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	ARM7®
Core Size	16/32-Bit
Speed	55MHz
Connectivity	CANbus, Ethernet, I²C, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	62
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.65V ~ 1.95V
Data Converters	A/D 8x10b
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/microchip-technology/at91sam7x128-au

12.5.2 EmbeddedICE (Embedded In-circuit Emulator)

The ARM7TDMI EmbeddedICE is supported via the ICE/JTAG port. The internal state of the ARM7TDMI is examined through an ICE/JTAG port.

The ARM7TDMI processor contains hardware extensions for advanced debugging features:

- In halt mode, a store-multiple (STM) can be inserted into the instruction pipeline. This exports the contents of the ARM7TDMI registers. This data can be serially shifted out without affecting the rest of the system.
- In monitor mode, the JTAG interface is used to transfer data between the debugger and a simple monitor program running on the ARM7TDMI processor.

There are three scan chains inside the ARM7TDMI processor that support testing, debugging, and programming of the EmbeddedICE. The scan chains are controlled by the ICE/JTAG port.

EmbeddedICE mode is selected when JTAGSEL is low. It is not possible to switch directly between ICE and JTAG operations. A chip reset must be performed after JTAGSEL is changed.

For further details on the EmbeddedICE, see the ARM7TDMI (Rev4) Technical Reference Manual (DDI0210B).

12.5.3 Debug Unit

The Debug Unit provides a two-pin (DXRD and TXRD) USART that can be used for several debug and trace purposes and offers an ideal means for in-situ programming solutions and debug monitor communication. Moreover, the association with two peripheral data controller channels permits packet handling of these tasks with processor time reduced to a minimum.

The Debug Unit also manages the interrupt handling of the COMMTX and COMMRX signals that come from the ICE and that trace the activity of the Debug Communication Channel. The Debug Unit allows blockage of access to the system through the ICE interface.

A specific register, the Debug Unit Chip ID Register, gives information about the product version and its internal configuration.

The SAM7X512 Debug Unit Chip ID value is 0x275C 0A40 on 32-bit width.

The SAM7X256 Debug Unit Chip ID value is 0x275B 0940 on 32-bit width.

The SAM7X128 Debug Unit Chip ID value is 0x275A 0740 on 32-bit width.

For further details on the Debug Unit, see the Debug Unit section.

12.5.4 IEEE 1149.1 JTAG Boundary Scan

IEEE 1149.1 JTAG Boundary Scan allows pin-level access independent of the device packaging technology.

IEEE 1149.1 JTAG Boundary Scan is enabled when JTAGSEL is high. The SAMPLE, EXTEST and BYPASS functions are implemented. In ICE debug mode, the ARM processor responds with a non-JTAG chip ID that identifies the processor to the ICE system. This is not IEEE 1149.1 JTAG-compliant.

It is not possible to switch directly between JTAG and ICE operations. A chip reset must be performed after JTAGSEL is changed.

A Boundary-scan Descriptor Language (BSDL) file is provided to set up test.

12.5.4.1 JTAG Boundary-scan Register

The Boundary-scan Register (BSR) contains 187 bits that correspond to active pins and associated control signals.

Table 12-2. SAM7X JTAG Boundary Scan Register (Continued)

Bit Number	Pin Name	Pin Type	Associated BSR Cells
117	PA18/SPI0_SPCK	IN/OUT	INPUT
116			OUTPUT
115			CONTROL
114	PB9/EMDIO	IN/OUT	INPUT
113			OUTPUT
112			CONTROL
111	PB8/EMDC	IN/OUT	INPUT
110			OUTPUT
109			CONTROL
108	PB14/ERX3/SPI0_NPCS2	IN/OUT	INPUT
107			OUTPUT
106			CONTROL
105	PB13/ERX2/SPI0_NPCS1	IN/OUT	INPUT
104			OUTPUT
103			CONTROL
102	PB6/ERX1	IN/OUT	INPUT
101			OUTPUT
100			CONTROL
99	PB5/ERX0	IN/OUT	INPUT
98			OUTPUT
97			CONTROL
96	PB15/ERXDV/ECRSDV	IN/OUT	INPUT
95			OUTPUT
94			CONTROL
93	PB17/ERXCK/SPI0_NPCS3	IN/OUT	INPUT
92			OUTPUT
91			CONTROL
90	PB7/ERXER	IN/OUT	INPUT
89			OUTPUT
88			CONTROL
87	PB12/ETXER/TCLK0	IN/OUT	INPUT
86			OUTPUT
85			CONTROL
84	PB0/ETXCK/EREFCK/PCK0	PB0/ETXCK/EREFCK/PCK0	INPUT
83			OUTPUT
82			CONTROL

