ABOV
SEMICONDUCTOR

OCD-I / II
Debugging tool
USER GUIDE

Release V1.00

OCD-I / II debugging tool USER GUIDE

Release information

Description	Issue	Change
March 2012	A	First release

Proprietary notice

The product described in this document is subject t0o continuous developments and improvements. All particulars of the product and its use contained in this document are given by ABOV Semiconductor Co.,Ltd. in good faith.
However, all warranties implied or expressed, including but not limited to implied warranties or merchantability, or fitness for purpose, are excluded.

This document is intended only to assist the reader in the use of the product. ABOV Semiconductor Co.,Ltd. shall not be liable for any loss or damage arising from the use of any information in this document, or any error or omission such information, any incorrect use of the product.

Document confidentiality status

This document is Open Access. This document has no restriction on distribution.

Product status

The information in this document is Final (information on a developed product).

ABOV web address

http://www.abov.co.kr

Contents

OCD-I / II Debugging tool USER GUIDE

Chapter 1
Getting Started

This chapter describes:

- System requirements
- Setup package installation
- Driver installation for MS-Windows

1.1 System requirements

This section described the hardware and software system requirements.

1.1.1 Software requirements

You must be using one of the following operation systems to install and run OCD-I / II debugger.

32bit version and 64bit version are prepared already.

- MS-Windows NT
- MS-Windows 2000
- MS-Windows XP
- MS-Windows Vista
- MS-Windows7

Disk space

If you wish to carry out a full installation of the software, up to 10MB of hard disk space is required.

1.1.2 Hardware requirements

The following are the minimum recommended hardware requirements for installing and running the OCD-I / II debugger.

- Pentium PC
- USB port

Performance is based on following factors:

- Processor performance
- USB port performance

OCD-I / II debugger does not care USB version (V1.0, V1.1, V2.0 or higher version).

Anyway, V2.0 is better than V1.1.

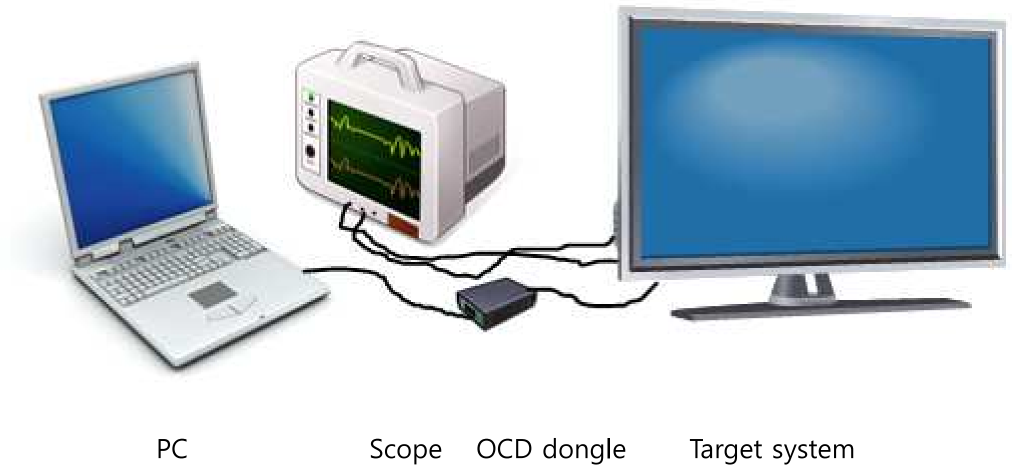
1.1.3 OCD dongle hardware

OCD-I / II debugger support OCD-I dongle hardware and OCD-II dongle hardware.
OCD means On Chip Debug.

It is very cheap solution to develop application software.
Furthermore, it is easy to use.
You do not need to set any complex configurations.
Just connect line and power ON.

Each MCU device have OCD block inside.

- OCD-I dongle hardware
  It used OCD-I interface protocol only.
  So, it can support OCD-I series devices only.
- OCD-II dongle hardware
  It can use OCD-I interface protocol and OCD-II interface protocol.
  So, it can support OCD-I series and OCD-II series devices.
  Its interface speed is higher than OCD-I dongle hardware.
  It used 2 interface line (SCLK, SDATA) and option line.



1.2 Setup package

You can download the install program from our website (http://www.abov.co.kr).

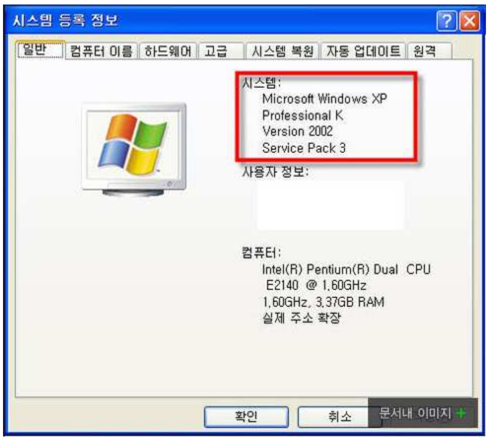
You had better to keep newest OCD-I / II software because we add new devices and newer features continuously.

We provide 32bit version and 64bit version.

If you do not know your PC OS and its version, refer followings.

- Open “Control panel”.
- Find “System” icon and select it.

Ex) MS-Windows XP (32bit)

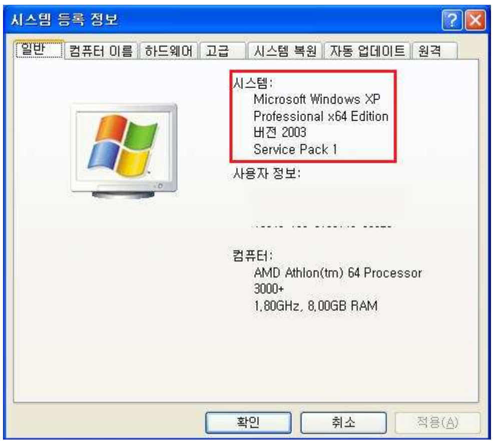


시스템:
Microsoft Windows XP
Professional K
Version 2002
Service Pack 3

사용자 정보:

컴퓨터:
Intel(R) Pentium(R) Dual CPU
E2140 @ 1.60GHz
1.60GHz, 3.37GB RAM
실제 주소 확장

Ex) MS-Windows XP (64bit)



시스템:
Microsoft Windows XP
Professional x64 Edition
버전 2003
Service Pack 1

사용자 정보:

컴퓨터:
AMD Athlon(tm) 64 Processor
3000+
1.80GHz, 8.00GB RAM

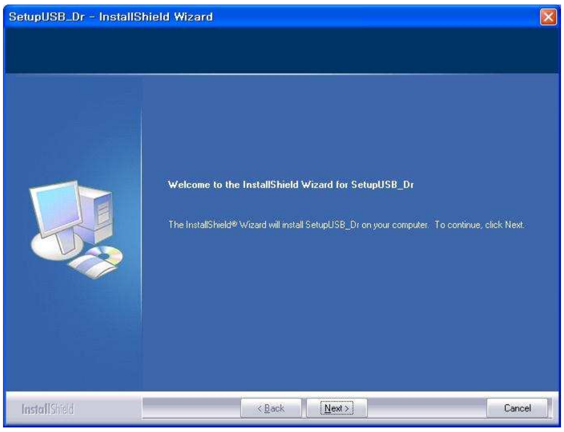
1.2.1 Software installation (32bit)

32bit version installation is easier than 64bit version.

Execute setup program.

You can see following dialog box.

Click “Next” button.

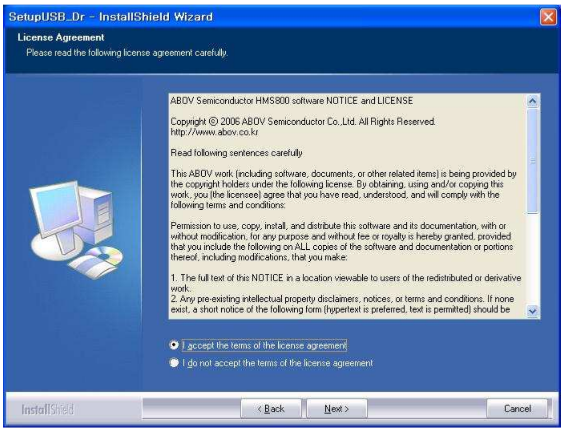


Welcome to the InstallShield Wizard for SetupUSB_Dr

The InstallShield® Wizard will install SetupUSB_Dr on your computer. To continue, click Next.

When the license agreement dialog box is appeared, select “I accept the items of the license agreement”.

Click the “Next” button.


SetupUSB_Dr - InstallShield Wizard

License Agreement

Please read the following license agreement carefully.

ABOV Semiconductor HMS800 software NOTICE and LICENSE

Copyright © 2006 ABOV Semiconductor Co.,Ltd. All Rights Reserved.
http://www.abov.co.kr

Read following sentences carefully

This ABOV work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

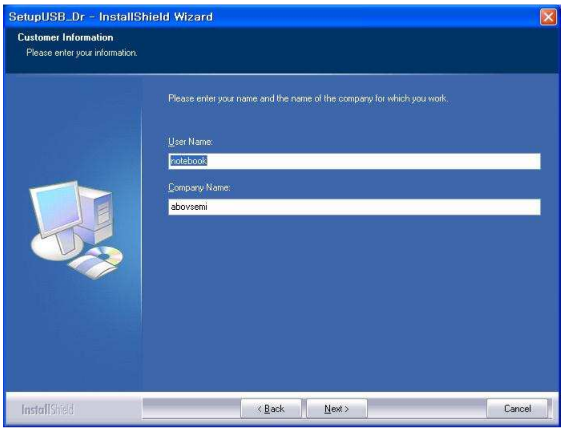
Permission to use, copy, install, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be

● I accept the terms of the license agreement
○ I do not accept the terms of the license agreement

Fill the user name and company name.

Click the “Next” button.


SetupUSB_Dr - InstallShield Wizard

Customer Information

Please enter your information.

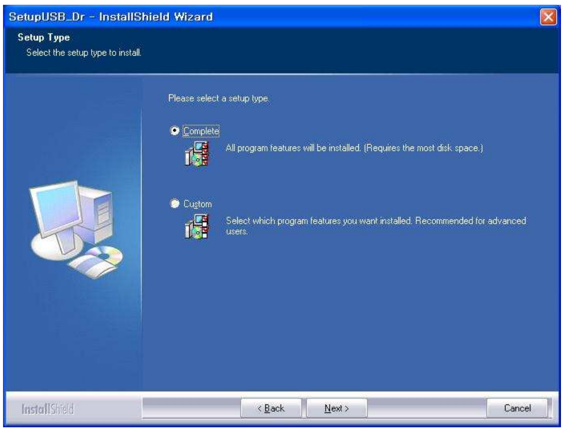
Please enter your name and the name of the company for which you work.

User Name: notebook

Company Name: abovsemi

Select “Complete”.

Click the “Next” button.


SetupUSB_Dr - InstallShield Wizard

Setup Type

Select the setup type to install.

Please select a setup type.

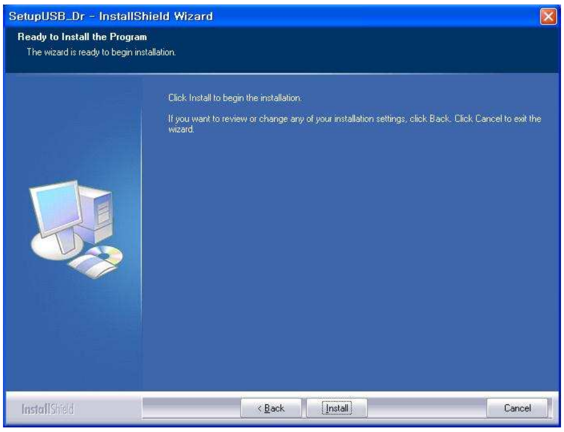
● Complete

All program features will be installed. (Requires the most disk space.)

○ Custom

Select which program features you want installed. Recommended for advanced users.

Click "Install" button.



SetupUSB_Dr - InstallShield Wizard

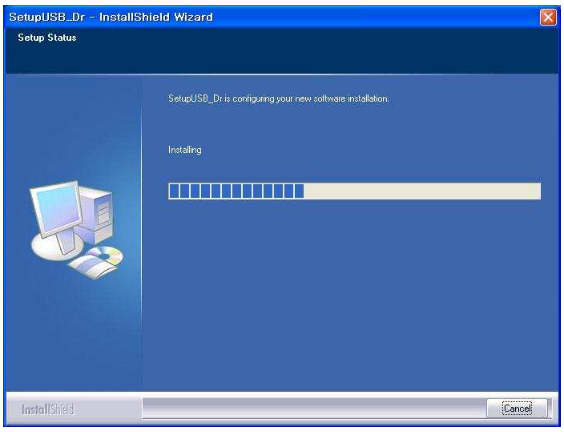
Ready to Install the Program

The wizard is ready to begin installation.

Click Install to begin the installation.

If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

Wait until it installs all of the program components.



SetupUSB_Dr - InstallShield Wizard

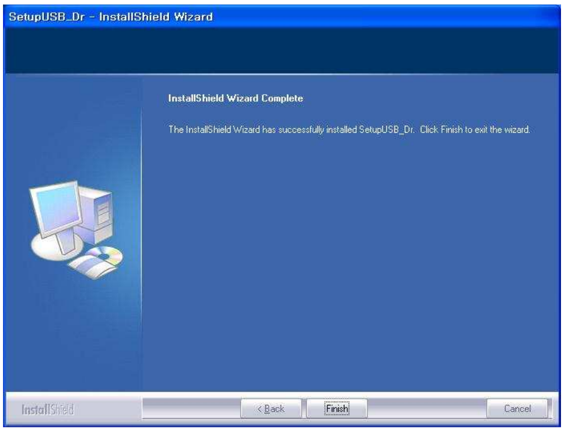
Setup Status

SetupUSB_Dr is configuring your new software installation.

Installing

Installation is completed.

Click "Finish" button.



SetupUSB_Dr - InstallShield Wizard

InstallShield Wizard Complete

The InstallShield Wizard has successfully installed SetupUSB_Dr. Click Finish to exit the wizard.

Installed folder is "C:₩Program Files₩ABOV Semiconductor₩OCD2_debugger32"

Remember this path. It will be used to install driver files.

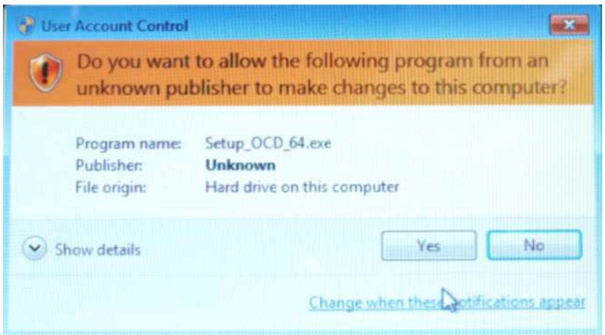
1.2.2 Software installation (64bit)

64bit version installation is bothersome than 32bit version.

Execute setup program.

You can see the warning message as below.

Click “Yes” button.

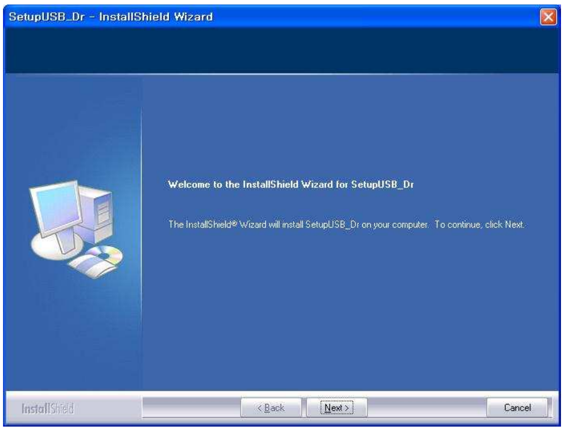


User Account Control

Do you want to allow the following program from an unknown publisher to make changes to this computer?

Program name: Setup_OCD_64.exe
Publisher: **Unknown**
File origin: Hard drive on this computer

Click “Next” button.



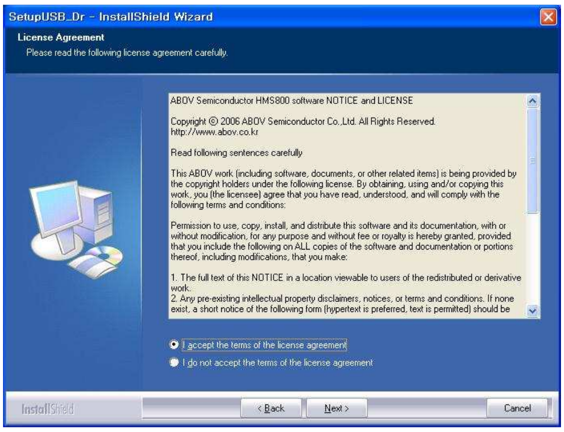
SetupUSB_Dr - InstallShield Wizard

Welcome to the InstallShield Wizard for SetupUSB_Dr

The InstallShield® Wizard will install SetupUSB_Dr on your computer. To continue, click Next.

When the license agreement dialog box is appeared, select “I accept the items of the license agreement”.

Click the “Next” button.



SetupUSB_Dr - InstallShield Wizard

License Agreement

Please read the following license agreement carefully.

ABOV Semiconductor HMS800 software NOTICE and LICENSE

Copyright © 2006 ABOV Semiconductor Co.,Ltd. All Rights Reserved.
http://www.abov.co.kr

Read following sentences carefully

This ABOV work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

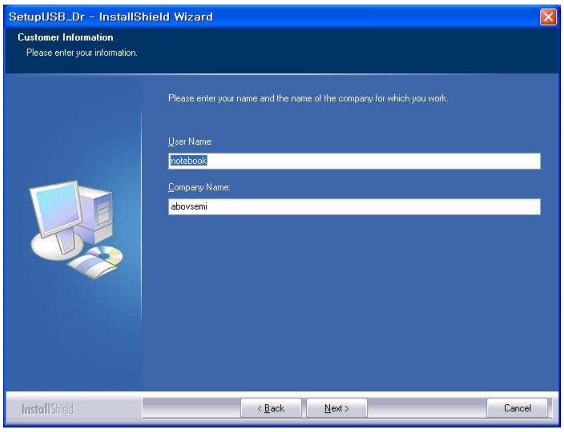
Permission to use, copy, install, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be

● I accept the terms of the license agreement
○ I do not accept the terms of the license agreement

Fill the user name and company name.

Click the "Next" button.



SetupUSB_Dr - InstallShield Wizard

Customer Information

Please enter your information.

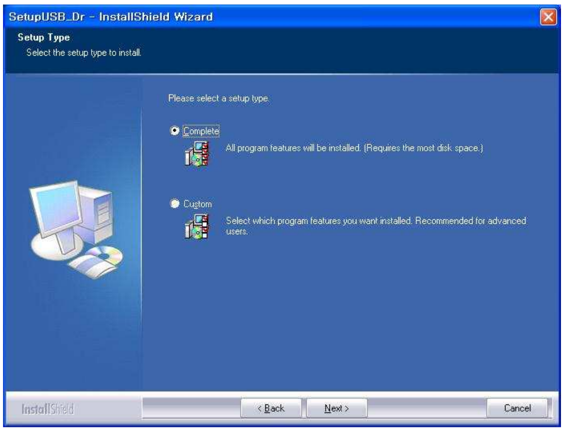
Please enter your name and the name of the company for which you work.

User Name: notebook

Company Name: abovsemi

Select "Complete".

Click the "Next" button.



SetupUSB_Dr - InstallShield Wizard

Setup Type

Select the setup type to install.

Please select a setup type.

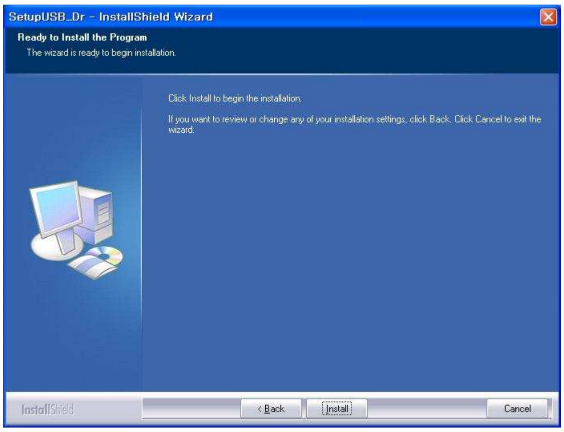
● Complete

All program features will be installed. (Requires the most disk space.)

○ Custom

Select which program features you want installed. Recommended for advanced users.

Click "Install" button.



SetupUSB_Dr - InstallShield Wizard

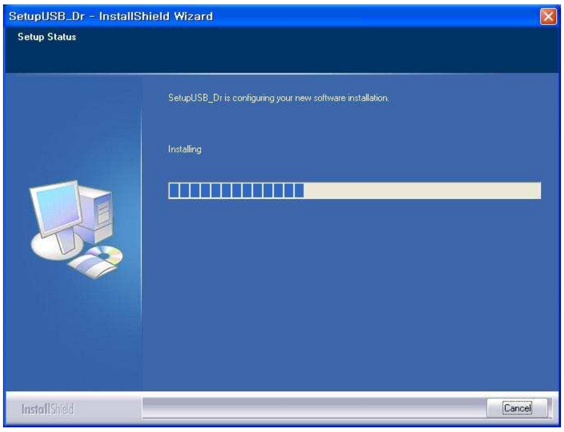
Ready to Install the Program

The wizard is ready to begin installation.

Click Install to begin the installation.

If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

Wait until it installs all of the program components.

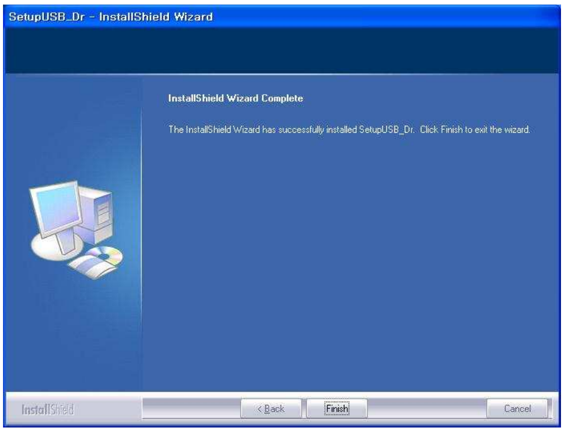


SetupUSB_Dr is configuring your new software installation.

Installing

Installation is completed.

