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-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atuc128l3u-aut

Table 3-4. Nexus OCD AUX Port Connections

Pin	AXS=1	AXS=0
EVTO_N	PA04	PA04
MCKO	PA06	PB01
MSEO[1]	PA07	PB11
MSEO[0]	PA11	PB12

3.2.5 Oscillator Pinout

The oscillators are not mapped to the normal GPIO functions and their muxings are controlled by registers in the System Control Interface (SCIF). Please refer to the SCIF chapter for more information about this.

Table 3-5. Oscillator Pinout

48-pin	64-pin	Pin Name	Oscillator Pin
3	3	PA08	XIN0
46	62	PA10	XIN32
26	34	PA13	XIN32_2
2	2	PA09	XOUT0
47	63	PA12	XOUT32
25	33	PA20	XOUT32_2

3.2.6 Other Functions

The functions listed in Table 3-6 are not mapped to the normal GPIO functions. The aWire DATA pin will only be active after the aWire is enabled. The aWire DATAOUT pin will only be active after the aWire is enabled and the 2_PIN_MODE command has been sent. The WAKE_N pin is always enabled. Please refer to Section 6.1.4.2 on page 44 for constraints on the WAKE_N pin.

Table 3-6. Other Functions

48-pin	64-pin	Pin Name	Function
27	35	PA11	WAKE_N
22	30	RESET_N	aWire DATA
11	15	PA00	aWire DATAOUT

single cycle. Load and store instructions have several different formats in order to reduce code size and speed up execution.

The register file is organized as sixteen 32-bit registers and includes the Program Counter, the Link Register, and the Stack Pointer. In addition, register R12 is designed to hold return values from function calls and is used implicitly by some instructions.

4.3 The AVR32UC CPU

The AVR32UC CPU targets low- and medium-performance applications, and provides an advanced On-Chip Debug (OCD) system, no caches, and a Memory Protection Unit (MPU). Java acceleration hardware is not implemented.

AVR32UC provides three memory interfaces, one High Speed Bus master for instruction fetch, one High Speed Bus master for data access, and one High Speed Bus slave interface allowing other bus masters to access data RAMs internal to the CPU. Keeping data RAMs internal to the CPU allows fast access to the RAMs, reduces latency, and guarantees deterministic timing. Also, power consumption is reduced by not needing a full High Speed Bus access for memory accesses. A dedicated data RAM interface is provided for communicating with the internal data RAMs.

A local bus interface is provided for connecting the CPU to device-specific high-speed systems, such as floating-point units and I/O controller ports. This local bus has to be enabled by writing a one to the LOCEN bit in the CPUCR system register. The local bus is able to transfer data between the CPU and the local bus slave in a single clock cycle. The local bus has a dedicated memory range allocated to it, and data transfers are performed using regular load and store instructions. Details on which devices that are mapped into the local bus space is given in the CPU Local Bus section in the Memories chapter.

Figure 4-1 on page 23 displays the contents of AVR32UC.

7.7.19 Performance Channel 0 Read Stall Cycles

Name: PRSTALL0

Access Type: Read-only

Offset: 0x808

Reset Value: 0x00000000

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

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

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

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

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

8.7.1.2 General Status Register

Register Name: USBSTA

Access Type: Read-Only

Offset: 0x0804

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
-	CLKUSABLE	SPEED		-	-	-	-

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

- **CLKUSABLE: Generic Clock Usable**
  This bit is cleared when the USB generic clock is not usable.
  This bit is set when the USB generic clock (that should be 48 Mhz) is usable.
- **SPEED: Speed Status**
  This field is set according to the controller speed mode.

SPEED	Speed Status
00	full-speed mode
01	Reserved
10	low-speed mode
11	Reserved

13.7.8 Unlock Register

Name: UNLOCK

Access Type: Write-only

Offset: 0x058

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
KEY							

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

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

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

To unlock a write protected register, first write to the UNLOCK register with the address of the register to unlock in the ADDR field and 0xAA in the KEY field. Then, in the next PB access write to the register specified in the ADDR field.

- **KEY: Unlock Key**
  Write this bit field to 0xAA to enable unlock.
- **ADDR: Unlock Address**
  Write the address of the register to unlock to this field.