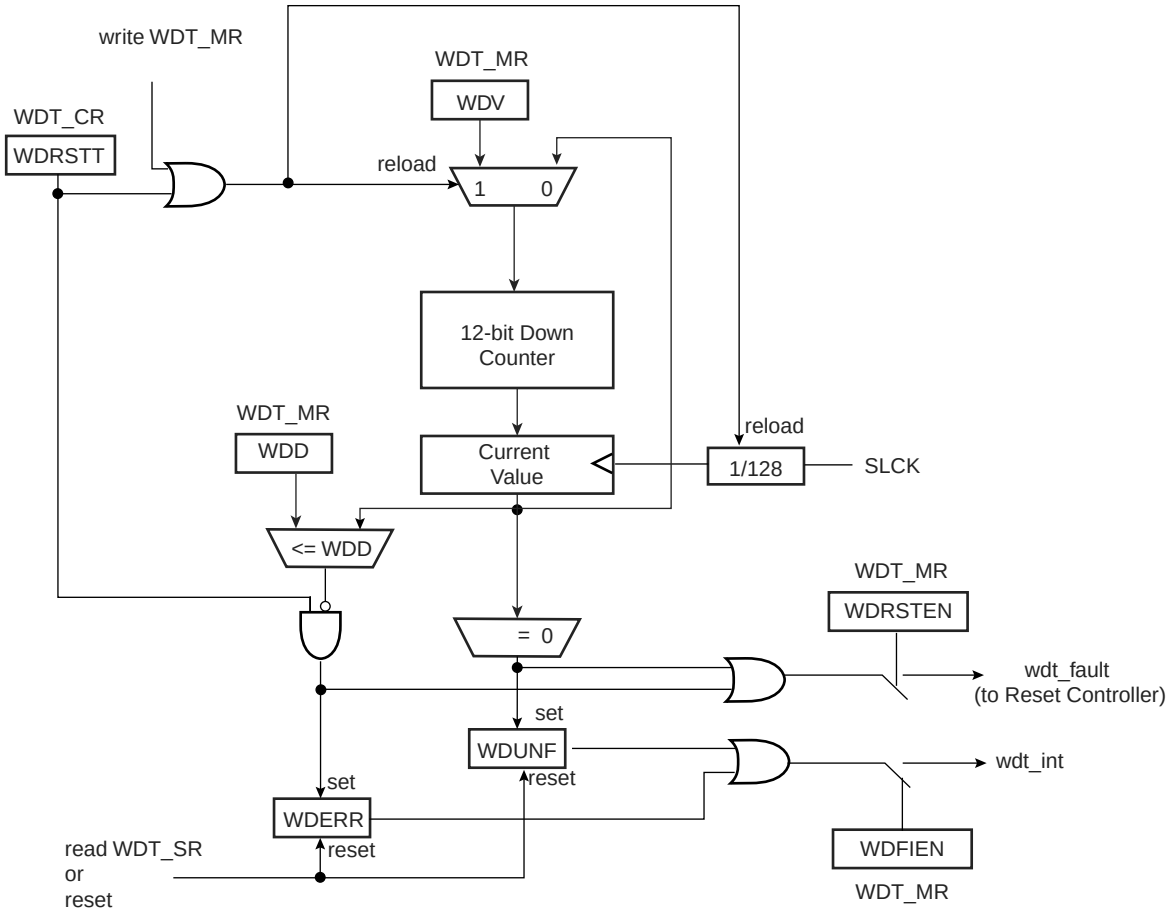
16. Watchdog Timer (WDT)

16.1 Overview

The Watchdog Timer can be used to prevent system lock-up if the software becomes trapped in a deadlock. It features a 12-bit down counter that allows a watchdog period of up to 16 seconds (slow clock at 32.768 kHz). It can generate a general reset or a processor reset only. In addition, it can be stopped while the processor is in debug mode or idle mode.

