B. Mersu

Bendix

Aerospace Systems Division

F. 64

INVESTIGATION OF ALSEP ADDRESS CODES	NO. ATM-843	REV. NO.
	PAGE ___ OF ___	
	DATE 2 December '69	

RECEIVED DEC 30 1969

SUMMARY

A computer simulation was generated to determine if one or more seven (7) bit code groups, other than those codes presently in use on ALSEP, are available for addressing additional ALSEP systems.

Two address code groups were found to be compatible with the eight existing ALSEP addresses during error free operation. Recommended addresses for 5A is (0110010) and for 5 B is (1100100).

The effect of a single bit error in transmission was also investigated using computer simulation techniques; first, for all code groups currently in use, then for the currently used codes plus the recommended codes.

The results indicate that another ALSEP can respond and execute a false command under certain single bit error conditions. The probability of this happening, however, is very low- best calculations indicate this probability to be approximately 10^{-10} to 10^{-11}.

Prepared by: Mitchell Goodkin
Mitchell Goodkin
E. E. Department

Approved by: R. J. Fatka
R. J. Fatka, Supervisor
Electronic Design Group
E. E. Department

Approved by: Warren Tosh
Warren Tosh, Supervisor
ALSEP Systems Engineering

REFERENCES

1. Command Decoder Specification
   AL 310800 C Amendment 1 - 5/14/68

2. ALSEP Address Codes
   ATM-696 - 9/20/67

3. Interface Control Specification for MSFN/ALSEP
   IC 314115 - 6/17/66

   Specification Change Notice No. 1 - 5/12/67

PROBLEM STATEMENT

The object of this investigation was to determine if one or more seven (7) bit code groups, other than the codes presently in use on ALSEP, are available for addressing additional ALSEP systems. A computer simulation program was to be used. The constraints applied in selecting the presently-used codes were to be considered and the extent of compromise incurred by partially relieving those constraints was to be assessed.

The codes presently in use on ALSEP will hereafter be referred to as the Existing ALSEP Addresses (or as the Existing Addresses).

RECOMMENDED ADDRESSES

Two (2) 7-bit binary numbers, (0110010) and (1100100), have been found that satisfy the constraints applied in selecting the presently used address codes.

These two numbers were found, by computer simulation of the command decoder, to be compatible with the eight Existing ALSEP Addresses. That is, if no bit-errors occur in the command word transmission (also referred to as the command message transmission), no erroneous command executions will take place.

Also, with the addition of a fifth ALSEP, using these numbers as the 5A and 5B addresses, the probability of an erroneous command execution by one of the ALSEPS will be approximately the same as when only four ALSEPS are in use.

All of the simulations and probability calculations were made with the assumption that all four (or more) of the ALSEP's were in operation.

BACKGROUND

When it is desired to have an ALSEP execute one of the allowed one-hundred (100) commands, a sixty-one (61) -bit serial command word (message) is transmitted to ALL of the ALSEP's. The command word format is shown in Figure 1. A twenty-bit preamble containing all ones (1's) is followed by a seven-bit address; then the seven-bit one's (digit) complement of the command code, the seven-bit command code, and twenty timing bits (all ones). Reference 1.

At some point in the Preamble during the serial input of the message each of the ALSEP's will "lock on" to the incoming signal. From that time (not necessarily exactly the same for all of the ALSEP's) each of the ALSEP's will start looking for seven (7) bits in succession that match one of its two addresses. It is likely that more than one of the ALSEP's will recognize its address. For instance, the command complement code may be the same as the address code for an ALSEP other than the one addressed. In that case the addressed ALSEP will recognize its address code at ADDRESS Bits 1 through 7 and the other will recognize its address seven bits later in the command word transmission.

To reduce the probability of erroneous command execution, not only must an address be recognized, the next fourteen bits must, also pass, a bit-by-bit parity check. In the command word, the seven bits preceding the command contain the bit-by-bit complement of the command. If Command Bit 1 is a "1" then Command Complement Bit 1 will be "0", etc. If there are no bit-errors in the transmission of the command word, a parity check starting at command complement Bit 1 will always pass, resulting in the execution of the command represented by the code in Command Bits 1 through 7. Note, though, that a parity check is not made by an ALSEP until <u>after</u> its address is recognized.

If some bit(s) transmitted to the ALSEP's is (are) in error, it is possible that one or more of the ALSEP's will recognize its address at some incorrect time and then make a parity check on the next fourteen bits that will pass. For example, suppose ALSEP-2 (B) were addressed with the Command Code $(0\ 111\ 110)_2$ and an error occurred in PREAMBLE Bit 18, then ALSEP-1 (A) will recognize its address starting at PREAMBLE Bit 17 and ending at ADDRESS Bit 3. The next 14 bits will pass a bit-by-bit parity check and the command corresponding to the command code $(0\ 001\ 011)_2$ will be erroneously executed (See Figure 3).

ADDRESS CODE SELECTION - PROCEDURE

The procedure followed to obtain the recommended address codes is illustrated in Figure 2. First, all of the 128 7-bit binary numbers were listed. Eighty-one of the numbers were eliminated for the reasons enumerated in Table 2. The remaining 47 numbers, called Candidate Addresses, are listed in Table 3. The blank spaces in Table 3 were occupied by numbers that should have been eliminated in the first pass.

The Hamming Distance between each of the 47 Candidate Addresses and each of the Existing ALSEP Addresses was checked. The Existing ALSEP Addresses are listed in Table 1, and the results of the Hamming Distance checks are shown in Table 4.

The Candidate Addresses that were at least a Hamming Distance of three (3) away from ALL of the Existing Addresses were included in the group of Possible Addresses, listed in Table 5. Next, the Hamming Distance between all pairs of the Possible Addresses was determined. The results of that last Hamming Distance check, listed in Table 5, were used to determine what combinations of the seven Possible Addresses might be used as ALSEP addresses. Note, for example, that if Possible Address #5 were used, then only one pair of addresses satisfying the Hamming Distance three (3) constraint would be available.

Next, a computer simulation of the command decoder which assumed error-free transmission was used to check the compatibility of each of the seven Possible Addresses with all of the eight Existing Addresses. It was found with the first run of this computer simulation that Possible Addresses #4 and #6 were not compatible with the Existing Addresses. If either of those two numbers were used as an ALSEP address, with no bit-errors in transmitting the command word, erroneous command executions could occur. Further use of the ERROR-FREE-TRANSMISSION COMPUTER SIMULATION showed that, if no bit-errors occurred in the transmission of the command word, all of the remaining five Possible Addresses (#1, 2, 3, 5, 7) could be used, at the same time, with the eight Existing ALSEP Addresses with no erroneous command executions occurring. Those remaining five Possible Addresses were relabled and listed in Figure 2 under the title "Compatible Addresses".

Since Possible Address #5 (Compatible Address #4) was not at least a Hamming Distance of three away from at least two of the other Compatible Addresses, it was not tested further. Also, Compatible Address #1 was not tested further since it was finally ("better late than never") noticed that that number was only a Hamming Distance of one (1) away from the alternate one-zero pattern that is periodically transmitted. Note that with a relaxation of the Hamming Distance constraint that Compatible Address #4 would be a prime candidate for use as an ALSEP address.

The eight Existing ALSEP Addresses and Compatible Addresses #2, 3, and 5 were tested with the ERROR-IN-TRANSMISSION COMPUTER SIMULATION. That simulation was essentially the same as the ERROR-FREE TRANSMISSION COMPUTER SIMULATION except that the effects of a single bit-error in each of the command word bits were determined. With a single-bit-error in transmission, the two Recommend Address, listed at the bottom of Figure 2, behaved as well as the Existing Addresses and did not degrade the overall behavior of the group.

ERROR-IN-TRANSMISSION SIMULATION RESULTS

The results of the ERROR-IN-TRANSMISSION SIMULATION are listed in Table 6 and 7. Note that at the top of Table 6-Sheet 1 there is a circled number above each column. Table 6 might be read in the following way: if it is intended that ALSEP 1 execute Command 2 and a transmission error occurs in bit 3 of the command message, then ALSEP 4 will execute command 5 . The ALSEP 4 address will be recognized starting at bit 6 of the serially input command message.

An erroneous Command acceptance (resulting in an erroneous execution) is illustrated in Figure 3. Note that the error, in PREAMBLE Bit 18, is shown in a thick-sided rectangle. Since $X_i = 0$ or 1 for all i, there are eight (8) 7-bit binary codes that could pass the parity check:

Intended Command		Erroneously Executed Command
0 001 110		0 001 000
0 011 110		0 001 001
0 101 110		0 001 010
0 111 110		0 001 011
.		.
.		.
.		.
1 111 110		0 001 111

Of the eight intended commands, octal command codes (016), (116), and (176) are not allowed. Also, of the eight "erroneously executed commands", octal command codes (101) and (016) are not allowed; therefore, only four out of the eight code combinations of X_1 X_2 X_3 can result in an erroneous command exectuion.

To obtain a quick upper bound on the probability of an erroneous command execution by each of the ALSEP's, calculations were made which assumed that all 4 (or 5) of the ALSEP's were in operation, that any address was equally probable, and that any command was equally probable. Also, the "not allowed" commands were not eliminated. With or without an ALSEP 5 (using the two Recommended Addresses), the probability of an erroneous command execution by any given ALSEP is less than or equal to approximately 10^{-10}. Roughly, the probability may vary between 3×10^{-10} and 10^{-11}. A sample calculation has been included, following Table 7.

ADDITIONAL ADDRESS POSSIBILITIES

If Possible Address #7 (Table 5 - Also listed as Compatible Address #5, Figure 2) were used as an additional ALSEP address, the added ALSEP command decoder would have two (2) to three (3) times the probability of erroneously accepting a command as would anyone of the other ten decoders (1A, 1B, ... , 5A, 5B) without the addition of Possible Address #7.

