

# Agenda

- Introducción
- Presentación de la Familia Cortex M3
- Arquitectura de los Cortex M3
- Repertorio de Instrucciones y Ejemplos
- Sistema de Memoria
- Excepciones, Interrupciones y el NVIC
- **La familia NXP LPC43xx**

Cortex LPC17xx

1

# Resumen de lo visto hasta ahora

El diseño de sistemas embebidos solo requiere respetar un conjunto de reglas bastante obvias y sencillas



Debemos contar con las herramientas de hardware y software adecuadas para cada aplicación

Diseño de Sistemas en Firmware - Cap 1

# Hoy



3

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Características Principales

- MPU: Permite la protección de zonas de memoria de la acción indeseada de programas
- NVIC: permite la generaci3n y priorizaci3n de interrupciones y minimiza la latencia de las interrupciones.
- WIC: permite la r3pida entrada en ejecuci3n de programa luego de un estado de bajo consumo.
- Acelerador de flash: Que permite ejecutar el programa sin wait states (ver Cap 3) superando las limitaciones tecnol3gicas de las memorias Flash.
- DMA: Muchos dispositivos pueden acceder a realizar trasnferencias entre s3 o con memoria.

Cortex LPC17xx

4

Compatibilidad

Se ha buscado facilitar la transici3n de desarrollos realizados con micros De la Familia ARM2300 haciendo que los componentes Cortex 176x sean Compatibles pin a pin con aquellos



Cortex LPC17xx

5

LPC4337

LPC4337JET256	LBGA256	Plastic low profile ball grid array package, 256 balls; body 17 × 17 × 1 mm	SOT740-2
LPC4337JET256	LBGA256	Plastic low profile ball grid array package, 256 balls; body 17 × 17 × 1 mm	SOT740-2
LPC4337JBD144	LQFP144	Plastic low profile quad flat package, 144 leads; body 20 × 20 × 1.4 mm	SOT486-1
LPC4337JET100	TFBGA100	Plastic thin fine-pitch ball grid array package, 100 balls; body 9 × 9 × 0.7 mm	SOT926-1

Type number	Flash total	Flash bank A	Flash bank B	Total SRAM	LCD	Ethernet	USB0 (Host, Device, OTG)	USB1 (Host, Device)/ULP interface	PWM	QEI	ADC channels	Temperature range	GPIO
LPC4337JET256	1 MB	512 kB	512 kB	136 kB	no	yes	yes	yes/yes	yes	yes	8	J	164
LPC4337JBD144	1 MB	512 kB	512 kB	136 kB	no	yes	yes	yes/no	yes	no	8	J	83
LPC4337JET100	1 MB	512 kB	512 kB	136 kB	no	yes	yes	yes/no	no	no	4	J	49

6

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

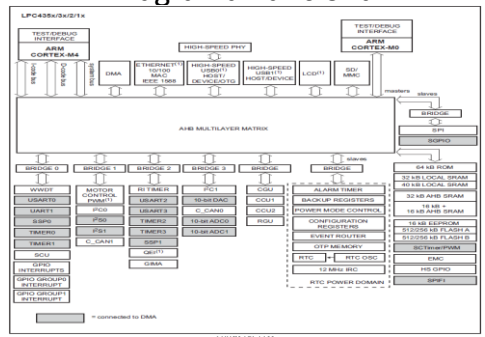
---

---

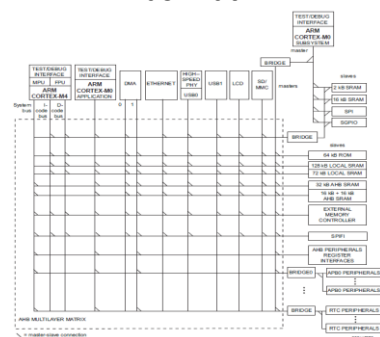
---

---

### Diagrama Funcional



## Bus Matrix



## Manejo de Memoria

Cortex LPC17xx

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

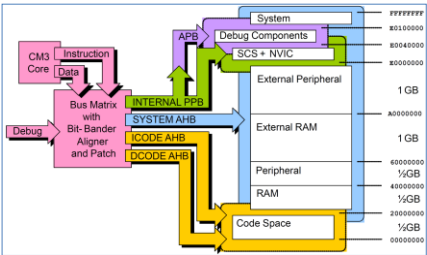
---

---

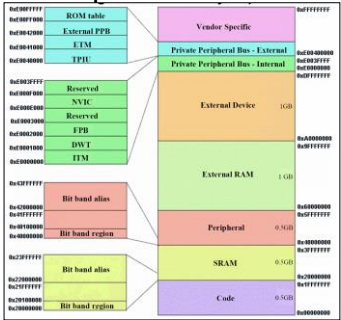
---

Mapa de Memoria

- Muy sencillo mapa de memoria lineal de 4GB
- Las particiones de la BUS matrix acceden a memoria mediante los buses AHB y PPB

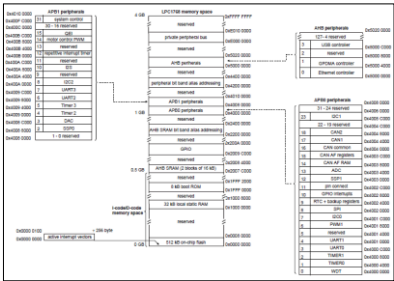


Mapa de Memoria



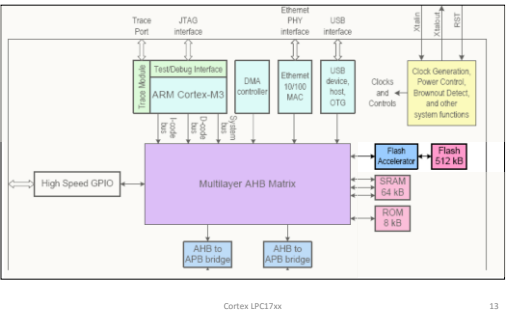
Cortex-LPC17xx

Mapa de memoria



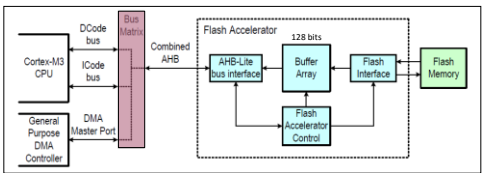
Cortex-LPC17xx

Acelerador de memoria Flash



Cortex-LPC17xx 13

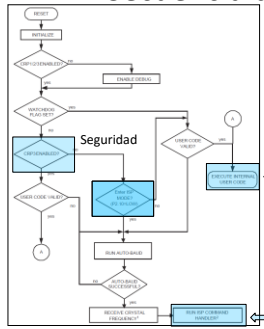
Acelerador de memoria Flash



- Permite ejecutar programas desde flash a máxima velocidad
- Hasta 100MHz sin introducir Wait States
- Provee interfaces separadas a los buses I-code y D-code
- Optimiza la interacción con los buffers de pre-búsqueda del Cortex-M4.
- Operación transparente para el usuario. Se programa a través del registro FLASHCFG con el que se puede graduar la aceleración.

Cortex-LPC17xx 14

Secuencia de arranque



- ISP = In system Programming. Se carga todo el programa desde la PC
- IAP = In Application Programming. Se modifica la memoria flash en tiempo de ejecución.

Ejecución Normal

In System Programming

Cortex-LPC17xx 15

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

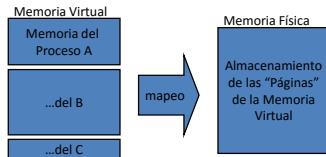
---

---

---

## Memoria Virtual y MMU

- Con sistemas operativos (tipo Linux) se pueden correr varios *procesos* "simultáneamente".



