Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

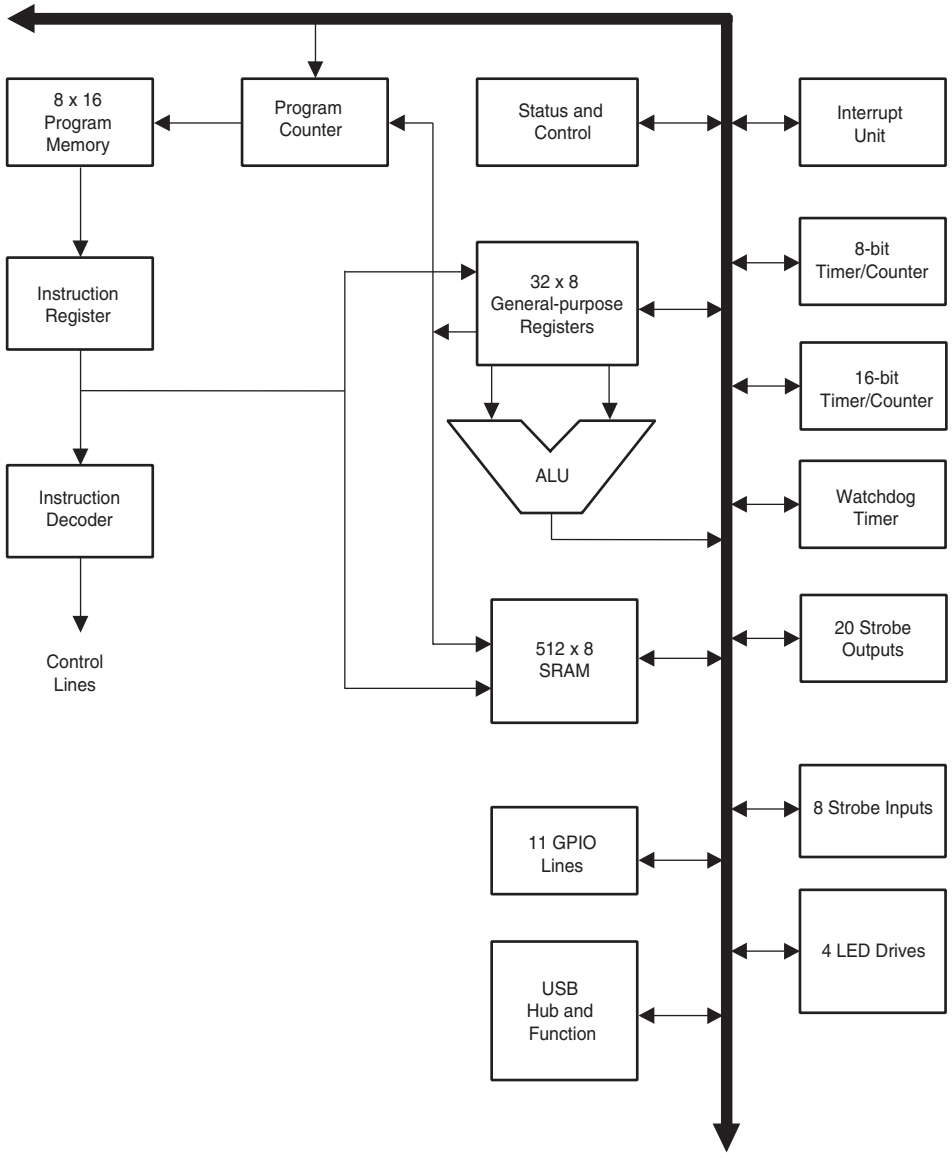
Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Obsolete
Applications	Keyboard Controller
Core Processor	AVR
Program Memory Type	SRAM (16kB)
Controller Series	AT43USB
RAM Size	512 x 8
Interface	SPI, 3-Wire Serial
Number of I/O	42
Voltage - Supply	4.4V ~ 5.25V
Operating Temperature	0°C ~ 70°C
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at43usb325e-ac

Figure 1-3. AT43USB325 Enhanced RISC Architecture with USB Keyboard Controller and Hub



The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, Timer/Counters, and other I/O functions. The I/O Memory can be accessed directly, or as the Data Space locations following those of the register file, $20 - $5F.

The AVR uses a Harvard architecture concept – with separate memories and buses for program and data. The program memory is executed with a single-level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is a downloadable SRAM or a mask programmed ROM.

With the relative jump and call instructions, the whole 24K address space is directly accessed. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16- or 32-bit instruction.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the stack. The stack is effectively allocated in the general data SRAM, and consequently, the stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the Stack Pointer (SP) in the reset routine (before subroutines or interrupts are executed). The 10-bit SP is read/write accessible in the I/O space.

The 512-byte data SRAM can be easily accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps. A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the status register. All interrupts have a separate interrupt vector in the interrupt vector table at the beginning of the program memory. The interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address, the higher the priority.

