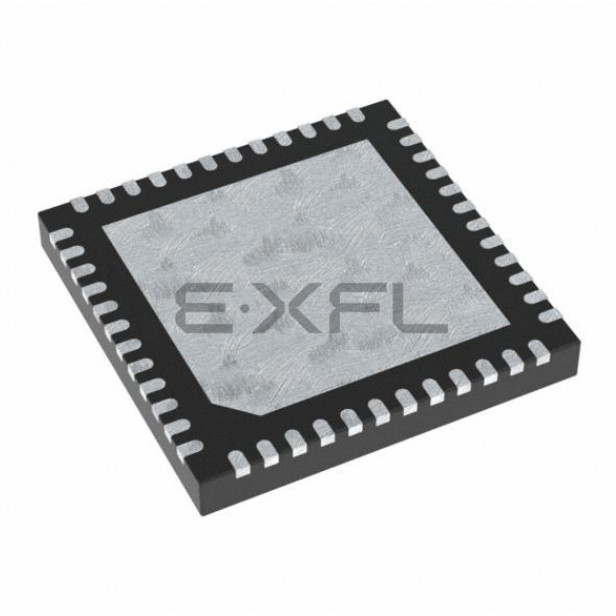


Welcome to E-XFL.COM

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®-M4F
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	EBI/EMI, I²C, IrDA, LINbus, MMC/SD, QSPI, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM
Number of I/O	37
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 3.63V
Data Converters	A/D 20x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd51g18a-mu

6.2.9 GPIO Clusters

6.2 Other Functions

6.2.1 Oscillator Pinout

The oscillators are not mapped to the normal PORT functions and their multiplexing is controlled by registers in the Oscillators Controller (OSCCTRL) and in the 32K Oscillators Controller (OSC32KCTRL).

Table 6-2. Oscillator Pinout

Oscillator	Supply	Signal	I/O pin
XOSC0	VDDIO	XIN	PA14
		XOUT	PA15
XOSC1	VDDIO	XIN	PB22
		XOUT	PB23
XOSC32K	VSWOUT	XIN32	PA00
		XOUT32	PA01

Note: To guarantee the XOSC32K behavior in crystal mode, PC00 must be static.

Table 6-3. XOSC32K Jitter Minimization

Package Pin Count	Steady Signal Recommended
128	PB00, PB01, PB02, PB03, PC00, PC01
100	PB00, PB01, PB02, PB03, PC00, PC01
120	PB00, PB01, PB02, PB03, PC00, PC01
64	PB00,PB01,PB02,PB03, PA02,PA03
48	PB02, PB03,PA02,PA03

6.2.2 Serial Wire Debug Interface Pinout

Only the SWCLK pin is mapped to the normal PORT functions. A debugger cold-plugging or hot-plugging detection will automatically switch the SWDIO port to the SWDIO function.

Table 6-4. Serial Wire Debug Interface Pinout

Signal	Supply	I/O pin
SWCLK	VDDIO	PA30
SWDIO	VDDIO	PA31

6.2.3 Trace Port Interface Unit Pinout

The Embedded Trace Module (ETM) is leaning on Trace Port Interface Unit (TPIU) to export data out of the system.

Module	Source	Line
	OVR 3	
	EVD 4..11	40
	OVR 4..11	
PAC - Peripheral Access Controller	ERR	41
RAM ECC	0	45
	1	
SERCOM0 - Serial Communication Interface 0(1)	0	46
	1	47
	2	48
	3	49
	4	
	5	
	6	
SERCOM1 - Serial Communication Interface 1(1)	0	50
	1	51
	2	52
	3	53
	4	
	5	
	6	
SERCOM2 - Serial Communication Interface 2(1)	0	54
	1	55
	2	56
	3	57
	4	
	5	
	6	
SERCOM3 - Serial Communication Interface 3(1)	0	58
	1	59
	2	60
	3	61
	4	

peripheral is reset. An interrupt flag is cleared by writing a '1' to the corresponding bit in the INTFLAG register. Each peripheral can have one interrupt request line per interrupt source or one common interrupt request line for all the interrupt sources.If the peripheral has one common interrupt request line for all the interrupt sources, the user must read the INTFLAG register to determine which interrupt condition is present.

Related Links

18. PM – Power Manager

10.2.1 Overview

15.6.5 Events

Not applicable.

