Xilinx
Zynq FPGA
TI DSP MCU 기반의
프로그래밍 및 회로 설계 전문가

강사 이상훈
gcccompil3r@gmail.com

학생 김민호
minking12@naver.com

17.3.1 RTI Global Control Register (RTIGCTRL)

The global control register starts/stops the counters and selects the signal compared with the timebase control circuit. This register is shown in Figure 17-12 and described in Table 17-2.

Figure 17-12. RTI Global Control Register (RTIGCTRL) [offset = 00]

31	20	19	16
Reserved		NTUSEL	
R-0		R/WP-0	

15	14	2	1	0
COS	Reserved		CNT1EN	CNT0EN
R/WP-0	R-0		R/WP-0	R/WP-0

LEGEND: R/W = Read/Write; R = Read only; WP = Write in privileged mode only; -*n* = value after reset

Table 17-2. RTI Global Control Register (RTIGCTRL) Field Descriptions

Bit	Field	Value	Description
31-20	Reserved	0	Reads return 0. Writes have no effect.
19-16	NTUSEL		Select NTU signal. These bits determine which NTU input signal is used as external timebase
		0h	NTU0
		5h	NTU1
		Ah	NTU2
		Fh	NTU3
		All other values	Tied to 0
15	COS		Continue on suspend. This bit determines if both counters are stopped when the device goes into halting debug mode or if they continue counting.
		0	Counters are stopped while in halting debug mode.
		1	Counters are running while in halting debug mode.
14-2	Reserved	0	Reads return 0. Writes have no effect.
1	CNT1EN		Counter 1 enable. This bit starts and stops counter block 1 (RTIUC1 and RTIFRC1).
		0	Counter block 1 is stopped.
		1	Counter block 1 is running.
0	CNT0EN		Counter 0 enable. This bit starts and stops counter block 0 (RTIUC0 and RTIFRC0).
		0	Counter block 0 is stopped.
		1	Counter block 0 is running.

NOTE: If the application uses the timebase circuit for synchronization between the communications controller and the operating system and the device enters halting debug mode, the synchronization may be lost depending on the COS setting in the RTI module and the halting debug mode behavior of the communications controller.

17.3.2 RTI Timebase Control Register (RTITBCTRL)

The timebase control register selects if the free running counter 0 is incremented by RTICLK or NTU. This register is shown in Figure 17-13 and described in Table 17-3.

Figure 17-13. RTI Timebase Control Register (RTITBCTRL) [offset = 04h]

31	8
Reserved	
R-0	

7	2	1	0
Reserved		INC	TBEXT
R-0		R/WP-0	R/WP-0

LEGEND: R/W = Read/Write; R = Read only; WP = Write in privileged mode only; -*n* = value after reset

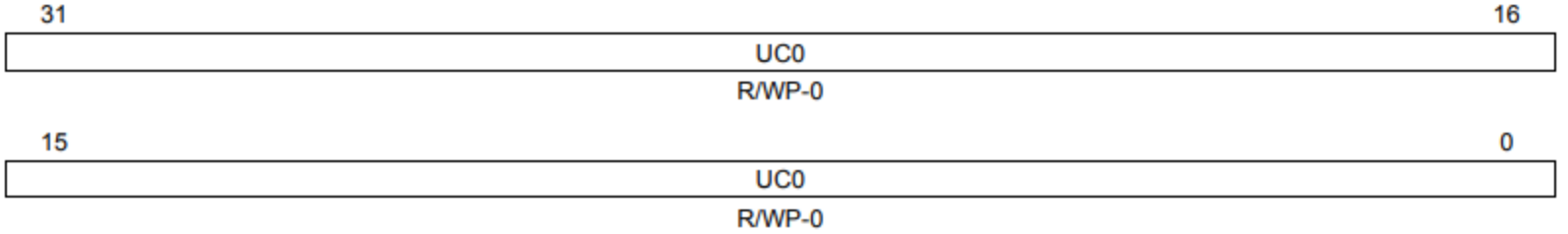
Table 17-3. RTI Timebase Control Register (RTITBCTRL) Field Descriptions

