



Universidad de Oviedo

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

**GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y
AUTOMÁTICA**

ÁREA DE INGENIERÍA ELECTRÓNICA

***EXPLOTACIÓN DE UN SENSOR IMU PARA LA
MONITORIZACIÓN EN TIEMPO REAL***

MARCOS GONZÁLEZ CAMÍN

TUTOR: D. JUAN CARLOS ÁLVAREZ ÁLVAREZ

FECHA: Julio 2024



Índice

Índice de figuras	4
1. Introducción.....	6
2. Objetivos y alcance.....	7
3. Sensores inerciales.....	8
3.1.- SENSOR	8
3.2.- ACELERÓMETRO	9
3.2.1.- Acelerómetros mecánicos	11
3.2.2.- Acelerómetros resistivos.....	11
3.2.3.- Acelerómetros capacitivos.....	11
3.2.4.- Acelerómetros piezoeléctricos	12
3.2.5.- Acelerómetros micromecanizados (MEMS)	12
3.3.- GIROSCOPIO	12
3.3.1.- Giroscopios mecánicos	14
3.3.2.- Giroscopios ópticos	14
3.3.3.- Giroscopios SAW	14
3.3.4.- Giroscopios MEMS	14
3.3.- COMUNICACIÓN I2C	15
3.4.- COMUNICACIÓN SPI	17
4. Desarrollo del prototipo.....	20
4.1- IMU.....	20
4.2- PLACA BASE	21
4.3- TARJETA SD Y SOFTWARE UTILIZADO	22
4.4- ENSAMBLAJE	24
5. Resultados	27
5.1- MEDICIONES EN ESTÁTICO	27



5.2- MEDICIONES EN MOVIMIENTO	34
5.2.1.- Movimiento controlado.....	34
5.2.2.- Movimiento aleatorio.....	38
5.3- COMPARACIÓN CON SENSOR COMERCIAL	41
5.3.1.- Medición en estático	43
5.3.2.- Medición en dinámico	47
5.4- DISCUSIÓN FINAL	52
6. PRESUPUESTO.....	54
6.1.- COSTES DE MANO DE OBRA.....	54
6.2.- COSTES DE MATERIAL.....	55
6.3.- COSTES DEL SOFTWARE	55
6.4.- COSTES TOTALES.....	56
7. Conclusiones	57
8. Bibliografía	58
ANEXO: MANUAL DE USUARIO.....	59

Índice de figuras

Figura 3.1.- Acelerómetro básico.....	10
Figura 3.2.- Ejemplo de MEMS.....	15
Figura 3.3.- Bus I2C.....	16
Figura 3.4.- Secuencia I2C.....	17
Figura 3.5.- Comunicación SPI.....	18
Figura 4.1.- Pines y forma de la IMU utilizada	21
Figura 4.2.- Foto de la placa base STM32F030R8	22
Figura 4.3.- Pines y forma del módulo PmodMicroSD	23
Figura 4.4.- Ejemplo de archivo de texto con datos recogidos sin procesar	23
Figura 4.5.- Dibujo que engloba todas las conexiones a realizar en el montaje.	24
Figura 4.6.- Pinout del Núcleo F030R8	25
Figura 5.1.- Resultados obtenidos del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática en una posición casi horizontal en la que el eje x es el afectado por la gravedad.....	29
Figura 5.2.- Resultados del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática sobre la mesa en una posición en la que el eje y es el afectado por la gravedad....	31
Figura 5.3.- Resultados del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática sobre la mesa en una posición en la que el eje y es el afectado por la gravedad....	32
Figura 5.4.- Comparación entre los datos obtenidos por el giroscopio en 2 de los casos anteriores	33
Figura 5.5.- Resultado del acelerómetro y giroscopio en el experimento donde se daban golpes a una caja situada encima de una mesa	35
Figura 5.6.- Gráfica de la figura 5.5 ampliada para ver con mayor claridad el suceso.	36
Figura 5.7.- Representación de la gráfica de la figura 5.6, pero ahora en el tiempo, para poder medir el desfase entre ambos en segundos.	37
Figura 5.8.- Resultados obtenidos de realizar el movimiento aleatorio.....	39
Figura 5.9.- Representación de los tres ejes del acelerómetro por separado para su correcta visualización.	40



Figura 5.10.- Representación de los tres ejes del acelerómetro por separado para su correcta visualización. 41

Figura 5.11.- Disposición de los sensores para el experimento, unidos mediante cinta adhesiva para simular que solo existe uno..... 42

Figura 5.12.- Resultados del acelerómetro de la medición en estático con los sensores apoyados en una dirección (eje X para la IMU y eje Z para Xsens). 44

Figura 5.13.- Resultados del giroscopio de la medición en estático con los sensores apoyados en una dirección (eje X para la IMU y eje Z para Xsens) 44

Figura 5.14.- Resultados del acelerómetro de la medición en estático con los sensores apoyados en una dirección (eje Z para la IMU y eje Y para Xsens) 45

Figura 5.15.- Resultados del giroscopio de la medición en estático con los sensores apoyados en una dirección (eje Z para la IMU y eje Y para Xsens) 45

Figura 5.16.- Dibujo básico para explicar el movimiento realizado 47

Figura 5.17.- Comportamiento esperado de un sensor cuando, partiendo del reposo se acelera hacia abajo y luego se frena. 48

Figura 5.18.- Comportamiento esperado de un sensor cuando, partiendo del reposo se acelera hacia arriba y luego se frena..... 48

Figura 5.19.- Resultado del proceso continuo..... 49

Figura 5.20.- Resultados obtenidos directamente por el acelerómetro de ambos sensores, sin tratar, a distintas frecuencias en el eje donde se produjo el movimiento. 50

Figura 5.21.- Resultados obtenidos por el acelerómetro de ambos sensores en el eje donde se produjo el movimiento, habiendo procesado los datos para poder comparar con el tiempo. 51



1. Introducción

El presente documento describe el desarrollo y testeo de un prototipo de sensor para la medida del movimiento basado en una unidad de medición inercial (Inertial Measurement Unit).

Las unidades de medición inercial están formadas principalmente por acelerómetros y giroscopios, los cuales obtienen, mediante uno o varios ejes ortogonales, datos que posteriormente envían a sistemas capaces de leerlos. Realizan un tratamiento de los datos necesario para poder obtener los valores de aceleración/velocidad y rotación.

Este sistema es muy utilizado en la actualidad, como por ejemplo en los teléfonos móviles, para detectar su orientación y rotar la pantalla acorde a la misma. Los drones utilizan las IMU para conocer su orientación en el espacio. Muchos dispositivos *wearables*, como relojes o anillos incorporan un IMU para medir la actividad física.

Para llevar a cabo este trabajo fue necesario poner en práctica aptitudes que se han ido impartiendo en los cursos del grado, como lenguajes de programación o saber manejar dispositivos físicos, pero también ha servido como enseñanza personal, ya que tanto los temas como algunas partes de la programación no se habían impartido nunca en el grado.



2. Objetivos y alcance

El objetivo del proyecto será realizar el montaje de un sistema que permita poder obtener valores y mediciones de una unidad de medida inercial.

Se comenzará con un estudio teórico sobre las partes fundamentales de la IMU, como son el acelerómetro y el giroscopio.

Una vez vista la teoría, se explicarán cuáles son los dispositivos utilizados en todo el proyecto, describiéndolos por separado para tener una idea global del funcionamiento y cómo funciona todo el conjunto.

La segunda parte del proyecto será, una vez conocidos todos los componentes, realizar la programación necesaria para poder llegar a realizar los experimentos requeridos. Esto se realizará mediante comunicación tanto SPI, para poder sacar los valores a una tarjeta SD, como comunicación I2C, para la programación del sensor.

Con los datos, se tratará de sacar unidades que sean del Sistema Internacional para poder ser interpretadas. Estas unidades son, (m/s²), para la aceleración, y (°/segundos) para la velocidad angular del giroscopio.

Este proyecto tiene muchas salidas a futuro, ya que una vez se dispone de una unidad de medida inercial completamente funcional, las posibilidades son infinitas. Podría continuarse en el futuro por alumnos o instituciones para conseguir un dispositivo con capacidad de poder ser utilizado en medicina, para realizar ensayos en pacientes que puedan presentar problemas motrices. Con este sistema se podría detectar problemas motrices no solo en pacientes, si no en personas que podrían sufrirlo y a las que una detección a tiempo permitiría prevenir problemas futuros.



3. Sensores inerciales

Una unidad de medición inercial es un dispositivo que, mediante acelerómetro y giroscopios puede medir velocidades, orientaciones y fuerzas gravitacionales. Para entender un poco mejor el funcionamiento hay que explicar qué son acelerómetros y giroscopios.

Estos dispositivos varían la posición relativa de las placas de un microcontrolador cuando se encuentra sometido a aceleraciones.

3.1.- SENSOR

Un sensor es un dispositivo que permite medir fenómenos físicos, químicos o biológicos. El transductor es la parte del sensor en la que se realiza una conversión de energía asociada a la magnitud de interés. Todo sensor incluye alguna acción de transducción, inherente o introducida *ad hoc*. Ella transforma los cambios en lo medido en salidas eléctricas/ópticas, más convenientes para su tratamiento posterior. Otros componentes de un sensor son las etapas de amplificación, filtros, conversores A/D, etapas de compensación, etc.

En robótica se emplean muchos tipos de sensores, especialmente para medir distancias, orientaciones y desplazamientos. Y debido a sus imprecisiones o carencias ante circunstancias adversas (tipo de iluminación, propiedades del ambiente y los obstáculos, etc.), pueden hacer recomendable el empleo simultáneo de múltiples sensores agrupados en sistemas de percepción.



3.2.- ACELERÓMETRO

Los acelerómetros son dispositivos electrónicos muy utilizados en la vida real. Determinan la posición de un objeto en el espacio y monitorizan su movimiento gracias a que miden las fuerzas de aceleración actuantes sobre él.

La medida de la aceleración, además de ser un elemento central en los sistemas de dirección inerciales, tiene aplicación en una amplia variedad de problemas tanto industriales como comerciales. Por ejemplo, la detección de choques en vehículos para la activación de los air-bags, análisis de vibración en maquinaria industrial, proporcionar señales de realimentación para estabilizar la imagen de una grabadora de vídeo debido a la vibración producida por la mano, etc.

Los acelerómetros son dispositivos para medir aceleración y vibración. Estos dispositivos convierten la aceleración de la gravedad o de movimiento en una señal eléctrica analógica proporcional a la fuerza aplicada al sistema, o mecanismo sometido a vibración o aceleración. Esta señal analógica indica en tiempo real la aceleración instantánea del objeto sobre el cual el acelerómetro está montado.

Los primeros sensores de aceleración eran unos sistemas muy complejos y no muy fiables que se basaban en la medida de los desplazamientos de una masa inercial sujeta a la aceleración con resortes que contrarrestaban el efecto de la fuerza generada por la masa.

Los acelerómetros son sensores inerciales que miden la segunda derivada de la posición. Un acelerómetro mide la fuerza de inercia generada cuando una masa es afectada por un cambio de velocidad. Esta fuerza puede variar la tensión de un muelle, deformar un elemento, o modificar la frecuencia de vibración de una masa.

El principio básico de funcionamiento de los acelerómetros es la detección de la fuerza ejercida en una masa con limitación elástica.

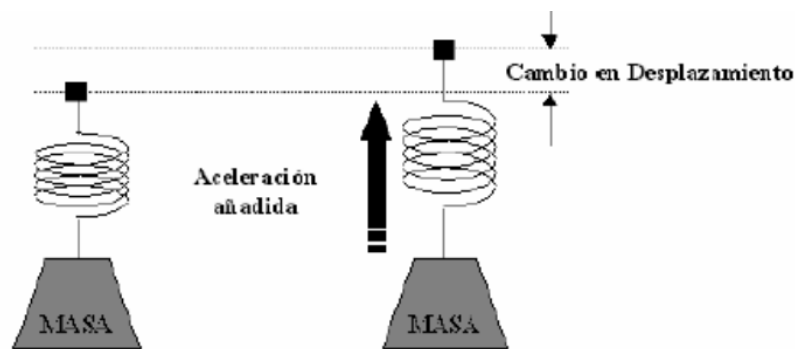


Figura 3.1.- Acelerómetro básico

$$F = m * a \quad (3.1)$$

Consideremos un sistema mecánico simple, que consiste en una masa fija m , con un muelle con una rigidez k (constante). Si la masa se desplaza una distancia x , la aceleración debido a la fuerza restauradora del muelle es:

$$F = k * x \quad (3.2)$$

$$a = \frac{k * x}{m} \quad (3.3)$$

Igualando las ecuaciones 3.1 y 3.2 obtenemos como resultado la ecuación 3.3 y se puede derivar la magnitud de la aceleración observando el desplazamiento x de la masa fija.

Este principio fundamental se utiliza hasta en el más sofisticado y caro acelerómetro electromecánico, e incluso en modernos acelerómetros micromecanizados

Existen diferentes tipos de acelerómetros según la tecnología: mecánicos, resistivos, capacitivos, piezoeléctricos, y micromecanizados (MEMS).



3.2.1.- Acelerómetros mecánicos

Se utilizan galgas extensiométricas (matriz de bobinas la cual varía en función de la carga aplicada sobre ellas) para medir los cambios, con sistemas de amortiguación para evitar la propia oscilación.

También son empleados en sistemas que originan movimientos oscilatorios cuando están sometidos a aceleración (servoacelerómetros) o detectan el desplazamiento de una masa inerte mediante cambios en la transferencia de calor (acelerómetros térmicos).

3.2.2.- Acelerómetros resistivos

El funcionamiento de estos acelerómetros se basa en que el material se deforma mecánicamente cuando las resistencias eléctricas cambian su valor.

En lugar de tener un cristal piezoeléctrico, como los sensores piezoeléctricos, tienen sustrato formando parte de un circuito que mide la intensidad de corriente mediante un puente de Wheatstone.

3.2.3.- Acelerómetros capacitivos

Son sensores formados por un conjunto de electrodos fijos y placas unidas a la masa de prueba, sujetas solo por sus extremos para tener libertad de movimiento. Están integrados en chips de silicio.

Estos dispositivos varían la posición relativa de las placas de un microcondensador cuando se encuentra sometido a aceleraciones.



3.2.4.- Acelerómetros piezoeléctricos

El funcionamiento de estos acelerómetros está basado en las propiedades de los cristales piezoeléctricos.

El cristal modifica su estructura cristalina cuando sufre una deformación física, con lo que cambian sus características eléctricas. Dicha deformación física puede deberse a una fuerza, la cual produce una corriente eléctrica. Se puede obtener la aceleración midiendo esa corriente.

3.2.5.- Acelerómetros micromecanizados (MEMS)

Este es el tipo de acelerómetro que se va a utilizar en este trabajo. Son los sensores más baratos, más pequeños y comunes. Una de sus ventajas es que se pueden implementar directamente en la placa de circuito impreso.

Se denomina MEMS a una tecnología de base que se utiliza para crear dispositivos diminutos. Estos acelerómetros disponen de resortes colocados en los diferentes ejes, los cuales están sujetos por un lado y en el otro unidos a unos pesos.

Cuando una fuerza actúa sobre el sensor, el peso se mueve sobre los resortes, lo que provoca un cambio en la distancia entre el elemento de condensación y la masa y por tanto afecta el cambio en la capacidad. Analizando el movimiento de los pesos se obtiene la aceleración.

3.3.- GIROSCOPIO

Un giroscopio es un dispositivo mecánico que utiliza el principio de conservación del momento angular para medir o mantener la orientación.



A diferencia de los acelerómetros, que miden la aceleración lineal, los giroscopios miden la velocidad angular.

Es un dispositivo con un diseño que le permite mantener siempre su orientación independientemente de cualquier movimiento o fuerza que se le aplique.

Para ello tiene una peculiar forma. Consta de un cuerpo que siempre está en rotación sobre un eje de simetría, sobre una base. Con ello obtiene velocidad angular. Las unidades de velocidad angular se miden en grados por segundo ($^{\circ}/s$) o revoluciones por segundo (RPS). La velocidad angular es, simplemente, la medición de la velocidad de rotación.

El giroscopio que viene en la unidad de medición inercial es un giroscopio analógico que, al detectar velocidades de rotación, nos devuelve la información de forma analógica para poder procesarla. La unidad mide pequeñas señales de corriente que manda el giroscopio a medida que cambia la velocidad angular.

Un giroscopio es un dispositivo que utiliza la gravedad de La Tierra para ayudar a determinar la orientación. Estos sensores son dispositivos que detectan la velocidad angular, el cambio en el ángulo rotacional por unidad de tiempo.

Su diseño consiste en un disco que rota libremente, llamado rotor, montado en un eje giratorio en el centro de una rueda más grande y estable. A medida que el eje da vueltas, el rotor permanece quieto para indicar la acción gravitacional central y, en consecuencia, la dirección hacia abajo.

