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®-M4
Core Size	32-Bit Single-Core
Speed	100MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	47
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 11x10b; D/A 1x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsam4n8ba-aur

10.2.3 PIO Controller C Multiplexing

Table 10-4. Multiplexing on PIO Controller C (PIOC)

I/O Line	Peripheral A	Peripheral B	Peripheral C	Extra Function	System Function	Comments
PC0						100 pins version
PC1						100 pins version
PC2						100 pins version
PC3						100 pins version
PC4		NPCS1				100 pins version
PC5						100 pins version
PC6						100 pins version
PC7		NPCS2				100 pins version
PC8		PWM0				100 pins version
PC9	RXD2	PWM1				100 pins version
PC10	TXD2	PWM2				100 pins version
PC11		PWM3				100 pins version
PC12				AD12(1)		100 pins version
PC13				AD10(1)		100 pins version
PC14	SCK2	PCK2				100 pins version
PC15				AD11(1)		100 pins version
PC16	RTS2	PCK0				100 pins version
PC17	CTS2	PCK1				100 pins version
PC18		PWM0				100 pins version
PC19		PWM1				100 pins version
PC20		PWM2				100 pins version
PC21		PWM3				100 pins version
PC22		PWM0				100 pins version
PC23		TIOA3				100 pins version
PC24		TIOB3				100 pins version
PC25		TCLK3				100 pins version
PC26		TIOA4				100 pins version
PC27		TIOB4				100 pins version
PC28		TCLK4				100 pins version
PC29		TIOA5		AD13(1)		100 pins version
PC30		TIOB5		AD14(1)		100 pins version
PC31		TCLK5		AD15(1)		100 pins version

Notes: 1. To select this extra function, refer to Section 34.5.3 “Analog Inputs”.

11.6.4 Memory Access Instructions

The table below shows the memory access instructions:

Table 11-17. Memory Access Instructions

Mnemonic	Description
ADR	Load PC-relative address
CLREX	Clear Exclusive
LDM{mode}	Load Multiple registers
LDR{type}	Load Register using immediate offset
LDR{type}	Load Register using register offset
LDR{type}T	Load Register with unprivileged access
LDR	Load Register using PC-relative address
LDRD	Load Register Dual
LDREX{type}	Load Register Exclusive
POP	Pop registers from stack
PUSH	Push registers onto stack
STM{mode}	Store Multiple registers
STR{type}	Store Register using immediate offset
STR{type}	Store Register using register offset
STR{type}T	Store Register with unprivileged access
STREX{type}	Store Register Exclusive

The value to load or store can be a byte, halfword, word, or two words. Bytes and halfwords can either be signed or unsigned. See “Address Alignment” .

The table below shows the ranges of offset for immediate, pre-indexed and post-indexed forms.

Table 11-18. Offset Ranges

Instruction Type	Immediate Offset	Pre-indexed	Post-indexed
Word, halfword, signed halfword, byte, or signed byte	-255 to 4095	-255 to 255	-255 to 255
Two words	multiple of 4 in the range -1020 to 1020	multiple of 4 in the range -1020 to 1020	multiple of 4 in the range -1020 to 1020

Restrictions

For load instructions:

- *Rt* can be SP or PC for word loads only
- *Rt* must be different from *Rt2* for two-word loads
- *Rn* must be different from *Rt* and *Rt2* in the pre-indexed or post-indexed forms.

When *Rt* is PC in a word load instruction:

- Bit[0] of the loaded value must be 1 for correct execution
- A branch occurs to the address created by changing bit[0] of the loaded value to 0
- If the instruction is conditional, it must be the last instruction in the IT block.

For store instructions:

- *Rt* can be SP for word stores only
- *Rt* must not be PC
- *Rn* must not be PC
- *Rn* must be different from *Rt* and *Rt2* in the pre-indexed or post-indexed forms.

Condition Flags

These instructions do not change the flags.