If the one-zero pattern that is periodically transmitted is less than seventeen (17) bits long, Possible Address #1 (Table 5 - Compatible Address #1, Figure 2) may also be an acceptable address.

Since the probability of erroneous command acceptance with a one-bit (1-bit) error in the command word (message) transmission is much greater than the probability of a two-bit (2-bit) error in the command word transmission, the address selection constraints might be relaxed to a Hamming Distance two (2) separation. If the Hamming Distance constraint is relaxed to a separation of two (2), Table 4 may be used to pick numbers from Table 3 for further testing with the Existing ALSEP Addresses.

If the Hamming Distance constraint were relaxed to two (2), Compatible Address #4 might be a prime candidate for use as an ALSEP address. The reason for this is explained under the heading "ADDRESS CODE SELECTION-PROCEDURE".

COMMAND DECODER SIMULATIONS - GENERAL DESCRIPTIONS

The ERROR-FREE-TRANSMISSION Simulation Program assumed that no transmission errors occurred. The program simulated the serial input of the command word into each of the ALSEP Command Decoders, the check by each of the Command Decoders for address recognition, and the bit-by-bit parity check of the next fourteen bits if an address was recognized. If the parity check passed, the program would print out sufficient data to reconstruct the situation that would cause the erroneous command execution.

Each of the Command Decoder addresses (i. e., A and B) was treated as an independent ALSEP. The affect of that simplification was to allow the A (B) or B (A) half of a Command Decoder to respond after the other half had already recognized its address.

The command word included, in succession, each of the addresses under test (including the Existing eight) in combination with each of the 128 possible 7-bit codes that could be inserted into the seven (7) Command bits. For each address-command combination, a comparison against each of the addresses under test, including the eight Existing ALSEP Addresses, was made starting at each bit position of the "incoming" command word.

The ERROR-IN-TRANSMISSION Simulation Program was essentially the same as the other simulation except that a single bit-error was included, in succession, in each position of the command word. That is, the ERROR-FREE-TRANSMISSION Simulation Program was run with the content of the error position in the command word containing the complement of the correct bit value. The error position was varied from Preamble Bit 14 through Timing Bit 20. A quick visual inspection of the command word format would show that an erroneous command execution could not be caused by a single bit-error in any of the first thirteen Preamble bits.

ERROR-FREE-TRANSMISSION SIMULATION PROGRAM

Appendix A contains an illustration, entitled COMPUTER STORAGE, of the major ERROR-FREE-TRANSMISSION Program variables.

The Address List contains all of the addresses to be tested in a given computer run. The number of address, R, is a variable read into the computer prior to reading the list of addresses.

The Command Storage is a "seven-bit" binary counter. Each "bit" is stored in a separate computer storage location. A counter simulation routine in the program increments the command from (0, 0, 0, 0, 0, 0, 0) binary to (1, 1, 1, 1, 1, 1, 1) binary by steps of one (1), and resets the counter. The counter goes through a complete cycle for every address inserted in the command word. (Actually, the addresses are varied in the inner program loop and the counter is stepped in the outer loop.)

The Command Complement contains the one's complement of the command. The command complement is loaded after each change in the command.

The Command Word: (Abbreviated) is a 48-bit binary vector. Note that the first thirteen Preamble bits have not been included in the simulation and that the command word bits have been renumbered with W(I) equal to command word bit (I + 13).

Whenever a parity check is passed, a printout occurs which includes the command code that passed the parity check ("command recognized"), the intended command (in decimal form, labeled "Command in Command Storage" or "CICW") and the numbers "I", "J", and "K". The number "I" indicates the starting position in the abbreviated command word of the recognized address. The numbers "J" and "K" refer to the address list which appears at the beginning of the output for each run. The number "J" is the number of the recognized address, and "K" is the number of the intended address.

Also in Appendix A are the flow chart and program listing for the ERROR-FREE-TRANSMISSION Simulation Program. The program was written in the GE 400 Series Basic Language.

ERROR-IN-TRANSMISSION SIMULATION PROGRAM

The major variables of the ERROR-IN-TRANSMISSION Program are the same as for the ERROR-FREE Program except that it was found that the Command Complement Storage could be eliminated (complement of command obtained while loading abbreviated command word). Also, a new variable, "ERPOS"(Error Position), was added to mark the position of error insertion.

In the ERROR-IN-TRANSMISSION Program, only those conditions affected by the error were tested. Those conditions are outlined in Appendix B "Program Execution Parameters". Also in Appendix B are a list of computer storage (of the variables), a flow diagram showing the modifications made of the ERROR-FREE Program, and a program listing. The program was written in the IBM System/360 Fortran IV language.

COMMAND WORD FORMAT AND TRANSMISSION

TRANSMISSION ORDER:	1	...	20	21	...	27	28	...	34	35	...	41	42	...	61	62	.....
BIT CONTENT:	PREAMBLE			ADDRESS			COMMAND COMPLEMENT			COMMAND			TIMING			TRAILING	
BIT NUMBER:	1	...	20	1	...	7	1	...	7	1	...	7	1	...	20	1	.....

SYNCHRONIZATION COMPLETED SOMETIME BETWEEN PREAMBLE BITS 12 THROUGH 18.

FIGURE 1

ADDRESS CODE SELECTION - FLOW CHART

128 7-BIT BINARY NUMBERS

↓ IMMEDIATE "ADDRESS" ELIMINATION - (TABLE 2)

47 CANDIDATE ADDRESSESS (TABLE 3)

↓ HAMMING DISTANCE CHECK AGAINST EXISTING EIGHT ALSEP ADDRESSES (TABLE 4)

7 POSSIBLE ADDRESSES (TABLE 5)

↓ ERROR-FREE-TRANSMISSION COMPUTER SIMULATION

5 COMPATIBLE ADDRESSES

1) 0100010
2) 0110010
3) 1100100
4) 1100010
5) 1000011

↓ ERROR-IN-TRANSMISSION COMPUTER (& MANUAL) SIMULATION

2 RECOMMENDED ADDRESSES

1) 0110010 - ALSEP 5A
2) 1100100 - ALSEP 5B

FIG. 2

EXAMPLE 1 — CLARIFICATION OF TABLE 6

SAMPLE CASE: FIRST ENTRY (ROW 1)

BIT CONTENT	PREAMBLE							INTENDED ADDRESS							COMMAND CMPLT							INTENDED COMMAND							TIMING							
BIT NUMBER	14	15	16	17	18	19	20	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8 .....
ERROR-FREE WORD	1	1	1	1	1	1	1	0	0	0	1	1	1	0	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_3$	0	0	0	1	x_1	x_2	x_3	1	1	1	0	1	1	1	1	1	1	1	1 .....
WORD WITH ERROR	1	1	1	1	$\boxed{0}$	1	1	0	0	0	1	1	1	0	$\bar{x}_1$	$\bar{x}_2$	$\bar{x}_3$	0	0	0	1	x_1	x_2	x_3	1	1	1	0	1	1	1	1	1	1	1	1 .....
ADDRESS RECOGNITION AND COMMAND VERIFICATION				1	0	1	1	0	0	0								0	0	0	1	x_1	x_2	x_3												
				ADDRESS RECOG.														COMMAND ACCEPTED BY 1A																		

EXAMPLE OF COMMAND INTERPRETATION: $x_i = 0,1$ FOR ALL i

FOR $x_1 = 0$ $x_2 = 1$ $x_3 = 1$

IF IT IS INTENDED THAT ALSEP 2B EXECUTE COMMAND

0 111 110

THEN ALSEP 1A WILL EXECUTE

0 001 011

NOTE THAT, IN THIS EXAMPLE, THE INTENDED COMMAND EXECUTION WILL ALSO TAKE PLACE.

ERROR POSITION : PREAMBLE BIT 18
ALSEP 2B ADDRESS : 0001110
ALSEP 1A ADDRESS : 1011000

FIGURE 3

TABLE 1
EXISTING ADDRESSES

ALSEP (Command Decoder Number)	Binary Address Code Number	Address Number (For Simulation Programs)
1A	1011000	1
1B	0011000	2
2A	1001110	3
2B	0001110	4
3A	1101001	5
3B	0101001	6
4A	0010101	7
4B	0110101	8

TABLE 2

REASONS FOR IMMEDIATE "ADDRESS" ELIMINATION

1. Alternate 1, 0's or a Hamming Distance of one from alternate 1, 0's.
2. Already used as one of the eight existing addresses.
3. A Hamming Distance of one from one of the eight existing addresses.
4. A string of one's followed by a string of zeroes or visa versa.
5. All zeroes.
6. All ones.
7. Hamming Distance one from $(1111111)_2$.
8. Hamming Distance one from $(0000000)_2$.

