Microchip Technology - AT94K10AL-25BQU Datasheet

Welcome to E-XFL.COM

Embedded - FPGAs (Field Programmable Gate Array) with Microcontrollers: Enhancing Flexibility and Performance

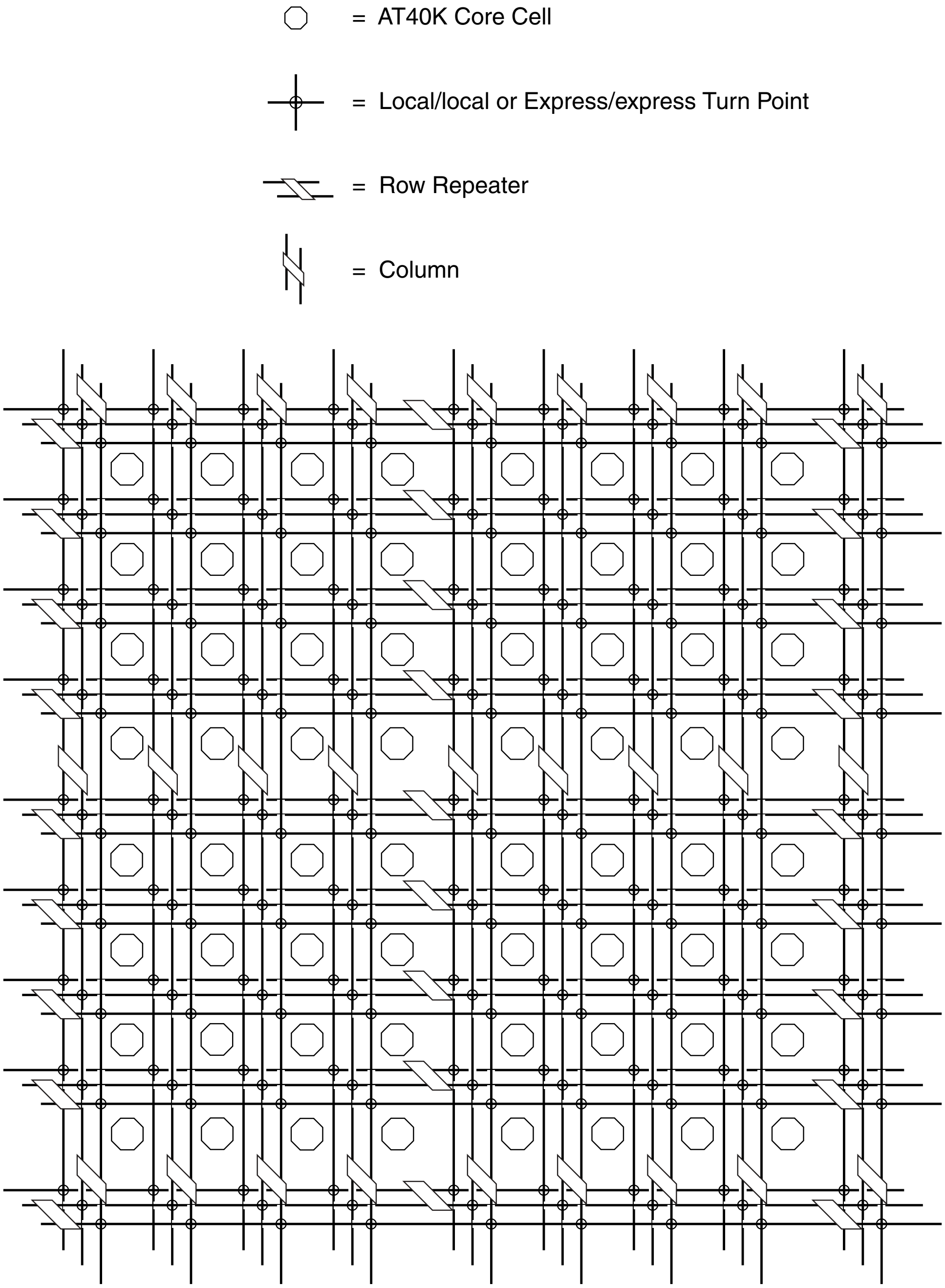
Embedded - FPGAs (Field Programmable Gate Arrays) with Microcontrollers represent a cutting-edge category of electronic components that combine the flexibility of FPGA technology with the processing power of integrated microcontrollers. This hybrid approach offers a versatile solution for designing and implementing complex digital systems that require both programmable logic and embedded processing capabilities.

What Are Embedded - FPGAs with Microcontrollers?

At their core, **FPGAs** are semiconductor devices that can

Details	
Product Status	Active
Core Type	8-Bit AVR
Speed	25 MHz
Interface	I²C, UART
Program SRAM Bytes	20K-32K
FPGA SRAM	4kb
EEPROM Size	-
Data SRAM Bytes	4K ~ 16K
FPGA Core Cells	576
FPGA Gates	10K
FPGA Registers	846
Voltage - Supply	3V ~ 3.6V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 85°C
Package / Case	144-LQFP
Supplier Device Package	144-LQFP (20x20)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at94k10al-25bqu

Figure 4. Busing Plane (One of Five)



Program and Data SRAM

Up to 36 Kbytes of 15 ns dual-port SRAM reside between the FPGA and the AVR. This SRAM is used by the AVR for program instruction and general-purpose data storage. The AVR is connected to one side of this SRAM; the FPGA is connected to the other side. The port connected to the FPGA is used to store data without using up bandwidth on the AVR system data bus.

The FPGA core communicates directly with the data SRAM(1) block, viewing all SRAM memory space as 8-bit memory.

Note: 1. The unused bits for the FPGA-SRAM address must tie to ‘0’ because there is no pull-down circuitry.

For the AT94K10 and AT94K40, the internal program and data SRAM is divided into three blocks: 10 Kbytes x 16 dedicated program SRAM, 4 Kbytes x 8 dedicated data SRAM and 6 Kbytes x 16 or 12 Kbytes x 8 configurable SRAM, which may be swapped between program and data memory spaces in 2 Kbytes x 16 or 4 Kbytes x 8 partitions.

For the AT94K05, the internal program and data SRAM is divided into three blocks: 4 Kbytes 16 dedicated program SRAM, 4 Kbytes x 8 dedicated data SRAM and 6 Kbytes x 16 or 12 Kbytes x 8 configurable SRAM, which may be swapped between program and data memory spaces in 2 Kbytes x 16 or 4 Kbytes x 8 partitions.

The addressing scheme for the configurable SRAM partitions prevents program instructions from overwriting data words and vice versa. Once configured (SCR41:40 – See “System Control Register – FPGA/AVR” on page 30.), the program memory space remains isolated from the data memory space. SCR41:40 controls internal muxes. Write enable signals allow the memory to be safely segmented. Figure 19 shows the FPSLIC configurable allocation SRAM memory.

Instruction Set Nomenclature (Summary)

The complete “AVR Instruction Set” document is available on the Atmel web site, at http://www.atmel.com/atmel/acrobat/doc0856.pdf.