3. General-purpose Register File

Table 3-1. AVR CPU General-purpose Working Register

Register	Address	Comment
R0	$00	
R1	$01	
R2	$02	
..		
R13	$0D	
R14	$0E	
R15	$0F	
R16	$10	
R17	$11	
..		
R26	$1A	X-register low byte
R27	$1B	X-register high byte
R28	$1C	Y-register low byte
R29	$1D	Y-register high byte
R30	$1E	Z-register low byte
R31	$1F	Z-register high byte

All register operating instructions in the instruction set have direct and single cycle access to all registers. The only exception is the five constant arithmetic and logic instructions SBCI, SUBI, CPI, ANDI, and ORI between a constant and a register, and the LDI instruction for load immediate constant data. These instructions apply to the second half of the registers in the register file – R16..R31. The general SBC, SUB, CP, AND, and OR and all other operations between two registers or on a single register apply to the entire register file.

As shown in Table 3-1, each register is also assigned a data memory address, mapping them directly into the first 32 locations of the user Data Space. Although not being physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X-, Y-, and Z-registers can be set to index any register in the file.

Table 3-4. USB Hub and Function Registers

Name	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GLB_STATE	$1FFB	–	KB INT EN	–	SUSP FLG	RESUME FLG	RMWUPE	CONFG	HADD EN
SPRSR	$1FFA	INTD	INTC	INTB	INTA	-	FRWUP	RSM	GLB SUSP
SPRSIE	$1FF9	INTD EN	INTC EN	INTB EN	INTA EN	-	FRWUP IE	RSM IE	GLB SUSP IE
SPRSMSK	$1FF8	INTD MSK	INTC MSK	INTB MSK	INTA MSK	-	FRWUP MSK	RSM MSK	GLB SUSP MSK
UISR	$1FF7	SOF INT	EOF2 INT	-	FEP3 INT	HEP0 INT	FEP2 INT	FEP1 INT	FEP0 INT
UIMSKR	$1FF6	SOF MSK	SOF2 MSK	-	FEP3 MSK	HEP0 MSK	FEP2 MSK	FEP1 MSK	FEP0 MSK
UIAR	$1FF5	SOF INTACK	EOF2 INTACK	-	FEP3 INTACK	HEP0 INTACK	FEP2 INTACK	FEP1 INTACK	FEP0 INTACK
UIER	$1FF3	SOF IE	EOF2 IE	-	FEP3 IE	HEP0 IE	FEP2 IE	FEP1 IE	FEP0 IE
UOVCER	$1FF2	–	–	–	–	–	OVC	–	–
ISCR	$1FF1	ISC71	ISC70	ISC61	ISC60	ISC51	ISC50	ISC41	ISC40
HADDR	$1FEF	SAEN	HADD6	HADD5	HADD4	HADD3	HADD2	HADD1	HADD0
FADDR	$1FEE	FEN	FADD6	FADD5	FADD4	FADD3	FADD2	FADD1	FADD0
HENDP0_CNTR	$1FE7	EPEN	–	–	–	DTGLE	EPDIR	EPTYPE1	EPTYPE0
FENDP0_CNTR	$1FE5	EPEN	–	–	–	DTGLE	EPDIR	EPTYPE1	EPTYPE0
FENDP1_CNTR	$1FE4	EPEN	–	–	–	DTGLE	EPDIR	EPTYPE1	EPTYPE0
FENDP2_CNTR	$1FE3	EPEN	–	–	–	DTGLE	EPDIR	EPTYPE1	EPTYPE0
FENDP3_CNTR	$1FE2	EPEN	-	-	-	DTGLE	EPDIR	EPTYPE1	EPTYPE0
HCSR0	$1FDF	–	–	–	–	STALL SENT	RX SETUP	RX OUT PACKET	TX CEMPLETE
FCSR0	$1FDD	–	–	–	–	STALL SENT	RX SETUP	RX OUT PACKET	TX COMPLETE
FCSR1	$1FDC	–	–	–	–	STALL SENT	RX SETUP	RX OUT PACKET	TX COMPLETE
FCSR2	$1FDB	–	–	–	–	STALL SENT	RX SETUP	RX OUT PACKET	TX COMPLETE
FCSR3	$1FDA	-	-	-	-	STALL SENT	-	RX OUT PACKET	TX CMPLETE
HDR0	$1FD7	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FDR0	$1FD5	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FDR1	$1FD4	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FDR2	$1FD3	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
FDR3	$1FD2	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
HBYTE_CNT0	$1FCF	–	–	BYTCT5	BYTCT4	BYTCT3	BYTCT2	BYTCT1	BYTCT0
FBYTE_CNT0	$1FCD	–	–	BYTCT5	BYTCT4	BYTCT3	BYTCT2	BYTCT1	BYTCT0
FBYTE_CNT1	$1FCC	–	–	BYTCT5	BYTCT4	BYTCT3	BYTCT2	BYTCT1	BYTCT0
FBYTE_CNT2	$1FCB	–	–	BYTCT5	BYTCT4	BYTCT3	BYTCT2	BYTCT1	BYTCT0
FBYTE_CNT3	$1FCA	-	-	-	BYTCT4	BYTCT3	BYTCT2	BYTCT1	BYTCT0
HSTR	$1FC7	–	–	–	–	OVLSC	LPSC	OVI	LPS
HPCON	$1FC5	–	HPCON2	HPCON1	HPCON0	–	HPADD2	HPADD1	HPADD0
HPSTAT5	$1FBC	-	LSP	PPSTAT	PRSTAT	POCI	PSSTAT	PESTAT	PCSTAT
HPSTAT4	$1FBB	-	LSP	PPSTAT	PRSTAT	POCI	PSSTAT	PESTAT	PCSTAT
HPSTAT3	$1FBA	–	LSP	PPSTAT	PRSTAT	POCI	PSSTAT	PESTAT	PCSTAT
HPSTAT2	$1FB9	–	LSP	PPSTAT	PRSTAT	POCI	PSSTAT	PESTAT	PCSTAT
HPSTAT1	$1FB8	–	LSP	PPSTAT	PRSTAT	POCI	PSSTAT	PESTAT	PCSTAT
HPSCR5	$1FB4	-	-	-	RSTSC	POCIC	PSSC	PESC	PCSC
HPSCR4	$1FB3	-	-	-	RSTSC	POCIC	PSSC	PESC	PCSC
HPSCR3	$1FB2	–	–	–	RSTSC	POCIC	PSSC	PESC	PCSC
HPSCR2	$1FB1	–	–	–	RSTSC	POCIC	PSSC	PESC	PCSC
HPSCR1	$1FB0	–	–	–	RSTSC	POCIC	PSSC	PESC	PCSC
PSTAT5	$1FAC	–	–	–	–	–	–	DPSTATE	DMSTATE
PSTAT4	$1FAB	–	–	–	–	–	–	DPSTATE	DMSTATE
PSTAT3	$1FAA	–	–	–	–	–	–	DPSTATE	DMSTATE
PSTAT2	$1FA9	–	–	–	–	–	–	DPSTATE	DMSTATE
HCAR0	$1FA7	CTL DIR	DATA END	FORCE STALL	TX PACKET READY	STALL_SENT-ACK	RX_SETUP_ACK	RX_OUT_PACKET_ACK	TX_COMPLETE-ACK
FCAR0	$1FA5	CTL DIR	DATA END	FORCE STALL	TX PACKET READY	STALL_SENT-ACK	RX_SETUP_ACK	RX_OUT_PACKET_ACK	TX_COMPLETE-ACK

