

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0624: Laboratorio de Microcontroladores

1 ciclo 2023

Laboratorio 4

STM32: GPIO, ADC, comunicaciones, IOT

Jimmy Manley Peña

Roberto Acevedo

Profesor: Marco Villalta Fallas

4 de junio del 2023

Resumen

Se implementa un sismógrafo mediante el microcontrolador STM32F29. Este microcontrolador posee un giroscopio que se usará como sensor de aceleración en los 3 ejes en 3 dimensiones X, Y y Z. Y así tomar medidas sismográficas.

También se cuenta con una batería de alimentación y un medidor del nivel de la batería que de encontrarse en un nivel bajo detiene el envío de datos a la plataforma IoT. Mediante una batería de 9V y un divisor de tensión se envía el voltaje de alimentación a la placa STM.

Link del repositorio: https://github.com/robertoacevedomora/Laboratorio_4

1. Nota teórica

1.2. Microcontrolador STM32F429

Se hace uso del STM32F429 de la familia STM. Este microcontrolador cuenta con las siguientes características:

- 2 Megabytes de memoria flash
- 256 Kilobytes de RAM
- Conexión USB OTG a Micro-AB
- El giroscopio I3G4250D ST MEMS con 3 ejes
- 6 LEDs
- Dos botones de usuario y reset
- 64-Mbit de SDRAM
- Pantalla QVGA TFT LCD de 2.4 pulgadas

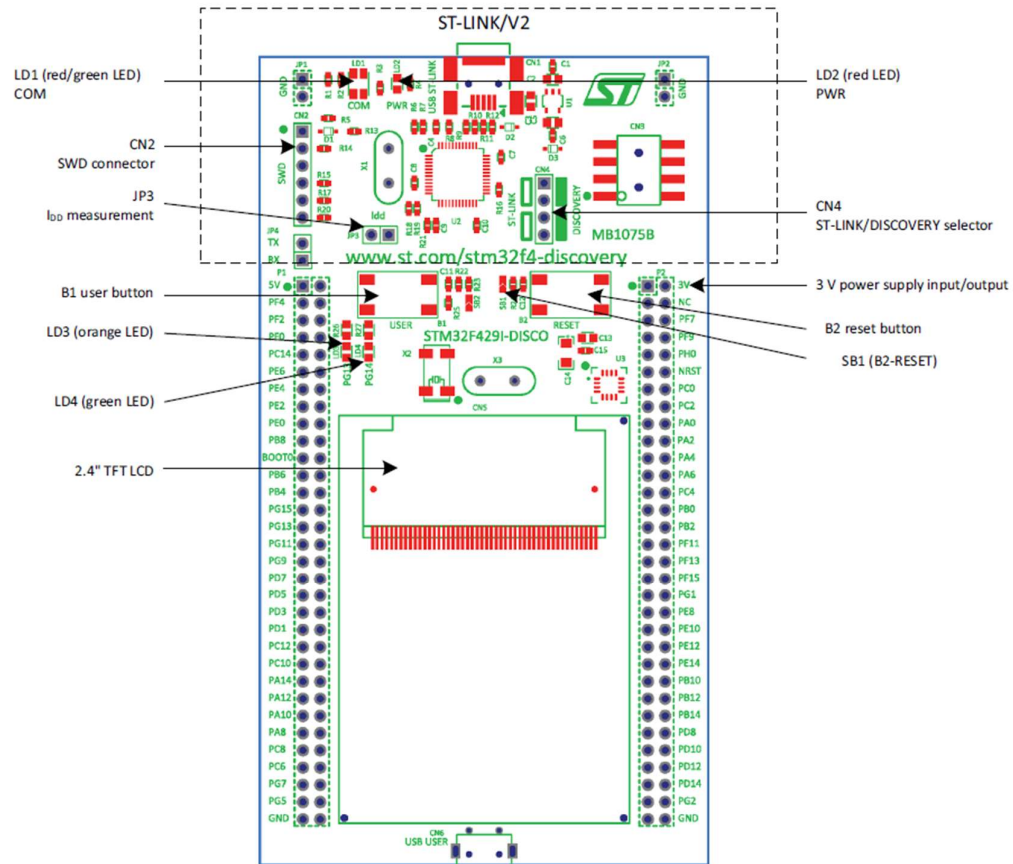


Figura 1. Diagrama de pines del microcontrolador STM32F429 por arriba. [1]

