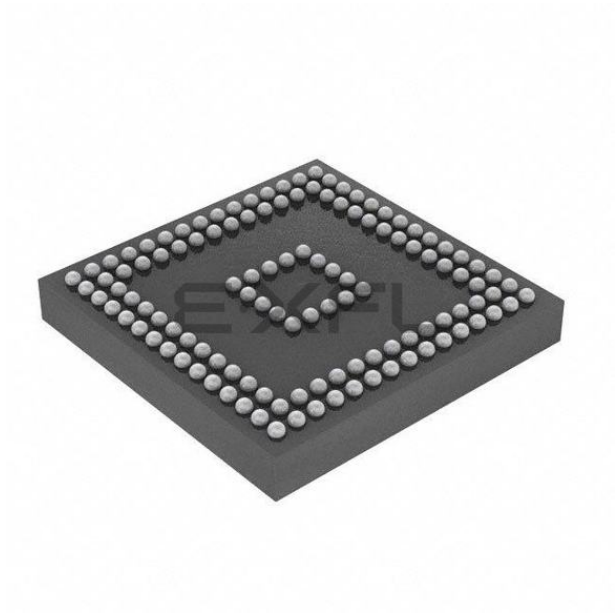


**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

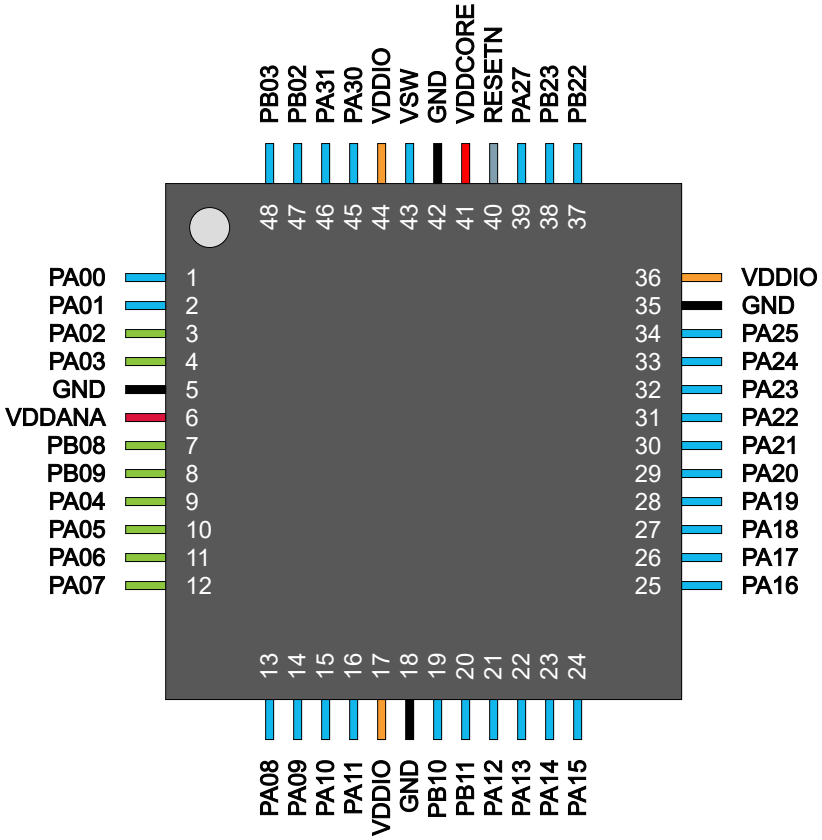
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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 | 99 |
| 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 32x12b; D/A 2x12b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 120-TFBGA |
| Supplier Device Package | 120-TFBGA (8x8) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/atsamd51p19a-ctut |

# 4. Pinout

## 4.1 Pin Count 48 (G)



### 16.8.1 Reset Cause

**Name:** RCAUSE
**Offset:** 0x00
**Property:** –

When a Reset occurs, the bit corresponding to the Reset source is set to '1' and all other bits are written to '0'.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BACKUP | SYST | WDT | EXT | NVM | BOD33 | BOD12 | POR |
| Access | R | R | R | R | R | R | R | R |
| Reset | x | x | x | x | x | x | x | x |

**Bit 7 – BACKUP** Backup Reset
This bit is set if either a Backup or Hibernate Reset has occurred. Refer to BKUPEXIT register to identify the source of the Backup Reset.

**Bit 6 – SYST** System Reset Request
This bit is set if a System Reset Request has occurred. Refer to the Cortex processor documentation for more details.

**Bit 5 – WDT** Watchdog Reset
This bit is set if a Watchdog Timer Reset has occurred.

**Bit 4 – EXT** External Reset
This bit is set if an external Reset has occurred.

**Bit 3 – NVM** NVM Reset
This bit is set if an NVM Reset has occurred.

**Bit 2 – BOD33** Brown Out 33 Detector Reset
This bit is set if a BOD33 Reset has occurred.

**Bit 1 – BOD12** Brown Out 12 Detector Reset
This bit is set if a BOD12 Reset has occurred.

**Bit 0 – POR** Power On Reset
This bit is set if a POR has occurred.

| Value | Description |
|---|---|
| 0 | No BOD12 detection. |
| 1 | BOD12 has detected that the core power supply is going below the BOD12 reference value. |

**Bit 3 – BOD12RDY** BOD12 Ready
The BOD12 can be enabled at start-up from NVM User Row.

| Value | Description |
|---|---|
| 0 | BOD12 is not ready. |
| 1 | BOD12 is ready. |

