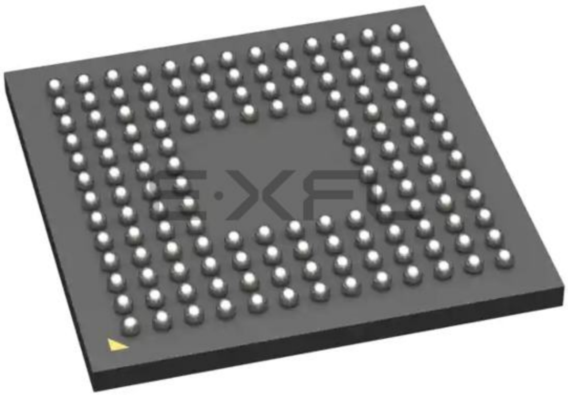


Welcome to E-XFL.COM

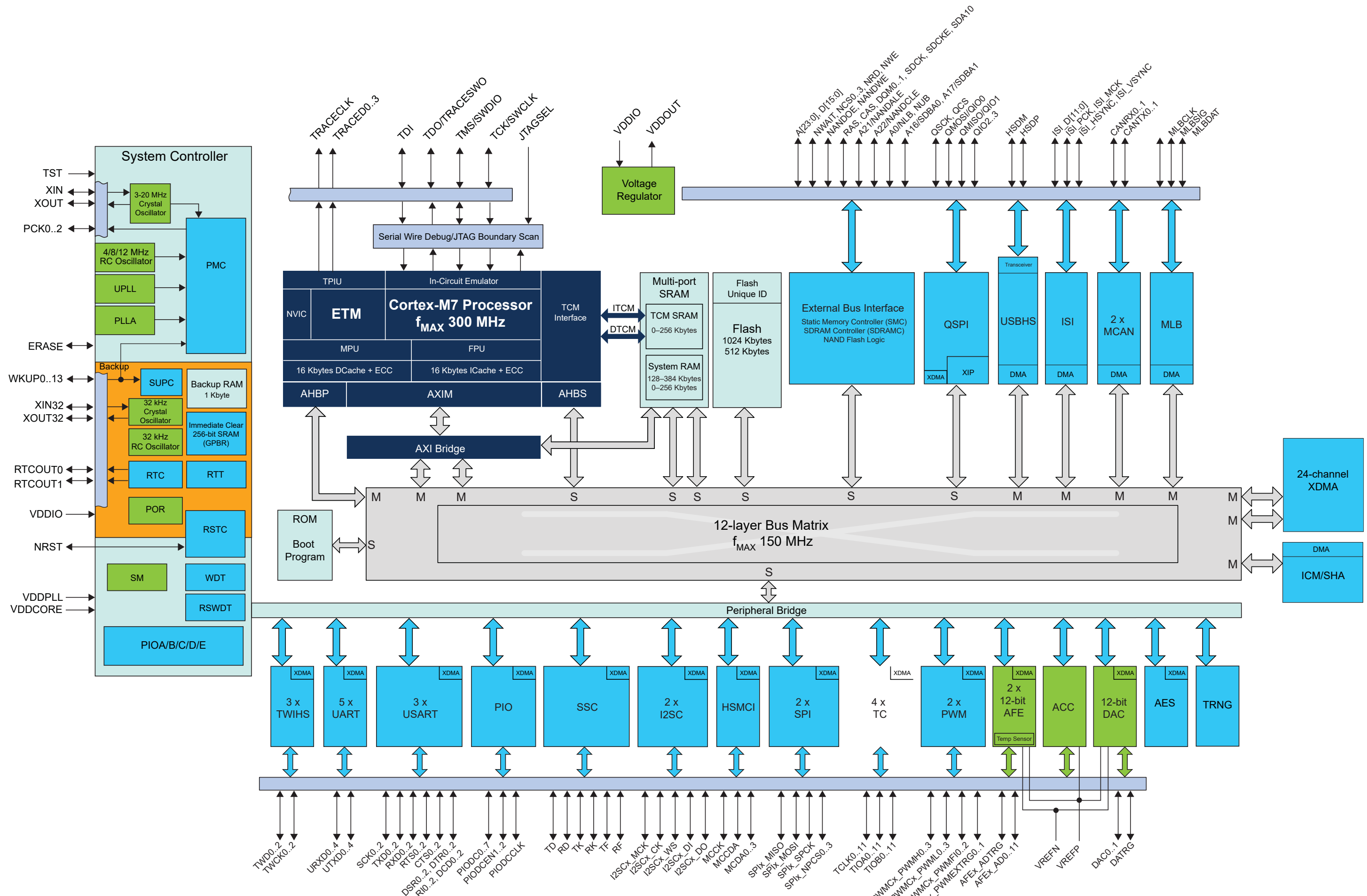
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	ARM® Cortex®-M7
Core Size	32-Bit Single-Core
Speed	300MHz
Connectivity	EBI/EMI, I²C, IrDA, LINbus, MMC/SD/SDIO, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	114
Program Memory Size	2MB (2M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384K x 8
Voltage - Supply (Vcc/Vdd)	1.08V ~ 3.6V
Data Converters	A/D 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	144-UFBGA
Supplier Device Package	144-UFBGA (6x6)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70q21a-cfnt

