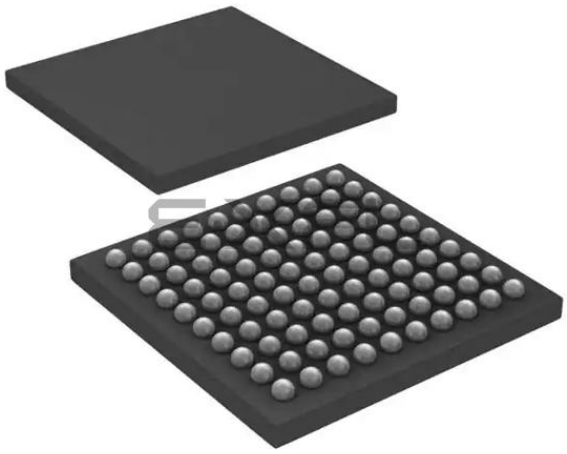


Welcome to E-XFL.COM

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	120MHz
Connectivity	EBI/EMI, I²C, IrDA, Memory Card, SPI, SSC, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	79
Program Memory Size	1MB (1M x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 16x12b; D/A 2x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-VFBGA
Supplier Device Package	100-VFBGA (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atsam4s16ca-cfn

4.1.6 100-ball VFBGA Pinout

Table 4-3. SAM4SD32/SD16/SA16/S16/S8/S4/S2 100-ball VFBGA Pinout

A1	ADVREF	C6	PC9	F1	VDDOUT	H6	PA12/PGMD0
A2	VDDPLL	C7	TMS/SWDIO/PB6	F2	PA18/PGMD6/AD1	H7	PA9/PGMM1
A3	PB9/PGMCK/XIN	C8	PA1/PGMEN1	F3	PA17/PGMD5/AD0	H8	VDDCORE
A4	PB8/XOUT	C9	PA0/PGMEN0	F4	GND	H9	PA6/PGMNOE
A5	JTAGSEL	C10	PC16	F5	GND	H10	PA5/PGMRDY
A6	DDP/PB11	D1	PB1/AD5	F6	PC26	J1	PA20/AD3/PGMD8
A7	DDM/PB10	D2	PC30/AD14	F7	PA4/PGMNCMD	J2	PC12/AD12
A8	PC20	D3	PC31	F8	PA28	J3	PA16/PGMD4
A9	PC19	D4	PC22	F9	TST	J4	PC6
A10	TDO/TRACESWO/PB5	D5	PC5	F10	PC8	J5	PA24/PGMD12
B1	GNDANA	D6	PA29	G1	PC15/AD11	J6	PA25/PGMD13
B2	PC25	D7	PA30	G2	PA19/PGMD7/AD2	J7	PA11/PGMM3
B3	PB14/DAC1	D8	GND	G3	PA21/AD8/PGMD9	J8	VDDCORE
B4	PB13/DAC0	D9	PC14	G4	PA15/PGMD3	J9	VDDCORE
B5	PC23	D10	PC11	G5	PC3	J10	TDI/PB4
B6	PC21	E1	VDDIN	G6	PA10/PGMM2	K1	PA23/PGMD11
B7	TCK/SWCLK/PB7	E2	PB3/AD7	G7	PC1	K2	PC0
B8	PA31	E3	PB2/AD6	G8	PC28	K3	PC7
B9	PC18	E4	GND	G9	NRST	K4	PA13/PGMD1
B10	PC17	E5	GND	G10	PA27/PGMD15	K5	PA26/PGMD14
C1	PB0/AD4	E6	GND	H1	PC13/AD10	K6	PC2
C2	PC29/AD13	E7	VDDIO	H2	PA22/AD9/PGMD10	K7	VDDIO
C3	PC24	E8	PC10	H3	PC27	K8	VDDIO
C4	ERASE/PB12	E9	PA2/PGMEN2	H4	PA14/PGMD2	K9	PA8/XOUT32/PGMM0
C5	VDDCORE	E10	PA3	H5	PC4	K10	PA7/XIN32/ PGMNVALID

6.3 Test Pin

The TST pin is used for JTAG Boundary Scan Manufacturing Test or Fast Flash programming mode of the SAM4S series. The TST pin integrates a permanent pull-down resistor of about 15 kΩ to GND, so that it can be left unconnected for normal operations. To enter fast programming mode, see Section 21. “Fast Flash Programming Interface (FFPI)”. For more on the manufacturing and test mode, refer to Section 13. “Debug and Test Features”.