Bit	Field	Value	Description
31-2	Reserved	0	Reads return 0. Writes have no effect.
1	INC		Increment free running counter 0. This bit determines whether the free running counter 0 (RTIFRC0) is automatically incremented if a failing clock on the NTU signal is detected.
		0	RTIFRC0 will not be incremented on a failing external clock.
		1	RTIFRC0 will be incremented on a failing external clock.
0	TBEXT		Timebase external. This bit selects whether the free running counter 0 (RTIFRC0) is clocked by the internal up counter 0 (RTIUC0) or from the external signal NTU. Setting the TBEXT bit from 0 to 1 will not increment RTIFRC0, since RTIUC0 is reset. When the timebase supervisor circuit detects a missing clock edge, then the TBEXT bit is reset. Only the software can select whether the external signal should be used.
		0	RTIUC0 clocks RTIFRC0.
		1	NTU clocks RTIFRC0.

17.3.6 RTI Up Counter 0 Register (RTIUC0)

The up counter 0 register holds the current value of prescale counter. This register is shown in Figure 17-17 and described in Table 17-7.

Figure 17-17. RTI Up Counter 0 Register (RTIUC0) [offset = 14h]



LLEGEND: R/W = Read/Write; WP = Write in privileged mode only; -*n* = value after reset

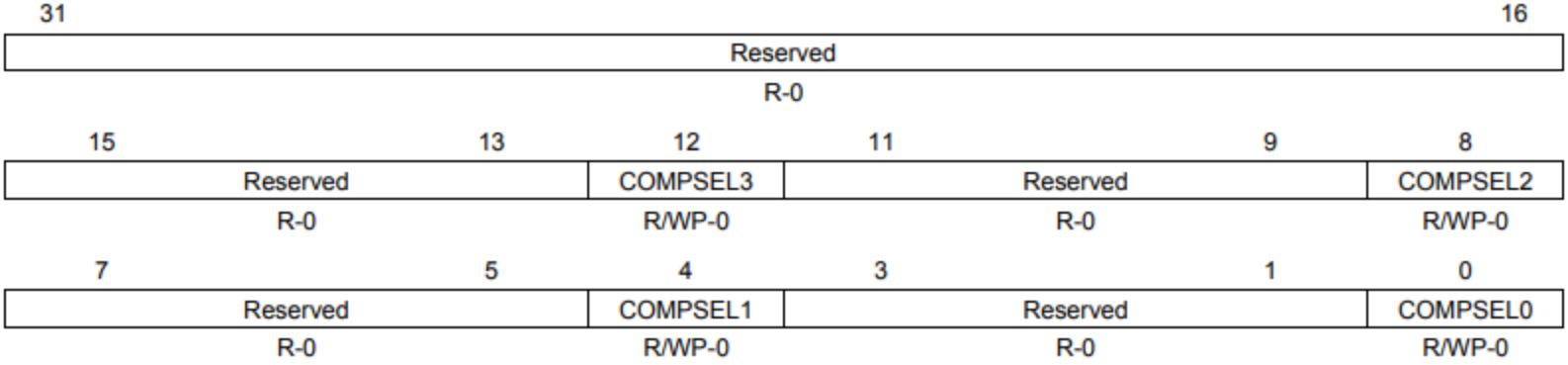
Table 17-7. RTI Up Counter 0 Register (RTIUC0) Field Descriptions

Bit	Field	Value	Description
31-0	UC0	0-FFFF FFFFh	Up counter 0. This register holds the current value of the up counter 0 and prescales the RTI clock. It will be only updated by a previous read of free running counter 0 (RTIFRC0). This method of updating effectively gives a 64-bit read of both counters, without having the problem of a counter being updated between two consecutive reads on up counter 0 (RTIUC0) and free running counter 0 (RTIFRC0). A read of this counter returns the value of the counter at the time RTIFRC0 was read. A write to this counter presets it with a value. The counter then increments from this written value upwards. Note: If counters must be preset, they must be disabled in the RTIGCTRL register to ensure consistency between RTIUC0 and RTIFRC0. **Note: If the preset value is bigger than the compare value stored in register RTICPUC0, then it can take a long time until a compare matches, since RTIUC0 has to count up until it overflows.**

17.3.4 RTI Compare Control Register (RTICOMPCTRL)

The compare control register controls the source for the compare registers. This register is shown in Figure 17-15 and described in Table 17-5.

