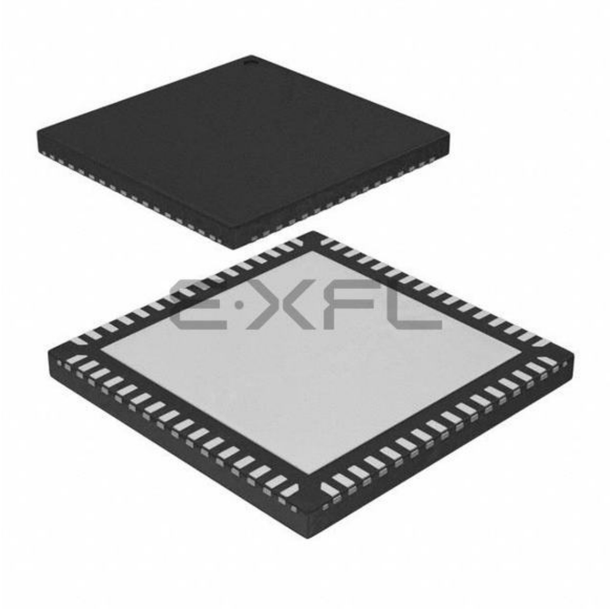
Microchip Technology - ATSAMS70J21A-MN Datasheet



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, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	44
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 5x12b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsams70j21a-mn

Instance ID	Instance Name	NVIC Interrupt	PMC Clock Control	Description
20	TWIHS1	X	X	Two-wire Interface (I2C-compatible)
21	SPI0	X	X	Serial Peripheral Interface
22	SSC	X	X	Synchronous Serial Controller
23	TC0_CHANNEL0	X	X	16-bit Timer Counter 0, Channel 0
24	TC0_CHANNEL1	X	X	16-bit Timer Counter 0, Channel 1
25	TC0_CHANNEL2	X	X	16-bit Timer Counter 0, Channel 2
26	TC1_CHANNEL0	X	X	16-bit Timer Counter 1, Channel 0
27	TC1_CHANNEL1	X	X	16-bit Timer Counter 1, Channel 1
28	TC1_CHANNEL2	X	X	16-bit Timer Counter 1, Channel 2
29	AFEC0	X	X	Analog Front-End Controller
30	DACC	X	X	Digital-to-Analog Converter
31	PWM0	X	X	Pulse Width Modulation Controller
32	ICM	X	X	Integrity Check Monitor
33	ACC	X	X	Analog Comparator Controller
34	USBHS	X	X	USB Host / Device Controller
35	MCAN0	X	X	CAN IRQ Line 0
36	MCAN0	INT1	–	CAN IRQ Line 1
37	MCAN1	X	X	CAN IRQ Line 0
38	MCAN1	INT1	–	CAN IRQ Line 1
39	GMAC	X	X	Ethernet MAC
35	–	–	–	Reserved
36	–	–	–	Reserved
37	–	–	–	Reserved
38	–	–	–	Reserved
39	–	–	–	Reserved
40	AFEC1	X	X	Analog Front End Controller
41	TWIHS2	X	X	Two-wire Interface
42	SPI1	X	X	Serial Peripheral Interface
43	QSPI	X	X	Quad I/O Serial Peripheral Interface
44	UART2	X	X	Universal Asynchronous Receiver/ Transmitter

32.6.1.33 PIO Output Write Enable Register

Name: PIO_OWER
Offset: 0x00A0
Property: Write-only

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

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

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

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

Bit	7	6	5	4	3	2	1	0
	P7	P6	P5	P4	P3	P2	P1	P0
Access								
Reset								

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31 – P PIO Output Write Enable

Value	Description
0	No effect.
1	Enables writing PIO_ODSR for the I/O line.

35.16.1.6 SMC Off-Chip Memory Scrambling Key1 Register

Name: SMC_KEY1
Offset: 0x84
Reset: 0x00000000
Property: Write-once

Note:

1. 'Write-once' access indicates that the first write access after a system reset prevents any further modification of the value of this register.

Note: 1.

Bit	31	30	29	28	27	26	25	24
	KEY1[31:24]							
Access								
Reset	0	0	0	0	0	0	0	0
Bit	23	22	21	20	19	18	17	16
	KEY1[23:16]							
Access								
Reset	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8
	KEY1[15:8]							
Access								
Reset	0	0	0	0	0	0	0	0
Bit	7	6	5	4	3	2	1	0
	KEY1[7:0]							
Access								
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – KEY1[31:0] Off-Chip Memory Scrambling (OCMS) Key Part 1
When off-chip memory scrambling is enabled, KEY1 and KEY2 values determine data scrambling.

