Microchip Technology - ATSAM4S2BA-AUR Datasheet

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	I²C, IrDA, Memory Card, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	47
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K 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	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsam4s2ba-aur

- Do not use SP and do not use PC.
- *RdHi* and *RdLo* must be different registers.

Condition Flags

These instructions do not affect the condition code flags.

Examples

```
SMLAL    R4, R5, R3, R8  ; Multiplies R3 and R8, adds R5:R4 and writes to
                         ; R5:R4
SMLALBT  R2, R1, R6, R7  ; Multiplies bottom halfword of R6 with top
                         ; halfword of R7, sign extends to 32-bit, adds
                         ; R1:R2 and writes to R1:R2
SMLALTB  R2, R1, R6, R7  ; Multiplies top halfword of R6 with bottom
                         ; halfword of R7,sign extends to 32-bit, adds R1:R2
                         ; and writes to R1:R2
SMLALD   R6, R8, R5, R1  ; Multiplies top halfwords in R5 and R1 and bottom
                         ; halfwords of R5 and R1, adds R8:R6 and writes to
                         ; R8:R6
SMLALDX  R6, R8, R5, R1  ; Multiplies top halfword in R5 with bottom
                         ; halfword of R1, and bottom halfword of R5 with
                         ; top halfword of R1, adds R8:R6 and writes to
                         ; R8:R6.
```

16.6.2 RTC Mode Register

Name: RTC_MR

Address: 0x400E1464

Access: Read/Write

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

23	22	21	20	19	18	17	16
–	OUT1			–	OUT0		

15	14	13	12	11	10	9	8
HIGHPPM	CORRECTION						

7	6	5	4	3	2	1	0
–	–	–	NEGPPM	–	–	PERSIAN	HRMOD

This register can only be written if the WPEN bit is cleared in the System Controller Write Protection Mode Register (SYSC_WPMR).

- **HRMOD: 12-/24-hour Mode**

0: 24-hour mode is selected.

1: 12-hour mode is selected.

- **PERSIAN: PERSIAN Calendar**

0: Gregorian calendar.

1: Persian calendar.

- **NEGPPM: NEGative PPM Correction**

0: Positive correction (the divider will be slightly higher than 32768).

1: Negative correction (the divider will be slightly lower than 32768).

Refer to CORRECTION and HIGHPPM field descriptions.

Note: NEGPPM must be cleared to correct a crystal slower than 32.768 kHz.

- **CORRECTION: Slow Clock Correction**

0: No correction

1–127: The slow clock will be corrected according to the formula given in HIGHPPM description.

- **HIGHPPM: HIGH PPM Correction**

0: Lower range ppm correction with accurate correction.

1: Higher range ppm correction with accurate correction.

If the absolute value of the correction to be applied is lower than 30 ppm, it is recommended to clear HIGHPPM. HIGHPPM set to 1 is recommended for 30 ppm correction and above.

Formula:

If HIGHPPM = 0, then the clock frequency correction range is from 1.5 ppm up to 98 ppm. The RTC accuracy is less than 1 ppm for a range correction from 1.5 ppm up to 30 ppm.

The correction field must be programmed according to the required correction in ppm; the formula is as follows:

Table 21-1. Signal Description List (Continued)

Signal Name	Function	Type	Active Level	Comments
PGMRDY	0: Device is busy 1: Device is ready for a new command	Output	High	Pulled-up input at reset
PGMNOE	Output Enable (active high)	Input	Low	Pulled-up input at reset
PGMNVALID	0: DATA[15:0] is in input mode 1: DATA[15:0] is in output mode	Output	Low	Pulled-up input at reset
PGMM[3:0]	Specifies DATA type (see Table 21-2)	Input	–	Pulled-up input at reset
PGMD[15:0]	Bi-directional data bus	Input/Output	–	Pulled-up input at reset

21.3.2 Signal Names

Depending on the MODE settings, DATA is latched in different internal registers.

Table 21-2. Mode Coding

MODE[3:0]	Symbol	Data
0000	CMDE	Command Register
0001	ADDR0	Address Register LSBs
0010	ADDR1	–
0011	ADDR2	–
0100	ADDR3	Address Register MSBs
0101	DATA	Data Register
Default	IDLE	No register

When MODE is equal to CMDE, then a new command (strobed on DATA[15:0] signals) is stored in the command register.

