Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!


## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

# UM10736

## LPC15xx User manual

Rev. 1.2 — 28 December 2017 User manual

### Document information

| Info | Content |
|---|---|
| **Keywords** | LPC1500, LPC1500 User manual , LPC15xx UM, LPC1549, LPC1548, LPC1547, LPC1519, LPC1518, LPC1517 |
| **Abstract** | LPC15xx User manual |

## Revision history

| Rev | Date | Description |
|---|---|---|
| 1.2 | 20171228 | LPC15xx User manual |
| Modifications: | • Section 27.7.5.2 “Calculating the C_CAN bit rate” added.<br>• IAP entry location corrected in Section 34.9 “IAP commands”.<br>• Bit description of AUTOBAUD bit updated in Table 348 “USART Control register (CTL, address 0x4004 0004 (USART0), 0x4004 4004 (USART1), 0x400C 0004 (USART2)) bit description”: This bit can only be set when UART is enabled in the CFG register and is cleared when the UART is disabled.<br>• Clarified clock settings in the ADC_CONFIG_T structure of the ADC API (Section 42.4.12.1 “ADC_CONFIG_T channel configuration structure”): system_clock >= adc_clock.<br>• Embedded Trace Macrocell (ETM) support clarified in Section 43.2. This feature is not available.<br>• Watchdog interrupt flag polarity corrected: This flag is cleared by writing a 1 to the WDINT bit in the MOD register (Section 17.6.1 “Watchdog mode register”).<br>• Use of power profiles with IAP commands clarified. See Section 34.9 and Section 35.2.<br>• CCLK parameter in IAP calls replaced by NULL. See Section 34.9 “IAP commands”.<br>• For autobaud, set BRG to 0x0 (default value). Section 24.7.5.<br>• Section 24.7.5 “Autobaud function” corrected (clearance of AUTOBAUD bit).<br>• The IREF_PD in the PDRUNCFG register must be turned on to use the comparator. See Table 75 “Power configuration register (PDRUNCFG, address 0x4007 4208) bit description”, Table 74 “Wake-up configuration register (PDAWAKECFG, address 0x4007 4204) bit description”, and Section 29.3.<br>• VREFP/VERFN voltage selection requirements added in Section 28.4.<br>• RTC oscillator 32 kHz frequency clarified in Chapter 18 (exact frequency is 32.768 kHz).<br>• Added text after Table 344 “Endpoint commands”: For EP0 transfers, the hardware will do auto handshake as long as the ACTIVE bit is set in EP0_IN/OUT command list. Unlike other endpoints, the hardware will not clear the ACTIVE bit after transfer is done. Thus, the software should manually clear the bit whenever it receives new setup packet and set it only after it has queued the data for control transfer.<br>• Changed the bit field name in QEI Digital filter on index input register from FITLINX to FILTINX. See Table 315 “QEI Digital filter on index input register (FILTERINX, 0x4005 8044) bit description”<br>• Updated Figure 1 “Memory mapping”. Changed base address of C_CAN peripheral to 0x400E_0000.<br>• Deleted duplicate entry of the list UART transmit and receive functions can be operated with the system DMA controller. See Section 24.2 “Features”.<br>• Changed: REGMODEn = 0: Registers operate as match and reload registers and REGMODEn = 1: Registers operate as capture and capture control registers. See Section 15.6 “Register description” and Section 16.6 “Register description”.<br>• Updated Table 36 “Peripheral reset control register 1 (PRESETCTRL1, address 0x4007 4048) bit description”: Bit 28 is reserved.<br>• Removed references to WRAPEN field from Table 434 “ADC Input Select Register (INSEL, addresses 0x4000 0004 (ADC0) and 0x4008 0004 (ADC1)) bit description”.<br>• Added a remark to Table 216 “SCT DMA 0 request register (DMAREQ0, address 0x1C01 805C) bit description” and Table 217 “SCT DMA 1 request register (DMAREQ1, address 0x1C01 8060) bit description”.<br>• Added a remark to Table 249 “SCT DMA 0 request register (DMAREQ0, address 0x1C02 005C (SCT2) and 0x1C02 405C (SCT3)) bit description” and Table 250 “SCT DMA 1 request register (DMAREQ1, address 0x1C02 0060 (SCT2) and 0x1C02 4060 (SCT3)) bit description”.<br>• Fixed typographical errors in Chapter 15 “Large SCTimer/PWM (SCTimer0/PWM, SCTimer1/PWM)” and Chapter 16 “Small SCTimer/PWM (SCTimer2/PWM, SCTimer3/PWM)”.<br>• Changed SCTIPU_ABORT to SCTIPU_ABORT0, SCTIPU_ABORT1, and SCTIPU_ABORT2. See Table 127, Table 128, and Table 129. | |

**Revision history** *...continued*

| Rev | Date | Description |
|---|---|---|
| 1.1 | 20140303 | LPC15xx User manual |
| Modifications: | | • Section 5.4.2 “Criterion for valid user code” corrected: If the signature is not valid, the part enumerates as USB MSC. Also see Figure 7.<br>• Number of bits corrected in Table 228 “SCT event state mask registers 0 to 15 (EV[0:15]_STATE, addresses 0x1C01 8300 (EV0_STATE) to 0x1C01 8378 (EV15_STATE) (SCT0) and 0x1C01 C300 (EV0_STATE) to 0x1C01 C378 (EV15_STATE) (SCT1)) bit description”.<br>• Figure 11 “Example: Connect function U0_RXD and U0_TXD to pins” corrected.<br>• Bit description of bits TSEG1 and TSEG2 corrected in Table 397 “CAN bit timing register (BT, address 0x400F 000C) bit description” and Figure 83 “Bit timing” updated for clarity. |
| 1 | 20140213 | First LPC15xx User manual revision |

## Contact information

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

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

## 1.1 Introduction

The LPC15xx are ARM Cortex-M3 based microcontrollers for embedded applications featuring a rich peripheral set with very low power consumption. The ARM Cortex-M3 is a next generation core that offers system enhancements such as enhanced debug features and a higher level of support block integration.

The LPC15xx operate at CPU frequencies of up to 72 MHz. The ARM Cortex-M3 CPU incorporates a 3-stage pipeline and uses a Harvard architecture with separate local instruction and data buses as well as a third bus for peripherals. The ARM Cortex-M3 CPU also includes an internal prefetch unit that supports speculative branching.

The LPC15xx include up to 256 KB of flash memory, 32 KB of ROM, a 4 KB EEPROM, and up to 36 KB of SRAM. The peripheral compliment includes one full-speed USB 2.0 device, two SPI interfaces, three USARTs, one Fast-mode Plus $I^2C$-bus interface, one C_CAN module, PWM/timer subsystem with four configurable, multi-purpose State Configurable Timers (SCTimer/PWM) with input pre-processing unit, a Real-time clock module with independent power supply and a dedicated oscillator, two 12-channel/12-bit, 2 Msamples/sec ADCs, one 12-bit, 500 kSamples/sec DAC, four voltage comparators with internal voltage reference, and a temperature sensor. A DMA engine can service most peripherals.

For additional documentation, see Section 45.2 “References”.

## 1.2 Features

- System:
  - ARM Cortex-M3 processor, running at frequencies of up to 72 MHz.
  - ARM Cortex-M3 built-in Nested Vectored Interrupt Controller (NVIC).
  - System tick timer.
  - Serial Wire Debug (SWD) with four breakpoints and two watchpoints.
  - Single-cycle multiplier supported.
  - Memory Protection Unit (MPU) included.
- Memory:
  - Up to 256 kB on-chip flash programming memory with 256 Byte page write and erase.
  - Up to 36 kB SRAM.
  - 4 kB EEPROM.

