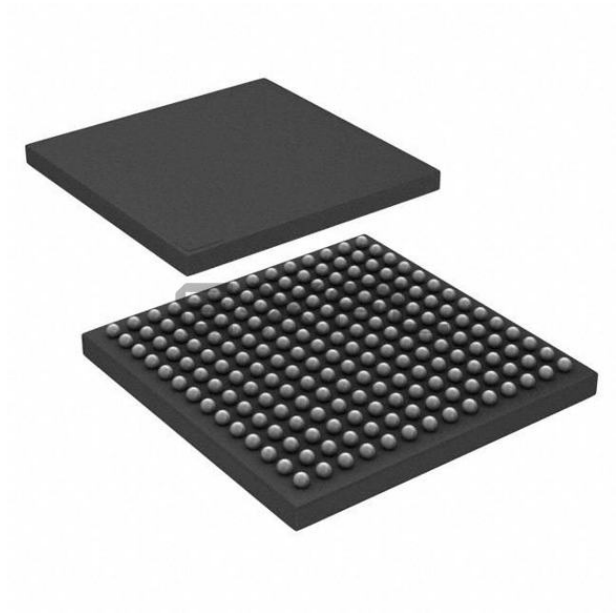


Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	ARM® Cortex®-A5
Number of Cores/Bus Width	1 Core, 32-Bit
Speed	500MHz
Co-Processors/DSP	Multimedia; NEON™ MPE
RAM Controllers	LPDDR1, LPDDR2, LPDDR3, DDR2, DDR3, DDR3L, QSPI
Graphics Acceleration	Yes
Display & Interface Controllers	Keyboard, LCD, Touchscreen
Ethernet	10/100Mbps (1)
SATA	-
USB	USB 2.0 + HSIC
Voltage - I/O	3.3V
Operating Temperature	-40°C ~ 105°C (TA)
Security Features	ARM TZ, Boot Security, Cryptography, RTIC, Secure Fusebox, Secure JTAG, Secure Memory, Secure RTC
Package / Case	196-TFBGA, CSBGA
Supplier Device Package	196-TFBGA (11x11)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsama5d22b-cnr

22. Watchdog Timer (WDT)

22.1 Description

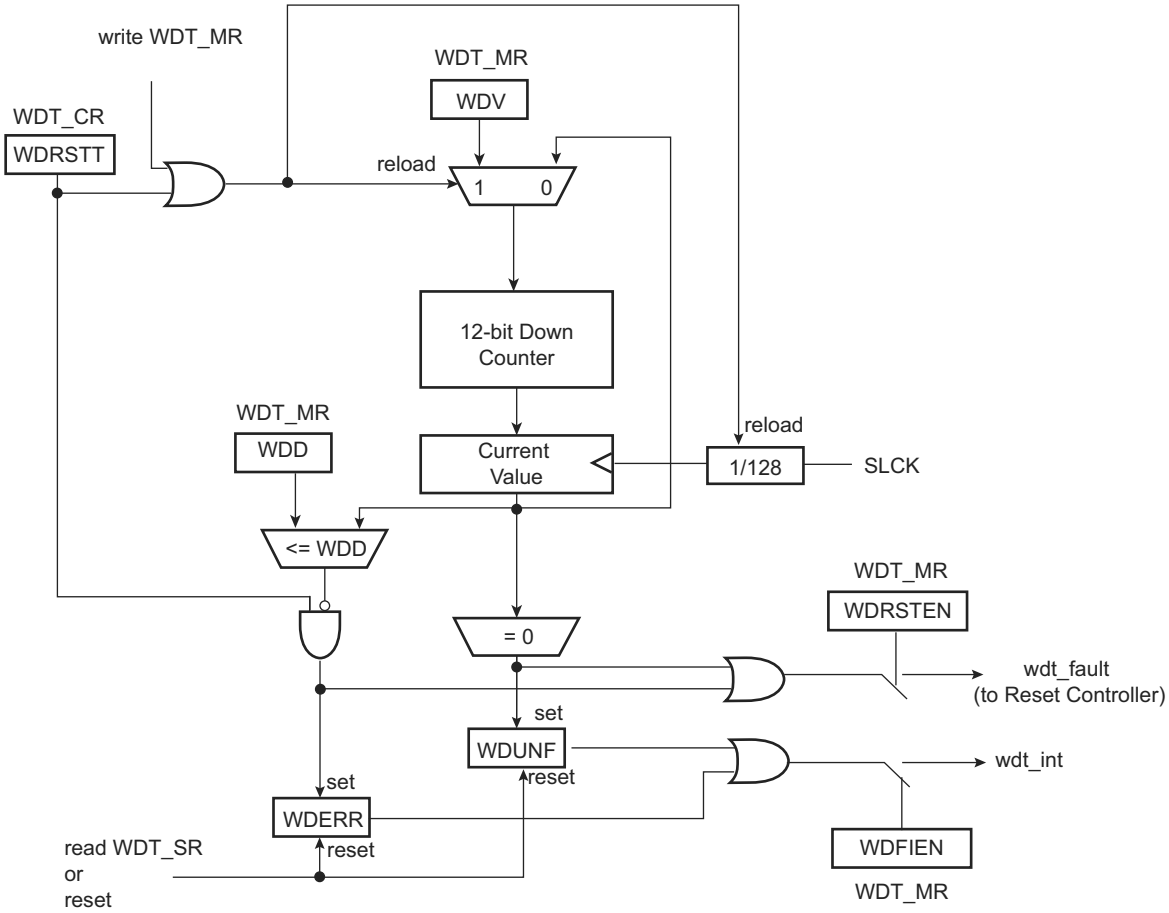
The Watchdog Timer (WDT) is used to prevent system lock-up if the software becomes trapped in a deadlock. It features a 12-bit down counter that allows a watchdog period of up to 16 seconds (slow clock around 32 kHz). It can generate a general reset or a processor reset only. In addition, it can be stopped while the processor is in Debug mode or Sleep mode (Idle mode).

