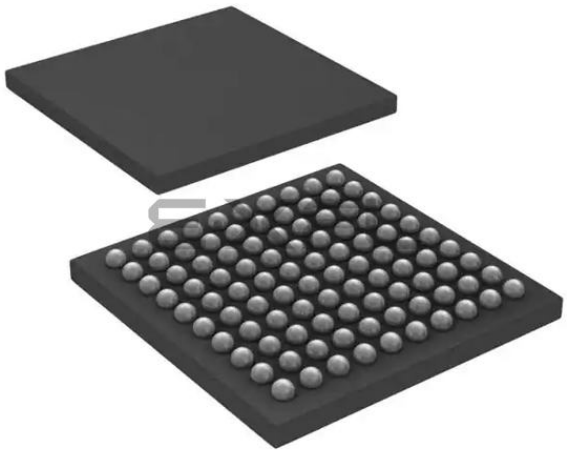


Welcome to E-XFL.COM

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"

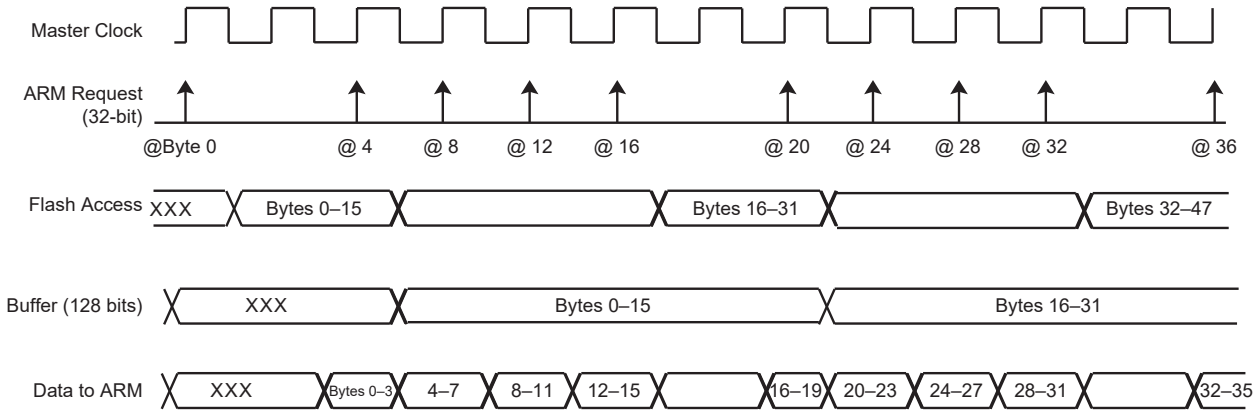
Details	
Product Status	Active
Core Processor	ARM® Cortex®-M7
Core Size	32-Bit Single-Core
Speed	300MHz
Connectivity	I²C, IrDA, LINbus, MMC/SD/SDIO, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	75
Program Memory Size	2MB (2M x 8)
Program Memory 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-VFBGA
Supplier Device Package	100-VFBGA (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70n21b-cfnt

22.4.2.3 Data Read Optimization

The organization of the Flash in 128 bits is associated with two 128-bit prefetch buffers and one 128-bit data read buffer, thus providing maximum system performance. This buffer is added in order to store the requested data plus all the data contained in the 128-bit aligned data. This speeds up sequential data reads if, for example, FWS is equal to 1 (see Figure 22-6). The data read optimization is enabled by default. If the bit EEFC_FMR.SCOD is set, this buffer is disabled and the data read is no longer optimized.

Note: No consecutive data read accesses are mandatory to benefit from this optimization.

Figure 22-6. Data Read Optimization for FWS = 1



22.4.3 Flash Commands

The EEFC offers a set of commands to manage programming the Flash memory, locking and unlocking lock regions, consecutive programming, locking and full Flash erasing, etc.

The commands are listed in the following table.

