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	AVR
Core Size	32-Bit Single-Core
Speed	50MHz
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	36
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	1.62V ~ 3.6V
Data Converters	A/D 8x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	48-TQFP
Supplier Device Package	48-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/atuc256l4u-aur

Table 4-4. Priority and Handler Addresses for Events

Priority	Handler Address	Name	Event source	Stored Return Address
1	0x80000000	Reset	External input	Undefined
2	Provided by OCD system	OCD Stop CPU	OCD system	First non-completed instruction
3	EVBA+0x00	Unrecoverable exception	Internal	PC of offending instruction
4	EVBA+0x04	TLB multiple hit	MPU	PC of offending instruction
5	EVBA+0x08	Bus error data fetch	Data bus	First non-completed instruction
6	EVBA+0x0C	Bus error instruction fetch	Data bus	First non-completed instruction
7	EVBA+0x10	NMI	External input	First non-completed instruction
8	Autovectored	Interrupt 3 request	External input	First non-completed instruction
9	Autovectored	Interrupt 2 request	External input	First non-completed instruction
10	Autovectored	Interrupt 1 request	External input	First non-completed instruction
11	Autovectored	Interrupt 0 request	External input	First non-completed instruction
12	EVBA+0x14	Instruction Address	CPU	PC of offending instruction
13	EVBA+0x50	ITLB Miss	MPU	PC of offending instruction
14	EVBA+0x18	ITLB Protection	MPU	PC of offending instruction
15	EVBA+0x1C	Breakpoint	OCD system	First non-completed instruction
16	EVBA+0x20	Illegal Opcode	Instruction	PC of offending instruction
17	EVBA+0x24	Unimplemented instruction	Instruction	PC of offending instruction
18	EVBA+0x28	Privilege violation	Instruction	PC of offending instruction
19	EVBA+0x2C	Floating-point	UNUSED	
20	EVBA+0x30	Coprocessor absent	Instruction	PC of offending instruction
21	EVBA+0x100	Supervisor call	Instruction	PC(Supervisor Call) +2
22	EVBA+0x34	Data Address (Read)	CPU	PC of offending instruction
23	EVBA+0x38	Data Address (Write)	CPU	PC of offending instruction
24	EVBA+0x60	DTLB Miss (Read)	MPU	PC of offending instruction
25	EVBA+0x70	DTLB Miss (Write)	MPU	PC of offending instruction
26	EVBA+0x3C	DTLB Protection (Read)	MPU	PC of offending instruction
27	EVBA+0x40	DTLB Protection (Write)	MPU	PC of offending instruction
28	EVBA+0x44	DTLB Modified	UNUSED	

6. Supply and Startup Considerations

6.1 Supply Considerations

6.1.1 Power Supplies

The ATUC64/128/256L3/4U has several types of power supply pins:

- VDDIO: Powers I/O lines. Voltage is 1.8 to 3.3V nominal.
- VDDIN: Powers I/O lines, the USB pins, and the internal regulator. Voltage is 1.8 to 3.3V nominal if USB is not used, and 3.3V nominal when USB is used.
- VDDANA: Powers the ADC. Voltage is 1.8V nominal.
- VDDCORE: Powers the core, memories, and peripherals. Voltage is 1.8V nominal.

The ground pins GND are common to VDDCORE, VDDIO, and VDDIN. The ground pin for VDDANA is GNDANA.

When VDDCORE is not connected to VDDIN, the VDDIN voltage must be higher than 1.98V.

Refer to Section 35. on page 897 for power consumption on the various supply pins.

For decoupling recommendations for the different power supplies, please refer to the schematic checklist.

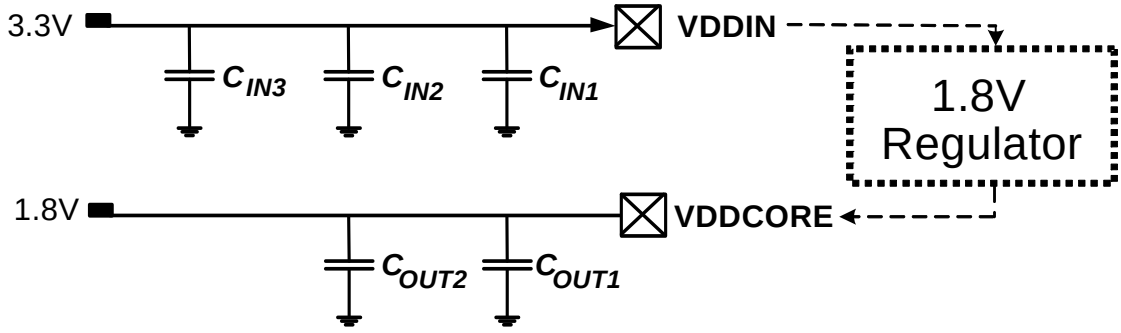
Refer to Section on page 10 for power supply connections for I/O pins.