22.2 Embedded Characteristics

- 12-bit Key-protected Programmable Counter
- Watchdog Clock is Independent from Processor Clock
- Provides Reset or Interrupt Signals to the System
- Counter May Be Stopped while the Processor is in Debug State or in Idle Mode

22.3 Block Diagram

Figure 22-1: Watchdog Timer Block Diagram



CALEVSEL: Calendar Event Selection

The event that generates the flag CALEV in RTC_SR depends on the value of CALEVSEL

Value	Name	Description
0	WEEK	Week change (every Monday at time 00:00:00)
1	MONTH	Month change (every 01 of each month at time 00:00:00)
2	YEAR	Year change (every January 1 at time 00:00:00)
3	–	Reserved

Note: In UTC mode, this field has no effect on the RTC_SR.

31. Analog Comparator Controller (ACC)

31.1 Description

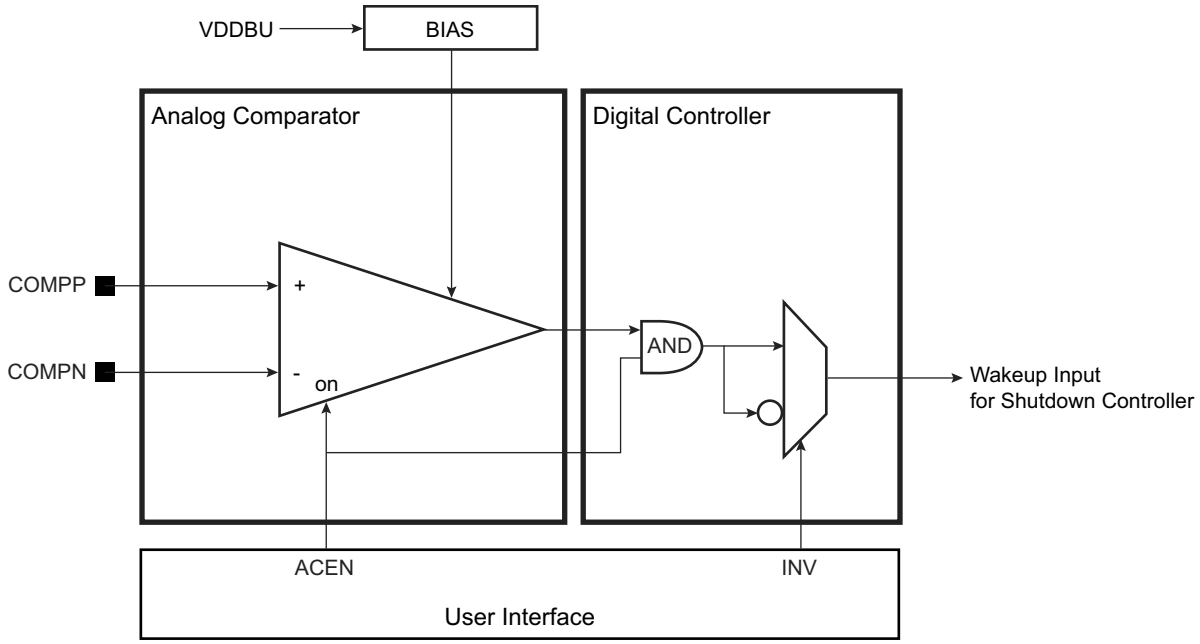
The Analog Comparator Controller (ACC) controls the analog comparator in order to provide an additional source of wakeup when the system wakes up from Wait mode.

31.2 Embedded Characteristics

- Source of Wakeup When System Wakes Up from Wait Mode and ULP1 Mode

31.3 Block Diagram

Figure 31-1: Analog Comparator Controller Block Diagram



31.4 Signal Description

Table 31-1: ACC Signal Description

Pin Name	Description	Type
COMPP, COMPN	External analog data inputs	Input

31.5 Product Dependencies

31.5.1 I/O Lines

The analog input pins (COMPP and COMPN) are not multiplexed with digital functions (PIO) on the IO line.

31.5.2 Power Management

By clearing the ACEN bit in the ACC Mode Register (ACC_MR), the analog comparator power consumption is reduced to current leakage only.

36.7.21 MPDDRC Current/Maximum Bandwidth Port 4-5-6-7 Register

Name: MPDDRC_BDW_PORT_4567

Address: 0xF000C058

Access: Read-only

31	30	29	28	27	26	25	24
–	BDW_P7						
23	22	21	20	19	18	17	16
–	BDW_P6						
15	14	13	12	11	10	9	8
–	BDW_P5						
7	6	5	4	3	2	1	0
–	BDW_P4						

BDW_Px: Current/Maximum Bandwidth from Port 4-5-6-7

Reset value is 0.

This field displays the current bandwidth or the maximum bandwidth for each port. This information is given in the “BDW_MAX_CUR: Bandwidth Max or Current” field description.

36.7.23 MPDDRC Monitor Configuration Register

Name: MPDDRC_MCFGR

Address: 0xF000C060

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
–	–	–	INFO		REFR_CALIB	READ_WRITE	
7	6	5	4	3	2	1	0
–	–	–	RUN	–	–	SOFT_RESET	EN_MONI

EN_MONI: Enable Monitor

0: Monitor is disabled.

1: Monitor is enabled.

SOFT_RESET: Soft Reset

0: Soft reset is not performed.

1: Soft reset is performed.

RUN: Control Monitor

0: Monitoring is halted. All counters are stopped.

1: Monitoring is launched.

READ_WRITE: Read/Write Access

This field is used to monitor different types of access.

Value	Name	Description
0	TRIG_RD_WR	Read and Write accesses are triggered.
1	TRIG_WR	Only Write accesses are triggered.
2	TRIG_RD	Only Read accesses are triggered.
3	–	Reserved

REFR_CALIB: Refresh Calibration

0: Monitoring depends on the refresh and calibration impact.

1: Monitoring depends on the refresh and calibration impact.