Table 22-1. Set of Commands

Command	Value	Mnemonic
Get Flash Descriptor	0x00	GETD
Write Page	0x01	WP
Write Page and Lock	0x02	WPL
Erase Page and Write Page	0x03	EWP
Erase Page and Write Page and then Lock	0x04	EWPL
Erase All	0x05	EA
Erase Pages	0x07	EPA
Set Lock Bit	0x08	SLB
Clear Lock Bit	0x09	CLB
Get Lock Bit	0x0A	GLB
Set GPNVM Bit	0x0B	SGPB
Clear GPNVM Bit	0x0C	CGPB

Once all of the previous steps have been completed, the peripheral clocks can be enabled and/or disabled via registers PMC_PCERx and PMC_PCDRx.

31.18 Clock Switching Details

31.18.1 Master Clock Switching Timings

The following two tables, Clock Switching Timings (Worst Case) and Clock Switching Timings Between Two PLLs (Worst Case) give the worst case timings required for MCK to switch from one selected clock to another one. This is in the event that the prescaler is deactivated. When the prescaler is activated, an additional time of 64 clock cycles of the newly selected clock has to be added.

Table 31-2. Clock Switching Timings (Worst Case)

From	MAINCK	SLCK	PLL Clock
To			
MAINCK	–	4 x SLCK + 2.5 x MAINCK	3 x PLL Clock + 4 x SLCK + 1 x MAINCK
SLCK	0.5 x MAINCK + 4.5 x SLCK	–	3 x PLL Clock + 5 x SLCK
PLL Clock	0.5 x MAINCK + 4 x SLCK + PLLCOUNT x SLCK + 2.5 x PLL Clock	2.5 x PLL Clock + 5 x SLCK + PLLCOUNT x SLCK	See the following table.

Note:

1. PLL designates any available PLL of the Clock Generator.
2. PLLCOUNT designates either PLLACOUNT or UPLLCOUNT.

Table 31-3. Clock Switching Timings Between Two PLLs (Worst Case)

	From	PLLACK	UPLL Clock
To			
PLLACK		–	3 x PLLACK + 4 x SLCK + 1.5 x PLLACK
UPLLCKDIV		3 x UPLLCKDIV + 4 x SLCK + 1.5 x UPLLCKDIV	–

Offset	Name	Bit Pos.								
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xC4	PIO_LSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xC8	PIO_ELSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xCC ... 0xCF	Reserved									
0xD0	PIO_FELLSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xD4	PIO_REHLSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xD8	PIO_FRLHSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xDC ... 0xDF	Reserved									
0xE0	PIO_LOCKSR	7:0	P7	P6	P5	P4	P3	P2	P1	P0
		15:8	P15	P14	P13	P12	P11	P10	P9	P8
		23:16	P23	P22	P21	P20	P19	P18	P17	P16
		31:24	P31	P30	P29	P28	P27	P26	P25	P24
0xE4	PIO_WPMR	7:0								WPEN
		15:8	WPKEY[7:0]							
		23:16	WPKEY[15:8]							
		31:24	WPKEY[23:16]							
0xE8	PIO_WPSR	7:0								WPVS
		15:8	WPVSRC[7:0]							
		23:16	WPVSRC[15:8]							
		31:24								
0xEC ... 0xFF	Reserved									
0x0100	PIO_SCHMITT	7:0	SCHMITT7	SCHMITT6	SCHMITT5	SCHMITT4	SCHMITT3	SCHMITT2	SCHMITT1	SCHMITT0
		15:8	SCHMITT15	SCHMITT14	SCHMITT13	SCHMITT12	SCHMITT11	SCHMITT10	SCHMITT9	SCHMITT8
		23:16	SCHMITT23	SCHMITT22	SCHMITT21	SCHMITT20	SCHMITT19	SCHMITT18	SCHMITT17	SCHMITT16

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.

Value	Description
	– If TDF optimization is enabled (TDF_MODE =1), TDF wait states will be inserted after the setup of NCS.
1	The write operation is controlled by the NWE signal. – If TDF optimization is enabled (TDF_MODE =1), TDF wait states will be inserted after the setup of NWE.

