

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE0624 - Laboratorio de Microcontroladores

II ciclo 2024

## **Reporte Laboratorio 5**

Prof. MSc. Marco Villalta Fallas

Paulette Perez Monge B95916

Joselyn Barquero Castillo B88858

20 de noviembre de 2024

# Índice

<b>1. Introducción</b>	<b>1</b>
<b>2. Nota teórica</b>	<b>1</b>
2.1. Periféricos . . . . .	3
2.2. Componentes . . . . .	4
2.3. Lista de componentes y precios . . . . .	5
<b>3. Diseño del circuito</b>	<b>6</b>
3.1. Resultados . . . . .	6
<b>4. Conclusiones</b>	<b>9</b>
<b>A. Anexo a: Hoja de datos del STM32F429</b>	<b>11</b>

*GitHub/Avances* e instrucciones de uso del script se encuentran en el siguiente link: <https://github.com/marpermon/LabMicros>.

## 1. Introducción

El objetivo de este laboratorio es desarrollar un sistema de reconocimiento de actividades humanas que sea capaz de capturar datos del giroscopio, etiquetarlos según el tipo de movimiento realizado y entrenar un modelo de red neuronal utilizando TensorFlow Lite para clasificar diferentes actividades humanas.

El sistema está basado en el microcontrolador STM32F429 Discovery, el cual actúa como el componente principal para la captura de datos en tiempo real desde el giroscopio. Los datos recopilados incluyen los movimientos de brazo hacia arriba o hacia abajo, brazo en círculos y golpe de brazo, y son enviados a una computadora mediante comunicación USB. Se almacenan en un archivo CSV, donde se etiquetan para proporcionar una base estructurada y clara para el entrenamiento del modelo.

Posteriormente, los datos etiquetados se procesan en Edge Impulse, donde se diseña una red neuronal específicamente adaptada para clasificar los movimientos. Una vez entrenado, el modelo se exporta en formato TensorFlow Lite y se implementa en el STM32F429, permitiendo que el sistema realice inferencias en tiempo real. Sin embargo, esto no se completa al 100 %.

## 2. Nota teórica

### STM32F49

El STM32f49 es un microcontrolador de 32 bits con procesador Arm Cortex-M4 y velocidad de 180 MHz diseñado para tener una alta capacidad de operación [1]. En particular, el STM32f429 posee una mayor eficiencia energética, consumiendo menos potencia estática que otros microcontroladores de la misma serie, además de una interfase controladora de LCD [2]. En la figura 1 se puede observar el diagrama de pines del STM32f429.



para ocupar menos memoria y adaptarlos al hardware disponible mediante técnicas como la cuantización de parámetros [4]. Esto asegura que los modelos sean eficientes tanto en términos de tiempo de inferencia como de consumo energético, lo cual es crítico en aplicaciones de sistemas embebidos.

## 2.1. Periféricos

El STM32F429 cuenta con los siguientes periféricos:

- USART / UART
- SPI / I2S
- I2C
- USB OTG (On-The-Go)
- CAN
- Ethernet MAC
- GPIO (General-Purpose Input/Output)
- Timers (16 bits y 32 bits)
- ADC (Analog-to-Digital Converter)
- DAC (Digital-to-Analog Converter)
- Controlador LCD-TFT
- Controlador SDRAM
- Giroscopio L3GD20
- RTC (Real-Time Clock)
- Módulos de generación de reloj
- Modos de bajo consumo (Sleep, Stop, Standby)

Las características eléctricas del microcontrolador aparecen en la tabla 1.

