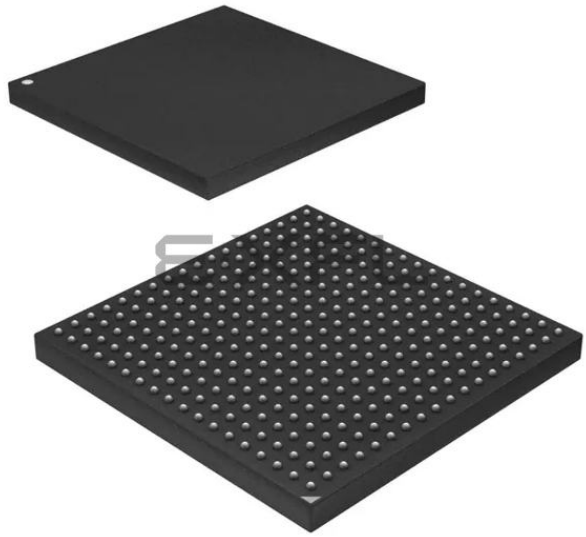


Welcome to E-XFL.COM

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	Obsolete
Core Processor	ARM926EJ-S
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	400MHz
Co-Processors/DSP	-
RAM Controllers	LPDDR, LPSDR, DDR2, SDR, SRAM
Graphics Acceleration	No
Display & Interface Controllers	LCD, Touchscreen
Ethernet	10/100Mbps
SATA	-
USB	USB 2.0 (3)
Voltage - I/O	1.8V, 3.3V
Operating Temperature	-40°C ~ 85°C (TA)
Security Features	-
Package / Case	324-TFBGA
Supplier Device Package	324-TFBGA (15x15)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at91sam9g45-cu

9. The application must go into Normal Mode, setting Mode to 0 in the Mode Register (see Section 22.7.1 on page 258) and performing a write access at any location in the low-power DDR1-SDRAM to acknowledge this command.
10. Perform a write access to any low-power DDR1-SDRAM address.
11. Write the refresh rate into the count field in the DDRSDRC Refresh Timer register (see page 259). (Refresh rate = delay between refresh cycles). The low-power DDR1-SDRAM device requires a refresh every 15.625 µs or 7.81 µs. With a 100 MHz frequency, the refresh timer count register must to be set with (15.625 /100 MHz) = 1562 i.e. 0x061A or (7.81 /100 MHz) = 781 i.e. 0x030d
12. After initialization, the low-power DDR1-SDRAM device is fully functional.

22.3.3 DDR2-SDRAM Initialization

The initialization sequence is generated by software. The DDR2-SDRAM devices are initialized by the following sequence:

1. Program the memory device type into the Memory Device Register (see Section 22.7.8 on page 269).
2. Program the features of DDR2-SDRAM device into the Timing Register (asynchronous timing (trc, tras, etc.)), and into the Configuration Register (number of columns, rows, banks, cas latency and output drive strength) (see Section 22.7.3 on page 260, Section 22.7.4 on page 263 and Section 22.7.5 on page 265).
3. An NOP command is issued to the DDR2-SDRAM. Program the NOP command into the Mode Register, the application must set Mode to 1 in the Mode Register (see Section 22.7.1 on page 258). Perform a write access to any DDR2-SDRAM address to acknowledge this command. Now clocks which drive DDR2-SDRAM device are enabled.

A minimum pause of 200 µs is provided to precede any signal toggle.

4. An NOP command is issued to the DDR2-SDRAM. Program the NOP command into the Mode Register, the application must set Mode to 1 in the Mode Register (see Section 22.7.1 on page 258). Perform a write access to any DDR2-SDRAM address to acknowledge this command. Now CKE is driven high.
5. An all banks precharge command is issued to the DDR2-SDRAM. Program all banks precharge command into the Mode Register, the application must set Mode to 2 in the Mode Register (See Section 22.7.1 on page 258). Perform a write access to any DDR2-SDRAM address to acknowledge this command
6. An Extended Mode Register set (EMRS2) cycle is issued to chose between commercial or high temperature operations. The application must set Mode to 5 in the Mode Register (see Section 22.7.1 on page 258) and perform a write access to the DDR2-SDRAM to acknowledge this command. The write address must be chosen so that BA[1] is set to 1 and BA[0] is set to 0. For example, with a 16-bit 128 MB DDR2-SDRAM (12 rows, 9 columns, 4 banks) bank address, the DDR2-SDRAM write access should be done at the address 0x20800000.

Note: This address is for example purposes only. The real address is dependent on implementation in the product.

