



# ***Timer/Counter II***



# *Roteiro*

- Revisão
- Programação:
  - Modo 0
  - Modo 1
  - Modo 2
  - Modo 3
  - Exemplos
- Referências...

## *Registradores Especiais*

**TH0**

8Ch

**TL0**

8Ah

**TH1**

8Dh

**TL1**

8Bh

# Revisão

## Registradores Especiais

**TMOD**

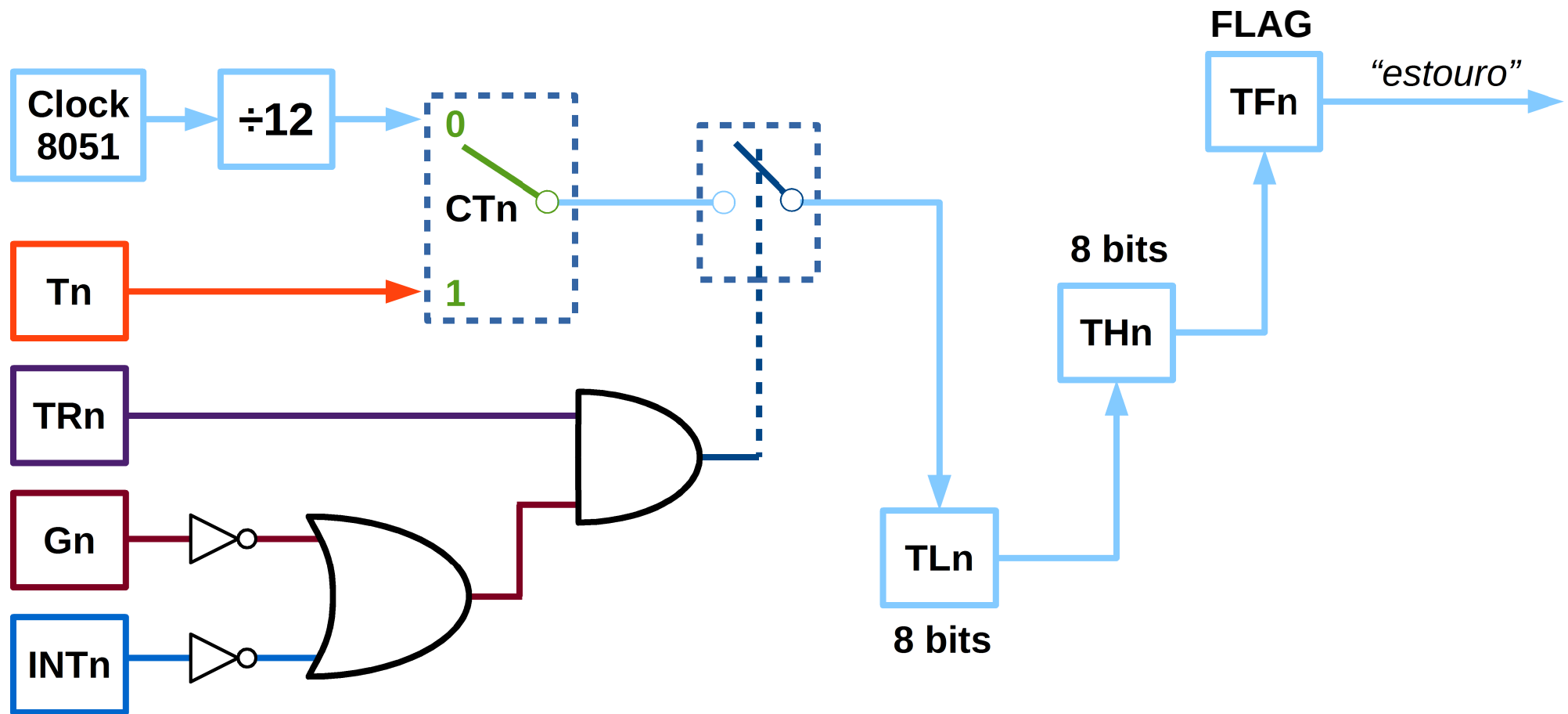
89h							
G1	CT1	M11	M10	G0	CT0	M01	M00

**TCON**

8Fh	8Eh	8Dh	8Ch	*	*	*	*
TF1	TR1	TF0	TR0				

# Timer\_n

## Organização Interna



\*A organização do Timer\_1 é idêntica.