El giroscopio mantiene su nivel de efectividad al ser capaz de medir la tasa de rotación en torno a un eje particular. Al medir la tasa de rotación en torno al eje de balanceo, identifica un valor actual hasta que el objeto se estabiliza. Usando los principios básicos del momento angular, el giroscopio ayuda a indicar la orientación.

Los giroscopios están disponibles para medir la velocidad rotacional en una, dos o tres direcciones. Los giroscopios de tres ejes a menudo se implementan con acelerómetros de tres ejes para ofrecer un sistema de seguimiento del movimiento con un total de 6 grados de libertad, lo cual es el caso de este proyecto.



3.3.1.- Giroscopios mecánicos

Un rotor simple gira sobre un eje, y su momento angular se mantiene constante en el espacio. Cuando se aplica un momento externo, el giroscopio responde con un movimiento de rotación alrededor de un eje perpendicular al momento aplicado. Estos giroscopios son comúnmente utilizados en aplicaciones educativas y juguetes.

También existe la opción de tener dos rotores girando en direcciones opuestas, estos giroscopios ofrecen mayor estabilidad.

3.3.2.- Giroscopios ópticos

Utilizan la interferometría para medir la rotación angular. La luz se envía a través de una fibra óptica en dos direcciones, y la diferencia en la fase de la luz que regresa se utiliza para calcular la velocidad angular. Estos giroscopios son compactos y se utilizan en dispositivos electrónicos portátiles y sistemas de navegación.

3.3.3.- Giroscopios SAW

Estos giroscopios miden la rotación utilizando ondas acústicas superficiales. Cuando el giroscopio rota, la frecuencia de las ondas acústicas se ve afectada, permitiendo la medición de la velocidad angular. Son adecuados para aplicaciones en las que el tamaño y la eficiencia energética son críticos.

3.3.4.- Giroscopios MEMS

El giroscopio MEMS de tres ejes puede medir la rotación alrededor de tres ejes: x,y,z. Es bastante pequeño (puede medir entre 1 y 100 micrómetros). Cuando se hace girar el

giroscopio, una pequeña masa se desplaza a medida que cambia la velocidad angular. Este movimiento se convierte en señales eléctricas de muy baja corriente que pueden ser amplificadas y leídas por un microcontrolador.

Un ejemplo de MEMS, lo utilizado en este proyecto tanto para acelerómetro como para giroscopio (vienen ambos integrados en el mismo sensor) sería el de la figura 3.2

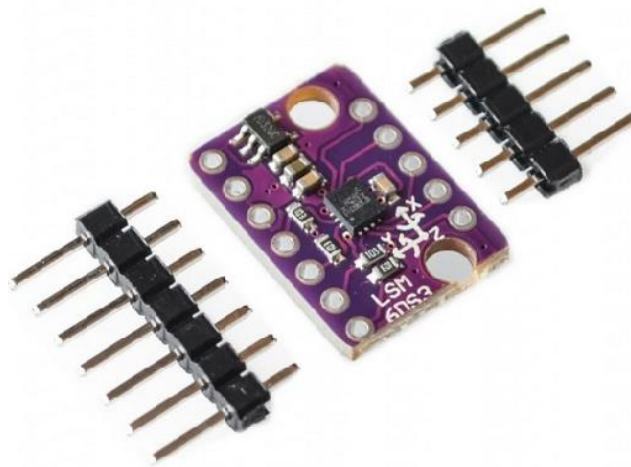


Figura 3.2.- Ejemplo de MEMS

3.3.- COMUNICACIÓN I2C

I2C es un protocolo de comunicación el cual utiliza un bus de comunicaciones de dos hilos. A través de estos dos hilos se pueden conectar dispositivos. A esos dispositivos se les llamará maestros o esclavos, en función del uso que tengan.

Es muy importante utilizar resistencias *pull up* para que cuando no haya dispositivos conectados al bus se mantenga un nivel alto.

Dado que todos los dispositivos conectados al bus se comunican por los mismos dos hilos, para saber exactamente con cuál conectarnos, a cada dispositivo se le asigna una dirección.

El maestro es el encargado de iniciar la conversación, decidir con cual o cuales de los esclavos se quiere comunicar y de qué forma (si es para enviar datos o para recibirlos) y es el encargado de finalizar la comunicación.

Estos dos hilos del bus de comunicación son líneas de colector abierto. En una línea se envía la señal del reloj (denominada SCL) y en la otra línea se envían los datos (denominada SDA).

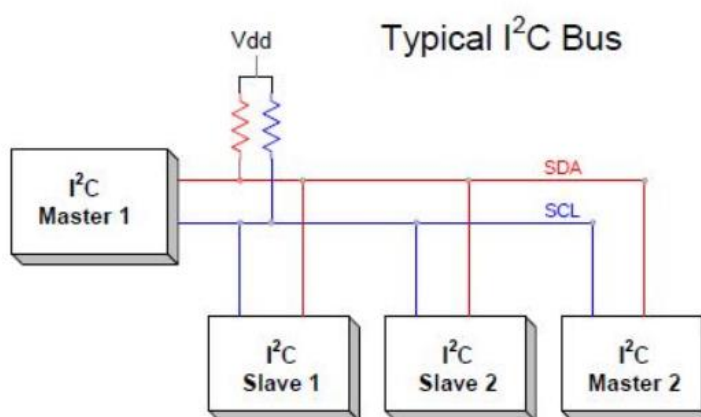


Figura 3.3.- Bus I2C

En la figura 4.4 se observa una posible conexión de un sistema con un maestro y varios esclavos.

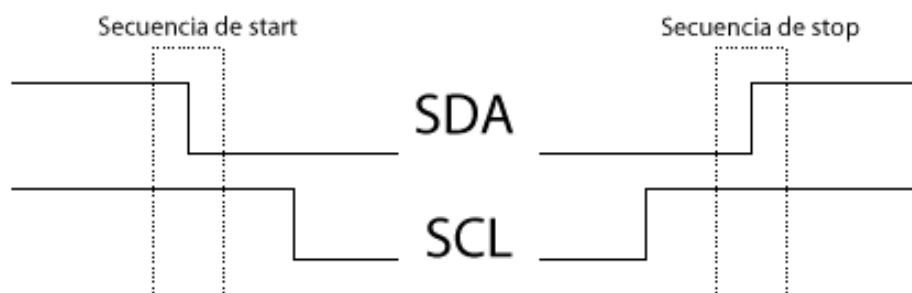


Figura 3.4.- Secuencia I2C

Como ya se ha comentado, la transmisión la empieza el maestro, con un bit de *start*, enviado por la línea SDA. Una vez recibido dicho bit por los esclavos se empieza la comunicación. Los siguientes datos que el maestro envía son los de la dirección del esclavo a el que va dirigido el mensaje, por lo que estos datos ya solo llegan a uno.

Después de los bits que contienen la dirección, se envía un bit el cual le indica al esclavo si lo que el maestro quiere es leer o escribir (Read/Write).

Seguidamente, el esclavo solicitado debe enviarle de vuelta al maestro un bit indicando la confirmación y que se queda a la espera de más indicaciones. Este bit es muy importante ya que si el maestro no recibe este bit, se interrumpe la comunicación.

Una vez recibida la confirmación, ya se pueden comunicar entre ellos. Para finalizarla, el maestro debe enviar el bit de *stop*.

3.4.- COMUNICACIÓN SPI

SPI es un protocolo de comunicación síncrono utilizado para la comunicación serial entre dispositivos. Utiliza full dúplex, que significa que los datos viajan en ambas direcciones simultáneamente.

Debido a que trabaja en full dúplex es una comunicación con una velocidad de transmisión muy alta.

En esta comunicación se utilizan cuatro señales:

- SCLK (Clock): Este reloj marca la sincronización. Con cada pulso se lee o se envía un bit.
- MOSI (Master Output Slave Input): Salida de datos del *Master* y entrada de datos al Esclavo,
- MISO (Master Input Slave Output): Salida de datos del Esclavo y entrada al *Master*.
- SELECT: Esta señal sirve para seleccionar un Esclavo, o para que el *Master* le diga al Esclavo que se active.

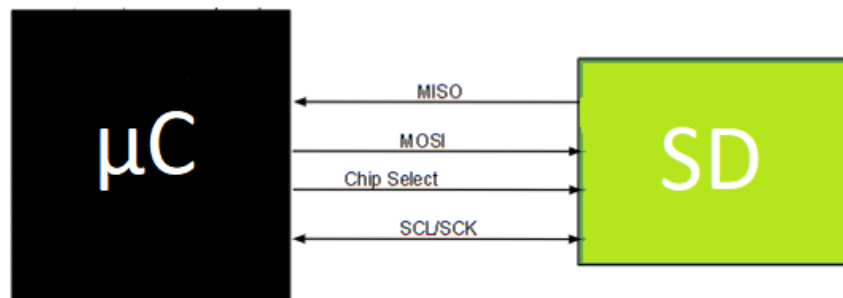


Figura 3.5.- Comunicación SPI

Es una comunicación síncrona por que la cadena de bits es enviada de manera síncrona con los pulsos del reloj.

En la figura 4.6 se tiene el ejemplo de una comunicación SPI, donde el microcontrolador es el maestro y una tarjeta SD el esclavo. No tiene por qué haber un solo esclavo, el protocolo permite tener varios.

La transmisión empieza cuando el maestro baja la señal *Select* a cero, activando al esclavo. Luego puede empezar a comunicarse con él.

La señal *Select* permite tener una comunicación con varios esclavos. Cada esclavo tiene una propia.

Las otras tres señales se pueden conexas dependiendo de si la conexión es en serie o en paralelo. Puede darse la opción de que cada esclavo tenga modos de configuración o



frecuencias diferentes. Es el maestro el que tiene que ir adaptándose a ellos en función de con cual esté comunicándose.



4. Desarrollo del prototipo

En este capítulo se explican todos los componentes que forman partes del proyecto, junto con sus características y configuraciones utilizadas. Los principales son la IMU, la placa base que contiene el microcontrolador y la tarjeta SD que se va a utilizar para guardar los datos recogidos.

4.1- IMU

El componente más importante de este proyecto es la unidad de medición inercial LSM6DS3. Combina un acelerómetro de tres ejes y un giroscopio de tres ejes. A continuación se detallan sus principales características y las configuraciones específicas utilizadas en el experimento:

- IMU del fabricante STMicroelectronics.
- Modelo LSM6DS3.
- Acelerómetro:
 - Rango de medición disponible: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$.
 - Rango de medición utilizado: $\pm 4g$.
 - Frecuencia de muestreo disponible hasta 6.7 kHz.
 - Frecuencia de muestreo utilizada: 104 Hz.
 - Sensibilidad de 0.122 mg/LSB para $\pm 4g$ escogido.
- Giroscopio:
 - Rango de medición disponible: ± 125 dps, ± 245 dps, ± 500 dps, ± 1000 dps, ± 2000 dps.
 - Rango de medición utilizado: ± 2000 dps.
 - Frecuencia de muestreo disponible hasta 6.7 kHz.
 - Frecuencia de muestreo utilizada: 104 Hz.
 - Sensibilidad de 70 mdps/LSB para ± 2000 dps escogido.

- Interfaces de comunicación:
 - I2C (Interfaz de comunicación escogida).
 - SPI.



Figura 4.1.- Pines y forma de la IMU utilizada

4.2- PLACA BASE

Esta placa base se utilizó para adquirir y procesar los datos del sensor IMU. A continuación se detallan sus características principales y las configuraciones específicas utilizadas en el experimento:

- Placa base del fabricante STMicroelectronics.
- Modelo STM32f030R8.
- Núcleo ARM Cortex-M0.
- Frecuencia de operación hasta 48 MHz.
- Memoria:
 - Flash de 64KB.
 - SRAM de 8KB.
- Interfaces de comunicación:
 - I2C (para comunicación con la IMU).
 - SPI (para comunicación con la SD).

- USART.
- GPIO.
- 55 pines de entrada/salida.

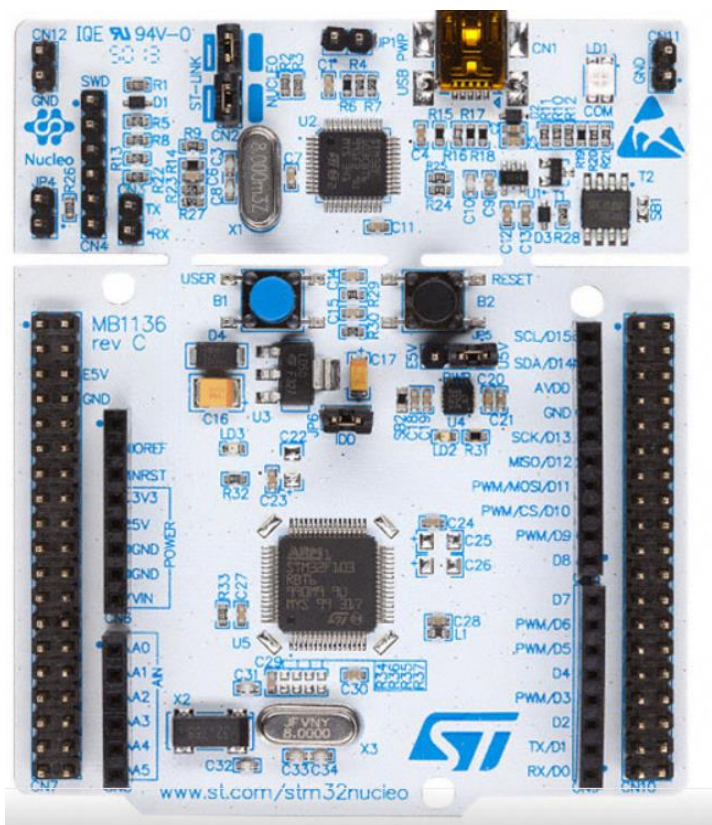


Figura 4.2.- Foto de la placa base STM32F030R8

4.3- TARJETA SD Y SOFTWARE UTILIZADO

Para poder guardar los datos obtenidos de la IMU, se necesita un sistema de almacenamiento. Para ello se utilizará una tarjeta SD de la marca RS.

Para poder leer y escribir en dicha tarjeta, es necesario un módulo que se pueda conectar a la placa base. Para ello se utilizará un módulo PmodMicroSD.

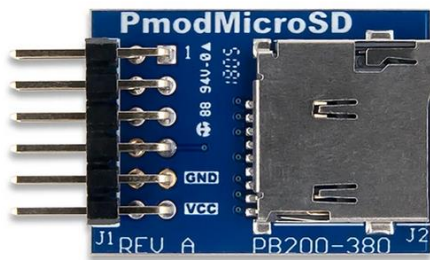


Figura 4.3.- Pines y forma del módulo PmodMicroSD

LECTURA1: Bloc de notas

Archivo Edición Formato Ver Ayuda

-486	-8263	-64	-13	-11	6	-1684
-478	-8261	-74	-13	-11	5	-1681
-475	-8264	-87	-14	-11	6	-1694
-483	-8263	-68	-13	-11	6	-1694
-499	-8265	-61	-14	-10	6	-1688
-509	-8268	-50	-14	-11	5	-1697
-498	-8267	-48	-13	-11	6	-1697
-488	-8262	-53	-14	-11	5	-1689
-466	-8255	-64	-14	-11	6	-1682
-469	-8255	-69	-14	-11	5	-1682
-475	-8257	-77	-13	-11	6	-1693
-490	-8268	-69	-14	-10	6	-1693
-503	-8268	-61	-13	-10	5	-1685
-482	-8257	-52	-13	-11	6	-1685
-482	-8262	-57	-14	-10	6	-1685
-483	-8269	-48	-14	-11	5	-1692
-497	-8261	-56	-13	-11	5	-1683
-503	-8270	-72	-14	-11	5	-1683
-501	-8272	-82	-14	-11	5	-1684
-476	-8252	-91	-14	-11	5	-1691
-470	-8252	-91	-14	-11	6	-1691
-473	-8255	-87	-14	-10	6	-1689
-490	-8259	-49	-14	-10	6	-1690
-507	-8266	-42	-14	-11	6	-1690
-511	-8272	-36	-13	-10	5	-1687
-486	-8262	-63	-14	-12	6	-1690
-466	-8251	-94	-14	-11	6	-1681
-468	-8251	-95	-15	-11	6	-1680
-483	-8261	-87	-15	-10	6	-1680
-511	-8271	-57	-14	-11	6	-1689
-522	-8274	-43	-14	-11	5	-1687
-515	-8278	-37	-14	-11	6	-1687
-486	-8263	-42	-14	-12	5	-1695
-470	-8254	-49	-14	-12	5	-1687
-459	-8254	-59	-14	-11	6	-1687

Figura 4.4.- Ejemplo de archivo de texto con datos recogidos sin procesar

El software utilizado en este proyecto son programas proporcionados por STM, como son el STMCUBEmx, STM32CubeIDE y el Atollic True Studio. En ellos se programarán los dispositivos anteriormente descritos.

También se ha utilizado el software MATLAB, para el procesamiento y análisis de los datos obtenidos en la tarjeta SD, ya que como se puede observar en la figura 4.4, los datos necesitan mucho tratamiento una vez conseguidos.

