

Contenido

GPIO	2
RELOJ.....	3
WATCHDOG.....	3
INTERRUPCIONES	4
TIMERS	5
ePWM.....	5
UNIDAD CAD.....	8
UNIDAD SCI	9

GPIO

- GPXCTRL: A y B

Controla la sincronización del reloj con la cualificación

QUALPRD3	QUALPRD2	QUALPRD1	QUALPRD0
31	24	16	8
			0

Se controlan por bytes:

0x00: SYNC a SYSCLKOUT
0x01:QUALPRD = SYSCLKOUT/2
0x02:QUALPRD =SYSCLKOUT/4
...
0xff:QUALPRD = SYSCLKOUT/510

- GPXQSEL1 y GPXQSEL2: A y B

Número de muestras para cualificación. 2 registros de 32 bits por puerto, 2 bits por pin

00 sync a sysclkout
01 3 muestras
10 6 muestras
11 no cualificación

- GPXMUX1 y GPXMUX2: A, B y C

Control del multiplexor de selección de función del pin. 2 registros de 32 bits por puerto, 2 bits por pin.

00 -> Función E/S digital

- GPXDIR: A, B y C

Selección de un pin como entrada o salida. 1 registro de 32 bits por puerto.

0 -> Entrada

1 -> Salida

- GPXDAT: A, B y C

Data register

- GPXSET

Data set

- GPXCLEAR

Data clear

- GPXTOGGLE

Data toggle

- GPXPUD

Configuración de las resistencias de pull-up/down 0->Habilitar 1->Deshabilitar

RELOJ

SysCtrlRegs.PLLCR.bit.DIV -> Selecciona el multiplicador de frecuencia para la entrada al core

DIV	CLKIN
0 0 0 0	OSCCLK / n * (PLL bypass)
0 0 0 1	OSCCLK x 1 / n
0 0 1 0	OSCCLK x 2 / n
0 0 1 1	OSCCLK x 3 / n
0 1 0 0	OSCCLK x 4 / n
0 1 0 1	OSCCLK x 5 / n
0 1 1 0	OSCCLK x 6 / n
0 1 1 1	OSCCLK x 7 / n
1 0 0 0	OSCCLK x 8 / n
1 0 0 1	OSCCLK x 9 / n
1 0 1 0	OSCCLK x 10 / n

SysCtrlRegs.PLLSTS.bit.DIVSEL -> n por la que se divide el multiplicador

DIVSEL	n
0x	/4 *
10	/2
11	/1

* default

SysCtrlRegs.HISPCP -> Salida de reloj para periféricos alta frecuencia (ADC)

SysCtrlRegs.LOSPCP -> Para baja frecuencia (I2C, SPI, SCI, ...)

H/LSPCLK (2-0)	Peripheral Clock Frequency
0 0 0	SYSCLKOUT / 1
0 0 1	SYSCLKOUT / 2 (default HISPCP)
0 1 0	SYSCLKOUT / 4 (default LOSPCP)
0 1 1	SYSCLKOUT / 6
1 0 0	SYSCLKOUT / 8
1 0 1	SYSCLKOUT / 10
1 1 0	SYSCLKOUT / 12
1 1 1	SYSCLKOUT / 14

WATCHDOG

SysCtrlRegs.WDCR

Reservado	WDFLAG	WDDIS	WDCHK	WDPS
15-8	7	6	5-3	2-0

WDFLAG: bandera de reseteo por WD -> Se borra escribiendo un 1

WDDIS: deshabilita el WD escribiendo un 1

WDCHK: cualquier palabra distinta a 101 causa un reseteo

WDPS: selección de prescaler

SysCtrlRegs.WDKEY -> Contiene el registro a cambiar para causar un reseteo normal del contador

Reservado	WDKEY
-----------	-------

Para causar el reseteo se debe escribir:

0x55 -> Habilita la siguiente escritura

0xAA -> Resetea el contador

Cualquier otro valor que se escriba no tiene efecto

SysCtrlRegs.SCSR -> Protección contra la deshabilitación del WD

Reservado	WDINTS	WDENINT	WDOVERRIDE
15-3	2	1	0

WDOVERRIDE = 0 -> No permite deshabilitar mediante 1 en WDDIS

= 1 -> Default en el reset, permite deshabilitar

WDENINT = 1 -> Genera una interrupción por desbordamiento

= 0 -> Genera un reset del micro por desbordamiento

WDINTS -> Status de la interrupción, 0 si está activo

INTERRUPCIONES

PIR -> Multiplexor de interrupciones

IER -> Habilitación de interrupciones

INTM -> Habilitación global de interrupciones, sólo se puede habilitar mediante asm (0)

Asm(" CLRC INTM"); habilita

Asm(" SETC IINTM"); deshabilita

IFR -> Registro de flags

TIMERS

TIMERxTCR -> 16 bits

TIF	TIE	Reserved		FREE	SOFT	Reserved			TRB	TSS	Reserved				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- TIF -> Flag (Escribir 1 borra el bit)
- TIE -> Int enable (Escribir 1 habilita)
- Free/soft -> Interacción con emulador
- TRB -> Bit de recarga del timer (1=recarga)
- TSS -> Arrancar el timer (0 = start)

