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®-M4/M4F
Core Size	32-Bit Dual-Core
Speed	120MHz
Connectivity	EBI/EMI, I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, LCD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 6x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°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/atsam4cmp16cc-au

6.2 System I/O Lines

System I/O lines are pins used by oscillators, test mode, reset and JTAG and other features.

Table 6-1 describes the system I/O lines shared with PIO lines. These pins are software-configurable as general-purpose I/O or system pins. At start-up, the default function of these pins is always used.

Table 6-1. System I/O Configuration Pin List

SYSTEM_IO Bit Number	Default Function after Reset	Other Function	Constraints for Normal Start	Configuration
0	TDI	PB0	–	In Matrix User Interface Registers (Refer to Section 26.9.4 “System I/O Configuration Register”)
1	TDO/TRACESWO	PB1	–	
2	TMS/SWDIO	PB2	–	
3	TCK/SWCLK	PB3	–	
4	ERASE	PC9	Low level at Start-up(1)	
–	PA31	XIN	–	(2)
–	PA30	XOUT	–	

Notes: 1. If PC9 is used as PIO input in user applications, a low level must be ensured at start-up to prevent Flash erase before the user application sets PC9 into PIO mode.
2. Refer to Section 29.5.3 “3 to 20 MHz Crystal or Ceramic Resonator-based Oscillator”.

6.2.1 Serial Wire JTAG Debug Port (SWJ-DP) and Serial Wire Debug Port (SW-DP) Pins

The SWJ-DP pins are TCK/SWCLK, TMS/SWDIO, TDO/TRACESWO, TDI and commonly provided on a standard 20-pin JTAG connector defined by ARM. For more details about voltage reference and reset state, refer to Table 11-6 “Multiplexing on PIO Controller B (PIOB)”.

At start-up, SWJ-DP pins are configured in SWJ-DP mode to allow connection with debugging probe. Refer to Section 13. “Debug and Test Features”.

SWJ-DP pins can be used as standard I/Os to provide users with more general input/output pins when the debug port is not needed in the end application. Mode selection between SWJ-DP mode (System IO mode) and general IO mode is performed through the AHB Matrix Special Function Registers (MATRIX_SFR). Configuration of the pad for pull-up, triggers, debouncing and glitch filters is possible regardless of the mode.

The JTAGSEL pin is used to select the JTAG boundary scan when asserted at a high level. It integrates a permanent pull-down resistor of about 15 kΩ to GND, so that it can be left unconnected for normal operations.

By default, the JTAG Debug Port is active. If the debugger host wants to switch to the Serial Wire Debug Port, it must provide a dedicated JTAG sequence on TMS/SWDIO and TCK/SWCLK which disables the JTAG-DP and enables the SW-DP. When the Serial Wire Debug Port is active, TDO/TRACESWO can be used for trace.

The asynchronous TRACE output (TRACESWO) is multiplexed with TDO. So the asynchronous trace can only be used with SW-DP, not JTAG-DP. For more information about SW-DP and JTAG-DP switching, refer to Section 13. “Debug and Test Features”. The SW-DP/SWJ-DP pins are used for debug access to both cores.

6.3 TST Pin

The TST pin is used for JTAG Boundary Scan Manufacturing Test or Fast Flash programming mode of the SAM4CM series. For details on entering Fast Programming mode, refer to Section 23. “Fast Flash Programming Interface (FFPI)”. For more information on the manufacturing and test modes, refer to Section 13. “Debug and Test Features”.

11. Peripherals

11.1 Peripheral Identifiers

Table 11-1 defines the Peripheral Identifiers. A peripheral identifier is required for the control of the peripheral interrupt with the Nested Vectored Interrupt Controller, and for the control of the peripheral clock with the Power Management Controller.

The two ARM Cortex-M4 processors share the same interrupt mapping, and thus, they share all the interrupts of the peripherals.

Note: Some peripherals are on the Bus Matrix 0/AHB to APB Bridge 0 and other peripherals are on the Bus Matrix 1/ AHB to APB Bridge 1. If Core 0 needs to access a peripheral on the Bus Matrix 1/AHB to APB Bridge 1, the Core 0 application must enable the Core 1 System Clock (Bus and peripherals) and release Core 1 System Reset (Bus and peripherals). Peripherals on Sub-system 0 or Sub-system 1 are mentioned in the Instance description table that follows.

Table 11-1. Peripheral Identifiers

Instance ID	Instance Name	NVIC Interrupt	PMC Clock Control	Instance Description
0	SUPC	X	–	Supply Controller
1	RSTC	X	–	Reset Controller
2	RTC	X	–	Real-time Clock
3	RTT	X	–	Real-time Timer
4	WDT	X	–	Watchdog Timer
5	PMC	X	–	Power Management Controller
6	EFC0	X	–	Enhanced Embedded Flash Controller 0
7	EFC1	X	–	Enhanced Embedded Flash Controller 1
8	UART0	X	X	UART 0 (Sub-system 0 Clock)
9	–	–	–	Reserved
10	SMC0	–	X	Static Memory Controller 0 (Sub-system 0 Clock)
11	PIOA	X	X	Parallel I/O Controller A (Sub-system 0 Clock)
12	PIOB	X	X	Parallel I/O Controller B (Sub-system 0 Clock)
13	–	–	–	Reserved
14	USART0	X	X	USART 0 (Sub-system 0 Clock)
15	USART1	X	X	USART 1 (Sub-system 0 Clock)
16	USART2	X	X	USART 2 (Sub-system 0 Clock)
17	USART3	X	X	USART 3 (Sub-system 0 Clock)
18	–	–	–	Reserved
19	TWI0	X	X	Two Wire Interface 0 (Sub-system 0 Clock)
20	TWI1	X	X	Two Wire Interface 1 (Sub-system 0 Clock)
21	SPI0	X	X	Serial Peripheral Interface 0 (Sub-system 0 Clock)
22	–	–	–	Reserved
23	TC0	X	X	Timer/Counter 0 (Sub-system 0 Clock)
24	TC1	X	X	Timer/Counter 1 (Sub-system 0 Clock)
25	TC2	X	X	Timer/Counter 2 (Sub-system 0 Clock)
26	TC3	X	X	Timer/Counter 3 (Sub-system 0 Clock)