6.1.2 Voltage Regulator

The ATUC64/128/256L3/4U embeds a voltage regulator that converts from 3.3V nominal to 1.8V with a load of up to 60 mA. The regulator supplies the output voltage on VDDCORE. The regulator may only be used to drive internal circuitry in the device. VDDCORE should be externally connected to the 1.8V domains. See Section 6.1.3 for regulator connection figures.

Adequate output supply decoupling is mandatory for VDDCORE to reduce ripple and avoid oscillations. The best way to achieve this is to use two capacitors in parallel between VDDCORE and GND as close to the device as possible. Please refer to Section 35.8 on page 911 for decoupling capacitors values and regulator characteristics.

Figure 6-1. Supply Decoupling.



The voltage regulator can be turned off in the shutdown mode to power down the core logic and keep a small part of the system powered in order to reduce power consumption. To enter this mode the 3.3V supply mode, with 1.8V regulated I/O lines power supply configuration must be used.

7.7.27 Performance Channel 1 Write Data Cycles

Name: PWDATA1

Access Type: Read-only

Offset: 0x828

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
DATA[31:24]							

23	22	21	20	19	18	17	16
DATA[23:16]							

15	14	13	12	11	10	9	8
DATA[15:8]							

7	6	5	4	3	2	1	0
DATA[7:0]							

- **DATA: Data Cycles Counted Since Last Reset**
  Clock cycles are counted using the CLK_PDCA_HSB clock

8.6.2.14 *Management of IN endpoints*

- *Overview*

IN packets are sent by the USBC device controller upon IN requests from the host.

The endpoint and its descriptor in RAM must be pre configured (see section "RAM management" on page 90 for more details).

When the current bank is clear, the TXINI and FIFO Control (UECONn.FIFOCON) bits will be set simultaneously. This triggers an EPnINT interrupt if the Transmitted IN Data Interrupt Enable (TXINE) bit in UECONn is one.

TXINI shall be cleared by software (by writing a one to the Transmitted IN Data Interrupt Enable Clear bit in the Endpoint n Control Clear register (UECONnCLR.TXINIC)) to acknowledge the interrupt. This has no effect on the endpoint FIFO.

The user writes the IN data to the bank referenced by the EPn descriptor and allows the USBC to send the data by writing a one to the FIFO Control Clear (UECONnCLR.FIFOCONC) bit. This will also cause a switch to the next bank if the IN endpoint is composed of multiple banks. The TXINI and FIFOCON bits will be updated accordingly.

TXINI should always be cleared before clearing FIFOCON to avoid missing an TXINI event.