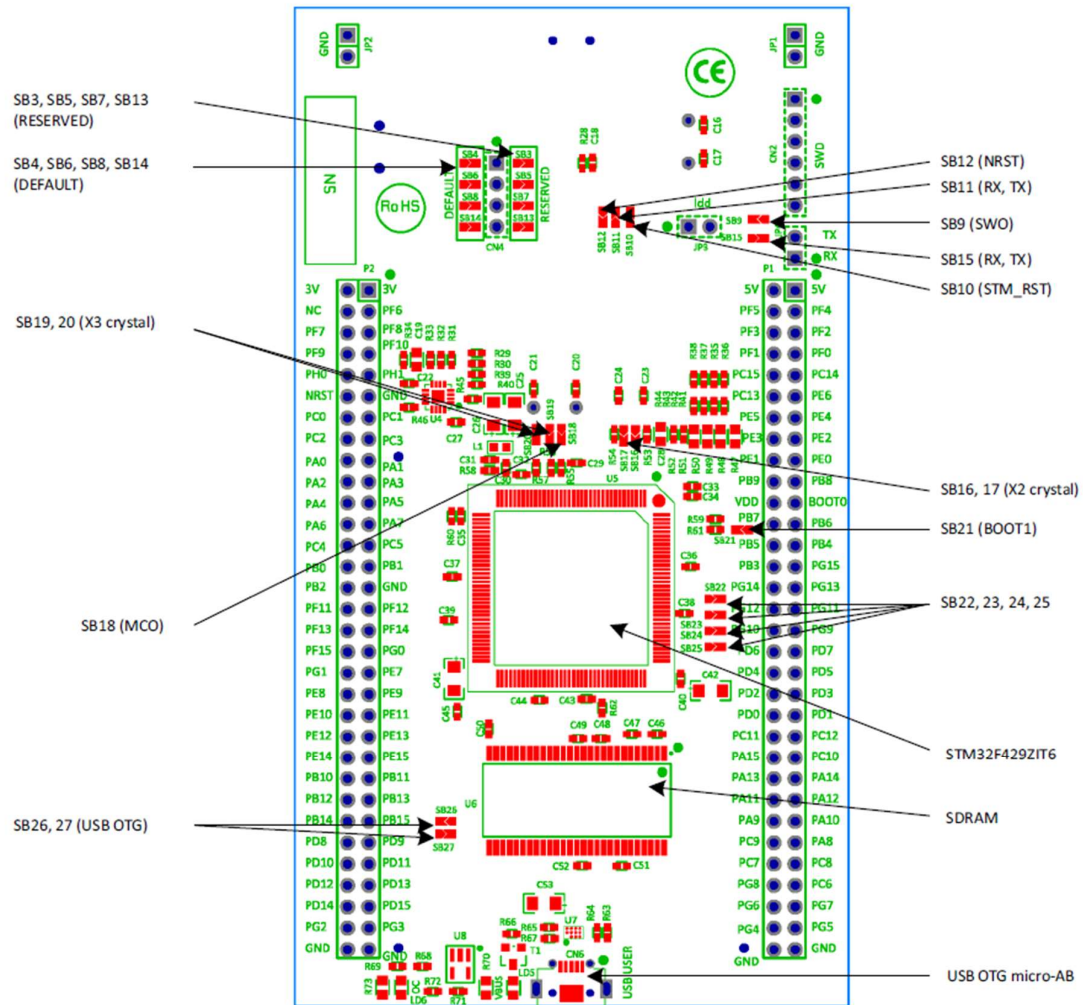


Figura 2. Diagrama de pines del microcontrolador STM32F429 por abajo. [1]

Las mediciones son realizadas con sensores de inercia. []

Pantalla LCD

Es una pantalla a colores táctil, con una resolución de 240x320 píxeles. Se compone de un controlador gráfico ILI9341, un controlador táctil XPT2046 y maneja comunicaciones I2C/SPI. Está conectada en la STM32F429 Discovery Kit al SPI5 y se configura por medio de instrucciones SPI.

