CRG_U
Block Guide
V01.01

Original Release Date: 29 Mar 2004
Revised: 29 Nov 2004

TSPG 8/16 Bit MCU
Freescale Semiconductor, Inc.

Revision History

Version Number	Revision Date	Effective Date	Author	Description of Changes
V01.00	29 Mar 2004		Wai-On Law	Release version.
V01.01	29 Nov 2004		Wai-On Law	Changed company logo.

Table of Contents

Section 6 Interrupts

List of Figures

List of Tables

Section 1 Introduction

Figure 1-1 is a block diagram of CRG_U, showing the functional organization of the block, including all off-chip signals that are available to an end user.

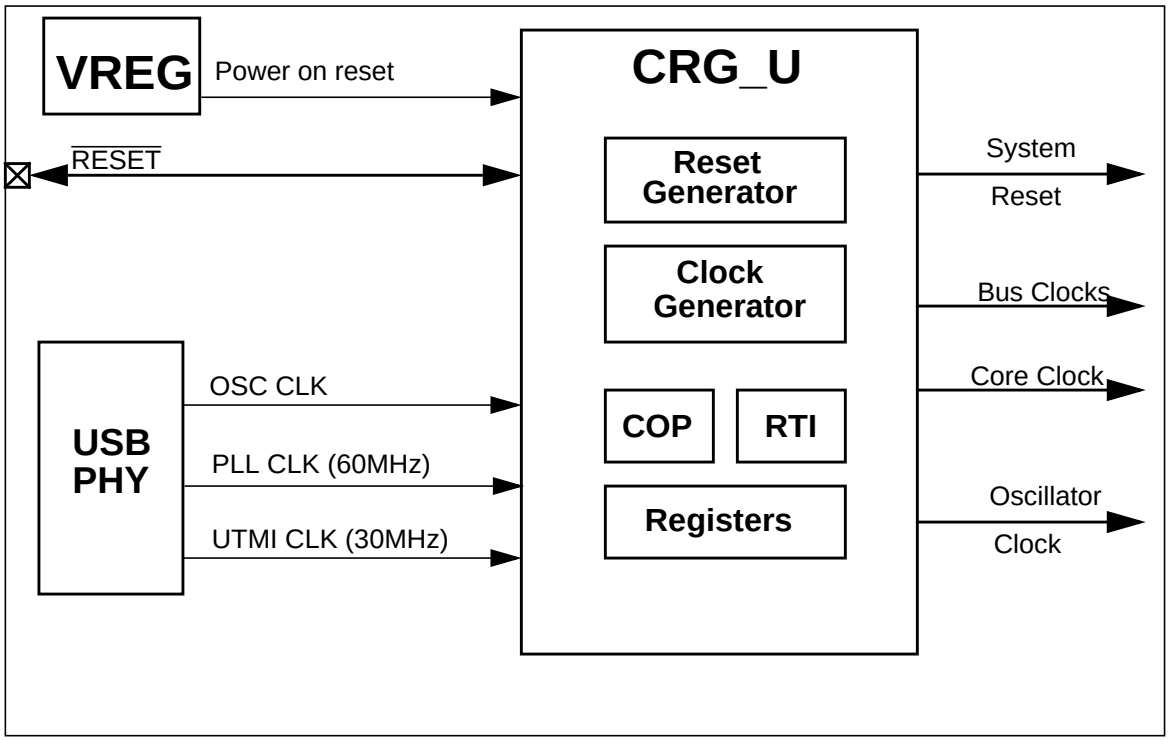


Figure 1-1 CRG_U Block Diagram

1.1 Overview

The CRG_U module implements the clock generator, the reset generator, and the real time interrupt.

1.2 Features

The CRG_U module includes these distinctive features:

- System Clock Generator
  - External clock mode
  - System clock switch
  - System clocks off during Wait Mode
- System Reset Generator

  - Computer Operating Properly (COP) watchdog timer with time-out clear window.
  - External pin reset
- Real-Time Interrupt (RTI)

1.3 Modes of Operation

This subsection lists and briefly describes all CRG_U operating modes supported by the CRG_U. This is a high level description only, detailed descriptions of operating modes are contained in later sections.

- Run Mode

  All functional parts of the CRG are running during normal Run Mode. If RTI or COP functionality is required, the individual bits of the associated rate select registers (COPCTL, RTICTL) have to be set to a non-zero value.

- Wait Mode

  Depending on the configuration of the individual bits in the CLKSEL register, this mode allows to disable the system and core clocks.

- Stop Mode

  All system and core clocks are stopped. The COP and the RTI remain frozen.

Section 2 External Signal Description

2.1 Overview

This section lists and describes the signals that connect off chip, and provide the necessary documentation for the end user of a system.

2.2 Detailed Signal Descriptions

2.2.1 $\overline{\text{RESET}}$

$\overline{\text{RESET}}$ is an active low bidirectional reset pin. As an input, it initializes the MCU asynchronously to a known start-up state. As an open-drain output, it indicates that a system reset (internal to MCU) has been triggered.

Section 3 Memory Map/Register Definition

Table 3-1 gives an overview of all CRG_U registers. Factory test control registers are in shaded portions of address maps and register diagram. Access to these registers is restricted to factory test mode only.