Los dos protocolos utilizados para la comunicación son, I2C y SPI, de los cuales se ha explicado su base teórica en el punto anterior.

4.4- ENSAMBLAJE

Una vez conocidos los materiales utilizados por separado, ahora hay que montar y conectar todo para poder empezar a realizar experimentos. En la figura 4.5 se puede apreciar un básico dibujo de cómo se realizarán las conexiones.

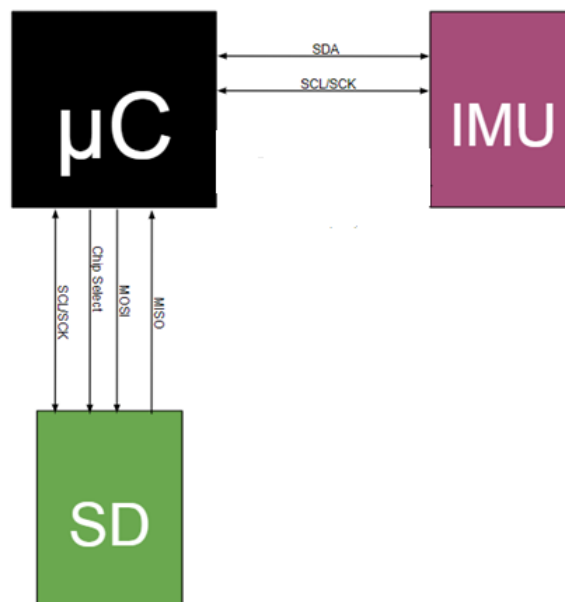


Figura 4.5.- Dibujo que engloba todas las conexiones a realizar en el montaje.

En la imagen se puede apreciar como la tarjeta SD y la unidad de medida inercial se programarán por separado. Esto quiere decir que, aunque estén en el mismo proyecto de

programa, la forma de comunicación es completamente diferente, como vimos anteriormente.

Aunque el material principal del proyecto es la IMU, de cara al ensamblaje la placa base es el objeto más importante, puesto que es la que tiene integrada el microcontrolador y a la que hay que conectar todo.

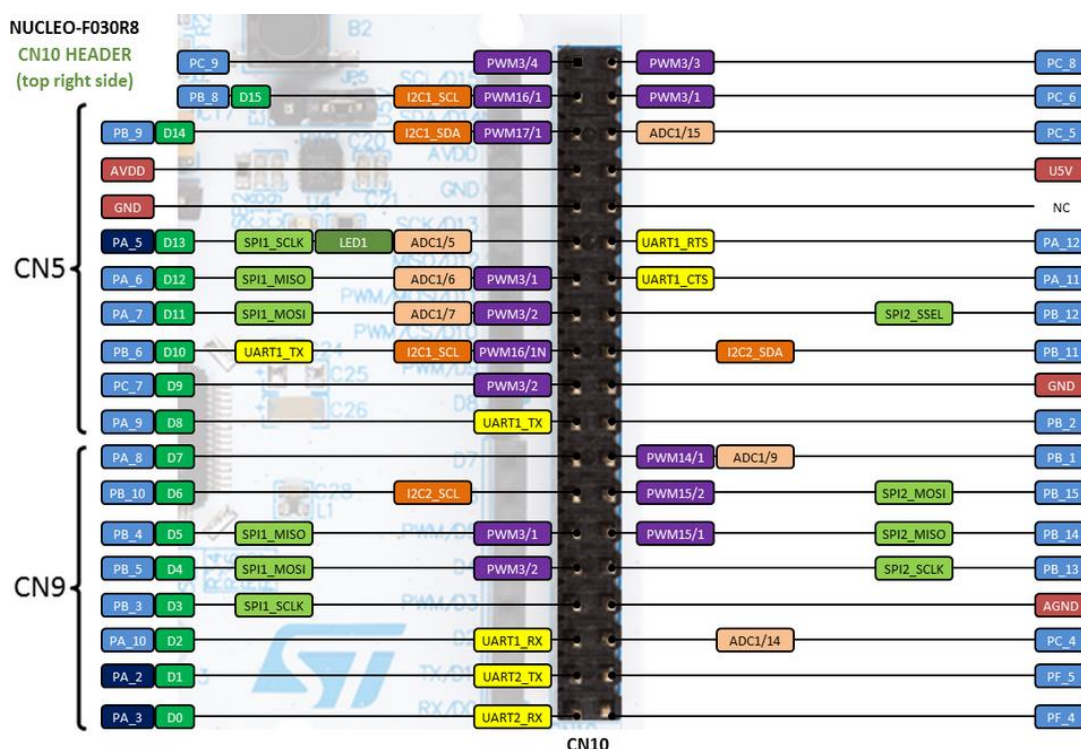


Figura 4.6.- Pinout del Núcleo F030R8

Para llevar a cabo los experimentos, el sensor LSM6DS3 se debe conectar a la placa base utilizando la interfaz I2C, por tanto se deberán conectar los pines correctamente teniendo estos datos en cuenta.

Por su parte, la tarjeta SD se conecta para tener una comunicación SPI.

Las conexiones entre la IMU y la placa base son las siguientes:

- Vcc de la IMU conectado a 3.3V de la placa base.
- GND de la IMU conectado a GND de la placa base.



- SCL de la IMU conectado a SCL (PB8) de la placa base.
- SDA de la IMU conectado a SDA (PB9) de la placa base.

Las conexiones entre la tarjeta SD y la placa base son las siguientes:

- Vcc de la IMU conectado a 3.3V de la placa base.
- GND de la IMU conectado a GND de la placa base.
- SCK del lector de tarjeta SD al pin PB13 de la placa base (SPI2 SCLK).
- MOSI del lector de tarjeta SD al pin PB15 de la placa base (SPI2 MOSI).
- MISO del lector de tarjeta SD al pin PB14 de la placa base (SPI2 MISO).
- CS del lector de tarjeta SD al pin SPI_CS de la placa base.

En el anexo de este informe se encuentra más detalladamente todas las conexiones y configuraciones necesarias en los programas para poner en funcionamiento todo el sistema

5. Resultados

Una vez programados tanto el sensor como la tarjeta SD, ya se pueden hacer experimentos, cuyos resultados se explican en este capítulo. La visualización de los resultados será procesada en el software MATLAB. Para ello, se importan los archivos de texto como archivos de datos para guardarlos en una matriz y procesarlos.

Los datos recogidos por la tarjeta SD son los datos de los registros de los sensores en crudo, es decir, es necesario transformarlos a unidades de medición. Para ello la hoja de características del sensor nos proporciona diferentes constantes por las que multiplicar los datos obtenidos del sensor para transformarlos. Se utilizarán para multiplicarlos por cada valor de la matriz en MATLAB. Tras estos pasos iniciales, se grafican los datos de cada eje de cada sensor para tener una visualización mucho más directa y simple de los movimientos del sensor.

Se ejecutarán programas para probar el funcionamiento del sensor en distintas situaciones, desde las más básicas como estar quieto en una dirección hasta moverlo aleatoriamente. También se harán experimentos conjuntos con un sensor comercial contrastado para comparar su funcionamiento.

5.1- MEDICIONES EN ESTÁTICO

Estas mediciones en estático son las primeras en realizarse, para comprobar que todas las configuraciones que hemos realizado son correctas y los resultados que obtenemos de la IMU son buenos. También nos puede decir como de precisa es la zona de trabajo en la que se han realizado los experimentos.

La teoría nos dice que cuando un sensor está en reposo (horizontal), el acelerómetro triaxial debe dar como resultado cero, o valores cercanos a cero, en dos de sus ejes, salvo en el eje en donde la aceleración de la gravedad está actuando. En este eje, los valores obtenidos



deben ser iguales o muy parecidos a dicho valor de la aceleración de la gravedad, es decir, $9,81 \text{ m/s}^2$.

Se colocará el sensor en estático sobre cada uno de los ejes, para comprobar que éstos nos den los valores esperados.

Este paso también es importante para saber los errores que tienen dichas mediciones y para poder obtener datos suficientes y en un futuro poder calibrar el sensor, ya que el error sería la diferencia entre lo obtenido y los valores teóricos esperados.

Así pues, el procedimiento para realizar este primer experimento es el siguiente:

- Conectar la IMU con la placa base asegurándose de que la tarjeta SD este conectada, vacía y lista para poder almacenar datos
- Colocar la IMU en una de las tres posiciones de su ejes.
- Comprobar que las especificaciones de frecuencia y amplitud son las correctas, detalladas anteriormente.
- Arrancar el sistema.
- Esperar unos segundos con la IMU lo más quieta posible.
- Recoger los datos obtenidos por la SD y tratarlos.
- Repetir el segundo paso para obtener los datos en los tres ejes de medición.

Una vez realizados estos pasos, se procede a visualizar los resultados obtenidos.

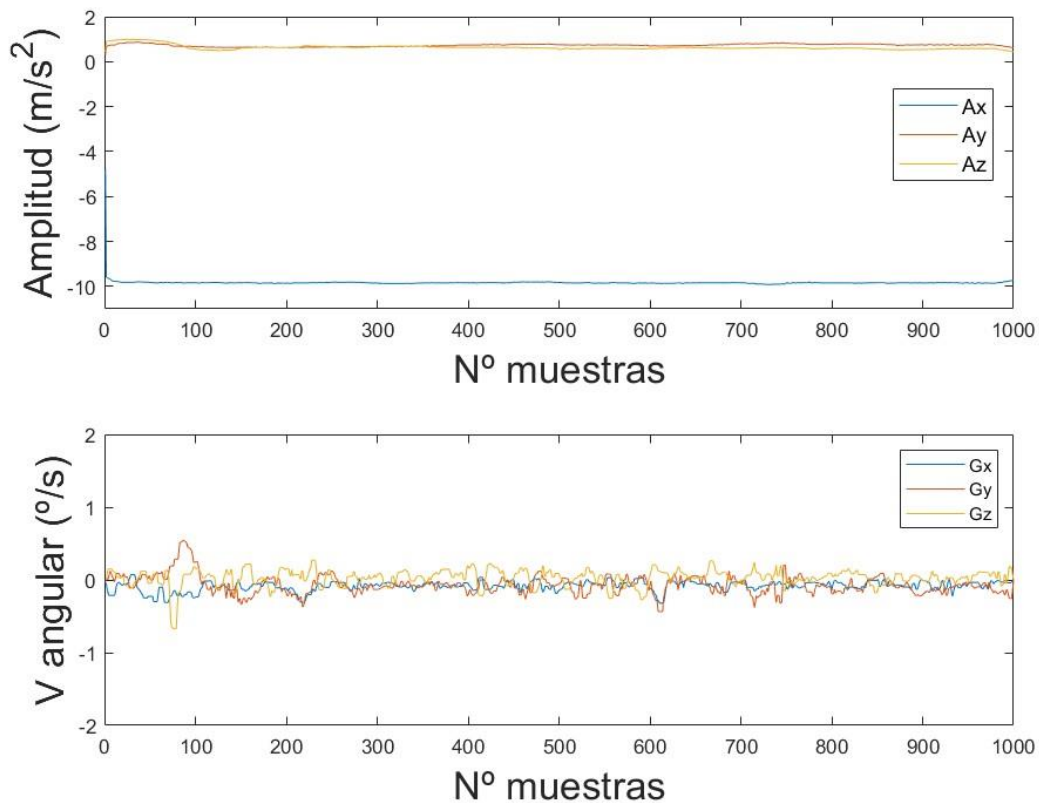


Figura 5.1.- Resultados obtenidos del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática en una posición casi horizontal en la que el eje x es el afectado por la gravedad.

	$A_x(m/s^2)$	$A_y(m/s^2)$	$A_z(m/s^2)$	$G_x(^{\circ}/s)$	$G_y(^{\circ}/s)$	$G_z(^{\circ}/s)$
Media	-9.8347	0.7281	0.6313	-0.0848	-0.0696	0.0280
Varianza	0.0261	0.0034	0.0104	0.0046	0.0165	0.0129
Desviación estándar	0.1615	0.0582	0.1019	0.0678	0.1285	0.1135

Tabla 5.1.- Valores que interesa detallar obtenidos de las señales representadas en la figura 5.1.

Se puede apreciar que el acelerómetro nos da la gravedad en el eje x (valores cercanos a 9,81) y los otros dos ejes cercanos a cero (con un pequeño error ya comentado), mientras



que el giroscopio debería dar cero en los tres ejes ya que no está en movimiento. Las medidas fluctúan pero se puede observar que lo hacen alrededor de cero.

Es posible sacar una cuantas conclusiones de los datos obtenidos. Las desviaciones nos dan valores esperados, pequeños ya que si el sensor siempre estuvo en la misma posición no debería haber movimiento.

Que lo ejes en donde la gravedad no está actuando nos de valores diferentes a cero puede ser por varias razones. La primera clara es que no se ha calibrado el sensor dentro de la zona de trabajo. También se ha cogido como referencia de eje x (o sistema horizontal) una mesa la cual no tiene por qué estar perfectamente horizontal.

En definitiva, todos estos pequeños inconvenientes hacen que los resultados que obtengamos sean parecidos a los esperados pero no exactos.

Para los demás ejes se debería ser un comportamiento parecido. Como se puede observar en las siguientes tablas y gráficas de los dos ejes restantes, éstos se desvían un poco más aún de los teóricos. Esto se debe principalmente a que a la hora de colocar los ejes éstos ya son más complicados de encontrar, y sobre todo de hacer que el sensor este perfectamente alineado.

Este pequeño inconveniente se suma a los ya detallados anteriormente y nos hace llegar una desviación algo mayor en los datos del acelerómetro.

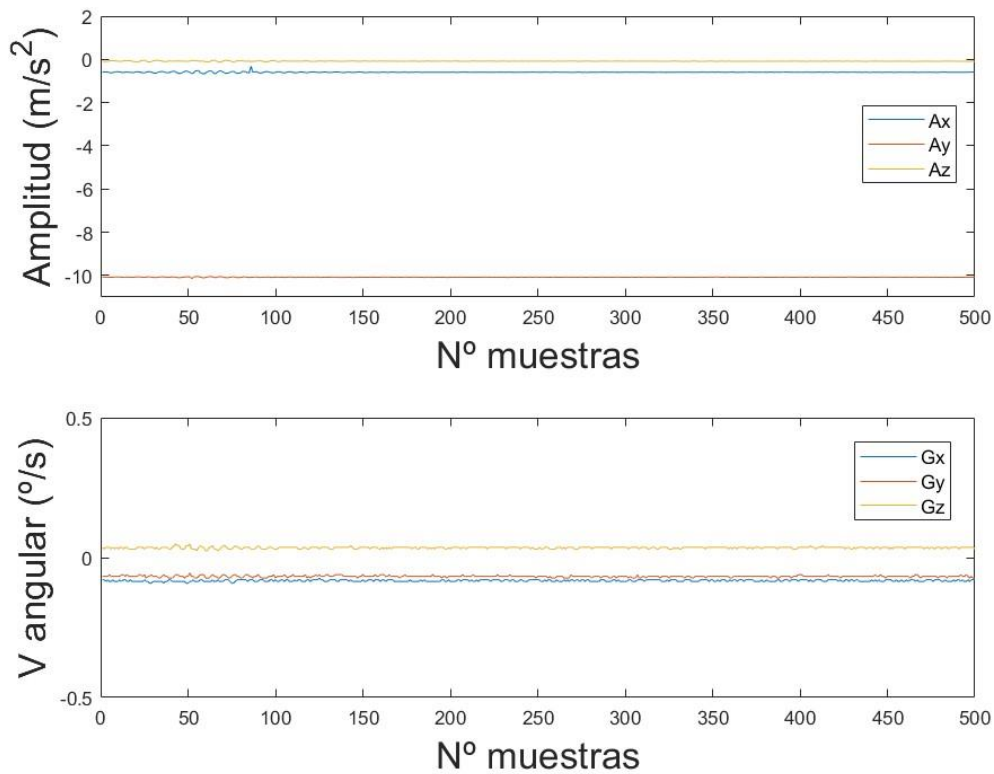


Figura 5.2.- Resultados del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática sobre la mesa en una posición en la que el eje y es el afectado por la gravedad.

	$A_x(m/s^2)$	$A_y(m/s^2)$	$A_z(m/s^2)$	$G_x(^{\circ}/s)$	$G_y(^{\circ}/s)$	$G_z(^{\circ}/s)$
Media	-0.5955	-10.0835	-0.0838	-0.0825	-0.0672	0.0351
Varianza	0.0004	0.0005	0.008	$3 \cdot 10^{-5}$	$1.2 \cdot 10^{-5}$	$4.6 \cdot 10^{-5}$
Desviación estándar	$2.3 \cdot 10^{-5}$	$6.3 \cdot 10^{-5}$	$5.8 \cdot 10^{-5}$	$2.1 \cdot 10^{-5}$	$3.1 \cdot 10^{-5}$	$1.5 \cdot 10^{-5}$

Tabla 5.2.- Valores medios de las señales representadas en la figura 5.2

En el eje y se obtiene el mismo comportamiento. Esta vez la gravedad la tiene este eje y por tanto es en el que se obtienen esos valores. Mismo comportamiento para el giroscopio, el cuál debería dar cero en los tres ejes.