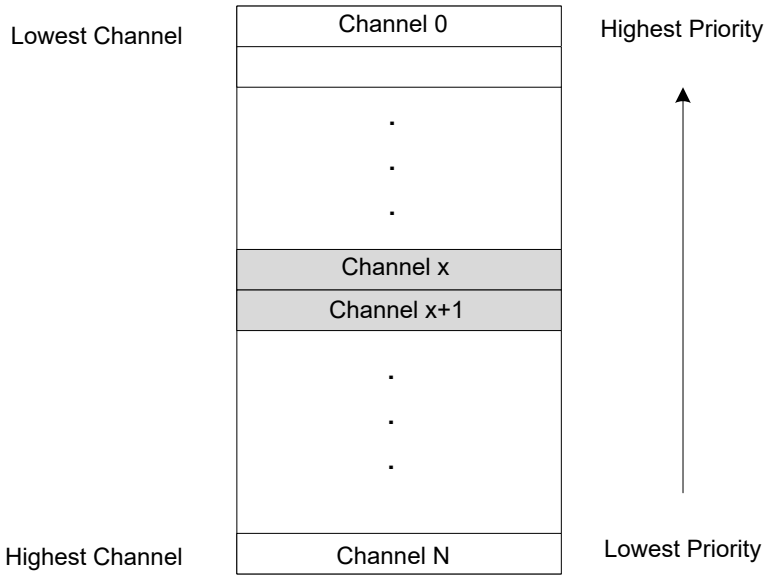
15.6.6 Sleep Mode Operation

In IDLE sleep mode, the MCLK is still running on the selected main clock.

In STANDBY sleep mode, the MCLK is frozen if no synchronous clock is required.

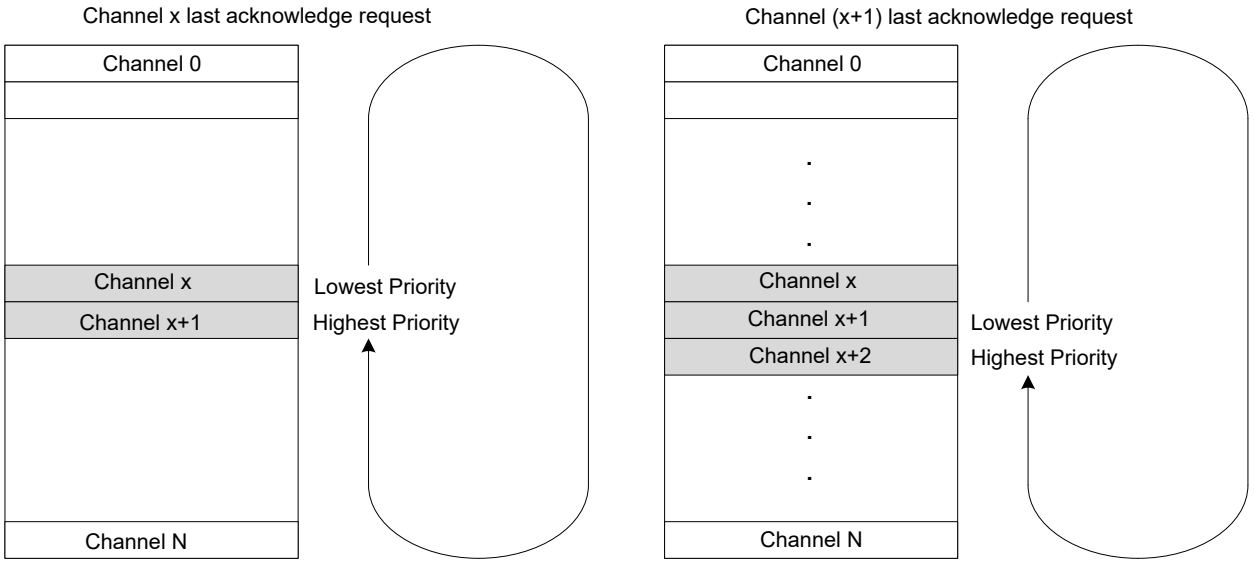
Within each priority level, the DMAC's arbiter can be configured to prioritize statically or dynamically. For the arbiter to perform static arbitration within a priority level, the Level X Round-Robin Scheduling Enable bit in the Priority Control x register (PRICTRL0.RRLVLENx) has to be written to '0'. When static arbitration is enabled (PRICTRL0.RRLVLENx is '0'), the arbiter will prioritize a low channel number over a high channel number as shown in Static Priority Scheduling. When using the static scheme, there is a risk of high channel numbers never being granted access as the active channel. This can be avoided using a dynamic arbitration scheme.

