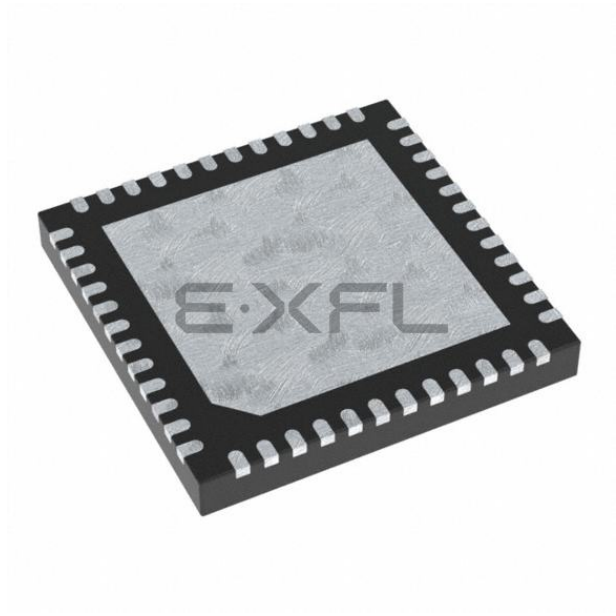


Welcome to E-XFL.COM

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
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	36
Program Memory Size	128KB (128K 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-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at32uc3l0128-zaut

The Frequency Meter (FREQM) allows accurate measuring of a clock frequency by comparing it to a known reference clock.

The device includes six identical 16-bit Timer/Counter (TC) channels. Each channel can be independently programmed to perform frequency measurement, event counting, interval measurement, pulse generation, delay timing, and pulse width modulation.

The Pulse Width Modulation controller (PWMA) provides 8-bit PWM channels which can be synchronized and controlled from a common timer. One PWM channel is available for each I/O pin on the device, enabling applications that require multiple PWM outputs, such as LCD backlight control. The PWM channels can operate independently, with duty cycles set individually, or in interlinked mode, with multiple channels changed at the same time.

The AT32UC3L0128/256 also features many communication interfaces, like USART, SPI, and TWI, for communication intensive applications. The USART supports different communication modes, like SPI Mode and LIN Mode.

A general purpose 8-channel ADC is provided, as well as eight analog comparators (AC). The ADC can operate in 10-bit mode at full speed or in enhanced mode at reduced speed, offering up to 12-bit resolution. The ADC also provides an internal temperature sensor input channel. The analog comparators can be paired to detect when the sensing voltage is within or outside the defined reference window.

The Capacitive Touch (CAT) module senses touch on external capacitive touch sensors, using the QTouch technology. Capacitive touch sensors use no external mechanical components, unlike normal push buttons, and therefore demand less maintenance in the user application. The CAT module allows up to 17 touch sensors, or up to 16 by 8 matrix sensors to be interfaced. All touch sensors can be configured to operate autonomously without software interaction, allowing wakeup from sleep modes when activated.

Atmel offers the QTouch library for embedding capacitive touch buttons, sliders, and wheels functionality into AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys as well as Adjacent Key Suppression® (AKS®) technology for unambiguous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop, and debug your own touch applications.

The AT32UC3L0128/256 integrates a class 2+ Nexus 2.0 On-chip Debug (OCD) System, with non-intrusive real-time trace and full-speed read/write memory access, in addition to basic run-time control. The NanoTrace interface enables trace feature for aWire- or JTAG-based debuggers. The single-pin aWire interface allows all features available through the JTAG interface to be accessed through the RESET pin, allowing the JTAG pins to be used for GPIO or peripherals.

Debug state can be entered as described in the *AVR32UC Technical Reference Manual*.

Debug state is exited by the *retd* instruction.

4.4.3.3 *Secure State*

The AVR32 can be set in a secure state, that allows a part of the code to execute in a state with higher security levels. The rest of the code can not access resources reserved for this secure code. Secure State is used to implement FlashVault technology. Refer to the *AVR32UC Technical Reference Manual* for details.

4.4.4 System Registers

The system registers are placed outside of the virtual memory space, and are only accessible using the privileged *mfsr* and *mtsr* instructions. The table below lists the system registers specified in the AVR32 architecture, some of which are unused in AVR32UC. The programmer is responsible for maintaining correct sequencing of any instructions following a *mtsr* instruction. For detail on the system registers, refer to the *AVR32UC Technical Reference Manual*.

