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	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	75
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 10x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70n21b-an

4. Signal Description

The following table provides details on signal names classified by peripheral.

Table 4-1. Signal Description List

Signal Name	Function	Type	Active Level	Voltage Reference	Comments
Power Supplies					
VDDIO	Peripherals I/O Lines Power Supply	Power	–	–	–
VDDIN	Voltage Regulator Input, AFE, DAC and Analog Comparator Power Supply (see **Note**)	Power	–	–	–
VDDOUT	Voltage Regulator Output	Power	–	–	–
VDDPLL	PLLA Power Supply	Power	–	–	–
VDDPLLUSB	USB PLL and Oscillator Power Supply	Power	–	–	–
VDDCORE	Powers the core, the embedded memories and the peripherals	Power	–	–	–
GND, GNDPLL, GNDPLLUSB, GNDANA, GNDUTMI	Ground	Ground	–	–	–
VDDUTMII	USB Transceiver Power Supply	Power	–	–	–
VDDUTMIC	USB Core Power Supply	Power	–	–	–
GNDUTMI	USB Ground	Ground	–	–	–
Clocks, Oscillators and PLLs					
XIN	Main Oscillator Input	Input	–	VDDIO	–
XOUT	Main Oscillator Output	Output	–		–
XIN32	Slow Clock Oscillator Input	Input	–		–
XOUT32	Slow Clock Oscillator Output	Output	–		–

8. Input/Output Lines

The SAM E70/S70/V70/V71 features both general purpose I/Os (GPIO) and system I/Os. GPIOs can have alternate functionality due to multiplexing capabilities of the PIO controllers. The same PIO line can be used, whether in I/O mode or by the multiplexed peripherals. System I/Os include pins such as test pins, oscillators, erase or analog inputs.

8.1 General-Purpose I/O Lines

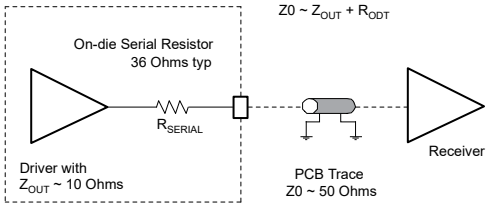
General-purpose (GPIO) lines are managed by PIO Controllers. All I/Os have several input or output modes such as pullup or pulldown, input Schmitt triggers, multi-drive (open-drain), glitch filters, debouncing or input change interrupt. Programming of these modes is performed independently for each I/O line through the PIO controller user interface. For more details, refer to 32. Parallel Input/Output Controller (PIO).

The input/output buffers of the PIO lines are supplied through VDDIO power supply rail.

The SAM E70/S70/V70/V71 embeds high-speed pads able to handle the high-speed clocks for HSMCI, SPI and QSPI (MCK/2). Refer to 58. Electrical Characteristics for SAM V70/V71 for more details. Typical pullup and pulldown value is 100 kΩ for all I/Os.

Each I/O line also embeds a R_{SERIAL} (On-die Serial Resistor), as shown in the following figure. It consists of an internal series resistor termination scheme for impedance matching between the driver output (SAM E70/S70/V70/V71) and the PCB trace impedance preventing signal reflection. The series resistor helps to reduce I/Os switching current (di/dt). thereby reducing in turn, EMI. It also decreases overshoot and undershoot (ringing) due to inductance of interconnect between devices or between boards. Finally, R_{SERIAL} helps diminish signal integrity issues.

Figure 8-1. On-Die Termination (ODT)



