



Instruction

Keil uVision3 IDE User Guide

Document No.:	INS10246
Version:	5
Description:	This document describes the steps required to use Keil uVision3 IDE as Z-Wave development environment.
Written By:	IHM;VVI;JFR;SSE
Date:	2008-07-25
Reviewed By:	CHL;SSA
Restrictions:	Partners Only

Approved by:

Date	CET	Initials	Name	Justification
2008-07-25	08:51:59	JFR	Jørgen Franck	on behalf of NTJ

This document is the property of Zensys A/S. The data contained herein, in whole or in part, may not be duplicated, used or disclosed outside the recipient for any purpose. This restriction does not limit the recipient's right to use information contained in the data if it is obtained from another source without restriction.



CONFIDENTIAL

REVISION RECORD				
Doc. Rev	Date	By	Pages affected	Brief description of changes
1	20050104	SSE JFR	All	Based on document number 900205401
1	20050414	JFR	All	Updated to Dev. Kit v4.00 with support for both ZW0102 and ZW0201 single chip
2	20060105	MVO	All	Added new 1 st page/header/footer contents and Document No.
3	20071223	IHM	All	Updated to uVision3 and Developer's Kit v5.01
4	20080331	VVI, IHM, JFR	All	Corrected minor typos
5	20080718	VVI	3-20	Sections 3.2 and 3.3 updated with the revised project settings (linker and compiler options to build the correct HEX output file); figures 5 to 9 and figure 15 have been updated accordingly.

Table of Contents

1	ABBREVIATIONS.....	1
2	INTRODUCTION.....	1
2.1	Purpose	1
2.2	Audience and prerequisites	1
2.3	Limitations	1
3	SETTING UP KEIL UVISION3.....	2
3.1	Getting started.....	2
3.2	Creating a new project	3
3.2.1	Select target device	5
3.2.2	Select project components	6
3.3	Setting the target options	10
3.3.1	Select folder for listings	13
3.3.2	Set compiler defines	14
3.3.3	Set assembler defines	17
3.3.4	Set linker defines	18
3.3.5	Set utilities	19
3.4	Building target.....	20
3.5	Uploading firmware to the Z-Wave Module.....	20
3.5.1	Set utilities	21
3.6	Creating and using multi project workspace	23
3.6.1	Creating new multi project workspace.....	23
3.6.2	Using multi project workspace.....	25
4	THE SIMULATOR.....	27
5	CREATING A NEW DEVICE IN THE DEVICE DATABASE.....	28
5.1	Adjusting the Options... ..	28
6	MODIFYING THE ZW0102 DEVICE IN THE DEVICE DATABASE	30
7	CREATING A MODEL FOR THE ZW0201/ZW0301 DEVICE	33
8	TROUBLESHOOTING.....	35
9	REFERENCES.....	36

Table of Figures

Figure 1.	Main window of Keil uVision3 IDE with default layout.....	3
Figure 2.	Create New Project dialog.....	4
Figure 3.	Select Device dialog and check required option	5
Figure 4.	Confirmation dialog for copying the Standard Startup Code.....	5
Figure 5.	Components, Environment and Books dialog with required Groups	6
Figure 6.	Required files for Application Source Code group	7
Figure 7.	Required files for Z-Wave Lib group	8
Figure 8.	Required files for Z-Wave Inc group	9
Figure 9.	Options for Target dialog with Target tab.....	10
Figure 10.	Browse for Folder dialog	11
Figure 11.	Options for Target dialog with Output tab	12
Figure 12.	Browse for Folder dialog	13
Figure 13.	Options for Target dialog with Listing tab.....	14

Figure 14. Folder Setup dialog	15
Figure 15. Options for Target dialog with C51 tab.....	16
Figure 16. Options for Target dialog with A51 tab.....	17
Figure 17. Options for Target dialog with LX51 Misc tab	18
Figure 18. Options for Target dialog with Utilities tab.....	19
Figure 19. Output that KEIL uVision3 generated during build	20
Figure 20. Flash menu.....	21
Figure 21. Z-Wave Programmer output.....	22
Figure 22. Workspace creation.....	23
Figure 23. Dialog to select the folder for new workspace.....	24
Figure 24. Create New Multi Project Workspace dialog with the required projects added	25
Figure 25. Shortcut menu to set the Active Project.....	25
Figure 26. Batch Build dialog.....	26
Figure 27. Options for Target dialog with Debug tab.....	27
Figure 28. Device selection in the Device Database dialog	30
Figure 29. Selecting ZW0102 options in the Device Database dialog	31
Figure 30. Correcting ZW0102 memory map in the Device Database dialog	32
Figure 31. Selecting database in the Device Database dialog.....	33
Figure 32. Entering option data in the Device Database dialog	34

1 ABBREVIATIONS

Abbreviation	Explanation
API	Application Programming Interface
ASIC	Application-specific integrated circuit
CD	Compact Disc
IDE	Integrated Development Environment

2 INTRODUCTION

2.1 Purpose

The purpose of this document is to describe the steps required to configure Keil uVision3 IDE for developing Z-Wave applications. Furthermore, the document describes how to download the hex file into target within the Keil uVision3 IDE.

Keil uVision3 projects created for all the embedded sample applications in Developer's Kit v4.50 and v5.01.

Notice: It is not possible to simulate any Z-Wave API calls and associated hardware.

2.2 Audience and prerequisites

The audience of this document is Z-Wave partners and Zensys. Assume that the Z-Wave partner already is familiar with the Z-Wave Developer's Kit.

2.3 Limitations

Developing Z-Wave applications in the Keil uVision3 IDE tested on only Windows XP.

3 SETTING UP KEIL uVISION3

To configure the Keil uVision3 IDE for development of Z-Wave applications requires a number of steps. For detailed descriptions of the necessary steps, refer to the following paragraphs.

3.1 Getting started

The first step is to install the Keil uVision3 IDE and the software on the Z-Wave Developer's Kit CD. Tests conducted on the following versions:

1. Keil uVision3 IDE v3.51 (Keil Development Release 01.2007)
2. Z-Wave Developer's Kit v4.0x, v4.1x, v4.2x, v4.5x and v5.0x

Notice that the directory paths etc. in the Developer's Kits may change slightly from version to version. Be also aware of that Developer's Kits use different Keil linkers e.g. v5.0x use the extended linker LX51 provided in PK51. Some versions of Keil uVision3 IDE may not contain the ZW0201 and/or ZW0301 single chips in the device database. Refer to Chapter 5 regarding how to create a Z-Wave single chip in the device database. With respect to Keil Assembler, C compiler, Linker and Librarian version refer to Chapter 9.

3.2 Creating a new project

This example shows how to create a project for the LED Dimmer application using US frequency (908,42MHz). To create new projects, select “**New...**” -> “**µVision Project...**” in the “**Project**” menu

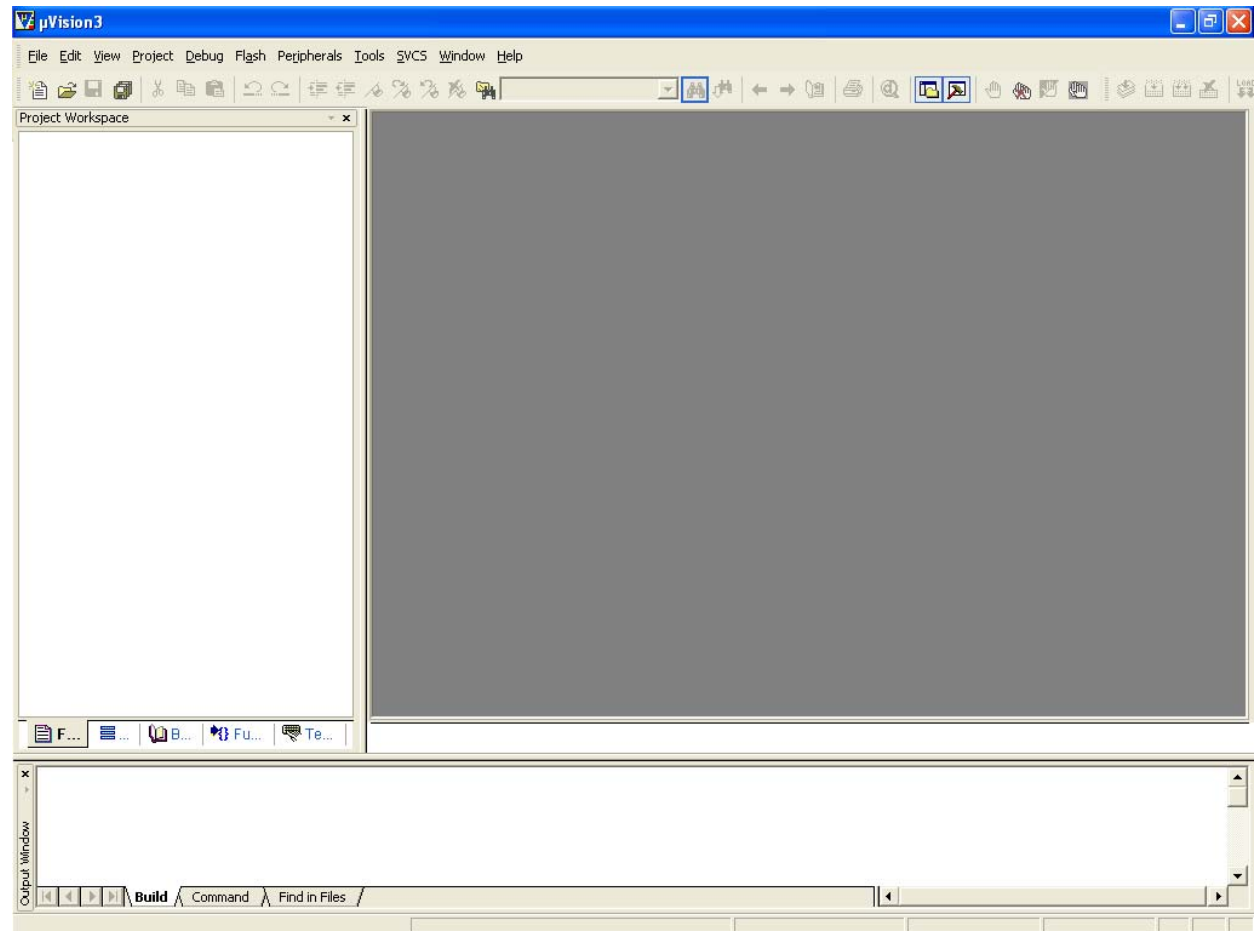


Figure 1. Main window of Keil uVision3 IDE with default layout

Select the project directory and name. In this example the project directory is C:\DevKit_5_01\Product\LED_Dimmer and the project name is MyProject.

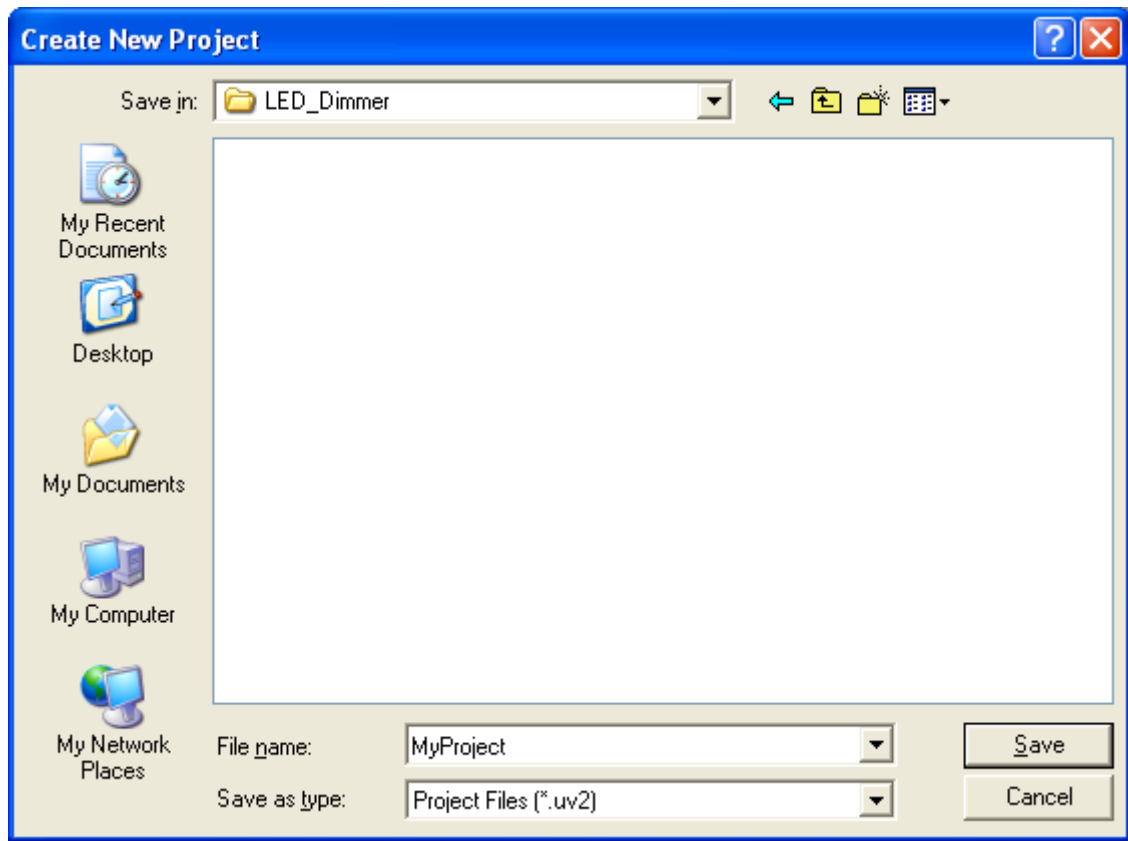


Figure 2. Create New Project dialog

3.2.1 Select target device

Now select the target device from the device database. In this case, we select ZW0201 under Zensys. The KEIL μ Vision project options with ZDK 5.01 based API libraries using extended linker LX51.

