

Welcome to **E-XFL.COM** 

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "<u>Embedded - Microcontrollers</u>"

Details	
Product Status	Active
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	EBI/EMI, I <sup>2</sup> C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	48
Program Memory Size	128KB (128K x 8)
Program Memory Type	FLASH
EEPROM Size	4K x 8
RAM Size	8K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at90usb1286-au

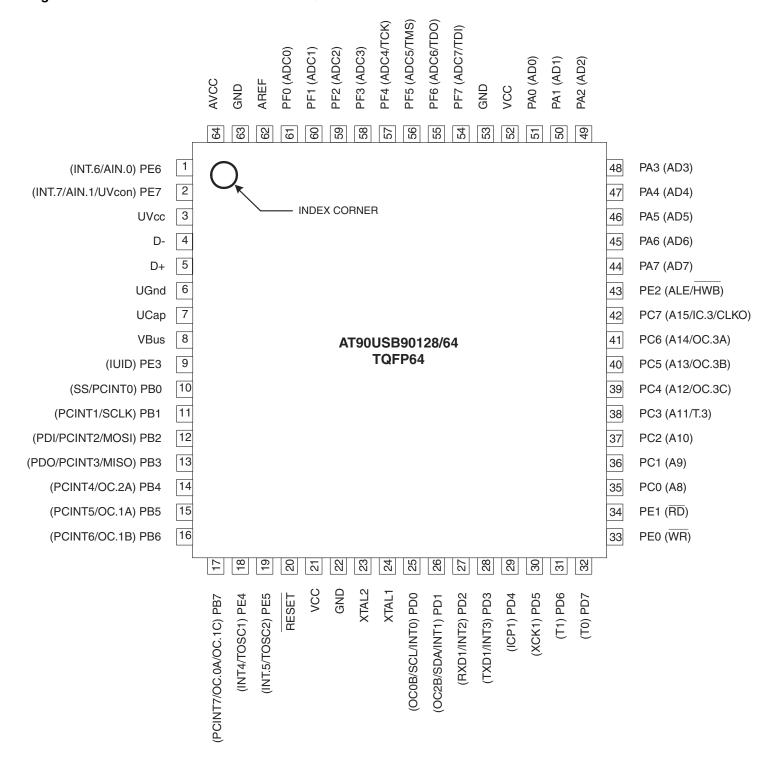
Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

- Real time counter with separate oscillator
- Four 8-bit PWM channels
- Six PWM channels with programmable resolution from 2 to 16 bits
- Output compare modulator
- 8-channels, 10-bit ADC
- Programmable serial USART
- Master/slave SPI serial interface
- Byte oriented 2-wire serial interface
- Programmable watchdog timer with separate on-chip oscillator
- On-chip analog comparator
- Interrupt and wake-up on pin change
- Special microcontroller features
  - Power-on reset and programmable brown-out detection
  - Internal calibrated oscillator
  - External and internal interrupt sources
  - Six sleep modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and packages
  - 48 programmable I/O lines
  - 64-lead TQFP and 64-lead QFN
- Operating voltages
  - 2.7 5.5V
- Operating temperature
  - Industrial (-40°C to +85°C)
- Maximum frequency
  - 8MHz at 2.7V industrial range
  - 16MHz at 4.5V industrial range

# 1. Pin configurations

Figure 1-1. Pinout Atmel AT90USB64/128-TQFP.



# 2. Overview

The Atmel® AVR® AT90USB64/128 is a low-power CMOS 8-bit microcontroller based on the Atmel® AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the AT90USB64/128 achieves throughputs approaching 1MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

# 2.2.8 Port E (PE7..PE0)

Port E is an 8-bit bidirectional I/O port with internal pull-up resistors (selected for each bit). The Port E output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port E pins that are externally pulled low will source current if the pull-up resistors are activated. The Port E pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port E also serves the functions of various special features of the AT90USB64/128 as listed on page 86.

# 2.2.9 Port F (PF7..PF0)

Port F serves as analog inputs to the A/D Converter.

Port F also serves as an 8-bit bidirectional I/O port, if the A/D Converter is not used. Port pins can provide internal pull-up resistors (selected for each bit). The Port F output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port F pins that are externally pulled low will source current if the pull-up resistors are activated. The Port F pins are tri-stated when a reset condition becomes active, even if the clock is not running. If the JTAG interface is enabled, the pull-up resistors on pins PF7(TDI), PF5(TMS), and PF4(TCK) will be activated even if a reset occurs.

Port F also serves the functions of the JTAG interface.

#### 2.2.10 D-

USB Full speed / Low Speed Negative Data Upstream Port. Should be connected to the USB D-connector pin with a serial  $22\Omega$  resistor.

#### 2.2.11 D+

USB Full speed / Low Speed Positive Data Upstream Port. Should be connected to the USB D+ connector pin with a serial  $22\Omega$  resistor.