7. An Extended Mode Register set (EMRS3) cycle is issued to set all registers to “0”. The application must set Mode to 5 in the Mode Register (see Section 22.7.1 on page 258) and perform a write access to the DDR2-SDRAM to acknowledge this command. The write address must be chosen so that BA[1] is set to 1 and BA[0] is set to 1. For example, with a 16-bit 128 MB DDR2-SDRAM (12 rows, 9 columns, 4 banks) bank address, the DDR2-SDRAM write access should be done at the address 0x20C00000.

23.7.3 ECC Parity Register 2

Register Name : ECC_PR2

Access Type : Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
0	NPARITY2						
15	14	13	12	11	10	9	8
NPARITY2				0	WORDADD2		
7	6	5	4	3	2	1	0
WORDADDR2					BITADDR2		

Once the entire main area of a page is written with data, the register content must be stored at any free location of the spare area.

- BITADDR2: corrupted Bit Address in the pa ge between the 512th and the 767th bytes

During a page read, this value contains the corrupted bit offset where an error occurred, if a single error was detected. If multiple errors were detected, this value is meaningless.

- WORDADDR2: corrupted Word Address in the pa ge between the 512th and the 767th bytes

During a page read, this value contains the word address (8-bit word) where an error occurred, if a single error was detected. If multiple errors were detected, this value is meaningless.

- NPARITY2:

Parity N

27.8.2 Interrupt Latencies

Global interrupt latencies depend on several parameters, including:

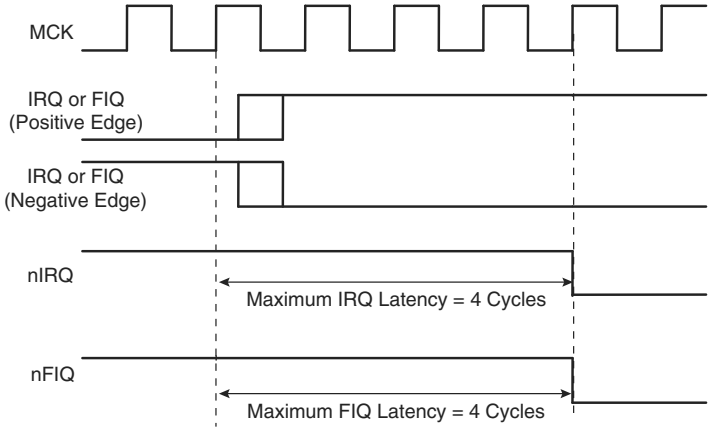
- The time the software masks the interrupts.
- Occurrence, either at the processor level or at the AIC level.
- The execution time of the instruction in progress when the interrupt occurs.
- The treatment of higher priority interrupts and the resynchronization of the hardware signals.

This section addresses only the hardware resynchronizations. It gives details of the latency times between the event on an external interrupt leading in a valid interrupt (edge or level) or the assertion of an internal interrupt source and the assertion of the nIRQ or nFIQ line on the processor. The resynchronization time depends on the programming of the interrupt source and on its type (internal or external). For the standard interrupt, resynchronization times are given assuming there is no higher priority in progress.

The PIO Controller multiplexing has no effect on the interrupt latencies of the external interrupt sources.

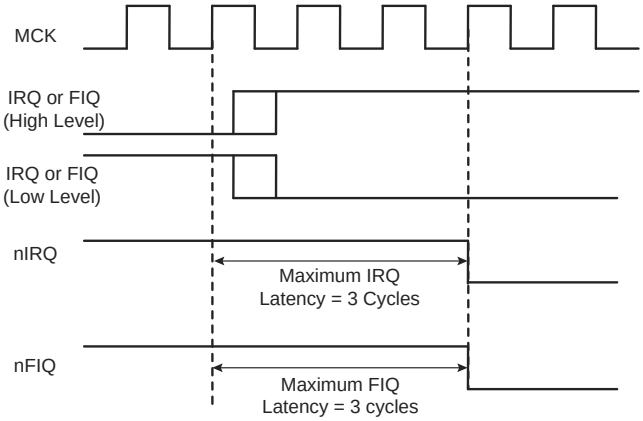
27.8.2.1 External Interrupt Edge Triggered Source

Figure 27-6. External Interrupt Edge Triggered Source



27.8.2.2 External Interrupt Level Sensitive Source

Figure 27-7. External Interrupt Level Sensitive Source



30.7.3.9 Peripheral Chip Select Decoding

The user can program the SPI to operate with up to 15 peripherals by decoding the four Chip Select lines, NPCS0 to NPCS3 with 1 of up to 16 decoder/demultiplexer. This can be enabled by writing the PCSDEC bit at 1 in the Mode Register (SPI_MR).

When operating without decoding, the SPI makes sure that in any case only one chip select line is activated, i.e., one NPCS line driven low at a time. If two bits are defined low in a PCS field, only the lowest numbered chip select is driven low.

When operating with decoding, the SPI directly outputs the value defined by the PCS field on NPCS lines of either the Mode Register or the Transmit Data Register (depending on PS).

