# UNIVAC SCIENTIFIC GENERAL-PURPOSE COMPUTER SYSTEM CONTENT OF REGISTERS

PX 19

**OCTOBER 1956** 



DIVISION OF SPERRY RAND CORPORATION

#### TABLE OF CONTENTS

													Page
1. 2.	GENERAL DEFINITION OF SYMBOLS .			• • •		•	•	•		•	•		1 1
INS	TRUCTIONS												
	Transmit Positive	(TPuv) (TMuv)	-	Code						•		•	3
	Transmit Magnitude	(TMuv)		Code Code					•	•	•	•	4 5
	Transmit Negative Interpret	(INUV) (IP)	-						•	•	•	•	6
	Transmit U Address	(TUuv)		Code Code					•	•	•	•	7
	Transmit V Address	(TUuv)	_	Code				•	•	•	•	•	8
	External Function	(EF-v)	_	Code				•	•	•	•	•	9
	Replace Add	(RAuv)		Code				•	•	•	•	•	10
	Left Transmit	(LTjkv)	_	Code			•	•	•	•	•	•	11
	Replace Subtract	(RSuv)		Code			•	•	•	•	•	•	12
	Controlled Complement	(CCuv)		Code			•	•	•	•	•	•	13
	Split Positive Entry	(SPuk)	_	Code			•	•	•	•	•	•	14
	Split Add	(SAuk)		Code			•	•	•	•	•	•	14
	Split Negative Entry	(SNuk)	_	Code			•	•	•	•	•	•	15
	Split Subtract	(SSuk)		Code				•	•	•	•	•	15
	Add and Transmit	(ATuv)		Code			-	•	•	•	•	•	16
	Subtract and Transmit	(STuv)		Code									17
	Return Jump	(RJuv)	-	Code									18
	Index Jump	(IJuv)	. •	Code									19
	Threshold Jump	(TJuv)	_	Code									21
	Equality Jump	(EJuv)	_	Code									21
	Q-Jump	(QJuv)		Code									22
	Manually Selective Jump			Code									23
	Sign Jump	(SJuv)		Code									23
	Zero Jump	(ZJuv)	(0p	Code	47).								23
	Q-Controlled Transmit	(QTuv)		Code									25
	Q-Controlled Add	(QAuv)	(0p	Code	52).								26
	Q-Controlled Substitute	(QSuv)	(0p	Code	53).								27
	Left Shift in A	(LAuk)	(0p	Code	54).					•			28
	Left Shift in Q	(LQuk)		Code						•			28
	Manually Selective Stop			Code				•		•	•	•	29
	Program Stop	(PS)		Code			•	•		•	•		29
	Print	(PR-v)	•	Code		•	•	•	•	•	•	•	30
	Punch	(PUjv)	_	Code			•	•	•	•	•	•	30
	Multiply	(MPuv)	-	Code		•	•	•	•	•	•	•	31
	Multiply Add	(MAuv)		Code		•	•	•	•	•	•		32
	Divide	(DVuv)	-	Code		•	•	•	•	•	•	•	33
	Scale Factor	(SFuv)		Code		•	•	•	•	•	•	•	34
	Repeat	(RPjnw)		Code		•	•	•	•	•	•	•	35
	External Read	(ERjv)	-	Code		•	•	•	•	•	•	•	36
	External Write	(EWiv)	a0	Code	77).		_	_	_	_	_	_	36

#### 1. GENERAL.

This volume is a tabulation of the final contents of the memory locations, the Q Register and the Accumulator, after the execution of each instruction. The instruction tables are listed in the numerical order of their octal operation codes. The left column, STORAGE CLASS SELECTION, lists the class of storage chosen for the u address and v address. The remainder of the table lists the final contents of MC, MD, A, and Q for all cases of storage class selections. Those instructions which do not result in changes contain no tables but, instead, have notes covering these exceptions. A series of dashes in any table position indicate that the memory location or register is not involved in the execution of the instruction. An SCC table entry indicates that an SCC Fault occurs causing the computer to stop.

#### 2. DEFINITION OF SYMBOLS.

- A The 72-bit Accumulator
- AR The right-hand 36 bits of A
- A<sub>I.</sub> The left-hand 36 bits of A
- Q The 36-bit Q Register
- MD Magnetic Drum Storage (16,384 36-bit words)
- MC Magnetic Core Storage (4096 36-bit words)
- u The first execution address  $(i_{29}, i_{28}, \ldots, i_{15})$
- v The second execution address  $(i_{14}, i_{13}, \ldots, i_0)$
- .2k Left circular shift k places
- () (Parentheses) Denotes "the content of"
- (); The "initial content of"
- ()<sub>f</sub> The "final content of"
- ()' (Prime) The "complement of the content of"

The absolute value of the expression A one digit octal number (u $_{14}$ , u $_{13}$ , u $_{12}$ ) j A four digit octal number (u<sub>11</sub>, u<sub>10</sub>, ..., u<sub>0</sub>) n D( ) A double extension of the contents of the parentheses S( ) A single extension of the contents of the parentheses The bit-by-bit product of (u) and (Q) L(Q)(u)The bit-by-bit product of (v) and the complement of (Q)L(Q)'(v) $\oplus$ Denotes a bit-by-bit sum without carries.

Instruction:	TRANSMIT POSITIVE (TPuv)	Operation Code:	11
Function:	Replace (v) with (u).		

