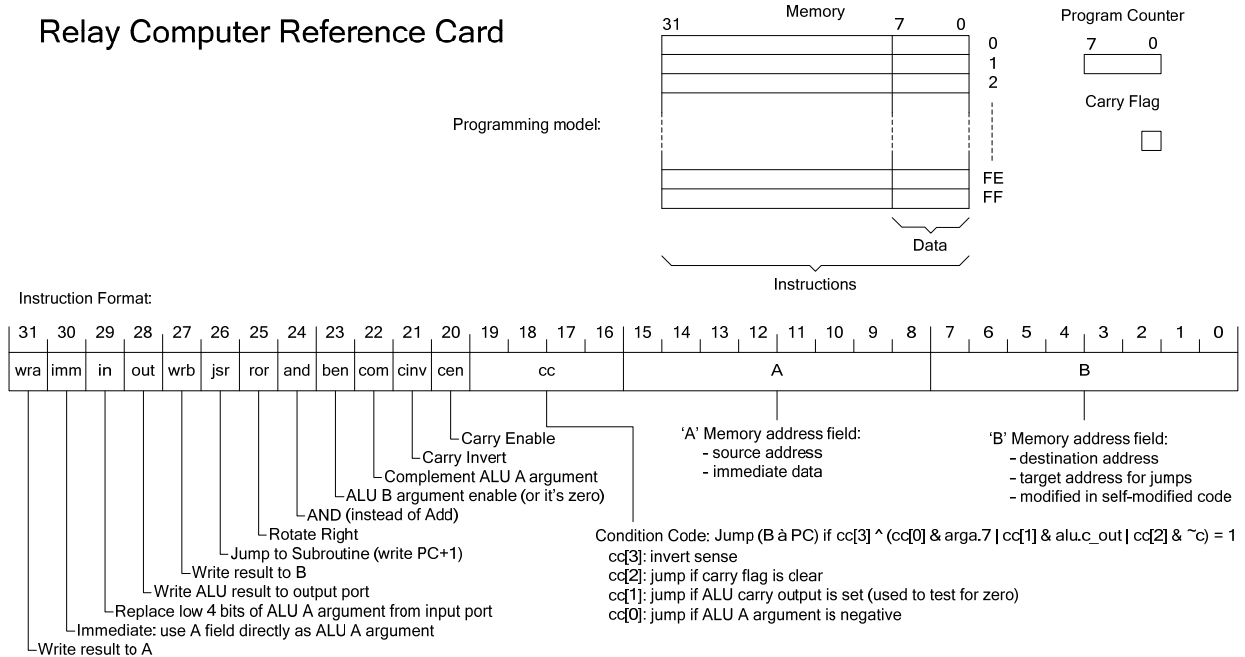


Relay Computer Reference Card



OP CODE	Mnemonic	Operation	OP CODE	Mnemonic	Operation
4010_FF00	NOP	No operation	0880_aabb	ADDTO aa, bb	[aa] + [bb] → [bb]
C810_FF00	HALT	Halt CPU	4880_aabb	ADDTO #aa, bb	aa + [bb] → [bb]
0000_0000	CLC	0 → C	0890_aabb	ADCTO aa, bb	[aa] + [bb] + C → [bb]
4020_FF00	STC	1 → C	4890_aabb	ADCTO #aa, bb	aa + [bb] + C → [bb]
0080_bbbb	NTOC bb	[bb].7 → C	80E0_aabb	RSB aa, bb	[bb] - [aa] → [aa]
1000_aa00	OUT aa	[aa] → output port	08E0_aabb	RSBTO aa, bb	[bb] - [aa] → [bb]
5000_aa00	OUT #aa	aa → output port	48E0_aabb	RSBTO #aa, bb	[bb] - aa → [bb]
6800_00bb	IN bb	Input port → [bb]	08D0_aabb	RSBCTO aa, bb	[bb] - [aa] - ~C → [bb]
4800_00bb	CLR bb	0 → [bb]	48D0_aabb	RSBCTO #aa, bb	[bb] - aa - ~C → [bb]
4880_01bb	INC bb	[bb] + 1 → [bb]	0980_aabb	ANDTO aa, bb	[aa] & [bb] → [bb]
48E0_01bb	DEC bb	[bb] - 1 → [bb]	4980_aabb	ANDTO #aa, bb	aa & [bb] → [bb]
0840_bbbb	COM bb	~[bb] → [bb]	09C0_aabb	BICTO aa, bb	~[aa] & [bb] → [bb]
0860_bbbb	NEG bb	-[bb] → [bb]	49C0_aabb	BICTO #aa, bb	~aa & [bb] → [bb]
0850_bbbb	NGC bb	~[bb] + C → [bb]	4018_FFbb	JMP bb	bb → PC
0A10_bbbb	ROR bb	{ C, [bb] } → { [bb], C }	8408_aabb	JSR aa, bb	PC + 1 → [aa], bb → PC
0A00_bbbb	LSR bb	{ 0, [bb] } → { [bb], C }	0062_aabb	JEQ aa, bb	If [aa] == 0 then bb → PC
0A20_bbbb	LSRO bb	{ 1, [bb] } → { [bb], C }	006A_aabb	JNE aa, bb	If [aa] != 0 then bb → PC
0890_bbbb	ROL bb	{ [bb], C } → { C, [bb] }	0061_aabb	JMI aa, bb	If [aa] < 0 then bb → PC
0880_bbbb	LSL bb	{ [bb], 0 } → { C, [bb] }	0069_aabb	JPL aa, bb	If [aa] >= 0 then bb → PC
08A0_bbbb	LSLO bb	{ [bb], 1 } → { C, [bb] }	0063_aabb	JLE aa, bb	If [aa] <= 0 then bb → PC
0800_aabb	ST aa, bb	[aa] → [bb]	006B_aabb	JGT aa, bb	If [aa] > 0 then bb → PC
