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	66MHz
Connectivity	CANbus, Ethernet, I²C, IrDA, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, I²S, POR, PWM, WDT
Number of I/O	45
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	16K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	A/D 11x12b; D/A 2x12b
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/microchip-technology/at32uc3c264c-z2ut

2. Overview
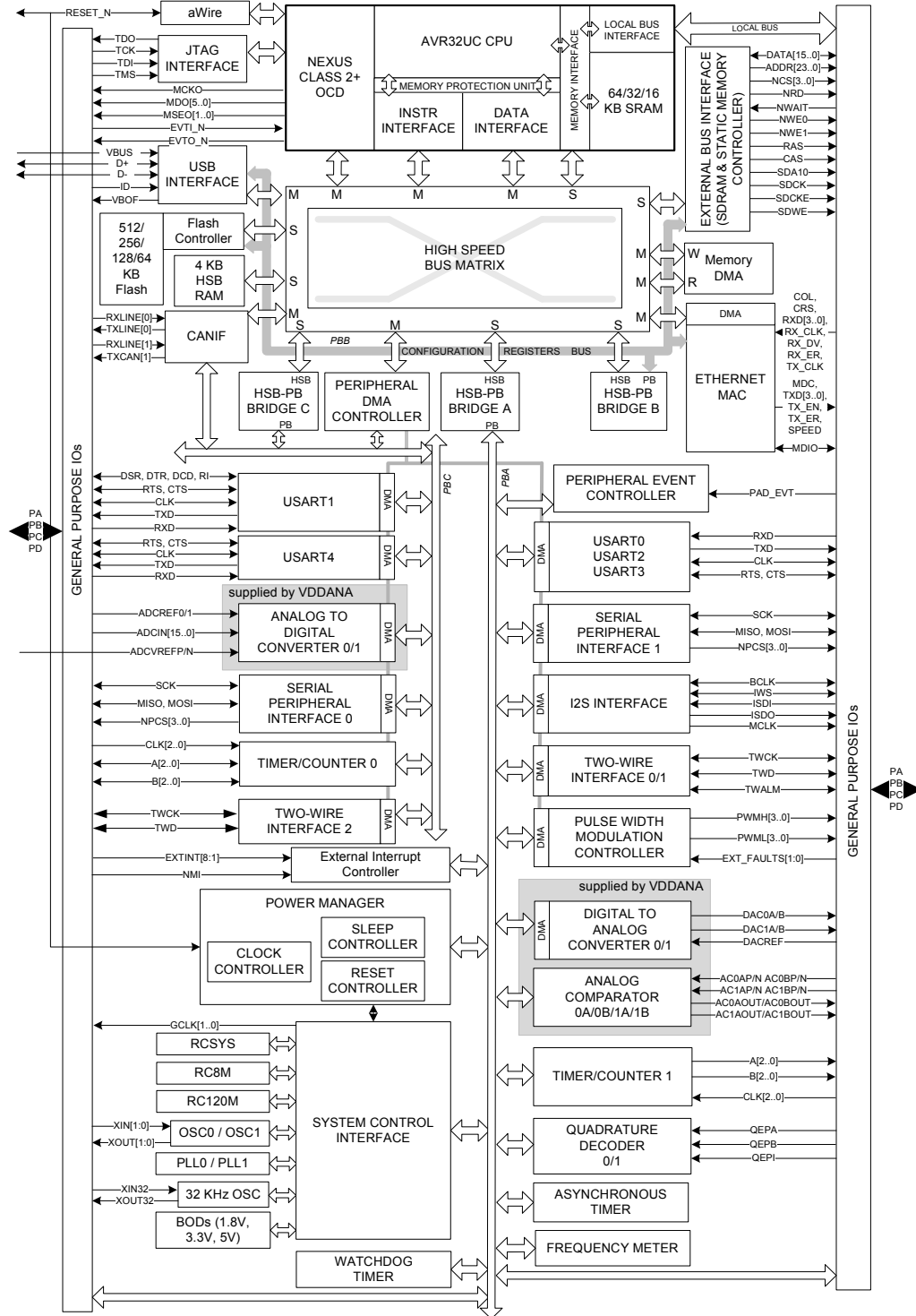
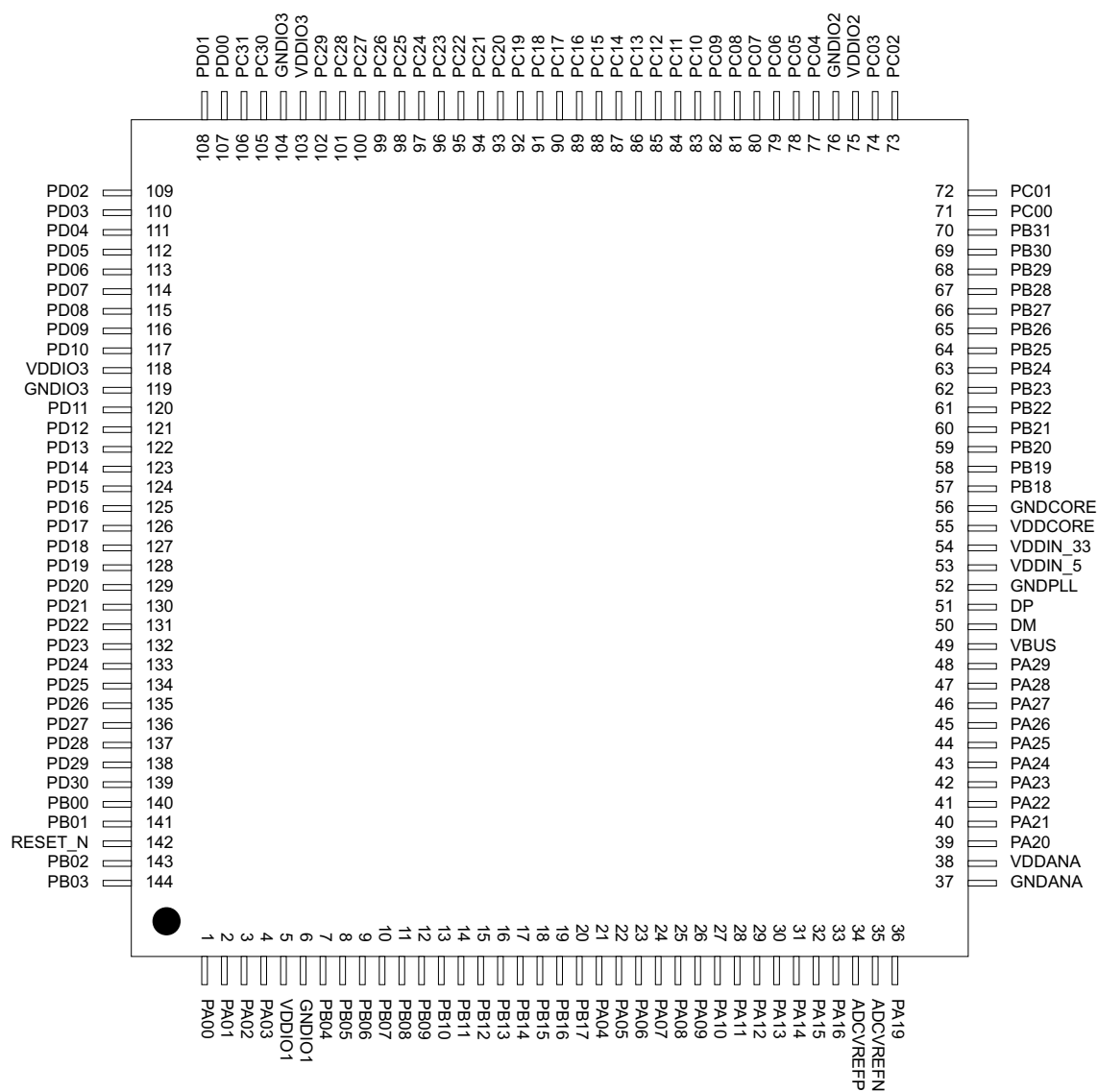## 2.1 Block diagram

Figure 2-1. Block diagram

Figure 3-3. LQFP144 Pinout

Table 3-1. GPIO Controller Function Multiplexing

TQFP / QFN 64	TQFP 100	LQFP 144	PIN	GPIO	Supply	Pin Type [1]	GPIO function					
							A	B	C	D	E	F
		124	PD15	111	VDDIO3	x1/x2	TC0 - A0	USART3 - TXD	EBI - ADDR[11]			
		125	PD16	112	VDDIO3	x1/x2	TC0 - B0	USART3 - RXD	EBI - ADDR[12]			
		126	PD17	113	VDDIO3	x1/x2	TC0 - A1	USART3 - CTS	EBI - ADDR[13]	USART3 - CLK		
		127	PD18	114	VDDIO3	x1/x2	TC0 - B1	USART3 - RTS	EBI - ADDR[14]			
		128	PD19	115	VDDIO3	x1/x2	TC0 - A2		EBI - ADDR[15]			
		129	PD20	116	VDDIO3	x1/x2	TC0 - B2		EBI - ADDR[16]			
57	88	130	PD21	117	VDDIO3	x1/x2	USART3 - TXD	EIC - EXTINT[0]	EBI - ADDR[17]	QDEC1 - QEPI		
	89	131	PD22	118	VDDIO1	x1/x2	USART3 - RXD	TC0 - A2	EBI - ADDR[18]	SCIF - GCLK[0]		
	90	132	PD23	119	VDDIO1	x1/x2	USART3 - CTS	USART3 - CLK	EBI - ADDR[19]	QDEC1 - QEPA		
	91	133	PD24	120	VDDIO1	x1/x2	USART3 - RTS	EIC - EXTINT[8]	EBI - NWE1	QDEC1 - QEPB		
		134	PD25	121	VDDIO1	x1/x2	TC0 - CLK0	USBC - ID	EBI - NWE0		USART4 - CLK	
		135	PD26	122	VDDIO1	x1/x2	TC0 - CLK1	USBC - VBOF	EBI - NRD			
58	92	136	PD27	123	VDDIO1	x1/x2	USART0 - TXD	CANIF - RXLINE[0]	EBI - NCS[1]	TC0 - A0	MACB - RX_ER	
59	93	137	PD28	124	VDDIO1	x1/x2	USART0 - RXD	CANIF - TXLINE[0]	EBI - NCS[2]	TC0 - B0	MACB - RX_DV	
60	94	138	PD29	125	VDDIO1	x1/x2	USART0 - CTS	EIC - EXTINT[6]	USART0 - CLK	TC0 - CLK0	MACB - TX_CLK	
