IBM® **Field Engineering**

Maintenance Diagrams

2030 Processing Unit

System/360 Model 30

PREFACE

This manual contains Condensed Logic Flow charts to be used for recall or instructional purposes.

The EC Level of the CAS Logic Diagrams (CLD) for the basic machine is128062. The Level for the 1400 Compatibility section is 128122.

The charts in this manual were drawn to show the general logic and flow of the microprogram used by the 2030. The charts contain the CAS logic diagram page numbers where the exact process can be located and followed.

Fifth Edition (June 1967)

This edition, Y24-3466-2, is a major revision of and obsoletes the previous edition, Y24-3466-1, and the Supplement Y24-3490. Principal changes include the addition of the diagnostic techniques charts and information pertaining to ROAR stop.

This manual has been prepared by the IBM Systems Development Division, Product Publications, Dept. 171, P. O. Box 6, Endicott, N. Y. 13760. Send comments concerning the manual to this address.

This manual has been prepared by the IBM Systems Development Division,
Product Publications, Dept. 171, PO 6, Endicott, New York 13760
Address comments concerning the manual to this address.

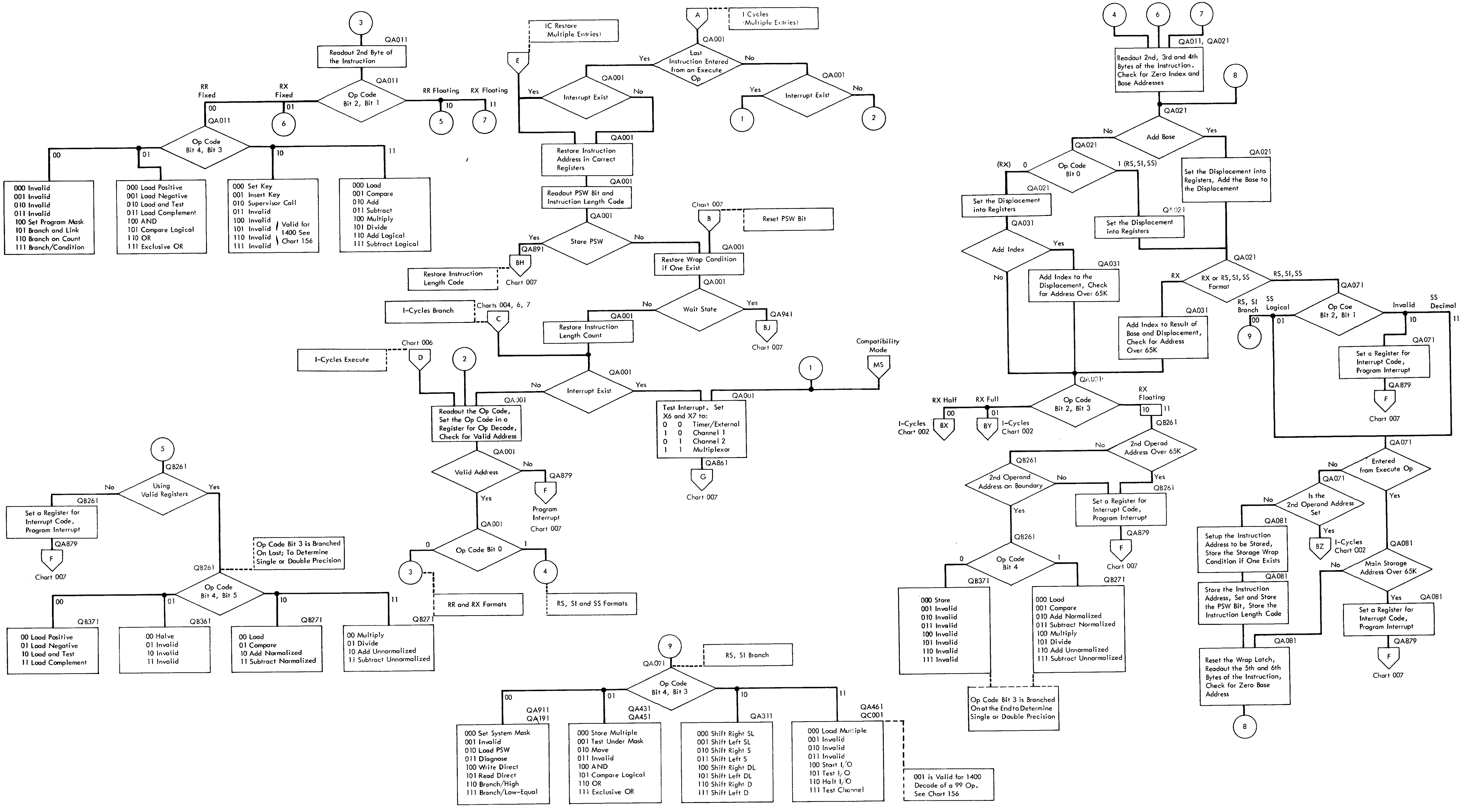
Contents

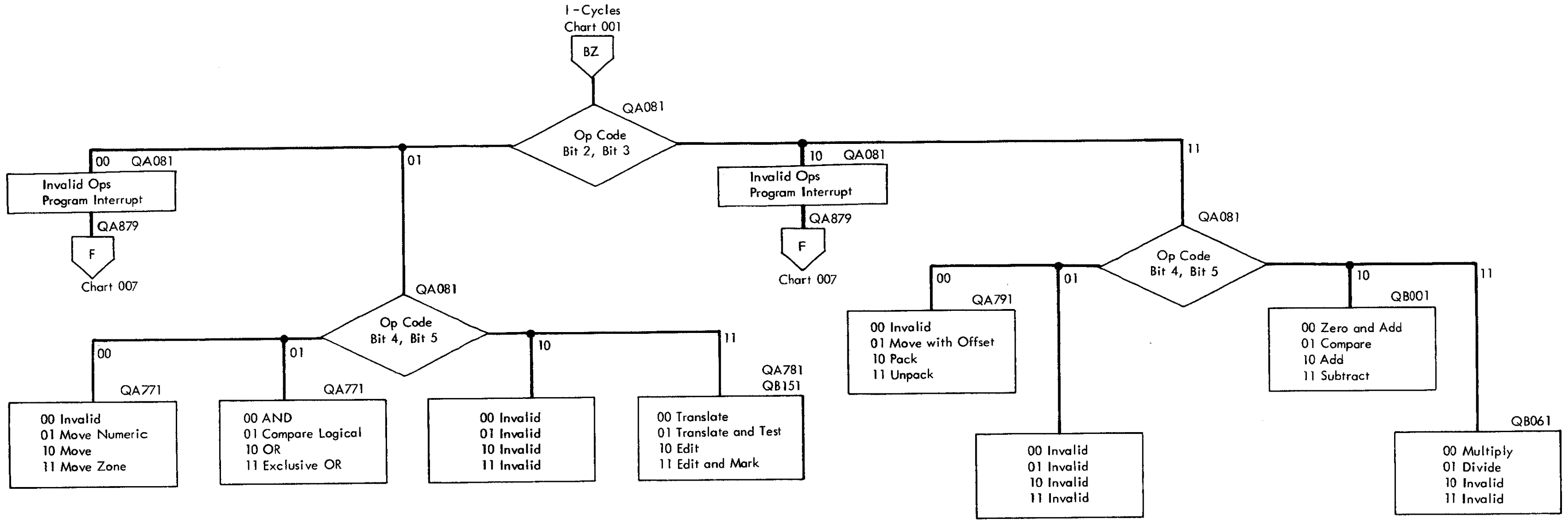
CONDENSED LOGIC FLOW CHARTS

DIAGNOSTIC TECHNIQUES CHARTS

ROAR STOP CHARTS

CLF 001 I-Cycles, Sheet 1





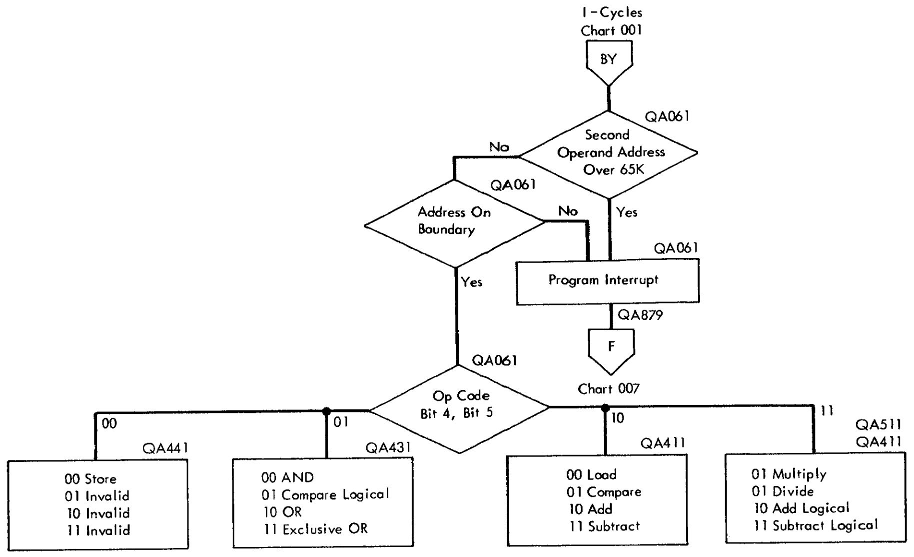
OP	Name	Chart
04	Set Program Mask	007
05	Branch and Link	006
06	Branch on Count	004
07	Branch on Condition	004
08	Set Key	003
09	Insert Key	003
0A	Supervisor Call	007
10	Load Positive	003
11	Load Negative	003
12	Load and Test	003
13	Load Complement	027
14	AND	004
15	Compare Logical	004
16	OR	004
17	Exclusive OR	004
18	Load	003
19	Compare	005
1A	Add	005
1B	Subtract	005
1C	Multiply	008
1D	Divide	009
1E	Add Logical	005
1F	Subtract Logical	005
20	Load Positive	003
21	Load Negative	003
22	Load and Test	003
23	Load Complement	027
24	Halve	026
28	Load	027
29	Compare	028
2A	Add N	028
2B	Subtract N	028
2C	Multiply	030
2D	Divide	032
2E	Add U	028
2F	Subtract U	028
30	Load Positive	003
31	Load Negative	003
32	Load and Test	003
33	Load Complement	027
34	Halve	026
38	Load	027
39	Compare	028
3A	Add N	028
3B	Subtract N	028

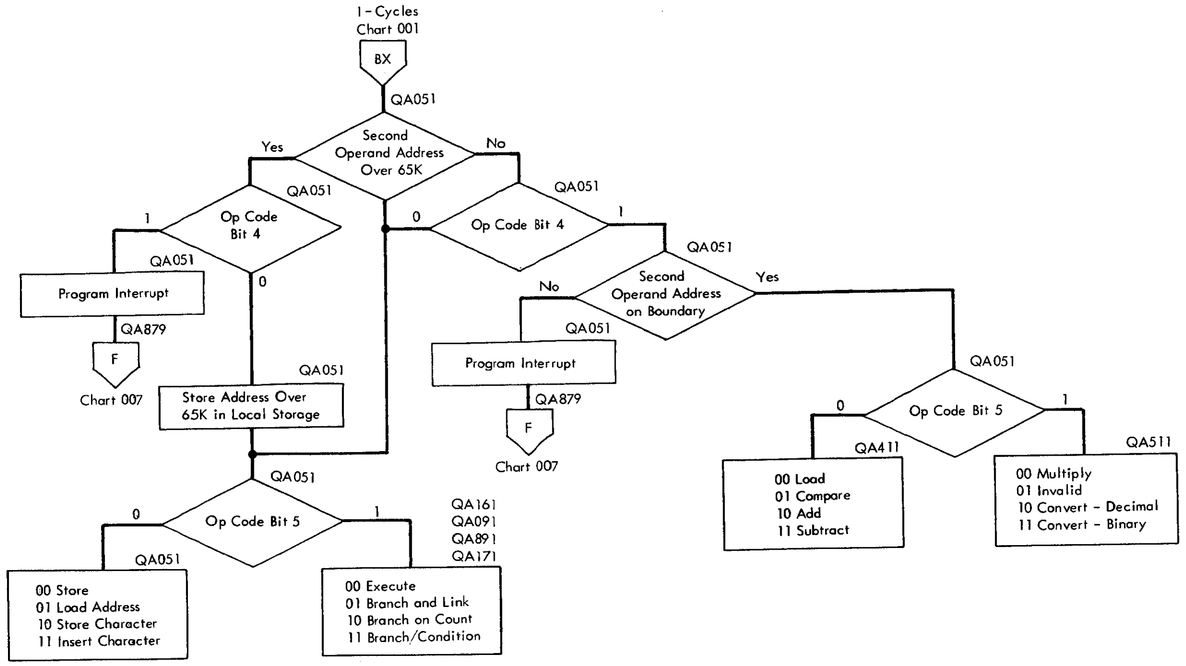
Op	Name	Chart
3C	Multiply	030
3D	Divide	032
3E	Add U	028
3F	Subtract U	028
40	Store	003
41	Load Address	003
42	Store Character	003
43	Insert Character	003
44	Execute	006
45	Branch and Link	006
46	Branch on Count	004
47	Branch on Condition	004
48	Load	003
49	Compare	005
4A	Add	005
4B	Subtract	005
4C	Multiply	008
4E	Convert - Decimal	012
4F	Convert - Binary	011
50	Store	003
54	AND	004
55	Compare Logical	004
56	OR	004
57	Execlusive OR	004
58	Load	003
59	Compare	005
5A	Add	005
5B	Subtract	005
5C	Multiply	008
5D	Divide	009
5E	Add Logical	005
5F	Subtract Logical	005
60	Store	026
68	Load	027
69	Compare	028
6A	Add N	028
6B	Subtract N	028
6C	Multiply	030
6D	Divide	032
6E	Add U	028
6F	Subtract U	028
70	Store	026
78	Load	027
79	Compare	028
7A	Add N	028

Op	Name	Chart
7B	Subtract N	028
7C	Multiply	030
7D	Divide	032
7E	Add U	028
7F	Subtract U	028
80	Set System Mask	007
82	Load PSW	007
83	No Chart	---
84	No Chart	---
85	No Chart	---
86	Branch/High	006
87	Branch/Low Equal	006
88	Shift Right SL	013
89	Shift Left SL	013
8A	Shift Right S	013
8B	Shift Left S	013
8C	Shift Right DL	013
8D	Shift Left DL	013
8E	Shift Right D	013
8F	Shift Left D	013
90	Store Multiple	014
91	Test Under Mask	014
92	Move	014
93	Test and Set	014
94	AND	004
95	Compare Logical	004
96	OR	004
97	Exclusive OR	004
98	Load Multiple	014
9C	Start I/O	034
9D	Test I/O	034
9E	Halt I/O	034
9F	Test Channel	034
D1	Move Numeric	015
D2	Move	015
D3	Move Zone	015
D4	AND	015
D5	Compare Logical	015
D6	OR	015
D7	Exclusive OR	015
DC	Translate	016
DD	Translate and Test	016
DE	Edit	017
DF	Edit and Mark	017
F1	Move with Offset	019

Op	Name	Chart
F2	Pack	019
F3	Unpack	019
F8	Zero and Add	020
F9	Compare	020
FA	Add	020
FB	Subtract	020
FC	Multiply	022
FD	Divide	024

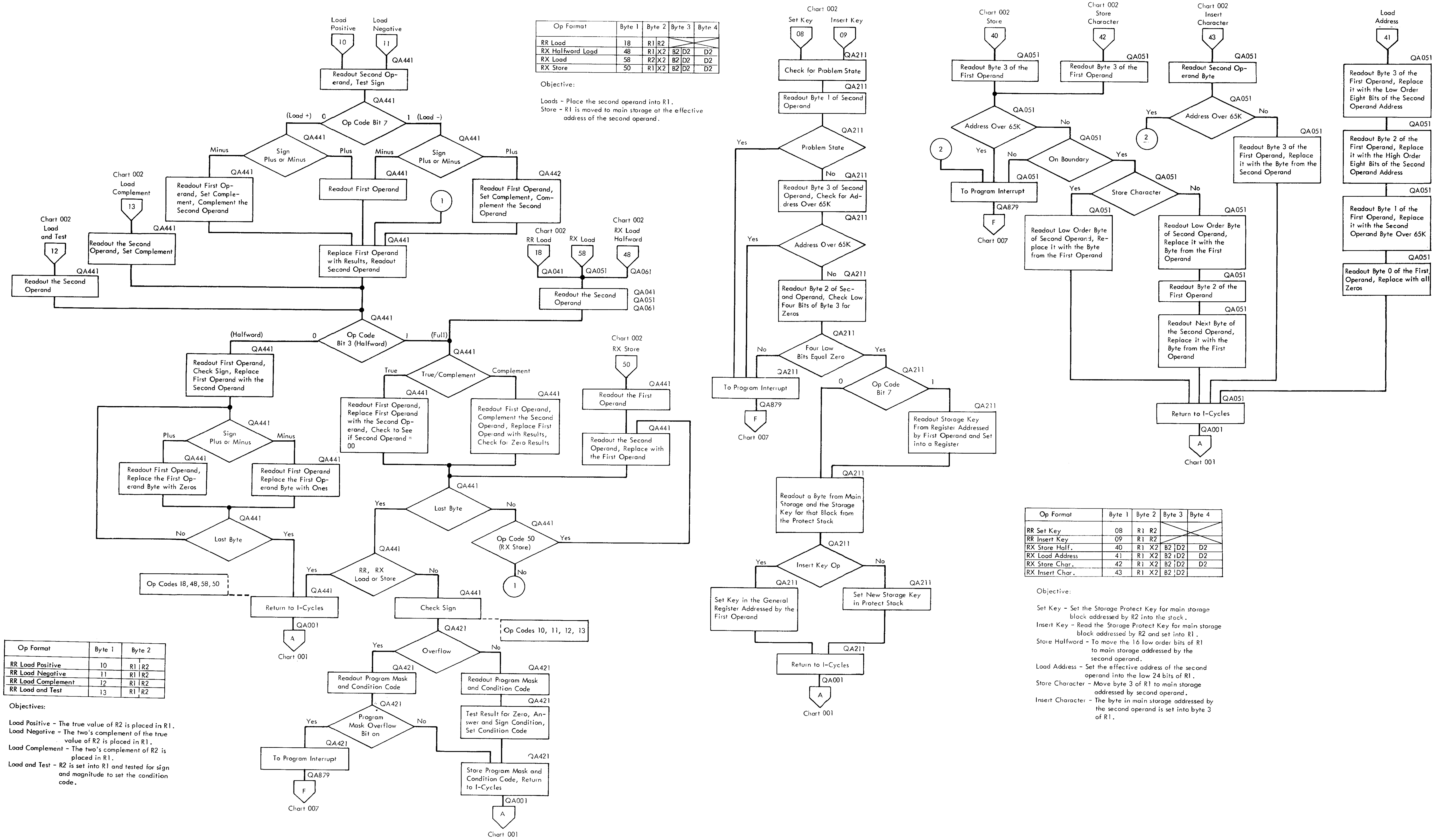
Notes: Codes Not Shown Are
Invalid and Cause a
Program Interrupt.





CLF 002 I-Cycles, Sheet 2

CLF 003 RR-RX Fixed Point, Multiple Codes 1



Op Format	Byte 1	Byte 2		Byte 3		Byte 4
RR Load	18	R1	R2			
RX Halfword Load	48	R1	X2	B2	D2	D2
RX Load	58	R2	X2	B2	D2	D2
RX Store	50	R1	X2	B2	D2	D2

Objective:

Loads - Place the second operand into R1.
Store - R1 is moved to main storage at the effective address of the second operand.

Op Format	Byte 1	Byte 2	
RR Load Positive	10	R1	R2
RR Load Negative	11	R1	R2
RR Load Complement	12	R1	R2
RR Load and Test	13	R1	R2

Objectives:

Load Positive - The true value of R2 is placed in R1.
Load Negative - The two's complement of the true value of R2 is placed in R1.
Load Complement - The two's complement of R2 is placed in R1.
Load and Test - R2 is set into R1 and tested for sign and magnitude to set the condition code.

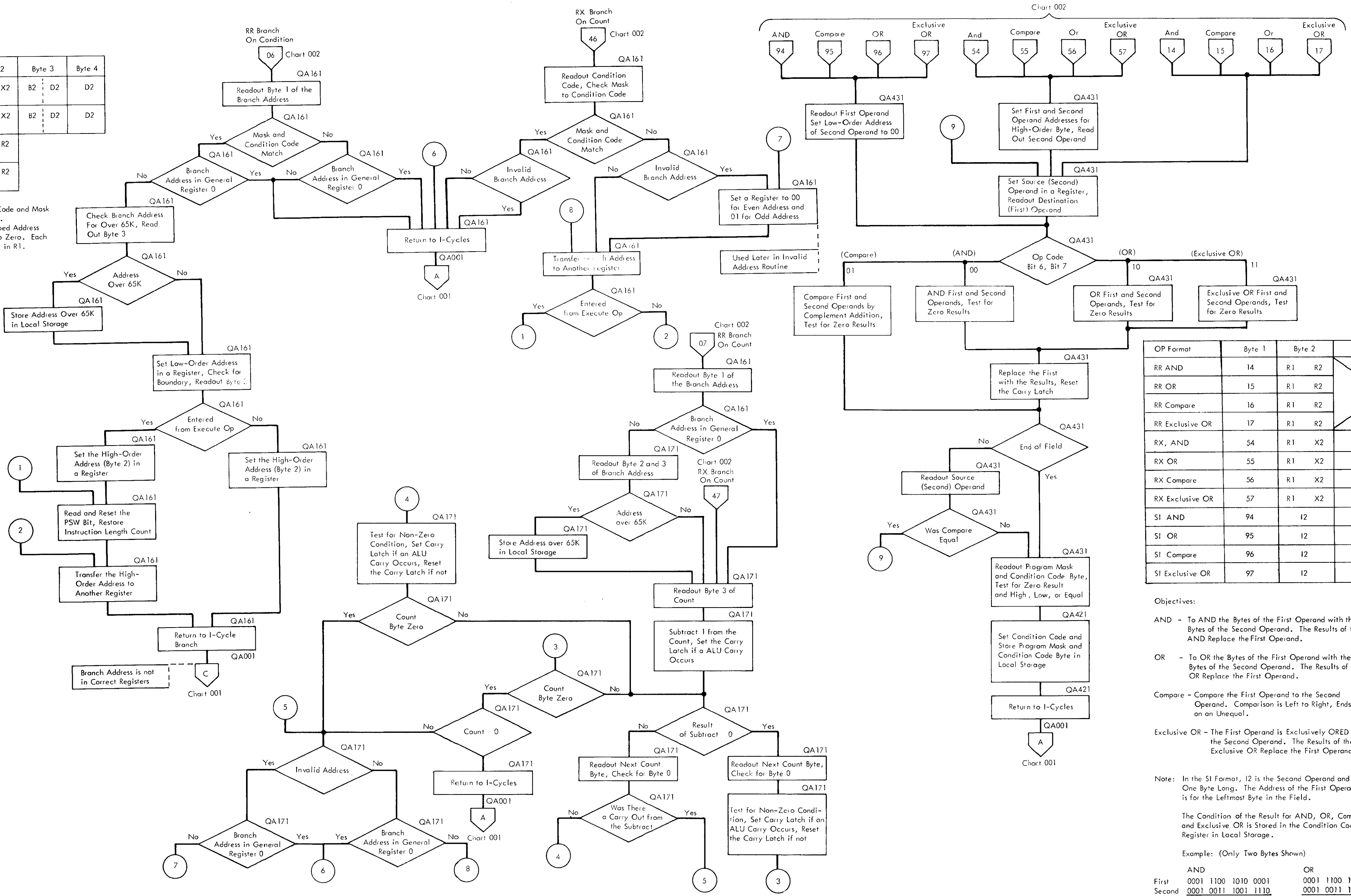
Op Format	Byte 1	Byte 2		Byte 3		Byte 4
RR Set Key	08	R1	R2			
RR Insert Key	09	R1	R2			
RX Store Half.	40	R1	X2	B2	D2	D2
RX Load Address	41	R1	X2	B2	D2	D2
RX Store Char.	42	R1	X2	B2	D2	D2
RX Insert Char.	43	R1	X2	B2	D2	

Objective:

Set Key - Set the Storage Protect Key for main storage block addressed by R2 into the stack.
Insert Key - Read the Storage Protect Key for main storage block addressed by R2 and set into R1.
Store Halfword - To move the 16 low order bits of R1 to main storage addressed by the second operand.
Load Address - Set the effective address of the second operand into the low 24 bits of R1.
Store Character - Move byte 3 of R1 to main storage addressed by second operand.
Insert Character - The byte in main storage addressed by the second operand is set into byte 3 of R1.

OP Format	Byte 1	Byte 2	Byte 3	Byte 4
RX Branch On Condition	46	M1 X2	B2 ¦ D2	D2
RX Branch On Count	47	R1 X2	B2 ¦ D2	D2
RR Branch On Condition	06	M1 R2		
RR Branch On Count	07	R1 R2		

Objective:
Branch On Condition: If Condition Code and Mask Match, Branch to Developed Address.
Branch On Count: Branch to Developed Address Each Time Until Sum In R1 Is Equal to Zero. Each Time Through Subtract 1 from Amount in R1.



OP Format	Byte 1	Byte 2	Byte 3	Byte 4
RR AND	14	R1 R2		
RR OR	15	R1 R2		
RR Compare	16	R1 R2		
RR Exclusive OR	17	R1 R2		
RX, AND	54	R1 X2	B2 ¦ —	D2 —
RX OR	55	R1 X2	B2 ¦ —	D2 —
RX Compare	56	R1 X2	B2 ¦ —	D2 —
RX Exclusive OR	57	R1 X2	B2 ¦ —	D2 —
SI AND	94	I2	B1 ¦ —	D1 —
SI OR	95	I2	B1 ¦ —	D1 —
SI Compare	96	I2	B1 ¦ —	D1 —
SI Exclusive OR	97	I2	B1 ¦ —	D1 —

Objectives:

AND – To AND the Bytes of the First Operand with the Bytes of the Second Operand. The Results of the AND Replace the First Operand.

OR – To OR the Bytes of the First Operand with the Bytes of the Second Operand. The Results of the OR Replace the First Operand.

Compare – Compare the First Operand to the Second Operand. Comparison is Left to Right, Ends on an Unequal.

Exclusive OR – The First Operand is Exclusively ORED with the Second Operand. The Results of the Exclusive OR Replace the First Operand.

Note: In the SI Format, I2 is the Second Operand and is One Byte Long. The Address of the First Operand is for the Leftmost Byte in the Field.

The Condition of the Result for AND, OR, Compare, and Exclusive OR is Stored in the Condition Code Register in Local Storage.

Example: (Only Two Bytes Shown)

	AND	OR
First	0001 1100 1010 0001	0001 1100 1010 0001
Second	0001 0011 1001 1110	0001 0011 1001 1110
Result	0001 0000 1000 0000	0001 1111 1011 1111

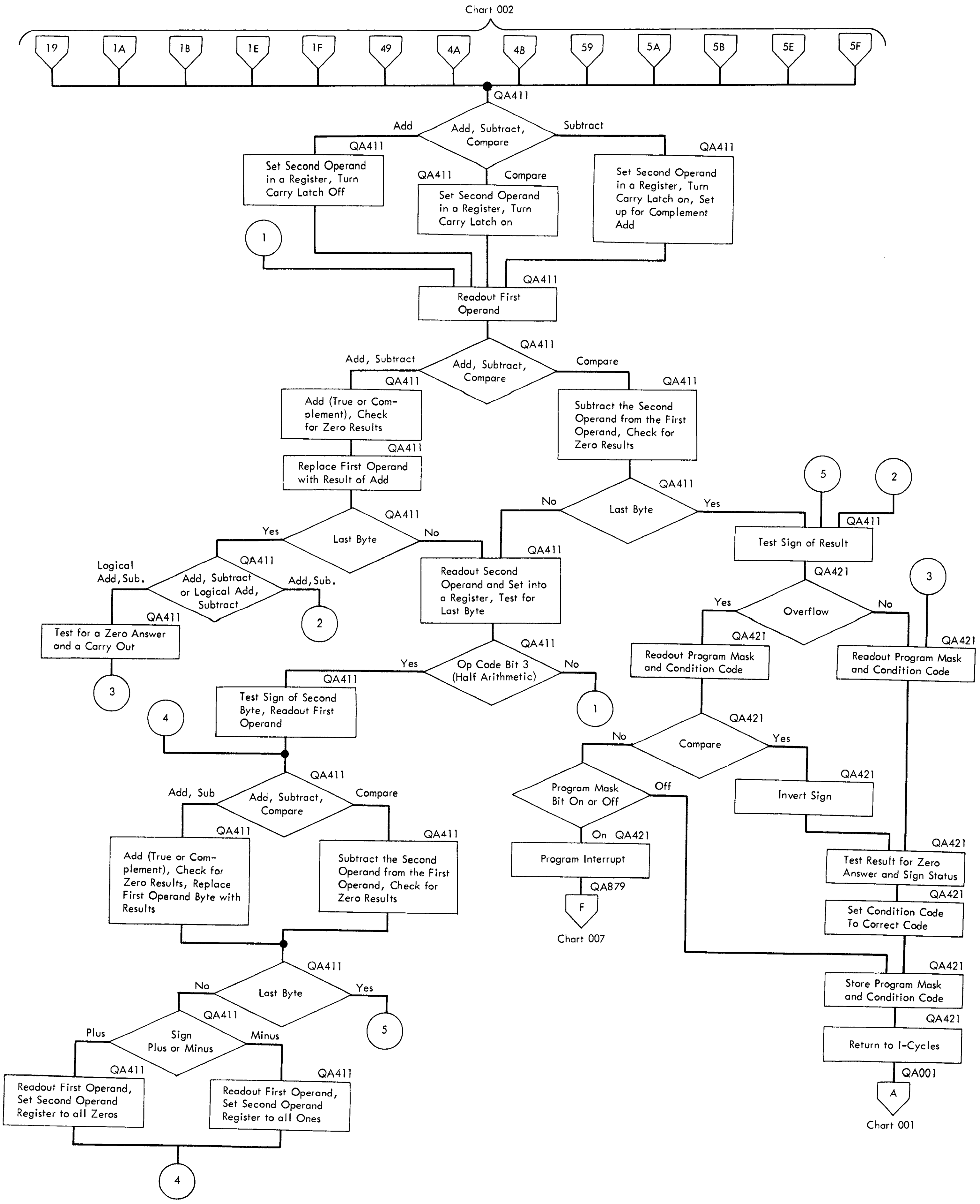
	Compare	Exclusive OR
First	0001 1100 1010 0001	0001 1100 1010 0001
Second	0001 0011 1001 1110	0001 0011 1001 1110
		0000 1111 0011 1111

First	0001 1100 1010 0001
Complement Second	1110 1100
Unequal	0000 1000
	Ends

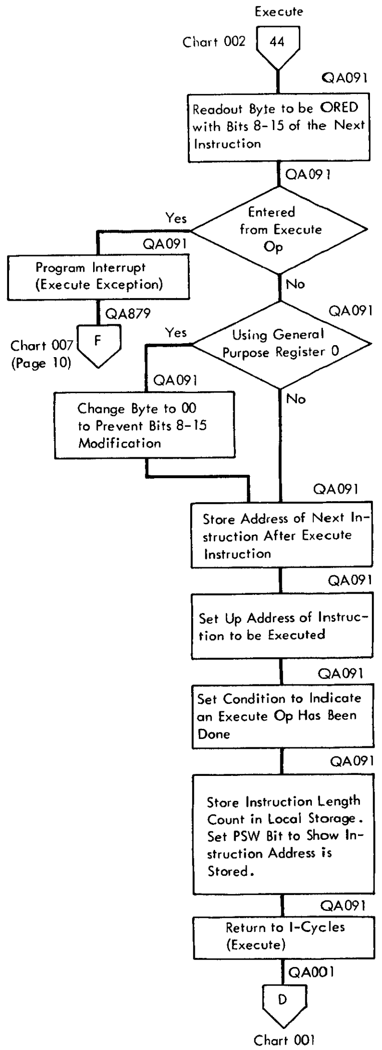
Op Format	Byte 1	Byte 2	Byte 3	Byte 4
RR Compare	19	R1 R2		
RR Add	1A	R1 R2		
RR Subtract	1B	R1 R2		
RR Add Logical	1E	R1 R2		
RR Subtract Log.	1F	R1 R2		
RX Half Compare	49	R1 X2	B2 D2	D2
RX Half Add	4A	R1 X2	B2 D2	D2
RX Half Subtract	4B	R1 X2	B2 D2	D2
RX Compare	59	R1 X2	B2 D2	D2
RX Add	5A	R1 X2	B2 D2	D2
RX Subtract	5B	R1 X2	B2 D2	D2
RX Add Logical	5E	R1 X2	B2 D2	D2
RX Subtract Log.	5F	R1 X2	B2 D2	D2

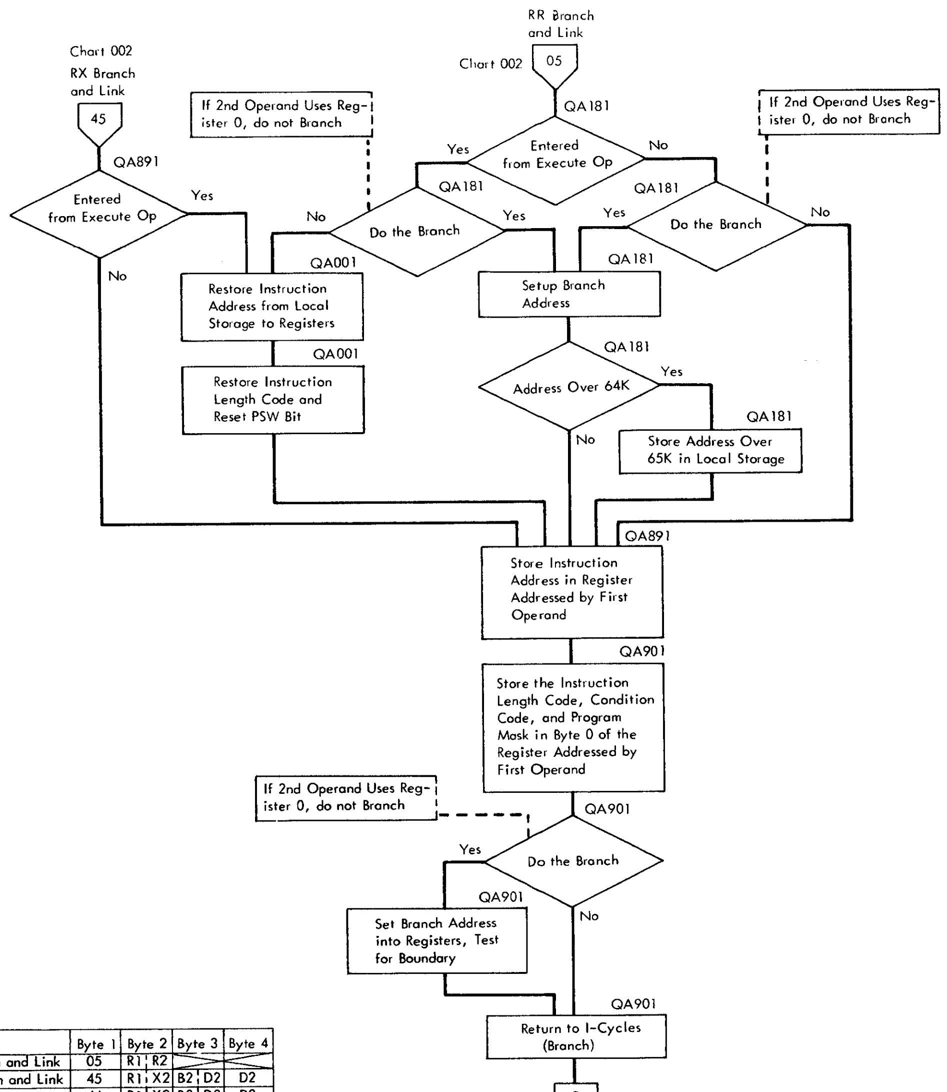
Objective:

Add	- The second operand is added to the first operand, and the sum is placed in the first operand location.
Add Logical	- Same as Add. The difference is in the setting of the condition code.
Subtract	- The second operand is subtracted from the first operand, and the difference is placed in the first operand location.
Subtract Log.	- Same as Subtract. The difference is in the setting of the condition code.
Compare	- The first operand is compared with the second operand, and the result determines the setting of the condition code.



CLF 005 RR-RX Fixed Point, Multiple Codes 3

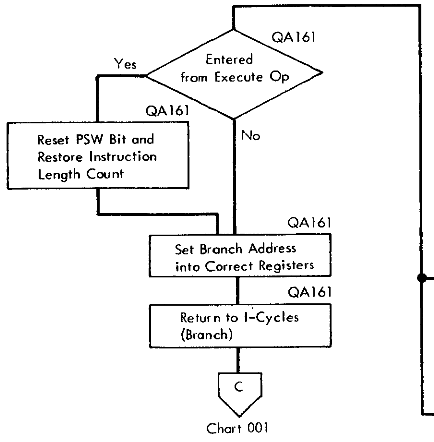


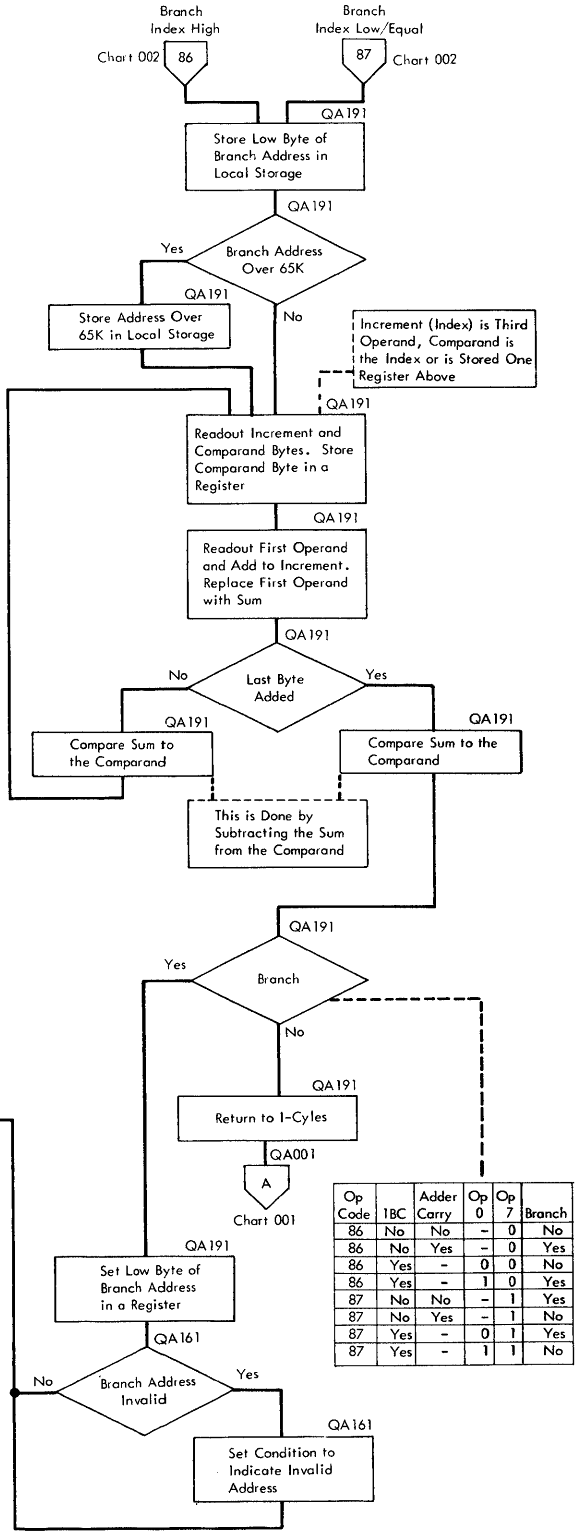


Op Format	Byte 1	Byte 2	Byte 3	Byte 4
RR - Branch and Link	05	R1 R2		
RX - Branch and Link	45	R1 X2	B2 D2	D2
RX - Execute	44	R1 X2	B2 D2	D2
RS - Branch High	86	R1 R3	R2 D2	D2
RS - Branch Low/Eq.	87	R1 R3	R2 D2	D2

Objective:

Branch and Link -	Store the Updated Rightmost 32 Bits of the Current PSW and Set the Branch Address into the Instruction Counter.
Execute -	Bits 8 - 15 of the Instruction at the Branch Address, Second Operand Effective Address, is Modified by ORING Byte 3 of R1 with Bits 8 - 15. The Instruction as Modified Will Now be Processed.
Branch High -	Add the Contents of R3 to R1 and is Compared to the Contents of R3 or R3 + 1. If the Compare is High, a Branch is Taken to the Effective Address of the Second Operand.
Branch Low/Equal -	Same as Branch High only Branch on Equal or Low Compare.