Restrictions

In these instructions:

- Do not use SP and do not use PC.
- *RdHi* and *RdLo* must be different registers.

Examples

```
SMULBT      R0, R4, R5  ; Multiplies the bottom halfword of R4 with the
                        ; top halfword of R5, multiplies results and
                        ; writes to R0
SMULBB      R0, R4, R5  ; Multiplies the bottom halfword of R4 with the
                        ; bottom halfword of R5, multiplies results and
                        ; writes to R0
SMULTT      R0, R4, R5  ; Multiplies the top halfword of R4 with the top
                        ; halfword of R5, multiplies results and writes
                        ; to R0
SMULTB      R0, R4, R5  ; Multiplies the top halfword of R4 with the
                        ; bottom halfword of R5, multiplies results and
                        ; and writes to R0
SMULWT      R4, R5, R3  ; Multiplies R5 with the top halfword of R3,
                        ; extracts top 32 bits and writes to R4
SMULWB      R4, R5, R3  ; Multiplies R5 with the bottom halfword of R3,
                        ; extracts top 32 bits and writes to R4.
```

12.8.3.3 Interrupt Set-pending Registers

Name: NVIC_ISPRx [x=0..7]

Access: Read/Write

Reset: 0x000000000

31	30	29	28	27	26	25	24
SETPEND							

23	22	21	20	19	18	17	16
SETPEND							

15	14	13	12	11	10	9	8
SETPEND							

7	6	5	4	3	2	1	0
SETPEND							

These registers force interrupts into the pending state, and show which interrupts are pending.

- **SETPEND: Interrupt Set-pending**

Write:

0: No effect.

1: Changes the interrupt state to pending.

Read:

0: Interrupt is not pending.

1: Interrupt is pending.

Notes:
1. Writing a 1 to an ISPR bit corresponding to an interrupt that is pending has no effect.
2. Writing a 1 to an ISPR bit corresponding to a disabled interrupt sets the state of that interrupt to pending.

12.9.1.5 Application Interrupt and Reset Control Register

Name: SCB_AIRCR

Access: Read/Write

31	30	29	28	27	26	25	24
VECTKEYSTAT/VECTKEY							

23	22	21	20	19	18	17	16
VECTKEYSTAT/VECTKEY							

15	14	13	12	11	10	9	8
ENDIANNESS	–	–	–	–	PRIGROUP		

7	6	5	4	3	2	1	0
–	–	–	–	–	SYSRESETREQ	VECTCLRACTIVE	VECTRESET

The SCB_AIRCR provides priority grouping control for the exception model, endian status for data accesses, and reset control of the system. To write to this register, write 0x5FA to the VECTKEY field, otherwise the processor ignores the write.

- **VECTKEYSTAT: Register Key (Read)**

Reads as 0xFA05.

- **VECTKEY: Register Key (Write)**

Writes 0x5FA to VECTKEY, otherwise the write is ignored.

- **ENDIANNESS: Data Endianness**

0: Little-endian.

1: Big-endian.

- **PRIGROUP: Interrupt Priority Grouping**

This field determines the split of group priority from subpriority. It shows the position of the binary point that splits the PRI_*n* fields in the Interrupt Priority Registers into separate *group priority* and *subpriority* fields. The table below shows how the PRIGROUP value controls this split.

	Interrupt Priority Level Value, PRI_*N*[7:0]			Number of	
PRIGROUP	**Binary Point**[1]	**Group Priority Bits**	**Subpriority Bits**	**Group Priorities**	**Subpriorities**
0b000	bxxxxxxx.y	[7:1]	None	128	2
0b001	bxxxxxx.yy	[7:2]	[4:0]	64	4
0b010	bxxxxx.yyy	[7:3]	[4:0]	32	8
0b011	bxxxx.yyyy	[7:4]	[4:0]	16	16
0b100	bxxx.yyyyy	[7:5]	[4:0]	8	32
0b101	bxx.yyyyyy	[7:6]	[5:0]	4	64
0b110	bx.yyyyyyy	[7]	[6:0]	2	128
0b111	b.yyyyyyy	None	[7:0]	1	256

Note: 1. PRI_n[7:0] field showing the binary point. x denotes a group priority field bit, and y denotes a subpriority field bit.

Determining preemption of an exception uses only the group priority field.

15. Reset Controller (RSTC)

15.1 Description

The Reset Controller (RSTC), driven by power-on reset (POR) cells, Software, external reset pin and peripheral events, handles all the resets of the system without any external components. It reports which reset occurred last.