Por último, el eje z:

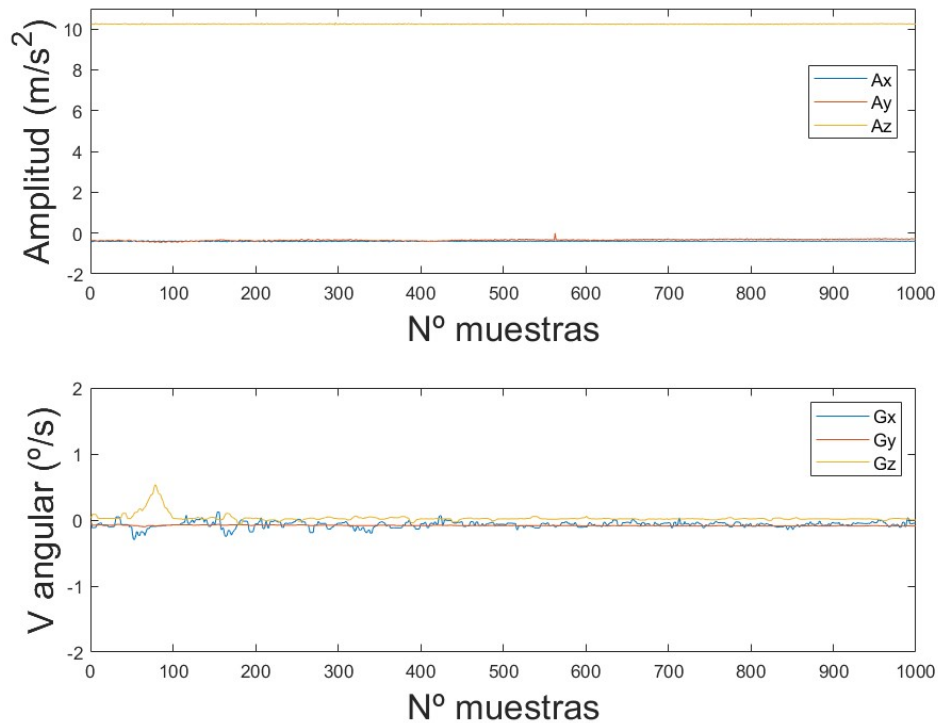


Figura 5.3.- Resultados del Acelerómetro y Giroscopio de la IMU cuándo ésta se queda estática sobre la mesa en una posición en la que el eje y es el afectado por la gravedad.

	$A_x(m/s^2)$	$A_y(m/s^2)$	$A_z(m/s^2)$	$G_x(^{\circ}/s)$	$G_y(^{\circ}/s)$	$G_z(^{\circ}/s)$
Media	-0.3981	-0.3403	10.2476	-0.0703	-0.0843	0.0301
Varianza	0.0135	0.0065	0.0142	0.0026	0.0098	0.0135
Desviación estándar	0.105	0.0987	0.0689	0.1502	0.0256	0.0658

Tabla 5.3.- Valores medios de las señales representadas en la figura 5.3

Como se ha comentado anteriormente, este trabajo quería servir como base para trabajos futuros en medicina con pacientes. Por ello, paralelamente, se ha realizado una

comprobación de comportamiento del giroscopio en dos situaciones: Una situación en la que se dejaba al sensor medir sobre una mesa y otra situación en la que la medición se realizó con el sensor cogido con la mano.

Como se puede observar en las gráficas anteriores hay una diferencia que, puede que de primera no sea muy perceptible pero cuando se calculan las varianzas y desviaciones salta a la vista con claridad y es la diferencia de datos recogidos por el giroscopio.

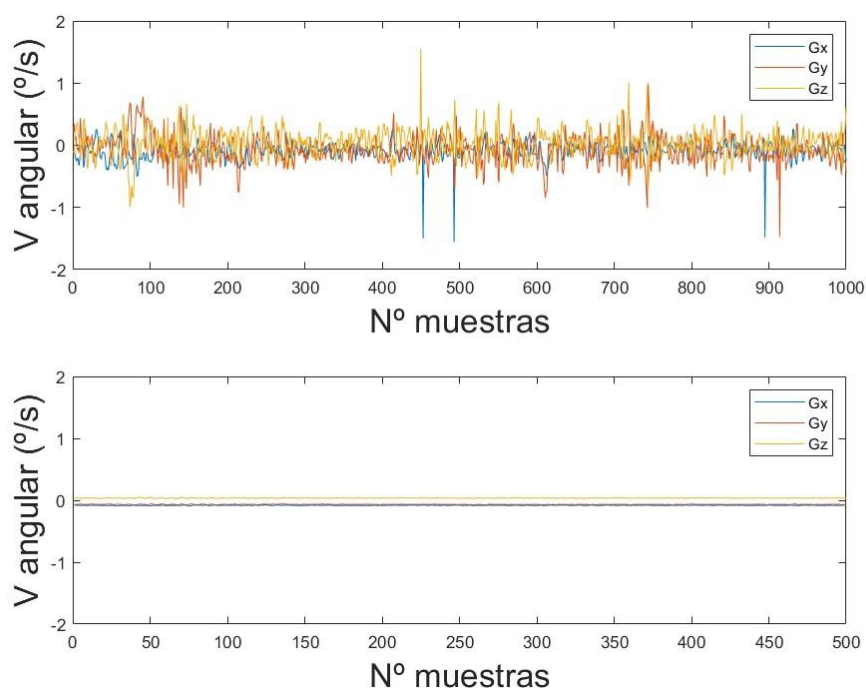


Figura 5.4.- Comparación entre los datos obtenidos por el giroscopio en 2 de los casos anteriores

En la medición en la que se encontraba fijo encima de una mesa, ésta es prácticamente inmóvil. Cuando el sensor está en contacto con una parte del cuerpo humano, en este caso una mano, debido a que una persona media no tiene un pulso preciso, este pequeño movimiento de la mano se ve reflejado en los resultados del sensor.

Este suceso puede ser muy beneficioso en proyectos futuros, ya que es aplicable en diversas situaciones medicas con pacientes.

5.2- MEDICIONES EN MOVIMIENTO

En el apartado anterior se han visto los resultados obtenidos con el sensor estático. El siguiente experimento que realizar será efectuar movimientos continuos controlados para ver el comportamiento del sensor en diferentes situaciones dinámicas.

5.2.1.- Movimiento controlado

El primer experimento que se va a realizar con el sensor en forma dinámica es realizar pequeños movimientos a lo largo de un eje. En este caso, dar golpes suaves a una caja por su lateral que se encuentra situada encima de la mesa. Con ello se pretende ver el comportamiento de la IMU en situaciones más complicadas y poder realizar comentarios sobre los datos que devuelvan el acelerómetro y giroscopio.

Antes de discutir sobre los resultados experimentales obtenidos, es prudente detenerse a considerar cual podría ser el resultado teórico esperado.

Si lo que realizamos es mover el sensor en una dirección para dar golpes suaves sobre un objeto, cabe esperar que el resultado que se obtiene será uno en el que el acelerómetro triaxial nos devuelva valores estables en dos ejes y ciertas curvas en el eje en el que se está realizando el movimiento. Por su parte, el giroscopio debería devolvernos valores distintos de cero ahora ya que se están realizando movimientos.

Una vez realizada una breve reflexión sobre el resultado que cabe esperar, se procede a realizar el experimento. El procedimiento para poder realizar el experimento es el siguiente:

- Conectar la IMU con la placa base asegurándose de que la tarjeta SD este conectada, vacía y lista para poder almacenar datos
- Comprobar que las especificaciones de frecuencia y amplitud son las correctas, detalladas anteriormente.

- Coger la IMU con una mano en el aire, más o menos a la altura de la mitad de la caja y mantenerse en dicha posición
- Arrancar el sistema.
- Mover la IMU hacia la caja repetidamente dándole 12 golpes
- Recoger los datos obtenidos por la SD y tratarlos.

Tras realizar estos pasos, se detallan en las siguientes figuras los resultados obtenidos y unas tablas de apoyo en las que visualizar mejor valores claves.

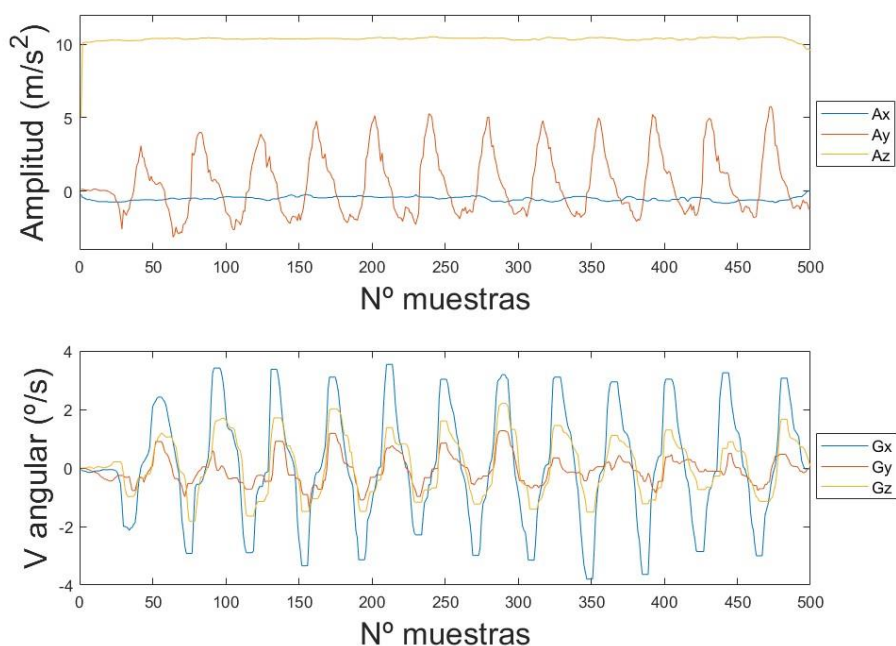


Figura 5.5.- Resultado del acelerómetro y giroscopio en el experimento donde se daban golpes a una caja situada encima de una mesa

Los resultados obtenidos de los sensores son parecidos a los discutidos anteriormente de forma teórica.

Dado que el sensor no ha sufrido ningún giro entorno a sus ejes, más allá de un movimiento a lo largo de uno de ellos, dos de los ejes del acelerómetro triaxial (x, z) no perciben el movimiento realizado como se puede apreciar en las curvas del acelerómetro. Sin embargo, en el eje y se percibe claramente movimiento.

Se perciben picos correspondientes a cada golpe realizado sobre la caja. 12 golpes, 12 picos positivos. Claramente se producen 12 picos negativos también que visualizan el movimiento contrario que hay que realizar siempre para volver a la posición inicial.

En cuanto al giroscopio, el resultado es una gráfica muy parecida a la obtenida en el eje y del acelerómetro pero en este caso en los tres ejes, los mismos 12 picos a la vez que se dan en el acelerómetro. Esto último no es del todo cierto, si se realiza un *zoom* para solo visualizar uno o dos picos podemos ver un comportamiento en el giroscopio algo diferente.

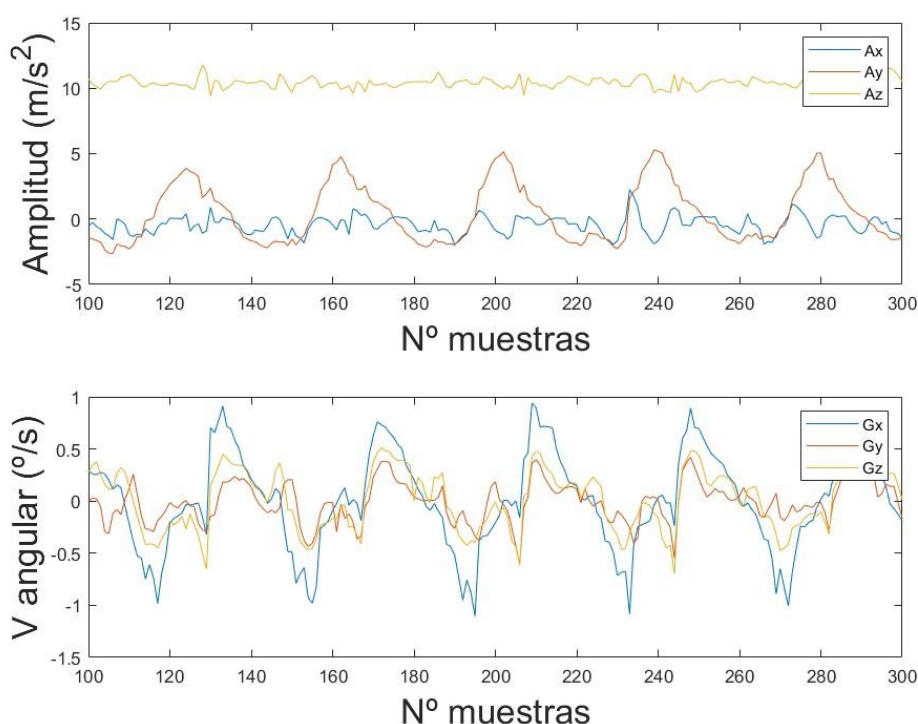


Figura 5.6.- Gráfica de la figura 5.5 ampliada para ver con mayor claridad el suceso.

Como se puede comprobar en la figura 5.6, los picos obtenidos en el acelerómetro y en el giroscopio no coinciden exactamente en la misma medida. Tienen un pequeño desfase. Esto no es un comportamiento que cabía esperar teóricamente, con lo cual hay que buscarle razones por las cuales se da.

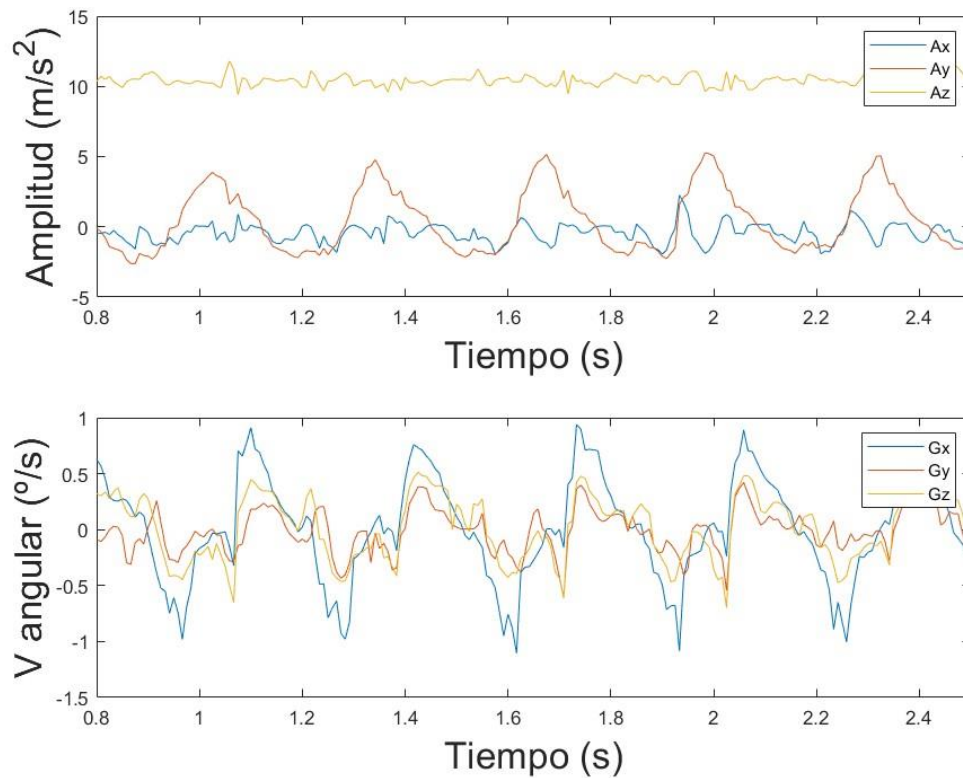


Figura 5.7.- Representación de la gráfica de la figura 5.6, pero ahora en el tiempo, para poder medir el desfase entre ambos en segundos.

Si se grafica el suceso sobre el tiempo en vez de sobre el número de muestras se puede apreciar con exactitud el tiempo de desfase que existe entre los picos del acelerómetro y los picos del giroscopio

	Pico 1 (s)	Pico 2 (s)	Pico 3 (s)	Pico 4 (s)
Acelerómetro	1.032	1.35	1.675	1.992
Giroscopio	1.1	1.41	1.733	2.058
Diferencia	0.068	0.06	0.058	0.066

Tabla 5.4.- Valores de tiempo de los picos de la gráfica 5.7 para poder realizar una comparación



La diferencia entre la medida del acelerómetro y la medida del giroscopio es del orden de las centésimas de segundo (aproximadamente 0.06 segundos). No es una diferencia muy grande cuando se hacen pruebas de larga duración pero para los experimentos que se están realizando, que tienen una duración de aproximadamente 5 segundos es una diferencia suficiente como para que se pueda apreciar visualmente.

