



Welcome to [E-XFL.COM](http://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
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	I <sup>2</sup> C, IrDA, Memory Card, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	47
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 11x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/atsam4s4bb-an">https://www.e-xfl.com/product-detail/microchip-technology/atsam4s4bb-an</a>

## 12.6.6 Multiply and Divide Instructions

The table below shows the multiply and divide instructions.

**Table 12-21. Multiply and Divide Instructions**

Mnemonic	Description
MLA	Multiply with Accumulate, 32-bit result
MLS	Multiply and Subtract, 32-bit result
MUL	Multiply, 32-bit result
SDIV	Signed Divide
SMLA[B,T]	Signed Multiply Accumulate (halfwords)
SMLAD, SMLADX	Signed Multiply Accumulate Dual
SMLAL	Signed Multiply with Accumulate ( $32 \times 32 + 64$ ), 64-bit result
SMLAL[B,T]	Signed Multiply Accumulate Long (halfwords)
SMLALD, SMLALDX	Signed Multiply Accumulate Long Dual
SMLAW[B,T]	Signed Multiply Accumulate (word by halfword)
SMLSD	Signed Multiply Subtract Dual
SMLSXD	Signed Multiply Subtract Long Dual
SMMLA	Signed Most Significant Word Multiply Accumulate
SMMLS, SMMLSR	Signed Most Significant Word Multiply Subtract
SMUAD, SMUADX	Signed Dual Multiply Add
SMUL[B,T]	Signed Multiply (word by halfword)
SMMUL, SMMULR	Signed Most Significant Word Multiply
SMULL	Signed Multiply ( $32 \times 32$ ), 64-bit result
SMULWB, SMULWT	Signed Multiply (word by halfword)
SMUSD, SMUSDX	Signed Dual Multiply Subtract
UDIV	Unsigned Divide
UMAAL	Unsigned Multiply Accumulate Accumulate Long ( $32 \times 32 + 32 + 32$ ), 64-bit result
UMLAL	Unsigned Multiply with Accumulate ( $32 \times 32 + 64$ ), 64-bit result
UMULL	Unsigned Multiply ( $32 \times 32$ ), 64-bit result

#### 12.6.7.6 UQASX and UQSAX

Saturating Add and Subtract with Exchange and Saturating Subtract and Add with Exchange, unsigned.

Syntax

$op\{cond\} \{Rd\}, Rm, Rn$

where:

type is one of:

UQASX Add and Subtract with Exchange and Saturate.

UQSAX Subtract and Add with Exchange and Saturate.

cond is an optional condition code, see “Conditional Execution” .

Rd is the destination register.

Rn, Rm are registers holding the first and second operands.

Operation

The UQASX instruction:

1. Adds the bottom halfword of the source operand with the top halfword of the second operand.
2. Subtracts the bottom halfword of the second operand from the top highword of the first operand.
3. Saturates the results of the sum and writes a 16-bit unsigned integer in the range  $0 \leq x \leq 2^{16} - 1$ , where  $x$  equals 16, to the top halfword of the destination register.
4. Saturates the result of the subtraction and writes a 16-bit unsigned integer in the range  $0 \leq x \leq 2^{16} - 1$ , where  $x$  equals 16, to the bottom halfword of the destination register.

The UQSAX instruction:

1. Subtracts the bottom halfword of the second operand from the top highword of the first operand.
2. Adds the bottom halfword of the first operand with the top halfword of the second operand.
3. Saturates the result of the subtraction and writes a 16-bit unsigned integer in the range  $0 \leq x \leq 2^{16} - 1$ , where  $x$  equals 16, to the top halfword of the destination register.
4. Saturates the results of the addition and writes a 16-bit unsigned integer in the range  $0 \leq x \leq 2^{16} - 1$ , where  $x$  equals 16, to the bottom halfword of the destination register.

Restrictions

Do not use SP and do not use PC.

Condition Flags

These instructions do not affect the condition code flags.

## Restrictions

Do not use SP and do not use PC.

## Condition Flags

These instructions do not affect the condition code flags.

## Examples

```
UQADD16  R7, R4, R2    ; Adds halfwords in R4 to corresponding halfword in R2,  
                        ; saturates to 16 bits, writes to corresponding halfword of R7  
UQADD8   R4, R2, R5    ; Adds bytes of R2 to corresponding byte of R5, saturates  
                        ; to 8 bits, writes to corresponding bytes of R4  
UQSUB16  R6, R3, R0    ; Subtracts halfwords in R0 from corresponding halfword  
                        ; in R3, saturates to 16 bits, writes to corresponding  
                        ; halfword in R6  
UQSUB8   R1, R5, R6    ; Subtracts bytes in R6 from corresponding byte of R5,  
                        ; saturates to 8 bits, writes to corresponding byte of R1.
```

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.

**Table 13-2. SWJ-DP Pin List**

Pin Name	JTAG Port	Serial Wire Debug Port
TMS/SWDIO	TMS	SWDIO
TCK/SWCLK	TCK	SWCLK
TDI	TDI	—
TDO/TRACESWO	TDO	TRACESWO (optional: trace)

SW-DP or JTAG-DP mode is selected when JTAGSEL is low. It is not possible to switch directly between SWJ-DP and JTAG boundary scan operations. A chip reset must be performed after JTAGSEL is changed.