38.8.50 GMAC Greater Than 1518 Byte Frames Transmitted Register

Name: GMAC_GTBFT1518
Offset: 0x130
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
	NFTX[31:24]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0
Bit	23	22	21	20	19	18	17	16
	NFTX[23:16]							
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
	NFTX[15:8]							
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
	NFTX[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 31:0 – NFTX[31:0] Greater than 1518 Byte Frames Transmitted without Error
This register counts the number of 1518 or above byte frames successfully transmitted without error i.e., no underrun and not too many retries.

Offset	Name	Bit Pos.								
		23:16								
		31:24								

Bit 16 – RWALL Read/Write Allowed
This bit is set for IN endpoints when the current bank is not full, i.e., the user can write further data into the FIFO.

This bit is set for OUT endpoints when the current bank is not empty, i.e., the user can read further data from the FIFO.

This bit is never set in case of error.

This bit is cleared otherwise.

Bits 15:14 – CURRBK[1:0] Current Bank
This field is used to indicate the current bank. It may be updated one clock cycle after the RWALL bit changes, so the user should not poll this field as an interrupt bit.

Value	Name	Description
0	BANK0	Current bank is bank0
1	BANK1	Current bank is bank1
2	BANK2	Current bank is bank2
3	Reserved	

Bits 13:12 – NBUSYBK[1:0] Number of Busy Banks
This field is set to indicate the number of busy banks:

For IN endpoints, it indicates the number of banks filled by the user and ready for IN transfer. When all banks are free, this triggers a PEP_x interrupt if NBUSYBKE = 1.

For OUT endpoints, it indicates the number of banks filled by OUT transactions from the host. When all banks are busy, this triggers a PEP_x interrupt if NBUSYBKE = 1.

When the USBHS_DEVEPTIMRx.FIFOCON bit is cleared (by writing a one to the USBHS_DEVEPTIMRx.FIFOCONC bit) to validate a new bank, this field is updated two or three clock cycles later to calculate the address of the next bank.

A PEP_x interrupt is triggered if:

Value	Name	Description
0	0_BUSY	0 busy bank (all banks free)
1	1_BUSY	1 busy bank
2	2_BUSY	2 busy banks
3	3_BUSY	3 busy banks • For IN endpoint, USBHS_DEVEPTIMRx.NBUSYBKE = 1 and all the banks are free. • For OUT endpoint, USBHS_DEVEPTIMRx.NBUSYBKE = 1 and all the banks are busy.

Bit 10 – ERRORTRANS High-bandwidth Isochronous OUT Endpoint Transaction Error Interrupt
This bit is set when a transaction error occurs during the current microframe (the data toggle sequencing is not compliant with the USB 2.0 standard). This triggers a PEP_x interrupt if USBHS_DEVEPTIMRx.ERRORTRANSE = 1.

This bit is set as long as the current bank (CURRBK) belongs to the bad n-transactions (n = 1, 2 or 3) transferred during the microframe. It is cleared by software by clearing (at least once) the

Bit 5 – OVERFIES Overflow Interrupt Enable

Bit 4 – NAKEDES NAKed Interrupt Enable

Bit 3 – PERRES Pipe Error Interrupt Enable

Bit 2 – UNDERFIES Underflow Interrupt Enable

Bit 1 – TXOUTES Transmitted OUT Data Interrupt Enable

Bit 0 – RXINES Received IN Data Interrupt Enable

Bit 17 – ACMDIS Alternative Command Mode Disable

Value	Description
0	No effect.
1	Alternative Command mode disabled.

Bit 16 – ACMEN Alternative Command Mode Enable

Value	Description
0	No effect.
1	Alternative Command mode enabled.

Bit 15 – CLEAR Bus CLEAR Command

Value	Description
0	No effect.
1	If Master mode is enabled, sends a bus clear command.

Bit 14 – PECRQ PEC Request

Value	Description
0	No effect.
1	A PEC check or transmission is requested.

Bit 13 – PECDIS Packet Error Checking Disable

Value	Description
0	No effect.
1	SMBus PEC (CRC) generation and check disabled.

Bit 12 – PECEN Packet Error Checking Enable

Value	Description
0	No effect.
1	SMBus PEC (CRC) generation and check enabled.

Bit 11 – SMBDIS SMBus Mode Disabled

Value	Description
0	No effect.
1	SMBus mode disabled.

