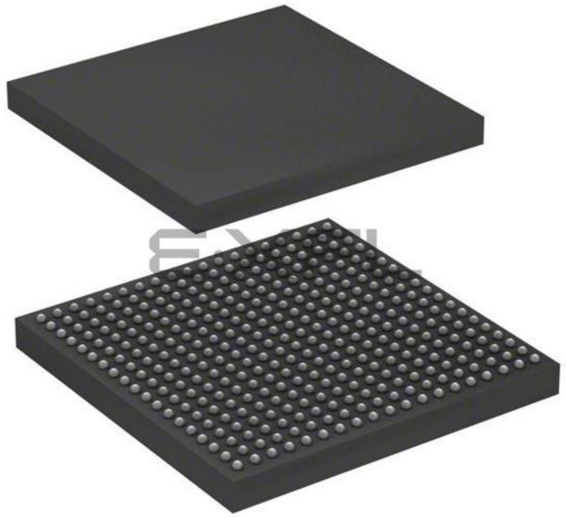
Microchip Technology - ATSAMA5D44B-CUR Datasheet



Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Active
Core Processor	ARM® Cortex®-A5
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	600MHz
Co-Processors/DSP	Multimedia; NEON™ SIMD
RAM Controllers	LPDDR, LPDDR2, DDR2
Graphics Acceleration	Yes
Display & Interface Controllers	LCD, Touchscreen
Ethernet	10/100Mbps (1)
SATA	-
USB	USB 2.0 (3)
Voltage - I/O	1.2V, 1.8V, 3.3V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	AES, SHA, TDES, TRNG
Package / Case	361-TFBGA
Supplier Device Package	361-TFBGA (16x16)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsama5d44b-cur

17.9.7 AIC Interrupt Pending Register 0

Name: AIC_IPR0

Address:0xFC06E020 (AIC), 0xFC068420 (SAIC)

Access: Read-only

31	30	29	28	27	26	25	24
PID31	PID30	PID29	PID28	PID27	PID26	PID25	PID24

23	22	21	20	19	18	17	16
PID23	PID22	PID21	PID20	PID19	PID18	PID17	PID16

15	14	13	12	11	10	9	8
PID15	PID14	PID13	PID12	PID11	PID10	PID9	PID8

7	6	5	4	3	2	1	0
PID7	PID6	PID5	PID4	PID3	PID2	SYS	FIQ

FIQ: Interrupt Pending

0: The corresponding interrupt is not pending.

1: The corresponding interrupt is pending.

SYS: Interrupt Pending

0: The corresponding interrupt is not pending.

1: The corresponding interrupt is pending.

PIDx: Interrupt Pending

0: The corresponding interrupt is not pending.

1: The corresponding interrupt is pending.

22.6 Real-time Clock (RTC) User Interface

Table 22-2: Register Mapping

Offset	Register	Name	Access	Reset
0x00	Control Register	RTC_CR	Read/Write	0x00000000
0x04	Mode Register	RTC_MR	Read/Write	0x00000000
0x08	Time Register	RTC_TIMR	Read/Write	0x00000000
0x0C	Calendar Register	RTC_CALR	Read/Write	0x01E111220
0x10	Time Alarm Register	RTC_TIMALR	Read/Write	0x00000000
0x14	Calendar Alarm Register	RTC_CALALR	Read/Write	0x01010000
0x18	Status Register	RTC_SR	Read-only	0x00000000
0x1C	Status Clear Command Register	RTC_SCCR	Write-only	–
0x20	Interrupt Enable Register	RTC_IER	Write-only	–
0x24	Interrupt Disable Register	RTC_IDR	Write-only	–
0x28	Interrupt Mask Register	RTC_IMR	Read-only	0x00000000
0x2C	Valid Entry Register	RTC_VER	Read-only	0x00000000
0xB0	TimeStamp Time Register 0	RTC_TSTR0	Read-only	0x00000000
0xB4	TimeStamp Date Register 0	RTC_TSDR0	Read-only	0x00000000
0xB8	TimeStamp Source Register 0	RTC_TSSR0	Read-only	0x00000000
0xBC	TimeStamp Time Register 1	RTC_TSTR1	Read-only	0x00000000
0xC0	TimeStamp Date Register 1	RTC_TSDR1	Read-only	0x00000000
0xC4	TimeStamp Source Register 1	RTC_TSSR1	Read-only	0x00000000
0xC8	Reserved	–	–	–
0xCC	Reserved	–	–	–
0xD0	Reserved	–	–	–
0xD4–0xF8	Reserved	–	–	–
0xFC	Reserved	–	–	–