61	95	139	PD30	126	VDDIO1	x1/x2	USART0 - RTS	EIC - EXTINT[3]	EBI - NWAIT	TC0 - A1	MACB - TX_EN	

Note: 1. Refer to "Electrical Characteristics" on page 50 for a description of the electrical properties of the pin types used.
See Section 3.3 for a description of the various peripheral signals.

3.2.2 Peripheral Functions

Each GPIO line can be assigned to one of several peripheral functions. The following table describes how the various peripheral functions are selected. The last listed function has priority in case multiple functions are enabled on the same pin.

Table 3-2. Peripheral Functions

Function	Description
GPIO Controller Function multiplexing	GPIO and GPIO peripheral selection A to F
Nexus OCD AUX port connections	OCD trace system

Table 3-7. Signal Description List

Signal Name	Function	Type	Active Level	Comments
SDCK	SDRAM Clock	Output		
SDCKE	SDRAM Clock Enable	Output		
SDWE	SDRAM Write Enable	Output	Low	
External Interrupt Controller - EIC				
EXTINT[8:1]	External Interrupt Pins	Input		
NMI_N = EXTINT[0]	Non-Maskable Interrupt Pin	Input	Low	
General Purpose Input/Output - GPIOA, GPIOB, GPIOC, GPIOD				
PA[29:19] - PA[16:0]	Parallel I/O Controller GPIOA	I/O		
PB[31:0]	Parallel I/O Controller GPIOB	I/O		
PC[31:0]	Parallel I/O Controller GPIOC	I/O		
PD[30:0]	Parallel I/O Controller GPIOD	I/O		
Inter-IC Sound (I2S) Controller - IISC				
IMCK	I2S Master Clock	Output		
ISCK	I2S Serial Clock	I/O		
ISDI	I2S Serial Data In	Input		
ISDO	I2S Serial Data Out	Output		
IWS	I2S Word Select	I/O		
JTAG				
TCK	Test Clock	Input		
TDI	Test Data In	Input		
TDO	Test Data Out	Output		
TMS	Test Mode Select	Input		
Ethernet MAC - MACB				
COL	Collision Detect	Input		
CRS	Carrier Sense and Data Valid	Input		
MDC	Management Data Clock	Output		
MDIO	Management Data Input/Output	I/O		
RXD[3:0]	Receive Data	Input		