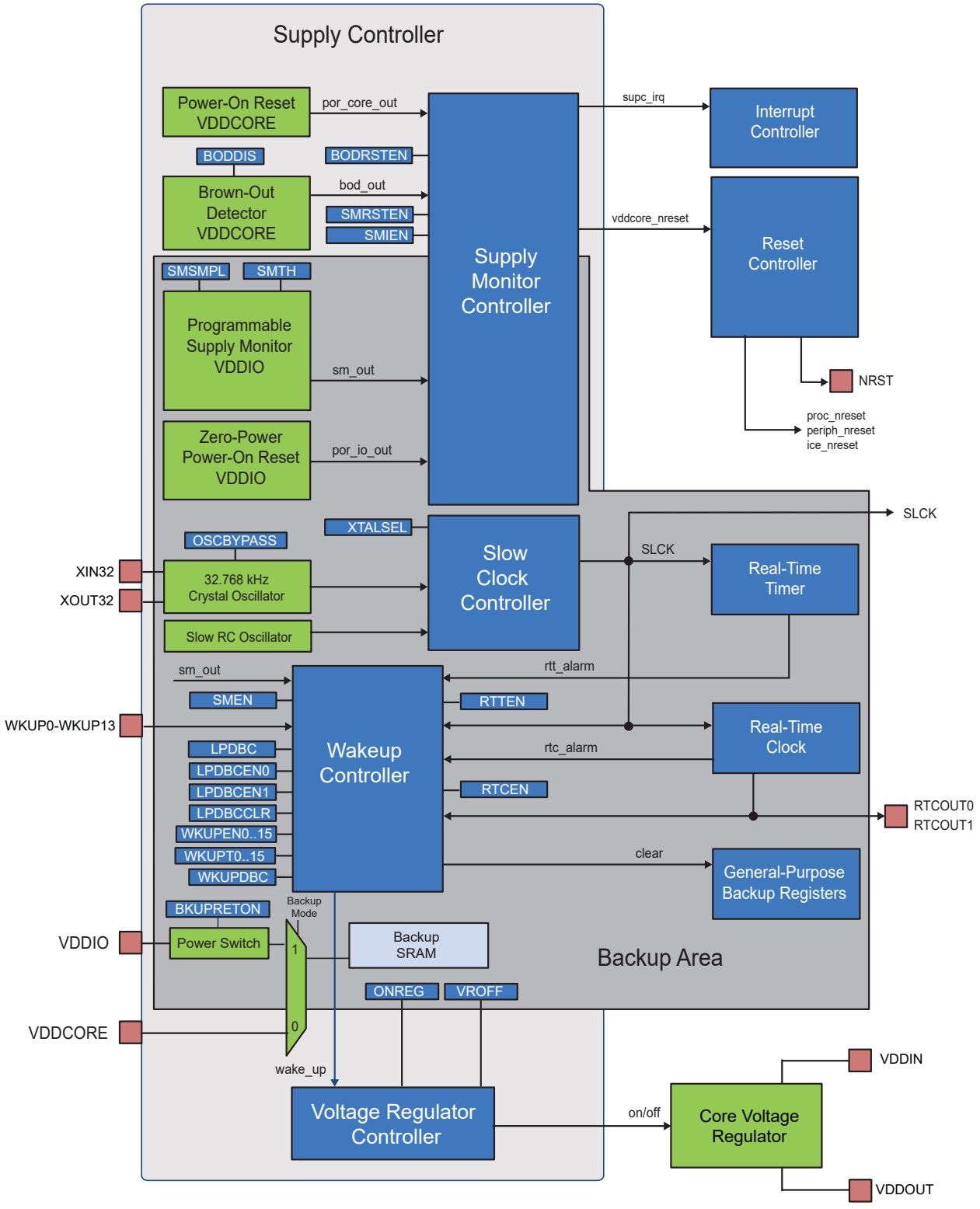
8.2 System I/O Lines

System I/O lines are pins used by oscillators, Test mode, reset, JTAG and other features. The following table lists the SAM E70/S70/V70/V71 system I/O lines shared with PIO lines.

These pins are software-configurable as general-purpose I/Os or system pins. At startup, the default function of these pins is always used.

23.3 Block Diagram

Figure 23-1. Supply Controller Block Diagram



24.5.2 Watchdog Timer Mode Register

Name: WDT_MR
Offset: 0x04
Reset: 0x3FFF2FFF
Property: Read/Write Once

The first write access prevents any further modification of the value of this register. Read accesses remain possible.

The WDT_MR register values must not be modified within three slow clock periods following a restart of the watchdog performed by a write access in WDT_CR. Any modification will cause the watchdog to trigger an end of period earlier than expected.

