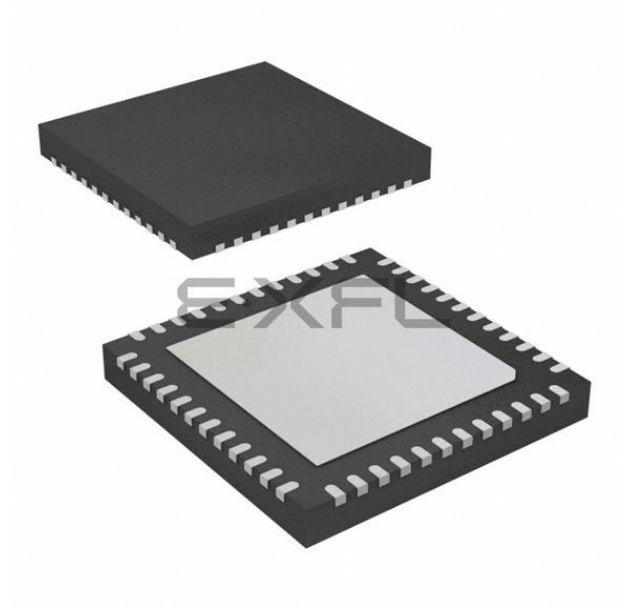


Welcome to E-XFL.COM

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 "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	16MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	23
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	512 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TC)
Mounting Type	Surface Mount
Package / Case	48-VFQFN Exposed Pad
Supplier Device Package	48-QFN (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/ata6612p-plqw

5. Electrical Characteristics (Continued)

5V < V_S < 27V, 40°C < T_{case} < 125°C, 40°C < T_j < 150°C, unless otherwise specified. All values refer to GND pins

No.	Parameters	Test Conditions	Pin	Symbol	Min.	Typ.	Max.	Unit	Type*
12	NRES Open Drain Output Pin								
12.1	Low-level output voltage	V_S 5.5V I_{NRES} = 1mA I_{NRES} = 250µA	NRES	V_{NRESL}			0.2 0.14	V V	A
12.2	Low-level output low	10k to V_{CC} V_{CC} = 0V	NRES	V_{NRESLL}			0.2	V	A
12.3	Undervoltage reset time	V_S 5.5V C_{NRES} = 20pF	NRES	t_{reset}	2	4	6	ms	A
12.4	Reset debounce time for falling edge	V_S 5.5V C_{NRES} = 20pF	NRES	t_{res_f}	1.5		10	µs	A
13	Watchdog Oscillator								
13.1	Voltage at WD_OSC in Normal Mode	I_{WD_OSC} = 200µA V_{VS} 4V	WD_OSC	V_{WD_OSC}	1.13	1.23	1.33	V	A
13.2	Possible values of resistor		WD_OSC	R_{OSC}	34		120	k	A
13.3	Oscillator period	R_{OSC} = 34k		t_{OSC}	10.65	13.3	15.97	µs	A
13.4	Oscillator period	R_{OSC} = 51k		t_{OSC}	15.68	19.6	23.52	µs	A
13.5	Oscillator period	R_{OSC} = 91k		t_{OSC}	26.83	33.5	40.24	µs	A
13.6	Oscillator period	R_{OSC} = 120k		t_{OSC}	34.2	42.8	51.4	µs	A
14	Watchdog Timing Relative to t_{OSC}								
14.1	Watchdog lead time after Reset			t_d		7895		cycles	A
14.2	Watchdog closed window			t_1		1053		cycles	A
14.3	Watchdog open window			t_2		1105		cycles	A
14.4	Watchdog reset time NRES		NRES	t_{nres}	3.2	4	4.8	ms	A
15	KL_15 Pin								
15.1	High-level input voltage R_V = 47 k	Positive edge initializes a wake-up	KL_15	V_{KL_15H}	4		V_S + 0.3V	V	A
15.2	Low-level input voltage R_V = 47 k		KL_15	V_{KL_15L}	1		+2	V	A
15.3	KL_15 pull-down current	V_S < 27V V_{KL_15} = 27V	KL_15	I_{KL_15}		50	65	µA	A
15.4	Internal debounce time	Without external capacitor	KL_15	Tdb_{KL_15}	80	160	250	µs	A
15.5	KL_15 wake-up time	R_V = 47k , C = 100nF	KL_15	Tw_{KL_15}	0.4	2	4.5	ms	C
16	WAKE Pin								
16.1	High-level input voltage		WAKE	V_{WAKEH}	V_S 1V		V_S + 0.3V	V	A
16.2	Low-level input voltage	Initializes a wake-up signal	WAKE	V_{WAKEL}	1		V_S 3.3V	V	A
16.3	WAKE pull-up current	V_S < 27V V_{WAKE} = 0V	WAKE	I_{WAKE}	30	10		µA	A

