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-TFBGA
Supplier Device Package	100-TFBGA (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70n21b-cnt

37.1. Description ... 590
37.2. Embedded Characteristics ... 591
37.3. Block Diagram ... 592
37.4. Product Dependencies ... 592
37.5. Functional Description ... 592
37.6. Register Summary ... 602

38. GMAC - Ethernet MAC ... 640
38.1. Description ... 640
38.2. Embedded Characteristics ... 640
38.3. Block Diagram ... 641
38.4. Signal Interface ... 641
38.5. Product Dependencies ... 642
38.6. Functional Description ... 642
38.7. Programming Interface ... 673
38.8. Register Summary ... 678

39. USB High-Speed Interface (USBHS) ... 831
39.1. Description ... 831
39.2. Embedded Characteristics ... 831
39.3. Block Diagram ... 832
39.4. Product Dependencies ... 833
39.5. Functional Description ... 834
39.6. Register Summary ... 859

40. High-Speed Multimedia Card Interface (HSMCI) ... 1042
40.1. Description ... 1042
40.2. Embedded Characteristics ... 1042
40.3. Block Diagram ... 1043
40.4. Application Block Diagram ... 1043
40.5. Pin Name List ... 1044
40.6. Product Dependencies ... 1044
40.7. Bus Topology ... 1044
40.8. High-Speed Multimedia Card Operations ... 1046
40.9. SD/SDIO Card Operation ... 1055
40.10. CE-ATA Operation ... 1056
40.11. HSMCI Boot Operation Mode ... 1057
40.12. HSMCI Transfer Done Timings ... 1058
40.13. Register Write Protection ... 1059
40.14. Register Summary ... 1060

41. Serial Peripheral Interface (SPI) ... 1095
41.1. Description ... 1095
41.2. Embedded Characteristics ... 1095
41.3. Block Diagram ... 1096
41.4. Application Block Diagram ... 1096
41.5. Signal Description ... 1097
41.6. Product Dependencies ... 1097

Value	Name	Description
4	16BEAT_BURST	16-beat Burst—The undefined length burst or bursts sequence is split into 16-beat bursts or less, allowing re-arbitration every 16 beats.
5	32BEAT_BURST	32-beat Burst —The undefined length burst or bursts sequence is split into 32-beat bursts or less, allowing re-arbitration every 32 beats.
6	64BEAT_BURST	64-beat Burst—The undefined length burst or bursts sequence is split into 64-beat bursts or less, allowing re-arbitration every 64 beats.
7	128BEAT_BURST	128-beat Burst—The undefined length burst or bursts sequence is split into 128-beat bursts or less, allowing re-arbitration every 128 beats. Note: Unless duly needed, the ULBT should be left at its default 0 value for power saving.

25.5.2 Reinforced Safety Watchdog Timer Mode Register

Name: RSWDT_MR
Offset: 0x04
Reset: 0x3FFFAFFF
Property: Read/Write Once

Note: The first write access prevents any further modification of the value of this register; read accesses remain possible.

Note: The WDV value must not be modified within three slow clock periods following a restart of the watchdog performed by means of a write access in the RSWDT_CR, else the watchdog may trigger an end of period earlier than expected.

Bit	31	30	29	28	27	26	25	24
			WDIDLEHLT	WDDBGHLT	ALLONES[11:8]			
Access								
Reset			1	1	1	1	1	1

Bit	23	22	21	20	19	18	17	16
	ALLONES[7:0]							
Access								
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								
Reset	1		1	0	1	1	1	1

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

Bit 29 – WDIDLEHLT Watchdog Idle Halt

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

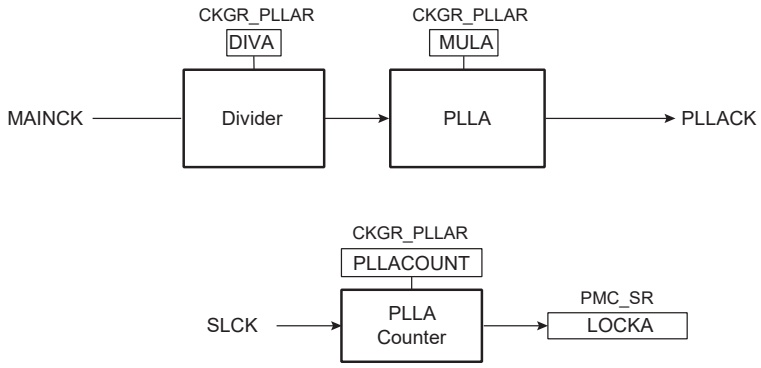
Bit 28 – WDDBGHLT Watchdog Debug Halt

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

Bits 27:16 – ALLONES[11:0] Must Always Be Written with 0xFFF

Bit 15 – WDDIS Watchdog Disable

Figure 30-4. Divider and PLLA Block Diagram



30.6.1 Divider and Phase Lock Loop Programming

The divider can be set between 1 and 255 in steps of 1. When a divider field (DIV) is cleared, the output of the corresponding divider and the PLL output is a continuous signal at level 0. On reset, each DIV field is cleared, thus the corresponding PLL input clock is stuck at ‘0’.

