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	Active
Core Processor	ARM® Cortex®-M4
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	EBI/EMI, I²C, IrDA, Memory Card, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	79
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	160K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 16x12b; D/A 2x12b
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/atsam4sa16ca-aur

6.2 System I/O Lines

System I/O lines are pins used by oscillators, test mode, reset and JTAG. Table 6-1 provides the SAM4S system I/O lines shared with PIO lines.

These pins are software configurable as general-purpose I/O or system pins. At startup, the default function of these pins is always used.

Table 6-1. System I/O Configuration Pin List

SYSTEM_IO Bit Number	Default Function After Reset	Other Function	Constraints For Normal Start	Configuration
12	ERASE	PB12	Low Level at startup(1)	In Matrix User Interface Registers (Refer to the System I/O Configuration Register in Section 25. “Bus Matrix (MATRIX)”.)
10	DDM	PB10	–	
11	DDP	PB11	–	
7	TCK/SWCLK	PB7	–	
6	TMS/SWDIO	PB6	–	
5	TDO/TRACESWO	PB5	–	
4	TDI	PB4	–	
–	PA7	XIN32	–	(2)
–	PA8	XOUT32	–	
–	PB9	XIN	–	(3)
–	PB8	XOUT	–	

Notes:
1. If PB12 is used as PIO input in user applications, a low level must be ensured at startup to prevent Flash erase before the user application sets PB12 into PIO mode,
2. Refer to “Slow Clock Generator” in Section 18. “Supply Controller (SUPC)”.
3. Refer to the 3 to 20 MHZ crystal oscillator information in Section 29. “Power Management Controller (PMC)”.

6.2.1 Serial Wire JTAG Debug Port (SWJ-DP) Pins

The SWJ-DP pins are TCK/SWCLK, TMS/SWDIO, TDO/SWO, TDI and commonly provided on a standard 20-pin JTAG connector defined by ARM. For more details about voltage reference and reset state, refer to Table 3-1 on page 13.

At startup, SWJ-DP pins are configured in SWJ-DP mode to allow connection with debugging probe. Please refer to Section 13. “Debug and Test Features”.

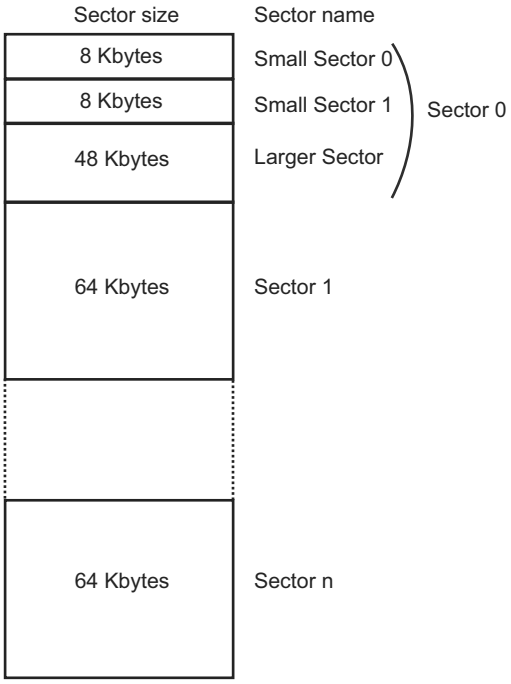
SWJ-DP pins can be used as standard I/Os to provide users more general input/output pins when the debug port is not needed in the end application. Mode selection between SWJ-DP mode (System IO mode) and general IO mode is performed through the AHB Matrix Special Function Registers (MATRIX_SFR). Configuration of the pad for pull-up, triggers, debouncing and glitch filters is possible regardless of the mode.

The JTAG pin and PA7 pin are used to select the JTAG Boundary Scan when asserted JTAGSEL at a high level and PA7 at low level. It integrates a permanent pull-down resistor of about 15 kΩ to GND, so that it can be left unconnected for normal operations.

By default, the JTAG Debug Port is active. If the debugger host wants to switch to the Serial Wire Debug Port, it must provide a dedicated JTAG sequence on TMS/SWDIO and TCK/SWCLK which disables the JTAG-DP and enables the SW-DP. When the Serial Wire Debug Port is active, TDO/TRACESWO can be used for trace.

The asynchronous TRACE output (TRACESWO) is multiplexed with TDO. So the asynchronous trace can only be used with SW-DP, not JTAG-DP. For more information about SW-DP and JTAG-DP switching, please refer to Section 13. “Debug and Test Features”.

Figure 8-1. Global Flash Organization



Each sector is organized in pages of 512 bytes.

For sector 0:

- The smaller sector 0 has 16 pages of 512 bytes
- The smaller sector 1 has 16 pages of 512 bytes
- The larger sector has 96 pages of 512 bytes

From Sector 1 to n:

The rest of the array is composed of 64-Kbyte sectors of 128 pages, each page of 512 bytes. Refer to Figure 8-2, "Flash Sector Organization".

12.6.5.12 SHSUB16 and SHSUB8

Signed Halving Subtract 16 and Signed Halving Subtract 8

Syntax

```
op{cond}{Rd,} Rn, Rm
```

where:

op is any of:

SHSUB16 Signed Halving Subtract 16.

