



MiDAS Family

Application Note #009

(AN009-V1.1)

3A

[MiDAS1.0/1.1/2.0] How to use Watchdog Timer

WDT Restart / Interrupt

V1.1

August 2005

- ◆ CoreRiver Semiconductor reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice.
- ◆ Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.
- ◆ The CoreRiver Semiconductor products listed in this document are intended for usage in general electronics applications. These CoreRiver Semiconductor products are neither intended nor warranted for usage in equipment that requires extraordinarily high quality and/or reliability or a malfunction or failure of which may cause loss of human life or bodily injury.

Contents

1. Flow Chart : WDT (Watchdog Timer)

2. WDT Scheme of MiDAS Families

3. Example

1) MiDAS1.0 / MiDAS2.0 Family

- WDT Reset

- WDT Reset & Interrupt

2) MiDAS1.1

- WDT Reset

- WDT Reset & Interrupt

4. Appendix

1) MiDAS1.0 : WDT

2) MiDAS1.1 : WDT

3) MiDAS2.0 : WDT

1. Flow Chart : WDT (Watchdog Timer)

WDT Interrupt Setting

[WDT Interrupt]

- ◆ Restart WDT counter using **RWT** flag. (WDT counter is continuously run after power-on.)
- ◆ Clear WDT interrupt flag (WDIF) and WDT reset flag (**WTRF**).
- ◆ And Select the Time-out using **WD[1:0]**.
- ◆ **Enable WDT Interrupt flag (EWDI)**.

[WDT Reset]

- ◆ Restart WDT counter using **RWT** flag. (WDT counter is continuously run after power-on.)
- ◆ Clear WDT reset flag (**WTRF**).
- ◆ And Select the Time-out using **WD[1:0]**.
- ◆ **Enable WDT reset enable flag (EWT)**.



User Program

[WDT Counter Restart]

- ◆ Restart the WDT counter using **RWT** flag within WDT time-out.
- ◆ If malfunction is occurred. → to right.

MALFUNCTION



Check IAP Return Value

[WDT Interrupt]

- ◆ Execute the WDT interrupt function.
- ◆ If the WDT reset flag (**EWT**) is enabled, MCU is restarted after delay clocks. (The contents of MCU RAM are not change.)

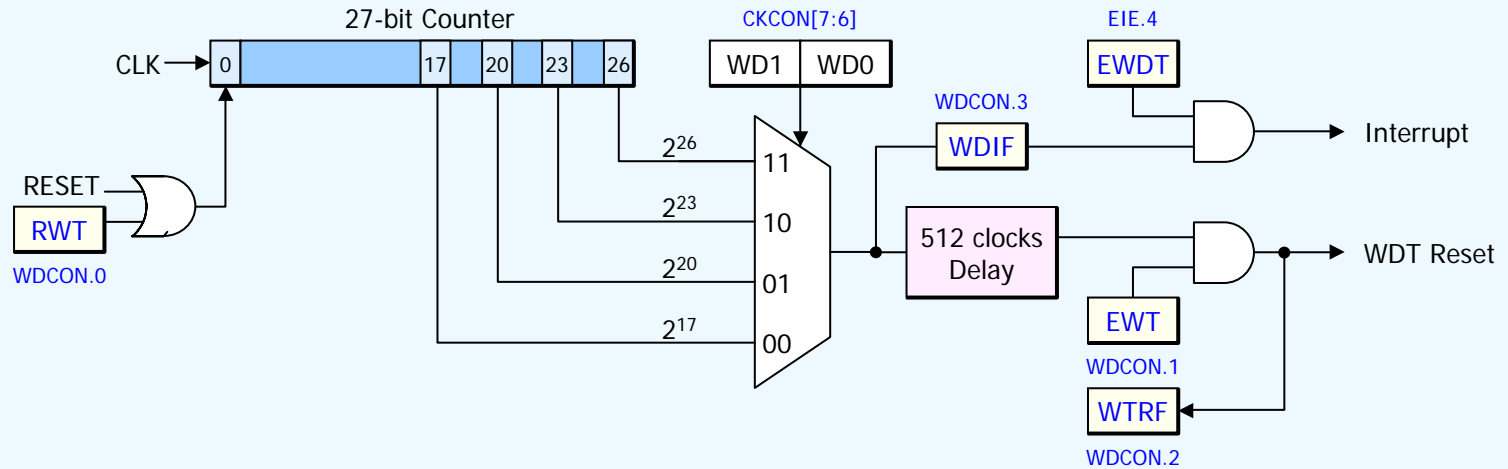
[WDT Reset]