Status Register (SREG)

SREG:	Status register
C:	Carry flag in status register
Z:	Zero flag in status register
N:	Negative flag in status register
V:	Two’s complement overflow indicator
S:	$N \oplus V$, For signed tests
H:	Half-carry flag in the status register
T:	Transfer bit used by BLD and BST instructions
I:	Global interrupt enable/disable flag

Registers and Operands

Rd:	Destination (and source) register in the register file
Rr:	Source register in the register file
R:	Result after instruction is executed
K:	Constant data
k:	Constant address
b:	Bit in the register file or I/O register ($0 \le b \le 7$)
s:	Bit in the status register ($0 \le s \le 2$)
X,Y,Z:	Indirect address register (X = R27:R26, Y = R29:R28 and Z = R31:R30)
A:	I/O location address
q:	Displacement for direct addressing ($0 \le q \le 63$)

I/O Registers

Stack

STACK:	Stack for return address and pushed registers
SP:	Stack Pointer to STACK

Flags

⇔:	Flag affected by instruction
0:	Flag cleared by instruction
1:	Flag set by instruction
-:	Flag not affected by instruction

The instructions EIJMP, EICALL, ELPM, GPM, ESPM (from the megaAVR Instruction Set) are not supported in the FPSLIC device.

Figure 29. The Parallel Instruction Fetches and Instruction Executions

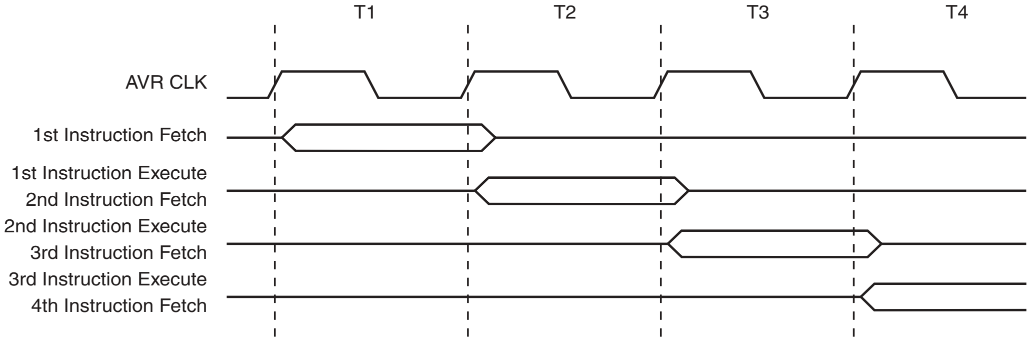
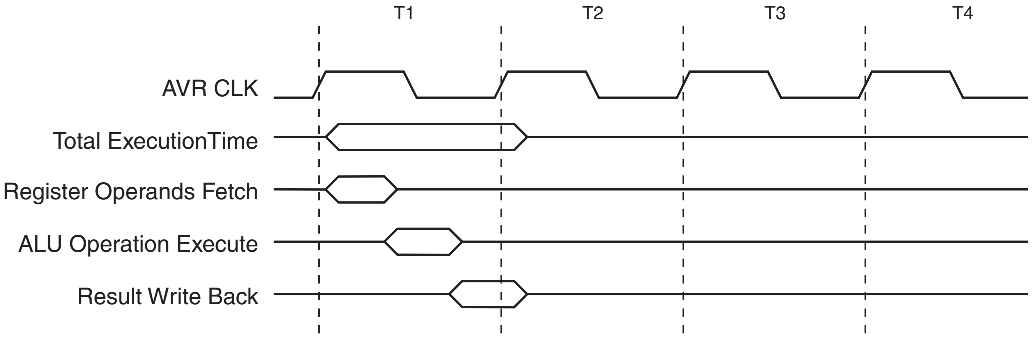


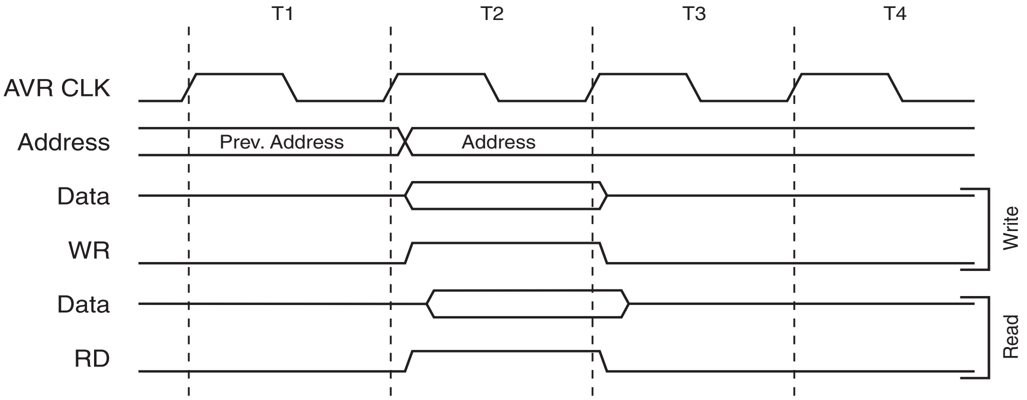
Figure 30 shows the internal timing concept for the register file. In a single clock cycle an ALU operation using two register operands is executed, and the result is stored back to the destination register.

Figure 30. Single Cycle ALU Operation



The internal data SRAM access is performed in two system clock cycles as described in Figure 31.

Figure 31. On-chip Data SRAM Access Cycles



MCU Control Status/Register – MCUR

The MCU Register contains control bits for general MCU functions and status bits to indicate the source of an MCU reset.

Bit	7	6	5	4	3	2	1	0	
$35 ($55)	**JTRF**	**JTD**	**SE**	**SM1**	**SM0**	**PORF**	**WDRF**	**EXTRF**	**MCUR**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	1	0	1	

- **Bit 7 - JTRF: JTAG Reset Flag**

This flag is set (one) upon issuing the AVR_RESET ($C) JTAG instruction. The flag can only be cleared (zero) by writing a zero to the JTRF bit or by a power-on reset. The bit will not be cleared by hardware during AVR reset.

- **Bit 6 - JTD: JTAG Disable**

When this bit is cleared (zero), and the System Control Register JTAG Enable bit is set (one), the JTAG interface is disabled. To avoid unintentional disabling or enabling of the JTAG interface, a timed sequence must be followed when changing this bit: the application software must write this bit to the desired value twice within four cycles to change its value.

- **Bit 5 - SE: Sleep Enable**

The SE bit must be set (one) to make the MCU enter the Sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the Sleep mode unless it is the programmers purpose, it is recommended to set the Sleep Enable SE bit just before the execution of the SLEEP instruction.

- **Bits 4, 3 - SM1/SM0: Sleep Mode Select Bits 1 and 0**

This bit selects between the three available Sleep modes as shown in Table 12.

- **Bit 2 - PORF: Power-on Reset Flag**

This flag is set (one) upon power-up of the device. The flag can only be cleared (zero) by writing a zero to the PORF bit. The bit will not be cleared by the hardware during AVR reset.