Figure 17-15. RTI Compare Control Register (RTICOMPCTRL) [offset = 0Ch]



LEGEND: R/W = Read/Write; R = Read only; WP = Write in privileged mode only; *-n* = value after reset

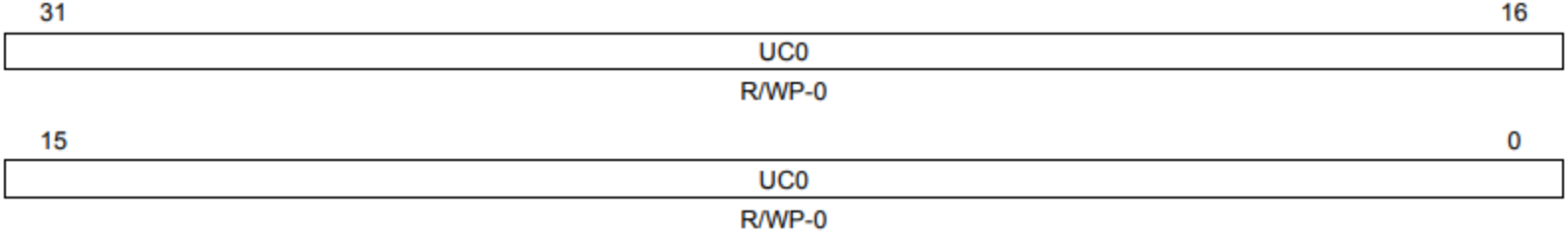
Table 17-5. RTI Compare Control Register (RTICOMPCTRL) Field Descriptions

Bit	Field	Value	Description
31-13	Reserved	0	Reads return 0. Writes have no effect.
12	COMPSEL3		Compare select 3. This bit determines the counter with which the compare value held in compare register 3 (RTICOMP3) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
11-9	Reserved	0	Reads return 0. Writes have no effect.
8	COMPSEL2		Compare select 2. This bit determines the counter with which the compare value held in compare register 2 (RTICOMP2) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
7-5	Reserved	0	Reads return 0. Writes have no effect.
4	COMPSEL1		Compare select 1. This bit determines the counter with which the compare value held in compare register 1 (RTICOMP1) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.
3-1	Reserved	0	Reads return 0. Writes have no effect.
0	COMPSEL0		Compare select 0. This bit determines the counter with which the compare value held in compare register 0 (RTICOMP0) is compared.
		0	Value will be compared with RTIFRC0.
		1	Value will be compared with RTIFRC1.

17.3.6 RTI Up Counter 0 Register (RTIUC0)

The up counter 0 register holds the current value of prescale counter. This register is shown in Figure 17-17 and described in Table 17-7.

Figure 17-17. RTI Up Counter 0 Register (RTIUC0) [offset = 14h]



LLEGEND: R/W = Read/Write; WP = Write in privileged mode only; -*n* = value after reset

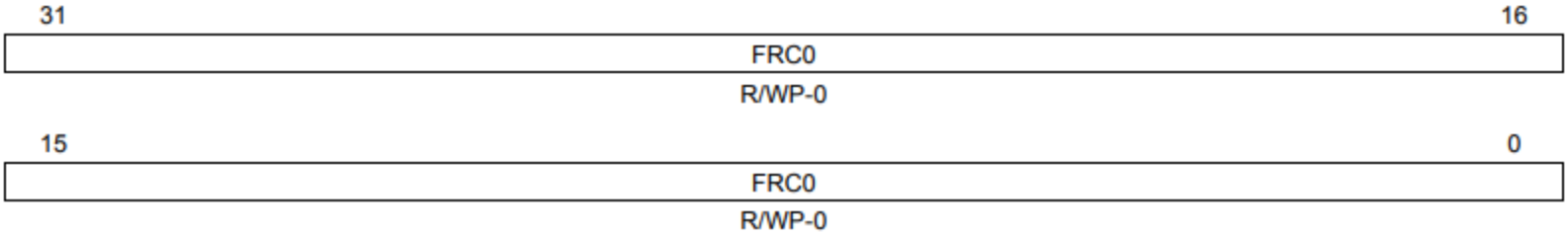
Table 17-7. RTI Up Counter 0 Register (RTIUC0) Field Descriptions

