

Read me for building the AN61345 project using SDCC Compiler

1) Windows

1) Download and install the appropriate package of sdcc compiler from:

<http://sdcc.sourceforge.net/snap.php>

2) Go to the directory where you downloaded and extracted the AN61345 project and go to
"..\AN61345 - Source files for FPGA code and FX2LP Firmware\001-61345\Firmware_SDCC".

3) Inside this directory, you will see the project files slave.c, fw.c etc. Also present is a directory called Release. This Release directory contains the files that are generated when the project files mentioned above are built using the sdcc compiler. The slave.hex file which will be used to program the FX2LP device is also present in this Release directory.

Note

User can opt to stop reading the Windows section here and directly program the FX2LP device with the slave.hex file present in the Release directory specified above.

But

*If the user wants to know the steps of how to actually compile the project files present in the
"..\AN61345 - Source files for FPGA code and FX2LP Firmware\001-61345\Firmware_SDCC" and generate the slave.hex file, then one can continue reading this section.*

4) Now open the command prompt

5) cd to the Project directory :

"..\AN61345 - Source files for FPGA code and FX2LP Firmware\001-61345\Firmware_SDCC".

7) There is a script file named "commands.bat" present in the above directory. User can run this script by simply typing commands.bat in the command prompt. Running this script will build the project and generate the files for the Release directory. The slave.hex file generated can then be used to program the FX2LP Device.

OR

In the command prompt, run the following commands and generate the files for the Release folder, which will replace the existing ones:

```
sdcc -c slave.c -o Release/slave.rel
```

```
sdcc -c fw.c -o Release/fw.rel
```

```
sdcc -c delay.c -o Release/delay.rel
```

```
sdcc -c discon.c -o Release/discon.rel
```

```
sdcc -c EZRegs.c -o Release/EZRegs.rel
```

```
sdcc -c get_strd.c -o Release/get_strd.rel
```

```
sdcc -c i2c.c -o Release/i2c.rel
```

```
sdcc -c i2c_rw.c -o Release/i2c_rw.rel
```

```
sdcc -c resume.c -o Release/resume.rel
```

```
sdas8051 -log Release/dscr.rel dscr.asm
```

```
sdas8051 -log Release/susp.rel susp.asm
```

```
sdas8051 -log Release/delayms.rel delayms.asm
```

```
sdas8051 -log Release/USBmpTb.rel USBmpTb.asm
```

```
cd Release
```

```
sdcc -o slave.hex fw.rel slave.rel delay.rel delayms.rel discon.rel dscr.rel USBmpTb.rel EZRegs.rel  
get_strd.rel i2c.rel i2c_rw.rel resume.rel susp.rel
```

9) The slave.hex will be generated after the step above. This file can now be used to program the FX2LP Device.

II) Linux(Ubuntu)

1) Download the appropriate package of sdcc compiler from:

<http://sdcc.sourceforge.net/snap.php>

2) Before installing the compiler, you need to give root permissions without which the compiler will not get installed properly. To do that, open the terminal and type the command "sudo su".

3) Unzip the installed package and go through the "INSTALL.txt" present at "./sdcc/share/doc". Please adhere strictly to the installation guide of sdcc or else you might face some unwanted issues while using sdcc for building the project. Also note that instead of the above procedure, if the sdcc compiler is directly installed from the "Software Center" in linux, user might face assembling and other issues.

After successfully installing the sdcc compiler, we can now download the FX3 SDK provided by Cypress.

4) Download the FX3 SDK(which works for FX2 as well) from

<http://www.cypress.com/?rID=57990>

5) Unzip the package and inside the unzipped directory, further unzip the "cyusb_linux_1.0.4" package which inside its bin directory contains the "cyusb_linux" utility (similar to Control Center used for Windows) that is used to program the FX2. Instructions on installation of cyusb_linux are given in the "cyusb_linux_user_guide.pdf" present at ".\cyusb_linux_1.0.4\docs".

6) Once the cyusb_linux is installed, we need to generate the slave.hex file for programming the FX2LP device.

After this step, all the further steps are same as those used in the Windows section (Step#3 onwards) except that instead of the script file "commands.bash" mentioned in the Step#7, in Linux we need to run the "commands.sh" file using the command `bash commands.sh`.

File Structure

Apart from the AN61345 project if the user wants to use the sdcc compiler for building any other project, then he/she needs to have a 2 sets of files 1) which are necessary and remain unchanged for any project and 2) project specific files.

Given below are the names of the files that remain unchanged :

1. delay.c
2. discon.c
3. EZRegs.c
4. getstrd.c
5. I2c.c
6. I2c_rw.c
7. resume.c
8. fw.c
9. Fx2.h
10. syncdly.h
11. fx2regs.h
12. susp.asm
13. delayms.asm
14. USBImpTb.asm

The files which are project specific and hence need to be changed for a future project are as under :

1. dscr.asm
2. slave.c

Keywords Changes

For using sdcc compiler for building a project, certain keywords like xdata, db, used inside some of the project files (taken from Windows project) need to be changed.

One can refer to the Section SDCC Technical Data >> ANSI-Compliance section of the SDCC Manual “sdccman” present at “\.\AN61345 - Source files for FPGA code and FX2LP Firmware\001-61345” for the list of keywords that are different in case of SDCC compiler.

In case of the AN61345 project, the files inside which these keywords were changed include :

1. fw.c
2. Fx2.h
3. Slave.c
4. fx2regs.h
5. slave.c
6. syncdly.h

Apart from these keywords changes, certain other changes could also be seen in the above files and similar changes need to be made to the files of any other project which the user wants to build using sdcc.