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	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/atsama5d23b-cur

Table 4-1: Signal Description List (Continued)

Signal Name	Function	Type	Comments	Active Level
External Bus Interface - EBI				
D[15:0]	Data Bus	I/O	–	–
A[25:0]	Address Bus	Output	–	–
NWAIT	External Wait Signal	Input	–	Low
Static Memory Controller - HSMC				
NCS0–NCS3	Chip Select Lines	Output	–	Low
NWR0–NWR1	Write Signal	Output	–	Low
NRD	Read Signal	Output	–	Low
NWE	Write Enable	Output	–	Low
NBS0–NBS1	Byte Mask Signal	Output	–	Low
NANDOE	NAND Flash Output Enable	Output	–	Low
NANDWE	NAND Flash Write Enable	Output	–	Low
DDR2/DDR3/LPDDR1/LPDDR2/LPDDR3 Controller				
DDR_CK, DDR_CLKN	DDR Differential Clock	Output	–	–
DDR_CKE	DDR Clock Enable	Output	When Backup Self-refresh mode is used, should be tied to GND using 100 KΩ pull-down	High
DDR_CS	DDR Controller Chip Select	Output	–	Low
DDR_BA[2:0]	Bank Select	Output	–	Low
DDR_WE	DDR Write Enable	Output	–	Low
DDR_RAS, DDR_CAS	Row and Column Signal	Output	–	Low
DDR_A[13:0]	DDR Address Bus	Output	–	–
DDR_D[31:0]	DDR Data Bus	I/O/-PD	–	–
DDR_DQS[3:0], DDR_DQSN[3:0]	Differential Data Strobe	I/O- PD	–	–
DDR_DQM[3:0]	Write Data Mask	Output	–	–
DDR_CAL	DDR/LPDDR Calibration	Input	–	–
DDR_VREF	DDR/LPDDR Reference	Input	–	–
DDR_RESETN	DDR3 Active Low Asynchronous Reset	Output	When Backup Self-refresh mode is used, should be tied to VDDIODDR using 100 KΩ pull-up	–
Secure Data Memory Card - SDMMCx [1:0]				
SDMMCx_CD	SDcard / e.MMC Card Detect	Input	–	–
SDMMCx_CMD	SDcard / e.MMC Command line	I/O	–	–
SDMMCx_WP	SDcard Connector Write Protect Signal	Input	–	–
SDMMCx_RSTN	e.MMC Reset Signal	Output	–	–
SDMMCx_1V8SEL	SDcard Signal Voltage Selection	Output	–	–

Table 4-1: Signal Description List (Continued)

Signal Name	Function	Type	Comments	Active Level
SDMMCx_CK	SDcard / e.MMC Clock Signal	Output	–	–
SDMMCx_DAT[7:0]	SDcard / e.MMC Data Lines	I/O	–	–
Flexible Serial Communication Controller - FLEXCOMx [4:0]				
FLEXCOMx_IO0	FLEXCOMx Transmit Data	I/O	–	–
FLEXCOMx_IO1	FLEXCOMx Receive Data	I/O	–	–
FLEXCOMx_IO2	FLEXCOMx Serial Clock	I/O	–	–
FLEXCOMx_IO3	FLEXCOMx Clear To Send / Peripheral Chip Select	I/O	–	–
FLEXCOMx_IO4	FLEXCOMx Request To Send / Peripheral Chip Select	Output	–	–
Universal Asynchronous Receiver Transmitter - UARTx [4..0]				
UTXDx	UARTx Transmit Data	Output	–	–
URXDx	UARTx Receive Data	Input	–	–
Inter-IC Sound Controller - I2SCx [1..0]				
I2SCx_MCK	Master Clock	Output	–	–
I2SCx_CK	Serial Clock	I/O	–	–
I2SCx_WS	I^2S Word Select	I/O	–	–
I2SCx_DI0	Serial Data Input	Input	–	–
I2SCx_DO0	Serial Data Output	Output	–	–
Synchronous Serial Controller - SSCx [1..0]				
TDx	SSC Transmit Data	Output	–	–
RDx	SSC Receive Data	Input	–	–
TKx	SSC Transmit Clock	I/O	–	–
RKx	SSC Receive Clock	I/O	–	–
TFx	SSC Transmit Frame Sync	I/O	–	–
RFx	SSC Receive Frame Sync	I/O	–	–
Timer/Counter - TCx [1..0]				
TCLK[5..0]	TC Channel y External Clock Input	Input	–	–
TIOA[5..0]	TC Channel y I/O Line A	I/O	–	–
TIOB[5..0]	TC Channel y I/O Line B	I/O	–	–
Quad IO SPI - QSPIx [1..0]				
QSPIx_SCK	QSPI Serial Clock	Output	–	–
QSPIx_CS	QSPI Chip Select	Output	–	–
QSPIx_IO[0..3]	QSPI I/O QIO0 is QMOSI Master Out - Slave In QIO1 is QMISO Master In - Slave Out	I/O	–	–

- NAND Flash (MLC and SLC) 8-bit datapath

The Static Memory Controller is able to drive up to four chip select. NCS3 is dedicated to the NAND Flash control.