Figure 22-5. Static Priority Scheduling



The dynamic arbitration scheme in the DMAC is round-robin. Round-robin arbitration is enabled by writing PRICTRL0.RRLVLEN to '1', for a given priority level x. With the round-robin scheme, the channel number of the last channel being granted access will have the lowest priority the next time the arbiter has to grant access to a channel within the same priority level, as shown in Figure 22-6. The channel number of the last channel being granted access as the active channel is stored in the Level x Channel Priority Number bit group in the Priority Control 0 register (PRICTRL0.LVLPRIx) for the corresponding priority level.

Figure 22-6. Dynamic (Round-Robin) Priority Scheduling



24.9.1 GMAC Network Control Register

Name: NCR
Offset: 0x000
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					LPI	FNP	TXPBPF	ENPBPR
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
	SRTSM			TXZQPF	TXPF	THALT	TSTART	BP
Access	R/W			R/W	R/W	R/W	R/W	R/W
Reset	0			0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
	WESTAT	INCSTAT	CLRSTAT	MPE	TXEN	RXEN	LBL	
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	

Bit 19 – LPI Low Power Idle Enable
Writing a '1' to this bit will enable low power idle (LPI) transmission, immediately transmitted on txd and tx_er.

Bit 18 – FNP Flush Next Packet
Writing a '1' to this bit will flush the next packet from the external RX DPRAM. Flushing the next packet will only take effect if the DMA is not currently writing a packet already stored in the DPRAM to memory.

Bit 17 – TXPBPF Transmit PFC Priority-based Pause Frame
Takes the values stored in the Transmit PFC Pause Register.

Bit 16 – ENPBPR Enable PFC Priority-based Pause Reception
Writing a '1' to this bit enables PFC Priority Based Pause Reception capabilities, enabling PFC negotiation and recognition of priority-based pause frames.

Value	Description
0	Normal operation
1	PFC Priority-based Pause frames are recognized

24.9.25 GMAC Wake on LAN Register

Name: WOL
Offset: 0x0B8
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
					MTI	SA1	ARP	MAG
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	15	14	13	12	11	10	9	8
	IP[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
	IP[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

Bit 19 – MTI Multicast Hash Event Enable

Value	Description
0	Wake on LAN multicast hash Event disabled
1	Wake on LAN multicast hash Event enabled

Bit 18 – SA1 Specific Address Register 1 Event Enable

Value	Description
0	Wake on Specific Address Register 1 Event disabled
1	Wake on Specific Address Register 1 Event enabled

Bit 17 – ARP ARP Request Event Enable

Value	Description
0	Wake on LAN ARP request Event disabled
1	Wake on LAN ARP request Event enabled

Bit 16 – MAG Magic Packet Event Enable

24.9.34 GMAC PTP Event Frame Transmitted Seconds High Register

Name: EFTSH
Offset: 0x0E8
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
	RUD[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
	RUD[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – RUD[15:0] Register Update
The register is updated with the value that the IEEE 1588 timer seconds register held when the SFD of a PTP transmit primary event crosses the MII interface. An interrupt is issued when the register is updated.

24.9.89 GMAC 1588 Timer Adjust Register

Name: TA
Offset: 0x1D8
Reset: 0x00000000
Property: Write-Only

Bit	31	30	29	28	27	26	25	24
	ADJ		ITDT[29:24]					
Access	W		W	W	W	W	W	W
Reset	0		0	0	0	0	0	0

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

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

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

Bit 31 – ADJ Adjust 1588 Timer
Write as '1' to subtract from the 1588 timer. Write as '0' to add to it.

Bits 29:0 – ITDT[29:0] Increment/Decrement
The number of nanoseconds to increment or decrement the IEEE 1588 Timer Nanoseconds Register. If necessary, the IEEE 1588 Seconds Register will be incremented or decremented.