Table 21-3. Command Bit Coding

DATA[15:0]	Symbol	Command Executed
0x0011	READ	Read Flash
0x0012	WP	Write Page Flash
0x0022	WPL	Write Page and Lock Flash
0x0032	EWP	Erase Page and Write Page
0x0042	EWPL	Erase Page and Write Page then Lock
0x0013	EA	Erase All
0x0014	SLB	Set Lock Bit
0x0024	CLB	Clear Lock Bit
0x0015	GLB	Get Lock Bit
0x0034	SGPB	Set General Purpose NVM bit
0x0044	CGPB	Clear General Purpose NVM bit
0x0025	GGPB	Get General Purpose NVM bit
0x0054	SSE	Set Security Bit

22.5.5 Cache Controller Maintenance Register 0

Name: CMCC_MAINT0

Address: 0x4007C020

Access: 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
–	–	–	–	–	–	–	–

7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	INVALL

- **INVALL: Cache Controller Invalidate All**

0: No effect.

1: All cache entries are invalidated.

27.3 Peripheral DMA Controller Connections

The Peripheral DMA Controller handles the data transfer between peripherals and memory and receives triggers from the peripherals listed in the following table.

The Peripheral DMA Controller handles transfer requests from the channel according to the following priorities (Channel 0 is high priority):

Table 27-1. Peripheral DMA Controller

Instance Name	Channel T/R	Channel Number
PWM	Transmit	21
TWI1	Transmit	20
TWI0	Transmit	19
UART1	Transmit	18
UART0	Transmit	17
USART1	Transmit	16
USART0	Transmit	15
DACC	Transmit	14
SPI	Transmit	13
SSC	Transmit	12
HSMCI	Transmit	11
PIOA	Receive	10
TWI1	Receive	9
TWI0	Receive	8
UART1	Receive	7
UART0	Receive	6
USART1	Receive	5
USART0	Receive	4
ADC	Receive	3
SPI	Receive	2
SSC	Receive	1
HSMCI	Receive	0

27.6.6 Receive Next Counter Register

Name: PERIPH_RNCR

Access: Read/Write

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

23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
RXNCTR							

7	6	5	4	3	2	1	0
RXNCTR							

- **RXNCTR: Receive Next Counter**

RXNCTR contains the next receive buffer size.

When a half-duplex peripheral is connected to the PDC, RXNCTR = TXNCTR.

the slow clock until it reaches 0. At this time, the LOCK bit is set in PMC_SR and can trigger an interrupt to the processor. The user has to load the number of slow clock cycles required to cover the PLL transient time into the PLLCOUNT field.

The PLL clock can be divided by 2 by writing the PLLDIV2 (PLLADIV2, PLLBDIV2) bit in PMC_MCKR.

It is prohibited to change the 4/8/12 MHz fast RC oscillator or the main oscillator selection in CKGR_MOR while the master clock source is the PLL and the PLL reference clock is the fast RC oscillator.

The user must:

1. Switch on the main RC oscillator by writing a 1 to the CSS field of PMC_MCKR.
2. Change the frequency (MOSCRCF) or oscillator selection (MOSCSEL) in CKGR_MOR.
3. Wait for MOSCRCS (if frequency changes) or MOSCSELS (if oscillator selection changes) in PMC_SR.
4. Disable and then enable the PLL.
5. Wait for the LOCK flag in PMC_SR.
6. Switch back to the PLL by writing the appropriate value to the CSS field of PMC_MCKR.

29.17.23PMC Peripheral Clock Enable Register 1

Name: PMC_PCER1

Address: 0x400E0500

Access: 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
–	–	–	–	–	–	–	–

7	6	5	4	3	2	1	0
–	–	–	–	–	PID34	PID33	PID32

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

- **PIDx: Peripheral Clock x Enable**

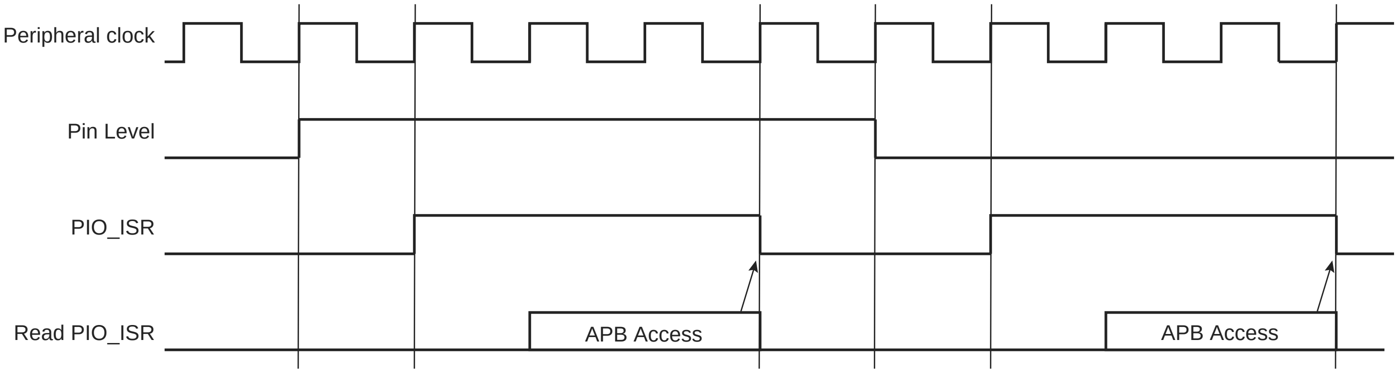
0: No effect.