Examples

```
LDR     R8, [R10]              ; Loads R8 from the address in R10.
LDRNE   R2, [R5, #960]!        ; Loads (conditionally) R2 from a word
                               ; 960 bytes above the address in R5, and
                               ; increments R5 by 960.
STR     R2, [R9,#const-struc]  ; const-struc is an expression evaluating
                               ; to a constant in the range 0-4095.
STRH    R3, [R4], #4           ; Store R3 as halfword data into address in
                               ; R4, then increment R4 by 4
LDRD    R8, R9, [R3, #0x20]    ; Load R8 from a word 32 bytes above the
                               ; address in R3, and load R9 from a word 36
                               ; bytes above the address in R3
STRD    R0, R1, [R8], #-16     ; Store R0 to address in R8, and store R1 to
                               ; a word 4 bytes above the address in R8,
                               ; and then decrement R8 by 16.
```

11.8.3.3 Interrupt Set-pending Registers

Name: NVIC_ISPRx [x=0..7]

Access: Read-write

Reset: 0x000000000

31	30	29	28	27	26	25	24
SETPEND							

23	22	21	20	19	18	17	16
SETPEND							

15	14	13	12	11	10	9	8
SETPEND							

7	6	5	4	3	2	1	0
SETPEND							

These registers force interrupts into the pending state, and show which interrupts are pending.

- **SETPEND: Interrupt Set-pending**

Write:

0: No effect.

1: Changes the interrupt state to pending.

Read:

0: Interrupt is not pending.

1: Interrupt is pending.

Notes:
1. Writing 1 to an ISPR bit corresponding to an interrupt that is pending has no effect.
2. Writing 1 to an ISPR bit corresponding to a disabled interrupt sets the state of that interrupt to pending.

This fault is chained to the handler. This means that when the processor sets this bit to 1, the original return stack is still present. The processor does not adjust the SP from the failing return, does not performed a new save, and does not write a fault address to the BFAR.

- **STKERR: Bus Fault on Stacking for Exception Entry**

This is part of “BFSR: Bus Fault Status Subregister” .

0: No stacking fault.

1: Stacking for an exception entry has caused one or more bus faults.

When the processor sets this bit to 1, the SP is still adjusted but the values in the context area on the stack might be incorrect. The processor does not write a fault address to the SCB_BFAR register.

- **BFARVALID: Bus Fault Address Register (BFAR) Valid flag**

This is part of “BFSR: Bus Fault Status Subregister” .

0: The value in SCB_BFAR is not a valid fault address.

1: SCB_BFAR holds a valid fault address.

The processor sets this bit to 1 after a bus fault where the address is known. Other faults can set this bit to 0, such as a memory management fault occurring later.

If a bus fault occurs and is escalated to a hard fault because of priority, the hard fault handler must set this bit to 0. This prevents problems if returning to a stacked active bus fault handler whose SCB_BFAR value has been overwritten.

- **UNDEFINSTR: Undefined Instruction Usage Fault**

This is part of “UFSR: Usage Fault Status Subregister” .

0: No undefined instruction usage fault.

1: The processor has attempted to execute an undefined instruction.

When this bit is set to 1, the PC value stacked for the exception return points to the undefined instruction.

An undefined instruction is an instruction that the processor cannot decode.

- **INVSTATE: Invalid State Usage Fault**

This is part of “UFSR: Usage Fault Status Subregister” .

0: No invalid state usage fault.

1: The processor has attempted to execute an instruction that makes illegal use of the EPSR.

When this bit is set to 1, the PC value stacked for the exception return points to the instruction that attempted the illegal use of the EPSR.

This bit is not set to 1 if an undefined instruction uses the EPSR.

- **INVPC: Invalid PC Load Usage Fault**

This is part of “UFSR: Usage Fault Status Subregister” . It is caused by an invalid PC load by EXC_RETURN:

0: No invalid PC load usage fault.

1: The processor has attempted an illegal load of EXC_RETURN to the PC, as a result of an invalid context, or an invalid EXC_RETURN value.

When this bit is set to 1, the PC value stacked for the exception return points to the instruction that tried to perform the illegal load of the PC.

11.9.1.15 Hard Fault Status Register

Name: SCB_HFSR

Access: Read-write

Reset: 0x000000000

31	30	29	28	27	26	25	24
DEBUGEVT	FORCED	–					

23	22	21	20	19	18	17	16
–							

15	14	13	12	11	10	9	8
–							

7	6	5	4	3	2	1	0
–						VECTTBL	–

The HFSR register gives information about events that activate the hard fault handler. This register is read, write to clear. This means that bits in the register read normally, but writing 1 to any bit clears that bit to 0.