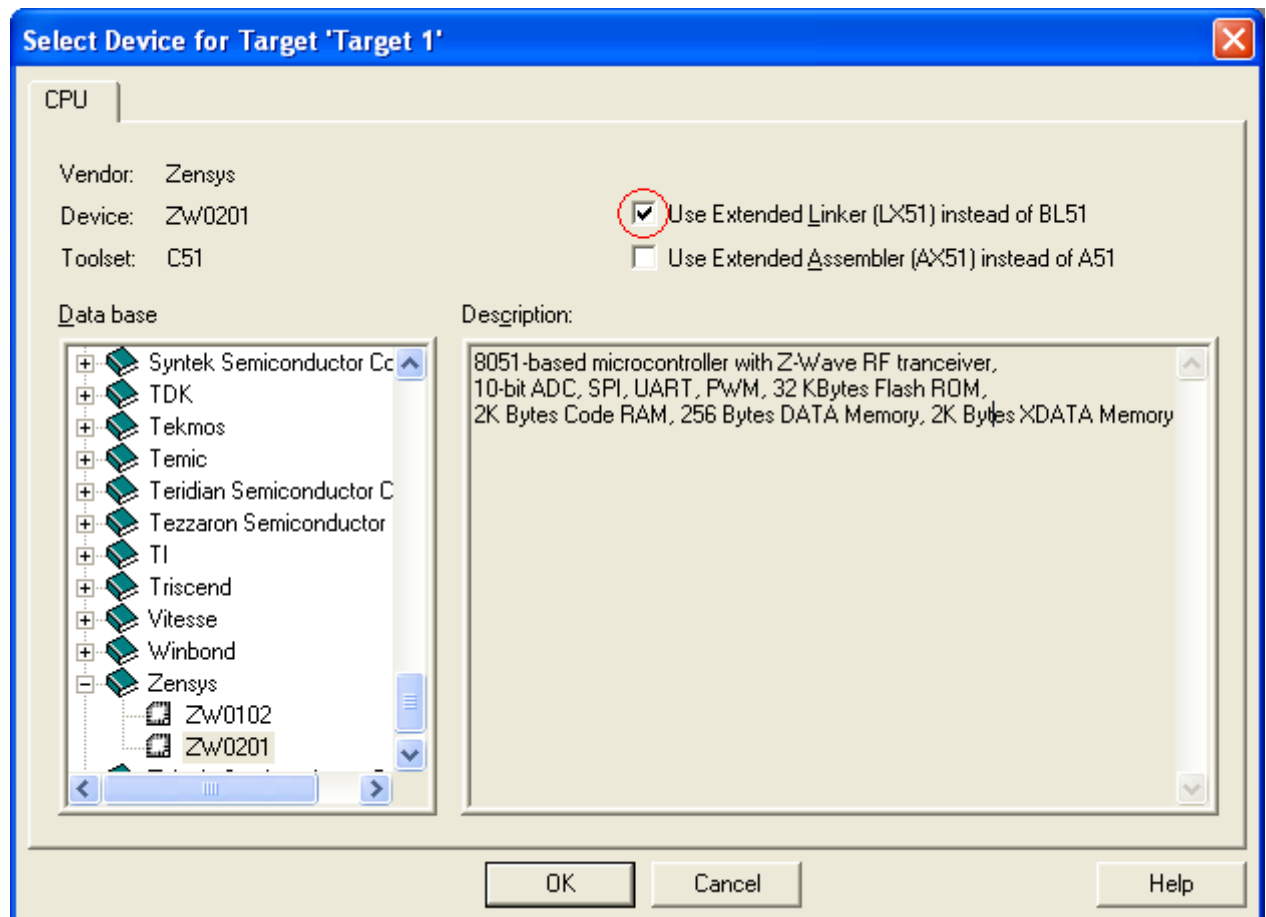


Figure 3. Select Device dialog and check required option

In case a ZW0102, ZW0201 or ZW0301 single chip is not available in the device list then read Chapters 5-7 for a description on how to create a new device. Alternatively use a newer version of the Keil μ Vision3 IDE.

In the next message box, select "No":

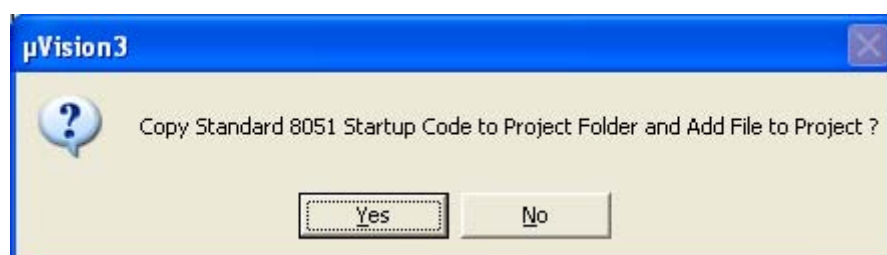


Figure 4. Confirmation dialog for copying the Standard Startup Code

3.2.2 Select project components

From the '**Project**' menu, "Manage" submenu, select the '**Component, Environment, Books...**' menu item.

Change the Project Target and the Groups as shown below:

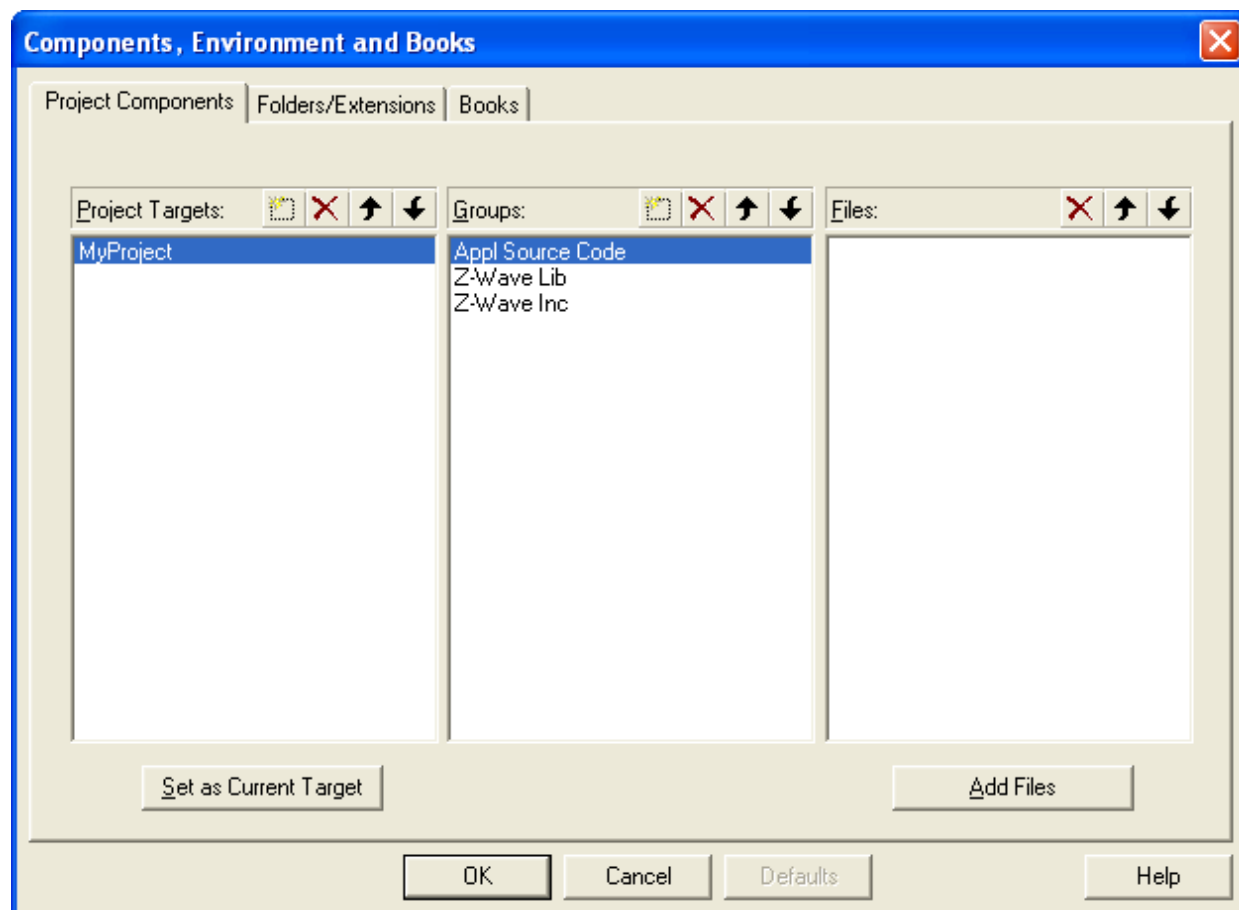


Figure 5. Components, Environment and Books dialog with required Groups

Change Project target name (double click on it) to "MyProject".

Add groups:

- Appl Source Code
- Z-Wave Lib
- Z-Wave Inc

In the "**Appl Source Code**" group (C:\DevKit_5_01\Product\LED_Dimmer) add the application source code files. Also add here the files from "C:\DevKit_5_01\Product\Util_Func" as shown below.

WARNING: Remember only to add the files from Util_Func used by the application to avoid linker errors.

The order of .c files in "Files:" list – is a order of corresponded .obj files in linker command line. Therefore, if .hex files from uVision3 should be identical to the hex files from makefiles (if used), the

order of .c files should be the same as order of corresponded .obj files in linker command line from makefiles (see .map file, which was generated by makefie). Same with .obj and .lib files in “Z-Wave Lib” group (see below). .c (.obj, .lib, etc.) files from other groups will be ordered as groups in “Groups:” list.

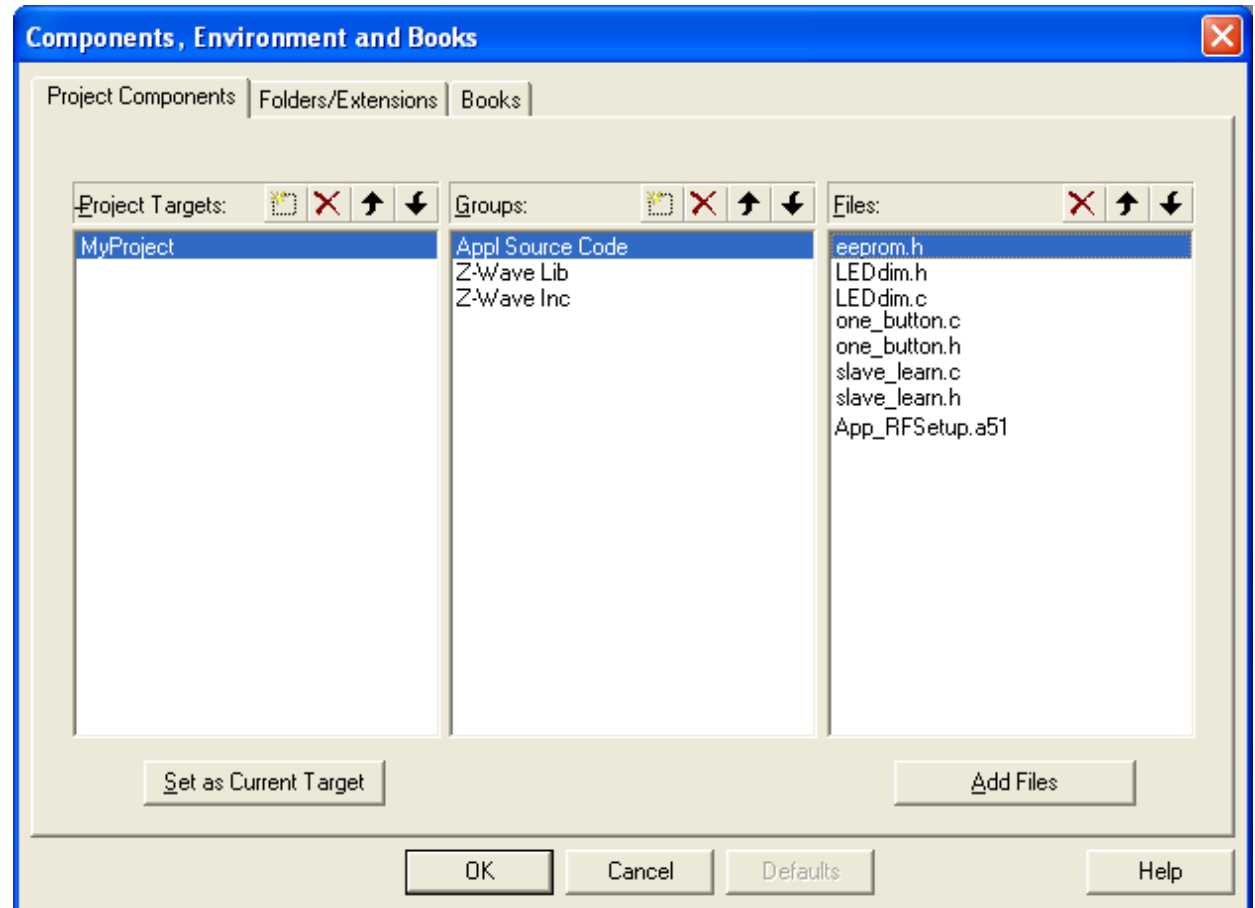


Figure 6. Required files for Application Source Code group

Take App_RFSetup.a51 from C:\DevKit_5_01\Z-Wave\IO_defines\

In the “**Z-Wave Lib**” group (C:\DevKit_5_01\Z-Wave\lib\slave_ZW020x) add the Z-Wave slave library for the ZW0201 single chip. Also, add RF .obj file (ZW_rf_020x_US.obj) for ZW0201 single chip and US country frequency from directory C:\DevKit_5_01\Z-Wave\rf_freq.

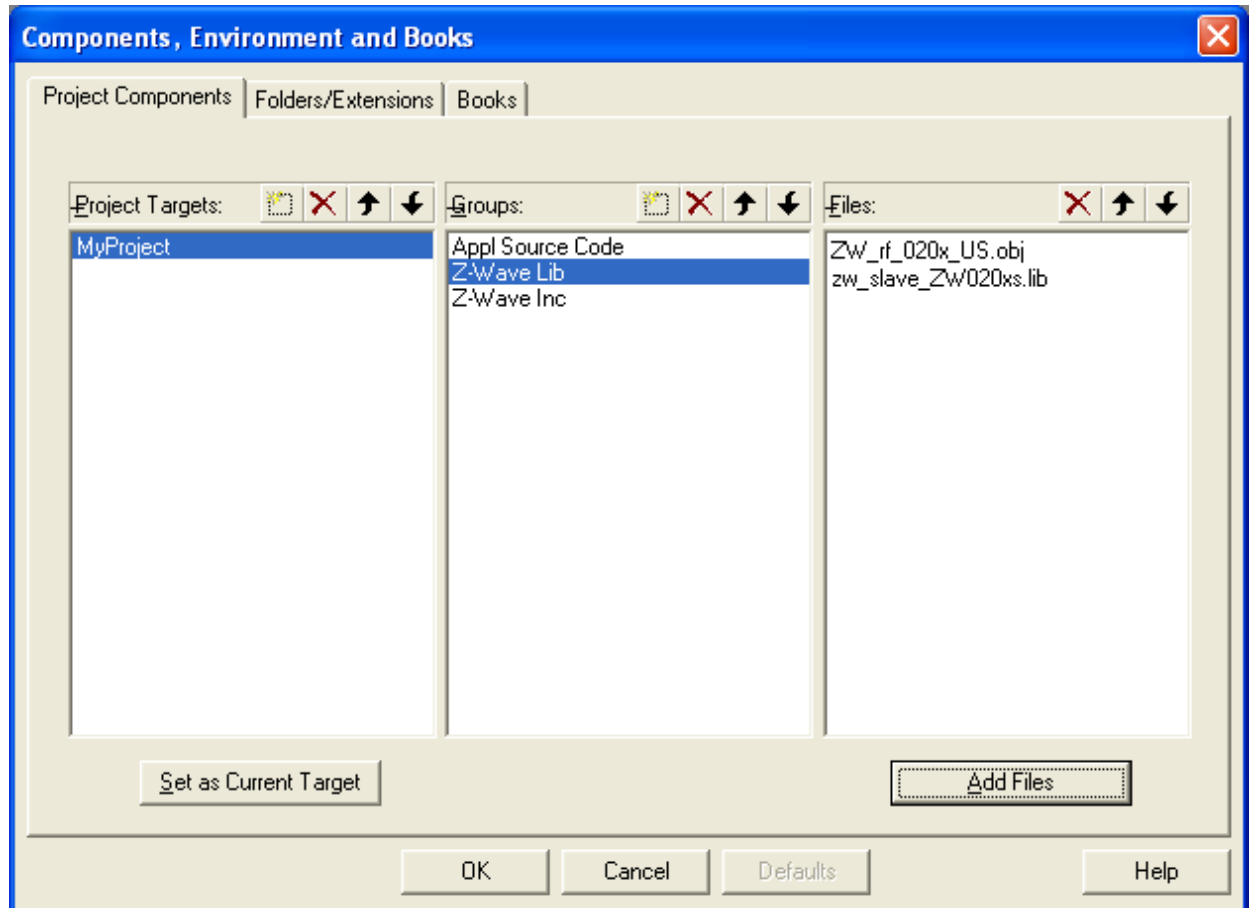


Figure 7. Required files for Z-Wave Lib group

Omit adding the RF .obj file in case of the Developer's Kit v5.00 CD or older is used because the directory "C:\DevKit_xxx\Z-Wave\rf_freq" doesn't exist.

In the “**Z-Wave Inc**” group (C:\DevKit_5_01\Z-Wave\include) add all the Z-Wave library headers files. The “**Z-Wave Inc**” group is the same, independent of the targeted sample application and Z-Wave single chip.