- ◆ MCU is restarted after delay clocks. (The contents of MCU RAM are not change.)

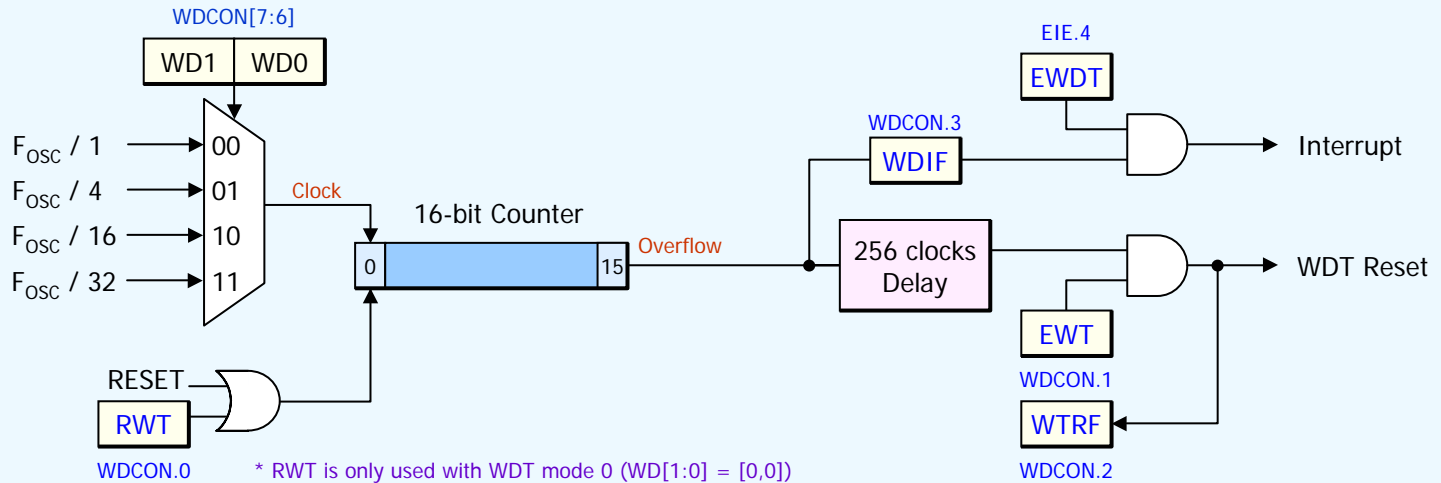
- ◆ User can use the WDT interrupt or WDT reset, or both.
- ◆ If user wants to use WDT interrupt → set the **EWDI** flag (EIE.4).
- ◆ If user wants to use WDT reset → set the **EWT** flag (WDCON.1).
- ◆ Interrupt Time-out / Reset Time-out
 - ✓ Time-out duty is controlled with **WD[1:0]**
 - ✓ **WD1** flag : MiDAS1.0/2.0 (CKCON.7), MiDAS1.1 (WDCON.7)
 - ✓ **WD0** flag : MiDAS1.0/2.0 (CKCON.6), MiDAS1.1 (WDCON.6)
- ◆ The WDT counter can be restarted with **RWT** (WDCON.6)
 - ✓ MiDAS1.0/2.0 : Set the **RWT** flag.
 - ✓ MiDAS1.1 : 1) Set the WDT time-out mode to mode 0. (Refer to Examples).
2) Set the **RWT** flag.
3) Set the WDT time-out mode 0 to the wanted mode.

