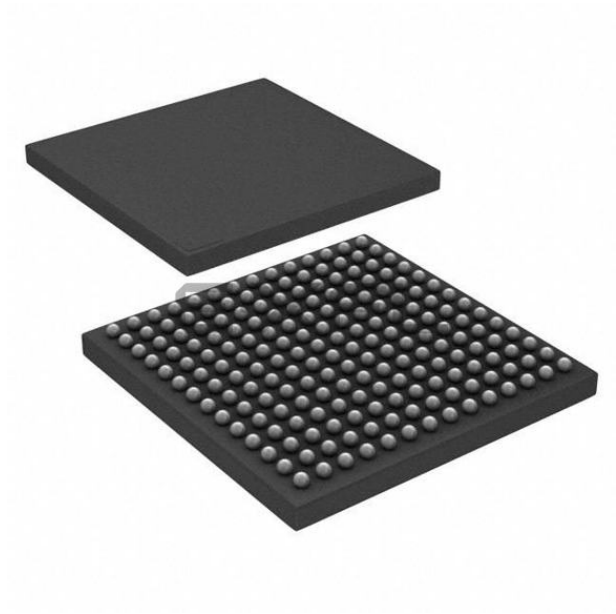


Welcome to E-XFL.COM

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.

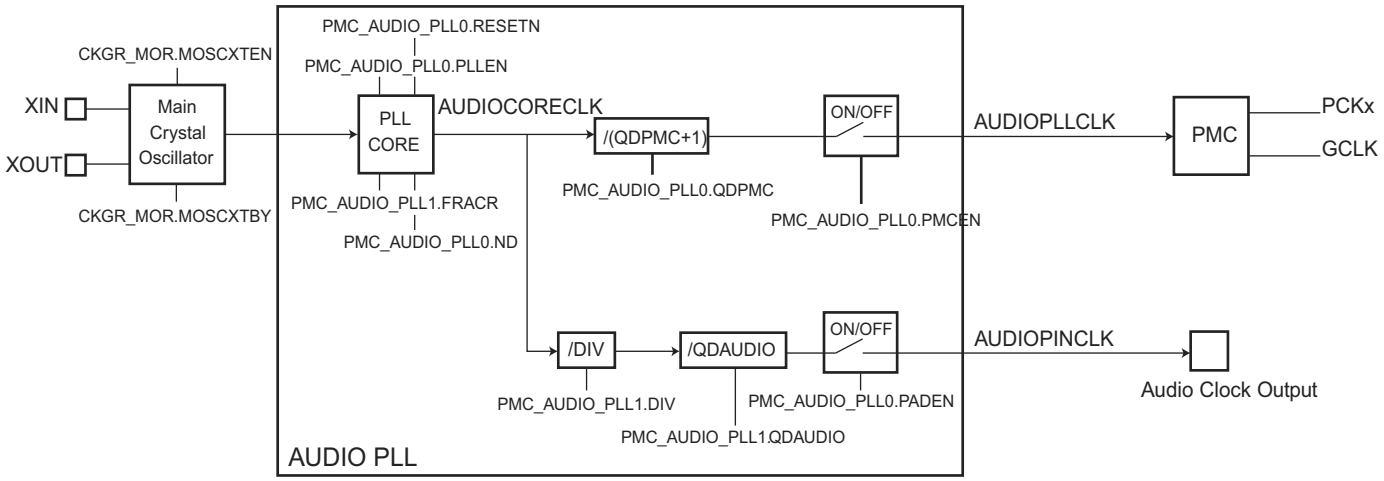
Applications of Embedded - Microprocessors

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	500MHz
Co-Processors/DSP	Multimedia; NEON™ MPE
RAM Controllers	LPDDR1, LPDDR2, LPDDR3, DDR2, DDR3, DDR3L, QSPI
Graphics Acceleration	Yes
Display & Interface Controllers	Keyboard, LCD, Touchscreen
Ethernet	10/100Mbps (1)
SATA	-
USB	USB 2.0 + HSIC
Voltage - I/O	3.3V
Operating Temperature	-40°C ~ 105°C (TA)
Security Features	ARM TZ, Boot Security, Cryptography, RTIC, Secure Fusebox, Secure JTAG, Secure Memory, Secure RTC
Package / Case	196-TFBGA, CSBGA
Supplier Device Package	196-TFBGA (11x11)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsama5d22a-cn

6. If needed, ND or FRACR can be adjusted at any time. The typical frequency settling time of this PLL is indicated in Section 66. “Electrical Characteristics”.

Figure 32-7: Audio PLL



KEY: Password

Value	Name	Description
0x37	PASSWD	Writing any other value in this field aborts the write operation.

MOSCSEL: Main Clock Oscillator Selection

0: The 12 MHz oscillator is selected.

1: The 8 to 24 MHz crystal oscillator is selected.

CFDEN: Clock Failure Detector Enable

0: The clock failure detector is disabled.

1: The clock failure detector is enabled.