1.3. Otros componentes electrónicos utilizados:

Batería externa de 9V junto con una protoboard para suministrar 5V exactamente. Se usaron resistores de 400 Ohm para diseñar el divisor de tensión.

Se diseña un divisor de tensión con resistores. Para pasar de 9 v A 5 v.

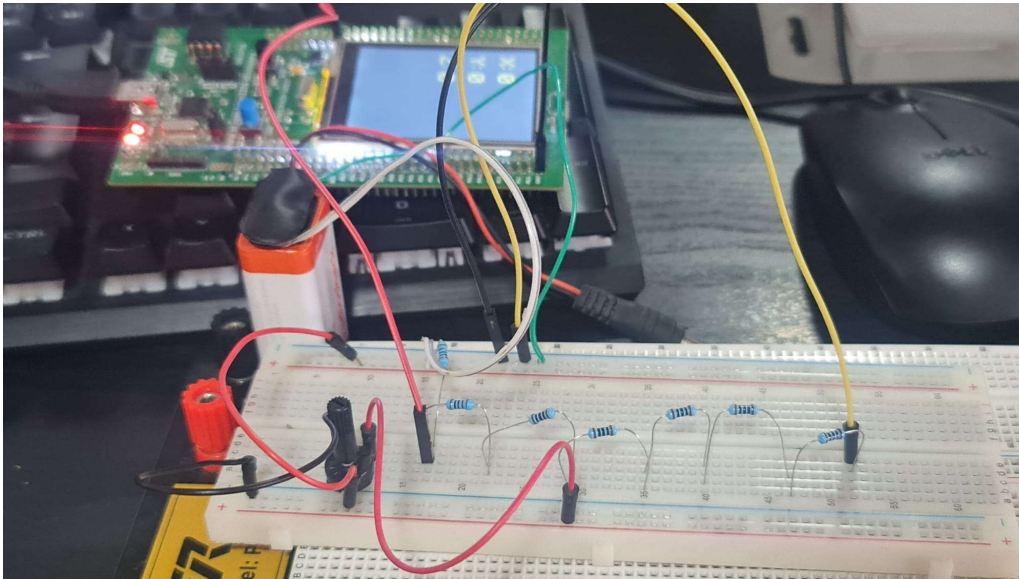


Figura 4. Divisor de tensión.

Este divisor de tensión permite que la máxima tensión medida a la entrada del microcontrolador sea 5 V cuando la tensión de la batería es de 9V, por lo cual no se superan los límites de operación del pin de entrada.

Ecuación: $V_{out} = R_2 \cdot V_{in} / (R_1 + R_2)$

Siendo $V_{out} = 5\text{ V}$ y $V_{in} = 9\text{ V}$. Los valores escogidos para el diseño fueron de $2.5\text{ k}\Omega$ para R_2 y $2\text{ k}\Omega$ para R_1 . Realmente se utilizan 5 resistencias de $100\ \Omega$ y 1 de $2\text{ k}\Omega$ en serie para.

La tensión de salida se mide en V_{out} . Dicha terminal se conecta al pin PA0.

BOM:

Componente	Tipo	Precio	Cantidad
STM32F429 Discovery Kit		\$90.00	1
Resistor	$2\text{ K}\ \Omega$	\$0.16	1
Resistor	$2.5\text{ K}\ \Omega$	\$0.16	1
Batería	9V	\$2.00	1
Total	-	\$93.31	4

Tabla 1: Lista de componentes utilizados

2. Librerías utilizadas

2.1. libopencm3

La librería libopencm3 es un proyecto de “open source” creado para facilitar el uso de varios microcontroladores ARM. Durante el laboratorio fue de gran ayuda ya que se usaron distintas funciones de repositorios disponibles en la librería. Funciones para controlar e inicializar el giroscopio por ejemplo, funciones para escribir en la pantalla del STM32 y funciones para leer el voltaje de un pin en específico y desplegarlo en la pantalla.

3. Desarrollo/Análisis de resultados

3.1. IoT

Usando el puerto serial y un script de python se desplegaron los datos de los tres ejes del giroscopio en la terminal y también en la plataforma Thingsboard usando un dashboard de tres columnas.