*) Type means: A = 100% tested, B = 100% correlation tested, C = Characterized on samples, D = Design parameter

6.8.7 MCU Status Register MCUSR

The MCU Status Register provides information on which reset source caused an MCU reset.

Bit	7	6	5	4	3	2	1	0	
					WDRF	BORF	EXTRF	PORF	MCUSR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	See Bit Description				

Bit 7..4: Res: Reserved Bits

These bits are unused bits in the Atmel® ATA6612/ATA6613, and will always read as zero.

Bit 3 WDRF: Watchdog System Reset Flag

This bit is set if a Watchdog System Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

Bit 2 BORF: Brown-out Reset Flag

This bit is set if a Brown-out Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

Bit 1 EXTRF: External Reset Flag

This bit is set if an External Reset occurs. The bit is reset by a Power-on Reset, or by writing a logic zero to the flag.

Bit 0 PORF: Power-on Reset Flag

This bit is set if a Power-on Reset occurs. The bit is reset only by writing a logic zero to the flag.

To make use of the Reset Flags to identify a reset condition, the user should read and then Reset the MCUSR as early as possible in the program. If the register is cleared before another reset occurs, the source of the reset can be found by examining the Reset Flags.

6.8.8 Internal Voltage Reference

Atmel ATA6612/ATA6613 features an internal bandgap reference. This reference is used for Brown-out Detection, and it can be used as an input to the Analog Comparator or the ADC.

6.8.8.1 Voltage Reference Enable Signals and Start-up Time

The voltage reference has a start-up time that may influence the way it should be used. The start-up time is given in Table 6-23 on page 74. To save power, the reference is not always turned on. The reference is on during the following situations:

1. When the BOD is enabled (by programming the BODLEVEL [2..0] Fuses).
2. When the bandgap reference is connected to the Analog Comparator (by setting the ACBG bit in ACSR).
3. When the ADC is enabled.

Thus, when the BOD is not enabled, after setting the ACBG bit or enabling the ADC, the user must always allow the reference to start up before the output from the Analog Comparator or ADC is used. To reduce power consumption in Power-down mode, the user can avoid the three conditions above to ensure that the reference is turned off before entering Power-down mode.

When the BOOTRST Fuse is unprogrammed, the Boot section size set to 2Kbytes and the IVSEL bit in the MCUCR Register is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses in Atmel fi ATA6612 is:

```
Address   Labels Code                        Comments
0x000     RESET: ldi     r16,high(RAMEND); Main program start
0x001            out     SPH,r16           ; Set Stack Pointer to top of RAM
0x002            ldi     r16,low(RAMEND)
0x003            out     SPL,r16
0x004            sei                       ; Enable interrupts
0x005            <instr> xxx
;
.org 0xC01
0xC01            rjmp    EXT_INT0          ; IRQ0 Handler
0xC02            rjmp    EXT_INT1          ; IRQ1 Handler
...              ...     ...               ;
0xC19            rjmp    SPM_RDY           ; Store Program Memory Ready Handler
```

