

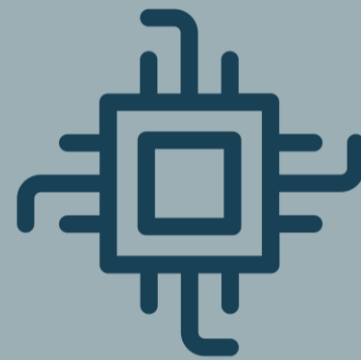


Universidade Federal
de Campina Grande



TIMERS

ATMEGA328P



Prof.

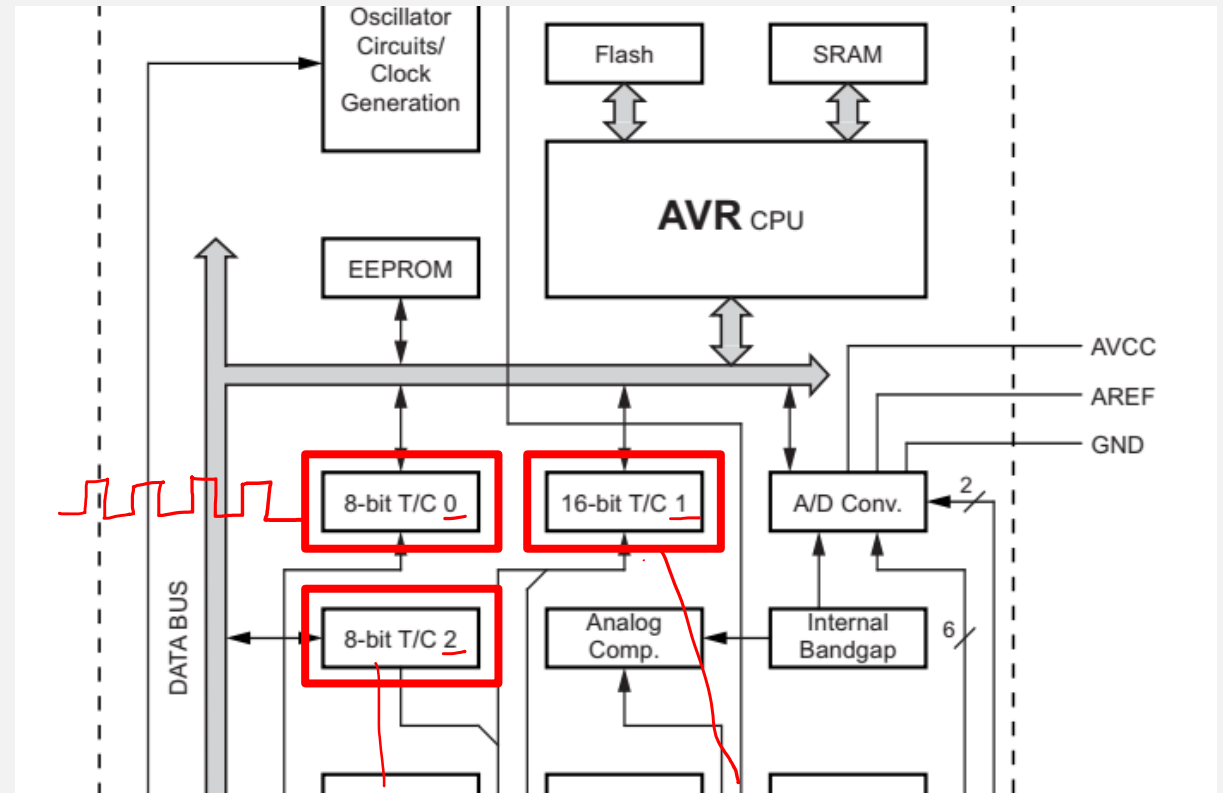
**Rafael
Lima**

TIMERS

Timers/Counters

São circuitos periféricos de contagem, que realizam essa tarefa de forma independente da CPU.

