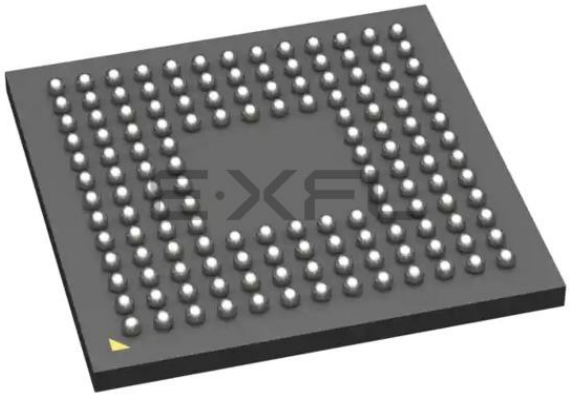


Welcome to E-XFL.COM

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	CANbus, EBI/EMI, Ethernet, 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)	1.62V ~ 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/atsame70q21b-cfn

19.4.4 Bus Matrix Priority Registers B For Slaves

Name: MATRIX_PRBSx
Offset: 0x84 + x*0x08 [x=0..8]
Reset: 0x00000222
Property: Read/Write

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
							M12PR[1:0]	
Access							R/W	R/W
Reset							0	0

Bit	15	14	13	12	11	10	9	8
			M11PR[1:0]				M10PR[1:0]	
Access			R/W	R/W			R/W	R/W
Reset			0	0			1	0

Bit	7	6	5	4	3	2	1	0
			M9PR[1:0]				M8PR[1:0]	
Access			R/W	R/W			R/W	R/W
Reset			1	0			1	0

Bits 0:1, 4:5, 8:9, 12:13, 16:17 – MxPR Master 8 Priority
Fixed priority of Master x for accessing the selected slave. The higher the number, the higher the priority.

All the masters programmed with the same MxPR value for the slave make up a priority pool.

Round-robin arbitration is used in the lowest (MxPR = 0) and highest (MxPR = 3) priority pools.

Fixed priority is used in intermediate priority pools (MxPR = 1) and (MxPR = 2).

See “Arbitration Priority Scheme” for details.

31.20.12 PMC USB Clock Register

Name: PMC_USB
Offset: 0x0038
Reset: 0x00000000
Property: Read/Write

This register can only be written if the WPEN bit is cleared in the PMC 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					USBDIV[3:0]			
Reset					0	0	0	0

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

Bits 11:8 – USBDIV[3:0] Divider for USB_48M
USB_48M is input clock divided by USBDIV+1.

Bit 0 – USBS USB Input Clock Selection

Value	Description
0	USB_48M input is PLLA.
1	USB_48M input is UPLL.

- The frequency of peripheral clock must be strictly superior to two times the frequency of the clock of the device which generates the parallel data.

32.5.14.4 Programming Sequence

32.5.14.4.1 Without DMA

1. Write PIO_PCIDR and PIO_PCIER in order to configure the Parallel Capture mode interrupt mask.
2. Write PIO_PCMR to set the fields DSIZE, ALWYS, HALFS and FRSTS in order to configure the Parallel Capture mode WITHOUT enabling the Parallel Capture mode.
3. Write PIO_PCMR to set the PCEN bit to one in order to enable the Parallel Capture mode WITHOUT changing the previous configuration.
4. Wait for a data ready by polling the DRDY flag in PIO_PCISR or by waiting for the corresponding interrupt.
5. Check OVRE flag in PIO_PCISR.
6. Read the data in PIO_PCRHR.
7. If new data are expected, go to step 4.
8. Write PIO_PCMR to set the PCEN bit to zero in order to disable the Parallel Capture mode WITHOUT changing the previous configuration.

32.5.14.4.2 With DMA

1. Write PIO_PCIDR and PIO_PCIER in order to configure the Parallel Capture mode interrupt mask.
2. Configure DMA transfer in DMA registers.
3. Write PIO_PCMR to set the fields DSIZE, ALWYS, HALFS and FRSTS in order to configure the Parallel Capture mode WITHOUT enabling the Parallel Capture mode.
4. Write PIO_PCMR to set PCEN bit to one in order to enable the Parallel Capture mode WITHOUT changing the previous configuration.
5. Wait for the DMA status flag to indicate that the buffer transfer is complete.
6. Check OVRE flag in PIO_PCISR.
7. If a new buffer transfer is expected, go to step 5.
8. Write PIO_PCMR to set the PCEN bit to zero in order to disable the Parallel Capture mode WITHOUT changing the previous configuration.