When the BOOTRST Fuse is programmed and the Boot section size set to 2Kbytes, the most typical and general program setup for the Reset and Interrupt Vector Addresses in Atmel ATA6612 is:

```
Address   Labels Code                  Comments
.org 0x001
0x001            rjmp    EXT_INT0      ; IRQ0 Handler
0x002            rjmp    EXT_INT1      ; IRQ1 Handler
...              ...     ...           ;
0x019            rjmp    SPM_RDY       ; Store Program Memory Ready Handler
;
.org 0xC00
0xC00     RESET: ldi     r16,high(RAMEND); Main program start
0xC01            out     SPH,r16       ; Set Stack Pointer to top of RAM
0xC02            ldi     r16,low(RAMEND)
0xC03            out     SPL,r16
0xC04            sei                   ; Enable interrupts
0xC05            <instr> xxx
```

When the BOOTRST Fuse is programmed, the Boot section size set to 2Kbytes and the IVSEL bit in the MCUCR Register is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses in Atmel ATA6612 is:

```
Address   Labels  Code                    Comments
;
.org 0xC00
0xC00             rjmp    RESET           ; Reset handler
0xC01             rjmp    EXT_INT0        ; IRQ0 Handler
0xC02             rjmp    EXT_INT1        ; IRQ1 Handler
...               ...     ...             ;
0xC19             rjmp    SPM_RDY         ; Store Program Memory Ready Handler
;
0xC1A     RESET:  ldi     r16,high(RAMEND); Main program start
0xC1B             out     SPH,r16         ; Set Stack Pointer to top of RAM
0xC1C             ldi     r16,low(RAMEND)
0xC1D             out     SPL,r16
0xC1E             sei                     ; Enable interrupts
0xC1F             <instr> xxx
```

6.9.2 Interrupt Vectors in ATA6613

Table 6-28. Reset and Interrupt Vectors in ATA6613

Vector No.	Program Address (2)	Source	Interrupt Definition
1	0x0000(1)	RESET	External Pin, Power-on Reset, Brown-out Reset and Watchdog System Reset
2	0x0002	INT0	External Interrupt Request 0
3	0x0004	INT1	External Interrupt Request 1
4	0x0006	PCINT0	Pin Change Interrupt Request 0
5	0x0008	PCINT1	Pin Change Interrupt Request 1
6	0x000A	PCINT2	Pin Change Interrupt Request 2
7	0x000C	WDT	Watchdog Time-out Interrupt
8	0x000E	TIMER2 COMPA	Timer/Counter2 Compare Match A
9	0x0010	TIMER2 COMPB	Timer/Counter2 Compare Match B
10	0x0012	TIMER2 OVF	Timer/Counter2 Overflow
11	0x0014	TIMER1 CAPT	Timer/Counter1 Capture Event
12	0x0016	TIMER1 COMPA	Timer/Counter1 Compare Match A
13	0x0018	TIMER1 COMPB	Timer/Coutner1 Compare Match B
14	0x001A	TIMER1 OVF	Timer/Counter1 Overflow
15	0x001C	TIMER0 COMPA	Timer/Counter0 Compare Match A
16	0x001E	TIMER0 COMPB	Timer/Counter0 Compare Match B
17	0x0020	TIMER0 OVF	Timer/Counter0 Overflow
18	0x0022	SPI, STC	SPI Serial Transfer Complete
19	0x0024	USART, RX	USART Rx Complete
20	0x0026	USART, UDRE	USART, Data Register Empty

When the BOOTRST Fuse is unprogrammed, the Boot section size set to 2K bytes and the IVSEL bit in the MCUCR Register is set before any interrupts are enabled, the most typical and general program setup for the Reset and Interrupt Vector Addresses in ATA6613 is:

```
Address   Labels Code                    Comments
0x0000    RESET: ldi     r16,high(RAMEND); Main program start
0x0001           out     SPH,r16         ; Set Stack Pointer to top of RAM
0x0002           ldi     r16,low(RAMEND)
0x0003           out     SPL,r16
0x0004           sei                     ; Enable interrupts
0x0005           <instr> xxx
;
.org 0xC02
0x1C02           jmp     EXT_INT0        ; IRQ0 Handler
0x1C04           jmp     EXT_INT1        ; IRQ1 Handler
...              ...     ...             ;
0x1C32           jmp     SPM_RDY         ; Store Program Memory Ready Handler
```

