Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	M8C
Core Size	8-Bit
Speed	12MHz
Connectivity	SPI
Peripherals	LVD, POR, WDT
Number of I/O	36
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	48-BSSOP (0.295", 7.50mm Width)
Supplier Device Package	48-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/infineon-technologies/cy7c60123-pvxc

3. Contents

4. Applications

The CY7C601xx and CY7C602xx are targeted for the following applications:

- PC wireless human interface devices (HID)
  - Mice (optomechanical, optical, trackball)
  - Keyboards
  - Presenter tools
- Gaming
  - Joysticks
  - Gamepad
- General-purpose wireless applications
  - Remote controls
  - Barcode scanners
  - POS terminal
  - Consumer electronics
  - Toys

5. Introduction

The enCoRe II LV family brings the features and benefits of the enCoRe II to non-USB applications. The enCoRe II family has an integrated oscillator that eliminates the external crystal or resonator, reducing overall cost. Other external components, such as wakeup circuitry, are also integrated into this chip.

The enCoRe II LV is a low-voltage, low-cost 8-bit flash-programmable microcontroller.

The enCoRe II LV features up to 36 GPIO pins. The I/O pins are grouped into five ports (Port 0 to 4). The pins on ports 0 and 1 are configured individually, when the pins on ports 2, 3, and 4 are only configured as a group. Each GPIO port supports high-impedance inputs, configurable pull-up, open-drain output, CMOS, and TTL inputs, and CMOS output with up to five pins that support programmable drive strength of up to 50-mA sink current. Additionally, each I/O pin is used to generate a GPIO interrupt to the microcontroller. Each GPIO port has its own GPIO interrupt vector with the exception of GPIO port 0. GPIO port 0 has, in addition to the port interrupt vector, three dedicated pins that have independent interrupt vectors (P0.2–P0.4).

The enCoRe II LV features an internal oscillator. Optionally, an external 1-MHz to 24-MHz crystal is used to provide a higher precision reference. The enCoRe II LV also supports external clock.

The enCoRe II LV has 8 KB of flash for user code and 256 bytes of RAM for stack space and user variables.

In addition, enCoRe II LV includes a WDT, a vectored interrupt controller, a 16-bit free-running timer with capture registers, and a 12-bit programmable interval timer. The power on reset (POR) circuit detects when power is applied to the device, resets the logic to a known state, and executes instructions at flash address 0x0000. When power falls below a programmable trip voltage, it generates a reset or is configured to generate an interrupt. There is a LVD circuit that detects when V_{CC} drops below a programmable trip voltage. This is configurable to generate a LVD interrupt to inform the processor about the low-voltage event. POR and LVD share the same interrupt; there is no separate interrupt for each. The WDT ensures the firmware never gets stalled in an infinite loop.

The microcontroller supports 17 maskable interrupts in the vectored interrupt controller. All interrupts can be masked. Interrupt sources include LVR or POR, a programmable interval timer, a nominal 1.024 ms programmable output from the free-running timer, two capture timers, five GPIO ports, three GPIO pins, two SPI, a 16-bit free-running timer wrap, and an internal wakeup timer interrupt. The wakeup timer causes periodic interrupts when enabled. The capture timers interrupt whenever a new timer value is saved due to a selected GPIO edge event. A total of eight GPIO interrupts support both TTL or CMOS thresholds. For additional flexibility, on the edge-sensitive GPIO pins, the interrupt polarity is programmable to be either rising or falling.

The free-running timer generates an interrupt at 1024-µs rate. It also generates an interrupt when the free-running counter overflow occurs – every 16.384 ms. The duration of an event under firmware control is measured by reading the timer at the start and end of an event, then calculating the difference between the two values. The two 8-bit capture timer registers save a programmable 8-bit range of the free-running timer when a GPIO edge occurs on the two capture pins (P0.5 and P0.6). The two 8-bit capture registers are ganged into a single 16-bit capture register.

The enCoRe II LV supports in-system programming by using the P1.0 and P1.1 pins as the serial programming mode interface.

6. Conventions

In this document, bit positions in the registers are shaded to indicate which members of the enCoRe II LV family implement the bits.

Available in all enCoRe II LV family members

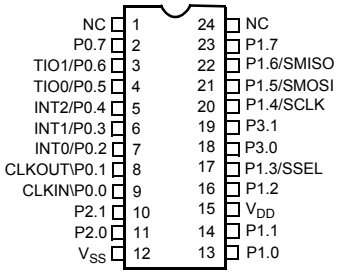
CY7C601xx only

7. Pinouts

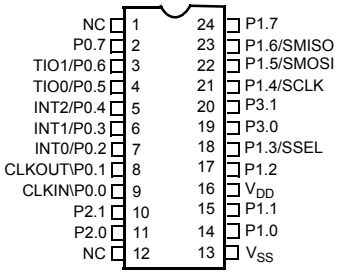
Figure 7-1. Package Configurations

Top View

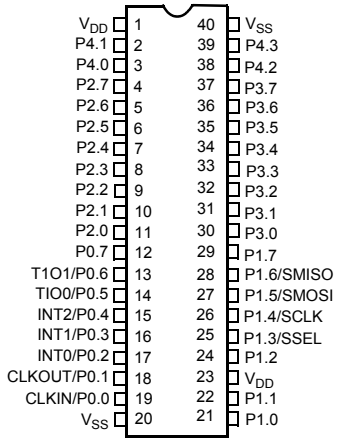
CY7C60223
24-Pin SOIC



CY7C60223
24-Pin QSOP



CY7C60123
40-Pin PDIP



CY7C60123
48-Pin SSOP

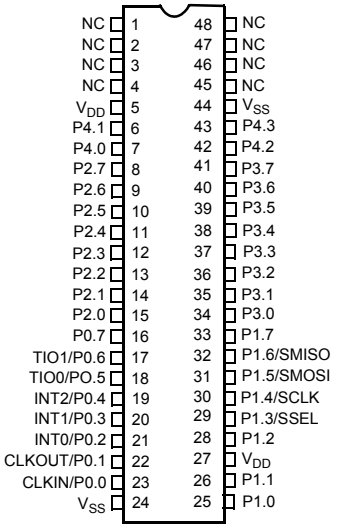


Table 8-1. enCoRe II LV Register Summary (continued)

The XIO bit in the CPU flags register must be set to access the extended register space for all registers above 0xFF.

