datos, giroscopio, acelerometro

II. INTRODUCCIÓN

Los sensores son comúnmente conocidos como aparatos o dispositivos capaces de detectar estímulos. Son frecuentemente relacionados con la electrónica y demás campos de investigación; sin embargo, es claro que su base funcional y conceptual se encuentra centrada en aspectos intrínsecos de los seres vivos. Siendo animales o plantas, humanos o demás seres vivos existentes; todos son capaces de captar los estímulos y proporcionar una respuesta acorde. A manera de símil, el funcionamiento de un sensor electrónico (electromecánico, neumático y demás) se centra en la obtención constante de información proveniente del entorno físico externo. Cada sensor, al igual que los sentidos en los seres vivos, reacciona solo a estímulos para los cuales fue configurado inicialmente razón por la cual, un sensor de color es inútil ante un alza o baja de temperatura o un cambio en la humedad medioambiental. Es pues, necesario que para cada tarea exista un dispositivo capaz de asimilarla e interpretarla. Teniendo en cuenta lo anterior, conceptos como la ubicación espacial y posicionamiento, velocidades y aceleraciones y demás aspectos dinámicos, posibilitan la existencia de aparatos lectores como las IMU y sistemas compuestos como los MEMS.

III. MEMS

Los MEMS son dispositivos tecnológicos configurados como sensores, procesadores de datos y actuadores, en una escala microscópica. Generalmente, actúan como parte de un sistema mas complejo. [1] Este tipo de dispositivos (MEMS) pueden variar desde estructuras relativamente simples que no tienen ninguna parte móvil, hasta sistemas electromecánicos muy complejos en escalas diminutas. Los MEMS tiene usos

de largo alcance en aplicaciones móviles e inalámbricas, automotrices, biomédicas, industriales y energéticas y su funcionalidad es amplia (tanto como el entorno o tarea que la requiera) y sus usos y aplicaciones varían en gran manera. Sin embargo se expondrán los mas relevantes para el enfoque del presente informe. En un sistema, los circuitos integrados son la parte "pensante"(microestructura de procesamiento de datos e interpretacion de los mismos), mientras que, los MEMS complementan esa inteligencia con una percepción activa y con funciones de control.[2] Por lo cual los MEMS se podrían clasificar así:

III-A. Sensores

Aparatos diseñados para la medición, captación y detección de aspectos y variables medioambientales, tales como temperatura, humedad, color, luz, **posición**, características químicas y demás.

III-B. Actuadores

Dispositivos receptores de la señal interpretada proveniente de los procesadores. Están dispuestos para ser la respuesta al estímulo (captado por el sensor, procesado por el embebido). Generalmente operados por electrostática o térmicamente. [3] Cabe aclarar que todas sus funciones son instrucciones previamente diseñadas.

Así que, teniendo en cuenta la multifuncionalidad de los MEMS y su amplia variedad de aplicaciones en distintos campos, es necesario consolidar un aspecto de enfoque, para definir el tipo de MEMS en una tarea específica. Por lo cual, buscando entender aspectos tales como la ubicación, posicionamiento espacial y variables dinámicas que influyen en un sistema complejo, es importante resaltar el uso de un dispositivo para esta situación.[4]

IV. IMUs

Por sus siglas en ingles, Inertial Measurement Unit.(unidad de medición inercial). Un IMU es una unidad electrónica de procesamiento que obtiene información del movimiento y cinemática de un sistema estructurado. Al estar relacionado con magnitudes espaciales, de cada uno de ellos se obtendrán tres valores numéricos; siendo estas las medidas en cada uno de los ejes del espacio.[5][6]

IV-A. Acelerómetro

Mide la aceleración en cada uno de los tres ejes de coordenadas espaciales.

