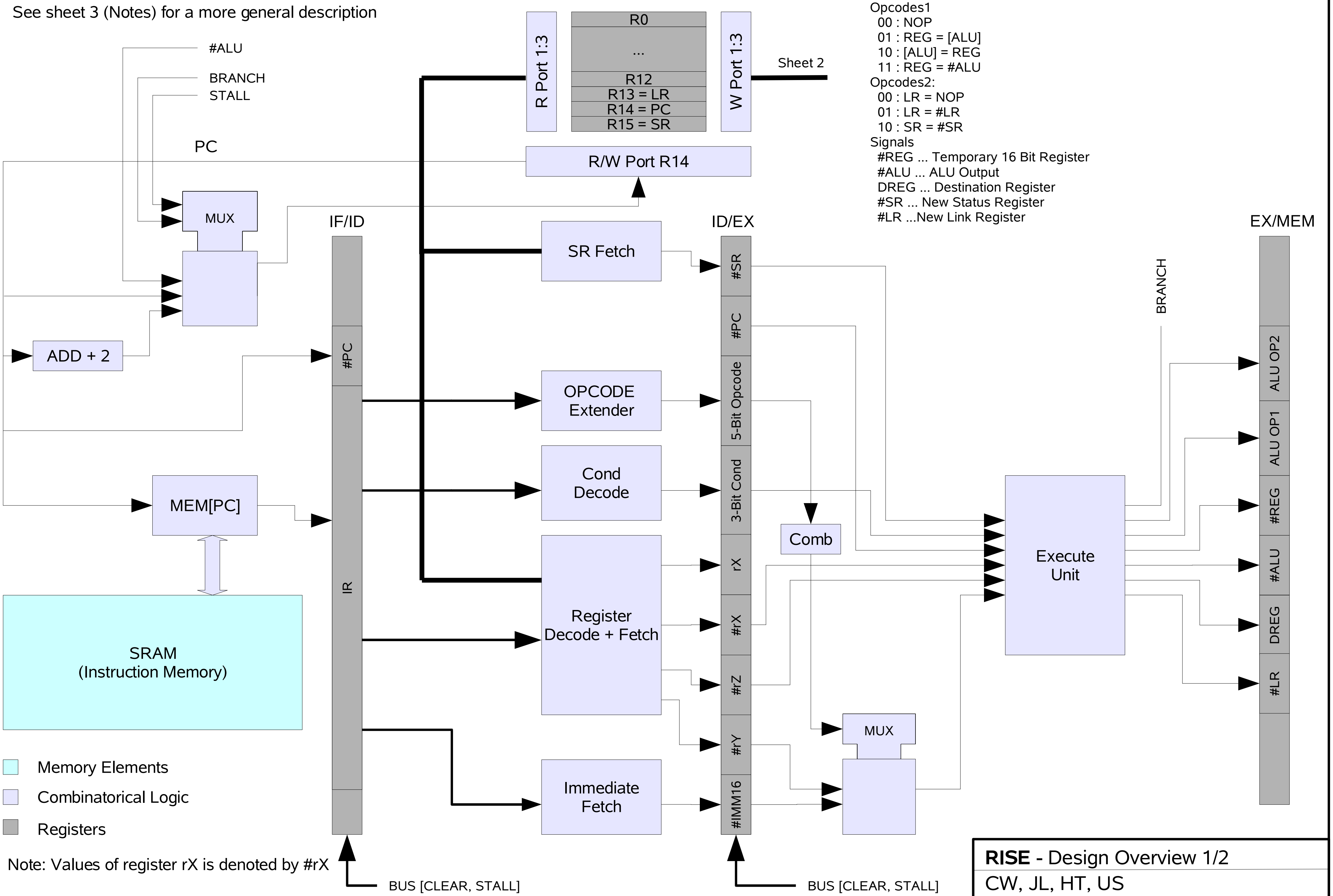
See sheet 3 (Notes) for a more general description



Opcodes1
00 : NOP
01 : REG = [ALU]
10 : [ALU] = REG
11 : REG = #ALU
Opcodes2:
00 : LR = NOP
01 : LR = #LR
10 : SR = #SR
Signals
#REG ... Temporary 16 Bit Register
#ALU ... ALU Output
DREG ... Destination Register
#SR ... New Status Register
#LR ...New Link Register