SHSUB8 Signed Halving Subtract 8.

cond is an optional condition code, see “Conditional Execution” .

Rd is the destination register.

Rn is the first operand register.

Rm is the second operand register.

Operation

Use these instructions to add 16-bit and 8-bit data and then to halve the result before writing the result to the destination register:

The SHSUB16 instruction:

1. Subtracts each halfword of the second operand from the corresponding halfwords of the first operand.
2. Shuffles the result by one bit to the right, halving the data.
3. Writes the halved halfword results in the destination register.

The SHSUBB8 instruction:

1. Subtracts each byte of the second operand from the corresponding byte of the first operand,
2. Shuffles the result by one bit to the right, halving the data,
3. Writes the corresponding signed byte results in the destination register.

Restrictions

Do not use SP and do not use PC.

Condition Flags

These instructions do not change the flags.

Examples

```
SHSUB16 R1, R0     ; Subtracts halfwords in R0 from corresponding halfword
                   ; of R1 and writes to corresponding halfword of R1
SHSUB8  R4, R0, R5 ; Subtracts bytes of R0 from corresponding byte in R5,
                   ; and writes to corresponding byte in R4.
```

12.6.5.17 UASX and USAX

Add and Subtract with Exchange and Subtract and Add with Exchange.

Syntax

```
op{cond} {Rd}, Rn, Rm
```

where:

op is one of:

UASX Add and Subtract with Exchange.

USAX Subtract and Add with Exchange.

cond is an optional condition code, see “Conditional Execution” .

Rd is the destination register.

Rn, Rm are registers holding the first and second operands.

Operation

The UASX instruction:

1. Subtracts the top halfword of the second operand from the bottom halfword of the first operand.
2. Writes the unsigned result from the subtraction to the bottom halfword of the destination register.
3. Adds the top halfword of the first operand with the bottom halfword of the second operand.
4. Writes the unsigned result of the addition to the top halfword of the destination register.

The USAX instruction:

1. Adds the bottom halfword of the first operand with the top halfword of the second operand.
2. Writes the unsigned result of the addition to the bottom halfword of the destination register.
3. Subtracts the bottom halfword of the second operand from the top halfword of the first operand.
4. Writes the unsigned result from the subtraction to the top halfword of the destination register.

Restrictions

Do not use SP and do not use PC.

Condition Flags

These instructions do not affect the condition code flags.

Examples

```
UASX  R0, R4, R5 ; Adds top halfword of R4 to bottom halfword of R5 and
                 ; writes to top halfword of R0
                 ; Subtracts bottom halfword of R5 from top halfword of R0
                 ; and writes to bottom halfword of R0
USAX  R7, R3, R2 ; Subtracts top halfword of R2 from bottom halfword of R3
                 ; and writes to bottom halfword of R7
                 ; Adds top halfword of R3 to bottom halfword of R2 and
                 ; writes to top halfword of R7.
```

12.6.5.18 UHADD16 and UHADD8

Unsigned Halving Add 16 and Unsigned Halving Add 8

Syntax

```
op{cond}{Rd,} Rn, Rm
```

where:

op is any of:

UHADD16 Unsigned Halving Add 16.

UHADD8 Unsigned Halving Add 8.

23.7 Cyclic Redundancy Check Calculation Unit (CRCCU) User Interface

Table 23-3. Register Mapping

Offset	Register	Name	Access	Reset
0x000	CRCCU Descriptor Base Register	CRCCU_DSCR	Read/Write	0x00000000
0x004	Reserved	–	–	–
0x008	CRCCU DMA Enable Register	CRCCU_DMA_EN	Write-only	–
0x00C	CRCCU DMA Disable Register	CRCCU_DMA_DIS	Write-only	–
0x010	CRCCU DMA Status Register	CRCCU_DMA_SR	Read-only	0x00000000
0x014	CRCCU DMA Interrupt Enable Register	CRCCU_DMA_IER	Write-only	–
0x018	CRCCU DMA Interrupt Disable Register	CRCCU_DMA_IDR	Write-only	–
0x001C	CRCCU DMA Interrupt Mask Register	CRCCU_DMA_IMR	Read-only	0x00000000
0x020	CRCCU DMA Interrupt Status Register	CRCCU_DMA_ISR	Read-only	0x00000000
0x024–0x030	Reserved	–	–	–
0x034	CRCCU Control Register	CRCCU_CR	Write-only	–
0x038	CRCCU Mode Register	CRCCU_MR	Read/Write	0x00000000
0x03C	CRCCU Status Register	CRCCU_SR	Read-only	0xFFFFFFFF
0x040	CRCCU Interrupt Enable Register	CRCCU_IER	Write-only	–
0x044	CRCCU Interrupt Disable Register	CRCCU_IDR	Write-only	–
0x048	CRCCU Interrupt Mask Register	CRCCU_IMR	Read-only	0x00000000
0x004C	CRCCU Interrupt Status Register	CRCCU_ISR	Read-only	0x00000000
0x050–0x0FC	Reserved	–	–	–

24.5.4 In Application Programming (IAP) Feature

The IAP feature is a function located in ROM that can be called by any software application.

When called, this function sends the desired FLASH command to the EEFC and waits for the Flash to be ready (looping while the FRDY bit is not set in the EEFC_FSR).

