- **Escrever um programa para alternar o estado dos LEDs vermelho e verde da placa MSP-EXP430G2 a cada 0,8s.**
- **Configurar a frequência de *Clock* para 1 MHz.**
- **Inicialmente os LEDs deverão estar apagados;**
- **Sempre que a interrupção do pino *P1.3* (*Botão S2*) for ativada, o estado dos LEDs deverá alternar entre apagado e piscante;**
- **A temporização deverá ser feita através do *Timer0_A***

Vetores de Interrupções do MSP430G2553

INTERRUPT SOURCE	INTERRUPT FLAG	SYSTEM INTERRUPT	WORD ADDRESS	PRIORITY
Power-Up External Reset Watchdog Timer+ Flash key violation PC out-of-range(1)	PORIFG RSTIFG WDTIFG KEYV(2)	Reset	0FFFEh	31, highest
NMI Oscillator fault Flash memory access violation	NMIIFG OFIFG ACCVIFG(2)(3)	(non)-maskable (non)-maskable (non)-maskable	0FFFCh	30
Timer1_A3	TA1CCR0 CCIFG(4)	maskable	0FFFAh	29
Timer1_A3	TA1CCR2 TA1CCR1 CCIFG, TAIFG(2)(4)	maskable	0FFF8h	28
Comparator_A+	CAIFG(4)	maskable	0FFF6h	27
Watchdog Timer+	WDTIFG	maskable	0FFF4h	26
Timer0_A3	TA0CCR0 CCIFG(4)	maskable	0FFF2h	25
Timer0_A3	TA0CCR2 TA0CCR1 CCIFG, TAIFG (5)(4)	maskable	0FFF0h	24
USCI_A0/USCI_B0 receive USCI_B0 I2C status	UCA0RXIFG, UCB0RXIFG(2)(5)	maskable	0FFEEh	23
USCI_A0/USCI_B0 transmit USCI_B0 I2C receive/transmit	UCA0TXIFG, UCB0TXIFG(2)(6)	maskable	0FFECh	22
ADC10 (MSP430G2x53 only)	ADC10IFG(4)	maskable	0FFEAh	21
			0FFE8h	20
I/O Port P2 (up to eight flags)	P2IFG.0 to P2IFG.7(2)(4)	maskable	0FFE6h	19
I/O Port P1 (up to eight flags)	P1IFG.0 to P1IFG.7(2)(4)	maskable	0FFE4h	18

P1 and P2 Interrupts

Each pin in ports P1 and P2 have interrupt capability, configured with the PxIFG, PxIE, and PxIES registers. All P1 pins source a single interrupt vector, and all P2 pins source a different single interrupt vector. The PxIFG register can be tested to determine the source of a P1 or P2 interrupt.

Port	Register	Short Form	Address	Register Type	Initial State
P1	Input	P1IN	020h	Read only	–
	Output	P1OUT	021h	Read/write	Unchanged
	Direction	P1DIR	022h	Read/write	Reset with PUC
	Interrupt Flag	P1IFG	023h	Read/write	Reset with PUC
	Interrupt Edge Select	P1IES	024h	Read/write	Unchanged
	Interrupt Enable	P1IE	025h	Read/write	Reset with PUC
	Port Select	P1SEL	026h	Read/write	Reset with PUC
	Port Select 2	P1SEL2	041h	Read/write	Reset with PUC
	Resistor Enable	P1REN	027h	Read/write	Reset with PUC

Table 12–3. Timer_A Registers

Register	Short Form	Register Type	Address	Initial State
Timer_A control	TACTL	Read/write	0160h	Reset with POR
Timer_A counter	TAR	Read/write	0170h	Reset with POR
Timer_A capture/compare control 0	TACCTL0	Read/write	0162h	Reset with POR
Timer_A capture/compare 0	TACCR0	Read/write	0172h	Reset with POR
Timer_A capture/compare control 1	TACCTL1	Read/write	0164h	Reset with POR
Timer_A capture/compare 1	TACCR1	Read/write	0174h	Reset with POR
Timer_A capture/compare control 2	TACCTL2†	Read/write	0166h	Reset with POR
Timer_A capture/compare 2	TACCR2†	Read/write	0176h	Reset with POR
Timer_A interrupt vector	TAIV	Read only	012Eh	Reset with POR

