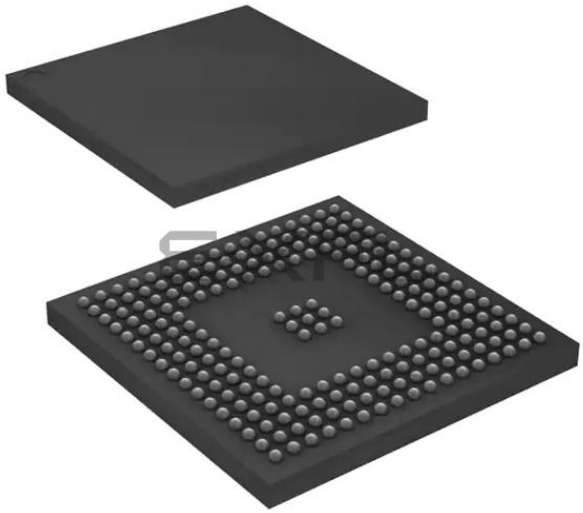


Welcome to E-XFL.COM

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	180MHz
Co-Processors/DSP	-
RAM Controllers	SDRAM, SRAM
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	10/100Mbps
SATA	-
USB	USB 2.0 (2)
Voltage - I/O	1.8V, 2.5V, 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/at91sam9260b-cu-999

7. System Controller

The System Controller is a set of peripherals that allows handling of key elements of the system, such as power, resets, clocks, time, interrupts, watchdog, etc.

The System Controller User Interface also embeds the registers that configure the Matrix and a set of registers for the chip configuration. The chip configuration registers configure EBI chip select assignment and voltage range for external memories

The System Controller's peripherals are all mapped within the highest 16 Kbytes of address space, between addresses 0xFFFF E800 and 0xFFFF FFFF.

However, all the registers of System Controller are mapped on the top of the address space. All the registers of the System Controller can be addressed from a single pointer by using the standard ARM instruction set, as the Load/Store instruction has an indexing mode of ±4 Kbytes.

Figure 7-1 on page 24 shows the System Controller block diagram.

Figure 6-1 on page 19 shows the mapping of the User Interfaces of the System Controller peripherals.