Figure 3-3. SAM V70 144-pin Block Diagram



21.3 Register Summary

Offset	Name	Bit Pos.								
0x00	CHIPID_CIDR	7:0	EPROC[2:0]			VERSION[4:0]				
		15:8	NVPSIZ2[3:0]				NVPSIZ[3:0]			
		23:16	ARCH[3:0]				SRAMSIZ[3:0]			
		31:24	EXT	NVPTYP[2:0]			ARCH[7:4]			
0x04	CHIPID_EXID	7:0	EXID[7:0]							
		15:8	EXID[15:8]							
		23:16	EXID[23:16]							
		31:24	EXID[31:24]							

32.6.1.10 PIO Set Output Data Register

Name: PIO_SODR
Offset: 0x0030
Property: Write-only

Bit	31	30	29	28	27	26	25	24
	P31	P30	P29	P28	P27	P26	P25	P24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	P23	P22	P21	P20	P19	P18	P17	P16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	P15	P14	P13	P12	P11	P10	P9	P8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
	P7	P6	P5	P4	P3	P2	P1	P0
Access								
Reset								

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 – P PIO Set Output Data

Value	Description
0	No effect.
1	Sets the data to be driven on the I/O line.

34.5.2 I/O Lines

The pins used for interfacing the SDRAMC may be multiplexed with the PIO lines. The programmer must first program the PIO controller to assign the SDRAMC pins to their peripheral function. If I/O lines of the SDRAMC are not used by the application, they can be used for other purposes by the PIO Controller.

34.5.3 Power Management

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

The SDRAM clock on pin SDCK is output as soon as the first access to the SDRAM is made during the initialization phase. To stop the SDRAM clock signal, the SDRAMC_LPR must be programmed with the self-refresh command.

34.5.4 Interrupt Sources

The SDRAMC interrupt (Refresh Error notification) is connected to the memory controller. This interrupt may be ORed with other system peripheral interrupt lines and is finally provided as the system interrupt source (Source 1) to the interrupt controller.

Using the SDRAMC interrupt requires the interrupt controller to be programmed first.

34.6 Functional Description

34.6.1 SDRAM Controller Write Cycle

The SDRAMC allows burst access or single access. In both cases, the SDRAMC keeps track of the active row in each bank, thus maximizing performance. To initiate a burst access, the SDRAMC uses the transfer type signal provided by the master requesting the access. If the next access is a sequential write access, writing to the SDRAM device is carried out. If the next access is a write-sequential access, but the current access is to a boundary page, or if the next access is in another row, then the SDRAMC generates a precharge command, activates the new row and initiates a write command. To comply with SDRAM timing parameters, additional clock cycles are inserted between precharge and active commands (t_{RP}), and between active and write commands (t_{RCD}). For definition of these timing parameters, refer to the SDRAMC Configuration Register. Refer to the following figure.

Value	Name	Description
		an “All Banks Precharge” command must be issued. To activate this mode, the command must be followed by a write to the SDRAM.
5	EXT_LOAD_MODEREG	The SDRAMC issues an “Extended Load Mode Register” command when the SDRAM device is accessed regardless of the cycle. To activate this mode, the “Extended Load Mode Register” command must be followed by a write to the SDRAM. The write in the SDRAM must be done in the appropriate bank; most low-power SDRAM devices use the bank 1.
6	DEEP_POWERDOWN	Deep Powerdown mode. Enters Deep Powerdown mode.

Offset	Name	Bit Pos.								
		23:16	SA[23:16]							
		31:24	SA[31:24]							
0x03E4	XDMAC_CDA14	7:0	DA[7:0]							
		15:8	DA[15:8]							
		23:16	DA[23:16]							
		31:24	DA[31:24]							
0x03E8	XDMAC_CNDA14	7:0	NDA[5:0]							NDAIF
		15:8	NDA[13:6]							
		23:16	NDA[21:14]							
		31:24	NDA[29:22]							
0x03EC	XDMAC_CNDC14	7:0				NDVIEW[1:0]		NDDUP	NDSUP	NDE
		15:8								
		23:16								
		31:24								
0x03F0	XDMAC_CUBC14	7:0	UBLEN[7:0]							
		15:8	UBLEN[15:8]							
		23:16	UBLEN[23:16]							
		31:24								
0x03F4	XDMAC_CBC14	7:0	BLEN[7:0]							
		15:8					BLEN[11:8]			
		23:16								
		31:24								
0x03F8	XDMAC_CC14	7:0	MEMSET	SWREQ		DSYNC		MBSIZE[1:0]		TYPE
		15:8		DIF	SIF	DWIDTH[1:0]		CSIZE[2:0]		
		23:16	WRIP	RDIP	INITD		DAM[1:0]		SAM[1:0]	
		31:24		PERID[6:0]						
0x03FC	XDMAC_CDS_MSP14	7:0	SDS_MSP[7:0]							
		15:8	SDS_MSP[15:8]							
		23:16	DDS_MSP[7:0]							
		31:24	DDS_MSP[15:8]							
0x0400	XDMAC_CSUS14	7:0	SUBS[7:0]							
		15:8	SUBS[15:8]							
		23:16	SUBS[23:16]							
		31:24								
0x0404	XDMAC_CDUS14	7:0	DUBS[7:0]							
		15:8	DUBS[15:8]							
		23:16	DUBS[23:16]							
		31:24								
0x0408 ... 0x040F	Reserved									
0x0410	XDMAC_CIE15	7:0		ROIE	WBIE	RBIE	FIE	DIE	LIE	BIE
		15:8								
		23:16								
		31:24								
0x0414	XDMAC_CID15	7:0		ROID	WBEID	RBEID	FID	DID	LID	BID

36.9.6 XDMAC Global Interrupt Mask Register

Name: XDMAC_GIM
Offset: 0x14
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23 – IM XDMAC Channel x Interrupt Mask

Value	Description
0	This bit indicates that the channel x interrupt source is masked. The interrupt line is not raised.
1	This bit indicates that the channel x interrupt source is unmasked.