5. Memories

5.1 Embedded Memories

- **Internal High-Speed Flash (See Table 5-1 on page 40)**
  - **– 512 Kbytes**
  - **– 256 Kbytes**
  - **– 128 Kbytes**
  - **– 64 Kbytes**
    - **0 Wait State Access at up to 33 MHz in Worst Case Conditions**
    - **1 Wait State Access at up to 66 MHz in Worst Case Conditions**
    - **Pipelined Flash Architecture, allowing burst reads from sequential Flash locations, hiding penalty of 1 wait state access**
    - **Pipelined Flash Architecture typically reduces the cycle penalty of 1 wait state operation to only 15% compared to 0 wait state operation**
    - **100 000 Write Cycles, 15-year Data Retention Capability**
    - **Sector Lock Capabilities, Bootloader Protection, Security Bit**
    - **32 Fuses, Erased During Chip Erase**
    - **User Page For Data To Be Preserved During Chip Erase**
- **Internal High-Speed SRAM, Single-cycle access at full speed (See Table 5-1 on page 40)**
  - **– 64 Kbytes**
  - **– 32 Kbytes**
  - **– 16 Kbytes**
- **Supplementary Internal High-Speed System SRAM (HSB RAM), Single-cycle access at full speed**
  - **– Memory space available on System Bus for peripherals data.**
  - **– 4 Kbytes**

Table 7-36. ADC and S/H Transfer Characteristics (Continued)10-bit Resolution Mode and S/H gain from 1 to 16[1]

Symbol	Parameter	Conditions	Min	Typ	Max	Units
RES	Resolution	Differential mode, V_{VDDANA} = 5V, $V_{ADCREF0}$ = 3V, ADCFIA.SEQCFGn.SRES = 1, S/H gain from 1 to 16 (F_{adc} = 1.5MHz)			10	Bit
INL	Integral Non-Linearity				1.5	LSB
DNL	Differential Non-Linearity				1.5	LSB
	Offset error		-25		25	mV
	Gain error		-15		15	mV

Note: 1. The measures are done without any I/O activity on VDDANA/GNDANA power domain.

7.8.7 Digital to Analog Converter (DAC) Characteristics

Table 7-37. Channel Conversion Time and DAC Clock

Symbol	Parameter	Conditions	Min	Typ	Max	Units
f_{DAC}	DAC clock frequency				1	MHz
$t_{STARTUP}$	Startup time				3	µs
t_{CONV}	Conversion time (latency)	No S/H enabled, internal DAC			1	µs
		One S/H			1.5	µs
		Two S/H			2	µs
	Throughput rate				$1/t_{CONV}$	MSPS

Table 7-38. External Voltage Reference Input

Symbol	Parameter	Conditions	Min	Typ	Max	Units
V_{DACREF}	DACREF input voltage range		1.2		V_{VDDANA}-0.7	V

Table 7-39. DAC Outputs

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	Output range	with external DAC reference	0.2		V_{DACREF}	V
		with internal DAC reference	0.2		V_{VDDANA}-0.7	
C_{LOAD}	Output capacitance		0		100	pF
R_{LOAD}	Output resitance		2			kΩ

Maximum SPI Frequency, Master Output

The maximum SPI master output frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(f_{PINMAX}, \frac{1}{SPIn}, \frac{f_{CLKSPI} \times 2}{9})$$

Where $SPIn$ is the MOSI delay, USPI2 or USPI5 depending on CPOL and NCPHA. f_{PINMAX} is the maximum frequency of the SPI pins. Please refer to the I/O Pin Characteristics section for the maximum frequency of the pins. f_{CLKSPI} is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock.

Maximum SPI Frequency, Master Input

The maximum SPI master input frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(\frac{1}{SPIn + t_{VALID}}, \frac{f_{CLKSPI} \times 2}{9})$$

Where $SPIn$ is the MISO setup and hold time, USPI0 + USPI1 or USPI3 + USPI4 depending on CPOL and NCPHA. T_{VALID} is the SPI slave response time. Please refer to the SPI slave datasheet for T_{VALID}. f_{CLKSPI} is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock.

7.9.3.2   Slave mode

Figure 7-8. USART in SPI Slave Mode With (CPOL= 0 and CPHA= 1) or (CPOL= 1 and CPHA= 0)
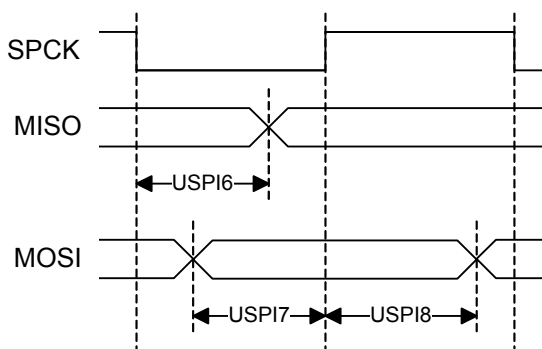
Figure 7-9. USART in SPI Slave Mode With (CPOL= CPHA= 0) or (CPOL= CPHA= 1)
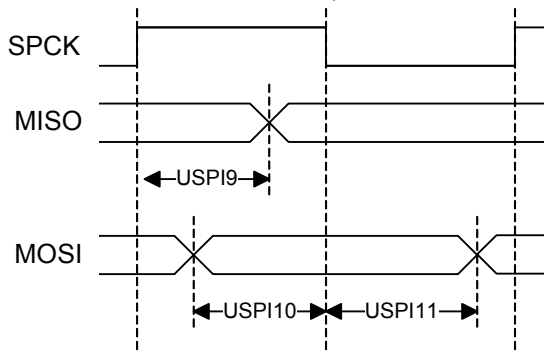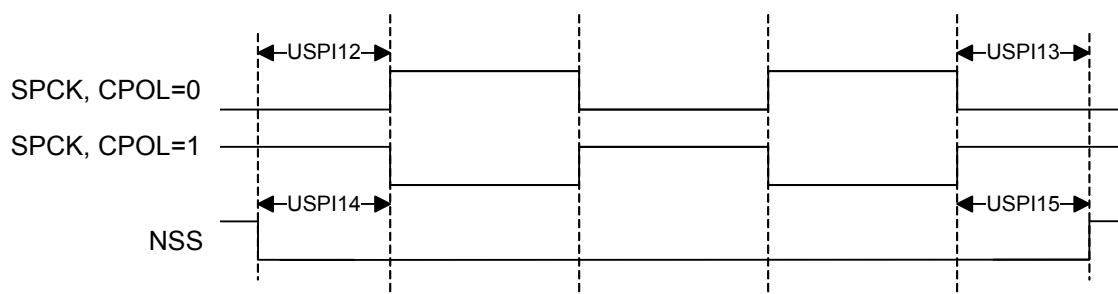

