

Universidad Nacional Autónoma de México Facultad de Ingeniería



MICROPROCESADORES Y MICROCONTROLADORES

Tema 6 Puertos de entrada/salida

M. I. Christo Aldair Lara Tenorio 2025-1

Tabla de contenidos

Objetivo del tema

Proyecto 3 – Microcontrolador de 8 bits

Unidades funcionales de un microcontrolador

Arquitectura de un microprocesador ARM Cortex-M

Puertos de entrada/salida

Diagrama de bloques del TIVA TM4C1294NCPDT

Tarjeta de desarrollo EK-TM4C1294NCPDT

Manual de usuario (Datasheet)

Configuración de los GPIO del TIVA TM4C1294NCPDT

Interruptores de entrada

Tarea 5 – Ejercicios de entrada/salida con GPIO

Fuentes de reloj del TIVA TM4C1294NCPDT

Temporizador del sistema (SysTick)



Objetivo del tema

Objetivo general:

El alumno programará los dispositivos de entrada y salida.

Contenido:

- 6.1. Conceptos básicos de entrada/salida.
- 6.2. Puertos paralelos de entrada/salida.
- 6.3. Programación de puertos paralelos de entrada/salida.

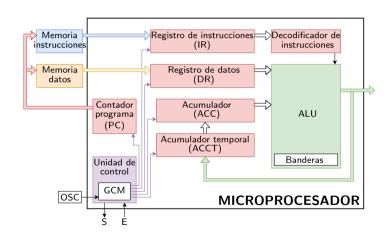




Proyecto 3 – Microcontrolador de 8 bits

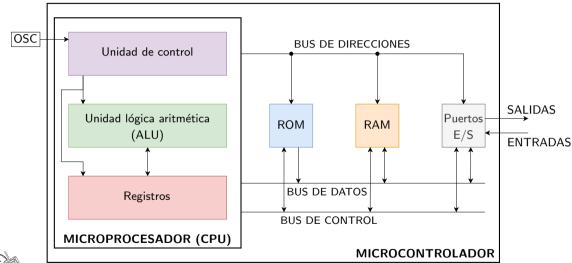
Agregar las siguientes características a la arquitectura de procesador diseñado para el proyecto 2:

- Dos puertos de entrada/salida de 8 bits y las instrucciones para utilizarlos.
- Instrucción de resta aritmética (ACC - DR).
- Instrucción de reset (cargar al PC la dirección 0).
- Registro de estados con las cuatro banderas (N, Z, C y V), agregando la capacidad de actualizarlo o no al ejecutar una instrucción (similar al sufijo S en ensamblador).





Unidades funcionales de un microcontrolador

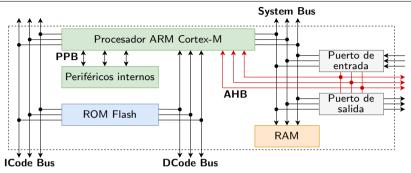




4 / 35

M. I. Christo Aldair Lara Tenorio

Arquitectura de un microprocesador ARM Cortex-M



Tareas que se pueden ejecutar en paralelo

- ICode Bus: Extraer código de la ROM.
- DCode Bus: Leer datos constantes de la ROM.
- System Bus: Leer/Escribir datos de la RAM o de los puertos de E/S y extraer código de la RAM.
- PPB (Private Peripheral Bus): Leer/Escribir datos de los periféricos internos.
- 🗽 AHB (Advanced High-Performance Bus): Leer/Escribir datos de los puertos de E/S con una alta velocidad.

Puertos de entrada/salida

Interfaces físicas que permiten la comunicación/interacción entre el microcontrolador y dispositivos externos.

- Reciben señales externas (botones, interruptores, sensores, etc.) que pueden utilizar para realizar acciones o cálculos (datos).
- Envía señales hacía dispositivos externos (LED, displays, actuadores, motores, etc.) para controlar su operación.
- Pueden configurarse para funcionar de diferentes formas:
 - Puertos de entrada/salida de propósito general (GPIO).
 - Comunicación serial (UART, I2C, SPI, CAN, USB, etc.).
 - Convertidores analógico digital (ADC) y digital analógico (DAC).
- Un mismo pin del microcontrolador puede tener diferentes funciones de entrada/salida.
- Sus funciones se configuran a través de registros.
- Tienen un límite de corriente que pueden suministrar o recibir (alrededor de algunos mA).