- **DEBUGEVT: Reserved for Debug Use**

When writing to the register, write 0 to this bit, otherwise the behavior is unpredictable.

- **FORCED: Forced Hard Fault**

It indicates a forced hard fault, generated by escalation of a fault with configurable priority that cannot be handles, either because of priority or because it is disabled:

0: No forced hard fault.

1: Forced hard fault.

When this bit is set to 1, the hard fault handler must read the other fault status registers to find the cause of the fault.

- **VECTTBL: Bus Fault on a Vector Table**

It indicates a bus fault on a vector table read during an exception processing:

0: No bus fault on vector table read.

1: Bus fault on vector table read.

This error is always handled by the hard fault handler.

When this bit is set to 1, the PC value stacked for the exception return points to the instruction that was preempted by the exception.

Note: The HFSR bits are sticky. This means that, as one or more fault occurs, the associated bits are set to 1. A bit that is set to 1 is cleared to 0 only by writing 1 to that bit, or by a reset.

11.12 Glossary

This glossary describes some of the terms used in technical documents from ARM.

Abort

A mechanism that indicates to a processor that the value associated with a memory access is invalid. An abort can be caused by the external or internal memory system as a result of attempting to access invalid instruction or data memory.

Aligned

A data item stored at an address that is divisible by the number of bytes that defines the data size is said to be aligned. Aligned words and halfwords have addresses that are divisible by four and two respectively. The terms word-aligned and halfword-aligned therefore stipulate addresses that are divisible by four and two respectively.

Banked register

A register that has multiple physical copies, where the state of the processor determines which copy is used. The Stack Pointer, SP (R13) is a banked register.

Base register

In instruction descriptions, a register specified by a load or store instruction that is used to hold the base value for the instruction's address calculation. Depending on the instruction and its addressing mode, an offset can be added to or subtracted from the base register value to form the address that is sent to memory.

See also "Index register"

Big-endian (BE)

Byte ordering scheme in which bytes of decreasing significance in a data word are stored at increasing addresses in memory.

See also "Byte-invariant" , "Endianness" , "Little-endian (LE)" .

Big-endian memory

Memory in which:
a byte or halfword at a word-aligned address is the most significant byte or halfword within the word at that address,
a byte at a halfword-aligned address is the most significant byte within the halfword at that address.

See also "Little-endian memory" .

Breakpoint

A breakpoint is a mechanism provided by debuggers to identify an instruction at which program execution is to be halted. Breakpoints are inserted by the programmer to enable inspection of register contents, memory locations, variable values at fixed points in the program execution to test that the program is operating correctly. Breakpoints are removed after the program is successfully tested.

15.6.4 RTC Calendar Register

Name: RTC_CALR

Address: 0x400E146C

Access: Read-write

31	30	29	28	27	26	25	24
–	–	DATE					

23	22	21	20	19	18	17	16
DAY			MONTH				

15	14	13	12	11	10	9	8
YEAR							

7	6	5	4	3	2	1	0
–	CENT						

- **CENT: Current Century**

The range that can be set is 19 - 20 (gregorian) or 13-14 (persian) (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

- **YEAR: Current Year**

The range that can be set is 00 - 99 (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

- **MONTH: Current Month**

The range that can be set is 01 - 12 (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

- **DAY: Current Day in Current Week**

The range that can be set is 1 - 7 (BCD).

The coding of the number (which number represents which day) is user-defined as it has no effect on the date counter.

- **DATE: Current Day in Current Month**

The range that can be set is 01 - 31 (BCD).

The lowest four bits encode the units. The higher bits encode the tens.

All non-significant bits read zero.

15.6.11 RTC Interrupt Mask Register

Name: RTC_IMR

Address: 0x400E1488

Access: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	CAL	TIM	SEC	ALR	ACK

- **ACK: Acknowledge Update Interrupt Mask**

0 = The acknowledge for update interrupt is disabled.

1 = The acknowledge for update interrupt is enabled.

- **ALR: Alarm Interrupt Mask**

0 = The alarm interrupt is disabled.

1 = The alarm interrupt is enabled.

- **SEC: Second Event Interrupt Mask**

0 = The second periodic interrupt is disabled.

1 = The second periodic interrupt is enabled.