### 13.5.3.1 SW-DP and JTAG-DP Selection Mechanism

Debug port selection mechanism is done by sending specific **SWDIOTMS** sequence. The JTAG-DP is selected by default after reset.

- Switch from JTAG-DP to SW-DP. The sequence is:
  - Send more than 50 **SWCLKTCK** cycles with **SWDIOTMS** = 1
  - Send the 16-bit sequence on **SWDIOTMS** = 0111100111100111 (0x79E7 MSB first)
  - Send more than 50 **SWCLKTCK** cycles with **SWDIOTMS** = 1
- Switch from SWD to JTAG. The sequence is:
  - Send more than 50 **SWCLKTCK** cycles with **SWDIOTMS** = 1
  - Send the 16-bit sequence on **SWDIOTMS** = 0011110011100111 (0x3CE7 MSB first)
  - Send more than 50 **SWCLKTCK** cycles with **SWDIOTMS** = 1

### 13.5.4 FPB (Flash Patch Breakpoint)

The FPB:

- Implements hardware breakpoints
- Patches code and data from code space to system space.

The FPB unit contains:

- Two literal comparators for matching against literal loads from Code space, and remapping to a corresponding area in System space.
- Six instruction comparators for matching against instruction fetches from Code space and remapping to a corresponding area in System space.
- Alternatively, comparators can also be configured to generate a Breakpoint instruction to the processor core on a match.

### 13.5.5 DWT (Data Watchpoint and Trace)

The DWT contains four comparators which can be configured to generate the following:

- PC sampling packets at set intervals
- PC or Data watchpoint packets
- Watchpoint event to halt core

### 16.6.3 RTC Time Register

**Name:** RTC\_TIMR

**Address:** 0x400E1468

**Access:** Read/Write

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

- **SEC: Current Second**

The range that can be set is 0–59 (BCD).

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

- **MIN: Current Minute**

The range that can be set is 0–59 (BCD).

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

- **HOUR: Current Hour**

The range that can be set is 1–12 (BCD) in 12-hour mode or 0–23 (BCD) in 24-hour mode.

- **AMPM: Ante Meridiem Post Meridiem Indicator**

This bit is the AM/PM indicator in 12-hour mode.

0: AM.

1: PM.

All non-significant bits read zero.

#### 18.4.8 Register Write Protection

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

The following registers can be write-protected:

- RSTC Mode Register
- RTT Mode Register
- RTT Alarm Register
- RTC Control Register
- RTC Mode Register
- RTC Time Alarm Register
- RTC Calendar Alarm Register
- General Purpose Backup Registers
- Supply Controller Control Register
- Supply Controller Supply Monitor Mode Register
- Supply Controller Mode Register
- Supply Controller Wake-up Mode Register
- Supply Controller Wake-up Inputs Register

#### 18.4.9 Register Bits in Backup Domain (VDDIO)

The following configuration registers, or certain bits of the registers, are physically located in the product backup domain:

- RSTC Mode Register (all bits)
- RTT Mode Register (all bits)
- RTT Alarm Register (all bits)
- RTC Control Register (all bits)
- RTC Mode Register (all bits)
- RTC Time Alarm Register (all bits)
- RTC Calendar Alarm Register (all bits)
- General Purpose Backup Registers (all bits)
- Supply Controller Control Register (see register description for details)
- Supply Controller Supply Monitor Mode Register (all bits)
- Supply Controller Mode Register (see register description for details)
- Supply Controller Wake-up Mode Register (all bits)
- Supply Controller Wake-up Inputs Register (all bits)
- Supply Controller Status Register (all bits)

The erase sequence is the following:

1. Erase starts as soon as one of the erase commands and the FARG field are written in EEFC\_FCR.
  - For the EPA command, the two lowest bits of the FARG field define the number of pages to be erased (FARG[1:0]):

**Table 20-4. EEFC\_FCR.FARG Field for EPA Command**

FARG[1:0]	Number of pages to be erased with EPA command
0	4 pages (only valid for small 8 KB sectors)
1	8 pages
2	16 pages
3	32 pages (not valid for small 8 KB sectors)

2. When erasing is completed, the bit EEFC\_FSR.FRDY rises. If an interrupt has been enabled by setting the bit EEFC\_FMR.FRDY, the interrupt line of the interrupt controller is activated.

Three errors can be detected in EEFC\_FSR after an erasing sequence:

- Command Error: A bad keyword has been written in EEFC\_FCR.
- Lock Error: At least one page to be erased belongs to a locked region. The erase command has been refused, no page has been erased. A command must be run previously to unlock the corresponding region.
- Flash Error: At the end of the erase period, the EraseVerify test of the Flash memory has failed.

#### 20.4.3.4 Lock Bit Protection

Lock bits are associated with several pages in the embedded Flash memory plane. This defines lock regions in the embedded Flash memory plane. They prevent writing/erasing protected pages.

