Features

- Utilizes the AVR[®] RISC Architecture
- AVR High-performance and Low-power RISC Architecture
 - 120 Powerful Instructions Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
- Up to 20 MIPS Throughput at 20 MHz
- Data and Non-volatile Program and Data Memories
 - 2K Bytes of In-System Self Programmable Flash Endurance 10,000 Write/Erase Cycles
 - 128 Bytes In-System Programmable EEPROM Endurance: 100,000 Write/Erase Cycles
 - 128 Bytes Internal SRAM
 - Programming Lock for Flash Program and EEPROM Data Security
- Peripheral Features
 - One 8-bit Timer/Counter with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture Modes
 - Four PWM Channels
 - On-chip Analog Comparator
 - Programmable Watchdog Timer with On-chip Oscillator
 - USI Universal Serial Interface
 - Full Duplex USART
- Special Microcontroller Features
 - debugWIRE On-chip Debugging
 - In-System Programmable via SPI Port
 - External and Internal Interrupt Sources
 - Low-power Idle, Power-down, and Standby Modes
 - Enhanced Power-on Reset Circuit
 - Programmable Brown-out Detection Circuit
 - Internal Calibrated Oscillator
- I/O and Packages
 - 18 Programmable I/O Lines
 - 20-pin PDIP, 20-pin SOIC, 20-pad QFN/MLF
- Operating Voltages
 - 1.8 5.5V (ATtiny2313V)
 - 2.7 5.5V (ATtiny2313)
- Speed Grades
 - ATtiny2313V: 0 4 MHz @ 1.8 5.5V, 0 10 MHz @ 2.7 5.5V
 - ATtiny2313: 0 10 MHz @ 2.7 5.5V, 0 20 MHz @ 4.5 5.5V
- Typical Power Consumption
 - Active Mode
 - 1 MHz, 1.8V: 230 μA
 - 32 kHz, 1.8V: 20 µA (including oscillator)
 - Power-down Mode
 - < 0.1 µA at 1.8V



8-bit **AVR**[®] Microcontroller with 2K Bytes In-System Programmable Flash

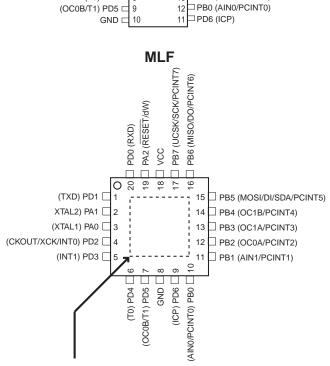
ATtiny2313/V

Summary









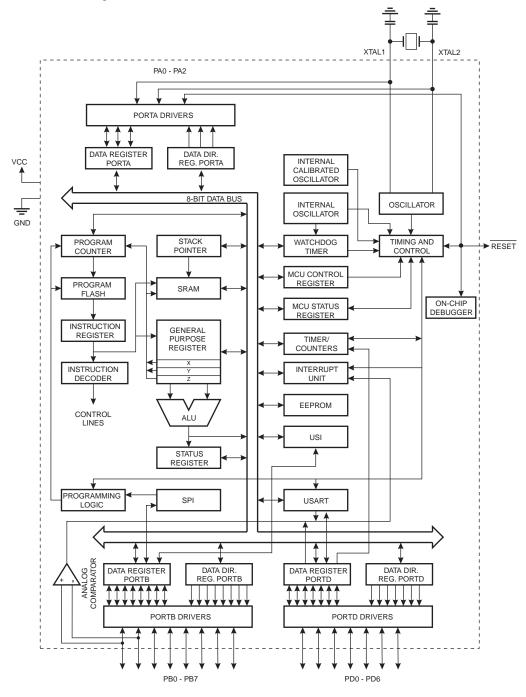
NOTE: Bottom pad should be soldered to ground.

Overview The ATtiny2313 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATtiny2313 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

² ATtiny2313

Block Diagram









The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATtiny2313 provides the following features: 2K bytes of In-System Programmable Flash, 128 bytes EEPROM, 128 bytes SRAM, 18 general purpose I/O lines, 32 general purpose working registers, a single-wire Interface for On-chip Debugging, two flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, Universal Serial Interface with Start Condition Detector, a programmable Watchdog Timer with internal Oscillator, and three software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel's high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, or by a conventional non-volatile memory programmer. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATtiny2313 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATtiny2313 AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

Pin Descriptions

GND Ground.

Port A (PA2..PA0) Port A is a 3-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port A output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port A pins that are externally pulled low will source current if the pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port A also serves the functions of various special features of the ATtiny2313 as listed on page 53.

Port B (PB7..PB0) Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port B also serves the functions of various special features of the ATtiny2313 as listed on page 53.

Port D (PD6..PD0) Port D is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATtiny2313 as listed on page 56.

- **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 15 on page 34. Shorter pulses are not guaranteed to generate a reset. The Reset Input is an alternate function for PA2 and dW.
- **XTAL1** Input to the inverting Oscillator amplifier and input to the internal clock operating circuit. XTAL1 is an alternate function for PA0.
- **XTAL2** Output from the inverting Oscillator amplifier. XTAL2 is an alternate function for PA1.





General Information	
Resources	A comprehensive set of development tools, application notes and datasheets are available for download at http://www.atmel.com/avr.
Code Examples	This documentation contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.
Data Retention	Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

