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	CANbus, 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	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 24x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsame51j18a-mut

17.8.1 Interrupt Enable Clear

Name: INTENCLR
Offset: 0x00
Reset: 0x00
Property: PAC Write-Protection

This register allows the user to disable an interrupt without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Set (INTENSET) register.

Bit	7	6	5	4	3	2	1	0
							DUALE	SINGLEE
Access							R/W	R/W
Reset							0	0

Bit 1 – DUALE Dual Bit Error Interrupt Enable Clear
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Dual Bit Error Interrupt Enable bit, which disables the Dual Bit Error interrupt.

Value	Description
0	The Dual Bit Error interrupt is disabled.
1	The Dual Bit Error interrupt is enabled.

Bit 0 – SINGLEE Single Bit Error Interrupt Enable Clear
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will clear the Single Bit Error Interrupt Enable bit, which disables the Single Bit Error interrupt.

Value	Description
0	The Single Bit Error interrupt is disabled.
1	The Single Bit Error interrupt is enabled.

Bit 6 – RUNSTDBY Run In Standby
The bit controls how the voltage reference behaves during standby sleep mode.

Value	Description
0	The voltage reference is halted during standby sleep mode.
1	The voltage reference is not stopped in standby sleep mode. If VREF.ONDEMAND=1, the voltage reference will be running when a peripheral is requesting it. If VREF.ONDEMAND=0, the voltage reference will always be running in standby sleep mode.

Bit 3 – TSSEL Temperature Sensor Channel Selection

Value	Description
0	The Temperature Sensor PTAT channel is selected.
1	The Temperature Sensor CTAT channel is selected.

Bit 2 – VREFOE Voltage Reference Output Enable

Value	Description
0	The Voltage Reference output (INTREF) is not available as an ADC input channel.
1	The Voltage Reference output (INTREF) is routed to an ADC input channel.

Bit 1 – TSEN Temperature Sensor Enable

Value	Description
0	Temperature Sensor is disabled.
1	Temperature Sensor is enabled and routed to an ADC input channel.

Offset	Name	Bit Pos.								
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							
0x8C	BKUP3	7:0	BKUP[7:0]							
		15:8	BKUP[15:8]							
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							
0x90	BKUP4	7:0	BKUP[7:0]							
		15:8	BKUP[15:8]							
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							
0x94	BKUP5	7:0	BKUP[7:0]							
		15:8	BKUP[15:8]							
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							
0x98	BKUP6	7:0	BKUP[7:0]							
		15:8	BKUP[15:8]							
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							
0x9C	BKUP7	7:0	BKUP[7:0]							
		15:8	BKUP[15:8]							
		23:16	BKUP[23:16]							
		31:24	BKUP[31:24]							

21.10 Register Description - Mode 1 - 16-Bit Counter

This Register Description section is valid if the RTC is in COUNT16 mode (CTRLA.MODE=1).

Registers can be 8, 16, or 32 bits wide. Atomic 8-, 16-, and 32-bit accesses are supported. In addition, the 8-bit quarters and 16-bit halves of a 32-bit register, and the 8-bit halves of a 16-bit register can be accessed directly.

Some registers require synchronization when read and/or written. Synchronization is denoted by the "Read-Synchronized" and/or "Write-Synchronized" property in each individual register description.

Optional write-protection by the Peripheral Access Controller (PAC) is denoted by the "PAC Write-Protection" property in each individual register description.

Some registers are enable-protected, meaning they can only be written when the module is disabled. Enable-protection is denoted by the "Enable-Protected" property in each individual register description.

Bits 5:0 – SECOND[5:0] Second
0 – 59

24.9.88 GMAC 1588 Timer Nanoseconds Register

Name: TN
Offset: 0x1D4
Reset: 0x00000000
Property: Read/Write

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.

1: SmartEEPROM is busy processing a read or a write operation.

Bit 1 – LOAD Page Buffer Loaded
0: SmartEEPROM has not left unwritten data in the page buffer.

1: SmartEEPROM has left unwritten data in the page buffer.