32.5.15 I/O Lines Programming Example

The programming example shown in the following table is used to obtain the following configuration:

- 4-bit output port on I/O lines 0 to 3 (should be written in a single write operation), open-drain, with pullup resistor
- Four output signals on I/O lines 4 to 7 (to drive LEDs for example), driven high and low, no pullup resistor, no pulldown resistor
- Four input signals on I/O lines 8 to 11 (to read push-button states for example), with pullup resistors, glitch filters and input change interrupts
- Four input signals on I/O line 12 to 15 to read an external device status (polled, thus no input change interrupt), no pullup resistor, no glitch filter
- I/O lines 16 to 19 assigned to peripheral A functions with pullup resistor
- I/O lines 20 to 23 assigned to peripheral B functions with pulldown resistor
- I/O lines 24 to 27 assigned to peripheral C with input change interrupt, no pullup resistor and no pulldown resistor
- I/O lines 28 to 31 assigned to peripheral D, no pullup resistor and no pulldown resistor

32.6.1.9 PIO Input Filter Status Register

Name: PIO_IFSR
Offset: 0x0028
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
	P31	P30	P29	P28	P27	P26	P25	P24
Access								
Reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	P23	P22	P21	P20	P19	P18	P17	P16
Access								
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	P15	P14	P13	P12	P11	P10	P9	P8
Access								
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	P7	P6	P5	P4	P3	P2	P1	P0
Access								
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, 24, 25, 26, 27, 28, 29, 30, 31 – P PIO Input Filter Status

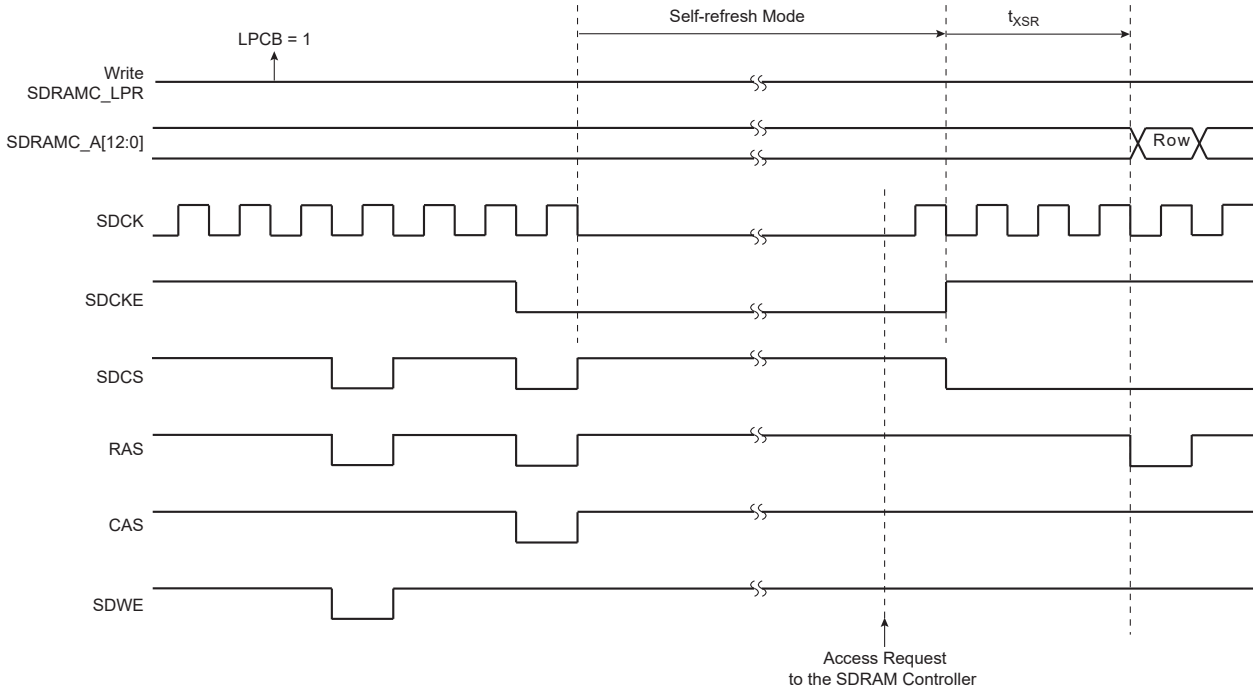
Value	Description
0	The input glitch filter is disabled on the I/O line.
1	The input glitch filter is enabled on the I/O line.

After initialization, as soon as the PASR/DS/TCSR fields are modified and Self-refresh mode is activated, the Extended Mode register is accessed automatically and the PASR/DS/TCSR bits are updated before entry into Self-refresh mode. This feature is not supported when SDRAMC shares an external bus with another controller.

The SDRAM device must remain in Self-refresh mode for a minimum period of t_{RAS} and may remain in Self-refresh mode for an indefinite period. Refer to the following figure.

Note: Some SDRAM providers impose some cycles of burst autorefresh immediately before self-refresh entry and immediately after self-refresh exit. For example, a SDRAM with 4096 rows will impose 4096 cycles of burst autorefresh. This constraint is not supported.

Figure 34-6. Self-refresh Mode Behavior



34.6.5.2 Low-power Mode

This mode is selected by configuring SDRAMC_LPR.LPCB to 2. Power consumption is greater than in Self-refresh mode. All the input and output buffers of the SDRAM device are deactivated except SDCKE, which remains low. In contrast to Self-refresh mode, the SDRAM device cannot remain in Low-power mode longer than the refresh period (64 ms for a whole device refresh operation). As no autorefresh operations are performed by the SDRAM itself, the SDRAMC carries out the refresh operation. The exit procedure is faster than in Self-refresh mode.