Detectado este comportamiento, ahora habría que centrarse en cuáles son las principales causas por las que se da.

La primera razón es que, aunque la IMU sea un sistema único, los acelerómetros y giroscopios que la componen son sensores diferentes. Esto hace que tengan sensibilidades y rangos de mediciones diferentes a parte de tiempos de respuesta algo diferentes también. Si vamos más allá, como ya se explicó con anterioridad, cada eje dispone de un acelerómetro y giroscopio por separado, lo que explicarían las pequeñas diferencias de desviación de tiempo también existentes entre los tres ejes.

Otra razón por la cual se puede dar este suceso es una ampliación de lo anterior. Los acelerómetros y giroscopios son elementos por separado y puede que o estén perfectamente alineados y con buena disposición física dentro de la IMU. Pueden tener ángulos diferentes.

Por último, otra posible razón sería que los golpes suaves a la caja pueden tener componentes tanto lineales como rotacionales. Esto puede hacer que los picos en los giroscopios y acelerómetros no estén alineados debido a las diferencias en cómo cada sensor detecta y responde a dichas componentes lineales y rotacionales.

5.2.2.- Movimiento aleatorio

Se han realizado experimentos en los que el sensor se encuentra parado y realizando movimientos controlados. Ahora se va a realizar un movimiento aleatorio, en este caso giros en el aire. Con este experimento se busca que todos los ejes, tanto del acelerómetro como del giroscopio actúen y se vean modificados.

Al ser un movimiento más difícil y menos previsible en este caso es más difícil de anticipar cual va a ser el comportamiento y los datos teóricos que se obtendrán. Al hacer movimientos en círculo en todas las direcciones se verán cambios en la aceleración de los tres ejes de positiva a negativa, y una reacción a esos cambios en el giroscopio.

El procedimiento para poder realizar el experimento es el siguiente:

- Conectar la IMU con la placa base asegurándose de que la tarjeta SD este conectada, vacía y lista para poder almacenar datos
- Comprobar que las especificaciones de frecuencia y amplitud son las correctas, detalladas anteriormente.
- Coger todo el sistema con la mano.
- Arrancar el sistema.
- Mover el sistema en una especie de círculos, aumentando la altura a cada paso.
- Recoger los datos obtenidos por la SD y tratarlos.

Tras realizar estos pasos, se detallan en las siguientes figuras los resultados obtenidos.

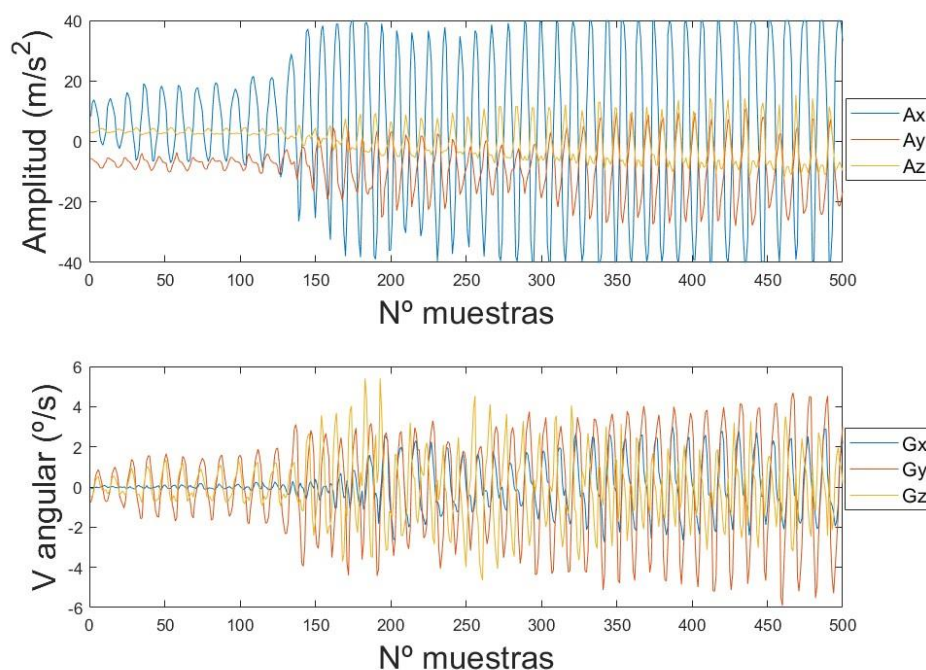


Figura 5.8.- Resultados obtenidos de realizar el movimiento aleatorio.

El resultado del experimento es el siguiente. Debido a que ahora los 3 ejes son importantes y contienen información, son difíciles de visualizar en una gráfica con la tipología con la que se habían expuesto todas las gráficas anteriores en el proyecto. Hay que separar los ejes en ventanas diferentes para poder representarlo mejor.

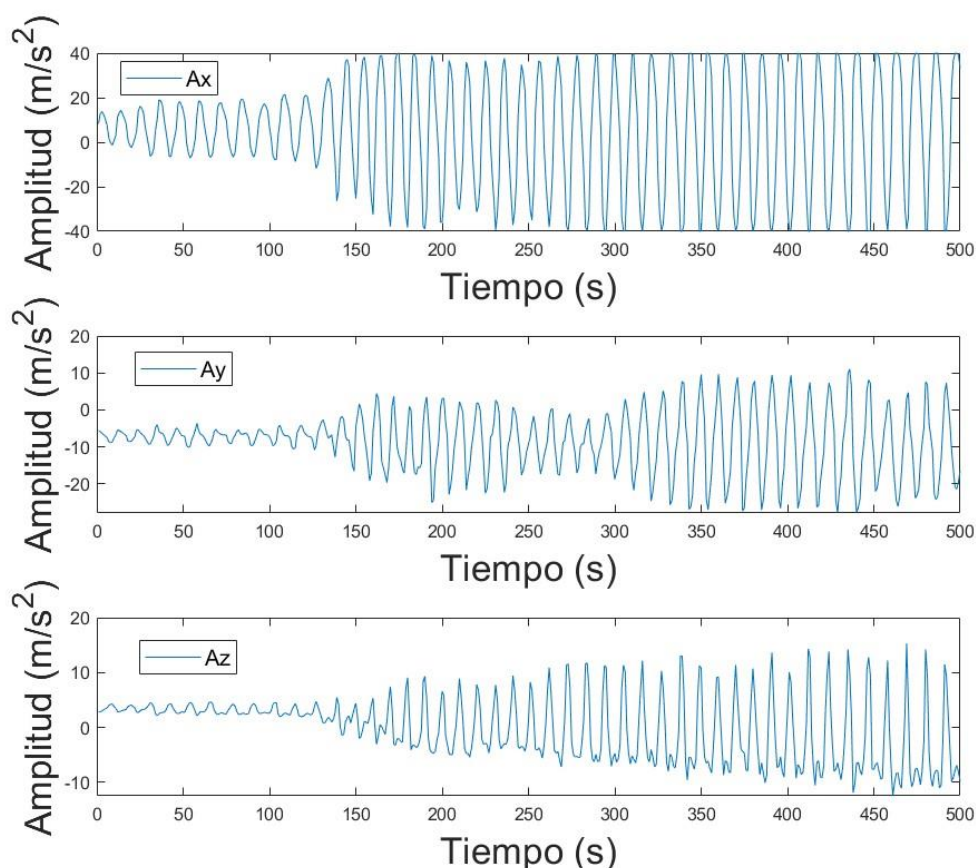


Figura 5.9.- Representación de los tres ejes del acelerómetro por separado para su correcta visualización.

Se pueden apreciar varias cosas interesantes visualizando los resultados. El comportamiento de los acelerómetros, visualizado en la figura 5.9, es bastante parecido al que se obtuvo en el experimento anterior, muchos picos, menos pronunciados, es decir, más redondeados, y unos valores de aceleración mucho mayores. Esto último se debe principalmente a que los giros en el aire se hicieron con más fuerza que los golpes suaves a la caja, con lo cual se le imprimió una aceleración mayor.

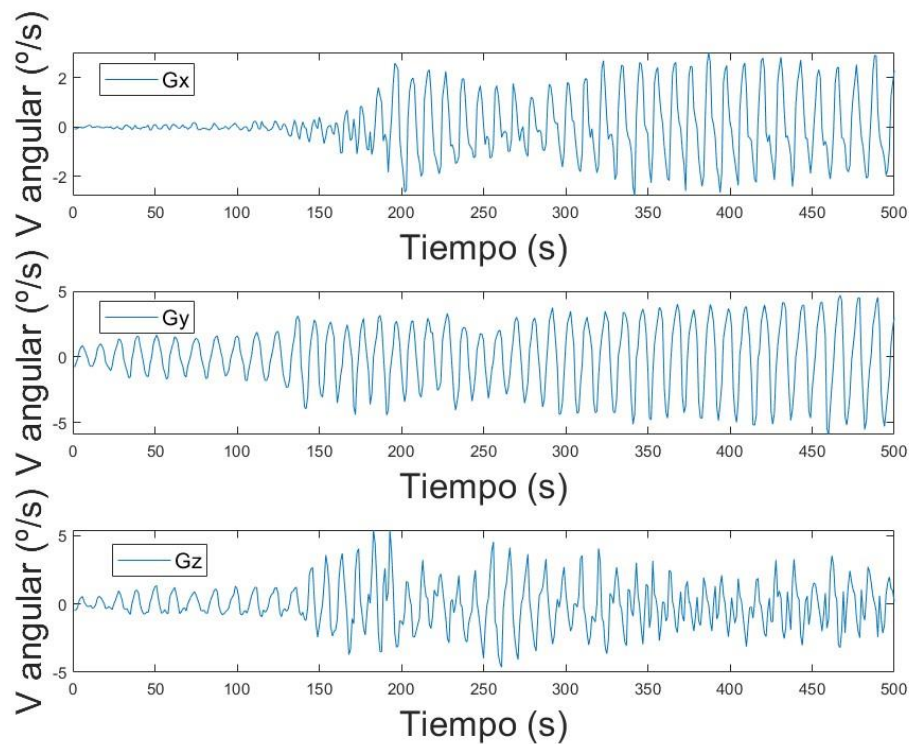


Figura 5.10.- Representación de los tres ejes del acelerómetro por separado para su correcta visualización.

En cuanto al giroscopio, visualizado en la figura 5.10 es un resultado esperado teniendo en cuenta los datos del acelerómetro. Registra cambios en la velocidad angular según se van produciendo los giros.

5.3- COMPARACIÓN CON SENSOR COMERCIAL

Una vez que se ha comprobado la capacidad del sensor de medir los distintos experimentos que se realizaron en los apartados anteriores, el siguiente paso sería comparar los resultados con un sensor comercial.

Para ello se realizarán los dos experimentos anteriores, las mediciones en estático y en movimiento.

Este siguiente experimento es perfecto para saber realmente cómo funciona nuestro sensor, ya que dará la oportunidad de poder contrastar perfectamente datos, capacidades y debilidades de éste.

Cualquier sensor comercial sería válido, pero para este trabajo se ha escogido el sensor Xsens DOT.



Figura 5.11.- Disposición de los sensores para el experimento, unidos mediante cinta adhesiva para simular que solo existe uno.

Observando la foto de la figura 5.11 se puede apreciar como los sensores están dispuestos con cinta adhesiva, y no perfectamente alineados el uno con el otro, con lo cual esa pequeña discrepancia se deberá tener en cuenta a la hora de visualizar los resultados obtenidos.

El sensor Xsens DOT funciona mediante una aplicación de móvil, con la cual se puede conectar mediante bluetooth con el sensor. Los datos que se pueden descargar de dicha aplicación móvil son datos en un archivo Excel, por tanto el proceso para poder leerlos es el mismo que con la IMU, se guardan en un archivo de texto y se procesan en MATLAB para poder graficarlos.



5.3.1.- Medición en estático

Primero se realizará un experimento para visualizar que valores obtenemos en ambos sensores cuando éstos están estáticos, al igual que se realizó en el primer apartado, pero ahora para comprobar los resultados en ambos sensores.

Así pues, el procedimiento para realizar este primer experimento es muy parecido al realizado en el apartado 5.1:

- Conectar la IMU con la placa base asegurándose de que la tarjeta SD este conectada, vacía y lista para poder almacenar datos.
- Unir mediante cinta adhesiva ambos sensores intentando conseguir la mayor alineación posible (figura 5.11).
- Colocar el bloque de sensores en una de las tres posiciones de su ejes.
- Comprobar que las especificaciones de frecuencia y amplitud son las correctas, detalladas anteriormente.
- Arrancar el sistema.
- Esperar unos segundos con la bloque de sensores lo más quieto posible.
- Recoger los datos obtenidos por la SD para la IMU y los obtenidos en formato Excel por el Xsens y tratarlos.
- Repetir el tercer paso para obtener los datos en los tres ejes de medición.

Realizando este experimento se puede tener una idea de cómo están dispuestos los sensores en el espacio. Idealmente tendríamos los ejes de los sensores perfectamente alineados y los resultados que obtendríamos serían iguales, pero como se puede observando la figura 5.11 se puede deducir que esto no va a suceder. Por ello, realizar este experimento permite cuantificar dicha diferencia para luego tenerla en cuenta en el siguiente experimento.

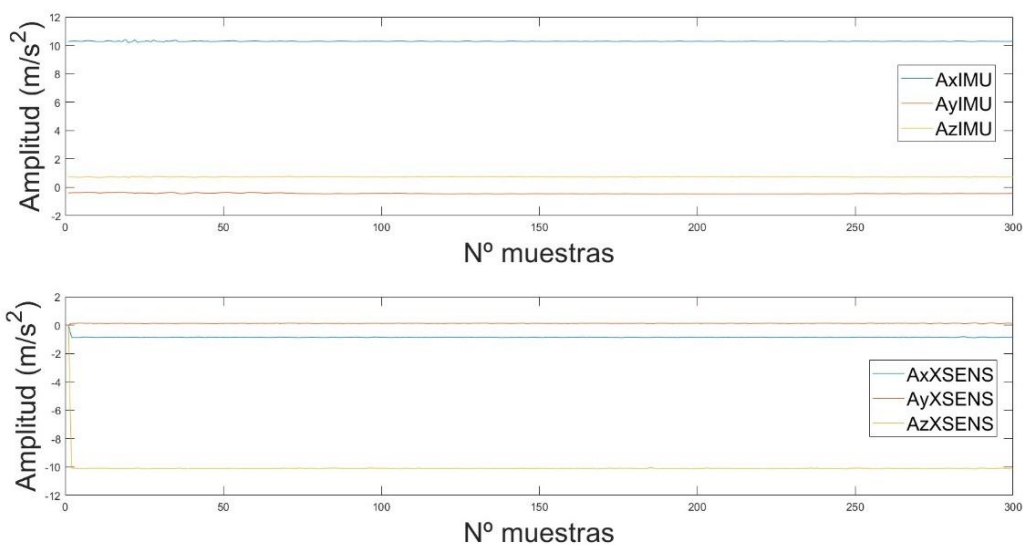


Figura 5.12.- Resultados del acelerómetro de la medición en estático con los sensores apoyados en una dirección (eje X para la IMU y eje Z para Xsens).