The lock sequence is the following:

1. Execute the 'Set Lock Bit' command by writing EEFC\_FCR.FCMD with the SLB command and EEFC\_FCR.FARG with a page number to be protected.
2. When the locking completes, the bit EEFC\_FSR.FRDY rises. If an interrupt has been enabled by setting the bit EEFC\_FMR.FRDY, the interrupt line of the interrupt controller is activated.
3. The result of the SLB command can be checked running a 'Get Lock Bit' (GLB) command.

Note: The value of the FARG argument passed together with SLB command must not exceed the higher lock bit index available in the product.

Two errors can be detected in EEFC\_FSR after a programming sequence:

- Command Error: A bad keyword has been written in EEFC\_FCR.
- Flash Error: At the end of the programming, the EraseVerify or WriteVerify test of the Flash memory has failed.

It is possible to clear lock bits previously set. After the lock bits are cleared, the locked region can be erased or programmed. The unlock sequence is the following:

1. Execute the 'Clear Lock Bit' command by writing EEFC\_FCR.FCMD with the CLB command and EEFC\_FCR.FARG with a page number to be unprotected.
2. When the unlock completes, the bit EEFC\_FSR.FRDY rises. If an interrupt has been enabled by setting the bit EEFC\_FMR.FRDY, the interrupt line of the interrupt controller is activated.

Note: The value of the FARG argument passed together with CLB command must not exceed the higher lock bit index available in the product.

Two errors can be detected in EEFC\_FSR after a programming sequence:

- Command Error: A bad keyword has been written in EEFC\_FCR.



### 21.3.5.5 Flash General-purpose NVM Commands

General-purpose NVM bits (GP NVM bits) can be set using the **Set GPNVM** command (**SGPB**). This command also activates GP NVM bits. A bit mask is provided as argument to the command. When bit 0 of the bit mask is set, then the first GP NVM bit is activated.

In the same way, the **Clear GPNVM** command (**CGPB**) is used to clear general-purpose NVM bits. The general-purpose NVM bit is deactivated when the corresponding bit in the pattern value is set to 1.

**Table 21-11. Set/Clear GP NVM Command**

Step	Handshake Sequence	MODE[3:0]	DATA[15:0]
1	Write handshaking	CMDE	SGPB or CGPB
2	Write handshaking	DATA	GP NVM bit pattern value

General-purpose NVM bits can be read using the **Get GPNVM Bit** command (**GGPB**). The  $n^{\text{th}}$  GP NVM bit is active when bit  $n$  of the bit mask is set.

**Table 21-12. Get GP NVM Bit Command**

Step	Handshake Sequence	MODE[3:0]	DATA[15:0]
1	Write handshaking	CMDE	GGPB
2	Read handshaking	DATA	GP NVM Bit Mask Status 0 = GP NVM bit is cleared 1 = GP NVM bit is set

### 21.3.5.6 Flash Security Bit Command

A security bit can be set using the **Set Security Bit** command (SSE). Once the security bit is active, the Fast Flash programming is disabled. No other command can be run. An event on the Erase pin can erase the security bit once the contents of the Flash have been erased.

**Table 21-13. Set Security Bit Command**

Step	Handshake Sequence	MODE[3:0]	DATA[15:0]
1	Write handshaking	CMDE	SSE
2	Write handshaking	DATA	0

Once the security bit is set, it is not possible to access FFPI. The only way to erase the security bit is to erase the Flash.

In order to erase the Flash, the user must perform the following:

1. Power-off the chip.
2. Power-on the chip with TST = 0.
3. Assert Erase during a period of more than 220 ms.
4. Power-off the chip.

Then it is possible to return to FFPI mode and check that Flash is erased.

### 21.3.5.7 Memory Write Command

This command is used to perform a write access to any memory location.

### 29.17.15PMC Interrupt Disable Register

**Name:** PMC\_IDR

**Address:** 0x400E0464

**Access:** Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	CFDEV	MOSCRCS	MOSCSELS
15	14	13	12	11	10	9	8
–	–	–	–	–	PCKRDY2	PCKRDY1	PCKRDY0
7	6	5	4	3	2	1	0
–	–	–	–	MCKRDY	LOCKB	LOCKA	MOSCXTS

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

0: No effect.

1: Disables the corresponding interrupt.

- **MOSCXTS: Main Crystal Oscillator Status Interrupt Disable**
- **LOCKA: PLLA Lock Interrupt Disable**
- **LOCKB: PLLB Lock Interrupt Disable**
- **MCKRDY: Master Clock Ready Interrupt Disable**
- **PCKRDYx: Programmable Clock Ready x Interrupt Disable**
- **MOSCSELS: Main Oscillator Selection Status Interrupt Disable**
- **MOSCRCS: Main On-Chip RC Status Interrupt Disable**
- **CFDEV: Clock Failure Detector Event Interrupt Disable**

- **START: Transmit Start Selection**

Value	Name	Description
0	CONTINUOUS	Continuous, as soon as a word is written in the SSC_THR (if Transmit is enabled), and immediately after the end of transfer of the previous data
1	RECEIVE	Receive start
2	TF_LOW	Detection of a low level on TF signal
3	TF_HIGH	Detection of a high level on TF signal
4	TF_FALLING	Detection of a falling edge on TF signal
5	TF_RISING	Detection of a rising edge on TF signal
6	TF_LEVEL	Detection of any level change on TF signal
7	TF_EDGE	Detection of any edge on TF signal

- **STTDLY: Transmit Start Delay**

If STTDLY is not 0, a delay of STTDLY clock cycles is inserted between the start event and the actual start of transmission of data. When the Transmitter is programmed to start synchronously with the Receiver, the delay is also applied.