1: Enables the corresponding peripheral clock.

Notes: 1. The values for PIDx are defined in the section “Peripheral Identifiers” in the product datasheet.

2. Programming the control bits of the Peripheral ID that are not implemented has no effect on the behavior of the PMC.

Figure 31-7. Input Change Interrupt Timings When No Additional Interrupt Modes



31.5.11 I/O Lines Lock

When an I/O line is controlled by a peripheral (particularly the Pulse Width Modulation Controller PWM), it can become locked by the action of this peripheral via an input of the PIO Controller. When an I/O line is locked, the write of the corresponding bit in PIO_PER, PIO_PDR, PIO_MDER, PIO_MDDR, PIO_PUDR, PIO_PUER, PIO_ABCDSR1 and PIO_ABCDSR2 is discarded in order to lock its configuration. The user can know at anytime which I/O line is locked by reading the PIO Lock Status Register (PIO_LOCKSR). Once an I/O line is locked, the only way to unlock it is to apply a hardware reset to the PIO Controller.

31.5.12 Programmable Schmitt Trigger

It is possible to configure each input for the Schmitt trigger. By default the Schmitt trigger is active. Disabling the Schmitt trigger is requested when using the QTouch® Library.

31.5.13 Parallel Capture Mode

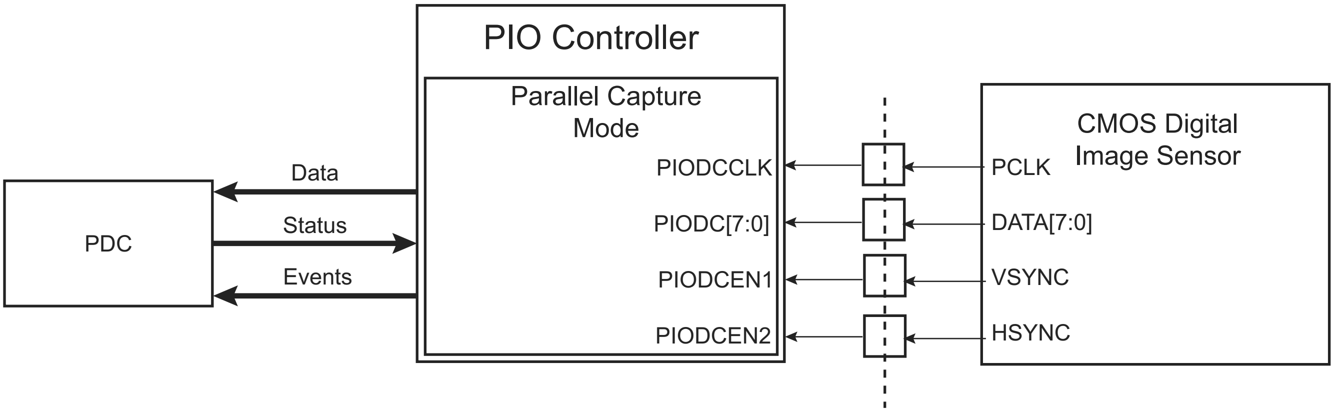
31.5.13.1 Overview

The PIO Controller integrates an interface able to read data from a CMOS digital image sensor, a high-speed parallel ADC, a DSP synchronous port in synchronous mode, etc. For better understanding and to ease reading, the following description uses an example with a CMOS digital image sensor.

31.5.13.2 Functional Description

The CMOS digital image sensor provides a sensor clock, an 8-bit data synchronous with the sensor clock and two data enables which are also synchronous with the sensor clock.