- **TIM: Time Event Interrupt Mask**

0 = The selected time event interrupt is disabled.

1 = The selected time event interrupt is enabled.

- **CAL: Calendar Event Interrupt Mask**

0 = The selected calendar event interrupt is disabled.

1 = The selected calendar event interrupt is enabled.

17.5 Supply Controller (SUPC) User Interface

The User Interface of the Supply Controller is part of the System Controller User Interface.

17.5.1 System Controller (SYSC) User Interface

Table 17-1. System Controller Registers

Offset	System Controller Peripheral	Name
0x00-0x0c	Reset Controller	RSTC
0x10-0x2C	Supply Controller	SUPC
0x30-0x3C	Real Time Timer	RTT
0x50-0x5C	Watchdog Timer	WDT
0x60-0x8C	Real Time Clock	RTC
0x90-0xDC	General Purpose Backup Register	GPBR
0xE0	Reserved	
0xE4	Write Protect Mode Register	SYSC_WPMR
0xE8-0xF8	Reserved	

17.5.2 Supply Controller (SUPC) User Interface

Table 17-2. Register Mapping

Offset	Register	Name	Access	Reset
0x00	Supply Controller Control Register	SUPC_CR	Write-only	N/A
0x04	Supply Controller Supply Monitor Mode Register	SUPC_SMMR	Read-write	0x0000_0000
0x08	Supply Controller Mode Register	SUPC_MR	Read-write	0x0000_5A00
0x0C	Supply Controller Wake-up Mode Register	SUPC_WUMR	Read-write	0x0000_0000
0x10	Supply Controller Wake-up Inputs Register	SUPC_WUIR	Read-write	0x0000_0000
0x14	Supply Controller Status Register	SUPC_SR	Read-only	0x0000_0000
0x18	Reserved			

21. SAM-BA Boot Program

21.1 Description

The SAM-BA Boot Program integrates an array of programs permitting download and/or upload into the different memories of the product.

21.2 Hardware and Software Constraints

- SAM-BA Boot uses the first 2048 bytes of the SRAM for variables and stacks. The remaining available size can be used for user's code.
- UART0 requirements: None.

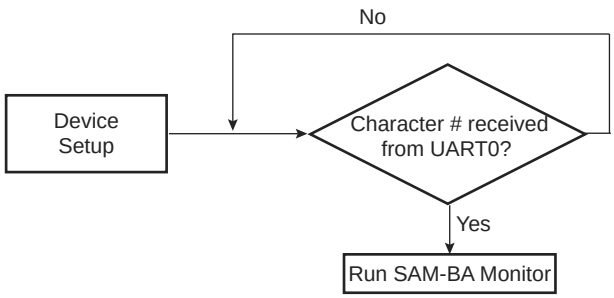
Table 21-1. Pins Driven during Boot Program Execution

Peripheral	Pin	PIO Line
UART0	URXD0	PA9
UART0	UTXD0	PA10

21.3 Flow Diagram

The Boot Program implements the algorithm illustrated in Figure 21-1.

Figure 21-1. Boot Program Algorithm Flow Diagram



The SAM-BA Boot Program uses the internal 12 MHz RC oscillator as source clock for PLL. The MCK runs from PLL divided by 2. The core runs at 48 MHz.

21.4 Device Initialization

The initialization sequence is the following:

1. Stack setup
2. Set up the Embedded Flash Controller
3. Switch on internal 12 MHz RC oscillator
4. Configure PLL to run at 96 MHz
5. Switch MCK to run on PLL divided by 2
6. Configure UART0
7. Disable Watchdog
8. Wait for a character on UART0
9. Jump to SAM-BA monitor (see Section 21.5 “SAM-BA Monitor”)

25.16.1 PMC System Clock Enable Register

Name: PMC_SCER

Address: 0x400E0400

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
–	–	–	–	–	PCK2	PCK1	PCK0
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	–

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

- **PCKx: Programmable Clock x Output Enable**

0 = No effect.

1 = Enables the corresponding Programmable Clock output.

27. Parallel Input/Output (PIO) Controller

27.1 Description

The Parallel Input/Output Controller (PIO) manages up to 32 fully programmable input/output lines. Each I/O line may be dedicated as a general-purpose I/O or be assigned to a function of an embedded peripheral. This assures effective optimization of the pins of a product.