The HSMC embeds a NAND Flash Controller (NFC). The NFC can handle automatic transfers, sending the commands and address cycles to the NAND Flash and transferring the contents of the page (for read and write) to the NFC SRAM. It minimizes the processor overhead.

In order to improve overall system performance, the DATA phase of the transfer can be DMA-assisted. The static memory embeds the NAND Flash Error Correcting Code controller with the following features:

- Algorithm based on BCH codes
- Supports also SLC 1-bit (BCH 2-bit), SLC 4-bit (BCH 4-bit)
- Programmable Error Correcting Capability
  - 2-bit, 4-bit, 8-bit and 16-bit errors for 512 bytes/sector (4-Kbyte page)
  - 24-bit error for 1024 bytes/sector (8-Kbyte page)
- Programmable sector size: 512 bytes or 1024 bytes
- Programmable number of sectors per page: 1, 2, 4 or 8 blocks of data per page
- Programmable spare area size
- Supports spare area ECC protection
- Supports 8-Kbyte page size using 1024 bytes/sector and 4-Kbyte page size using 512 bytes/sector
- Error detection is interrupt-driven
- Provides hardware acceleration for error location
- Finds roots of error-locator polynomial
- Programmable number of roots

8.2.4 DDR and SDMMC I/Os Calibration

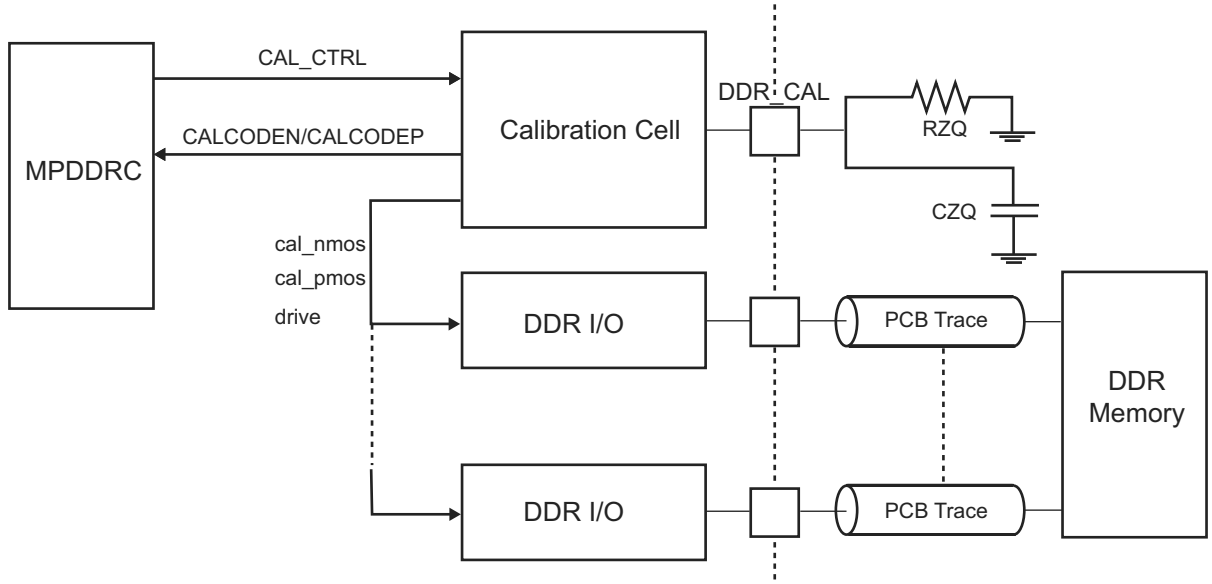
8.2.4.1 DDR I/O Calibration

The DDR2/DDR3/LPDDR1/LPDDR2/LPDDR3/DDR3L I/Os embed an automatic impedance matching control to avoid overshoots and reach the best performance levels depending on the bus load and external memories. A serial termination connection scheme, where the driver has an output impedance matched to the characteristic impedance of the line, is used to improve signal quality and reduce EMI.

One specific analog input, DDR_CAL, is used to calibrate all DDR / IOs.

The MPDDRC supports the ZQ calibration procedure used to calibrate the SAMA5D2 DDR I/O drive strength and the commands to setup the external DDR device drive strength (refer to Section 36. “Multiport DDR-SDRAM Controller (MPDDRC)”). The calibration cell supports all the memory types listed above.