Offset	Name	Bit Pos.								
0x2D	CHINTENSET1	7:0							EVD	OVR
0x2E	CHINTFLAG1	7:0							EVD	OVR
0x2F	CHSTATUS1	7:0							BUSYCH	RDYUSR
0x30	CHANNEL2	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x34	CHINTENCLR2	7:0							EVD	OVR
0x35	CHINTENSET2	7:0							EVD	OVR
0x36	CHINTFLAG2	7:0							EVD	OVR
0x37	CHSTATUS2	7:0							BUSYCH	RDYUSR
0x38	CHANNEL3	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x3C	CHINTENCLR3	7:0							EVD	OVR
0x3D	CHINTENSET3	7:0							EVD	OVR
0x3E	CHINTFLAG3	7:0							EVD	OVR
0x3F	CHSTATUS3	7:0							BUSYCH	RDYUSR
0x40	CHANNEL4	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x44	CHINTENCLR4	7:0							EVD	OVR
0x45	CHINTENSET4	7:0							EVD	OVR
0x46	CHINTFLAG4	7:0							EVD	OVR
0x47	CHSTATUS4	7:0							BUSYCH	RDYUSR
0x48	CHANNEL5	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x4C	CHINTENCLR5	7:0							EVD	OVR
0x4D	CHINTENSET5	7:0							EVD	OVR
0x4E	CHINTFLAG5	7:0							EVD	OVR
0x4F	CHSTATUS5	7:0							BUSYCH	RDYUSR
0x50	CHANNEL6	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								
		31:24								
0x54	CHINTENCLR6	7:0							EVD	OVR
0x55	CHINTENSET6	7:0							EVD	OVR
0x56	CHINTFLAG6	7:0							EVD	OVR
0x57	CHSTATUS6	7:0							BUSYCH	RDYUSR
0x58	CHANNEL7	7:0	EVGEN[7:0]							
		15:8	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
		23:16								

37.8.2 Control B

Name: CTRLB
Offset: 0x04
Reset: 0x00000000
Property: PAC Write-Protection

Control B

Bit	31	30	29	28	27	26	25	24
	DLYCS[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

Bit	23	22	21	20	19	18	17	16
	DLYBCT[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

Bit	15	14	13	12	11	10	9	8
					DATALEN[3:0]			
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit	7	6	5	4	3	2	1	0
			CSMODE[1:0]		SMEMREG	WDRBT	LOOPEN	MODE
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 31:24 – DLYCS[7:0] Minimum Inactive CS Delay
This bit field defines the minimum delay between the inactivation and the activation of CS. The DLYCS time guarantees the slave minimum deselect time.

If DLYCS is 0x00, one CLK_QSPI_AHB period will be inserted by default.

Otherwise, the following equation determines the delay:

Bits 23:16 – DLYBCT[7:0] Delay Between Consecutive Transfers
This field defines the delay between two consecutive transfers with the same peripheral without removing the chip select. The delay is always inserted after each transfer and before removing the chip select if needed.

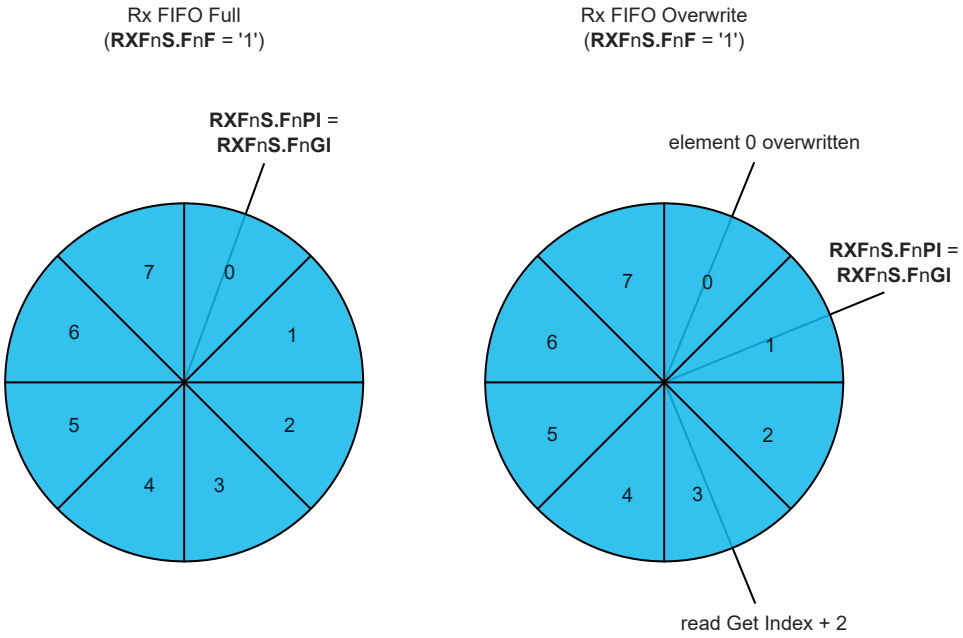
When DLYBCT=0x00, no delay between consecutive transfers is inserted and the clock keeps its duty cycle over the character transfers. In Serial Memory mode (MODE=1), DLYBCT is ignored and no delay is inserted. Otherwise, the following equation determines the delay:

Bits 11:8 – DATALEN[3:0] Data Length
The DATALEN field determines the number of data bits transferred. Reserved values should not be used.