Table 3-1 CRG_U Memory Map

Address Offset	Use	Access
$_00	(Reserved)	-
$_01	CRG Reference Divider Register (REFDV)	R/W
$_02	CRG Test Flags Register (CTFLG)[1]	R/W
$_03	CRG Flags Register (CRGFLG)	R/W
$_04	CRG Interrupt Enable Register (CRGINT)	R/W
$_05	CRG Clock Select Register (CLKSEL)	R/W
$_06	(Reserved)	-
$_07	CRG RTI Control Register (RTICTL)	R/W
$_08	CRG COP Control Register (COPCTL)	R/W
$_09	CRG Force and Bypass Test Register (FORBYP)[2]	R/W
$_0A	CRG Test Control Register (CTCTL)[3]	R/W
$_0B	CRG COP Arm/Timer Reset (ARMCOP)	R/W

NOTES:
1. CTFLG is intended for factory test purposes only.
2. FORBYP is intended for factory test purposes only.
3. CTCTL is intended for factory test purposes only.

NOTE: *Register Address = Base Address + Address Offset, where the Base Address is defined at the MCU level and the Address Offset is defined at the module level.*

3.1 Register Descriptions

This section consists of register descriptions in address order.

3.1.1 CRG Reference Divider Register (REFDV)

Ther REFDV register controls the ratio of system clock. The count in the reference divider divides the system clock source frequency by REFDV+1. The system clock source is selected by the PLLSEL bit of CLKSEL.

Address Offset: $_01

	7	6	5	4	3	2	1	0
R	0	0	0	0	REFDV3	REFDV2	REFDV1	REFDV0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-1 CRG Reference Divider Register (REFDV)

Read: anytime

Write: anytime except when PLLSEL=1

3.1.2 Reserved Register (CTFLG)

This register is reserved for factory testing of the CRG_U module and is not available in normal modes.

Address Offset: $_02

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-2 Reserved Register (CTFLG)

Read: always reads $00 in normal modes

Write: unimplemented in normal modes

NOTE: *Writing to this register when in special mode can alter the CRG_U functionality.*

3.1.3 CRG Flags Register (CRGFLG)

This register provides CRG status bits and flags.

Address Offset: $_03

	7	6	5	4	3	2	1	0
R	RTIF	PORF	0	LOCKIF	LOCK	0	0	0
W								
RESET:	0	1	0	0	0	0	0	0

= Unimplemented or Reserved

NOTES:
1. PORF is set to 1 when a power on reset occurs. Unaffected by non-POR resets.

Figure 3-3 CRG Flags Register (CRGFLG)

Read: anytime

Write: refer to each bit for individual write conditions

RTIF — Real Time Interrupt Flag

RTIF is set to 1 at the end of the RTI period. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (RTIE=1), RTIF causes an interrupt request.

1 = RTI time-out has occurred.
0 = RTI time-out has not yet occurred.

PORF — Power on Reset Flag

PORF is set to 1 when a power on reset occurs. This flag can only be cleared by writing a 1. Writing a 0 has no effect.
1 = Power on reset has occurred.
0 = Power on reset has not occurred.

LOCKIF — PLL Lock Interrupt Flag

LOCKIF is set to 1 when LOCK status bit changes. This flag can only be cleared by writing a 1. Writing a 0 has no effect. If enabled (LOCKIF=1), LOCKIF causes an interrupt request.
1 = LOCK bit has changed.
0 = No change in LOCK bit.

LOCK — Lock Status Bit

LOCK reflects the current state of PLL lock condition. Writes have no effect.
1 = PLL is within the desired tolerance of the target frequency.
0 = PLL is not within the desired tolerance of the target frequency.

3.1.4 CRG Interrupt Enable Register (CRGINT)

This register enables CRG interrupt requests.

Address Offset: $_04

	7	6	5	4	3	2	1	0
R	RTIE	0	0	LOCKIE	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-4 CRG Interrupt Enable Register (CRGINT)

Read: anytime

Write: anytime

RTIE — Real Time Interrupt Enable Bit
1 = Interrupt will be requested whenever RTIF is set.
0 = Interrupt requests from RTI are disabled.

LOCKIE — Lock Interrupt Enable Bit
1 = Interrupt will be requested whenever LOCKIF is set.
0 = LOCK interrupt requests are disabled.

3.1.5 CRG Clock Select Register (CLKSEL)

This register controls CRG clock selection.

Address Offset: $_05

	7	6	5	4	3	2	1	0
R	PLLSEL	0	SYSWAI	ROAWAI	0	CWAI	RTIWAI	COPWAI
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-5 CRG Clock Select Register (CLKSEL)

Read: anytime

Write: refer to each bit for individual write conditions

PLLSEL — PLL Select Bit

Write: anytime.
1 = System clocks are derived from PLLCLK (i.e. USB PHY 60MHz clock).
0 = System clocks are derived from OSCCLK.

NOTE: *If CRGFLG:LOCK = 0, PLLSEL doesn't work and OSCLK is always chosen as the source of system clocks.*

SYSWAI — System clocks stop in Wait Mode Bit

Write: anytime
1 = In Wait Mode the system clocks stop.
0 = In Wait Mode the system clocks continue to run.

NOTE: *RTI and COP are not affected by SYSWAI bit.*

ROAWAI — Reduced Oscillator Amplitude in Wait Mode Bit

Write: anytime
1 = Reduced oscillator amplitude in Wait Mode.
0 = Normal oscillator amplitude in Wait Mode.

