Atmel - ATUC128L3U-Z3UR Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

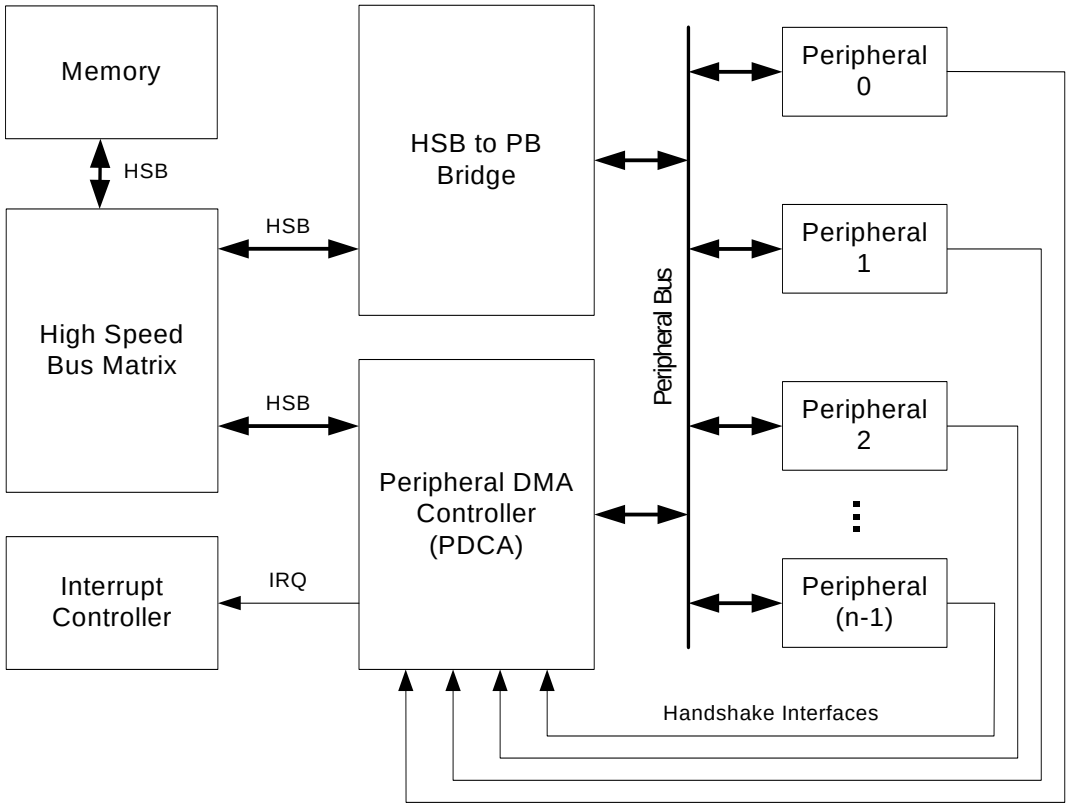
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	51
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	64-VFQFN Exposed Pad
Supplier Device Package	64-QFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/atuc128l3u-z3ur

7.3 Block Diagram

Figure 7-1. PDCA Block Diagram



7.4 Product Dependencies

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

7.4.1 Power Management

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

7.4.2 Clocks

The PDCA has two bus clocks connected: One High Speed Bus clock (CLK_PDCA_HSB) and one Peripheral Bus clock (CLK_PDCA_PB). These clocks are generated by the Power Manager. Both clocks are enabled at reset, and can be disabled in the Power Manager. It is recommended to disable the PDCA before disabling the clocks, to avoid freezing the PDCA in an undefined state.

7.4.3 Interrupts

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

7.7.14 Interrupt Disable Register

Name: IDR

Access Type: Write-only

Offset: 0x024 + n*0x040

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
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	-	TERR	TRC	RCZ

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 IMR.

7.7.15 Interrupt Mask Register

Name: IMR

Access Type: Read-only

Offset: 0x028 + n*0x040

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
-	-	-	-	-	-	-	-

7	6	5	4	3	2	1	0
-	-	-	-	-	TERR	TRC	RCZ

0: The corresponding interrupt is disabled.