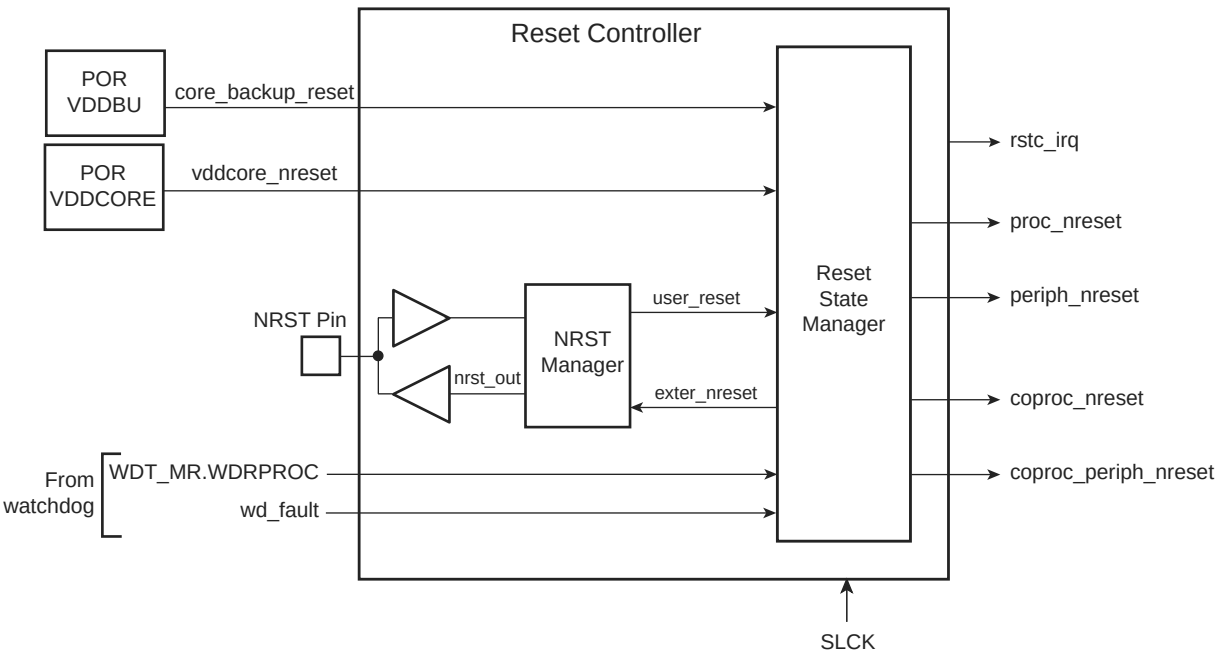
The RSTC also drives independently or simultaneously the external reset and the peripheral and processor resets.

15.2 Embedded Characteristics

- Driven by Embedded Power-on Reset, Software, External Reset Pin and Peripheral Events
- Management of All System Resets, Including
  - External Devices through the NRST Pin
  - Processor and Coprocessor (second processor)
  - Peripheral Set
- Reset Source Status
  - Status of the Last Reset
  - Either VDDCORE and VDDBU POR Reset, Software Reset, User Reset, Watchdog Reset
- External Reset Signal Control and Shaping

15.3 Block Diagram

Figure 15-1. Reset Controller Block Diagram



20.4.5 Using Backup Battery/Automatic Power Switch

The power switch automatically selects either VDDBU or VDDIO as a power source.

As soon as VDDIO is present (higher than 1.9V), it supplies the backup area of the device (VDDBU_SW = VDDIO) even if the voltage of VDDBU is higher than VDDIO. If not, the backup area is supplied by the VDDBU voltage source (VDDBU_SW = VDDBU). For more information on power supply schematics, refer to the section “Power Supplies”.

20.4.6 Supply Monitor

The SUPC embeds a supply monitor located in the VDDBU_SW power domain and which monitors VDDIO power supply.

The supply monitor can be used to prevent the processor from falling into an unpredictable state if the main power supply drops below a certain level.

The threshold of the supply monitor is programmable. It can be selected from 1.9V to 3.4V by steps of 100 mV. This threshold is configured in the SMTH field of the Supply Controller Supply Monitor Mode register (SUPC_SMMR).

The supply monitor can also be enabled during one slow clock period on every one of either 32, 256 or 2048 slow clock periods, depending on the user selection. This is configured in the SMSMPL field in SUPC_SMMR.

Enabling the supply monitor for such reduced times divides the typical supply monitor power consumption by factors of 2, 16 or 128, respectively, if continuous monitoring of the VDDIO power supply is not required.

A supply monitor detection can either generate a system reset (vddcore_nreset signal is asserted) or a system wakeup. Generating a system reset when a supply monitor detection occurs is enabled by setting the SMRSTEN bit in SUPC_SMMR.