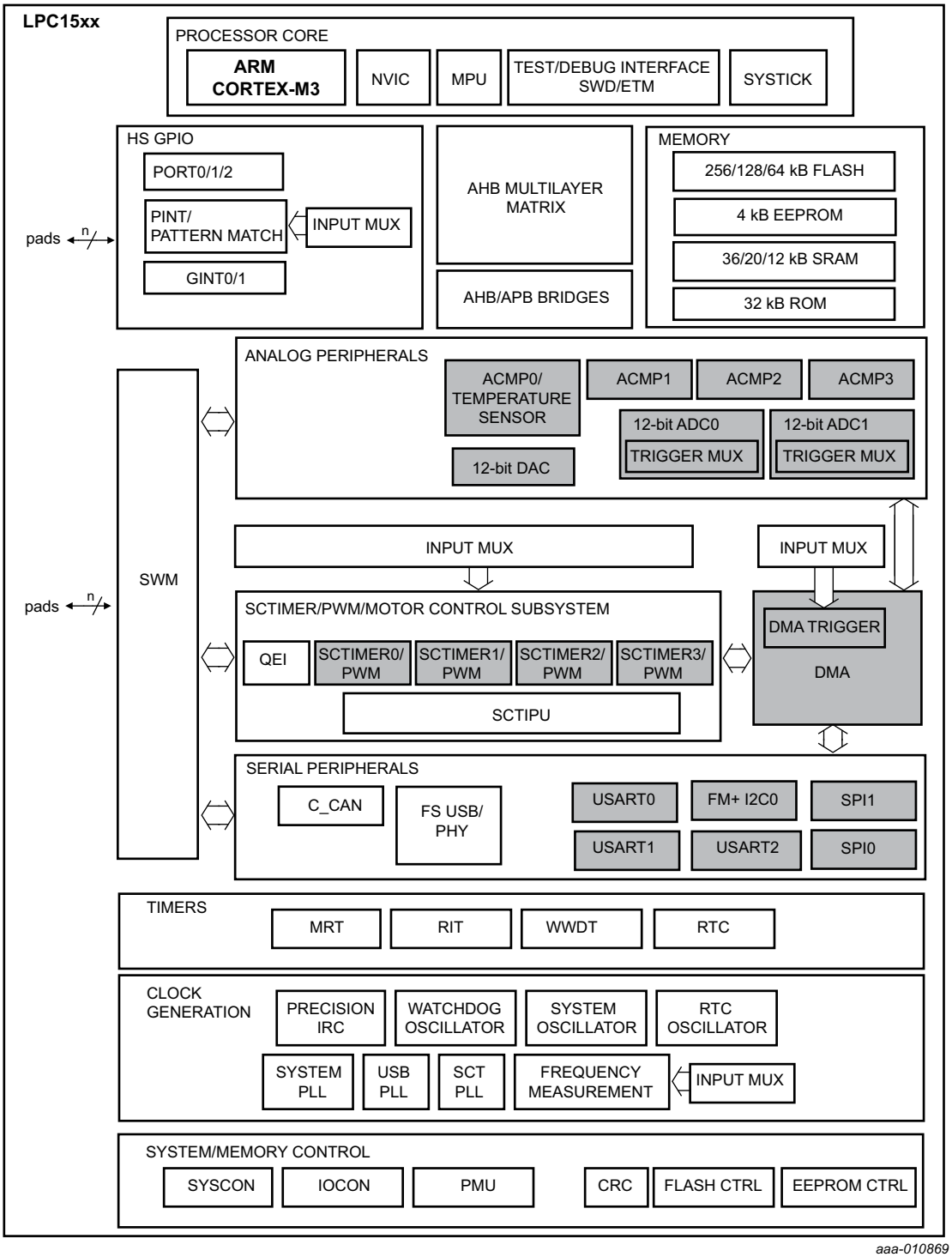
- ROM API support:
  - Boot loader with boot options from flash or external source via USART, C_CAN, or USB
  - USB drivers
  - ADC drivers
  - SPI drivers
  - USART drivers
  - I2C drivers
  - Power profiles and power mode configuration with low-power mode configuration option
  - DMA drivers
  - C_CAN drivers
  - Flash In-Application Programming (IAP) and In-System Programming (ISP).
- Digital peripherals:
  - Simple DMA engine with 18 channels and 20 programmable input triggers.
  - High-speed GPIO interface with up to 76 General-Purpose I/O (GPIO) pins with configurable pull-up/pull-down resistors, open-drain mode, input inverter, and programmable digital glitch filter.
  - GPIO interrupt generation capability with boolean pattern-matching feature on eight external inputs.
  - Two GPIO grouped port interrupts.
  - Switch matrix for flexible configuration of each I/O pin function.
  - CRC engine.
  - Quadrature Encoder Interface (QEI).
- Configurable PWM/timer/motor control subsystem:
  - Up to four 32-bit counter/timers or up to eight 16-bit counter/timers or combinations of 16-bit and 32-bit timers.
  - Up to 28 match outputs and 22 configurable capture inputs with input multiplexer.
  - Dither engine for improved average resolution of pulse edges.
  - Four State Configurable Timers (SCTimers) for highly flexible, event-driven timing and PWM applications.
  - SCT Input Pre-processor Unit (SCTIPU) for processing timer inputs and immediate handling of abort situations.
  - Integrated with ADC threshold compare interrupts, temperature sensor, and analog comparator outputs for motor control feedback using analog signals.
- Special-application and simple timers:
  - 24-bit, four-channel, multi-rate timer (MRT) for repetitive interrupt generation at up to four programmable, fixed rates.
  - Repetitive interrupt timer for general purpose use.
  - Windowed Watchdog timer (WWDT).

  - High-resolution 32-bit Real-time clock (RTC) with selectable 1 s or 1 ms time resolution running in the always-on power domain. RTC can be used for wake-up from all low power modes including Deep power-down.

- Analog peripherals:
  - Two 12-bit ADC with up to 12 input channels per ADC and with multiple internal and external trigger inputs and sample rates of up to 2 Msamples/s. Each ADC supports two independent conversion sequences. ADC conversion clock can be the system clock or an asynchronous clock derived from one of the three PLLs.
  - One 12-bit DAC.
  - Integrated temperature sensor and band gap internal reference voltage.
  - Four comparators with external and internal voltage references (ACMP0 to 3). Comparator outputs are internally connected to the SCTimer/PWMs and ADCs and externally to pins. Each comparator output contains a programmable glitch filter.
- Serial interfaces:
  - Three USART interfaces with DMA, RS-485 support, auto-baud, and with synchronous mode and 32 kHz mode for wake-up from Deep-sleep and Power-down modes. The USARTs share a fractional baud-rate generator.
  - Two SPI controllers.
  - One $I^2C$-bus interface supporting fast mode and Fast-mode Plus with data rates of up to 1Mbit/s and with multiple address recognition and monitor mode.
  - One C_CAN controller.
  - One USB 2.0 full-speed device controller with on-chip PHY.
- Clock generation:
  - 12 MHz internal RC oscillator trimmed to 1 % accuracy for $-25\ °C \leq T_{amb} \leq +85\ °C$ that can optionally be used as a system clock.
  - Crystal oscillator with an operating range of 1 MHz to 25 MHz.
  - Watchdog oscillator running at the fixed frequency of 503 kHz.
  - 32 kHz low-power RTC oscillator with 32 kHz, 1 kHz, and 1 Hz outputs.
  - System PLL allows CPU operation up to the maximum CPU rate without the need for a high-frequency crystal. May be run from the system oscillator or the internal RC oscillator.
  - Two additional PLLs for generating the USB and SCTimer/PWM clocks.
  - Clock output function with divider that can reflect the crystal oscillator, the main clock, the IRC, or the watchdog oscillator.
- Power control:
  - Integrated PMU (Power Management Unit) to minimize power consumption.
  - Reduced power modes: Sleep mode, Deep-sleep mode, Power-down mode, and Deep power-down mode.
  - APIs provided for optimizing power consumption in active and sleep modes and for configuring Deep-sleep, Power-down, and Deep power-down modes.
  - Wake-up from Deep-sleep and Power-down modes on activity on USB, USART, SPI, and I2C peripherals.