Bit 0 – READ_MODE Read Mode

Value	Description
0	The read operation is controlled by the NCS signal. – If TDF cycles are programmed, the external bus is marked busy after the rising edge of NCS. – If TDF optimization is enabled (TDF_MODE =1), TDF wait states are inserted after the setup of NCS.
1	The read operation is controlled by the NRD signal. – If TDF cycles are programmed, the external bus is marked busy after the rising edge of NRD. – If TDF optimization is enabled (TDF_MODE =1), TDF wait states are inserted after the setup of NRD.

Offset	Name	Bit Pos.								
		15:8	NDA[13:6]							
		23:16	NDA[21:14]							
		31:24	NDA[29:22]							
0x012C	XDMAC_CNDC3	7:0				NDVIEW[1:0]		NDDUP	NDSUP	NDE
		15:8								
		23:16								
		31:24								
0x0130	XDMAC_CUBC3	7:0	UBLEN[7:0]							
		15:8	UBLEN[15:8]							
		23:16	UBLEN[23:16]							
		31:24								
0x0134	XDMAC_CBC3	7:0	BLEN[7:0]							
		15:8					BLEN[11:8]			
		23:16								
		31:24								
0x0138	XDMAC_CC3	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]						
0x013C	XDMAC_CDS_MSP3	7:0	SDS_MSP[7:0]							
		15:8	SDS_MSP[15:8]							
		23:16	DDS_MSP[7:0]							
		31:24	DDS_MSP[15:8]							
0x0140	XDMAC_CSUS3	7:0	SUBS[7:0]							
		15:8	SUBS[15:8]							
		23:16	SUBS[23:16]							
		31:24								
0x0144	XDMAC_CDUS3	7:0	DUBS[7:0]							
		15:8	DUBS[15:8]							
		23:16	DUBS[23:16]							
		31:24								
0x0148 ... 0x014F	Reserved									
0x0150	XDMAC_CIE4	7:0		ROIE	WBIE	RBIE	FIE	DIE	LIE	BIE
		15:8								
		23:16								
		31:24								
0x0154	XDMAC_CID4	7:0		ROID	WBEID	RBEID	FID	DID	LID	BID
		15:8								
		23:16								
		31:24								
0x0158	XDMAC_CIM4	7:0		ROIM	WBEIM	RBEIM	FIM	DIM	LIM	BIM
		15:8								
		23:16								
		31:24								

Offset	Name	Bit Pos.								
0x06E0	GMAC_ST2ER0	7:0	COMPVAL[7:0]							
		15:8	COMPVAL[15:8]							
		23:16								
		31:24								
0x06E4	GMAC_ST2ER1	7:0	COMPVAL[7:0]							
		15:8	COMPVAL[15:8]							
		23:16								
		31:24								
0x06E8	GMAC_ST2ER2	7:0	COMPVAL[7:0]							
		15:8	COMPVAL[15:8]							
		23:16								
		31:24								
0x06EC	GMAC_ST2ER3	7:0	COMPVAL[7:0]							
		15:8	COMPVAL[15:8]							
		23:16								
		31:24								
0x06F0 ... 0x06FF	Reserved									
0x0700	GMAC_ST2CW00	7:0	MASKVAL[7:0]							
		15:8	MASKVAL[15:8]							
		23:16	COMPVAL[7:0]							
		31:24	COMPVAL[15:8]							
0x0704	GMAC_ST2CW10	7:0	OFFSSTRT[0:0]	OFFSVAL[6:0]						
		15:8								OFFSSTRT[1:1]
		23:16								
		31:24								
0x0708	GMAC_ST2CW01	7:0	MASKVAL[7:0]							
		15:8	MASKVAL[15:8]							
		23:16	COMPVAL[7:0]							
		31:24	COMPVAL[15:8]							
0x070C	GMAC_ST2CW11	7:0	OFFSSTRT[0:0]	OFFSVAL[6:0]						
		15:8								OFFSSTRT[1:1]
		23:16								
		31:24								
0x0710	GMAC_ST2CW02	7:0	MASKVAL[7:0]							
		15:8	MASKVAL[15:8]							
		23:16	COMPVAL[7:0]							
		31:24	COMPVAL[15:8]							
0x0714	GMAC_ST2CW12	7:0	OFFSSTRT[0:0]	OFFSVAL[6:0]						