# Configurações

**Modo:**

Modo de operação	BIT	
	<i>M01</i>	<i>M00</i>
Modo 0	0	0
Modo 1	0	1
Modo 2	1	0
Modo 3	1	1

*Tabela para Timer\_0.*

Modo de operação	BIT	
	<i>M11</i>	<i>M10</i>
Modo 0	0	0
Modo 1	0	1
Modo 2	1	0
Modo 3	1	1

*Tabela para Timer\_0.*

# Configurações

## *Fonte de clock:*

Para seleção da origem do clock devemos setar o bit CT:

$CT_0=0$  : Clock do sistema, dividido por 12, para o Timer0.

$CT_0=1$  : seleciona um clock externo pelo pino T0 (P3.4).

$CT_1=0$  : Clock do sistema, dividido por 12, para o Timer1.

$CT_1=1$  : seleciona um clock externo pelo pino T1 (P3.5)

# Configurações

## Acionamento por *INT*

Fazendo  $G_x=1$  dependerá do valor de  $INT_x$ :

$INT_0 = 0$       Habilita contagem sequencial para Timer0.

$INT_0 = 1$       Interrompe a contagem do Timer0.

$INT_1 = 0$       Habilita contagem sequencial para Timer1.

$INT_1 = 1$       Interrompe a contagem do Timer1.

Nota:  $INT_0$  pino P3.2 e  $INT_1$  no pino P3.3



# Exemplo 1

## **Modo 0:**

Configurar o **Timer1** para o Modo 0, logo precisamos fazer com que os bits de seleção de modo sejam:  $M_{11}=0$  e  $M_{10}=0$ .

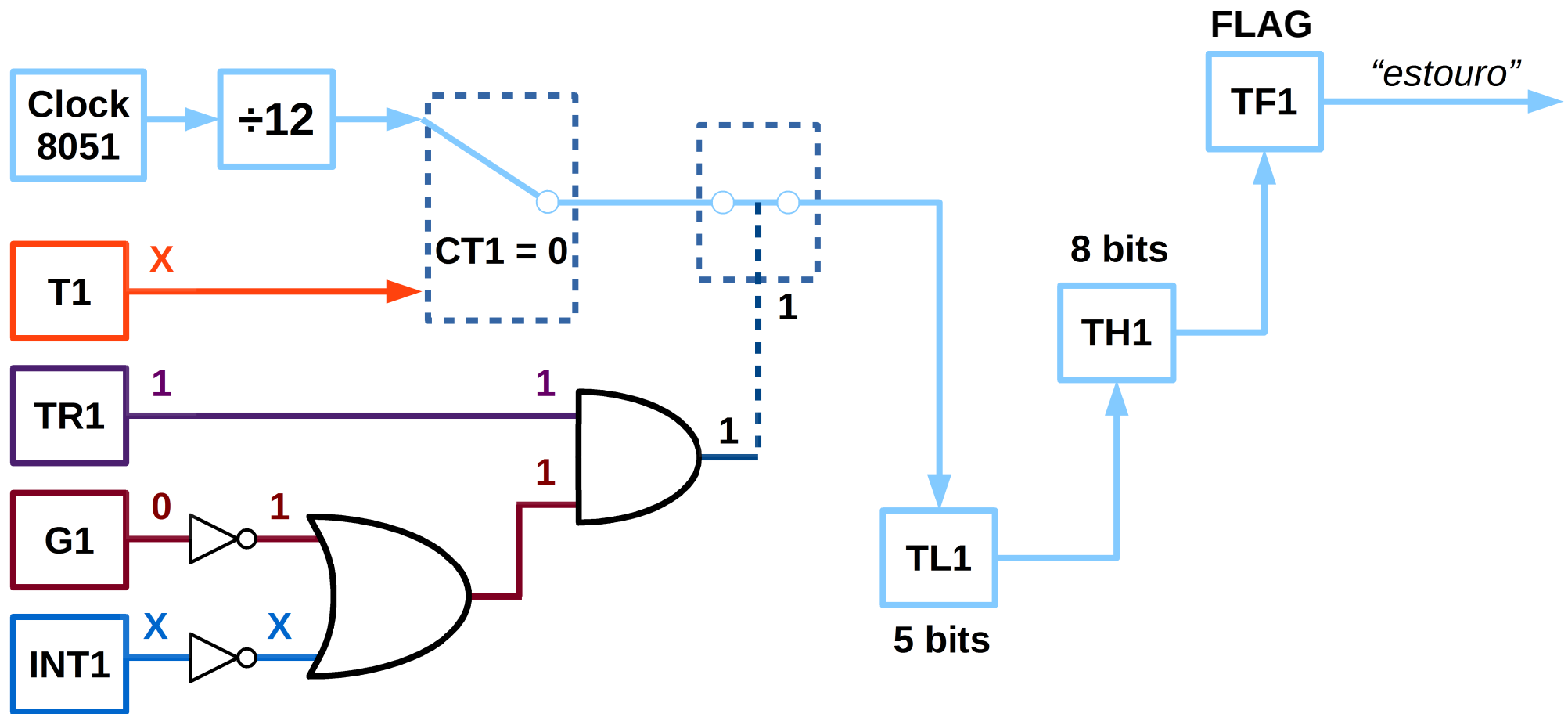
Se desejamos selecionar o clock do chip devemos fazer o bit  $CT_1=0$  e se desejamos desabilitar o pino de interrupção externa ( $INT_1$ ) devemos fazer  $G_1=0$ , logo  $INT_1=X$ .

