PWM NOTES

24 Aralık 2024 Salı 17:10

Introduction

* PWM enables the creation of intermediate voltage levels through rapid toggling between high (ON) & low (OFF) states.

Key formulas

$$\text{Duty Cycle} = \frac{\text{On\_time}}{(\text{On\_time} + \text{Off\_time})} \cdot 100$$

$$\text{Output Voltage} = \text{Duty-Cycle} \cdot \text{Input-Voltage}$$

PWM Modes

* There are 3 different modes

1- Fast PWM → works the same way as the normal counter.

a. The control logic receives the clock signal and increments TCNTn register.

b. When a match is detected, the OCFnx flag is set & signal is send to the Waveform Generator.

c. The Waveform Generator then changes the state of the OCnx pin (the state is determined by the selected mode)

d. When the TCNTn register passes the TOP value (0xFF/OCRnA) it simply overflows (or overruns) back to 0, at the same time the OCFnx flag is set.

e. The OCFnx flag can be configured to trigger an interrupt.

f. The OCFnx flag can be cleared by software, but as always is cleared automatically when an interrupt request is triggered.

$$\text{PWM\_frequency} = \frac{\text{clock\_speed}}{\text{prescaller\_value} \cdot (1 + \text{TOP\_Value})}$$

Usage: DAC, fading LEDs, Power regulation

2- Phase Corrected PWM

a. It counts up until it hits the TOP value (FIXED, OCRnA /ICRn) then starts to count down until it hits the BOTTOM (0).

b. The control logic receives the clock signal & increments the TCNTn register.

c. When a match is detected the OCFnx flag is set & signal is send to Waveform Generator.

d. The Waveform Generator then changes the state of the OCnx pin (the state is determined by the selected mode).

e. When the TCNTn register hits the TOP value (FIXED, OCRnA/ICRn) the OCFnx flag is set.

f. The OCFnx flag can be configured to trigger an interrupt. The OCFnx flag can be cleared by software, but as always is cleared automatically when an interrupt request is triggered.

This mode can be inverted or none-inverted:

* In none-inverting mode, the OCn pin is LOW(GND) on the Compare Match btw TCNTn & OCRnx while up-counting, & HIGH (VCC) on the Compare Match while down-counting.

* In inverting mode, the OCn pin is HIGH (VCC) on the Compare Match btw TCNTn & OCRnx while up-counting, & LOW(GND) on the Compare Match while down-counting.

$$PWM\_frequency = \frac{clock\_speed}{2.prescaller\_value \cdot TOP\_Value}$$

Usage: motor control.

3- Phase & Frequency Corrected PWM → works the same way as 2nd ↑.

* Works the same way as the Phase Corrected version if we are not planning on changing our TOP value once the PWM mode is started.

* The difference: PFC PWM mode updates its TOP value when it hits BOTTOM while PC PWM updates its TOP value when it hits the TOP.

$$PWM\_frequency = \frac{clock\_speed}{2.prescaller\_value \cdot TOP\_value}$$

Usage: motor control.