3.2. Funcionalidad electrónica

En las siguientes dos imágenes se muestra en microcontrolador sin movimiento (figura 8) y seguidamente el microcontrolador en movimiento (figura 9). Los ejes son representados con las letras X, Y y Z. Se le puso a la pantalla un fondo negro con dígitos amarillos para mayor visibilidad.

Figura 8. Datos del giroscopio en la placa STM32F429 sin aceleración.

Figura 9. Datos del giroscopio en la placa STM32F429 con aceleración.

La conexión con el puerto serial se logró satisfactoriamente y se muestra a continuación cómo se despliegan los datos de la aceleración de los ejes en la terminal. Se tiene un delay de menos de 5 segundos entre cada medición.

Figura 10. Resultados en consola de datos provenientes del acelerómetro.

3.3 Firmware

La construcción del firmware se lleva a cabo en lenguaje c, se crean una serie de funciones para programar los periféricos del controlador, los puertos que se utilizan y los sistemas de comunicación.

Para poder ejecutar las funciones que permiten utilizar el giroscopio L3GD20 se utiliza el protocolo SPI, en este caso se usa el SPI5. Tomando como referencia el ejemplo spi-mems.c, para acceder al puerto SPI5 se configura el SCK (PF7), MISO (PF8), MOSI (PF9), CS (PC1) , INT1 = PA1, MEMS INT2 = PA2. Por lo tanto, el protocolo SPI se configura con la función spi_setup (como referencia el archivo spi.c de los ejemplos), se utilizan las funciones de la librería libopenm3 para inicializar el protocolo de comunicación SPI. Se habilita el reloj con la función rcc_periph_clock_enable para el SPI5, el puerto C y el puerto F. Para configurar los puertos se utiliza la función gpio_mode_setup.

Para poder transmitir los datos se utiliza el protocolo USART, al igual que con la función spi_setup se tomó como referencia el ejemplo spi.c. En este caso se habilitó el pin 9 del puerto A para la transmisión y se configuraron los parámetros de transmisión.

La función `led_con_setup` habilita el reloj de los puertos A y G. En el caso del puerto A, el pin 0 como entrada y para el puerto G, los pines 13 y 14 como salidas.

El dispositivo contiene un conjunto de registros que se utilizan para controlar su comportamiento y recuperar datos de velocidad angular. La dirección de registro, que consta de 7 bits, se utiliza para identificarlos y para escribir los datos a través de la interfaz serial, esto se lleva a cabo en la función `giro_setup`.

La función `leer_ejes` permite obtener el valor de los 3 ejes del giroscopio y los devuelve a las variables `gyro_x`, `gyro_y` y `gyro_z`. Se utilizan los registros de lectura `STATUS_REG`, `OUT_TEMP`, `OUT_X_H`, `OUT_X_L`, `OUT_Y_H`, `OUT_Y_L`, `OUT_Z_H`, `OUT_Z_L` y el registro identificador único `WHO_AM_I`. La función `print_decimal` se utiliza para imprimir los valores en la consola, la función se tomó de los ejemplos, específicamente del archivo `spi-mems.c`.

3.4 Análisis de resultados

En la figura 4 se observa el funcionamiento del giroscopio, es posible visualizar el valor del eje y, x y z. En este caso la conexión o transmisión de datos está apagada, por lo cual el LED del puerto G pin 13 se encuentra apagado tal y como se muestra en la figura 4.



Figura 5. Ejes del giroscopio.

En la figura 6, se observa la conexión encendida “ON”, por lo cual se están transmitiendo los datos del microcontrolador para ser desplegados en un dashboard de una plataforma IoT thingsboard. También es posible ver que el LED del puerto G pin 13 se enciende por la transmisión serial.