Parámetro	Valor
Voltaje de operación	1.7V a 3.6V
Voltaje de I/O tolerante	Hasta 5V
Fuente de alimentación	USB: 5V Externa: 3V o 5V VBAT: Batería de respaldo
Corriente máxima por pin GPIO	25 mA
Rango de entrada ADC	0V a 3.6V

Tabla 1: Características Eléctricas del STM32F429

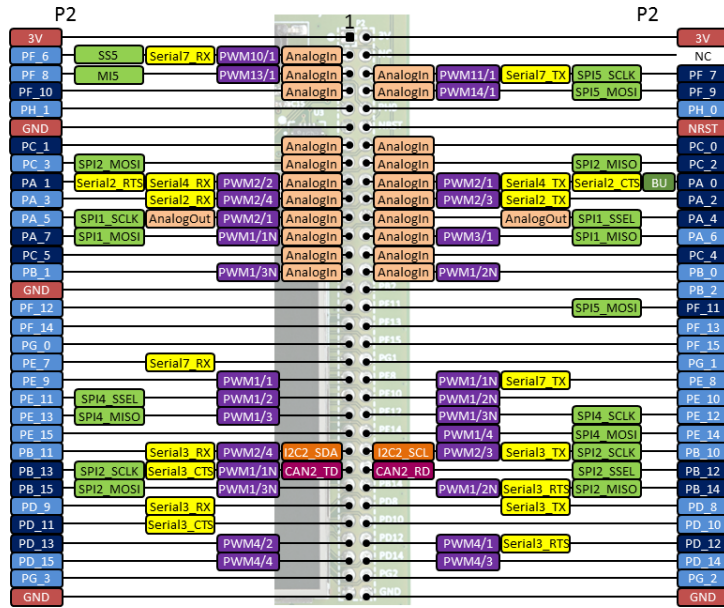


Figura 3: Diagrama de pines a la derecha del STM32

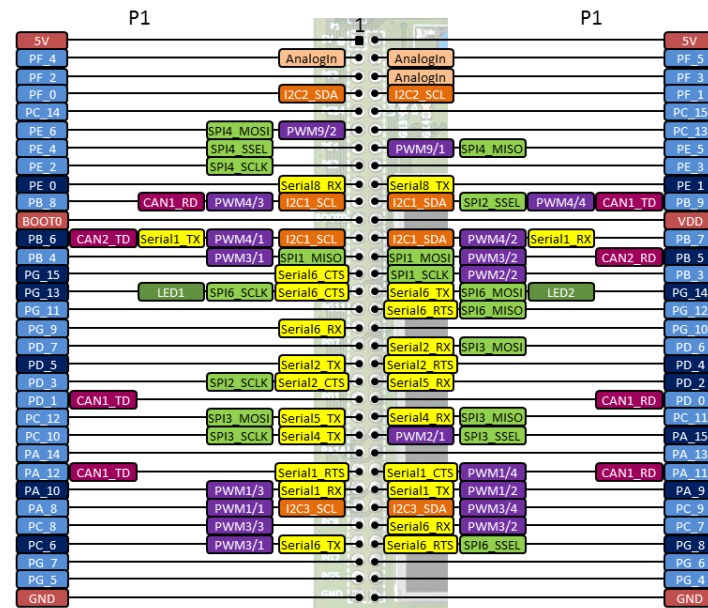


Figura 2: Diagrama de pines a la izquierda del STM32

## 2.2. Componentes

El giroscopio incluido en el STM32F429 Discovery Kit es el L3GD20, un sensor de movimiento de 3 ejes desarrollado por STMicroelectronics. Este sensor mide la velocidad angular en los ejes X, Y, y Z, lo que permite detectar rotaciones y vibraciones. Es ideal para aplicaciones donde se requiere monitorear el movimiento o la orientación del dispositivo en tiempo real.

**2.3. Lista de componentes y precios**

Componente	Precio	Cantidad	Fuente
STM32F429I-DISC1	29,90	1	Link
Total	\$ 29,90		

Tabla 2: Precio de cada componente con sus respectivos enlaces

### 3. Diseño del circuito

En este laboratorio se aprovecharon las capacidades del STM32f4. Se utilizó el SPI5 tanto para leer los valores del giroscopio a la consola usando el protocolo USART y desplegados en esta con el programa minicom.

Se usó el microcontrolador para detectar distintos movimientos con la mano: golpe, arriba-abajo y círculo antihorario.