Table 4-3. System Registers

Reg #	Address	Name	Function
0	0	SR	Status Register
1	4	EVBA	Exception Vector Base Address
2	8	ACBA	Application Call Base Address
3	12	CPUCR	CPU Control Register
4	16	ECR	Exception Cause Register
5	20	RSR_SUP	Unused in AVR32UC
6	24	RSR_INT0	Unused in AVR32UC
7	28	RSR_INT1	Unused in AVR32UC
8	32	RSR_INT2	Unused in AVR32UC
9	36	RSR_INT3	Unused in AVR32UC
10	40	RSR_EX	Unused in AVR32UC
11	44	RSR_NMI	Unused in AVR32UC
12	48	RSR_DBG	Return Status Register for Debug mode
13	52	RAR_SUP	Unused in AVR32UC
14	56	RAR_INT0	Unused in AVR32UC
15	60	RAR_INT1	Unused in AVR32UC
16	64	RAR_INT2	Unused in AVR32UC
17	68	RAR_INT3	Unused in AVR32UC
18	72	RAR_EX	Unused in AVR32UC
19	76	RAR_NMI	Unused in AVR32UC
20	80	RAR_DBG	Return Address Register for Debug mode
21	84	JECR	Unused in AVR32UC
22	88	JOSP	Unused in AVR32UC
23	92	JAVA_LV0	Unused in AVR32UC

7.5.10 Priority

If more than one PDCA channel is requesting transfer at a given time, the PDCA channels are prioritized by their channel number. Channels with lower numbers have priority over channels with higher numbers, giving channel zero the highest priority.

7.5.11 Error Handling

If the Memory Address Register (MAR) is set to point to an invalid location in memory, an error will occur when the PDCA tries to perform a transfer. When an error occurs, the Transfer Error bit in the Interrupt Status Register (ISR.TERR) will be set and the DMA channel that caused the error will be stopped. In order to restart the channel, the user must program the Memory Address Register to a valid address and then write a one to the Error Clear bit in the Control Register (CR.ECLR). If the Transfer Error interrupt is enabled, an interrupt request will be generated when a transfer error occurs.

7.5.12 Peripheral Event Trigger

Peripheral events can be used to trigger PDCA channel transfers. Peripheral Event synchronizations are enabled by writing a one to the Event Trigger bit in the Mode Register (MR.ETRIG). When set, all DMA requests will be blocked until a peripheral event is received. For each peripheral event received, only one data item is transferred. If no DMA requests are pending when a peripheral event is received, the PDCA will start a transfer as soon as a peripheral event is detected. If multiple events are received while the PDCA channel is busy transferring data, an overflow condition will be signaled in the Peripheral Event System. Refer to the Peripheral Event System chapter for more information.

7.6 Performance Monitors

Up to two performance monitors allow the user to measure the activity and stall cycles for PDCA transfers. To monitor a PDCA channel, the corresponding channel number must be written to one of the MON0/1CH fields in the Performance Control Register (PCONTROL) and a one must be written to the corresponding CH0/1EN bit in the same register.

Due to performance monitor hardware resource sharing, the two monitor channels should NOT be programmed to monitor the same PDCA channel. This may result in UNDEFINED performance monitor behavior.

7.6.1 Measuring mechanisms

Three different parameters can be measured by each channel:

- The number of data transfer cycles since last channel reset, both for read and write
- The number of stall cycles since last channel reset, both for read and write
- The maximum latency since last channel reset, both for read and write

These measurements can be extracted by software and used to generate indicators for bus latency, bus load, and maximum bus latency.

Each of the counters has a fixed width, and may therefore overflow. When an overflow is encountered in either the Performance Channel Data Read/Write Cycle registers (PRDATA0/1 and PWDATA0/1) or the Performance Channel Read/Write Stall Cycles registers (PRSTALL0/1 and PWSTALL0/1) of a channel, all registers in the channel are reset. This behavior is altered if the Channel Overflow Freeze bit is one in the Performance Control register (PCONTROL.CH0/1OVF). If this bit is one, the channel registers are frozen when either DATA or STALL reaches its maximum value. This simplifies one-shot readout of the counter values.

- Undefined Length Burst Arbitration

In order to avoid long slave handling during undefined length bursts (INCR), the Bus Matrix provides specific logic in order to re-arbitrate before the end of the INCR transfer. A predicted end of burst is used as a defined length burst transfer and can be selected among the following five possibilities:

1. Infinite: No predicted end of burst is generated and therefore INCR burst transfer will never be broken.
2. One beat bursts: Predicted end of burst is generated at each single transfer inside the INCP transfer.
3. Four beat bursts: Predicted end of burst is generated at the end of each four beat boundary inside INCR transfer.
4. Eight beat bursts: Predicted end of burst is generated at the end of each eight beat boundary inside INCR transfer.
5. Sixteen beat bursts: Predicted end of burst is generated at the end of each sixteen beat boundary inside INCR transfer.