Register Summary

Defi SREE I T H S V N Z C = 64D 0460 SPL 9F2 SPB SPS SP1 SP3 SP1 SP1 SP3 SP1 SP1 <td< th=""><th>Address</th><th>Name</th><th>Bit 7</th><th>Bit 6</th><th>Bit 5</th><th>Bit 4</th><th>Bit 3</th><th>Bit 2</th><th>Bit 1</th><th>Bit 0</th><th>Page</th></td<>	Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
Outling Ling Personal International Probability B International Probability B International Probability B Personal Probability B	0x3E (0x5E)	SBEG	1	т	н	S	v	N	7	C	
0x00 0x00 0CP080 PC/E			-	-		-	-		-		0
0.68 (0.68) GMSK NTT NTO PC/F - - - - - 0 0.68 (0.64) TMSK TOTE O.CDEA ØR10 O.CDEA 7010 O.CDEA 7810 O.CDEA 781 TOTE O.CDEA 781 O.CDEA 771 O.CDEA 771 O.CDEA 772 O.CDEA 772 O.CDEA 772 O.CDEA FORM FORM <td></td> <td></td> <td>SP7</td> <td>SP6</td> <td>SP5</td> <td>SP4</td> <td>SP3</td> <td>SP2</td> <td>SP1</td> <td>SP0</td> <td>11</td>			SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	11
0xA (0xA) EFR INIFI INIFI OCE IA OCE IA <thoce ia<="" th=""> OCE IA OCE IA OCE IA<td>0x3C (0x5C)</td><td>OCR0B</td><td></td><td></td><td>-</td><td>Timer/Counter0 -</td><td>Compare Registe</td><td>er B</td><td></td><td></td><td>77</td></thoce>	0x3C (0x5C)	OCR0B			-	Timer/Counter0 -	Compare Registe	er B			77
ch2@0.89 TMMK TOTEL OCEFA OCEFA OCEFA OFA OFA <thofa< th=""> OFA OFA</thofa<>	0x3B (0x5B)	GIMSK	INT1	INT0	PCIE	-	-	-	-	-	60
0-28 (0.58) TPF TPF OCF/B						-	-	-	-	-	
0-27 (007) SPACSB - - CTPR PIA PAYMET PAYMET PIAL SELFPACEN 1161 0:058 (056) MCURR PUD SM1 SE SM0 ISC11 ISC10 ISC10<						-					
Ox80 055 OCRAN The Counter - Compare Register A Test - Solo 77 0530 0551 MCURN P.D SMI SEC11 SEC01 ISC01 ISC01 ISC01 SEC02 SS1 0530 0551 TCKR0 FOCMA FOCMA FOCMA SC02 CS10 SC02 CS10 SC020 SS1 0530 0551 TCKR0 FOCMA COMA COMA COMA COMA SC020 SS1 0520 0551 TCCR1A COMA COMA COMA COMA COMA Test - Comment Register Altig SS1 CS11 SS10 104 0520 0401 TCCR1A COMA COMA COMA COMA SS1 Test - Comment Register Altig SS1 SS10 105 0520 0401 TCNT11 Test - Comment Register Altig SS1 SS1 SS1 SS10 107 SS10 106 SS1 SS10 107 SS10 107 SS10 106 SS10 107 SS10 SS10 SS10											
Ox8 (0.95) MU2R PUD SM1 SE SM0 SC11 ISC10 ISC20 ISC20 SG20 SG3 Ox8 (0.95) TCCR08 FCC0A FCC0A CA10 A CA10 SG20 CS30			-	-					PGERS	SELFPRGEN	
dr.d. (b.d.) MCURE MCURE BCRF EVR PORF 97 0.363 (b.S.) TONRB F-004 FOORB FOORB 75 0.363 (b.S.) TONRB FOORB CALA Trans-Caunier1 - Caunier Register Major Major Trans-Caunier1 - Trans-Caunier Trans-Caunie			PLID	SM1			, , , , , , , , , , , , , , , , , , ,		19001	18000	
0x32 (0x32) TOCN09 FOC0A FOC0A FOC0A COUNT			-	-	-	-					
0x82 (0x57) CTXT0 ThereCounted (8:00) 77 0x82 (0x57) COSCAL - CAL CAL3 CAL2 CAL1 ThereCountert - Counter Register Hight Byte CAL2 CAL1 ThereCountert - Counter Register Hight Byte CAL2 CAL1 CAL2 CAL2 CAL1 CAL2 CAL2 CAL1 CAL2 CAL1 CAL1 CAL1 <td></td> <td></td> <td>FOC0A</td> <td>FOC0B</td> <td>-</td> <td>-</td> <td></td> <td></td> <td></td> <td></td> <td></td>			FOC0A	FOC0B	-	-					
0.00_050] TCCRNA COMMA1 COMMA0 COMM00 COMM00				•	•	Timer/Co			•	•	
obd/ Odd/ Odd/ Odd/ Odd/ Odd/ Odd/ Odd/ O	0x31 (0x51)	OSCCAL	-	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	26
0x62 (0x4E) TCGTH ICKT	0x30 (0x50)	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00	73
0x00 (bx0) TXRTH ImmerCounter 1 - Counter Register Man Byte 108 0x02 (bx6) OCR1AH TimerCounter 1 - Compare Register A Low Byte 108 0x02 (bx6) OCR1AH TimerCounter 1 - Compare Register A Low Byte 109 0x02 (bx6) OCR1BH TimerCounter 1 - Compare Register A Low Byte 109 0x02 (bx6) OCR1BH TimerCounter 1 - Compare Register B Man Byte 109 0x02 (bx6) OCR1BH TimerCounter 1 - Compare Register Byte Byte 109 0x02 (bx6) OLKPS0 CLKPS0 CLKPS0 CLKPS0 0x02 (bx6) OLKPS0 TimerCounter 1 - Input Capture Register Low Byte 109 0x02 (bx6) OLKPS0 ZLKPS0 CLKPS0 28 0x02 (bx6) ICHPS0 TimerCounter 1 - Input Capture Register Low Byte 109 0x02 (bx6) ICHPS0 ZCKPS0 ZLKPS0 ZLKPS0 0x12 (bx6) ICHPS0 TimerCounter 1 - Input Capture Register Low Byte 109 0x22 (bx6) ICHPS0 ZCKPS0 ZLKPS0 ZLKPS0 0x12 (bx6) ICHPS0 FOCHS0 </td <td>0x2F (0x4F)</td> <td>TCCR1A</td> <td>COM1A1</td> <td>COM1A0</td> <td>COM1B1</td> <td>COM1BO</td> <td>-</td> <td>-</td> <td>WGM11</td> <td>WGM10</td> <td>104</td>	0x2F (0x4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1BO	-	-	WGM11	WGM10	104
