Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	ARM® Cortex®-M7
Core Size	32-Bit Single-Core
Speed	300MHz
Connectivity	CANbus, Ethernet, I²C, IrDA, LINbus, MMC/SD/SDIO, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	75
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 10x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TFBGA
Supplier Device Package	100-TFBGA (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsame70n19b-cnt

21. Chip Identifier (CHIPID)

21.1 Description

Chip Identifier (CHIPID) registers are used to recognize the device and its revision. These registers provide the sizes and types of the on-chip memories, as well as the set of embedded peripherals.

Two CHIPID registers are embedded: Chip ID Register (CHIPID_CIDR) and Chip ID Extension Register (CHIPID_EXID). Both registers contain a hard-wired value that is read-only.

The CHIPID_CIDR register contains the following fields:

- VERSION: Identifies the revision of the silicon
- EPROC: Indicates the embedded ARM processor
- NVPTYP and NVPSIZ: Identify the type of embedded non-volatile memory and the size
- SRAMSIZ: Indicates the size of the embedded SRAM
- ARCH: Identifies the set of embedded peripherals
- EXT: Shows the use of the extension identifier register

The CHIPID_EXID register is device-dependent and reads 0 if CHIPID_CIDR.EXT = 0.

21.2 Embedded Characteristics

- Chip ID Registers
  - Identification of the Device Revision, Sizes of the Embedded Memories, Set of Peripherals, Embedded Processor

Table 21-1. SAM E70SAM S70SAM V71SAM V70 Chip ID Registers

Chip Name	CHIPID_CIDR (see Notes 1 and 2)	CHIPID_EXID
SAME70Q21	0xA102_0E0x	0x00000002
SAME70Q20	0xA102_0C0x	0x00000002
SAME70Q19	0xA10D_0A0x	0x00000002
SAME70N21	0xA102_0E0x	0x00000001
SAME70N20	0xA102_0C0x	0x00000001
SAME70N19	0xA10D_0A0x	0x00000001
SAME70J21	0xA102_0E0x	0x00000000
SAME70J20	0xA102_0C0x	0x00000000
SAME70J19	0xA10D_0A0x	0x00000000
SAMS70Q21	0xA112_0E0x	0x00000002
SAMS70Q20	0xA112_0C0x	0x00000002
SAMS70Q19	0xA11D_0A0x	0x00000002

Bit	31	30	29	28	27	26	25	24
	FKEY[7:0]							
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
	FARG[15:8]							
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
	FARG[7:0]							
Access								
Reset								

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

Bits 31:24 – FKEY[7:0] Flash Writing Protection Key

Value	Name	Description
0x5A	PASSWD	The 0x5A value enables the command defined by the bits of the register. If the field is written with a different value, the write is not performed and no action is started.

Bits 23:8 – FARG[15:0] Flash Command Argument

Bits 7:0 – FCMD[7:0] Flash Command

Value	Name	Description
0x00	GETD	Get Flash descriptor
0x01	WP	Write page
0x02	WPL	Write page and lock
0x03	EWP	Erase page and write page
0x04	EWPL	Erase page and write page then lock
0x05	EA	Erase all
0x06	EPL	Erase plane
0x07	EPA	Erase pages
0x08	SLB	Set lock bit
0x09	CLB	Clear lock bit
0x0A	GLB	Get lock bit
0x0B	SGPB	Set GPNVM bit
0x0C	CGPB	Clear GPNVM bit
0x0D	GGPB	Get GPNVM bit
0x0E	STUI	Start read unique identifier
0x0F	SPUI	Stop read unique identifier
0x10	GCALB	Get CALIB bit
0x11	ES	Erase sector

31.20.17 PMC Interrupt Mask Register

Name: PMC_IMR
Offset: 0x006C
Reset: 0x00000000
Property: Read-only

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

0: No effect.

1: Enables the corresponding interrupt.