16.2 Block Diagram

Figure 16-1. Watchdog Timer Block Diagram



18.4.3 MC Abort Address Status Register

Register Name: MC_AASR

Access Type: Read-only

Reset Value: 0x0

Offset: 0x08

31	30	29	28	27	26	25	24
ABTADD							
23	22	21	20	19	18	17	16
ABTADD							
15	14	13	12	11	10	9	8
ABTADD							
7	6	5	4	3	2	1	0
ABTADD							

- **ABTADD: Abort Address**

This field contains the address of the last aborted access.

- **PAGEN: Page Number**

Command	PAGEN Description
Write Page Command	PAGEN defines the page number to be written.
Write Page and Lock Command	PAGEN defines the page number to be written and its associated lock region.
Erase All Command	This field is meaningless
Set/Clear Lock Bit Command	PAGEN defines one page number of the lock region to be locked or unlocked.
Set/Clear General Purpose NVM Bit Command	PAGEN defines the general-purpose bit number.
Set Security Bit Command	This field is meaningless

Note: Depending on the command, all the possible unused bits of PAGEN are meaningless.

- **KEY: Write Protection Key**

This field should be written with the value 0x5A to enable the command defined by the bits of the register. If the field is written with a different value, the write is not performed and no action is started.

25.9.14 PMC Status Register

Register Name: PMC_SR

Access Type: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	PCKRDY2	PCKRDY1	PCKRDY0
7	6	5	4	3	2	1	0
–	–	–	–	MCKRDY	LOCK	–	MOSCS

- **MOSCS: MOSCS Flag Status**

0 = Main oscillator is not stabilized.

1 = Main oscillator is stabilized.

- **LOCK: PLL Lock Status**

0 = PLL is not locked

1 = PLL is locked.

- **MCKRDY: Master Clock Status**

0 = Master Clock is not ready.

1 = Master Clock is ready.

- **PCKRDYx: Programmable Clock Ready Status**

0 = Programmable Clock x is not ready.

1 = Programmable Clock x is ready.

27.6.11 PIO Controller Clear Output Data Register

Name: PIO_CODR

Access Type: Write-only

31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

- **P0-P31: Set Output Data**

0 = No effect.

1 = Clears the data to be driven on the I/O line.

27.6.12 PIO Controller Output Data Status Register

Name: PIO_ODSR

Access Type: Read-only or Read-write

31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

- **P0-P31: Output Data Status**

0 = The data to be driven on the I/O line is 0.

1 = The data to be driven on the I/O line is 1.

27.6.21 PIO Pull Up Disable Register

Name: PIO_PUDR

Access Type: Write-only

31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

- **P0-P31: Pull Up Disable.**

0 = No effect.

1 = Disables the pull up resistor on the I/O line.

