

WESTERN ELECTRIC REPOSITIONAL HEXADECIMAL PUNCH  
PROGRAM NO. J4-187

Author

J.H. Boatwright

Disclaimer:

"The authors of this program material, the Pool organization and General Precision believe this program to be correct; however, they bear no responsibility, financial or otherwise, for errors resulting from its use. This program is distributed only to individual and installation members of Pool. Further distribution of this manual and accompanying tapes for use by non-members is prohibited."

## WESTERN ELECTRIC REPOSITIONAL HEXADECIMAL PUNCH

J. H. Boatwright

April 18, 1960

**Purpose:** This routine punches out from any given area of memory a relative hex tape, with check sum, which may be loaded and repositioned by the Western Electric Repositional Hexadecimal Input.

**Operation:** Enter at starting address of this program.  
Enter the following data in decimal:

- (1) LoLf as xxxxyyyy'; the boundaries of the area to be punched out.
- (2) M as xxxx'; the modifier to be subtracted from each word W whose operand address Aw is  $Lo \leq Aw \leq Lf$
- (3) LohLfh as xxxxyyyy'; the boundaries of any area within LoLf which is to be punched in absolute form. Only one such entry allowed. The program uses 10.4 for the binarization of xxxx and yyyy.

A B.P.-16 halt will now occur. Either the high-speed or Flexowriter punch may be turned on. If the High-Speed Punch is used, depress B.P.-32.

Punching will then commence in the following order:

- (1) N' where N = the number of sectors to be punched between Lo and Lf in TTSS form.  $N = (Lf + 1) - Lo$ .

Example: If  $Lo = 5000$  and  $Lf = 5163$ ,  $Lf + 1 = 5200$ ,  
 $N = 0200$  or 128 sectors.

N is used by the Western Electric Repositional Hex Input to determine end of load. (N is punched with a 1 at 31)

- (2) Words whose operand addresses fall without LoLf and all I orders are punched with a 1 at a q of 31. A 1 at q = 31 is the same as the "X" in a decimal tape, i. e., this word is not to be modified on input. Words whose operand address fall within LoLf are punched with a "0" at a q of 31. A "0" at a q of 31 means that this word will be modified on input.
- (3) The check sum

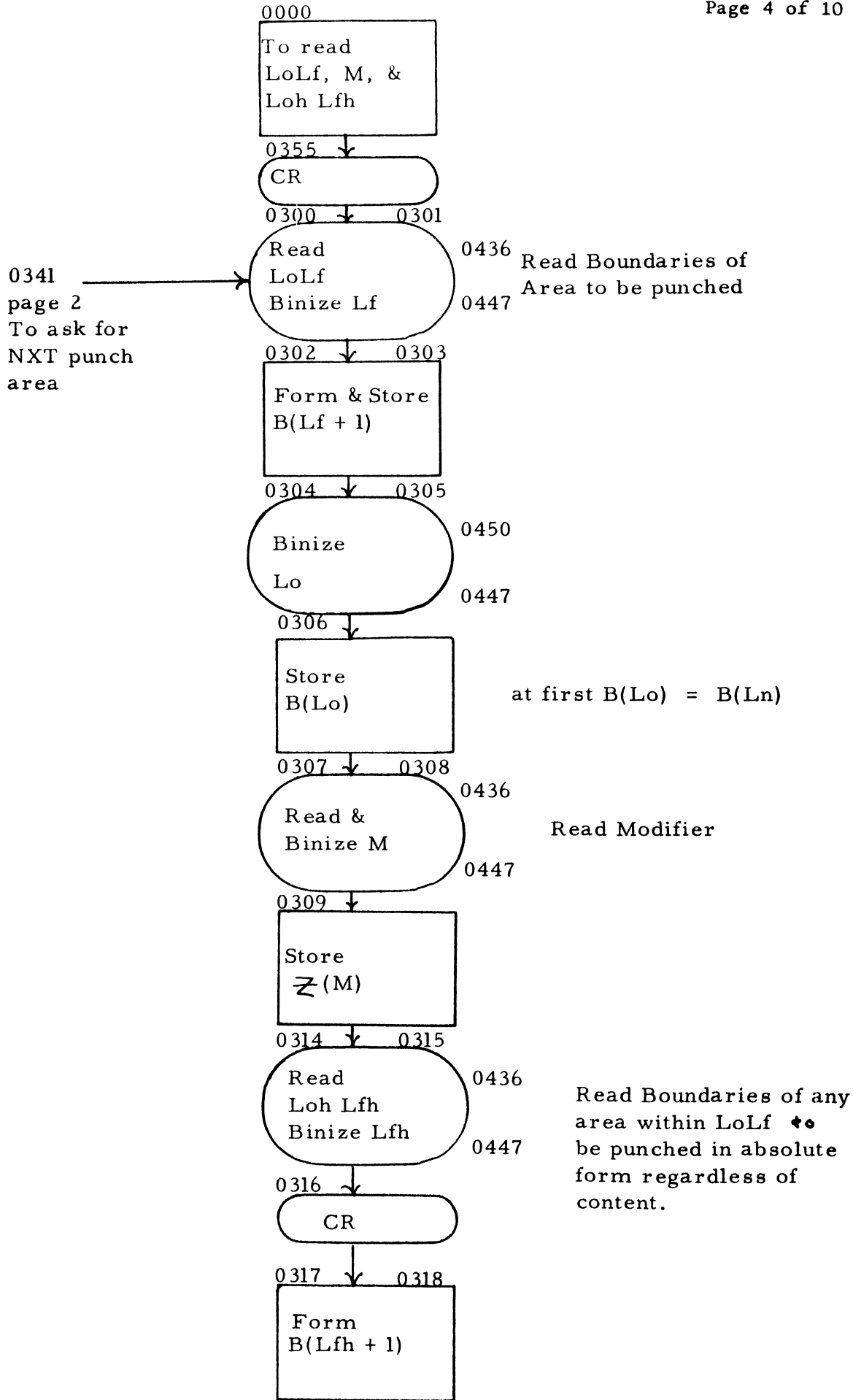
The check sum contains N and is punched with a 1 at 31.