Each I/O line is associated with a bit number in all of the 32-bit registers of the 32-bit wide User Interface.

Each I/O line of the PIO Controller features:

- An input change interrupt enabling level change detection on any I/O line.
- Additional Interrupt modes enabling rising edge, falling edge, low level or high level detection on any I/O line.
- A glitch filter providing rejection of glitches lower than one-half of PIO clock cycle.
- A debouncing filter providing rejection of unwanted pulses from key or push button operations.
- Multi-drive capability similar to an open drain I/O line.
- Control of the pull-up and pull-down of the I/O line.
- Input visibility and output control.

The PIO Controller also features a synchronous output providing up to 32 bits of data output in a single write operation.

27.2 Embedded Characteristics

- Up to 32 Programmable I/O Lines
- Fully Programmable through Set/Clear Registers
- Multiplexing of Four Peripheral Functions per I/O Line
- For each I/O Line (Whether Assigned to a Peripheral or Used as General Purpose I/O)
  - Input Change Interrupt
  - Programmable Glitch Filter
  - Programmable Debouncing Filter
  - Multi-drive Option Enables Driving in Open Drain
  - Programmable Pull Up on Each I/O Line
  - Pin Data Status Register, Supplies Visibility of the Level on the Pin at Any Time
  - Additional Interrupt Modes on a Programmable Event: Rising Edge, Falling Edge, Low Level or High Level
- Synchronous Output, Provides Set and Clear of Several I/O lines in a Single Write
- Write Protect Registers
- Programmable Schmitt Trigger Inputs

- **SVEN: TWI Slave Mode Enabled**

0 = No effect.

1 = If SVDIS = 0, the slave mode is enabled.

Note: Switching from Master to Slave mode is only permitted when TXCOMP = 1.

- **SVDIS: TWI Slave Mode Disabled**

0 = No effect.

1 = The slave mode is disabled. The shifter and holding characters (if it contains data) are transmitted in case of read operation. In write operation, the character being transferred must be completely received before disabling.

- **QUICK: SMBUS Quick Command**

0 = No effect.

1 = If Master mode is enabled, a SMBUS Quick Command is sent.

- **SWRST: Software Reset**

0 = No effect.

1 = Equivalent to a system reset.

30.6.9 UART Baud Rate Generator Register

Name: UART_BRGR

Address: 0x400E0620 (0), 0x400E0820 (1), 0x40044020 (2), 0x40048020 (3)