Figure 7-10. USART in SPI Slave Mode NPCS Timing

Table 7-47. USART in SPI mode Timing, Slave Mode[1]

Symbol	Parameter	Conditions	Min	Max	Units
USPI6	SPCK falling to MISO delay			27	ns
USPI7	MOSI setup time before SPCK rises		t_{SAMPLE}[2] + t_{CLK_USART}		ns
USPI8	MOSI hold time after SPCK rises		0		ns
USPI9	SPCK rising to MISO delay			28	ns
USPI10	MOSI setup time before SPCK falls	external capacitor = 40pF	t_{SAMPLE}[2] + t_{CLK_USART}		ns
USPI11	MOSI hold time after SPCK falls		0		ns
USPI12	NSS setup time before SPCK rises		33		ns
USPI13	NSS hold time after SPCK falls		0		ns
USPI14	NSS setup time before SPCK falls		33		ns
USPI15	NSS hold time after SPCK rises		0		ns

Note: 1. 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.

2. Where: $t_{SAMPLE} = t_{SPCK} - \left(\left\lfloor \dfrac{t_{SPCK}}{2 \times t_{CLKUSART}} \right\rfloor + \dfrac{1}{2} \right) \times t_{CLKUSART}$

Maximum SPI Frequency, Slave Input Mode

The maximum SPI slave input frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(\frac{f_{CLKSPI} \times 2}{9}, \frac{1}{SPIn})$$

Where $SPIn$ is the MOSI setup and hold time, USPI7 + USPI8 or USPI10 + USPI11 depending on CPOL and NCPHA. f_{CLKSPI} is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock.

Maximum SPI Frequency, Slave Output Mode

The maximum SPI slave output frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(\frac{f_{CLKSPI} \times 2}{9}, f_{PINMAX}, \frac{1}{SPIn + t_{SETUP}})$$

Where $SPIn$ is the MISO delay, USPI6 or USPI9 depending on CPOL and NCPHA. T_{SETUP} is the SPI master setup time. Please refer to the SPI masterdatasheet for $T_{SETUP} \cdot f_{CLKSPI}$ is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock. f_{PINMAX} is the maximum frequency of the SPI pins. Please refer to the I/O Pin Characteristics section for the maximum frequency of the pins.

7.9.4   SPI Timing

7.9.4.1   Master mode

Figure 7-11.   SPI Master Mode With (CPOL= NCPHA= 0) or (CPOL= NCPHA= 1)
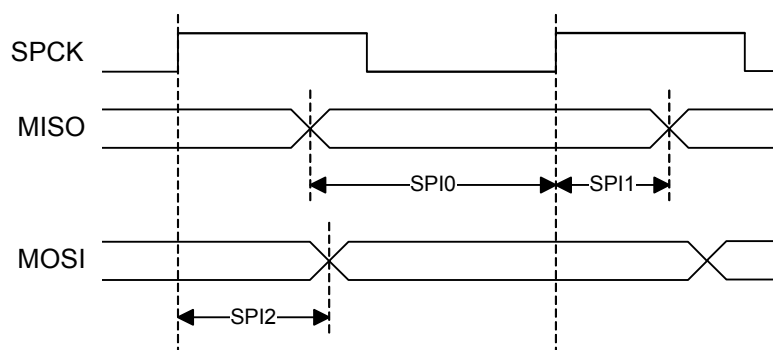
7.9.4.2 Slave mode

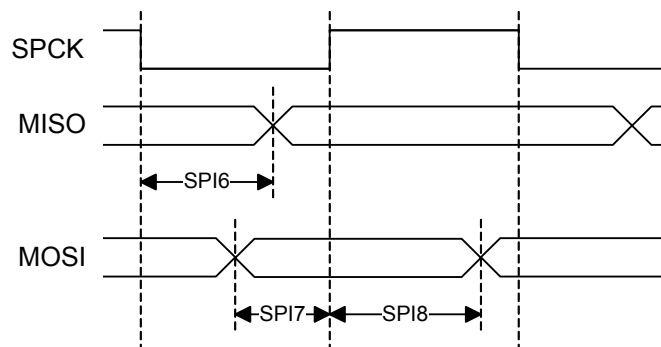**Figure 7-13.** SPI Slave Mode With (CPOL= 0 and NCPHA= 1) or (CPOL= 1 and NCPHA= 0)

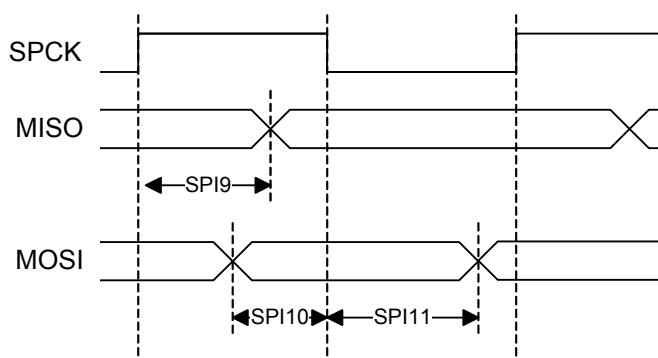**Figure 7-14.** SPI Slave Mode With (CPOL= NCPHA= 0) or (CPOL= NCPHA= 1)
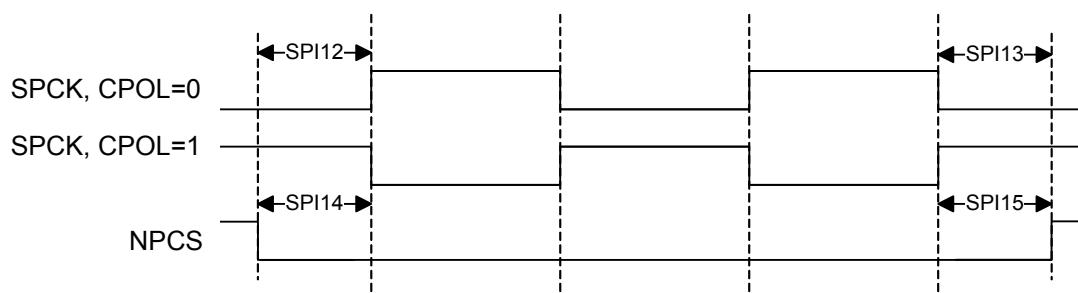