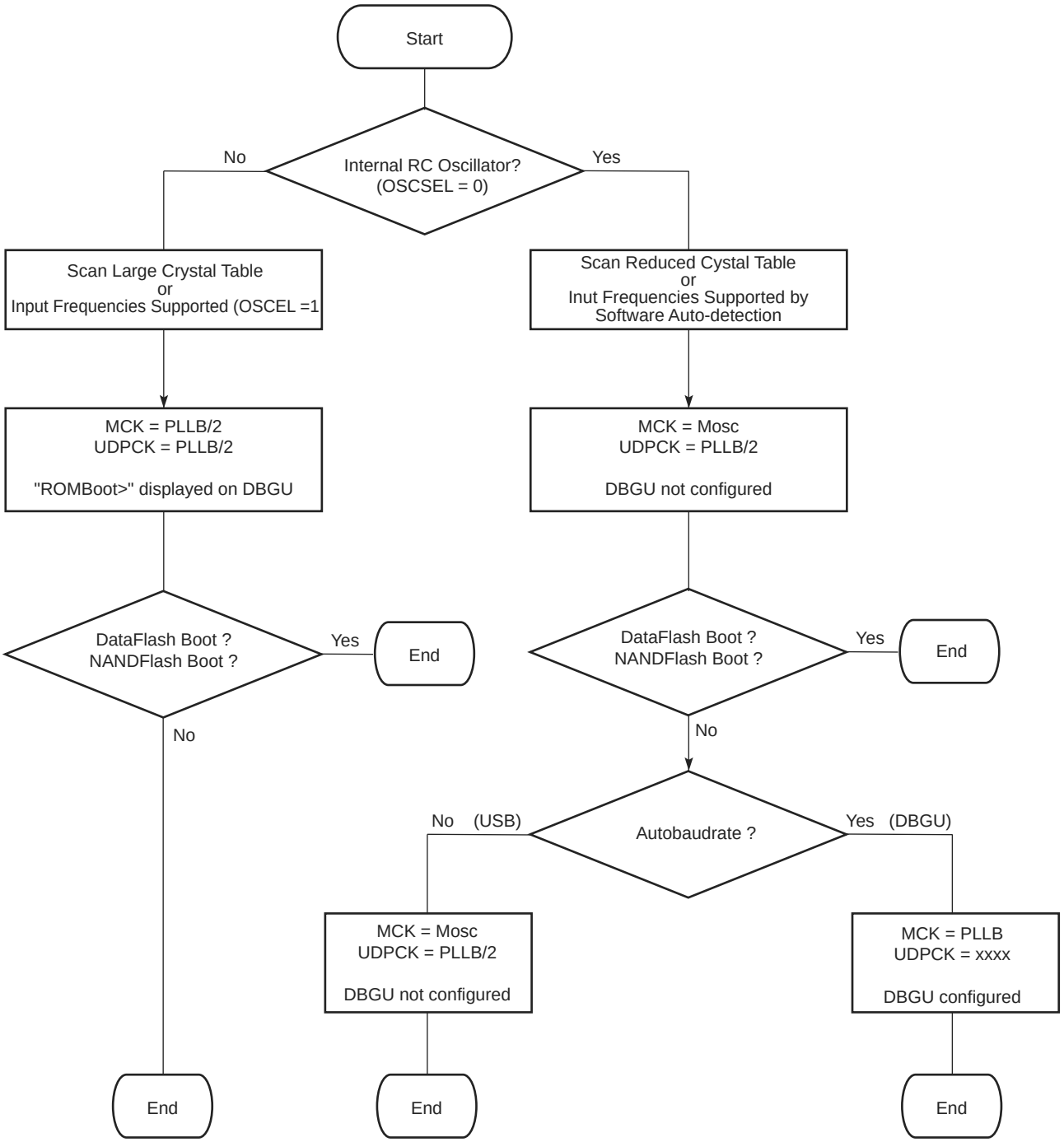
8.3.1 PIO Controller A Multiplexing

Table 8-2. Multiplexing on PIO Controller A

PIO Controller A					Application Usage		
I/O Line	**Peripheral A**	**Peripheral B**	**Comments**	**Reset State**	**Power Supply**	**Function**	**Comments**
PA0	SPI0_MISO	MCDB0		I/O	VDDIOP0		
PA1	SPI0_MOSI	MCCDB		I/O	VDDIOP0		
PA2	SPI0_SPCK			I/O	VDDIOP0		
PA3	SPI0_NPCS0	MCDB3		I/O	VDDIOP0		
PA4	RTS2	MCDB2		I/O	VDDIOP0		
PA5	CTS2	MCDB1		I/O	VDDIOP0		
PA6	MCDA0			I/O	VDDIOP0		
PA7	MCCDA			I/O	VDDIOP0		
PA8	MCCK			I/O	VDDIOP0		
PA9	MCDA1			I/O	VDDIOP0		
PA10	MCDA2	ETX2		I/O	VDDIOP0		
PA11	MCDA3	ETX3		I/O	VDDIOP0		
PA12	ETX0			I/O	VDDIOP0		
PA13	ETX1			I/O	VDDIOP0		
PA14	ERX0			I/O	VDDIOP0		
PA15	ERX1			I/O	VDDIOP0		
PA16	ETXEN			I/O	VDDIOP0		
PA17	ERXDV			I/O	VDDIOP0		
PA18	ERXER			I/O	VDDIOP0		
PA19	ETXCK			I/O	VDDIOP0		
PA20	EMDC			I/O	VDDIOP0		
PA21	EMDIO			I/O	VDDIOP0		
PA22	ADTRG	ETXER		I/O	VDDIOP0		
PA23	TWD	ETX2		I/O	VDDIOP0		
PA24	TWCK	ETX3		I/O	VDDIOP0		
PA25	TCLK0	ERX2		I/O	VDDIOP0		
PA26	TIOA0	ERX3		I/O	VDDIOP0		
PA27	TIOA1	ERXCK		I/O	VDDIOP0		
PA28	TIOA2	ECRS		I/O	VDDIOP0		
PA29	SCK1	ECOL		I/O	VDDIOP0		
PA30[1]	SCK2	RXD4		I/O	VDDIOP0		
PA31[1]	SCK0	TXD4		I/O	VDDIOP0		

Note: 1. Not available in the 208-lead PQFP package.

Figure 11-2. Clocks and DBGU Configurations



17.7.1 EBI Chip Select Assignment Register

Name: EBI_CSA

Access: Read/Write

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

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

15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	EBI_DBPUC

7	6	5	4	3	2	1	0
–	–	EBI_CS5A	EBI_CS4A	EBI_CS3A	–	EBI_CS1A	–

- **EBI_CS1A: EBI Chip Select 1 Assignment**

0: EBI Chip Select 1 is assigned to the Static Memory Controller.

1: EBI Chip Select 1 is assigned to the SDRAM Controller.

- **EBI_CS3A: EBI Chip Select 3 Assignment**

0: EBI Chip Select 3 is only assigned to the Static Memory Controller and EBI_NCS3 behaves as defined by the SMC.

1: EBI Chip Select 3 is assigned to the Static Memory Controller and the SmartMedia Logic is activated.

- **EBI_CS4A: EBI Chip Select 4 Assignment**

0: EBI Chip Select 4 is only assigned to the Static Memory Controller and EBI_NCS4 behaves as defined by the SMC.

1: EBI Chip Select 4 is assigned to the Static Memory Controller and the CompactFlash Logic (first slot) is activated.

- **EBI_CS5A: EBI Chip Select 5 Assignment**

0: EBI Chip Select 5 is only assigned to the Static Memory Controller and EBI_NCS5 behaves as defined by the SMC.

1: EBI Chip Select 5 is assigned to the Static Memory Controller and the CompactFlash Logic (second slot) is activated.

- **EBI_DBPUC: EBI Data Bus Pull-Up Configuration**

0: EBI D0–D15 Data Bus bits are internally pulled-up to the VDDIOM power supply.

1: EBI D0–D15 Data Bus bits are not internally pulled-up.