Bit 10 – SMBEN SMBus Mode Enabled

Value	Description
0	No effect.
1	If SMBDIS = 0, SMBus mode enabled.

Bit 9 – HSDIS TWIHS High-Speed Mode Disabled

Value	Description
0	No effect.
1	High-speed mode disabled.

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.

46.6.9.9 Identifier Parity

A protected identifier consists of two subfields: the identifier and the identifier parity. Bits 0 to 5 are assigned to the identifier and bits 6 and 7 are assigned to the parity.

The USART interface can generate/check these parity bits, but this feature can also be disabled. The user can choose between two modes using US_LINMR.PARDIS:

- PARDIS = 0:
  - During header transmission, the parity bits are computed and sent with the six least significant bits of US_LINIR.IDCHR. The bits 6 and 7 of this register are discarded.
  - During header reception, the parity bits of the identifier are checked. If the parity bits are wrong, an Identifier Parity error occurs (see Parity). Only the six least significant bits of the IDCHR field are updated with the received Identifier. The bits 6 and 7 are stuck to 0.
- PARDIS = 1:
  - During header transmission, all the bits of US_LINIR.IDCHR are sent on the bus.
  - During header reception, all the bits of IDCHR are updated with the received Identifier.

46.6.9.10 Node Action

Depending on the identifier, the node is affected – or not – by the LIN response. Consequently, after sending or receiving the identifier, the USART must be configured. There are three possible configurations:

- PUBLISH: the node sends the response.
- SUBSCRIBE: the node receives the response.
- IGNORE: the node is not concerned by the response, it does not send and does not receive the response.

This configuration is made by the field Node Action (NACT) in the US_LINMR (see USART LIN Mode Register).

Example: a LIN cluster that contains a master and two slaves:

- Data transfer from the master to the slave1 and to the slave2:

NACT(master)=PUBLISH

NACT(slave1)=SUBSCRIBE

NACT(slave2)=SUBSCRIBE

- Data transfer from the master to the slave1 only:

NACT(master)=PUBLISH

NACT(slave1)=SUBSCRIBE

NACT(slave2)=IGNORE

- Data transfer from the slave1 to the master:

NACT(master)=SUBSCRIBE

NACT(slave1)=PUBLISH

NACT(slave2)=IGNORE

- Data transfer from the slave1 to the slave2:

NACT(master)=IGNORE

- Not send/check a checksum (CHKDIS = 1)

This configuration is made by the Checksum Type (CHKTYP) and Checksum Disable (CHKDIS) fields of US_LINMR.

If the checksum feature is disabled, the user can send it manually all the same, by considering the checksum as a normal data byte and by adding 1 to the response data length (see Response Data Length).

46.6.9.13 Frame Slot Mode

This mode is useful only for master nodes. It complies with the following rule: each frame slot should be longer than or equal to $t_{Frame\_Maximum}$.

If the Frame Slot mode is enabled (FSDIS = 0) and a frame transfer has been completed, the TXRDY flag is set again only after $t_{Frame\_Maximum}$ delay, from the start of frame. So the master node cannot send a new header if the frame slot duration of the previous frame is inferior to $t_{Frame\_Maximum}$.

If the Frame Slot mode is disabled (FSDIS = 1) and a frame transfer has been completed, the TXRDY flag is set again immediately.

The $t_{Frame\_Maximum}$ is calculated as below:

If the Checksum is sent (CHKDIS = 0):

$t_{Header\_Nominal} = 34 \times t_{bit}$

$t_{Response\_Nominal} = 10 \times (NData + 1) \times t_{bit}$

$t_{Frame\_Maximum} = 1.4 \times (t_{Header\_Nominal} + t_{Response\_Nominal} + 1)$[1]

$t_{Frame\_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1 + 1) + 1) \times t_{bit}$

$t_{Frame\_Maximum} = (77 + 14 \times DLC) \times t_{bit}$

If the Checksum is not sent (CHKDIS = 1):

$t_{Header\_Nominal} = 34 \times t_{bit}$

$t_{Response\_Nominal} = 10 \times NData \times t_{bit}$

$t_{Frame\_Maximum} = 1.4 \times (t_{Header\_Nominal} + t_{Response\_Nominal} + 1)$[1]

$t_{Frame\_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1) + 1) \times t_{bit}$

$t_{Frame\_Maximum} = (63 + 14 \times DLC) \times t_{bit}$

Note: 1. The term “+1” leads to an integer result for $t_{Frame\_Maximum}$ (LIN Specification 1.3).