Figure 8-2: DDR Calibration Cell



The calibration cell provides an input pin, DDR_CAL, loaded with one of the following resistor RZQ values:

15. Debug and Test Features

15.1 Description

The device features a number of complementary debug and test capabilities.

A common JTAG/ICE (In-Circuit Emulator) port is used for standard debugging functions, such as downloading code and single-stepping through programs.

A 2-pin debug port Serial Wire Debug (SWD) replaces the 5-pin JTAG port and provides an easy and risk-free alternative to JTAG as the two signals, SWDIO and SWCLK, are overlaid on the TMS and TCK pins, allowing for bi-modal devices that provide the other JTAG signals. These extra JTAG pins can be switched to other uses when in SWD mode.

A set of dedicated debug and test input/output pins gives direct access to these capabilities from a PC-based test environment.

15.2 Embedded Characteristics

- Cortex-A5 In-circuit Emulator
  - Two Real-time Watchpoint Units
  - Two Independent Registers: Debug Control Register and Debug Status Register
  - Test Access Port Accessible through JTAG Protocol
  - Debug Communications Channel
  - Serial Wire Debug
  - Trace
- Chip ID Register
- IEEE1149.1 JTAG Boundary-scan on All Digital Pins
- ETM, ETB: 8-Kbyte Embedded Trace Buffer

Figure 18-4 shows a Scalable Securable slave example. This example is constructed with the following hypothesis:

- The slave is an external memory with dedicated slave containing four regions, for example an external DDR.
- The slave size is 512 Mbytes.
- The slave base address equals 0x40000000. It is connected to a 256-Mbyte external memory.
- As the connected memory size is 256 Mbytes and there are four regions, the size of each region is 64 Mbytes. This gives the value of the slave region Max Size and Top Size. The slave region 0 Top Size must be configured to 64 Mbytes.
- The slave software configuration is:
  - TOP0 is set to 64 Mbytes
  - SPLIT0 is set to 4 Kbytes
  - SPLIT1 is set to 64 Mbytes, so its low area occupies the whole region 1
  - SPLIT2 is set to 4 Kbytes
  - SPLIT3 is set to 32 Mbytes
  - LANSECH0 is set to 0, the low area of region 0 is the Securable one
  - RDNSECH0 is set to 1, region 0 Securable area is non-secured for reads
  - WRNSECH0 is set to 0, region 0 Securable area is secured for writes
  - LANSECH1 is set to 1, the low area of region 1 is the non-securable one
  - RDNSECH1 is ‘don’t care’ since the low area occupies the whole region 1
  - WRNSECH1 is ‘don’t care’ since the low area occupies the whole region 1
  - LANSECH2 is set to 1, the low area of region 2 is the non-securable one
  - RDNSECH2 is set to 0, region 2 Securable area is secured for reads
  - WRNSECH2 is set to 0, region 2 Securable area is secured for writes
  - LANSECH3 is set to 0, the low area of region 3 is the Securable one
  - RDNSECH3 is set to 0, region 3 Securable area is secured for reads
  - WRNSECH3 is set to 0, region 3 Securable area is secured for writes

20.3.4 SFRBU RXLP Pull-Up Control Register

Name: SFRBU_RXLPPUCR

Address: 0xFC05C014

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
–	–	–	–	–	–	–	RXDPUCTRL

RXDPUCTRL: RXLP RXD Pull-Up Control

0 (Reset value): Pull-up enabled on RXD IO.

1: Pull-up disabled on RXD IO.

Note: If the RXLP is not used, it is recommended to enable the pull-up to avoid power consumption on VDDBU rail.

25.5.4 PIT Image Register

Name: PIT_PIIR

Address: 0xF804803C

Access: Read-only

31	30	29	28	27	26	25	24
PICNT							

23	22	21	20	19	18	17	16
PICNT				CPIV			

15	14	13	12	11	10	9	8
CPIV							

7	6	5	4	3	2	1	0
CPIV							

CPIV: Current Periodic Interval Value

Returns the current value of the periodic interval timer.

PICNT: Periodic Interval Counter

Returns the number of occurrences of periodic intervals since the last read of PIT_PIVR.