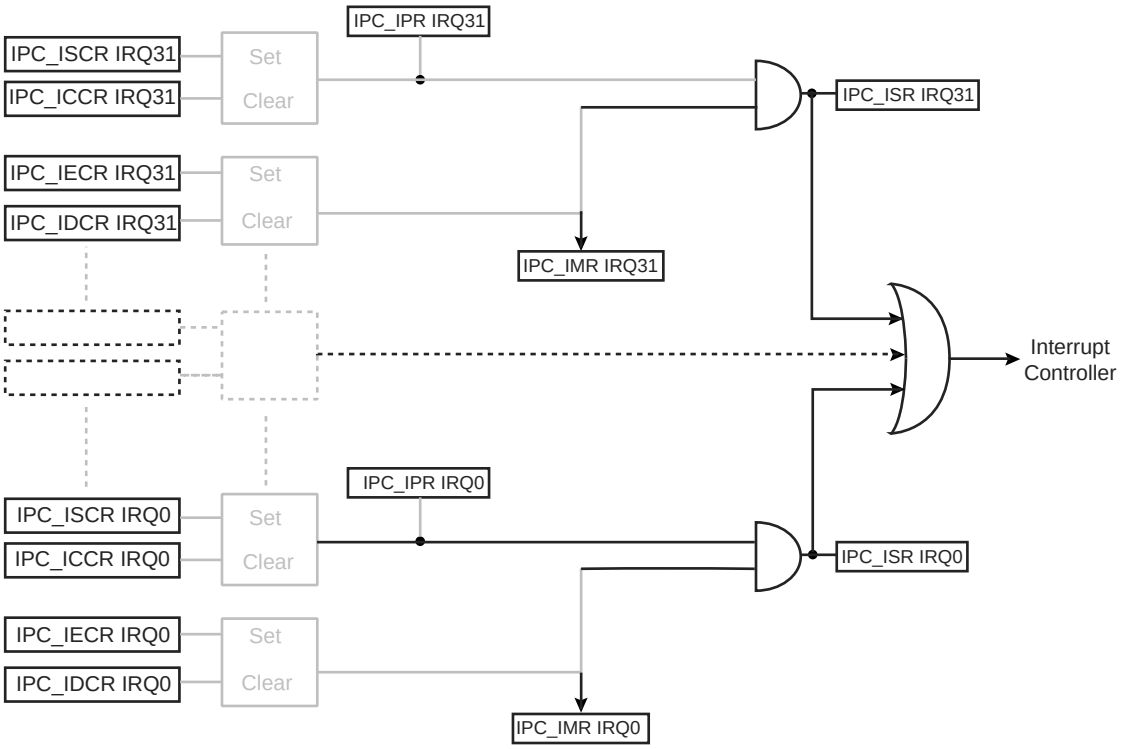
Waking up the system when a supply monitor detection occurs is enabled by setting the SMEN bit in the Supply Controller Wakeup Mode register (SUPC_WUMR).

The SUPC provides two status bits for the supply monitor in the SUPC_SR. These bits determine whether the last wakeup was due to the supply monitor:

- The SMOS bit provides real-time information, updated at each measurement cycle or updated at each Slow Clock cycle, if the measurement is continuous.
- The SMS bit provides saved information and shows a supply monitor detection has occurred since the last read of SUPC_SR.

The SMS bit generates an interrupt if the SMIEN bit is set in SUPC_SMMR.

Figure 25-3. Interrupt Input Stage



25.5.2 IPC Interrupt Clear Command Register

Name: IPC_ICCR

Address: 0x4004C004 (0), 0x48014004 (1)

Access: Write-only

31	30	29	28	27	26	25	24
IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24
23	22	21	20	19	18	17	16
IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
15	14	13	12	11	10	9	8
IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8
7	6	5	4	3	2	1	0
IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0

- **IRQ0-IRQ31: Interrupt Clear**

0: No effect.

1: Clears the corresponding interrupt.

26.9.6 Core Debug Configuration Register

Name: MATRIX_CORE_DEBUG

Address: 0x400E0328 (0), 0x48010128 (1)

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	CROSS_TRG0	CROSS_TRG1	–

- **CROSS_TRG1: Core 1 --> Core 0 Cross Triggering**

0: Core 1 is not able to trigger an event on core 0.

1: Core 1 is able to trigger an event on core 0.

- **CROSS_TRG0: Core 0 --> Core 1 Cross Triggering**

0: Core 0 is not able to trigger an event on core 1.

1: Core 0 is able to trigger an event on core 1.

28.5.9 Transfer Control Register

Name: PERIPH_PTCR

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	TXTDIS	TXTEN
7	6	5	4	3	2	1	0
–	–	–	–	–	–	RXTDIS	RXTEN

- **RXTEN: Receiver Transfer Enable**

0: No effect.

1: Enables PDC receiver channel requests if RXTDIS is not set.

When a half-duplex peripheral is connected to the PDC, enabling the receiver channel requests automatically disables the transmitter channel requests. It is forbidden to set both TXTEN and RXTEN for a half-duplex peripheral.

- **RXTDIS: Receiver Transfer Disable**

0: No effect.

1: Disables the PDC receiver channel requests.

When a half-duplex peripheral is connected to the PDC, disabling the receiver channel requests also disables the transmitter channel requests.

- **TXTEN: Transmitter Transfer Enable**

0: No effect.

1: Enables the PDC transmitter channel requests.

When a half-duplex peripheral is connected to the PDC, it enables the transmitter channel requests only if RXTEN is not set. It is forbidden to set both TXTEN and RXTEN for a half-duplex peripheral.

- **TXTDIS: Transmitter Transfer Disable**

0: No effect.

1: Disables the PDC transmitter channel requests.

When a half-duplex peripheral is connected to the PDC, disabling the transmitter channel requests disables the receiver channel requests.