Value	Description
0	The USART does not filter the receive line.
1	The USART filters the receive line using a three-sample filter (1/16-bit clock) (2 over 3 majority).

Bits 26:24 – MAX_ITERATION[2:0] Maximum Number of Automatic Iteration

Value	Description
0-7	Defines the maximum number of iterations in ISO7816 mode, protocol T = 0.

Bit 23 – INVDATA Inverted Data

Value	Description
0	The data field transmitted on TXD line is the same as the one written in US_THR or the content read in US_RHR is the same as RXD line. Normal mode of operation.
1	The data field transmitted on TXD line is inverted (voltage polarity only) compared to the value written on US_THR or the content read in US_RHR is inverted compared to what is received on RXD line (or ISO7816 IO line). Inverted mode of operation, useful for contactless card application. To be used with configuration bit MSBF.

Bit 22 – VAR_SYNC Variable Synchronization of Command/Data Sync Start Frame Delimiter

Value	Description
0	User defined configuration of command or data sync field depending on MODSYNC value.
1	The sync field is updated when a character is written into US_THR.

Bit 21 – DSNACK Disable Successive NACK

Value	Description
0	NACK is sent on the ISO line as soon as a parity error occurs in the received character (unless INACK is set).
1	Successive parity errors are counted up to the value specified in the MAX_ITERATION field. These parity errors generate a NACK on the ISO line. As soon as this value is reached, no additional NACK is sent on the ISO line. The flag ITER is asserted. Note: MAX_ITERATION field must be set to 0 if DSNACK is cleared.

Bit 20 – INACK Inhibit Non Acknowledge

Value	Description
0	The NACK is generated.
1	The NACK is not generated.

Bit 19 – OVER Oversampling Mode

Value	Description
0	16X Oversampling
1	8X Oversampling

Bit 18 – CLKO Clock Output Select

Value	Description
0	Logic zero is coded as a zero-to-one transition, Logic one is coded as a one-to-zero transition.
1	Logic zero is coded as a one-to-zero transition, Logic one is coded as a zero-to-one transition.

Bits 25:24 – RX_PP[1:0] Receiver Preamble Pattern detected
The following values assume that RX_MPOL field is not set:

Value	Name	Description
00	ALL_ONE	The preamble is composed of ‘1’s
01	ALL_ZERO	The preamble is composed of ‘0’s
10	ZERO_ONE	The preamble is composed of ‘01’s
11	ONE_ZERO	The preamble is composed of ‘10’s

Bits 19:16 – RX_PL[3:0] Receiver Preamble Length

Value	Description
0	The receiver preamble pattern detection is disabled
1-15	The detected preamble length is RX_PL × Bit Period

Bit 12 – TX_MPOL Transmitter Manchester Polarity

Value	Description
0	Logic zero is coded as a zero-to-one transition, Logic one is coded as a one-to-zero transition.
1	Logic zero is coded as a one-to-zero transition, Logic one is coded as a zero-to-one transition.

Bits 9:8 – TX_PP[1:0] Transmitter Preamble Pattern
The following values assume that TX_MPOL field is not set:

Value	Name	Description
0	ALL_ONE	The preamble is composed of ‘1’s
1	ALL_ZERO	The preamble is composed of ‘0’s
2	ZERO_ONE	The preamble is composed of ‘01’s
3	ONE_ZERO	The preamble is composed of ‘10’s

Bits 3:0 – TX_PL[3:0] Transmitter Preamble Length

Value	Description
0	The transmitter preamble pattern generation is disabled
1-15	The preamble length is TX_PL × Bit Period

Value	Description
0	No effect.
1	The transmitter is disabled. If a character is being processed and a character has been written in the UART_THR and RSTTX is not set, both characters are completed before the transmitter is stopped.

Bit 6 – TXEN Transmitter Enable

Value	Description
0	No effect.
1	The transmitter is enabled if TXDIS is 0.

Bit 5 – RXDIS Receiver Disable

Value	Description
0	No effect.
1	The receiver is disabled. If a character is being processed and RSTRX is not set, the character is completed before the receiver is stopped.

Bit 4 – RXEN Receiver Enable

Value	Description
0	No effect.
1	The receiver is enabled if RXDIS is 0.

Bit 3 – RSTTX Reset Transmitter

Value	Description
0	No effect.
1	The transmitter logic is reset and disabled. If a character is being transmitted, the transmission is aborted.