This selection can be done through the ULBT field in the Master Configuration Registers (MCFG).

- Slot Cycle Limit Arbitration

The Bus Matrix contains specific logic to break long accesses, such as very long bursts on a very slow slave (e.g., an external low speed memory). At the beginning of the burst access, a counter is loaded with the value previously written in the SLOT_CYCLE field of the related Slave Configuration Register (SCFG) and decreased at each clock cycle. When the counter reaches zero, the arbiter has the ability to re-arbitrate at the end of the current byte, halfword, or word transfer.

10.4.2.2 Round-Robin Arbitration

This algorithm allows the Bus Matrix arbiters to dispatch the requests from different masters to the same slave in a round-robin manner. If two or more master requests arise at the same time, the master with the lowest number is first serviced, then the others are serviced in a round-robin manner.

There are three round-robin algorithms implemented:

1. Round-Robin arbitration without default master
2. Round-Robin arbitration with last default master
3. Round-Robin arbitration with fixed default master

- Round-Robin Arbitration without Default Master

This is the main algorithm used by Bus Matrix arbiters. It allows the Bus Matrix to dispatch requests from different masters to the same slave in a pure round-robin manner. At the end of the current access, if no other request is pending, the slave is disconnected from all masters. This configuration incurs one latency cycle for the first access of a burst. Arbitration without default master can be used for masters that perform significant bursts.

- Round-Robin Arbitration with Last Default Master

This is a biased round-robin algorithm used by Bus Matrix arbiters. It allows the Bus Matrix to remove the one latency cycle for the last master that accessed the slave. At the end of the cur-

11.6.2 Interrupt Request Registers

Name: IRR0...IRR63

Access Type: Read-only

Offset: 0x0FF - 0x1FC

Reset Value: N/A

31	30	29	28	27	26	25	24
IRR[32*x+31]	IRR[32*x+30]	IRR[32*x+29]	IRR[32*x+28]	IRR[32*x+27]	IRR[32*x+26]	IRR[32*x+25]	IRR[32*x+24]
23	22	21	20	19	18	17	16
IRR[32*x+23]	IRR[32*x+22]	IRR[32*x+21]	IRR[32*x+20]	IRR[32*x+19]	IRR[32*x+18]	IRR[32*x+17]	IRR[32*x+16]
15	14	13	12	11	10	9	8
IRR[32*x+15]	IRR[32*x+14]	IRR[32*x+13]	IRR[32*x+12]	IRR[32*x+11]	IRR[32*x+10]	IRR[32*x+9]	IRR[32*x+8]
7	6	5	4	3	2	1	0
IRR[32*x+7]	IRR[32*x+6]	IRR[32*x+5]	IRR[32*x+4]	IRR[32*x+3]	IRR[32*x+2]	IRR[32*x+1]	IRR[32*x+0]

- **IRR: Interrupt Request line**

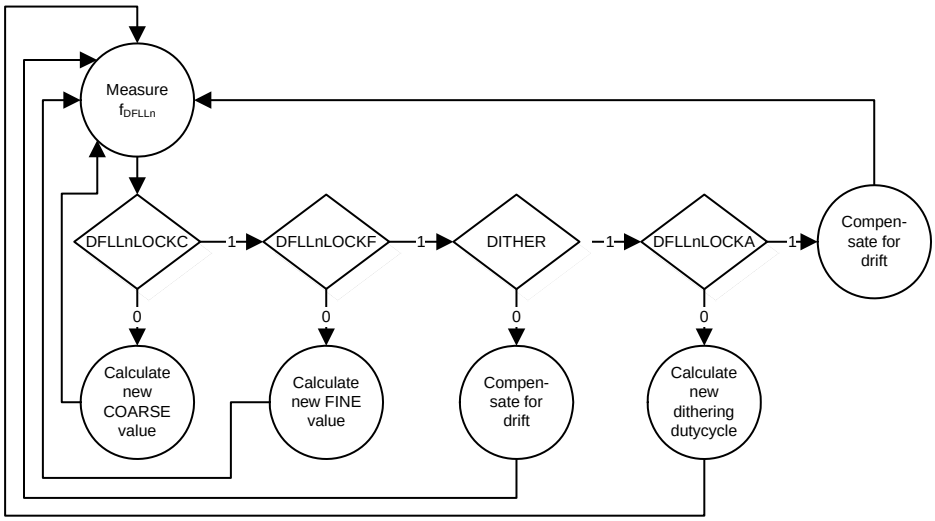
  This bit is cleared when no interrupt request is pending on this input request line.

  This bit is set when an interrupt request is pending on this input request line.

