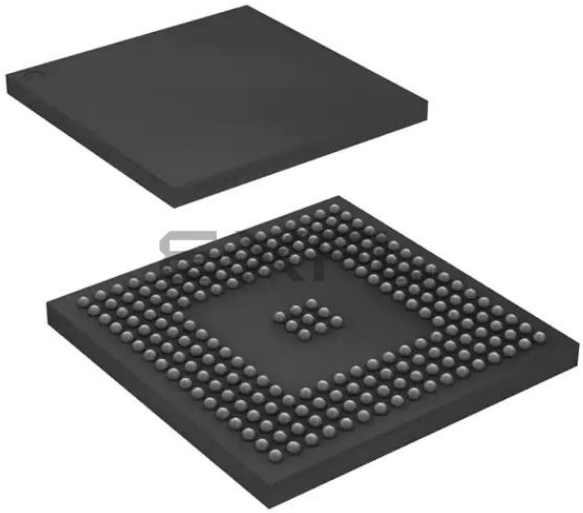


Welcome to E-XFL.COM

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.

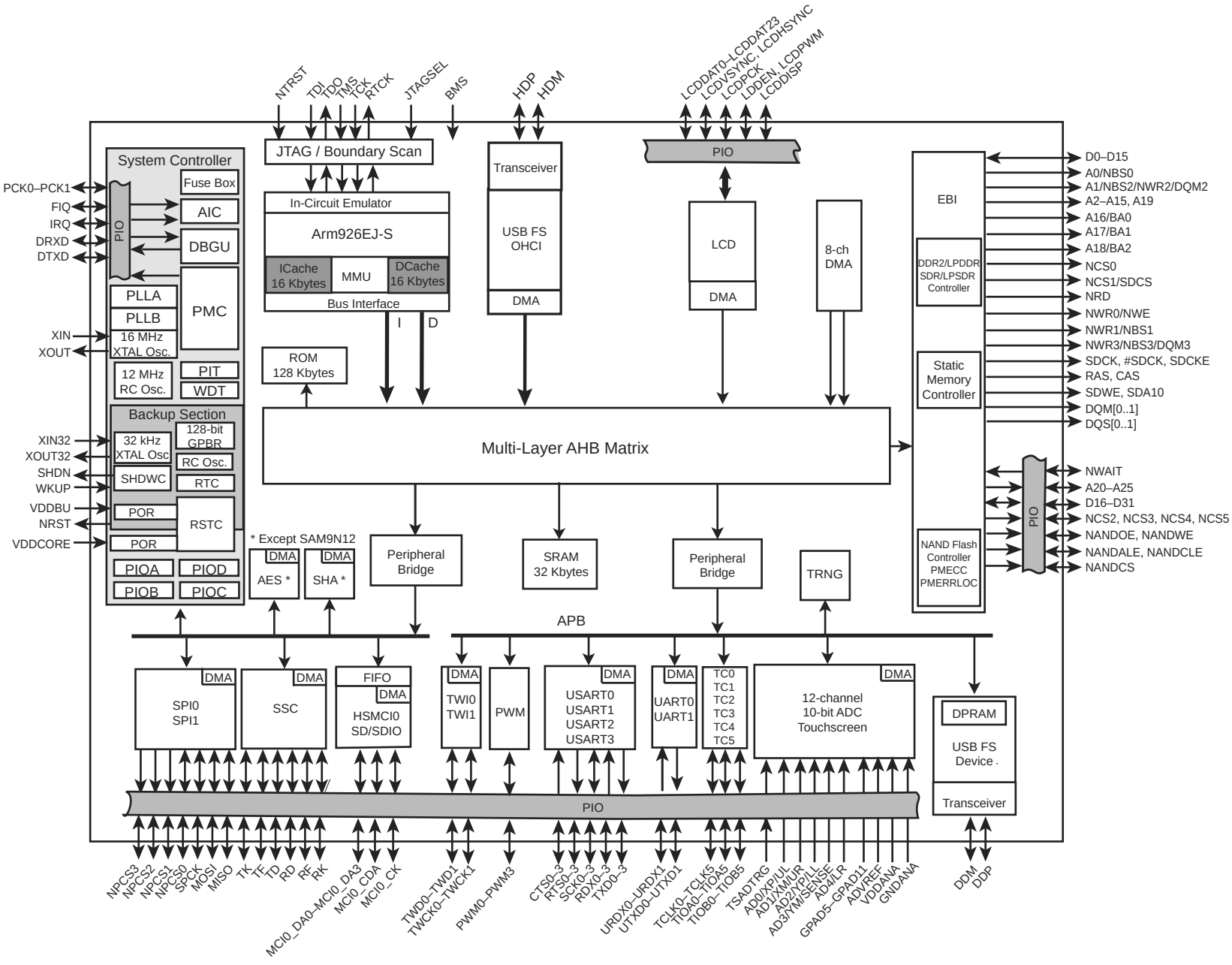
Applications of Embedded - Microprocessors

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

Details	
Product Status	Active
Core Processor	ARM926EJ-S
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	400MHz
Co-Processors/DSP	-
RAM Controllers	LPDDR, LPDDR2, DDR2, SDR, SRAM
Graphics Acceleration	No
Display & Interface Controllers	LCD, Touchscreen
Ethernet	-
SATA	-
USB	USB 2.0 (2)
Voltage - I/O	1.8V, 3.3V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	-
Package / Case	217-LFBGA
Supplier Device Package	217-LFBGA (15x15)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at91sam9cn12b-cur

1. Block Diagram

Figure 1-1: SAM9N12/CN11/CN12 Block Diagram



10.9.2 AIC Source Vector Register

Name:AIC_SVR0..AIC_SVR31

Address:0xFFFFF080

Access: Read/Write

Reset:0x0

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

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

VECTOR: Source Vector

The user may store in these registers the addresses of the corresponding handler for each interrupt source.