6.4 NRST Pin

The NRST pin is bidirectional. It is handled by the on-chip reset controller and can be driven low to provide a reset signal to the external components or asserted low externally to reset the microcontroller. It will reset the Core and the peripherals except the Backup region (RTC, RTT and Supply Controller). There is no constraint on the length of the reset pulse and the reset controller can guarantee a minimum pulse length. The NRST pin integrates a permanent pull-up resistor to VDDIO of about 100 kΩ. By default, the NRST pin is configured as an input.

6.5 ERASE Pin

The ERASE pin is used to reinitialize the Flash content (and some of its NVM bits) to an erased state (all bits read as logic level 1). The ERASE pin and the ROM code ensure an in-situ reprogrammability of the Flash content without the use of a debug tool. When the security bit is activated, the ERASE pin provides the capability to reprogram the Flash content. It integrates a pull-down resistor of about 100 kΩ to GND, so that it can be left unconnected for normal operations.

This pin is debounced by SCLK to improve the glitch tolerance. To avoid unexpected erase at power-up, a minimum ERASE pin assertion time is required. This time is defined in Table 44-74 “AC Flash Characteristics”.

The ERASE pin is a system I/O pin and can be used as a standard I/O. At startup, the ERASE pin is not configured as a PIO pin. If the ERASE pin is used as a standard I/O, startup level of this pin must be low to prevent unwanted erasing. Refer to Section 11.2 “Peripheral Signal Multiplexing on I/O Lines” on page 51. Also, if the ERASE pin is used as a standard I/O output, asserting the pin to low does not erase the Flash.

6.6 Anti-tamper Pins/Low-power Tamper Detection

WKUP0 and WKUP1 generic wake-up pins can be used as anti-tamper pins. Anti-tamper pins detect intrusion, for example, into a housing box. Upon detection through a tamper switch, automatic, asynchronous and immediate clear of registers in the backup area will be performed. Anti-tamper pins can be used in all power modes (Backup/Wait/Sleep/Active). Anti-tampering events can be programmed so that half of the General Purpose Backup Registers (GPBR) are erased automatically. See "Supply Controller" section for further description.

RTCOUT0 and RTCOUT1 pins can be used to generate waveforms from the RTC in order to take advantage of the RTC inherent prescalers while the RTC is the only powered circuitry (low-power mode, Backup mode) or in any active mode. Entering backup or low-power modes does not affect the waveform generation outputs. Anti-tampering pin detection can be synchronized with this signal.