- **Bit 1 - WDRF: Watchdog Reset Flag**

This bit is set if a watchdog reset occurs. The bit is cleared by writing a logic 0 to the flag.

- **Bit 0 - EXTRF: External (Software) Reset Flag**

This flag is set (one) in three separate circumstances: power-on reset, use of Resetn/AVRResetn and writing a one to the SRST bit in the Software Control Register – SFTCR. The PORF flag can be checked to eliminate power-on reset as a cause for this flag to be set. There is no way to differentiate between use of Resetn/AVRResetn and software reset. The flag can only be cleared (zero) by writing a zero to the EXTRF bit. The bit will not be cleared by the hardware during AVR reset.

Table 12. Sleep Mode Select

SM1	SM0	Sleep Mode
0	0	Idle
0	1	Reserved
1	0	Power-down
1	1	Power-save

Idle Mode

When the SM1/SM0 bits are set to 00, the SLEEP instruction makes the MCU enter the Idle mode, stopping the CPU but allowing UARTs, Timer/Counters, Watchdog 2-wire Serial and the Interrupt System to continue operating. This enables the MCU to wake-up from external triggered interrupts as well as internal ones like the Timer Overflow and UART Receive Complete interrupts. When the MCU wakes up from Idle mode, the CPU starts program execution immediately.

Power-down Mode

When the SM1/SM0 bits are set to 10, the SLEEP instruction makes the MCU enter the Power-down mode. In this mode, the external oscillator is stopped, while the external interrupts and the watchdog (if enabled) continue operating. Only an external reset, a watchdog reset (if enabled), or an external level interrupt can wake-up the MCU.

In Power-down and Power-save modes, the four external interrupts, EXT_INT0...3, and FPGA interrupts, FPGA_INT0...3, are treated as low-level triggered interrupts.

If a level-triggered interrupt is used for wake-up from Power-down mode, the changed level must be held for some time to wake-up the MCU. This makes the MCU less sensitive to noise. The changed level is sampled twice by the watchdog oscillator clock, and if the input has the required level during this time, the MCU will wake-up. The period of the watchdog oscillator is 1 µs (nominal) at 3.3V and 25°C. The frequency of the watchdog oscillator is voltage dependent.

When waking up from Power-down mode, there is a delay from the wake-up condition occurs until the wake-up becomes effective. This allows the clock to restart and become stable after having been stopped. The wake-up period is defined by the same time-set bits that define the reset time-out period. The wake-up period is equal to the clock reset period, as shown in Figure 22 on page 89.

If the wake-up condition disappears before the MCU wakes up and starts to execute, the interrupt causing the wake-up will not be executed.

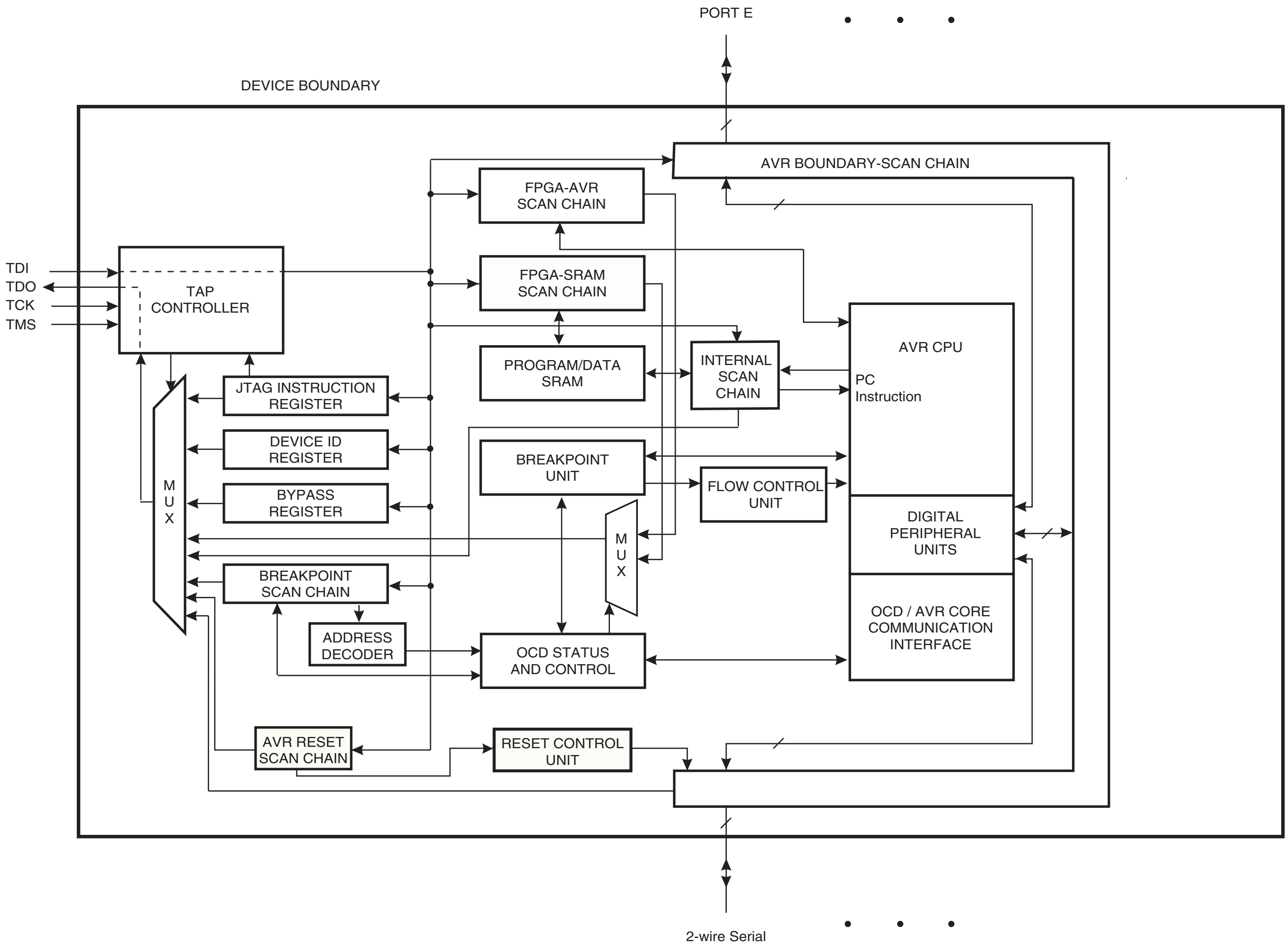
Power-save Mode

When the SM1/SM0 bits are 11, the SLEEP instruction makes the MCU enter the Power-save mode. This mode is identical to power-down, with one exception:

If Timer/Counter2 is clocked asynchronously, i.e., the AS2 bit in ASSR is set, Timer/Counter2 will run during sleep. In addition to the power-down wake-up sources, the device can also wake-up from either Timer Overflow or Output Compare event from Timer/Counter2 if the corresponding Timer/Counter2 interrupt enable bits are set in TIMSK. To ensure that the part executes the Interrupt routine when waking up, also set the global interrupt enable bit in SREG.

