

Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Ingeniería Eléctrica
IE0624 – Laboratorio de Microcontroladores
II ciclo 2024

Laboratorio # 4
STM32: GPIO, ADC, comunicaciones, IoT

Leonardo Serrano Arias C17484
Lorena Solís Extteny B97657

Profesor: Marco Villalta

30 de Octubre, 2024

Índice

Introducción	2
Nota teórica	2
STM32F429	2
Sensor L3GD20	5
Pantalla LCD/TFT ILI9341	6
GPIOs (General Purpose Input/Output)	6
ADC	6
Comunicaciones	6
USART	6
SPI	7
USB	7
IoT (Internet of Things)	7
Hardware	7
Tabla de componentes electrónicos	7
Firmware y libopencm3	8
Archivos de libopencm3	8
Desarrollo	9
Diagrama de flujo	9
Diseño del circuito con divisor de tensión	9
Funcionamiento del circuito	9
Análisis electrónico	13
Conclusiones y recomendaciones	15
Apéndices	16

Índice de figuras

1. Diagrama de pines del STM32F429 [1]	2
2. Diagrama de bloques del STM32F429 [1]	3
3. Diagrama de pines del sensor L3GD20[2]	6
4. Diagrama de flujo del Sismógrafo	9
5. STM32F429 Discovery Kit encendido	10
6. Lectura de valores de los ejes del giroscopio	11
7. Circuito para conectar la placa a una batería de 9V	11
8. USB COM activado	12
9. Valores enviados	12
10. Dashboard de Thingsboard del Sismógrafo para varios valores medidos	13
11. Valor medido a la salida del regulador	14
12. Tensión medida de la batería	14

Introducción

Este informe detalla el diseño y desarrollo de un sismógrafo digital, implementado con la placa STM32F429 Discovery Kit y la biblioteca libopencm3. El dispositivo está alimentado por una batería y se utiliza para registrar y analizar oscilaciones en el entorno. Entre las funcionalidades incluidas están la lectura de los ejes del giroscopio, el monitoreo del nivel de batería, y la habilitación de comunicación mediante USART/USB. Para facilitar el monitoreo remoto, se implementa una conexión con la plataforma IoT Thingsboard, permitiendo la visualización de las mediciones y el estado del dispositivo a distancia. El repositorio utilizado para el desarrollo de este laboratorio se encuentra en la siguiente dirección: https://github.com/Leonardo-SA/IE-0624_II_2024_C17484_B97657.git

Nota teórica

STM:32F429

El STM32F429 [1] es un microcontrolador de la serie STM32 de STMicroelectronics, caracterizado por su alta capacidad de procesamiento y múltiples interfaces de comunicación, lo que lo convierte en una opción ideal para aplicaciones avanzadas. El STM32F429 cuenta con una Unidad de Punto Flotante (FPU) y una amplia gama de periféricos, como ADC (Convertidores Analógico-Digitales) y USART (Transmisor-Receptor Asíncrono Universal). Estas características permiten realizar cálculos complejos y gestionar comunicaciones de manera eficiente, lo cual es fundamental para el funcionamiento de un sismógrafo digital que requiere un procesamiento rápido de los datos del giroscopio y un monitoreo constante del nivel de batería.

A continuación se muestra el diagrama de pines:

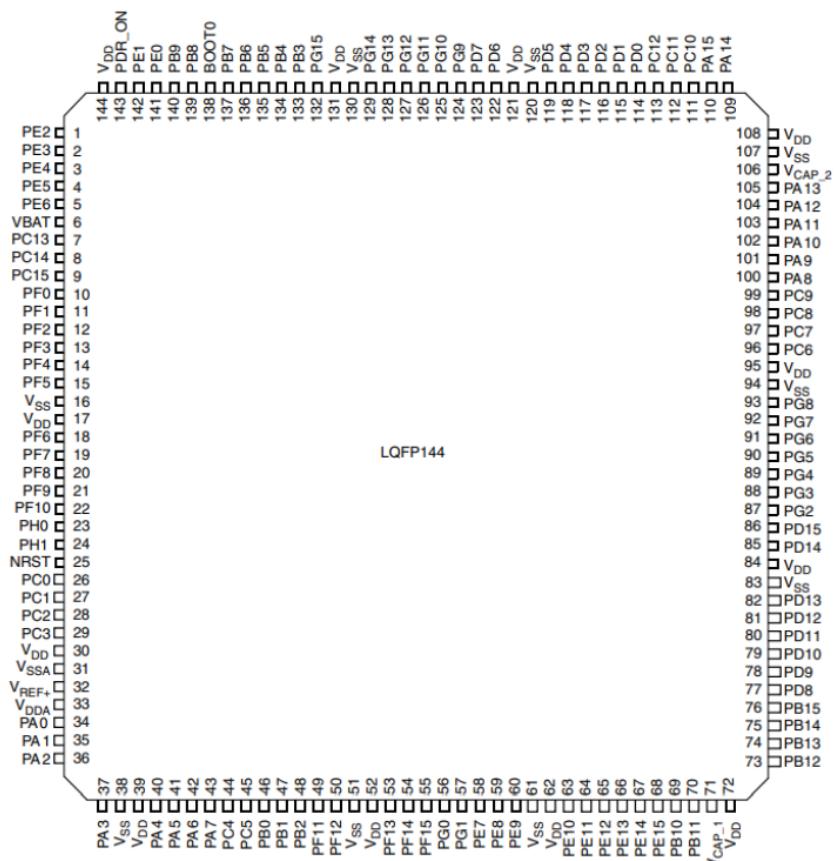


Figura 1: Diagrama de pines del STM32F429 [1]

En la siguiente figura es posible apreciar el diagrama de bloques de este microcontrolador:

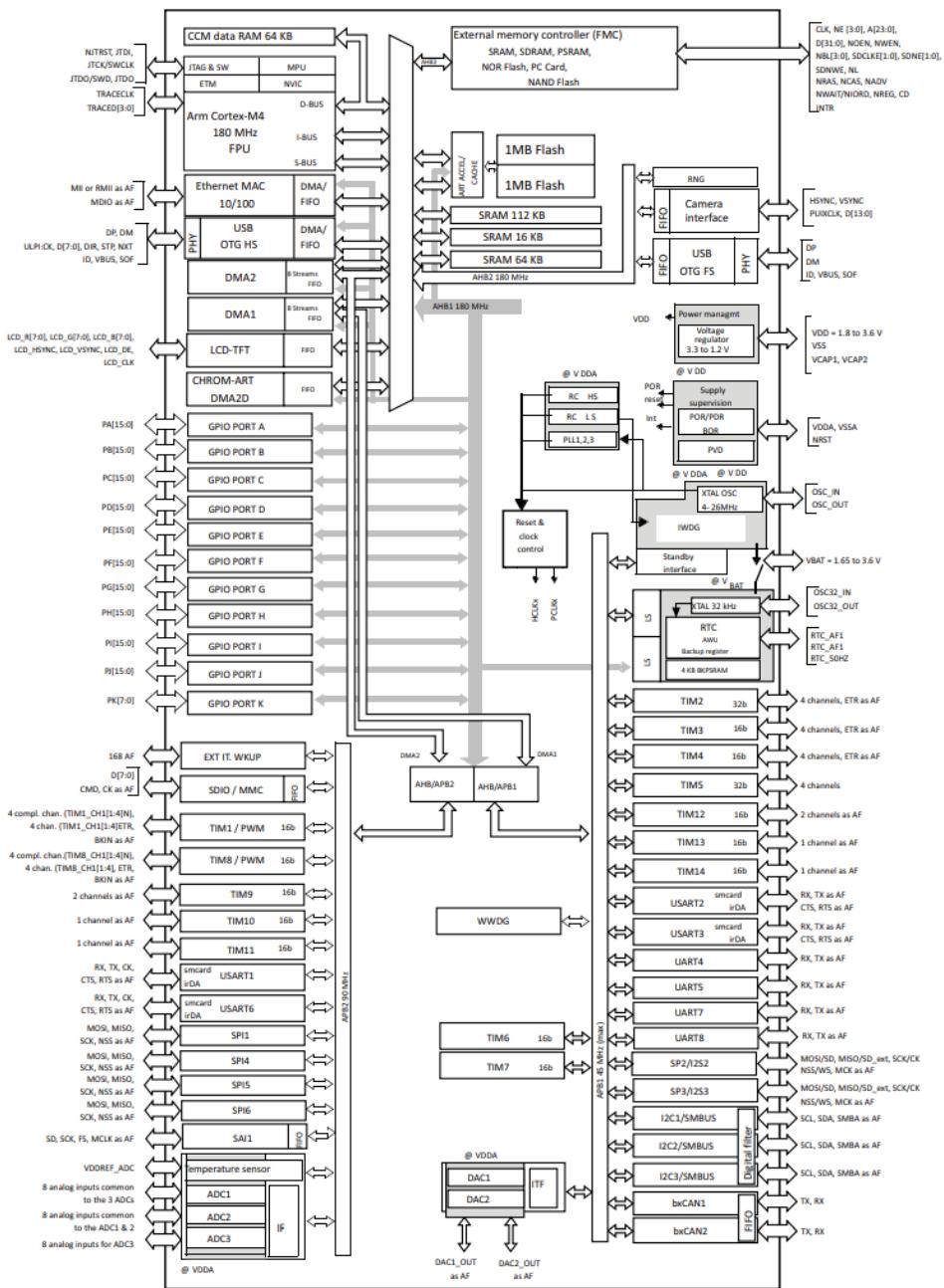


Figura 2: Diagrama de bloques del STM32F429 [1]

Algunas características de este microcontrolador se mencionan a continuación [1]:

- Microcontrolador STM32F429ZI con 2 Mbytes de memoria Flash, 256 Kbytes de RAM en un encapsulado LQFP144.
- ST-LINK/V2 integrado en 32F429IDISCOVERY o ST-LINK/V2-B en STM32F429I-DISC1.
- Soporte para ARM® mbed™ (<http://mbed.org>) con ST-LINK/V2-B solamente.
- ST-LINK USB con capacidad de re-enumeración y tres interfaces diferentes:
 - Puerto COM virtual (solo con ST-LINK/V2-B).