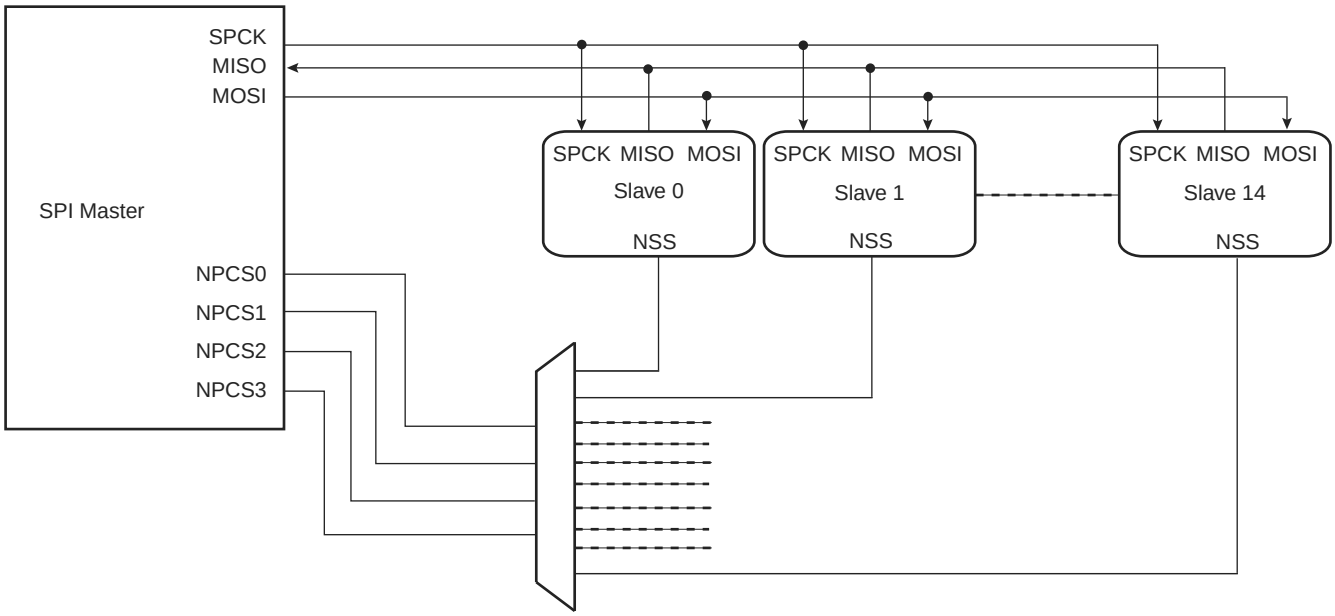
As the SPI sets a default value of 0xF on the chip select lines (i.e. all chip select lines at 1) when not processing any transfer, only 15 peripherals can be decoded.

The SPI has only four Chip Select Registers, not 15. As a result, when decoding is activated, each chip select defines the characteristics of up to four peripherals. As an example, SPI_CRS0 defines the characteristics of the externally decoded peripherals 0 to 3, corresponding to the PCS values 0x0 to 0x3. Thus, the user has to make sure to connect compatible peripherals on the decoded chip select lines 0 to 3, 4 to 7, 8 to 11 and 12 to 14. Figure 30-11 below shows such an implementation.

If the CSAAT bit is used, with or without the PDC, the Mode Fault detection for NPCS0 line must be disabled. This is not needed for all other chip select lines since Mode Fault Detection is only on NPCS0.

If the CSAAT bit is used, with or without the DMAC, the Mode Fault detection for NPCS0 line must be disabled. This is not needed for all other chip select lines since Mode Fault Detection is only on NPCS0.

Figure 30-11. Chip Select Decoding Application Block Diagram: Single Master/Multiple Slave Implementation



30.8.4 SPI Transmit Data Register

Name: SPI_TDR

Addresses: 0xFFFA400C (0), 0xFFFA800C (1)

Access: Write-only

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

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

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 SPI Interface is stored in this register. Information to be transmitted must be written to the transmit data register in a right-justified format.

- PCS: Peripheral Chip Select

This field is only used if Variable Peripheral Select is active (PS = 1).

If PCSDEC = 0:

PCS = xxx0 NPCS[3:0] = 1110

PCS = xx01 NPCS[3:0] = 1101

PCS = x011 NPCS[3:0] = 1011

PCS = 0111 NPCS[3:0] = 0111

PCS = 1111 forbidden (no peripheral is selected)

(x = don’t care)

If PCSDEC = 1:

NPCS[3:0] output signals = PCS

- LASTXFER: Last Transfer

0 = No effect.