39.7.62 Overlay 2 Head Register

Name: LCDC_OVR2HEAD

Address: 0xF000025C

Access: Read/Write

31	30	29	28	27	26	25	24
HEAD							

23	22	21	20	19	18	17	16
HEAD							

15	14	13	12	11	10	9	8
HEAD							

7	6	5	4	3	2	1	0
HEAD						–	–

HEAD: DMA Head Pointer

The Head Pointer points to a new descriptor.

- A buffer descriptor with its ownership bit set is read.
- Bit 10, THALT, of the Network Control register is written.
- There is a transmit error such as too many retries or a transmit underrun.

To set TXGO, write TSTART to the bit 9 of the Network Control register. Transmit halt does not take effect until any ongoing transmit finishes.

If the DMA is configured for packet buffer Partial Store and Forward mode and a collision occurs during transmission of a multi-buffer frame, transmission will automatically restart from the first buffer of the frame. For packet buffer mode, the entire contents of the frame are read into the transmit packet buffer memory, so the retry attempt will be replayed directly from the packet buffer memory rather than having to re-fetch through the AHB.

If an used bit is read midway through transmission of a multi-buffer frame, this is treated as a transmit error. Transmission stops, GTXER is asserted and the FCS will be bad.

If transmission stops due to a transmit error or a used bit being read, transmission restarts from the first buffer descriptor of the frame being transmitted when the transmit start bit is rewritten.

Table 40-5: Transmit Buffer Descriptor Entry

Bit	Function
	Word 0
31:0	Byte address of buffer
	Word 1
31	Used—must be zero for the GMAC to read data to the transmit buffer. The GMAC sets this to one for the first buffer of a frame once it has been successfully transmitted. Software must clear this bit before the buffer can be used again.
30	Wrap—marks last descriptor in transmit buffer descriptor list. This can be set for any buffer within the frame.
29	Retry limit exceeded, transmit error detected
28	Reserved.
27	Transmit frame corruption due to AHB error—set if an error occurs while midway through reading transmit frame from the AHB, including HRESP errors and buffers exhausted mid frame (if the buffers run out during transmission of a frame then transmission stops, FCS shall be bad and GTXER asserted). Also set if single frame is too large for configured packet buffer memory size.
26	Late collision, transmit error detected.
25:23	Reserved
22:20	Transmit IP/TCP/UDP checksum generation offload errors: 000: No Error. 001: The Packet was identified as a VLAN type, but the header was not fully complete, or had an error in it. 010: The Packet was identified as a SNAP type, but the header was not fully complete, or had an error in it. 011: The Packet was not of an IP type, or the IP packet was invalidly short, or the IP was not of type IPv4/IPv6. 100: The Packet was not identified as VLAN, SNAP or IP. 101: Non supported packet fragmentation occurred. For IPv4 packets, the IP checksum was generated and inserted. 110: Packet type detected was not TCP or UDP. TCP/UDP checksum was therefore not generated. For IPv4 packets, the IP checksum was generated and inserted. 111: A premature end of packet was detected and the TCP/UDP checksum could not be generated.
19:17	Reserved
16	No CRC to be appended by MAC. When set, this implies that the data in the buffers already contains a valid CRC, hence no CRC or padding is to be appended to the current frame by the MAC. This control bit must be set for the first buffer in a frame and will be ignored for the subsequent buffers of a frame. Note that this bit must be clear when using the transmit IP/TCP/UDP checksum generation offload, otherwise checksum generation and substitution will not occur.

- A priority enable vector taken from Transmit PFC Pause register
- 8 Pause Quantum registers
- Fill of 00 to take the frame to minimum frame length
- Valid FCS

The Pause Quantum registers used in the generated frame will depend on the trigger source for the frame as follows:

- If bit 17 of the Network Control register is written with a one, then the priority enable vector of the priority-based pause frame will be set equal to the value stored in the Transmit PFC Pause register [7:0]. For each entry equal to zero in the Transmit PFC Pause register [15:8], the pause quantum field of the pause frame associated with that entry will be taken from the Transmit Pause Quantum register. For each entry equal to one in the Transmit PFC Pause register [15:8], the pause quantum associated with that entry will be zero.
- The Transmit Pause Quantum register resets to a value of 0xFFFF giving maximum pause quantum as default.

After transmission, a pause frame transmitted interrupt will be generated (bit 14 of the Interrupt Status register) and the only statistics register that will be incremented will be the Pause Frames Transmitted register.

PFC Pause frames can also be transmitted by the MAC using normal frame transmission methods.

3: The index range for the following registers is from 1 to 2:
- GMAC_ISRPQ
- GMAC_TBQBAPQ
- GMAC_RBQBAPQ
- GMAC_RBSRPQ
- GMAC_IERPQ (cont'd.)
- GMAC_IDRPQ
- GMAC_IMRPQ

4: The index for GMAC_ST1RPQ registers ranges from 0 to 3.

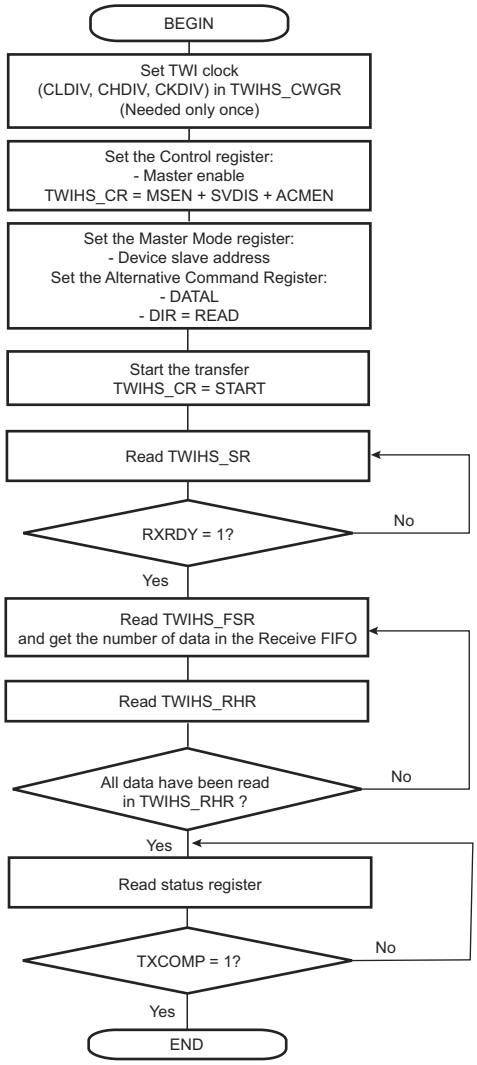
5: The index for GMAC_ST2RPQ registers ranges from 0 to 7.