TABLE 3

CANDIDATE ADDRESSES – ADDRESSES NOT ELIMINATED IN FIRST PASS THROUGH THE 128 7-BIT CONFIGURATIONS

1)	———	(29)	0100100
2)	1100010	(30)	0100110
3)	1000101	(31)	0101111
4)	0001011	(32)	0110010
5)	0010110	(33)	0110011
6)	0101100	(34)	0110110
7)	0110000	(35)	0111011
8)	1000001	(36)	0111100
9)	0011101	(37)	0111110
10)	0111010	(38)	1000010
11)	———	(39)	1000100
12)	1010011	(40)	1000111
13)	0100111	(41)	1001011
14)	———	(42)	1001101
15)	———	(43)	1010001
16)	1100001	(44)	———
17)	1000011	(45)	1011011
18)	1010010	(46)	1011101
19)	0100101	(47)	1100011
20)	———	(48)	1100100
21)	1010110	(49)	1100101
22)	———	(50)	1100110
23)	0001101	(51)	1100111
24)	0010010	(52)	1101100
25)	0010011	(53)	1110001
26)	0011011	(54)	1111010
27)	0100010		
28)	0100011		

TABLE 4 SHEET 1 OF 3

CANDIDATE ADDRESSES – HAMMING DISTANCE CHECK AGAINST EXISTING ALSEP ADDRESSES

HAMMING DISTANCE

EXISTING ALSEP ADDR. / CANDIDATE ADDRESSES	ALSEP 1A	1B	2A	2B	3A	3B	4A	4B	NUMBER OF ALSEP ADDRS WITH HAMMING DIST. LESS THAN 3 FROM CANDIDATE ADDR.
(1)								1	
(2)	4	5	3	4	3	4	6	5	0
(3)	4	5	3	4	3	4	2	3	1
(4)	4	3	3	2	3	2	4	5	2
(5)	4	3	3	2	7	6	2	3	2
(6)	4	3	3	2	3	2	4	3	2
(7)	3	2	6	4	4	3	3	2	2
(8)	3	4	4	5	2	3	3	4	1
(9)							1		
(10)	3	2	4	3	4	3	5	4	1
(11)									ALSEP 3A ADDR.
(12)	3	4	4	5	4	5	3	4	0
(13)	7	6	4	3	4	3	3	2	1
(14)		1							
(15)		1							
(16)	4	5	5	5	2	2	4	3	2
(17)	4	5	3	4	3	4	4	5	0
(18)	2	3	3	4	5	6	4	5	1
(19)								1	
(20)			1						
(21)	3	4	2	3	6	7	3	4	1
(22)						1			

TABLE 4 - SHEET 2 OF 3

CANDIDATE ADDRESSES – HAMMING DISTANCE CHECK AGAINST EXISTING ALSEP ADDRESSES

HAMMING DISTANCE

CANDIDATE ADDRESSES \ EXISTING ALSEP ADDR	ALSEP 1A	1B	2A	2B	3A	3B	4A	4B	No of ALSEP ADDRESSES WITH A HAMMING DISTANCE LESS THAN 3 FROM THE CANDIDATE ADDRESS.
(23)	4	3	3	2	3	2	2	3	3
(24)	3	2	4	3	6	5	3	4	1
(25)	4	3	5	4	5	4	2	3	1
(26)	3	2	4	3	4	3	3	4	1
(27)	5	4	4	3	4	3	5	4	0
(28)	6	5	5	4	3	2	4	3	1
(29)	5	4	4	3	4	3	3	2	1
(30)	6	5	3	2	5	4	4	3	1
(31)	6	5	3	2	3	2	4	3	2
(32)	4	3	5	4	4	4	4	3	0
(33)	5	4	6	5	4	3	3	2	1
(34)	5	4	4	3	5	5	3	2	1
(35)	4	3	5	4	3	2	4	3	1
(36)	3	2	4	3	4	3	3	2	2
(37)	4	3	3	2	5	4	4	3	1
(38)	3	4	2	3	4	5	5	6	1
(39)	3	4	2	3	4	5	3	4	1
(40)	5	6	2	3	4	5	3	4	1
(41)	3	4	2	3	2	3	5	6	2
(42)	3	4	2	3	2	3	3	4	2

TABLE 4 SHEET 3 OF 3

CANDIDATE ADDRESSES - HAMMING DISTANCE CHECK
AGAINST EXISTING ALSEP ADDRESSES

HAMMING DISTANCE

EXISTING ALSEP ADDRS. / CANDIDATE ADDRESSES	ALSEP 1A	1B	2A	2B	3A	3B	4A	4B	NUMBER OF ALSEP ADDRS. WITH HAMMING DISTANCE LESS THAN 3 (THREE) FROM THE CANDIDATE ADDRESS.
(43)	2	3	5	6	3	4	2	3	2
(44)	4	5	3	4	5	6	2	3	1
(45)	2	3	3	4	3	4	4	5	1
(46)	2	3	3	3	3	4	2	3	2
(47)	5	6	4	5	2	3	5	4	1
(48)	4	5	3	4	3	4	4	3	0
(49)	5	6	4	5	2	3	3	2	2
(50)	5	6	2	3	4	5	5	4	1
(51)	6	7	3	4	3	4	4	3	0
(52)	3	4	2	3	2	4	5	3	2
(53)	3	4	6	7	2	3	3	2	2
(54)	2	3	3	4	3	4	6	5	1

TABLE 5

POSSIBLE ADDRESSES: NUMBERS THAT MIGHT BE USED AS NEW ADDRESSES - PENDING CHECK BY SIMULATION PROGRAM

THESE ADDRESSES CANNOT ALL BE USED SIMULTANEOUSELY.

(1) 0100010

(2) 0110010

(3) 1100100

(4) 1100111

(5) 1100010

(6) 1010011

(7) 1000011

HAMMING DISTANCE CHECK - BETWEEN THE POSSIBLE ADDRESSES

POS. ADDRESS / POS. ADDRESS	(1)	(2)	(3)	(4)	(5)	(6)	(7)	
(1)	X	1	3	3	1	4	3	
(2)	1	X	4	4	2	3	4	
(3)	3	4	X	2	2	5	4	
(4)	3	4	2	X	2	3	2	
(5)	1	2	2	2	X	3	2	
(6)	4	3	5	3	3	X	1	
(7)	3	4	4	2	2	1	X	

TABLE 6 - SHEET 1 OF 2

④ ALSEP ERRONEOUSLY RECOGNIZING ADDRESS	① INTENDED ALSEP ADDRESS	⑤ ERRONEOUS COMMAND EXECUTED (BINARY)	② INTENDED COMMAND (BINARY)	③ POSITION OF ERROR IN THE COMMAND WORD	⑥ STARTING POSITION IN COMMAND WORD OF ADDR. RECOG.
1A	2B	$0001x_1x_2x_3$	$x_1x_2x_3 1110$	PREAMBLE BIT 18	PREAMBLE BIT 17
1B	———	NONE	———	ANY SINGLE BIT	———
2A	3A OR 3B	$x_1x_2x_3x_4 111$	$000x_1x_2x_3x_4$	CMND CMPLT BIT 3	ADDRESS BIT 4
2A	3A OR 3B	$x_1x_2x_3x_4 111$	$001x_1x_2x_3x_4$	COMMAND BIT 3	ADDRESS BIT 4
2A	3A OR 3B	$x_1x_2x_3x_4 110$	$001x_1x_2x_3x_4$	TIMING BIT 3	ADDRESS BIT 4
2A	5B	$x_1x_2x_3 1111$	$0000x_1x_2x_3$	CMND CMPLT BIT 4	ADDRESS BIT 5
2A	5B	$x_1x_2x_3 1111$	$0001x_1x_2x_3$	COMMAND BIT 4	ADDRESS BIT 5
2A	5B	$x_1x_2x_3 1110$	$0001x_1x_2x_3$	TIMING BIT 4	ADDRESS BIT 5
2B	1A OR 1B	$x_1x_2x_3 1111$	$0000x_1x_2x_3$	CMND CMPLT BIT 4	ADDRESS BIT 5
2B	1A OR 1B	$x_1x_2x_3 1111$	$0001x_1x_2x_3$	COMMAND BIT 4	ADDRESS BIT 5
2B	1A OR 1B	$x_1x_2x_3 1110$	$0001x_1x_2x_3$	TIMING BIT 4	ADDRESS BIT 5
3A	1B	$x_1x_2x_3 1000$	$0111x_1x_2x_3$	PREAMBLE BIT 19	PREAMBLE BIT 17
3A	2A	$x_1x_2x_3x_4 110$	$001x_1x_2x_3x_4$	PREAMBLE BIT 20	PREAMBLE BIT 18
3A	4A	$x_1x_2x_3 0101$	$1010x_1x_2x_3$	PREAMBLE BIT 19	PREAMBLE BIT 17
3A	4B	$x_1x_2x_3x_4x_5x_6 1$	$0x_1x_2x_3x_4x_5x_6$	ADDRESS BIT 7	ADDRESS BIT 2
3A	5B	$x_1x_2x_3x_4x_5x_6 1$	$0x_1x_2x_3x_4x_5x_6$	ADDRESS BIT 3	ADDRESS BIT 2
3B	4A	$x_1x_2x_3x_4x_5x_6 1$	$0x_1x_2x_3x_4x_5x_6$	ADDRESS BIT 7	ADDRESS BIT 2
4A	2A OR 2B	$x_1x_2x_3x_4x_5x_6 1$	$0x_1x_2x_3x_4x_5x_6$	ADDRESS BIT 5	ADDRESS BIT 2
4A	ANY ADDRESS	1010101	ANY	TIMING BIT 20* OR ANY DROPPED "1" IN ALTERNATE 1/0'S.	SAME AS ERROR POSITION
4A	5A	$x_1x_2x_3x_4 101$	$010x_1x_2x_3x_4$	TIMING BIT 2	ADDRESS BIT 4

* FOR ALTERNATE ONE/ZERO TRAILING BIT PATTERN

TABLE 6 - SHEET 2 OF 2

ALSEP RECOGNIZING ADDRESS	INTENDED ALSEP ADDRESS	ERRONEOUS COMMAND EXECUTED (BINARY)	INTENDED COMMAND (BINARY)	POSITION OF ERROR IN THE COMMAND WORD	STARTING POSITION IN COMMAND WORD OF ADDR. RECOG.
4A	5A	$x_1 x_2 x_3 x_4\,111$	$010\,x_1 x_2 x_3 x_4$	COMMAND BIT 2	ADDRESS BIT 4
4A	5A	$x_1 x_2 x_3 x_4\,111$	$000\,x_1 x_2 x_3 x_4$	CMND CMPLT BIT 2	ADDRESS BIT 2
4B	1A OR 1B	$x_1 x_2 x_3 x_4 x_5 x_6\,1$	$0\,x_1 x_2 x_3 x_4 x_5 x_6$	ADDRESS BIT 6	ADDRESS BIT 2
4B	3B	$110\,x_1 x_2 x_3 x_4$	$x_1 x_2 x_3 x_4\,001$	PREAMBLE BIT 18	PREAMBLE BIT 18
4B	ANY	1010101	ANY	TIMING BIT 18 *	TIMING BIT 18
5A	4A	$010\,x_1 x_2 x_3 x_4$	$x_1 x_2 x_3 x_4\,101$	PREAMBLE BIT 18	PREAMBLE BIT 18
5B	——	NONE	——	ANY SINGLE BIT	——

* FOR ALTERNATE ONE/ZERO TRAILING BIT PATTERN.

