



Welcome to [E-XFL.COM](https://www.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	66MHz
Connectivity	EBI/EMI, I ² C, IrDA, Memory Card, SPI, SSC, UART/USART, USB OTG
Peripherals	Brown-out Detect/Reset, DMA, POR, WDT
Number of I/O	110
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	128K x 8
Voltage - Supply (Vcc/Vdd)	1.75V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	144-TFBGA
Supplier Device Package	144-FFBGA (11x11)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at32uc3a3256s-ctut

2. Overview

2.1 Block Diagram

Figure 2-1. Block Diagram

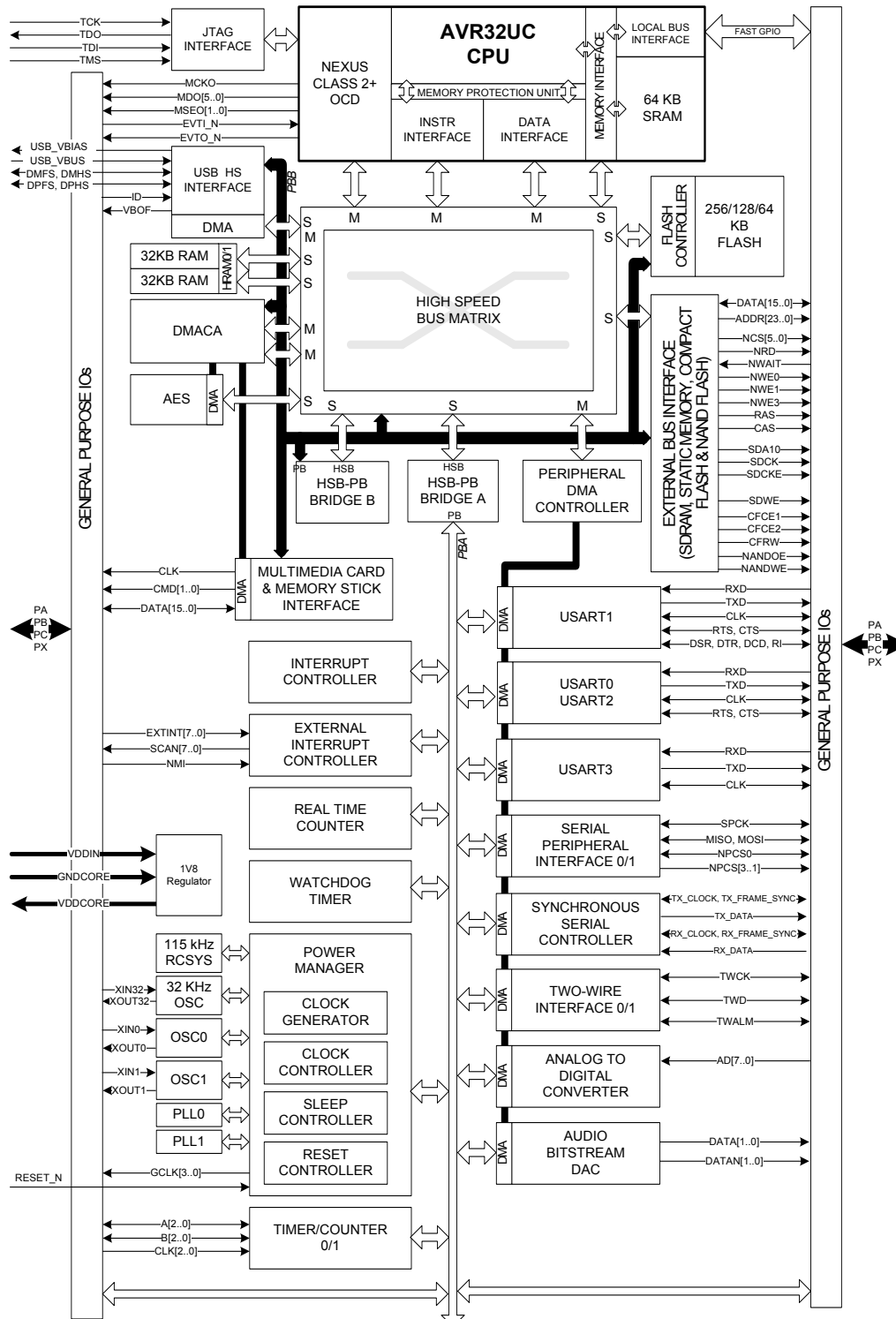


Figure 3-2. LQFP144 Pinout

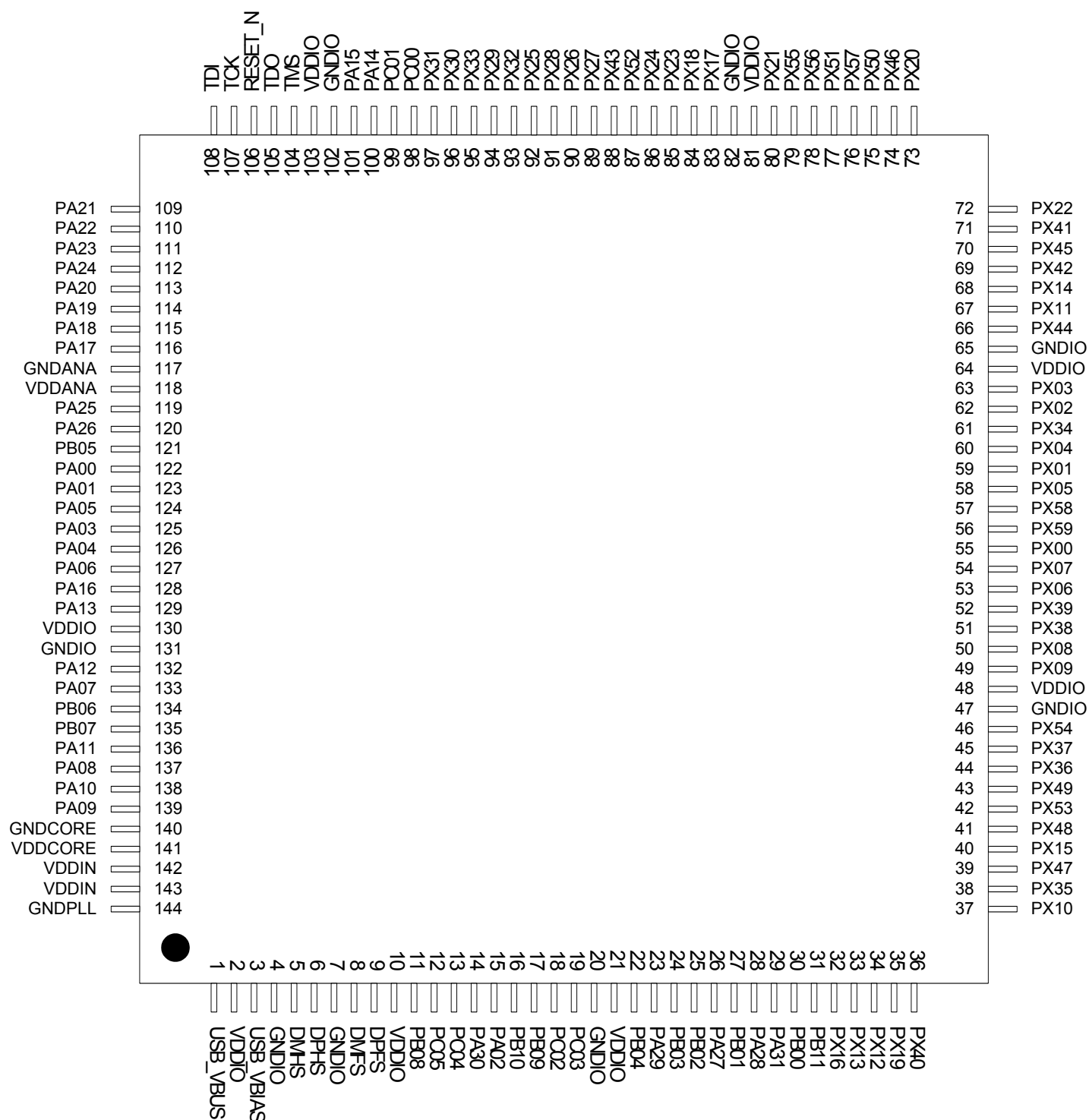


Table 3-1. GPIO Controller Function Multiplexing

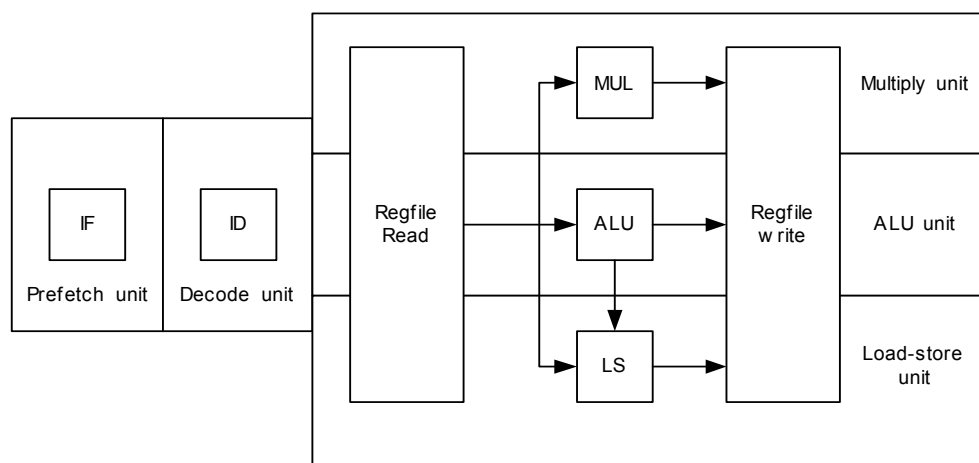
BGA 144	QFP 144	BGA 100	PIN	G P I O	Supply	PIN Type (2)	GPIO function			
							A	B	C	D
C7	14	A4	PA30	30	VDDIO	x1	MCI - DATA[1]	USART3 - CLK	DMACA - DMAACK[0]	MSI - DATA[1]
B3	29	C2	PA31	31	VDDIO	x1	MCI - DATA[2]	USART2 - RXD	DMACA - DMARQ[0]	MSI - DATA[2]
A2	30	B1	PB00	32	VDDIO	x1	MCI - DATA[3]	USART2 - TXD	ADC - TRIGGER	MSI - DATA[3]
C4	27	B2	PB01	33	VDDIO	x1	MCI - DATA[4]	ABDAC - DATA[1]	EIC - SCAN[0]	MSI - INS
B4	25	B3	PB02	34	VDDIO	x1	MCI - DATA[5]	ABDAC - DATAN[1]	EIC - SCAN[1]	
A5	24	C4	PB03	35	VDDIO	x1	MCI - DATA[6]	USART2 - CLK	EIC - SCAN[2]	
B6	22	A3	PB04	36	VDDIO	x1	MCI - DATA[7]	USART3 - RXD	EIC - SCAN[3]	
H12	121	F7(1)	PB05	37	VDDIO	x3	USB - ID	TC0 - A0	EIC - SCAN[4]	