1: The corresponding interrupt is enabled.

A bit in this register is cleared when the corresponding bit in IDR is written to one.

A bit in this register is set when the corresponding bit in IER is written to one.

7.7.29 Performance Channel 1 Write Max Latency

Name: PWLAT1

Access Type: Read/Write

Offset: 0x830

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
LAT[15:8]							

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

- **LAT: Maximum Transfer Initiation Cycles Counted Since Last Reset**
  Clock cycles are counted using the CLK_PDCA_HSB clock
  This counter is saturating. The register is reset only when PCONTROL.CH1RES is written to one.

8.7.2.12 *Endpoint n Status Clear Register*

Register Name: UESTAnCLR, n in [0..6]

Access Type: Write-Only

Offset: 0x0160 + (n * 0x04)

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
-	-	-	-	RAMACERIC	-	-	-

7	6	5	4	3	2	1	0
-	STALLEDIC/ CRCERRIC	-	NAKINIC	NAKOUTIC	RXSTPIC/ ERRORFIC	RXOUTIC	TXINIC

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 UESTA.
These bits always read as zero.

Table 12-3. Interrupt Request Signal Map

8	0	System Control Interface	SCIF
9	0	Asynchronous Timer	AST ALARM
10	0	Asynchronous Timer	AST PER
	1	Asynchronous Timer	AST OVF
	2	Asynchronous Timer	AST READY
	3	Asynchronous Timer	AST CLKREADY
11	0	External Interrupt Controller	EIC 1
	1	External Interrupt Controller	EIC 2
	2	External Interrupt Controller	EIC 3
	3	External Interrupt Controller	EIC 4
12	0	External Interrupt Controller	EIC 5
13	0	Frequency Meter	FREQM
14	0	General-Purpose Input/Output Controller	GPIO 0
	1	General-Purpose Input/Output Controller	GPIO 1
	2	General-Purpose Input/Output Controller	GPIO 2
	3	General-Purpose Input/Output Controller	GPIO 3
	4	General-Purpose Input/Output Controller	GPIO 4
	5	General-Purpose Input/Output Controller	GPIO 5
	6	General-Purpose Input/Output Controller	GPIO 6
	7	General-Purpose Input/Output Controller	GPIO 7
15	0	Universal Synchronous Asynchronous Receiver Transmitter	USART0
16	0	Universal Synchronous Asynchronous Receiver Transmitter	USART1
17	0	Universal Synchronous Asynchronous Receiver Transmitter	USART2
18	0	Universal Synchronous Asynchronous Receiver Transmitter	USART3
19	0	Serial Peripheral Interface	SPI
20	0	Two-wire Master Interface	TWIM0
21	0	Two-wire Master Interface	TWIM1
22	0	Two-wire Slave Interface	TWIS0
23	0	Two-wire Slave Interface	TWIS1
24	0	Pulse Width Modulation Controller	PWMA
25	0	Timer/Counter	TC00
	1	Timer/Counter	TC01
	2	Timer/Counter	TC02

13.7.13 Interrupt Clear Register

Name: ICR

Access Type: Write-only

Offset: 0x0D0

Reset Value: 0x00000000

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

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 ISR.

14.6.20 Temperature Sensor Configuration Register

Name: TSENS

Access Type: Read/Write

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
-	-	-	-	-	-	-	-

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

- **EN: Temperature Sensor Enable**
  0: The Temperature Sensor is disabled.
  1: The Temperature Sensor 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.

16.6 User Interface

Table 16-1. WDT Register Memory Map

Offset	Register	Register Name	Access	Reset
0x000	Control Register	CTRL	Read/Write	0x00010080
0x004	Clear Register	CLR	Write-only	0x00000000
0x008	Status Register	SR	Read-only	0x00000003
0x3FC	Version Register	VERSION	Read-only	-(1)

Note: 1. The reset value for this register is device specific. Please refer to the Module Configuration section at the end of this chapter.