Figure 8-8. Example of an IN endpoint with one data bank

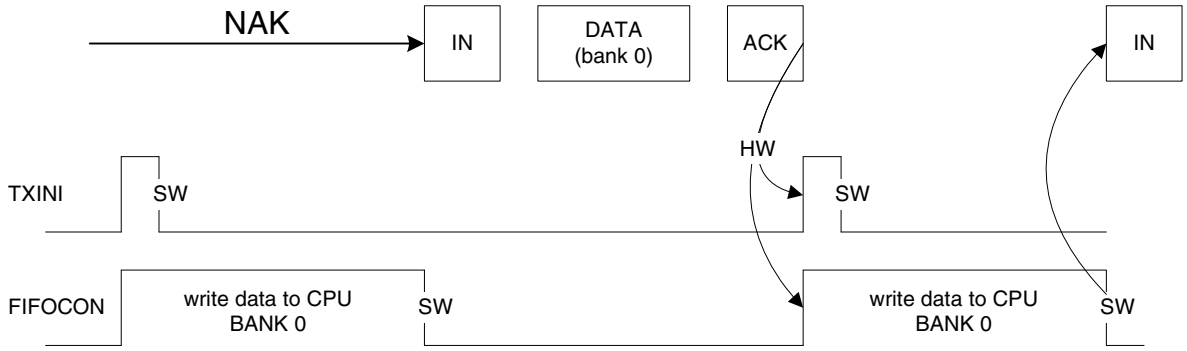
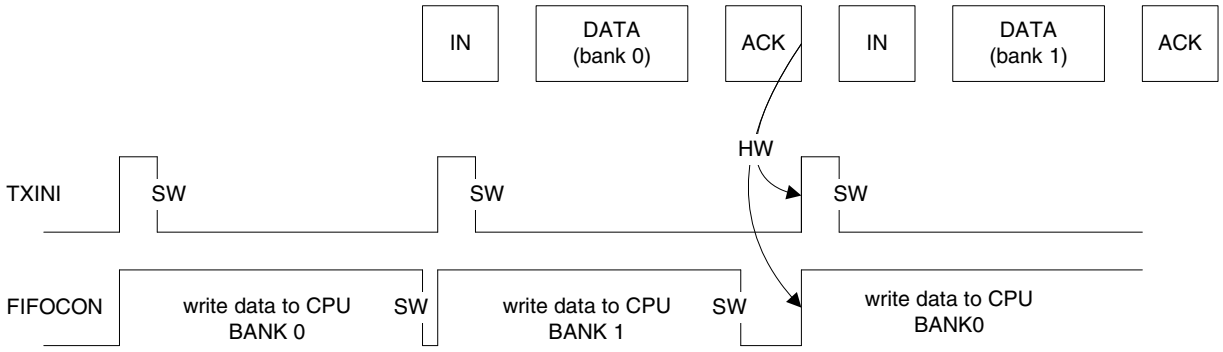


Figure 8-9. Example of an IN endpoint with two data banks



9.3.4 Debug Operation

When an external debugger forces the CPU into debug mode, the FLASHCDW continues normal operation. If the FLASHCDW is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

9.4 Functional Description

9.4.1 Bus Interfaces

The FLASHCDW has two bus interfaces, one High Speed Bus (HSB) interface for reads from the flash memory and writes to the page buffer, and one Peripheral Bus (PB) interface for issuing commands and reading status from the controller.

9.4.2 Memory Organization

The flash memory is divided into a set of pages. A page is the basic unit addressed when programming the flash. A page consists of several words. The pages are grouped into 16 regions of equal size. Each of these regions can be locked by a dedicated fuse bit, protecting it from accidental modification.

- *p* pages (*FLASH_P*)
- *w* bytes in each page and in the page buffer (*FLASH_W*)
- *pw* bytes in total (*FLASH_PW*)
- *f* general-purpose fuse bits (*FLASH_F*), used as region lock bits and for other device-specific purposes
- 1 security fuse bit
- 1 User page

9.4.3 User Page

The User page is an additional page, outside the regular flash array, that can be used to store various data, such as calibration data and serial numbers. This page is not erased by regular chip erase. The User page can only be written and erased by a special set of commands. Read accesses to the User page are performed just as any other read accesses to the flash. The address map of the User page is given in Figure 9-1 on page 138.

9.4.4 Read Operations

The on-chip flash memory is typically used for storing instructions to be executed by the CPU. The CPU will address instructions using the HSB bus, and the FLASHCDW will access the flash memory and return the addressed 32-bit word.

In systems where the HSB clock period is slower than the access time of the flash memory, the FLASHCDW can operate in 0 wait state mode, and output one 32-bit word on the bus per clock cycle. If the clock frequency allows, the user should use 0 wait state mode, because this gives the highest performance as no stall cycles are encountered.

The FLASHCDW can also operate in systems where the HSB bus clock period is faster than the access speed of the flash memory. Wait state support and a read granularity of 64 bits ensure efficiency in such systems.