D12	134	D7	PB06	38	VDDIO	x1	USB - VBOF	TC0 - B0	EIC - SCAN[5]	
D11	135	D6	PB07	39	VDDIO	x3	SPI1 - SPCK	SSC - TX_CLOCK	EIC - SCAN[6]	
C8	11	C6	PB08	40	VDDIO	x2	SPI1 - MISO	SSC - TX_DATA	EIC - SCAN[7]	
E7	17	C5	PB09	41	VDDIO	x2	SPI1 - NPCS[0]	SSC - RX_DATA	EBI - NCS[4]	
D7	16	D5	PB10	42	VDDIO	x2	SPI1 - MOSI	SSC - RX_FRAME_SYNC	EBI - NCS[5]	
B2	31	C1	PB11	43	VDDIO	x1	USART1 - RXD	SSC - TX_FRAME_SYNC	PM - GCLK[1]	
K5	98	K5(1)	PC00	45	VDDIO	x1				
H6	99	K6	PC01	46	VDDIO	x1				
A7	18	A5	PC02	47	VDDIO	x1				
B7	19	A6	PC03	48	VDDIO	x1				
A8	13	B7	PC04	49	VDDIO	x1				
A9	12	A7	PC05	50	VDDIO	x1				
G1	55	G4	PX00	51	VDDIO	x2	EBI - DATA[10]	USART0 - RXD	USART1 - RI	
H1	59	G2	PX01	52	VDDIO	x2	EBI - DATA[9]	USART0 - TXD	USART1 - DTR	
J2	62	G3	PX02	53	VDDIO	x2	EBI - DATA[8]	USART0 - CTS	PM - GCLK[0]	
K1	63	J1	PX03	54	VDDIO	x2	EBI - DATA[7]	USART0 - RTS		
J1	60	H1	PX04	55	VDDIO	x2	EBI - DATA[6]	USART1 - RXD		
G2	58	G1	PX05	56	VDDIO	x2	EBI - DATA[5]	USART1 - TXD		
F3	53	F3	PX06	57	VDDIO	x2	EBI - DATA[4]	USART1 - CTS		
F2	54	F4	PX07	58	VDDIO	x2	EBI - DATA[3]	USART1 - RTS		
D1	50	E3	PX08	59	VDDIO	x2	EBI - DATA[2]	USART3 - RXD		
C1	49	E4	PX09	60	VDDIO	x2	EBI - DATA[1]	USART3 - TXD		
B1	37	D2	PX10	61	VDDIO	x2	EBI - DATA[0]	USART2 - RXD		
L1	67	K7(1)	PX11	62	VDDIO	x2	EBI - NWE1	USART2 - TXD		
D6	34	D1	PX12	63	VDDIO	x2	EBI - NWE0	USART2 - CTS	MCI - CLK	
C6	33	D3	PX13	64	VDDIO	x2	EBI - NRD	USART2 - RTS	MCI - CLK	