Since this function is executed from ROM, this allows Flash programming (such as sector write) to be done by code running in Flash.

The IAP function entry point is retrieved by reading the NMI vector in ROM (0x00800008).

This function takes two arguments in parameter: the EFC number and the command to be sent to the EEFC.

This function returns the value of the EEFC_FSR.

IAP software code example:

```
(unsigned int) (*IAP_Function)(unsigned long);
void main (void){

                  unsigned long FlashSectorNum = 200; //
                  unsigned long flash_cmd = 0;
                  unsigned long flash_status = 0;
                  unsigned long EFCIndex = 0; // 0:EEFC0, 1: EEFC1

/* Initialize the function pointer (retrieve function address from NMI vector)
*/

                  IAP_Function = ((unsigned long) (*)(unsigned long))
0x00800008;

/* Send your data to the sector here */

/* build the command to send to EEFC */

                  flash_cmd =  (0x5A << 24) | (FlashSectorNum << 8) |
AT91C_MC_FCMD_EWP;

/* Call the IAP function with appropriate command */

                  flash_status = IAP_Function (EFCIndex, flash_cmd);

}
```

26.13.3 Ready Mode

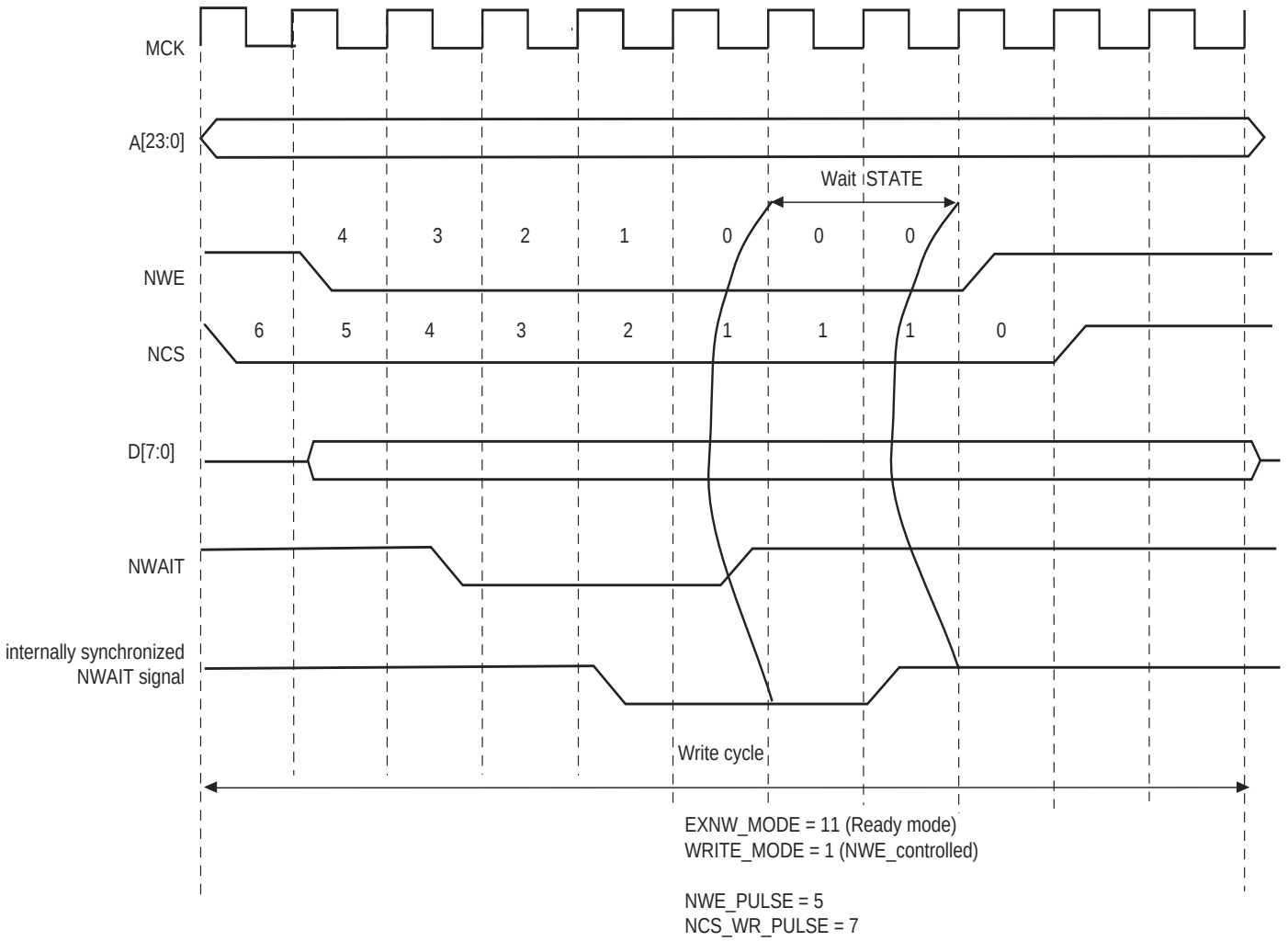
In Ready mode (EXNW_MODE = 11), the SMC behaves differently. Normally, the SMC begins the access by down counting the setup and pulse counters of the read/write controlling signal. In the last cycle of the pulse phase, the resynchronized NWAIT signal is examined.