When the glitch and/or debouncing filter is enabled, it does not modify the behavior of the inputs on the peripherals. It acts only on the value read in PIO_PDSRx and on the input change interrupt detection. The glitch and debouncing filters require that the PIO Controller clock is enabled.

Figure 34-4: Input Glitch Filter Timing

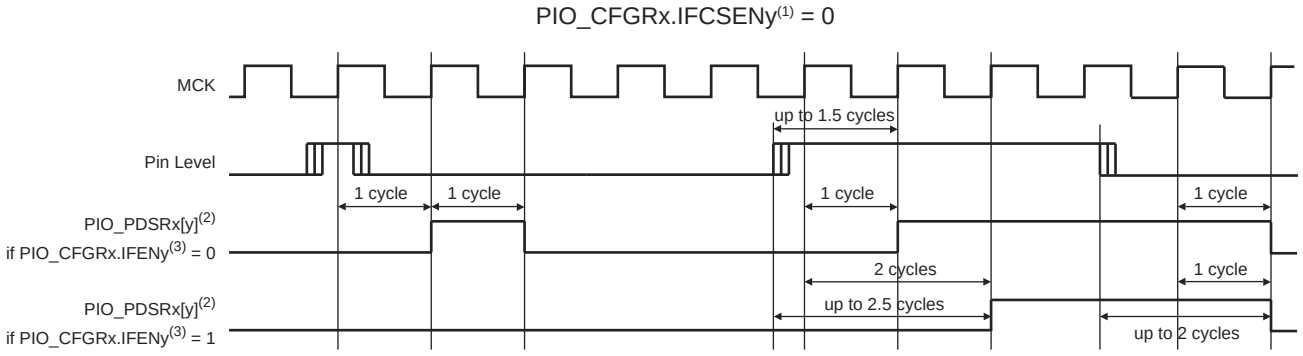
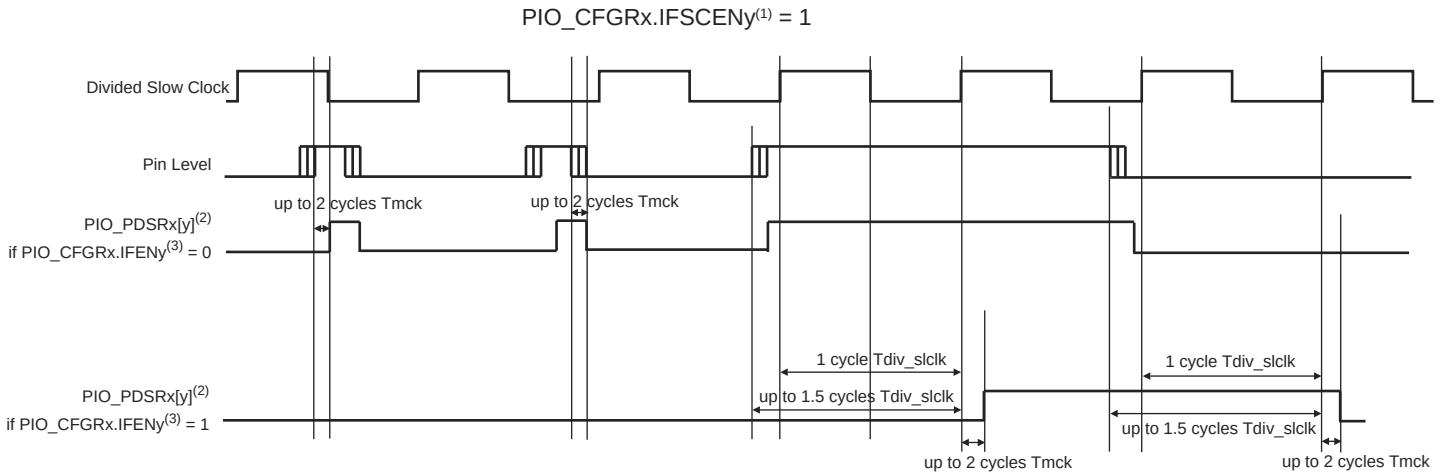


Figure 34-5: Input Debouncing Filter Timing



Note 1: Means IFCSEN bit of the I/O line y of the I/O group x

2: Means PIO Data Status value of the I/O line y of the I/O group x

3: Means IFEN bit of the I/O line y of the I/O group x

34.5.10 Input Edge/Level Interrupt

Each I/O group can be programmed to generate an interrupt when it detects an edge or a level on an I/O line. The Input Edge/Level interrupts are controlled by writing the PIO Interrupt Enable Register (PIO_IERx) and the PIO Interrupt Disable Register (PIO_IDRx), which enable and disable the input change interrupt respectively by setting and clearing the corresponding bit in the PIO Interrupt Mask Register (PIO_IMRx). For the Secure I/O lines, the Input Edge/Level interrupts are controlled by writing S_PIO_IERx and S_PIO_IDRx, which enable and disable input change interrupts respectively by setting and clearing the corresponding bit in the S_PIO_IMRx. As input change detection is possible only by comparing two successive samplings of the input of the I/O line, the PIO Controller clock must be enabled. The Input Change interrupt is available regardless of the configuration of the I/O line, i.e., configured as an input only, controlled by the PIO Controller or assigned to a peripheral function.