- Almacenamiento masivo (solo con ST-LINK/V2-B).
 - Puerto de depuración.
- Alimentación de la placa: a través del bus USB o mediante una fuente externa de 3 V o 5 V.
 - Pantalla TFT LCD QVGA de 2.4”.
 - SDRAM de 64 Mbits.
 - Sensor de movimiento L3GD20, giroscopio digital de tres ejes con salida digital.
 - Seis LEDs:
 - LD1 (rojo/verde) para comunicación USB.
 - LD2 (rojo) para indicador de encendido de 3.3 V.
 - Dos LEDs de usuario: LD3 (verde), LD4 (rojo).
 - Dos LEDs OTG USB: LD5 (verde) para VBUS y LD6 (rojo) para OC (sobrecorriente).
 - Dos botones pulsadores (usuario y reset).
 - Conector USB OTG con conector micro-AB.
 - Conector de expansión para LQFP144 I/Os, que permite una conexión rápida a la placa de prototipos y facilita el sondeo.
 - Software libre completo que incluye una variedad de ejemplos, como parte de STM32CubeF4 package o STSW-STM32138 para uso con bibliotecas estándar heredadas.

En cuanto a las características eléctricas las siguientes tablas presentan la información de voltajes y corrientes:

Tabla 1: Características de voltaje [1]

Símbolo	Especificaciones	Mín	Máx	Unidad
$V_{DD} - V_{SS}$	Voltaje principal de alimentación externo (incluye V_{DDA} , V_{DD} y V_{BAT}) ⁽¹⁾	-0.3	4.0	V
V_{IN}	Voltaje de entrada en pines FT ⁽²⁾	$V_{SS} - 0,3$	$V_{DD} + 4,0$	V
V_{IN}	Voltaje de entrada en pines TTa	$V_{SS} - 0,3$	4.0	V
V_{IN}	Voltaje de entrada en cualquier otro pin	$V_{SS} - 0,3$	4.0	V
V_{IN}	Voltaje de entrada en el pin BOOT0	V_{SS}	9.0	V
$ \Delta V_{DDx} $	Variaciones entre diferentes pines de alimentación V_{DD}	-	50	mV
$ V_{SSX} - V_{SS} $	Variaciones entre todos los pines de tierra incluyendo V_{REF-}	-	50	mV
$V_{ESD(HBM)}$	Voltaje de descarga electrostática (modelo de cuerpo humano)			

Tabla 2: Características de corriente [1]

Símbolo	Especificaciones	Máx.	Unidad
$\sum I_{VDD}$	Corriente total de entrada en todas las líneas de alimentación V_{DD_x} (fuente) ⁽¹⁾	270	mA
$\sum I_{VSS}$	Corriente total de salida en todas las líneas de tierra V_{SS_x} (sumidero) ⁽¹⁾	-270	mA
I_{VDD}	Corriente máxima de entrada en cada línea de alimentación V_{DD_x} (fuente) ⁽¹⁾	100	mA
I_{VSS}	Corriente máxima de salida en cada línea de tierra V_{SS_x} (sumidero) ⁽¹⁾	-100	mA
I_{IO}	Corriente de salida hundida por cualquier pin de I/O o de control	25	mA
I_{IO}	Corriente de salida suministrada por cualquier pin de I/O o de control	-25	mA
$\sum I_{IO}$	Corriente total de salida hundida por todos los pines de I/O y de control ⁽²⁾	120	mA
$\sum I_{IO}$	Corriente total de salida suministrada por todos los pines de I/O y de control ⁽²⁾	-120	mA
$I_{INJ(PIN)}$	Corriente inyectada en pines FT ⁽⁴⁾	-5/+0	mA
$I_{INJ(PIN)}$	Corriente inyectada en los pines NRST y BOOT0 ⁽⁴⁾	-5/+0	mA
$I_{INJ(PIN)}$	Corriente inyectada en pines TTa ⁽⁵⁾	± 5	mA
$\sum I_{INJ(PIN)}$	Corriente total inyectada en todos los pines de I/O y de control ⁽⁶⁾	± 25	mA

Sensor L3GD20

El L3GD20[2] es un giroscopio de tres ejes utilizado en este proyecto para detectar las variaciones en las oscilaciones en los ejes X, Y y Z. Este sensor de movimiento permite capturar cambios en la orientación y velocidad angular, facilitando así la detección precisa de movimientos del edificio. Sus datos son críticos para registrar las oscilaciones y movimientos que pueden ser causados por vibraciones estructurales, ya que estos valores serán transmitidos al sistema IoT. Seguidamente, se muestra el diagrama de pines del mismo:

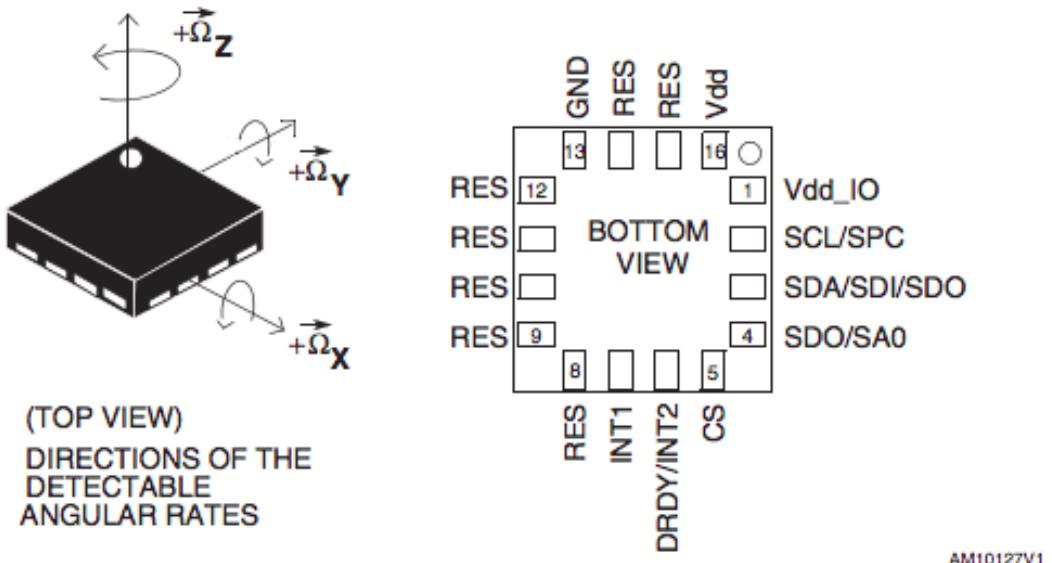


Figura 3: Diagrama de pines del sensor L3GD20[2]

Pantalla LCD/TFT ILI9341

La pantalla LCD/TFT ILI9341 [1] es una pantalla táctil a color que permite la visualización de datos en una resolución de 240x320 píxeles. Además, esta cuenta con un controlador gráfico(**ILI9341**) que facilita la gestión y representación de gráficos en la misma. También cuenta con un controlador táctil (**XPT2046**) que hace posible la interacción táctil. Finalmente, utiliza comunicaciones I2C/SPI que se conecta en el Discovery Kit a través del bus SPI5.

GPIOs (General Purpose Input/Output)

Los pines GPIO del STM32F429 [3] permiten conectar y configurar dispositivos externos en modos de entrada, salida, función alternativa o analógico. En este laboratorio, se plantea utilizar el pin PB0 para configurar en modo analógico las lecturas del voltaje de la batería a través del canal ADC1 IN8, lo cual permitiría el monitoreo continuo del nivel de carga.