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
			XT32KERR			CFDEV	MOSCRCS	MOSCSELS
Access								
Reset			0			0	0	0

Bit	15	14	13	12	11	10	9	8
		PCKRDY6	PCKRDY5	PCKRDY4	PCKRDY3	PCKRDY2	PCKRDY1	PCKRDY0
Access								
Reset		0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
		LOCKU			MCKRDY		LOCKA	MOSCXTS
Access								
Reset		0			0		0	0

Bit 21 – XT32KERR 32.768 kHz Crystal Oscillator Error Interrupt Mask

Bit 18 – CFDEV Clock Failure Detector Event Interrupt Mask

Bit 17 – MOSCRCS Main RC Status Interrupt Mask

Bit 16 – MOSCSELS Main Clock Source Oscillator Selection Status Interrupt Mask

Bits 8, 9, 10, 11, 12, 13, 14 – PCKRDY Programmable Clock Ready x Interrupt Mask

Bit 6 – LOCKU UTMI PLL Lock Interrupt Mask

Bit 3 – MCKRDY Master Clock Ready Interrupt Mask

Bit 1 – LOCKA PLLA Lock Interrupt Mask

Bit 0 – MOSCXTS Main Crystal Oscillator Status Interrupt Mask

35.16.1.1 SMC Setup Register

Name: SMC_SETUP[0..3]
Offset: 0x00
Reset: 0
Property: R/W

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

Bit	31	30	29	28	27	26	25	24
			NCS_RD_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
			NRD_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
			NCS_WR_SETUP[5:0]					
Access								
Reset			0	0	0	0	0	0

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

Bits 29:24 – NCS_RD_SETUP[5:0] NCS Setup Length in READ Access
In read access, the NCS signal setup length is defined as:

NCS setup length = (128* NCS_RD_SETUP[5] + NCS_RD_SETUP[4:0]) clock cycles

Bits 21:16 – NRD_SETUP[5:0] NRD Setup Length
The NRD signal setup length is defined in clock cycles as:

NRD setup length = (128* NRD_SETUP[5] + NRD_SETUP[4:0]) clock cycles

Bits 13:8 – NCS_WR_SETUP[5:0] NCS Setup Length in WRITE Access
In write access, the NCS signal setup length is defined as:

NCS setup length = (128* NCS_WR_SETUP[5] + NCS_WR_SETUP[4:0]) clock cycles

Bits 5:0 – NWE_SETUP[5:0] NWE Setup Length
The NWE signal setup length is defined as:

NWE setup length = (128* NWE_SETUP[5] + NWE_SETUP[4:0]) clock cycles

36.6 Linked List Descriptor Operation

36.6.1 Linked List Descriptor View

36.6.1.1 Channel Next Descriptor View 0–3 Structures

Table 36-2. Channel Next Descriptor View 0–3 Structures

Channel Next Descriptor	Offset	Structure member	Name
View 0 Structure	DSCR_ADDR+0x00	Next Descriptor Address Member	MBR_NDA
	DSCR_ADDR+0x04	Microblock Control Member	MBR_UBC
	DSCR_ADDR+0x08	Transfer Address Member	MBR_TA
View 1 Structure	DSCR_ADDR+0x00	Next Descriptor Address Member	MBR_NDA
	DSCR_ADDR+0x04	Microblock Control Member	MBR_UBC
	DSCR_ADDR+0x08	Source Address Member	MBR_SA
	DSCR_ADDR+0x0C	Destination Address Member	MBR_DA
View 2 Structure	DSCR_ADDR+0x00	Next Descriptor Address Member	MBR_NDA
	DSCR_ADDR+0x04	Microblock Control Member	MBR_UBC
	DSCR_ADDR+0x08	Source Address Member	MBR_SA
	DSCR_ADDR+0x0C	Destination Address Member	MBR_DA
	DSCR_ADDR+0x10	Configuration Register	MBR_CFG
View 3 Structure	DSCR_ADDR+0x00	Next Descriptor Address Member	MBR_NDA
	DSCR_ADDR+0x04	Microblock Control Member	MBR_UBC
	DSCR_ADDR+0x08	Source Address Member	MBR_SA
	DSCR_ADDR+0x0C	Destination Address Member	MBR_DA
	DSCR_ADDR+0x10	Configuration Member	MBR_CFG
	DSCR_ADDR+0x14	Block Control Member	MBR_BC
	DSCR_ADDR+0x18	Data Stride Member	MBR_DS
	DSCR_ADDR+0x1C	Source Microblock Stride Member	MBR_SUS
	DSCR_ADDR+0x20	Destination Microblock Stride Member	MBR_DUS