- El mapeo lo hace una combinación de HW y SW
  - Al bloque de HW que se ocupa de eso se le dice *Unidad de Administración de Memoria (MMU)*
- La memoria virtual tiene contras similares a las cache

16 de 22

## Otro camino. Evolución ARM7-Cortex

FEATURES	ARM7TDMI	ARM Cortex M3
Architecture	ARMv4T (von Neumann)	ARMv7-M (Harvard)
ISA Support	Thumb <sup>®</sup> /ARM	Thumb-2 <sup>®</sup>
Pipeline	5-stage	5-stage + branch prediction
Interrupts	FIQ/IRQ	NMI + 1 to 240 physical interrupts
Inter-Interrupt Latency	24 ~ 42 cycles	12 cycles
Sleep Modes	None	Idle/Deep Sleep
Memory Protection	None	8 region MPU
Die/stone	0.95 DMIPS/MHz (ARM) 0.74 DMIPS/MHz (Thumb <sup>®</sup> )	1.25 DMIPS/MHz
Power Consumption	0.28 mW/MHz	0.19 mW/MHz

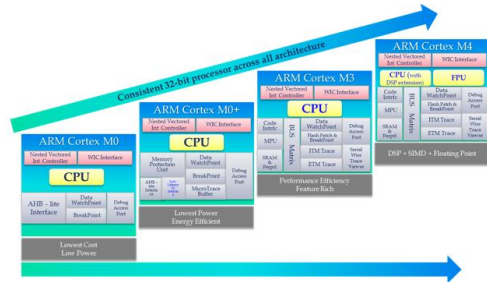
17

El Cortex M4 aparece desplazando al Cortex M3 en nuevos diseños.

- ¿En qué se asemejan?
- ¿En qué se diferencian?
- ¿En qué cambiarán nuestros hábitos de desarrollo?
- ¿Para qué aplicaciones es recomendable el reemplazo?
- ¿Puede un Cortex M4 reemplazar a un DSP?

18

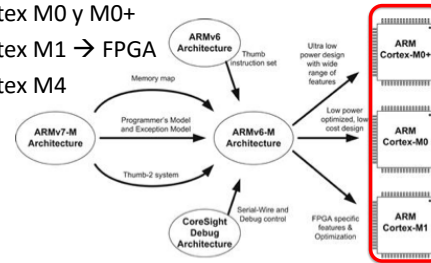
## La familia Cortex M



19

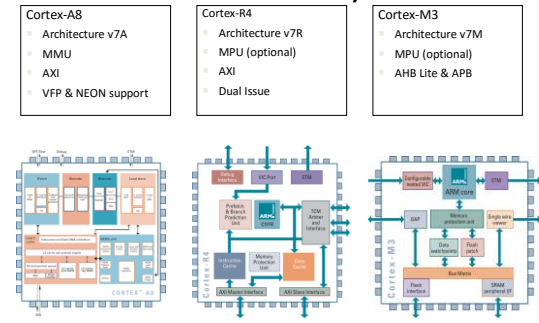
## Famílias Arquitectura v7

- Cortex M0 y M0+
- Cortex M1 → FPG
- Cortex M4



20

## Cortex family



21

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

## Diferencias entre Cortex M4 y M0

- Cortex M0 → 12.000 compuertas
- No existe User Thread ni Faultmask ni Basepri
- No existe IT ni su flag asociado en xPSR
- No existen algunas faults ni debug monitor
- Sólo 4 Niveles de prioridad y no pueden modificarse dinámicamente
- Tabla de vectores fija e inamovible
- Casi todas las instrucciones son de 16 bits

22

## Diferencias entre Cortex M4 y M0

- No existe MPU
- No existe bit banding
- No soporta transferencias no alineadas
- Menos watchpoints (2) and breakpoints (4)
- Debugging sólo con SWI

23

## Cortex M0

Núcleos de 32 bits con excepcionalmente pequeña área de silicio, de baja potencia y código muy pequeño. El bajo costo de estos cores los hacen reemplazos ideales para microcontroladores de 8 - bits y 16 bits. Incluyen un conjunto reducido de instrucciones de los procesadores Cortex-M3 y Cortex-M4 para la compatibilidad del software.

24

---

---

---

---

---

---

---

---



---

---

---

---

---

---

---

---



---

---

---

---

---

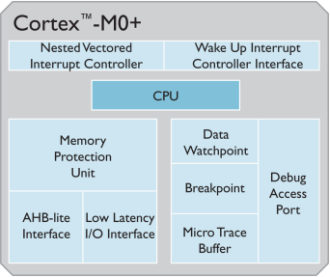
---

---

---



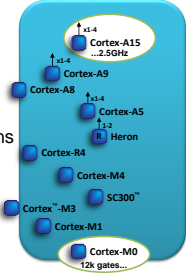
Cortex M0+



25

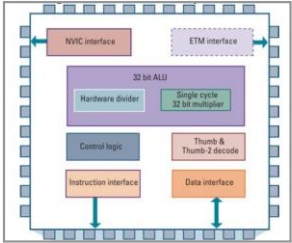
ARM Cortex Processors (v7)

- ARM Cortex-A family (v7-A):
  - Applications processors for full OS and 3<sup>rd</sup> party applications
- ARM Cortex-R family (v7-R):
  - Embedded processors for real-time signal processing, control applications
- ARM Cortex-M family (v7-M):
  - Microcontroller-oriented processors for MCU and SoC applications



26

¿Qué es Cortex™-M?



- Arquitectura Harvard
- pipeline de 3 etapas
- Multiplicación en un solo ciclo
- División por Hardware
- Thumb-2
- Controlador de interrupciones vectorizadas

27

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

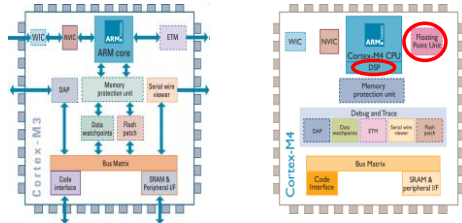
---

---

---

---

## Cortex™-M3 → Cortex™-M4



**Mejoras fundamentales de Cortex™-M4 sobre Cortex-M3:**  
**Instrucciones DSP, instrucciones SIMD, MPU y punto flotante**  
**opcional.**

28

## SIMD

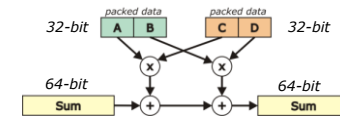
- es una técnica empleada para conseguir paralelismo a nivel de datos.
- Los repertorios SIMD consisten en instrucciones que aplican una misma operación sobre un conjunto más o menos grande de datos. Es una organización en donde una única unidad de control común despacha las instrucciones a diferentes unidades de procesamiento. Todas éstas reciben la misma instrucción, pero operan sobre diferentes conjuntos de datos. Es decir, la misma instrucción es ejecutada de manera sincronizada por todas las unidades de procesamiento.