Addr	Name	7	6	5	4	3	2	1	0	R/W	Default
34	IOSCTR	foffset[2:0]			Gain[4:0]					bbbbbbbb	000ddddd
35	XOSCTR	Reserved			XOSC XGM [2:0]			Reserved	Mode	---bbb-b	000ddddd
36	LPOSCTR	32 kHz low power	Reserved	32 kHz bias trim [1:0]		32 kHz freq trim [3:0]				b-bbbbbb	d-dddddd
3C	SPIDATA	SPIData[7:0]								bbbbbbbb	00000000
3D	SPICR	Swap	LSB first	Comm mode		CPOL	CPHA	SCLK select		bbbbbbbb	00000000
DA	INT_CLR0	GPIO port 1	Sleep timer	INT1	GPIO Port 0	SPI Receive	SPI transmit	INT0	POR/LVD	bbbbbbbb	00000000
DB	INT_CLR1	TCAP0	Prog interval timer	1 ms timer	Reserved					bbb-----	00000000
DC	INT_CLR2	Reserved	GPIO port 4	GPIO port 3	GPIO port 2	Reserved	INT2	16-bit counter wrap	TCAP1	-bbb-bbb	00000000
DE	INT_MSK3	ENSWINT	Reserved							r-------	00000000
DF	INT_MSK2	Reserved	GPIO port 4 int enable	GPIO port 3 int enable	GPIO port 2 int enable	Reserved	INT2 Int enable	16-bit counter wrap int enable	TCAP1 Int enable	-bbb-bbb	00000000
E1	INT_MSK1	TCAP0 int enable	Prog interval timer int enable	1 ms timer int enable	Reserved					bbb-----	00000000
E0	INT_MSK0	GPIO Port 1 int enable	Sleep timer int enable	INT1 int enable	GPIO port 0 int enable	SPI receive int enable	SPI transmit int enable	INT0 int enable	POR/LVD int enable	bbbbbbbb	00000000
E2	INT_VC	Pending interrupt [7:0]								bbbbbbbb	00000000
E3	RESWDT	Reset watchdog timer [7:0]								wwwwwww w	00000000
--	CPU_A	Temporary register T1 [7:0]								--------	00000000
--	CPU_X	X[7:0]								--------	00000000
--	CPU_PCL	Program counter [7:0]								--------	00000000
--	CPU_PCH	Program counter [15:8]								--------	00000000
--	CPU_SP	Stack pointer [7:0]								--------	00000000
F7	CPU_F	Reserved			XIO	Super	Carry	Zero	Global IE	---brbbb	00000010
FF	CPU_SCR	GIES	Reserved	WDRS	PORS	Sleep	Reserved	Reserved	Stop	r-ccb--b	00010100
1E0	OSC_CR0	Reserved		No buzz	Sleep timer [1:0]		CPU speed [2:0]			--bbbbbb	00001000
1E3	LVDCR	Reserved		PORLEV[1:0]		Reserved	VM[2:0]			--bb-bbb	00000000
1EB	ECO_TR	Sleep duty cycle [1:0]		Reserved						bb------	00000000
1E4	VLTCMP	Reserved						LVD	PPOR	------rr	00000000

Note In the R/W column:

b = Both read and write

r = Read only

w = Write only

c = Read or clear

d = Calibration value. Must not change during normal use

10.2.5 Destination Indexed

The result of an instruction using this addressing mode is placed within either the RAM memory space or the register space. Operand 1 is added to the X register forming the address that points to the location of the result. The source for the instruction is the A register. Arithmetic instructions require two sources; the second source is the location specified by Operand 1 added with the X register. Instructions using this addressing mode are two bytes in length.

Table 10-11. Destination Indexed

Opcode	Operand 1
Instruction	Destination index

Example

```
ADD   [X+7],      A     ;In this case, the value in the memory
                        location at address X+7 is added with the
                        accumulator and the result is placed in the
                        memory location at address X+7. The
                        accumulator is unchanged.
```

10.2.6 Destination Direct Source Immediate

The result of an instruction using this addressing mode is placed within either the RAM memory space or the register space. Operand 1 is the address of the result. The source for the instruction is Operand 2, which is an immediate value. Arithmetic instructions require two sources; the second source is the location specified by Operand 1. Instructions using this addressing mode are three bytes in length.

Table 10-12. Destination Direct Source Immediate

Opcode	Operand 1	Operand 2
Instruction	Destination address	Immediate value

Examples

```
ADD   [7],        5     ;In this case, value in the memory location
                        at address 7 is added to the immediate
                        value of 5, and the result is placed in the
                        memory location at address 7.
MOV   REG[8],     6     ;In this case, the immediate value of 6 is
                        moved into the register space location at
                        address 8.
```

10.2.7 Destination Indexed Source Immediate

The result of an instruction using this addressing mode is placed within either the RAM memory space or the register space. Operand 1 is added to the X register to form the address of the result. The source for the instruction is Operand 2, which is an immediate value. Arithmetic instructions require two sources; the second source is the location specified by Operand 1 added with the X register. Instructions using this addressing mode are three bytes in length.

Table 10-13. Destination Indexed Source Immediate

Opcode	Operand 1	Operand 2
Instruction	Destination index	Immediate value

Examples

```
ADD   [X+7],      5     ;In this case, the value in the memory
                        location at address X+7 is added
                        with the immediate value of 5, and
                        the result is placed in the memory
                        location at address X+7.
MOV   REG[X+8],   6     ;In this case, the immediate value of
                        6 is moved into the location in the
                        register space at address X+8.
```

10.2.8 Destination Direct Source Direct

The result of an instruction using this addressing mode is placed within the RAM memory. Operand 1 is the address of the result. Operand 2 is an address that points to a location in the RAM memory that is the source for the instruction. This addressing mode is only valid on the MOV instruction. The instruction using this addressing mode is three bytes in length.

Table 10-14. Destination Direct Source Direct

Opcode	Operand 1	Operand 2
Instruction	Destination address	Source address

Example

```
MOV   [7],        [8]   ;In this case, the value in the memory location
                        at address 8 is moved to the memory location
                        at address 7.
```

12.5.3 WriteBlock Function

The WriteBlock function is used to store data in flash. Data is moved 64 bytes at a time from SRAM to flash using this function. The WriteBlock function first checks the protection bits and determines if the desired BLOCKID is writable. If write protection is turned on, the WriteBlock function exits setting the accumulator and KEY2 back to 00h. KEY1 has a value of 01h, indicating a write failure. The configuration of the WriteBlock function is straightforward. The BLOCKID of the flash block, where the data is stored, is determined and stored at SRAM address FAh.

The SRAM address of the first of the 64 bytes to be stored in flash is indicated using the POINTER variable in the parameter block (SRAM address FBh). Finally, the CLOCK and DELAY value are set correctly. The CLOCK value determines the length of the write pulse used to store the data in flash. The CLOCK and DELAY values are dependent on the CPU speed and must be set correctly. Refer to the Clocking section for additional information.