Figura 6. Transmisión de datos encendida

Los datos enviados por el puerto USB son enviados a un script de Python tal y como se muestra en la figura 7. Posteriormente se envían a la plataforma IoT y se visualizan en la figura 8.

```
data published to thingsboard /n
{"X": "-15", "Y": "87", "Z": "10", "Nivel": "0"}
data published to thingsboard /n
{"X": "-10", "Y": "61", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "-3", "Y": "23", "Z": "3", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "15", "Z": "3", "Nivel": "0"}
data published to thingsboard /n
{"X": "1", "Y": "3", "Z": "2", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "0", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "-2", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "-1", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "-3", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "1", "Y": "-1", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "-1", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "0", "Y": "-1", "Z": "0", "Nivel": "0"}
data published to thingsboard /n
{"X": "4", "Y": "-16", "Z": "1", "Nivel": "0"}
data published to thingsboard /n
{"X": "6", "Y": "-44", "Z": "1", "Nivel": "0"}
data published to thingsboard /n
{"X": "-5", "Y": "-70", "Z": "5", "Nivel": "0"}
data published to thingsboard /n
```

Figura 7. Transmisión de datos (Python)

Timeseries table 🔍 ⚙️

🕒 Tiempo-real - último(s) minutos

Timestamp ↓	X	Y	Z
2023-06-04 18:51:06	16	-109	-24
2023-06-04 18:51:05	-3	23	4
2023-06-04 18:51:04	10	9	-1
2023-06-04 18:51:03	-37	98	0
2023-06-04 18:51:02	-15	5	-4
2023-06-04 18:51:01	5	-39	-15
2023-06-04 18:51:00	-9	-104	-7
2023-06-04 18:50:59	43	-156	-34
2023-06-04 18:50:58	5	-76	7
2023-06-04 18:50:57	-1	111	12

Ítems por página: 1 ~ 10 de 60 ⏪ ⏩

Figura 8. Datos en la plataforma IoT (Widget).

5. Conclusiones y recomendaciones

- Se logra utilizar el STM32F4, el giroscopio y la pantalla de manera exitosa.
- Se encuentra una utilidad extra del microcontrolador y su potencial para trabajar en tareas más complejas que no necesariamente tienen que ver con variables digitales sino también con variables y mediciones físicas.
- Se experimentó la utilidad del Internet of Things haciendo uso de la plataforma Thingsboard, siendo capaz de habilitar pizarras para el análisis y monitoreo de diferentes tareas.
- La librería libopen fue de gran ayuda y facilita el trabajo ya que tiene definidos los registros del microcontrolador y sus periféricos. De igual forma, el repositorio que contiene los ejemplos para este microcontrolador fue la base para poder desarrollar el trabajo, especialmente cuando se utilizaron los protocolos de comunicación del giroscopio y el uso de la pantalla LCD.

Como recomendación, se tiene que realizar una investigación previa y lograr el entendimiento completo del microcontrolador antes de empezar a redactar código. Se recomienda también el uso de librerías de “open source” que faciliten el uso del microcontrolador incluyendo funciones ya precodificadas.

Referencias

[1] Discovery kit with STM32F429ZI MCU User Manual. Recuperado de:
<https://docs.rs-online.com/b72f/0900766b815247be.pdf>

[2] Discovery kit with STM32F429ZI MCU Data brief. Recuperado de:
<https://docs.rs-online.com/87ec/0900766b815247bd.pdf>

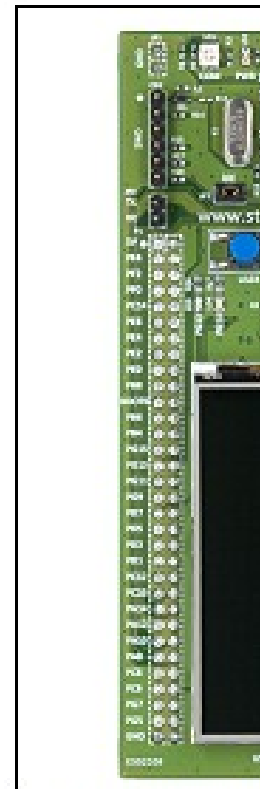
[3] Protoboard 400 Datasheet. Recuperado de:
<http://cdn.sparkfun.com/datasheets/Prototyping/breadboard.pdf>

[4] Giroscopio I3G4250D Datasheet. Recuperado de:
<https://www.st.com/en/mems-and-sensors/i3g4250d.htmlApéndices>

Anexos:

Features

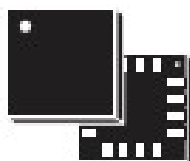
- STM32F429ZIT6 microcontroller featuring 2 Mbytes of Flash memory, 256 Kbytes of RAM in an LQFP144 package
- On-board ST-LINK/V2 on STM32F429I-DISCO or ST-LINK/V2-B on STM32F429I-DISC1
- mbed™-enabled (mbed.org) with ST-LINK/V2-B only
- USB functions:
 - debug port
 - virtual COM port with ST-LINK/V2-B only
 - mass storage with ST-LINK/V2-B only
- Board power supply: through the USB bus or from an external 3 V or 5 V supply voltage
- 2.4" QVGA TFT LCD
- 64-Mbit SDRAM
- L3GD20, ST MEMS motion sensor 3-axis digital output gyroscope
- Six LEDs:
 - LD1 (red/green) for USB communication
 - LD2 (red) for 3.3 V power-on
 - Two user LEDs: LD3 (green), LD4 (red)
 - Two USB OTG LEDs: LD5 (green) VBUS and LD6 (red) OC (over-current)
- Two push-buttons (user and reset)



1. Picture not contractually binding

Description

The STM32F429 Discovery Kit (STM32F429IDISCOVER) is a development board for the STM32F429 microcontroller. It features a 2.4-inch QVGA TFT LCD, a USB port, a push-button, and several LEDs. The board is designed for developing applications with the STM32F429 microcontroller.



LGA-16 (4x4x1 mm)

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

Applications

- Gaming and virtual reality
- Motion control (with digital interface)
- GPS navigation systems
- Appliances and robotics

Description

The L3GD20 is a low-power, low-cost, low-rate sensor.

It includes a sensing element capable of providing the external world through a digital interface (I²C/SPI).

The sensing element is a dedicated micro-machined silicon sensor from STMicroelectronics that integrates the sensing element and actuators on silicon.

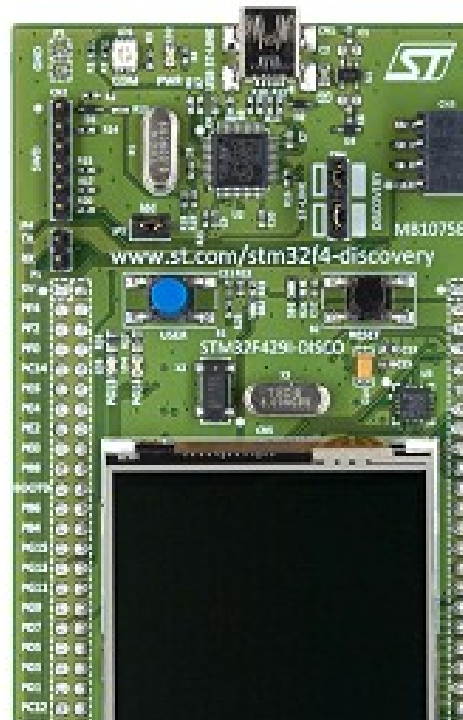
The IC interface is a non-volatile memory process that allows the user to design a dedicated circuit to better match the sensor's characteristics. The L3GD20 has a full-scale range of 250/500/2000 dps and is capable of providing a user-selectable bandwidth.

The L3GD20 is available in two packages:

Introduction

The STM32F429 Discovery kit (32F429IDISCOVERY) helps you evaluate the performance of the STM32F4 series and to develop your applications. It features the STM32F429ZIT6 and includes an ST-LINK/V2 embedded debugger, a 3.5" TFT LCD, SDRAM 64 Mbits, Gyroscope ST MEMS, LEDs, pushbutton, and a USB connector.