Each I/O group can generate a Non-Secure interrupt and a Secure interrupt according to the security level of the I/O line which triggers the interrupt.

According to the EVTSEL field value in the PIO Configuration Register (PIO_CFGRx) or the Secure PIO Configuration Register (S_PIO_CFGRx) in case of a Secure I/O line, the interrupt signal of the I/O group x can be generated on the following occurrence:

- (S_)PIO_CFGRx.EVTSELy = 0: The interrupt signal of the I/O group x is generated on the I/O line y falling edge detection (assuming that (S_)PIO_IMRx[y] = 1).

Table 36-31: Register Mapping (Continued)

Offset	Register	Name	Access	Reset
0x7C	Monitor Address High/Low Port 6 Register	MPDDRC_MADDR6	Read/Write	0x00000000
0x80	Monitor Address High/Low Port 7 Register	MPDDRC_MADDR7	Read/Write	0x00000000
0x84	Monitor Information Port 0 Register	MPDDRC_MINFO0	Read-only	0x00000000
0x88	Monitor Information Port 1 Register	MPDDRC_MINFO1	Read-only	0x00000000
0x8C	Monitor Information Port 2 Register	MPDDRC_MINFO2	Read-only	0x00000000
0x90	Monitor Information Port 3 Register	MPDDRC_MINFO3	Read-only	0x00000000
0x94	Monitor Information Port 4 Register	MPDDRC_MINFO4	Read-only	0x00000000
0x98	Monitor Information Port 5 Register	MPDDRC_MINFO5	Read-only	0x00000000
0x9C	Monitor Information Port 6 Register	MPDDRC_MINFO6	Read-only	0x00000000
0xA0	Monitor Information Port 7 Register	MPDDRC_MINFO7	Read-only	0x00000000
0xA4–0xE0	Reserved	–	–	–
0xE4	Write Protection Mode Register	MPDDRC_WPMR	Read/Write	0x00000000
0xE8	Write Protection Status Register	MPDDRC_WPSR	Read-only	0x00000000
0xEC–0x1FC	Reserved	–	–	–

37.20.9 PMECC Configuration Register

Name: HSMC_PMECCFG

Address: 0xF8014070

Access: Read/Write

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

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

15	14	13	12	11	10	9	8
–	–	–	NANDWR	–	–	PAGESIZE	

7	6	5	4	3	2	1	0
–	–	–	SECTORSZ	–	BCH_ERR		

BCH_ERR: Error Correcting Capability

Value	Name	Description
0	BCH_ERR2	2 errors
1	BCH_ERR4	4 errors
2	BCH_ERR8	8 errors
3	BCH_ERR12	12 errors
4	BCH_ERR24	24 errors
5	BCH_ERR32	32 errors

SECTORSZ: Sector Size

0: The ECC computation is based on a sector of 512 bytes.

1: The ECC computation is based on a sector of 1024 bytes.

PAGESIZE: Number of Sectors in the Page

Value	Name	Description
0	PAGESIZE_1SEC	1 sector for main area (512 or 1024 bytes)
1	PAGESIZE_2SEC	2 sectors for main area (1024 or 2048 bytes)
2	PAGESIZE_4SEC	4 sectors for main area (2048 or 4096 bytes)
3	PAGESIZE_8SEC	8 sectors for main area (4096 or 8192 bytes)

NANDWR: NAND Write Access

0: NAND read access

1: NAND write access

6. Program XDMAC_CCx register (see single block transfer configuration).
7. Program XDMAC_CBCx.BLEN with the number of microblocks of data.
8. Clear the following four registers:
   - XDMAC_CNDCx
   - XDMAC_CDS_MSPx
   - XDMAC_CSUSx
   - XDMAC_CDUSx

   This indicates that the linked list is disabled and striding is disabled.
9. Enable the Block interrupt by writing a ‘1’ to XDMAC_CIEx.BIE, enable the Channel x Interrupt Enable bit by writing a ‘1’ to XDMAC_GIEx.IEx.
10. Enable channel x by writing a ‘1’ to the XDMAC_GE.ENx. XDMAC_GS.STx is set by hardware.
11. Once completed, the DMA channel sets XDMAC_CISx.BIS (End of Block Interrupt Status bit) and generates an interrupt. XDMAC_GS.STx is cleared by hardware. The software can either wait for an interrupt or poll the channel status bit.