Bit	31	30	29	28	27	26	25	24
			WDIDLEHLT	WDDBGHLT	WDD[11:8]			
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			1	1	1	1	1	1

Bit	23	22	21	20	19	18	17	16
	WDD[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
	WDDIS		WDRSTEN	WDFIEN	WDV[11:8]			
Access	R/W		R/W	R/W	R/W	R/W	R/W	R/W
Reset	0		1	0	1	1	1	1

Bit	7	6	5	4	3	2	1	0
	WDV[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Bit 29 – WDIDLEHLT Watchdog Idle Halt

Value	Description
0	The watchdog runs when the system is in idle state.
1	The watchdog stops when the system is in idle state.

Bit 28 – WDDBGHLT Watchdog Debug Halt

Value	Description
0	The watchdog runs when the processor is in debug state.
1	The watchdog stops when the processor is in debug state.

Bits 27:16 – WDD[11:0] Watchdog Delta Value
Defines the permitted range for reloading the Watchdog Timer.

If the Watchdog Timer value is less than or equal to WDD, setting bit WDT_CR.WDRSTT restarts the timer.

27.6.3 RTC Time Register

Name: RTC_TIMR
Offset: 0x08
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
		AMPM	HOUR[5:0]					
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		MIN[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		SEC[6:0]						
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

Bit 22 – AMPM Ante Meridiem Post Meridiem Indicator
This bit is the AM/PM indicator in 12-hour mode.

Value	Description
0	AM.
1	PM.

Bits 21:16 – HOUR[5:0] Current Hour
The range that can be set is 1–12 (BCD) in 12-hour mode or 0–23 (BCD) in 24-hour mode.

Bits 14:8 – MIN[6:0] Current Minute
The range that can be set is 0–59 (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

Bits 6:0 – SEC[6:0] Current Second
The range that can be set is 0–59 (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

28.5 Register Summary

Offset	Name	Bit Pos.								
0x00	RTT_MR	7:0	RTPRES[7:0]							
		15:8	RTPRES[15:8]							
		23:16				RTTDIS		RTTRST	RTTINCIEN	ALMIEN
		31:24								RTC1HZ
0x04	RTT_AR	7:0	ALMV[7:0]							
		15:8	ALMV[15:8]							
		23:16	ALMV[23:16]							
		31:24	ALMV[31:24]							
0x08	RTT_VR	7:0	CRTV[7:0]							
		15:8	CRTV[15:8]							
		23:16	CRTV[23:16]							
		31:24	CRTV[31:24]							
0x0C	RTT_SR	7:0							RTTINC	ALMS
		15:8								
		23:16								
		31:24								

32.6.1.26 PIO Input Filter Slow Clock Disable Register

Name: PIO_IFSCDR
Offset: 0x0080
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 Peripheral Clock Glitch Filtering Select

Value	Description
0	No effect.
1	The glitch filter is able to filter glitches with a duration < $t_{peripheral\ clock}/2$.

Offset	Name	Bit Pos.								
0x0400	GMAC_ISRPQ1	7:0	TCOMP	TFC	RLEX			RXUBR	RCOMP	
		15:8					HRESP	ROVR		
		23:16								
		31:24								
0x0404	GMAC_ISRPQ2	7:0	TCOMP	TFC	RLEX			RXUBR	RCOMP	
		15:8					HRESP	ROVR		
		23:16								
		31:24								
0x0408	GMAC_ISRPQ3	7:0	TCOMP	TFC	RLEX			RXUBR	RCOMP	
		15:8					HRESP	ROVR		
		23:16								
		31:24								
0x040C	GMAC_ISRPQ4	7:0	TCOMP	TFC	RLEX			RXUBR	RCOMP	
		15:8					HRESP	ROVR		
		23:16								
		31:24								
0x0410	GMAC_ISRPQ5	7:0	TCOMP	TFC	RLEX			RXUBR	RCOMP	
		15:8					HRESP	ROVR		
		23:16								
		31:24								
0x0414 ... 0x043F	Reserved									
0x0440	GMAC_TBQBAPQ1	7:0	TXBQBA[5:0]							
		15:8	TXBQBA[13:6]							
		23:16	TXBQBA[21:14]							
		31:24	TXBQBA[29:22]							
0x0444	GMAC_TBQBAPQ2	7:0	TXBQBA[5:0]							
		15:8	TXBQBA[13:6]							
		23:16	TXBQBA[21:14]							
		31:24	TXBQBA[29:22]							
0x0448	GMAC_TBQBAPQ3	7:0	TXBQBA[5:0]							
		15:8	TXBQBA[13:6]							
		23:16	TXBQBA[21:14]							
		31:24	TXBQBA[29:22]							
0x044C	GMAC_TBQBAPQ4	7:0	TXBQBA[5:0]							
		15:8	TXBQBA[13:6]							
		23:16	TXBQBA[21:14]							
		31:24	TXBQBA[29:22]							
0x0450	GMAC_TBQBAPQ5	7:0	TXBQBA[5:0]							
		15:8	TXBQBA[13:6]							
		23:16	TXBQBA[21:14]							
		31:24	TXBQBA[29:22]							
0x0454 ... 0x047F	Reserved									