Table 12-5. WriteBlock Parameters

Name	Address	Description
KEY1	0,F8h	3Ah
KEY2	0,F9h	Stack pointer value, when SSC is executing
BLOCK ID	0,FAh	8 KB flash block number (00h–7Fh) 4 KB flash block number (00h–3Fh) 3 KB flash block number (00h–2Fh)
POINTER	0,FBh	First 64 addresses in SRAM where the data is stored in flash is located before calling WriteBlock
CLOCK	0,FCh	Clock divider used to set the write pulse width
DELAY	0,FEh	For a CPU speed of 12 MHz set to 56h

12.5.4 EraseBlock Function

The EraseBlock function is used to erase a block of 64 contiguous bytes in flash. The EraseBlock function first checks the protection bits and determines if the desired BLOCKID is writable. If write protection is turned on, the EraseBlock function exits setting the accumulator and KEY2 back to 00h. KEY1 has a value of 01h, indicating a write failure. The EraseBlock function is only useful as the first step in programming. Erasing a block does not make data in a block fully unreadable. If the objective is to obliterate data in a block, the best method is to perform an EraseBlock followed by a WriteBlock of all zeros.

To set up the parameter block for EraseBlock, correct key values must be stored in KEY1 and KEY2. The block number to be erased is stored in the BLOCKID variable and the CLOCK and DELAY values are set based on the current CPU speed.

Table 12-6. EraseBlock Parameters

Name	Address	Description
KEY1	0,F8h	3Ah
KEY2	0,F9h	Stack pointer value, when SSC is executed
BLOCKID	0,FAh	Flash block number (00h–7Fh)
CLOCK	0,FCh	Clock divider used to set the erase pulse width
DELAY	0,FEh	For a CPU speed of 12 MHz set to 56h

12.5.5 ProtectBlock Function

The enCoRe II LV devices offer flash protection on a block-by-block basis. Table 12-7 lists the protection modes available. In the table, ER and EW indicate the ability to perform external reads and writes; IW is used for internal writes. Internal reading is always permitted using the ROMX instruction. The ability to read using the SROM ReadBlock function is indicated by SR. The protection level is stored in two bits according to Table 12-7. These bits are bit packed into 64 bytes of the protection block. Therefore, each protection block byte stores the protection level for four flash blocks. The bits are packed into a byte, with the lowest numbered block's protection level stored in the lowest numbered bits in Table 12-7.

The first address of the protection block contains the protection level for blocks 0 through 3; the second address is for blocks 4 through 7. The 64th byte stores the protection level for blocks 252 through 255.

Table 12-7. Protection Modes

Mode	Settings	Description	Marketing
00b	SR ER EW IW	Unprotected	Unprotected
01b	$\overline{SR}$ ER EW IW	Read protect	Factory upgrade
10b	$\overline{SR}$ ER $\overline{EW}$ IW	Disable external write	Field upgrade
11b	$\overline{SR}$ ER $\overline{EW}$ $\overline{IW}$	Disable internal write	Full protection

7	6	5	4	3	2	1	0
Block n+3		Block n+2		Block n+1		Block n	

Only an EraseAll decreases the protection level by placing zeros in all locations of the protection block. To set the level of protection, the ProtectBlock function is used. This function takes data from SRAM, starting at address 80h, and ORs it with the current values in the protection block. The result of the OR operation is then stored in the protection block. The EraseBlock function does not change the protection level for a block. Because the SRAM location for the protection data is fixed and there is only one protection block for every flash macro, the ProtectBlock function expects very few variables in the parameter block to be set before calling the function. The parameter block values that are, besides the keys, are the CLOCK and DELAY values.

Table 13-2. OSC Control 0 (OSC_CR0) [0x1E0] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved		No Buzz	Sleep Timer [1:0]		CPU Speed [2:0]		
Read/Write	–	–	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	1	0	0	0

Bit [7:6]: Reserved

Bit 5: No buzz

During sleep (the sleep bit is set in the CPU_SCR register—Table 14-1 on page 31), the LVD and POR detection circuit is turned on periodically to detect any POR and LVD events on the V_{CC} pin (the sleep duty cycle bits in the ECO_TR are used to control the duty cycle—Table 16-3 on page 36). To facilitate the detection of POR and LVD events, the 'No Buzz' bit is used to continuously enable the LVD and POR detection circuit during sleep. This results in a faster response to an LVD or POR event during sleep at the expense of a slightly higher than average sleep current. Obtaining the absolute lowest power usage in sleep mode requires the 'No Buzz' bit to be clear.

0 = The LVD and POR detection circuit is turned on periodically as configured in the sleep duty cycle.

1 = The sleep duty cycle value is overridden. The LVD and POR detection circuit is always enabled.

Note The periodic sleep duty cycle enabling is independent with the sleep interval shown in the Sleep [1:0] bits below.

Bit [4:3]: Sleep timer [1:0]

Sleep Timer [1:0]	Sleep Timer Clock Frequency (Nominal)	Sleep Period (Nominal)	Watchdog Period (Nominal)
00	512 Hz	1.95 ms	6 ms
01	64 Hz	15.6 ms	47 ms
10	8 Hz	125 ms	375 ms
11	1 Hz	1 sec	3 sec

Note Sleep intervals are approximate.

Bit [2:0]: CPU speed [2:0]

The enCoRe II LV operates over a range of CPU clock speeds. The reset value for the CPU Speed bits is zero; therefore, the default CPU speed is 3 MHz.

CPU Speed [2:0]	CPU when Internal Oscillator is selected	External Clock
000	3 MHz (Default)	Clock In/8
001	6 MHz	Clock In/4
010	12 MHz	Clock In/2
011	Reserved	Reserved
100	1.5 MHz	Clock In/16
101	750 kHz	Clock In/32
110	187 kHz	Clock In/128
111	Reserved	Reserved

Note This register exists in the second bank of I/O space. This requires setting the XIO bit in the CPU flags register.

Table 13-1. Timer Clock Configuration (TMRCLKCR) [0x31] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	TCAPCLK divider		TCAPCLK select		ITMRCLK divider		ITMRCLK select	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	1	0	0	0	1	1	1	1

Bit [7:6]: TCAPCLK divider [1:0]

TCAPCLK Divider controls the TCAPCLK divisor.

0 0 = Divider Value 2

0 1 = Divider Value 4

1 0 = Divider Value 6

1 1 = Divider Value 8

Bit [5:4]: TCAPCLK select

The TCAPCLK Select field controls the source of the TCAPCLK.