NOTE: *Lower oscillator amplitude exhibits lower power consumption but could have adverse effects during any Electro-Magnetic Susceptibility (EMS) tests.*

CWAI — Core stops in Wait Mode Bit

Write: anytime
1 = Core clock stops in Wait Mode.
0 = Core clock keeps running in Wait Mode.

RTIWAI — RTI stops in Wait Mode Bit

Write: anytime

1 = RTI stops and initializes the RTI dividers whenever the part goes into Wait Mode.
0 = RTI keeps running in Wait Mode.

COPWAI — COP stops in Wait Mode Bit

Normal modes: Write once

Special modes: Write anytime

1 = COP stops and initializes the COP dividers whenever the part goes into Wait Mode.
0 = COP keeps running in Wait Mode.

3.1.6 CRG RTI Control Register (RTICTL)

This register selects the time-out period for the Real Time Interrupt.

Address Offset: $_07

	7	6	5	4	3	2	1	0
R	0	RTR6	RTR5	RTR4	RTR3	RTR2	RTR1	RTR0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-6 CRG RTI Control Register (RTICTL)

Read: anytime

Write: anytime

NOTE: *A write to this register initializes the RTI counter.*

RTR[6:4] — Real Time Interrupt Prescale Rate Select Bits

These bits select the prescale rate for the RTI. See **Table 3-2**.

RTR[3:0] — Real Time Interrupt Modulus Counter Select Bits

These bits select the modulus counter target value to provide additional granularity.**Table 3-2** shows all possible divide values selectable by the RTICTL register. The source clock for the RTI is OSCCLK.

Table 3-2 RTI Frequency Divide Rates

RTR[3:0]	RTR[6:4] =							
	000 (OFF)	**001 (2^{10})**	**010 (2^{11})**	**011 (2^{12})**	**100 (2^{13})**	**101 (2^{14})**	**110 (2^{15})**	**111 (2^{16})**
0000 (÷1)	OFF*	2^{10}	2^{11}	2^{12}	2^{13}	2^{14}	2^{15}	2^{16}

Table 3-2 RTI Frequency Divide Rates

RTR[3:0]	RTR[6:4] =							
0001 (÷2)	OFF*	$2x2^{10}$	$2x2^{11}$	$2x2^{12}$	$2x2^{13}$	$2x2^{14}$	$2x2^{15}$	$2x2^{16}$
0010 (÷3)	OFF*	$3x2^{10}$	$3x2^{11}$	$3x2^{12}$	$3x2^{13}$	$3x2^{14}$	$3x2^{15}$	$3x2^{16}$
0011 (÷4)	OFF*	$4x2^{10}$	$4x2^{11}$	$4x2^{12}$	$4x2^{13}$	$4x2^{14}$	$4x2^{15}$	$4x2^{16}$
0100 (÷5)	OFF*	$5x2^{10}$	$5x2^{11}$	$5x2^{12}$	$5x2^{13}$	$5x2^{14}$	$5x2^{15}$	$5x2^{16}$
0101 (÷6)	OFF*	$6x2^{10}$	$6x2^{11}$	$6x2^{12}$	$6x2^{13}$	$6x2^{14}$	$6x2^{15}$	$6x2^{16}$
0110 (÷7)	OFF*	$7x2^{10}$	$7x2^{11}$	$7x2^{12}$	$7x2^{13}$	$7x2^{14}$	$7x2^{15}$	$7x2^{16}$
0111 (÷8)	OFF*	$8x2^{10}$	$8x2^{11}$	$8x2^{12}$	$8x2^{13}$	$8x2^{14}$	$8x2^{15}$	$8x2^{16}$
1000 (÷9)	OFF*	$9x2^{10}$	$9x2^{11}$	$9x2^{12}$	$9x2^{13}$	$9x2^{14}$	$9x2^{15}$	$9x2^{16}$
1001 (÷10)	OFF*	$10x2^{10}$	$10x2^{11}$	$10x2^{12}$	$10x2^{13}$	$10x2^{14}$	$10x2^{15}$	$10x2^{16}$
1010 (÷11)	OFF*	$11x2^{10}$	$11x2^{11}$	$11x2^{12}$	$11x2^{13}$	$11x2^{14}$	$11x2^{15}$	$11x2^{16}$
1011 (÷12)	OFF*	$12x2^{10}$	$12x2^{11}$	$12x2^{12}$	$12x2^{13}$	$12x2^{14}$	$12x2^{15}$	$12x2^{16}$
1100 (÷ 13)	OFF*	$13x2^{10}$	$13x2^{11}$	$13x2^{12}$	$13x2^{13}$	$13x2^{14}$	$13x2^{15}$	$13x2^{16}$
1101 (÷14)	OFF*	$14x2^{10}$	$14x2^{11}$	$14x2^{12}$	$14x2^{13}$	$14x2^{14}$	$14x2^{15}$	$14x2^{16}$
1110 (÷15)	OFF*	$15x2^{10}$	$15x2^{11}$	$15x2^{12}$	$15x2^{13}$	$15x2^{14}$	$15x2^{15}$	$15x2^{16}$
1111 (÷ 16)	OFF*	$16x2^{10}$	$16x2^{11}$	$16x2^{12}$	$16x2^{13}$	$16x2^{14}$	$16x2^{15}$	$16x2^{16}$

* Denotes the default value out of reset.This value should be used to disable the RTI to ensure future backwards compatibility.