12.4.1.3 Core Registers

Figure 12-2. Processor Core Registers

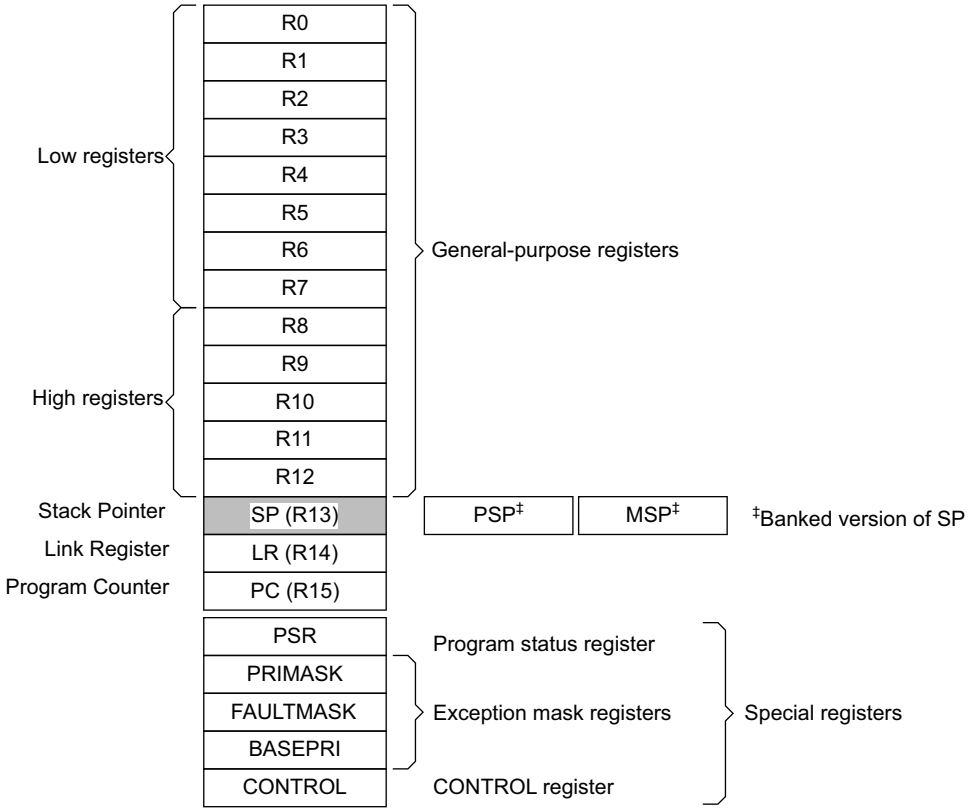


Table 12-2. Core Processor Registers

Register	Name	Access[1]	Required Privilege[2]	Reset
General-purpose registers	R0–R12	Read/Write	Either	Unknown
Stack Pointer	MSP	Read/Write	Privileged	See description
Stack Pointer	PSP	Read/Write	Either	Unknown
Link Register	LR	Read/Write	Either	0xFFFFFFFF
Program Counter	PC	Read/Write	Either	See description
Program Status Register	PSR	Read/Write	Privileged	0x01000000
Application Program Status Register	APSR	Read/Write	Either	0x00000000
Interrupt Program Status Register	IPSR	Read-only	Privileged	0x00000000
Execution Program Status Register	EPSR	Read-only	Privileged	0x01000000
Priority Mask Register	PRIMASK	Read/Write	Privileged	0x00000000
Fault Mask Register	FAULTMASK	Read/Write	Privileged	0x00000000
Base Priority Mask Register	BASEPRI	Read/Write	Privileged	0x00000000
Control Register	CONTROL	Read/Write	Privileged	0x00000000

Notes: 1. Describes access type during program execution in thread mode and Handler mode. Debug access can differ.
2. An entry of Either means privileged and unprivileged software can access the register.

12.9.1.6 System Control Register

Name: SCB_SCR

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
–	–	–	–	–	–	–	–
7	6	5	4	3	2	1	0
–	–	–	SEVONPEND	–	SLEEPDEEP	SLEEPONEXIT	–

- **SEVONPEND: Send Event on Pending Bit**

0: Only enabled interrupts or events can wake up the processor; disabled interrupts are excluded.

1: Enabled events and all interrupts, including disabled interrupts, can wake up the processor.

When an event or an interrupt enters the pending state, the event signal wakes up the processor from WFE. If the processor is not waiting for an event, the event is registered and affects the next WFE.

The processor also wakes up on execution of an SEV instruction or an external event.

- **SLEEPDEEP: Sleep or Deep Sleep**

Controls whether the processor uses sleep or deep sleep as its low power mode:

0: Sleep.

1: Deep sleep.

- **SLEEPONEXIT: Sleep-on-exit**

Indicates sleep-on-exit when returning from the Handler mode to the Thread mode:

0: Do not sleep when returning to Thread mode.

1: Enter sleep, or deep sleep, on return from an ISR.

Setting this bit to 1 enables an interrupt-driven application to avoid returning to an empty main application.

12.9.1.12 System Handler Control and State Register

Name: SCB_SHCSR

Access: Read/Write

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	–	USGFAULTENA	BUSFAULTENA	MEMFAULTENA

15	14	13	12	11	10	9	8
SVCALLPENDED	BUSFAULTPENDED	MEMFAULTPENDED	USGFAULTPENDED	SYSTICKACT	PENDSVACT	–	MONITORACT

7	6	5	4	3	2	1	0
SVCALLACT	–	–	–	USGFAULTACT	–	BUSFAULTACT	MEMFAULTACT

The SHCSR enables the system handlers, and indicates the pending status of the bus fault, memory management fault, and SVC exceptions; it also indicates the active status of the system handlers.