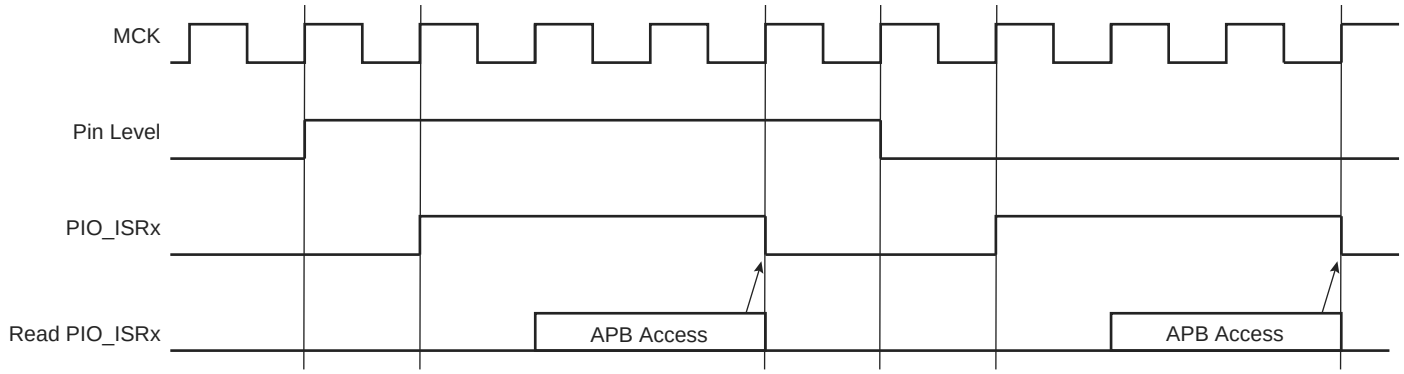
30.5.4 Register Write Protection

To prevent any single software error from corrupting RXLP behavior, certain registers in the address space can be write-protected by setting the WPEN bit in the RXLP Write Protection Mode Register (RXLP_WPMR).

The following registers can be write-protected:

- RXLP Mode Register
- RXLP Baud Rate Generator Register
- RXLP Comparison Register

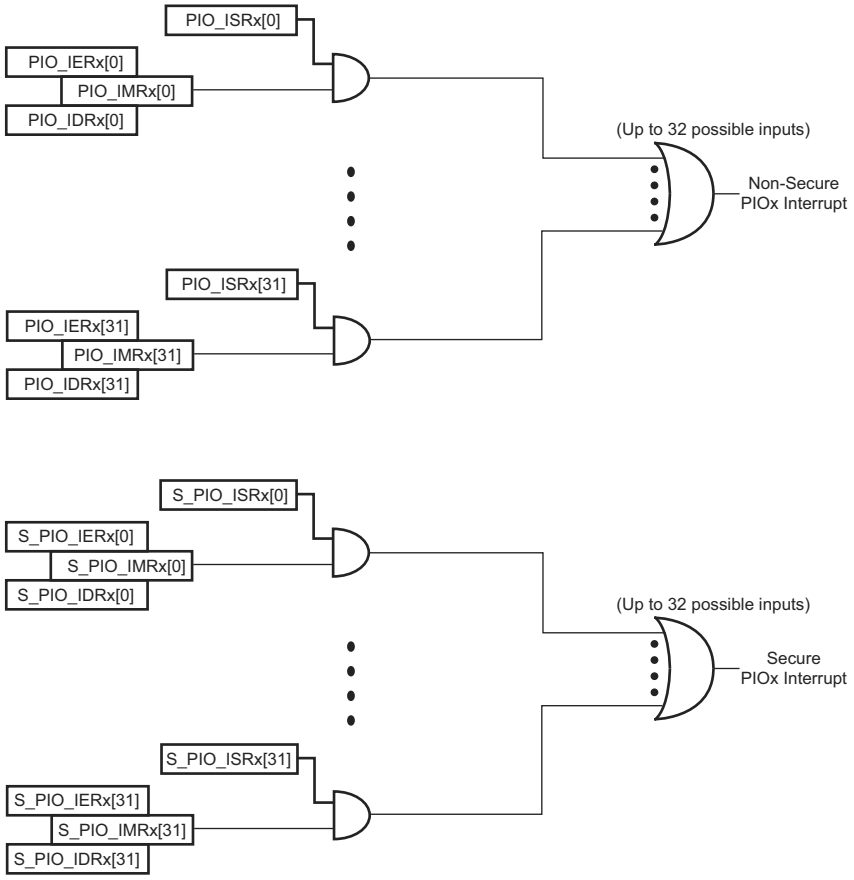
Figure 34-7: Input Change Interrupt Timings When No Additional Interrupt Modes



34.5.11 Interrupt Management

The PIO Controller can drive one secure interrupt signal and one non-secure interrupt signal per I/O group (see Figure 34-1). Secure interrupt signals are connected to the secure interrupt controller of the system. Non-secure interrupt signals are connected to the non-secure interrupt controller of the system.

Figure 34-8: PIO Interrupt Management



39.7.2 LCD Controller Configuration Register 1

Name: LCDC_LCDCFG1

Address: 0xF0000004

Access: Read/Write

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

23	22	21	20	19	18	17	16
VSPW							

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

7	6	5	4	3	2	1	0
HSPW							

HSPW: Horizontal Synchronization Pulse Width

Width of the LCDHSYNC pulse, given in pixel clock cycles. Width is (HSPW+1) LCDPCLK cycles.