37. Image Sensor Interface (ISI)

37.1 Description

The Image Sensor Interface (ISI) connects a CMOS-type image sensor to the processor and provides image capture in various formats. The ISI performs data conversion, if necessary, before the storage in memory through DMA.

The ISI supports color CMOS image sensor and grayscale image sensors with a reduced set of functionalities.

In Grayscale mode, the data stream is stored in memory without any processing and so is not compatible with the LCD controller.

Internal FIFOs on the preview and codec paths are used to store the incoming data. The RGB output on the preview path is compatible with the LCD controller. This module outputs the data in RGB format (LCD compatible) and has scaling capabilities to make it compliant to the LCD display resolution (see the table RGB Format in Default Mode, RGB_CFG = 00, No Swap).

Several input formats such as preprocessed RGB or YCbCr are supported through the data bus interface.

The ISI supports two synchronization modes:

- Hardware with ISI_VSYNC and ISI_HSYNC signals
- International Telecommunication Union Recommendation ITU-R BT.656-4 Start-of-Active-Video (SAV) and End-of-Active-Video (EAV) synchronization sequence

Using EAV/SAV for synchronization reduces the pin count (ISI_VSYNC, ISI_HSYNC not used). The polarity of the synchronization pulse is programmable to comply with the sensor signals.

Table 37-1. I/O Description

Signal	Direction	Description
ISI_VSYNC	In	Vertical Synchronization
ISI_HSYNC	In	Horizontal Synchronization
ISI_DATA[11..0]	In	Sensor Pixel Data
ISI_MCK	Out	Master Clock provided to the Image Sensor. Refer to “Clocks”.
ISI_PCK	In	Pixel Clock provided by the Image Sensor

38.8.48 GMAC 512 to 1023 Byte Frames Transmitted Register

Name: GMAC_TBFT1023
Offset: 0x128
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
	NFTX[31:24]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
Bit	23	22	21	20	19	18	17	16
	NFTX[23:16]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8
	NFTX[15:8]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
Bit	7	6	5	4	3	2	1	0
	NFTX[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – NFTX[31:0] 512 to 1023 Byte Frames Transmitted without Error
This register counts the number of 512 to 1023 byte frames successfully transmitted without error, i.e., no underrun and not too many retries.

39.5.2.12 Management of IN Endpoints

Overview

IN packets are sent by the USB device controller upon IN requests from the host. All data which acknowledges or not the bank can be written when it is full.

The endpoint must be configured first.

The USBHS_DEVEPTISRx.TXINI bit is set at the same time as USBHS_DEVEPTIMRx.FIFOCON when the current bank is free. This triggers a PEP_x interrupt if the Transmitted IN Data Interrupt Enable (USBHS_DEVEPTIMRx.TXINE) bit is one.

USBHS_DEVEPTISRx.TXINI is cleared by software (by writing a one to the Transmitted IN Data Interrupt Clear bit (USBHS_DEVEPTIDRx.TXINIC) to acknowledge the interrupt, which has no effect on the endpoint FIFO.

The user then writes into the FIFO and writes a one to the FIFO Control Clear (USBHS_DEVEPTIDRx.FIFOCONC) bit to clear the USBHS_DEVEPTIMRx.FIFOCON bit. This allows the USBHS to send the data. If the IN endpoint is composed of multiple banks, this also switches to the next bank. The USBHS_DEVEPTISRx.TXINI and USBHS_DEVEPTIMRx.FIFOCON bits are updated in accordance with the status of the next bank.

USBHS_DEVEPTISRx.TXINI is always cleared before clearing USBHS_DEVEPTIMRx.FIFOCON.

The USBHS_DEVEPTISRx.RWALL bit is set when the current bank is not full, i.e., when the software can write further data into the FIFO.