IV-B. Giroscopio

Son dispositivos que miden o mantienen el movimiento de rotación. Son pequeños sensores, de bajo costo para medir la velocidad angular. Las unidades de velocidad angular se miden en grados por segundo revoluciones por segundo (RPS). La velocidad angular es una medida de la velocidad de rotación.[7]

IV-C. Magnetometro

Mide la orientación angular respecto a los polos magnéticos de la tierra en sus ejes de coordenadas. Una vez comprendidas las capacidades implícitas del IMU, es necesario contextualizar este elemento al trabajo presente.

V. METODOLOGÍA

V-A. Aspectos generales

Para la adquisición de datos, sin importar el dispositivo sensorial; se involucra un sistema de procesamiento de datos, ya que la interpretación y organización de variables y datos, es programable. Tomando pues, las características esenciales del dispositivo y enfocándolas en la tarea, es necesario utilizar un sistema embebido. Para este proyecto, se hace uso de una tarjeta de desarrollo STM32. La STM32 es una familia

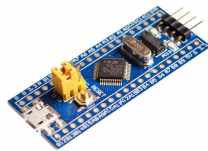


Figura 1. STM32

de microcontroladores Flash de 32 bits desarrollada por ST Microelectronics. Se basa en el procesador ARM® Cortex® y ofrece una gama de productos de 32 bits que combina un rendimiento muy alto, capacidades en tiempo real, procesamiento de señales digitales y funcionamiento de baja potencia y bajo voltaje.[8]

También se define el IMU MPU6050 para ser utilizado



Figura 2. IMU MPU6050

Así que, para configurar y programar la adquisición de datos por parte del sensor, es necesario consolidar sus características principales. Estas se encuentran en su datasheet.


Part #	Gyro Full Scale Range	Gyro Sensitivity	Gyro Rate Noise	Accel Full Scale Range	Accel Sensitivity
UNITS:	(°/sec)	(LSB/°/sec)	dps/√Hz	(g)	LSB/g
 MPU-6050	±250	131	0.005	±2	16384
	±500	65.5	0.005	±4	8192
	±1000	32.8	0.005	±8	4096
	±2000	16.4	0.005	±16	2048

Figura 3. Características principales del MPU 6050

Teniendo en cuenta las características de giro, su rango y lectura de movimiento, se inicia la programación en Mbed (plataforma de programación de microcontroladores y sistemas embebidos)

En esta parte del código se configuran las características del IMU (Conectividad y reconocimiento) junto con las características de fabricante. Se muestra la conexión del esclavo conectado reconocido dentro del código (I2C). Aquí se describen los límites del acelerómetro y giroscopio y sus respectivas escalas.

```
#include "mbed.h"
#define MPU6500_address 0xD0

#define GYRO_FULL_SCALE_250_DPS 0x00
#define GYRO_FULL_SCALE_500_DPS 0x08
#define GYRO_FULL_SCALE_1000_DPS 0x10
#define GYRO_FULL_SCALE_2000_DPS 0x18

#define CC_FULL_SCALE_2_G 0x00
#define CC_FULL_SCALE_4_G 0x08
#define CC_FULL_SCALE_8_G 0x10
#define CC_FULL_SCALE_16_G 0x18
```

Figura 4. Primer parte del código

En esta segunda parte del código, se denota los valores de conversión del acelerómetro, giroscopio y magnetómetro, dentro de sus escalas respectivas. También se definen las variables, estas son tipo RAW, por tener valores numéricos provenientes de un instrumento, en este caso el IMU MPU6050. Se asignan los nombres de acuerdo a cada eje coordenado que trabaja. Como se recordara, este IMU tiene seis grados de libertad. Se calibran las salidas y se configura un temporizador.

```
#define SENSITIVITY_ACCEL 2.0/32768.0
#define SENSITIVITY_GYRO 250.0/32768.0
#define SENSITIVITY_TEMP 333.87
#define TEMP_OFFSET 21
#define SENSITIVITY_MAGN (10.0*4800.0)/32768.0

int16_t raw_accelx, raw_accely, raw_accelz;
int16_t raw_gyrox, raw_gyroxy, raw_gyroz;
int16_t raw_temp;

char cmd[2];
char data[1];
char GirAcel[14];

float buffer[500][8];
int i;
Timer t;
float timer=0;

Serial pc(SERIAL_TX, SERIAL_RX);
I2c i2c(PB_7, PB_6);
```

Figura 5. Segunda parte del código

Sin embargo, en el caso del presente proyecto, no fue posible la ejecución y adquisición de datos puesto que no se cuenta con una tarjeta de procesamiento de los mismos. Ni arduino ni STM32 u otro sistema embebido, por lo cual, la comprobación del programa ni la adquisición de datos se pudieron generar.