29

## Operaciones SIMD

SIMD extensions perform multiple operations in one cycle

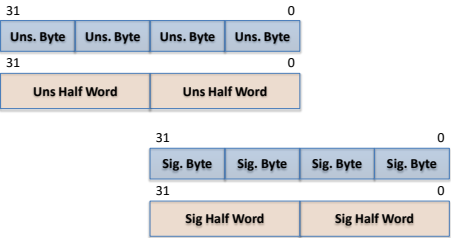
$$Sum = Sum + (A \times C) + (B \times D)$$



SIMD techniques operate with packed data

30

Operaciones SIMD



31

Cortex™-M4 Single Cycle MAC

OPERATION	INSTRUCTION	CYCLES
16 x 16 = 32	SMULBB, SMULBT, SMULTB, SMULTT	1
16 x 16 + 32 = 32	SMLABB, SMLABT, SMLATB, SMLATT	1
16 x 16 + 64 = 64	SMLALBB, SMLALBT, SMLALTB, SMLALTT	1
16 x 32 = 32	SMULWB, SMULWT	1
(16 x 32) + 32 = 32	SMLAWB, SMLAWT	1
(16 x 16) ± (16 x 16) = 32	SMUAD, SMUADX, SMUSD, SMUSDx	1
(16 x 16) ± (16 x 16) + 32 = 32	SMLAD, SMLADX, SMLSOD, SMLSODx	1
(16 x 16) ± (16 x 16) + 64 = 64	SMLALD, SMLALDX, SMLSLOD, SMLSLODX	1
SIMD = Single Instruction Multiple Data		
32 x 32 = 32	MUL	1
32 ± (32 x 32) = 32	MLA, MLS	1
32 x 32 = 64	SMULL, UMULL	1
(32 x 32) + 64 = 64	SMLAL, UMLAL	1
(32 x 32) + 32 + 32 = 64	UMAAL	1
32 ± (32 x 32) = 32 (upper)	SMMLA, SMMLAR, SMMLS, SMMLSR	1
(32 x 32) = 32 (upper)	SMMUL, SMMULR	1

32

Cortex™-M4 SIMD arithmetic

Prefix Instr	S Signed	Q Signed Saturating	SH Signed Halving	U Unsigned	UQ Unsigned Saturating	UH Unsigned Halving
ADD8	SADD8	QADD8	SHADD8	USADD8	UQADD8	UHADD8
SUB8	SSUB8	QSUB8	SHSUB8	USUB8	UQSUB8	UHSUB8
ADD16	SADD16	QADD16	SHADD16	UADD16	UQADD16	UHADD16
SUB16	SSUB16	QSUB16	SHSUB16	USUB16	UQSUB16	UHSUB16
ASX	SASX	QASX	SHASX	UASX	UQASX	UHASX
SAX	SSAX	QSAX	SHSAX	USAX	UQSAX	UHSAX
USAD8	Unsigned Sum of Absolute Difference (8 bits)					
USADA8	Unsigned Sum of Absolute Difference and Accumulate (8 bits)					
ASX			SAX			
1. Exchanges halfwords of the second operand register			1. Exchanges halfwords of the second operand register			
2. Adds top halfwords and subtracts bottom halfwords			2. Subtracts top halfwords and adds bottom halfwords			

33

## Aplicaciones Cortex M4

**Finite Impulse Response (FIR):**  $y_n = \sum_{l=0}^N b_l \cdot x_{n-l}$

**Infinite Impulse Response (IIR):**  $y_n = \sum_{k=0}^N a_k \cdot x_{n-k} + \sum_{k=1}^N b_k \cdot y_{n-k}$

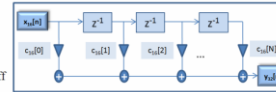
**Discrete Fourier Transform (DFT):**  $X_k = \sum_{n=0}^{N-1} x_n \cdot W_N^{kn}$   
where  $W_N = e^{-j2\pi/N}$

**Fast Fourier Transform (FFT):**  $a \rightarrow A = a + W_N/b$   
Radix-2 Complex DIT Butterfly  
 $b \rightarrow B = a - W_N/b$

34

## DSP Example: Cortex™-M4 FIR

$y_{16}[n] = \sum_{l=0}^N x_{16}[n-l] * c_{16}[l]$   
 $y_{16}$  - filter output (32 bit);  $N$  - filter order  
 $x_{16}$  - 16 bit input data;  $c_{16}$  - 16 bit filter coeff



**Cortex-M3 Code Segment:**

```
FIR_LOOP:
LDR R2,[R0],#4 ;(2) Load input x_16
LDR R3,[R1],#4 ;(2) Load coeff c_16
SXTH R4,R2 ;(1) Extract x_16[n-1]
ASR R2,R2,#16 ;(1) Extract x_16[n-1-1]
SXTH R5,R3 ;(1) Extract c_16[1]
ASR R3,R3,#16 ;(1) Extract c_16[1+1]
PLA R6,R4,R5 ;(2) y_16 += x_16[n-1]*c_16[1]
PLA R6,R2,R3 ;(2) y_16 += x_16[n-1-1]*c_16[1+1]
SUBS R7,R7,#2 ;(1) loop count -= 2
BNE FIR_LOOP ;(2)
```

Note:  
1. In these examples, FIR\_LOOP is unrolled by 2  
2. This example assumes number of taps is even.

Example (non-binding) from Ittiam Systems, a leading provider of signal processing software on ARM platforms

**Cortex-M4 Code Segment:**

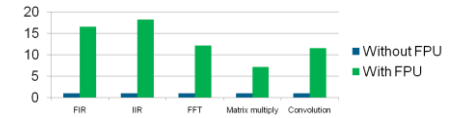
```
FIR_LOOP:
LDR R2,[R0],#4 ;(1) Load input x_16[n-1],x_16[n-1-1]
LDR R3,[R1],#4 ;(2) Load coeff c_16[1],c_16[1+1]
SUBS R5,R5,#2 ;(1) loop count -= 2
SMLAD R6,R2,R3 ;(1) y_16 += x_16[n-1,n-1-1]*c_16[1,1+1]
BNE FIR_LOOP ;(1)
```

Processor	Kernel cycles	Total Cycles	Number of Instructions	Register usage
Cortex-M3	8	15	10	7
Cortex-M4	1	7	5	5
Advantage	8x	~2.2x	2x	1.4x

35

## Cortex™-M4 Single Precision Floating Point

- Floating point benefits
  - Extended range, Highly accurate measurements
- Cortex™-M4 FPU
  - IEEE 754 standard compliant
  - Single-precision floating point math
- Graph below shows Cortex™-M4 single precision floating point algorithm performance normalized to "Without FPU"



36

CMSIS-DSP Library

- Collection of 61 algorithms ([www.onarm.com](http://www.onarm.com))
  - C Source Code, optimized for Cortex-M microcontrollers
  - For CMSIS compliant C Compilers (AM/Keil, IA, GCC)
  - M4 supported with SIMD and FPU operations



37

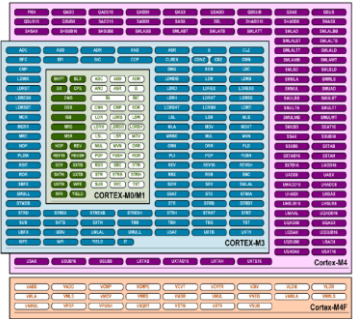
Cortex™-M4 Floating Point Unit

- Single-precision floating point math
  - Add, subtract, multiply, divide, MAC and square root
- Fused MAC

OPERATION	CYCLE COUNT
Add/Subtract	1
Divide	14
Multiply	1
Multiply Accumulate (MAC)	3
Fused MAC	3
Square Root	14

38

Cortex™-M4 Instruction Set Architecture



39

### Cortex™ MCU Software Standard

- Cortex™ Microcontroller Software Interface Standard
  - Abstraction layer for all Cortex-M processor based devices
  - Extended for Cortex™-M4 processor



- Benefits to the embedded developer
  - Consistent software interfaces for silicon and middleware vendors
    - Simplifies re-use across Cortex-M processor-based devices
  - Reduces software development cost and time-to-market

40

---

---

---

---

---

---

---

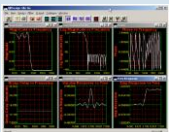
---

### Cortex™-M4 CMSIS Extensions

- C Compiler intrinsic functions for Cortex-M4 DSP and SIMD instructions
- Optimized Floating Point Library using FPU CPU instructions

- Optimized Library using CMSIS
  - Designed to make DSP programs easy to develop for MCU users
  - **General Functions**  
math, trigonometric, control functions (building blocks)
  - **Digital Filter Algorithms**  
for filter design utilities and DSP toolkits (MathLab, LabVIEW, etc.)