  The are 64 IRRs, one for each group. Each IRR has 32 bits, one for each possible interrupt request, for a total of 2048 possible input lines. The IRRs are read by the software interrupt handler in order to determine which interrupt request is pending. The IRRs are sampled continuously, and are read-only.

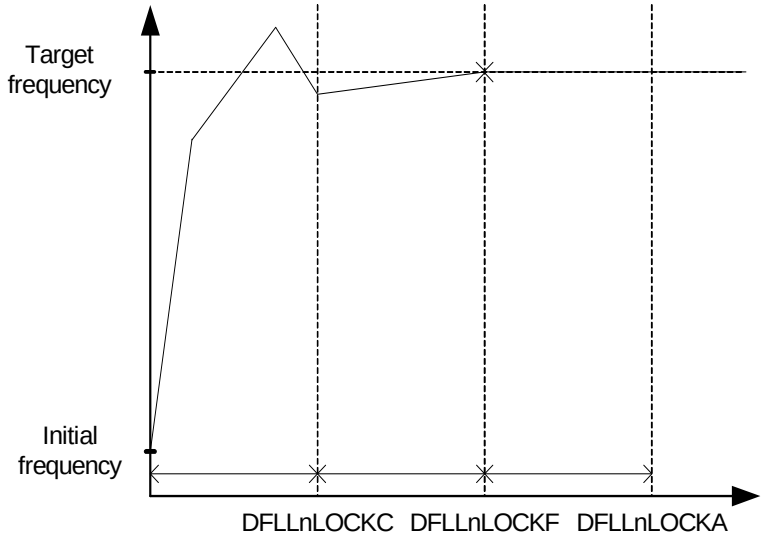
only be executed if the Dithering Enable bit (DITHER) in DFLLnCONF has been written to a one. If DITHER is written to a zero DFLLnLOCKA will never occur. If dithering is enabled, the frequency of the dithering is decided by a generic clock (CLK_DFLLIF_DITHER). This clock has to be set up correctly before enabling dithering. Please refer to the Generic Clocks section for details.

Figure 13-3. DFLL Closed loop State Diagram



When dithering is enabled the accuracy of the average output frequency of the DFLL will be higher. However, the actual frequency will be alternating between two frequencies. If a fixed frequency is required, the dithering should not be enabled.

Figure 13-4. DFLL Locking in Closed loop



CLK_DFLL is ready to be used when the DFLLn Synchronization Ready bit (DFLLnRDY) in PCLKSR is set after enabling the DFLL. However, the accuracy of the output frequency depends on which locks are set.

For lock times, please refer to the Electrical Characteristics chapter.

13.6 User Interface

Table 13-2. SCIF Register Memory Map

Offset	Register	Register Name	Access	Reset
0x0000	Interrupt Enable Register	IER	Write-only	0x00000000
0x0004	Interrupt Disable Register	IDR	Write-only	0x00000000
0x0008	Interrupt Mask Register	IMR	Read-only	0x00000000
0x000C	Interrupt Status Register	ISR	Read-only	0x00000000
0x0010	Interrupt Clear Register	ICR	Write-only	0x00000000
0x0014	Power and Clocks Status Register	PCLKSR	Read-only	0x00000000
0x0018	Unlock Register	UNLOCK	Write-only	0x00000000
0x001C	Oscillator 0 Control Register	OSCCTRL0	Read/Write	0x00000000
0x0020	Oscillator 32 Control Register	OSCCTRL32	Read/Write	0x00000004
0x0024	DFLL Config Register	DFLL0CONF	Read/Write	0x00000000
0x0028	DFLL Multiplier Register	DFLL0MUL	Write-only	0x00000000
0x002C	DFLL Step Register	DFLL0STEP	Write-only	0x00000000
0x0030	DFLL Spread Spectrum Generator Control Register	DFLL0SSG	Write-only	0x00000000
0x0034	DFLL Ratio Register	DFLL0RATIO	Read-only	0x00000000
0x0038	DFLL Synchronization Register	DFLL0SYNC	Write-only	0x00000000
0x003C	BOD Level Register	BOD	Read/Write	-(2)
0x0044	Voltage Regulator Calibration Register	VREGCR	Read/Write	-(2)
0x0048	System RC Oscillator Calibration Register	RCCR	Read/Write	-(2)
0x004C	Supply Monitor 33 Calibration Register	SM33	Read/Write	-(2)
0x0050	Temperature Sensor Calibration Register	TSENS	Read/Write	0x00000000
0x0058	120MHz RC Oscillator Control Register	RC120MCR	Read/Write	0x00000000
0x005C-0x0068	Backup Registers	BR	Read/Write	0x00000000
0x006C	32kHz RC Oscillator Control Register	RC32KCR	Read/Write	0x00000000
0x0070	Generic Clock Control0	GCCTRL0	Read/Write	0x00000000
0x0074	Generic Clock Control1	GCCTRL1	Read/Write	0x00000000
0x0078	Generic Clock Control2	GCCTRL2	Read/Write	0x00000000
0x007C	Generic Clock Control3	GCCTRL3	Read/Write	0x00000000
0x0080	Generic Clock Control4	GCCTRL4	Read/Write	0x00000000
0x0084	Generic Clock Control5	GCCTRL5	Read/Write	0x00000000
0x0088	Generic Clock Control6	GCCTRL6	Read/Write	0x00000000
0x008C	Generic Clock Control7	GCCTRL7	Read/Write	0x00000000
0x0090	Generic Clock Control8	GCCTRL8	Read/Write	0x00000000
0x0094	Generic Clock Control9	GCCTRL9	Read/Write	0x00000000