Op Code	1BC	Adder Carry	Op 0	Op 7	Branch
86	No	No	-	0	No
86	No	Yes	-	0	Yes
86	Yes	-	0	0	No
86	Yes	-	1	0	Yes
87	No	No	-	1	Yes
87	No	Yes	-	1	No
87	Yes	-	0	1	Yes
87	Yes	-	1	1	No

	1st Operand	Increment	Comparand
Start	00 00 02 04	00 00 00 60	00 00 02 12
Finish	00 00 02 64	00 00 00 60	00 00 02 12
Low Order Byte Address	23	43	53

Readout Low-Order Bytes, Add A to C, Subtract Sum of A+C from B.
Replace 1st Operand Byte with Sum of A + C.

1st Operand Address	Increment Address	Comparand Address	Register A	Register B	Register C	1st Operand
2 3	4 3	5 3	-	-	-	00 00 02 04
	Readout 4 3		6 0			
		Readout 5 3		1 2		
Readout 2 3				No Carry A E	0 4	00 00 02 64
2 2	4 2	5 2				
Readout Next Byte, Repeat Routine						
	Readout 4 2		0 0			
		Readout 5 2		0 2		
Readout 2 2				No Carry F F	0 2	00 00 02 64
2 1	4 1	5 1				
Readout Next Byte, Repeat Routine						
	Readout 4 1		0 0			
		Readout 5 1		0 0		
Readout 2 1				No Carry F F	0 0	00 00 02 64
2 0	4 0	5 0				
Readout Next (Last) Byte, Repeat Routine						
	Readout 4 0		0 0			
		Readout 5 0		0 0		
Readout 2 0				No Carry F F	0 0	00 00 02 64

Check for a Carry from Bit 1 (in Example, Bit 1 Carry did not Occur)
Now Add High 4 Bits of the Last Compare, B-(A+C), to Op Code.

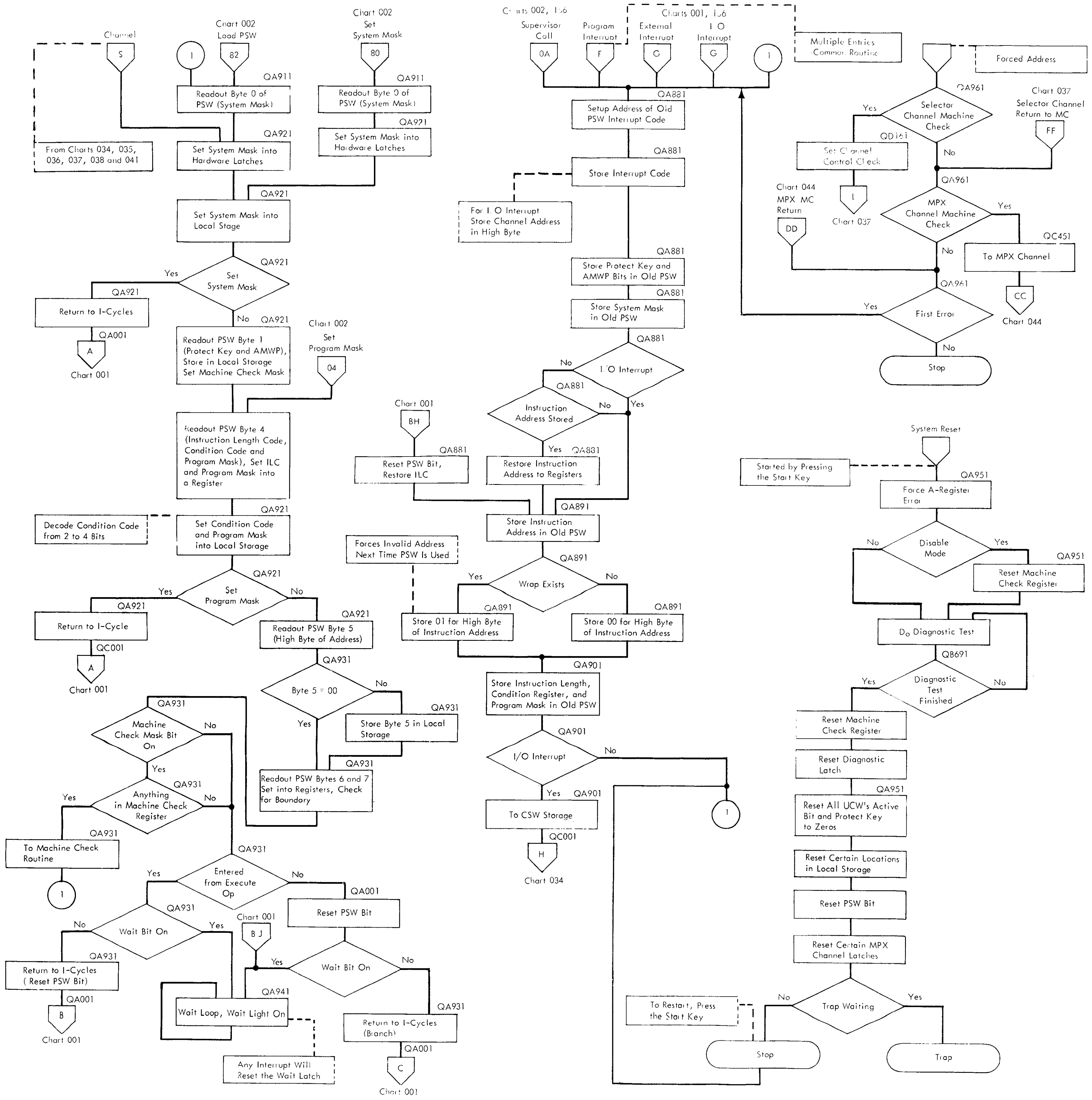
Branch High
86 = 1000 0110
1111 0110 No Carry ry

Adder Carry 0111 0110

Branch Low/Equal
87 = 1000 0111
1111 0000 No Carry

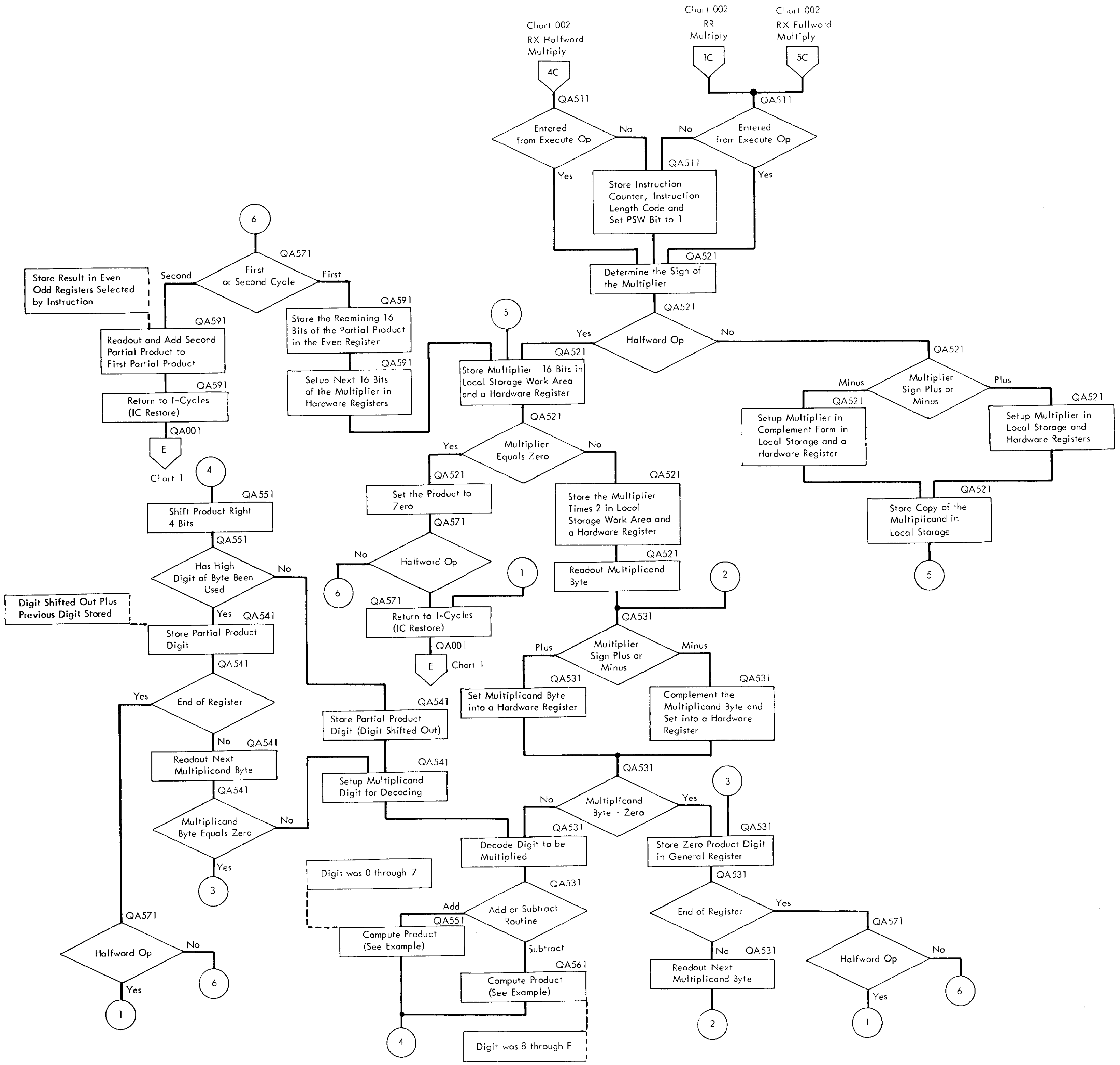
Adder Carry 0111 0111

Branches are Now Taken on Result of this Add, Adder Carry and Op Bit 7, to Determine if the Branch is to be Taken. If a Carry from Bit 1 had Occurred then the Branch is Determined by Op Bit 0 and Op Bit 7 after the Add.



CLF 007 RR Supervisor Call, Interrupts and System Reset

CLF 008 RR-RX Binary Multiply



Multiply Examples:

Multiply Table

Multiplicand Digit	Multiplier Value
0	+0
1	+1X
2	+2X
3	+2X +1X
4	+2X +2X
5	+2X +2X +1X
6	+2X +2X +2X
7	+2X +2X +2X +1X
8	-2X -2X -2X -2X
9	-2X -2X -2X -1X
A	-2X -2X -2X
B	-2X -2X -1X
C	-2X -2X
D	-2X -1X
E	-2X
F	-1X

This Same Table is Used for Add or Subtract. If the Last Cycle was a Subtract, Add One to the Multiplicand Digit.

RX Halfword Multiply

4C	R1	X2	B2	D2

Multiplier is 16 Bits
Multiplicand is 32 Bits
Product is 32 Bits

Operation is the Same as Fullword Except Only One Partial Product is Formed. The Low 32 Bits of This Product are Set into a Registeras the Product.
R1 can be an Even or Odd Numbered Register.

RR Multiply

1C	R1	R2

RX Fullword Multiply

5C	R1	X2	B2	D2	D2

Multiplier is 32 Bits Long.
Multiplicand is 32 Bits Long.
Product is 64 Bits Long, Using Both R1 and R1+1.

The Multiplier is Found in R2 or at the Effective Address in Main Storage.

The General Register Addressed by R1 must be an Even Number Register. The Multiplicand is Found in the Odd Register After R1. Example: R1 Addresses General Register 6, the Multiplicand is in General Register 7.

The Product Replaces the Multiplicand.

R1 Even Register — R1+1 Odd Register

32 Bits (Not Used)	S	31 Bits	S	31 Bits

Product — Multiplicand — Multiplier

To do a Full Word Multiply, Two Partial Products are Developed, then Added together to Form the Product.
Example of Full Word Multiply. Multiplier = 00100024 Multiplicand = 00300904

1. Readout Multiplier, Complement if Minus. Put the High-Order 16 Bits into the 16 High-Order Bits of General Register Addressed by R1. Put Low-Order 16 Bits into Hardware Registers. Set a Copy of the Multiplicand into Local Storage.

R1	R1 + 1	Hardware Registers	Local Storage
00 10 XX XX	00 30 09 04	00 | 24	00 30 09 04
High 16 Bits of Multiplier	Multiplicand	Low 16 Bits of Multiplier	Copy of Multiplicand

2. Setup in Local Storage and Hardware Register 1 Times and 2 Times the Low 16 Bits of the Multiplier
   1X = 0024
   2X = 0048

3. Develop the First Partial Product by Reading Out the Multiplicand, Decoding each Digit and Adding or Subtracting the 1X or 2X of the Low Multiplier. The Product Digits Replace the Multiplicand and the Low-Order 16 Bits of the Even Register. Put High 16 Bits of the Multiplier in the Hardward Registers.

R1	R1 + 1	Hardware Registers
00 10 00 00	06 C1 44 96	0 0 | 1 0
High 16 Bits of Multiplier	First Partial Product	High 16 Bits of Multiplier

4. Setup in Local Storage and Hardware Registers 1 Times and 2 Times the Hihg 16 Bits of the Multipl
   1X = 0010
   2X - 0020

5. Develop the Second Partial Product by Reading Out the Copy of the Multiplicand, Decoding Each Digit and Adding or Subtracting the 1X or 2X of the High Multiplier. The Second Product Digits Replaces the Multiplicand Copy and the High 16 Bits are Stored in Hardware Registers.

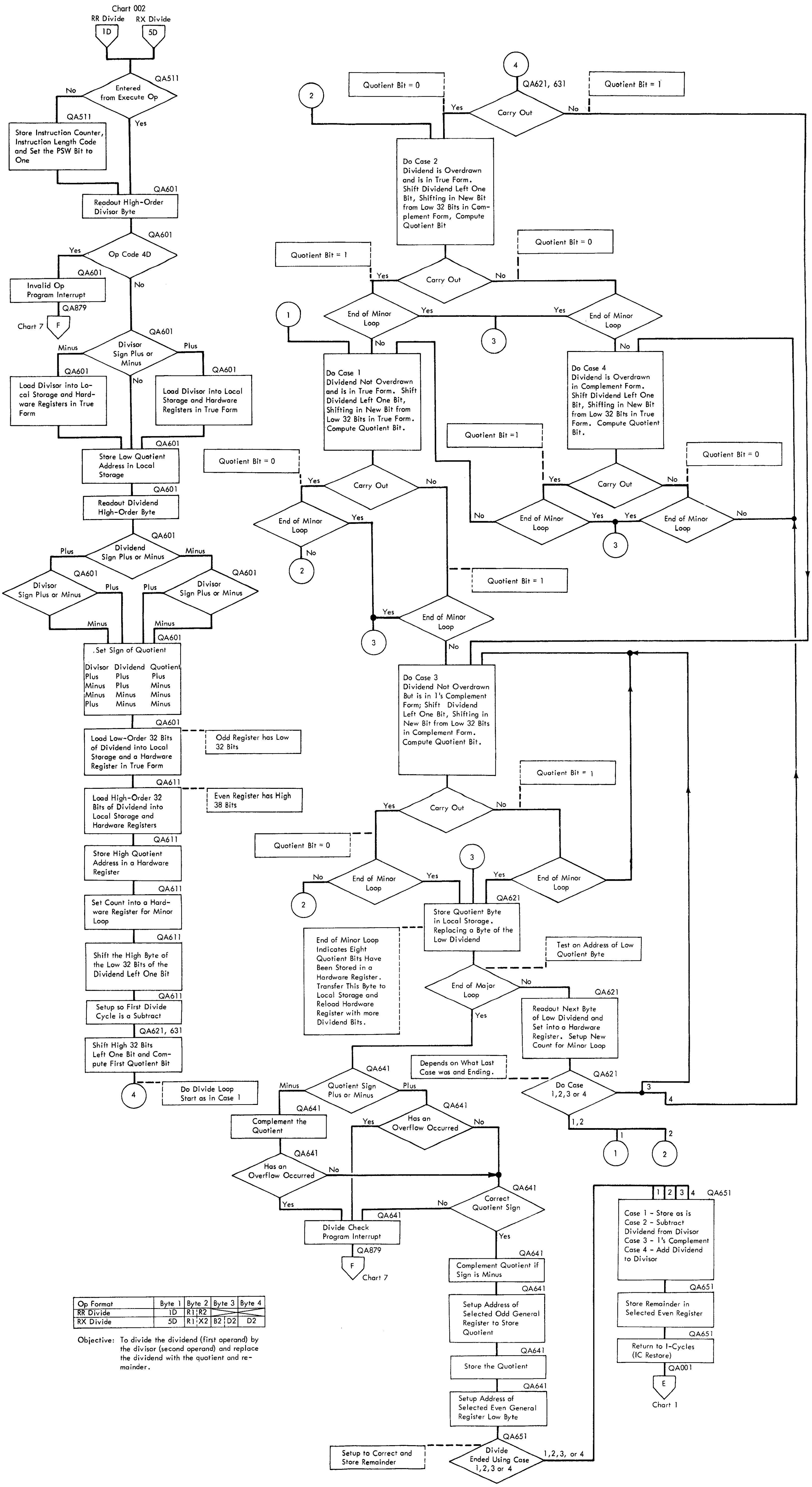
Hardware Register	Local Storage
0 0 | 0 0	03 00 90 40

High 16 Bits of Second Partial Product. Low 32 Bits of Second Partial Product.

6. Add the Second Partial Product to the First Partial Product. The Second Partial Product is Shifted the Correct Amount. Result is Set into the Even and Odd Registers Selected by R1.

```
        00 00 06 C1 44 96  First Partial Product
+00 00 03 00 90 40         Second Partial Product
```

R1	R1+1
00 00 03 00	97 01 44 96



Op Format	Byte 1	Byte 2		Byte 3		Byte 4
RR Divide	1D	R1	R2			
RX Divide	5D	R1	X2	B2	D2	D2

Objective: To divide the dividend (first operand) by the divisor (second operand) and replace the dividend with the quotient and remainder.

Note: For this example, assume a divisor of 16 bits and a dividend of 32 bits. This is done to shorten the example but still show how the auotient is developed.

1. Divisor = 1 0 | 5 2 The divisor is readout of its General Register or Main Storage location and copied into Local Storage and a hardware register.

2. Signs. Check sign of divisor and dividend and set quotient sing. In example, both signs are plus, so the quotient sign is plus.

3. Dividend = 07 | 35 | FB | 0A (Even, Odd) The dividend is located in two General Registers. The high dividend in the even-numbered register and the low dividend in the odd-numbered register. The dividend is readout and the two parts, high and low, are copied into Local Storage and hardware registers.

4. Quotient. The quotient is developed 1 bit at a time and set into a hardware register until the end of a minor loop. Then it is placed in local storage.

Note: Shifting of the dividend uses the copy in local storage work area. The 4 low bits of a hardware register are set to F to be used in the minor loop count. When the 4 low bits equal 7, the minor loop is ended.

Note: Setup to start as in case 1.

5. Set high byte of low dividend into a hardware register.
6. Shift byte left 1 bit.

7. Add carry in high dividend, shift left 1 bit and do 1's complement.

8. Add complemented high dividend to divisor to compute quotient bit and new high dividend.
Case 1, Carry out = Quotient bit 0 and do case 2.
9. Shift byte left 1 bit and set quotient bit in low-order position, subtract 1 from count.

10. Add carry in high dividend, shift left 1 bit.

11. Subtract dividend from divisor.

Case 2, Carry out = Quotient bit 1 and do case 1.
12. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

13. Add carry in high dividend, shift left 1 bit.

14. Subtract dividend from divisor.

Case 1, No Carry = Quotient bit 1 and do case 3.
15. Shift byte left 1 bit and set quotient bit in.. Subtract 1 from count.

16. Add carry in high dividend, Shift left 1 bit.

17. Add dividend to divisor.

Case 3, No carry = Quotient bit 1 and do case 3.
18. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

19. Add carry in high dividend, Shift left 1 bit.

20. Add dividend to divisor.

Case 3, Carry out = Quotient bit 0 and do case 2.

21. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

22. Shift high dividend left 1 bit.

23. Subtract dividend from divisor.

Case 2, No Carry = Quotient bit 0 and do case 4.
24. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

25. Add carry in high dividend, Shift left 1 bit.

26. Add dividend to divisor.

Case 4, No Carry = Quotient bit 0 and do case 4.
27. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

28. Add carry in high dividend, Shift left 1 bit.

29. Add dividend to divisor.

Case 4, Carry out = Quotient bit 1 and do case 1.
30. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

31. Count has gone to 7. End of minor loop. A quotient byte is in the hardware register. The quotient byte is set into local storage replacing the high byte of the low dividend copy. The low byte of the low dividend is placed in the hardware register for shifting and the count is set to F for the next minor loop.

32. Shift high dividend left 1 bit.

33. Subtract dividend from divisor.

Case 1, Carry out = Quotient bit 0 and do case 2
34. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

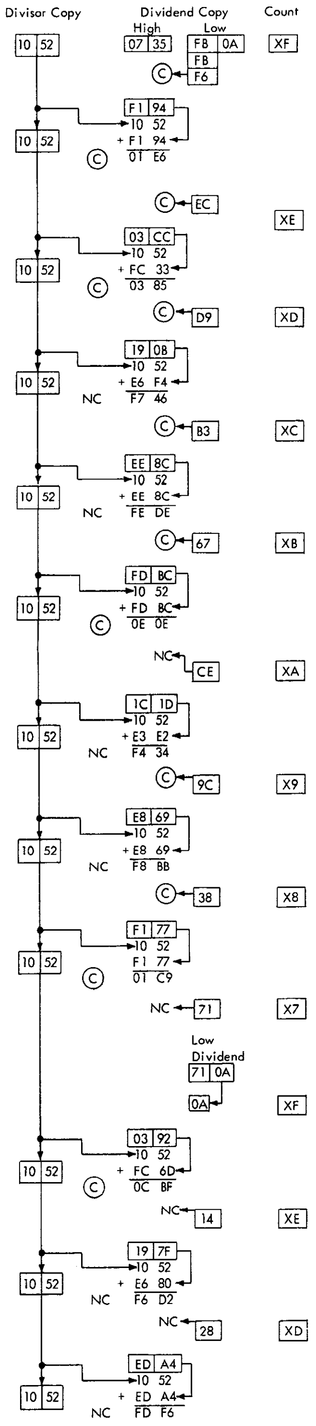
35. Shift high dividend left 1 bit.

36. Subtract dividend from divisor.

Case 2, No Carry = Quotient bit 0 and do case 4.
37. Shift byte left 1 bit and set quotient bit in. Subtract 1 from count.

38. Shift high dividend left 1 bit.

Case 4, No Carry = Quotient bit 0 and do case 4.



Do case 4.

40. Shift byte left 1 bit. Set quotient bit in and subtract 1 from count.

41. Shift high dividend left 1 bit.

42. Add dividend to divisor.

Case 4, Carry out = Quotient bit 1 and do case 1.

43. Shift byte left 1 bit. Set quotient bit in and subtract 1 from count.

44. Shift highdivided left 1 bit.

45. Subtract dividend from divisor.

Case 1, No Carry = Quotient bit 1 and do case 3.

46. Shift byte left 1 bit Set quotient bit in and subtract 1 from count.

47. Shift high dividend left 1 bit.

48. Add dividend to divisor.

Case 3, No Carry = Quotient bit 1 and do case 3,
49. Shift byte left 1 bit. Set quotient bit in and subtract 1 from count.

50. Shift high dividend left 1 bit

51. Add dividend to divisor.

Case 3, Carry out = Quotient bit 0 and do case 2.
52. Shift byte left 1 bit, Set quotient bit in and subtract 1 from count.

53. Shift high dividend left 1 bit.

54. Subtract dividend from divisor

Case 2, No Carry = Quotient bit 0 and do case 4.
55. Shift byte left 1 bit. Set quotient bit in and subtract 1 from count.
56. Count has gone to 7. End of minor loop. Set quotient byte into low byte of low dividend.
57. Address of low quotient byte has been reached indicating end of major loop. Check signs and set the quotient into the odd numbered general register.

58. The remainder is developed and set into the even numbered general register. The correct sign is set for the reaminder. In the example, we ended in case 4. Case 4 causes the high dividend that has been developed to be added to the divisor. The result is the remainder.

The even/odd general registers have been set with the remainder and quotient with proper sign and divide is ended.

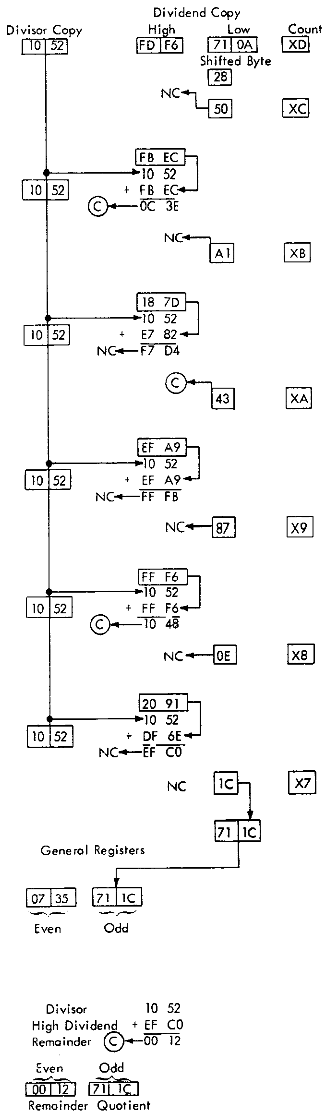
Note: The divisor and dividend in actual form would be:

Divisor 00 | 00 | 10 | 52

Dividend 00 | 00 | 00 | 00 | 07 | 35 | F8 | 0A

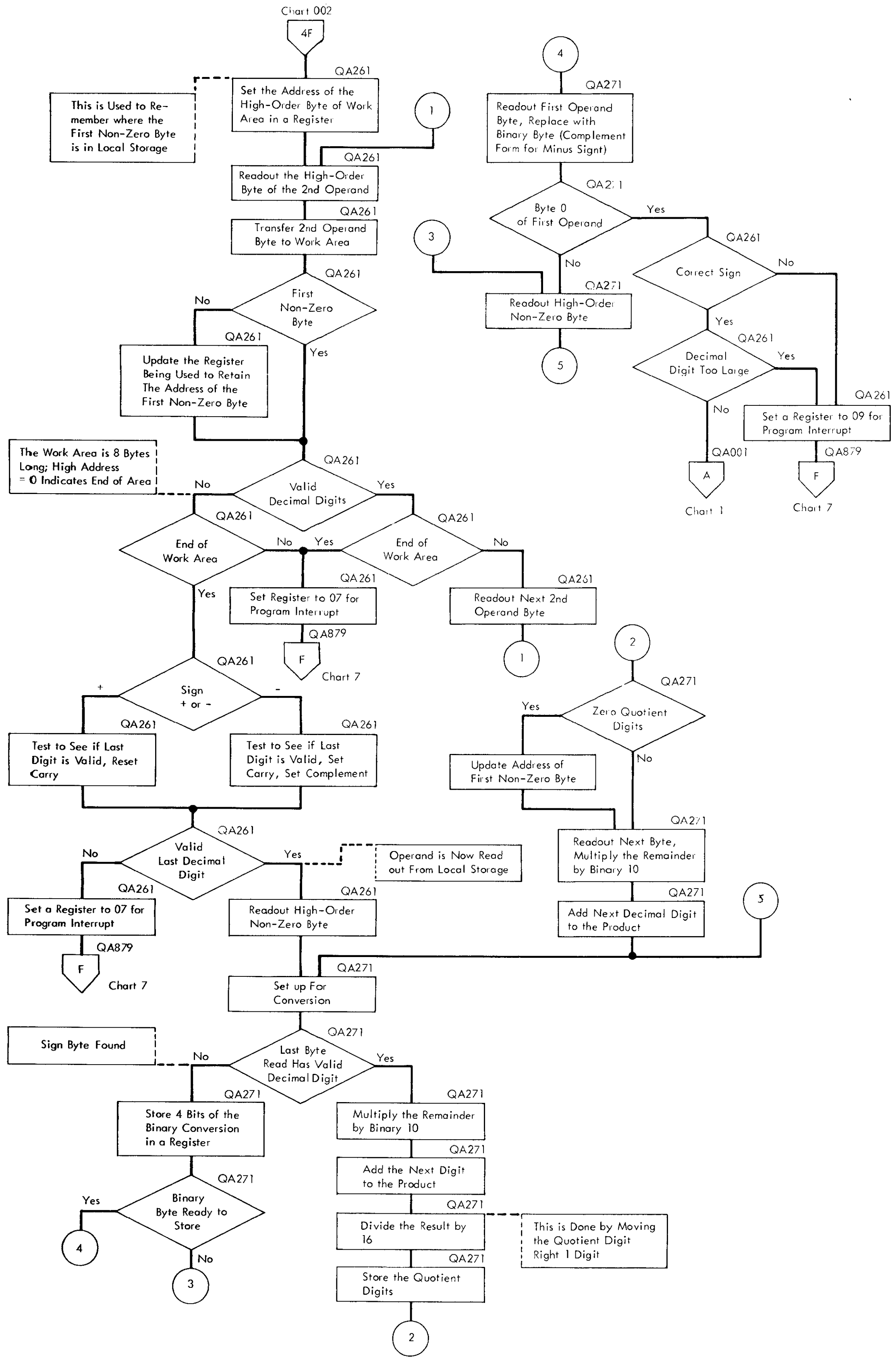
The Quotient would be: Even | Odd

Quotient 00 | 00 | 00 | 12 | 00 | 00 | 71 | 1C



CLF 010 RR-RX Binary Divide Example

RX Convert to Binary



Instruction

Format	Byte 0	Byte 1	Byte 2	Byte 3
RX	4F	R1 X2	B2 D	2

First Operand

Start	XX	XX	XX	XX
Finish	00	00	11	61

Second Operand - Main Storage

Byte	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Digits	00	00	00	00	00	04	44	9C

Work Area - Local Storage

Load	00	00	00	00	00	04	44	9C
Finish	00	00	00	00	00	00	00	0C

The convert to binary routine works something like a manual conversion of decimal to binary (hexadecimal). Example:

```
   0278            017           01        0
16)04449       16)0278       16)017    16)01
   00             00            00        00
    44             27            17        1
    32             16            16
    124            118           1
    112            112
     129           6
     128
      1
      |
(Hex) 1161    0001 0001 0110 0001 (Binary)
```

By successively dividing the decimal number by 16 and using the remainder the hexadecimal number 1161 is developed.

The 2030 converts a decimal number to a binary number by multiplying each digit by binary 10 and then adding the next digit to the product. Then this result is divided by 16. The Quotient is stored and the Remainder is multiplied by 10. This continues until the number has been converted.

In the example to the right, the register numbers are for example only. The work area has been loaded and R4 has the location of the first high-order non-zero byte.

To multiply a digit by binary 10, shift the number left 3 times (by adding). Then add the result of the first shift to the third shift----
Examples:

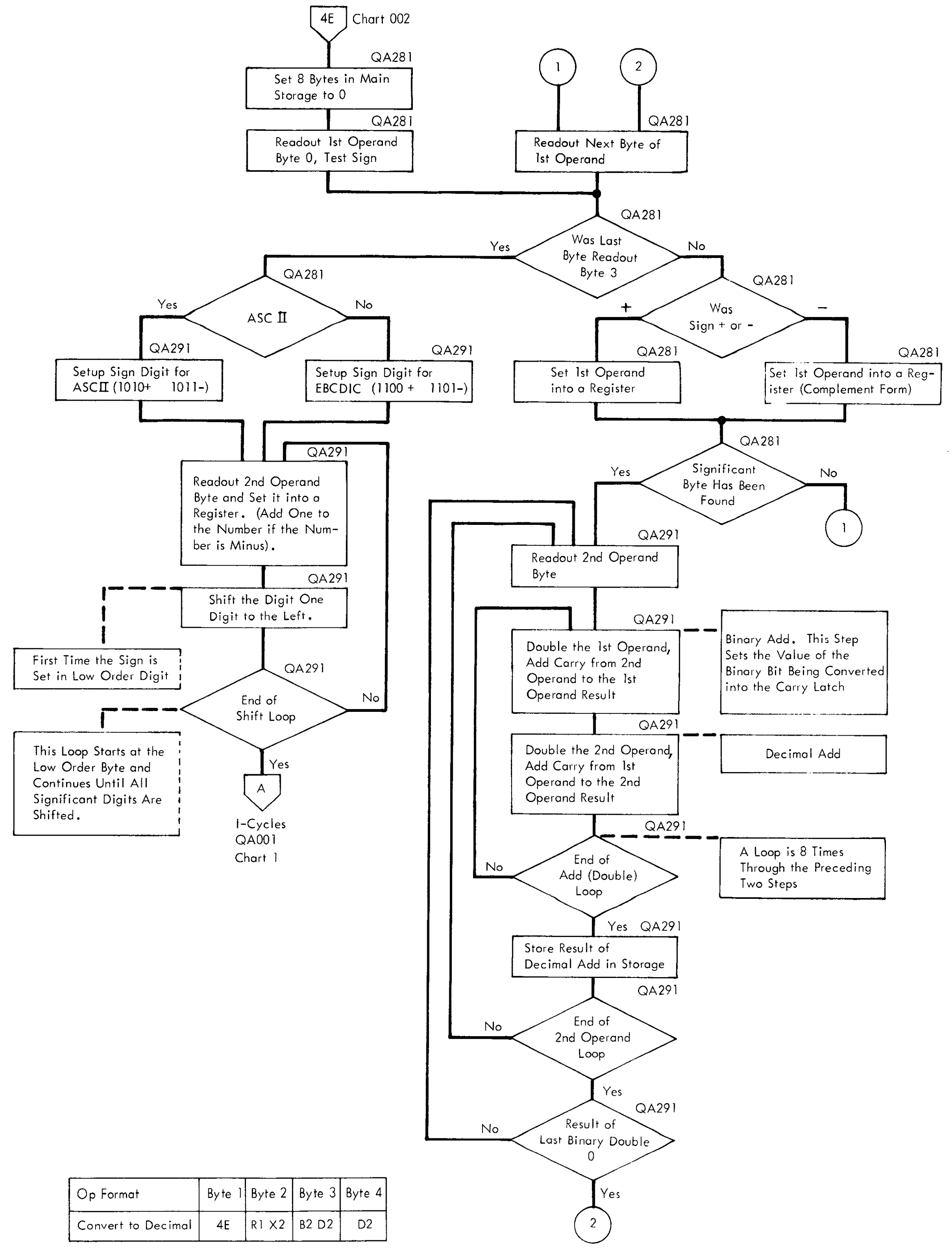
```
04  0000  0100
04 +0000  0100       1
08  0000  1000 --
08  0000  1000       2
16  0001  0000
16  0002  0000       3
32  0010  0000
 8  0000  1000 <-
40  0010  1000
```

To divide a digit by 16, the digit is shifted 1 digit to the right.
Example:

```
    2       64 48 32 16  8 4 2 1
16)33        0  0  1  0  0 0 0 1 -- Remainder
   32
    1                    0 0 1 0 -- Quotient
```

In Example R1, 2, and 3 are used to convert the number. R4 retains the work area byte with the first significant digit. R5 hold the low four bits of each convert byte for storing complete byte.

				Work Area					First Operand		
Comments	R1	R2	R3	Byte 5	Byte 6	Byte 7	R4	R5	Byte 1	Byte 2	Byte 3
Set R1 to 00 and set byte 5 in R2. Note:	00	04		04	44	9C	5		00	00	00
The R1 high digit is called Quotient (Q1) and the	Quotient	1st									
R1 low digit is called the Remainder.	Remainder	Next									
	00										
The R2 1st digit is added to the Remainder. Q1	+0										
2nd R2 next digit are set into R3. A test for	00		04								
sign digit is made. Multiply by 10 if the byte	xA										
does not contain a sign digit.	00										
Add next digit to the product.	+4										
Divide by 16. This is done by shifting the high	04										
digit right 1 digit. Store Q1 and the Quotient	0			00	44	9C	6				
(Q2) of this divide in the work area. If bytes											
stored equal zero, change R4.											
Readout next byte. Multiply the Remainder by	04	44									
10.	xA										
The Product and 1st digit are added. Q1 and	28										
next digit are set into R3. Test for sign.	+4										
	2C		24								
	0C										
Multiply remainder by 10.	xA										
	78										
Add next digit.	+4										
	7C										
Divide by 16, store Q1 and Q2, test for Zero.	7			00	27		6				
Readout next byte.	0C	9C									
Multiply Remainder by 10.	xA										
	78										
Add 1st digit.	+9										
Set the Quotient and next digit in R3, test for sign.	81		8C								
Sign is found, set remainder in R5 and set R3 into low											
byte of work area.				00	27	8C	6	1			
Set R1 to 00 and set byte 6 in R2	00	27		00	27	8C					
Add 1st digit to Remainder.	+2										
Set Q1 and next digit in R3, test for sign.	02		07								
Multiply Remainder by 10.	xA										
	14										
Add next digit.	+7										
Divide by 16, store Q1 and Q2, test for zero quotients.	1B										
	1			00	01	8C	6				
Readout next byte.	0B	8C									
Multiply Remainder by 10.	xA										
Add 1st digit.	67										
	x8										
Set the Quotient and next digit in R3, test for sign.	76		7C						00	00	61
Sign is found. Store the remainder and low digit in											
R5 in the First Operand byte 3, store R3 in the low byte											
of the work area.				00	01	7C	6				
Set R1 to 00 and set byte 6 in R2.	00	01									
Add 1st digit.	+0										
Set Q1 and next digit in R3, test for sign.	00		01								
multiply Remainder by 10.	xA										
	00										
	+1										
Divide by 16, store Q1 and Q2, test for zero quotient.	01										
Zeros found. Change R4.	0			00	00	7C	7				
Readout next byte.	01	7C									
Multiple Remainder by 10.	xA										
	0A										
Add 1st digit.	+7										
Set the Quotient and next digit in R3, test for sign.	11		1C								
Sign is found. Set the Remainder in R5 and set R3 in											
low byte of the work area.				00	00	1C		1			
Set R1 to 00 and set byte 7in R2.	00	1C									
	+1										
Set Q1 and next digit in R3, test for sign.	01		0C						00	11	61
Sign is found. Store remainder and low digit in R5 in the											
First Operand byte 2, store R3 in low byte of the work area.				00	00	0C					
The number has been converted, but the loop continues											
until the first operand remaining bytes have been set to 00.											



Op Format	Byte 1	Byte 2	Byte 3	Byte 4
Convert to Decimal	4E	R1 X2	B2 D2	D2

First Operand

Byte 0	Byte 1	Byte 2	Byte 3
00	00	11	61

Second Operand

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Zeroed	00	00	00	00	00	00	00	00
Added	00	00	00	00	00	00	44	49
Shifted	00	00	00	00	00	04	44	9C

To Manually Convert the Hexadecimal Number 1161 to Decimal, multiply each Digit by 16, then Add the Next Digit to the Product. The Sum is Multiplied by 16 and the Next Digit Added, etc.