Refer to the following figure.

35. Static Memory Controller (SMC)

35.1 Description

The External Bus Interface (EBI) is designed to ensure the successful data transfer between several external devices and the ARM-based microcontroller. The Static Memory Controller (SMC) is part of the EBI.

The SMC handles several types of external memory and peripheral devices, such as SRAM, PSRAM, PROM, EPROM, EEPROM, LCD Module, NOR Flash and NAND Flash.

The SMC generates the signals that control the access to the external memory devices or peripheral devices. It has 4 chip selects, a 24-bit address bus, and a configurable 8 or 16-bit data bus. Separate read and write control signals allow for direct memory and peripheral interfacing. Read and write signal waveforms are fully adjustable.

The SMC can manage wait requests from external devices to extend the current access. The SMC is provided with an automatic Slow clock mode. In Slow clock mode, it switches from user-programmed waveforms to slow-rate specific waveforms on read and write signals. The SMC supports asynchronous burst read in Page mode access for page sizes up to 32 bytes.

The external data bus can be scrambled/unscrambled by means of user keys.

35.2 Embedded Characteristics

- Four Chip Selects Available
- 16-Mbyte Address Space per Chip Select
- 8-bit or 16-bit Data Bus
- Zero Wait State Scrambling/Unscrambling Function with User Key
- Word, Halfword, Byte Transfers
- Byte Write or Byte Select Lines
- Programmable Setup, Pulse And Hold Time for Read Signals per Chip Select
- Programmable Setup, Pulse And Hold Time for Write Signals per Chip Select
- Programmable Data Float Time per Chip Select
- External Wait Request
- Automatic Switch to Slow Clock Mode
- Asynchronous Read in Page Mode Supported: Page Size Ranges from 4 to 32 Bytes
- Register Write Protection

35.3 I/O Lines Description

Table 35-1. I/O Line Description

Name	Description	Type	Active Level
NCS[3:0]	Static Memory Controller Chip Select Lines	Output	Low
NRD	Read Signal	Output	Low

Offset	Name	Bit Pos.								
		31:24								
0x2C	XDMAC_GWS	7:0	WS7	WS6	WS5	WS4	WS3	WS2	WS1	WS0
		15:8	WS15	WS14	WS13	WS12	WS11	WS10	WS9	WS8
		23:16	WS23	WS22	WS21	WS20	WS19	WS18	WS17	WS16
		31:24								
0x30	XDMAC_GRWS	7:0	RWS7	RWS6	RWS5	RWS4	RWS3	RWS2	RWS1	RWS0
		15:8	RWS15	RWS14	RWS13	RWS12	RWS11	RWS10	RWS9	RWS8
		23:16	RWS23	RWS22	RWS21	RWS20	RWS19	RWS18	RWS17	RWS16
		31:24								
0x34	XDMAC_GRWR	7:0	RWR7	RWR6	RWR5	RWR4	RWR3	RWR2	RWR1	RWR0
		15:8	RWR15	RWR14	RWR13	RWR12	RWR11	RWR10	RWR9	RWR8
		23:16	RWR23	RWR22	RWR21	RWR20	RWR19	RWR18	RWR17	RWR16
		31:24								
0x38	XDMAC_GSWR	7:0	SWREQ7	SWREQ6	SWREQ5	SWREQ4	SWREQ3	SWREQ2	SWREQ1	SWREQ0
		15:8	SWREQ15	SWREQ14	SWREQ13	SWREQ12	SWREQ11	SWREQ10	SWREQ9	SWREQ8
		23:16	SWREQ23	SWREQ22	SWREQ21	SWREQ20	SWREQ19	SWREQ18	SWREQ17	SWREQ16
		31:24								
0x3C	XDMAC_GSWS	7:0	SWRS7	SWRS6	SWRS5	SWRS4	SWRS3	SWRS2	SWRS1	SWRS0
		15:8	SWRS15	SWRS14	SWRS13	SWRS12	SWRS11	SWRS10	SWRS9	SWRS8
		23:16	SWRS23	SWRS22	SWRS21	SWRS20	SWRS19	SWRS18	SWRS17	SWRS16
		31:24								
0x40	XDMAC_GSWF	7:0	SWF7	SWF6	SWF5	SWF4	SWF3	SWF2	SWF1	SWF0
		15:8	SWF15	SWF14	SWF13	SWF12	SWF11	SWF10	SWF9	SWF8
		23:16	SWF23	SWF22	SWF21	SWF20	SWF19	SWF18	SWF17	SWF16
		31:24								
0x44 ... 0x4F	Reserved									
0x50	XDMAC_CIE0	7:0		ROIE	WBIE	RBIE	FIE	DIE	LIE	BIE
		15:8								
		23:16								
		31:24								
0x54	XDMAC_CID0	7:0		ROID	WBEID	RBEID	FID	DID	LID	BID
		15:8								
		23:16								
		31:24								
0x58	XDMAC_CIM0	7:0		ROIM	WBEIM	RBEIM	FIM	DIM	LIM	BIM
		15:8								
		23:16								
		31:24								
0x5C	XDMAC_CIS0	7:0		ROIS	WBEIS	RBEIS	FIS	DIS	LIS	BIS
		15:8								
		23:16								
		31:24								
0x60	XDMAC_CSA0	7:0	SA[7:0]							
		15:8	SA[15:8]							

Signal Name	Function	MII	RMII
GMDC	Management Data Clock	MDC	MDC
GMDIO	Management Data Input/Output	MDIO	MDIO

Note:

1. Input only. GTXCK must be provided with a 25 MHz / 50 MHz external crystal oscillator for MII / RMII interfaces, respectively.