Offset	Name	Bit Pos.								
0x05E0	USBHS_HSTPIPIMR8 (INTPIPES)	7:0	SHORTPACKETIE	RXSTALLDE	OVERFIE	NAKEDE	PERRE	UNDERFIE	TXOUTE	RXINE
		15:8		FIFOCON		NBUSYBKE				
		23:16						RSTDT	PFREEZE	PDISHDMA
		31:24								
0x05E0	USBHS_HSTPIPIMR8 (ISOPIPES)	7:0	SHORTPACKETIE	CRCERRE	OVERFIE	NAKEDE	PERRE	UNDERFIE	TXOUTE	RXINE
		15:8		FIFOCON		NBUSYBKE				
		23:16						RSTDT	PFREEZE	PDISHDMA
		31:24								
0x05E4	USBHS_HSTPIPIMR9	7:0	SHORTPACKETIE	RXSTALLDE	OVERFIE	NAKEDE	PERRE	TXSTPE	TXOUTE	RXINE
		15:8		FIFOCON		NBUSYBKE				
		23:16						RSTDT	PFREEZE	PDISHDMA
		31:24								
0x05E4	USBHS_HSTPIPIMR9 (INTPIPES)	7:0	SHORTPACKETIE	RXSTALLDE	OVERFIE	NAKEDE	PERRE	UNDERFIE	TXOUTE	RXINE
		15:8		FIFOCON		NBUSYBKE				
		23:16						RSTDT	PFREEZE	PDISHDMA
		31:24								
0x05E4	USBHS_HSTPIPIMR9 (ISOPIPES)	7:0	SHORTPACKETIE	CRCERRE	OVERFIE	NAKEDE	PERRE	UNDERFIE	TXOUTE	RXINE
		15:8		FIFOCON		NBUSYBKE				
		23:16						RSTDT	PFREEZE	PDISHDMA
		31:24								
0x05E8 ... 0x05EF	Reserved									
0x05F0	USBHS_HSTPIPIER0	7:0	SHORTPACKETIES	RXSTALLDES	OVERFIES	NAKEDES	PERRES	TXSTPES	TXOUTES	RXINES
		15:8				NBUSYBKES				
		23:16						RSTDTS	PFREEZES	PDISHDMAS
		31:24								
0x05F0	USBHS_HSTPIPIER0 (INTPIPES)	7:0	SHORTPACKETIES	RXSTALLDES	OVERFIES	NAKEDES	PERRES	UNDERFIES	TXOUTES	RXINES
		15:8				NBUSYBKES				
		23:16						RSTDTS	PFREEZES	PDISHDMAS
		31:24								
0x05F0	USBHS_HSTPIPIER0 (ISOPIPES)	7:0	SHORTPACKETIES	CRCERRES	OVERFIES	NAKEDES	PERRES	UNDERFIES	TXOUTES	RXINES
		15:8				NBUSYBKES				
		23:16						RSTDTS	PFREEZES	PDISHDMAS
		31:24								
0x05F4	USBHS_HSTPIPIER1	7:0	SHORTPACKETIES	RXSTALLDES	OVERFIES	NAKEDES	PERRES	TXSTPES	TXOUTES	RXINES
		15:8				NBUSYBKES				
		23:16						RSTDTS	PFREEZES	PDISHDMAS

Offset	Name	Bit Pos.								
		23:16								
		31:24								

Bit 1 – RXOUTIC Received OUT Data Interrupt Clear

Bit 0 – TXINIC Transmitted IN Data Interrupt Clear

Bit 5 – END_BF_ST End of Channel Buffer Status
Valid until the CHANN_ENB flag is cleared at the end of the next buffer transfer.