Click “Finish” button.



InstallShield Wizard Complete

The InstallShield Wizard has successfully installed SetupUSB_Dr. Click Finish to exit the wizard.

Installed folder is “C:₩Program Files (x86)₩ABOV Semiconductor₩OCD2_debugger64”

Remember this path. It will be used to install driver files.

1.3 Driver installation for MS-Windows

OCD-I or OCD-II dongle hardware does not work until appropriate OCD driver is installed in your PC.

OCD driver files are copied during application software installation. So, you have to install application software first.

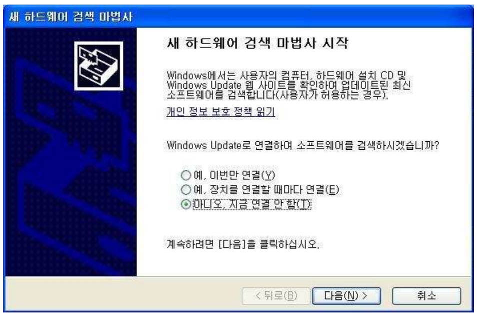
1.3.1 Driver installation (32bit)

Following dialog box pictures are using Korean (Hangeul) characters. Anyway, you can understand following examples.

Connect OCD-I or OCD-II dongle hardware to USB port of your PC.

PC detects new hardware and popup following dialog box.

Click "Next" button



새 하드웨어 검색 마법사

새 하드웨어 검색 마법사 시작

Windows에서는 사용자의 컴퓨터, 하드웨어 설치 CD 및 Windows Update 웹 사이트를 확인하여 업데이트된 최신 소프트웨어를 검색합니다(사용자가 허용하는 경우).

개인 정보 보호 정책 읽기

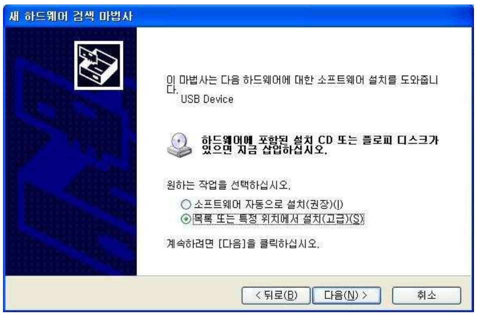
Windows Update로 연결하여 소프트웨어를 검색하시겠습니까?

○ 예, 이번만 연결(Y)
○ 예, 장치를 연결할 때마다 연결(E)
● 아니오, 지금 연결 안 함(T)

계속하려면 [다음]을 클릭하십시오.

Select lower item.

Click "Next" button



새 하드웨어 검색 마법사

이 마법사는 다음 하드웨어에 대한 소프트웨어 설치를 도와줍니다.
USB Device

하드웨어에 포함된 설치 CD 또는 플로피 디스크가 있으면 지금 삽입하십시오.

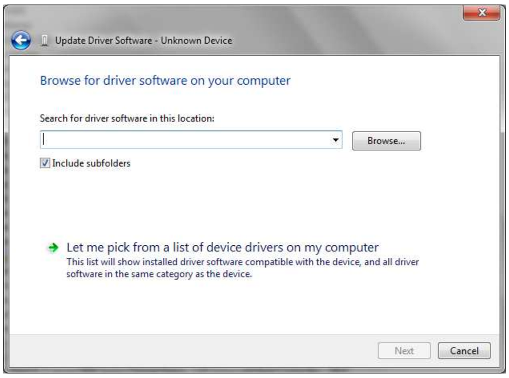
원하는 작업을 선택하십시오.

○ 소프트웨어 자동으로 설치(권장)(I)
● 목록 또는 특정 위치에서 설치(고급)(S)

계속하려면 [다음]을 클릭하십시오.

Fill INF path that you installed software folder.

For example, “C:₩Program Files₩ABOV Semiconductor₩OCD2_debugger32”.



Update Driver Software - Unknown Device

Browse for driver software on your computer

Search for driver software in this location:

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

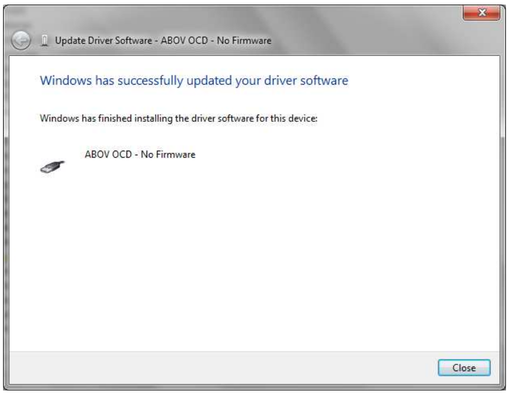
Now, MS-Windows asks to continue install or not.

Click “Continue” button and wait until installation is completed.

OCD-I dongle hardware driver installation

If you connected OCD-I dongle hardware, “ABOV OCD – No Firmware” will be installed at first time. Following picture shows it.

You have to install hardware one more time.



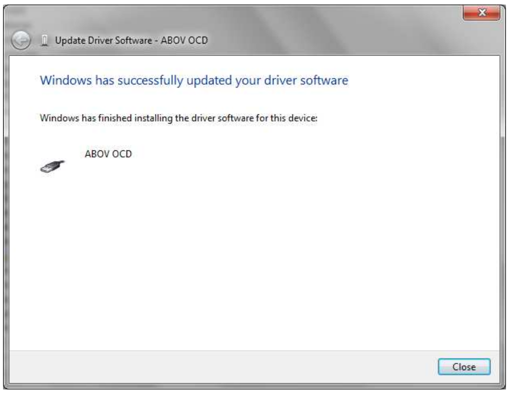
Update Driver Software - ABOV OCD - No Firmware

Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

ABOV OCD - No Firmware

ABOV OCD-I dongle hardware is installed correctly.



Update Driver Software - ABOV OCD

Windows has successfully updated your driver software

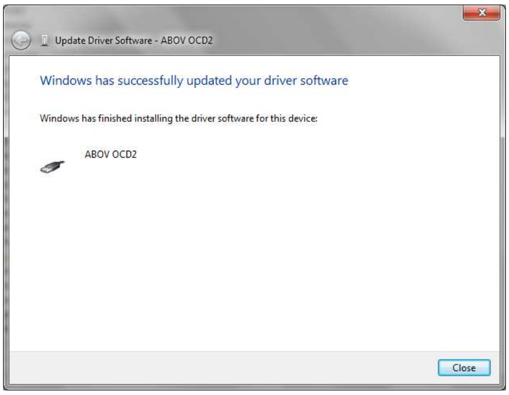
Windows has finished installing the driver software for this device:

ABOV OCD

OCD-II dongle hardware driver installation

If you connected OCD-II dongle hardware, it will be installed just one time.

ABOV OCD-II dongle hardware is installed correctly.



Update Driver Software - ABOV OCD2

Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

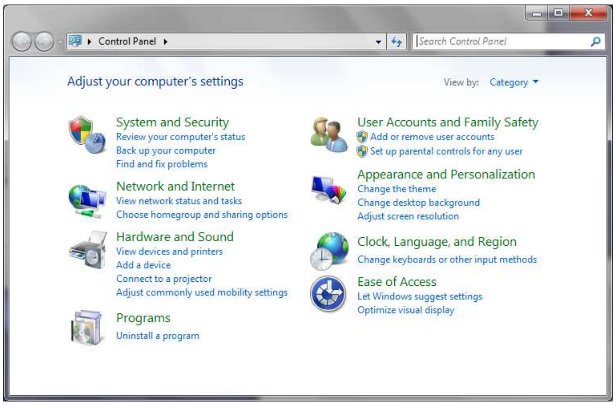
ABOV OCD2

1.3.2 Driver installation (64bit)

64bit OS of Microsoft® manage drivers more strictly than 32bit OS.
Because of, it maintains itself safely from unauthorized system drivers.
As a result, 64bit OS works very stably.
But, driver installation is not so easy.

Connect OCD-I or OCD-II dongle hardware to USB port of your PC.
You have to install driver files manually.

Click MS-Windows's "Start" button and execute "Control panel".



Control Panel ▸
Search Control Panel

Adjust your computer's settings

View by: Category ▾

System and Security
Review your computer's status
Back up your computer
Find and fix problems

Network and Internet
View network status and tasks
Choose homegroup and sharing options

Hardware and Sound
View devices and printers
Add a device
Connect to a projector
Adjust commonly used mobility settings

Programs
Uninstall a program

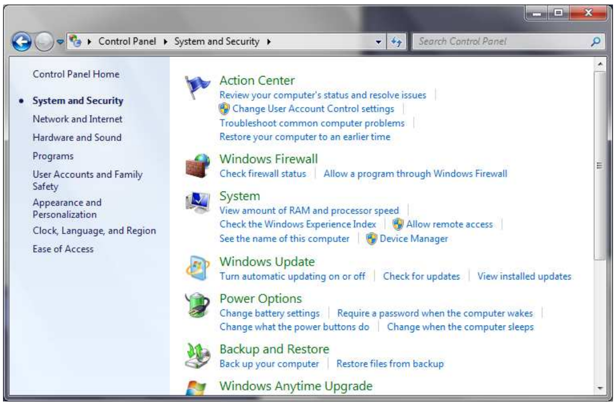
User Accounts and Family Safety
Add or remove user accounts
Set up parental controls for any user

Appearance and Personalization
Change the theme
Change desktop background
Adjust screen resolution

Clock, Language, and Region
Change keyboards or other input methods

Ease of Access
Let Windows suggest settings
Optimize visual display

Click "System and Security".



Control Panel ▸ System and Security ▸
Search Control Panel

Control Panel Home
- **System and Security**

Network and Internet
Hardware and Sound
Programs
User Accounts and Family Safety
Appearance and Personalization
Clock, Language, and Region
Ease of Access

Action Center
Review your computer's status and resolve issues | Change User Account Control settings | Troubleshoot common computer problems | Restore your computer to an earlier time

Windows Firewall
Check firewall status | Allow a program through Windows Firewall

System
View amount of RAM and processor speed | Check the Windows Experience Index | Allow remote access | See the name of this computer | Device Manager

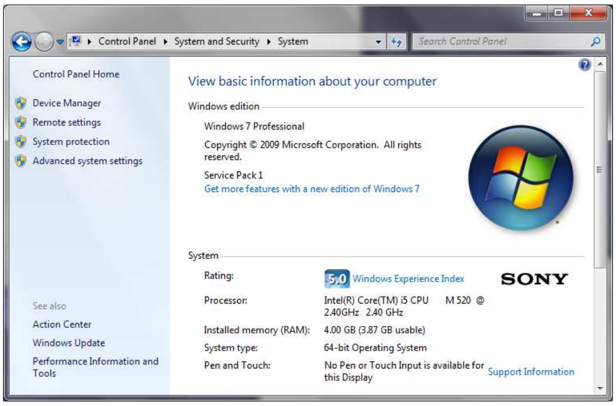
Windows Update
Turn automatic updating on or off | Check for updates | View installed updates

Power Options
Change battery settings | Require a password when the computer wakes | Change what the power buttons do | Change when the computer sleeps

Backup and Restore
Back up your computer | Restore files from backup

Windows Anytime Upgrade

Click “Device Manager”.



View basic information about your computer

Windows edition

Windows 7 Professional

Copyright © 2009 Microsoft Corporation. All rights reserved.

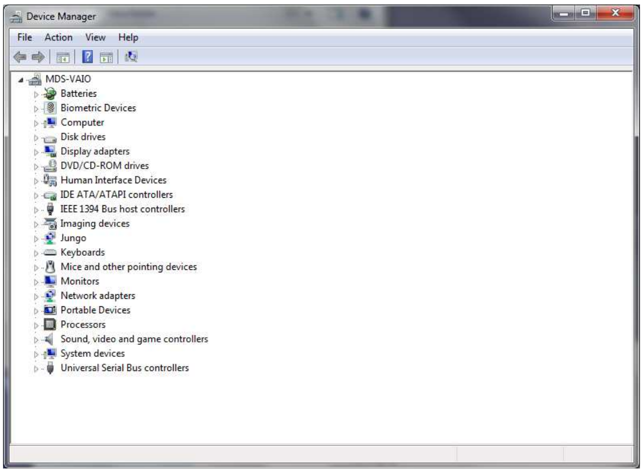
Service Pack 1

Get more features with a new edition of Windows 7

System

Rating:	5.0 Windows Experience Index
Processor:	Intel(R) Core(TM) i5 CPU M 520 @ 2.40GHz 2.40 GHz
Installed memory (RAM):	4.00 GB (3.87 GB usable)
System type:	64-bit Operating System
Pen and Touch:	No Pen or Touch Input is available for this Display

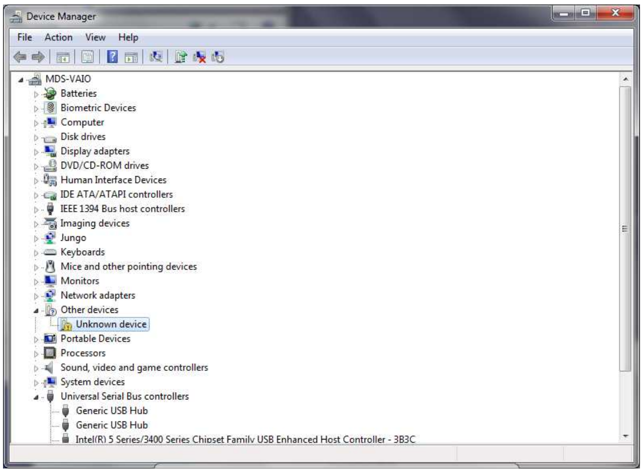
You can see following window.



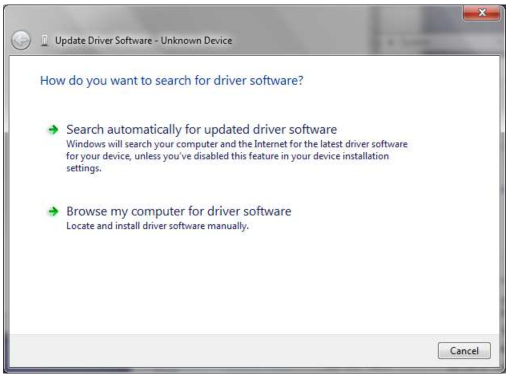
Connect OCD-I or OCD-II dongle hardware.

Move mouse point to following “Unknown device” and click right button.

Click “Update Driver Software”.



Click “Browse my computer for driver software”.



Update Driver Software - Unknown Device

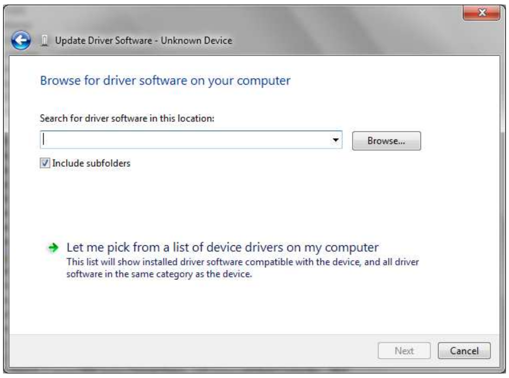
How do you want to search for driver software?

Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.

You have to fill driver path.

Click “Browse” button.



Update Driver Software - Unknown Device

Browse for driver software on your computer

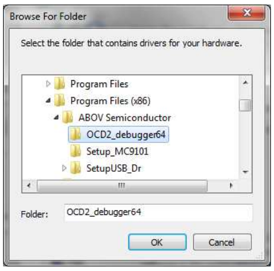
Search for driver software in this location:

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

Select installed folder and click “OK” button.

For example, “C:₩Program Files (x86)₩ABOV Semiconductor₩OCD2_debugger64”



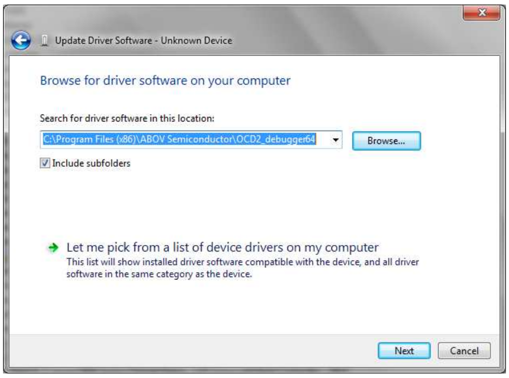
Browse For Folder

Select the folder that contains drivers for your hardware.

- Program Files
- Program Files (x86)
  - ABOV Semiconductor
    - OCD2_debugger64
    - Setup_MC9101
    - SetupUSB_Dr

Folder: OCD2_debugger64

Click “Next” button.



Update Driver Software - Unknown Device

Browse for driver software on your computer

Search for driver software in this location:

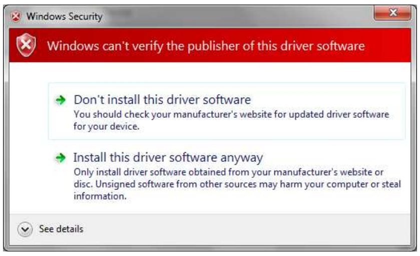
C:\Program Files (x86)\ABOV Semiconductor\OCD2_debugger64

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

MS-Windows will popup warning dialog to you as following.

Select “Install this driver software anyway”.



Windows Security

Windows can't verify the publisher of this driver software

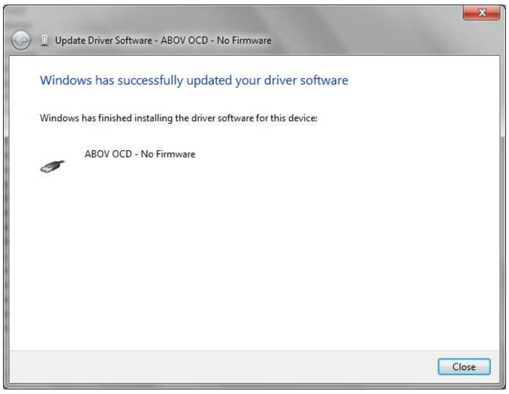
Don't install this driver software
You should check your manufacturer's website for updated driver software for your device.

Install this driver software anyway
Only install driver software obtained from your manufacturer's website or disc. Unsigned software from other sources may harm your computer or steal information.

OCD-I dongle hardware driver installation

If you connected OCD-I dongle hardware, “ABOV OCD – No Firmware” will be installed at first time. Following picture shows it.

You have to install hardware one more time.



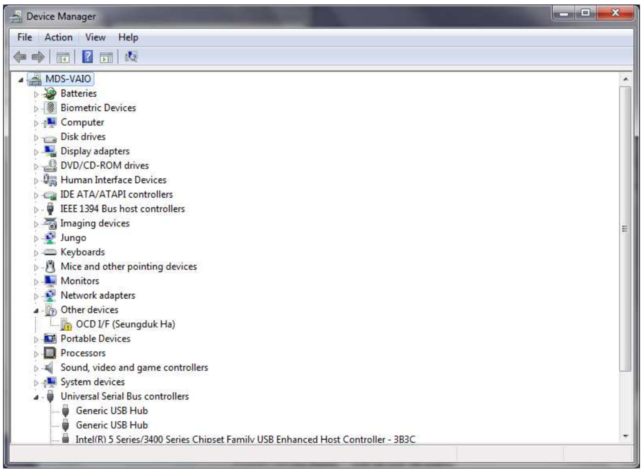
Update Driver Software - ABOV OCD - No Firmware

Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

ABOV OCD - No Firmware

See the device manager window.

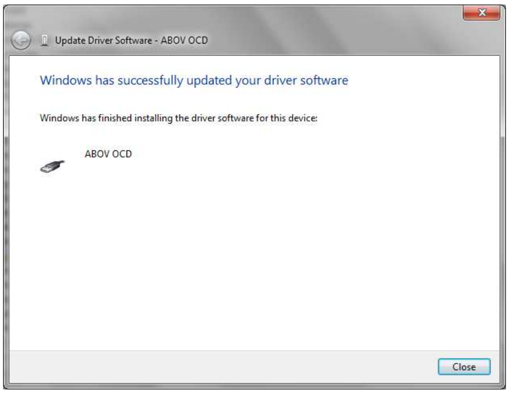


Move mouse point to “OCD I/F (Seungduk Ha)” and click right button.

Click “Update Driver Software”.

Do the same as above.

ABOV OCD-I dongle hardware is installed correctly.

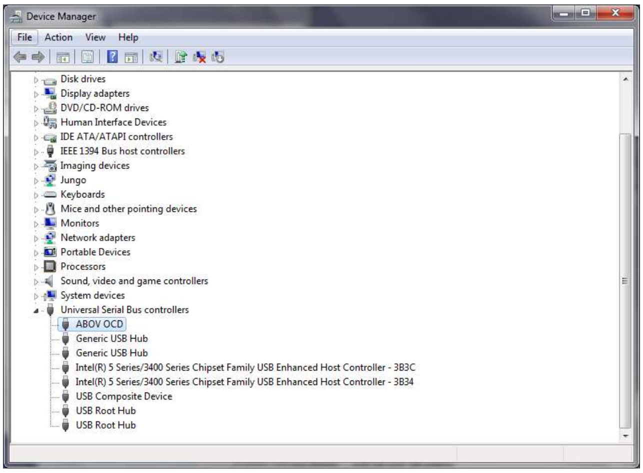


Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

ABOV OCD

You can confirm it within the Device Manager window.

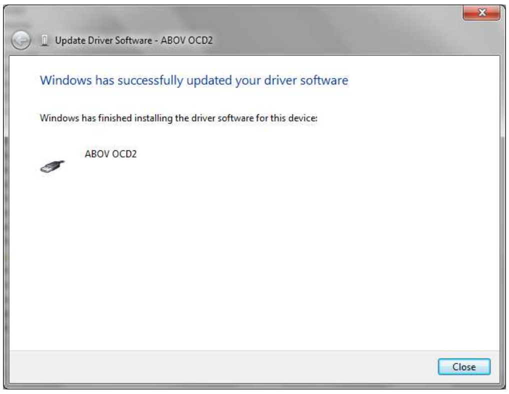


- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Human Interface Devices
- IDE ATA/ATAPI controllers
- IEEE 1394 Bus host controllers
- Imaging devices
- Jungo
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Portable Devices
- Processors
- Sound, video and game controllers
- System devices
- Universal Serial Bus controllers
  - ABOV OCD
  - Generic USB Hub
  - Generic USB Hub
  - Intel(R) 5 Series/3400 Series Chipset Family USB Enhanced Host Controller - 3B3C
  - Intel(R) 5 Series/3400 Series Chipset Family USB Enhanced Host Controller - 3B34
  - USB Composite Device
  - USB Root Hub
  - USB Root Hub

OCD-II dongle hardware driver installation

If you connected OCD-II dongle hardware, it will be installed just one time.