10.9.8 AIC Core Interrupt Status Register

Name: AIC_CISR

Address:0xFFFFF114

Access: Read-only

Reset:0x0

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
–	–	–	–	–	–	NIRQ	NFIQ

NFIQ: NFIQ Status

0: nFIQ line is deactivated.

1: nFIQ line is active.

NIRQ: NIRQ Status

0: nIRQ line is deactivated.

1: nIRQ line is active.

10.9.15 AIC Debug Control Register

Name:AIC_DCR

Address:0xFFFFF138

Access:Read/Write

Reset:0x0

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
–	–	–	–	–	–	GMSK	PROT

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

PROT: Protection Mode

0: The Protection Mode is disabled.

1: The Protection Mode is enabled.

GMSK: General Interrupt Mask

0: The nIRQ and nFIQ lines are normally controlled by the AIC.

1: The nIRQ and nFIQ lines are tied to their inactive state.

29.12.2 Frozen Mode

When the external device asserts the NWAIT signal (active low), and after internal synchronization of this signal, the SMC state is frozen, i.e., SMC internal counters are frozen, and all control signals remain unchanged. When the resynchronized NWAIT signal is deasserted, the SMC completes the access, resuming the access from the point where it was stopped. See Figure 29-26. This mode must be selected when the external device uses the NWAIT signal to delay the access and to freeze the SMC.

The assertion of the NWAIT signal outside the expected period is ignored as illustrated in Figure 29-27.

Figure 29-26: Write Access with NWAIT Assertion in Frozen Mode (EXNW_MODE = 10)

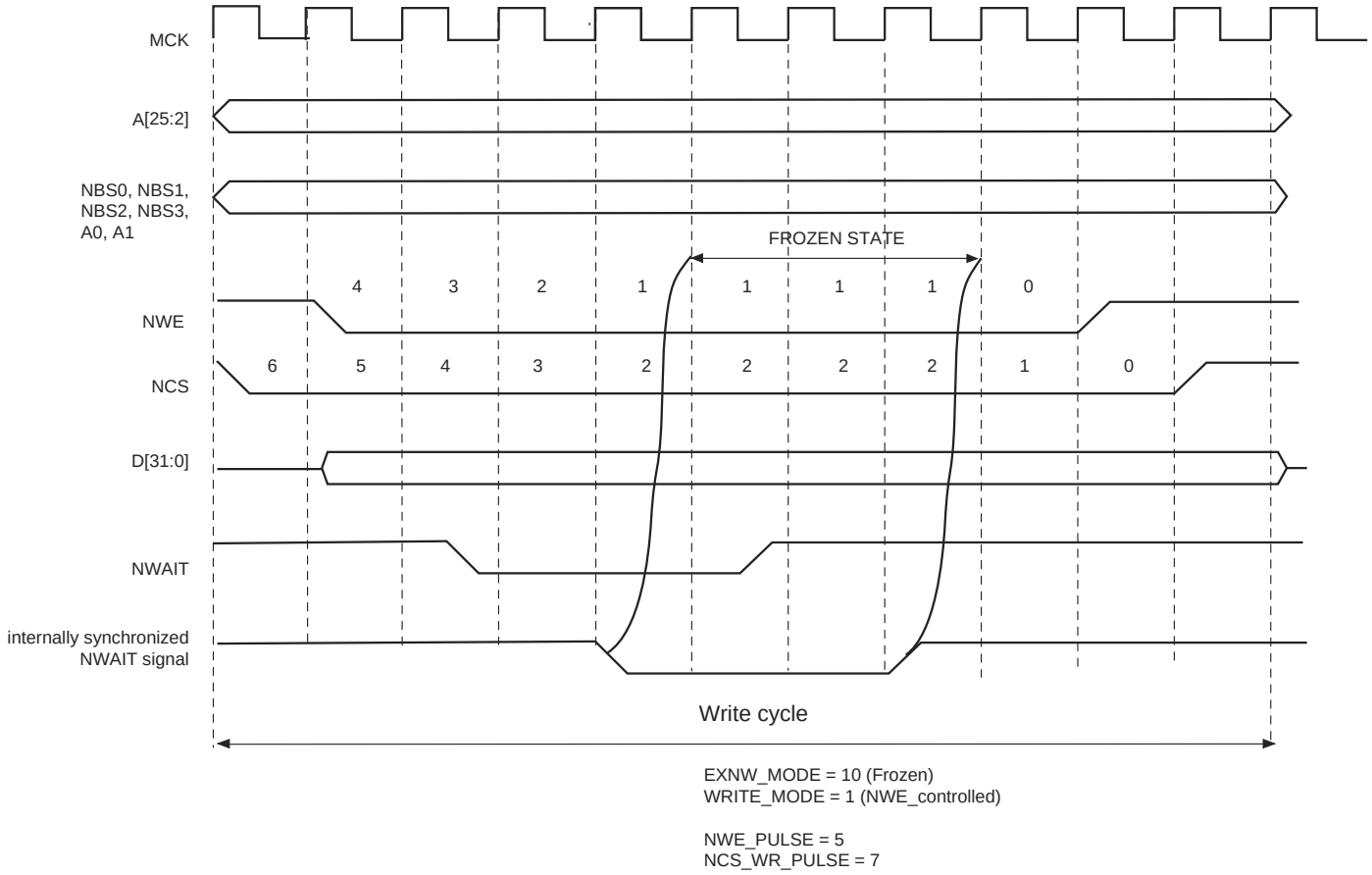
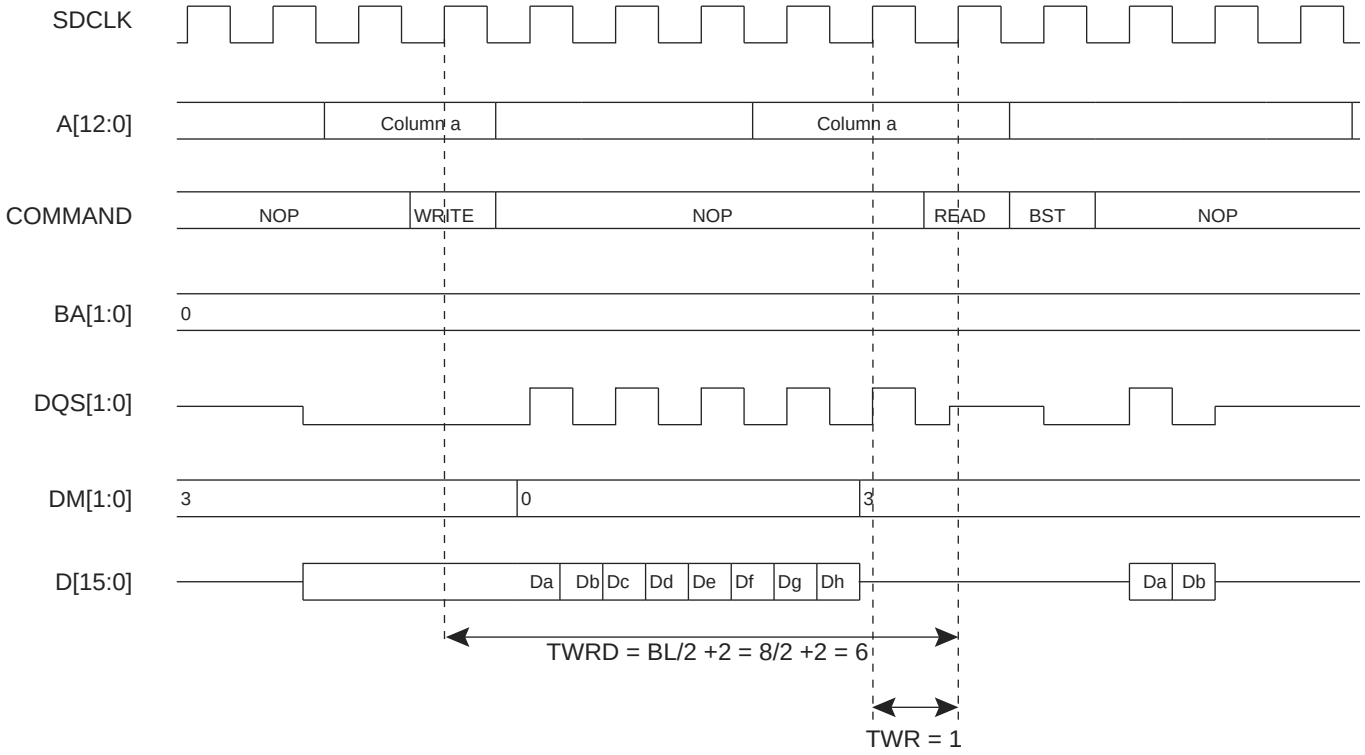