14.6.4 Interrupt Status Register

Name: ISR

Access Type: Read-only

Offset: 0x000C

Reset Value: 0x00000000

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

23	22	21	20	19	18	17	16
-	-	-	-	-	PLLLOCKLOST0	PLLLOCK0	BRIFARDY

15	14	13	12	11	10	9	8
DFLL0RCS	DFLL0RDY	DFLL0LOCKLOSTA	DFLL0LOCKLOSTF	DFLL0LOCKLOSTC	DFLL0LOCKA	DFLL0LOCKF	DFLL0LOCKC

7	6	5	4	3	2	1	0
BODDET	SM33DET	VREGOK	-	-	-	OSC0RDY	OSC32RDY

0: The corresponding interrupt is cleared.

1: The corresponding interrupt is pending.

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

A bit in this register is set when the corresponding interrupt occurs.

15.6.17 Event Mask Register

Name: EVM

Access Type: Read-only

Offset: 0x50

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

0: The corresponding peripheral event is disabled.

1: The corresponding peripheral event is enabled.

This bit is cleared when the corresponding bit in EVD is written to one.

This bit is set when the corresponding bit in EVE is written to one.

16. Watchdog Timer (WDT)

Rev: 4.1.0.0

16.1 Features

- **Watchdog Timer counter with 32-bit counter**
- **Timing window watchdog**
- **Clocked from system RC oscillator or the 32 KHz crystal oscillator**
- **Configuration lock**
- **WDT may be enabled at reset by a fuse**

16.2 Overview

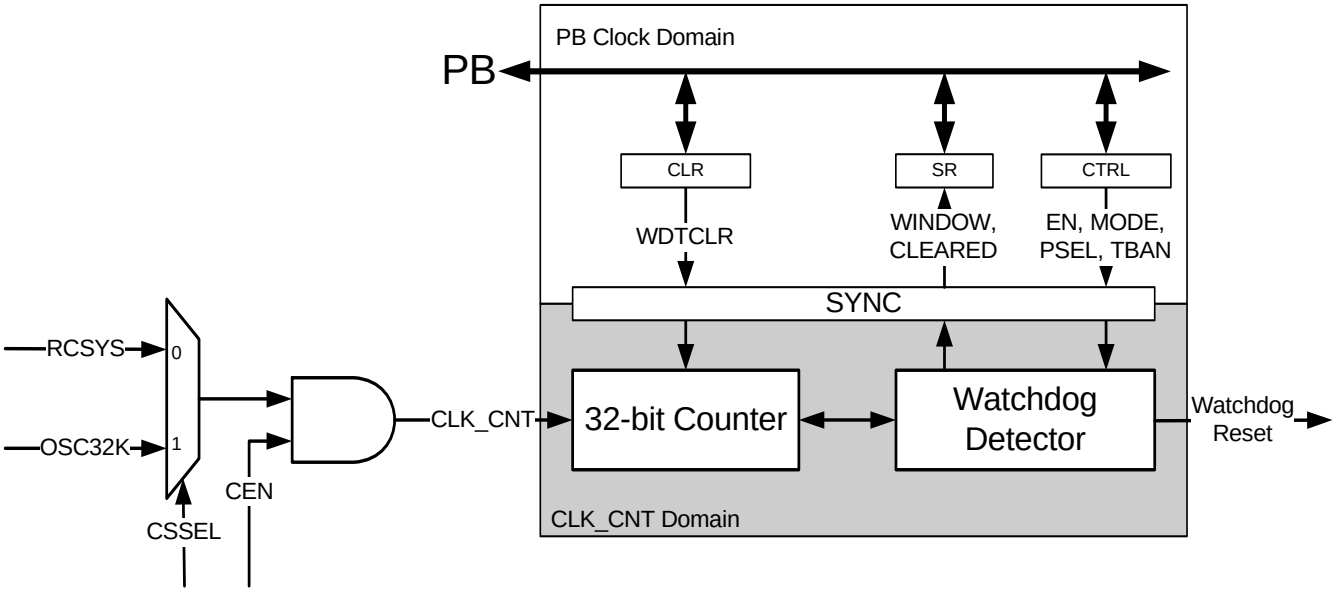
The Watchdog Timer (WDT) will reset the device unless it is periodically serviced by the software. This allows the device to recover from a condition that has caused the system to be unstable.