32.6.34 PIO Output Write Disable Register

Name: PIO_OWDR

Address: 0x400E0EA4 (PIOA), 0x400E10A4 (PIOB), 0x4800C0A4 (PIOC)

Access: Write-only

31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

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

- **P0–P31: Output Write Disable**

0: No effect.

1: Disables writing PIO_ODSR for the I/O line.

33.7.3 Master Mode Operations

When configured in Master mode, the SPI operates on the clock generated by the internal programmable baud rate generator. It fully controls the data transfers to and from the slave(s) connected to the SPI bus. The SPI drives the chip select line to the slave and the serial clock signal (SPCK).

The SPI features two holding registers, the Transmit Data Register (SPI_TDR) and the Receive Data Register (SPI_RDR), and a single shift register. The holding registers maintain the data flow at a constant rate.

After enabling the SPI, a data transfer starts when the processor writes to the SPI_TDR. The written data is immediately transferred in the Shift register and the transfer on the SPI bus starts. While the data in the Shift register is shifted on the MOSI line, the MISO line is sampled and shifted in the Shift register. Data cannot be loaded in the SPI_RDR without transmitting data. If there is no data to transmit, dummy data can be used (SPI_TDR filled with ones). When the SPI_MR.WDRBT bit is set, new data cannot be transmitted if the SPI_RDR has not been read. If Receiving mode is not required, for example when communicating with a slave receiver only (such as an LCD), the receive status flags in the SPI Status register (SPI_SR) can be discarded.

Before writing the SPI_TDR, the PCS field in the SPI_MR must be set in order to select a slave.

If new data is written in the SPI_TDR during the transfer, it is kept in the SPI_TDR until the current transfer is completed. Then, the received data is transferred from the Shift register to the SPI_RDR, the data in the SPI_TDR is loaded in the Shift register and a new transfer starts.

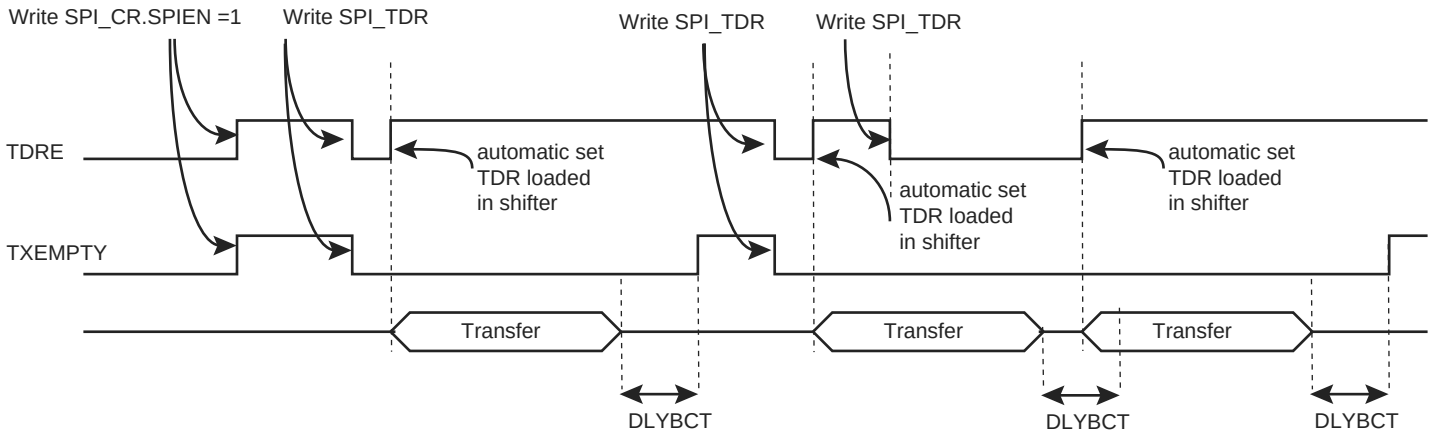
As soon as the SPI_TDR is written, the Transmit Data Register Empty (TDRE) flag in the SPI_SR is cleared. When the data written in the SPI_TDR is loaded into the Shift register, the TDRE flag in the SPI_SR is set. The TDRE bit is used to trigger the Transmit PDC channel.

See Figure 33-5.

The end of transfer is indicated by the TXEMPTY flag in the SPI_SR. If a transfer delay (DLYBCT) is greater than 0 for the last transfer, TXEMPTY is set after the completion of this delay. The peripheral clock can be switched off at this time.

Note: When the SPI is enabled, the TDRE and TXEMPTY flags are set.

Figure 33-5. TDRE and TXEMPTY flag behavior



The transfer of received data from the Shift register to the SPI_RDR is indicated by the Receive Data Register Full (RDRF) bit in the SPI_SR. When the received data is read, the RDRF bit is cleared.

If the SPI_RDR has not been read before new data is received, the Overrun Error (OVRES) bit in the SPI_SR is set. As long as this flag is set, data is loaded in the SPI_RDR. The user has to read the SPI_SR to clear the OVRES bit.

Figure 33-6 shows a block diagram of the SPI when operating in Master mode. Figure 33-7 shows a flow chart describing how transfers are handled.