When waking up from Power-save mode by an external interrupt, two instruction cycles are executed before the interrupt flags are updated. When waking up by the asynchronous timer, three instruction cycles are executed before the flags are updated. During these cycles, the processor executes instructions, but the interrupt condition is not readable, and the interrupt routine has not started yet. See Table 3 on page 15 for clock activity during Power-down, Power-save and Idle modes.

Figure 39. Block Diagram



undetermined state when exiting the test mode. If needed, the BYPASS instruction can be issued to make the shortest possible scan chain through the device. The AVR can be set in the reset state either by pulling the external $\overline{\text{AVR RESET}}$ pin Low, or issuing the AVR_RESET instruction with appropriate setting of the Reset Data Register.

The EXTEST instruction is used for sampling external pins and loading output pins with data. The data from the output latch will be driven out on the pins as soon as the EXTEST instruction is loaded into the JTAG IR-register. Therefore, the SAMPLE/PRELOAD should also be used for setting initial values to the scan ring, to avoid damaging the board when issuing the EXTEST instruction for the first time. SAMPLE/PRELOAD can also be used for taking a snapshot of the AVR's external pins during normal operation of the part.

The JTAG Enable bit must be programmed and the JTD bit in the I/O register MCUR must be cleared to enable the JTAG Test Access Port.

When using the JTAG interface for Boundary-Scan, using a JTAG TCK clock frequency higher than the internal chip frequency is possible. The chip clock is not required to run.

Data Registers

The Data Registers are selected by the JTAG instruction registers described in section "Boundary-scan Specific JTAG Instructions" on page 75. The data registers relevant for Boundary-Scan operations are:

- Bypass Register
- Device Identification Register
- AVR Reset Register
- AVR Boundary-Scan Chain

Bypass Register

The Bypass register consists of a single shift-register stage. When the Bypass register is selected as path between TDI and TDO, the register is reset to 0 when leaving the Capture-DR controller state. The Bypass register can be used to shorten the scan chain on a system when the other devices are to be tested.

Device Identification Register

Figure 41 shows the structure of the Device Identification register.

Figure 41. The format of the Device Identification Register

	MSB						LSB
Bit	31	28	27	12	11	1	0
Device ID	**Version**		**Part Number**		**Manufacturer ID**		**1**
	4 bits		16 bits		11 bits		1 bit

Version

Version is a 4-bit number identifying the revision of the component. The relevant version numbers are shown in Table 18.

Table 18. JTAG Part Version

Device	Version (Binary Digits)
AT94K05	–
AT94K10	0010
AT94K40	–

EXTEST; $0

Mandatory JTAG instruction for selecting the Boundary-Scan Chain as Data Register for testing circuitry external to the AVR package. For port-pins, Pull-up Disable, Output Control, Output Data, and Input Data are all accessible in the scan chain. For Analog circuits having off-chip connections, the interface between the analog and the digital logic is in the scan chain. The contents of the latched outputs of the Boundary-Scan chain are driven out as soon as the JTAG IR-register is loaded by the EXTEST instruction.

The active states are:

- Capture-DR: Data on the external pins are sampled into the Boundary-Scan Chain.
- Shift-DR: The Internal Scan Chain is shifted by the TCK input.
- Update-DR: Data from the scan chain is applied to output pins.

IDCODE; $1

Optional JTAG instruction selecting the 32-bit ID register as Data Register. The ID register consists of a version number, a device number and the manufacturer code chosen by JEDEC. This is the default instruction after power-up.

The active states are:

- Capture-DR: Data in the IDCODE register is sampled into the Boundary-Scan Chain.
- Shift-DR: The IDCODE scan chain is shifted by the TCK input.

SAMPLE_PRELOAD; $2

Mandatory JTAG instruction for pre-loading the output latches and taking a snap-shot of the input/output pins without affecting the system operation. However, the output latches are not connected to the pins. The Boundary-Scan Chain is selected as Data Register.

The active states are:

- Capture-DR: Data on the external pins are sampled into the Boundary-Scan Chain.
- Shift-DR: The Boundary-Scan Chain is shifted by the TCK input.
- Update-DR: Data from the Boundary-Scan chain is applied to the output latches. However, the output latches are not connected to the pins.

AVR_RESET; $C

The AVR specific public JTAG instruction for forcing the AVR device into the Reset Mode or releasing the JTAG reset source. The TAP controller is not reset by this instruction. The one bit Reset Register is selected as Data Register. Note that the reset will be active as long as there is a logic “1” in the Reset Chain. The output from this chain is not latched.

The active state is:

- Shift-DR: The Reset Register is shifted by the TCK input.

BYPASS; $F

Mandatory JTAG instruction selecting the Bypass Register for Data Register.

The active states are:

- Capture-DR: Loads a logic “0” into the Bypass Register.
- Shift-DR: The Bypass Register cell between TDI and TDO is shifted.

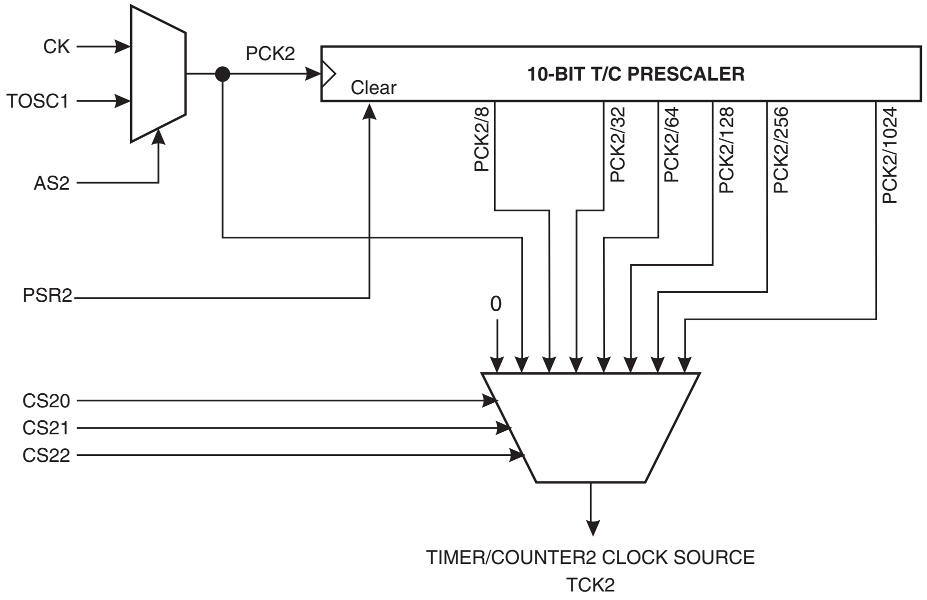
Boundary-scan Chain

The Boundary-Scan chain has the capability of driving and observing the logic levels on the AVR’s digital I/O pins.