27.6.22 PIO Pull Up Enable Register

Name: PIO_PUER

Access Type: Write-only

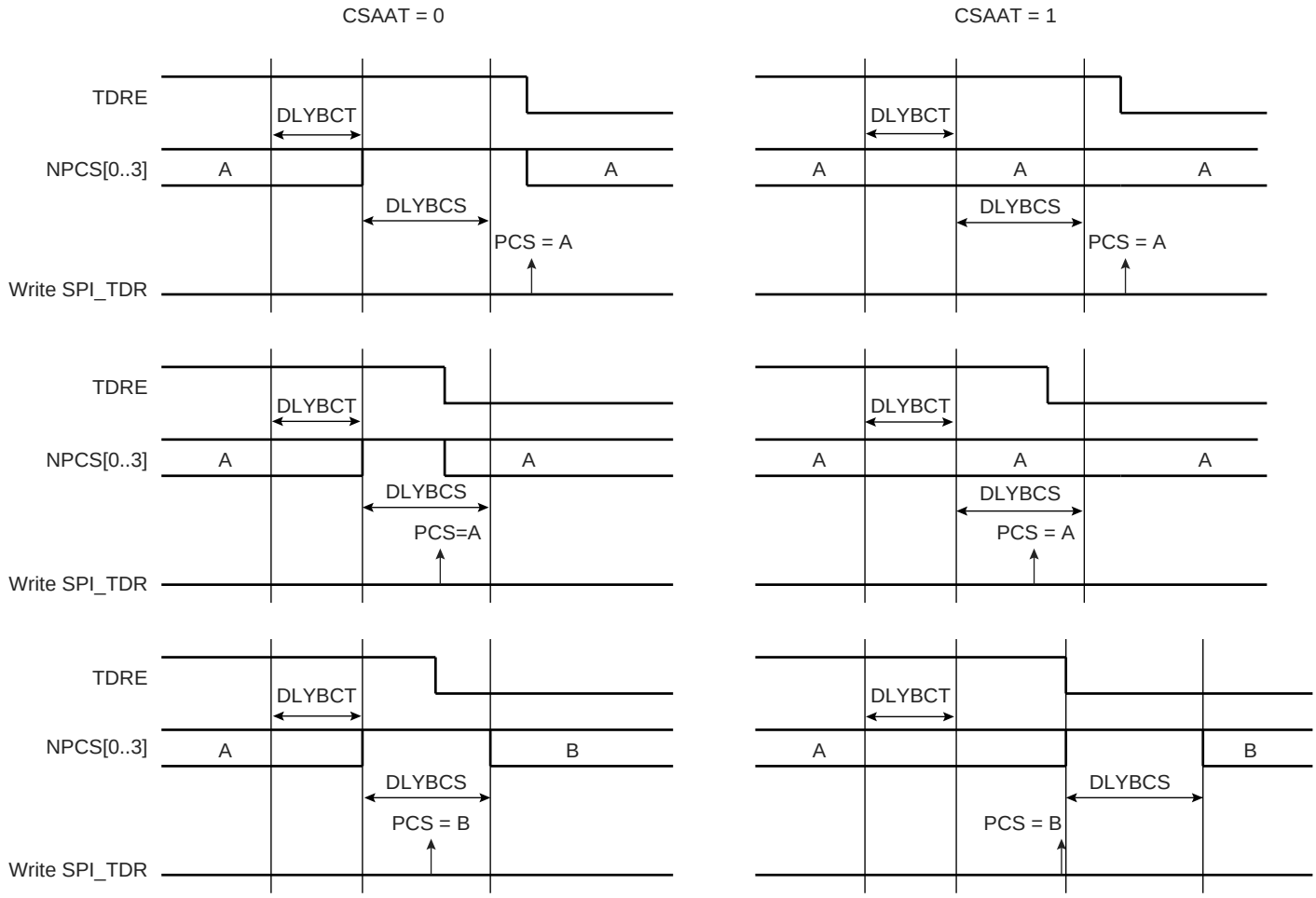
31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

- **P0-P31: Pull Up Enable.**

0 = No effect.

1 = Enables the pull up resistor on the I/O line.

Figure 28-8. Peripheral Deselection



28.6.3.8 Mode Fault Detection

A mode fault is detected when the SPI is programmed in Master Mode and a low level is driven by an external master on the NPCS0/NSS signal. NPCS0, MOSI, MISO and SPCK must be configured in open drain through the PIO controller, so that external pull up resistors are needed to guarantee high level.

When a mode fault is detected, the MODF bit in the SPI_SR is set until the SPI_SR is read and the SPI is automatically disabled until re-enabled by writing the SPIEN bit in the SPI_CR (Control Register) at 1.

By default, the Mode Fault detection circuitry is enabled. The user can disable Mode Fault detection by setting the MODFDIS bit in the SPI Mode Register (SPI_MR).

28.6.4 SPI Slave Mode

When operating in Slave Mode, the SPI processes data bits on the clock provided on the SPI clock pin (SPCK).