M4	68	K5(1)	PX14	65	VDDIO	x2	EBI - NCS[1]		TC0 - A0	
E6	40	K4(1)	PX15	66	VDDIO	x2	EBI - ADDR[19]	USART3 - RTS	TC0 - B0	
C5	32	D4(1)	PX16	67	VDDIO	x2	EBI - ADDR[18]	USART3 - CTS	TC0 - A1	
K6	83	J10(1)	PX17	68	VDDIO	x2	EBI - ADDR[17]	DMACA - DMARQ[1]	TC0 - B1	

Table 3-6. Signal Description List

Signal Name	Function	Type	Active Level	Comments
RESET_N	Reset Pin	Input	Low	
DMA Controller - DMACA (optional)				
DMAACK[1:0]	DMA Acknowledge	Output		
DMARQ[1:0]	DMA Requests	Input		
External Interrupt Controller - EIC				
EXTINT[7:0]	External Interrupt Pins	Input		
SCAN[7:0]	Keypad Scan Pins	Output		
NMI	Non-Maskable Interrupt Pin	Input	Low	
General Purpose Input/Output pin - GPIOA, GPIOB, GPIOC, GPIOX				
PA[31:0]	Parallel I/O Controller GPIO port A	I/O		
PB[11:0]	Parallel I/O Controller GPIO port B	I/O		
PC[5:0]	Parallel I/O Controller GPIO port C	I/O		
PX[59:0]	Parallel I/O Controller GPIO port X	I/O		
External Bus Interface - EBI				
ADDR[23:0]	Address Bus	Output		
CAS	Column Signal	Output	Low	
CFCE1	Compact Flash 1 Chip Enable	Output	Low	
CFCE2	Compact Flash 2 Chip Enable	Output	Low	
CFRNW	Compact Flash Read Not Write	Output		
DATA[15:0]	Data Bus	I/O		
NANDOE	NAND Flash Output Enable	Output	Low	
NANDWE	NAND Flash Write Enable	Output	Low	
NCS[5:0]	Chip Select	Output	Low	
NRD	Read Signal	Output	Low	
NWAIT	External Wait Signal	Input	Low	
NWE0	Write Enable 0	Output	Low	
NWE1	Write Enable 1	Output	Low	
RAS	Row Signal	Output	Low	