Bit 1 – RXOUTIS Received OUT Data Interrupt Set

Bit 0 – TXINIS Transmitted IN Data Interrupt Set

40.14.2 HSMCI Mode Register

Name: HSMCI_MR
Offset: 0x04
Reset: 0x0
Property: Read/Write

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								0

Bit	15	14	13	12	11	10	9	8
Access								
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Access								
Reset	0	0	0	0	0	0	0	0

Bit 16 – CLKODD Clock divider is odd
This bit is the least significant bit of the clock divider and indicates the clock divider parity.

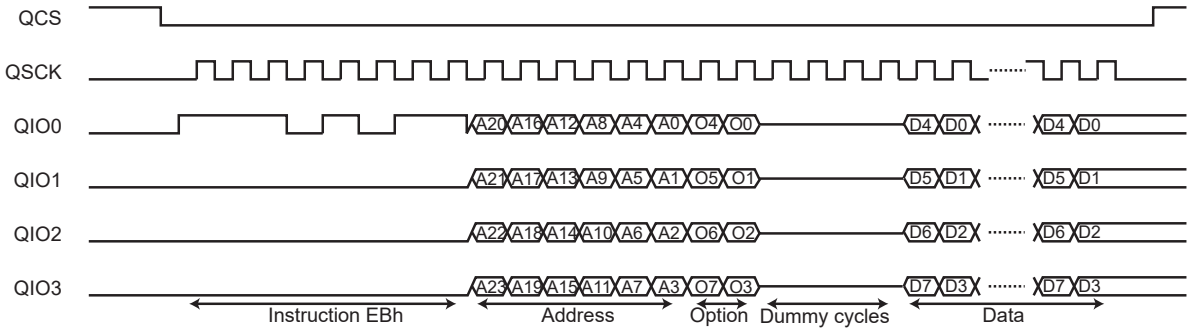
Bit 14 – PADV Padding Value
PADV may be only in manual transfer.

Value	Description
0	0x00 value is used when padding data in write transfer.
1	0xFF value is used when padding data in write transfer.

Bit 13 – FBYTE Force Byte Transfer
Enabling Force Byte Transfer allow byte transfers, so that transfer of blocks with a size different from modulo 4 can be supported.

WARNING BLKLEN value depends on FBYTE.

Figure 42-8. Instruction Frame



42.6.5.2 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.