38.5 Product Dependencies

38.5.1 I/O Lines

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

38.5.2 Power Management

The GMAC is not continuously clocked. The user must first enable the GMAC clock in the Power Management Controller before using it.

38.5.3 Interrupt Sources

The GMAC interrupt line is connected to one of the internal sources of the interrupt controller. Using the GMAC interrupt requires prior programming of the interrupt controller.

The GMAC features 6 interrupt sources. Refer to the table "Peripheral Identifiers" in the section "Peripherals" for the interrupt numbers for GMAC priority queues.

Related Links

14.1 Peripheral Identifiers

38.6 Functional Description

38.6.1 Media Access Controller

The Transmit Block of the Media Access Controller (MAC) takes data from FIFO, adds preamble, checks and adds padding and frame check sequence (FCS). Both half duplex and full duplex Ethernet modes of operation are supported.

When operating in half duplex mode, the MAC Transmit Block generates data according to the Carrier Sense Multiple Access with Collision Detect (CSMA/CD) protocol. The start of transmission is deferred if Carrier Sense (CRS) is active. If Collision (COL) is detected during transmission, a jam sequence is asserted and the transmission is retried after a random back off. The CRS and COL signals have no effect in full duplex mode.

The Receive Block of the MAC checks for valid preamble, FCS, alignment and length, and presents received frames to the MAC address checking block and FIFO. Software can configure the GMAC to receive jumbo frames of up to 10240 Bytes. It can optionally strip CRC (Cyclic Redundancy Check) from the received frame before transferring it to FIFO.

The Address Checker recognizes four specific 48-bit addresses, can recognize four different types of ID values, and contains a 64-bit Hash register for matching multicast and unicast addresses as required. It

Cleared on read.

Bit 24 – PDRQFT PDelay Request Frame Transmitted
Indicates a PTP pdelay_req frame has been transmitted.

Cleared on read.

Bit 23 – PDRSFR PDelay Response Frame Received
Indicates a PTP pdelay_resp frame has been received.

Cleared on read.

Bit 22 – PDRQFR PDelay Request Frame Received
Indicates a PTP pdelay_req frame has been received.

Cleared on read.

Bit 21 – SFT PTP Sync Frame Transmitted
Indicates a PTP sync frame has been transmitted.

Cleared on read.

Bit 20 – DRQFT PTP Delay Request Frame Transmitted
Indicates a PTP delay_req frame has been transmitted.

Cleared on read.

Bit 19 – SFR PTP Sync Frame Received
Indicates a PTP sync frame has been received.

Cleared on read.

Bit 18 – DRQFR PTP Delay Request Frame Received
Indicates a PTP delay_req frame has been received.

Cleared on read.

Bit 14 – PFTR Pause Frame Transmitted
Indicates a pause frame has been successfully transmitted after being initiated from the Network Control Register.

Cleared on read.

Bit 13 – PTZ Pause Time Zero
Set when either the Pause Time Register at address 0x38 decrements to zero, or when a valid pause frame is received with a zero pause quantum field.

Cleared on read.

Bit 12 – PFNZ Pause Frame with Non-zero Pause Quantum Received
Indicates a valid pause has been received that has a non-zero pause quantum field.

Cleared on read.

Bit 11 – HRESP HRESP Not OK
Set when the DMA block sees HRESP not OK.