3.1.7 CRG COP Control Register (COPCTL)

This register controls the COP (Computer Operating Properly) watchdog.

Address Offset: $_08

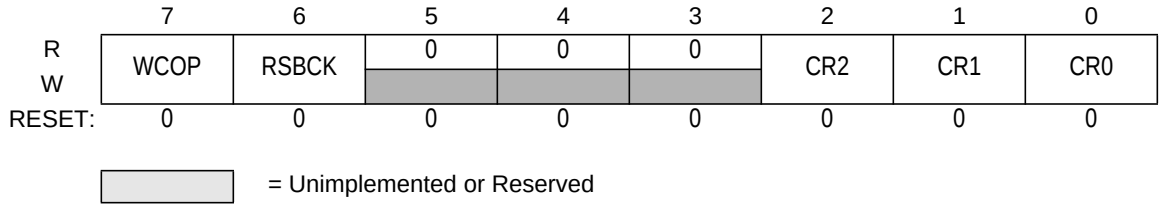


Figure 3-7 CRG COP Control Register (COPCTL)

Read: anytime

Write: once in user mode, anytime in special mode

WCOP — Window COP Mode Bit

When set, a write to the ARMCOP register must occur in the last 25% of the selected period. A write during the first 75% of the selected period will reset the part. As long as all writes occur during this window, $55 can be written as often as desired. Once $AA is written after the $55, the time-out logic restarts and the user must wait until the next window before writing to ARMCOP. **Table 3-3** shows the exact duration of this window for the seven available COP rates.

1 = Window COP operation
0 = Normal COP operation

RSBCK — COP and RTI stop in Active BDM mode Bit

1 = Stops the COP and RTI counters whenever the part is in Active BDM mode.
0 = Allows the COP and RTI to keep running in Active BDM mode.

CR[2:0] — COP Watchdog Timer Rate select

These bits select the COP time-out rate (see **Table 3-3**). The COP time-out period is OSCCLK period divided by CR[2:0] value. Writing a nonzero value to CR[2:0] enables the COP counter and starts the time-out period. A COP counter time-out causes a system reset. This can be avoided by periodically (before time-out) re-initializing the COP counter via the ARMCOP register.

Table 3-3 COP Watchdog Rates[1]

CR2	CR1	CR0	OSCCLK cycles to time-out
0	0	0	COP disabled
0	0	1	2^{14}
0	1	0	2^{16}
0	1	1	2^{18}
1	0	0	2^{20}
1	0	1	2^{22}
1	1	0	2^{23}
1	1	1	2^{24}

NOTES:
1. OSCCLK cycles are referenced from the previous COP time-out reset (writing $55/$AA to the ARMCOP register)

3.1.8 Reserved Register (FORBYP)

NOTE: *This reserved register is designed for factory test purposes only, and is not intended for general user access. Writing to this register when in special modes can alter the CRG_U's functionality.*

Address Offset: $_09

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-8 Reserved Register (FORBYP)

Read: always read $00 except in special modes

Write: only in special modes

3.1.9 Reserved Register (CTCTL)

NOTE: *This reserved register is designed for factory test purposes only, and is not intended for general user access. Writing to this register when in special test modes can alter the CRG_U's functionality*

Address Offset: $_0A

	7	6	5	4	3	2	1	0
R	1	0	0	0	0	0	0	0
W								
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-9 Reserved Register (CTCTL)

Read: always read $80 except in special modes

Write: only in special modes

3.1.10 CRG COP Timer Arm/Reset Register (ARMCOP)

This register is used to restart the COP time-out period.

Address Offset: $_0B

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	0	0
W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RESET:	0	0	0	0	0	0	0	0

= Unimplemented or Reserved

Figure 3-10 ARMCOP Register Diagram

Read: always reads $00

Write: anytime

When the COP is disabled (CR[2:0] = “000”) writing to this register has no effect.

When the COP is enabled by setting CR[2:0] nonzero, the following applies:

Writing any value other than $55 or $AA causes a COP reset. To restart the COP time-out period you must write $55 followed by a write of $AA. Other instructions may be executed between these writes but the sequence ($55, $AA) must be completed prior to COP end of time-out period to avoid a COP reset. Sequences of $55 writes or sequences of $AA writes are allowed. When the WCOP bit is set, $55 and $AA writes must be done in the last 25% of the selected time-out period; writing any value in the first 75% of the selected period will cause a COP reset.

Section 4 Functional Description

This section provide a complete functional description of the CRG_U. It gives detailed information on the internal operation of the design.

