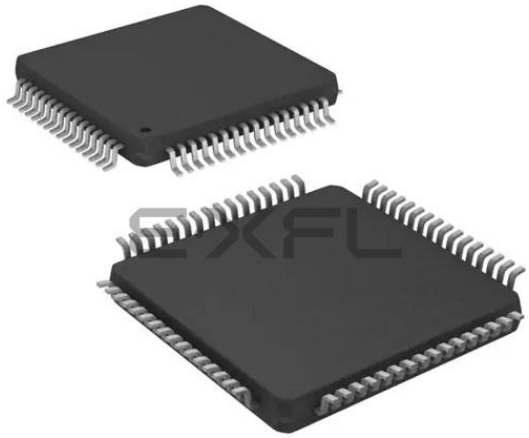


Welcome to E-XFL.COM

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	51
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	192K x 8
Voltage - Supply (Vcc/Vdd)	1.71V ~ 3.63V
Data Converters	A/D 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsamd51j19a-aut

Module	Source	Line
EIC NMI - External Interrupt Control	NMI	NMI
PM- Power Manager	SLEEPRDY	0
MCLK- Main Clock	CKRDY	1
OSCCTRL - Oscillators Control	XOSCFAIL 0	2
	XOSCRDY 0	
	XOSCFAIL 1	3
	XOSCRDY 1	
	DFLLLOCKC	4
	DFLLLOCKF	
	DFLLOOB	
	DFLLRCS	
	DFLLRDY	
	DPLLLCKF 0	5
	DPLLLCKR 0	
	DPLLLDRTO 0	
	DPLLLTO 0	
	DPLLLCKF 1	6
	DPLLLCKR 1	
	DPLLLDRTO 1	
	DPLLLTO 1	
OSC32KCTRL - 32kHz Oscillators Control	OSC32KRDY	7
	XOSC32KFAIL	
	XOSC32KRDY	
SUPC - Supply Controller	BOD12RDY	8
	BOD33RDY	
	B12SRDY	
	B33SRDY	
	VCORERDY	
	VREGRDY	
	BOD12DET	9
	BOD33DET	
WDT - Watchdog Timer	EW	10

The algorithm employed is the industry standard CRC32 algorithm using the generator polynomial 0xEDB88320 (reversed representation).

12.11.3.1 Starting CRC32 Calculation

CRC32 calculation for a memory range is started after writing the start address into the Address register (ADDR) and the size of the memory range into the Length register (LENGTH). Both must be word-aligned.

The initial value used for the CRC32 calculation must be written to the Data register (DATA). This value will usually be 0xFFFFFFFF, but can be, for example, the result of a previous CRC32 calculation if generating a common CRC32 of separate memory blocks.

Once completed, the calculated CRC32 value can be read out of the Data register. The read value must be complemented to match standard CRC32 implementations or kept non-inverted if used as starting point for subsequent CRC32 calculations.

The actual test is started by writing a '1' in the 32-bit Cyclic Redundancy Check bit of the Control register (CTRL.CRC). A running CRC32 operation can be canceled by resetting the module (writing '1' to CTRL.SWRST).

Related Links

25. NVMCTRL – Nonvolatile Memory Controller

12.11.3.2 Interpreting the Results

The user should monitor the Status A register. When the operation is completed, STATUSA.DONE is set. Then the Bus Error bit of the Status A register (STATUSA.BERR) must be read to ensure that no bus error occurred.

12.11.4 Debug Communication Channels

The Debug Communication Channels (DCCO and DCC1) consist of a pair of registers with associated handshake logic, accessible by both CPU and debugger even if the device is protected by the NVMCTRL security bit. The registers can be used to exchange data between the CPU and the debugger, during run time as well as in debug mode. This enables the user to build a custom debug protocol using only these registers.

The DCC0 and DCC1 registers are accessible when the protected state is active. When the device is protected, however, it is not possible to connect a debugger while the CPU is running (STATUSA.CRSTEXT is not writable and the CPU is held under Reset).

Two Debug Communication Channel status bits in the Status B registers (STATUS.DCCDx) indicate whether a new value has been written in DCC0 or DCC1. These bits, DCC0D and DCC1D, are located in the STATUSB registers. They are automatically set on write and cleared on read.

Note: The DCC0 and DCC1 registers are shared with the on-board memory testing logic (MBIST). Accordingly, DCC0 and DCC1 must not be used while performing MBIST operations.