TABLE 7

INTENDED ALSEP ADDRESS	ALSEP RECOGNIZING ADDRESS
1A	2B
	4A
1B	2B
	3A
	4B
2A	3A
	4A
2B	1A
	4A
3A	2A
3B	2A
	4B
4A	3A
	3B
	5A
4B	3A
5A	4A
5B	2A
	3A

SAMPLE CALCULATION— ERRONEOUS COMMAND EXECUTION

LET E = ALSEP (COMMAND DECODER NUMBER) 2A WILL ERRONEOUSLY RECOGNIZE ITS ADDRESS AND A COMMAND.

THEN, FOR EXISTING ADDRESSES ONLY

$$\text{Prob}(E) = \left[\sum_{\substack{\text{INTENDED ALSEP} \\ \text{ADDRESSES THAT,} \\ \text{WITH A BIT-ERROR} \\ \text{IN TRANSMISSION,} \\ \text{COULD RESULT IN} \\ \text{ALSEP 2A ERRONEOUSLY} \\ \text{EXECUTING A COMMAND.}}} \left(\frac{1}{\#\text{ EXISTING ADDRS.}} \times \frac{(\#\text{ OF COMMANDS THAT COULD BE EXECUTED})}{(\#\text{ OF ALLOWED COMMANDS})}\right)\right] \times \left[\begin{matrix}\text{PROBABILITY} \\ \text{OF BIT} \\ \text{ERROR IN} \\ \text{TRANSMISSION}\end{matrix}\right]$$

IF (# OF COMMANDS THAT COULD BE EXECUTED) IS REPLACED BY (# OF 7-BIT PATTERNS THAT COULD PASS THE BIT-BY-BIT PARITY CHECK), THEN Prob(E) WOULD BE LESS THAN OR EQUAL TO THE RIGHT HAND SIDE OF THE EQUATION (Almost always "less than")

$$\text{Prob}(E) \leq \left[6 \times \left(\frac{1}{8} \times \frac{16}{100}\right)\right] \times 10^{-9} = 12 \times 10^{-11} \approx \underline{\underline{10^{-10}}}$$

INCLUDING THE PAIR OF RECOMMENDED NEW ADDRESSES

$$\text{Prob}(E) \leq \sum \left(\frac{1}{\text{TOTAL \# OF ADDRS.}} \times \frac{\#\text{ OF COMMAND THAT COULD BE EXEC.}}{\#\text{ OF ALLOWED COMMANDS}}\right) \times \left(\begin{matrix}\text{PROBABILITY OF} \\ \text{BIT-ERROR IN} \\ \text{TRANSMISSION}\end{matrix}\right)$$

$$P(E) \leq \left[6 \times \left(\frac{1}{10} \times \frac{16}{100}\right) + 3 \times \left(\frac{1}{10} \times \frac{8}{100}\right)\right] \times 10^{-9}$$

$$= 12 \times 10^{-11} \approx \underline{\underline{10^{-10}}}$$

APPENDIX A

COMPUTER STORAGE

ADDRESS LIST:

		FOR EXAMPLE
FIRST ADDRESS	A(1,1) ... A(1,7)	ALSEP 1A
SECOND ADDRESS	A(2,1) ... A(2,7)	ALSEP 1B
	⋮	⋮
R th ADDRESS	A(R,1) ... A(R,7)	ALSEP $\left(\frac{R}{2}\right)$B
	MOST SIGNIFICANT BIT (MSB) ... LSB	

COMMAND STORAGE: CC

C(1), C(2), ..., C(7)
MSB ... LSB

COMMAND COMPLEMENT: $\overline{CC}$

N(1), N(2), ..., N(7)
MSB ... LSB

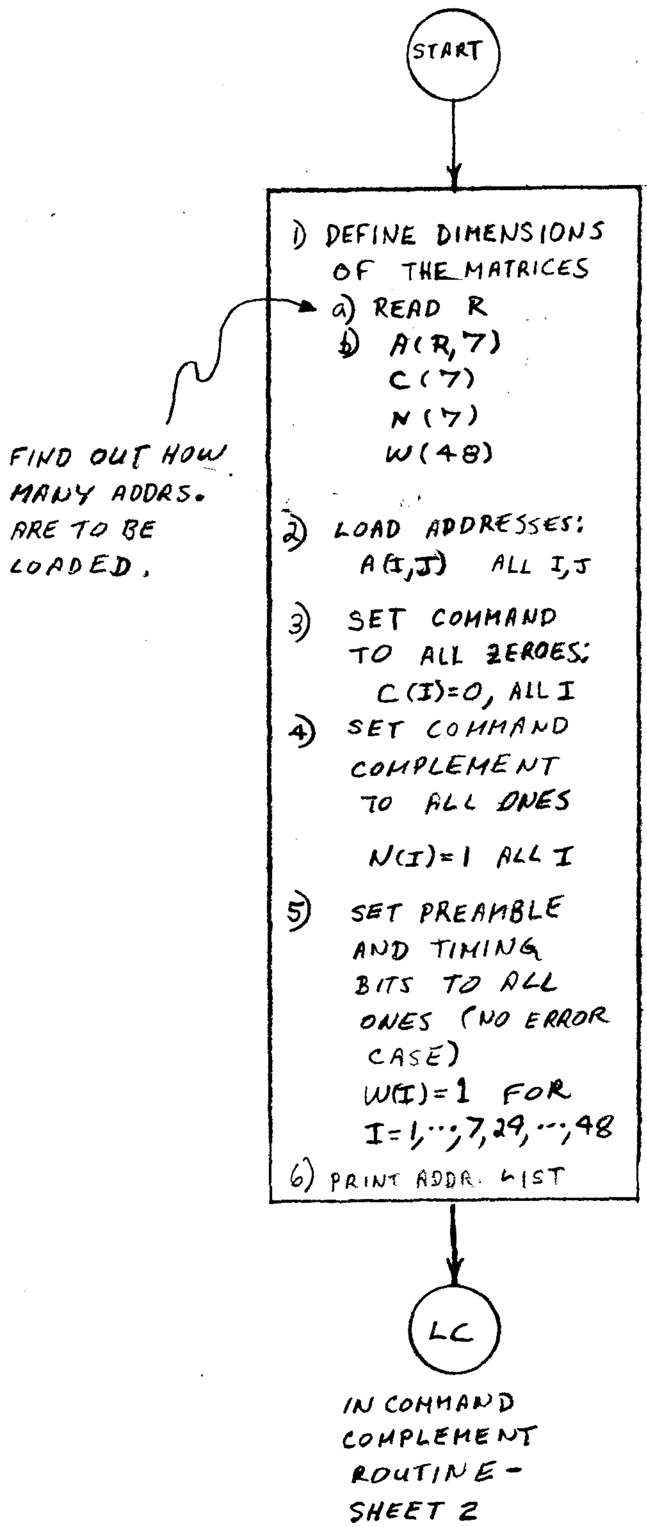
COMMAND WORD: (ABBREVIATED)

PREAMBLE		ADDRESS	$\overline{CC}$	CC	TIMING	TRAILING BITS
1, 2, ..., 13,	14, ... , 20,	21 ... 27	28, ... 34	35, ... 41,	42, ... , 61	
	W(1), ..., W(7)	W(8), ... W(14)	W(15), ..., W(21)	W(22), ... W(28)	W(29), ..., W(48)	