**Bit 2 – B33SRDY** BOD33 Synchronization Ready

| Value | Description |
|---|---|
| 0 | BOD33 synchronization is ongoing. |
| 1 | BOD33 synchronization is complete. |

**Bit 1 – BOD33DET** BOD33 Detection

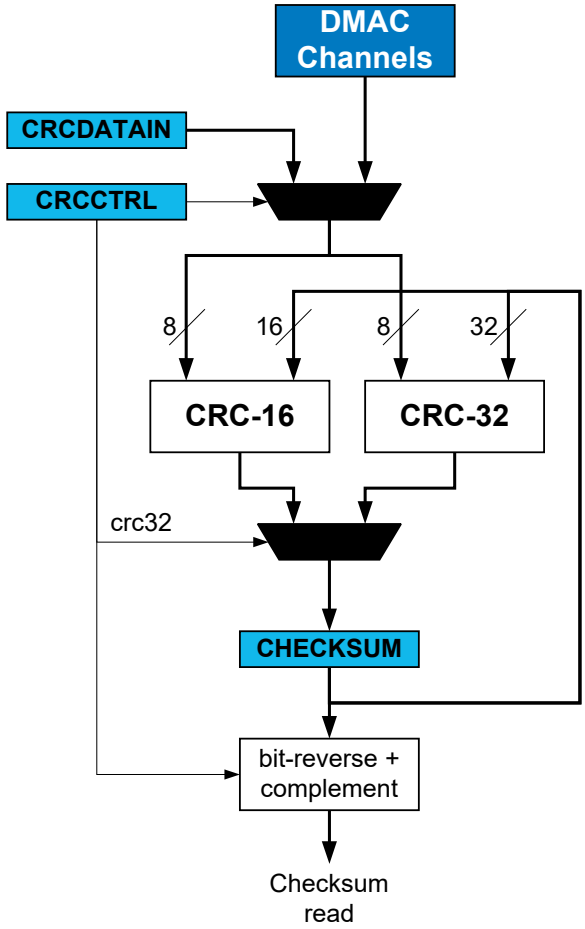
| Value | Description |
|---|---|
| 0 | No BOD33 detection. |
| 1 | BOD33 has detected that the I/O power supply is going below the BOD33 reference value. |

**Bit 0 – BOD33RDY** BOD33 Ready
The BOD33 can be enabled at start-up from NVM User Row.

| Value | Description |
|---|---|
| 0 | BOD33 is not ready. |
| 1 | BOD33 is ready. |

**Related Links**
9.4 NVM User Page Mapping

**Figure 22-19. CRC Generator Block Diagram**




**CRC on DMA data** CRC-16 or CRC-32 calculations can be performed on data passing through any DMA channel. Once a DMA channel is selected as the source, the CRC engine will continuously generate the CRC on the data passing through the DMA channel. The checksum is available for readout once the DMA transaction is completed or aborted. A CRC can also be generated on SRAM, Flash, or I/O memory by passing these data through a DMA channel. If the latter is done, the destination register for the DMA data can be the data input (CRCDATAIN) register in the CRC engine.

**CRC using the I/O interface** Before using the CRC engine with the I/O interface, the application must set the CRC Beat Size bits in the CRC Control register (CRCCTRL.CRCBEATSIZE). 8/16/32-bit bus transfer type can be selected.

CRC can be performed on any data by loading them into the CRC engine using the CPU and writing the data to the CRCDATAIN register. Using this method, an arbitrary number of bytes can be written to the register by the CPU, and CRC is done continuously for each byte. This means if a 32-bit data is written to the CRCDATAIN register the CRC engine takes four cycles to calculate the CRC. The CRC complete is signaled by a set CRCBUSY bit in the CRCSTATUS register. New data can be written only when CRCBUSY flag is not set.

#### 22.6.3.9 Memory CRC Generation

When enabled, it is possible to automatically calculate a memory block checksum. When the channel is enabled and the descriptor is fetched, the CRC Checksum register (CRCCHKSUM) is reloaded with the

| Offset | Name | Bit Pos. | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 31:24 | | | | | | | | |
| 0xF0 | PEFTSH | 7:0 | RUD[7:0] | | | | | | | |
| | | 15:8 | RUD[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xF4 | PEFRSH | 7:0 | RUD[7:0] | | | | | | | |
| | | 15:8 | RUD[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0xF8 ... 0xFF | Reserved | | | | | | | | | |
| 0x0100 | OTLO | 7:0 | TXO[7:0] | | | | | | | |
| | | 15:8 | TXO[15:8] | | | | | | | |
| | | 23:16 | TXO[23:16] | | | | | | | |
| | | 31:24 | TXO[31:24] | | | | | | | |
| 0x0104 | OTHI | 7:0 | TXO[7:0] | | | | | | | |
| | | 15:8 | TXO[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0108 | FT | 7:0 | FTX[7:0] | | | | | | | |
| | | 15:8 | FTX[15:8] | | | | | | | |
| | | 23:16 | FTX[23:16] | | | | | | | |
| | | 31:24 | FTX[31:24] | | | | | | | |
| 0x010C | BCFT | 7:0 | BFTX[7:0] | | | | | | | |
| | | 15:8 | BFTX[15:8] | | | | | | | |
| | | 23:16 | BFTX[23:16] | | | | | | | |
| | | 31:24 | BFTX[31:24] | | | | | | | |
| 0x0110 | MFT | 7:0 | MFTX[7:0] | | | | | | | |
| | | 15:8 | MFTX[15:8] | | | | | | | |
| | | 23:16 | MFTX[23:16] | | | | | | | |
| | | 31:24 | MFTX[31:24] | | | | | | | |
| 0x0114 | PFT | 7:0 | PFTX[7:0] | | | | | | | |
| | | 15:8 | PFTX[15:8] | | | | | | | |
| | | 23:16 | | | | | | | | |
| | | 31:24 | | | | | | | | |
| 0x0118 | BFT64 | 7:0 | NFTX[7:0] | | | | | | | |
| | | 15:8 | NFTX[15:8] | | | | | | | |
| | | 23:16 | NFTX[23:16] | | | | | | | |
| | | 31:24 | NFTX[31:24] | | | | | | | |
| 0x011C | TBFT127 | 7:0 | NFTX[7:0] | | | | | | | |
| | | 15:8 | NFTX[15:8] | | | | | | | |
| | | 23:16 | NFTX[23:16] | | | | | | | |
| | | 31:24 | NFTX[31:24] | | | | | | | |
| 0x0120 | TBFT255 | 7:0 | NFTX[7:0] | | | | | | | |
| | | 15:8 | NFTX[15:8] | | | | | | | |