Related Links

25. NVMCTRL – Nonvolatile Memory Controller

12.11.5 Debug Communication Channels DMA connection

The DCC0 and DCC1 registers can be used as a source or a destination of a DMA channel. The DSU generates one DMA request per Debug Communication Channels. The level of this DMA request is selectable writing the CFG.DCCDMALEVELx bit. Writing a 0 to this bit will configure the DMA request to trig on DCCx register empty. Writing a 1 to this bit will configure the DMA request to trig on DCCx register full.

Value	Description
0	The AHB clock for the PUKCC is stopped.
1	The AHB clock for the PUKCC is enabled.

Bit 19 – ICM ICM AHB Clock Enable

Value	Description
0	The AHB clock for the ICM is stopped.
1	The AHB clock for the ICM is enabled.

Bits 17, 18 – CANn CANn AHB Clock Enable

Value	Description
0	The AHB clock for the CANn is stopped.
1	The AHB clock for the CANn is enabled.

Bits 15, 16 – SDHCn SDHCn AHB Clock Enable

Value	Description
0	The AHB clock for the SDHCn is stopped.
1	The AHB clock for the SDHCn is enabled.

Bit 14 – GMAC GMAC AHB Clock Enable

Value	Description
0	The AHB clock for the GMAC is stopped.
1	The AHB clock for the GMAC is enabled.

Bit 13 – QSPI QSPI AHB Clock Enable

Value	Description
0	The AHB clock for the QSPI is stopped.
1	The AHB clock for the QSPI is enabled.

Bit 12 – PAC PAC AHB Clock Enable

Value	Description
0	The AHB clock for the PAC is stopped.
1	The AHB clock for the PAC is enabled.

Bits 11,7,5 – Reserved Reserved bits
Reserved bits are unused and reserved for future use. For compatibility with future devices, always write reserved bits to their reset value. If no reset value is given, write 0.

Bit 10 – USB USB AHB Clock Enable

Value	Description
0	The AHB clock for the USB is stopped.
1	The AHB clock for the USB is enabled.

Bit 9 – DMAC DMAC AHB Clock Enable

Value	Description
0	There is not reset operation ongoing
1	The reset operation is ongoing

23.8.13 Pin State

Name: PINSTATE
Offset: 0x38
Reset: 0x00000000

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
	PINSTATE[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
	PINSTATE[7:0]							
Access	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 15:0 – PINSTATE[15:0] Pin State
These bits return the valid pin state of the debounced external interrupt pin EXTINTx.

Bit	Function
	11: Type ID register 4 match If more than one Type ID is matched only one is indicated with priority 4 down to 1. With RX checksum offloading enabled: (bit 24 set in Network Configuration Register) 00: Neither the IP header checksum nor the TCP/UDP checksum was checked. 01: The IP header checksum was checked and was correct. Neither the TCP nor UDP checksum was checked. 10: Both the IP header and TCP checksum were checked and were correct. 11: Both the IP header and UDP checksum were checked and were correct.
21	VLAN tag detected—type ID of 0x8100. For packets incorporating the stacked VLAN processing feature, this bit will be set if the second VLAN tag has a type ID of 0x8100
20	Priority tag detected—type ID of 0x8100 and null VLAN identifier. For packets incorporating the stacked VLAN processing feature, this bit will be set if the second VLAN tag has a type ID of 0x8100 and a null VLAN identifier.
19:17	VLAN priority—only valid if bit 21 is set.
16	Canonical format indicator (CFI) bit (only valid if bit 21 is set).
15	End of frame—when set the buffer contains the end of a frame. If end of frame is not set, then the only valid status bit is start of frame (bit 14).
14	Start of frame—when set the buffer contains the start of a frame. If both bits 15 and 14 are set, the buffer contains a whole frame.
13	This bit has a different meaning depending on whether jumbo frames and ignore FCS modes are enabled. If neither mode is enabled this bit will be zero. With jumbo frame mode enabled: (bit 3 set in Network Configuration Register) Additional bit for length of frame (bit[13]), that is concatenated with bits[12:0] With ignore FCS mode enabled and jumbo frames disabled: (bit 26 set in Network Configuration Register and bit 3 clear in Network Configuration Register) This indicates per frame FCS status as follows: 0: Frame had good FCS 1: Frame had bad FCS, but was copied to memory as ignore FCS enabled.
12:0	These bits represent the length of the received frame which may or may not include FCS depending on whether FCS discard mode is enabled. With FCS discard mode disabled: (bit 17 clear in Network Configuration Register) Least significant 12 bits for length of frame including FCS. If jumbo frames are enabled, these 12 bits are concatenated with bit[13] of the descriptor above. With FCS discard mode enabled: (bit 17 set in Network Configuration Register) Least significant 12 bits for length of frame excluding FCS. If jumbo frames are enabled, these 12 bits are concatenated with bit[13] of the descriptor above.

