Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ARM9®
Core Size	16/32-Bit
Speed	125MHz
Connectivity	CANbus, I²C, LINbus, SPI, UART/USART, USB
Peripherals	DMA, POR, PWM, WDT
Number of I/O	60
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	16K x 8
RAM Size	24K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 3.6V
Data Converters	A/D 16x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/lpc2923fbd100-551

- 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.

- 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

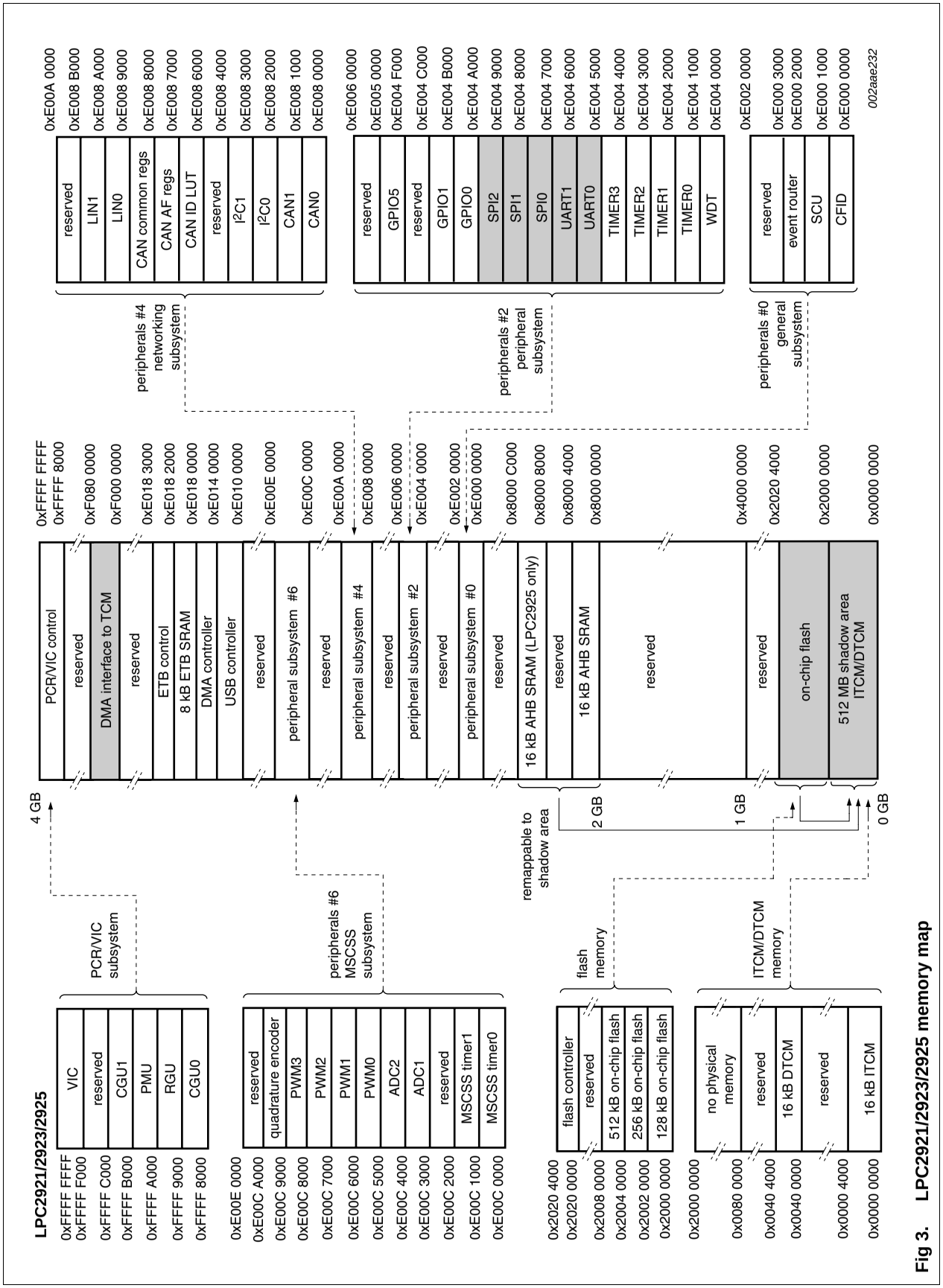


Fig 3. LPC2921/2923/2925 memory map

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

6.8 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

- 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.