6: The index for GMAC_ST2ER registers ranges from 0 to 3.

7: The index for GMAC_ST2CW0 and GMAC_ST2CW1 registers ranges from 0 to 23.

When data are present in the Receive FIFO (RXRDY flag set to ‘1’), the exact number of data can be checked with the RXFL field in the TWIHS_FLR and all the data read successively in the TWIHS_RHR without checking the RXRDY flag between each access.

Figure 46-30: Receiving Data with FIFO Flowchart



- Clearing/Flushing FIFOs

Each FIFO can be cleared/flushed using the TXFCLR and RXFCLR bits in the TWIHS_CR.

- TXRDY and RXRDY Behavior

If FIFOs are enabled, the behavior of the TXRDY and RXRDY flags will be slightly different.

TXRDY will indicate if a data can be written in the Transmit FIFO. By default, the TXRDY flag will then stay at level ‘1’ as long as the Transmit FIFO is not full (TXRDYM = 0x0).

2: At the end of the read sequence, TXCOMP is set after a STOP or after a REPEATED_START + an address different from SADR.

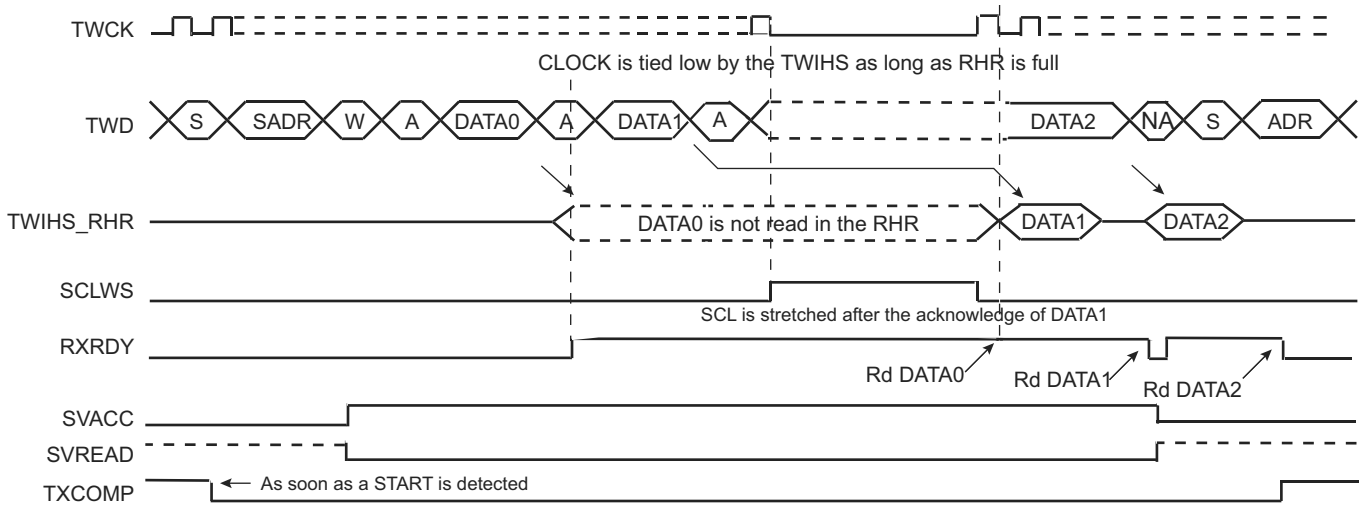
3: SCLWS is automatically set when the clock stretching mechanism is started.

Clock Stretching in Write Mode

The clock is tied low if the internal shifter and the TWIHS_RHR is full. If a STOP or REPEATED_START condition was not detected, it is tied low until TWIHS_RHR is read.

Figure 46-40 describes the clock stretching in Write mode.

Figure 46-40: Clock Stretching in Write Mode



Note 1: At the end of the read sequence, TXCOMP is set after a STOP or after a REPEATED_START + an address different from SADR.

2: SCLWS is automatically set when the clock stretching mechanism is started and automatically reset when the mechanism is finished.