When an Rx FIFO full condition (RXFnS.FnPI = RXFnS.FnGI) is signaled by RXFnS.FnF = ‘1’, the next message accepted for the FIFO will overwrite the oldest FIFO message. Put and get index are both incremented by one.

When an Rx FIFO is operated in overwrite mode and an Rx FIFO full condition is signaled, reading of the Rx FIFO elements should start at least at get index + 1. The reason for that is, that it might happen, that a received message is written to the Message RAM (put index) while the CPU is reading from the Message RAM (get index). In this case inconsistent data may be read from the respective Rx FIFO element. Adding an offset to the get index when reading from the Rx FIFO avoids this problem. The offset depends on how fast the CPU accesses the Rx FIFO. The figure below shows an offset of two with respect to the get index when reading the Rx FIFO. In this case the two messages stored in element 1 and 2 are lost.

Figure 39-8. Rx FIFO Overflow Handling



After reading from the Rx FIFO, the number of the last element read has to be written to the Rx FIFO Acknowledge Index RXFnA.FnA. This increments the get index to that element number. In case the put index has not been incremented to this Rx FIFO element, the Rx FIFO full condition is reset (RXFnS.FnF = ‘0’).

39.6.5.3 Dedicated Rx Buffers

The CAN supports up to 64 dedicated Rx Buffers. The start address of the dedicated Rx Buffer section is configured via RXBC.RBSA.

For each Rx Buffer a Standard or Extended Message ID Filter Element with SFEC / EFEC = “111” and SFID2 / EFID2[10:9] = “00” has to be configured (see 39.9.5 Standard Message ID Filter Element and 39.9.6 Extended Message ID Filter Element).

After a received message has been accepted by a filter element, the message is stored into the Rx Buffer in the Message RAM referenced by the filter element. The format is the same as for an Rx FIFO element. In addition the flag IR.DRX (Message stored in Dedicated Rx Buffer) in the interrupt register is set.

Timer/Counter Inputs (TC)

The TC waveform output WO[0] can be used as input source for the LUT (LUTCTRLx.INSELy=TC). Only consecutive instances of the TC, i.e. TCx and the subsequent TC(x+1), are available as default and alternative TC selections (e.g., TC0 and TC1 are sources for LUT0, TC1 and TC2 are sources for LUT1, etc). See the figure below for an example for LUT0. More general, the Timer/Counter selection for each LUT follows the formula:

$\mathrm{IN}[N][i] = DefaultTC[N \,\%\, \mathrm{TC\_Instance\_Number}]$

$\mathrm{IN}[N][i] = AlternativeTC[(N+1) \,\%\, \mathrm{TC\_Instance\_Number}]$

Where N represents the LUT number and i represents the LUT input index (i=0,1,2).

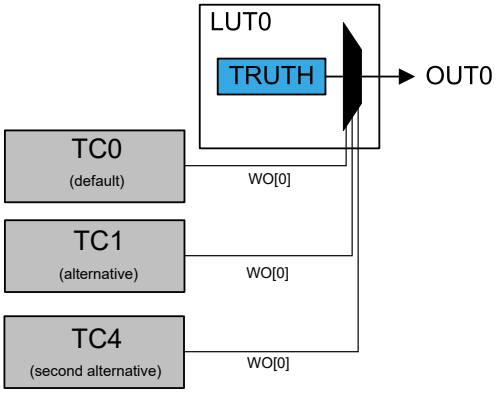
For devices with more than four TC instances, it is also possible to enable a second alternative option (LUTCTRLx.INSEL=ALT2TC). This option is intended to relax the alternative pin function or PCB design constraints when the default or the alternative TC instances are used for other purposes. When enabled, the Timer/Counter selection for each LUT follows the formula:

$\mathrm{IN}[N][i] = SecondAlternativeTC[(N+4) \,\%\, \mathrm{TC\_Instance\_Number}]$

Note that for not implemented TC_Instance_Number, the corresponding input is tied to ground.

Before selecting the waveform outputs, the TC must be configured first.

Figure 41-9. TC Input Selection



Timer/Counter for Control Application Inputs (TCC)

The TCC waveform outputs can be used as input source for the LUT. Only WO[2:0] outputs can be selected and routed to the respective LUT input (i.e., IN0 is connected to WO0, IN1 to WO1, and IN2 to WO2), as shown in the figure below.