ABOV OCD-II dongle hardware is installed correctly.

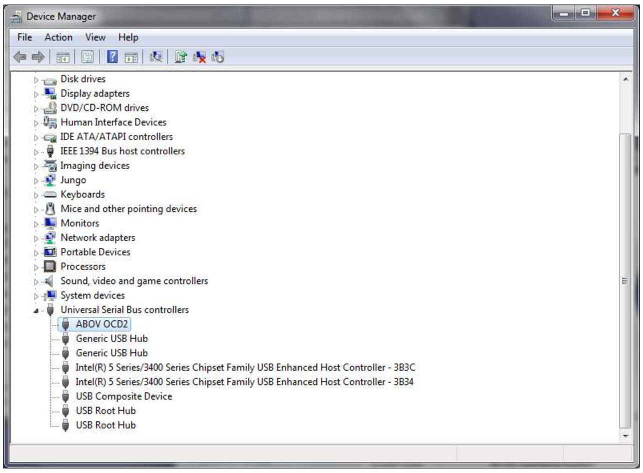


Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

ABOV OCD2

You can confirm it within the Device Manager window.



- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Human Interface Devices
- IDE ATA/ATAPI controllers
- IEEE 1394 Bus host controllers
- Imaging devices
- Jungo
- Keyboards
- Mice and other pointing devices
- Monitors
- Network adapters
- Portable Devices
- Processors
- Sound, video and game controllers
- System devices
- Universal Serial Bus controllers
  - ABOV OCD2
  - Generic USB Hub
  - Generic USB Hub
  - Intel(R) 5 Series/3400 Series Chipset Family USB Enhanced Host Controller - 3B3C
  - Intel(R) 5 Series/3400 Series Chipset Family USB Enhanced Host Controller - 3B34
  - USB Composite Device
  - USB Root Hub
  - USB Root Hub

Chapter 2
OCD-I / II dongle hardware

This chapter describes:

- OCD-I dongle and interface connection
- OCD-II dongle and interface connection
- Hardware connection sequence

2.1 OCD-I dongle and interface connection

OCD-I dongle hardware support ABOV 8bit MCU which adopted OCD-I interface logic.

2.1.1 Features of OCD-I dongle hardware

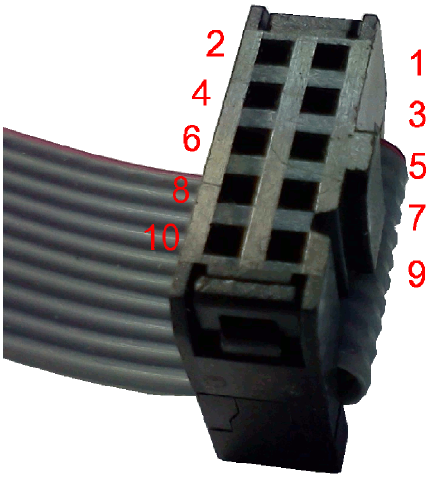
OCD-I dongle hardware is the cheapest debugging solution of ABOV Semiconductor Co.,Ltd. But its performance is good enough to debug target MCU device.

- 2 interface line : SCLK, SDATA
- 2 LED display : Power, Debug Run
- Target system operating voltage : 3 ~ 6V
- It can debug full range of the target device's operating frequency.
- It does not supply power to user's target system.
- It does not support Hot-Plug

  It means, your target system must not be powered during OCD-I dongle hardware insert or release.

2.1.2 Interface connection

Cable side view



Pin assignment

Pin #	Name	Function
1		
2	Vcc detect	It detects target system's power and use interface voltage level.
3		
4	Ground	System ground.
5		
6	SCLK	Serial clock of OCD-I interface..
7		
8	SDATA	Serial data of OCD-I interface. If your target system is very noisy, you had better adding a small capacitance to this line.
9		
10		

2.2 OCD-II dongle and interface connection

OCD-II dongle hardware support ABOV 8bit MCU which adopted OCD-I interface logic and OCD-II interface logic.

So, you can debug OCD-I MCU series and OCD-II MCU series by using this hardware.

2.2.1 Features of OCD-II dongle hardware

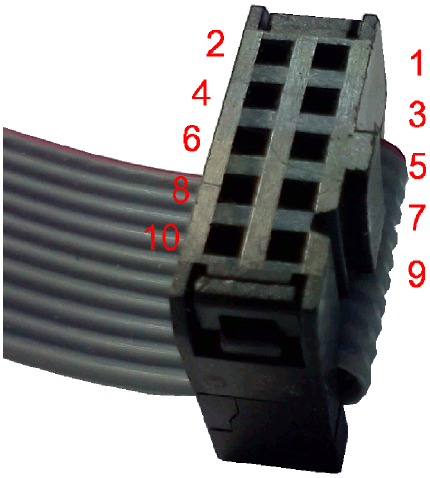
OCD-II dongle hardware is sophisticated debugging tool.

It is very flexible and faster than OCD-I dongle hardware.

- 2 interface line : SCLK, SDATA
  1 option line : RUNTIME (run time measuring)
- 1 LED display : Debug Run
- Fast interface speed than OCD-I dongle hardware.
- It can measure the exact operating time of the target MCU.
  In this case, RUNTIME must be connected.
  It guarantees micro second unit.
- Target system operating voltage : 3 ~ 6V
- It can debug full range of the target device's operating frequency.
- It does not supply power to user's target system.
- It does not support Hot-Plug
  It means, your target system must not be powered during OCD-I dongle hardware insert or release.

2.2.2 Interface connection

Cable side view



Pin assignment

Pin #	Name	Function
1		
2	Vcc detect	It detects target system's power and use interface voltage level.
3		
4	Ground	System ground.
5	RTIME (Option)	Run time measuring. This is not a mandatory OCD-II interface pin. OCD-II interface can work, even if this pin is not connected.
6	SCLK	Serial clock of OCD-II interface.
7		
8	SDATA	Serial data of OCD-II interface. If your target system is very noisy, you had better adding a small capacitance to this line.
9		
10		

2.3 Hardware connecting sequence

As mentioned before, OCD-I and OCD-II dongle hardware does not support Hot-Plug.
Hot-Plug means hardware plugging during target system is powered.

Dongle hardware will be damaged permanently by Hot-Plug.
So, you have to care about this.

2.3.1 OCD-I dongle connecting sequence

Even if the target MCU adopted OCD-I interface inside, it will not be entered to debug mode with wrong sequence.

- Power off your target system.
- Boot your PC.
- If OCD-I dongle is not connected with PC, connect it.
- Connect OCD-I dongle and your target system.
- Execute debugger software.
- Power on your target system

2.3.2 OCD-I dongle disconnecting sequence

Power off sequence is important too.
Wrong sequence may destroy OCD-I dongle hardware.

- Power off your target system first.
- The other sequences are not important.

2.3.3 OCD-II dongle connecting sequence

OCD-II dongle can support OCD-I interface and OCD-II interface.
But, OCD-I interface protocol and OCD-II interface protocol is different.
So, you have to select target protocol first.

Even if the target MCU adopted OCD-I or OCD-II interface inside, it will not be entered to debug mode with wrong sequence.

- Power off your target system.
- Boot your PC.
- If OCD-II dongle is not connected with PC, connect it.
- Connect OCD-I dongle and your target system.
- Execute debugger software.
- Select target OCD interface series in debugger software and wait until OCD-II dongle hardware is initialized.
- Power on your target system

2.3.4 OCD-II dongle disconnecting sequence

Power off sequence is important too.
Wrong sequence may destroy OCD-II dongle hardware.

- Power off your target system first.
- The other sequences are not important.

Chapter 3
Debugger software for MS-Windows (32bit, 64bit)

This chapter describes:

- Debugger software feature
- Menu usage
- Child windows

3.1 Debugger software feature

Debugger supports OCD-I dongle hardware and OCD-II dongle hardware.

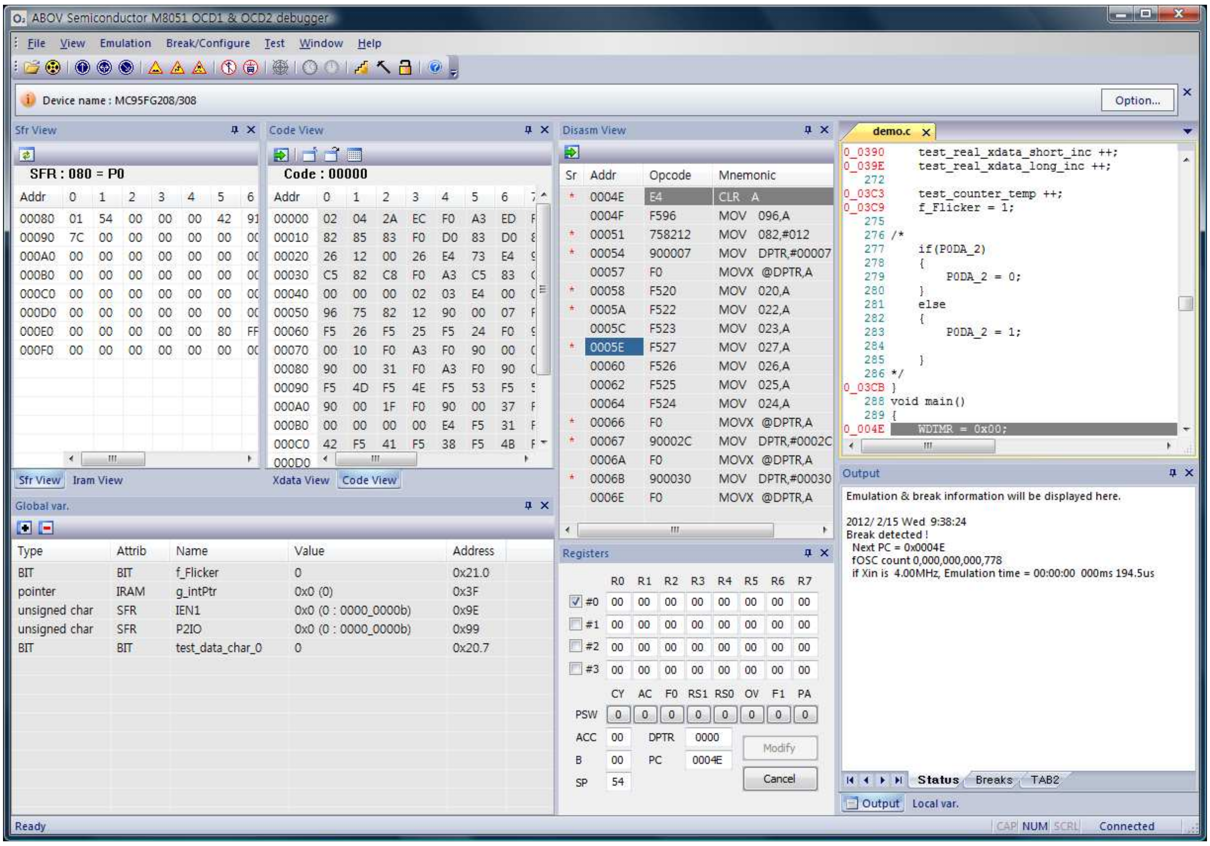
OCD-I dongle hardware does not need to initialize by debugger.

But OCD-II dongle hardware needs to initialize by debugger, because it can support OCD-I MCU series and OCD-II MCU series.

OCD-II interface protocol is not same to OCD-I interface protocol.

So, you have to select target MCU series

Ex) Debugger screen shot



Device name : MC95FG208/308

SFR : 080 = P0

Addr	0	1	2	3	4	5	6
00080	01	54	00	00	00	42	91
00090	7C	00	00	00	00	00	0
000A0	00	00	00	00	00	00	0
000B0	00	00	00	00	00	00	0
000C0	00	00	00	00	00	00	0
000D0	00	00	00	00	00	00	0
000E0	00	00	00	00	00	80	FF
000F0	00	00	00	00	00	00	0

Code : 00000

Addr	0	1	2	3	4	5	6
00000	02	04	2A	EC	F0	A3	ED
00010	82	85	83	F0	D0	83	D0
00020	26	12	00	26	E4	73	E4
00030	C5	82	C8	F0	A3	C5	83
00040	00	00	00	02	03	E4	00
00050	96	75	82	12	90	00	07
00060	F5	26	F5	25	F5	24	F0
00070	00	10	F0	A3	F0	90	00
00080	90	00	31	F0	A3	F0	90
00090	F5	4D	F5	4E	F5	53	F5
000A0	90	00	1F	F0	90	00	37
000B0	00	00	00	00	E4	F5	31
000C0	42	F5	41	F5	38	F5	4B
000D0							

Sr	Addr	Opcode	Mnemonic
*	0004E	E4	CLR A
	0004F	F596	MOV 096,A
*	00051	758212	MOV 082,#012
*	00054	900007	MOV DPTR,#00007
	00057	F0	MOVX @DPTR,A
*	00058	F520	MOV 020,A
*	0005A	F522	MOV 022,A
	0005C	F523	MOV 023,A
*	0005E	F527	MOV 027,A
	00060	F526	MOV 026,A
	00062	F525	MOV 025,A
	00064	F524	MOV 024,A
*	00066	F0	MOVX @DPTR,A
*	00067	90002C	MOV DPTR,#0002C
	0006A	F0	MOVX @DPTR,A
*	0006B	900030	MOV DPTR,#00030
	0006E	F0	MOVX @DPTR,A

```
0_0390      test_real_xdata_short_inc ++;
0_039E      test_real_xdata_long_inc ++;
   272
0_03C3      test_counter_temp ++;
0_03C9      f_Flicker = 1;
   275
   276 /*
   277      if(P0DA_2)
   278      {
   279          P0DA_2 = 0;
   280      }
   281      else
   282      {
   283          P0DA_2 = 1;
   284
   285      }
   286 */
0_03CB }
   288 void main()
   289 {
0_004E      WDTMR = 0x00;
```

Type	Attrib	Name	Value	Address
BIT	BIT	f_Flicker	0	0x21.0
pointer	IRAM	g_intPtr	0x0 (0)	0x3F
unsigned char	SFR	IEN1	0x0 (0 : 0000_0000b)	0x9E
unsigned char	SFR	P2IO	0x0 (0 : 0000_0000b)	0x99
BIT	BIT	test_data_char_0	0	0x20.7

	R0	R1	R2	R3	R4	R5	R6	R7
☑ #0	00	00	00	00	00	00	00	00
□ #1	00	00	00	00	00	00	00	00
□ #2	00	00	00	00	00	00	00	00
□ #3	00	00	00	00	00	00	00	00

	CY	AC	F0	RS1	RS0	OV	F1	PA
PSW	0	0	0	0	0	0	0	0

ACC 00 DPTR 0000

B 00 PC 0004E

SP 54

```
Emulation & break information will be displayed here.

2012/ 2/15 Wed  9:38:24
Break detected !
 Next PC = 0x0004E
 fOSC count 0,000,000,000,778
 if Xin is  4.00MHz, Emulation time = 00:00:00  000ms 194.5us
```

Ready

Connected

3.1.1 Common feature

It supports MC9x series of ABOV Semiconductor Co.,Ltd.

Followings are commonly supported.

It does not care for OCD-I & OCD-II dongle hardware and device series

- It detects target device automatically.
- It uses symbolic debugging.
  - Source file view.
  - Global / Local variables view.
  - Each device's SFR (Special Function Register) names.
- It displays various target memory.
  - CODE, XDATA, IDATA, SFR.
  - You can edit these data directly in debugger.
- It displays code data using disassembled format.
- It supports line assemble.
- It can toggle breaks.
- It supports Hex file download.

  It can be used as ISP (In System Programming).
- It can calculate code checksum.
- It supports following emulation methods.
  - Real time emulation.
  - Step emulation (source line level or code level).
  - Emulation aborting.
- It saves and loads the last debugging environment automatically.

3.1.2 OCD- I dongle only feature

OCD-I dongle hardware does not need to initialized before use.

By default, it is initialized for OCD-I series.

- It can support all of OCD-I device series.
- OCD-I devices have 8 PC (Program Counter) breaks.
- It can NOT support all of OCD-II device series.

3.1.3 OCD-I I dongle only feature

OCD-II dongle hardware must be initialized before use.

Because of, OCD-II dongle hardware could not recognize what kind of OCD series will be used.

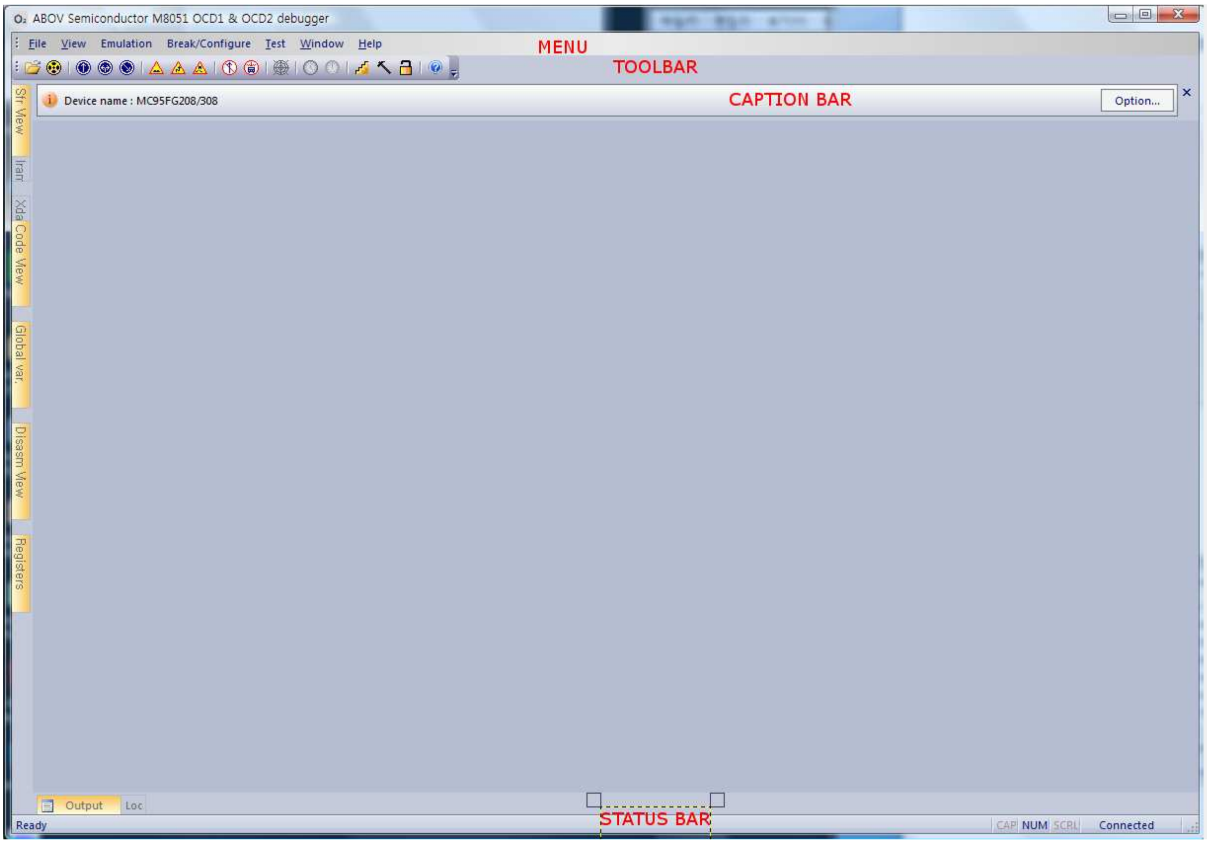
- It can support all of OCD-I device series.
  In this case, the feature is the same to OCD-I dongle hardware only feature.
- It can support all of OCD-II device series.

OCD-II device only feature

- 12 breaks are prepared.
  - Fixed 4 PC (Program Counter) breaks.
  - The other 8 breaks can be combined to various event breaks.
    PC break
    Access break (bit, byte short, long)
      Support signed / unsigned
      Support Big endian / Little endian.
- It can display run time data monitoring.
  Global variable's values are updated automatically during emulation time.
- It can measure target device's operating frequency.
- It can trim device's internal OSC frequency.
- It supports emulation time measuring.
  NOTE : RTIME pin must be connected to measure emulation time.

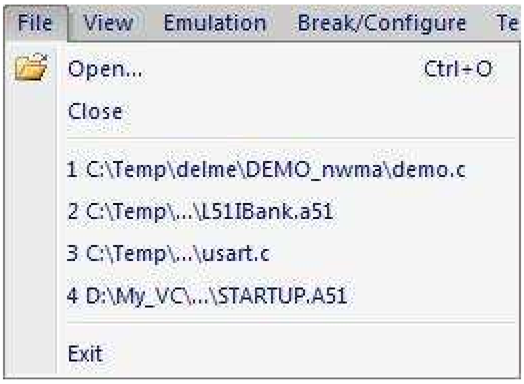
3.2 Menu usage

This section gives an overview of the menu options.



3.2.1 File

The File menu displays the following options:


File
View
Emulation
Break/Configure
Te
Open...
Ctrl+O
Close
1 C:\Temp\delme\DEMO_nwma\demo.c
2 C:\Temp\...\L51IBank.a51
3 C:\Temp\...\usart.c
4 D:\My_VC\...\STARTUP.A51
Exit

Open...

It reads text file from HDD, and open a child text window to display.

Shortcut key is Control + O.

Close

It closes top most child text window.

Recent File List

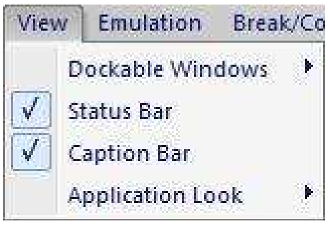
It displays a list of the 4 most recent files you have read.

Exit

It quits from the debugger software.

3.2.2 View

The View menu controls the display of the debugger software frame and child windows.

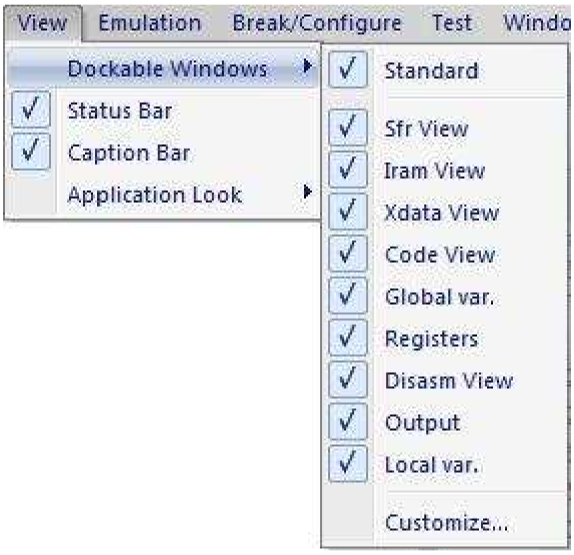


Dockable Windows

It shows or hides variety child views.