Example:

```
  01   1  6  1
 x16   |  |  |
  06   |  |  |
  10   |  |  |
  16   |  |  |
  +1 --+  |  |
  17      |  |
 x16      |  |
 102      |  |
  17      |  |
 272      |  |
 + 6 -----+  |
 278         |
 x16         |
1668         |
 278         |
4448         |
+  1 --------+
4449
```

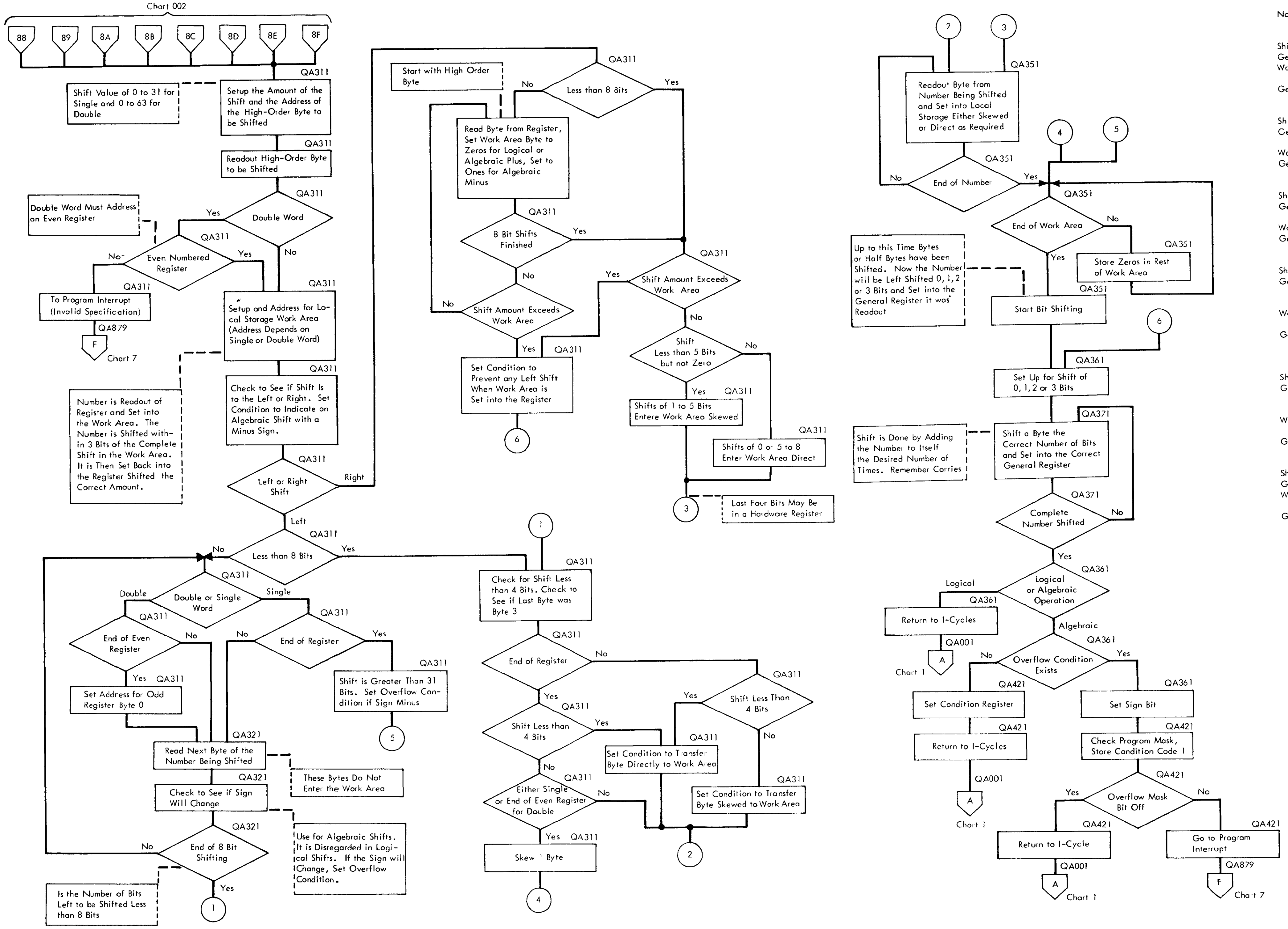
The Operation in the 2030 is Similar to the Manual Operation. Each Byte is Doubled Eight Times. Whenever a Carry Occurs from the ALU During the Binary Add, it is Set into a Register and is Decimally Added to Itself. Any Carry during a Decimal Add is Added to the Result of the Next Binary Add. This Continues until the Binary Number is Converted to a Decimal Number. The Decimal Number is then Shifted Left One Digit so the Correct Sign can be Stored in the Low Order Digit of the Low Order Byte.

In the Example at the Right, the Register Numbers are for this Example Only. The 2nd Operand has been Set to all Zeros and the 1st Significant Byte of the 1st Operand is in R1. Note: RX is shown in Binary Form while RY and the 2nd Operand are shown in Decimal.

Set Sign in RX, RY to RZ;
RX Low to RY Low; RZ Low to RY High;
RZ High to RX Low;
RY to RZ, etc. until the Shift of the
Decimal Number is Completed.

```
Carry  Binary Add    Carry  Decimal Add    Storage - 8 Bytes
       Register X           Register Y     2nd Operand

       0001 0001             0 0           0000000000000000
       0001 0001
       0010 0010             0 0
                             0 0
       0010 0010
       0100 0100             0 0
                             0 0
       0100 0100
       1000 1000             0 0
                             0 0
       1000 1000
  1    0001 0000             0 0
                               1
       0001 0000             0 1
       0010 0000             0 1
       0010 0000             0 2
       0100 0000             0 2
       0100 0000             0 4
       1000 0000             0 4
       1000 0000             0 8
  1    0000 0000             0 8
                               1
                             1 7
                                           0000000000000017
       0000 0000
       0000 0000
       End of Add Loop, End of 2nd Operand Loop, and Result of Binary Add  0
       Readout Byte 3

       0110 0001             1 7
       0110 0001
       1100 0010             1 7
                             3 4
       1100 0010
  1    1000 0100             3 4
                               1
       1000 0100             6 9
  1    0000 1000             6 9
                               1
       0000 1000  <------ 1  3 9
               1
       0001 0001             3 9
                             7 8
       0001 0001
       0010 0010             7 8
       0010 0010  <------ 1  5 6
               1
       0100 0101             5 6
       0100 0101  <------ 1  1 2
               1
       1000 1011             1 2
       1000 1011             2 4
  1    0001 0110             2 4
                               1
                             4 9           0000000000000049
       0001 0110
       0010 1100
       End of Add Loop, Not the End of 2nd Operand Loop

       0010 1100             0 0
       0010 1100
       0101 1000             0 0
                             0 0
       0101 1000
       1011 0000             0 0
                             0 0
       1011 0000
  1    0110 0000             0 0
                               1
       0110 0000             0 1
       1100 0000             0 1
       1100 0000             0 2
  1    1000 0000             0 2
                               1
       1000 0000             0 5
  1    0000 0000             0 5
                               1
       0000 0000             1 1
       0000 0000             1 1
       0000 0000             2 2
       0000 0000             2 2
                             4 4           0000000000004449
       0000 0000
       0000 0000
       End of Add Loop, End of 2nd Operand Loop, and Result of Binary Add  0
       Last Byte Read Out was Byte 3
       Sign was Plus, not in ASCII Mode - Set Sign in R1
       Note: Contents in the Register for the Rest of the Example are in Hexadecimal.

       Register X     Register Y     Register Z     Storage

         F C -----+      4 9            4 9         0000000000004449
                  |      9 C                        000000000000449C
         0 4      +--------+
                         4 4            4 4
         0 4             4 4                        000000000000449C
                         0 0            0 0
         0 0             0 4                        000000000004449C
       End of Shift Loop
```

CLF 013 RS Shifts (Logical and Algebraic)



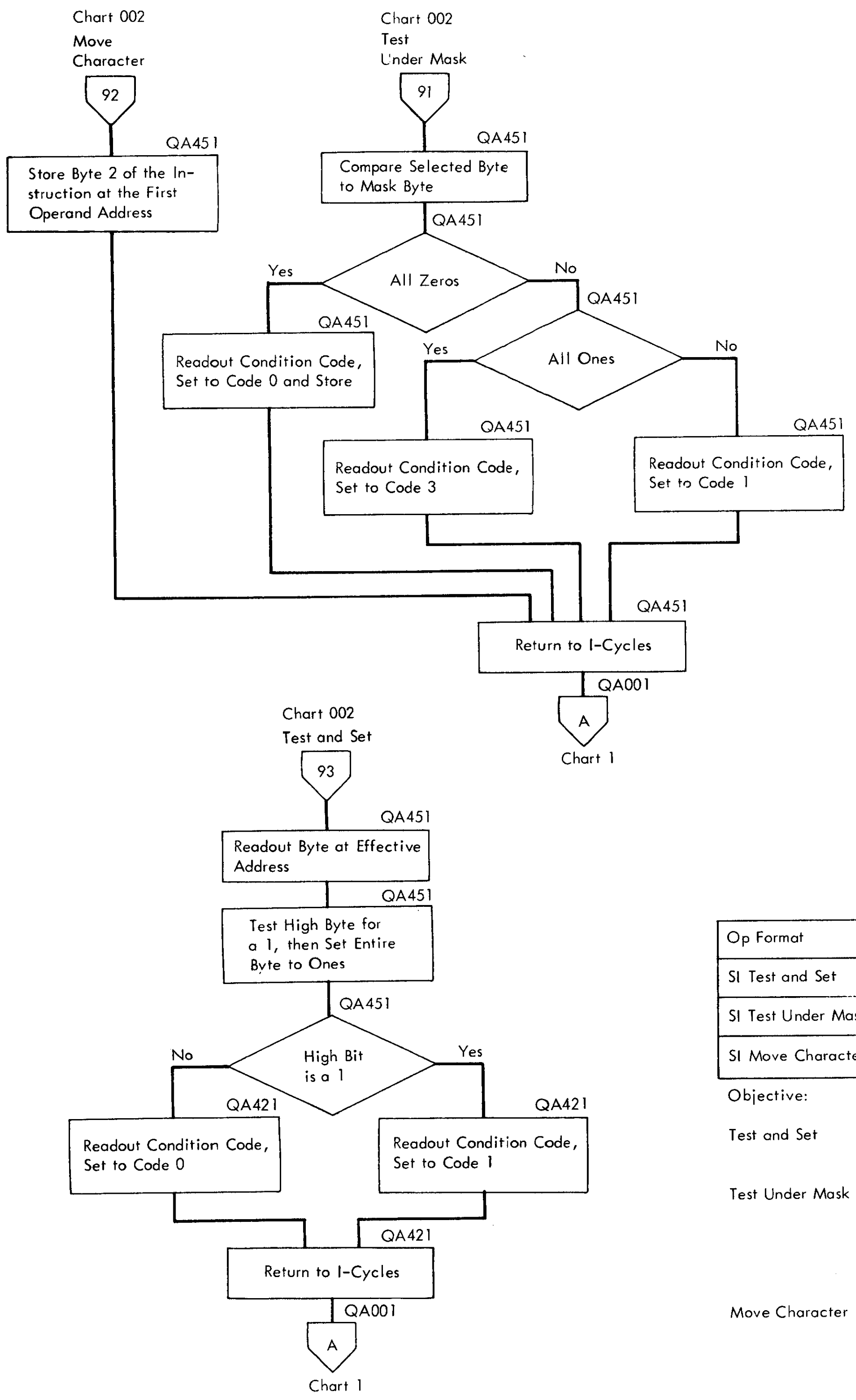
Note: All Examples Show a Single Word Shift.

Shift Left 3 Logical Single		
General Register Start	0000 0000 0000 0000 0101 0110 0001 0010	Transfer to Work Area Direct.
Work Area	0000 0000 0000 0000 0101 0110 0001 0010	Shift of 3 is Accomplished by Adding the Byte to Itself.
General Register Finished	0000 0000 0000 0010 1011 0000 1001 0000	Three Times and Remembering Carries.
Shift Left 10 Algebraic +		
General Register Start	0000 0000 0000 0001 0110 1000 0111 0001	Transfer to Work Area Direct but Shift 8 Bits.
Work Area	0000 0001 0110 1000 0111 0001 0000 0000	Add on 1 Byte of Zeros.
General Register Finished	0000 0101 1010 0001 1100 0100 0000 0000	Transfer Back to General Register, Shifted 2 Bits.
Shift Left 12 Algebraic –		
General Register Start	1111 1111 1111 1111 1001 0010 1101 1011	Transfer to Work Area Skewed Shifted 12 Bits.
Work Area	1111 1001 0010 1101 1011 0000 0000 0000	Add Low-Order Zeros.
General Register Finished	1111 1001 0010 1101 1011 0000 0000 0000	Register Direct.
Shift Right 3 Logical Single		
General Register Start	0000 0000 0000 0000 0101 0110 0001 0010	These Four Bits are held in a Hardware Register after First Shift.
Work Area	0000 0000 0000 0000 0000 0101 0110 0001	Transfer to Work Area Skewed. Low 4 Bits Shifted Out. Add High-Order Zeros.
General Register Finished	0000 0000 0000 0000 0000 1010 1100 0010	Transfer Back to General Register, Shifted 1 Bit.
Shift Right 12 Algebraic –		
General Register Start	1111 1111 1111 1111 1001 0010 1101 1011	These Four Bits are held in a Hardware Register.
Work Area	1111 1111 1111 1111 1111 1111 1111 1001	Transfer to Work Area Skiewed. Shift 12 Bits. Add High-Order Ones.
General Register Finished	1111 1111 1111 1111 1111 1111 1111 1001	
Shift Right 5 Logical Single		
General Register Start	0000 0000 0000 1001 1110 0001 1010 1001	This Byte held in a Hardware Register.
Work Area	0000 0000 0000 0000 0000 1001 1110 0001	Transfer to Work Area Direct but Shifted 8 Bits. Add High-Order Zeros.
General Register Finished	0000 0000 0000 0000 0100 1111 0000 1101	Transfer Back to General Register. Shift Left 3 Bits. Three Low-Order Bits are Determined by Byte in Hardware Register.

Op Format	Byte 1	Byte 2	Byte 3	Byte 4
Right Single Log.	8 8	R1 ▨	B2 D2	D2
Left Single Log.	8 9	R1 ▨	B2 D2	D2
Right Single Alg.	8 A	R1 ▨	B2 D2	D2
Left Single Alg.	8 B	R1 ▨	B2 D2	D2
Right Double Log.	8 C	R1 ▨	B2 D2	D2
Left Double Log	8 D	R1 ▨	B2 D2	D2
Right Double Alg	8 E	R1 ▨	B2 D2	D2
Left Double Alg	8 F	R1 ▨	B2 D2	D2

Objective:

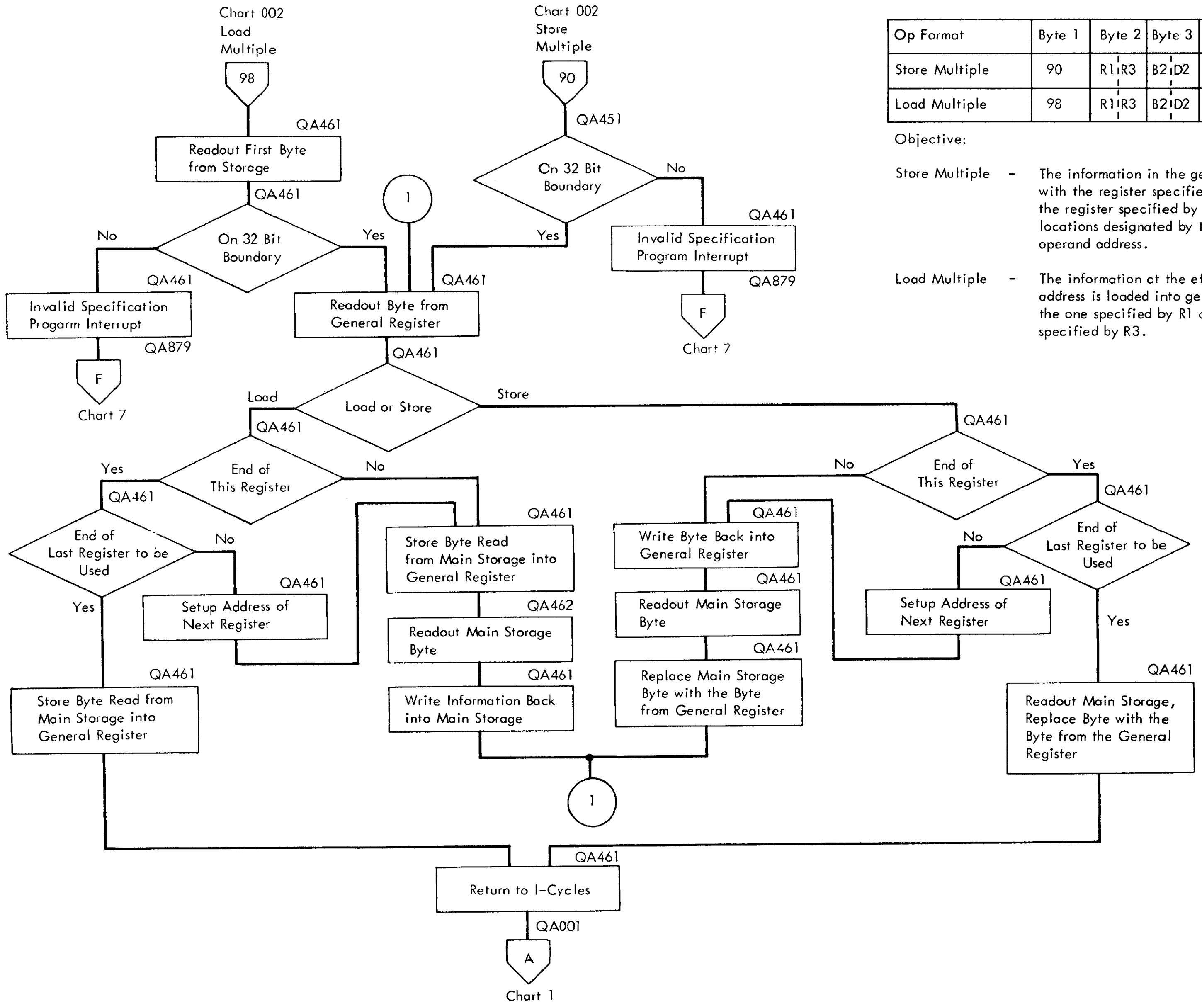
Shifts – To Obtain a Binary Value of the Second Operand Effective Address Low Order Six Bits. Example – 001011 = 11. Then shift the First Operand Either Left or Right that Amount. If the Shift is Algebraic, the Condition Code is Set.



Op Format	Byte 1	Byte 2	Byte 3	Byte 4
SI Test and Set	91	I2	B1 D1	D1
SI Test Under Mask	92	I2	B1 D1	D1
SI Move Character	93	I2	B1 D1	D1

Objective:

Test and Set	-	Test the high-order bit of the byte at the effective address and set condition code.
Test Under Mask	-	The byte of immediate data, I2, is used as a eight-bit mask. The bits of the mask are made to correspond one for one with the bits of the byte in storage specified by the first operand effective address.
Move Character	-	The second operand, one eight-bit byte, is placed at the effective address of the first operand.



Op Format	Byte 1	Byte 2	Byte 3	Byte 4
Store Multiple	90	R1 R3	B2 D2	D2
Load Multiple	98	R1 R3	B2 D2	D2

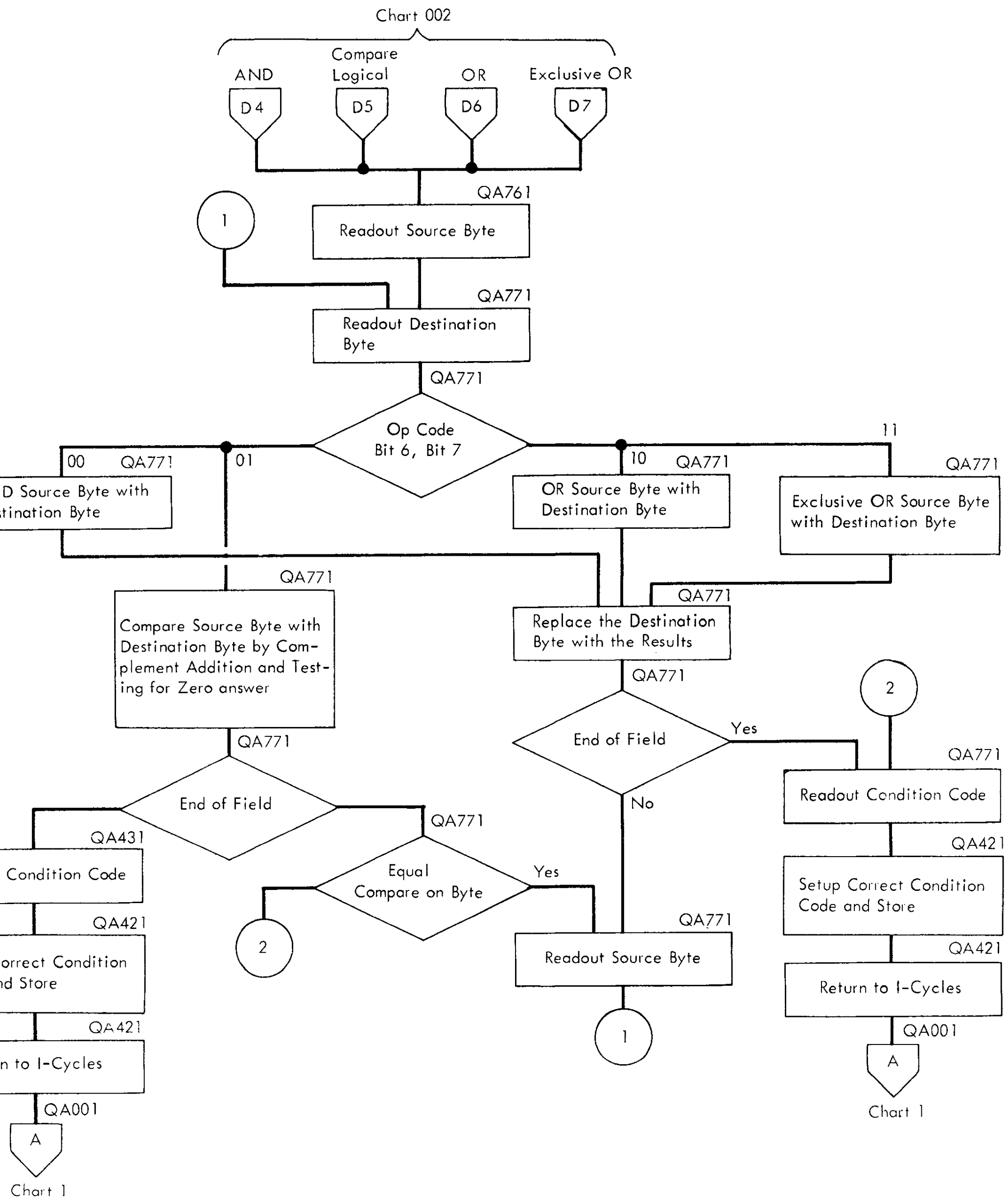
Objective:

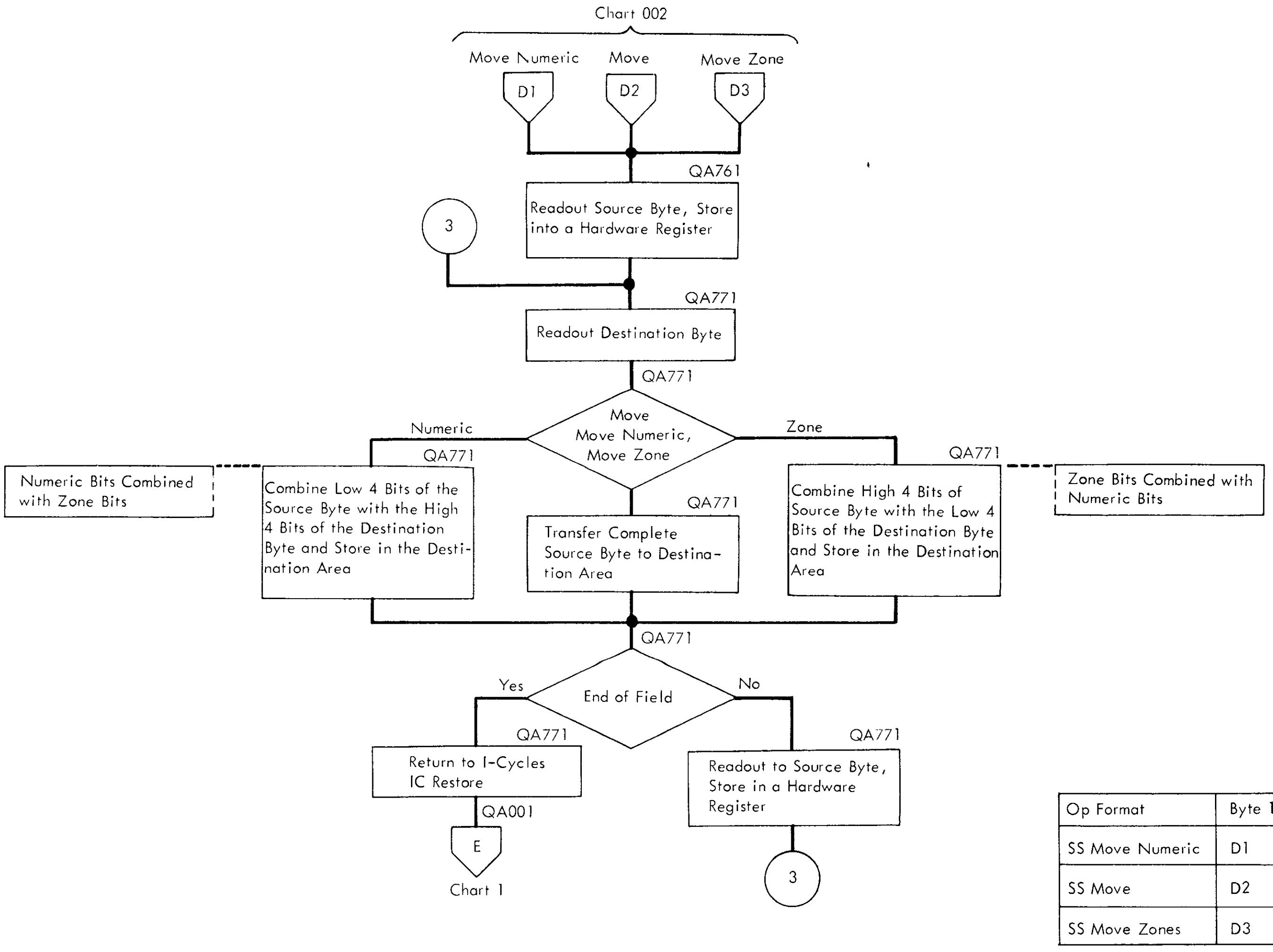
Store Multiple	-	The information in the general register starting with the register specified by R1 and ending with the register specified by R3 is stored at the locations designated by the effective second operand address.
Load Multiple	-	The information at the effective second operand address is loaded into general register starting with the one specified by R1 and ending with the one specified by R3.

Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS AND	D4	L	B1 D1	D1	B2 D2	D2
SS Compare	D5	L	B1 D1	D1	B2 D2	D2
SS OR	D6	L	B1 D1	D1	B2 D2	D2
SS Exclusive OR	D7	L	B1 D1	D1	B2 D2	D2

Objective:

And	- The logical answer (AND) of the bits of the first and second operand is placed in the first operand location.
Compare Logical	- The first operand is compared with the second operand, and the result is indicated in the condition code.
OR	- The logical sum (OR) of the bits of the first and second operand is placed in the first operand location.
Exclusive OR	- The modulo - two sum (exclusive OR) of the bits of the first and second operand is placed in the first operand location.



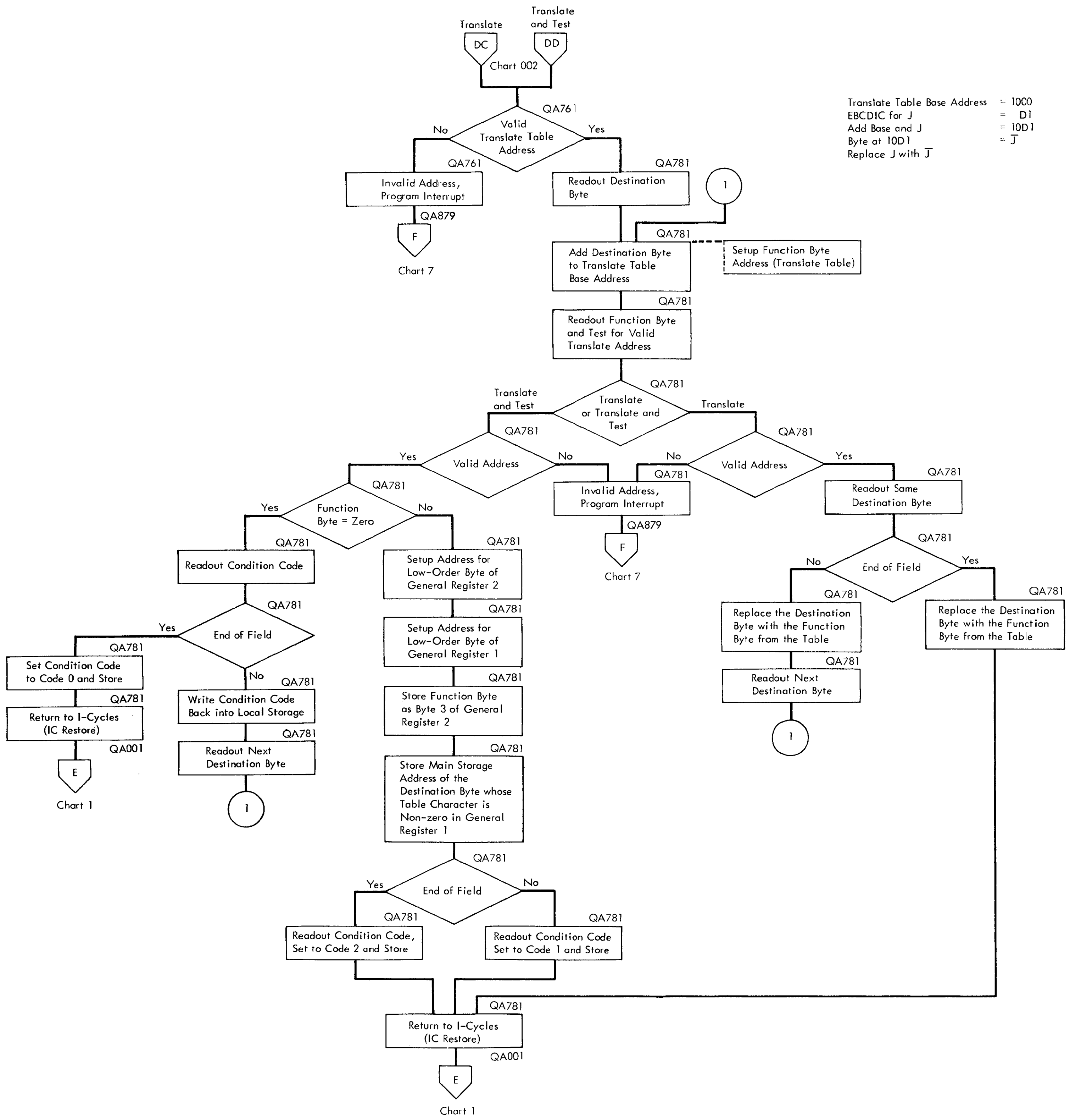


Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS Move Numeric	D1	L	B1 D1	D1	B2 D2	D2
SS Move	D2	L	B1 D1	D1	B2 D2	D2
SS Move Zones	D3	L	B1 D1	D1	B2 D2	D2

Objectives:

Move Numeric	-	The low-order four bits of each byte in the second operand field are placed in the low-order bit positions of the corresponding bytes in the first operand field. Movement is storage to storage, left to right.
Move	-	The second operand is placed in the first operand location. Movement is storage to storage, left to right.
Move Zones	-	The high-order four bits of each byte in the second opeand field are placed in the high-order four bit postions of the corresponding bytes in the first operand field. Movement is storage to storage, left to right.

CLF 016 SS Translate, Translate and Test



Translate Table Base Address = 1000
EBCDIC for J = D1
Add Base and J = 10D1
Byte at 10D1 = $\overline{\text{J}}$
Replace J with $\overline{\text{J}}$

Translate

Assume a stream of 20 characters comes into location 2100 in EBCDIC. Translate to ASCII.

Assume:

Reg 12	00 00 20 00
Reg 15	00 00 10 00
Loc 2100 - 2113 (before)	JOHN JONES 257 W. 95

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
TR	13	C	100	F	0

Loc 2100 - 2113 (after)	$\overline{\text{JOHN JONES 257 W. 95}}$

where the overbar means the same graphic in ASCII
Condition code: unchanged

Translate Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
100X																
101X																
102X																
103X																
104X	BI											$\overline{.}$	$\overline{\leftarrow}$	$\overline{(}$	$\overline{+}$	
105X	$\overline{\&}$											$\overline{\$}$	$\overline{*}$	$\overline{)}$		
106X	$\overline{-}$	$\overline{/}$										$\overline{,}$	$\overline{\%}$			
107X												$\overline{\#}$	$\overline{@}$	$\overline{\blacktriangledown}$	$\overline{=}$	
108X		$\overline{\text{a}}$	$\overline{\text{b}}$	$\overline{\text{c}}$	$\overline{\text{d}}$	$\overline{\text{e}}$	$\overline{\text{f}}$	$\overline{\text{g}}$	$\overline{\text{h}}$	$\overline{\text{i}}$						
109X		$\overline{\text{j}}$	$\overline{\text{k}}$	$\overline{\text{l}}$	$\overline{\text{m}}$	$\overline{\text{n}}$	$\overline{\text{o}}$	$\overline{\text{p}}$	$\overline{\text{q}}$	$\overline{\text{r}}$						
10AX			$\overline{\text{s}}$	$\overline{\text{t}}$	$\overline{\text{u}}$	$\overline{\text{v}}$	$\overline{\text{w}}$	$\overline{\text{x}}$	$\overline{\text{y}}$	$\overline{\text{z}}$						
10BX																
10CX		$\overline{\text{A}}$	$\overline{\text{B}}$	$\overline{\text{C}}$	$\overline{\text{D}}$	$\overline{\text{E}}$	$\overline{\text{F}}$	$\overline{\text{G}}$	$\overline{\text{H}}$	$\overline{\text{I}}$						
10DX		$\overline{\text{J}}$	$\overline{\text{K}}$	$\overline{\text{L}}$	$\overline{\text{M}}$	$\overline{\text{N}}$	$\overline{\text{O}}$	$\overline{\text{P}}$	$\overline{\text{Q}}$	$\overline{\text{R}}$						
10EX			$\overline{\text{S}}$	$\overline{\text{T}}$	$\overline{\text{U}}$	$\overline{\text{V}}$	$\overline{\text{W}}$	$\overline{\text{X}}$	$\overline{\text{Y}}$	$\overline{\text{Z}}$						
10FX	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	$\overline{7}$	$\overline{8}$	$\overline{9}$						

Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS Translate	DC	L	B1 ¦ D1	D1	B2 ¦ D2	D2
SS Translate and Test	DD	L	B1 ¦ D1	D1	B2 ¦ D2	D2

Objective:

Translate - The eight-bit bytes of the first operand are used as arguments to reference the list designated by the second operand address. Each eight-bit function byte selected from the list replaces the corresponding argument in the first operand.

Translate and Test - The eight-bit bytes of the first operand are used as arguments to reference the list designated by the second operand address. Each eight-bit function byte thus selected from the list is used to determine the continuation of the operation. When the function byte is a zero, the operation proceeds by fetching and translating the next argument byte. When the function byte is nonzero, the operation is completed by inserting the related argument address in general register 1, and by inserting the function byte in general register 2. The first operand is not changed.

Translate and Test

Assume that an assembly-language statement, located at 3000 - 3016, is to be scanned for various punctuation marks. A translate and test table is constructed with zeros in all positions except where punctuation marks are assigned.

Assume:

Reg 1 (before)	00 00 00 00
Reg 2 (before)	00 00 00 00
Reg 12	00 00 30 00
Reg 15	00 00 20 00
Loc 3000-3016	UNPK PROUT (9), WORD(5)

EBCDIC for (is 4D.

The instruction is:

Op Code	L	B_1	D_1	B_2	D_2
TRT	16	C	0	F	0

Reg 1 (after)	00 00 30 0B
Reg 2 (after)	00 00 00 20

Condition code = 1; scan not completed.

Translate and Test Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
200X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
201X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
202X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
203X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
204X	0	0	0	0	0	0	0	0	0	0	0	10	15	20	25	0
205X	0	0	0	0	0	0	0	0	0	0	0	30	35	40	45	0
206X	0	0	0	0	0	0	0	0	0	0	0	50	55	0	0	0
207X	0	0	0	0	0	0	0	0	0	0	0	60	65	70	75	0
208X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
209X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20AX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20BX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20CX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20DX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20EX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20FX	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

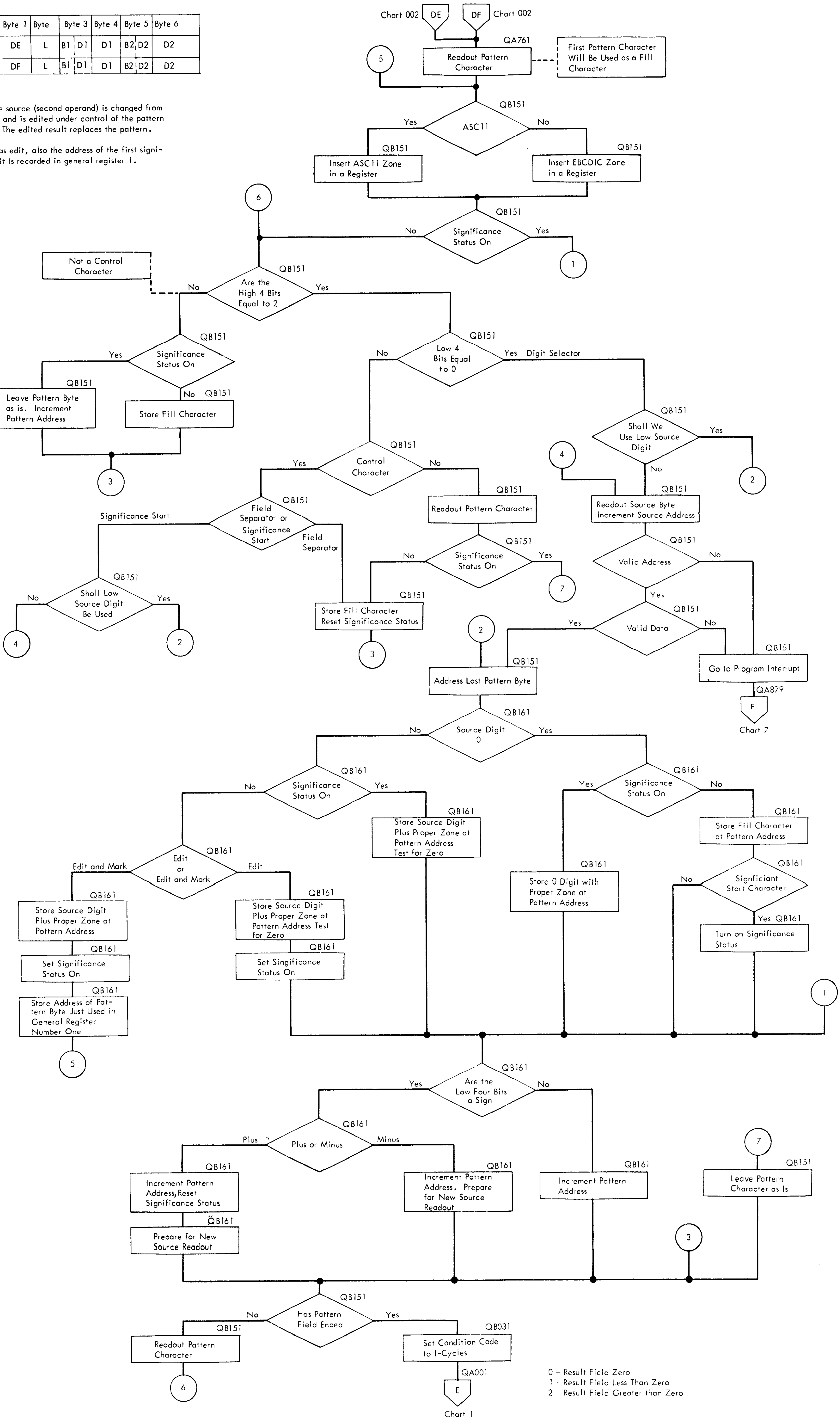
Note:
If all possible combinations of eight bits (i.e., 256 combinations) cannot appear in the statement being scanned, then a table less than 256 bytes can be used.