Note:
The TCC selection for each LUT follows the formula:

$\mathrm{IN}[N][i] = TCC[N \,\%\, TC\mathrm{C\_Instance\_Number}]$

Where *N* represents the LUT number.

Before selecting the waveform outputs, the TCC must be configured first.

42.5.7 Debug Operation

When the CPU is halted in debug mode, the AES module continues normal operation. If the AES module is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging. The AES module can be forced to halt operation during debugging.

42.5.8 Register Access Protection

All registers with write-access are optionally write-protected by the peripheral access controller (PAC), except the

following register:

- Interrupt Flag Register (INTFLAG)

Write-protection is denoted by the Write-Protected property in the register description.

Write-protection does not apply to accesses through an external debugger. Refer to *PAC - Peripheral Access Controller* chapter for details.

Related Links

27. PAC - Peripheral Access Controller

42.5.9 Analog Connections

Not applicable.

42.6 Functional Description

42.6.1 Principle of Operation

The following is a high level description of the algorithm. These are the steps:

- KeyExpansion: Round keys are derived from the cipher key using Rijndael's key schedule.
- InitialRound:
  - AddRoundKey: Each byte of the state is combined with the round key using bitwise XOR.
- Rounds:
  - SubBytes: A non-linear substitution step where each byte is replaced with another according to a lookup table.
  - ShiftRows: A transposition step where each row of the state is shifted cyclically a certain number of steps.
  - MixColumns: A mixing operation which operates on the columns of the state, combining the four bytes in each column.
  - AddRoundKey
- Final Round (no MixColumns):
  - SubBytes
  - ShiftRows
  - AddRoundKey

The relationship between the module's clock frequency and throughput (in bytes per second) is given by:

Clock Frequency = (Throughput/2) x (Nr+1) for 2 byte parallel processing

Clock Frequency = (Throughput/4) x (Nr+1) for 4 byte parallel processing

Parameter	Type	Direction	Location	Data Length	Before Executing the Service	After Executing the Service
nu1OrderPointBase	nu1	I	Crypto RAM	u2ScalarLength + 4	Order of the Point A in the elliptic curve	Unchanged
nu1PrivateKey	nu1	I/O	Crypto RAM	u2ScalarLength + 4	Base of the Private Key	Unchanged
nu1HashBase (see **Note 1**)	nu1	I	Crypto RAM	u2ScalarLength + 4	Base of the hash value resulting from the previous SHA	Unchanged
u2ScalarLength	u2	I	–	–	Length of scalar (same length as the length of order)	Length of scalar
nu1PointABase	nu1	I/O	Crypto RAM	3*u2ModLength + 12	Input point A (three coordinates (X,Y) affine and Z = 1)	Resulting signature (R,S,0)
nu1ABase	nu1	I	Crypto RAM	2*u2ModLength + 8	Parameter a of the elliptic curve	Unchanged
nu1Workspace	nu1	I	Crypto RAM	8*u2ModLength + 44	–	Corrupted workspace

Note:

1. Whatever the chosen SHA, the resulting hash value may have a length inferior or equal to the modulo length and be padded with zeros until its total length is u2ModLength + 4.

43.3.7.9.5 Code Example

```
PUKCL_PARAM PUKCLParam;
PPUKCL_PARAM pvPUKCLParam = &PUKCLParam;

// ! The Random Number Generator must be initialized and started
// ! following the directives given for the RNG on the chip

PUKCL (u2Option) = 0;

// Depending on the option specified, not all fields should be filled
PUKCL _GF2NEcDsaGenerate(nu1ModBase) = <Base of the ram location of P>;
PUKCL _GF2NEcDsaGenerate(u2ModLength) = <Byte length of P>;
PUKCL _GF2NEcDsaGenerate(nu1CnsBase) = <Base of the ram location of Cns>;
PUKCL _GF2NEcDsaGenerate(nu1PointABase) = <Base of the A point>;
PUKCL _GF2NEcDsaGenerate(nu1PrivateKey) = <Base of the Private Key>;
PUKCL _GF2NEcDsaGenerate(nu1ScalarNumber) = <Base of the ScalarNumber>;
PUKCL _GF2NEcDsaGenerate(nu1OrderPointBase) = <Base of the order of A point>;
PUKCL _GF2NEcDsaGenerate(nu1ABase) = <Base of the a parameter of the curve>; PUKCL
_GF2NEcDsaGenerate(nu1Workspace) = <Base of the workspace>;
PUKCL _GF2NEcDsaGenerate(nu1HashBase) = <Base of the SHA resulting hash>;
...
```

45.8.2 Event Control

Name: EVCTRL
Offset: 0x02
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
			WINMONEO	RESRDYEO	STARTINV	FLUSHINV	STARTEI	FLUSHEI
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bit 5 – WINMONEO Window Monitor Event Out
This bit indicates whether the Window Monitor event output is enabled or not and an output event will be generated when the window monitor detects something.