- Geração de sinais periódicos ✓
- Geração de eventos síncronos ✓
- Contagem de eventos ✓
- Funções de captura e comparação ✓



$0 - (2^8 - 1)$ $0 - 65535$
 $0 - 255$
x 0 0 0 0 0 0 0 0

TIMERS DE 8BITS – ATMEGA328P

Timer/Counter 0 (TC0)

- 8 bits (permite contagens de 0 até 255). ✓
- Fonte de clock interna ou externa.
- Divisor do *clock* para o contador de até 10 bits (*prescaler*).
- 2 comparadores independentes ✓
- Gerador para 2 sinais PWM (pinos OC0A e OC0B).
- Gerador de frequência (onda quadrada). ✓
- 3 fontes independentes de interrupção: Por estouro (TOV0) e igualdades de comparação OCF0A/OCF0B

Timer/Counter 2 (TC2)

- Características similares ao TC0
- Função adicional: Contagem precisa de 1s através de um clock externo de 32,768 kHz
- Gerador para 2 sinais PWM (pinos OC2A e OC2B).

TIMER DE 16BITS – ATMEGA328P

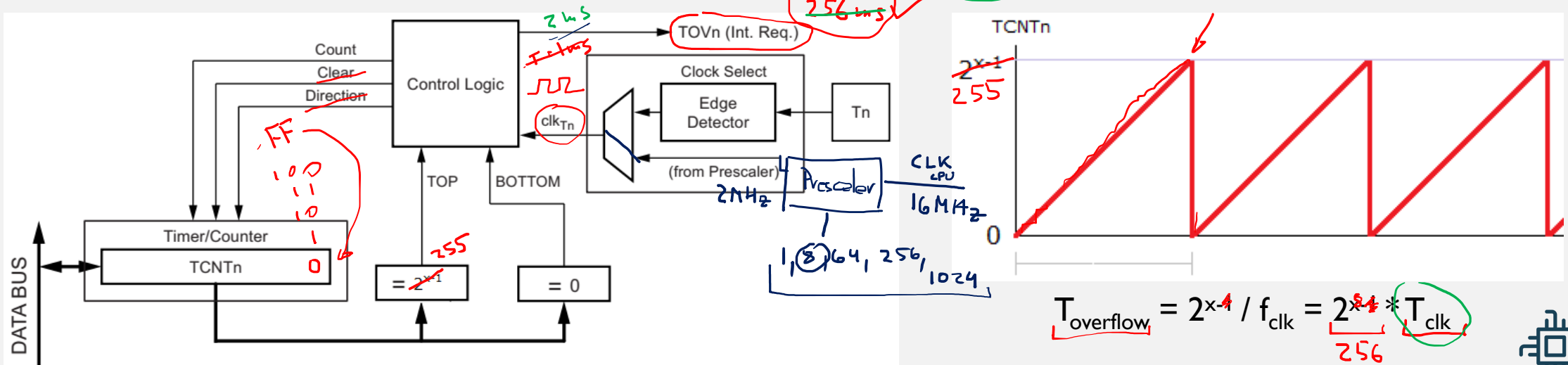
Timer/Counter 1 (TC1)

- 16 bits (permite contagens de 0 até 65535). ✓
- Fonte de *clock* interna ou externa.
- Divisor do *clock* para o contador de até 10 bits (*prescaler*).
- Gerador para 2 sinais PWM (pinos OC1A e OC1B) com inúmeras possibilidades de configuração.
- Gerador de frequência (onda quadrada).
- 4 fontes independentes de interrupção (por estouro e igualdades de comparação).

