Single Cycle CPU Architecture

The architecture contains a total of 9 modules:

1. IMEM : Accepts instruction address from CPU and sends instruction value to the CPU.

2. DMEM : Sends the value stored in memory to the CPU.

3. Register File : Sends values from registers addressed by rs1 and rs2 to the CPU.

4. CPU : This acts like the control unit and sends the required signals to the remaining modules. It detects the type of instruction from the opcode and selects the required output using a multiplexer from the outputs of the following 5 modules. Since JAL, JALR, LUI and AUIPC are instructions with unique opcodes, the CPU generates the required output within itself without depending on an external module.

5. R_type : This module is used for the R - type instructions. It accepts idata, rv1 and rv2 from the CPU and uses the value of funct3 and idata[30] as select lines to a multiplexer to choose the specific R_type instruction and assign the value of output (regdata_R) based on the correct ALU operation. This is sent to the CPU.

6. I_type : This module is used for the I - type instructions. It accepts idata, rv1 and imm from the CPU and uses the value of funct3 as select lines to a multiplexer to choose the specific I_type instruction and assign the value of

output (regdata_I) based on the correct ALU operation. This is sent to the CPU.

7. L_type : This module is used for the Load type instructions. It accepts idata, daddr and drdata from the CPU and uses the value of funct3 as select lines to a multiplexer to choose the specific L_type instruction. Based on this, the 32 bit value of the output (regdata_L) is assigned with or without sign extension based on the instruction. The output is sent to the CPU which sends it to the Register file as the data to be written.

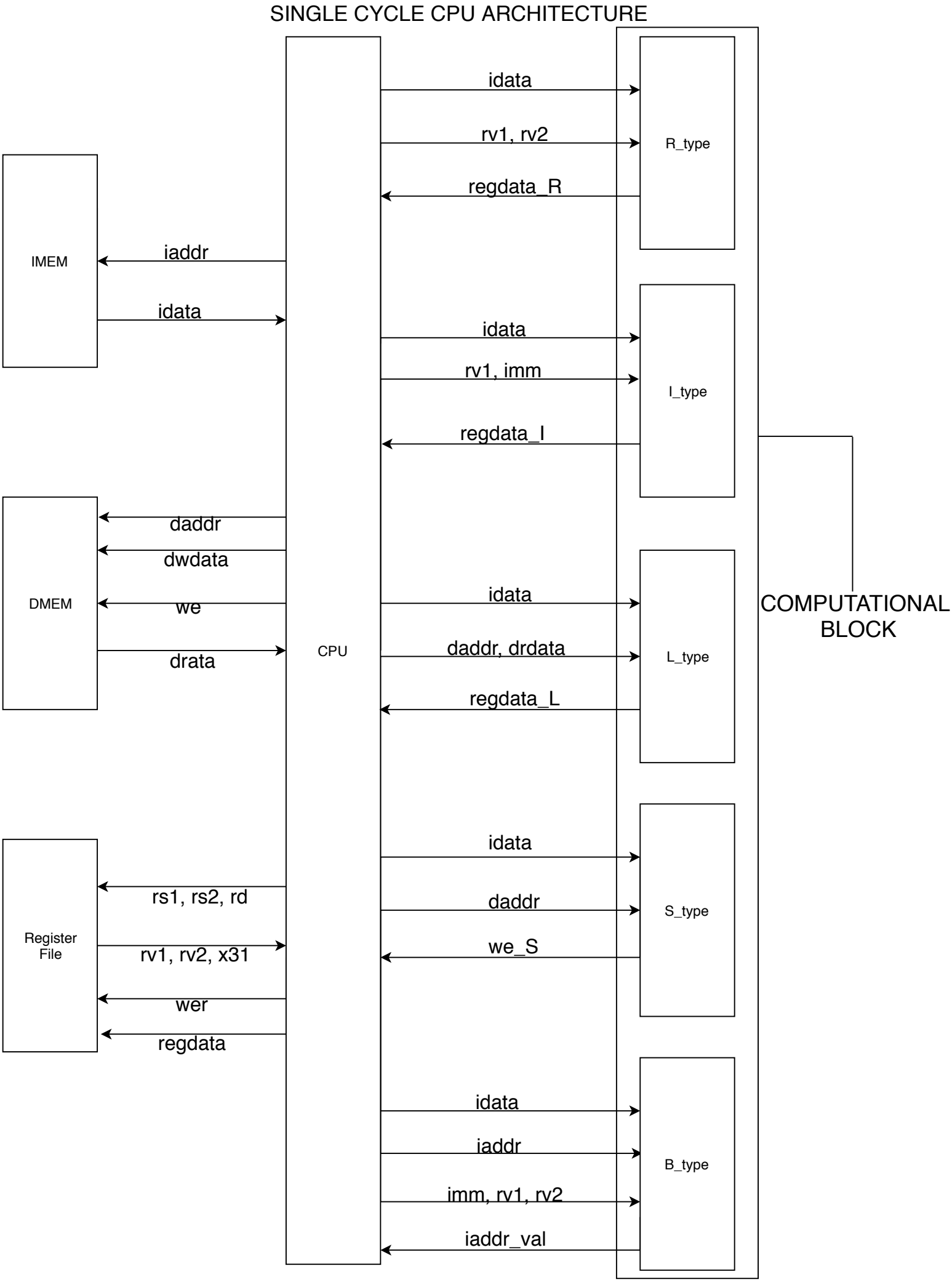
8. S_type: This module is used for the Store type instructions. It accepts idata, daddr and we_S from the CPU and uses the value of funct3 as select lines to a multiplexer to choose the specific S_type instruction. Based on this, the 4 bit value of the write enable signal (we_S) is assigned depending on whether a word, half word or a byte is to be written to DMEM. we_S is sent to the CPU.

9. B_type: This module is used for the Branch type instructions. It accepts idata, iaddr, imm, rv1 and rv2 from the CPU and uses the value of funct3 as select lines to a multiplexer to choose the specific B - type instruction. Based on this, the comparison is performed and the value of the output (iaddr_val) is assigned as pc+offset or pc+4 as required. The output is sent to the cpu which assigns the value of PC accordingly.

The Computational block shown in the block diagram on the next page consists of the R_type, I_type, L_type, S_type and B_type modules. It is the block that takes care of computing the

required values depending on the instruction and returning appropriate signals to the CPU.

Note: The CPU outputs iaddr and PC are both related to the program counter value. iaddr gives the current Program counter value and PC gives the value that the Program counter will take in the next clock cycle.


SINGLE CYCLE CPU ARCHITECTURE
IMEM
iaddr
idata
DMEM
daddr
dwdata
we
drata
Register File
rs1, rs2, rd
rv1, rv2, x31
wer
regdata
CPU
idata
rv1, rv2
regdata_R
R_type
idata
rv1, imm
regdata_I
I_type
idata
daddr, drdata
regdata_L
L_type
idata
daddr
we_S
S_type
idata
iaddr
imm, rv1, rv2
iaddr_val
B_type
COMPUTATIONAL BLOCK