Stops: BP-16 at 0332, Halt to turn on Punch.

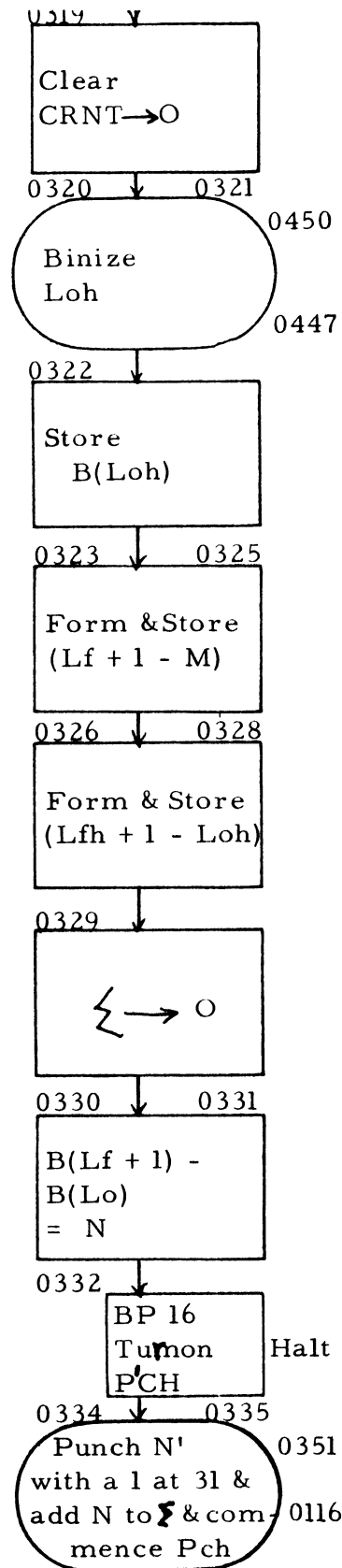
TC: TC up: 8 words per line; TC down: 14 words  
per line.

Storage: 5 Tracks

Time: On the High Speed Punch, about 48 seconds/  
track.



so that a CR will be given after N' is punched and CRNT will be set to proper CR constant.



N will be first number to be added to  $\Sigma$

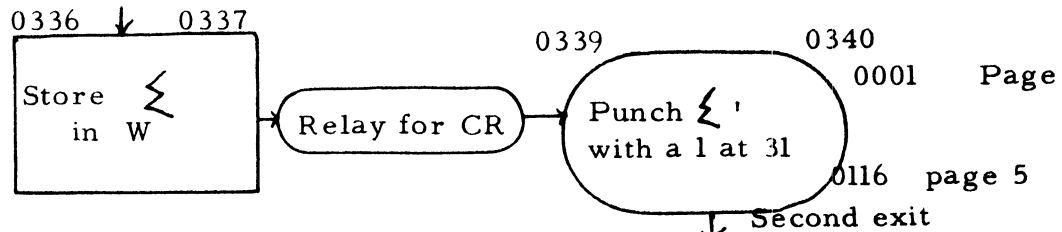
N = number of Sectors between Lf and Lo in TTSS Form,

Ex: if Lo = 5000  
Lf = 5163 . . Lf + 1 = 5200  
N = 5200 - 5000 = 0200  
N = 128 sectors or 0200 or 2 Tracks

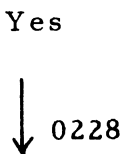
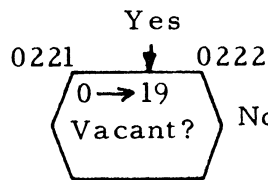
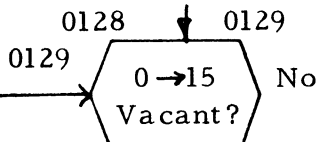
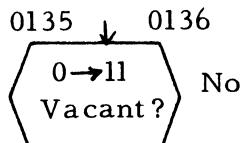
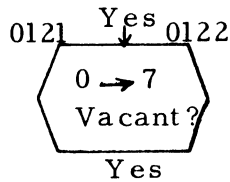
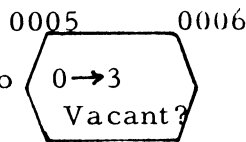
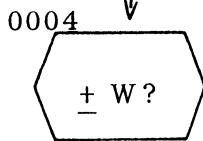
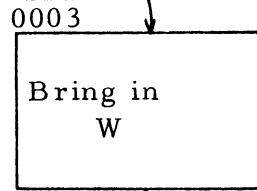
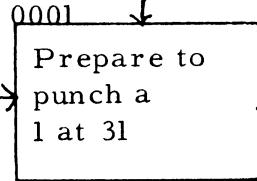
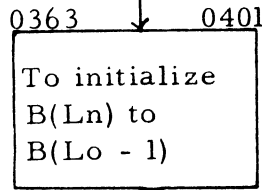
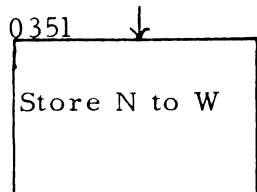
on pages  
3 - 7

punch out  
from Lo to  
Lf

First exit  
here when entire area  
has been punched



0300 page 1



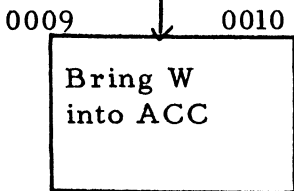
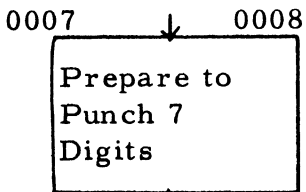
here to punch a hex word  
from page 6 and page 7