Value	Description
0	Cleared automatically when read by software.
1	Set by hardware when the BUFF_COUNT count-down reaches zero.

Bit 4 – END_TR_ST End of Channel Transfer Status
Valid until the CHANN_ENB flag is cleared at the end of the next buffer transfer.

Value	Description
0	Cleared automatically when read by software.
1	Set by hardware when the last packet transfer is complete, if the USBHS device has ended the transfer.

Bit 1 – CHANN_ACT Channel Active Status
When a packet transfer is ended, this bit is automatically reset.

When a packet transfer cannot be completed due to an END_BF_ST, this flag stays set during the next channel descriptor load (if any) and potentially until completion of a USBHS packet transfer, if allowed by the new descriptor.

Value	Description
0	The DMA channel is no longer trying to source the packet data.
1	The DMA channel is currently trying to source packet data, i.e., selected as the highest-priority requesting channel.

Bit 0 – CHANN_ENB Channel Enable Status
When any transfer is ended either due to an elapsed byte count or to completion of a USBHS device-initiated transfer, this bit is automatically reset.

This bit is normally set or cleared by writing into the USBHS_DEVDMACONTROLx.CHANN_ENB bit field either by software or descriptor loading.

If a channel request is currently serviced when the USBHS_DEVDMACONTROLx.CHANN_ENB bit is cleared, the DMA FIFO buffer is drained until it is empty, then this status bit is cleared.

Value	Description
0	If cleared, the DMA channel no longer transfers data, and may load the next descriptor if the USBHS_DEVDMACONTROLx.LDNXT_DSC bit is set.
1	If set, the DMA channel is currently enabled and transfers data upon request.

41.5 Signal Description

Table 41-1. Signal Description

Pin Name	Pin Description	Type	
		Master	Slave
MISO	Master In Slave Out	Input	Output
MOSI	Master Out Slave In	Output	Input
SPCK	Serial Clock	Output	Input
NPCS1–NPCS3	Peripheral Chip Selects	Output	Unused
NPCS0/NSS	Peripheral Chip Select/Slave Select	Output	Input

41.6 Product Dependencies

41.6.1 I/O Lines

The pins used for interfacing the compliant external devices can be multiplexed with PIO lines. The programmer must first program the PIO controllers to assign the SPI pins to their peripheral functions.

41.6.2 Power Management

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

41.6.3 Interrupt

The SPI interface has an interrupt line connected to the interrupt controller. Handling the SPI interrupt requires programming the interrupt controller before configuring the SPI.

41.6.4 Direct Memory Access Controller (DMAC)

The SPI interface can be used in conjunction with the DMAC in order to reduce processor overhead. For a full description of the DMAC, refer to the relevant section.

41.7 Functional Description

41.7.1 Modes of Operation

The SPI operates in Master mode or in Slave mode.

- The SPI operates in Master mode by setting the MSTR bit in the SPI Mode Register (SPI_MR):
  - Pins NPCS0 to NPCS3 are all configured as outputs
  - The SPCK pin is driven
  - The MISO line is wired on the receiver input
  - The MOSI line is driven as an output by the transmitter.
- The SPI operates in Slave mode if the MSTR bit in SPI_MR is written to ‘0’:
  - The MISO line is driven by the transmitter output
  - The MOSI line is wired on the receiver input

41.8.6 SPI Interrupt Enable Register

Name: SPI_IER
Offset: 0x14
Reset: –
Property: Write-only

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

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

0: No effect.

1: Enables the corresponding interrupt.