Table 3-6. Signal Description List

Signal Name	Function	Type	Active Level	Comments
SDA10	SDRAM Address 10 Line	Output		
SDCK	SDRAM Clock	Output		
SDCKE	SDRAM Clock Enable	Output		
SDWE	SDRAM Write Enable	Output	Low	
MultiMedia Card Interface - MCI				
CLK	Multimedia Card Clock	Output		
CMD[1:0]	Multimedia Card Command	I/O		
DATA[15:0]	Multimedia Card Data	I/O		
Memory Stick Interface - MSI				
SCLK	Memory Stick Clock	Output		
BS	Memory Stick Command	I/O		
DATA[3:0]	Multimedia Card Data	I/O		
Serial Peripheral Interface - SPI0, SPI1				
MISO	Master In Slave Out	I/O		
MOSI	Master Out Slave In	I/O		
NPCS[3:0]	SPI Peripheral Chip Select	I/O	Low	
SPCK	Clock	Output		
Synchronous Serial Controller - SSC				
RX_CLOCK	SSC Receive Clock	I/O		
RX_DATA	SSC Receive Data	Input		
RX_FRAME_SYNC	SSC Receive Frame Sync	I/O		
TX_CLOCK	SSC Transmit Clock	I/O		
TX_DATA	SSC Transmit Data	Output		
TX_FRAME_SYNC	SSC Transmit Frame Sync	I/O		
Timer/Counter - TC0, TC1				
A0	Channel 0 Line A	I/O		
A1	Channel 1 Line A	I/O		
A2	Channel 2 Line A	I/O		

Figure 4-2. The AVR32UC Pipeline

4.3.2 AVR32A Microarchitecture Compliance

AVR32UC implements an AVR32A microarchitecture. The AVR32A microarchitecture is targeted at cost-sensitive, lower-end applications like smaller microcontrollers. This microarchitecture does not provide dedicated hardware registers for shadowing of register file registers in interrupt contexts. Additionally, it does not provide hardware registers for the return address registers and return status registers. Instead, all this information is stored on the system stack. This saves chip area at the expense of slower interrupt handling.

Upon interrupt initiation, registers R8-R12 are automatically pushed to the system stack. These registers are pushed regardless of the priority level of the pending interrupt. The return address and status register are also automatically pushed to stack. The interrupt handler can therefore use R8-R12 freely. Upon interrupt completion, the old R8-R12 registers and status register are restored, and execution continues at the return address stored popped from stack.

The stack is also used to store the status register and return address for exceptions and *scall*. Executing the *rete* or *rets* instruction at the completion of an exception or system call will pop this status register and continue execution at the popped return address.

4.3.3 Java Support

AVR32UC does not provide Java hardware acceleration.

4.3.4 Memory Protection

The MPU allows the user to check all memory accesses for privilege violations. If an access is attempted to an illegal memory address, the access is aborted and an exception is taken. The MPU in AVR32UC is specified in the AVR32UC Technical Reference manual.

4.3.5 Unaligned Reference Handling

AVR32UC does not support unaligned accesses, except for doubleword accesses. AVR32UC is able to perform word-aligned *st.d* and *ld.d*. Any other unaligned memory access will cause an address exception. Doubleword-sized accesses with word-aligned pointers will automatically be performed as two word-sized accesses.

status register. Upon entry into Debug mode, hardware sets the SR[D] bit and jumps to the Debug Exception handler. By default, Debug mode executes in the exception context, but with dedicated Return Address Register and Return Status Register. These dedicated registers remove the need for storing this data to the system stack, thereby improving debuggability. The mode bits in the status register can freely be manipulated in Debug mode, to observe registers in all contexts, while retaining full privileges.

Debug mode is exited by executing the *retd* instruction. This returns to the previous context.

4.5.5 Entry Points for Events

Several different event handler entry points exist. In AVR32UC, the reset address is 0x8000_0000. This places the reset address in the boot flash memory area.

TLB miss exceptions and *scall* have a dedicated space relative to EVBA where their event handler can be placed. This speeds up execution by removing the need for a jump instruction placed at the program address jumped to by the event hardware. All other exceptions have a dedicated event routine entry point located relative to EVBA. The handler routine address identifies the exception source directly.

AVR32UC uses the ITLB and DTLB protection exceptions to signal a MPU protection violation. ITLB and DTLB miss exceptions are used to signal that an access address did not map to any of the entries in the MPU. TLB multiple hit exception indicates that an access address did map to multiple TLB entries, signalling an error.

All external interrupt requests have entry points located at an offset relative to EVBA. This autovector offset is specified by an external Interrupt Controller. The programmer must make sure that none of the autovector offsets interfere with the placement of other code. The autovector offset has 14 address bits, giving an offset of maximum 16384 bytes.

Special considerations should be made when loading EVBA with a pointer. Due to security considerations, the event handlers should be located in non-writeable flash memory, or optionally in a privileged memory protection region if an MPU is present.

If several events occur on the same instruction, they are handled in a prioritized way. The priority ordering is presented in Table 4-4. If events occur on several instructions at different locations in the pipeline, the events on the oldest instruction are always handled before any events on any younger instruction, even if the younger instruction has events of higher priority than the oldest instruction. An instruction B is younger than an instruction A if it was sent down the pipeline later than A.

The addresses and priority of simultaneous events are shown in Table 4-4. Some of the exceptions are unused in AVR32UC since it has no MMU, coprocessor interface, or floating-point unit.

5. Memories

5.1 Embedded Memories

- Internal High-Speed Flash
 - 256KBytes (AT32UC3A3256/S)
 - 128Kbytes (AT32UC3A3128/S)
 - 64Kbytes (AT32UC3A364/S)
 - 0 wait state access at up to 42MHz in worst case conditions
 - 1 wait state access at up to 84MHz 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
 - 64KBytes, Single-cycle access at full speed on CPU Local Bus and accessible through the High Speed Bud (HSB) matrix
 - 2x32KBytes, accessible independently through the High Speed Bud (HSB) matrix

5.2 Physical Memory Map

The System Bus is implemented as a bus matrix. All system bus addresses are fixed, and they are never remapped in any way, not even in boot.

Note that AVR32 UC CPU uses unsegmented translation, as described in the AVR32UC Technical Architecture Manual.