here to punch  $\Sigma$  page 2

here to punch

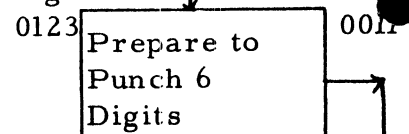
a - inst  
or - hex word  
from page 6

punch  
8 digits

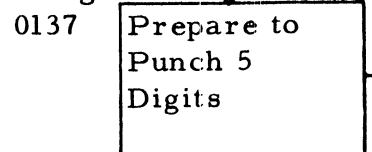


↓ 0012 page 4

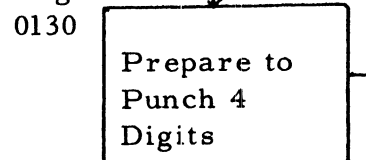
No → punch 7 digits → 0126



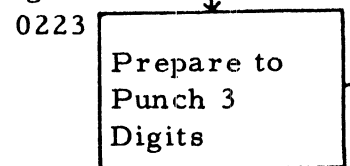
No → punch 6 digits → 0140



No → punch 5 digits → 0133



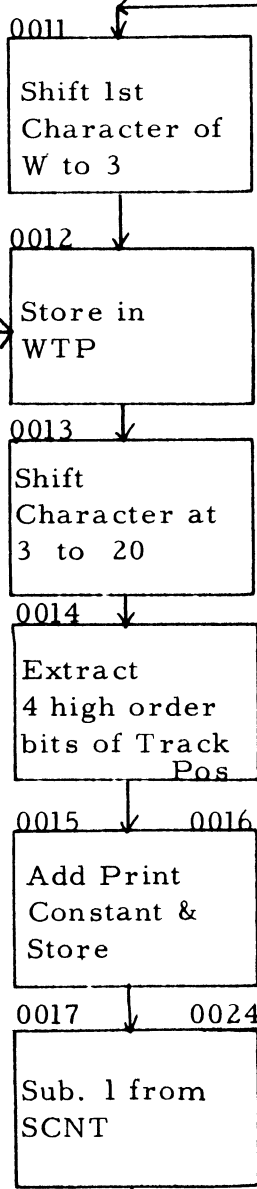
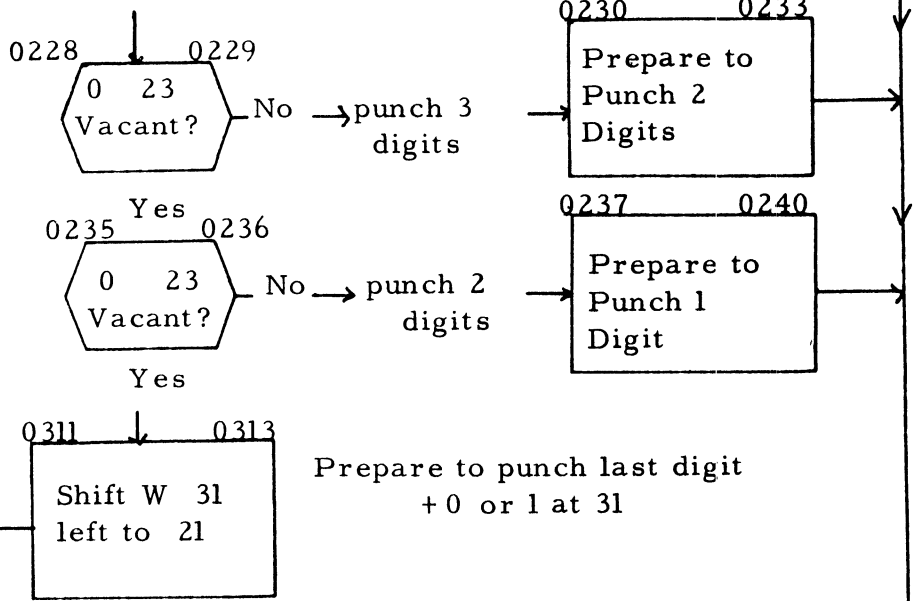
No → punch 4 digits → 0225



Page 4 0011

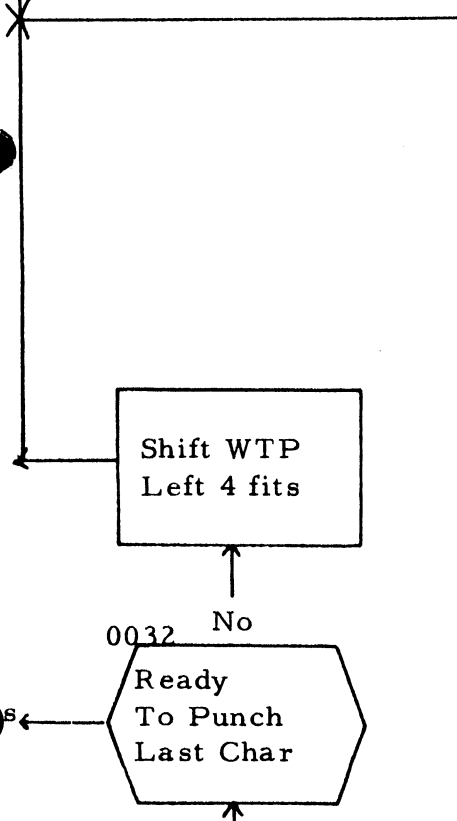
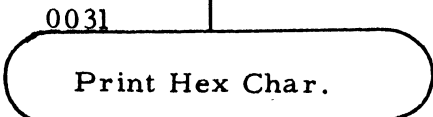
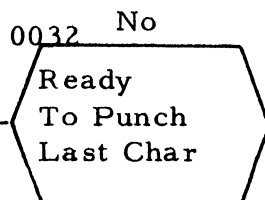
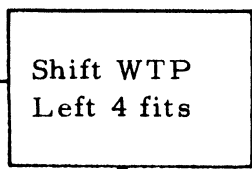
From page 3  
To punch 8 digits

page 5  
0042

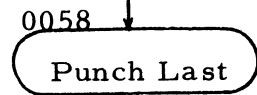
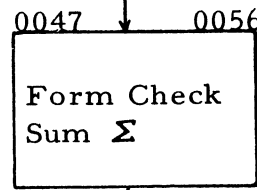
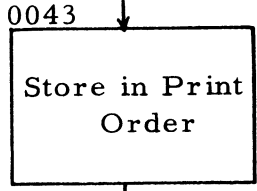
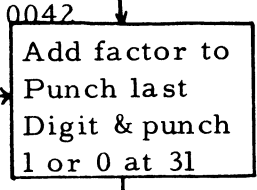
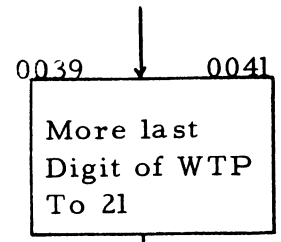