When the BOOTRST Fuse is programmed and the Boot section size set to 2K bytes, the most typical and general program setup for the Reset and Interrupt Vector Addresses in ATA6613 is:

```
Address   Labels Code                    Comments
.org 0x0002
0x0002           jmp     EXT_INT0        ; IRQ0 Handler
0x0004           jmp     EXT_INT1        ; IRQ1 Handler
...              ...     ...             ;
0x0032           jmp     SPM_RDY         ; Store Program Memory Ready Handler
;
.org 0x1C00
0x1C00    RESET: ldi     r16,high(RAMEND); Main program start
0x1C01           out     SPH,r16         ; Set Stack Pointer to top of RAM
0x1C02           ldi     r16,low(RAMEND)
0x1C03           out     SPL,r16
0x1C04           sei                     ; Enable interrupts
0x1C05           <instr> xxx
```

The following code example shows how to set port B pins 0 and 1 high, 2 and 3 low, and define the port pins from 4 to 7 as input with pull-ups assigned to port pins 6 and 7. The resulting pin values are read back again, but as previously discussed, a nop instruction is included to be able to read back the value recently assigned to some of the pins.

Assembly Code Example[1]

```
...
;  Define pull-ups and set outputs high
;  Define directions for port pins
ldi     r16,(1<<PB7)|(1<<PB6)|(1<<PB1)|(1<<PB0)
ldi     r17,(1<<DDB3)|(1<<DDB2)|(1<<DDB1)|(1<<DDB0)
out     PORTB,r16
out     DDRB,r17
;  Insert nop for synchronization
nop
;  Read port pins
in      r16,PINB
...
```

C Code Example

```
unsigned char       i;
...
/*   Define pull-ups and set outputs high                    */
/*   Define directions for port pins                         */
PORTB = (1<<PB7)|(1<<PB6)|(1<<PB1)|(1<<PB0);
DDRB = (1<<DDB3)|(1<<DDB2)|(1<<DDB1)|(1<<DDB0);
/*   Insert nop for synchronization                          */
__no_operation();
/*   Read port pins        */
i = PINB;
...
```

Note: 1. For the assembly program, two temporary registers are used to minimize the time from pull-ups are set on pins 0, 1, 6, and 7, until the direction bits are correctly set, defining bit 2 and 3 as low and redefining bits 0 and 1 as strong high drivers.

Figure 6-61 shows the same timing data, but with the prescaler enabled.

Figure 6-61. Timer/Counter Timing Diagram, with Prescaler ($f_{clk\_I/O}/8$)

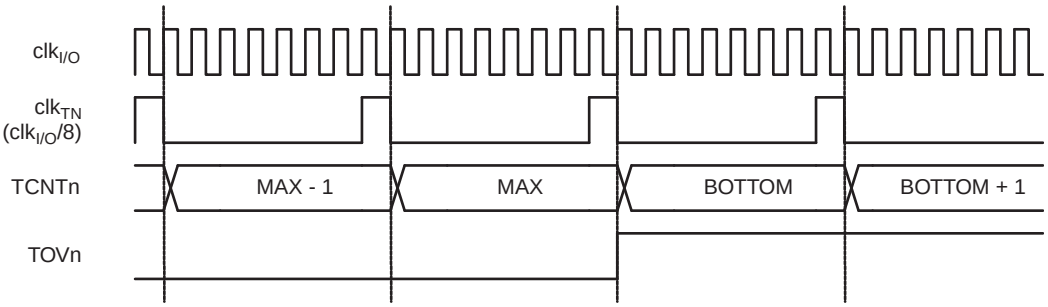


Figure 6-62 shows the setting of OCF2A in all modes except CTC mode.