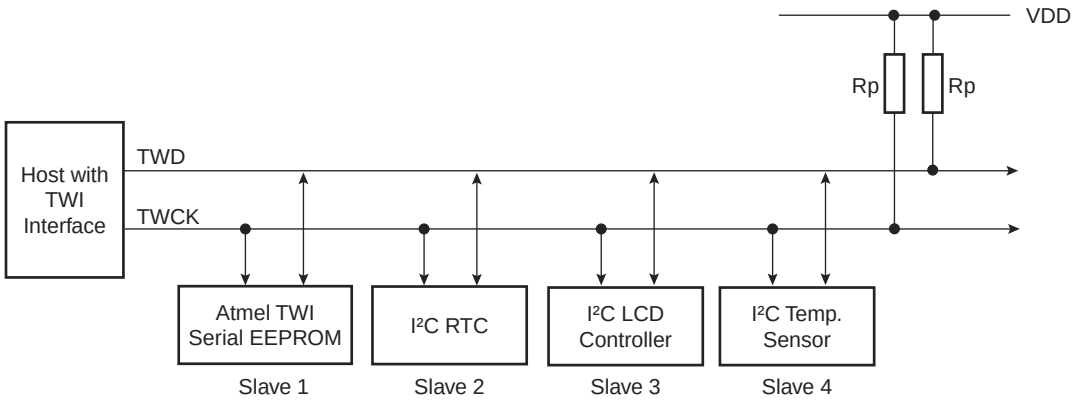
The SPI waits for NSS to go active before receiving the serial clock from an external master. When NSS falls, the clock is validated on the serializer, which processes the number of bits defined by the BITS field of the Chip Select Register 0 (SPI_CSR0). These bits are processed following a phase and a polarity defined respectively by the NCPHA and CPOL bits of the SPI_CSR0. Note that BITS, CPOL and NCPHA of the other Chip Select Registers have no effect when the SPI is programmed in Slave Mode.

The bits are shifted out on the MISO line and sampled on the MOSI line.

When all the bits are processed, the received data is transferred in the Receive Data Register and the RDRF bit rises. If the SPI_RDR (Receive Data Register) has not been read before new data is received, the Overrun Error bit (OVRES) in

29.3 Application Block Diagram

Figure 29-2. Application Block Diagram



Rp: Pull up value as given by the I²C Standard

29.3.1 I/O Lines Description

Table 29-2. I/O Lines Description

Pin Name	Pin Description	Type
TWD	Two-wire Serial Data	Input/Output
TWCK	Two-wire Serial Clock	Input/Output

29.4 Product Dependencies

29.4.1 I/O Lines

Both TWD and TWCK are bidirectional lines, connected to a positive supply voltage via a current source or pull-up resistor (see Figure 29-2 on page 270). When the bus is free, both lines are high. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function.

TWD and TWCK pins may be multiplexed with PIO lines. To enable the TWI, the programmer must perform the following steps:

- Program the PIO controller to:
  - Dedicate TWD and TWCK as peripheral lines.
  - Define TWD and TWCK as open-drain.

29.4.2 Power Management

- Enable the peripheral clock.

The TWI interface may be clocked through the Power Management Controller (PMC), thus the programmer must first configure the PMC to enable the TWI clock.

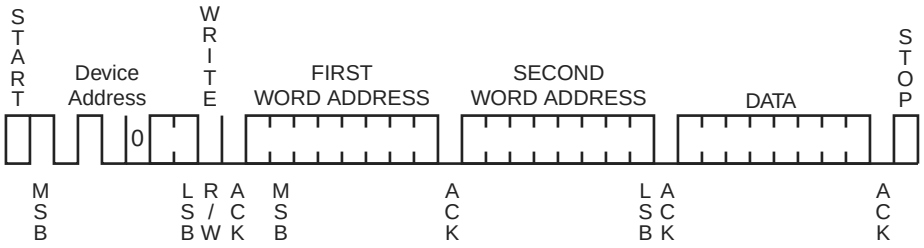
29.4.3 Interrupt

The TWI interface has an interrupt line connected to the Advanced Interrupt Controller (AIC). In order to handle interrupts, the AIC must be programmed before configuring the TWI.

1. Program IADRSZ = 1,
2. Program DADR with 1 1 1 1 0 b1 b2 (b1 is the MSB of the 10-bit address, b2, etc.)
3. Program TWI_IADR with b3 b4 b5 b6 b7 b8 b9 b10 (b10 is the LSB of the 10-bit address)

Figure 29-12 below shows a byte write to an Atmel AT24LC512 EEPROM. This demonstrates the use of internal addresses to access the device.

Figure 29-12. Internal Address Usage



31.8.1 SSC Control Register

Name: SSC_CR

Access Type: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
SWRST	–	–	–	–	–	TXDIS	TXEN
7	6	5	4	3	2	1	0
–	–	–	–	–	–	RXDIS	RXEN

- **RXEN: Receive Enable**

0: No effect.

1: Enables Receive if RXDIS is not set.

- **RXDIS: Receive Disable**

0: No effect.

1: Disables Receive. If a character is currently being received, disables at end of current character reception.

- **TXEN: Transmit Enable**

0: No effect.