Bit 0 – ASEES Active SmartEEPROM Sector
This bit field is automatically loaded during startup from a special fuse in the NVM.

Indicates the active SEES

0: SEES0 is active

1: SEES1 is active

26.8.4 Interrupt Enable Register

Name: IER
Offset: 0x10
Reset: 0x00000000
Property: Write-Only

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

Bit	23	22	21	20	19	18	17	16
	RSU[3:0]				REC[3:0]			
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
	RWC[3:0]				RBE[3:0]			
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
	RDM[3:0]				RHC[3:0]			
Access	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

Bit 24 – URAD Undefined Register Access Detection Interrupt Enable
0: No effect

1: The Undefined Register Access interrupt is enabled.

Bits 23:20 – RSU[3:0] Region Status Updated Interrupt Enable
0: No effect

1: When RSU[i] is written to ‘1’, the region i Status Updated interrupt is enabled.

Bits 19:16 – REC[3:0] Region End bit Condition Detected Interrupt Enable
0: No effect

1: When REC[i] is written to ‘1’, the region i End bit Condition interrupt is enabled.

Bits 15:12 – RWC[3:0] Region Wrap Condition detected Interrupt Enable
0: No effect

1: When RWC[i] is written to ‘1’, the Region i Wrap Condition interrupt is enabled.

Bits 11:8 – RBE[3:0] Region Bus Error Interrupt Enable

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

Writing a '1' to this bit will clear the PUKCC interrupt flag.

Bit 11 – ICM Interrupt Flag for ICM
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the ICM, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the ICM interrupt flag.

Bit 10 – TRNG Interrupt Flag for TRNG
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the TRNG, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the TRNG interrupt flag.

Bit 9 – AES Interrupt Flag for AES
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the AES, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the AES interrupt flag.

Bit 7 – PDEC Interrupt Flag for PDEC
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the PDEC, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the PDEC interrupt flag.

Bit 6 – TC5 Interrupt Flag for TC5
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the TC5, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the TC5 interrupt flag.

Bit 5 – TC4 Interrupt Flag for TC4
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the TC4, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing a '1' to this bit will clear the TC4 interrupt flag.

Bit 4 – TCC3 Interrupt Flag for TCC3
This flags is set when a Peripheral Access Error occurs while accessing the peripheral associated with the TCC3, and will generate an interrupt request if INTENCLR/SET.ERR is '1'.

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

Writing '1' to CTRLA.SWRST will always take precedence, meaning that all other writes in the same write-operation will be discarded. Any register write access during the ongoing reset will result in an APB error. Reading any register will return the reset value of the register.

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.

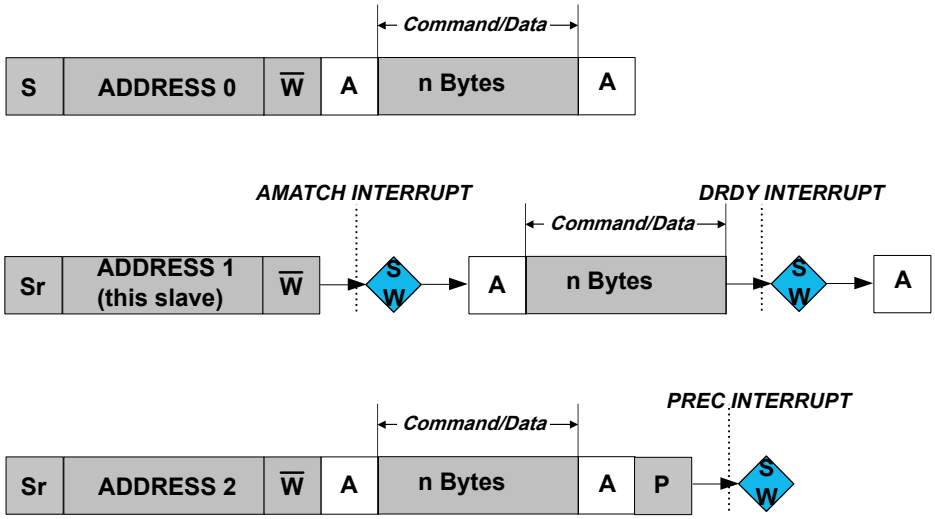
This bit is not enable-protected.