VSPW: Vertical Synchronization Pulse Width

Width of the LCDVSYNC pulse, given in number of lines. Width is (VSPW+1) lines.

CLUTMODE: Color Lookup Table Mode Input Selection

Value	Name	Description
0	CLUT_1BPP	Color Lookup Table mode set to 1 bit per pixel
1	CLUT_2BPP	Color Lookup Table mode set to 2 bits per pixel
2	CLUT_4BPP	Color Lookup Table mode set to 4 bits per pixel
3	CLUT_8BPP	Color Lookup Table mode set to 8 bits per pixel

39.7.81 High-End Overlay Interrupt Mask Register

Name: LCDC_HEOIMR

Address: 0xF0000354

Access: Read-only

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

23	22	21	20	19	18	17	16
–	VOVR	VDONE	VADD	VDSCR	VDMA	–	–

15	14	13	12	11	10	9	8
–	UOVR	UDONE	UADD	UDSCR	UDMA	–	–

7	6	5	4	3	2	1	0
–	OVR	DONE	ADD	DSCR	DMA	–	–

DMA: End of DMA Transfer Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

DSCR: Descriptor Loaded Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

ADD: Head Descriptor Loaded Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

DONE: End of List Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

OVR: Overflow Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

UDMA: End of DMA Transfer for U or UV Chrominance Component Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

UDSCR: Descriptor Loaded for U or UV Chrominance Component Interrupt Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

UADD: Head Descriptor Loaded for U or UV Chrominance Component Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

UDONE: End of List for U or UV Chrominance Component Mask

0: Interrupt source is disabled

1: Interrupt source is enabled

Table 40-4: Receive Buffer Descriptor Entry (Continued)

Bit	Function
13	This bit has a different meaning depending on whether jumbo frames and ignore FCS modes are enabled. If neither mode is enabled this bit will be zero. **With jumbo frame mode enabled:** (bit 3 set in Network Configuration Register) Additional bit for length of frame (bit[13]), that is concatenated with bits[12:0] **With ignore FCS mode enabled and jumbo frames disabled:** (bit 26 set in Network Configuration Register and bit 3 clear in Network Configuration Register) This indicates per frame FCS status as follows: 0: Frame had good FCS 1: Frame had bad FCS, but was copied to memory as ignore FCS enabled.
12:0	These bits represent the length of the received frame which may or may not include FCS depending on whether FCS discard mode is enabled. **With FCS discard mode disabled:** (bit 17 clear in Network Configuration Register) Least significant 12 bits for length of frame including FCS. If jumbo frames are enabled, these 12 bits are concatenated with bit[13] of the descriptor above. **With FCS discard mode enabled:** (bit 17 set in Network Configuration Register) Least significant 12 bits for length of frame excluding FCS. If jumbo frames are enabled, these 12 bits are concatenated with bit[13] of the descriptor above.

To receive frames, the AHB buffer descriptors must be initialized by writing an appropriate address to bits 31:2 in the first word of each list entry. Bit 0 must be written with zero. Bit 1 is the wrap bit and indicates the last entry in the buffer descriptor list.

The start location of the receive buffer descriptor list must be written with the receive buffer queue base address before reception is enabled (receive enable in the Network Control register). Once reception is enabled, any writes to the Receive Buffer Queue Base Address register are ignored. When read, it will return the current pointer position in the descriptor list, though this is only valid and stable when receive is disabled.

If the filter block indicates that a frame should be copied to memory, the receive data DMA operation starts writing data into the receive buffer. If an error occurs, the buffer is recovered.

An internal counter within the GMAC represents the receive buffer queue pointer and it is not visible through the CPU interface. The receive buffer queue pointer increments by two words after each buffer has been used. It re-initializes to the receive buffer queue base address if any descriptor has its wrap bit set.

As receive AHB buffers are used, the receive AHB buffer manager sets bit zero of the first word of the descriptor to logic one indicating the AHB buffer has been used.

Software should search through the “used” bits in the AHB buffer descriptors to find out how many frames have been received, checking the start of frame and end of frame bits.

When the DMA is configured in the packet buffer Partial Store And Forward mode, received frames are written out to the AHB buffers as soon as enough frame data exists in the packet buffer. For both cases, this may mean several full AHB buffers are used before some error conditions can be detected. If a receive error is detected the receive buffer currently being written will be recovered. Previous buffers will not be recovered. As an example, when receiving frames with cyclic redundancy check (CRC) errors or excessive length, it is possible that a frame fragment might be stored in a sequence of AHB receive buffers. Software can detect this by looking for start of frame bit set in a buffer following a buffer with no end of frame bit set.