DoC (DAC) TONTL There/Counter 1 - Counter Register Low Bye 108 0x8 (0x40) OCB1AH There/Counter 1 - Compare Register A Lew Byr 108 0x8 (0x40) OCB1BL There/Counter 1 - Compare Register Barby Byr 109 0x8 (0x40) OCB1BL There/Counter 1 - Compare Register Barby Byr 109 0x8 (0x40) OCB1BL There/Counter 1 - Input Capture Register High Byra 109 0x6 (0x45) I/OR1H There/Counter 1 - Input Capture Register High Byra 109 0x6 (0x45) I/OR1H There/Counter 1 - Input Capture Register High Byra 109 0x6 (0x45) I/OR1H There/Counter 1 - Input Capture Register High Byra 109 0x6 (0x45) I/OR1H There/Counter 1 - Input Capture Register High Byra 109 0x6 (0x45) I/OR1H PORTA PORTA PORTA 109 0x6 (0x45) PORTA PORTA </td <td></td> <td></td> <td>ICNC1</td> <td>ICES1</td> <td></td> <td></td> <td></td> <td></td> <td>CS11</td> <td>CS10</td> <td></td>			ICNC1	ICES1					CS11	CS10	
0x88 0x081.44 COR14.44 TimerCounter1 - Compare Register A Low Byte 108 0x28 0x081.44 TimerCounter1 - Compare Register B Low Byte 109 0x28 0x081.44 TimerCounter1 - Compare Register B Low Byte 109 0x28 0x081.048 0x081.048 CLKPS3 CLKPS3 CLKPS3 CLKPS3 2.8 0x26 0x45 0x47 Reserved - 109 0x24 0x34 0x31 0x41 NDTE NDE	. ,						* .				
0xA 0xA) OCR1AL ThereCounter 1 - Compare Register A Low Myre 108 0xB 0x49) OCR1BH TimerCounter 1 - Compare Register Blys Byre 109 0xB 0x49) OCR1BH TimerCounter 1 - Compare Register Blys Byre 109 0xB 0x49) OCR1BH CLKPS2 CLKPS2 CLKPS1 CLKPS0 28 0xB 0x45) ICR1H TimerCounter 1 - Input Capture Register High Byre 109 002 004 0044 ICR1E 109 0x26 0x44) ICR1H TimerCounter 1 - Input Capture Register High Byre 109 002 0040 0041	. ,						*				
0x29 (0x49) COR1BH TimerCounter1 - Compare Register B Lyn Byte 109 0x29 (0x47) Reserved								· · · ·			
0261 (0x48) OCR181. TimerCounter1 - Compare Register BLow Byte 109 0267 (0x47) Reserved - 109 0x3 (0x4) 109 0x4 (0x4) 109 0x2 (0x4) 109 0x4 (0x4) 100 10	. ,										
0±27 (0x7) Reserved 0±26 (0x6) − − − − − − − 0±27 (0x6) 0±47 (0x6) 0±17 (0x6) 0±18 (0x6) 0±18								0,			
0x26 (0x46) CLKPER CLKPCE Image: Counter 1 input Capture Register High Byte CLKPS3 CLKPS3 CLKPS4			_	_			1	_	_	_	100
0x26 (0x43) CPC1H Timer/Counter1 - Input Capture Register Low Byte 109 0x28 (0x44) GTCCR - - - - PSR10 81 0x28 (0x43) GTCCR - - - - - PSR10 81 0x22 (0x42) TCCRIC FCC1A FCC1B - - - - - - 108 0x21 (0x41) WOTCSR WOTF PCINT2 FCINT3 PCINT2 PCINT3 PCINT3 <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>CLKPS2</td> <td>CLKPS1</td> <td>CLKPS0</td> <td>28</td>								CLKPS2	CLKPS1	CLKPS0	28
0d23 (0x43) OTCCR - - - - - PSR10 81 0x22 (0x42) TCCR1C FOC1A FOC1B - - - - - - - - - - - - 108 0x21 (0x41) WDTCS FOC1B WDP3 WDCE WDP2 WDP1 WDP0 42 0x16 (0x40) PCMSK PCINT7 PCINT6 PCINT5 PCINT2 PCINT2 PCINT0 61 0x16 (0x30) EEAR - <td></td> <td></td> <td></td> <td></td> <td>Timer/</td> <td>Counter1 - Input (</td> <td>Capture Register</td> <td>High Byte</td> <td>•</td> <td>•</td> <td>109</td>					Timer/	Counter1 - Input (Capture Register	High Byte	•	•	109
0x22 (xx42) TCCR1C FOC1A FOC1B - - - - - - - - 108 0x21 (xx41) WDTCSR WDIF WDIF WDP3 WDCE WDE WDP2 WDP1 WDP0 42 0x20 (xx40) PCMSK PCINTS PCINT1 PCINT0 661 661 0x1F (0x3F) Reserved - PORTA1 PORTA0 58 - - - - PORTA1 PORTA0 58 - - - - PORTA1 PORTA1 PORTA0	0x24 (0x44)	ICR1L			Timer/	Counter1 - Input	Capture Register	Low Byte			109
Dx21 (bx11) WDIF WDIF WDIE WDP3 WDCE WDE WDP2 WDP1 WDP0 42 0x21 (bx41) PCMSK PCINT7 PCINT6 PCINT5 PCINT3 P	0x23 (0x43)	GTCCR	-	-	-	-	-	-	-	PSR10	81
0x20 (0x40) PCMSK PCINT7 PCINT6 PCINT5 PCINT4 PCINT3 PCINT2 PCINT1 PCINT0 61 0x1F (0x3F) Reserved _ 17 0	0x22 (ox42)	TCCR1C	FOC1A	FOC1B	-	-	-	-	-	-	108
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $											