  - Wake-up from Sleep, Deep-sleep, Power-down, and Deep power-down modes from the RTC alarm or wake-up interrupts.
  - Timer-controlled self wake-up from Deep power-down mode using the RTC high-resolution/wake-up 1 kHz timer.
  - Power-On Reset (POR).
  - BrownOut Detect BOD).
- JTAG boundary scan modes supported.
- Unique device serial number for identification.
- Single power supply 2.4 V to 3.6 V.
- Temperature range -40 °C to +105 °C.
- Available as LQFP100, LQFP64, and LQFP48 packages.

## 1.3 Block diagram




Grey-shaded blocks show peripherals that can provide hardware triggers for DMA transfers or have DMA request lines.

**Fig 1. Block diagram**

## 1.4 Functional description

### 1.4.1 Architectural overview

The ARM Cortex-M3 includes three AHB-Lite buses, one system bus and the I-code and D-code buses. One bus is dedicated for instruction fetch (I-code), and one bus is dedicated for data access (D-code). The use of two core buses allows for simultaneous operations if concurrent operations target different devices.

A multi-layer AHB matrix connects the Cortex-M3 buses and other bus masters to peripherals in a flexible manner that optimizes performance by allowing peripherals on different slaves ports of the matrix to be accessed simultaneously by different bus masters. Details of the multilayer matrix connections are shown in Figure 2.

APB peripherals are connected to the CPU via two APB buses using separate slave ports from the multilayer AHB matrix. This allows for better performance by reducing collisions between the CPU and the DMA controller. The APB bus bridges are configured to buffer writes so that the CPU or DMA controller can write to APB devices without always waiting for APB write completion.

### 1.4.2 ARM Cortex-M3 processor

The ARM Cortex-M3 is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The Cortex-M3 includes a Thumb-2 instruction set and provides low interrupt latency, hardware divide, interruptible/continuable multiple load and store instructions, automatic state save and restore for interrupts, tightly integrated interrupt controller, and multiple core buses capable of simultaneous accesses.

Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

#### 1.4.2.1 Cortex-M3 Configuration Options

The LPC15xx uses the r2p1 version of the Cortex-M3 CPU, which includes a number of configurable options, as noted below.

**System options:**

- The Nested Vectored Interrupt Controller (NVIC) is included.
- SYSTICK timer is included.
- Single-cycle multiplier supported.
- A Memory Protection Unit (MPU) is included.
- A ROM Table in included. The ROM Table provides addresses of debug components to external debug systems.

**Debug related options:**

- Serial Wire Debug is included. Serial Wire Debug allows debug operations using only two wires, simple trace functions can be added with a third wire.

- The Data Watchpoint and Trace (DWT) unit is included. The DWT allows data address or data value matches to be trace information or trigger other events. The DWT includes four comparators and counters for certain internal events.
- An Instrumentation Trace Macrocell (ITM) is included. Software can write to the ITM in order to send messages to the trace port.
- The Trace Port Interface Unit (TPIU) is included. The TPIU encodes and provides trace information to the outside world using the serial wire output pin function.

### 1.4.3 PWM/timer/motor control subsystem

The SCTs (State Configurable Timers) and the analog peripherals support multiple ways of interconnecting their inputs and outputs and of interfacing to the pins and the DMA controller. Using the highly flexible and programmable connection scheme makes it easy to configure various subsystems for motor control and complex timing and tracking applications. Specifically, the inputs to the SCTs and the trigger inputs of the ADCs and DMA are selected through the input mux which offers a choice of many possible sources for each input or trigger. SCT outputs are assigned to pins through the switch matrix allowing for many pinout solutions.

#### 1.4.3.1 PWW/timer subsystem

The SCTs can be configured to build a PWM controller with multiple outputs by programming the MATCH and MATCHRELOAD registers of the SCTs to control the base frequency and the duty cycle of each SCT output. More complex waveforms that span multiple counter cycles or change behavior across or within counter cycles can be generated using the state capability built into the SCT timers.

Combining the PWM functions with the analog functions, the PWM output can react to control signals like comparator outputs or the ADC interrupts. The SCT IPU adds emergency shut-down functions and pre-processing of controlling events. For an overview of the PWM subsystem, see Figure 2 “PWM-Analog subsystem”.