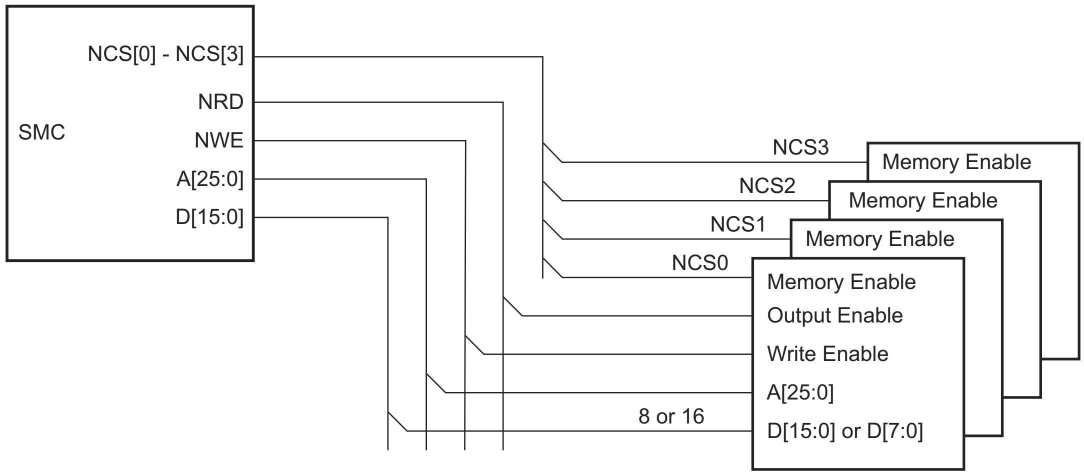
Note: If an offset is not listed in the table it must be considered as reserved.

Table 29-31: Register Mapping (Continued)

Offset	Register	Name	Access	Reset
0x10C	MPDDRC DLL Slave Offset 1 Register	MPDDRC_DLL_SO1	Read/Write	0x0(1)
0x110	MPDDRC DLL CLKWR Offset Register	MPDDRC_DLL_WRO	Read/Write	0x0(1)
0x114	MPDDRC DLL CLKAD Offset Register	MPDDRC_DLL_ADO	Read/Write	0x0(1)
0x118	MPDDRC DLL Status Master 0 Register	MPDDRC_DLL_SM0	Read-only	0x00000000
0x11C	MPDDRC DLL Status Master 1 Register	MPDDRC_DLL_SM1	Read-only	0x00000000
0x120	MPDDRC DLL Status Master 2 Register	MPDDRC_DLL_SM2	Read-only	0x00000000
0x124	MPDDRC DLL Status Master 3 Register	MPDDRC_DLL_SM3	Read-only	0x00000000
0x128	MPDDRC DLL Status Slave 0 Register	MPDDRC_DLL_SSL0	Read-only	0x00000000
0x12C	MPDDRC DLL Status Slave 1 Register	MPDDRC_DLL_SSL1	Read-only	0x00000000
0x130	MPDDRC DLL Status Slave 2 Register	MPDDRC_DLL_SSL2	Read-only	0x00000000
0x134	MPDDRC DLL Status Slave 3 Register	MPDDRC_DLL_SSL3	Read-only	0x00000000
0x138	MPDDRC DLL Status Slave 4 Register	MPDDRC_DLL_SSL4	Read-only	0x00000000
0x13C	MPDDRC DLL Status Slave 5 Register	MPDDRC_DLL_SSL5	Read-only	0x00000000
0x140	MPDDRC DLL Status Slave 6 Register	MPDDRC_DLL_SSL6	Read-only	0x00000000
0x144	MPDDRC DLL Status Slave 7 Register	MPDDRC_DLL_SSL7	Read-only	0x00000000
0x148	MPDDRC DLL Status CLKWR 0 Register	MPDDRC_DLL_SWR0	Read-only	0x00000000
0x14C	MPDDRC DLL Status CLKWR 1 Register	MPDDRC_DLL_SWR1	Read-only	0x00000000
0x150	MPDDRC DLL Status CLKWR 2 Register	MPDDRC_DLL_SWR2	Read-only	0x00000000
0x154	MPDDRC DLL Status CLKWR 3 Register	MPDDRC_DLL_SWR3	Read-only	0x00000000
0x158	MPDDRC DLL Status CLKAD Register	MPDDRC_DLL_SAD	Read-only	0x00000000
0x15C–0x1FC	Reserved	–	–	–

Note 1: Values vary with the product implementation.