1: Enables Transmit if TXDIS is not set.

- **TXDIS: Transmit Disable**

0: No effect.

1: Disables Transmit. If a character is currently being transmitted, disables at end of current character transmission.

- **SWRST: Software Reset**

0: No effect.

1: Performs a software reset. Has priority on any other bit in SSC_CR.

32.5 Functional Description

32.5.1 TC Description

The three channels of the Timer Counter are independent and identical in operation. The registers for channel programming are listed in Table 32-4 on page 389.

32.5.2 16-bit Counter

Each channel is organized around a 16-bit counter. The value of the counter is incremented at each positive edge of the selected clock. When the counter has reached the value 0xFFFF and passes to 0x0000, an overflow occurs and the COVFS bit in TC_SR (Status Register) is set.

The current value of the counter is accessible in real time by reading the Counter Value Register, TC_CV. The counter can be reset by a trigger. In this case, the counter value passes to 0x0000 on the next valid edge of the selected clock.

32.5.3 Clock Selection

At block level, input clock signals of each channel can either be connected to the external inputs TCLK0, TCLK1 or TCLK2, or be connected to the internal I/O signals TIOA0, TIOA1 or TIOA2 for chaining by programming the TC_BMR (Block Mode). See Figure 32-2 on page 378.

Each channel can independently select an internal or external clock source for its counter:

- Internal clock signals: TIMER_CLOCK1, TIMER_CLOCK2, TIMER_CLOCK3, TIMER_CLOCK4, TIMER_CLOCK5
- External clock signals: XC0, XC1 or XC2

This selection is made by the TCCLKS bits in the TC Channel Mode Register.

The selected clock can be inverted with the CLKI bit in TC_CMR. This allows counting on the opposite edges of the clock.

The burst function allows the clock to be validated when an external signal is high. The BURST parameter in the Mode Register defines this signal (none, XC0, XC1, XC2). See Figure 32-3 on page 378

Note: In all cases, if an external clock is used, the duration of each of its levels must be longer than the master clock period. The external clock frequency must be at least 2.5 times lower than the master clock

32.6.6 TC Counter Value Register

Register Name: TC_CVx [x=0..2]

Access Type: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
CV							
7	6	5	4	3	2	1	0
CV							

- **CV: Counter Value**

CV contains the counter value in real time.

32.6.7 TC Register A

Register Name: TC_RAx [x=0..2]

Access Type: Read-only if WAVE = 0, Read-write if WAVE = 1

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
RA							
7	6	5	4	3	2	1	0
RA							

- **RA: Register A**

RA contains the Register A value in real time.

33.6.12 PWM Channel Counter Register

Register Name: PWM_CCNT[0..X-1]

Access Type: Read-only

31	30	29	28	27	26	25	24
CNT							

23	22	21	20	19	18	17	16
CNT							

15	14	13	12	11	10	9	8
CNT							

7	6	5	4	3	2	1	0
CNT							

- **CNT: Channel Counter Register**

Internal counter value. This register is reset when:

- the channel is enabled (writing CHIDx in the PWM_ENA register).
- the counter reaches CPRD value defined in the PWM_CPRDx register if the waveform is left aligned.

- **TSTP: Timestamp Interrupt Mask**

0 = TSTP interrupt is disabled.

1 = TSTP interrupt is enabled.

- **CERR: CRC Error Interrupt Mask**

0 = CRC Error interrupt is disabled.

1 = CRC Error interrupt is enabled.

- **SERR: Stuffing Error Interrupt Mask**

0 = Bit Stuffing Error interrupt is disabled.

1 = Bit Stuffing Error interrupt is enabled.

- **AERR: Acknowledgment Error Interrupt Mask**

0 = Acknowledgment Error interrupt is disabled.

1 = Acknowledgment Error interrupt is enabled.

- **FERR: Form Error Interrupt Mask**

0 = Form Error interrupt is disabled.

1 = Form Error interrupt is enabled.

- **BERR: Bit Error Interrupt Mask**

0 = Bit Error interrupt is disabled.

1 = Bit Error interrupt is enabled.