Figure 7-15. SPI Slave Mode NPCS Timing

Table 7-49. SPI Timing, Slave Mode[1]

Symbol	Parameter	Conditions	Min	Max	Units
SPI6	SPCK falling to MISO delay	external capacitor = 40pF		29	ns
SPI7	MOSI setup time before SPCK rises		0		ns
SPI8	MOSI hold time after SPCK rises		6.5		ns
SPI9	SPCK rising to MISO delay			30	ns
SPI10	MOSI setup time before SPCK falls		0		ns
SPI11	MOSI hold time after SPCK falls		5		ns
SPI12	NPCS setup time before SPCK rises		0		ns
SPI13	NPCS hold time after SPCK falls		1.5		ns
SPI14	NPCS setup time before SPCK falls		0		ns
SPI15	NPCS hold time after SPCK rises		1.5		ns

Note: 1. 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.

Maximum SPI Frequency, Slave Input Mode

The maximum SPI slave input frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(f_{CLKSPI}, \frac{1}{SPIn})$$

Where $SPIn$ is the MOSI setup and hold time, SPI7 + SPI8 or SPI10 + SPI11 depending on CPOL and NCPHA. f_{CLKSPI} is the maximum frequency of the CLK_SPI. Refer to the SPI chapter for a description of this clock.

Maximum SPI Frequency, Slave Output Mode

The maximum SPI slave output frequency is given by the following formula:

$$f_{SPCKMAX} = MIN(f_{PINMAX}, \frac{1}{SPIn + t_{SETUP}})$$

Where $SPIn$ is the MISO delay, SPI6 or SPI9 depending on CPOL and NCPHA. t_{SETUP} is the SPI master setup time. Please refer to the SPI masterdatasheet for t_{SETUP}. f_{PINMAX} is the maximum frequency of the SPI pins. Please refer to the I/O Pin Characteristics section for the maximum frequency of the pins.

7.9.5 TWIM/TWIS Timing

Figure 7-50 shows the TWI-bus timing requirements and the compliance of the device with them. Some of these requirements (t_r and t_f) are met by the device without requiring user intervention. Compliance with the other requirements (t_{HD-STA}, t_{SU-STA}, t_{SU-STO}, t_{HD-DAT}, $t_{SU-DAT-I2C}$, $t_{LOW-I2C}$, t_{HIGH}, and f_{TWCK}) requires user intervention through appropriate programming of the relevant

7.9.7 EBI Timings

See EBI I/O lines description for more details.

Table 7-52. SMC Clock Signal.

Symbol	Parameter	Max[1]	Units
$1/(t_{CPSMC})$	SMC Controller clock frequency	f_{cpu}	MHz

Note: 1. The maximum frequency of the SMC interface is the same as the max frequency for the HSB.

Table 7-53. SMC Read Signals with Hold Settings[1]

Symbol	Parameter	Conditions	Min	Units
		NRD Controlled (READ_MODE = 1)		
SMC_1	Data setup before NRD high		32.5	
SMC_2	Data hold after NRD high		0	
SMC_3	NRD high to NBS0/A0 change[2]		nrd hold length * t_{CPSMC} - 1.5	
SMC_4	NRD high to NBS1 change[2]	V_{VDD} = 3.0V, drive strength of the pads set to the lowest, external capacitor = 40pF	nrd hold length * t_{CPSMC} - 0	
SMC_5	NRD high to NBS2/A1 change[2]		nrd hold length * t_{CPSMC} - 0	ns
SMC_7	NRD high to A2 - A25 change[2]		nrd hold length * t_{CPSMC} - 5.6	
SMC_8	NRD high to NCS inactive[2]		(nrd hold length - ncs rd hold length) * t_{CPSMC} - 1.3	
SMC_9	NRD pulse width		nrd pulse length * t_{CPSMC} - 0.6	
		NRD Controlled (READ_MODE = 0)		
SMC_{10}	Data setup before NCS high		34.1	
SMC_{11}	Data hold after NCS high		0	
SMC_{12}	NCS high to NBS0/A0 change[2]		ncs rd hold length * t_{CPSMC} - 3	
SMC_{13}	NCS high to NBS0/A0 change[2]	V_{VDD} = 3.0V, drive strength of the pads set to the lowest, external capacitor = 40pF	ncs rd hold length * t_{CPSMC} - 2	
SMC_{14}	NCS high to NBS2/A1 change[2]		ncs rd hold length * t_{CPSMC} - 1.1	ns
SMC_{16}	NCS high to A2 - A25 change[2]		ncs rd hold length * t_{CPSMC} - 7.2	
SMC_{17}	NCS high to NRD inactive[2]		(ncs rd hold length - nrd hold length) * t_{CPSMC} - 2.2	
SMC_{18}	NCS pulse width		ncs rd pulse length * t_{CPSMC} - 3	

Note: 1. 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.

2. hold length = total cycle duration - setup duration - pulse duration. "hold length" is for "ncs rd hold length" or "nrd hold length".