4.1 COP - Computer Operating Properly Watchdog

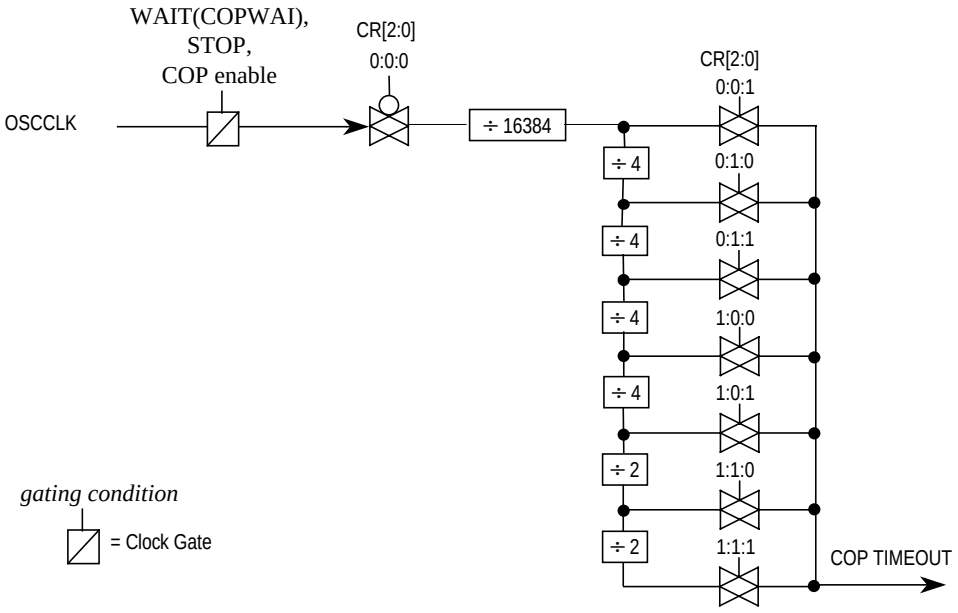


Figure 4-1 Clock Chain for COP

The COP (free running watchdog timer) enables the user to check that a program is running and sequencing properly When the COP is being used, software is responsible for keeping the COP from timing out. If the COP times out it is an indication that the software is no longer being executed in the intended sequence; thus a system reset is initiated (see **5.2.1 Computer Operating Properly Watchdog (COP) Reset**). The COP runs with a gated OSCCLK (see **Figure 4-1 Clock Chain for COP**). Three control bits in the COPCTL register allow selection of seven COP time-out periods.

When COP is enabled, the program must write $55 and $AA (in this order) to the ARMCOP register during the selected time-out period. Once this is done, the COP time-out period is restarted. If the program fails to do this and the COP times out, the part will reset. Also, if any value other than $55 or $AA is written, the part is immediately reset.

Windowed COP operation is enabled by setting WCOP in the COPCTL register. In this mode, writes to the ARMCOP register to clear the COP timer must occur in the last 25% of the selected time-out period. A premature write will immediately reset the part.

4.2 Real Time Interrupt (RTI)

The RTI can be used to generate a hardware interrupt at a fixed periodic rate. If enabled (by setting RTIE=1), this interrupt will occur at the rate selected by the RTICTL register. The RTI runs with a gated

OSCCLK (see **Figure 4-2 Clock Chain for RTI.**). At the end of the RTI time-out period, the RTIF flag is set to one and a new RTI time-out period starts immediately.

A write to the RTICTL register restarts the RTI time-out period.

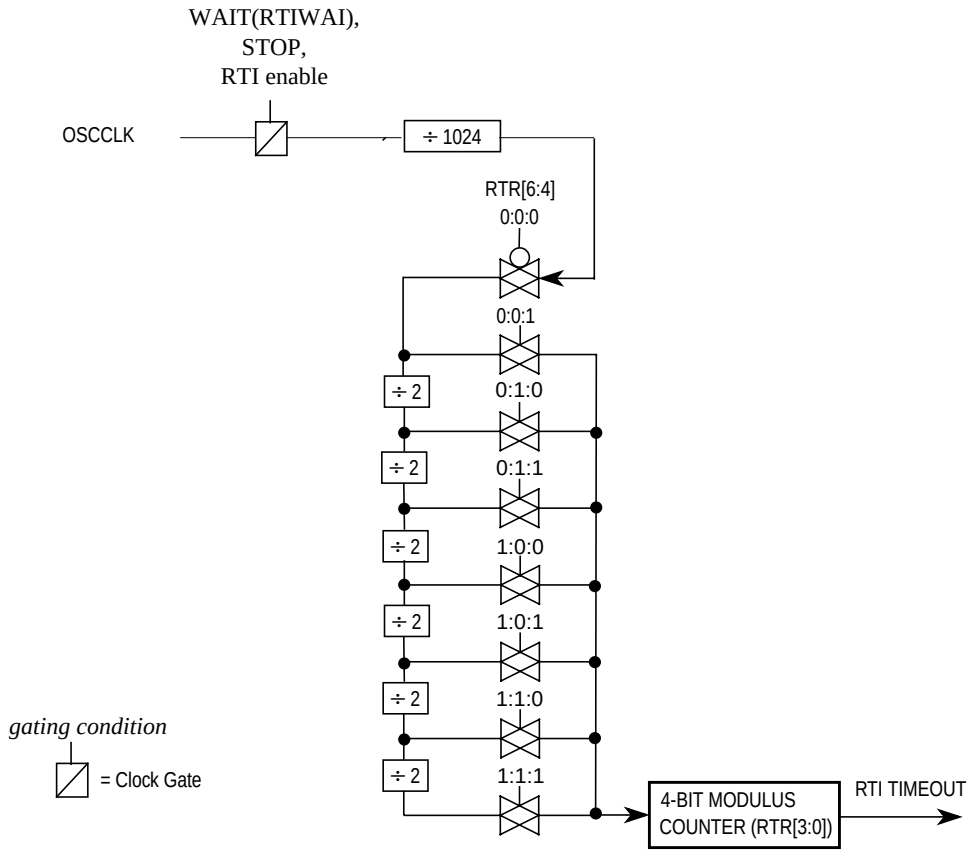


Figure 4-2 Clock Chain for RTI.