Figure 6-62. Timer/Counter Timing Diagram, Setting of OCF2A, with Prescaler ($f_{clk\_I/O}/8$)

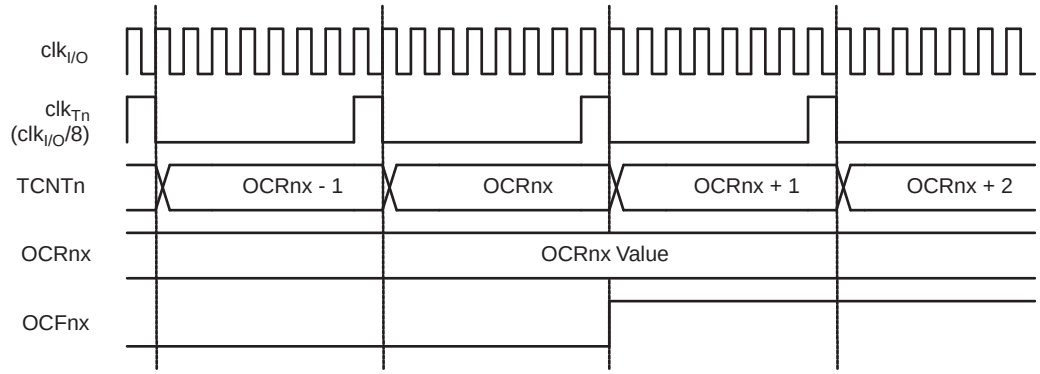
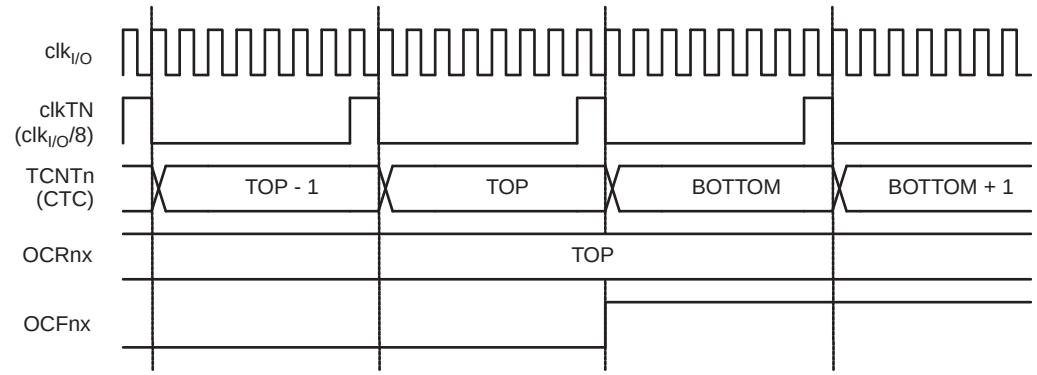


Figure 6-63 shows the setting of OCF2A and the clearing of TCNT2 in CTC mode.

Figure 6-63. Timer/Counter Timing Diagram, Clear Timer on Compare Match mode, with Prescaler ($f_{clk\_I/O}/8$)



6.22 debugWIRE On-chip Debug System

6.22.1 Features

- **Complete Program Flow Control**
- **Emulates All On-chip Functions, Both Digital and Analog, except RESET Pin**
- **Real-time Operation**
- **Symbolic Debugging Support (Both at C and Assembler Source Level, or for Other HLLs)**
- **Unlimited Number of Program Break Points (Using Software Break Points)**
- **Non-intrusive Operation**
- **Electrical Characteristics Identical to Real Device**
- **Automatic Configuration System**
- **High-Speed Operation**
- **Programming of Non-volatile Memories**

6.22.2 Overview

The debugWIRE On-chip debug system uses a One-wire, bi-directional interface to control the program flow, execute AVR® instructions in the CPU and to program the different non-volatile memories.