Figure 30-3: Memory Connections for External Devices



30.9 Connection to External Devices

30.9.1 Data Bus Width

A data bus width of 8 or 16 bits can be selected for each chip select. This option is controlled by the bit DBW in the SMC Mode Register (HSMC_MODE) for the corresponding chip select.

Figure 30-4 shows how to connect a 512 KB x 8-bit memory on NCS2. Figure 30-5 shows how to connect a 512 KB x 16-bit memory on NCS2.

Figure 30-4: Memory Connection for an 8-bit Data Bus

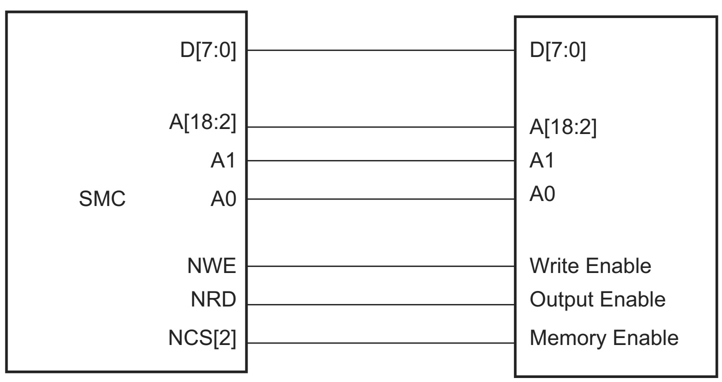


Figure 30-5: Memory Connection for a 16-bit Data Bus

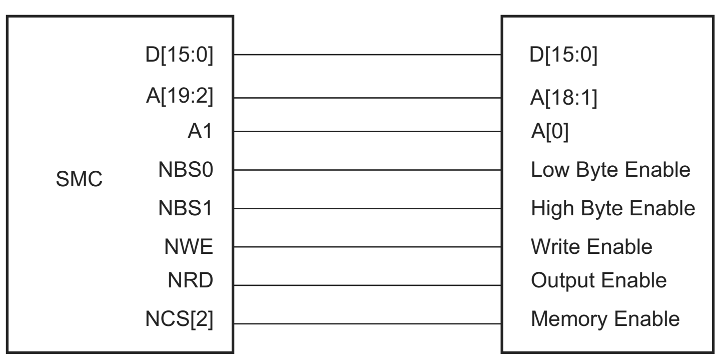


Figure 30-14: Early Read Wait State: Write with No Hold Followed by Read with No Setup

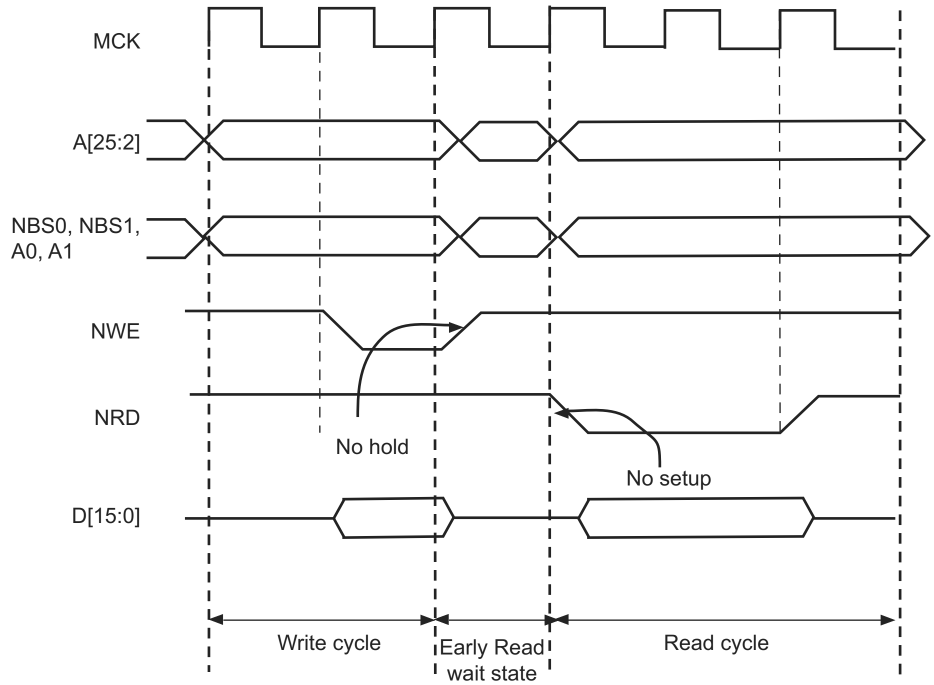
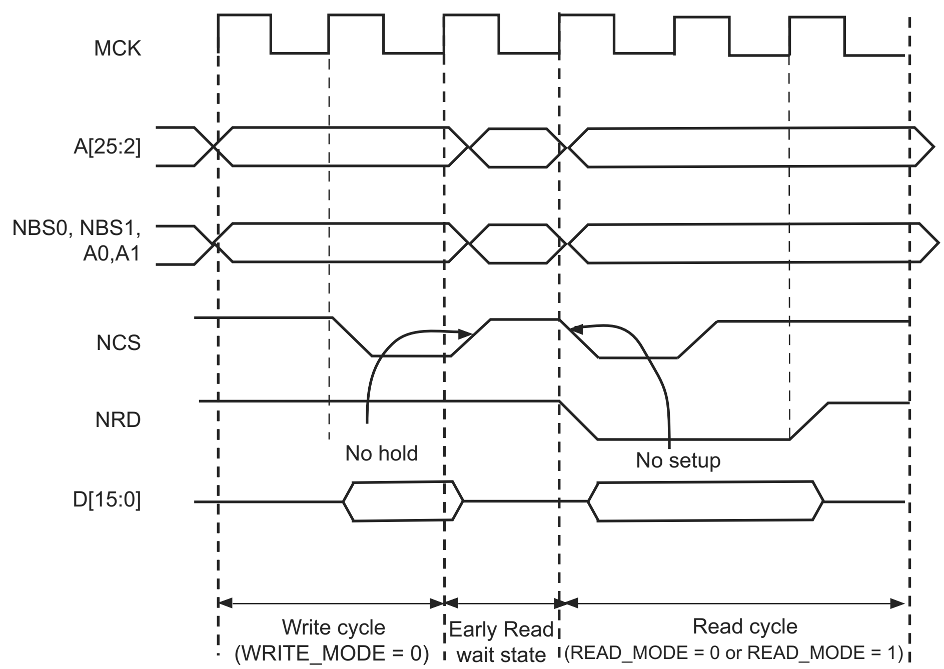