4.3 Low Power Options

This section summarizes the low power options available in the CRG_U.

4.3.1 Run Mode

The RTI can be stopped by setting the associated rate select bits to zero

The COP can be stopped by setting the associated rate select bits to zero

4.3.2 Wait Mode

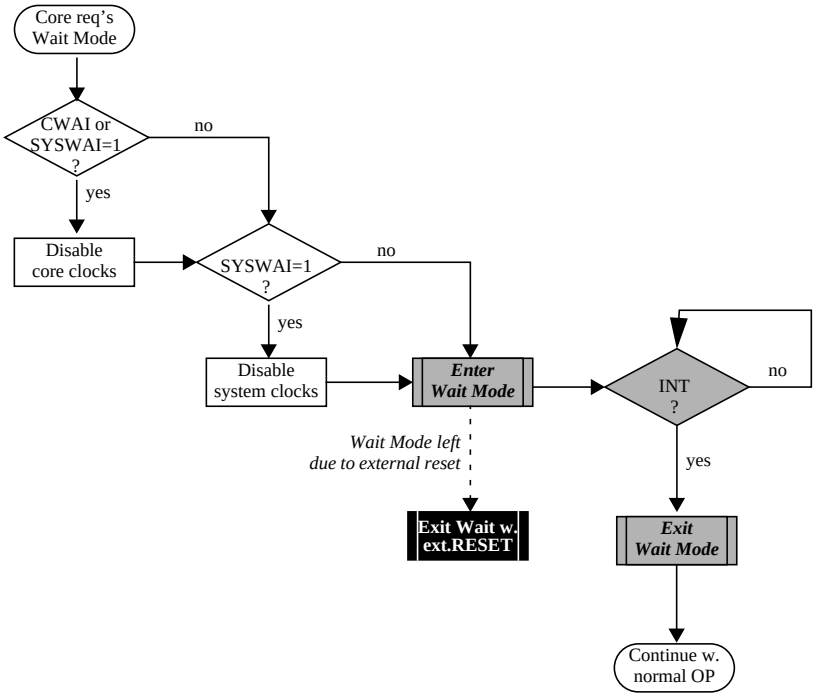
The WAI instruction puts the MCU in a low power consumption stand-by mode depending on setting of the individual bits in the CLKSEL register. All individual Wait Mode configuration bits can be superposed. This provides enhanced granularity in reducing the level of power consumption during Wait Mode. **Table 4-1** lists the individual configuration bits and the parts of the MCU that are affected in Wait Mode.

Table 4-1 MCU configuration during Wait Mode

	CWAI	SYSWAI	RTIWAI	COPWAI
Core	stopped	stopped	-	-
System	-	stopped	-	-
RTI	-	-	stopped	-
COP	-	-	-	stopped

After executing the WAI instruction, the core requests the CRG_U to switch MCU into Wait Mode. The CRG_U then checks whether the CWAI and SYSWAI bits are asserted (see **Figure 4-3 Wait Mode Entry/Exit Sequence**). Depending on the configuration, the CRG_U switches the system and core clocks to OSCCLK by clearing the PLLSEL bit, disables the PLL, disables the core clocks and finally disables the remaining system clocks. As soon as all clocks are switched off, Wait Mode is active.

Figure 4-3 Wait Mode Entry/Exit Sequence



There are five different scenarios for the CRG_U to restart the MCU from Wait Mode:

- External Reset
- COP Reset
- RTI

In case the MCU gets an external reset during Wait Mode active, the CRG_U asynchronously restores all configuration bits in the register space to its default settings and starts the reset generator. After completing the reset sequence, processing begins by fetching the normal reset vector. Wait Mode is left and the MCU is in Run Mode again.

*I*In case any other interrupt source (e.g. RTI) triggers exit from Wait Mode, the MCU immediately continues with normal operation. If the PLL has been powered-down during Wait-Mode, the PLLSEL bit is cleared and the MCU runs on OSCCLK after leaving Wait-Mode. The software must manually set the PLLSEL bit again, in order to switch system and core clocks to the PLLCLK.

4.3.3 CPU Stop Mode

All clocks are stopped in STOP mode. All counters and dividers remain frozen but do not initialize. In addition to disabling system and core clocks, the CRG_U requests other functional units of the MCU (e.g. voltage-regulator) to enter their individual power-saving modes (if available). As soon as all clocks are switched off, Stop-Mode is active.