Se conocen los requisitos implícitos del IMU. Siendo estos rangos, tipos de lecturas (aceleración y velocidad angular), conexiones y demas aspectos característicos esenciales para la composición del programa.

Se realiza un repositorio usando el programa GIT, para dar la entrega del informe. Es necesaria la asimilacion de conceptos a un nivel mas amplio dentro de Git; por parte del escritor del presente informe, puesto que fue bastante dificil lograr internar la informacion dentro del programa.

Se comprendern las conexiones de multiples circuitos con distintas funciones (I2C), viendo en el programa la programacion para acoplar varios a traves de dichas conexiones.

ción dentro de sus rangos, dando un límite de 300 muestras por paso de la IMU. También, da respuesta una vez que culmina la toma de muestras; presenta los resultados de aceleración, velocidad angular y el tiempo en la toma de muestras.

REFERENCIAS

- [1] “Definicion y breve historia de los mems,” <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/213/4/A4.pdf>, consultado: 2020-09-03.
- [2] “Mems (microelectromechanical systems),” <https://www.etsist.upm.es/estaticos/ingeniatic/index.php/tecnologias/item/516-mems-microelectromechanical-systems.html>.
- [3] “Tecnologia mems,” <http://bibing.us.es/proyectos/abreproy/4966/fichero/e.+Tecnologia+>, consultado: 2020-09-03.
- [4] “Sensores mems,” <https://www.interempresas.net/Electronica/Articulos/170292-Sensores-MEMS-de-Bosch.html>, consultado: 2020-09-03.
- [5] “Introduccion al imu,” <http://blascarr.com/lessons/introduccion-al-imu-sistemas-de-navegacion-inercial/>, consultado: 2020-09-03.
- [6] “Unidad de medicion inercial,” [https://es.wikipedia.org/wiki/Unidad de medicion inercial](https://es.wikipedia.org/wiki/Unidad_de_medicion_inercial), consultado: 2020-09-03.

“Giroscopio,” https://www.5hertz.com/index.php?route=tutoriales/tutorialtutorial_id=13, consultado : 2020-09-03.

“tutorial stm,” <https://riptutorial.com/es/stm32>, consultado: 2020-09-03.

“Microelectromecanicos,” https://es.wikipedia.org/wiki/Sistemas_microelectromecanicos, consultado: 2020-09-03.

“microprocesadores,” <https://es.electronics-council.com/integrating-mems-with-dsps-microprocessors-63802>, consultado: 2020-09-03.

“Datasheet,” <https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>, consultado: 2020-09-03.

“raw,” <https://www.nlsinfo.org/content/cohorts/nlsy97/using-and-understanding-the-data/types-variables-raw-symbols-rosters-created>, consultado: 2020-09-03.

“i2c,” <http://robots-argentina.com.ar/didactica/descripcion-y-funcionamiento-del-bus-i2c/:text=DEFINICION>, consultado: 2020-09-03.

Una vez culminada la compilación del código y posterior descarga en la tarjeta STM32, se integra el programa y se ejecuta. A través del programa Coolterm (interfaz de lectura de datos) se observa las variables generadas (300 muestras) en sus tiempos de muestra y dentro de sus rangos de medida.

VI. CONCLUSIONES

Se conoce el funcionamiento general de los MEMS, se hace la configuración de uno de ellos, el IMU MPU6050, se programan sus rangos y especificaciones de acuerdo al datasheet del sensor en cuestión. Finalmente, se ejecuta el programa y se analizan los resultados múltiples obtenidos.