2. WDT Scheme of MiDAS1.0 Families

MIDAS1.0 / MiDAS2.0



MIDAS1.1



* RWT is only used with WDT mode 0 (WD[1:0] = [0,0]) for MiDAS1.1 Family

3. Example : 1) MiDAS1.0 / MiDAS2.0 Family

◆ WDT Reset

✓ SFR Setting

```
//-----  
// SFR Setting (Initialization)  
  
// Restart WDT counter (27 bits)  
RWT = 1;  
  
// Clear WDT interrupt/reset flag  
WDIF = 0;  
WTRF = 0;  
  
// Select the WDT reset time-out mode  
// ex) Fosc = 12MHz  
// mode 0 -> 10.97 msec  
// mode 1 -> 87.42 msec  
// mode 2 -> 699.09 msec  
// mode 3 -> 5,592.45 msec  
CKCON |= WD1_; // mode 2  
// (Apprx. 0.7sec)  
//CKCON |= WD0_;  
  
// Enable WDT reset enable flag  
EWT = 1;  
  
// All interrupts enable  
EA = 1;
```

✓ WDT Counter Restart

```
//-----  
// WDT Counter Restart  
//  
// Restart WDT counter (27bits)  
RWT = 1;
```

3. Example : 1) MiDAS1.0 / MiDAS2.0 Family

◆ WDT Interrupt & Reset

✓ SFR Setting

```
//-----  
// SFR Setting (Initialization)  
  
// Restart WDT counter (27 bits)  
RWT = 1;  
  
// Clear WDT interrupt/reset flag  
WDIF = 0;  
WTRF = 0;  
  
// Select the WDT reset time-out mode  
// ex) Fosc = 12MHz  
// mode 0 -> 10.97 msec  
// mode 1 -> 87.42 msec  
// mode 2 -> 699.09 msec  
// mode 3 -> 5,592.45 msec  
CKCON |= WD1_; // mode 2  
// (Apprx. 0.7sec)  
  
//CKCON |= WD0_;  
  
// Enable WDT reset enable flag  
EWT = 1;  
  
// Enable WDT interrupt enable flag  
EWDI = 1;  
  
// All interrupts enable  
EA = 1;
```

✓ WDT Counter Restart

```
//-----  
// WDT Counter Restart  
//  
// Restart WDT counter (27bits)  
RWT = 1;
```

✓ WDT Interrupt Function

```
//-----  
// WDT Interrupt Function  
void wdt_int(void) interrupt WDT_VECTOR {  
  
// ** INSERT USER CODE **  
// (Note : After MCU is restarted,  
// the contents of MCU RAM is not  
// change.  
// And the data for malfunction can  
// be stored to MCU RAM. Or etc...  
  
WDIF = 0;  
  
}
```

3. Example : 2) MiDAS1.1 Family

◆ WDT Reset

✓ SFR Setting

```
//-----  
// SFR Setting (Initialization)  
  
// Restart WDT counter (16 bits)  
// * User must set the WDT time-out  
// mode to mode 0 before restart  
// the WDT counter.  
WD1 = 0; // Mode 0  
WDO = 0;  
  
RWT = 1;  
  
// Clear WDT interrupt/reset flag  
WDIF = 0;  
WTRF = 0;  
  
// Select the WDT reset time-out mode  
// ex) Fosc = 4MHz  
// mode 0 -> 15.51 msec  
// mode 1 -> 66.05 msec  
// mode 2 -> 264.19 msec  
// mode 3 -> 528.38 msec  
WD1 = 1; // mode 3 (Apprx. 0.5sec)  
WDO = 1;  
  
// Enable WDT reset enable flag  
EWT = 1;  
  
// All interrupts enable  
EA = 1;
```