Figure 30-15: Early Read Wait State: NCS Controlled Write with No Hold Followed by a Read with No NCS



Setup

```
/*      End Compute Sigma (Mu+1)                 */
/*      And L(mu)                                */
/*************************************************/
/* In either case compute delta */
delta[i+1]  = (mu[i+1] * 2 - lmu[i+1]) >> 1;
/* In either case compute the discrepancy */
for (k = 0 ; k <= (lmu[i+1]>>1); k++)
{
    if (k == 0)
    dmu[i+1] = si[2*(i-1)+3];
    /* check if one operand of the multiplier is null, its index is -1 */
    else if (smu[i+1][k] && si[2*(i-1)+3-k])
    dmu[i+1] = gf_antilog[(gf_log[smu[i+1][k]] + gf_log[si[2*(i-1)+3-k]])%nn] ^ dmu[i+1];
}
}
return 0;
}
```

30.19.3 Finding the Error Position

The output of the get_sigma() procedure is a polynomial stored in the smu[NB_ERROR+1][] table. The error positions are the roots of that polynomial. The degree of that polynomial is a very important information, as it gives the number of errors. PMERRLOC module provides hardware accelerator for that step.

30.19.3.1 Error Location

The PMECC Error Location controller provides hardware acceleration for determining roots of polynomials over two finite fields: GF(2^13) and GF(2^14). It integrates 24 fully programmable coefficients. These coefficients belong to GF(2^13) or GF(2^14). The coefficient programmed in the PMERRLOC{*i*} is the coefficient of X ^ *i* in the polynomial.

The search operation is started as soon as a write access is detected in the ELEN register and can be disabled writing to the ELDIS register. The ENINIT field of the ELEN register shall be initialized with the number of galois field elements to test. The set of the roots can be limited to a valid range.

Table 30-18: ENINIT Field Value for a Sector Size of 512 Bytes

Error Correcting Capability	ENINIT Value
2	4122
4	4148
8	4200
12	4252
24	4408

Table 30-19: ENINIT Field Value for a Sector Size of 1024 Bytes

Error Correcting Capability	ENINIT Value
2	8220
4	8248
8	8304
12	8360
24	8528

When the PMECC engine is searching for roots, the BUSY field of the ELSR register remains asserted. An interrupt is asserted at the end of the computation, and the DONE bit of the PMECC Error Location Interrupt Status Register (HSMC_ELSIR) is set. The ERR_CNT field of the HSMC_ELISR indicates the number of errors. The error position can be read in the PMERRLOCX registers.