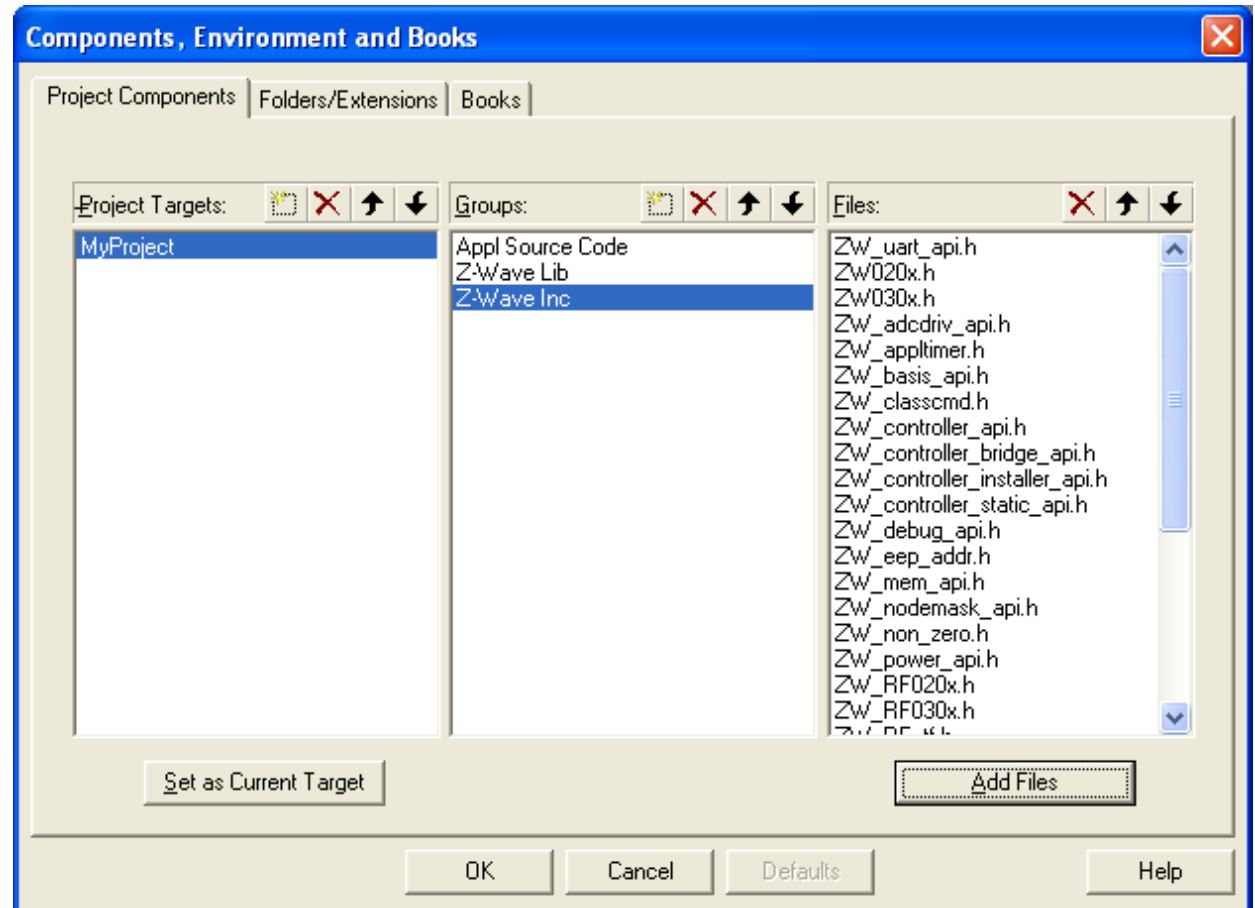


Figure 8. Required files for Z-Wave Inc group

3.3 Setting the target options

From the “**Project**” menu, select the ‘**Options for target MyProject**’.

In the “**Target**” tab select Large Memory Model and Code Rom Size.

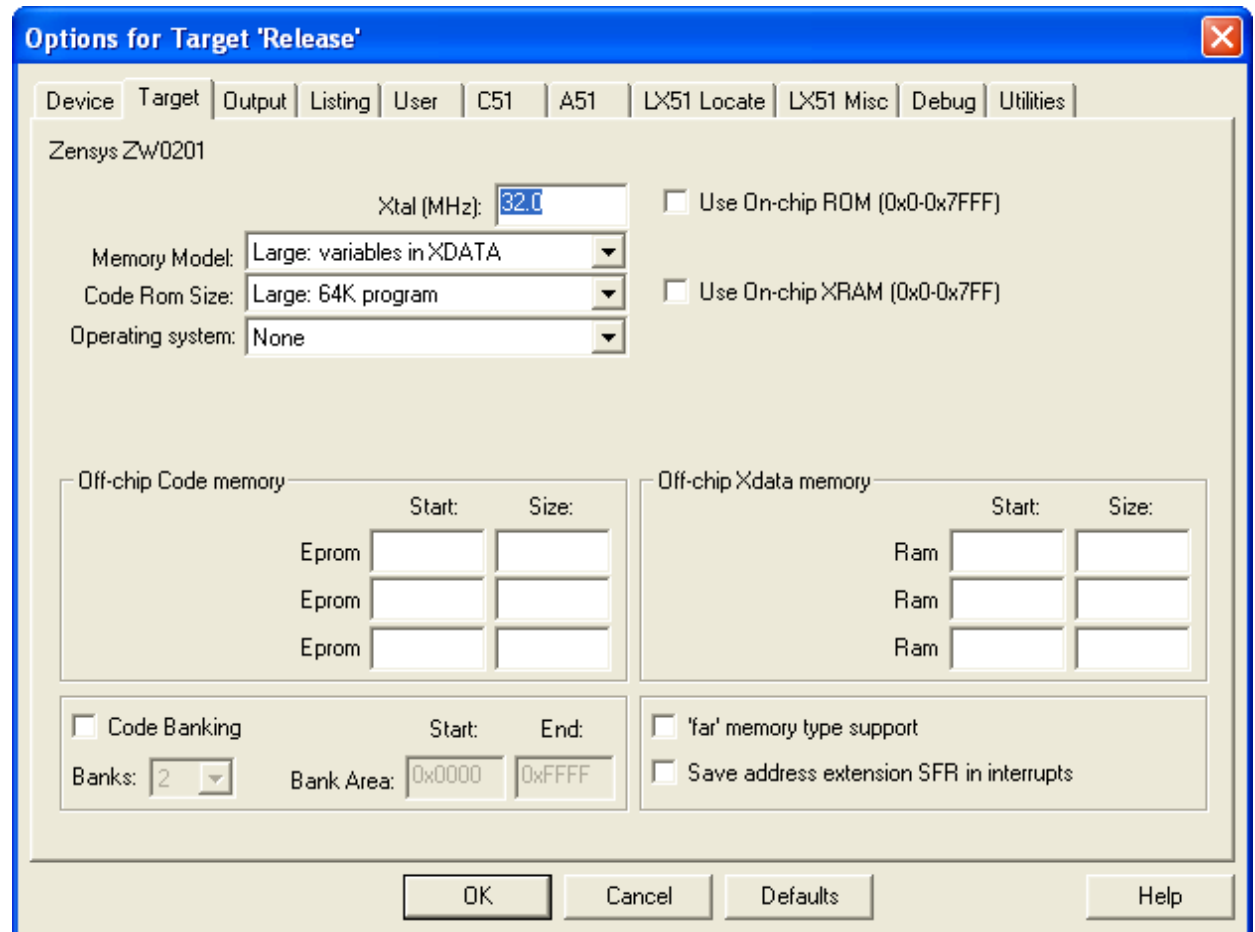


Figure 9. Options for Target dialog with Target tab

In the **"Output"** tab activate the **"Select Folder for Objects..."** button and browse to the directory **"C:\DevKit_5_01\Product\LED_Dimmer\build\"**

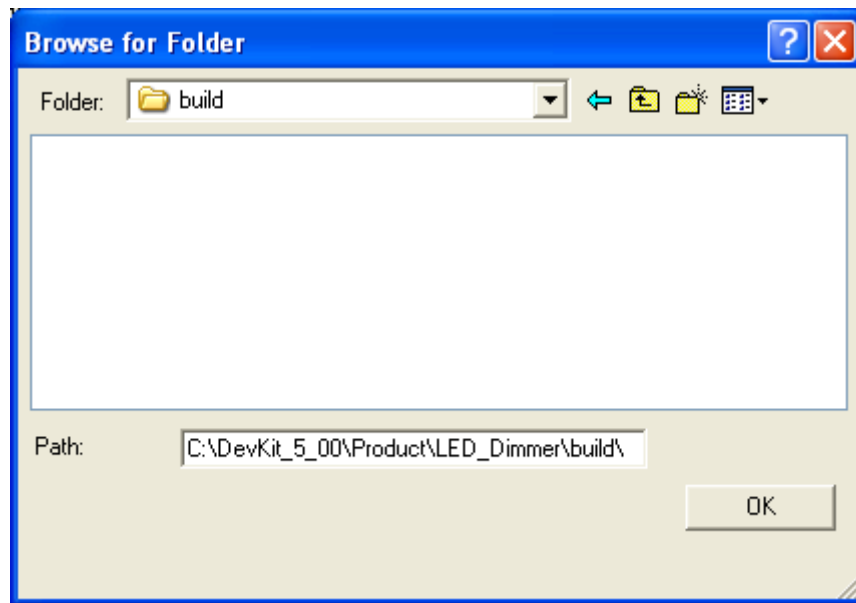


Figure 10. Browse for Folder dialog

Change the '**Name of Executable**' to MyProject_US. Set the '**Create HEX file**' check box.

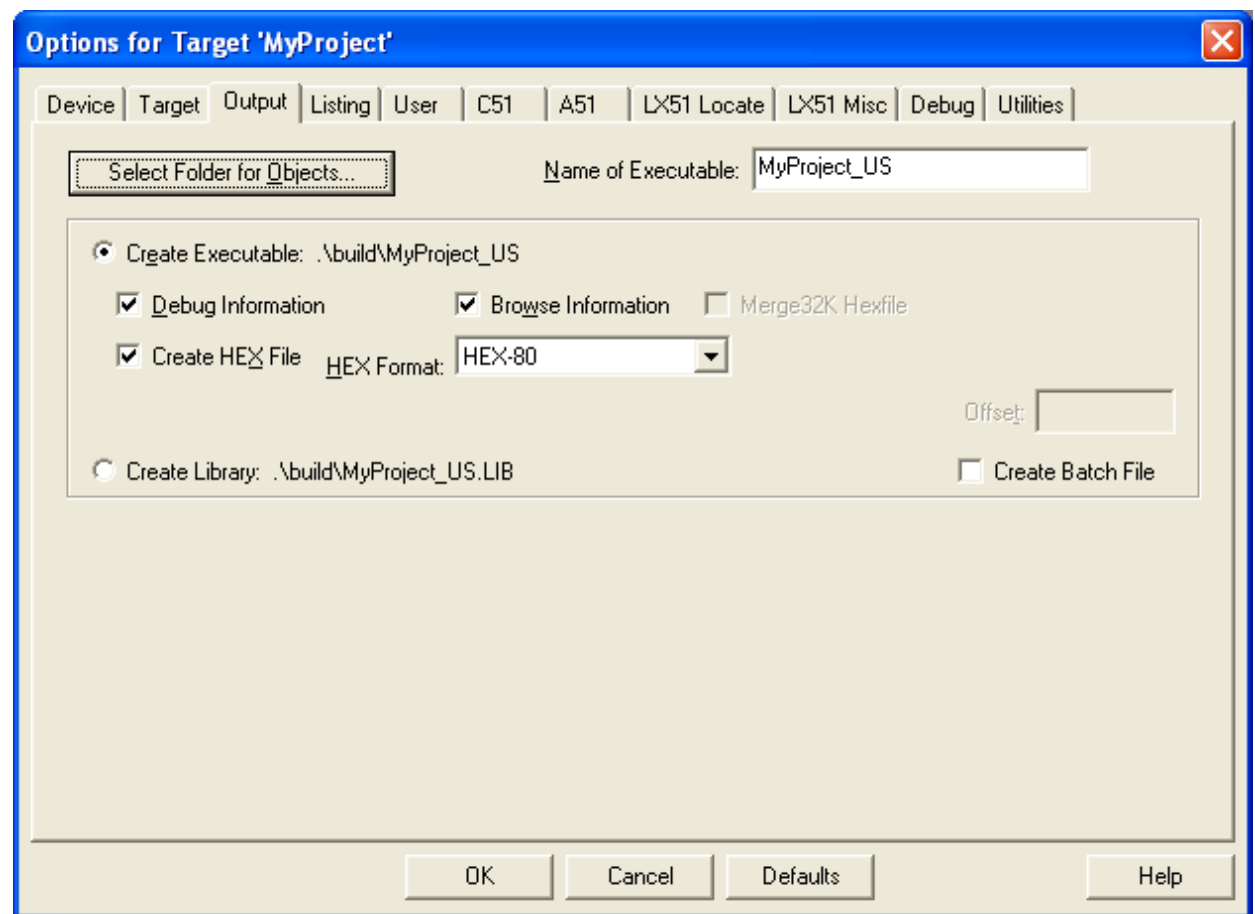


Figure 11. Options for Target dialog with Output tab

3.3.1 Select folder for listings

In the “**Listing**” tab click on the “**Select Folder for Listings...**” button and browse to the folder “C:\DevKit_5_01\Product\LED_Dimmer\build”:

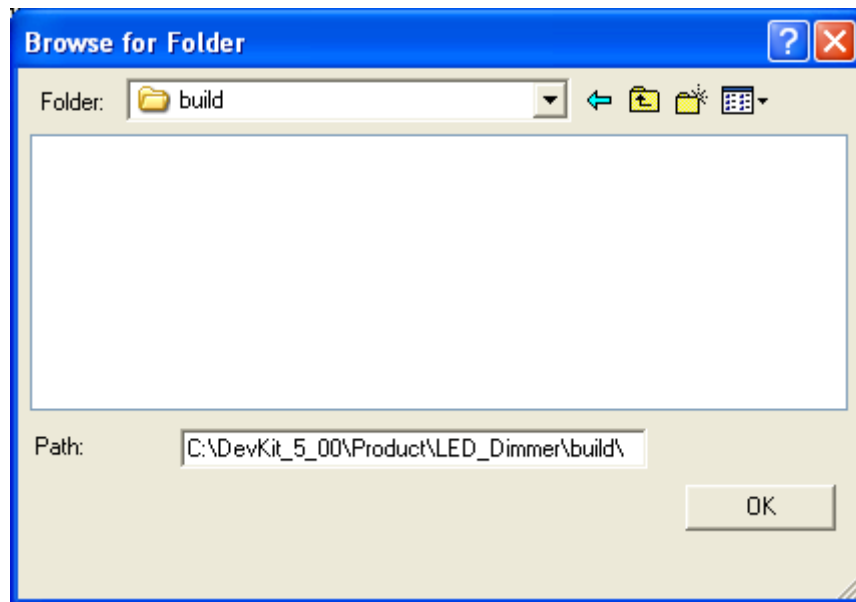


Figure 12. Browse for Folder dialog

Set the listing option for the compiler, assembler and the linker as shown or use the default values:

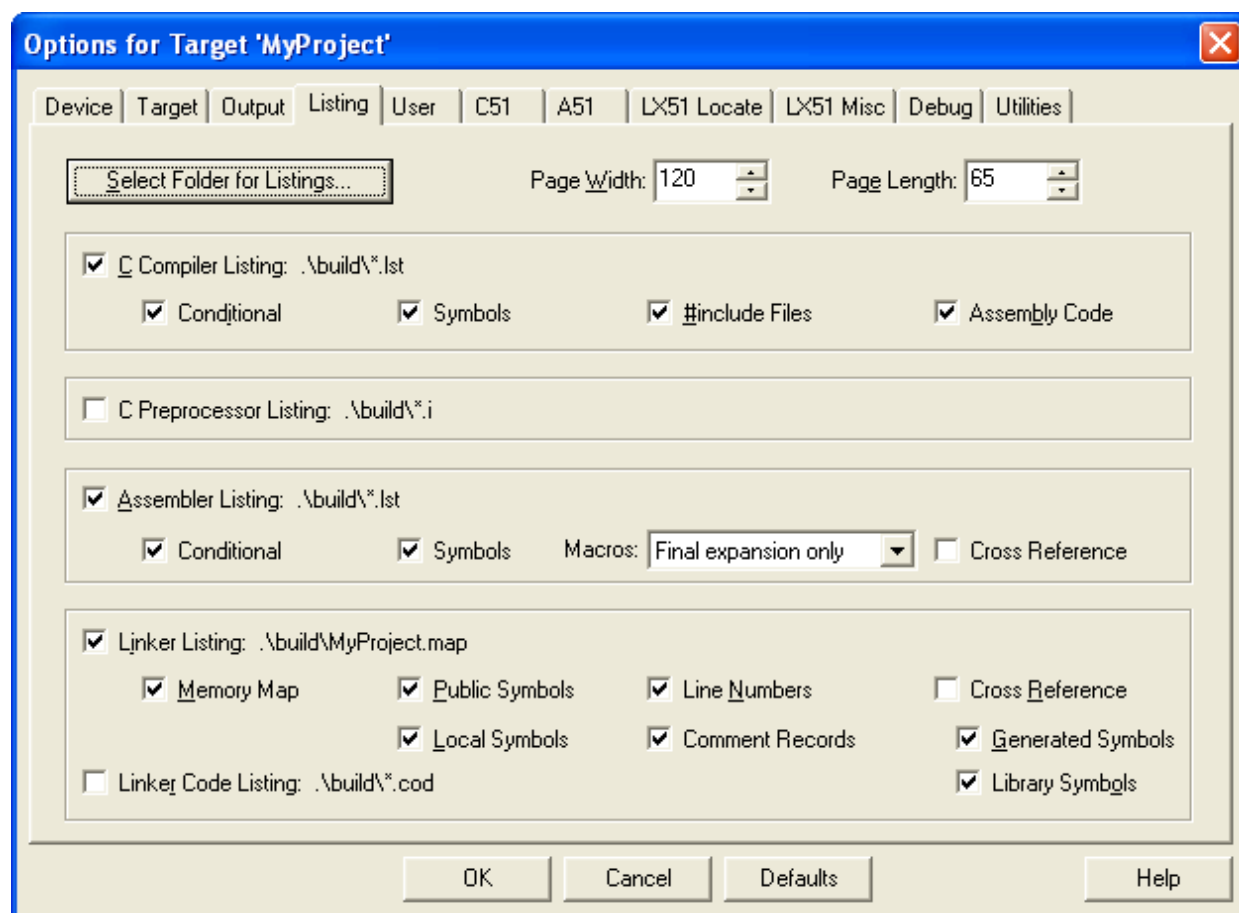


Figure 13. Options for Target dialog with Listing tab

3.3.2 Set compiler defines

In the “C51” tab set the following defines in the “Define” Edit box:

ZW_SLAVE, ZW020x, ZW0201, US, NEW_NODEINFO, APPL_PROD_TEST

Defines are found in the relevant makefiles in the application directory and the common makefile directory.

The LED Dimmer application is located in the directory ‘C:\DevKit_5_01\Product\LED_Dimmer’ where the relevant makefiles for the ZW0201 single chip are located. The needed defines for an US ZM1220 LED Dimmer are assigned to the CDEFINES parameter in the makefiles:

```
CDEFINES+=US, \
    APPL_PROD_TEST
```

The common makefile is located in the directory ‘C:\DevKit_5_01\Product\Common’. The LED Dimmer application uses a slave library and therefore the relevant makefile for the ZW0201 single chip is ‘Makefile.common_ZW020x_slave’. Add the needed defines to the COMDEFINES parameter:

```
COMDEFINES:=ZW_SLAVE,\
            ZW020x,\
            ZW0201,\
            NEW_NODEINFO
```

Be aware that some applications can have nested makefiles and it is therefore necessary to look through all the relevant makefiles after the CDEFINES/COMDEFINES. Regarding a description of defines; please refer to [4].

Set the “**Include Paths**” to:

```
..\..\Z-Wave\IO_Defines;..\..\Z-Wave\include;..\Util_Func
```

For Developer’s Kit v3.2x and v3.3x set the “**Include Paths**” to

```
..\Slave;..\..\Z-Wave\include;..\Util_Func
```

The paths correspond to the groups “**Z-Wave Lib**” “**Z-Wave Inc**” and “**Utilities**”. The paths for the “**Z-Wave Inc**” and “**Utilities**” groups are always the same, independent of the targeted sample application.

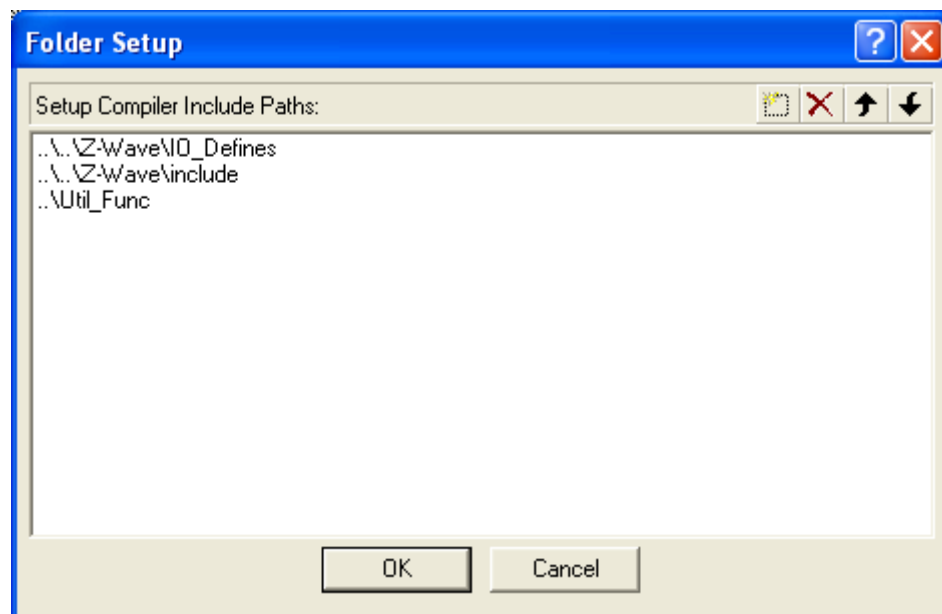


Figure 14. Folder Setup dialog

In the “**Misc Controls**” edit box enter the following compiler options:

DB OE NOCO SB LC CD

For an explanation for the C51 compiler options, please refer to [1].

Set the other options as shown below:

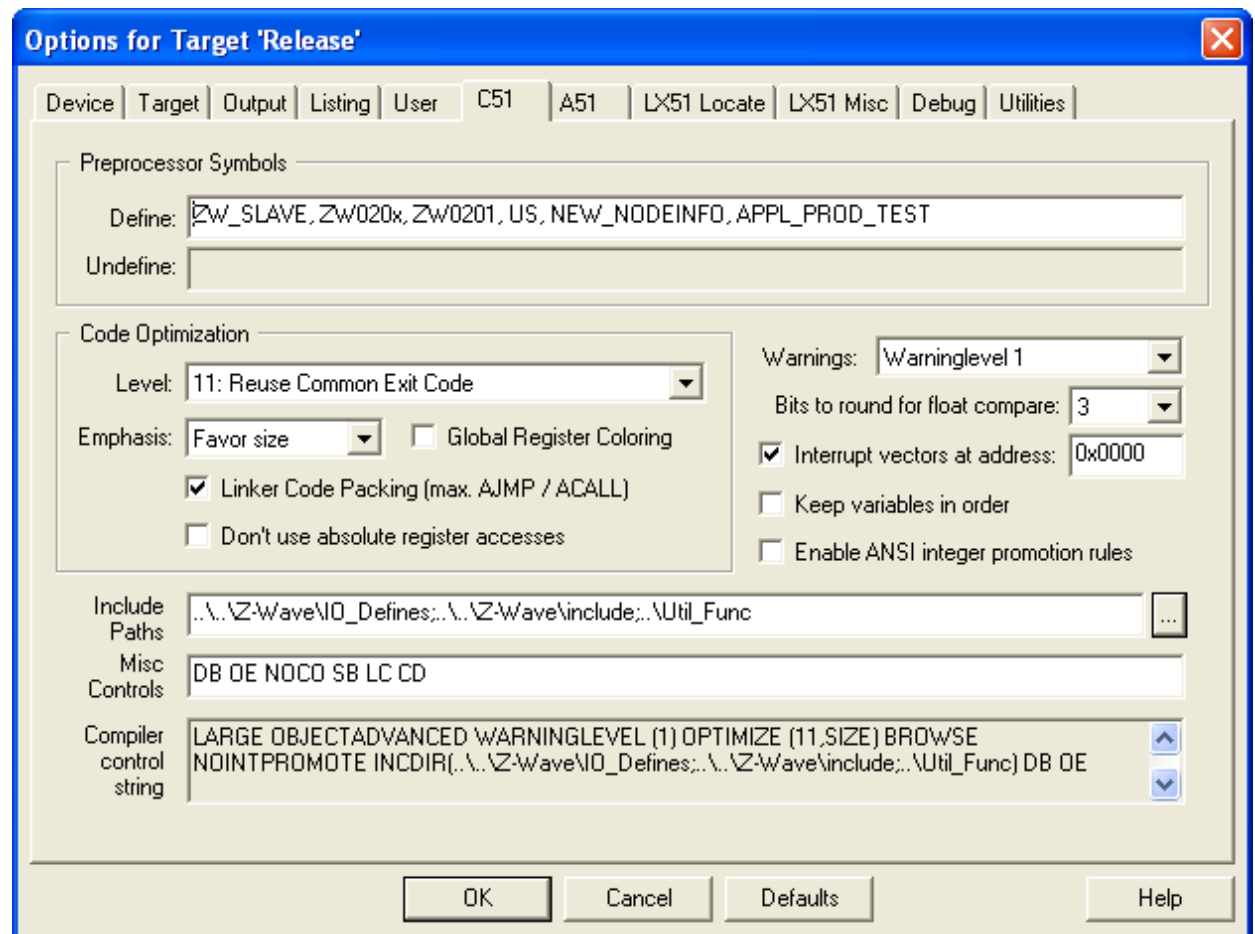


Figure 15. Options for Target dialog with C51 tab

3.3.3 Set assembler defines

In the 'A51' tab set the 'Include Paths' values to the same as used in the "C51" tab.

In the 'Misc Controls' edit box set the following options:

NORB DF(ZW_SLAVE,ZW020x,ZW0201,US,NEW_NODEINFO)

For an explanation of the A51 assembler options please refer to [2], but be aware that the DF directive is an undocumented feature. Finally, for an explanation of the C defines please refer to [4]

Set the other options as shown below:

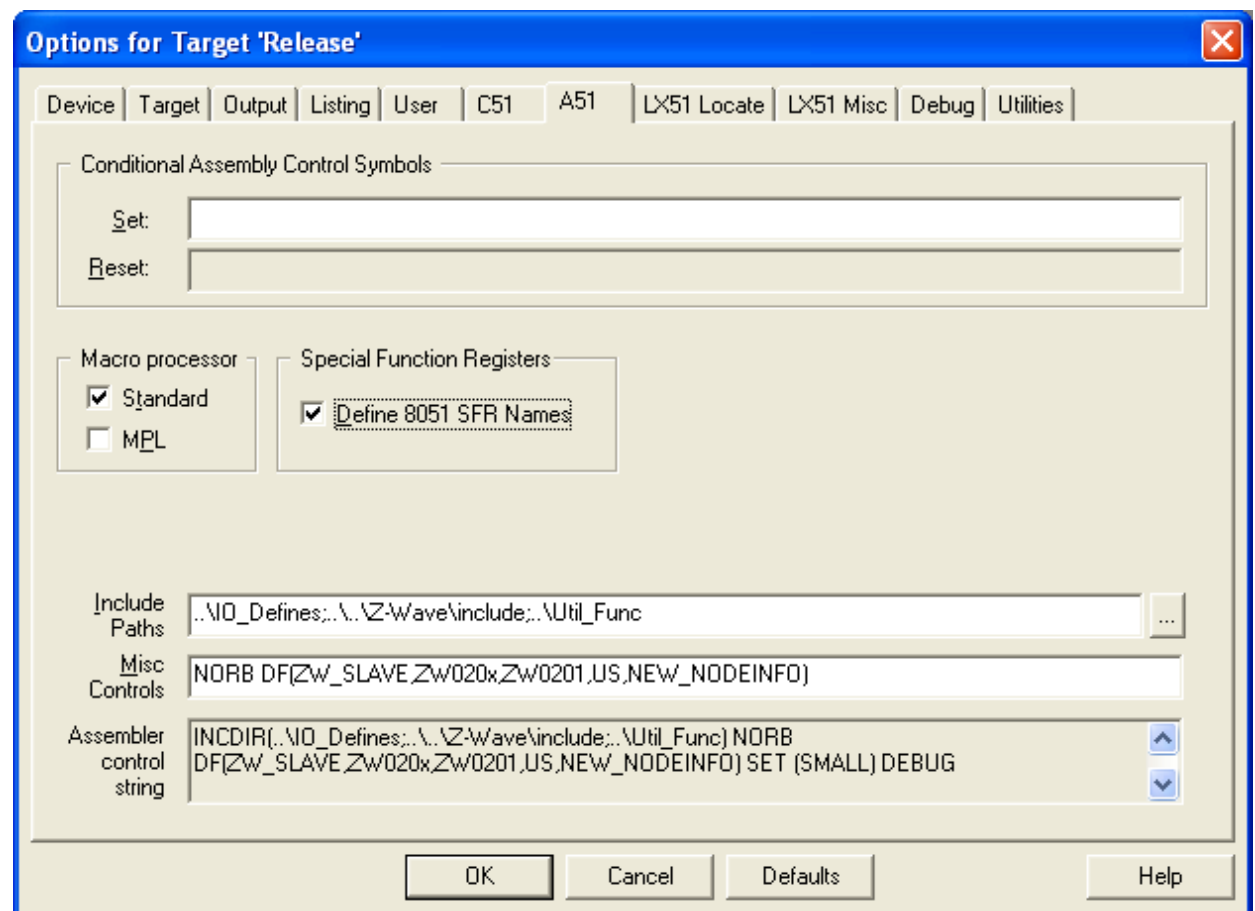


Figure 16. Options for Target dialog with A51 tab

3.3.4 Set linker defines

In the 'LX51 Misc' tab, set linker options in the 'Misc Controls' edit box as follows:

DW (13,16,25) NOOL CLASSES(IDATA(D:0-D:0xff),XDATA(X:0-X:0x7ff),CODE(C:0-C:0x7fff))
SEGMENTS(?CO?ZW_RF020X_FTBL (C:0X7F80))

Now Target option configured correctly. For an explanation of the linker options please refer to [2].