Op Format	Byte 1	Byte	Byte 3	Byte 4	Byte 5	Byte 6
SS Edit	DE	L	B1 D1	D1	B2 D2	D2
SS Edit and Mark	DF	L	B1 D1	D1	B2 D2	D2

Objective:

Edit – The format of the source (second operand) is changed from packed to zoned and is edited under control of the pattern (first operand). The edited result replaces the pattern.

Edit and Mark – Same as edit, also the address of the first significant result digit is recorded in general register 1.



0 – Result Field Zero
1 – Result Field Less Than Zero
2 – Result Field Greater than Zero

Edit, and Edit and Mark

The format of the source (second operand) is changed from packed to zoned and is edited under control of the pattern (first operand). The edited result replaces the pattern.

The Edit and Mark also performs the operation of storing the byte address of the first significant digit. The address is not inserted when significance is forced by the significance start character of the pattern.

pattern field	b	d	d	,	d	d	(	.	d	d	b	C	R

source field	0 2	5 7	4 2	6 +

1. Read out pattern character. Retain as fill character and store back into pattern field.

   Pattern Character b

b

2. Read out next pattern character. Decoded as digit select. Read out source byte. High digit 0 significance status is still off, store fill character in pattern field.

   Pattern Character d
   Source Byte 0 2

b	b

3. Read out next pattern character. Decoded as digit select. Low source digit significant, insert proper zone and store at pattern digit location.
   Turn on significance status.

   Pattern Character d

   Edit and Mark operation would store the address of this significant digit into General register number one.

b	b	Zone 2

4. Read out next pattern character. Decoded as non control character, leave as is because significance status on.

   Pattern character,

b	b	Zone 2	,

5. Read out next pattern character. Decoded as digit select. Read out source byte, store high source digit with proper zone at pattern digit location.

   Pattern character d
   Source Byte 5 7

b	b	Zone 2	,	Zone 5

6. Read out next pattern character. Decoded as digit select. Use low source digit of last source byte, insert proper zone and store at pattern digit location.

   Pattern Character d

b	d	Zone 2	,	Zone 5	Zone 7

7. Read out next pattern character. Decoded as significance start. Significance start status is on already, this character performs like the digit select. Read out next source byte. Store high source digit with proper zone, at pattern digit location.

   Pattern Character (
   Source Byte 4 2

b	b	Zone 2	,	Zone 5	Zone 7	Zone 4

8. Read out next pattern character. Decode as non control type. Leave as is. Read out next pattern character, digit select, store low source digit with zone at pattern digit location.

b	b	Zone 2,	,	Zone 5	Zone 7	Zone 4	.	Zone 2

9. Read out next pattern character, digit select. Read out source byte. This byte contains plus sign, turn off significance status. If sign had been minus, significance status would have been left on. Store high source digit with proper zone at pattern digit location. Store fill characters for remainder of field.

   source byte 6 +

Edited field replacing pattern field	b	b	Zone 2	,	Zone 5	Zone 7	Zone 4	.	Zone 2	Zone 6	b	b	b

Set condition register

0 = Result field is source
1 = Result field is less than zero
2 = Result field is greater than zero

Pack, Unpack, and Move with Offset.
None of These Instructions Check for
Valid Digits or Signs.

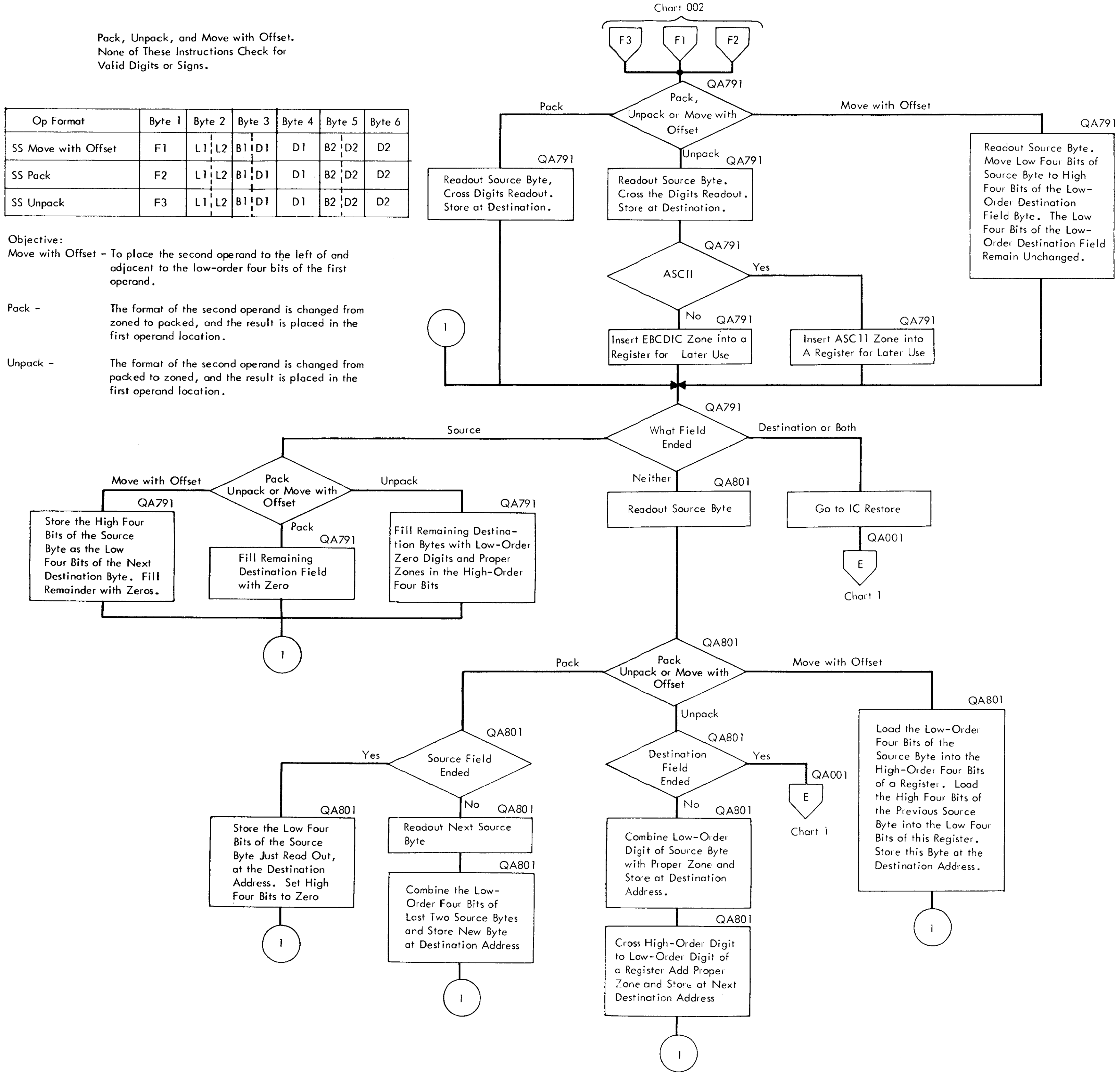
Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS Move with Offset	F1	L1 ¦ L2	B1 ¦ D1	D1	B2 ¦ D2	D2
SS Pack	F2	L1 ¦ L2	B1 ¦ D1	D1	B2 ¦ D2	D2
SS Unpack	F3	L1 ¦ L2	B1 ¦ D1	D1	B2 ¦ D2	D2

Objective:

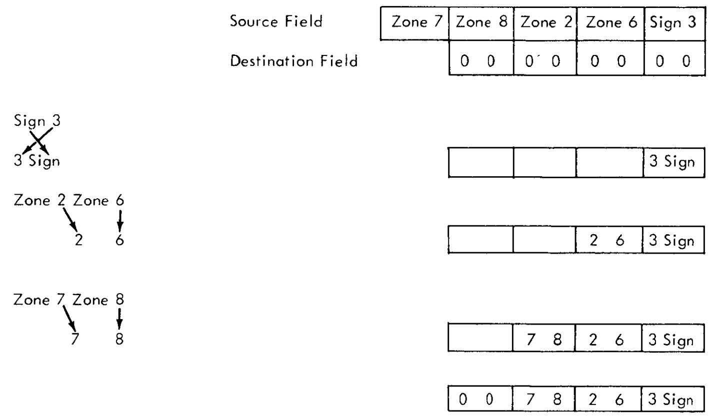
Move with Offset - To place the second operand to the left of and adjacent to the low-order four bits of the first operand.

Pack - The format of the second operand is changed from zoned to packed, and the result is placed in the first operand location.

Unpack - The format of the second operand is changed from packed to zoned, and the result is placed in the first operand location.



Pack Instruction



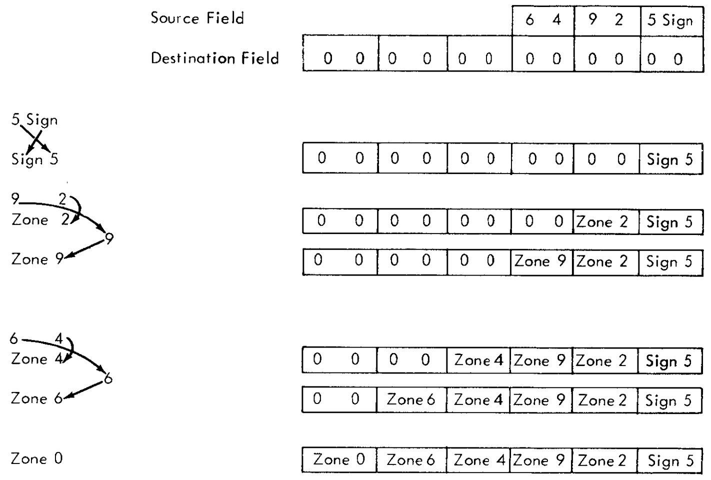
① Readout Low Source Byte. Cross Digits. Store at Destination Low Address.

② Readout Next Two Source Bytes. Combine Low Order Digits. Store at Next Higher Destination Address.

③ Readout Next Two Source Bytes. Combine Low Order Digits. Store at Next Higher Destination Address.

④ Source Field Has Ended. Supply Zero's for Destination Field Until it Ends.

Unpack Instruction



① Readout Low-Order Source Byte. Cross and Store at Low Order Destination Address.

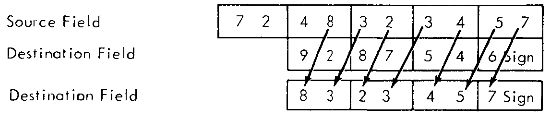
② Readout Next Source Byte. Combine Low-Order Digit with Proper Zone. Store at Destination. Cross High-Order Digit into Low Order of a Register. Read Register Out. Insert Proper Zone Store at Destination.

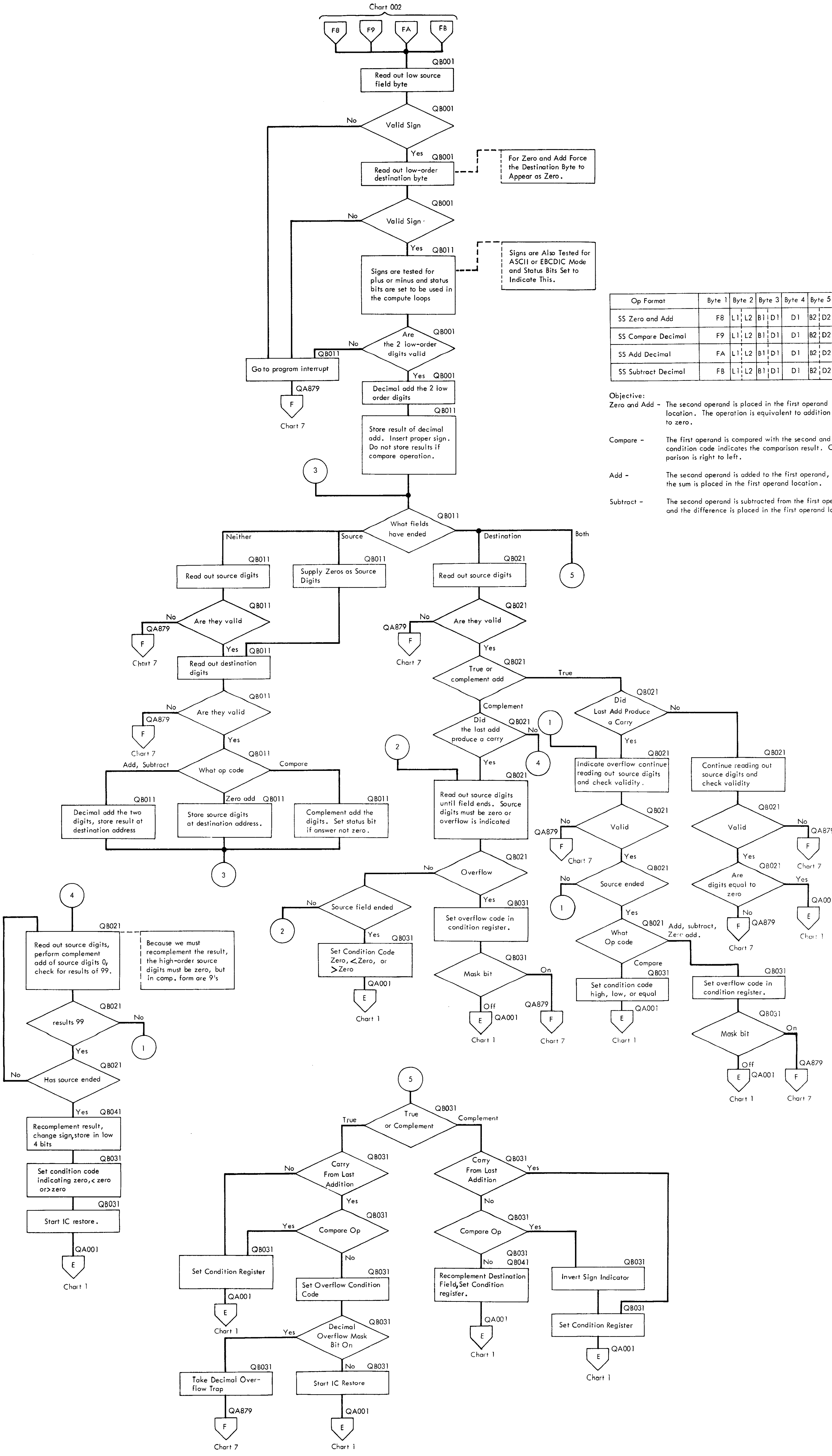
③ Repeat Step 2.

④ Source Ended, Supply Zero Digits with Proper Zones until Destination Ends.

Move with Offset

The Source Field is Moved to the Destination Field to the Left of the Low-Order Four Bits of the Destination. If the Destination Field Ends Before the Source Field, the Remaining Source Digits are Ignored.





Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS Zero and Add	F8	L1 L2	B1 D1	D1	B2 D2	D2
SS Compare Decimal	F9	L1 L2	B1 D1	D1	B2 D2	D2
SS Add Decimal	FA	L1 L2	B1 D1	D1	B2 D2	D2
SS Subtract Decimal	FB	L1 L2	B1 D1	D1	B2 D2	D2

Objective:

Zero and Add – The second operand is placed in the first operand location. The operation is equivalent to addition to zero.

Compare – The first operand is compared with the second and the condition code indicates the comparison result. Comparison is right to left.

Add – The second operand is added to the first operand, and the sum is placed in the first operand location.

Subtract – The second operand is subtracted from the first operand, and the difference is placed in the first operand location.

CLF 020 SS Decimal-Add, Subtract, Compare, and Zero Add

DECIMAL ADD, SUBTRACT EXAMPLE

Sign analysis is done first and will indicate if the operands will be true or complement added.

Eight conditions may occur as shown in the following table:

Operation	First Operand Sign	Second Operand Sign	True or Complement
Add	Plus	Plus	True
Add	Minus	Plus	Complement
Add	Minus	Minus	True
Add	Plus	Minus	Complement
Subtract	Plus	Plus	Complement
Subtract	Minus	Plus	True
Subtract	Minus	Minus	Complement
Subtract	Plus	Minus	True

After sign analysis, the operation for both add and subtract are the same.

Decimal Add

```
               1st Operand = Destination field            0 4 6 +
               2nd Operand = Source field               0 0 1 3 4 -

1. Read out low source byte.                                               4 -
2. Read out low destination byte.                                          6 +
                                                  6
3. Decimal add the digits.                       @5  9's complement of 4
     Complement add indicated.                    1  Insert carry
                                      Carry  ⓒ←2  Converts 9's complement to 10's complement

     Insert Destination Sign.                     Intermediate sum = 2 +
                                                  1  Carry from previous cycle
4. Read out source digits.                       04
   Read out destination digits.                 @86  9's complement of destination byte.
     Complement add.                  NC         91

5. Destination field ended.                       Intermediate sum  9 1 2 +
   There was no carry from high-order
   significant addition, indicating
   recomplementing will be necessary.

6. Read out source digits.                       00
     Complement add to zero.                    @99
     9's must result or overflow                 99
     will be indicated.

7. Source field ended, recomplement               Intermediate sum  9 9 9 1 2 +
   answer and change sign.                        Result stored in destination field  0 8 8 -
```

Cases when the source field is longer than the destination and a complement addition is being performed.

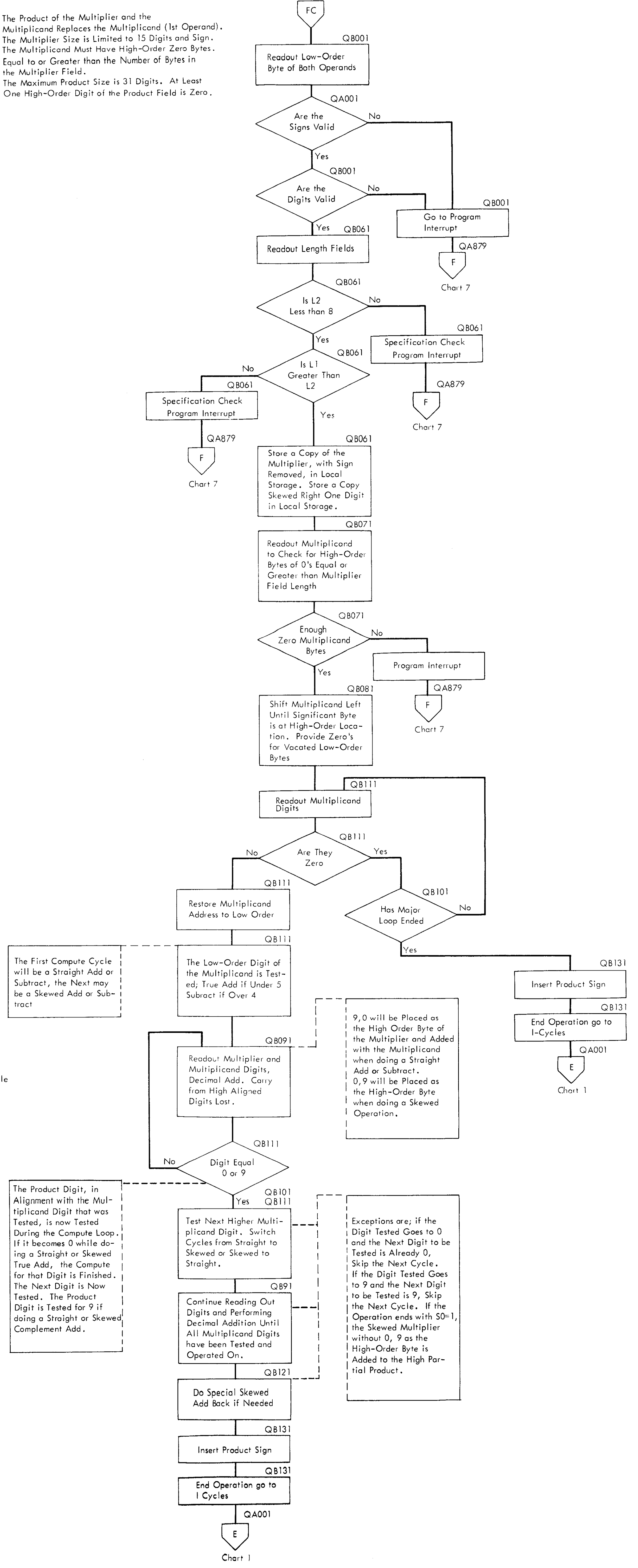
1. Destination ends and no carry results from last addition; all further adds must produce 9's or an overflow results.
2. Destination ends, with a carry resulting from last addition:
   The answer produced so far is zero and all further adds must produce zeros.
   The answer produced so far is not zero; examine carry from the addition following the one in which the destination ended. If there was a carry, the result just produced must be zero; if there was no carry, the result just produced must be 9's. In either case, all further cycles must produce the same result as the first.

When doing a true add, all cycles after the destination field ends must produce zeros.
Overflow when doing a complement add causes a recomplement, then a branch to overflow case when setting condition register.
Overflow when doing a true add forces overflow when setting condition register.

Condition Code
0 = Result is zero
1 = Result less than zero
2 = Result greater than zero
3 = Overflow

SS Multiply Decimal	FC	L1	L2	B1	D1	D1	B2	D2	D2

The Product of the Multiplier and the Multiplicand Replaces the Multiplicand (1st Operand).
The Multiplier Size is Limited to 15 Digits and Sign.
The Multiplicand Must Have High-Order Zero Bytes. Equal to or Greater than the Number of Bytes in the Multiplier Field.
The Maximum Product Size is 31 Digits. At Least One High-Order Digit of the Product Field is Zero.



Refer to multiply example for multiply loop.

CLF 022 SS Decimal Multiply

Decimal Multiply

The product of the multiplier (second operand) and the multiplicand (first operand) replaces the multiplicand. The multiplicand must have high-order zero digits for at least a field size equal to the multiplier field.

```
multiplicand       0 0 0 0 7 2 4 +
multiplier                 3 4 6 +
```

1. Make 2 copies of the multiplier, with sign stripped out, into local storage. One copy straight and one copy skewed right.

```
straight copy of multiplier    3 4 6 0
skewed copy of multiplier      0 3 4 6
```

2. Test for enought zeros in multiplicand field.

3. Move the most significant byte and following byes of the multiplicand to the high-order byte of the multiplicand. Supply zeros for vacated bytes, sign is stripped out.

```
0 0 0 0 7 2 4 +

7 2 4 0 0 0 0 0
```

4. Start compute loop. First cycle will be a straight cycle. The first low significant digit of the multiplicand is tested. If over 4, a straight subtract will be done. If under 5, a straight true add will be done.

```
                                        Test digit to determine operation straight add
                                  7 2 4 0 0 0 0 0
The 90 byte supplied by       +         9 0 3 4 6 0
microprogram.                     7 2 3 0 3 4 6 0   this digit tested for 0, keep
The high 2 digits do not      +         9 0 3 4 6 0   adding until 0 reached
participate this cycle and        7 2 2 0 6 9 2 0
are not affected by carries.  +         9 0 3 4 6 0
                                  7 2 1 1 0 3 8 0
                              +         9 0 3 4 6 0
                                  7 2 0 1 3 8 4 0
                                                    Reached 0, switch cycles.
```

```
5. Cycle now will be a skewed cycle, test digit to        Test digit, under 5, operation add.
   determine add or subtract                     7 2 0 1 3 8  4 0
        0 9  supplied by                       + 0 9 0 3 4 6
        microprogram                             7 1 0 4 8 4 4 0   digit tested for 0, keep
                                               + 0 9 0 3 4 6       adding.
                                                 7 0 0 8 3 0 4 0   digit 0, switch cycles.
```

```
6. Cycle now will be a straight cycle, test digit to                  Test digit over 4, subtract cycles to
   determine add or subtract                     7 0 0 8 3 0 4 0   be taken.  Set subtract status.
                                               - 9 0 3 4 6 0
                                                 7 9 7 3 7 0 4 0
                                                                   Test this digit for 9, continue
                                               - 9 0 3 4 6 0       subtracting.
                                                 8 9 3 9 1 0 4 0
                                               - 9 0 3 4 6 0
                                                 9 9 0 4 5 0 4 0
                                                                   digit reached 9, stop operation.
```

```
7. Operation stopped with subtract status on; the
   skewed multiplier must be added to partial     9 9 0 4 5 0 4 0
   product to obtain correct product.             0 3 4 6
                                                  0 2 5 0 5 0 4 0
```

```
8. Insert sign and end operation                 | 0 2 | 5 0 | 5 0 | 4 + |  sign inserted, final produtt.
   Product located in first operand location.
```

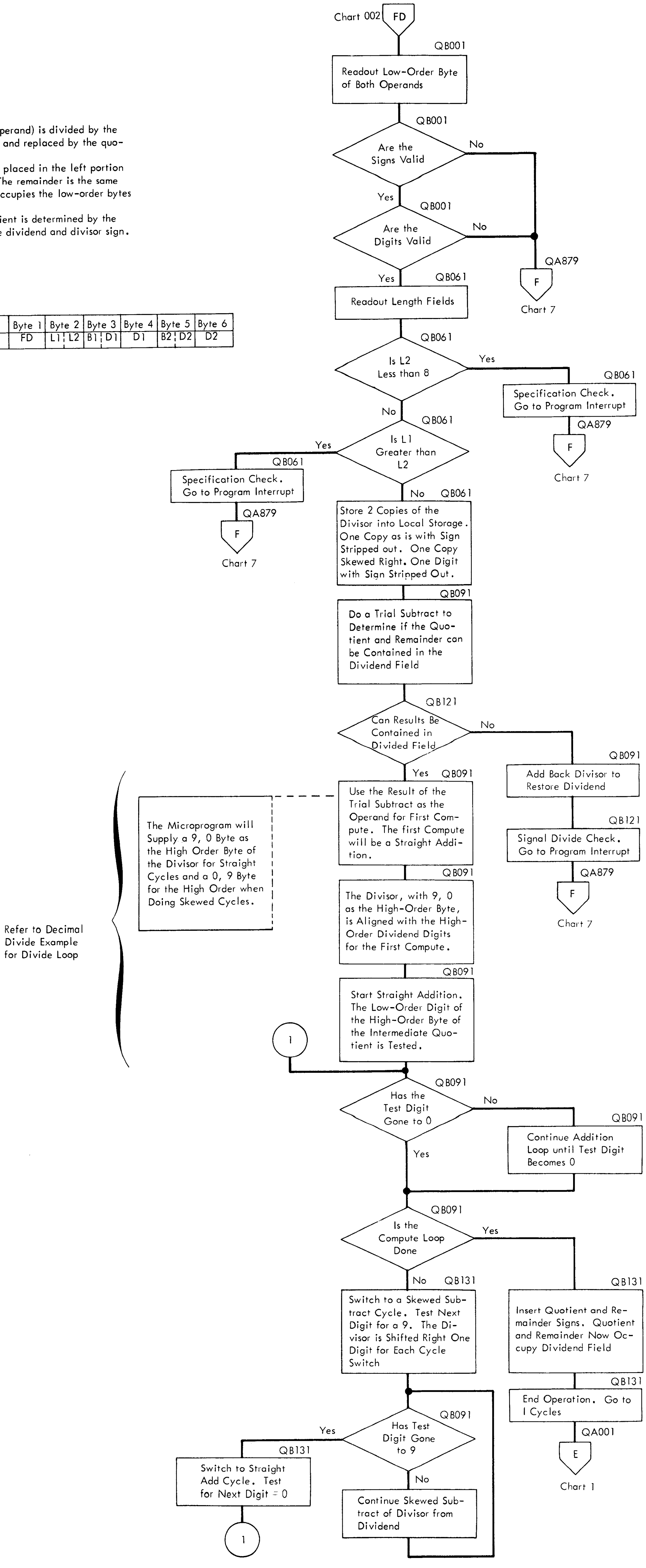
Decimal Divide

The dividend (first operand) is divided by the divisor (second operand) and replaced by the quotient and remainder.

The quotient field is placed in the left portion of the dividend field. The remainder is the same size as the divisor and occupies the low-order bytes of the dividend field.

The sign of the quotient is determined by the rules of algebra from the dividend and divisor sign.

Op Format	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
SS Divide Decimal	FD	L1 L2	B1 D1	D1	B2 D2	D2



Decimal Divide

The dividend (first operand) is divded by the divisor (second operand) and replaced by the quotient and remainder.

```
Dividend equal      0 1 2 3 5 6 8 +
Divisor equals      8 5 2 +
```

1. Copy divisor into local storage, with sign stripped out, in straight form and skewed right one digit.

```
8 5 2 0 straight
0 8 5 2 skewed
```

2. Setup data for trial subtract using skewed divisor. A carry out of the high-order position on the trial subtract would indicate a divide check. If divide check occurred, the skewed divisor is added back to restore dividend and a divide check taken.

```
Dividend                 0 1 2 3 5 6 8 +
Skewed divisor         - 0 8 5 2
Operand in storage       9 2 7 1 5 6 8 0
```

3. Add straight divisor to operand in working storage until indicated digit goes to zero.

```
Operand supplied by      9 2 7 1 5 6 8 0
Microprogram           + 9 0 8 5 2 0
Not zero,add again       8 3 5 6 7 6 8 0
                       + 9 0 8 5 2 0
                         7 4 4 1 9 6 8 0
                       + 9 0 8 5 2 0
                         6 5 2 7 1 6 8 0
                       + 9 0 8 5 2 0
                         5 6 1 2 3 6 8 0
                       + 9 0 8 5 2 0
                         4 6 9 7 5 6 8 0
                       + 9 0 8 5 2 0
                         3 7 8 2 7 6 8 0
                       + 9 0 8 5 2 0
                         2 8 6 7 9 6 8 0
                       + 9 0 8 5 2 0
                         1 9 5 3 1 6 8 0
                       + 9 0 8 5 2 0
Stop straight add,       1 0 3 8 3 6 8 0
digit now zero
```

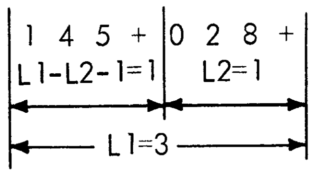
4. Subtract skewed divisor from result until indicated digit goes to nine.

```
This digit does not
participate.             1 0 3 8 3 6 8 0
This digit supplied    -   9 0 8 5 2
by Microprogram to       1 1 2 9 8 4 8 0
skewed divisor.        -     9 0 8 5 2
                         1 2 2 1 3 2 8 0
This digit not 9,      -     9 0 8 5 2
subtract again.          1 3 1 2 8 0 8 0
                       -     9 0 8 5 2
                         1 4 0 4 2 8 8 0
                       -     9 0 8 5 2
                         1 4 9 5 7 6 8 0

Stop skewed subtract
digit = 9.
```

5. Add straight divisor to result will indicated digit goes to zero.

```
These digits do not      1 4 9 5 7 6 8 0
participate.           +     9 0 8 5 2 0
This digit not           1 4 8 6 6 2 0 0
zero, add again.       +     9 0 8 5 2 0
                         1 4 7 7 4 7 2 0
                       +     9 0 8 5 2 0
                         1 4 6 8 3 2 4 0
                       +     9 0 8 5 2 0
                         1 4 5 9 1 7 6 0
                       +     9 0 8 5 2 0
Stop, this digit         1 4 5 0 0 2 8 0
now zero.                Quotient  Remainder
```

6. Insert dividend sign into low-order 4 bits.
Insert quotient sign into low-order quotient 4 bits.
Quotient and remainder have now replaced dividend.

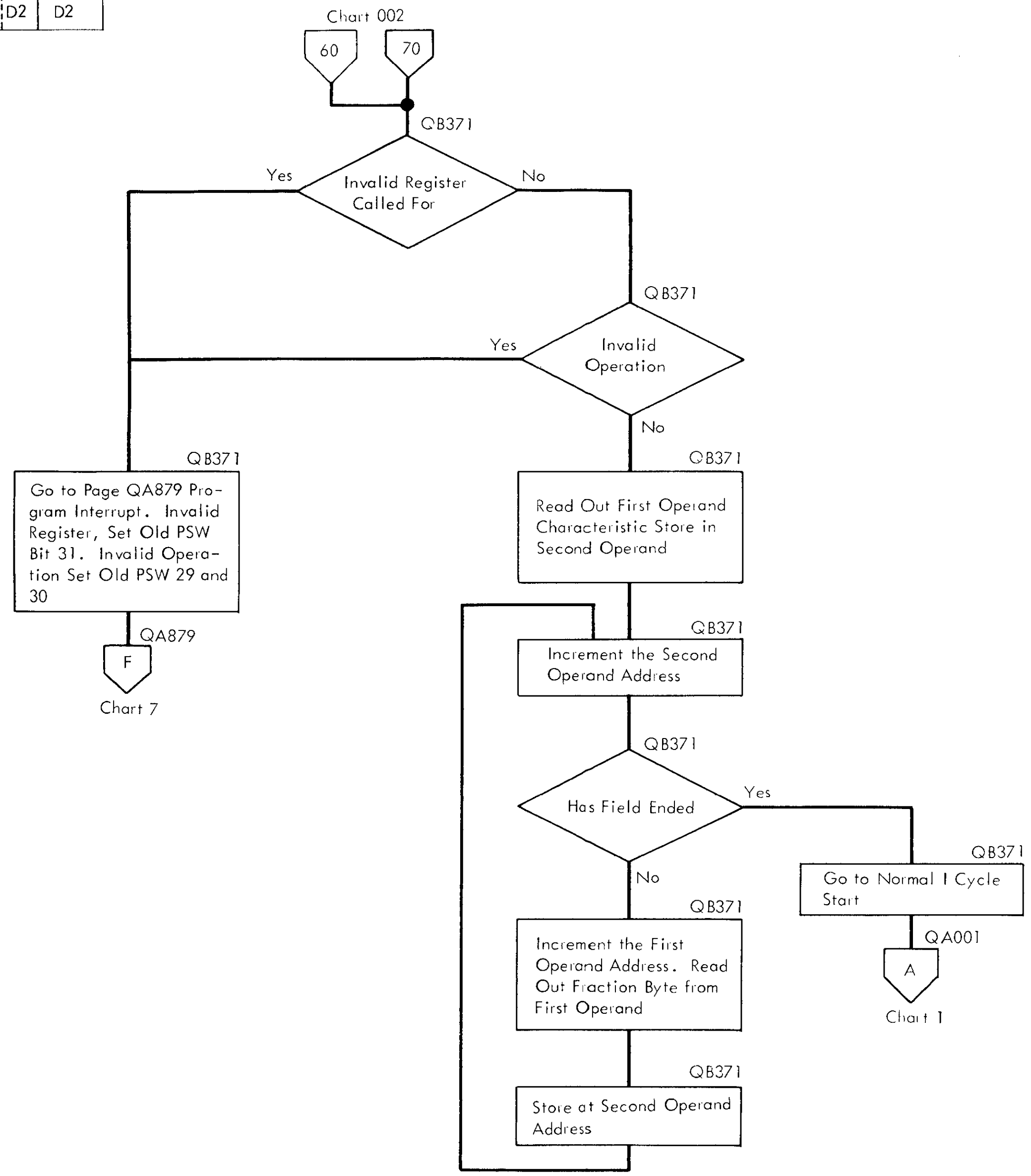


OP Format	Byte 1	Byte 2	Byte 3	Byte 4
RX FP Store Double	60	R1 X2	B2 D2	D2
RX FP Store Single	70	R1 X2	B2 D2	D2

Floating Point Store RX Format
Single and Double Precision.

Single Precision; the Low Order Half of the First Operand Register is Ignored.

Mnemoics
STD
STE

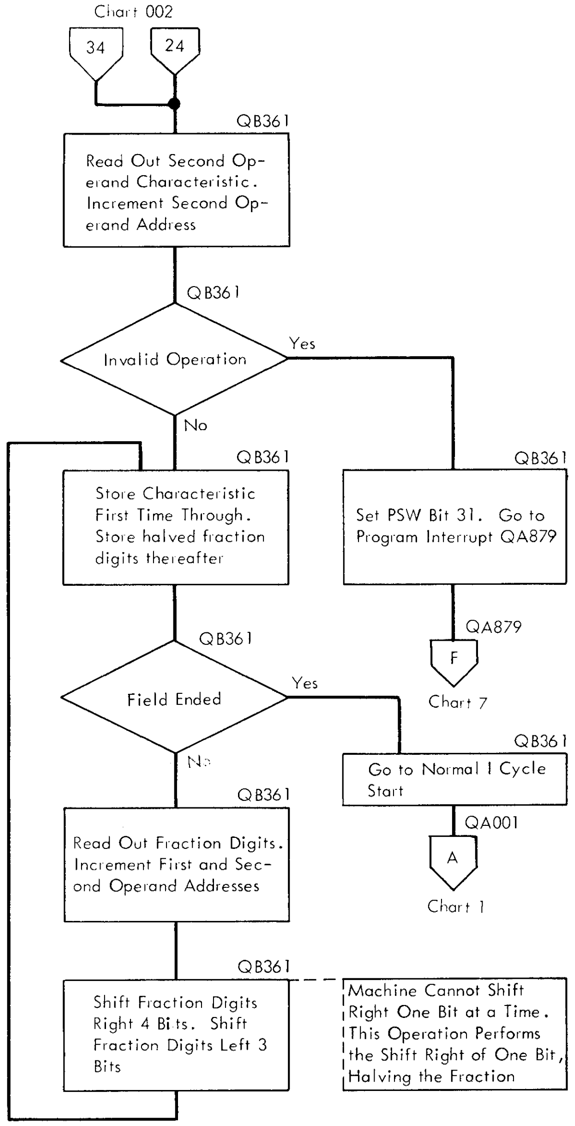


Op Format	Byte 1	Byte 2
RR FP Halve, Double	24	R1 R2
RR FP Halve, Single	34	R1 R2

Floating Point Halve RR Format
Single and Double Precision.
Single Precision, the Low Order Half of the Result Register Remains Unchanged.

The Second Operand is Divided by 2 and the Quotient is Placed in the First Operand Location. Second Operand Sign and Characteristic is Stored without Change.

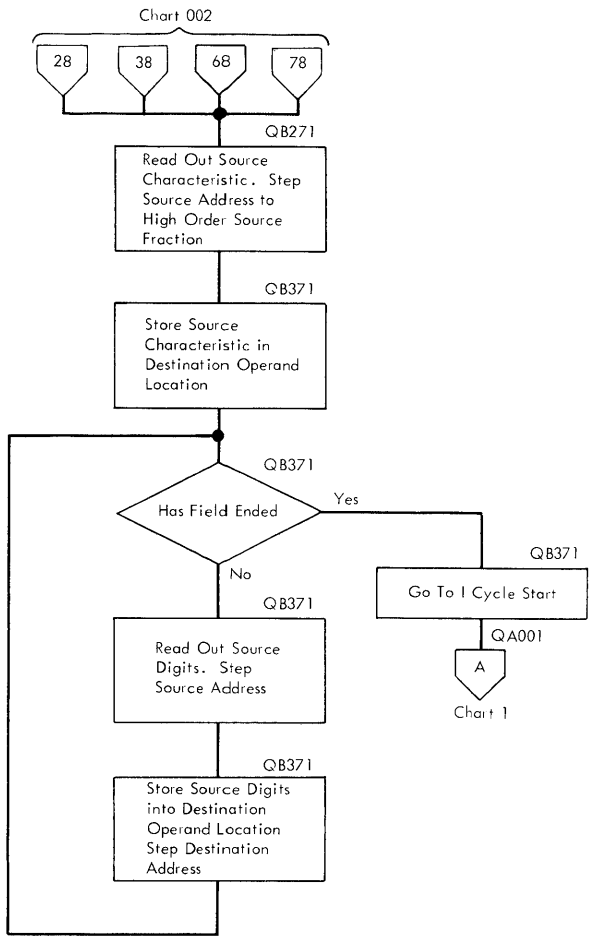
Mnemonics
HDR
HER



Op Format	Byte 1	Byte 2	Byte 3	Byte 4
RR FP Load, Double	28	R1 R2		
RR FP Load, Single	38	R1 R2		
RX FP Load, Double	68	R1 X2	B2 D2	D2
RX FP Load, Single	78	R1 X2	B2 D2	D2

Load RR and RX Formats
Single or Double Precision

Second Operand is Placed in First Operand Location. In Single Precision the Low Half of the Destination Operand Remains Unchanged.