Diagrama de bloques del TIVA TM4C1294NCPDT

Periféricos de sistema

Relacionados con la operación interna del microcontrolador y su funcionamiento en general.

Periféricos seriales

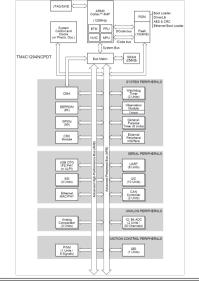
Permiten la comunicación con otros dispositivos a través de una interfaz serial (se requieren de varios pines trabajando en conjunto).

Periféricos analógicos

Permiten la interacción con señales analógicas.

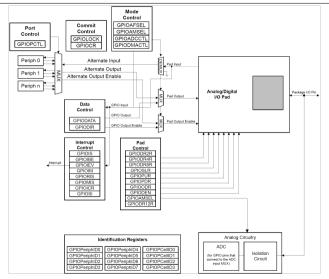
Periféricos de control de movimiento

Permiten controlar el movimiento de dispositivos mecánicos externos.





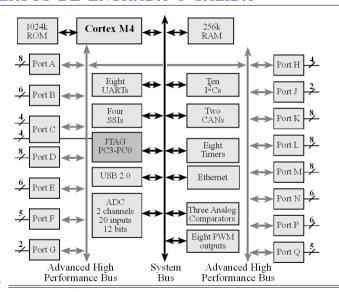
Puertos de entrada y salida digitales y analógicos





2025-1

Puertos de entrada y salida

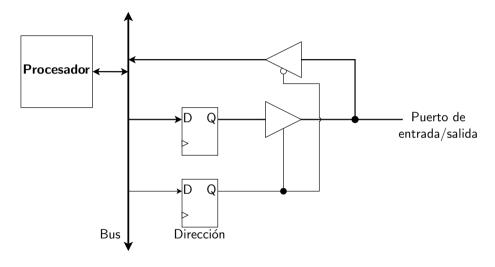


Puertos GPIO

- PortA[7..0]
- PortB[5..0]
- PortC[7..0]
- PortD[7..0]
- PortE[5..0]
- PortF[4..0]
- PortG[1..0]PortH[3..0]
- PortJ[1..0]
- PortJ[1..0
- PortK[7..0]
- PortL[7..0]
- PortM[7..0]
- PortN[5..0]
- PortP[5..0]
- PortQ[4..0]

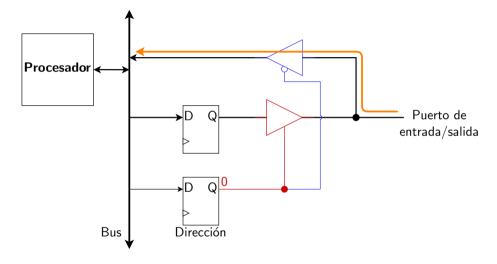


CIRCUITO DE LOS PUERTOS DE ENTRADA/SALIDA



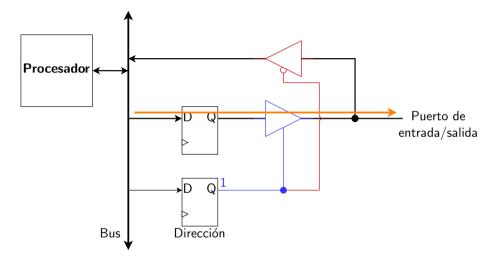


CIRCUITO DE LOS PUERTOS DE ENTRADA/SALIDA (ENTRADA)



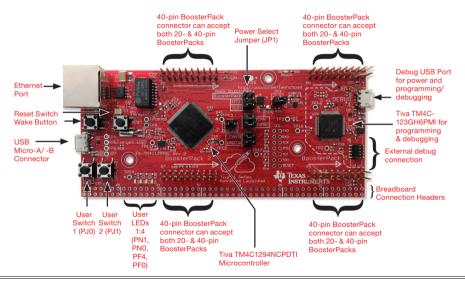


CIRCUITO DE LOS PUERTOS DE ENTRADA/SALIDA (SALIDA)