Access: Read-write

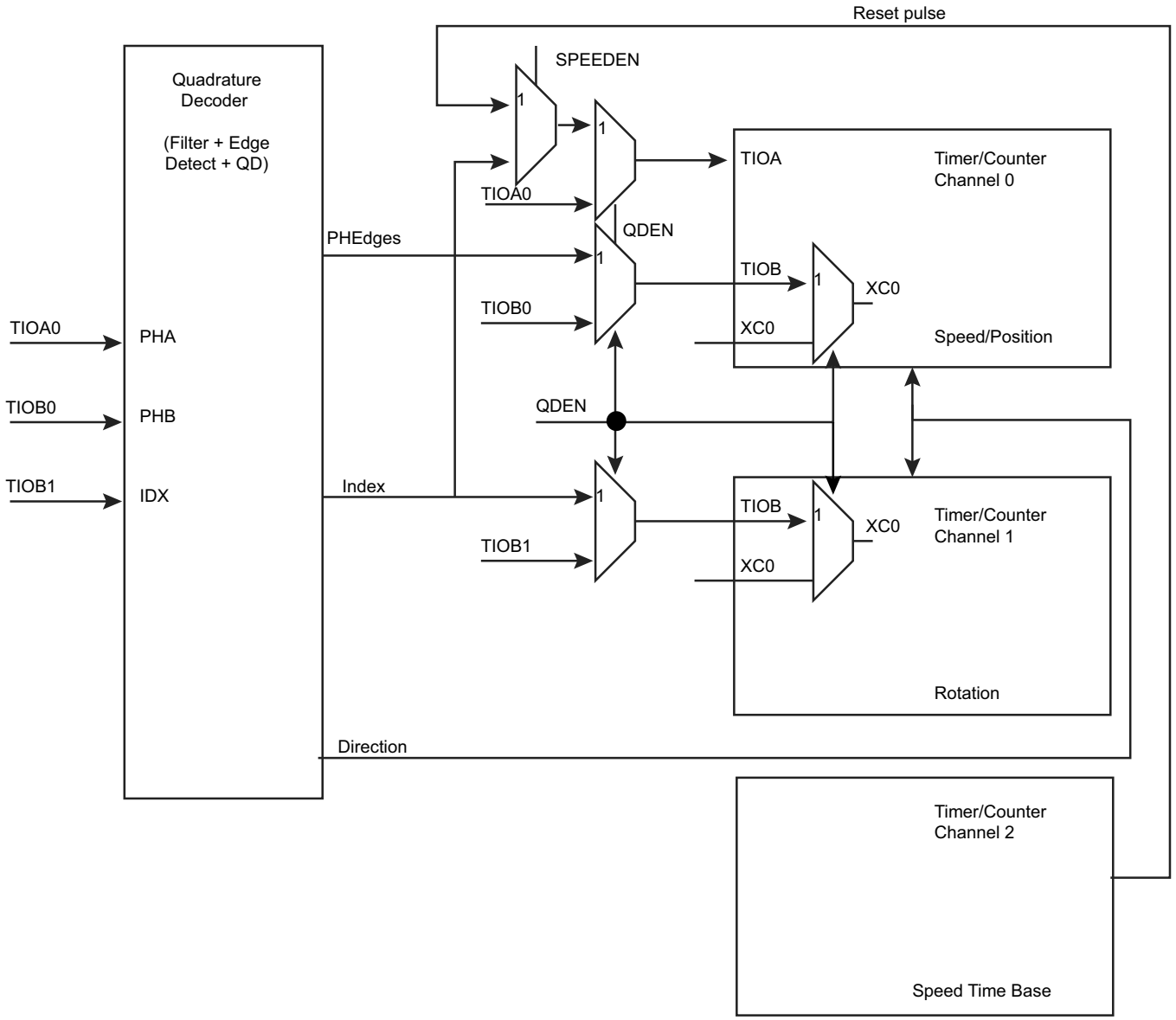
31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
–	–	–	–	–	–	–	–
15	14	13	12	11	10	9	8
CD							
7	6	5	4	3	2	1	0
CD							

- **CD: Clock Divisor**

0 = Baud Rate Clock is disabled

1 to 65,535 = MCK / (CD x 16)

Figure 32-16. Predefined Connection of the Quadrature Decoder with Timer Counters