The WDT has an internal counter clocked from the system RC oscillator or the 32 KHz crystal oscillator.

The WDT counter must be periodically cleared by software to avoid a watchdog reset. If the WDT timer is not cleared correctly, the device will reset and start executing from the boot vector.

16.3 Block Diagram

Figure 16-1. WDT Block Diagram



16.4 Product Dependencies

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

17.7.8 Level Register

Name: LEVEL

Access Type: Read/Write

Offset: 0x01C

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
-	INT30	INT29	INT28	INT27	INT26	INT25	INT24
23	22	21	20	19	18	17	16
INT23	INT22	INT21	INT20	INT19	INT18	INT17	INT16
15	14	13	12	11	10	9	8
INT15	INT14	INT13	INT12	INT11	INT10	INT9	INT8
7	6	5	4	3	2	1	0
INT7	INT6	INT5	INT4	INT3	INT2	INT1	NMI

- **INTn: External Interrupt n**
  0: The external interrupt triggers on low level.
  1: The external interrupt triggers on high level.
  Please refer to the Module Configuration section for the number of external interrupts.
- **NMI: Non-Maskable Interrupt**
  0: The Non-Maskable Interrupt triggers on low level.
  1: The Non-Maskable Interrupt triggers on high level.

19.7.14 Interrupt Mode Register 1

Name: IMR1

Access: Read/Write, Set, Clear, Toggle

Offset: 0x0B0, 0x0B4, 0x0B8, 0x0BC

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 Mode Bit 1**

{IMR1, IMR0}	Interrupt Mode
00	Pin Change
01	Rising Edge
10	Falling Edge
11	Reserved

20.7.3 Interrupt Enable Register

Name: IER

Access Type: Write-only

Offset: 0x8

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
–	–	LINSNRE	LINCE	LINIPE	LINISFE	LINBE	–
23	22	21	20	19	18	17	16
–	–	–	–	CTSIC	–	–	–
15	14	13	12	11	10	9	8
LINTC	LINID	NACK/LINBK	RXBUFF	–	ITER/UNRE	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	–	–	RXBRK	TXRDY	RXRDY

Writing a zero to a bit in this register has no effect.
Writing a one to a bit in this register will set the corresponding bit in IMR.

20.7.6 Channel Status Register

Name: CSR

Access Type: Read-only

Offset: 0x14

Reset Value: 0x00000000

31	30	29	28	27	26	25	24
–	–	LINSNRE	LINCE	LINIPE	LINISFE	LINBE	–
23	22	21	20	19	18	17	16
CTS	–	–	–	CTSIC	–	–	–
15	14	13	12	11	10	9	8
LINTC	LINID	NACK/LINBK	RXBUFF	–	ITER/UNRE	TXEMPTY	TIMEOUT
7	6	5	4	3	2	1	0
PARE	FRAME	OVRE	–	–	RXBRK	TXRDY	RXRDY

- **LINSNRE: LIN Slave Not Responding Error**
  0: No LIN Slave Not Responding Error has been detected since the last RSTSTA.
  1: A LIN Slave Not Responding Error has been detected since the last RSTSTA.
- **LINCE: LIN Checksum Error**
  0: No LIN Checksum Error has been detected since the last RSTSTA.
  1: A LIN Checksum Error has been detected since the last RSTSTA.
- **LINIPE: LIN Identifier Parity Error**
  0: No LIN Identifier Parity Error has been detected since the last RSTSTA.
  1: A LIN Identifier Parity Error has been detected since the last RSTSTA.
- **LINISFE: LIN Inconsistent Sync Field Error**
  0: No LIN Inconsistent Sync Field Error has been detected since the last RSTSTA
  1: The USART is configured as a Slave node and a LIN Inconsistent Sync Field Error has been detected since the last RSTSTA.
- **LINBE: LIN Bit Error**
  0: No Bit Error has been detected since the last RSTSTA.
  1: A Bit Error has been detected since the last RSTSTA.
- **CTS: Image of CTS Input**
  0: CTS is low.
  1: CTS is high.
- **CTSIC: Clear to Send Input Change Flag**
  0: No change has been detected on the CTS pin since the last CSR read.
  1: At least one change has been detected on the CTS pin since the last CSR read.
- **LINTC: LIN Transfer Completed**
  0: The USART is either idle or a LIN transfer is ongoing.
  1: A LIN transfer has been completed since the last RSTSTA.
- **LINID: LIN Identifier**
  0: No LIN Identifier has been sent or received.
  1: A LIN Identifier has been sent (master) or received (slave), since the last RSTSTA.
- **NACK: Non Acknowledge**
  0: No Non Acknowledge has been detected since the last RSTNACK.
  1: At least one Non Acknowledge has been detected since the last RSTNACK.
- **RXBUFF: Reception Buffer Full**
  0: The Buffer Full signal from the Peripheral DMA Controller channel is inactive.

