ATmega32U4 Clock & Timers
System Clock

8.0MHz external oscillator, with a default /8 prescaler
6. System Clock and Clock Options

6.1 Clock Systems and their Distribution

Figure 6-1 presents the principal clock systems in the AVR and their distribution. All of the clocks need not be active at a given time. In order to reduce power consumption, the clocks to modules not being used can be halted by using different sleep modes, as described in “Power Management and Sleep Modes” on page 42. The clock systems are detailed below.

6.1.1 CPU Clock – clk CPU

The CPU clock is routed to parts of the system concerned with operation of the AVR core. Examples of such modules are the General Purpose Register File, the Status Register and the data memory holding the Stack Pointer. Halting the CPU clock inhibits the core from performing general operations and calculations.

6.1.2 I/O Clock – clk I/O

The I/O clock is used by the majority of the I/O modules, like Timer/Counters, SPI, and USART. The I/O clock is also used by the External Interrupt module, but note that some external interrupts are detected by asynchronous logic, allowing such interrupts to be detected even if the I/O clock is halted. Also, TWI address recognition is handled in all sleep modes.

6.1.3 Flash Clock – clk FLASH

The Flash clock controls operation of the Flash interface. The Flash clock is usually active simultaneously with the CPU clock.
System Clock

Enable Changes to Clock Prescaler

CLKPR : CLKPCE

Set Clock Prescaler

CLKPR = n

(where prescaler = / (2^ n), up to n=8)

Example

set(CLKPR,CLKPCE);  // enable prescaler changes
CLKPR = 4;          // set prescaler = 2^4 = /16
**Timer 0**

- 8-bit timer/counter
- 2 compare outputs (OC0A, OC0B)
- PWM capability
- timer overflow flag
- 6 timer modes

**Timers 1 & 3**

- 8/10/16-bit timer/counter
- 4 compare outputs (OC1A, OC1B, OC1C & OC3A)
- 2 capture inputs (IPC1 & IPC3)
- PWM capability
- timer overflow flag
- many timer modes
13. Timer/Counter0

Timer/Counter0 is a general purpose 8-bit Timer/Counter module, with two independent Output Compare Units, and with PWM support. It allows accurate program execution timing (event management) and wave generation. The main features are:

- Two Independent Output Compare Units
- Double Buffered Output Compare Registers
- Clear Timer on Compare Match (Auto Reload)
- Glitch Free, Phase Correct Pulse Width Modulator (PWM)
- Variable PWM Period
- Frequency Generator
- Three Independent Interrupt Sources (TOV0, OCF0A, and OCF0B)

13.1 Overview

A simplified block diagram of the 8-bit Timer/Counter is shown in Figure 13-1. For the actual placement of I/O pins, refer to "Pinout ATmega16U4/ATmega32U4" on page 3. CPU accessible I/O Registers, including I/O bits and I/O pins, are shown in bold. The device-specific I/O Register and bit locations are listed in the "8-bit Timer/Counter Register Description" on page 100.
Timer 0 : registers

- **TCNT0**: timer value
- **TCCR0A**: control register A
- **TCCR0B**: control register B
- **OCR0A**: compare register A
- **OCR0B**: compare register B
- **TIFR0**: interrupt flags
Timer 0 : bits

select the clock prescaler

- TCCR0B : CS02
- TCCR0B : CS01
- TCCR0B : CS00

set the timer mode

- TCCR0B : WGM02
- TCCR0A : WGM01
- TCCR0A : WGM00

set the compare options

- TCCR0A : COM0A0
- TCCR0A : COM0A1
- TCCR0B : COM0B0
- TCCR0B : COM0B1

watch the flags

- TIFR0 : OCF0A
- TIFR0 : OCF0B
- TIFR0 : TOV0
Timer 0 : modes

WGMxx

0,0,0

0,1,0

1,0,1
An interrupt can be generated each time the counter value reaches the TOP value by using the OCF0A Flag. If the interrupt is enabled, the interrupt handler routine can be used for updating the TOP value. However, changing TOP to a value close to BOTTOM when the counter is running with none or a low prescaler value must be done with care since the CTC mode does not have the double buffering feature. If the new value written to OCR0A is lower than the current value of TCNT0, the counter will miss the Compare Match. The counter will then have to count to its maximum value (0xFF) and wrap around starting at 0x00 before the Compare Match can occur.