ADC

El ADC es un componente que permite la conversión de señales analógicas en digitales. En el caso de este laboratorio, el ADC se desea emplear para medir el nivel de voltaje de la batería. Con una lectura continua del voltaje, es posible determinar el nivel de carga restante y activar una alerta en caso de que el nivel esté cerca del mínimo requerido para el funcionamiento seguro del sistema, asegurando así una operación estable y segura del sismógrafo. [1]

Comunicaciones

La implementación del sismógrafo digital se apoya en diferentes protocolos de comunicación, cada uno con una función específica en el sistema. A continuación, se describe el uso de cada protocolo:

USART

El protocolo USART [6] es fundamental en este laboratorio para la comunicación serie asíncrona entre el microcontrolador y otros dispositivos externos. Este protocolo permite la

transmisión y recepción de datos sin requerir un reloj compartido, lo cual es ventajoso para comunicaciones de bajo costo y configuraciones de largo alcance. En el caso del sismógrafo, UART es utilizado para enviar datos críticos como los valores del giroscopio en los ejes X, Y y Z, y el nivel de batería. Estos datos son recibidos por un script en Python que los envía a un dashboard IoT para la visualización remota y en tiempo real del estado del sistema.

SPI

La interfaz SPI [5] es un protocolo de comunicación sincrónico ideal para la transferencia rápida de datos entre el microcontrolador y otros periféricos. En este laboratorio, SPI se utiliza para la comunicación con el display LCD y el sensor de movimiento L3GD20. SPI permite una transmisión de datos más rápida en comparación con UART, lo cual es especialmente útil para la actualización de imágenes en la pantalla y la lectura continua de datos del giroscopio. La comunicación SPI se basa en un modelo maestro-esclavo, donde el STM32F429 actúa como maestro controlando el reloj y la sincronización de los datos enviados y recibidos.

USB

El USB [7] es una interfaz de comunicación versátil utilizada para la transmisión de datos y la depuración del sistema. Además de soportar modos de depuración y emulación de puerto serie, el USB también ofrece funciones de almacenamiento masivo, permitiendo al dispositivo actuar como una unidad de almacenamiento externa cuando está conectado a una computadora. Esto facilita la transferencia de archivos y la actualización de firmware. La placa STM32F429 Discovery también puede alimentarse a través del puerto USB o mediante una fuente externa de 3V o 5V, lo que otorga flexibilidad en las opciones de alimentación y asegura que el dispositivo continúe operando incluso si el suministro principal es interrumpido.

IoT (Internet of Things)

El Internet de las Cosas (IoT) se refiere a la interconexión de dispositivos y sistemas a través de la red. El sismógrafo digital envía datos a la plataforma IoT Thingsboard a través de un script en Python que recibe los datos por comunicación serial desde el microcontrolador. Esto permite monitorear las mediciones y el estado del dispositivo en tiempo real desde una ubicación remota.[8]

Hardware

Tabla de componentes electrónicos

A continuación, se muestra una tabla con los componentes utilizados para el desarrollo del proyecto:

Tabla 3: Componentes con el precio por unidad

Componente	Cantidad	Precio
STM32F429 Discovery kit	1	15000
Batería 9V	1	3340
Broche para batería de 9V	1	250
Resistencia de 1000Ω	1	15
Resistencia de 550Ω	1	15
Protoboard	1	5500
Total		24120

Firmware y libopencm3

A continuación, se describen las principales librerías utilizadas y su respectiva función:

- **clock.h**: Contiene las declaraciones de funciones para la configuración y control del sistema de reloj en el microcontrolador.
- **console.h**: Define una serie de funciones para la configuración y operación de una consola a través de un puerto serie utilizando el periférico USART.
- **sdram.h**: Define la interfaz para la inicialización y el manejo de la memoria SDRAM.
- **lcd-spi.h**: Define la interfaz y configuración para controlar una pantalla LCD mediante el protocolo SPI.
- **gfx.h**: Define las funciones gráficas de propósito general para el manejo de gráficos en la pantalla TFT/LCD.

Archivos de libopencm3

Para el desarrollo del laboratorio se utilizaron archivos del repositorio de libopencm3 [9] donde se tomaron scripts de los ejemplos que incluye para agilizar el la elaboración, a continuación se explica cada uno de ellos:

- **clock.c**: Este contiene las funciones para configurar los ajustes del clock.
- **console.c**: Este es un ring buffer que mantiene los caracteres a medida que son escritos.
- **gfx.c**: Se encarga de manejar la manipulación gráfica.
- **lcd-spi.c**: Este archivo implementa la comunicación SPI para poder interactuar con la pantalla.
- **sdram.c**: Permite la configuración y operación de la SDRAM, que almacena temporalmente los datos.
- **spi-mems.c**: Implementa la comunicación SPI que interactúa con el giroscopio MEMS.

Desarrollo

Diagrama de flujo

Seguidamente, se muestra el diagrama de flujo que se desarrolla para el sismógrafo:

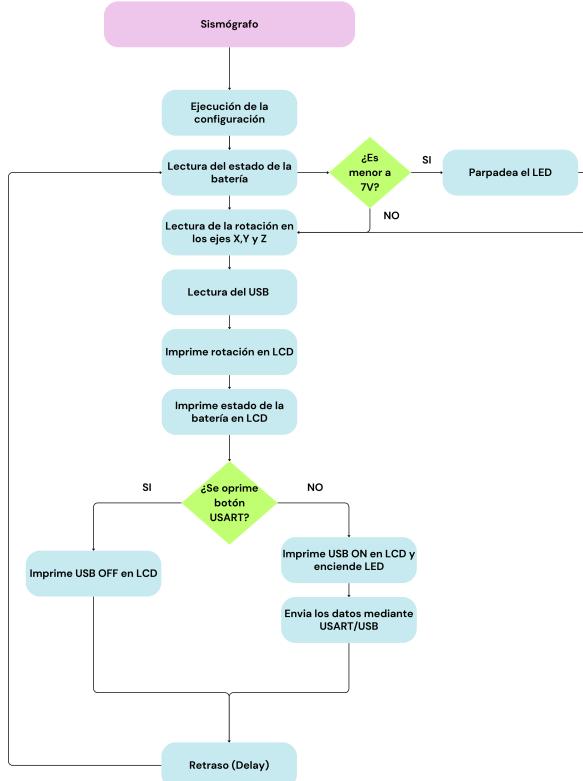


Figura 4: Diagrama de flujo del Sismógrafo

Diseño del circuito con divisor de tensión

Para realizar el diseño del circuito es necesario desarrollar un divisor de tensión. Lo cual es fundamental para la correcta alimentación del sistema y la medición del nivel de tensión de la batería. A continuación, se realizó el cálculo del divisor de tensión considerando que se tiene una resistencia de $R_1 = 1000\Omega$ y otra de $R_2 = 550\Omega$, de manera que se obtiene lo siguiente:

$$V_{out} = V_{in} \cdot \frac{R_1}{R_2 + R_1}$$
$$V_{out} = 7,55 \cdot \frac{1000\Omega}{550\Omega + 1000\Omega}$$
$$V_{out} = \frac{151}{31} \approx 4,87V$$

Por lo tanto, gracias a este divisor de tensión se garantiza una segura utilización de la batería con el microcontrolador para que esta no supere los límites de tensión del mismo.

Funcionamiento del circuito

Se diseñó un código capaz de recrear un sismógrafo digital en el microcontrolador STM32F429. Para ello se debió implementar las siguientes funcionalidades:

- Capturar los valores de los ejes del giroscopio (X, Y, Z).
- Añadir un interruptor o botón que permita activar o desactivar las comunicaciones mediante USART/USB.
- Implementar un LED que parpadee para indicar que la transmisión o recepción de datos mediante USART/USB está activa.
- Medir el nivel de carga de la batería en un rango de [0,9]V. Si la batería se aproxima al nivel mínimo operativo del microcontrolador, se debe activar un LED de alerta intermitente y enviar una notificación de batería baja al tablero de control en Thingsboard.
- Mostrar en la pantalla LCD el nivel de carga de la batería, los valores de los ejes X, Y y Z, y el estado de la comunicación serial/USB.
- Crear un script en Python que lea o escriba datos en el puerto serial/USB y que envíe la información del giroscopio y el nivel de batería a un tablero de IoT en thingsboard.

Al encender el Discovery Kit y cargar el código principal al microcontrolador se puede ver lo siguiente en la pantalla.

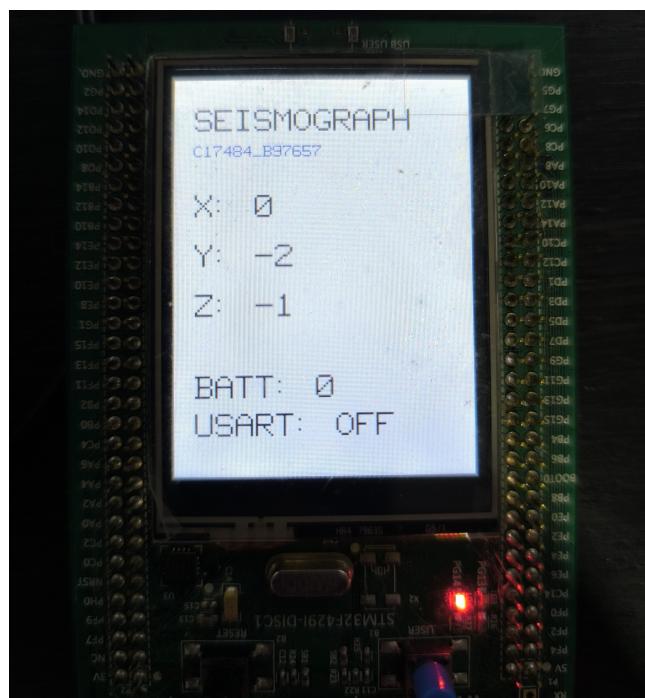


Figura 5: STM32F429 Discovery Kit encendido

En esta pantalla se muestran los valores de los ejes (xyz), el nivel de batería y el estado de la comunicación USART/USB.