0 0 = Internal 24-MHz oscillator

0 1 = External Crystal Oscillator—external crystal oscillator on CLKIN and CLKOUT if the external crystal oscillator is enabled, CLKIN input if the external crystal oscillator is disabled (the XOSC Enable bit of the CLKIOCR Register is cleared—Table 13-3 on page 26.)

1 0 = Internal 32-kHz oscillator

1 1 = TCAPCLK disabled

Note The 1024 µs interval timer is based on the assumption that TCAPCLK is running at 4 MHz. Changes in TCAPCLK frequency cause a corresponding change in the 1024 µs interval timer frequency.

Bit [3:2]: ITMRCLK divider

ITMRCLK Divider controls the ITMRCLK divisor.

0 0 = Divider value of 1

0 1 = Divider value of 2

1 0 = Divider value of 3

1 1 = Divider value of 4

Bit [1:0]: ITMRCLK select

0 0 = Internal 24-MHz oscillator

0 1 = External crystal oscillator—external crystal oscillator on CLKIN and CLKOUT if the external crystal oscillator is enabled, CLKIN input if the external crystal oscillator is disabled.

1 0 = Internal 32-kHz oscillator

1 1 = TCAPCLK

Note Changing the source of TMRCLK requires both the source and destination clocks to be running. It is not possible to change the clock source away from TCAPCLK after that clock is stopped.

13.2.6 LPOSC Trim

Table 13-4. LPOSC Trim (LPOSCTR) [0x36] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	32 kHz Low Power	Reserved	32 kHz Bias Trim [1:0]		32 kHz Freq Trim [3:0]			
Read/Write	R/W	–	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	–	D	D	D	D	D	D

This register is used to calibrate the 32-kHz low-speed oscillator. The reset value is undefined but during boot the SROM writes a calibration value that is determined during manufacturing test. This is the meaning of 'D' in the Default field. The trim value is adjusted vs. voltage as noted in Table 13-1 on page 24.

Bit 7: 32 kHz low-power

0 = The 32-kHz low-speed oscillator operates in normal mode.

1 = The 32-kHz low-speed oscillator operates in a low-power mode. The oscillator continues to function normally but with reduced accuracy.

Bit 6: Reserved

Bit [5:4]: 32 kHz bias trim [1:0]

These bits control the bias current of the low power oscillator.

0 0 = Mid bias

0 1 = High bias

1 0 = Reserved

1 1 = Reserved

Note Do not program the 32-kHz bias trim [1:0] field with the reserved 10b value as the oscillator does not oscillate at all corner conditions with this setting.

Bit [3:0]: 32-kHz Freq Trim [3:0]

These bits are used to trim the frequency of the low-power oscillator.

13.3 CPU Clock During Sleep Mode

When the CPU enters sleep mode the CPUCLK select (Bit 0, Table 13-1 on page 24) is forced to the internal oscillator, and the oscillator is stopped. When the CPU comes out of sleep mode it runs on the internal oscillator. The internal oscillator recovery time is three clock cycles of the internal 32-kHz low-power oscillator.

If the system requires the CPU to run off the external clock after waking from sleep mode, firmware needs to switch the clock source for the CPU. If the external clock source is the external oscillator and the oscillator is disabled, firmware needs to enable the external oscillator, wait for it to stabilize, and then change the clock source.

16.1 POR Compare State

Table 16-2. Voltage Monitor Comparators Register (VLTCMP) [0x1E4] [R]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved						LVD	PPOR
Read/Write	–	–	–	–	–	–	R	R
Default	0	0	0	0	0	0	0	0

This read-only register allows reading the current state of the LVD and PPOR comparators.

Bit [7:2]: Reserved

Bit 1: LVD

This bit is set to indicate that the LVD comparator has tripped, indicating that the supply voltage has gone below the trip point set by VM[2:0] (See Table 16-1 on page 35).

0 = No low-voltage-detect event

1 = A low-voltage-detect has tripped

Bit 0: PPOR

This bit is set to indicate that the PPOR comparator has tripped, indicating that the supply voltage is below the trip point set by PORLEV[1:0].

0 = No PPOR event

1 = A PPOR event has occurred

Note This register exists in the second bank of I/O space. This requires setting the XIO bit in the CPU flags register.

16.2 ECO Trim Register

Table 16-3. ECO (ECO_TR) [0x1EB] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Sleep duty cycle [1:0]		Reserved					
Read/Write	R/W	R/W	–	–	–	–	–	–
Default	0	0	0	0	0	0	0	0

This register controls the ratios (in numbers of 32 kHz clock periods) of “on” time versus “off” time for LVD and POR detection circuit.

Bit [7:6]: Sleep duty cycle [1:0]

0 0 = 1/128 periods of the Internal 32 kHz low speed oscillator.

0 1 = 1/512 periods of the Internal 32 kHz low speed oscillator.

1 0 = 1/32 periods of the Internal 32 kHz low speed oscillator.

1 1 = 1/8 periods of the Internal 32 kHz low speed oscillator.

Note This register is only accessed in the second bank of I/O space. This requires setting the XIO bit in the CPU flags register.

17. General-Purpose I/O Ports

17.1 Port Data Registers

17.1.1 P0 Data

Table 17-1. P0 Data Register (P0DATA)[0x00] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	P0.7	P0.6/TIO1	P0.5/TIO0	P0.4/INT2	P0.3/INT1	P0.2/INT0	P0.1/CLKOUT	P0.0/CLKIN
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This register contains the data for Port 0. Writing to this register sets the bit values to be output on output enabled pins. Reading from this register returns the current state of the Port 0 pins.

Bit 7: P0.7 Data

Bit [6:5]: P0.6–P0.5 Data/TIO1 and TIO0

Beside their use as the P0.6–P0.5 GPIOs, these pins are also used for alternate functions as the Capture Timer input or timer output pins (TIO1 and TIO0). To configure the P0.5 and P0.6 pins, refer to the P0.5/TIO0–P0.6/TIO1 Configuration Register (Table 17-4 on page 41).

Bit [4:2]: P0.4–P0.2 Data/INT2–INT0

Beside their use as the P0.4–P0.2 GPIOs, these pins are also used for the alternate functions as the interrupt pins (INT0–INT2). To configure the P0.4–P0.2 pins, refer to the P0.2/INT0–P0.4/INT2 Configuration Register (Table 17-3 on page 40).

Bit 1: P0.1/CLKOUT

Beside its use as the P0.1 GPIO, this pin is also used for the alternate function as the CLK OUT pin. To configure the P0.1 pin, refer to the P0.1/CLKOUT Configuration Register (Table 17-2 on page 40).

Bit 0: P0.0/CLKIN

Beside its use as the P0.0 GPIO, this pin is also used for the alternate function as the CLKIN pin. To configure the P0.0 pin, refer to the P0.0/CLKIN Configuration Register (Table 17-1 on page 39).