The PLL (PLLA) allows multiplication of the divider’s outputs. The PLL clock signal has a frequency that depends on the respective source signal frequency and on the parameters DIV (DIVA) and MUL (MULA). The factor applied to the source signal frequency is (MUL + 1)/DIV. When MUL is written to ‘0’ or DIV = 0, the PLL is disabled and its power consumption is saved. Note that there is a delay of two SLCK clock cycles between the disable command and the real disable of the PLL. Re-enabling the PLL can be performed by writing a value higher than ‘0’ in the MUL field and DIV higher than ‘0’.

Whenever the PLL is re-enabled or one of its parameters is changed, the LOCK (LOCKA) bit in PMC_SR is automatically cleared. The values written in the PLLCOUNT field (PLLACOUNT) in CKGR_PLLR (CKGR_PLLAR) are loaded in the PLL counter. The PLL counter then decrements at the speed of SLCK until it reaches ‘0’. At this time, PMC_SR.LOCK is set and can trigger an interrupt to the processor. The user has to load the number of SLCK cycles required to cover the PLL transient time into the PLLCOUNT field.

To avoid programming the PLL with a multiplication factor that is too high, the user can saturate the multiplication factor value sent to the PLL by setting the PLLA_MMAX field in the PLL Maximum Multiplier Value Register (PMC_PMMR).

It is forbidden to change the MAINCK characteristics (oscillator selection, frequency adjustment of the Main RC oscillator) when:

- MAINCK is selected as the PLLA clock source, and
- MCK is sourced from PLLA.

To change the MAINCK characteristics, the user must:

1. Switch the MCK source to MAINCK by writing a ‘1’ to PMC_MCKR.CSS.
2. Change the Main RC oscillator frequency (MOSCRCF) or oscillator selection (MOSCSEL) in CKGR_MOR.
3. Wait for MOSCRCS (if frequency changes) or MOSCSELS (if oscillator selection changes) in PMC_SR.
4. Disable and then enable the PLL.
5. Wait for the LOCK flag in PMC_SR.
6. Switch back MCK to the PLLA by writing the appropriate value to PMC_MCKR.CSS.

36.9.4 XDMAC Global Interrupt Enable Register

Name: XDMAC_GIE
Offset: 0x0C
Reset: –
Property: Write-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	IE23	IE22	IE21	IE20	IE19	IE18	IE17	IE16
Access	W	W	W	W	W	W	W	W
Reset	–	–	–	–	–	–	–	–

Bit	15	14	13	12	11	10	9	8
	IE15	IE14	IE13	IE12	IE11	IE10	IE9	IE8
Access	W	W	W	W	W	W	W	W
Reset	–	–	–	–	–	–	–	–

Bit	7	6	5	4	3	2	1	0
	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
Access	W	W	W	W	W	W	W	W
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 – IE XDMAC Channel x Interrupt Enable

Value	Description
0	This bit has no effect. The Channel x Interrupt Mask bit (XDMAC_GIM.IMx) is not modified.
1	The corresponding mask bit is set. The XDMAC Channel x Interrupt Status register (XDMAC_GIS) can generate an interrupt.

Table 37-5. RGB Format in Default Mode, RGB_CFG = 00, Swap Activated

Mode	Byte	D7	D6	D5	D4	D3	D2	D1	D0
RGB 8:8:8	Byte 0	R0(i)	R1(i)	R2(i)	R3(i)	R4(i)	R5(i)	R6(i)	R7(i)
	Byte 1	G0(i)	G1(i)	G2(i)	G3(i)	G4(i)	G5(i)	G6(i)	G7(i)
	Byte 2	B0(i)	B1(i)	B2(i)	B3(i)	B4(i)	B5(i)	B6(i)	B7(i)
	Byte 3	R0(i+1)	R1(i+1)	R2(i+1)	R3(i+1)	R4(i+1)	R5(i+1)	R6(i+1)	R7(i+1)
RGB 5:6:5	Byte 0	G3(i)	G4(i)	G5(i)	R0(i)	R1(i)	R2(i)	R3(i)	R4(i)
	Byte 1	B0(i)	B1(i)	B2(i)	B3(i)	B4(i)	G0(i)	G1(i)	G2(i)
	Byte 2	G3(i+1)	G4(i+1)	G5(i+1)	R0(i+1)	R1(i+1)	R2(i+1)	R3(i+1)	R4(i+1)
	Byte 3	B0(i+1)	B1(i+1)	B2(i+1)	B3(i+1)	B4(i+1)	G0(i+1)	G1(i+1)	G2(i+1)

The RGB 5:6:5 input format is processed to be displayed as RGB 5:6:5 format, compliant with the 16-bit mode of the LCD controller.

37.5.3 Clocks

The sensor master clock (ISI_MCK) can be generated either by the Advanced Power Management Controller (APMC) through a Programmable Clock output or by an external oscillator connected to the sensor.

None of the sensors embed a power management controller, so providing the clock by the APMC is a simple and efficient way to control power consumption of the system.