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

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.

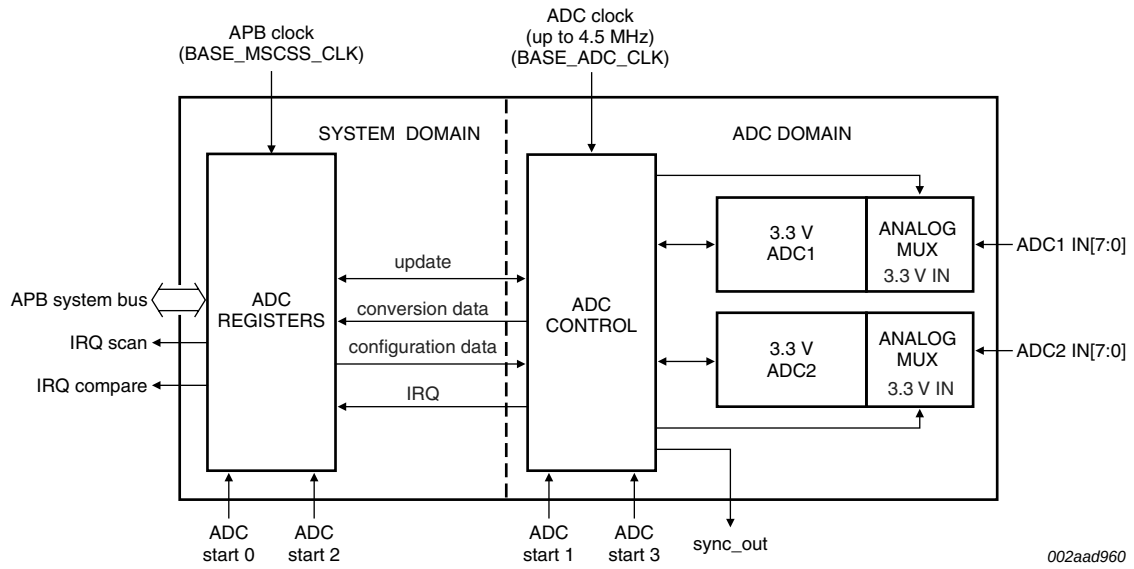


Fig 6. ADC block diagram

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.

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

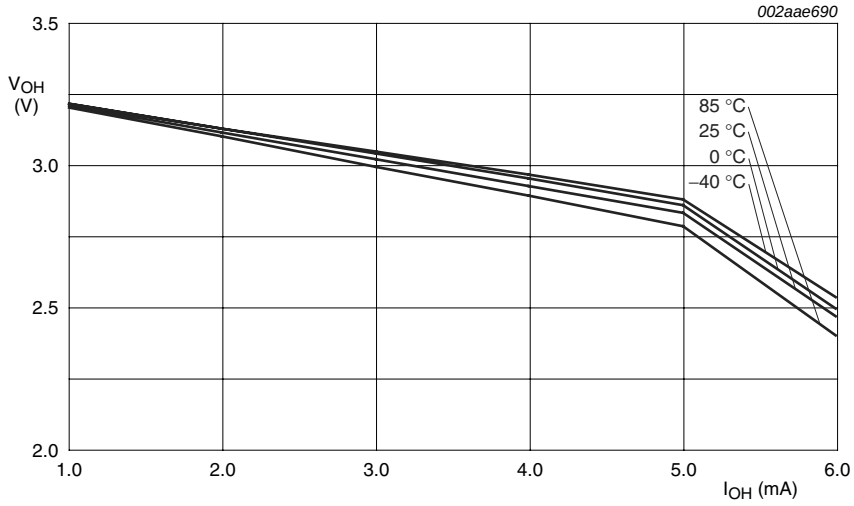
The Vectored Interrupt Controller routes incoming interrupt requests to the ARM processor. The interrupt target is configured for each interrupt request input of the VIC. The targets are defined as follows:

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

Table 31. Static characteristics *…continued*

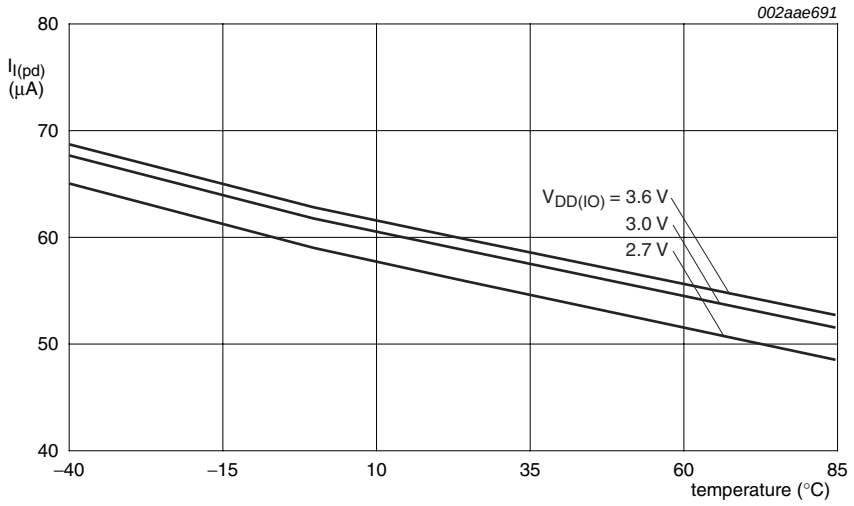
$V_{DD(CORE)}$ = $V_{DD(OSC\_PLL)}$; $V_{DD(IO)}$ = 2.7 V to 3.6 V; $V_{DDA(ADC3V3)}$ = 3.0 V to 3.6 V; T_{vj} = −40 °C to +85 °C; all voltages are measured with respect to ground; positive currents flow into the IC; unless otherwise specified.[1]

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



$V_{DD(IO)}$ = 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

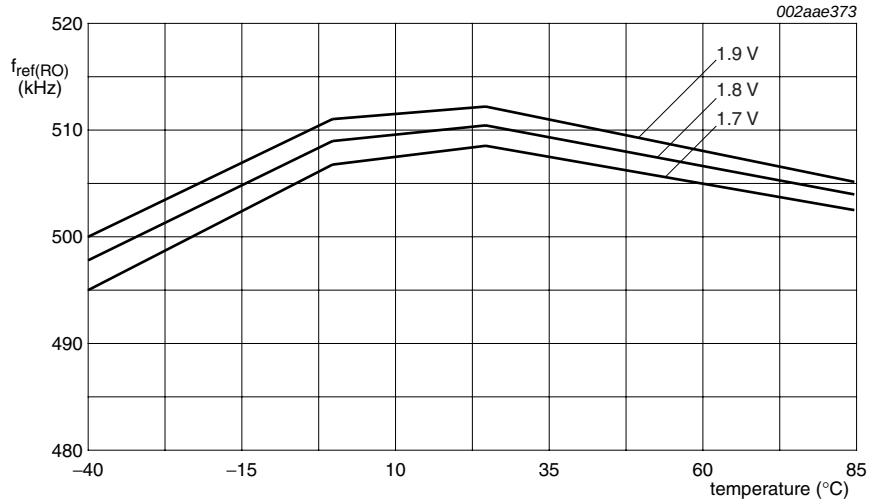


Fig 22. Low-power ring oscillator thermal characteristics

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	$^1/_{65024} f_{clk(SPI)}$	-	$^1/_2 f_{clk(SPI)}$	MHz
		slave operation	$^1/_{65024} f_{clk(SPI)}$	-	$^1/_4 f_{clk(SPI)}$	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.