4. Functional Description

4.1 On-chip Power Supply

The AT43USB325 contains two on-chip power supplies that generate 3.3V with a capacity of 30 mA each from the 5V power input. The on-chip power supplies are intended to supply the AT43USB325 internal circuit and the 1.5K pull-up resistor only and should not be used for other purposes. External 2.2 µF filter capacitors are required at the power supply outputs, CEXT1 and CEXT2. The internal power supplies can be disabled as described in the next paragraph.

The user should be careful when the GPIO pins are required to supply high-load currents. If the application requires that the GPIO supply currents beyond the capability of the on-chip power supply, the AT43USB325 should be supplied by an external 3.3V power supply. In this case, the 5V V_{CC} power supply pin should be left unconnected and the 3.3V power supplied to the chip through the CEXT1 and CEXT2 pins.

4.2 I/O Pin Characteristics

The I/O pins of the AT43USB325 should not be directly connected to voltages less than V_{SS} or more than the voltage at the CEXT pins. If it is necessary to violate this rule, insert a series resistor between the I/O pin and the source of the external signal source that limits the current into the I/O pin to less than 2 mA. Under no circumstance should the external voltage exceed 5.5V. To do so will put the chip under excessive stress.

4.3 Oscillator and PLL

All clock signals required to operate the AT43USB325 are derived from an on-chip oscillator. To reduce EMI and power dissipation, the oscillator is designed to operate with a 6 MHz crystal. An on-chip PLL generates the high frequency for the clock/data separator of the Serial Interface Engine. In the suspended state, the oscillator circuitry is turned off.

The oscillator of the AT43USB325 is a special, low-drive type, designed to work with most crystals without any external components. The crystal must be of the parallel resonance type requiring a load capacitance of about 10 pF. If the crystal requires a higher value capacitance, external capacitors can be added to the two terminals of the crystal and ground to meet the required value. To assure quick start-up, a crystal with a high Q, or low ESR, should be used. To meet the USB hub frequency accuracy and stability requirements for hubs, the crystal should have an accuracy and stability of better than 100 PPM. The use of a ceramic resonator in place of the crystal is not recommended because a resonator would not have the necessary frequency accuracy and stability.

The clock can also be externally sourced. In this case, connect the clock source to the XTAL1 pin, while leaving XTAL2 pin floating. The switching level at the OSC1 pin can be as low as 0.47V and a CMOS device is required to drive this pin to maintain good noise margins at the low switching level.

For proper operation of the PLL, an external RC filter consisting of a series RC network of 100Ω and 0.1 µF in parallel with a 0.01 µF capacitor must be connected from the LFT pin to V_{SS}. Use only high-quality ceramic capacitors.

4.12 USB Interrupt Sources

The USB interrupts are described below.