25.8.6 Interrupt Flag Status and Clear

Name: INTFLAG
Offset: 0x10
Reset: 0x0000
Property: -

Bit	15	14	13	12	11	10	9	8
						SEEWRC	SEESOVF	SEESFULL
Access						R/W	R/W	R/W
Reset						0	0	0

Bit	7	6	5	4	3	2	1	0
	SUSP	NVME	ECCDE	ECCSE	LOCKE	PROGE	ADDRE	DONE
Access	R/W	R/W	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

Bit 10 – SEEWRC SEE Write Completed

- Unbuffered mode:
  0: AHB write is pending.

  1: AHB write has completed, and NVM is programmed with correct values.
- Buffered mode:
  0: AHB write is pending.

  1: AHB write has completed.

  If SEESTAT.LOAD=0, then the NVM is programmed with correct values.

  If SEESTAT.LOAD=1, then data is still pending in the Page Buffer.

Bit 9 – SEESOVF Active SEES Overflow
0: No SEES overflow have been detected since the last clear.

1: At least SEES overflow has been detected since the last clear.

This bit can be cleared by writing a one to its bit location.

Bit 8 – SEESFULL Active SEES Full
0: The active SEES is not full

1: The active SEES is Full, meaning that the next write will fail if the active sector is not reallocated.

This bit can be cleared by writing a one to its bit location.

Bit 7 – SUSP Suspended Write Or Erase Operation
0: No write/suspend has occurred since the last clear.

1: A write or erase operation has been suspended since the last clear.

This bit can be cleared by writing a one to its bit location.

31.7.8 Channel n Control

Name: CHANNEL
Offset: 0x20 + n*0x08 [n=0..31]
Reset: 0x00008000
Property: PAC Write-Protection

This register allows the user to configure channel n. To write to this register, do a single, 32-bit write of all the configuration data.

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
	ONDEMAND	RUNSTDBY			EDGSEL[1:0]		PATH[1:0]	
Access	RW	RW			RW	RW	RW	RW
Reset	1	0			0	0	0	0

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

Bit 15 – ONDEMAND Generic Clock On Demand

Value	Description
0	Generic clock for a channel is always on, if the channel is configured and generic clock source is enabled.
1	Generic clock is requested on demand while an event is handled

Bit 14 – RUNSTDBY Run in Standby
This bit is used to define the behavior during standby sleep mode.

Value	Description
0	The channel is disabled in standby sleep mode.
1	The channel is not stopped in standby sleep mode and depends on the CHANNEL.ONDEMAND bit.

Bits 11:10 – EDGSEL[1:0] Edge Detection Selection
These bits set the type of edge detection to be used on the channel.

These bits must be written to zero when using the asynchronous path.

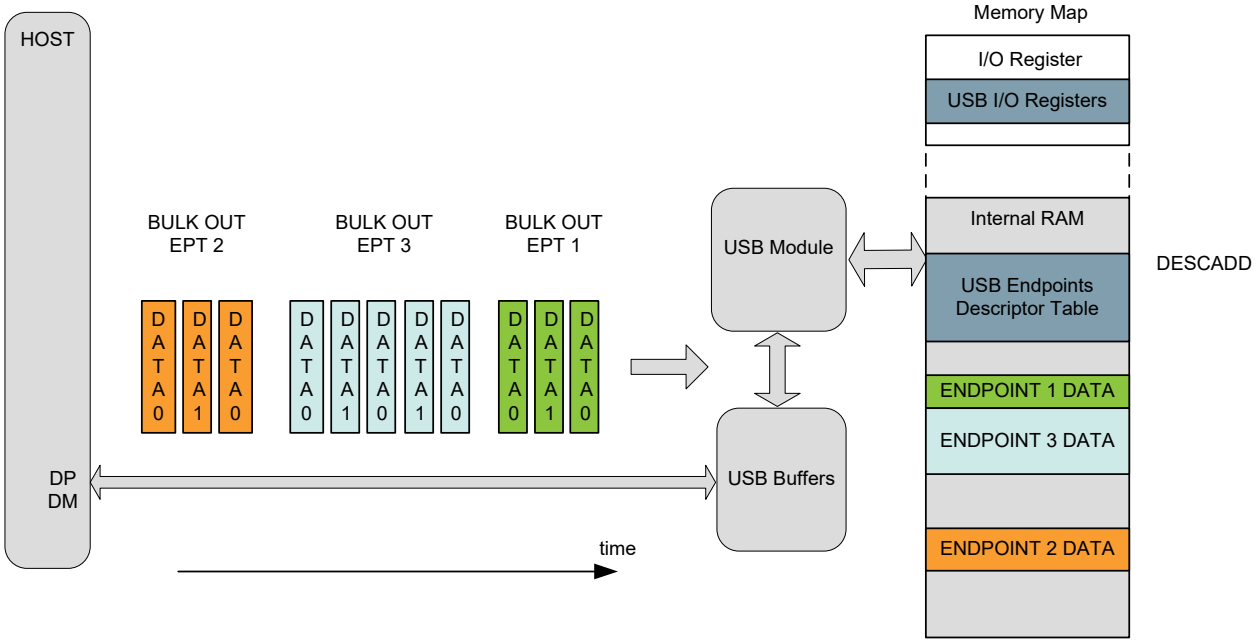
When the data PID matches and if the Received Setup Complete interrupt bit in the Device Endpoint Interrupt Flag register (EPINTFLAG.RXSTP) is equal to zero, ignoring the Bank 0 Ready bit in the Device Endpoint Status register (EPSTATUS.BK0RDY), the incoming data is written to the data buffer pointed to by the Data Buffer Address (ADDR). If the number of received data bytes exceeds the endpoint's maximum data payload size as specified by the PCKSIZE.SIZE, the remainders of the received data bytes are discarded. The packet will still be checked for bit-stuff and CRC errors. Software must never report a endpoint size to the host that is greater than the value configured in PCKSIZE.SIZE. If a bit-stuff or CRC error is detected in the packet, the USB module returns to idle and waits for the next token packet.