Performance for systems with high clock frequency is increased since the internal read word width of the flash memory is 64 bits. When a 32-bit word is to be addressed, the word itself and

9.8 User Interface

Table 9-6. FLASHCDW Register Memory Map

Offset	Register	Register Name	Access	Reset
0x00	Flash Control Register	FCR	Read/Write	0x00000000
0x04	Flash Command Register	FCMD	Read/Write	0x00000000
0x08	Flash Status Register	FSR	Read-only	-(1)
0x0C	Flash Parameter Register	FPR	Read-only	-(3)
0x10	Flash Version Register	FVR	Read-only	-(3)
0x14	Flash General Purpose Fuse Register Hi	FGPFRHI	Read-only	-(2)
0x18	Flash General Purpose Fuse Register Lo	FGPFRLO	Read-only	-(2)

Note:
1. The value of the Lock bits depend on their programmed state. All other bits in FSR are 0.
2. All bits in FGPRHI/LO are dependent on the programmed state of the fuses they map to. Any bits in these registers not mapped to a fuse read as 0.
3. The reset values for these registers are device specific. Please refer to the Module Configuration section at the end of this chapter.

14.6.25 PLL Control Register

Name: PLLn

Access Type: Read/Write

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
-	-	PLLCOUNT					

23	22	21	20	19	18	17	16
-	-	-	-	PLLMUL			

15	14	13	12	11	10	9	8
-	-	-	-	PLLDIV			

7	6	5	4	3	2	1	0
-	-	PLLOPT			PLLOSC		PLLEN

- **PLLCOUNT: PLL Count**
  Specifies the number of RCSYS clock cycles before ISR.PLLLOCKn will be set after PLLn has been written, or after PLLn has been automatically re-enabled after exiting a sleep mode.
- **PLLMUL: PLL Multiply Factor**
- **PLLDIV: PLL Division Factor**
  These fields determine the ratio of the PLL output frequency to the source oscillator frequency:
  f_{vco} = (PLLMUL+1)/PLLDIV • f_{REF} if PLLDIV >0
  f_{vco} = 2•(PLLMUL+1) • f_{REF} if PLLDIV = 0
  Note that the PLLMUL field should always be greater than 1 or the behavior of the PLL will be undefined.
- **PLLOPT: PLL Option**
  PLLOPT[0]: Selects the VCO frequency range (f_{vco}).
  0: 80MHz<f_{vco}<180MHz
  1: 160MHz<f_{vco}<240MHz
  PLLOPT[1]: Divides the output frequency by 2.
  0: f_{PLL}= f_{vco}
  1: f_{PLL} = f_{vco}/2
  PLLOPT[2]:Wide-Bandwidth mode.
  0: Wide Bandwidth Mode enabled
  1: Wide Bandwidth Mode disabled
- **PLLOSC: PLL Oscillator Select**
  Reference clock source select for the reference clock, please refer to the “PLL Clock Sources” table in the SCIF Module Configuration section for details.

15.6.16 Event Disable Register

Name: EVD

Access Type: Write-only

Offset: 0x4C

Reset Value: 0x00000000

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

23	22	21	20	19	18	17	16
-	-	-	-	-	-	PER1	PER0

15	14	13	12	11	10	9	8
-	-	-	-	-	-	ALARM1	ALARM0

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	OVF

When the SR.BUSY bit is set writes to this register will be discarded and this register will read as zero.

Writing a zero to a bit in this register has no effect.

Writing a one to a bit in this register will clear the corresponding bit in EVM.