# 2.2.12 UGND

USB Pads Ground.

#### 2.2.13 UVCC

USB Pads Internal Regulator Input supply voltage.

# 2.2.14 UCAP

USB Pads Internal Regulator Output supply voltage. Should be connected to an external capacitor (1µF).

# 2.2.15 VBUS

USB VBUS monitor and OTG negociations.

#### 2.2.16 **RESET**

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 9-1 on page 58. Shorter pulses are not guaranteed to generate a reset.

# 2.2.17 XTAL1

Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

# 5. Register summary

<u> </u>		· · · · · · · · · · · · · · · ·			1	i		T		
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
(0xFF)	Reserved	-	-	-	-	-	-	-	-	
(0xFE)	Reserved	-	-	-	-	-	-	-	-	
(0xFD)	Reserved	-	-	-	-	-	-	-	-	
(0xFC)	Reserved	-	-	-	-	-	-	-	-	
(0xFB)	Reserved	-	-	-	-	-	-	-	-	
(0xFA)	Reserved	-	-	-	-	-	-	-	-	
(0xF9)	OTGTCON		P.A	AGE				VA	LUE	
(0xF8)	UPINT		•		PII	NT7:0				
(0xF7)	UPBCHX	-	-	-	-	-		PBYCT10:8		
(0xF6)	UPBCLX		•	•	PB\	CT7:0				
(0xF5)	UPERRX	-	COUN	ITER1:0	CRC16	TIMEOUT	PID	DATAPID	DATATGL	
(0xF4)	UEINT					EPINT6:0				
(0xF3)	UEBCHX	-	-	-	-	-		BYCT10:8		
(0xF2)	UEBCLX				BY	CT7:0				
(0xF1)	UEDATX				DA	AT7:0				
(0xF0)	UEIENX	FLERRE	NAKINE	-	NAKOUTE	RXSTPE	RXOUTE	STALLEDE	TXINE	
(0xEF)	UESTA1X	-	-	-	-	-	CTRLDIR	CURF	RBK1:0	
(0xEE)	UESTA0X	CFGOK	OVERFI	UNDERFI	-	DTS	EQ1:0	NBUS	SYBK1:0	
(0xED)	UECFG1X			EPSIZE2:0		EPB	3K1:0	ALLOC		
(0xEC)	UECFG0X	EPTY	/PE1:0				-	-	EPDIR	
(0xEB)	UECONX			STALLRQ	STALLRQC	RSTDT			EPEN	
(0xEA)	UERST			•	•	EPRST6:0				
(0xE9)	UENUM							EPNUM2:0		
(0xE8)	UEINTX	FIFOCON	NAKINI	RWAL	NAKOUTI	RXSTPI	RXOUTI	STALLEDI	TXINI	
(0xE7)	Reserved			-	-	-	-			
(0xE6)	UDMFN				FNCERR					
(0xE5)	UDFNUMH							FNUM10:8		
(0xE4)	UDFNUML		•	•	FN	JM7:0				
(0xE3)	UDADDR	ADDEN				UADD6:0				
(0xE2)	UDIEN		UPRSME	EORSME	WAKEUPE	EORSTE	SOFE		SUSPE	
(0xE1)	UDINT		UPRSMI	EORSMI	WAKEUPI	EORSTI	SOFI		SUSPI	
(0xE0)	UDCON						LSM	RMWKUP	DETACH	
(0xDF)	OTGINT			STOI	HNPERRI	ROLEEXI	BCERRI	VBERRI	SRPI	
(0xDE)	OTGIEN			STOE	HNPERRE	ROLEEXE	BCERRE	VBERRE	SRPE	
(0xDD)	OTGCON			HNPREQ	SRPREQ	SRPSEL	VBUSHWC	VBUSREQ	VBUSRQC	
(0xDC)	Reserved									
(0xDB)	Reserved									
(0xDA)	USBINT							IDTI	VBUSTI	
(0xD9)	USBSTA					SPEED		ID	VBUS	
(0xD8)	USBCON	USBE	HOST	FRZCLK	OTGPADE			IDTE	VBUSTE	
(0xD7)	UHWCON	UIMOD	UIDE		UVCONE				UVREGE	
(0xD6)	Reserved									
(0xD5)	Reserved									
(0xD4)	Reserved									
(0xD3)	Reserved									
(0xD2)	Reserved	-	-	-	-	-	-	-	-	
(0xD1)	Reserved	-	-	-	-	-	-	-	-	_
(0xD0)	Reserved	-	-	-	-	-	-	-	-	
(0xCF)	Reserved	-	-	-	-	-	-	-	-	
(0xCE)	UDR1				USART1 I/C	Data Register				
(0xCD)	UBRR1H	-	-	-	-		SART1 Baud Ra	te Register High B	Byte	
(0xCC)	UBRR1L				JSART1 Baud Ra	te Register Low I				
(0xCB)	Reserved	-	-	-	-	-	-	-	-	
(0xCA)	UCSR1C	UMSEL11	UMSEL10	UPM11	UPM10	USBS1	UCSZ11	UCSZ10	UCPOL1	
(0xC9)	UCSR1B	RXCIE1	TXCIE1	UDRIE1	RXEN1	TXEN1	UCSZ12	RXB81	TXB81	
(0xC8)	UCSR1A	RXC1	TXC1	UDRE1	FE1	DOR1	PE1	U2X1	MPCM1	
(0xC7)	Reserved	-	-	-	-	-	-	-	-	
(0xC6)	Reserved	-	-	-	-	-	-	-	-	
(0xC5)	Reserved	-	-	-	-	-	-	-	-	
(0xC4)	Reserved	-	-	-	-	-	-	-	-	
(0xC3)	Reserved	-	-	-	-	-	-	-	-	
<del></del>	Reserved	-	-	-	-	-	-	-	-	
(0xC2)	neserveu							+		
(0xC2) (0xC1)	Reserved	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x1B (0x3B)	PCIFR	-	-	-	-	-	-	-	PCIF0	
0x1A (0x3A)	Reserved	-	-	-	-	-	-	-	-	
0x19 (0x39)	Reserved	-	-	-	-	-	-	-	-	
0x18 (0x38)	TIFR3	-	-	ICF3	-	OCF3C	OCF3B	OCF3A	TOV3	
0x17 (0x37)	TIFR2	-	-	-	-	-	OCF2B	OCF2A	TOV2	
0x16 (0x36)	TIFR1	-	-	ICF1	-	OCF1C	OCF1B	OCF1A	TOV1	
0x15 (0x35)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0	
0x14 (0x34)	Reserved	-	-	-	-	-	-	-	-	
0x13 (0x33)	Reserved	-	-	-	-	-	-	-	-	
0x12 (0x32)	Reserved	-	-	-	-	-	-	-	-	
0x11 (0x31)	PORTF	PORTF7	PORTF6	PORTF5	PORTF4	PORTF3	PORTF2	PORTF1	PORTF0	
0x10 (0x30)	DDRF	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0	
0x0F (0x2F)	PINF	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0	
0x0E (0x2E)	PORTE	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	
0x0D (0x2D)	DDRE	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	
0x0C (0x2C)	PINE	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0	
0x0B (0x2B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	
0x0A (0x2A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	
0x09 (0x29)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	
0x08 (0x28)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	
0x07 (0x27)	DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	
0x06 (0x26)	PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	
0x05 (0x25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	
0x04 (0x24)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	
0x03 (0x23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	
0x02 (0x22)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	
0x01 (0x21)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	
0x00 (0x20)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	