- **VDDIOMSEL: Memory Voltage Selection**

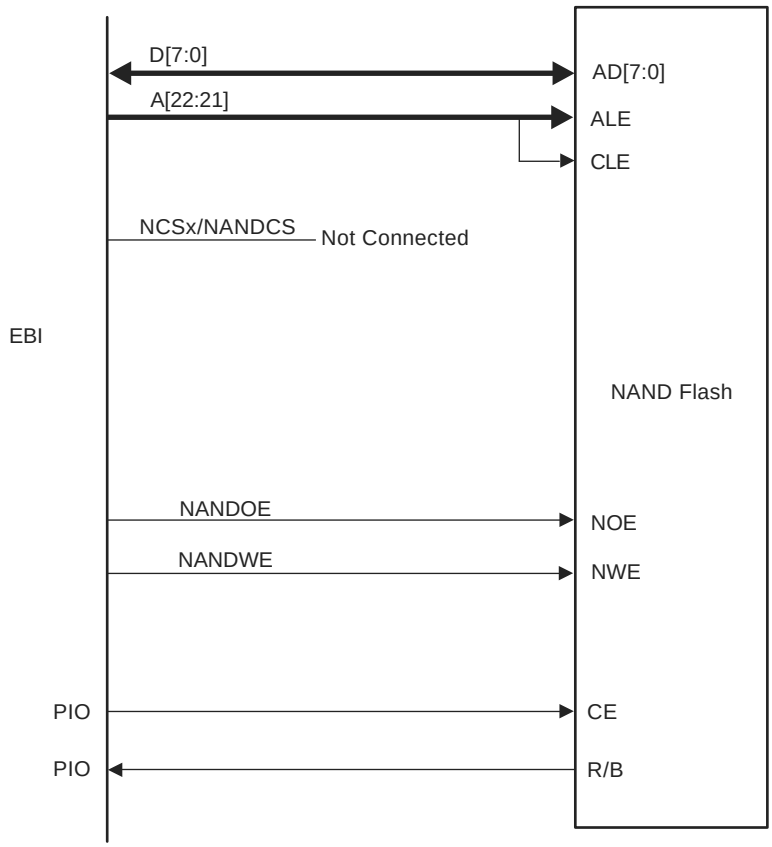
0: Memories are 1.8V powered.

1: Memories are 3.3V powered.

18.6.7.2 NAND Flash Signals

The address latch enable and command latch enable signals on the NAND Flash device are driven by address bits A22 and A21 of the EBI address bus. The command, address or data words on the data bus of the NAND Flash device are distinguished by using their address within the NCSx address space. The chip enable (CE) signal of the device and the ready/busy (R/B) signals are connected to PIO lines. The CE signal then remains asserted even when NCSx is not selected, preventing the device from returning to standby mode.

Figure 18-7. NAND Flash Application Example



Note: The External Bus Interface is also able to support 16-bit devices.

19.7.2.1 Byte Write Access

Byte write access supports one byte write signal per byte of the data bus and a single read signal.

Note that the SMC does not allow boot in Byte Write Access mode.

- For 16-bit devices: the SMC provides NWR0 and NWR1 write signals for respectively byte0 (lower byte) and byte1 (upper byte) of a 16-bit bus. One single read signal (NRD) is provided.

  Byte Write Access is used to connect 2 x 8-bit devices as a 16-bit memory.

- For 32-bit devices: NWR0, NWR1, NWR2 and NWR3, are the write signals of byte0 (lower byte), byte1, byte2 and byte 3 (upper byte) respectively. One single read signal (NRD) is provided.

  Byte Write Access is used to connect 4 x 8-bit devices as a 32-bit memory.

  Byte Write option is illustrated on Figure 19-6.

19.7.2.2 Byte Select Access

In this mode, read/write operations can be enabled/disabled at a byte level. One byte-select line per byte of the data bus is provided. One NRD and one NWE signal control read and write.

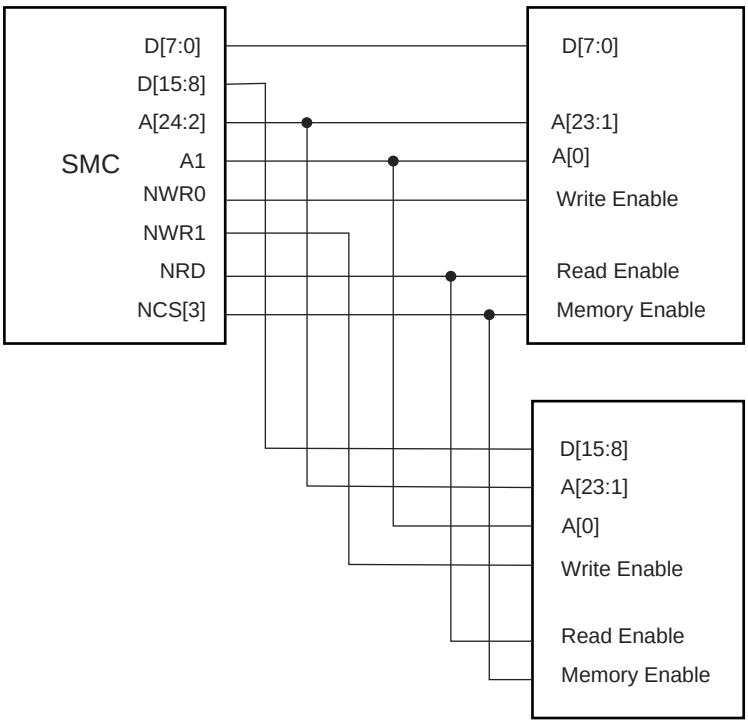
- For 16-bit devices: the SMC provides NBS0 and NBS1 selection signals for respectively byte0 (lower byte) and byte1 (upper byte) of a 16-bit bus.

  Byte Select Access is used to connect one 16-bit device.

- For 32-bit devices: NBS0, NBS1, NBS2 and NBS3, are the selection signals of byte0 (lower byte), byte1, byte2 and byte 3 (upper byte) respectively. Byte Select Access is used to connect two 16-bit devices.

Figure 19-7 shows how to connect two 16-bit devices on a 32-bit data bus in Byte Select Access mode, on NCS3 (BAT = Byte Select Access).