**Digital Filter Selection**

The digital filter selection can be changed from the filter selection register DPLLnCTRLB.FILTER. The DPLL digital filter coefficients are automatically adjusted in order to provide a good compromise between stability and jitter. For more information, refer to DPLLnCTRLB.

**Sigma-Delta DCO Filter Selection**

The sigma-delta DAC low pass filter can be controlled and adjusted from the DCO filter selection register DPLLnCTRLB.DCOFILTER[2:0]. For more information, refer to DPLLnCTRLB.

**Related Links**

14. GCLK - Generic Clock Controller

### 28.6.6 DMA Operation

Not applicable.

### 28.6.7 Interrupts

The OSCCTRL has the following interrupt sources:

- XOSCRDY - Multipurpose Crystal Oscillator Ready: A 0-to-1” transition on the STATUS.XOSCRDY bit is detected
- CLKFAIL - Clock Failure . A “0-to-1” transition on the STATUS.CLKFAIL bit is detected.
- DFLLRDY - DFLL48m Ready: A “0-to-1” transition on the STATUS.DFLLRDY bit is detected
- DPLLnLOCKR - DPLLn Lock Rise: A “0-to-1” transition on the STATUS.DPLLnLOCKR bit is detected
- DPLLnLOCKF - DPLLn Lock Fall: A “0-to-1” transition on the STATUS.DPLLnLOCKF bit is detected
- DPLLnLTTO - DPLLn Lock Timer Time-out: A “0-to-1” transition on the STATUS.DPLLnLTTO bit is detected
- DPLLnLDRTO - DPLLn Loop Divider Ratio Update Complete. A “0-to-1” transition on the STATUS.DPLLnLDRTO bit is detected

Each interrupt source has an interrupt flag associated with it. The interrupt flag in the Interrupt Flag Status and Clear register (INTFLAG) is set when the interrupt condition occurs. Each interrupt can be individually enabled by writing a one to the corresponding bit in the Interrupt Enable Set register (INTENSET), and disabled by writing a one to the corresponding bit in the Interrupt Enable Clear register (INTENCLR). An interrupt request is generated when the interrupt flag is set and the corresponding interrupt is enabled. The interrupt request remains active until the interrupt flag is cleared, the interrupt is disabled or the OSCCTRL is reset. INTFLAG register for details on how to clear interrupt flags.

The OSCCTRL has one common interrupt request line for all the interrupt sources. The user must read the INTFLAG register to determine which interrupt condition is present.

Note that interrupts must be globally enabled for interrupt requests to be generated.

### 28.6.8 Events

The CFD can generate the following output event:

- Clock Failure (CLKFAIL): Generated when the Clock Failure status bit is set in the Status register (STATUS.CLKFAIL). The CFD event is not generated when the Clock Switch bit (STATUS.CLKSW) in the Status register is set.

Writing a '1' to an Event Output bit in the Event Control register (EVCTRL.CFDEO) enables the CFD output event. Writing a '0' to this bit disables the CFD output event. Refer to the *Event System* chapter for details on configuring the event system.

### 31.4.5 Interrupts

The interrupt request line is connected to the Interrupt Controller. Using the EVSYS interrupts requires the interrupt controller to be configured first. Refer to *Nested Vector Interrupt Controller* for details.

**Related Links**

10.2 Nested Vector Interrupt Controller

### 31.4.6 Events

Not applicable.

### 31.4.7 Debug Operation

When the CPU is halted in debug mode, this peripheral will continue normal operation. If the peripheral is configured to require periodical service by the CPU through interrupts or similar, improper operation or data loss may result during debugging. This peripheral can be forced to halt operation during debugging.

### 31.4.8 Register Access Protection

Registers with write-access can be optionally write-protected by the Peripheral Access Controller (PAC), except for the following:

- Channel Pending Interrupt (INTPEND)
- Channel n Interrupt Flag Status and Clear (CHINTFLAGn)

**Note:** Optional write-protection is indicated by the "PAC Write-Protection" property in the register description.

Write-protection does not apply for accesses through an external debugger.

### 31.4.9 Analog Connections

Not applicable.