Bit	Field	Value	Description
31-0	UC0	0-FFFF FFFFh	Up counter 0. This register holds the current value of the up counter 0 and prescales the RTI clock. It will be only updated by a previous read of free running counter 0 (RTIFRC0). This method of updating effectively gives a 64-bit read of both counters, without having the problem of a counter being updated between two consecutive reads on up counter 0 (RTIUC0) and free running counter 0 (RTIFRC0). A read of this counter returns the value of the counter at the time RTIFRC0 was read. A write to this counter presets it with a value. The counter then increments from this written value upwards. Note: If counters must be preset, they must be disabled in the RTIGCTRL register to ensure consistency between RTIUC0 and RTIFRC0. **Note: If the preset value is bigger than the compare value stored in register RTICPUC0, then it can take a long time until a compare matches, since RTIUC0 has to count up until it overflows.**

17.3.5 RTI Free Running Counter 0 Register (RTIFRC0)

The free running counter 0 register holds the current value of free running counter 0. This register is shown in Figure 17-16 and described in Table 17-6.

Figure 17-16. RTI Free Running Counter 0 Register (RTIFRC0) [offset = 10h]



LEGEND: R/W = Read/Write; WP = Write in privileged mode only; *-n* = value after reset

Table 17-6. RTI Free Running Counter 0 Register (RTIFRC0) Field Descriptions

Bit	Field	Value	Description
31-0	FRC0	0-FFFF FFFFh	Free running counter 0. This registers holds the current value of the free running counter 0. A read of this counter returns the current value of the counter. The counter can be preset by writing (in privileged mode only) to this register. The counter increments then from this written value upwards. **Note: If counters must be preset, they must be disabled in the RTIGCTRL register to ensure consistency between RTIUC0 and RTIFRC0.**

17.3.7 RTI Compare Up Counter 0 Register (RTICPUC0)

The compare up counter 0 register holds the value to be compared with prescale counter 0 (RTIUC0). This register is shown in Figure 17-18 and described in Table 17-8.

Figure 17-18. RTI Compare Up Counter 0 Register (RTICPUC0) [offset = 18h]

31	16
CPUC0	
R/WP-0	

15	0
CPUC0	
R/WP-0	

LEGEND: R/W = Read/Write; WP = Write in privileged mode only; -*n* = value after reset

Table 17-8. RTI Compare Up Counter 0 Register (RTICPUC0) Field Descriptions

Bit	Field	Value	Description

31-0	CPUC0	0-FFFF FFFFh	Compare up counter 0. This register holds the value that is compared with the up counter 0. When the compare shows a match, the free running counter 0 (RTIFRC0) is incremented. RTIUC0 is set to 0 when the counter value matches the RTICPUC0 value. The value set in this register prescales the RTI clock.
			If CPUC0 = 0, then
			f_{FRC0} = RTICLK/(2^{32}+1) (Setting CPUC0 equal to 0 is not recommended. Doing so will hold the Up Counter at 0 for 2 RTICLK cycles after it overflows from FFFF FFFFh to 0.)
			If CPUC0 ≠ 0, then
			f_{FRC0} = RTICLK/(RTICPUC0+1)
			A read of this register returns the current compare value.
			A write to this register:
			• If TBEXT = 0, the compare value is updated.
			• If TBEXT = 1, the compare value is unchanged.

