LPC2921/2923/2925

ARM9 microcontroller with CAN, LIN, and USB

Rev. 03 — 14 April 2010

Product data sheet

1. General description

The LPC2921/2923/2925 combine an ARM968E-S CPU core with two integrated TCM blocks operating at frequencies of up to 125 MHz, Full-speed USB 2.0 device controller, CAN and LIN, up to 40 kB SRAM, up to 512 kB flash memory, two 10-bit ADCs, and multiple serial and parallel interfaces in a single chip targeted at consumer, industrial, and communication markets. To optimize system power consumption, the LPC2921/2923/2925 has a very flexible Clock Generation Unit (CGU) that provides dynamic clock gating and scaling.

2. Features and benefits

- ARM968E-S processor running at frequencies of up to 125 MHz maximum.
- Multilayer AHB system bus at 125 MHz with four separate layers.
- On-chip memory:
    - Two Tightly Coupled Memories (TCM), 16 kB Instruction (ITCM) and 16 kB Data TCM (DTCM).
    - On the LPC2925, two separate internal Static RAM (SRAM) instances, 16 kB each.
    - On the LPC2923 and LPC2921, one 16 kB SRAM block.
    - 8 kB ETB SRAM, also usable for code execution and data.
    - Up to 512 kB high-speed flash-program memory.
    - 16 kB true EEPROM, byte-erasable/programmable.
- Dual-master, eight-channel GPDMA controller on the AHB multilayer matrix which can be used with the Serial Peripheral Interface (SPI) interfaces and the UARTs, as well as for memory-to-memory transfers including the TCM memories.
- Serial interfaces:
    - USB 2.0 full-speed device controller with dedicated DMA controller and on-chip device PHY.
    - Two-channel CAN controller supporting FullCAN and extensive message filtering.
    - Two LIN master controllers with full hardware support for LIN communication. The LIN interface can be configured as UART to provide two additional UART interfaces.
    - Two 550 UARTs with 16-byte Tx and Rx FIFO depths, DMA support, and RS-485/EIA-485 (9-bit) support.
    - Three full-duplex Q-SPIs with four slave-select lines; 16 bits wide; 8 locations deep; Tx FIFO and Rx FIFO.
    - Two I^{2}C-bus interfaces.

- Other peripherals:
    - Two 10-bit ADCs, 8-channels each, with 3.3 V measurement range provide 8 analog inputs each with conversion times as low as 2.44 μs per channel. Each channel provides a compare function to minimize interrupts.
    - Multiple trigger-start option for all ADCs: timer, PWM, other ADC and external signal input.
    - Four 32-bit timers each containing four capture-and-compare registers linked to I/Os.
    - Four six-channel PWMs (Pulse Width Modulators) with capture and trap functionality.
    - Two dedicated 32-bit timers to schedule and synchronize PWM and ADC.
    - Quadrature encoder interface that can monitor one external quadrature encoder.
    - 32-bit watchdog with timer change protection, running on safe clock.
- Up to 60 general-purpose I/O pins with programmable pull-up, pull-down, or bus keeper.
- Vectored Interrupt Controller (VIC) with 16 priority levels.
- Up to 16 level-sensitive external interrupt pins, including USB, CAN and LIN wake-up features.
- Configurable clock out pin for driving external system clocks.
- Processor wake-up from power-down via external interrupt pins and CAN or LIN activity.
- Flexible Reset Generator Unit (RGU) able to control resets of individual modules.
- Flexible Clock-Generation Unit (CGU) able to control clock frequency of individual modules:
    - On-chip very low-power ring oscillator; fixed frequency of 0.4 MHz; always on to provide a Safe_Clock source for system monitoring.
    - On-chip crystal oscillator with a recommended operating range from 10 MHz to 25 MHz. PLL input range 10 MHz to 25 MHz.
    - On-chip PLL allows CPU operation up to a maximum CPU rate of 125 MHz.
    - Generation of up to 11 base clocks.
    - Seven fractional dividers.
- Second, dedicated CGU with its own PLL generates the USB clock and a configurable clock output.
- Highly configurable system Power Management Unit (PMU):
    - clock control of individual modules.
    - allows minimization of system operating power consumption in any configuration.
- Standard ARM test and debug interface with real-time in-circuit emulator.
- Boundary-scan test supported.
- ETM/ETB debug functions with 8 kB of dedicated SRAM also accessible for application code and data storage.
- Dual power supply:
    - CPU operating voltage: 1.8 V ± 5 %.
    - I/O operating voltage: 2.7 V to 3.6 V; inputs tolerant up to 5.5 V.
- 100-pin LQFP package.
- –40 °C to +85 °C ambient operating temperature range.

3. Ordering information

Table 1. Ordering information

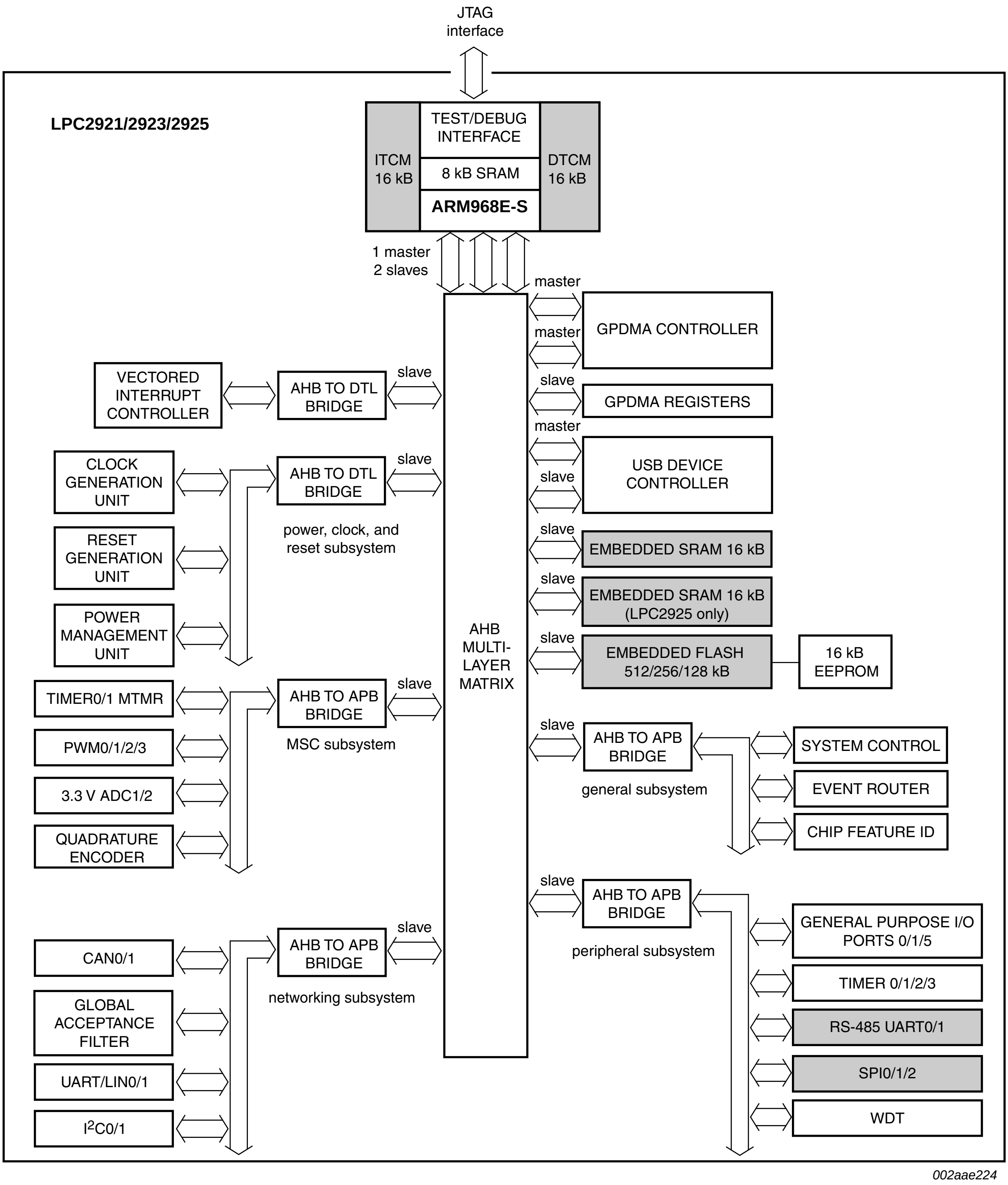
Type number	Package		
	Name	Description	Version
LPC2921FBD100	LQFP100	plastic low profile quad flat package; 100 leads; body 14 × 14 × 1.4 mm	SOT407-1
LPC2923FBD100	LQFP100	plastic low profile quad flat package; 100 leads; body 14 × 14 × 1.4 mm	SOT407-1
LPC2925FBD100	LQFP100	plastic low profile quad flat package; 100 leads; body 14 × 14 × 1.4 mm	SOT407-1

3.1 Ordering options

Table 2. Part options

Type number	Flash memory	SRAM (incl. ETB SRAM)	USB device	UART RS-485	LIN 2.0/ UART	CAN	Package
LPC2921FBD100	128 kB	24 kB	yes	2	2	2	LQFP100
LPC2923FBD100	256 kB	24 kB	yes	2	2	2	LQFP100
LPC2925FBD100	512 kB	40 kB	yes	2	2	2	LQFP100

4. Block diagram



Grey-shaded blocks represent peripherals and memory regions accessible by the GPDMA.

Fig 1. LPC2921/2923/2925 block diagram

5. Pinning information

5.1 Pinning

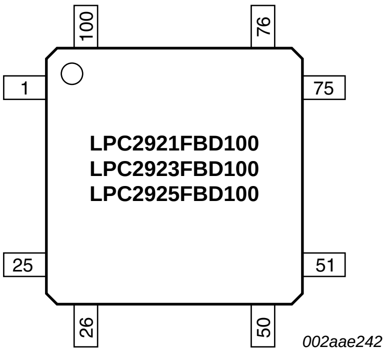


Fig 2. Pin configuration for SOT407-1 (LQFP100)

5.2 Pin description

5.2.1 General description

The LPC2921/2923/2925 uses three ports: port 1 with 32 pins, port 1 with 28 pins, and port 5 with 2 pins. Ports 4/3/2 are not used. The pin to which each function is assigned is controlled by the SFSP registers in the SCU. The functions combined on each port pin are shown in the pin description tables in this section.

5.2.2 LQFP100 pin assignment

Table 3. LQFP100 pin assignment

Pin name	Pin	Description			
		Function 0 (default)	Function 1	Function 2	Function 3
TDO	1[1]	IEEE 1149.1 test data out			
P0[24]/TXD1/ TXDC1/SCS2[0]	2[1]	GPIO0, pin 24	UART1 TXD	CAN1 TXD	SPI2 SCS0
P0[25]/RXD1/ RXDC1/SDO2	3[1]	GPIO0, pin 25	UART1 RXD	CAN1 RXD	SPI2 SDO
P0[26]/TXD1/SDI2	4[1]	GPIO0, pin 26	-	UART1 TXD	SPI2 SDI
P0[27]/RXD1/SCK2	5[1]	GPIO0, pin 27	-	UART1 RXD	SPI2 SCK
P0[28]/CAP0[0]/ MAT0[0]	6[1]	GPIO0, pin 28	-	TIMER0 CAP0	TIMER0 MAT0
P0[29]/CAP0[1]/ MAT0[1]	7[1]	GPIO0, pin 29	-	TIMER0 CAP1	TIMER0 MAT1
$V_{DD(IO)}$	8	3.3 V power supply for I/O			
P0[30]/CAP0[2]/ MAT0[2]	9[1]	GPIO0, pin 30	-	TIMER0 CAP2	TIMER0 MAT2
P0[31]/CAP0[3]/ MAT0[3]	10[1]	GPIO0, pin 31	-	TIMER0 CAP3	TIMER0 MAT3
$V_{SS(IO)}$	11	ground for I/O			

Table 3. LQFP100 pin assignment *...continued*

Pin name	Pin	Description			
		Function 0 (default)	**Function 1**	**Function 2**	**Function 3**
P5[19]/USB_D+	12[2]	GPIO5, pin 19	USB_D+	-	-
P5[18]/USB_D–	13[2]	GPIO5, pin 18	USB_D–	-	-
$V_{DD(IO)}$	14	3.3 V power supply for I/O			
$V_{DD(CORE)}$	15	1.8 V power supply for digital core			
$V_{SS(CORE)}$	16	ground for core			
$V_{SS(IO)}$	17	ground for I/O			
P1[27]/CAP1[2]/ TRAP2/PMAT3[3]	18[1]	GPIO1, pin 27	TIMER1 CAP2, ADC2 EXT START	PWM TRAP2	PWM3 MAT3
P1[26]/PMAT2[0]/ TRAP3/PMAT3[2]	19[1]	GPIO1, pin 26	PWM2 MAT0	PWM TRAP3	PWM3 MAT2
$V_{DD(IO)}$	20	3.3 V power supply for I/O			
P1[25]/PMAT1[0]/ USB_VBUS/ PMAT3[1]	21[1]	GPIO1, pin 25	PWM1 MAT0	USB_VBUS	PWM3 MAT1
P1[24]/PMAT0[0]/ $\overline{\text{USB\_CONNECT}}$/ PMAT3[0]	22[1]	GPIO1, pin 24	PWM0 MAT0	$\overline{\text{USB\_CONNECT}}$	PWM3 MAT0
P1[23]/RXD0	23[1]	GPIO1, pin 23	UART0 RXD	-	-
P1[22]/TXD0/ $\overline{\text{USB\_UP\_LED}}$	24[1]	GPIO1, pin 22	UART0 TXD	$\overline{\text{USB\_UP\_LED}}$	-
TMS	25[1]	IEEE 1149.1 test mode select, pulled up internally			
TCK	26[1]	IEEE 1149.1 test clock			
P1[21]/CAP3[3]/ CAP1[3]	27[1]	GPIO1, pin 21	TIMER3 CAP3	TIMER1 CAP3, MSCSS PAUSE	-
P1[20]/CAP3[2]/ SCS0[1]	28[1]	GPIO1, pin 20	TIMER3 CAP2	SPI0 SCS1	-
P1[19]/CAP3[1]/ SCS0[2]	29[1]	GPIO1, pin 19	TIMER3 CAP1	SPI0 SCS2	-
P1[18]/CAP3[0]/ SDO0	30[1]	GPIO1, pin 18	TIMER3 CAP0	SPI0 SDO	-
P1[17]/CAP2[3]/ SDI0	31[1]	GPIO1, pin 17	TIMER2 CAP3	SPI0 SDI	-
$V_{SS(IO)}$	32	ground for I/O			
P1[16]/CAP2[2]/ SCK0	33[1]	GPIO1, pin 16	TIMER2 CAP2	SPI0 SCK	-
P1[15]/CAP2[1]/ SCS0[0]	34[1]	GPIO1, pin 15	TIMER2 CAP1	SPI0 SCS0	-
P1[14]/CAP2[0]/ SCS0[3]	35[1]	GPIO1, pin 14	TIMER2 CAP0	SPI0 SCS3	-
P1[13]/EI3/SCL1	36[1]	GPIO1, pin 13	EXTINT3	I^{2}C1 SCL	-
P1[12]/EI2/SDA1	37[1]	GPIO1, pin 12	EXTINT2	I^{2}C1 SDA	-
$V_{DD(IO)}$	38	3.3 V power supply for I/O			
P1[11]/SCK1/SCL0	39[1]	GPIO1, pin 11	SPI1 SCK	I^{2}C0 SCL	-

Table 3. LQFP100 pin assignment *…continued*

Pin name	Pin	Description			
		Function 0 (default)	Function 1	Function 2	Function 3
P1[10]/SDI1/SDA0	40[1]	GPIO1, pin 10	SPI1 SDI	I^2C0 SDA	-
$V_{SS(CORE)}$	41	ground for digital core			
$V_{DD(CORE)}$	42	1.8 V power supply for digital core			
P1[9]/SDO1	43[1]	GPIO1, pin 9	SPI1 SDO	-	-
$V_{SS(IO)}$	44	ground for I/O			
P1[8]/SCS1[0]/ TXDL1/CS0	45[1]	GPIO1, pin 8	SPI1 SCS0	-	-
P1[7]/SCS1[3]/RXD1	46[1]	GPIO1, pin 7	SPI1 SCS3	UART1 RXD	-
P1[6]/SCS1[2]/TXD1	47[1]	GPIO1, pin 6	SPI1 SCS2	UART1 TXD	-
P1[5]/SCS1[1]/ PMAT3[5]	48[1]	GPIO1, pin 5	SPI1 SCS1	PWM3 MAT5	-
P1[4]/SCS2[2]/ PMAT3[4]	49[1]	GPIO1, pin 4	SPI2 SCS2	PWM3 MAT4	-
$\overline{\text{TRST}}$	50[1]	IEEE 1149.1 test reset NOT; active LOW; pulled up internally			
$\overline{\text{RST}}$	51[1]	asynchronous device reset; active LOW; pulled up internally			
$V_{SS(OSC)}$	52	ground for oscillator			
XOUT_OSC	53[3]	crystal out for oscillator			
XIN_OSC	54[3]	crystal in for oscillator			
$V_{DD(OSC\_PLL)}$	55	1.8 V supply for oscillator and PLL			
$V_{SS(PLL)}$	56	ground for PLL			
$V_{DD(IO)}$	57	3.3 V power supply for I/O			
P1[3]/SCS2[1]/ PMAT3[3]	58[1]	GPIO1, pin 3	SPI2 SCS1	PWM3 MAT3	-
P1[2]/SCS2[3]/ PMAT3[2]	59[1]	GPIO1, pin 2	SPI2 SCS3	PWM3 MAT2	-
P1[1]/EI1/PMAT3[1]	60[1]	GPIO1, pin 1	EXTINT1	PWM3 MAT1	-
$V_{SS(CORE)}$	61	ground for digital core			
$V_{DD(CORE)}$	62	1.8 V power supply for digital core			
P1[0]/EI0/PMAT3[0]	63[1]	GPIO1, pin 0	EXTINT0	PWM3 MAT0	-
P0[0]/PHB0/ TXDC0/D24	64[1]	GPIO0, pin 0	QEI0 PHB	CAN0 TXD	-
$V_{SS(IO)}$	65	ground for I/O			
P0[1]/PHA0/RXDC0	66[1]	GPIO0, pin 1	QEI0 PHA	CAN0 RXD	-
P0[2]/CLK_OUT/ PMAT0[0]	67[1]	GPIO0, pin 2	CLK_OUT	PWM0 MAT0	-
P0[3]/$\overline{\text{USB\_UP\_LED}}$/ PMAT0[1]	68[1]	GPIO0, pin 3	$\overline{\text{USB\_UP\_LED}}$	PWM0 MAT1	-
P0[4]/PMAT0[2]	69[1]	GPIO0, pin 4	-	PWM0 MAT2	-
P0[5]/PMAT0[3]	70[1]	GPIO0, pin 5	-	PWM0 MAT3	-
$V_{DD(IO)}$	71	3.3 V power supply for I/O			
P0[6]/PMAT0[4]	72[1]	GPIO0, pin 6	-	PWM0 MAT4	-
P0[7]/PMAT0[5]	73[1]	GPIO0, pin 7	-	PWM0 MAT5	-

Table 3. LQFP100 pin assignment *…continued*

Pin name	Pin	Description			
		Function 0 (default)	Function 1	Function 2	Function 3
$V_{DDA(ADC3V3)}$	74	3.3 V power supply for ADC			
JTAGSEL	75[1]	TAP controller select input; LOW-level selects the ARM debug mode; HIGH-level selects boundary scan; pulled up internally.			
n.c.	76	not connected to a function; must be tied to 3.3 V power supply for ADC $V_{DDA(ADC3V3)}$.			
VREFP	77[3]	HIGH reference for ADC			
VREFN	78[3]	LOW reference for ADC			
P0[8]/IN1[0]	79[4]	GPIO0, pin 8	ADC1 IN0	-	-
P0[9]/IN1[1]	80[4]	GPIO0, pin 9	ADC1 IN1	-	-
P0[10]/IN1[2]/ PMAT1[0]	81[4]	GPIO0, pin 10	ADC1 IN2	PWM1 MAT0	-
P0[11]/IN1[3]/ PMAT1[1]	82[4]	GPIO0, pin 11	ADC1 IN3	PWM1 MAT1	-
$V_{SS(IO)}$	83	ground for I/O			
P0[12]/IN1[4]/ PMAT1[2]	84[4]	GPIO0, pin 12	ADC1 IN4	PWM1 MAT2	-
P0[13]/IN1[5]/ PMAT1[3]	85[4]	GPIO0, pin 13	ADC1 IN5	PWM1 MAT3	-
P0[14]/IN1[6]/ PMAT1[4]	86[4]	GPIO0, pin 14	ADC1 IN6	PWM1 MAT4	-
P0[15]/IN1[7]/ PMAT1[5]	87[4]	GPIO0, pin 15	ADC1 IN7	PWM1 MAT5	-
P0[16]IN2[0]/TXD0	88[4]	GPIO0, pin 16	ADC2 IN0	UART0 TXD	-
P0[17]/IN2[1]/ RXD0/A23	89[4]	GPIO0, pin 17	ADC2 IN1	UART0 RXD	-
$V_{DD(CORE)}$	90	1.8 V power supply for digital core			
$V_{SS(CORE)}$	91	ground for digital core			
$V_{DD(IO)}$	92	3.3 V power supply for I/O			
P0[18]/IN2[2]/ PMAT2[0]	93[4]	GPIO0, pin 18	ADC2 IN2	PWM2 MAT0	-
P0[19]/IN2[3]/ PMAT2[1]	94[4]	GPIO0, pin 19	ADC2 IN3	PWM2 MAT1	-
P0[20]/IN2[4]/ PMAT2[2]	95[4]	GPIO0, pin 20	ADC2 IN4	PWM2 MAT2	-
P0[21]/IN2[5]/ PMAT2[3]	96[4]	GPIO0, pin 21	ADC2 IN5	PWM2 MAT3	-
P0[22]/IN2[6]/ PMAT2[4]/A18	97[4]	GPIO0, pin 22	ADC2 IN6	PWM2 MAT4	-
$V_{SS(IO)}$	98	ground for I/O			

Table 3. LQFP100 pin assignment *…continued*

Pin name	Pin	Description			
		Function 0 (default)	Function 1	Function 2	Function 3
P0[23]/IN2[7]/ PMAT2[5]/A19	99[4]	GPIO0, pin 23	ADC2 IN7	PWM2 MAT5	-
TDI	100[1]	IEEE 1149.1 data in, pulled up internally			

[1] Bidirectional pad; analog port; plain input; 3-state output; slew rate control; 5 V tolerant; TTL with hysteresis; programmable pull-up/pull-down/repeater.

[2] USB pad.

[3] Analog pad; analog I/O.

[4] Analog I/O pad.

6. Functional description

6.1 Architectural overview

The LPC2921/2923/2925 consists of:

- An ARM968E-S processor with real-time emulation support
- An AMBA multilayer Advanced High-performance Bus (AHB) for interfacing to the on-chip memory controllers
- Two DTL buses (an universal NXP interface) for interfacing to the interrupt controller and the Power, Clock and Reset control SubSystem (PCRSS).
- Three ARM Peripheral Buses (APB - a compatible super set of ARM's AMBA advanced peripheral bus) for connection to on-chip peripherals clustered in subsystems.
- One ARM Peripheral Bus for event router and system control.

The LPC2921/2923/2925 configures the ARM968E-S processor in little-endian byte order. All peripherals run at their own clock frequency to optimize the total system power consumption. The AHB-to-APB bridge used in the subsystems contains a write-ahead buffer one transaction deep. This implies that when the ARM968E-S issues a buffered write action to a register located on the APB side of the bridge, it continues even though the actual write may not yet have taken place. Completion of a second write to the same subsystem will not be executed until the first write is finished.

6.2 ARM968E-S processor

The ARM968E-S is a general purpose 32-bit RISC processor, which offers high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are much simpler than those of microprogrammed Complex Instruction Set Computers (CISC). This simplicity results in a high instruction throughput and impressive real-time interrupt response from a small and cost-effective controller core.