If child view is checked, the selected child view will be shown.

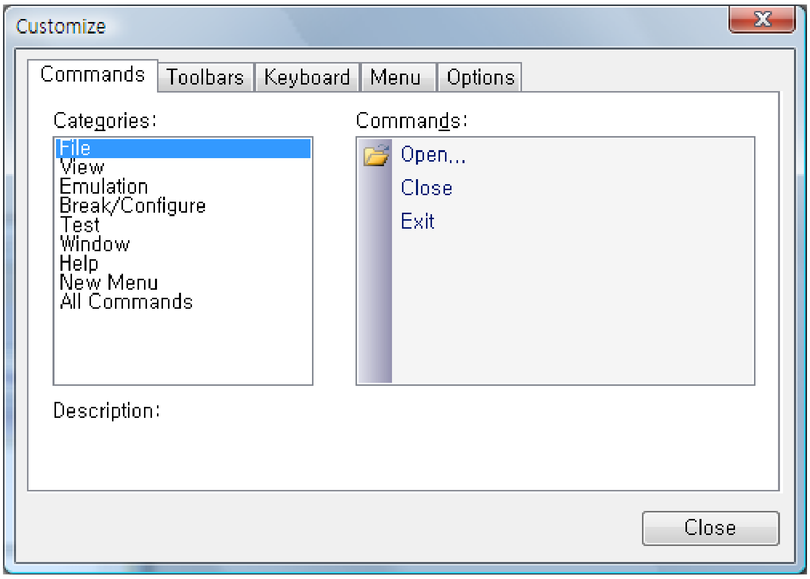
Or not, the child will be hidden.



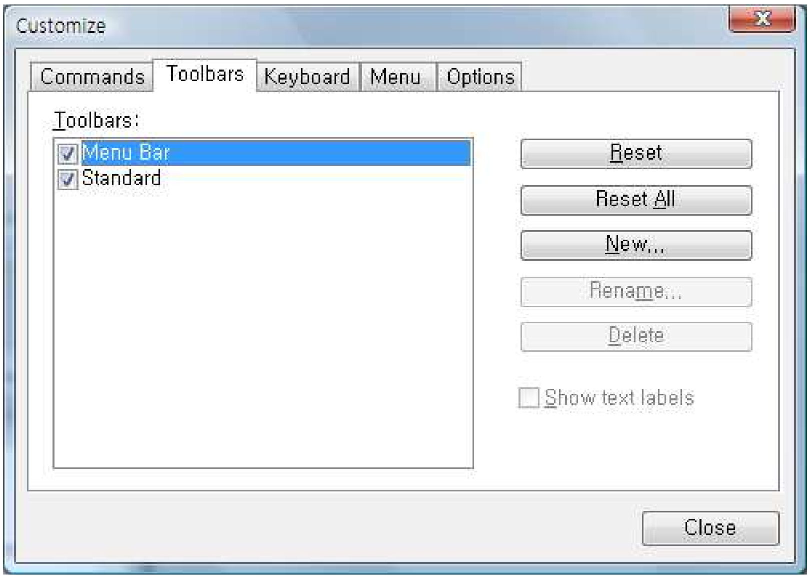
Customize

It offers to modify debugger software Command, Toolbar, Keyboard, Menu, Options to user. So, each user can change debugger software GUI environment to their taste.

It offers to modify each menu's sub-item

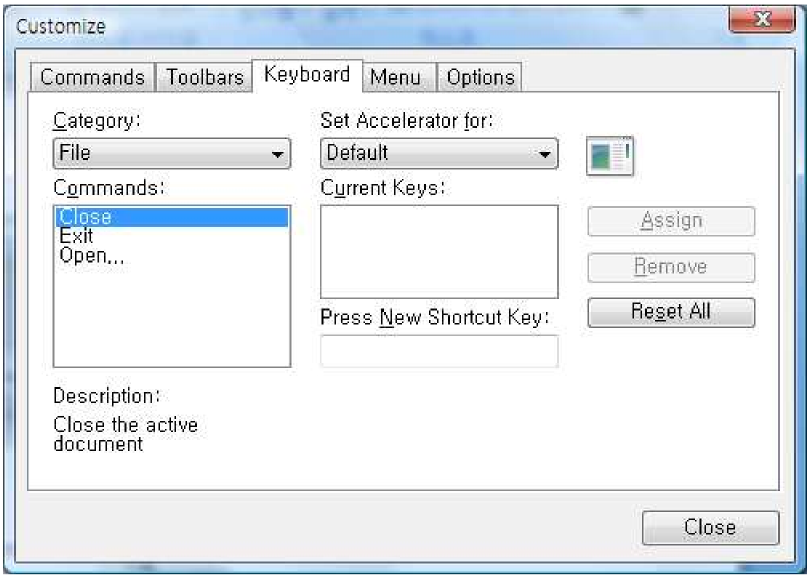


It offers Toolbar style



You can define each Command's Shortcut keys.

And, you can reset it or return to default setting.



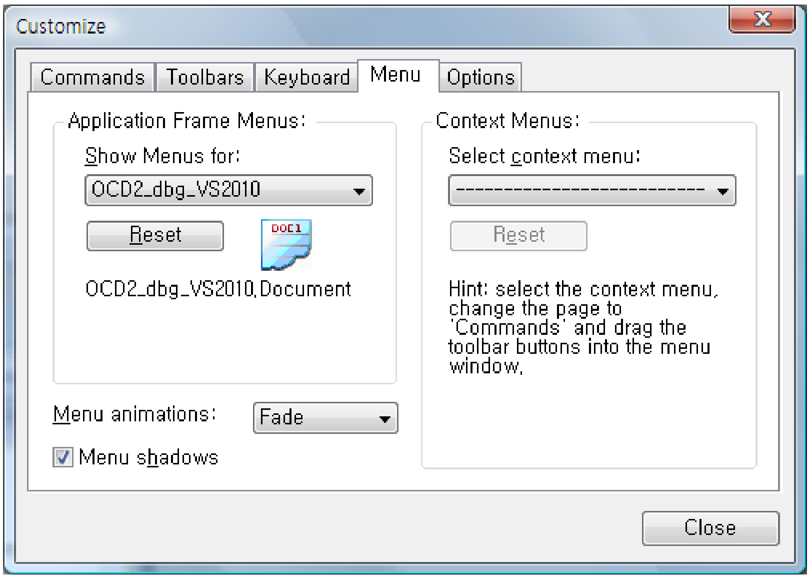
Category: File

Set Accelerator for: Default

Commands:
Close
Exit
Open...

Description:
Close the active document

You can change the Menu style.



Show Menus for: OCD2_dbg_VS2010

OCD2_dbg_VS2010.Document

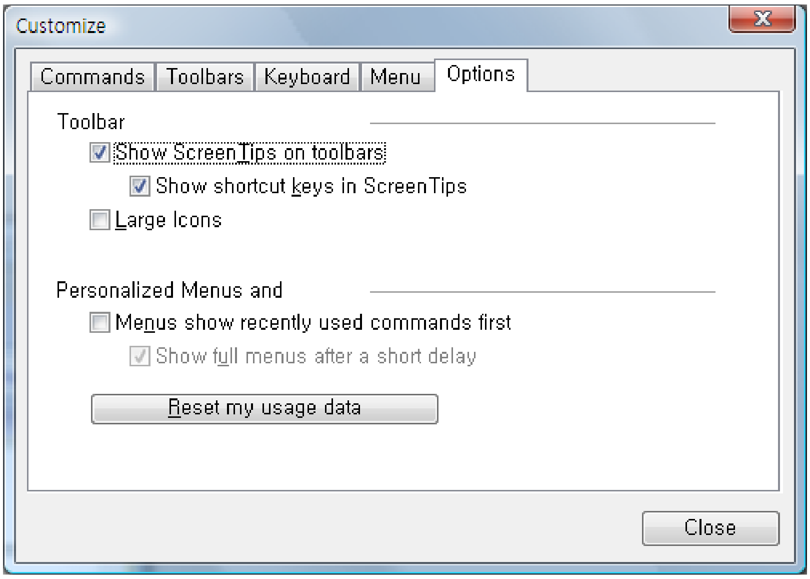
Select context menu: --------------------------

Hint: select the context menu, change the page to 'Commands' and drag the toolbar buttons into the menu window.

Menu animations: Fade

☑ Menu shadows

You can change Toolbar tip display, Icon size, etc.



Toolbar

☑ Show ScreenTips on toolbars

☑ Show shortcut keys in ScreenTips

□ Large Icons

Personalized Menus and

□ Menus show recently used commands first

☑ Show full menus after a short delay

Status Bar

It turns the status bar on or off.

The Status bar displays information on the current state of debugger.

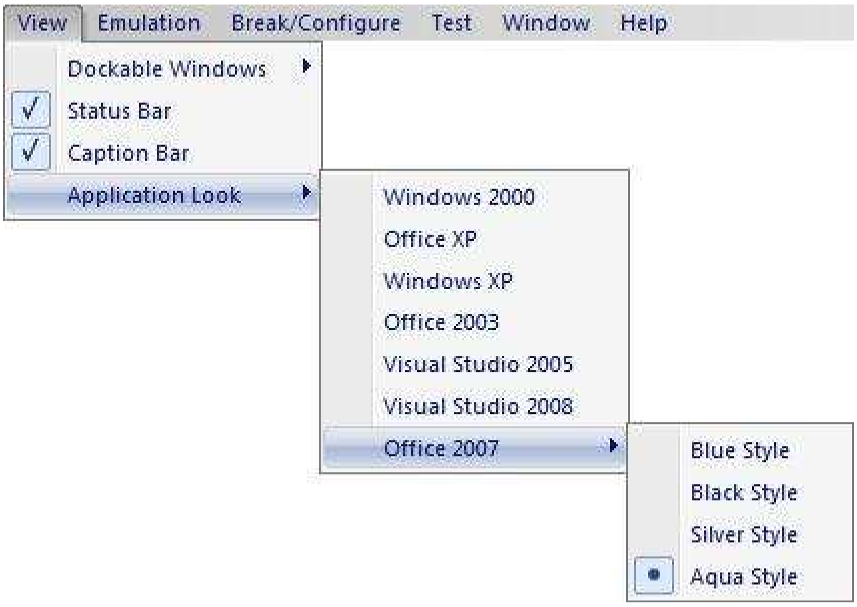
Caption Bar

It turns the Caption bar on or off.

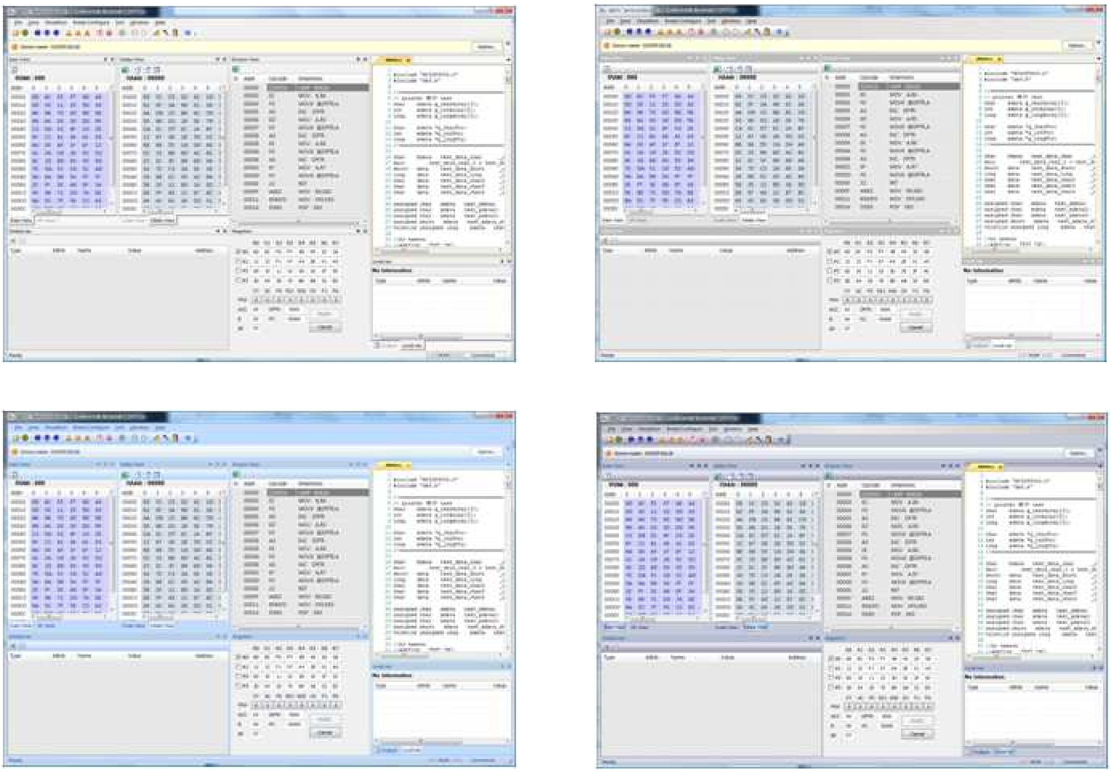
The Caption bar displays device name which is connected with OCD-I or OCD-II dongle hardware.

Application Look

It changes debugger software's GUI style at once.

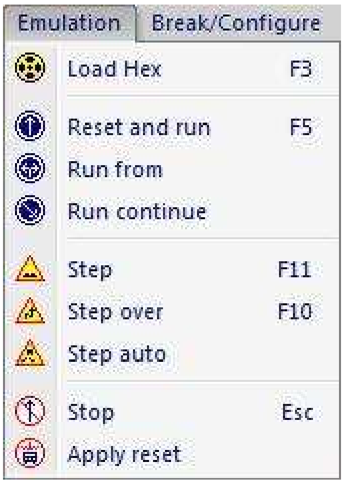


Ex) Changed Look



3.2.3 Emulation

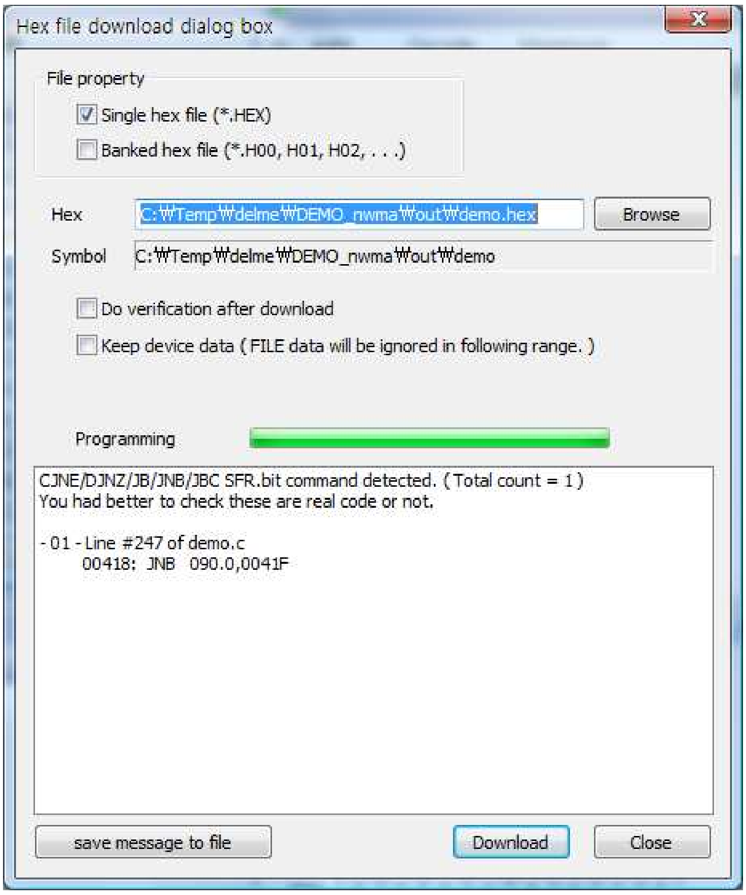
The Emulation menu controls the stopping and start of core.



Load Hex

It displays a dialog box that you use to enter the hex file name.

Connected device will be programmed using this hex file.



File property

☑ Single hex file (*.HEX)

☐ Banked hex file (*.H00, H01, H02, . . .)

Hex C:₩Temp₩delme₩DEMO_nwma₩out₩demo.hex

Symbol C:₩Temp₩delme₩DEMO_nwma₩out₩demo

☐ Do verification after download

☐ Keep device data (FILE data will be ignored in following range.)

Programming

```
CJNE/DJNZ/JB/JNB/JBC SFR.bit command detected. ( Total count = 1 )
You had better to check these are real code or not.

- 01 - Line #247 of demo.c
      00418:  JNB   090.0,0041F
```

File property

If your target device size is smaller than or equal to 64KB, compiler generates Single hex file only.

If your target device's code area is bigger than 64KB, you have to select hex file property.

- Single hex file : only 1 hex file, Linear address.
- Banked hex file : more than 2 hex file, Banked address.

Hex

Hex file name to download.

Symbol

Symbol file name to use by debugger software.

It depends on Hex file name.

Do verification after download

If it is checked, debugger will verify the code memory, after hex file download.

Keep device data (FILE data will be ignored following range.)

You can keep device's data rather than Hex file data within specified address range.

If it is checked, you have to input address range.

CJNE/DJNZ/JB/JNB/JBC SFR.bit command detection

Some old devices have instruction bugs.

It is bit compare and branch instruction.

Debugger detects these instructions during hex file download.

But, debugger could not distinguish between instruction and data pattern.

So, you had better to check these detected output is real instruction or not.

Save message to file

It saves "CJNE/DJNZ/JB/JNB/JBC SFR.bit command detection" list as a file.

You can use this information when you modify your source code.

Download

Hex file will be downloaded.

Close

Close this dialog box.

If the target device has configurations, configuration dialog box will be appeared.

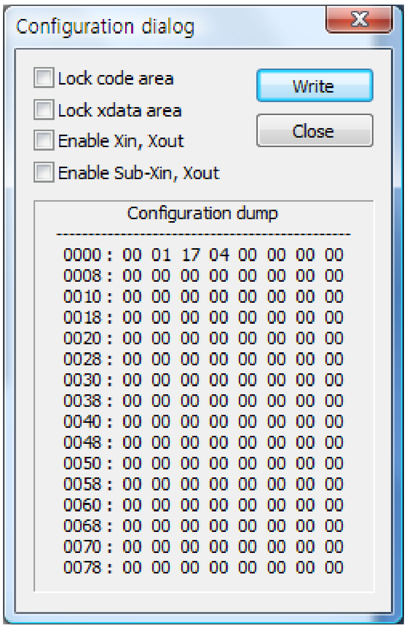
Configuration dialog box is slightly different for each device series.

Because of, each device series have different configurations

Activate device configuration.

- Write configuration.
- Power off the target system.
- Power on the target system.

Device catch configuration(s) during power is rising to operation voltage.



□ Lock code area
□ Lock xdata area
□ Enable Xin, Xout
□ Enable Sub-Xin, Xout

```
Configuration dump
-----------------------------------------
0000 :  00  01  17  04  00  00  00  00
0008 :  00  00  00  00  00  00  00  00
0010 :  00  00  00  00  00  00  00  00
0018 :  00  00  00  00  00  00  00  00
0020 :  00  00  00  00  00  00  00  00
0028 :  00  00  00  00  00  00  00  00
0030 :  00  00  00  00  00  00  00  00
0038 :  00  00  00  00  00  00  00  00
0040 :  00  00  00  00  00  00  00  00
0048 :  00  00  00  00  00  00  00  00
0050 :  00  00  00  00  00  00  00  00
0058 :  00  00  00  00  00  00  00  00
0060 :  00  00  00  00  00  00  00  00
0068 :  00  00  00  00  00  00  00  00
0070 :  00  00  00  00  00  00  00  00
0078 :  00  00  00  00  00  00  00  00
```

Reset and run

It starts emulation from address 0000h, after reset the target device.

Its action is the same to the real situation.

Emulation will be continued until break is occurred or developer stop emulation.

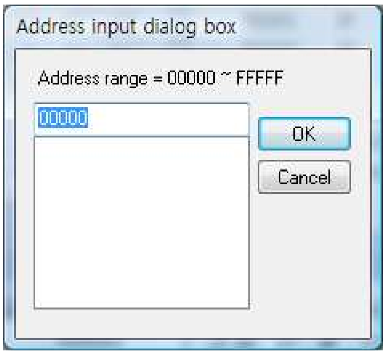
Run from

It starts emulation from user specified address.

It is used to debug each software module.

It asks emulation start address as below.

You can fill 20bit address directly.



Address range = 00000 ~ FFFFF

00000

Run continue

It starts emulation from device's current address.

Current means:

- Stopped address of previous emulation.
- If device was reset, it is 0000h.

Step

If you want to follow your code's execution more closely, you can step through the code.

The program moves the next line of source code, or next mnemonic code.

Source line unit Step run asks tens of or hundreds of mnemonic code unit Step run.

So, Source line unit Step run is slower than mnemonic code unit Step run

You can select above by using "Step run option" of Break/Configure menu.

Step over

The program moves to the next line of code or next mnemonic code.

It does NOT work perfectly yet. It will be updated in a future.

Step auto

It executes Step run every 100ms.

Its execution will be continued unless you halt it by Stop

Stop

It halts current emulation immediately.

Apply reset

Target OCD devices have variety reset source as following.

- Power ON reset.
- External reset pin input.
- Watch-dog reset.
- OCD debugger's command reset

These results are wholly same. Target device will be reset.

This menu act OCD debugger's command reset.