4800_aabb	ST #aa, bb	aa → [bb]	0064_00bb	JCC bb	If C == 0 then bb → PC
0860_aabb	NEGTO aa, bb	-[aa] → [bb]	006C_00bb	JCS bb	If C == 1 then bb → PC
0850_aabb	NGCTO aa, bb	~[aa] + C → [bb]	0066_aabb	JLS aa, bb	If C == 0 [aa] == 0 then bb → PC
0840_aabb	COMTO aa, bb	~[aa] → [bb]	006E_aabb	JHI aa, bb	If C == 1 && [aa] != 0 then bb → PC
0A00_aabb	LSRTO aa, bb	{ 0, [aa] } → { [bb], C }	020A_aabb	JE aa, bb	If [aa].0==0 then bb→PC
0A20_aabb	LSROTO aa, bb	{ 1, [aa] } → { [bb], C }	0202_aabb	JO aa, bb	If [aa].0==1 then bb→PC
0A10_aabb	RORTO aa, bb	{ C, [aa] } → { [bb], C }	802A_aabb	INCJNE aa, bb	[aa] + 1 → [aa]; if [aa] != 0 then bb → PC
8080_aabb	ADD aa, bb	[aa] + [bb] → [aa]	8022_aabb	INCJEQ aa, bb	[aa] + 1 → [aa]; if [aa] == 0 then bb→PC

OP CODE	Mnemonic	Operation	OP CODE	Mnemonic	Operation
E800_00bb	INWAIT bb	Halt until input port changes or keypad or serial console character available. Save data to bb.			
9800_aa00	OUTC aa	Print [aa] to console serial port	D800_aa00	OUTC #aa	Print aa to console serial port.

Missing Instructions

Bitwise OR

; Compute bitwise OR: $Y = Y \mid X$.

```

    bicto      x, y    ; Clear bits in y which are set in x
    addto      x, y    ; Add the bits which are set in x into y

```

Exclusive OR

; Compute Exclusive-OR: $Y = Y \wedge X$.

; This can be computed as follows : $Y + X - 2*(Y \& X)$

```

    st         y, tmp
    andto      x, tmp
    lsl        tmp
    addto      x, y
    rsbto      tmp, y

```

Assembly Pseudo-ops

; Comment

```

    org    0x10        ; Set location to 0x10

    skip  1            ; Reserve one memory location

    data  0x23         ; Emit 8-bit data byte

    insn  0x4010ff00   ; Emit 32-bit instruction

fred    equ    0x50    ; Set label to value

```