Tarjeta de desarrollo EK-TM4C1294NCPDT

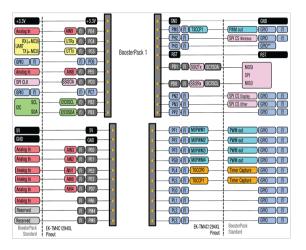


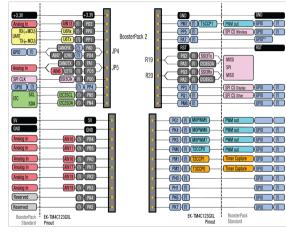


13 / 35

M. I. Christo Aldair Lara Tenorio

BOOSTER PACKS

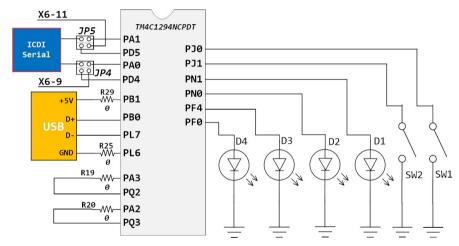






LED Y BOTONES DE LA TARJETA DE DESARROLLO

EK-TM4C1294GXL LaunchPad





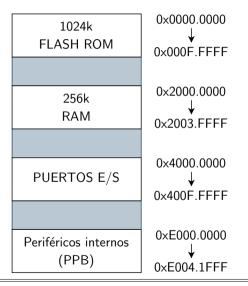
Manual de usuario (Datasheet)

Manual de usuario que desarrolla el fabricante del microcontrolador, en donde documenta toda la información técnica necesaria para comprender el uso el microcontrolador.

- Es esencial para que el diseñador utilice el dispositivo.
- Describe la arquitectura del microcontrolador y sus características.
- Describe los bloques funcionales y los periféricos del microcontrolador.
- Describe el proceso de configuración de los periféricos del microcontrolador y todas sus funciones.
- Contiene las especificaciones eléctricas del microcontrolador.
- Detalla los modos de operación del microcontrolador.
- Presenta ejemplos sobre cómo utilizar algunas funciones del microcontrolador.



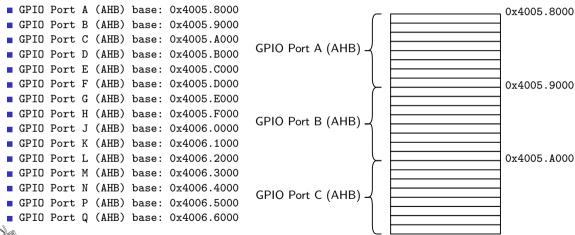
Mapa de memoria del TM4C1294





Direcciones de memoria de los puertos de entrada/salida

La programación de un microcontrolador consiste en almacenar información en registros de memoria.