Figure 31-8. PIO Controller Connection with CMOS Digital Image Sensor



31.6.7 PIO Input Filter Enable Register

Name: PIO_IFER

Address: 0x400E0E20 (PIOA), 0x400E1020 (PIOB), 0x400E1220 (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

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

- **P0–P31: Input Filter Enable**

0: No effect.

1: Enables the input glitch filter on the I/O line.

31.6.17 PIO Interrupt Status Register

Name: PIO_ISR

Address: 0x400E0E4C (PIOA), 0x400E104C (PIOB), 0x400E124C (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: Input Change Interrupt Status**

0: No input change has been detected on the I/O line since PIO_ISR was last read or since reset.

1: At least one input change has been detected on the I/O line since PIO_ISR was last read or since reset.

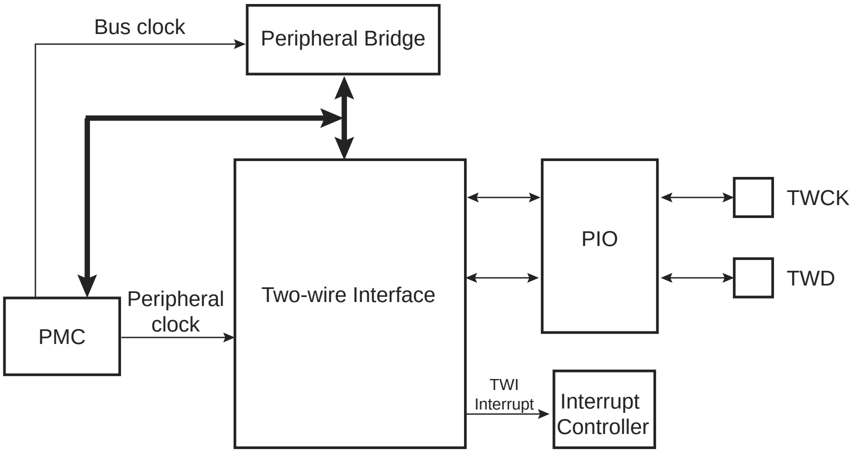
34.3 List of Abbreviations

Table 34-2. Abbreviations

Abbreviation	Description
TWI	Two-wire Interface
A	Acknowledge
NA	Non Acknowledge
P	Stop
S	Start
Sr	Repeated Start
SADR	Slave Address
ADR	Any address except SADR
R	Read
W	Write

34.4 Block Diagram

Figure 34-1. Block Diagram

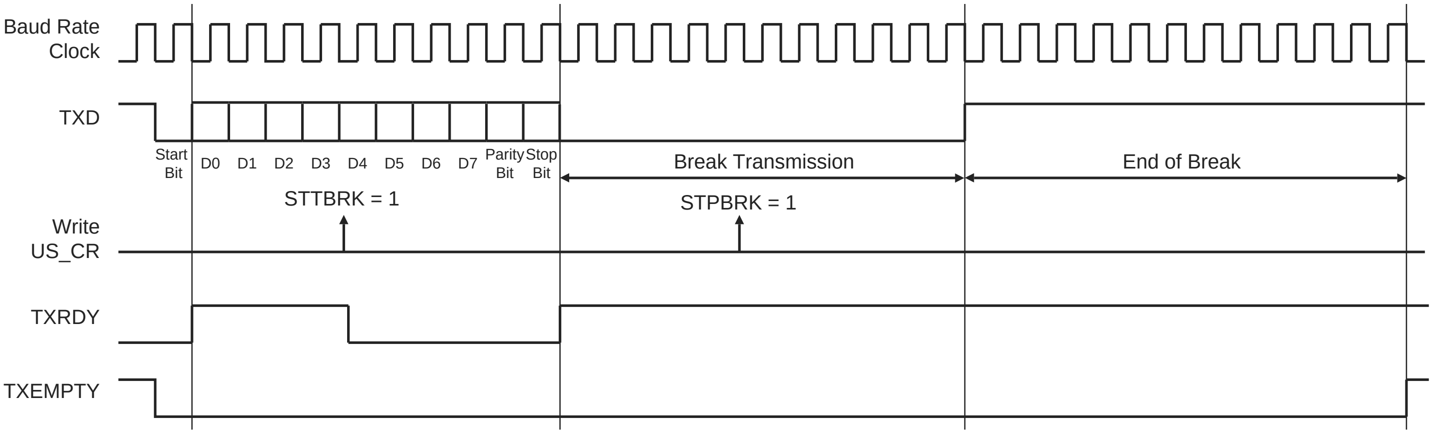


34.5 I/O Lines Description

Table 34-3. I/O Lines Description

Name	Description	Type
TWD	Two-wire Serial Data (drives external serial data line – SDA)	Input/Output
TWCK	Two-wire Serial Clock (drives external serial clock line – SCL)	Input/Output

Figure 36-25. Break Transmission



36.6.3.14 Receive Break

The receiver detects a break condition when all data, parity and stop bits are low. This corresponds to detecting a framing error with data to 0x00, but FRAME remains low.