Figure 19-6. Connection of 2 x 8-bit Devices on a 16-bit Bus: Byte Write Option



20.6.4 SDRAMC Low Power Register

Name: SDRAMC_LPR

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
–	–	TIMEOUT		DS		TCSR	
7	6	5	4	3	2	1	0
–	PASR			–	–	LPCB	

- **LPCB: Low-power Configuration Bits**

Value	Description
00	Low Power Feature is inhibited: no Power-down, Self-refresh or Deep Power-down command is issued to the SDRAM device.
01	The SDRAM Controller issues a Self-refresh command to the SDRAM device, the SDCLK clock is deactivated and the SDCKE signal is set low. The SDRAM device leaves the Self Refresh Mode when accessed and enters it after the access.
10	The SDRAM Controller issues a Power-down Command to the SDRAM device after each access, the SDCKE signal is set to low. The SDRAM device leaves the Power-down Mode when accessed and enters it after the access.
11	The SDRAM Controller issues a Deep Power-down command to the SDRAM device. This mode is unique to low-power SDRAM.

- **PASR: Partial Array Self-refresh (only for low-power SDRAM)**

PASR parameter is transmitted to the SDRAM during initialization to specify whether only one quarter, one half or all banks of the SDRAM array are enabled. Disabled banks are not refreshed in self-refresh mode. This parameter must be set according to the SDRAM device specification.

- **TCSR: Temperature Compensated Self-Refresh (only for low-power SDRAM)**

TCSR parameter is transmitted to the SDRAM during initialization to set the refresh interval during self-refresh mode depending on the temperature of the low-power SDRAM. This parameter must be set according to the SDRAM device specification.

- **DS: Drive Strength (only for low-power SDRAM)**

DS parameter is transmitted to the SDRAM during initialization to select the SDRAM strength of data output. This parameter must be set according to the SDRAM device specification.

- **TIMEOUT: Time to define when low-power mode is enabled**

Value	Description
00	The SDRAM controller activates the SDRAM low-power mode immediately after the end of the last transfer.
01	The SDRAM controller activates the SDRAM low-power mode 64 clock cycles after the end of the last transfer.
10	The SDRAM controller activates the SDRAM low-power mode 128 clock cycles after the end of the last transfer.
11	Reserved.

20.6.8 SDRAMC Interrupt Status Register

Name: SDRAMC_ISR

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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	RES

- **RES: Refresh Error Status**

0: No refresh error has been detected since the register was last read.

1: A refresh error has been detected since the register was last read.

The PLLACOUNT field specifies the number of slow clock cycles before LOCKA bit is set in the PMC_SR after CKGR_PLLAR has been written.

Once CKGR_PLLAR has been written, the user is obliged to wait for the LOCKA bit to be set in the PMC_SR. This can be done either by polling the status register or by waiting the interrupt line to be raised if the associated interrupt to LOCKA has been enabled in the PMC_IER.

All parameters in CKGR_PLLAR can be programmed in a single write operation. If at some stage one of the following parameters, SRCA, MULA, DIVA is modified, LOCKA bit will go low to indicate that PLL A is not ready yet. When PLL A is locked, LOCKA will be set again. User has to wait for LOCKA bit to be set before using the PLL A output clock.

Code Example:

```
write_register(CKGR_PLLAR,0x20030605)
```

PLL A and divider A are enabled. PLL A input clock is main clock divided by 5. PLL An output clock is PLL A input clock multiplied by 4. Once CKGR_PLLAR has been written, LOCKA bit will be set after six slow clock cycles.

4. Setting PLL B and divider B:

   All parameters needed to configure PLL B and divider B are located in the CKGR_PLLBR.

   The DIVB field is used to control divider B itself. A value between 0 and 255 can be programmed. Divider B output is divider B input divided by DIVB parameter. By default DIVB parameter is set to 0 which means that divider B is turned off.

   The OUTB field is used to select the PLL B output frequency range.

   The MULB field is the PLL B multiplier factor. This parameter can be programmed between 0 and 2047. If MULB is set to 0, PLL B will be turned off, otherwise the PLL B output frequency is PLL B input frequency multiplied by (MULB + 1).

   The PLLBCOUNT field specifies the number of slow clock cycles before LOCKB bit is set in the PMC_SR after CKGR_PLLBR has been written.

   Once the PMC_PLLB register has been written, the user must wait for the LOCKB bit to be set in the PMC_SR. This can be done either by polling the status register or by waiting the interrupt line to be raised if the associated interrupt to LOCKB has been enabled in the PMC_IER. All parameters in CKGR_PLLBR can be programmed in a single write operation. If at some stage one of the following parameters, MULB, DIVB is modified, LOCKB bit will go low to indicate that PLL B is not ready yet. When PLL B is locked, LOCKB will be set again. The user is constrained to wait for LOCKB bit to be set before using the PLL A output clock.

   The USBDIV field is used to control the additional divider by 1, 2 or 4, which generates the USB clock(s).

   Code Example:

   ```
   write_register(CKGR_PLLBR,0x00040805)
   ```

   If PLL B and divider B are enabled, the PLL B input clock is the main clock. PLL B output clock is PLL B input clock multiplied by 5. Once CKGR_PLLBR has been written, LOCKB bit will be set after eight slow clock cycles.

5. Selection of Master Clock and Processor Clock

   The Master Clock and the Processor Clock are configurable via the PMC_MCKR.

   The CSS field is used to select the Master Clock divider source. By default, the selected clock source is slow clock.

generated if enabled in the interrupt enable register (TWI_IER). If the slave acknowledges the byte, the data written in the TWI_THR, is then shifted in the internal shifter and transferred. When an acknowledge is detected, the TXRDY bit is set until a new write in the TWI_THR. When no more data is written into the TWI_THR, the master generates a stop condition to end the transfer. The end of the complete transfer is marked by the TWI_TXCOMP bit set to one. See Figure 29-6, Figure 29-7, and Figure 29-8.