M. I. Christo Aldair Lara Tenorio

Mapa de registros de los GPIO

Table 10-7. GPIO Register Map

Offset	Name	Туре	Reset	Description	See page
0x000	GPIODATA	RW	0x0000.0000	GPIO Data	759
0x400	GPIODIR	RW	0x0000.0000	GPIO Direction	760
0x404	GPIOIS	RW	0x0000.0000	GPIO Interrupt Sense	761
0x408	GPIOIBE	RW	0x0000.0000	GPIO Interrupt Both Edges	762
0x40C	GPIOIEV	RW	0x0000.0000	GPIO Interrupt Event	763
0x410	GPIOIM	RW	0x0000.0000	GPIO Interrupt Mask	764
0x414	GPIORIS	RO	0x0000.0000	GPIO Raw Interrupt Status	765
0x418	GPIOMIS	RO	0x0000.0000	GPIO Masked Interrupt Status	767
0x41C	GPIOICR	W1C	0x0000.0000	GPIO Interrupt Clear	769
0x420	GPIOAFSEL	RW		GPIO Alternate Function Select	770
0x500	GPIODR2R	RW	0x0000.00FF	GPIO 2-mA Drive Select	772
0x504	GPIODR4R	RW	0x0000.0000	GPIO 4-mA Drive Select	773
0x508	GPIODR8R	RW	0x0000.0000	GPIO 8-mA Drive Select	774
0x50C	GPIOODR	RW	0x0000.0000	GPIO Open Drain Select	775
0x510	GPIOPUR	RW		GPIO Pull-Up Select	776
0x514	GPIOPDR	RW	0x0000.0000	GPIO Pull-Down Select	778
0x518	GPIOSLR	RW	0x0000.0000	GPIO Slew Rate Control Select	780
0x51C	GPIODEN	RW		GPIO Digital Enable	781
0x520	GPIOLOCK	RW	0x0000.0001	GPIO Lock	783
0x524	GPIOCR	-		GPIO Commit	784
00028	GPIOAMSEL	RW	0x0000.0000	GPIO Analog Mode Select	786

GPIO Port A (AHB) 0x4005.8000

GPIODATA	0x4005.8000
GPIODIR	0x4005.8 <mark>400</mark>
GPIOIS	0x4005.8404
GPIOIBE	0x4005.8408
GPIOIEV	0x4005.840C
GPIOIM	0x4005.8 <mark>410</mark>
GPIORIS	0x4005.8414
GPIOMIS	0x4005.8418
GPIOICR	0x4005.841C
GPIOAFSEL	0x4005.8420
GPIODR2R	0x4005.8500
GPIODR4R	0x4005.8504
GPIODR8R	0x4005.8 <mark>508</mark>
GPIOODR	0x4005.850C
GPIOPUR	0x4005.8 <mark>510</mark>
GPIOPDR	0x4005.8 <mark>514</mark>
GPIOSLR	0x4005.8518
GPIODEN	0x4005.851C
GPIOLOCK	0x4005.8 <mark>520</mark>
GPIOCR	0x4005.8 <mark>524</mark>
GPIOAMSEL	0x4005.8 <mark>528</mark>

MÁSCARA DEL REGISTRO GPIODATA

El offset del registro GPIODATA depende de los bits/pines del puerto en donde se requiere escribir/leer información.

Figure 10-3. GPIODATA Write Example

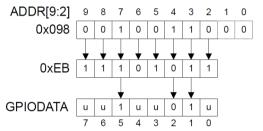
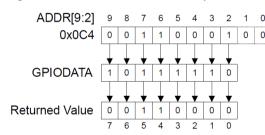


Figure 10-4. GPIODATA Read Example



Por ejemplo, para el puerto A: Dirección base: 0x4005.8000

Pines a de	senmascarar	Dirección de GPIODATA
7, 6, 5,	4	0x4005.83C0
	3, 2, 1, 0	0x4005.8 <mark>03C</mark>
7, 6, 5,	4, 3, 2, 1, 0	0x4005.83FC
	3, 2, 1, 0	0x4005.803C



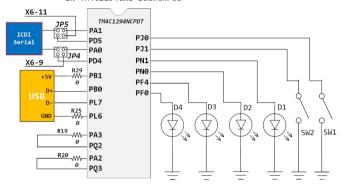
Inicialización y configuración de los GPIO

- 1. Habilitar el reloj del puerto en el registro RCGCGPIO (pp. 382), esperando dos ciclos de instrucción para que se estabilice el reloj.
- 2. Configurar la dirección del GPIO en el registro GPIODIR (pp. 760).
- 3. (OPCIONAL) Configurar la función alterna del GPIO en el registro GPIOAFSEL (pp. 770).
- 4. (OPCIONAL) Configurar los modos de controlador extendidos en el registro GPIOPC (pp. 800).
- (OPCIONAL) Configurar la corriente de entrada/salida del GPIO en los registros GPIODR4R, GPIODR8R y GPIODR12R (pp. 773, 774 y 792) (la corriente máxima con la que operan los GPIO del TIVA TM4C12N4NCPDT es de 12mA).
- 6. (OPCIONAL) Habilitar las resistencias de pull-up (GPIOPUR) o de pull-down (GPIOPDR), o configurar como drenaje abierto (GPIOODR) (pp. 776, 778 y 775).
- 7. Configurar las funciones digitales en el registro GPIODEN (pp. 781).
- 8. (OPCIONAL) Configurar los campos de interrupción en los registros GPIOIS, GPIOIBE, GPIOIEV y GPIOIM (pp. 761, 762, 763 y 764).
 - O. (OPCIONAL) Configurar el bloqueo de puertos en el registro GPIOLOCK (pp. 783).