30.20.20 PMECC Remainder x Register

Name: HSMC_REMx [x=0..11] [sec_num=0..7]

Address:0xFC05C2B0 [0][0] .. 0xFC05C2DC [11][0]
0xFC05C2F0 [0][1] .. 0xFC05C31C [11][1]
0xFC05C330 [0][2] .. 0xFC05C35C [11][2]
0xFC05C370 [0][3] .. 0xFC05C39C [11][3]
0xFC05C3B0 [0][4] .. 0xFC05C3DC [11][4]
0xFC05C3F0 [0][5] .. 0xFC05C41C [11][5]
0xFC05C430 [0][6] .. 0xFC05C45C [11][6]
0xFC05C470 [0][7] .. 0xFC05C49C [11][7]

Access: Read-only

31	30	29	28	27	26	25	24
–	–	REM2NP3					

23	22	21	20	19	18	17	16
REM2NP3							

15	14	13	12	11	10	9	8
–	–	REM2NP1					

7	6	5	4	3	2	1	0
REM2NP1							

REM2NP1: BCH Remainder 2 * N + 1

When sector size is set to 512 bytes, bit REM2NP1[13] is not used and read as zero.

If bit *i* of the REM2NP1 field is set to one, then the coefficient of the X ^ *i* is set to one; otherwise, the coefficient is zero.

REM2NP3: BCH Remainder 2 * N + 3

When sector size is set to 512 bytes, bit REM2NP3[29] is not used and read as zero.

If bit *i* of the REM2NP3 field is set to one, then the coefficient of the X ^ *i* is set to one; otherwise, the coefficient is zero.

32.7.123 High End Overlay Configuration Register 28

Name: LCDC_HEOCFG28

Address:0xF00003FC

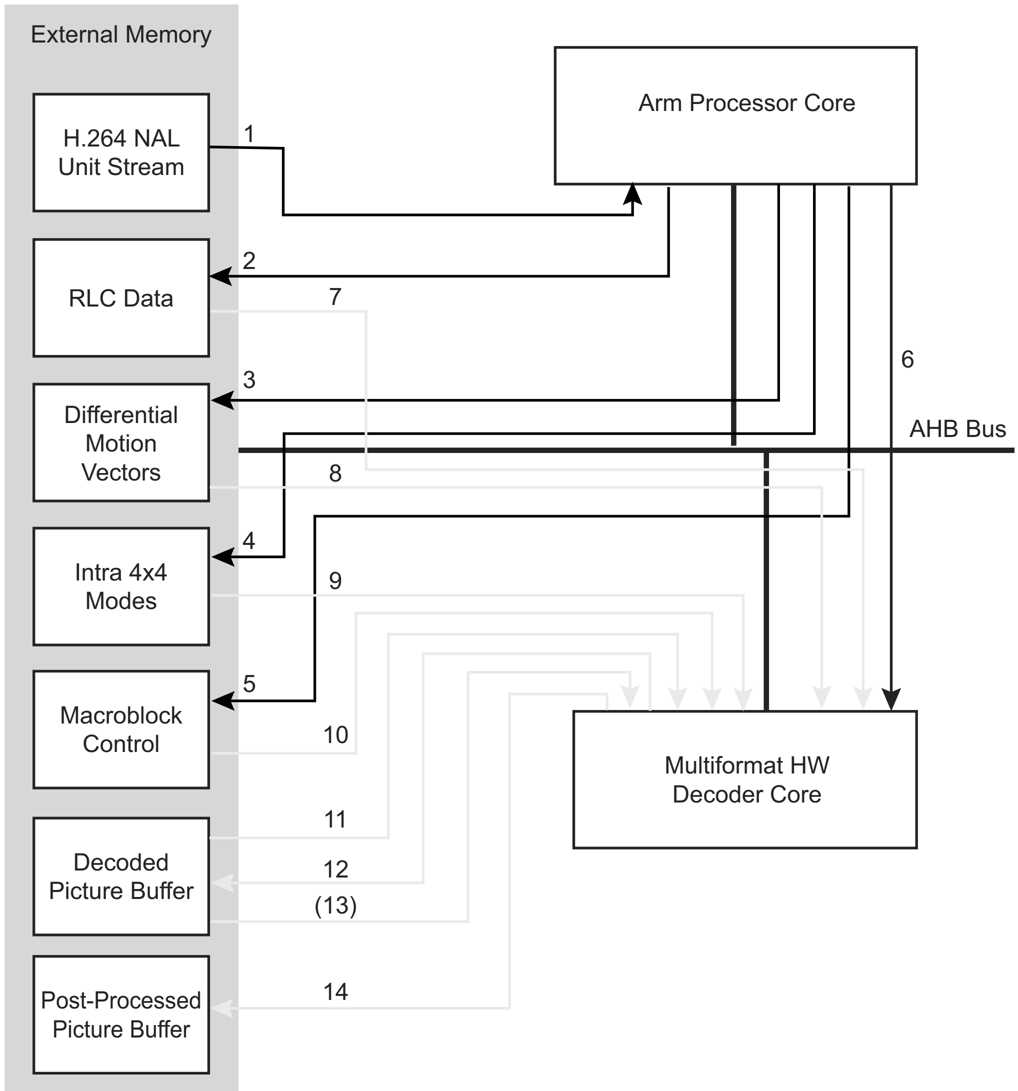
Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
–	–	–	–	–	–	–	–
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
–	–	–	–	–	–	–	–
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
XPHI5COEFF4							