18.6.6 Interrupt Disable Register

Name: IDR

Access Type: Write-only

Offset: 0x014

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
-	-	-	-	-	-	-	-

7	6	5	4	3	2	1	0
-	-	-	-	-	-	RCLKRDY	DONE

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 IMR.

19.7.9 Output Value Register

Name: OVR

Access: Read/Write, Set, Clear, Toggle

Offset: 0x050, 0x054, 0x058, 0x05C

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: Output Value**
  0: The value to be driven on the GPIO pin is 0.
  1: The value to be driven on the GPIO pin is 1.

Figure 20-16. Receiver Behavior when Operating with Hardware Handshaking

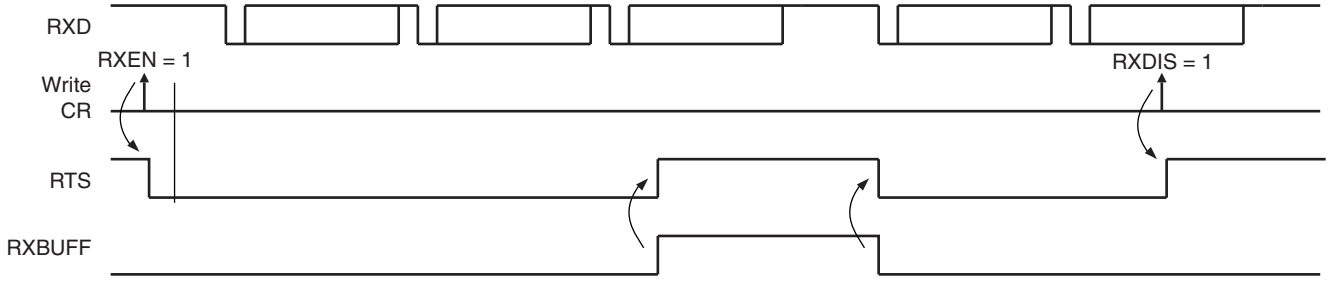


Figure 20-17. Transmitter Behavior when Operating with Hardware Handshaking

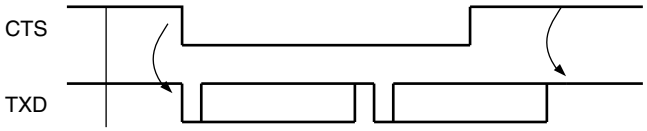


Figure 20-18.

20.6.4 SPI Mode

The USART features a Serial Peripheral Interface (SPI) link compliant mode, supporting synchronous, full-duplex communication, in both master and slave mode. Writing 0xE (master) or 0xF (slave) to MR.MODE will enable this mode. A SPI in master mode controls the data flow to and from the other SPI devices, who are in slave mode. It is possible to let devices take turns being masters (aka multi-master protocol), and one master may shift data simultaneously into several slaves, but only one slave may respond at a time. A slave is selected when its slave select (NSS) signal has been raised by the master. The USART can only generate one NSS signal, and it is possible to use standard I/O lines to address more than one slave.

20.6.4.1 Modes of Operation

The SPI system consists of two data lines and two control lines:

- Master Out Slave In (MOSI): This line supplies the data shifted from master to slave. In master mode this is connected to TXD, and in slave mode to RXD.
- Master In Slave Out (MISO): This line supplies the data shifted from slave to master. In master mode this is connected to RXD, and in slave mode to TXD.
- Serial Clock (CLK): This is controlled by the master. One period per bit transmission. In both modes this is connected to CLK.
- Slave Select (NSS): This control line allows the master to select or deselect a slave. In master mode this is connected to RTS, and in slave mode to CTS.

Changing SPI mode after initial configuration has to be followed by a transceiver software reset in order to avoid unpredictable behavior.

20.6.4.2 Baud Rate

The baud rate generator operates as described in ”Baud Rate in Synchronous and SPI Mode” on page 439, with the following requirements:

In SPI Master Mode:

- **BITS: Bits Per Transfer**
  The BITS field determines the number of data bits transferred. Reserved values should not be used.