Offset	Name	Bit Pos.								
0x0180	USBHS_DEVEPTICR8	7:0	SHORTPACKETC	STALLEDIC	OVERFIC	NAKINIC	NAKOUTIC	RXSTPIC	RXOUTIC	TXINIC
		15:8								
		23:16								
		31:24								
0x0180	USBHS_DEVEPTICR8 (ISOENPT)	7:0	SHORTPACKETC	CRCERRIC	OVERFIC	HBISOFLUSHIC	HBISOINERRIC	UNDERFIC	RXOUTIC	TXINIC
		15:8								
		23:16								
		31:24								
0x0184	USBHS_DEVEPTICR9	7:0	SHORTPACKETC	STALLEDIC	OVERFIC	NAKINIC	NAKOUTIC	RXSTPIC	RXOUTIC	TXINIC
		15:8								
		23:16								
		31:24								
0x0184	USBHS_DEVEPTICR9 (ISOENPT)	7:0	SHORTPACKETC	CRCERRIC	OVERFIC	HBISOFLUSHIC	HBISOINERRIC	UNDERFIC	RXOUTIC	TXINIC
		15:8								
		23:16								
		31:24								
0x0188 ... 0x018F	Reserved									
0x0190	USBHS_DEVEPTIFR0	7:0	SHORTPACKETS	STALLEDIS	OVERFIS	NAKINIS	NAKOUTIS	RXSTPIS	RXOUTIS	TXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0190	USBHS_DEVEPTIFR0 (ISOENPT)	7:0	SHORTPACKETS	CRCERRIS	OVERFIS	HBISOFLUSHIS	HBISOINERRIS	UNDERFIS	RXOUTIS	TXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0194	USBHS_DEVEPTIFR1	7:0	SHORTPACKETS	STALLEDIS	OVERFIS	NAKINIS	NAKOUTIS	RXSTPIS	RXOUTIS	TXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0194	USBHS_DEVEPTIFR1 (ISOENPT)	7:0	SHORTPACKETS	CRCERRIS	OVERFIS	HBISOFLUSHIS	HBISOINERRIS	UNDERFIS	RXOUTIS	TXINIS
		15:8				NBUSYBKS				
		23:16								
		31:24								
0x0198	USBHS_DEVEPTIFR2	7:0	SHORTPACKETS	STALLEDIS	OVERFIS	NAKINIS	NAKOUTIS	RXSTPIS	RXOUTIS	TXINIS
		15:8				NBUSYBKS				
		23:16								

39.6.42 Host Pipe Register

Name: USBHS_HSTPIP
Offset: 0x0041C
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
								PRST8
Access								
Reset								0

Bit	23	22	21	20	19	18	17	16
	PRST7	PRST6	PRST5	PRST4	PRST3	PRST2	PRST1	PRST0
Access								
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
								PEN8
Access								
Reset								0

Bit	7	6	5	4	3	2	1	0
	PEN7	PEN6	PEN5	PEN4	PEN3	PEN2	PEN1	PEN0
Access								
Reset	0	0	0	0	0	0	0	0

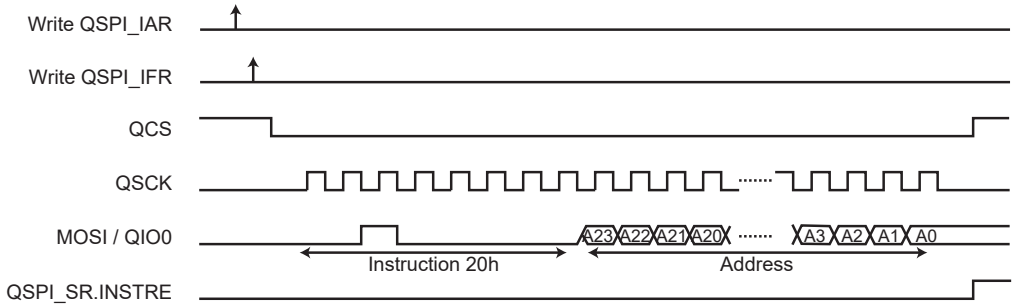
Bits 16, 17, 18, 19, 20, 21, 22, 23, 24 – PRST Pipe x Reset

Value	Description
0	Completes the reset operation and allows to start using the FIFO.
1	Resets the Pipe x FIFO. This resets the pipe x registers (USBHS_HSTPIPCFGx, USBHS_HSTPIPISRx, USBHS_HSTPIPIMRx), but not the pipe configuration (ALLOC, PBK, PSIZE, PTOKEN, PTYPE, PEPNUM, INTFRQ). The whole pipe mechanism (FIFO counter, reception, transmission, etc.) is reset, apart from the Data Toggle management. The pipe configuration remains active and the pipe is still enabled.

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8 – PEN Pipe x Enable

Value	Description
0	Disables Pipe x, which forces the Pipe x state to inactive and resets the pipe x registers (USBHS_HSTPIPCFGx, USBHS_HSTPIPISRx, USBHS_HSTPIPIMRx), but not the pipe configuration (USBHS_HSTPIPCFGx.ALLOC, USBHS_HSTPIPCFGx.PBK, USBHS_HSTPIPCFGx.PSIZE).
1	Enables Pipe x.

Figure 42-13. Instruction Transmission Waveform 3



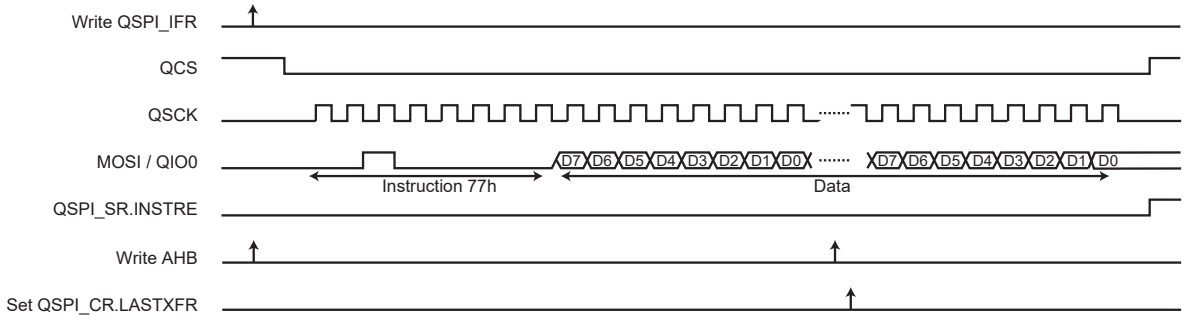
Example 4:

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

Command: SET BURST (77h)

- Write 0x0000_0077 in QSPI_ICR.
- Write 0x0000_2090 in QSPI_IFR.
- Read QSPI_IFR (dummy read) to synchronize system bus accesses.
- Write data in the system bus memory space (0x80000000).
  The address of system bus write accesses is not used.
- Write a ‘1’ to QSPI_CR.LASTXFR.
- Wait for QSPI_SR.INSTRE to rise.

Figure 42-14. Instruction Transmission Waveform 4



Example 5:

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

Command: BYTE/PAGE PROGRAM (02h)

- Write 0x0000_0002 in QSPI_ICR.
- Write 0x0000_30B3 in QSPI_IFR.
- Read QSPI_IFR (dummy read) to synchronize system bus accesses.
- Write data in the QSPI system bus memory space (0x80000000).
  The address of the first system bus write access is sent in the instruction frame.
  The address of the next system bus write accesses is not used.
- Write a ‘1’ to QSPI_CR.LASTXFR.
- Wait for QSPI_SR.INSTRE to rise.

42.7.1 QSPI Control Register

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

Bit	31	30	29	28	27	26	25	24
								LASTXFER
Access								W
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
	SWRST						QSPIDIS	QSPIEN
Access	W						W	W
Reset	–						–	–

Bit 24 – LASTXFER Last Transfer

Value	Description
0	No effect.
1	The chip select is deasserted after the character written in QSPI_TDR.TD has been transferred.

Bit 7 – SWRST QSPI Software Reset
DMA channels are not affected by software reset.

Value	Description
0	No effect.
1	Reset the QSPI. A software-triggered hardware reset of the QSPI interface is performed.

Bit 1 – QSPIDIS QSPI Disable
As soon as QSPIDIS is set, the QSPI finishes its transfer.

All pins are set in Input mode and no data is received or transmitted.

If a transfer is in progress, the transfer is finished before the QSPI is disabled.

If both QSPIEN and QSPIDIS are equal to one when QSPI_CR is written, the QSPI is disabled.

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

Value	Description
0	No overrun has been detected since the last read of QSPI_SR.
1	At least one overrun error has occurred since the last read of QSPI_SR.

Bit 2 – TXEMPTY Transmission Registers Empty (cleared by writing QSPI_TDR)

Value	Description
0	As soon as data is written in QSPI_TDR.
1	QSPI_TDR and the internal shifter are empty. If a transfer delay has been defined, TXEMPTY is set after the completion of such delay.

Bit 1 – TDRE Transmit Data Register Empty (cleared by writing QSPI_TDR)
TDRE equals zero when the QSPI is disabled or at reset. The QSPI enable command sets this bit to one.

Value	Description
0	Data has been written to QSPI_TDR and not yet transferred to the serializer.
1	The last data written in the QSPI_TDR has been transferred to the serializer.

Bit 0 – RDRF Receive Data Register Full (cleared by reading QSPI_RDR)

Value	Description
0	No data has been received since the last read of QSPI_RDR.
1	Data has been received and the received data has been transferred from the serializer to QSPI_RDR since the last read of QSPI_RDR.

46.7.24 USART Receiver Timeout Register

Name: US_RTOR
Offset: 0x0024
Reset: 0x0
Property: Read/Write

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

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

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

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

Bits 16:0 – TO[16:0] Timeout Value

Value	Description
0	The receiver timeout is disabled.
1-65535	The receiver timeout is enabled and TO is Timeout Delay / Bit Period.
1-131071	The receiver timeout is enabled and TO is Timeout Delay / Bit Period.

49.6.28 MCAN Receive FIFO 0 Status

Name: MCAN_RXF0S
Offset: 0xA4
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
							RF0L	F0F
Access							R	R
Reset							0	0

Bit	23	22	21	20	19	18	17	16
			F0PI[5:0]					
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			F0GI[5:0]					
Access			R	R	R	R	R	R
Reset			0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		F0FL[6:0]						
Access		R	R	R	R	R	R	R
Reset		0	0	0	0	0	0	0

Bit 25 – RF0L Receive FIFO 0 Message Lost
This bit is a copy of interrupt flag MCAN_IR.RF0L. When MCAN_IR.RF0L is reset, this bit is also reset.

Overwriting the oldest message when MCAN_RXF0C.F0OM = ‘1’ will not set this flag.

Value	Description
0	No Receive FIFO 0 message lost
1	Receive FIFO 0 message lost, also set after write attempt to Receive FIFO 0 of size zero

Bit 24 – F0F Receive FIFO 0 Full

Value	Description
0	Receive FIFO 0 not full.
1	Receive FIFO 0 full.