Figure 29-6. Master Write with One Data Byte

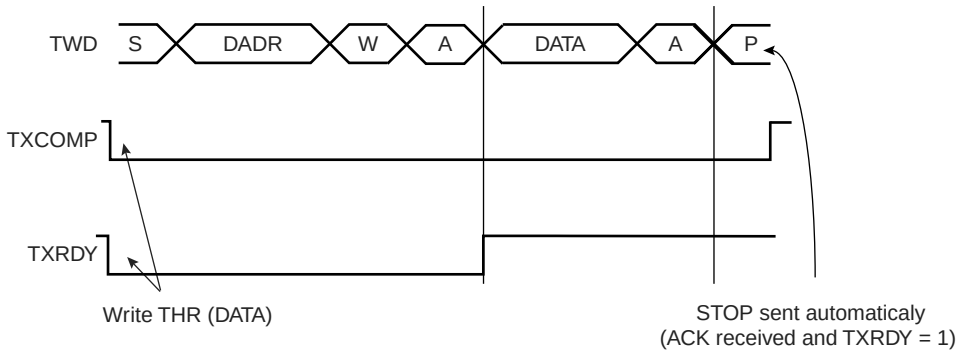


Figure 29-7. Master Write with Multiple Data Byte

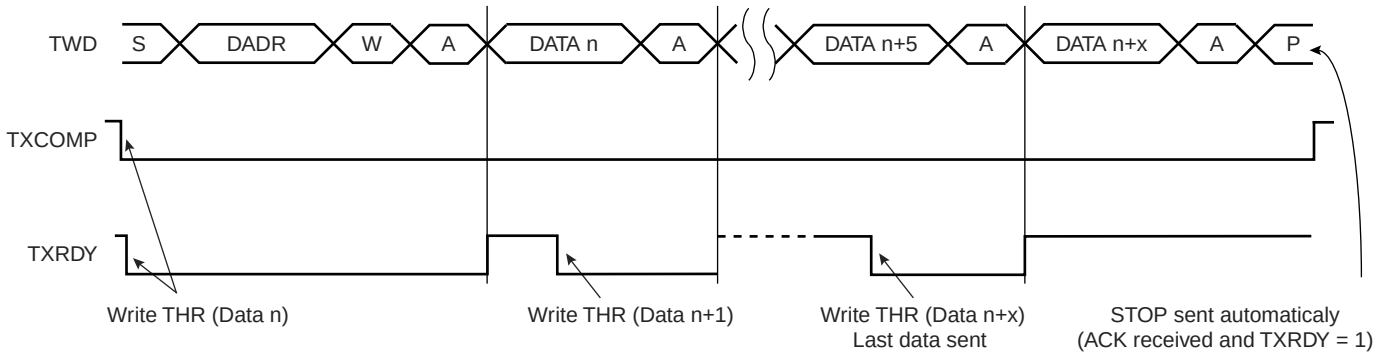
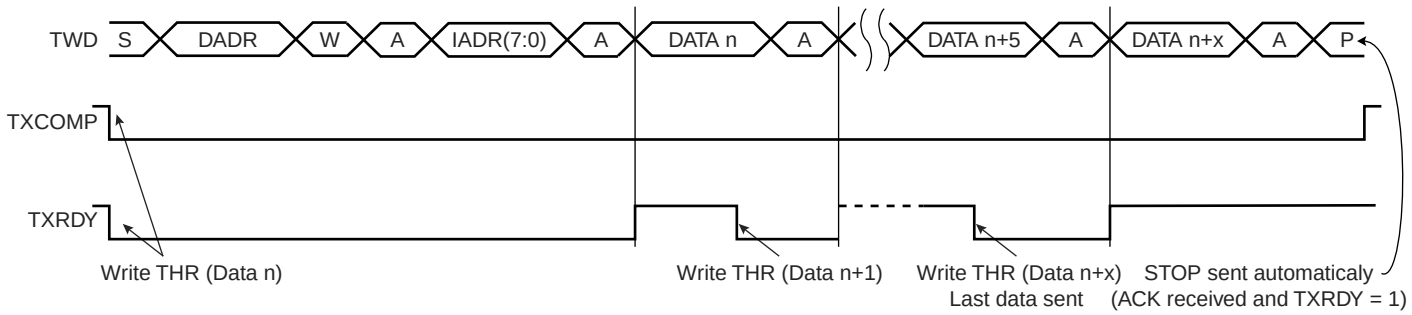


Figure 29-8. Master Write with One Byte Internal Address and Multiple Data Bytes



29.7.3.6 Internal Address

The TWI interface can perform various transfer formats: Transfers with 7-bit slave address devices and 10-bit slave address devices.

7-bit Slave Addressing

When Addressing 7-bit slave devices, the internal address bytes are used to perform random address (read or write) accesses to reach one or more data bytes, within a memory page location in a serial memory, for example. When performing read operations with an internal address, the TWI performs a write operation to set the internal address into the slave device, and then switch to Master Receiver mode. Note that the second start condition (after sending the IADR) is sometimes called “repeated start” (Sr) in I2C fully-compatible devices. See Figure 29-12. See Figure 29-11 and Figure 29-13 for Master Write operation with internal address.

The three internal address bytes are configurable through the Master Mode register (TWI_MMR).

If the slave device supports only a 7-bit address, i.e. no internal address, **IADRSZ** must be set to 0.