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

35.7 Register Summary

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0	RUNSTDBY			MODE[2:0]			ENABLE	SWRST
		15:8								IBON
		23:16			DIPO[1:0]				DOPO[1:0]	
		31:24		DORD	CPOL	CPHA	FORM[3:0]			
0x04	CTRLB	7:0		PLOADEN				CHSIZE[2:0]		
		15:8	AMODE[1:0]		MSSEN				SSDE	
		23:16							RXEN	
		31:24								
0x08	CTRLC	7:0			ICSPACE[5:0]					
		15:8								
		23:16								
		31:24								DATA32B
0x0C	BAUD	7:0	BAUD[7:0]							
0x0D ... 0x13	Reserved									
0x14	INTENCLR	7:0	ERROR				SSL	RXC	TXC	DRE
0x15	Reserved									
0x16	INTENSET	7:0	ERROR				SSL	RXC	TXC	DRE
0x17	Reserved									
0x18	INTFLAG	7:0	ERROR				SSL	RXC	TXC	DRE
0x19	Reserved									
0x1A	STATUS	7:0						BUFOVF		
		15:8					LENERR			
0x1C	SYNCBUSY	7:0				LENGTH		CTRLB	ENABLE	SWRST
		15:8								
		23:16								
		31:24								
0x20 ... 0x21	Reserved									
0x22	LENGTH	7:0	LEN[7:0]							
		15:8								LENEN
0x24	ADDR	7:0	ADDR[7:0]							
		15:8								
		23:16	ADDRMASK[7:0]							
		31:24								
0x28	DATA	7:0	DATA[7:0]							
		15:8	DATA[15:8]							
		23:16	DATA[23:16]							
		31:24	DATA[31:24]							
0x2C ...	Reserved									

Figure 36-13. PMBus Group Command Example



36.6.3 Additional Features

36.6.3.1 SMBus

The I^{2}C includes three hardware SCL low time-outs which allow a time-out to occur for SMBus SCL low time-out, master extend time-out, and slave extend time-out. This allows for SMBus functionality These time-outs are driven by the GCLK_SERCOM_SLOW clock. The GCLK_SERCOM_SLOW clock is used to accurately time the time-out and must be configured to use a 32KHz oscillator. The I^{2}C interface also allows for a SMBus compatible SDA hold time.

- $T_{TIMEOUT}$: SCL low time of 25..35ms – Measured for a single SCL low period. It is enabled by CTRLA.LOWTOUTEN.
- $T_{LOW:SEXT}$: Cumulative clock low extend time of 25 ms – Measured as the cumulative SCL low extend time by a slave device in a single message from the initial START to the STOP. It is enabled by CTRLA.SEXTTOEN.
- $T_{LOW:MEXT}$: Cumulative clock low extend time of 10 ms – Measured as the cumulative SCL low extend time by the master device within a single byte from START-to-ACK, ACK-to-ACK, or ACK-to-STOP. It is enabled by CTRLA.MEXTTOEN.

36.6.3.2 Smart Mode

The I^{2}C interface has a smart mode that simplifies application code and minimizes the user interaction needed to adhere to the I^{2}C protocol. The smart mode accomplishes this by automatically issuing an ACK or NACK (based on the content of CTRLB.ACKACT) as soon as DATA.DATA is read.

36.6.3.3 4-Wire Mode

Writing a '1' to the Pin Usage bit in the Control A register (CTRLA.PINOUT) will enable 4-wire mode operation. In this mode, the internal I^{2}C tri-state drivers are bypassed, and an external I^{2}C compliant tri-state driver is needed when connecting to an I^{2}C bus.