ACC contains a  
1 at a proper q  
to shift 1st char-  
acter of W to 3

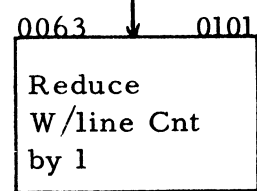
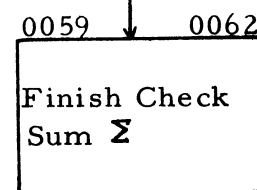
Hex punch all except  
last character



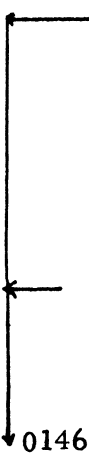
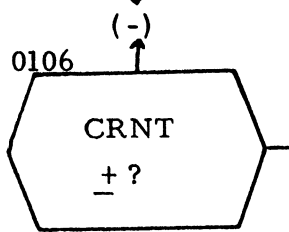
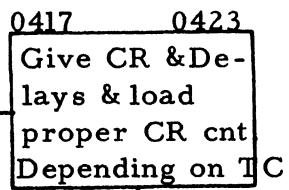
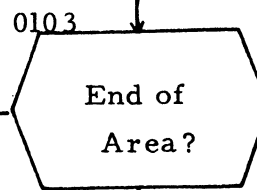
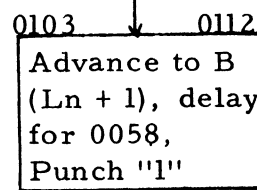
Page 4  
0313



Punch Last Char. with 1 or 0 at 31



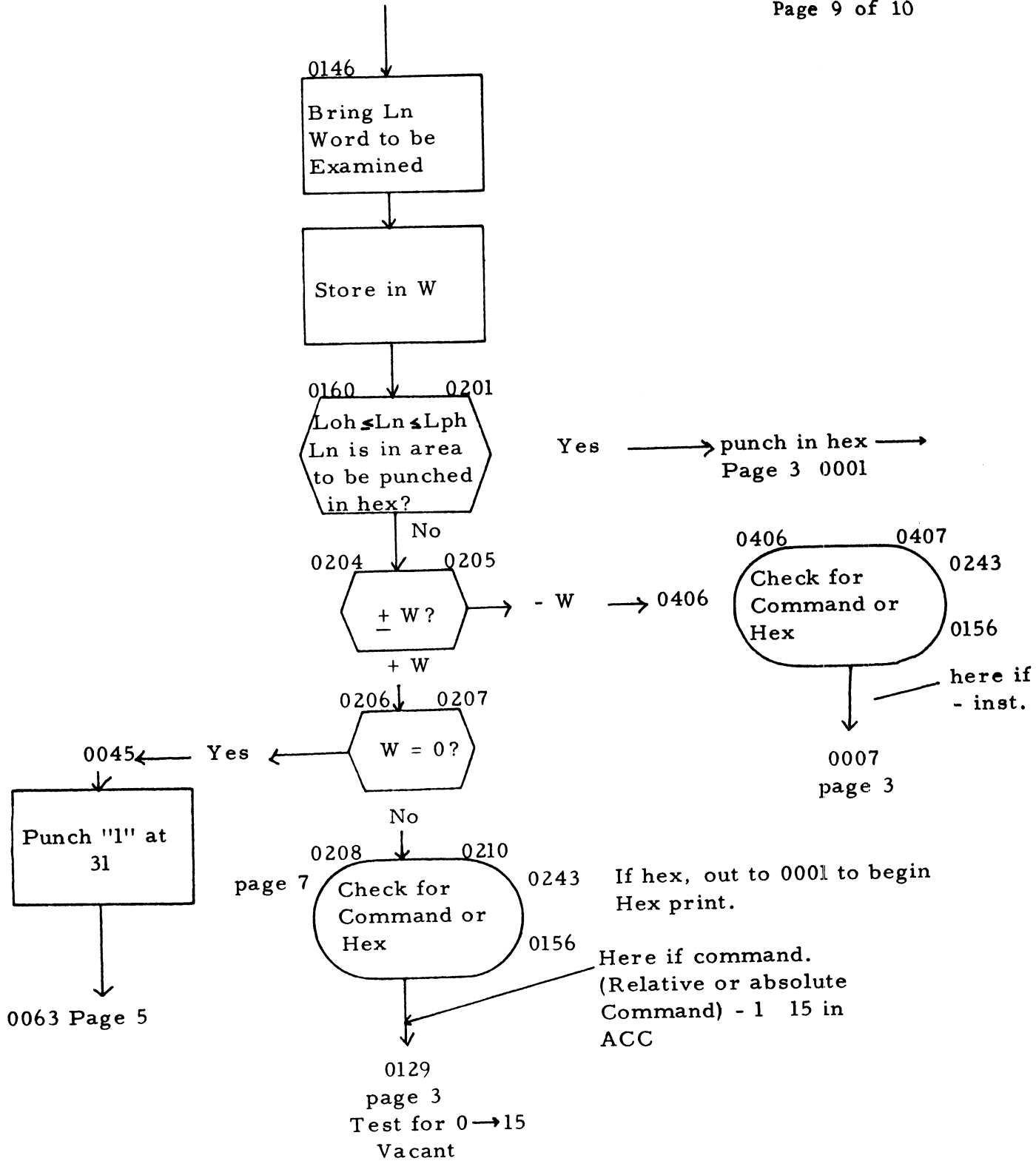
W/line cnt = CRNT counts for C.R.