In the figures below the following abbreviations are used:

- S Start
- Sr Repeated Start
- P Stop
- W Write
- R Read
- A Acknowledge
- N Not Acknowledge
- DADR Device Address
- IADR Internal Address

Figure 29-11. Master Write with One, Two or Three Bytes Internal Address and One Data Byte

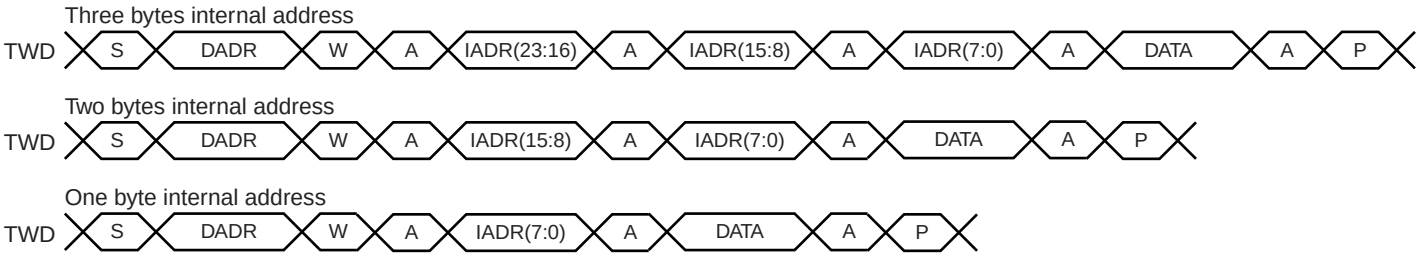
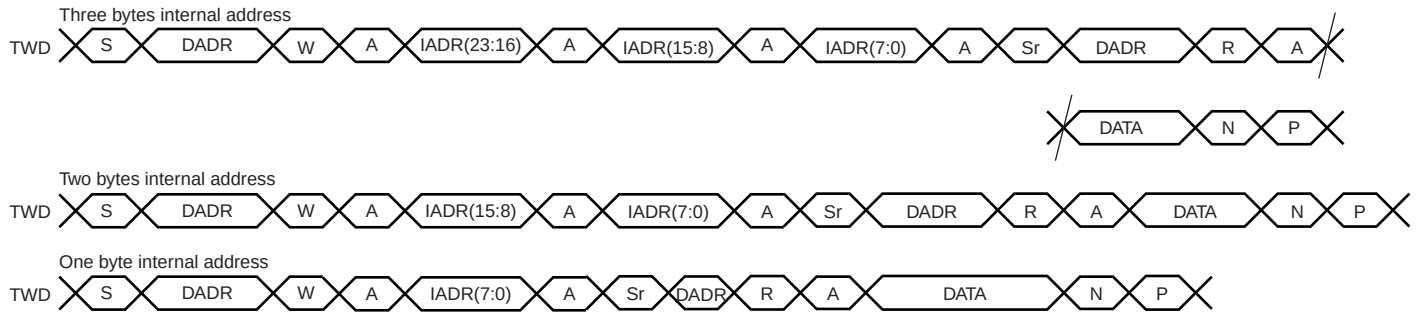
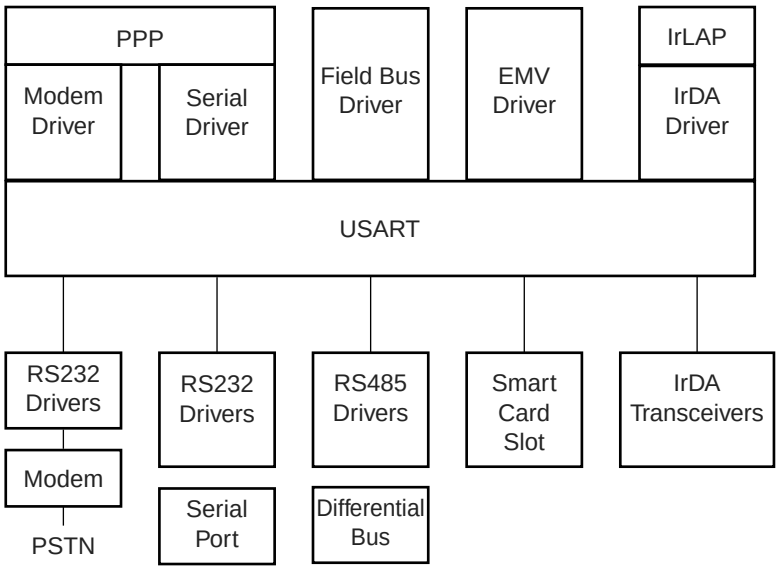


Figure 29-12. Master Read with One, Two or Three Bytes Internal Address and One Data Byte



30.4 Application Block Diagram

Figure 30-2. Application Block Diagram



30.5 I/O Lines Description

Table 30-1. I/O Line Description

Name	Description	Type	Active Level
SCK	Serial Clock	I/O	–
TXD	Transmit Serial Data	I/O	–
RXD	Receive Serial Data	Input	–
RI	Ring Indicator	Input	Low
DSR	Data Set Ready	Input	Low
DCD	Data Carrier Detect	Input	Low
DTR	Data Terminal Ready	Output	Low
CTS	Clear to Send	Input	Low
RTS	Request to Send	Output	Low

Figure 30-18 shows how the receiver operates if hardware handshaking is enabled. The RTS pin is driven high if the receiver is disabled and 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 to the PDC clears the status bit RXBUFF and, as a result, asserts the pin RTS low.