21.7.3.3 Clock generation

The SPI Baud rate clock is generated by dividing the CLK_SPI , by a value between 1 and 255.

This allows a maximum operating baud rate at up to CLK_SPI and a minimum operating baud rate of CLK_SPI divided by 255.

Writing the Serial Clock Baud Rate field in the CSRn registers (CSRn.SCBR) to zero is forbidden. Triggering a transfer while CSRn.SCBR is zero can lead to unpredictable results.

At reset, CSRn.SCBR is zero and the user has to configure it at a valid value before performing the first transfer.

The divisor can be defined independently for each chip select, as it has to be configured in the CSRn.SCBR field. This allows the SPI to automatically adapt the baud rate for each interfaced peripheral without reprogramming.

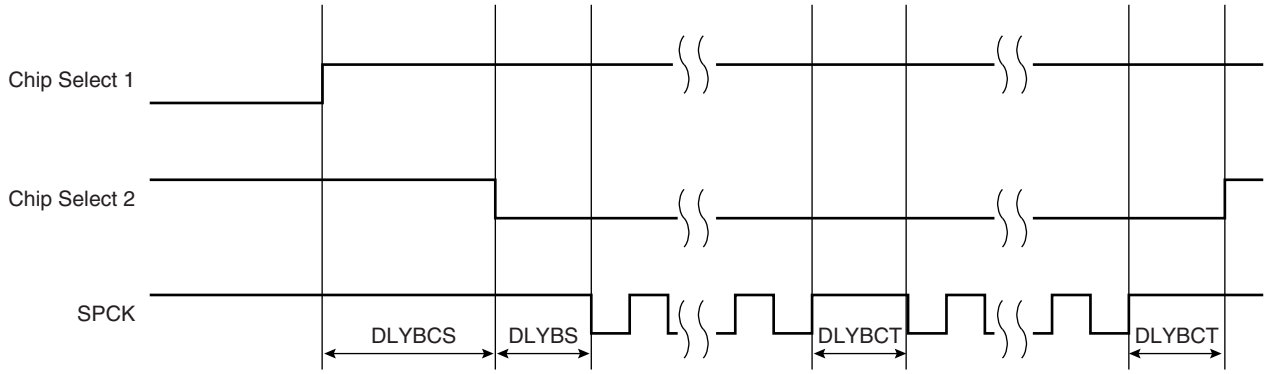
21.7.3.4 Transfer delays

Figure 21-7 on page 493 shows a chip select transfer change and consecutive transfers on the same chip select. Three delays can be configured to modify the transfer waveforms:

- The delay between chip selects, programmable only once for all the chip selects by writing to the Delay Between Chip Selects field in the MR register (MR.DLYBCS). Allows insertion of a delay between release of one chip select and before assertion of a new one.
- The delay before SPCK, independently programmable for each chip select by writing the Delay Before SPCK field in the CSRn registers (CSRn.DLYBS). Allows the start of SPCK to be delayed after the chip select has been asserted.
- The delay between consecutive transfers, independently programmable for each chip select by writing the Delay Between Consecutive Transfers field in the CSRn registers (CSRn.DLYBCT). Allows insertion of a delay between two transfers occurring on the same chip select

These delays allow the SPI to be adapted to the interfaced peripherals and their speed and bus release time.