0x1E (bx3E) EEAR - EEPROM datress Register 16 0x1D (0x3D) EEDR - - EEPROM Data Register 17 0x1D (0x3D) EECR - - EEPROM Data Register 17 0x1B (0x3B) PORTA - - - PORTA2 PORTA1 PORTA0 58 0x1B (0x3B) PORTA - - - PORTA2 PORTA1 PORTA0 58 0x19 (0x39) PINA - - - PORTB3 PORTB1 PORTB0 58 0x18 (0x38) PORTB PORTB7 PORTB6 PORTB6 PORTB4 PORTB3 PORTB2 PORTB1 PORTB0 58 0x16 (0x36) PINB7 PINB6 PINB5 PINB4 PINB3 PINB2 PINB1 PORTB0 58 0x14 (0x34) GPIOR2 General Purpose IO Register 2 21 21 21 21 21 21 21 21 21 21 21 21 21 <td></td> <td></td> <td>PCINT7</td> <td>PCINT6</td> <td></td> <td>PCINT4</td> <td>PCINT3</td> <td>PCINT2</td> <td>PCINT1</td> <td>1</td> <td>61</td>			PCINT7	PCINT6		PCINT4	PCINT3	PCINT2	PCINT1	1	61
0x1D (0x3D) EEDR EEPRM Data Register 17 0x1C (0x3C) EECR - - EEPM1 EEPM0 EENR EENR EENR EENR EERR 17 0x1G (0x3G) EECR - - - PORTA2 PORTA1 PORTA0 58 0x16 (0x38) PORTB PORTB PORTB PORTB2 PORTB1 PORTB0 58 0x19 (0x39) PINA - - - PORTB2 PORTB1 PORTB0 58 0x16 (0x36) PORTB PORTB7 PORTB6 PORTB5 PORTB4 PORTB3 PORTB2 PORTB1 PORTB0 58 0x16 (0x36) PINB7 PINB6 PINB5 PINB4 PINB3 PINB3 PINB1 PINB0 58 0x16 (0x36) GPIOR1			-	-	-			-	-	-	10
0x1C (0x3C) EECR - - EEPM1 EEPM0 EERIE EEMPE EEPE EERE 17 0x1B (0x3B) PORTA - - - - PORTA2 PORTA1 PORTA0 58 0x1B (0x3B) DPAA - - - DDA2 DDA1 DDA0 58 0x19 (0x39) PINA - - - PORTB2 PORTB4 PORTB2 PORTB1 PORTB0 58 0x17 (0x37) DDRB DBB7 DDB6 DDS5 DDB4 DDB3 DDB4 DDB3 DDB4 DDB3 DDB4 DB8 DDB0 58 0x16 (0x36) PINB PINB7 PINB6 PINB5 PINB4 PINB3 PINB2 PINB1 PINB0 58 0x16 (0x36) GPIOR2 CGeneral Purpose I/O Register 2 21 21 0x13 (0x33) GPIOR1 - 21 0x14 (0x34) DPIOR1 PORTD - 21 0x14 (0x34) DPIND -			_					egister			
Dx1B (bx3B) PORTA _			_	-	EEPM1			FEMPE	FFPF	FEBE	
Dx1A (bx3A) DDRA - - - - DDA2 DDA1 DDA0 58 0x19 (bx39) PINA - - - - PINA2 PINA1 PINA0 58 0x19 (bx38) PORTB PORTB1 PORTB1 PORTB1 PORTB1 PORTB1 PORTB0 58 0x17 (bx37) DDR8 DDB7 DDB6 DDB5 DDB4 DDB3 DDB2 DDB1 DDB0 58 0x16 (bx36) GPIOR PINB4 PINB5 PINB4 PINB3 PINB2 PINB1 PORTB0 58 0x16 (bx35) GPIOR0 - General Purpose I/O Register 2 21 0.14 (b.34) GPIOR0 - 21 0.14 (b.34) DDD1 DDD0 58 0.11 (b.31) DDR1 - PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x16 (bx25) USIGR USIGP PIND6 PIND4 PIND3 PIND2 PIND1 PIND0			_	_							
0x18 (0x38) PORTB PORTB PORTB6 PORTB5 PORTB4 PORTB3 PORTB2 PORTB1 PORTB0 58 0x17 (0x37) DDRB DDB7 DDB6 DDB5 DDB4 DDB3 DDB2 DDB1 DDB0 58 0x16 (0x35) GPIOR2			-	-	-	-	-				58
0x17 (0x37) DDRB DDB7 DDB6 DDB5 DDB4 DDB3 DDB2 DDB1 DDB0 58 0x16 (0x36) PINB PINB7 PINB6 PINB5 PINB4 PINB3 PINB2 PINB1 PINB1 PINB0 58 0x15 (0x35) GPIOR1 GPIOR1 General Purpose I/O Register 1 21 0x13 (0x33) GPIOR0 - PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x11 (0x31) DDD0 - PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 DORTD1 PORTD0 58 0x10 (0x30) PIND - PIND6 PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x06 (0x2E) USISR USISIF USIOF USIC USICNT3 USICNT1 USICNT0 144 0x06 (0x2C) UDR USISF USIC USICNT3 USICNT1 USICNT0 145 0x06 (0x2C) UDR <	0x19 (0x39)	PINA	-	-	-	-	-	PINA2	PINA1	PINA0	58
0x16 (0x36) PINB PINB7 PINB6 PINB5 PINB4 PINB3 PINB2 PINB1 PINB0 58 0x14 (0x34) GPIOR2 General Purpose I/O Register 2 21 0x14 (0x34) GPIOR1 General Purpose I/O Register 0 21 0x12 (0x32) PORTD – PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x11 (0x31) DDRD – DDD6 DDD5 DDD4 DDD3 DDD2 DDD1 DDD0 58 0x10 (0x30) PIND – PIND6 PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x0F (0x2F) USISIR USISIF USIOF USIPF USIC USICNT3 USICNT1 USICNT0 144 0x0B (0x2E) USISR USISIF USIOF USIPF USIC USICS1 USICNT1 USICNT0 145 0x0B (0x2B) UCSRA RXC TXC UDRE FE DOR	0x18 (0x38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	58
Ox15 (0x35) GPIOR2 General Purpose I/O Register 2 21 0x14 (0x34) GPIOR1 General Purpose I/O Register 1 21 0x13 (0x33) GPIOR0 General Purpose I/O Register 0 21 0x13 (0x33) GPIOR0 - PORTD5 PORTD4 PORTD2 PORTD1 PORTD0 58 0x11 (0x31) DDR0 - DDD6 DDD5 PDD4 PDD3 DDD2 DDD1 DD00 58 0x10 (0x30) PIND - PIND6 PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x06 (0x2F) USIR USISIF USIOF USID2 USICNT1 USICNT1 USICNT0 144 0x06 (0x2C) UDR USIOF USINM1 USINM0 USICS1 USICNT1 USICNT0 145 0x06 (0x2D) USICR USISIE USINH USIWM0 USICS1 USICX USICLK USITC 145 0x06 (0x2A) UCSRB RXCI TXC UDRE											