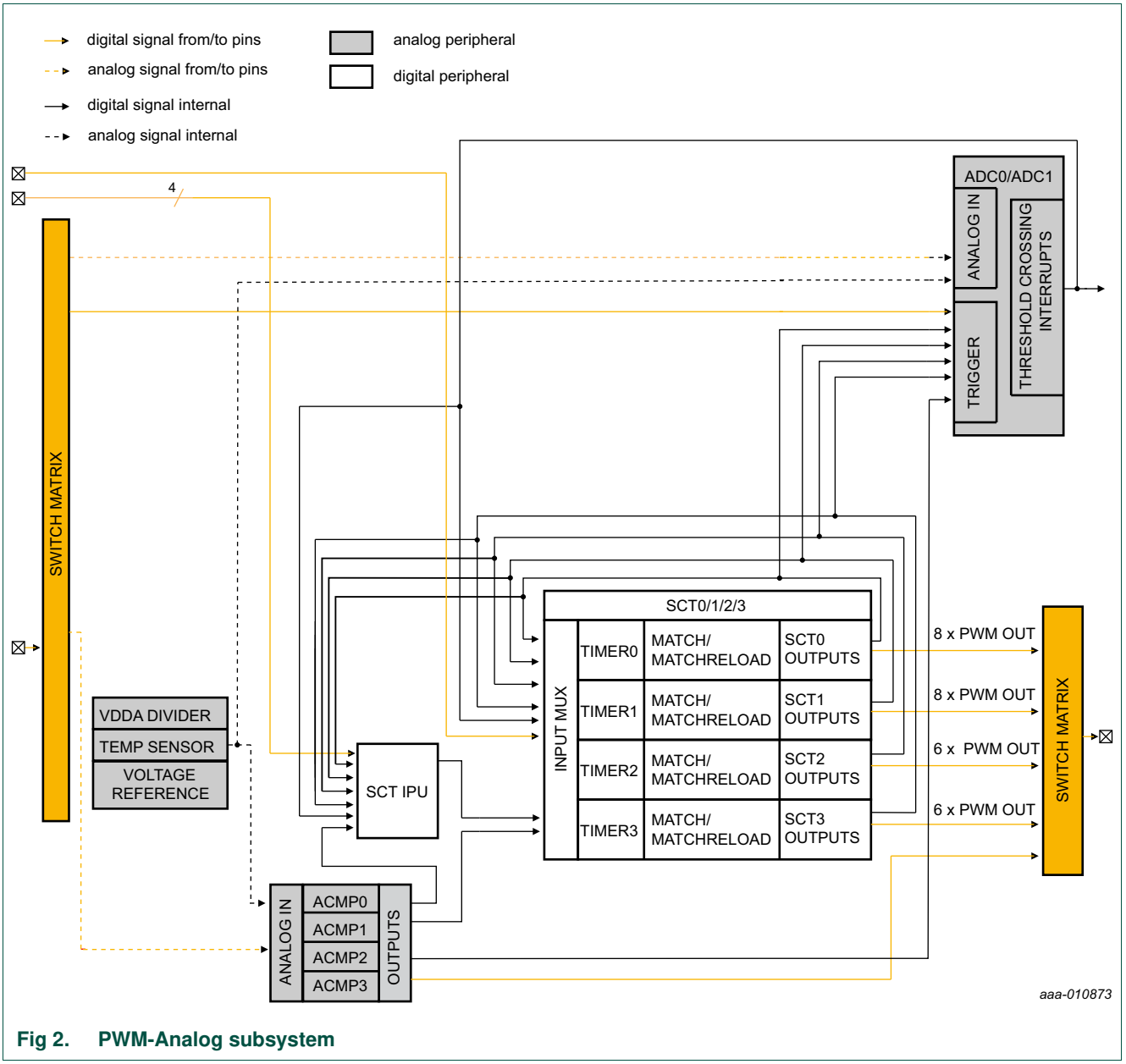



**Fig 2. PWM-Analog subsystem**

### 1.4.3.2 Timer controlled subsystem

The timers, the analog components, and the DMA can be configured to form a subsystem that can run independently of the main processor under the control of the SCTs and any events that are generated by the A/D converters, the comparators, the SCT output themselves, or the external pins. A/D conversions can be triggered by the timer outputs, the comparator outputs or by events from external pins. Data can be transferred from the ADCs to memory using the DMA controller, and the DMA transfers can be triggered by the ADCs, the comparator outputs, or by the timer outputs.

For an overview of the subsystem, see Figure 3 “Subsystem with timers, switch matrix, DMA, and analog components”.

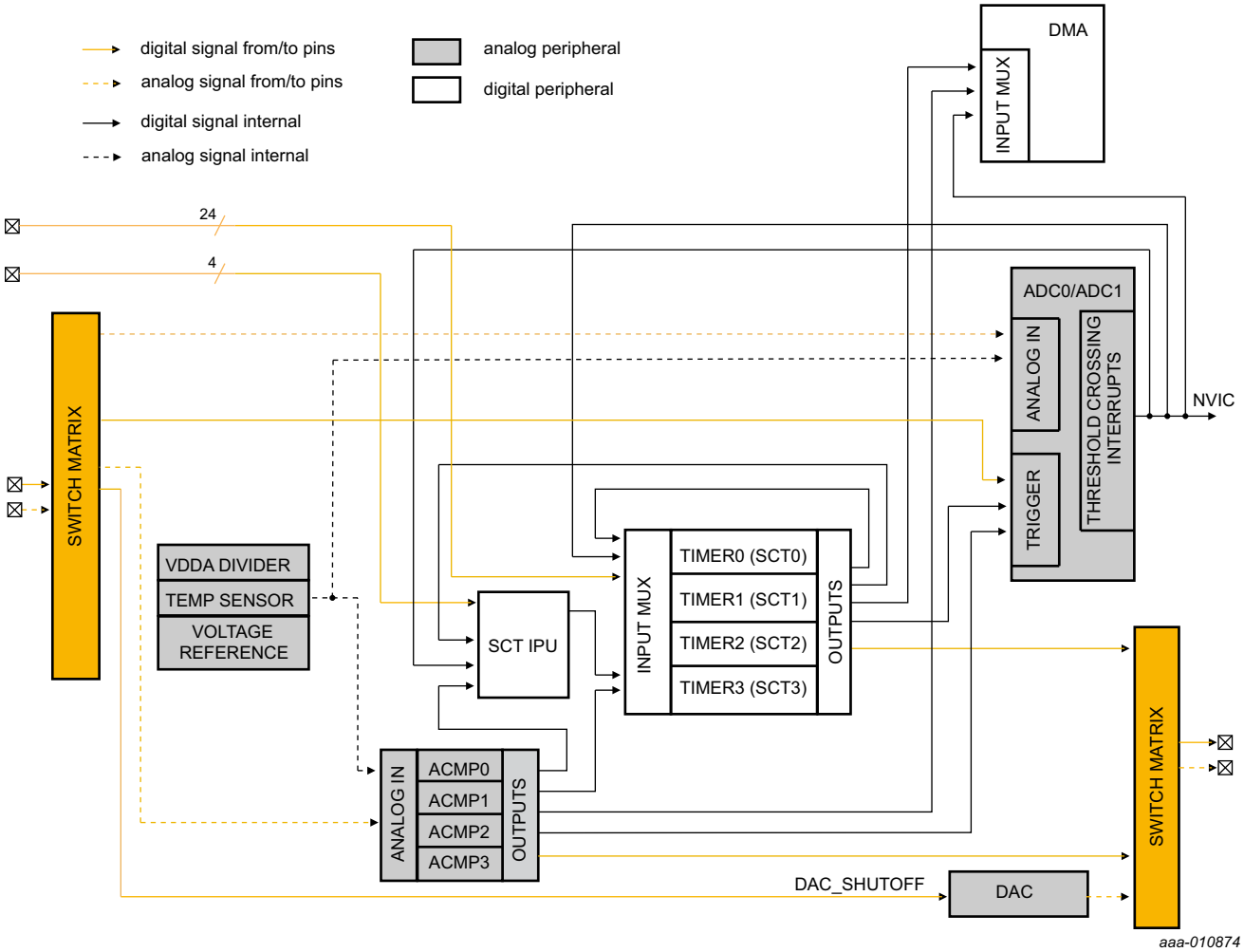



**Fig 3. Subsystem with timers, switch matrix, DMA, and analog components**

## 1.1 How to read this chapter

USB is not available on parts LPC1519/18/17.

## 1.2 General description

The LPC15xx incorporates several distinct memory regions. Figure 1 shows the overall map of the entire address space from the user program viewpoint following reset.

The APB peripheral area is 2 x 512 kB in size and is divided to allow for two blocks of up to 32 peripherals. Each peripheral is allocated 16 kB of space simplifying the address decoding.

The registers incorporated into the ARM Cortex-M3 core, such as NVIC, SysTick, and sleep mode control, are located on the private peripheral bus.