Table 17-28. RTI Interrupt Flag Register (RTIINTFLAG) Field Descriptions

Bit	Field	Value	Description
31-19	Reserved	0	Reads return 0. Writes have no effect.
18	OVL1INT		Free running counter 1 overflow interrupt flag. This bit determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
17	OVL0INT		Free running counter 0 overflow interrupt flag. This bit determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
16	TBINT		Timebase interrupt flag. This flag is set when the TBEXT bit is cleared by detection of a missing external clock edge. It will not be set by clearing TBEXT by software. It determines if an interrupt is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
15-4	Reserved	0	Reads return 0. Writes have no effect.
3	INT3		Interrupt flag 3. These bits determine if an interrupt due to a Compare 3 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
2	INT2		Interrupt flag 2. These bits determine if an interrupt due to a Compare 2 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.
1	INT1		Interrupt flag 1. These bits determine if an interrupt due to a Compare 1 match is pending.
		0	*Read:* No interrupt is pending. *Write:* Bit is unchanged.
		1	*Read:* Interrupt is pending. *Write:* Bit is cleared to 0.

17.3.26 RTI Clear Interrupt Enable Register (RTICLEARINTENA)

This register prevents the necessity of a read-modify-write operation if a particular interrupt should be disabled. This register is shown in Figure 17-37 and described in Table 17-27.

Figure 17-37. RTI Clear Interrupt Control Register (RTICLEARINTENA) [offset = 84h]

31	24
Reserved	
R-0	

23	19	18	17	16
Reserved		CLEAROVL1INT	CLEAROVL0INT	CLEARTBINT
R-0		R/WP-0	R/WP-0	R/WP-0

15	12	11	10	9	8
Reserved		CLEARDMA3	CLEARDMA2	CLEARDMA1	CLEARDMA0
R-0		R/WP-0	R/WP-0	R/WP-0	R/WP-0

7	4	3	2	1	0
Reserved		CLEARINT3	CLEARINT2	CLEARINT1	CLEARINT0
R-0		R/WP-0	R/WP-0	R/WP-0	R/WP-0

LEGEND: R/W = Read/Write; R = Read only; WP = Write in privileged mode only; *-n* = value after reset

Table 17-27. RTI Clear Interrupt Control Register (RTICLEARINTENA) Field Descriptions

Bit	Field	Value	Description
31-19	Reserved	0	Reads return 0. Writes have no effect.
18	CLEAROVL1INT		Clear free running counter 1 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
17	CLEAROVL0INT		Clear free running counter 0 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
16	CLEARTBINT		Clear timebase interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* Interrupt is enabled. *Write:* Interrupt is disabled.
15-12	Reserved	0	Reads return 0. Writes have no effect.
11	CLEARDMA3		Clear compare DMA request 3.
		0	*Read:* DMA request is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read:* DMA request is enabled.

23.4.18 NHET Direction Register (HETDIR)

N2HET1: offset = FFF7 B84Ch; **N2HET2:** offset = FFF7 B94Ch

Figure 23-73. N2HET Direction Register (HETDIR)

31	16
HETDIR	
R/W-0	

15	0
HETDIR	
R/W-0	

LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 23-34. N2HET Direction Register (HETDIR) Field Descriptions

Bit	Field	Value	Description
31-0	HETDIR[n]		Data direction of NHET pins
		0	Pin HET[n] is an input (and its output buffer is tristated).
		1	Pin HET[n] is an output.

NOTE: Table 23-9 shows how the register bits of DIR, PULDIS and PULSEL are affecting the N2HET pins.

17.3 RTI Control Registers

Table 17-1 provides a summary of the registers. The registers support 8-bit, 16-bit, and 32-bit writes. The offset is relative to the associated peripheral select. See the following sections for detailed descriptions of the registers. The base address for the control registers is FFFF FC00h. The address locations not listed are reserved.