36.6.2 Descriptor Structure Members Description

2. Allocate an area 8N bytes for the transmit buffer descriptor list in system memory and create N entries in this list. Mark all entries in this list as owned by GMAC, i.e., bit 31 of word 1 set to 0.
3. Mark the last descriptor in the queue with the wrap bit (bit 30 in word 1 set to 1).
4. Write address of transmit buffer descriptor list and control information to GMAC register transmit buffer queue pointer.
5. The transmit circuits can then be enabled by writing to the Network Control register.

Note: The queue pointers must be initialized and point to USED descriptors for all queues including those not intended for use.

38.7.1.4 Address Matching

The GMAC Hash register pair and the four Specific Address register pairs must be written with the required values. Each register pair comprises of a bottom register and top register, with the bottom register being written first. The address matching is disabled for a particular register pair after the bottom register has been written and re-enabled when the top register is written. Each register pair may be written at any time, regardless of whether the receive circuits are enabled or disabled.

As an example, to set Specific Address register 1 to recognize destination address 21:43:65:87:A9:CB, the following values are written to Specific Address register 1 bottom and Specific Address register 1 top:

- Specific Address register 1 bottom bits 31:0 (0x98): 0x8765_4321.
- Specific Address register 1 top bits 31:0 (0x9C): 0x0000_CBA9.
  Note: The address matching is the first level of filtering. If there is a match, the screeners are the next level of filtering for routing the data to the appropriate queue. See Priority Queueing in the DMA for more details.

38.7.1.5 PHY Maintenance

The PHY Maintenance register is implemented as a shift register. Writing to the register starts a shift operation which is signalled as complete when bit two is set in the Network Status register (about 2000 MCK cycles later when bits 18:16 are set to 010 in the Network Configuration register). An interrupt is generated as this bit is set.

During this time, the MSB of the register is output on the MDIO pin and the LSB updated from the MDIO pin with each Management Data Clock (MDC) cycle. This causes the transmission of a PHY management frame on MDIO. See section 22.2.4.5 of the IEEE 802.3 standard.

Reading during the shift operation will return the current contents of the shift register. At the end of the management operation the bits will have shifted back to their original locations. For a read operation the data bits are updated with data read from the PHY. It is important to write the correct values to the register to ensure a valid PHY management frame is produced.

The Management Data Clock (MDC) should not toggle faster than 2.5 MHz (minimum period of 400 ns), as defined by the IEEE 802.3 standard. MDC is generated by dividing down MCK. Three bits in the Network Configuration register determine by how much MCK should be divided to produce MDC.

38.7.1.6 Interrupts

There are 18 interrupt conditions that are detected within the GMAC. The conditions are ORed to make multiple interrupts. Depending on the overall system design this may be passed through a further level of interrupt collection (interrupt controller). On receipt of the interrupt signal, the CPU enters the interrupt handler. Refer to the device interrupt controller documentation to identify that it is the GMAC that is generating the interrupt. To ascertain which interrupt, read the Interrupt Status register. Note that in the default configuration this register will clear itself after being read, though this may be configured to be write-one-to-clear if desired.