The 32-bit physical address space is mapped as follows:

Table 5-1. AT32UC3A3A4 Physical Memory Map

Device	Start Address	Size	Size	Size
		AT32UC3A3256S AT32UC3A3256 AT32UC3A4256S AT32UC3A4256	AT32UC3A3128S AT32UC3A3128 AT32UC3A4128S AT32UC3A4128	AT32UC3A364S AT32UC3A364 AT32UC3A464S AT32UC3A464
Embedded CPU SRAM	0x00000000	64KByte	64KByte	64KByte
Embedded Flash	0x80000000	256KByte	128KByte	64KByte
EBI SRAM CS0	0xC0000000	16MByte	16MByte	16MByte
EBI SRAM CS2	0xC8000000	16MByte	16MByte	16MByte
EBI SRAM CS3	0xCC000000	16MByte	16MByte	16MByte
EBI SRAM CS4	0xD8000000	16MByte	16MByte	16MByte
EBI SRAM CS5	0xDC000000	16MByte	16MByte	16MByte
EBI SRAM CS1 /SDRAM CS0	0xD0000000	128MByte	128MByte	128MByte
USB Data	0xE0000000	64KByte	64KByte	64KByte

Table 5-1. AT32UC3A3A4 Physical Memory Map

Device	Start Address	Size	Size	Size
		AT32UC3A3256S AT32UC3A3256 AT32UC3A4256S AT32UC3A4256	AT32UC3A3128S AT32UC3A3128 AT32UC3A4128S AT32UC3A4128	AT32UC3A364S AT32UC3A364 AT32UC3A464S AT32UC3A464
HRAMC0	0xFF000000	32KByte	32KByte	32KByte
HRAMC1	0xFF008000	32KByte	32KByte	32KByte
HSB-PB Bridge A	0xFFFF0000	64KByte	64KByte	64KByte
HSB-PB Bridge B	0xFFFE0000	64KByte	64KByte	64KByte

5.3 Peripheral Address Map

Table 5-2. Peripheral Address Mapping

Address		Peripheral Name
0xFF100000	DMACA	DMA Controller - DMACA
0xFFFD0000	AES	Advanced Encryption Standard - AES
0xFFFE0000	USB	USB 2.0 Device and Host Interface - USB
0xFFFE1000	HMATRIX	HSB Matrix - HMATRIX
0xFFFE1400	FLASHC	Flash Controller - FLASHC
0xFFFE1C00	SMC	Static Memory Controller - SMC
0xFFFE2000	SDRAMC	SDRAM Controller - SDRAMC
0xFFFE2400	ECCHRS	Error code corrector Hamming and Reed Solomon - ECCHRS
0xFFFE2800	BUSMON	Bus Monitor module - BUSMON
0xFFFE4000	MCI	Multimedia Card Interface - MCI
0xFFFE8000	MSI	Memory Stick Interface - MSI
0xFFFF0000	PDCA	Peripheral DMA Controller - PDCA
0xFFFF0800	INTC	Interrupt controller - INTC

Table 5-2. Peripheral Address Mapping

0xFFFF0C00	PM	Power Manager - PM
0xFFFF0D00	RTC	Real Time Counter - RTC
0xFFFF0D30	WDT	Watchdog Timer - WDT
0xFFFF0D80	EIC	External Interrupt Controller - EIC
0xFFFF1000	GPIO	General Purpose Input/Output Controller - GPIO
0xFFFF1400	USART0	Universal Synchronous/Asynchronous Receiver/Transmitter - USART0
0xFFFF1800	USART1	Universal Synchronous/Asynchronous Receiver/Transmitter - USART1
0xFFFF1C00	USART2	Universal Synchronous/Asynchronous Receiver/Transmitter - USART2
0xFFFF2000	USART3	Universal Synchronous/Asynchronous Receiver/Transmitter - USART3
0xFFFF2400	SPI0	Serial Peripheral Interface - SPI0
0xFFFF2800	SPI1	Serial Peripheral Interface - SPI1
0xFFFF2C00	TWIM0	Two-wire Master Interface - TWIM0
0xFFFF3000	TWIM1	Two-wire Master Interface - TWIM1
0xFFFF3400	SSC	Synchronous Serial Controller - SSC
0xFFFF3800	TC0	Timer/Counter - TC0
0xFFFF3C00	ADC	Analog to Digital Converter - ADC
0xFFFF4000	ABDAC	Audio Bitstream DAC - ABDAC
0xFFFF4400	TC1	Timer/Counter - TC1

6. Boot Sequence

This chapter summarizes the boot sequence of the AT32UC3A3/A4. The behavior after power-up is controlled by the Power Manager. For specific details, refer to [Section 7. "Power Manager \(PM\)" on page 86](#).

6.1 Starting of Clocks

After power-up, the device will be held in a reset state by the Power-On Reset circuitry, until the power has stabilized throughout the device. Once the power has stabilized, the device will use the internal RC Oscillator as clock source.

On system start-up, the PLLs are disabled. All clocks to all modules are running. No clocks have a divided frequency, all parts of the system receives a clock with the same frequency as the internal RC Oscillator.

6.2 Fetching of Initial Instructions

After reset has been released, the AVR32 UC CPU starts fetching instructions from the reset address, which is 0x8000_0000. This address points to the first address in the internal Flash.

The internal Flash uses VDDIO voltage during read and write operations. BOD33 monitors this voltage and maintains the device under reset until VDDIO reaches the minimum voltage, preventing any spurious execution from flash.

The code read from the internal Flash is free to configure the system to use for example the PLLs, to divide the frequency of the clock routed to some of the peripherals, and to gate the clocks to unused peripherals.

When powering up the device, there may be a delay before the voltage has stabilized, depending on the rise time of the supply used. The CPU can start executing code as soon as the supply is above the POR threshold, and before the supply is stable. Before switching to a high-speed clock source, the user should use the BOD to make sure the VDDCORE is above the minimum-level (1.62V).