For generating a waveform output in CTC mode, the OC0A output can be set to toggle its logical level on each Compare Match by setting the Compare Output mode bits to toggle mode (COM0A1:0 = 1). The OC0A value will not be visible on the port pin unless the data direction for the pin is set to output. The waveform generated will have a maximum frequency of $f_{OC0} = f_{clk_I/O}/2$ when OCR0A is set to zero (0x00). The waveform frequency is defined by the following equation:

$$f_{OC0} = \frac{f_{clk_I/O}}{2 \cdot (N + 1)}$$

where $N$ represents the prescaler factor (1, 8, 64, 256, or 1024).

As for the Normal mode of operation, the TOV0 Flag is set in the same timer clock cycle that the counter counts from MAX to 0x00.

### 13.6.3 Fast PWM Mode

The fast Pulse Width Modulation or fast PWM mode (WGM02:0 = 3 or 7) provides a high frequency PWM waveform generation option. The fast PWM differs from the other PWM option by its single-slope operation. The counter counts from BOTTOM to TOP then restarts from BOTTOM. TOP is defined as 0xFF when WGM2:0 = 3, and OCR0A when WGM2:0 = 7. In non-inverting Compare Output mode, the Output Compare (OC0x) is cleared on the Compare Match between TCNT0 and OCR0x, and set at BOTTOM. In inverting Compare Output mode, the output is set on Compare Match and cleared at BOTTOM. Due to the single-slope operation, the operating frequency of the fast PWM mode can be twice as high as the phase correct PWM mode that use dual-slope operation. This high frequency makes the fast PWM mode well suited for power regulation, rectification, and DAC applications. High frequency allows physically small sized external components (coils, capacitors), and therefore reduces total system cost.

In fast PWM mode, the counter is incremented until the counter value matches the TOP value. The counter is then cleared at the following timer clock cycle. The timing diagram for the fast PWM mode is shown in the figure below.
The Timer/Counter Overflow Flag (TOV0) is set each time the counter reaches TOP. If the interrupt is enabled, the interrupt handler routine can be used for updating the compare value.

In fast PWM mode, the compare unit allows generation of PWM waveforms on the OC0x pins. Setting the COM0x1:0 bits to two will produce a non-inverted PWM and an inverted PWM output can be generated by setting the COM0x1:0 to three. Setting the COM0A1:0 bits to one allows the OC0A pin to toggle on Compare Matches if the WGM02 bit is set. This option is not available for the OC0B pin (See Table 13-2 on page 101). The actual OC0x value will only be visible on the port pin if the data direction for the port pin is set as output. The PWM waveform is generated by setting (or clearing) the OC0x Register at the Compare Match between OCR0x and TCNT0, and clearing (or setting) the OC0x Register at the timer clock cycle the counter is cleared (changes from TOP to BOTTOM).

The PWM frequency for the output can be calculated by the following equation:

$$\text{f}_{\text{OC0PWM}} = \frac{\text{f}_{\text{clk\_I/O}}}{2 \times N}$$

The $N$ variable represents the prescaler factor (1, 8, 64, 256, or 1024).

The extreme values for the OCR0A Register represents special cases when generating a PWM waveform output in the fast PWM mode. If the OCR0A is set equal to BOTTOM, the output will be a narrow spike for each MAX+1 timer clock cycle. Setting the OCR0A equal to MAX will result in a constantly high or low output (depending on the polarity of the output set by the COM0A1:0 bits.)

A frequency (with 50% duty cycle) waveform output in fast PWM mode can be achieved by setting OC0x to toggle its logical level on each Compare Match (COM0x1:0 = 1). The waveform generated will have a maximum frequency of $\text{f}_{\text{OC0}} = \frac{\text{f}_{\text{clk\_I/O}}}{2}$ when OCR0A is set to zero.
The Timer/Counter Overflow Flag (TOV0) is set each time the counter reaches BOTTOM. The Interrupt Flag can be used to generate an interrupt each time the counter reaches the BOTTOM value.

In phase correct PWM mode, the compare unit allows generation of PWM waveforms on the OC0x pins. Setting the COM0x1:0 bits to two will produce a non-inverted PWM. An inverted PWM output can be generated by setting the COM0x1:0 to three: Setting the COM0A0 bits to

Figure 13-7. Phase Correct PWM Mode, Timing Diagram

© 2009, Jonathan Fiene, University of Pennsylvania
Timer 0: example

```c
set(DDRD,0);                  // D0 as output for OCR0B
OCR0B = 0x84;
set(TCCR0B,COM0B1);           // clear at OCR0B, set at 0xFF
set(TCCR0B,CS00);             // set prescaler to /1
```