Bit 2 – RSTRX Reset Receiver

Value	Description
0	No effect.
1	The receiver logic is reset and disabled. If a character is being received, the reception is aborted.

49. Controller Area Network (MCAN)

49.1 Description

The Controller Area Network (MCAN) performs communication according to ISO 11898-1:2015 and to Bosch CAN-FD specification. Additional transceiver hardware is required for connection to the physical layer.

All functions concerning the handling of messages are implemented by the Rx Handler and the Tx Handler. The Rx Handler manages message acceptance filtering, the transfer of received messages from the CAN core to the Message RAM, as well as providing receive message status information. The Tx Handler is responsible for the transfer of transmit messages from the Message RAM to the CAN core, as well as providing transmit status information.

Acceptance filtering is implemented by a combination of up to 128 filter elements, where each element can be configured as a range, as a bit mask, or as a dedicated ID filter.

49.2 Embedded Characteristics

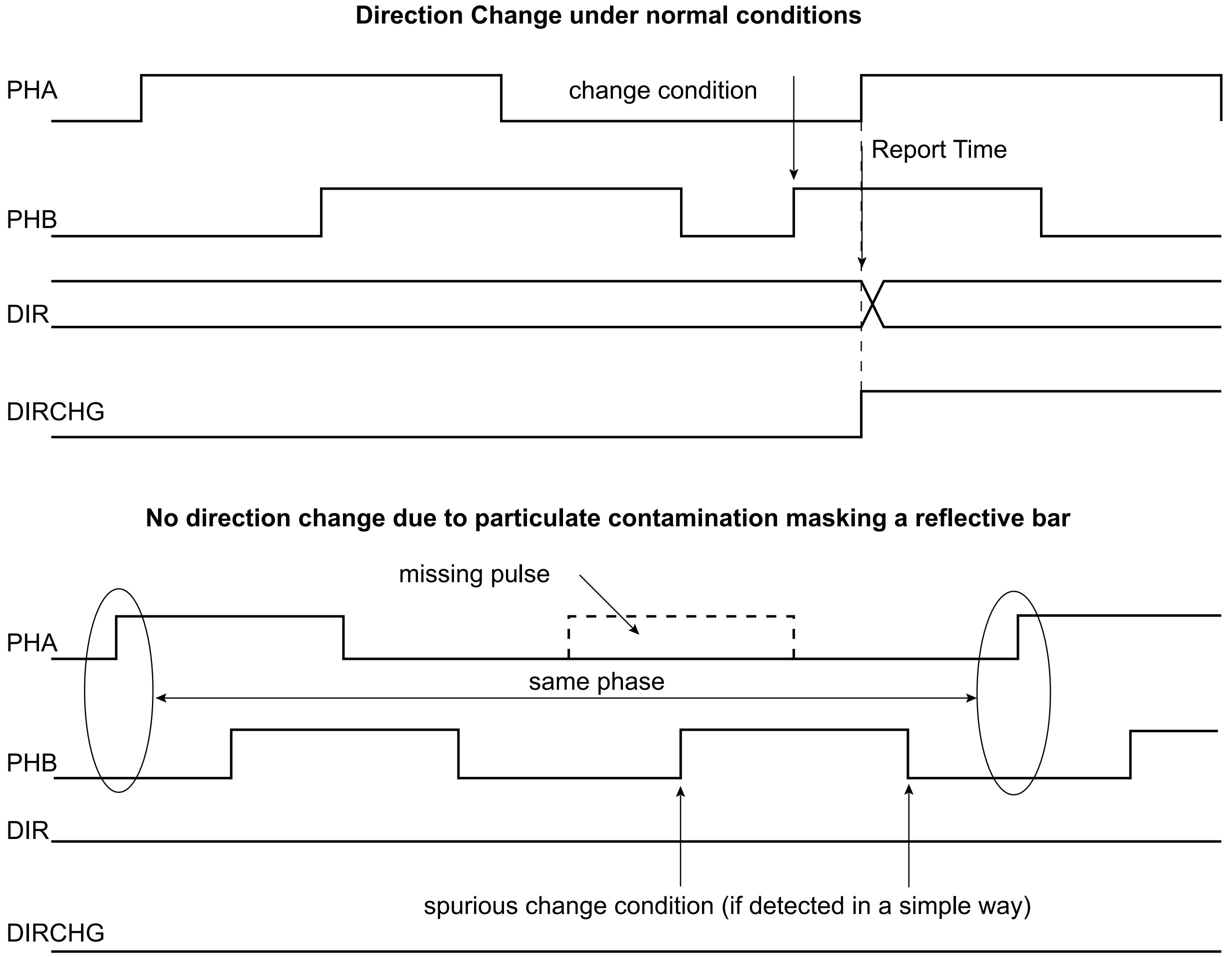
- Compliant with CAN Protocol Version 2.0 Part A, B and ISO 11898-1
- CAN-FD with up to 64 Data Bytes Supported
- CAN Error Logging
- AUTOSAR Optimized
- SAE J1939 Optimized
- Improved Acceptance Filtering
- Two Configurable Receive FIFOs
- Separate Signalling on Reception of High Priority Messages
- Up to 64 Dedicated Receive Buffers
- Up to 32 Dedicated Transmit Buffers
- Configurable Transmit FIFO
- Configurable Transmit Queue
- Configurable Transmit Event FIFO
- Direct Message RAM Access for Processor
- Multiple MCANs May Share the Same Message RAM
- Programmable Loop-back Test Mode
- Maskable Module Interrupts
- Support for Asynchronous CAN and System Bus Clocks
- Power-down Support
- Debug on CAN Support