7. Electrical Characteristics

7.1 Absolute Maximum Ratings*

Operating Temperature.....	-40°C to +85°C
Storage Temperature	-60°C to +150°C
Voltage on Input Pin with respect to Ground	-0.3V to 3.6V
Maximum Operating Voltage (VDDCORE)	1.95V
Maximum Operating Voltage (VDDIO).....	3.6V
Total DC Output Current on all I/O Pin	
for TQFP144 package	370 mA
for TFBGA144 package	370 mA

*NOTICE: Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

7.11 EBI Timings

7.11.1 SMC Signals

These timings are given for worst case process, T = 85°C, VDDIO = 3V and 40 pF load capacitance.

Table 7-30. SMC Clock Signal

Symbol	Parameter	Max. ⁽¹⁾	Unit
1/(t _{CPSMC})	SMC Controller Clock Frequency	1/(t _{cpCPU})	MHz

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

Table 7-31. SMC Read Signals with Hold Settings

Symbol	Parameter	Min.	Unit
NRD Controlled (READ_MODE = 1)			
SMC ₁	Data Setup before NRD High	12	ns
SMC ₂	Data Hold after NRD High	0	ns
SMC ₃	NRD High to NBS0/A0 Change ⁽¹⁾	nrd hold length * t _{CPSMC} - 1.3	ns
SMC ₄	NRD High to NBS1 Change ⁽¹⁾	nrd hold length * t _{CPSMC} - 1.3	ns
SMC ₅	NRD High to NBS2/A1 Change ⁽¹⁾	nrd hold length * t _{CPSMC} - 1.3	ns
SMC ₇	NRD High to A2 - A23 Change ⁽¹⁾	nrd hold length * t _{CPSMC} - 1.3	ns
SMC ₈	NRD High to NCS Inactive ⁽¹⁾	(nrd hold length - ncs rd hold length) * t _{CPSMC} - 2.3	ns
SMC ₉	NRD Pulse Width	nrd pulse length * t _{CPSMC} - 1.4	ns
NRD Controlled (READ_MODE = 0)			
SMC ₁₀	Data Setup before NCS High	11.5	ns
SMC ₁₁	Data Hold after NCS High	0	ns
SMC ₁₂	NCS High to NBS0/A0 Change ⁽¹⁾	ncs rd hold length * t _{CPSMC} - 2.3	ns
SMC ₁₃	NCS High to NBS0/A0 Change ⁽¹⁾	ncs rd hold length * t _{CPSMC} - 2.3	ns
SMC ₁₄	NCS High to NBS2/A1 Change ⁽¹⁾	ncs rd hold length * t _{CPSMC} - 2.3	ns
SMC ₁₆	NCS High to A2 - A23 Change ⁽¹⁾	ncs rd hold length * t _{CPSMC} - 4	ns
SMC ₁₇	NCS High to NRD Inactive ⁽¹⁾	ncs rd hold length - nrd hold length) * t _{CPSMC} - 1.3	ns
SMC ₁₈	NCS Pulse Width	ncs rd pulse length * t _{CPSMC} - 3.6	ns

Note: 1. hold length = total cycle duration - setup duration - pulse duration. "hold length" is for "ncs rd hold length" or "nrd hold length".

Table 7-32. SMC Read Signals with no Hold Settings

Symbol	Parameter	Min.	Unit
NRD Controlled (READ_MODE = 1)			
SMC ₁₉	Data Setup before NRD High	13.7	ns
SMC ₂₀	Data Hold after NRD High	1	ns
NRD Controlled (READ_MODE = 0)			
SMC ₂₁	Data Setup before NCS High	13.3	ns
SMC ₂₂	Data Hold after NCS High	0	ns

Table 7-33. SMC Write Signals with Hold Settings

Symbol	Parameter	Min.	Unit
NRD Controlled (READ_MODE = 1)			
SMC ₂₃	Data Out Valid before NWE High	$(nwe \text{ pulse length} - 1) * t_{CPSMC} - 0.9$	ns
SMC ₂₄	Data Out Valid after NWE High ⁽¹⁾	$nwe \text{ hold length} * t_{CPSMC} - 6$	ns
SMC ₂₅	NWE High to NBS0/A0 Change ⁽¹⁾	$nwe \text{ hold length} * t_{CPSMC} - 1.9$	ns
SMC ₂₆	NWE High to NBS1 Change ⁽¹⁾	$nwe \text{ hold length} * t_{CPSMC} - 1.9$	ns
SMC ₂₉	NWE High to A1 Change ⁽¹⁾	$nwe \text{ hold length} * t_{CPSMC} - 1.9$	ns
SMC ₃₁	NWE High to A2 - A23 Change ⁽¹⁾	$nwe \text{ hold length} * t_{CPSMC} - 1.7$	ns
SMC ₃₂	NWE High to NCS Inactive ⁽¹⁾	$(nwe \text{ hold length} - ncs \text{ wr hold length}) * t_{CPSMC} - 2.9$	ns
SMC ₃₃	NWE Pulse Width	$nwe \text{ pulse length} * t_{CPSMC} - 0.9$	ns
NRD Controlled (READ_MODE = 0)			
SMC ₃₄	Data Out Valid before NCS High	$(ncs \text{ wr pulse length} - 1) * t_{CPSMC} - 4.6$	ns
SMC ₃₅	Data Out Valid after NCS High ⁽¹⁾	$ncs \text{ wr hold length} * t_{CPSMC} - 5.8$	ns
SMC ₃₆	NCS High to NWE Inactive ⁽¹⁾	$(ncs \text{ wr hold length} - nwe \text{ hold length}) * t_{CPSMC} - 0.6$	ns

Note: 1. hold length = total cycle duration - setup duration - pulse duration. "hold length" is for "ncs wr hold length" or "nwe hold length"