1 = The current NPCS will be deasserted after the character written in TD has been transferred. When CSAAT is set, this allows to close the communication with the current serial peripheral by raising the corresponding NPCS line as soon as TD transfer has completed.

This field is only used if Variable Peripheral Select is active (PS = 1).

- Multi-master transmitter mode
- Multi-master receiver mode
- Slave transmitter mode
- Slave receiver mode

These modes are described in the following chapters.

31.11.1 TWI Control Register

Name: TWI_CR

Addresses: 0xFFF84000 (0), 0xFFF88000 (1)

Access: Write-only

Reset: 0x00000000

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
SWRST	QUICK	SVDIS	SVEN	MSDIS	MSEN	STOP	START

- START: Send a START Condition

0 = No effect.

1 = A frame beginning with a START bit is transmitted according to the features defined in the mode register.

This action is necessary when the TWI peripheral wants to read data from a slave. When configured in Master Mode with a write operation, a frame is sent as soon as the user writes a character in the Transmit Holding Register (TWI_THR).

- STOP: Send a STOP Condition

0 = No effect.

1 = STOP Condition is sent just after completing the current byte transmission in master read mode.

  – In single data byte master read, the START and STOP must both be set.
  – In multiple data bytes master read, the STOP must be set after the last data received but one.
  – In master read mode, if a NACK bit is received, the STOP is automatically performed.
  – In master data write operation, a STOP condition will be sent after the transmission of the current data is finished.

- MSEN: TWI Master Mode Enabled

0 = No effect.

1 = If MSDIS = 0, the master mode is enabled.

Note: Switching from Slave to Master mode is only permitted when TXCOMP = 1.

- MSDIS: TWI Master Mode Disabled

0 = No effect.

1 = The master mode is disabled, all pending data is transmitted. The shifter and holding characters (if it contains data) are transmitted in case of write operation. In read operation, the character being transferred must be completely received before disabling.

0	1	6 bits
1	0	7 bits
1	1	8 bits

- SYNC/CPHA: Synchronous Mode Select or SPI Clock Phase
  – If USART does not operate in SPI Mode (USART_MODE is 0xE and 0xF):

SYNC = 0: USART operates in Asynchronous Mode.

SYNC = 1: USART operates in Synchronous Mode.

  – If USART operates in SPI Mode (USART_MODE = 0xE or 0xF):

CPHA = 0: Data is changed on the leading edge of SPCK and captured on the following edge of SPCK.

CPHA = 1: Data is captured on the leading edge of SPCK and changed on the following edge of SPCK.

CPHA determines which edge of SPCK causes data to change and which edge causes data to be captured. CPHA is used with CPOL to produce the required clock/data relationship between master and slave devices.

- PAR: Parity Type

PAR			Parity Type
0	0	0	Even parity
0	0	1	Odd parity
0	1	0	Parity forced to 0 (Space)
0	1	1	Parity forced to 1 (Mark)
1	0	x	No parity
1	1	x	Multidrop mode

- NBSTOP: Number of Stop Bits

NBSTOP		Asynchronous (SYNC = 0)	Synchronous (SYNC = 1)
0	0	1 stop bit	1 stop bit
0	1	1.5 stop bits	Reserved
1	0	2 stop bits	2 stop bits
1	1	Reserved	Reserved

- CHMODE: Channel Mode

CHMODE		Mode Description
0	0	Normal Mode
0	1	Automatic Echo. Receiver input is connected to the TXD pin.
1	0	Local Loopback. Transmitter output is connected to the Receiver Input.
1	1	Remote Loopback. RXD pin is internally connected to the TXD pin.

- MSBF/CPOL: Bit Order or SPI Clock Polarity
  – If USART does not operate in SPI Mode (USART_MODE 0xE and 0xF):

33.7.1 TC Block Control Register

Name: TC_BCR

Addresses: 0xFFF7C0C0 (0), 0xFFFD40C0 (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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	SYNC

- SYNC: Synchro Command

0 = no effect.

1 = asserts the SYNC signal which generates a software trigger simultaneously for each of the channels.

33.7.2 TC Block Mode Register

Name: TC_BMR

Addresses: 0xFFF7C0C4 (0), 0xFFFD40C4 (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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	TC2XC2S		TC1XC1S		TC0XC0S	

- TC0XC0S: External Clock Signal 0 Selection

TC0XC0S		Signal Connected to XC0
0	0	TCLK0
0	1	none
1	0	TIOA1
1	1	TIOA2

- TC1XC1S: External Clock Signal 1 Selection

TC1XC1S		Signal Connected to XC1
0	0	TCLK1
0	1	none
1	0	TIOA0
1	1	TIOA2

- TC2XC2S: External Clock Signal 2 Selection

TC2XC2S		Signal Connected to XC2
0	0	TCLK2
0	1	none
1	0	TIOA0
1	1	TIOA1