Figure 30-18. Receiver Behavior when Operating with Hardware Handshaking

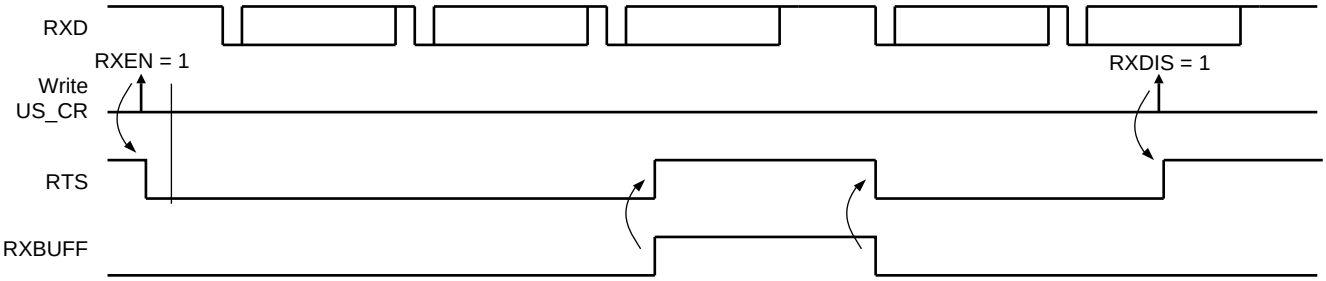
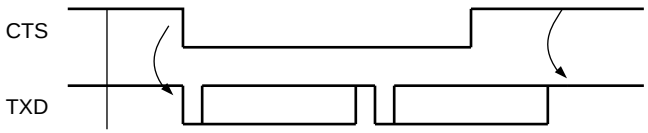


Figure 30-19 shows how the transmitter operates if hardware handshaking is enabled. The CTS pin disables the transmitter. If a character is being processing, the transmitter is disabled only after the completion of the current character and transmission of the next character happens as soon as the pin CTS falls.

Figure 30-19. Transmitter Behavior when Operating with Hardware Handshaking



30.7.4 ISO7816 Mode

The USART features an ISO7816-compatible operating mode. This mode permits interfacing with smart cards and Security Access Modules (SAM) communicating through an ISO7816 link. Both T = 0 and T = 1 protocols defined by the ISO7816 specification are supported.

Setting the USART in ISO7816 mode is performed by writing the USART_MODE field in the Mode Register (US_MR) to the value 0x4 for protocol T = 0 and to the value 0x5 for protocol T = 1.

30.7.4.1 ISO7816 Mode Overview

The ISO7816 is a half duplex communication on only one bidirectional line. The baud rate is determined by a division of the clock provided to the remote device (see “Baud Rate Generator” on page 447).

The USART connects to a smart card as shown in Figure 30-20. The TXD line becomes bidirectional and the Baud Rate Generator feeds the ISO7816 clock on the SCK pin. As the TXD pin becomes bidirectional, its output remains driven by the output of the transmitter but only when the transmitter is active while its input is directed to the input of the receiver. The USART is considered as the master of the communication as it generates the clock.

Figure 30-20. Connection of a Smart Card to the USART

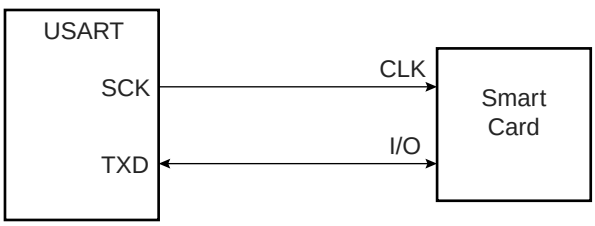
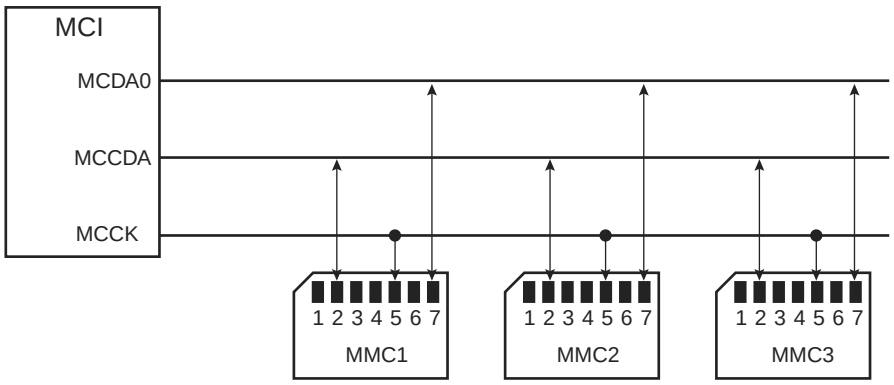
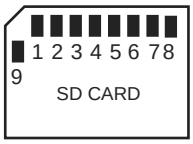


Figure 33-4. MMC Bus Connections (One Slot)



Note: When several MCI (x MCI) are embedded in a product, MCCK refers to MCIx_CK, MCCDA to MCIx_CDA MCDAy to MCIx_DAy.

Figure 33-5. SD Memory Card Bus Topology



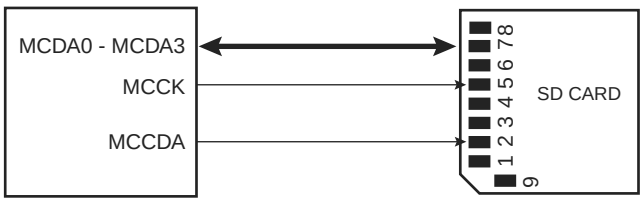
The SD Memory Card bus includes the signals listed in Table 33-3.

Table 33-3. SD Memory Card Bus Signals

Pin Number	Name	Type [1]	Description	MCI Pin Name[2] (Slot z)
1	CD/DAT[3]	I/O/PP	Card detect/ Data line Bit 3	MCDz3
2	CMD	PP	Command/response	MCCDz
3	VSS1	S	Supply voltage ground	VSS
4	VDD	S	Supply voltage	VDD
5	CLK	I/O	Clock	MCCK
6	VSS2	S	Supply voltage ground	VSS
7	DAT[0]	I/O/PP	Data line Bit 0	MCDz0
8	DAT[1]	I/O/PP	Data line Bit 1 or Interrupt	MCDz1
9	DAT[2]	I/O/PP	Data line Bit 2	MCDz2

Notes: 1. I: input, O: output, PP: Push Pull, OD: Open Drain.
2. When several MCI (x MCI) are embedded in a product, MCCK refers to MCIx_CK, MCCDA to MCIx_CDA, MCCDB to MCIx_CDB, MCDAy to MCIx_DAy, MCDBy to MCIx_DBy.