To function properly, a 10/100 Ethernet system should have no excessive length frames or frames greater than 128 bytes with CRC errors. Collision fragments will be less than 128 bytes long, therefore it will be a rare occurrence to find a frame fragment in a receive AHB buffer, when using the default value of 128 bytes for the receive buffers size.

When in packet buffer full store and forward mode, only good received frames are written out of the DMA, so no fragments will exist in the AHB buffers due to MAC receiver errors. There is still the possibility of fragments due to DMA errors, for example used bit read on the second buffer of a multi-buffer frame.

40.8.58 GMAC 1024 to 1518 Byte Frames Transmitted Register

Name: GMAC_TBFT1518

Address: 0xF800812C

Access: Read-only

31	30	29	28	27	26	25	24
NFTX							

23	22	21	20	19	18	17	16
NFTX							

15	14	13	12	11	10	9	8
NFTX							

7	6	5	4	3	2	1	0
NFTX							

NFTX: 1024 to 1518 Byte Frames Transmitted without Error

This register counts the number of 1024 to 1518 byte frames successfully transmitted without error, i.e., no underrun and not too many retries.

45.9.9 SSC Receive Synchronization Holding Register

Name: SSC_RSHR

Address: 0xF8004030 (0), 0xFC004030 (1)

Access: Read-only

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

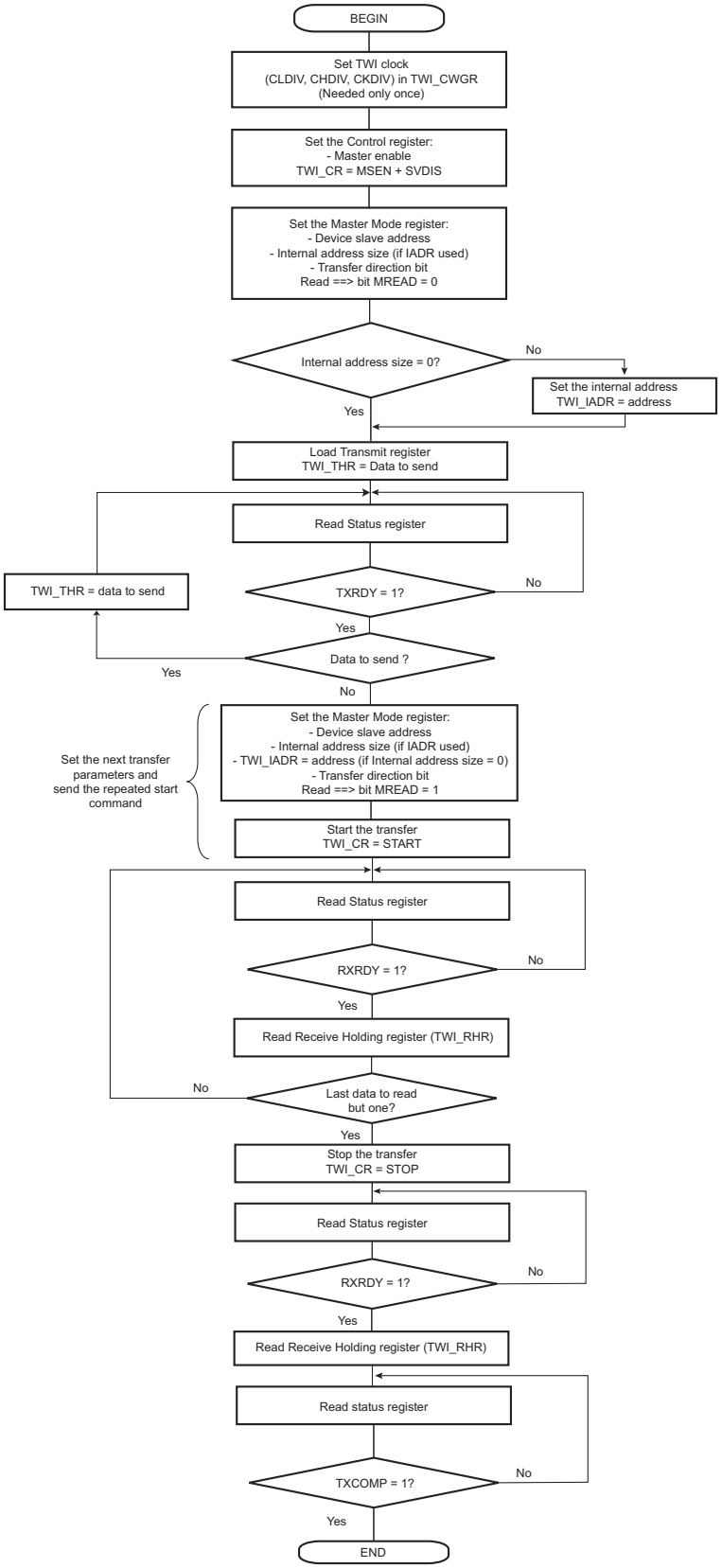
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–

15	14	13	12	11	10	9	8
RSDAT							

7	6	5	4	3	2	1	0
RSDAT							

RSDAT: Receive Synchronization Data

Figure 46-19: TWIHS Write Operation with Multiple Data Bytes and Read Operation with Multiple Data Bytes (Sr)



NACK used in Slave Read mode:

0: Each data byte has been correctly received by the master.