W(1), ..., W(48) MAKES UP THE ABBREVIATED COMMAND WORD THAT WILL BE LOOKED AT IN THE PROGRAM

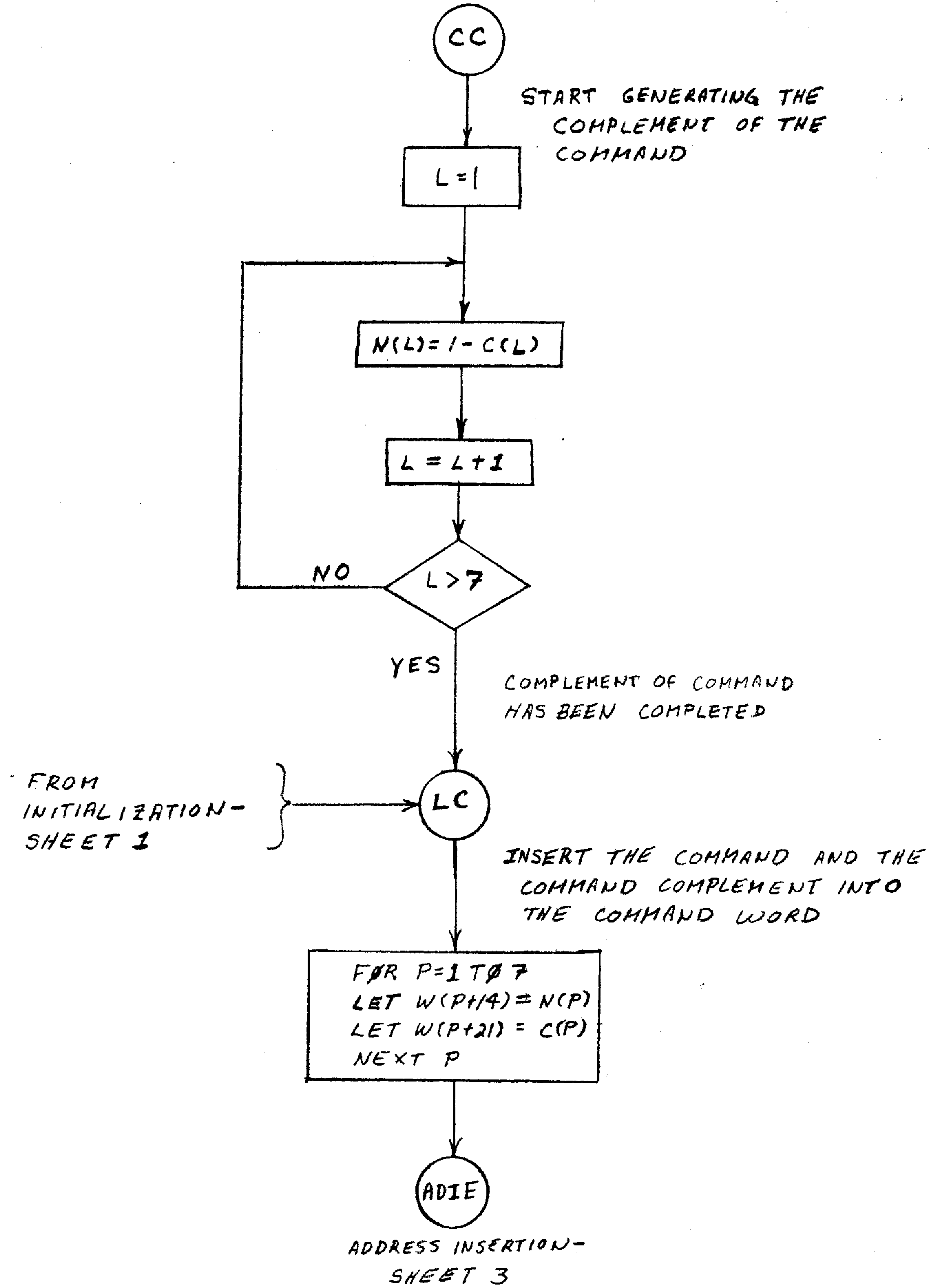
SHEET 1

INITIALIZATION



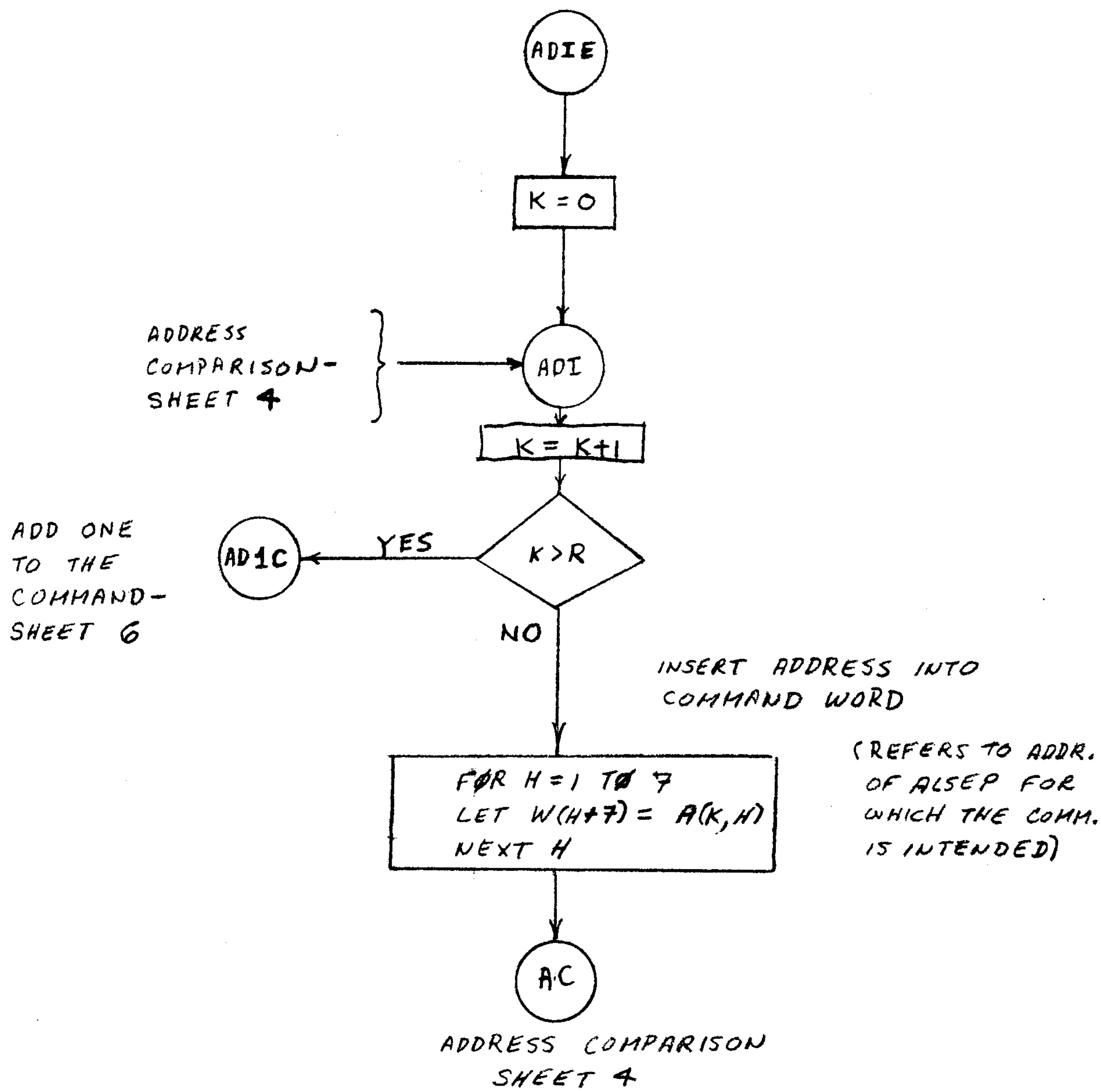
SHEET 2

COMMAND COMPLEMENT ROUTINE



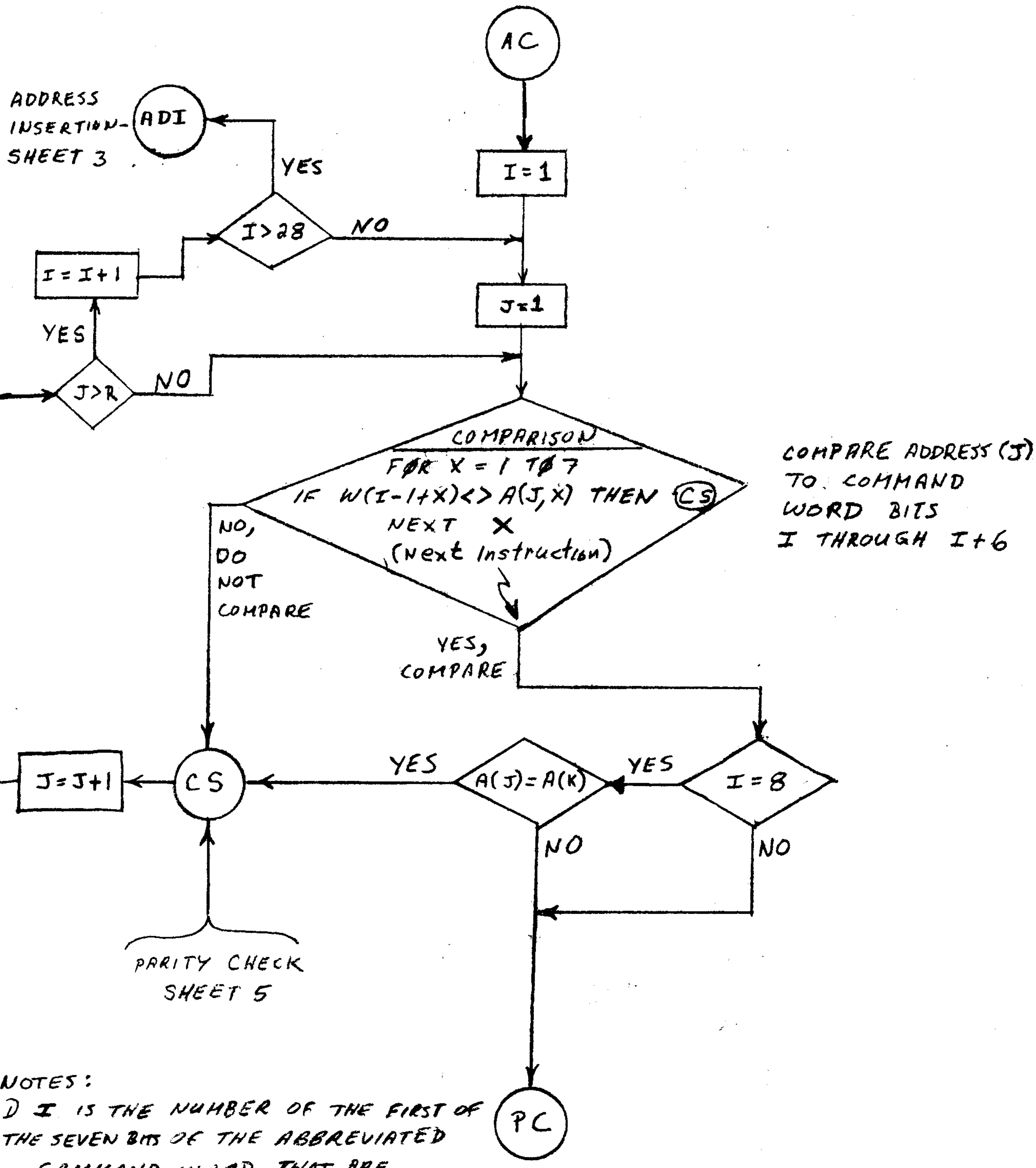
SHEET 3

ADDRESS INSERTION ROUTINE



SHEET 4

ADDRESS COMPARISON ROUTINE



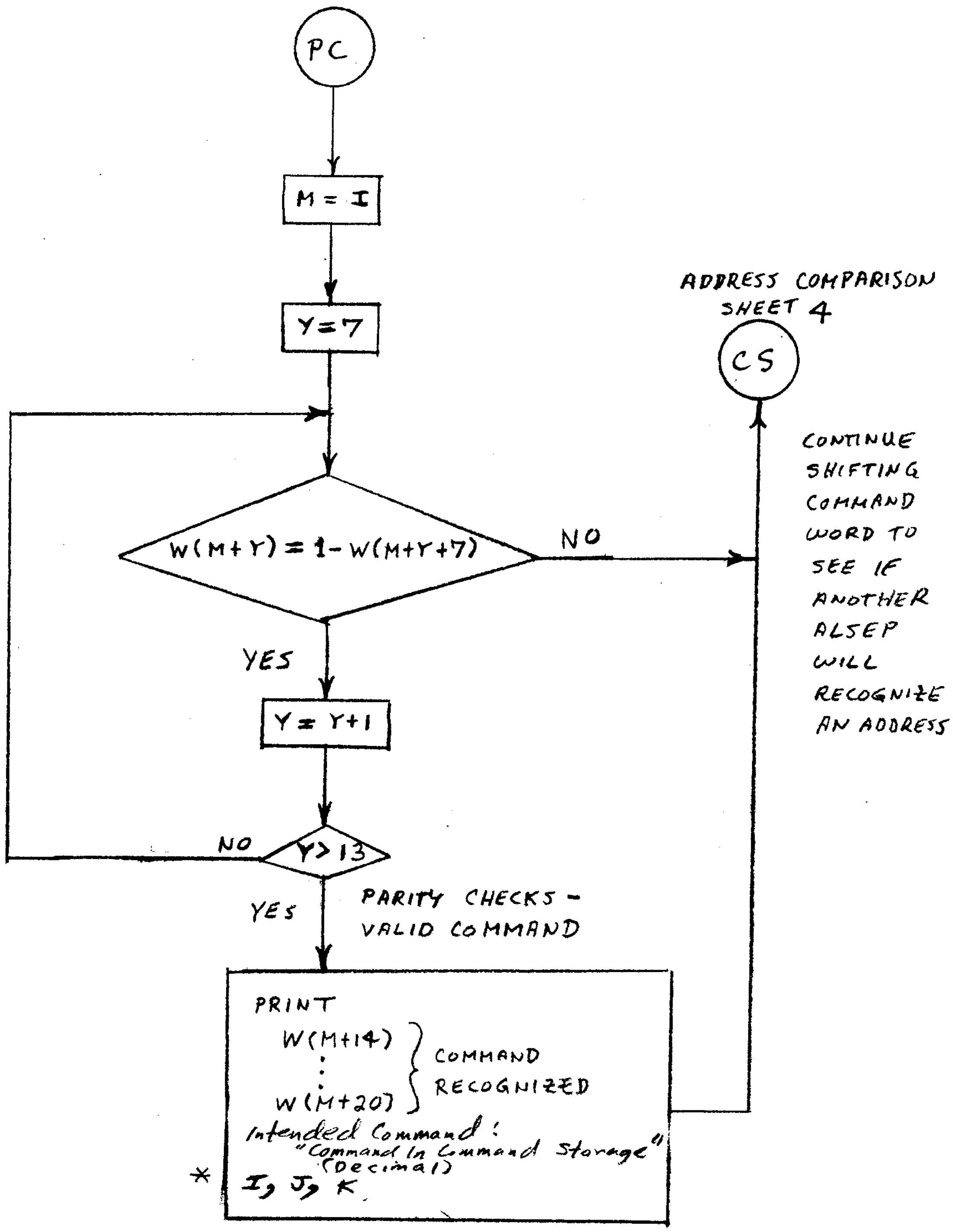
NOTES:

1) I IS THE NUMBER OF THE FIRST OF THE SEVEN BITS OF THE ABBREVIATED COMMAND WORD THAT ARE BEING COMPARED WITH THE ALSEP ADDRESSES.

2) K DESIGNATES THE ADDRESS IN THE COMMAND WORD.

SHEET 5

PARITY CHECK

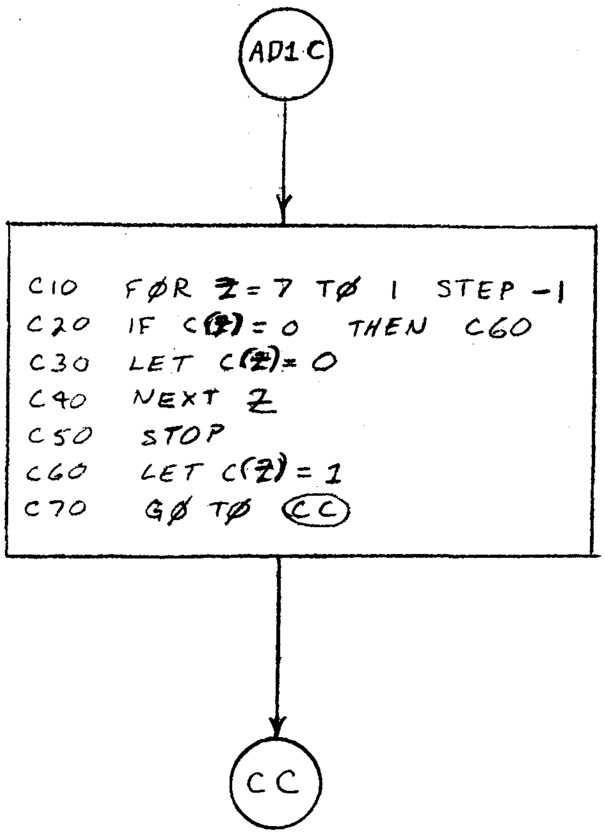


* I = starting position of recognized address
J = No. of recognized address as listed at start of output
K = No. of intended address.

SHEET 6

ADD ONE TO COMMAND ROUTINE

MIGHT USE A "GO TO" STATEMENT HERE TO EXTEND PROGRAM



COMMAND COMPLEMENT
SHEET 2

THE VECTOR USED TO STORE THE COMMAND IS TREATED AS A BINARY UP-COUNTER.

THE COUNTER COUNTS FROM $(0000000)_2$ TO $(1111111)_2$

AFTER EACH INCREMENT OF "1" THE ROUTINE TRANSFERS OUT TO CC.

WHEN AD1C IS ENTERED WITH $C(1),\cdots,C(7) = (1111111)_2$

$C(1),\cdots,C(7)$ IS SET TO $(0000000)_2$

AND THE STOP COMMAND IS EXECUTED.

ERROR-FREE-TRANSMISSION SIMULATION PROGRAM - SHEET 1

VERSION 2 (LANGUAGE: G.E.-400 SERIES BASIC)

```
9    LET B=0
10   DIM C(7), N(7), W(48), A(20,7)
20   READ R
30   MAT READ A(R,7)
40   MAT C=ZER
50   MAT N=CØN
60   MAT W=CØN
70   FØR I=1 TØ R
80   PRINT "ADDRESS ";I,
90   FØR J=1 TØ 6
100  PRINT A(I,J);
110  NEXT J
120  PRINT A(I,7)
130  NEXT I
140  GØ TØ 320
210  REM COMMAND COMPLEMENT RØUTINE
220  FØR L=1 TØ 7
230  LET N(L)=1-C(L)
240  NEXT L
320  FOR P=1 TØ 7
330  LET W(P+14)= N(P)
340  LET W(P+21) = C(P)
350  NEXT P
360  GØ TØ 410
410  REM ADDRESS INSERTIØN RØUTINE
420  LET K=0
440  LET K=K+1
450  IF K>R THEN 1310
470  FØR H=1 TØ 7
480  LET W(H+7) = A(K,H)