Scanning the Digital Port Pins

Figure 43 shows the boundary-scan cell for bi-directional port pins with pull-up function. The cell consists of a standard boundary-scan cell for the pull-up function, and a bi-directional pin cell that combines the three signals Output Control (OC), Output Data (OD), and Input Data (ID), into only a two-stage shift register.

Figure 49. Timer/Counter2 Prescaler



Special Function I/O Register – SFIOR

Bit	7	6	5	4	3	2	1	0	
$30 ($50)	-	-	-	-	-	-	**PSR2**	**PSR10**	**SFIOR**
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits 7..2 - Res: Reserved Bits**

These bits are reserved bits in the FPSLIC and are always read as zero.

- **Bit 1 - PSR2: Prescaler Reset Timer/Counter2**

When this bit is set (one) the Timer/Counter2 prescaler will be reset. The bit will be cleared by the hardware after the operation is performed. Writing a zero to this bit will have no effect. This bit will always be read as zero if Timer/Counter2 is clocked by the internal CPU clock. If this bit is written when Timer/Counter2 is operating in asynchronous mode; however, the bit will remain as one until the prescaler has been reset. See "Asynchronous Operation of Timer/Counter2" on page 94 for a detailed description of asynchronous operation.

- **Bit 0 - PSR10: Prescaler Reset Timer/Counter1 and Timer/Counter0**

When this bit is set (one) the Timer/Counter1 and Timer/Counter0 prescaler will be reset. The bit will be cleared by the hardware after the operation is performed. Writing a zero to this bit will have no effect. Note that Timer/Counter1 and Timer/Counter0 share the same prescaler and a reset of this prescaler will affect both timers. This bit will always be read as zero.

8-bit Timers/Counters T/C0 and T/C2

Figure 50 shows the block diagram for Timer/Counter0. Figure 51 shows the block diagram for Timer/Counter2.

Table 26. PWM Outputs OCRn = $00 or $FF[1]

COMn1[2]	COMn0[2]	OCRn[2]	Output PWMn[2]
1	0	$00	L
1	0	$FF	H
1	1	$00	H
1	1	$FF	L

Notes: 1. n overflow PWM mode, this table is only valid for OCRn = $FF

2. n = 0 or 2

In up/down PWM mode, the Timer Overflow Flag, TOV0 or TOV2, is set when the counter advances from $00. In overflow PWM mode, the Timer Overflow Flag is set as in normal Timer/Counter mode. Timer Overflow Interrupts 0 and 2 operate exactly as in normal Timer/Counter mode, i.e. they are executed when TOV0 or TOV2 are set provided that Timer Overflow Interrupt and global interrupts are enabled. This does also apply to the Timer Output Compare flag and interrupt.

Asynchronous Status Register – ASSR

Bit	7	6	5	4	3	2	1	0	
$26 ($46)	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	ASSR
Read/Write	R	R	R	R	R/W	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7..4 - Res: Reserved Bits**

These bits are reserved bits in the FPSLIC and are always read as zero.

- **Bit 3 - AS2: Asynchronous Timer/Counter2 Mode**

When this bit is cleared (zero) Timer/Counter2 is clocked from the internal system clock, CK. If AS2 is set, the Timer/Counter2 is clocked from the TOSC1 pin. When the value of this bit is changed the contents of TCNT2, OCR2 and TCCR2 might get corrupted.

- **Bit 2 - TCN2UB: Timer/Counter2 Update Busy**

When Timer/Counter2 operates asynchronously and TCNT2 is written, this bit becomes set (one). When TCNT2 has been updated from the temporary storage register, this bit is cleared (zero) by the hardware. A logic 0 in this bit indicates that TCNT2 is ready to be updated with a new value.

- **Bit 1 - OCR2UB: Output Compare Register2 Update Busy**

When Timer/Counter2 operates asynchronously and OCR2 is written, this bit becomes set (one). When OCR2 has been updated from the temporary storage register, this bit is cleared (zero) by the hardware. A logic 0 in this bit indicates that OCR2 is ready to be updated with a new value.

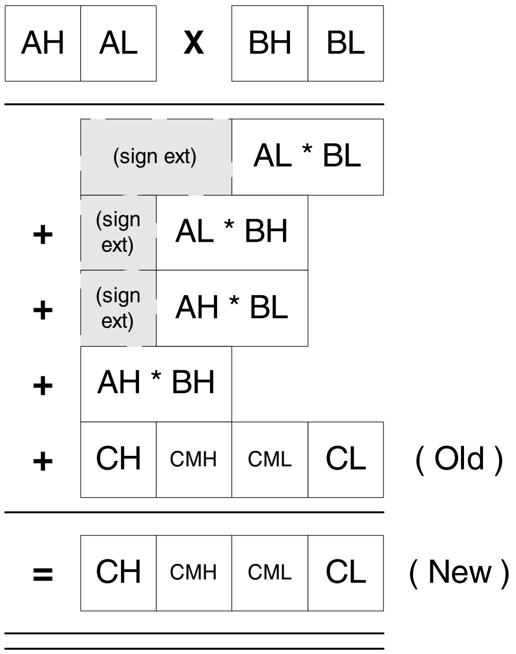
- **Bit 0 - TCR2UB: Timer/Counter Control Register2 Update Busy**

When Timer/Counter2 operates asynchronously and TCCR2 is written, this bit becomes set (one). When TCCR2 has been updated from the temporary storage register, this bit is cleared (zero) by the hardware. A logic 0 in this bit indicates that TCCR2 is ready to be updated with a new value.

If a write is performed to any of the three Timer/Counter2 registers while its update busy flag is set (one), the updated value might get corrupted and cause an unintentional interrupt to occur.

16-bit Multiply-accumulate Operation

Figure 63. 16-bit Multiplication, 32-bit Accumulated Result



Using Fractional Numbers

Unsigned 8-bit fractional numbers use a format where numbers in the range [0, 2> are allowed. Bits 6 - 0 represent the fraction and bit 7 represents the integer part (0 or 1), i.e. a 1.7 format. The FMUL instruction performs the same operation as the MUL instruction, except that the result is left-shifted 1 bit so that the high byte of the 2-byte result will have the same 1.7 format as the operands (instead of a 2.6 format). Note that if the product is equal to or higher than 2, the result will not be correct.

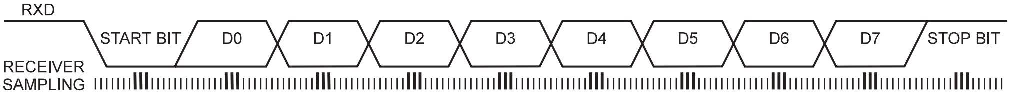
To fully understand the format of the fractional numbers, a comparison with the integer number format is useful: Table 20 illustrates the two 8-bit unsigned numbers formats. Signed fractional numbers, like signed integers, use the familiar two's complement format. Numbers in the range [-1, 1> may be represented using this format.