† Not present on MSP430x20xx Devices

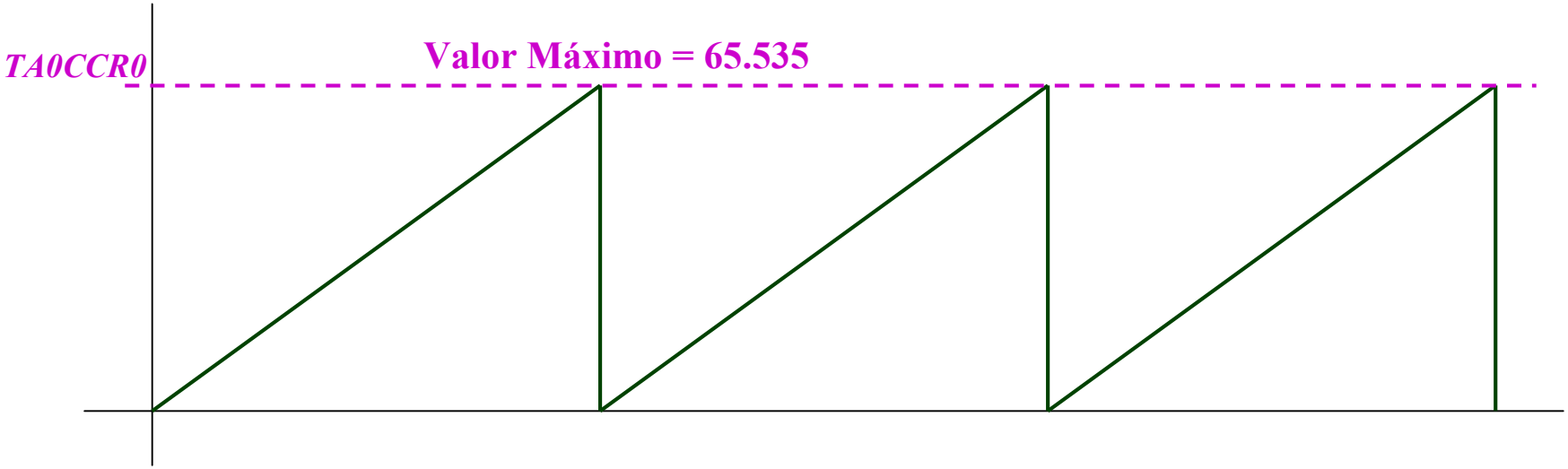
Timer0_A*: *TA0xxxxx

Timer1_A*: *TA1xxxxx

Como configurar o Timer para temporizar 0,8s (800ms)