Table 7-56. SMC Write Signals with No Hold Settings (NWE Controlled only)[1]

Symbol	Parameter	Conditions	Min	Units
SMC$_{37}$	NWE rising to A2-A25 valid	V_{VDD} = 3.0V, drive strength of the pads set to the lowest, external capacitor = 40pF	8.7	ns
SMC$_{38}$	NWE rising to NBS0/A0 valid		7.6	
SMC$_{40}$	NWE rising to A1/NBS2 change		8.7	
SMC$_{42}$	NWE rising to NCS rising		8.4	
SMC$_{43}$	Data Out valid before NWE rising		(nwe pulse length - 1) * t$_{CPSMC}$ - 1.2	
SMC$_{44}$	Data Out valid after NWE rising		8.4	
SMC$_{45}$	NWE pulse width		nwe pulse length * t$_{CPSMC}$ - 0	

Note: 1. 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 7-17. SMC Signals for NCS Controlled Accesses
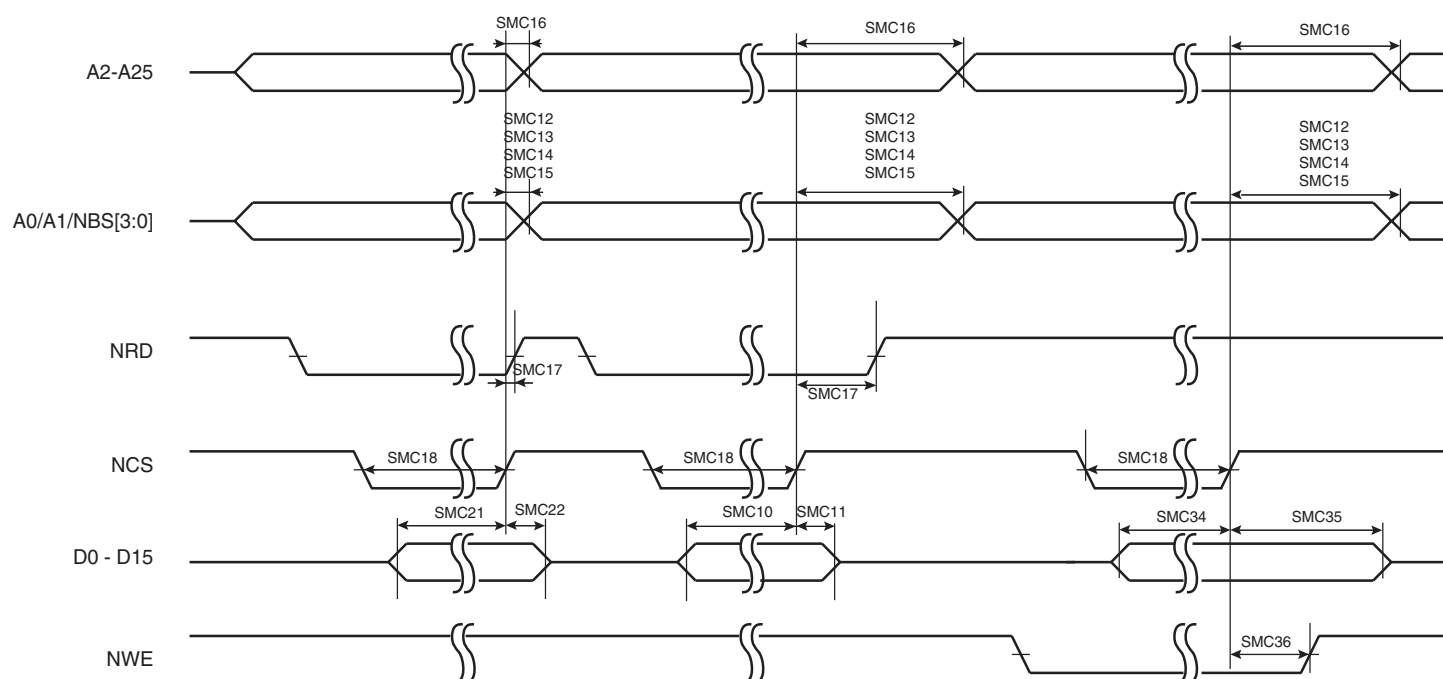
Table 7-58. SDRAM Signal[1]

Symbol	Parameter	Conditions	Min	Units
SDRAMC$_1$	SDCKE high before SDCK rising edge		5.6	
SDRAMC$_2$	SDCKE low after SDCK rising edge		7.3	
SDRAMC$_3$	SDCKE low before SDCK rising edge		6.8	
SDRAMC$_4$	SDCKE high after SDCK rising edge		8.3	
SDRAMC$_5$	SDCS low before SDCK rising edge		6.1	
SDRAMC$_6$	SDCS high after SDCK rising edge		8.4	
SDRAMC$_7$	RAS low before SDCK rising edge		7	
SDRAMC$_8$	RAS high after SDCK rising edge		7.7	
SDRAMC$_9$	SDA10 change before SDCK rising edge		6.4	
SDRAMC$_{10}$	SDA10 change after SDCK rising edge		7.1	
SDRAMC$_{11}$	Address change before SDCK rising edge	V_{VDD} = 3.0V, drive strength of the pads set to the highest, external capacitor = 40 pF on SDRAM pins except 8 pF on SDCK pins	4.7	ns
SDRAMC$_{12}$	Address change after SDCK rising edge		4.4	
SDRAMC$_{13}$	Bank change before SDCK rising edge		6.2	
SDRAMC$_{14}$	Bank change after SDCK rising edge		6.9	
SDRAMC$_{15}$	CAS low before SDCK rising edge		6.6	
SDRAMC$_{16}$	CAS high after SDCK rising edge		7.8	
SDRAMC$_{17}$	DQM change before SDCK rising edge		6	
SDRAMC$_{18}$	DQM change after SDCK rising edge		6.7	
SDRAMC$_{19}$	D0-D15 in setup before SDCK rising edge		6.4	
SDRAMC$_{20}$	D0-D15 in hold after SDCK rising edge		0	
SDRAMC$_{23}$	SDWE low before SDCK rising edge		7	
SDRAMC$_{24}$	SDWE high after SDCK rising edge		7.4	
SDRAMC$_{25}$	D0-D15 Out valid before SDCK rising edge		5.2	
SDRAMC$_{26}$	D0-D15 Out valid after SDCK rising edge		5.6	