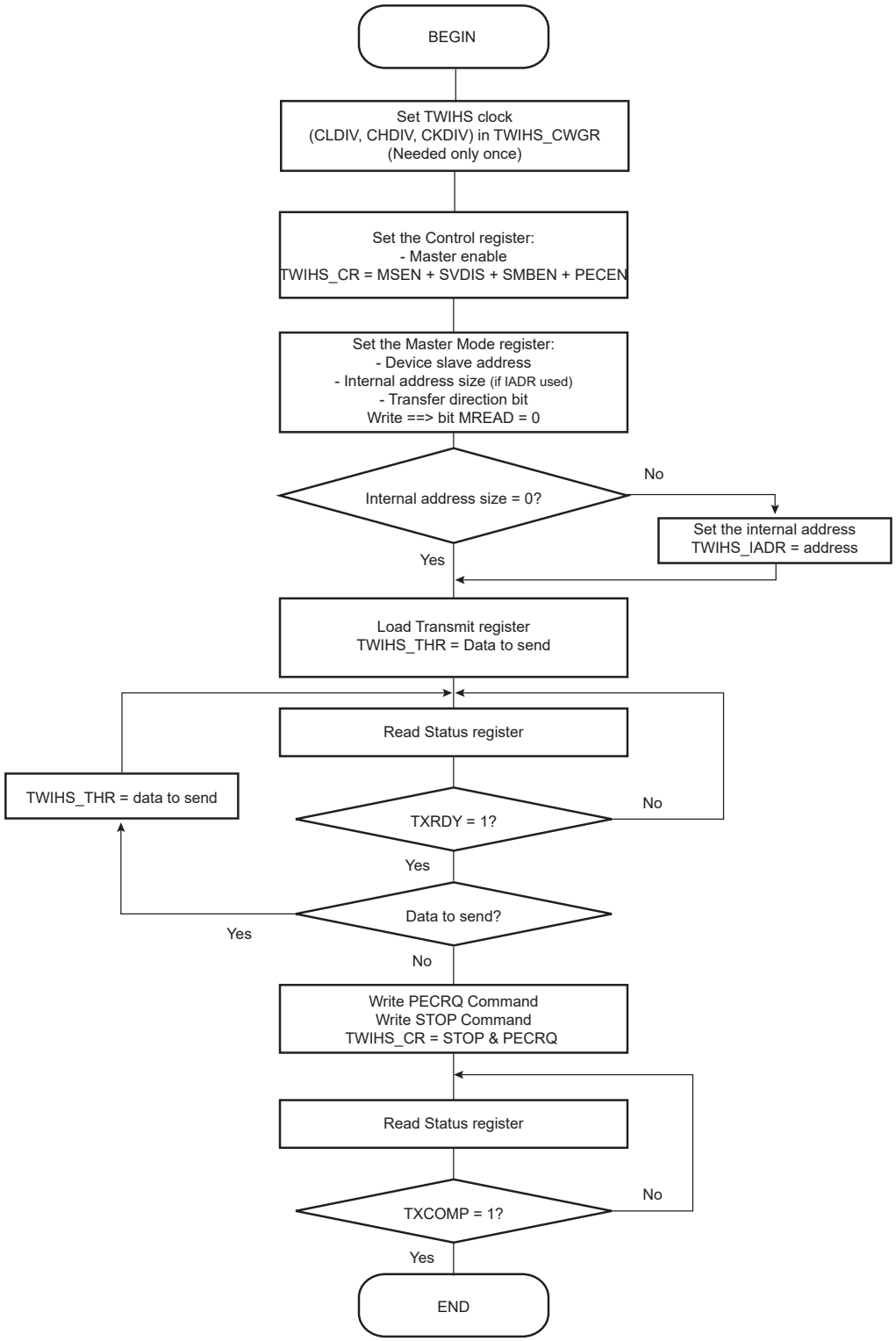
The instruction frame is configured by the following bits and fields of QSPI_IFR:

- 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.
- DATAEN bit—used to enable the transfer of data (READ or PROGRAM instruction).
- OPTL field—used to configure the option code length. The value written in OPTL must be consistent with the value written in the field WIDTH. For example: OPTL = 0 (1-bit option code) is not consistent with WIDTH = 6 (option code sent with QuadSPI protocol, thus the minimum length of the option code is 4 bits).
- ADDRL bit—used to configure the address length.
- TFRTYP field—used to define which type of data transfer must be performed.
- NBDUM field—used to configure the number of dummy cycles when reading data from the serial Flash memory. Between the address/option and the data, with some instructions, dummy cycles are inserted by the serial Flash memory.

Refer to 42.6.5.2 Instruction Frame Transmission.

If data transfer is enabled, the user can access the serial memory by reading or writing the QSPI memory space:

Figure 43-17. SMBus Write Operation with Multiple Data Bytes with or without Internal Address and PEC Sending



44.9.15 SSC Interrupt Disable Register

Name: SSC_IDR
Offset: 0x48
Reset: –
Property: Write-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
					RXSYN	TXSYN	CP1	CP0
Access					W	W	W	W
Reset					–	–	–	–

Bit	7	6	5	4	3	2	1	0
			OVRUN	RXRDY			TXEMPTY	TXRDY
Access			W	W			W	W
Reset			–	–			–	–

Bit 11 – RXSYN Rx Sync Interrupt Enable

Value	Description
0	No effect.
1	Disables the Rx Sync Interrupt.