Se creó un script de python que lograba enviar los datos del puerto serial hasta un archivo csv durante 15 segundos, de los cuales se hicieron uno con cada movimiento. Estos se introdujeron a la plataforma EdgeImpulse, dedicada a crear modelos de redes neuronales para sistemas embebidos, lo que generó librerías de C utilizables.

Los datos capturados del giroscopio se organizaron en un archivo CSV con columnas para las lecturas de los ejes X, Y y Z, junto con una columna de etiquetas. Este archivo fue cargado a Edge Impulse desde el panel de datos, configurando las lecturas como características de entrada y las etiquetas como variables de salida.

En la etapa de procesamiento de datos, se utilizó el bloque de Spectral Analysis. Aquí, no se aplicó ningún filtro específico, y se seleccionó el tipo de análisis como FFT con un tamaño de ventana de 64 muestras. Este paso permitió extraer características relevantes de las señales en el dominio de la frecuencia, optimizando la representación de los datos para el entrenamiento del modelo.

Para la creación del modelo, se utilizó el bloque de clasificación, configurando una red neuronal con una capa de entrada para los datos procesados, dos capas ocultas, y una capa de salida con activación para clasificar entre las distintas actividades. Durante el entrenamiento, se establecieron parámetros específicos para optimizar el desempeño del modelo. Se configuró el número de ciclos en 200 y un batch size de 32, lo que permitió un balance adecuado entre eficiencia y precisión en el entrenamiento.

#### 3.1. Resultados

Se logró desplegar las lecturas del giroscopio tanto en python como en pantalla, como se puede ver en la figura 4. Se decidió usar minicom la menor cantidad de tiempo posible, pues se descubrió que usar éste programa y el script al mismo tiempo se generaban conflictos y lecturas erróneas.

Tras el entrenamiento utilizando Edge Impulse, el modelo logró alcanzar una precisión del 94 % como se puede observar en la figura 5, lo que representa un alto nivel de exactitud en la clasificación de los movimientos. Este resultado es fundamental para garantizar que el sistema pueda diferenciar correctamente entre las actividades humanas definidas.

Después de darle *build* al proyecto el enunciado sugirió utilizar la librería <https://github.com/MarkoSagadin/MicroML?tab=readme-ov-file#Building-your-projects>, sin embargo, esta tarea fue sumamente complicada, debido a la dificultad que supuso implementar, primeramente, las librerías originadas por EdgeImpulse. La carpeta *spi* presente en el repositorio de este laboratorio pretendía reemplazar a la carpeta del mismo nombre en los ejemplos de libopencm3, con el archivo main.pp dentro. Modificando el Makefile, sólo se pudo llegar a éste punto en la compilación:



```

-2 1 -7
-2 1 -7
-2 1 -7
-2 1 -7
-2 1 -7
-2 1 -7
5 0 -4
5 0 -4
5 0 -4
5 0 -4
5 0 -4
5 0 -4
4 5 -5
4 5 -5
4 5 -5
4 5 -5
4 5 -5
4 5 -5
4 5 -5
4 5 -5
-1 6 -6
-1 6 -6
-1 6 -6
-1 6 -6
-1 6 -6
-1 6 -6
1 1 -4
1 1 -4
1 1 -4
1 1 -4
1 1 -4
1 1 -4
5 3 -6
5 3 -6
5 3 -6
5 3 -6
5 3 -6
5 3 -6
4 -1 -6
4 -1 -6
4 -1 -6
4 -1 -6
4 -1 -6
4 -1 -6
4 2 -4
yoojanny: $