6.22.3 Physical Interface

When the debugWIRE Enable (DWEN) Fuse is programmed and Lock bits are unprogrammed, the debugWIRE system within the target device is activated. The RESET port pin is configured as a wire-AND (open-drain) bi-directional I/O pin with pull-up enabled and becomes the communication gateway between target and emulator.

Figure 6-113. The debugWIRE Setup

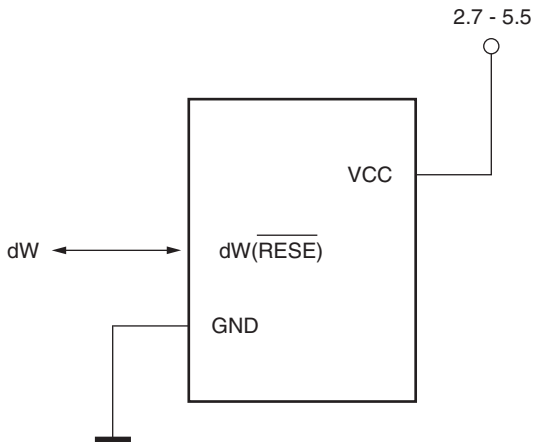


Figure 6-113 shows the schematic of a target MCU, with debugWIRE enabled, and the emulator connector. The system clock is not affected by debugWIRE and will always be the clock source selected by the CKSEL Fuses.

If the user has a fixed voltage source connected to the AREF pin, the user may not use the other reference voltage options in the application, as they will be shorted to the external voltage. If no external voltage is applied to the AREF pin, the user may switch between AV_{CC} and 1.1V as reference selection. The first ADC conversion result after switching reference voltage source may be inaccurate, and the user is advised to discard this result.

6.21.5 ADC Noise Canceler

The ADC features a noise canceler that enables conversion during sleep mode to reduce noise induced from the CPU core and other I/O peripherals. The noise canceler can be used with ADC Noise Reduction and Idle mode. To make use of this feature, the following procedure should be used:

a. Make sure that the ADC is enabled and is not busy converting. Single Conversion mode must be selected and the ADC conversion complete interrupt must be enabled.
b. Enter ADC Noise Reduction mode (or Idle mode). The ADC will start a conversion once the CPU has been halted.
c. If no other interrupts occur before the ADC conversion completes, the ADC interrupt will wake up the CPU and execute the ADC Conversion Complete interrupt routine. If another interrupt wakes up the CPU before the ADC conversion is complete, that interrupt will be executed, and an ADC Conversion Complete interrupt request will be generated when the ADC conversion completes. The CPU will remain in active mode until a new sleep command is executed.

Note that the ADC will not be automatically turned off when entering other sleep modes than Idle mode and ADC Noise Reduction mode. The user is advised to write zero to ADEN before entering such sleep modes to avoid excessive power consumption.

6.21.5.1 Analog Input Circuitry

The analog input circuitry for single ended channels is illustrated in Figure 6-107 on page 273 An analog source applied to ADCn is subjected to the pin capacitance and input leakage of that pin, regardless of whether that channel is selected as input for the ADC. When the channel is selected, the source must drive the S/H capacitor through the series resistance (combined resistance in the input path).

The ADC is optimized for analog signals with an output impedance of approximately 10 k or less. If such a source is used, the sampling time will be negligible. If a source with higher impedance is used, the sampling time will depend on how long time the source needs to charge the S/H capacitor, with can vary widely. The user is recommended to only use low impedant sources with slowly varying signals, since this minimizes the required charge transfer to the S/H capacitor.

Signal components higher than the Nyquist frequency ($f_{ADC}/2$) should not be present for either kind of channels, to avoid distortion from unpredictable signal convolution. The user is advised to remove high frequency components with a low-pass filter before applying the signals as inputs to the ADC.

Figure 6-114. Read-While-Write versus No Read-While-Write