3.2.4 Break/Configure

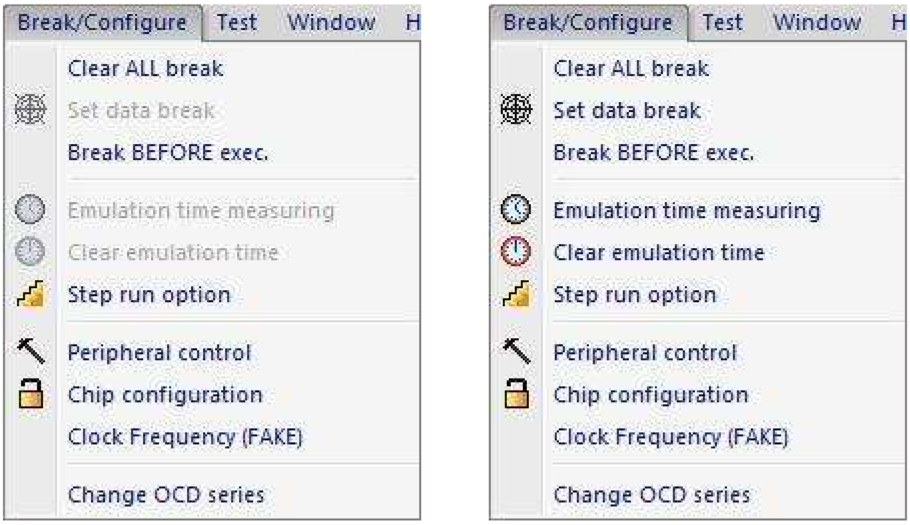
It controls BREAK settings, emulation options, device configurations, etc.

You can emulate your program more sophistically by using these controls.

Some menus are not work with OCD-I device series.

Because of, OCD-I interface SPEC. does not support these functions.

Ex) Menu difference between OCD-I device series and OCD-II device series

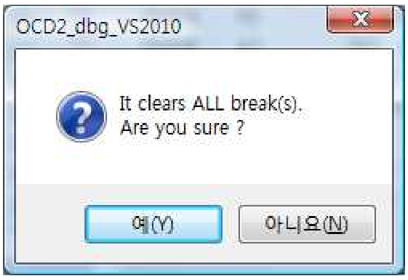


Clear ALL break

It clears all PC-breaks and data breaks (OCD-II devices only).

If break(s) were defined already, it asks as following.

If there is no break definition, it does not ask anything.


OCD2_dbg_VS2010

It clears ALL break(s).
Are you sure ?

Set data break

This menu works only for OCD-II devices.

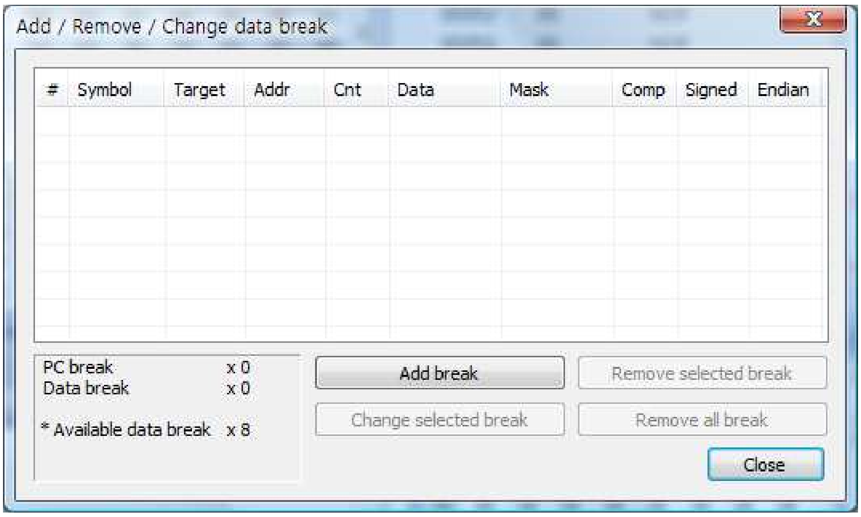
OCD-II break function is more powerful and flexible than OCD-I devices.

OCD-II supports 12 break event triggers.

- 4 of them are fixed to PC breaks.
- Rest of 8 breaks can be used combined or individual event triggers.
  It is called data break.
- Individual event triggers.
  PC break, Byte break, Bit break.
- Combined event triggers.
  2byte break (int, short type), 4byte break (long type).

This menu shows current break conditions as following dialog box.

You can Add / Remove / Change data breaks here.



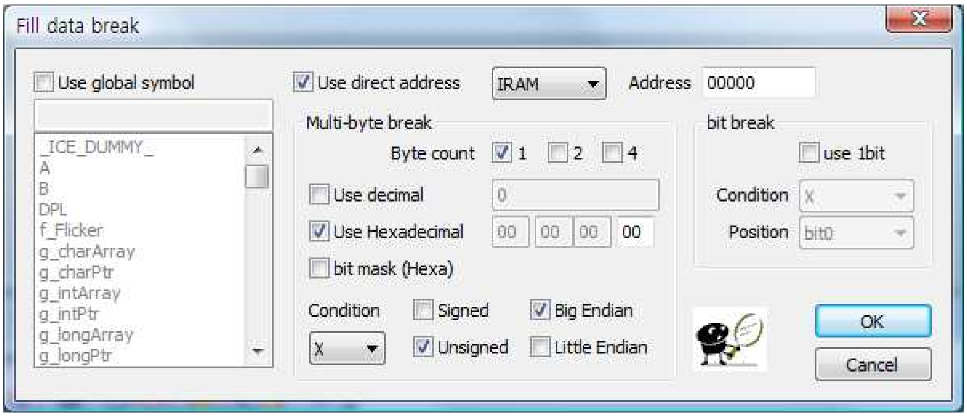
Add / Remove / Change data break

#	Symbol	Target	Addr	Cnt	Data	Mask	Comp	Signed	Endian

PC break x 0
Data break x 0

* Available data break x 8

Fill data break dialog box



Fill data break

☐ Use global symbol

_ICE_DUMMY_
A
B
DPL
f_Flicker
g_charArray
g_charPtr
g_intArray
g_intPtr
g_longArray
g_longPtr

☑ Use direct address IRAM Address 00000

Multi-byte break

Byte count ☑ 1 ☐ 2 ☐ 4

☐ Use decimal 0

☑ Use Hexadecimal 00 00 00 00

☐ bit mask (Hexa)

Condition X ☐ Signed ☑ Unsigned ☑ Big Endian ☐ Little Endian

bit break

☐ use 1bit

Condition X

Position bit0

It helps setting data break conditions.

You can select target by direct address or symbol name.

You can set the target memory (CODE, XDATA, IRAM, SFR).

- Multi-byte break
  - Supports comparing byte count.
  - Supports decimal number or hexadecimal number to compare.
  - Supports bit mask, signed / unsigned, Big / Little endian type.
  - Supports comparing condition.

X	: Don't care
!=	: Not equal
==	: Equal
>	: Great than
>=	: Great or equal
<	: Less than
<=	: Less or equal

if you select '<=' and input number is 56 (decimal), data break's condition will be operates as following

- if (target device's value <= 56) BREAK occur;

- bit break
  - Supports comparing condition.

X	: Don't care
!=	: Not equal
==	: Equal

  - You can change bit position.

Basic knowledge

- bit mask

  It is used to data compare with specified bits only.

  If bit7 of bit mask value is 1, bit7 will not be used to data compare.

  Its default value is 0x00.

  If data length is more than 1byte, bit mask is not supported by OCD-II SPEC..

- Signed / Unsigned

  Signed variable use the variable's MSB(Most Significant Bit) as + or – sign.

  Unsigned variable use the variable's MSB as a number.

  Ex) Signed / Unsigned variable's value range

Byte	Signed value		Unsigned value	
count	Min.	Max.	Min.	Max.
1	-128	127	0	255
2	-32,768	32,767	0	65,535
4	-2,147,483,648	2,147,483,647	0	4,294,967,295

- Endian

  It is data placing method.

  Keil C-compiler use Big Endian.

  The most common cases refer to how bytes are ordered within a single 16, 32, or 64 word, and endianness is then the same as byte order. The usual contrast is whether the most significant or least significant byte is ordered first—i.e., at the lowest byte address—within the larger data item.

  - Big endian

    It stores the most significant byte first.

  - Little endian

    It stores the least significant byte first.

Break BEFORE / AFTER exec.

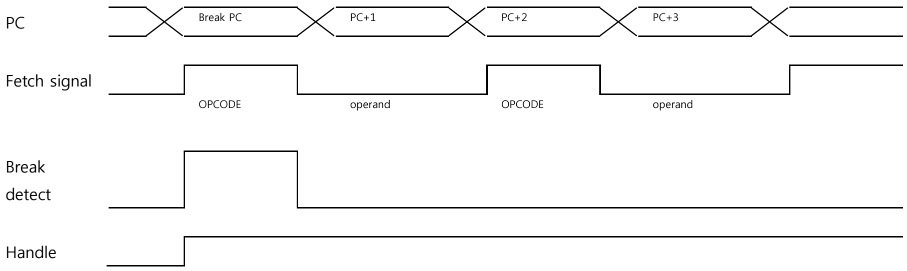
It changes break event detection time.

You can toggle by click this menu.

- "Break BEFORE exec."

  If break event was detected, target device is stopped immediately.

  The OPCODE of Break PC is not executed.

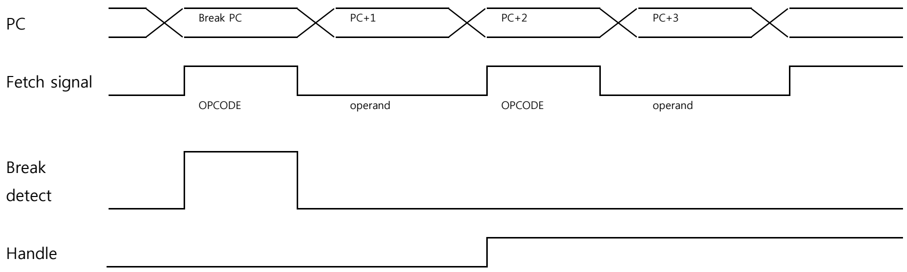


- "Break AFTER exec."

  If break event was detected, target device is not stopped yet.

  The OPCODE of Break PC is executed.

  And then target device will be stopped.



Emulation time measuring

This menu works only for OCD-II devices.

It shows / changes emulation time measuring option.

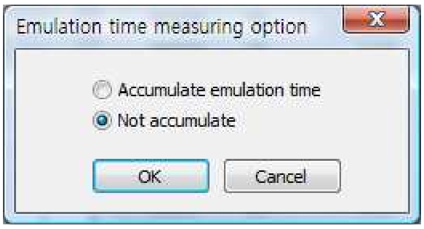
- Accumulate emulation time

  It does not clear the last emulation time and execution clock information whenever emulation is started.

  OCD-I does not support this.

- Not accumulate

  It clears the last emulation time and execution clock information whenever emulation is started.



Emulation time measuring option

○ Accumulate emulation time

● Not accumulate

Clear emulation time

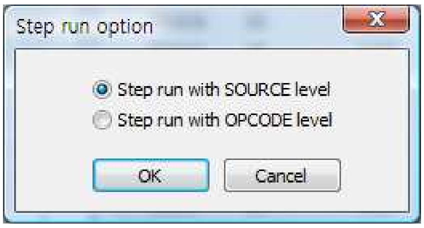
This menu works only for OCD-II devices.

It clears the last emulation time and execution clock information.

Step run option

Step run unit is one of source line level and OPCODE level.

You can select it here.



Step run option

● Step run with SOURCE level

○ Step run with OPCODE level

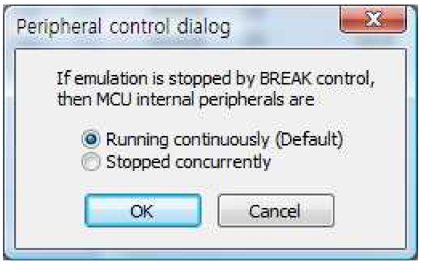
Peripheral control

It asks to you to select target device's peripheral works or not during idle mode.

It is used usually timer interrupt timing measuring.

It does not control each peripheral's operation individually.

So, you have to care to use this.



Peripheral control dialog

If emulation is stopped by BREAK control,
then MCU internal peripherals are

● Running continuously (Default)
○ Stopped concurrently

Chip configuration

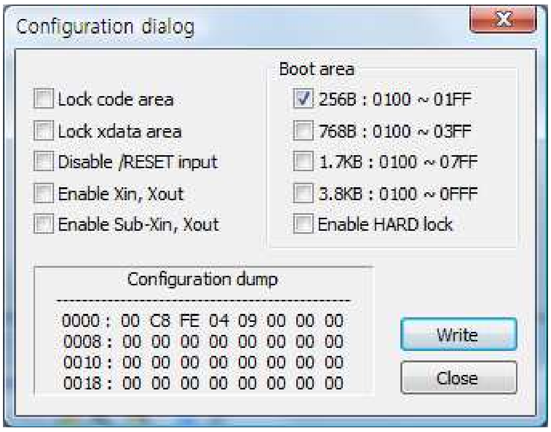
It is used to configure the target device's hardware configuration.

For example, code protection, oscillation control, I/O port option, etc.

Each device series has different configurations.

If you attempted unlock a locked device, then the device will be erased all of its data.

This is device specification that protect its data from hacking.



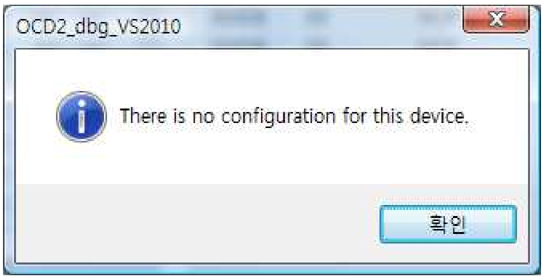
Configuration dialog

□ Lock code area
□ Lock xdata area
□ Disable /RESET input
□ Enable Xin, Xout
□ Enable Sub-Xin, Xout

Boot area
☑ 256B : 0100 ~ 01FF
□ 768B : 0100 ~ 03FF
□ 1.7KB : 0100 ~ 07FF
□ 3.8KB : 0100 ~ 0FFF
□ Enable HARD lock

```
Configuration dump
--------------------------------------------
0000 :  00  C8  FE  04  09  00  00  00
0008 :  00  00  00  00  00  00  00  00
0010 :  00  00  00  00  00  00  00  00
0018 :  00  00  00  00  00  00  00  00
```

If target device configuration is not exist, you can see below dialog box.



OCD2_dbg_VS2010

There is no configuration for this device.

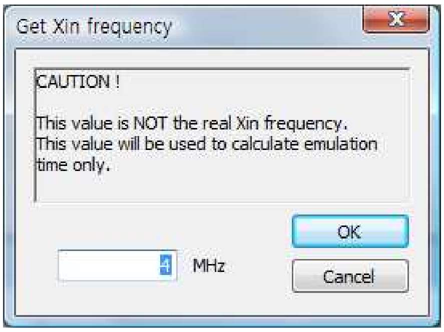
Clock Frequency (FAKE)

OCD-I interface does not support measuring emulation time.

OCD-II supports it, but you have to connect RTIME pin.

This clock frequency is FAKE, not the real emulation clock.

This value will be used to calculate emulation time from device's operated clock count.



Get Xin frequency

CAUTION !

This value is NOT the real Xin frequency.
This value will be used to calculate emulation time only.

4 MHz

If you have connected with OCD-II and RTIME, this value will be ignored.

Because of, debugger can get the real emulation time.

Change OCD series

OCD-dongle can detect most of its supporting device series automatically.

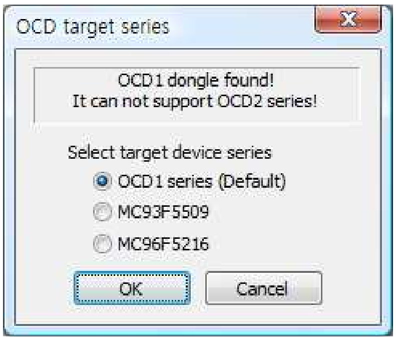
But, some devices have slightly different interface algorithm.

In that case, OCD-dongle must be re-configured to interface these devices.

If you want to change the target OCD series, do following sequence.

- Select target device series and click "OK" button.
- Turn off your target system and click "Yes" button.
- Wait under 1 second.
- Turn on your target system.

Ex) Dialog box for OCD-I dongle hardware



OCD target series

OCD1 dongle found!
It can not support OCD2 series!

Select target device series

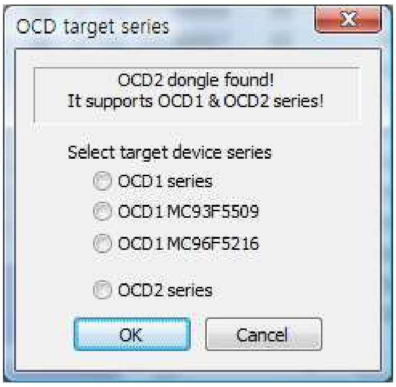
● OCD1 series (Default)
○ MC93F5509
○ MC96F5216

Ex) Dialog box for OCD-II dongle hardware

It does not have default device series.

You must select one of these series.

Or not, this dialog box will not be closed.

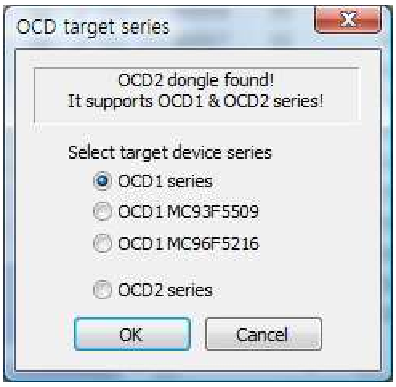


OCD target series

OCD2 dongle found!
It supports OCD1 & OCD2 series!

Select target device series

○ OCD1 series
○ OCD1 MC93F5509
○ OCD1 MC96F5216
○ OCD2 series



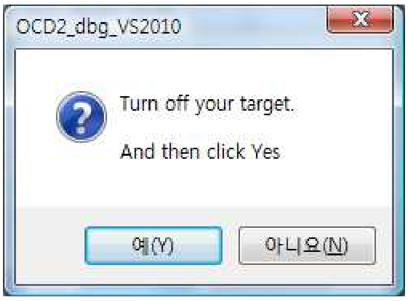
OCD target series

OCD2 dongle found!
It supports OCD1 & OCD2 series!

Select target device series

● OCD1 series
○ OCD1 MC93F5509
○ OCD1 MC96F5216
○ OCD2 series

You must turn off your target during OCD dongle is re-configuring.

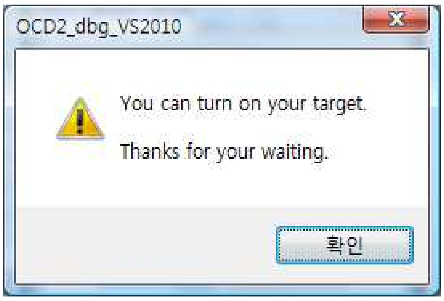


OCD2_dbg_VS2010

Turn off your target.

And then click Yes

You can turn on your target system when OCD dongle re-configuring was finished.



OCD2_dbg_VS2010

You can turn on your target.

Thanks for your waiting.

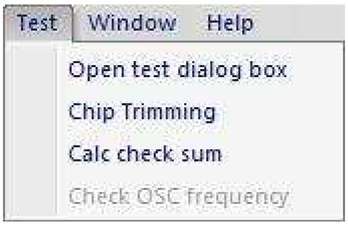
3.2.5 Test

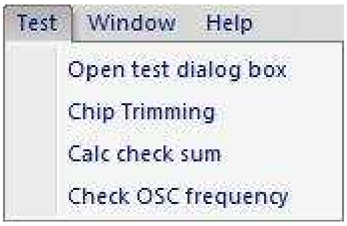
It is used to examine dongle hardware or device.

These functions help ABOV Semiconductor's MDS development, not for customers

The other functions work for users.

Ex) Menu difference between OCD-I device series and OCD-II device series



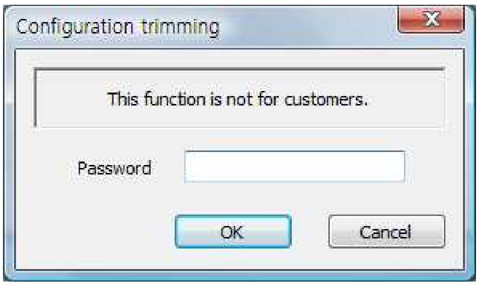


Open test dialog box

This function is not for users.

It is used for OCD-I and OCD-II dongle hard testing or repairing.

It is necessary to enter password, or not it will not be work.



Configuration trimming

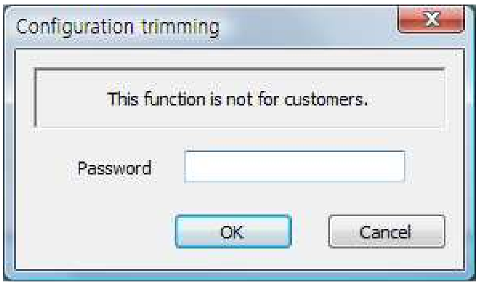
This function is not for customers.

Chip Trimming

This function is not for users.

It is used for device configuration changing, include user configuration and trimming values.

It is necessary to enter password, or not it will not be work.



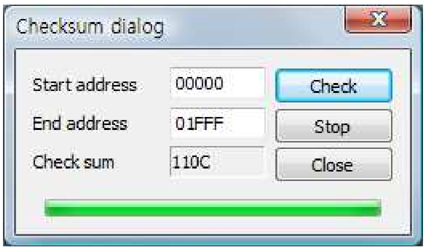
Configuration trimming

This function is not for customers.

Calc check sum

It reads target device's code memory and displays checksum.

If target device is protected by Lock configuration, it could not read target device properly.



Checksum dialog

Start address 00000

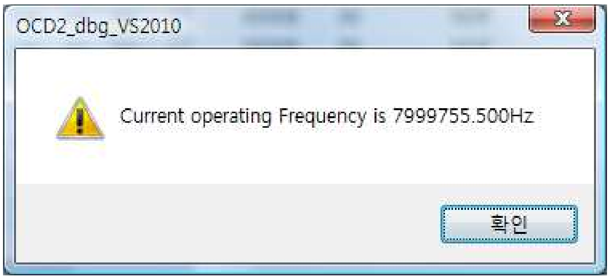
End address 01FFF

Check sum 110C

Check OSC frequency

This menu works only for OCD-II devices.

It shows target device's oscillation frequency.

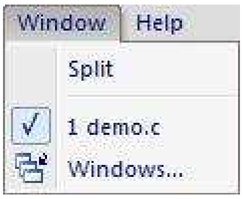


OCD2_dbg_VS2010

Current operating Frequency is 7999755.500Hz

3.2.6 Window

It controls text file window’s view.