Bit 10 – TXSYN Tx Sync Interrupt Enable

Value	Description
0	No effect.
1	Disables the Tx Sync Interrupt.

Bit 9 – CP1 Compare 1 Interrupt Disable

Value	Description
0	No effect.
1	Disables the Compare 1 Interrupt.

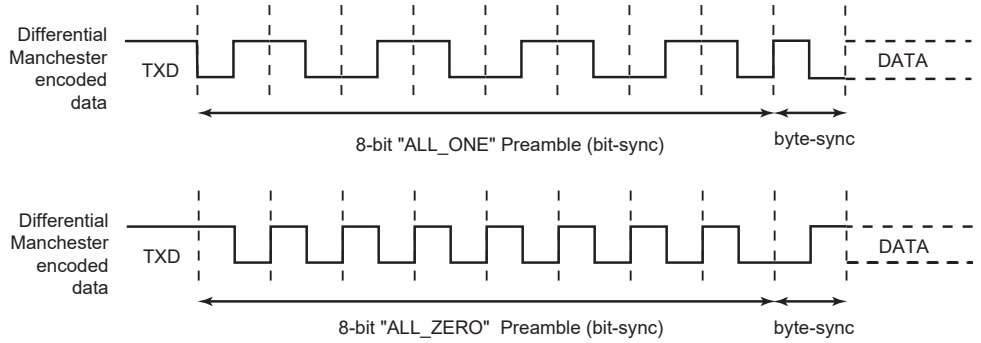
Bit 8 – CP0 Compare 0 Interrupt Disable

Value	Description
0	No effect.
1	Disables the Compare 0 Interrupt.

The LON implementation allows two different preamble patterns ALL_ONE and ALL_ZERO which can be configured via US_MAN.TX_PL. The following figure illustrates and defines the valid patterns.

Other preamble patterns are not supported.

Figure 46-57. Preamble Patterns



46.6.10.5.3 Preamble Reception

LON received frames begin with a preamble of variable length. The receiving algorithm does not check the preamble length, although a minimum of length of 4 bits is required for the receiving algorithm to consider the received preamble as valid.

As is the case with LON preamble transmission, two preamble patterns (ALL_ONE and ALL_ZERO) are allowed and can be configured through US_MAN.RX_PL. Figure 46-57 illustrates and defines the valid patterns.

Other preamble patterns are not supported.

46.6.10.5.4 Header Transmission

Each LON frame, after sending the preamble, starts with the frame header also called L2HDR according to the CEA-709 specification. This header consist of the priority bit, the alternative path bit and the backlog increment. It is the first data to be sent.

In LON mode the transmitting algorithm starts when the US_LONL2HDR register is written (it is the first data to send).

46.6.10.5.5 Header Reception

Each LON frame, after receiving the preamble, receives the frame header also called L2HDR according to the CEA-709 specification. This header consists of the priority bit, the alternative path bit, and the backlog increment.

The frame header is the first received data and the RXRDY bit rises as soon as the frame header as been received and stored in the Receive Holding register (US_RHR).

46.6.10.5.6 Data

Data are sent/received serially after the preamble transmission/reception. Data can be either sent/ received MSB first or LSB first depending on US_MR.MSBF.

46.6.10.5.7 CRC

The two last bytes of LON frames are dedicated to CRC.

When transmitting, the CRC of the frame is automatically generated and sent when expected.

When receiving frames the CRC is automatically checked and a LCRCE flag is set in US_CSR if the calculated CRC do not match the received one. Note that the two received CRC bytes are seen as two additional data from the user point of view.

46.7.12 USART Interrupt Disable Register (LON_MODE)

Name: US_IDR (LON_MODE)
Offset: 0x000C
Property: Write-only

This configuration is relevant only if USART_MODE = 0x9 in the USART Mode Register.