## 31.5 Functional Description

### 31.5.1 Principle of Operation

The Event System consists of several channels which route the internal events from peripherals (generators) to other internal peripherals or I/O pins (users). Each event generator can be selected as source for multiple channels, but a channel cannot be set to use multiple event generators at the same time.

A channel path can be configured in asynchronous, synchronous or resynchronized mode of operation. The mode of operation must be selected based on the requirements of the application.

When using synchronous or resynchronized path, the Event System includes options to transfer events to users when rising, falling or both edges are detected on event generators.

For further details, refer to the Channel Path section of this chapter.

**Related Links**

31.5.2.6 Channel Path

**Protocol T=0**

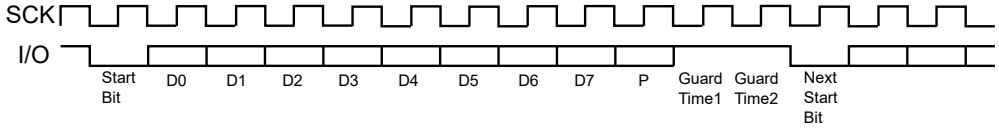
In T=0 protocol, a character is made up of:

- one start bit,
- eight data bits,
- one parity bit
- and one guard time, which lasts two bit times.

The transfer is synchronous (CTRLA.CMODE=1). The transmitter shifts out the bits and does not drive the I/O line during the guard time. Additional guard time can be added by programming the Guard Time (CTRLC.GTIME).

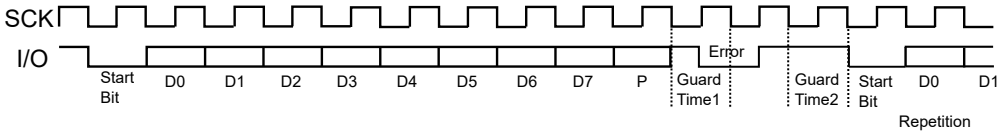
If no parity error is detected, the I/O line remains during the guard time and the transmitter can continue with the transmission of the next character, as shown in the figure below.

**Figure 34-18. T=0 Protocol without Parity Error**




If a parity error is detected by the receiver, it drives the I/O line to 0 during the guard time, as shown in the next figure. This error bit is also named NACK, for Non Acknowledge. In this case, the character lasts 1 bit time more, as the guard time length is the same and is added to the error bit time, which lasts 1 bit time.

**Figure 34-19. T=0 Protocol with Parity Error**




When the USART is the receiver and it detects a parity error, the parity error bit in the Status Register (STATUS.PERR) is set and the character is not written to the receive FIFO.

**Receive Error Counter**

The receiver also records the total number of errors (receiver parity errors and NACKs from the remote transmitter) up to a maximum of 255. This can be read in the Receive Error Count (RXERRCNT) register. RXERRCNT is automatically cleared on read.

**Receive NACK Inhibit**

The receiver can also be configured to inhibit error generation. This can be achieved by setting the Inhibit Not Acknowledge (CTRLC.INACK) bit. If CTRLC.INACK is 1, no error signal is driven on the I/O line even if a parity error is detected. Moreover, if CTRLC.INACK is set, the erroneous received character is stored in the receive FIFO, and the STATUS.PERR bit is set. Inhibit not acknowledge (CTRLC.INACK) takes priority over disable successive receive NACK (CTRLC.DSNACK).

**Transmit Character Repetition**

When the USART is transmitting a character and gets a NACK, it can automatically repeat the character before moving on to the next character. Repetition is enabled by writing the Maximum Iterations register (CTRLC.MAXITER) to a non-zero value. The USART repeats the character the number of times specified in CTRLC.MAXITER.

### 35.8.4 Baud Rate

**Name:** BAUD
**Offset:** 0x0C
**Reset:** 0x00
**Property:** PAC Write-Protection, Enable-Protected

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BAUD[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 7:0 – BAUD[7:0]** Baud Register
These bits control the clock generation, as described in the *SERCOM Clock Generation – Baud-Rate Generator*.

**Related Links**

33.6.2.3 Clock Generation – Baud-Rate Generator

33.6.2.3.1 Asynchronous Arithmetic Mode BAUD Value Selection

### 38.6.3 Host Operations

This section gives an overview of the USB module Host operation during normal transactions. For more details on general USB and USB protocol, refer to Universal Serial Bus Specification revision 2.1.

#### 38.6.3.1 Device Detection and Disconnection

Prior to device detection the software must set the VBUS is OK bit (CTRLB.VBUSOK) register when the VBUS is available. This notifies the USB host that USB operations can be started. When the bit CTRLB.VBUSOK is zero and even if the USB HOST is configured and enabled, host operation is halted. Setting the bit CTRLB.VBUSOK will allow host operation when the USB is configured.

The Device detection is managed by the software using the Line State field in the Host Status (STATUS.LINESTATE) register. The device connection is detected by the host controller when DP or DM is pulled high, depending of the speed of the device.

The device disconnection is detected by the host controller when both DP and DM are pulled down using the STATUS.LINESTATE registers.

The Device Connection Interrupt bit (INTFLAG.DCONN) is set if a device connection is detected.

The Device Disconnection Interrupt bit (INTFLAG.DDISC) is set if a device disconnection is detected.

#### 38.6.3.2 Host Terminology

In host mode, the term pipe is used instead of endpoint. A host pipe corresponds to a device endpoint, refer to "Universal Serial Bus Specification revision 2.1." for more information.