If the byte "1011 0010" is interpreted as an unsigned integer, it will be interpreted as 128 + 32 + 16 + 2 = 178. On the other hand, if it is interpreted as an unsigned fractional number, it will be interpreted as 1 + 0.25 + 0.125 + 0.015625 = 1.390625. If the byte is assumed to be a signed number, it will be interpreted as 178 - 256 = -122 (integer) or as 1.390625 - 2 = -0.609375 (fractional number).

```
   mulsu r23, r20             ; (signed)ah * bl
   sbc   r19, r2
   add   r17, r0
   adc   r18, r1
   adc   r19, r2

   mulsu r21, r22             ; (signed)bh * al
   sbc   r19, r2              ; Sign extend
   add   r17, r0
   adc   r18, r1
   adc   r19, r2

   ret

mac16x16_32_method_B:         ; uses two temporary registers (r4,r5), Speed / Size
                              Optimized
                              ; but reduces cycles/words by 1
   clr   r2

   muls  r23, r21             ; (signed)ah * (signed)bh
   movw  r5:r4,r1:r0

   mul   r22, r20             ; al * bl

   add   r16, r0
   adc   r17, r1
   adc   r18, r4
   adc   r19, r5

mulsu r23, r20                ; (signed)ah * bl
   sbc   r19, r2              ; Sign extend
   add   r17, r0
   adc   r18, r1
   adc   r19, r2

   mulsu r21, r22             ; (signed)bh * al
   sbc   r19, r2              ; Sign extend
   add   r17, r0
   adc   r18, r1
   adc   r19, r2

   ret
```

If however, a valid start bit is detected, sampling of the data bits following the start bit is performed. These bits are also sampled at samples 8, 9 and 10. The logical value found in at least two of the three samples is taken as the bit value. All bits are shifted into the transmitter shift register as they are sampled. Sampling of an incoming character is shown in Figure 66. Note that the description above is not valid when the UART transmission speed is doubled. See “Double Speed Transmission” on page 128 for a detailed description.

Figure 66. Sampling Received Data[1]



Note: 1. This figure is not valid when the UART speed is doubled. See “Double Speed Transmission” on page 128 for a detailed description.

When the stop bit enters the receiver, the majority of the three samples must be one to accept the stop bit. If two or more samples are logic 0s, the Framing Error (FEn) flag in the UART Control and Status Register (UCSRnA) is set. Before reading the UDRn register, the user should always check the FEn bit to detect Framing Errors.

Whether or not a valid stop bit is detected at the end of a character reception cycle, the data is transferred to UDRn and the RXCn flag in UCSRnA is set. UDRn is in fact two physically separate registers, one for transmitted data and one for received data. When UDRn is read, the Receive Data register is accessed, and when UDRn is written, the Transmit Data register is accessed. If the 9-bit data word is selected (the CHR9n bit in the UART Control and Status Register, UCSRnB is set), the RXB8n bit in UCSRnB is loaded with bit 9 in the Transmit shift register when data is transferred to UDRn.

If, after having received a character, the UDRn register has not been read since the last receive, the OverRun (ORn) flag in UCSRnB is set. This means that the last data byte shifted into to the shift register could not be transferred to UDRn and has been lost. The ORn bit is buffered, and is updated when the valid data byte in UDRn is read. Thus, the user should always check the ORn bit after reading the UDRn register in order to detect any overruns if the baud-rate is High or CPU load is High.

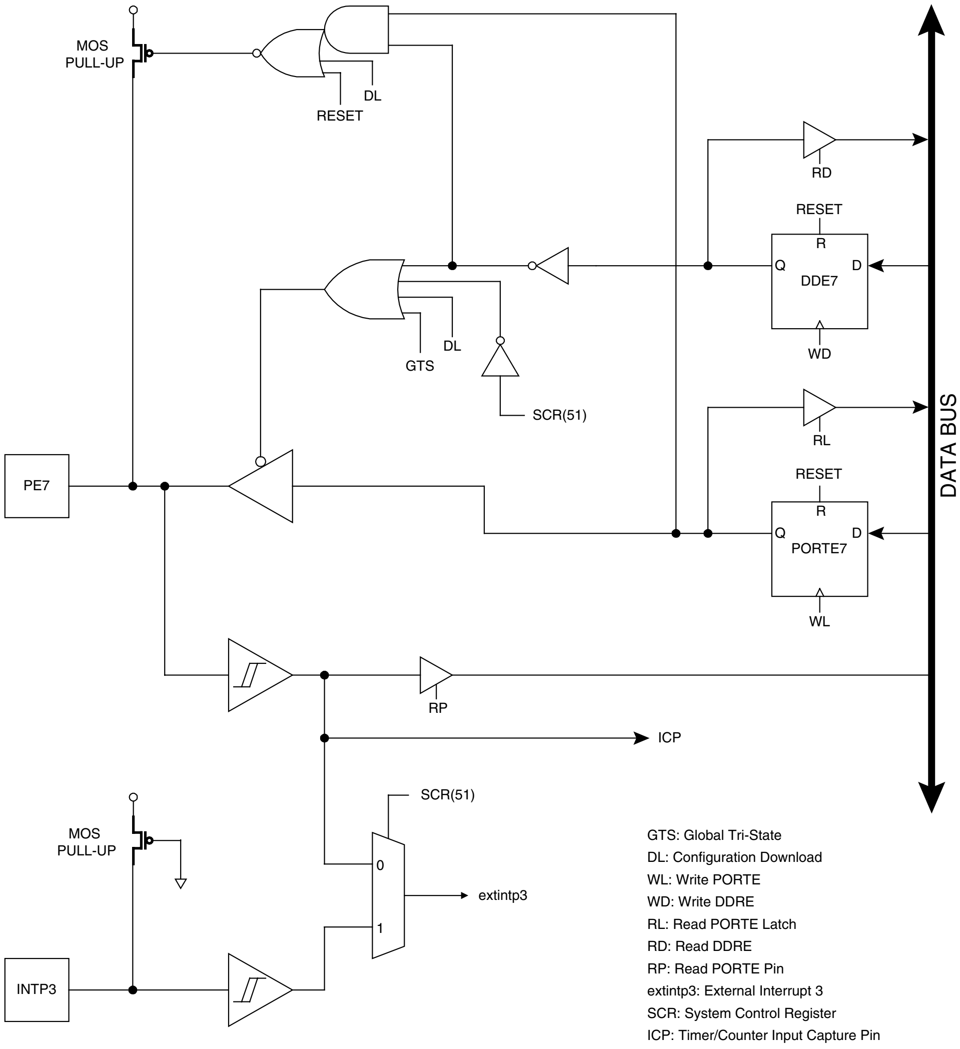
When the RXEN bit in the UCSRnB register is cleared (zero), the receiver is disabled. This means that the PE1 (n=0) or PE3 (n=1) pin can be used as a general I/O pin. When RXEN_n is set, the UART Receiver will be connected to PE1 (UART0) or PE3 (UART1), which is forced to be an input pin regardless of the setting of the DDE1 in DDRE (UART0) or DDB2 bit in DDRB (UART1). When PE1 (UART0) or PE3 (UART1) is forced to input by the UART, the PORTE1 (UART0) or PORTE3 (UART1) bit can still be used to control the pull-up resistor on the pin.