D  
UNDER  
EVELOPMENT



41

---

---

---

---

---

---

---

---

### RESUMEN Cortex™-M4

- Combination of MCU and DSP features
- Optional floating point unit
- Code-efficient architecture
- Program completely in C, using CMSIS

42

---

---

---

---

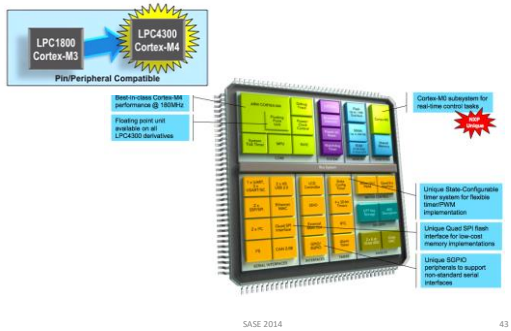
---

---

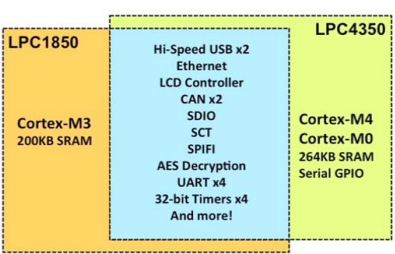
---

---

Procesadores multinúcleo asimétricos



CORTEX M4 vs Cortex M3



Multinúcleo Simétrico vs. Asimétrico

Symmetric	Asymmetric
Single application distributed over N processors of the same type.	Each processor runs a different application.
Requires OS support	Specialized OS not required
Shared program memory	Separate program resource per core

The diagram illustrates the difference between Symmetric and Asymmetric multi-core architectures. In the Symmetric architecture, two cores (Core 1 and Core 2) share a single cache and program memory. In the Asymmetric architecture, each core (Core 1 and Core 2) has its own cache and program memory.

SASE 2014 45

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

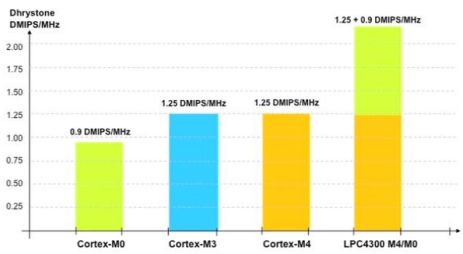
---

---

---

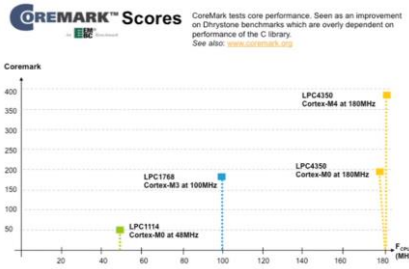
---

Benchmarks - Performance

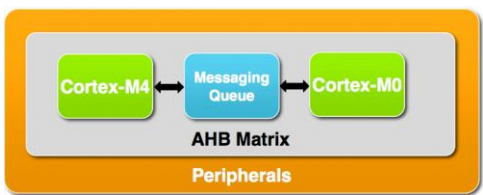


46

Benchmarks - Performance



47



SASE 2014

48

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

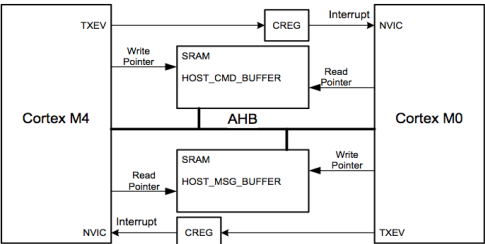
---

---

---

---

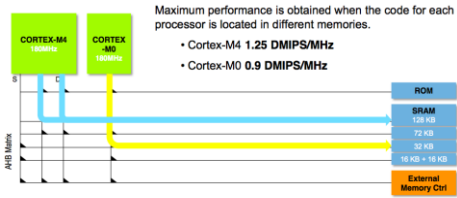




SASE 2014

49

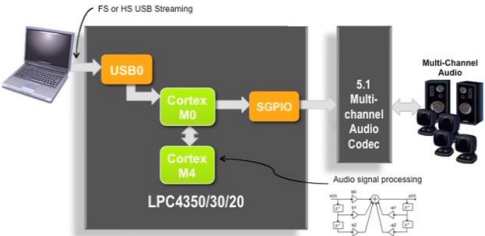
Bus Matrix Cortex M4 + M0



SASE 2014

50

Aplicaciones



51

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

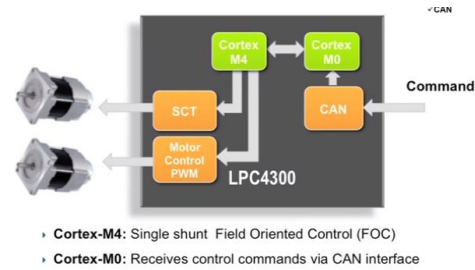
---

---

---

---

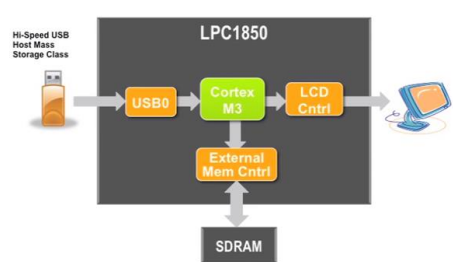
Aplicaciones



- › Cortex-M4: Single shunt Field Oriented Control (FOC)
- › Cortex-M0: Receives control commands via CAN interface

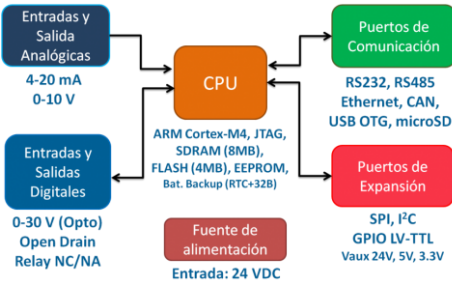
52

Aplicaciones



53

Un caso de estudio CIAA



SAISE 2014

54

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Reloj es

Cortex LPC17xx

55

Características de los tres distintos  
relojes

- 1. Oscilador RC interno (IRC) (Default)**

  - Fuente de reloj para el watchdog y/o el reloj que excita al PLL y subsecuentemente a la CPU.
  - La frecuencia normal del IRC es de 4 MHz con una exactitud de ±1% en todo el entorno de temperatura y tensión
- 2. Oscilador Principal**

  - Fuente de reloj de la CPU usando o no el PLL
  - El oscilador principal también provee la fuente de reloj para el PLL dedicado del USB.
  - Opera a frecuencias desde 1 MHz a 25 MHz
- 3. Oscilador del RTC**

  - Fuente de reloj para el bloque del RTC, el PLL principal y subsecuentemente la CPU
  - Clock de 1 Hz al RTC