XPHI5COEFF4: Horizontal Coefficient for phase 5 tap 4

Coefficient format is 1 sign bit and 7 fractional bits.

Figure 33-3: Multi-format Decoder and External Memory Data Flow in RLC Mode



37.8.6 GMAC Transmit Status Register

Name:GMAC_TSR

Address:0xF8020014 (0), 0xFC028014 (1)

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**
–	–	–	–	–	–	–	HRESP
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
–	UND	TXCOMP	TFC	TXGO	RLE	COL	UBR

UBR: Used Bit Read

Set when a transmit buffer descriptor is read with its used bit set. Writing a one clears this bit.

COL: Collision Occurred

Set by the assertion of collision. Writing a one clears this bit. When operating in 10/100 mode, this status indicates either a collision or a late collision.

RLE: Retry Limit Exceeded

Writing a one clears this bit.

TXGO: Transmit Go

Transmit go, if high transmit is active. When using the DMA interface this bit represents the TXGO variable as specified in the transmit buffer description.

TFC: Transmit Frame Corruption Due to AHB Error

Transmit frame corruption due to AHB error. Set if an error occurs while midway through reading transmit frame from the AHB, including HRESP errors and buffers exhausted mid frame (if the buffers run out during transmission of a frame then transmission stops, FCS shall be bad and GTXER asserted).

Writing a one clears this bit.

TXCOMP: Transmit Complete

Set when a frame has been transmitted. Writing a one clears this bit.

UND: Transmit Underrun

This bit is set if the transmitter was forced to terminate a frame that it had already began transmitting due to further data being unavailable.

This bit is set if a transmitter status write back has not completed when another status write back is attempted.

When using the DMA interface configured for internal FIFO mode, this bit is also set when the transmit DMA has written the SOP data into the FIFO and either the AHB bus was not granted in time for further data, or because an AHB not OK response was returned, or because a used bit was read.

Writing a one clears this bit.

HRESP: HRESP Not OK

Set when the DMA block sees HRESP not OK. Writing a one clears this bit.

38.14.17 HSMCI Configuration Register

Name: HSMCI_CFG

Address:0xF8000054 (0), 0xFC000054 (1)

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**
–	–	–	LSYNC	–	–	–	HSMODE
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
–	–	–	FERRCTRL	–	–	–	FIFOMODE

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

FIFOMODE: HSMCI Internal FIFO control mode

0: A write transfer starts when a sufficient amount of data is written into the FIFO.

When the block length is greater than or equal to 3/4 of the HSMCI internal FIFO size, then the write transfer starts as soon as half the FIFO is filled. When the block length is greater than or equal to half the internal FIFO size, then the write transfer starts as soon as one quarter of the FIFO is filled. In other cases, the transfer starts as soon as the total amount of data is written in the internal FIFO.

1: A write transfer starts as soon as one data is written into the FIFO.

FERRCTRL: Flow Error flag reset control mode

0: When an underflow/overflow condition flag is set, a new Write/Read command is needed to reset the flag.

1: When an underflow/overflow condition flag is set, a read status resets the flag.

HSMODE: High Speed Mode

0: Default bus timing mode.