Figure 30-8: Write Command Followed By a Read Command Without Burst Write Interrupt, Low-power DDR1-SDRAM Device



In the case of a single write access, write operation should be interrupted by a read access but DM must be input 1 cycle prior to the read command to avoid writing invalid data. (See Figure 30-9.)

Figure 30-9: Single Write Access Followed By A Read Access Low-power DDR1-SDRAM Devices

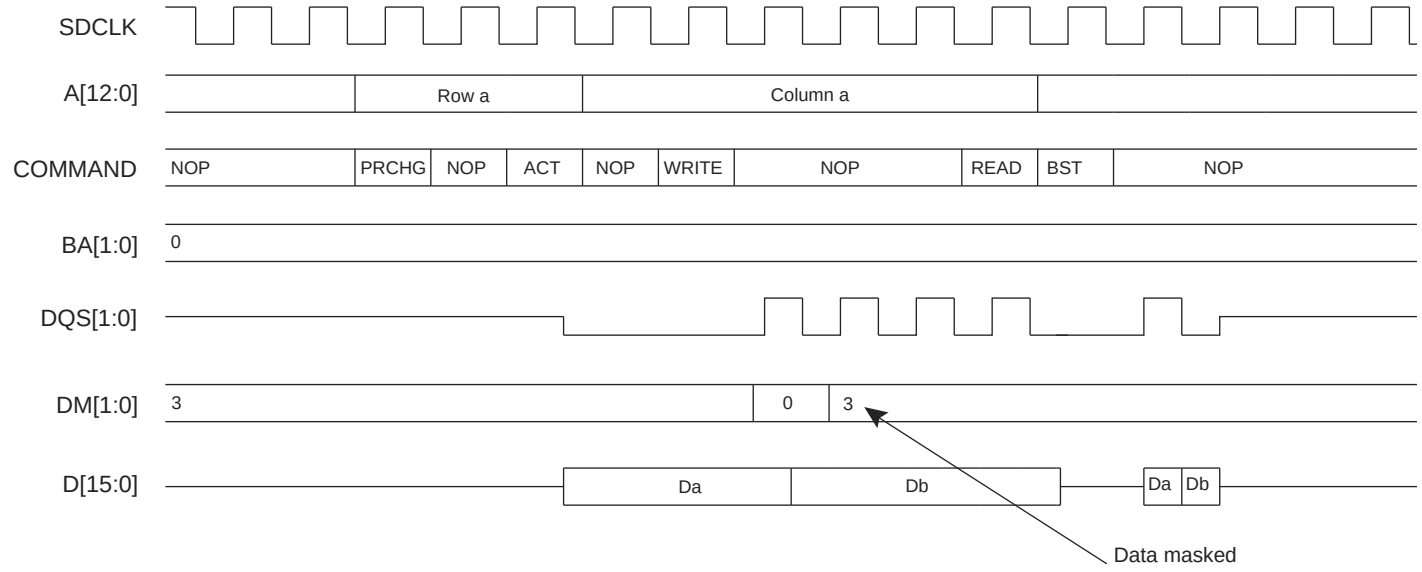


Table 30-3: Linear Mapping for SDRAM Configuration: 8K Rows, 512/1024/2048/4096 Columns