If asserted, the SMC suspends the access as shown in Figure 26-25 and Figure 26-26. After deassertion, the access is completed: the hold step of the access is performed.

This mode must be selected when the external device uses deassertion of the NWAIT signal to indicate its ability to complete the read or write operation.

If the NWAIT signal is deasserted before the end of the pulse, or asserted after the end of the pulse of the controlling read/write signal, it has no impact on the access length as shown in Figure 26-26.

Figure 26-25. NWAIT Assertion in Write Access: Ready Mode (EXNW_MODE = 11)



31.6.39 PIO Edge Select Register

Name: PIO_ESR

Address: 0x400E0EC0 (PIOA), 0x400E10C0 (PIOB), 0x400E12C0 (PIOC)

Access: 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: Edge Interrupt Selection**

0: No effect.

1: The interrupt source is an edge-detection event.

31.6.44 PIO Fall/Rise - Low/High Status Register

Name: PIO_FRLHSR

Address: 0x400E0ED8 (PIOA), 0x400E10D8 (PIOB), 0x400E12D8 (PIOC)

Access: Read-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: Edge/Level Interrupt Source Selection**

0: The interrupt source is a falling edge detection (if PIO_ELSR = 0) or low-level detection event (if PIO_ELSR = 1).

1: The interrupt source is a rising edge detection (if PIO_ELSR = 0) or high-level detection event (if PIO_ELSR = 1).

- **RXBUFF: Receive Buffer Full**

0: SSC_RCR or SSC_RNCR have a value other than 0.

1: Both SSC_RCR and SSC_RNCR have a value of 0.

- **CP0: Compare 0**

0: A compare 0 has not occurred since the last read of the Status Register.

1: A compare 0 has occurred since the last read of the Status Register.

- **CP1: Compare 1**

0: A compare 1 has not occurred since the last read of the Status Register.

1: A compare 1 has occurred since the last read of the Status Register.

- **TXSYN: Transmit Sync**

0: A Tx Sync has not occurred since the last read of the Status Register.

1: A Tx Sync has occurred since the last read of the Status Register.

- **RXSYN: Receive Sync**

0: An Rx Sync has not occurred since the last read of the Status Register.

1: An Rx Sync has occurred since the last read of the Status Register.

- **TXEN: Transmit Enable**

0: Transmit is disabled.

1: Transmit is enabled.

- **RXEN: Receive Enable**

0: Receive is disabled.

1: Receive is enabled.

33.6.3 Interrupt

The SPI interface has an interrupt line connected to the interrupt controller. Handling the SPI interrupt requires programming the interrupt controller before configuring the SPI.

Table 33-3. Peripheral IDs

Instance	ID
SPI	21

33.6.4 Peripheral DMA Controller (PDC)

The SPI interface can be used in conjunction with the PDC in order to reduce processor overhead. For a full description of the PDC, refer to the corresponding section in the full datasheet.

The command ALL_SEND_CID and the fields and values for the HSMCI_CMDR are described in Table 38-6 and Table 38-7.

Table 38-6. ALL_SEND_CID Command Description

CMD Index	Type	Argument	Response	Abbreviation	Command Description
CMD2	bcr[1]	[31:0] stuff bits	R2	ALL_SEND_CID	Asks all cards to send their CID numbers on the CMD line

Note: 1. bcr means broadcast command with response.

Table 38-7. Fields and Values for HSMCI_CMDR

Field	Value
CMDNB (command number)	2 (CMD2)
RSPTYP (response type)	2 (R2: 136 bits response)
SPCMD (special command)	0 (not a special command)
OPCMD (open drain command)	1
MAXLAT (max latency for command to response)	0 (NID cycles ==> 5 cycles)
TRCMD (transfer command)	0 (No transfer)
TRDIR (transfer direction)	X (available only in transfer command)
TRTYP (transfer type)	X (available only in transfer command)
IOSPCMD (SDIO special command)	0 (not a special command)

The HSMCI_ARGR contains the argument field of the command.

To send a command, the user must perform the following steps:

- Fill the argument register (HSMCI_ARGR) with the command argument.
- Set the command register (HSMCI_CMDR) (see Table 38-7).