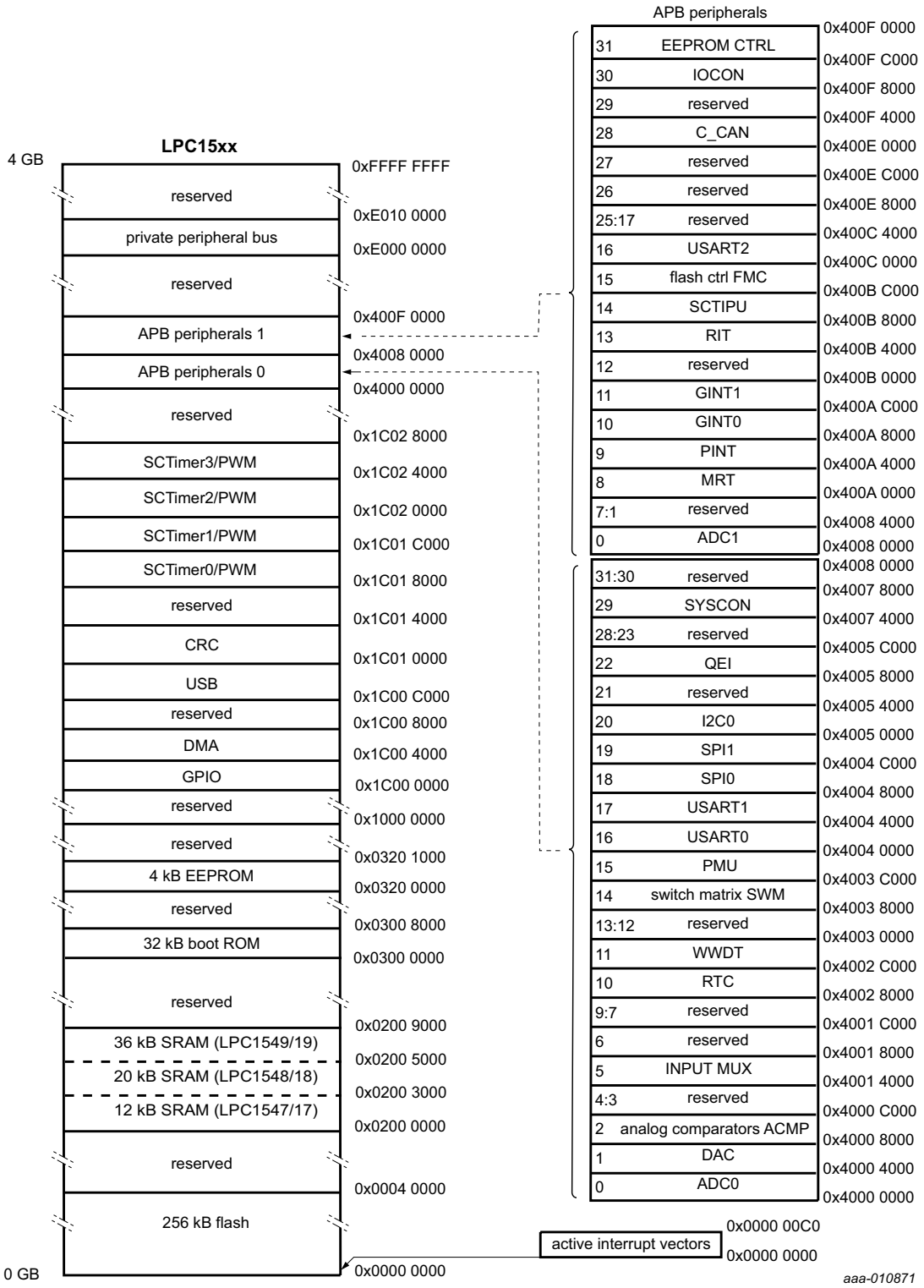
### 1.2.1 SRAM

The parts contain a total 36 kB, 20 kB or 12 kB of contiguous, on-chip static RAM memory. For each SRAM configuration, the SRAM is divided into three blocks: 2 x 16 kB + 4 kB for 36 kB SRAM, 2 x 8 kB + 4 kB for 20 kB SRAM, and 2 x 4 kB + 4 kB for 12 kB SRAM. The bottom 16 kB, 8 kB, or 4 kB are enabled by the boot loader and cannot be disabled. The next two SRAM blocks in each configuration can be disabled or enabled individually in the SYSCON block to save power. See Section 3.6.22 “System clock control register 0”.

**Table 1. LPC15xx SRAM configurations**

| | SRAM0 | SRAM1 | SRAM2 |
|---|---|---|---|
| **LPC1549/19 (total SRAM = 36 kB)** | | | |
| Address range | 0x0200 0000 to 0x0200 3FFF | 0x0200 4000 to 0x0200 7FFF | 0x0200 8000 to 0x0200 8FFF |
| Size | 16 kB | 16 kB | 4 kB |
| Control | cannot be disabled | disable/enable | disable/enable |
| Default | enabled | enabled | enabled |
| **LPC1548/18 (total SRAM = 20 kB)** | | | |
| Address range | 0x0200 0000 to 0x0200 1FFF | 0x0200 2000 to 0x0200 3FFF | 0x0200 4000 to 0x0200 4FFF |
| Size | 8 kB | 8 kB | 4 kB |
| Control | cannot be disabled | disable/enable | disable/enable |
| Default | enabled | enabled | enabled |
| **LPC1547/17 (total SRAM = 12 kB)** | | | |
| Address range | 0x0200 0000 to 0x0200 0FFF | 0x0200 1000 to 0x0200 1FFF | 0x0200 2000 to 0x0200 2FFF |
| Size | 4 kB | 4 kB | 4 kB |
| Control | cannot be disabled | disable/enable | disable/enable |
| Default | enabled | enabled | enabled |

### 1.2.2 Memory mapping




The private peripheral bus includes the ARM Cortex-M3 peripherals such as the NVIC, SysTick, and the core control registers.

**Fig 1. Memory mapping**

### 1.2.3 AHB multilayer matrix

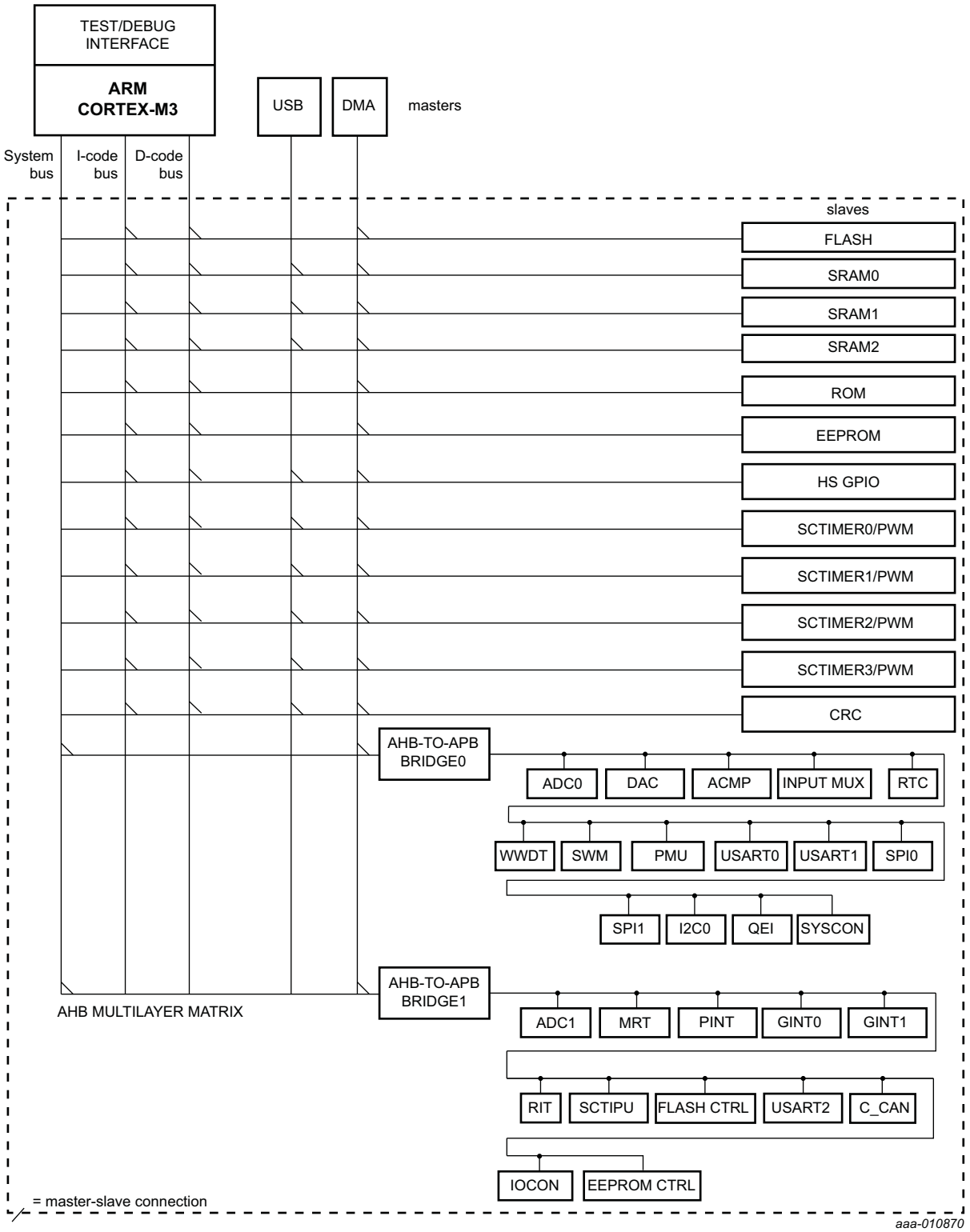



**Fig 2. AHB multilayer matrix connections**

### 1.2.4 Memory Protection Unit (MPU)

The Cortex-M3 processor has a memory protection unit (MPU) that provides fine grain memory control, enabling applications to implement security privilege levels, separating code, data and stack on a task-by-task basis. Such requirements are critical in many embedded applications.