Cortex LPC17xx

56

Características detalladas de los  
distintos relojes (1/2)

- PLL Principal (PLL0)**

  - Frecuencia de entrada en el rango de 32 kHz a 25 MHz
  - Puede utilizar el oscilador principal, el oscilador RC interno o el oscilador del RTC
  - Frecuencia de salida: de 10 MHz a la máxima de la CPU
- PLL Secundario (PLL1)**

  - Dedicado a proveer el reloj con la precisión requerida por la interfaz USB. Agrega flexibilidad al PLL Principal. Recibe el PLL0 y genera 48 MHz para USB
- ~~**• Registro(s) de selección del Reloj de Periféricos**~~

  - Utilizados para controlar la señal de reloj que será utilizada para los periféricos individualmente
  - Cada periférico tiene sus ajuste individual de reloj y que puede ser llevado a ser igual al de la CPU o puede ser dividido desde la misma

Cortex LPC17xx

57

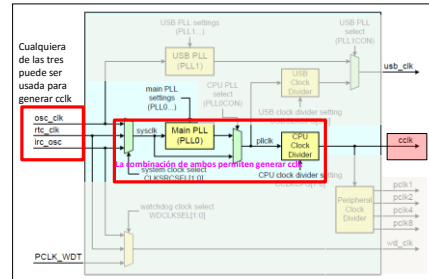
## Características detalladas de los distintos relojes (2/2)

- **Funciones de salida de Reloj**
- Desde la frecuencia generada por el oscilador, se puede dividir para disminuir la potencia consumida al disminuir la frecuencia de trabajo
- Usada para sincronizar dispositivos externos con el reloj de la CPU
- Usada durante el proceso de desarrollo para permitir verificar el funcionamiento del oscilador principal, el reloj IRC, el reloj RTC, el reloj de la CPU (clk), o el reloj de USB

Cortex LPC17xx

58

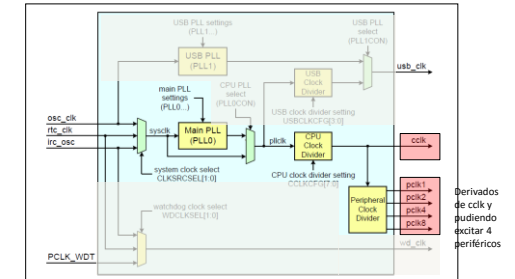
## Ampliación del funcionamiento



Cortex LPC17xx

59

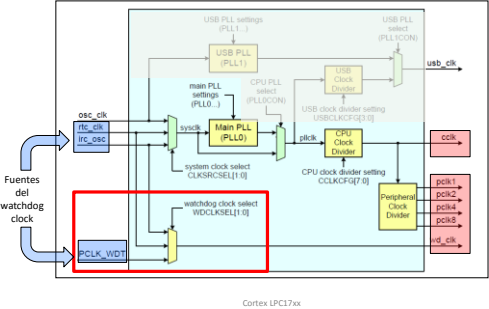
## Ampliación del funcionamiento



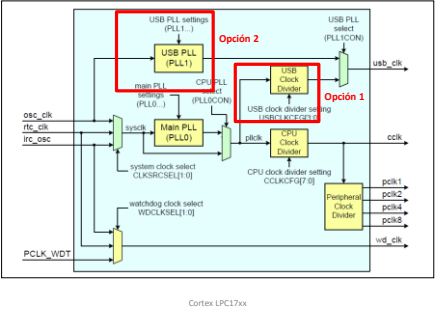
Cortex LPC17xx

60

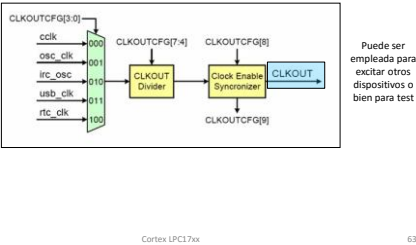
Ampliación del funcionamiento



Ampliación del funcionamiento



Salida de Reloj



---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Controlador de interrupciones  
anidadas NVIC

Cortex LPC17xx

64

Interrupciones

- LPC 17xx soporta 35 interrupciones vectorizadas con 32 niveles de prioridad programables para cada interrupción
  - Se agruparán los valores de campos de prioridad en grupos y subgrupos de prioridades
- NMI = Interrupción externa no enmascarable
- Latencia de interrupciones determinística
- Características avanzadas
  - Prioridad de pre-vaciado
  - Tail chaining (encadenado de colas)

Cortex LPC17xx

65

Terminología (a ser ampliada)

- Determinístico: Que es predecible. Dado un conjunto de entradas, siempre producirá las mismas salidas pasando por la misma secuencia de estados.
- Prevaciado (preemption) Habilidad de un sistema operativo o programa similar de detener la ejecución de una tarea programada a favor de una tarea de mayor prioridad.
- Tail chaining: Forma de encolar los retornos de interrupciones anidadas, de forma de optimizar el tiempo total de ejecución.
- Latencia de interrupciones: Demora experimentada desde que se genera una interrupción hasta que es atendida

Cortex LPC17xx

66

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

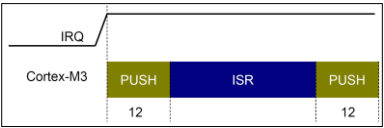
---

---

---

---

Latencia de Interrupciones del NVIC



Latencia de interrupciones determinística

- Cortex-M4 tiene una latencia de interrupción de 12 ciclos y 12 ciclos para retornar de la ISR
- ARM7 tiene una latencia de interrupción entre 24 y 42 ciclos y 16 de retorno. NO ES DETERMINÍSTICO

Cortex LPC17xx

67

Tipos de Excepciones en Cortex M4

Propias de núcleo Cortex

No.	Exception Type	Priority	Type of Priority	Descriptions
1	Reset	-3 (highest)	fixed	Reset
2	NMI	-2	fixed	Non-Maskable Interrupt
3	Hard Fault	-1	fixed	Default fault if other handler not implemented
4	MemManage Fault	0	settable	MPU violation or access to illegal locations
5	Bus Fault	1	settable	Fault if AHB interface receives error
6	Usage Fault	2	settable	Exceptions due to program errors
7-10	Reserved	NA	NA	
11	SVCall	3	settable	System Service Call
12	Debug Monitor	4	settable	Breakpoints, watch points, external debug
13	Reserved	NA	NA	
14	PendSV	5	settable	Pendable request for System Device
15	SYSTICK	6	settable	System Tick Timer
16	Interrupt#0	7	settable	External Interrupt #0
---	---	---	---	---
256	Interrupt#255	247	settable	External Interrupt #240

Cortex LPC17xx

68

Tabla de Vectores

Comienza en dir 0 →  
Pero puede ser reubicado en RAM programando el NVIC

Address	Vector
0x00	Initial Main SP
0x04	Reset
0x08	NMI
0x0C	Hard Fault
0x10	Memory Manage
0x14	Bus Fault
0x18	Usage Fault
0x1C - 0x28	Reserved
0x2C	SVCall
0x30	Debug Monitor
0x34	Reserved
0x38	PendSV
0x3C	Systick
0x40	IRQ0
...	More IRQs

Direcciones (no programa)

Cortex LPC17xx

69

Priorización de la interrupción

- Cada fuente de interrupción tiene un valor de prioridad de 5 bits
- Controladas por los registros del NVIC Reset Contro Reg y el campo PRIGROUP del registro Application Interrupt

Cortex LPC17xx

70

Application Interrupt and Reset Control Register del NVIC

Bits	Name	Type	Function
[31:16]	VECTKEYSTAT	RW	Register key. Reads as 1 (1/1). On writes, write 1s to VECTKEY; otherwise the write is ignored.
[15]	ENDIANESS	RO	Data endianness bit. 0 = Little-endian
[14:13]	Reserved		
[10:8]	PRIGROUP	RW	Interrupt priority grouping field. This field determines the split of group priority from subpriority, see Section 4.4.5.3.1.
[7:6]	Reserved		
[2]	SYSRESETREQ	WO	System reset request. 0 = no system reset request. 1 = asserts a signal to the outer system that requests a reset. This is intended to force a large system reset of all major components except for debug. Note: support for SYSRESETREQ is not included in LPC17xx devices. This bit reads as 0.
[1]	VECTCLRACTIVE	WO	Reserved for Debug use. This bit reads as 0. When writing to the register you must write 0 to this bit, otherwise behavior is Unpredictable.
[0]	VECTRESET	WO	Reserved for Debug use. This bit reads as 0. When writing to the register you must write 0 to this bit, otherwise behavior is Unpredictable.