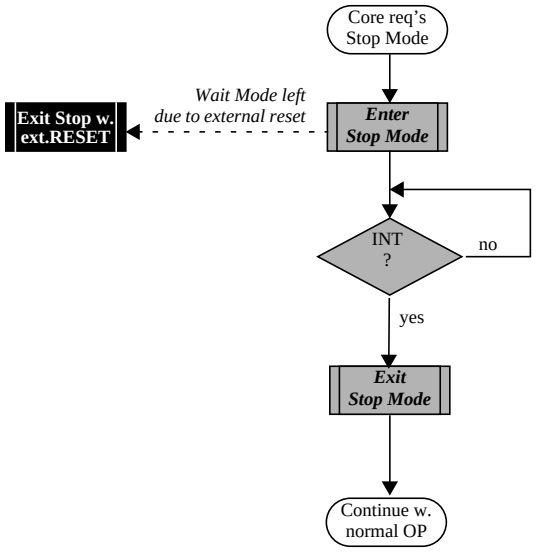


Figure 4-4 Stop Mode Entry/Exit Sequence

4.3.3.1 Wake-up from Stop Mode

The MCU requires an interrupt or an external reset in order to wake-up from Stop-Mode.

In case the MCU gets an external reset during Full Stop Mode active, the CRG_U asynchronously restores all configuration bits in the register space to its default settings and starts the reset generator. After

completing the reset sequence, processing begins by fetching the normal reset vector. Full Stop-Mode is left and the MCU is in Run Mode again.

If the MCU is woken up by an interrupt, the CRG_U will release all system and core clocks and will continue with normal operation.

Section 5 Resets

5.1 General

This section describes how to reset the CRG_U and how the CRG_U itself controls the reset of the MCU. It explains all special reset requirements. Since the reset generator for the MCU is part of the CRG_U, this section also describes all automatic actions that occur during or as a result of individual reset conditions. The reset values of registers and signals are provided in **Section 3 Memory Map/Register Definition**. All reset sources are listed in **Table 5-1**. Refer to MCU specification for related vector addresses and priorities.

Table 5-1Reset Summary

Reset Source	Local Enable
Power-on Reset	None
External Reset	None
COP Watchdog Reset	COPCTL (CR[2:0] nonzero)

5.2 Description of Reset Operation

The reset sequence is initiated by any of the following events:

- Low level is detected at the $\overline{\text{RESET}}$ pin (External Reset).
- Power-on is detected.
- COP watchdog times out.

Upon detection of any reset event, an internal circuit drives the $\overline{\text{RESET}}$ pin low for 128 SYSCLK cycles (see **Figure 5-1**). Since entry into reset is asynchronous it does not require a running SYSCLK. However, the internal reset circuit of the CRG_U cannot sequence out of current reset condition without a running SYSCLK. The number of 128 SYSCLK cycles might be increased by n=3 to 6 additional SYSCLK cycles depending on the internal synchronization latency. After 128+n SYSCLK cycles the $\overline{\text{RESET}}$ pin is released. The reset generator of the CRG_U waits for additional 64 SYSCLK cycles and then samples the RESET pin to determine the originating source. **Table 5-2** shows which vector will be fetched.

Table 5-2 Reset Vector Selection

sampled $\overline{\text{RESET}}$ pin (64 cycles after release)	COP Reset pending	Vector fetch
1	0	POR / External Reset
1	1	COP Reset

Table 5-2 Reset Vector Selection

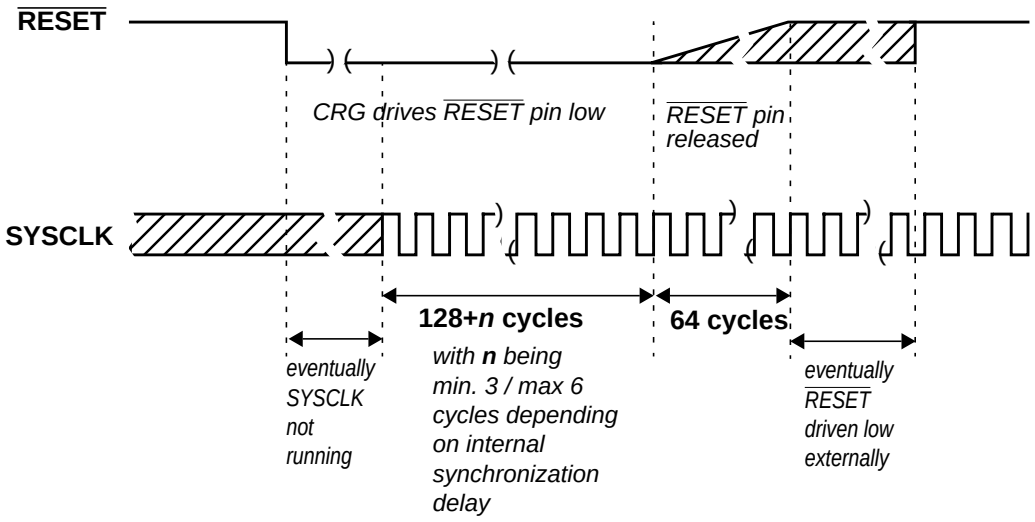
sampled $\overline{\text{RESET}}$ pin (64 cycles after release)	COP Reset pending	Vector fetch
0	X	POR / External Reset with rise of $\overline{\text{RESET}}$ pin

NOTE: *External circuitry connected to the $\overline{RESET}$ pin should not include a large capacitance that would interfere with the ability of this signal to rise to a valid logic one within 64 SYSCLK cycles after the low drive is released.*

The internal reset of the MCU remains asserted while the reset generator completes the 192 SYSCLK long reset sequence. The reset generator circuitry always makes sure the internal reset is de-asserted synchronously after completion of the 192 SYSCLK cycles. In case the $\overline{\text{RESET}}$ pin is externally driven low for more than these 192 SYSCLK cycles (External Reset), the internal reset remains asserted too.