The MPU register interface is located on the private peripheral bus and is described in detail in Ref. 1.

## 2.1 How to read this chapter

USB is available on parts LPC1549/48/47.

## 2.2 Features

- Nested Vectored Interrupt Controller that is an integral part of the ARM Cortex-M3.
- Tightly coupled interrupt controller provides low interrupt latency.
- Controls system exceptions and peripheral interrupts.
- The NVIC supports 47 vectored interrupts.
- Eight programmable interrupt priority levels with hardware priority level masking.
- Software interrupt generation using the ARM exceptions SVCall and PendSV.
- Support for NMI.
- ARM Cortex-M3 Vector table offset register VTOR implemented.

## 2.3 General description

The Nested Vectored Interrupt Controller (NVIC) is an integral part of the Cortex-M3. The tight coupling to the CPU allows for low interrupt latency and efficient processing of late arriving interrupts.

### 2.3.1 Interrupt sources

Table 2 lists the interrupt sources for each peripheral function. Each peripheral device may have one or more interrupt lines to the Vectored Interrupt Controller. Each line may represent more than one interrupt source. The interrupt number does not imply any interrupt priority.

See Ref. 1 for a detailed description of the NVIC and the NVIC register description.

**Table 2. Connection of interrupt sources to the NVIC**

| Interrupt number | Name | Description | Flags |
|---|---|---|---|
| 0 | WDT | Windowed watchdog timer interrupt | WARNINT - watchdog warning interrupt |
| 1 | BOD | BOD interrupt | BODINTVAL - BOD interrupt level |
| 2 | FLASH | Flash controller | - |
| 3 | EE | EEPROM controller interrupt | - |
| 4 | DMA | DMA | Interrupt A and interrupt B, error interrupt |
| 5 | GINT0 | GPIO group0 interrupt | Enabled pin interrupts |
| 6 | GINT1 | GPIO group1 interrupt | Enabled pin interrupts |

**Table 2. Connection of interrupt sources to the NVIC**

| Interrupt number | Name | Description | Flags |
|---|---|---|---|
| 7 | PIN_INT0 | Pin interrupt 0 or pattern match engine slice 0 interrupt | PSTAT - pin interrupt status |
| 8 | PIN_INT1 | Pin interrupt 1 or pattern match engine slice 1 interrupt | PSTAT - pin interrupt status |
| 9 | PIN_INT2 | Pin interrupt 2 or pattern match engine slice 2 interrupt | PSTAT - pin interrupt status |
| 10 | PIN_INT3 | Pin interrupt 3 or pattern match engine slice 3 interrupt | PSTAT - pin interrupt status |
| 11 | PIN_INT4 | Pin interrupt 4 or pattern match engine slice 4 interrupt | PSTAT - pin interrupt status |
| 12 | PIN_INT5 | Pin interrupt 5 or pattern match engine slice 5 interrupt | PSTAT - pin interrupt status |
| 13 | PIN_INT6 | Pin interrupt 6 or pattern match engine slice 6 interrupt | PSTAT - pin interrupt status |
| 14 | PIN_INT7 | Pin interrupt 7 or pattern match engine slice 7 interrupt | PSTAT - pin interrupt status |
| 15 | RIT | RIT interrupt | RITINT; masked compare interrupt |
| 16 | SCT0 | State configurable timer interrupt | EVFLAG SCT event |
| 17 | SCT1 | State configurable timer interrupt | EVFLAG SCT event |
| 18 | SCT2 | State configurable timer interrupt | EVFLAG SCT event |
| 19 | SCT3 | State configurable timer interrupt | EVFLAG SCT event |
| 20 | MRT | Multi-rate timer interrupt | Global MRT interrupt.<br>GFLAG0<br>GFLAG1<br>GFLAG2<br>GFLAG3 |
| 21 | UART0 | USART0 interrupt | See Table 350 “USART Interrupt Enable read and set register (INTENSET, address 0x4004 000C(USART0), 0x4004 400C (USART1), 0x400C 000C (USART2)) bit description”. |
| 22 | UART1 | USART1 interrupt | Same as UART0 |
| 23 | UART2 | USART2 interrupt | Same as UART0 |
| 24 | I2C0 | I2C0 interrupt | See Table 379 “Interrupt Enable Set and read register (INTENSET, address 0x4005 0008) bit description”. |

**Table 2. Connection of interrupt sources to the NVIC**

| Interrupt number | Name | Description | Flags |
|---|---|---|---|
| 25 | SPI0 | SPI0 interrupt | See Table 364 “SPI Interrupt Enable read and Set register (INTENSET, addresses 0x4004 800C (SPI0), 0x4004 C00C (SPI1)) bit description”. |
| 26 | SPI1 | SPI1 interrupt | Same as SPI0 |
| 27 | C_CAN0 | C_CAN0 interrupt | INTID. See Table 398. |
| 28 | USB | USB interrupt | USB_Int_Req. See Table 342. |
| 29 | USB_FIQ | USB interrupt | USB_Int_Req_FIQ. See Table 342. |
| 30 | USB_WAKEUP | USB wake-up interrupt | USB need_clock signal |
| 31 | ADC0_SEQA | ADC0 sequence A completion. | See Table 443. |
| 32 | ADC0_SEQB | ADC0 sequence B completion. | See Table 443. |
| 33 | ADC0_THCMP | ADC0 threshold compare. | See Table 443. |
| 34 | ADC0_OVR | ADC0 overrun. | See Table 443. |
| 35 | ADC1_SEQA | ADC1 sequence A completion. | See Table 443. |
| 36 | ADC1_SEQB | ADC1 sequence B completion. | See Table 443. |
| 37 | ADC1_THCMP | ADC1 threshold compare. | See Table 443. |
| 38 | ADC1_OVR | ADC1 overrun. | See Table 443. |
| 39 | DAC | DAC interrupt | Internal DMA timer overflow. |
| 40 | CMP0 | Analog comparator 0 interrupt (ACMP0) | COMPEDGE - rising, falling, or both edges can set the bit. |
| 41 | CMP1 | Analog comparator 1 interrupt (ACMP1) | COMPEDGE - rising, falling, or both edges can set the bit. |
| 42 | CMP2 | Analog comparator 2 interrupt (ACMP2) | COMPEDGE - rising, falling, or both edges can set the bit. |
| 43 | CMP3 | Analog comparator 3 interrupt (ACMP3) | COMPEDGE - rising, falling, or both edges can set the bit. |
| 44 | QEI | QEI interrupt | See Table 321. |
| 45 | RTC_ALARM | RTC alarm interrupt | ALARM1HZ. See Table 272. |
| 46 | RTC_WAKE | RTC wake-up interrupt | WAKEHIRES. See Table 272. |

## 2.4 Register description

The NVIC registers are located on the ARM private peripheral bus.