Note:

- 1. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
- 2. I/O registers within the address range \$00 \$1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions.
- 3. Some of the status flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers 0x00 to 0x1F only.
- 4. When using the I/O specific commands IN and OUT, the I/O addresses \$00 \$3F must be used. When addressing I/O registers as data space using LD and ST instructions, \$20 must be added to these addresses. The Atmel AT90USB64/128 is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from \$60 \$1FF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRVC	k	Branch if Overflow Flag is Cleared	ch if Overflow Flag is Cleared if $(V = 0)$ then $PC \leftarrow PC + k + 1$		1/2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC $\leftarrow$ PC + k + 1	None	1/2
BRID	k	Branch if Interrupt Disabled	if ( I = 0) then PC $\leftarrow$ PC + k + 1	None	1/2
	BIT AN	D BIT-TEST INSTRUCTIONS			
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P,b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	2
LSL	Rd	Logical Shift Left	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z,C,N,V	1
LSR	Rd	Logical Shift Right	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	$Rd(0)\leftarrow C,Rd(n+1)\leftarrow Rd(n),C\leftarrow Rd(7)$	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	$Rd(7)\leftarrow C,Rd(n)\leftarrow Rd(n+1),C\leftarrow Rd(0)$	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	$Rd(n) \leftarrow Rd(n+1), n=06$	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(30)←Rd(74),Rd(74)←Rd(30)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	Т	1
BLD	Rd, b	Bit load from T to Register	Rd(b) ← T	None	1
SEC	112, 2	Set Carry	C ← 1	С	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	1←1	1	1
CLI		Global Interrupt Disable	1←1	1	1
SES		Set Signed Test Flag	I ← 0 S ← 1	S	1
			<del>-</del>	S	1
CLS		Clear Signed Test Flag	S ← 0	V	1
SEV		Set Twos Complement Overflow.	V ← 1		1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T -	1
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	Н	1
CLH		Clear Half Carry Flag in SREG	H ← 0	Н	1
		TRANSFER INSTRUCTIONS		1	
MOV	Rd, Rr	Move Between Registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Copy Register Word	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	$Rd \leftarrow (X)$	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	$Rd \leftarrow (X), X \leftarrow X + 1$	None	2
LD	Rd, - X		$X \leftarrow X - 1$ , $Rd \leftarrow (X)$		2
		Load Indirect and Pre-Dec.		None	
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y Rd, Y+	Load Indirect Load Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$	None None	2 2
LD LD	Rd, Y Rd, Y+ Rd, - Y	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$	None None None	2 2 2
LD LD LDD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q	Load Indirect Load Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$	None None	2 2 2 2
LD LD LDD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$	None None None None None	2 2 2 2 2 2
LD LD LDD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$	None None None	2 2 2 2 2 2 2
LD LDD LDD LD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$	None None None None None	2 2 2 2 2 2 2 2 2
LD LD LDD LD LD LD LD LD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, -Z Rd, -Z Rd, -Z Rd, Z+q	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$	None None None None None None	2 2 2 2 2 2 2 2 2 2
LD LDD LDD LD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$	None None None None None None None None	2 2 2 2 2 2 2 2 2
LD LD LDD LD LD LD LD LD LD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, -Z Rd, -Z Rd, -Z Rd, Z+q	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LDD LD LD LD LD LD LD LD LD LDD LDD	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, -Z Rd, -Z Rd, K	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect from SRAM	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2
LD LDS ST	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, -Z Rd, -Z Rd, -Z Rd, -X Rd,	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LDD LD LD LD LD LD LDS ST ST	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, K X, Rr X+, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LDD LD LD LD LD LD LD ST ST	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+ Rd, k X, Rr X+, Rr - X, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LDD LDD LD LD LD LD LDS ST ST ST	Rd, Y Rd, Y+ Rd, - Y Rd,Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, k X, Rr X+, Rr - X, Rr Y, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow $	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LDD LD LD LD LD LDS ST ST ST ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y+, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y \cdot 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z \cdot 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (k)$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr, X \leftarrow X + 1$ $X \leftarrow X \cdot 1, (X) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y) \leftarrow Rr, Y \leftarrow Y + 1$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LDD LDD LD LDD LDS ST ST ST ST ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y+, Rr - Y, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect Store Indirect and Pre-Dec. Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (K + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LDD LD LD LDD LDS ST ST ST ST ST STD	Rd, Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y+, Rr - Y, Rr Y+q, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (K)$ $(X) \leftarrow Rr$ $(Y) \leftarrow Rr$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LD LD LD LD LD LDS ST ST ST ST ST ST ST ST STD ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z- Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y, Rr - Y, Rr Y+q, Rr Z, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect with Displacement Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD  LD  LD  LD  LD  LD  LD  LDS  ST  ST  ST  ST  ST  ST  ST  ST  ST	Rd, Y Rd, Y+ Rd, -Y Rd, -Y Rd, -Y Rd, Z+ Rd, -Z Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr -X, Rr Y+, Rr -Y, Rr Y+q, Rr Z, Rr Z+q, Rr Z+q, Rr Z+q	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect with Displacement Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect Store Indirect	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (K)$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr, X \leftarrow X + 1$ $X \leftarrow X - 1, (X) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y + q) \leftarrow Rr$ $(Y + q) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z) \leftarrow Rr, Z \leftarrow Z + 1$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD  LD  LD  LD  LD  LD  LDS  ST  ST  ST  ST  ST  ST  ST  ST  ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, k X, Rr X+, Rr -X, Rr Y, Rr -Y, Rr Y+q, Rr Z, Rr Z+q, Rr -Z, Rr -Z, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect with Displacement Load Indirect with Displacement Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect and Post-Inc. Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (X + q)$ $Rd $	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD  LD  LD  LD  LD  LD  LDS  ST  ST  ST  ST  ST  ST  ST  ST  ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, k X, Rr X+, Rr - X, Rr Y+, Rr - Y, Rr Y+q, Rr Z, Rr Z+q, Rr Z+q, Rr Z+q, Rr Z+q, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect with Displacement Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (X + q)$ $Rd $	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LD LD LD LD LDS ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y+, Rr - Y, Rr Y+q, Rr Z, Rr Z+q, Rr k, Rr k, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (Z)$ $(X) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y + q) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z + q) \leftarrow Rr$ $(Z + q) \leftarrow Rr$ $(K) \leftarrow Rr$ $(K) \leftarrow Rr$ $(K) \leftarrow Rr$ $(K) \leftarrow Rr$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LD LD LD LD LDS ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y, Rr Y+, Rr - Y, Rr Y+q,Rr Z, Rr Z+q, Rr k, Rr Rd, Z	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect with Displacement Load Indirect with Displacement Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (X)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LD LD LD LD LD LDS ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Z+ Rd, -Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y+, Rr - Y, Rr Y+q, Rr Z, Rr Z+q, Rr k, Rr k, Rr	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect Store Indirect Store Indirect Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect to SRAM Load Program Memory Load Program Memory	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z + q)$ $Rd \leftarrow (R)$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr$ $(X) \leftarrow Rr$ $(Y) \leftarrow Rr$ $(Y + q) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z) \leftarrow Rr$ $(Z + q) \leftarrow Rr$ $(Z + q) \leftarrow Rr$ $(R) \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
LD LD LD LD LD LD LDS ST	Rd, Y Rd, Y+ Rd, - Y Rd, Y+ Rd, - Y Rd, Y+q Rd, Z Rd, Z+ Rd, -Z Rd, Z+q Rd, k X, Rr X+, Rr - X, Rr Y, Rr Y+, Rr - Y, Rr Y+q,Rr Z, Rr Z+q, Rr k, Rr Rd, Z	Load Indirect Load Indirect and Post-Inc. Load Indirect and Pre-Dec. Load Indirect with Displacement Load Indirect Load Indirect Load Indirect and Post-Inc. Load Indirect with Displacement Load Indirect with Displacement Load Indirect with Displacement Load Direct from SRAM Store Indirect Store Indirect Store Indirect Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Post-Inc. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect and Pre-Dec. Store Indirect with Displacement Store Indirect and Post-Inc.	$Rd \leftarrow (Y)$ $Rd \leftarrow (Y), Y \leftarrow Y + 1$ $Y \leftarrow Y - 1, Rd \leftarrow (Y)$ $Rd \leftarrow (Y + q)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z)$ $Rd \leftarrow (Z), Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, Rd \leftarrow (Z)$ $Rd \leftarrow (X)$	None None None None None None None None	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