Para habilitar o **Timer1** devemos fazer  $TR_1 = 1$ .

Dessa forma, resulta no circuito a seguir:

# Exemplo 1

## Organização Interna



Timer\_1 programado em Modo 0.

# *Exemplo 1*

ORG 0000h

SJMP *INICIO*

ORG 0030h

*INICIO:* MOV A, P1

*LOOP:* MOV B, P2

NOP

NOP

ADD A, B

NOP

NOP

END

## Exemplo 2

### Modo 1:

Configurar o **Timer0** no Modo 1, para isso devemos fazer com que os bits de seleção de modo sejam:  $M_{01}=0$  e  $M_{00}=1$ .

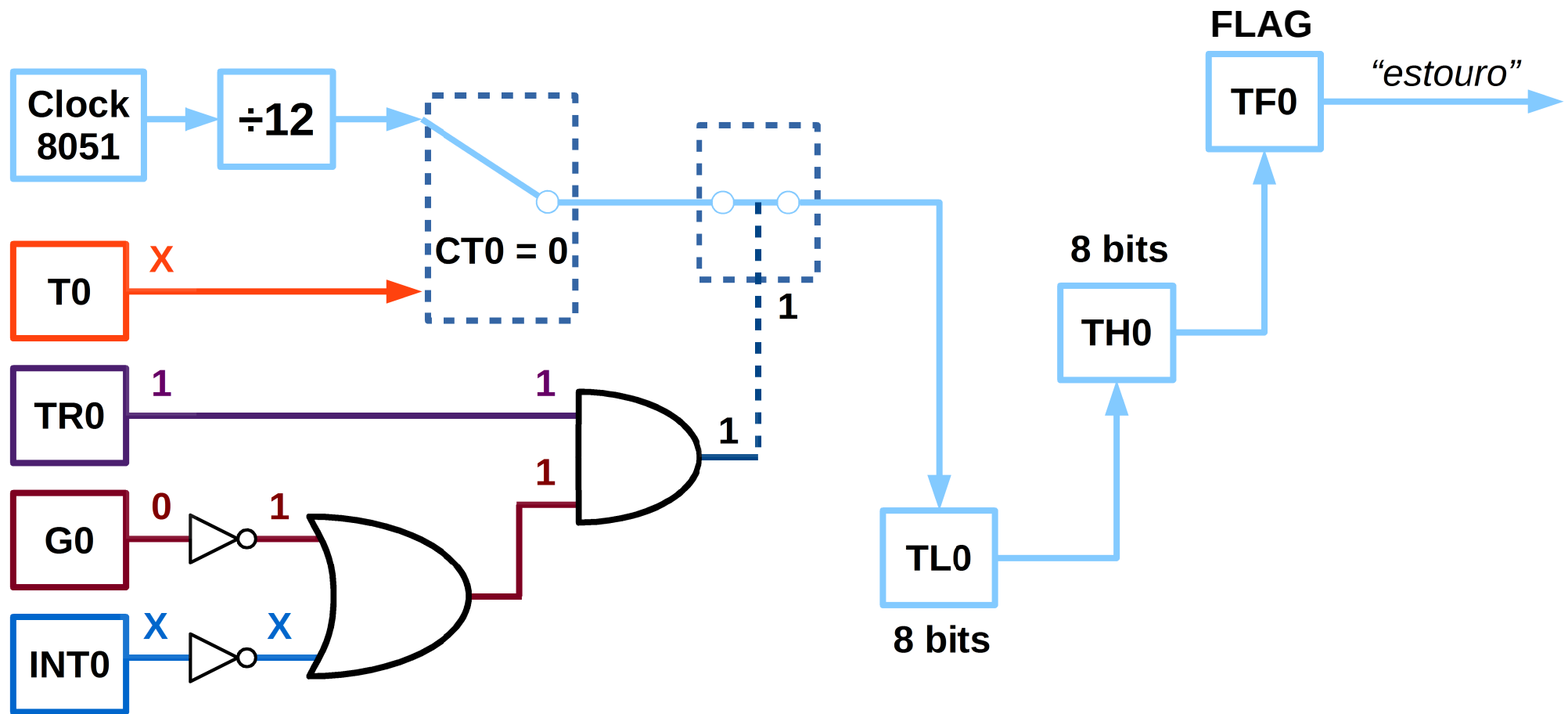
Se desejamos selecionar o clock do chip devemos fazer o bit  $CT_0=0$  e se desejamos desabilitar o pino de interrupção externa ( $INT_0$ ) devemos fazer  $G_0=0$ , logo  $INT_0=X$ .