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
						UNDES	TXEMPTY	NSSR
Access						W	W	W
Reset						–	–	–

Bit	7	6	5	4	3	2	1	0
					OVRES	MODF	TDRE	RDRF
Access					W	W	W	W
Reset					–	–	–	–

Bit 10 – UNDES Underrun Error Interrupt Enable

Bit 9 – TXEMPTY Transmission Registers Empty Enable

Bit 8 – NSSR NSS Rising Interrupt Enable

Bit 3 – OVRES Overrun Error Interrupt Enable

Bit 2 – MODF Mode Fault Error Interrupt Enable

Bit 1 – TDRE SPI Transmit Data Register Empty Interrupt Enable

Bit 0 – RDRF Receive Data Register Full Interrupt Enable

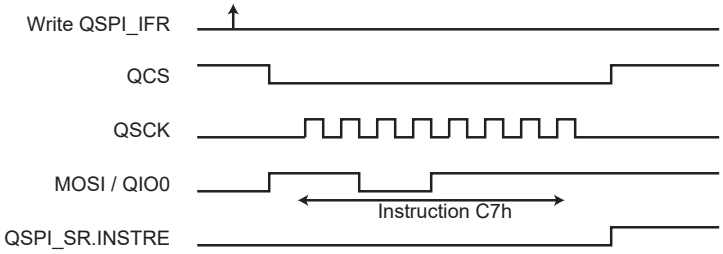
Example 1:

Instruction in Single-bit SPI, without address, without option, without data.

Command: CHIP ERASE (C7h).

- Write 0x0000_00C7 in QSPI_ICR.
- Write 0x0000_0010 in QSPI_IFR.
- Wait for QSPI_SR.INSTRE to rise.

Figure 42-11. Instruction Transmission Waveform 1



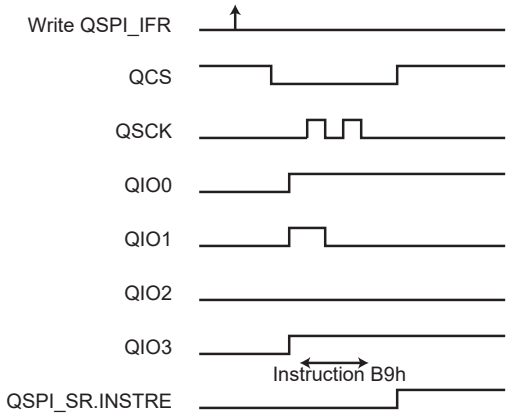
Example 2:

Instruction in Quad SPI, without address, without option, without data.

Command: POWER DOWN (B9h)

- Write 0x0000_00B9 in QSPI_ICR.
- Write 0x0000_0016 in QSPI_IFR.
- Wait for QSPI_SR.INSTRE to rise.

Figure 42-12. Instruction Transmission Waveform 2



Example 3:

Instruction in Single-bit SPI, with address in Single-bit SPI, without option, without data.

Command: BLOCK ERASE (20h)

- Write the address (of the block to erase) in QSPI_AR.
- Write 0x0000_0020 in QSPI_ICR.
- Write 0x0000_0030 in QSPI_IFR.
- Wait for QSPI_SR.INSTRE to rise.

50.7.21 TC Fault Mode Register

Name: TC_FMR
Offset: 0xD8
Reset: 0x00000000
Property: Read/Write

This register can only be written if the WPEN bit is cleared in the TC 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								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
							ENCF1	ENCF0
Access							R/W	R/W
Reset							0	0

Bit 1 – ENCF1 Enable Compare Fault Channel 1

Value	Description
0	Disables the FAULT output source (CPCS flag) from channel 1.
1	Enables the FAULT output source (CPCS flag) from channel 1.

Bit 0 – ENCF0 Enable Compare Fault Channel 0

Value	Description
0	Disables the FAULT output source (CPCS flag) from channel 0.
1	Enables the FAULT output source (CPCS flag) from channel 0.

51. Pulse Width Modulation Controller (PWM)

51.1 Description

The Pulse Width Modulation Controller (PWM) generates output pulses on 4 channels independently according to parameters defined per channel. Each channel controls two complementary square output waveforms. Characteristics of the output waveforms such as period, duty-cycle, polarity and dead-times (also called dead-bands or non-overlapping times) are configured through the user interface. Each channel selects and uses one of the clocks provided by the clock generator. The clock generator provides several clocks resulting from the division of the PWM peripheral clock. External triggers can be managed to allow output pulses to be modified in real time.