Storage Class Selection		Conte		and Storage Posit on is Executed.	ions
		$(MC)_{f}$ or $($	(MD) <sub>f</sub>		
u	v	u	v	(A) <sub>f</sub>	$(Q)_{\mathbf{f}}$
MC	MC or MD	No Change	(u)		
or MD	A	No Change		D(u)	
	Q	No Change			(u)
	MC or MD		(A <sub>R</sub> )	No Change	
A	A			D(A <sub>R</sub> ) <sub>i</sub>	~
	Q			No Change	(A <sub>R</sub> )
	MC or MD		(Q)	~	No Change
Q	A			D(Q)	No Change
	Q				No Change

Instruction: TRANSMIT MAGNITUDE (TMuv) Operation Code: 12

Function: Replace (v) with the absolute magnitude of (u).

Storage Class		Content of Registers and Storage Positions after Operation is Executed.			
	ction	$(MC)_{f f}$ or	$(MD)_{\mathbf{f}}$	(4)	(0)
u	v	u	v	(A) <sub>f</sub>	(Q) <sub>f</sub>
мс	MC or MD	No Change	(u)		
or MD	A	No Change		D   (u)	
	Q	No Change			(u)
	MC or MD		(A <sub>R</sub> )	No Change	
A	A			$D   (A_R)_i  $	
	Q			No Change	$ (A_R) $
	MC or MD		(Q)		No Change
Q	A			D   (Q)	No Change
	Q				$ Q_i $

Instruction:	TRANSMIT NEGATIVE (TNuv)	Operation Code:	13
Function:	Replace (v) with the complement of (u).		

Storage Class Selection		Content of Registers and Storage Positions after Operation is Executed.					
		$(MC)_{\mathbf{f}}$ or	(MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>F</sub>		
u	v	u	V	_			
MC	MC or MD	No Change	(u)'				
or MD	A	No Change		D(u)'			
	Q	No Change			(u)'		
	MC or MD		(A <sub>R</sub> )'	No Change			
A	A			D(A <sub>R</sub> ) <sub>i</sub> '			
	Q			No Change	(A <sub>R</sub> )'		
	MC or MD		(Q)'		No Change		
Q	A			D(Q)'	No Change		
	Q				(Q) <sub>i</sub> '		

Instruction: INTERPRET (IP--) Operation Code: 14

Function: Let y represent the address from which CI was obtained. Replace the right-hand 15 bits of  $(F_1)$  with the quantity y+1. Then take  $(F_2)$  as the next instruction.  $F_1 \text{ and } F_2 \text{ are MC addresses 00000 and 00001 respectively.}$  The right-hand 30 bits of (y) are unaffected by this operation. The contents of A and Q are left unchanged.

Inst	cuction:	TRANSMIT U ADD	RESS (TUuv)	0pera	tion Code: 15			
Function:		Replace the 15	bits of (v) desig	nated v <sub>15</sub> throu	gh $v_{29}$ with the			
		corresponding	corresponding bits of (u). The remaining 21 bits of (v) are					
		not to be dist	urbed.					
		Com	tont of Degisters	and Stampa Das	itions			
Stora Clas		Con	tent of Registers after Operat	ion is Executed				
Seled	ction	(MC) <sub>f</sub> o	r (MD) <sub>f</sub>	$(A)_{\mathbf{f}}$	(Q) <sub>f</sub>			
u	V	u	V					
MC	MC or MD	No Change	$\left\{ \left\{ \left(\mathbf{v}_{0-14}\right)_{\mathbf{i}}\right\} \right $					
or MD	A	٦	(u <sub>15-29</sub> )					
	Q	}	SCC FAULT		1			
	MC	· <i>J</i>						
A	or MD	 	$\left\{\begin{array}{c} {}^{(v_{0-14})}_{i} \\ {}^{(A_{15-29})} \end{array}\right\}$	No Change				
	A	)	(v <sub>30-35</sub> ) <sub>i</sub>					
	Q	}	SCC FAULT		[			
	MC		\[ \int \( (v_{0-14})_i \) \]		No Change			
	or MD				onango			
			(Q <sub>15-29</sub> ) (v <sub>30-35</sub> ) i					
Q								
	А	}						
	Q	J	SCC FAULT					

Inst	ruction	: TRANSMIT V AD	DRESS (TVuv)	Opera	ation Code: 16				
Func	tion:	Replace the r	ight-hand 15 bits of (v), $v_0$ through $v_{14}$ , with the						
corresponding bits of (u). The remaining 21 bits of (v) are									
	not to be disturbed.								
		Co	ontent of Registers						
Stor Cla	SS		-	tion is Executed	i.				
Sele	ction v	(MC) <sub>f</sub> or	· (MD) <sub>f</sub>	(A) <sub>f</sub>	$(Q)_{\mathbf{f}}$				
u		u							
	MC or	No Change	(u <sub>0-14</sub> )						
MC or	MD		(v <sub>15-35</sub> ) <sub>i</sub>						
MD	A	}	SCC FAULT						
	Q	J							
	MC or		(A <sub>0-14</sub> ) (v <sub>15-35</sub> ) <sub>i</sub>	No Change					
A	MD		(v <sub>15-35</sub> ) <sub>i</sub>						
	A	]	SCC FAULT		•				
	Q	<b>\</b>	SCC FAULI						
	MC		(Q <sub>0-14</sub> )		No Chango				
	or MD		(v <sub>15-35</sub> ) <sub>i</sub>		No Change				
Q									
	A								
	Q		SCC FAULT						
	]								