```

Figura 4: Valores de los ejes x, y, z del giroscopio impresos en la consola

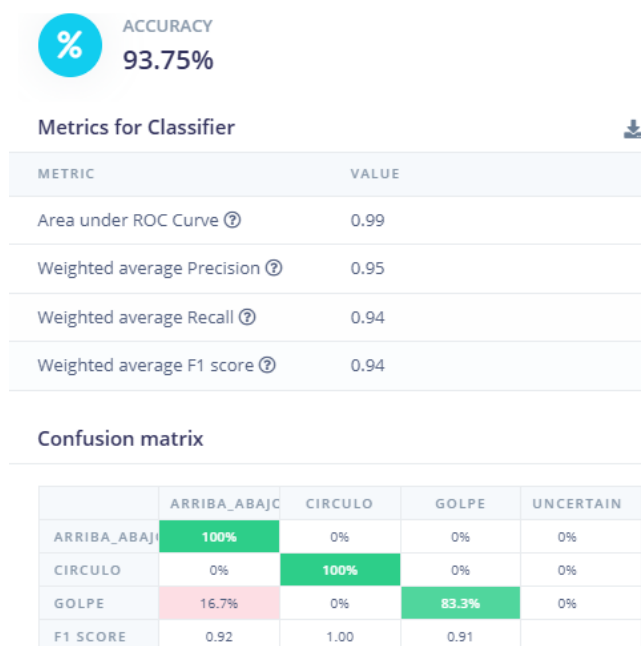


Figura 5: Precisión de la red neuronal

Figura 6: Último punto de la compilación del repositorio

Si desea compilar el código que sólo toma datos del puerto serial, puede mover el makefile que está en raíz (fuera de la carpeta spi) y spi-mems.c eliminando main.cpp, dentro de dicha carpeta en los ejemplos de libopencm3 y utilizar los siguientes comandos:

make

```
arm-none-eabi-objcopy -O binary spi-mems.elf firmware.bin
```

```
st-flash --reset write firmware.bin 0x8000000
```

## 4. Conclusiones

- Se logró cumplir los objetivos del laboratorio de manera parcial.
- EdgeImpulse es una herramienta útil para crear redes neuronales
- La lectura de datos del puerto serial en minicom y python al mismo tiempo puede causar conflictos.
- Las librerías sugeridas son bastante complicadas de implementar en el STM.

## Referencias

- [1] STMicroelectronics. Stm32 32-bit arm cortex mcus. <https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html>, 2024.
- [2] STMicroelectronics. Stm32f429/439. <https://www.st.com/en/microcontrollers-microprocessors/stm32f429-439.html>, 2024.
- [3] Inc. Edge Impulse. Edge impulse, 2024.
- [4] Google. Tensorflow lite, 2024.

# A. Anexo a: Hoja de datos del STM32F429



## STM32F427xx STM32F429xx

32b Arm® Cortex®-M4 MCU+FPU, 225DMIPS, up to 2MB Flash/256+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 20 com. interfaces, camera & LCD-TFT

Datasheet - production data

### 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
    - Sleep, Stop and Standby modes
    - V<sub>BAT</sub> supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
  - 3×12-bit, 2.4 MSPS ADC: up to 24 channels and 7.2 MSPS in triple interleaved mode
  - 2×12-bit D/A converters
  - General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
  - Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 180 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
- LQFP100 (14 × 14 mm) UFBGA176 (10 × 10 mm) WLCSP143  
LQFP144 (20 × 20 mm) UFBGA169 (7 × 7 mm)  
LQFP176 (24 × 24 mm) TFBGA216 (13 × 13 mm)  
LQFP208 (28 × 28 mm)
- Debug mode
    - SWD & JTAG interfaces
    - Cortex-M4 Trace Macrocell™
  - Up to 168 I/O ports with interrupt capability
    - Up to 164 fast I/Os up to 90 MHz
    - Up to 166 5 V-tolerant I/Os
  - Up to 21 communication interfaces
    - Up to 3 × I<sup>2</sup>C interfaces (SMBus/PMBus)
    - Up to 4 USARTs/4 UARTs (11.25 Mbit/s, ISO7816 interface, LIN, IrDA, modem control)
    - Up to 6 SPIs (45 Mbits/s), 2 with muxed full-duplex I<sup>2</sup>S for audio class accuracy via internal audio PLL or external clock
    - 1 × SAI (serial audio interface)
    - 2 × CAN (2.0B Active) and SDIO interface
  - Advanced connectivity
    - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
    - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
    - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
  - 8- to 14-bit parallel camera interface up to 54 Mbytes/s
  - True random number generator
  - CRC calculation unit
  - RTC: subsecond accuracy, hardware calendar
  - 96-bit unique ID