BITS	Bits Per Transfer
0000	8
0001	9
0010	10
0011	11
0100	12
0101	13
0110	14
0111	15
1000	16
1001	4
1010	5
1011	6
1100	7
1101	Reserved
1110	Reserved
1111	Reserved

- **CSAAT: Chip Select Active After Transfer**
  1: The Peripheral Chip Select does not rise after the last transfer is achieved. It remains active until a new transfer is requested on a different chip select.
  0: The Peripheral Chip Select Line rises as soon as the last transfer is achieved.
- **CSNAAT: Chip Select Not Active After Transfer (Ignored if CSAAT = 1)**
  0: The Peripheral Chip Select does not rise between two transfers if the TDR is reloaded before the end of the first transfer and if the two transfers occur on the same Chip Select.
  1: The Peripheral Chip Select rises systematically between each transfer performed on the same slave for a minimal duration of:

  $\frac{DLYBCS}{CLKSPI}$ (if DLYBCT field is different from 0)

  $\frac{DLYBCS + 1}{CLK\overline{SPI}}$ (if DLYBCT field equals 0)
- **NCPHA: Clock Phase**
  1: Data is captured after the leading (inactive-to-active) edge of SPCK and changed on the trailing (active-to-inactive) edge of SPCK.
  0: Data is changed on the leading (inactive-to-active) edge of SPCK and captured after the trailing (active-to-inactive) edge of SPCK.
  NCPHA determines which edge of SPCK causes data to change and which edge causes data to be captured. NCPHA is used with CPOL to produce the required clock/data relationship between master and slave devices.
- **CPOL: Clock Polarity**
  1: The inactive state value of SPCK is logic level one.
  0: The inactive state value of SPCK is logic level zero.

Below, Table 23-2 lists the compatibility level of the Atmel Two-wire Slave Interface and a full SMBus compatible device.

Table 23-2. Atmel TWIS Compatibility with SMBus Standard

SMBus Standard	Atmel TWIS
Bus Timeouts	Supported
Address Resolution Protocol	Supported
Alert	Supported
Packet Error Checking	Supported

23.3 List of Abbreviations

Table 23-3. Abbreviations

Abbreviation	Description
TWI	Two-wire Interface
A	Acknowledge
NA	Non Acknowledge
P	Stop
S	Start
Sr	Repeated Start
SADR	Slave Address
ADR	Any address except SADR
R	Read
W	Write

23.4 Block Diagram

Figure 23-1. Block Diagram

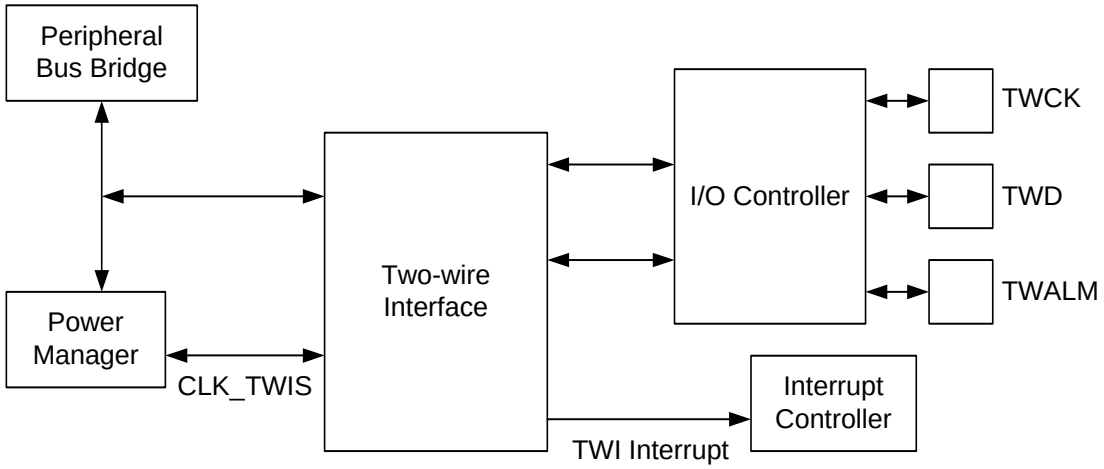
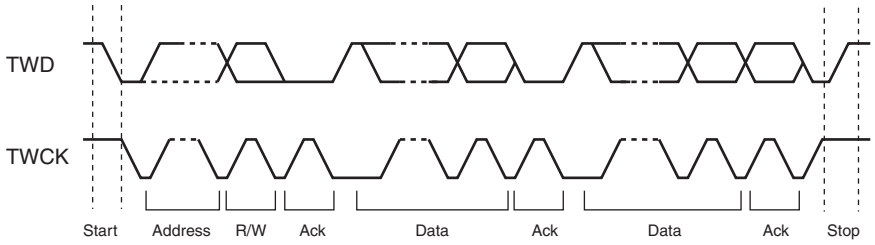


