

Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Ingeniería Eléctrica
IE-0624 Laboratorio de Microcontroladores

EIE

Escuela de
Ingeniería Eléctrica

STM32F429 / LibOpenCM3

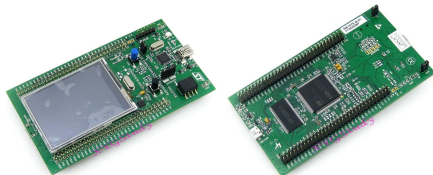
MSc. Marco Villalta Fallas - `marco.villalta@ucr.ac.cr`

II Ciclo 2022

STM32

STM32F429 Discovery kit

Características



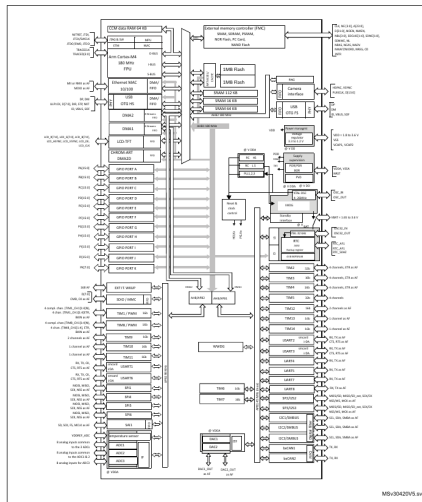
- Microcontrolador STM32F429ZIT6
- 2.4"QVGA TFT LCD
- USB OTG con conector Micro-AB
- Sensor de movimiento I3G4250D, Giroscopio ST MEMS de 3-ejes
- 6 LEDs: LD1 (USB Comms), LD2(3.3V PowerOn, 2 LEDs de usuario (LD3 y LD4), 2 LEDs USB OTG (LD5 y LD6)
- 2 push-button (Usuario y reset)
- 64-Mbit SDRAM
- Header para LQFP144 I/Os
- On-board ST-LINK/V2-B
- Funciones USB: Debug, virtual COM, almacenamiento
- Alimentación por USB o fuente externa de 3V o 5V

Microcontrolador STM32F429

Características

- Core: ARM 32 bits Cortex-M4 con FPU (RISC)
- 180 MHz
- 2MB flash, 256 KB SRAM
- Controlador LCD-TFT
- Low Power
- 3x12bit ADC
- 2x12bit convertidor D/A
- 17 timers: 12 timers de 16bit, 2 de 32bit de hasta 180MHz, c/u con 4IC/OC/PWM
- Debug: SWD, JTAG
- 168 I/O con capacidad de interrupcion
- 21 interfaces de comunicaciones(I2C,USART,SPI,SAI,CAN)
- Conectividad avanzada USB 2.0
- Intefaz de camara
- True RNG
- CRC
- Controladores DMA

Diagrama de bloques



Microcontrolador STM32F429

Características electricas

Table 14. Voltage characteristics

Symbol	Ratings	Min	Max	Unit
$V_{DD}-V_{SS}$	External main supply voltage (including V_{DDA} , V_{DD} and V_{BAT}) ⁽¹⁾	- 0.3	4.0	V
V_{IN}	Input voltage on FT pins ⁽²⁾	$V_{SS} - 0.3$	$V_{DD} + 4.0$	
	Input voltage on TTa pins	$V_{SS} - 0.3$	4.0	
	Input voltage on any other pin	$V_{SS} - 0.3$	4.0	
	Input voltage on BOOT0 pin	V_{SS}	9.0	
$ \Delta V_{DDx} $	Variations between different V_{DD} power pins	-	50	mV
$ V_{SSx}-V_{SS} $	Variations between all the different ground pins including V_{REF} .	-	50	
$V_{ESD(HBM)}$	Electrostatic discharge voltage (human body model)	see Section 6.3.15: <i>Absolute maximum ratings (electrical sensitivity)</i>		