#### 38.6.3.3 USB Reset

The USB sends a USB reset signal when the user writes a one to the USB Reset bit (CTRLB.BUSRESET). When the USB reset has been sent, the USB Reset Sent Interrupt bit in the INTFLAG (INTFLAG.RST) is set and all pipes will be disabled.

If the bus was previously in a suspended state (i.e., the Start of Frame Generation Enable bit (CTRLB.SOFE) is zero), the USB will switch it to the Resume state, causing the bus to asynchronously set the Host Wakeup Interrupt flag (INTFLAG.WAKEUP). The CTRLB.SOFE bit will be set in order to generate SOFs immediately after the USB reset.

During USB reset the following registers are cleared:

- All Host Pipe Configuration register (PCFG)
- Host Frame Number register (FNUM)
- Interval for the Bulk-Out/Ping transaction register (BINTERVAL)
- Host Start-of-Frame Control register (HSOFC)
- Pipe Interrupt Enable Clear/Set register (PINTENCLR/SET)
- Pipe Interrupt Flag register (PINTFLAG)
- Pipe Freeze bit in Pipe Status register (PSTATUS.FREEZE)

After the reset the user should check the Speed Status field in the Status register (STATUS.SPEED) to find out the current speed according to the capability of the peripheral.

#### 38.6.3.4 Pipe Configuration

Pipe data can be placed anywhere in the RAM. The USB controller accesses these pipes directly through the AHB master (built-in DMA) with the help of the pipe descriptors. The base address of the pipe descriptors needs to be written in the Descriptor Address register (DESCADD) by the user. Refer also to 38.8.7.1 Pipe Descriptor Structure.

Before using a pipe, the user should configure the direction and type of the pipe in Type of Pipe field in the Host Pipe Configuration register (PCFG.PTYPE). The pipe descriptor registers should be initialized to

**Bit 17 – MRAFE** Message RAM Access Failure Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 16 – TSWE** Timestamp Wraparound Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 15 – TEFLE** Tx Event FIFO Event Lost Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 14 – TEFFE** Tx Event FIFO Full Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 13 – TEFWE** Tx Event FIFO Watermark Reached Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 12 – TEFNE** Tx Event FIFO New Entry Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 11 – TFEE** Tx FIFO Empty Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 10 – TCFE** Transmission Cancellation Finished Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

**Bit 9 – TCE** Transmission Completed Interrupt Enable

| Value | Description |
|---|---|
| 0 | Interrupt disabled. |
| 1 | Interrupt enabled. |

### 39.8.21 Standard ID Filter Configuration

**Name:** SIDFC
**Offset:** 0x84
**Reset:** 0x00000000
**Property:** Write-restricted

This register is write-restricted and only writable if bit fields CCCR.CCE = 1 and CCCR.INIT = 1.

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| Access | | | | | | | | |
| Reset | | | | | | | | |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | LSS[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|---|
| | FLSSA[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 |
|---|---|---|---|---|---|---|---|---|
| | FLSSA[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 23:16 – LSS[7:0]** List Size Standard

| Value | Description |
|---|---|
| 0 | No standard Message ID filter. |
| 1 - 128 | Number of standard Message ID filter elements. |
| > 128 | Values greater than 128 are interpreted as 128. |

**Bits 15:0 – FLSSA[15:0]** Filter List Standard Start Address
Start address of standard Message ID filter list. When the CAN module addresses the Message RAM it addresses 32-bit words, not single bytes. The configurable start addresses are 32-bit word addresses, i.e. only bits 15 to 2 are evaluated, the two least significant bits are ignored. Bits 1 to 0 will always be read back as “00”.

### 41.8.3 LUT Control x

**Name:** LUTCTRL
**Offset:** 0x08 + n*0x04 [n=0..3]
**Reset:** 0x00000000
**Property:** PAC Write-Protection

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 |
|---|---|---|---|---|---|---|---|---|
| | TRUTH[7:0] | | | | | | | |
| Access | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Bit | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|
| | | LUTEO | LUTEI | INVEI | INSELx[3:0] | | | |
| Access | | R/W | R/W | R/W | R/W | R/W | R/W | R/W |
| Reset | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

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

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | EDGESEL | | FILTSEL[1:0] | | | | ENABLE | |
| Access | R/W | | R/W | R/W | | | R/W | |
| Reset | 0 | | 0 | 0 | | | 0 | |

**Bits 31:24 – TRUTH[7:0]** Truth Table
These bits define the value of truth logic as a function of inputs IN[2:0].

**Bit 22 – LUTEO** LUT Event Output Enable

| Value | Description |
|---|---|
| 0 | LUT event output is disabled. |
| 1 | LUT event output is enabled. |

**Bit 21 – LUTEI** LUT Event Input Enable

| Value | Description |
|---|---|
| 0 | LUT incoming event is disabled. |
| 1 | LUT incoming event is enabled. |

**Bit 20 – INVEI** Inverted Event Input Enable

| Value | Description |
|---|---|
| 0 | Incoming event is not inverted. |
| 1 | Incoming event is inverted. |

**Bit 7 – EDGESEL** Edge Selection

### 47.8.2 Control B

**Name:** CTRLB
**Offset:** 0x01
**Reset:** 0x02
**Property:** PAC Write-Protection

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | REFSEL[1:0] | | DIFF |
| Access | | | | | | R/W | R/W | R/W |
| Reset | | | | | | 0 | 1 | 0 |