17.1.2 P1 Data

Table 17-2. P1 Data Register (P1DATA) [0x01] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	P1.7	P1.6/SMISO	P1.5/SMOSI	P1.4/SCLK	P1.3/SSEL	P1.2	P1.1	P1.0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This register contains the data for Port 1. Writing to this register sets the bit values to be output on output enabled pins. Reading from this register returns the current state of the Port 1 pins.

Bit 7: P1.7 data

Bit [6:3]: P1.6–P1.3 Data/SPI Pins (SMISO, SMOSI, SCLK, SSEL)

Beside their use as the P1.6–P1.3 GPIOs, these pins are also used for the alternate function as the SPI interface pins. To configure the P1.6–P1.3 pins, refer to the P1.3–P1.6 configuration register (Table 17-9 on page 42).

Bit [2:0]: P1.2–P1.0

17.2.10 P0.1/CLKOUT Configuration

Table 17-2. P0.1/CLKOUT Configuration (P01CR) [0x06] R/W]

Bit #	7	6	5	4	3	2	1	0
Field	CLK Output	Int Enable	Int Act Low	TTL Thresh	High Sink	Open Drain	Pull-up Enable	Output Enable
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This pin is shared between the P0.1 GPIO use and the CLKOUT pin for the external crystal oscillator. When the external oscillator is enabled the settings of this register are ignored. When CLK output is set, the internally selected clock is sent out onto P0.1CLKOUT pin.

The alternate function of the pin as the CLKOUT is only available in the CY7C601xx. When the external oscillator is enabled (the XOSC enable bit of the CLKIOCR register is set—Table 13-3 on page 26), the GPIO function of the pin is disabled.

The 50 mA sink drive capability is only available in the CY7C601xx. In the CY7C602xx, only 8 mA sink drive capability is available on this pin regardless of the setting of the high sink bit.

Bit 7: CLK output

0 = The clock output is disabled.

1 = The clock selected by the CLK Select field (Bit [1:0] of the CLKIOCR register—Table 13-3 on page 26) is driven out to the pin.

17.2.11 P0.2/INT0–P0.4/INT2 Configuration

Table 17-3. P0.2/INT0–P0.4/INT2 Configuration (P02CR–P04CR) [0x07–0x09] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved		Int Act Low	TTL Thresh	Reserved	Open Drain	Pull-up Enable	Output Enable
Read/Write	–	–	R/W	R/W	–	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

These registers control the operation of pins P0.2–P0.4 respectively. These pins are shared between the P0.2–P0.4 GPIOs and the INT0–INT2. The INT0–INT2 interrupts are different from all other GPIO interrupts. These pins are connected directly to the interrupt controller to provide three edge-sensitive interrupts with independent interrupt vectors. These interrupts occur on a rising edge when Int Act Low is clear and on a falling edge when Int Act Low is set. These pins are enabled as interrupt sources in the interrupt controller registers (Table 20-7 on page 59 and Table 20-5 on page 58).

To use these pins as interrupt inputs, configure them as inputs by clearing the corresponding Output Enable. If the INT0–INT2 pins are configured as outputs with interrupts enabled, firmware generates an interrupt by writing the appropriate value to the P0.2, P0.3, and P0.4 data bits in the P0 Data Register.

Regardless of whether the pins are used as interrupt or GPIO pins, the Int Enable, Int Act Low, TTL Threshold, Open Drain, and pull-up enable bits control the behavior of the pin.

The P0.2/INT0–P0.4/INT2 pins are individually configured with the P02CR (0x07), P03CR (0x08), and P04CR (0x09) respectively.

Note Changing the state of the Int Act Low bit generates an unintentional interrupt. When configuring these interrupt sources, follow this procedure:

1. Disable interrupt source
2. Configure interrupt source
3. Clear any pending interrupts from the source
4. Enable interrupt source

17.2.15 P1.1 Configuration

Table 17-7. P1.1 Configuration (P11CR) [0x0E] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved	Int Enable	Int Act Low	Reserved		Open Drain	Reserved	Output Enable
Read/Write	–	R/W	R/W	–	–	R/W	–	R/W
Default	0	0	0	0	0	0	0	0

This register controls the operation of the P1.1 pin.

The pull-up resistor on this pin is enabled by the P10CR Register.

Note There is no 2 mA sourcing capability on this pin. The pin can only sink 5 mA at V_{OL3} (see DC Characteristics on page 60)

If this pin is used as a general purpose output, it draws current. It is, therefore, configured as an input to reduce current draw.

17.2.16 P1.2 Configuration

Table 17-8. P1.2 Configuration (P12CR) [0x0F] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	CLK Output	Int Enable	Int Act Low	TTL Threshold	Reserved	Open Drain	Pull-up Enable	Output Enable
Read/Write	R/W	R/W	R/W	R/W	–	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This register controls the operation of the P1.2.

Bit 7: CLK Output

0 = The internally selected clock is not sent out onto P1.2 pin.

1 = This CLK Output is used to observe connected external crystal oscillator clock connected in CY7C601xx. When CLK Output is set, the internally selected clock is sent out onto P1.2 pin.

Note: Table 13-3 on page 26 is used to select the external or internal clock in enCoRe II devices

17.2.17 P1.3 Configuration (SSEL)

Table 17-9. P1.3 Configuration (P13CR) [0x10] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved	Int Enable	Int Act Low	Reserved	High Sink	Open Drain	Pull-up Enable	Output Enable
Read/Write	–	R/W	R/W	–	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This register controls the operation of the P1.3 pin. This register exists in all enCoRe II LVparts.

The P1.3 GPIO's threshold is always set to TTL.

When the SPI hardware is enabled or disabled, the pin is controlled by the Output Enable bit and the corresponding bit in the P1 data register.

Regardless of whether the pin is used as an SPI or GPIO pin the Int Enable, Int act Low, high sink, open drain, and pull-up enable control the behavior of the pin.

17.2.21 P3 Configuration

Table 17-13. P3 Configuration (P3CR) [0x16] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved	Int Enable	Int Act Low	TTL Thresh	High Sink	Open Drain	Pull-up Enable	Output Enable
Read/Write	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

In CY7C602xx, this register controls the operation of pins P3.0–P3.1. In CY7C601xx, this register controls the operation of pins P3.0–P3.7.

The 50-mA sink drive capability is only available on pin P3.7 and only on CY7C601xx. In CY7C602xx, only an 8-mA sink drive capability is available on this pin regardless of the setting of the high sink bit.