DCO = 1 MHZ ➔ 800.000 ciclos

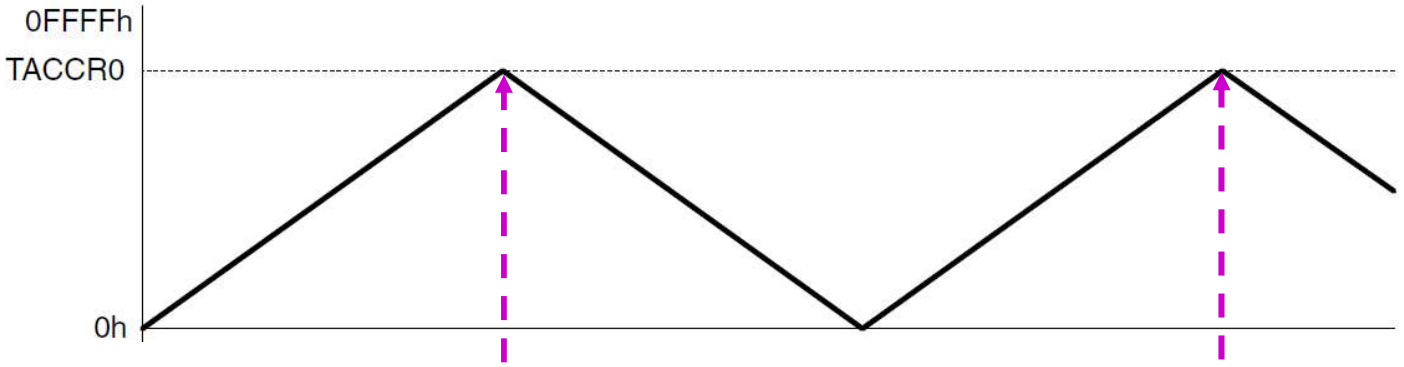
Timer* no modo *UP



Divisor = 8

8 × 65.535 = 528.280

Modo de Contagem *UP/DOWN*

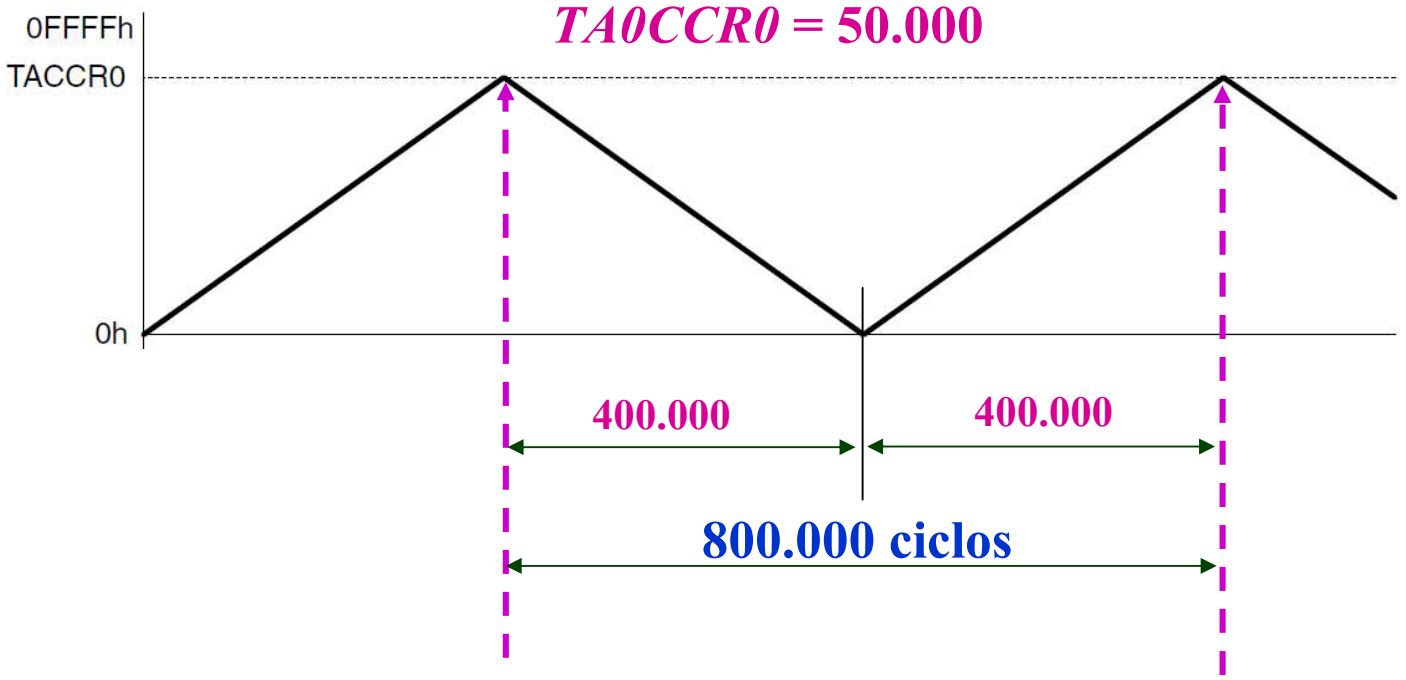


O *Timer* conta repetidamente de modo ascendente até o valor de comparação armazenado no Registrador *TACCR0* zero.

Quando atinge o valor de comparação (*TACCR0*) o *Timer* passa a contar de modo descendente até zero.

A *flag* de interrupção *CCIFG* é setada quando o *Timer* atinge o valor armazenado no Registrador de comparação *TACCR0*.

Modos de Contagem *UP/DOWN*



Divisor = 8

8 × 50.000 = 400.000

12.3.1 TACTL, Timer_A Control Register

15	14	13	12	11	10	9	8
Unused						TASSELx	
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
IDx		MCx		Unused	TACLR	TAIE	TAIFG
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

Unused Bits 15-10 Unused

TASSELx Bits 9-8 Timer_A clock source select

- 00 TACLK
- 01 ACLK
- 10 SMCLK
- 11 INCLK (INCLK is device-specific and is often assigned to the inverted TBCLK) (see the device-specific data sheet)

IDx Bits 7-6 Input divider. These bits select the divider for the input clock.

- 00 /1
- 01 /2
- 10 /4
- 11 /8

MCx Bits 5-4 Mode control. Setting MCx = 00h when Timer_A is not in use conserves power.

- 00 Stop mode: the timer is halted.
- 01 Up mode: the timer counts up to TACCR0.
- 10 Continuous mode: the timer counts up to 0FFFFh.
- 11 Up/down mode: the timer counts up to TACCR0 then down to 0000h.

