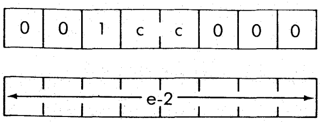
JR cc, e

Jump e relative on condition.

Function: if cc true, PC ← PC + e

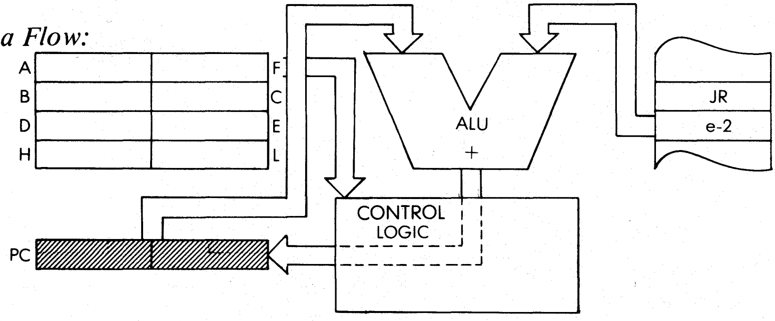
Format:



Description: **If the specified condition is met, the given offset value is added to the program counter using two's complement arithmetic so as to enable both forward and backward jumps. The offset value is added to the value of PC + 2 (after the jump). As a result, the effective offset is −126 to +129 bytes. The assembler automatically subtracts the value of PC + 2 from the source offset value to generate the hex code. If the condition is not met, the offset value is ignored and instruction execution continues in sequence. cc may any one of:**

NZ	00	NC	10
Z	01	C	11

Data Flow:



Timing:

	M cycles:	*T states:*	*usec* *@ 2 MHz:*
condition met:	3	12	6
condition not met:	2	7	3.5

Addressing Mode: Relative.

Byte Codes:

cc:	NZ	Z	NC	C
	20	28	30	38

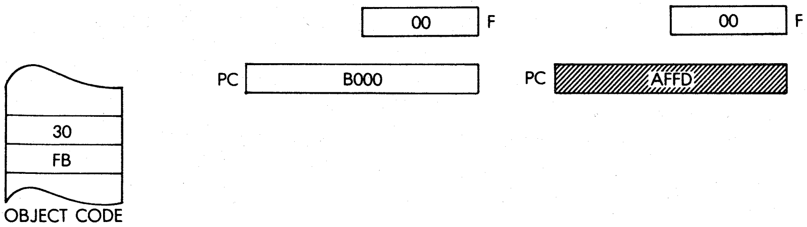
Flags:

S	Z		H		P/V	N	C

(no effect).

Example: JR NC, $ − 3 $ = current PC

Before: After:


00
F
PC
B000
00
F
PC
AFFD
30
FB
OBJECT CODE