Note: Note: STTDLY must be set carefully. If STTDLY is too short in respect to TAG (Transmit Sync Data) emission, data is emitted instead of the end of TAG.

- **PERIOD: Transmit Period Divider Selection**

This field selects the divider to apply to the selected Transmit Clock to generate a new Frame Sync Signal. If 0, no period signal is generated. If not 0, a period signal is generated at each  $2 \times (\text{PERIOD} + 1)$  Transmit Clock.

- **RXBUFF: Receive Buffer Full Interrupt Enable**

0: No effect.

1: Enables the Receive Buffer Full Interrupt.

- **CP0: Compare 0 Interrupt Enable**

0: No effect.

1: Enables the Compare 0 Interrupt.

- **CP1: Compare 1 Interrupt Enable**

0: No effect.

1: Enables the Compare 1 Interrupt.

- **TXSYN: Tx Sync Interrupt Enable**

0: No effect.

1: Enables the Tx Sync Interrupt.

- **RXSYN: Rx Sync Interrupt Enable**

0: No effect.

1: Enables the Rx Sync Interrupt.

When the start frame delimiter is a sync pattern (ONEBIT field to 0), both command and data delimiter are supported. If a valid sync is detected, the received character is written as RXCHR field in the US\_RHR and the RXSYNH is updated. RXCHR is set to 1 when the received character is a command, and it is set to 0 if the received character is a data. This mechanism alleviates and simplifies the direct memory access as the character contains its own sync field in the same register.

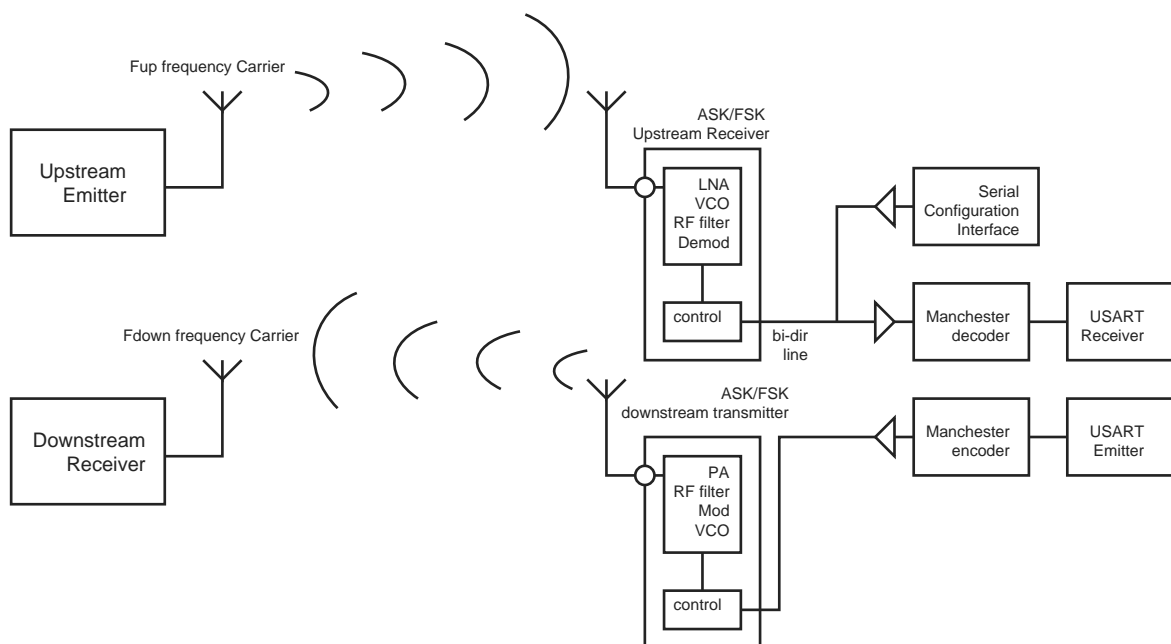
As the decoder is setup to be used in Unipolar mode, the first bit of the frame has to be a zero-to-one transition.

### 36.6.3.5 Radio Interface: Manchester Encoded USART Application

This section describes low data rate RF transmission systems and their integration with a Manchester encoded USART. These systems are based on transmitter and receiver ICs that support ASK and FSK modulation schemes.

The goal is to perform full duplex radio transmission of characters using two different frequency carriers. See the configuration in Figure 36-16.