Table 17-1. RTI Registers

Offset	Acronym	Register Description	Section
00h	RTIGCTRL	RTI Global Control Register	Section 17.3.1
04h	RTITBCTRL	RTI Timebase Control Register	Section 17.3.2
08h	RTICAPCTRL	RTI Capture Control Register	Section 17.3.3
0Ch	RTICOMPCTRL	RTI Compare Control Register	Section 17.3.4
10h	RTIFRC0	RTI Free Running Counter 0 Register	Section 17.3.5
14h	RTIUC0	RTI Up Counter 0 Register	Section 17.3.6
18h	RTICPUC0	RTI Compare Up Counter 0 Register	Section 17.3.7
20h	RTICAFRC0	RTI Capture Free Running Counter 0 Register	Section 17.3.8
24h	RTICAUC0	RTI Capture Up Counter 0 Register	Section 17.3.9
30h	RTIFRC1	RTI Free Running Counter 1 Register	Section 17.3.10
34h	RTIUC1	RTI Up Counter 1 Register	Section 17.3.11
38h	RTICPUC1	RTI Compare Up Counter 1 Register	Section 17.3.12
40h	RTICAFRC1	RTI Capture Free Running Counter 1 Register	Section 17.3.13
44h	RTICAUC1	RTI Capture Up Counter 1 Register	Section 17.3.14
50h	RTICOMP0	RTI Compare 0 Register	Section 17.3.15
54h	RTIUDCP0	RTI Update Compare 0 Register	Section 17.3.16
58h	RTICOMP1	RTI Compare 1 Register	Section 17.3.17
5Ch	RTIUDCP1	RTI Update Compare 1 Register	Section 17.3.18
60h	RTICOMP2	RTI Compare 2 Register	Section 17.3.19
64h	RTIUDCP2	RTI Update Compare 2 Register	Section 17.3.20
68h	RTICOMP3	RTI Compare 3 Register	Section 17.3.21
6Ch	RTIUDCP3	RTI Update Compare 3 Register	Section 17.3.22
70h	RTITBLCOMP	RTI Timebase Low Compare Register	Section 17.3.23
74h	RTITBHCOMP	RTI Timebase High Compare Register	Section 17.3.24
80h	RTISETINTENA	RTI Set Interrupt Enable Register	Section 17.3.25
84h	RTICLEARINTENA	RTI Clear Interrupt Enable Register	Section 17.3.26
88h	RTIINTFLAG	RTI Interrupt Flag Register	Section 17.3.27
90h	RTIDWDCTRL	Digital Watchdog Control Register	Section 17.3.28
94h	RTIDWDPRLD	Digital Watchdog Preload Register	Section 17.3.29
98h	RTIWDSTATUS	Watchdog Status Register	Section 17.3.30
9Ch	RTIWDKEY	RTI Watchdog Key Register	Section 17.3.31
A0h	RTIDWDCNTR	RTI Digital Watchdog Down Counter Register	Section 17.3.32
A4h	RTIWWDRXNCTRL	Digital Windowed Watchdog Reaction Control Register	Section 17.3.33