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

0: No effect

1: Disables the corresponding interrupt.

Bit	31	30	29	28	27	26	25	24
				LBLOVFE	LRXD	LFET	LCOL	LTXD
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
						UNRE	TXEMPTY	
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
	LCRCE	LSFE	OVRE				TXRDY	RXRDY
Access								
Reset								

Bit 28 – LBLOVFE LON Backlog Overflow Error Interrupt Disable

Bit 27 – LRXD LON Reception Done Interrupt Disable

Bit 26 – LFET LON Frame Early Termination Interrupt Disable

Bit 25 – LCOL LON Collision Interrupt Disable

Bit 24 – LTXD LON Transmission Done Interrupt Disable

Bit 10 – UNRE Underrun Error Interrupt Disable

Bit 9 – TXEMPTY TXEMPTY Interrupt Disable

Bit 7 – LCRCE LON CRC Error Interrupt Disable

Bit 6 – LSFE LON Short Frame Error Interrupt Disable

A dedicated Tx Buffer allocates Element Size 32-bit words in the Message RAM (see the table below). Therefore the start address of a dedicated Tx Buffer in the Message RAM is calculated by adding transmit buffer index (0…31) × Element Size to the Tx Buffer Start Address TXBC.TBSA.

Table 49-6. Tx Buffer / FIFO / Queue Element Size

TXESC.TBDS[2:0]	Data Field [bytes]	Element Size [RAM words]
0	8	4
1	12	5
2	16	6
3	20	7
4	24	8
5	32	10
6	48	14
7	64	18

49.5.5.3 Tx FIFO

Tx FIFO operation is configured by programming MCAN_TXBC.TFQM to ‘0’. Messages stored in the Tx FIFO are transmitted starting with the message referenced by the Get Index MCAN_TXFQS.TFGI. After each transmission the Get Index is incremented cyclically until the Tx FIFO is empty. The Tx FIFO enables transmission of messages with the same Message ID from different Tx Buffers in the order these messages have been written to the Tx FIFO. The MCAN calculates the Tx FIFO Free Level MCAN_TXFQS.TFFL as difference between Get and Put Index. It indicates the number of available (free) Tx FIFO elements.

New transmit messages have to be written to the Tx FIFO starting with the Tx Buffer referenced by the Put Index MCAN_TXFQS.TFQPI. An Add Request increments the Put Index to the next free Tx FIFO element. When the Put Index reaches the Get Index, Tx FIFO Full (MCAN_TXFQS.TFQF = ‘1’) is signalled. In this case no further messages should be written to the Tx FIFO until the next message has been transmitted and the Get Index has been incremented.

When a single message is added to the Tx FIFO, the transmission is requested by writing a ‘1’ to the TXBAR bit related to the Tx Buffer referenced by the Tx FIFO’s Put Index.

When multiple (n) messages are added to the Tx FIFO, they are written to n consecutive Tx Buffers starting with the Put Index. The transmissions are then requested via MCAN_TXBAR. The Put Index is then cyclically incremented by n. The number of requested Tx buffers should not exceed the number of free Tx Buffers as indicated by the Tx FIFO Free Level.

When a transmission request for the Tx Buffer referenced by the Get Index is cancelled, the Get Index is incremented to the next Tx Buffer with pending transmission request and the Tx FIFO Free Level is recalculated. When transmission cancellation is applied to any other Tx Buffer, the Get Index and the FIFO Free Level remain unchanged.

A Tx FIFO element allocates Element Size 32-bit words in the Message RAM (see the table Table 49-6). Therefore the start address of the next available (free) Tx FIFO Buffer is calculated by adding Tx FIFO/ Queue Put Index MCAN_TXFQS.TFQPI (0…31) × Element Size to the Tx Buffer Start Address MCAN_TXBC.TBSA.