Table 39-2. Coding of DLC in CAN FD

DLC	9	10	11	12	13	14	15
Number of Data Bytes	12	16	20	24	32	48	64

In CAN FD frames, the bit timing will be switched inside the frame, after the BRS (Bit Rate Switch) bit, if this bit is recessive. Before the BRS bit, in the CAN FD arbitration phase, the nominal CAN bit timing is used as defined by the Nominal Bit Timing & Prescaler Register NBTP. In the following CAN FD data phase, the fast CAN bit timing is used as defined by the Data Bit Timing & Prescaler Register DBTP. The bit timing is switched back from the fast timing at the CRC delimiter or when an error is detected, whichever occurs first.

The maximum configurable bit rate in the CAN FD data phase depends on the CAN clock frequency (GCLK_CAN). Example: with a CAN clock frequency of 20MHz and the shortest configurable bit time of 4 t_q, the bit rate in the data phase is 5 Mbit/s.

In both data frame formats, CAN FD long and CAN FD fast, the value of the bit ESI (Error Status Indicator) is determined by the transmitter's error state at the start of the transmission. If the transmitter is error passive, ESI is transmitted recessive, else it is transmitted dominant.

39.6.2.4 Transceiver Delay Compensation

During the data phase of a CAN FD transmission only one node is transmitting, all others are receivers. The length of the bus line has no impact. When transmitting via pin CAN_TX the CAN receives the transmitted data from its local CAN transceiver via pin CAN_RX. The received data is delayed by the CAN transceiver's loop delay. In case this delay is greater than TSEG1 (time segment before sample point), a bit error is detected. In order to enable a data phase bit time that is even shorter than the transceiver loop delay, the delay compensation is introduced. Without transceiver delay compensation, the bit rate in the data phase of a CAN FD frame is limited by the transceivers loop delay.

Description

The CAN's protocol unit has implemented a delay compensation mechanism to compensate the transmitter delay, thereby enabling transmission with higher bit rates during the CAN FD data phase independent of the delay of a specific CAN transceiver.

To check for bit errors during the data phase of transmitting nodes, the delayed transmit data is compared against the received data at the Secondary Sample Point SSP. If a bit error is detected, the transmitter will react on this bit error at the next following regular sample point. During arbitration phase the delay compensation is always disabled.

The transmitter delay compensation enables configurations where the data bit time is shorter than the transmitter delay, it is described in detail in the new ISO11898-1. It is enabled by setting bit DBTP.TDC.

The received bit is compared against the transmitted bit at the SSP. The SSP position is defined as the sum of the measured delay from the CAN's transmit output CAN_TX through the transceiver to the receive input CAN_RX plus the transmitter delay compensation offset as configured by TDCR.TDCO. The transmitter delay compensation offset is used to adjust the position of the SSP inside the received bit (e.g. half of the bit time in the data phase). The position of the secondary sample point is rounded down to the next integer number of mtq.

PSR.TDCV shows the actual transmitter delay compensation value. PSR.TDCV is cleared when CCCR.INIT is set and is updated at each transmission of an FD frame while DBTP.TDC is set.

The following boundary conditions have to be considered for the transmitter delay compensation implemented in the CAN:

39.8.29 Rx FIFO 0 Acknowledge

Name: RXF0A
Offset: 0xA8
Reset: 0x00000000
Property: -

Bit	31	30	29	28	27	26	25	24
Access								
Reset								

Bit	23	22	21	20	19	18	17	16
Access								
Reset								

Bit	15	14	13	12	11	10	9	8
Access								
Reset								

Bit	7	6	5	4	3	2	1	0
			F0AI[5:0]					
Access			R/W	R/W	R/W	R/W	R/W	R/W
Reset			0	0	0	0	0	0

Bits 5:0 – F0AI[5:0] Rx FIFO 0 Acknowledge Index
After the Host has read a message or a sequence of messages from Rx FIFO 0 it has to write the buffer index of the last element read from Rx FIFO 0 to F0AI. This will set the Rx FIFO 0 Get Index RXF0S.F0GI to F0AI + 1 and update the FIFO 0 Fill Level RXF0S.F0FL.

Bits 21:16 – F1PI[5:0] Rx FIFO 1 Put Index
Rx FIFO 1 write index pointer, range 0 to 63.

Bits 13:8 – F1GI[5:0] Rx FIFO 1 Get Index
Rx FIFO 1 read index pointer, range 0 to 63.