Care must be taken when programming the system clock. The ISI has two clock domains, the sensor master clock and the pixel clock provided by sensor. The two clock domains are not synchronized, but the sensor master clock must be faster than the pixel clock.

37.5.4 Preview Path

37.5.4.1 Scaling, Decimation (Subsampling)

This module resizes captured 8-bit color sensor images to fit the LCD display format. The resize module performs only downscaling. The same ratio is applied for both horizontal and vertical resize, then a fractional decimation algorithm is applied.

The decimation factor is a multiple of 1/16; values 0 to 15 are forbidden.

Table 37-6. Decimation Factor

Decimation Value	0–15	16	17	18	19	...	124	125	126	127
Decimation Factor	—	1	1.063	1.125	1.188	...	7.750	7.813	7.875	7.938

Table 37-7. Decimation and Scaler Offset Values

OUTPUT	INPUT	352 × 288	640 × 480	800 × 600	1280 × 1024	1600 × 1200	2048 × 1536
VGA 640 × 480	F	—	16	20	32	40	51
QVGA	F	16	32	40	64	80	102

The bank is really killed: USBHS_DEVEPTISRx.NBUSYBK is decremented.

The bank is not cleared but sent (IN transfer): USBHS_DEVEPTISRx.NBUSYBK is decremented.

The bank is not cleared because it was empty.

The user should wait for this bit to be cleared before trying to kill another packet.

This kill request is refused if at the same time an IN token is coming and the last bank is the current one being sent on the USB line. If at least two banks are ready to be sent, there is no problem to kill a packet even if an IN token is coming. Indeed, in this case, the current bank is sent (IN transfer) while the last bank is killed.

Bit 12 – NBUSYBKE Number of Busy Banks Interrupt

Value	Description
0	Cleared when USBHS_DEVEPTIDRx.NBUSYBKEC = 0. This disables the Number of Busy Banks interrupt (USBHS_DEVEPTISRx.NBUSYBK).
1	Set when the USBHS_DEVEPTIERx.NBUSYBKES = 1. This enables the Number of Busy Banks interrupt (USBHS_DEVEPTISRx.NBUSYBK).

Bit 7 – SHORTPACKETE Short Packet Interrupt
If this bit is set for non-control IN endpoints, a short packet transmission is guaranteed upon ending a DMA transfer, thus signaling an end of isochronous frame or a bulk or interrupt end of transfer, provided that the End of DMA Buffer Output Enable (END_B_EN) bit and the Automatic Switch (AUTOSW) = 1.

Value	Description
0	Cleared when USBHS_DEVEPTIDRx.SHORTPACKETEC = 1. This disables the Short Packet interrupt (USBHS_DEVEPTISRx.SHORTPACKET).
1	Set when USBHS_DEVEPTIERx.SHORTPACKETES = 1. This enables the Short Packet interrupt (USBHS_DEVEPTISRx.SHORTPACKET).

Bit 6 – STALLEDE STALLed Interrupt

Value	Description
0	Cleared when USBHS_DEVEPTIDRx.STALLEDEC = 1. This disables the STALLed interrupt (USBHS_DEVEPTISRx.STALLEDI).
1	Set when USBHS_DEVEPTIERx.STALLEDES = 1. This enables the STALLed interrupt (USBHS_DEVEPTISRx.STALLEDI).

Bit 5 – OVERFE Overflow Interrupt

Value	Description
0	Cleared when USBHS_DEVEPTIDRx.OVERFEC = 1. This disables the Overflow interrupt (USBHS_DEVEPTISRx.OVERFI).
1	Set when USBHS_DEVEPTIERx.OVERFES = 1. This enables the Overflow interrupt (USBHS_DEVEPTISRx.OVERFI).

Bit 4 – NAKINE NAKed IN Interrupt

39.6.55 Host Pipe x Mask Register (Interrupt Pipes)

Name: USBHS_HSTPIPIMRx (INTPIPES)
Offset: 0x05C0 + x*0x04 [x=0..9]
Reset: 0
Property: Read/Write

This register view is relevant only if PTYPE = 0x3 in "Host Pipe x Configuration Register".

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

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

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

Bit	7	6	5	4	3	2	1	0
	SHORTPACKETIE	RXSTALLDE	OVERFIE	NAKEDE	PERRE	UNDERFIE	TXOUTE	RXINE
Access								
Reset	0	0	0	0	0	0	0	0