🟦 Memory Elements

⬜ Combinatorical Logic

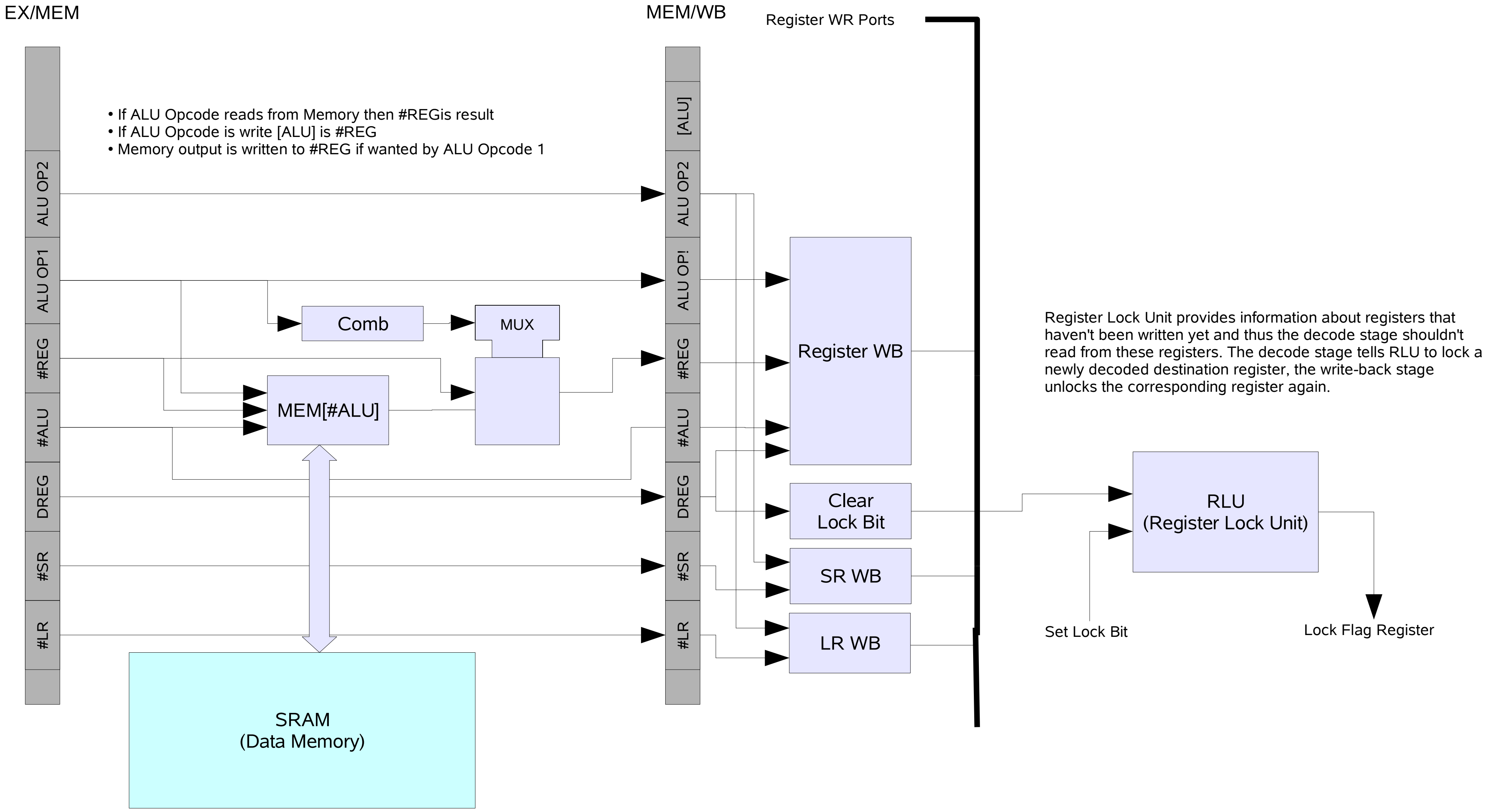
⬛ Registers

Note: Values of register rX is denoted by #rX

**RISE** - Design Overview 1/2

CW, JL, HT, US

21. November 2006 1/2



- If ALU Opcode reads from Memory then #REGis result
- If ALU Opcode is write [ALU] is #REG
- Memory output is written to #REG if wanted by ALU Opcode 1

Register Lock Unit provides information about registers that haven't been written yet and thus the decode stage shouldn't read from these registers. The decode stage tells RLU to lock a newly decoded destination register, the write-back stage unlocks the corresponding register again.

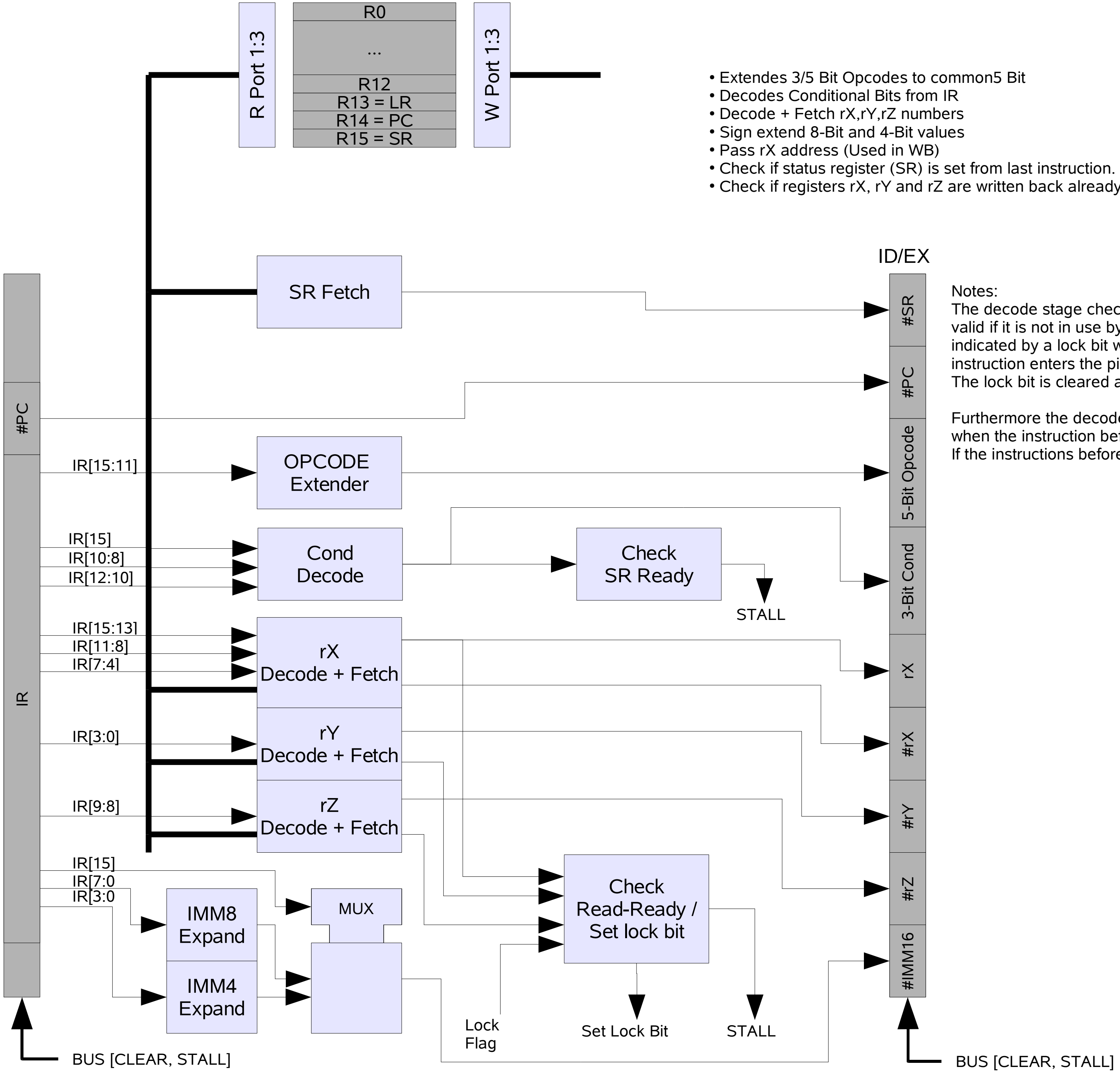
**RISE** – Design Overview 2/2

CW, JL, HT, US

21. November 2006

2/4

- Extendes 3/5 Bit Opcodes to common5 Bit
- Decodes Conditional Bits from IR
- Decode + Fetch rX,rY,rZ numbers
- Sign extend 8-Bit and 4-Bit values
- Pass rX address (Used in WB)
- Check if status register (SR) is set from last instruction. If not, stall the pipeline
- Check if registers rX, rY and rZ are written back already. If not, stall the pipeline.



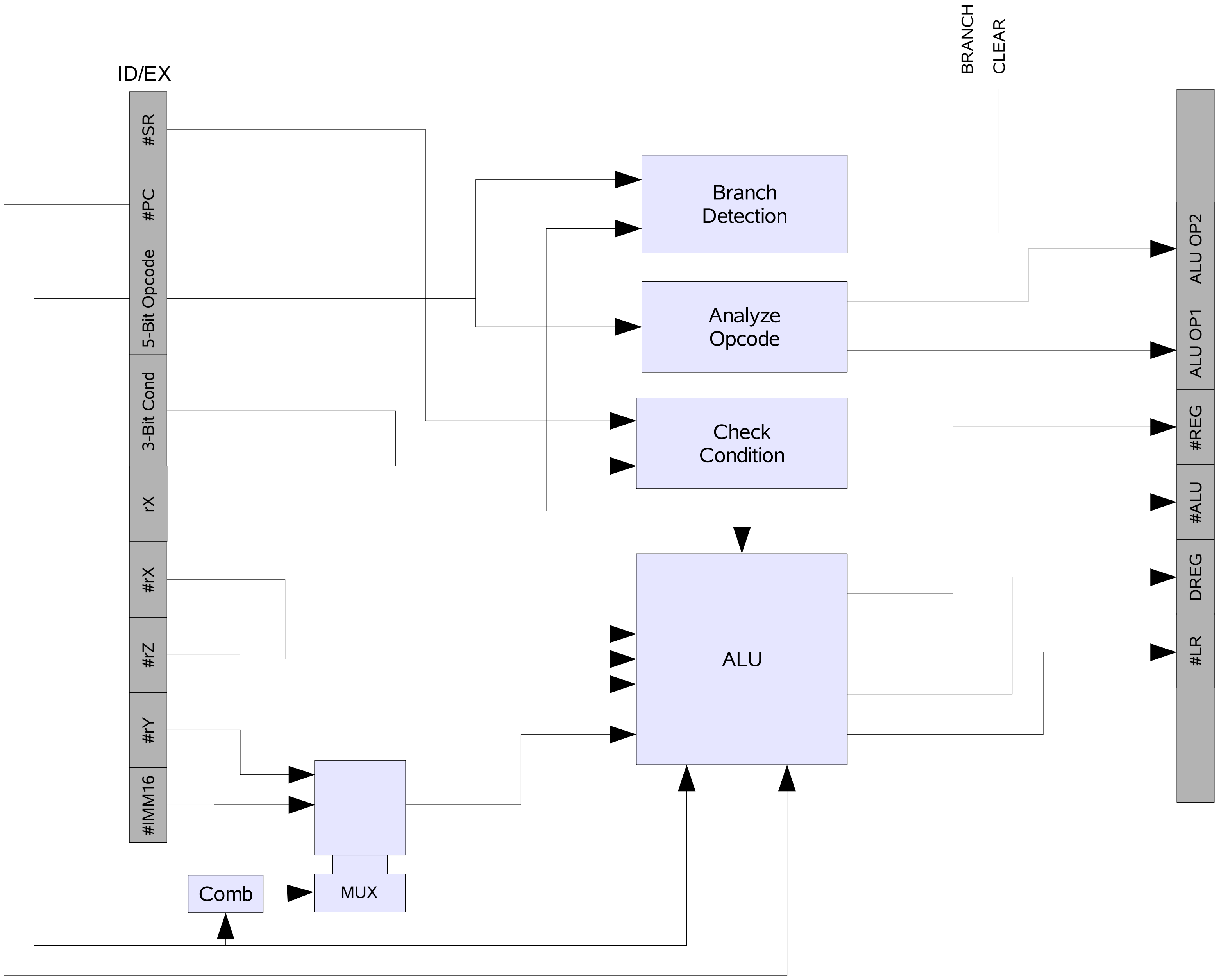
Notes:

The decode stage checks if all registers are valid and can be read. A register is valid if it is not in use by an already pending instruction in the pipeline. This is indicated by a lock bit which is set by the instruction decode stage when the instruction enters the pipeline. If a register is not ready then the pipeline is stalled. The lock bit is cleared again in the writeback stage.

Furthermore the decode stage checks if the status register is ready. It is ready when the instruction before this instructions before has updated the status register. If the instructions before do not modify the SR no stall is necessary.

| | |
|---|---|
| **RISE** – Decode Stage | |
| CW, JL, HT, US | |
| 21. November 2006 | 1/1 |

Notes:
The execution stage checks if there is a branch. If there is a branch it clears the previous two pipeline stages and updates the programm counter in the register file.



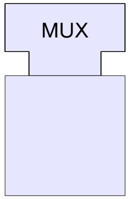
**RISE** – Execute Stage

CW, JL, HT, US

21. November 2006 1/1

Pipeline

- If a write to PC  is detected then the branch signal is generated. In this case the content of the pipeline in the IF/ID and ID/EX is cleared.
- If a operation is conditional it must be stalled until the previous SR is available in the register file.



**RISE** - Symbols

Christian Walter

| | |
|---|---|
| 21. November 2006 | 1/2 |