Cuando se gira en alguno de los ejes el giroscopio incluido en la placa es posible notar cómo se alteran los valores en la pantalla, tal como en la siguiente imagen.



Figura 6: Lectura de valores de los ejes del giroscopio

Por otro lado, el sismógrafo está conectado a una batería de 9V, con un regulador de voltaje a 5V, el cual es el suministro de voltaje externo recomendado.

El circuito se muestra a continuación:

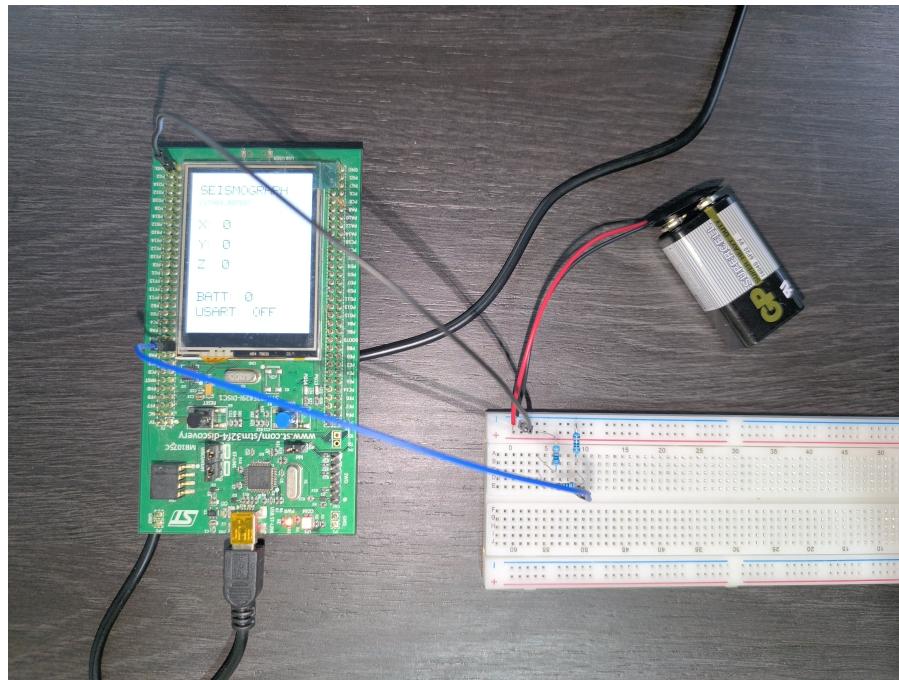


Figura 7: Circuito para conectar la placa a una batería de 9V

En las figuras anteriores se puede notar que cuando se detecta un nivel de batería de 0V el led PG14 de la placa se enciende de forma parpadeante, pues se debe alertar cuando la batería se approxima al nivel mínimo operativo del microcontrolador.

El análisis de voltaje medido se encuentra en la siguiente sección.

Asimismo, otra funcionalidad implementada es la de un botón que permite activar o desactivar las comunicaciones por medio de USART/USB. A continuación se muestra cómo al presionar el botón se cambia el estado en la pantalla a ON y comienza a parpadear la luz led PG13.

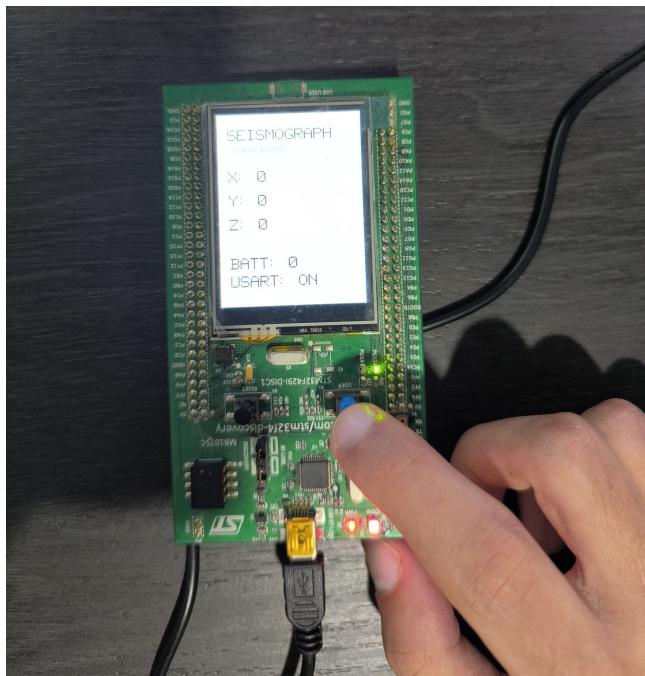


Figura 8: USB COM activado

Finalmente, se creó un código de Python capaz de leer y escribir al puerto serial/USB y enviar la información recolectada por la placa a un dashboard de Thingsboard. Los valores que se imprimen y el dashboard de Thingsboard se pueden observar en las siguientes imágenes.

```
f429i-discovery/Laboratorio4Microcontroladores/iot-script$ python3 iot.py
Connected
['Axis movement and battery level']
[{'X': 0.0, 'Y': 0.0, 'Z': 0.0, 'BATT': 0.0}
{'X': -8.0, 'Y': 41.0, 'Z': 2.0, 'BATT': 0.0}
{'X': 5.0, 'Y': -32.0, 'Z': 0.0, 'BATT': 0.0}
{'X': -9.0, 'Y': 39.0, 'Z': 1.0, 'BATT': 0.0}
{'X': -16.0, 'Y': -14.0, 'Z': -3.0, 'BATT': 0.0}
{'X': -13.0, 'Y': 62.0, 'Z': -1.0, 'BATT': 0.0}
{'X': 1.0, 'Y': -41.0, 'Z': -7.0, 'BATT': 0.0}
{'X': -6.0, 'Y': -23.0, 'Z': 10.0, 'BATT': 0.0}
{'X': -13.0, 'Y': 23.0, 'Z': 2.0, 'BATT': 0.0}
{'X': -10.0, 'Y': -103.0, 'Z': -5.0, 'BATT': 0.0}
{'X': 28.0, 'Y': 23.0, 'Z': 0.0, 'BATT': 0.0}
{'X': -2.0, 'Y': 0.0, 'Z': 0.0, 'BATT': 0.0}]
```