Unused Bit 3 Unused

TACLR Bit 2 Timer_A clear. Setting this bit resets TAR, the clock divider, and the count direction. The TACLR bit is automatically reset and is always read as zero.

TAIE Bit 1 Timer_A interrupt enable. This bit enables the TAIFG interrupt request.

- 0 Interrupt disabled
- 1 Interrupt enabled

TAIFG Bit 0 Timer_A interrupt flag

- 0 No interrupt pending
- 1 Interrupt pending

12.3.4 TACCTLx, Capture/Compare Control Register

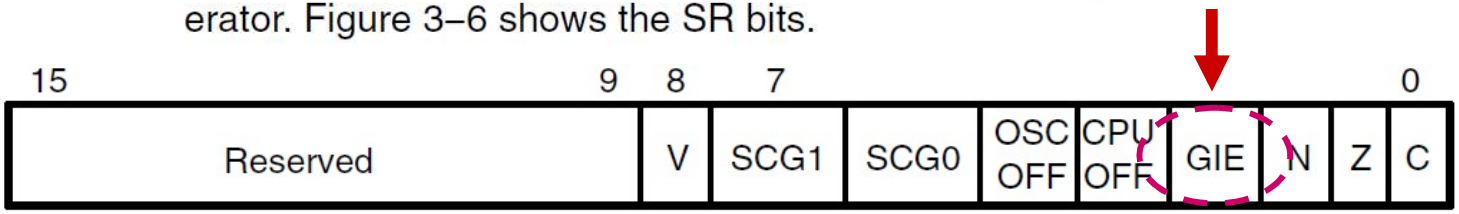
TA0CCTL0

15	14	13	12	11	10	9	8
CMx		**CCISx**		**SCS**	**SCCI**	**Unused**	**CAP**
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	r	r0	rw-(0)
7	6	5	4	3	2	1	0
OUTMODx			**CCIE**	**CCI**	**OUT**	**COV**	**CCIFG**

OUTMODx	Bits 7-5	Output mode. Modes 2, 3, 6, and 7 are not useful for TACCR0, because EQUx = EQU0.
	000	OUT bit value
	001	Set
	010	Toggle/reset
	011	Set/reset
	100	Toggle
	101	Reset
	110	Toggle/set
	111	Reset/set
CCIE	Bit 4	Capture/compare interrupt enable. This bit enables the interrupt request of the corresponding CCIFG flag.
	0	Interrupt disabled
	1	Interrupt enabled
CCI	Bit 3	Capture/compare input. The selected input signal can be read by this bit.
OUT	Bit 2	Output. For output mode 0, this bit directly controls the state of the output.
	0	Output low
	1	Output high
COV	Bit 1	Capture overflow. This bit indicates a capture overflow occurred. COV must be reset with software.
	0	No capture overflow occurred
	1	Capture overflow occurred
CCIFG	Bit 0	Capture/compare interrupt flag
	0	No interrupt pending
	1	Interrupt pending

Status Register (SR)

The status register (SR/R2), used as a source or destination register, can be used in the register mode only addressed with word instructions. The remaining combinations of addressing modes are used to support the constant generator. Figure 3–6 shows the SR bits.