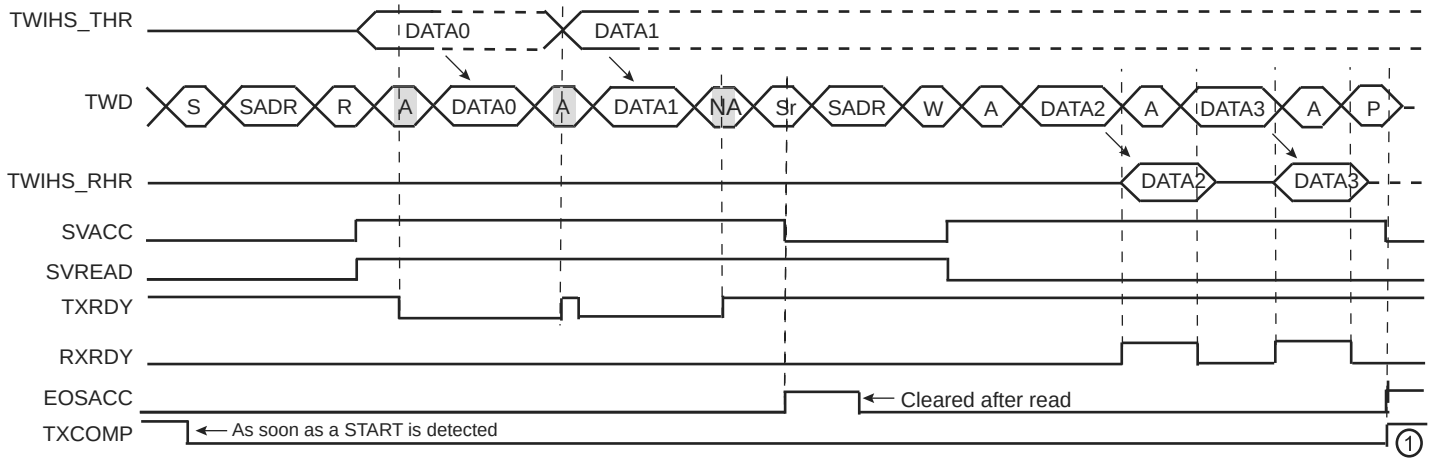
- Reversal after a Repeated Start

Reversal of Read to Write

The master initiates the communication by a read command and finishes it by a write command.

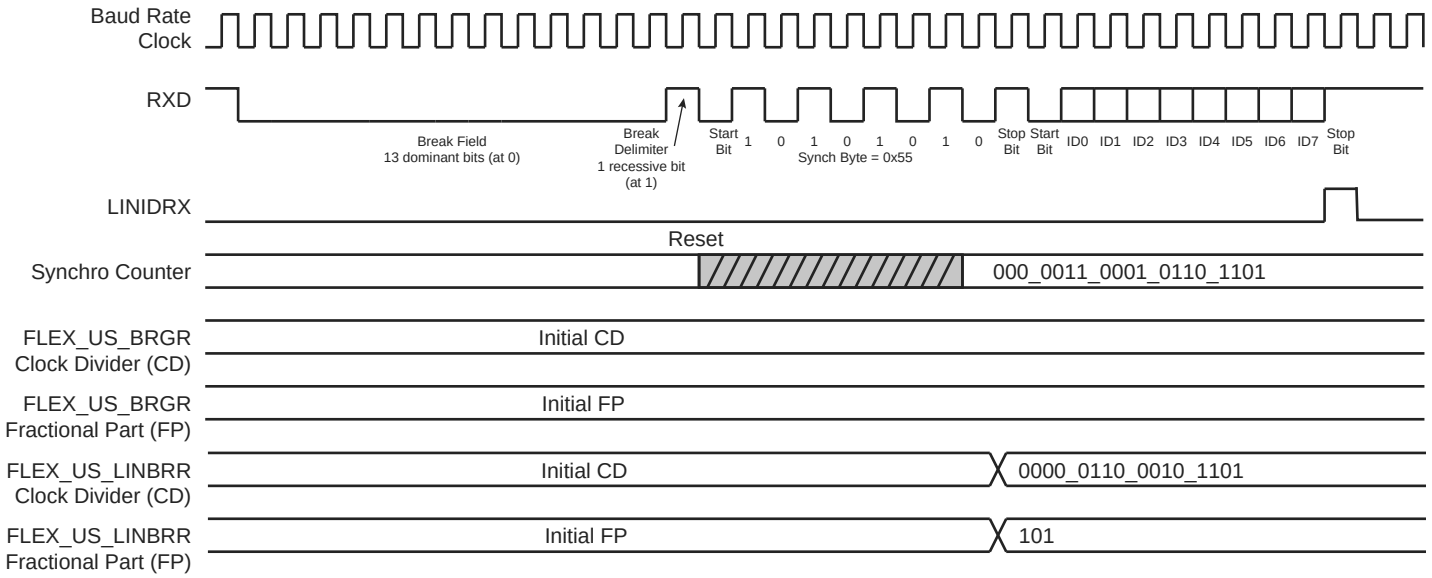
Figure 46-41 describes the REPEATED START and the reversal from Read mode to Write mode.

Figure 46-41: Repeated Start and Reversal from Read Mode to Write Mode



Note: TXCOMP is only set at the end of the transmission. This is because after the REPEATED START, SADR is detected again.

Figure 47-44: Slave Node Synchronization



The synchronization accuracy depends on several parameters:

- The nominal clock frequency (f_{Nom}) (the theoretical slave node clock frequency)
- The baud rate
- The oversampling (OVER = 0 => 16X or OVER = 1 => 8X)

The following formula is used to compute the deviation of the slave bit rate relative to the master bit rate after synchronization (f_{SLAVE} is the real slave node clock frequency).

$$\text{Baud rate deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baud rate}}{8 \times f_{SLAVE}}\right)\%$$

$$\text{Baud rate deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baud rate}}{8 \times \left(\frac{f_{TOL\_UNSYNCH}}{100}\right) \times f_{Nom}}\right)\%$$

$$-0.5 \le \alpha \le +0.5 \quad -1 < \beta < +1$$

$f_{TOL\_UNSYNCH}$ is the deviation of the real slave node clock from the nominal clock frequency. The LIN Standard imposes that it must not exceed ±15%. The LIN Standard imposes also that for communication between two nodes, their bit rate must not differ by more than ±2%. This means that the baud rate deviation must not exceed ±1%.

Therefore, a minimum value for the nominal clock frequency can be computed as follows:

$$f_{Nom}(\text{min}) = \left(100 \times \frac{[0.5 \times 8 \times (2 - \text{Over}) + 1] \times \text{Baud rate}}{8 \times \left(\frac{-15}{100} + 1\right) \times 1\%}\right)\text{Hz}$$

Examples:

- Baud rate = 20 kbit/s, OVER = 0 (Oversampling 16X) => f_{Nom}(min) = 2.64 MHz
- Baud rate = 20 kbit/s, OVER = 1 (Oversampling 8X) => f_{Nom}(min) = 1.47 MHz
- Baud rate = 1 kbit/s, OVER = 0 (Oversampling 16X) => f_{Nom}(min) = 132 kHz
- Baud rate = 1 kbit/s, OVER = 1 (Oversampling 8X) => f_{Nom}(min) = 74 kHz

47.8 SPI Functional Description

47.8.1 Modes of Operation

The SPI operates in Master mode or in Slave mode.

