Laboratory Practice

Design of a 4 bit CPU

Writing Program Code

During the practice the task is to create own program code running on the CPU. The instruction memory of the CPU (INST_MEM) is described initially by a VHDL code (INST_MEM.vhd). It is possible to overwrite the content of given cells manually with the hexadecimal code of the instructions. This way it is possible to edit the instructions in the memory of processor.

The easiest way to create program code is to define them with their mnemonics. The code written this way should be processed by a program, which compiles it to binary code. The program code will be created during laboratory practice with the program **dt4bit_CPU_assembler**[1] (Fig. 1.). This program is capable to edit the program code, to compile it to binary code and to replace the default content of the program memory. After the content of program memory is changed (INST_MEM.vhd), then the project has to be translated again with the ISE Webpack. This way there is possibility to simulate the operation and to upload the configuration file to the FPGA. The FPGA programmed this way will contain in its program memory the modified program code.

On the editorial interface can the program code be edited, this is where the instructions have to be defined. The accepted instructions can be fined in the section **List of accepted instructions** (Elérhető utasítások listája). On the editorial interface only the following characters may be used:

- lower case letters of English alphabet (a..z),
- numbers (0..9),
- colon, to assign label (:),
- tabulator to separate instruction (TAB),
- enter to separate line (ENTER).

Lines should be separated the following way:

label: ⟶ instruction ⟶ argument

(címke: ⟶ utasítás ⟶ argumentum)

[1] The program is at the moment available only in Hungarian. Therefore, the Hungarian texts are listed everywhere with their translations. Also a brief dictionary can be found at the end of the document, where translation of the most important phrases can be found.