Para habilitar o **Timer0** devemos fazer  $TR_0 = 1$ .

Dessa forma, resulta no circuito a seguir:

# Exemplo 2

## Organização Interna



*Timer\_0 programado em Modo 1.*

# *Exemplo 2*

ORG 0000h

SJMP *INICIO*

ORG 0030h

*INICIO:* MOV A, P1

*LOOP:* MOV B, P2

NOP

NOP

ADD A, B

NOP

NOP

END

# Exemplo 3

## Modo 2:

Configurar o **Timer0** no Modo 2, para isso devemos fazer com que os bits de seleção de modo sejam:  $M_{01}=1$  e  $M_{00}=0$ .

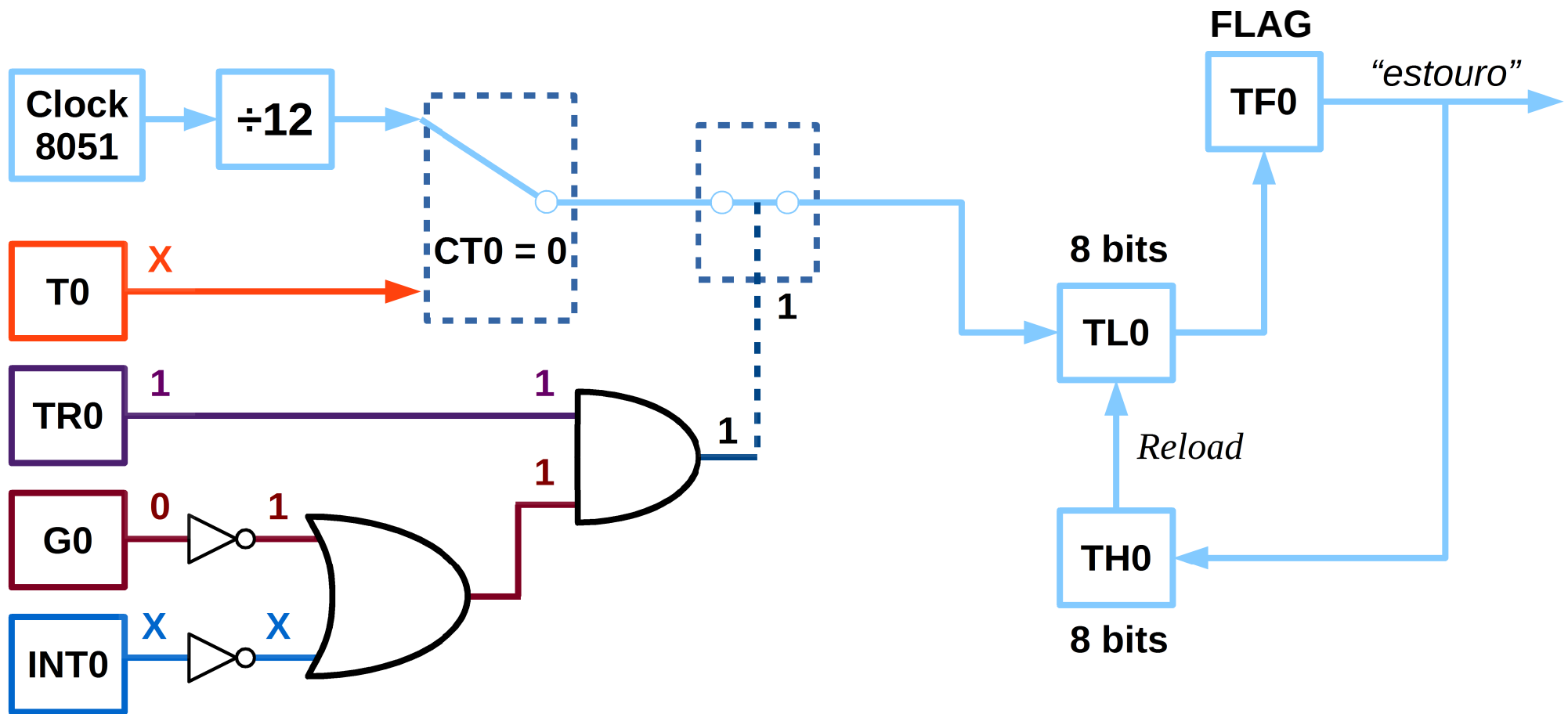
Se desejamos selecionar o clock do chip devemos fazer o bit  $CT_0=0$  e se desejamos desabilitar o pino de interrupção externa ( $INT_0$ ) devemos fazer  $G_0=0$ , logo  $INT_0=X$ .