Figure 39-11. Example of an IN Endpoint with one Data Bank

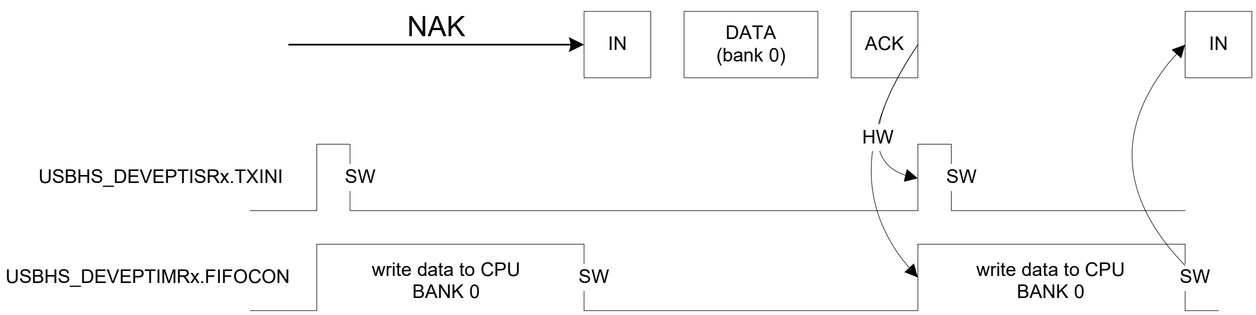
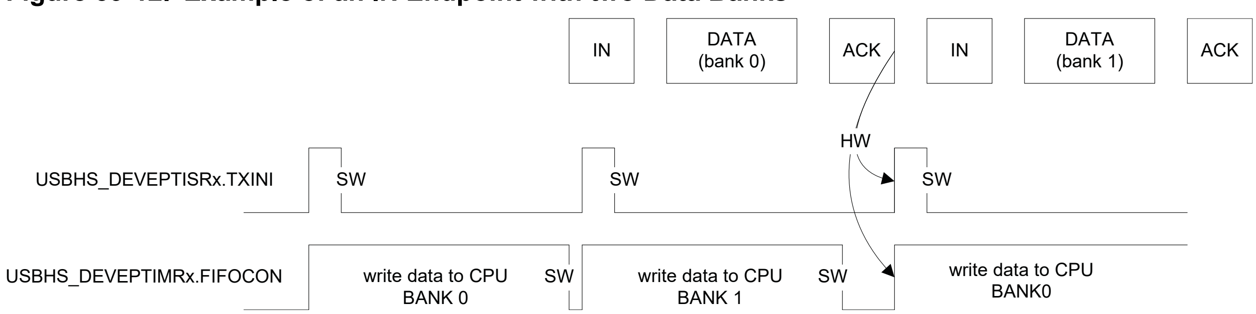


Figure 39-12. Example of an IN Endpoint with two Data Banks



Detailed Description

The data is written as follows:

- When the bank is empty, USBHS_DEVEPTISRx.TXINI and USBHS_DEVEPTIMRx.FIFOCON are set, which triggers a PEP_x interrupt if USBHS_DEVEPTIMRx.TXINE = 1.
- The user acknowledges the interrupt by clearing USBHS_DEVEPTISRx.TXINI.

Value	Name	Description
6	BOR	Boot Operation Request. Start a boot operation mode, the host processor can read boot data from the MMC device directly.
7	EBO	End Boot Operation. This command allows the host processor to terminate the boot operation mode.

Bits 7:6 – RSPTYP[1:0] Response Type

Value	Name	Description
0	NORESP	No response
1	48_BIT	48-bit response
2	136_BIT	136-bit response
3	R1B	R1b response type

Bits 5:0 – CMDNB[5:0] Command Number
This is the command index.

Value	Description
0	As soon as data is written in SPI_TDR.
1	SPI_TDR and internal shift register are empty. If a transfer delay has been defined, TXEMPTY is set after the end of this delay.

Bit 8 – NSSR NSS Rising (cleared on read)

Value	Description
0	No rising edge detected on NSS pin since the last read of SPI_SR.
1	A rising edge occurred on NSS pin since the last read of SPI_SR.

Bit 3 – OVRES Overrun Error Status (cleared on read)
An overrun occurs when SPI_RDR is loaded at least twice from the internal shift register since the last read of SPI_RDR.

Value	Description
0	No overrun has been detected since the last read of SPI_SR.
1	An overrun has occurred since the last read of SPI_SR.

Bit 2 – MODF Mode Fault Error (cleared on read)

Value	Description
0	No mode fault has been detected since the last read of SPI_SR.
1	A mode fault occurred since the last read of SPI_SR.

Bit 1 – TDRE Transmit Data Register Empty (cleared by writing SPI_TDR)
0: Data has been written to SPI_TDR and not yet transferred to the internal shift register.

1: The last data written in SPI_TDR has been transferred to the internal shift register.

TDRE is cleared when the SPI is disabled or at reset. Enabling the SPI sets the TDRE flag.

Bit 0 – RDRF Receive Data Register Full (cleared by reading SPI_RDR)
0: No data has been received since the last read of SPI_RDR.

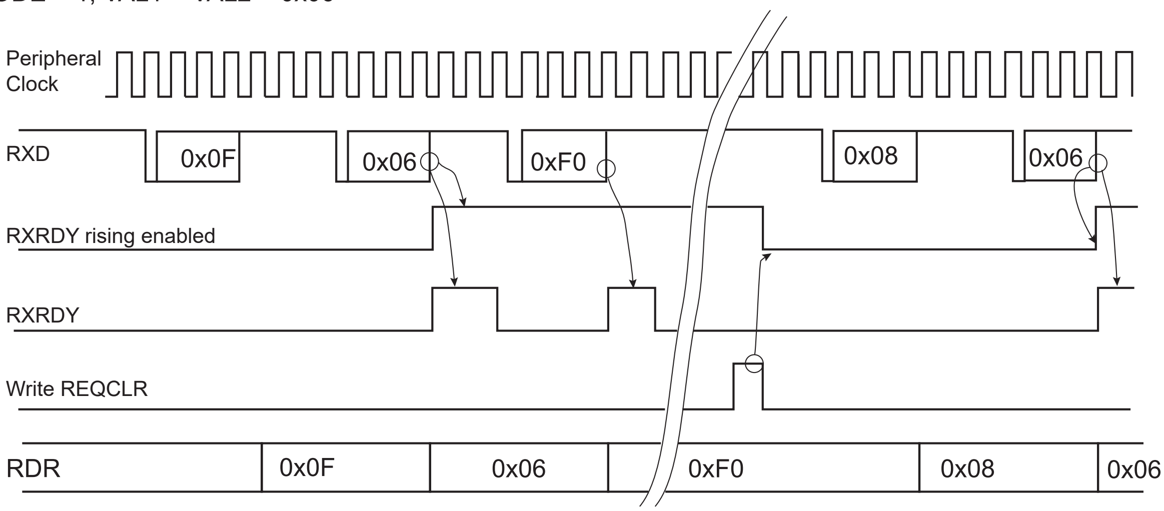
1: Data has been received and the received data has been transferred from the internal shift register to SPI_RDR since the last read of SPI_RDR.