34h	RTIUC1	RTI Up Counter 1 Register	Section 17.3.11
38h	RTICPUC1	RTI Compare Up Counter 1 Register	Section 17.3.12
40h	RTICAFRC1	RTI Capture Free Running Counter 1 Register	Section 17.3.13
44h	RTICAUC1	RTI Capture Up Counter 1 Register	Section 17.3.14
50h	RTICOMP0	RTI Compare 0 Register	Section 17.3.15
54h	RTIUDCP0	RTI Update Compare 0 Register	Section 17.3.16
58h	RTICOMP1	RTI Compare 1 Register	Section 17.3.17
5Ch	RTIUDCP1	RTI Update Compare 1 Register	Section 17.3.18
60h	RTICOMP2	RTI Compare 2 Register	Section 17.3.19
64h	RTIUDCP2	RTI Update Compare 2 Register	Section 17.3.20
68h	RTICOMP3	RTI Compare 3 Register	Section 17.3.21
6Ch	RTIUDCP3	RTI Update Compare 3 Register	Section 17.3.22
70h	RTITBLCOMP	RTI Timebase Low Compare Register	Section 17.3.23
74h	RTITBHCOMP	RTI Timebase High Compare Register	Section 17.3.24
80h	RTISETINTENA	RTI Set Interrupt Enable Register	Section 17.3.25
84h	RTICLEARINTENA	RTI Clear Interrupt Enable Register	Section 17.3.26
88h	RTIINTFLAG	RTI Interrupt Flag Register	Section 17.3.27
90h	RTIDWDCTRL	Digital Watchdog Control Register	Section 17.3.28
94h	RTIDWDPRLD	Digital Watchdog Preload Register	Section 17.3.29
98h	RTIWDSTATUS	Watchdog Status Register	Section 17.3.30
9Ch	RTIWDKEY	RTI Watchdog Key Register	Section 17.3.31
A0h	RTIDWDCNTR	RTI Digital Watchdog Down Counter Register	Section 17.3.32
A4h	RTIWWDRXNCTRL	Digital Windowed Watchdog Reaction Control Register	Section 17.3.33
A8h	RTIWWDSIZECTRL	Digital Windowed Watchdog Window Size Control Register	Section 17.3.34
ACh	RTIINTCLRENABLE	RTI Compare Interrupt Clear Enable Register	Section 17.3.35
B0h	RTICOMP0CLR	RTI Compare 0 Clear Register	Section 17.3.36
B4h	RTICOMP1CLR	RTI Compare 1 Clear Register	Section 17.3.37
B8h	RTICOMP2CLR	RTI Compare 2 Clear Register	Section 17.3.38
BCh	RTICOMP3CLR	RTI Compare 3 Clear Register	Section 17.3.39

17.3.25 RTI Set Interrupt Enable Register (RTISETINTENA)

This register prevents the necessity of a read-modify-write operation if a particular interrupt should be enabled. This register is shown in Figure 17-36 and described in Table 17-26.

Figure 17-36. RTI Set Interrupt Control Register (RTISETINTENA) [offset = 80h]

31	24
Reserved	
R-0	

23	19	18	17	16
Reserved		SETOVL1INT	SETOVL0INT	SETTBINT
R-0		R/WP-0	R/WP-0	R/WP-0

15	12	11	10	9	8
Reserved		SETDMA3	SETDMA2	SETDMA1	SETDMA0
R-0		R/WP-0	R/WP-0	R/WP-0	R/WP-0

7	4	3	2	1	0
Reserved		SETINT3	SETINT2	SETINT1	SETINT0
R-0		R/WP-0	R/WP-0	R/WP-0	R/WP-0

LEGEND: R/W = Read/Write; R = Read only; WP = Write in privileged mode only; -*n* = value after reset

Table 17-26. RTI Set Interrupt Control Register (RTISETINTENA) Field Descriptions

Bit	Field	Value	Description
31-19	Reserved	0	Reads return 0. Writes have no effect.
18	SETOVL1INT		Set free running counter 1 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read or Write:* Interrupt is enabled.
17	SETOVL0INT		Set free running counter 0 overflow interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read or Write:* Interrupt is enabled.
16	SETTBINT		Set timebase interrupt.
		0	*Read:* Interrupt is disabled. *Write:* Corresponding bit is unchanged.
		1	*Read or Write:* Interrupt is enabled.
15-12	Reserved	0	Reads return 0. Writes have no effect.
11	SETDMA3		Set compare DMA request 3.
		0	*Read:* DMA request is disabled. *Write:* DMA request is unchanged.
		1	*Read or Write:* DMA request is enabled.
10	SETDMA2		Set compare DMA request 2.
		0	*Read:* DMA request is disabled. *Write:* DMA request is unchanged.

```
512 ; Designid : CORE_Designid_022
513 ; Requirements: HL_CONQ_CORE_SR8
514
515         .def _enable_IRQ_interrupt_
516         .asmfunc
517
518 _enable_IRQ_interrupt_
519
520         cpsie i
521         bx    lr
522
523         .endasmfunc
```

```
void rtiStartCounter(rtiBASE_t *rtiREG, uint32 counter)
{
/* USER CODE BEGIN (4) */
/* USER CODE END */

    rtiREG->GCTRL |= ((uint32)1U << (counter & 3U));
```