NTDS COMPILING SYSTEM (CS-1) RCH

REPERTOIRE OF MONO-CODES

MONO-CODE-w-	ALLIED OPERAND-r-	OPERATION PERFORMED
RSH	Q, A, AQ	SHIFT (r) RIGHT BY (y)
LSH	Q, A, AQ	SHIFT (r) LEFT BY (y)
ENT	Q, A, $B^{n**}C^{n**}$	(y) → r
	LP	L(y)(Q) → A
	Y+Q, Y−Q	(r) → A
STR	Q, A, B^{n**}, C^{n**}, A+Q, A−Q	(r) → y
	LP	L(A)(Q) → y, (A) i = (A) f
RPL	A+Y, A−Y, Y+Q, Y−Q, Y+I, Y−I	(r) → y AND A
	LP	L(y)(Q) → y AND A
	A+LP, A−LP	(A) ± L(y)(Q) → y AND A
JP*	(NOT USED)	JUMP TO ADDRESS y
RJP*	(NOT USED)	RETURN JUMP TO ADDRESS y
ADD	A, Q*	(r)+(y) → r
	LP	L(y)(Q) + (A) → A
SUB	A, Q*	(r)−(y) → r
	LP	(A) − L(y)(Q) → A
MUL	(NOT USED)	(Q)(y) → AQ
DIV*	(NOT USED)	(AQ) / (y) → Q, R → Af
SEL	SET, CP, CL	MODIFY (A)n FOR (y)n = I
	SU	(y)n → (A)n FOR (Q)n = I
RSE	SET, CP, CL	MODIFY (A)n FOR (y)n = I → y AND A
	SU	(y) → (A)n FOR (Q) n = I → y
COM	A,* Q,* AQ*	SENSE (j) (A)i = (A)f
	MASK,	(A) − L(y)(Q) SENSE (j), (A)i = (A)f
RPT*	(NOT USED)	EXECUTE NI (y) TIMES
BSK**	B^n	$(B)^n$ = (y) SKIP NI, CLEAR B^n
		$(B)^n \neq$ (y) ADV B^n, READ NI
BJP**	B^n	$(B)^n$ = 0 READ NI
		$(B)^n \neq 0$ DEC B^n, JUMP TO ADDRESS (y)
IN**	C^n	BUFFER, k = W, (y) → (n), SET (d) TO 4
		k ≠ W, (y) LOWER → (n), SET (d) TO 4
OUT**	C^n	BUFFER, k = W, (y) → (n), SET (d) TO 6
		k ≠ W, (y) LOWER → (n), SET (d) TO 6
CP	A, Q	COMPLEMENTS r
CL	A, Q, $B^{n**}C^{n**}$	CLEARS r
	(BLANK)	CLEARS (y)
SKIP	(NOT USED)	INACTIVE INSTRUCTION

* SPECIAL j-OPERANDS REQUIRED (SEE OTHER SIDE)

** USE OF j-OPERANDS NOT PERMITTED

MNEMONIC K-DESIGNATORS

READ:		STORE:		REPLACE:		
CODE	SIGNIFICANCE	CODE	SIG.	CODE	READ SIG.	STORE SIG.
(BLANK)	$(U)_L \rightarrow X_L$, $0 \rightarrow X_U$	Q	(X) → Q	(NOT USED)	—	—
L	$(Z)_L \rightarrow X_L$, $0 \rightarrow X_U$	L	$(X)_L \rightarrow Y_L$	L	$(Z)_L \rightarrow X_L$	$(X)_L \rightarrow Y_L$
U	$(Z)_U \rightarrow X_L$, $0 \rightarrow X_U$	U	$(X)_L \rightarrow Y_U$	U	$(Z)_U \rightarrow X_L$	$(X)_L \rightarrow Y_U$
W	(Z) → X	W	(X) → Y	W	(Z) → X	(X) → Y
X	$(U)_L \rightarrow X_L$, $(U)_{14} \rightarrow X_U$	A	(X) → A	(NOT USED)	—	—
LX	$(Z)_L \rightarrow X_L$, $(Z)_{14} \rightarrow X_U$	CPL	$(X)'_L \rightarrow Y_L$	LX	$(Z)_L \rightarrow X_L$, $(Z)_{14} \rightarrow X_U$	$(X)_L \rightarrow Y_L$
UX	$(Z)_U \rightarrow X_L$, $(Z)_{29} \rightarrow X_U$	CPU	$(X)'_L \rightarrow Y_U$	UX	$(Z)_U \rightarrow X_L$, $(Z)_{29} \rightarrow X_U$	$(X)_L \rightarrow Y_U$
A	(A) → X	CPW	$(X)' \rightarrow Y$	(NOT USED)	—	—

NTDS COMPILING SYSTEM (CS-1)