38.5.4.3 Master Transfer

1. Read the XDMAC_GS register to choose a free channel.
2. Clear the pending Interrupt Status bit by reading the chosen XDMAC_CISx register.
3. Build a linked list of transfer descriptors in memory. The descriptor view is programmable on a per descriptor basis. The linked list items structure must be word aligned. MBR_UBC.NDE must be configured to 0 in the last descriptor to terminate the list.
4. Configure field NDA in the XDMAC Channel x Next Descriptor Address Register (XDMAC_CNDAx) with the first descriptor address and bit XDMAC_CNDAx.NDAIF with the master interface identifier.
5. Configure the XDMAC_CNDCx register:
   a) Set XDMAC_CNDCx.NDE to enable the descriptor fetch.
   b) Set XDMAC_CNDCx.NDSUP to update the source address at the descriptor fetch time, otherwise clear this bit.
   c) Set XDMAC_CNDCx.NDDUP to update the destination address at the descriptor fetch time, otherwise clear this bit.
   d) Configure XDMAC_CNDCx.NDVIEW to define the length of the first descriptor.
6. Enable the End of Linked List interrupt by writing a ‘1’ to XDMAC_CIEx.LIE.
7. Enable channel x by writing a ‘1’ to XDMAC_GE.ENx. XDMAC_GS.STx is set by hardware.
8. Once completed, the DMA channel sets XDMAC_CISx.BIS (End of Block Interrupt Status bit) and generates an interrupt. XDMAC_GS.STx is cleared by hardware. The software can either wait for an interrupt or poll the channel status bit.

38.5.4.4 Disabling A Channel Before Transfer Completion

Under normal operation, the software enables a channel by writing a ‘1’ to XDMAC_GE.ENx, then the hardware disables a channel on transfer completion by clearing bit XDMAC_GS.STx. To disable a channel, write a ‘1’ to bit XDMAC_GD.DIx and poll the XDMAC_GS register.

UOVR: Overflow for U or UV Chrominance Interrupt Enable

0: No effect

1: Interrupt source is enabled

VDMA: End of DMA for V Chrominance Transfer Interrupt Enable

0: No effect

1: Interrupt source is enabled

VDSCR: Descriptor Loaded for V Chrominance Interrupt Enable

0: No effect

1: Interrupt source is enabled

VADD: Head Descriptor Loaded for V Chrominance Interrupt Enable

0: No effect

1: Interrupt source is enabled

VDONE: End of List for V Chrominance Interrupt Enable

0: No effect

1: Interrupt source is enabled

VOVR: Overflow for V Chrominance Interrupt Enable

0: No effect

1: Interrupt source is enabled

39.7.86 High-End Overlay DMA Next Register

Name: LCDC_HEONEXT

Address: 0xF0000368

Access: Read/Write

31	30	29	28	27	26	25	24
NEXT							

23	22	21	20	19	18	17	16
NEXT							

15	14	13	12	11	10	9	8
NEXT							

7	6	5	4	3	2	1	0
NEXT							

NEXT: DMA Descriptor Next Address

The transfer descriptor address must be aligned on a 64-bit boundary.

7. Write to the transmit start bit (TSTART) in the Network Control register.

40.7.1.8 Receiving Frames

When a frame is received and the receive circuits are enabled, the GMAC checks the address and, in the following cases, the frame is written to system memory:

- If it matches one of the four Specific Address registers.
- If it matches one of the four Type ID registers.
- If it matches the hash address function.
- If it is a broadcast address (0xFFFFFFFFFFFF) and broadcasts are allowed.
- If the GMAC is configured to “copy all frames”.

The register receive buffer queue pointer points to the next entry in the receive buffer descriptor list and the GMAC uses this as the address in system memory to write the frame to.

Once the frame has been completely and successfully received and written to system memory, the GMAC then updates the receive buffer descriptor entry (see Table 40-4 “Receive Buffer Descriptor Entry”) with the reason for the address match and marks the area as being owned by software. Once this is complete, a receive complete interrupt is set. Software is then responsible for copying the data to the application area and releasing the buffer (by writing the ownership bit back to 0).

If the GMAC is unable to write the data at a rate to match the incoming frame, then a receive overrun interrupt is set. If there is no receive buffer available, i.e., the next buffer is still owned by software, a receive buffer not available interrupt is set. If the frame is not successfully received, a statistics register is incremented and the frame is discarded without informing software.

42.7.21 EHCI: REG08 - HSIC Enable/Disable

Name: UHPHS_INSNREG08

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
–					HSIC_EN	–	

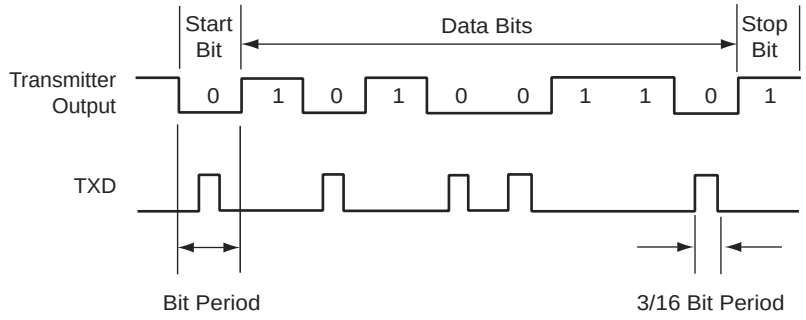
HSIC_EN: HSIC Enable/Disable

This register has R/W access to the host driver and gives control to the host driver to enable/disable the HSIC interface of PORT C.