Offset	Name	Bit Pos.								
		31:24								
0xF0	GMAC_PEFTSH	7:0	RUD[7:0]							
		15:8	RUD[15:8]							
		23:16								
		31:24								
0xF4	GMAC_PEFRSH	7:0	RUD[7:0]							
		15:8	RUD[15:8]							
		23:16								
		31:24								
0xF8 ... 0xFF	Reserved									
0x0100	GMAC_OTLO	7:0	TXO[7:0]							
		15:8	TXO[15:8]							
		23:16	TXO[23:16]							
		31:24	TXO[31:24]							
0x0104	GMAC_OTHI	7:0	TXO[7:0]							
		15:8	TXO[15:8]							
		23:16								
		31:24								
0x0108	GMAC_FT	7:0	FTX[7:0]							
		15:8	FTX[15:8]							
		23:16	FTX[23:16]							
		31:24	FTX[31:24]							
0x010C	GMAC_BCFT	7:0	BFTX[7:0]							
		15:8	BFTX[15:8]							
		23:16	BFTX[23:16]							
		31:24	BFTX[31:24]							
0x0110	GMAC_MFT	7:0	MFTX[7:0]							
		15:8	MFTX[15:8]							
		23:16	MFTX[23:16]							
		31:24	MFTX[31:24]							
0x0114	GMAC_PFT	7:0	PFTX[7:0]							
		15:8	PFTX[15:8]							
		23:16								
		31:24								
0x0118	GMAC_BFT64	7:0	NFTX[7:0]							
		15:8	NFTX[15:8]							
		23:16	NFTX[23:16]							
		31:24	NFTX[31:24]							
0x011C	GMAC_TBFT127	7:0	NFTX[7:0]							
		15:8	NFTX[15:8]							
		23:16	NFTX[23:16]							
		31:24	NFTX[31:24]							
0x0120	GMAC_TBFT255	7:0	NFTX[7:0]							
		15:8	NFTX[15:8]							

Value	Description
0	Normal operation
1	All received frames' CRC is replaced with a time stamp.

Bit 12 – TXZQPF Transmit Zero Quantum Pause Frame
Writing a '1' to this bit causes a pause frame with zero quantum to be transmitted.

Writing a '0' to this bit has no effect.

Bit 11 – TXPF Transmit Pause Frame
Writing one to this bit causes a pause frame to be transmitted.

Writing a '0' to this bit has no effect.

Bit 10 – THALT Transmit Halt
Writing a '1' to this bit halts transmission as soon as any ongoing frame transmission ends.

Writing a '0' to this bit has no effect.

Bit 9 – TSTART Start Transmission
Writing a '1' to this bit starts transmission.

Writing a '0' to this bit has no effect.

Bit 8 – BP Back Pressure
In 10M or 100M half duplex mode, writing a '1' to this bit forces collisions on all received frames. Ignored in gigabit half duplex mode.

Value	Description
0	Frame collisions are not forced.
1	Frame collisions are forced in 10M and 100M half duplex mode.

Bit 7 – WESTAT Write Enable for Statistics Registers
Writing a '1' to this bit makes the statistics registers writable for functional test purposes.

Value	Description
0	Statistics Registers are write-protected.
1	Statistics Registers are write-enabled.

Bit 6 – INCSTAT Increment Statistics Registers
Writing a '1' to this bit increments all Statistics Registers by one for test purposes.

Writing a '0' to this bit has no effect.

This bit will always read '0'.

Bit 5 – CLRSTAT Clear Statistics Registers
Writing a '1' to this bit clears the Statistics Registers.

Writing a '0' to this bit has no effect.

This bit will always read '0'.

Bit 4 – MPE Management Port Enable
Writing a '1' to this bit enables the Management Port.

38.8.86 GMAC 1588 Timer Nanoseconds Register

Name: GMAC_TN
Offset: 0x1D4
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
			TNS[29:24]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
	TNS[23:16]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
	TNS[15:8]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	TNS[7:0]							
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bits 29:0 – TNS[29:0] Timer Count in Nanoseconds
This register is writable. It can also be adjusted by writes to the IEEE 1588 Timer Adjust Register. It increments by the value of the IEEE 1588 Timer Increment Register each clock cycle.