CPU Address Line																											
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**
			Bk[1:0]		Row[12:0]													Column[8:0]									M0
		Bk[1:0]			Row[12:0]													Column[9:0]									M0
	Bk[1:0]				Row[12:0]													Column[10:0]									M0
Bk[1:0]					Row[12:0]													Column[11:0]									M0

Table 30-4: Linear Mapping for SDRAM Configuration: 16K Rows, 512/1024/2048 Columns

CPU Address Line																											
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**
		Bk[1:0]			Row[13:0]													Column[8:0]									M0
	Bk[1:0]				Row[13:0]													Column[9:0]									M0
Bk[1:0]					Row[13:0]													Column[10:0]									M0

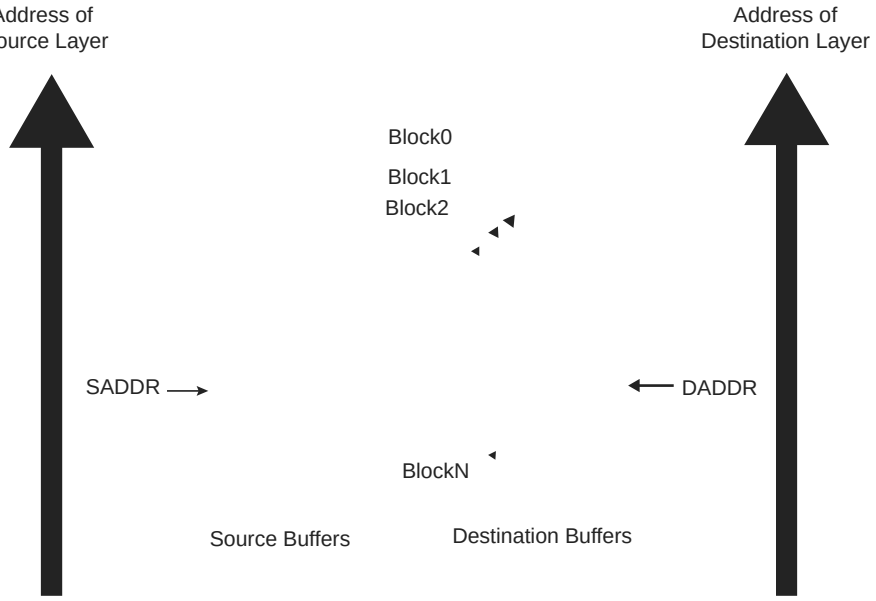
Table 30-5: Interleaved Mapping for SDRAM Configuration, 2K Rows, 512/1024/2048/4096 Columns

CPU Address Line																											
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**
					Row[10:0]											Bk[1:0]		Column[8:0]									M0
					Row[10:0]										Bk[1:0]			Column[9:0]									M0
					Row[10:0]									Bk[1:0]				Column[10:0]									M0
					Row[10:0]								Bk[1:0]					Column[11:0]									M0

Table 30-6: Interleaved Mapping for SDRAM Configuration: 4K Rows, 512/1024/2048/4096 Columns

CPU Address Line																											
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**
				Row[11:0]												Bk[1:0]		Column[8:0]									M0
			Row[11:0]												Bk[1:0]			Column[9:0]									M0
		Row[11:0]												Bk[1:0]				Column[10:0]									M0
	Row[11:0]												Bk[1:0]					Column[11:0]									M0

Figure 31-10: Multi-buffer DMAC Transfer with Source and Destination Address Auto-reloaded



31.8.16 DMAC Channel x [x = 0..7] Control A Register

Name: DMAC_CTRLAx [x = 0..7]

Address:0xFFFFEC48 [0], 0xFFFFEC70 [1], 0xFFFFEC98 [2], 0xFFFFECC0 [3], 0xFFFFECE8 [4], 0xFFFFED10 [5], 0xFFFFED38 [6], 0xFFFFED60 [7]

Access: Read/Write

31	30	29	28	27	26	25	24
DONE	–	DST_WIDTH		–	–	SRC_WIDTH	

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

15	14	13	12	11	10	9	8
BTSIZE							

7	6	5	4	3	2	1	0
BTSIZE							

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

BTSIZE: Buffer Transfer Size

The transfer size relates to the number of transfers to be performed, that is, for writes it refers to the number of source width transfers to perform when DMAC is flow controller. For reads, BTSIZE refers to the number of transfers completed on the Source Interface. When this field is cleared, the DMAC module is automatically disabled when the relevant channel is enabled.