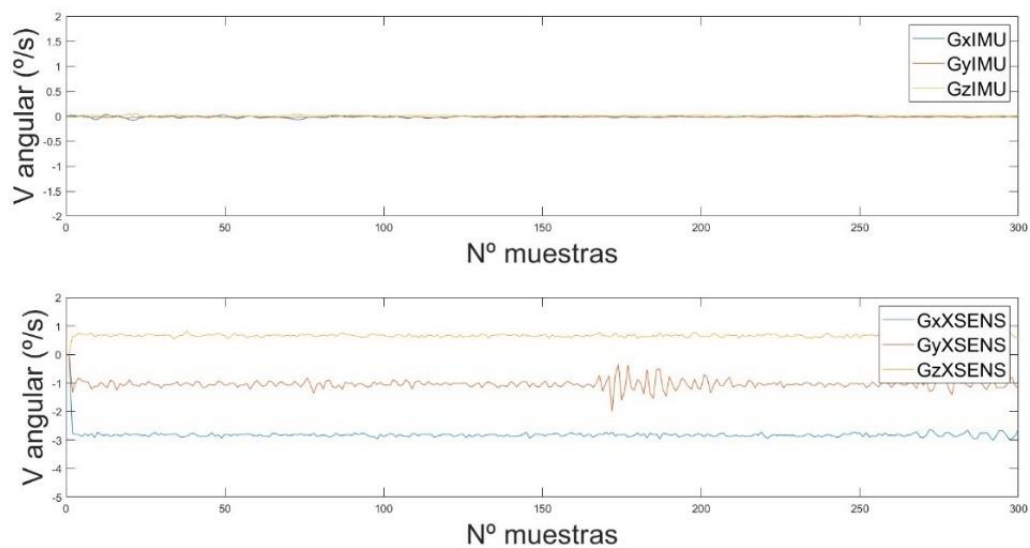


Figura 5.13.- Resultados del giroscopio de la medición en estático con los sensores apoyados en una dirección (eje X para la IMU y eje Z para Xsens)

	$A_x(m/s^2)$	$A_y(m/s^2)$	$A_z(m/s^2)$	$G_x(^{\circ}/s)$	$G_y(^{\circ}/s)$	$G_z(^{\circ}/s)$
IMU	0.7445	-0.4303	10.2897	-0.0167	-0.0147	0.0055
Xsens	-9.7697	0.0760	0.2237	-2.989	-1.236	0.876

Tabla 5.5.- Valores medios de las señales obtenidas en la medición en estático con los sensores apoyados en una dirección (eje X para la IMU y eje Z para Xsens)

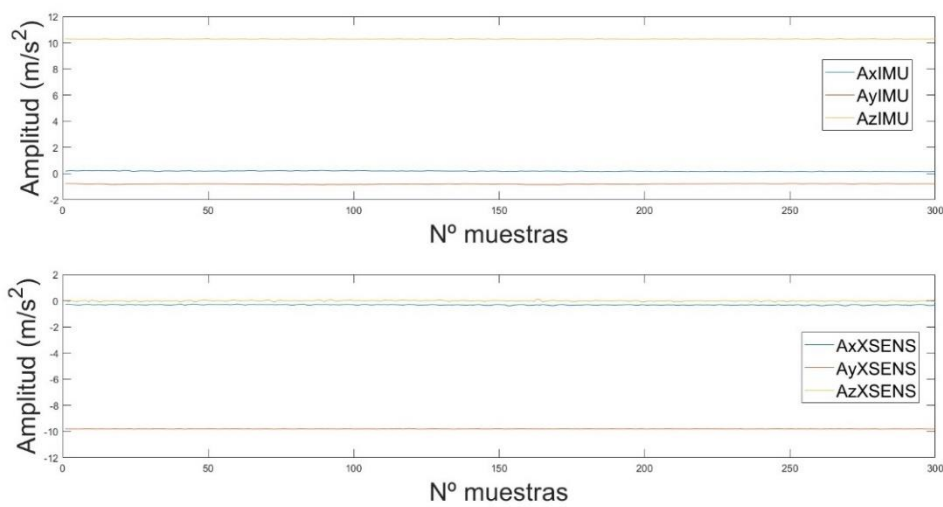


Figura 5.14.- Resultados del acelerómetro de la medición en estático con los sensores apoyados en una dirección (eje Z para la IMU y eje Y para Xsens)

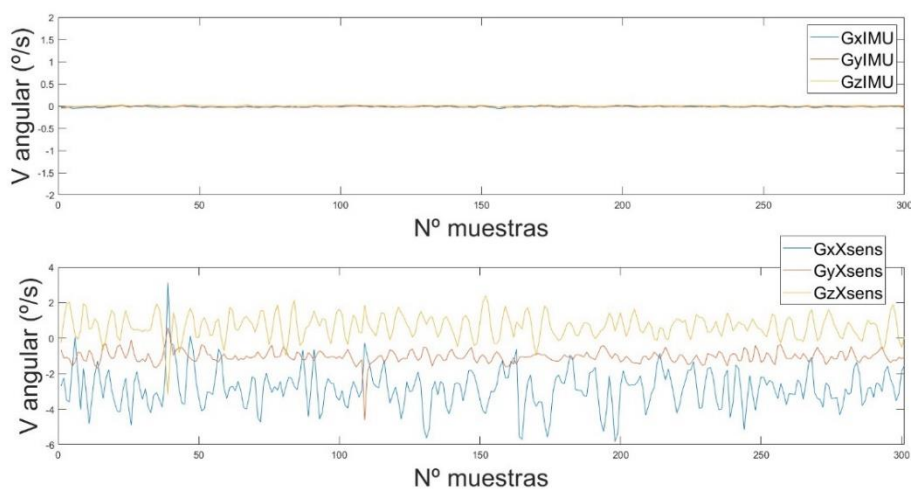


Figura 5.15.- Resultados del giroscopio de la medición en estático con los sensores apoyados en una dirección (eje Z para la IMU y eje Y para Xsens)

	$A_x(m/s^2)$	$A_y(m/s^2)$	$A_z(m/s^2)$	$G_x(^{\circ}/s)$	$G_y(^{\circ}/s)$	$G_z(^{\circ}/s)$
IMU	0.7445	-0.4303	10.2897	-0.0167	-0.0147	0.0055
Xsens	-9.7697	0.0760	0.2237	-3.142	-0.115	0.829

Tabla 5.6.- Valores medios de las señales obtenidas en la medición en estático con los sensores apoyados en una dirección (eje Z para la IMU y eje Y para Xsens)

Una vez realizado el experimento para dos ejes (no es necesario el tercero ya que se puede comprobar comportamientos perfectamente con los dos ya ilustrados) se pueden apreciar sucesos muy interesantes.

En el apartado de los acelerómetros no hay nada importante, más allá de que existe cierta desviación de los valores teóricos esperados (0 y 9.8). Sucede lo mismo que lo explicado en el primer experimento. Los acelerómetros no están perfectamente alineados y salen erros de medida. Explica también por qué no salen los mismos números en ambos sensores, no están perfectamente alineados entre sí. Viendo la figura 5.11 y el montaje casero que se tiene, es algo completamente lógico.

En el apartado de los giroscopios hay datos importantes y que dan de pensar, y es que el giroscopio del sensor Xsens se encuentra mucho más desviado que el de la IMU. Para un experimento tan simple como tener el sensor parado, los resultados no son para nada los esperados. Mientras que la teoría nos dice que si el sensor se encuentra en reposo, los valores del giroscopio deberían ser cero y constantes, en la gráfica se puede observar que, ruido aparte, si con constantes pero no son cero. Y en las dos mediciones con el sensor parado en distintas posiciones de ejes, los números medios son muy parecidos.

Entonces hay que responder a la pregunta de por qué pasa eso. Lo primero que se puede pensar es que el sensor no está bien calibrado, pero es un sensor comercial que el fabricante asegura que está calibrado. Por tanto pueden ser que pequeños errores a lo largo del tiempo, con el uso, hiciese que tuviese algo de deriva el sensor.

5.3.2.- Medición en dinámico

Una vez visto cómo se comportan los sensores en estático, y habiendo comprobado y explicado las pequeñas diferencias en las medidas, el siguiente paso es comprobar cómo es la calidad de la medida con el sensor en movimiento.

Para ello se va a realizar un movimiento dinámico físico sencillo que se pueda entender con claridad para poder comparar sensores de forma mucho más certera.

Se dispone el bloque de sensores en el aire y se realizan movimiento hacia abajo y hacia arriba consecutivamente. Con esto se quiere apreciar bien el comportamiento ante cambios sobre todo de aceleración, y tratar de explicar cuáles son los resultados que se obtienen y por qué de dichos resultados.

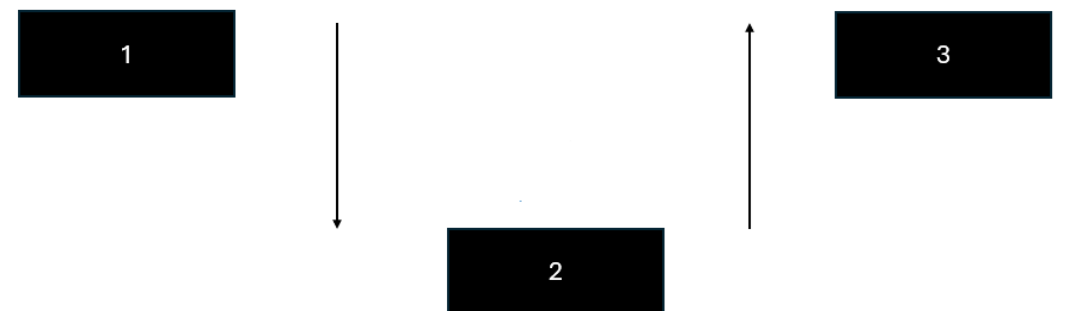


Figura 5.16.- Dibujo básico para explicar el movimiento realizado

En la figura 5.16 se puede apreciar mejor el movimiento realizado y las posiciones del sensor en todo momento. Solo se han dibujado 3 posiciones dado que estos son los que se repiten a lo largo del experimento.

Teóricamente, saber la gráfica que va a salir de este movimiento no es muy complicado. En el movimiento entre la posición 1 y la posición 2 de la figura 5.16, lo que se está realizando es una aceleración sobre un eje, lo que se vería representado de la siguiente manera:

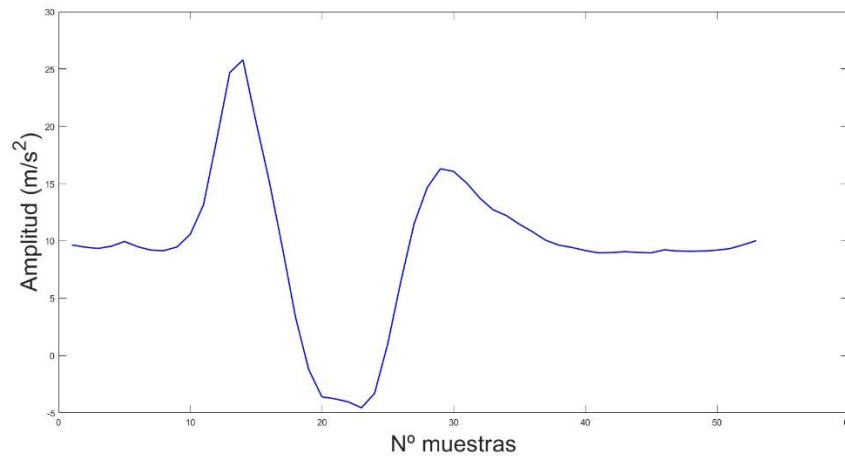


Figura 5.17.- Comportamiento esperado de un sensor cuando, partiendo del reposo se acelera hacia abajo y luego se frena.

El sensor sufre una aceleración, en este caso positiva, tras la cual se obtiene una deceleración para volver otra vez a un estado de reposo.

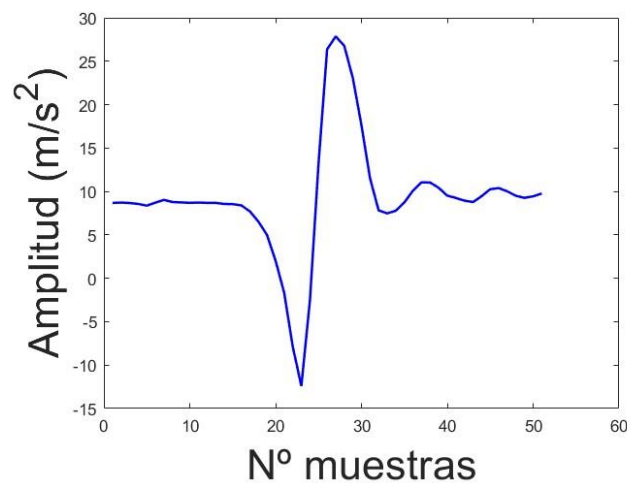


Figura 5.18.- Comportamiento esperado de un sensor cuando, partiendo del reposo se acelera hacia arriba y luego se frena.

Para el movimiento de la posición 2 a la 3 se obtiene como resultado esperado la misma gráfica con valores opuestos, primero la aceleración es negativa y la deceleración que le sigue, positiva.

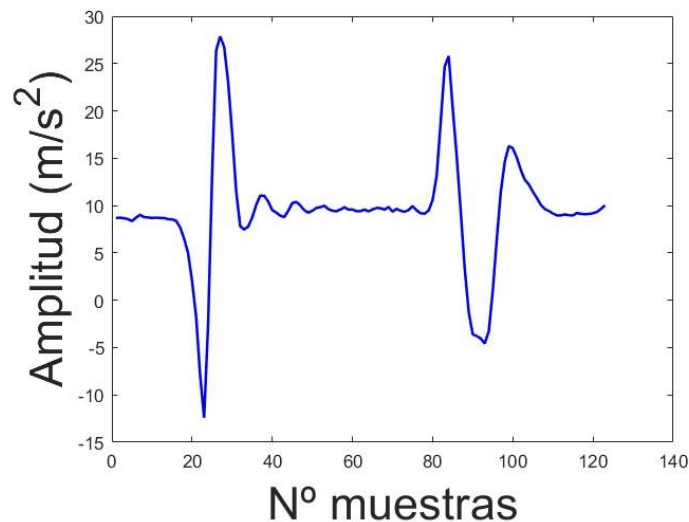


Figura 5.19.- Resultado del proceso continuo

En consecuencia, si se realiza todo el movimiento de seguido se obtendría una gráfica como la de la figura 5.19.

Ahora que se dispone de la teoría se puede realizar el experimento. Los pasos para ello son los siguientes:

- Conectar la IMU con la placa base asegurándose de que la tarjeta SD esté conectada, vacía y lista para poder almacenar datos.
- Unir mediante cinta adhesiva ambos sensores intentando conseguir la mayor alineación posible (figura 5.6).
- Comprobar que las especificaciones de frecuencia y amplitud son las correctas, detalladas anteriormente.
- Arrancar el sistema.
- Realizar el movimiento explicado anteriormente durante un periodo de tiempo.
- Recoger los datos obtenidos por la SD para la IMU y los obtenidos en formato Excel por el Xsens y tratarlos.

Durante toda la duración de este informe se ha ido viendo los comportamientos de todos los ejes tanto del acelerómetro como del giroscopio, y ya tenemos respuesta a como éstos se comportan.

Los primeros resultados a visualizar serán los obtenidos por ambos acelerómetros en los ejes en los que se ha realizado el movimiento, desechando de momento el resto.

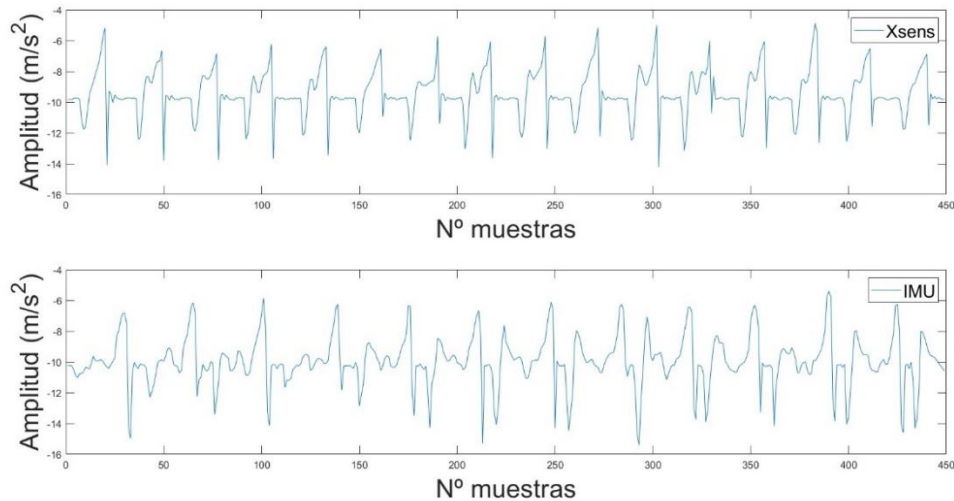


Figura 5.20.- Resultados obtenidos directamente por el acelerómetro de ambos sensores, sin tratar, a distintas frecuencias en el eje donde se produjo el movimiento.

Se puede observar como las gráficas tienen ciertas similitudes con las teóricas esperadas, aunque cuentan con pequeños cambios que se procederán a explicar.

Antes de nada, en la figura 5.20 se muestran las gráficas de ambos sensores en crudo, recién salidos los datos de los propios sensores. Así no se puede comparar nada ya ambos actúan a frecuencias diferentes. Lo mejor sería representar ambas mediciones en función del tiempo.

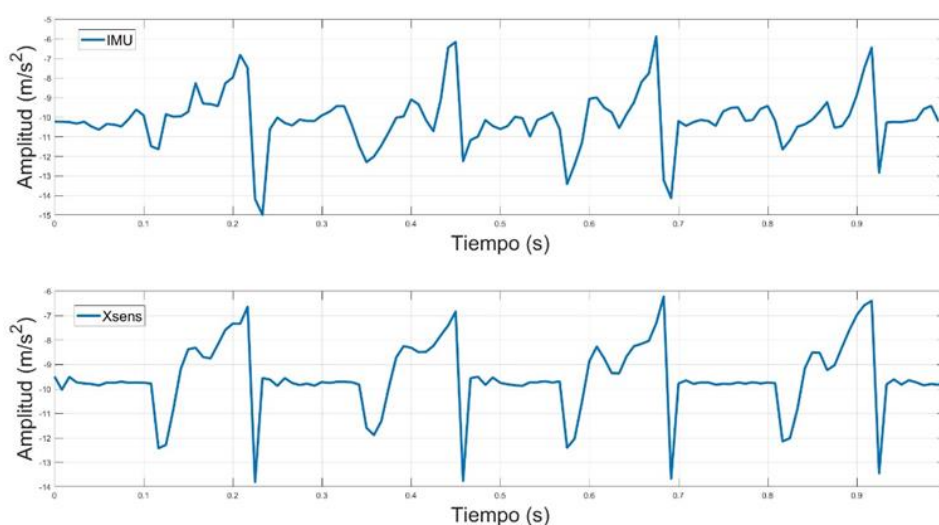


Figura 5.21.- Resultados obtenidos por el acelerómetro de ambos sensores en el eje donde se produjo el movimiento, habiendo procesado los datos para poder comparar con el tiempo.