Table 7-34. SMC Write Signals with No Hold Settings (NWE Controlled only)

Symbol	Parameter	Min.	Unit
SMC ₃₇	NWE Rising to A2-A25 Valid	5.4	ns
SMC ₃₈	NWE Rising to NBS0/A0 Valid	5	ns
SMC ₃₉	NWE Rising to NBS1 Change	5	ns
SMC ₄₀	NWE Rising to A1/NBS2 Change	5	ns
SMC ₄₁	NWE Rising to NBS3 Change	5	ns
SMC ₄₂	NWE Rising to NCS Rising	5.1	ns

Table 7-36. SDRAM Clock Signal

Symbol	Parameter	Conditions	Min.	Max.	Unit
SDRAMC ₁₃	Bank Change before SDCK Rising Edge		6.3		ns
SDRAMC ₁₄	Bank Change after SDCK Rising Edge		2.4		ns
SDRAMC ₁₅	CAS Low before SDCK Rising Edge		7.4		ns
SDRAMC ₁₆	CAS High after SDCK Rising Edge		1.9		ns
SDRAMC ₁₇	DQM Change before SDCK Rising Edge		6.4		ns
SDRAMC ₁₈	DQM Change after SDCK Rising Edge		2.2		ns
SDRAMC ₁₉	D0-D15 in Setup before SDCK Rising Edge		9		ns
SDRAMC ₂₀	D0-D15 in Hold after SDCK Rising Edge		0		ns
SDRAMC ₂₃	SDWE Low before SDCK Rising Edge		7.6		ns
SDRAMC ₂₄	SDWE High after SDCK Rising Edge		1.8		ns
SDRAMC ₂₅	D0-D15 Out Valid before SDCK Rising Edge		7.1		ns
SDRAMC ₂₆	D0-D15 Out Valid after SDCK Rising Edge		1.5		ns

Figure 7-9. SDRAMC Signals relative to SDCK.