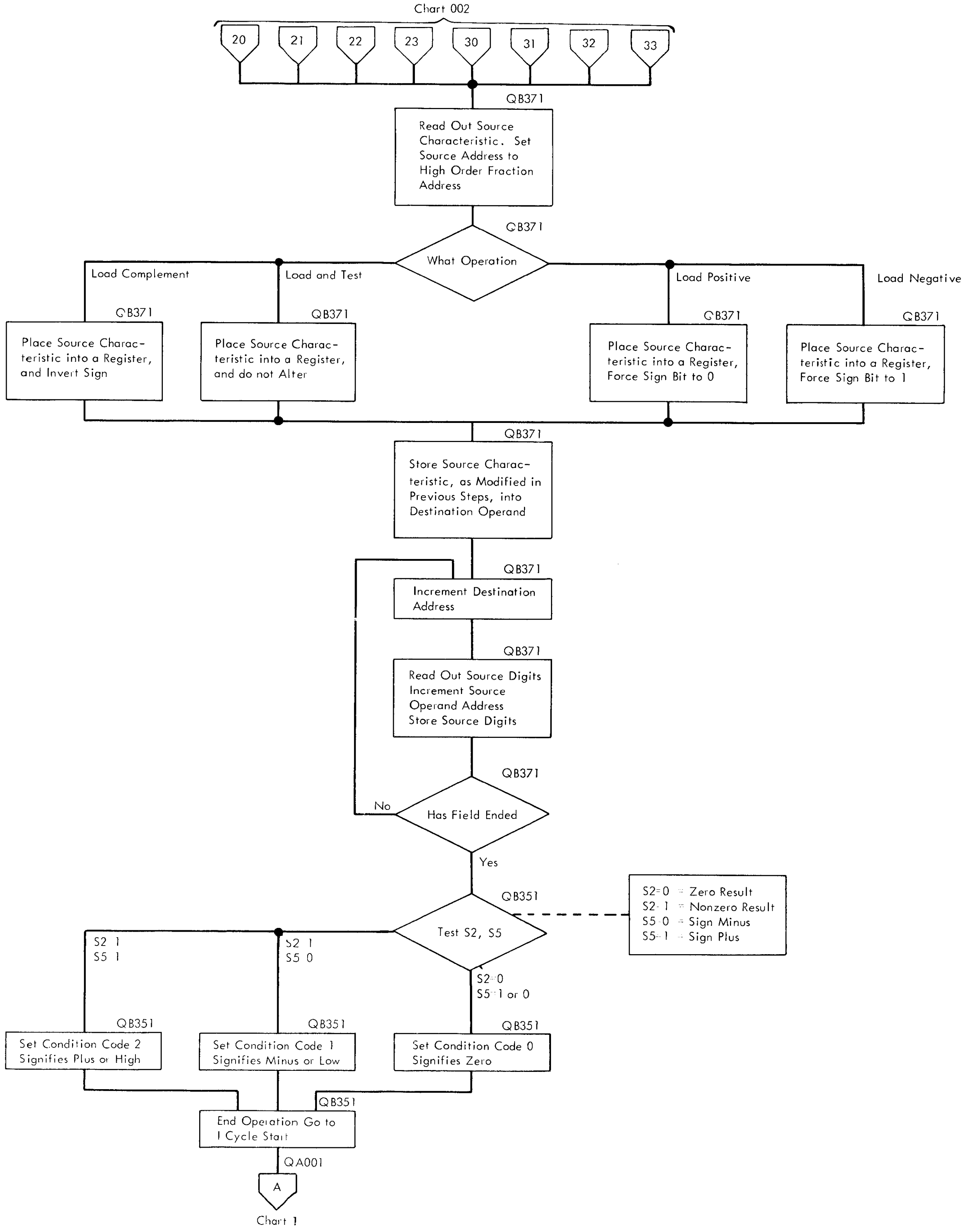
Load and Test; the Second Operand is Placed in the First Operand Location. Condition Codes are Set.

Load Complement; the Second Operand is Placed in the First Operand Location with the Sign Changed.

Load Positive; the Second Operand is Placed in the First Operand Location with the Sign Forced Positive.

Load Negative; the Second Operand is Placed in the First Operand Location with the Sign Forced Minus.

Op Format	Byte 1	Byte 2
RR FP Load Positive	20	R1 R2
RR FP Load Positive	30	R1 R2
RR FP Load Negative	21	R1 R2
RR FP Load Negative	31	R1 R2
RR FP Load and Test	22	R1 R2
RR FP Load and Test	32	R1 R2
RR FP Load Complement	23	R1 R2
RR FP Load Complement	33	R1 R2



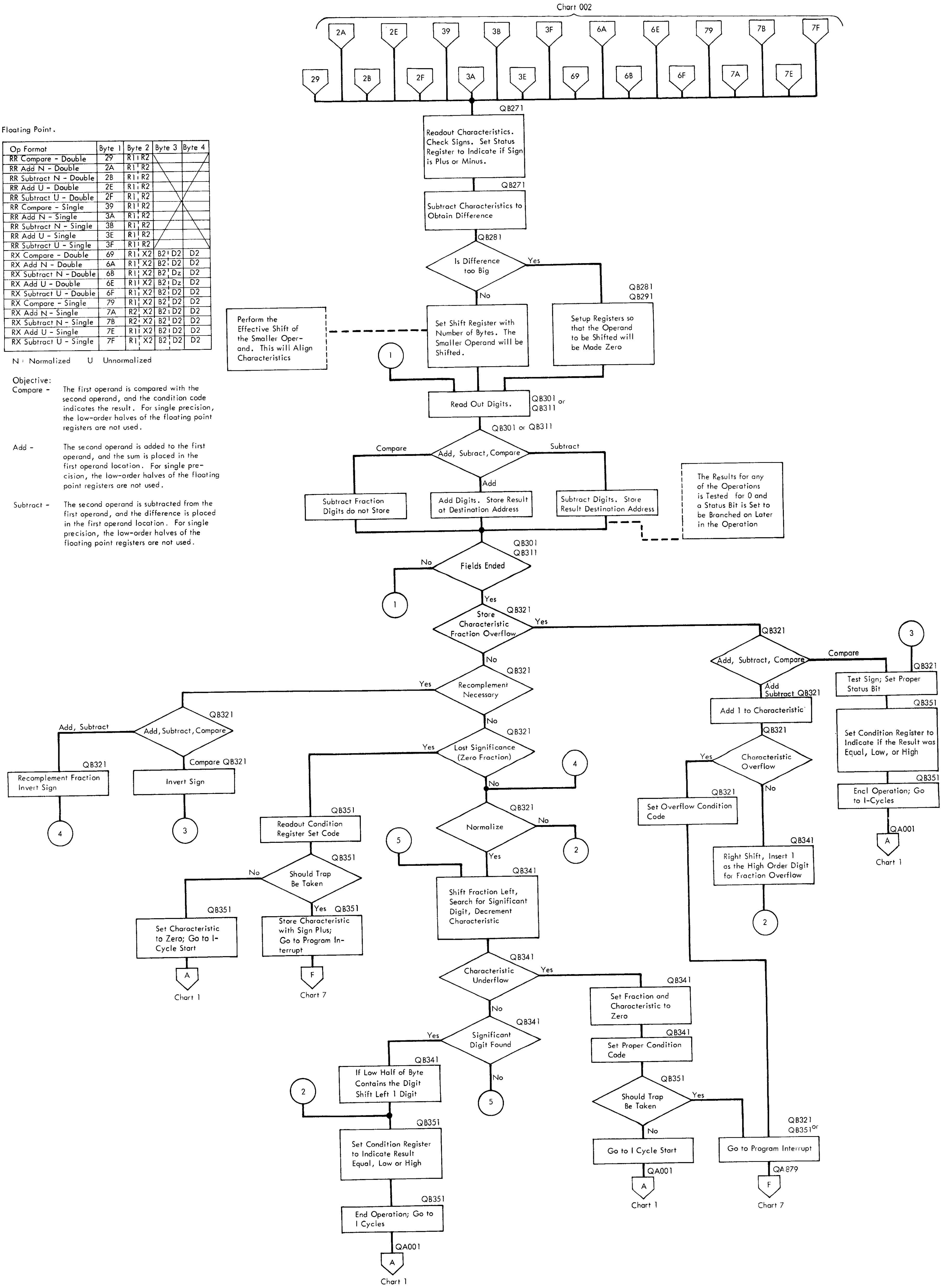
Floating Point.

Op Format	Byte 1	Byte 2	Byte 3	Byte 4
RR Compare - Double	29	R1 R2		
RR Add N - Double	2A	R1 R2		
RR Subtract N - Double	2B	R1 R2		
RR Add U - Double	2E	R1 R2		
RR Subtract U - Double	2F	R1 R2		
RR Compare - Single	39	R1 R2		
RR Add N - Single	3A	R1 R2		
RR Subtract N - Single	3B	R1 R2		
RR Add U - Single	3E	R1 R2		
RR Subtract U - Single	3F	R1 R2		
RX Compare - Double	69	R1 X2	B2 D2	D2
RX Add N - Double	6A	R1 X2	B2 D2	D2
RX Subtract N - Double	6B	R1 X2	B2 Dz	D2
RX Add U - Double	6E	R1 X2	B2 Dz	D2
RX Subtract U - Double	6F	R1 X2	B2 D2	D2
RX Compare - Single	79	R1 X2	B2 D2	D2
RX Add N - Single	7A	R2 X2	B2 D2	D2
RX Subtract N - Single	7B	R2 X2	B2 D2	D2
RX Add U - Single	7E	R1 X2	B2 D2	D2
RX Subtract U - Single	7F	R1 X2	B2 D2	D2

N : Normalized U Unnormalized

Objective:

Compare -	The first operand is compared with the second operand, and the condition code indicates the result. For single precision, the low-order halves of the floating point registers are not used.
Add -	The second operand is added to the first operand, and the sum is placed in the first operand location. For single precision, the low-order halves of the floating point registers are not used.
Subtract -	The second operand is subtracted from the first operand, and the difference is placed in the first operand location. For single precision, the low-order halves of the floating point registers are not used.



CLF 028 RR-RX Floating Point - Add, Subtract and Compare

Floating Point Normalized Add

The second operand is added to the first operand, and the normalized sum is placed in the first operand location.

Addition consists of characteristic comparison and fraction addition.

QB271 Read out characteristics, shift them left one bit to determine signs. Sign bit is the high-order bit of the characteristic.

Subtract characteristics to determine characterfic difference.

QB281 The difference represents double the number of hexadecimal digits that the smaller operand will effectively be shifted right. Shifting is actually accomplished by changing the address of the first fraction byte to be read out. This aligns the low-order fraction bytes for addition.

QB291 Determine the new address of the effective low-order byte, store the high-order digit of the next lower byte to be used as a guard digit.

QB301 or QB311 Read out low-order fraction bytes and add. Store result in low-order 1st operand location. Continue addition until fields end. When smaller operand field ends, 0's are supplied until larger ends.

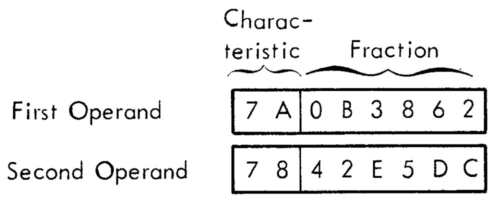
Partial result characteristic will be characteristic of the larger operand.

QB341 High-order partial result digit =0, normalization required. Shift left until high-order digit is significant, add in guard digit as low-order result digit.

QB351 Condition register is set to indicate:

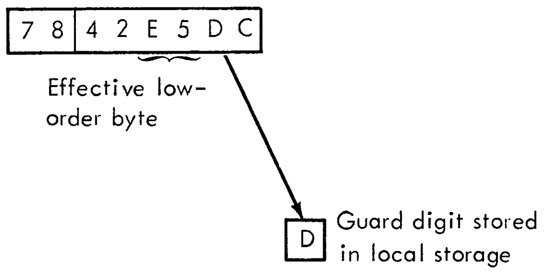
Code

0 = Result fraction is zero
1 = Result fraction less than zero
2 = Result fraction greater than zero
3 = Result exponent overflows



First Operand	7 A	shift left	F 4
Second Operand	7 8	shift left	F 0
Signs Plus			

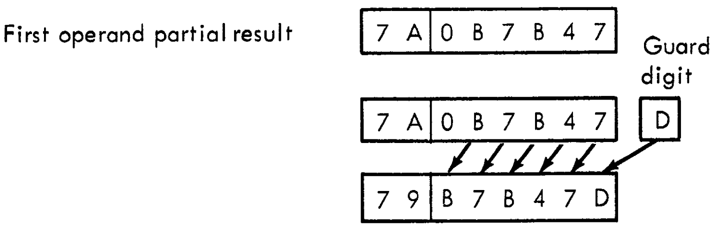
$$\begin{array}{lr} & F\ 4 \\ & \underline{F\ 0} \\ \text{Characteristic difference} & 4 \end{array}$$



$$\begin{array}{lr} \text{First Operand} & 6\ 2 \\ \text{Second Operand} & \underline{E\ 5} \\ \text{Carry} & 4\ 7 \end{array}$$

$$\begin{array}{lr} \text{First Operand} & 3\ 8 \\ \text{Second Operand} & \underline{4\ 2} \\ & 7\ B \end{array}$$

$$\begin{array}{lr} \text{First Operand} & 0\ B \\ \text{Zeros Supplied} & \underline{0\ 0} \\ & 0\ B \end{array}$$



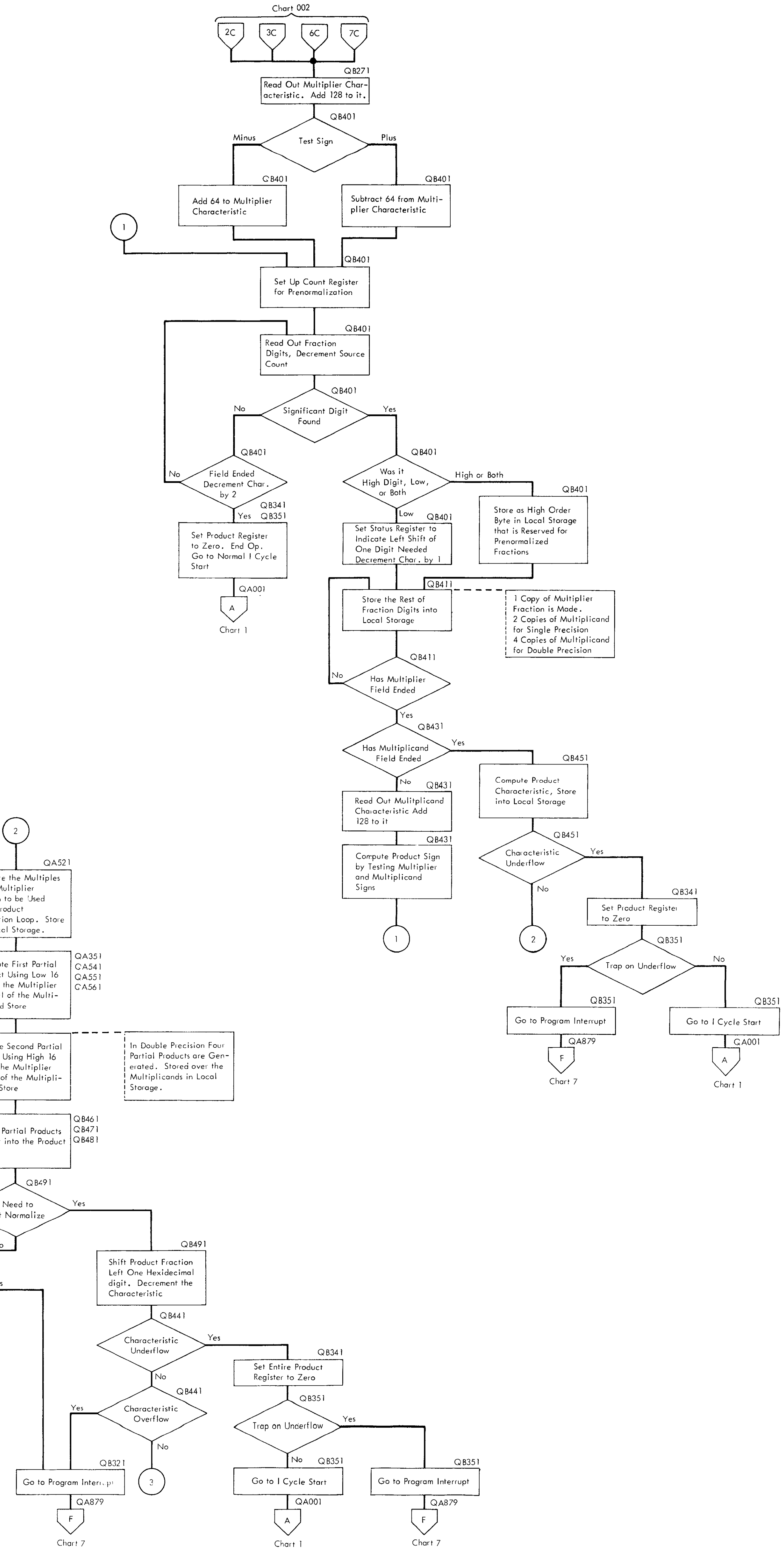
Characteristic is decremented 1 for every hexadecimal digit shifted left.

Op Format
RR Double
RR Single
RX Double
RX Single

Byte 1	Byte 2	Byte 3	Byte 4
2C	R1 R2		
3C	R1 R2		
6C	R1 X2	B2 D2	D2
7C	R1 X2	B2 D2	D2

Floating Point Multiply RR, RX Formats
Mnemonics
MER
ME
MDR
MD

The Normalized Product of the Multiplier (Second Operand) and Multiplicand (First Operand) Replaces the Multiplicand



CLF 030 RR-RX Floating Point - Multiply

Floating Point Multiply, Single Precision

The two 24-bit fractions are multiplied together and form a 56-bit product. The product replaces the first operand (multiplicand).

The product characteristic is the sum of the two characteristics minus 64.

	Characteristic	Fraction
R1 (multiplicand)	2 4	7 2 A 9 B 6
R2 (multiplier)	3 7	2 4 1 3 1 2

QB401 1. Prenormalized the multiplier fraction into the work area of storage.

QB411 2. Make two prenormalized copies of the multiplicand into the work area of storage.

QB431 3. Compute product sign, store it until product is computed.

QB451 4. Compute product characteristic, test for underflow. Store into local storage until product is computed.

```
                          2 4  =  multiplicand characteristic
                        + 3 7  =  multiplier characteristic
                          5 B
Minus binary 64 equals    1 B  =  product characteristic
```

QB451 5. Load 16 bits of the multiplier into hardware registers. The first time through this preparation step, only 8 multiplier bits are loaded. Zeros are loaded into the other hardware register. Second time through, the next 16 multiplier bits are loaded into hardware registers.

```
Low 8 bits of multiplier into hardware register.   1 2
Zeros loaded into another register.                      0 0
                 1st 16 bits of multiplier         1 2  0 0
```

QA521 6. Generate multiples of the multiplier to be used in the compute loop. Store multiples into working storage and hardware registers.

Two separate multiply loops will be done, each using 16 bits of the multiplier and all bits of the multiplicand.

Each multiply loop will produce a 40-bit partial product, the partial products are stored over the copies of the multiplicand.

```
                        For 1st multiply
                        1X = 1 2 0 0
For 2nd multiply        2X = 2 4 0 0
1X = 2 4 1 3
2X = 4 8 2 6
```

7. Decode and operate on the multiplicand, one digit at a time. Decoding the multiplicand digits sets up branching conditions in the microprogram, which will cause the correct combinations of multiples to be added or subtracted to form a partial product.

Example: 1st multiplicand digit = 6

6 decodes as 2X, 2X, 2X

this means, the multiplier times 2 is added 3 times into the product accumulator.

```
     2X = 2 4 0 0
   + 2X = 2 4 0 0
          4 8 0 0
   + 2X = 2 4 0 0
          6 C 0 0
```

8. After a digit is decoded and operated on, the low digit in the product accumulator is stored as a digit in the partial product location. Each multiplicand digit decoded, and operated on, produces one partial product digit.

Store low digit at low-digit location of 1st partial product. Shift digits right one digit in product accumulator and await next decode

```
decode         product accumulator
                 0 6 C 0
```

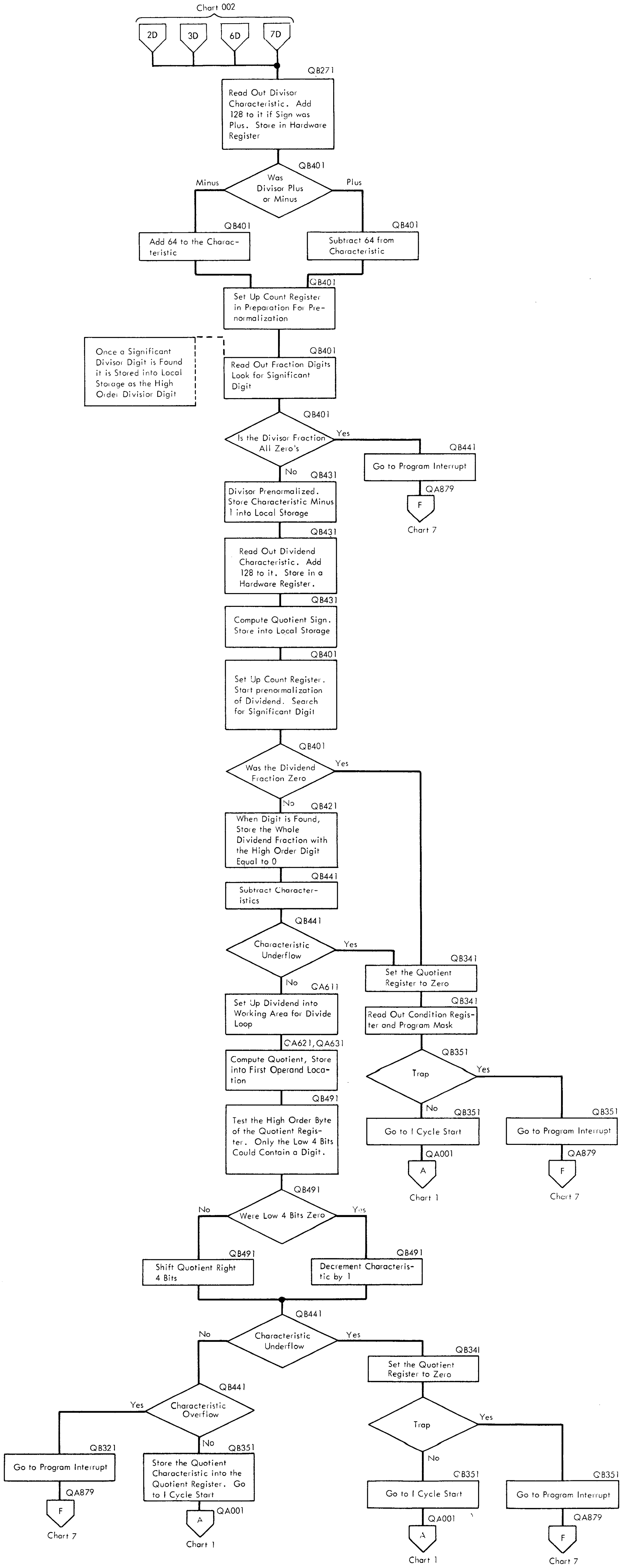
9. Multiply loop continues until two partial products are produced.
Align the two partial products and add to produce a 56-bit product. Normalize if needed. Insert characteristic and sign.

```
          1st partial product              0 8 0 F E E C C 0 0
          2nd partial product      1 0 2 8 6 0 3 0 8 2
          product                  1 0 2 8 6 8 4 0 7 0 C C 0 0

Insert characteristic     1  B     1 0 2 8 6 8 4 0 7 0 C C 0 0
Insert sign              +1  B     1 0 2 8 6 8 4 0 7 0 C C 0 0
```

Op Format	Byte 1	Byte 2	Byte 3	Byte 4
RR Divide, Double	2D	R1 R2		
RR Divide, Single	3D	R1 R2		
RX Divide, Double	6D	R1 X2	B2 D2	D2
RX Divide, Single	7D	R1 X2	B2 D2	D2

Floating Point Divide RR, RX Formats
Single and Double Precision
Mnemonics
DER
DE
DDR
DD

The First Operand is Divided by the Second Operand, the Quotient Replaces the First Operand



CLF 032 RR-RX Floating Point - Divide

Floating Point Divide

All numbers shown are in hexadecimal

		Characteristic	Fraction
First Operand is the	Dividend	2 5	9 6 7 2 E A
Second Operand is the	Divisor	1 9	7 1 3 4 B 5

The divisor in this example is a normalized number and will not be prenormalized. The dividend is prenormalized with the high-order digit equal to zero. In this example, the dividend will be shifted right one digit to accomplish this. The low-order digit will be shifted out into the hardware register in which the quotient will be formed. The first time through the divide loop, only 4 quotient digits will be computed and stored. The next divide loops will produce 8 quotient digits until the operation is completed.

The characteristics are carried in excess 64 notation at all times.

Page No.

QB271 Divisor characteristic is read out, a bit is added to the high-order position (sign). The characteristic will now be without sign if it was minus, or effectively have a value of 128 higher if it was plus.

Divisor Characteristic

$$\begin{array}{r} 1\ 9 \\ +8\ 0 \\ \hline 9\ 9 \end{array}$$

Sign was plus, characteristic is now expressed in excess 192. If sign were minus it would still be expressed in excess 64.

QB401 Divisor characteristic has 64 added to it or subtracted from it, depending on the sign position. If the sign had been plus, 64 will be subtracted from it; if it was minus, 64 is added. The characteristic now represents the numeric value of the exponent +128 instead of +64 as it usually does.

Divisor Characteristic

$$\begin{array}{r} 9\ 9 \\ +C\ 0 \\ \hline 5\ 9 \end{array}$$

Subtracting 64 is done by adding C to high order.

$$\begin{array}{r} 5\ 9 \\ +F\ F \\ \hline 5\ 8 \end{array}$$

Subtract 1 from the characteristic leaving it expressed in excess 127. This is done to compensate for the fact that the dividend is shifted right one digit as it is loaded.

QB431 Dividend characteristic is read out and sign is tested. 128 is added to characteristic; it now represents the dividend exponent +192.

Dividend Characteristic

$$\begin{array}{r} 2\ 5 \\ +8\ 0 \\ \hline A\ 5 \end{array}$$

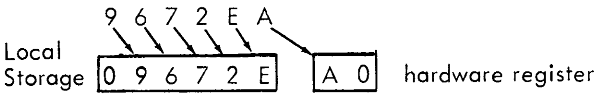
Sign was plus; characteristic is now expressed in excess 192.

Compute quotient sign and store. If signs are alike, quotient sign will be plus. If signs are unlike, quotient sign will be minus.

QB421 Do the skewed dividend prenormalize into local storage and hardware

Divisor fraction is loaded into hardware, or hardware and local storage.

Dividend Fraction



QB441 Compute the Quotient characteristic. The divisor characteristic is complement added to the dividend characteristic

$$\begin{array}{rl} A\ 5 & \text{dividend characteristic} \longleftrightarrow \text{excess 192} \\ A\ 7 & \text{divisor characteristic complemented} \longleftrightarrow \text{excess 127} \\ +\quad 1 & \text{plus one} \\ \hline c \longleftarrow 4\ D & \text{quotient characteristic} \end{array}$$

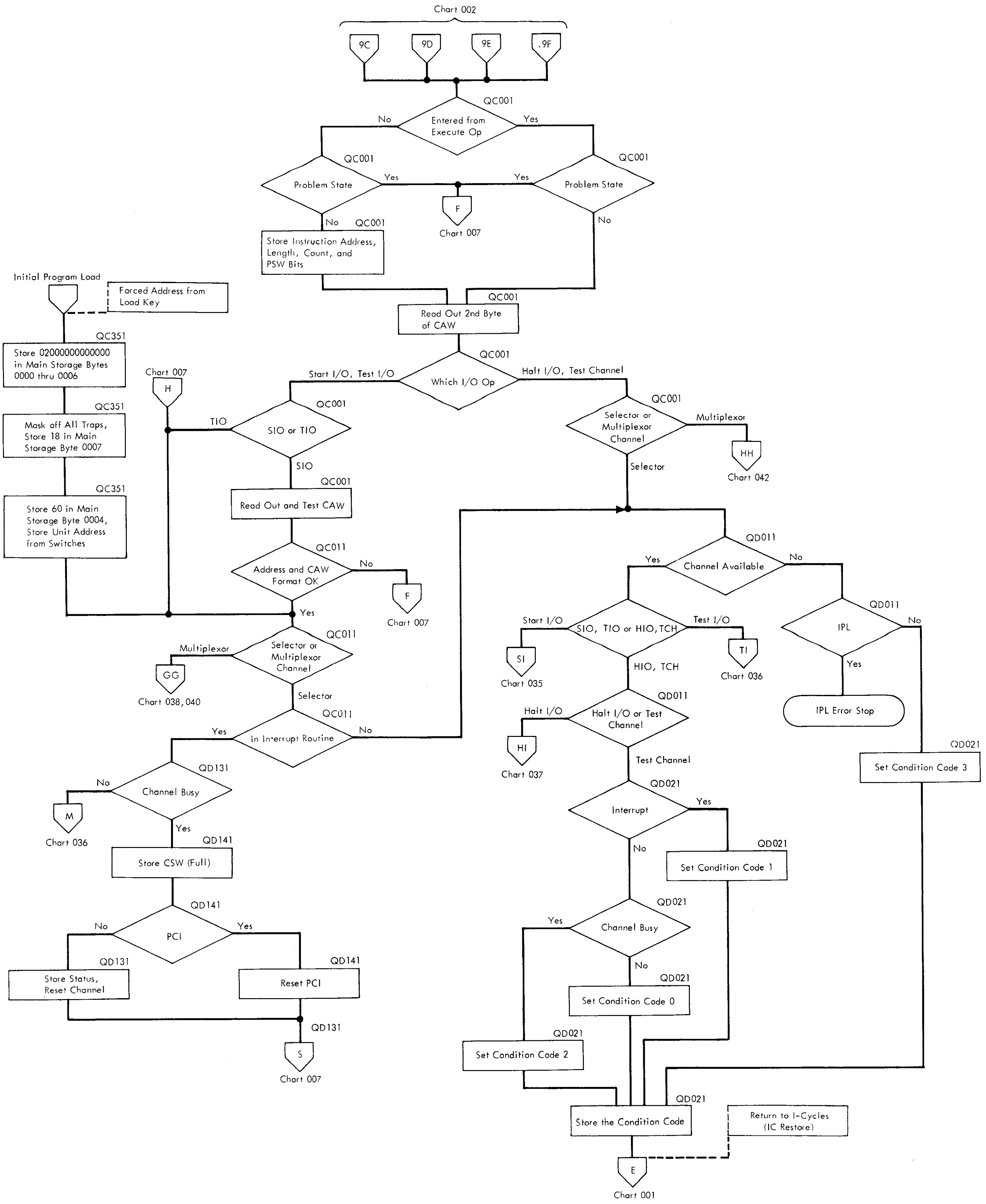
The result is now excess 65; however, when the final quotient is produced (shifted left 4 bits), it will be excess 64.

If the computation of the quotient characteristic failed to produce a carry, an underflow would be sensed. The quotient register would be set to zero and the program would return to I-cycles or to program interrupt, depending on the mask bit.

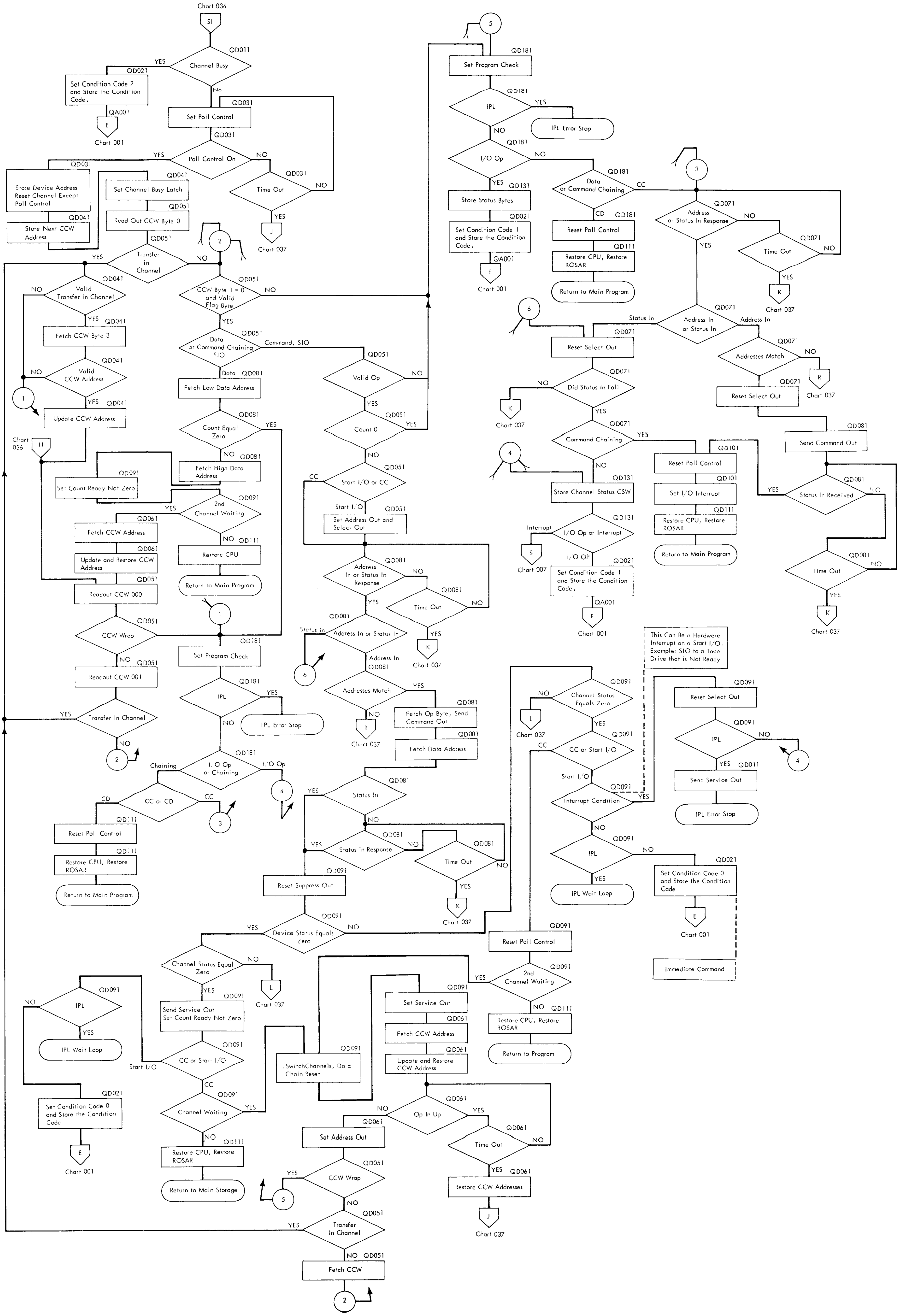
QA621
QA631 The divide function is performed on these 2 pages. The quotient is produced one bit at a time into a hardware register. The first time through the divide loop, 4 quotient bits are produced and stored in the high-order address of the quotient register. In each of the following loops, 8 quotient bits are produced and stored until the entire quotient has been computed.

	Characteristic	Fraction
Final Result	4D	1 5 4 3 8 5

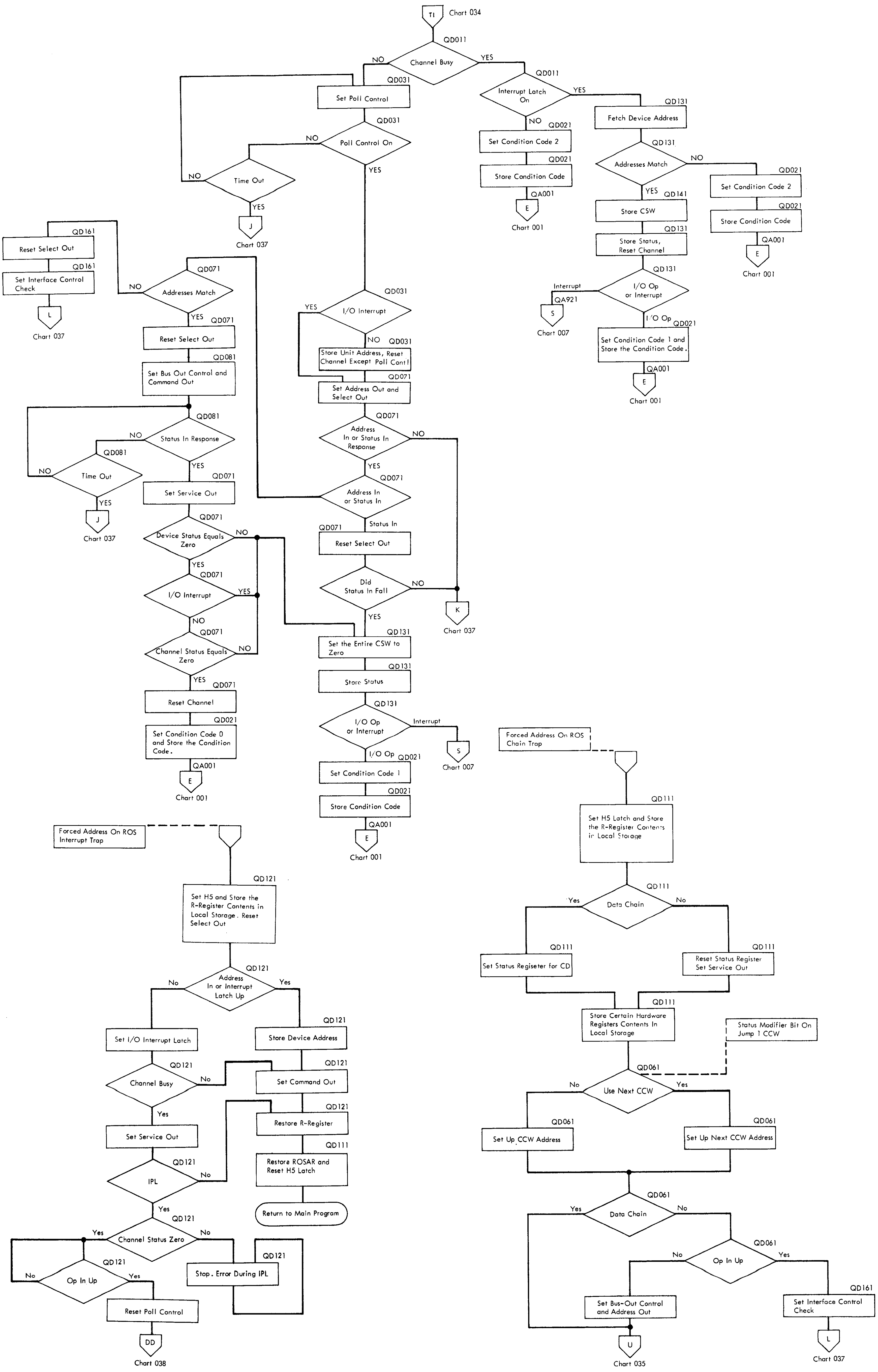
The quotient has been produced and is located in all bytes of the quotient register. The high-order byte must be checked to see if the quotient characteristic may be stored there. If the high-order byte is zero, the characteristic is stored there decremented by 1 to indicate high-order zero. If the high-order byte contains a low-order digit, (can never contain high-order digit) the quotient is right shifted one digit, the low quotient digit is lost, and the characteristic is stored unchanged. The characteristic is always checked, before storing, for overflow and underflow. If overflow, go to program interrupt. If underflow, set the quotient register to zero, test for trap, and go to normal I-cycles or program interrupt.