0: PORT C is in the HSIC Disable state (see *High-Speed Inter-Chip USB Electrical Specification*, Version 1.0, Section 3.1.2).). HSIC is in the Disabled state after a power-on reset.

1: PORT C is in the HSIC Enable state (see High-Speed Inter-Chip USB Electrical Specification, Version 1.0, Section 3.1.2).

Figure 47-34: IrDA Modulation



47.7.5.2 IrDA Baud Rate

Table 47-13 gives some examples of CD values, baud rate error and pulse duration. Note that the requirement on the maximum acceptable error of ±1.87% must be met.

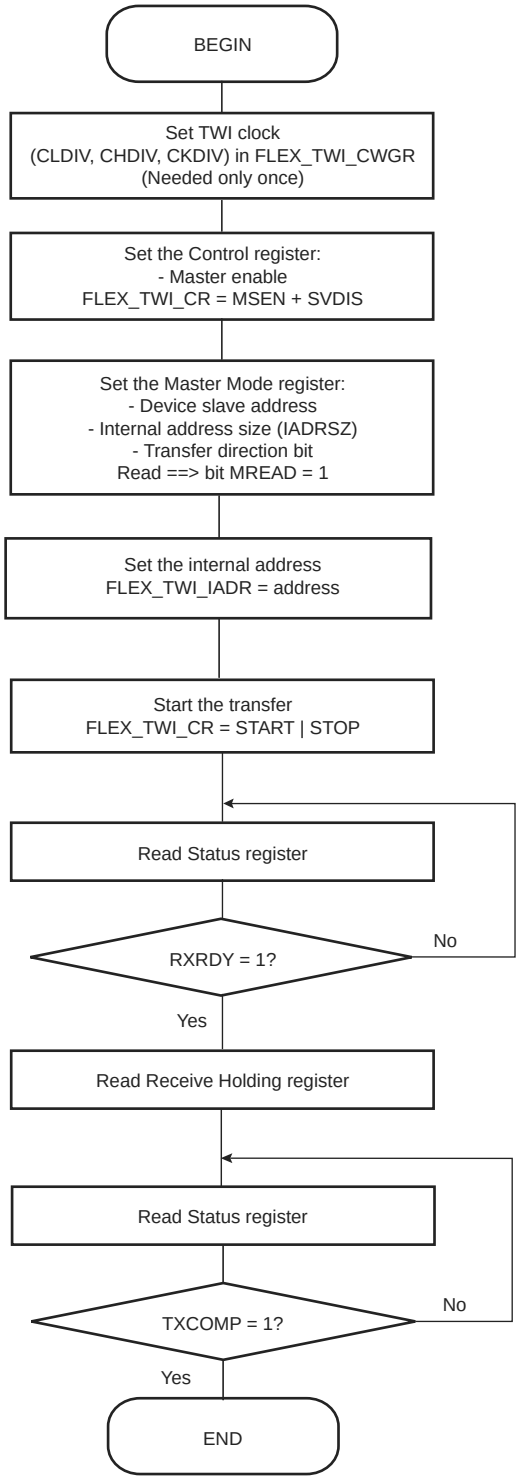
Table 47-13: IrDA Baud Rate Error

Peripheral Clock	Baud Rate (bit/s)	CD	Baud Rate Error	Pulse Time (µs)
3,686,400	115,200	2	0.00%	1.63
20,000,000	115,200	11	1.38%	1.63
32,768,000	115,200	18	1.25%	1.63
40,000,000	115,200	22	1.38%	1.63
3,686,400	57,600	4	0.00%	3.26
20,000,000	57,600	22	1.38%	3.26
32,768,000	57,600	36	1.25%	3.26
40,000,000	57,600	43	0.93%	3.26
3,686,400	38,400	6	0.00%	4.88
20,000,000	38,400	33	1.38%	4.88
32,768,000	38,400	53	0.63%	4.88
40,000,000	38,400	65	0.16%	4.88
3,686,400	19,200	12	0.00%	9.77
20,000,000	19,200	65	0.16%	9.77
32,768,000	19,200	107	0.31%	9.77
40,000,000	19,200	130	0.16%	9.77
3,686,400	9,600	24	0.00%	19.53
20,000,000	9,600	130	0.16%	19.53
32,768,000	9,600	213	0.16%	19.53
40,000,000	9,600	260	0.16%	19.53
3,686,400	2,400	96	0.00%	78.13
20,000,000	2,400	521	0.03%	78.13
32,768,000	2,400	853	0.04%	78.13

47.7.5.3 IrDA Demodulator

The demodulator is based on the IrDA Receive filter comprised of an 8-bit down counter which is loaded with the value programmed in FLEX_US_IF. When a falling edge is detected on the RXD pin, the Filter Counter starts counting down at the peripheral clock speed. If a rising edge is detected on the RXD pin, the counter stops and is reloaded with FLEX_US_IF. If no rising edge is detected when the counter reaches 0, the input of the receiver is driven low during one bit time.