**Table 3. Register overview: NVIC (base address 0xE000 E000)**

| Name | Access | Address offset | Description | Reset value | Reference |
|---|---|---|---|---|---|
| ISER0 | R/W | 0x100 | Interrupt Set Enable Register 0. This register allows enabling interrupts and reading back the interrupt enables for specific peripheral functions. | 0 | Table 4 |
| ISER1 | R/W | 0x104 | Interrupt Set Enable Register 1. | 0 | Table 5 |
| ICER0 | R/W | 0x180 | Interrupt Clear Enable Register 0. This register allows disabling interrupts and reading back the interrupt enables for specific peripheral functions. | 0 | Table 6 |
| ICER1 | R/W | 0x184 | Interrupt Clear Enable Register 1. | 0 | Table 7 |
| ISPR0 | R/W | 0x200 | Interrupt Set Pending Register 0. This register allows changing the interrupt state to pending and reading back the interrupt pending state for specific peripheral functions. | 0 | Table 8 |
| ISPR1 | R/W | 0x204 | Interrupt Set Pending Register 1. | 0 | Table 9 |
| ICPR0 | R/W | 0x280 | Interrupt Clear Pending Register 0. This register allows changing the interrupt state to not pending and reading back the interrupt pending state for specific peripheral functions. | 0 | Table 10 |
| ICPR1 | R/W | 0x284 | Interrupt Clear Pending Register 1. | 0 | Table 11 |
| IABR0 | R | 0x300 | Interrupt Active Bit Register 0. This register allows reading the current interrupt active state for specific peripheral functions. | 0 | Table 12 |
| IABR1 | R | 0x304 | Interrupt Active Bit Register 1. | 0 | Table 13 |
| IPR0 | R/W | 0x400 | Interrupt Priority Registers 0. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 0 to 3. | 0 | Table 14 |
| IPR1 | R/W | 0x404 | Interrupt Priority Registers 1 This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 4 to 7. | 0 | Table 15 |
| IPR2 | R/W | 0x408 | Interrupt Priority Registers 2. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 8 to 11. | 0 | Table 16 |
| IPR3 | R/W | 0x40C | Interrupt Priority Registers 3. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 12 to 15. | 0 | Table 17 |
| IPR4 | R/W | 0x410 | Interrupt Priority Registers 4. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 16 to 19. | 0 | Table 18 |
| IPR5 | R/W | 0x414 | Interrupt Priority Registers 5. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 20 to 23. | 0 | Table 19 |
| IPR6 | R/W | 0x418 | Interrupt Priority Registers 6. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 24 to 27. | 0 | Table 20 |
| IPR7 | R/W | 0x41C | Interrupt Priority Registers 7. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 28 to 31. | 0 | Table 21 |
| IPR8 | R/W | 0x420 | Interrupt Priority Registers 8. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 32 to 35. | 0 | Table 22 |

**Table 3. Register overview: NVIC (base address 0xE000 E000)** *…continued*

| Name | Access | Address offset | Description | Reset value | Reference |
|---|---|---|---|---|---|
| IPR9 | R/W | 0x424 | Interrupt Priority Registers 9. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 36 to 39. | 0 | Table 23 |
| IPR10 | R/W | 0x428 | Interrupt Priority Registers 10. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 40 to 43. | 0 | Table 24 |
| IPR11 | R/W | 0x42C | Interrupt Priority Registers 11. This register allows assigning a priority to each interrupt. This register contains the 3-bit priority fields for interrupts 44 to 46. | 0 | Table 25 |
| STIR | W | 0xF00 | Software Trigger Interrupt Register. This register allows software to generate an interrupt. | 0 | Table 26 |

### 2.4.1 Interrupt Set Enable Register 0 register

The ISER0 register allows to enable peripheral interrupts or to read the enabled state of those interrupts. Disable interrupts through the ICER0.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 enables the interrupt.

**Read —** 0 indicates that the interrupt is disabled, 1 indicates that the interrupt is enabled.

**Table 4. Interrupt Set Enable Register 0 register (ISER0, address 0xE000 E100) bit description**

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 0 | ISE_WDT | Interrupt enable. | 0 |
| 1 | ISE_BOD | Interrupt enable. | 0 |
| 2 | ISE_FLASH | Interrupt enable. | 0 |
| 3 | ISE_EE | Interrupt enable. | 0 |
| 4 | ISE_DMA | Interrupt enable. | 0 |
| 5 | ISE_GINT0 | Interrupt enable. | 0 |
| 6 | ISE_GINT1 | Interrupt enable. | 0 |
| 7 | ISE_PIN_INT0 | Interrupt enable. | 0 |
| 8 | ISE_PIN_INT1 | Interrupt enable. | 0 |
| 9 | ISE_PIN_INT2 | Interrupt enable. | 0 |
| 10 | ISE_PIN_INT3 | Interrupt enable. | 0 |
| 11 | ISE_PIN_INT4 | Interrupt enable. | 0 |
| 12 | ISE_PIN_INT5 | Interrupt enable. | 0 |
| 13 | ISE_PIN_INT6 | Interrupt enable. | 0 |
| 14 | ISE_PIN_INT7 | Interrupt enable. | 0 |
| 15 | ISE_RIT | Interrupt enable. | 0 |
| 16 | ISE_SCT0 | Interrupt enable. | 0 |
| 17 | ISE_SCT1 | Interrupt enable. | 0 |
| 18 | ISE_SCT2 | Interrupt enable. | 0 |
| 19 | ISE_SCT3 | Interrupt enable. | 0 |

**Table 4. Interrupt Set Enable Register 0 register (ISER0, address 0xE000 E100) bit description** *…continued*

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 20 | ISE_MRT | Interrupt enable. | 0 |
| 21 | ISE_UART0 | Interrupt enable. | 0 |
| 22 | ISE_UART1 | Interrupt enable. | 0 |
| 23 | ISE_UART2 | Interrupt enable. | 0 |
| 24 | ISE_I2C0 | Interrupt enable. | 0 |
| 25 | ISE_SPI0 | Interrupt enable. | 0 |
| 26 | ISE_SPI1 | Interrupt enable. | 0 |
| 27 | ISE_CCAN0 | Interrupt enable. | 0 |
| 28 | ISE_USB | Interrupt enable. | 0 |
| 29 | ISE_USB_FIQ | Interrupt enable. | 0 |
| 30 | ISE_USB_WAKEKUP | Interrupt enable. | 0 |
| 31 | ISE_ADC0_SEQA | Interrupt enable. | 0 |

### 2.4.2 Interrupt Set Enable Register 1 register

The ISER1 register allows to enable peripheral interrupts or to read the enabled state of those interrupts. Disable interrupts through the ICER1.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 enables the interrupt.

**Read —** 0 indicates that the interrupt is disabled, 1 indicates that the interrupt is enabled.

**Table 5. Interrupt Set Enable Register 1 register (ISER1, address 0xE000 E104) bit description**

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 0 | ISE_ADC0_SEQB | Interrupt enable. | 0 |
| 1 | ISE_ADC0_THCMP | Interrupt enable. | 0 |
| 2 | ISE_ADC0_OVR | Interrupt enable. | 0 |
| 3 | ISE_ADC1_SEQA | Interrupt enable. | 0 |
| 4 | ISE_ADC1_SEQB | Interrupt enable. | 0 |
| 5 | ISE_ADC1_THCMP | Interrupt enable. | 0 |
| 6 | ISE_ADC1_OVR | Interrupt enable. | 0 |
| 7 | ISE_DAC | Interrupt enable. | 0 |
| 8 | ISE_ACMP0 | Interrupt enable. | 0 |
| 9 | ISE_ACMP1 | Interrupt enable. | 0 |
| 10 | ISE_ACMP2 | Interrupt enable. | 0 |
| 11 | ISE_ACMP3 | Interrupt enable. | 0 |
| 12 | ISE_QEI | Interrupt enable. | 0 |
| 13 | ISE_RTC_ALARM | Interrupt enable. | 0 |
| 14 | ISE_RTC_WAKE | Interrupt enable. | 0 |
| 31:15 | - | Reserved | 0 |

### 2.4.3 Interrupt clear enable register 0

The ICER0 register allows disabling the peripheral interrupts, or for reading the enabled state of those interrupts. Enable interrupts through the ISER0 register.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 disables the interrupt.