Cortex LPC17xx

71

Conjunto de bits PRIGROUP

PRIGROUP	Interrupt priority level value, PRI_N[7:0]		Number of	
	Binary point <sup>[1]</sup>	Group priority bits	Subpriority bits	Subpriorities
b010	bxxxx.000	[7:3]	none	32
b011	bxxxx.y000	[7:4]	[3]	16
b100	bxxx.yy000	[7:5]	[4:3]	8
b101	bxx.yyy000	[7:6]	[5:3]	4
b110	bx.yyyy000	[7]	[6:3]	2
b111	b.yyyyy000	None	[7:3]	1

Define cuantos bits son utilizados para el grupo y cuantos para el sub-grupo

Cortex LPC17xx

72



Ejemplo

PRIGROUP (3 bits)	Binary Point (Group: sub)		Pre-empting Priority (Group Priority)		Sub-Priority (Sub-Level)	
			Bits	Levels	Bits	Levels
2	5.0	5 bits for groups, 0 bit for sub-priority	5	32	0	0
3	4.1	4 bits for groups, 1 bit for sub-priority	4	16	1	2
4	3.2	3 bits for groups, 2 bits for sub-priority	3	8	2	4
5	2.3	2 bits for groups, 3 bits for sub-priority	2	4	3	8
6	1.4	1 bit for group, 4 bits for sub-priority	1	2	4	16
7	0.5	0 bit for group, 5 bits for sub-priority	0	0	5	32

Cortex LPC17xx

73

GPIO

Entradas y Salidas (GPIO)

- 70 GPIOs de alta velocidad (LQFP100) y 52 GPIOs (LQFP80)
- Todas las patas tienen pull-ups, pull-downs, o ninguna.
- Todas las patas pueden ser configuradas como open drain
- Los registros de los GPIO estan ubicados en el bus de periféricos AHB para accesos rápidos a E/S
- Los registros de GPIO son accesibles por medio del GPDMA
- Entradas externas de interrupción
  - 46 entradas de interrupción sensibles a flanco
  - 4 entradas de interrupción sensibles a nivel o a flanco
  - Opcionalmente pueden despertar al procesador desde el Power-down

Cortex LPC17xx

74

Cortex LPC17xx

75

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

GPIO e Interrupciones Externas

- Se pueden seleccionar (programando los pines) hasta 46 entradas de interrupción sensibles a flancos (42 GPIO + 4 EINT).
- Las entradas de interrupción pueden ser opcionalmente utilizadas para despertar al procesador de los modos de Power down
- Cualquier pin de los puertos 0 y 2 pueden ser usados para generar interrupción.

Cortex LPC17xx

76

Pinsel

- Permite seleccionar la función de cada pin por ejemplo, de la parte superior del puerto 0

PINSEL1	Pin name	Function when 00	Function when 01	Function when 10	Function when 11	Reset value
1:0	P0_16	GPIO Port 0.16	RXD1	SSEL0	SSEL	00
3:2	P0_17	GPIO Port 0.17	CTS1	MISO0	MISO	00
5:4	P0_18	GPIO Port 0.18	DCD1	MOSI0	MOSI	00
7:6	P0_19	GPIO Port 0.19	DSR1	Reserved	SDA1	00
9:8	P0_20	GPIO Port 0.20	DTR1	Reserved	SCL1	00
11:10	P0_21	GPIO Port 0.21	RT1	Reserved	RD1	00
13:12	P0_22	GPIO Port 0.22	RTS1	Reserved	TD1	00
15:14	P0_23	GPIO Port 0.23	AD0.0	I2SRX_CLK	CAP3.0	00
17:16	P0_24	GPIO Port 0.24	AD0.1	I2SRX_WS	CAP3.1	00
19:18	P0_25	GPIO Port 0.25	AD0.2	I2SRX_SDA	TXD3	00
21:20	P0_26	GPIO Port 0.26	AD0.3	AOUT	RXD3	00
23:22	P0_27	GPIO Port 0.27	SDA0	USB_SDA	Reserved	00
25:24	P0_28	GPIO Port 0.28	SCL0	USB_SCL	Reserved	00

Cortex LPC17xx

77

Timers

Cortex LPC17xx

78

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

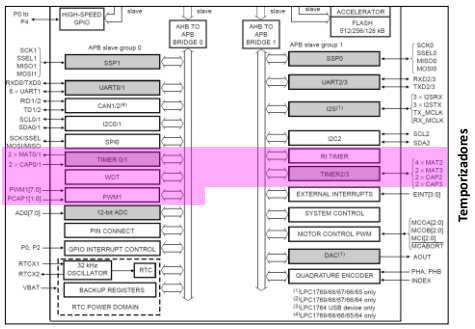
---

---

---

---

Timers del LPC4337



Temporizadores

Timers

- Cuatro timers de propósito general
- Watchdog
- Timer generador de interrupciones repetitivo
- PWM (Operación del timer)
- SysTick

Cortex LPC17xx

80

Timer (0, 1, 2, y 3), Watchdog

- **Timers/counters de 32-bit**
- 8 entradas de captura externa
- 4 salidas externas de coincidencia (match)
- Modo de temporización o de conteo
- Soporte de controlador GPDMA
  - Permite transferencias de memoria a memoria temporizadas
- **Watchdog Timer**
- Tiempo programable de 32-bits
- Temporizado por el reloj IRC o por el de periféricos
- Puede ser utilizado para despertar al sistema en los modos de bajo consumo
- NO puede ser deshabilitado por programa

Cortex LPC17xx

81

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

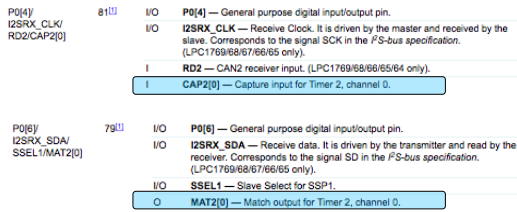
Particularidades de los Timers

- **Timer de interrupción repetitiva (RIT)**
- Timer de 32-bits
  - Astable (Free-running)
  - Reset on Match Interrupt
- Útil para eventos repetitivos definidos por el usuario
- **Timer SYSTICK (parte del NVIC)**
- Timer decreciente de 24 bits dedicado de 10 ms.
- Temporizado desde el reloj de la CPU o desde una pata (STCLK)
- Útil para sistemas operativos o despachadores de tareas

Cortex LPC17xx

82

Pines del timer



Cortex LPC17xx

83

Real-Time Clock (RTC)