Figure 1. STM32F429 Discovery board



**a-Si TFT LCD Single Chip
240RGBx320 Resolution and
Specification**

Getting started with the STM32

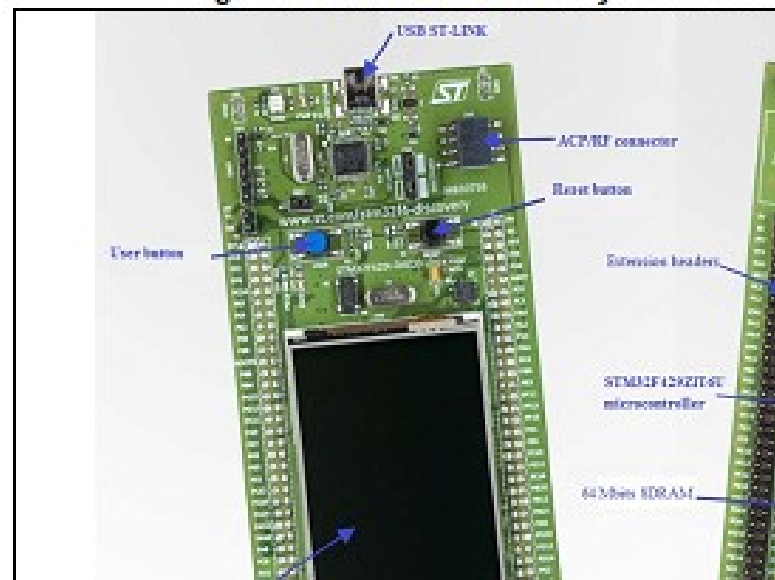
Introduction

This document describes the software, firmware and hardware development recommendations required to build an application on the STM32F429 Discovery kit (32F429DISCOVERY) with demonstration firmware.

The STM32F429 Discovery kit is a low-cost and easy-to-use development board to evaluate and start applications with an STM32F4 32-bit ARM[®] high-performance microcontroller. Before installing and using the kit, please read the Evaluation Product License Agreement from www.st.com/stm32f429discovery.

For more information on the STM32F429 Discovery kit visit www.st.com/stm32f429discovery. To order the STM32F429 Discovery kit, use the STM32F429I-


Figure 1. STM32F429 Discovery board: STM32F429





Features

- **Core:** Arm® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 180 MHz, MPU, 225 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
 - **Memories**
 - Up to 2 MB of Flash memory organized into two banks allowing read-while-write
 - Up to 256+4 KB of SRAM including 64-KB of CCM (core coupled memory) data RAM
 - Flexible external memory controller with up to 32-bit data bus: SRAM, PSRAM, SDRAM/LPDDR SDRAM, Compact Flash/NOR/NAND memories
 - LCD parallel interface, 8080/6800 modes
 - LCD-TFT controller with fully programmable resolution (total width up to 4096 pixels, total height up to 2048 lines and pixel clock up to 83 MHz)
 - Chrom-ART Accelerator™ for enhanced graphic content creation (DMA2D)
 - Clock, reset and supply management
 - 1.7 V to 3.6 V application supply and I/Os
 - POR, PDR, PVD and BOR
 - 4-to-26 MHz crystal oscillator
 - Internal 16 MHz factory-trimmed RC (1% accuracy)
 - 32 kHz oscillator for RTC with calibration
 - Internal 32 kHz RC with calibration
 - Low power
 - Standby: 100 nA
 - Sleep: 10 nA
 - Deep Sleep: 1 nA
 - Deep Sleep with RTC: 10 nA
 - Deep Sleep with RTC and I/Os: 100 nA
 - Deep Sleep with RTC and I/Os and USB: 100 nA
 - Deep Sleep with RTC and I/Os and USB and DMA: 100 nA
 - Deep Sleep with RTC and I/Os and USB and DMA and ART Accelerator: 100 nA
 - Deep Sleep with RTC and I/Os and USB and DMA and ART Accelerator and Chrom-ART Accelerator: 100 nA



LQFP100 (14 × 14 mm)
 LQFP144 (20 × 20 mm)
 LQFP176 (24 × 24 mm)
 LQFP208 (28 × 28 mm)

 - **Debug mode**
 - SWD & JTAG
 - Cortex-M4 Tr
 - Up to 168 I/O pins
 - Up to 164 fast
 - Up to 168 5 V
 - Up to 21 communication
 - Up to 3 × I²C
 - Up to 4 USART
 - ISO7816 inter control)
 - Up to 6 SPIs
 - full-duplex I²S
 - internal audio
 - 1 × SAI (serial
 - 2 × CAN (2.0E
 - **Advanced connection**
 - USB 2.0 full-speed controller with
 - USB 2.0 high-speed device/host/OTG



LQFP100 (14 × 14 mm)
LQFP144 (20 × 20 mm)
LQFP176 (24 × 24 mm)
LQFP208 (28 × 28 mm)