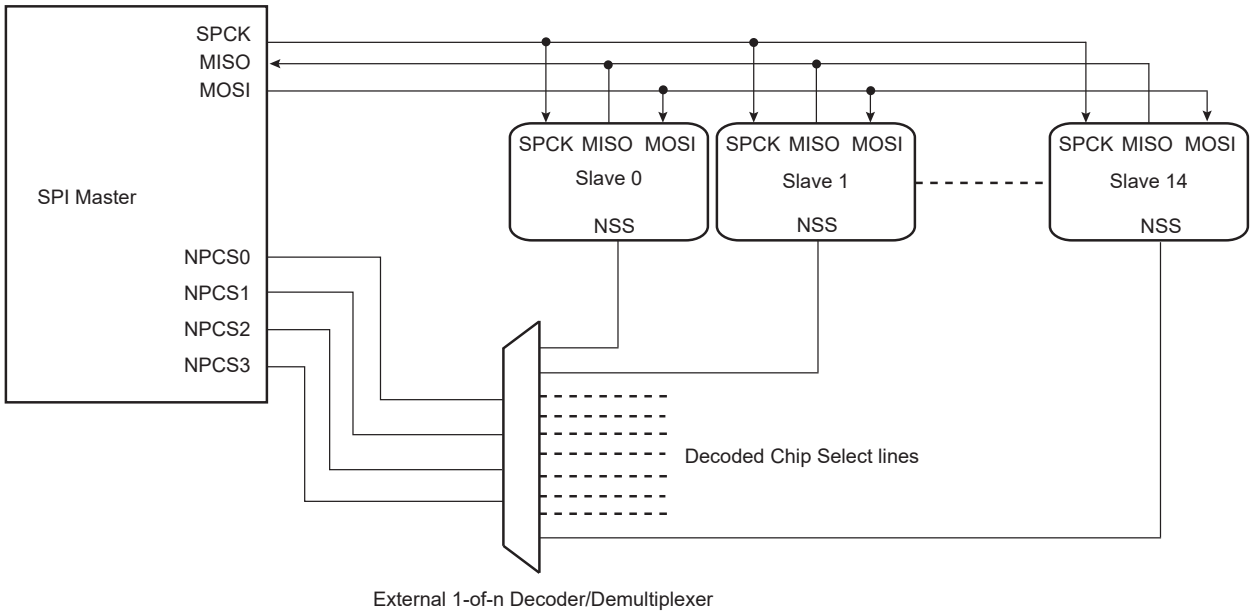
Bit 18 – RSTDT Reset Data Toggle

Value	Description
0	0: No reset of the Data Toggle is ongoing.
1	Set when USBHS_HSTPIPIER.RSTDTS = 1. This resets the Data Toggle to its initial value for the current pipe.

Bit 17 – PFREEZE Pipe Freeze
This freezes the pipe request generation.

Value	Description
0	Cleared when USBHS_HSTPIPIDR.PFREEZEC = 1. This enables the pipe request generation.
1	Set when one of the following conditions is met: • USBHS_HSTPIPIER.PFREEZES = 1 • The pipe is not configured. • A STALL handshake has been received on the pipe. • An error has occurred on the pipe (USBHS_HSTPIPISR.PERRI = 1).

Figure 41-10. Chip Select Decoding Application Block Diagram: Single Master/Multiple Slave Implementation



41.7.3.8 Peripheral Deselection without DMA

During a transfer of more than one unit of data on a chip select without the DMA, SPI_TDR is loaded by the processor, the TDRE flag rises as soon as the content of SPI_TDR is transferred into the internal shift register. When this flag is detected high, SPI_TDR can be reloaded. If this reload by the processor occurs before the end of the current transfer and if the next transfer is performed on the same chip select as the current transfer, the chip select is not deasserted between the two transfers. But depending on the application software handling the SPI status register flags (by interrupt or polling method) or servicing other interrupts or other tasks, the processor may not reload SPI_TDR in time to keep the chip select active (low). A null DLYBCT value (delay between consecutive transfers) in SPI_CSR, gives even less time for the processor to reload SPI_TDR. With some SPI slave peripherals, if the chip select line must remain active (low) during a full set of transfers, communication errors can occur.

To facilitate interfacing with such devices, the chip select registers [SPI_CSR0...SPI_CSR3] can be programmed with the Chip Select Active After Transfer (CSAAT) bit at 1. This allows the chip select lines to remain in their current state (low = active) until a transfer to another chip select is required. Even if SPI_TDR is not reloaded, the chip select remains active. To deassert the chip select line at the end of the transfer, the Last Transfer (LASTXFER) bit in SPI_CR must be set after writing the last data to transmit into SPI_TDR.

41.7.3.9 Peripheral Deselection with DMA

DMA provides faster reloads of SPI_TDR compared to software. However, depending on the system activity, it is not guaranteed that SPI_TDR is written with the next data before the end of the current transfer. Consequently, data can be lost by the deassertion of the NPCS line for SPI slave peripherals requiring the chip select line to remain active between two transfers. The only way to guarantee a safe transfer in this case is the use of the CSAAT and LASTXFER bits.

When the CSAAT bit is configured to 0, the NPCS does not rise in all cases between two transfers on the same peripheral. During a transfer on a chip select, the TDRE flag rises as soon as the content of SPI_TDR is transferred into the internal shift register. When this flag is detected, SPI_TDR can be reloaded. If this reload occurs before the end of the current transfer and if the next transfer is performed on the same chip select as the current transfer, the chip select is not deasserted between the two

44.9.18 SSC Write Protection Status Register

Name: SSC_WPSR
Offset: 0xE8
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
	WPVSRC[15:8]							
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
	WPVSRC[7:0]							
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
								WPVS
Access								R
Reset								0

Bits 23:8 – WPVSRC[15:0] Write Protect Violation Source
When WPVS = 1, WPVSRC indicates the register address offset at which a write access has been attempted.

