|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| instruction\_1\_i | logic[31:0] | Input for instruction 1 |
| instruction\_2\_i | logic[31:0] | Input for instruction 2 |
| data\_out\_o | logic[31:0] | Output |
| pc\_o | Logic[4:0] | Program Counter |

Register renaming (look up table, which handles the WAW and WAR hazards)

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| Instruction\_1\_i | Logic[31:0] | Instruction 1 for renaming |
| Instruction\_2\_i | Logic[31:0] | Instruction 2 for renaming |
| Instruction\_1\_o | Logic[31:0] | Instruction 1 after renaming |
| Instruction\_2\_o | Logic[31:0] | Instruction 2 after renaming |
| Rename\_full\_o | Wire | When there is no more register available for renaming, this signal is set to high and sent back to the previous pipeline stage to stop getting new instructions for renaming. |
|  |  |  |

Issue queue (handles RAW hazard)

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| Instruction\_1\_i | Logic[31:0] |  |
| Instruction\_2\_i | Logic[31:0] |  |
| flush\_en | Wire | When this signal is high, flush everything stored in the issue queue. |
| Instruction\_1\_o | Logic[31:0] | Can issue at most 4 instructions every cycle. |
| Instruction\_2\_o | Logic[31:0] |  |
| Instruction\_3\_o | Logic[31:0] |  |
| Instruction\_4\_o | Logic[31:0] |  |
| Issue\_queue\_full | Wire | When full, set this signal to high. |
| issue\_queue\_empty | Wire | When empty, set this signal to high. |
|  |  |  |

Register Signal Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| instruction\_1\_read\_reg\_1\_i | logic[4:0] | Read register 1 for instruction 1 |
| instruction\_1\_read\_reg\_2\_i | logic[4:0] | Read register 2 for instruction 1 |
| instruction\_2\_read\_reg\_1\_i | logic[4:0] | Read register 1 for instruction 2 |
| instruction\_2\_read\_reg\_2\_i | logic[4:0] | Read register 2 for instruction 2 |
| instruction\_1\_write\_reg\_i | logic[4:0] | Write register for instruction 1 |
| instruction\_2\_write\_reg\_i | logic[4:0] | Write register for instruction 2 |
| instruction\_1\_write\_data\_i | logic[31:0] | Write data for instruction 1 |
| instruction\_2\_write\_data\_i | logic[31:0] | Write data for instruction 2 |
| instruction\_1\_read\_data\_1\_o | logic[31:0] | Read data out 1 for instruction 1 |
| instruction\_2\_read\_data\_1\_o | logic[31:0] | Read data out 1 for instruction 2 |
| instruction\_1\_read\_data\_2\_o | logic[31:0] | Read data out 2 for instruction 1 |
| instruction\_2\_read\_data\_2\_o | logic[31:0] | Read data out 2 for instruction 2 |
| enable\_1\_i | logic | Enable for instruction register 1 |
| enable\_2\_i | logic | Enable for instruction register 2 |

Data Memory Signal Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| instruction\_1\_addr\_i | logic[31:0] | Address for instruction 1 |
| instruction\_2\_addr\_i | logic[31:0] | Address for instruction 2 |
| instruction\_1\_write\_i | logic[31:0] | Write data for instruction 1 |
| instruction\_2\_write\_i | logic[31:0] | Write data for instruction 2 |
| instruction\_1\_read\_o | logic[31:0] | Read data out for instruction 1 |
| instruction\_2\_read\_o | logic[31:0] | Read data out for instruction 2 |
| enable\_read\_1\_i | logic | Enable for instruction 1 read memory |
| enable\_read\_2\_i | logic | Enable for instruction 2 read memory |
| enable\_write\_1\_i | logic | Enable for instruction 1 write memory |
| enable\_write\_2\_i | logic | Enable for instruction 1 read memory |

Control Unit Signal Table

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| op\_1\_i | logic[5:0] | Opcode for instruction 1 |
| op\_2\_i | logic[5:0] | Opcode for instruction 2 |
| regDest\_1\_o | logic | Control bit for the mux to write register |
| regDest\_2\_o | logic | Control bit for the mux to write register |
| aluSrc\_1\_o | logic | Control bit for the mux to the ALU |
| aluSrc\_2\_o | logic | Control bit for the mux to the ALU |
| memToReg\_1\_o | logic | Control bit for the mux for the write back memory to register |
| memToReg\_2\_o | logic | Control bit for the mux for the write back memory to register |
| regToWrite\_1\_o | logic | Control bit to enable register |
| regToWrite\_2\_o | logic | Control bit to enable register |
| memRead\_1\_o | logic | Enable memory for read |
| memRead\_2\_o | logic | Enable memory for read |
| memWrite\_1\_o | logic | Enable memory for write |
| memWrite\_2\_o | logic | Enable memory for write |
| branch\_1\_o | logic | Goes High on branch, else goes low |
| branch\_2\_o | logic | Goes High on branch, else goes low |
| aluOp\_1\_o | logic[1:0] | The ALU operations |
| aluOp\_2\_o | logic[1:0] | The ALU Operations |