

8-bit ALU

K Raja Pradyumna and G V V Sharma*

CONTENTS

1	Model outline	1
2	Compiling the Assembly code	1
3	Uploading the verilog models to IcoBoard	1

Abstract—This manual explains how to run a minimalistic 8-bit CPU on an IcoBoard.

1 MODEL OUTLINE

The CPU model implementation is based on the Intel8085 but it is quite minimal in the sense that it is being designed only as a wrapper to the ALU to test its functionality. A standard Intel8085 Instruction consists of the instruction code, followed by the input parameters.

ADD R1

The above instruction adds the value in the Register R1 to the values in the register in the accumulator and stores the resulting sum into the accumulator.

The ALU being implemented performs 20 different operations and these operations are called the outer CPU module which provides the operands and OP codes to the ALU and displays the results to the user.

2 COMPILING THE ASSEMBLY CODE

2.1 Download the following assembly code

<https://github.com/pradyuishere/8-bit-alu/blob/master/hello.txt>

2.2 Use the following code to compile the assembly code downloaded above and check for the generated out.txt file.

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadevall@iith.ac.in. All solutions in this manual is released under GNU GPL. Free and open source.

<https://github.com/pradyuishere/8-bit-alu/blob/master/compiler.py>

3 UPLOADING THE VERILOG MODELS TO IcoBOARD

- 3.1 Clone the following repository Github link and verify the output generated in the out.txt file and the lines 83 through 94 of cpu.v are the same. If not, copy the out.txt to the above lines in the file.

<https://github.com/pradyuishere/8-bit-alu>

- 3.2 Now, the task is to connect the inout pins of the IcoBoard with the arduino as mentioned in the combined.pcf and the arduino codes. Also connect the 16x2 lcd display to the Arduino following the pin numbering in the Arduino codes.

clk R9

```
rst A11
data_in[7] A5
data_in[6] A2
data_in[5] C3
data_in[4] B4
data_in[3] D8
data_in[2] B9
data_in[1] B10
data_in[0] B11
```

```
data_out[7] N6
data_out[6] L7
data_out[5] G5
data_out[4] L9
data_out[3] R10
data_out[2] T11
data_out[1] T14
data_out[0] T15
```

```
next_in B7
output_done R14
```

- 3.3 Now, once the pin connections are made, we upload the logic to the IcoBoard by running the following command.

```
make v_fname=combined
```

- 3.4 Now, check the output on the 16 by 2 lcd and verify the results. The frequency of displaying the results can be modified in the combined.v file.