Instruction:	EXTERNAL FUNCTION (EF-v)	Operation Code: 17
Function:	Select a unit of external equipment and	perform the function
	designated by (v).	
	(No Change in Content of Registers)	

Instruction: REPLACE ADD (RAuv) Operation Code: 21 Function: Form in A the sum of D(u) and D(v). Then replace (u) with  $(A_R)$ .

	rage ass	(		ers and Storage Pos ration is Executed	itions
	ection	(MC) <sub>f</sub>	or (MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>f</sub>
u	v	u	v	1	
MC or	MC or MD	(A <sub>R</sub> ) <sub>f</sub>	No Change	D(u) <sub>i</sub> +D(v)	
MD	A	$(A_R)_f$		2D(u)	
	Q	$(A_R)_f$		$D(u)_{i}^{+D(Q)}$	No Change
	MC or MD		No Change	D(A <sub>R</sub> ) <sub>i</sub> +D(v)	
A	A			2D(A <sub>R</sub> ) <sub>i</sub>	
	Q			$D(A_R)_{i}^{+D(Q)}$	No Change
Q	MC or MD		No Change	D(Q) <sub>i</sub> +D(v)	$(A_R)_f$
y	A			2D(Q)	$(A_R)_f$
	Q			2D(Q) <sub>i</sub>	$(A_R)_f$

Instruction	: LEFT TRANSMIT (LTjkv)		Operation Code: 22
Function: I	Left circular shift (A)	by k places. Then r	eplace
ı	(v) with ( $A_L$ ) if j=0, or	replace (v) with (A	<sub>R</sub> )
:	if j=1.		
Storage Class		gisters and Storage peration is Executed	
Selection for v	(MC) <sub>f</sub> or (MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>f</sub>
	j=	0	
MC or MD	$(A_L)_f$	$(A)_{i} \cdot 2^{k}$	
A		$D(A_L)_k$ where $(A)_k$ is $(A)_i \cdot 2^k$	
Q		(A) <sub>i</sub> ·2 <sup>k</sup>	$(A_L)_f$
	j=	1	
MC or MD	$(A_R)_f$	(A) <sub>i</sub> ·2 <sup>k</sup>	
A		$D(A_R)_k$ where $(A)_k$	
		is (A) <sub>i</sub> ·2 <sup>k</sup>	
Q		(A) <sub>i</sub> ·2 <sup>k</sup>	$(A_R)_f$

Instruction:	REPLACE SUBTRACT (RSuv)	Operation Code:	23
Function:	Form in A the difference D(u) minus D(v).	Then replace	
	(u) with $(A_R)$ .		

Storage Class		Content of Registers a after Operatio		and Storage Positions on is Executed.	
Sele	ction	(MC) <sub>f</sub>	or (MD) $_{f f}$	$(A)_{\mathbf{f}}$	$(Q)_{\mathbf{f}}$
u	v	u	v		
MC or	MC or MD	$(A_R)_f$	No Change	D(u) <sub>i</sub> -D(v)	
MD	A	0		0	
	Q	$(A_R)_f$		D(u) <sub>i</sub> -D(Q)	No Change
_	MC or MD		No Change	$D(A_R)_i-D(v)$	
A	A			0	
	Q			$D(A_R)_{i}-D(Q)$	No Change
	MC or MD		No Change	D(Q) <sub>i</sub> -D(v)	$(A_R)_f$
Q	A			0	0
	Q			0	0

Inst	truction:	CONTROLLED	CONTROLLED COMPLEMENT (CCuv)			on Code: 27		
Func	ction:	Replace ( $A_R$ ) with (u) leaving ( $A_L$ ) undisturbed.						
		Then complement those bits of $(A_R)$ that correspond						
	to ones in (v). Then replace (u) with $(A_R)$ .							
	rage ass	Con	tent of Regis after Op	ters and Stor eration is Ex				
	ection	(MC) <sub>f</sub> or	(MD) <sub>f</sub>					
u	v	u	v	$(\mathtt{A_L})_{\mathbf{f}}$	(A <sub>R</sub> ) <sub>f</sub>	(Q) <sub>f</sub>		
	MC or	$(A_R)_f$	No Change	No Change	(u); (v)			
MC	MD	(AK)I	iyo change	No Change				
MD	A	$(A_R)_f$		No Change	Zero			
	Q	$(A_R)_f$		No Change	(u) <sub>i</sub> (Q)	No Change		
	MC or		No Change	No Change	$(A_R)_i \oplus (v)$			
A	MD		No onange	140 Onunge	(AR)1			
	A			No Change	Zero			
	Q			No Change	$(A_R)_i \oplus (Q)$	No Change		
	MC or MD		No Change	No Change	(Q) <sub>i</sub> (v)	$(A_R)_\mathbf{f}$		
Q	A			No Change	Zero	Zero		
	Q			No Change	Zero	$(A_{R})_{\mathbf{f}}$		

Instruction:	SPLIT POSITIVE ENTR	Y (SPuk)	Operation Code: 31			
Function:	Form S(u) in A. The	en left circular shift	(A) by k places.			
Storage Class	Content of Registers and Storage Positions after Operation is Executed.					
Selection	d1	rer operation is execu	teu.			
for u	$(\texttt{MC})_{\mathbf{f}}$ or $(\texttt{MD})_{\mathbf{f}}$	(A) <sub>f</sub>	$(Q)_{\mathbf{f}}$			
MC	No Change	S(u) · 2 <sup>k</sup>				
or MD	No Change	S(u) 2-				
A		$S(A_R)_i.2^k$ $S(Q).2^k$				
Q		S(Q) .2k	No Change			