39.6.42 Host Pipe Register

Name: USBHS_HSTPIP
Offset: 0x0041C
Reset: 0x00000000
Property: Read/Write

Bit	31	30	29	28	27	26	25	24
								PRST8
Access								
Reset								0

Bit	23	22	21	20	19	18	17	16
	PRST7	PRST6	PRST5	PRST4	PRST3	PRST2	PRST1	PRST0
Access								
Reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
								PEN8
Access								
Reset								0

Bit	7	6	5	4	3	2	1	0
	PEN7	PEN6	PEN5	PEN4	PEN3	PEN2	PEN1	PEN0
Access								
Reset	0	0	0	0	0	0	0	0

Bits 16, 17, 18, 19, 20, 21, 22, 23, 24 – PRST Pipe x Reset

Value	Description
0	Completes the reset operation and allows to start using the FIFO.
1	Resets the Pipe x FIFO. This resets the pipe x registers (USBHS_HSTPIPCFGx, USBHS_HSTPIPISRx, USBHS_HSTPIPIMRx), but not the pipe configuration (ALLOC, PBK, PSIZE, PTOKEN, PTYPE, PEPNUM, INTFRQ). The whole pipe mechanism (FIFO counter, reception, transmission, etc.) is reset, apart from the Data Toggle management. The pipe configuration remains active and the pipe is still enabled.

Bits 0, 1, 2, 3, 4, 5, 6, 7, 8 – PEN Pipe x Enable

Value	Description
0	Disables Pipe x, which forces the Pipe x state to inactive and resets the pipe x registers (USBHS_HSTPIPCFGx, USBHS_HSTPIPISRx, USBHS_HSTPIPIMRx), but not the pipe configuration (USBHS_HSTPIPCFGx.ALLOC, USBHS_HSTPIPCFGx.PBK, USBHS_HSTPIPCFGx.PSIZE).
1	Enables Pipe x.

Source Clock (MHz)	Expected Baud Rate (bit/s)	Calculation Result	CD	Actual Baud Rate (bit/s)	Error
8,000,000	38,400	13.02	13	38,461.54	0.16%
12,000,000	38,400	19.53	20	37,500.00	2.40%
12,288,000	38,400	20.00	20	38,400.00	0.00%
14,318,180	38,400	23.30	23	38,908.10	1.31%
14,745,600	38,400	24.00	24	38,400.00	0.00%
18,432,000	38,400	30.00	30	38,400.00	0.00%
24,000,000	38,400	39.06	39	38,461.54	0.16%
24,576,000	38,400	40.00	40	38,400.00	0.00%
25,000,000	38,400	40.69	40	38,109.76	0.76%
32,000,000	38,400	52.08	52	38,461.54	0.16%
32,768,000	38,400	53.33	53	38,641.51	0.63%
33,000,000	38,400	53.71	54	38,194.44	0.54%
40,000,000	38,400	65.10	65	38,461.54	0.16%
50,000,000	38,400	81.38	81	38,580.25	0.47%
60,000,000	38,400	97.66	98	38,265.31	0.35%
70,000,000	38,400	113.93	114	38,377.19	0.06%

In this example, the baud rate is calculated with the following formula:

$$\text{Baud Rate} = \text{Selected Clock}/\text{CD} \times 16$$

The baud rate error is calculated with the following formula. It is not recommended to work with an error higher than 5%.

$$\text{Error} = 1 - \left(\frac{\text{Expected Baud Rate}}{\text{Actual Baud Rate}}\right)$$

46.6.1.2 Fractional Baud Rate in Asynchronous Mode

The baud rate generator is subject to the following limitation: the output frequency changes only by integer multiples of the reference frequency. An approach to this problem is to integrate a fractional N clock generator that has a high resolution. The generator architecture is modified to obtain baud rate changes by a fraction of the reference source clock. This fractional part is programmed using US_BRGR.FP. If FP is not 0, the fractional part is activated. The resolution is one-eighth of the clock divider. The fractional baud rate is calculated using the following formula:

$$\text{Baud Rate} = \frac{\text{Selected Clock}}{\left(8(2 - \text{OVER})\left(\text{CD} + \frac{\text{FP}}{8}\right)\right)}$$

The modified architecture is presented in the following figure.