33.8.2 SPI Mode Register

Name: SPI_MR

Address: 0x40008004 (0), 0x48000004 (1)

Access: Read/Write

31	30	29	28	27	26	25	24
DLYBCS							
23	22	21	20	19	18	17	16
–	–	–	–	PCS			
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
LLB	–	WDRBT	MODFDIS	–	PCSDEC	PS	MSTR

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

- **MSTR: Master/Slave Mode**

0: SPI is in Slave mode

1: SPI is in Master mode

- **PS: Peripheral Select**

0: Fixed Peripheral Select

1: Variable Peripheral Select

- **PCSDEC: Chip Select Decode**

0: The chip selects are directly connected to a peripheral device.

1: The four NPCS chip select lines are connected to a 4-bit to 16-bit decoder.

When PCSDEC = 1, up to 15 Chip Select signals can be generated with the four NPCS lines using an external 4-bit to 16-bit decoder. The Chip Select registers define the characteristics of the 15 chip selects, with the following rules:

SPI_CSR0 defines peripheral chip select signals 0 to 3.

SPI_CSR1 defines peripheral chip select signals 4 to 7.

SPI_CSR2 defines peripheral chip select signals 8 to 11.

SPI_CSR3 defines peripheral chip select signals 12 to 14.

- **MODFDIS: Mode Fault Detection**

0: Mode fault detection enabled

1: Mode fault detection disabled

- **WDRBT: Wait Data Read Before Transfer**

0: No Effect. In Master mode, a transfer can be initiated regardless of the SPI_RDR state.

1: In Master mode, a transfer can start only if the SPI_RDR is empty, i.e., does not contain any unread data. This mode prevents overrun error in reception.

34.8.3 TWI Slave Mode Register

Name: TWI_SMR

Address: 0x40018008 (0), 0x4001C008 (1)

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	SADR						

15	14	13	12	11	10	9	8
–	–	–	–	–	–		

7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	–

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

- **SADR: Slave Address**

The slave device address is used in Slave mode in order to be accessed by master devices in Read or Write mode.

SADR must be programmed before enabling the Slave mode or after a general call. Writes at other times have no effect.

When OPT_EN = 1, the URXD pad is automatically configured in Analog mode and the analog comparator is enabled (see Figure 35-11).

To match the characteristics of the off-chip optical receiver circuitry, the voltage reference threshold of the embedded comparator can be adjusted from VDDIO/10 up to VDD/2 by programming the OPT_CMPTH field in UART_MR.

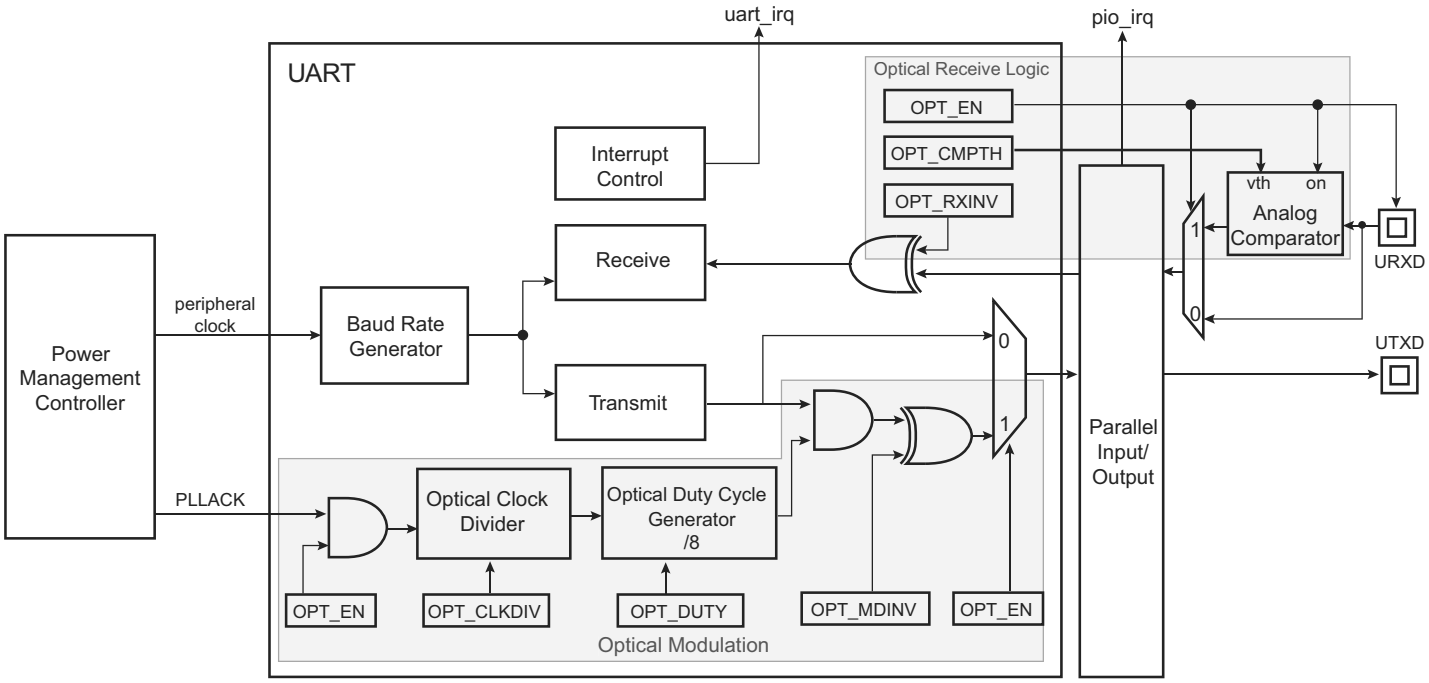
The NRZ output of the UART transmitter sub-module is modulated with the 30 up to 60 kHz modulation clock prior to driving the PIO controller.