ConfigCpuTimer(&CpuTimer0, f, t)

- F -> frecuencia reloj
- T -> tiempo en us

ePWM

TBCTL (alto y bajo) -> Time base control

TBCTLH

Free-Soft	PHSDIR	CLKDIV	HSPCLKDIV
15 14	13	12 11 10	9 8 7

- Free_Soft -> Emulation halt behavior
 - 1x free run (no stop)
- PHSDIR -> Dirección de fase
 - 0 : count down after sync
 - 1 : count up after sync
- CLKDIV -> TB Clock prescale
 - 000 -> /1
 - 001 -> /2
 - ... 111 -> /128
- HSPCLKDIV -> High speed tb clock prescale
 - 000 -> /1
 - 001 -> /2
 - ... 111 -> /14

TBCTL

SWFSYNC	SYNCOSEL	PRDL	PHSEN	CTRM
6	5 4	3	2	1 0

- SWFSYNC -> Software force sync pulse
 - 0: no action
 - 1: force one-time sync
- SYNCOSEL -> Selecciona salida SYNCOUT
 - 00 : EPWMxSYNCl
 - 01: CTR = 0
 - 10: CTR = CMPB
 - 11: disable sync-out
- PRDL: carga periodo shadow
 - 0: Load en ctr = 0
 - 1: Carga inmediata
- PHSEN -> Habilitación fase
 - 0: deshabilitar
 - 1: CTR = TBPHS en señal EPWMxSYNCl
- CTRM -> modo del contador
 - 00: up
 - 01: down
 - 10: up y down
 - 11: stop

$$TBPRD = \frac{1}{2} \cdot \frac{f_{tbclk}}{f_{pwm}}$$

$$CMPA = (100 - Duty) * TBPRD$$

EPwmxRegs.CMPCTL -> Control del comparador

DBCTL -> Control de dead band

Reserved	IN_Mode	POLSEL	OUT_Mode
15- 6	5 4	3 2	1 0

- IN_Mode -> Modo in control
 - 00 PWMxA is source for RED and FED
 - 01 PWMxA is source for FED, PWMxB for RED
 - 10 PWMxB is source for FED, PWMxA for RED
 - 11 PWMxB is source for RED and FED
- POLSEL -> Seleccionar polaridad
 - 00 activo high
 - 01 activo low complementario (RED)
 - 10 activo high complementario (FED)
 - 11 activo low
-

- OUT_Mode -> Modo de salida
 - 00 disabled
 - 01 PWMxA = no delay, PWMxB = FED
 - 10 PWMxA = RED, PWMxB = no delay
 - 11 RED and FED

AQCTLx -> Qué hacer en el punto donde se cruzan los contadores con las referencias

--	CBD	CBU	CAD	CAU	PRD	ZRO
15 – 12	11 10	9 8	7 6	5 4	3 2	1 0

- CBD -> CTR = CMPB contando hacia abajo
- CBU -> CTR = CMPB contando hacia arriba
- CAD -> CTR = CMPA hacia abajo
- CAU -> CTR = CMPA hacia arriba
- PRD -> CTR = PRD
- ZRO -> CTR = 0

¿Qué hacer?

- 00: nada
- 01: poner pin a nivel bajo
- 10: poner pin a nivel alto
- 11: toggle

EPwmxBRegs.TZCTL

---	TZB	TZA
15 – 4	3 2	1 0

TZ1 a TZ6 acción que se va a hacer en EPWMxB/EPWMxA

- 00: high impedance
- 01: forzar a high
- 10: forzar a low
- 11: deshabilitar

EPwmxBRegs.TZEINT -> Modo de interrupción

--	OST	CBC	--
15 – 3	2	1	0

- OST Habilitar interrupción de un solo disparo
 - 1 = enable
- CBC Habilitar interrupciones en cada ciclo
 - 1 = enable

EPwmxBRegs.TZSEL -> Habilitación de pines desde los que entra la interrupción TZ para one-shot (OSHTx) o cycle-by-cycle (CBCx) -> (1 = enable)

UNIDAD CAD

AdcRegs.ADCTRL1

CONT_RUN	SEQ_OVRD	SEQ_CASC	--
6	5	4	3-0

- Cont_Run -> run continuo
 - 0: stop después de alcanzar el fin de la secuencia
 - 1: empieza otra vez desde el estado inicial
- Seq_OVRD -> funciona solo si cont_run = 1
 - 0: sequencer pointer resets to initial state at end of MAX_CONVn
 - 1: sequencer pointer resets to initial state after end state
- Seq_CASC: modo del secuenciador
 - 0: dual
 - 1: cascada