Note: 1. 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 8-2. TQFP-64 package drawing
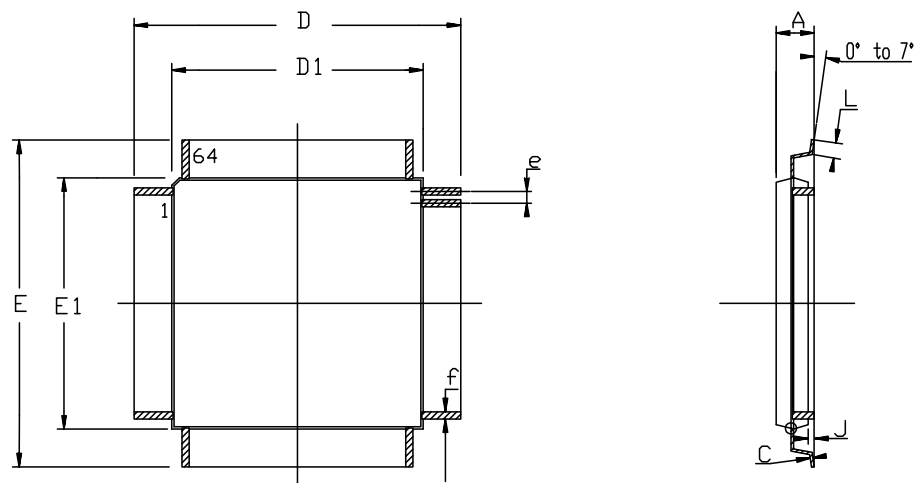
COMMON DIMENSIONS IN MM

SYMBOL	Min	Max	NOTES
A	----	1.20	
A1	0.95	1.05	
C	0.09	0.20	
D	12.00 BSC		
D1	10.00 BSC		
E	12.00 BSC		
E1	10.00 BSC		
J	0.05	0.15	
L	0.45	0.75	
e	0.50 BSC		
f	0.17	0.27	
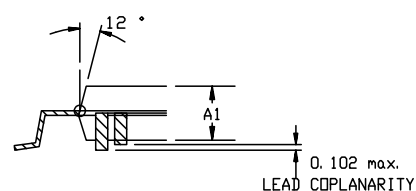
Table 8-5. Device and Package Maximum Weight

300	mg

Table 8-6. Package Characteristics

Moisture Sensitivity Level	Jdec J-STD0-20D - MSL 3

Table 8-7. Package Reference

JEDEC Drawing Reference	MS-026
JESD97 Classification	E3

Figure 8-3. TQFP-100 package drawing
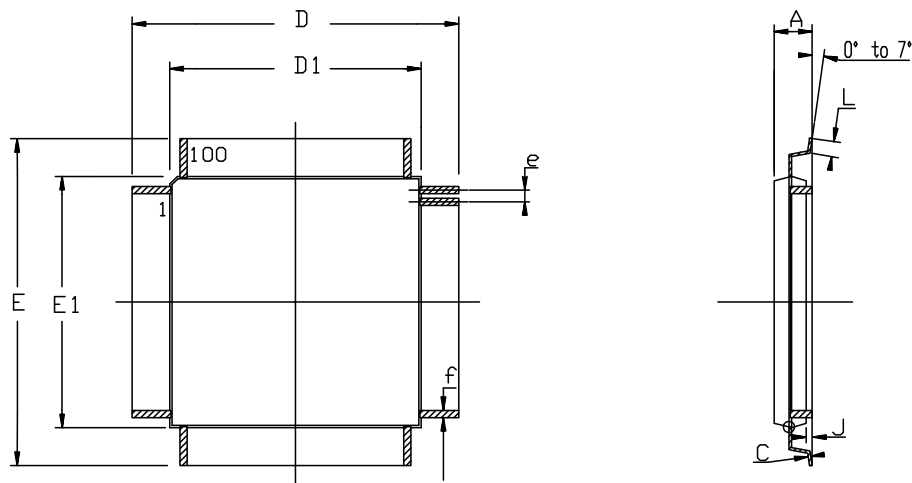
COMMON DIMENSIONS IN MM

SYMBOL	Min	Max	NOTES
A	----	1.20	
A1	0.95	1.05	
C	0.09	0.20	
D	16.00 BSC		
D1	14.00 BSC		
E	16.00 BSC		
E1	14.00 BSC		
J	0.05	0.15	
L	0.45	0.75	
e	0.50 BSC		
f	0.17	0.27	
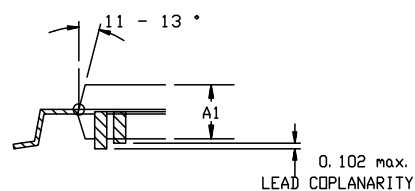
Table 8-8. Device and Package Maximum Weight

500	mg

Table 8-9. Package Characteristics

Moisture Sensitivity Level	Jdec J-STD0-20D - MSL 3

Table 8-10. Package Reference

JEDEC Drawing Reference	MS-026
JESD97 Classification	E3

4   SPI bad serial clock generation on 2nd chip_select when SCBR=1, CPOL=1, and NCPHA=0

When multiple chip selects (CS) are in use, if one of the baudrates equal 1 while one (CSRn.SCBR=1) of the others do not equal 1, and CSRn.CPOL=1 and CSRn.NCPHA=0, then an additional pulse will be generated on SCK.

Fix/Workaround

When multiple CS are in use, if one of the baudrates equals 1, the others must also equal 1 if CSRn.CPOL=1 and CSRn.NCPHA=0.

10.1.7   TC

1   Channel chaining skips first pulse for upper channel

When chaining two channels using the Block Mode Register, the first pulse of the clock between the channels is skipped.

Fix/Workaround

Configure the lower channel with RA = 0x1 and RC = 0x2 to produce a dummy clock cycle for the upper channel. After the dummy cycle has been generated, indicated by the SR.CPCS bit, reconfigure the RA and RC registers for the lower channel with the real values.

10.1.8   TWIM

1   SMBALERT bit may be set after reset

For TWIM0 and TWIM1 modules, the SMBus Alert (SMBALERT) bit in the Status Register (SR) might be erroneously set after system reset.

Fix/Workaround

After system reset, clear the SR.SMBALERT bit before commencing any TWI transfer.

For TWIM2 module, the SMBus Alert (SMBALERT) is not implemented but the bit in the Status Register (SR) is erroneously set once TWIM2 is enabled.

Fix/Workaround

None.