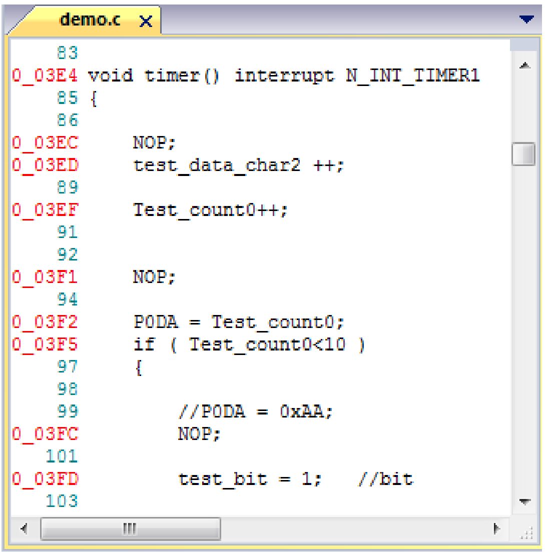


Split

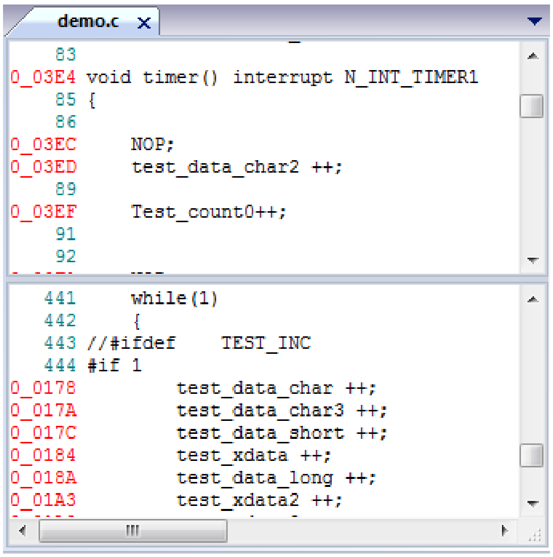
You can split text file window’s view like following example.

You can move or remove the splitter by mouse dragging.

Ex) Original view -> Split view



```
    83
0_03E4 void timer() interrupt N_INT_TIMER1
    85 {
    86
0_03EC     NOP;
0_03ED     test_data_char2 ++;
    89
0_03EF     Test_count0++;
    91
    92
0_03F1     NOP;
    94
0_03F2     P0DA = Test_count0;
0_03F5     if ( Test_count0<10 )
    97     {
    98
    99         //P0DA = 0xAA;
0_03FC         NOP;
   101
0_03FD         test_bit = 1;   //bit
   103
```



```
    83
0_03E4 void timer() interrupt N_INT_TIMER1
    85 {
    86
0_03EC     NOP;
0_03ED     test_data_char2 ++;
    89
0_03EF     Test_count0++;
    91
    92
```

```
   441     while(1)
   442     {
   443 //#ifdef    TEST_INC
   444 #if 1
0_0178         test_data_char ++;
0_017A         test_data_char3 ++;
0_017C         test_data_short ++;
0_0184         test_xdata ++;
0_018A         test_data_long ++;
0_01A3         test_xdata2 ++;
```

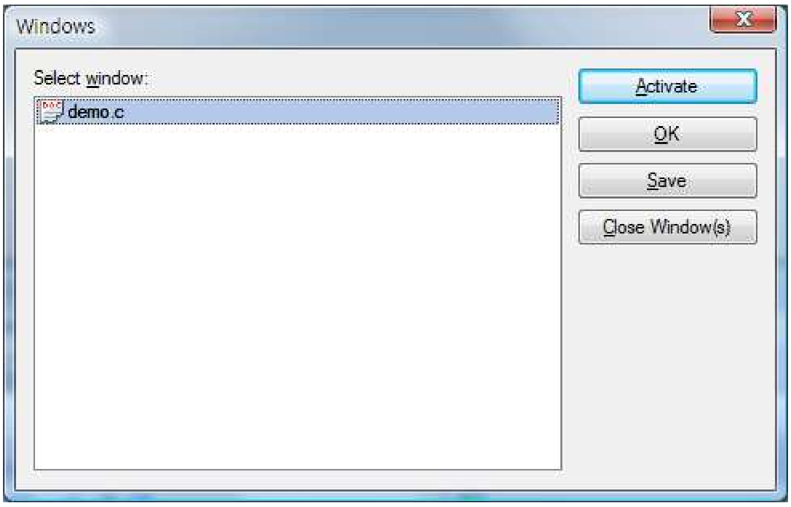
Windows number and file name

Debugger assigned serial number 1, 2, 3, … to each text windows by opened order.

You can select opened child window by this number.

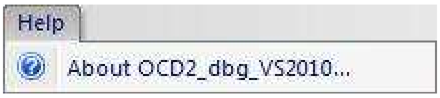
Windows

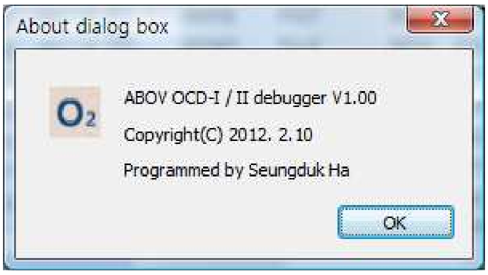
It helps to select a text window what you want at once.



3.2.7 Help

It shows debugger version only.





ABOV OCD-I / II debugger V1.00

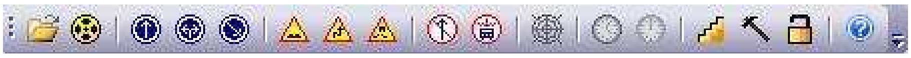
Copyright(C) 2012. 2.10

Programmed by Seungduk Ha

3.2.8 Tool bar

Tool bar provide easy, one-click access to most often used commands, which have menu buttons.

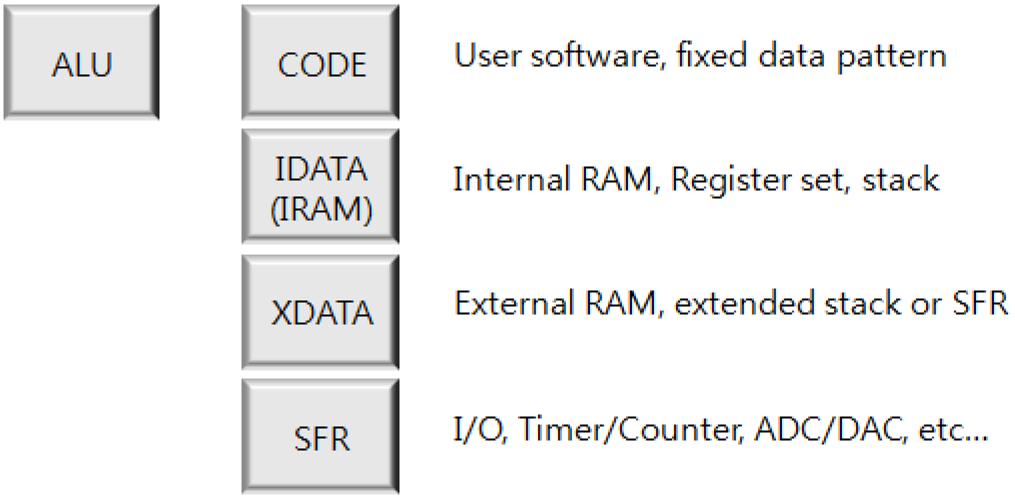
There are variety buttons that controls hex file download, Emulation, device configurations, etc.



3.3 Child windows

MCS51 CPU architecture is constructed like following picture.

Each child windows prepared editing function and display its data.


ALU
CODE
User software, fixed data pattern
IDATA (IRAM)
Internal RAM, Register set, stack
XDATA
External RAM, extended stack or SFR
SFR
I/O, Timer/Counter, ADC/DAC, etc...

Debugger shows all of target device's internal data and status powerfully.

Debugger can show dumped format, disassembled format, display various information what you want to see.

This is very helpful to debug your application code programming.

3.3.1 Child window alignment

Re-size, Move, Docking, Hide, etc.

All windows support docking feature, except text window.

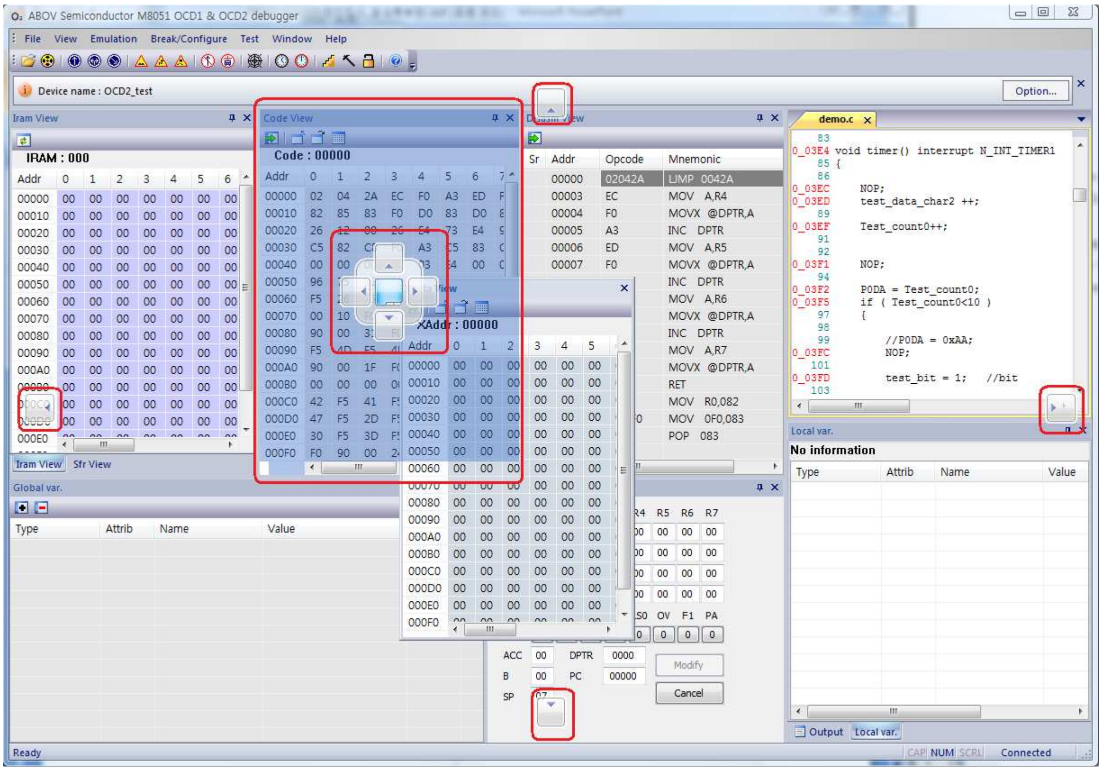
Docking means that the moving window will be placed each window's border or move into the other window, etc.

Ex) Moving a child docking view in a debugger.

If you move a child window, you can see like this following example window.

Red colored shape is slightly different to each application look.

- Place the mouse pointer on the border of the selected window (the mouse pointer will change to the drag shape when placed over the window border).
- Hold down the left mouse button while dragging the window to its new location.
- Release the mouse button.

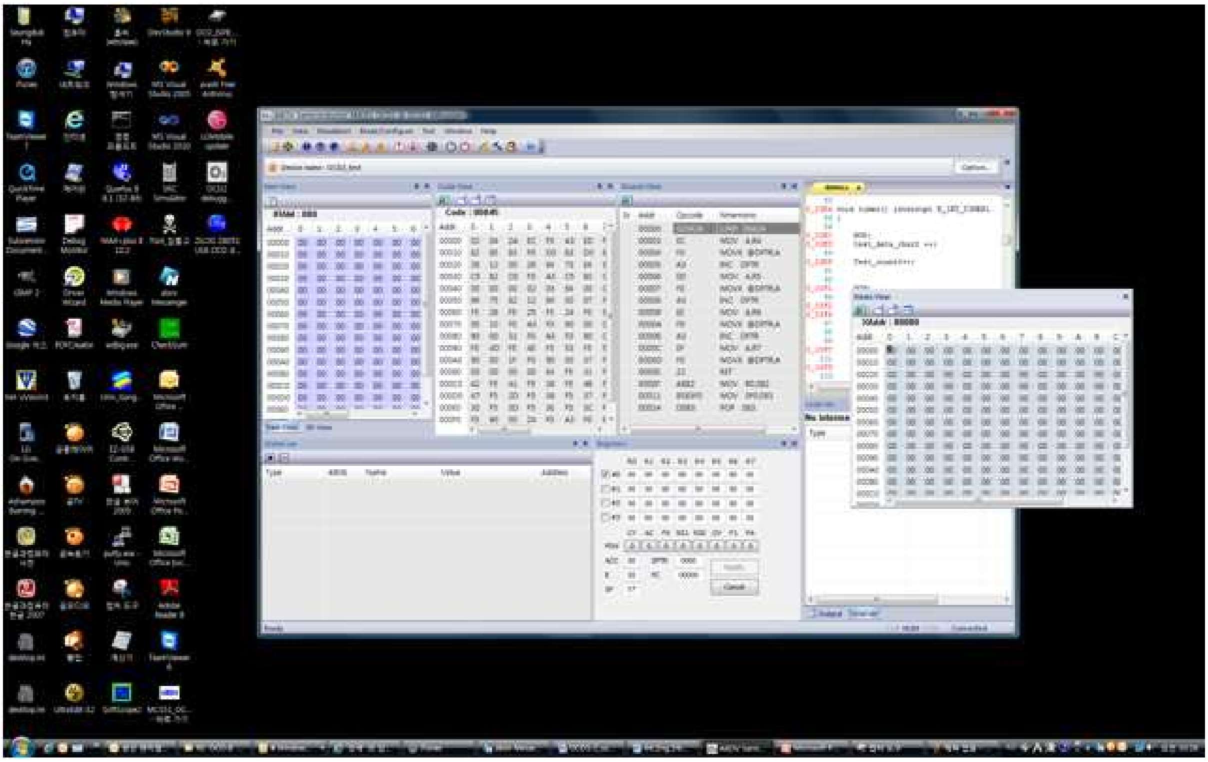


Sr	Addr	Opcode	Mnemonic
	00000	02042A	LJMP 0042A
	00003	EC	MOV A,R4
	00004	F0	MOVX @DPTR,A
	00005	A3	INC DPTR
	00006	ED	MOV A,R5
	00007	F0	MOVX @DPTR,A
			INC DPTR
			MOV A,R6
			MOVX @DPTR,A
			INC DPTR
			MOV A,R7
			MOVX @DPTR,A
			RET
			MOV R0,082
			MOV 0F0,083
			POP 083

```
     83
0_03E4 void timer() interrupt N_INT_TIMER1
     85 {
     86
0_03EC     NOP;
0_03ED     test_data_char2 ++;
     89
0_03EF     Test_count0++;
     91
     92
0_03F1     NOP;
     94
0_03F2     P0DA = Test_count0;
0_03F5     if ( Test_count0<10 )
     97     {
     98
     99         //P0DA = 0xAA;
0_03FC         NOP;
    101
0_03FD         test_bit = 1;   //bit
    103
```

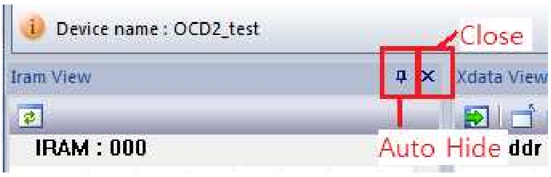
All dock-able windows can move to out of debugger frame window like following example.

Ex) Xdata window is moved out of debugger frame.



All dock-able windows can hide and close buttons.
Hide button works that child window moved each side of debugger frame but not closed.
Close button works that closing the child window.

Ex) Hide, Close button



Device name : OCD2_test

3.3.2 CODE dump View

It shows target device's code memory with hex dumped format.

Display unit is a page (hexadecimal address 0xXX00 ~ 0xXXFF).

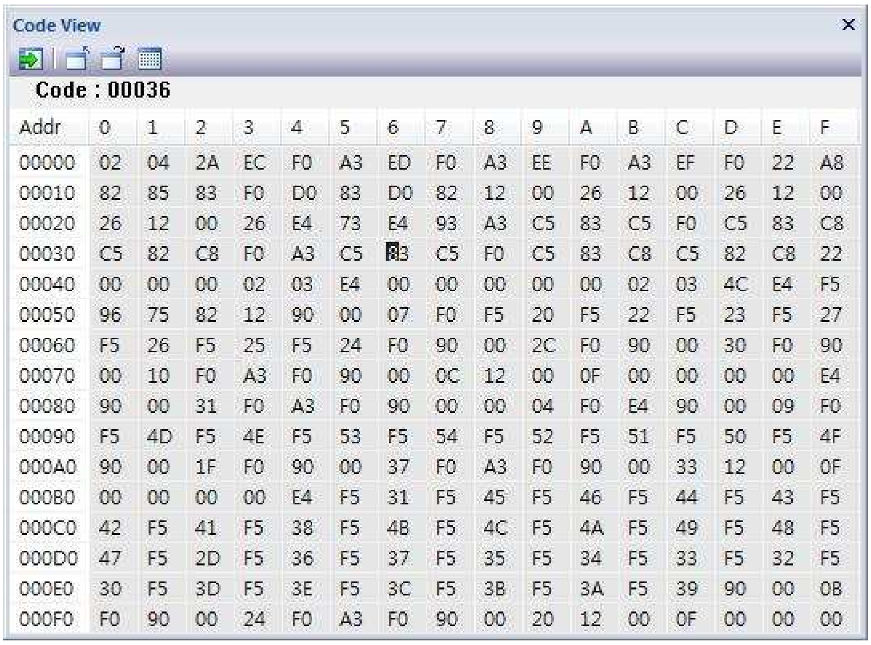
You can use page up or page down keys to move display address by page unit.

Upper side of this wildow displays address of current caret position

Edit

You can edit its data here by key typing directly.

Even if you typed in and changed data, it will not be transmitted to target device until you press "Enter" key.



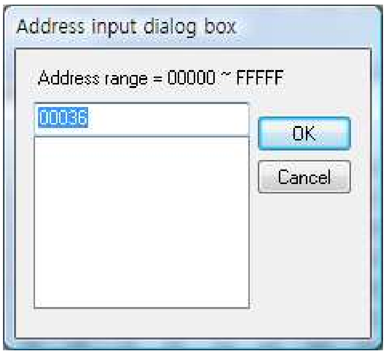
Code : 00036

Addr	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000	02	04	2A	EC	F0	A3	ED	F0	A3	EE	F0	A3	EF	F0	22	A8
00010	82	85	83	F0	D0	83	D0	82	12	00	26	12	00	26	12	00
00020	26	12	00	26	E4	73	E4	93	A3	C5	83	C5	F0	C5	83	C8
00030	C5	82	C8	F0	A3	C5	83	C5	F0	C5	83	C8	C5	82	C8	22
00040	00	00	00	02	03	E4	00	00	00	00	00	02	03	4C	E4	F5
00050	96	75	82	12	90	00	07	F0	F5	20	F5	22	F5	23	F5	27
00060	F5	26	F5	25	F5	24	F0	90	00	2C	F0	90	00	30	F0	90
00070	00	10	F0	A3	F0	90	00	0C	12	00	0F	00	00	00	00	E4
00080	90	00	31	F0	A3	F0	90	00	00	04	F0	E4	90	00	09	F0
00090	F5	4D	F5	4E	F5	53	F5	54	F5	52	F5	51	F5	50	F5	4F
000A0	90	00	1F	F0	90	00	37	F0	A3	F0	90	00	33	12	00	0F
000B0	00	00	00	00	E4	F5	31	F5	45	F5	46	F5	44	F5	43	F5
000C0	42	F5	41	F5	38	F5	4B	F5	4C	F5	4A	F5	49	F5	48	F5
000D0	47	F5	2D	F5	36	F5	37	F5	35	F5	34	F5	33	F5	32	F5
000E0	30	F5	3D	F5	3E	F5	3C	F5	3B	F5	3A	F5	39	90	00	0B
000F0	F0	90	00	24	F0	A3	F0	90	00	20	12	00	0F	00	00	00

Move button

: You can move display address and caret position by key typing.

Address range is 0x00000 ~ 0xFFFFF (1MB).



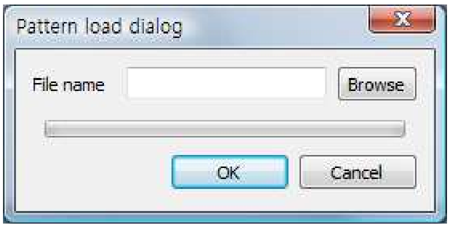
Address range = 00000 ~ FFFFF

00036

Load pattern

: You can fill the code area with hex file.

Using format is Intel-Hex format only.



Pattern load dialog

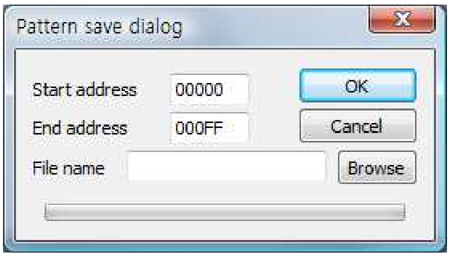
File name

Save pattern

: You can save the code area to hex file.

Using format is Intel-Hex format only.

In this time, you have to set start address and end address to save.



Pattern save dialog

Start address 00000

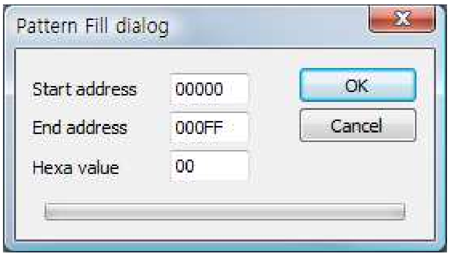
End address 000FF

File name

Fill pattern

: You can fill the code area to specified pattern.

In this time, you have to set start address and end address and data to fill.



Pattern Fill dialog

Start address 00000

End address 000FF

Hexa value 00

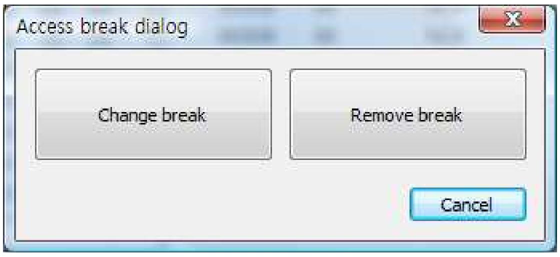
Set data break

This function works only for OCD-II devices.