Amongst the most compelling features of the ARM968E-S are:

- Separate directly connected instruction and data Tightly Coupled Memory (TCM) interfaces.

- Write buffers for the AHB and TCM buses.
- Enhanced 16 × 32 multiplier capable of single-cycle MAC operations and 16-bit fixed-point DSP instructions to accelerate signal-processing algorithms and applications.

Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. The ARM968E-S is based on the ARMv5TE five-stage pipeline architecture. Typically, in a three-stage pipeline architecture, while one instruction is being executed its successor is being decoded and a third instruction is being fetched from memory. In the five-stage pipeline additional stages are added for memory access and write-back cycles.

The ARM968E-S processor also employs a unique architectural strategy known as THUMB, which makes it ideally suited to high-volume applications with memory restrictions or to applications where code density is an issue.

The key idea behind THUMB is that of a super-reduced instruction set. Essentially, the ARM968E-S processor has two instruction sets:

- Standard 32-bit ARMv5TE set
- 16-bit THUMB set

The THUMB set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit controller using 16-bit registers. This is possible because THUMB code operates on the same 32-bit register set as ARM code.

THUMB code can provide up to 65 % of the code size of ARM, and 160 % of the performance of an equivalent ARM controller connected to a 16-bit memory system.

The ARM968E-S processor is described in detail in the ARM968E-S data sheet Ref. 2.

6.3 On-chip flash memory system

The LPC2921/2923/2925 includes a 128 kB, 256 kB, or 512 kB flash memory system. This memory can be used for both code and data storage. Programming of the flash memory can be accomplished via the flash memory controller or the JTAG.

The flash controller also supports a 16 kB, byte-accessible on-chip EEPROM integrated on the LPC2921/2923/2925.

6.4 On-chip static RAM

In addition to the two 16 kB TCMs, the LPC2921/2923/2925 includes two static RAM memories of 16 kB each for a total of 32 kB (LPC2925 only) or one block of 16 kB (LPC2921/2923). They may be used for code and/or data storage.

The 8 kB SRAM block for the ETB can be used as static memory for code and data storage as well. However, DMA access to this memory region is not supported.

6.5 Memory map

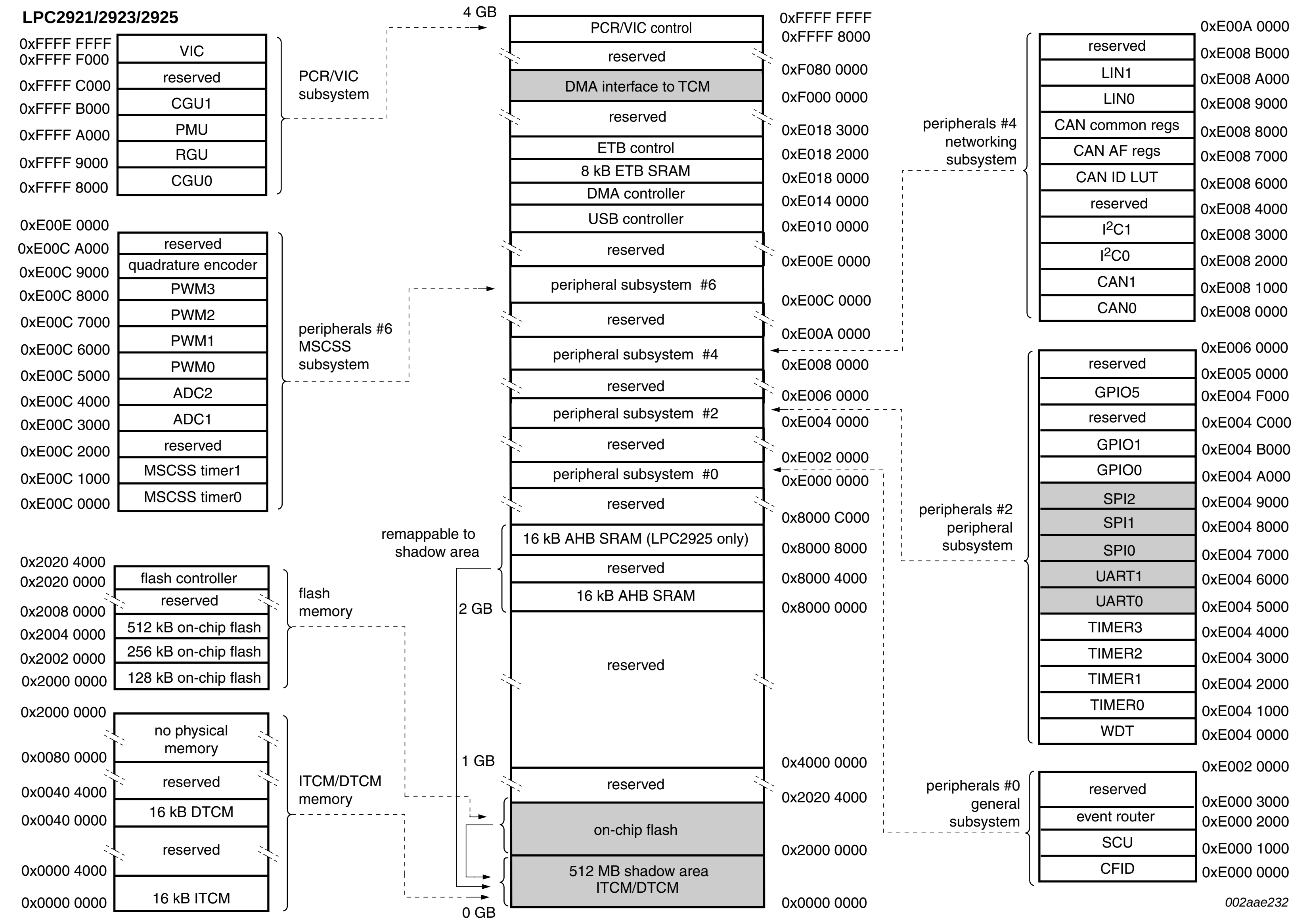


Fig 3. LPC2921/2923/2925 memory map

6.6 Reset, debug, test, and power description

6.6.1 Reset and power-up behavior

The LPC2921/2923/2925 contains external reset input and internal power-up reset circuits. This ensures that a reset is extended internally until the oscillators and flash have reached a stable state. See Section 8 for trip levels of the internal power-up reset circuit[1]. See Section 9 for characteristics of the several start-up and initialization times. Table 4 shows the reset pin.

Table 4. Reset pin

Symbol	Direction	Description
$\overline{\text{RST}}$	IN	external reset input, active LOW; pulled up internally

At activation of the $\overline{\text{RST}}$ pin the JTAGSEL pin is sensed as logic LOW. If this is the case the LPC2921/2923/2925 is assumed to be connected to debug hardware, and internal circuits re-program the source for the BASE_SYS_CLK to be the crystal oscillator instead of the Low-Power Ring Oscillator (LP_OSC). This is required because the clock rate when running at LP_OSC speed is too low for the external debugging environment.

6.6.2 Reset strategy

The LPC2921/2923/2925 contains a central module, the Reset Generator Unit (RGU) in the Power, Clock and Reset control Subsystem (PCRSS), which controls all internal reset signals towards the peripheral modules. The RGU provides individual reset control as well as the monitoring functions needed for tracing a reset back to source.

6.6.3 IEEE 1149.1 interface pins (JTAG boundary-scan test)

The LPC2921/2923/2925 contains boundary-scan test logic according to IEEE 1149.1, also referred to in this document as Joint Test Action Group (JTAG). The boundary-scan test pins can be used to connect a debugger probe for the embedded ARM processor. Pin JTAGSEL selects between boundary-scan mode and debug mode. Table 5 shows the boundary-scan test pins.

Table 5. IEEE 1149.1 boundary-scan test and debug interface

Symbol	Description
JTAGSEL	TAP controller select input. LOW level selects ARM debug mode and HIGH level selects boundary scan and flash programming; pulled up internally
$\overline{\text{TRST}}$	test reset input; pulled up internally (active LOW)
TMS	test mode select input; pulled up internally
TDI	test data input, pulled up internally
TDO	test data output
TCK	test clock input

---

[1] Only for 1.8 V power sources

6.6.3.1 ETM/ETB

The ETM provides real-time trace capability for deeply embedded processor cores. It outputs information about processor execution to a trace buffer. A software debugger allows configuration of the ETM using a JTAG interface and displays the trace information that has been captured in a format that a user can easily understand. The ETB stores trace data produced by the ETM.

The ETM/ETB module has the following features:

- Closely tracks the instructions that the ARM core is executing.
- On-chip trace data storage (ETB).
- All registers are programmed through JTAG interface.
- Does not consume power when trace is not being used.
- THUMB/Java instruction set support.

6.6.4 Power supply pins

Table 6 shows the power supply pins.

Table 6. Power supply pins

Symbol	Description
$V_{DD(CORE)}$	digital core supply 1.8 V
$V_{SS(CORE)}$	digital core ground (digital core, ADC1/2)
$V_{DD(IO)}$	I/O pins supply 3.3 V
$V_{SS(IO)}$	I/O pins ground
$V_{DD(OSC\_PLL)}$	oscillator and PLL supply
$V_{SS(OSC)}$	oscillator ground
$V_{SS(PLL)}$	PLL ground
$V_{DDA(ADC3V3)}$	ADC1 and ADC2 3.3 V supply

6.7 Clocking strategy

6.7.1 Clock architecture

The LPC2921/2923/2925 contains several different internal clock areas. Peripherals like timers, SPI, UART, CAN and LIN have their own individual clock sources called base clocks. All base clocks are generated by the Clock Generator Unit (CGU0). They may be unrelated in frequency and phase and can have different clock sources within the CGU.

The system clock for the CPU and AHB Bus infrastructure has its own base clock. This means most peripherals are clocked independently from the system clock. See Figure 4 for an overview of the clock areas within the device.

Within each clock area there may be multiple branch clocks, which offers very flexible control for power-management purposes. All branch clocks are outputs of the Power Management Unit (PMU) and can be controlled independently. Branch clocks derived from the same base clock are synchronous in frequency and phase. See Section 6.15 for more details of clock and power control within the device.

Two of the base clocks generated by the CGU0 are used as input into a second, dedicated CGU (CGU1). The CGU1 uses its own PLL and fractional dividers to generate the base clock for the USB controller and one base clock for an independent clock output.

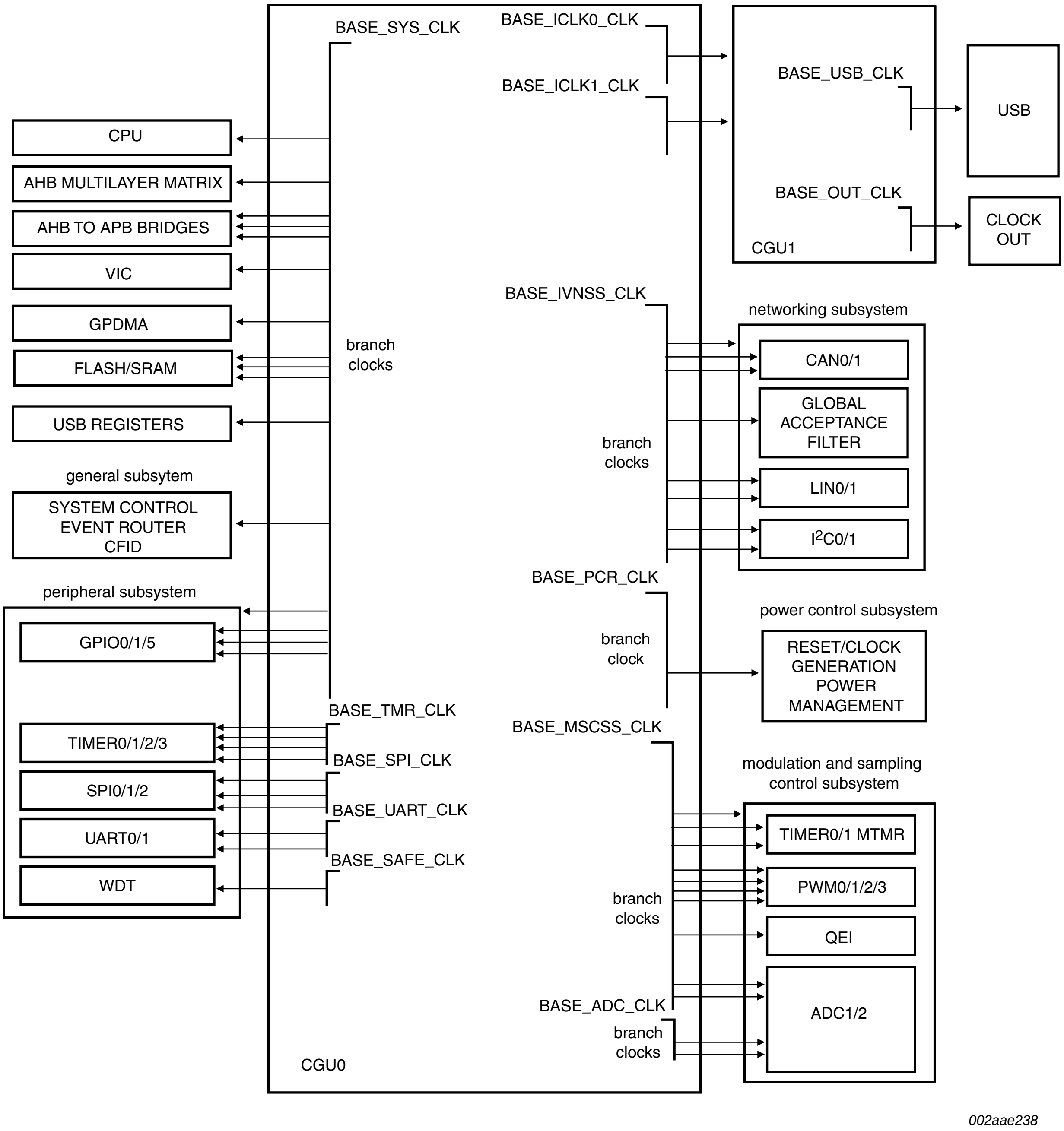


Fig 4. LPC2921/2923/2925 overview of clock areas

6.7.2 Base clock and branch clock relationship

Table 7 contains an overview of all the base blocks in the LPC2921/2923/2925 and their derived branch clocks. A short description is given of the hardware parts that are clocked with the individual branch clocks. In relevant cases more detailed information can be found in the specific subsystem description. Some branch clocks have special protection since they clock vital system parts of the device and should not be switched off. See Section 6.15.5 for more details of how to control the individual branch clocks.

Table 7. Base clock and branch clock overview

Base clock	Branch clock name	Parts of the device clocked by this branch clock	Remark
BASE_SAFE_CLK	CLK_SAFE	watchdog timer	[1]
BASE_SYS_CLK	CLK_SYS_CPU	ARM968E-S and TCMs	
	CLK_SYS_SYS	AHB bus infrastructure	
	CLK_SYS_PCRSS	AHB side of bridge in PCRSS	
	CLK_SYS_FMC	Flash Memory Controller	
	CLK_SYS_RAM0	Embedded SRAM Controller 0 (16 kB)	
	CLK_SYS_RAM1	Embedded SRAM Controller 1 (16 kB) (LPC2925 only)	
	CLK_SYS_GESS	General Subsystem	
	CLK_SYS_VIC	Vectored Interrupt Controller	
	CLK_SYS_PESS	Peripheral Subsystem	[2] [3]
	CLK_SYS_GPIO0	GPIO bank 0	
	CLK_SYS_GPIO1	GPIO bank 1	
	CLK_SYS_GPIO5	GPIO bank 5	
	CLK_SYS_IVNSS_A	AHB side of bridge of IVNSS	
	CLK_SYS_MSCSS_A	AHB side of bridge of MSCSS	
	CLK_SYS_DMA	GPDMA	
	CLK_SYS_USB	USB registers	
BASE_PCR_CLK	CLK_PCR_SLOW	PCRSS, CGU, RGU and PMU logic clock	[1] [4]
BASE_IVNSS_CLK	CLK_IVNSS_APB	APB side of the IVNSS	
	CLK_IVNSS_CANCA	CAN controller Acceptance Filter	
	CLK_IVNSS_CANC0	CAN channel 0	
	CLK_IVNSS_CANC1	CAN channel 1	
	CLK_IVNSS_I2C0	I^2C0	
	CLK_IVNSS_I2C1	I^2C1	
	CLK_IVNSS_LIN0	LIN channel 0	
	CLK_IVNSS_LIN1	LIN channel 1	

Table 7. Base clock and branch clock overview *…continued*

Base clock	Branch clock name	Parts of the device clocked by this branch clock	Remark
BASE_MSCSS_CLK	CLK_MSCSS_APB	APB side of the MSCSS	
	CLK_MSCSS_MTMR0	Timer 0 in the MSCSS	
	CLK_MSCSS_MTMR1	Timer 1 in the MSCSS	
	CLK_MSCSS_PWM0	PWM0	
	CLK_MSCSS_PWM1	PWM1	
	CLK_MSCSS_PWM2	PWM2	
	CLK_MSCSS_PWM3	PWM3	
	CLK_MSCSS_ADC1_APB	APB side of ADC1	
	CLK_MSCSS_ADC2_APB	APB side of ADC2	
	CLK_MSCSS_QEI	Quadrature encoder	
BASE_UART_CLK	CLK_UART0	UART0 interface clock	
	CLK_UART1	UART1 interface clock	
BASE_ICLK0_CLK	-	clock for CGU1 input	
BASE_SPI_CLK	CLK_SPI0	SPI0 interface clock	
	CLK_SPI1	SPI1 interface clock	
	CLK_SPI2	SPI2 interface clock	
BASE_TMR_CLK	CLK_TMR0	Timer 0 clock for counter part	
	CLK_TMR1	Timer 1 clock for counter part	
	CLK_TMR2	Timer 2 clock for counter part	
	CLK_TMR3	Timer 3 clock for counter part	
BASE_ADC_CLK	CLK_ADC1	Control of ADC1, capture sample result	
	CLK_ADC2	Control of ADC2, capture sample result	
reserved	-	-	
BASE_ICLK1_CLK	-	clock for CGU1 input	

[1] This clock is always on (cannot be switched off for system safety reasons)

[2] In the peripheral subsystem parts of the timers, watchdog timer, SPI and UART have their own clock source. See Section 6.12 for details.

[3] The clock should remain activated when system wake-up on timer or UART is required.

[4] In the Power, Clock and Reset control SubSystem (PCRSS) parts of the CGU, RGU, and PMU have their own clock source. See Section 6.15 for details.

Table 8. CGU1 base clock and branch clock overview

Base clock	Branch clock name	Parts of the device clocked by this branch clock
BASE_OUT_CLK	CLK_OUT_CLK	clock out pin
BASE_USB_CLK	CLK_USB_CLK	USB clock

6.8 Flash memory controller

The flash memory has a 128-bit wide data interface and the flash controller offers two 128-bit buffer lines to improve system performance. The flash has to be programmed initially via JTAG. In-system programming must be supported by the bootloader. Flash memory contents can be protected by disabling JTAG access. Suspension of burning or erasing is not supported.

The Flash Memory Controller (FMC) interfaces to the embedded flash memory for two tasks:

- Memory data transfer
- Memory configuration via triggering, programming, and erasing

The key features are:

- Programming by CPU via AHB
- Programming by external programmer via JTAG
- JTAG access protection
- Burn-finished and erase-finished interrupt

6.8.1 Functional description

After reset flash initialization is started. During this initialization, flash access is not possible and AHB transfers to flash are stalled, blocking the AHB bus.

During flash initialization, the index sector is read to identify the status of the JTAG access protection and sector security. If JTAG access protection is active, the flash is not accessible via JTAG. In this case, ARM debug facilities are disabled and flash memory contents cannot be read. If sector security is active, only the unsecured sections can be read.

Flash can be read synchronously or asynchronously to the system clock. In synchronous operation, the flash goes into standby after returning the read data. Started reads cannot be stopped, and speculative reading and dual buffering are therefore not supported.

With asynchronous reading, transfer of the address to the flash and of read data from the flash is done asynchronously, giving the fastest possible response time. Started reads can be stopped, so speculative reading and dual buffering are supported.

Buffering is offered because the flash has a 128-bit wide data interface while the AHB interface has only 32 bits. With buffering a buffer line holds the complete 128-bit flash word, from which four words can be read. Without buffering every AHB data port read starts a flash read. A flash read is a slow process compared to the minimum AHB cycle time, so with buffering the average read time is reduced improving system performance.

With single buffering, the most recently read flash word remains available until the next flash read. When an AHB data-port read transfer requires data from the same flash word as the previous read transfer, no new flash read is done and the read data is given without wait cycles.

When an AHB data port read transfer requires data from a different flash word to that involved in the previous read transfer, a new flash read is done and wait states are given until the new read data is available.

With dual buffering, a secondary buffer line is used, the output of the flash being considered as the primary buffer. On a primary buffer, hit data can be copied to the secondary buffer line, which allows the flash to start a speculative read of the next flash word.

Both buffer lines are invalidated after:

- Initialization
- Configuration-register access
- Data-latch reading
- Index-sector reading

The modes of operation are listed in Table 9.

Table 9. Flash read modes

Synchronous timing	
No buffer line	for single (non-linear) reads; one flash-word read per word read
Single buffer line	default mode of operation; most recently read flash word is kept until another flash word is required
Asynchronous timing	
No buffer line	one flash-word read per word read
Single buffer line	most recently read flash word is kept until another flash word is required
Dual buffer line, single speculative	on a buffer miss a flash read is done, followed by at most one speculative read; optimized for execution of code with small loops (less than eight words) from flash
Dual buffer line, always speculative	most recently used flash word is copied into second buffer line; next flash-word read is started; highest performance for linear reads

6.8.2 Flash layout

The ARM processor can program the flash for ISP (In-System Programming) through the flash memory controller. Note that the flash always has to be programmed by ‘flash words’ of 128 bits (four 32-bit AHB bus words, hence 16 bytes).

The flash memory is organized into eight ‘small’ sectors of 8 kB each and up to 11 ‘large’ sectors of 64 kB each. The number of large sectors depends on the device type. A sector must be erased before data can be written to it. The flash memory also has sector-wise protection. Writing occurs per page which consists of 4096 bits (32 flash words). A small sector contains 16 pages; a large sector contains 128 pages.

Table 10 gives an overview of the flash-sector base addresses.

Table 10. Flash sector overview

Flash memory sector number	Sector size (kB)	Flash memory address	LPC2921	LPC2923	LPC2925
11	8	0x2000 0000	yes	yes	yes
12	8	0x2000 2000	yes	yes	yes
13	8	0x2000 4000	yes	yes	yes
14	8	0x2000 6000	yes	yes	yes
15	8	0x2000 8000	yes	yes	yes

Table 10. Flash sector overview *…continued*

Flash memory sector number	Sector size (kB)	Flash memory address	LPC2921	LPC2923	LPC2925
16	8	0x2000 A000	yes	yes	yes
17	8	0x2000 C000	yes	yes	yes
18	8	0x2000 E000	yes	yes	yes
0	64	0x2001 0000	yes	yes	yes
1	64	0x2002 0000	no	yes	yes
2	64	0x2003 0000	no	yes	yes
3	64	0x2004 0000	no	no	yes
4	64	0x2005 0000	no	no	yes
5	64	0x2006 0000	no	no	yes
6	64	0x2007 0000	no	no	yes

The index sector is a special sector in which the JTAG access protection and sector security are located. The address space becomes visible by setting the FS_ISS bit and overlaps the regular flash sector's address space.

Note that the index sector, once programmed, cannot be erased. Any flash operation must be executed out of SRAM (internal or external).

6.8.3 Flash bridge wait-states

To eliminate the delay associated with synchronizing flash-read data, a predefined number of wait-states must be programmed. These depend on flash memory response time and system clock period. The minimum wait-states value can be calculated with the following formulas:

Synchronous reading:

$$WST > \frac{t_{acc(clk)}}{t_{t_{tclk(sys)}}} - 1 \quad (1)$$

Asynchronous reading:

$$WST > \frac{t_{acc(addr)}}{t_{tclk(sys)}} - 1 \quad (2)$$

Remark: If the programmed number of wait-states is more than three, flash-data reading cannot be performed at full speed (i.e. with zero wait-states at the AHB bus) if speculative reading is active.