MODOS DE OPERAÇÃO

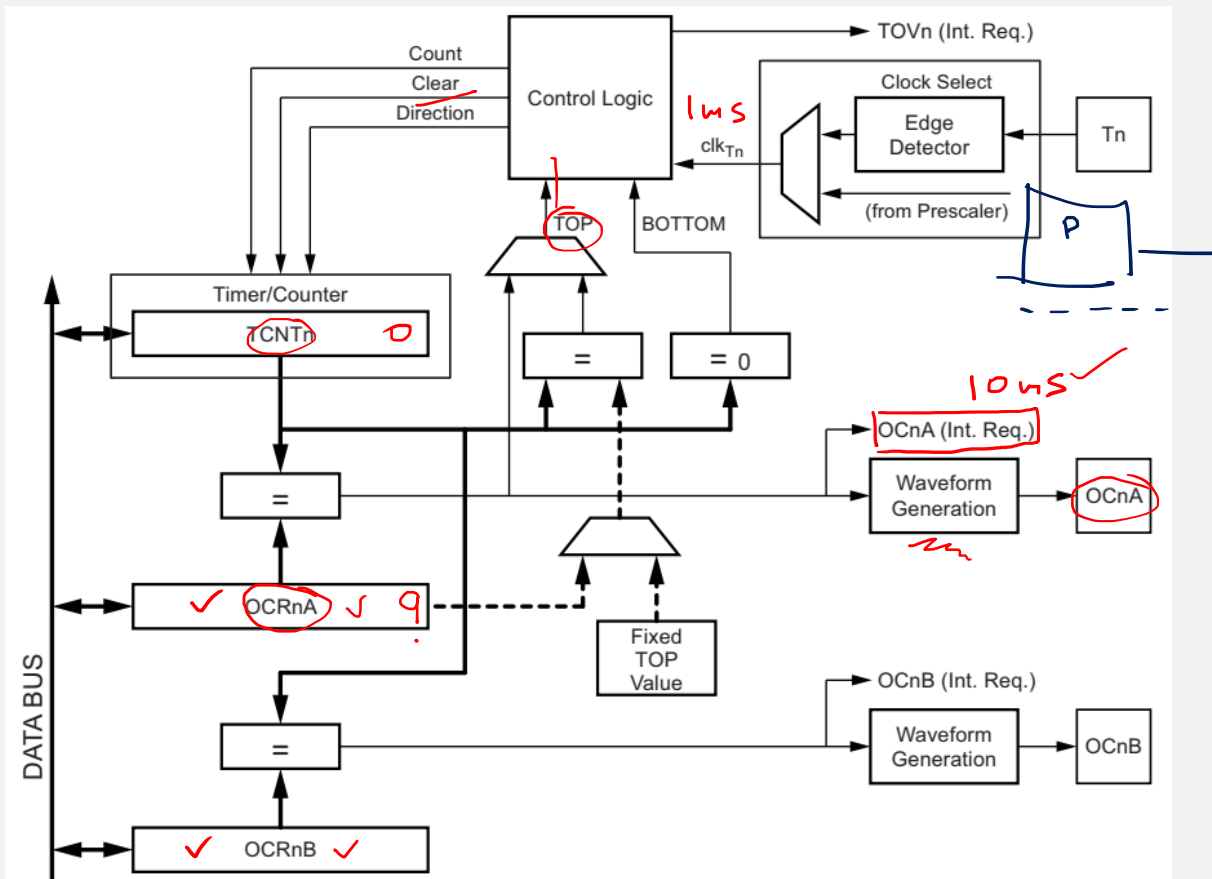
Modo NORMAL

- O TCn conta continuamente de forma crescente.
- A contagem se dá de 0 até $2^8 - 1$ voltando a 0, num ciclo contínuo. ²⁵⁵
- O valor da contagem é armazenado no registrador TCNTn
- Quando a contagem estoura, o bit sinalizador de estouro (TOVn) é colocado em 1. Se habilitada, uma interrupção é gerada (TIMERn OVF).

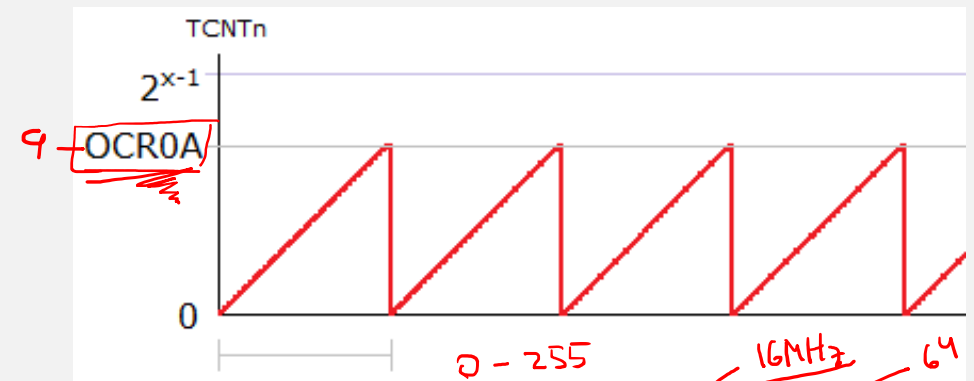


MODOS DE OPERAÇÃO

Modo CTC (Clear Timer on Compare)



