Final: Lina Mi @01337283

Work out the values of Control Signals for instructions: `andci rt,rs,`*Immediate*

The instruction output= A&(!B)

This instruction takes the A operand from register rs and the B operand from the (sign-extended) lower 16 bits of the instruction, and stores the result in register rt.

Data flowing path analysis:

1) the PC points to the instruction memory location holding the instruction, and the instruction memory fetches the instruction.
2) The register fill reads operand A specified by $instr_{25:21}$ on port A1, reads operand B (Immediate) specified by $instr_{15:0}$, since the SrcB comes from immediate, so the **ALUSrc**=1;
3) The **ALUControl** = 100 for instruction andci
4) Since the result comes from ALU, not from Memory, so **MemtoReg**=0;
5) The result should be write into the register file, so **RegWrite**=1;
6) the destination register is specified in the rt filed of the instruction($instr_{20:16}$), so **RegDst**=0;
7) since the result of the instruction should not be written to Memory, so **MemWrite**=0;
8) since the instruction does not *branch*, so ***branch***=0;
9) since the instruction does not jump, so **Jump**=0.

to summary, the control signals are listed in the following table:

Instruction	MemtoReg	MemWrite	Banch	$ALUControl_{2:0}$	ALUSrc	RegDst	RegWrite	Jump
Andci	0	0	0	100	1	0	1	0