

Winmate® API Specifications

Project Name: PPC Watchdog SDK

Revision: 2.2

April 21, 2015

Contents

1.	General Description	3
1.1	Introduction.....	3
1.2	Noteic Of User Precautions	3
2.	Driver	4
2.1	Install WMWDG Driver.....	4
3.	Programing environment.....	5
3.1	Project Setting	5
3.2	WMWDGDLL.h File Reference:	5
4.	API Definition	6
4.1	Function Procedure:.....	6
4.1.1	Function Block.....	6
4.1.2	WatchDog Function Block.....	6
4.2	Function Name:	7
4.2.1	WM_SetWatchDog(WORD nTimer,WORD nCounter);.....	7

Revision History

Revision	Date	Description
2.0	2014/03/28	1. Initial draft.
2.1	2015/04/13	1. Add Watchdog function support Fintek F81865 and F81866 superio.
2.2	2015/04/21	1. Fixed IH32/IH70/IHWW watchdog can not active by api when bios setup watchdog function disabled.

1. General Description

1.1 Introduction

The Drivers and API provide the features to control the Watchdog function.

1.2 Noteic Of User Precautions

※For motherboard:ID31/ID32 must turn off fast boot feature in bios setup.

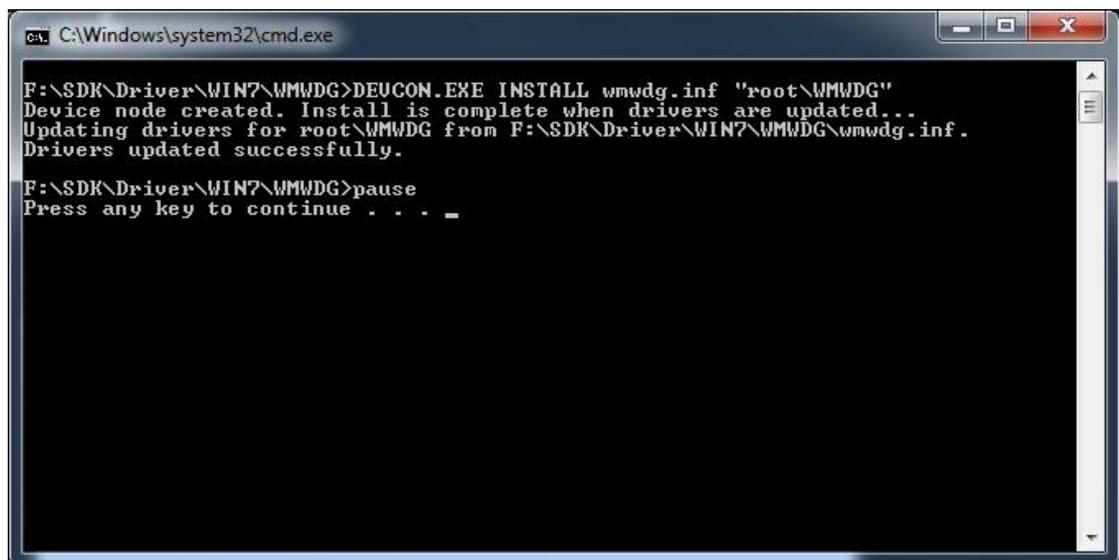
※For motherboard:IV70 must turn on watch dog timer 1~10 min in bios setup.

2.Driver

2.1 Install WMWDG Driver

The drivers are for Windows 7/8. Please double click install.bat in driver directories to start install.

When the driver is successfully installed, you can see "Drivers Updated successfully" message in the dos prompt.



```
C:\Windows\system32\cmd.exe
F:\SDK\Driver\WIN7\WMWDG>DEUCON.EXE INSTALL wwdg.inf "root\WMWDG"
Device node created. Install is complete when drivers are updated...
Updating drivers for root\WMWDG from F:\SDK\Driver\WIN7\WMWDG\wwdg.inf .
Drivers updated successfully.
F:\SDK\Driver\WIN7\WMWDG>pause
Press any key to continue . . . _
```

And "WMWDG" devices are also added in the Device Manger under "System devices"

Warning: It is important that only one WMWDG devices can appear in Device Manager.