- If VAL1 is strictly higher than VAL2, then the flag CMP is set to 1 if either received character equals VAL1 or VAL2.

By programming the CMPMODE bit to 1, the comparison function result triggers the start of the loading of UART_RHR (see the figure below). The trigger condition occurs as soon as the received character value matches the condition defined by the programming of VAL1, VAL2 and CMPPAR in UART_CMPR. The comparison trigger event can be restarted by writing a one to the REQCLR bit in UART_CR.

Figure 47-11. Receive Holding Register Management

CMPMODE = 1, VAL1 = VAL2 = 0x06



47.5.6 Asynchronous and Partial Wake-up (SleepWalking)

Asynchronous and partial wake-up (SleepWalking) is a means of data pre-processing that qualifies an incoming event, thus allowing the UART to decide whether or not to wake up the system. SleepWalking is used primarily when the system is in Wait mode (refer to section “Power Management Controller (PMC)”) but can also be enabled when the system is fully running.

No access must be performed in the UART between the enable of asynchronous partial wake-up and the wake-up performed by the UART.

If the system is in Wait mode and asynchronous and partial wake-up is enabled, the maximum baud rate that can be achieved equals 19200.

If the system is running or in Sleep mode, the maximum baud rate that can be achieved equals 115200 or higher. This limit is bounded by the peripheral clock frequency divided by 16.

The UART_RHR must be read before enabling asynchronous and partial wake-up.

When SleepWalking is enabled for the UART (refer to section “Power Management Controller (PMC)”), the PMC decodes a clock request from the UART. The request is generated as soon as there is a falling edge on the RXD line as this may indicate the beginning of a start bit. If the system is in Wait mode (processor and peripheral clocks switched off), the PMC restarts the fast RC oscillator and provides the clock only to the UART.

As soon as the clock is provided by the PMC, the UART processes the received frame and compares the received character with VAL1 and VAL2 in UART_CMPR (UART Comparison Register).

The UART instructs the PMC to disable the clock if the received character value does not meet the conditions defined by VAL1 and VAL2 fields in UART_CMPR (see Asynchronous Event Generating Only Partial Wake-up).

Value (see Note)	RxStatus	Description
		channel format or was out of sequence. Only allowed on control and asynchronous channels.
74h... 7Eh	rsvd	Reserved
System Responses (Rx Device response in System Channel):		
00h	DeviceNotPresent	
80h	DevicePresent	
82h	DeviceServiceRequest	Device response to DeviceAddress scan (MLBScan), where the scanned Device needs some or all its ChannelAddresses configured.
84h...FEh	rsvd	Reserved

Note: All odd values (LSB set) are reserved.

48.6.1.5 System Commands

The Controller sends out System commands in the physical channel associated with the FRAMESYNC MediaLB frame alignment ChannelAddress (PC0). The NoData command indicates no command exists on the System Channel for this frame. All System commands are optional and may or may not be implemented on the MediaLB Controller. Additionally, System responses (including dynamic configuration) are optional and may or may not be implemented on a specific MediaLB Device.

The MOSTLock and MOSTUnlock commands indicate the status of the Controller relative to the MOST Network. When the Controller is not locked to the MOST Network (MOSTUnlock), all MediaLB data being transferred to or from the MOST Network must also stop. Buffers in the Controller could delay the stopping point to beyond when MOSTUnlock shows up on MediaLB.