Table 15. Current characteristics

Symbol	Ratings	Max.	Unit
ΣI_{VDD}	Total current into sum of all V_{DD_x} power lines (source) ⁽¹⁾	270	mA
ΣI_{VSS}	Total current out of sum of all V_{SS_x} ground lines (sink) ⁽¹⁾	~ 270	
I_{VDD}	Maximum current into each V_{DD_x} power line (source) ⁽¹⁾	100	
I_{VSS}	Maximum current out of each V_{SS_x} ground line (sink) ⁽¹⁾	~ 100	
I_{IO}	Output current sunk by any I/O and control pin	25	
	Output current sourced by any I/Os and control pin	~ 25	
ΣI_{IO}	Total output current sunk by sum of all I/O and control pins ⁽²⁾	120	
	Total output current sourced by sum of all I/Os and control pins ⁽²⁾	~ 120	
$I_{INJ(PIN)}^{(3)}$	Injected current on FT pins ⁽⁴⁾	~ 5/+0	
	Injected current on NRST and BOOT0 pins ⁽⁴⁾		
	Injected current on TTA pins ⁽⁵⁾	±5	
$\Sigma I_{INJ(PIN)}^{(5)}$	Total injected current (sum of all I/O and control pins) ⁽⁶⁾	±25	

LibOpenCM3

Toolchains/Bibliotecas para MCU STM32

- IAR EWARM (IAR Embedded Workbench)
- Keil MDK-ARM TM
- ARM mbed TM online
- GCC-based IDEs (free AC6: SW4STM32, Atollic TrueSTUDIO ,...)
- LibOpenCM3 (Basado GCC)

LibOpenCM3

- Proyecto de SL
- Biblioteca de bajo nivel para microcontroladores ARM Cortex-M3
- Es necesario instalar un toolchain de ARM para compilar la biblioteca y ejemplos
- Se programa en C utilizando funciones.
- Se recomienda <https://launchpad.net/gcc-arm-embedded>
- Biblioteca en <https://github.com/libopencm3/libopencm3/wiki>
- Ejemplos en <https://github.com/libopencm3/libopencm3-examples>
- LibOpenCM3-plus en <https://github.com/arcoslab/libopencm3-plus>

LibOpenCM3

ST-Link

- ST-Link es un programa para subir/programar los microcontroladores STM32.
- Instalación en Linux puede ser desde código fuente o desde repositorio.
- Instalación desde repositorio para distribuciones basadas en Debian:

```
sudo apt install stlink-tools gcc-arm-none-eabi
```

LibOpenCM3

Hola mundo: Configuración

```
#include <libopencm3/stm32/rcc.h>
#include <libopencm3/stm32/gpio.h>

/* Set STM32 to 168 MHz. */
static void clock_setup(void)
{
    rcc_clock_setup_pll(&rcc_hse_8mhz_3v3(RCC_CLOCK_3V3_168MHZ));
    rcc_periph_clock_enable(RCC_GPIOG);    /* Enable GPIOG clock. */
}

static void gpio_setup(void)
{
    /* Set GPIO13-14 (in GPIO port G) to 'output push-pull'. */
    gpio_mode_setup(GPIOG, GPIO_MODE_OUTPUT, GPIO_PUPD_NONE, GPIO13 | GPIO14);
}
```

LibOpenCM3

Hola mundo: Función principal

```
int main(void)
{
    int i;

    clock_setup();
    gpio_setup();

    gpio_set(GPIOG, GPIO13);          /* Set two LEDs for wigwag effect when toggling. */

    /* Blink the LEDs (PG13 and PG14) on the board. */
    while (1) {
        /* Toggle LEDs. */
        gpio_toggle(GPIOG, GPIO13 | GPIO14);
        for (i = 0; i < 6000000; i++) { /* Wait a bit. */
            __asm__("nop");
        }
    }
    return 0;
}
```

LibOpenCM3

Ejemplos de la biblioteca

Revisar ejemplos para stm32/f4/stm32f429i-discovery

- adc-dac
- Manejo de botones
- Blink
- LCD
- RNG
- SPI
- USART
- USB