13.6.10 DFLLn Configuration Register

Name: DFLLnCONF

Access Type: Read/Write

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
COARSE[7:0]							

23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	FINE[8]

15	14	13	12	11	10	9	8
FINE[7:0]							

7	6	5	4	3	2	1	0
-	QLEN	CCEN	-	LLAW	DITHER	MODE	EN

- **COARSE: Coarse Calibration Value**
  Set the value of the coarse calibration register. If in closed loop mode, this field is Read-only.
- **FINE: FINE Calibration Value**
  Set the value of the fine calibration register. If in closed loop mode, this field is Read-only.
- **QLEN: Quick Lock Enable**
  0: Quick Lock is disabled.
  1: Quick Lock is enabled.
- **CCEN: Chill Cycle Enable**
  0: Chill Cycle is disabled.
  1: Chill Cycle is enabled.
- **LLAW: Lose Lock After Wake**
  0: Locks will not be lost after waking up from sleep modes.
  1: Locks will be lost after waking up from sleep modes where the DFLL clock has been stopped.
- **DITHER: Enable Dithering**
  0: The fine LSB input to the VCO is constant.
  1: The fine LSB input to the VCO is dithered to achieve sub-LSB approximation to the correct multiplication ratio.
- **MODE: Mode Selection**
  0: The DFLL is in open loop operation.
  1: The DFLL is in closed loop operation.
- **EN: Enable**
  0: The DFLL is disabled.
  1: The DFLL is enabled.

Note that this register is protected by a lock. To write to this register the UNLOCK register has to be written first. Please refer to the UNLOCK register description for details.

13.6.26 High Resolution Prescaler Control Register

Name: HRPCR

Access Type: Read/Write

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
HRCOUNT[23:16]							

23	22	21	20	19	18	17	16
HRCOUNT[15:8]							

15	14	13	12	11	10	9	8
HRCOUNT[7:0]							

7	6	5	4	3	2	1	0
-	-	-	-	CKSEL			HRPEN

- **HRCOUNT: High Resolution Counter**
  Specify the input clock period to count to generate the output clock edge.
  HRCOUNT can be written to dynamically in order to tune the HRPCLK frequency on-the-go.
- **CKSEL: Clock input selection**
  This field selects the Clock input for the prescaler. See the “HRP clock sources” table in the SCIF Module Configuration section for details. It must not be changed if the HRPEN is one.
- **HRPEN: High Resolution Prescaler Enable**
  0: The High Resolution Prescaler is disabled.
  1: The High Resolution Prescaler is enabled.

13.6.44 Generic Clock Version Register

Name: GCLKVERSION

Access Type: Read-only

Offset: 0x03F8

Reset Value: -

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

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

15	14	13	12	11	10	9	8
-	-	-	-	VERSION[11:8]			

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

- **VARIANT: Variant number**
  Reserved. No functionality associated.
- **VERSION: Version number**
  Version number of the module. No functionality associated.

15.5.3 Disabling the WDT

The WDT is disabled by writing a zero to the CTRL.EN bit. When disabling the WDT no other bits in the CTRL Register should be changed until the CTRL.EN bit reads back as zero. If the CTRL.CEN bit is written to zero, the CTRL.EN bit will never read back as zero if changing the value from one to zero.

15.5.4 Flash Calibration

The WDT can be enabled at reset. This is controlled by the WDTAUTO fuse. The WDT will be set in basic mode, RCSYS is set as source for CLK_CNT, and PSEL will be set to a value giving T_{psel} above 100 ms. Please refer to the Fuse Settings chapter for details about WDTAUTO and how to program the fuses.