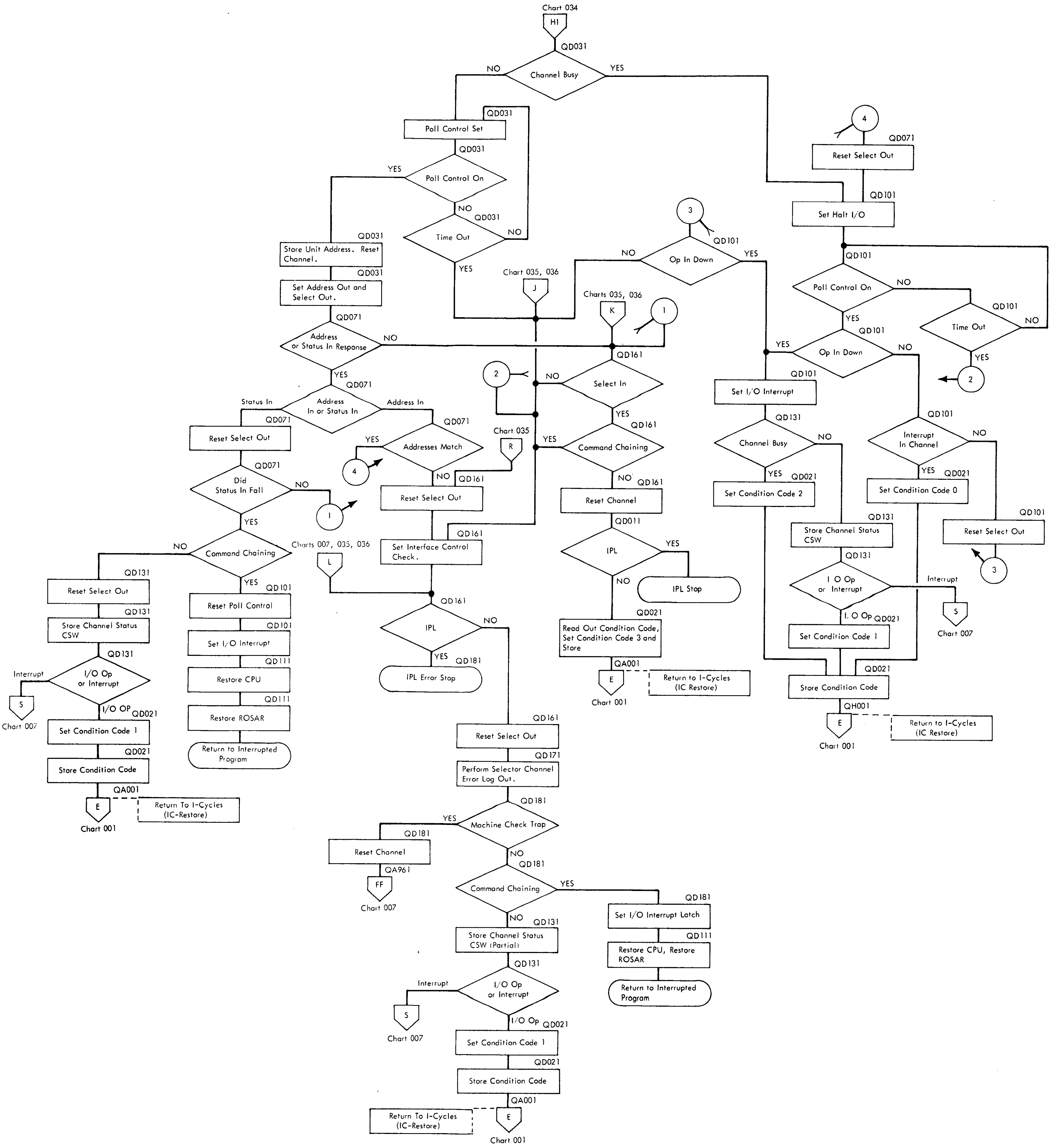
CLF 034 I/O Ops Initialization and Test Channel - Selector Channel



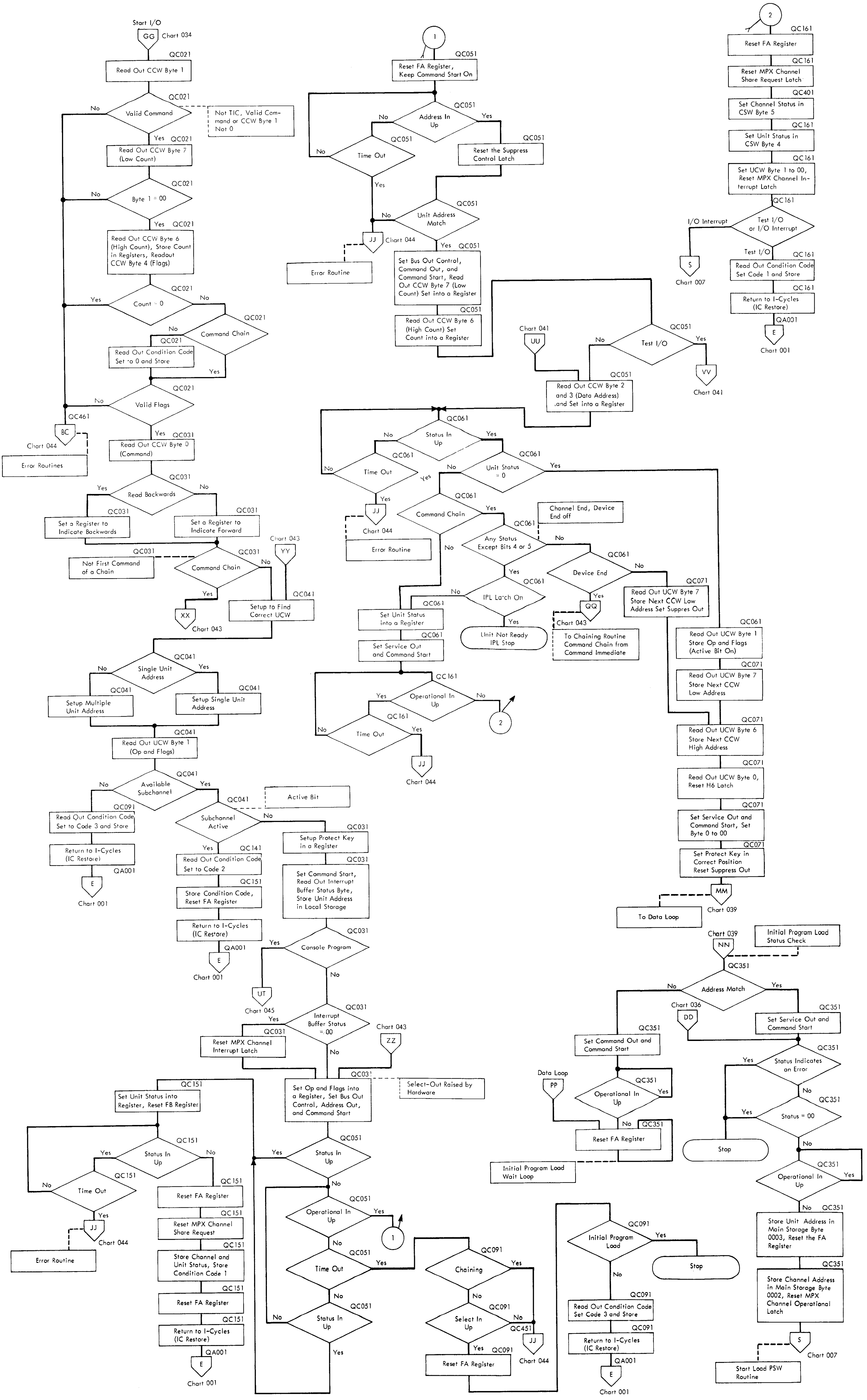
CLF 035 Start I/O - Selector Channel



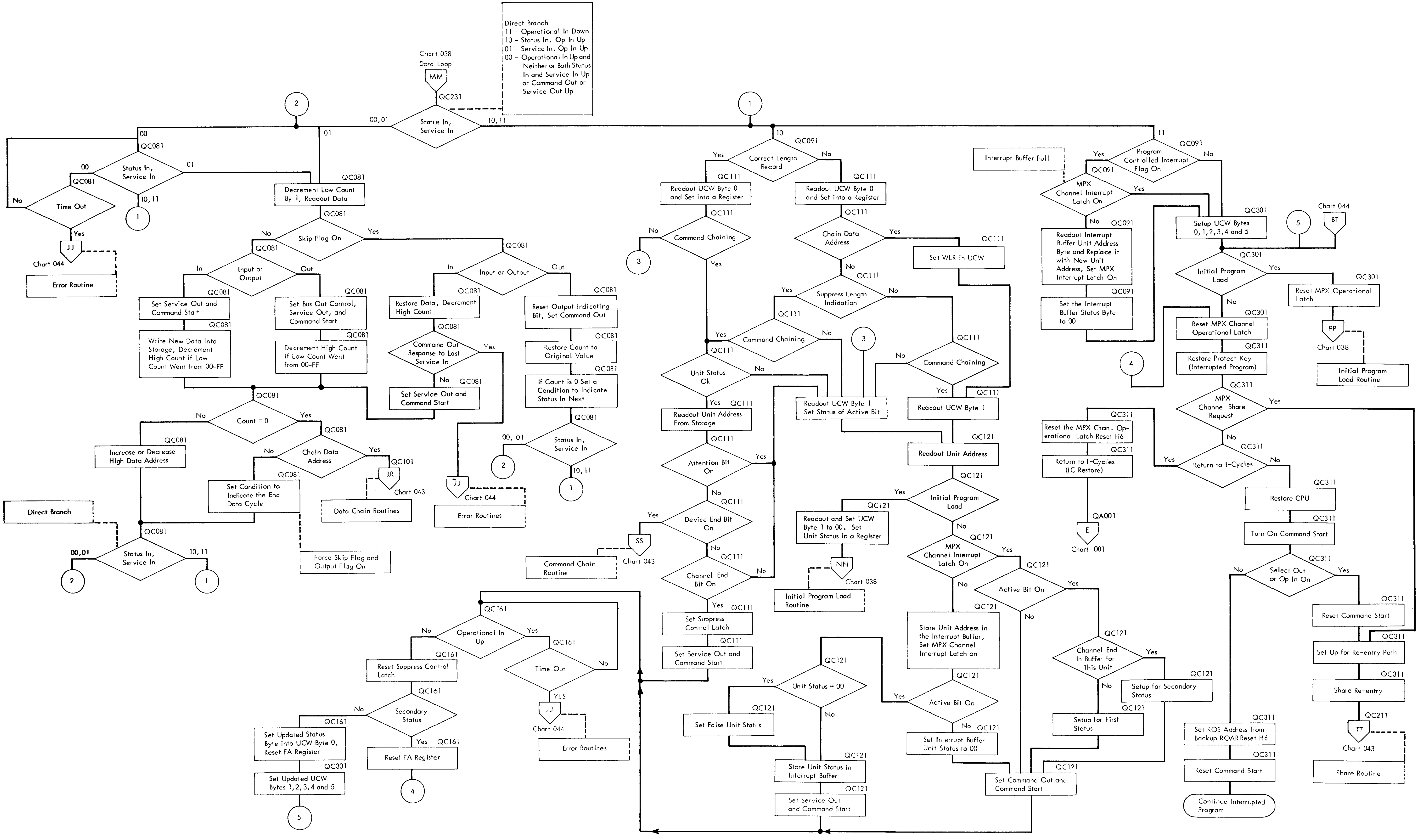
CLF 036 Test I/O and Traps-Selector Channel



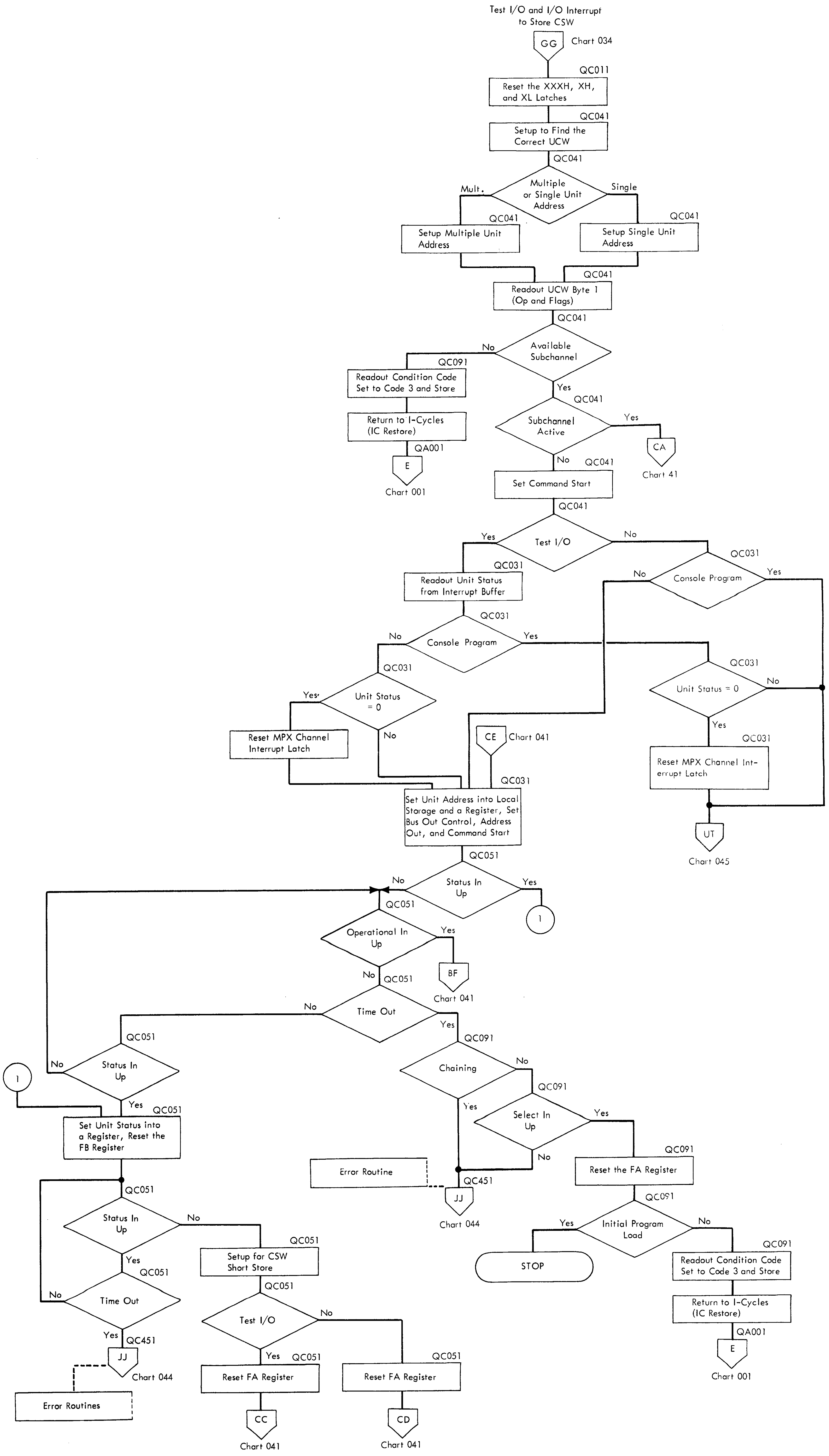
CLF 037 Halt I/O - Selector Channel



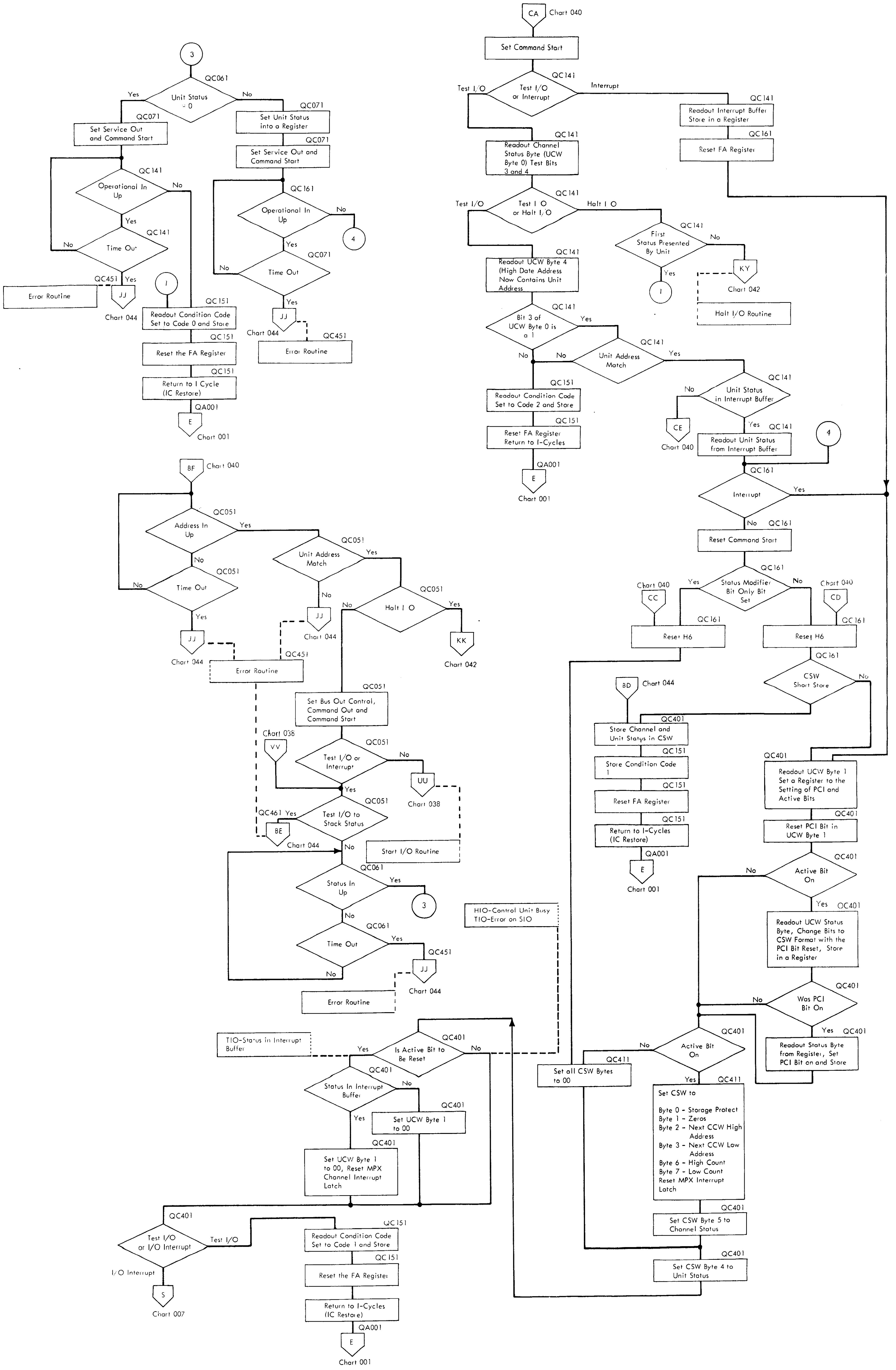
CLF 038 MPX Start I/O 1

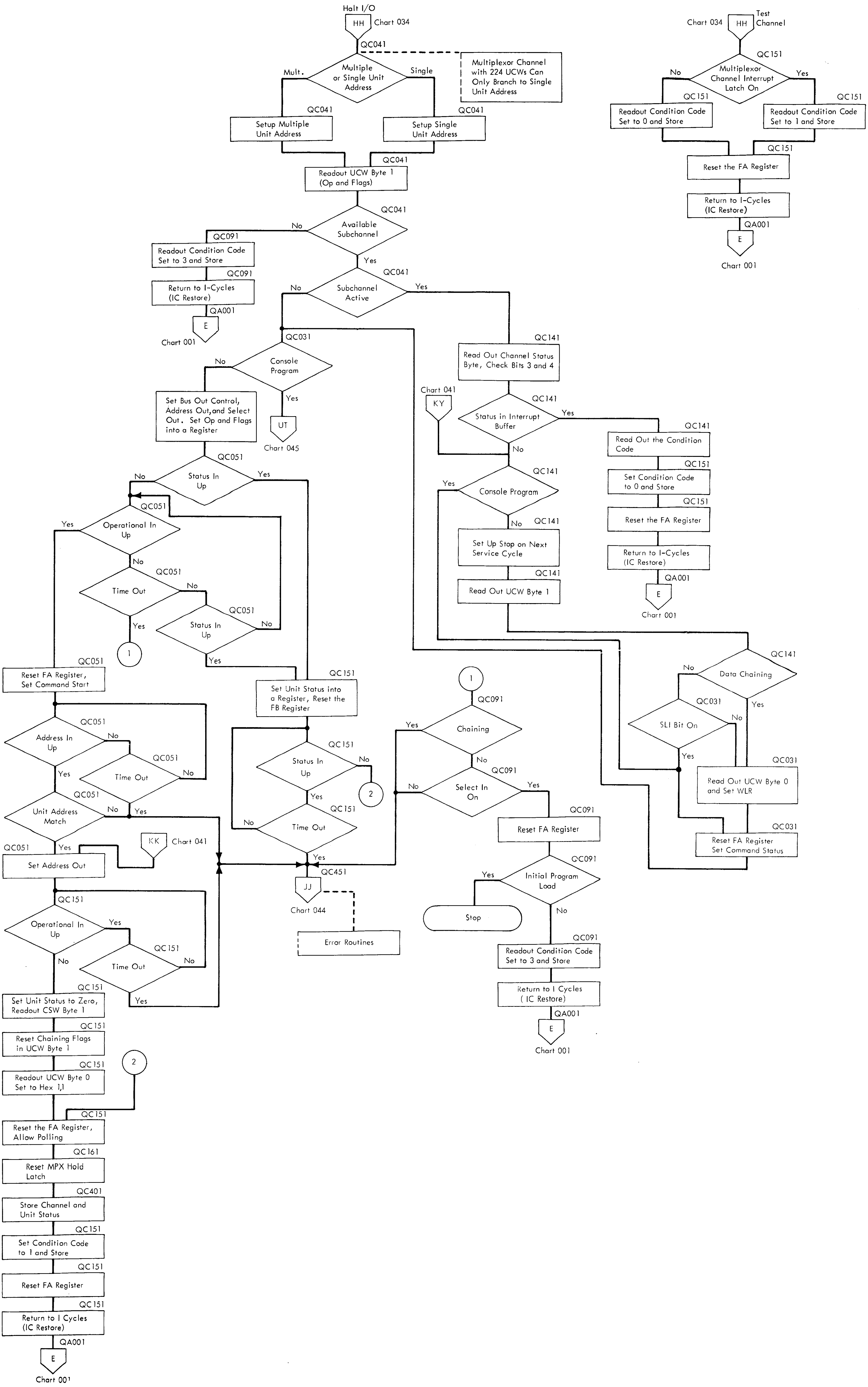


CLF 039 MPX Data Loop



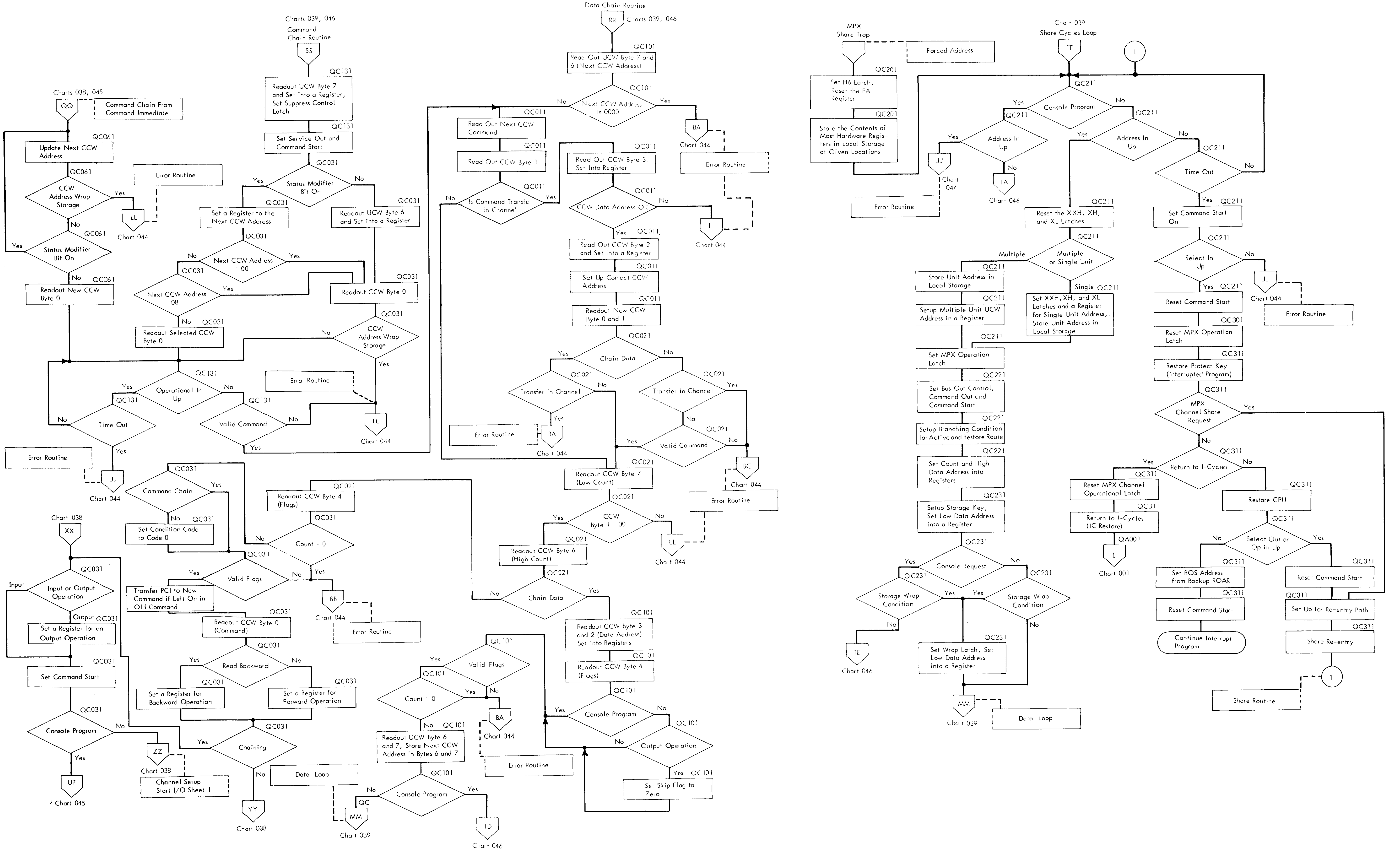
CLF 040 Test I/O - Interrupt to Store CSW

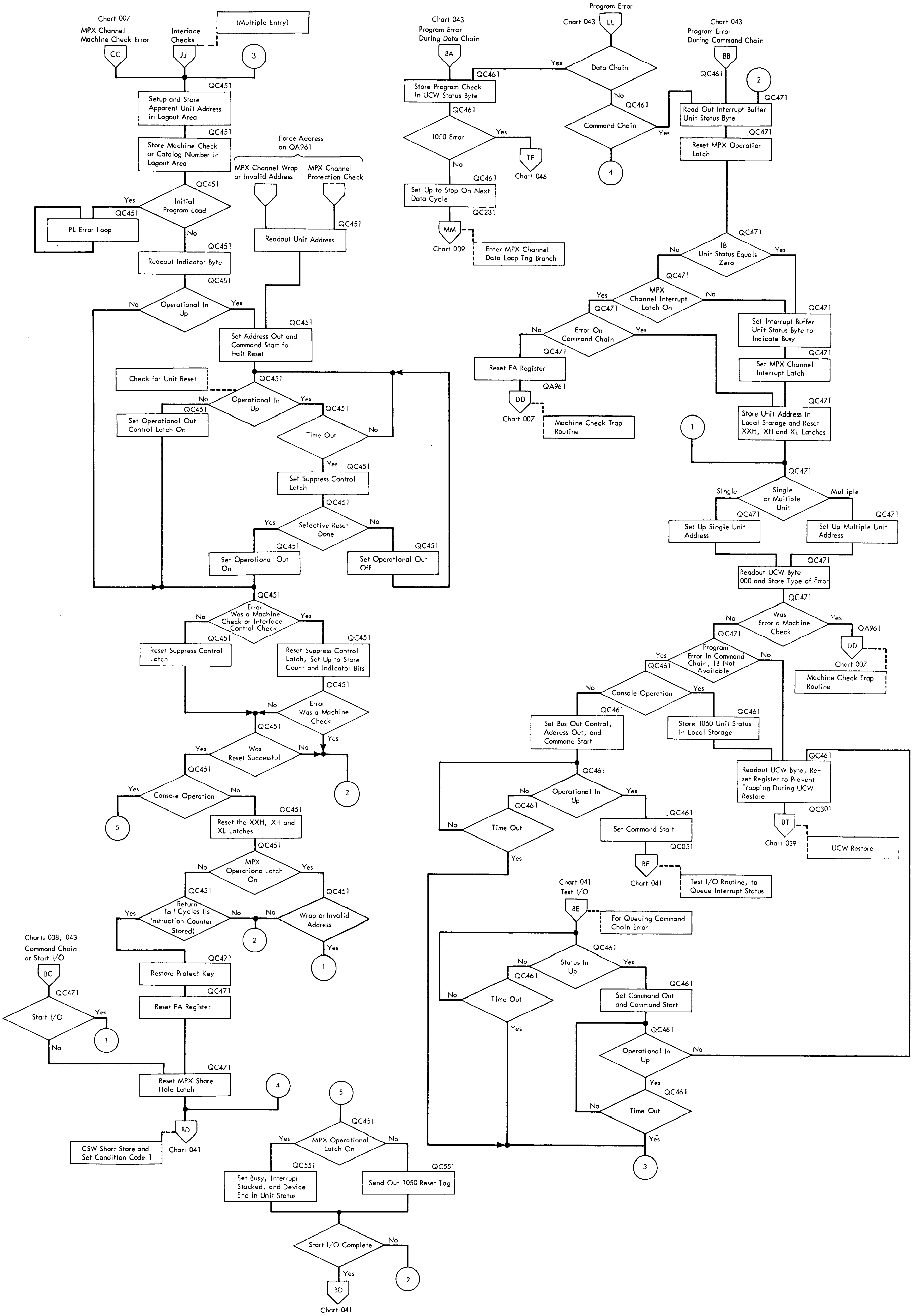

3
QC061
Unit Status = 0
Yes
No
QC071
Set Service Out and Command Start
QC071
Set Unit Status into a Register
QC071
Set Service Out and Command Start
QC141
Operational In Up
No
Yes
QC141
Time Out
No
QC451
Yes
Error Routine
JJ
Chart 044
1
QC151
Readout Condition Code Set to Code 0 and Store
QC151
Reset the FA Register
QC151
Return to I Cycle (IC Restore)
QA001
E
Chart 001
QC161
Operational In Up
No
4
Yes
QC071
Time Out
No
Yes
JJ
Chart 044
QC451
Error Routine
CA
Chart 040
Set Command Start
QC141
Test I/O or Interrupt
Test I/O
Interrupt
QC141
Readout Interrupt Buffer Store in a Register
QC161
Reset FA Register
QC141
Readout Channel Status Byte (UCW Byte 0) Test Bits 3 and 4
QC141
Test I/O or Halt I/O
Test I/O
Halt I/O
QC141
First Status Presented By Unit
No
KY
Chart 042
Yes
1
Halt I/O Routine
QC141
Readout UCW Byte 4 (High Date Address Now Contains Unit Address
QC141
Bit 3 of UCW Byte 0 is a 1
Yes
No
No
QC141
Unit Address Match
Yes
QC141
Unit Status in Interrupt Buffer
No
CE
Chart 040
Yes
QC141
Readout Unit Status from Interrupt Buffer
4
QC151
Readout Condition Code Set to Code 2 and Store
QC151
Reset FA Register Return to I-Cycles
QA001
E
Chart 001
BF
Chart 040
QC051
Address In Up
Yes
No
QC051
Time Out
No
Yes
JJ
Chart 044
QC051
Unit Address Match
Yes
No
JJ
Chart 044
QC051
Halt I/O
No
Yes
KK
Chart 042
QC451
Error Routine
QC051
Set Bus Out Control, Command Out and Command Start
Chart 038
VV
QC051
Test I/O or Interrupt
No
UU
Chart 038
Yes
QC051
Test I/O to Stack Status
QC461
Yes
BE
Chart 044
No
Start I/O Routine
QC061
Status In Up
Yes
3
No
QC061
Time Out
Yes
No
QC451
JJ
Chart 044
Error Routine
HIO-Control Unit Busy
TIO-Error on SIO
TIO-Status in Interrupt Buffer
QC401
Is Active Bit to Be Reset
Yes
No
QC401
Status In Interrupt Buffer
No
QC401
Set UCW Byte 1 to 00
Yes
QC401
Set UCW Byte 1 to 00, Reset MPX Channel Interrupt Latch
QC401
Test I/O or I/O Interrupt
Test I/O
I/O Interrupt
S
Chart 007
QC151
Readout Condition Code Set to Code 1 and Store
QC151
Reset the FA Register
Return to I-Cycles (IC Restore)
QA001
E
Chart 001
QC161
Interrupt
Yes
No
QC161
Reset Command Start
QC161
Status Modifier Bit Only Bit Set
Yes
No
Chart 040
CC
Chart 040
CD
QC161
Reset H6
QC161
Reset H6
QC161
CSW Short Store
No
BD
Chart 044
QC401
Store Channel and Unit Status in CSW
QC151
Store Condition Code 1
QC151
Reset FA Register
QC151
Return to I-Cycles (IC Restore)
QA001
E
Chart 001
QC401
Readout UCW Byte 1 Set a Register to the Setting of PCI and Active Bits
QC401
Reset PCI Bit in UCW Byte 1
QC401
Active Bit On
No
Yes
QC401
Readout UCW Status Byte, Change Bits to CSW Format with the PCI Bit Reset, Store in a Register
QC401
Was PCI Bit On
No
Yes
QC401
Readout Status Byte from Register, Set PCI Bit on and Store
QC401
Active Bit On
No
QC411
Set all CSW Bytes to 00
Yes
QC411
Set CSW to
Byte 0 - Storage Protect
Byte 1 - Zeros
Byte 2 - Next CCW High Address
Byte 3 - Next CCW Low Address
Byte 6 - High Count
Byte 7 - Low Count
Reset MPX Interrupt Latch
QC401
Set CSW Byte 5 to Channel Status
QC401
Set CSW Byte 4 to Unit Status