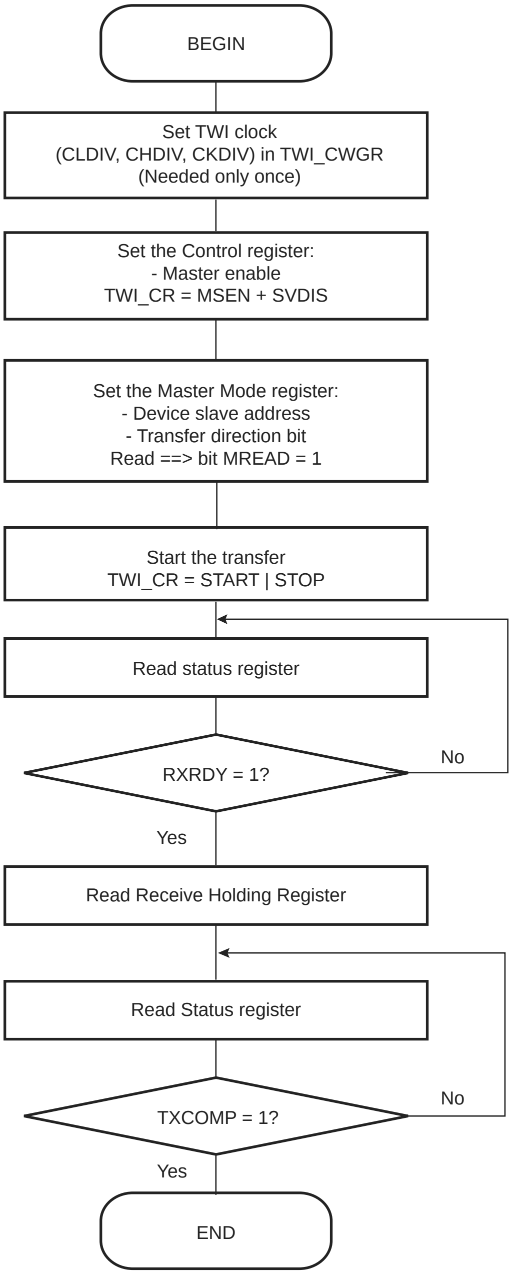
1: If set to one, the host controller outputs command line and data lines on the rising edge of the card clock. The Host driver shall check the high speed support in the card registers.

LSYNC: Synchronize on the last block

0: The pending command is sent at the end of the current data block.

1: The pending command is sent at the end of the block transfer when the transfer length is not infinite (block count shall be different from zero).

Figure 40-16: TWI Read Operation with Single Data Byte without Internal Address



41.9.11 SSC Receive Compare 0 Register

Name:SSC_RC0R

Address:0xF8008038 (0), 0xFC014038 (1)

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**
CP0							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CP0							

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

CP0: Receive Compare Data 0

41.9.15 SSC Interrupt Disable Register

Name: SSC_IDR

Address: 0xF8008048 (0), 0xFC014048 (1)

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
–	–	–	–	RXSYN	TXSYN	CP1	CP0

7	6	5	4	3	2	1	0
–	–	OVRUN	RXRDY	–	–	TXEMPTY	TXRDY

TXRDY: Transmit Ready Interrupt Disable

0: No effect.

1: Disables the Transmit Ready Interrupt.

TXEMPTY: Transmit Empty Interrupt Disable

0: No effect.

1: Disables the Transmit Empty Interrupt.

RXRDY: Receive Ready Interrupt Disable

0: No effect.

1: Disables the Receive Ready Interrupt.

OVRUN: Receive Overrun Interrupt Disable

0: No effect.

1: Disables the Receive Overrun Interrupt.

CP0: Compare 0 Interrupt Disable

0: No effect.

1: Disables the Compare 0 Interrupt.

CP1: Compare 1 Interrupt Disable

0: No effect.

1: Disables the Compare 1 Interrupt.

TXSYN: Tx Sync Interrupt Enable

0: No effect.

1: Disables the Tx Sync Interrupt.

RXSYN: Rx Sync Interrupt Enable

0: No effect.

1: Disables the Rx Sync Interrupt.

44.7.16 USART Receiver Timeout Register

Name: US_RTOR

Address: 0xF802C024 (0), 0xF8030024 (1), 0xFC008024 (2), 0xFC00C024 (3), 0xFC010024 (4)

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**
TO							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TO							

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

TO: Timeout Value

0: The receiver timeout is disabled.

1–65535: The receiver timeout is enabled and TO is Timeout Delay / Bit Period.

44.7.20 USART IrDA Filter Register

Name: US_IF

Address: 0xF802C04C (0), 0xF803004C (1), 0xFC00804C (2), 0xFC00C04C (3), 0xFC01004C (4)

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**
–	–	–	–	–	–	–	–
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IRDA_FILTER							

This register is relevant only if USART_MODE = 0x8 in the USART Mode Register.

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

IRDA_FILTER: IrDA Filter

The IRDA_FILTER value must be defined to meet the following criteria:

$$t_{\text{peripheral clock}} \times (\text{IRDA\_FILTER} + 3) < 1.41\ \mu s$$

47.7.25 PWM Fault Clear Register

Name: PWM_FCR

Address: 0xF800C064

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
FCLR							

Refer to Section 47.5.4 “Fault Inputs” for details on fault generation.