32.6.15.2 Input Pre-processing

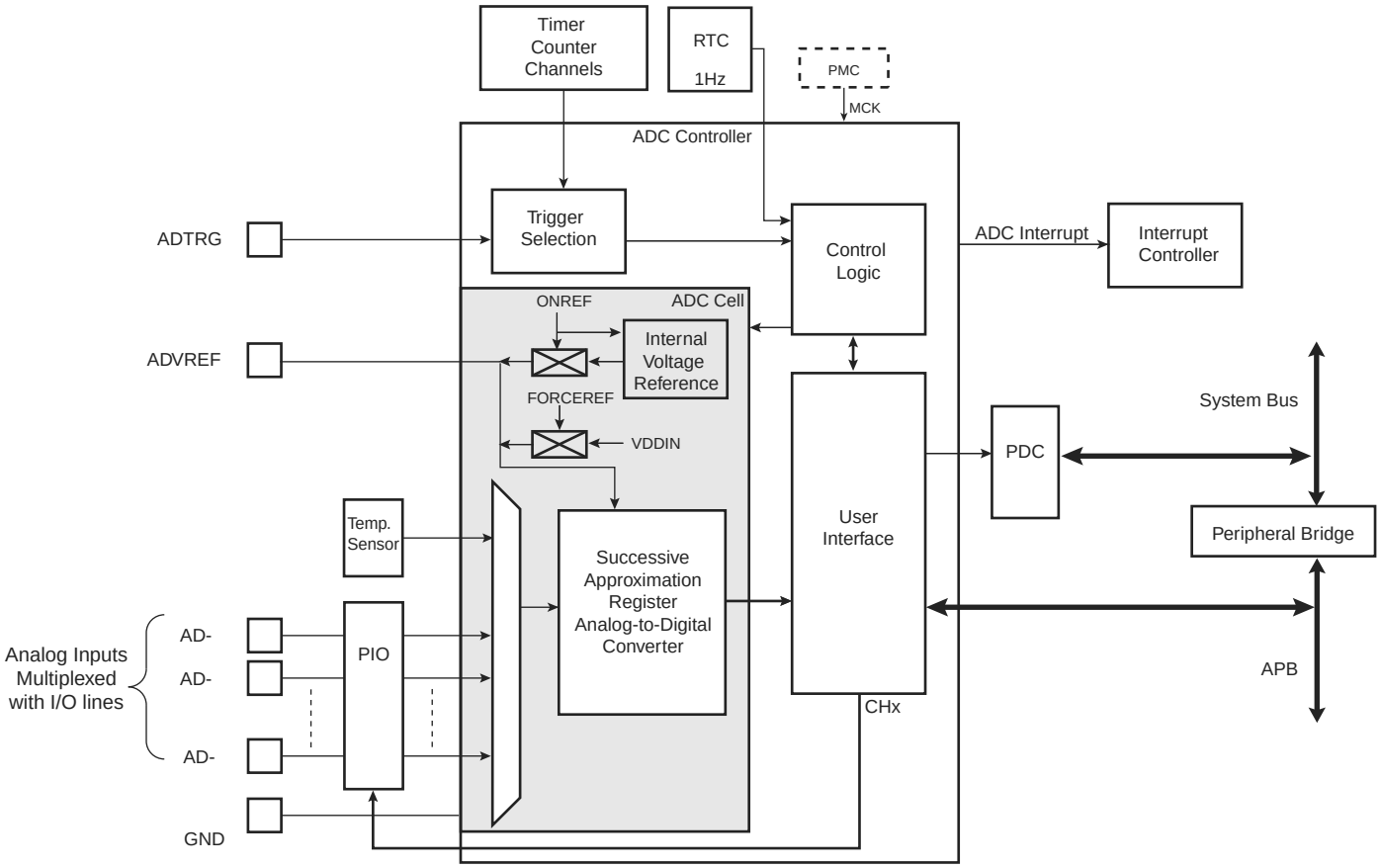
Input pre-processing consists of capabilities to take into account rotary sensor factors such as polarities and phase definition followed by configurable digital filtering.

Each input can be negated and swapping PHA, PHB is also configurable.

The MAXFILT field in the TC_BMR is used to configure a minimum duration for which the pulse is stated as valid. When the filter is active, pulses with a duration lower than MAXFILT +1 $\times$ $t_{peripheral\ clock}$ ns are not passed to down-stream logic.

34.3 Block Diagram

Figure 34-1. Analog-to-Digital Converter Block Diagram



34.4 Signal Description

Table 34-1. ADC Pin Description

Pin Name	Description
ADVREF	External Reference voltage
AD0 - AD16[1]	Analog input channels
ADTRG	External trigger

Note: 1. AD16 is not an actual pin but is internally connected to a temperature sensor.

34.7.10 ADC Interrupt Disable Register

Name: ADC_IDR

Address: 0x40038028

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	RXBUFF	ENDRX	COMPE	GOVRE	DRDY
23	22	21	20	19	18	17	16
EOCAL	–	–	–	TEMPCHG	–	–	EOC16
15	14	13	12	11	10	9	8
EOC15	EOC14	EOC13	EOC12	EOC11	EOC10	EOC9	EOC8
7	6	5	4	3	2	1	0
EOC7	EOC6	EOC5	EOC4	EOC3	EOC2	EOC1	EOC0

- **EOCx: End of Conversion Interrupt Disable x**
- **TEMPCHG: Temperature Change Interrupt Disable**
- **EOCAL: End of Calibration Sequence**
- **DRDY: Data Ready Interrupt Disable**
- **GOVRE: General Overrun Error Interrupt Disable**
- **COMPE: Comparison Event Interrupt Disable**
- **ENDRX: End of Receive Buffer Interrupt Disable**
- **RXBUFF: Receive Buffer Full Interrupt Disable**

0 = No effect.

1 = Disables the corresponding interrupt.