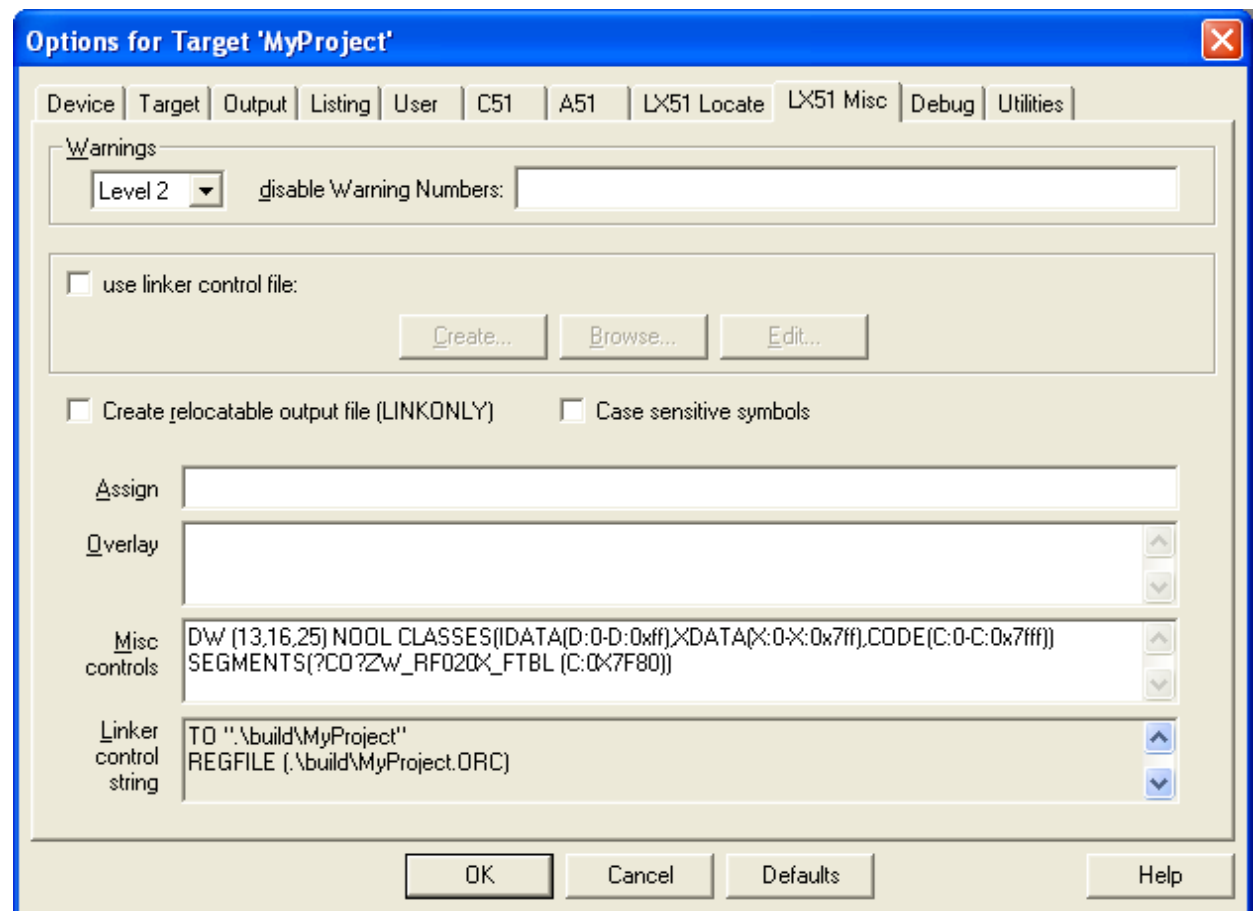


Figure 17. Options for Target dialog with LX51 Misc tab

3.3.5 Set utilities

Configuring Flash Tools.

In the '**Utilities**' tab set the flasher options as follows:

Select "Use External Tool for Flash Programming"

In "Command" field type the full path to the Z-Wave Programmer:

C:\Program Files\Zensys\ZWaveProgrammer\ZWaveProgrammer.exe

in the "Arguments" field type the Z-Wave Programmer command line:

-c COM19 -pf US -p "#H"

where COM19 indicates the COM port to what Programmer is connected and US select the frequency for a given country code.

For country codes and other command line options (home id, EEPROM programming) see the Zensys, INS10679, Instruction, Z-Wave Programmer User Guide.

Select "Run Independent" check box.

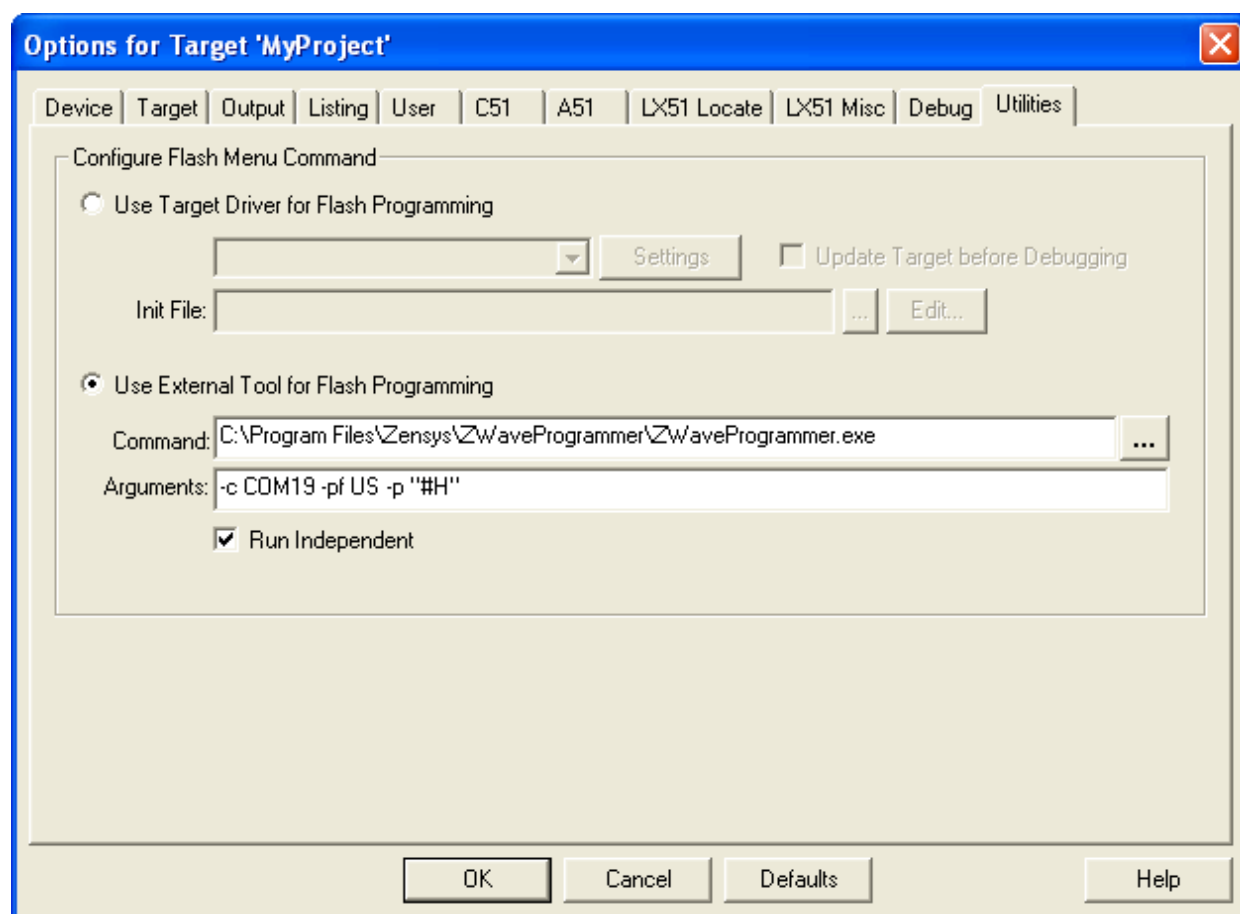


Figure 18. Options for Target dialog with Utilities tab

Refer to "Uploading firmware to the Z-Wave Module" regarding how to upload the firmware.

3.4 Building target

To build the target, select '**Build target**' from the '**Project**' menu. The hex file must be successfully generated which can be verified in the Output Window.

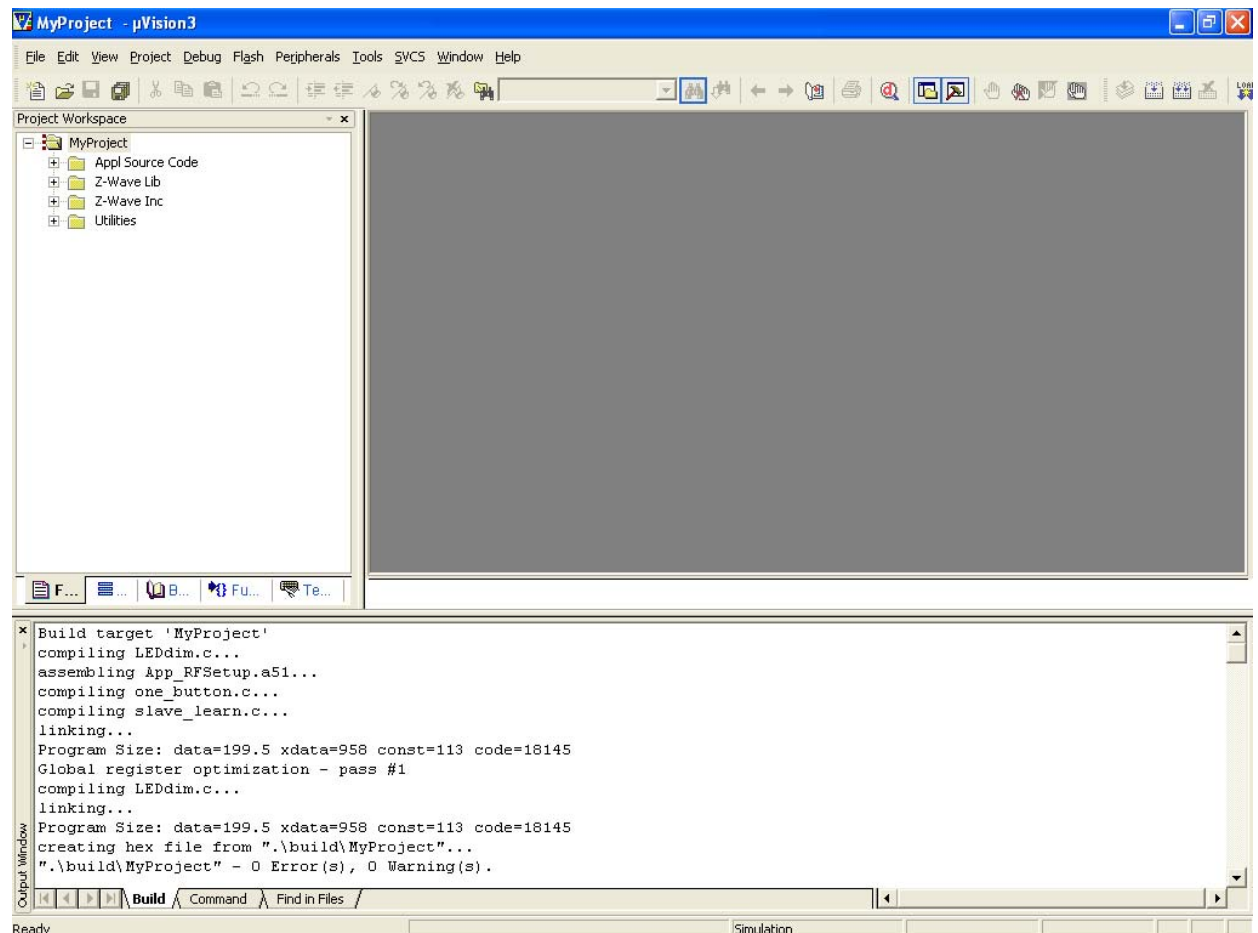


Figure 19. Output that KEIL uVision3 generated during build

3.5 Uploading firmware to the Z-Wave Module

After successfully build, upload firmware to the Z-Wave Module using Z-Wave Programmer.

Regarding how to configure the flash tool refer to

3.5.1 Set utilities

Configuring Flash Tools.

To download the firmware, select the **Download** from the **Flash** menu:

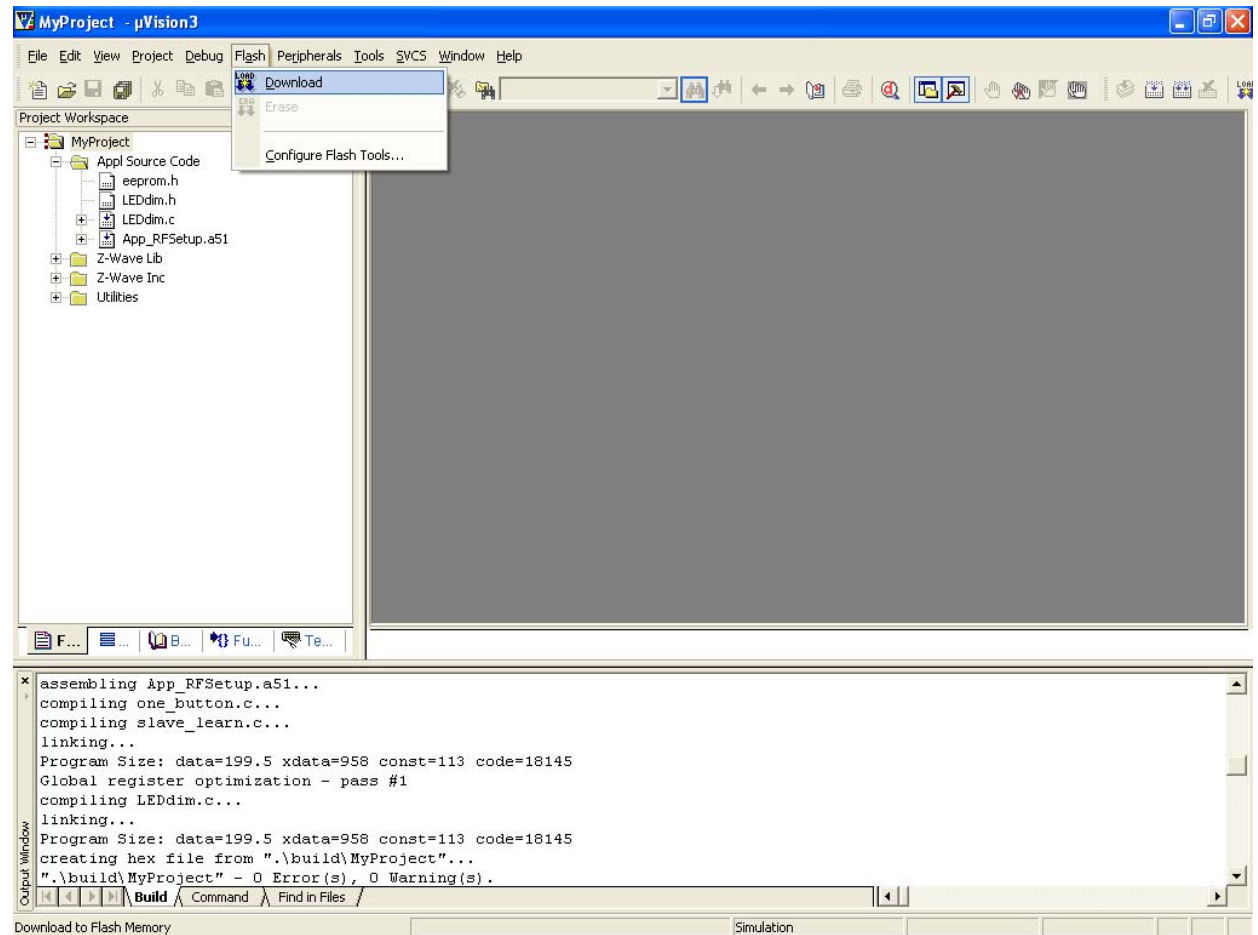
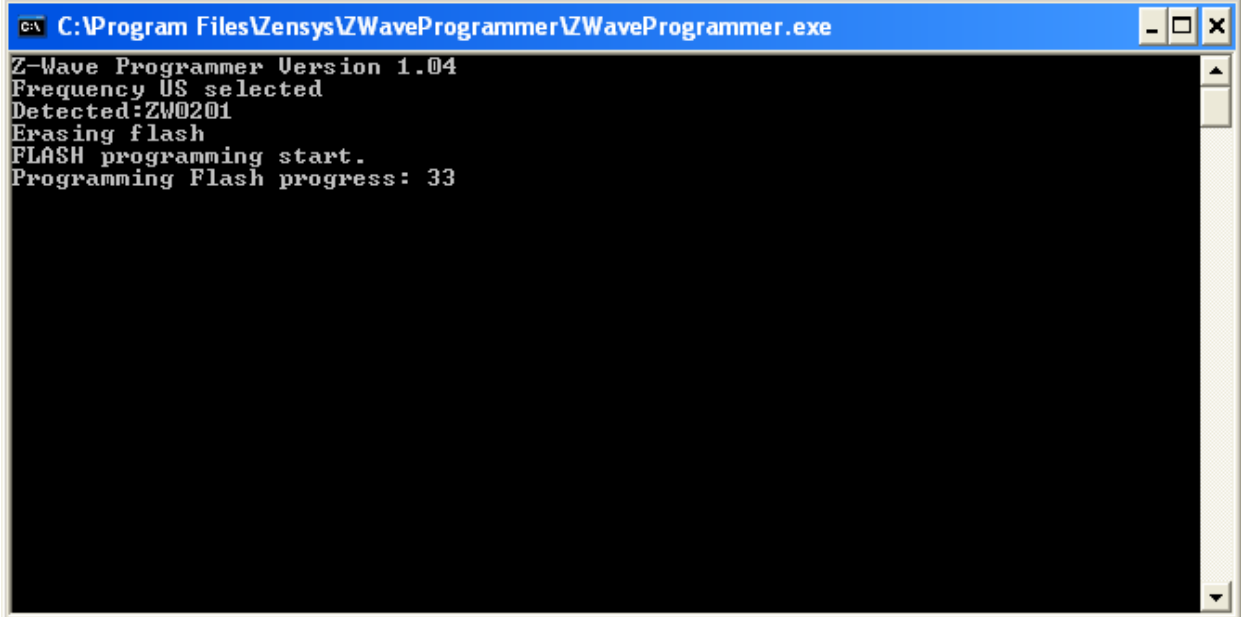