- The SPI operates in Master mode by writing a 1 to the MSTR bit in the SPI Mode Register (FLEX_SPI_MR):
  - The pins NPCS0 to NPCS1 are all configured as outputs.
  - The SPCK pin is driven.
  - The MISO line is wired on the receiver input.
  - The MOSI line is driven as an output by the transmitter.
- The SPI operates in Slave mode if the MSTR bit in FLEX_SPI_MR is written to 0:
  - The MISO line is driven by the transmitter output.
  - The MOSI line is wired on the receiver input.
  - The SPCK pin is driven by the transmitter to synchronize the receiver.
  - The NPCS0 pin becomes an input, and is used as a slave select signal (NSS).
  - Pin NPCS1 is not are not are not driven and can be used for other purposes.

The data transfers are identically programmable for both modes of operation. The bit rate generator is activated only in Master mode.

47.8.2 Data Transfer

Four combinations of polarity and phase are available for data transfers. The clock polarity is programmed with the CPOL bit in the SPI Chip Select Register (FLEX_SPI_CSR). The clock phase is programmed with the NCPHA bit. These two parameters determine the edges of the clock signal on which data are driven and sampled. Each of the two parameters has two possible states, resulting in four possible combinations that are incompatible with one another. Consequently, a master/slave pair must use the same parameter pair values to communicate. If multiple slaves are connected and require different configurations, the master must reconfigure itself each time it needs to communicate with a different slave.

Table 47-17 shows the four modes and corresponding parameter settings.

Table 47-17: SPI Bus Protocol Mode

SPI Mode	CPOL	NCPHA	Shift SPCK Edge	Capture SPCK Edge	SPCK Inactive Level
0	0	1	Falling	Rising	Low
1	0	0	Rising	Falling	Low
2	1	1	Rising	Falling	High
3	1	0	Falling	Rising	High

Figure 47-65 and Figure 47-66 show examples of data transfers.

47.10.28 USART Number of Errors Register

Name: FLEX_US_NER

Address: 0xF8034244 (0), 0xF8038244 (1), 0xFC010244 (2), 0xFC014244 (3), 0xFC018244 (4)

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
NB_ERRORS							

This register is relevant only if USART_MODE = 0x4 or 0x6 in the USART Mode Register.

NB_ERRORS: Number of Errors

Total number of errors that occurred during an ISO7816 transfer. This register automatically clears when read.

RXFPTEF: Receive FIFO Pointer Error Flag

0: No Receive FIFO pointer occurred

1: Receive FIFO pointer error occurred. Receiver must be reset

See Section 47.7.11.9 ”FIFO Pointer Error” for details.

TXFLOCK: Transmit FIFO Lock

0: The Transmit FIFO is not locked.

1: The Transmit FIFO is locked.

RXFTHF2: Receive FIFO Threshold Flag 2 (cleared by writing the FLEX_US_CR.RSTSTA bit)

0: Number of unread data in Receive FIFO is above RXFTHRES threshold.

1: Number of unread data in Receive FIFO has reached RXFTHRES2 threshold since the last RSTSTA command was issued.

47.10.62 TWI Master Mode Register

Name: FLEX_TWI_MMR

Address: 0xF8034604 (0), 0xF8038604 (1), 0xFC010604 (2), 0xFC014604 (3), 0xFC018604 (4)

Access: Read/Write

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

23	22	21	20	19	18	17	16
–	DADR						

15	14	13	12	11	10	9	8
–	–	–	MREAD	–	–	IADRSZ	

7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	–

IADRSZ: Internal Device Address Size

Value	Name	Description
0	NONE	No internal device address
1	1_BYTE	One-byte internal device address
2	2_BYTE	Two-byte internal device address
3	3_BYTE	Three-byte internal device address

MREAD: Master Read Direction

0: Master write direction.

1: Master read direction.

DADR: Device Address

The device address is used to access slave devices in Read or Write mode. Those bits are only used in Master mode.

49.8.10 SPI Interrupt Disable Register

Name: SPI_IDR

Address: 0xF8000018 (0), 0xFC000018 (1)

Access: Write-only

31	30	29	28	27	26	25	24
RXFPTEF	TXFPTEF	RXFTHF	RXFFF	RXFEF	TXFTHF	TXFFF	TXFEF
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	CMP	UNDES	TXEMPTY	NSSR
7	6	5	4	3	2	1	0
–	–	–	–	OVRES	MODF	TDRE	RDRF

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

0: No effect.

1: Disables the corresponding interrupt.