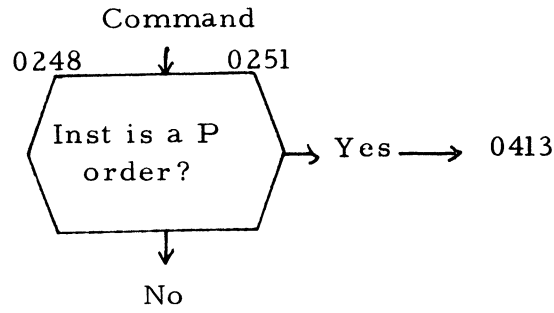
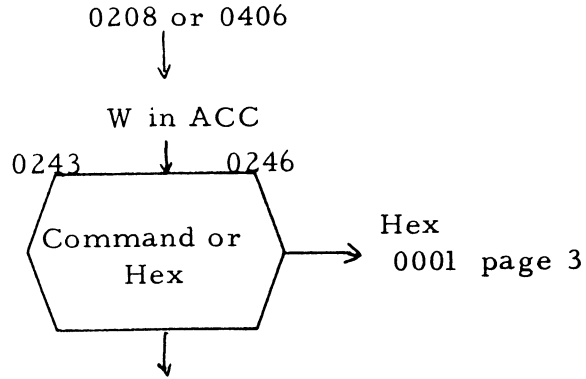


0116 U ( )

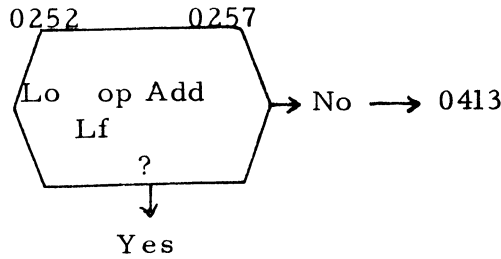
Exit set to:  
First, 0336, to Punch  $\Sigma$   
Second, 0341, after punching  $\Sigma$  to go to 0300



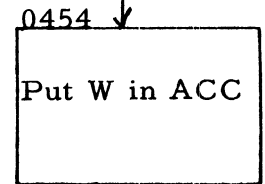




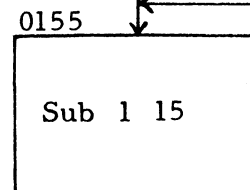
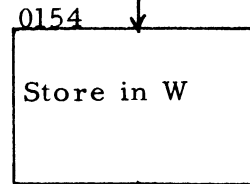
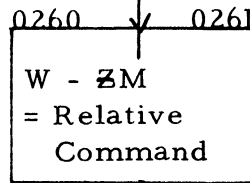
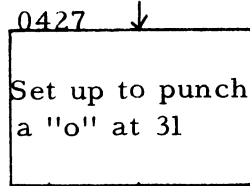
Should the command be punch as relative ?



Prepare to punch a "1" at an absolute Inst.



Prepare to punch a relative Inst.



Prepare test for 0→15 vacant

JOB NO. \_\_\_\_\_ PROG NO. \_\_\_\_\_ PREP BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

Repositional Hex Punch

TRACK

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		00	U 0 3 5 5			
		01	B 0 4 1 6			→ to read Lo Lf, M, and Loh Lfh XP 0609 Prepare to punch a word with 0246 0200 0402 0340
		02	C 0 4 5 7			P(L) 1 at 31
		03	B 0 4 3 3			W hex word
		04	T 0 0 0 7			W is (-) punch 8 digits
		05	S 0 0 2 0			1,3
		06	T 0 1 2 1			→ bits 0 → 3 are vacant
		07	B 0 0 2 2			6,29 Set up to punch 8
		08	C 0 4 0 3			S cnt digits
		09	B 0 4 3 3			W
		10	U 0 0 1 2			to begin Punch 8 digits
		11	N 0 4 3 3			W 31 shift 1st char acter of W to 3
		12	H 0 4 6 2			WTP Word to Print
		13	M 0 0 2 8			1,18 shift to 21 to track position
		14	E 0 0 2 9			XZ 6000 pull 4 high order track bits
		15	A 0 0 3 0			XP 0246 Form Digit
		16	C 0 0 3 1			P[ digit ] to punch
		17	B 0 4 0 3			S cnt
		18	S 0 3 4 7			1,29 Count strokes
		19	U 0 0 2 4			
0 0 0 0 0 0 0 2		20	1 0 0 0 0 0 0 0			1,3
		21	4 0 0 0 0 0 0 0			1,1
		22	X Z 0 0 0 6			6,29
0 0 0 0 0 0 0 1		23	3 w w w w w w w q			1,1 - 1,30
		24	H 0 4 0 3			S cnt
		25	U 0 0 3 1			
		26	X Z 0 0 0 0			Vacant
		27	X B 0 0 0 0			1,15
		28	X Z 3 2 0 0			1,18
		29	X Z 6 0 0 0			mask
		30	X P 0 2 4 6			to form print order
		31	P ( )			Punch a digit

CONDITIONAL STOP CODE

1 of 10

☒ CARRIAGE RETURN

JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		<input checked="" type="checkbox"/>				
		00, 32	T 0 0 3 9			→ to punch last digit
		33	B 0 4 6 2			WTP
		34	N 0 1 2 0			1,27 move left 4 bits
		35	U 0 0 1 2			<input checked="" type="checkbox"/> → to continue hex word punch
0,0,0,0,0,0,1		36	w,w,w,w,1,0,0,0			1,19 - 1,15 0221
		37				Vacant
		38	X,Z 0 0 0 2			2,29 0223
		39	X,Z 3 2 5 4			<input checked="" type="checkbox"/> delay for next to last digit ← 0039
		40	B 0 4 6 2			WTP 7
		41	M 0 1 2 7			1,14 move last digit to 21
		42	A 0 4 5 7			P(L) P(L) = XP 0609 or XP 0209 To punch 1 or 0 at 31 ← 0313
		43	Y 0 0 5 8			<input checked="" type="checkbox"/> P(Last digit)
		44	U 0 0 4 7			to form check sum Σ
here if W = 0		45	X,P 0 6 0 9			"1" also constant to punch 1 at 31 ← 0207
		46	U 0 0 6 3			→ end of hex word punch
		47	B 0 4 3 3			<input checked="" type="checkbox"/> W 0044
		48	T 0 2 1 2			-W to add 1 1
		49	S 0 0 2 1			1,1
		50	T 0 2 1 6			to add 1,1 ← 0213
		51	S 0 0 2 3			<input checked="" type="checkbox"/> 1,1 - 1,30
		52	T 0 2 1 6			to add 1,1
		53	A 0 4 3 2			1,30
		54	A 0 4 4 0			Σ ← 0217
		55	U 0 0 5 6			<input checked="" type="checkbox"/>
		56	S 0 0 2 1			1,1 ← 0055
		57	U 0 0 5 8			
		58	X,P [3 8 0 9]			punch last digit with 1 or 0 at 31 ← 0043 ← 0057
		59	T 0 2 1 9			<input checked="" type="checkbox"/> To add 1 1
		60	S 0 4 2 5			1,1 - 1,30
		61	T 0 2 1 9			To add 1,1
		62	C 0 4 4 0			Σ ← 0220
		63	B 0 4 4 9			<input checked="" type="checkbox"/> CRNT Word/line Counter ← 0046

JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		01	S 0 3 4 5		1029 } Reduce CRNT	
		01	C 0 4 4 9		CRNT } by 1	
		02	U 0 1 0 3		to check for end of Area	
		03	B 0 1 4 6		<input checked="" type="checkbox"/> B(Ln) advance to B(Ln + 1)	← 0102
		04	A 0 3 4 7		1 29	
		05	U 0 1 0 9		to delay for last digit	
		06	B 0 4 4 9		CRNT	← 0113
		07	T 0 4 1 7		<input checked="" type="checkbox"/> → time to give CR	
		08	U 0 1 4 6		NOT Time for CR → to examine next word	
		09	X Z 3 2 5 9		delay for last digit	← 0105
		10	H 0 1 4 6		B(Ln + 1)	
		11	S 0 4 2 6		<input checked="" type="checkbox"/> B(Lf + 1)	
		12	X P 3 2 2 7		"1"	
		13	T 0 1 0 6		→ NOT end of Area → to check for CR	
		14	X Z 0 0 3 9		end of area; delay for "1"	
		15	X P 1 6 0 1		<input checked="" type="checkbox"/> CR	
		16	U [ 0 3 4 1 ]		EXIT punching	0334 END AREA 0339
		17			Vacant	
(xz000 on tape)		18	Z 0 0 0 0		Modifier ZM	0253 0309 0260 0324
		19			<input checked="" type="checkbox"/> Vacant	
		20	X Z 0 0 0 4		1,27	0034
		21	S 0 1 5 7		1,7 - 1,3	← 0006
		22	T 0 1 3 5		→ bits 0 → 7 are vacant	
		23	B 0 2 0 2		<input checked="" type="checkbox"/> 5,29 Prepare to punch 7 digits	
		24	C 0 4 0 3		S cnt	
		25	B 0 1 5 2		1,27 to shift 1st character of W to B	
		26	U 0 0 1 1			
		27	X Y 0 0 0 0		<input checked="" type="checkbox"/> 1,14	0041 0451
		28	S 0 1 4 3		1,15 - 1,11	← 0136
		29	T 0 2 2 1		→ bits 0 → 15 are vacant	← 0211
		30	B 0 1 4 5		3,29 prepare to print 5 digits	
		31	C 0 4 0 3		<input checked="" type="checkbox"/> S cnt	

CG4155 M3B1T TORONTO, CANADA

CONDITIONAL STOP CODE

3 of 10



CARRIAGE RETURN

JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
	<input checked="" type="checkbox"/>	01 32	B 0 3 6 1		1,19	prepare shift 1st character of W
		33	U 0 0 1 1			to 3
		34	[		L <sub>f</sub> + 1 - M	0255 0325
		35	S 0 3 5 0	<input checked="" type="checkbox"/>	1,11 - 1,7	0122
		36	T 0 1 2 8		→ bits 0 → 11 are vacant	
		37	B 0 1 5 2		4,29	prepare to print 6
		38	C 0 4 0 3		S cnt digits	
		39	B 0 2 1 8	<input checked="" type="checkbox"/>	1,23	prepare to shift 1st character
		40	U 0 0 1 1			of W to 3
		41			Vacant	
		42			(L <sub>h</sub> +1) then later (L <sub>h</sub> +1 - L <sub>oh</sub> )	0163 0326 0318 0328
0 0 0 0 0 0 0 1		43	w w w 1 0 0 0 0	<input checked="" type="checkbox"/>	1,15 - 1,11	0128
		44			Vacant	
		45	X Z 0 0 0 3			0130
		46	X B [		0423 0363 0401 4110 0306 0103	B(L <sub>n</sub> ) Bring m Word to examine ← 0108
		47	H 0 4 3 3	<input checked="" type="checkbox"/>	W	0160 0331
		48	U 0 1 6 0			
0 0 0 0 0 0 0 3		49		2	1,30	0206
		50			Vacant	
		51		2	<input checked="" type="checkbox"/> 1,30	0208
		52	X Z 0 0 0 4		4,29	0125 0137
		53			NOT used	
here after forming Relative Address		54	H 0 4 3 3		W	→ 0262
		55	S 0 0 2 7	<input checked="" type="checkbox"/>	1,15	To TEST for 11 → 15 vacant → 0455
(W - ZM) or W in ACC		56	U 0 2 1 1			0209, 0406
0 0 0 0 0 0 0 3		57	w 1 0 0 0 0 0 0		1,7 - 1,3	0121
		58			Vacant	
		59			<input checked="" type="checkbox"/>	
		60	B 0 1 4 6		B(L <sub>n</sub> )	To see if W is within Aoh A <sub>ph</sub>
		61	S 0 4 1 2		B(L <sub>oh</sub> )	to be punched in hex regardless
		62	T 0 2 0 4			NOT in Hex Area
		63	S 0 1 4 2	<input checked="" type="checkbox"/>	(L <sub>h</sub> + 1) - (L <sub>oh</sub> )	



JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		<input checked="" type="checkbox"/>				
		02, 00	T 0 0 0 1		W within Aoh Aph - punch as hex	
		01	U 0 2 0 4		Aoh } W } Aph	0123
		02	X Z 0 0 0 5			
		03			<input checked="" type="checkbox"/> Vacant	
		04	B 0 4 3 3		W	← 0162 ← 0201
		05	T 0 4 0 6		-W either hex or -inst.	
		06	S 0 1 4 9		1,30	
		07	T 0 0 4 5		<input checked="" type="checkbox"/> → W = O	
		08	A 0 1 5 1		1,30	
		09	R 0 1 5 6		check for Command or hex and form	
		10	U 0 2 4 3		relative or absolute command	
		11	U 0 1 2 9		<input checked="" type="checkbox"/> → Command W-1 15 IN ACC	← 0156
		12	A 0 3 6 2		1,1 } Σ Routine	← 0048
		13	U 0 0 5 0		}	
		14			NOT USED	
		15	X B 0 0 0 0		<input checked="" type="checkbox"/> 1,15	0250
		16	A 0 3 5 9		1,1 } Σ Routine	← 0050 ← 0052
		17	U 0 0 5 4		}	
		18	X Z 0 1 0 0		1,23	0139
		19	A 0 3 6 2		<input checked="" type="checkbox"/> 1,1 } Σ Routine	← 0059 ← 0061
		20	U 0 0 6 2		}	
		21	S 0 0 3 6		1,19 - 1,15	← 0129
		22	T 0 2 2 8		→ 0-19 Vacant	
		23	B 0 0 3 8		<input checked="" type="checkbox"/> 2,29 Prepare to punch 4 digits	
		24	C 0 4 0 3		S cnt	
		25	B 0 3 5 4		1,15 prepare to move 1st character	
		26	U 0 0 1 1		of W to 3	
		27	X P 0 0 0 0		<input checked="" type="checkbox"/> constant to test for P order	0248 ← 0222
		28	S 0 3 4 3		1,23 - 1,19	
		29	T 0 2 3 5		→ O → 23 Vacant	
		30	B 0 3 4 5		1,29 prepare to punch 3 digits	
		31	C 0 4 0 3		<input checked="" type="checkbox"/> S cnt	



JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		02	B 0 4 0 4			prepare to move 1st character of W to
		32				
		33	U 0 0 1 1			q = 3
		34			Vacant	
		35	S 0 3 5 7		☒ 1,27 - 1,23	← 0229
		36	T 0 3 1 1		→ O → 27 vacant	→ to punch last digit
		37	C 0 4 5 9			zero ACC
		38	C 0 4 0 3			S cnt → O To punch 2 digits
		39	B 0 4 1 1		☒ 1,7 to shift	1st character of W to
		40	U 0 0 1 1			q = 0
		41			Vacant	
		42			Vacant	
		43	E 0 2 5 8		☒ 7WW0 J002	extract to zero Inst bits /
		44	S 0 2 5 9		1,30	← 0210 ← 0407
		45	T 0 2 4 7			→ Command
		46	U 0 0 0 1			→ Hex
Command →		47	B 0 4 3 3		☒ W	← 0245
		48	S 0 2 2 7			XP 0000
		49	T 0 2 5 2			→ not a P order
		50	S 0 2 1 5			XB 0000
		51	T 0 4 1 3		☒ → P order	→ punch absolute command
		52	E 0 4 3 1			XZ 2636 pull Track & Sector ← 0249
		53	S 0 1 1 8			ZM
check to see if		54	T 0 4 1 3			→ punch absolute command
$A_0 \leq A_w \leq A_f$		55	S 0 1 3 4		☒ (L <sub>f</sub> + 1 - m)	
		56	T 0 4 2 7			→ punch relative command
		57	U 0 4 1 3			→ punch absolute command
0 0 0 0 0 0 0 2		58	7 w w 0 j 0 0 2			MASK to zero Inst bits 0243
		59			☒ 1,30	0244
		60	S 0 1 1 8			ZM Form - ZM ← 0429
		61	A 0 4 3 3			W W-AM = relative command
		62	U 0 1 5 4			
		63			☒ Vacant	



J4-187

JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK D. BY \_\_\_\_\_

DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		<input checked="" type="checkbox"/>				
		03 00	R 0 4 4 7		> To read Lo Lf	← 0341, 0356
		01	U 0 4 3 6		BINIZE Lf make B(Lf)	← 0447
		02	A 0 3 4 7		1,29	
		03	C 0 4 2 6		<input checked="" type="checkbox"/> B(Lf + 1)	
		04	R 0 4 4 7		> BINIZE Lo	
		05	U 0 4 5 0			
		06	C 0 1 4 6		B(Ln) now, B(Ln) = B(Lo)	
		07	R 0 4 4 7		<input checked="" type="checkbox"/> > Read M	
		08	U 0 4 3 5		BIN M	
		09	Y 0 1 1 8		ZM	
		10	U 0 3 1 4			
		11	B 0 4 3 3		<input checked="" type="checkbox"/> W, 31	← 0-27 Vacant ← 0236
		12	N 0 4 3 4		1,21 shift left to 21	
		13	U 0 0 4 2		→ to punch last digit	
		14	R 0 4 4 7		> Read Loh Lfh	← 0310
		15	U 0 4 3 5		<input checked="" type="checkbox"/> binize Lfh	
		16	X P 1 6 0 0		CR	
		17	A 0 3 4 7		1,29	
		18	C 0 1 4 2		B(Lfh + 1)	
		19	C 0 4 4 9		<input checked="" type="checkbox"/> CRNT → O	
		20	R 0 4 4 7		> BIN	
		21	U 0 4 4 9		Loh	
		22	C 0 4 1 2		B(Loh)	
		23	B 0 4 2 6		<input checked="" type="checkbox"/> B(Lf + 1)	
		24	S 0 1 1 8		ZM	
		25	Y 0 1 3 4		(Lf + 1 - M)	
		26	B 0 1 4 2		B(Lfh + 1)	
		27	S 0 4 1 2		<input checked="" type="checkbox"/> B(Loh)	
		28	C 0 1 4 2		Lfh + 1 - Loh	
		29	C 0 4 4 0		Σ → O	
		30	B 0 4 2 6		B(Lf + 1)	Form Number of Sectors to punch in TTSS
		31	S 0 1 4 6		<input checked="" type="checkbox"/> B(Ln) Ln = Lo now form TTSS = N	



JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP.	ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		<input checked="" type="checkbox"/>					
		03	XZ	1600			Halt to turn on punch
			XP	1648			CR
			R	0116			> to print N' (with 1,31 & add N to Σ)
			U	0351		<input checked="" type="checkbox"/>	and commence punch out
here for end area			B	0440			Σ
			H	0433			W
			XZ	3224			delay
			R	0116		<input checked="" type="checkbox"/>	> Print Σ' with 1 at 31
			U	0001			
			U	0300			to read next Ao Af ←0116
			XZ	0000			Vacant
, 00000008			w w w w w	100		<input checked="" type="checkbox"/>	1,23 - 1,19 0228
							0100 0230
							Vacant
						<input checked="" type="checkbox"/>	0018 0104 0302 0317 0400
							Vacant
							0135
			w w w	1000000			1,11 - 1,7
			H	0433		<input checked="" type="checkbox"/>	W ← Prepare to p print N' ←0335
			U	0363			
			XZ	0000			Vacant
			XB	0000			1,15 0225
			XP	1600		<input checked="" type="checkbox"/>	CR to give initial CR ←0000
			U	0300			to read Ao Af
, 00000006			w w w w w w	10			1,27 - 1,23 0235
							Vacant
			4 0 0 0 0 0 0 0	0		<input checked="" type="checkbox"/>	1,1 0216
							Vacant
							0132
			4 0 0 0 0 0 0 0	0			1,1 0212 0219
			B	0146		<input checked="" type="checkbox"/>	To initialize B(Ln) to B(Lo-1) ←0352



JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_ PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_ DATE \_\_\_\_\_

Repositional Hex Punch

PROBLEM \_\_\_\_\_

TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
		04, 00	S 0 3 4 7		1,29	to compensate for advancing to/
		01	H 0 1 4 6		B(Lo - 1)	B(Ln + 1)
		02	U 0 0 0 1			during N <sup>1</sup> punch
0 0 0 0 0 0 0 3		03			⊗ S cnt stroke	0003 0017 0024 0124 cnt 0138 0224 0231 0238
		04	1 0 0 0 0 0		1,11	0232 0131
		05				0419
		06	R 0 1 5 6		← -W	← 0205
		07	U 0 2 4 3		⊗	To determine if command or hex and form absolute or Relative Instruction.
		08	U 0 0 0 7		→	Command, punch as 8 digits hex
		09	B 0 4 3 0		/3,29 ← TC	To punch 14 w/line ← 0420
		10	U 0 4 2 1			to store
0 0 0 0 0 0 0 1		11	1 0 0 0 0 0 0		⊗ 1,7	0239
		12	X B(0 0 0 0)		B(Loh)	0161 0327 ← 0322
		13	B 0 4 5 6		P0609 set as to print 1,31	← 0251, 0257 ← 0254
		14	C 0 4 5 7			P(L) in last digit as absolute
		15	U 0 4 5 4		⊗	
		16	X P 0 6 0 9			0001
here to DO CR		17	X Z 3 2 0 3			← 0107
		18	X P 1 6 3 3			CR
		19	B 0 4 0 5		⊗ 7,29	prepare to do 8 w/line
		20	8 0 0 T 0 4 0 9			← 0410
		21	C 0 4 4 9			CRNT
		22	X Z 3 2 3 7			delay
		23	U 0 1 4 6		⊗	
		24	X Z 1 6 0 0		1,19	
0 0 0 0 0 0 0 1		25	3 w w w w w w i q		1,1 - 1,30	0060
		26	X B[4 5 0 0		B(L <sub>f</sub> + 1)	0111 0303 0323 0330
		27	B 0 4 6 3		⊗ P0209	Set up to punch O at 31 ← 0256
		28	C 0 4 5 7			P(L) in last digit as relative inst
		29	U 0 2 6 0			
		30	X Z 0 0 1 3			to punch 14 w/line 0409
		31	X Z 6 3 6 3		⊗ MASK	0252 0445



JOB NO. \_\_\_\_\_ PROG. NO. \_\_\_\_\_

PREP. BY \_\_\_\_\_ CK'D. BY \_\_\_\_\_

DATE \_\_\_\_\_

PROBLEM Repositional Hex Punch

TRACK \_\_\_\_\_

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION OP. ADDRESS	STOP	CONTENTS OF ADDRESS	NOTES
0 0 0 0 0 0 2		04 32			1,30 0454 0003 0009 0011 0047 0147 0154	
		33			W 0204 0247 0261 0311 0337 0351	
		34	XZ 0 4 0 0		1,21 0312	
		35	XP 0 3 5 0		Space 0308	
		36	XZ 3 2 5 1		delay 0315	0301
		37	C 0 4 6 2		ACC → O	
		38	XP 0 0 5 3		Read	
		39	U 0 4 4 1			
		40			Σ	0054 0062 0329 0336
		41	XI 0 0 0 0		Input	← 0439
		42	H 0 4 6 2		W 1	
		43	XR 0 0 6 3		> to Binize	
		44	XU 0 0 5 0		in PIR 10.4	
		45	E 0 4 3 1		XZ 6363 pull track & sector	
		46	A 0 4 6 1		XB 0000 Form B( )	
		47	U [ ]		exit binize 0300 0304 0320 0307 0314	
		48	XZ 0 0 0 0		Vacant	
		49	XZ 0 0 0 8		CRNT ← 0063 0101 0106 0319 0321 0421	
		50	P 0 4 6 2		W 1	0305
		51	M 0 1 2 7		1,14 shift left half of ACC to right half	
		52	XU 0 0 5 1		to Binize in PIR 10.4	
		53			Vacant	
		54	B 0 4 3 3		W ← Punch absolute inst	← 0415
		55	U 0 1 5 5			
		56	XP 0 6 0 9		to form 1,31 0413	
		57	[ ]		P(L) 0002 0042 0414 0428	
		58			Vacant	
		59				0237
		60			Vacant	
		61	XB 0 0 0 0		1,15 0446	
		62	XZ 0 0 0 1		W1 first, then WTP 0012 0033 0040 0437 0442 0450	
		63	XP 0 2 0 9		to form O at 31 0427	