0x14 (0x34) GPIOR1 General Purpose I/O Register 1 21 0x13 (0x33) GPIOR0 - General Purpose I/O Register 0 21 0x12 (0x32) PORTD - PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x11 (0x30) DDR0 - DD06 DD05 DD04 DD03 DD02 DD01 DD00 58 0x10 (0x30) PIND - PIND6 PIND5 PIND4 PIND2 PIND1 PIND0 58 0x0F (0x2F) USIDR - PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x0F (0x2F) USISR USISIF USIOF USIPF USIOC USICNT3 USICNT1 USICNT0 145 0x06 (0x2C) UDR - - UART Data Register (8-bit) 129 145 145 0x06 (0x2B) USRA RXC TXC UDRE FE DOR UPE U2X MPCM			PINB7	PINB6	PINB5				PINB1	PINB0	
0x13 (0x33) GPIOR0 General Purpose I/O Register 0 21 0x12 (0x32) PORTD - PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x11 (0x31) DDR0 - DDD6 DDD5 DDD4 DDD3 DDD2 DDD1 DDD0 58 0x10 (0x30) PIND - PIND6 PIND5 PIND4 PIND3 PIND2 DDD1 DDD0 58 0x0F (0x2F) USISR USISIF USIOF USIPF USIDC USICNT3 USICNT1 USICNT0 145 0x0D (0x2D) USICR USISIF USIOF USIW11 USIW0 USICS1 USICN1 USICNT1 145 0x0D (0x2C) UDR - UART Data Register (8-bit) 129 145 129 145 0x0A (0x2A) UCSRA RXC TXC UDRE FE DOR UPE U2X MPCM 129 0x0A (0x2A) UCSRB RXCI TXC<											
0x12 (0x32) PORTD - PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0 58 0x11 (0x31) DDRD - DDD6 DDD5 DDD4 DDD3 DDD2 DDD1 DDD0 58 0x10 (0x30) PIND - PIND6 PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x06 (0x2F) USIDR - PIND6 PIND5 PIND4 PIND3 PIND2 PIND1 PIND0 58 0x06 (0x2F) USIDR USISIF USIOF USIPF USIDC USICNT3 USICNT1 USICNT0 144 0x02 (0x2C) UDR USIR USIOF USIW11 USIWM0 USICS1 USICNT1 USICNT0 145 0x02 (0x2C) UDR RXC TXC UDRE FE DOR UPE U2X MPCM 129 0x08 (0x28) USRBR RXCIE TXCIE UDRIE RXEN TXEN	()						*				
$\begin{array}{c c c c c c c c c c c c c c c c c c c $			_	PORTD6	PORTD5		, , , , , , , , , , , , , , , , , , ,	1	PORTD1	PORTDO	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $			_								
0x0F (0x2F) USIDR USISIF USIOF USIDE USIDE USIC USICNT3 USICNT2 USICNT1 USICNT0 144 0x0E (0x2E) USISR USISIF USIOF USIPF USIDC USICNT3 USICNT2 USICNT1 USICNT0 145 0x0D (0x2D) USICR USISIE USIOE USIWM1 USIWM0 USICS1 USICS0 USICLK USICT 145 0x0C (0x2C) UDR UART Data Register (8-bit) 129 129 0x0B (0x2A) UCSRB RXCIE TXCIE UDRE FE DOR UPE U2X MPCM 129 0x0A (0x2A) UCSRB RXCIE TXCIE UDRIE RXEN TXEN UCSZ2 RXB8 TXB8 131 0x09 (0x29) UBRRL UBRH[7:0] 133 133 0x06 (0x26) Reserved - - - - - - - - - - - <td< td=""><td></td><td></td><td>-</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></td<>			-								
$\begin{array}{c c c c c c c c c c c c c c c c c c c $				·	·			·	·	·	
0x0C (0x2C) UDR UART Data Register (8-bit) 129 0x0B (0x2B) UCSRA RXC TXC UDRE FE DOR UPE U2X MPCM 129 0x0A (0x2A) UCSRB RXCIE TXCIE UDRIE RXEN TXEN UCSZ2 RXB8 TXB8 131 0x09 (0x29) UBRRL UBRRL UBRRH[7:0] 133 133 0x08 (0x28) ACSR ACD ACBG ACO ACI ACIE ACIC ACIS1 ACIS0 149 0x07 (0x27) Reserved - <	0x0E (0x2E)	USISR	USISIF	USIOIF	USIPF			USICNT2	USICNT1	USICNT0	145
0x0B (0x2B) UCSRA RXC TXC UDRE FE DOR UPE U2X MPCM 129 0x0A (0x2A) UCSRB RXCIE TXCIE UDRIE RXEN TXEN UCSZ2 RXB8 TXB8 131 0x09 (0x29) UBRL UBRRL UBRRH(7:0) 133 0x08 (0x28) ACSR ACD ACBG ACO ACI ACIE ACIC ACIS1 ACIS0 149 0x07 (0x27) Reserved - <	0x0D (0x2D)	USICR	USISIE	USIOIE	USIWM1			USICS0	USICLK	USITC	145
0x0A (0x2A) UCSRB RXCIE TXCIE UDRIE RXEN TXEN UCSZ2 RXB8 TXB8 131 0x09 (0x29) UBRL	. ,			r	1				r		
0x09 (0x29) UBRRL UBRRL 133 0x08 (0x28) ACSR ACD ACBG ACO ACI ACIE ACIC ACIS1 ACIS0 149 0x07 (0x27) Reserved -	. ,										
0x08 (0x28) ACSR ACD ACBG ACO ACI ACIE ACIC ACIS1 ACIS0 149 0x07 (0x27) Reserved -			RXCIE	TXCIE	UDRIE			UCSZ2	RXB8	TXB8	
0x07 (0x27) Reserved -	()			ACRO	400				10101	40100	
0x06 (0x26) Reserved -						ACI	AGIE	ACIC	A0151	A0150	149
0x05 (0x25) Reserved -				_	_	_	_		_	_	
0x04 (0x24) Reserved -				_	_	_	_	_	_		
0x03 (0x23) UCSRC - UMSEL UPM1 UPM0 USBS UCSZ1 UCSZ0 UCPOL 132 0x02 (0x22) UBRRH - - - - UBRRH - 133 0x01 (0x21) DIDR - - - - AIN1D AIN0D 150				-	_	-	_	_	-	-	
0x02 (0x22) UBRRH - - - - UBRRH - 133 0x01 (0x21) DIDR - - - - AIN1D AIN0D 150			-	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	132
			-					UBRI			
0x00 (0x20) Reserved	0x01 (0x21)	DIDR	-	-	-	-	-	-	AIN1D	AIN0D	150
	0x00 (0x20)	Reserved	-	-	-	-	-	-	-	-	