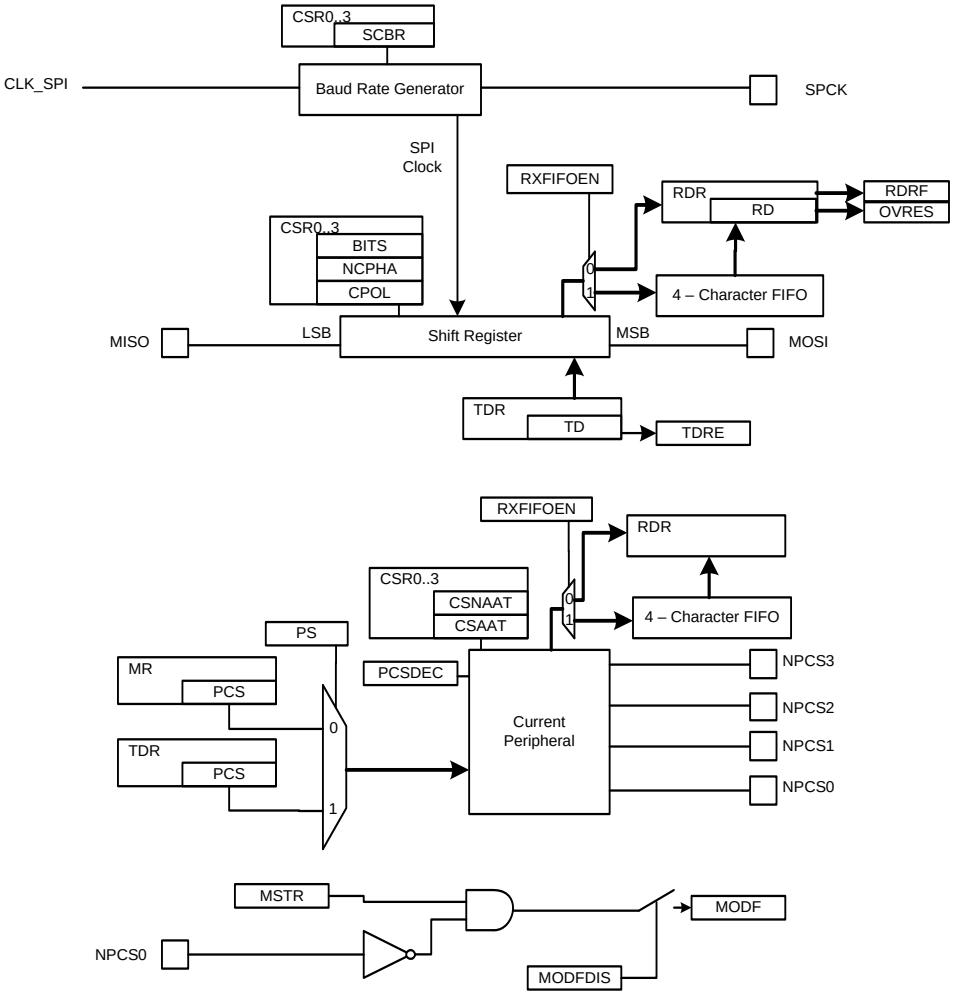
Table 20-18.

0	1	SUBSCRIBE: The USART receives the response.
1	0	IGNORE: The USART does not transmit and does not receive the response.
1	1	Reserved

Figure 21-5 on page 491shows a block diagram of the SPI when operating in master mode. Figure 21-6 on page 492 shows a flow chart describing how transfers are handled.

21.7.3.1 Master mode block diagram

Figure 21-5. Master Mode Block Diagram



- **SWRST: Software Reset**
  This bit will always read as 0.
  Writing a zero to this bit has no effect.
  Writing a one to this bit resets the TWIS.
- **STREN: Clock Stretch Enable**
  0: Disables clock stretching if RHR/THR buffer full/empty. May cause over/underrun.
  1: Enables clock stretching if RHR/THR buffer full/empty.
- **GCMATCH: General Call Address Match**
  0: Causes the TWIS not to acknowledge the General Call Address.
  1: Causes the TWIS to acknowledge the General Call Address.
- **SMATCH: Slave Address Match**
  0: Causes the TWIS not to acknowledge the Slave Address.
  1: Causes the TWIS to acknowledge the Slave Address.
- **SMEN: SMBus Mode Enable**
  0: Disables SMBus mode.
  1: Enables SMBus mode.
- **SEN: Slave Enable**
  0: Disables the slave interface.
  1: Enables the slave interface.

Data words are right-justified in the RHR and THR registers. For 16-bit compact stereo, the left sample uses bits 15 through 0 and the right sample uses bits 31 through 16 of the same data word. For 8-bit compact stereo, the left sample uses bits 7 through 0 and the right sample uses bits 15 through 8 of the same data word.

24.6.8 DMA Operation

The Receiver and the Transmitter can each be connected either to one single Peripheral DMA channel or to one Peripheral DMA channel per data channel. This is selected by writing to the MR.RXDMA and MR.TXDMA bits. If a single Peripheral DMA channel is selected, all data samples use IISC Receiver or Transmitter DMA channel 0.

The Peripheral DMA reads from the RHR register and writes to the RHR register for both audio channels, successively.

The Peripheral DMA transfers may use 32-bit word, 16-bit halfword, or 8-bit byte according to the value of the MR.DATALENGTH field.