37.3.3 Pause Frame Support

The start of an 802.3 pause frame is as follows:

Table 37-3. Start of an 802.3 Pause Frame

Destination Address	Source Address	Type (Mac Control Frame)	Pause Opcode	Pause Time
0x0180C2000001	6 bytes	0x8808	0x0001	2 bytes

The network configuration register contains a receive pause enable bit (13). If a valid pause frame is received, the pause time register is updated with the frame's pause time, regardless of its current contents and regardless of the state of the configuration register bit 13. An interrupt (12) is triggered when a pause frame is received, assuming it is enabled in the interrupt mask register. If bit 13 is set in the network configuration register and the value of the pause time register is non-zero, no new frame is transmitted until the pause time register has decremented to zero.

The loading of a new pause time, and hence the pausing of transmission, only occurs when the EMAC is configured for full-duplex operation. If the EMAC is configured for half-duplex, there is no transmission pause, but the pause frame received interrupt is still triggered.

A valid pause frame is defined as having a destination address that matches either the address stored in specific address register 1 or matches 0x0180C2000001 and has the MAC control frame type ID of 0x8808 and the pause opcode of 0x0001. Pause frames that have FCS or other errors are treated as invalid and are discarded. Valid pause frames received increment the Pause Frame Received statistic register.

The pause time register decrements every 512 bit times (i.e., 128 `rx_clks` in nibble mode) once transmission has stopped. For test purposes, the register decrements every `rx_clk` cycle once transmission has stopped if bit 12 (retry test) is set in the network configuration register. If the pause enable bit (13) is not set in the network configuration register, then the decrementing occurs regardless of whether transmission has stopped or not.

An interrupt (13) is asserted whenever the pause time register decrements to zero (assuming it is enabled in the interrupt mask register).

37.3.4 Receive Block

The receive block checks for valid preamble, FCS, alignment and length, presents received frames to the DMA block and stores the frames destination address for use by the address checking block. If, during frame reception, the frame is found to be too long or rx_er is asserted, a bad frame indication is sent to the DMA block. The DMA block then ceases sending data to memory. At the end of frame reception, the receive block indicates to the DMA block whether the frame is good or bad. The DMA block recovers the current receive buffer if the frame was bad. The receive block signals the register block to increment the alignment error, the CRC (FCS) error, the short frame, long frame, jabber error, the receive symbol error statistics and the length field mismatch statistics.

The enable bit for jumbo frames in the network configuration register allows the EMAC to receive jumbo frames of up to 10240 bytes in size. This operation does not form part of the IEEE802.3 specification and is disabled by default. When jumbo frames are enabled, frames received with a frame size greater than 10240 bytes are discarded.

37.3.5 Address Checking Block

The address checking (or filter) block indicates to the DMA block which receive frames should be copied to memory. Whether a frame is copied depends on what is enabled in the network configuration register, the state of the external match pin, the contents of the specific address and hash registers and the frame's destination address. In this implementation of the EMAC, the frame's source address is not checked. Provided that bit 18 of the Network Configuration register is not set, a frame is not copied to memory if the EMAC is transmitting in half duplex mode at the time a destination address is received. If bit 18 of the Network Configuration register is set, frames can be received while transmitting in half-duplex mode.

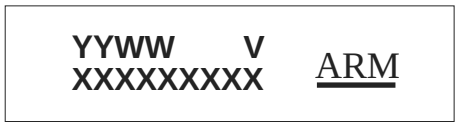
Ethernet frames are transmitted a byte at a time, least significant bit first. The first six bytes (48 bits) of an Ethernet frame make up the destination address. The first bit of the destination address, the LSB of the first byte of the frame, is the

41. SAM7X512/256/128 Errata

41.1 Marking

All devices are marked with the Atmel logo and the ordering code.

Additional marking has the following format:



where

- “YY”: manufactory year
- “WW”: manufactory week
- “V”: revision
- “XXXXXXXXX”: lot number

42. Revision History

The most recent version appears first in the tables that follow.

The acronymn “rfo” indicates change requests made by technical experts during document approval.