- **USGFAULTENA: Usage Fault Enable**

0: Disables the exception.

1: Enables the exception.

- **BUSFAULTENA: Bus Fault Enable**

0: Disables the exception.

1: Enables the exception.

- **MEMFAULTENA: Memory Management Fault Enable**

0: Disables the exception.

1: Enables the exception.

- **SVCALLPENDED: SVC Call Pending**

Read:

0: The exception is not pending.

1: The exception is pending.

Note: The user can write to these bits to change the pending status of the exceptions.

- **BUSFAULTPENDED: Bus Fault Exception Pending**

Read:

0: The exception is not pending.

1: The exception is pending.

Note: The user can write to these bits to change the pending status of the exceptions.

Table 12-38 shows the AP encodings that define the access permissions for privileged and unprivileged software.

Table 12-38. AP Encoding

AP[2:0]	Privileged Permissions	Unprivileged Permissions	Description
000	No access	No access	All accesses generate a permission fault
001	RW	No access	Access from privileged software only
010	RW	RO	Writes by unprivileged software generate a permission fault
011	RW	RW	Full access
100	Unpredictable	Unpredictable	Reserved
101	RO	No access	Reads by privileged software only
110	RO	RO	Read only, by privileged or unprivileged software
111	RO	RO	Read only, by privileged or unprivileged software

12.11.1.1 MPU Mismatch

When an access violates the MPU permissions, the processor generates a memory management fault, see “Exceptions and Interrupts” . The MMFSR indicates the cause of the fault. See “MMFSR: Memory Management Fault Status Subregister” for more information.

12.11.1.2 Updating an MPU Region

To update the attributes for an MPU region, update the MPU_RNR, MPU_RBAR and MPU_RASRs. Each register can be programed separately, or a multiple-word write can be used to program all of these registers. MPU_RBAR and MPU_RASR aliases can be used to program up to four regions simultaneously using an STM instruction.

12.11.1.3 Updating an MPU Region Using Separate Words

Simple code to configure one region:

```
; R1 = region number
; R2 = size/enable
; R3 = attributes
; R4 = address
LDR R0,=MPU_RNR         ; 0xE000ED98, MPU region number register
STR R1, [R0, #0x0]      ; Region Number
STR R4, [R0, #0x4]      ; Region Base Address
STRH R2, [R0, #0x8]     ; Region Size and Enable
STRH R3, [R0, #0xA]     ; Region Attribute
```

Disable a region before writing new region settings to the MPU, if the region being changed was previously enabled. For example:

```
; R1 = region number
; R2 = size/enable
; R3 = attributes
; R4 = address
LDR R0,=MPU_RNR          ; 0xE000ED98, MPU region number register
STR R1, [R0, #0x0]       ; Region Number
BIC R2, R2, #1           ; Disable
STRH R2, [R0, #0x8]      ; Region Size and Enable
STR R4, [R0, #0x4]       ; Region Base Address
STRH R3, [R0, #0xA]      ; Region Attribute
ORR R2, #1               ; Enable
```

15.5 Real-time Timer (RTT) User Interface

Table 15-1. Register Mapping

Offset	Register	Name	Access	Reset
0x00	Mode Register	RTT_MR	Read/Write	0x0000_8000
0x04	Alarm Register	RTT_AR	Read/Write	0xFFFF_FFFF
0x08	Value Register	RTT_VR	Read-only	0x0000_0000
0x0C	Status Register	RTT_SR	Read-only	0x0000_0000

27.6.10 Transfer Status Register

Name: PERIPH_PTSR

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
–	–	–	–	–	–	–	TXTEN
7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	RXTEN

- **RXTEN: Receiver Transfer Enable**

0: PDC receiver channel requests are disabled.

1: PDC receiver channel requests are enabled.

- **TXTEN: Transmitter Transfer Enable**

0: PDC transmitter channel requests are disabled.

1: PDC transmitter channel requests are enabled.

the slow clock until it reaches 0. At this time, the LOCK bit is set in PMC_SR and can trigger an interrupt to the processor. The user has to load the number of slow clock cycles required to cover the PLL transient time into the PLLCOUNT field.

The PLL clock can be divided by 2 by writing the PLLDIV2 (PLLADIV2, PLLBDIV2) bit in PMC_MCKR.