6.8.4 Pin description

The flash memory controller has no external pins. However, the flash can be programmed via the JTAG pins, see Section 6.6.3.

6.8.5 Clock description

The flash memory controller is clocked by CLK_SYS_FMC, see Section 6.7.2.

6.8.6 EEPROM

EEPROM is a non-volatile memory mostly used for storing relatively small amounts of data, for example for storing settings. It contains one 16 kB memory block and is byte-programmable and byte-erasable.

The EEPROM can be accessed only through the flash controller.

6.9 General Purpose DMA (GPDMA) controller

The GPDMA controller allows peripheral-to-memory, memory-to-peripheral, peripheral-to-peripheral, and memory-to-memory transactions. Each DMA stream provides unidirectional serial DMA transfers for a single source and destination. For example, a bidirectional port requires one stream for transmit and one for receives. The source and destination areas can each be either a memory region or a peripheral, and can be accessed through the same AHB master or one area by each master.

The GPDMA controls eight DMA channels with hardware prioritization. The DMA controller interfaces to the system via two AHB bus masters, each with a full 32-bit data bus width. DMA operations may be set up for 8-bit, 16-bit, and 32-bit data widths, and can be either big-endian or little-endian. Incrementing or non-incrementing addressing for source and destination are supported, as well as programmable DMA burst size. Scatter or gather DMA is supported through the use of linked lists. This means that the source and destination areas do not have to occupy contiguous areas of memory.

6.9.1 DMA support for peripherals

The GPDMA supports the following peripherals: SPI0/1/2 and UART0/1. The GPDMA can access both embedded SRAM blocks, both TCMs, external static memory, and flash memory.

6.9.2 Clock description

The DMA controller is clocked by CLK_SYS_DMA derived from BASE_SYS_CLK, see Section 6.7.2.

6.10 USB interface

The Universal Serial Bus (USB) is a 4-wire bus that supports communication between a host and one or more (up to 127) peripherals. The bus supports hot plugging and dynamic configuration of the devices. All transactions are initiated by the Host controller.

The LPC2921/2923/2925 USB interface includes a device controller with on-chip PHY for device. Details on typical USB interfacing solutions can be found in Section 10.2.

6.10.1 USB device controller

The device controller enables 12 Mbit/s data exchange with a USB Host controller. It consists of a register interface, serial interface engine, endpoint buffer memory, and a DMA controller. The serial interface engine decodes the USB data stream and writes data to the appropriate endpoint buffer. The status of a completed USB transfer or error condition is indicated via status registers. An interrupt is also generated if enabled. When enabled, the DMA controller transfers data between the endpoint buffer and the on-chip SRAM.

The USB device controller has the following features:

- Fully compliant with *USB 2.0 specification (full speed)*.
- Supports 32 physical (16 logical) endpoints with a 2 kB endpoint buffer RAM.
- Supports Control, Bulk, Interrupt and Isochronous endpoints.
- Scalable realization of endpoints at run time.
- Endpoint Maximum packet size selection (up to USB maximum specification) by software at run time.
- Supports SoftConnect and GoodLink features.
- While USB is in the Suspend mode, the LPC2921/2923/2925 can enter the reduced power mode and wake up on USB activity.
- Supports DMA transfers with the on-chip SRAM blocks on all non-control endpoints.
- Allows dynamic switching between CPU-controlled slave and DMA modes.
- Double buffer implementation for Bulk and Isochronous endpoints.

6.10.2 Pin description

Table 11. USB device port pins

Pin name	Direction	Description
USB_VBUS	I	USB_VBUS status input. When this function is not enabled via its corresponding PINSEL register, it is driven HIGH internally.
USB_D+	I/O	positive differential data
USB_D–	I/O	negative differential data
$\overline{\mathrm{USB\_CONNECT}}$	O	SoftConnect control signal
$\overline{\mathrm{USB\_UP\_LED}}$	O	GoodLink LED control signal

6.10.3 Clock description

Access to the USB registers is clocked by the CLK_SYS_USB, derived from BASE_SYS_CLK, see Section 6.7.2. The CGU1 provides an independent base clock to the USB block, BASE_USB_CLK (see Section 6.15.3).

6.11 General subsystem

6.11.1 General subsystem clock description

The general subsystem is clocked by CLK_SYS_GESS, see Section 6.7.2.

6.11.2 Chip and feature identification

The Chip/Feature ID (CFID) module contains registers which show and control the functionality of the chip. It contains an ID to identify the silicon and also registers containing information about the features enabled or disabled on the chip.

The key features are:

- Identification of product
- Identification of features enabled

The CFID has no external pins.

6.11.3 System Control Unit (SCU)

The system control unit contains system-related functions.The key feature is configuration of the I/O port-pins multiplexer. It defines the function of each I/O pin of the LPC2921/2923/2925. The I/O pin configuration should be consistent with peripheral function usage.

The SCU has no external pins.

6.11.4 Event router

The event router provides bus-controlled routing of input events to the vectored interrupt controller for use as interrupt or wake-up signals.

Key features:

- Up to 16 level-sensitive external interrupt pins, including the receive pins of SPI, CAN, LIN, and UART, as well as the I^{2}C-bus SCL pins plus three internal event sources.
- Input events can be used as interrupt source either directly or latched (edge-detected).
- Direct events disappear when the event becomes inactive.
- Latched events remain active until they are explicitly cleared.
- Programmable input level and edge polarity.
- Event detection maskable.
- Event detection is fully asynchronous, so no clock is required.

The event router allows the event source to be defined, its polarity and activation type to be selected and the interrupt to be masked or enabled. The event router can be used to start a clock on an external event.

The vectored interrupt-controller inputs are active HIGH.

6.11.4.1 Pin description

The event router module in the LPC2921/2923/2925 is connected to the pins listed below. The pins are combined with other functions on the port pins of the LPC2921/2923/2925. Table 12 shows the pins connected to the event router.

Table 12. Event-router pin connections

Symbol	Direction	Description	Default polarity
EXTINT[0:3]	I	external interrupt inputs 0 to 3	1
CAN0 RXD	I	CAN0 receive data input wake-up	0
CAN1 RXD	I	CAN1 receive data input wake-up	0
I2C0_SCL	I	I^2C0 SCL clock input	0
I2C1_SCL	I	I^2C1 SCL clock input	0
LIN0 RXD	I	LIN0 receive data input wake-up	0
LIN1 RXD	I	LIN1 receive data input wake-up	0
SPI0 SDI	I	SPI0 receive data input	0
SPI1 SDI	I	SPI1 receive data input	0
SPI2 SDI	I	SPI2 receive data input	0
UART0 RXD	I	UART0 receive data input	0
UART1 RXD	I	UART1 receive data input	0
-	n/a	CAN interrupt (internal)	1
-	n/a	VIC FIQ (internal)	1
-	n/a	VIC IRQ (internal)	1

6.12 Peripheral subsystem

6.12.1 Peripheral subsystem clock description

The peripheral subsystem is clocked by a number of different clocks:

- CLK_SYS_PESS
- CLK_UART0/1
- CLK_SPI0/1/2
- CLK_TMR0/1/2/3
- CLK_SAFE see Section 6.7.2

6.12.2 Watchdog timer

The purpose of the watchdog timer is to reset the ARM9 processor within a reasonable amount of time if the processor enters an error state. The watchdog generates a system reset if the user program fails to trigger it correctly within a predetermined amount of time.

Key features:

- Internal chip reset if not periodically triggered
- Timer counter register runs on always-on safe clock
- Optional interrupt generation on watchdog time-out
- Debug mode with disabling of reset

- Watchdog control register change-protected with key
- Programmable 32-bit watchdog timer period with programmable 32-bit prescaler.

6.12.2.1 Functional description

The watchdog timer consists of a 32-bit counter with a 32-bit prescaler.

The watchdog should be programmed with a time-out value and then periodically restarted. When the watchdog times out, it generates a reset through the RGU.

To generate watchdog interrupts in watchdog debug mode the interrupt has to be enabled via the interrupt enable register. A watchdog-overflow interrupt can be cleared by writing to the clear-interrupt register.

Another way to prevent resets during debug mode is via the Pause feature of the watchdog timer. The watchdog is stalled when the ARM9 is in debug mode and the PAUSE_ENABLE bit in the watchdog timer control register is set.

The Watchdog Reset output is fed to the Reset Generator Unit (RGU). The RGU contains a reset source register to identify the reset source when the device has gone through a reset. See Section 6.15.4.

6.12.2.2 Clock description

The watchdog timer is clocked by two different clocks; CLK_SYS_PESS and CLK_SAFE, see Section 6.7.2. The register interface towards the system bus is clocked by CLK_SYS_PESS. The timer and prescale counters are clocked by CLK_SAFE which is always on.

6.12.3 Timer

The LPC2921/2923/2925 contains six identical timers: four in the peripheral subsystem and two in the Modulation and Sampling Control SubSystem (MSCSS) located at different peripheral base addresses. This section describes the four timers in the peripheral subsystem. Each timer has four capture inputs and/or match outputs. Connection to device pins depends on the configuration programmed into the port function-select registers. The two timers located in the MSCSS have no external capture or match pins, but the memory map is identical, see Section 6.14.6. One of these timers has an external input for a pause function.

The key features are:

- 32-bit timer/counter with programmable 32-bit prescaler.
- Up to four 32-bit capture channels per timer. These take a snapshot of the timer value when an external signal connected to the TIMERx CAPn input changes state. A capture event may also optionally generate an interrupt.
- Four 32-bit match registers per timer that allow:
  - Continuous operation with optional interrupt generation on match.
  - Stop timer on match with optional interrupt generation.
  - Reset timer on match with optional interrupt generation.
- Up to four external outputs per timer corresponding to match registers, with the following capabilities:
  - Set LOW on match.

  - Set HIGH on match.
  - Toggle on match.
  - Do nothing on match.
- Pause input pin (MSCSS timers only).

The timers are designed to count cycles of the clock and optionally generate interrupts or perform other actions at specified timer values, based on four match registers. They also include capture inputs to trap the timer value when an input signal changes state, optionally generating an interrupt. The core function of the timers consists of a 32 bit prescale counter triggering the 32 bit timer counter. Both counters run on clock CLK_TMRx (x runs from 0 to 3) and all time references are related to the period of this clock. Note that each timer has its individual clock source within the Peripheral SubSystem. In the Modulation and Sampling SubSystem each timer also has its own individual clock source. See Section 6.15.5 for information on generation of these clocks.

6.12.3.1 Pin description

The four timers in the peripheral subsystem of the LPC2921/2923/2925 have the pins described below. The two timers in the modulation and sampling subsystem have no external pins except for the pause pin on MSCSS timer 1. See Section 6.14.6 for a description of these timers and their associated pins. The timer pins are combined with other functions on the port pins of the LPC2921/2923/2925, see Section 6.11.3. Table Table 13 shows the timer pins (x runs from 0 to 3).

Table 13. Timer pins

Symbol	Pin name	Direction	Description
TIMERx CAP[0]	CAPx[0]	IN	TIMERx capture input 0[1]
TIMERx CAP[1]	CAPx[1]	IN	TIMERx capture input 1[1]
TIMERx CAP[2]	CAPx[2]	IN	TIMERx capture input 2
TIMERx CAP[3]	CAPx[3]	IN	TIMERx capture input 3
TIMERx MAT[0]	MATx[0]	OUT	TIMERx match output 0
TIMERx MAT[1]	MATx[1]	OUT	TIMERx match output 1
TIMERx MAT[2]	MATx[2]	OUT	TIMERx match output 2
TIMERx MAT[3]	MATx[3]	OUT	TIMERx match output 3

[1] Note that CAP1[0] and CAP1[1] are not pinned out on Timer1.

6.12.3.2 Clock description

The timer modules are clocked by two different clocks; CLK_SYS_PESS and CLK_TMRx (x = 0 to 3), see Section 6.7.2. Note that each timer has its own CLK_TMRx branch clock for power management. The frequency of all these clocks is identical as they are derived from the same base clock BASE_CLK_TMR. The register interface towards the system bus is clocked by CLK_SYS_PESS. The timer and prescale counters are clocked by CLK_TMRx.

6.12.4 UARTs

The LPC2921/2923/2925 contains two identical UARTs located at different peripheral base addresses. The key features are:

- 16-byte receive and transmit FIFOs.

- Register locations conform to 550 industry standard.
- Receiver FIFO trigger points at 1 byte, 4 bytes, 8 bytes and 14 bytes.
- Built-in baud rate generator.
- Support for RS-485/9-bit mode allows both software address detection and automatic address detection using 9-bit mode.

The UART is commonly used to implement a serial interface such as RS232. The LPC2921/2923/2925 contains two industry-standard 550 UARTs with 16-byte transmit and receive FIFOs, but they can also be put into 450 mode without FIFOs.

Remark: The LIN controller can be configured to provide two additional standard UART interfaces (see Section 6.13.2).

6.12.4.1 Pin description

The UART pins are combined with other functions on the port pins of the LPC2921/2923/2925. Table 14 shows the UART pins (x runs from 0 to 1).

Table 14. UART pins

Symbol	Pin name	Direction	Description
UARTx TXD	TXDx	OUT	UART channel x transmit data output
UARTx RXD	RXDx	IN	UART channel x receive data input

6.12.4.2 Clock description

The UART modules are clocked by two different clocks; CLK_SYS_PESS and CLK_UARTx (x = 0 to 1), see Section 6.7.2. Note that each UART has its own CLK_UARTx branch clock for power management. The frequency of all CLK_UARTx clocks is identical since they are derived from the same base clock BASE_CLK_UART. The register interface towards the system bus is clocked by CLK_SYS_PESS. The baud generator is clocked by the CLK_UARTx.

6.12.5 Serial Peripheral Interface (SPI)

The LPC2921/2923/2925 contains three Serial Peripheral Interface modules (SPIs) to allow synchronous serial communication with slave or master peripherals.

The key features are:

- Master or slave operation.
- Each SPI supports up to four slaves in sequential multi-slave operation.
- Supports timer-triggered operation.
- Programmable clock bit rate and prescale based on SPI source clock. (BASE_SPI_CLK), independent of system clock.
- Separate transmit and receive FIFO memory buffers; 16 bits wide, 32 locations deep.
- Programmable choice of interface operation: Motorola SPI or Texas Instruments Synchronous Serial Interfaces.
- Programmable data-frame size from 4 bits to 16 bits.
- Independent masking of transmit FIFO, receive FIFO and receive overrun interrupts.
- Serial clock-rate master mode: $fserial\_clk \leq f_{clk(SPI)} / 2$.
- Serial clock-rate slave mode: $fserial\_clk = f_{clk(SPI)} / 4$.

- Internal loopback test mode.

The SPI module can operate in:

- Master mode:
  - Normal transmission mode
  - Sequential slave mode
- Slave mode

6.12.5.1 Functional description

The SPI module is a master or slave interface for synchronous serial communication with peripheral devices that have either Motorola SPI or Texas Instruments Synchronous Serial Interfaces.

The SPI module performs serial-to-parallel conversion on data received from a peripheral device. The transmit and receive paths are buffered with FIFO memories (16 bits wide × 32 words deep). Serial data is transmitted on pins SDOx and received on pins SDIx.

The SPI module includes a programmable bit-rate clock divider and prescaler to generate the SPI serial clock from the input clock CLK_SPIx.

The SPI module’s operating mode, frame format, and word size are programmed through the SLVn_SETTINGS registers.

A single combined interrupt request SPI_INTREQ output is asserted if any of the interrupts are asserted and unmasked.

Depending on the operating mode selected, the SPI SCS outputs operate as an active-HIGH frame synchronization output for Texas Instruments synchronous serial frame format or an active-LOW chip select for SPI.

Each data frame is between four and 16 bits long, depending on the size of words programmed, and is transmitted starting with the MSB.

6.12.5.2 Pin description

The SPI pins are combined with other functions on the port pins of the LPC2921/2923/2925, see Section 6.11.3. Table 15 shows the SPI pins (x runs from 0 to 2; y runs from 0 to 3).

Table 15. SPI pins

Symbol	Pin name	Direction	Description
SPIx SCSy	SCSx[y]	IN/OUT	SPIx chip select[1][2]
SPIx SCK	SCKx	IN/OUT	SPIx clock[1]
SPIx SDI	SDIx	IN	SPIx data input
SPIx SDO	SDOx	OUT	SPIx data output

[1] Direction of SPIx SCS and SPIx SCK pins depends on master or slave mode. These pins are output in master mode, input in slave mode.

[2] In slave mode there is only one chip select input pin, SPIx SCS0. The other chip selects have no function in slave mode.

6.12.5.3 Clock description

The SPI modules are clocked by two different clocks; CLK_SYS_PESS and CLK_SPIx (x = 0, 1, 2), see Section 6.7.2. Note that each SPI has its own CLK_SPIx branch clock for power management. The frequency of all clocks CLK_SPIx is identical as they are derived from the same base clock BASE_CLK_SPI. The register interface towards the system bus is clocked by CLK_SYS_PESS. The serial-clock rate divisor is clocked by CLK_SPIx.

The SPI clock frequency can be controlled by the CGU. In master mode the SPI clock frequency (CLK_SPIx) must be set to at least twice the SPI serial clock rate on the interface. In slave mode CLK_SPIx must be set to four times the SPI serial clock rate on the interface.

6.12.6 General-purpose I/O

The LPC2921/2923/2925 contains two general-purpose I/O ports located at different peripheral base addresses. All I/O pins are bidirectional, and the direction can be programmed individually. The I/O pad behavior depends on the configuration programmed in the port function-select registers.

The key features are:

- General-purpose parallel inputs and outputs.
- Direction control of individual bits.
- Synchronized input sampling for stable input-data values.
- All I/O pins default to input at reset to avoid any possible bus conflicts.

6.12.6.1 Functional description

The general-purpose I/O provides individual control over each bidirectional port pin. There are two registers to control I/O direction and output level. The inputs are synchronized to achieve stable read-levels.

To generate an open-drain output, set the bit in the output register to the desired value. Use the direction register to control the signal. When set to output, the output driver actively drives the value on the output. When set to input, the signal floats and can be pulled up internally or externally.

6.12.6.2 Pin description

The five GPIO ports in the LPC2921/2923/2925 have the pins listed below. The GPIO pins are combined with other functions on the port pins of the LPC2921/2923/2925. Table 16 shows the GPIO pins.

Table 16. GPIO pins

Symbol	Pin name	Direction	Description
GPIO0 pin[31:0]	P0[31:0]	IN/OUT	GPIO port x pins 31 to 0
GPIO1 pin[27:0]	P1[27:0]	IN/OUT	GPIO port x pins 27 to 0
GPIO5 pin[19:18]	P5[19:18]	IN/OUT	GPIO port x pins 19 and 18

6.12.6.3 Clock description

The GPIO modules are clocked by several clocks, all of which are derived from BASE_SYS_CLK; CLK_SYS_PESS and CLK_SYS_GPIOx (x = 0, 1, 5), see Section 6.7.2. Note that each GPIO has its own CLK__SYS_GPIOx branch clock for power management. The frequency of all clocks CLK_SYS_GPIOx is identical to CLK_SYS_PESS since they are derived from the same base clock BASE_SYS_CLK.

6.13 Networking subsystem

6.13.1 CAN gateway

Controller Area Network (CAN) is the definition of a high-performance communication protocol for serial data communication. The two CAN controllers in the LPC2921/2923/2925 provide a full implementation of the CAN protocol according to the *CAN specification version 2.0B*. The gateway concept is fully scalable with the number of CAN controllers, and always operates together with a separate powerful and flexible hardware acceptance filter.

The key features are:

- Supports 11-bit as well as 29-bit identifiers.
- Double receive buffer and triple transmit buffer.
- Programmable error-warning limit and error counters with read/write access.
- Arbitration-lost capture and error-code capture with detailed bit position.
- Single-shot transmission (i.e. no re-transmission).
- Listen-only mode (no acknowledge; no active error flags).
- Reception of ‘own’ messages (self-reception request).
- FullCAN mode for message reception.

6.13.1.1 Global acceptance filter

The global acceptance filter provides look-up of received identifiers - called acceptance filtering in CAN terminology - for all the CAN controllers. It includes a CAN ID look-up table memory, in which software maintains one to five sections of identifiers. The CAN ID look-up table memory is 2 kB large (512 words, each of 32 bits). It can contain up to 1024 standard frame identifiers or 512 extended frame identifiers or a mixture of both types. It is also possible to define identifier groups for standard and extended message formats.

6.13.1.2 Pin description

The two CAN controllers in the LPC2921/2923/2925 have the pins listed below. The CAN pins are combined with other functions on the port pins of the LPC2921/2923/2925. Table 17 shows the CAN pins (x runs from 0 to 1).

Table 17. CAN pins

Symbol	Pin name	Direction	Description
CANx TXD	TXDC0/1	OUT	CAN channel x transmit data output
CANx RXD	RXDC0/1	IN	CAN channel x receive data input

6.13.2 LIN

The LPC2921/2923/2925 contain two LIN 2.0 master controllers. These can be used as dedicated LIN 2.0 master controllers with additional support for sync break generation and with hardware implementation of the LIN protocol according to spec 2.0.

The key features are:

- Complete LIN 2.0 message handling and transfer
- One interrupt per LIN message
- Slave response time-out detection
- Programmable sync-break length
- Automatic sync-field and sync-break generation
- Programmable inter-byte space
- Hardware or software parity generation
- Automatic checksum generation
- Fault confinement
- Fractional baud rate generator

6.13.2.1 Pin description

The two LIN 2.0 master controllers in the LPC2921/2923/2925 have the pins listed below. The LIN pins are combined with other functions on the port pins of the LPC2921/2923/2925. Table 18 shows the LIN pins. For more information see Ref. 1 *subsection 3.43, LIN master controller*.

Table 18. LIN controller pins

Symbol	Pin name	Direction	Description
LIN0/1 TXD	TXDL0/1	OUT	LIN channel 0/1 transmit data output
LIN0/1 RXD	RXDL0/1	IN	LIN channel 0/1 receive data input

Remark: Both LIN channels can be also configured as UART channels.

6.13.3 I^{2}C-bus serial I/O controllers

The LPC2921/2923/2925 each contain two I^{2}C-bus controllers.

The I^{2}C-bus is bidirectional for inter-IC control using only two wires: a Serial CLock line (SCL) and a Serial DAta line (SDA). Each device is recognized by a unique address and can operate as either a receiver-only device (e.g., an LCD driver) or as a transmitter with the capability to both receive and send information (such as memory). Transmitters and/or receivers can operate in either master or slave mode, depending on whether the chip has to initiate a data transfer or is only addressed. The I^{2}C is a multi-master bus, and it can be controlled by more than one bus master connected to it.

The main features if the I^{2}C-bus interfaces are:

- I^{2}C0 and I^{2}C1 use standard I/O pins with bit rates of up to 400 kbit/s (Fast I^{2}C-bus) and do not support powering off of individual devices connected to the same bus lines.
- Easy to configure as master, slave, or master/slave.

- Programmable clocks allow versatile rate control.
- Bidirectional data transfer between masters and slaves.
- Multi-master bus (no central master).
- Arbitration between simultaneously transmitting masters without corruption of serial data on the bus.
- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus.
- Serial clock synchronization can be used as a handshake mechanism to suspend and resume serial transfer.
- The I^{2}C-bus can be used for test and diagnostic purposes.
- All I^{2}C-bus controllers support multiple address recognition and a bus monitor mode.

6.13.3.1 Pin description

Table 19. I^{2}C-bus pins[1]

Symbol	Pin name	Direction	Description
I2C SCL0/1	SCL0/1	I/O	I^{2}C clock input/output
I2C SDA0/1	SDA0/1	I/O	I^{2}C data input/output

[1] Note that the pins are not I^{2}C-bus compliant open-drain pins.