Figura 9: Valores enviados

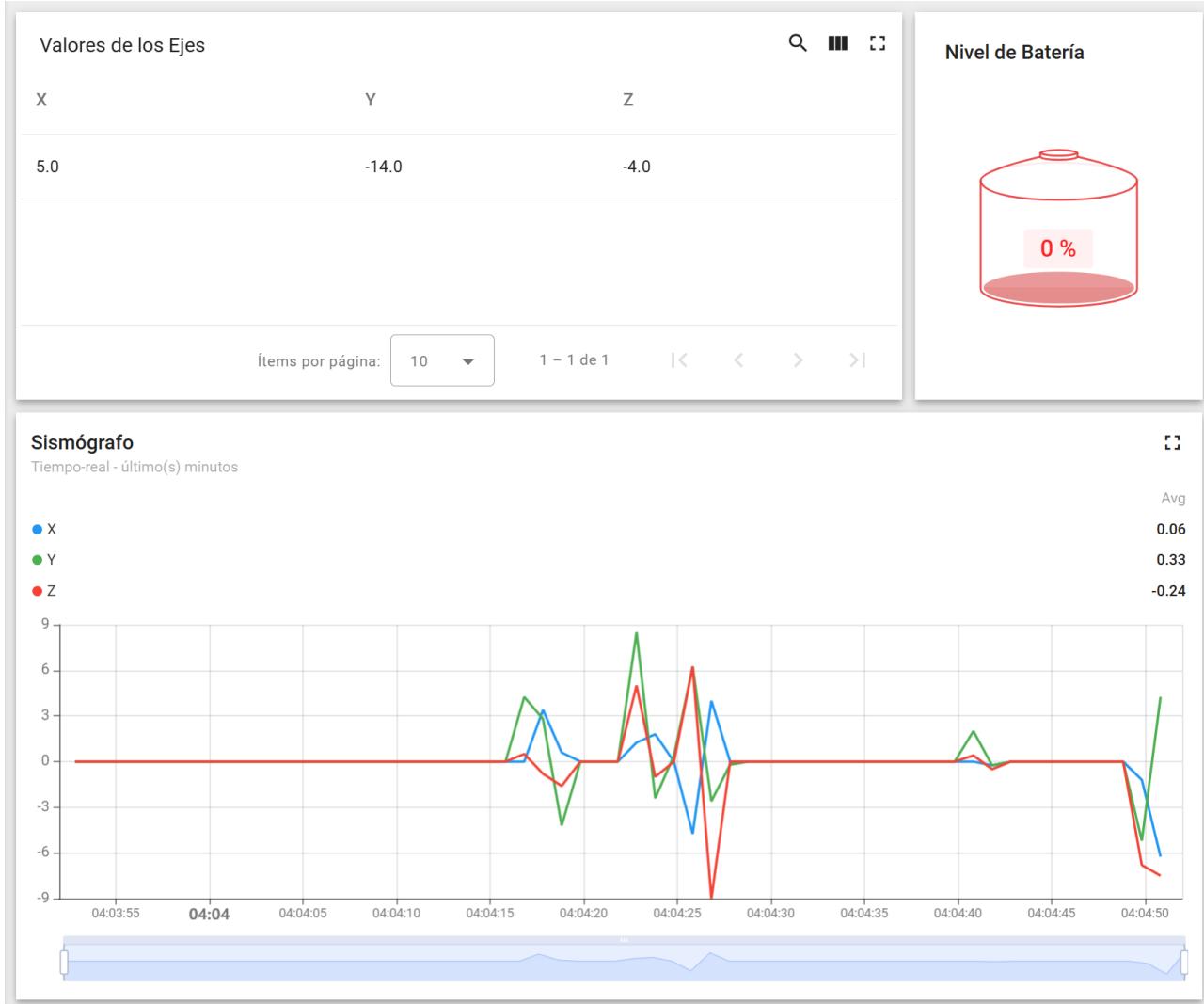


Figura 10: Dashboard de Thingsboard del Sismógrafo para varios valores medidos

Análisis electrónico

Además de los códigos utilizados para darle instrucciones al microcontrolador, se realizó un circuito para conectar este a una batería de 9V, la cual se encontraba a un nivel de 7.55V. Tal como se mencionó anteriormente, se emplearon 3 resistencias ($1k\ \Omega$, $220\ \Omega$ y $330\ \Omega$) para realizar un divisor de tensión y regular esta a 5V. Mediante el uso de un multímetro se midió la tensión a la salida del regulador, obteniendo los 4.87V teóricos calculados previamente.

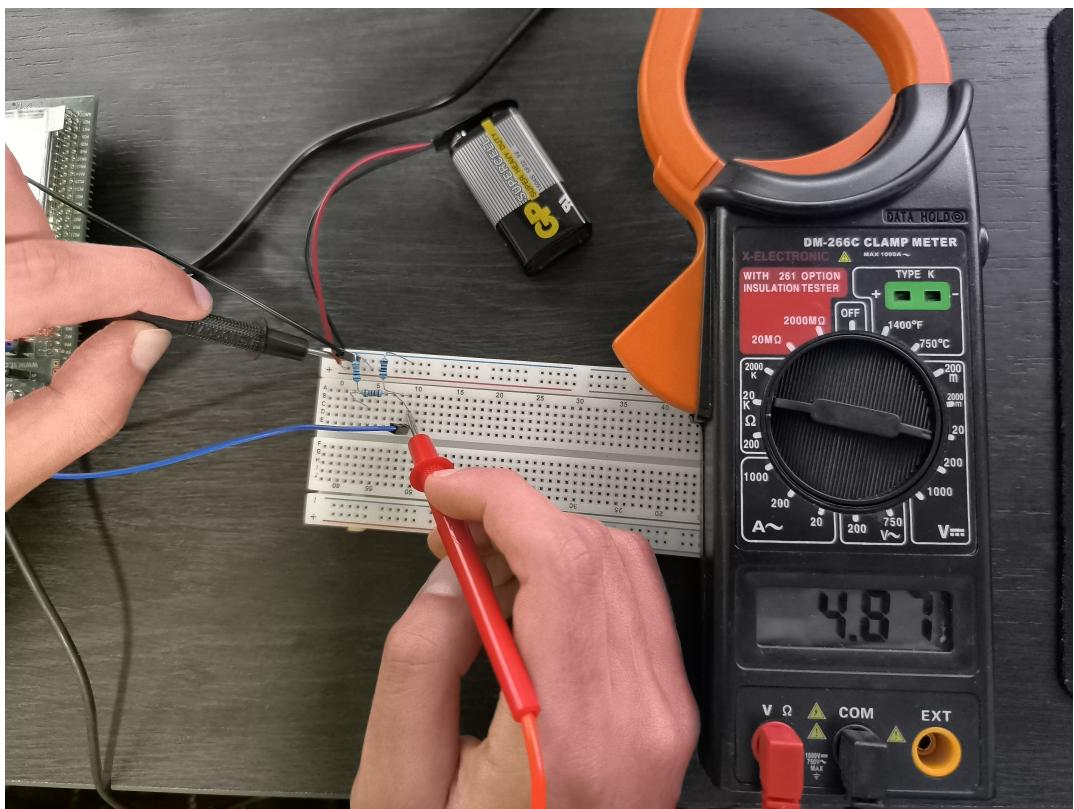


Figura 11: Valor medido a la salida del regulador



Figura 12: Tensión medida de la batería

Se configuró el código para leer la señal de tensión usando el pin PA3, mientras que la tierra se conectó al pin GND. Se debe resaltar que no fue posible leer el valor correcto de la tensión de la batería. Se intentaron distintas alternativas de código y con distintos pines de la placa, pero ninguna con éxito. No obstante, el circuito realizado cumple con lo requerido y las funcionalidades de impresión del valor en la LCD, el parpadeo de una led de alarma si el valor es bajo y la implementación en Thingsboard fueron efectuadas correctamente.

Conclusiones y recomendaciones

El sensor L3GD20 ha mostrado alta sensibilidad y precisión en la detección de oscilaciones, cumpliendo con los requisitos del sismógrafo para detectar vibraciones o giros. Se obtuvieron valores positivos y negativos según el movimiento de forma acertada.

La implementación de protocolos USART y SPI permitió una comunicación eficiente con periféricos como el display LCD y el sensor de movimiento, asegurando una respuesta rápida en la transmisión de datos y enviando estos al puerto serial para posteriormente ser leídos y mostrados en la plataforma IoT.

El uso de la plataforma Thingsboard permitió una visualización remota confiable de los datos del sismógrafo, facilitando el monitoreo de las mediciones en tiempo real, pues a través de widgets en el dashboard la información se procesa de una forma más accesible. Además, la correcta programación del envío de los datos hacia esta plataforma fue esencial, ya que cuenta con un formato específico.

Aunque el circuito para medir la tensión fue funcional, existieron dificultades para lograr las lecturas correspondientes del nivel de batería, lo que requiere ajustes adicionales en el código. Se recomienda revisar las configuraciones del ADC y explorar otras alternativas del código para lograr satisfactoriamente las mediciones de la batería.

Se recomienda además, iniciar explorando cada uno de los ejemplos que ofrece la biblioteca libopencm3, ya que estos ejemplos son muy útiles y cubren una amplia gama de las funcionalidades del microcontrolador STM32F429. Estos ejemplos proporcionan una base sólida para comprender tanto el uso de la biblioteca como el comportamiento y las capacidades del microcontrolador, facilitando así el aprendizaje y la experimentación. Por otro lado, es sumamente necesario iniciar con antelación este laboratorio, con el fin de conseguir los componentes necesarios a tiempo y evacuar dudas en el proceso, ya que se trata de un microcontrolador complejo.

Referencias

- [1] STM Microelectronics, *STM32F427xx STM32F429xx* [Online]. Disponible en: <https://www.st.com/resource/en/datasheet/stm32f427vg.pdf>
- [2] STMicroelectronics, *L3GD20 MEMS Motion Sensor: Three-Axis Digital Output Gyroscope*, Datasheet [Online]. Disponible en: <https://www.st.com/resource/en/datasheet/l3gd20.pdf>.
- [3] Renesas. *Essentials of microcontroller use learning about peripherals: GPIO*. Disponible en: <https://www.renesas.com/us/en/support/engineer-school/mcu-programming-peripherals-01-gpio>.
- [4] Electronics Tutorials. *Analogue to digital converter*. Disponible en: <https://www.electronics-tutorials.ws/combination/analogue-to-digital-converter.html>.
- [5] Grusin, M. *Serial peripheral interface (SPI)*. Disponible en: <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all>.

- [6] Texas Instruments, *Universal synchronous asynchronous receive/transmit USART*. Disponible en: https://www.ti.com/sc/docs/products/micro/msp430/userguid/ag_12.pdf .
- [7] STMicroelectronics. *Introduction to USB with STM32*. Disponible en: https://wiki.st.com/stm32mcu/wiki/Introduction_to_USB_with_STM32.
- [8] Gillis, A. *Internet of Things (IoT)*. Disponible en: <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>.
- [9] Libopencm3 contributors, “libopencm3-examples,” Disponible en: <https://github.com/libopencm3/libopencm3-examples>.