**Figure 36-16. Manchester Encoded Characters RF Transmission**



The USART peripheral is configured as a Manchester encoder/decoder. Looking at the downstream communication channel, Manchester encoded characters are serially sent to the RF emitter. This may also include a user defined preamble and a start frame delimiter. Mostly, preamble is used in the RF receiver to distinguish between a valid data from a transmitter and signals due to noise. The Manchester stream is then modulated. See Figure 36-17 for an example of ASK modulation scheme. When a logic one is sent to the ASK modulator, the power amplifier, referred to as PA, is enabled and transmits an RF signal at downstream frequency. When a logic zero is transmitted, the RF signal is turned off. If the FSK modulator is activated, two different frequencies are used to transmit data. When a logic 1 is sent, the modulator outputs an RF signal at frequency F0 and switches to F1 if the data sent is a 0. See Figure 36-18.

From the receiver side, another carrier frequency is used. The RF receiver performs a bit check operation examining demodulated data stream. If a valid pattern is detected, the receiver switches to Receiving mode. The demodulated stream is sent to the Manchester decoder. Because of bit checking inside RF IC, the data transferred to the microcontroller is reduced by a user-defined number of bits. The Manchester preamble length is to be defined in accordance with the RF IC configuration.

### 36.7.7 USART Interrupt Disable Register

**Name:** US\_IDR

**Address:** 0x4002400C (0), 0x4002800C (1)

**Access:** Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	MANE
23	22	21	20	19	18	17	16
–	–	–	–	CTSIC	DCDIC	DSRIC	RIIC
15	14	13	12	11	10	9	8
–	–	NACK	RXBUFF	TXBUFE	ITER	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	ENDTX	ENDRX	RXBRK	TXRDY	RXRDY

For SPI specific configuration, see Section 36.7.8 "USART Interrupt Disable Register (SPI\_MODE)".

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

0: No effect

1: Disables the corresponding interrupt.

- **RXRDY: RXRDY Interrupt Disable**
- **TXRDY: TXRDY Interrupt Disable**
- **RXBRK: Receiver Break Interrupt Disable**
- **ENDRX: End of Receive Buffer Transfer Interrupt Disable (available in all USART modes of operation)**
- **ENDTX: End of Transmit Buffer Interrupt Disable (available in all USART modes of operation)**
- **OVRE: Overrun Error Interrupt Enable**
- **FRAME: Framing Error Interrupt Disable**
- **PARE: Parity Error Interrupt Disable**
- **TIMEOUT: Time-out Interrupt Disable**
- **TXEMPTY: TXEMPTY Interrupt Disable**
- **ITER: Max Number of Repetitions Reached Interrupt Disable**
- **TXBUFE: Transmit Buffer Empty Interrupt Disable (available in all USART modes of operation)**
- **RXBUFF: Receive Buffer Full Interrupt Disable (available in all USART modes of operation)**
- **NACK: Non Acknowledge Interrupt Disable**
- **RIIC: Ring Indicator Input Change Disable**

### 37.6.14.3 Direction Status and Change Detection

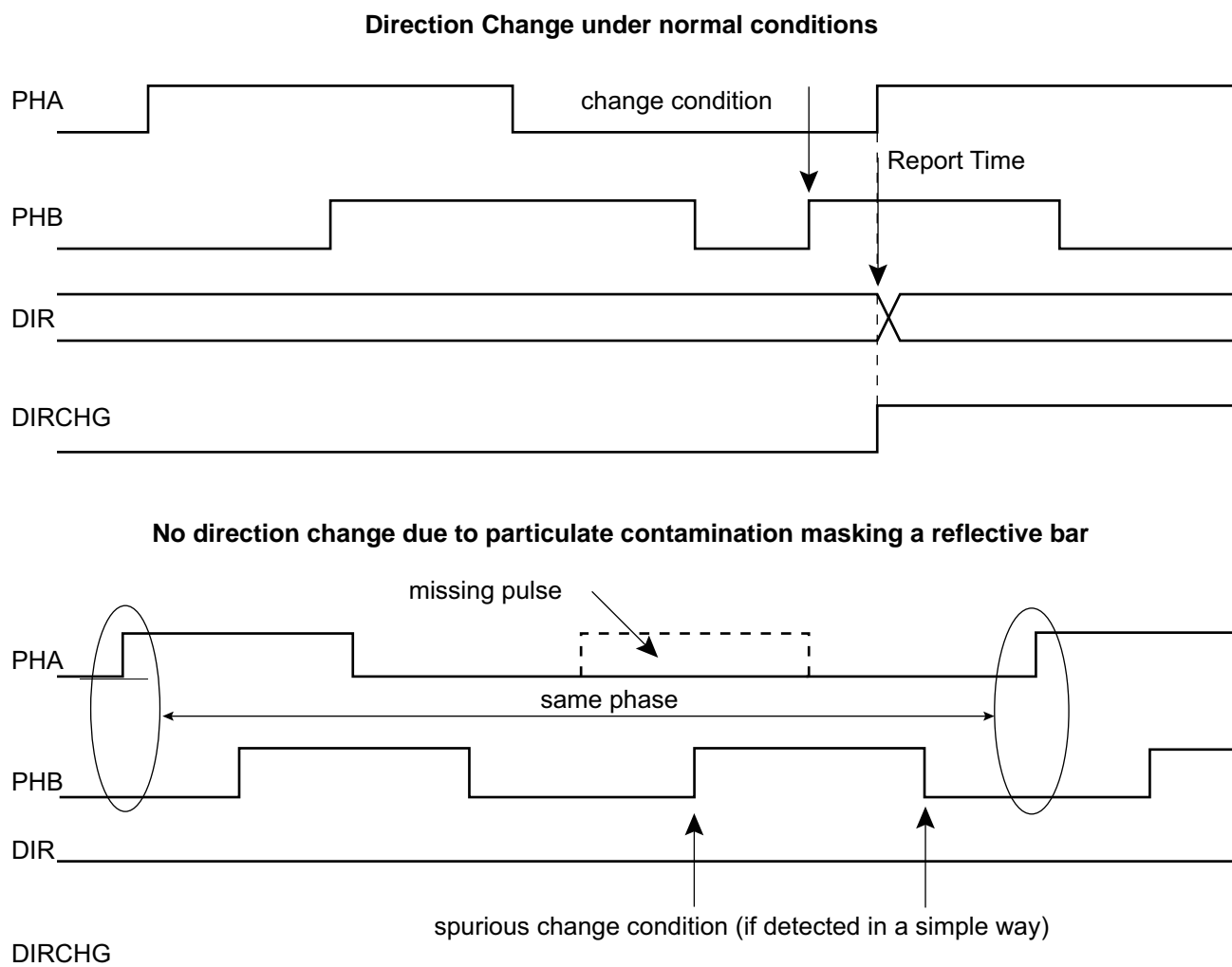
After filtering, the quadrature signals are analyzed to extract the rotation direction and edges of the two quadrature signals detected in order to be counted by timer/counter logic downstream.