Figure 20. Flash menu

Now the firmware should start downloading to the ASIC.

A screenshot of a Windows-style application window titled "C:\Program Files\Zensys\ZWaveProgrammer\ZWaveProgrammer.exe". The window contains a text area with the following output: "Z-Wave Programmer Version 1.04", "Frequency US selected", "Detected:ZW0201", "Erasing flash", "FLASH programming start.", and "Programming Flash progress: 33". The text is in a monospaced font on a black background. The window has standard minimize, maximize, and close buttons in the title bar.

```
C:\Program Files\Zensys\ZWaveProgrammer\ZWaveProgrammer.exe
Z-Wave Programmer Version 1.04
Frequency US selected
Detected:ZW0201
Erasing flash
FLASH programming start.
Programming Flash progress: 33
```

Figure 21. Z-Wave Programmer output

3.6 Creating and using multi project workspace

3.6.1 Creating new multi project workspace

In case when project must be able to generate firmware for different countries, then create multiple uVision projects for each country. After this, add all such projects to single uVision workspace.

To create workspace select “New...” -> “Project Workspace” from “Project” menu of the uVision3.

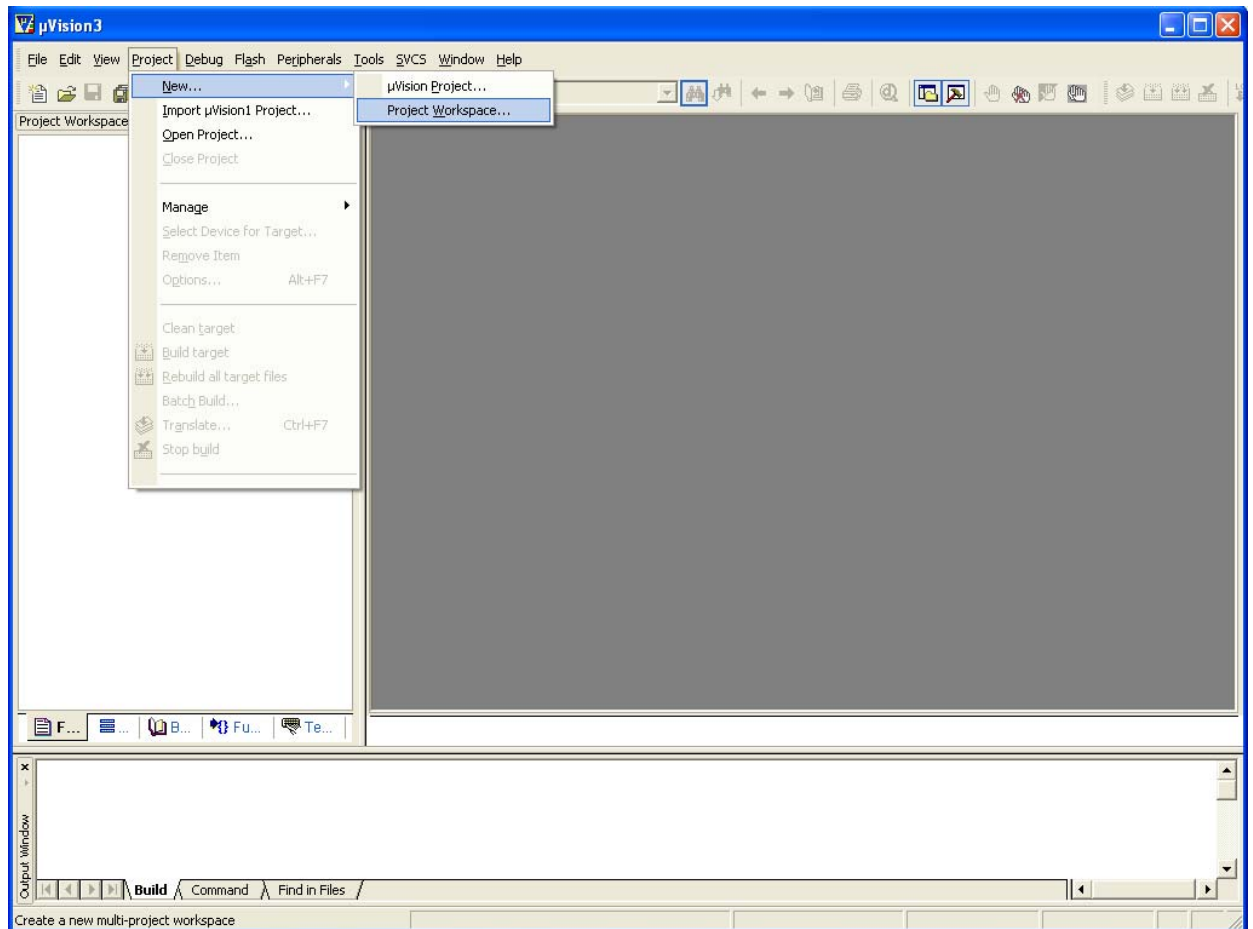


Figure 22. Workspace creation

Select the workspace directory and name. In this example the workspace directory is C:\DevKit_5_01\Product\LED_Dimmer and the workspace name is MyProject.

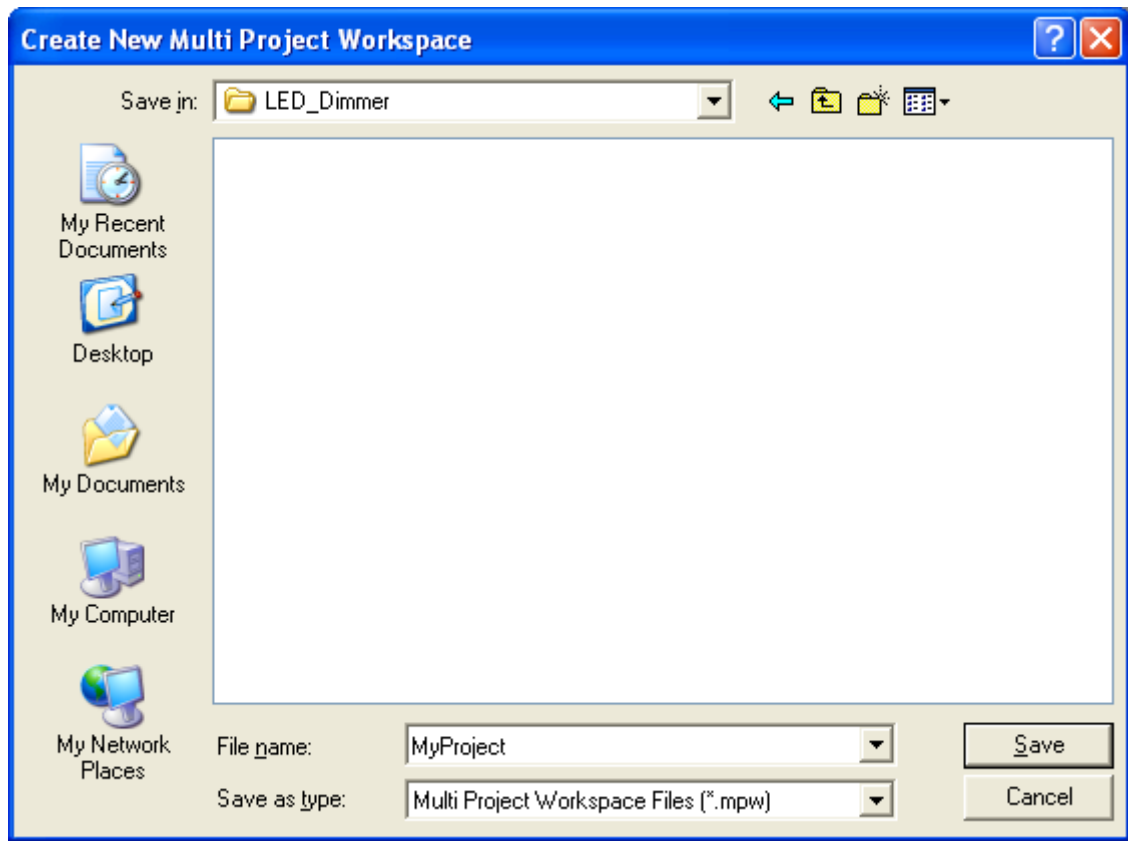


Figure 23. Dialog to select the folder for new workspace

Now add the all uVision project files to the work space:

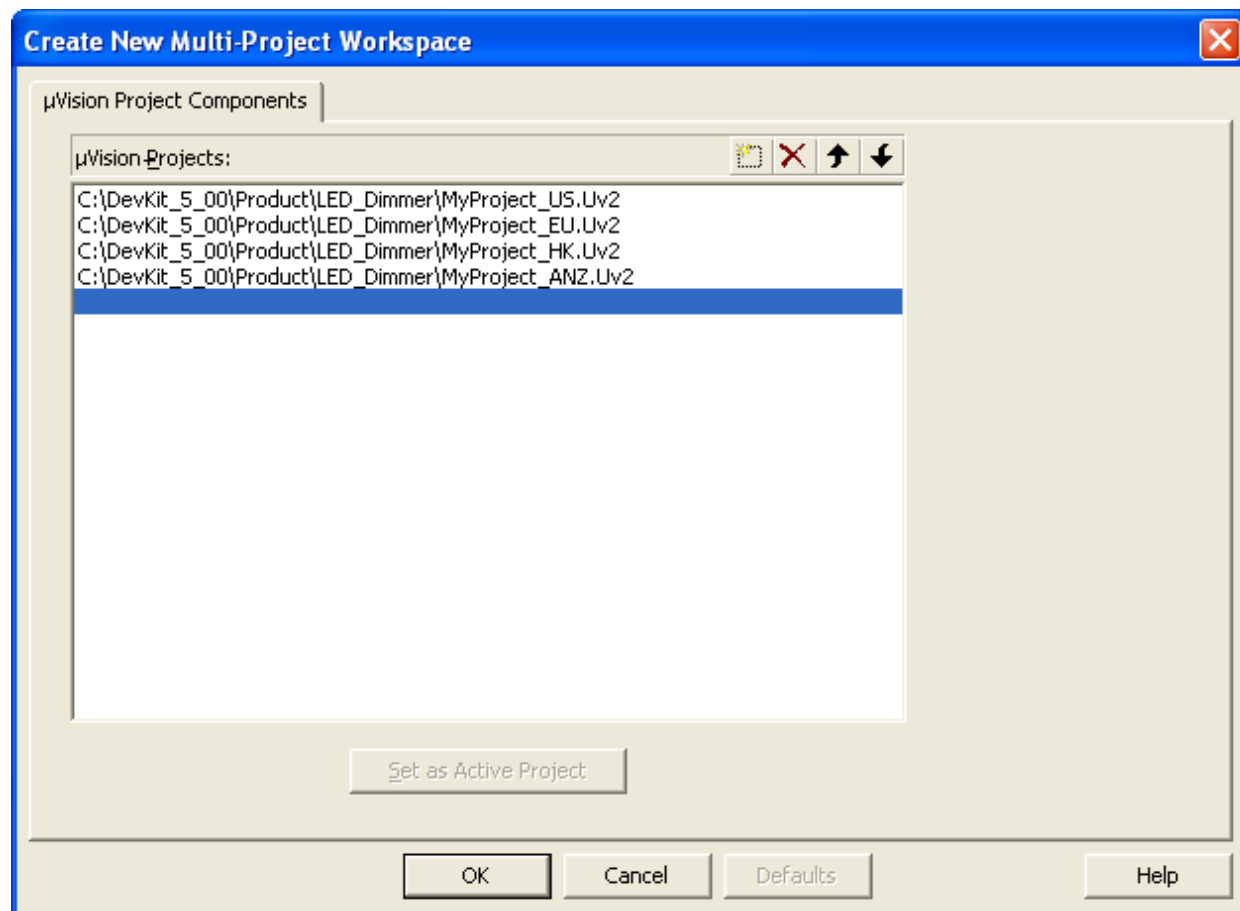


Figure 24. Create New Multi Project Workspace dialog with the required projects added

3.6.2 Using multi project workspace

- To set active project in the workspace, select the needed project in the Project Workspace window, click right mouse button on it, and select "Set as Active Project".

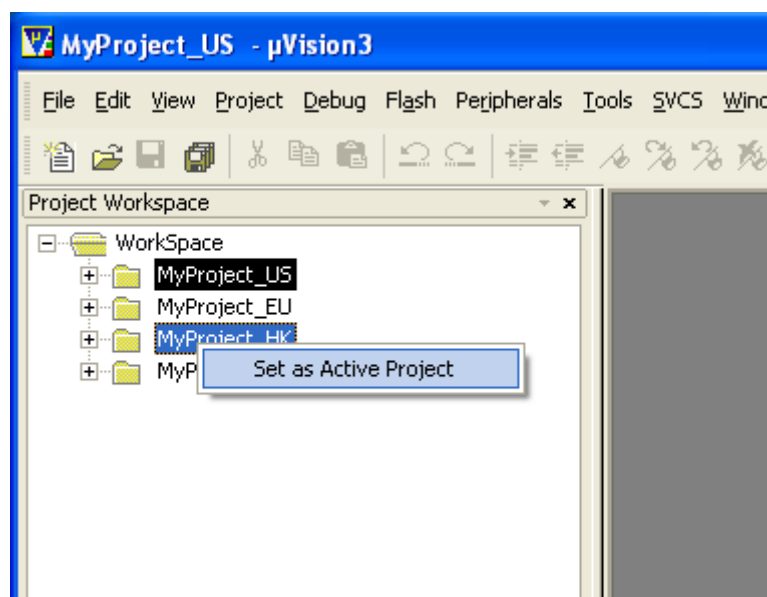


Figure 25. Shortcut menu to set the Active Project

- To build only current active project, see Building target.
- To build multiple projects at once, select "Batch Build..." from "Project" menu, and select Project targets, which need to build, or press "Select All" to select all Targets.