All accesses to the PWM are made through registers mapped on the peripheral bus. All channels integrate a double buffering system in order to prevent an unexpected output waveform while modifying the period, the spread spectrum, the duty-cycle or the dead-times.

Channels can be linked together as synchronous channels to be able to update their duty-cycle or dead-times at the same time.

The update of duty-cycles of synchronous channels can be performed by the DMA Controller channel which offers buffer transfer without processor Intervention.

The PWM includes a spread-spectrum counter to allow a constantly varying period (only for Channel 0). This counter may be useful to minimize electromagnetic interference or to reduce the acoustic noise of a PWM driven motor.

The PWM provides 8 independent comparison units capable of comparing a programmed value to the counter of the synchronous channels (counter of channel 0). These comparisons are intended to generate software interrupts, to trigger pulses on the 2 independent event lines (in order to synchronize ADC conversions with a lot of flexibility independently of the PWM outputs) and to trigger DMA Controller transfer requests.

PWM outputs can be overridden synchronously or asynchronously to their channel counter.

The PWM provides a fault protection mechanism with 8 fault inputs, capable to detect a fault condition and to override the PWM outputs asynchronously (outputs forced to ‘0’, ‘1’ or Hi-Z).

For safety usage, some configuration registers are write-protected.

51.2 Embedded Characteristics

- 4 Channels
- Common Clock Generator Providing Thirteen Different Clocks
  - A Modulo n Counter Providing Eleven Clocks
  - Two Independent Linear Dividers Working on Modulo n Counter Outputs
- Independent Channels
  - Independent 16-bit Counter for Each Channel
  - Independent Complementary Outputs with 12-bit Dead-Time Generator (Also Called Dead-Band or Non-Overlapping Time) for Each Channel
  - Independent Push-Pull Mode for Each Channel
  - Independent Enable Disable Command for Each Channel

Value	Name	Description
6	TRGSEL6	PWM1 Event 0
7	TRGSEL7	PWM 1 Event 1

Bits 0, 1 – TRGENx Trigger Enable of Channel x

Value	Name	Description
0	DIS	Trigger mode disabled. DACC is in Free-running mode or Max speed mode.
1	EN	Trigger mode enabled.

53.7.11 DACC Interrupt Status Register

Name: DACC_ISR
Offset: 0x30
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
			EOC1	EOC0			TXRDY1	TXRDY0
Access			R	R			R	R
Reset			0	0			0	0

Bits 4, 5 – EOCx End of Conversion Interrupt Flag of channel x

Value	Description
0	No conversion has been performed since the last read of DACC_ISR.
1	At least one conversion has been performed since the last read of DACC_ISR.

Bits 0, 1 – TXRDYx Transmit Ready Interrupt Flag of channel x

Value	Description
0	DACC is not ready to accept new conversion requests.
1	DACC is ready to accept new conversion requests.

Symbol		VDDIO Supply		Unit
	Parameter	Min	Max	
SMC_5	A0–A22 Valid before NRD High	(NRD_SETUP + NRD_PULSE) × t_{CPMCK} - 4.3	–	ns
SMC_6	NCS low before NRD High	(NRD_SETUP + NRD_PULSE - NCS_RD_SETUP) × t_{CPMCK} - 2.4	–	ns
SMC_7	NRD Pulse Width	NRD_PULSE × t_{CPMCK} - 0.3	–	ns

Table 58-61. SMC Read Signals - NCS Controlled (READ_MODE = 0)