Apéndices

1.3 System requirement

- Windows PC (XP, Vista, 7)
- USB type A to Mini-B USB cable
- ST-LINK/V2
- Supported IDE are EWARM (IAR Embedded Workbench®), MDK-ARM™ and Atollic TrueSTUDIO®

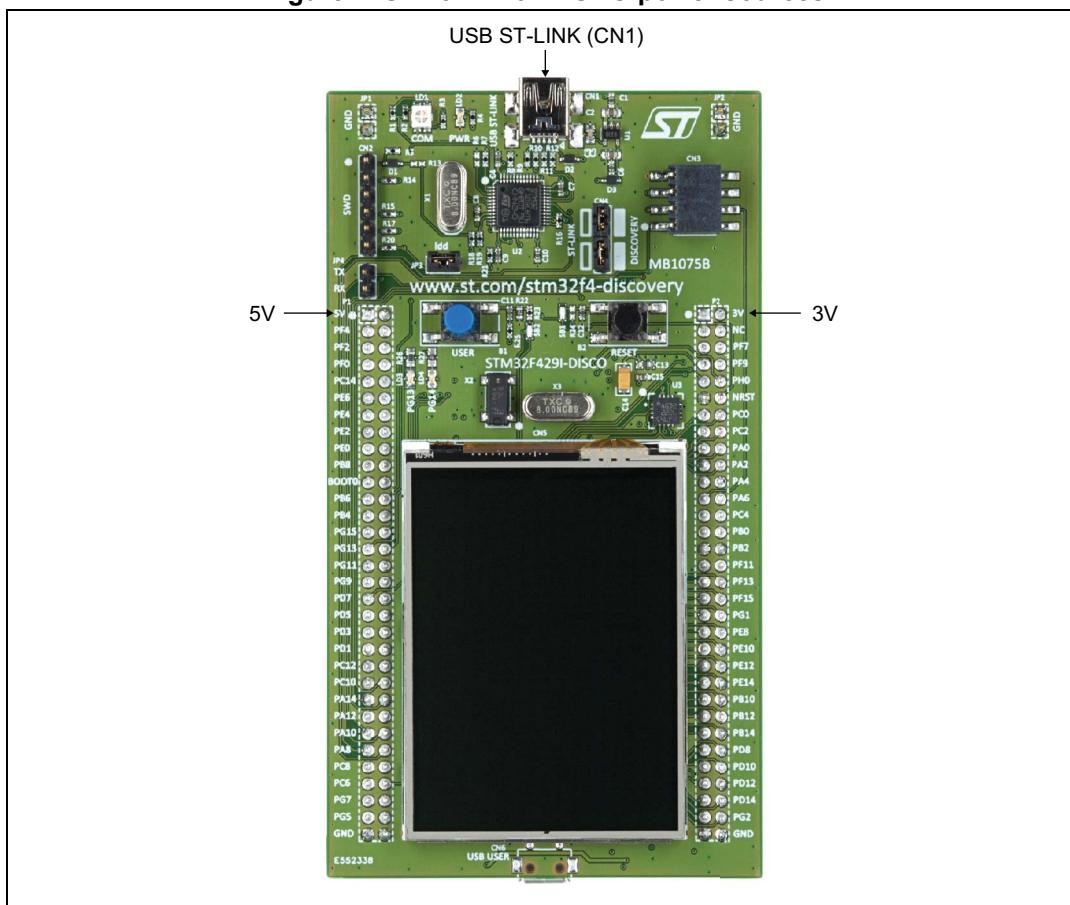
Note: *Required information to download and install desired IDE and ST-LINK/V2 are detailed in Getting started with STM32F429 Discovery software development tools document.*

1.4 Powering up the board

The STM32F429I-DISCO board can be powered up from three sources.

- USB ST-LINK: To power the board from the USB connector CN1, use the 'USB type A to Mini-B' cable and connect it between the host and the board USB connector CN1.
- External sources: DC power supply can be inserted in the GND and 3 V (or 5 V) pin.

Figure 2. STM32F429I-DISCO power sources



1.5 Reset the board

There are three ways to reset the board:

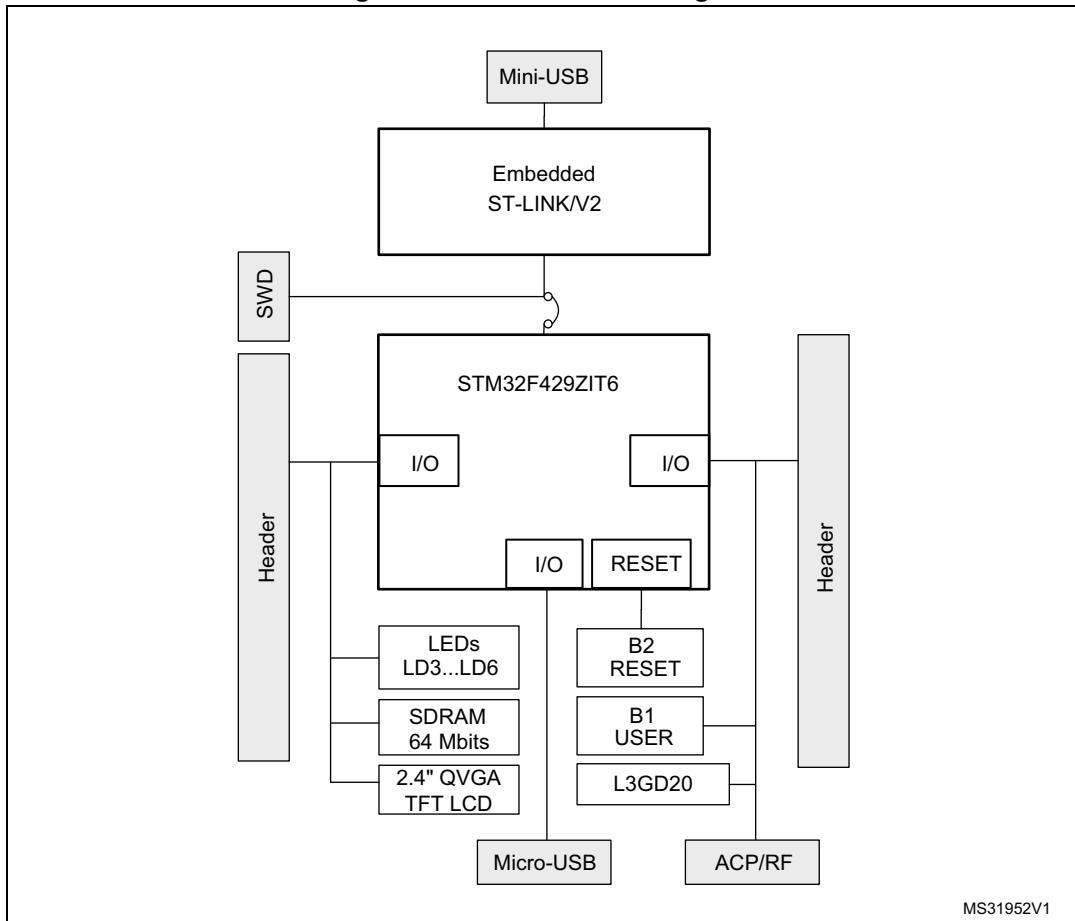
- Push the reset button mounted on the STM32F429I-DISCO.
- Remove and reinsert the USB cable.
- The MCU can also be reset by debuggers.

1.6 Hardware block diagram

The STM32F429I-DISCO is designed around the STM32F429ZIT6U microcontroller in a 144-pin LQFP package. *Figure 3* illustrates the connections between the STM32F429ZIT6U and its peripherals (STLINK/V2, pushbutton, LED, USB and connectors).

Please refer to schematic under www.st.com/stm32f4-discovery for more details.