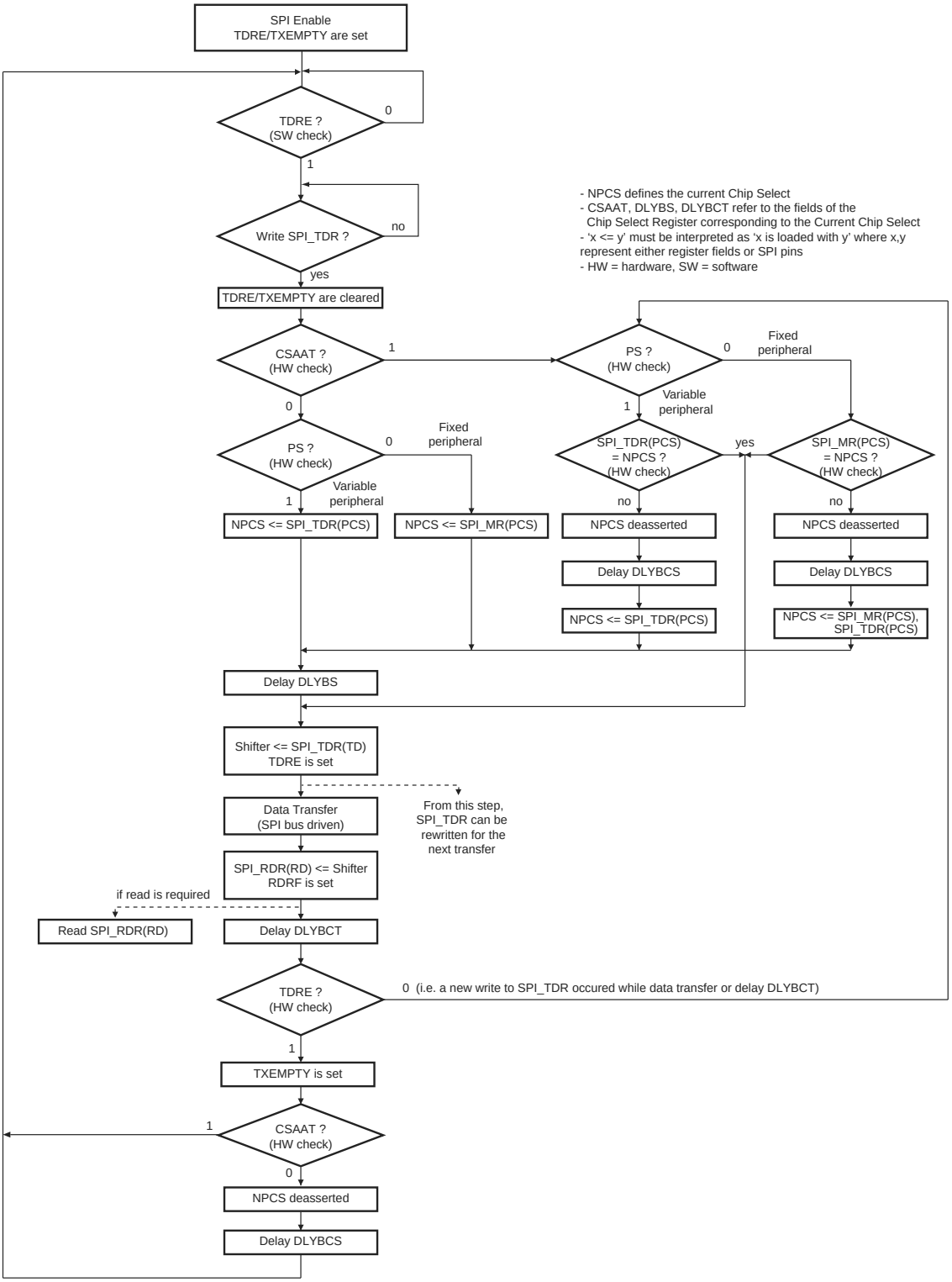
It is prohibited to change the 4/8/12 MHz fast RC oscillator or the main oscillator selection in CKGR_MOR while the master clock source is the PLL and the PLL reference clock is the fast RC oscillator.