If data is successfully received, an ACK handshake is returned to the host, and the number of received data bytes, excluding the CRC, is written to the Byte Count (PCKSIZE.BYTE_COUNT). If the number of received data bytes is the maximum data payload specified by PCKSIZE.SIZE, no CRC data is written to the data buffer. If the number of received data bytes is the maximum data payload specified by PCKSIZE.SIZE minus one, only the first CRC data is written to the data buffer. If the number of received data is equal or less than the data payload specified by PCKSIZE.SIZE minus two, both CRC data bytes are written to the data buffer.

Finally the EPSTATUS is updated. Data Toggle OUT bit (EPSTATUS.DTGLOUT), the Data Toggle IN bit (EPSTATUS.DTGLIN), the current bank bit (EPSTATUS.CURRBK) and the Bank Ready 0 bit (EPSTATUS.BK0RDY) are set. Bank Ready 1 bit (EPSTATUS.BK1RDY) and the Stall Bank 0/1 bit (EPSTATUS.STALLQR0/1) are cleared on receiving the SETUP request. The RXSTP bit is set and triggers an interrupt if the Received Setup Interrupt Enable bit is set in Endpoint Interrupt Enable Set/Clear register (EPINTENSET/CLR.RXSTP).

38.6.2.7 Management of OUT Transactions

Figure 38-4. OUT Transfer: Data Packet Host to USB Device



When an OUT token is detected, and the device address of the token packet does not match DADD.DADD, the packet is discarded and the USB module returns to idle and waits for the next token packet.

read is performed is indicated by the Data Pointer field in the Data Buffer Pointer register (DATABUFPTR). This field is incremented by one or wrapped by hardware when a read from the DATA register address is performed. This field can also be programmed, giving the user direct control over which output buffer register to read from. Note that when AES module is in the CFB operation mode with the data segment size less than 128 bits, the output data must be read from the first (DATABUFPTR = 0) and/or second (DATABUFPTR = 1) output buffer registers (see Table 42-1). The Encryption Complete bit (INTFLAG.ENCCMP) is cleared by hardware after the processed data has been read from the relevant output buffer registers.

Table 42-1. Relevant Input/Output Data Registers for Different Confidentiality Modes

Confidentiality Mode	Relevant Input / Output Data Registers
ECB	All
CBC	All
OFB	All
128-bit CFB	All
64-bit CFB	First and Second
32-bit CFB	First
16-bit CFB	First
8-bit CFB	First
CTR	All

42.6.2.4 Start Modes

The Start mode field in the Control A Register (CTRLA.STARTMODE) allows the selection of encryption start mode.