Figure 47-103: TWI Read Operation with Single Data Byte and Internal Address



0: Slave address 3 matching is disabled.

1: Slave address 3 matching is enabled.

DATAMEN: Data Matching Enable

0: Data matching on first received data is disabled.

1: Data matching on first received data is enabled.

50.7.4 QSPI Transmit Data Register

Name: QSPI_TDR

Address: 0xF002000C (0), 0xF002400C (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
TD							
7	6	5	4	3	2	1	0
TD							

TD: Transmit Data

Data to be transmitted by the QSPI is stored in this register. Information to be transmitted must be written to the Transmit Data register in a right-justified format.

53.6.27 MCAN Receive FIFO 0 Configuration

Name: MCAN_RXF0C

Address: 0xF80540A0 (0), 0xFC0500A0 (1)

Access: Read/Write

31	30	29	28	27	26	25	24
F0OM	F0WM						
23	22	21	20	19	18	17	16
–	F0S						
15	14	13	12	11	10	9	8
F0SA							
7	6	5	4	3	2	1	0
F0SA						–	–

This register can only be written if the bits CCE and INIT are set in MCAN CC Control Register.

F0SA: Receive FIFO 0 Start Address

Start address of Receive FIFO 0 in Message RAM (32-bit word address, see Figure 53-12).

Write F0SA with the bits [15:2] of the 32-bit address.

F0S: Receive FIFO 0 Size

0: No Receive FIFO 0

1-64: Number of Receive FIFO 0 elements.

>64: Values greater than 64 are interpreted as 64.

The Receive FIFO 0 elements are indexed from 0 to F0S-1.

F0WM: Receive FIFO 0 Watermark

0: Watermark interrupt disabled.

1-64: Level for Receive FIFO 0 watermark interrupt (MCAN_IR.RF0W).

>64: Watermark interrupt disabled.

F0OM: FIFO 0 Operation Mode

FIFO 0 can be operated in Blocking or in Overwrite mode (see Section 53.5.4.2).

0: FIFO 0 Blocking mode.

1: FIFO 0 Overwrite mode.

55.7 Pulse Density Modulation Interface Controller (PDMIC) User Interface

Table 55-4: Register Mapping

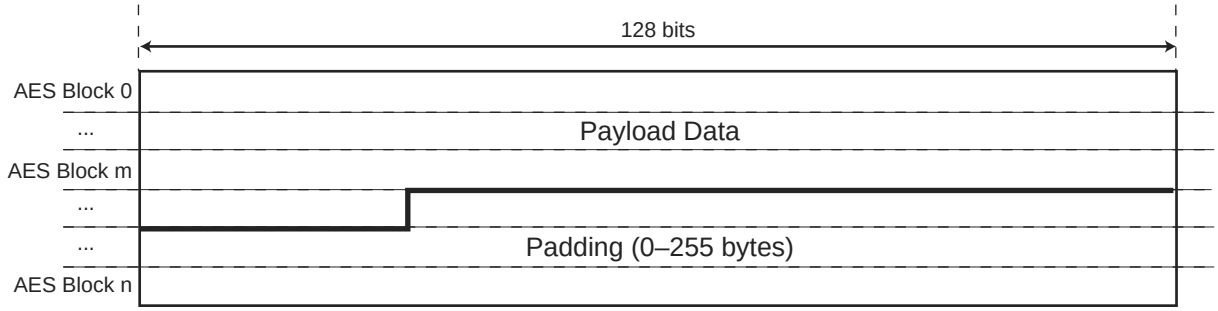
Offset[1]	Register	Name	Access	Reset
0x00	Control Register	PDMIC_CR	Read/Write	0x00000000
0x04	Mode Register	PDMIC_MR	Read/Write	0x00F00000
0x08–0x10	Reserved	–	–	–
0x14	Converted Data Register	PDMIC_CDR	Read-only	0x00000000
0x18	Interrupt Enable Register	PDMIC_IER	Write-only	–
0x1C	Interrupt Disable Register	PDMIC_IDR	Write-only	–
0x20	Interrupt Mask Register	PDMIC_IMR	Read-only	0x00000000
0x24	Interrupt Status Register	PDMIC_ISR	Read-only	0x00000000
0x28–0x54	Reserved	–	–	–
0x58	DSP Configuration Register 0	PDMIC_DSPR0	Read/Write	0x00000000
0x5C	DSP Configuration Register 1	PDMIC_DSPR1	Read/Write	0x00000001
0x60–0xE0	Reserved	–	–	–
0xE4	Write Protection Mode Register	PDMIC_WPMR	Read/Write	0x00000000
0xE8	Write Protection Status Register	PDMIC_WPSR	Read-only	0x00000000
0xEC–0xFC	Reserved	–	–	–

Note 1: If an offset is not listed in the table, it must be considered as “reserved”.