Bit 0 – WPVS Write Protection Violation Status

Value	Description
0	No write protection violation has occurred since the last read of the SSC_WPSR.
1	A write protection violation has occurred since the last read of the SSC_WPSR. If this violation is an unauthorized attempt to write a protected register, the associated violation is reported into field WPVSRC.

46.6.10.9 LON Errors

All these flags can be read in the Channel Status register (LON_MODE) (US_CSR) and will generate interrupts if configured in the Interrupt Enable register (LON_MODE) (US_IER).

These flags can be reset through US_CR.RSTSTA.

46.6.10.9.1 Underrun Error

If the USART is in LON mode and if a character is sent while the Transmit Holding register (US_THR) is empty, the UNRE bit flag is set.

46.6.10.9.2 Collision Detection

The LCOL flag is set whenever a valid collision has been detected and the LON node is configured to report it (see “Collision Detection”).

46.6.10.9.3 LON Frame Early Termination

The LFET flag is set whenever a LON frame has been terminated early due to collision detection.

46.6.10.9.4 Reception Error

The LCRCE flag is set if the received frame has an erroneous CRC and the flag LSFE is set if the received frame is too short (LON frames must be at least 8 bytes long).

These flags can be read in US_CSR.

46.6.10.9.5 Backlog Overflow

The LBLOVFE flag is set if the LON node backlog estimation goes over 63 which is the maximum backlog value.

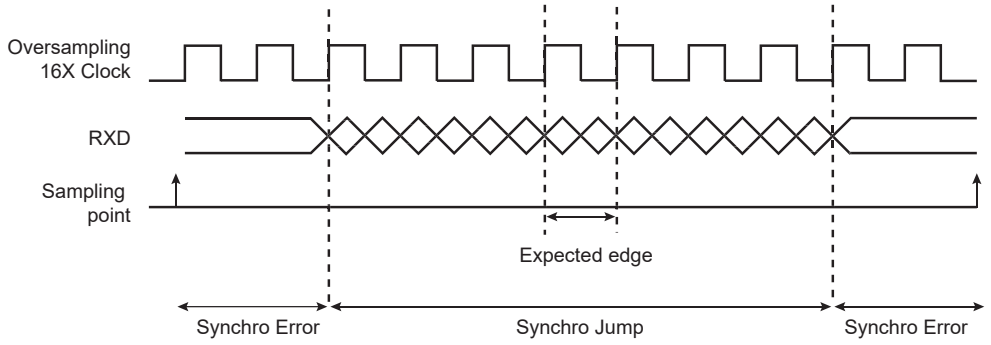
46.6.10.10 Drift Compensation

While receiving a frame, the baud rate used by the sender may not be exactly the one expected. In this case, the hardware drift compensation algorithm recovers up to 16% clock drift (expected baud rate ±16% will be supported).

Drift compensation is available only in 16X Oversampling mode. To enable the hardware system, US_MAN.DRIFT must be set. If the RXD edge is between one and three 16X clock cycles far from the expected edge, then the period is shortened or lengthened accordingly, to center the RXD edge.

The drift compensation hardware feature allows up to 16% clock drift to be handled, provided the system clock is fast enough compared to the selected baud rate.

Figure 46-60. Bit Resynchronization

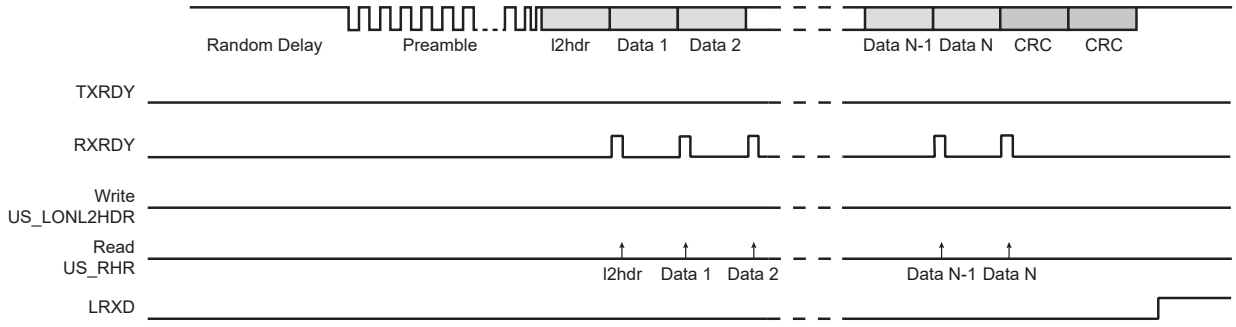


46.6.10.11 LON Frame Handling

46.6.10.11.1 Sending A Frame

1. Write TXEN and RXEN in US_CR to enable both the transmitter and the receiver.
2. Write USART_MODE in US_MR to select the LON mode configuration.

Figure 46-62. Rx Frame



46.6.10.12 LON Frame Handling with the Peripheral DMA Controller

The USART can be used in association with the DMA Controller in order to transfer data directly into/from the on- and off-chip memories without any processor intervention.