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, INVIDX, 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, as particulate contamination may mask one or more reflective bars on the optical or magnetic disk of the sensor. Refer to the following figure for waveforms.

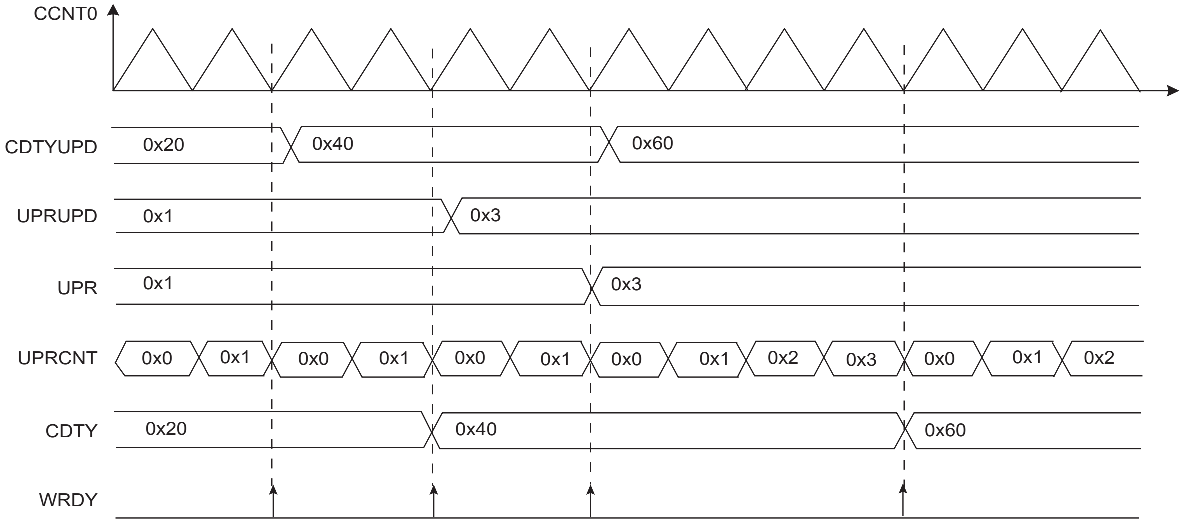
Figure 50-20. Rotation Change Detection


Direction Change under normal conditions
PHA
change condition
Report Time
PHB
DIR
DIRCHG
No direction change due to particulate contamination masking a reflective bar
missing pulse
PHA
same phase
PHB
DIR
spurious change condition (if detected in a simple way)
DIRCHG

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

A quadrature error is also reported by the QDEC via TC_QISR.QERR. This error is reported if the time difference between two edges on PHA, PHB is lower than a predefined value. This predefined value is configurable and corresponds to (TC_BMR.MAXFILT + 1) × $t_{\text{peripheral clock}}$ ns. After being filtered, there is no reason to have two edges closer than (TC_BMR.MAXFILT + 1) × $t_{\text{peripheral clock}}$ ns under normal mode of operation.

Figure 51-20. Method 2 (UPDM = 1)



51.6.2.9.3 Method 3: Automatic write of duty-cycle values and automatic trigger of the update

In this mode, the update of the duty cycle values is made automatically by the DMA Controller. The update of the period value, 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_DTUPDx and PWM_SCUPUPD).