Value	Description
0	Window Monitor event output is disabled and an event will not be generated.
1	Window Monitor event output is enabled and an event will be generated.

Bit 4 – RESRDYEO Result Ready Event Out
This bit indicates whether the Result Ready event output is enabled or not and an output event will be generated when the conversion result is available.

Value	Description
0	Result Ready event output is disabled and an event will not be generated.
1	Result Ready event output is enabled and an event will be generated.

Bit 3 – STARTINV Start Conversion Event Invert Enable
For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	Start event input source is not inverted.
1	Start event input source is inverted.

Bit 2 – FLUSHINV Flush Event Invert Enable
For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	Flush event input source is not inverted.
1	Flush event input source is inverted.

Bit 1 – STARTEI Start Conversion Event Input Enable
For the slave ADC, this bit has no effect when the SLAVEEN bit is set (CTRLA.SLAVEEN= 1).

Value	Description
0	A new conversion will not be triggered on any incoming event.
1	A new conversion will be triggered on any incoming event.

47.8.14 Data Buffer DAC1

Name: DATABUF1
Offset: 0x16
Reset: 0x0000
Property: Write-Synchronized

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

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

Bits 15:0 – DATABUF[15:0] DAC1 Data Buffer
DATABUF1 contains the value to be transferred into DATA1 when a START1 event occurs.

48.7.3.4 Event Control

Name: EVCTRL
Offset: 0x06
Reset: 0x0000
Property: PAC Write-Protection, Enable-Protected

Bit	15	14	13	12	11	10	9	8
			MCEO1	MCEO0				OVFEO
Access			R/W	R/W				R/W
Reset			0	0				0

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

Bit 13 – MCEO1 Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

Value	Description
0	Match/Capture event on channel x is disabled and will not be generated.
1	Match/Capture event on channel x is enabled and will be generated for every compare/capture.

Bit 12 – MCEO0 Match or Capture Channel x Event Output Enable [x = 1..0]
These bits enable the generation of an event for every match or capture on channel x.

Value	Description
0	Match/Capture event on channel x is disabled and will not be generated.
1	Match/Capture event on channel x is enabled and will be generated for every compare/capture.

Bit 8 – OVFEO Overflow/Underflow Event Output Enable
This bit enables the Overflow/Underflow event. When enabled, an event will be generated when the counter overflows/underflows.

Value	Description
0	Overflow/Underflow event is disabled and will not be generated.
1	Overflow/Underflow event is enabled and will be generated for every counter overflow/underflow.

Bit 5 – TCEI TC Event Enable
This bit is used to enable asynchronous input events to the TC.

Value	Description
0	Incoming events are disabled.
1	Incoming events are enabled.

3. If down-counting operation is desired, write the Counter Direction bit in the Control B Set register (CTRLBSET.DIR) to '1'.
4. Select the Waveform Generation operation in the WAVE register (WAVE.WAVEGEN).
5. Select the Waveform Output Polarity in the WAVE register (WAVE.POL).
6. The waveform output can be inverted for the individual channels using the Waveform Output Invert Enable bit group in the Driver register (DRVCTRL.INVEN).

49.6.2.2 Enabling, Disabling, and Resetting

The TCC is enabled by writing a '1' to the Enable bit in the Control A register (CTRLA.ENABLE). The TCC is disabled by writing a zero to CTRLA.ENABLE.

The TCC is reset by writing '1' to the Software Reset bit in the Control A register (CTRLA.SWRST). All registers in the TCC, except DBGCTRL, will be reset to their initial state, and the TCC will be disabled. Refer to Control A (49.8.1 CTRLA) register for details.

The TCC should be disabled before the TCC is reset to avoid undefined behavior.

49.6.2.3 Prescaler Selection

The GCLK_TCCx clock is fed into the internal prescaler.

The prescaler consists of a counter that counts up to the selected prescaler value, whereupon the output of the prescaler toggles.