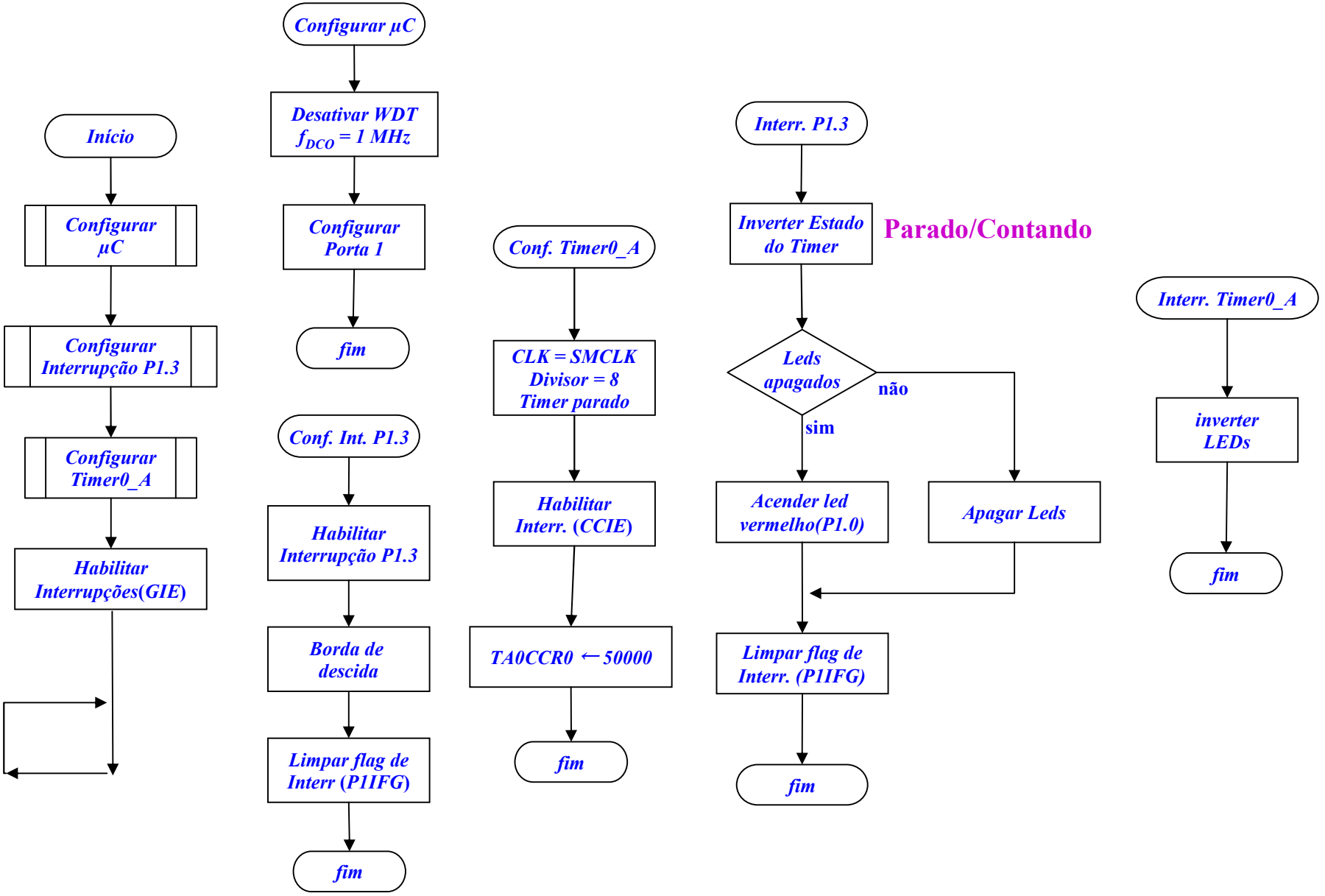
GIE General interrupt enable. This bit, when set, enables maskable interrupts. When reset, all maskable interrupts are disabled.

N Negative bit. This bit is set when the result of a byte or word operation is negative and cleared when the result is not negative.

- Word operation: N is set to the value of bit 15 of the result
- Byte operation: N is set to the value of bit 7 of the result

Z Zero bit. This bit is set when the result of a byte or word operation is 0 and cleared when the result is not 0.

C Carry bit. This bit is set when the result of a byte or word operation produced a carry and cleared when no carry occurred.


Início
Configurar µC
Configurar Interrupção P1.3
Configurar Timer0_A
Habilitar Interrupções(GIE)
Configurar µC
Desativar WDT
f_{DCO} = 1 MHz
Configurar Porta 1
fim
Conf. Int. P1.3
Habilitar Interrupção P1.3
Borda de descida
Limpar flag de Interr (P1IFG)
fim
Conf. Timer0_A
CLK = SMCLK
Divisor = 8
Timer parado
Habilitar Interr. (CCIE)
TA0CCR0 ← 50000
fim
Interr. P1.3
Inverter Estado do Timer
Parado/Contando
Leds apagados
não
sim
Acender led vermelho(P1.0)
Apagar Leds
Limpar flag de Interr. (P1IFG)
fim
Interr. Timer0_A
inverter LEDs
fim

Habilitação das interrupções:

```
_BIS_SR(GIE)
__bis_SR_register(GIE)
__enable_interrupt()
```

} Qualquer uma delas

Rotina de Interrupção da porta 1:

```
#pragma     vector = PORT1_VECTOR
__interrupt  void  interr_P1 (void)
   {
   . . .
   }
```

→ *0xFFE4*

Rotina de Interrupção do Timer0_A:

```
#pragma     vector = TIMER0_A0_VECTOR
__interrupt  void  interr_timer_A (void)
   {
   . . .
   }
```

→ *0xFFF2*