24.6.9 Loop-back Mode

For debugging purposes, the IISC can be configured to loop back the Transmitter to the Receiver. Writing a one to the MR.LOOP bit will internally connect ISDO to ISDI, so that the transmitted data is also received. Writing a zero to MR.LOOP will restore the normal behavior with independent Receiver and Transmitter. As for other changes to the Receiver or Transmitter configuration, the IISC Receiver and Transmitter must be disabled before writing to the MR register to update MR.LOOP.

24.6.10 Interrupts

An IISC interrupt request can be triggered whenever one or several of the following bits are set in the Status Register (SR): Receive Ready (RXRDY), Receive Overrun (RXOR), Transmit Ready (TXRDY), or Transmit Underrun (TXOR).

The interrupt request will be generated if the corresponding bit in the Interrupt Mask Register (IMR) is set. Bits in IMR are set by writing a one to the corresponding bit in the Interrupt Enable Register (IER), and cleared by writing a one to the corresponding bit in the Interrupt Disable Register (IDR). The interrupt request remains active until the corresponding bit in SR is cleared by writing a one the corresponding bit in the Status Clear Register (SCR).

For debugging purposes, interrupt requests can be simulated by writing a one to the corresponding bit in the Status Set Register (SSR).

25.5.1 I/O Lines

The pins used for interfacing the PWMA may be multiplexed with I/O Controller lines. The programmer must first program the I/O Controller to assign the desired PWMA pins to their peripheral function.

It is only required to enable the PWMA outputs actually in use.

25.5.2 Power Management

If the CPU enters a sleep mode that disables clocks used by the PWMA, the PWMA will stop functioning and resume operation after the system wakes up from sleep mode.

25.5.3 Clocks

The clock for the PWMA bus interface (CLK_PWMA) is controlled by the Power Manager. This clock is enabled at reset, and can be disabled in the Power Manager. It is recommended to disable the PWMA before disabling the clock, to avoid freezing the PWMA in an undefined state.

Additionally, the PWMA depends on a dedicated Generic Clock (GCLK). The GCLK can be set to a wide range of frequencies and clock sources and must be enabled in the System Control Interface (SCIF) before the PWMA can be used.

25.5.4 Interrupts

The PWMA interrupt request lines are connected to the interrupt controller. Using the PWMA interrupts requires the interrupt controller to be programmed first.

25.5.5 Peripheral Events

The PWMA peripheral events are connected via the Peripheral Event System. Refer to the Peripheral Event System chapter for details.

25.5.6 Debug Operation

When an external debugger forces the CPU into debug mode, the PWMA continues normal operation. If the PWMA is configured in a way that requires it to be periodically serviced by the CPU through interrupts, improper operation or data loss may result during debugging.

25.6 Functional Description

The PWMA embeds a number of PWM channel submodules, each providing an output PWM waveform. Each PWM channel contains a duty cycle register and a comparator. A common timebase counter for all channels determines the frequency and the period for all the PWM waveforms.

25.6.1 Enabling the PWMA

Once the GCLK has been enabled, the PWMA is enabled by writing a one to the EN bit in the Control Register (CR).

25.6.2 Timebase Counter

The top value of the timebase counter defines the period of the PWMA output waveform. The timebase counter starts at zero when the PWMA is enabled and counts upwards until it reaches its effective top value (ETV). The effective top value is defined by specifying the desired number of GCLK clock cycles in the TOP field of Top Value Register (TVR.TOP) in normal operation (the

26. Timer/Counter (TC)

Rev: 2.2.3.1.3

26.1 Features

- **Three 16-bit Timer Counter channels**
- **A wide range of functions including:**
  - **Frequency measurement**
  - **Event counting**
  - **Interval measurement**
  - **Pulse generation**
  - **Delay timing**
  - **Pulse width modulation**
  - **Up/down capabilities**
- **Each channel is user-configurable and contains:**
  - **Three external clock inputs**
  - **Five internal clock inputs**
  - **Two multi-purpose input/output signals**
- **Internal interrupt signal**
- **Two global registers that act on all three TC channels**
- **Peripheral event input on all A lines in capture mode**

26.2 Overview

The Timer Counter (TC) includes three identical 16-bit Timer Counter channels.

Each channel can be independently programmed to perform a wide range of functions including frequency measurement, event counting, interval measurement, pulse generation, delay timing, and pulse width modulation.

Each channel has three external clock inputs, five internal clock inputs, and two multi-purpose input/output signals which can be configured by the user. Each channel drives an internal interrupt signal which can be programmed to generate processor interrupts.