If the Flash Calibration Done (FCD) bit in the CTRL Register is zero at a watchdog reset the flash calibration will be redone, and the CTRL.FCD bit will be set when the calibration is done. If CTRL.FCD is one at a watchdog reset, the configuration of the WDT will not be changed during flash calibration. After any other reset the flash calibration will always be done, and the CTRL.FCD bit will be set when the calibration is done.

15.5.5 Special Considerations

Care must be taken when selecting the PSEL/TBAN values so that the timeout period is greater than the startup time of the device. Otherwise a watchdog reset will reset the device before any code has been run. This can also be avoided by writing the CTRL.DAR bit to one when configuring the WDT.

If the Store Final Value (SFV) bit in the CTRL Register is one, the CTRL Register is locked for further write accesses. All writes to the CTRL Register will be ignored. Once the CTRL Register is locked, it can only be unlocked by a reset (e.g. POR, OCD, and WDT).

The CTRL.MODE bit can only be changed when the WDT is disabled (CTRL.EN=0).

17.6.4 Value Register

Name: VALUE

Access Type: Read-only

Offset: 0x00C

Reset Value: 0x00000000

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

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

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

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

- **VALUE:**
  Result from measurement.

18.7.16 Interrupt Flag Register

Name: IFR

Access: Read, Clear

Offset: 0x0D0, 0x0D8

Reset Value: -

31	30	29	28	27	26	25	24
P31	P30	P29	P28	P27	P26	P25	P24
23	22	21	20	19	18	17	16
P23	P22	P21	P20	P19	P18	P17	P16
15	14	13	12	11	10	9	8
P15	P14	P13	P12	P11	P10	P9	P8
7	6	5	4	3	2	1	0
P7	P6	P5	P4	P3	P2	P1	P0

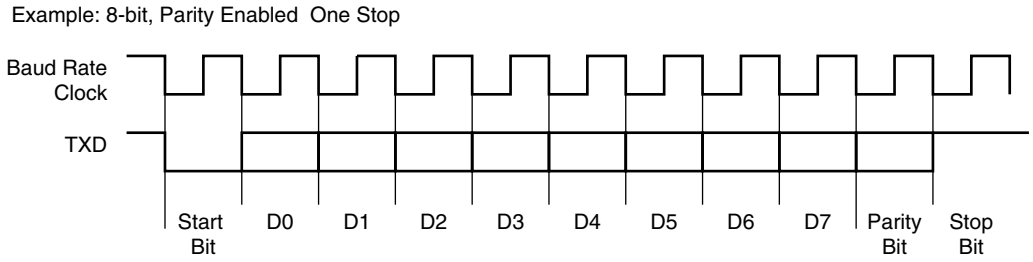
- **P0-31: Interrupt Flag**
  0: No interrupt condition has been detected on the corresponding pin.
  1: An interrupt condition has been detected on the corresponding pin.
  The number of interrupt request lines depends on the number of GPIO pins on the MCU. Refer to the product specific data for details. Note also that a bit in the Interrupt Flag register is only valid if the corresponding bit in IER is one.

19.6.3 Synchronous and Asynchronous Modes

19.6.3.1 Transmitter Operations

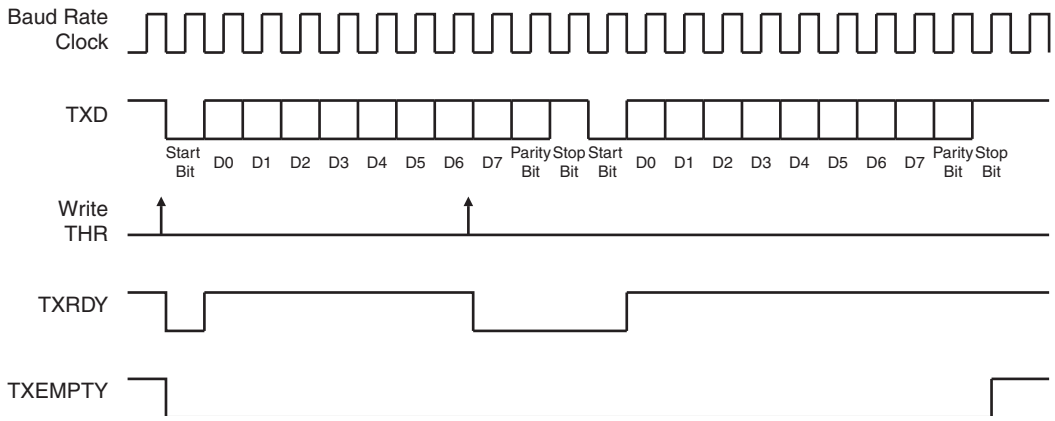
The transmitter performs equally in both synchronous and asynchronous operating modes (MR.SYNC). One start bit, up to 9 data bits, an optional parity bit, and up to two stop bits are successively shifted out on the TXD pin at each falling edge of the serial clock. The number of data bits is selected by the Character Length field (MR.CHRL) and the MR.MODE9 bit. Nine bits are selected by writing a one to MODE9, overriding any value in CHRL. The parity bit configuration is selected in the MR.PAR field. The Most Significant Bit First bit (MR.MSBF) selects which data bit to send first. The number of stop bits is selected by the MR.NBSTOP field. The 1.5 stop bit configuration is only supported in asynchronous mode.