SCSIZE: Source Chunk Transfer Size

Value	**Name**	**Description**
000	CHK_1	1 data transferred
001	CHK_4	4 data transferred
010	CHK_8	8 data transferred
011	CHK_16	16 data transferred

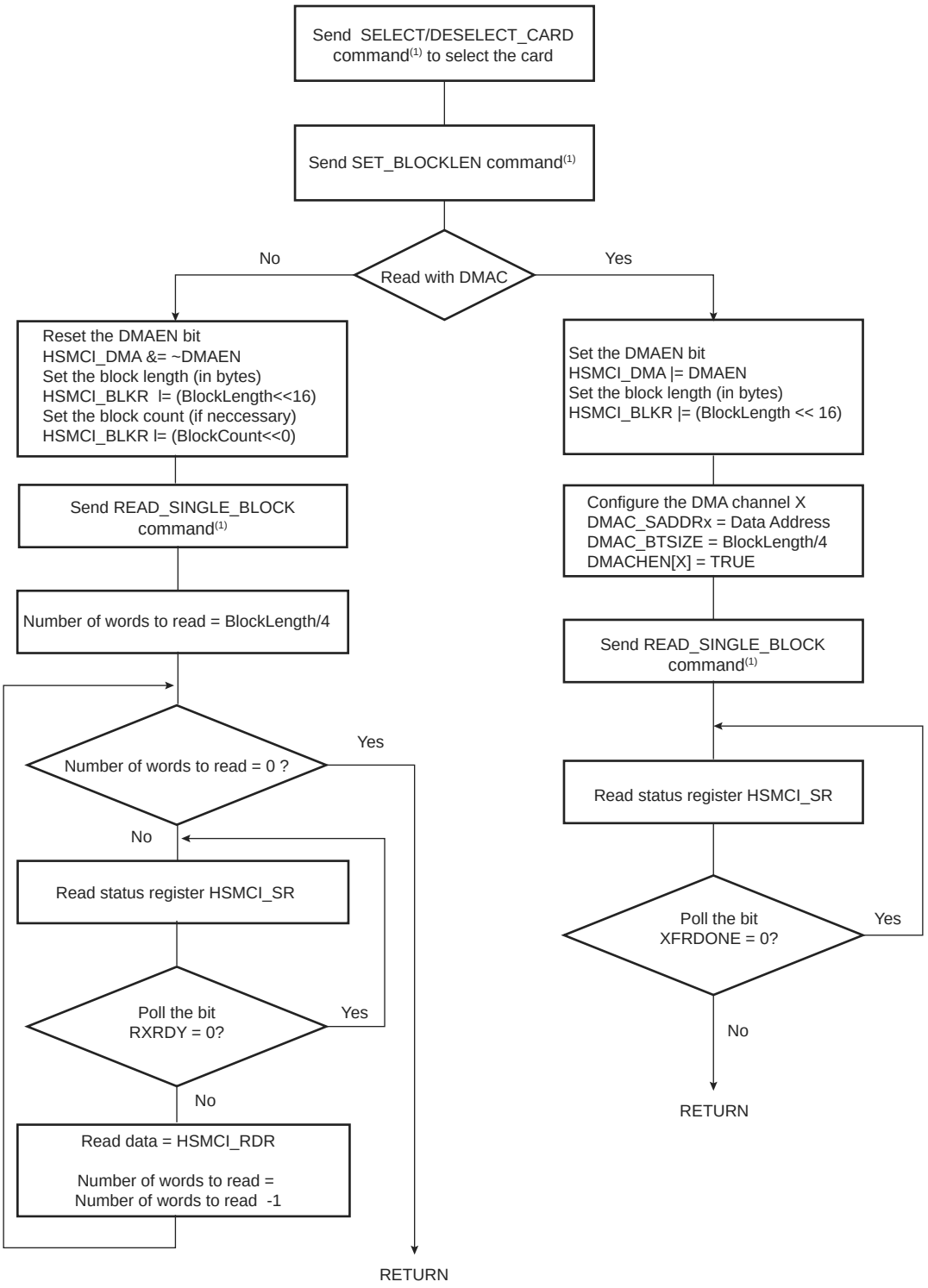
DCSIZE: Destination Chunk Transfer Size

Value	**Name**	**Description**
000	CHK_1	1 data transferred
001	CHK_4	4 data transferred
010	CHK_8	8 data transferred
011	CHK_16	16 data transferred

SRC_WIDTH: Transfer Width for the Source

Value	**Name**	**Description**
00	BYTE	The transfer size is set to 8-bit width
01	HALF_WORD	The transfer size is set to 16-bit width
1X	WORD	The transfer size is set to 32-bit width

Figure 34-8: Read Functional Flow Diagram



Note 1: It is assumed that this command has been correctly sent (see Figure 34-7).

34.14.17 HSMCI Configuration Register

Name: HSMCI_CFG

Address:0xF0008054

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).

38.8.5 TWI Clock Waveform Generator Register

Name: TWI_CWGR

Address:0xF8010010 (0), 0xF8014010 (1)

Access: Read/Write

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

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

15	14	13	12	11	10	9	8
CHDIV							

7	6	5	4	3	2	1	0
CLDIV							

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

TWI_CWGR is only used in Master mode.

CLDIV: Clock Low Divider

The TWCK low period is defined as follows: $t_{low} = ((CLDIV \times 2^{CKDIV}) + 4 \times t_{peripheral\ clock}$

CHDIV: Clock High Divider

The TWCK high period is defined as follows: $t_{high} = ((CHDIV \times 2^{CKDIV}) + 4 \times t_{peripheral\ clock}$

CKDIV: Clock Divider

The CKDIV field is used to increase both TWCK high and low periods.

39.7.10 USART Interrupt Disable Register (LIN_MODE)

Name:US_IDR (LIN_MODE)

Address:0xF801C00C (0), 0xF802000C (1), 0xF802400C (2), 0xF802800C (3)

Access: Write-only

31	30	29	28	27	26	25	24
–	–	LINSNRE	LINCE	LINIPE	LINISFE	LINBE	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
LINTC	LINID	LINBK	–	–	–	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	–	–	–	TXRDY	RXRDY

This configuration is relevant only if USART_MODE = 0xA or 0xB in the USART Mode Register.

