Lab 12

Single Cycle MIPS Datapath

(R-type and Load/Store)

Learning objective of this lab

* Adding ALU and Control Unit to the previous module
* Pre-Lab

The following two modules are already connected in the previous Labs

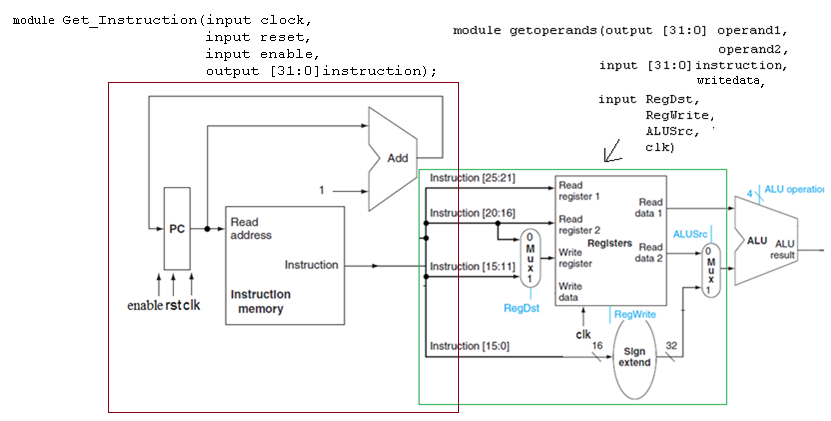


Figure 12.1

* 1. Today’s Tasks
     1. ALU

Now connect the previous block to ALU.. The ALU is shown by diagram below

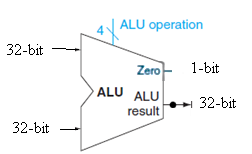


Figure 12.2

****

Table 12.1

It has two 32-bit data inputs, a 4-bit Control input ‘ALU operation’, one 32-bit output and one 1-bit output (Zero). The 1-bit output is useful for jump instruction such as SLT. It is used in the subtract instruction. If the result of subtraction is zero it is 1,otherwise it is zero.

* + 1. TestBench

The final output of the complete circuit is given in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Program Counter | Data Read from Instruction Memory | | Output of Complete Block  **‘Read data 1’** & **‘Read data 2’** | If ALU control lines =0010 |
| Actual Instruction | Data Read |
| 0 | add $9, $12, $13 | 018d4820 | Value of $12 and $13 | $12+$13 |
| 1 | add $10, $14, $15 | 01cf5020 | Value of $14 and $15 | $14+$15 |
| 2 | sub $8, $10, $9 | 1494022 | Value of $10 and $9 | $10+$9 |

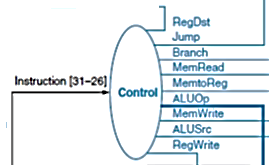
Table 12.2

* + 1. Control

Write module for Control Unit and ALU Control, and name it **Control**. Connect it with the previous blocks.

* + - 1. **Control Unit**

Note that for each instruction type, bits 26-31 are allocated for opcode which is used by the Control Unit to generate the relevant control signals for the instructions. Table given below explains the input to output relationship of control unit. Table for our complete implementation including branch and jump instructions is given here, so the control signals ‘Branch’ and ‘Jump’ will not be used today

****

**Figure 12.3**

****

Table 12.3

* + - 1. **ALU Control**

It uses 2-bit ALUOP and 4-bits of ‘funct’ in R-type instructions to generate 4-bit output.(Use casex)

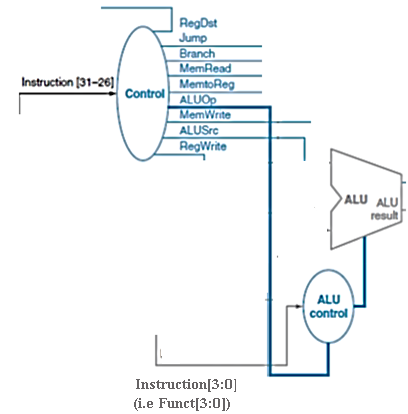


Figure 12.4

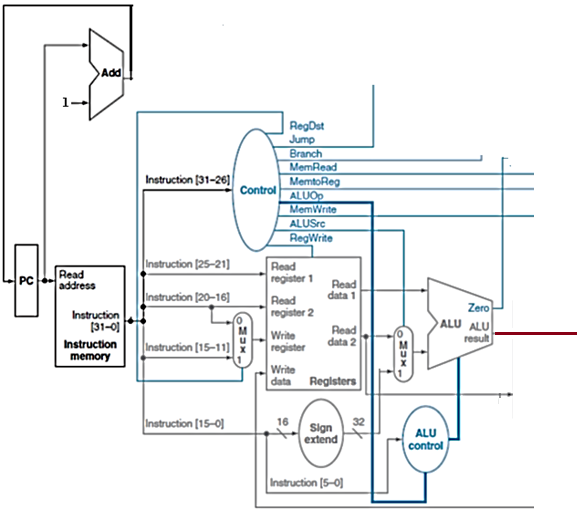


Table 12.4

Its output is input to ALU. Based on its output ALU decides the operation to be performed (add, sub, etc)

* + - 1. **Test Bench**

Figure 12.5 shows all the connected blocks. Write it’s testbench. ALU is it’s output.

****

**Figure 12.5**

* In-Lab

**Today’s Tasks**

* **Write ALU HDL description**
* **Combine and test after combining with previous Labs**
* **Write Control Unit HDL description**
* Post-Lab
* **Combine both with previous blocks and write Testbench**

**Submission details**

* **Your lab report, a .doc file, should contain properly commented Post-Lab task code, with Screenshots(of print preview) of Schematic and waveforms, and Critical Analysis.**
* **The report must have a title page in the pescribed format.**
* **Name the .doc file RegNo.docx; eg SP14-BCE-99.docx**
* **Sumbit on portal.**