✓ WDT Counter Restart

```
//-----  
// WDT Counter Restart  
//  
// Restart WDT counter (16 bits)  
// * User must set the WDT time-out  
// mode to mode 0 before restart  
// the WDT counter.  
WD1 = 0; // Mode 0  
WDO = 0;  
  
RWT = 1;  
  
WD1 = 1; // Mode 3  
WDO = 1;
```

3. Example : 2) MiDAS1.1 Family

◆ WDT Interrupt & Reset

✓ SFR Setting

```
//-----  
// SFR Setting (Initialization)  
  
// Restart WDT counter (16 bits)  
// * User must set the WDT time-out  
// mode to mode 0 before restart  
// the WDT counter.  
WD1 = 0; // Mode 0  
WDO = 0;  
  
RWT = 1;  
  
// Clear WDT interrupt/reset flag  
WDIF = 0;  
WTRF = 0;  
  
// Select the WDT reset time-out mode  
// ex) Fosc = 4MHz  
// mode 3 -> 528.38 msec  
WD1 = 1; // mode 3 (Approx. 0.5sec)  
WDO = 1;  
  
// Enable WDT reset enable flag  
EWT = 1;  
  
// Enable WDT interrupt enable flag  
EWDI = 1;  
  
// All interrupts enable  
EA = 1;
```

✓ WDT Counter Restart

```
//-----  
// WDT Counter Restart  
//  
// Restart WDT counter (16 bits)  
// * User must set the WDT time-out  
// mode to mode 0 before restart  
// the WDT counter.  
WD1 = 0; // Mode 0  
WDO = 0;  
  
RWT = 1;  
  
WD1 = 1; // Mode 3  
WDO = 1;
```

✓ WDT Interrupt Function

```
//-----  
// WDT Interrupt Function  
void wdt_int(void) interrupt WDT_VECTOR {  
  
// ** INSERT USER CODE **  
// (Note : After MCU is restarted,  
// the contents of MCU RAM is not  
// change.  
// And the data for malfunction can  
// be stored to MCU RAM. Or etc...  
  
WDIF = 0;  
}
```


Appendix. MiDAS1.0 : WDT (Watchdog Timer)

- ◆ Detect the malfunction of program due to external noise or other causes
- ◆ Return the operation to the normal condition using WDT interrupt

◆ Watchdog Time-out Values

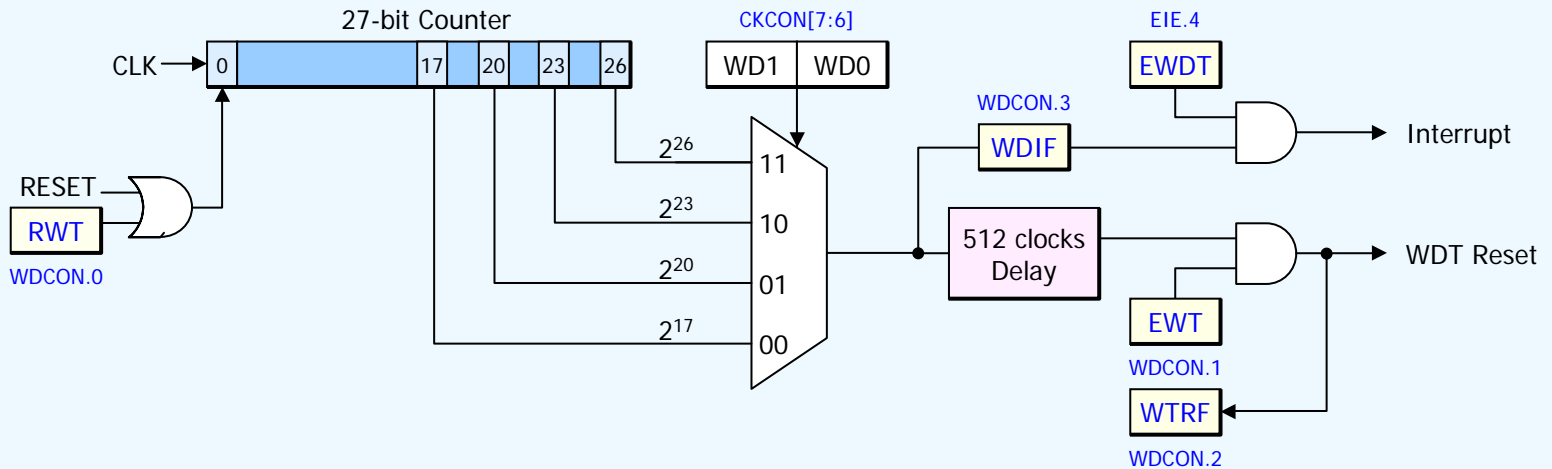
✓ Default : WD[1:0] = [0,0]