#### 7.2 Atmel AT90USB647

Speed [MHz]	Power supply [V]	Ordering code (2)	USB interface	Package (1)	Operating range
16 <sup>(3)</sup>	2.7-5.5	AT90USB647-AU AT90USB647-MU	USB OTG	MD PS	Industrial (-40° to +85°C)

- Notes: 1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
  - 2. Pb-free packaging complies to the European directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully green.
  - 3. See "Maximum speed vs. VCC" on page 392.

MD	64 - lead, 14 × 14mm body size, 1.0mm body thickness 0.8mm lead pitch, thin profile plastic quad flat package (TQFP)
PS	64 - lead, 9 × 9mm body size, 0.50mm pitch Quad flat no lead package (QFN)

#### 7.4 Atmel AT90USB1287

Speed [MHz]	Power supply [V]	Ordering code (2)	USB interface	Package (1)	Operating range
16 <sup>(3)</sup>	2.7-5.5	AT90USB1287-AU AT90USB1287-MU	Host (OTG)	MD PS	Industrial (-40° to +85°C)

- Notes: 1. This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.
  - 2. Pb-free packaging complies to the European directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully green.
  - 3. See "Maximum speed vs. VCC" on page 392.

MD	64 - lead, 14 × 14mm body size, 1.0mm body thickness 0.8mm lead pitch, thin profile plastic quad flat package (TQFP)
PS	64 - lead, 9 × 9mm body size, 0.50mm pitch Quad flat no lead package (QFN)

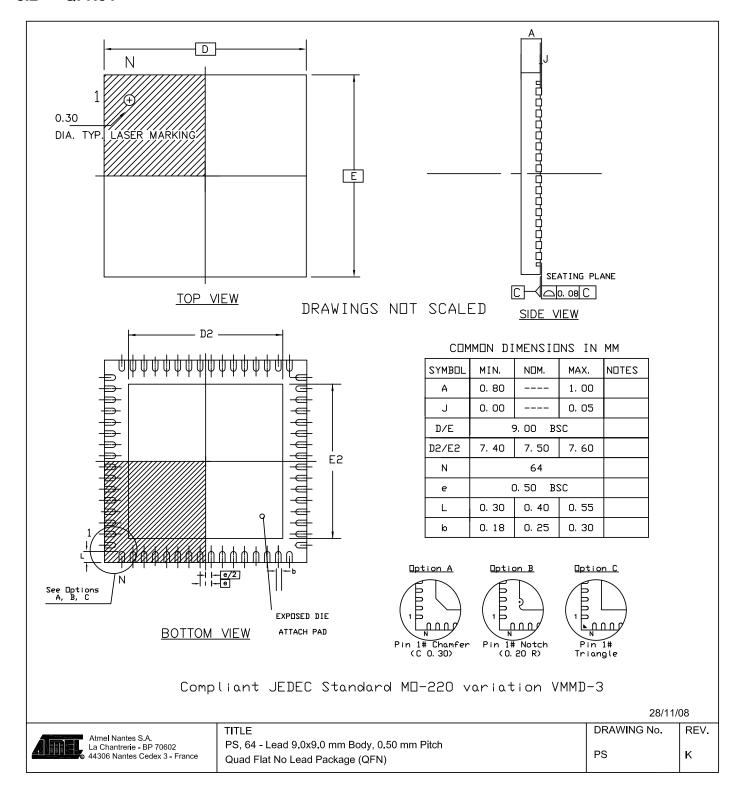
NOTES: STANDARD NOTES FOR PQFP/VQFP/TQFP/DQFP

- 1. DIMENSIONING & TOLERANCING CONFORM TO ASME Y14.5M. 1982.
- 2. "D1 AND E1" DIMENSIONS DO NOT INCLUDE MOLD PROTUSIONS MOLD PROTUSIONS SHALL NOT EXCEED 0.25 mm (0.010 INCH).