1: In Read mode, a data byte has not been acknowledged by the master. When NACK is set, the user must not fill TWIHS_THR even if TXRDY is set, because it means that the master stops the data transfer or re-initiate it.

Note: in Slave Write mode, all data are acknowledged by the TWIHS.

ARBLST: Arbitration Lost (cleared on read)

This bit is used in Master mode only.

0: Arbitration won.

1: Arbitration lost. Another master of the TWIHS bus has won the multimaster arbitration. TXCOMP is set at the same time.

SCLWS: Clock Wait State

This bit is used in Slave mode only.

0: The clock is not stretched.

1: The clock is stretched. TWIHS_THR / TWIHS_RHR buffer is not filled / emptied before the transmission / reception of a new character.

SCLWS behavior can be seen in Figure 46-39 and Figure 46-40.

EOSACC: End Of Slave Access (cleared on read)

This bit is used in Slave mode only.

0: A slave access is being performing.

1: The Slave Access is finished. End Of Slave Access is automatically set as soon as SVACC is reset.

EOSACC behavior can be seen in Figure 46-41 and Figure 46-42.

MCACK: Master Code Acknowledge (cleared on read)

MACK used in Slave mode:

0: No Master Code has been received since the last read of TWIHS_SR.

1: A Master Code has been received since the last read of TWIHS_SR.

TOUT: Timeout Error (cleared on read)

0: No SMBus timeout occurred since the last read of TWIHS_SR.

1: An SMBus timeout occurred since the last read of TWIHS_SR.

PECERR: PEC Error (cleared on read)

0: No SMBus PEC error occurred since the last read of TWIHS_SR.

1: An SMBus PEC error occurred since the last read of TWIHS_SR.

SMBDAM: SMBus Default Address Match (cleared on read)

0: No SMBus Default Address received since the last read of TWIHS_SR.

1: An SMBus Default Address was received since the last read of TWIHS_SR.

SMBHHM: SMBus Host Header Address Match (cleared on read)

0: No SMBus Host Header Address received since the last read of TWIHS_SR.

1: An SMBus Host Header Address was received since the last read of TWIHS_SR.

50.6.5 QSPI Serial Memory Mode

In Serial Memory mode, the QSPI acts as a serial Flash memory controller. The QSPI can be used to read data from the serial Flash memory allowing the CPU to execute code from it (XIP execute in place). The QSPI can also be used to control the serial Flash memory (Program, Erase, Lock, etc.) by sending specific commands. In this mode, the QSPI is compatible with single-bit SPI, Dual SPI and Quad SPI protocols.

To activate this mode, QSPI_MR.SMM must be written to ‘1’.

In Serial Memory mode, data is transferred either by QSPI_TDR and QSPI_RDR or by writing or read in the QSPI memory space (0x9000_00000-0x9800_00000/0XD000_0000--0XD800_0000) depending on TFRTYP and SMRM configuration.

50.6.5.1 Instruction Frame

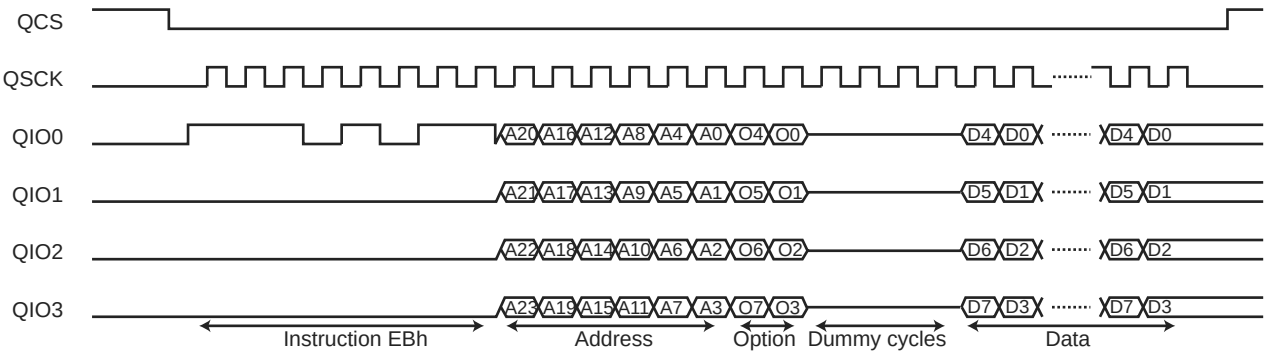
In order to control serial Flash memories, the QSPI is able to send instructions via the SPI bus (ex: READ, PROGRAM, ERASE, LOCK, etc.). Because the instruction set implemented in serial Flash memories is memory vendor dependant, the QSPI includes a complete Instruction Frame register (QSPI_IFR), which makes it very flexible and compatible with all serial Flash memories.