Figure 19-4. Character Transmit



The characters are sent by writing to the Character to be Transmitted field (THR.TXCHR). The transmitter reports status with the Transmitter Ready (TXRDY) and Transmitter Empty (TXEMPTY) bits in the Channel Status Register (CSR). TXRDY is set when THR is empty. TXEMPTY is set when both THR and the transmit shift register are empty (transmission complete). Both TXRDY and TXEMPTY are cleared when the transmitter is disabled. Writing a character to THR while TXRDY is zero has no effect and the written character will be lost.

Figure 19-5. Transmitter Status



19.6.3.2 Asynchronous Receiver

If the USART is configured in an asynchronous operating mode (MR.SYNC = 0), the receiver will oversample the RXD input line by either 8 or 16 times the baud rate clock, as selected by the Oversampling Mode bit (MR.OVER). If the line is zero for half a bit period (four or eight consecutive samples, respectively), a start bit will be assumed, and the following 8th or 16th sample will determine the logical value on the line, in effect resulting in bit values being determined at the middle of the bit period.

23.7.3 Interlinked Multiple Value Duty Register

Name: IMDUTY

Access Type: Write-only

Offset: 0x08

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
DUTY3							

23	22	21	20	19	18	17	16
DUTY2							

15	14	13	12	11	10	9	8
DUTY1							

7	6	5	4	3	2	1	0
DUTY0							

- **DUTYn: Duty Cycle**

  The value written to DUTY field *n* will be updated for the selected channels. Which channel is selected for updating is defined by the corresponding SEL field in the IMCHSEL register.

  If the value zero is written to DUTY all affected channels will be disabled. In this state the output waveform will be zero all the time.

24.7.5 Channel Register A

Name: RA

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

Offset: 0x14 + 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
RA[15:8]							

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

- **RA: Register A**
  RA contains the Register A value in real time.

Problems with QMatrix acquisition of small sense capacitor voltages can be solved by connecting the negative reference pin (ACREFN) to a voltage divider that produces a small positive voltage (20 mV, typically) to cancel any negative input offset voltage. With a 3.3V supply, recommended values for the voltage divider are Ra (resistor from positive supply to ACREFN) of 8200 ohm and Rb (resistor from ACREFN to ground) of 50 ohm. These recommended values will produce 20 mV on the ACREFN pin, which should generally be enough to compensate for the worst-case negative input offset of the analog comparators.

Unfortunately, such a voltage divider constantly draws a small current from the power supply, reducing battery life in portable applications. In order to prevent this constant power drain, the CAT module provides a voltage divider enable pin (VDIVEN) that can be used for driving the voltage divider. The VDIVEN pin provides power to the voltage divider only when the comparators are actually performing QMatrix comparisons. When the comparators are inactive, the VDIVEN output is zero. This minimizes the power consumed by the voltage divider.

28.7.25 Autonomous Touch Base Count Register

Name: ATBASE

Access Type: Read-only

Offset: 0x6C

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
COUNT[15:8]							

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

- **COUNT: Count value**
  The base count currently stored by the autonomous touch sensor. This is useful for autonomous touch debugging purposes.

selectable by OCD Registers, minimizing the chance that the AUX port will need to be shared with an application.

Table 31-6. Auxiliary Port Signals

Pin Name	Pin Description	Direction	Active Level	Type
MCKO	Trace data output clock	Output		Digital
MDO[5:0]	Trace data output	Output		Digital
MSEO[1:0]	Trace frame control	Output		Digital
EVTI_N	Event In	Input	Low	Digital
EVTO_N	Event Out	Output	Low	Digital

31.3.4 Product Dependencies

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

31.3.4.1 Power Management

The OCD clock operates independently of the CPU clock. If enabled in the Power Manager, the OCD clock (CLK_OCD) will continue running even if the CPU enters a sleep mode that disables the CPU clock.