17.2.22 P4 Configuration

Table 17-14. P4 Configuration (P4CR) [0x17] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Reserved	Int Enable	Int Act Low	TTL Thresh	High Sink	Open Drain	Pull-up Enable	Output Enable
Read/Write	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

This register exists only in CY7C601xx. This register controls the operation of pins P4.0–P4.3.

18. Serial Peripheral Interface (SPI)

The SPI master and slave interface core logic runs on the SPI clock domain. The SPI clock is a divider off of the CPUCLK when in the master mode. SPI is a four pin serial interface comprised of a clock, an enable, and two data pins.

Figure 18-1. SPI Block Diagram

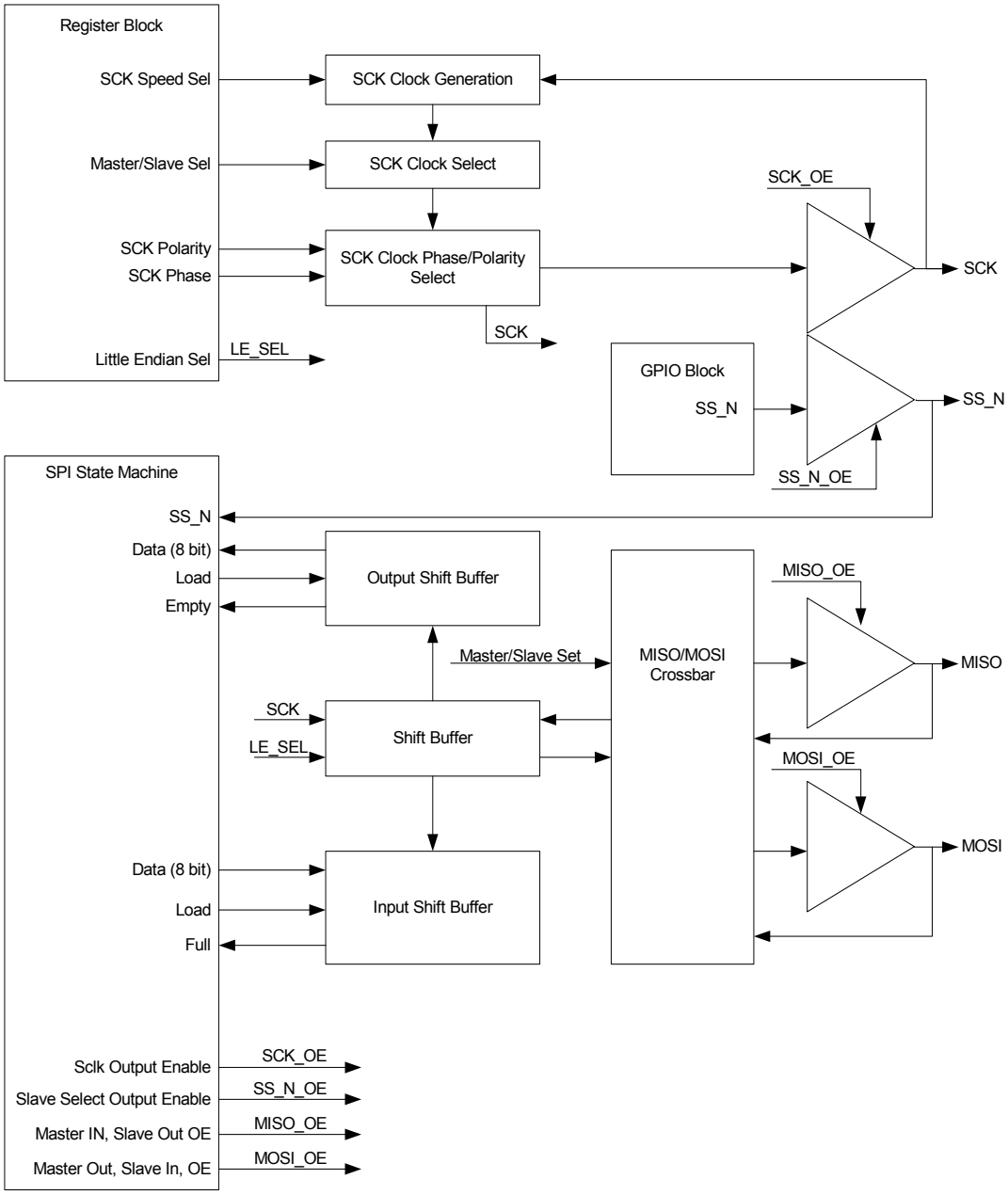


Table 18-4. SPI SCLK Frequency

SCLK Select	CPUCLK Divisor	SCLK Frequency when CPUCLK = 12 MHz
00	6	2 MHz
01	12	1 MHz
10	48	250 kHz
11	96	125 kHz

18.3 SPI Interface Pins

The SPI interface uses the P1.3–P1.6 pins. These pins are configured using the P1.3 and P1.4–P1.6 configuration.

19. Timer Registers

All timer functions of the enCoRe II LV are provided by a single-timer block. The timer block is asynchronous from the CPU clock. The 16-bit free-running counter is used as the time base for timer captures and also as a general time base by software.

19.1 Registers

19.1.1 Free-Running Counter

The 16-bit free-running counter is clocked by the timer capture clock (TCAPCLK). It is read in software for use as a general-purpose time base. When reading the low-order byte, the high-order byte is registered. Reading the high-order byte reads this register allowing the CPU to read the 16-bit value atomically (loads all bits at one time). The free-running timer generates an interrupt at 1024 µs rate when clocked by a 4-MHz source. It also generates an interrupt when the free-running counter overflow occurs – every 16.384 ms (with a 4-MHz source). This extends the length of the timer.

Figure 19-1. 16-Bit Free-Running Counter Block Diagram

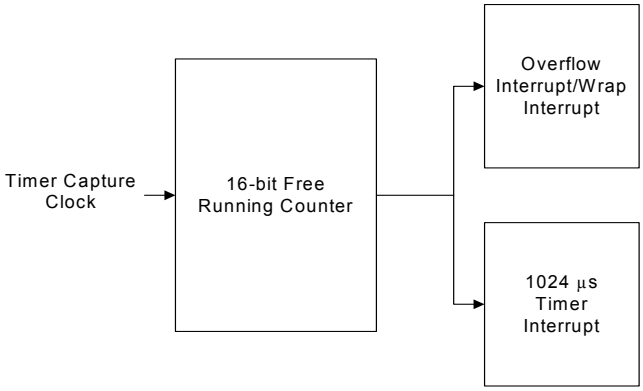


Table 19-1. Free-Running Timer Low-Order Byte (FRTMRL) [0x20] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Free-running Timer [7:0]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Bit [7:0]: Free-running timer [7:0]

This register holds the low-order byte of the 16-bit free-running timer. Reading this register moves the high-order byte into a holding register allowing an automatic read of all 16 bits simultaneously.