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

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

1: Dual ID filter for EF1ID or EF2ID

2: Classic filter: EF1ID = filter, EF2ID = mask

3: Range filter from EF1ID to EF2ID (EF2ID ≥ EF1ID), MCAN_XIDAM mask not applied

• F1 Bits 28:0 EFID2[28:0]: Extended Filter ID 2

This field has a different meaning depending on the configuration of EFEC:

• EFEC = “001”...“110”–Second ID of extended ID filter element

• EFEC = “111”–Filter for Rx Buffers or for debug messages

EFID2[10:9] decides whether the received message is stored into an Rx Buffer or treated as message A, B, or C of the debug message sequence.

0: Store message in an Rx buffer

1: Debug Message A

2: Debug Message B

3: Debug Message C

EFID2[5:0] defines the index of the dedicated Rx Buffer element to which a matching message is stored.

49.5.8 Hardware Reset Description

After hardware reset, the registers of the MCAN hold the reset values listed in the register descriptions. Additionally the Bus_Off state is reset and the output CANTX is set to recessive (HIGH). The value 0x0001 (MCAN_CCCR.INIT = ‘1’) in the CC Control register enables software initialization. The MCAN does not influence the CAN bus until the processor resets MCAN_CCCR.INIT to ‘0’.

49.5.9 Access to Reserved Register Addresses

In case the application software accesses one of the reserved addresses in the MCAN register map (read or write access), interrupt flag MCAN_IR.ARA is set and, if enabled, the selected interrupt line is risen.

49.6.1 MCAN Core Release Register

Name: MCAN_CREL
Offset: 0x00
Reset: 0xrrrddddd
Property: Read-only

Due to clock domain crossing, there is a delay between when a register bit or field is written and when the related status register bits are updated.

Bit	31	30	29	28	27	26	25	24
	REL[3:0]				STEP[3:0]			
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bit	23	22	21	20	19	18	17	16
	SUBSTEP[3:0]				YEAR[3:0]			
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bit	15	14	13	12	11	10	9	8
	MON[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bit	7	6	5	4	3	2	1	0
	DAY[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	x	x	x	x	x	x	x	x

Bits 31:28 – REL[3:0] Core Release
One digit, BCD-coded.

Bits 27:24 – STEP[3:0] Step of Core Release
One digit, BCD-coded.

Bits 23:20 – SUBSTEP[3:0] Sub-step of Core Release
One digit, BCD-coded.

Bits 19:16 – YEAR[3:0] Timestamp Year
One digit, BCD-coded. This field is set by generic parameter on MCAN synthesis.

Bits 15:8 – MON[7:0] Timestamp Month
Two digits, BCD-coded. This field is set by generic parameter on MCAN synthesis.

Bits 7:0 – DAY[7:0] Timestamp Day
Two digits, BCD-coded. This field is set by generic parameter on MCAN synthesis.

49.6.14 MCAN Protocol Status Register

Name: MCAN_PSR
Offset: 0x44
Reset: 0x00000707
Property: Read-only

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
		TDCV[6:0]						
Access		R	R	R	R	R	R	R
Reset		0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
		PXE	RFDF	RBRS	RESI	DLEC[2:0]		
Access								
Reset		0	0	0	0	1	1	1

Bit	7	6	5	4	3	2	1	0
	BO	EW	EP	ACT[1:0]		LEC[2:0]		
Access	R	R	R	R	R			
Reset	0	0	0	0	0	1	1	1

Bits 22:16 – TDCV[6:0] Transmitter Delay Compensation Value
0 to 127: Position of the secondary sample point, in CAN core clock periods, defined by the sum of the measured delay from CANTX to CANRX and MCAN_TDCR.TDCO.