Instruction: SPLIT ADD (SAuk) Operation Code: 32
--

Function: Add S(u) to A. Then left circular shift (A) by k places

Storage Class Selection	Content of Registers and Storage Positions after Operation is Executed.				
for u	$(MC)_{f f}$ or $(MD)_{f f}$	(A) <sub>f</sub>	$(Q)_{\mathbf{f}}$		
MC or MD	No Change	$\left[ (A)_{i} + S(u) \right] \cdot 2^{k}$	<b></b> -		
A		$\left[ (A)_{i} + S(A_{R})_{i} \right] \cdot 2^{k}$			
Q		$\left[ (A)_{i} + S(Q) \right] \cdot 2^{k}$	No Change		

Instruction:	SPLIT NEGATIVE ENTRY	(SNuk)	Operation Code: 33	
Function:	Form in A the comple	ement of S(u). Then	left circular shift	
	(A) by k places.			
Storage Content of Registers and Storage Positions after Operation is Executed.				
Class Selection for u	$(\mathrm{MC})_{\mathbf{f}}$ or $(\mathrm{MD})_{\mathbf{f}}$	(A) <sub>f</sub>	(Q) <sub>f</sub>	
MC or MD	No Change	S(u)'·2 <sup>k</sup>		
A		$S(A_R)_i$ ' $\cdot 2^k$		

S(Q)' ·  $2^k$ 

No Change

Instruction:	SPLIT SUBTRACT (SSu	k)	Operation Code:	34
Function:	Subtract S(u) from A	A. Then left circular	shift (A) by	
	k places.			
Storage Class	Content of Registers and Storage Positions after Operation is Executed.			
Selection for u	(MC) <sub>f</sub> or (MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>f</sub>	
MC or MD	No Change	$\left[ (A)_{i} - S(u) \right] \cdot 2^{k}$		
A		$\begin{bmatrix} (A)_{i} - S(u) \end{bmatrix} \cdot 2^{k}$ $\begin{bmatrix} (A)_{i} - S(A_{R})_{i} \end{bmatrix} \cdot 2^{k}$ $\begin{bmatrix} (A)_{i} - S(Q) \end{bmatrix} \cdot 2^{k}$		
Q		$\left[ (A)_{i} - S(Q) \right] \cdot 2^{k}$	No Change	i

Instruction: ADD AND TRANSMIT (ATuv) Operation Code: 35

Function: Add D(u) to (A). Then replace (v) with (A $_{R}$ ).

Storage Class Selection		Content of Registers and Storage Positions after Operation is Executed.			ons
		(MC) <sub>f</sub> or (MD) <sub>f</sub>		(A) <sub>f</sub>	(Q) <sub>f</sub>
u	v	u	v	1	
MC	MC or MD	No Change	$(A_R)_f$	(A) <sub>i</sub> +D(u)	
or MD	A	No Change		(A) <sub>i</sub> +D(u)	
	Q	No Change		(A) <sub>i</sub> +D(u)	$(A_R)_f$
	MC or MD		(A <sub>R</sub> ) <sub>f</sub>	$(A)_i + D(A_R)_i$	
A	A			$(A)_{i}^{+D}(A_{R})_{i}$	
	Q			$(A)_{i}+D(A_{R})_{i}$	$(\mathtt{A_R})_{\mathbf{f}}$
_	MC or MD		$(A_R)_f$	(A) <sub>i</sub> + D(Q)	No Change
Q	A			$(A)_{i}^{+} D(Q)$	No Change
	Q	au an		$(A)_i^{+D(Q)}_i$	$(A_R)_f$

Instruction:	SUBTRACT AND TRANSMIT (STuv)	Operation Code: 36
Function:	Subtract D(u) from (A). Then replace (v	) with $(A_R)$ .

Storage Class		Content of Registers and Storage Positions after Operation is Executed.				
Sele	ection	(MC) <sub>f</sub>	or (MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>f</sub>	
u	V	u	V			
MC	MC or MD	No Change	$(A_R)_{\mathbf{f}}$	(A) <sub>i</sub> -D(u)		
or MD	A	No Change		(A) <sub>i</sub> -D(u)		
	Q	No Change		(A) <sub>i</sub> -D(u)	$(A_R)_f$	
	MC or MD		$(A_R)_{\mathbf{f}}$	$(A)_i$ -D $(A_R)_i$		
A	A			$(A)_i-D(A_R)_i$		
	Q			$(A)_i-D(A_R)_i$	$(A_R)_f$	
Q	MC or MD		$(A_R)_f$	(A) <sub>i</sub> -D(Q)	No Change	
	A			$(A)_{i}$ -D(Q)	No Change	
	Q			$(A)_i$ -D $(Q)_i$	$(A_R)_f$	

Instruction:	RETURN JUMP (RJuv)	Operation Code: 37		
Function	Let y represent the address from which CI was obtained.			
	Replace the right-hand 15 bits of (u) with quantity y			
	plus 1. Then take (v) as NI.			
	(If both u and v refer to MC or MD there is	s no change in contents		
	of A and Q. If u is Q or A, an SCC FAULT	occurs. Also see		
	page 24.)			