To trigger the update of the period value and the dead-time values, the user must use the bit UPDULOCK which allows to update 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 value is triggered automatically after an update period.

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

Using the DMA Controller removes processor overhead by reducing its intervention during the transfer. This significantly reduces the number of clock cycles required for a data transfer, which improves microcontroller performance.

The DMA Controller must write the duty-cycle values in the synchronous channels index order. For example if the channels 0, 1 and 3 are synchronous channels, the DMA Controller must write the duty-cycle of the channel 0 first, then the duty-cycle of the channel 1, and finally the duty-cycle of the channel 3.

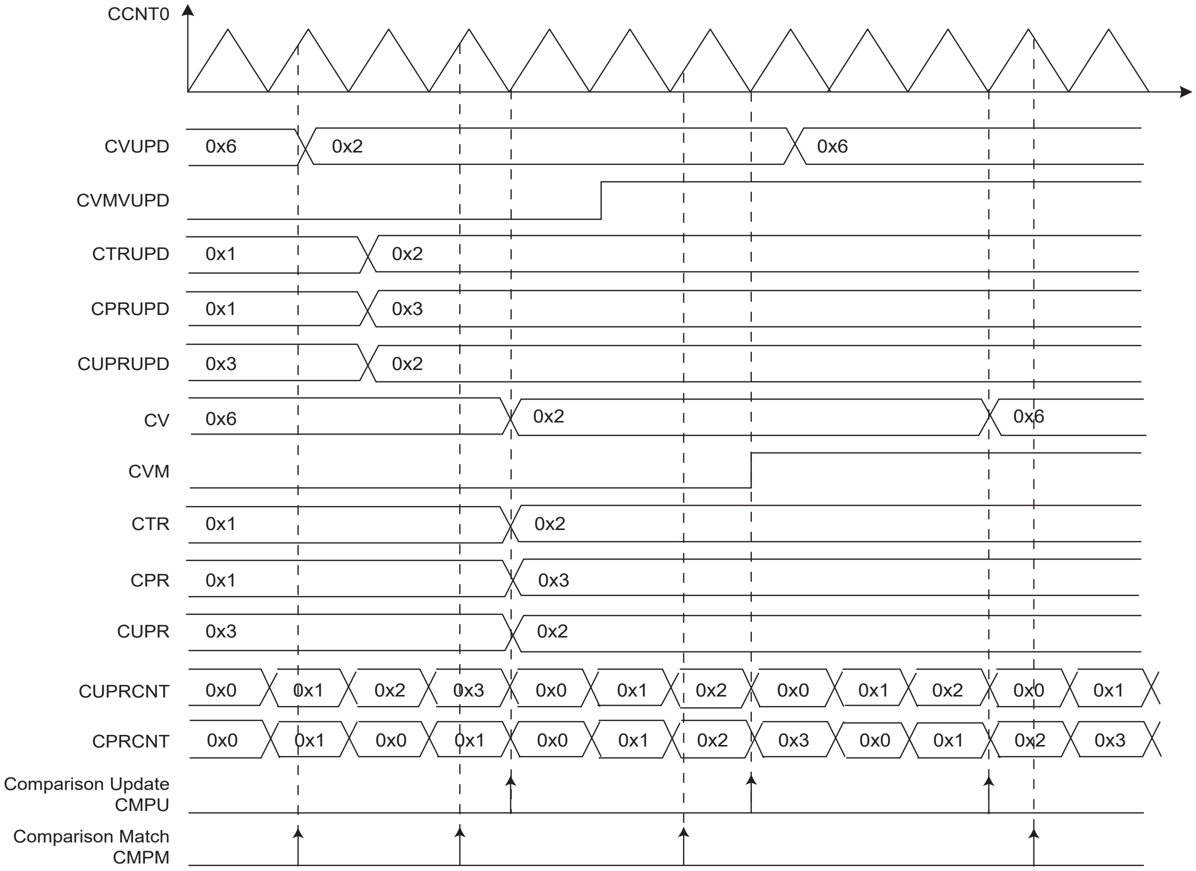
The status of the DMA Controller transfer is reported in PWM_ISR2 by the following flags:

- 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 PWM_ISR2 is read. The user can choose to synchronize the WRDY flag and the DMA Controller transfer request with a comparison match (see PWM Comparison Units), by the fields PTRM and PTRCS in the PWM_SCM register.