The user must:

1. Switch on the main RC oscillator by writing a 1 to the CSS field of PMC_MCKR.
2. Change the frequency (MOSCRCF) or oscillator selection (MOSCSEL) in CKGR_MOR.
3. Wait for MOSCRCS (if frequency changes) or MOSCSELS (if oscillator selection changes) in PMC_SR.
4. Disable and then enable the PLL.
5. Wait for the LOCK flag in PMC_SR.
6. Switch back to the PLL by writing the appropriate value to the CSS field of PMC_MCKR.

33.7.3.2 Master Mode Flow Diagram

Figure 33-7. Master Mode Flow Diagram



If STTTO is performed, the counter clock is stopped until a first character is received. The idle state on RXD before the start of the frame does not provide a time-out. This prevents having to obtain a periodic interrupt and enables a wait of the end of frame when the idle state on RXD is detected.

If RETTO is performed, the counter starts counting down immediately from the value TO. This enables generation of a periodic interrupt so that a user time-out can be handled, for example when no key is pressed on a keyboard.

Figure 36-23 shows the block diagram of the Receiver Time-out feature.

Figure 36-23. Receiver Time-out Block Diagram

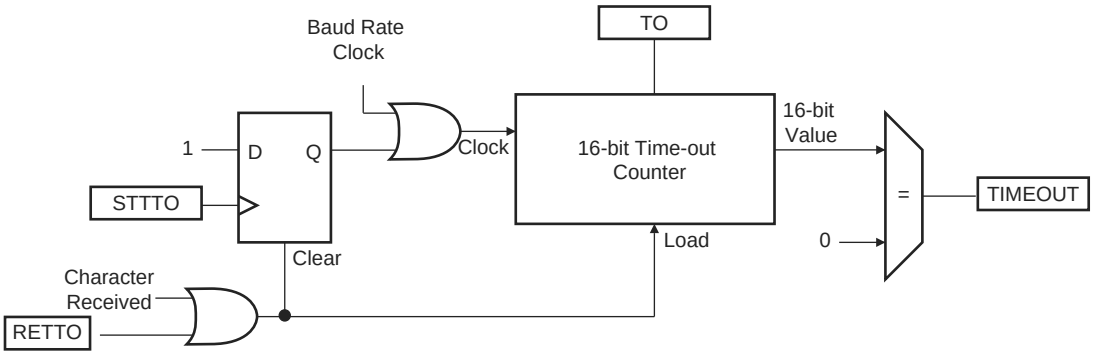


Table 36-10 gives the maximum time-out period for some standard baud rates.

Table 36-10. Maximum Time-out Period

Baud Rate (bit/s)	Bit Time (µs)	Time-out (ms)
600	1,667	109,225
1,200	833	54,613
2,400	417	27,306
4,800	208	13,653
9,600	104	6,827
14,400	69	4,551
19,200	52	3,413
28,800	35	2,276
38,400	26	1,704
56,000	18	1,170
57,600	17	1,138
200,000	5	328

36.6.3.12 Framing Error

The receiver is capable of detecting framing errors. A framing error happens when the stop bit of a received character is detected at level 0. This can occur if the receiver and the transmitter are fully desynchronized.

A framing error is reported on the FRAME bit of US_CSR. The FRAME bit is asserted in the middle of the stop bit as soon as the framing error is detected. It is cleared by writing a 1 to the RSTSTA bit in the US_CR.

- **PADV: Padding Value**

0: 0x00 value is used when padding data in write transfer.

1: 0xFF value is used when padding data in write transfer.

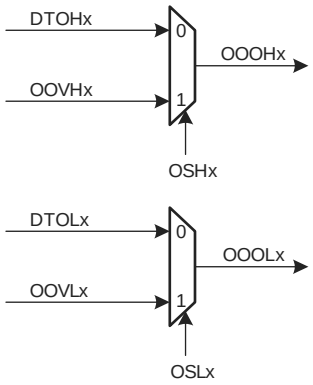
PADV may be only in manual transfer.

- **PDCMODE: PDC-oriented Mode**

0: Disables PDC transfer

1: Enables PDC transfer. In this case, UNRE and OVRE flags in the HSMCI Status Register (HSMCI_SR) are deactivated after the PDC transfer has been completed.