For reads, the actual read occurs in the cycle when the low-order is read. For writes, the actual time the write occurs is the cycle when the high-order is written.

When reading the free-running timer, the low-order byte is read first and the high-order second. When writing, the low-order byte is written first then the high-order byte.

Table 19-2. Free-Running Timer High-Order Byte (FRTMRH) [0x21] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	Free-running timer [15:8]							
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Bit [7:0]: Free-running timer [15:8]
When reading the free-running timer, the low-order byte is read first and the high-order second. When writing, the low-order byte is written first, then the high-order byte.

19.1.2 Time Capture

enCoRe II LV has two 8-bit captures. Each capture has a separate register for rising and falling time. The two 8-bit captures can be configured as a single 16-bit capture. When configured in this way, the capture 1 registers hold the high-order byte of the 16-bit timer capture value. Each of the four capture registers can be programmed to generate an interrupt when it is loaded.

Figure 19-2. Time Capture Block Diagram

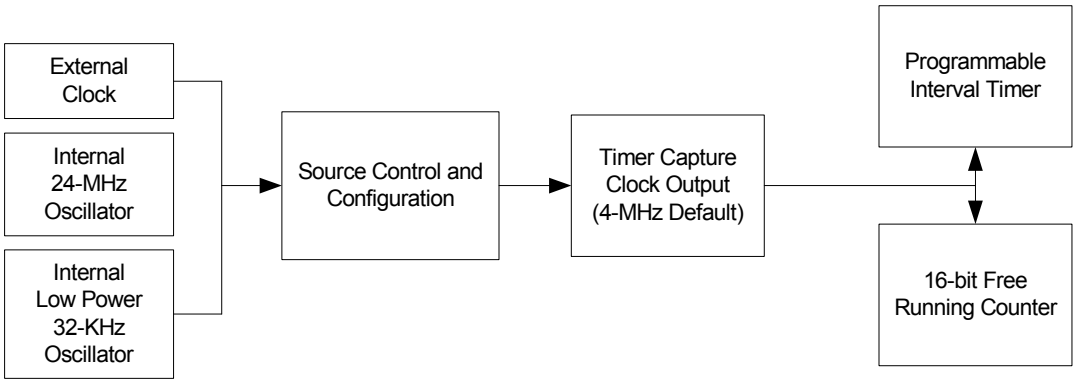


Table 19-1. Timer Configuration (TMRCR) [0x2A] [R/W]

Bit #	7	6	5	4	3	2	1	0
Field	First Edge Hold	8-bit Capture Prescale [2:0]			Cap0 16-bit Enable	Reserved		
Read/Write	R/W	R/W	R/W	R/W	R/W	–	–	–
Default	0	0	0	0	0	0	0	0

Bit 7: First edge hold
The first edge hold function applies to all four capture timers.
0 = The time of the most recent edge is held in the capture timer data register. If multiple edges have occurred since reading the capture timer, the time for the most recent one is read.
1 = The time of the first occurrence of an edge is held in the capture timer data register until the data is read. Subsequent edges are ignored until the capture timer data register is read.
Bit [6:4]: 8-bit capture prescale [2:0]
This field controls which eight bits of the 16 free-running timer are captured when in bit mode.
0 0 0 = capture timer[7:0]
0 0 1 = capture timer[8:1]
0 1 0 = capture timer[9:2]
0 1 1 = capture timer[10:3]
1 0 0 = capture timer[11:4]
1 0 1 = capture timer[12:5]
1 1 0 = capture timer[13:6]
1 1 1 = capture timer[14:7]
Bit 3: Cap0 16-bit Enable
0 = Capture 0 16-bit mode is disabled
1 = Capture 0 16-bit mode is enabled. Capture 1 is disabled and the Capture 1 rising and falling registers are used as an extension to the Capture 0 registers—extending them to 16 bits.
Bit [2:0]: Reserved