6.14 Modulation and Sampling Control SubSystem (MSCSS)

The Modulation and Sampling Control Subsystem (MSCSS) in the LPC2921/2923/2925 includes four Pulse Width Modulators (PWMs), two 10-bit successive approximation Analog-to-Digital Converters (ADCs) and two timers.

The key features of the MSCSS are:

- Two 10-bit, 400 ksample/s, 8-channel ADCs with 3.3 V inputs and various trigger-start options.
- Four 6-channel PWMs (Pulse Width Modulators) with capture and trap functionality.
- Two dedicated timers to schedule and synchronize the PWMs and ADCs.
- Quadrature encoder interface.

6.14.1 Functional description

The MSCSS contains Pulse Width Modulators (PWMs), Analog-to-Digital Converters (ADCs) and timers.

Figure 5 provides an overview of the MSCSS. An AHB-to-APB bus bridge takes care of communication with the AHB system bus. Two internal timers are dedicated to this subsystem. MSCSS timer 0 can be used to generate start pulses for the ADCs and the first PWM. The second timer (MSCSS timer 1) is used to generate ‘carrier’ signals for the PWMs. These carrier patterns can be used, for example, in applications requiring current control. Several other trigger possibilities are provided for the ADCs (external, cascaded or following a PWM). The capture inputs of both timers can also be used to capture the start pulse of the ADCs.

The PWMs can be used to generate waveforms in which the frequency, duty cycle and rising and falling edges can be controlled very precisely. Capture inputs are provided to measure event phases compared to the main counter. Depending on the applications, these inputs can be connected to digital sensor motor outputs or digital external signals. Interrupt signals are generated on several events to closely interact with the CPU.

The ADCs can be used for any application needing accurate digitized data from analog sources. To support applications like motor control, a mechanism to synchronize several PWMs and ADCs is available (sync_in and sync_out).

Note that the PWMs run on the PWM clock and the ADCs on the ADC clock, see Section 6.15.2.

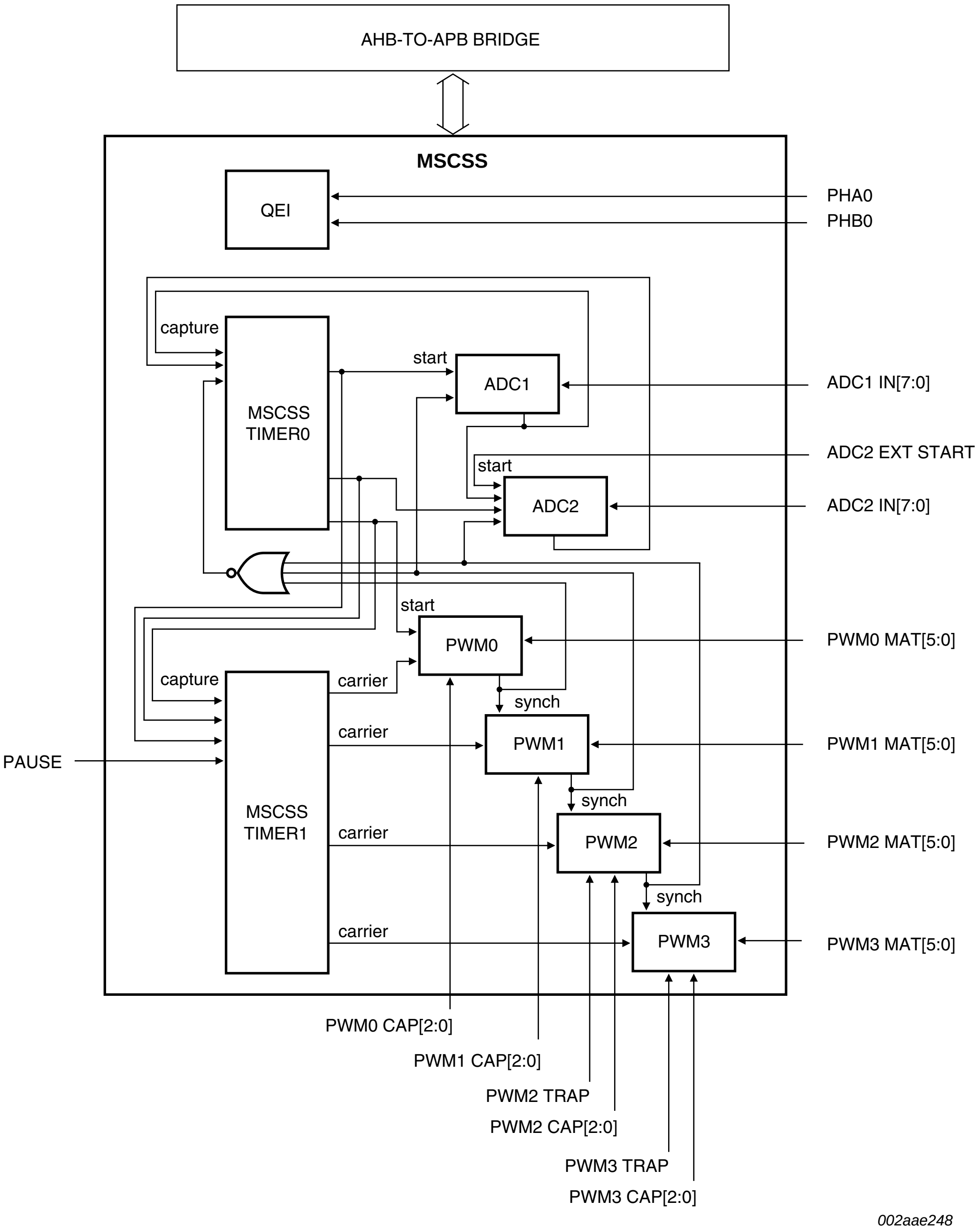


Fig 5. Modulation and Sampling Control SubSystem (MSCSS) block diagram

6.14.2 Pin description

The pins of the LPC2921/2923/2925 MSCSS associated with the two ADC modules are described in Section 6.14.4.2. Pins connected to the four PWM modules are described in Section 6.14.5.4, pins directly connected to the MSCSS timer 1 module are described in Section 6.14.6.1, and pins connected to the quadrature encoder interface are described in Section 6.14.7.1.

Remark: The IDX0 function for the QEI, the external start function for ADC1, and the TRAP0/1 functions for the PWM0/1 are not pinned out on the LPC2921/2923/2925.

6.14.3 Clock description

The MSCSS is clocked from a number of different sources:

- CLK_SYS_MSCSS_A clocks the AHB side of the AHB-to-APB bus bridge
- CLK_MSCSS_APB clocks the subsystem APB bus
- CLK_MSCSS_MTMR0/1 clocks the timers
- CLK_MSCSS_PWM[0:3] clocks the PWMs.

Each ADC has two clock areas; an APB part clocked by CLK_MSCSS_ADCx_APB (x = 1 or 2) and a control part for the analog section clocked by CLK_ADCx = 1 or 2), see Section 6.7.2.

All clocks are derived from the BASE_MSCSS_CLK, except for CLK_SYS_MSCSS_A which is derived form BASE_SYS_CLK, and the CLK_ADCx clocks which are derived from BASE_CLK_ADC. If specific PWM or ADC modules are not used their corresponding clocks can be switched off.

6.14.4 Analog-to-digital converter

The MSCSS in the LPC2921/2923/2925 includes two 10-bit successive-approximation analog-to-digital converters.

The key features of the ADC interface module are:

- ADC1 and ADC2: Eight analog inputs; time-multiplexed; measurement range up to 3.3 V.
- External reference-level inputs.
- 400 ksamples per second at 10-bit resolution up to 1500 ksamples per second at 2-bit resolution.
- Programmable resolution from 2-bit to 10-bit.
- Single analog-to-digital conversion scan mode and continuous analog-to-digital conversion scan mode.
- Optional conversion on transition on external start input, timer capture/match signal, PWM_sync or ‘previous’ ADC.
- Converted digital values are stored in a register for each channel.
- Optional compare condition to generate a ‘less than’ or an ‘equal to or greater than’ compare-value indication for each channel.
- Power-down mode.

6.14.4.1 Functional description

The ADC block diagram, Figure 6, shows the basic architecture of each ADC. The ADC functionality is divided into two major parts; one part running on the MSCSS Subsystem clock, the other on the ADC clock. This split into two clock domains affects the behavior from a system-level perspective. The actual analog-to-digital conversions take place in the ADC clock domain, but system control takes place in the system clock domain.

A mechanism is provided to modify configuration of the ADC and control the moment at which the updated configuration is transferred to the ADC domain.

The ADC clock is limited to 4.5 MHz maximum frequency and should always be lower than or equal to the system clock frequency. To meet this constraint or to select the desired lower sampling frequency, the clock generation unit provides a programmable fractional system-clock divider dedicated to the ADC clock. Conversion rate is determined by the ADC clock frequency divided by the number of resolution bits plus one. Accessing ADC registers requires an enabled ADC clock, which is controllable via the clock generation unit, see Section 6.15.2.

Each ADC has four start inputs. Note that start 0 and start 2 are captured in the system clock domain while start 1 and start 3 are captured in the ADC domain. The start inputs are connected at MSCSS level, see Section 6.14 for details.

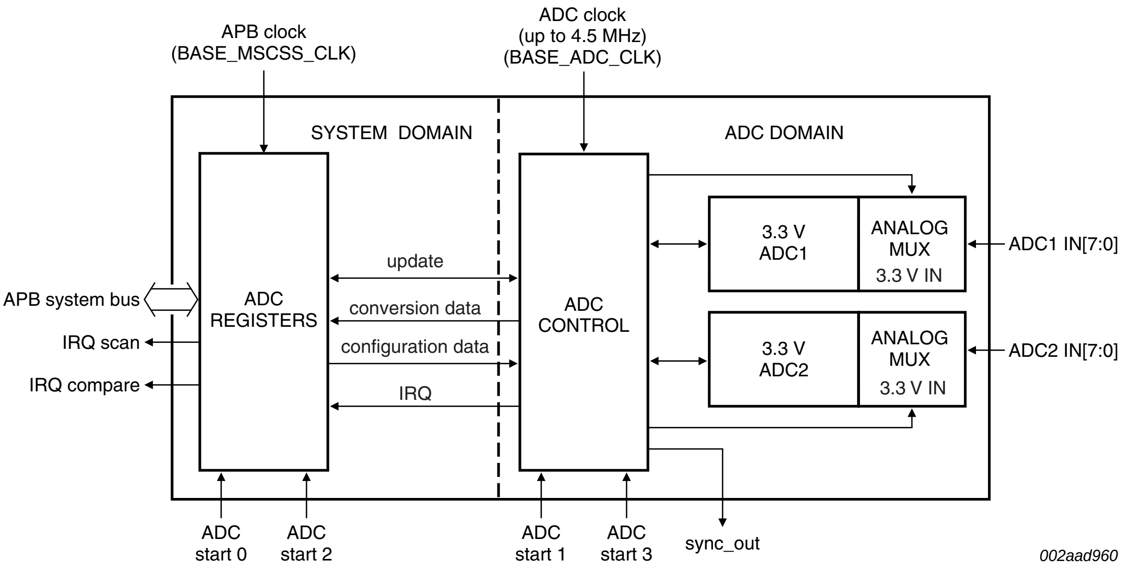


Fig 6. ADC block diagram

6.14.4.2 Pin description

The two ADC modules in the MSCSS have the pins described below. The ADCx input pins are combined with other functions on the port pins of the LPC2921/2923/2925. The VREFN and VREFP pins are common for both ADCs. Table 20 shows the ADC pins.

Table 20. Analog to digital converter pins

Symbol	Pin name	Direction	Description
ADC1/2 IN[7:0]	IN1/2[7:0]	IN	analog input for 3.3 V ADC1/2, channel 7 to channel 0
ADC2_EXT_START	CAP1[2]	IN	ADC external start-trigger input
VREFN	VREFN	IN	ADC LOW reference level
VREFP	VREFP	IN	ADC HIGH reference level
$V_{DDA(ADC3V3)}$	$V_{DDA(ADC3V3)}$	IN	ADC1 and ADC2 3.3 V supply

Remark: Note that the ADC1 and ADC2 accept an input voltage up to of 3.6 V (see Table 31) on the ADC1/2 IN pins. If the ADC is not used, the pins are 5 V tolerant.

6.14.4.3 Clock description

The ADC modules are clocked from two different sources; CLK_MSCSS_ADCx_APB and CLK_ADCx (x = 1 or 2), see Section 6.7.2. Note that each ADC has its own CLK_ADCx and CLK_MSCSS_ADCx_APB branch clocks for power management. If an ADC is unused both its CLK_MSCSS_ADCx_APB and CLK_ADCx can be switched off.

The frequency of all the CLK_MSCSS_ADCx_APB clocks is identical to CLK_MSCSS_APB since they are derived from the same base clock BASE_MSCSS_CLK. Likewise the frequency of all the CLK_ADCx clocks is identical since they are derived from the same base clock BASE_ADC_CLK.

The register interface towards the system bus is clocked by CLK_MSCSS_ADCx_APB. Control logic for the analog section of the ADC is clocked by CLK_ADCx, see also Figure 6.

6.14.5 Pulse Width Modulator (PWM)

The MSCSS in the LPC2921/2923/2925 includes four PWM modules with the following features.

- Six pulse width modulated output signals
- Double edge features (rising and falling edges programmed individually)
- Optional interrupt generation on match (each edge)
- Different operation modes: continuous or run-once
- 16-bit PWM counter and 16-bit prescale counter allow a large range of PWM periods
- A protective mode (TRAP) holding the output in a software-controllable state and with optional interrupt generation on a trap event
- Three capture registers and capture trigger pins with optional interrupt generation on a capture event
- Interrupt generation on match event, capture event, PWM counter overflow or trap event
- A burst mode mixing the external carrier signal with internally generated PWM
- Programmable sync-delay output to trigger other PWM modules (master/slave behavior)

6.14.5.1 Functional description

The ability to provide flexible waveforms allows PWM blocks to be used in multiple applications; e.g. dimmer/lamp control and fan control. Pulse width modulation is the preferred method for regulating power since no additional heat is generated, and it is energy-efficient when compared with linear-regulating voltage control networks.

The PWM delivers the waveforms/pulses of the desired duty cycles and cycle periods. A very basic application of these pulses can be in controlling the amount of power transferred to a load. Since the duty cycle of the pulses can be controlled, the desired amount of power can be transferred for a controlled duration. Two examples of such applications are:

- Dimmer controller: The flexibility of providing waves of a desired duty cycle and cycle period allows the PWM to control the amount of power to be transferred to the load. The PWM functions as a dimmer controller in this application.

- Motor controller: The PWM provides multi-phase outputs, and these outputs can be controlled to have a certain pattern sequence. In this way the force/torque of the motor can be adjusted as desired. This makes the PWM function as a motor drive.

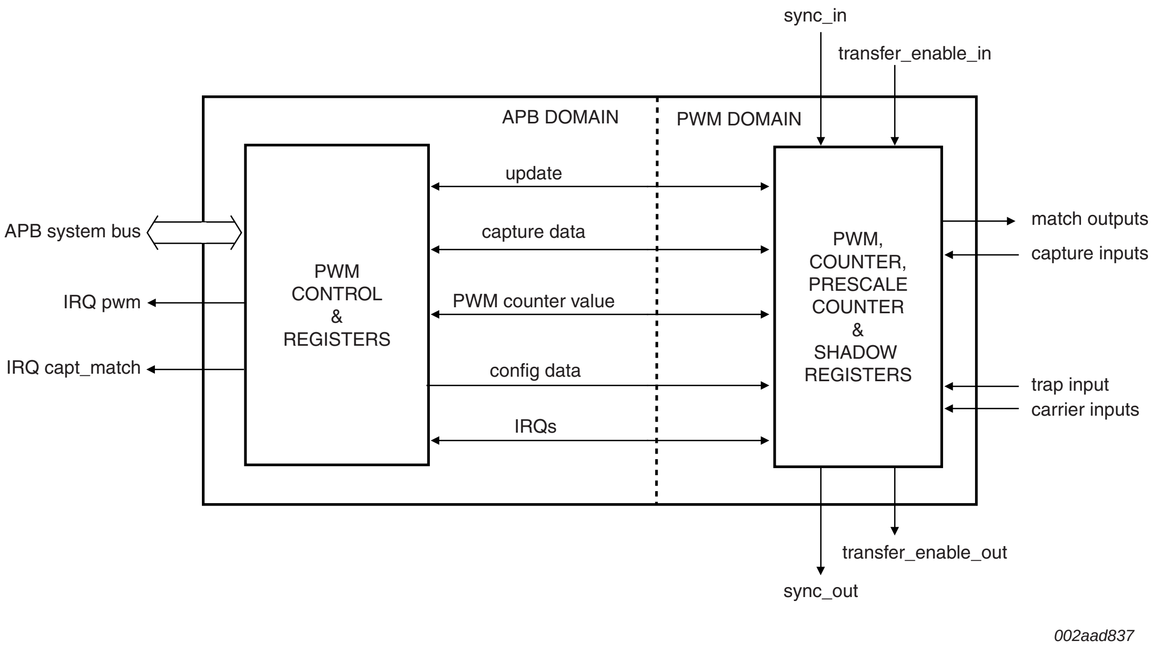


Fig 7. PWM block diagram

The PWM block diagram in Figure 7 shows the basic architecture of each PWM. PWM functionality is split into two major parts, a APB domain and a PWM domain, both of which run on clocks derived from the BASE_MSCSS_CLK. This split into two domains affects behavior from a system-level perspective. The actual PWM and prescale counters are located in the PWM domain but system control takes place in the APB domain.

The actual PWM consists of two counters; a 16-bit prescale counter and a 16-bit PWM counter. The position of the rising and falling edges of the PWM outputs can be programmed individually. The prescale counter allows high system bus frequencies to be scaled down to lower PWM periods. Registers are available to capture the PWM counter values on external events.

Note that in the Modulation and Sampling Control SubSystem (MSCSS), each PWM has its individual clock source CLK_MSCSS_PWMx (x runs from 0 to 3). Both the prescale and the timer counters within each PWM run on this clock CLK_MSCSS_PWMx, and all time references are related to the period of this clock. See Section 6.15 for information on generation of these clocks.

6.14.5.2 Synchronizing the PWM counters

A mechanism is included to synchronize the PWM period to other PWMs by providing a sync input and a sync output with programmable delay. Several PWMs can be synchronized using the trans_enable_in/trans_enable_out and sync_in/sync_out ports. See Figure 5 for details of the connections of the PWM modules within the MSCSS in the LPC2921/2923/2925. PWM0 can be master over PWM1; PWM1 can be master over PWM2, etc.

6.14.5.3 Master and slave mode

A PWM module can provide synchronization signals to other modules (also called Master mode). The signal sync_out is a pulse of one clock cycle generated when the internal PWM counter (re)starts. The signal trans_enable_out is a pulse synchronous to sync_out, generated if a transfer from system registers to PWM shadow registers occurred when the PWM counter restarted. A delay may be inserted between the counter start and generation of trans_enable_out and sync_out.

A PWM module can use input signals trans_enable_in and sync_in to synchronize its internal PWM counter and the transfer of shadow registers (Slave mode).

6.14.5.4 Pin description

Each of the four PWM modules in the MSCSS has the following pins. These are combined with other functions on the port pins of the LPC2921/2923/2925. Table 21 shows the PWM0 to PWM3 pins (n = 0 to 3).

Table 21. PWM pins

Symbol	Pin name	Direction	Description
PWMn CAP[0]	PCAPn[0]	IN	PWMn capture input 0
PWMn CAP[1]	PCAPn[1]	IN	PWMn capture input 1
PWMn CAP[2]	PCAPn[2]	IN	PWMn capture input 2
PWMn MAT[0]	PMATn[0]	OUT	PWMn match output 0
PWMn MAT[1]	PMATn[1]	OUT	PWMn match output 1
PWMn MAT[2]	PMATn[2]	OUT	PWMn match output 2
PWMn MAT[3]	PMATn[3]	OUT	PWMn match output 3
PWMn MAT[4]	PMATn[4]	OUT	PWMn match output 4
PWMn MAT[5]	PMATn[5]	OUT	PWMn match output 5
PWMn TRAP	TRAPn	IN	PWMn trap input (on the LPC2921/2923/2925 n = 2, 3)

6.14.5.5 Clock description

The PWM modules are clocked by CLK_MSCSS_PWMx (x = 0 to 3), see Section 6.7.2. Note that each PWM has its own CLK_MSCSS_PWMx branch clock for power management. The frequency of all these clocks is identical to CLK_MSCSS_APB since they are derived from the same base clock BASE_MSCSS_CLK.

Also note that unlike the timer modules in the Peripheral SubSystem, the actual timer counter registers of the PWM modules run at the same clock as the APB system interface CLK_MSCSS_APB. This clock is independent of the AHB system clock.

If a PWM module is not used its CLK_MSCSS_PWMx branch clock can be switched off.

6.14.6 Timers in the MSCSS

The two timers in the MSCSS are functionally identical to the timers in the peripheral subsystem, see Section 6.12.3. The features of the timers in the MSCSS are the same as the timers in the peripheral subsystem, but the capture inputs and match outputs are not available on the device pins. These signals are instead connected to the ADC and PWM modules as outlined in the description of the MSCSS, see Section 6.14.1.

See Section 6.12.3 for a functional description of the timers.

6.14.6.1 Pin description

MSCSS timer 0 has no external pins.

MSCSS timer 1 has a PAUSE pin available as external pin. The PAUSE pin is combined with other functions on the port pins of the LPC2921/2923/2925. Table 22 shows the MSCSS timer 1 external pin.

Table 22. MSCSS timer 1 pin

Symbol	Direction	Description
MSCSS PAUSE	IN	pause pin for MSCSS timer 1

6.14.6.2 Clock description

The timer modules in the MSCSS are clocked by CLK_MSCSS_MTMRx (x = 0 to 1), see Section 6.7.2. Note that each timer has its own CLK_MSCSS_MTMRx branch clock for power management. The frequency of all these clocks is identical to CLK_MSCSS_APB since they are derived from the same base clock BASE_MSCSS_CLK.

Note that, unlike the timer modules in the Peripheral SubSystem, the actual timer counter registers run at the same clock as the APB system interface CLK_MSCSS_APB. This clock is independent of the AHB system clock.

If a timer module is not used its CLK_MSCSS_MTMRx branch clock can be switched off.

6.14.7 Quadrature Encoder Interface (QEI)

A quadrature encoder, also known as a 2-channel incremental encoder, converts angular displacement into two pulse signals. By monitoring both the number of pulses and the relative phase of the two signals, the user can track the position, direction of rotation, and velocity. In addition, a third channel, or index signal, can be used to reset the position counter. The quadrature encoder interface decodes the digital pulses from a quadrature encoder wheel to integrate position over time and determine direction of rotation. In addition, the QEI can capture the velocity of the encoder wheel.

The QEI has the following features:

- Tracks encoder position.
- Increments/decrements depending on direction.
- Programmable for 2× or 4× position counting.
- Velocity capture using built-in timer.
- Velocity compare function with less than interrupt.
- Uses 32-bit registers for position and velocity.
- Three position compare registers with interrupts.
- Index counter for revolution counting.
- Index compare register with interrupts.
- Can combine index and position interrupts to produce an interrupt for whole and partial revolution displacement.
- Digital filter with programmable delays for encoder input signals.
- Can accept decoded signal inputs (clk and direction).
- Connected to APB.

6.14.7.1 Pin description

The QEI module in the MSCSS has the following pins. These are combined with other functions on the port pins of the LPC2921/2923/2925. Table 23 shows the QEI pins.

Table 23. QEI pins

Symbol	Pin name	Direction	Description
QEI0 PHA	PHA0	IN	Sensor signal. Corresponds to PHA in quadrature mode and to direction in clock/direction mode.
QEI0 PHB	PHB0	IN	Sensor signal. Corresponds to PHB in quadrature mode and to clock signal in clock/direction mode.

Remark: The index function for the QEI is not pinned out on the LPC2921/2923/2925.

6.14.7.2 Clock description

The QEI module is clocked by CLK_MSCSS_QEI, see Section 6.7.2. The frequency of this clock is identical to CLK_MSCSS_APB since they are derived from the same base clock BASE_MSCSS_CLK.