Bits 21:16 – F0PI[5:0] Receive FIFO 0 Put Index
Receive FIFO 0 write index pointer, range 0 to 63.

Bits 13:8 – F0GI[5:0] Receive FIFO 0 Get Index
Receive FIFO 0 read index pointer, range 0 to 63.

Bits 6:0 – F0FL[6:0] Receive FIFO 0 Fill Level
Number of elements stored in Receive FIFO 0, range 0 to 64.

51.6.2.9.2 Method 2: Manual write of duty-cycle values and automatic trigger of the update

In this mode, the update of the period value, the duty-cycle values, the dead-time values and the update period value must be done by writing in their respective update registers with the processor (respectively PWM_CPRDUPDx, PWM_CDTYUPDx, PWM_DTUPDx and PWM_SCUPUPD).

To trigger the update of the period value and the dead-time values, the user must use the bit UPDULOCK in the PWM_SCUC register, which updates synchronously (at the same PWM period) the synchronous channels:

- If the bit UPDULOCK is set to '1', the update is done at the next PWM period of the synchronous channels.
- If the UPDULOCK bit is not set to '1', the update is locked and cannot be performed.

After writing the UPDULOCK bit to '1', it is held at this value until the update occurs, then it is read 0.

The update of the duty-cycle values and the update period is triggered automatically after an update period.

To configure the automatic update, the user must define a value for the update period by the UPR field in the PWM_SCUP register. The PWM controller waits UPR+1 period of synchronous channels before updating automatically the duty values and the update period value.

The status of the duty-cycle value write is reported in the PWM Interrupt Status Register 2 (PWM_ISR2) by the following flags:

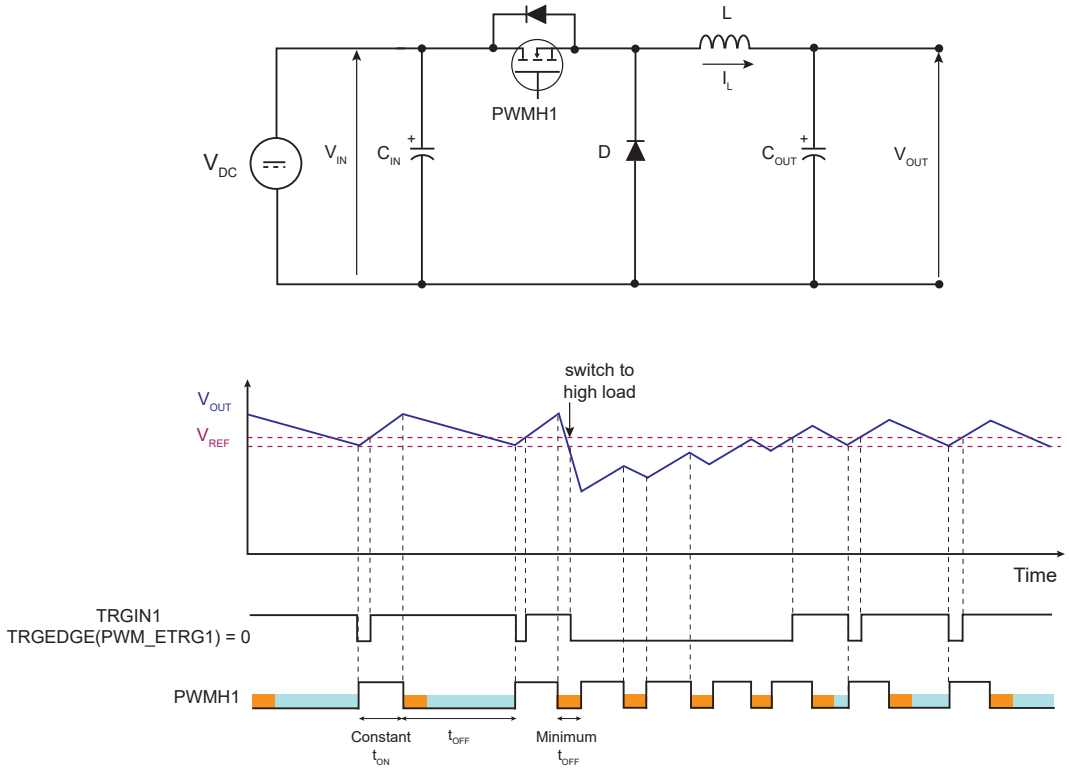
- WRDY: this flag is set to '1' when the PWM Controller is ready to receive new duty-cycle values and a new update period value. It is reset to '0' when the PWM_ISR2 register is read.

Depending on the interrupt mask in the PWM Interrupt Mask Register 2 (PWM_IMR2), an interrupt can be generated by these flags.