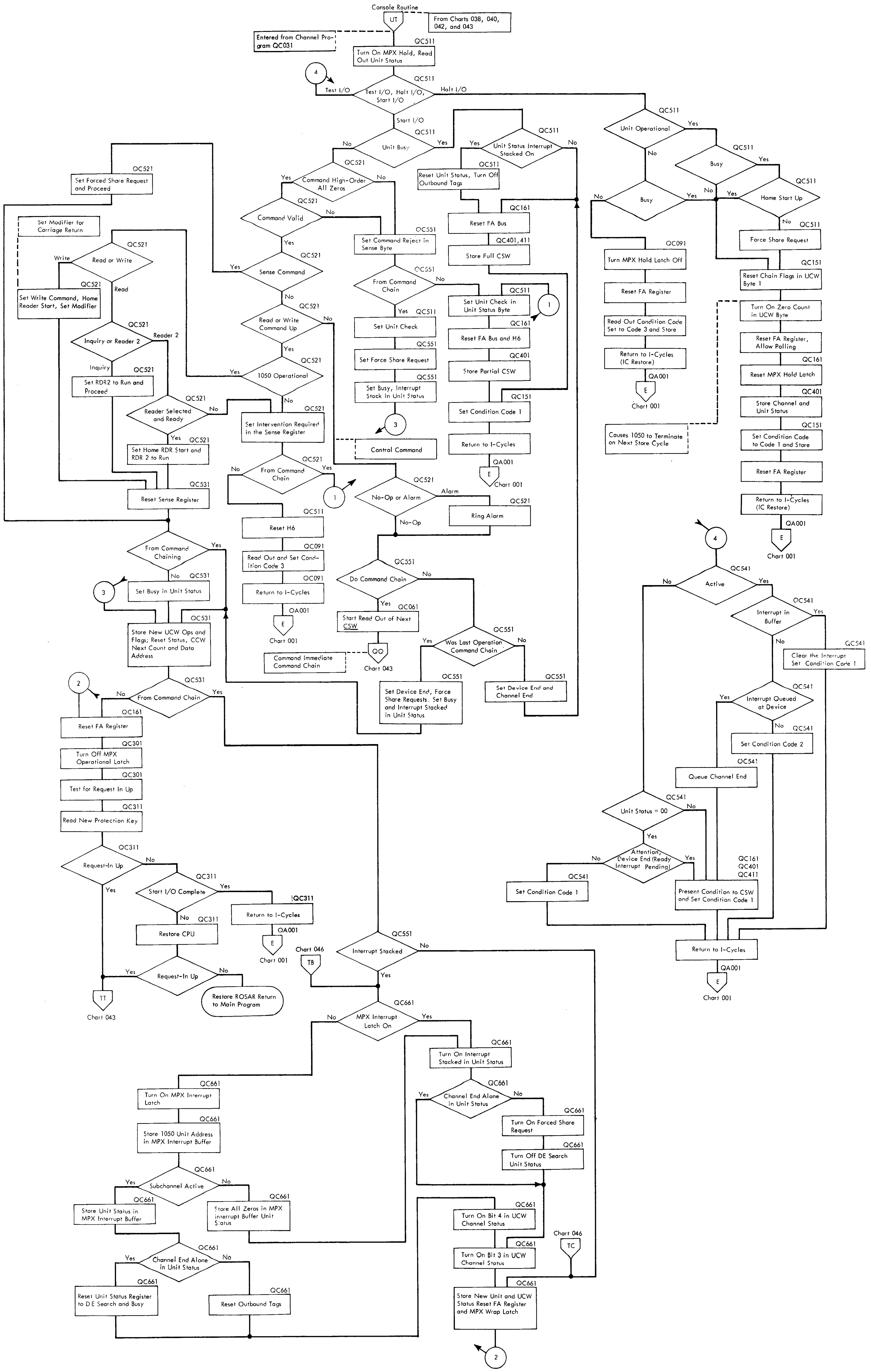
CLF 042 MPX Halt I/O, Test Channel

CLF 043 MPX Chaining, Transfer in Channel and Share Request

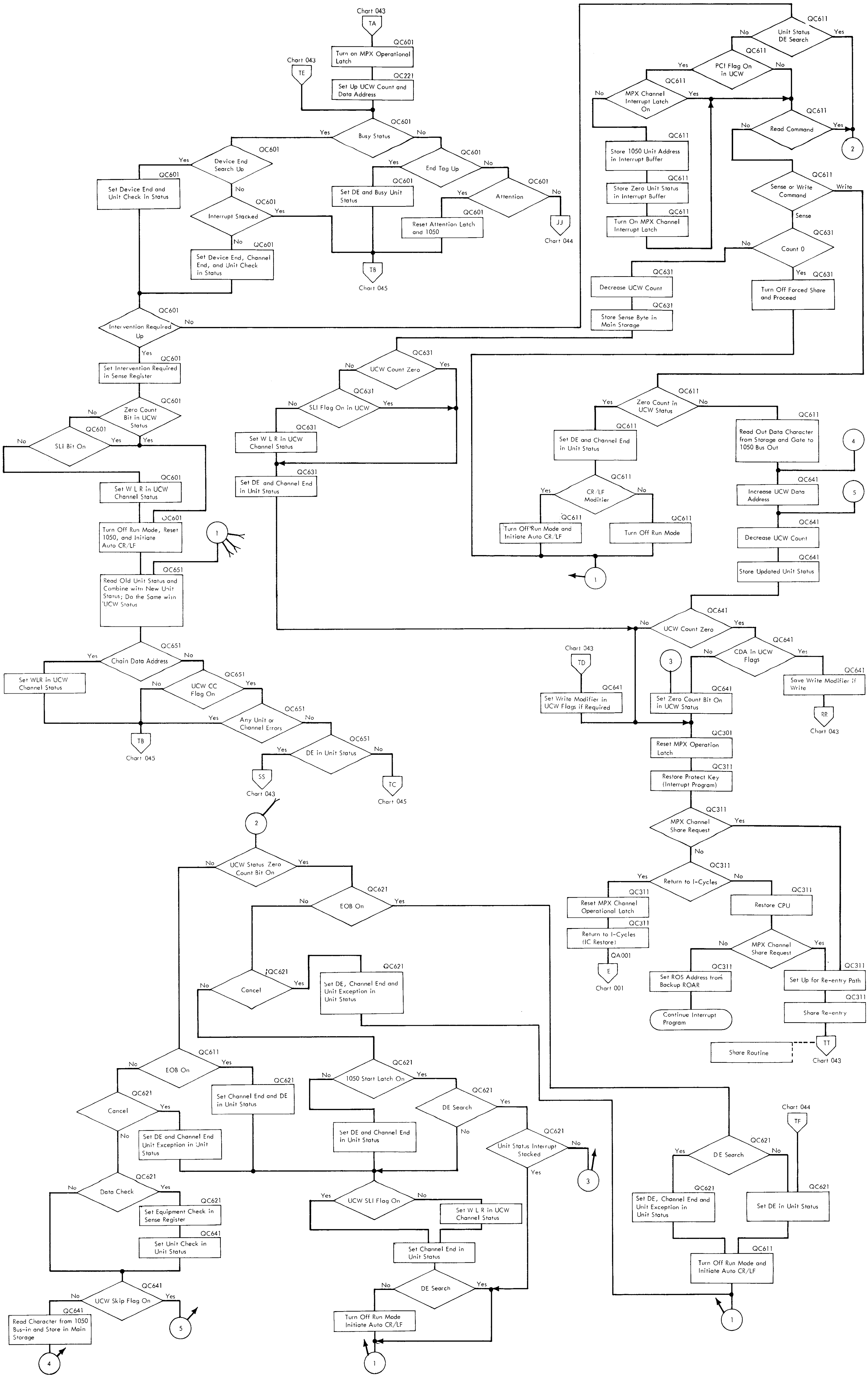




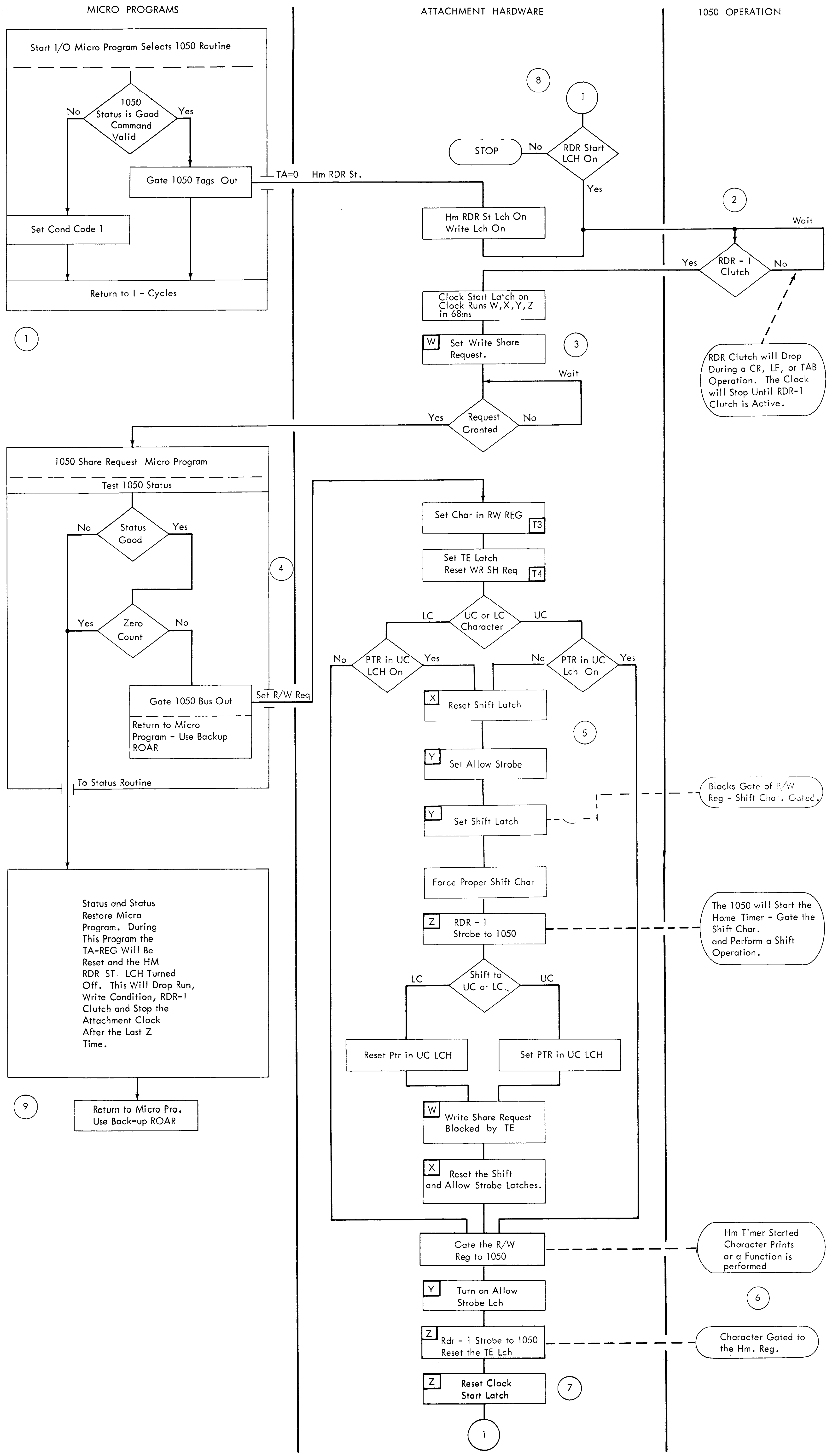
CLF 044 MPX Error Routines



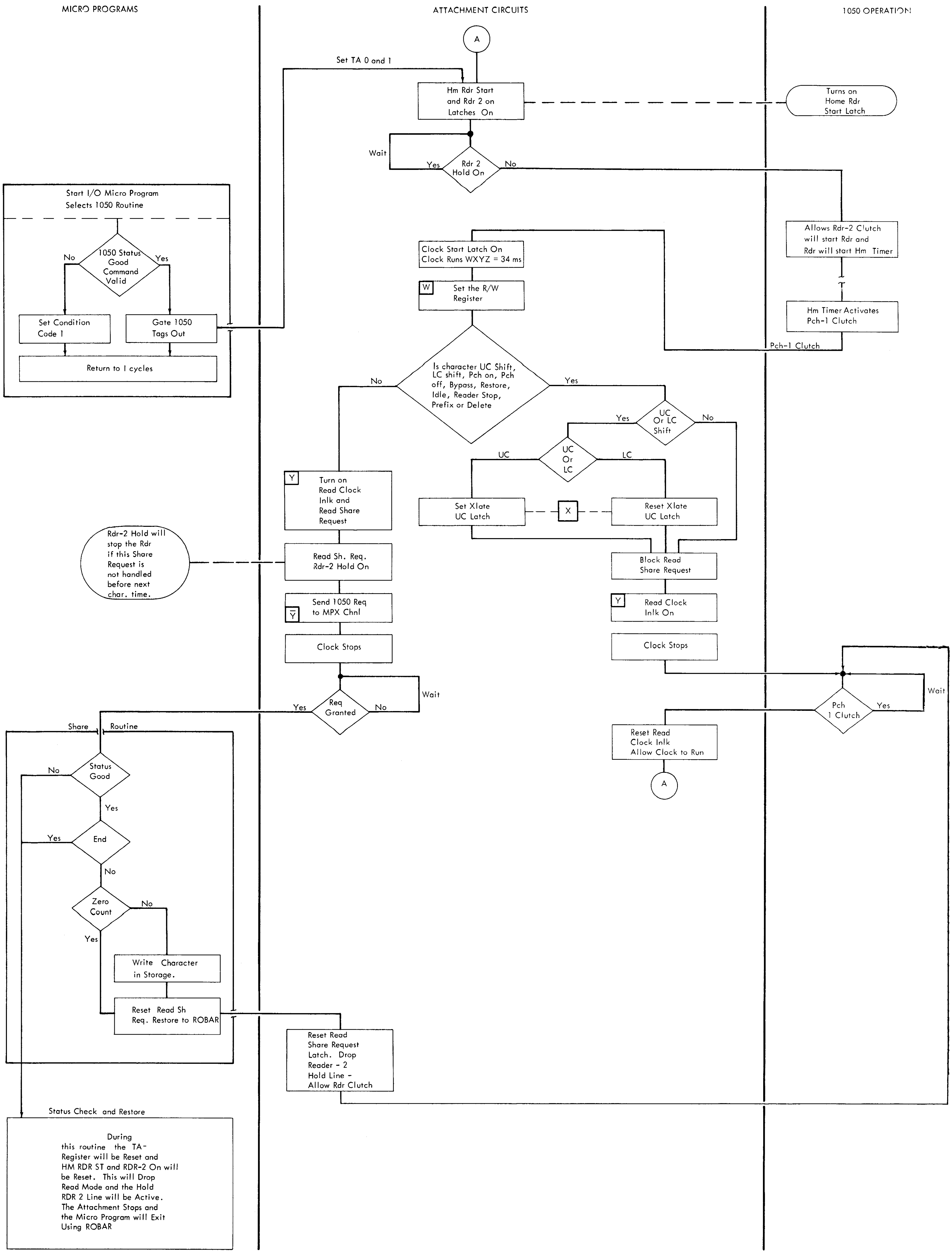
CLF 045 1050 Operations, Sheet 1



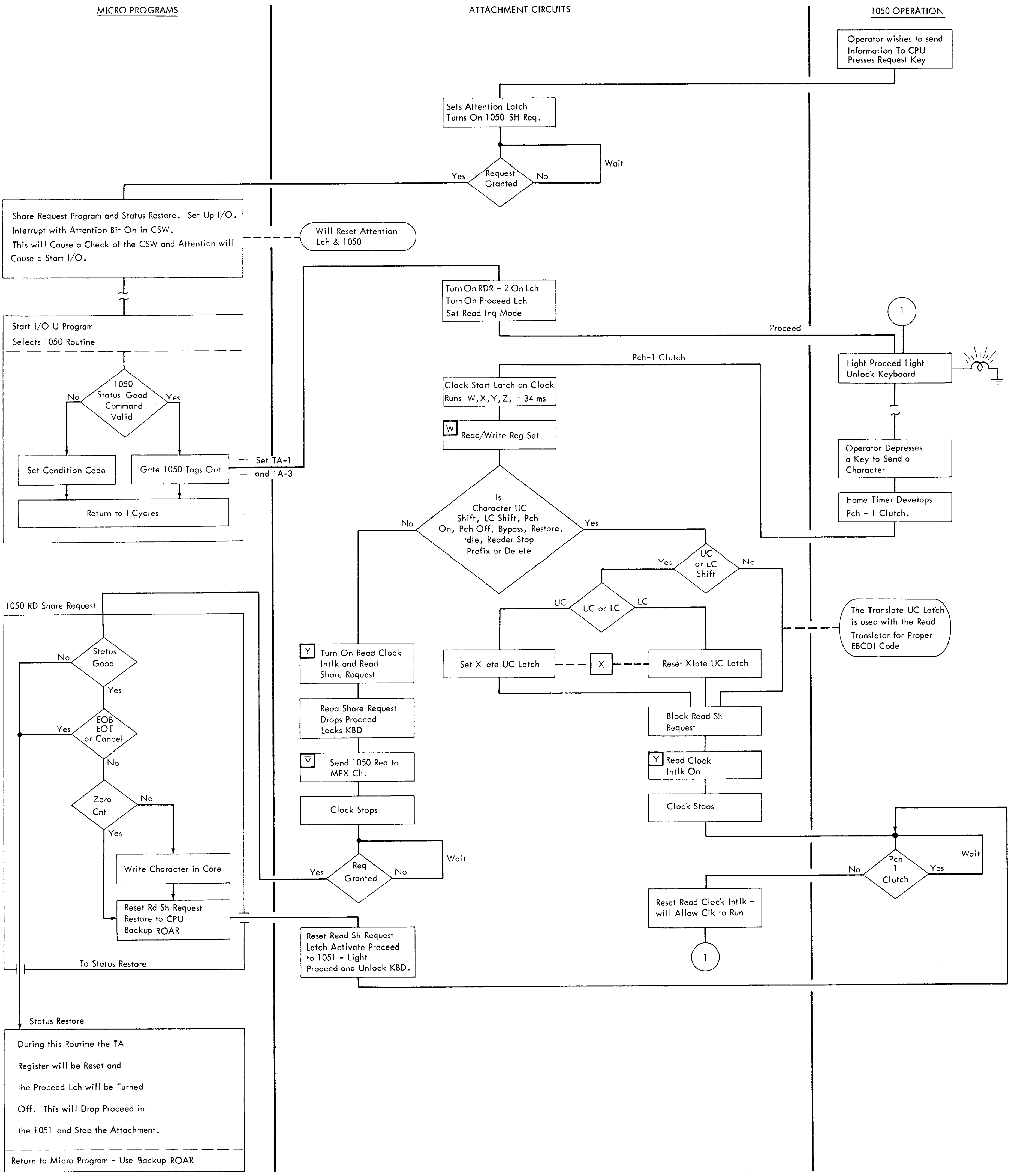
CLF 046   1050 Operations, Sheet 2



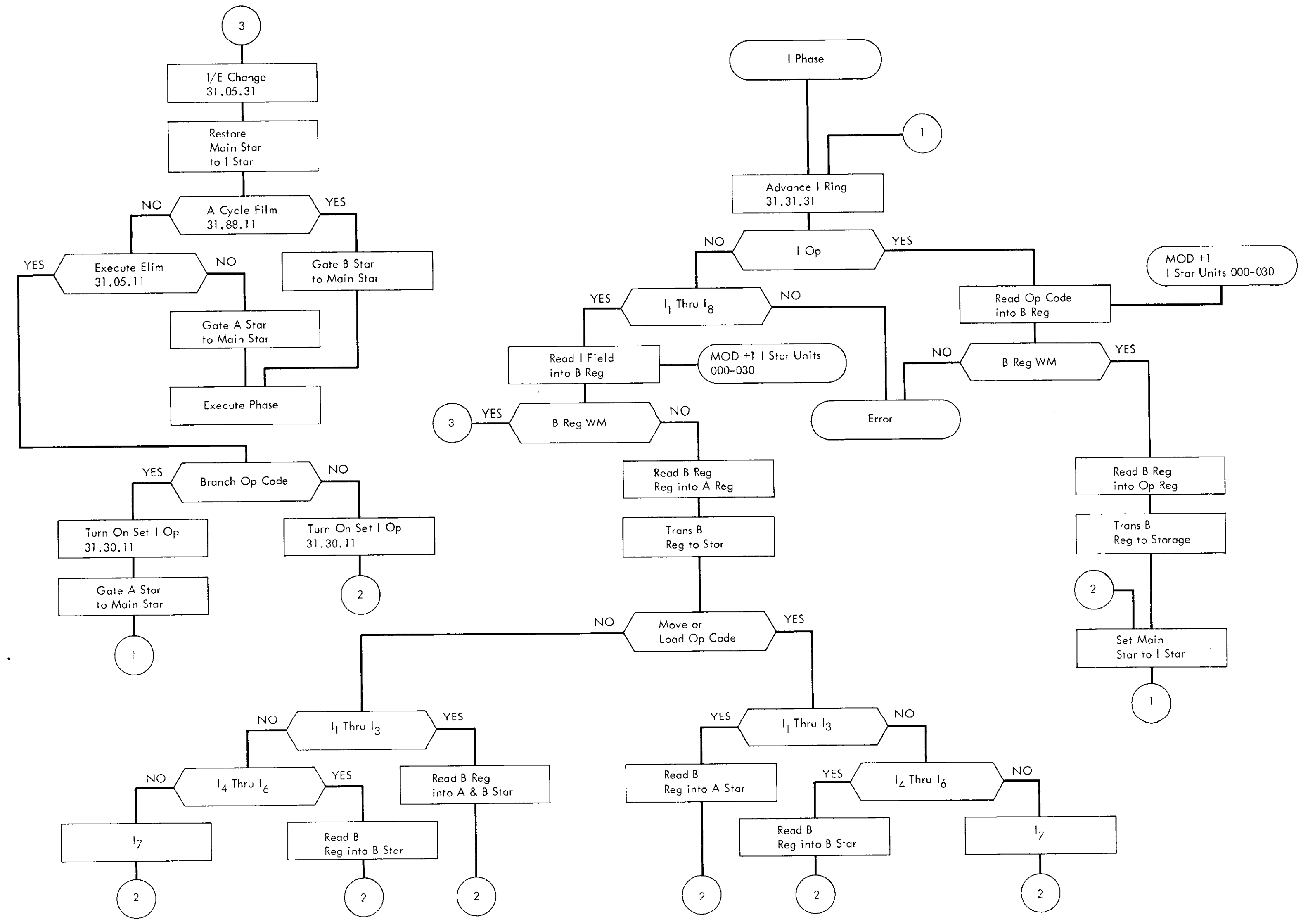
CLF 047 1050 Write Operation



CLF 048 1050 – Read Reader-2 Operation



CLF 049 1050 - Read Inquiry Operation

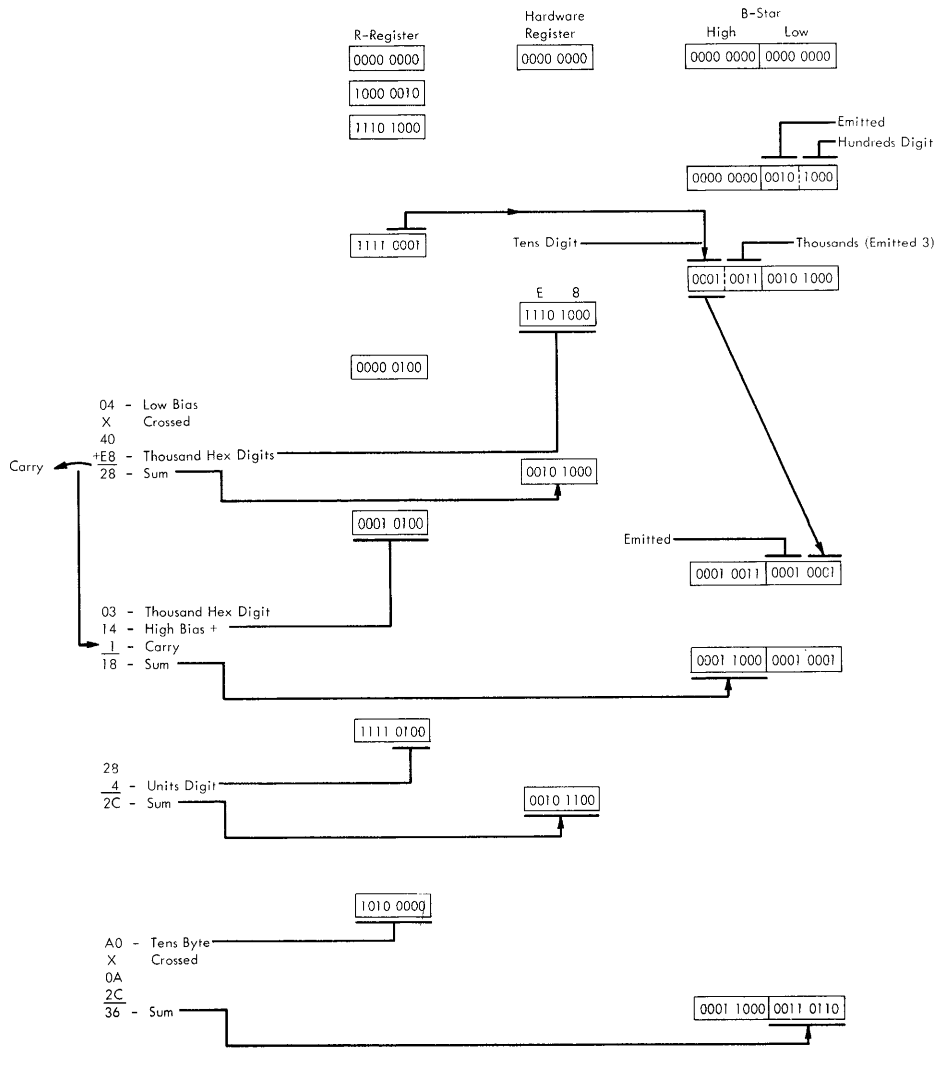


Example for A-Star or B-Star Address Development During I-Phase of 1401 Compatibility Feature on the 2030.

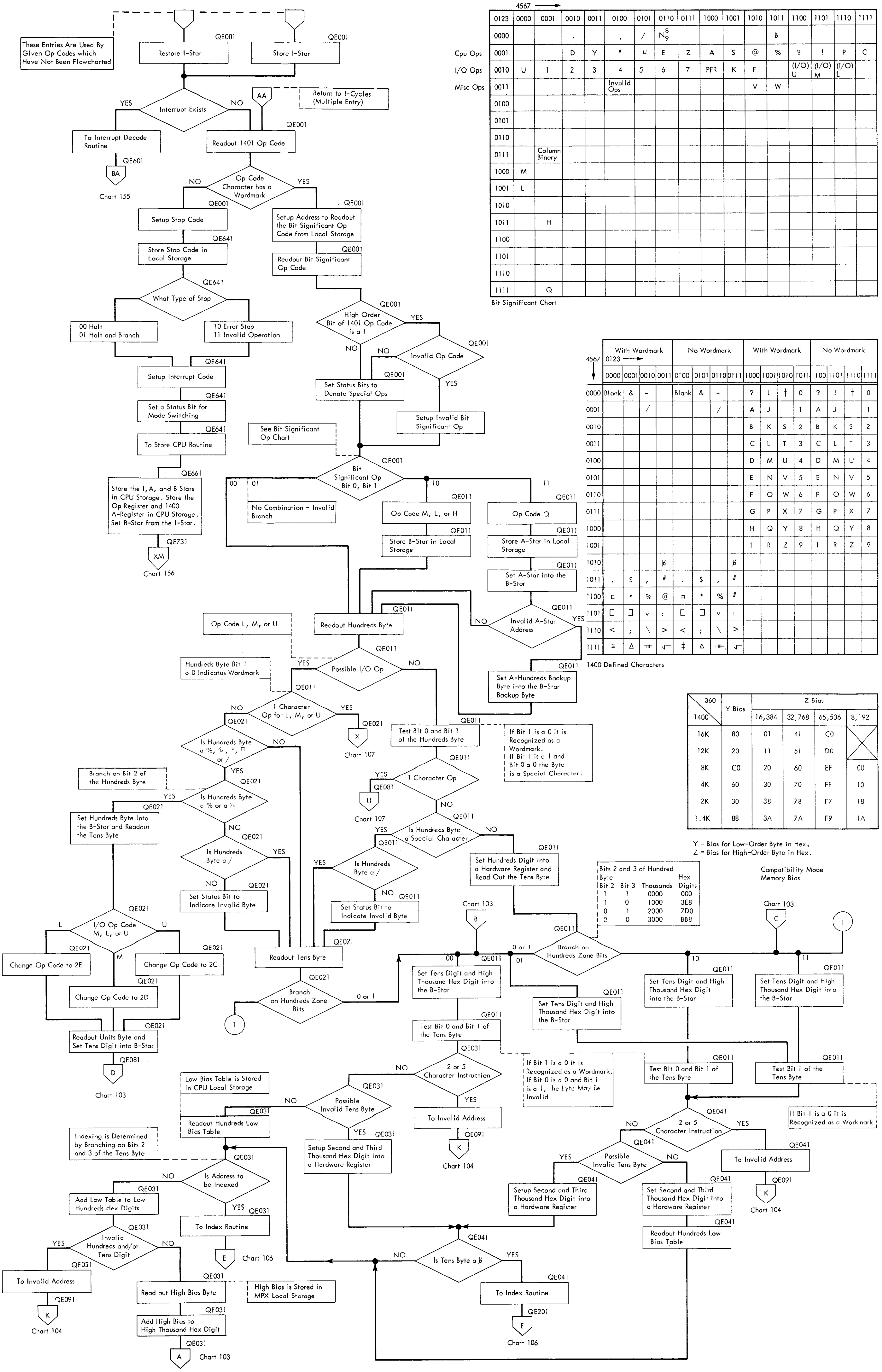
Assume the 2030 has 16,384 Positions of Storage and the 1401 Program is for a 12,000 Positions of Storage 1401. The Bias is 1120

Instruction is B̲ Y14 E A̲

1. Read Op Code
2. Readout Hundreds Byte
3. Setup Address for Table Lookup by Emitting a Digit and Using the Hundreds Digit. Set Address into Low B-Star.
4. Readout Tens Byte and Test Hundreds Zone to Determine Thousands Digit. This Example has Zone of 10 for 1 Thousand (Hex Equivalent is 3E8). Set Tens Digit and Thousands High Hex Digit into the High B-Star.
5. Set Remaining Two Hex Digits (Emitted) into a Hardware Register.
6. Readout Low Bias Plus Hundreds Digit Hex Equivalent Byte from Table in CPU Storage. Low B-Star has Address of Position in Table. For this Example Table Readout is 04.
7. A Test is made on the Tens Byte for Zones. If Tens Byte is Zoned, Indexing is Required. For Example Tens Digit is not Zoned. Add Low Bias Plus Hundreds Digit Hex Equivalent Byte (Remember this Byte is Crossed in the Storage Table) to Digits in the Hardware Register. Retain any Carry in the Carry Latch.
8. Readout the High Bias Plus Hundreds Digit Hex Equivalent Byte from Table in MPX Storage. Use Same Address as Used for Low Bias. For this Example Table Readout is 14.
9. Setup Table Address for Tens Digit Hex Equivalent by Setting the Tens Digit from the High B-Star to the Low B-Star and Emitting a Digit to the Low B-Star.
10. Add High Hex Thousands Digit in High B-Star and High Bias Plus Hundreds Digit Hex. Equivalent Byte and any Carry from Low Bias Addition to Thousand Digit. Set Sum into B-Star.
11. Readout Units Byte.
12. Add Units Digit to Amount in the Hardware Register.
13. Test Zone Bits of Units Byte to Determine Remaining Thousands Address. For this Example Units Zone is 11 (No Zone) for 0 Thousands.
14. Readout the Hex Equivalent Byte of the Tens Digit from the Table in CPU Storage Using The Address Developed in the Low B-Star. The Digits in the Byte are Crossed in the Storage Table. For this Example Table Readout is A0.
15. Add the Hex Equivalent Byte (Uncrossed) to the Amount in the Hardware Register and Set the Sum into the Low B-Star. Any Carry is Added to the High B-Star.
16. The B-Star Now Contains the Required Hex Equivalent Address for the Address Y14 for the Condition in the Example. The B-Star is Transferred to the A-Star and I-Phase Continues. If Instruction has a B-Address the B-Star is Developed in Same Manner.



CLF 101 1400 I-Cycles, Address Example



4567 →

	0123	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	0000			.		,	/	N_9^8					B				
Cpu Ops	0001			D	Y	#	¤	E	Z	A	S	@	%	?	!	P	C
I/O Ops	0010	U	1	2	3	4	5	6	7	PFR	K	F		(I/O) U	(I/O) M	(I/O) L	
Misc Ops	0011					Invalid Ops						V	W				
	0100																
	0101																
	0110																
	0111		Column Binary														
	1000	M															
	1001	L															
	1010																
	1011		H														
	1100																
	1101																
	1110																
	1111		Q														

Bit Significant Chart

4567 ↓	With Wordmark 0123 →				No Wordmark				With Wordmark				No Wordmark			
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	Blank	&	-		Blank	&	-		?	!	ǂ	0	?	!	ǂ	0
0001			/				/		A	J		1	A	J		1
0010									B	K	S	2	B	K	S	2
0011									C	L	T	3	C	L	T	3
0100									D	M	U	4	D	M	U	4
0101									E	N	V	5	E	N	V	5
0110									F	O	W	6	F	O	W	6
0111									G	P	X	7	G	P	X	7
1000									H	Q	Y	8	H	Q	Y	8
1001									I	R	Z	9	I	R	Z	9
1010				b̸				b̸								
1011	.	$	,	#	.	$	,	#								
1100	¤	*	%	@	¤	*	%	#								
1101	[	]	v	:	[	]	v	:								
1110	<	;	\	>	<	;	\	>								
1111	ǂ	Δ	⧺	√	ǂ	Δ	⧺.	√								

1400 Defined Characters

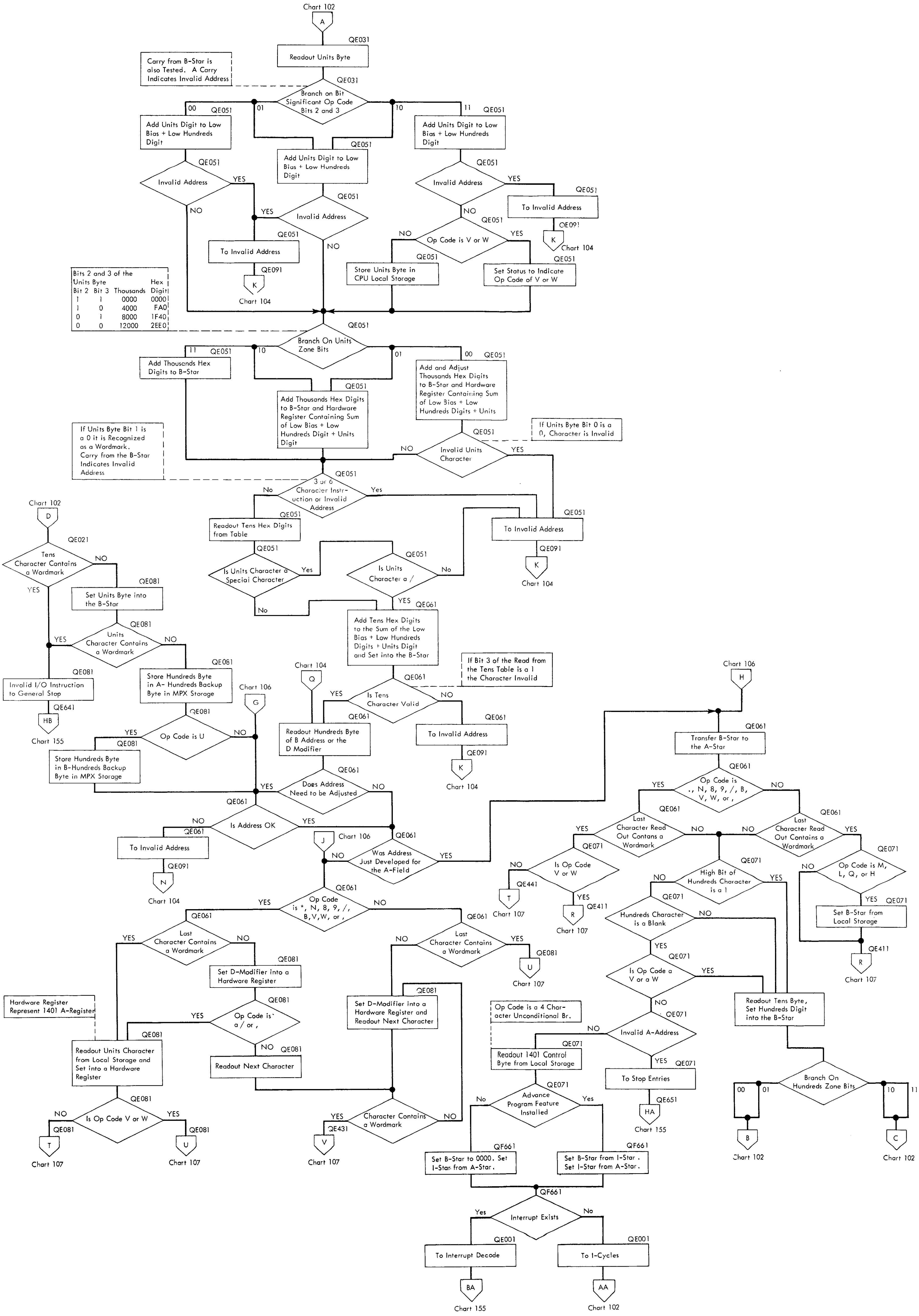
360 / 1400	Y Bias	Z Bias			
		16,384	32,768	65,536	8,192
16K	80	01	41	C0	☒
12K	20	11	51	D0	
8K	C0	20	60	EF	00
4K	60	30	70	FF	10
2K	30	38	78	F7	18
1.4K	88	3A	7A	F9	1A

Y = Bias for Low-Order Byte in Hex.
Z = Bias for High-Order Byte in Hex.

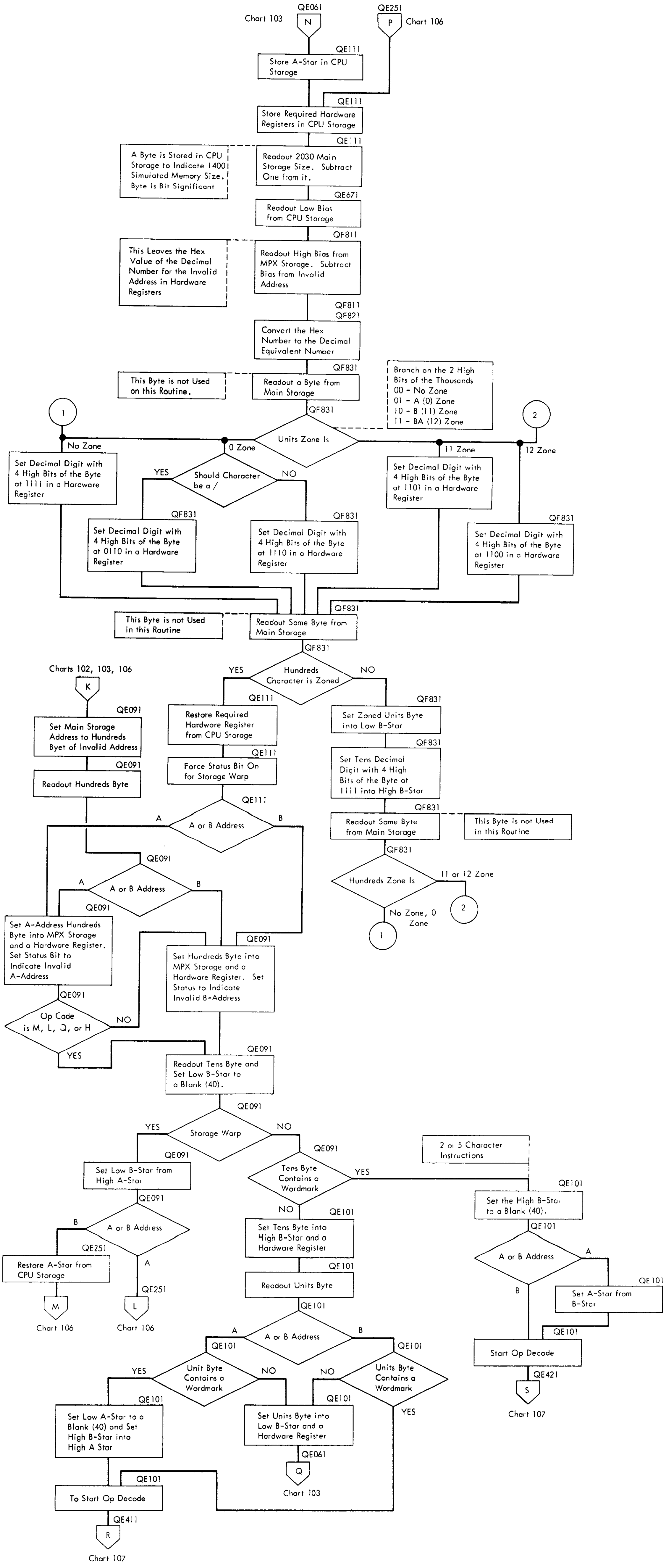
Compatibility Mode Memory Bias

Bits 2 and 3 of Hundred Byte

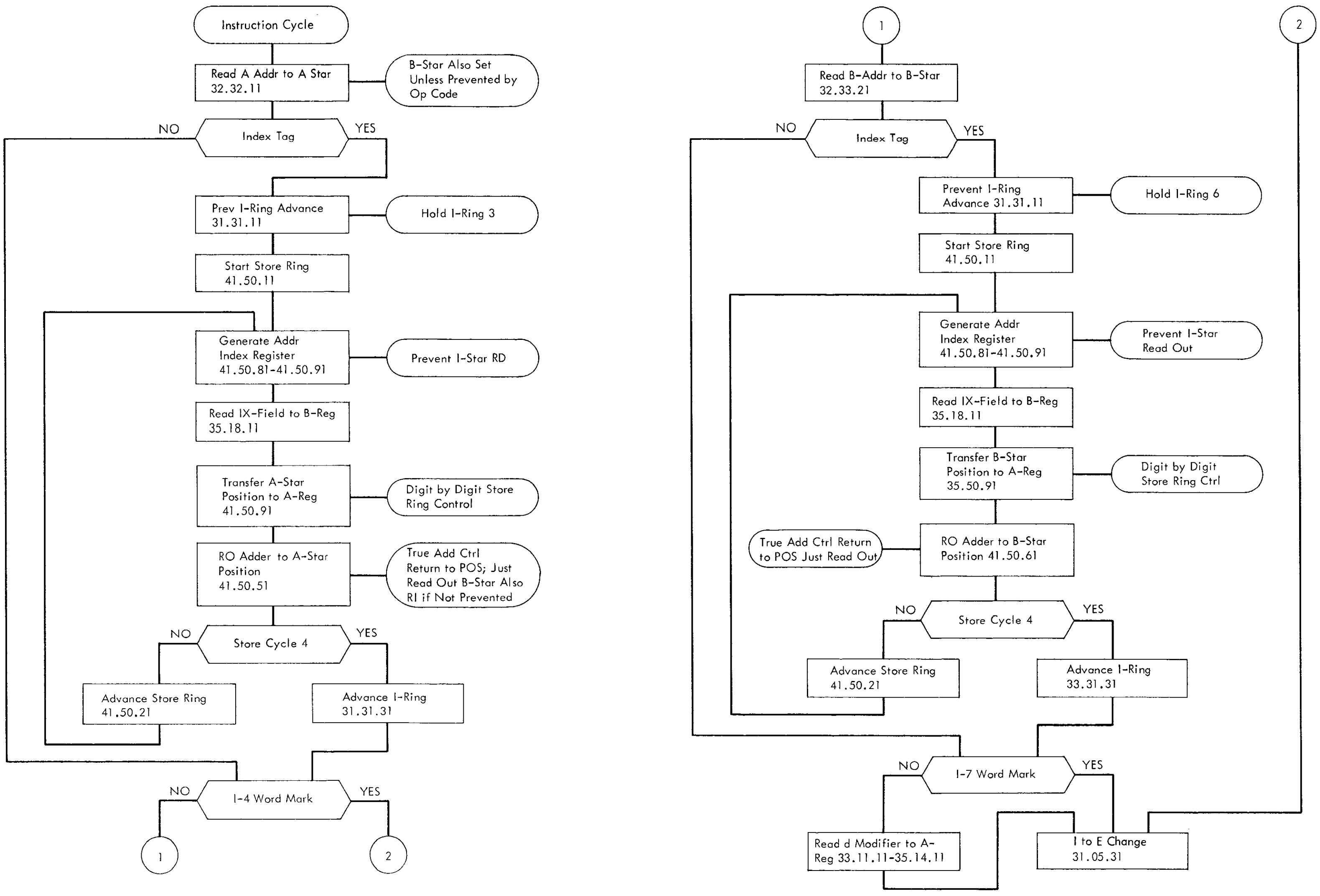
Bit 2	Bit 3	Thousands	Hex Digits
1	1	0000	000
1	0	1000	3E8
0	1	2000	7D0
0	0	3000	BB8