If the QEI is not used its CLK_MSCSS_QEI branch clock can be switched off.

6.15 Power, Clock, and Reset control SubSystem (PCRSS)

The Power, Clock, and Reset control SubSystem (PCRSS) in the LPC2921/2923/2925 includes a Clock Generator Unit (CGU), a Reset Generator Unit (RGU) and a Power Management Unit (PMU).

Figure 8 provides an overview of the PCRSS. An AHB-to-DTL bridge controls the communication with the AHB system bus.

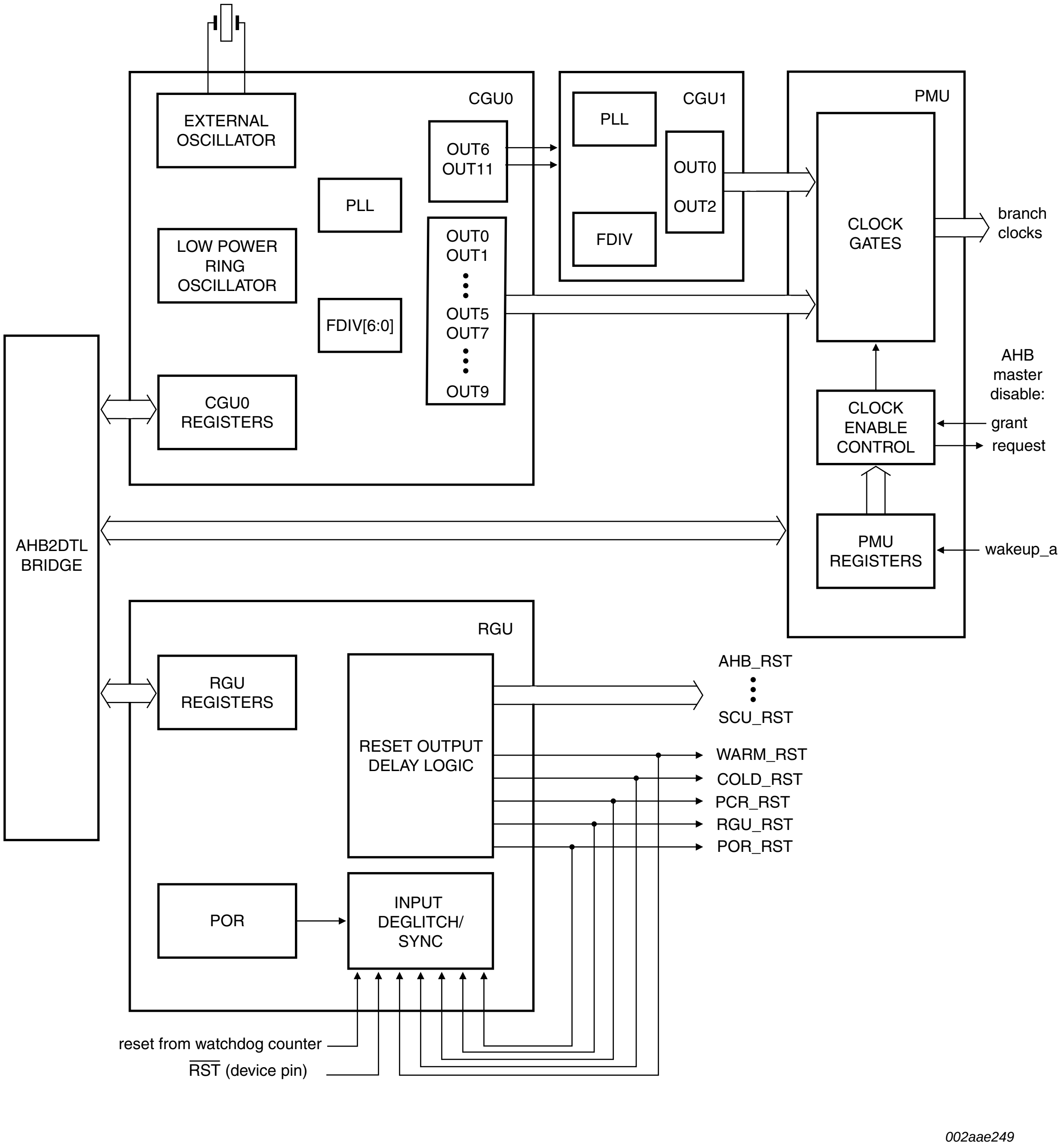


Fig 8. Power, Clock, and Reset control SubSystem (PCRSS) block diagram

6.15.1 Clock description

The PCRSS is clocked by a number of different clocks. CLK_SYS_PCRSS clocks the AHB side of the AHB to DTL bus bridge and CLK_PCR_SLOW clocks the CGU, RGU and PMU internal logic, see Section 6.7.2. CLK_SYS_PCRSS is derived from BASE_SYS_CLK, which can be switched off in low-power modes. CLK_PCR_SLOW is derived from BASE_PCR_CLK and is always on in order to be able to wake up from low-power modes.

6.15.2 Clock Generation Unit (CGU0)

The key features are:

- Generation of 11 base clocks selectable from several embedded clock sources.
- Crystal oscillator with power-down.
- Control PLL with power-down.
- Very low-power ring oscillator, always on to provide a safe clock.
- Seven fractional clock dividers with L/D division.
- Individual source selector for each base clock, with glitch-free switching.
- Autonomous clock-activity detection on every clock source.
- Protection against switching to invalid or inactive clock sources.
- Embedded frequency counter.
- Register write-protection mechanism to prevent unintentional alteration of clocks.

Remark: Any clock-frequency adjustment has a direct impact on the timing of all on-board peripherals.

6.15.2.1 Functional description

The clock generation unit provides 11 internal clock sources as described in Table 24.

Table 24. CGU0 base clocks

Number	Name	Frequency (MHz) [1]	Description
0	BASE_SAFE_CLK	0.4	base safe clock (always on)
1	BASE_SYS_CLK	125	base system clock
2	BASE_PCR_CLK	0.4 [2]	base PCR subsystem clock
3	BASE_IVNSS_CLK	125	base IVNSS subsystem clock
4	BASE_MSCSS_CLK	125	base MSCSS subsystem clock
5	BASE_ICLK0_CLK	125	base internal clock 0, for CGU1
6	BASE_UART_CLK	125	base UART clock
7	BASE_SPI_CLK	50	base SPI clock
8	BASE_TMR_CLK	125	base timers clock
9	BASE_ADC_CLK	4.5	base ADCs clock
10	reserved	-	-
11	BASE_ICLK1_CLK	125	base internal clock 1, for CGU1

[1] Maximum frequency that guarantees stable operation of the LPC2921/2923/2925.

[2] Fixed to low-power oscillator.

For generation of these base clocks, the CGU consists of primary and secondary clock generators and one output generator for each base clock.

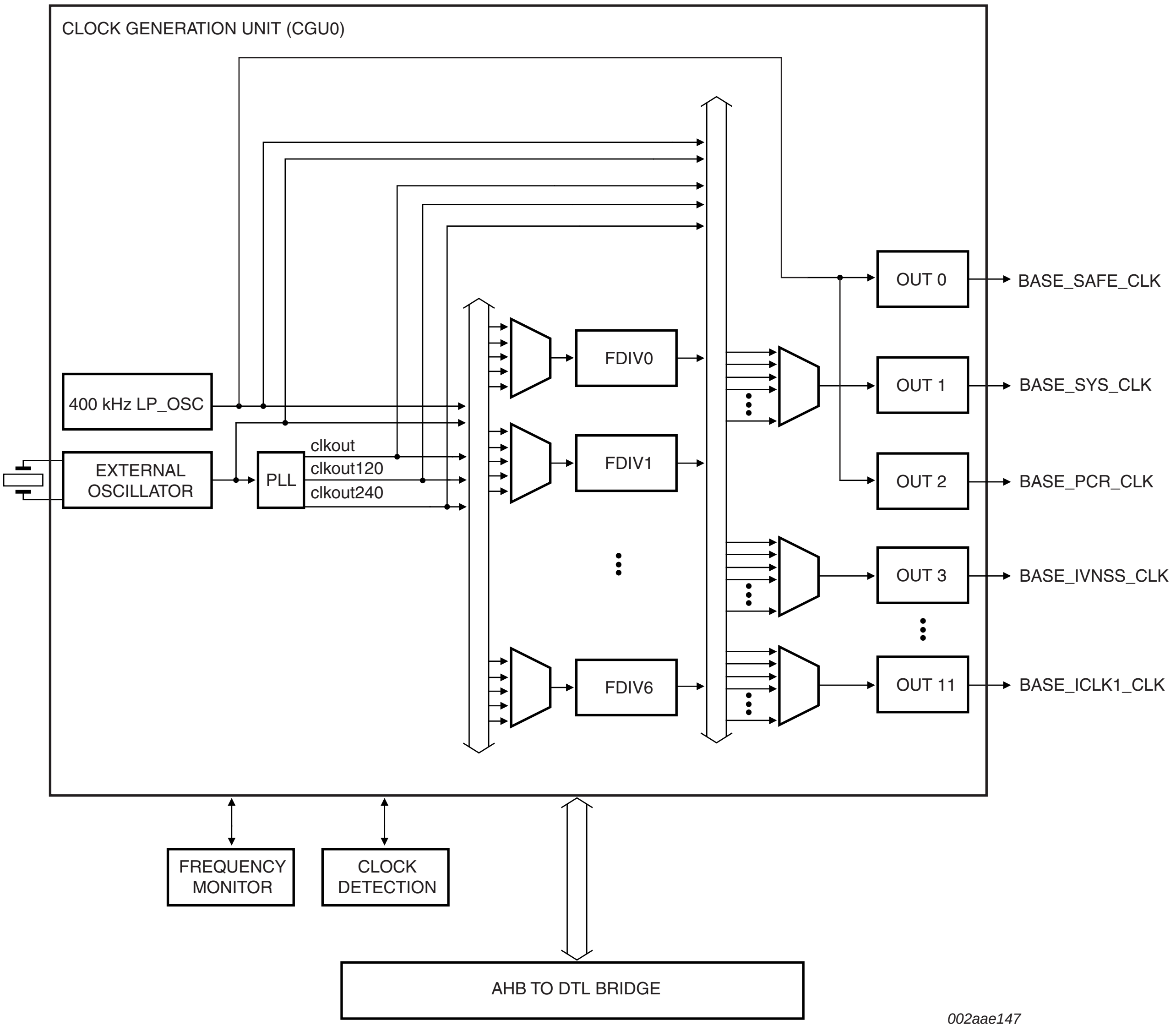


Fig 9. **Block diagram of the CGU0 (see Table 24 for all base clocks)**

There are two primary clock generators: a low-power ring oscillator (LP_OSC) and a crystal oscillator. See Figure 9.

LP_OSC is the source for the BASE_PCR_CLK that clocks the CGU itself and for BASE_SAFE_CLK that clocks a minimum of other logic in the device (like the watchdog timer). To prevent the device from losing its clock source LP_OSC cannot be put into power-down. The crystal oscillator can be used as source for high-frequency clocks or as an external clock input if a crystal is not connected.

Secondary clock generators are a PLL and seven fractional dividers (FDIV[0:6]). The PLL has three clock outputs: normal, 120° phase-shifted and 240° phase-shifted.

Configuration of the CGU0: For every output generator generating the base clocks a choice can be made from the primary and secondary clock generators according to Figure 10.

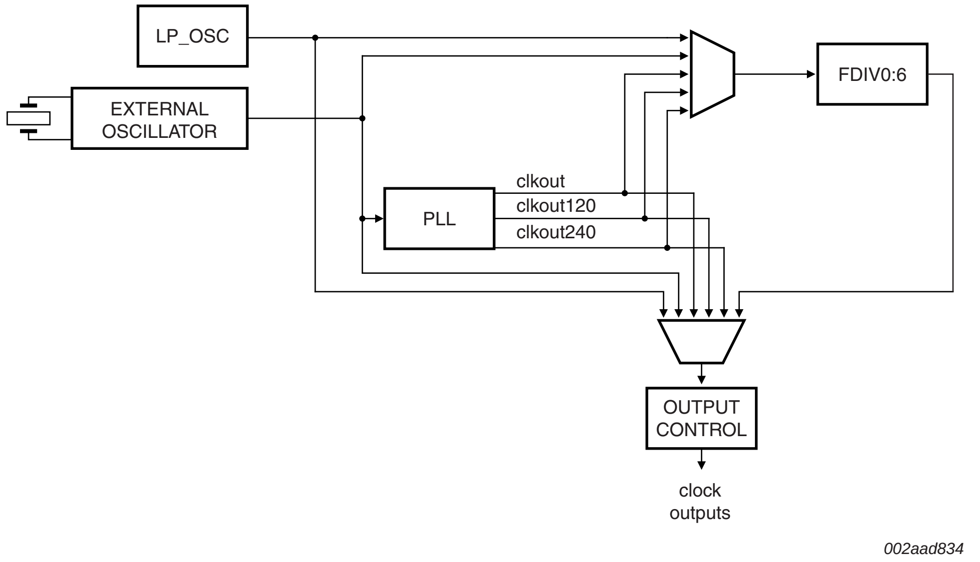


Fig 10. Structure of the clock generation scheme

Any output generator (except for BASE_SAFE_CLK and BASE_PCR_CLK) can be connected to either a fractional divider (FDIV[0:6]) or to one of the outputs of the PLL or to LP_OSC/crystal oscillator directly. BASE_SAFE_CLK and BASE_PCR_CLK can use only LP_OSC as source.

The fractional dividers can be connected to one of the outputs of the PLL or directly to LP_OSC/crystal Oscillator.

The PLL is connected to the crystal oscillator.

In this way every output generating the base clocks can be configured to get the required clock. Multiple output generators can be connected to the same primary or secondary clock source, and multiple secondary clock sources can be connected to the same PLL output or primary clock source.

Invalid selections/programming - connecting the PLL to an FDIV or to one of the PLL outputs itself for example - will be blocked by hardware. The control register will not be written, the previous value will be kept, although all other fields will be written with new data. This prevents clocks being blocked by incorrect programming.

Default Clock Sources: Every secondary clock generator or output generator is connected to LP_OSC at reset. In this way the device runs at a low frequency after reset. It is recommended to switch BASE_SYS_CLK to a high-frequency clock generator as (one of) the first step(s) in the boot code after verifying that the high-frequency clock generator is running.

Clock Activity Detection: Clocks that are inactive are automatically regarded as invalid, and values of ‘CLK_SEL’ that would select those clocks are masked and not written to the control registers. This is accomplished by adding a clock detector to every clock generator. The RDET register keeps track of which clocks are active and inactive, and the appropriate ‘CLK_SEL’ values are masked and unmasked accordingly. Each clock detector can also generate interrupts at clock activation and deactivation so that the system can be notified of a change in internal clock status.

Clock detection is done using a counter running at the *BASE_PCR_CLK* frequency. If no positive clock edge occurs before the counter has 32 cycles of BASE_PCR_CLK the clock is assumed to be inactive. As *BASE_PCR_CLK* is slower than any of the clocks to be detected, normally only one BASE_PCR_CLK cycle is needed to detect activity. After reset all clocks are assumed to be ‘non-present’, so the RDET status register will be correct only after 32 *BASE_PCR_CLK* cycles.

Note that this mechanism cannot protect against a currently-selected clock going from active to inactive state. Therefore an inactive clock may still be sent to the system under special circumstances, although an interrupt can still be generated to notify the system.

Glitch-Free Switching: Provisions are included in the CGU to allow clocks to be switched glitch-free, both at the output generator stage and also at secondary source generators.

In the case of the PLL the clock will be stopped and held low for long enough to allow the PLL to stabilize and lock before being re-enabled. For all non-PLL Generators the switch will occur as quickly as possible, although there will always be a period when the clock is held low due to synchronization requirements.

If the current clock is high and does not go low within 32 *cycles of BASE_PCR_CLK* it is assumed to be inactive and is asynchronously forced low. This prevents deadlocks on the interface.

6.15.2.2 PLL functional description

A block diagram of the PLL is shown in Figure 11. The input clock is fed directly to the analog section. This block compares the phase and frequency of the inputs and generates the main clock[2]. These clocks are either divided by 2 × P by the programmable post divider to create the output clock, or sent directly to the output. The main output clock is then divided by M by the programmable feedback divider to generate the feedback clock. The output signal of the analog section is also monitored by the lock detector to signal when the PLL has locked onto the input clock.

---

[2] Generation of the main clock is restricted by the frequency range of the PLL clock input. See Table 33 “Dynamic characteristics”.

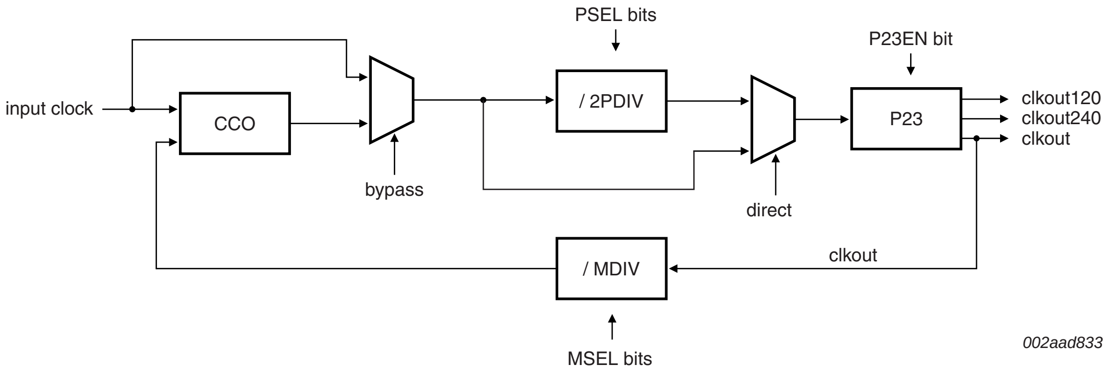


Fig 11. PLL block diagram

Triple output phases: For applications that require multiple clock phases two additional clock outputs can be enabled by setting register P23EN to logic 1, thus giving three clocks with a 120° phase difference. In this mode all three clocks generated by the analog section are sent to the output dividers. When the PLL has not yet achieved lock the second and third phase output dividers run unsynchronized, which means that the phase relation of the output clocks is unknown. When the PLL LOCK register is set the second and third phase of the output dividers are synchronized to the main output clock CLKOUT PLL, thus giving three clocks with a 120° phase difference.

Direct output mode: In normal operating mode (with DIRECT set to logic 0) the CCO clock is divided by 2, 4, 8 or 16 depending on the value on the PSEL[1:0] input, giving an output clock with a 50 % duty cycle. If a higher output frequency is needed the CCO clock can be sent directly to the output by setting DIRECT to logic 1. Since the CCO does not directly generate a 50 % duty cycle clock, the output clock duty cycle in this mode can deviate from 50 %.

Power-down control: A Power-down mode has been incorporated to reduce power consumption when the PLL clock is not needed. This is enabled by setting the PD control register bit. In this mode the analog section of the PLL is turned off, the oscillator and the phase-frequency detector are stopped and the dividers enter a reset state. While in Power-down mode the LOCK output is low, indicating that the PLL is not in lock. When Power-down mode is terminated by clearing the PD control-register bit the PLL resumes normal operation, and makes the LOCK signal high once it has regained lock on the input clock.

6.15.2.3 Pin description

The CGU0 module in the LPC2921/2923/2925 has the pins listed in Table 25 below.

Table 25. CGU0 pins

Symbol	Direction	Description
XOUT_OSC	OUT	Oscillator crystal output
XIN_OSC	IN	Oscillator crystal input or external clock input

6.15.3 Clock generation for USB (CGU1)

The CGU1 block is functionally identical to the CGU0 block and generates the clock for the USB interface and a dedicated output clock. The CGU1 block uses its own PLL and fractional divider. The PLLs used in CGU0 and CGU1 are identical (see Section 6.15.2.2).

The clock input to the CGU1 PLL is provided by one of two base clocks generated in the CGU0: BASE_ICLK0_CLK or BASE_ICLK1_CLK. The base clock not used for the PLL can be configured to drive the output clock directly.

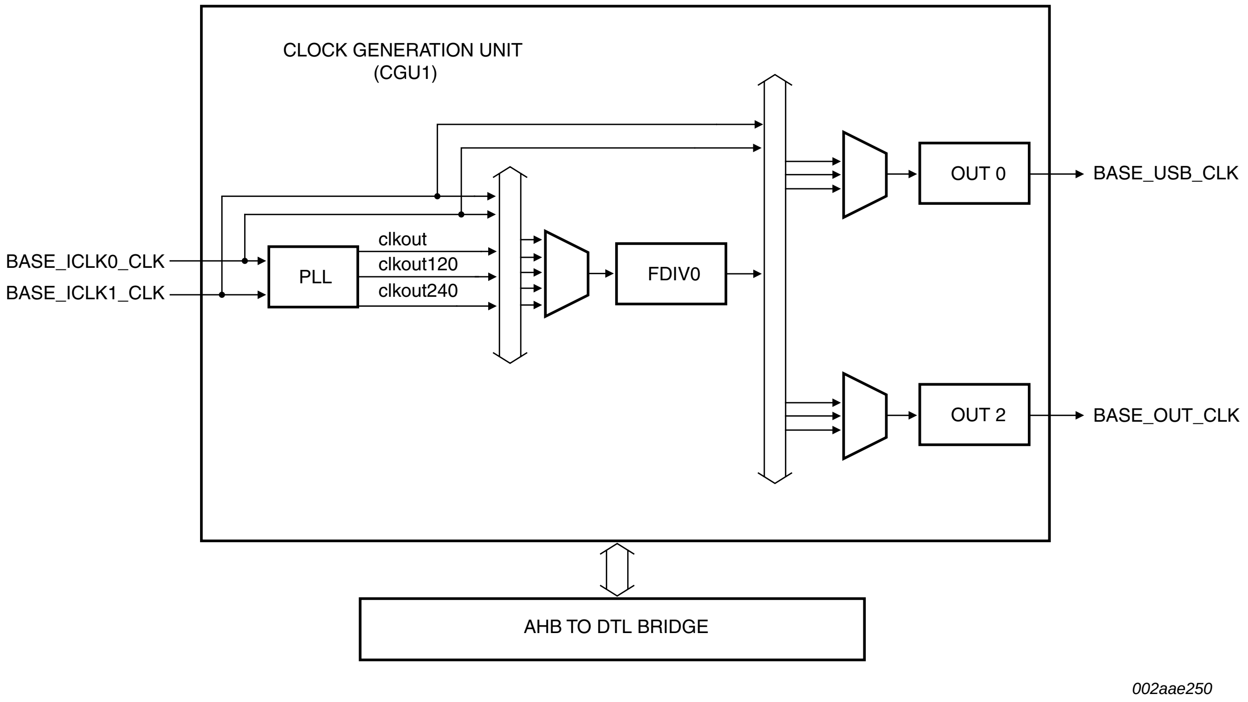


Fig 12. Block diagram of the CGU1

6.15.3.1 Pin description

The CGU1 module in the LPC2921/2923/2925 has the pins listed in Table 25 below.

Table 26. CGU1 pins

Symbol	Direction	Description
CLK_OUT	OUT	clock output

6.15.4 Reset Generation Unit (RGU)

The RGU controls all internal resets.

The key features of the Reset Generation Unit (RGU) are:

- Reset controlled individually per subsystem
- Automatic reset stretching and release
- Monitor function to trace resets back to source

- Register write-protection mechanism to prevent unintentional resets

6.15.4.1 Functional description

Each reset output is defined as a combination of reset input sources including the external reset input pins and internal power-on reset, see Table 27. The first five resets listed in this table form a sort of cascade to provide the multiple levels of impact that a reset may have. The combined input sources are logically OR-ed together so that activating any of the listed reset sources causes the output to go active.