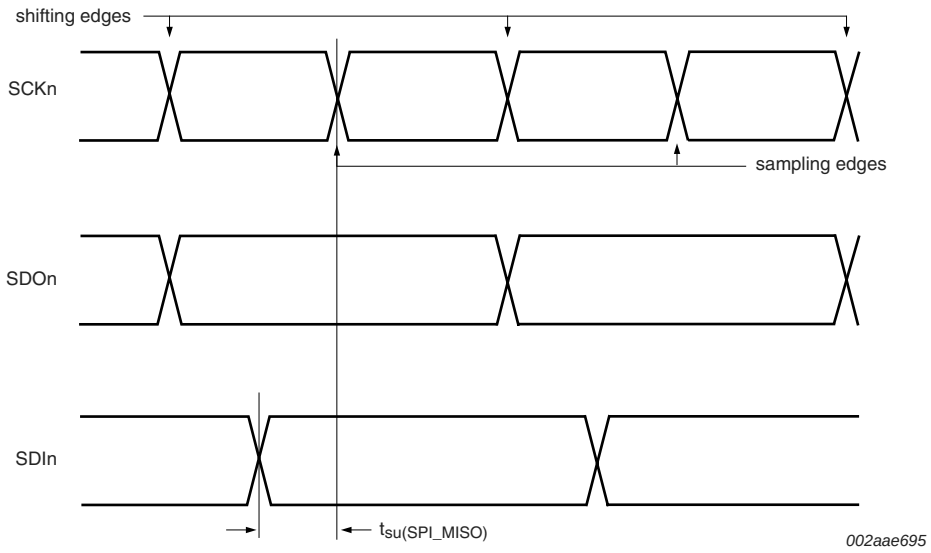


Fig 24. SPI data input set-up time in SSP Master mode

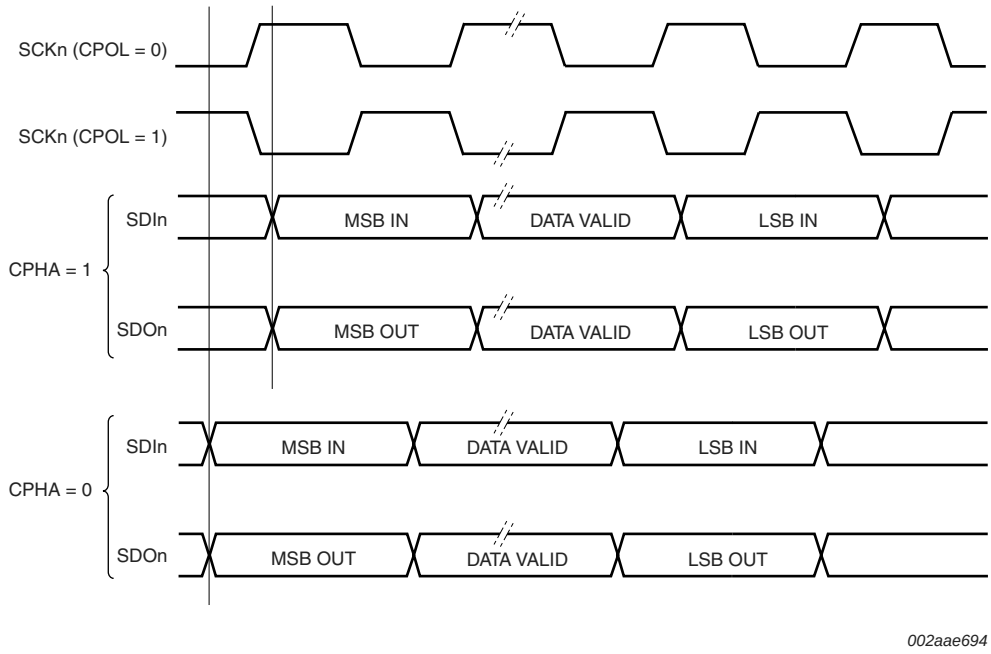


Fig 30. SPI timing in slave mode

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*.