Instruction:		INDEX JUMP (IJuv)			Operation Code: 41		
Function:		Form in A the difference D(u) minus 1. If A <sub>71</sub> is then 1,					
į		continue with	continue with the present sequence of instructions; if A <sub>71</sub>				
		is O, replace	e (u) with $(A_R)$ and	d take (v) a	s the next		
		instruction.					
					-		
Stor Cla	iss	Con	tent of Registers a after Operation				
Sele	ection	$(MC)_{\mathbf{f}}$ or $(MD)_{\mathbf{f}}$		(A) <sub>f</sub>	(Q) <sub>f</sub>		
u	v	u	V	(1)1	(4)1		
			$FOR A_{71} = 1$				
	MC or MD	No Change	No Change	D(u)-1			
MC or MD	A*	No Change		D(u)-1			
- CIII	Q*	No Change		D(u)-1	No Change		
	MC or MD		No Change	(A) <sub>i</sub> -1			
A	A*			(A) <sub>i</sub> -1			
	Q*			(A) <sub>i</sub> -1	No Change		
0	MC or MD		No Change	D(Q)-1	No Change		
Q	A*			D(Q)-1	No Change		
	Q*			D(Q)-1	No Change		

<sup>\*</sup> see page 24

INDEX JUMP (IJuv) continued Content of Registers and Storage Positions Storage after Operation is Executed. Class  $(MC)_{f}$  or  $(MD)_{f}$ Selection (A)<sub>f</sub>  $(Q)_{\mathbf{f}}$ FOR  $A_{71} = 0$ MC  $(u)_{i}-1$  $D(u)_i - 1$ No Change or MC MD  $D(u)_i - 1$  $(u)_{i}$ -1 A\* MD Q\*  $(u)_{i}$ -l  $D(u)_{i}-1$ No Change MC $(A)_i - 1$ No Change or MD Α  $(A)_i - 1$ A\*  $(A)_i$ - 1 Q\* No Change MC  $D(Q)_i - 1$  $(Q)_{i}^{-1}$ No Change orMD Q  $(Q)_{i}$ -1  $D(Q)_{i}-1$ A\*

Q\*

 $D(Q)_i - 1$ 

 $(Q)_{i}-1$ 

<sup>\*</sup> see page 24

Instruction:	THRESHOLD JUMP (TJuv)	Operation Code: 42		
Function:	Subtract (u) from (A). If $A_{71}$ is then	1, take (v) as the next		
	instruction; if $A_{71}$ is 0, continue wit	h the present sequence of		
	instructions. Then, in either case, restore (A) to its initial			
	state.			
	(No change in Content of Registers or	Storage Positions event		
	(No change in Content of Registers of	Storage Positions except		
	for those special cases outlined on pa	ges 24 and 35.)		

Instruction:	EQUALITY JUMP (EJuv)	Operation Code: 43
Function:	Subtract (u) from (A). If (A) is then	zero, take (v) as the
	next instruction; if (A) is not zero,	continue with the present
	sequence of instructions. In either c	ase restore (A) to its
	initial state.	
	(No Change in Content of Registers or	Storage Positions except
	for those special cases outlined on pa	ges 24 and 35.)

Inst	ruction:	Q-JUMP (QJuv)	)		Operation Code: 44	
Function:		If ${\bf Q}_{35}$ is 1, take (u) as the next instruction; if ${\bf Q}_{35}$ is 0,				
		take (v) as	take (v) as the next instruction. Then, in either case,			
		left circula	r shift (Q) by one	place.		
			Content of Registe	ore and Storage	Dosition	
Stor Cla		'		ration is Execu		
Selection		(MC) <sub>f</sub> o	r (MD) <sub>f</sub> (A) <sub>f</sub>		(Q) <sub>f</sub>	
u	v	u	v	1	_	
	MC	No Change	No Chango		(Q) <sub>i</sub> ·2	
	or MD	No Change	No Change		(Q) <sub>i</sub> 2	
MC or						
MD	A*	No Change		No Change	(Q) <sub>i</sub> ·2	
	Q*	No Change			(Q) <sub>i</sub> ·2	
	MC		N. Cl.	No. Chamas	(0). •2	
	or MD		No Change	No Change	(Q) <sub>i</sub> ·2	
A	A*			No Change	(Q) <sub>i</sub> ·2	
	Q*			No Change	(Q) <sub>i</sub> ·2	
*u-c	MC					
	or MD		No Change		(Q) <sub>i</sub> ·2	

A\*

Q#

 $(Q)_{i}$  '2

(Q) $_{i}$   $\cdot_{2}$ 

No Change

<sup>\*</sup> see page 24

Instruction:	MANUALLY SELECTIVE JUMP (MJjv)	Operation Code: 45				
Function:	If the number j(given by $u_{13}$ $u_{12}$ ) is 0,	take (v) as the next				
	instruction. If j is 1, 2 or 3 and the	correspondingly				
	numbered manual jump-selecting switch is	numbered manual jump-selecting switch is set to "jump", take				
	(v) as the next instruction; otherwise	if this switch is not				
	set to "jump", continue with the present	t sequence of				
	instruction.					
	(No Change in Content of Registers or St	corage Positions				
	and see page 24.)					
Instruction:	SIGN JUMP (SJuv)	Operation Code: 46				
Function:	If $A_{71}$ is 1, take (u) as the next instruction. If $A_{71}$ is 0,					
	take (v) as the next instruction.					
	(No Change in Content of Registers or St	Corage Positions				
	and see page 24.)					
	and too page 11.7					
Instruction:	ZERO JUMP (ZJuv)	Operation Code: 47				
Function:	If (A) <u>is</u> <u>not</u> zero, take (u) as the next	instruction; if (A) <u>is</u>				
	zero, take (v) as the next instruction.	In either case leave				
	(A) in its initial state.					
	(No Changes in Content of Registers or S	torage Positions				
	and see page 24.)					