The following configuration values are valid for all listed bit names of this register:

0: No effect

1: Disables the corresponding interrupt.

RXRDY: RXRDY Interrupt Disable

TXRDY: TXRDY Interrupt Disable

OVRE: Overrun Error Interrupt Disable

FRAME: Framing Error Interrupt Disable

PARE: Parity Error Interrupt Disable

TIMEOUT: Time-out Interrupt Disable

TXEMPTY: TXEMPTY Interrupt Disable

LINBK: LIN Break Sent or LIN Break Received Interrupt Disable

LINID: LIN Identifier Sent or LIN Identifier Received Interrupt Disable

LINTC: LIN Transfer Completed Interrupt Disable

LINBE: LIN Bus Error Interrupt Disable

LINISFE: LIN Inconsistent Synch Field Error Interrupt Disable

LINIPE: LIN Identifier Parity Interrupt Disable

LINCE: LIN Checksum Error Interrupt Disable

LINSNRE: LIN Slave Not Responding Error Interrupt Disable

39.7.17 USART Receive Holding Register

Name:US_RHR

Address:0xF801C018 (0), 0xF8020018 (1), 0xF8024018 (2), 0xF8028018 (3)

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.

41.7.16 ADC Analog Control Register

Name:ADC_ACR

Address:0xF804C094

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
–	–	–	–	–	–	PENDETSENS	

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

PENDETSENS: Pen Detection Sensitivity

Modifies the pen detection input pull-up resistor value. See the section ‘Electrical Characteristics’ for further details.

PENDET: Pen Contact Detection Enable

0: Pen contact detection disabled.

1: Pen contact detection enabled.

When PENDET = 1, XPOS, YPOS, Z1, Z2 values of ADC_XPOSR, ADC_YPOSR, ADC_PRESSR are automatically cleared when PENS = 0 in ADC_ISR.

NOTSDMA: No TouchScreen DMA

0: XPOS, YPOS, Z1, Z2 are transmitted in ADC_LCDR.

1: XPOS, YPOS, Z1, Z2 are never transmitted in ADC_LCDR, therefore the buffer does not contains touchscreen values.

PENDBC: Pen Detect Debouncing Period

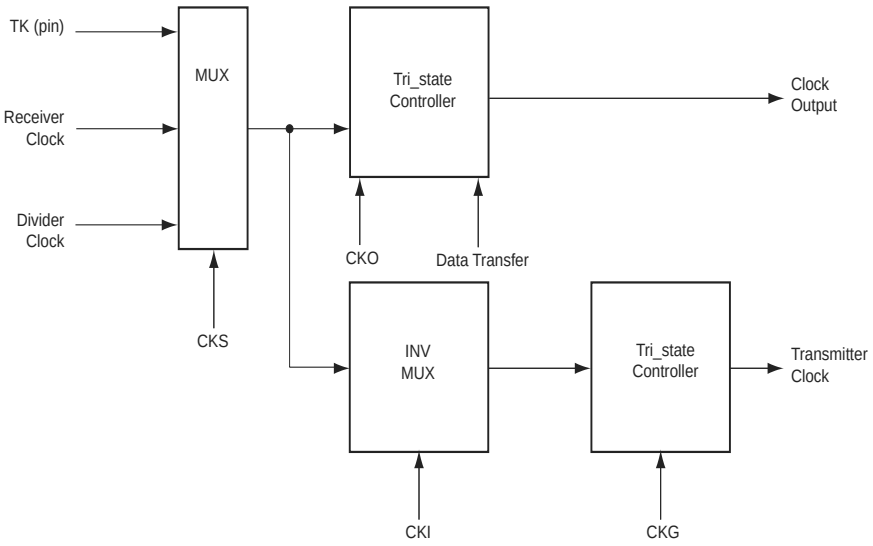
Debouncing period = 2^{PENDBC} ADCCLK periods.

42.8.1.2 Transmitter Clock Management

The transmitter clock is generated from the receiver clock or the divider clock or an external clock scanned on the TK I/O pad. The transmitter clock is selected by the CKS field in the Transmit Clock Mode Register (SSC_TCMR). Transmit Clock can be inverted independently by the CKI bits in the SSC_TCMR.

The transmitter can also drive the TK I/O pad continuously or be limited to the actual data transfer. The clock output is configured by the SSC_TCMR. The Transmit Clock Inversion (CKI) bits have no effect on the clock outputs. Programming the SSC_TCMR to select TK pin (CKS field) and at the same time Continuous Transmit Clock (CKO field) can lead to unpredictable results.

Figure 42-9: Transmitter Clock Management



42.8.1.3 Receiver Clock Management

The receiver clock is generated from the transmitter clock or the divider clock or an external clock scanned on the RK I/O pad. The Receive Clock is selected by the CKS field in SSC_RCMR (Receive Clock Mode Register). Receive Clocks can be inverted independently by the CKI bits in SSC_RCMR.

The receiver can also drive the RK I/O pad continuously or be limited to the actual data transfer. The clock output is configured by the SSC_RCMR. The Receive Clock Inversion (CKI) bits have no effect on the clock outputs. Programming the SSC_RCMR to select RK pin (CKS field) and at the same time Continuous Receive Clock (CKO field) can lead to unpredictable results.