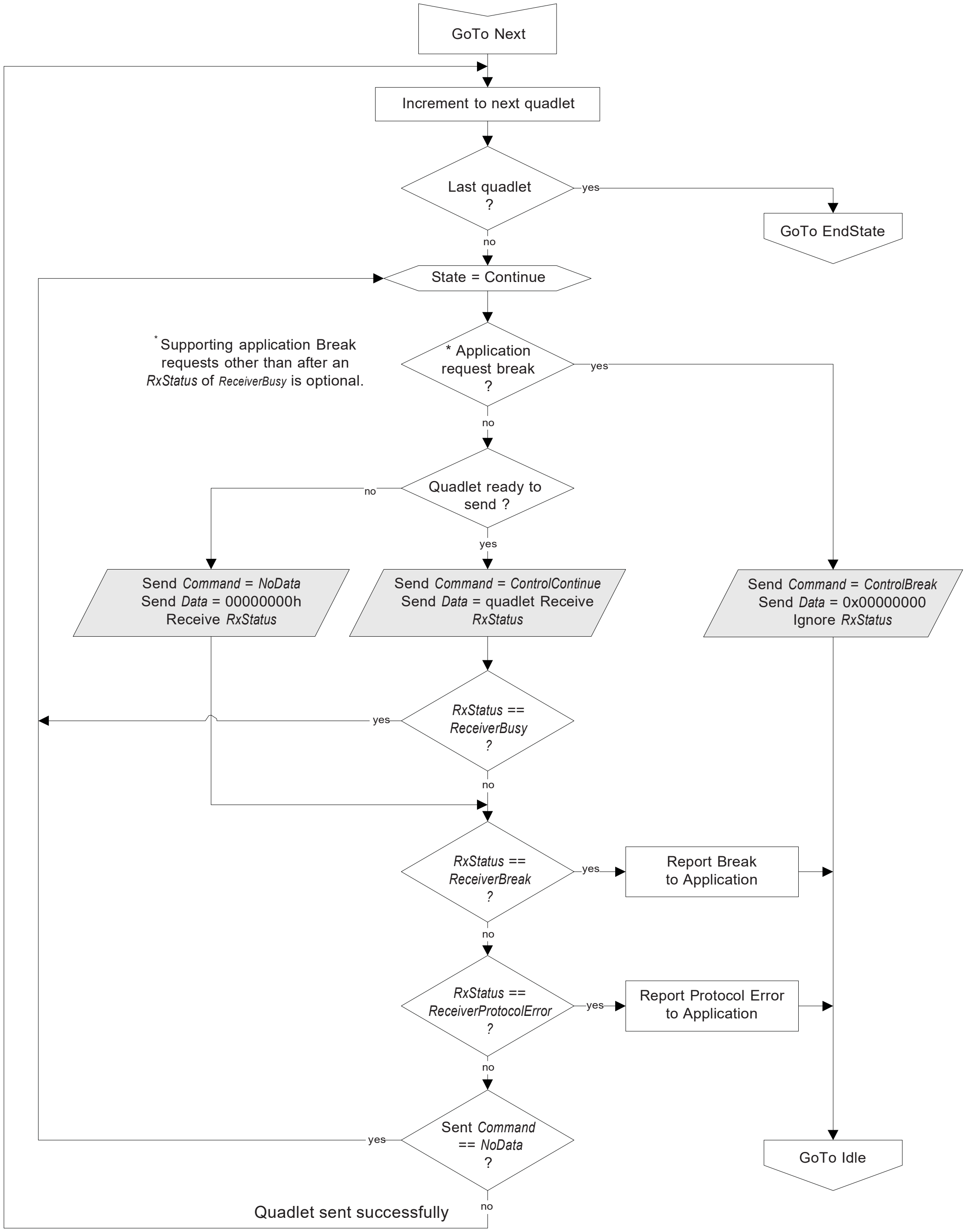
The MLBReset command is designed to place the MediaLB interface in one or all Devices in a known state. When a MediaLB Device receives the MLBReset command, it will look at the corresponding first two received (most significant) data bytes on the MLBD line:

- If the first two bytes are zero, then all MediaLB Devices must reset their MediaLB interface to an initialized known state (broadcast reset to all Devices).
- If the first two bytes match the local DeviceAddress, then only the Device with the matching DeviceAddress will reset its MediaLB interface to an initialized known state (reset targeted to only one Device).

The MLBSubCmd command is used for configuration and status information from the Controller to Devices. A sub-command is contained in the first byte of the MLBD quadlet. When MediaLB Device interfaces receive the MLBSubCmd command, they will store the command and corresponding data quadlet (sub-command). Currently, only one sub-command is defined (scSetCA) and is used in dynamic configuration.

MediaLB Devices and ChannelAddresses can be configured using two methods: static or dynamic. When the EHC MediaLB Device uses the dynamic method, it instructs the Controller to scan for other MediaLB