Bits 6:0 – F1FL[6:0] Rx FIFO 1 Fill Level
Number of elements stored in Rx FIFO 1, range 0 to 64.

42.8.2 Control B

Name: CTRLB
Offset: 0x04
Reset: 0x00
Property: PAC Write-Protection

Bit	7	6	5	4	3	2	1	0
					GFMUL	EOM	NEWMSG	START
Access					R/W	R/W	R/W	R/W
Reset					0	0	0	0

Bit 3 – GFMUL GF Multiplication
This bit is applicable only to GCM mode.

Value	Description
0	No action
1	Setting this bit calculates GF multiplication with data buffer content and hashkey register content.

Bit 2 – EOM End of Message
This bit is applicable only to GCM mode.

Value	Description
0	No action
1	Setting this bit generates final GHASH value for the message.

Bit 1 – NEWMSG New Message
This bit is used in cipher block chaining (CBC), cipher feedback (CFB) and output feedback (OFB), counter (CTR) modes to indicate the hardware to use Initialization vector for encrypting the first block of message.

Value	Description
0	No action
1	Setting this bit indicates start of new message to the module.

Bit 0 – START Start Encryption/Decryption

Value	Description
0	No action
1	Start encryption / decryption in manual mode.

The minimum value for u2ModLength is 8 bytes, so the significant length of Num must be at least 8 bytes. To divide by a 32-bit value, the divider and numerator shall be multiplied by 232. The resulting remainder will have to be divided by 2^{32}, the quotient will be exact.

43.3.4.11.5 Code Example

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

// Fill all the fields
// In that case, the quotient will be computed
// If it was not needed, set nu1QuoBase to NULL
PUKCL_Div(nu1NumBase) = <Base of the ram location of Num>;
PUKCL_Div(nu1ModBase) = <Base of the ram location of Mod>;
PUKCL_Div(nu1QuoBase) = <Base of the ram location of Quo>;
PUKCL_Div(nu1WorkSpace) = <Base of the workspace>;
PUKCL_Div(nu1RBase) = <Base of the ram location of R>;
PUKCL_Div(u2NumLength) = <Length of Num>;
PUKCL_Div(u2ModLength) = <Length of Mod>;