Figure 23-4. Transfer Format



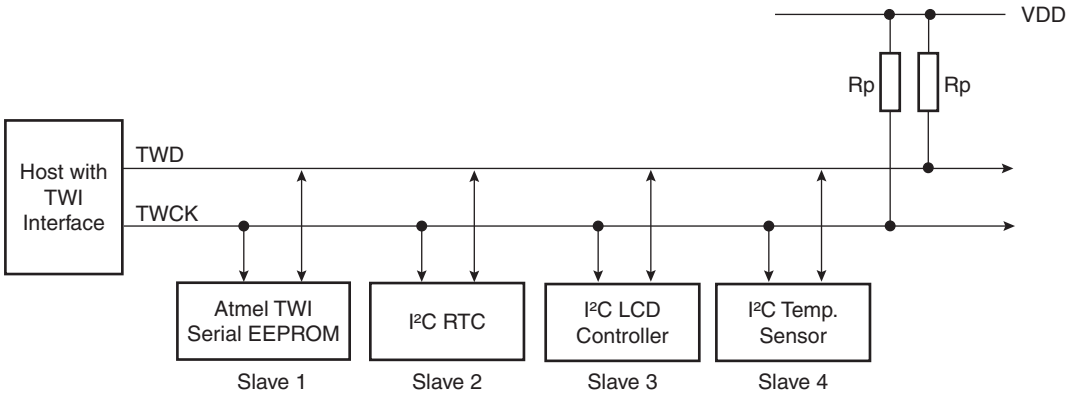
23.8.2 Operation

The TWIS has two modes of operation:

- Slave transmitter mode
- Slave receiver mode

A master is a device which starts and stops a transfer and generates the TWCK clock. A slave is assigned an address and responds to requests from the master. These modes are described in the following chapters.

Figure 23-5. Typical Application Block Diagram



Rp: Pull up value as given by the I²C Standard

23.8.2.1 Bus Timing

The Timing Register (TR) is used to control the timing of bus signals driven by the TWIS. TR describes bus timings as a function of cycles of the prescaled CLK_TWIS. The clock prescaling can be selected through TR.EXP.

$$f_{PRESCALED} = \frac{f_{CLK\_TWIS}}{2^{(EXP+1)}}$$

TR has the following fields:

TLOWS: Prescaled clock cycles used to time SMBUS timeout $T_{LOW:SEXT}$.

30.9.11 Window Configuration Register

Name: CONFWn

Access Type: Read/Write

Offset: 0x80,0x84,0x88,0x8C

Reset Value: 0x00000000

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

- **WFEN: Window Mode Enable**
  - 0: The window mode is disabled.
  - 1: The window mode is enabled.
- **WEVEN: Window Event Enable**
  - 0: Event from awout is disabled.
  - 1: Event from awout is enabled.
- **WEVSRC: Event Source Selection for Window Mode**
  - 000: Event on acwout rising edge.
  - 001: Event on acwout falling edge.
  - 010: Event on awout rising or falling edge.
  - 011: Inside window.
  - 100: Outside window.
  - 101: Measure done.
  - 110-111: Reserved.
- **WIS: Window Mode Interrupt Settings**
  - 00: Window interrupt as soon as the input voltage is inside the window.
  - 01: Window interrupt as soon as the input voltage is outside the window.
  - 10: Window interrupt on toggle of window compare output.
  - 11: Window interrupt when evaluation of input voltage is done.