Figure 42-10: Receiver Clock Management

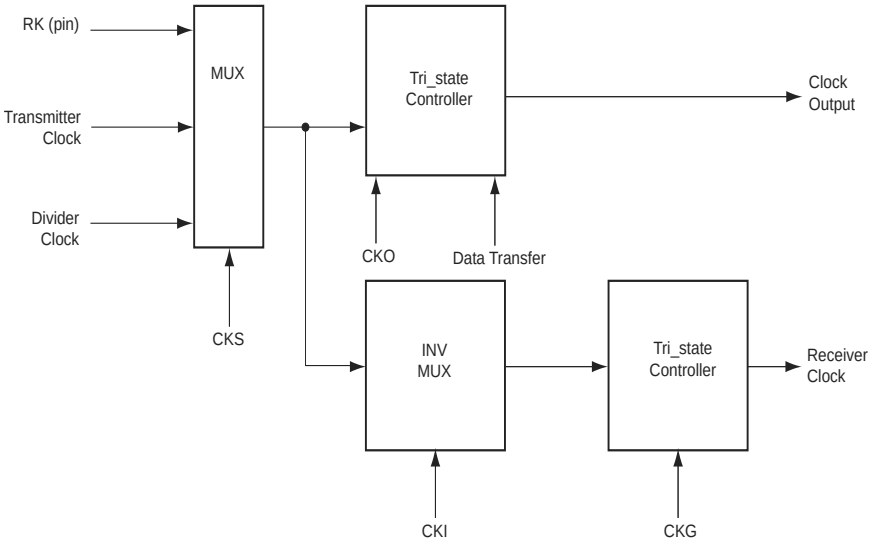
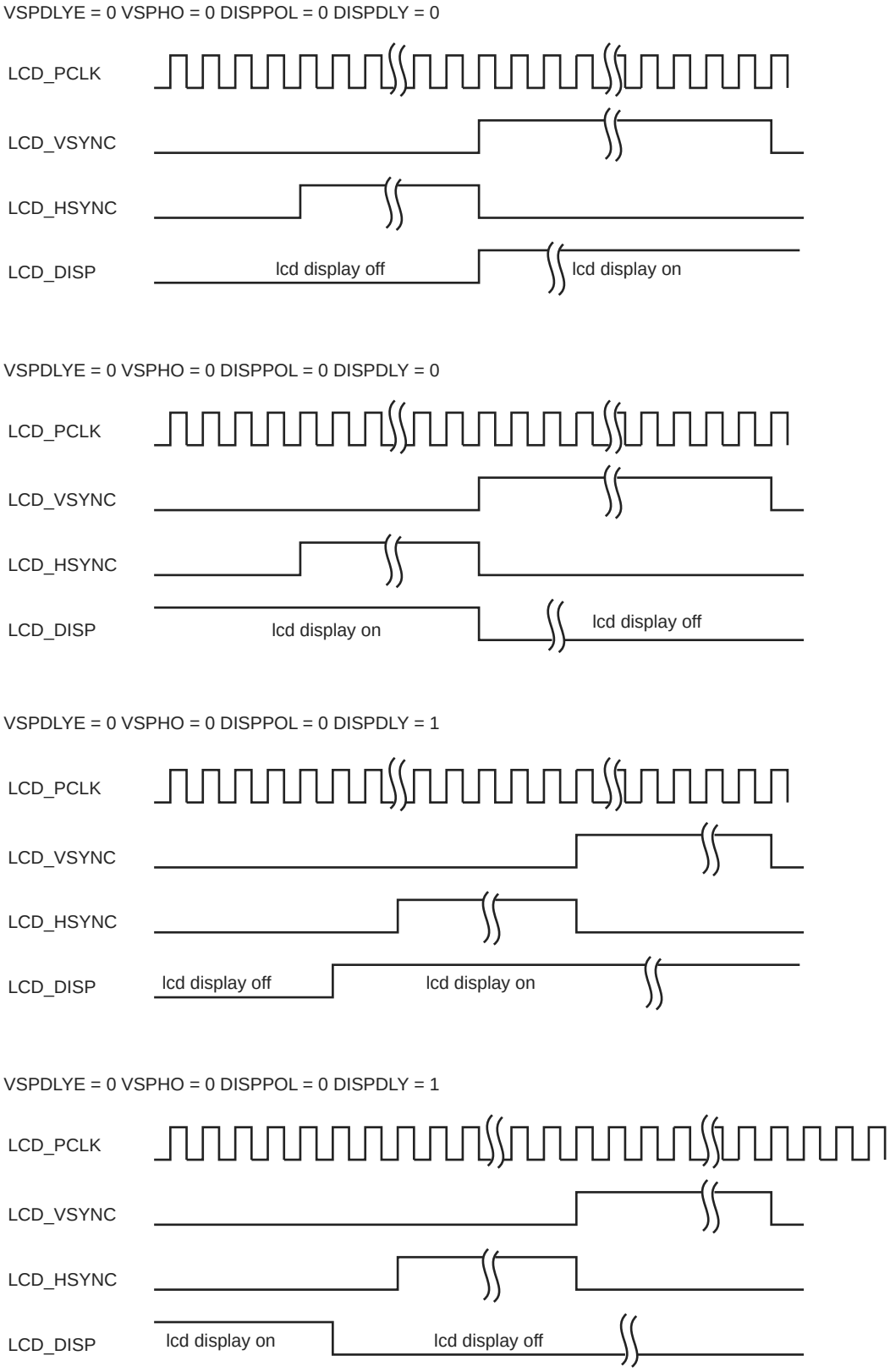


Figure 43-5: DISP Signal Timing Diagram



44.4.4 Start Modes

The SMOD field in the AES_MR allows selection of the encryption (or decryption) Start mode.

44.4.4.1 Manual Mode

The sequence order is as follows:

1. Write the AES_MR with all required fields, including but not limited to SMOD and OPMOD.
2. Write the 128-bit/192-bit/256-bit key in the AES_KEYWRx.
3. Write the initialization vector (or counter) in the AES_IVRx.