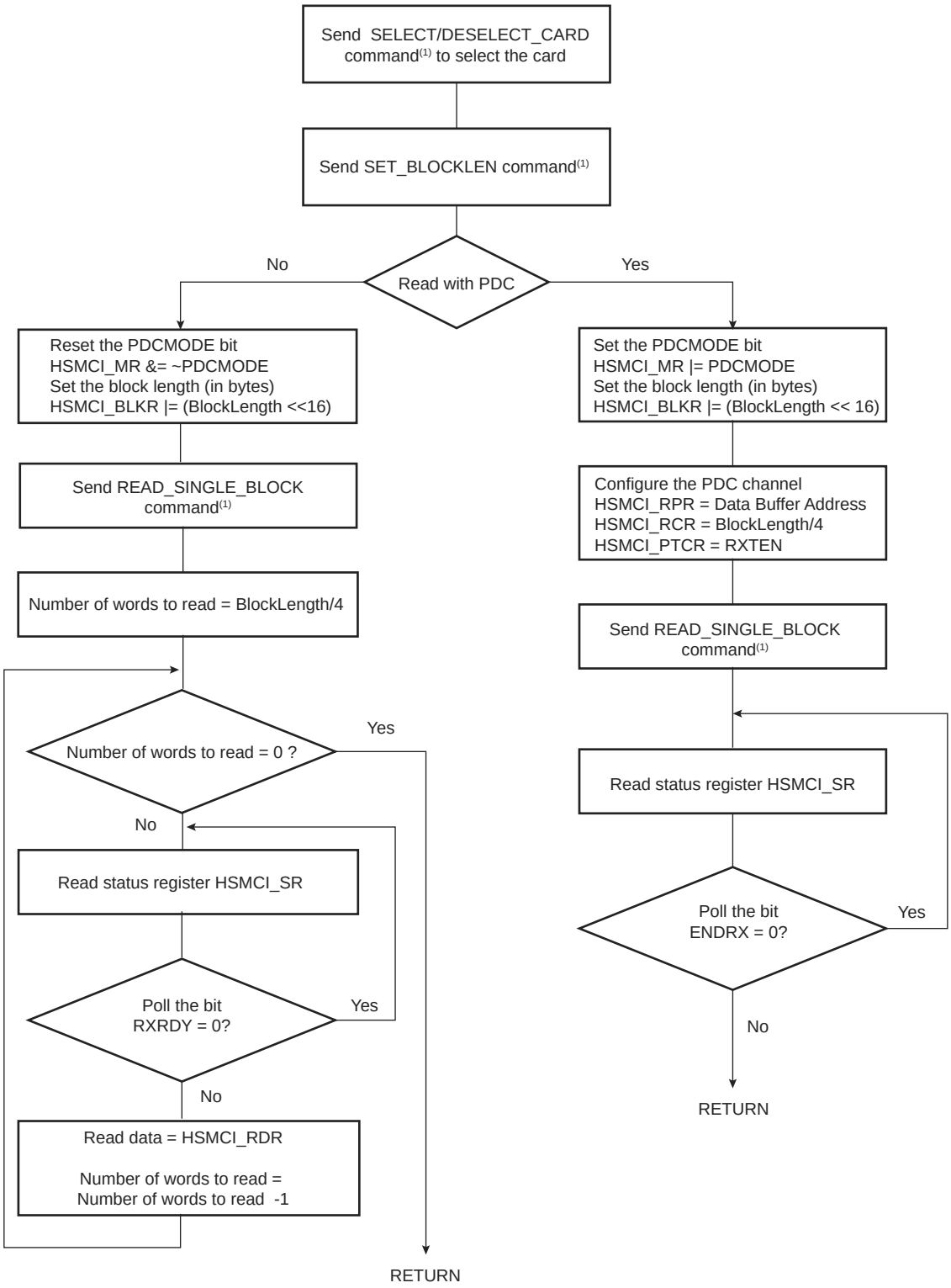
The command is sent immediately after writing the command register.

While the card maintains a busy indication (at the end of a STOP_TRANSMISSION command CMD12, for example), a new command shall not be sent. The NOTBUSY flag in the Status Register (HSMCI_SR) is asserted when the card releases the busy indication.

If the command requires a response, it can be read in the HSMCI Response Register (HSMCI_RSPR). The response size can be from 48 bits up to 136 bits depending on the command. The HSMCI embeds an error detection to prevent any corrupted data during the transfer.

The following flowchart shows how to send a command to the card and read the response if needed. In this example, the status register bits are polled but setting the appropriate bits in the HSMCI Interrupt Enable Register (HSMCI_IER) allows using an interrupt method.

Figure 38-8. Read Functional Flow Diagram



Note: 1. It is assumed that this command has been correctly sent (see Figure 38-7).

39.7.16 PWM Interrupt Status Register 2

Name: PWM_ISR2

Address: 0x40020040

Access: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
CMPU7	CMPU6	CMPU5	CMPU4	CMPU3	CMPU2	CMPU1	CMPU0

15	14	13	12	11	10	9	8
CMPM7	CMPM6	CMPM5	CMPM4	CMPM3	CMPM2	CMPM1	CMPM0

7	6	5	4	3	2	1	0
–	–	–	–	UNRE	TXBUFE	ENDTX	WRDY

- **WRDY: Write Ready for Synchronous Channels Update**

0: New duty-cycle and dead-time values for the synchronous channels cannot be written.

1: New duty-cycle and dead-time values for the synchronous channels can be written.

- **ENDTX: PDC End of TX Buffer**

0: The Transmit Counter register has not reached 0 since the last write of the PDC.

1: The Transmit Counter register has reached 0 since the last write of the PDC.

- **TXBUFE: PDC TX Buffer Empty**

0: PWM_TCR or PWM_TCNR has a value other than 0.

1: Both PWM_TCR and PWM_TCNR have a value other than 0.

- **UNRE: Synchronous Channels Update Underrun Error**

0: No Synchronous Channels Update Underrun has occurred since the last read of the PWM_ISR2 register.

1: At least one Synchronous Channels Update Underrun has occurred since the last read of the PWM_ISR2 register.