490  NEXT H
500  GØ TØ 620
610  REM ADDRESS CØMPARISØN RØUTINE
620  LET I=1
640  LET J=1
660  FØR X=1 TØ 7
670  IF W(I-1+X) <> A(J,X) THEN 920
680  NEXT X
690  IF I=8 THEN 780
695  GØ TØ 1110
```

ERROR-FREE-TRANSMISSION SIMULATION PROGRAM - SHEET 2
VERSION 2

```
780   FØR X=1 TØ 7
790   IF A(J,X) <> A(K,X) THEN 1110
800   NEXT X
920   LET J=J+1
940   IF J>R THEN 970
950   GØ TØ 660
970   LET I=I+1
990   IF I>28 THEN 440
1000  GØ TØ 640

1110  REM PARITY CHECK RØUTINE
1120  LET M=I
1130  LET Y=7
1140  IF W(M+Y)=1-W(M+Y+7) THEN 920
1150  LET Y=Y+1
1160  IF Y>13 THEN 1180
1170  GØ TØ 1140
1180  PRINT "CØMMAND RECØGNIZED ";
1190  FØR X=M+14 TØ M+19
1200  PRINT W(X)
1210  NEXT X
1215  PRINT W(M+20)
1220  PRINT "CØMMAND IN CØMMAND STØRAGE ";
1230  PRINT B
1240  PRINT "I= "; I, "J= "; J, "K= "; K
1245  PRINT "CICW";
1250  FOR G=22 TØ 27
1252  PRINT W(G)
1255  NEXT G
1257  PRINT W(28)
1260  GØ TØ 920

1308  REM ADD ØNE TØ CØMMAND RØUTINE
1310  FØR Z=7 TØ 1 STEP -1
1320  IF C(Z)=0 THEN 1360
1330  LET C(Z)=0
1340  NEXT Z
1350  STOP
1360  LET C(Z)=1
1365  LET B=B+1
1366  PRINT B
1370  GØ TØ 210
```

ERROR-FREE-TRANSMISSION SIMULATION PROGRAM - SHEET 3

VERSION 2

```
1400  DATA  (NO. OF ADDRESSES, R)
1410  DATA  1,0,1,1,0,0,0
1420  DATA  0,0,1,1,0,0,0
1430  DATA  1,0,0,1,1,1,0
1440  DATA  0,0,0,1,1,1,0
1450  DATA  1,1,0,1,0,0,1
1460  DATA  0,1,0,1,0,0,1
1470  DATA  0,0,1,0,1,0,1
1480  DATA  0,1,1,0,1,0,1
1490  DATA  (FIRST ADDRESS TO BE CHECKED)
1500  DATA  (SECOND   "     "  "    "   )
  :          :
  :         (Nth      "     "  "    "   )
1600  END
```

APPENDIX B

PROGRAM EXECUTION PARAMETERS FOR SIMULATION INCLUDING TRANSMISSION ERROR

1) $9 \leq ERP\O S \leq 28$

2) FOR

$9 \leq ERP\O S \leq 21$, $9 \leq I \leq ERP\O S$

3) FOR

$22 \leq ERP\O S \leq 28$, $9 \leq I \leq 21$

COMMENTS: (ITEM BY ITEM)

1) $1 \leq ERP\O S \leq 8$, $29 \leq ERP\O S \leq 48$: SIMULATION DONE MANUALLY.