Figure 21-7. Programmable Delays



1. Wait until RHR is empty, stretching low period of TWCK. SR.RXRDY indicates the state of RHR. Software or the Peripheral DMA Controller must read any data byte present in RHR.
2. Release TWCK generating a clock that the slave uses to transmit a data byte.
3. Place the received data byte in RHR, set RXRDY.
4. If NBYTES=0, generate a NAK after the data byte, otherwise generate an ACK.
5. Decrement NBYTES
6. If (NBYTES==0) and STOP=1, transmit STOP condition.

Writing CMDR with START=STOP=1 and NBYTES=0 will generate a transmission with no data bytes, ie START, DADR+R, STOP

The TWI transfers require the master to acknowledge each received data byte. During the acknowledge clock pulse (9th pulse), the slave releases the data line (HIGH), enabling the master to pull it down in order to generate the acknowledge. All data bytes except the last are acknowledged by the master. Not acknowledging the last byte informs the slave that the transfer is finished.

RXRDY is used as Receive Ready for the Peripheral DMA Controller receive channel.

Figure 22-8. Master Read with One Data Byte

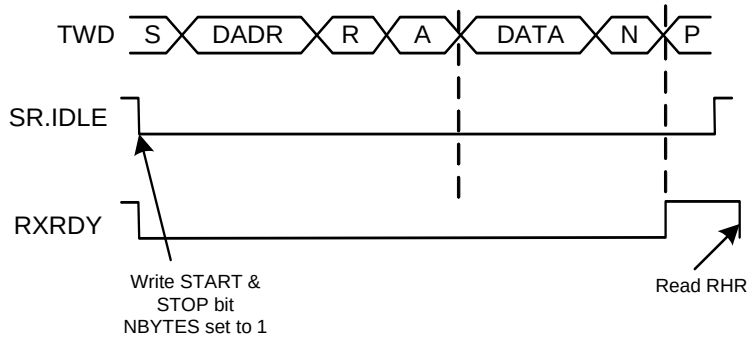
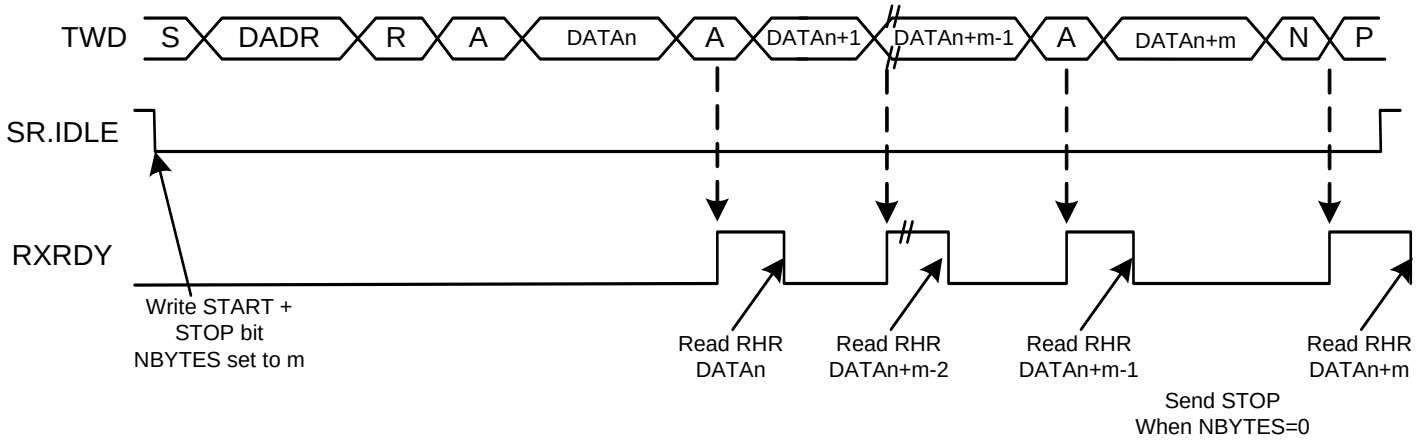