#### Notes Concerning the Jump Instructions

- 1. If v refers to A, an SCC FAULT occurs.
- 2. If v refers to Q, no fault occurs, and Control obtains the NI from (Q). If the (Q) is a legal instruction, it will be executed in the normal manner. Unless (Q) is a jump instruction, however, the following will occur: (1) PAK will be advanced and the (Q) will be taken as NI; (2) Control will be directed to Q again after executing the (Q), PAK will be advanced, and (Q) executed again. This process will continue until a FORCE stop is made. PAK advances from 31000 to 31777 and then starts over from 31000, each time, of course, referencing Q as the address of NI.
- The above remarks also apply to u for the two way jump instruction,
   QJ, SJ, and ZJ.

Instruction:	Q-CONTROLLED TRANSMIT (QTuv)		Operation Code:	51
Function:	Form in A the number L(Q)(u).	Then replac	e (v) by (A <sub>R</sub> ).	

	rage ass	Cont	ent of Registers after Operati	and Storage Posi	tions
Sel	ection	See Storage	f or (MD) <sub>f</sub> Class Selection	(A) <sub>f</sub>	(Q) <sub>f</sub>
u	V	u	V	,1	
MC or	MC or MD	No Change	(A <sub>R</sub> ) <sub>f</sub>	L(Q)(u)	No Change
MD	A	No Change		L(Q)(u)	No Change
	Q	No Change		L(Q)(u)	$(A_R)_f$
_	MC or MD		$(A_R)_\mathbf{f}$	L(Q) (A <sub>R</sub> ) <sub>i</sub>	No Change
A	A			L(Q) (A <sub>R</sub> ) <sub>i</sub>	No Change
	Q			$L(Q) (A_R)_i$	$(A_R)_f$
0	MC or MD		(Q)	S(Q)	No Change
Q	A			S(Q)	No Change
	Q			S(Q)	No Change

Instruction: Q-CONTROLLED ADD(QAuv) Operation Code: 52 Function: Add to (A) the number L(Q)(u). Then replace (v) by  $(A_R)$ .

Storage Class		Content of Registers and Storage Positions after Operation is Executed.				
Sele	ection	$\left(  exttt{MC}  ight)_{ exttt{f}}$ or $\left(  exttt{MD}  ight)_{ exttt{f}}$ See Storage Class Selection		(A) <sub>f</sub>	(Q) <sub>f</sub>	
u	V	u	V	-		
MC	MC or MD	No Change	(A <sub>R</sub> ) <sub>f</sub>	(A) <sub>i</sub> + L(Q)(u)	No Change	
or MD	A	No Change		$(A)_i + L(Q)(u)$	No Change	
	Q	No Change		$(A)_i + L(Q)(u)$	$(A_R)_f$	
	MC or MD		$(A_R)_{\mathbf{f}}$	$(A)_i + L(Q)(A_R)_i$	No Change	
A	A			$(A)_i + L(Q)(A_R)_i$	No Change	
	Q			$(A)_i + L(Q)(A_R)_i$	$(A_R)_f$	
	MC or MD		$(A_R)_f$	$(A)_{i+S(Q)}$	No Change	
Q	A			$(A)_{i}+S(Q)$	No Change	
	Q			$(A)_{i}+S(Q)$	$(\mathtt{A}_{\mathrm{R}})_{\mathbf{f}}$	

Insti	uction:	Q-CONTROLLED	Q-CONTROLLED SUBSTITUTE (QSuv) Operation Code:				
Funct	ion:	Form in A the	quantity L(Q	)(u)+L(Q)'(v); t	hen replace (v)		
		with $A_R$ . (Th	e effect of t	his is to replac	e the digits of (v)	)	
		with the digi	ts of (u) whe	re there are 1's	in Q.)		
		Con	tent of Regis	ters and Storage	Positions		
Stora Clas				eration is Execu			
1	tion	$(MC)_{\mathbf{f}}$	(0)				
u	v	u v		(A) <sub>f</sub>	(Q) <sub>f</sub>		
	MC	v		T (0) ( ) IT (0			
MC or	or MD	No Change	(A <sub>R</sub> ) <sub>f</sub>	L(Q)(u)+L(Q	))'(v) No Chang	je	
MD	A	No Change		L(Q)(u)	No Chanç	ge	
	Q	No Change		L(Q)(u)+S(Q	$(A_R)_f$		
	MC or		(1)	L(Q)(A <sub>R</sub> );+L(	Q)'(v) No Chang	70	
	MD		(A <sub>R</sub> ) <sub>f</sub>	E(Q)(AR)1'E(	Q) (V) No Chang	je	
A	A			$L(Q)(A_R)_i$	No Chang	ge	
	Q			$L(Q)(A_R)_i + $	$S(Q)'$ $(A_R)_f$		
	мс		(4)	9/0):-/-			
	or MD		(A <sub>R</sub> ) <sub>f</sub>	S(Q)+L(Q)'(v	No Chang	ge	
Q	A			S(Q)	No Chanç	ge	
	Q			2 <sup>36</sup> -1	2 <sup>36</sup> -1		

