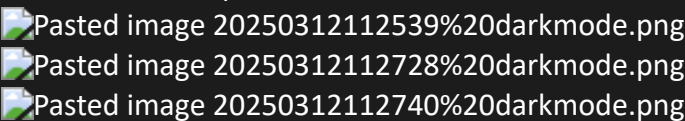


Таймер 0 МК ATmega8515

8-битный таймер/счётчик

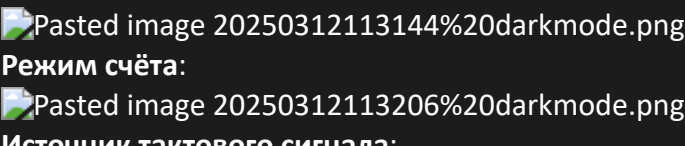


$$F_{cnt} = \frac{F_{cnt}}{K} = \frac{1}{T_{cnt}}$$
$$t = (N_{max} - N_0) T_{cnt} = (2^n - N_0) \frac{K}{F_{ck}}$$
$$N_0 = 2^n - \frac{tF_{ck}}{K}$$

- Отсчёт заданного интервала

- Начальное значение

Регистр управления таймера TCCR0



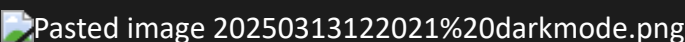
Режим счёта:



Источник тактового сигнала:

CS02	CS01	CS00	Desc.
0	0	0	No clock source (Timer/counter stopped)
0	0	1	$clk_{IO}/$ (No prescaling)
0	1	0	$clk_{IO}/8$ (From prescaler)
0	1	1	$clk_{IO}/64$ (From prescaler)
1	0	0	$clk_{IO}/256$ (From prescaler)
1	0	1	$clk_{IO}/1024$ (From prescaler)
1	1	0	External clock source on T0 pin. Clock on <b>falling</b> edge //negedge essentially
1	1	1	External clock source on T0 pin. Clock on <b>rising</b> edge //posedge essentially

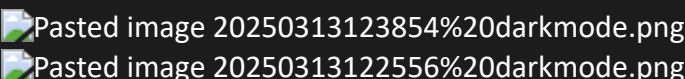
Функция сравнения при нормальном режиме счёта



COM01	COM00	Desc.
0	0	Normal port operation, OC0 disconnected
0	1	<b>Toggle OC0 on Compare Match</b>
1	0	Clear OC0 on Compare Match
1	1	Set OC0 on Compare Match

$$T = 2t_{и} = 2 * 2^n \frac{K}{F_{ck}}$$

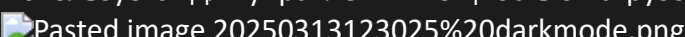
Режим счёта "сброс при совпадении" (Clear Timer on Compare Match)



$$T = 2t_{и} = 2 * (OCRn + 1) \frac{K}{F_{ck}}$$

ШИМ-сигналы (Широтно-Импульсная Модуляция)

... - aka Pulse-Width Modulation (PWM) - представление аналогового сигнала через цифровой с задаваемой шириной импульсной. В частном случае: периодическим сигналом с заданным коэффициентом заполнения. Используется для управления мощностью нагрузок: яркость лампы, скорость вращения двигателя и т.д.

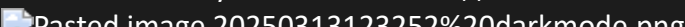


Коэффициент заполнения  $= \frac{t_{и}}{T}$

Скважность  $= \frac{T}{t_{и}}$

Режим счёта "ШИМ" (Phase Correct PWM)

Позволяет получать сигналы с заданной скважностью

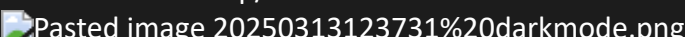


$$T = 2TOP \frac{K}{F_{ck}}$$
$$t_{и} = 2OCRn \frac{K}{F_{ck}}$$

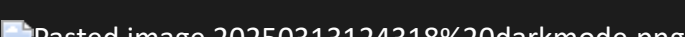
COM01	COM00	Desc.
0	0	Normal port connection, OC0 disconnected
0	1	Reserved
1	0	<b>Clear OC0 on Compare Match when up-counting</b> <b>Set OC0 on Compare when down-counting</b>
1	1	Set OC0 on Compare Match when up-counting Clear OC0 on Compare Match when down-counting

Таймер 1 МК ATmega8515

16-битный таймер/счётчик



Особенности работы



- При чтении младшего байта TCNTL старший TCNTH копируется в скрытый регистр TEMP
  - При записи младшего байта TCNTL одновременно копируется значение TEMP в старший байт TCNTH
- СЛЕДОВАТЕЛЬНО**
- При записи первым записываем старший байт
  - При чтении первым читаем младший байт
  - Запрещаем прерывания при чтении/записи иначе пизда

Ниже - код чтения/записи

```
; Set TCNT1 -> 0x01FF
ldi r17, 0x01
ldi r16, 0xFF
out TCNT1H, r17
out TCNT1L, r16
```

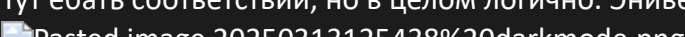
```
; Read TCNT1 -> r17:r16
in r16, TCNT1L
in r17, TCNT1H
```

Схема захвата



Режимы счёта

Тут ебать соответствий, но в целом логично. Энивей не думаю что её запоминать придётся лол



Watchdog AVR (он же сторожевой таймер)

**Watchdog Timer** нужен для предотвращения заикливаний

В общем случае когда всё нормально работает программа через команду WDR периодически его сбрасывает

При переполнении - сброс всего МК (ака мы зависли нахуй). В некоторых МК вместо сброса отсылается прерываение

Для настройки таймера юзают FUSE биты и управляющий регистр

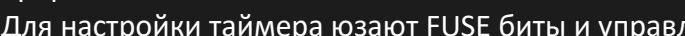
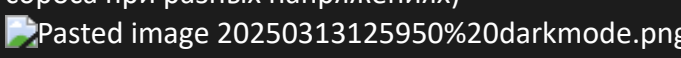


Таблица ниже - соответствие значений в WDP и количества тиков Watchdog'a (а соответственно и времени

сброса при разных напряжениях)

Pasted image 20250313125950%20darkmode.png