FCLR: Fault Clear

For each bit y of FCLR, where y is the fault input number:

0: No effect.

1: If bit y of FMOD field is set to ‘1’ and if the fault input y is not at the level defined by the bit y of FPOL field, the fault y is cleared and becomes inactive (FMOD and FPOL fields belong to PWM Fault Mode Register), else writing this bit to ‘1’ has no effect.

52. Secure Hash Algorithm (SHA)

52.1 Description

The Secure Hash Algorithm (SHA) is compliant with the American *FIPS (Federal Information Processing Standard) Publication 180-2* specification.

The 512/1024-bit block of message is respectively stored in 16/32 x 32-bit registers, (SHA_IDATARx/SHA_IODATARx) which are write-only.

As soon as the input data is written, the hash processing may be started. The registers comprising the block of a padded message must be entered consecutively. Then the message digest is ready to be read out on the 5 up to 8/16 x 32-bit output data registers (SHA_IODATARx) or through the DMA channels.

52.2 Embedded Characteristics

- Supports Secure Hash Algorithm (SHA1, SHA224, SHA256, SHA384, SHA512)
- Compliant with *FIPS Publication 180-2*
- Supports initial hash values registers (HMAC acceleration or other)
- Configurable Processing Period:
  - 85 Clock Cycles to obtain a fast SHA1 runtime, 88 clock cycles for SHA384, SHA512 or 209 Clock Cycles for Maximizing Bandwidth of Other Applications
  - 72 Clock Cycles to obtain a fast SHA224, SHA256 runtime or 194 Clock Cycles for Maximizing Bandwidth of Other Applications
- Connection to DMA Channel Capabilities Optimizes Data Transfers
- Double Input Buffer Optimizes Runtime

52.3 Product Dependencies

52.3.1 Power Management

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

52.3.2 Interrupt Sources

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

Handling the SHA interrupt requires programming the interrupt controller before configuring the SHA.

Table 52-1: Peripheral IDs

Instance	ID
SHA	15

52.4 Functional Description

The Secure Hash Algorithm (SHA) module requires a padded message according to FIPS180-2 specification. The first block of the message must be indicated to the module by a specific command. The SHA module produces an N-bit message digest each time a block is written and processing period ends, where N is 160 for SHA1, 224 for SHA224,256 for SHA256, 384 for SHA384, 512 for SHA512.

52.4.1 SHA Algorithm

The SHA can process SHA1, SHA224, SHA256, SHA384, SHA512 by configuring the ALGO field in the SHA Mode register (SHA_MR).

52.4.2 Processing Period

The processing period can be configured.

The short processing period allocates bandwidth to the SHA module, whereas the long processing period allocates more bandwidth on the system bus to other applications. An example is DMA channels not associated with SHA.

In SHA1 mode, the shortest processing period is 85 clock cycles + 2 clock cycles for start command synchronization. The longest period is 209 clock cycles + 2 clock cycles.

54.6.7 ICM Interrupt Status Register

Name: ICM_ISR

Address:0xFC04001C

Access: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	URAD
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
RSU				REC			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
RWC				RBE			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
RDM				RHC			

RHC: Region Hash Completed

When RHC[*i*] is set, it indicates that the ICM has completed the region with identifier *i*.

RDM: Region Digest Mismatch

When RDM[*i*] is set, it indicates that there is a digest comparison mismatch between the hash value of the region with identifier *i* and the reference value located in the Hash Area.

RBE: Region Bus Error

When RBE[*i*] is set, it indicates that a bus error has been detected while hashing memory region *i*.

RWC: Region Wrap Condition Detected

When RWC[*i*] is set, it indicates that a wrap condition has been detected.

REC: Region End bit Condition Detected

When REC[*i*] is set, it indicates that an end bit condition has been detected.

RSU: Region Status Updated Detected

When RSU[*i*] is set, it indicates that a region status updated condition has been detected.

URAD: Undefined Register Access Detection Status

0: No undefined register access has been detected since the last SWRST.

1: At least one undefined register access has been detected since the last SWRST.

The URAD bit is only reset by the SWRST bit in the ICM_CTRL register.

The URAT field in the ICM_UASR indicates the unspecified access type.

58. Schematic Checklist

The schematic checklist provides the user with the requirements regarding the different pin connections that must be considered before starting any new board design. It also provides information on the minimum hardware resources required to quickly develop an application with the SAMA5D4. It does not consider PCB layout constraints.

It also provides recommendations regarding low-power design constraints to minimize power consumption.

This information is not intended to be exhaustive. Its objective is to cover as many configurations of use as possible.

The checklist contains a column for use by designers, making it easy to track and verify each line item.