- O contador é zerado quando o valor de TCNTn é igual a OCRnA
- Uma interrupção pode ser gerada cada vez que o contador atinge o valor de comparação (OCRnA ou OCRnB)
- Permite geração de ondas quadradas nos pinos OCnA e OCnB



$$T_{OCRnA} = (OCRnA + 1) / f_{clk}$$

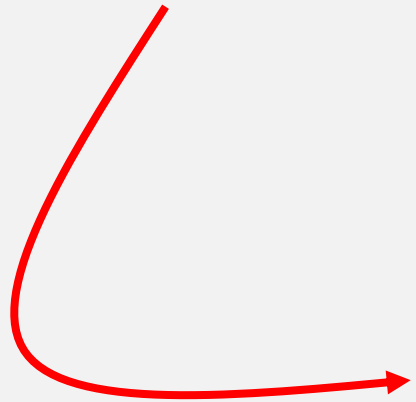
$$= (OCRnA + 1) * T_{clk}$$

MODOS DE OPERAÇÃO

Modo PWM rápido ✓

Modo PWM com fase corrigida ✓

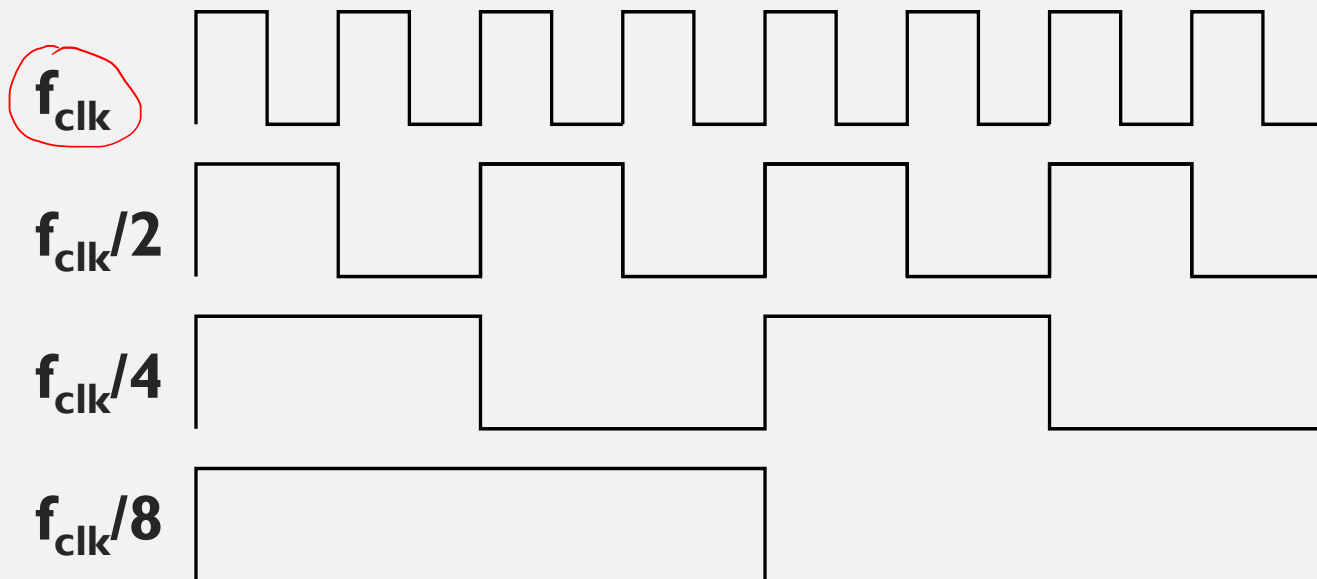
Modo captura de entrada ✓



**Serão vistos na
próxima aula**

PRESCALER - TIMER

Prescaler: divisor de frequência



Prescaler dos **TIMERn**

- ✓ $clock/1$ (prescaler=1)
- ✓ $clock/8$ (prescaler = 8).
- ✓ $clock/64$ (prescaler = 64).
- ✓ $clock/256$ (prescaler = 256).
- ✓ $clock/1024$ (prescaler = 1024).

REGISTRADORES DOS TIMERS/COUNTERS

TCCRnA (Timer/Counter Control n Register A)

- Controle do Modo de operação

Bit	7	6	5	4	3	2	1	0
	TCCR0A							
	COM0A1	COM0A0	COM0B1	COM0B0	-	-	WGM01	WGM00
Lê/Escr.	L/E	L/E	L/E	L/E	L	L	L/E	L/E
Valor Inic.	0	0	0	0	0	0	0	0

Modo	WGM02	WGM01	WGM00	Modo de Operação TC	TOP	Atualização de OCR0A no valor:	Sinalização do bit TOV0 no valor:
0	0	0	0	Normal ✓	0xFF	Imediata	0xFF
1	0	0	1	PWM com fase corrigida	0xFF	0xFF	0x00
2	0	1	0	CTC ✓	OCR0A	Imediata	0xFF
3	0	1	1	PWM rápido	0xFF	0x00	0xFF
4	1	0	0	Reservado	-	-	-
5	1	0	1	PWM com fase corrigida	OCR0A	OCR0A	0x00
6	1	1	0	Reservado	-	-	-
7	1	1	1	PWM rápido	OCR0A	0x00	OCR0A

- P/ modo CTC:

COM0A1	COM0A0	Descrição
0	0	Operação normal do pino, OC0A desconectado.