If you double click mouse's left button in window, you can set or remove data break

If you set data break already, you can see a below dialog box.

You can refer "Set data break" section in this manual



If data break was set, its address is filled by BLUE color.

3.3.3 CODE disassemble View

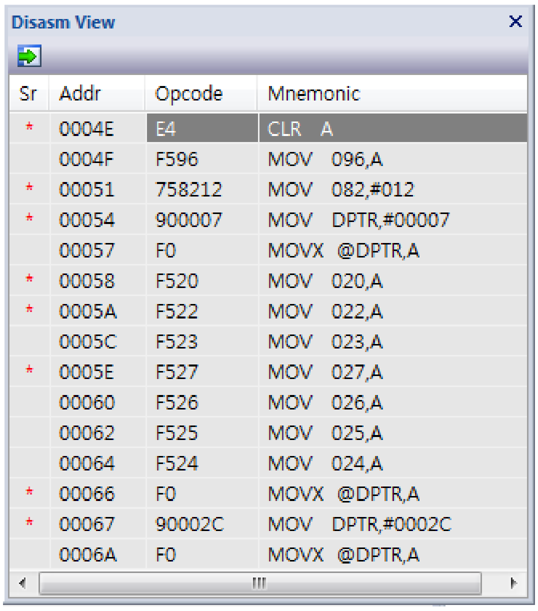
It shows target device's code memory with disassembled format.

All operands are displayed with hexadecimal number.

Dark gray colored line shows current device program counter.

Red asterisk '*' means that this line has source file information.

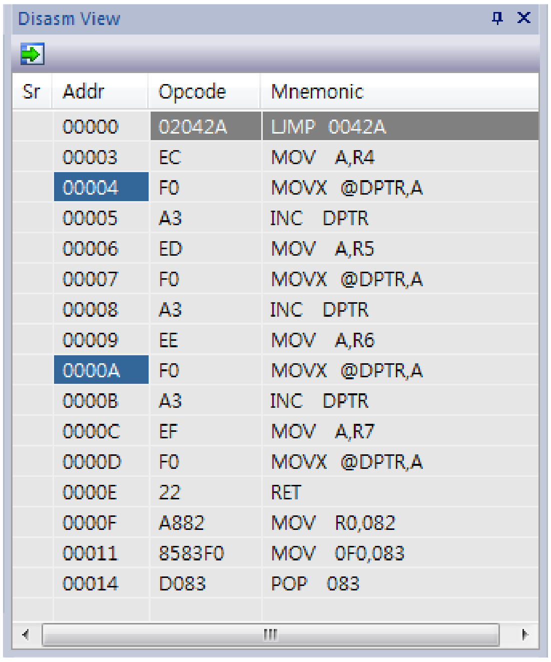
If you double click this, source file will be opened and shows with that address.



Sr	Addr	Opcode	Mnemonic
*	0004E	E4	CLR A
	0004F	F596	MOV 096,A
*	00051	758212	MOV 082,#012
*	00054	900007	MOV DPTR,#00007
	00057	F0	MOVX @DPTR,A
*	00058	F520	MOV 020,A
*	0005A	F522	MOV 022,A
	0005C	F523	MOV 023,A
*	0005E	F527	MOV 027,A
	00060	F526	MOV 026,A
	00062	F525	MOV 025,A
	00064	F524	MOV 024,A
*	00066	F0	MOVX @DPTR,A
*	00067	90002C	MOV DPTR,#0002C
	0006A	F0	MOVX @DPTR,A

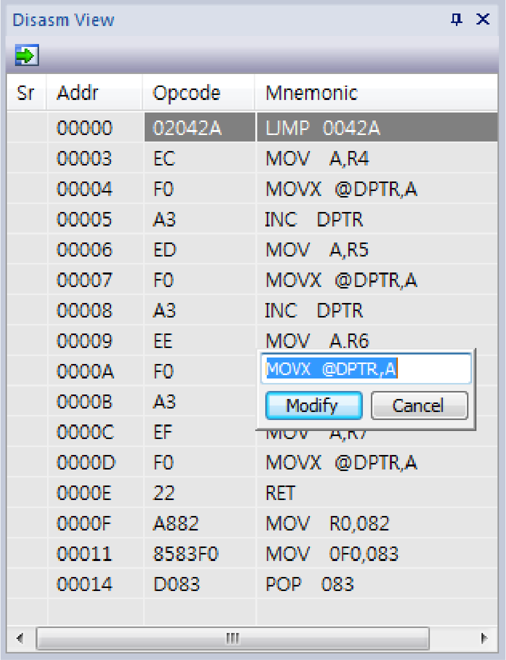
If you double click address area of each line, PC break will be toggled.

Break line is displayed BLUE colored box.



Sr	Addr	Opcode	Mnemonic
	00000	02042A	LJMP 0042A
	00003	EC	MOV A,R4
	00004	F0	MOVX @DPTR,A
	00005	A3	INC DPTR
	00006	ED	MOV A,R5
	00007	F0	MOVX @DPTR,A
	00008	A3	INC DPTR
	00009	EE	MOV A,R6
	0000A	F0	MOVX @DPTR,A
	0000B	A3	INC DPTR
	0000C	EF	MOV A,R7
	0000D	F0	MOVX @DPTR,A
	0000E	22	RET
	0000F	A882	MOV R0,082
	00011	8583F0	MOV 0F0,083
	00014	D083	POP 083

If you double click Mnemonic area of each line, you can change the data by assemble code. Change code, and then click “Modify” button.



Disasm View

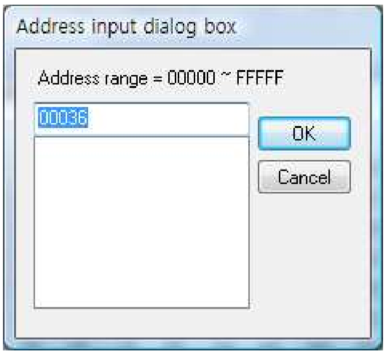
Sr	Addr	Opcode	Mnemonic
	00000	02042A	LJMP 0042A
	00003	EC	MOV A,R4
	00004	F0	MOVX @DPTR,A
	00005	A3	INC DPTR
	00006	ED	MOV A,R5
	00007	F0	MOVX @DPTR,A
	00008	A3	INC DPTR
	00009	EE	MOV A,R6
	0000A	F0	
	0000B	A3	
	0000C	EF	MOV A,R7
	0000D	F0	MOVX @DPTR,A
	0000E	22	RET
	0000F	A882	MOV R0,082
	00011	8583F0	MOV 0F0,083
	00014	D083	POP 083

MOVX @DPTR,A

Move button

: You can move display starting address and caret position by key typing.

Address range is 0x00000 ~ 0xFFFFF (1MB).



Address input dialog box

Address range = 00000 ~ FFFFF

00036

3.3.4 XDATA dump View

It shows target device's XDATA memory with hex dumped format.

Display unit is a page (hexadecimal address 0xXX00 ~ 0xXXFF).

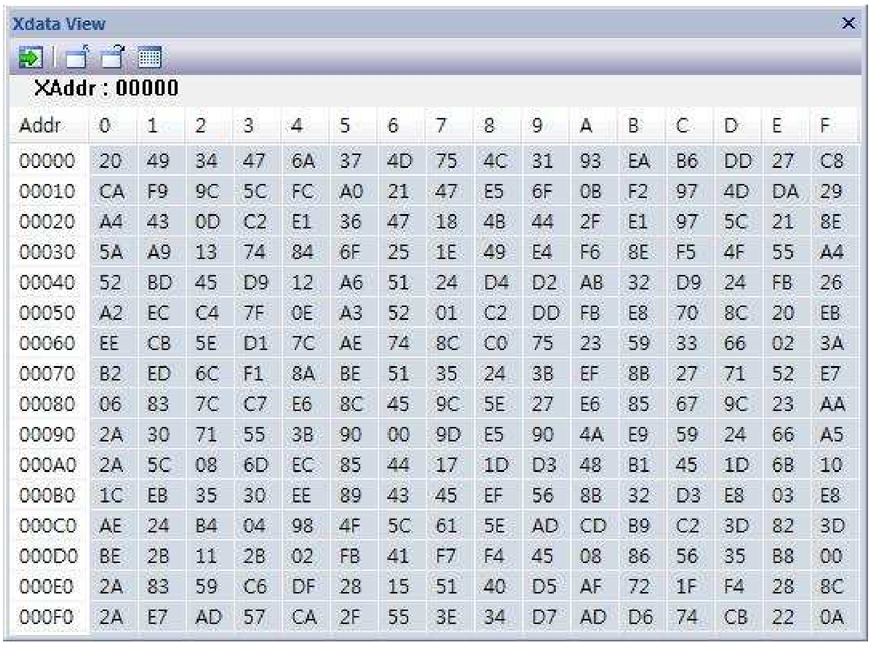
You can use page up or page down keys to move display address by page unit.

Upper side of this wildow displays address of current caret position

Edit

You can edit its data here by key typing directly.

Even if you typed in and changed data, it will not be transmitted to target device until you press "Enter" key.



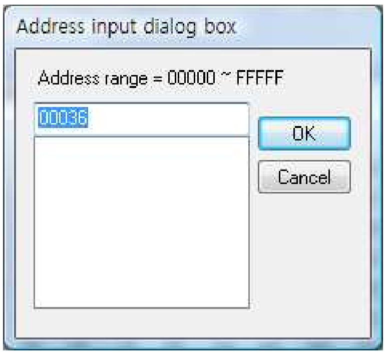
XAddr : 00000

Addr	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000	20	49	34	47	6A	37	4D	75	4C	31	93	EA	B6	DD	27	C8
00010	CA	F9	9C	5C	FC	A0	21	47	E5	6F	0B	F2	97	4D	DA	29
00020	A4	43	0D	C2	E1	36	47	18	4B	44	2F	E1	97	5C	21	8E
00030	5A	A9	13	74	84	6F	25	1E	49	E4	F6	8E	F5	4F	55	A4
00040	52	BD	45	D9	12	A6	51	24	D4	D2	AB	32	D9	24	FB	26
00050	A2	EC	C4	7F	0E	A3	52	01	C2	DD	FB	E8	70	8C	20	EB
00060	EE	CB	5E	D1	7C	AE	74	8C	C0	75	23	59	33	66	02	3A
00070	B2	ED	6C	F1	8A	BE	51	35	24	3B	EF	8B	27	71	52	E7
00080	06	83	7C	C7	E6	8C	45	9C	5E	27	E6	85	67	9C	23	AA
00090	2A	30	71	55	3B	90	00	9D	E5	90	4A	E9	59	24	66	A5
000A0	2A	5C	08	6D	EC	85	44	17	1D	D3	48	B1	45	1D	6B	10
000B0	1C	EB	35	30	EE	89	43	45	EF	56	8B	32	D3	E8	03	E8
000C0	AE	24	B4	04	98	4F	5C	61	5E	AD	CD	B9	C2	3D	82	3D
000D0	BE	2B	11	2B	02	FB	41	F7	F4	45	08	86	56	35	B8	00
000E0	2A	83	59	C6	DF	28	15	51	40	D5	AF	72	1F	F4	28	8C
000F0	2A	E7	AD	57	CA	2F	55	3E	34	D7	AD	D6	74	CB	22	0A

Move button

➡ : You can move display starting address and caret position by key typing.

Address range is 0x00000 ~ 0xFFFFF (1MB).



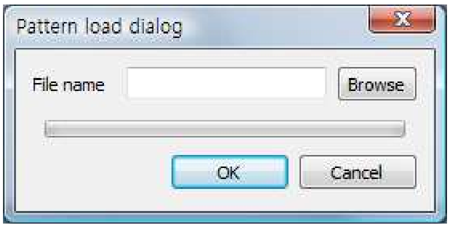
Address range = 00000 ~ FFFFF

00036

Load pattern

: You can fill the XDATA area with hex file.

Using format is Intel-Hex format only.

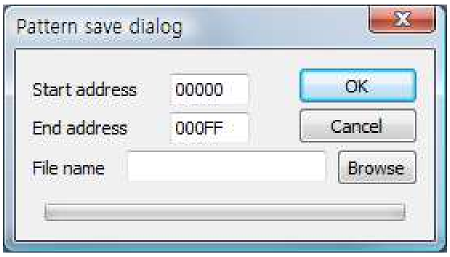


Save pattern

: You can save the XDATA area to hex file.

Using format is Intel-Hex format only.

In this time, you have to set start address and end address to save.



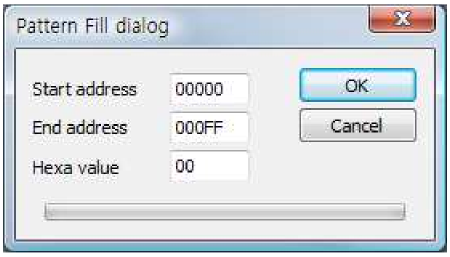
Start address 00000

End address 000FF

Fill pattern

: You can fill the XDATA area to specified pattern.

In this time, you have to set start address and end address and data to fill.



Start address 00000

End address 000FF

Hexa value 00

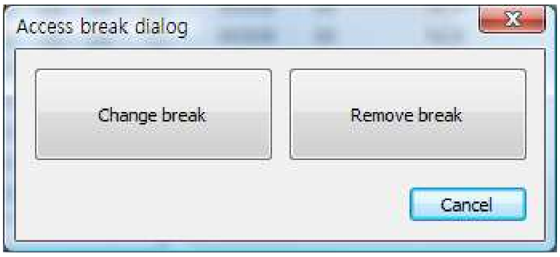
Set data break

This function works only for OCD-II devices.

If you double click mouse's left button in window, you can set or remove data break

If you set data break already, you can see a below dialog box.

You can refer "Set data break" section in this manual



If data break was set, its address is filled by BLUE color.

3.3.5 IDATA (IRAM) dump View

It shows target device's IDATA (IRAM) memory with hex dumped format.

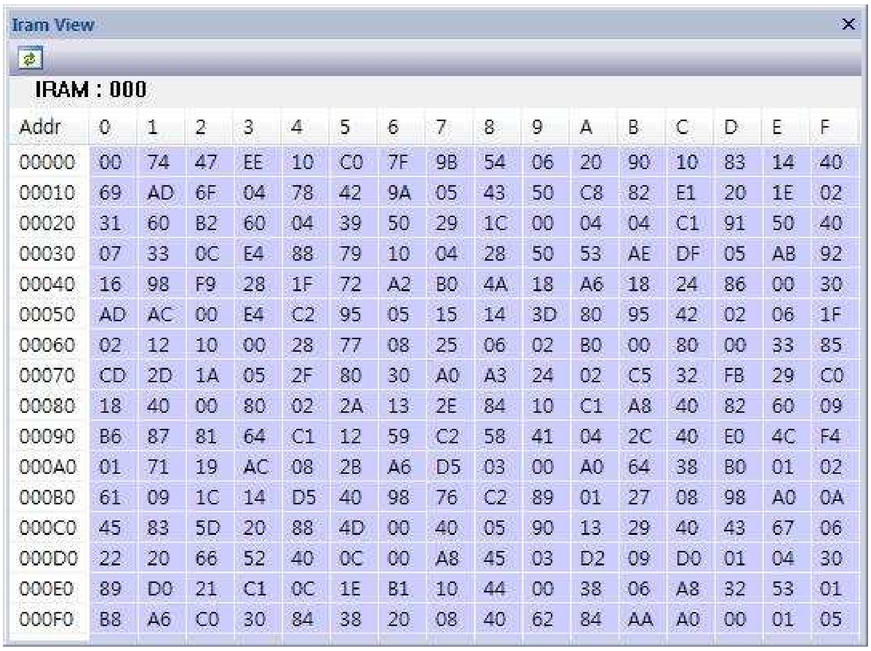
Display unit is a page (hexadecimal address 0x0000 ~ 0x00FF).

Upper side of this wildow displays address of current caret position

Edit

You can edit its data here by key typing directly.

Even if you typed in and changed data, it will not be transmitted to target device until you press "Enter" key.



IRAM : 000

```
Addr   0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00000  00 74 47 EE 10 C0 7F 9B 54 06 20 90 10 83 14 40
00010  69 AD 6F 04 78 42 9A 05 43 50 C8 82 E1 20 1E 02
00020  31 60 B2 60 04 39 50 29 1C 00 04 04 C1 91 50 40
00030  07 33 0C E4 88 79 10 04 28 50 53 AE DF 05 AB 92
00040  16 98 F9 28 1F 72 A2 B0 4A 18 A6 18 24 86 00 30
00050  AD AC 00 E4 C2 95 05 15 14 3D 80 95 42 02 06 1F
00060  02 12 10 00 28 77 08 25 06 02 B0 00 80 00 33 85
00070  CD 2D 1A 05 2F 80 30 A0 A3 24 02 C5 32 FB 29 C0
00080  18 40 00 80 02 2A 13 2E 84 10 C1 A8 40 82 60 09
00090  B6 87 81 64 C1 12 59 C2 58 41 04 2C 40 E0 4C F4
000A0  01 71 19 AC 08 2B A6 D5 03 00 A0 64 38 B0 01 02
000B0  61 09 1C 14 D5 40 98 76 C2 89 01 27 08 98 A0 0A
000C0  45 83 5D 20 88 4D 00 40 05 90 13 29 40 43 67 06
000D0  22 20 66 52 40 0C 00 A8 45 03 D2 09 D0 01 04 30
000E0  89 D0 21 C1 0C 1E B1 10 44 00 38 06 A8 32 53 01
000F0  B8 A6 C0 30 84 38 20 08 40 62 84 AA A0 00 01 05
```

Address 0x00 ~ 0x7F is direct addressing area (128bytes).

Address 0x80 ~ 0xFF is indirect addressing area (128bytes).

Refresh button

: It reloads data from target device and re-new current display.

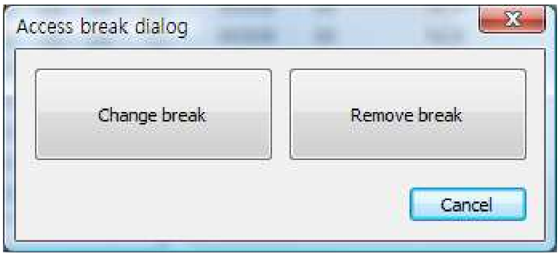
Set data break

This function works only for OCD-II devices.

If you double click mouse's left button in window, you can set or remove data break

If you set data break already, you can see a below dialog box.

You can refer "Set data break" section in this manual



If data break was set, its address is filled by BLUE color.

Even if byte (1, 2, 4) break is displayed, bit break is not displayed.

You can see bit breaks with data break dialog or global variable view.

3.3.6 SFR dump View

It shows target device's SFR (Special Function Register) with hex dumped format.

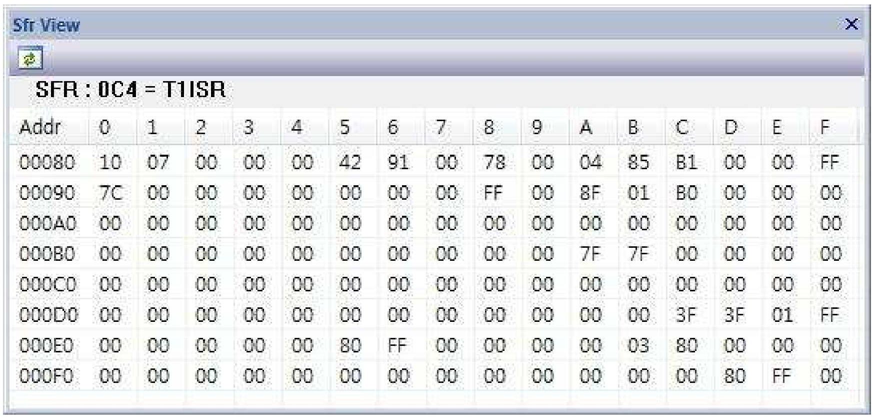
It displays 128bytes (hexadecimal address 0x0080 ~ 0x00FF) .

Upper side of this wildow displays address and SFR name of current caret position

Edit

You can edit its data here by key typing directly.

Even if you typed in and changed data, it will not be transmitted to target device until you press "Enter" key.



```
SFR : 0C4 = T1ISR

Addr   0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
00080  10 07 00 00 00 42 91 00 78 00 04 85 B1 00 00 FF
00090  7C 00 00 00 00 00 00 00 FF 00 8F 01 B0 00 00 00
000A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000B0  00 00 00 00 00 00 00 00 00 00 7F 7F 00 00 00 00
000C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000D0  00 00 00 00 00 00 00 00 00 00 00 00 3F 3F 01 FF
000E0  00 00 00 00 00 80 FF 00 00 00 00 03 80 00 00 00
000F0  00 00 00 00 00 00 00 00 00 00 00 00 00 80 FF 00
```

Address 0x80 ~ 0xFF is direct addressing area.

Refresh button

: It reloads data from target device and re-new current display.

SFR is constructed with register, timer/counter, UART, I/O port, etc.

It means that SFR value is not fixed whenever.

Using this button, you can see timer counting up or I/O port value changing, etc.

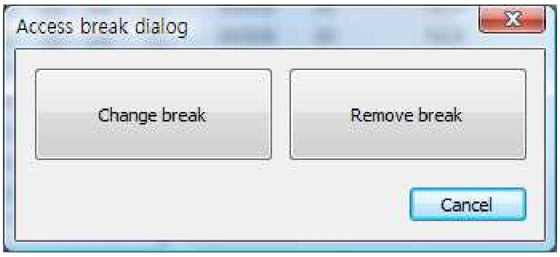
Set data break

This function works only for OCD-II devices.

If you double click mouse's left button in window, you can set or remove data break

If you set data break already, you can see a below dialog box.

You can refer "Set data break" section in this manual



If data break was set, its address is filled by BLUE color.

Even if byte (1, 2, 4) break is displayed, bit break is not displayed.

You can see bit breaks with data break dialog or global variable view.

3.3.7 Registers View

It shows target device's Registers with hex dumped format.

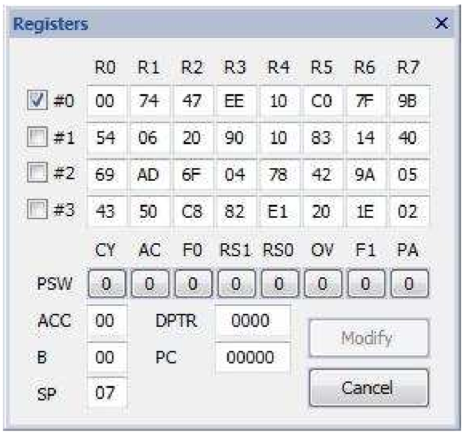
PSW (Program Status Word) is using binary format.