Table 4-3. USB Interrupt Sources

Interrupt	Description
SOF Received	Whenever USB hardware decodes a valid Start of Frame. The frame number is stored in the two Frame Number Registers.
EOF2	Activated whenever the hub's frame timer reaches its EOF2 time point.
Function EP0 Interrupt	See “Control Transfers at Control End-point EP0” on page 64 for details.
Function EP1 Interrupt	For an OUT endpoint it indicates that Function Endpoint 1 has received a valid OUT packet and that the data is in the FIFO. For an IN endpoint it means that the endpoint has received an IN token, sent out the data in the FIFO and received an ACK from the Host. The FIFO is now ready to be written by new data from the microcontroller.
Function EP2 Interrupt	For an OUT endpoint it indicates that Function Endpoint 2 has received a valid OUT packet and that the data is in the FIFO. For an IN endpoint it means that the endpoint has received an IN token, sent out the data in the FIFO and received an ACK from the Host. The FIFO is now ready to be written by new data from the microcontroller.
Function EP3 Interrupt	For an OUT end-point it indicates that Function End-point 3 has received a valid OUT packet and that the data is in the FIFO. For an IN end-point it means that the end-point has received an IN token, sent out the data in the FIFO and received an ACK from the Host. The FIFO is now ready to be written by new data from the microcontroller.
Hub EP0 Interrupt	See “Control Transfers at Control End-point EP0” on page 64 for details.
FRWUP	USB hardware has received a embedded function remote wakeup request.
GLB SUSP	USB hardware has received global suspend signaling and is preparing to put the hub in the suspend mode. The microcontroller's firmware should place the embedded function in the suspend state.
RSM	USB hardware received resume signaling and is propagating the resume signaling. The microcontroller's firmware should take the embedded function out of the suspended state.
BUS RESET	USB hardware received a USB bus reset. This applies only in cases where a separation between USB bus reset and microcontroller reset is required. Be very careful when using this feature.

All interrupts have individual enable, status, and mask bits through the interrupt enable register and interrupt mask register. The Suspend and Resume interrupts are cleared by writing a 0 to the particular interrupt bit. All other interrupts are cleared when the microcontroller sets a bit in an interrupt acknowledge register.

4.13.2 USB Interrupt Mask Register – UIMSKR

Bit	7	6	5	4	3	2	1	0	
$1FF6	SOF IMSK	EOF2 IMSK	–	FEP3 IMSK	HEP0 IMSK	FEP2 IMSK	FEP1 IMSK	FEP0 IMSK	UIMSKR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – SOF IMSK: Enable Start of Frame Interrupt Mask**

When the SOF IMSK bit is set (1), the Start of Frame Interrupt is masked.

- **Bit 6 – EOF2 IMSK: Enable EOF2 Interrupt**

When the EOF2 IMSK bit is set (1), the EOF2 Interrupt is masked.

- **Bit 5 – Res: Reserved Bit**

This bit is reserved and always read as zero.

- **Bit 4 – FEP3 IMSK: Function End-point 3 Interrupt Mask**

When the FE3 IMSK bit is set (1), the Function End-point 3 Interrupt is masked.

- **Bit 3 – HEP0 IMSK: Enable Endpoint 0 Interrupt**

When the HEP0 IMSK bit is set (1), the Hub Endpoint 0 Interrupt is masked.

- **Bit 2 – FEP2 IMSK: Enable Endpoint 2 Interrupt**

When the FE2 IMSK bit is set (1), the Function Endpoint 2 Interrupt is masked.

- **Bit 1 – FEP1 IMSK: Enable Endpoint 1 Interrupt**

When the FE1 IMSK bit is set (1), the Function Endpoint 1 Interrupt is masked.

- **Bit 0 – FEP0 IMSK: Enable Endpoint 0 Interrupt**

When the FE0 IMSK bit is set (1), the Function Endpoint 0 Interrupt is masked.

4.13.4 USB Interrupt Enable Register – UIER

Bit	7	6	5	4	3	2	1	0	
$1FF3	SOF IE	EOF2 IE	–	FEP3 IE	HEP0 IE	FEP2 IE	FEP1 IE	FEP0 IE	UIER
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – SOF IE: Enable Start of Frame Interrupt**

When the SOF IE bit is set (1), the Start of Frame Interrupt is enabled.

- **Bit 6 – EOF2 IE: Enable EOF2 Interrupt**

When the EOF2 IE bit is set (1), the EOF2 Interrupt is enabled.

- **Bit 5 – Res: Reserved bit**

This bit is reserved and always read as zero.

- **Bit 4 – FEP3 IE: Enable Function End-point 3 Interrupt**

When the FE3 IE bit is set (1), the Function End-point 3 Interrupt is enabled.

- **Bit 3 – HEP0 IE: Enable Endpoint 0 Interrupt**

When the HEP0 IE bit is set (1), the Hub Endpoint 0 Interrupt is enabled.

- **Bit 2 – FEP2 IE: Enable Endpoint 2 Interrupt**

When the FE2 IE bit is set (1), the Function Endpoint 2 Interrupt is enabled.

- **Bit 1 – FEP1 IE: Enable Endpoint 1 Interrupt**

When the FE1 IE bit is set (1), the Function Endpoint 1 Interrupt is enabled.