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

The comparison match and the comparison update can be source of an interrupt, but only if it is enabled and not masked. These interrupts can be enabled by the PWM Interrupt Enable Register 2 and disabled by the PWM Interrupt Disable Register 2. The comparison match interrupt and the comparison update interrupt are reset by reading the PWM Interrupt Status Register 2.

Figure 51-24. Comparison Waveform



51.6.4 PWM Event Lines

The PWM provides 2 independent event lines intended to trigger actions in other peripherals (e.g., for the Analog-to-Digital Converter (ADC)).

A pulse (one cycle of the peripheral clock) is generated on an event line, when at least one of the selected comparisons is matching. The comparisons can be selected or unselected independently by the CSEL bits in the PWM Event Line x Register (PWM_ELMRx for the Event Line x).

An example of event generation is provided in the figure Event Line Generation Waveform (Example).

51.7.4 PWM Status Register

Name: PWM_SR
Offset: 0x0C
Reset: 0x00000000
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

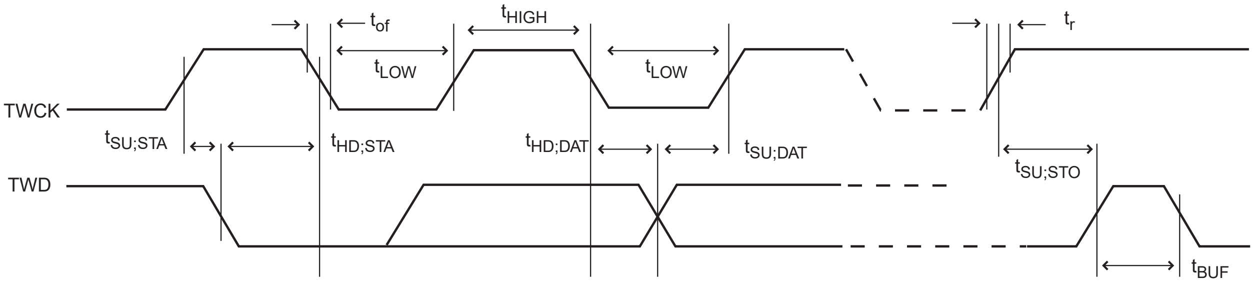
Bit	7	6	5	4	3	2	1	0
					CHID3	CHID2	CHID1	CHID0
Access					R	R	R	R
Reset					0	0	0	0

Bits 0, 1, 2, 3 – CHIDx Channel ID

Value	Description
0	PWM output for channel x is disabled.
1	PWM output for channel x is enabled.

5. t_{CPMCK} = MCK bus period

Figure 59-29. Two-wire Serial Bus Timing



59.13.1.13 GMAC Characteristics

59.13.1.13.1 Timing Conditions

Table 59-69. Load Capacitance on Data, Clock Pads

Supply	C_L	
	Max	Min
3.3V	20 pF	0 pF

59.13.1.13.2 Timing Constraints

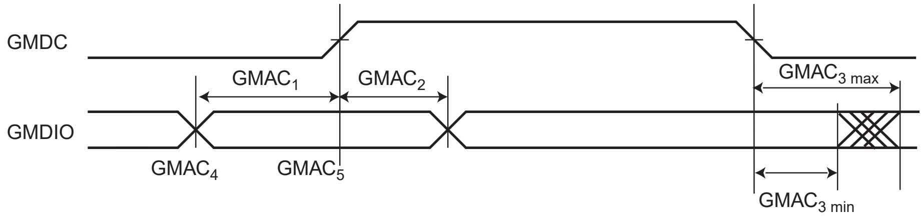
The GMAC must be constrained so as to satisfy the timings of standards shown below and in 58.13.1.13.3 MII Mode, in MAX corner.

Table 59-70. GMAC Signals Relative to GMDC

Symbol	Parameter	Min	Max	Unit
$GMAC_1$	Setup for GMDIO from GMDC rising	10	–	ns
$GMAC_2$	Hold for GMDIO from GMDC rising	10	–	
$GMAC_3$	GMDIO toggling from GMDC falling	0[1]	10[1]	

Note: 1. For GMAC output signals, min and max access time are defined. The min access time is the time between the GMDC falling edge and the signal change. The max access timing is the time between the GMDC falling edge and the signal stabilizes. The figure below illustrates min and max accesses for $GMAC_3$.

Figure 59-30. Min and Max Access Time of GMAC Output Signals


GMDC
$GMAC_1$
$GMAC_2$
$GMAC_{3\ max}$
GMDIO
$GMAC_4$
$GMAC_5$
$GMAC_{3\ min}$