// vPUKCL_Process() is a macro command, which populates the service name
// and then calls the library...
vPUKCL_Process(Div,pvPUKCLParam);
if (PUKCL(u2Status) == PUKCL_OK)
            {
            // The Division has been executed correctly
            ...
            }
else // Manage the error
```

43.3.4.11.6 Constraints

The following conditions must be avoided to ensure the service works correctly:

- nu1ModBase, nu1RBase, nu1QuoBase, nu1WorkSpace or nu1NumBase are not aligned on 32-bit boundaries
- {nu1ModBase, u2ModLength}, {nu1RBase, u2ModLength}, {nu1WorkSpace, 64} or{nu1NumBase, u2NumLength} are not in Crypto RAM
- u2ModLength, u2NumLength is either: < 4, > 0xffc or not a 32-bit length
- One or more overlaps exist between two of the areas: {nu1ModBase,u2ModLength},{nu1RBase, u2ModLength} {nu1NumBase, u2NumLength}(1) or {nu1WorkSpace,64}
- If nu1QuoBase is different from zero and: {nu1QuoBase, u2NumLength - u2ModLength + 4} are not in Crypto RAM
- If nu1QuoBase is different from zero and one or more overlaps exist between two of the areas: {nu1QuoBase, u2NumLength - u2ModLength + 4}, {nu1ModBase, u2ModLength}, {nu1RBase, u2ModLength}, {nu1NumBase, u2NumLength} or {nu1WorkSpace, 64}

Overlaps between {nu1RBase, u2ModLength} and {nu1NumBase, u2NumLength} are forbidden, but the equality between nu1RBase and nu1NumBase is authorized

43.3.4.11.7 Status Returned Values

Table 43-36. Div Service Return Codes

Returned Status	Importance	Meaning
PUKCL_OK	–	Service functioned correctly.
PUKCL_DIVISION_BY_ZERO	Severe	The operation was not performed because the Denominator value is zero.