**Read —** 0 indicates that the interrupt is disabled, 1 indicates that the interrupt is enabled.

**Table 6. Interrupt clear enable register 0 (ICER0, address 0xE000 E180)**

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 0 | ICE_WDT | Interrupt disable. | 0 |
| 1 | ICE_BOD | Interrupt disable. | 0 |
| 2 | ICE_FLASH | Interrupt disable. | 0 |
| 3 | ICE_EE | Interrupt disable. | 0 |
| 4 | ICE_DMA | Interrupt disable. | 0 |
| 5 | ICE_GINT0 | Interrupt disable. | 0 |
| 6 | ICE_GINT1 | Interrupt disable. | 0 |
| 7 | ICE_PIN_INT0 | Interrupt disable. | 0 |
| 8 | ICE_PIN_INT1 | Interrupt disable. | 0 |
| 9 | ICE_PIN_INT2 | Interrupt disable. | 0 |
| 10 | ICE_PIN_INT3 | Interrupt disable. | 0 |
| 11 | ICE_PIN_INT4 | Interrupt disable. | 0 |
| 12 | ICE_PIN_INT5 | Interrupt disable. | 0 |
| 13 | ICE_PIN_INT6 | Interrupt disable. | 0 |
| 14 | ICE_PIN_INT7 | Interrupt disable. | 0 |
| 15 | ICE_RIT | Interrupt disable. | 0 |
| 16 | ICE_SCT0 | Interrupt disable. | 0 |
| 17 | ICE_SCT1 | Interrupt disable. | 0 |
| 18 | ICE_SCT2 | Interrupt disable. | 0 |
| 19 | ICE_SCT3 | Interrupt disable. | 0 |
| 20 | ICE_MRT | Interrupt disable. | 0 |
| 21 | ICE_UART0 | Interrupt disable. | 0 |
| 22 | ICE_UART1 | Interrupt disable. | 0 |
| 23 | ICE_UART2 | Interrupt disable. | 0 |
| 24 | ICE_I2C0 | Interrupt disable. | 0 |
| 25 | ICE_SPI0 | Interrupt disable. | 0 |
| 26 | ICE_SPI1 | Interrupt disable. | 0 |
| 27 | ICE_CCAN0 | Interrupt disable. | 0 |
| 28 | ICE_USB | Interrupt disable. | 0 |
| 29 | ICE_USB_FIQ | Interrupt disable. | 0 |
| 30 | ICE_USB_WAKEKUP | Interrupt disable. | 0 |
| 31 | ICE_ADC0_SEQA | Interrupt disable. | 0 |

### 2.4.4 Interrupt clear enable register 1

The ICER1 register allows disabling the peripheral interrupts, or for reading the enabled state of those interrupts. Enable interrupts through the ISER1 register.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 disables the interrupt.

**Read —** 0 indicates that the interrupt is disabled, 1 indicates that the interrupt is enabled.

**Table 7. Interrupt clear enable register1 (ICER1, address 0xE000 E184)**

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 0 | ICE_ADC0_SEQB | Interrupt disable. | 0 |
| 1 | ICE_ADC0_THCMP | Interrupt disable. | 0 |
| 2 | ICE_ADC0_OVR | Interrupt disable. | 0 |
| 3 | ICE_ADC1_SEQA | Interrupt disable. | 0 |
| 4 | ICE_ADC1_SEQB | Interrupt disable. | 0 |
| 5 | ICE_ADC1_THCMP | Interrupt disable. | 0 |
| 6 | ICE_ADC1_OVR | Interrupt disable. | 0 |
| 7 | ICE_DAC | Interrupt disable. | 0 |
| 8 | ICE_ACMP0 | Interrupt disable. | 0 |
| 9 | ICE_ACMP1 | Interrupt disable. | 0 |
| 10 | ICE_ACMP2 | Interrupt disable. | 0 |
| 11 | ICE_ACMP3 | Interrupt disable. | 0 |
| 12 | ICE_QEI | Interrupt disable. | 0 |
| 13 | ICE_RTC_ALARM | Interrupt disable. | 0 |
| 14 | ICE_RTC_WAKE | Interrupt disable. | 0 |
| 31:15 | - | Reserved | 0 |

### 2.4.5 Interrupt Set Pending Register 0 register

The ISPR0 register allows setting the pending state of the peripheral interrupts, or for reading the pending state of those interrupts. Clear the pending state of interrupts through the ICPR0 registers.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 changes the interrupt state to pending.

**Read —** 0 indicates that the interrupt is not pending, 1 indicates that the interrupt is pending.

**Table 8. Interrupt set pending register 0 register (ISPR0, address 0xE000 E200) bit description**

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 0 | ISP_WDT | Interrupt pending set. | 0 |
| 1 | ISP_BOD | Interrupt pending set. | 0 |
| 2 | ISP_FLASH | Interrupt pending set. | 0 |
| 3 | ISP_EE | Interrupt pending set. | 0 |
| 4 | ISP_DMA | Interrupt pending set. | 0 |

**Table 8. Interrupt set pending register 0 register (ISPR0, address 0xE000 E200) bit description** *…continued*

| Bit | Symbol | Description | Reset value |
|---|---|---|---|
| 5 | ISP_GINT0 | Interrupt pending set. | 0 |
| 6 | ISP_GINT1 | Interrupt pending set. | 0 |
| 7 | ISP_PIN_INT0 | Interrupt pending set. | 0 |
| 8 | ISP_PIN_INT1 | Interrupt pending set. | 0 |
| 9 | ISP_PIN_INT2 | Interrupt pending set. | 0 |
| 10 | ISP_PIN_INT3 | Interrupt pending set. | 0 |
| 11 | ISP_PIN_INT4 | Interrupt pending set. | 0 |
| 12 | ISP_PIN_INT5 | Interrupt pending set. | 0 |
| 13 | ISP_PIN_INT6 | Interrupt pending set. | 0 |
| 14 | ISP_PIN_INT7 | Interrupt pending set. | 0 |
| 15 | ISP_RIT | Interrupt pending set. | 0 |
| 16 | ISP_SCT0 | Interrupt pending set. | 0 |
| 17 | ISP_SCT1 | Interrupt pending set. | 0 |
| 18 | ISP_SCT2 | Interrupt pending set. | 0 |
| 19 | ISP_SCT3 | Interrupt pending set. | 0 |
| 20 | ISP_MRT | Interrupt pending set. | 0 |
| 21 | ISP_UART0 | Interrupt pending set. | 0 |
| 22 | ISP_UART1 | Interrupt pending set. | 0 |
| 23 | ISP_UART2 | Interrupt pending set. | 0 |
| 24 | ISP_I2C0 | Interrupt pending set. | 0 |
| 25 | ISP_SPI0 | Interrupt pending set. | 0 |
| 26 | ISP_SPI1 | Interrupt pending set. | 0 |
| 27 | ISP_CCAN0 | Interrupt pending set. | 0 |
| 28 | ISP_USB | Interrupt pending set. | 0 |
| 29 | ISP_USB_FIQ | Interrupt pending set. | 0 |
| 30 | ISP_USB_WAKEKUP | Interrupt pending set. | 0 |
| 31 | ISP_ADC0_SEQA | Interrupt pending set. | 0 |

### 2.4.6 Interrupt Set Pending Register 1 register

The ISPR1 register allows setting the pending state of the peripheral interrupts, or for reading the pending state of those interrupts. Clear the pending state of interrupts through the ICPR1 registers.

The bit description is as follows for all bits in this register:

**Write —** Writing 0 has no effect, writing 1 changes the interrupt state to pending.

**Read —** 0 indicates that the interrupt is not pending, 1 indicates that the interrupt is pending.