10.1.9   TWIS

1   Clearing the NAK bit before the BTF bit is set locks up the TWI bus

When the TWIS is in transmit mode, clearing the NAK Received (NAK) bit of the Status Register (SR) before the end of the Acknowledge/Not Acknowledge cycle will cause the TWIS to attempt to continue transmitting data, thus locking up the bus.

Fix/Workaround

Clear SR.NAK only after the Byte Transfer Finished (BTF) bit of the same register has been set.

10.1.10   USBC

1   UPINRQx.INRQ field is limited to 8-bits

In Host mode, when using the UPINRQx.INRQ feature together with the multi-packet mode to launch a finite number of packet among multi-packet, the multi-packet size (located in the descriptor table) is limited to the UPINRQx.INRQ value multiply by the pipe size.

Fix/Workaround

UPINRQx.INRQ value shall be less than the number of configured multi-packet.

2   In USB host mode, downstream resume feature does not work (UHCON.RESUME=1).

10.2 rev D

10.2.1 ADCIFA

1   ADCREFP/ADCREFN can not be selected as an external ADC reference by setting the ADCIFA.CFG.EXREF bit to one
Fix/Workaround
A voltage reference can be applied on ADCREFP/ADCREFN pins if the ADCIFA.CFG.EXREF bit is set to zero, the ADCIFA.CFG.RS bit is set to zero and the voltage reference applied on ADCREFP/ADCREFN pins is higher than the internal 1V reference.

10.2.2 AST

1   AST wake signal is released one AST clock cycle after the BUSY bit is cleared
After writing to the Status Clear Register (SCR) the wake signal is released one AST clock cycle after the BUSY bit in the Status Register (SR.BUSY) is cleared. If entering sleep mode directly after the BUSY bit is cleared the part will wake up immediately.
Fix/Workaround
Read the Wake Enable Register (WER) and write this value back to the same register. Wait for BUSY to clear before entering sleep mode.

10.2.3 aWire

1   aWire MEMORY_SPEED_REQUEST command does not return correct CV
The aWire MEMORY_SPEED_REQUEST command does not return a CV corresponding to the formula in the aWire Debug Interface chapter.
Fix/Workaround
Issue a dummy read to address 0x100000000 before issuing the MEMORY_SPEED_REQUEST command and use this formula instead:

$$f_{sab} = \frac{7f_{aw}}{CV - 3}$$

10.2.4 GPIO

1   Clearing Interrupt flags can mask other interrupts
When clearing interrupt flags in a GPIO port, interrupts on other pins of that port, happening in the same clock cycle will not be registered.
Fix/Workaround
Read the PVR register of the port before and after clearing the interrupt to see if any pin change has happened while clearing the interrupt. If any change occurred in the PVR between the reads, they must be treated as an interrupt.

10.2.5 Power Manager

1   Clock Failure Detector (CFD) can be issued while turning off the CFD
While turning off the CFD, the CFD bit in the Status Register (SR) can be set. This will change the main clock source to RCSYS.
Fix/Workaround
Solution 1: Enable CFD interrupt. If CFD interrupt is issues after turning off the CFD, switch back to original main clock source.
Solution 2: Only turn off the CFD while running the main clock on RCSYS.

2   Disabling SPI has no effect on the SR.TDRE bit
Disabling SPI has no effect on the SR.TDRE bit whereas the write data command is filtered when SPI is disabled. Writing to TDR when SPI is disabled will not clear SR.TDRE. If SPI is disabled during a PDCA transfer, the PDCA will continue to write data to TDR until its buffer is empty, and this data will be lost.
Fix/Workaround
Disable the PDCA, add two NOPs, and disable the SPI. To continue the transfer, enable the SPI and PDCA.

3   SPI disable does not work in SLAVE mode
SPI disable does not work in SLAVE mode.
Fix/Workaround
Read the last received data, then perform a software reset by writing a one to the Software Reset bit in the Control Register (CR.SWRST).

4   SPI bad serial clock generation on 2nd chip_select when SCBR=1, CPOL=1, and NCPHA=0
When multiple chip selects (CS) are in use, if one of the baudrates equal 1 while one (CSRn.SCBR=1) of the others do not equal 1, and CSRn.CPOL=1 and CSRn.NCPHA=0, then an additional pulse will be generated on SCK.
Fix/Workaround
When multiple CS are in use, if one of the baudrates equals 1, the others must also equal 1 if CSRn.CPOL=1 and CSRn.NCPHA=0.

10.2.8   TC

1   Channel chaining skips first pulse for upper channel
When chaining two channels using the Block Mode Register, the first pulse of the clock between the channels is skipped.
Fix/Workaround
Configure the lower channel with RA = 0x1 and RC = 0x2 to produce a dummy clock cycle for the upper channel. After the dummy cycle has been generated, indicated by the SR.CPCS bit, reconfigure the RA and RC registers for the lower channel with the real values.

10.2.9   TWIM

1   SMBALERT bit may be set after reset
For TWIM0 and TWIM1 modules, the SMBus Alert (SMBALERT) bit in the Status Register (SR) might be erroneously set after system reset.
Fix/Workaround
After system reset, clear the SR.SMBALERT bit before commencing any TWI transfer.

For TWIM2 module, the SMBus Alert (SMBALERT) is not implemented but the bit in the Status Register (SR) is erroneously set once TWIM2 is enabled.
Fix/Workaround
None.

2   TWIM TWALM polarity is wrong
The TWALM signal in the TWIM is active high instead of active low.
Fix/Workaround
Use an external inverter to invert the signal going into the TWIM. When using both TWIM and TWIS on the same pins, the TWALM cannot be used.

Headquarters

Atmel Corporation
2325 Orchard Parkway
San Jose, CA 95131
USA
Tel: 1(408) 441-0311
Fax: 1(408) 487-2600

International

Atmel Asia
Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
Tel: (852) 2245-6100
Fax: (852) 2722-1369

Atmel Europe
Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-
Yvelines Cedex
France
Tel: (33) 1-30-60-70-00
Fax: (33) 1-30-60-71-11

Atmel Japan
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
Tel: (81) 3-3523-3551
Fax: (81) 3-3523-7581

Product Contact

Web Site
www.atmel.com

Technical Support
avr32@atmel.com

Sales Contact
www.atmel.com/contacts

Literature Requests
www.atmel.com/literature