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.

Note: One signal can be mapped on several pins.

Related Links

6. I/O Multiplexing and Considerations

45.5 Product Dependencies

In order to use this peripheral, other parts of the system must be configured correctly, as described below.

45.5.1 I/O Lines

Using the ADC's I/O lines requires the I/O pins to be configured using the port configuration (PORT).

Related Links

32. PORT - I/O Pin Controller

45.5.2 Power Management

The ADC will continue to operate in any sleep mode where the selected source clock is running. The ADC's interrupts except the OVERRUN interrupt, can be used to wake up the device from sleep modes. Events connected to the event system can trigger other operations in the system without exiting sleep modes.

Related Links

18. PM – Power Manager

45.5.3 Clocks

The ADC bus clocks (CLK_APB_ADCx) can be enabled in the Main Clock, which also defines the default state.

Each ADC requires a generic clock (GCLK_ADCx). This clock must be configured and enabled in the Generic Clock Controller (GCLK) before using the ADC.

A generic clock is asynchronous to the bus clock. Due to this asynchronicity, writes to certain registers will require synchronization between the clock domains. Refer to *Synchronization* for further details.

Related Links

15.6.2.6 Peripheral Clock Masking

14. GCLK - Generic Clock Controller

45.5.4 DMA

The DMA request line is connected to the DMA Controller (DMAC). Using the ADC DMA requests requires the DMA Controller to be configured first.

Related Links