The DMA uses the trigger flags, TXRDY and RXRDY, to write or read into the USART. The DMA always writes in US_THR and it always reads in US_RHR. The size of the data written or read by the DMA in the USART is always a byte.

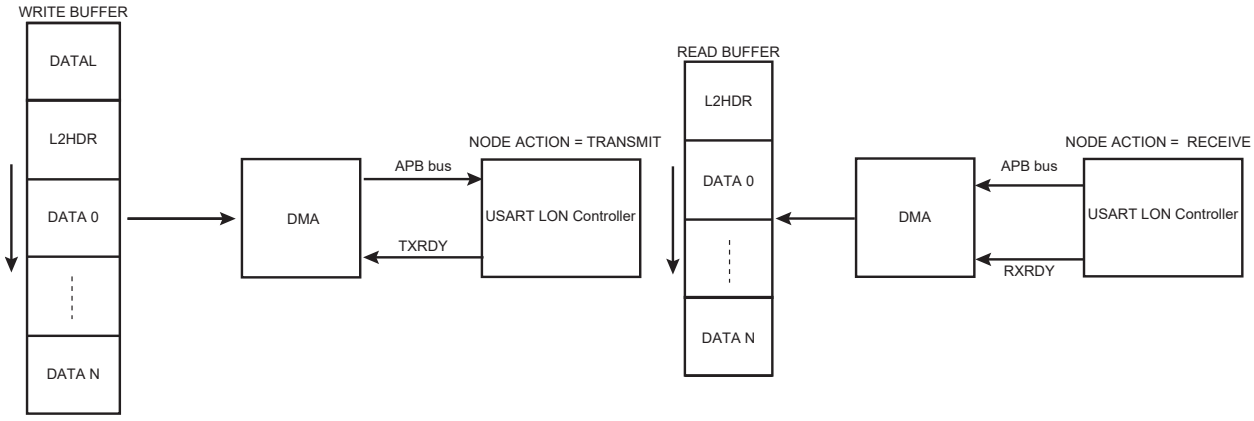
46.6.10.12.1 Configuration

The DMA mode is configured in USLONMR.DMAM:

- DMAM = 1: The LON frame data length (DATAL) is stored in the WRITE buffer and it is written by the DMA in US_THR (instead of the LON Data Length register US_LONDL).
- DMAM = 0: The LON frame data length (DATAL) is not stored in the WRITE buffer and it must be written by the user in US_LONDL.

In both DMA modes L2HDR is considered as a data and its value must be stored in the WRITE buffer as the first data to write.

Figure 46-63. DMAM = 1



46.7.2 USART Control Register (SPI_MODE)

Name: US_CR (SPI_MODE)
Offset: 0x0000
Property: Write-only

This configuration is relevant only if USART_MODE = 0xE or 0xF in the USART Mode Register.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset					RCS	FCS		

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

Bit	7	6	5	4	3	2	1	0
Access								
Reset	TXDIS	TXEN	RXDIS	RXEN	RSTTX	RSTRX		

Bit 19 – RCS Release SPI Chip Select
Applicable if USART operates in SPI Master mode (USART_MODE = 0xE):

Value	Description
0	No effect.
1	Releases the Slave Select Line NSS (RTS pin).

Bit 18 – FCS Force SPI Chip Select
Applicable if USART operates in SPI Master mode (USART_MODE = 0xE):

Value	Description
0	No effect.
1	Forces the Slave Select Line NSS (RTS pin) to 0, even if USART is not transmitting, in order to address SPI slave devices supporting the CSAAT mode (Chip Select Active After Transfer).

Bit 8 – RSTSTA Reset Status Bits

Value	Description
0	No effect.
1	Resets the status bits OVRE, UNRE in US_CSR.

Offset	Name	Bit Pos.								
0xE4	UART_WPMR	7:0								WPEN
		15:8	WPKEY[7:0]							
		23:16	WPKEY[15:8]							
		31:24	WPKEY[23:16]							

49.6.15 MCAN Transmitter Delay Compensation Register

Name: MCAN_TDCR
Offset: 0x48
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
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
		TDCO[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
		TDCF[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

Bits 14:8 – TDCO[6:0] Transmitter Delay Compensation Offset
0 to 127: Offset value, in CAN core clock periods, defining the distance between the measured delay from CANTX to CANRX and the secondary sample point.

Bits 6:0 – TDCF[6:0] Transmitter Delay Compensation Filter
0 to 127: defines the minimum value for the SSP position, in CAN core clock periods. Dominant edges on CANRX that would result in an earlier SSP position are ignored for transmitter delay measurement. The feature is enabled when TDCF is configured to a value greater than TDCO.

49.6.31 MCAN Receive FIFO 1 Configuration

Name: MCAN_RXF1C
Offset: 0xB0
Reset: 0x00000000
Property: Read/Write

This register can only be written if the bits CCE and INIT are set in MCAN CC Control Register.