2) $I > ERP\O S$: SAME AS NO-TRANSMISSION-ERROR SIMULATION.

3) $I > 21$; ANY SINGLE (OR MULTIPLE) ERROR IN THE COMMAND (BITS 22 TO 28) WILL CONVERT IT TO ANOTHER COMMAND (SINCE ALL 128 POSSIBLE NOS. WERE CONSIDERED) THAT WAS CHECKED IN THE NO-TRANSMISSION-ERROR SIMULATION.

COMPUTER STORAGE FOR ERROR-IN-TRANSMISSION SIMULATION

ADDRESS LIST :

ADDRESS 1	A(1,1) ··· A(1,7)	ALSEP 1A
ADDRESS 2	A(2,1) ··· A(2,7)	ALSEP 1B
⋮	⋮ ⋮	
ADDRESS 8	A(8,1) ··· A(8,7)	ALSEP 4A
" 9	A(9,1) ··· A(9,7)	POSSIBLE ADDRESS # 3
" 10	A(10,1) ··· A(10,7)	" " # 7
" 11	A(11,1) ··· A(11,7)	" " # 1

COMMAND STORAGE:

C(1) ··· C(7)

COMMAND WORD : (ABBREVIATED)

SAME AS FOR ERROR-FREE-TRANSMISSION SIMULATION

PROGRAM MODIFICATIONS: SHEET 1

INITIALIZATION

START

1) DIMENSIONS: A(12,7), C(7), W(42)
2) READ R
3) READ ADDRESS LIST
   A(1,1), ..., A(1,7)
   ⋮
   A(R,1), ..., A(R,7)
4) PRINT ADDRESS LIST
5) SET ERROR POSITION TO 9:
   ERPØS = 9

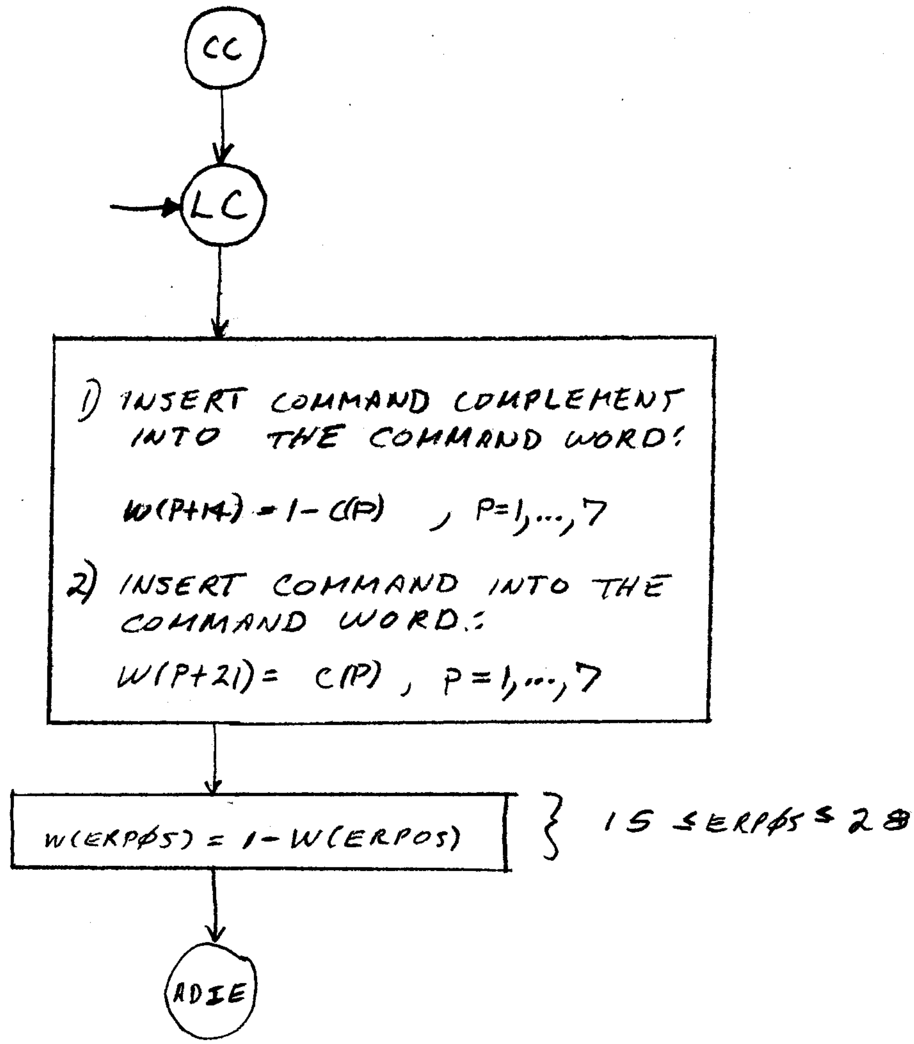
   ERPØS = 15 For 2nd Half of simulation
6) PRINT ERROR POSITION:

   TRANSFER TO ERPØS WRITE AND BACK TO INITIALIZATION (7).
7) SET COMMAND TO ALL ZEROS:
   C(I) = 0   I = 1, ..., 7
8) SET COMMAND WORD TO ALL ONES:
   W(I) = 1   I = 8, ..., 42
9) SET (DECIMAL) COMMAND TO ZERO:
   DC = 0

LC

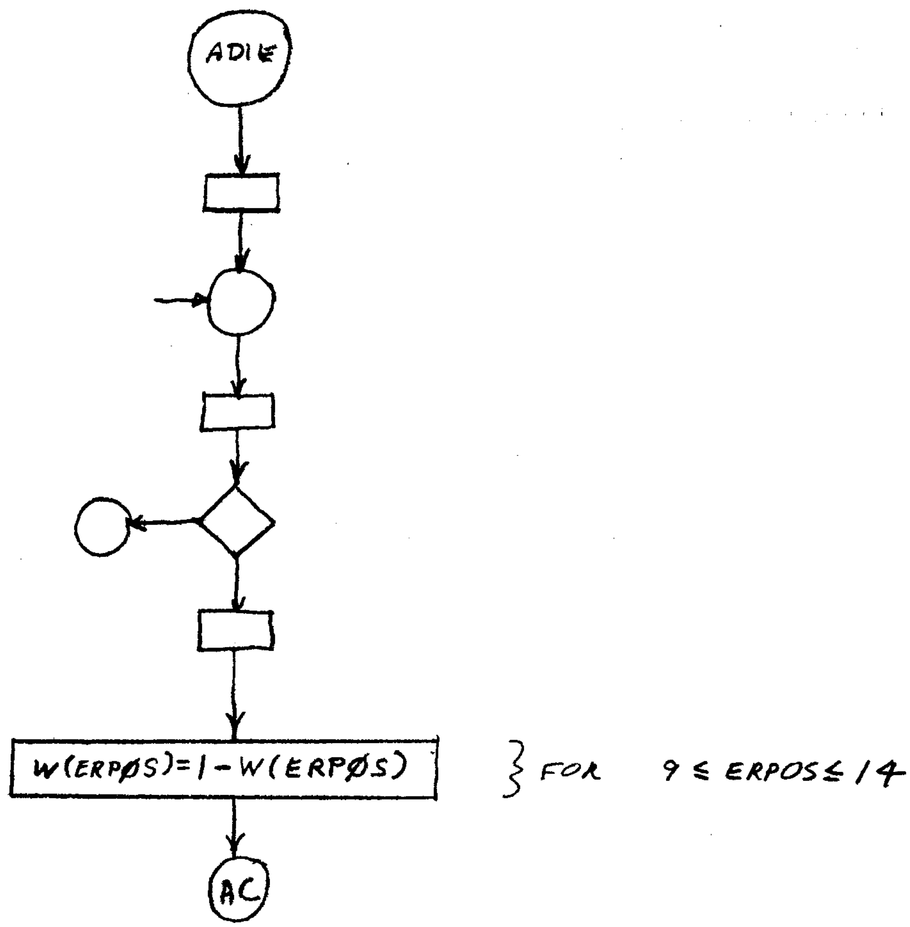
PROGRAM MODIFICATIONS: SHEET 2

COMMAND COMPLEMENT (COMMAND INSERTION)