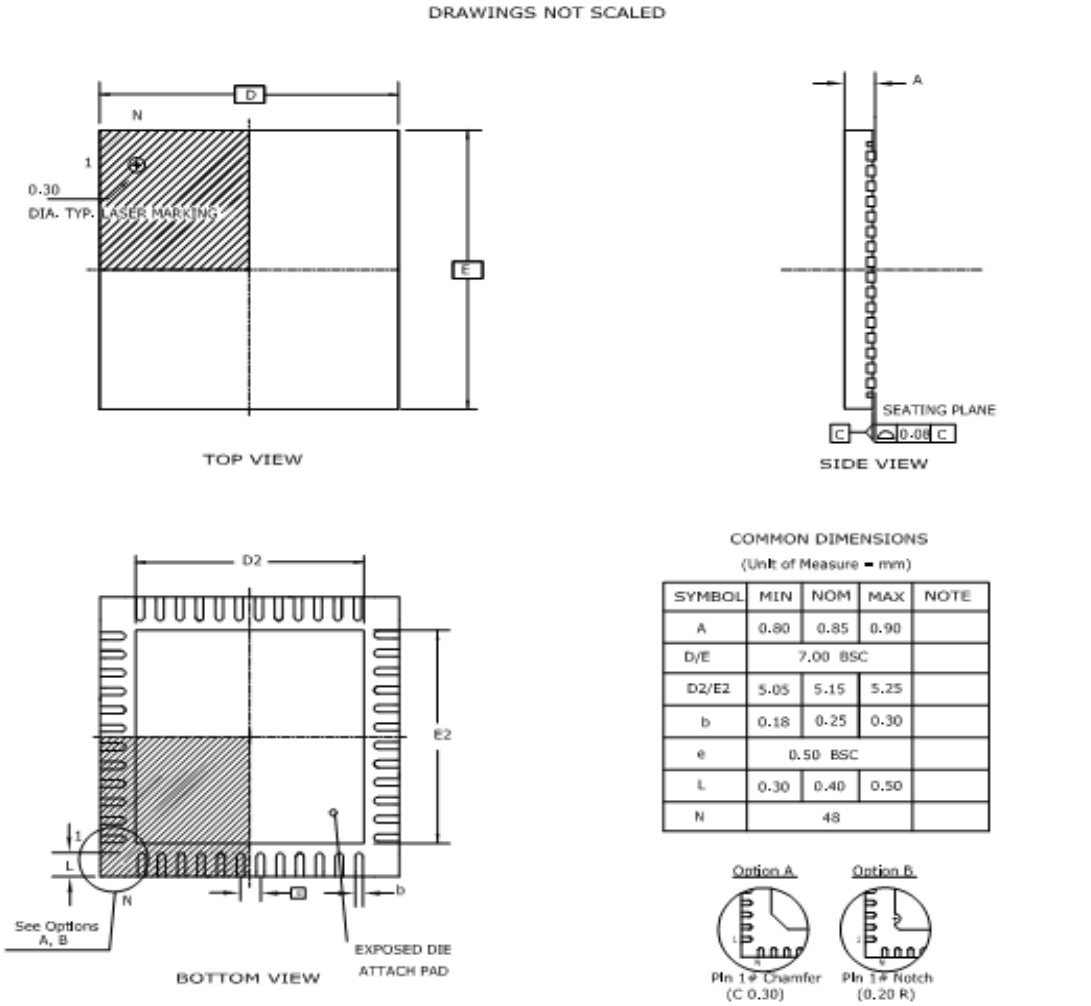
31.3.4.2 Clocks

The OCD has a clock (CLK_OCD) running synchronously with the CPU clock. This clock is generated by the Power Manager. The clock is enabled at reset, and can be disabled by writing to the Power Manager.

31.3.4.3 Interrupt

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

Figure 33-2. QFN-48 Package Drawing



SYMBOL	MIN	NOM	MAX	NOTE
A	0.80	0.85	0.90	
D/E		7.00 BSC		
D2/E2	5.05	5.15	5.25	
b	0.18	0.25	0.30	
e		0.50 BSC		
L	0.30	0.40	0.50	
N		48		

Notes : 1. This drawing is for general information only. Refer to JEDEC Drawing MO-220, Variation VKKD-4, for proper dimensions, tolerances, datums, etc.
2. Dimension b applies to metallized terminal and is measured between 0.15mm and 0.30mm from the terminal tip.
If the terminal has the optical radius on the other end of the terminal, the dimension should not be measured in that radius area.

07/27/2011

Note: The exposed pad is not connected to anything internally, but should be soldered to ground to increase board level reliability.

Table 33-5. Device and Package Maximum Weight

140	mg

Table 33-6. Package Characteristics

Moisture Sensitivity Level	MSL3

Table 33-7. Package Reference

JEDEC Drawing Reference	M0-220
JESD97 Classification	E3