- 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 0x00 0x1F 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.
 - Some of the status flags are cleared by writing a logical one to them. Note that, unlike most other AVRs, the CBI and SBI instructions will only operate on the specified bit, and can therefore be used on registers containing such status flags. 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 0x00 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these addresses.

ATtiny2313

Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND L	OGIC INSTRUCTIONS	3			•
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \gets Rd + Rr + C$	Z,C,N,V,H	1
ADIW	Rdl,K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	Rd ← Rd - Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd \cdot K$	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
SBIW	Rdl,K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl - K	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \bullet Rr$	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \bullet K$	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd v Rr$	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← 0x00 - Rd	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \bullet (0xFF - K)$	Z,N,V	1
INC	Rd		$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd – 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \bullet Rd$	Z,N,V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z,N,V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
BRANCH INSTRUC	1			T	
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)		None	3
RET		Subroutine Return		None	4
RETI	B15	Interrupt Return			4
CPSE	Rd,Rr	Compare, Skip if Equal	if $(Rd = Rr) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
CP	Rd,Rr	Compare	Rd – Rr	Z, N,V,C,H	1
CPC	Rd,Rr	Compare with Carry	Rd – Rr – C	Z, N,V,C,H	1
CPI	Rd,K	Compare Register with Immediate		Z, N,V,C,H	1
SBRC SBRS	Rr, b Rr, b	Skip if Bit in Register Cleared	if $(\text{Rr}(b)=0) \text{PC} \leftarrow \text{PC} + 2 \text{ or } 3$ if $(\text{Rr}(b)=1) \text{PC} \leftarrow \text{PC} + 2 \text{ or } 3$	None	1/2/3 1/2/3
	P, b	Skip if Bit in Register is Set		None	
SBIC SBIS	P, b	Skip if Bit in I/O Register Cleared Skip if Bit in I/O Register is Set	if $(P(b)=0) PC \leftarrow PC + 2 \text{ or } 3$ if $(P(b)=1) PC \leftarrow PC + 2 \text{ or } 3$	None None	1/2/3 1/2/3
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then $PC \leftarrow PC+k + 1$	None	1/2/3
BRBC	s, k	Branch if Status Flag Cleared	if (SREG(s) = 0) then $PC \leftarrow PC+k + 1$	None	1/2
BREQ	s, ĸ	Branch if Equal	if $(Z = 1)$ then PC \leftarrow PC + k + 1	None	1/2
BRNE	k	Branch if Not Equal	if (Z = 0) then PC \leftarrow PC + k + 1	None	1/2
BRCS	k	Branch if Carry Set	if (C = 1) then PC \leftarrow PC + k + 1	None	1/2
BRCC	k	Branch if Carry Cleared	if (C = 0) then PC \leftarrow PC + k + 1	None	1/2
BRSH	k	Branch if Same or Higher	if (C = 0) then PC \leftarrow PC + k + 1	None	1/2
BRLO	k	Branch if Lower	if (C = 1) then PC \leftarrow PC + k + 1	None	1/2
BRMI	k	Branch if Minus	if (N = 1) then PC \leftarrow PC + k + 1	None	1/2
BRPL	k	Branch if Plus	if (N = 0) then PC \leftarrow PC + k + 1	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if $(N \oplus V = 0)$ then PC \leftarrow PC + k + 1	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	if $(N \oplus V = 0)$ then PC \leftarrow PC + k + 1	None	1/2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then PC \leftarrow PC + k + 1	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then PC \leftarrow PC + k + 1	None	1/2
BRTS	k	Branch if T Flag Set	if (T = 1) then PC \leftarrow PC + k + 1	None	1/2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then PC \leftarrow PC + k + 1	None	1/2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then PC \leftarrow PC + k + 1	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then PC \leftarrow PC + k + 1	None	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 AND BIT-TEST	INSTRUCTIONS				
SBI	P,b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
301	1				2
CBI	P,b	Clear Bit in I/O Register	$I/O(P,b) \leftarrow 0$	None	2
	P,b Rd	Clear Bit in I/O Register Logical Shift Left	$I/O(P,b) \leftarrow 0$ Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0	Z,C,N,V	1
CBI					