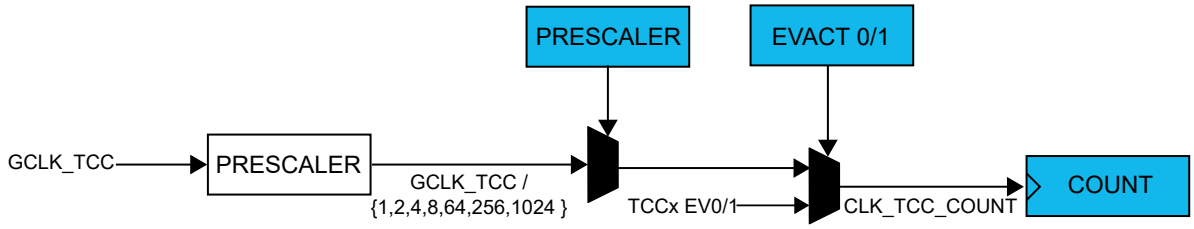
If the prescaler value is higher than one, the counter update condition can be optionally executed on the next GCLK_TCC clock pulse or the next prescaled clock pulse. For further details, refer to the Prescaler (CTRLA.PRESCALER) and Counter Synchronization (CTRLA.PRESYNC) descriptions.

Prescaler outputs from 1 to 1/1024 are available. For a complete list of available prescaler outputs, see the register description for the Prescaler bit group in the Control A register (CTRLA.PRESCALER).

Note: When counting events, the prescaler is bypassed.

The joint stream of prescaler ticks and event action ticks is called CLK_TCC_COUNT.

Figure 49-2. Prescaler



49.6.2.4 Counter Operation

Depending on the mode of operation, the counter is cleared, reloaded, incremented, or decremented at each TCC clock input (CLK_TCC_COUNT). A counter clear or reload mark the end of current counter cycle and the start of a new one.

The counting direction is set by the Direction bit in the Control B register (CTRLB.DIR). If the bit is zero, it's counting up and one if counting down.

The counter will count up or down for each tick (clock or event) until it reaches TOP or ZERO. When it's counting up and TOP is reached, the counter will be set to zero at the next tick (overflow) and the Overflow Interrupt Flag in the Interrupt Flag Status and Clear register (INTFLAG.OVF) will be set. When down-counting, the counter is reloaded with the TOP value when ZERO is reached (underflow), and INTFLAG.OVF is set.

Symbol	Parameter	Conditions		Min.	Typ.	Max.	Unit
INL	Integral Non Linearity	fADC = 500 ksps - R2R disabled	V_{DDANA} = 3.0V V_{REF} = V_{DDANA}	-	±2.6	±8.9	LSB
			V_{DDANA} = 3.0V V_{REF} = 2.0V	-	±3.3	±10.7	
		fADC = 1 Msps - R2R disabled	V_{DDANA} = 3.0V V_{REF} = V_{DDANA}	-	±2.9	±9.4	
DNL	Differential Non Linearity	fADC = 500 ksps - R2R disabled	V_{DDANA} = 3.0V V_{REF} = V_{DDANA}	-	±1	±1	LSB
			V_{DDANA} = 3.0V V_{REF} = 2.0V	-	±1	-1/+1.3	
		fADC = 1 Msps - R2R disabled	V_{DDANA} = 3.0V V_{REF} = VDDANA	-	±1	-1/+1.2	
Gain	Gain Error	fADC = 500 ksps - R2R disabled with gain compensation	V_{DDANA} = 3.0V V_{REF} = V_{DDANA}		±0.02	±0.3	%
			V_{DDANA} = 3.0V V_{REF} = 2.0V		±0.05	±0.2	
			V_{DDANA} = 3.0V 1V internal Ref		±1.1	±4.2	
			V_{DDANA} = 3.0V V_{REF} = V_{DDANA}/2		±0.1	±0.4	
Offset	Offset Error	fADC = 500 ksps - R2R disabled	V_{DDANA} = 3.0V V_{REF} = V_{DDANA}		±7	±17	mV
			V_{DDANA} = 3.0V V_{REF} = 2.0V		±5	±22	
			V_{DDANA} = 3.0V 1V internal Ref		±4.2	±10	
			V_{DDANA} = 3.0V V_{REF} = V_{DDANA}/2		±3.1	±22	
SFDR		Spurious Free Dynamic Range	fs = 1 Msps / F_{IN} = 14 kHz / Full range Input signal V_{DDANA} = 3.0V V_{REF} = V_{DDANA}	67.95	69.2	76.3	dB
SINAD at FS		Signal to Noise and Distortion ratio		55.7	57.5	61.1	
SNR at -3dB		Signal to Noise ratio		54.7	57.5	61.1	
THD				-73.7	-68.2	-65.8	