The TC block has two global registers which act upon all three TC channels.

The Block Control Register (BCR) allows the three channels to be started simultaneously with the same instruction.

The Block Mode Register (BMR) defines the external clock inputs for each channel, allowing them to be chained.

26.7.6 Channel Register B

Name: RB

Access Type: Read-only if CMRn.WAVE = 0, Read/Write if CMRn.WAVE = 1

Offset: 0x18 + n * 0x40

Reset Value: 0x00000000

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

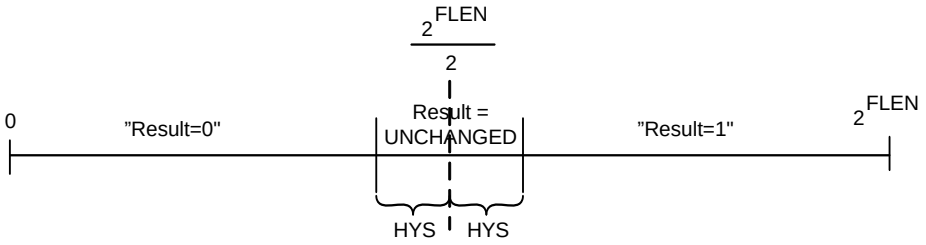
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-

15	14	13	12	11	10	9	8
RB[15:8]							

7	6	5	4	3	2	1	0
RB[7:0]							

- **RB: Register B**
  RB contains the Register B value in real time.

Figure 30-3. The Filtering Algorithm



30.7 Peripheral Event Triggers

Peripheral events from other modules can trigger comparisons in the ACIFB. All channels that are set up in Event Triggered Single Measurement Mode will be started simultaneously when a peripheral event is received. Channels that are operating in Continuous Measurement Mode or User Triggered Single Measurement Mode will be unaffected by the received event. The software can still operate these channels independently of channels in Event Triggered Single Measurement Mode.

A peripheral event will trigger one or more comparisons, in normal or window mode.

30.8 AC Test mode

By writing the Analog Comparator Test Mode (CR.ACTEST) bit to one, the outputs from the ACs are overridden by the value in the Test Register (TR), see Figure 30-1. This is useful for software development.

(MBLEN) when using the QMatrix acquisition method. Two additional handshakes support DMA-Touch by regulating transfers from memory to the DMATouch State Write Register (DMATSW) and from the DMATouch State Read Register (DMATSR) to memory. The Peripheral DMA Controller must be configured properly and enabled in order to perform direct memory access transfers to/from the CAT module.

31.5.6 Analog Comparators

When the CAT module is performing QMatrix acquisition, it requires that on-chip analog comparators be used as part of the process. These analog comparators are not controlled directly by the CAT module, but by a separate Analog Comparator (AC) Interface. This interface must be configured properly and enabled before the CAT module is used. This includes configuring the generic clock input for the analog comparators to the proper sampling frequency.

The CAT will automatically use the negative peripheral events from the AC Interface on every Y pin in QMatrix mode. When QMatrix acquisition is used the analog comparator corresponding to the selected Y pins must be enabled and converting continuously, using the Y pin as the positive reference and the ACREFN as negative reference.

31.5.7 Debug Operation

When an external debugger forces the CPU into debug mode, the CAT continues normal operation. If the CAT is configured in a way that requires it to be periodically serviced by the CPU through interrupts or similar, improper operation or data loss may result during debugging.

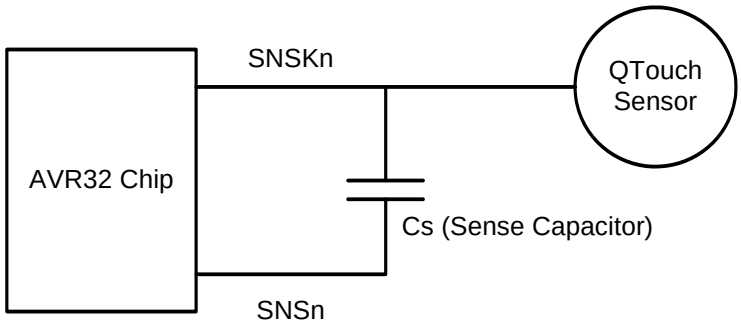
31.6 Functional Description

31.6.1 Acquisition Types

The CAT module can perform several types of QTouch acquisition from capacitive touch sensors: autonomous QTouch (one sensor only), DMATouch, QTouch group A, and QTouch group B. The CAT module can also perform QMatrix acquisition. Each type of acquisition has an associated set of pin selection and configuration registers that allow a large degree of flexibility.