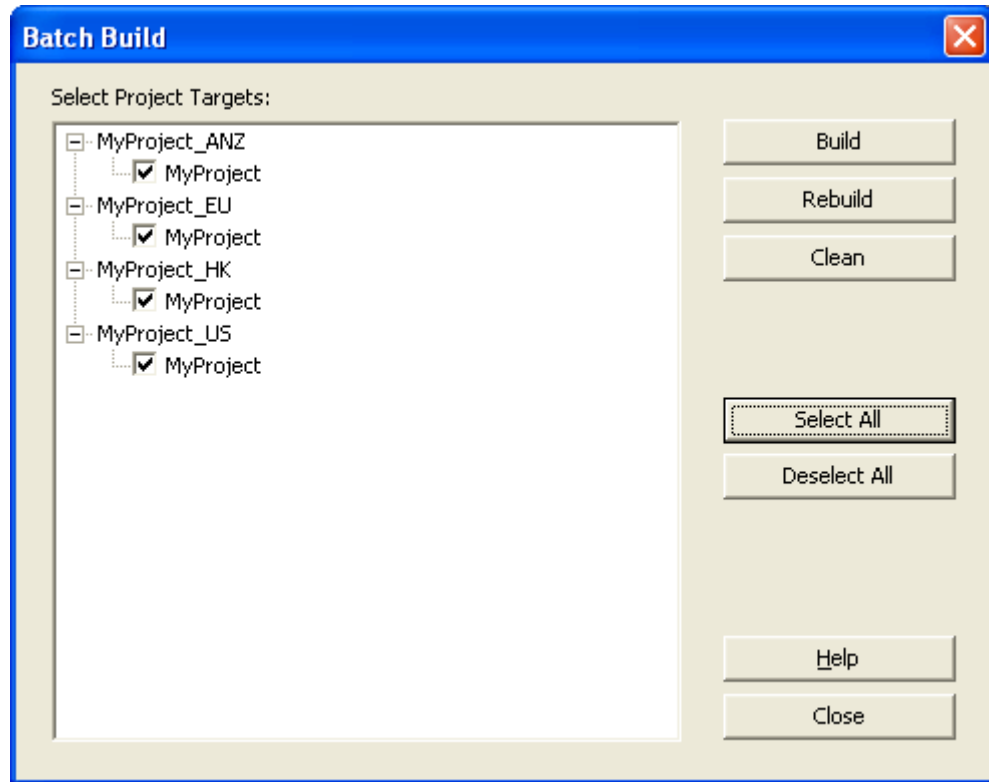


Figure 26. Batch Build dialog

Now press button with desired action: Build, Rebuild or Clean.

- To have possibility in opening workspace files in uVision from Windows Explorer see Associate uVision multi project workspace file in Chapter 8.

4 THE SIMULATOR

The Keil uVision3 IDE has an integrated simulator. To use the simulator to debug Z-Wave application an add-in simulator DLL module must be installed to simulate the ZW0102 or ZW0201 specific hardware and the external EEPROM if present. There are no plans regarding development of these add-in DLLs, but the description below explains how to add them to the IDE.

1. Assume that the add-in DLL called '**zw0102.dll**' or '**zw0201.dll**'
2. Copy the file to 'c:\keil\c51\bin', if this is the path to the IDE.
3. From the '**Project**' menu select '**Options for target LEDdim_US**'.
4. Select the Debug pane.
5. Write the following in the '**CPU DLL**' edit box: **S8051.DLL**.
6. Write the following in the '**Dialog DLL**' edit box: **DP51.DLL**
7. Write the following in the '**Parameter**' edit box: **-p320 -dzw0102** or: **-p320 -dzw0201**.

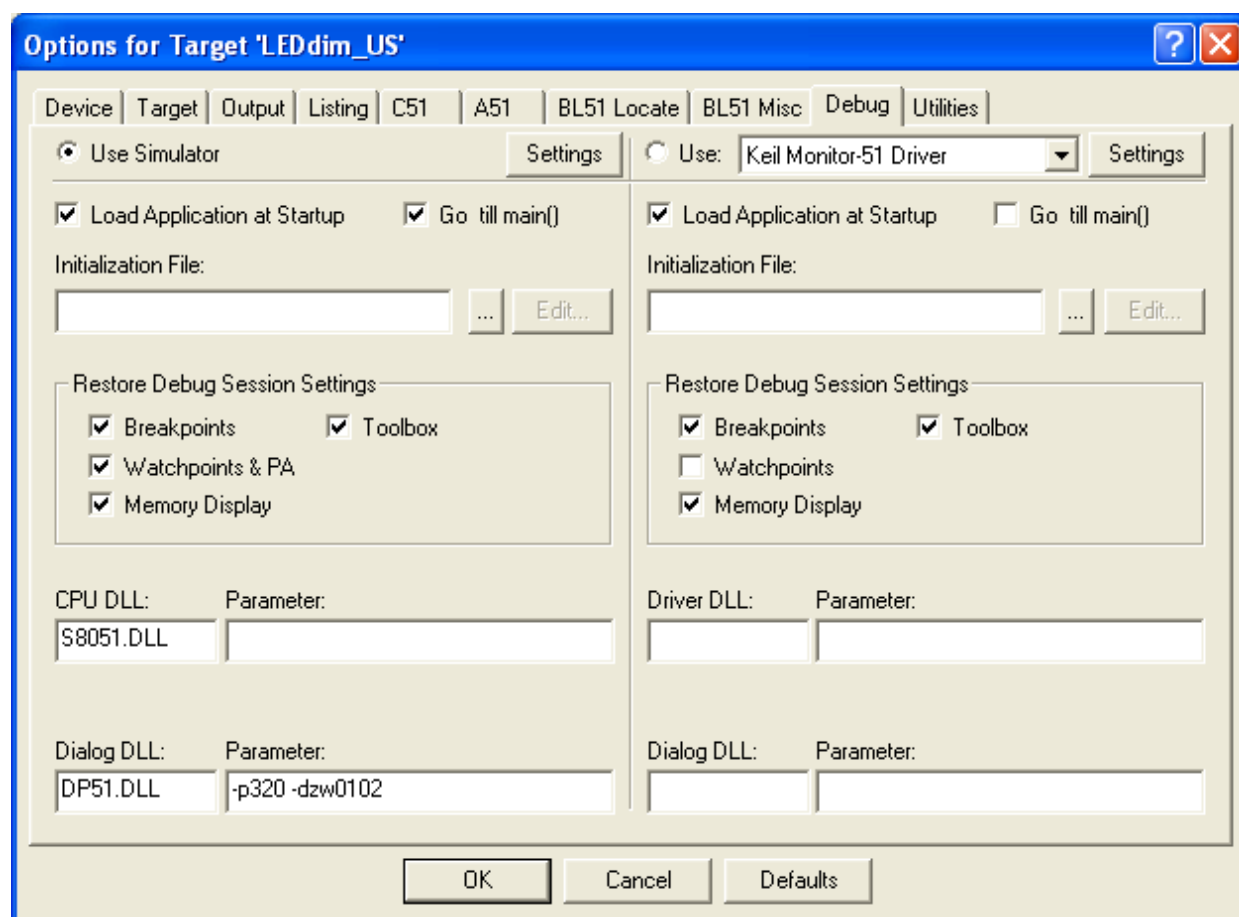


Figure 27. Options for Target dialog with Debug tab

5 CREATING A NEW DEVICE IN THE DEVICE DATABASE

To add new devices follow the steps below:

- 1 Select a CPU that is similar to the new device. Be sure to double-click to bring the settings for this device into focus.
- 2 Change the name of the chip vendor.
- 3 Select the appropriate CPU family.
- 4 Enter the part number in the device text box.
- 5 Modify the description to match the new device.
- 6 Change the options for the tools (see below).
- 7 Single-click in the database window (to change focus). This enables the Add button.
- 8 Click Add to add the new device.

5.1 Adjusting the Options...

In the Options box, **CPU=** specifies the basic tool settings for the chip. The parameters for **CPU=** are:

- **IRAM (range)**
Address location of the on-chip IRAM.
- **XRAM (range)**
Address location of the on-chip XRAM.
- **XRAM2 (range)**
Address location of a 2nd on-chip XRAM area (i.e. for C167CS).
- **IROM (range)**
Address location of the on-chip (flash) ROM. The start address must be 0; the part is automatically split into two sections, if the size is more than 32KB. The range specifies the physical ROM size.
- **ICAN (range)**
Address location of the on-chip CAN module. Use for C167 and variants only.
- **CPUTYPE (variant)**
Specify CPU variant for ARM based controllers. Currently only ARM7TDMI is supported.
- **EBIG**
Default to BIG endian for ARM based controllers.
- **CLOCK (val)**
Default CPU clock speed (in Hz) used when selecting the device. For example CLOCK(12000000), indicates a 12MHz oscillator.
- **MOD167**
Use the extended instruction set of the C167 and variants.
- **MOD517DP**
Allow to enable Infineon specific multiple DPTR registers.
- **MOD517AU**
Allow to enable the Infineon specific Arithmetic Unit.
- **MODA2**
Allow to enable Atmel specific multiple DPTR registers (like on AT89S8252).
- **MODAB2**
Allow to enable Analog Devices specific multiple DPTR registers.
- **MOD_CONT**
Enable support for the Dallas Contiguous Mode.
- **MODDA**
Allow to enable Dallas specific Arithmetic Accelerator.

- **MODDP2**
Allow to enable Dallas specific multiple DPTR registers.
- **MODP2**
Allow to enable Philips specific multiple DPTR registers. (Note also some Atmel devices are using this variant).
- **MODE2**
Use the Infineon SLE66 instruction set extensions.
- **LP51**
Use the Infineon SLE66 specific LP51 linker instead of LX51.
- **MODV2**
Use the Infineon XC16x / ST Super10 instruction set extensions.
- **MX**
Enable support for the Philips 80C51MX architecture.
- **MXP**
Enable support for the Philips SmartMX SmartCard architecture.
- **DPX**
Enable 24-bit DPTR register for the Analog Devices ADuC812.

In addition there are two other options that specify files. These are:

- **REGFILE (*sfr_file*)**
Specify the SFR register definition file for the CPU.
- **SFILE (*startup_file*)**
Specify the default startup file that should be copied to a project.

Other options specify data books and uVision3 debugging DLLs. Leave these variables unchanged when adding a new device to the database.

6 MODIFYING THE ZW0102 DEVICE IN THE DEVICE DATABASE

The ZW0102 model in the device database is incorrect with respect to Flash and IRAM size. The correct values are 32 KB of Flash ROM and 128 KB IRAM.

The ZW0102 model is corrected as follows:

1. From the “File” menu choose “Device Database”.
2. In the Device Database dialog scroll down to the Zensys devices, double click on Zensys.

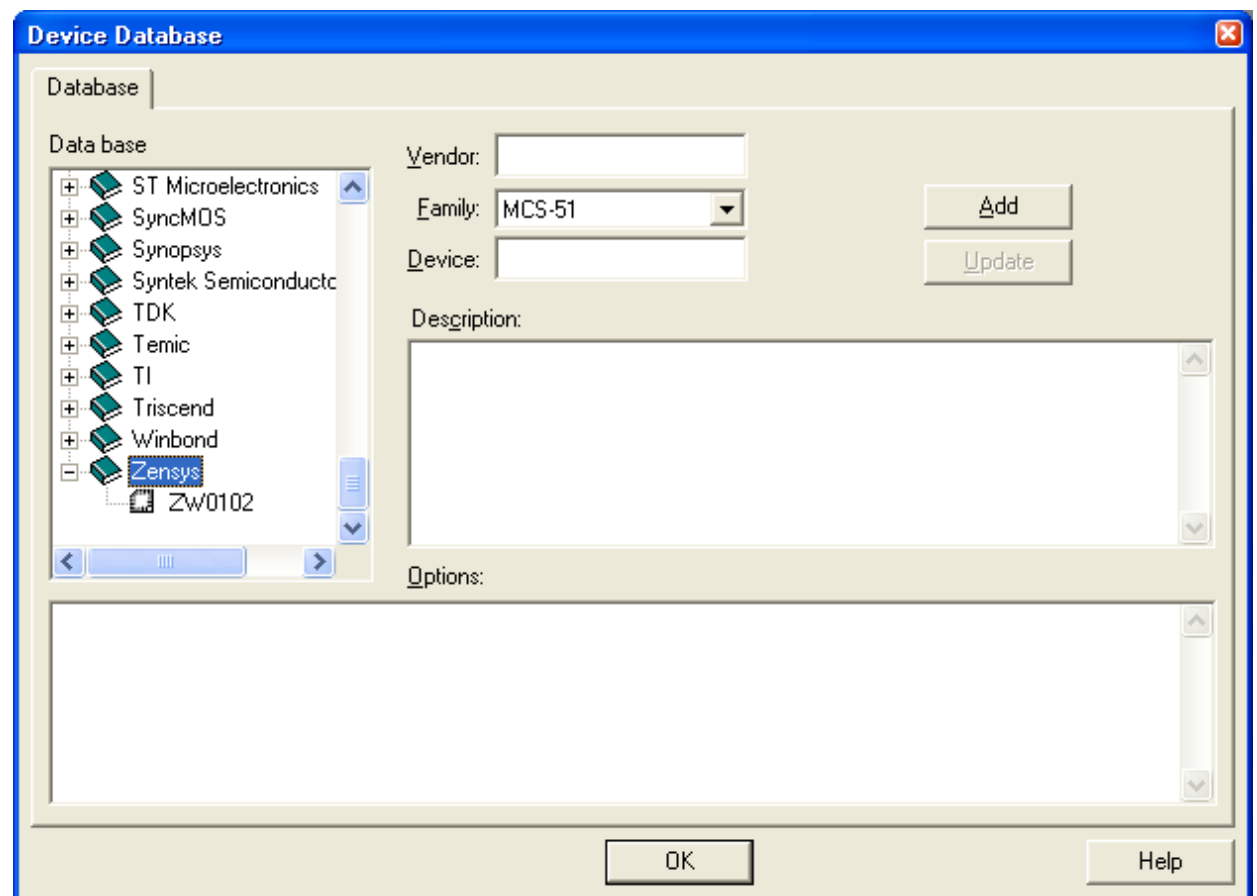


Figure 28. Device selection in the Device Database dialog

- Now double click on ZW0102 and the “**Description**” and “**Options**” textboxes shows the ZW0102 properties.