Offset	Name	Bit Pos.								
		31:24								
0x01A0	PWM_CMPV7	7:0	CV[7:0]							
		15:8	CV[15:8]							
		23:16	CV[23:16]							
		31:24								CVM
0x01A4	PWM_CMPVUPD7	7:0	CVUPD[7:0]							
		15:8	CVUPD[15:8]							
		23:16	CVUPD[23:16]							
		31:24								CVMUPD
0x01A8	PWM_CMPM7	7:0	CTR[3:0]							CEN
		15:8	CPRCNT[3:0]				CPR[3:0]			
		23:16	CUPRCNT[3:0]				CUPR[3:0]			
		31:24								
0x01AC	PWM_CMPMUPD7	7:0	CTRUPD[3:0]							CENUPD
		15:8					CPRUPD[3:0]			
		23:16					CUPRUPD[3:0]			
		31:24								
0x01B0 ... 0x01FF	Reserved									
0x0200	PWM_CMR0	7:0					CPRE[3:0]			
		15:8			TCTS	DPOLI	UPDS	CES	CPOL	CALG
		23:16					PPM	DTLI	DTHI	DTE
		31:24								
0x0204	PWM_CDTY0	7:0	CDTY[7:0]							
		15:8	CDTY[15:8]							
		23:16	CDTY[23:16]							
		31:24								
0x0208	PWM_CDTYUPD0	7:0	CDTYUPD[7:0]							
		15:8	CDTYUPD[15:8]							
		23:16	CDTYUPD[23:16]							
		31:24								
0x020C	PWM_CPRD0	7:0	CPRD[7:0]							
		15:8	CPRD[15:8]							
		23:16	CPRD[23:16]							
		31:24								
0x0210	PWM_CPRDUPD0	7:0	CPRDUPD[7:0]							
		15:8	CPRDUPD[15:8]							
		23:16	CPRDUPD[23:16]							
		31:24								
0x0214	PWM_CCNT0	7:0	CNT[7:0]							
		15:8	CNT[15:8]							
		23:16	CNT[23:16]							
		31:24								
0x0218	PWM_DT0	7:0	DTH[7:0]							
		15:8	DTH[15:8]							

56.6.4 TRNG Interrupt Mask Register

Name: TRNG_IMR
Offset: 0x18
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
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
								DATRDY
Access								R
Reset								0

Bit 0 – DATRDY Data Ready Interrupt Mask

Value	Description
0	The corresponding interrupt is not enabled.
1	The corresponding interrupt is enabled.

62. Revision History

Table 62-1. Rev. A - 04/2018

Section Name or Type	Update Description
General Updates	• Updated from Atmel to Microchip style and template • Literature number: was changed from the Atmel 44003E to a Microchip DS number • Data sheet revision letter restarted to "A" • ISBN number added

Table 62-2. SAM E70/S70/V70/V71 Datasheet Rev. 44003E – Revision History

Date	Changes
12-Oct-16	Removed Preliminary status from the data sheet.
	Renamed instances of Timer Counter (TC) in: - Figure 10-1 “Product Mapping” - Table 12-1 “Real-time Event Mapping List” - Table 14-1 “Peripheral Identifiers”
	Restructured Section 1. ”Description”.
	Table 2-1 “Configuration Summary” : Added Note (3) on USART/UART functionality. Reorganized table notes.
	Table 6-3 “64-lead LQFP Package Pinout” : deleted signal names for pins 50, 51, 53 and 54 for PIO Peripheral D. (now unassigned)
	Table 14-1 “Peripheral Identifiers” : TWIHS0/1 instances read now as I2C-compatible.
	Section 15. ”ARM Cortex-M7” Number of IRQs changed to 74 in Table 15-3 “ARM Cortex-M7 Configuration” and Section 15.4.6.3 ”Interrupt Program Status Register”.
	Section 23. ”Supply Controller (SUPC)” Section 23.4.10 ”Register Write Protection”: in list of protectable registers, removed “System Controller Write Protection Mode Register”.
	Section 24. ”Watchdog Timer (WDT)” Removed references to LOCKMR in Section 24.4 ”Functional Description”, Section 24.5.1 ”Watchdog Timer Control Register” and Section 24.5.2 ”Watchdog Timer Mode Register”. Section 24.5.2 ”Watchdog Timer Mode Register”: corrected access to ‘Read/Write Once’.
	Section 27. ”Real-time Clock (RTC)” Reworked Positive Correction section in Figure 27-5 “Calibration Circuitry Waveforms”.
	Section 30. ”Clock Generator”