Figure 39-8. Override Output Selection



The fields OSHx and OSLx in the PWM Output Selection Register (PWM_OS) allow the outputs of the dead-time generator DTOHx and DTOLx to be overridden by the value defined in the fields OOVHx and OOVLx in the PWM Output Override Value Register (PWM_OOV).

The set registers PWM Output Selection Set Register (PWM_OSS) and PWM Output Selection Set Update Register (PWM_OSSUPD) enable the override of the outputs of a channel regardless of other channels. In the same way, the clear registers PWM Output Selection Clear Register (PWM_OSC) and PWM Output Selection Clear Update Register (PWM_OSCUPD) disable the override of the outputs of a channel regardless of other channels.

By using buffer registers PWM_OSSUPD and PWM_OSCUPD, the output selection of PWM outputs is done synchronously to the channel counter, at the beginning of the next PWM period.

By using registers PWM_OSS and PWM_OSC, the output selection of PWM outputs is done asynchronously to the channel counter, as soon as the register is written.

The value of the current output selection can be read in PWM_OS.

While overriding PWM outputs, the channel counters continue to run, only the PWM outputs are forced to user defined values.

39.7.8 PWM Interrupt Status Register 1

Name: PWM_ISR1

Address: 0x4002001C

Access: Read-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–

23	22	21	20	19	18	17	16
–	–	–	–	FCHID3	FCHID2	FCHID1	FCHID0

15	14	13	12	11	10	9	8
–	–	–	–	–	–	–	–

7	6	5	4	3	2	1	0
–	–	–	–	CHID3	CHID2	CHID1	CHID0

- **CHIDx: Counter Event on Channel x**

0: No new counter event has occurred since the last read of the PWM_ISR1.

1: At least one counter event has occurred since the last read of the PWM_ISR1.

- **FCHIDx: Fault Protection Trigger on Channel x**

0: No new trigger of the fault protection since the last read of the PWM_ISR1.

1: At least one trigger of the fault protection since the last read of the PWM_ISR1.

Note: Reading PWM_ISR1 automatically clears CHIDx and FCHIDx flags.

39.7.14 PWM Interrupt Disable Register 2

Name: PWM_IDR2

Address: 0x40020038

Access: Write-only

31	30	29	28	27	26	25	24
–	–	–	–	–	–	–	–
23	22	21	20	19	18	17	16
CMPU7	CMPU6	CMPU5	CMPU4	CMPU3	CMPU2	CMPU1	CMPU0
15	14	13	12	11	10	9	8
CMPM7	CMPM6	CMPM5	CMPM4	CMPM3	CMPM2	CMPM1	CMPM0
7	6	5	4	3	2	1	0
–	–	–	–	UNRE	TXBUFE	ENDTX	WRDY

- **WRDY: Write Ready for Synchronous Channels Update Interrupt Disable**
- **ENDTX: PDC End of TX Buffer Interrupt Disable**
- **TXBUFE: PDC TX Buffer Empty Interrupt Disable**
- **UNRE: Synchronous Channels Update Underrun Error Interrupt Disable**
- **CMPMx: Comparison x Match Interrupt Disable**
- **CMPUx: Comparison x Update Interrupt Disable**

39.7.41 PWM Channel Counter Register

Name: PWM_CCNTx [x=0..3]

Address: 0x40020214 [0], 0x40020234 [1], 0x40020254 [2], 0x40020274 [3]

Access: Read-only

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

Only the first 16 bits (channel counter size) are significant.

- **CNT: Channel Counter Register**

Channel counter value. This register is reset when:

- the channel is enabled (writing CHIDx in the PWM_ENA register).
- the channel counter reaches CPRD value defined in the PWM_CPRDx register if the waveform is left-aligned.

39.7.43 PWM Channel Dead Time Update Register

Name: PWM_DTUPDx [x=0..3]

Address: 0x4002021C [0], 0x4002023C [1], 0x4002025C [2], 0x4002027C [3]

Access: Write-only

31	30	29	28	27	26	25	24
DTLUPD							

23	22	21	20	19	18	17	16
DTLUPD							

15	14	13	12	11	10	9	8
DTHUPD							

7	6	5	4	3	2	1	0
DTHUPD							

This register can only be written if bits WPSWS4 and WPHWS4 are cleared in the PWM Write Protection Status Register.

This register acts as a double buffer for the DTH and DTL values. This prevents an unexpected waveform when modifying the dead-time values.