An instruction frame includes:

- An instruction code (size: 8 bits). The instruction is optional in some cases (see **Section 50.6.5.4 “Continuous Read Mode”**).
- An address (size: 24 bits or 32 bits). The address is optional but is required by instructions such as READ, PROGRAM, ERASE, LOCK. By default the address is 24 bits long, but it can be 32 bits long to support serial Flash memories larger than 128 Mbits (16 Mbytes).
- An option code (size: 1/2/4/8 bits). The option code is not required, but it is useful to activate the XIP mode or the Continuous Read mode (see **Section 50.6.5.4 “Continuous Read Mode”**) for READ instructions, in some serial Flash memory devices. These modes improve the data read latency.
- Dummy cycles. Dummy cycles are optional but required by some READ instructions.
- Data bytes are optional. Data bytes are present for data transfer instructions such as READ or PROGRAM.

The instruction code, the address/option and the data can be sent with Single-bit SPI, Dual SPI or Quad SPI protocols.

Figure 50-8: Instruction Frame



50.6.5.2 Instruction Frame Transmission

To send an instruction frame, the user must first configure the address to send by writing the field ADDR in the Instruction Address register (QSPI_IAR). This step is required if the instruction frame includes an address and no data. When data is present, the address of the instruction is defined by the address of the data accesses in the QSPI memory space, not by QSPI_IAR.

If the instruction frame includes the instruction code and/or the option code, the user must configure the instruction code and/or the option code to send by writing the fields INST and OPT in the Instruction Code register (QSPI_ICR).

Then, the user must write QSPI_IFR to configure the instruction frame depending on which instruction must be sent. If the instruction frame does not include data, writing in this register triggers the send of the instruction frame in the QSPI. If the instruction frame includes data, the send of the instruction frame is triggered by the first data access in the QSPI memory space.

The instruction frame is configured by the following bits and fields of QSPI_IFR:

- WIDTH field—used to configure which data lanes are used to send the instruction code, the address, the option code and to transfer the data. It is possible to use two unidirectional data lanes (MISO-MOSI Single-bit SPI), two bidirectional data lanes (QIO0-QIO1 Dual SPI) or four bidirectional data lanes (QIO0–QIO3 Quad SPI).
- INSTEN bit—used to enable the send of an instruction code.
- ADDREN bit—used to enable the send of an address after the instruction code.
- OPTEN bit—used to enable the send of an option code after the address.

56.7.10 PWM DMA Register

Name: PWM_DMAR

Address: 0xF802C024

Access: Write- only

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

23	22	21	20	19	18	17	16
DMADUTY							

15	14	13	12	11	10	9	8
DMADUTY							

7	6	5	4	3	2	1	0
DMADUTY							

Only the first 16 bits (channel counter size) are significant.

DMADUTY: Duty-Cycle Holding Register for DMA Access

Each write access to PWM_DMAR sequentially updates the CDTY field of PWM_CDTYx with DMADUTY (only for channel configured as synchronous). See “Method 3: Automatic write of duty-cycle values and automatic trigger of the update” .

OPMOD: Operating Mode

Value	Name	Description
0	ECB	Electronic Code Book mode
1	CBC	Cipher Block Chaining mode
2	OFB	Output Feedback mode
3	CFB	Cipher Feedback mode

For CBC-MAC operating mode, set OPMOD to CBC and LOD to 1.

LOD: Last Output Data Mode

0: No effect.

After each end of encryption/decryption, the output data is available either on the output data registers (Manual and Auto modes) .

In Manual and Auto modes, the DATRDY flag is cleared when at least one of the Output Data registers is read.

1: The DATRDY flag is cleared when at least one of the Input Data Registers is written.

No more Output Data Register reads are necessary between consecutive encryptions/decryptions (see Section 62.4.3 “Last Output Data Mode”).

Warning: In DMA mode, reading to the Output Data registers before the last data encryption/decryption process may lead to unpredictable result.

CFBS: Cipher Feedback Data Size

Value	Name	Description
0	SIZE_64BIT	64-bit
1	SIZE_32BIT	32-bit
2	SIZE_16BIT	16-bit
3	SIZE_8BIT	8-bit