0	1	Mudança do estado de OC0A na igualdade de comparação.
1	0	OC0A é limpo na igualdade de comparação.
1	1	OC0A é ativo na igualdade de comparação.

REGISTRADORES DOS TIMERS/COUNTERS

TCCRnB (Timer/Counter Control n Register B)

Bit	7	6	5	4	3	2	1	0
	FOC0A	FOC0B	-	-	WGM02	CS02	CS01	CS00
Lê/Escr.	E	E	L	L	L/E	L/E	L/E	L/E
Valor Inic.	0	0	0	0	0	0	0	0

- **FOC0A:B** – Force Output Compare A e B, quando modo não-PWM, força uma comparação no modulo gerador de onda
- **WGM02** – Tabela anterior
- **CS02:0** – Seleção de clock

CS02	CS01	CS00	Descrição
0	0	0	Sem fonte de clock (TC0 parado).
0	0	1	clock/1 (prescaler=1) - sem prescaler.
0	1	0	clock/8 (prescaler = 8).
0	1	1	clock/64 (prescaler = 64).
1	0	0	clock/256 (prescaler = 256).
1	0	1	clock/1024 (prescaler = 1024).
1	1	0	clock externo no pino T0. Contagem na borda de descida. ✓
1	1	1	clock externo no pino T0. Contagem na borda de subida. ✓

REGISTRADORES DOS TIMERS/COUNTERS

TCNTn (Timer/Counter n Register)

- Registrador onde é realizada a contagem do TCn, pode ser lido ou escrito a qualquer tempo.

OCRnA (Output Compare n Register A)

- Registrador de comparação A, possui o valor que é continuamente comparado com o valor do contador (TCNTn). A igualdade pode ser utilizada para gerar uma interrupção ou uma forma de onda no pino OCnA.

OCRnB (Output Compare n Register B)

- Registrador de comparação B, possui o valor que é continuamente comparado com o valor do contador (TCNTn). A igualdade pode ser utilizada para gerar uma interrupção ou uma forma de onda no pino OCnB.