AdcRegs.ADCTRL2

ePWM_SOCB_SEQ	RST_SEQ1	SOC_SEQ1	-	INT_ENA_SEQ1	INT_MOD_SEQ1	-	ePWM_SOCA_SEQ1
15	14	13	12	11	10	9	8

- ePWM SOC B -> solo modo cascada
 - 0: no acción
 - 1: start con la señal ePWM
- RST_SEQ1 -> reset SEQ1
 - 0: nada
 - 1: reseteo inmediato de SEQ1 al estado inicial
- SOC_SEQ1 -> Start conversión
 - 0: clear pending trigger
 - 1: software trigger-start SEQ1
- INT_ENA_SEQ1 -> interrupt enable
 - 0: disable
 - 1: enable
- INT_MOD_SEQ1 -> Interrupt mode
 - 0: interrupt cada EOS
 - 1: interrupt cada otro EOS
- ePWM_SOCA_SEQ1
 - 0: no se empieza por trigger ePWM
 - 1: se empieza por trigger ePWM

AdcRegs.ADCTRL3

--	ADCBGRFND	ADCPWDN	ADCCLKPS	SMODE_SEL
15 – 8	7 6	5	4 -1	0

- ADCBGRFND -> Bandgap and reference power down
 - 00: powered down
 - 11: powered up
- ADCPWDN -> ADC power down
 - 0: powered down
 - 1: powered up
- ADCCLKPS -> clock prescale
 - 0: FCLK = HSPCLK
 - 1 a 15: FCLK = HSPCLK/(2*ADCCLKPS)
- SMODE_SEL -> modo de sampling
 - 0: modo secuencial
 - 1: modo simultaneo

AdcRegs.ADCMAXCONV -> Número de conversiones a realizar en cada evento

AdcRegs.ADCCHSELSEQ1 -> Selección de secuencia de canales

AdcRegs.ADCRESULTx -> Resultados de la conversión

AdcRegs.ADCREFEL.REF_SEL -> Selección de la referencia para el adc

- 00: interna
- 01: externa 2.048 V
- 10: externa 1.5 V
- 11: externa 1.024 V

UNIDAD SCI

ScixRegs.SCRICCR -> Control de comunicaciones

Stop Bits	Even/Odd Parity	Parity enable	Loopback enable	Addr/Idle mode	SCI char2	SCI char1	SCI char0
7	6	5	4	3	2	1	0

0 = 1 stop ; 0: odd 0: disabled ; 0: disabled; 0: idle-line Nº bits de datos = (binario +1)

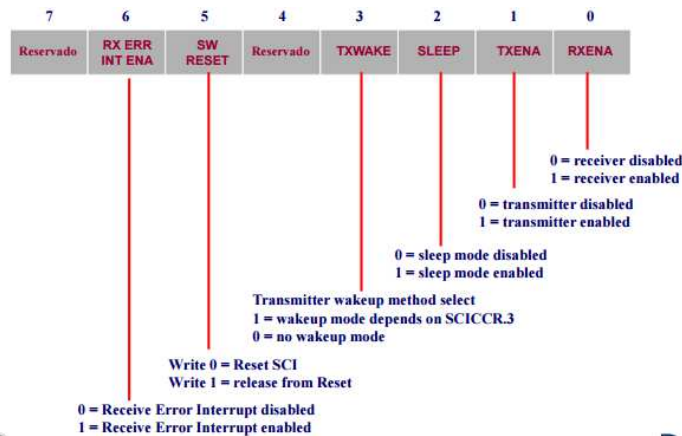
1 = 2 stop; 1: Even 1: enabled; 1: enabled ;1: addr-bit

$$SCI \text{ baud rate} = \frac{LSPCLK}{(BRR + 1) \cdot 8} \quad \text{si } BRR \text{ va de } 1 \text{ a } 65535$$

$$SCI \text{ baud rate} = \frac{LSPCLK}{16} \quad \text{si } BRR \text{ es } 0$$

→ Al registro baud-select SCIHBAUD y SCILBAUD (alto y bajo de 8 bits cada uno)

SCICTL1 -> reset, habilitar transmisión/recepción, wake-up, sleep, habilitar interrupciones por error



SCICTL2 -> flag de tx buffer, enable interrupción de rx break

--	TXRDY	TX EMPTY	--	RX/BK INT ENA	TX INT ENA
15-8	7	6	5-2	1	0

- TXRDY -> TX ready
 - 0: SCITXBUF is full
 - 1: SCITXBUF is empty
- TX EMPTY
 - 0: TXBUF or shift register are loaded with data
 - 1: Transmitter buffer and shift register are empty
- RX/BK INT ENA
 - 0: disable RXRDY/BRKDT interrupt
 - 1: enable
- TX INT ENA
 - 0: disable TXRDY interrupt
 - 1: enable

SCIRXST -> Status del RX

SCIFFTX -> Status del TX y FIFO (opciones de interrupción, reset, enable...)

SCIFFCT

ABD	ABD CLR	CDC	---	FFTXDLY
15	14	13	12-8	7-0

- ABD: auto baud detection 0 = not complete, 1 = complete
- ABD CLR = auto baud detection flag clear, 1 = clear
- CDC = CDC Calibrate 'A' 0 = disables ABD alignment, 1 = enables
- FFTXDLY = delay between every transfer from FIFO to transmit shift register inn number of SCI baud clock cycles (0 a 255)