Ejemplo: Encender los LED de la tarjeta de desarrollo

FK-TM4C1294GXI LaunchPad

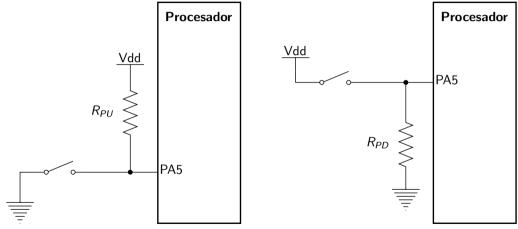


- Habilitar el reloj del puerto N (RCGCGPIO).
- 2. Esperar dos ciclos de instrucción para que se estabilice el reloj.
- Configurar a PortN[1,0] como salidas (GPIODIR).
- Habilitar las funciones digitales de PortN[1,0] (GPIODEN).
- Definición del offset de GPIODATA.
- 6. Encender los LED con GPIODATA.



Interfaz para interruptores de entrada

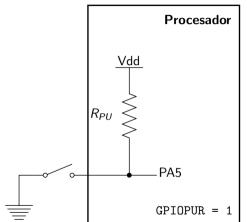
Utilizando resistencias de Pull-up y Pull-down externas:

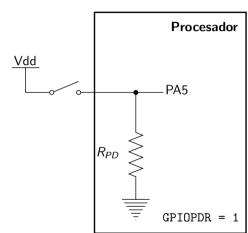




Interfaz para interruptores de entrada

Habilitando las resistencias de Pull-up y Pull-down embebidas en el microcontrolador:







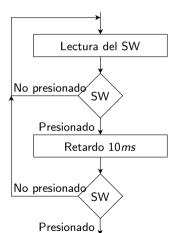
EFECTO REBOTE (BOUNCE)

Pequeñas fluctuaciones que suceden al presionar o soltar un botón, causadas por el contacto mecánico no ideal del botón



Para solucionarlo, se genera un retardo de aproximadamente 10*ms* (dependiendo del botón) al detectar que se presiona el SW, para posteriormente volver a leer el SW y determinar si el cambio de estado es valido.

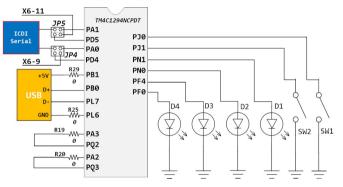
$$\left(\frac{1}{16M}\right)\left(2+3n\right)=10$$
 ms





Ejemplo: Lectura del SW1 de la tarjeta de desarrollo

EK-TM4C1294GXL LaunchPad



- 1. Habilitar el reloj del puerto J (RCGCGPIO).
- 2. Esperar dos ciclos de instrucción para que se estabilice el reloj.
- 3. Configurar a PortJ[0] como entradas (GPIODIR).
- Habilitar las resistencias de Pull-up de PortJ[0] (GPIOPUR).
- Habilitar las funciones digitales de PortJ[0] (GPIODEN).
- Definición del offset de GPIODATA.
- 7. Leer el valor del SW1 con GPIODATA.



Tarea 5 – Ejercicios de entrada/salida con GPIO

Desarrollar los códigos en ensamblador que realicen lo siguiente:

C01: Encender y apagar los LED de usuario de la tarjeta de desarrollo en el siguiente orden, con una frecuencia de encendido/apagado de 1*Hz*, repitiendo el proceso 5 veces.