REGISTRADORES DOS TIMERS/COUNTERS

TIMSKn (Timer/Counter n Interrupt Mask Register)

Bit	7	6	5	4	3	2	1	0
TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0
Lê/Escreve	L	L	L	L	L	L/E	L/E	L/E
Valor Inicial	0	0	0	0	0	0	0	0

OCIE_{nB} (Timer/Counter n Output Compare Match B Interrupt Enable)

- A escrita 1 neste bit ativa a interrupção do TC_n na igualdade de comparação com o registrador OCR_{nB}.

OCIE_{nA} (Timer/Counter n Output Compare Match A Interrupt Enable)

- A escrita 1 neste bit ativa a interrupção do TC_n na igualdade de comparação com o registrador OCR_{nA}.

TOIE_n (Timer/Counter n Overflow Interrupt Enable) ✓

- A escrita 1 neste bit ativa a interrupção por estouro do TC_n

REGISTRADORES DOS TIMERS/COUNTERS

TIFRn (Timer/Counter n Interrupt Flag Register)

Bit	7	6	5	4	3	2	1	0
TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0
Lê/Escreve	L	L	L	L	L	L/E	L/E	L/E
Valor Inicial	0	0	0	0	0	0	0	0

- Os flags são ativados quando os respectivos eventos ocorrem

EXEMPLO: INTERRUPÇÃO OVERFLOW DO TC0

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <avr/interrupt.h>
```

```
ISR(TIMERO_OVF_vect) //interrupção do TC0
```

```
{
    PORTB ^= 0b00100000; //Inverte o estado do PB5
}
```

```
int main()
```

```
{
    ✓ DDRB = 0b00100000; //habilita PB5 como saída e demais pinos como entrada
    ✓ PORTB = 0b11011111; //apaga PB5 e habilita pull-ups nos pinos não utilizados
    ✓ TCCR0A = 0b00000000; //TC0 em operação normal
    ✓ TCCR0B = 0b00000001; //TC0 sem prescaler. Overflow a cada 16us = 256/16MHz * 8 = 128µs
    ✓ TIMSK0 = 0b00000001; //habilita a interrupção por estouro do TC0
    ✓ sei(); //habilita a chave de interrupção global
    while(1)
```

```
    //a cada estouro do TC0 o programa desvia para ISR(TIMERO_OVF_vect)
}
```



+1024

+128

.64

8*

0,0625µs

EXEMPLO: INTERRUPTÃO CTC DO TC0

```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <avr/interrupt.h>
```

```
ISR(TIMERO0_COMPA_vect) //interrupção do TC0 a cada 1ms = (64*(249+1))/16MHz
```

```
{
    PORTB ^= 0b00100000; //Inverte o estado do PB5
}
```

```
int main(void)
{
```

```
    DDRB = 0b00100000; //habilita PB5 como saída e demais pinos como entrada
    PORTB = 0b11011111; //apaga PB5 e habilita pull-ups nos pinos não utilizados
```

```
    TCCR0A = 0b00000010; //habilita modo CTC do TC0
```

```
    TCCR0B = 0b00000011; //liga TC0 com prescaler = 64
```

```
    OCR0A = 249; //ajusta o comparador para o TC0 contar até 249
```

```
    TIMSK0 = 0b00000010; //habilita a interrupção na igualdade de comparação com OCR0A. A
    interrupção ocorre a cada 1ms = (64*(249+1))/16MHz
```

```
    sei();
```

```
    while(1)
```

```
    {
        //
    }
```

```
}
```

REFERÊNCIAS

IDE

- Atmel Studio 7 (gratuito) <https://www.microchip.com/mplab/avr-support/atmel-studio-7>

Simuladores

- <https://www.simulide.com/p/blog-page.html>
- <https://github.com/lcgamboa/picsimlab/releases>
- <https://www.labcenter.com/downloads/>

Material de referência:

- Datasheet do Atmega 328p: <https://www.microchip.com/wwwproducts/en/ATmega328p#datasheet-toggle>
- Livro texto: <http://borgescorporation.blogspot.com/2012/05/avr-e-arduino-tecnicas-de-projeto.html>