- **CMPMx: Comparison x Match**

0: The comparison x has not matched since the last read of the PWM_ISR2 register.

1: The comparison x has matched at least one time since the last read of the PWM_ISR2 register.

- **CMPUx: Comparison x Update**

0: The comparison x has not been updated since the last read of the PWM_ISR2 register.

1: The comparison x has been updated at least one time since the last read of the PWM_ISR2 register.

Note: Reading PWM_ISR2 automatically clears flags WRDY, UNRE and CMPSx.

40.4 Product Dependencies

For further details on the USB Device hardware implementation, see the specific Product Properties document.

The USB physical transceiver is integrated into the product. The bidirectional differential signals DDP and DDM are available from the product boundary.

One I/O line may be used by the application to check that VBUS is still available from the host. Self-powered devices may use this entry to be notified that the host has been powered off. In this case, the pull-up on DDP must be disabled in order to prevent feeding current to the host. The application should disconnect the transceiver, then remove the pull-up.

40.4.1 I/O Lines

The USB pins are shared with PIO lines. By default, the USB function is activated, and pins DDP and DDM are used for USB. To configure DDP or DDM as PIOs, the user needs to configure the system I/O configuration register (CCFG_SYSIO) in the MATRIX.

40.4.2 Power Management

The USB device peripheral requires a 48 MHz clock. This clock must be generated by a PLL driven by a clock source with an accuracy of ± 0.25% (note that the fast RC oscillator cannot be used).

Thus, the USB device receives two clocks from the Power Management Controller (PMC): the master clock, MCK, used to drive the peripheral user interface, and the UDPCK, used to interface with the bus USB signals (recovered 12 MHz domain).

WARNING: The UDP peripheral clock in the Power Management Controller (PMC) must be enabled before any read/write operations to the UDP registers including the UDP_TXVC register.

40.4.3 Interrupt

The USB device interface has an interrupt line connected to the Interrupt Controller.

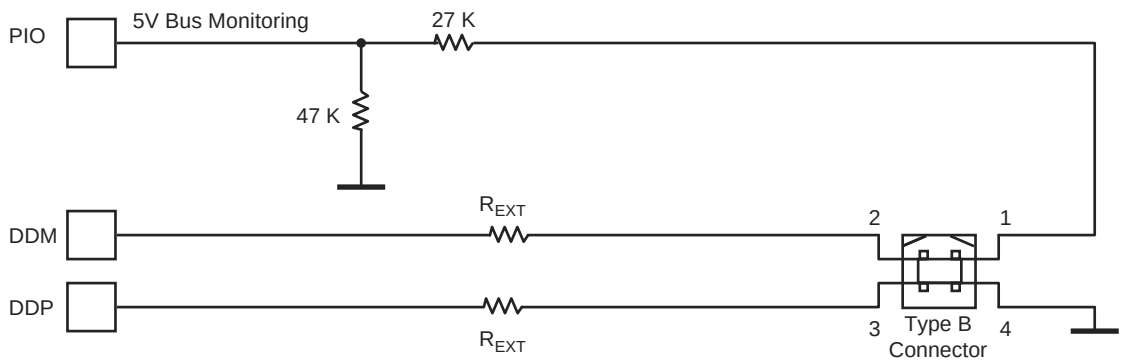
Handling the USB device interrupt requires programming the Interrupt Controller before configuring the UDP.

Table 40-3. Peripheral IDs

Instance	ID
UDP	34

40.5 Typical Connection

Figure 40-2. Board Schematic to Interface Device Peripheral



42.7.4 ADC Channel Sequence 2 Register

Name: ADC_SEQR2

Address: 0x4003800C

Access: Read/Write

31	30	29	28	27	26	25	24
–				USCH15			
23	22	21	20	19	18	17	16
USCH14				USCH13			
15	14	13	12	11	10	9	8
USCH12				USCH11			
7	6	5	4	3	2	1	0
USCH10				USCH9			

This register can only be written if the WPEN bit is cleared in the ADC Write Protection Mode Register.

- **USCHx: User Sequence Number x**

The sequence number x (USCHx) can be programmed by the Channel number CHy where y is the value written in this field. The allowed range is 0 up to 15. So it is only possible to use the sequencer from CH0 to CH15.

This register activates only if the USEQ field in ADC_MR is set to ‘1’.

Any USCHx field is processed only if the CHx field in ADC_CHSR reads logical ‘1’. Else any value written in USCHx does not add the corresponding channel in the conversion sequence.

Configuring the same value in different fields leads to multiple samples of the same channel during the conversion sequence. This can be done consecutively, or not, according to user needs.

47. Ordering Information

Devices in TFBGA, VFBGA, LQFP and QFN packages can be ordered in trays or in tape and reel. Devices in a WLCSP package are available in tape and reel only.

Table 47-1 provides ordering codes for tray packing. For tape and reel, append an ‘R’ to the tray ordering code; e.g., ATSAM4SD32CA-CUR.