3. Programming environment

3.1 Project Setting

To control the device power state, brightness, led flash control and obtain SMBIOS information, dynamic file (dll), library (lib) and header (h) files are provided to develop the Application.

WMWDGDLL.lib	Library
WMWDGDLL.dll	Dynamic Library
WMWDGDLL.h	Header File

1. Include “WMWDGDLL.h” in the project.
2. Add “WMWDGDLL.lib” into project Link.
3. Put “WMWDGDLL.dll” in the same path with application or into “windows” directory.
4. Check “WMWDGDLL.dll” version is 2.0.0.0

3.2 WMWDGDLL.h File Reference:

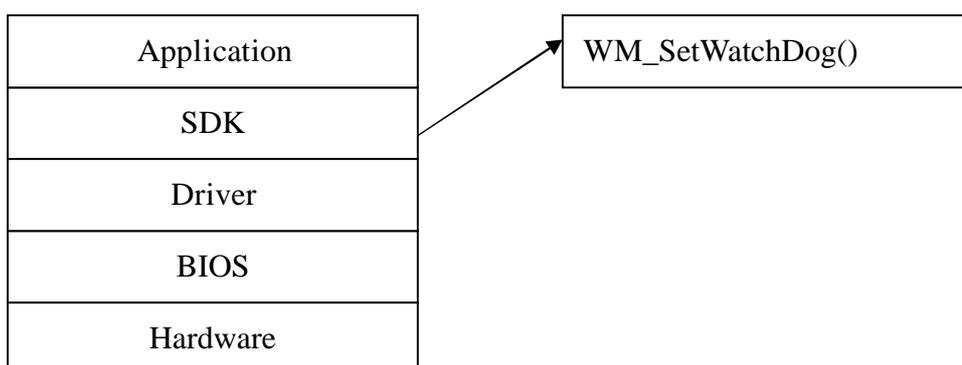
```
#ifndef WMWDGDLL_EXPORTS
#define WMWDGDLL_API extern "C" __declspec(dllexport)
#else
#define WMWDGDLL_API extern "C" __declspec(dllimport)
#endif

WMWDGDLL_API bool WM_SetWatchDog(WORD nTimer,WORD
nCounter);
```

4.API Definition

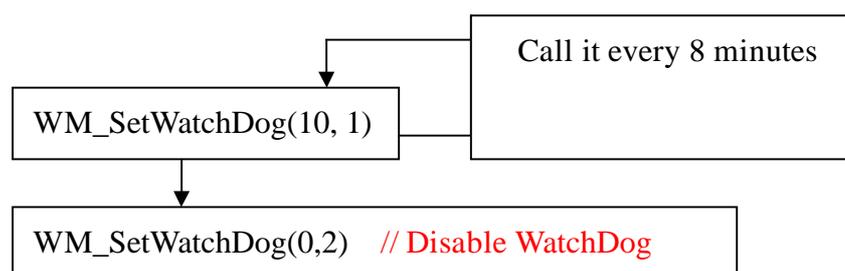
4.1 Function Procedure:

4.1.1 Function Block



4.1.2 WatchDog Function Block

In this example, it set watchdog to minute mode, and set counter to 10 minutes. To avoid watchdog reboot the system, application sets counter to 10 every 8 minutes.



4.2 Function Name:

4.2.1 WM_SetWatchDog(WORD nTimer,WORD nCounter);

The WM_SetWatchDog() all in one Function provides the interface to set the counter or disable the WatchDog.

```
WMWDGDLL_API bool WM_SetWatchDog(  
WORD nTimer,  
WORD nCounter);
```

Parameters

nTime:

Watch Dog Timer Time-out value.

Set a non-zero value causes the counter to load the value to Watch Dog Counter and start counting down.

00h: Time-out Disable.

01h: Time-out occurs after 1 second/minute

02h: Time-out occurs after 2 second/minutes

.....

FFh: Time-out occurs after 255 second/minutes

nCounter:

Select WatchDog count mode.

0: Second Mode.

1: Minute Mode.

2: Disable WatchDog.

Return Value

The function returns TRUE if it is successful set the WatchDog, and FALSE otherwise.

Requirements

Header: Declared in WMWDGDLL.h