Instruction:	LEFT SHIFT IN A (L	Operation Code: 54				
Function:	Replace (A) with D	Replace (A) with D(u); then left circular shift (A) by k				
	places; then repla	ce (u) with $(A_R)$ .				
Storage Class		Content of Registers and Storage Positions after Operation is Executed.				
Selection for u	$(MC)_{f f}$ or $(MD)_{f f}$	(A) <sub>f</sub>	(Q) <sub>f</sub>			
MC or MD	$(A_R)_f$	$D(\mathfrak{u})_{\dot{1}} \cdot 2^{\dot{k}}$				
A		$^{\rm (A)}{}_{\rm i}\cdot2^{\rm k}$				
Q		$\begin{array}{c} \text{(A)}_{\mathbf{i}}  \cdot  2^{\mathbf{k}} \\ \\ \text{D(Q)}_{\mathbf{i}}  \cdot  2^{\mathbf{k}} \end{array}$	$(A_R)_f$			

Instruction:	LEFT SHIFT IN Q (L	Quk)	Operation Code: 55
Function:	Replace (Q) with (	u); then left circular	shift (Q) by k places;
	then replace (u) w	rith (Q).	
Storage Class Selection		f Registers and Storage fter Operation is Execu	
for u	$(\mathtt{MC})_{\mathbf{f}}$ or $(\mathtt{MD})_{\mathbf{f}}$	(A) <sub>f</sub>	(Q) <sub>f</sub>
MC or MD	(u) <sub>i</sub> ·2 <sup>k</sup>		(u) i · 2k
A		$D(Q)_{f}$	$(A_R)_i \cdot 2^k$
Q			$(A_R)_i \cdot 2^k$ $(Q)_i \cdot 2^k$

Instruction:	MANUALLY SELECTIVE STOP (MSjv)	Operation Code: 56			
Function:	If the number j (given by $u_{14}$ , $u_{13}$ , $u_{12}$ )	is O, stop the			
	computer operation and provide suitable indication. If j is				
1, 2, or 3 and the correspondingly numbered manual stop					
	selecting switch is set to "stop", stop	the computer operation			
	and provide suitable indication. Whethe	r or not a stop occurs,			
	take (v) as the next instruction.				
	(No Change in Content of Registers or St	orage Positions.)			

Instruction:	PROGRAM STOP	(PS)	Operation Code:	57
Function:	Stop computer	operation and provide suit	able indication.	
	(No Change in	Content of Registers or St	orage Positions.)	

Instruction:	PRINT (PR-v) Operation Code: 6			
Function:	Replace (TWR) with the right-hand 6 bits	of (v). Cause		
	the electric typewriter to print the char	racter to which		
	this code corresponds.			
	(No Change in Content of Registers or St	orage Positions.)		

Instruction:	PUNCH (PUjv)	Operation Code: 63	
Function:	Replace (HPR) with the right-hand 6 bits	of (v). Cause	
	the punch to respond to (HPR). If $j = 0$	, omit seventh	
	level hole; if $j = 1$ , include seventh level hole.		
	(No Change in Content of Registers or St	orage Positions.)	

Instruction: MULTIPLY (MPuv) Operation Code: 71

Function: Form in A the 72-bit product of (u) and (v), leaving in

Q the multiplier (u).

Storage Class Selection		Content of Registers and Storage Positions after Operation is Executed.				
		(MC) <sub>f</sub>	or (MD) <sub>f</sub>	(A) <sub>f</sub>	(Q) <sub>f</sub>	
u	v	u	v			
MC or MD	MC or MD	No Change	No Change	(u)·(v)	(u)	
אודי	A	No Change		0	(u)	
	Q	No Change		(u) <sub>i</sub> <sup>2</sup>	(u)	
	MC or MD		No Change	(A <sub>R</sub> ) <sub>i</sub> .(v)	(A <sub>R</sub> ) <sub>i</sub>	
A	A			0	(A <sub>R</sub> ) <sub>i</sub>	
	Q			$(A_R)_i^2$	(A <sub>R</sub> ) <sub>i</sub>	
	MC or MD		No Change	(Q)·(v)	No Change	
Q	A			0	No Change	
	Q			$(Q)^2$	No Change	

Instruction: MULTIPLY ADD (MAuv)

Operation Code: 72

Function: Add to (A) the 72-bit product of (u) and (v), leaving in

Q the multiplier (u).

Storage Class Selection		Co	Content of Registers and Storage Positions after Operation is Executed.				
		(MC) <sub>f</sub>	$(MC)_{\mathbf{f}}$ or $(MD)_{\mathbf{f}}$		(Q) <sub>f</sub>		
u	v	u	v		_		
MC	MC or MD	No Change	No Change	(A) <sub>i</sub> + (u) · (v)	(u)		
or MD	A	No Change		$(A)_{i}^{+} (u) \cdot (A_{L})_{i}$	(u)		
	A	No Change		$(A)_{i}^{+} (u)^{2}$	(u)		
	MC or MD		No Change	$(A)_{i}^{+} (A_{R})_{i} \cdot (v)$	(A <sub>R</sub> ) <sub>i</sub>		
A	A			$(A)_{i}^{+} (A_{R})_{i} \cdot (A_{L})_{i}$	(A <sub>R</sub> ) <sub>i</sub>		
	Q			$(A)_{i}^{+} (A_{R})_{i}^{2}$	(A <sub>R</sub> ) <sub>i</sub>		
	MC or MD		No Change	(A) <sub>i</sub> + (Q)· (v)	No Change		
Q	A			$(A)_i^+(Q) \cdot (A_L)_i$	No Change		
	Q			$(A)_{i}+(Q)^{2}$	No Change		

