

Tutorial - DLX Simulator

Customized Embedded Processor Design

Application Specific Instruction-Set Processors- ASIP
Lab (Praktikum)

Responsible/Author:
MSc. Sajjad Hussain

Supervisors:

MSc. Sajjad Hussain, Dr.-Ing. Lars Bauer, Prof. Dr.-Ing. Jörg Henkel

Chair of Embedded Systems,
Building 07.21, Haid-und-Neu-Str. 7,
76131 Karlsruhe, Germany.

November 2, 2023

DLX SIMULATOR -TUTORIAL

Simulation using DLXSim

A. Simulating an Assembly file:

1. Login to any *i83labpcXX.itec.kit.edu* directly or using SSH or using X2Go Client. For example login as *asip-sajjad04* into *i83labpc02.itec.kit.edu*
2. Open shell terminal from the start menu. It should be in your default home directory. Go to the directory “ */ASIP_SS17/Session1/ASIPMeisterProjects/brownie:\$*”
3. Set the proper path and parameters in “env_settings” like dlxsim path, project path and project name.
4. Go to the application directory, for example: “ */ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:\$*” and type “*make clean*” clean this directory if there are previously generated files.

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make clean
/bin/rm -rf BUILD_SIM BUILD_FPGA
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$ls
1_Arith.s Makefile
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$
```

5. As this application subdirectory contains .s file, you can directly simulate it using “*make dlxsim*” without compiling it. If this application has .c file, then you have to compile it using “*make sim*”. For example to load “*1_Arith.s*” and using no forwarding, use the following parameters. A directory “*BUILD_SIM*” is created which contains different temporary files and a .dlxsim file to be simulated in dlxsim (in this case it is “*Arith.dlxsim*”).

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
➔ dlxsim DLXSIMPARAM="-f1_Arith.s -da0 -pf0"
```

```
Transforming file "1_Arith.s" for target SIMULATION.
```

```
Assembling/Linking for target SIMULATION:
```

```
Creating combined files.
```

```
STACK_START: 0xFFFFFC
```

```
FINISHED ASSEMBLING/LINKING for target SIMULATION.
```

```
Starting dlxsim:
```

```
/Software/epp/dlxsim_Laboratory/dlxsim -fBUILD_SIM/Arith.dlxsim -f1_Arith.s -da0 -pf0
Biggest used address for Text Section (word aligned): 0x1c
Biggest used address for Data Section (word aligned): 0x0
(dlxsim)
```

6. Then in dlxsim you can use “go” or “step” command to simulate all instructions or each instruction step by step respectively.

```
(dlxsim) step
stopped after (single) step, pc = _main+0x04 (0x0004): addi r2,r0,0x9
(dlxsim) step
stopped after (single) step, pc = _main+0x08 (0x0008): or r3,r1,r2
(dlxsim) get r2
r2: 0x00000009
(dlxsim) go
TRAP #0 received
Altogether 41,0e0(41) cycles executed.
0 Warnings for unresolved data dependencies printed.
0 Warnings for successive load/store commands printed.
0 Warnings for load/stores in the text section printed.
(dlxsim)
```

7. You can see different statistics using “**stats**” command.
8. Enter “**quit**” command to exit from dlxsim simulator.

B. Simulating a C file:

9. If the application consists of C files then you can use “**make sim**”, which will compile your application into assembly file and automatically starts dlxsim. The other steps remain the same. Remember, “**make sim**” only works if you have already created Compiler.

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
sim
```

10. You can have different parameter to “**make sim**” like optimization identifier and number of NOPS added for simulating your application in hardware.

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
sim GCC\PARAM=-O3
```

11. You can now start dlxsim simulation using following different commands:

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
↪ dlxsim GCCPARAM=-O3
```

OR

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
dlxsim DLXSIMPARAM="-fBUILD_SIM/Arith.dlxsim -da0 -pfl"
```

12. You can save dlxsim simulation output to different file using “**-lf**”, “**-uf**”, or “**-af**” for LCD, UART or audio respectively as following:

```
asip04@i83labpc04:~/ASIP_SS17/Session1/ASIPMeisterProjects/brownie/Applications/Arith:$make
dlxsim DLXSIMPARAM="-fBUILD_SIM/Arith.dlxsim -da0
-pfl -lfoutput_dlxsim.txt"
```