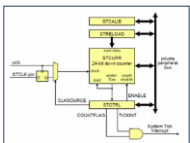
- Ultra-low power 32 kHz oscillator provides 1 Hz clock to the RTC
- Separate battery power supply
- Uses CPU power supply, when present
- Calibration mechanism
- ±1 second per day
- Battery-backed registers -20 bytes
- Alarm function generates interrupts
- Wakes CPU from reduced power modes
- 1 second resolution
- Extremely low power consumption
- 390 nA (typical @ 25°C)
- Calendar function does not require CPU involvement
- RTC works with Vbat as low as 2.1 V

Cortex LPC17xx

84

Systick

- Timer de 24 bits que se decreuenta a 0 y provee 10 ms entre interrupciones (a 100 MHz de CPU)
- Puede utilizar como base de tiempo el clock de la CPU o STCLK

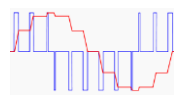


Cortex LPC17xx

85

Modulación del ancho de pulso (PWM)

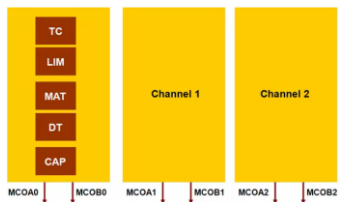
- **Bloque estandar PWM**
- Resolución de 32-bits
- 6 salidas PWM single-edge ó 3 double-edge
- Mecanismo de Latch sombreado
  - Operación sin glitches
- Bloque PWM con operación de contador o de timer
  - Fuente de reloj: Reloj de periféricos o entradas de captura
  - Puede ser usado como timer/contador de propósito general



Cortex LPC17xx

86

PWM para control de motores trifásicos



Cortex LPC17xx

87

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

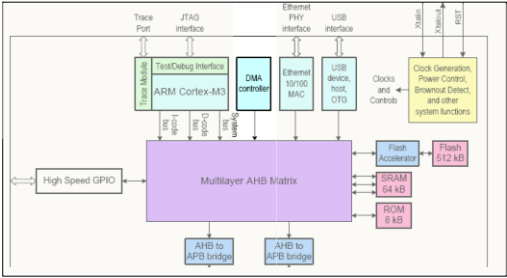
---

---



DMA

Controlador de DMA de propósito General (GPDMA)



Cortex LPC17xx

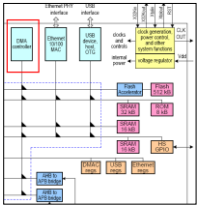
91

Cortex LPC17xx

92

Controlador de DMA de propósito General (GPDMA)

- Soporta transferencias de periféricos de alta velocidad como transferencias de memoria a memoria
  - 32-bit master bus width (soporta transferencias de 8-, 16-, or 32-bits)
  - 8 Canales de DMA, cada uno con una FIFO de 4 palabras
  - 16 líneas de requerimiento de DMA
- Transferencias soportadas:
  - Memoria a Memoria
  - Memoria a Periférico
  - Periférico a Memoria
  - Periférico a Periférico



Cortex LPC17xx

93

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

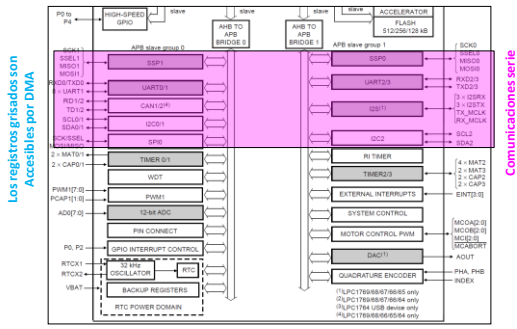
Controlador de DMA de propósito General (GPDMA)

- DMA soporta los siguientes periféricos:
    - ADC de 12-bits
    - DAC de 10-bits
    - Pueden ser disparados por una Timer match condition
    - Todas las UARTs
  - SSP
  - I2S
  - GPIO
- Señales de solicitud de DMA simple y DMA en ráfagas

Cortex LPC17xx

94

Periféricos del LPC4337



DMA

Peripheral Function	DMA Single Request Input (DMACSREQ)	DMA Burst Request Input (DMACBREQ)
SSP0 Tx	0	0
SSP0 Rx	1	1
SSP1 Tx	2	2
SSP1 Rx	3	3
ADC	4	4
PS channel 0	-	5
PS channel 1	-	6
DAC	-	7
UART0 Tx / MAT0.0	-	8
UART0 Rx / MAT0.1	-	9
UART1 Tx / MAT1.0	-	10
UART1 Rx / MAT1.1	-	11
UART2 Tx / MAT2.0	-	12
UART2 Rx / MAT2.1	-	13
UART3 Tx / MAT3.0	-	14
UART3 Rx / MAT3.1	-	15

Cortex LPC17xx

96



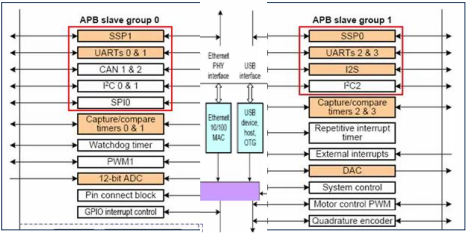
Interfaces Serie



Cortex LPC17xx

97

Interfaces serie



Cortex LPC17xx

98

SPI y SSP(0 y 1)

- **Controlador SPI (Serial Peripheral Interface)**
- Comunicación síncrona, serie y full Duplex.
- SPI master ó slave
- Transferencia de 8 ó 16 bits
- Reloj con polaridad y fase programables para operaciones de recepción/transmisión
- Máxima velocidad (master/slave) 12.5 Mbps

- **Controlador SSP (Synchronous Serial Communication)**
- FIFOs de 8 tramas tanto para Transmisión y Recepción con múltiples masters y slaves (una comunicación por vez)
- Transferencias de datos de 4 a 16-bits
- Soporte de DMA
- Máxima velocidad
  - •50 Mbps (Master Mode)
  - •8 Mbps (Slave Mode)

Cortex LPC17xx

99

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

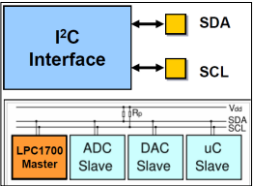
---

---

---

I<sup>2</sup>C (0, 1, & 2)

- I2C Bus interface (Inter-Integrated Circuit)
- Satisface la interfaz I2C
- Puede ser configurado como Master, Slave o Master/Slave
- Transferencia de datos bidireccional entre Masters y Slaves.
- Puerto compatible con Fast-mode Plus I2C (1 Mbit/sec) (I2C0)
- Reloj Programable para permitir el ajuste de las tasas de transferencia I2C
- Reconocimiento opcional de hasta 4 distintas direcciones de esclavo
- Modo de monitoreo del bus

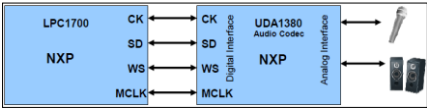


Cortex LPC17xx

100

Interfaz I<sup>2</sup>S (Inter-IC Sound)

- Entrada o salida de audio digital con control de velocidad fraccionario.
- Soporta conexiones combinadas de transmisión o recepción de 3 ó 4 cables
- Entrada/salida de reloj de audio Master (utilizada por muchos CODECs I2S)
- La entrada y salida I2S pueden operar independientemente como master o slave.
- Soporta tanto los flujos de datos mono o estereo sobre un amplio rango de frecuencias de muestreo que pueden variar de 16 a 96 kHz
- El soporte GPDMA permite el flujo de datos de audio sobre la interfaz I2S