1. LED1 $\rightarrow ON$

3. LED3 $\rightarrow ON$

5. LED1 $\rightarrow OFF$

7. LED3 $\rightarrow OFF$

2. LED2 $\rightarrow ON$

4. LED4 $\rightarrow ON$

6. LED2 $\rightarrow OFF$

8. LED4 \rightarrow OFF

- C02: Encender y apagar los LED de usuario de la tarjeta de desarrollo, dependiendo de los interruptores que se presionen.
 - Si se presiona el SW1, conmutar el LED1.
 - Si se presiona el SW2, conmutar el LED2.
 - * Considerar un retardo de rebote de aproximadamente 10 ms.



Fuentes de reloj del TIVA TM4C1294NCPDT

Los microcontroladores cuentan con un componente/circuito que genera las señales de reloj que sincronizan el funcionamiento del sistema y todos sus procesos internos (periféricos).

- La velocidad de ejecución de un microcontrolador se determina por un cristal externo.
- Entre mayor sea la frecuencia de oscilación del reloj, el microcontrolador podrá ejecutar un mayor número de instrucciones por segundo (mayor velocidad), pero tendrá un mayor consumo energético y generará una cantidad mayor de calor.
- La tarjeta EK-TM4C1294XL tiene:
 - Un cristal externo de 25MHz.
 - Un cristal de baja frecuencia (32.768KHz) para el modo de hibernación del TIVA TM4C1294NCPDT.
- La mayoría de los microcontroladores cuentan con un lazo de seguimiento de fase (PLL, por sus siglas en inglés) que permite ajustar la velocidad de ejecución a través de software.
- El TIVA TM4C1294NCPDT tiene un oscilador interno de precisión (PIOSC), que trabaja a una velocidad pre-configurada de $16MHz\pm1\%$.



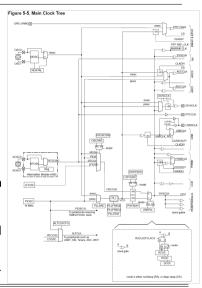
ÁRBOL DE RELOJ DEL TIVA TM4C1294NCPDT

Fuentes de reloj fundamentales

- Oscilador interno de precisión (PIOSC = 16MHz)
 Oscilador integrado en el microcontrolador que proporciona la velocidad de ejecución inicial, con una alta precisión.
- Oscilador principal (MOSC = 25MHz)
 Oscilador principal del sistema.
- Oscilador interno de baja frecuencia (LFIOSC = 33KHz)
 Oscilador utilizado en funciones de bajo consumo energético o para tareas que no requieren de una alta precisión en tiempo.
- Fuente de reloj del oscilador RTC del módulo de hibernación (RTCOSC = 32.768 KHz)

 Mantiene la cuenta mientras el sistema se encuentra en modo de bajo consumo de energía (hibernación), manteniendo "dormido" al microcontrolador temporalmente.

Tiva TM4C1294NCPDT Microcontroller DATA SHEET (pp. 233).





Ciclos de reloj consumidos en ensamblador

Cada instrucción en lenguaje ensamblador consume un número distinto de ciclos de reloj, dependiendo de la instrucción, el estado del pipeline y el modo de direccionamiento.

		Table 3-1 Cortex-M4 instruct	ion set summary
Operation	Description	Assembler	Cycles
Move	Register	MOV Rd, <op2></op2>	1
	16-bit immediate	MOVW Rd, # <imm></imm>	1
	Immediate into top	MOVT Rd, # <imm></imm>	1
	To PC	MOV PC, Rm	1 + P
Add	Add	ADD Rd, Rn, <op2></op2>	1
	Add to PC	ADD PC, PC, Rm	1 + P
	Add with carry	ADC Rd, Rn, <op2></op2>	1
	Form address	ADR Rd, <label></label>	1

Operation	Description	Assembler	Cycles
Subtract	Subtract	SUB Rd, Rn, <op2></op2>	1
	Subtract with borrow	SBC Rd, Rn, <op2></op2>	1
	Reverse	RSB Rd, Rn, <op2></op2>	1
Multiply	Multiply	MUL Rd, Rn, Rm	1
	Multiply accumulate	MLA Rd, Rn, Rm	2
	Multiply subtract	MLS Rd, Rn, Rm	2
	Long signed	SMULL RdLo, RdHi, Rn, Rm	1
	Long unsigned	UMULL RdLo, RdHi, Rn, Rm	1
	Long signed accumulate	SMLAL RdLo, RdHi, Rn, Rm	1
	Long unsigned accumulate	UMLAL RdLo, RdHi, Rn, Rm	1

Cortex-M4 Technical Reference Manual (DDI0439B).