WD1	WD0	Interrupt Time-out (@25MHz)	Reset Time-out (@25MHz)
0	0	2^{17} clocks 5.24 ms	$2^{17} + 512$ clocks 5.26 ms
0	1	2^{20} clocks 41.94 ms	$2^{20} + 512$ clocks 41.96 ms
1	0	2^{23} clocks 335.54 ms	$2^{23} + 512$ clocks 335.56 ms
1	1	2^{26} clocks 2,684.35 ms	$2^{26} + 512$ clocks 2,684.38 ms

✓ WDCON (D8h) : Watchdog & Power Status Register

-	POR	EPFI	PFI	WDIF	WTRF	EWT	RWT
	R/W(1)	R/W(0)	R/W(1)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

- POR : Power-on Reset Flag
- EPFI : Enable Power-fail Interrupt
- PFI : Power-Fail interrupt Flag
- WDIF : Watchdog Timer Interrupt Flag
- WTRF : Watchdog Timer Reset Flag. Only cleared by S/W.
- EWT : Watchdog Timer Reset Enable
- RWT : Restart Watchdog Timer



Appendix. MiDAS1.1 : WDT (Watchdog Timer)

- ◆ Detect the malfunction of program due to external noise or other causes.
- ◆ Return the operation to the normal condition using WDT interrupt.
- ◆ If enabled, WDT interrupt or WDT reset makes MCU wake up from Stop Mode 2.
- ◆ Watchdog Time-out Values

✓ Default : WD[1:0] = [0,0]