  THE TOP PACKAGE BODY SIZE MAY BE SMALLER THAN THE BOTTOM PACKAGE BODY SIZE BY AS MUCH AS 0.15 mm.
- 3. DATUM PLANE "H" LOCATED AT MOLD PARTING LINE AND COINCIDENT WITH LEAD, WHERE LEAD EXISTS PLASTIC BODY AT BOTTOM OF PARTING LINE.
- 4. DATUM "A" AND "D" TO BE DETERMINED AT DATUM PLANE H.
- 5. DIMENSION "f" DOES NOT INCLUDE DAMBAR PROTUSION ALLOWABLE
  DAMBAR PROTUSION SHALL BE 0.08 mm/.003" TOTAL EXCESS OF THE
  "f" DIMENSION AT MAXIMUM MATERIAL CONDITION.

DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.

# 8.2 QFN64



# 9. Errata

#### 9.1 Atmel AT90USB1287/6 errata

# 9.1.1 AT90USB1287/6 errata history

Silicon Release	90USB1286-16MU	90USB1287-16AU	90USB1287-16MU
First Release	Date Code up to 0648	Date Code up to 0714 and lots 0735 6H2726 <sup>(1)</sup>	Date Code up to 0701
Second Release	Date Code from 0709 to 0801 except lots 0801 7H5103 (1)	from Date Code 0722 to 0806 except lots 0735 6H2726 <sup>(1)</sup>	Date Code from 0714 to 0810 except lots 0748 7H5103 <sup>(1)</sup>
Third Release	Lots 0801 7H5103 <sup>(1)</sup> and Date Code from 0814	Date Code from 0814	Lots 0748 7H5103 <sup>(1)</sup> and Date Code from 0814
Fourth Release	TBD	TBD	TBD

Notes: 1. A blank or any alphanumeric string.

#### 9.1.2 AT90USB1287/6 first release

- Incorrect CPU behavior for VBUSTI and IDTI interrupts routines
- USB Eye Diagram violation in low-speed mode
- Transient perturbation in USB suspend mode generates over consumption
- VBUS Session valid threshold voltage
- USB signal rate
- VBUS residual level
- · Spike on TWI pins when TWI is enabled
- High current consumption in sleep mode
- Async timer interrupt wake up from sleep generate multiple interrupts

# 9. Incorrect CPU behavior for VBUSTI and IDTI interrupts routines

The CPU core may incorrectly execute the interrupt vector related to the VBUSTI and IDTI interrupt flags.

# Problem fix/workaround

Do not enable these interrupts, firmware must process these USB events by polling VBUSTI and IDTI flags.

# 8. USB Eye Diagram violation in low-speed mode

The low to high transition of D- violates the USB eye diagram specification when transmitting with low-speed signaling.

#### Problem fix/workaround

None.

#### 7. Transient perturbation in USB suspend mode generates overconsumption

In device mode and when the USB is suspended, transient perturbation received on the USB lines generates a wake up state. However the idle state following the perturbation does

1. Asynchronous timer interrupt wake up from sleep generates multiple interrupts

If the CPU core is in sleep and wakes-up from an asynchronous timer interrupt and then go back in sleep again it may wake up multiple times.

# Problem fix/workaround

A software workaround is to wait with performing the sleep instruction until TCNT2>OCR2+1.

#### Problem fix/workaround

No known workaround, enable Atmel AT90USB64/128 TWI first versus the others nodes of the TWI network.

# 2. High current consumption in sleep mode

If a pending interrupt cannot wake the part up from the selected mode, the current consumption will increase during sleep when executing the SLEEP instruction directly after a SEI instruction.

#### Problem fix/workaround

Before entering sleep, interrupts not used to wake up the part from the sleep mode should be disabled.

# 1. Asynchronous timer interrupt wake up from sleep generates multiple interrupts

If the CPU core is in sleep and wakes-up from an asynchronous timer interrupt and then go back in sleep again it may wake up multiple times.

#### Problem fix/workaround

A software workaround is to wait with performing the sleep instruction until TCNT2>OCR2+1.