The direction status can be directly read at anytime in the TC\_QISR. The polarity of the direction flag status depends on the configuration written in TC\_BMR. INVA, INVB, INVDX, SWAP modify the polarity of DIR flag.

Any change in rotation direction is reported in the TC\_QISR and can generate an interrupt.

The direction change condition is reported as soon as two consecutive edges on a phase signal have sampled the same value on the other phase signal and there is an edge on the other signal. The two consecutive edges of one phase signal sampling the same value on other phase signal is not sufficient to declare a direction change, for the reason that particulate contamination may mask one or more reflective bars on the optical or magnetic disk of the sensor. Refer to Figure 37-18 for waveforms.

Figure 37-18. Rotation Change Detection



The direction change detection is disabled when QDTRANS is set in the TC\_BMR. In this case, the DIR flag report must not be used.

A quadrature error is also reported by the QDEC via the QERR flag in the TC\_QISR. This error is reported if the time difference between two edges on PHA, PHB is lower than a predefined value. This predefined value is

- Configuration of the period for each channel (CPRD in the PWM\_CPRDx register). Writing in PWM\_CPRDx register is possible while the channel is disabled. After validation of the channel, the user must use PWM\_CPRDUPDx register to update PWM\_CPRDx as explained below.
- Configuration of the duty-cycle for each channel (CDTY in the PWM\_CDTYx register). Writing in PWM\_CDTYx register is possible while the channel is disabled. After validation of the channel, the user must use PWM\_CDTYUPDx register to update PWM\_CDTYx as explained below.
- Configuration of the dead-time generator for each channel (DTH and DTL in PWM\_DTx) if enabled (DTE bit in the PWM\_CMRx). Writing in the PWM\_DTx register is possible while the channel is disabled. After validation of the channel, the user must use PWM\_DTUPDx register to update PWM\_DTx
- Selection of the synchronous channels (SYNCx in the PWM\_SCM register)
- Selection of the moment when the WRDY flag and the corresponding Peripheral DMA Controller transfer request are set (PTRM and PTRCS in the PWM\_SCM register)
- Configuration of the Update mode (UPDM in PWM\_SCM register)
- Configuration of the update period (UPR in PWM\_SCUP register) if needed
- Configuration of the comparisons (PWM\_CMPVx and PWM\_CMPMx)
- Configuration of the event lines (PWM\_ELMRx)
- Configuration of the fault inputs polarity (FPOL in PWM\_FMR)
- Configuration of the fault protection (FMOD and FFIL in PWM\_FMR, PWM\_FPV and PWM\_FPE1)
- Enable of the interrupts (writing CHIDx and FCHIDx in PWM\_IER1, and writing WRDYE, ENDTXE, TXBUFE, UNRE, CMPMx and CMPUx in PWM\_IER2)
- Enable of the PWM channels (writing CHIDx in the PWM\_ENA register)

#### 39.6.5.2 Source Clock Selection Criteria

The large number of source clocks can make selection difficult. The relationship between the value in the PWM Channel Period Register (PWM\_CPRDx) and the PWM Channel Duty Cycle Register (PWM\_CDTYx) helps the user select the appropriate clock. The event number written in the Period Register gives the PWM accuracy. The Duty-Cycle quantum cannot be lower than  $1/CPRD_x$  value. The higher the value of PWM\_CPRDx, the greater the PWM accuracy.