Figure 21-4. SPI Slave Timing, CPHA = 1

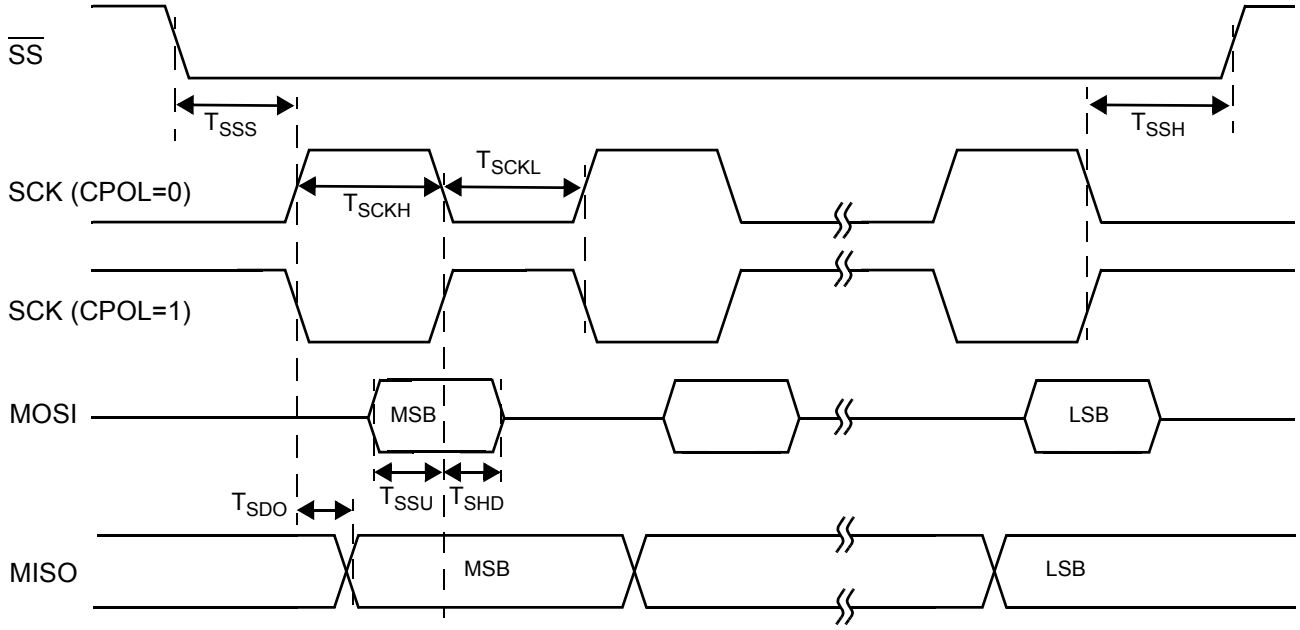
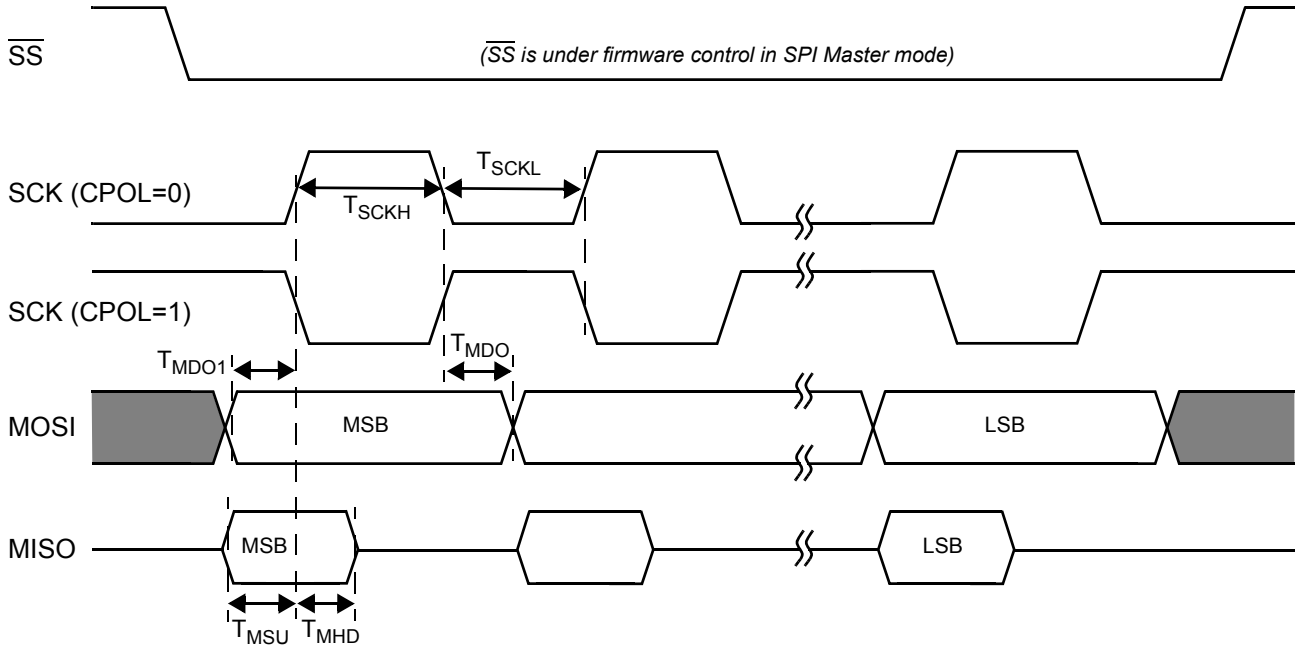


Figure 21-5. SPI Master Timing, CPHA = 0



25. Document History Page

Document Title: CY7C601xx/CY7C602xx, enCoRe™ II Low-Voltage Microcontroller Document Number: 38-16016				
Rev.	**ECN**	**Orig. of Change**	**Submission Date**	**Description of Change**
**	327601	BON	See ECN	New data sheet
*A	400134	BHA	See ECN	Updated Power consumption values Corrected Pin Assignment Table for 24 QSOP, 24 PDIP and 28 SSOP packages Minor text changes for clarification purposes Corrected INT_MSK0 and INT_MSK1 register address Corrected register bit definitions Corrected Protection Mode Settings in Table 10-7 Updated LVD Trip Point values Added Block diagrams for Timer functional timing Replaced TBD's with actual values Added SPI Block Diagram Added Timing Block Diagrams Removed CY7C60123 DIE from Figure 5-1 Removed CY7C60123-WXC from Section 22.0 Ordering Information Updated internal 24 MHz oscillator accuracy information Added information on sending/receiving data when using 32 KHz oscillator
*B	505222	TYJ	See ECN	Minor text changes GPIO capacitance and timing diagram included Method to clear Capture Interrupt Status bit discussed Sleep and Wakeup sequence documented PIT Timer registers' R/W capability corrected to read only Modified Free-running Counter text in section 17.1.1
*C	524104	KKVTMP	See ECN	Change title from Wireless enCoRe II to enCoRe II Low Voltage
*D	1821746	VGT / FSU / AESA	See ECN	Changed "High current drive" on GPIO pins to "2 mA source current on all GPIO pins". Changed the storage temperature from -40C to 90C in "Absolute Maximum ratings" section. Added the line "The GPIOs interrupts are edge-triggered." in Tables 19-2 and 19-6. Made timing changes in Table 43. Added Figure 12-1 (SROM Table) and text after it. Also modified Table 12-1 based on Figure 12-1 (SROM Table). Changed "CAPx" to "TIOx" in Tables 18-8 and 18-9. Changed "Capturex" to "TIOx" in Figure 18-3.
*E	2620679	CMCC / PYRS	12/12/08	Added Package Handling information Formatted code in Clocking section, Removed reference to external crystal oscillator in Tables 12-2 and 12-4
*F	2761532	DVJA	09/09/2009	Changed default value of the Sleep Timer from 00(512 Hz) to 01(64 Hz) in the OSC_CR0 [0x1E0] register.
*G	2899862	XUT	03/26/10	Removed obsolete parts from the ordering information table Updated package diagrams
*H	2978027	DATT	07/12/2010	Sunset review; no technical updates. Updated content to meet style guide and template requirements.
*I	2999570	MLIM	08/03/2010	Minor change to correct revision in the document footer.
*J	3275367	NXZ	06/06/2011	Removed "CY7C60223 24-pin PDIP and CY7C60113 28-pin SSOP" from Figure 7-1. Removed "28 SSOP" and "24 PDIP" columns from Table 7-1. Removed Figure 24-2 (24-pin PDIP) and Figure 24-4 (28-pin SSOP) Updated description field of P1.0 and P1.1 in Table 7-1 on page 5

Document Title: CY7C601xx/CY7C602xx, enCoRe™ II Low-Voltage Microcontroller Document Number: 38-16016				
Rev.	**ECN**	**Orig. of Change**	**Submission Date**	**Description of Change**
*K	4499620	SELV	09/11/2014	Updated Package Diagrams: spec 51-85025 – Changed revision from *E to *F. spec 51-85055 – Changed revision from *C to *D. spec 51-85019 – Changed revision from *B to *C. spec 51-85061 – Changed revision from *D to *F. Updated in new template. Completing Sunset Review.