Bit 14 – PXE Protocol Exception Event (cleared on read)

Value	Description
0	No protocol exception event occurred since last read access
1	Protocol exception event occurred

Bit 13 – RFDF Received a CAN FD Message (cleared on read)
This bit is set independently from acceptance filtering.

Value	Description
0	Since this bit was reset by the CPU, no CAN FD message has been received
1	Message in CAN FD format with FDF flag set has been received

Bit 12 – RBRS BRS Flag of Last Received CAN FD Message (cleared on read)
This bit is set together with RFDF, independently from acceptance filtering.

Bit 7 – ETRGS External Trigger Status (cleared on read)

Value	Description
0	External trigger has not occurred since the last read of the Status Register.
1	External trigger has occurred since the last read of the Status Register.

Bit 6 – LDRBS RB Loading Status (cleared on read)

Value	Description
0	RB Load has not occurred since the last read of the Status Register or TC_CMRx.WAVE = 1.
1	RB Load has occurred since the last read of the Status Register, if TC_CMRx.WAVE = 0.

Bit 5 – LDRAS RA Loading Status (cleared on read)

Value	Description
0	RA Load has not occurred since the last read of the Status Register or TC_CMRx.WAVE = 1.
1	RA Load has occurred since the last read of the Status Register, if TC_CMRx.WAVE = 0.

Bit 4 – CPCS RC Compare Status (cleared on read)

Value	Description
0	RC Compare has not occurred since the last read of the Status Register.
1	RC Compare has occurred since the last read of the Status Register.

Bit 3 – CPBS RB Compare Status (cleared on read)

Value	Description
0	RB Compare has not occurred since the last read of the Status Register or TC_CMRx.WAVE = 0.
1	RB Compare has occurred since the last read of the Status Register, if TC_CMRx.WAVE = 1.

Bit 2 – CPAS RA Compare Status (cleared on read)

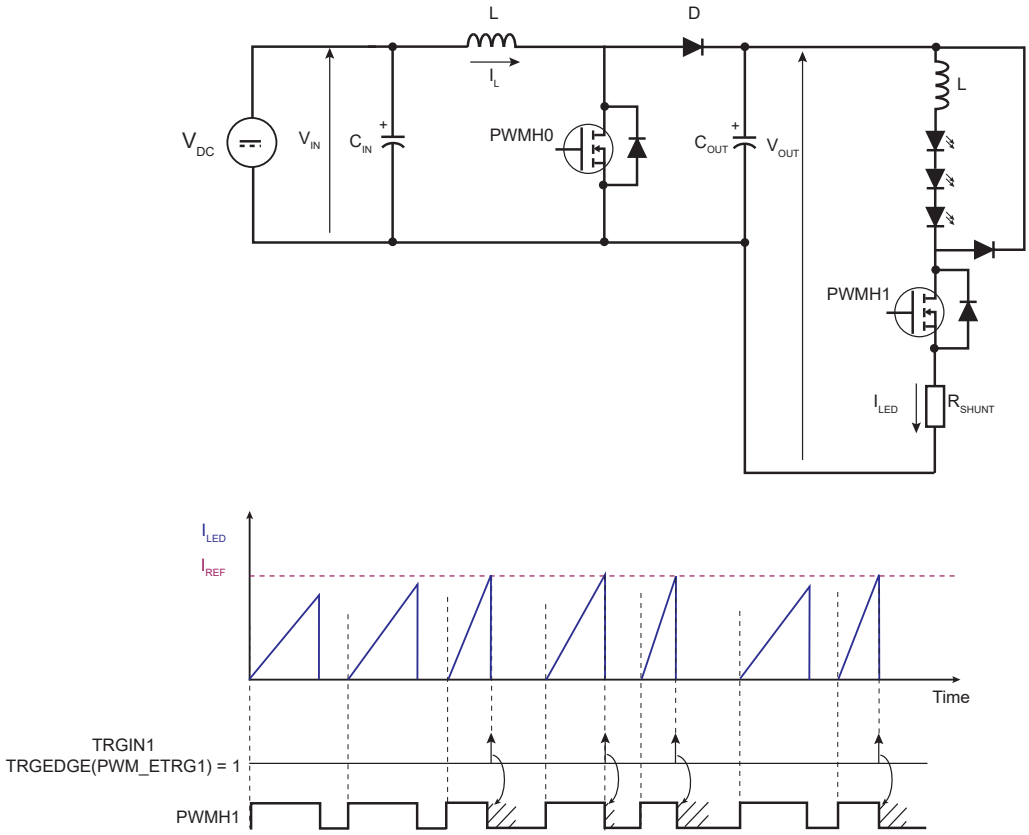
Value	Description
0	RA Compare has not occurred since the last read of the Status Register or TC_CMRx.WAVE = 0.
1	RA Compare has occurred since the last read of the Status Register, if TC_CMRx.WAVE = 1.