For example, if the user sets 15 (in decimal) in PWM\_CPRDx, the user is able to set a value from between 1 up to 14 in PWM\_CDTYx. The resulting duty-cycle quantum cannot be lower than 1/15 of the PWM period.

#### 39.6.5.3 Changing the Duty-Cycle, the Period and the Dead-Times

It is possible to modulate the output waveform duty-cycle, period and dead-times.

To prevent unexpected output waveform, the user must use the PWM Channel Duty Cycle Update Register (PWM\_CDTYUPDx), the PWM Channel Period Update Register (PWM\_CPRDUPDx) and the PWM Channel Dead Time Update Register (PWM\_DTUPDx) to change waveform parameters while the channel is still enabled.

- If the channel is an asynchronous channel (SYNCx = 0 in PWM Sync Channels Mode Register (PWM\_SCM)), these registers hold the new period, duty-cycle and dead-times values until the end of the current PWM period and update the values for the next period.
- If the channel is a synchronous channel and update method 0 is selected (SYNCx = 1 and UPDM = 0 in PWM\_SCM register), these registers hold the new period, duty-cycle and dead-times values until the bit UPDULOCK is written at '1' (in PWM Sync Channels Update Control Register (PWM\_SCUC)) and the end of the current PWM period, then update the values for the next period.
- If the channel is a synchronous channel and update method 1 or 2 is selected (SYNCx = 1 and UPDM = 1 or 2 in PWM\_SCM register):
  - registers PWM\_CPRDUPDx and PWM\_DTUPDx hold the new period and dead-times values until the bit UPDULOCK is written at '1' (in PWM\_SCUC) and the end of the current PWM period, then update the values for the next period.



### 39.7.22 PWM Output Selection Clear Update Register

**Name:** PWM\_OSCUPD

**Address:** 0x40020058

**Access:** Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	OSCUPL3	OSCUPL2	OSCUPL1	OSCUPL0
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	OSCUPH3	OSCUPH2	OSCUPH1	OSCUPH0

- **OSCUPHx: Output Selection Clear for PWMH output of the channel x**

0: No effect.

1: Dead-time generator output DTOHx selected as PWMH output of channel x at the beginning of the next channel x PWM period.

- **OSCUPLx: Output Selection Clear for PWML output of the channel x**

0: No effect.

1: Dead-time generator output DTOLx selected as PWML output of channel x at the beginning of the next channel x PWM period.

### 41.7.9 ACC Write Protection Status Register

**Name:** ACC\_WPSR

**Address:** 0x400400E8

**Access:** Read-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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	WPVS

- **WPVS: Write Protection Violation Status**

0: No write protection violation has occurred since the last read of ACC\_WPSR.

1: A write protection violation (WPEN = 1) has occurred since the last read of ACC\_WPSR.

- **TAG: Tag Selection Mode**

Value	Name	Description
0	DIS	Tag selection mode disabled. Using USER_SEL to select the channel for the conversion.
1	EN	Tag selection mode enabled

- **MAXS: Maximum Speed Mode**

Value	Name	Description
0	NORMAL	Normal mode
1	MAXIMUM	Maximum speed mode enabled

- **STARTUP: Startup Time Selection**

Value	Name	Description
0	0	0 periods of peripheral clock
1	8	8 periods of peripheral clock
2	16	16 periods of peripheral clock
3	24	24 periods of peripheral clock
4	64	64 periods of peripheral clock
5	80	80 periods of peripheral clock
6	96	96 periods of peripheral clock
7	112	112 periods of peripheral clock
8	512	512 periods of peripheral clock
9	576	576 periods of peripheral clock
10	640	640 periods of peripheral clock
11	704	704 periods of peripheral clock
12	768	768 periods of peripheral clock
13	832	832 periods of peripheral clock
14	896	896 periods of peripheral clock
15	960	960 periods of peripheral clock
16	1024	1024 periods of peripheral clock
17	1088	1088 periods of peripheral clock
18	1152	1152 periods of peripheral clock
19	1216	1216 periods of peripheral clock
20	1280	1280 periods of peripheral clock
21	1344	1344 periods of peripheral clock
22	1408	1408 periods of peripheral clock
23	1472	1472 periods of peripheral clock
24	1536	1536 periods of peripheral clock
25	1600	1600 periods of peripheral clock
26	1664	1664 periods of peripheral clock
27	1728	1728 periods of peripheral clock

## 44.10 Analog Comparator Characteristics

**Table 44-61. Analog Comparator Characteristics**

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_R$	Voltage Range	Analog comparator is supplied by VDDIN	1.62	3.3	3.6	V
$V_{IR}$	Input Voltage Range		GND + 0.2	–	VDDIN - 0.2	V
$V_{IO}$	Input Offset Voltage		–	–	20	mV
$I_{VDDIN}$	Current Consumption (VDDIN)	Low-Power Option (ACC_ACR.ISEL = 0)	–	–	25	$\mu$ A
		High-Speed Option (ACC_ACR.ISEL = 1)	–	–	170	
$V_{hys}$	Hysteresis Voltage	ACC_ACR.HYST = 0x01 or 0x10	–	15	50	mV
		ACC_ACR.HYST = 0x11	–	30	90	
$t_{sa}$	Settling Time	Overdrive > 100 mV; Low-power option	–	–	1	$\mu$ s
		Overdrive > 100 mV; High-speed option	–	–	0.1	

## 44.11 Temperature Sensor

The temperature sensor is connected to channel 15 of the ADC.