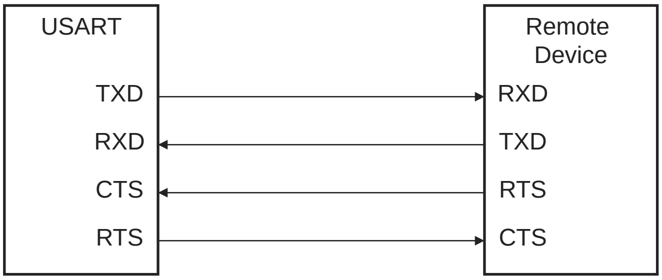
When the low stop bit is detected, the receiver asserts the RXBRK bit in US_CSR. This bit may be cleared by writing a 1 to the RSTSTA bit in the US_CR.

An end of receive break is detected by a high level for at least 2/16 of a bit period in Asynchronous operating mode or one sample at high level in Synchronous operating mode. The end of break detection also asserts the RXBRK bit.

36.6.3.15 Hardware Handshaking

The USART features a hardware handshaking out-of-band flow control. The RTS and CTS pins are used to connect with the remote device, as shown in Figure 36-26.

Figure 36-26. Connection with a Remote Device for Hardware Handshaking



Setting the USART to operate with hardware handshaking is performed by writing the USART_MODE field in US_MR to the value 0x2.

The USART behavior when hardware handshaking is enabled is the same as the behavior in standard Synchronous or Asynchronous mode, except that the receiver drives the RTS pin as described below and the level on the CTS pin modifies the behavior of the transmitter as described below. Using this mode requires using the PDC channel for reception. The transmitter can handle hardware handshaking in any case.

Figure 36-27 shows how the receiver operates if hardware handshaking is enabled. The RTS pin is driven high if the receiver is disabled or if the status RXBUFF (Receive Buffer Full) coming from the PDC channel is high. Normally, the remote device does not start transmitting while its CTS pin (driven by RTS) is high. As soon as the receiver is enabled, the RTS falls, indicating to the remote device that it can start transmitting. Defining a new buffer in the PDC clears the status bit RXBUFF and, as a result, asserts the pin RTS low.

36.7.13 USART Receive Holding Register

Name: US_RHR

Address: 0x40024018 (0), 0x40028018 (1)

Access: 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
RXSYNH	–	–	–	–	–	–	RXCHR