#### 9.1.4 Atmel AT90USB1287/6 Third Release

- Incorrect CPU behavior for VBUSTI and IDTI interrupts routines
- Transient perturbation in USB suspend mode generates over consumption
- Spike on TWI pins when TWI is enabled
- High current consumption in sleep mode
- Async timer interrupt wake up from sleep generate multiple interrupts

# 5. Incorrect CPU behavior for VBUSTI and IDTI interrupts routines

The CPU core may incorrectly execute the interrupt vector related to the VBUSTI and IDTI interrupt flags.

#### Problem fix/workaround

Do not enable these interrupts, firmware must process these USB events by polling VBUSTI and IDTI flags.

# 4. Transient perturbation in USB suspend mode generates overconsumption

In device mode and when the USB is suspended, transient perturbation received on the USB lines generates a wake up state. However the idle state following the perturbation does not set the SUSPI bit. The internal USB engine remains in suspend mode but the USB differential receiver is still enabled and generates a typical 300µA extra-power consumption. Detection of the suspend state after the transient perturbation should be performed by software (instead of reading the SUSPI bit).

#### Problem fix/workaround

USB waiver allows bus powered devices to consume up to 2.5mA in suspend state.

# 3. Spike on TWI pins when TWI is enabled

100ns negative spike occurs on SDA and SCL pins when TWI is enabled.

#### Problem fix/workaround

No known workaround, enable AT90USB64/128 TWI first, before the others nodes of the TWI network.

# 2. High current consumption in sleep mode

If a pending interrupt cannot wake the part up from the selected mode, the current consumption will increase during sleep when executing the SLEEP instruction directly after a SEI instruction.

#### Problem fix/workaround

Before entering sleep, interrupts not used to wake up the part from sleep mode should be disabled.

# 1. Asynchronous timer interrupt wake up from sleep generates multiple interrupts

If the CPU core is in sleep mode and wakes-up from an asynchronous timer interrupt and then goes back into sleep mode, it may wake up multiple times.

# Problem fix/workaround

A software workaround is to wait before performing the sleep instruction: until TCNT2>OCR2+1.

#### 9.1.5 Atmel AT90USB1287/6 Fourth Release

- Transient perturbation in USB suspend mode generates over consumption
- Spike on TWI pins when TWI is enabled
- High current consumption in sleep mode
- Async timer interrupt wake up from sleep generate multiple interrupts

# 4. Transient perturbation in USB suspend mode generates overconsumption

In device mode and when the USB is suspended, transient perturbation received on the USB lines generates a wake up state. However the idle state following the perturbation does not set the SUSPI bit. The internal USB engine remains in suspend mode but the USB differential receiver is still enabled and generates a typical 300µA extra-power consumption. Detection of the suspend state after the transient perturbation should be performed by software (instead of reading the SUSPI bit).

#### Problem fix/workaround

USB waiver allows bus powered devices to consume up to 2.5mA in suspend state.

# 3. Spike on TWI pins when TWI is enabled

100ns negative spike occurs on SDA and SCL pins when TWI is enabled.

#### Problem fix/workaround

No known workaround, enable Atmel AT90USB64/128 TWI first, before the others nodes of the TWI network.

# 2. High current consumption in sleep mode

If a pending interrupt cannot wake the part up from the selected mode, the current consumption will increase during sleep when executing the SLEEP instruction directly after a SEI instruction.

#### Problem fix/workaround

Before entering sleep, interrupts not used to wake up the part from sleep mode should be disabled.

# Asynchronous timer interrupt wake up from sleep generates multiple interrupts

If the CPU core is in sleep mode and wakes-up from an asynchronous timer interrupt and then goes back into sleep mode, it may wake up multiple times.

#### Problem fix/workaround

A software workaround is to wait before performing the sleep instruction: until TCNT2>OCR2+1.

#### 9.2.3 Atmel AT90USB646/7 Second Release.

- USB Eye Diagram violation in low-speed mode
- Transient perturbation in USB suspend mode generates over consumption
- Spike on TWI pins when TWI is enabled
- High current consumption in sleep mode
- Async timer interrupt wake up from sleep generate multiple interrupts

# 5. USB Eye Diagram violation in low-speed mode

The low to high transition of D- violates the USB eye diagram specification when transmitting with low-speed signaling.

#### Problem fix/workaround

None.

# 4. Transient perturbation in USB suspend mode generates overconsumption

In device mode and when the USB is suspended, transient perturbation received on the USB lines generates a wake up state. However the idle state following the perturbation does not set the SUSPI bit anymore. The internal USB engine remains in suspend mode but the USB differential receiver is still enabled and generates a typical 300µA extra-power consumption. Detection of the suspend state after the transient perturbation should be performed by software (instead of reading the SUSPI bit).

#### Problem fix/workaround

USB waiver allows bus powered devices to consume up to 2.5mA in suspend state.

#### 3. Spike on TWI pins when TWI is enabled

100ns negative spike occurs on SDA and SCL pins when TWI is enabled.