	Máximo	Índice del máximo	Mínimo	Índice del mínimo
IMU	-5.9878	82	-14.2828	29
Xsens	-6.2303	82	-13.8956	29

Tabla 5.7.- Valores interesantes sacados de la gráfica de la figura 5.21, como son los máximos y mínimos y los índices en los que se encuentran.

Representadas ambas señales en función del tiempo se pueden visualizar mucho mejor los resultados obtenidos. En la tabla se puede observar cómo los máximos y los mínimos de cada sensor no se distancia mucho uno de otro. Las pequeñas diferencias que hay se deben a todos los problemas mencionados para el resto de los experimentos. Una mala colocación o incluso que el sensor que se tenía cogido por la mano era el Xsens y la IMU hacia el viaje un poco siguiéndole, por lo que es probable que sufra algo más de aceleración, como se puede observar con los números.



Lo que sí se puede observar con claridad, más en la gráfica del Xsens que en la gráfica de la IMU, la forma repetida en la gráfica que se asemeja a la propuesta en la teoría. Ninguna de las dos gráficas es una gráfica limpia, más bien los datos intentan asemejarse pero a veces no les da tiempo a completar el movimiento completo, debido a la rapidez con la que se realizó el proceso.

La gráfica de la IMU da datos más inestables debido un poco también a lo anterior, que el sensor que se movía era el Xsens y la IMU iba de acompañante.

5.4- DISCUSIÓN FINAL

En este proyecto se realizaron experimentos para comprobar el funcionamiento de una IMU bajo diferentes hipótesis de funcionamiento. Se realizaron pruebas en las que el sensor se mantuvo estático, el sensor se sometió a movimientos dinámicos sobre un eje y sobre todo los ejes y se comparó con un sensor comercial de referencia para comprobar que los resultados eran correctos.

Como puntos clave a favor se encuentran que el sensor obtenía valores precisos y cercanos a los esperados tanto cuando se encontraba en estático como con movimiento dinámico. Encontrado los picos que referenciaban los golpes que se le daba contra una caja perfectamente tanto en el acelerómetro como en el giroscopio. Juntándolo con un sensor comercial también se encontraron medidas esperadas, salvando algunas distancias con las medidas en alguna situación.

También se encontraron unas cuantas discrepancias y errores en las mediciones, debidas a las diferencias de calibración o, en su defecto, a la falta de ello de los sensores, teniendo incluso consideración de que los errores de *offset* contribuyeron a que valores no nulos en reposo devolvieran valor que no deberían aparecer.

Como limitaciones del proyecto está claramente el espacio de trabajo, que para el experimento de los golpes en la caja podía haber ayudado un espacio más grande para poder realizar un movimiento más amplio y la limitación que supone tener los dos sensores



pegados con cinta adhesiva y a ojo, lo que supone un error en la alineación de ambos sensores bastante amplia.

Como recomendaciones para futuros trabajos y experimentos con mayor nivel de precisión se recomienda realizar una calibración mucho más detallada de la IMU antes de los experimentos para minimizar errores y ófsets. También sería beneficioso realizar e implementar técnicas de filtrado para reducir ruidos y mejorar la precisión de las mediciones.

6. PRESUPUESTO

En este apartado se especifica el presupuesto del trabajo realizado, así como los costes futuros. Todos los dispositivos utilizados y mencionados en el documento, así como los softwares utilizados son comprados o descargados, por lo que suponen cierta cantidad de dinero que deberá ser tenida en cuenta.

Se tendrán en cuenta tres tipos de costes, los costes de mano de obra, de material y de licencias de software.

6.1.- COSTES DE MANO DE OBRA

Los costes de mano de obra vienen dados por el personal a cargo del proyecto. Cada persona tiene un puesto de trabajo y unas horas asignadas a dicho trabajo, con lo que en la siguiente tabla quedan definidos los costes.

Mano de obra	Trabajo (horas)	Coste de trabajo(€/h)	Coste total (€)
Ingeniero	150	10	1500
Supervisor	40	18	720
Total			2220

Tabla 6.1.- Costes de mano de obra

6.2.- COSTES DE MATERIAL

Los costes debidos a la utilización de material tienen en cuenta todo el material usado durante el proyecto, así como posibles reemplazos por desperfectos y futuros materiales a utilizar.

Material	Cantidad (unidades)	Coste (€/unidad)	Coste total (€)
Ordenador portátil	1	1000	1000
Placa de desarrollo	1	10.23	10.23
Módulo SD	2	7.14	14.28
Sensor IMU	4	2.81	11.24
Total			1035.75

Tabla 6.2.- Costes de material

6.3.- COSTES DEL SOFTWARE

El software utilizado para el desarrollo del proyecto necesita licencias, las cuales se pueden descargar gratis en internet mediante las páginas oficiales de la empresa, pero se incluyen en el presupuesto igualmente.

Mano de obra	Cantidad (unidades)	Coste (€/unidad)	Coste total (€)
Ingeniero	1	0	0
Supervisor	1	0	0
Total			0

Tabla 6.3.- Costes del software



6.4.- COSTES TOTALES

Por último, se juntan todos los costes anteriores en una sola tabla para poder calcular los costes totales.

Tipo	Coste total (€)
Mano de obra	2220
Material	1035.75
Software	0
Total	3255.75

Tabla 6.4.- Costes totales

7. Conclusiones

En este proyecto se ha explorado el uso de un sensor de movimiento que mide aceleración y velocidad angular, concretamente una unidad de medición inercial. Para ello era necesario contar con una placa base que dispusiese de microcontrolador y así poder programar el sensor de forma que éste trabajase bajo las configuraciones que se requerían.

Se han estudiado distintos protocolos de comunicación que había que implementar para poder conectar tanto la IMU como la tarjeta SD a la placa base escogida y que la comunicación entre ellos fuese la correcta.

Para poder realizar las mediciones requeridas en los experimentos se estudió la forma de programar en el lenguaje del software del fabricante STMicroelectronics el cual no había conocido durante carrera. Esta programación no solo consistía en configurar el sensor si no que había que configurar los protocolos de comunicación SPI, para mandar los datos a la tarjeta SD y la comunicación I2C para hacer lo propio con la IMU.

Con todo el material ensamblado y programado, se realizaron diferentes experimentos para comprobar el funcionamiento del conjunto en distintas situaciones, tanto estáticas como dinámicas y experimentos en los cuales poder comparar los resultados obtenidos con un sensor comercial.

Los resultados obtenidos se alinearon, en cierta medida con los esperados, teniendo en cuenta los errores esperados que deberán ser considerados y subsanados en trabajos futuros, como el mencionado en objetivos y alcances, de la realización de dispositivos *wearables* en el área de la medicina.



8. Bibliografía

- [1] Estándar SPI para comunicación | <https://www.puntofloante.net/COMUNICACION-SPI-TUTORIAL.htm> (Junio 2024)
- [2] T. Fin de Grado, Pablo Martínez Gallido, “UNIDAD DE MEDICIÓN INERCIAL: ESTUDIO DE FIABILIDAD Y PRECISIÓN,” 2016.
- [3] An SD card over SPI using FatFS | <https://01001000.xyz/2020-08-09-Tutorial-STM32CubeIDE-SD-card/> (Junio 2024)
- [4] “Sistemas Electrónicos Digitales UNIDAD N° 7 Registros”.
- [5] T. Fin de Grado, Francisco González Mañero, “ESTUDIO Y MODELADO DE ERRORES QUE AFECTAN A UNA UNIDAD DE MEDIDAS INCERCIALES DE BAJO COSTE,” 2016.
- [6] “Dispositivos Electrónicos Programables UNIDAD N°3 Modelos de sensores”.



ANEXO: MANUAL DE USUARIO

EXPLOTACIÓN DE UN SENSOR IMU PARA LA MONITORIZACIÓN EN TIEMPO REAL

MARCOS GONZÁLEZ CAMÍN

TUTOR: D. D. JUAN CARLOS ÁLVAREZ ÁLVAREZ

FECHA: Julio 2024



En este anexo se explicará paso a paso el proceso de conexión de los dispositivos en el software utilizado para que le sea útil a una futura persona que se encargue de continuar con el proyecto.

Primero, se empezará con la conexión de la nucleoboard, la tarjeta SD y el STM32CubeID.

- Abrir STM32CubeIDE.
- Abrir un nuevo proyecto y seleccionar el núcleo utilizado.
- Responder ‘Yes’ a ‘*Initialize all peripherals in their default configuration?*’.
- Responder ‘Yes’ a ‘*Open device configuration view?*’.

El objetivo de esta parte de la guía es conectar la tarjeta SD al núcleo vía una comunicación SPI de tal manera habrá que configurar los pines correspondientes. La sincronización y la transmisión de datos se realiza por medio de 4 señales:

- SCLK (reloj).
- MOSI (Master Output Slave Input);
- MISO (Master Input Slave Output);
- SS/Select: Para seleccionar un Esclavo.

Por lo tanto vamos a usar un periférico SPI para las primeras cuatro señales y un pin GPIO para usar como selección de chip (chip select).

Ahora se configurarán los pines del SPI2. Seleccione la comunicación SPI2 a la izquierda y luego configúrelo en ‘Full Duplex Master’. Luego configure el tamaño de datos (data size) en 8 bits y el Prescaler del reloj en 128.

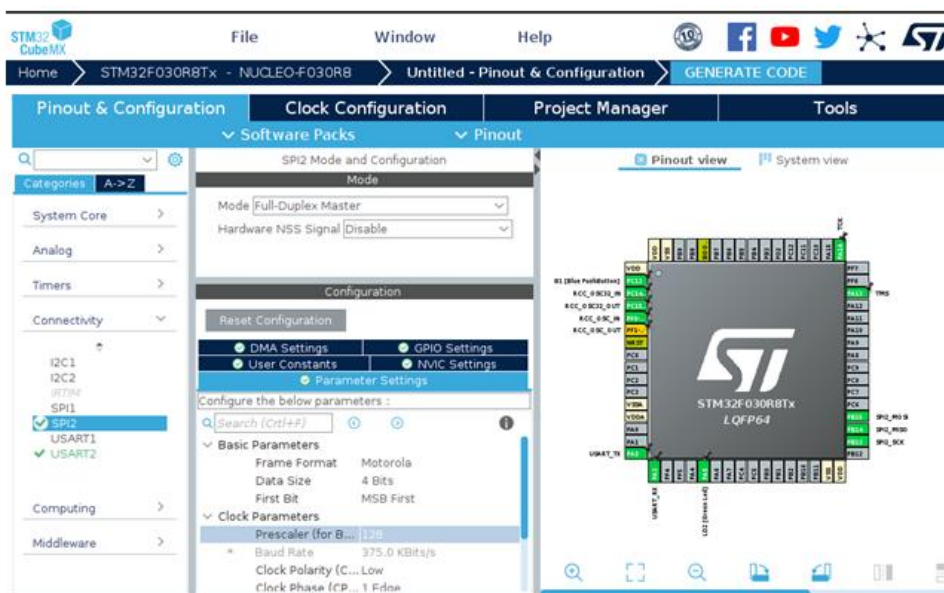


Figura 1.- Configuración SPI

Luego se selecciona el pin PB1 y se escoge el modo GPIO_Output. Y se le cambia el nombre.

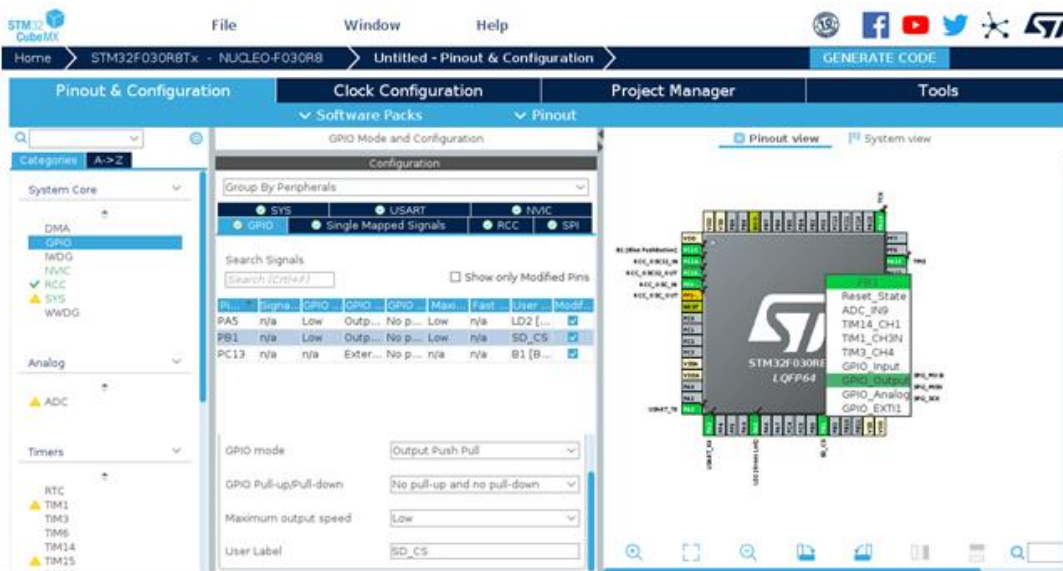


Figura 2.- GPIO

Finalmente, como se va a utilizar un driver para la tarjeta SD con el sistema de archivos FAT, hay que ir a la categoría Middleware y luego habilitar FATFS como definido por el usuario. Se puede dejar todos los demás parámetros con sus valores predeterminados.

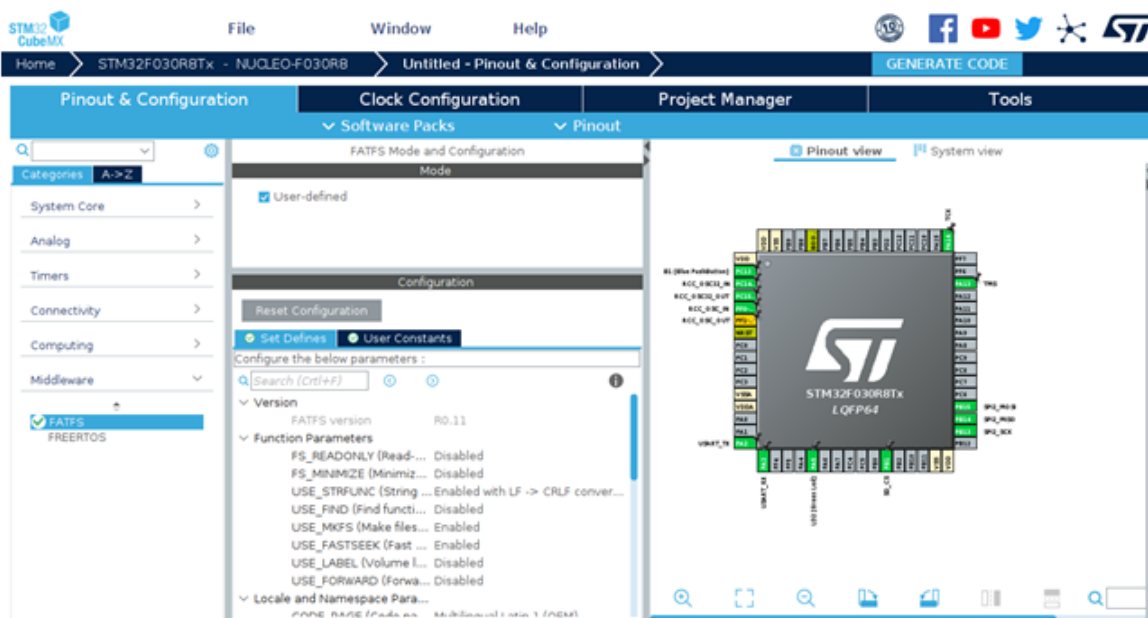


Figura 3.- FATFS

Después de haber configurado los pines, hay que escoger un nombre de proyecto en la sección 'Project Manager', seleccionando la generación de código para la plataforma TrueSTUDIO y pulsar la opción 'Generate Code'.