Figure 3. Hardware block diagram



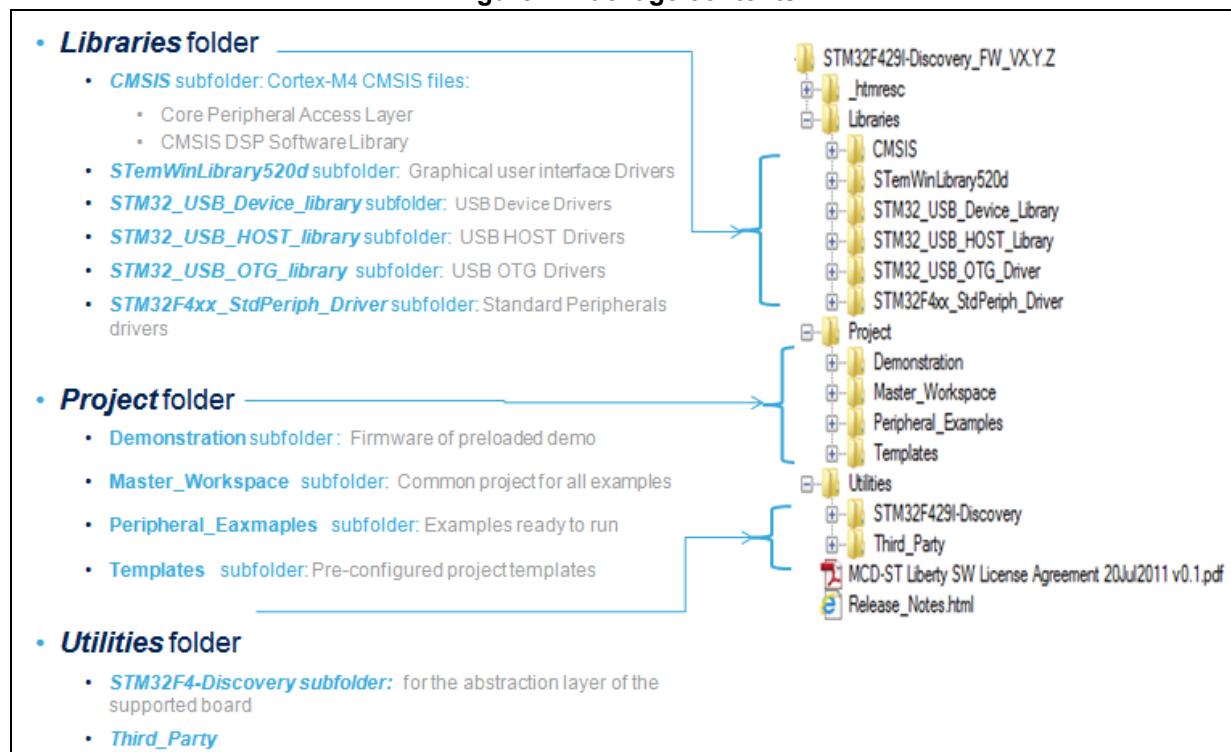
2 Firmware package

To get started with the STM32F429 Discovery kit, a firmware package that contains a set of IP examples and demonstrations of some features exists under www.st.com/stm32f4-discovery.

2.1 Package description

The STM32F429 Discovery firmware applications, demonstration and IPs examples are provided in one single package and supplied in one single zip file. The extraction of the zip file generates one folder, *STM32F429I-Discovery_FW_VX.Y.Z*, which contains the following subfolders.

Figure 4. Package contents



User can run examples provided within this package. A set of examples for each peripheral are ready to be run.

2.2 Programming firmware application

To start programming, user must:

- Install preferred Integrated Development Environment (IDE)
- Install the ST-LINK V2 driver from ST web site

2.2.1 Programming application

To program application (demonstration or example), follow the sequence below:

1. Go under application folder
2. Chose the desired IDE project
3. Double click on the project file (ex. *STM32F429I-Discovery_Demo.eww* for EWARM)
4. Rebuild all files: Project->Rebuild all
5. Load project image: Project->Debug
6. Run program: Debug->Go

Please refer to **Getting started with STM32F429 Discovery software development tools** for more details.

2.2.2 Run pre-loaded demo

To run and develop any firmware applications on your STM32F429 Discovery board, the minimum requirements are as follows:

- Windows PC (XP, Vista, 7)
- 'USB type A to Mini-B' cable, used to power the board (through USB connector CN1) from host PC and connect to the embedded ST-LINK/V2 for debugging and Programming.

Additional hardware accessories will be needed to run some applications:

- 'USB type A to Micro-B' cable, used to connect the board (through USB connector CN5) as USB Device to host PC.

Establish the connection with the STM32F429 Discovery board as follows:

Figure 5. Hardware environnement



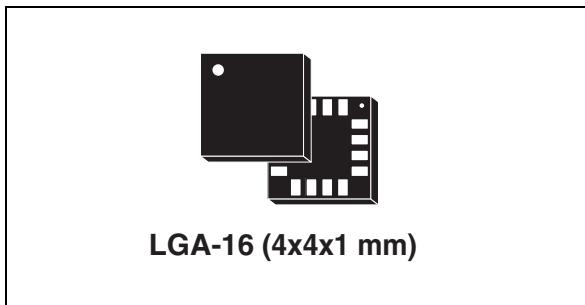
The demonstration software, based on the STemWin GUI library, is already preloaded in the board's Flash memory. It uses the LCD TFT mounted on the board to show the Menu based-on-icon view widget (Image Browser, Game, Performance, Clock/Calendar, Video and System Info module). The status bar indicate the CPU Usage, date, USB disk flash connection state, alarm and time.

Follow the sequence below to configure the STM32F429 Discovery board and launch the DISCOVER application:

1. Ensure that the jumpers JP3 and CN4 are set to "on" (Discovery mode).
2. Connect the STM32F429 Discovery board to a PC using a USB cable type A/mini-B through the USB ST-LINK connector CN1, to power the board. The LEDs LD2 (PWR) and LD1 (COM).
3. The following applications are available on the screen:
 - Clock/Calendar and Game
 - Video Player and Image Browser (play videos and view images from the USB mass storage connected to CN6)
 - Performance monitor (watch the CPU load and run a graphical benchmark)
 - System Info
4. The demo software, as well as other software examples that allow you to discover the STM32 F4 series features, are available on www.st.com/stm32f4-discovery.
5. Develop your own applications starting from the examples.

MEMS motion sensor: three-axis digital output gyroscope

Datasheet - production data



Features

- Three selectable full scales (250/500/2000 dps)
- I²C/SPI digital output interface
- 16 bit-rate value data output
- 8-bit temperature data output
- Two digital output lines (interrupt and data ready)
- Integrated low- and high-pass filters with user-selectable bandwidth
- Wide supply voltage: 2.4 V to 3.6 V
- Low voltage-compatible IOs (1.8 V)
- Embedded power-down and sleep mode
- Embedded temperature sensor
- Embedded FIFO
- High shock survivability
- Extended operating temperature range (-40 °C to +85 °C)
- ECOPACK® RoHS and “Green” compliant

Applications

- Gaming and virtual reality input devices
- Motion control with MMI (man-machine interface)
- GPS navigation systems
- Appliances and robotics

Description

The L3GD20 is a low-power three-axis angular rate sensor.

It includes a sensing element and an IC interface capable of providing the measured angular rate to the external world through a digital interface (I²C/SPI).

The sensing element is manufactured using a dedicated micro-machining process developed by STMicroelectronics to produce inertial sensors and actuators on silicon wafers.

The IC interface is manufactured using a CMOS process that allows a high level of integration to design a dedicated circuit which is trimmed to better match the sensing element characteristics. The L3GD20 has a full scale of ±250/±500/ ±2000 dps and is capable of measuring rates with a user-selectable bandwidth.

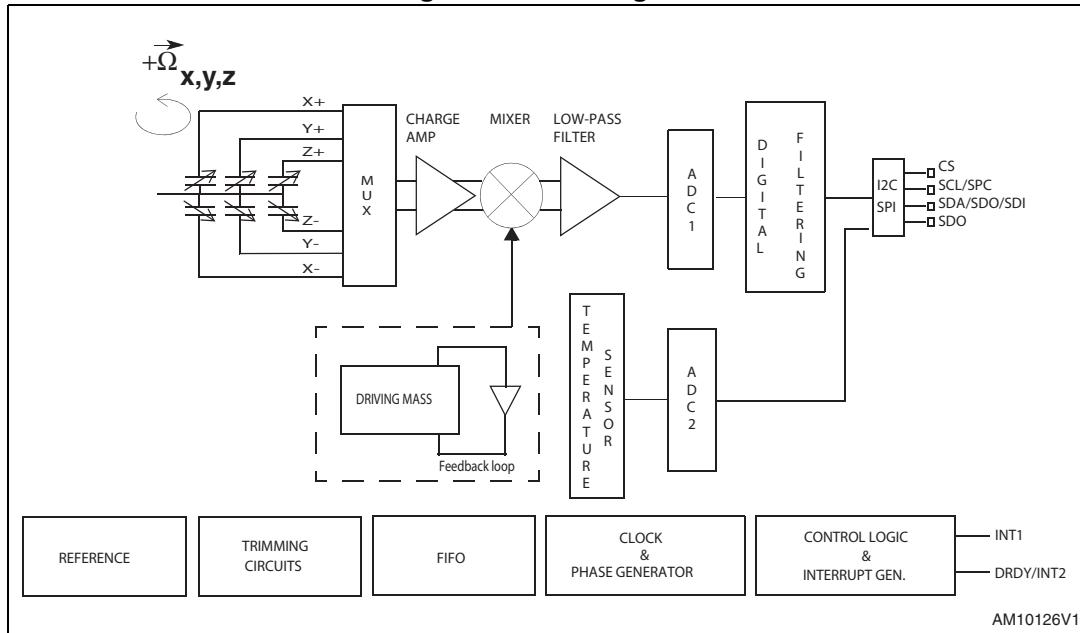
The L3GD20 is available in a plastic land grid array (LGA) package and can operate within a temperature range of -40 °C to +85 °C.

Table 1. Device summary

Order code	Temperature range (°C)	Package	Packing
L3GD20	-40 to +85	LGA-16 (4x4x1 mm)	Tray
L3GD20TR	-40 to +85	LGA-16 (4x4x1 mm)	Tape and reel

1 Block diagram and pin description

Figure 1. Block diagram



Note: The vibration of the structure is maintained by drive circuitry in a feedback loop. The sensing signal is filtered and appears as a digital signal at the output.