Version 6120K	Comments	Change Request Ref.
	Section 29. ”Two-wire Interface (TWI)” Added bit OVRE to Section 29.6.5 ”TWI Status Register”, Section 29.6.6 ”TWI Interrupt Enable Register”, Section 29.6.7 ”TWI Interrupt Disable Register” and Section 29.6.8 ”TWI Interrupt Mask Register”.	7199
	Section 16. ”Watchdog Timer (WDT)” Section 16.3 ”Functional Description”, added new 5th paragraph with description of watchdog restart behavior. Section 16.4.2 ”Watchdog Timer Mode Register”, added note regarding watchdog restart and WDD/WDV values.	8128 8128
	Section 41. ”SAM7X512/256/128 Errata” Table 41-1, “Errata Summary Table”: added AT91SAM7X512 rev. B column. Section 41.4 ”AT91SAM7X512 Errata - Rev. A Parts”: added chip ID. Added new section Section 41.6 ”AT91SAM7X512 Errata - Rev. B Parts”. Added new errata Section 41.7.1.1 ”BSDL: BSDL File for Rev. B Devices Not Compatible with Rev. C Devices”	 rfo rfo 9404 9849
	Section 40. ”AT91SAM7X Ordering Information” Added three ordering codes for AT91SAM7X512 MRL B.	 rfo

Version 6120J	Comments	Change Request Ref.
	Atmel internal document	

Version 6120G (Continued)	Comments	Change Request Ref.
	SPI,	
	Section 28.6.4 “SPI Slave Mode” on page 254, corrected information on OVRES (SPI_SR) and data read in SPI_RDR.	3943
	SSC,	
	Section 31.6.5.1 ”Frame Sync Data”, defined max Frame Sync Data length.	2293
	Section 31.6.6.1 ”Compare Functions”, updated with max FSLEN length.	
	TC,	
	Figure 32-2,”Clock Chaining Selection”, added to Section 32.5 ”Functional Description”.	3342
	Section 32.6 ”Timer Counter (TC) User Interface” Register mapping tables consolidated in Table 32-4 on page 389 and register offsets indexed.	4583
	Section 32.6.3 on page 392 to Section 32.6.13 on page 406, register names updated with indexed offset.	
	Section 32.6.4 ”TC Channel Mode Register: Capture Mode” bit field 15 and WAVE bit field description updated.	
	TWI,	
	“Two-wire Interface (TWI)” , section has been updated.	4247
	Important changes to this datasheet include a clarification of Atmel TWI compatibility with I^2C Standard.	
	UDP,	
	Table 34-2, “USB Communication Flow”, Supported Endpoint column updated.	3476
	In the USB_CSR register, the control endpoints are not effected by the bit field, “EPEDS: Endpoint Enable Disable” on page 475	4063
	Updated: write 1 =.... in “RX_DATA_BK0: Receive Data Bank 0” bit field of USB_CSR register. Updated: write 0 = ...in “TXPKTRDY: Transmit Packet Ready” bit field of USB_CSR register.	4099
	Section 34.6.10 “UDP Endpoint Control and Status Register” on page 457, update to code and added instructions regarding USB clock and system clock cycle, and updated “note” appearing under the code.	4462 4487
	“wait 3 USB clock cycles and 3 system clock cycles before accessing DPR from RX_DATAx and TXPKTRDY bit fields, ditto for **RX_DATAx** and **TXPKTRDY** bit field descriptions.”	4508
	Section 34.2 ”Block Diagram”, in the text below the block diagram, MCK specified as clock used by Master Clock domain, UDPCK specified as 48 MHz clock used by 12 MHz domain, in peripheral clock requirements.	
	Section 34.6 ”USB Device Port (UDP) User Interface”, The register mapping table has been updated	4802
	Section 34.6.6 ”UDP Interrupt Mask Register” Bit 12 of has been defined as BIT12 and cannot be masked.	
	USART,	
	“CLKO: Clock Output Select” on page 323, bit field in US_MR register, typo fixed in bit field description.	3306
	“USCLKS: Clock Selection” on page 321, bit field in US_MR register, DIV= 8 in Selected Clock column.	3763
	Section 30.5.1 ”I/O Lines”, 2nd and 3rd paragraphsupdated.	3851/4285
	“TXEMPTY: Transmitter Empty” on page 328, no characters when at 1 updated.	3895
	Section 30.6.2 ”Receiver and Transmitter Control”, In the fourth paragraph, Software reset effects (RSTRX and RSTTX in US_CR register) updated by replacing 2nd sentence.	4367
	Section 30.6.5 ”IrDA Mode”, updated with instruction to receive IrDA signals.	4912
	Section 30.2 ”Block Diagram”, signal directions from pads to PIO updated in the block diagram.	4905