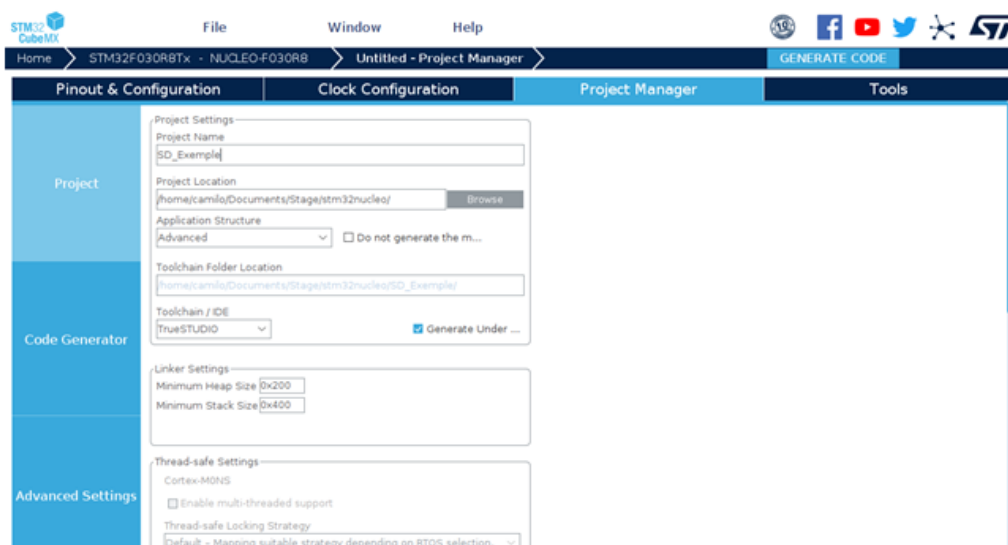


Figura 4.- Proyecto

Para conectar la tarjeta SD al microcontrolador se deben utilizar 6 pines los cuales ya se han configurado.

- SCK del lector de tarjeta SD al pin PB13 del núcleo (SPI2 SCLK).
- MOSI del lector de tarjeta SD al pin PB15 del núcleo (SPI2 MOSI).
- MISO del lector de tarjeta SD al pin PB14 del núcleo (SPI2 MISO).
- CS del lector de tarjeta SD al pin PB1 del núcleo (Nombre que se le otorgo en el paso anterior).
- Tensión y masa a los pines correspondientes.

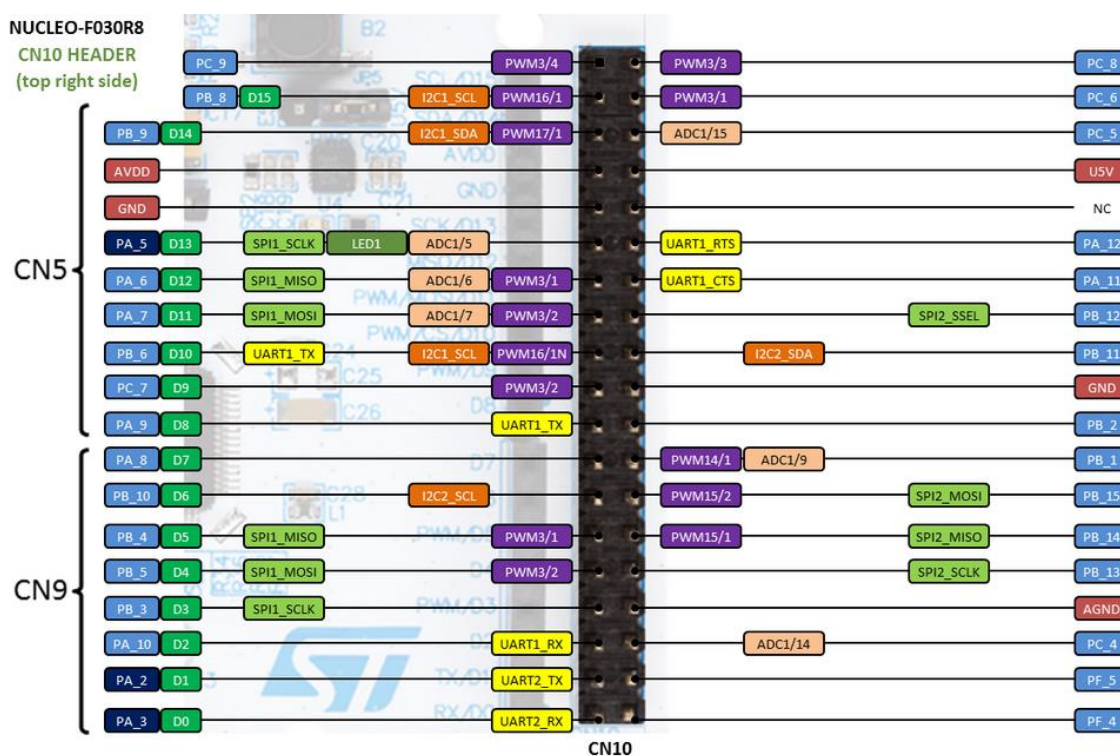


Figura 5.- Pinout Nucleo F030R8

Ahora se debe escribir el software para poder guardar datos en la memoria SD. El primer paso es copiar el driver en el proyecto. Este driver contiene dos archivos, `user_diskio_spi.c` y `user_diskio_spi.h`.

Una vez descargados, se copian en el directorio FATFS/Target y se abre TrueStudio para hacer algunas modificaciones en el software.



Primero hay que vincular el driver a FatFS. Esto implicará unos cambios en el archivo `user_diskio.c`. Las modificaciones necesarias se hacen en aquellas líneas en las que exista la anotación `//Agregar esta línea`.

```
/* USER CODE BEGIN DECL */
```

```
/* Includes -----*/
```

```
#include <string.h>
```

```
#include "ff_gen_drv.h"
```

```
#include "user_diskio_spi.h" //Agregar esta linea
```

```
/* Private functions -----*/
```

```
/**
```

```
 * @brief Initializes a Drive
```

```
 * @param pdrv: Physical drive number (0..)
```

```
 * @retval DSTATUS: Operation status
```

```
 */
```

```
DSTATUS USER_initialize (
```

```
    BYTE pdrv    /* Physical drive number to identify the drive */
```

```
)
```

```
{
```

```
    /* USER CODE BEGIN INIT */
```

```
        return USER_SPI_initialize(pdrv); //Agregar esta linea
```

```
    /* USER CODE END INIT */
```

```
}
```

```
/**
```

```
 * @brief Gets Disk Status
```

```
 * @param pdrv: Physical drive number (0..)
```

```
 * @retval DSTATUS: Operation status
```

```
 */
```

```
DSTATUS USER_status (
```

```
    BYTE pdrv    /* Physical drive number to identify the drive */
```




```
)  
{  
    /* USER CODE BEGIN STATUS */  
    return USER_SPI_status(pdrv); //Agregar esta linea  
    /* USER CODE END STATUS */  
}  
  
/**  
 * @brief Reads Sector(s)  
 * @param pdrv: Physical drive number (0..)  
 * @param *buff: Data buffer to store read data  
 * @param sector: Sector address (LBA)  
 * @param count: Number of sectors to read (1..128)  
 * @retval DRESULT: Operation result  
 */  
DRESULT USER_read (  
    BYTE pdrv, /* Physical drive number to identify the drive */  
    BYTE *buff, /* Data buffer to store read data */  
    DWORD sector, /* Sector address in LBA */  
    UINT count /* Number of sectors to read */  
)  
{  
    /* USER CODE BEGIN READ */  
    return USER_SPI_read(pdrv, buff, sector, count); //Agregar esta linea  
    /* USER CODE END READ */  
}  
  
/**  
 * @brief Writes Sector(s)  
 * @param pdrv: Physical drive number (0..)  
 * @param *buff: Data to be written  
 * @param sector: Sector address (LBA)  
 * @param count: Number of sectors to write (1..128)  
 * @retval DRESULT: Operation result
```



```
*/
#if _USE_WRITE == 1
DRESULT USER_write (
    BYTE pdrv,    /* Physical drive number to identify the drive */
    const BYTE *buff, /* Data to be written */
    DWORD sector, /* Sector address in LBA */
    UINT count    /* Number of sectors to write */
)
{
    /* USER CODE BEGIN WRITE */
    /* USER CODE HERE */
    return USER_SPI_write(pdrv, buff, sector, count); //Agregar esta linea
    /* USER CODE END WRITE */
}
#endif /* _USE_WRITE == 1 */

/**
 * @brief I/O control operation
 * @param pdrv: Physical drive number (0..)
 * @param cmd: Control code
 * @param *buff: Buffer to send/receive control data
 * @retval DRESULT: Operation result
 */
#if _USE_IOCTL == 1
DRESULT USER_ioctl (
    BYTE pdrv,    /* Physical drive number (0..) */
    BYTE cmd,     /* Control code */
    void *buff    /* Buffer to send/receive control data */
)
{
    /* USER CODE BEGIN IOCTL */
    return USER_SPI_ioctl(pdrv, cmd, buff); //Agregar esta linea
    /* USER CODE END IOCTL */
}
}
```



```
#endif /* _USE_IOCTL == 1 */
```

El último paso para que el driver esté listo para funcionar es comprobar en `user_diskio_spi.c`, qué driver se está utilizando.

```
//Make sure you set #define SD_SPI_HANDLE as some hspix in main.h  
//Make sure you set #define SD_CS_GPIO_Port as some GPIO port in main.h  
//Make sure you set #define SD_CS_Pin as some GPIO pin in main.h  
extern SPI_HandleTypeDef SD_SPI_HANDLE;
```

Con lo que en el `main.h` se agrega la siguiente línea.

```
/* USER CODE BEGIN Private defines */  
#define SD_SPI_HANDLE hspi2  
/* USER CODE END Private defines */
```

Por último, hay que asegurarse que las velocidades de reloj de nuestro controlador de tarjeta SD son compatibles con las que seleccionamos anteriormente con lo que en la línea 35 del `user_diskio_spi.c`

```
//(Note that the _256 is used as a mask to clear the prescalar bits as it provides binary 111 in the  
correct position)  
  
#define FCLK_SLOW() { MODIFY_REG(SD_SPI_HANDLE.Instance->CR1,  
SPI_BAUDRATEPRESCALER_256, SPI_BAUDRATEPRESCALER_128); } /* Set SCLK = slow, approx 280  
KBits/s*/  
  
#define FCLK_FAST() { MODIFY_REG(SD_SPI_HANDLE.Instance->CR1,  
SPI_BAUDRATEPRESCALER_256, SPI_BAUDRATEPRESCALER_8); } /* Set SCLK = fast, approx 4.5  
MBits/s */
```

Segundo, se explicará la conexión del núcleo y el sensor con el programa. El objetivo de esta parte de la guía es conectar el sensor al núcleo vía una comunicación I2C de tal manera que habrá que configurar los pines correspondientes.

- Abrir STM32CubeIDE.
- Abrir un nuevo proyecto y seleccionar el núcleo utilizado.
- Responder ‘Yes’ a ‘Initialize all peripherals in their default configuration?’.
- Responder ‘Yes’ a ‘Open device configuration view?’.

La transmisión de datos en I²C se realiza por medio de 2 señales: reloj (SCL, Serial Clock) y la línea de datos (SDA, Serial Data).

En la ventana Pinout & Configuration, se asigna PB8 y PB9 a las funciones I2C1_SCL e I2C1_SDA, respectivamente.

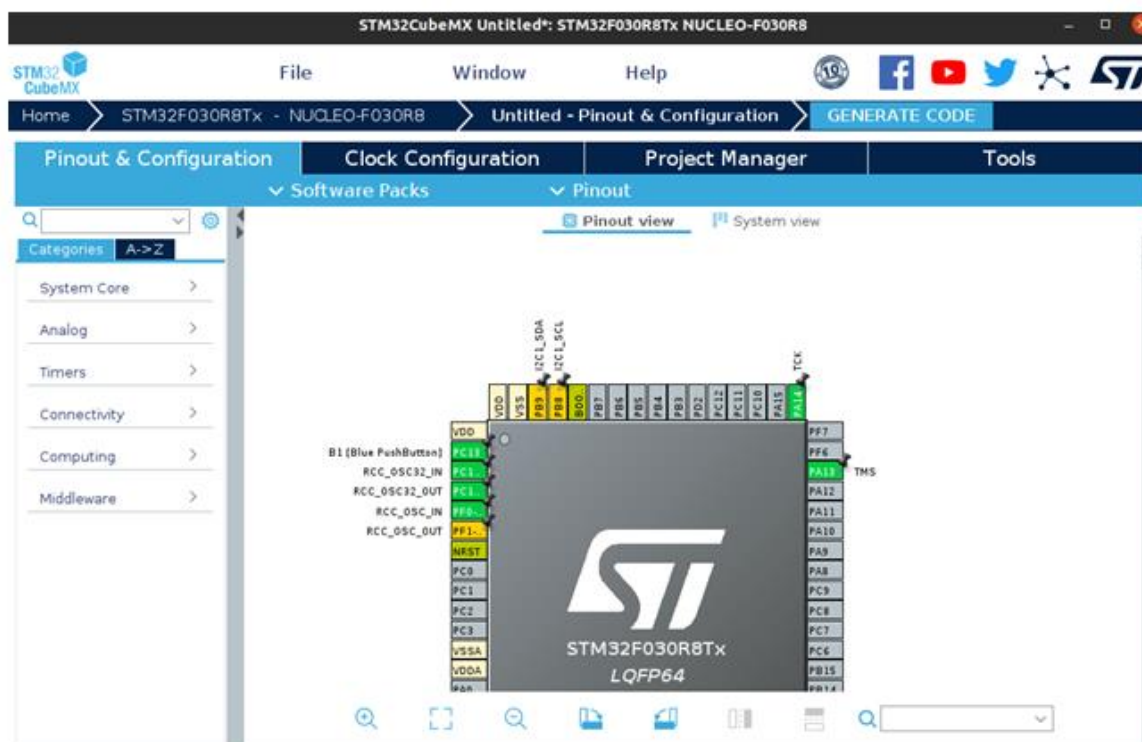


Figura 6.- Asignación pines

Los pines deben ser de color amarillo, lo que indica que aunque los pines están asignados a un periférico, ese periférico aún no se ha inicializado en el software CubeMX. En el panel izquierdo de CubeMX, se selecciona Categorías > Conectividad > I2C1. En el panel Modo, cambie Deshabilitar I2C a I2C.

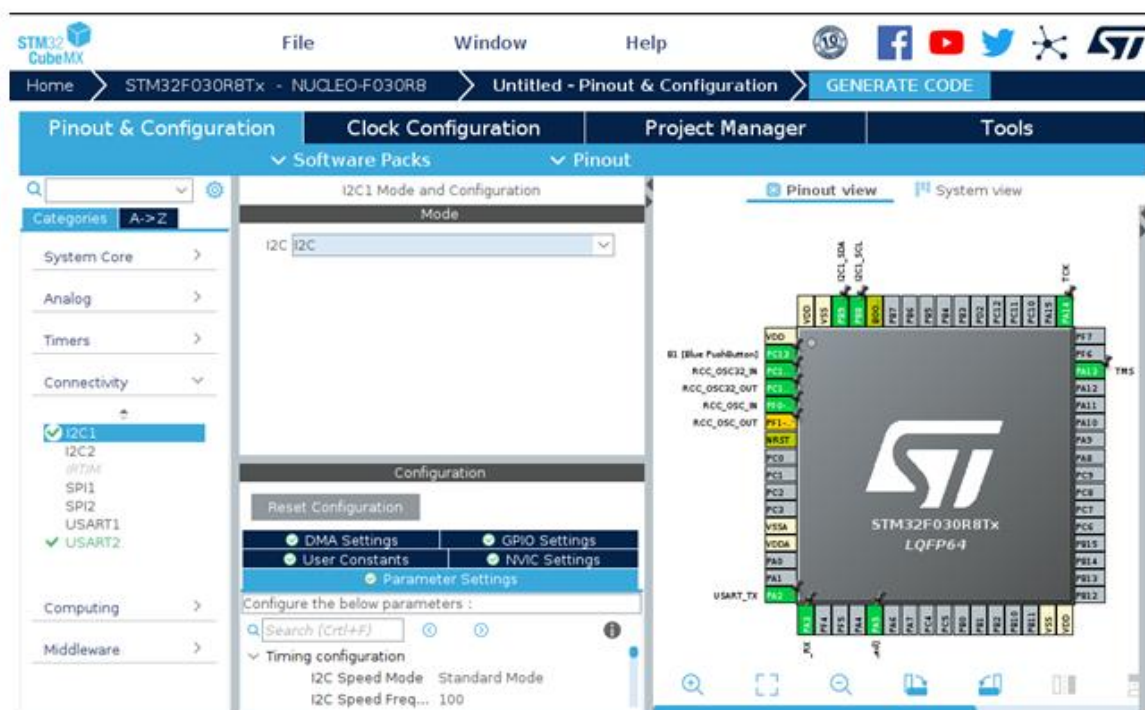


Figura 7.- Configurar I2C1

Se debería observar que los pines PB8 y PB9 se vuelven verdes, lo que indica que se ha configurado completamente el periférico I2C.

Haciendo clic en Archivo > Guardar y clic en Sí para generar código, hay que asegurarse de que al momento de generar el código, se seleccione la generación del código para la plataforma TrueSTUDIO.

Para conectar el sensor al microcontrolador se deben usar 4 pines los cuales ya se han configurado.

- SCL del sensor al pin PB8 del núcleo (I2C1_SCL).
- SDA del sensor al pin PB8 del núcleo. (I2C_SDA).
- Tensión y tierra a los pines correspondientes.

Ya tenemos el montaje realizado con las pertinentes configuraciones, ya está todo listo para empezar a programar.