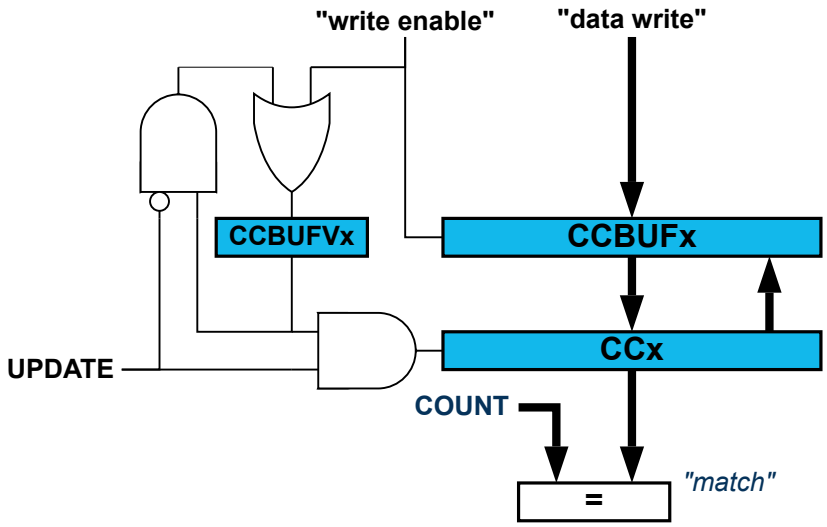
**Bits 2:1 – REFSEL[1:0]** Reference Selection
This bit field selects the Reference Voltage for both DACs.

| Value | Name | Description |
|---|---|---|
| 0x0 | VREFAU | Unbuffered external voltage reference (not buffered in DAC, direct connection) |
| 0x1 | VDDANA | Voltage supply |
| 0x2 | VREFAB | Buffered external voltage reference (buffered in DAC) |
| 0x3 | INTREF | Internal bandgap reference |

**Bit 0 – DIFF** Differential Mode Enable
This bit defines the conversion mode for both DACs.

| Value | Description |
|---|---|
| 0 | Single mode |
| 1 | Differential mode |

**Figure 48-7. Compare Channel Double Buffering**



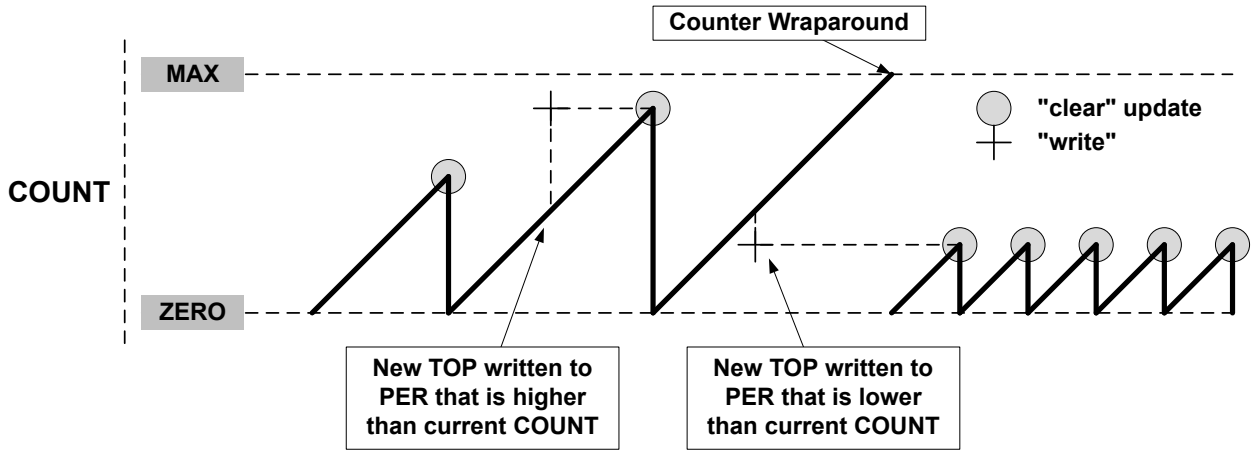

Both the registers (PER/CCx) and corresponding buffer registers (PERBUF/CCBUFx) are available in the I/O register map, and the double buffering feature is not mandatory. The double buffering is disabled by writing a '1' to CTRLBSET.LUPD.

**Note:** In NFRQ, MFRQ or PWM down-counting counter mode (CTRLBSET.DIR=1), when double buffering is enabled (CTRLBCLR.LUPD=1), PERBUF register is continously copied into the PER independently of update conditions.

### Changing the Period

The counter period can be changed by writing a new TOP value to the Period register (PER or CC0, depending on the waveform generation mode), which is available in 8-bit mode. Any period update on registers (PER or CCx) is effective after the synchronization delay.