Table 47-1. Ordering Codes for SAM4S Devices

Ordering Code	MRL	Flash (Kbytes)	SRAM (Kbytes)	Package	Carrier Type	Operating Temperature Range
ATSAM4SD32CA-CU	A	2*1024	160	TFBGA100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD32CB-CU	B					
ATSAM4SD32CA-CFU	A	2*1024	160	VFBGA100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD32CB-CFU	B					
ATSAM4SD32CA-AU	A	2*1024	160	LQFP100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD32CB-AU	B					
ATSAM4SD32CA-AN	A	2*1024	160	LQFP100	Tray	Industrial (-40°C to +105°C)
ATSAM4SD32CB-AN	B					
ATSAM4SD32BA-MU	A	2*1024	160	QFN64	Tray	Industrial (-40°C to +85°C)
ATSAM4SD32BB-MU	B					
ATSAM4SD32BA-AU	A	2*1024	160	LQFP64	Tray	Industrial (-40°C to +85°C)
ATSAM4SD32BB-AU	B					
ATSAM4SD32BA-AN	A	2*1024	160	LQFP64	Tray	Industrial (-40°C to +105°C)
ATSAM4SD32BB-AN	B					
ATSAM4SD32BA-UUR	A	2*1024	160	WLCSP64	Tape and reel	Industrial (-40°C to +85°C)
ATSAM4SD32BB-UUR	B					
ATSAM4SD16CA-CU	A	2*512	160	TFBGA100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD16CB-CU	B					
ATSAM4SD16CA-CFU	A	2*512	160	VFBGA100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD16CB-CFU	B					
ATSAM4SD16CA-AU	A	2*512	160	LQFP100	Tray	Industrial (-40°C to +85°C)
ATSAM4SD16CB-AU	B					
ATSAM4SD16CA-AN	A	2*512	160	LQFP100	Tray	Industrial (-40°C to +105°C)
ATSAM4SD16CB-AN	B					
ATSAM4SD16BA-MU	A	2*512	160	QFN64	Tray	Industrial (-40°C to +85°C)
ATSAM4SD16BB-MU	B					
ATSAM4SD16BA-AU	A	2*512	160	LQFP64	Tray	Industrial (-40°C to +85°C)
ATSAM4SD16BB-AU	B					
ATSAM4SD16BA-AN	A	2*512	160	LQFP64	Tray	Industrial (-40°C to +105°C)
ATSAM4SD16BB-AN	B					

49. Revision History

In the tables that follow, the most recent version of the document appears first.

Table 49-1. SAM4S Datasheet Rev. 11100K Revision History

Doc. Date	Changes
09-Jun-15	**"Features":** updated "Memories" section.
	Added section "Safety Features Highlight".
	Section 8., "Memories" Added Section 8.1.3.4 "Error Code Correction (ECC)".
	Section 44., "Electrical Characteristics" Added Table 44-24, "SAM4SD32/SA16/SD16 Typical Active Power Consumption with VDDCORE@ 1.2V running from Flash Memory (AMP2) or SRAM".

Table 49-2. SAM4S Datasheet Rev. 11100J Revision History

Doc. Date	Changes
28-May-15	**"Features":** updated "System" and "Peripherals" sections
	Section 2., "Block Diagram" Updated Figures
	Section 4., "Package and Pinout" Modified AD13 and AD14 position in Table 4-2 "SAM4SD32/SD16/SA16/S16/S8/S4/S2 100-ball TFBGA Pinout" and Table 4-3 "SAM4SD32/SD16/SA16/S16/S8/S4/S2 100-ball VFBGA Pinout"
	Section 44., "Electrical Characteristics" Updated Table 44-41, "ADC Timing Characteristics" Table 44-74, "AC Flash Characteristics": added one "Endurance" value
	Section 48., "Errata" Section 48.1.5, "Low-power Mode" and Section 48.3.4, "Low-power Mode": modified Workaround 2 (code example)

Table 49-6. SAM4S Datasheet Rev. 11100F 29-Jan-14 Revision History (Continued)