1. Manual Start Mode
   In the Manual Start Mode the sequence is as follows:
   1.1. Write the 128/192/256 bit key in the Key Register (KEYWORD)
   1.2. Write the initialization vector or counter in the Initialization Vector Register (INTVECT). The initialization vector concerns all modes except ECB
   1.3. Enable interrupts in Interrupt Enable Set Register (INTENSET), depending on whether an interrupt is required or not at the end of processing.
   1.4. Write the data to be encrypted or decrypted in the Data Registers (DATA).
   1.5. Set the START bit in Control B Register (CTRLB.START) to begin the encryption or the decryption process.
   1.6. When the processing completes, the Encryption Complete bit in the Interrupt Flag Register (INTFLAG.ENCCMP) raises. If Encryption Complete interrupt has been enabled, the interrupt line of the AES is activated.
   1.7. When the software reads one of the Output Data Registers (DATA), INTFLAG.ENCCMP bit is automatically cleared.
2. Auto start Mode
   The Auto Start Mode is similar to the manual one, except in this mode, as soon as the correct number of input data registers is written, processing is automatically started without setting the START bit in the Control B Register. DMA operation uses this mode.

43. Public Key Cryptography Controller (PUKCC)

43.1 Overview

The Public Key Cryptography Controller (PUKCC) processes public key cryptography algorithm calculus in both GF(p) and GF(2n) fields.

The PUKCL (PUblic Key Cryptography Library) is stored in ROM inside the device. This library can be used in applications to access features of PUKCC.

The Public Key Cryptography Library includes complete implementation of the following public key cryptography algorithms:

- RSA (Rivest-Shamir-Adleman public key cryptosystem), DSA (Digital Signature Algorithm):
  - Modular Exponentiation with CRT up to 7168 bits
  - Modular Exponentiation without CRT up to 5376 bits
  - Prime generation
  - Utilities: GCD/modular Inverse, Divide, Modular reduction, Multiply, ...
- Elliptic Curves:
  - ECDSA GF(p) up to 521 bits for common curves (up to 1120 bits for future use)
  - ECDSA GF(2n) up to 571 bits for common curves (up to 1440 bits for future use)
  - Choice of the curves parameters so compatibility with NIST Curves or other curves in Weierstrass equation
  - Point Multiply
  - Point Add/Doubling
  - Other high level elliptic curves algorithms (ECDH, ...) can be implemented by user using library functions
- Deterministic Random Number Generation (DRNG ANSI X9.31) for DSA

43.2 Product Dependencies

43.2.1 I/O Lines

Not applicable.

43.2.2 Power Management

The PUKCC will continue to operate in any sleep mode, as long as its source clock is running.

43.2.3 Clocks

The bus clock (CLK_PUKCC_AHB) can be enabled and disabled by the Main Clock Controller.

Related Links

15. MCLK – Main Clock

43.2.4 DMA

Not applicable.

43.2.5 Interrupts

Not applicable.

Important: If the condition is verified, the length of R must be greater or equal to the length of X.

43.3.4.6.4 Parameters Definition

This service can easily be accessed through the use of the `PUKCL_CondCopy()` and `PUKCL()` macros.

Table 43-15. CondCopy Service Parameters

Parameter	Type	Direction	Location	Data Length	Before Executing the Service	After Executing the Service
u2Options	u2	I	–	–	Option for condition (see the following table)	Option for condition (see the following table)
Specific/ CarryIn	Bit	I	–	–	Bit CarryIn	Bit CarryIn
nu1XBase	nu1	I	Crypto RAM	u2XLength	Base of X	Base of X number untouched
nu1RBase	nu1	I	Crypto RAM	u2RLength	Base of R	Base of R filled with X if condition holds
u2RLength	u2	I	–	–	Length of R	Length of R
u2XLength	u2	I	–	–	Length of X	Length of X

43.3.4.6.5 Available Options

The option for the condition is set by the u2Options input parameter that must take one of the values listed in the following table.

Table 43-16. CondCopy Service Options

Option	Purpose	Needed parameters
PUKCL_CONDCOPY_ALWAYS	Always perform the copy	nu1XBase,u2XLength,nu1RBase, u2RLength
PUKCL_CONDCOPY_NEVER	Never perform the copy	None
PUKCL_CONDCOPY_IF_CARRY	Perform the copy if CarryIn is 1	Specific/CarryIn nu1XBase,u2XLength,nu1RBase, u2RLength
PUKCL_CONDCOPY_IF_NOT_CARRY	Perform the copy if CarryIn is zero	Specific/CarryIn nu1XBase,u2XLength,nu1RBase, u2RLength