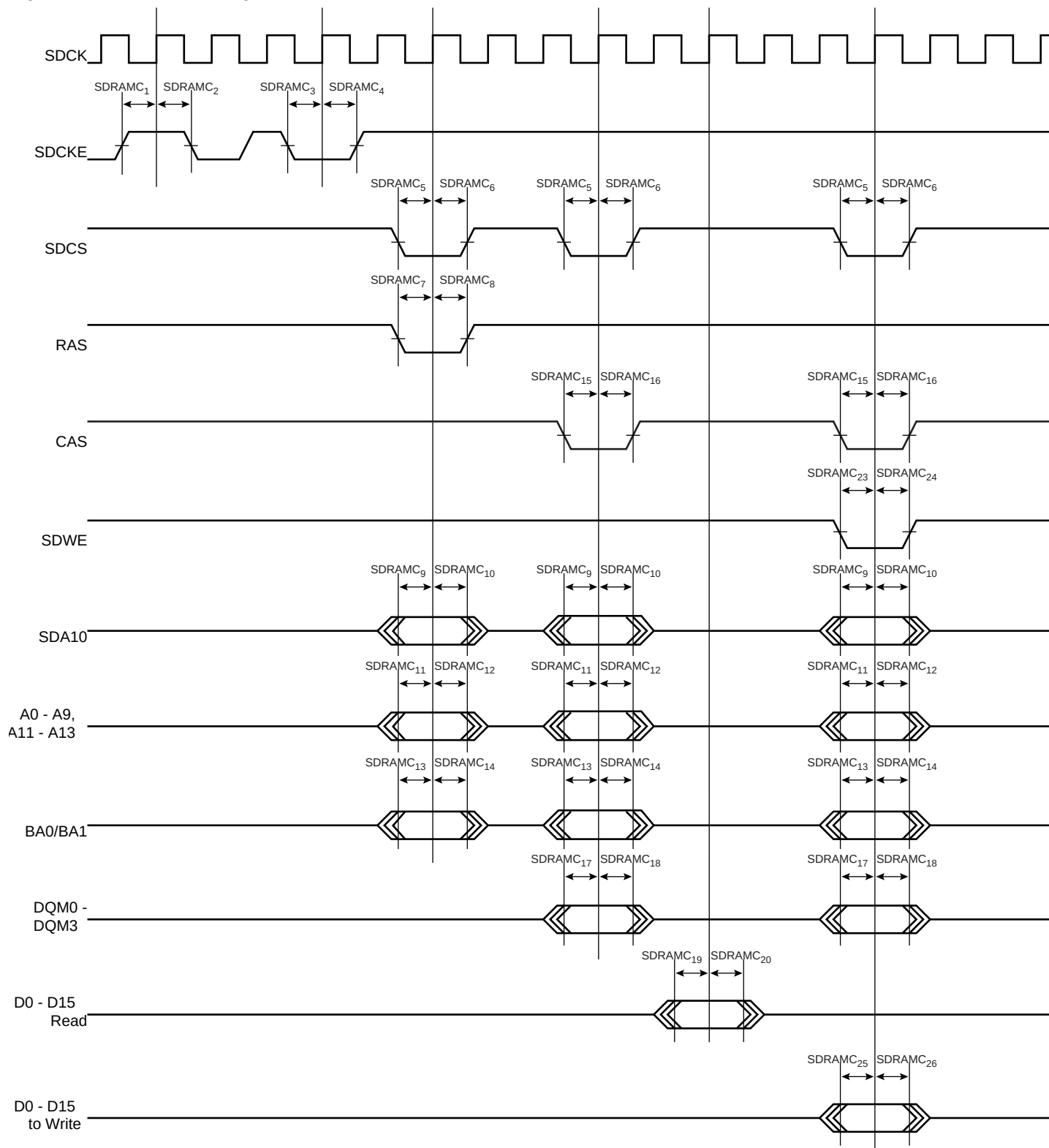


Table 7-38. SPI Timings

Symbol	Parameter	Conditions ⁽¹⁾	Min.	Max.	Unit
SPI ₀	MISO Setup time before SPCK rises (master)	3.3V domain	22 + (t _{CPMCK})/2 ⁽²⁾		ns
SPI ₁	MISO Hold time after SPCK rises (master)	3.3V domain	0		ns
SPI ₂	SPCK rising to MOSI Delay (master)	3.3V domain		7	ns
SPI ₃	MISO Setup time before SPCK <u>falls</u> (master)	3.3V domain	22 + (t _{CPMCK})/2 ⁽³⁾		ns
SPI ₄	MISO Hold time after SPCK falls (master)	3.3V domain	0		ns
SPI ₅	SPCK falling to MOSI Delay (master)	3.3V domain		7	ns
SPI ₆	SPCK falling to MISO Delay (slave)	3.3V domain		26.5	ns
SPI ₇	MOSI Setup time before SPCK rises (slave)	3.3V domain	0		ns
SPI ₈	MOSI Hold time after SPCK rises (slave)	3.3V domain	1.5		ns
SPI ₉	SPCK rising to MISO Delay (slave)	3.3V domain		27	ns
SPI ₁₀	MOSI Setup time before SPCK falls (slave)	3.3V domain	0		ns
SPI ₁₁	MOSI Hold time after SPCK falls (slave)	3.3V domain	1		ns

1. 3.3V domain: V_{VDDIO} from 3.0V to 3.6V, maximum external capacitor = 40 pF

2. t_{CPMCK}: Master Clock period in ns.

3. t_{CPMCK}: Master Clock period in ns.

7.14 MCI

The High Speed MultiMedia Card Interface (MCI) supports the MultiMedia Card (MMC) Specification V4.2, the SD Memory Card Specification V2.0, the SDIO V1.1 specification and CE-ATA V1.1.

9. Ordering Information

Device	Ordering Code	Package	Conditioning	Temperature Operating Range
AT32UC3A3256S	AT32UC3A3256S-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3256S-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A3256S-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3256S-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A3256	AT32UC3A3256-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3256-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A3256-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3256-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A3128S	AT32UC3A3128S-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3128S-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A3128S-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3128S-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A3128	AT32UC3A3128-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3128-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A3128-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A3128-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A364S	AT32UC3A364S-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A364S-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A364S-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A364S-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A364	AT32UC3A364-ALUT	144-lead LQFP	Tray	Industrial (-40-C to 85-C)
	AT32UC3A364-ALUR	144-lead LQFP	Reels	Industrial (-40-C to 85-C)
	AT32UC3A364-CTUT	144-ball TFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A364-CTUR	144-ball TFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A4256S	AT32UC3A4256S-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A4256S-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A4256	AT32UC3A4256-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A4256-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A4128S	AT32UC3A4128S-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A4128S-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A4128	AT32UC3A4128-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A4128-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A464S	AT32UC3A464S-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A464S-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)
AT32UC3A464	AT32UC3A464-C1UT	100-ball VFBGA	Tray	Industrial (-40-C to 85-C)
	AT32UC3A464-C1UR	100-ball VFBGA	Reels	Industrial (-40-C to 85-C)

SMBALERT bit may be set after reset

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.

10.2.13 TWIS

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.

TWIS stretch on Address match error

When the TWIS stretches TWCK due to a slave address match, it also holds TWD low for the same duration if it is to be receiving data. When TWIS releases TWCK, it releases TWD at the same time. This can cause a TWI timing violation.

Fix/Workaround

None.

10.2.14 MCI

MCI_CLK features is not available on PX12, PX13 and PX40

Fix/Workaround

MCI_CLK feature is available on PA27 only.

The busy signal of the responses R1b is not taken in account for CMD12 STOP_TRANSFER

It is not possible to know the busy status of the card during the response (R1b) for the commands CMD12.

Fix/Workaround

The card busy line should be polled through the GPIO Input Value register (IVR) for commands CMD12.

10.2.15 SSC

Frame Synchro and Frame Synchro Data are delayed by one clock cycle

The frame synchro and the frame synchro data are delayed from 1 SSC_CLOCK when:

- Clock is CKDIV
- The START is selected on either a frame synchro edge or a level
- Frame synchro data is enabled
- Transmit clock is gated on output (through CKO field)

Fix/Workaround

Transmit or receive CLOCK must not be gated (by the mean of CKO field) when START condition is performed on a generated frame synchro.

1	Description	3
2	Overview	4
2.1	Block Diagram	4
2.2	Configuration Summary	5
3	Package and Pinout	6
3.1	Package	6
3.2	Peripheral Multiplexing on I/O lines	9
3.3	Signal Descriptions	14
3.4	I/O Line Considerations	19
3.5	Power Considerations	20
4	Processor and Architecture	21
4.1	Features	21
4.2	AVR32 Architecture	21
4.3	The AVR32UC CPU	22
4.4	Programming Model	26
4.5	Exceptions and Interrupts	30
5	Memories	34
5.1	Embedded Memories	34
5.2	Physical Memory Map	34
5.3	Peripheral Address Map	35
5.4	CPU Local Bus Mapping	37
6	Boot Sequence	39
6.1	Starting of Clocks	39
6.2	Fetching of Initial Instructions	39
7	Electrical Characteristics	40
7.1	Absolute Maximum Ratings*	40
7.2	DC Characteristics	41
7.3	I/O pin Characteristics	42
7.4	Regulator characteristics	43
7.5	Analog characteristics	44
7.6	Power Consumption	48
7.7	System Clock Characteristics	51
7.8	Oscillator Characteristics	52
7.9	ADC Characteristics	54