22. DMAC – Direct Memory Access Controller

45.5.5 Interrupts

The interrupt request line is connected to the interrupt controller. Using the ADC interrupt requires the interrupt controller to be configured first.

Related Links

10.2 Nested Vector Interrupt Controller

Figure 51-8. Time Slot Application Block Diagram

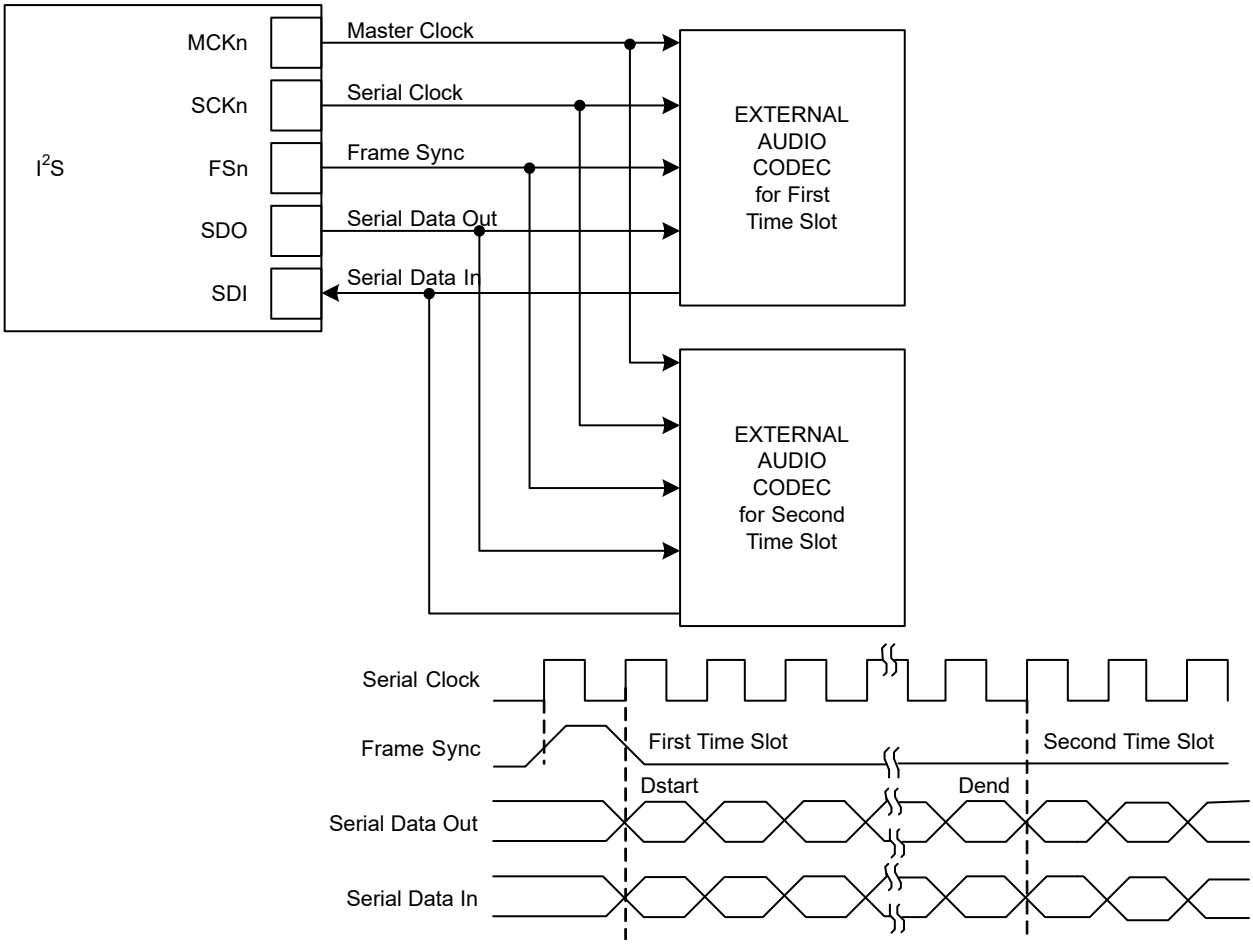


Figure 51-9. Codec Application Block Diagram

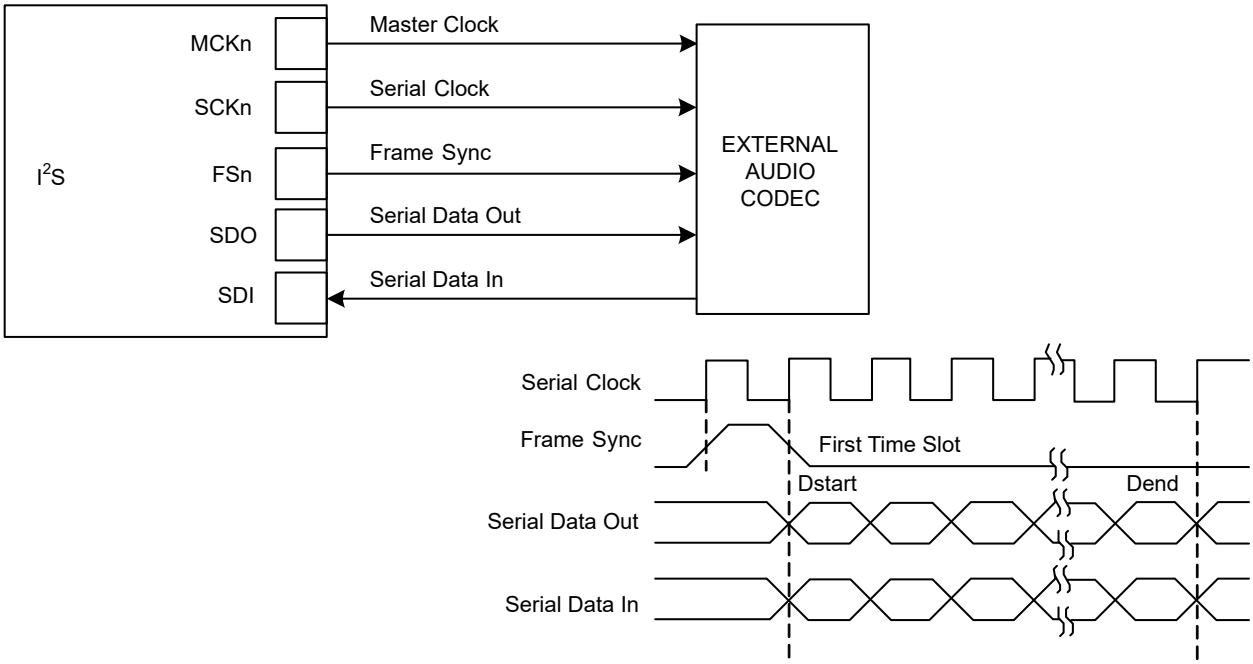


Figure 56-6. External Analog Reference Schematic With One Reference

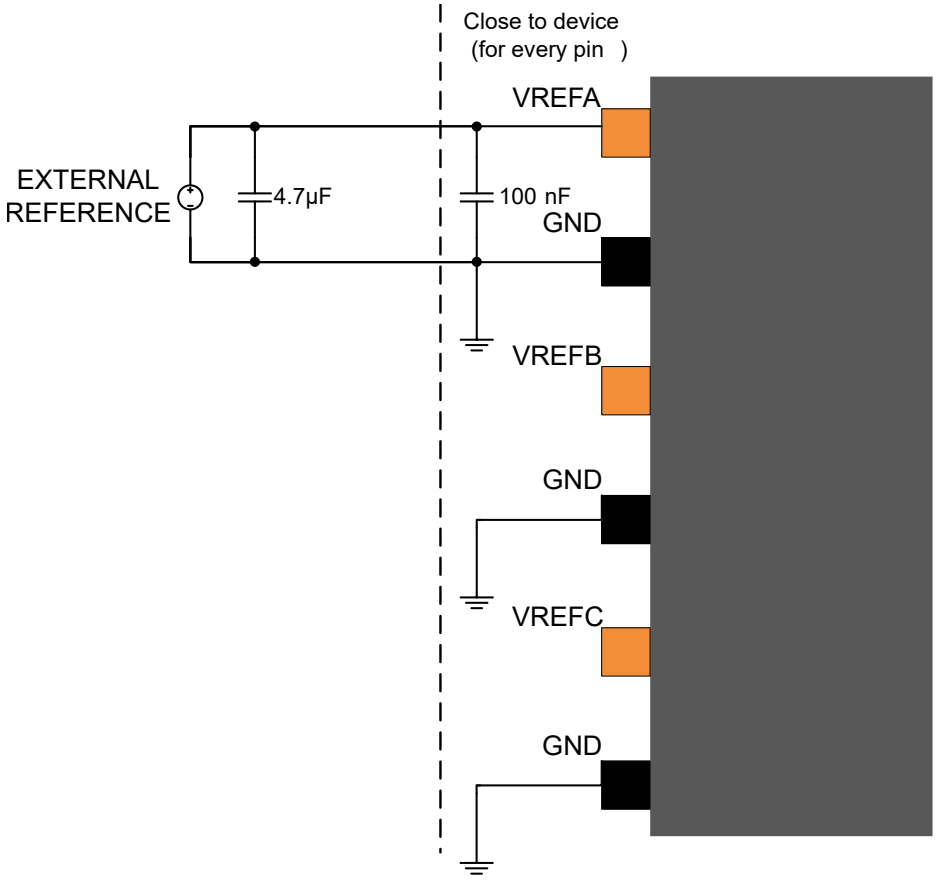


Table 56-2. External Analog Reference Connections

Signal Name	Recommended Pin Connection	Description
VREFx	1.0V to (V_{DDANA} - 0.6V) for ADC 1.0V to (V_{DDANA} - 0.6V) for DAC Decoupling/filtering capacitors 100nF(1)(2) and 4.7µF(1)	External reference VREFx for the analog port
GND		Ground

1. These values are only given as a typical example.

2. Decoupling capacitor should be placed close to the device for each supply pin pair in the signal group.

56.4 External Reset Circuit

When the external Reset function is used, connect the external Reset circuit to the $\overline{\text{RESET}}$ pin as shown below. If the external Reset function is not required, the circuit is not necessary: the $\overline{\text{RESET}}$ pin can either remain unconnected, or be driven LOW externally by the application circuitry.

The Reset switch can also be removed if a manual Reset is not necessary. The $\overline{\text{RESET}}$ pin itself has an internal pull-up resistor, hence it is optional to add any external pull-up resistor.