33.7 User Interface

Table 33-2. aWire UART user interface Register Memory Map

Offset	Register	Register Name	Access	Reset
0x00	Control Register	CTRL	Read/Write	0x00000000
0x04	Status Register	SR	Read-only	0x00000000
0x08	Status Clear Register	SCR	Write-only	-
0x0C	Interrupt Enable Register	IER	Write-only	-
0x10	Interrupt Disable Register	IDR	Write-only	-
0x14	Interrupt Mask Register	IMR	Read-only	0x00000000
0x18	Receive Holding Register	RHR	Read-only	0x00000000
0x1C	Transmit Holding Register	THR	Read/Write	0x00000000
0x20	Baud Rate Register	BRR	Read/Write	0x00000000
0x24	Version Register	VERSION	Read-only	-(1)
0x28	Clock Request Register	CLKR	Read/Write	0x00000000

Note: 1. The reset values are device specific. Please refer to the Module Configuration section at the end of this chapter.

33.7.5 Interrupt Disable Register

Name: IDR

Access Type: Write-only

Offset: 0x10

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
-	-	TRMIS	-	-	OVERRUN	DREADYINT	READYINT
7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

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 IMR.

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.

Under certain circumstances, the lock signal from the Phase Locked Loop (PLL) oscillator may not go back to zero after the PLL oscillator has been disabled. This can cause the propagation of clock signals with the wrong frequency to parts of the system that use the PLL clock.

Fix/Workaround

PLL must be turned off before entering STOP, DEEPSTOP or STATIC sleep modes. If PLL has been turned off, a delay of 30us must be observed after the PLL has been enabled again before the SCIF.PLL0LOCK bit can be used as a valid indication that the PLL is locked.

3. **PLLCOUNT value larger than zero can cause PLLEN glitch**

   Initializing the PLLCOUNT with a value greater than zero creates a glitch on the PLLEN signal during asynchronous wake up.

   Fix/Workaround

   The lock-masking mechanism for the PLL should not be used.
   The PLLCOUNT field of the PLL Control Register should always be written to zero.

4. **RCSYS is not calibrated**

   The RCSYS is not calibrated and will run faster than 115.2kHz. Frequencies around 150kHz can be expected.

   Fix/Workaround

   If a known clock source is available the RCSYS can be runtime calibrated by using the frequency meter (FREQM) and tuning the RCSYS by writing to the RCCR register in SCIF.

5. **Writing 0x5A5A5A5A to the SCIF memory range will enable the SCIF UNLOCK feature**

   The SCIF UNLOCK feature will be enabled if the value 0x5A5A5A5A is written to any location in the SCIF memory range.

   Fix/Workaround

   None.

38.5.4 WDT

1. **Clearing the Watchdog Timer (WDT) counter in second half of timeout period will issue a Watchdog reset**

   If the WDT counter is cleared in the second half of the timeout period, the WDT will immediately issue a Watchdog reset.

   Fix/Workaround

   Use twice as long timeout period as needed and clear the WDT counter within the first half of the timeout period. If the WDT counter is cleared after the first half of the timeout period, you will get a Watchdog reset immediately. If the WDT counter is not cleared at all, the time before the reset will be twice as long as needed.

2. **WDT Control Register does not have synchronization feedback**

   When writing to the Timeout Prescale Select (PSEL), Time Ban Prescale Select (TBAN), Enable (EN), or WDT Mode (MODE) fieldss of the WDT Control Register (CTRL), a synchronizer is started to propagate the values to the WDT clcok domain. This synchronization takes a finite amount of time, but only the status of the synchronization of the EN bit is reflected back to the user. Writing to the synchronized fields during synchronization can lead to undefined behavior.

   Fix/Workaround

   -When writing to the affected fields, the user must ensure a wait corresponding to 2 clock cycles of both the WDT peripheral bus clock and the selected WDT clock source.
   -When doing writes that changes the EN bit, the EN bit can be read back until it reflects the written value.