RDRF: Receive Data Register Full Interrupt Disable

TDRE: SPI Transmit Data Register Empty Interrupt Disable

MODF: Mode Fault Error Interrupt Disable

OVRES: Overrun Error Interrupt Disable

NSSR: NSS Rising Interrupt Disable

TXEMPTY: Transmission Registers Empty Disable

UNDES: Underrun Error Interrupt Disable

CMP: Comparison Interrupt Disable

TXFEF: TXFEF Interrupt Disable

TXFFF: TXFFF Interrupt Disable

TXFTHF: TXFTHF Interrupt Disable

RXFEF: RXFEF Interrupt Disable

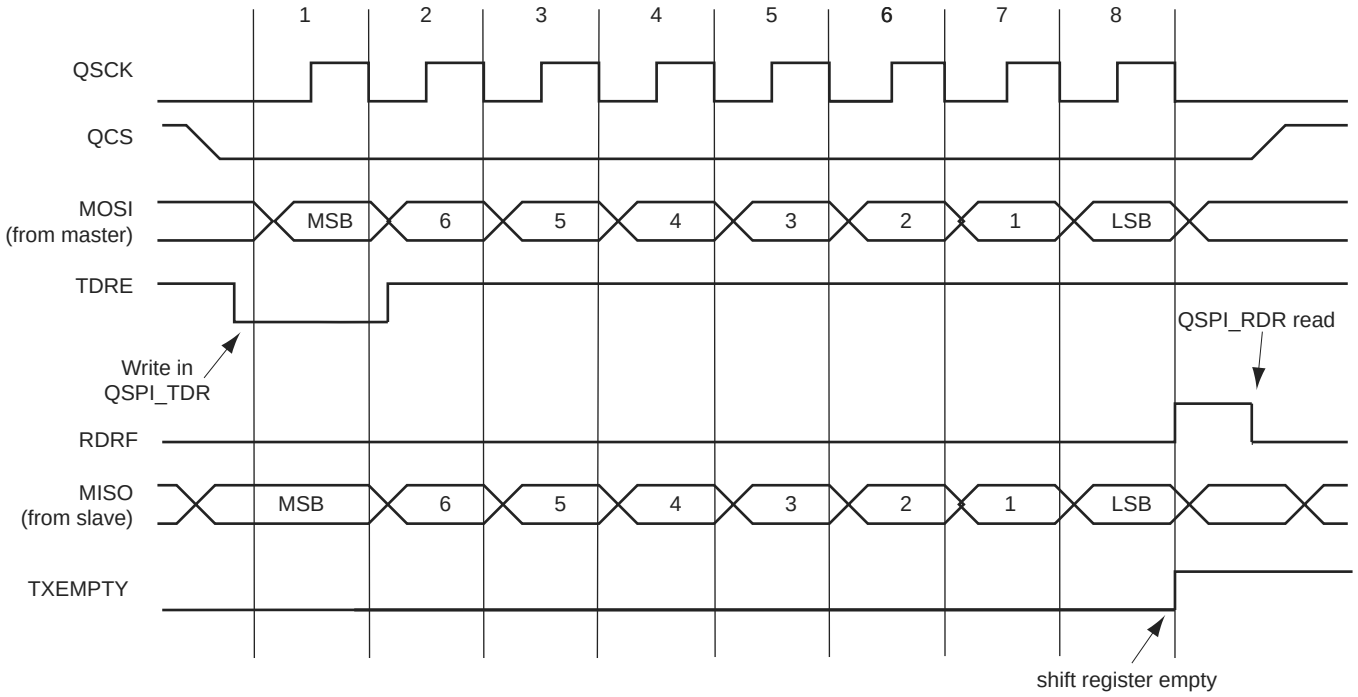
RXFFF: RXFFF Interrupt Disable

RXFTHF: RXFTHF Interrupt Disable

TXFPTEF: TXFPTEF Interrupt Disable

RXFPTEF: RXFPTEF Interrupt Disable

Figure 50-7: Status Register Flags Behavior



50.6.4.4 Peripheral Deselection without DMA

During a transfer of more than one data on a Chip Select without the DMA, the QSPI_TDR is loaded by the processor and the flag TDRE rises as soon as the content of the QSPI_TDR is transferred into the internal shift register. When this flag is detected high, the QSPI_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. Depending on the application software handling the QSPI_SR flags (by interrupt or polling method) or servicing other interrupts or other tasks, the processor may not reload the QSPI_TDR in time to keep the chip select active (low). A null Delay Between Consecutive Transfer (DLYBCT) value in the QSPI_MR gives even less time for the processor to reload the QSPI_TDR. With some SPI slave peripherals, requiring the chip select line to remain active (low) during a full set of transfers may lead to communication errors.

To facilitate interfacing with such devices, QSPI_MR.CSMODE may be configured to ‘1’. This allows the chip select lines to remain in their current state (low = active) until the end of transfer is indicated by the Last Transfer (LASTXFER) bit in the Control register (QSPI_CR). Even if the QSPI_TDR is not reloaded, the chip select remains active. To have the chip select line rise at the end of the last data transfer, QSPI_CR.LASTXFER must be written to ‘1’ at the same time or after writing the last data to transmit into the QSPI_TDR.

50.6.4.5 Peripheral Deselection with DMA

When the DMA Controller is used, the Chip Select line remains low during the transfer since the TDRE flag is managed by the DMA itself. Reloading the QSPI_TDR by the DMA is done as soon as the TDRE flag is set. In this case, writing QSPI_MR.CSMODE to ‘1’ may not be needed. However, when other DMA channels connected to other peripherals are also in use, the QSPI DMA could be delayed by another DMA with a higher priority on the bus. Having DMA buffers in slower memories like Flash memory or SDRAM compared to fast internal SRAM, may lengthen the reload time of the QSPI_TDR by the DMA as well. This means that the QSPI_TDR might not be reloaded in time to keep the chip select line low. In this case, the chip select line may toggle between data transfer and according to some SPI Slave devices, the communication might get lost. It may be necessary to configure QSPI_MR.CSMODE to ‘1’.

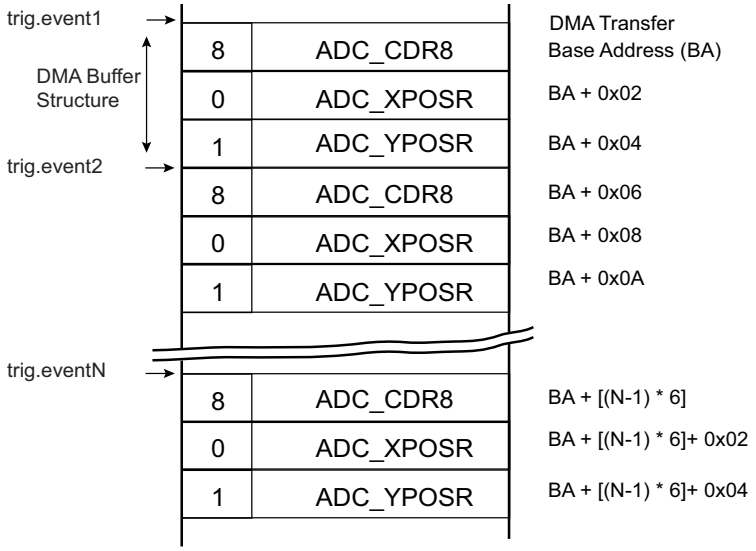
When QSPI_MR.CSMODE is configured to ‘0’, the QCS does not rise in all cases between two transfers on the same peripheral. During a transfer on a Chip Select, the flag TDRE rises as soon as the content of the QSPI_TDR is transferred into the internal shifter. When this flag is detected, the QSPI_TDR 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 transfers. This might lead to difficulties for interfacing with some serial peripherals requiring the chip select to be deasserted after each transfer. To facilitate interfacing with such devices, the QSPI_MR may be configured with QSPI_MR.CSMODE at ‘2’.