43.3.4.6.6 Code Example

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

// CarryIn shall be beforehand filled (with zero or one) PUKCL(Specific).CarryIn = ...;
```

Parameter	Type	Direction	Location	Data Length	Before Executing the Service	After Executing the Service
u2ExpLength	u2	I	–	–	Significant length of EP or EQ	Significant length of EP or EQ
u1Blinding (see **Note 3**)	u4	I	–	–	Exponent unblinding value	Exponent unblinding value

Note:

1. This zone contains the number to be exponentiated (u2ModLength bytes) and is used during the computations as a workspace (four 32-bit words longer than the number to be exponentiated). At the end of the computation, it contains the correct result of the operation.
2. If the PUKCL_EXPMOD_EXPINPUKCCRAM option is not set, the location of the exponent MUST NOT be placed in the Crypto RAM, even partially.
3. It is possible to mask the exponent in memory using a 32-bit XOR mask value. Be aware that not only the exponent, but also the supplemental spill word has to be masked. If masking is not desired, the parameter should be set to 0.

43.3.5.4.5 Options

Most of the CRT options configure the Modular Exponentiation steps of the CRT and so are very similar to the Fast Modular Exponentiation options.

The options are set by the u2Options input parameter, which is composed of:

- the mandatory Calculus Mode Option described in Table 43-63
- the mandatory Window Size Option described in Table 43-64
- the indication of the presence of the exponent in Crypto RAM

Important: Please check precisely if one part of the exponent area (containing EP and EQ) is in Crypto RAM. If this is the case, the PUKCL_EXPMOD_EXPINPUKCCRAM option must be used.

The u2Options number is calculated by an “Inclusive OR” of the options. Some Examples in C language are:

- Operation: CRT using the Fast Modular Exponentiation with the window size equal to 1 and with no part of the Exponent area in the Crypto RAM
  ```
  PUKCL(u2Options) = PUKCL_EXPMOD_FASTRSA | PUKCL_EXPMOD_WINDOWSIZE_1;
  ```
- Operation:CRT using the Regular Modular Exponentiation with the window size equal to 2 and with one part the Exponent area in the Crypto RAM
  ```
  PUKCL(u2Options) = PUKCL_EXPMOD_REGULARRSA | PUKCL_EXPMOD_WINDOWSIZE_2 |
  PUKCL_EXPMOD_EXPINPUKCCRAM;
  ```

For this service, two exclusive Calculus Modes for the Modular Exponentiation steps of the CRT are possible. The following table describes the Calculus Mode Options.

- All overlapping between {nu1ModBase, u2ModLength + 4}, {nu1CnsBase, u2ModLength +8}, {nu1PointABase, 3*u2ModLength + 12}, {nu1PointBBase, 3*u2ModLength + 12} and {nu1Workspace, 5*u2ModLength + 32}

43.3.6.2.7 Status Returned Values

Table 43-69. ZpEccAddFast Service Return Codes

Returned Status	Importance	Meaning
PUKCL_OK	–	The computation passed without problem.

43.3.6.3 Point Addition and Subtraction

43.3.6.3.1 Purpose

This service is used to perform a point addition and point subtraction, based on a given elliptic curve over GF(p). Please note that:

- This service is not intended to add the same point twice. In this particular case, use the doubling service (see 43.3.6.4 Fast Point Doubling).

43.3.6.3.2 How to Use the Service

43.3.6.3.3 Description

The operation performed is:

$Pt_C = Pt_A \pm Pt_B$

In this computation, the following parameters need to be provided:

- A the input point is filled in projective coordinates (X,Y,Z) (pointed by {nu1PointABase, 3*u2ModLength + 12}). This point can be the Infinite Point.
- B the input point is filled in projective coordinates (X,Y,Z) (pointed by {nu1PointBBase, 3*u2ModLength + 12}). This point can be the Infinite Point.
- Cns the Fast Modular Constant filled (pointed by {nu1CnsBase,u2ModLength +8})
- P the modulus filled (pointed by {nu1ModBase,u2ModLength +4})
- The workspace not initialized (pointed by {nu1WorkSpace, 5*u2ModLength +32}
- The operator filled with the operation to perform (Addition or Subtraction)

The resulting C point is represented in projective coordinates (X,Y,Z) and is stored at the very same place than the input point A. This Point can be the Infinite Point.