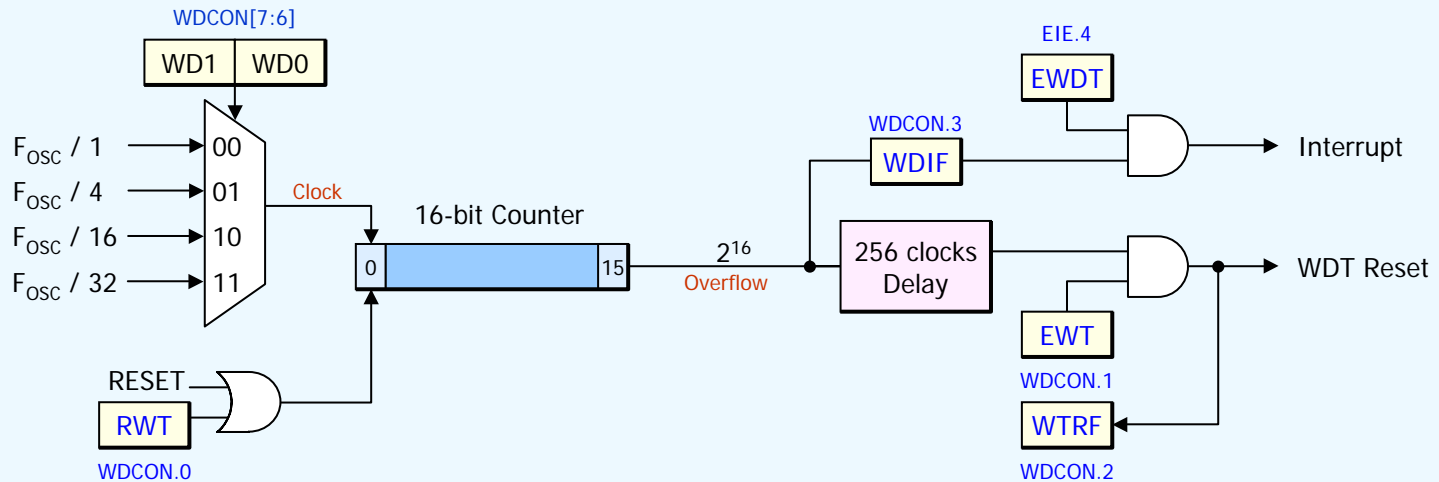
WD1	WD0	Interrupt Time-out (@4MHz)		Reset Time-out (@4MHz)	
0	0	1x2 ¹⁶ clocks	16.38 ms	1x2 ¹⁶ + 256 clocks	16.45 ms
0	1	4x2 ¹⁶ clocks	65.54 ms	4x2 ¹⁶ + 256 clocks	65.60 ms
1	0	16x2 ¹⁶ clocks	262.14 ms	16x2 ¹⁶ + 256 clocks	262.21 ms
1	1	32x2 ¹⁶ clocks	524.29 ms	32x2 ¹⁶ + 256 clocks	524.35 ms

✓ **WDCON** (D8h) : Watchdog Timer Control Register

WD1	WD0	-	-	WDIF	WTRF	EWT	RWT
-----	-----	---	---	------	------	-----	-----

R/W(1) R/W(1) R/W(0) R/W(1) R/W(0) R/W(0) R/W(0) R/W(0)

- WD[1:0] : WDT Clock Divide(1/4/8/32)
- WDIF : Watchdog Timer Interrupt Flag
- WTRF : Watchdog Timer Reset Flag. Only cleared by S/W.
- EWT : Watchdog Timer Reset Enable
- RWT : Restart Watchdog Timer



* RWT is only used with WDT mode 0 (WD[1:0] = [0,0]) for MiDAS1.1 Family (Refer to Application Note #009 (AN009))

Appendix. MiDAS2.0 : WDT (Watchdog Timer)

- ◆ Detect the malfunction of program due to external noise or other causes.
- ◆ Return the operation to the normal condition using WDT interrupt
- ◆ If enabled, WDT interrupt or WDT reset makes MCU wake up from Stop Mode 2.
- ◆ Watchdog Time-out Values

✓ Default : $WD[1:0] = [1,1]$

WD1	WD0	Interrupt Time-out (@25MHz)		Reset Time-out (@25MHz)	
0	0	2^{17} clocks	5.24 ms	$2^{17} + 512$ clocks	5.26 ms
0	1	2^{20} clocks	41.94 ms	$2^{20} + 512$ clocks	41.96 ms
1	0	2^{23} clocks	335.54 ms	$2^{23} + 512$ clocks	335.56 ms
1	1	2^{26} clocks	2,684.35 ms	$2^{26} + 512$ clocks	2,684.38 ms

✓ **WDCON** (D8h) : Watchdog Timer & Power Status Register

-	POR	EPFI	PFI	WDIF	WTRF	EWT	RWT
	R/W(1)	R/W(0)	R/W(1)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

- POR : Power-on Reset Flag
- EPFI : Enable Power-fail Interrupt
- PFI : Power-Fail interrupt Flag
- WDIF : Watchdog Timer Interrupt Flag
- WTRF : Watchdog Timer Reset Flag
- EWT : Watchdog Timer Reset Enable
- RWT : Restart Watchdog Timer