Table 27. Reset output configuration

Reset output	Reset source	Parts of the device reset when activated
POR_RST	power-on reset module	LP_OSC; source for RGU_RST
RGU_RST	POR_RST, RST_N pin	RGU internal; source for PCR_RST
PCR_RST	RGU_RST, WATCHDOG	PCR internal; source for COLD_RST
COLD_RST	PCR_RST	parts with COLD_RST as reset source below
WARM_RST	COLD_RST	parts with WARM_RST as reset source below
SCU_RST	COLD_RST	SCU
CFID_RST	COLD_RST	CFID
FMC_RST	COLD_RST	embedded Flash Memory Controller (FMC)
EMC_RST	COLD_RST	embedded SRAM-Memory Controller
GESS_A2V_RST	WARM_RST	GeSS AHB-to-APB bridge
PESS_A2V_RST	WARM_RST	PeSS AHB-to-APB bridge
GPIO_RST	WARM_RST	all GPIO modules
UART_RST	WARM_RST	all UART modules
TMR_RST	WARM_RST	all timer modules in PeSS
SPI_RST	WARM_RST	all SPI modules
IVNSS_A2V_RST	WARM_RST	IVNSS AHB-to-APB bridge
IVNSS_CAN_RST	WARM_RST	all CAN modules including Acceptance filter
IVNSS_LIN_RST	WARM_RST	all LIN modules
MSCSS_A2V_RST	WARM_RST	MSCSS AHB to APB bridge
MSCSS_PWM_RST	WARM_RST	all PWM modules
MSCSS_ADC_RST	WARM_RST	all ADC modules
MSCSS_TMR_RST	WARM_RST	all timer modules in MSCSS
I2C_RST	WARM_RST	all I^{2}C modules
QEI_RST	WARM_RST	Quadrature encoder
DMA_RST	WARM_RST	DMA controller
USB_RST	WARM_RST	USB controller
VIC_RST	WARM_RST	Vectored Interrupt Controller (VIC)
AHB_RST	WARM_RST	CPU and AHB Bus infrastructure

6.15.4.2 Pin description

The RGU module in the LPC2921/2923/2925 has the following pins. Table 28 shows the RGU pins.

Table 28. RGU pins

Symbol	Direction	Description
$\overline{\text{RST}}$	IN	external reset input, active LOW; pulled up internally

6.15.5 Power Management Unit (PMU)

This module enables software to actively control the system’s power consumption by disabling clocks not required in a particular operating mode.

Using the base clocks from the CGU as input, the PMU generates branch clocks to the rest of the LPC2921/2923/2925. Output clocks branched from the same base clock are phase- and frequency-related. These branch clocks can be individually controlled by software programming.

The key features are:

- Individual clock control for all LPC2921/2923/2925 sub-modules.
- Activates sleeping clocks when a wake-up event is detected.
- Clocks can be individually disabled by software.
- Supports AHB master-disable protocol when AUTO mode is set.
- Disables wake-up of enabled clocks when Power-down mode is set.
- Activates wake-up of enabled clocks when a wake-up event is received.
- Status register is available to indicate if an input base clock can be safely switched off (i.e. all branch clocks are disabled).

6.15.5.1 Functional description

The PMU controls all internal clocks coming out of the CGU0 for power-mode management. With some exceptions, each branch clock can be switched on or off individually under control of software register bits located in its individual configuration register. Some branch clocks controlling vital parts of the device operate in a fixed mode. Table 29 shows which mode-control bits are supported by each branch clock.

By programming the configuration register the user can control which clocks are switched on or off, and which clocks are switched off when entering Power-down mode.

Note that the standby-wait-for-interrupt instructions of the ARM968E-S processor (putting the ARM CPU into a low-power state) are not supported. Instead putting the ARM CPU into power-down should be controlled by disabling the branch clock for the CPU.

Remark: For any disabled branch clocks to be re-activated their corresponding base clocks must be running (controlled by the CGU0).

Table 29 shows the relation between branch and base clocks, see also Section 6.7.1. Every branch clock is related to one particular base clock: it is not possible to switch the source of a branch clock in the PMU.

Table 29. Branch clock overview

Legend:

‘1’ Indicates that the related register bit is tied off to logic HIGH, all writes are ignored

‘0’ Indicates that the related register bit is tied off to logic LOW, all writes are ignored

‘+’ Indicates that the related register bit is readable and writable

Branch clock name	Base clock	Implemented switch on/off mechanism		
		WAKE-UP	AUTO	RUN
CLK_SAFE	BASE_SAFE_CLK	0	0	1
CLK_SYS_CPU	BASE_SYS_CLK	+	+	1
CLK_SYS	BASE_SYS_CLK	+	+	1
CLK_SYS_PCR	BASE_SYS_CLK	+	+	1
CLK_SYS_FMC	BASE_SYS_CLK	+	+	+
CLK_SYS_RAM0	BASE_SYS_CLK	+	+	+
CLK_SYS_RAM1	BASE_SYS_CLK	+	+	+
CLK_SYS_GESS	BASE_SYS_CLK	+	+	+
CLK_SYS_VIC	BASE_SYS_CLK	+	+	+
CLK_SYS_PESS	BASE_SYS_CLK	+	+	+
CLK_SYS_GPIO0	BASE_SYS_CLK	+	+	+
CLK_SYS_GPIO1	BASE_SYS_CLK	+	+	+
CLK_SYS_GPIO5	BASE_SYS_CLK	+	+	+
CLK_SYS_IVNSS_A	BASE_SYS_CLK	+	+	+
CLK_SYS_MSCSS_A	BASE_SYS_CLK	+	+	+
CLK_SYS_DMA	BASE_SYS_CLK	+	+	+
CLK_SYS_USB	BASE_SYS_CLK	+	+	+
CLK_PCR_SLOW	BASE_PCR_CLK	+	+	1
CLK_IVNSS_APB	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_CANC0	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_CANC1	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_I2C0	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_I2C1	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_LIN0	BASE_IVNSS_CLK	+	+	+
CLK_IVNSS_LIN1	BASE_IVNSS_CLK	+	+	+
CLK_MSCSS_APB	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_MTMR0	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_MTMR1	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_PWM0	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_PWM1	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_PWM2	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_PWM3	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_ADC1_APB	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_ADC2_APB	BASE_MSCSS_CLK	+	+	+
CLK_MSCSS_QEI	BASE_MSCSS_CLK	+	+	+
CLK_OUT_CLK	BASE_OUT_CLK	+	+	+

Table 29. Branch clock overview *…continued*

Legend:
‘1’ Indicates that the related register bit is tied off to logic HIGH, all writes are ignored
‘0’ Indicates that the related register bit is tied off to logic LOW, all writes are ignored
‘+’ Indicates that the related register bit is readable and writable

Branch clock name	Base clock	Implemented switch on/off mechanism		
		WAKE-UP	AUTO	RUN
CLK_UART0	BASE_UART_CLK	+	+	+
CLK_UART1	BASE_UART_CLK	+	+	+
CLK_SPI0	BASE_SPI_CLK	+	+	+
CLK_SPI1	BASE_SPI_CLK	+	+	+
CLK_SPI2	BASE_SPI_CLK	+	+	+
CLK_TMR0	BASE_TMR_CLK	+	+	+
CLK_TMR1	BASE_TMR_CLK	+	+	+
CLK_TMR2	BASE_TMR_CLK	+	+	+
CLK_TMR3	BASE_TMR_CLK	+	+	+
CLK_ADC1	BASE_ADC_CLK	+	+	+
CLK_ADC2	BASE_ADC_CLK	+	+	+
CLK_USB	BASE_USB_CLK	+	+	+

6.16 Vectored interrupt controller

The LPC2921/2923/2925 contains a very flexible and powerful Vectored Interrupt Controller (VIC) to interrupt the ARM processor on request.

The key features are:

- Level-active interrupt request with programmable polarity.
- 56 interrupt-request inputs.
- Software-interrupt request capability associated with each request input.
- Interrupt request state can be observed before masking.
- Software-programmable priority assignments to interrupt requests up to 15 levels.
- Software-programmable routing of interrupt requests towards the ARM-processor inputs IRQ and FIQ.
- Fast identification of interrupt requests through vector.
- Support for nesting of interrupt service routines.

6.16.1 Functional description

The Vectored Interrupt Controller routes incoming interrupt requests to the ARM processor. The interrupt target is configured for each interrupt request input of the VIC. The targets are defined as follows:

- Target 0 is ARM processor FIQ (fast interrupt service).
- Target 1 is ARM processor IRQ (standard interrupt service).

Interrupt-request masking is performed individually per interrupt target by comparing the priority level assigned to a specific interrupt request with a target-specific priority threshold. The priority levels are defined as follows:

- Priority level 0 corresponds to ‘masked’ (i.e. interrupt requests with priority 0 never lead to an interrupt).
- Priority 1 corresponds to the lowest priority.
- Priority 15 corresponds to the highest priority.

Software interrupt support is provided and can be supplied for:

- Testing RTOS (Real-Time Operating System) interrupt handling without using device-specific interrupt service routines.
- Software emulation of an interrupt-requesting device, including interrupts.

6.16.2 Clock description

The VIC is clocked by CLK_SYS_VIC, see Section 6.7.2.

7. Limiting values

Table 30. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions	Min	Max	Unit
Supply pins					
P_{tot}	total power dissipation	[1]	-	1.5	W
$V_{DD(CORE)}$	core supply voltage		–0.5	+2.0	V
$V_{DD(OSC\_PLL)}$	oscillator and PLL supply voltage		–0.5	+2.0	V
$V_{DDA(ADC3V3)}$	3.3 V ADC analog supply voltage		–0.5	+4.6	V
$V_{DD(IO)}$	input/output supply voltage		–0.5	+4.6	V
I_{DD}	supply current	average value per supply pin [2]	-	98	mA
I_{SS}	ground current	average value per ground pin [2]	-	98	mA
Input pins and I/O pins					
V_{XIN_OSC}	voltage on pin XIN_OSC		–0.5	+2.0	V
$V_{I(IO)}$	I/O input voltage	[3][4][5]	–0.5	$V_{DD(IO)}$ + 3.0	V
$V_{I(ADC)}$	ADC input voltage	for ADC1/2: I/O port 0 pin 8 to pin 23. [4][5]	–0.5	$V_{DDA(ADC3V3)}$ + 0.5	V
V_{VREFP}	voltage on pin VREFP		–0.5	+3.6	V
V_{VREFN}	voltage on pin VREFN		–0.5	+3.6	V
$I_{I(ADC)}$	ADC input current	average value per input pin [2]	-	35	mA
Output pins and I/O pins configured as output					
I_{OHS}	HIGH-level short-circuit output current	drive HIGH, output shorted to $V_{SS(IO)}$ [6]	-	–33	mA
I_{OLS}	LOW-level short-circuit output current	drive LOW, output shorted to ${}_{VDD(IO)}$ [6]	-	+38	mA
General					
T_{stg}	storage temperature		–65	+150	°C
T_{amb}	ambient temperature		–40	+85	°C

Table 30. Limiting values *…continued*

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions		Min	Max	Unit
ESD						
V_{ESD}	electrostatic discharge voltage	on all pins				
		human body model	[7]	–2000	+2000	V
		charged device model		–500	+500	V
		on corner pins				
		charged device model		–750	+750	V

[1] Based on package heat transfer, not device power consumption.

[2] Peak current must be limited at 25 times average current.

[3] For I/O Port 0, the maximum input voltage is defined by $V_{I(ADC)}$.

[4] Only when $V_{DD(IO)}$ is present.

[5] Note that pull-up should be off. With pull-up do not exceed 3.6 V.

[6] 112 mA per $V_{DD(IO)}$ or $V_{SS(IO)}$ should not be exceeded.

[7] Human-body model: discharging a 100 pF capacitor via a 10 kΩ series resistor.

8. Static characteristics

Table 31. Static characteristics

$V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{vj} = –40 °C to +85 °C; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Supplies						
Core supply						
$V_{DD(CORE)}$	core supply voltage		1.71	1.80	1.89	V
$I_{DD(CORE)}$	core supply current	Device state after reset; system clock at 125 MHz; T_{amb} = 85 °C; executing code `while(1){}` from flash.	-	75	-	mA
		all clocks off [2]	-	30	475	µA
I/O supply						
$V_{DD(IO)}$	input/output supply voltage		2.7	-	3.6	V
$I_{DD(IO)}$	I/O supply current	Power-down mode	-	0.5	3.25	µA
Oscillator/PLL supply						
$V_{DD(OSC\_PLL)}$	oscillator and PLL supply voltage		1.71	1.80	1.89	V
$I_{DD(OSC\_PLL)}$	oscillator and PLL supply current	Normal mode	-	-	1	mA
		Power-down mode	-	-	2	µA
Analog-to-digital converter supply						
$V_{DDA(ADC3V3)}$	3.3 V ADC analog supply voltage		3.0	3.3	3.6	V
$I_{DDA(ADC3V3)}$	3.3 V ADC analog supply current	Normal mode	-	-	1.9	mA
		Power-down mode	-	-	4	µA
Input pins and I/O pins configured as input						
V_I	input voltage	all port pins and $V_{DD(IO)}$ applied; [3][4] see Section 7	–0.5	-	+ 5.5	V
		port 0 pin 8 to pin 23 when ADC1/2 is used [4]	-	-	V_{VREFP}	
		all port pins and $V_{DD(IO)}$ not applied	–0.5	-	+3.6	V
		all other I/O pins, $\overline{\text{RST}}$, $\overline{\text{TRST}}$, TDI, JTAGSEL, TMS, TCK	–0.5	-	$V_{DD(IO)}$	V
V_{IH}	HIGH-level input voltage	all port pins, $\overline{\text{RST}}$, $\overline{\text{TRST}}$, TDI, JTAGSEL, TMS, TCK	2.0	-	-	V
V_{IL}	LOW-level input voltage	all port pins, $\overline{\text{RST}}$, $\overline{\text{TRST}}$, TDI, JTAGSEL, TMS, TCK	-	-	0.8	V
V_{hys}	hysteresis voltage		0.4	-	-	V

Table 31. Static characteristics *…continued*

$V_{DD(CORE)} = V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{vj} = −40 °C to +85 °C; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{LIH}	HIGH-level input leakage current		-	-	1	µA
I_{LIL}	LOW-level input leakage current		-	-	1	µA
$I_{I(pd)}$	pull-down input current	all port pins, V_I = 3.3 V; V_I = 5.5 V; see Figure 20	25	50	100	µA
$I_{I(pu)}$	pull-up input current	all port pins, $\overline{RST}$, $\overline{TRST}$, TDI, JTAGSEL, TMS: V_I = 0 V; V_I > 3.6 V is not allowed; Figure 21	−25	−50	−115	µA
C_i	input capacitance	[5]	-	3	8	pF
Output pins and I/O pins configured as output						
V_O	output voltage		0	-	$V_{DD(IO)}$	V
V_{OH}	HIGH-level output voltage	I_{OH} = −4 mA; see Figure 19	$V_{DD(IO)} - 0.4$	-	-	V
V_{OL}	LOW-level output voltage	I_{OL} = 4 mA; Figure 18	-	-	0.4	V
C_L	load capacitance		-	-	25	pF
USB pins USB_D+ and USB_D−						
Input characteristics						
V_{IH}	HIGH-level input voltage		1.5	-	-	V
V_{IL}	LOW-level input voltage		-	-	1.3	V
V_{hys}	hysteresis voltage		0.4	-	-	V
Output characteristics						
Z_o	output impedance	with 33 Ω series resistor	36.0	-	44.1	Ω
V_{OH}	HIGH-level output voltage	(driven) for low-/full-speed; R_L of 15 kΩ to GND	2.9	-	3.5	V
V_{OL}	LOW-level output voltage	(driven) for low-/full-speed; with 1.5 kΩ resistor to 3.6 V external pull-up	-	-	0.18	V
I_{OH}	HIGH-level output current	at $V_{OH} = V_{DD(IO)} - 0.3$ V; without 33 Ω external series resistor	20.8	-	41.7	mA
		at $V_{OH} = V_{DD(IO)} - 0.3$ V; with 33 Ω external series resistor	4.8	-	5.3	mA
I_{OL}	LOW-level output current	at V_{OL} = 0.3 V; without 33 Ω external series resistor	26.7	-	57.2	mA
		at V_{OL} = 0.3 V; with 33 Ω external series resistor	5.0	-	5.5	mA
I_{OHS}	HIGH-level short-circuit output current	drive high; pad connected to ground	-	-	90.0	mA

Table 31. Static characteristics *…continued*

$V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{vj} = −40 °C to +85 °C; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
I_{OLS}	LOW-level short-circuit output current	drive high; pad connected to $V_{DD(IO)}$		-	-	95.1	mA
Oscillator							
V_{XIN_OSC}	voltage on pin XIN_OSC			0	-	1.8	V
$R_{s(xtal)}$	crystal series resistance	f_{osc} = 10 MHz to 15 MHz	[6]				
		C_{xtal} = 10 pF; C_{ext} = 18 pF		-	-	160	Ω
		C_{xtal} = 20 pF; C_{ext} = 39 pF		-	-	60	Ω
		f_{osc} = 15 MHz to 20 MHz	[6]				
		C_{xtal} = 10 pF; C_{ext} = 18 pF		-	-	80	Ω
C_i	input capacitance	of XIN_OSC	[7]	-	-	2	pF
Power-up reset							
$V_{trip(high)}$	high trip level voltage		[8]	1.1	1.4	1.6	V
$V_{trip(low)}$	low trip level voltage		[8]	1.0	1.3	1.5	V
$V_{trip(dif)}$	difference between high and low trip level voltage		[8]	50	120	180	mV

[1] All parameters are guaranteed over the virtual junction temperature range by design. Pre-testing is performed at T_{amb} = 85 °C on wafer level. Cased products are tested at T_{amb} = 25 °C (final testing). Both pre-testing and final testing use correlated test conditions to cover the specified temperature and power-supply voltage range.

[2] Leakage current is exponential to temperature; worst-case value is at 85 °C T_{vj}. All clocks off. Analog modules and flash powered down.

[3] Not 5 V-tolerant when pull-up is on.

[4] For I/O Port 0, the maximum input voltage is defined by $V_{I(ADC)}$.

[5] For Port 0, pin 0 to pin 15 add maximum 1.5 pF for input capacitance to ADC. For Port 0, pin 16 to pin 31 add maximum 1.0 pF for input capacitance to ADC.

[6] C_{xtal} is crystal load capacitance and C_{ext} are the two external load capacitors.

[7] This parameter is not part of production testing or final testing, hence only a typical value is stated. Maximum and minimum values are based on simulation results.

[8] The power-up reset has a time filter: $V_{DD(CORE)}$ must be above $V_{trip(high)}$ for 2 μs before reset is de-asserted; $V_{DD(CORE)}$ must be below $V_{trip(low)}$ for 11 μs before internal reset is asserted.

Table 32. ADC static characteristics

$V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{amb} = −40 °C to +85 °C unless otherwise specified; ADC frequency 4.5 MHz.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{VREFN}	voltage on pin VREFN		0	-	V_{VREFP} − 2	V
V_{VREFP}	voltage on pin VREFP		V_{VREFN} + 2	-	$V_{DDA(ADC3V3)}$	V
V_{IA}	analog input voltage		V_{VREFN}	-	V_{VREFP}	V
Z_i	input impedance	between V_{VREFN} and V_{VREFP}	4.4	-	-	kΩ
C_{ia}	analog input capacitance		-	-	1	pF
E_D	differential linearity error	[1][2][3]	-	-	±1	LSB
$E_{L(adj)}$	integral non-linearity	[1][4]	-	-	±2	LSB
E_O	offset error	[1][5]	-	-	±3	LSB
E_G	gain error	[1][6]	-	-	±0.5	%
E_T	absolute error	[1][7]	-	-	±4	LSB
R_{vsi}	voltage source interface resistance	[8]	-	-	40	kΩ
FSR	full scale range		2	-	10	bit

[1] Conditions: $V_{SS(IO)}$ = 0 V, $V_{DDA(ADC3V3)}$ = 3.3 V.

[2] The ADC is monotonic, there are no missing codes.

[3] The differential linearity error (E_D) is the difference between the actual step width and the ideal step width. See Figure 14.

[4] The integral non-linearity ($E_{L(adj)}$) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset errors. See Figure 14.

[5] The offset error (E_O) is the absolute difference between the straight line which fits the actual curve and the straight line which fits the ideal curve. See Figure 14.

[6] The gain error (E_G) is the relative difference in percent between the straight line fitting the actual transfer curve after removing offset error, and the straight line which fits the ideal transfer curve. See Figure 14.

[7] The absolute error (E_T) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve. See Figure 14.

[8] See Figure 13.

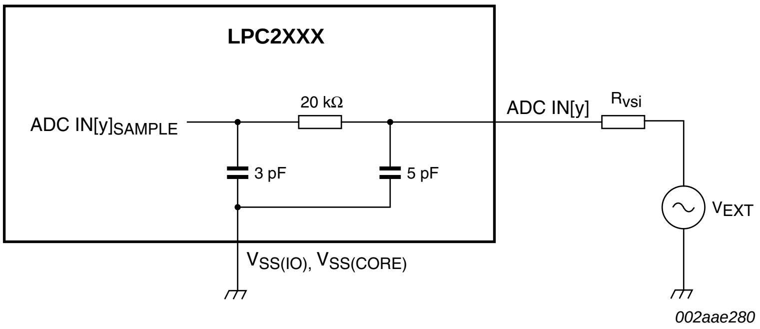
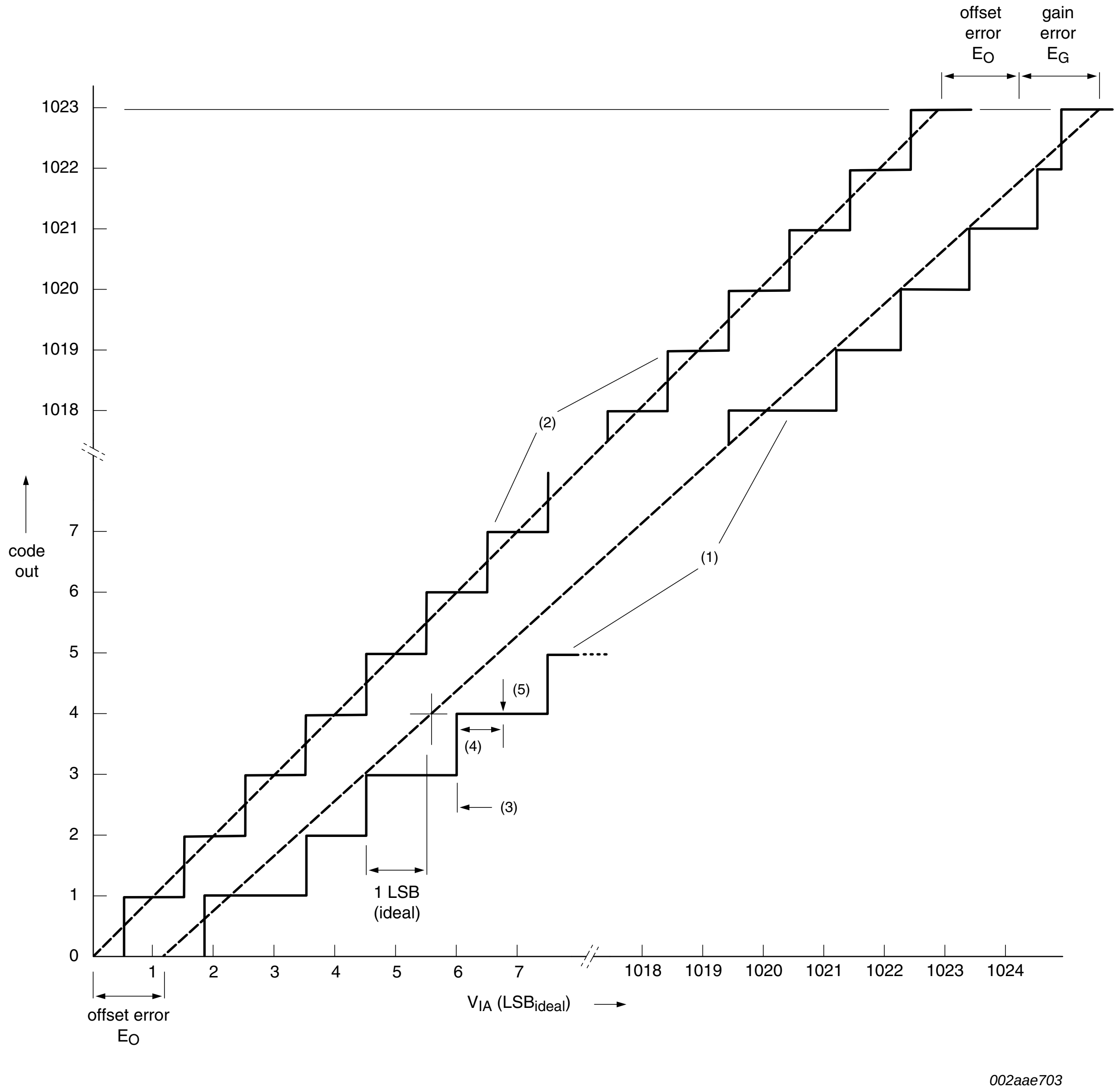


Fig 13. Suggested ADC interface - LPC2921/2923/2925 ADC1/2 IN[y] pin



(1) Example of an actual transfer curve.