- **Bit 0 – FEP0 IE: Enable Endpoint 0 Interrupt**

When the FE0 IE bit is set (1), the Function Endpoint 0 Interrupt is enabled.

6.4.6 Timer/Counter1 Input Capture Register – ICR1H and ICR1L

Bit	15	14	13	12	11	10	9	8	
$25 ($45)	MSB	–	–	–	–	–	–	–	ICR1H
$24 ($44)	–	–	–	–	–	–	–	LSB	ICR1L
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

The input capture register is a 16-bit read-only register.

When the rising or falling edge (according to the input capture edge setting - ICES1) of the signal at the input capture pin (ICP) is detected, the current value of the Timer/Counter1 is transferred to the Input Capture Register (ICR1). At the same time, the Input Capture Flag (ICF1) is set (one).

Since the ICR1 is a 16-bit register, a temporary register TEMP is used when ICR1 is read to ensure that both bytes are read simultaneously. When the CPU reads the low byte ICR1L, the data is sent to the CPU and the data of the high byte ICR1H is placed in the TEMP register. When the CPU reads the data in the high byte ICR1H, the CPU receives the data in the TEMP register. Consequently, the low byte ICR1L must be accessed first for a full 16-bit register read operation.

The TEMP register is also used when accessing TCNT1, OCR1A and OCR1B. If the main program and also interrupt routines perform access to registers using TEMP, interrupts must be disabled during access from the main program and from interrupt routines, if interrupts are allowed from within interrupt routines.

6.4.7 Timer/Counter1 In PWM Mode

When the PWM mode is selected, Timer/Counter1, the Output Compare Register1A (OCR1A) and the Output Compare Register1B (OCR1B) form a dual 8-, 9- or 10-bit, free-running, glitch-free and phase correct PWM with outputs on the PD5 (OC1A) and OC1B pins. Timer/Counter1 acts as an up/down counter, counting up from $0000 to TOP (see Table 6-5), where it turns and counts down again to zero before the cycle is repeated. When the counter value matches the contents of the 10 least significant bits of OCR1A or OCR1B, the PD5(OC1A)/OC1B pins are set or cleared according to the settings of the COM1A1/COM1A0 or COM1B1/COM1B0 bits in the Timer/Counter1 Control Register TCCR1A. Refer to Table 6-6 for details.

Table 6-5. Timer TOP Values and PWM Frequency

PWM Resolution	Timer TOP value	Frequency
8-bit	$00FF (255)	$f_{TCK1}/510$
9-bit	$01FF (511)	$f_{TCK1}/1022$
10-bit	$03FF(1023)	$f_{TCK1}/2046$

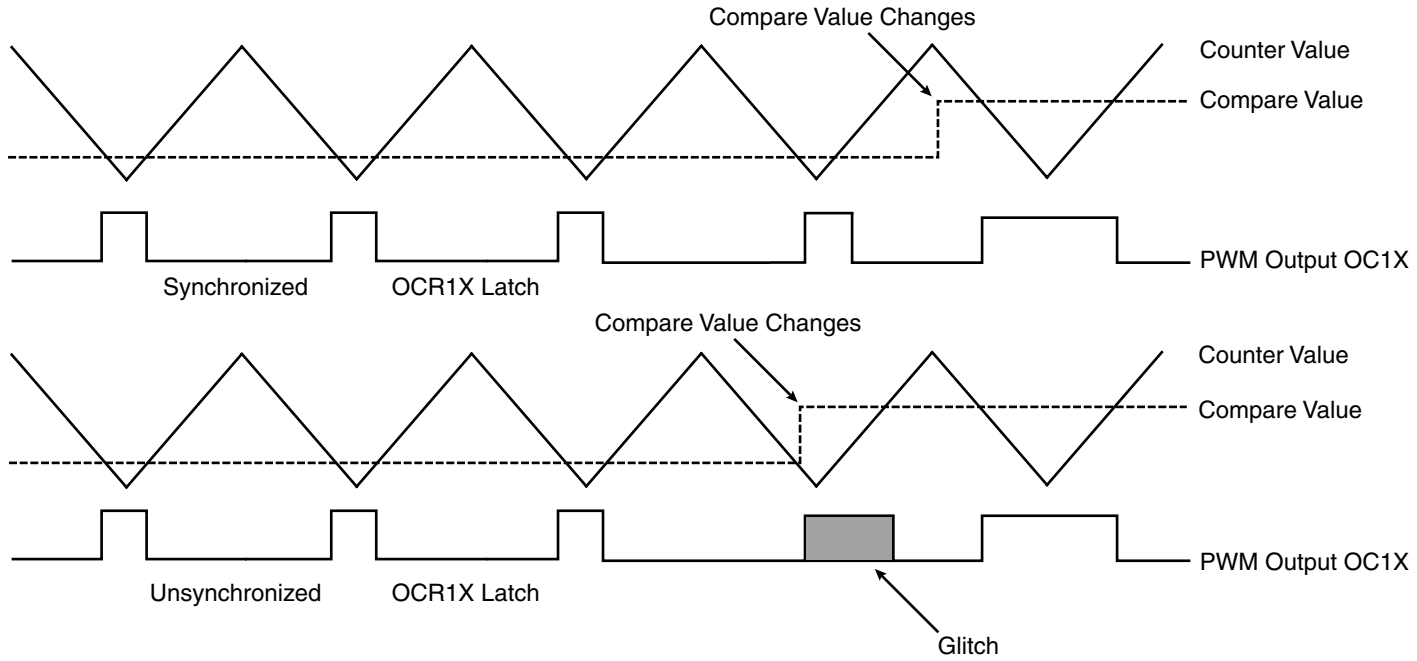
Table 6-6. Compare1 Mode Select in PWM Mode