7	6	5	4	3	2	1	0
RXCHR							

- **RXCHR: Received Character**

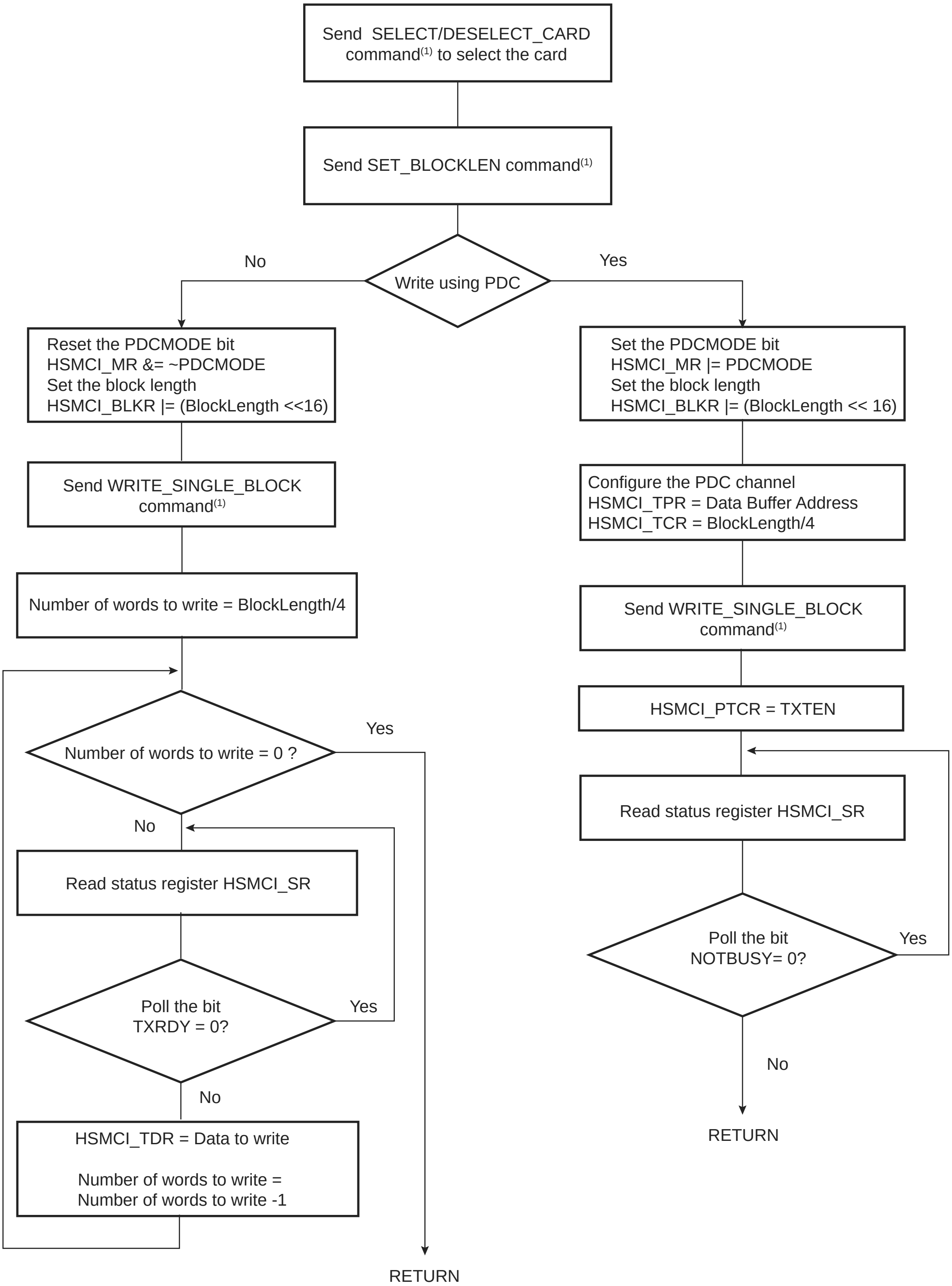
Last character received if RXRDY is set.

- **RXSYNH: Received Sync**

0: Last character received is a data.

1: Last character received is a command.

Figure 38-9. Write Functional Flow Diagram


Send SELECT/DESELECT_CARD command(1) to select the card
Send SET_BLOCKLEN command(1)
Write using PDC
No
Yes
Reset the PDCMODE bit
HSMCI_MR &= ~PDCMODE
Set the block length
HSMCI_BLKR |= (BlockLength <<16)
Send WRITE_SINGLE_BLOCK command(1)
Number of words to write = BlockLength/4
Number of words to write = 0 ?
Yes
No
Read status register HSMCI_SR
Poll the bit TXRDY = 0?
Yes
No
HSMCI_TDR = Data to write
Number of words to write = Number of words to write -1
RETURN
Set the PDCMODE bit
HSMCI_MR |= PDCMODE
Set the block length
HSMCI_BLKR |= (BlockLength << 16)
Configure the PDC channel
HSMCI_TPR = Data Buffer Address
HSMCI_TCR = BlockLength/4
Send WRITE_SINGLE_BLOCK command(1)
HSMCI_PTCR = TXTEN
Read status register HSMCI_SR
Poll the bit NOTBUSY= 0?
Yes
No
RETURN

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

The flowchart in Figure 38-10 shows how to manage a multiple write block transfer with the PDC. Polling or interrupt method can be used to wait for the end of write according to the contents of the HSMCI_IMR.

42.7.1 ADC Control Register

Name: ADC_CR

Address: 0x40038000

Access: 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**
–	–	–	–	–	–	–	–
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
–	–	–	–	AUTOCAL	–	START	SWRST

- **SWRST: Software Reset**

0: No effect.

1: Resets the ADC, simulating a hardware reset.

- **START: Start Conversion**

0: No effect.

1: Begins analog-to-digital conversion.

- **AUTOCAL: Automatic Calibration of ADC**

0: No effect.

1: Launches an automatic calibration of the ADC cell on the next sequence.