Para habilitar o **Timer0** devemos fazer  $TR_0 = 1$ .

Dessa forma, resulta no circuito a seguir:

# Exemplo 3

## Organização Interna



Timer\_0 programado em Modo 2.



# *Exemplo 3*

ORG 0000h

SJMP *INICIO*

ORG 0030h

*INICIO:* MOV A, P1

*LOOP:* MOV B, P2

NOP

NOP

ADD A, B

NOP

NOP

END

# Exemplo 4

## Modo 3:

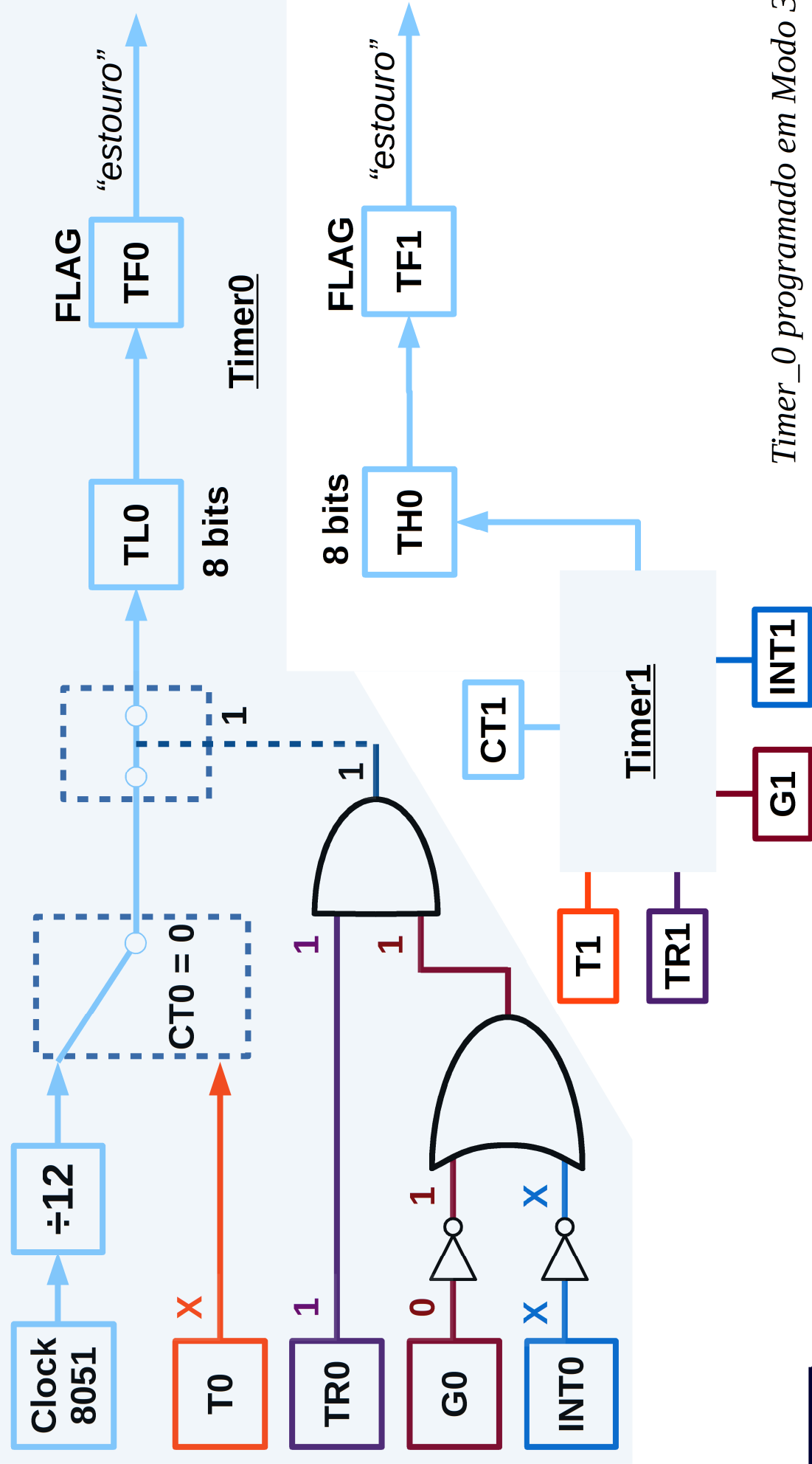
Configurar o **Timer0** no Modo 4, para isso devemos fazer com que os bits de seleção de modo sejam:  $M_{01}=1$  e  $M_{00}=1$ , e ainda,  $M_{11}=1$  e  $M_{10}=1$ , pois nesse momento será utilizado apenas o *Timer0*.

Se desejamos selecionar o clock do chip devemos fazer o bit  $CT_0=0$  e se desejamos desabilitar o pino de interrupção externa ( $INT_0$ ) devemos fazer  $G_0=0$ , logo  $INT_0=X$ .

Para habilitar o **Timer0** devemos fazer  $TR_0 = 1$ .

Dessa forma, resulta no circuito a seguir:

# Exemplo 4



Timer\_0 programado em Modo 3.

# Exemplo 4

ORG 0000h

SJMP *INICIO*

ORG 0030h

*INICIO:* MOV A, P1

*LOOP:* MOV B, P2