Cortex LPC17xx

101

Interfaces Serie UART (0, 1, 2, y 3)

- **UART (Universal Asynchronous Receiver/Transmitter)**
- FIFO de 16 bytes para Recepción y Transmisión con soporte de DMA
- Divisor fraccionario para control del baud rate , capacidad de auto-baud, e implementación de control de flujo por hardware o software.
- La UART1 soporta EIA-485/RS-485 y operaciones con 9-bits.
  - Permite tanto la detección de la dirección por software y detección automática de la dirección por medio de la comunicación de 9 bits.
  - Control automático del sentido de la comunicación (Auto Direction)
    - Línea de Control RTS/CTS para habilitar o deshabilitar el driver
    - El Software levanta la línea RTS antes de comenzar la transmisión a fin de habilitar el driver, posteriormente baja la línea RTS luego de completar la transmisión.
- Soporte para el control de modem (UART1)
- Soporte de comunicaciones infrarrojas IrDA (UART3)
- Velocidad máxima posible de la UART ~ 6 Mbps

Cortex LPC17xx

102

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

CAN (1 y 2), Ethernet, USB

- **Controlador CAN 2.0B (Controller Area Network)**
- Compatible con la especificación CAN 2.0B
- Velocidades de hasta 1 Mbps en cada bus
- Filtro de aceptación por hardware que reconoce identificadores de 11 y 29-bit Rx
- **Interfaz Ethernet**
- MAC Ethernet con interfaz RMII (Reduced Media Independent Interface)
- Soporta dispositivos con PHY de 10 o 100 Mbps
- Controlador dedicado de DMA
- Completamente compatible 802.3x Full Duplex Flow Control y Half Duplex back pressure
- **Bloque USB**
- USB 2.0 Full Speed (12 Mbps) Device, Host, OTG
- On-chip PHY para implementar funciones tanto como Host o dispositivo
- Controlador dedicado de DMA

Cortex LPC17xx

103

USB

- USB 2.0 Full Speed (12 Mbps) como Device e interfaz de control On-the-Go/Open Host
- Phy incorporada para funciones Device/Host/OTG
- Controlador de DMA dedicado

Core	Product	On-Chip Controller			No. of Ports	On-chip PHY
		Device	Host	OTG		
Cortex-M3	LPC1751	FS	-	-	1	Device
	LPC1752	FS	-	-	1	Device
	LPC1754	FS	FS	FS	1	Device, Host
	LPC1756	FS	FS	FS	1	Device, Host
	LPC1758	FS	FS	FS	1	Device, Host
	LPC1764	FS	-	-	1	Device
	LPC1765	FS	FS	FS	1	Device, Host
	LPC1766	FS	FS	FS	1	Device, Host
	LPC1768	FS	FS	FS	1	Device, Host

Cortex LPC17xx

104

Bloques Analógicos

Cortex LPC17xx

105

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

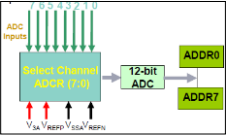
---

---

---

Conversor A/D

- **12-bit ADC (Analog-to-Digital Converter)**
- Aproximaciones sucesivas.
- Max. Velocidad de conversión a 12 bits: 200 kHz.
- Ocho entradas analógicas
- Soporte para el controlador GPDMA
- Modo de bajo consumo
- Rango de entradas al ADC: VREFN hasta VREFP
- Modo de conversión en ráfagas para una o varias entradas
- Conversión con la transición de la entrada o por timer
- No linealidad integral (INL)  $\pm 3$  LSB
- No linealidad diferencial (DNL)  $\pm 1$  LSB



Cortex LPC17xx

106

DAC (Digital-to-Analog Converter) de 10 bits

- Salida Analógica desde VREFN hasta VREFP en 1024 pasos
- Soporte del controlador GPDMA
- Tiempo de conversión = 2.5  $\mu$ s con una excitación de 350  $\mu$ A
- Tiempo de conversión = 1  $\mu$ s con una excitación de 700  $\mu$ A

Cortex LPC17xx

107

POR y BOD

- Power on reset
- BOD: Si la tensión de alimentación cae debajo de 2,95 V se puede generar una interrupción que permita salvar registros y variables en algún medio no volátil

Cortex LPC17xx

108

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

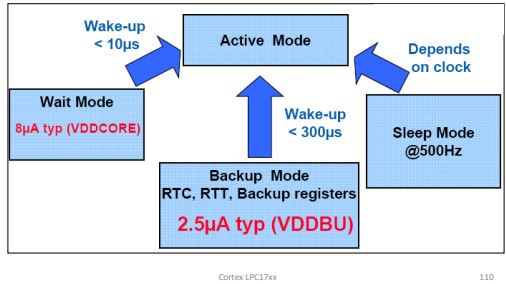
---

Modos de baja energía consumida

Cortex LPC17xx

109

Modos bajo consumo



Cortex LPC17xx

110

Alimentaciones

- 3,3 V (2,4 a 3,6 V)
  - Vdd (reg) (3V3) – regulador interno
  - Vdd (3V3) – Pads de E/S
- Vbat (2,1 V a 3,6 V)
  - Alimenta sólo al RTC
  - No alimenta RAM
  - Alimenta 20 bytes de registros de backup
- Conversor A/D de 12 bits
  - Vdda (parte analógica)
  - Vrefp

Cortex LPC17xx

111

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

Opciones de energía

- **Modos de energía reducida**
- Amplio rango de fuentes de reloj
- Posibilidad de dividir o apagar relojes a los periféricos on-chip en forma individual
- Ireg= 45 mA a 100 MHz (Típico a@ 25°C)
- Potencia activa de Cortex-M4 ~ 500µA/MHz
- Potencia activa de ARM7 ~ 700µA/MHz

Cortex LPC17xx

112

Modos de energía reducida (1/2)

- **Sleep**
- Se suspende la ejecución de la CPU
- Los periféricos continúan trabajando
- (Similar al modo Idle del ARM7)
- Ireg= 2.28 mA (valores típicos@ 25°C)
- **Deep-Sleep**
- El oscilador principal y todos los relojes internos excepto el IRC son detenidos
- La memoria Flash está en standby, lista para uso inmediato
- Ireg= 276 µA (valores típicos@ 25°C)

Cortex LPC17xx

113

Modos de energía reducida (2/2)

- **Power-down**
- Igual que en el modo Deep-Sleep excepto que se apagan Flash e IRC
- Se recupera por medio de algunas interrupciones
- Se preserva el estado
- Ireg= 51 µA (valores típicos@ 25°C)
- **Power-down Profundo**
- Todos los relojes, incluyendo el IRC son detenidos y la tensión interna es desconectada
- Se pierde completamente el estado del sistema, sólo se preservan los registros del dominio del RTC
- Se reactiva por medio de Reset, señal externa o alarma del RTC
- Ibat= 390 nA (valores típicos@ 25°C)
- )

Cortex LPC17xx

114

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

# Wakeup Interrupt Controller (WIC)

- Permite despertar de los modos Deep Sleep y Power down (sin utilizar el NVIC).
- Completamente controlada por hardware (no requiere programación)
- Interrupciones empleadas
- NMI, Interrupciones externas EINT0 a EINT3, interrupciones de GPIO, Interrupción de despertado por Ethernet, Alarma del RTC, CAN ó USB.
- Watchdog despierta al modo Deep sleep si utiliza el oscilador IRC

Cortex-LPC17xx

115

---

---

---

---

---

---

---