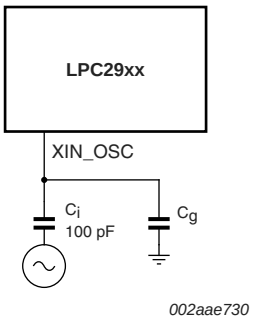


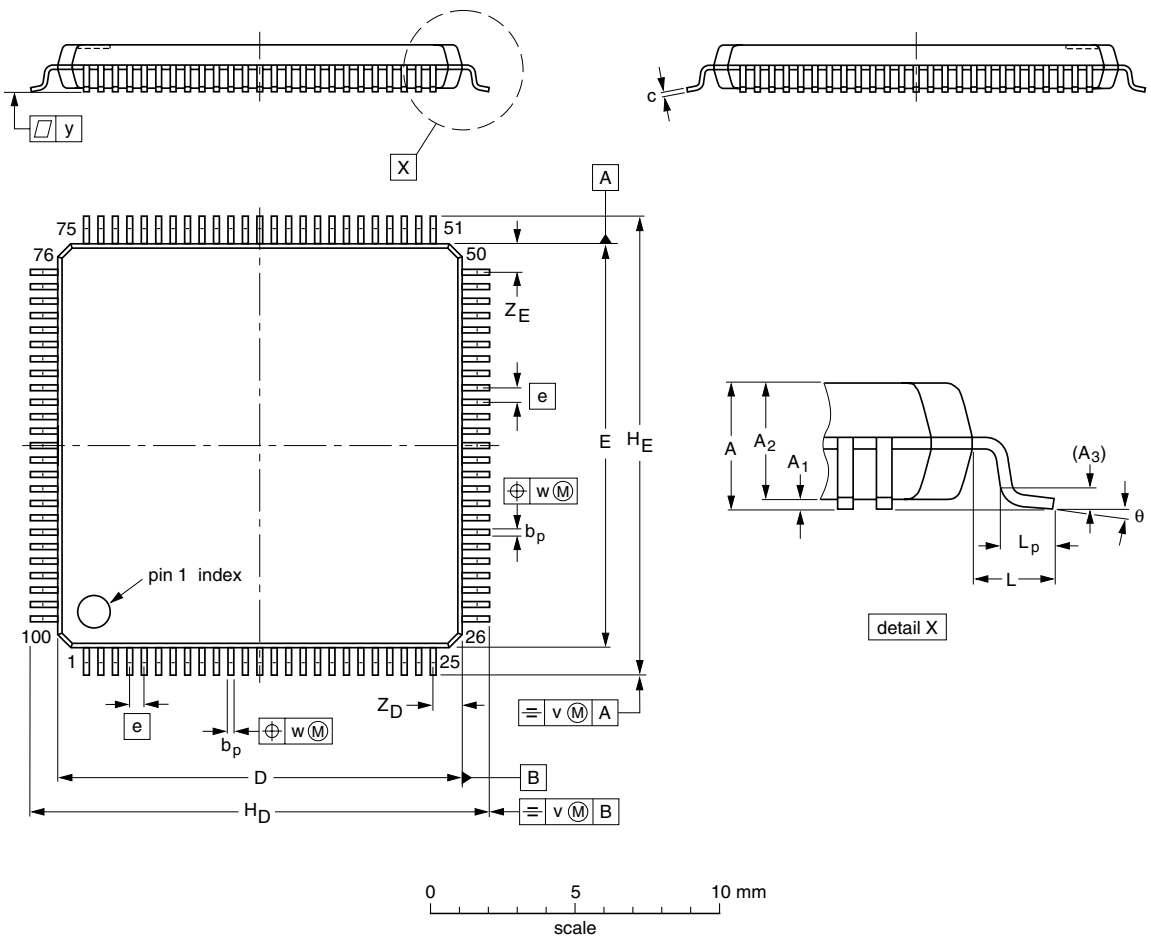
Fig 31. Slave mode operation of the on-chip oscillator

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^o 0^o

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

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	-	-