NOP

NOP

ADD A, B

NOP

NOP

END

# Referências

- [1] UFSC. A família de microcontroladores 8051. Notas de aula. disponível em:  
<<http://user.das.ufsc.br/~werner/eel7030/8051/Apostila8051Hari.pdf>> acesso: fev/16
- [2] NICOLOSI, D. E. C. Microcontrolador 8051 detalhado. 8 Ed. São Paulo: Érica, 2007.
- [3] FERLIN, E. P. Apostila 8051 - Manual. Curitiba: 2004. Disponível em:  
<[files.professorferlin.webnode.com.br/200000338.../apostila%208051.pdf](http://files.professorferlin.webnode.com.br/200000338.../apostila%208051.pdf)> Acesso Fev/2016.
- [4] RUIZ, W. Microcontroladores – Família MCS-51: Conceitos, Aplicações e Projetos. v. 6.0, 2004. Disponível em: <<http://aragao.spo.ifsp.edu.br/files/apostilas/microcontroladores.pdf>>  
Acesso Fev. 2016
- [5] FERLIN, E. P. Apostila 8051 - Manual. Curitiba: 2004. Disponível em:  
<[files.professorferlin.webnode.com.br/200000338.../apostila%208051.pdf](http://files.professorferlin.webnode.com.br/200000338.../apostila%208051.pdf)> Acesso Fev/2016.