Bit	31	30	29	28	27	26	25	24
	F1OM	F1WM[6:0]						
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

Bit	23	22	21	20	19	18	17	16
		F1S[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	15	14	13	12	11	10	9	8
	F1SA[13:6]							
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
	F1SA[5:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W		
Reset	0	0	0	0	0	0		

Bit 31 – F1OM FIFO 1 Operation Mode
FIFO 1 can be operated in Blocking or in Overwrite mode (see Rx FIFOs).

Value	Description
0	FIFO 1 Blocking mode.
1	FIFO 1 Overwrite mode.

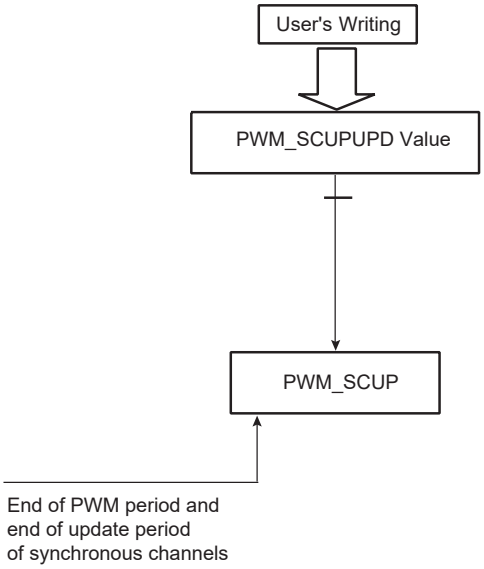
Bits 30:24 – F1WM[6:0] Receive FIFO 1 Watermark

Value	Description
0	Watermark interrupt disabled
1-64	Level for Receive FIFO 1 watermark interrupt (MCAN_IR.RF1W).
>64	Watermark interrupt disabled.

Bits 22:16 – F1S[6:0] Receive FIFO 1 Size
The elements in Receive FIFO 1 are indexed from 0 to F1S - 1.

Value	Description
0	No Receive FIFO 1
1-64	Number of elements in Receive FIFO 1.
>64	Values greater than 64 are interpreted as 64.

Figure 51-35. Synchronized Update of Update Period Value of Synchronous Channels



51.6.6.5 Changing the Comparison Value and the Comparison Configuration

It is possible to change the comparison values and the comparison configurations while the channel 0 is enabled (see PWM Comparison Units).

To prevent unexpected comparison match, the user must use the PWM Comparison x Value Update Register (PWM_CMPVUPDx) and the PWM Comparison x Mode Update Register (PWM_CMPMUPDx) to change, respectively, the comparison values and the comparison configurations while the channel 0 is still enabled. These registers hold the new values until the end of the comparison update period (when CUPRCNT is equal to CUPR in PWM Comparison x Mode Register (PWM_CMPMx) and the end of the current PWM period, then update the values for the next period.

⚠ CAUTION The write of the register PWM_CMPVUPDx must be followed by a write of the register PWM_CMPMUPDx.

Note: If the update registers PWM_CMPVUPDx and PWM_CMPMUPDx are written several times between two updates, only the last written value are taken into account.

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
		ESR	–	–	2	Ohm
t_{ON}	Turn-on Time	C_{DOUT} = 1 µF, V_{DDOUT} reaches DC output voltage	–	1	2.5	ms

Note:

1. A 4.7 µF (±20%) or higher ceramic capacitor must be connected between V_{DDIN} and the closest GND pin of the device. This large decoupling capacitor is mandatory to reduce startup current, improving transient response and noise rejection.
2. To ensure stability, an external 1 µF (±20%) output capacitor, C_{DOUT}, must be connected between V_{DDOUT} and the closest GND pin of the device. Solid tantalum and multilayer ceramic capacitors are suitable as output capacitors. A 100 nF bypass capacitor between V_{DDOUT} and the closest GND pin of the device helps decrease output noise and improves the load transient response.

Table 59-6. Core Power Supply Brownout Detector Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{T-}	Supply Falling Threshold (see **Note 1**)	–	0.97	1.0	1.04	V
V_{hys}	Hysteresis Voltage	–	–	25	50	mV
t_{START}	Startup Time	From disabled state to enabled state	–	–	400	µs

Note:

1. The Brownout Detector is configured using the BODDIS bit in the SUPC_MR register.

Figure 59-1. Core Brownout Output Waveform

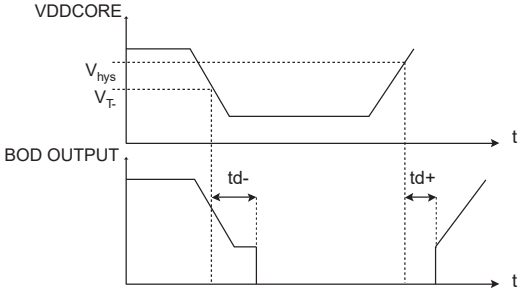
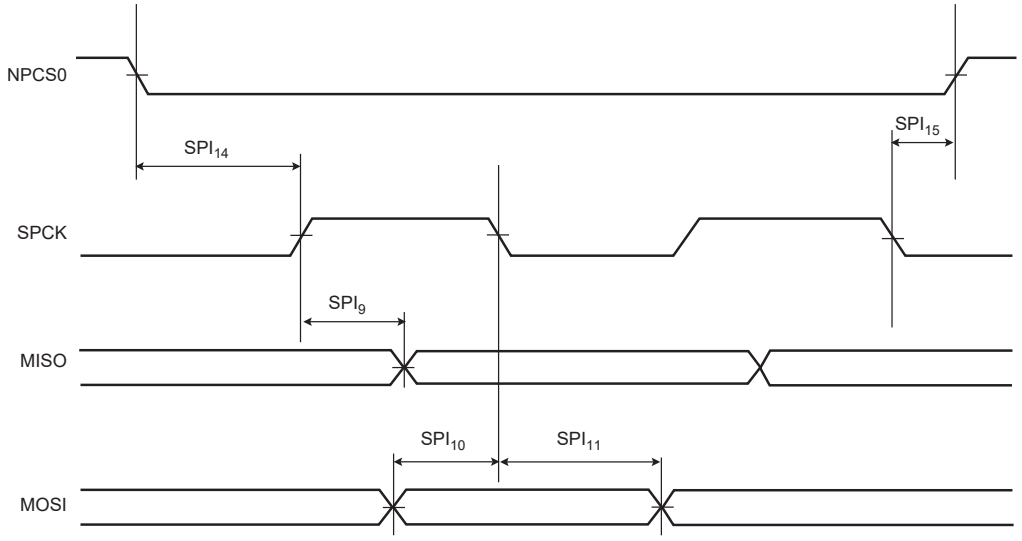


Table 59-7. VDDCORE Power-on Reset Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{T+}	Threshold Voltage Rising	–	0.79	0.95	1.07	V
V_{T-}	Threshold Voltage Falling	–	0.66	0.89	–	V
V_{hys}	Hysteresis Voltage	–	10	60	115	mV
t_{RES}	Reset Timeout Period	–	240	350	800	µs

Figure 59-23. SPI Slave Mode with (CPOL = NCPHA = 0) or (CPOL= NCPHA= 1)



59.13.1.6.1 Maximum SPI Frequency

The following formulas give maximum SPI frequency in Master read and write modes and in Slave read and write modes.

$$f_{SPCK}\max = \frac{1}{\mathrm{SPI}_0(\text{or } \mathrm{SPI}_3) + t_{\text{valid}}}$$

t_{valid} is the slave time response to output data after detecting an SPCK edge.

For a nonvolatile memory with t_{valid} (or t_v) = 5 ns, f_{SPCK}max = 57 MHz at V_{DDIO} = 3.3V.

$$f_{SPCK}\max = \frac{1}{2x(\mathit{SPI}_{6\max}(\text{or } \mathrm{SPI}_{9\max}) + t_{\text{setup}})}$$

t_{setup} is the setup time from the master before sampling data.

Master Write Mode

The SPI sends data to a slave device only, e.g. an LCD. The limit is given by SPI_2 (or SPI_5) timing. Since it gives a maximum frequency above the maximum pad speed (see I/O Characteristics), the max SPI frequency is the one from the pad.

Master Read Mode

Slave Read Mode

In slave mode, SPCK is the input clock for the SPI. The max SPCK frequency is given by setup and hold timings SPI_7/SPI_8(or SPI_{10}/SPI_{11}). Since this gives a frequency well above the pad limit, the limit in slave read mode is given by SPCK pad.

Slave Write Mode

59.13.1.6.2 SPI Timings

Timings are given in the following domains:

- 1.8V domain: V_{DDIO} from 1.7V to 1.95V, maximum external capacitor = 20 pF
- 3.3V domain: V_{DDIO} from 2.85V to 3.6V, maximum external capacitor = 40 pF

Symbol	VDDIO Supply	1.8V Domain	3.3V Domain	1.8V Domain	3.3V Domain	Unit
	Parameter	Min		Max		
NO HOLD Settings (NWE_HOLD = 0)						
SMC_{21}	NWE High to Data OUT, NBS0/A0 NBS1, A1, A2–A25, NCS change(1)	2.1	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”

Table 59-63. SMC Write NCS Controlled (WRITE_MODE = 0)

Symbol	VDDIO Supply	1.8V Domain	3.3V Domain	1.8V Domain	3.3V Domain	Unit
	Parameter	Min		Max		
SMC_{22}	Data Out Valid before NCS High	NCS_WR_PULSE × t_{CPMCK} - 2.8	NCS_WR_PULSE × t_{CPMCK} - 3.9	—	—	ns
SMC_{23}	NCS Pulse Width	NCS_WR_PULSE × t_{CPMCK} - 0.9	NCS_WR_PULSE × t_{CPMCK} - 0.2	—	—	ns
SMC_{24}	A0–A22 valid before NCS low	NCS_WR_SETUP × t_{CPMCK} - 4.0	NCS_WR_SETUP × t_{CPMCK} - 4.6	—	—	ns
SMC_{25}	NWE low before NCS high	(NCS_WR_SETUP - NWE_SETUP + NCS pulse) × t_{CPMCK} - 4.6	(NCS_WR_SETUP - NWE_SETUP + NCS pulse) × t_{CPMCK} - 4.6	—	—	ns
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.