**Figure 48-8. Unbuffered Single-Slope Up-Counting Operation**



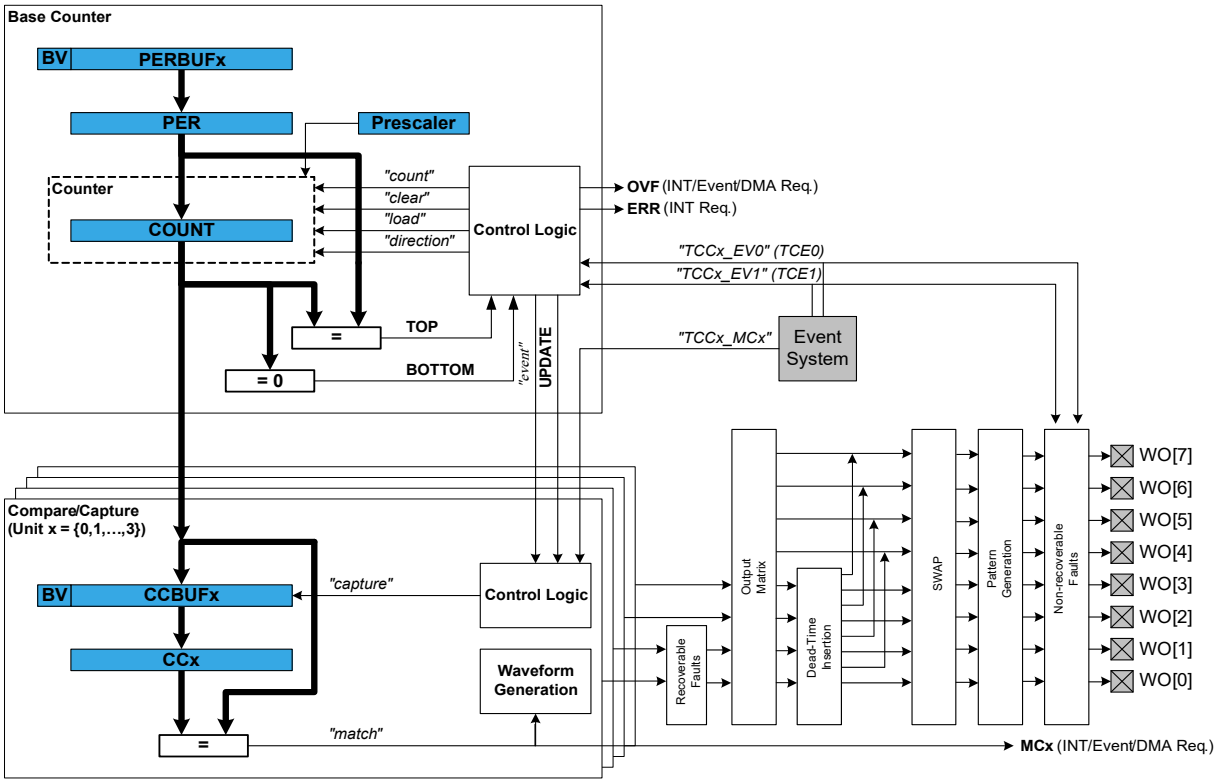

A counter wraparound can occur in any operation mode when up-counting without buffering, see Figure 48-8.

COUNT and TOP are continuously compared, so when a new TOP value that is lower than current COUNT is written to TOP, COUNT will wrap before a compare match.

  - Two non-recoverable fault sources
  - Debugger can be a source of non-recoverable fault
- Input events:
  - Two input events (EVx) for counter
  - One input event (MCx) for each channel
- Output events:
  - Three output events (Count, Re-Trigger and Overflow) are available for counter
  - One Compare Match/Input Capture event output for each channel
- Interrupts:
  - Overflow and Re-Trigger interrupt
  - Compare Match/Input Capture interrupt
  - Interrupt on fault detection

## 49.3 Block Diagram

**Figure 49-1. Timer/Counter for Control Applications - Block Diagram**




## 49.4 Signal Description

| Pin Name | Type | Description |
|---|---|---|
| TCC/WO[0] | Digital output | Compare channel 0 waveform output |
| TCC/WO[1] | Digital output | Compare channel 1 waveform output |

### 52.8.5 Interrupt Status Register

**Name:** ISR
**Offset:** 0x10
**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 |
|---|---|---|---|---|---|---|---|---|
| | | | | | RXBUFF | ENDRX | OVRE | DRDY |
| Access | | | | | R | R | R | R |
| Reset | | | | | 0 | 0 | 0 | 0 |

**Bit 3 – RXBUFF** Reception Buffer Full

| Value | Description |
|---|---|
| 0 | The signal Buffer Full from the reception PDC channel is inactive. |
| 1 | The signal Buffer Full from the reception PDC channel is active. |

**Bit 2 – ENDRX** End of Reception Transfer

| Value | Description |
|---|---|
| 0 | The End of Transfer signal from the reception PDC channel is inactive. |
| 1 | The End of Transfer signal from the reception PDC channel is active. |

**Bit 1 – OVRE** Overrun Error Interrupt Status
The OVRE flag is automatically reset when this register is read or when the PCC is disabled.

| Value | Description |
|---|---|
| 0 | No overrun error occurred since the last read of this register. |
| 1 | At least one overrun error occurred since the last read of this register. |

**Bit 0 – DRDY** Data Ready Interrupt Status
The DRDY flag is automatically reset when RHR is read or when the PCC is disabled.

### 53.8.6 Interrupt Enable Set

**Name:** INTENSET
**Offset:** 0x09
**Reset:** 0x00
**Property:** PAC Write-Protection

This register allows the user to change this register without doing a read-modify-write operation. Changes in this register will also be reflected in the Interrupt Enable Clear (INTENCLR) register.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | MC1 | MC0 | VLC | DIR | ERR | OVF |
| Access | | | RW | RW | RW | RW | RW | RW |
| Reset | | | 0 | 0 | 0 | 0 | 0 | 0 |

**Bits 4, 5 – MC** Channel x Compare Match Enable
Writing a '0' to MCx has no effect.

Writing a '1' to MCx will set the corresponding Match Channel x Interrupt Disable/Enable bit, which enables the Match Channel x interrupt.

| Value | Description |
|---|---|
| 0 | The Match Channel x interrupt is disabled. |
| 1 | The Match Channel x interrupt is enabled. |

**Bit 3 – VLC** Velocity Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Velocity Interrupt Disable/Enable bit, which enables the Velocity interrupt.