NORMAL J-OPERANDS

MNEMONIC CODE	OPERAND SIGNIFICANCE
SKIP	SKIP NEXT INSTRUCTION UNCONDITIONALLY
Q POS	SKIP NEXT INSTRUCTION IF Q IS POSITIVE
Q NEG	SKIP NEXT INSTRUCTION IF Q IS NEGATIVE
A ZERO	SKIP NEXT INSTRUCTION IF A IS ZERO
A NOT	SKIP NEXT INSTRUCTION IF A IS NON-ZERO
A POS	SKIP NEXT INSTRUCTION IF A IS POSITIVE
A NEG	SKIP NEXT INSTRUCTION IF A IS NEGATIVE

SPECIAL J-OPERANDS

WITH MONO-CODES JP OR RJP:

MNEMONIC CODE	OPERAND SIGNIFICANCE
Q POS	JUMP OR RETURN JUMP TO *y* IF Q IS POSITIVE
Q NEG	JUMP OR RETURN JUMP TO *y* IF Q IS NEGATIVE
A ZERO	JUMP OR RETURN JUMP TO *y* IF A IS ZERO
A NOT	JUMP OR RETURN JUMP TO *y* IF A IS NON-ZERO
A POS	JUMP OR RETURN JUMP TO *y* IF A IS POSITIVE
A NEG	JUMP OR RETURN JUMP TO *y* IF A IS NEGATIVE
(BLANK)	JUMP OR RETURN JUMP TO *y* UNCONDITIONALLY
KEY 1	JUMP OR RETURN JUMP TO *y* IF KEY 1 IS SET
KEY 2	JUMP OR RETURN JUMP TO *y* IF KEY 2 IS SET
KEY 3	JUMP OR RETURN JUMP TO *y* IF KEY 3 IS SET
STOP	JUMP OR RETURN JUMP TO *y* AND STOP UNCONDITIONALLY
STOP 5	JUMP OR RETURN JUMP TO *y*. STOP IF KEY 5 IS SET
STOP 6	JUMP OR RETURN JUMP TO *y*. STOP IF KEY 6 IS SET
STOP 7	JUMP OR RETURN JUMP TO *y*. STOP IF KEY 7 IS SET
C^n FULL	JUMP OR RETURN JUMP TO *y* IF REGISTER C^n HAS BEEN FILLED
C^n EMPTY	JUMP OR RETURN JUMP TO *y* IF REGISTER C^n HAS NOT BEEN FILLED

WITH MONO-CODES; ADD Q OR SUB Q:

SKIP	SKIP NEXT INSTRUCTION UNCONDITIONALLY
A POS	SKIP NEXT INSTRUCTION IF A IS POSITIVE
A NEG	SKIP NEXT INSTRUCTION IF A IS NEGATIVE
Q ZERO	SKIP NEXT INSTRUCTION IF Q IS ZERO
Q NOT	SKIP NEXT INSTRUCTION IF Q IS NON-ZERO
Q POS	SKIP NEXT INSTRUCTION IF Q IS POSITIVE
Q NEG	SKIP NEXT INSTRUCTION IF Q IS NEGATIVE

WITH MONO-CODE DIV:

SKIP	SKIP NEXT INSTRUCTION UNCONDITIONALLY
NO OF	SKIP NEXT INSTRUCTION IF THERE IS NOT A DIVIDE OVERFLOW
OF	SKIP NEXT INSTRUCTION IF THERE IS A DIVIDE OVERFLOW
A ZERO	SKIP NEXT INSTRUCTION IF A IS ZERO
A NOT	SKIP NEXT INSTRUCTION IF A IS NON-ZERO
A POS	SKIP NEXT INSTRUCTION IF A IS POSITIVE
A NEG	SKIP NEXT INSTRUCTION IF A IS NEGATIVE

WITH MONO-CODE RPT:

(BLANK)	NO MODIFICATION TO Y OF REPEATED INSTRUCTION
ADV	INCREASE Y OF REPEATED INSTRUCTION BY ONE ON EACH EXECUTION
BACK	DECREASE Y OF REPEATED INSTRUCTION BY ONE ON EACH EXECUTION
ADD B	ADD ASSOCIATED B-REGISTER TO Y ON EACH EXECUTION

WITH NON-MASK COMPARE MONO-CODES:

Y LESS	SKIP NEXT INSTRUCTION IF: (1) $y \leq Q$ OR (2) $y \leq A$
Y MORE	SKIP NEXT INSTRUCTION IF: (1) $y > Q$ OR (2) $y > A$
Y IN	SKIP NEXT INSTRUCTION IF $Q \geq y$ AND $y > A$
Y OUT	SKIP NEXT INSTRUCTION IF $Q < y$ OR $y \leq A$