Mnemonics	Operands	Description	Operation	Flags	#Clocks
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	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	Т	1
BLD	Rd, b	Bit load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	C ← 1	С	1
CLC		Clear Carry	C ← 0	С	1
SEN		Set Negative Flag	N ← 1	Ν	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
CLI		Global Interrupt Disable	← 0	1	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Twos Complement Overflow.	V ← 1	V	1
CLV		Clear Twos Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	Т	1
CLT		Clear T in SREG	T ← 0	Т	1
SEH		Set Half Carry Flag in SREG	H ← 1	Н	1
CLH		Clear Half Carry Flag in SREG	H ← 0	Н	1
DATA TRANSFER I	NSTRUCTIONS				
MOV	Rd, Rr	Move Between Registers	$Rd \leftarrow Rr$	None	1
MOVW	Rd, Rr	Copy Register Word	$Rd+1:Rd \leftarrow Rr+1:Rr$	None	1
LDI	Rd, K	Load Immediate	$Rd \leftarrow K$	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	Load Indirect and Pre-Dec.	$X \leftarrow X - 1, Rd \leftarrow (X)$	None	2
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	None	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	$Y \leftarrow Y - 1$, Rd $\leftarrow (Y)$	None	2
LDD	Rd,Y+q	Load Indirect with Displacement	$Rd \leftarrow (Y + q)$	None	2
LD	Rd, Z	Load Indirect	$Rd \leftarrow (Z)$	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	$Rd \leftarrow (Z), Z \leftarrow Z+1$	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd \leftarrow (Z + q)$	None	2
LDS	Rd, k	Load Direct from SRAM	$Rd \leftarrow (k)$	None	2
ST	X, Rr	Store Indirect	$(X) \leftarrow Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) \leftarrow \operatorname{Rr}, X \leftarrow X + 1$	None	2
ST	- X, Rr	Store Indirect and Pre-Dec.	$X \leftarrow X - 1, (X) \leftarrow Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) \leftarrow Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	None	2
ST	- Y, Rr	Store Indirect and Pre-Dec.	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	None	2
STD	Y+q,Rr	Store Indirect with Displacement	$(Y + q) \leftarrow Rr$	None	2
ST	Z, Rr	Store Indirect with Displacement	$(T + q) \leftarrow nT$ $(Z) \leftarrow Rr$	None	2
ST	Z, Rr Z+, Rr	Store Indirect and Post-Inc.	$(Z) \leftarrow Rr$ $(Z) \leftarrow Rr, Z \leftarrow Z + 1$	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	$(Z) \leftarrow \operatorname{Rr}, Z \leftarrow Z + 1$ $Z \leftarrow Z - 1, (Z) \leftarrow \operatorname{Rr}$	None	2
STD		Store Indirect and Pre-Dec. Store Indirect with Displacement		None	2
STS	Z+q,Rr k, Rr	Store Direct to SRAM	$(Z + q) \leftarrow Rr$ $(k) \leftarrow Rr$	None	2
	K, HI				3
LPM	Pd 7	Load Program Memory	$R0 \leftarrow (Z)$	None	
LPM	Rd, Z	Load Program Memory	$Rd \leftarrow (Z)$	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	$Rd \leftarrow (Z), Z \leftarrow Z + I$	None	3
SPM		Store Program Memory	(Z) ← R1:R0	None	-
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
MCU CONTROL INS	TRUCTIONS				
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1
BREAK	1	Break	For On-chip Debug Only	None	N/A

ATtiny2313

Ordering Information

Speed (MHz) ⁽³⁾	Power Supply (V)	Ordering Code ⁽⁴⁾	Package ⁽²⁾	Operation Range
10	1.8 - 5.5	ATtiny2313V-10PU ATtiny2313V-10SU ATtiny2313V-10SUR ATtiny2313V-10MU ATtiny2313V-10MUR	20P3 20S 20S 20M1 20M1	Industrial (-40°C to +85°C) ⁽¹⁾
20	2.7 - 5.5	ATtiny2313-20PU ATtiny2313-20SU ATtiny2313-20SUR ATtiny2313-20MU ATtiny2313-20MUR	20P3 20S 20S 20M1 20M1	Industrial (-40°C to +85°C) ⁽¹⁾

Notes: 1. These devices 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 alternative, complies to the European Directive for Restriction of Hazardous Substances (RoHS directive). Also Halide free and fully Green.

- 3. For Speed vs. V_{CC} see Figure 82 on page 180 and Figure 83 on page 180.
- 4. Code Indicators:

– U: matte tin

- R: tape & reel

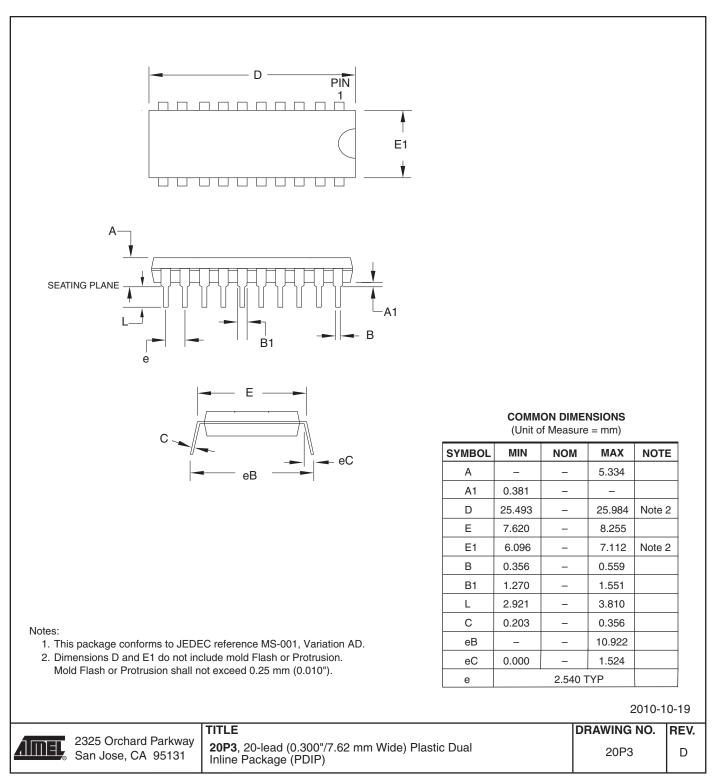
Package Type				
20P3	20-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)			
20S	20-lead, 0.300" Wide, Plastic Gull Wing Small Outline Package (SOIC)			
20M1	20-pad, 4 x 4 x 0.8 mm Body, Quad Flat No-Lead/Micro Lead Frame Package (MLF)			