Sequence for Method 2:

1. Select the manual write of duty-cycle values and the automatic update by setting the field UPDM to '1' in the PWM_SCM register
2. Define the synchronous channels by the bits SYNCx in the PWM_SCM register.
3. Define the update period by the field UPR in the PWM_SCUP register.
4. Enable the synchronous channels by writing CHID0 in the PWM_ENA register.
5. If an update of the period value and/or of the dead-time values is required, write registers that need to be updated (PWM_CPRDUPDx, PWM_DTUPDx), else go to Step 8.
6. Set UPDULOCK to '1' in PWM_SCUC.
7. The update of these registers will occur at the beginning of the next PWM period. At this moment the bit UPDULOCK is reset, go to Step 5. for new values.
8. If an update of the duty-cycle values and/or the update period is required, check first that write of new update values is possible by polling the flag WRDY (or by waiting for the corresponding interrupt) in PWM_ISR2.
9. Write registers that need to be updated (PWM_CDTYUPDx, PWM_SCUPUPD).
10. The update of these registers will occur at the next PWM period of the synchronous channels when the Update Period is elapsed. Go to Step 8. for new values.

Figure 51-30. External PWM Start Mode: Buck DC/DC Converter



51.6.5.3 Cycle-By-Cycle Duty Mode

51.6.5.3.1 Description

Cycle-by-cycle duty mode is selected by programming TRGMODE = 3 in PWM_ETRGx.

In this mode, the PWM frequency is constant and is defined by the CPRD value in the PWM Channel Period Register.

An external trigger event has no effect on the PWM output if it occurs while the internal PWM counter value is above the CDTY value of the PWM Channel Duty Cycle Register.

If the internal PWM counter value is below the value of CDTY of the PWM Channel Duty Cycle Register, an external trigger event makes the PWM output inactive.

The external trigger event can be detected on rising or falling edge according to the TRGEDGE bit in PWM_ETRGx.

When the TXRDY flag of a channel in the DACC_ISR is active, the DACC is ready to accept conversion requests by writing data into the corresponding DACC_CDRx. Data which cannot be converted immediately are stored in the FIFO of the corresponding channel.

When the FIFO is full or the DACC is not ready to accept conversion requests, the TXRDY flag is inactive.

The DACC also offers the possibility of writing two data words in one access by setting the bit WORD in the DACC_MR. In this case, bits 11:0 contain the first data to be converted and bits 27:16 contain the second data to be converted. The two data are written into the FIFO of the selected channel. The TXRDY flag takes into account this double write access. Changing this access mode implies first switching off all channels.

WARNING Writing in DACC_CDRx while TXRDY flag is inactive will corrupt FIFO data.

53.6.6 Register Write Protection

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

If a write access to a write-protected register is detected, the WPVS bit in the DACC Write Protection Status Register (DACC_WPSR) is set and the field WPVSRC indicates the register in which the write access has been attempted.

The WPVS bit is automatically cleared after reading the DACC_WPSR.

The following registers can be write-protected :

- DACC Mode Register
- DACC Channel Enable Register
- DACC Channel Disable Register
- DACC Analog Current Register
- DACC Trigger Register

Offset	Name	Bit Pos.								
0x30	AES_KEYWR4	7:0	KEYW[7:0]							
		15:8	KEYW[15:8]							
		23:16	KEYW[23:16]							
		31:24	KEYW[31:24]							
0x34	AES_KEYWR5	7:0	KEYW[7:0]							
		15:8	KEYW[15:8]							
		23:16	KEYW[23:16]							
		31:24	KEYW[31:24]							
0x38	AES_KEYWR6	7:0	KEYW[7:0]							
		15:8	KEYW[15:8]							
		23:16	KEYW[23:16]							
		31:24	KEYW[31:24]							
0x3C	AES_KEYWR7	7:0	KEYW[7:0]							
		15:8	KEYW[15:8]							
		23:16	KEYW[23:16]							
		31:24	KEYW[31:24]							
0x40	AES_IDATAR0	7:0	IDATA[7:0]							
		15:8	IDATA[15:8]							
		23:16	IDATA[23:16]							
		31:24	IDATA[31:24]							
0x44	AES_IDATAR1	7:0	IDATA[7:0]							
		15:8	IDATA[15:8]							
		23:16	IDATA[23:16]							
		31:24	IDATA[31:24]							
0x48	AES_IDATAR2	7:0	IDATA[7:0]							
		15:8	IDATA[15:8]							
		23:16	IDATA[23:16]							
		31:24	IDATA[31:24]							
0x4C	AES_IDATAR3	7:0	IDATA[7:0]							
		15:8	IDATA[15:8]							
		23:16	IDATA[23:16]							
		31:24	IDATA[31:24]							
0x50	AES_ODATAR0	7:0	ODATA[7:0]							
		15:8	ODATA[15:8]							
		23:16	ODATA[23:16]							
		31:24	ODATA[31:24]							
0x54	AES_ODATAR1	7:0	ODATA[7:0]							
		15:8	ODATA[15:8]							
		23:16	ODATA[23:16]							
		31:24	ODATA[31:24]							
0x58	AES_ODATAR2	7:0	ODATA[7:0]							
		15:8	ODATA[15:8]							
		23:16	ODATA[23:16]							
		31:24	ODATA[31:24]							
0x5C	AES_ODATAR3	7:0	ODATA[7:0]							
		15:8	ODATA[15:8]							