COM1X1	COM1X0	Effect on OCX1
0	0	Not connected
0	1	Not connected
1	0	Cleared on compare match, up-counting. Set on compare match, down-counting (non-inverted PWM).
1	1	Cleared on compare match, down-counting. Set on compare match, up-counting (inverted PWM).

Note: X = A or B

Note that in the PWM mode, the 10 least significant OCR1A/OCR1B bits, when written, are transferred to a temporary location. They are latched when Timer/Counter1 reaches the value TOP. This prevents the occurrence of odd-length PWM pulses (glitches) in the event of an unsynchronized OCR1A/OCR1B write. See Figure 6-5 for an example.

Figure 6-5. Effects on Unsynchronized OCR1 Latching



Note: X = A or B

During the time between the write and the latch operation, a read from OCR1A or OCR1B will read the contents of the temporary location. This means that the most recently written value always will read out of OCR1A/B

When the OCR1 contains $0000 or TOP, the output OC1A/OC1B is updated to low or high on the next compare match, according to the settings of COM1A1/COM1A0 or COM1B1/COM1B0. This is shown in Table 6-7.

Note: If the compare register contains the TOP value and the prescaler is not in use (CS12..CS10 = 001), the PWM output will not produce any pulse at all, because up-counting and down-counting values are reached simultaneously. When the prescaler is in use (CS12..CS10 = 001 or 000), the PWM output goes active when the counter reaches the TOP value, but the

7.2 Port B

Port B is an 8-bit bi-directional I/O port with open drain outputs and controlled slew rate. It is designed for use as the column driver in a keyboard controller. The Port B output buffers can sink or source 4 mA.

Three I/O memory address locations are allocated for the Port B, one each for the Data Register - PORTB, $18($38), Data Direction Register (DDRB), $17($37) and the Port B Input Pins (PINB), $16($36). The Port B Input Pins address is read only, while the Data Register and the Data Direction Register are read/write.

7.2.1 Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
$18 ($38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
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	0	0	0	

7.2.2 Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
$17 ($37)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
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	0	0	0	

7.2.3 Port B Input Pins Address – PINB

Bit	7	6	5	4	3	2	1	0	
$16 ($36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The Port B Input Pins address (PINB) is not a register, and this address enables access to the physical value on each Port B pin. When reading PORTB, the Port B Data Latch is read, and when reading PINB, the logical values present on the pins are read.

7.3 Port C

Port C is an 8-bit bi-directional I/O port with an internal pull-up resistor at each pin. Port C is designed for use as the row inputs of a keyboard controller. Its output buffers can sink 4 mA.

Three I/O memory address locations are allocated for the Port C, one each for the Data Register - PORTC, $15($35), Data Direction Register - DDRC, $14($34) and the Port C Input Pins – PINC, $13($33). The Port C Input Pin's address is read only, while the Data Register and the Data Direction Register are read/write.

7.3.1 Port C Data Register – PORTC

Bit	7	6	5	4	3	2	1	0	
$15 ($35)	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	PORTC
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	0	0	0	

7.3.2 Port C Data Direction Register – DDRC

Bit	7	6	5	4	3	2	1	0	
$14 ($34)	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	DDRC
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	0	0	0	

7.3.3 Port C Input Pins Address – PINC

Bit	7	6	5	4	3	2	1	0	
$13 ($33)	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	PINC
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The Port C Input Pins address (PINC) is not a register, and this address enables access to the physical value on each Port C pin. When reading PORTC, the Port C Data Latch is read, and when reading PINC, the logical values present on the pins are read.

7.4.4 Port D as General Digital I/O

PDn, General I/O Pin: The DDDn bit in the DDRD register selects the direction of this pin. If DDDn is set (one), PDn is con-figured as an output pin. If DDDn is cleared (zero), PDn is configured as an input pin. The value of PORTDn has no meaning in this mode. The Port D pins are tri-stated when a reset condition becomes active.

Table 7-3. DDDn Bits on Port D Pins

DDDn	PORTDn	I/O	Comment
0	0	Input	Tri-state (Hi-Z)
0	1	Input	Tri-state (Hi-Z)
1	0	Output	Push-pull Zero Output
1	1	Output	Push-pull One Output

Note: n: 7,6,5,4,3,1,0, pin number

7.4.5 Alternate Functions of Port D

INT1, External Interrupt Source 1: The PD3 pin can serve as an external interrupt source to the MCU. See the interrupt description for further details, and how to enable the source.