A logical 0 on the UART transmitter sub-module output generates the said modulated signal (see Figure 35-12) having a frequency programmable from 30 kHz up to 60 kHz (38 kHz is the default value assuming the PLLA clock frequency is 8192 kHz). A logical 1 on the UART transmitter sub-module output generates a stuck-at 1 output signal (no modulation). The idle polarity of the modulated signal is 1 (OPT_MDINV = 0 in UART_MR).

The idle polarity of the modulated signal can be inverted by writing a 1 to the OPT_MDINV bit in UART_MR.

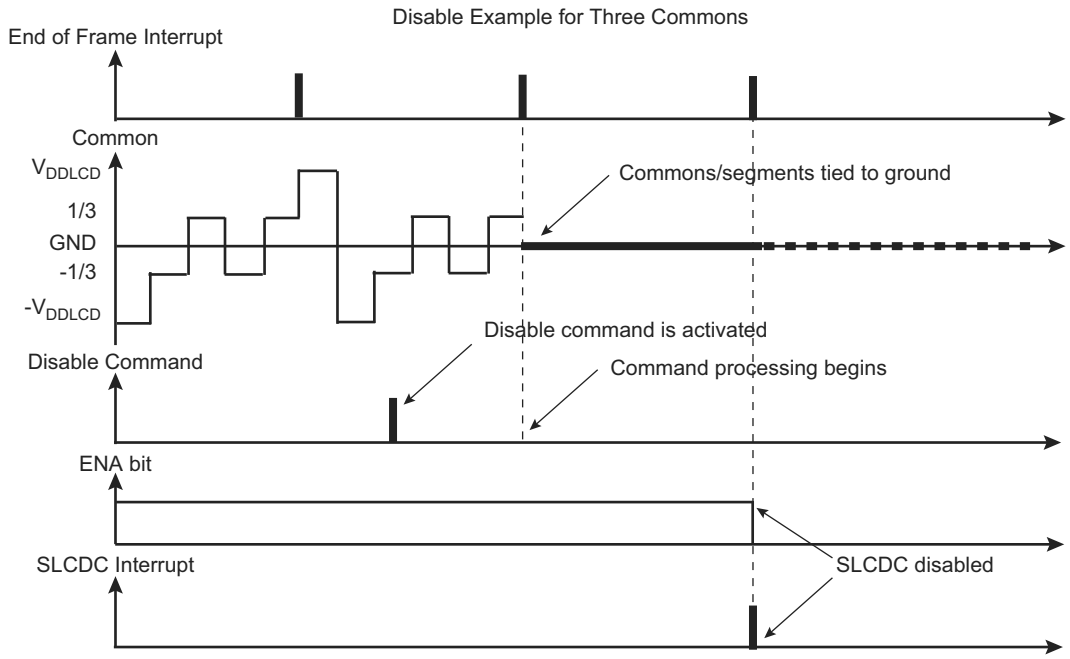
The duty cycle of the modulated signal can be adjusted from 6.25% up to 50% (default value) by steps of 6.25% by programming the OPT_DUTY field in UART_MR.

Figure 35-11. Optical Interface Block Diagram



- **CTSIC: Clear to Send Input Change Interrupt Disable**
- **MANE: Manchester Error Interrupt Disable**

Figure 39-10. Disabling Sequence

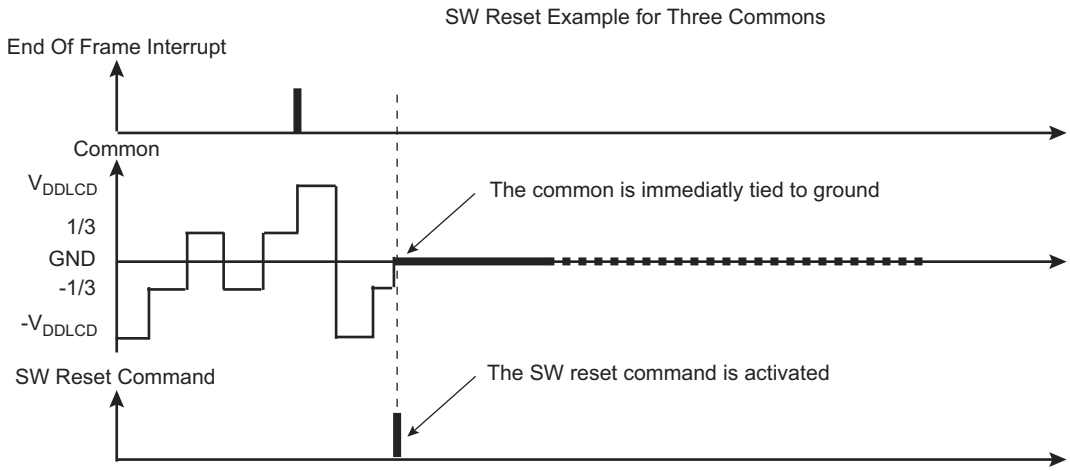


39.6.7.2 Software Reset

When the SLCDC software reset command is activated during a frame, it is immediately processed and all commons and segments are tied to ground.

Note that in the case of a software reset, the disable interrupt is not asserted.