Instruction:	DIVIDE (DVuv)	Operation Code: 73	
Function:	Divide the 72-bit number in A by (u),	putting the quotient in	
	Q and leaving in A a non-negative remai	nder, R. Then replace	
	(v) by (Q). The quotient and remainder are defined by:		
	$(A)_{i} = (u) \cdot (Q) + R \text{ where } 0 \le R <  (u) $		

Storage Class Selection		Content of Registers and Storage Positions after Operation is Executed.				
		$(\mathrm{MC})_{\mathbf{f}}$ or $(\mathrm{MD})_{\mathbf{f}}$		(A) <sub>f</sub>	(Q) <sub>f</sub>	
u	v	u	v	_	-	
MC or	MC or MD	No Change	[(A) <sub>i</sub> -R]/(u)	R	$\left[ (A)_{i} - R \right] / (u)$	
MD	A	No Change		$D(Q)_{\mathbf{f}}$	$\left[ (A)_{i}-R\right]/(u)$	
	Q	No Change		R	$\left[ (A)_{i} - R \right] / (u)$	
	MC or MD		$[(A)_i-R]/(A_R)_i$	R	$[(A)_i-R]/(A_R)_i$	
A	A			$D(Q)_{\mathbf{f}}$	$[(A)_i-R]/(A_R)_i$	
	Q			R	$\left[ (A)_{i} - R \right] / (A_{R})_{i}$	
	MC or MD		$[(A)_i-R]/(Q)_i$	R	$\left[ \left( A \right)_{i} - R \right] / \left( Q \right)_{i}$	
Q	A			$D(Q)_{\mathbf{f}}$	$[(A)_i-R]/(Q)_i$	
	Q			R	$\left[ (A)_{i} - R \right] / (Q)_{i}$	

Instruction	n: SCALE FACTOR (SFuv)	Operation Code: 74
Function:	Replace (A) with D(u) unless u is A. Ther	n left circular shift
	(A) 36 places and continue shifting until	$A_{35} \neq A_{34}$ . Replace
	the right-hand 15 bits of (v) with the num	nber of left shifts, k,
	necessary to return the final contents of	A or $(A)_f$ to the origi
	nal position. The range of k, if u is A, is	s 0≤ k≤ 71; if u is
	MC, MD, or Q, k may be 0 or $37 \le k \le 71$ . Ef	ffectively, the ini-
	tial content of A,or $(A)_i$ , which may be $D(u)$ or $D(Q)$ after the	
	above replacement, is positioned in $\mathbf{A}_{R}$ (with the sign bit re-	
	presented by $A_{35}$ and the most significant bit by $A_{34}$ ) so that	
	$(A)_f = (A)_i \cdot 2^s$ . If $0 \le k \le 36$ , the Scale Factor, $s = -k$ ; if	
	$37 \le k \le 71$ , s = 72 - k. Note that for $0 < k$	x≤36, this positioning
	scales (A) <sub>i</sub> "down"; for $37 < k \le 71$ , (A) <sub>i</sub> is scaled "up". If	
	k = 0, (A) was properly positioned before any shifting oper-	
	ations; if $k = 37$ , $(A)_i$ is all ones or zero	···
	Contact of Posisters and Change Posist	•

Storage Class Selection		Con			Storage Positions is Executed.	
		$(MC)_{\mathbf{f}}$ or $(MD)_{\mathbf{f}}$		(A) <sub>f</sub>	$(Q)_{\mathbf{f}}$	
u	v	u	v <sub>15-35</sub>	v <sub>0-14</sub>		
MC	MC or MD	No Change	No Change	k	D(u)·2 <sup>72-k</sup>	
or MD A Q		SCC FAULT				
	MC or MD	No Change	No Change	k	(A) $_{i} \cdot 2^{72-k}$ , $37 \le k \le 71$ (A) $_{i} \cdot 2^{-k}$ , $0 \le k \le 36$	
$ \begin{array}{c c} A & A \\ Q \end{array} $		SCC FAULT				
	MC or MD			k	(D(Q) · 2 <sup>72-k</sup>	No Change
$\left  \begin{array}{c c} Q & A \\ Q \end{array} \right $			scc i	FAULT		

Instruction: REPEAT (RPjnw) Operation Code: 75

Function: Execute the following instruction n times modifying the u and v addresses of the instruction to be repeated according to the value of j. Afterward, continue the program by the execution of the instruction at  $F_1$  whose v address is replaced by w.

(No change in registers or storage positions except the v address of  $F_1$  which is replaced by w.)

- Notes: 1. If the repeated instruction is a Threshold Jump (42uv) or an Equality Jump (43uv) and a jump occurs, the quantity j(n-r) from PAK is sent to the Q Register thus altering its contents.
  - 2. If the n of the Repeat instruction is a zero, the Normal Repeat Termination is executed immediately and the next instruction is taken from  $F_1$ .

Instruction:	EXTERNAL READ (ERjv) Operation Code: 76		
Function:	If $j = 0$ , replace the right-hand 8 bits of (v) with (IOA);		
	if $j = 1$ , replace (v) with (IOB). If t	he external unit util-	
	izes step-by-step operation, advance one step.		
	(No change in registers except as indic	ated by function above.)	

Instruction:	EXTERNAL WRITE (EWjv) Operation Code: 77		
Function:	If $j = 0$ , replace (IOA) with the right-hand 8 bits of (v);		
	if $j = 1$ , replace (IOB) with (v). Cause the previously		
	selected unit to respond to the information in IOA or IOB.		
	(No change in content of registers.)		