This bit has no effect when COUNTER operation mode is selected.

| Value | Description |
|---|---|
| 0 | The Velocity interrupt is disabled. |
| 1 | The Velocity interrupt is enabled. |

**Bit 2 – DIR** Direction Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Direction Change Interrupt Disable/Enable bit, which enables the Direction Change interrupt.

This bit has no effect when COUNTER operation mode is selected.

| Value | Description |
|---|---|
| 0 | The Direction Change interrupt is disabled. |
| 1 | The Direction Change interrupt is enabled. |

**Bit 1 – ERR** Error Interrupt Enable
Writing a '0' to this bit has no effect.

Writing a '1' to this bit will set the Error Interrupt Disable/Enable bit, which enables the Error interrupt.

**Figure 56-3. Power Supply Connection for Battery Backup**

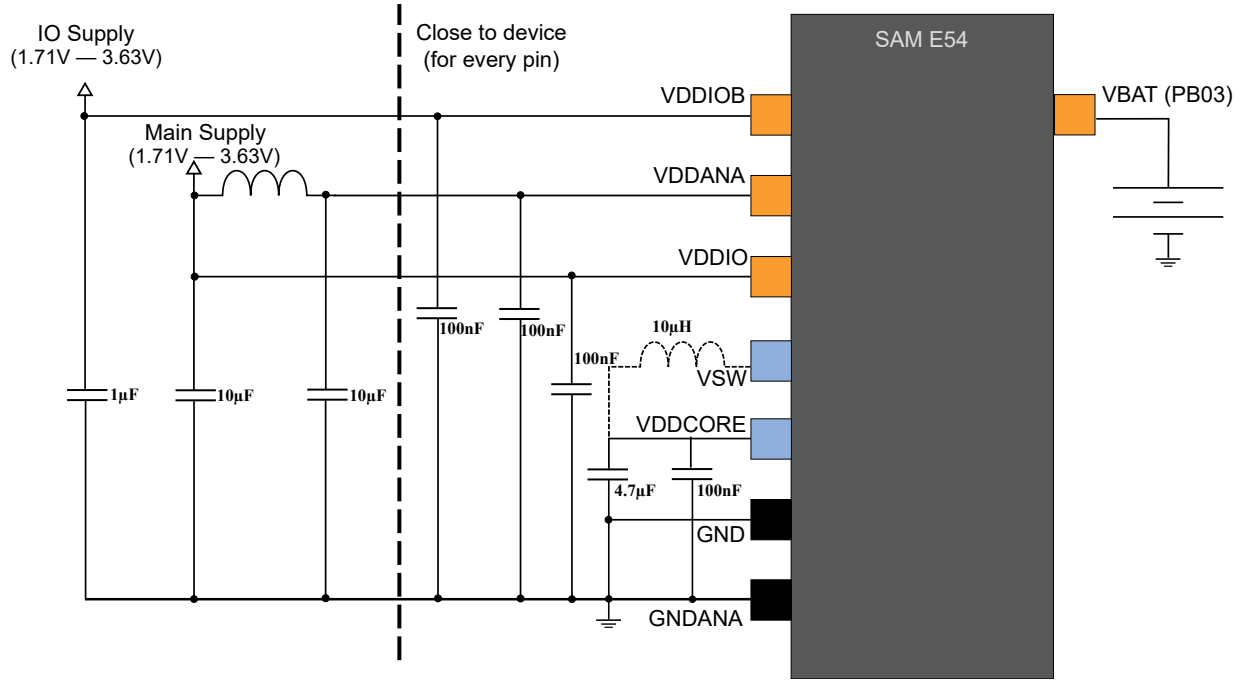



**Table 56-1. Power Supply Connections, $V_{DDCORE}$ or $V_{DDOUT}$ From Internal Regulator**

| Signal Name | Recommended Pin Connection | Description |
|---|---|---|
| $V_{DDIO}$ | 1.71V to 3.6V<br>Decoupling/filtering capacitors 100 nF(1,2) and 10 µF(1)<br>Decoupling/filtering inductor 10 µH(1,3) | Digital supply voltage |
| $V_{DDANA}$ | 1.71V to 3.6V<br>Decoupling/filtering capacitors 100 nF(1,2) and 10 µF(1)<br>Ferrite bead(4) prevents the $V_{DD}$ noise interfering with $V_{DDANA}$ | Analog supply voltage |
| $V_{DDIOB}$ | 1.71V to 3.6V<br>Decoupling/filtering capacitor 1 µF(1) | Digital supply voltage |
| $V_{BAT}$ | 1.71V to 3.6V when connected | External battery supply input |
| $V_{DDCORE}$ | 1V to 1.2V typical<br>Decoupling/filtering capacitors 100 nF(1,2) and 4.7 µF(1) | Linear regulator mode: Core supply voltage output/ external decoupling pin<br>Switched regulator mode: Core supply voltage input, must be connected to $V_{DDOUT}$ via inductor |
| $V_{SW}$ | Switching regulator mode: 10 µH inductor with saturation current above 500 mA and ESR = 0.7Ω | On-chip switching mode regulator output |