Edit

You can edit its data here by key typing directly.

"Modify" button will be enabled whenever you changed data.

Even if you typed in and changed data, it will not be transmitted to target device until you click "Modify" button.



	R0	R1	R2	R3	R4	R5	R6	R7
☑ #0	00	74	47	EE	10	C0	7F	9B
□ #1	54	06	20	90	10	83	14	40
□ #2	69	AD	6F	04	78	42	9A	05
□ #3	43	50	C8	82	E1	20	1E	02

	CY	AC	F0	RS1	RS0	OV	F1	PA
PSW	0	0	0	0	0	0	0	0

ACC 00 DPTR 0000

B 00 PC 00000

SP 07

3.3.8 Output View

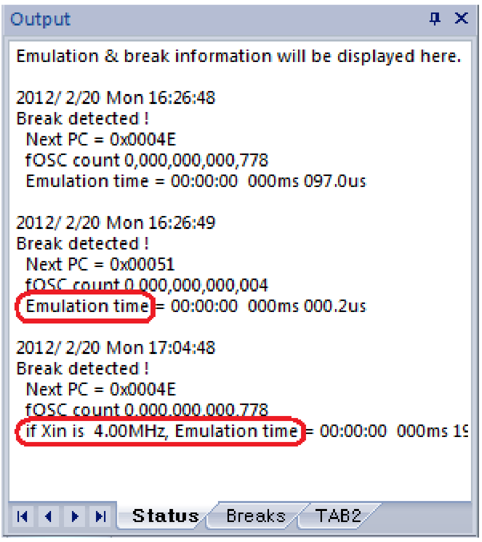
Output View is constructed with TAB window.

Each TAB window shows different information.

Status TAB window

It shows break occurrence status.

- Current time
- Next program counter
- Target device’s clock count.
- Emulation time
  - If you use OCD-II device and connected RTIME, it shows real emulation.
    It will be displayed as “Emulation time =”
  - If you use OCD-I device or OCD-II device without RTIME connection, then,FAKE clock input is used to calculate emulation time.
    Message out is “If Xin is x.xxMHz, Emulation time = “.

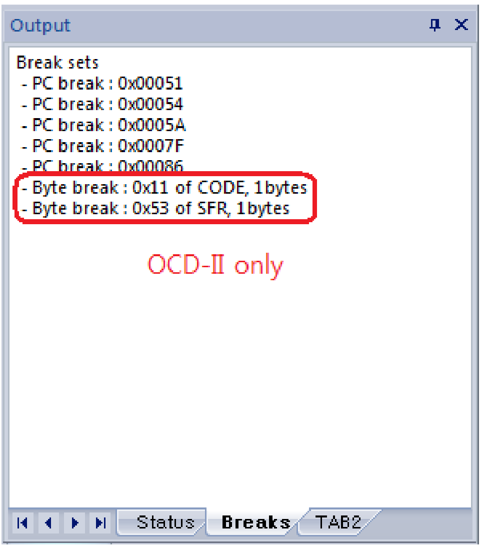


```
Emulation & break information will be displayed here.

2012/ 2/20 Mon 16:26:48
Break detected !
  Next PC = 0x0004E
  fOSC count 0,000,000,000,778
  Emulation time = 00:00:00  000ms 097.0us

2012/ 2/20 Mon 16:26:49
Break detected !
  Next PC = 0x00051
  fOSC count 0,000,000,000,004
  Emulation time = 00:00:00  000ms 000.2us

2012/ 2/20 Mon 17:04:48
Break detected !
  Next PC = 0x0004E
  fOSC count 0,000,000,000,778
  if Xin is  4.00MHz, Emulation time = 00:00:00  000ms 1
```

Break TAB window

It shows break setting status.

If you use OCD-II device, you can see data break sets too.



```
Break sets
 - PC break : 0x00051
 - PC break : 0x00054
 - PC break : 0x0005A
 - PC break : 0x0007F
 - PC break : 0x00086
 - Byte break : 0x11 of CODE, 1bytes
 - Byte break : 0x53 of SFR, 1bytes
```

3.3.9 Source View

It shows text file or source code file with line number.

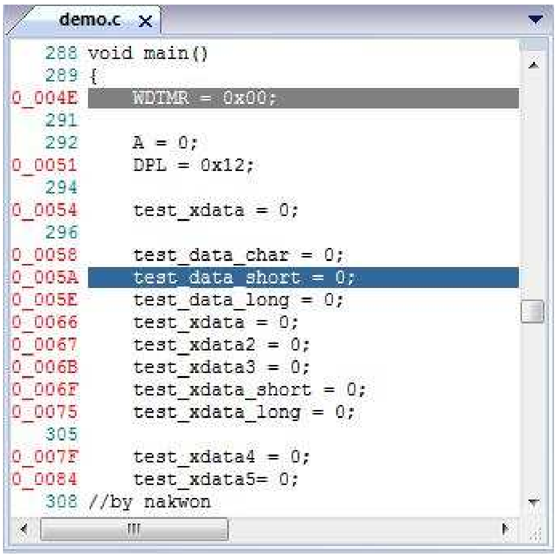
File editing is not supported.

Its displaying TAB size is fixed to 4.

If Symbol file was loaded already, source file will be displayed line number and real address like following capture.

If you mouse's left button double click at an address, disassemble view is re-new to show that address.

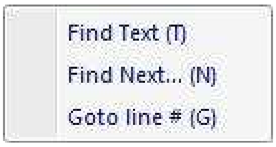
If PC break found, the line will be displayed BLUE line.



```
    288 void main()
    289 {
0_004E      WDTMR = 0x00;
    291
    292     A = 0;
0_0051      DPL = 0x12;
    294
0_0054      test_xdata = 0;
    296
0_0058      test_data_char = 0;
0_005A      test_data_short = 0;
0_005E      test_data_long = 0;
0_0066      test_xdata = 0;
0_0067      test_xdata2 = 0;
0_006B      test_xdata3 = 0;
0_006F      test_xdata_short = 0;
0_0075      test_xdata_long = 0;
    305
0_007F      test_xdata4 = 0;
0_0084      test_xdata5= 0;
    308 //by nakwon
```

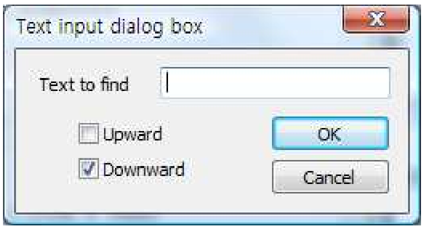
It popup sub-function dialog box when you click mouse's right button.

Shortcut key is 'T', 'N', 'G'.



Find Text (T)

Find specified text in file, and then change text color.

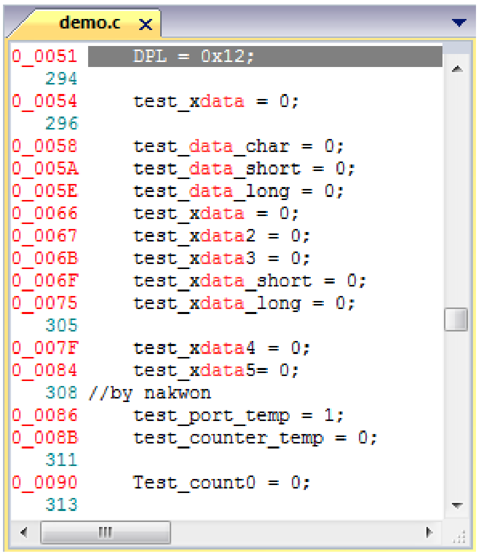


Text to find

□ Upward

☑ Downward

Ex) Find text "test" is RED colored.

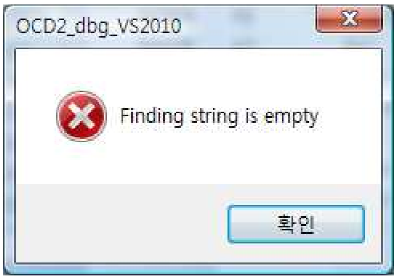


```
0_0051     DPL = 0x12;
   294
0_0054     test_xdata = 0;
   296
0_0058     test_data_char = 0;
0_005A     test_data_short = 0;
0_005E     test_data_long = 0;
0_0066     test_xdata = 0;
0_0067     test_xdata2 = 0;
0_006B     test_xdata3 = 0;
0_006F     test_xdata_short = 0;
0_0075     test_xdata_long = 0;
   305
0_007F     test_xdata4 = 0;
0_0084     test_xdata5= 0;
   308 //by nakwon
0_0086     test_port_temp = 1;
0_008B     test_counter_temp = 0;
   311
0_0090     Test_count0 = 0;
   313
```

Find Next... (N)

Find next position of the finding text to Downward (or Upward).

If debugger could not find the text, it shows following message box.



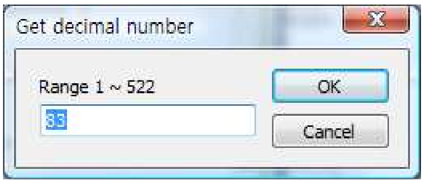
Finding string is empty

Goto line # (G)

Move current display line

It asks decimal line number of the text file.

Ex) 522 is the last line number of this text file.



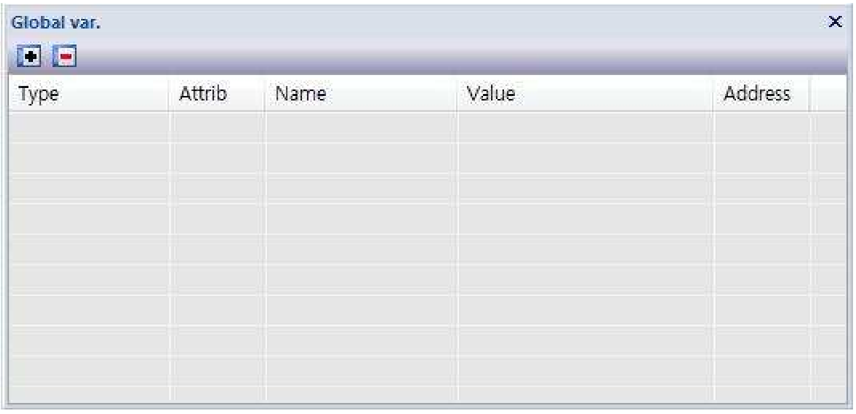
Get decimal number

Range 1 ~ 522

83

3.3.10 Global variable View

It shows and support modification for global variables of source code.



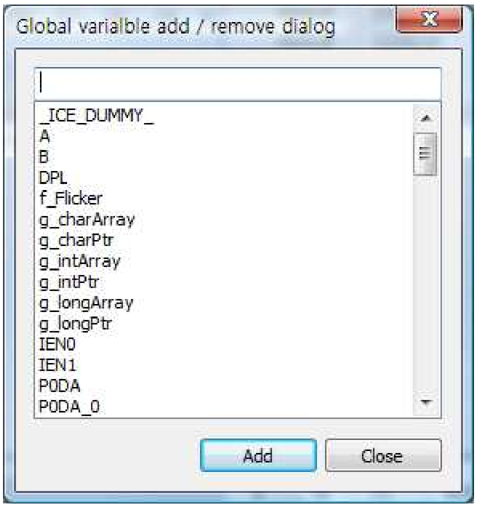
Add global variable

⊞ : You can add global symbols to this view.

This button is disabled when the symbol information is empty.

You can add global symbols by following methods

- Double click mouse's left button at a symbol name.
- Select variable and click "Add" button.
- Type in the symbol name and click "Add" button.



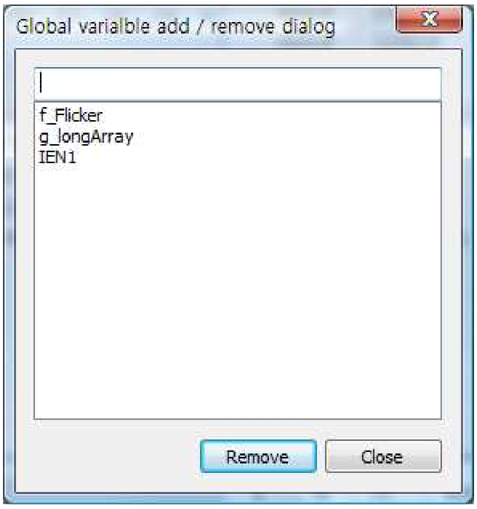
Remove global variable

⊟ : You can remove global symbols from this view.

This button is disabled when the symbol information is empty.

You can remove global symbols by following methods

- Double click mouse's left button at a symbol name.
- Select variable and click "Remove" button.
- Type in the symbol name and click "Remove" button.

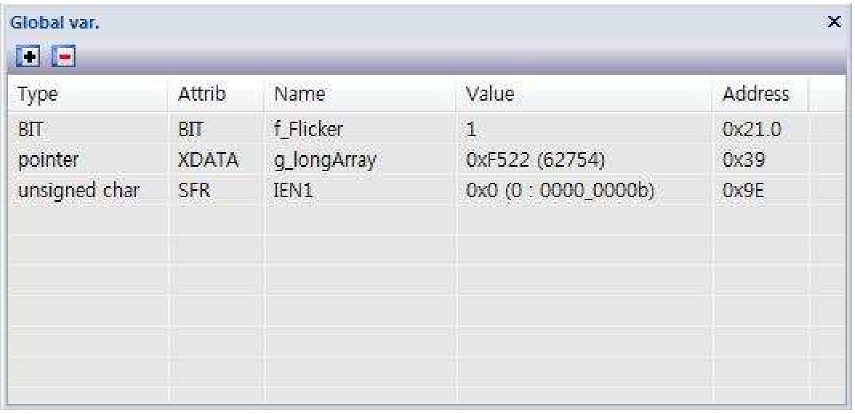


f_Flicker
g_longArray
IEN1

Display values

It uses hexadecimal , decimal , binary number.

- Bit variable : Use 0, 1 only.
- 1byte variable : Use hexadecimal, decimal, binary number.
- 2 or 4byte variable : Use hexadecimal, decimal number.
- Arrary / pointer / structure : Use 2byte value only.
  This is pointer value not data.
  Array data is not supported yet.



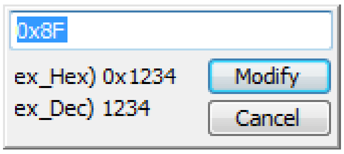
Type	Attrib	Name	Value	Address
BIT	BIT	f_Flicker	1	0x21.0
pointer	XDATA	g_longArray	0xF522 (62754)	0x39
unsigned char	SFR	IEN1	0x0 (0 : 0000_0000b)	0x9E

Edit

Move mouse pointer to the data area where you want to edit.

Double click mouse's left button.

Then, you can see a following dialog box.


0x8F

ex_Hex) 0x1234

ex_Dec) 1234

Using data format

- Binary : bit variable only. Input value is 0 or 1.
- Decimal : decimal number (ex: 12345)
- Hexadecimal : hexadecimal number (ex : 0x1234)

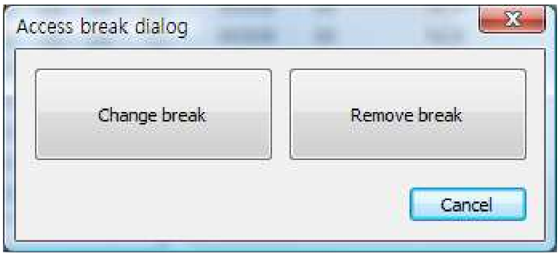
Set break

This function works only for OCD-II devices.

If you double click mouse's left button in window, you can set or remove data break

If you set data break already, you can see a below dialog box.

You can refer "Set data break" section in this manual



Real time monitoring

This function works only for OCD-II devices.

In general, developer could not see the target device's internal data.

Anyway, OCD-II interface supports data monitoring even if CPU is operating

It is possible that the real time monitoring of global variables.

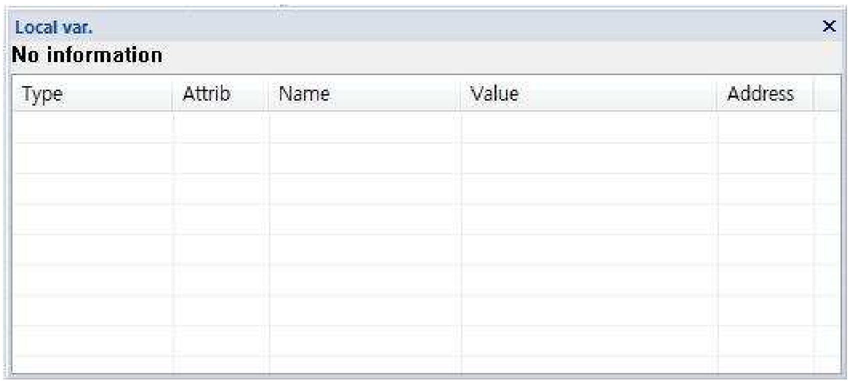
Because of, global variables occupy fixed address.

By the same reason, it is not possible that the real time monitoring of local variables.

Local variables use stack or volatile address.

3.3.11 Local variable View

It shows and support modification for local variables of source code.



No information

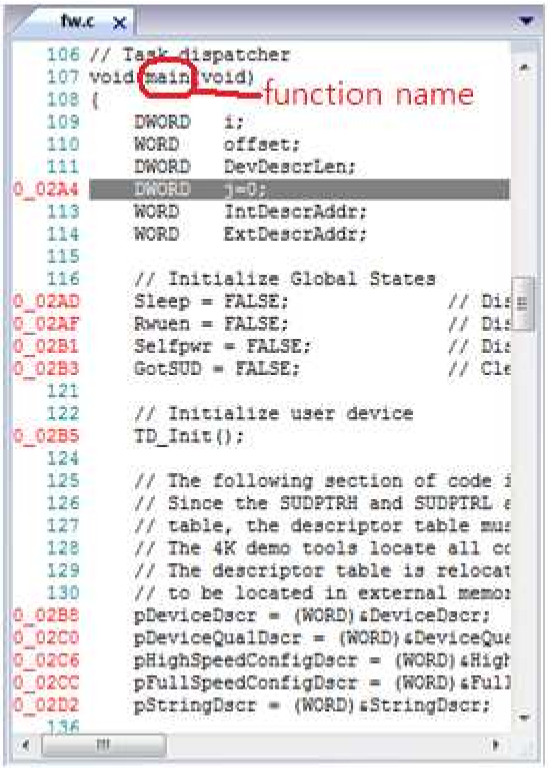
Type	Attrib	Name	Value	Address

Add / Remove local variable

Add or remove the local variables to this view is processed automatically by debugger.

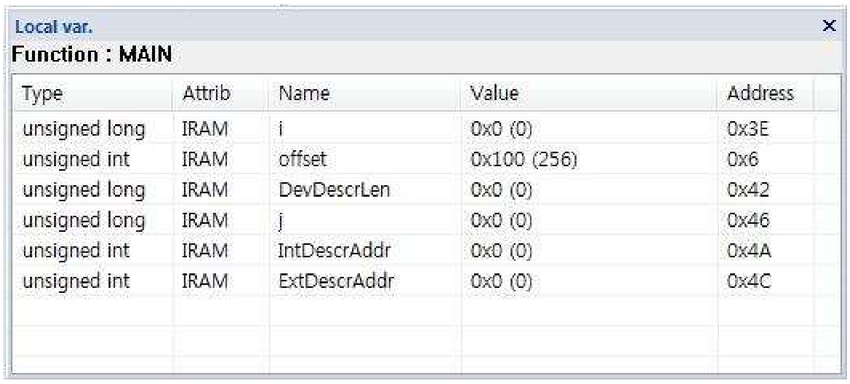
If your program is placed in a local function, then it shows local function name and its variables.

Ex) your program is placed address 0_02A4 of main(void)



```
   106 // Task dispatcher
   107 void main(void)
   108 {
   109         DWORD    i;
   110         WORD     offset;
   111         DWORD    DevDescrLen;
0_02A4         DWORD    j=0;
   113         WORD     IntDescrAddr;
   114         WORD     ExtDescrAddr;
   115
   116         // Initialize Global States
0_02AD         Sleep = FALSE;                      // Dis
0_02AF         Rwuen = FALSE;                      // Dis
0_02B1         Selfpwr = FALSE;                    // Dis
0_02B3         GotSUD = FALSE;                     // Cle
   121
   122         // Initialize user device
0_02B5         TD_Init();
   124
   125         // The following section of code i
   126         // Since the SUDPTRH and SUDPTRL a
   127         // table, the descriptor table mus
   128         // The 4K demo tools locate all co
   129         // The descriptor table is relocat
   130         // to be located in external memor
0_02B8         pDeviceDscr = (WORD)&DeviceDscr;
0_02C0         pDeviceQualDscr = (WORD)&DeviceQua
0_02C6         pHighSpeedConfigDscr = (WORD)&High
0_02CC         pFullSpeedConfigDscr = (WORD)&Full
0_02D2         pStringDscr = (WORD)&StringDscr;
   136
```

In this time, local variable view shows “Function : MAIN” and its local variables as below.



Function : MAIN

Type	Attrib	Name	Value	Address
unsigned long	IRAM	i	0x0 (0)	0x3E
unsigned int	IRAM	offset	0x100 (256)	0x6
unsigned long	IRAM	DevDescrLen	0x0 (0)	0x42
unsigned long	IRAM	j	0x0 (0)	0x46
unsigned int	IRAM	IntDescrAddr	0x0 (0)	0x4A
unsigned int	IRAM	ExtDescrAddr	0x0 (0)	0x4C

Display values

It uses hexadecimal , decimal , binary number.

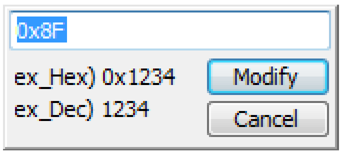
- Bit variable : Use 0, 1 only.
- 1byte variable : Use hexadecimal, decimal, binary number.
- 2 or 4byte variable : Use hexadecimal, decimal number.
- Arrary / pointer / structure : Use 2byte value only.
  This is pointer value not data.
  Array data is not supported yet.

Edit

Move mouse pointer to the data area where you want to edit.

Double click mouse’s left button.

Then, you can see a following dialog box.



0x8F

ex_Hex) 0x1234

ex_Dec) 1234

Using data format

- Binary : bit variable only. Input value is 0 or 1.
- Decimal : decimal number (ex: 12345)
- Hexadecimal : hexadecimal number (ex : 0x1234)

End of document.