Figure 22-9. Master Read with Multiple Data Bytes



22.9.5 Next Command Register

Name: NCMDR

Access Type: Read/Write

Offset: 0x10

Reset Value: 0x00000000

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

23	22	21	20	19	18	17	16
NBYTES							

15	14	13	12	11	10	9	8
VALID	STOP	START	REPSAME	TENBIT	SADR[9:7]		

7	6	5	4	3	2	1	0
SADR[6:0]							READ

This register is identical to CMDR. When the VALID bit in CMDR becomes 0, the content of NCMDR is copied into CMDR, clearing the VALID bit in NCMDR. If the VALID bit in CMDR is cleared when NCMDR is written, the content is copied immediately.

26.7.1 Channel Control Register

Name: CCR

Access Type: Write-only

Offset: 0x00 + 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
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	-	SWTRG	CLKDIS	CLKEN

- **SWTRG: Software Trigger Command**
  1: Writing a one to this bit will perform a software trigger: the counter is reset and the clock is started.
  0: Writing a zero to this bit has no effect.
- **CLKDIS: Counter Clock Disable Command**
  1: Writing a one to this bit will disable the clock.
  0: Writing a zero to this bit has no effect.
- **CLKEN: Counter Clock Enable Command**
  1: Writing a one to this bit will enable the clock if CLKDIS is not one.
  0: Writing a zero to this bit has no effect.

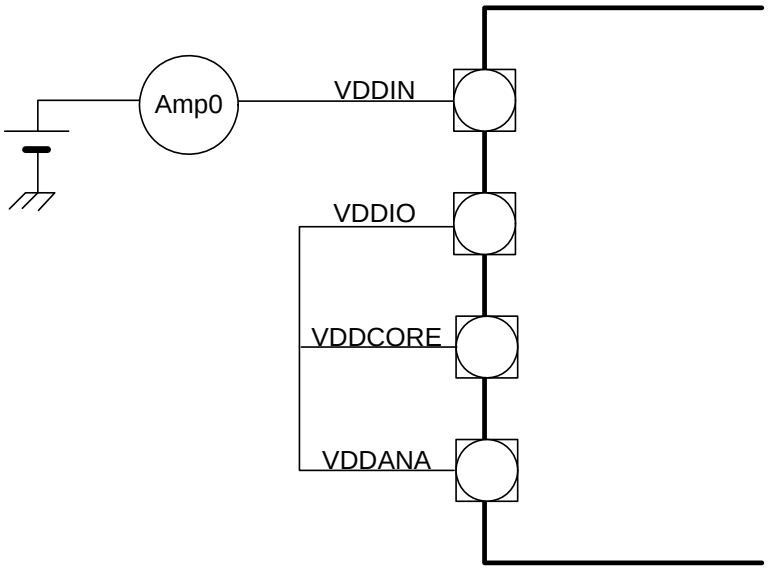
  - $V_{VDDCORE}$ = 1.62V, supplied by the internal regulator
  - Corresponds to the 3.3V supply mode with 1.8V regulated I/O lines, please refer to the Supply and Startup Considerations section for more details
    - Equivalent to the 3.3V single supply mode
    - Consumption in 1.8V single supply mode can be estimated by subtracting the regulator static current
- Operating conditions, external core supply (Figure 35-2) - used only when noted
  - V_{VDDIN} = $V_{VDDCORE}$ = 1.8V
  - Corresponds to the 1.8V single supply mode, please refer to the Supply and Startup Considerations section for more details
- T_A = 25°C
- Oscillators
  - OSC0 (crystal oscillator) stopped
  - OSC32K (32KHz crystal oscillator) running with external 32KHz crystal
  - DFLL running at 50MHz with OSC32K as reference
- Clocks
  - DFLL used as main clock source
  - CPU, HSB, and PBB clocks undivided
  - PBA clock divided by 4
  - The following peripheral clocks running
    - PM, SCIF, AST, FLASHCDW, PBA bridge
  - All other peripheral clocks stopped
- I/Os are inactive with internal pull-up
- Flash enabled in high speed mode
- POR18 enabled
- POR33 disabled