When the CHR9n bit in the UCSRnB register is set, transmitted and received characters are 9 bits long plus start and stop bits. The 9th data bit to be transmitted is the TXB8n bit in UCSRnB register. This bit must be set to the wanted value before a transmission is initiated by writing to the UDRn register. The 9th data bit received is the RXB8n bit in the UCSRnB register.

Table 37. UBR Settings at Various Crystal Frequencies in Double UART Speed Mode

Clock	UBRRHI		UBR		Actual	Desired	%	Clock	UBRRHI		UBR		Actual	Desired	%
MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error	MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error
1	0000	00110011	033	51	2404	2400	0.2	1.843	0000	01011111	05F	95	2400	2400	0.0
	0000	00011001	019	25	4808	4800	0.2		0000	00101111	02F	47	4800	4800	0.0
	0000	00001100	00C	12	9615	9600	0.2		0000	00010111	017	23	9600	9600	0.0
	0000	00001000	008	8	13889	14400	3.7		0000	00001111	00F	15	14400	14400	0.0
	0000	00000110	006	6	17857	19200	7.5		0000	00001011	00B	11	19200	19200	0.0
	0000	00000011	003	3	31250	28880	7.6		0000	00000111	007	7	28800	28880	0.3
	0000	00000010	002	2	41667	38400	7.8		0000	00000101	005	5	38400	38400	0.0
	0000	00000001	001	1	62500	57600	7.8		0000	00000011	003	3	57600	57600	0.0
	0000	00000001	001	1	62500	76800	22.9		0000	00000010	002	2	76800	76800	0.0
	0000	00000000	000	0	125000	115200	7.8		0000	00000001	001	1	115200	115200	0.0
Clock	UBRRHI		UBR		Actual	Desired	%	Clock	UBRRHI		UBR		Actual	Desired	%
MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error	MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error
9.216	0001	11011111	1DF	479	2400	2400	0.0	18.43	0011	10111111	3BF	959	2400	2400	0.0
	0000	11101111	0EF	239	4800	4800	0.0		0001	11011111	1DF	479	4800	4800	0.0
	0000	01110111	077	119	9600	9600	0.0		0000	11101111	0EF	239	9600	9600	0.0
	0000	01001111	04F	79	14400	14400	0.0		0000	10011111	09F	159	14400	14400	0.0
	0000	00111011	03B	59	19200	19200	0.0		0000	01110111	077	119	19200	19200	0.0
	0000	00100111	027	39	28800	28880	0.3		0000	01001111	04F	79	28800	28880	0.3
	0000	00011101	01D	29	38400	38400	0.0		0000	00111011	03B	59	38400	38400	0.0
	0000	00010011	013	19	57600	57600	0.0		0000	00100111	027	39	57600	57600	0.0
	0000	00001110	00E	14	76800	76800	0.0		0000	00011101	01D	29	76800	76800	0.0
	0000	00001001	009	9	115200	115200	0.0		0000	00010011	013	19	115200	115200	0.0
	0000	00000100	004	4	230400	230400	0.0		0000	00001001	009	9	230400	230400	0.0
	0000	00000010	002	2	384000	460800	20.0		0000	00000100	004	4	460800	460800	0.0
	0000	00000000	000	0	1152000	912600	20.8		0000	00000010	002	2	768000	912600	18.8
Clock	UBRRHI		UBR		Actual	Desired	%	Clock	UBRRHI		UBR		Actual	Desired	%
MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error	MHz	7:4 or 3:0	UBRRn	HEX	UBR	Freq	Freq.	Error
25.576	0101	00110011	533	1331	2400	2400	0.0	40	1000	00100010	822	2082	2400	2400	0.0
	0010	10011001	299	665	4800	4800	0.0		0100	00010001	411	1041	4798	4800	0.0
	0001	01001110	14E	334	9543	9600	0.6		0010	00001000	208	520	9597	9600	0.0
	0000	11011101	0DD	221	14401	14400	0.0		0001	01011010	15A	346	14409	14400	0.1
	0000	10100110	0A6	166	19144	19200	0.3		0001	00000011	103	259	19231	19200	0.2
	0000	01101110	06E	110	28802	28880	0.3		0000	10101100	0AC	172	28902	28880	0.1
	0000	01010010	052	82	38518	38400	0.3		0000	10000001	081	129	38462	38400	0.2
	0000	00110111	037	55	57089	57600	0.9		0000	01010110	056	86	57471	57600	0.2
	0000	00101001	029	41	76119	76800	0.9		0000	01000000	040	64	76923	76800	0.2
	0000	00011011	01B	27	114179	115200	0.9		0000	00101010	02A	42	116279	115200	0.9
	0000	00001101	00D	13	228357	230400	0.9		0000	00010101	015	21	227273	230400	1.4
	0000	00000110	006	6	456714	460800	0.9		0000	00001010	00A	10	454545	460800	1.4
	0000	00000011	003	3	799250	912600	14.2		0000	00000100	004	4	1000000	912600	8.7

Figure 82. PortE Schematic Diagram (Pin PE7)



GTS: Global Tri-State
DL: Configuration Download
WL: Write PORTE
WD: Write DDRE
RL: Read PORTE Latch
RD: Read DDRE
RP: Read PORTE Pin
extintp3: External Interrupt 3
SCR: System Control Register
ICP: Timer/Counter Input Capture Pin

External Clock Drive Waveforms

Figure 85. External Clock Drive Waveforms

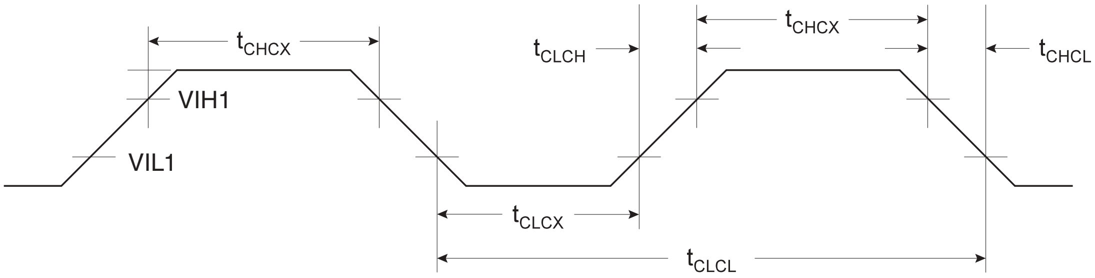


Table 53. External Clock Drive, V_{CC} = 3.0V to 3.6V

Symbol	Parameter	Minimum	Maximum	Units
$1/t_{CLCL}$	Oscillator Frequency	0	25	MHz
t_{CLCL}	Clock Period	40	–	ns
t_{CHCX}	High Time	15	–	ns
t_{CLCX}	Low Time	15	–	ns
t_{CLCH}	Rise Time	–	1.6	μs
t_{CHCL}	Fall Time	–	1.6	μs

AC Timing Characteristics – 3.3V Operation

Delays are based on fixed loads and are described in the notes.
Maximum times based on worst case: V_{CC} = 3.0V, temperature = 70°C
Minimum times based on best case: V_{CC} = 3.6V, temperature = 0°C
Maximum delays are the average of t_{PDLH} and t_{PDHL}.