Also, upon reset, CRG_U always uses the OSC as the clock source for deriving all system clocks because the PHY will be reset upon reset and there will be no PHY clock.

Figure 5-1 RESET Timing



5.2.1 Computer Operating Properly Watchdog (COP) Reset

When COP is enabled, the CRG_U expects sequential write of $55 and $AA (in this order) to the ARMCOP register during the selected time-out period. Once this is done, the COP time-out period restarts. If the program fails to do this, the CRG_U will generate a reset. Also, if any value other than $55 or $AA is written, the CRG_U immediately generates a reset. In case windowed COP operation is enabled, writes ($55 or $AA) to the ARMCOP register must occur in the last 25% of the selected time-out period. In case of a premature write, the CRG_U will immediately generate a reset.

As soon as the reset sequence is completed, the reset generator checks the reset condition. If no clock monitor failure is indicated and the latched state of the COP timeout is true, processing begins by fetching the COP vector.

5.2.2 Power-On Reset

An on-chip power-on detector circuit detects when VDD to the MCU has reached a certain level and asserts reset to the internal circuits. The detector circuit is triggered by the slew rate.**Figure 5-2** and **Figure 5-3** show the power-up sequence for cases when the $\overline{\text{RESET}}$ pin is tied to VDD and when the $\overline{\text{RESET}}$ pin is held low.

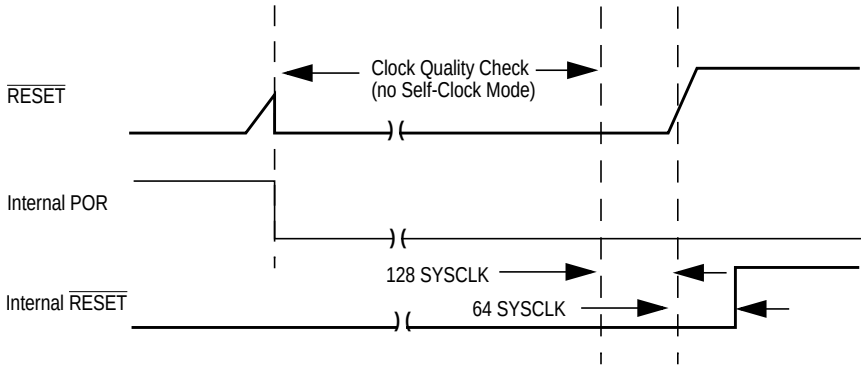


Figure 5-2 $\overline{\text{RESET}}$ pin tied to VDD (by a pull-up resistor)

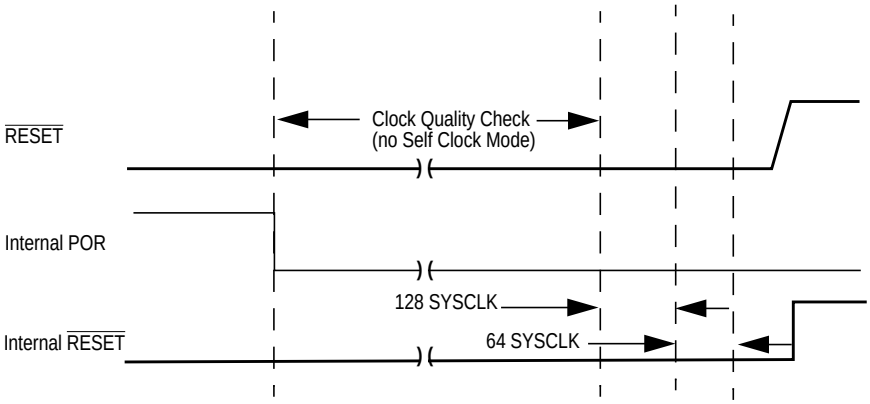


Figure 5-3 $\overline{\text{RESET}}$ pin held low externally

Section 6 Interrupts

6.1 General

This section describes all interrupts originated by the CRG_U.

The interrupts/reset vectors requested by the CRG_U are listed in **Table 6-1**. Refer to MCU specification for related vector addresses and priorities.

Table 6-1 CRG Interrupt Vectors

Interrupt Source	CCR Mask	Local Enable
Real time interrupt	I bit	CRGINT (RTIE)
LOCK interrupt	1 bit	CRGINT (LOCKIE)

6.2 Description of Interrupt Operation

6.2.1 Real Time Interrupt

The CRG_U generates a real time interrupt when the selected interrupt time period elapses. RTI interrupts are locally disabled by setting the RTIE bit to zero. The real time interrupt flag (RTIF) is set to 1 when a timeout occurs, and is cleared to 0 by writing a 1 to the RTIF bit.

6.2.2 PLL Lock Interrupt

The CRG_U generates a PLL Lock interrupt when the LOCK condition of the PLL (i.e. PHY 60MHz clock) has changed, either from a locked state to an unlocked state or vice versa. Lock interrupts are locally disabled by setting LOCKIE bit to zero. The PLL Lock interrupt flag (LOCKIF) is set to 1 when the LOCK condition has changed, and is cleared to 0 by writing a 1 to the LOCKIF bit.

Index