65.2 Embedded Characteristics

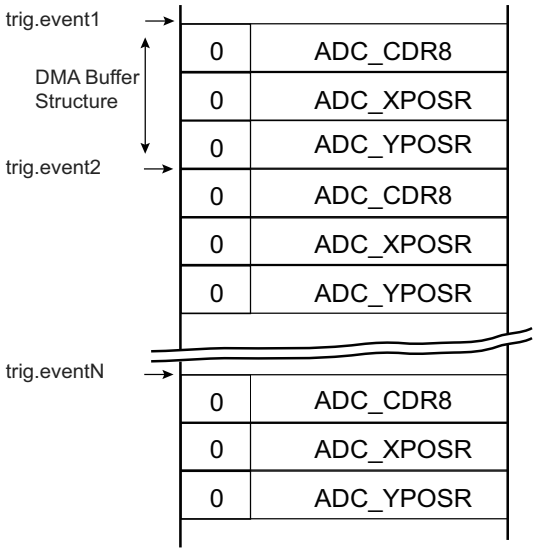
- 12-bit Resolution with Enhanced Mode up to 14 bits
- 1 Msps Conversion Rate
- Digital Averaging Function providing Enhanced Resolution Mode up to 14 bits
- Wide Range of Power Supply Operation
- Selectable Single-Ended or Differential Input Voltage
- Digital correction of offset and gain errors
- Resistive 4-wire and 5-wire Touchscreen Controller
  - Position and Pressure Measurement for 4-wire Screens
  - Position Measurement for 5-wire Screens
  - Average of Up to 8 Measures for Noise Filtering
- Programmable Pen Detection Sensitivity
- Integrated Multiplexer Offering Up to 12 Independent Analog Inputs
- Individual Enable and Disable of Each Channel
- Hardware or Software Trigger from:
  - External Trigger Pin
  - Timer Counter Outputs (Corresponding TIOA Trigger)
  - ADC Internal Trigger Counter
  - Trigger on Pen Contact Detection
  - PWM Event Line
- Drive of PWM Fault Input
- DMA Support
- Two Sleep Modes (Automatic Wakeup on Trigger)
  - Lowest Power Consumption (Voltage Reference OFF Between Conversions)
  - Fast Wakeup Time Response on Trigger Event (Voltage Reference ON Between Conversions)
- Channel Sequence Customizing
- Automatic Window Comparison of Converted Values
- Asynchronous Partial Wakeup (SleepWalking) on external trigger
- Register Write Protection

Figure 65-23: Buffer Structure When Classic ADC and Touchscreen Channels are Interleaved

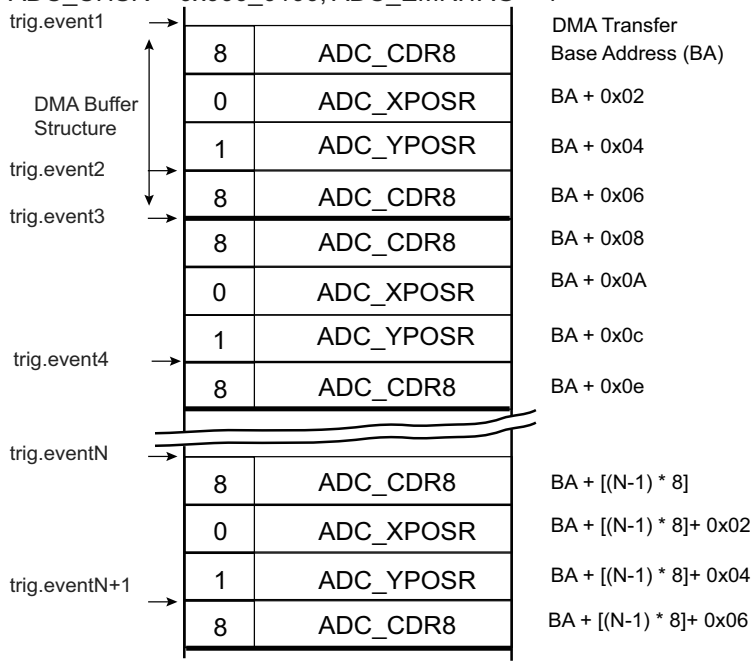
Assuming ADC_TSMR.TSMOD = 1
ADC_TSMR.TSAV = ADC_TSMR.TSFREQ = 0
ADC_CHSR = 0x000_0100, ADC_EMR.TAG = 1



Assuming ADC_TSMR.TSMOD = 1
ADC_TSMR.TSAV = ADC_TSMR.TSFREQ = 0
ADC_CHSR = 0x000_0100, ADC_EMR.TAG = 0



Assuming ADC_TSMR.TSMOD = 1
ADC_TSMR.TSAV = 0, ADC_TSMR.TSFREQ = 1
ADC_CHSR = 0x000_0100, ADC_EMR.TAG = 1



Assuming ADC_TSMR.TSMOD = 1
ADC_TSMR.TSAV = 1, ADC_TSMR.TSFREQ = 1
ADC_CHSR = 0x000_0100, ADC_EMR.TAG = 1