Table 35-5. Power Consumption for Different Operating Modes

Mode	Conditions	Measured on	Consumption Typ	Unit
Active[1]	CPU running a recursive Fibonacci algorithm	Amp0	300	µA/MHz
	CPU running a division algorithm		174	
Idle[1]			96	
Frozen[1]			57	
Standby[1]			46	
Stop			38	µA
DeepStop			25	
Static	-OSC32K and AST stopped -Internal core supply		14	
	-OSC32K running -AST running at 1KHz -External core supply (Figure 35-2)		7.3	
	-OSC32K and AST stopped -External core supply (Figure 35-2)		6.7	
Shutdown	-OSC32K running -AST running at 1KHz		800	nA
	AST and OSC32K stopped		220	

Note: 1. These numbers are valid for the measured condition only and must not be extrapolated to other frequencies.

Figure 35-1. Measurement Schematic, Internal Core Supply



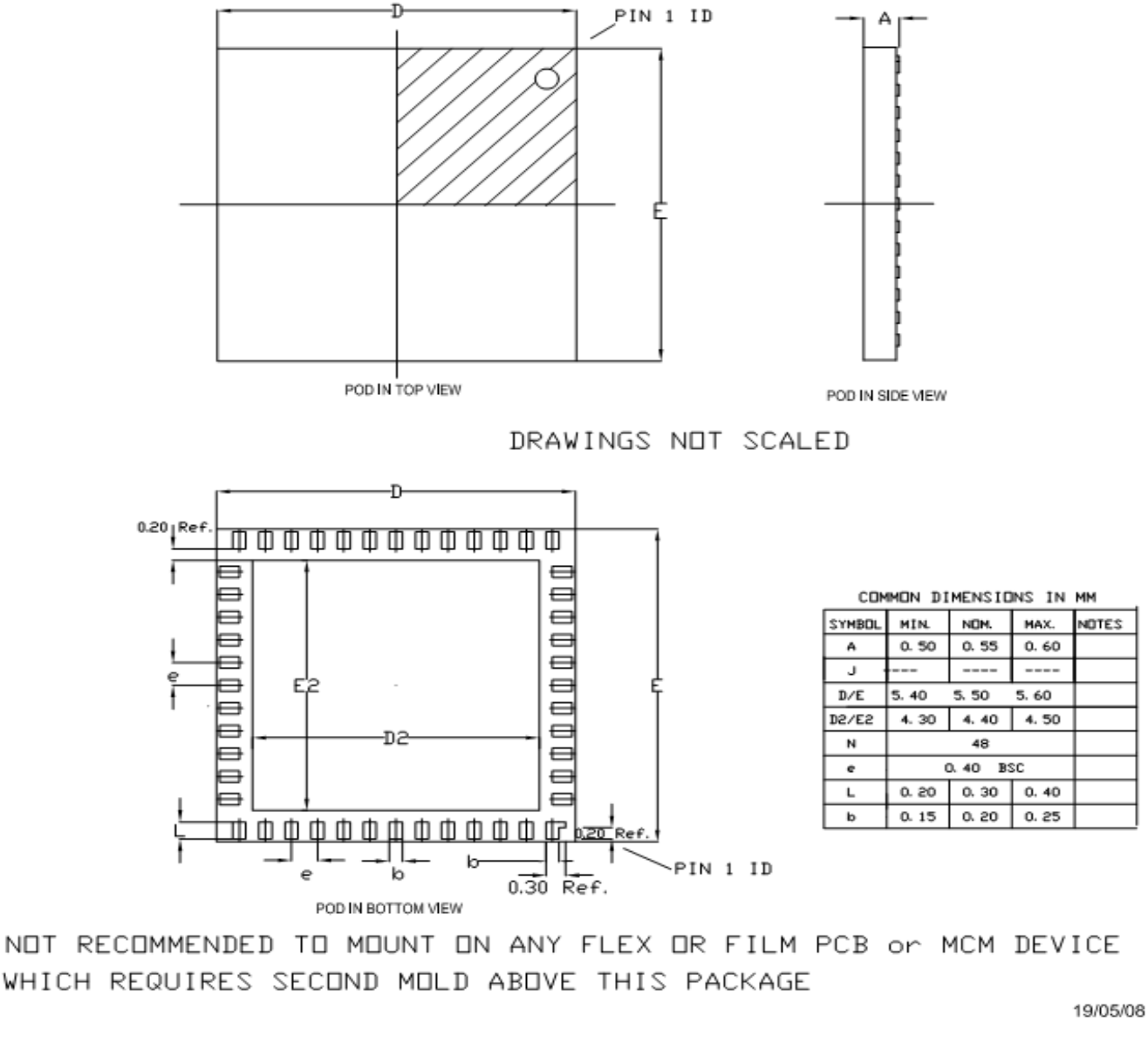
35.5 I/O Pin Characteristics

Table 35-6. Normal I/O Pin Characteristics[1]

Symbol	Parameter	Condition	Min	Typ	Max	Units
R_{PULLUP}	Pull-up resistance		75	100	145	kOhm
V_{IL}	Input low-level voltage	V_{VDD} = 3.0V	-0.3		0.3 * V_{VDD}	V
		V_{VDD} = 1.62V	-0.3		0.3 * V_{VDD}	
V_{IH}	Input high-level voltage	V_{VDD} = 3.6V	0.7 * V_{VDD}		V_{VDD} + 0.3	V
		V_{VDD} = 1.98V	0.7 * V_{VDD}		V_{VDD} + 0.3	
V_{OL}	Output low-level voltage	V_{VDD} = 3.0V, I_{OL} = 3mA			0.4	V
		V_{VDD} = 1.62V, I_{OL} = 2mA			0.4	
V_{OH}	Output high-level voltage	V_{VDD} = 3.0V, I_{OH} = 3mA	V_{VDD} - 0.4			V
		V_{VDD} = 1.62V, I_{OH} = 2mA	V_{VDD} - 0.4			
f_{MAX}	Output frequency[2]	V_{VDD} = 3.0V, load = 10pF			45	MHz
		V_{VDD} = 3.0V, load = 30pF			23	