The service name for this operation is `ZpEccAddSubFast`. This service uses Fast mode and Fast Modular Reduction for computations.

Note: Before using this service, ensure that the constant Cns has been calculated with the setup of the modular reduction functions.

Value	Description
0	The ADC is disabled.
1	The ADC is enabled.

Bit 0 – SWRST Software Reset
Writing a '0' to this bit has no effect.

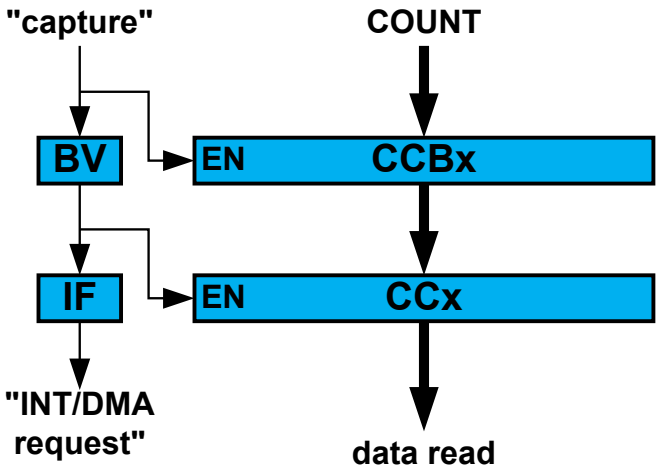
Writing a '1' to this bit resets all registers in the ADC, except DBGCTRL, to their initial state, and the ADC will be disabled.

Writing a '1' to CTRL.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded.

Due to synchronization there is a delay from writing CTRLA.SWRST until the reset is complete. CTRLA.SWRST and SYNCBUSY.SWRST will both be cleared when the reset is complete.

Value	Description
0	There is no reset operation ongoing.
1	The reset operation is ongoing.

Figure 48-11. Capture Double Buffering



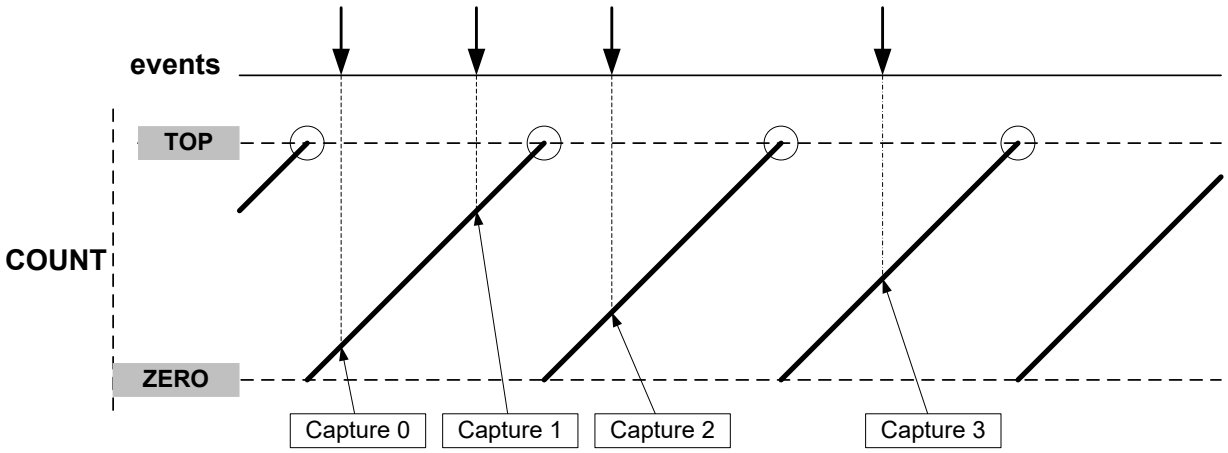
For input capture, the buffer register and the corresponding CCx act like a FIFO. When CCx is empty or read, any content in CCBUFx is transferred to CCx. The buffer valid flag is passed to set the CCx interrupt flag (IF) and generate the optional interrupt, event or DMA request. The CCBUFx register value can't be read, all captured data must be read from CCx register.

Note:
When up-counting (CTRLBSET.DIR=0), counter values lower than 1 cannot be captured. To capture the full range including value 0, the TC must be in down-counting mode (CTRLBSET.DIR=0).

48.6.2.8.1 Event Capture Action

The compare/capture channels can be used as input capture channels to capture events from the Event System and give them a timestamp. The following figure shows four capture events for one capture channel.

Figure 48-12. Input Capture Timing



The TC can detect capture overflow of the input capture channels: When a new capture event is detected while the Capture Interrupt flag (INTFLAG.MCx) is still set, the new timestamp will not be stored and INTFLAG.ERR will be set.

48.6.2.8.2 Period and Pulse-Width (PPW) Capture Action

The TC can perform two input captures and restart the counter on one of the edges. This enables the TC to measure the pulse width and period and to characterize the frequency *f* and duty cycle of an input signal:

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.MCEOx) enables the corresponding output event. The output event is disabled by writing EVCTRL.MCEOx=0.

One of the following event actions can be selected by the Event Action bit group in the Event Control register (EVCTRL.EVACT):

- Disable event action (OFF)
- Start TC (START)
- Re-trigger TC (RETRIGGER)
- Count on event (COUNT)
- Capture time stamp (STAMP)
- Capture Period (PPW and PWP)
- Capture Pulse Width (PW)

Writing a '1' to the TC Event Input bit in the Event Control register (EVCTRL.TCEI) enables input events to the TC. Writing a '0' to this bit disables input events to the TC. The TC requires only asynchronous event inputs. For further details on how configuring the asynchronous events, refer to *EVSYS - Event System*.

Related Links

31. EVSYS – Event System

48.6.7 Sleep Mode Operation

The TC can be configured to operate in any sleep mode. To be able to run in standby, the RUNSTDBY bit in the Control A register (CTRLA.RUNSTDBY) must be '1'. This peripheral can wake up the device from any sleep mode using interrupts or perform actions through the Event System.

If the On Demand bit in the Control A register (CTRLA.ONDEMAND) is written to '1', the module stops requesting its peripheral clock when the STOP bit in STATUS register (STATUS.STOP) is set to '1'. When a re-trigger or start condition is detected, the TC requests the clock before the operation starts.

48.6.8 Synchronization

Due to asynchronicity between the main clock domain and the peripheral clock domains, some registers need to be synchronized when written or read.

The following bits are synchronized when written:

- Software Reset and Enable bits in Control A register (CTRLA.SWRST and CTRLA.ENABLE)
- Capture Channel Buffer Valid bit in STATUS register (STATUS.CCBUFVx)

The following registers are synchronized when written:

- Control B Clear and Control B Set registers (CTRLBCLR and CTRLBSET)
- Count Value register (COUNT)
- Period Value and Period Buffer Value registers (PER and PERBUF)
- Channel x Compare/Capture Value and Channel x Compare/Capture Buffer Value registers (CCx and CCBUFx)

The following registers are synchronized when read:

- Count Value register (COUNT): synchronization is done on demand through READSYNC command (CTRLBSET.CMD).

Required write-synchronization is denoted by the "Write-Synchronized" property in the register description.

Name	Description
	or the Compare Channel 0 (CC0) register value depending on the waveform generator mode in 49.6.2.5.1 Waveform Output Generation Operations.
ZERO	The counter reaches ZERO when it contains all zeroes.
MAX	The counter reaches maximum when it contains all ones.
UPDATE	The timer/counter signals an update when it reaches ZERO or TOP, depending on the direction settings.
Timer	The timer/counter clock control is handled by an internal source.
Counter	The clock control is handled externally (e.g., counting external events).
CC	For compare operations, the CC are referred to as "compare channels." For capture operations, the CC are referred to as "capture channels."

Each TCC instance has up to four compare/capture channels (CCx).

The counter register (COUNT), period registers with buffer (PER and PERBUF), and compare and capture registers with buffers (CCx and CCBUFx) are 16- or 24-bit registers, depending on each TCC instance. Each buffer register has a buffer valid (BUFV) flag that indicates when the buffer contains a new value.

Under normal operation, the counter value is continuously compared to the TOP or ZERO value to determine whether the counter has reached TOP or ZERO. In either case, the TCC can generate interrupt requests or generate events for the Event System. In waveform generator mode, these comparisons are used to set the waveform period or pulse width.

A prescaled generic clock (GCLK_TCCx) and events from the event system can be used to control the counter. The event system is also used as a source to the input capture.

The Recoverable Fault Unit enables event controlled waveforms by acting directly on the generated waveforms of the TCC compare channels output. These events can restart, halt the timer/counter period, shorten the output pulse active time, or disable waveform output as long as the fault condition is present. This can typically be used for current sensing regulation, and zero-crossing and demagnetization re-triggering.

The MCE0 and MCE1 asynchronous event sources are shared with the Recoverable Fault Unit. Only asynchronous events are used internally when fault unit extension is enabled. For further details on how to configure asynchronous events routing, refer to *EVSYS – Event System*.

Recoverable fault sources can be filtered and/or windowed to avoid false triggering, for example from I/O pin glitches, by using digital filtering, input blanking, and qualification options. See also 49.6.3.5 Recoverable Faults.

In order to support applications with different types of motor control, ballast, LED, H-bridge, power converter, and other types of power switching applications, the following independent units are implemented in some of the TCC instances as optional and successive units:

- Recoverable faults and non-recoverable faults
- Output matrix
- Dead-time insertion

53.8.9 Debug Control

Name: DBGCTRL
Offset: 0x0F
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
								DBGRUN
Access								RW
Reset								0

Bit 0 – DBGRUN Debug Run Mode
This bit is not affected by software reset and should not be changed by software while the PDEC module is enabled.

Value	Description
0	The PDEC module is halted when the device is halted in debug mode.
1	The PDEC module continues normal operation when the device is halted in debug mode.