Only the first 12 bits (dead-time counter size) of fields DTHUPD and DTLUPD are significant.

- **DTHUPD: Dead-Time Value Update for PWMHx Output**

Defines the dead-time value for PWMHx output. This value must be defined between 0 and the value (CPRD – CDTY) (PWM_CPRDx and PWM_CDTYx). This value is applied only at the beginning of the next channel x PWM period.

- **DTLUPD: Dead-Time Value Update for PWMLx Output**

Defines the dead-time value for PWMLx output. This value must be defined between 0 and CDTY (PWM_CDTYx). This value is applied only at the beginning of the next channel x PWM period.

40.7.1 UDP Frame Number Register

Name: UDP_FRM_NUM

Address: 0x40034000

Access: Read-only

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

- **FRM_NUM[10:0]: Frame Number as Defined in the Packet Field Formats**

This 11-bit value is incremented by the host on a per frame basis. This value is updated at each start of frame.

Value updated at the SOF_EOP (Start of Frame End of Packet).

- **FRM_ERR: Frame Error**

This bit is set at SOF_EOP when the SOF packet is received containing an error.

This bit is reset upon receipt of SOF_PID.

- **FRM_OK: Frame OK**

This bit is set at SOF_EOP when the SOF packet is received without any error.

This bit is reset upon receipt of SOF_PID (Packet Identification).

In the Interrupt Status Register, the SOF interrupt is updated upon receiving SOF_PID. This bit is set without waiting for EOP.

Note: In the 8-bit Register Interface, FRM_OK is bit 4 of FRM_NUM_H and FRM_ERR is bit 3 of FRM_NUM_L.

Table 41-1. List of External Analog Data Inputs

Pin Name	Description
AD0..AD7	ACC Analog PLUS inputs
AD0..AD3	ACC Analog MINUS inputs
ADVREF	ADCVoltage reference

41.4 Pin Name List

Table 41-2. ACC Pin List

Pin Name	Description	Type
AD0..AD7	External analog data inputs	Input
TS	On-chip temperature sensor	Input
ADVREF	ADC voltage reference	Input
DAC0, DAC1	On-chip DAC inputs	Input

41.5 Product Dependencies

41.5.1 I/O Lines

The analog input pins (AD0–AD7 and DAC0–1) are multiplexed with digital functions (PIO) on the IO line. By writing the SELMINUS and SELPLUS fields in the ACC Mode Register (ACC_MR), the associated IO lines are set to Analog mode.

41.5.2 Power Management

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

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

41.5.3 Interrupt

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

Table 41-3. Peripheral IDs

Instance	ID
ACC	33

41.5.4 Fault Output

The ACC has the FAULT output connected to the FAULT input of PWM. Please refer to chapter Section 41.6.4 ”Fault Mode” and the implementation of the PWM in the product.

43.7 Digital-to-Analog Converter Controller (DACC) User Interface

Table 43-3. Register Mapping

Offset	Register	Name	Access	Reset
0x00	Control Register	DACC_CR	Write-only	–
0x04	Mode Register	DACC_MR	Read/Write	0x00000000
0x08–0x0C	Reserved	–	–	–
0x10	Channel Enable Register	DACC_CHER	Write-only	–
0x14	Channel Disable Register	DACC_CHDR	Write-only	–
0x18	Channel Status Register	DACC_CHSR	Read-only	0x00000000
0x1C	Reserved	–	–	–
0x20	Conversion Data Register	DACC_CDR	Write-only	–
0x24	Interrupt Enable Register	DACC_IER	Write-only	–
0x28	Interrupt Disable Register	DACC_IDR	Write-only	–
0x2C	Interrupt Mask Register	DACC_IMR	Read-only	0x00000000
0x30	Interrupt Status Register	DACC_ISR	Read-only	0x00000000
0x34–0x90	Reserved	–	–	–
0x94	Analog Current Register	DACC_ACR	Read/Write	0x00000000
0x98–0xE0	Reserved	–	–	–
0xE4	Write Protection Mode Register	DACC_WPMR	Read/Write	0x00000000
0xE8	Write Protection Status Register	DACC_WPSR	Read-only	0x00000000
0xEC–0xFC	Reserved	–	–	–