Note: The AES_IVRx concern all modes except ECB.

4. Set the bit DATRDY (Data Ready) in the AES Interrupt Enable Register (AES_IER), depending on whether an interrupt is required or not at the end of processing.
5. Write the data to be encrypted/decrypted in the authorized AES_IDATARx (see Table 44-2).
6. Set the START bit in the AES Control Register (AES_CR) to begin the encryption or the decryption process.
7. When processing completes, the DATRDY flag in the AES Interrupt Status Register (AES_ISR) is raised. If an interrupt has been enabled by setting the DATRDY bit in the AES_IER, the interrupt line of the AES is activated.
8. When software reads one of the AES_ODATARx, the DATRDY bit is automatically cleared.

Table 44-2: Authorized Input Data Registers

Operation Mode	Input Data Registers to Write
ECB	All
CBC	All
OFB	All
128-bit CFB	All
64-bit CFB	AES_IDATAR0 and AES_IDATAR1
32-bit CFB	AES_IDATAR0
16-bit CFB	AES_IDATAR0
8-bit CFB	AES_IDATAR0
CTR	All

Note 1: In 64-bit CFB mode, writing to AES_IDATAR2 and AES_IDATAR3 is not allowed and may lead to errors in processing.

2: In 32, 16, and 8-bit CFB modes, writing to AES_IDATAR1, AES_IDATAR2 and AES_IDATAR3 is not allowed and may lead to errors in processing.

44.4.4.2 Auto Mode

The Auto Mode is similar to the manual one, except that in this mode, as soon as the correct number of AES_IDATARx is written, processing is automatically started without any action in the AES_CR.

44.4.4.3 DMA Mode

The DMA Controller can be used in association with the AES to perform an encryption/decryption of a buffer without any action by software during processing.

The SMOD field in the AES_MR must be configured to 0x2 and the DMA must be configured with non-incremental addresses.

The start address of any transfer descriptor must be configured with the address of AES_IDATAR0.

The DMA chunk size configuration depends on the AES mode of operation and is listed in Table 44-3 “DMA Data Transfer Type for the Different Operation Modes”.

Doc. Rev. 11063L	Comments (Continued)
28-Oct-15	Section 43. ”LCD Controller (LCDC)”
	Removed reset values from individual register description sections (reset values are provided in Table 43-31 ”Register Mapping”)
	Table 43-31 ”Register Mapping”:
	- removed reset value from write-only registers LCDC_BASECHER, LCDC_BASECHDR, LCDC_BASEIER, and LCDC_BASEIDR
	- added access Read/Write and reset value 0x00000000 for LCDC_BASECLUT255
	- removed five registers LCDC_ADDRSIZE, LCDC_IPNAME1, LCDC_IPNAME2, LCDC_FEATURES, and LCDC_VERSION (offsets 0x1FEC–0x1FFC now reserved)
	Section 44. ”Advanced Encryption Standard (AES)”
	Section 44.1 ”Description”: in second paragraph, “Peripheral DMA Controller” changed to “DMA Controller”
	Added Section 44.4.1 ”AES Register Endianism”
	Updated Section 44.4.2 ”Operation Modes”
	Section 44.4.5.1 ”Manual and Auto Modes”: updated content under “If AES_MR.LOD = 1”
	Section 44.4.5.2 ”DMA Mode”: updated content under “If AES_MR.LOD = 1”
	Section 44.5.2 ”AES Mode Register”: updated PROCDLY field description
	Section 44.5.6 ”AES Interrupt Status Register”: updated field descriptions
	Section 45. ”Secure Hash Algorithm (SHA)”
	Harmonized register naming throughout
	Updated Section 45.4.4.3 ”DMA Mode”:
	Section 45.4.4.4 ”SHA Register Endianism”: rephrased first part of section; corrected values in the examples; corrected IO register numbering
	Table 45-2 ”Register Mapping”: reserved offset range 0x94-0xFC changed to 0xA0–0xFC
	Section 45.5.6 ”SHA Interrupt Status Register”: updated bit descriptions
	Section 45.5.8 ”SHA Output Data Register x”: updated ODATA field description
	Section 46. ”True Random Number Generator (TRNG)”
	Updated names of referenced test suites in Section 46.1 ”Description” and Section 46.2 ”Embedded Characteristics”
	Section 46.5 ”Functional Description”: updated terminology in text and in Figure 46-2 "TRNG Data Generation Sequence"
	Table 46-2 ”Register Mapping”: defined offset ranges 0x04–0x0C, 0x20–0x4C, and 0x54–0xFC as reserved
	Removed the line “Reset” in Section 46.6.4 ”TRNG Interrupt Mask Register”, Section 46.6.5 ”TRNG Interrupt Status Register”, and Section 46.6.6 ”TRNG Output Data Register” (reset values are provided in Table 46-2 ”Register Mapping”)
	Section 47. ”Electrical Characteristics”
	Table 47-5 ”Processor Clock Waveform Parameters”: added footnote “With DDR2 usage. There are no limitations for LPDDR, SDRAM and mobile SDRAM.”
	Table 47-12 ”32 kHz Crystal Characteristics”: added footnote “R_S is the equivalent series resistance.”
	Table 47-13 ”XIN32 Clock Electrical Characteristics”: updated conditions
	Section 47.12 ”Core Power Supply POR Characteristics”: transferred subsections “Power Sequence Requirements” and “Power-Up Sequence” to Section 4. ”Power Considerations”
	In Section 47.15.1.1 ”Maximum SPI Frequency”, updated description under “Master Write Mode” and “Master Read Mode”
	Section 47.15.2.1 ”Timing Conditions”: deleted sentence “These values may be product dependant and should be confirmed by the specification.”