42.7.2 ADC Mode Register

Name: ADC_MR

Address: 0x40038004

Access: Read/Write

31	30	29	28	27	26	25	24
USEQ	–	TRANSFER		TRACKTIM			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
ANACH	–	SETTLING		STARTUP			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
PRESCAL							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FREERUN	FWUP	SLEEP	–	TRGSEL			TRGEN

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

- **TRGEN: Trigger Enable**

Value	Name	Description
0	DIS	Hardware triggers are disabled. Starting a conversion is only possible by software.
1	EN	Hardware trigger selected by TRGSEL field is enabled.

- **TRGSEL: Trigger Selection**

Value	Name	Description
0	ADC_TRIG0	External trigger
1	ADC_TRIG1	TIO Output of the Timer Counter Channel 0
2	ADC_TRIG2	TIO Output of the Timer Counter Channel 1
3	ADC_TRIG3	TIO Output of the Timer Counter Channel 2
4	ADC_TRIG4	PWM Event Line 0
5	ADC_TRIG5	PWM Event Line 1
6	ADC_TRIG6	Reserved
7	ADC_TRIG7	Reserved

- **SLEEP: Sleep Mode**

Value	Name	Description
0	NORMAL	Normal Mode: The ADC core and reference voltage circuitry are kept ON between conversions.
1	SLEEP	Sleep Mode: The wake-up time can be modified by programming FWUP bit.

- **FWUP: Fast Wake Up**

Value	Name	Description
0	OFF	If SLEEP is 1, then both ADC core and reference voltage circuitry are OFF between conversions
1	ON	If SLEEP is 1, then Fast Wake-up Sleep mode: The voltage reference is ON between conversions and ADC core is OFF

For power-saving options, see Section 43.6.6 ”DACC Timings”.

43.5.2 Interrupt Sources

The DACC interrupt line is connected to one of the internal sources of the interrupt controller. Using the DACC interrupt requires the interrupt controller to be programmed first.

Table 43-2. Peripheral IDs

Instance	ID
DACC	30

43.5.3 Conversion Performances

For performance and electrical characteristics of the DACC, see the product DC Characteristics section of the datasheet.

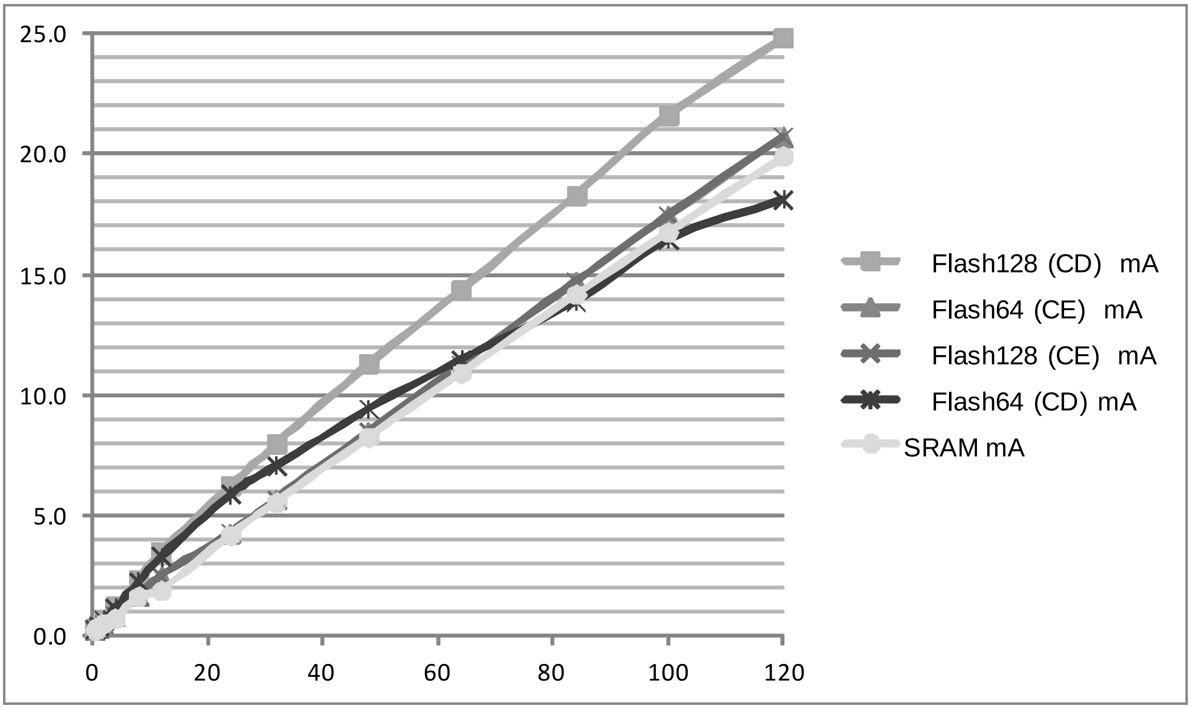
44.4.3.3 SAM4SD32/SD16/SA16 Active Power Consumption