60.4.8.2 SSL Padding

Auto Padding is enabled by writing a ‘1’ to AES_EMR.APEN and SSL padding mode is selected by writing a ‘1’ to AES_EMR.APM.

Figure 60-10: SSL Padding



The padding length is configured in AES_EMR.PADLEN.

AES_BCNT.BCNT defines the length, in bytes, of the message to process. It must be configured before writing the first data in AES_IDATARx and the remaining bytes to process can be read at anytime (BCNT value is decremented after each AES_IDATARx access).

AES_BCNT.BCNT and AES_EMR.PADLEN must be configured so that the length of the message plus the length of the padding section is a multiple of the AES block size (128 bits).

To process a complete SSL message, the sequence is as follows:

1. Set AES_MR.OPMOD to either CBC or CTR mode.
2. Set AES_EMR.APEN to ‘1’, AES_EMR.APM to ‘1’, AES_EMR.PADLEN to the desired padding length in bytes.
3. Set AES_BCNT.BCNT with the whole message length, without padding, in bytes.
4. Set the AES Key Register.
5. Set AES_IVRx.IV if needed.
6. Fill AES_IDATARx.IDATA with the message to process according to the SMOD configuration used. On the last data block write only what is necessary (e.g., write only AES_IDATAR0 if last block size is ≤ 32 bits).
7. Wait for the DATRDY flag to be raised, meaning auto-padding completion and last block processing.

60.4.8.3 Flags

AES_ISR.EOPAD rises as soon as the automatic padding phase is over, meaning that all the extra padding blocks have been processed. Reading AES_ISR clears this flag.

AES_ISR.PLENERR indicates an error in the frame configuration, meaning that the whole message length including padding does not respect the standard selected. AES_ISR.PLENERR rises at the end of the frame in case of wrong message length and is cleared reading AES_ ISR.

In IPSEC/SSL standard message length including padding must be a multiple of the AES block size when CBC mode is used and multiple of 32-bit if CTR mode is used.

60.4.9 Secure Protocol Layers Improved Performances

Secure protocol layers such as IPSec require encryption and authentication. For IPSec, the authentication is based on HMAC, thus SHA is required. To optimize performance, the AES embeds a mode of operation that enables the SHA module to process the input or output data of the AES module. If this mode is enabled, write access is required only into AES_IDATARx registers, since SHA input data registers are automatically written by AES without software intervention. When the DMA is configured to transfer a buffer of data (input frame), only one transfer descriptor is required for both authentication and encryption/decryption processes and only one buffer is transferred through the system bus (reducing the load of the system bus).

Improved performance for secure protocol layers requires AES_EMR.PLIPEN to be set.

62. Triple Data Encryption Standard (TDES)

62.1 Description

The Triple Data Encryption Standard (TDES) is compliant with the American *FIPS (Federal Information Processing Standard) Publication 46-3* specification.

The TDES supports the four different confidentiality modes of operation (ECB, CBC, OFB and CFB), specified in the *FIPS (Federal Information Processing Standard) Publication 81* and is compatible with the Peripheral Data Controller channels for all of these modes, minimizing processor intervention for large buffer transfers.

The TDES key is loaded by the software. The software can write up to three 64-bit keys, each stored in two 32-bit write-only registers, i.e., Key x Word Registers TDES_KEYxWR0 and TDES_KEYxWR1.

The input data (and initialization vector for some modes) are stored in two corresponding 32-bit write-only registers:

Input Data Registers TDES_IDATAR0 and TDES_IDATAR1

Initialization Vector Registers TDES_IVR0 and TDES_IVR1

As soon as the initialization vector, the input data and the keys are configured, the encryption/decryption process may be started. Then the encrypted/decrypted data is ready to be read out on the two 32-bit Output Data registers (TDES_ODATARx) or through the DMA channels.

62.2 Embedded Characteristics

- Supports Single Data Encryption Standard (DES) and Triple Data Encryption Standard (TDES)
- Compliant with *FIPS Publication 46-3, Data Encryption Standard (DES)*
- 64-bit Cryptographic Key for TDES
- Two-key or Three-key Algorithms for TDES
- 18-clock Cycles Encryption/Decryption Processing Time for DES
- 50-clock Cycles Encryption/Decryption Processing Time for TDES
- Supports eXtended Tiny Encryption Algorithm (XTEA)
- 128-bit key for XTEA and Programmable Round Number up to 64
- Supports the Four Standard Modes of Operation specified in the *FIPS Publication 81, DES Modes of Operation*
  - Electronic Code Book (ECB)
  - Cipher Block Chaining (CBC)
  - Cipher Feedback (CFB)
  - Output Feedback (OFB)
- 8-, 16-, 32- and 64-bit Data Sizes Possible in CFB Mode
- Last Output Data Mode Allowing Optimized Message (Data) Authentication Code (MAC) Generation
- Connection to DMA Optimizes Data Transfers for all Operating Modes

62.3 Product Dependencies

62.3.1 Power Management

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

62.3.2 Interrupt Sources

The TDES interface has an interrupt line connected to the Interrupt Controller. In order to handle interrupts, the Interrupt Controller must be programmed before configuring the TDES.