(2) The ideal transfer curve.

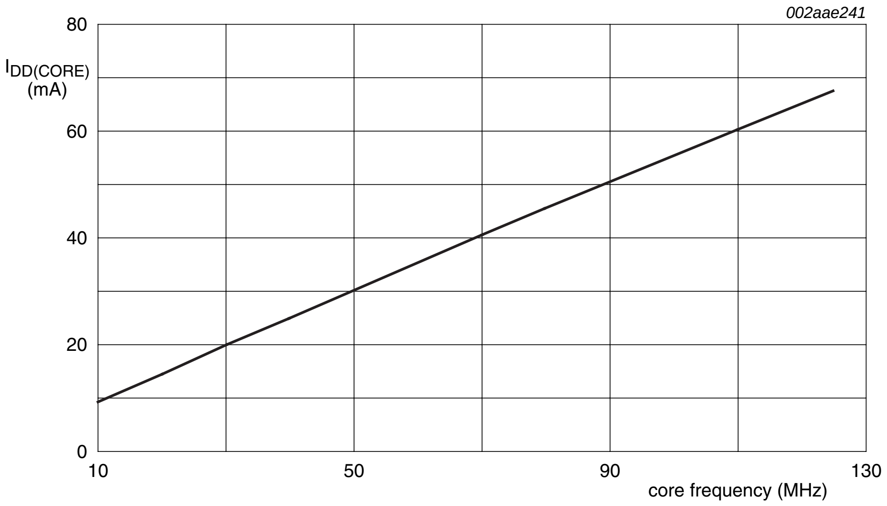
(3) Differential linearity error (E_D).

(4) Integral non-linearity ($E_{L(adj)}$).

(5) Center of a step of the actual transfer curve.

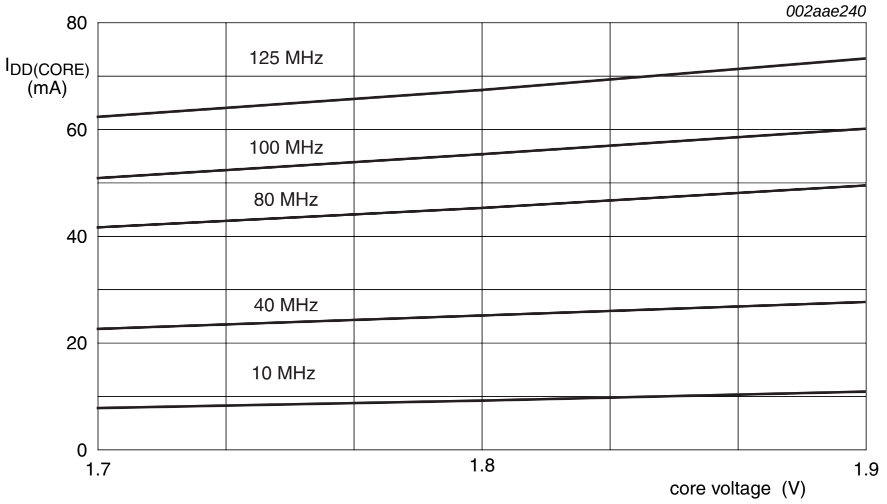
Fig 14. ADC characteristics

8.1 Power consumption



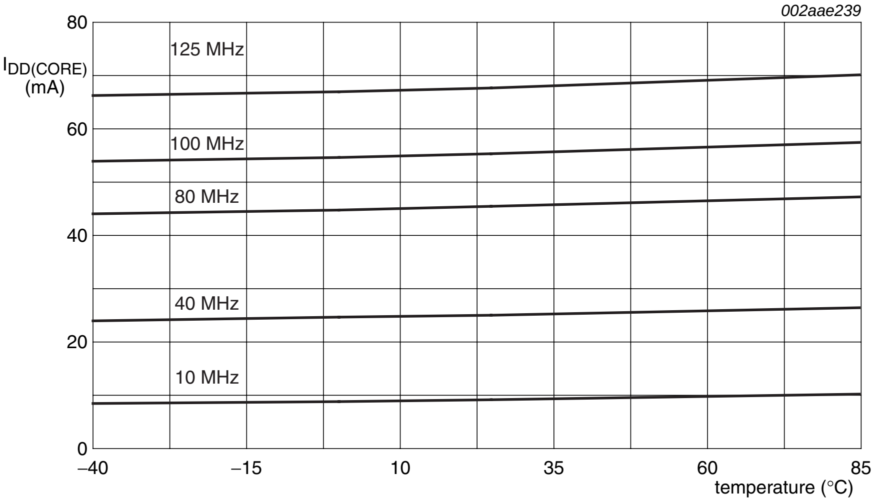
Conditions: T_{amb} = 25 °C; active mode entered executing code from flash; core voltage 1.8 V; all peripherals enabled but not configured to run.

Fig 15. $I_{DD(CORE)}$ at different core frequencies (active mode)



Conditions: T_{amb} = 25 °C; active mode entered executing code from flash; all peripherals enabled but not configured to run.

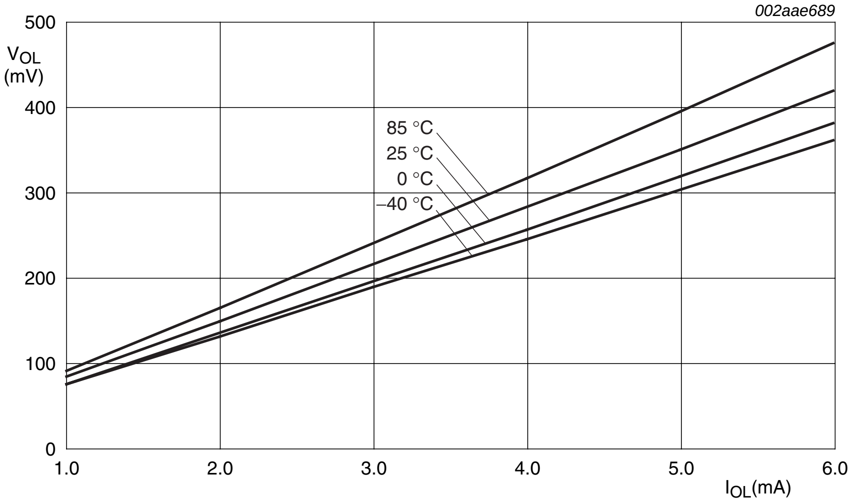
Fig 16. $I_{DD(CORE)}$ at different core voltages $V_{DD(CORE)}$ (active mode)



Conditions: active mode entered executing code from flash; core voltage 1.8 V; all peripherals enabled but not configured to run.

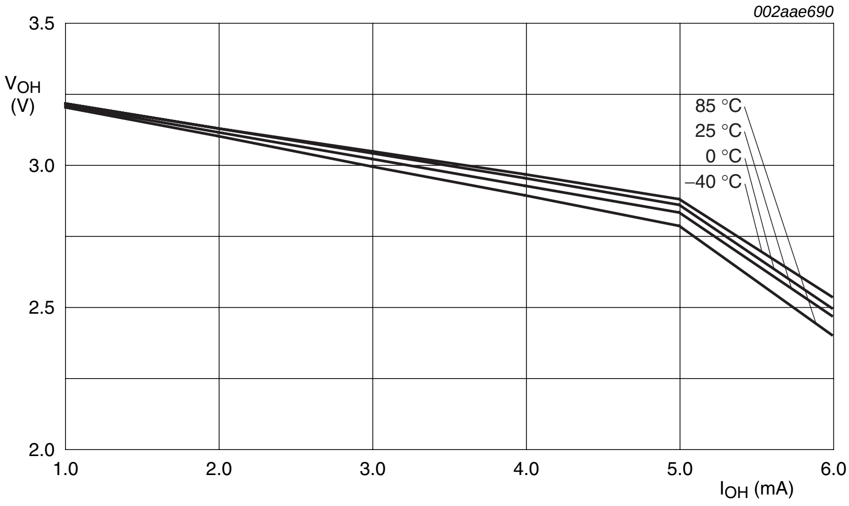
Fig 17. $I_{DD(CORE)}$ **at different temperatures (active mode)**

8.2 Electrical pin characteristics



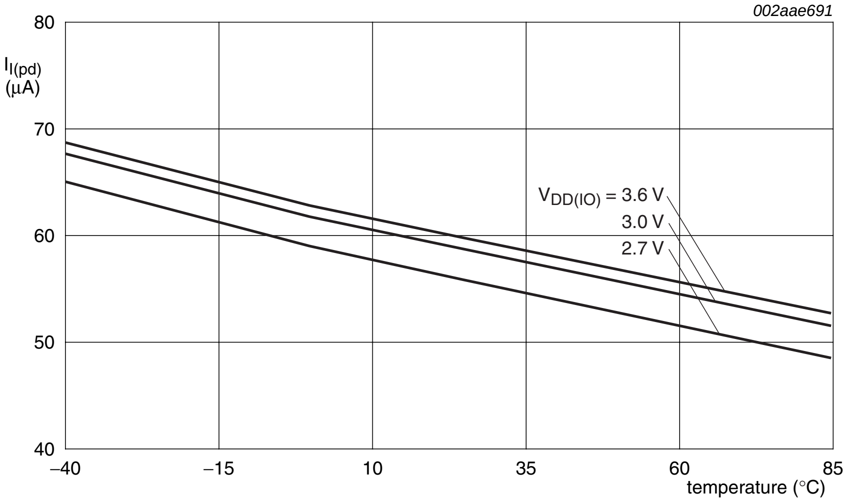
$V_{DD(IO)} = 3.3$ V.

Fig 18. Typical LOW-level output voltage versus LOW-level output current



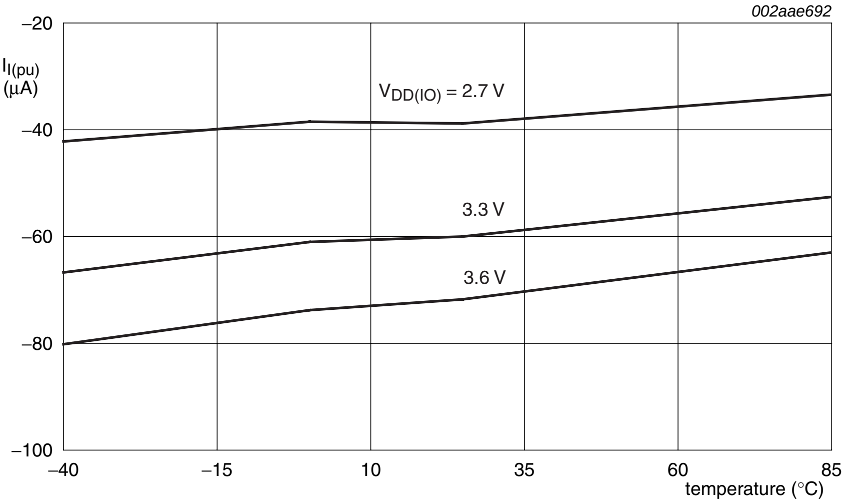
$V_{DD(IO)}$ = 3.3 V.

Fig 19. Typical HIGH-level output voltage versus HIGH-level output current



V_I = 3.3 V.

Fig 20. Typical pull-down current versus temperature



V_I = 0 V.

Fig 21. Typical pull-up current versus temperature

9. Dynamic characteristics

9.1 Dynamic characteristics: I/O and CLK_OUT pins, internal clock, oscillators, PLL, and CAN

Table 33. Dynamic characteristics

$V_{DD(CORE)} = V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
I/O pins							
t_{THL}	HIGH to LOW transition time	C_L = 30 pF		4	-	13.8	ns
t_{TLH}	LOW to HIGH transition time	C_L = 30 pF		4	-	13.8	ns
CLK_OUT pin							
f_{clk}	clock frequency	on pin CLK_OUT		-	-	40	MHz
Internal clock							
$f_{clk(sys)}$	system clock frequency		[2]	10	-	125	MHz
$T_{clk(sys)}$	system clock period		[2]	8	-	100	ns
Low-power ring oscillator							
$f_{ref(RO)}$	RO reference frequency			0.4	0.5	0.6	MHz
$t_{startup}$	start-up time	at maximum frequency	[3]	-	6	-	µs
Oscillator							
$f_{i(osc)}$	oscillator input frequency	maximum frequency is the clock input of an external clock source applied to the XIN_OSC pin		10	-	100	MHz
$t_{startup}$	start-up time	at maximum frequency	[3] [4]	-	500	-	µs
PLL							
$f_{i(PLL)}$	PLL input frequency			10	-	25	MHz
$f_{o(PLL)}$	PLL output frequency			10	-	160	MHz
		CCO; direct mode		156	-	320	MHz
$t_{a(clk)}$	clock access time			-	-	63.4	ns
$t_{a(A)}$	address access time			-	-	60.3	ns
Jitter specification for CAN							
$t_{jit(cc)(p-p)}$	cycle to cycle jitter (peak-to-peak value)	on CAN TXDC pin	[3]	-	0.4	1	ns

[1] All parameters are guaranteed over the virtual junction temperature range by design. Pre-testing is performed at T_{amb} = 85 °C ambient temperature on wafer level. Cased products are tested at T_{amb} = 25 °C (final testing). Both pre-testing and final testing use correlated test conditions to cover the specified temperature and power supply voltage range.

[2] See Table 24.

[3] This parameter is not part of production testing or final testing, hence only a typical value is stated.

[4] Oscillator start-up time depends on the quality of the crystal. For most crystals it takes about 1000 clock pulses until the clock is fully stable.

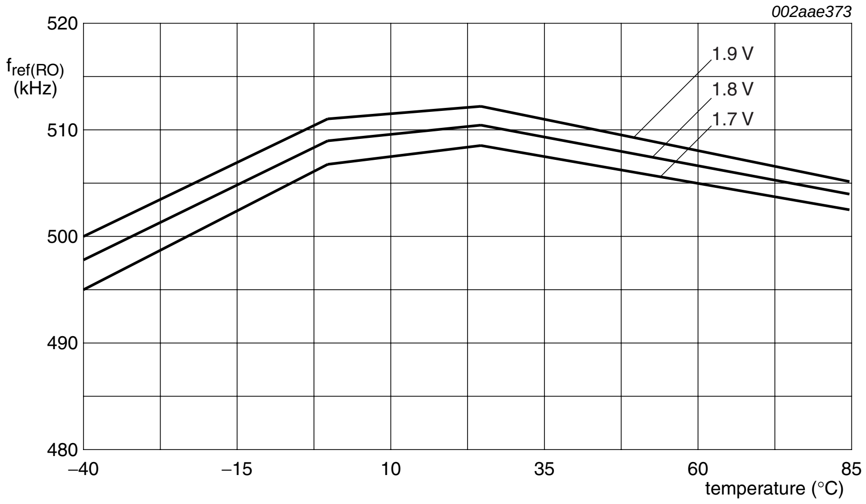


Fig 22. Low-power ring oscillator thermal characteristics

9.2 USB interface

Table 34. Dynamic characteristics: USB pins (full-speed)
C_L = 50 pF; R_{pu} = 1.5 kΩ on D+ to $V_{DD(3V3)}$, unless otherwise specified.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
t_r	rise time	10 % to 90 %		8.5	-	13.8	ns
t_f	fall time	10 % to 90 %		7.7	-	13.7	ns
t_{FRFM}	differential rise and fall time matching	t_r/t_f		-	-	109	%
V_{CRS}	output signal crossover voltage			1.3	-	2.0	V
t_{FEOPT}	source SE0 interval of EOP	see Figure 23		160	-	175	ns
t_{FDEOP}	source jitter for differential transition to SE0 transition	see Figure 23		−2	-	+5	ns
t_{JR1}	receiver jitter to next transition			−18.5	-	+18.5	ns
t_{JR2}	receiver jitter for paired transitions	10 % to 90 %		−9	-	+9	ns
t_{EOPR1}	EOP width at receiver	must reject as EOP; see Figure 23	[1]	40	-	-	ns
t_{EOPR2}	EOP width at receiver	must accept as EOP; see Figure 23	[1]	82	-	-	ns

[1] Characterized but not implemented as production test. Guaranteed by design.

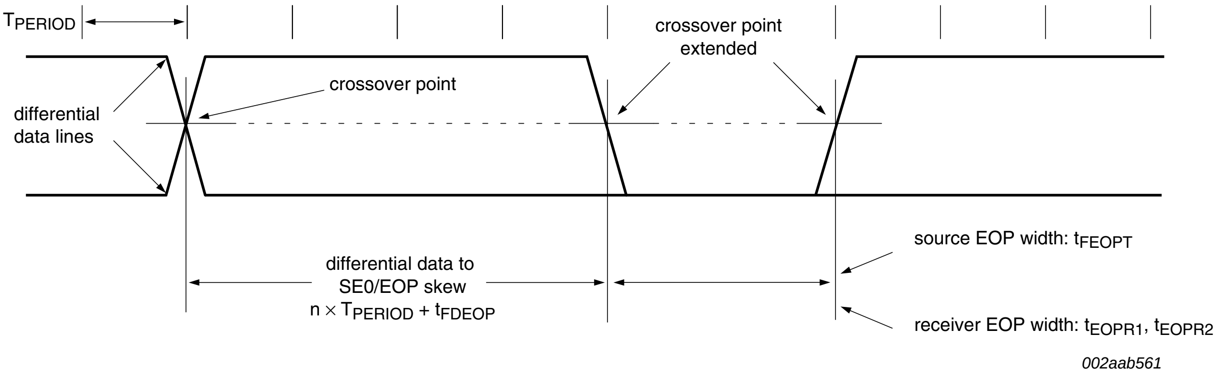


Fig 23. Differential data-to-EOP transition skew and EOP width

9.3 Dynamic characteristics: I^{2}C-bus interface

Table 35. Dynamic characteristic: I^{2}C-bus pins

$V_{DD(CORE)} = V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified[1]

Symbol	Parameter	Conditions	Min	Typ[2]	Max	Unit
$t_{f(o)}$	output fall time	V_{IH} to V_{IL}	$20 + 0.1 \times C_b$[3]	-	-	ns

[1] All parameters are guaranteed over the virtual junction temperature range by design. Pre-testing is performed at T_{amb} = 85 °C ambient temperature on wafer level. Cased products are tested at T_{amb} = 25 °C (final testing). Both pre-testing and final testing use correlated test conditions to cover the specified temperature and power supply voltage range.

[2] Typical ratings are not guaranteed. The values listed are at room temperature (25 °C), nominal supply voltages.

[3] Bus capacitance C_b in pF, from 10 pF to 400 pF.

9.4 Dynamic characteristics: SPI

Table 36. Dynamic characteristics of SPI pins

$V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{vj} = −40 °C to +85 °C; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{SPI}	SPI operating frequency	master operation	$\frac{1}{65024}f_{clk(SPI)}$	-	$\frac{1}{2}f_{clk(SPI)}$	MHz
		slave operation	$\frac{1}{65024}f_{clk(SPI)}$	-	$\frac{1}{4}f_{clk(SPI)}$	MHz
$t_{su(SPI\_MISO)}$	SPI_MISO set-up time	T_{amb} = 25 °C; measured in SPI Master mode; see Figure 24	-	11	-	ns

[1] All parameters are guaranteed over the virtual junction temperature range by design. Pre-testing is performed at T_{amb} = 85 °C ambient temperature on wafer level. Cased products are tested at T_{amb} = 25 °C (final testing). Both pre-testing and final testing use correlated test conditions to cover the specified temperature and power supply voltage range.

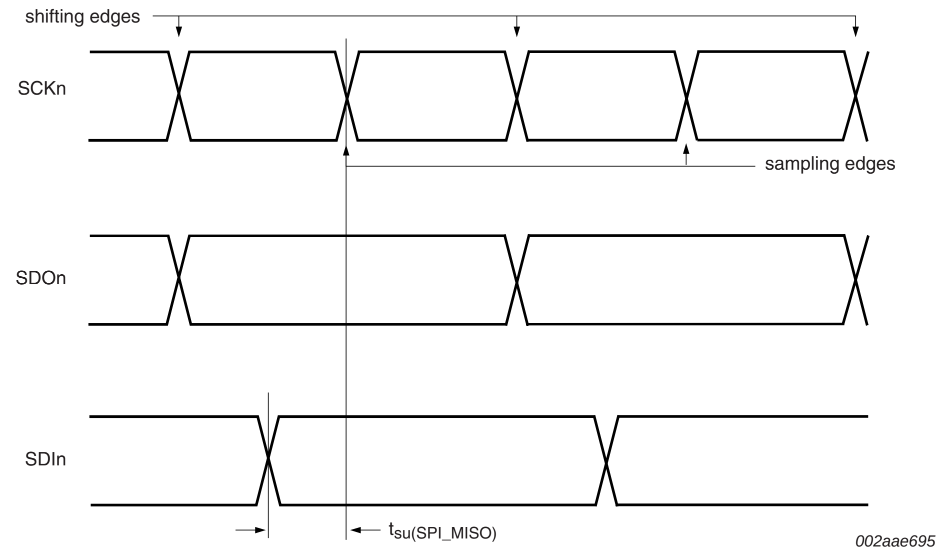


Fig 24. SPI data input set-up time in SSP Master mode

9.5 Dynamic characteristics: flash memory and EEPROM

Table 37. Flash characteristics

T_{amb} = −40 °C to +85 °C; $V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; all voltages are measured with respect to ground.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
N_{endu}	endurance	[1]	10000	-	-	cycles
t_{ret}	retention time	powered	10	-	-	years
		unpowered	20	-	-	years
t_{prog}	programming time	word	0.95	1	1.05	ms
t_{er}	erase time	global	95	100	105	ms
		sector	95	100	105	ms
t_{init}	initialization time		-	-	150	µs
$t_{wr(pg)}$	page write time		0.95	1	1.05	ms
$t_{fl(BIST)}$	flash word BIST time		-	38	70	ns
$t_{a(clk)}$	clock access time		-	-	63.4	ns
$t_{a(A)}$	address access time		-	-	60.3	ns

[1] Number of program/erase cycles.

Table 38. EEPROM characteristics

T_{amb} = −40 °C to +85 °C; $V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; all voltages are measured with respect to ground.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{clk}	clock frequency		200	375	400	kHz
N_{endu}	endurance		100000	500000	-	cycles
t_{ret}	retention time	powered	10	-	-	years

9.6 Dynamic characteristics: ADC1/2

Table 39. ADC dynamic characteristics

$V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; all voltages are measured with respect to ground.[1]

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$f_{i(ADC)}$	ADC input frequency	[2]	4	-	4.5	MHz
$f_{s(max)}$	maximum sampling rate	$f_{i(ADC)}$ = 4.5 MHz; $f_s = f_{i(ADC)} / (n + 1)$ with n = resolution				
		resolution 2 bit	-	-	1500	ksample/s
		resolution 10 bit	-	-	400	ksample/s
t_{conv}	conversion time	In number of ADC clock cycles	3	-	11	cycles
		In number of bits	2	-	10	bits

[1] All parameters are guaranteed over the virtual junction temperature range by design. Pre-testing is performed at T_{amb} = 85 °C ambient temperature on wafer level. Cased products are tested at T_{amb} = 25 °C (final testing). Both pre-testing and final testing use correlated test conditions to cover the specified temperature and power supply voltage range.