7.5 Port E

Port E[0:3] each is a bi-directional I/O port with open drain outputs and controlled slew rate and is designed for use as the column drivers in a keyboard controller. The Port E[0:3] output buffers can sink 4 mA. Port E[4:7] are bi-directional I/O with outputs capable of driving LEDs directly. Each pin of Port E[4:7] has a series resistor to limit the LEDs current.

Three I/O memory address locations are allocated for the Port E, one each for the Data Register - PORTE, $03($23), Data Direction Register - DDRE, $02($22) and the Port E Input Pins - PINE, $01($21). The Port E Input Pins address is read only, while the Data Register and the Data Direction Register are read/write.

7.5.1 Port E Data Register – PORTE

Bit	7	6	5	4	3	2	1	0	
$03($23)	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	PORTE
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	0	0	0	

7.5.2 Port E Data Direction Register – DDRE

Bit	7	6	5	4	3	2	1	0	
$02 ($22)	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	DDRE
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	0	0	0	

8.1.3 Setup Response State

The Function Interface Unit (FIU) receives a SETUP token with 8 bytes of data from the Host. The FIU stores the data in the FIFO, sends an ACK back to the host and asserts an RX_SETUP interrupt.

Hardware

1. SETUP token, DATA from Host
2. ACK to Host
3. Store data in FIFO
4. Set RX SETUP → INT

Firmware

5. Read UISR
6. Read CSR0
7. Read Byte Count
8. Read FIFO
9. Parse command data
10. Write to H/FCAR0:
11. If Control Read: set DIR, clear RX SETUP, fill FIFO, set TX Packet Ready in CAR0
12. If Control Write: clear DIR in CAR0
13. If no Data Stage: set Data End, clear DIR, set Force STALL in CAR0
14. Set UIAR[EP0 INTACK] to clear the interrupt source

The microcontroller should write into the FIFO only if this bit is cleared. After it has completed writing the data, it should set this bit. This data can be of zero length.

Hardware clears this bit after it receives an ACK. If the interrupt is enabled and if the TX Complete bit is set, clearing the TX Packet Ready bit by the hardware causes an interrupt to the microcontroller.

- **Bit 3 – STALL_SENT_ACK: Acknowledge Stall Sent Interrupt**

Firmware sets this bit to clear STALL SENT, CSR bit 3. The 1 written in the CSRACK3 bit is not actually stored and thus does not have to be cleared.

- **Bit 2 – RX_SETUP_ACK: Acknowledge RX SETUP Interrupt**

Firmware sets this bit to clear RX SETUP, CSR bit2. The 1 written in the CSRACK2 bit is not actually stored and thus does not have to be cleared.

- **Bit 1 – RX_OUT_PACKET_ACK: Acknowledge RX OUT PACKET Interrupt**

Firmware sets this bit to clear RX OUT PACKET, CSR bit1. The 1 written in the CSRACK1 bit is not actually stored and thus does not have to be cleared.

- **Bit 0 – TX_COMPLETE_ACK: Acknowledge TX COMPLETE Interrupt**

Firmware sets this bit to clear TX COMPLETE, CSR bit0. The 1 written in the CSRACK0 bit is not actually stored and thus does not have to be cleared.

8.3.12 Function Endpoint 1,2,3 Service Routine Register – FCSR1,2,3

Bit	7	6	5	4	3	2	1	0	
Function EP1 $1FDC	–	–	–	–	STALL SENT	–	RX OUT PACKET	TX COMPLETE	FCSR1
Function EP2 $1FDB	–	–	–	–	STALL SENT	–	RX OUT PACKET	TX COMPLETE	FCSR2
Function EP3 $1FDA	–	–	–	–	STALL SENT	–	RX OUT PACKET	TX COMPLETE	FCSR3
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7..4 – Reserved**

These bits are reserved in the AT43USB325 and will read as zero.

- **Bit 3 – STALL SENT**

The USB hardware sets this bit after a STALL has been sent to the host. The firmware uses this bit when responding to a Get Status[Endpoint] request. It is a read only bit and that is cleared indirectly by writing a one to the STALL_SENT_ACK bit of the Control and Acknowledge Register.

- **Bit 2 – Reserved**

This bit is reserved in the AT43USB325 and will read as zero.

- **Bit 1 – RX OUT PACKET**

The USB hardware sets this bit after it has stored the data of an OUT transaction in the FIFO. While this bit is set, the hardware will NAK all OUT tokens. The USB hardware will not overwrite the data in the FIFO except for an early set-up. RX OUT Packet is used by a BULK OUT or ISO OUT or INT OUT endpoint.

Setting this bit causes an interrupt to the microcontroller if the interrupt is enabled. FW clears this bit after the FIFO contents have been read by writing a one to the RX_SETUP_ACK bit of the Control and Acknowledge Register.

- **Bit 0 – TX COMPLETE: Transmit Completed**

This bit is used by the endpoint hardware to signal to the microcontroller that the IN transaction was completed successfully. This bit is read only and is cleared indirectly by writing a one to the TX_COMPLETE_ACK bit of the Control and Acknowledge Register.

8.4.2.2 Hub Port Control Register – HPCON

Bit	7	6	5	4	3	2	1	0	
$1FC5	–	HPCON2	HPCON1	HPCON0	–	HPADD2	HPADD1	HPADD0	HPCON
Read/Write	R	R/W	R/W	R/W	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – Reserved**

This bits is reserved in the AT43USB325 and will read as zero.