Table 62-1: Peripheral IDs

Instance	ID
TDES	11

Table 66-55: FLEXCOM2 in SPI Mode IOSET1 Timings

Symbol	Power Supply / Parameter	1.8V Min	1.8V Max	3.3V Min	3.3V Max	Unit
Master Mode						
SPI_0	MISO Setup time before SPCK rises	15.3	–	13.4	–	ns
SPI_1	MISO Hold time after SPCK rises	0	–	0	–	ns
SPI_2	SPCK rising to MOSI	0	2.5	0	3	ns
SPI_3	MISO Setup time before SPCK falls	15.6	–	13.7	–	ns
SPI_4	MISO Hold time after SPCK falls	0	–	0	–	ns
SPI_5	SPCK falling to MOSI	0	2.6	0	3.1	ns
Slave Mode						
SPI_6	SPCK falling to MISO	11.7	13.5	9.4	11.7	ns
SPI_7	MOSI Setup time before SPCK rises	2	–	1.6	–	ns
SPI_8	MOSI Hold time after SPCK rises	0.5	–	0.5	–	ns
SPI_9	SPCK rising to MISO	11.7	13.5	9.4	11.5	ns
SPI_{10}	MOSI Setup time before SPCK falls	2	–	1.6	–	ns
SPI_{11}	MOSI Hold time after SPCK falls	0.5	–	0.5	–	ns
SPI_{12}	NPCS0 setup to SPCK rising	4	–	3.7	–	ns
SPI_{13}	NPCS0 hold after SPCK falling	0.6	–	0.6	–	ns
SPI_{14}	NPCS0 setup to SPCK falling	3.9	–	3.7	–	ns
SPI_{15}	NPCS0 hold after SPCK rising	0.4	–	0.3	–	ns
SPI_{16}	NPCS0 falling to MISO valid	16.8	–	13.6	–	ns

crosstalk. Based on EMI testing experience, the minimum suggested spacing to clock signals is 50 mil.
- Use a minimum of 20 mil spacing between the differential signal pairs and other signal traces for optimal signal quality. This helps to prevent crosstalk.
- Route all traces over continuous planes (VCC or GND) with no interruptions.
- Avoid crossing over anti-etch if at all possible. Crossing over anti-etch (split planes) increases inductance and radiation levels by forcing a greater loop area.

68.12.3 DDR Layout and Design Considerations

Refer to the document “SAMA5D2 Layout Recommendations”, document no. 44041.

68.12.4 e.MMC routing

Refer to the Micron Technical Note TN-FC-35: e•MMC PCB Design Guide. This document is intended as guide for PCB designers using Micron e•MMC devices and discusses the primary issues affecting design and layout.

68.12.5 USB Trace Routing Guidelines

- Maintain parallelism between USB differential signals with the trace spacing needed to achieve 90-ohm differential impedance. Deviations will normally occur due to package breakout and routing to connector pins. Just ensure the amount and lengths of the deviations are kept to the minimum possible.
- Use an impedance calculator to determine the trace width and spacing required for the specific board stack-up being used. Example: Layer Stacking, 7.5-mil traces with 7.5-mil spacing results in approximately 90-ohm differential trace impedance.
- Minimize the length of high-speed clock and periodic signal traces that run parallel to high-speed USB signal lines, to minimize crosstalk. Based on EMI testing experience, the minimum suggested spacing to clock signals is 50 mils.
- Based on simulation data, use 20-mil minimum spacing between high-speed USB signal pairs and other signal traces for optimal signal quality. This helps to prevent crosstalk.

Table 68-14: USB Trace Routing Guidelines

Parameter	Trace Routing
Signal length allowance for the SAMA5D2	14.0 inches Valid for a damping value of the PCB trace of 0.11 dB/ inch @ 0.4 GHz (common value for FR-4 based material)
Differential impedance	90 ohms +/-15%
Single-ended impedance	45 ohms +/-10%
Trace width (W)	5 mils (microstrip routing)
Spacing between differential pairs (intra-pair)	6 mils (microstrip routing)
Spacing between pairs (inter-pair)	Min. 20 mils
Spacing between differential pairs and high-speed periodic signals	Min. 50 mils
Spacing between differential pairs and low-speed non-periodic signals	Min. 20 mils
Length matching between differential pairs (intra-pair)	150 mils
Reference plane	GND referenced preferred
Spacing from edge of plane	Min. 40 mils
Vias usage	Try to minimize the number of vias

68.12.6 QSPI Pull-up Resistors

The ROM code **removes** the internal pull-up resistors when it configures PIO controller to mux the QSPI controller I/O lines. Therefore the probing step may fail if the Quad I/O mode of the memory has not been enabled yet and if this memory does not embed internal pull-up resistor on #HOLD or #RESET pin.