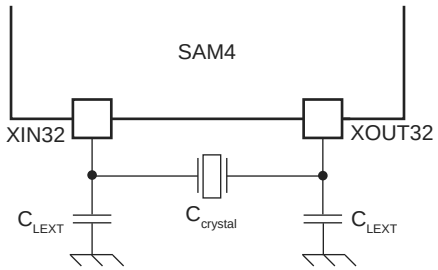
36.4.3 32.768 kHz Crystal Oscillator Characteristics

Table 36-21. 32.768 kHz Crystal Oscillator Characteristics

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
f_{OSC}	Operating Frequency	Normal mode with crystal				32.768	kHz
$V_{rip(VDDIO)}$	Supply Ripple Voltage (on VDDIO)	RMS value, 10 kHz to 10 MHz				30	mV
	Duty Cycle			40	50	60	%
t_{START}	Startup Time	R_S < 50 kΩ(1)	$C_{crystal}$ = 12.5 pF $C_{crystal}$ = 6 pF			900 300	ms
		R_S < 100 kΩ(1)	$C_{crystal}$ = 12.5 pF $C_{crystal}$ = 6 pF			1200 500	
I_{DDON}	Current consumption	R_S < 50 kΩ(1)	$C_{crystal}$ = 12.5 pF $C_{crystal}$ = 6 pF		550 380	1150 980	nA
		R_S < 100 kΩ(1)	$C_{crystal}$ = 12.5 pF $C_{crystal}$ = 6 pF		820 530	1600 1350	
P_{ON}	Drive level					0.1	µW
R_f	Internal Resistor	Between XIN32 and XOUT32			10		MΩ
C_{LEXT}	Maximum External Capacitor on XIN32 and XOUT32					20	pF
$C_{crystal}$	Allowed Crystal Capacitance Load	From crystal specification		6		12.5	pF
C_{para}	Internal Parasitic Capacitance			0.6	0.7	0.8	pF

Note: 1. R_S is the series resistor.

Figure 36-10. 32.768 kHz Crystal Oscillator Schematics



$C_{LEXT} = 2 \times (C_{crystal} - C_{para} - C_{PCB})$

where:

C_{PCB} is the capacitance of the printed circuit board (PCB) track layout from the crystal to the SAM4 pin.

36.4.4 32.768 kHz Crystal Characteristics

Table 36-22. Crystal Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
ESR	Equivalent Series Resistor (R_S)	Crystal @ 32.768 kHz		50	100	kΩ
C_m	Motional Capacitance	Crystal @ 32.768 kHz	0.6		3	fF
C_{SHUNT}	Shunt Capacitance	Crystal @ 32.768 kHz	0.6		2	pF
C_{LOAD}	Load Capacitance	Cystal @ 32.768 kHz Max external capacitor 20 pF	6		12.5	pF

36.4.6 3 to 20 MHz Crystal Characteristics

Table 36-24. Crystal Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
ESR	Equivalent Series Resistor (Rs)	Fundamental @ 3 MHz Fundamental @ 8 MHz Fundamental @ 16 MHz Fundamental @ 20 MHz			200 100 80 50	Ω
C_m	Motional Capacitance				8	fF
C_{SHUNT}	Shunt Capacitance				7	pF
C_{LOAD}	Load Capacitance	Max external capacitors: 17 pF	12.5		17.5	pF

36.4.7 3 to 20 MHz XIN Clock Input Characteristics in Bypass Mode

Table 36-25. XIN Clock Electrical Characteristics (In Bypass Mode)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$1/(t_{CPXIN})$	XIN Clock Frequency	3–20 MHz crystal oscillator is in Bypass mode			20	MHz
t_{CPXIN}	XIN Clock Period		50			ns
t_{CHXIN}	XIN Clock High Half-period		20			ns
t_{CLXIN}	XIN Clock Low Half-period		20			ns
t_{CLCH}	Rise Time		15			ns
t_{CHCL}	Fall Time		15			ns
V_{XIN_IL}	V_{XIN} Input Low-level Voltage		-0.3		$[0.8V:0.3 \times V_{DDIO}]$	V
V_{XIN_IH}	V_{XIN} Input High-level Voltage		$[2.0V:0.7 \times V_{DDIO}]$		V_{VDDIO} + 0.3V	V

Figure 36-12. XIN Clock Timing