#### Problem fix/workaround

No known workaround, enable Atmel AT90USB64/128 TWI first versus the others nodes of the TWI network.

# 2. High current consumption in sleep mode

If a pending interrupt cannot wake the part up from the selected mode, the current consumption will increase during sleep when executing the SLEEP instruction directly after a SEI instruction.

## Problem fix/workaround

Before entering sleep, interrupts not used to wake up the part from the sleep mode should be disabled.

# 1. Asynchronous timer interrupt wake up from sleep generates multiple interrupts

If the CPU core is in sleep and wakes-up from an asynchronous timer interrupt and then go back in sleep mode again it may wake up several times.

#### Problem fix/workaround

A software workaround is to wait with performing the sleep instruction until TCNT2>OCR2+1.

# 10. Datasheet revision history for Atmel AT90USB64/128

Please note that the referring page numbers in this section are referred to this document. The referring revision in this section are referring to the document revision.

# 10.1 Changes from 7593A to 7593B

- 1. Changed default configuration for fuse bytes and security byte.
- 2. Suppression of timer 4,5 registers which does not exist.
- 3. Updated typical application schematics in USB section

# 10.2 Changes from 7593B to 7593C

1. Update to package drawings, MQFP64 and TQFP64.

# 10.3 Changes from 7593C to 7593D

 For further product compatibility, changed USB PLL possible prescaler configurations. Only 8MHz and 16MHz crystal frequencies allows USB operation (see Table 7-11 on page 50).

# 10.4 Changes from 7593D to 7593E

- 1. Updated PLL Prescaler table: configuration words are different between AT90USB64x and AT90USB128x to enable the PLL with a 16MHz source.
- 2. Cleaned up some bits from USB registers, and updated information about OTG timers, remote wake-up, reset and connection timings.
- 3. Updated clock distribution tree diagram (USB prescaler source and configuration register).
- 4. Cleaned up register summary.
- 5. Suppressed PCINT23:8 that do not exist from External Interrupts.
- 6. Updated Electrical Characteristics.
- 7. Added Typical Characteristics.
- 8. Update Errata section.

# 10.5 Changes from 7593E to 7593F

- 1. Removed 'Preliminary' from document status.
- 2. Clarification in Stand by mode regarding USB.

# 10.6 Changes from 7593F to 7593G

1. Updated Errata section.

# 10.7 Changes from 7593G to 7593H

- 1. Added Signature information for 64K devices.
- 2. Fixed figure for typical bus powered application
- 3. Added min/max values for BOD levels
- 4. Added ATmega32U6 product
- 5. Update Errata section
- 6. Modified descriptions for HWUPE and WAKEUPE interrupts enable (these interrupts should be enabled only to wake up the CPU core from power down mode).

7. Added description to access unique serial number located in Signature Row see "Reading the Signature Row from software" on page 354.

# 10.8 Changes from 7593H to 7593I

1. Updated Table 9-2 in "Brown-out detection" on page 60. Unused BOD levels removed.

# 10.9 Changes from 7593I to 7593J

- 1. Updated Table 9-2 in "Brown-out detection" on page 60. BOD level 100 removed.
- 2. Updated "Ordering information" on page 18.
- 3. Removed ATmega32U6 errata section.

# 10.10 Changes from 7593J to 7593K

- 1. Corrected Figure 6-7 on page 34, Figure 6-8 on page 34 and Figure 6-9 on page 35.
- Corrected ordering information for Section 7.3 "Atmel AT90USB1286" on page 20, Section 7.4 "Atmel AT90USB1287" on page 21 and Section 7.2 "Atmel AT90USB647" on page 19.
- 3. Removed the ATmega32U6 device and updated the datasheet accordingly.
- 4. Updated Assembly Code Example in "Watchdog reset" on page 61.

# 10.11 Changes from 7593K to 7593L

- 1. Updated the "Ordering information" on page 18. Changed the speed from 20MHz to 16MHz.
- 2. Replaced ATmegaAT90USBxxxx by AT90USBxxxx through the datasheet.
- 3. Updated the first paragraph of "Overview" on page 307. Port A replaced by Port F.
- 4. Updated ADC equation in "ADC conversion result" on page 318. The equation has 1024 instead of 1023.
- 5. Created "Packaging Information" chapter.
- 6. Replaced the "QFN64" Packaging by an updated QFN64 Packaging drawing.
- 7. Updated "Errata" on page 26. AT90USB1286/7 has a fourth release, while AT90USB646/7 updated with a second release.
- 8. In Section "Overview" on page 307, "Port A" has been replaced by "Port F" in the first section.
- 9. In Section "Atmel AT90USB647" on page 19 the USB interface has been changed to USB OTG.
- 10. In Section "Atmel AT90USB1286" on page 20 the USB interface has been changed to Device.
- 11. In Section "Atmel AT90USB1287" on page 21 the USB interface has been changed to Host OTG.
- 12. General update according to new template.