Doc. Date	Changes
	Section 16. "Real-time Clock (RTC)" Section 16.1 "Description": updated to explain need for accurate external 32.768 kHz clock Section 16.2 "Embedded Characteristics": added feature "Write-Protected Registers" Section 16.5.6 "Updating Time/Calendar": reworded second paragraph for clarity Section 16.5.7 "RTC Accurate Clock Calibration": replaced sentence "The period interval between 2 correction events is programmable in order to cover the possible crystal oscillator clock variations" with "According to the CORRECTION, NEGPPM and HIGHPPM values configured in the RTC Mode Register (RTC_MR), the period interval between two correction events differs" Section 16.6.1 "RTC Control Register", Section 16.6.2 "RTC Mode Register", Section 16.6.5 "RTC Time Alarm Register", Section 16.6.6 "RTC Calendar Alarm Register": added sentence "This register can only be written if the WPEN bit is cleared in the System Controller Write Protection Mode Register (SYSC_WPMR)" and updated description of UPDCAL bit Section 16.6.2 "RTC Mode Register": corrected typo (THIGH value 2 description now reads "3.91 ms")
	Section 18. "Supply Controller (SUPC)" Section 18.1 "Embedded Characteristics": added bullets on tamper detection and on anti-tampering. Figure 18-1 "Supply Controller Block Diagram" modified. Section 18.3.4 "Supply Monitor": Supply Monitor sampling mode, power reduction factor: replaced incorrect values of 32, 256 or 2048 by the correct values of 2, 16 and 128. Section 18.3.6.2 "Brownout Detector Reset": Reworked 1st paragraph for clarity Section 18.3.7.1 "Wake-up Inputs": corrected WKUPPLx pins to WKUPTx pins. WKUP0, WKUP15 references changed to WKUPx. Figure 18-4 "Wake-up Sources": Defined a section of the graphic as Low-power Tamper Detection Logic. Section 18.3.7.2 "Low-power Tamper Detection and Anti-Tampering": Changed all references to RTCOUT1 and RTCOUT 0 to RTCOUTx. Other minor modifications to improve clarity. Figure 18-5 "Low-power Debouncer (Push-to-Make Switch, Pull-up Resistors)", Figure 18-6 "Low-power Debouncer (Push-to-Break Switch, Pull-down Resistors)", Figure 18-7 "Using WKUP Pins Without RTCOUTx Pins": Modified pin names. Added Section 18.3.8 "Register Write Protection". In Section 18.4.9 "System Controller Write Protection Mode Register", updated register name and bit descriptions. Added Section 18.3.9 "Register Bits in Backup Domain (VDDIO)". Section 18.4.3 "Supply Controller Control Register": Added sentence on WPEN bit below register table and added note to descriptions of bits VROFF and XTALSEL indicating the bits are in the backup domain. Section 18.4.5 "Supply Controller Mode Register": Added sentence on WPEN bit below register table and added note to all bit descriptions except bit KEY indicating the bits are in the backup domain Section 18.4.4 "Supply Controller Supply Monitor Mode Register", , Section 18.4.6 "Supply Controller Wake-up Mode Register" and Section 18.4.7 "Supply Controller Wake-up Inputs Register": Added sentence on WPEN bit below register table and added a sentence below the register tables stating that the register is located in the backup domain Section 18.4.7 "Supply Controller Wake-up Inputs Register": corrected register name (was "System Controller Wake-Up Inputs Register")

Table 49-6. SAM4S Datasheet Rev. 11100F 29-Jan-14 Revision History (Continued)

Doc. Date	Changes
	Section 43. “Digital-to-Analog Converter Controller (DACC)” Section 43.7.7 “DACC Interrupt Enable Register”, Section 43.7.8 “DACC Interrupt Disable Register” and Section 43.7.9 “DACC Interrupt Mask Register”: modified bit descriptions. Rework of all “refresh” related paragraphs, Section 43.7.3 “DACC Channel Enable Register” and Section 43.6.7 “DACC Timings”. Modified description for ”REFRESH: Automatic Refresh Period” field in Section 43.7.2 “DACC Mode Register”. Re-worked Section 43.6.8 “Register Write Protection” and associated registers and bit/field descriptions in Section 43.7.12 “DACC Write Protection Mode Register” and Section 43.7.13 “DACC Write Protection Status Register”.
	Section 44. “Electrical Characteristics” Added Section 44.2 “Recommended Operating Conditions”. Section 44.4 “Power Consumption”: Added power consumption values for SAM4S4/SAM4S2. Updated Section 44.4.1 “Backup Mode Current Consumption”. Removed Supply Ripple Voltage parameter from Table 44-30, “3 to 20 MHz Crystal Oscillator Characteristics” Table 44-32 ”XIN Clock Electrical Characteristics (In Bypass Mode)”: Added $C_{PARASTANDBY}$ AND $R_{PARASTANDBY}$ parameters. Updated and re-worked Section 44.8 “12-bit ADC Characteristics”: Updated Section 44.9 “12-bit DAC Characteristics”. Removed Max Voltage Ripple parameter from Table 44-55, “Analog Power Supply Characteristics”. Added Refresh Time to Table 44-56, “Channel Conversion Time and DAC Clock”. In Section 44.12 “AC Characteristics” modified ● Table 44-64, “SPI Timings”. ● Table 44-65, “SSC Timings” ● Table 44-66, “SMC Read Signals - NRD Controlled (READ_MODE = 1)” ● Table 44-68, “SMC Write Signals - NWE Controlled (WRITE_MODE = 1)” ● Table 44-69, “SMC Write Signals - NCS Controlled (WRITE_MODE = 0)” ● Table 44-70, “USART SPI Timings” Table 44-71 ”Two-wire Serial Bus Requirements”: Added parameter t_{BUF} Section 44.12.9 “Embedded Flash Characteristics”: modified Table 44-72, “Embedded Flash Wait State at 105°C”. Table 44-73, “AC Flash Characteristics”: Full Chip Erase: Added values for 256 Kbytes and 128 Kbytes. Added new parameter Page Program Time.
	Section 45. “Mechanical Characteristics” Table 45-20 ”64-ball WLCSP Package Dimensions (in mm)” Added body size for SAM4S4 for WLCSP64 package. Figure 45-8 ”48-lead LQFP Package Drawing” and corresponding characteristics added. Figure 45-9 ”48-lead QFN Package Drawing” and corresponding characteristics added.
	Section 48. “Errata” Added Section 48.3 “Errata SAM4S4/S2 Rev. A Parts”.
	Section 47. “Ordering Information” Added information on carrier type availability. Updated Table 47-1 ”Ordering Codes for SAM4S Devices”. Added new ordering codes for SAM4S4 and SAM4S2 devices.