

Temporizador/Contador

ATmega328p

Prof. Roberto de Matos

roberto.matos@ifsc.edu.br



**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Objetivos

Objetivo

- Entender o funcionamento básico do periférico Temporizador/Contador (TC)
- Entender como programar utilizando as interrupções do T/C

Referências:

- Datasheet do ATmega328p
- Manual online do montador
- Manual do conjunto de instruções do AVR



Motivação

Como contar tempo com precisão?

■ Contagem do número de ciclos:

loop:

```
dec r16  
brne loop  
dec r17  
brne loop
```

$$\left[\begin{array}{c} 1 \text{ ciclo} \\ + 2 \text{ ciclos} \end{array} \right] \times 255 + \left[\begin{array}{c} 1 \text{ ciclo} \\ + 1 \text{ ciclo} \end{array} \right] = \left[\begin{array}{c} 767 \text{ ciclos} \\ + 1 \text{ ciclo} \\ + 2 \text{ ciclos} \end{array} \right] \times 255 + \left[\begin{array}{c} 767 \text{ ciclos} \\ + 1 \text{ ciclo} \\ + 1 \text{ ciclo} \end{array} \right] = 197119 \text{ ciclos}$$

■ Problemas:

- Difícil (quase impossível) de ajustar com precisão
- Susceptível a interferências (interrupção)
- Bloqueante (espera ocupada)
- Gasto de energia de forma desnecessária

Solução

Usar um periférico para contar com precisão



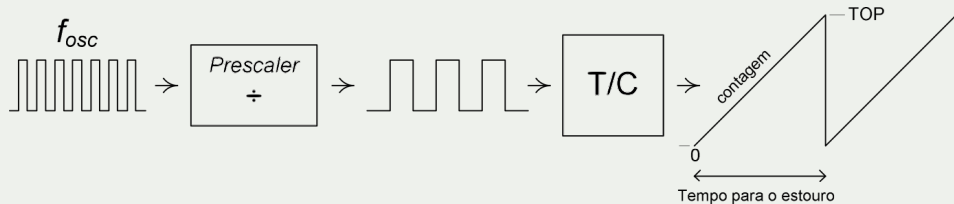
Temporizadores/Contadores

Características ATmega328p

- Dois T/C de 8 bits (TC0 e TC2) e
- Um T/C 16 bits (TC1)
- Todos independentes e com características próprias
- Funcionamento básico (encontrado na maioria dos MCUs):
 - Contador de clock (temporizador)
 - Contador de eventos externos
 - Interrupção por estouro ou comparação
- Funcionamento avançado:
 - PWM
 - Gerador de frequência (onda quadrada)



Temporizador/Contador 0



$$t_{estouro} = \frac{(TOP + 1) \times prescaler}{f_{osc}}$$

■ Sendo:

- Prescaler = 1, 8, 64, 256 ou 1024
- TOP = 255 (8 bits)



- Dessa forma, com um $f_{osc} = 16 \text{ MHz}$, é possível programar o estouro e, conseqüentemente, a interrupção para os seguintes intervalos:

$$t_{estouro} = \frac{(255 + 1) \times \mathbf{1}}{16 \text{ MHz}} = 16 \mu\text{s}$$

$$t_{estouro} = \frac{(255 + 1) \times \mathbf{8}}{16 \text{ MHz}} = 128 \mu\text{s}$$

$$t_{estouro} = \frac{(255 + 1) \times \mathbf{64}}{16 \text{ MHz}} = 1,024 \text{ ms}$$

$$t_{estouro} = \frac{(255 + 1) \times \mathbf{256}}{16 \text{ MHz}} = 4,096 \text{ ms}$$

$$t_{estouro} = \frac{(255 + 1) \times \mathbf{1024}}{16 \text{ MHz}} = 16,384 \text{ ms}$$



Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(0x6E)	TIMSK0	–	–	–	–	–	OCIE0B	OCIE0A	TOIE0
• • •									
0x28 (0x48)	OCR0B	Timer/Counter0 Output Compare Register B							
0x27 (0x47)	OCR0A	Timer/Counter0 Output Compare Register A							
0x26 (0x46)	TCNT0	Timer/Counter0 (8-bit)							
0x25 (0x45)	TCCR0B	FOC0A	FOC0B	–	–	WGM02	CS02	CS01	CS00
0x24 (0x44)	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	–	–	WGM01	WGM00
• • •									
0x15 (0x35)	TIFR0	–	–	–	–	–	OCF0B	OCF0A	TOV0



TCCR0B – Timer/Counter Control Register B

Bit	7	6	5	4	3	2	1	0	
0x25 (0x45)	FOC0A	FOC0B	–	–	WGM02	CS02	CS01	CS00	TCCR0B
Read/Write	W	W	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

CS02	CS01	CS00	Descrição
0	0	0	sem fonte de clock
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 subida.
1	1	1	clock externo no pino T0. Contagem na borda de subida.

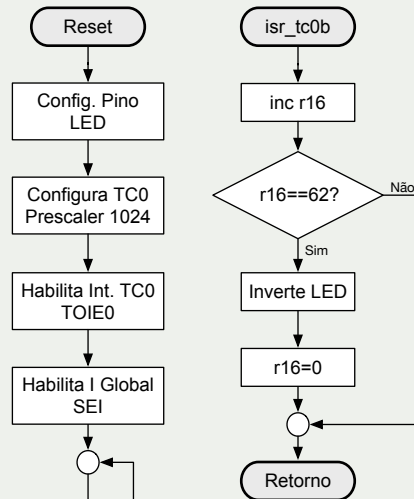
TIMSK0 – Timer/Counter Interrupt Mask Register

Bit	7	6	5	4	3	2	1	0	
(0x6E)	–	–	–	–	–	OCIE0B	OCIE0A	TOIE0	TIMSK0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



Exemplo: Pisca led

```
1 .ORG 0x0000      ; Vetor reset
2   RJMP setup
3
4 .ORG 0x0020      ; Vetor (endereço na Flash) do estouro do T/C0
5   RJMP isr_tc0b
6
7 setup:
8   SBI DDRB, PB5
9   CBI PORTB, PB5      ; desliga led
10
11   LDI R16, 0b00000101 ;TC0 com prescaler de 1024, a 16 MHz gera
12   OUT TCCR0B, R16      ; uma interrupção a cada 16,384 ms
13   LDI R16, 1
14   STS TIMSK0, R16      ; habilita int. do TCOB (TIMSK0(0)=TOIE0 <- 1)
15
16   SEI                  ; habilita as interrupções globais
17
18 main:
19   RJMP main
20
21 isr_tc0b:
22   INC r16
23   CPI r16, 62
24   BRNE fim
25   SBI PINB, PB5        ; inverte o estado do LED depois de entrar 62 vezes
26                        ; na interrupção: 62 * 16,384 ms = 1.015,81ms
27   LDI r16,0
28 fim:
29   RETI
```



Experimentos

- Testar e gravar todos os trechos de códigos apresentados.
- Fazer atividade proposta.



Fim!