Table 44-23. SAM4SD32/SA16/SD16 Active Power Consumption with VDDCORE @ 1.2V running from Flash Memory (IDDCORE- AMP1) or SRAM

	CoreMark					
	Cache Enable (CE)		Cache Disable (CD)			
Core Clock (MHz)	128-bit Flash access[1]	64-bit Flash access[1]	128-bit Flash access[1]	64-bit Flash access[1]	SRAM	Unit
120	20.7	20.7	24.8	18.1	19.9	mA
100	17.5	17.4	21.6	16.5	16.8	
84	14.7	14.7	18.3	13.9	14.2	
64	11.3	11.3	14.4	11.5	10.9	
48	8.5	8.5	11.3	9.4	8.3	
32	5.7	5.7	8.0	7.1	5.6	
24	4.3	4.3	6.3	5.9	4.2	
12	2.5	2.5	3.5	3.3	1.9	
8	1.7	1.7	2.4	2.3	1.7	
4	0.9	0.9	1.2	1.2	0.7	
2	0.5	0.5	0.7	0.7	0.5	
1	0.4	0.4	0.5	0.5	0.4	
0.5	0.3	0.3	0.3	0.3	0.3	

Note: 1. Flash Wait State (FWS) in EEFC_FMR adjusted versus core frequency

Figure 44-12. SAM4SD32/SD16/SA16 Current Consumption in Active Mode (AMP1) versus Master Clock Ranges



44.10 Analog Comparator Characteristics

Table 44-61. Analog Comparator Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_R	Voltage Range	Analog comparator is supplied by VDDIN	1.62	3.3	3.6	V
V_{IR}	Input Voltage Range		GND + 0.2	–	VDDIN - 0.2	V
V_{IO}	Input Offset Voltage		–	–	20	mV
I_{VDDIN}	Current Consumption (VDDIN)	Low-Power Option (ACC_ACR.ISEL = 0)	–	–	25	µA
		High-Speed Option (ACC_ACR.ISEL = 1)	–	–	170	
V_{hys}	Hysteresis Voltage	ACC_ACR.HYST = 0x01 or 0x10	–	15	50	mV
		ACC_ACR.HYST = 0x11	–	30	90	
t_{sa}	Settling Time	Overdrive > 100 mV; Low-power option	–	–	1	µs
		Overdrive > 100 mV; High-speed option	–	–	0.1	

44.11 Temperature Sensor

The temperature sensor is connected to channel 15 of the ADC.

The temperature sensor provides an output voltage (V_{O_TS}) that is proportional to absolute temperature (PTAT). V_{O_TS} linearly varies with a temperature slope dV_{O_TS}/dT = 4.7 mV/°C.

V_{O_TS} equals 1.44 V at T_A 27°C, with a ±60 mV accuracy. The V_{O_TS} slope versus temperature dV_{O_TS}/dT = 4.7 mV/°C only shows a ±7% slight variation over process, mismatch and supply voltage.

The user needs to calibrate it (offset calibration) at ambient temperature in order to get rid of the V_{O_TS} spread at ambient temperature (±15%).

Table 44-62. Temperature Sensor Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{O_TS}	Output Voltage	T_A = 27°C (1)		1.44	–	V
$V_{O\_TS(accuracy)}$	Output Voltage Accuracy	T_A = 27°C (1)	-60	–	60	mV
dV_{O_TS}/dT	Temperature Sensitivity (Slope Voltage vs Temperature)	(1)	–	4.7	–	mV/°C
	Slope Accuracy	Over temperature range -40 to 105 °C (1)	-7	–	7	%
	Temperature Accuracy(2)	After offset calibration over temperature range -40 to 105 °C	-6	–	6	°C
		After offset calibration over temperature range 0 to 80 °C	-5	–	5	°C
t_{START}	Startup Time	After ADC_ACR.TSON = 1 (1)	–	5	10	µs
$I_{VDDCORE}$	Current Consumption	(1)	50	70	80	µA

Note: 1. The value of TS only (the value does not take into account the ADC offset/gain/errors)

2. The temperature accuracy takes into account the ADC offset error and gain error in single-ended mode with Gain = 1.