TEMPORIZADOR DEL SISTEMA (SYSTICK)

Contador presente en los procesadores ARM Cortex-M que proporciona una forma simple y eficiente de medir el tiempo y generar interrupciones periódicas.

- Temporizador descendente de 24 bits (comienza desde un valor establecido de máximo 24 bits y termina cuando la cuenta llega a cero).
- Utiliza dos fuentes de reloj:
 - Oscilador interno de precisión (PIOSC) dividido entre 4 (4MHz, considerando la velocidad pre-configurada).
 - Reloj del sistema.
- Modo multi-shot, en donde al terminar la cuenta (llegar a 0), este comienza de nuevo desde el valor configurado inicialmente.
- El SysTick consiste en tres registros:
 - SysTick Control and Status (STCTRL) → Habilitación y configuración de los parámetros del SysTick.
 - SysTick Reload Value (STRELOAD) → Valor de carga/inicio del SysTick.
 - SysTick Current Value (STCURRENT) \rightarrow Valor presente en el contador.

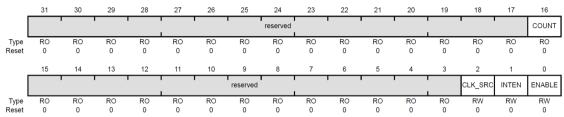


SysTick Control and Status Register (STCTRL)

SysTick Control and Status Register (STCTRL)

Base 0xE000.E000

Type RW, reset 0x0000.0000



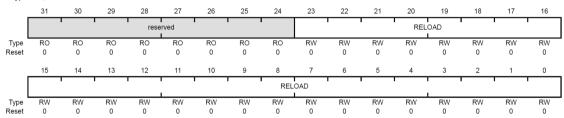
- COUNT → Bandera que indica que el SysTick terminó su cuenta.
- CLK_SRC → Selección de la fuente de reloj.
- INTEN → Habilitación de la interrupción.
- ENABLE → Habilitación del SysTick.



SysTick Reload Value Register (STRELOAD)

SysTick Reload Value Register (STRELOAD)

Base 0xE000.E000 Offset 0x014 Type RW, reset -



■ RELOAD → Valor de carga de 24 bits del SysTick.

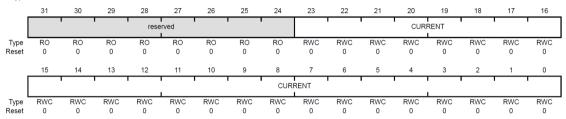


33 / 35

SysTick Current Value (STCURRENT)

SysTick Current Value Register (STCURRENT)

Base 0xE000.E000 Offset 0x018 Type RWC, reset -



ullet CURRENT o Valor de 24 bits presente en el contador.



M. I. Christo Aldair Lara Tenorio

Proceso de inicialización del SysTick (código 13)

- 1. Programar el valor de cuenta en el registro STRELOAD.
- 2. Limpiar el registro STCURRENT al escribir un valor cualquiera.
- 3. Configurar el registro STCTRL para la configuración requerida.

Consideraciones importantes:

- Cuando está habilitado, el SysTick contará hacia abajo en cada ciclo de reloj hasta llegar a cero. Cuando llegue a cero, el bit COUNT del registro STCTRL se colocará en '1'. Después, se volverá a cargar el valor de cuenta y el SysTick comenzará de nuevo la cuenta descendente (cuenta periódica).
- Limpiar el registro STRELOAD con cualquier valor, deshabilitará el SysTick en el siguiente ciclo.
- Escribir en el registro STCURRENT limpiará el registro y el bit COUNT.
- Leer el registro STCTRL limpiará la bandera COUNT.
- Si el procesador está en modo de debug, el SysTick no contará.



35 / 35