Figure 39-11. Software Reset



The selected oversampling ratio applies to all enabled channels except for the temperature sensor channel when triggered by an RTC event.

The average result is valid into the ADC_CDRx register (x corresponding to the index of the channel) only if EOCn flag is set in ADC_ISR and OVREn flag is cleared in ADC_OVER. The average result for all channels is valid in ADC_LCDR only if DRDY is set and GOVRE is cleared in ADC_ISR.

Note that registers ADC_CDRx are not buffered. Therefore, when an averaging sequence is ongoing, the value in these registers changes after each averaging sample. However, overrun flags in ADC_OVER rise as soon as the first sample of an averaging sequence is received. Thus the previous averaged value is not read even if the new averaged value is not ready.

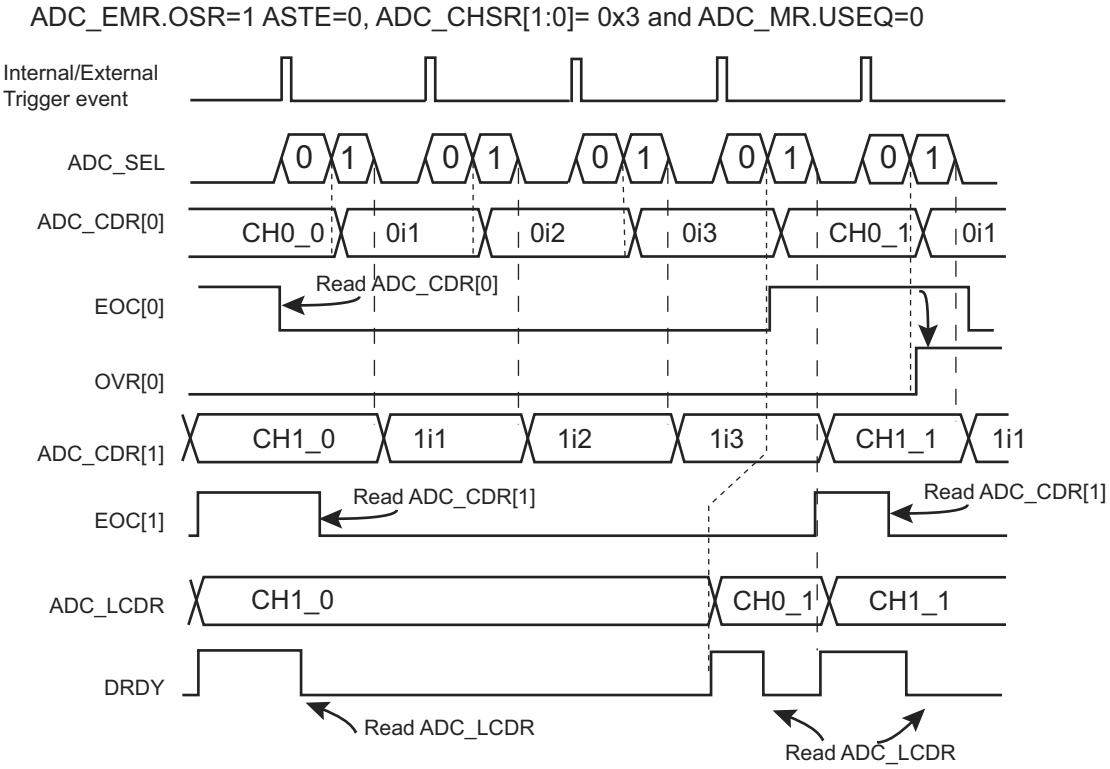
As a result, when an overrun flag rises in ADC_OVER, the previous unread data is lost. However, the data has not been overwritten by the new averaged value, as the averaging sequence for this channel may still be on-going.

40.6.11.1 Averaging Function versus Trigger Events

The samples can be defined in different ways for the averaging function depending on the configuration of the ASTE bit in ADC_EMR and the USEQ bit in ADC_MR.

When USEQ is cleared, there are two ways to generate the averaging through the trigger event. If ASTE is cleared in ADC_EMR, every trigger event generates one sample for each enabled channel as described in Figure 40-9, "Digital Averaging Function Waveforms over Multiple Trigger Events". Therefore, four trigger events are requested to get the result of averaging if OSR = 1.

Figure 40-9. Digital Averaging Function Waveforms over Multiple Trigger Events



Notes: ADC_SEL: Command to the ADC cell
0i1,0i2,0i3, 1i1, 1i2, 1i3 are intermediate results and CH0/1_0/1 are final results of average function.

If ASTE = 1 in ADC_EMR and USEQ = 0 in ADC_MR, then the sequence to be converted, defined in ADC_CHSR, is automatically repeated n times, where n corresponds to the oversampling ratio defined in the OSR field in ADC_EMR. As a result, only one trigger is required to obtain the result of the averaging function as described in Figure 40-9, "Digital Averaging Function Waveforms over Multiple Trigger Events".

45.6.6 TRNG Output Data Register

Name: TRNG_ODATA

Address: 0x40048050

Access: Read-only

31	30	29	28	27	26	25	24
ODATA							
23	22	21	20	19	18	17	16
ODATA							
15	14	13	12	11	10	9	8
ODATA							
7	6	5	4	3	2	1	0
ODATA							

- **ODATA: Output Data**

The 32-bit Output Data register contains the 32-bit random data.