[2] Duty cycle clock should be as close as possible to 50 %.

10. Application information

10.1 Operating frequency selection

The LPC2921/2923/2925 is specified to operate at a maximum frequency of 125 MHz, maximum temperature of 85 °C, and maximum core voltage of 1.89 V. Figure 25 and Figure 26 show that the user can achieve higher operating frequencies for the LPC2921/2923/2925 by controlling the temperature and the core voltage accordingly.

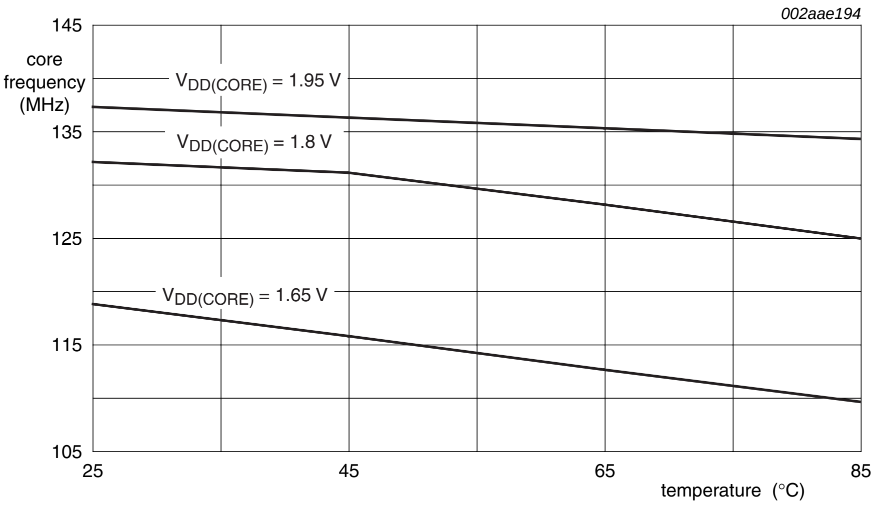


Fig 25. Core operating frequency versus temperature for different core voltages

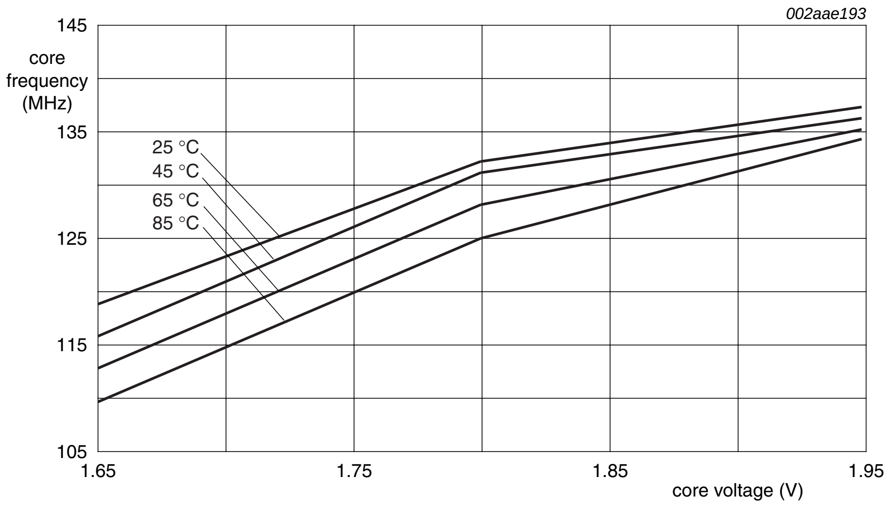


Fig 26. Core operating frequency versus core voltage for different temperatures

10.2 Suggested USB interface solutions

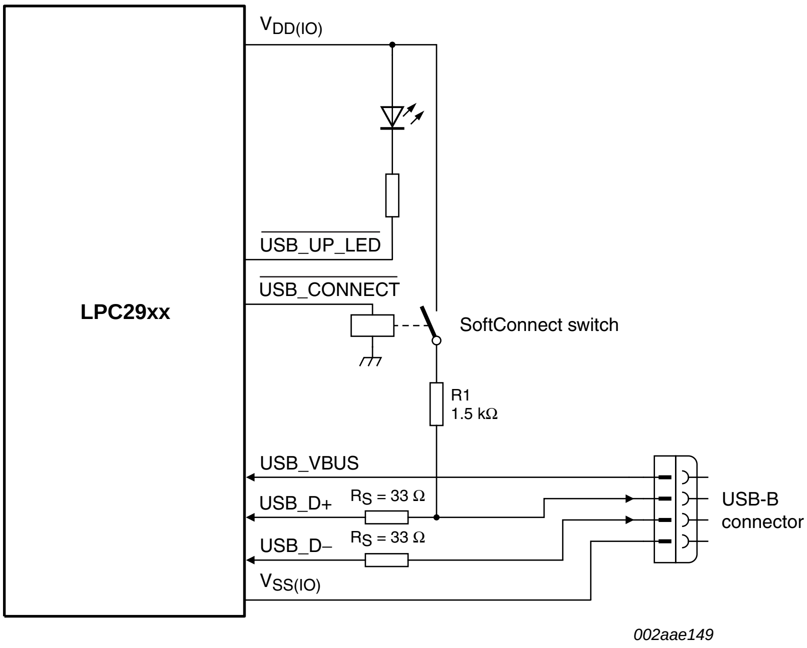


Fig 27. LPC2921/2923/2925 USB interface on a self-powered device

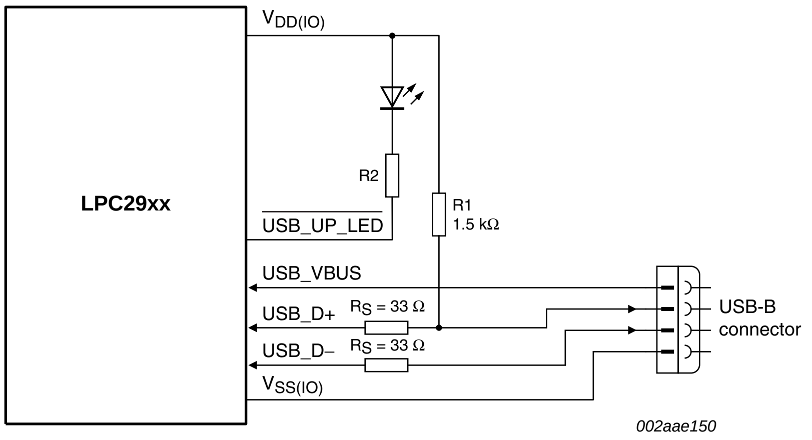


Fig 28. LPC2921/2923/2925 USB interface on a bus-powered device

10.3 SPI signal forms

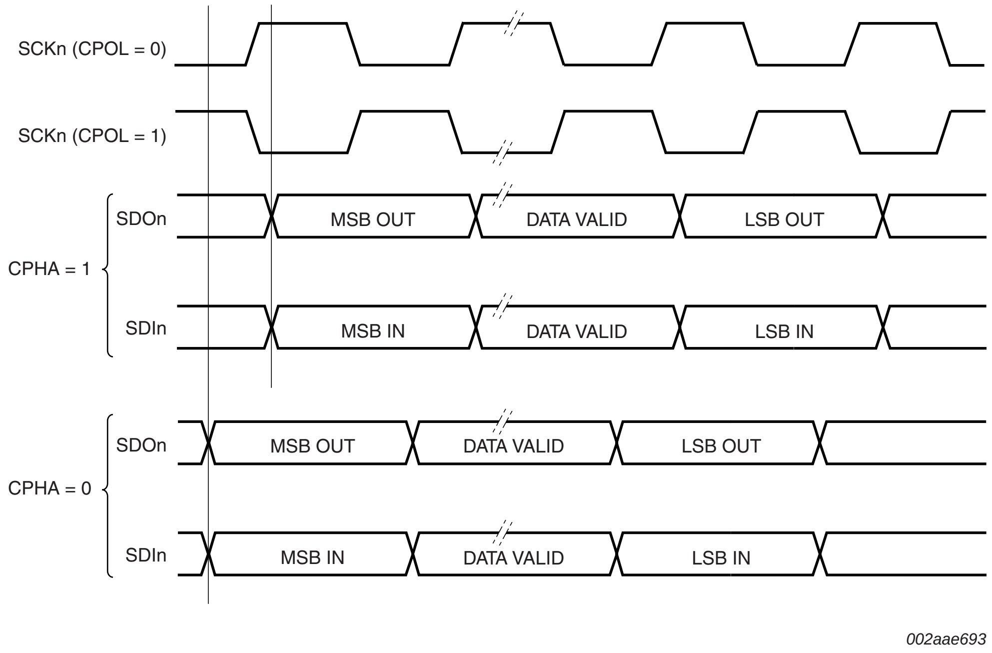


Fig 29. SPI timing in master mode

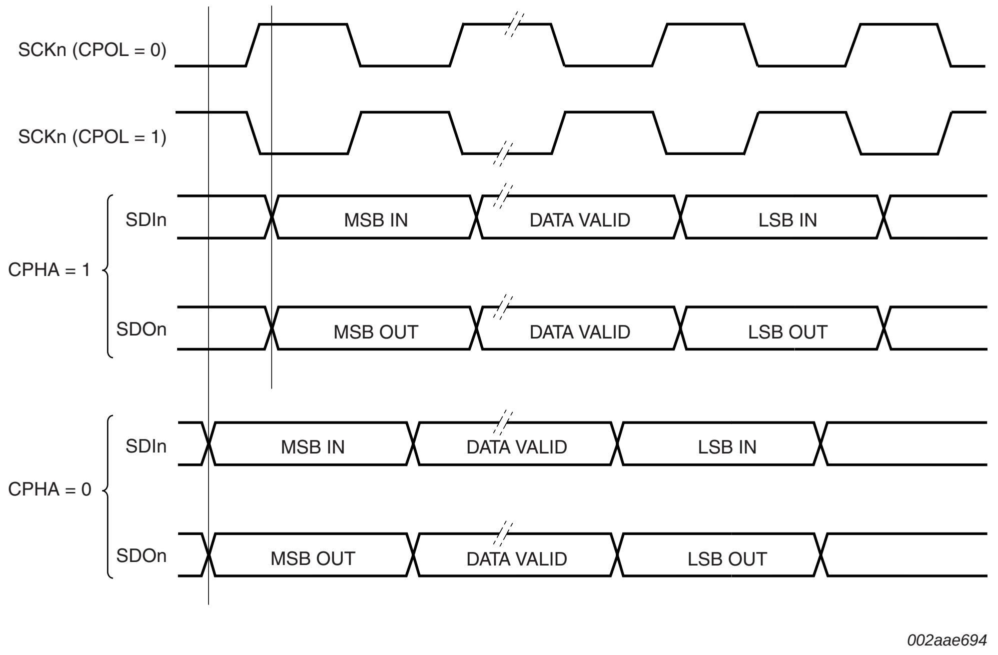


Fig 30. SPI timing in slave mode

10.4 XIN_OSC input

The input voltage to the on-chip oscillators is limited to 1.8 V. If the oscillator is driven by a clock in slave mode, it is recommended that the input be coupled through a capacitor with C_i = 100 pF. To limit the input voltage to the specified range, choose an additional capacitor to ground C_g which attenuates the input voltage by a factor $C_i/(C_i + C_g)$. In slave mode, a minimum of 200 mV (RMS) is needed. For more details see the *LPC29xx User manual UM10316*.

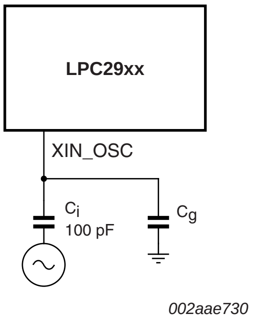


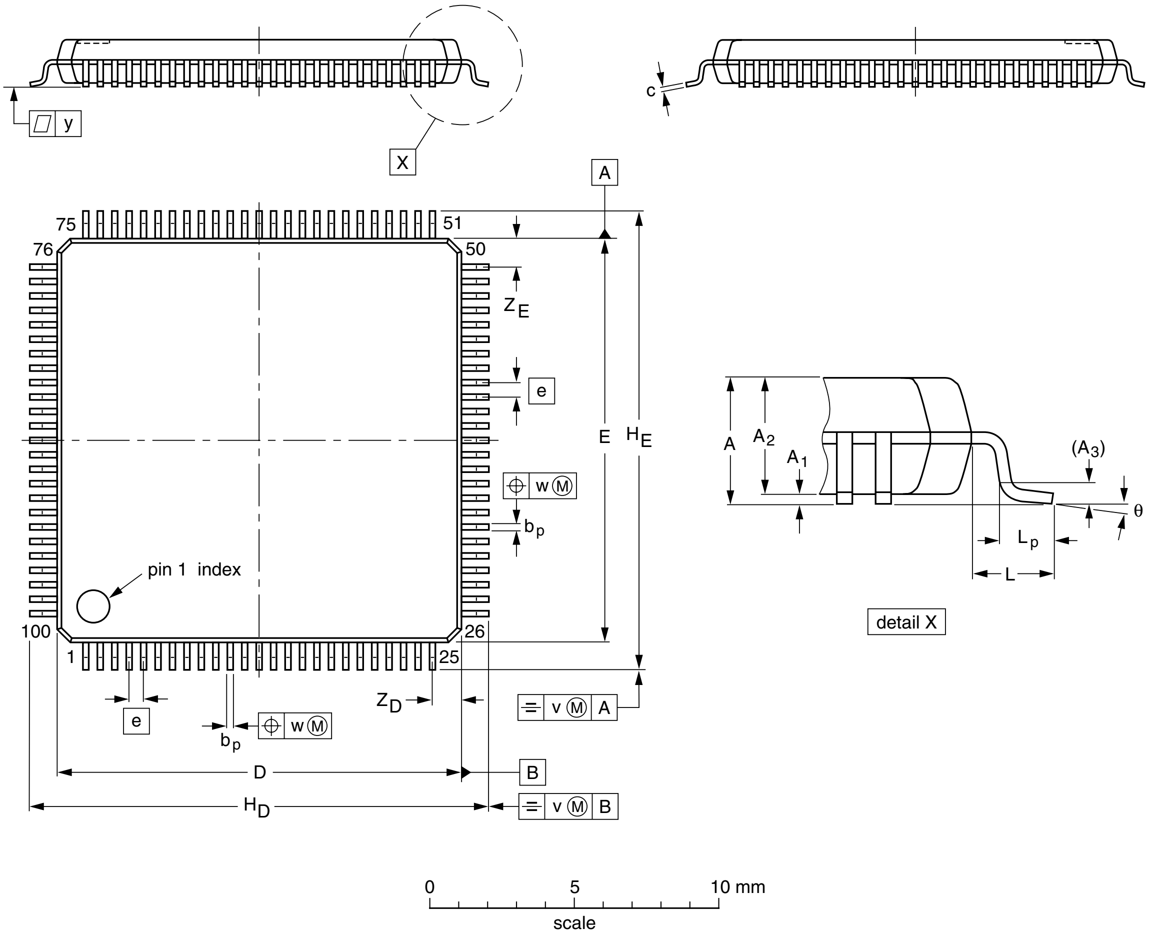
Fig 31. Slave mode operation of the on-chip oscillator

10.5 XIN_OSC Printed Circuit Board (PCB) layout guidelines

The crystal should be connected on the PCB as close as possible to the oscillator input and output pins of the chip. Take care that the load capacitors C_{x1} and C_{x2}, and C_{x3} in case of third overtone crystal usage, have a common ground plane. The external components must also be connected to the ground plain. Loops must be made as small as possible, in order to keep the noise coupled in via the PCB as small as possible. Also parasitics should stay as small as possible. Values of C_{x1} and C_{x2} should be chosen smaller accordingly to the increase in parasitics of the PCB layout.

11. Package outline

LQFP100: plastic low profile quad flat package; 100 leads; body 14 x 14 x 1.4 mm **SOT407-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	1.6	0.15 0.05	1.45 1.35	0.25	0.27 0.17	0.20 0.09	14.1 13.9	14.1 13.9	0.5	16.25 15.75	16.25 15.75	1	0.75 0.45	0.2	0.08	0.08	1.15 0.85	1.15 0.85	7° 0°

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT407-1	136E20	MS-026				~~00-02-01~~ 03-02-20

Fig 32. Package outline (LQFP100)

12. Soldering of SMD packages

This text provides a very brief insight into a complex technology. A more in-depth account of soldering ICs can be found in Application Note *AN10365 “Surface mount reflow soldering description”*.

12.1 Introduction to soldering

Soldering is one of the most common methods through which packages are attached to Printed Circuit Boards (PCBs), to form electrical circuits. The soldered joint provides both the mechanical and the electrical connection. There is no single soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and Surface Mount Devices (SMDs) are mixed on one printed wiring board; however, it is not suitable for fine pitch SMDs. Reflow soldering is ideal for the small pitches and high densities that come with increased miniaturization.

12.2 Wave and reflow soldering

Wave soldering is a joining technology in which the joints are made by solder coming from a standing wave of liquid solder. The wave soldering process is suitable for the following:

- Through-hole components
- Leaded or leadless SMDs, which are glued to the surface of the printed circuit board

Not all SMDs can be wave soldered. Packages with solder balls, and some leadless packages which have solder lands underneath the body, cannot be wave soldered. Also, leaded SMDs with leads having a pitch smaller than ~0.6 mm cannot be wave soldered, due to an increased probability of bridging.

The reflow soldering process involves applying solder paste to a board, followed by component placement and exposure to a temperature profile. Leaded packages, packages with solder balls, and leadless packages are all reflow solderable.

Key characteristics in both wave and reflow soldering are:

- Board specifications, including the board finish, solder masks and vias
- Package footprints, including solder thieves and orientation
- The moisture sensitivity level of the packages
- Package placement
- Inspection and repair
- Lead-free soldering versus SnPb soldering

12.3 Wave soldering

Key characteristics in wave soldering are:

- Process issues, such as application of adhesive and flux, clinching of leads, board transport, the solder wave parameters, and the time during which components are exposed to the wave
- Solder bath specifications, including temperature and impurities

12.4 Reflow soldering

Key characteristics in reflow soldering are:

- Lead-free versus SnPb soldering; note that a lead-free reflow process usually leads to higher minimum peak temperatures (see Figure 33) than a SnPb process, thus reducing the process window
- Solder paste printing issues including smearing, release, and adjusting the process window for a mix of large and small components on one board
- Reflow temperature profile; this profile includes preheat, reflow (in which the board is heated to the peak temperature) and cooling down. It is imperative that the peak temperature is high enough for the solder to make reliable solder joints (a solder paste characteristic). In addition, the peak temperature must be low enough that the packages and/or boards are not damaged. The peak temperature of the package depends on package thickness and volume and is classified in accordance with Table 40 and 41

Table 40. SnPb eutectic process (from J-STD-020C)

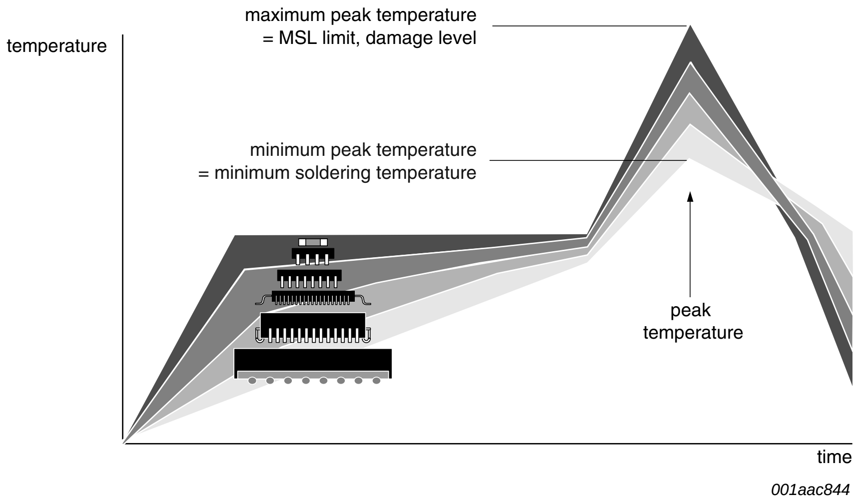
Package thickness (mm)	**Package reflow temperature (°C)**	
	Volume (mm^3)	
	< 350	**≥ 350**
< 2.5	235	220
≥ 2.5	220	220

Table 41. Lead-free process (from J-STD-020C)

Package thickness (mm)	**Package reflow temperature (°C)**		
	Volume (mm^3)		
	< 350	**350 to 2000**	**> 2000**
< 1.6	260	260	260
1.6 to 2.5	260	250	245
> 2.5	250	245	245

Moisture sensitivity precautions, as indicated on the packing, must be respected at all times.

Studies have shown that small packages reach higher temperatures during reflow soldering, see Figure 33.



MSL: Moisture Sensitivity Level

Fig 33. Temperature profiles for large and small components

For further information on temperature profiles, refer to Application Note *AN10365 “Surface mount reflow soldering description”*.

13. Abbreviations

Table 42. Abbreviations list

Abbreviation	Description
AF	Acceptance Filter
AHB	Advanced High-performance Bus
AMBA	Advanced Microcontroller Bus Architecture
APB	ARM Peripheral Bus
CCO	Current Controlled Oscillator
CISC	Complex Instruction Set Computers
DMA	Direct Memory Access
DSP	Digital Signal Processing
DTL	Device Transaction Level
EOP	End Of Packet
ETB	Embedded Trace Buffer
ETM	Embedded Trace Macrocell
FIQ	Fast Interrupt reQuest
GPDMA	General Purpose DMA
GPIO	General Purpose Input/Output
IRQ	Interrupt ReQuest
LIN	Local Interconnect Network
LUT	Look-Up Table
MAC	Media Access Control
MSC	Modulation and Sampling Control
PHY	PHYsical layer
PLL	Phase-Locked Loop
Q-SPI	Queued SPI
RISC	Reduced Instruction Set Computer
SCU	System Control Unit
SFSP	SCU Function Select Port
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus

14. References

[1] **UM10316 —** LPC29xx user manual

[2] **ARM —** ARM web site

[3] **ARM-SSP —** ARM primecell synchronous serial port (PL022) technical reference manual

[4] **CAN —** ISO 11898-1: 2002 road vehicles - Controller Area Network (CAN) - part 1: data link layer and physical signalling

[5] **LIN —** LIN specification package, revision 2.0

15. Revision history

Table 43. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
LPC2921_23_25_3	20100414	Product data sheet		LPC2921_23_25_2
Modifications:	• Section 1: Target market “medical” removed. • Document template updated. • USB logo added.			
LPC2921_23_25_2	20091208	Product data sheet	-	LPC2921_23_25_1
LPC2921_23_25_1	20090615	Preliminary data sheet	-	-

16. Legal information

16.1 Data sheet status

Document status[1][2]	Product status[3]	Definition
Objective [short] data sheet	Development	This document contains data from the objective specification for product development.
Preliminary [short] data sheet	Qualification	This document contains data from the preliminary specification.
Product [short] data sheet	Production	This document contains the product specification.

[1] Please consult the most recently issued document before initiating or completing a design.

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL http://www.nxp.com.

16.2 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Short data sheet — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

Product specification — The information and data provided in a Product data sheet shall define the specification of the product as agreed between NXP Semiconductors and its customer, unless NXP Semiconductors and customer have explicitly agreed otherwise in writing. In no event however, shall an agreement be valid in which the NXP Semiconductors product is deemed to offer functions and qualities beyond those described in the Product data sheet.

16.3 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on a weakness or default in the customer application/use or the application/use of customer's third party customer(s) (hereinafter both referred to as "Application"). It is customer's sole responsibility to check whether the NXP Semiconductors product is suitable and fit for the Application planned. Customer has to do all necessary testing for the Application in order to avoid a default of the Application and the product. NXP Semiconductors does not accept any liability in this respect.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

Non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

16.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

I²C-bus — logo is a trademark of NXP B.V.

17. Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

18. Contents

continued >>

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 14 April 2010

Document identifier: LPC2921_23_25_3