Symbol		VDDIO Supply		Unit
	Parameter	Min	Max	
NO HOLD Settings (NCS_RD_HOLD = 0)				
SMC_8	Data Setup before NCS High	21.4	–	ns
SMC_9	Data Hold after NCS High	0	–	ns
HOLD Settings (NCS_RD_HOLD ≠ 0)				
SMC_{10}	Data Setup before NCS High	11.7	–	ns
SMC_{11}	Data Hold after NCS High	0	–	ns
HOLD or NO HOLD Settings (NCS_RD_HOLD ≠ 0, NCS_RD_HOLD = 0)				
SMC_{12}	A0–A22 valid before NCS High	(NCS_RD_SETUP + NCS_RD_PULSE) × t_{CPMCK} - 3.9	–	ns
SMC_{13}	NRD low before NCS High	(NCS_RD_SETUP + NCS_RD_PULSE - NRD_SETUP) × t_{CPMCK} - 4.2	–	ns
SMC_{14}	NCS Pulse Width	NCS_RD_PULSE length × t_{CPMCK} - 0.2	–	ns

58.13.1.9.3 Write Timings

Table 58-62. SMC Write Signals - NWE Controlled (WRITE_MODE = 1)

Symbol	VDDIO Supply	1.8V Domain	3.3V Domain	1.8V Domain	3.3V Domain	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} - 5.4	NWE_PULSE × t_{CPMCK} - 4.6	–	–	ns
SMC_{16}	NWE Pulse Width	NWE_PULSE × t_{CPMCK} - 0.7	NWE_PULSE × t_{CPMCK} - 0.3	–	–	ns
SMC_{17}	A0–A22 valid before NWE low	NWE_SETUP × t_{CPMCK} - 4.9	NWE_SETUP × t_{CPMCK} - 4.2	–	–	ns

Date	Changes
	Updated Figure 35-5, “NAND Flash Signal Multiplexing on SMC Pins” and added Note 1 below the figure. Section 35.10 “Scrambling/Unscrambling Function”: added details on access for SMC_KEY1 and SMC_KEY2 registers. In Table 35-10 “Register Mapping” and register table sections: SMC OCMS Mode Register now ”“SMC Off-Chip Memory Scrambling Register”. SMC OCMS Key1 Register now ”“SMC Off-Chip Memory Scrambling Key1 Register”. SMC OCMS Key2 Register now “SMC Off-Chip Memory Scrambling Key2 Register”. Section 35.16.5 “SMC Off-Chip Memory Scrambling Register”: corrected bits 8 to 11 to ‘CSxSE’ (were reserved). Section 35.16.6 “SMC Off-Chip Memory Scrambling Key1 Register” and Section 35.16.7 “SMC Off-Chip Memory Scrambling Key2 Register”: added Note (1) to clarify Write-once access.
	Section 36. “DMA Controller (XDMAC)” Updated TC peripheral names and added I2SC in Table 36-1 “Peripheral Hardware Requests”. Section 36.2 “Embedded Characteristics”: added FIFO size. Updated Figure 36-1, “DMA Controller (XDMAC) Block Diagram”. Section 36.5.4.1 “Single Block With Single Microblock Transfer”: in Step 6, deleted sub-step to activate a secure channel. Table 36-3 “Register Mapping“: corrected access of XDMAC_GTYPE, XDMAC_GWAC, XDMAC_CIM. Section 36.9.6 “XDMAC Global Interrupt Mask Register”: corrected access to Read-only. Section 36.9.28 “XDMAC Channel x [x = 0..23] Configuration Register”: bit 5 now reserved (was PROT). Deleted PROT bit description. Updated PERIF field description. Modified INITD bit description.
	Section 38. “USB High-Speed Interface (USBHS)” Table 38-1 “Description of USB Pipes/Endpoints”: corrected value in ‘High Bandwidth’ column for Pipe/Endpoint 1. Added Section 38.4.1 “I/O Lines”. Updated Figure 38-2, “General States”. Updated Section 38.5.3.3 “Device Detection” and added Note on VBUS supply. Section 38.6.1 “General Control Register”: added bit 8, VBUSHWC. Section 38.6.4 “General Status Set Register”: added bit 9, VBUSRQS. Section 38.6.12 “Device Endpoint Register”: bit 9 changed from ‘reserved’ to EPEN9. Bit 25 changed from ‘reserved’ to EPRST9. Bits 10 and 11 now reserved in registers: