

Lab Exercise #15 -- ARC+ Datapath and Control

The ARC+ microprocessor is derived from the SPARC microprocessor; however, it only recognizes a subset of the SPARC instructions.

Arithmetic: ADD, SUB, ADDX, SUBX, UMUL, SMUL, UDIV, SDIV
Logical: AND, OR, XOR, ANDN, ORN, XNOR
Shifting: SLL, SRL, SRA
Memory: LD, ST
Other: SETHI, Bicc, CALL, JMPL

In addition, it recognizes the "CC" versions of the arithmetic and logical instructions ("ADDcc", for example).

The datapath for a single-cycle (non-pipelined) implementation of the ARC+ is given in the accompanying diagram.

The contents of the control and general-purpose registers are shown below (in hexadecimal) at the end of the fetch phase of the instruction cycle.

nPC: 00010004	PC: 00010000	PSR: 00000000	Y: 00000000
R[00]: 00000000	R[08]: 08080808	R[10]: 10101010	R[18]: 18181818
R[01]: 01010101	R[09]: 09090909	R[11]: 11111111	R[19]: 19191919
R[02]: 02020202	R[0A]: 0A0A0A0A	R[12]: 12121212	R[1A]: 1A1A1A1A
R[03]: 03030303	R[0B]: 0B0B0B0B	R[13]: 13131313	R[1B]: 1B1B1B1B
R[04]: 04040404	R[0C]: 0C0C0C0C	R[14]: 14141414	R[1C]: 1C1C1C1C
R[05]: 05050505	R[0D]: 0D0D0D0D	R[15]: 15151515	R[1D]: 1D1D1D1D
R[06]: 06060606	R[0E]: 0E0E0E0E	R[16]: 16161616	R[1E]: 1E1E1E1E
R[07]: 07070707	R[0F]: 0F0F0F0F	R[17]: 17171717	R[1F]: 1F1F1F1F

For each of the following independent cases, assume that the IR (instruction register) contains the value shown (in hexadecimal) at the end of the fetch phase. Give the hexadecimal value of the requested signals. If the value of a signal cannot be determined from the information given, write "unknown".

a) IR: B40C8017 (AND instruction)

Output of MUX-A: __17171717__

Output of ALU: __12121212__

Output of MUX-B: __12121212__

Output of MUX-C: __00010008__

b) IR: A61560FF (OR instruction)

Output of MUX-A: __000000FF__

Output of ALU: __151515FF__

Output of MUX-B: __151515FF__

Output of MUX-C: __00010008__

- c) IR: 292A1908 (SETHI instruction)
- Output of MUX-A: __08080808__
- Output of ALU: __08080800__ (assuming ALU uses OP3 bits)
- Output of MUX-B: __A8642000__
- Output of MUX-C: __00010008__
- d) IR: E802A040 (LD instruction)
- Output of MUX-A: __00000040__
- Output of ALU: __0A0A0A4A__
- Output of MUX-B: __unknown__ (contents of data memory unknown)
- Output of MUX-C: __00010008__
- e) IR: C6228011 (ST instruction)
- Output of MUX-A: __11111111__
- Output of ALU: __1B1B1B1B__
- Output of MUX-B: __unknown__ (output of MUX-B not used for ST)
- Output of MUX-C: __00010008__
- f) IR: ABC0E080 (JMPL instruction)
- Output of MUX-A: __00000080__
- Output of ALU: __03030383__
- Output of MUX-B: __00010000__
- Output of MUX-C: __03030383__
- g) IR: 4000004A (CALL instruction)
- Output of MUX-A: __0A0A0A0A__
- Output of ALU: __0A0A0A0A__ (assuming ALU uses OP3 bits)
- Output of MUX-B: __00010000__
- Output of MUX-C: __00010128__
- h) IR: 1EBFFFF8 (BVC instruction)
- Output of MUX-A: __FFFFFFF8__
- Output of ALU: __1F1F1F18__ (assuming ALU uses OP3 bits)
- Output of MUX-B: __unknown__ (output of MUX-B not used for BVC)
- Output of MUX-C: __0000FFE0__