- **Bit 6..4 – HPCON2..0: Hub Port Control Command**

These bits are written by firmware to control the port states upon receipt of a Host request.

Bit6	Bit5	Bit4	Action
0	0	0	Disable port
0	0	1	Enable port
0	1	0	Reset and enable port
0	1	1	Suspend port
1	0	0	Resume port

Disable Port = ClearPortFeature(PORT_ENABLE)

Action: USB hardware places addressed port in disabled state. Port 1 is placed in disabled state by firmware.

Enable Port = SetPort Feature(PORT_ENABLE)

Action: USB hardware places addressed port in enabled state. Firmware is responsible for placing Port 1 in enabled state.

Reset and Enable Port = SetPort Feature(PORT_RESET)

Action: USB hardware drives reset signaling through addressed port. USB hardware and firmware resets their embedded function registers to the default state.

Suspend Port = SetPortFeature(PORT_SUSPEND)

Action: USB hardware places port in idle state and stops propagating traffic through the addressed port. Firmware places Port 1 in suspend state by disabling its endpoints and placing the peripheral function in its low power state.

Resume Port = ClearPortFeature(PORT_SUSPEND)

Action: USB hardware sends resume signaling to addressed port and then enables port. Firmware takes the embedded function out of the suspend state and enables Port 1's endpoints.

- **Bit 3 – Reserved**

This bits is reserved in the AT43USB325 and will read as zero.

- **Bit 2..0 – HPCON2..0: Hub Port Address**

These bits define which port is being addressed for the command defined by bits [2:0].

Bit2	Bit1	Bit0	Port addresses
1	0	1	Port5
1	0	0	Port4
0	1	1	Port3
0	1	0	Port2

8.4.3 Selective Suspend and Resume

The host can selectively suspend and resume a port through the Set Port Feature (PORT_SUSPEND) and Clear Port Feature (PORT_SUSPEND).

A port enters the suspend state after the microcontroller interprets the suspend request and sets the appropriate bits of the Hub Port Control Register, HPCON. From this point on he hub repeater hardware is responsible for proper actions in placing Ports 2:5 in the suspend mode. For Port 1, the embedded function port, the hardware will stop responding to any normal bus traffic, but the microcontroller firmware must place all external circuitry associated with the function in the low-power state.

A port exits from the suspend state when the hub receives a Clear Port Feature (PORT_SUSPEND) or Set Port Feature (PORT_RESET). If the Clear Port Feature (PORT_SUSPEND) is directed towards Ports 2:5, the USB hardware drives a “K” downstream for at least 20 ms followed by a low speed EOP. It then places the port in the enabled state. A Clear Port Feature (PORT_SUSPEND) to Port 1 (the embedded function) causes the firmware to wait 20 ms, take the embedded function out of the suspended state and then enable the port.

The ports can also exit from the suspended state through a remote wakeup if this feature is enabled. For Ports 2:5, this means detection of a connect/disconnect or an upstream directed J to K signaling. Remote wakeup for the embedded function is initiated through a key depression which triggers a KB INT.

8.5.4.3 *Remote Wake-up, Downstream Ports*

The hardware detects a connect/disconnect/port resume and propagates resume signaling upstream. Finally, the hardware enables the oscillator and asserts the RSM interrupt.

Hardware	Firmware
1. Connect/disconnect/port resume detected	
2. Propagate resume signaling	
3. Enable Oscillator	
4. Set RSM bit → interrupt	
	5. Reset RSM and GBL SUSP bits
	6. Restore GPIO states if required
	7. Clear UOVCER bit 2
	8. Enable peripheral activity

8.5.4.4 *Remote Wake-up, Embedded Function*

The hardware detects an INT0/INT1 and propagates resume signaling upstream. Finally, the hardware enables the oscillator and asserts the RSM and FRWUP interrupts.

Hardware	Firmware
1. External event activates INT0/INT1	
2. Propagate resume signaling	
3. Enable Oscillator	
4. Set RSM and FRMWUP bits → interrupt	
	5. Clear GLB SUSP, RSM, FRMWUP bits
	6. Restore GPIO states if required
	7. Clear UOVCER bit 2
	8. Enable peripheral activity

8.5.4.5 *Selective Suspend, Downstream Ports*

Hardware	Firmware
	1. Set or Clear Port Feature PORT_SUSPEND decoded
	2. Write HPCON[2:0] and HPADD[2:0] bits
3. Suspend or resume port per command	

10. Ordering Information

10.1 Standard Package Options

Program Memory	Ordering Code	Package	Operation Range
SRAM	AT43USB325E-AC	64AA LQFP	Commercial (0°C to 70°C)

10.2 Green Package Options (Pb/Halide-free/RoHS Compliant)

Program Memory	Ordering Code	Package	Operation Range
SRAM	AT43USB325E-AU	64AA LQFP	Green, Industrial (-40°C to 85°C)

Package Type	
64AA	64-lead, Low-profile (1.4 mm) Plastic Quad Flat Package (LQFP)