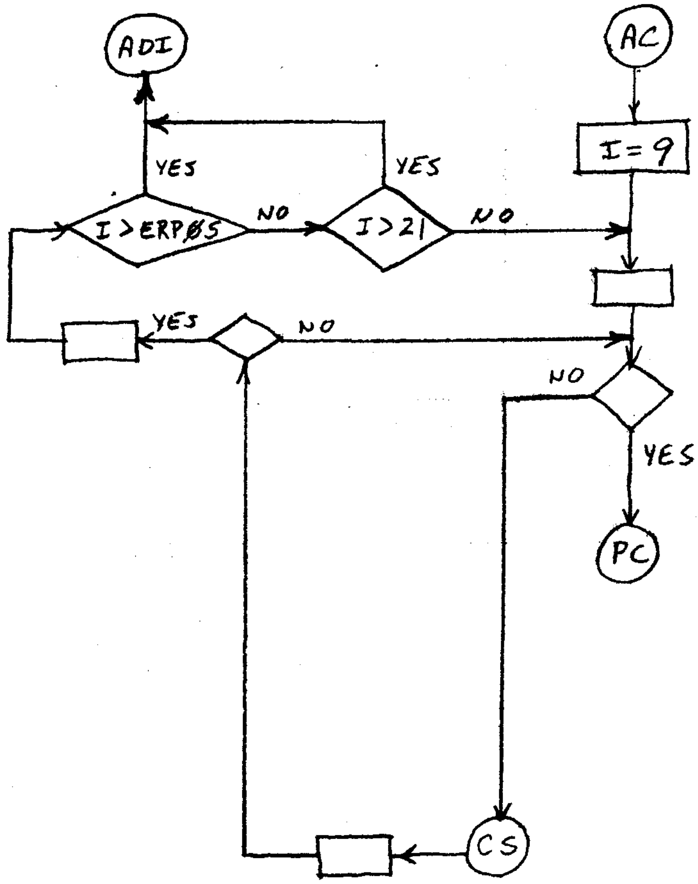
PROGRAM MODIFICATIONS : SHEET 3

ADDRESS INSERTION

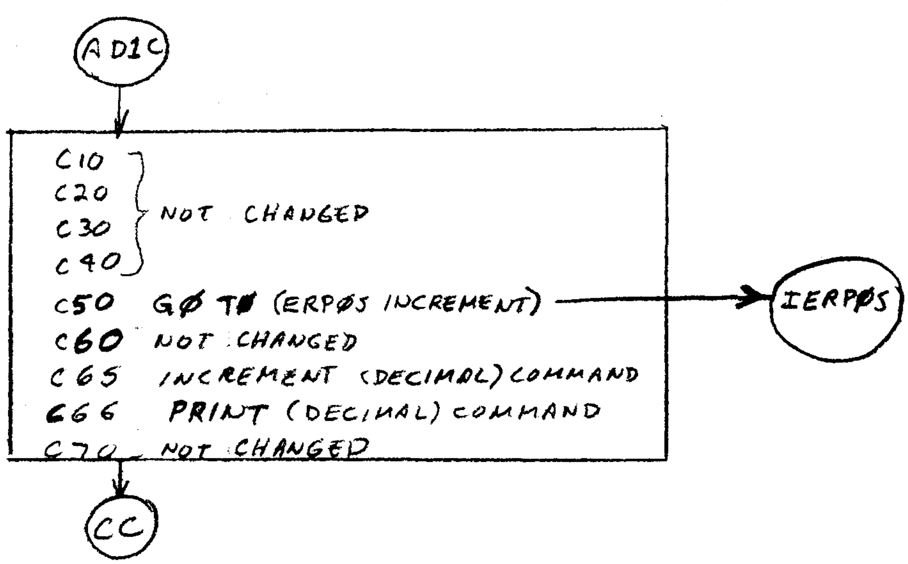


PROGRAM MODIFICATIONS: SHEET 4

ADDRESS COMPARISON

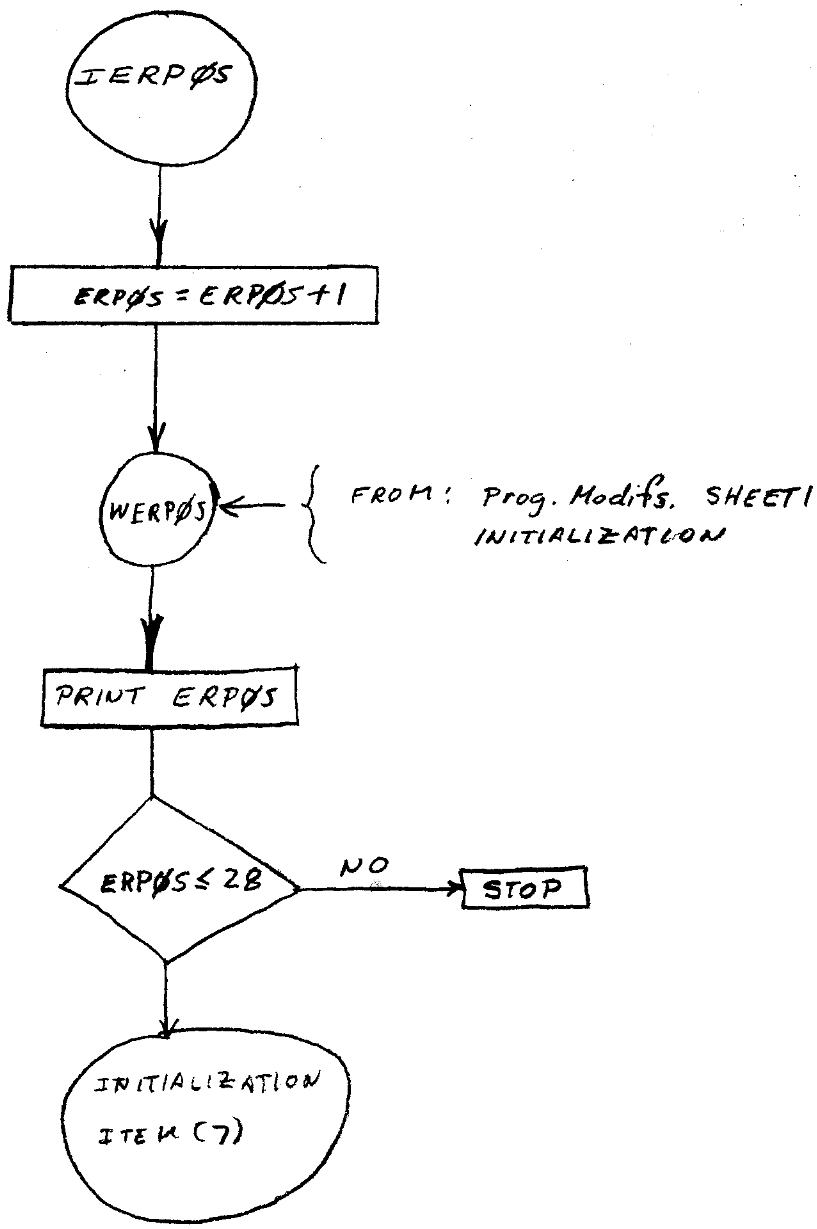


ADD ONE TO COMMAND



PROGRAM MODIFICATIONS: SHEET 5

ERROR POSITION INCREMENT



```
          Y=Y+1
          IF (Y.GT.13)  GO TO 213
          GO TO 212
      213 IPC14=IPC+14
          IPC20=IPC+20
          WRITE (6,502) (W(L),L=IPC14,IPC20)
      502 FORMAT (19HOCOMMAND RECOGNIZED,4X,7I2)
          WRITE (6,503) DC
      503 FORMAT (17H INTENDED COMMAND,6X,I3)
          WRITE (6,504)  I,J,K,ERPOS
      504 FORMAT (3H I=,  I3,5X,'J=',  I3,5X,'K=',  I3,5X,'ERPOS=',  I3)
          GO TO 206
C   ADD ONE TO COMMAND
      214 DO 108  Z=1,7
          IZ=8-Z
          IF (C(IZ).EQ.0)  GO TO 216
          C(IZ)=0
      108 CONTINUE
          GO TO 215
      216 C(IZ)=1
          DC=DC+1
          WRITE(6,505)  DC
      505 FORMAT (1H ,I4)
          GO TO 202
      215 ERPOS=ERPOS+1
      200 WRITE (6,506) ERPOS
      506 FORMAT (7HOERPOS=,2X,I2/)
          IF (ERPOS.LE.26)  GO TO 201
          STOP
          END
```

ERPØS = 21 TO 26


```
      DIMENSION  C(7),W(42),A(12,7)
      INTEGER  A,C,DC,ERPOS,P,R,W,X,Y,Z,H
      READ (5,600) R
  600 FORMAT (I10)
      DO 101 I=1,R
      READ(5,601)  (A(I,J),J=1,7)
  601 FORMAT (7I1)
  101 CONTINUE
      DO 100  I=1,R
      WRITE (6,500)  I,(A(I,J),J=1,7)
  500 FORMAT (8H ADDRESS,2X,I2,6X,7I2)
  100 CONTINUE
      ERPOS=21
      GO TO  200
  201 DO 102  I=1,7
      C(I)=0
  102 CONTINUE
      DO 103  I=8,42
      W(I)=1
  103 CONTINUE
      DC=0
      GO TO  202
C  COMMAND AND COMMAND COMPLEMENT INSERTION
  202 DO 104  P=1,7
      W(P+14)=1-C(P)
      W(P+21)=C(P)
  104 CONTINUE
      W(ERPOS)=1-W(ERPOS)
      GO TO  203
C  ADDRESS INSERTION
  203 K=0
  209 K=K+1
      IF (K.GT.R) GO TO 214
      DO 105  H=1,7
      W(H+7)=A(K,H)
  105 CONTINUE
      GO TO  204
C  ADDRESS COMPARISON
  204 I=9
  210 J=1
  207 DO 106  X=1,7
      IX=I-1+X
      IF (W(IX).NE.A(J,X)) GO TO 206
  106 CONTINUE
      GO TO 211
  206 J=J+1
      IF (J.GT.R)  GO TO 208
      GO TO  207
  208 I=I+1
      IF (I.GT.ERPOS)  GO TO 209
      IF (I.GT.15)     GO TO 209
      GO TO 210
C  PARITY CHECK
  211 IPC=I
      Y=7
  212 IY=I+Y
      IY7=IY+7
      IF (W(IY).NE.(1-W(IY7)))  GO TO 206
```