Figure 48-7. Control Packet Tx Device Protocol: Middle



Bit 18 – TOOL Timeout Occurred Interrupt Line

Bit 17 – MRAFL Message RAM Access Failure Interrupt Line

Bit 16 – TSWL Timestamp Wraparound Interrupt Line

Bit 15 – TEFLL Tx Event FIFO Event Lost Interrupt Line

Bit 14 – TEFFL Tx Event FIFO Full Interrupt Line

Bit 13 – TEFWL Tx Event FIFO Watermark Reached Interrupt Line

Bit 12 – TEFNL Tx Event FIFO New Entry Interrupt Line

Bit 11 – TFEL Tx FIFO Empty Interrupt Line

Bit 10 – TCFL Transmission Cancellation Finished Interrupt Line

Bit 9 – TCL Transmission Completed Interrupt Line

Bit 8 – HPML High Priority Message Interrupt Line

Bit 7 – RF1LL Receive FIFO 1 Message Lost Interrupt Line

Bit 6 – RF1FL Receive FIFO 1 Full Interrupt Line

Bit 5 – RF1WL Receive FIFO 1 Watermark Reached Interrupt Line

Bit 4 – RF1NL Receive FIFO 1 New Message Interrupt Line

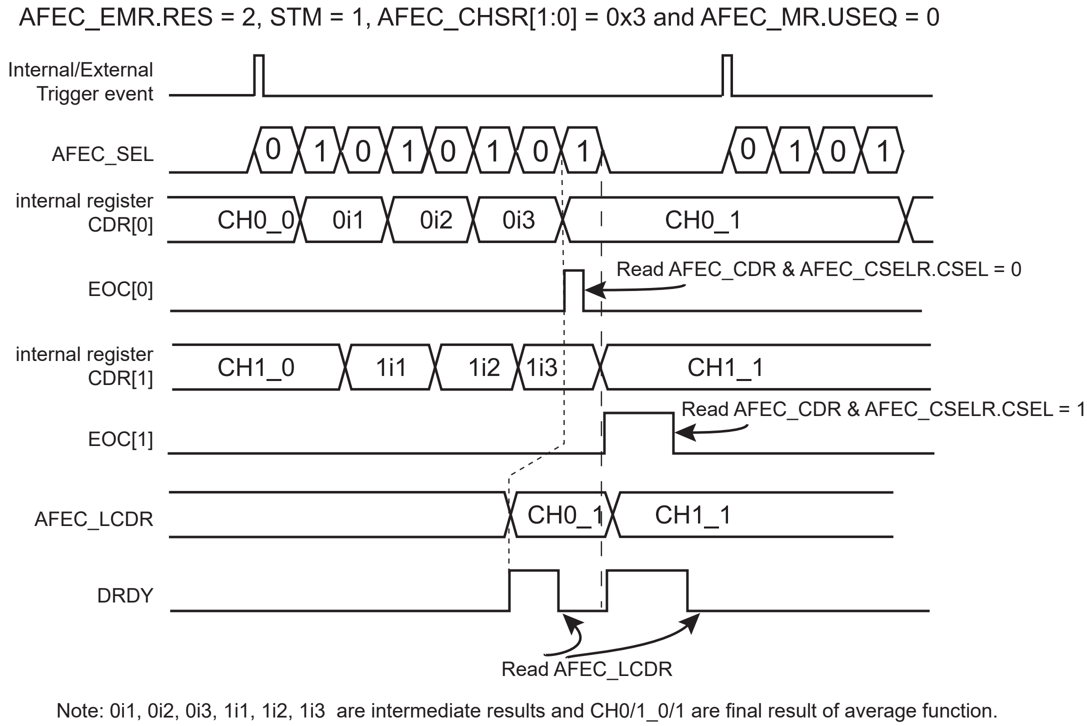
Bit 3 – RF0LL Receive FIFO 0 Message Lost Interrupt Line

Bit 2 – RF0FL Receive FIFO 0 Full Interrupt Line

Bit 1 – RF0WL Receive FIFO 0 Watermark Reached Interrupt Line

Bit 0 – RF0NL Receive FIFO 0 New Message Interrupt Line

Figure 52-12. Digital Averaging Function Waveforms on a Single Trigger Event



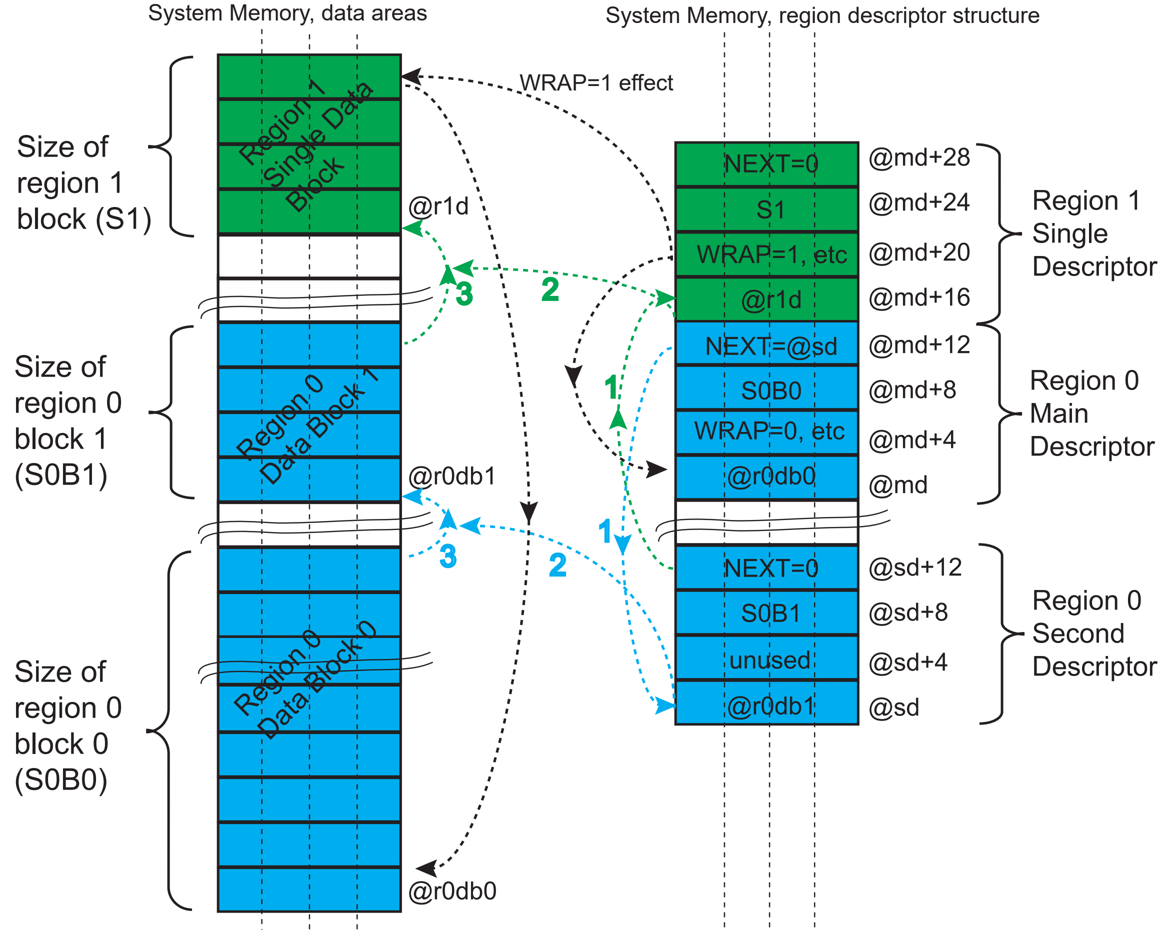
When USEQ is set, the user can define the channel sequence to be converted by configuring AFEC_SEQxR and AFEC_CHER so that channels are not interleaved during the averaging period. Under these conditions, a sample is defined for each end of conversion as described in the figure below.