All input IO characteristics measured from a V_{IH} of 50% of V_{DD} at the pad (CMOS threshold) to the internal V_{IH} of 50% of V_{DD}. All output IO characteristics are measured as the average of t_{PDLH} and t_{PDHL} to the pad V_{IH} of 50% of V_{DD}.

Cell Function	Parameter	Path	-25	Units	Notes
Repeaters					
Repeater	t_{PD} (Maximum)	L -> E	2.2	ns	1 Unit Load
Repeater	t_{PD} (Maximum)	E -> E	2.2	ns	1 Unit Load
Repeater	t_{PD} (Maximum)	L -> L	2.2	ns	1 Unit Load
Repeater	t_{PD} (Maximum)	E -> L	2.2	ns	1 Unit Load
Repeater	t_{PD} (Maximum)	E -> IO	1.4	ns	1 Unit Load
Repeater	t_{PD} (Maximum)	L -> IO	1.4	ns	1 Unit Load

All input IO characteristics measured from a V_{IH} of 50% of V_{DD} at the pad (CMOS threshold) to the internal V_{IH} of 50% of V_{DD}. All output IO characteristics are measured as the average of t_{PDLH} and t_{PDHL} to the pad V_{IH} of 50% of V_{DD}.

Cell Function	Parameter	Path	-25	Units	Notes
IO					
Input	t_{PD} (Maximum)	pad -> x/y	1.9	ns	No Extra Delay
Input	t_{PD} (Maximum)	pad -> x/y	5.8	ns	1 Extra Delay
Input	t_{PD} (Maximum)	pad -> x/y	11.5	ns	2 Extra Delays
Input	t_{PD} (Maximum)	pad -> x/y	17.4	ns	3 Extra Delays
Output, Slow	t_{PD} (Maximum)	x/y/E/L -> pad	9.1	ns	50 pf Load
Output, Medium	t_{PD} (Maximum)	x/y/E/L -> pad	7.6	ns	50 pf Load
Output, Fast	t_{PD} (Maximum)	x/y/E/L -> pad	6.2	ns	50 pf Load
Output, Slow	t_{PZX} (Maximum)	oe -> pad	9.5	ns	50 pf Load
Output, Slow	t_{PXZ} (Maximum)	oe -> pad	2.1	ns	50 pf Load
Output, Medium	t_{PZX} (Maximum)	oe -> pad	7.4	ns	50 pf Load
Output, Medium	t_{PXZ} (Maximum)	oe -> pad	2.7	ns	50 pf Load
Output, Fast	t_{PZX} (Maximum)	oe -> pad	5.9	ns	50 pf Load
Output, Fast	t_{PXZ} (Maximum)	oe -> pad	2.4	ns	50 pf Load

Packaging and Pin List Information

FPSLIC devices should be laid out to support a split power supply for both AL and AX families. Please refer to the “Designing in Split Power Supply Support for AT94KAL and AT94SAL Devices” application note, available on the Atmel web site.

Table 54. Part and Package Combinations Available

Part #	Package	AT94K05	AT94K10	AT94K40
PLCC 84	AJ	46	46	
TQ 100	AQ	58	58	
LQ144	BQ	82	84	84
PQ 208	DQ	96	116	120

Table 55. AT94K JTAG ICE Pin List

Pin	AT94K05 96 FPGA I/O	AT94K10 192 FPGA I/O	AT94K40 384 FPGA I/O
TDI	IO34	IO50	IO98
TDO	IO38	IO54	IO102
TMS	IO43	IO63	IO123
TCK	IO44	IO64	IO124

Table 56. AT94K Pin List

AT94K05 96 FPGA I/O	AT94K10 192 FPGA I/O	AT94K40 384 FPGA I/O	Packages			
			PC84	TQ100	PQ144	PQ208
West Side						
GND	GND	GND	12	1	1	2
I/O1, GCK1 (A16)	I/O1, GCK1 (A16)	I/O1, GCK1 (A16)	13	2	2	4
I/O2 (A17)	I/O2 (A17)	I/O2 (A17)	14	3	3	5
I/O3	I/O3	I/O3			4	6
I/O4	I/O4	I/O4			5	7
I/O5 (A18)	I/O5 (A18)	I/O5 (A18)	15	4	6	8
I/O6 (A19)	I/O6 (A19)	I/O6 (A19)	16	5	7	9
		GND				
		I/O7				
		I/O8				
		I/O9				

Notes: 1. VCC is I/O high voltage. Please refer to the “Designing in Split Power Supply Support for AT94KAL and AT94SAL Devices” application note.
2. VDD is core high voltage. Please refer to the “Designing in Split Power Supply Support for AT94KAL and AT94SAL Devices” application note.
3. Unbonded pins are No Connects.

Table 56. AT94K Pin List (Continued)

AT94K05 96 FPGA I/O	AT94K10 192 FPGA I/O	AT94K40 384 FPGA I/O	Packages			
			PC84	TQ100	PQ144	PQ208
I/O55	I/O83	I/O167			62	88
I/O56	I/O84	I/O168			63	89
GND	GND	GND			64	90
		I/O169				
		I/O170				
	I/O85	I/O171				
	I/O86	I/O172				
		I/O173				
		I/O174				
		GND				
		I/O175				
		I/O176				
	I/O87	I/O177				91
	I/O88	I/O178				92
I/O57	I/O89	I/O179				93
I/O58	I/O90	I/O180				94
		GND				
		VCC[1]				
		I/O181				
		I/O182				
I/O59 (TD2)	I/O91 (TD2)	I/O183 (TD2)	48	45	65	95
I/O60 (TD1)	I/O92 (TD1)	I/O184 (TD1)	49	46	66	96
		I/O185				
		I/O186				
		GND				
		I/O187				
		I/O188				
I/O61	I/O93	I/O189			67	97
I/O62	I/O94	I/O190			68	98
I/O63 (TD0)	I/O95 (TD0)	I/O191 (TD0)	50	47	69	99
I/O64, GCK4	I/O96, GCK4	I/O192, GCK4	51	48	70	100
GND	GND	GND	52	49	71	101
$\overline{\text{CON}}$	$\overline{\text{CON}}$	$\overline{\text{CON}}$	53	50	72	103
East Side						

Notes: 1. VCC is I/O high voltage. Please refer to the “Designing in Split Power Supply Support for AT94KAL and AT94SAL Devices” application note.

2. VDD is core high voltage. Please refer to the “Designing in Split Power Supply Support for AT94KAL and AT94SAL Devices” application note.

3. Unbonded pins are No Connects.