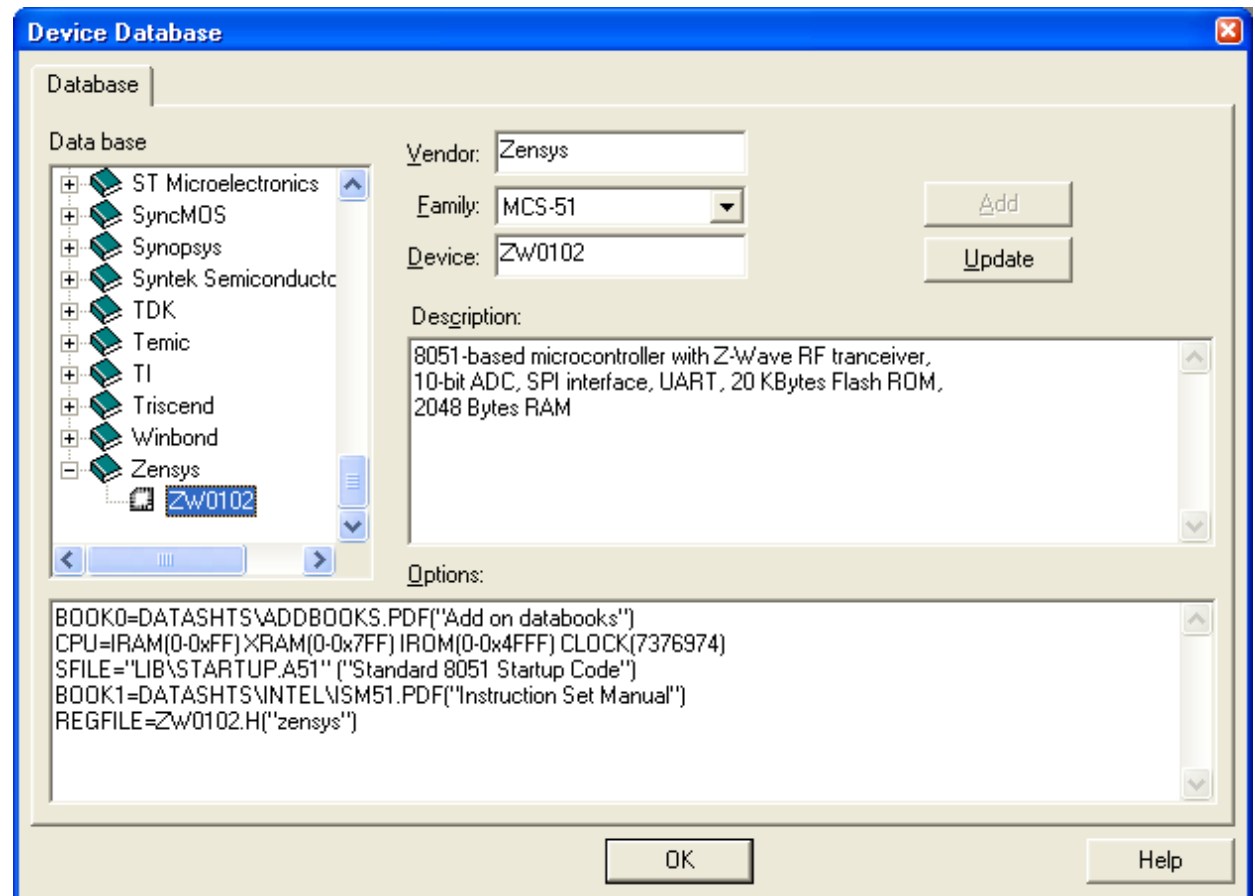


Figure 29. Selecting ZW0102 options in the Device Database dialog

4. In the “Options” textbox, change **IRAM(0-0xFF)** to **IRAM(0-0x7F)**, and **IROM(0-0x4FFF)** to **IROM(0-0x7FFF)**. Remember also to update the “Description” textbox.

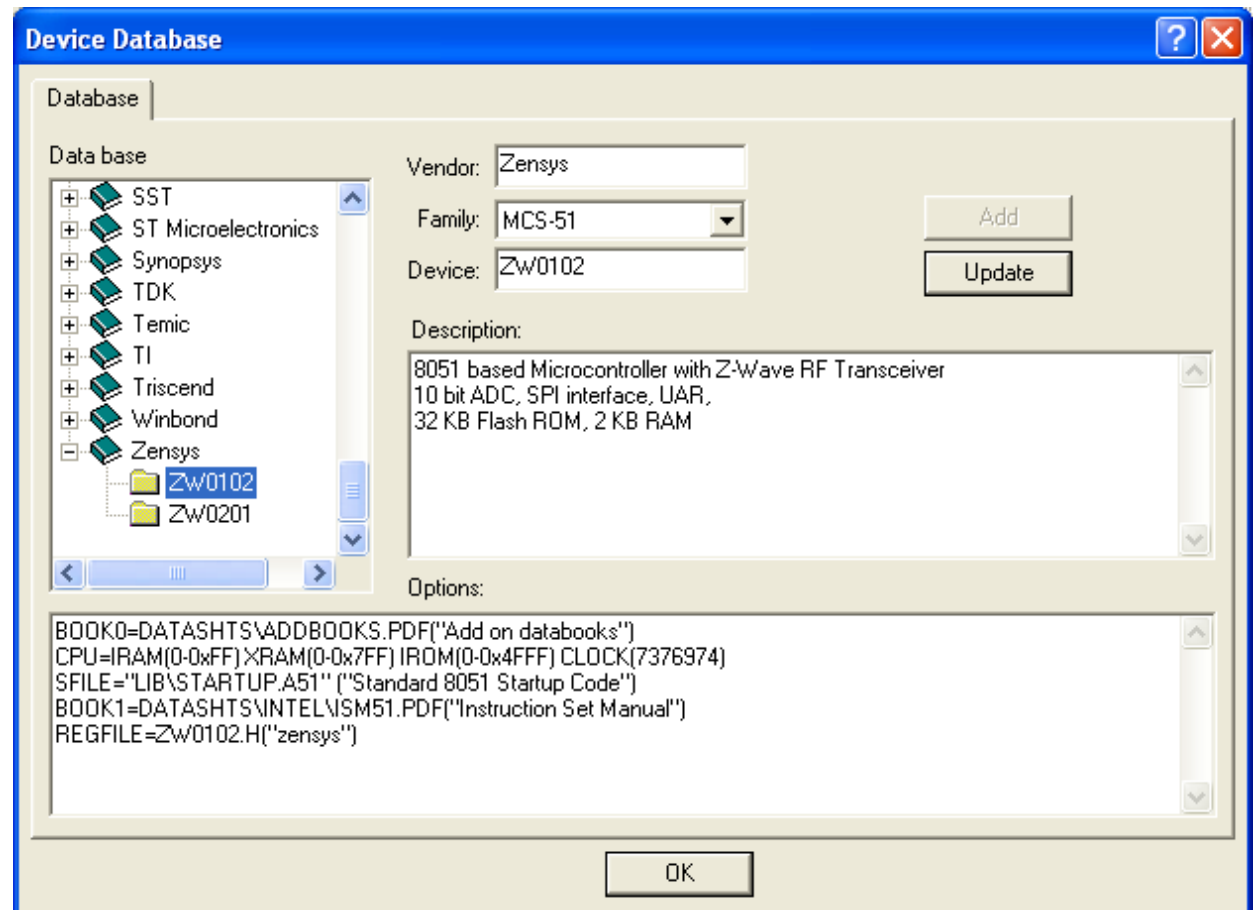


Figure 30. Correcting ZW0102 memory map in the Device Database dialog

5. Finally activate the “Update” button to save the changes.

7 CREATING A MODEL FOR THE ZW0201/ZW0301 DEVICE

It is necessary to create a model in case the Keil uVision IDE do not support the ZW0201/ZW0301 single chip as follows:

1. From the “File” menu choose “Device Database”.
2. The ZW0201/ZW0301 chip uses the Mentor Graphics 8051EW core. Select the Mentor Graphics devices from the “Data base” and double click on 8051EW device.

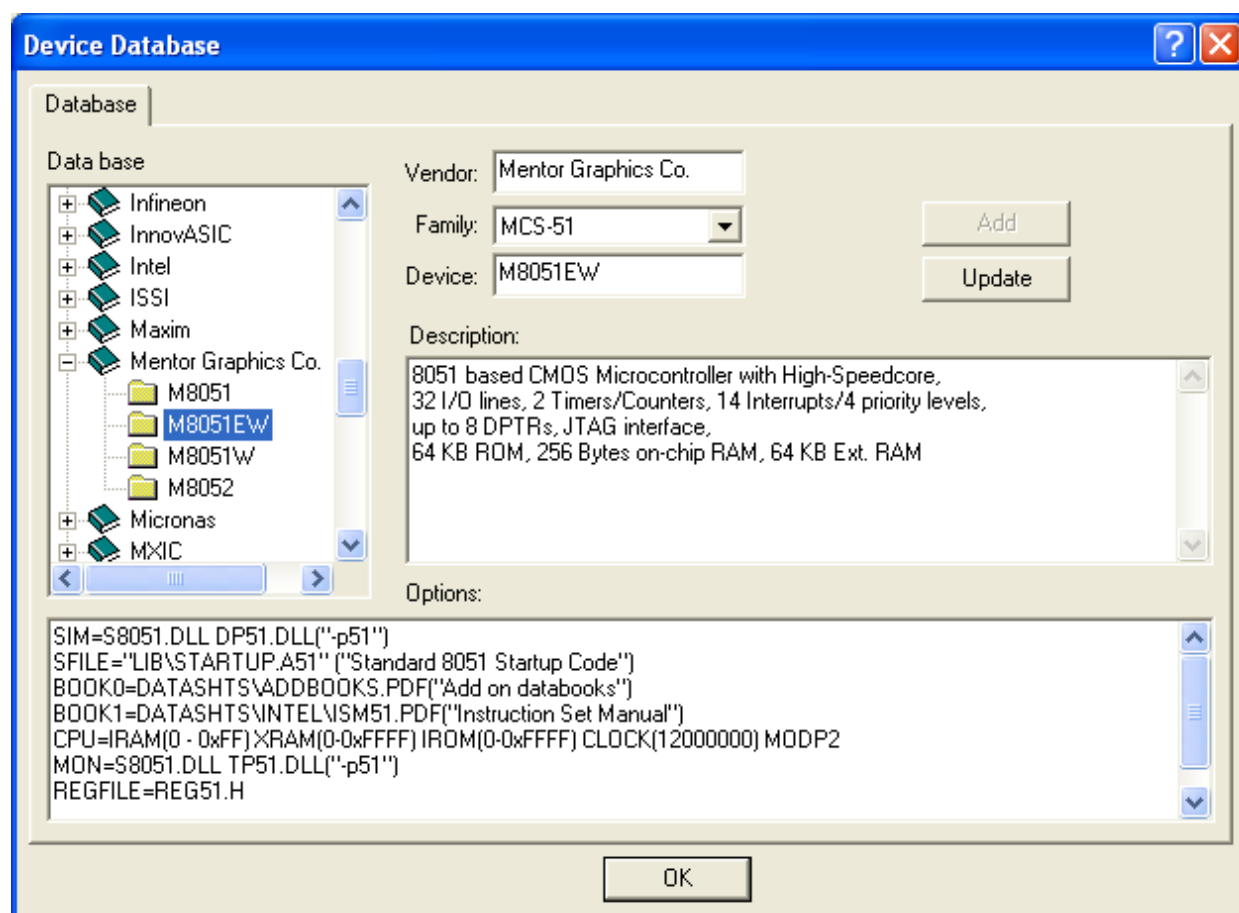


Figure 31. Selecting database in the Device Database dialog

3. Select the Zensys devices from the “Data base”. Update “Vendor” to Zensys and “Device” to ZW0201/ZW0301. Enter a description of the ZW0201/ZW0301 single chip in the “Description” textbox. Modify the “Options” textbox as follow: **CPU=IRAM(0 - 0xFF) XRAM(0-0x07FF) IROM(0-0x7FFF) CLOCK(16000000)**. Finally activate the “Add” button to create the model in the Zensys device database.

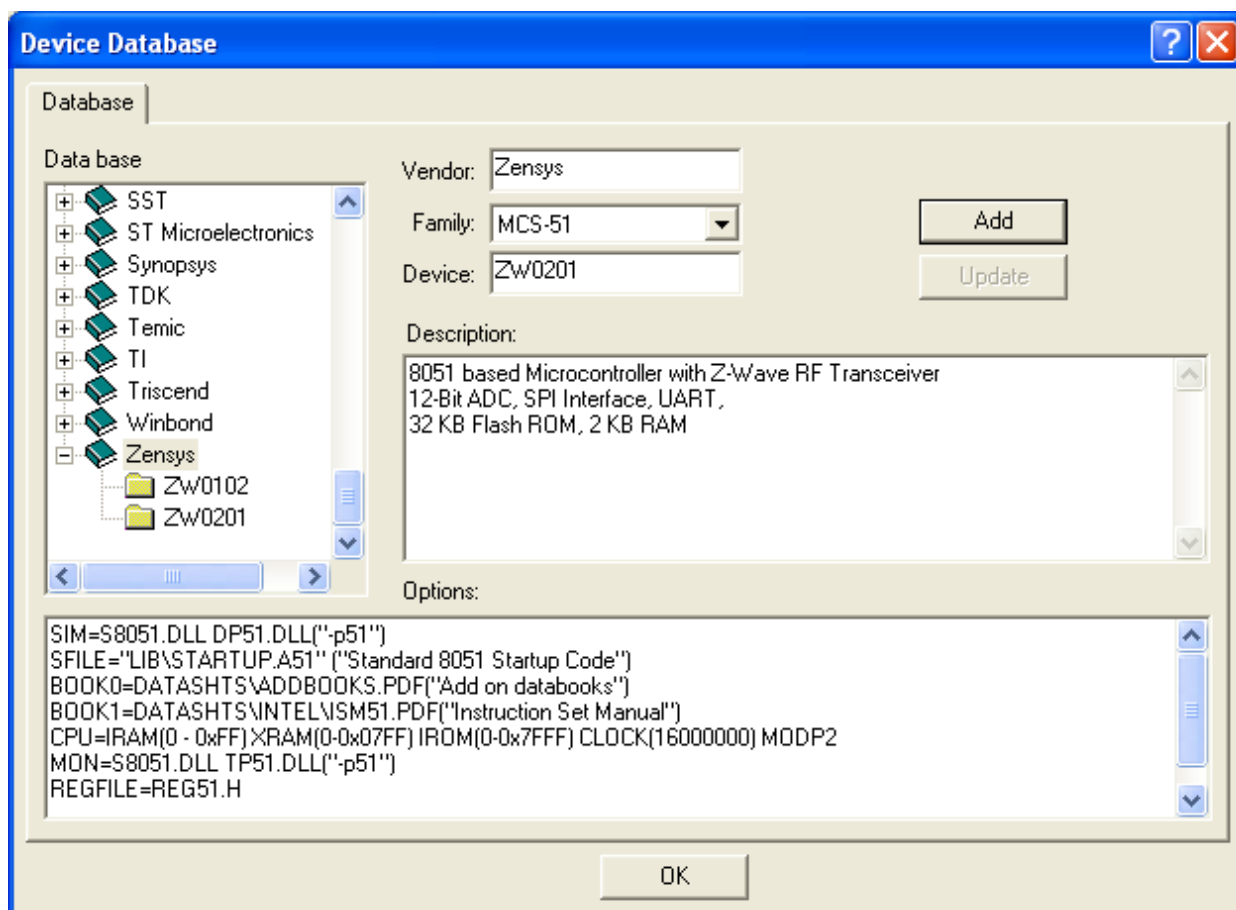


Figure 32. Entering option data in the Device Database dialog

8 TROUBLESHOOTING

- Make sure to add all needed libraries and RF .obj files to the project (see section 3.2).
- Make sure all needed defines and control words is added in the target options tab such as country type, chip name, etc. (see section 3.3).

- Associate uVision multi project workspace file (.mpw) with Keil uVision3 IDE.

If you double-click a .mpw file in Windows and it opens in the wrong program, follow these steps to associate this file extension with Keil uVision3 IDE.

- Open the folder that contains the file you want to change.
- Click on .mpw file or right-click the file, and then click Open.
- Choose "Select the program from a list" and then click "Ok".
- In Programs list select uVision3 IDE.
- Select the "Always use the selected program to open this kind of file" check box, and then click OK

9 REFERENCES

- [1] Keil, C51 Compiler User's Guide.
- [2] Keil, Macro Assembler and Utilities User's Guide.
- [3] Keil, uVision3 User's Guide
- [4] Zensys, INS10247, Instruction, Z-Wave ZW0201/ZW0301 Application Programming Guide.
- [5] Zensys, INS10679, Instruction, Z-Wave Programmer User Guide