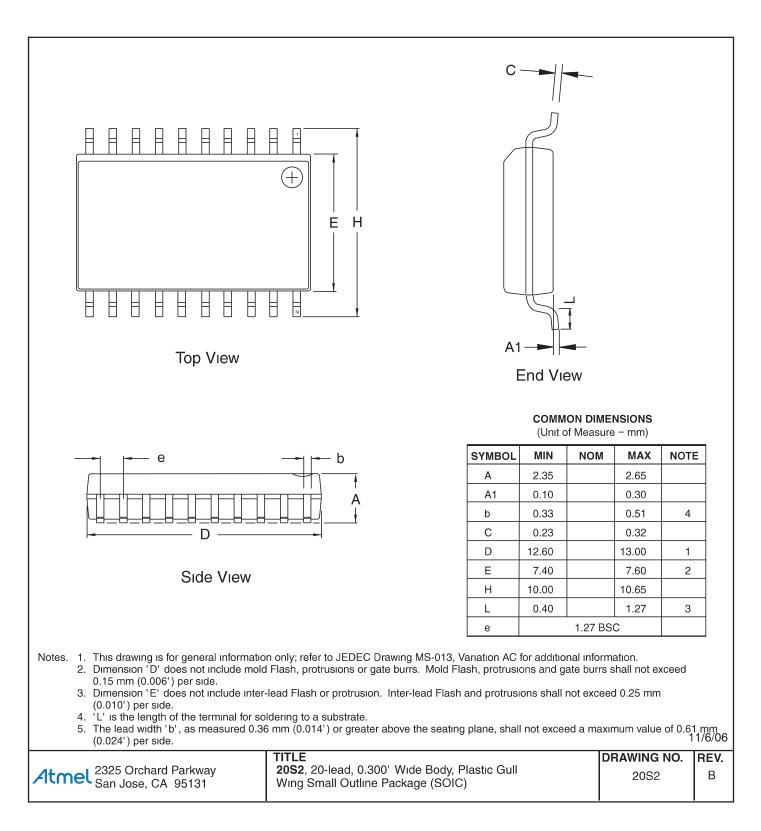




Packaging Information

20P3

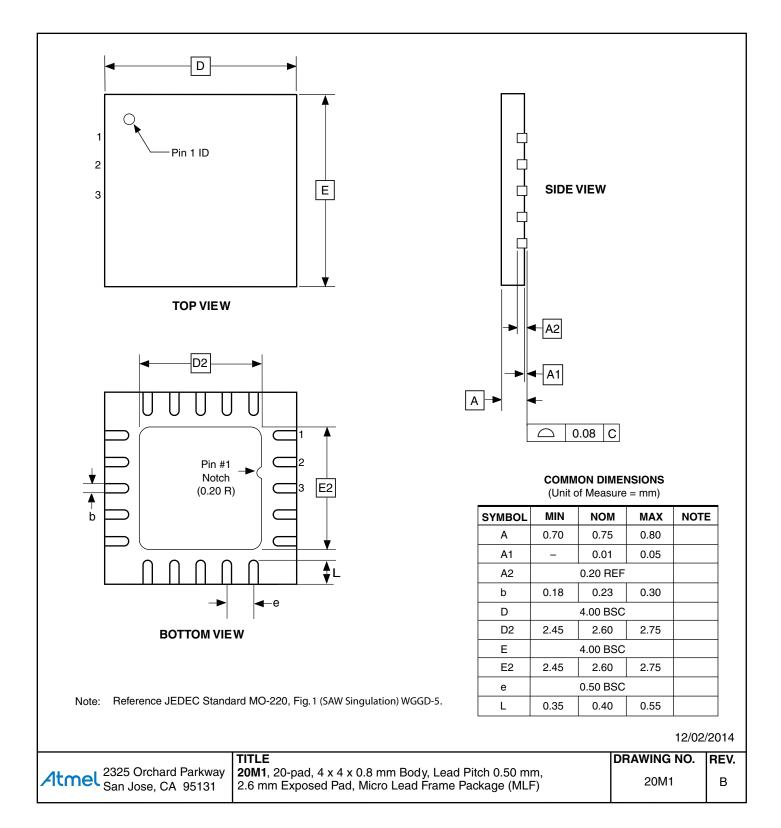








20M1



Errata The revision in this section refers to the revision of the ATtiny2313 device.

ATtiny2313 Rev C No known errata

ATtiny2313 Rev B

- Wrong values read after Erase Only operation
- Parallel Programming does not work
- Watchdog Timer Interrupt disabled
- EEPROM can not be written below 1.9 volts

1. Wrong values read after Erase Only operation

At supply voltages below 2.7 V, an EEPROM location that is erased by the Erase Only operation may read as programmed (0x00).

Problem Fix/Workaround

If it is necessary to read an EEPROM location after Erase Only, use an Atomic Write operation with 0xFF as data in order to erase a location. In any case, the Write Only operation can be used as intended. Thus no special considerations are needed as long as the erased location is not read before it is programmed.

2. Parallel Programming does not work

Parallel Programming is not functioning correctly. Because of this, reprogramming of the device is impossible if one of the following modes are selected:

- In-System Programming disabled (SPIEN unprogrammed)
- Reset Disabled (RSTDISBL programmed)

Problem Fix/Workaround

Serial Programming is still working correctly. By avoiding the two modes above, the device can be reprogrammed serially.

3. Watchdog Timer Interrupt disabled

If the watchdog timer interrupt flag is not cleared before a new timeout occurs, the watchdog will be disabled, and the interrupt flag will automatically be cleared. This is only applicable in interrupt only mode. If the Watchdog is configured to reset the device in the watchdog timeout following an interrupt, the device works correctly.

Problem fix / Workaround

Make sure there is enough time to always service the first timeout event before a new watchdog timeout occurs. This is done by selecting a long enough time-out period.

4. EEPROM can not be written below 1.9 volts

Writing the EEPROM at V_{CC} below 1.9 volts might fail.

Problem fix / Workaround

Do not write the EEPROM when V_{CC} is below 1.9 volts.

ATtiny2313 Rev A Revision A has not been sampled.





Datasheet Revision History

Refer to the complete datasheet for revision history change log.





18 ATtiny2313

2543MS-AVR-10/16





Headquarters

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

International

Atmel Asia Unit 1-5 & 16, 19/F BEA Tower, Millennium City 5 418 Kwun Tong Road Kwun Tong, Kowloon Hong Kong Tel: (852) 2245-6100 Fax: (852) 2722-1369 Atmel Europe Le Krebs 8, Rue Jean-Pierre Timbaud BP 309 78054 Saint-Quentin-en-Yvelines Cedex France Tel: (33) 1-30-60-70-00 Fax: (33) 1-30-60-71-11

Atmel Japan

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

Product Contact

Web Site www.atmel.com Technical Support avr@atmel.com Sales Contact www.atmel.com/contacts

Literature Requests www.atmel.com/literature

Disclaimer: The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. EXCEPT AS SET FORTH IN ATMEL'S TERMS AND CONDI-TIONS OF SALE LOCATED ON ATMEL'S WEB SITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDEN-TAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel's products are not intended, or warranted for use as components in applications intended to support or sustain life.

© 2016 Atmel Corporation. All rights reserved. Atmel[®], Atmel logo and combinations thereof, AVR[®] and others are registered trademarks or trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others.