Figure 33-6. SD Card Bus Connections with One Slot



Note: When several MCI (x MCI) are embedded in a product, MCCK refers to MCIx_CK, MCCDA to MCIx_CDA MCDAy to MCIx_DAy.

34.6.21 Specific Address 3 Top Register

Name: EMAC_SA3T

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
ADDR							

7	6	5	4	3	2	1	0
ADDR							

- **ADDR**

The most significant bits of the destination address, that is bits 47 to 32.

37.4.5.2 Memory Interface

Dedicated FIFO are used to support packed memory mapping. YCrCb pixel components are sent in a single 32-bit word in a contiguous space (packed). Data is stored in the order of natural scan lines. Planar mode is not supported.

37.4.5.3 DMA Features

Unlike preview datapath, codec datapath DMA mode does not support linked list operation. Only the CODEC_DMA_ADDR is used to configure the frame buffer base address.

38. Analog-to-Digital Converter (ADC)

38.1 Description

The ADC is based on a Successive Approximation Register (SAR) 10-bit Analog-to-Digital Converter (ADC). It also integrates an 4-to-1 analog multiplexer, making possible the analog-to-digital conversions of 4 analog lines. The conversions extend from 0V to ADVREF.

The ADC supports an 8-bit or 10-bit resolution mode, and conversion results are reported in a common register for all channels, as well as in a channel-dedicated register. Software trigger, external trigger on rising edge of the ADTRG pin or internal triggers from Timer Counter output(s) are configurable.

The ADC also integrates a Sleep Mode and a conversion sequencer and connects with a PDC channel. These features reduce both power consumption and processor intervention.

Finally, the user can configure ADC timings, such as Startup Time and Sample & Hold Time.

38.2 Embedded Characteristics

- 4-channel ADC
- 10-bit 312K samples/sec. Successive Approximation Register ADC
- -2/+2 LSB Integral Non Linearity, -1/+1 LSB Differential Non Linearity
- Individual enable and disable of each channel
- External voltage reference for better accuracy on low voltage inputs
- Multiple trigger source – Hardware or software trigger – External trigger pin – Timer Counter 0 to 2 outputs TIOA0 to TIOA2 trigger
- Sleep Mode and conversion sequencer – Automatic wakeup on trigger and back to sleep mode after conversions of all enabled channels
- Four analog inputs shared with digital signals

38.7.5 ADC Channel Status Register

Name: ADC_CHSR

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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
-	-	-	-	CH3	CH2	CH1	CH0

- **CHx: Channel x Status**

0: Corresponding channel is disabled.

1: Corresponding channel is enabled.

Revision 6221K	Comments	Change Request Ref
	”SAM9260 Errata”, removed device marking -- moved to Section 41. “Marking” Removed 208-pin package and 217-ball package outlines: Formerly, Figure 4-1 and Figure 4-2. Added: Figure 40-1 ”217-ball LFBGA: Ball A1 Position” and Figure 40-3 ”208-lead PQFP: Pin 1 Position”. Changed document format: pagination has changed.	8450
	”SAM9260 Errata”, changes to Section 43.1.7 “Reset Controller (RSTC)”, Section 43.2.7 “Reset Controller (RSTC)” involves user reset, watchdog reset, user reset.	8305

Revision 6221J	Comments	Change Reques Ref
	Introduction: Documument title and name of product updated to conform to AT91SAM Marketing standards: AT91SAM ARM-based MPU. AT91SAM9260 now referenced in text as SAM9260. “Features” , removed SDCard from System list, boot possibilities.	7142
	BOOT ROM: Section 11.5 “NAND Flash Boot”, opening paragraph updated.	7643
	EMAC: Section 34.4.1.1 “FIFO”, restored “receive” and “transmit” to first line of text. “The FIFO depths are 28 bytes for receive and 28 bytes for transmit and...”	6980
	SHDWC: Section 16.7.3 “Shutdown Status Register”, RTTWK occupies bitfield 16.	6583
	SMC: Section 19.8.6 “Reset Values of Timing Parameters”, former Table 20-5. “Reset Values of Timing Parameters” removed and added cross reference to Table 19-8, “Register Mapping”.	6742
	Electrical Characteristics: Section 39.4.2 “Power-up Sequence”, This section updated, detailing startup with VDDBU powered by battery and startup sequence without. Section 39.4.2.1 “VDDBU is Continuously Powered (used with a battery)” Section 39.4.2.2 “VDDBU is not Continuously Powered (no backup features used)” Section 39.6.7 “PLL Characteristics”, added T, Startup Time to Table 39-16, “PLLB Characteristics”	7731 6675
	Section 39.11.1 “SPI”, added sections giving maximum SPI frequency in master and slave modes as follows: Section 39.11.1.1 “Maximum SPI Frequency”, Section “Master Write Mode”, Section “Master Read Mode”, Section “Slave Read Mode”, Section “Slave Write Mode”.	7173
	Section 39.11.1 “SPI”, simplified figure titles. The new titles are as follows: Figure 39-10 ”SPI Master Mode 1 and 2”, Figure 39-11 ”SPI Master Mode 0 and 3”, Figure 39-12 ”SPI Slave Mode 0 and 3”, Figure 39-13 ”SPI Slave Mode 1 and 2”.	6872