t_{RISE}	Rise time[2]	V_{VDD} = 3.0V, load = 10pF			4.7	ns
		V_{VDD} = 3.0V, load = 30pF			11.5	
t_{FALL}	Fall time[2]	V_{VDD} = 3.0V, load = 10pF			4.8	
		V_{VDD} = 3.0V, load = 30pF			12	
I_{LEAK}	Input leakage current	Pull-up resistors disabled			1	µA
C_{IN}	Input capacitance, all normal I/O pins except PA05, PA07, PA17, PA20, PA21, PB04, PB05	TQFP48 package		1.4		pF
		QFN48 package		1.1		
		TLLGA48 package		1.1		
		TQFP64 package		1.5		
		QFN64 package		1.1		
C_{IN}	Input capacitance, PA20	TQFP48 package		2.7		
		QFN48 package		2.4		
		TLLGA48 package		2.4		
		TQFP64 package		2.8		
		QFN64 package		2.4		
C_{IN}	Input capacitance, PA05, PA07, PA17, PA21, PB04, PB05	TQFP48 package		3.8		
		QFN48 package		3.5		
		TLLGA48 package		3.5		
		TQFP64 package		3.9		
		QFN64 package		3.5		

Notes: 1. V_{VDD} corresponds to either V_{VDDIN} or V_{VDDIO}, depending on the supply for the pin. Refer to Section on page 10 for details.
2. These values are based on simulation and characterization of other AVR microcontrollers manufactured in the same process technology. These values are not covered by test limits in production.

Figure 36-3. TLLGA-48 Package Drawing



COMMON DIMENSIONS IN MM

SYMBOL	MIN.	NOM.	MAX.	NOTES
A	0.50	0.55	0.60	
J	----	----	----	
D/E	5.40	5.50	5.60	
D2/E2	4.30	4.40	4.50	
N	48			
e	0.40 BSC			
L	0.20	0.30	0.40	
b	0.15	0.20	0.25	

Table 36-8. Device and Package Maximum Weight

39.3	mg

Table 36-9. Package Characteristics

Moisture Sensitivity Level	MSL3

Table 36-10. Package Reference

JEDEC Drawing Reference	N/A
JESD97 Classification	E4