1.1 Pin description

Figure 2. Pin connection

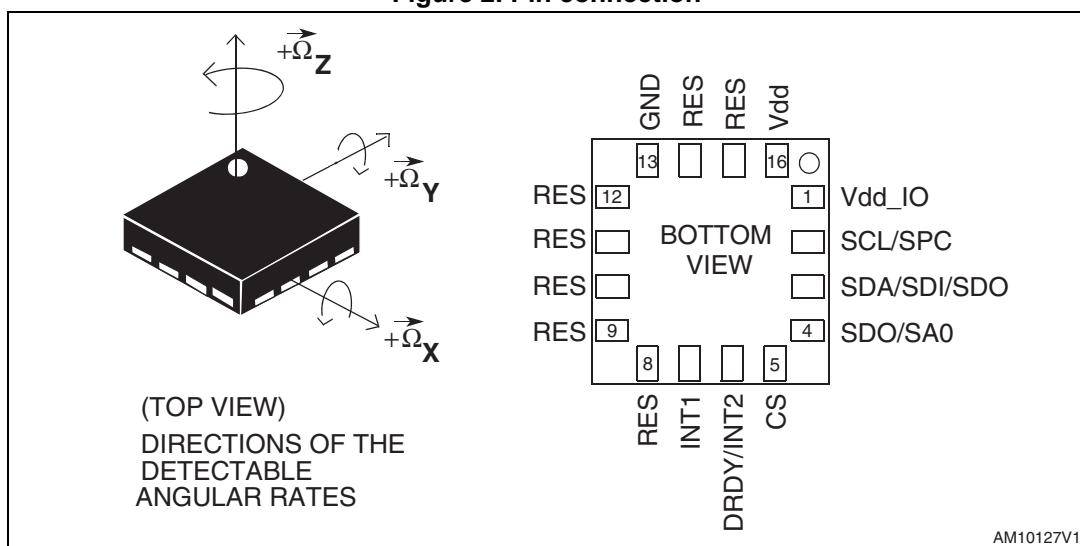


Table 2. Pin description**Table 3.**

Pin#	Name	Function
1	Vdd_IO ⁽¹⁾	Power supply for I/O pins
2	SCL SPC	I ² C serial clock (SCL) SPI serial port clock (SPC)
3	SDA SDI SDO	I ² C serial data (SDA) SPI serial data input (SDI) 3-wire interface serial data output (SDO)
4	SDO SA0	SPI serial data output (SDO) I ² C less significant bit of the device address (SA0)
5	CS	I ² C/SPI mode selection (1: SPI idle mode / I ² C communication enabled; 0: SPI communication mode / I ² C disabled)
6	DRDY/INT2	Data ready/FIFO interrupt (Watermark/Overrun/Empty)
7	INT1	Programmable interrupt
8	Reserved	Connect to GND
9	Reserved	Connect to GND
10	Reserved	Connect to GND
11	Reserved	Connect to GND
12	Reserved	Connect to GND
13	GND	0 V supply
14	Reserved	Connect to GND with ceramic capacitor ⁽²⁾
15	Reserved	Connect to Vdd
16	Vdd ⁽³⁾	Power supply

1. 100 nF filter capacitor recommended.
2. 1 nF min value must be guaranteed under 11 V bias condition.
3. 100 nF plus 10 μ F capacitors recommended.

2 Mechanical and electrical specifications

2.1 Mechanical characteristics

@ Vdd = 3.0 V, T = 25 °C unless otherwise noted.

Table 4. Mechanical characteristics⁽¹⁾

Symbol	Parameter	Test condition	Min.	Typ. ⁽²⁾	Max.	Unit
FS	Measurement range	User-selectable		±250		dps
				±500		
				±2000		
So	Sensitivity	FS = 250 dps		8.75		mdps/digit
		FS = 500 dps		17.50		
		FS = 2000 dps		70		
SoDr	Sensitivity change vs. temperature	From -40 °C to +85 °C		±2		%
DVoff	Digital zero-rate level	FS = 250 dps		±10		dps
		FS = 500 dps		±15		
		FS = 2000 dps		±75		
OffDr	Zero-rate level change vs. temperature	FS = 250 dps		±0.03		dps/°C
		FS = 2000 dps		±0.04		dps/°C
NL	Non linearity	Best fit straight line		0.2		% FS
Rn	Rate noise density			0.03		fps/(√Hz)
ODR	Digital output data rate			95/190/ 380/760		Hz
Top	Operating temperature range		-40		+85	°C

1. The product is factory calibrated at 3.0 V. The operational power supply range is specified in [Table 5](#).

2. Typical specifications are not guaranteed.

2.2 Electrical characteristics

@ Vdd =3.0 V, T=25 °C unless otherwise noted.

Table 5. Electrical characteristics (1)

Symbol	Parameter	Test condition	Min.	Typ. ⁽²⁾	Max.	Unit
Vdd	Supply voltage		2.4	3.0	3.6	V
Vdd_IO	I/O pins supply voltage ⁽³⁾		1.71		Vdd+0.1	V
Idd	Supply current			6.1		mA
IddSL	Supply current in sleep mode ⁽⁴⁾	Selectable by digital interface		2		mA
IddPdn	Supply current in power-down mode	Selectable by digital interface		5		µA
VIH	Digital high level input voltage		0.8*Vdd_I_O			V
VIL	Digital low level input voltage				0.2*Vdd_I_O	V
Top	Operating temperature range		-40		+85	°C

1. The product is factory calibrated at 3.0 V.
2. Typical specifications are not guaranteed.
3. It is possible to remove Vdd maintaining Vdd_IO without blocking the communication busses; in this condition the measurement chain is powered off.
4. Sleep mode introduces a faster turn-on time relative to power-down mode.

2.3 Temperature sensor characteristics

@ Vdd =3.0 V, T=25 °C unless otherwise noted.

Table 6. Electrical characteristics (1)

Symbol	Parameter	Test condition	Min.	Typ. ⁽²⁾	Max.	Unit
TSDr	Temperature sensor output change vs. temperature	-		-1		°C/digit
TODR	Temperature refresh rate			1		Hz
Top	Operating temperature range		-40		+85	°C

1. The product is factory calibrated at 3.0 V.
2. Typical specifications are not guaranteed.

2.4 Communication interface characteristics

2.4.1 SPI - serial peripheral interface

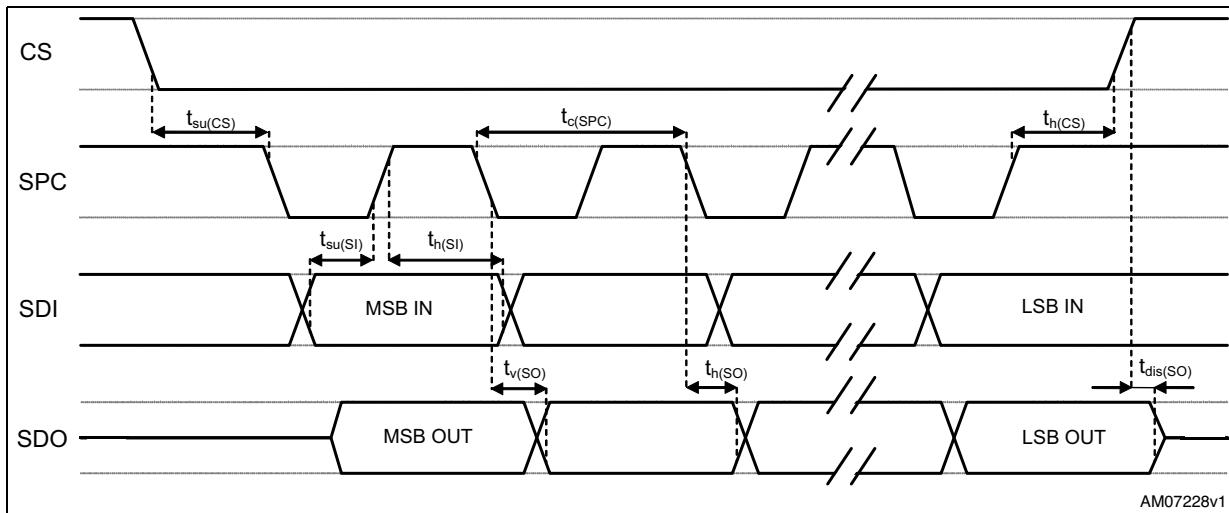
Subject to general operating conditions for Vdd and T_{op} .

Table 7. SPI slave timing values

Symbol	Parameter	Value ⁽¹⁾		Unit
		Min	Max	
tc(SPC)	SPI clock cycle	100		ns
fc(SPC)	SPI clock frequency		10	MHz
tsu(CS)	CS setup time	5		ns
th(CS)	CS hold time	8		
tsu(SI)	SDI input setup time	5		
th(SI)	SDI input hold time	15		
tv(SO)	SDO valid output time		50	
th(SO)	SDO output hold time	6		
tdis(SO)	SDO output disable time		50	

1. Values are guaranteed at a 10 MHz clock frequency for SPI with both 4 and 3 wires, based on characterization results; not tested in production.

Figure 3. SPI slave timing diagram (a)



a. Measurement points are at $0.2 \cdot V_{dd_IO}$ and $0.8 \cdot V_{dd_IO}$, for both input and output port.