The temperature sensor provides an output voltage ( $V_{O\_TS}$ ) that is proportional to absolute temperature (PTAT).  $V_{O\_TS}$  linearly varies with a temperature slope  $dV_{O\_TS}/dT = 4.7$  mV/°C.

$V_{O\_TS}$  equals 1.44 V at  $T_A$  27°C, with a  $\pm 60$  mV accuracy. The  $V_{O\_TS}$  slope versus temperature  $dV_{O\_TS}/dT = 4.7$  mV/°C only shows a  $\pm 7\%$  slight variation over process, mismatch and supply voltage.

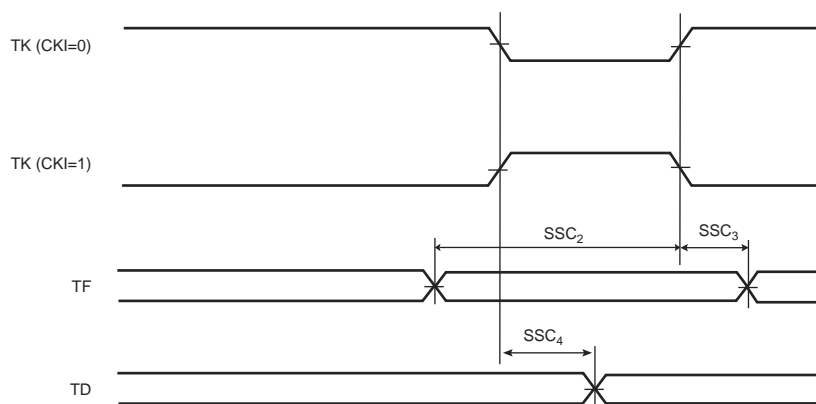
The user needs to calibrate it (offset calibration) at ambient temperature in order to get rid of the  $V_{O\_TS}$  spread at ambient temperature ( $\pm 15\%$ ).

**Table 44-62. Temperature Sensor Characteristics**

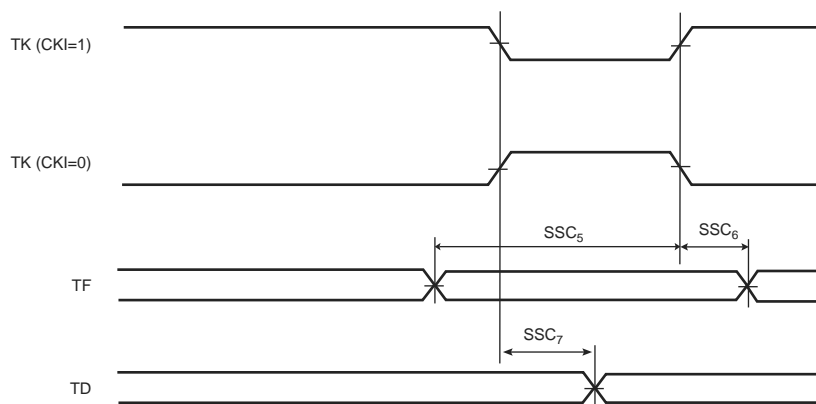
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$V_{O\_TS}$	Output Voltage	$T_A = 27^\circ\text{C}$ <sup>(1)</sup>		1.44	–	V
$V_{O\_TS(\text{accuracy})}$	Output Voltage Accuracy	$T_A = 27^\circ\text{C}$ <sup>(1)</sup>	-60	–	60	mV
$dV_{O\_TS}/dT$	Temperature Sensitivity (Slope Voltage vs Temperature)	<sup>(1)</sup>	–	4.7	–	mV/°C
	Slope Accuracy	Over temperature range -40 to 105 °C <sup>(1)</sup>	-7	–	7	%
	Temperature Accuracy <sup>(2)</sup>	After offset calibration over temperature range -40 to 105 °C	-6	–	6	°C
		After offset calibration over temperature range 0 to 80 °C	-5	–	5	°C
$t_{\text{START}}$	Startup Time	After ADC_ACR.TSON = 1 <sup>(1)</sup>	–	5	10	$\mu$ s
$I_{VDDCORE}$	Current Consumption	<sup>(1)</sup>	50	70	80	$\mu$ A

Note: 1. The value of TS only (the value does not take into account the ADC offset/gain/errors)  
 2. The temperature accuracy takes into account the ADC offset error and gain error in single-ended mode with Gain = 1.

**Figure 44-28. SSC Transmitter, TK as Output and TF as Input**



**Figure 44-29. SSC Transmitter, TK and TF as Input**



**Figure 44-30. SSC Receiver RK and RF as Input**