Bit 1 – LOVRS Load Overrun Status (cleared on read)

Value	Description
0	Load overrun has not occurred since the last read of the Status Register or TC_CMRx.WAVE = 1.
1	RA or RB have been loaded at least twice without any read of the corresponding register since the last read of the Status Register, if TC_CMRx.WAVE = 0.

Bit 0 – COVFS Counter Overflow Status (cleared on read)

Figure 51-32. Cycle-By-Cycle Duty Mode: LED String Control



51.6.5.4 Leading-Edge Blanking (LEB)

PWM channels 1 and 2 support leading-edge blanking. Leading-edge blanking masks the external trigger input when a transient occurs on the corresponding PWM output. It masks potential spurious external events due to power transistor switching.

The blanking delay on each external trigger input is configured by programming the LEBDELAYx in the PWM Leading-Edge Blanking Register.

The LEB can be enabled on both the rising and the falling edges for the PWMH and PWML outputs through the bits PWMLFEN, PWMLREN, PWMHFEN, PWMHREN.

Any event on the PWMEXTRGx input which occurs during the blanking time is ignored.

Value	Description
0	The comparison x is disabled and can not match.
1	The comparison x is enabled and can match.

54.4 Signal Description

Table 54-1. ACC Signal Description

Pin Name	Description	Type
AFE0_AD[5:0]	External analog data inputs	Input
AFE1_AD[1:0]		
TS	On-chip temperature sensor	Input
VREFP	AFE and DAC voltage reference	Input
DAC0, DAC1	On-chip DAC outputs	Input

54.5 Product Dependencies

54.5.1 I/O Lines

The analog input pins are multiplexed with digital functions (PIO) on the IO line. By writing the SELMINUS and SELPLUS fields in the ACC Mode Register (ACC_MR), the associated IO lines are set to Analog mode.

54.5.2 Power Management

The ACC is clocked through the Power Management Controller (PMC), thus the programmer must first configure the PMC to enable the ACC clock.

Note that the voltage regulator must be activated to use the analog comparator.

54.5.3 Interrupt Sources

The ACC has an interrupt line connected to the Interrupt Controller (IC). In order to handle interrupts, the Interrupt Controller must be programmed before configuring the ACC.

54.5.4 Fault Output

The ACC has the FAULT output connected to the FAULT input of PWM. See Fault Mode and the implementation of the PWM in the product.

54.6 Functional Description

54.6.1 Description

The Analog Comparator Controller (ACC) controls the analog comparator settings and performs postprocessing of the analog comparator output.

When the analog comparator settings are modified, the output of the analog cell may be invalid. The ACC masks the output for the invalid period.

A comparison flag is triggered by an event on the output of the analog comparator and an interrupt is generated. The event on the analog comparator output can be selected among falling edge, rising edge or any edge.

The ACC registers are listed in the Register Summary.