Therefore, if the same channel is configured to be converted four times consecutively and AFEC_EMR.RES = 2, the averaging result is placed in the corresponding channel internal data register (read by means of the AFEC_CDR) and the AFEC_LCDR for each trigger event.

In this case, the AFE real sample rate remains the maximum AFE sample rate divided by 4.

When USEQ is set and the RES field enables the Enhanced Resolution mode, it is important to note that the user sequence must be a sequence being an integer multiple of 4 (i.e., the number of the enabled channel in the Channel Status register (AFEC_CHSR) must be an integer multiple of 4 and the AFEC_SEQxR must be a series of 4 times the same channel index).

Figure 55-5. Example: Monitoring of 3 Memory Data Blocks (Defined as 2 Regions)



55.5.2 ICM Region Descriptor Structure

The ICM Region Descriptor Area is a contiguous area of system memory that the controller and the processor can access. When the ICM is activated, the controller performs a descriptor fetch operation at *(ICM_DSCR) address. If the Main List contains more than one descriptor (i.e., more than one region is to be monitored), the fetch address is *(ICM_DSCR) + (RID<<4) where RID is the region identifier.

Table 55-1. Region Descriptor Structure (Main List)

Offset	Structure Member	Name
ICM_DSCR+0x000+RID*(0x10)	ICM Region Start Address	ICM_RADDR
ICM_DSCR+0x004+RID*(0x10)	ICM Region Configuration	ICM_RCFG
ICM_DSCR+0x008+RID*(0x10)	ICM Region Control	ICM_RCTRL
ICM_DSCR+0x00C+RID*(0x10)	ICM Region Next Address	ICM_RNEXT

56.6.1 TRNG Control Register

Name: TRNG_CR
Offset: 0x00
Reset: –
Property: Write-only

Bit	31	30	29	28	27	26	25	24
	WAKEY[23:16]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	WAKEY[15:8]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	WAKEY[7:0]							
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	–

Bit	7	6	5	4	3	2	1	0
								ENABLE
Access								W
Reset								–

Bits 31:8 – WAKEY[23:0] Register Write Access Key

Value	Name	Description
0x524E47	PASSWD	Writing any other value in this field aborts the write operation.

Bit 0 – ENABLE Enables the TRNG to Provide Random Values

Value	Description
0	Disables the TRNG.
1	Enables the TRNG if 0x524E47 (“RNG” in ASCII) is written in KEY field at the same time.

Symbol	VDDIO Supply	1.8V Domain	3.3V Domain	1.8V Domain	3.3V Domain	Unit
	Parameter	Min		Max		
SMC_{26}	NCS High to Data Out, A0–A25, change	NCS_WR_HOLD × t_{CPMCK} - 4.4	NCS_WR_HOLD × t_{CPMCK} - 3.4	—	—	ns
SMC_{27}	NCS High to NWE Inactive	(NCS_WR_HOLD - NWE_HOLD) × t_{CPMCK} - 2.8	(NCS_WR_HOLD - NWE_HOLD) × t_{CPMCK} - 2.4	—	—	ns

Timings are given in the 3.3V domain, with VDDIO from 2.85V to 3.6V, maximum external capacitor = 50 pF.

Timings are given assuming a capacitance load on data, control and address pads.

In the tables that follow, t_{CPMCK} is MCK period.

58.13.1.9.4 Write Timings

Table 58-64. SMC Write Signals - NWE Controlled (WRITE_MODE = 1)

Symbol		VDDIO Supply		Unit
	Parameter	Min	Max	
HOLD or NO HOLD Settings (NWE_HOLD ≠ 0, NWE_HOLD = 0)				
SMC_{15}	Data Out Valid before NWE High	NWE_PULSE × t_{CPMCK} - 4.6	–	ns
SMC_{16}	NWE Pulse Width	NWE_PULSE × t_{CPMCK} - 0.3	–	ns
SMC_{17}	A0–A22 valid before NWE low	NWE_SETUP × t_{CPMCK} - 4.2	–	ns
SMC_{18}	NCS low before NWE high	(NWE_SETUP - NCS_RD_SETUP + NWE_PULSE) × t_{CPMCK} - 2.2	–	ns
HOLD Settings (NWE_HOLD ≠ 0)				
SMC_{19}	NWE High to Data OUT, NBS0/A0 NBS1, NBS2/A1, NBS3, A2–A25 change	NWE_HOLD × t_{CPMCK} - 3.9	–	ns
SMC_{20}	NWE High to NCS Inactive [(1)]	(NWE_HOLD - NCS_WR_HOLD) × t_{CPMCK} - 3.6	–	ns
NO HOLD Settings (NWE_HOLD = 0)				
SMC_{21}	NWE High to Data OUT, NBS0/A0 NBS1, NBS2/A1, NBS3, A2–A25, NCS change[(1)]	1.5	–	ns

Note:
Hold length = total cycle duration - setup duration - pulse duration. “hold length” is for “NCS_WR_HOLD length” or “NWE_HOLD length”