8BC2 Instruction Set

Machine Language

X = don't care

Instruction	binary	bytes	clocks	
ALU1[f] a,b,c?	000ABCDE	5/7	7/10	Perform ALU function, !Cn = 1, f is the hexadecimal value of ABCDE.
ALU0[f] a,b,c?	001ABCDE	5/7	7/10	Perform ALU function, f is the hexadecimal value of ABCDE.
TC a,b	010XXXXX	5	7	Test two numbers and set condition codes.
LPC	011XXXXX	1	1	Store value of program counter in two input ports.
IN[p] h,l	100XXXXX	4	5	Read from an input port p.
OUT[p] h,l	101XXXXX	4	5	Write to an output port p.
RETI	110XXXXX	1	4	Return from interrupt.
JC[c] h,l	111XXABC	5	7	Conditional jump, !Cn = 1, c is the condition on which to jump (number or symbols).

If !Cn is not noted as high, then !Cn is low. 'a', 'b', 'c', 'h', and 'l' are each two bytes. 'p' is single byte, 'f' is the lower five bits (ABCDE) of the ALU instruction bytes, and 'c' is the lower three bits (ABC) of the JC instruction byte.

Computer uses 74181 ALU. 74181's connection when using ALU instructions is:

M = A   S3 = B   S2 = C   S1 = D   S0 = E

When using JC instruction, ABC determines the condition on which to jump.

A	B	C	Symbol	Condition
0	0	0	0	Don't jump
0	0	1	=	A = B
0	1	0	<	A < B
0	1	1	<=	A ≤ B
1	0	0	>	A > B
1	0	1	>=	A ≥ B
1	1	0	!=	A ≠ B
1	1	1	1	Jump

Assembly Language

The above syntax (lines 4-14) is acceptable.

Potential Macros:

ALU[f] a,b,d ‘f’ can be a number (same as ALU0/1) or a symbol (+, -, ^, &, etc.). ‘c’ may or may not be used depending upon f. The last letter is always the destination address.

IF[acb] d,e Combination of TC and JC instructions. ‘c’ is condition on which to jump.

SUB pch,pcl,h,l Save value of program counter in “pch” and “pcl” and jump to the contents of ‘h’ and ‘l’.

RETS h,l Return to instruction after jumping to a subroutine. Address should have been stored in ‘h’ and ‘l’ during SUB instruction.

END Assembler instruction that starts the conversion of assembly to machine code.

‘a’, ‘b’, ‘d’, ‘e’, “pch”, “pcl”, ‘h’, and ‘l’ each represent two bytes. ‘c’ is the three bit condition code.