The following schematic diagrams show typical hardware connections for QTouch and QMatrix sensors, respectively:

Figure 31-2. CAT Touch Connections



34.6 aWire Debug Interface (AW)

Rev.: 2.3.0.1

34.6.1 Features

- **Single pin debug system.**
- **Half Duplex asynchronous communication (UART compatible).**
- **Full duplex mode for direct UART connection.**
- **Compatible with JTAG functionality, except boundary scan.**
- **Failsafe packet-oriented protocol.**
- **Read and write on-chip memory and program on-chip flash and fuses through SAB interface.**
- **On-Chip Debug access through SAB interface.**
- **Asynchronous receiver or transmitter when the aWire system is not used for debugging.**

34.6.2 Overview

The aWire Debug Interface (AW) offers a single pin debug solution that is fully compatible with the functionality offered by the JTAG interface, except boundary scan. This functionality includes memory access, programming capabilities, and On-Chip Debug access.

Figure 34-8 on page 881 shows how the AW is connected in a 32-bit AVR device. The RESET_N pin is used both as reset and debug pin. A special sequence on RESET_N is needed to block the normal reset functionality and enable the AW.

The Service Access Bus (SAB) interface contains address and data registers for the Service Access Bus, which gives access to On-Chip Debug, programming, and other functions in the device. The SAB offers several modes of access to the address and data registers, as discussed in Section 34.6.6.8.

Section 34.6.7 lists the supported aWire commands and responses, with references to the description in this document.

If the AW is not used for debugging, the aWire UART can be used by the user to send or receive data with one stop bit, eight data bits, no parity bits, and one stop bit. This can be controlled through the aWire user interface.

1. The size of the data field: 7 (size and starting address + read length indicator) in the length field.
2. The size of the transfer: Words, halfwords, or bytes.
3. The starting address of the transfer.
4. The number of **bytes** to read (max 65532).

The 4 MSB of the 36 bit SAB address are submitted together with the size field (2 bits). The 4 remaining address bytes are submitted before the number of bytes to read. The size of the transfer is specified using the values from the following table:

Table 34-43. Size Field Decoding

Size field	Description
00	Byte transfer
01	Halfword transfer
10	Word transfer
11	Reserved

Below is an example read command:

1. 0x55 (sync)
2. 0x81 (command)
3. 0x00 (length MSB)
4. 0x07 (length LSB)
5. 0x25 (size and address MSB, the two MSB of this byte are unused and set to zero)
6. 0x00
7. 0x00
8. 0x00
9. 0x04 (address LSB)
10. 0x00
11. 0x04
12. 0xXX (CRC MSB)
13. 0xXX (CRC LSB)

The length field is set to 0x0007 because there are 7 bytes of additional data: 5 bytes of address and size and 2 bytes with the number of bytes to read. The address and size field indicates one word (four bytes) should be read from address 0x500000004.

Table 34-44. MEMORY_READ Details

Command	Details
Command value	0x81
Additional data	Size, Address and Length
Possible responses	0xC1: MEMDATA (Section 34.6.8.4) 0xC2: MEMORY_READWRITE_STATUS (Section 34.6.8.5) 0x41: NACK (Section 34.6.8.2)

Table 34-58. STATUS_INFO Details

Response	Details
Response value	0xC4
Additional data	2 status bytes

34.6.8.8 MEMORY_SPEED

Counts the number of RC120M clock cycles it takes to sync one message to the SAB interface and back again. The SAB clock speed (f_{sab}) can be calculated using the following formula:

$$f_{sab} = \frac{3f_{aw}}{CV - 3}$$

Table 34-59. MEMORY_SPEED Details

Response	Details
Response value	0xC5
Additional data	Clock cycle count (MS)

34.6.9 Security Restrictions

When the security fuse in the Flash is programmed, the following aWire commands are limited:

- MEMORY_WRITE
- MEMORY_READ

Unlimited access to these instructions is restored when the security fuse is erased by the CHIP_ERASE aWire command.

Note that the security bit will read as programmed and block these instructions also if the Flash Controller is statically reset.