CLF 103 1400 I-Cycles A and B Address Setup



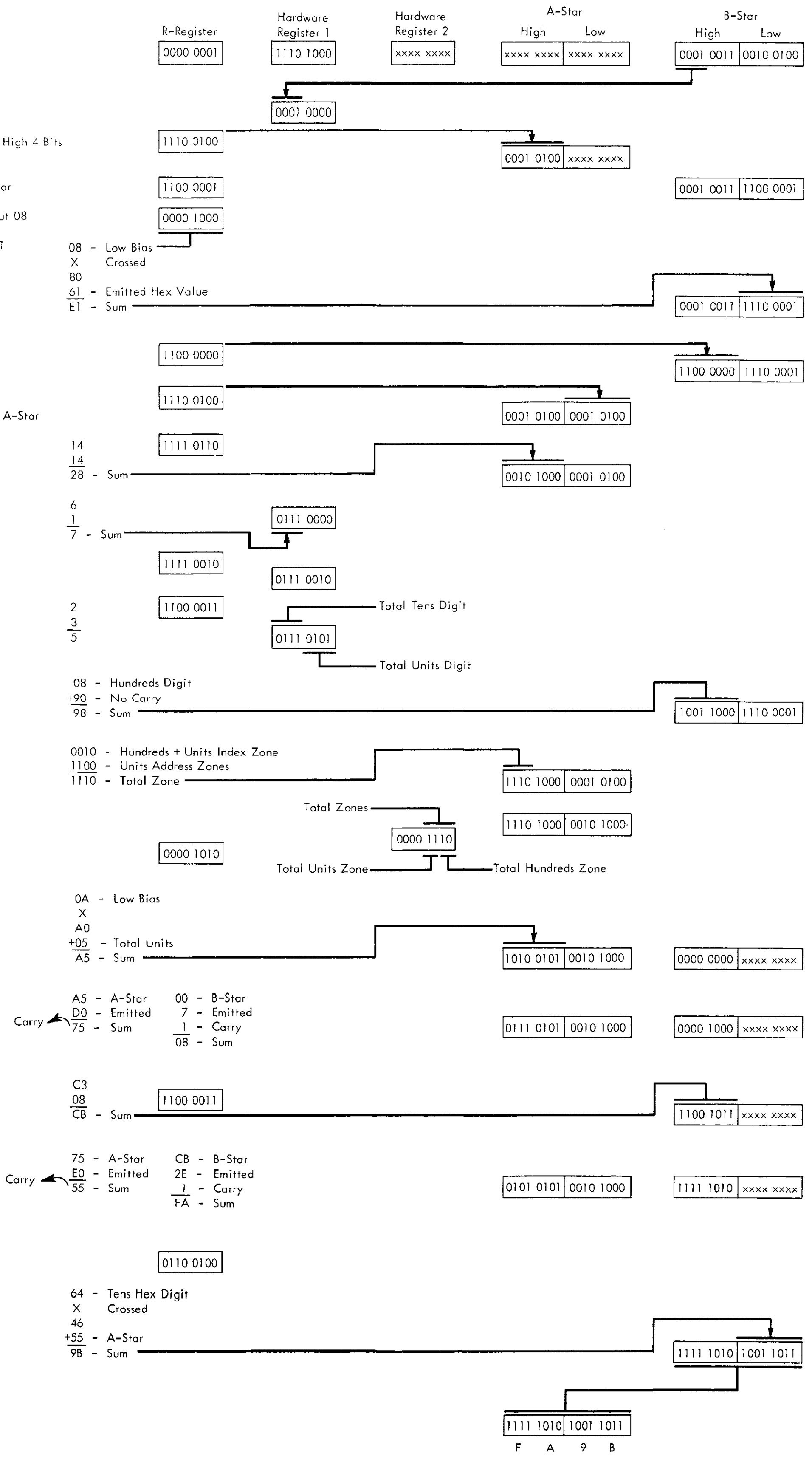
CLF 104 Invalid Address Convert

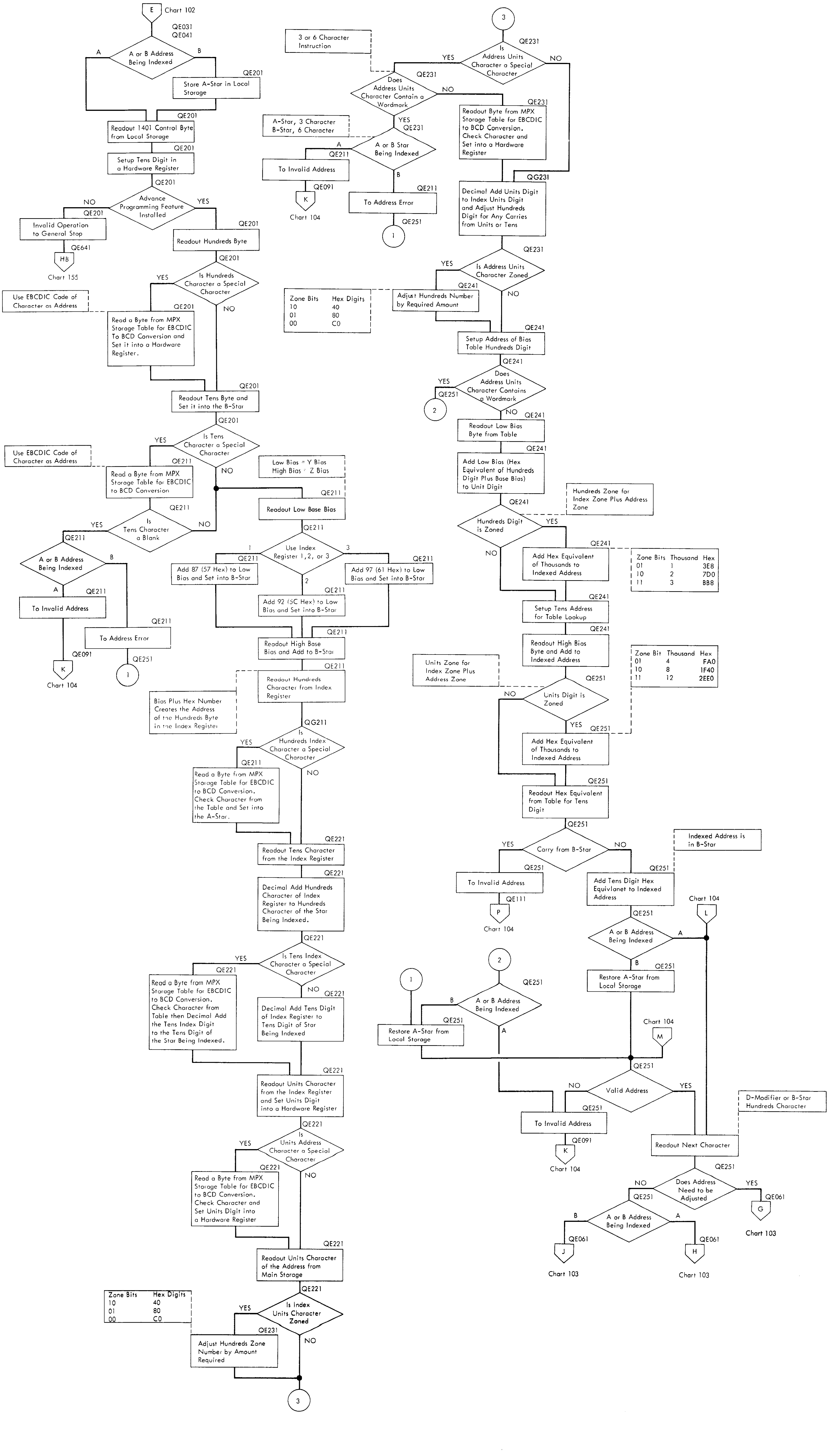


Example for A-Star or B-Star Development with Indexing During I-Phase of 1401 Compatibility Feature on the 2030. Program is for a 16,000 Positions of Storage 1401 and the 2030 Has 65,536 Positions of Storage.

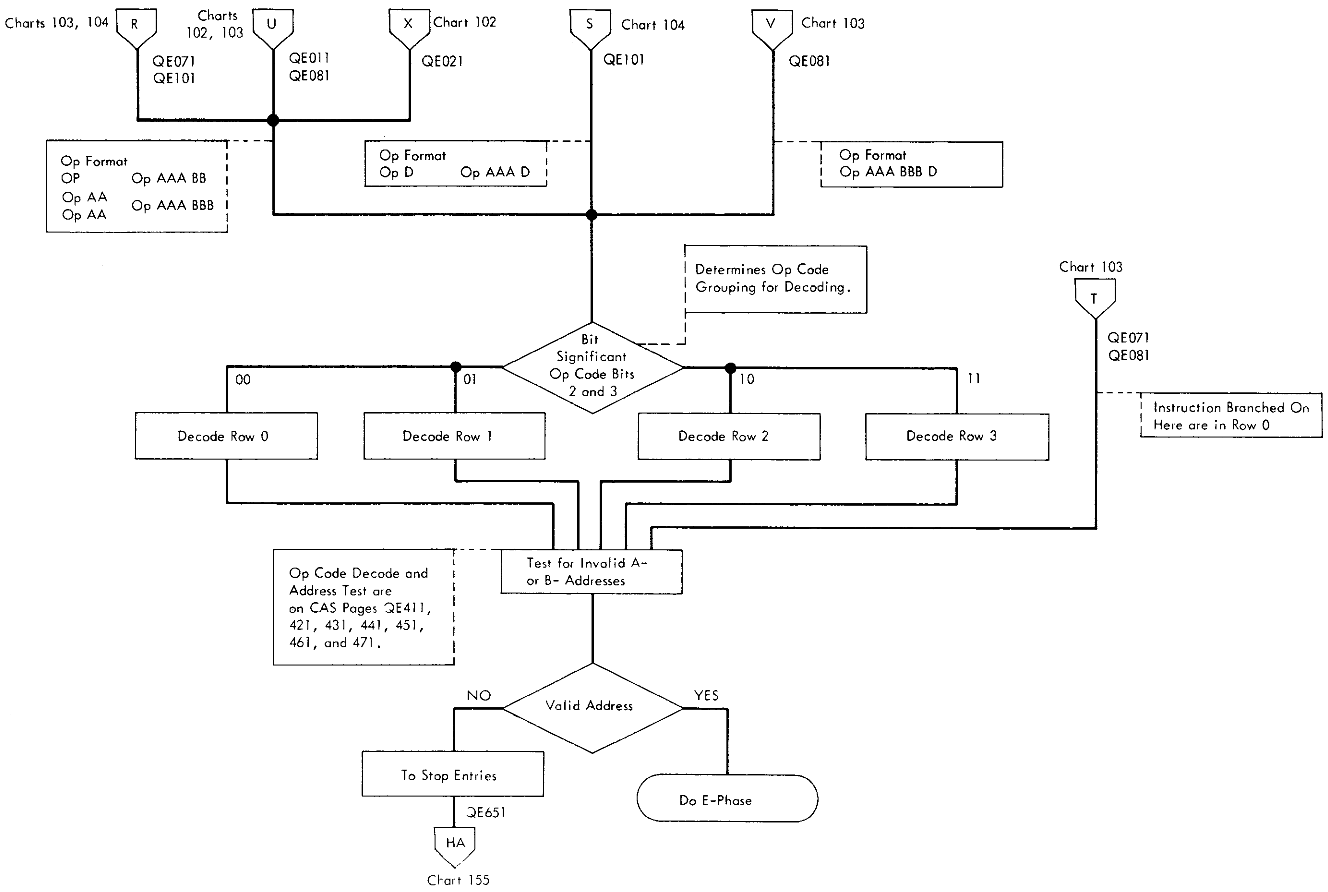
Instruction is A UAC 456 B. Index Register 3 has U62 Stored.

1. Recognize Indexing Required (Tens Byte Zoned).
2. Set Tens Digit from High B-Star to the High 4-Bits of Hardware Register 1.
3. Readout Hundreds Byte Again. If Special Character, Do Table Lookup of BCD Equivalent. For this Example Invert High 4 Bits and Set Byte into High A-Star
4. Readout Tens Byte Again. If Special Character, Do Table Lookup of BCD Equivalent. Set Character into Low B-Star
5. Readout Low Bias from CPU Storage (Address is Emitted). Bias is Crossed in Storage Table. For this Example Readout 08
6. Test Zone Bits of Tens Character to Determine Index Register. For this Example Zone is 00 (Index Register 3). 1401 Index Register 3 Address is 0097 (Hex Equivalent is 61). Add the Hex Value to Low Bias and Set into Low B-Star.
7. Readout High Bias from MPX Storage (Same Address as Low Bias). For this Example Readout C0. Set High Bias into High B-Star
8. Address of Index Register Hundreds Byte in B-Star. Readout Hundreds Byte from Index Register. If Special Character, Do BCD Table Lookup. For this Example Character is U, Invert High 4 Bits and Set into Low A-Star
9. Read Tens Byte from Index Register. Decimal Add Index Hundreds and Address Hundreds Bytes and Set Sum into High A-Star
10. If Tens Byte is Special Character, Do Table Lookup for BCD Equivalent. Decimal Add Index Tens Digit (6) and Address Tens Digit (1) and Set Sum into High 4 Bits of Hardware Register 1.
11. Readout Units Byte from Index Register. If Special Character, Do BCD Table Lookup. For this Example Units Character is 2. Set Index Register Units Digit in the Low 4 Bits of Hardware Register 1.
12. Readout Units Byte for Address. If Special Character, Do BCD Table Lookup. Check Index Units Byte for Zone Bits; For Example Bits are 11 (No Zone) for 0 Thousands. Decimal Add Index Units Digit and Address Units Digit. Set Sum into Low 4 Bits of Hardware Register 1.
13. Decimal Add Any Carry (from Units or Tens Addition) to Combined Hundreds Digit and Set into High B-Star. For this Example No Carry. Emitt a 9 to Force a High Carry Should Low Digits Give a Carry.
14. Check Units Address Zone Bits. For Example Zone Bits are 00 (12,000). This Adds a C (Hex) to Accumulated Zones in High A-Star. Accumulated Zones are in High 4 Bits of the High A Star.
15. Setup Address for Table Lookup of Low Bias + Hundreds Digit Hex Equivalent Digit in Low A-Star. Set Total Zone Digit into Low 4 Bits of Hardware Register 2. Readout Low Bias + Hundreds Byte from CPU Storage. For this Example 0A is Readout Because Byte is Crossed in Storage Table.
16. Add Low Bias Plus Hundreds Digit Hex Equivalent to Total Units Digit and Set into High A-Star. Set a Carry into High B-Star. For this Example No Carry.
17. Test Total Hundreds Zone to Determine Correct Hex Digits to Add to High A-Star and High B-Star. For this Example Bits are 10 (Hex Digits are 7D0). Add D0 to High A-Star, Any Carry from A-Star is Added to High B-Star. Add the 7 to High B-Star.
18. Readout High Bias Plus Hundreds Digit Hex Equivalent from MPX Storage. Setup Total Tens Digit Hex Equivalent Byte Address. Add High Bias Byte to High B-Star.
19. Test Total Units Zone to Determine Correct Hex Digits to Add to High A-Star and High B-Star. For this Example Bits are 11 (Hex Digits are 2EE0). Add E0 to High A-Star, Any Carry from A-Star is Added to High B-Star. Add 2E to High B-Star
20. Readout Total Tens Hex Equivalent Digit from CPI Storage. For this Example Readout 64. Remember Byte is Crossed in Storage Table.
21. Add Tens Hex Byte (Uncrossed) to High A-Star and Set Sum in Low B-Star. Any Carry is Added to High B-Star.
22. **End of Indexing this Address Continue I-Phase. If Address Being Indexed was the A-Star, Transfer A-Star to B-Star. If Address Being Indexed was B-Star, Restore the A-Star from Local Storage. For Example Set B-Star into A Star.**





CLF 106 1400 Address Indexing

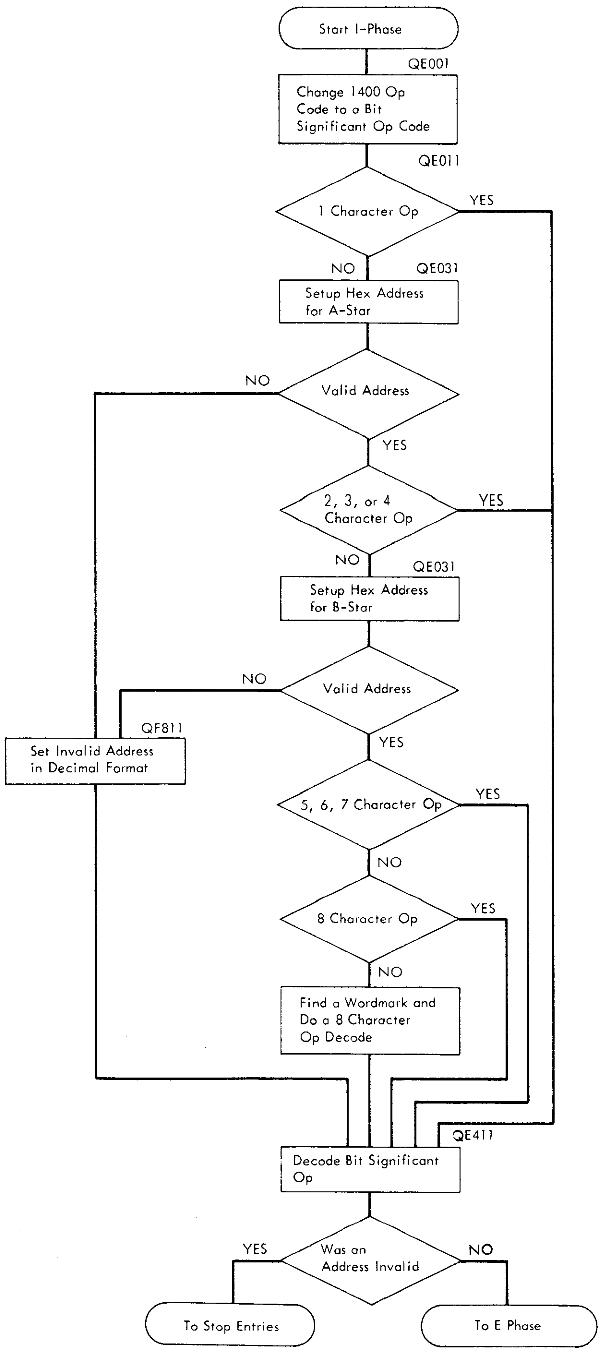


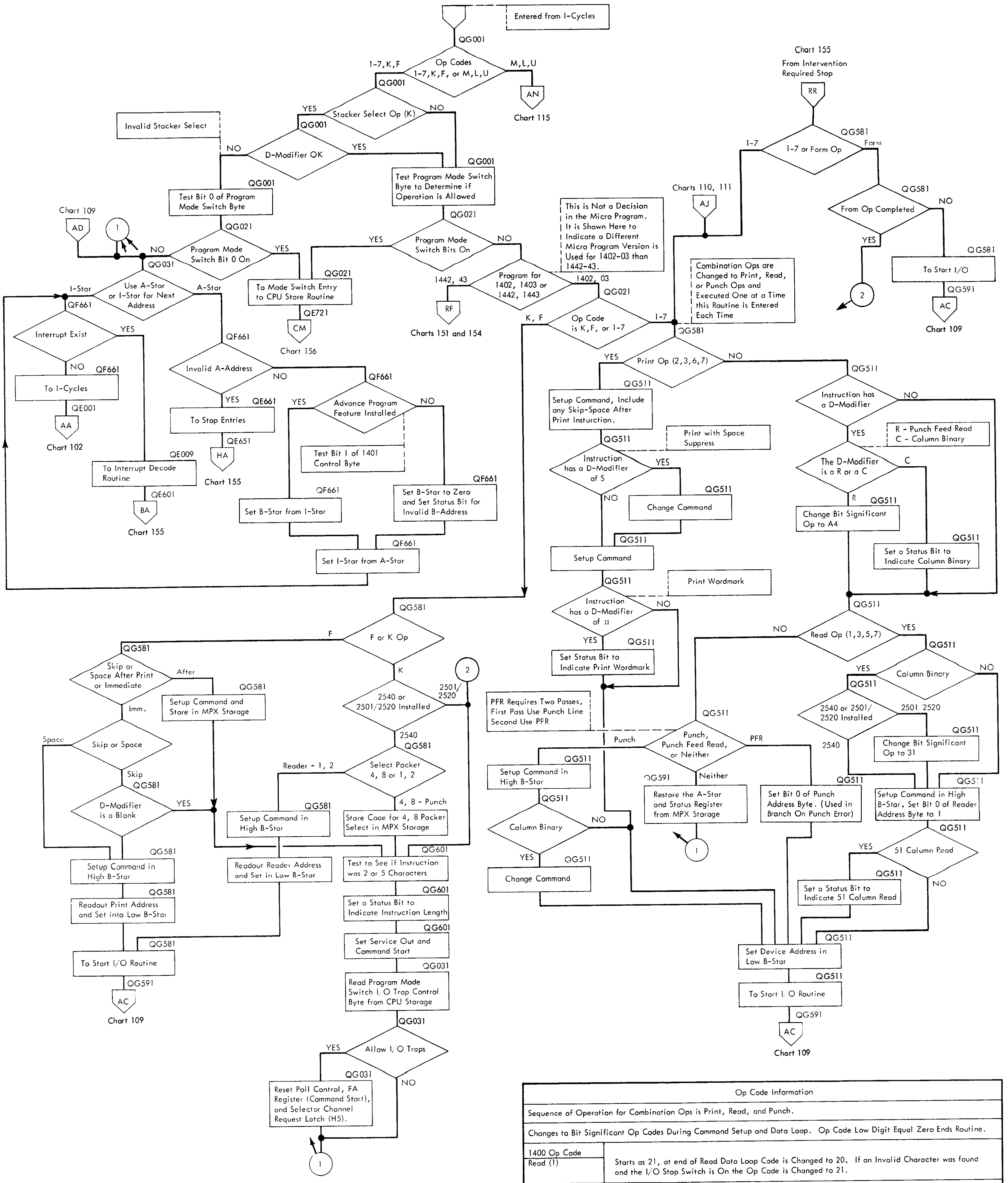
Row	Significant Op Byte	Instruction Name	1401 Sym.	CLF Ch.	Row	Significant Op Byte	Instruction Name	1401 Sym.	CLF Ch.	Row	Significant Op Byte	Instruction Name	1401 Sym.	CLF Ch.	Row	Significant Op Byte	Instruction Name	1401 Sym.	CLF Ch.
0	0000 0010	**Halt**	·		1	0001 0010	Move Digit	D		2	0010 0000	Control (Note 1)	U	108	3	0011 0100	Invalid		155
0	0000 0100	Set Wordmark	,		1	0001 0011	Move Zone	Y		2	0010 0001	*Read	1	108	3	0011 1010	Branch on WM/Zone	V	
0	0000 0101	Clear			1	0001 0100	Address Modify	#		2	0010 0010	*Print	2	108	3	0011 1011	Bit Test	W	
0	0000 0110	No Op	N		1	0001 0101	Clear Wordmark	⌑		2	0010 0011	*Read-Print	3	108	3	1011 0001	Store A-Star	H	
0	0000 0110	Early Read (Note 2)	8		1	0001 0110	Edit	E		2	0010 0100	*Punch	4	108	3	1111 0001	Store B-Star	Q	
0	0000 0110	Early Punch (Note 2)	9		1	0001 0111	Move Zero Suppress	Z		2	0010 0101	*Read-Punch	5	108					
0	0000 0110	Branch	B		1	0001 1000	Add	A		2	0010 0110	*Print-Punch	6	108					
0	1000 0000	Move (Note 1)	M		1	0001 1001	Subtract	S		2	0010 0111	*Read-Punch-Print	7	108					
0	1000 0000	Move, Column Bin.	M		1	0001 1010	Multiply			2	0010 0100	*Punch Feed Read	4R	108					
0	1000 0000	Move, Sterling	M		1	0001 1011	Divide	%		2	0010 1001	Stacker Select	K	108					
					1	0001 1100	Reset Add	?		2	0010 1010	Form	F	108					
					1	0001 1101	Reset Subtract	!		2	0010 1100	Control	U	115					
					1	0001 1110	Move Record	P		2	0010 1101	Move	M	115					
					1	0001 1111	Compare	C		2	0010 1110	Load	L	115					
					1	1001 0000	Load (Note 1)	L											
												*Note 3							

Note 1. If I/O Operation Change Op Code L to 0010 1110, M to 0010,1101, and U to 0010 1100

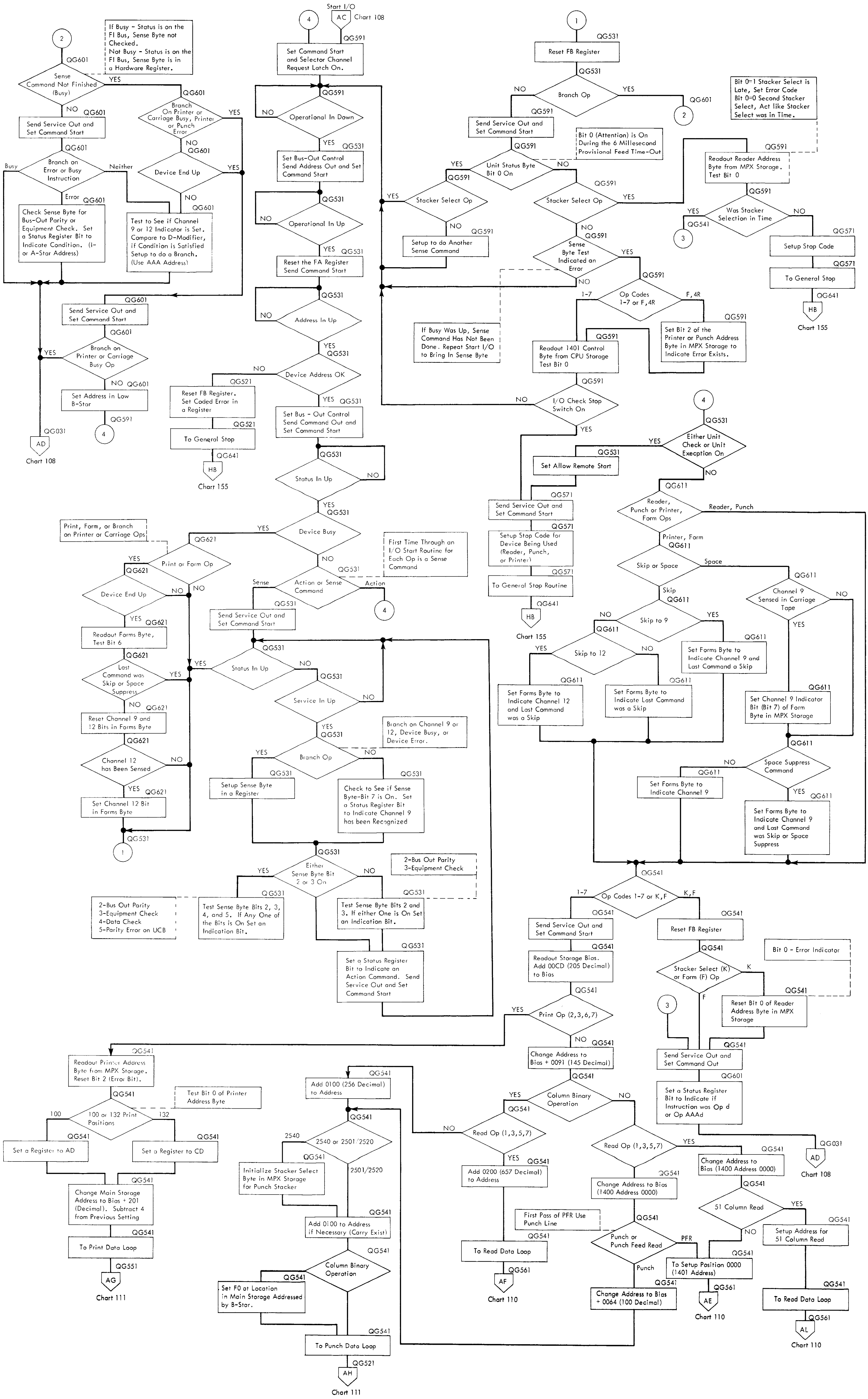
Note 2. These Op Codes are Handled as No Ops.

Note 3. This Op Codes Start with Bit Significant Byte as Shown. As the Op is Performed the Op Code is Changed to Indicate the First Pass or the Op Code has a D-Modifier, Also to Indicate when a Combination Op is Finished.

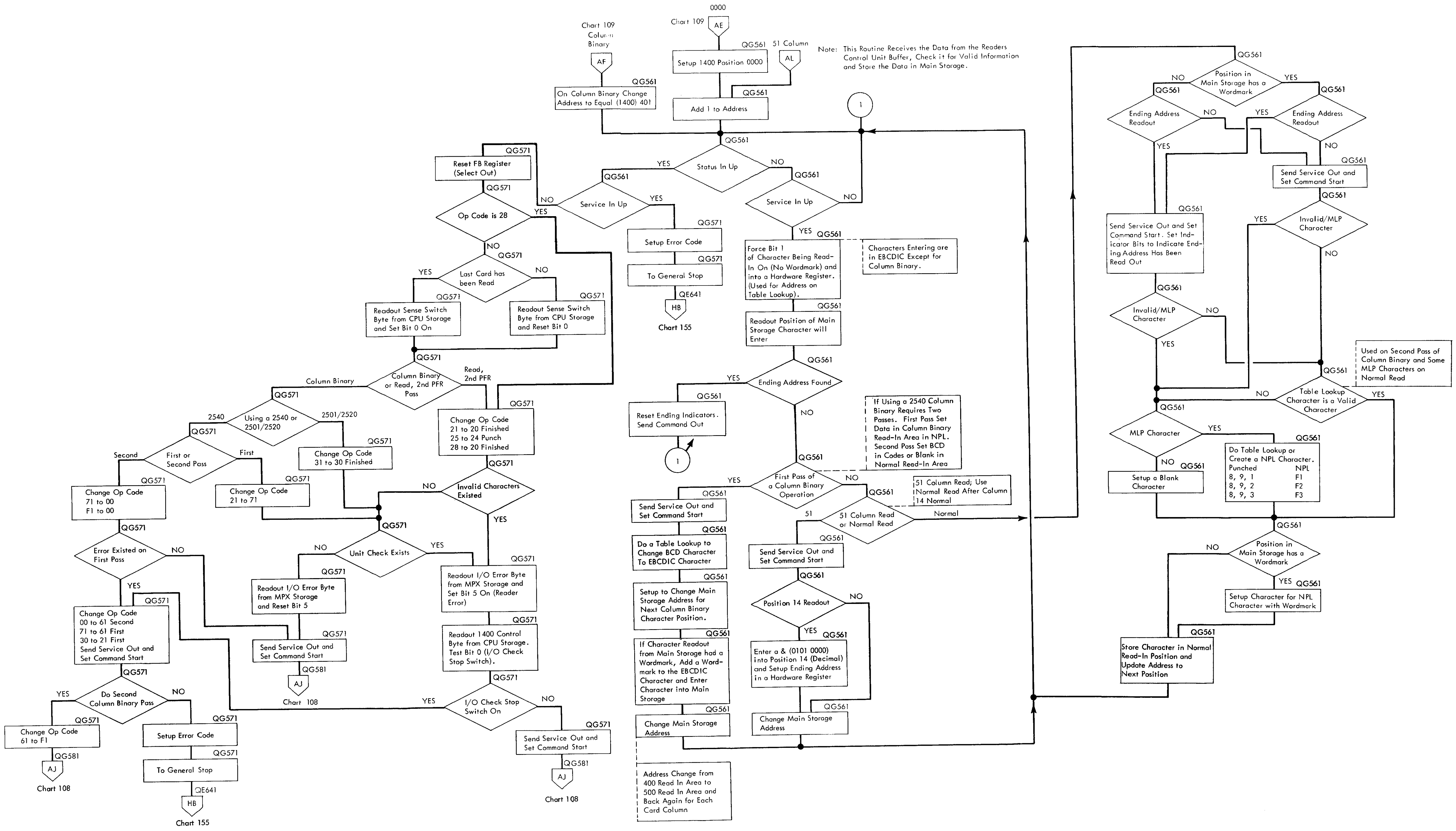




Op Code Information	
Sequence of Operation for Combination Ops is Print, Read, and Punch.	
Changes to Bit Significant Op Codes During Command Setup and Data Loop. Op Code Low Digit Equal Zero Ends Routine.	
1400 Op Code Read (1)	Starts as 21, at end of Read Data Loop Code is Changed to 20. If an Invalid Character was found and the I/O Stop Switch is On the Op Code is Changed to 21.
Print (2)	Starts as 22, at End of Print Data Loop Code is Changed to 20. If Operational In Drops During the Data Loop Code Stays 22 and a Stop Occurs.
Print, Read (3)	Starts as 23, at End of Print Data Loop Changes to 21 unless Operational In Drops During Loop. The 21 Causes a Read Routine to Start. End of Read Data Loop Change to 20 Unless Error Exists.
Punch (4) Punch Col Bin (4C)	Starts as 24, at End of Punch Data Loop Code Changes to 20 Unless Operational In Dropped During Data Loop or Last Card Punch had an Error Then Op Code Stays 24.
Read, Punch (5)	Starts as 25, End of Read Data Loop Code Changes to 24. If an Error has Occurred and I/O Stop Switch is On Op Code is Changed to 25 Again and Stops. No Error the 24 Cause a Punch Operation. If Operational End Drops During Data Loop or Last Card Punched had an Error Code Stays 24, Otherwise Changes to 20.
Print, Punch (6)	Starts as 26, End of Print Data Loop Unless an Error Occurs the Op Code Changes to 24 and a Punch Operation is Performed. The 24 Changes to 20 Unless Operational In Drops During the Data Loop or Last Card Punch was in Error.
Print, Read, Punch (7)	Starts as 27, End of Print Data Loop the Op Code Changes to 25 Unless an Error has Occurred. The 25 Starts a Read Operation; at the End of the Read Data Loop the Op Code is Changed to 24 Unless an Error has Occurred and the I/O Stop Switch is On. The 24 Causes a Punch Operation to Start, at the End of the Punch Data Loop the Op Code Changes to 20 Unless an Error has Occurred
Punch Feed Read (4R)	Starts as 24, the d-Modifier Causes the Op Code to Change to A4 then a Normal Punch Routine is Started. At the End of the Punch Data Loop if an Error Situation has Occurred the Op Code Stays as A4 and a Stop Occurs. No Error, the A4 Changes to 28 which Cause a Read Operation to be Performed. If no Errors Occur During Read Operation the Op Code Changes to 20. If Error Occurred and I/O Switch On, Stop with Op Code 20.
Read Col Binary (1C-2540)	Starts as 21. Note: Column Binary Read on a 2540 Requires Two Passes through the Read Data Loop: At the End of the First Pass (Information Entered into 401-480 and 501-580) the Op Code is Changed to 71, if an Error Occurred on the First Pass and the I/O Stop Switch is On the Op Code is Changed to 61 and then F1. No Error or I/O Stop Switch Off Op Code Stays 71. The Second Pass is Made Entering Some Information or Blanks into 0001-0080. Now if the Op Code had Changed to F1 it will be Changed to 61 and a Stop Occurs, but if the Op Code was 71 Entering the Second Pass the Op Code Changes to 00.
Read Col Binary (1C-2501/2520)	Starts as 21 and is Changed to 31 to Indicate 2501/2520 which Only take One Pass to Enter Information (Only Enters 401-480 and 501 to 580). At End of Read Data Loop for Column Binary First Pass the Op Code is Changed to 30. If an Error Occurred During the First Pass the and the I/O Stop Switch is On the Op Code is Changed to 21 and a Stop Occurs.

If Busy - Status is on the FI Bus, Sense Byte not Checked. Not Busy - Status is on the FI Bus, Sense Byte is in a Hardware Register.
2
QG601
Sense Command Not Finished (Busy)
YES
NO
Send Service Out and Set Command Start
Branch on Error or Busy Instruction
Busy
Neither
Error
Check Sense Byte for Bus-Out Parity or Equipment Check. Set a Status Register Bit to Indicate Condition. (I- or A-Star Address)
Branch On Printer or Carriage Busy, Printer or Punch Error
Device End Up
Test to See if Channel 9 or 12 Indicator is Set. Compare to D-Modifier, if Condition is Satisfied Setup to do a Branch. (Use AAA Address)
Send Service Out and Set Command Start
Branch on Printer or Carriage Busy Op
Set Address in Low B-Star
QG591
4
QG031
AD
Chart 108
Start I/O
AC
Chart 108
QG591
Set Command Start and Selector Channel Request Latch On.
Operational In Down
QG531
Set Bus-Out Control Send Address Out and Set Command Start
Operational In Up
Reset the FA Register Send Command Start
Address In Up
Device Address OK
QG521
Reset FB Register. Set Coded Error in a Register
To General Stop
QG641
HB
Chart 155
Set Bus - Out Control Send Command Out and Set Command Start
Status In Up
Device Busy
1
Reset FB Register
Branch Op
Send Service Out and Set Command Start
Unit Status Byte Bit 0 On
Bit 0 (Attention) is On During the 6 Millesecond Provisional Feed Time-Out
Stacker Select Op
Setup to do Another Sense Command
Sense Byte Test Indicated an Error
If Busy Was Up, Sense Command Has Not Been Done. Repeat Start I/O to Bring In Sense Byte
Op Codes 1-7 or F,4R
1-7
F,4R
Readout 1401 Control Byte from CPU Storage Test Bit 0
Set Bit 2 of the Printer or Punch Address Byte in MPX Storage to Indicate Error Exists.
I/O Check Stop Switch On
Bit 0=1 Stacker Select is Late, Set Error Code Bit 0=0 Second Stacker Select, Act like Stacker Select was in Time.
Readout Reader Address Byte from MPX Storage. Test Bit 0
Was Stacker Selection in Time
3
QG541
QG571
Setup Stop Code
To General Stop
Print, Form, or Branch on Printer or Carriage Ops
QG621
Print or Form Op
Device End Up
Readout Forms Byte, Test Bit 6
Last Command was Skip or Space Suppress
Reset Channel 9 and 12 Bits in Forms Byte
Channel 12 has Been Sensed
Set Channel 12 Bit in Forms Byte
First Time Through an I/O Start Routine for Each Op is a Sense Command
Action or Sense Command
Sense
Action
Send Service Out and Set Command Start
Status In Up
Service In Up
Branch on Channel 9 or 12, Device Busy, or Device Error.
Branch Op
Setup Sense Byte in a Register
Check to See if Sense Byte-Bit 7 is On. Set a Status Register Bit to Indicate Channel 9 has been Recognized
Either Sense Byte Bit 2 or 3 On
2-Bus Out Parity 3-Equipment Check
2-Bus Out Parity 3-Equipment Check 4-Data Check 5-Parity Error on UCB
Test Sense Byte Bits 2, 3, 4, and 5. If Any One of the Bits is On Set an Indication Bit.
Test Sense Byte Bits 2 and 3. If either One is On Set an Indication Bit.
Set a Status Register Bit to Indicate an Action Command. Send Service Out and Set Command Start
Either Unit Check or Unit Exception On
Set Allow Remote Start
Send Service Out and Set Command Start
Setup Stop Code for Device Being Used (Reader, Punch, or Printer)
To General Stop Routine
Reader, Punch or Printer, Form Ops
Reader, Punch
Printer, Form
Skip or Space
Space
Skip
Skip to 9
Skip to 12
Set Forms Byte to Indicate Channel 9 and Last Command a Skip
Set Forms Byte to Indicate Channel 12 and Last Command was a Skip
Set Forms Byte to Indicate Last Command was a Skip
Channel 9 Sensed in Carriage Tape
Set Channel 9 Indicator Bit (Bit 7) of Form Byte in MPX Storage
Space Suppress Command
Set Forms Byte to Indicate Channel 9
Set Forms Byte to Indicate Channel 9 and Last Command was Skip or Space Suppress
Op Codes 1-7 or K,F
K,F
Send Service Out and Set Command Start
Readout Storage Bias. Add 00CD (205 Decimal) to Bias
Reset FB Register
Bit 0 - Error Indicator
Stacker Select (K) or Form (F) Op
K
F
Reset Bit 0 of Reader Address Byte in MPX Storage
Send Service Out and Set Command Out
QG601
Set a Status Register Bit to Indicate if Instruction was Op d or Op AAAd
Print Op (2,3,6,7)
Change Address to Bias + 0091 (145 Decimal)
Column Binary Operation
Read Op (1,3,5,7)
Add 0200 (657 Decimal) to Address
To Read Data Loop
QG561
AF
Chart 110
Read Op (1,3,5,7)
Change Address to Bias (1400 Address 0000)
Change Address to Bias (1400 Address 0000)
51 Column Read
Setup Address for 51 Column Read
To Read Data Loop
AL
Chart 110
First Pass of PFR Use Punch Line
Punch or Punch Feed Read
PFR
Punch
To Setup Position 0000 (1401 Address)
AE
Chart 110
Change Address to Bias + 0064 (100 Decimal)
Readout Printer Address Byte from MPX Storage. Reset Bit 2 (Error Bit).
Test Bit 0 of Printer Address Byte
100 or 132 Print Positions
100
132
Set a Register to AD
Set a Register to CD
Change Main Storage Address to Bias + 201 (Decimal). Subtract 4 from Previous Setting
To Print Data Loop
QG551
AG
Chart 111
Add 0100 (256 Decimal) to Address
2540 or 2501/2520
2540
2501/2520
Initialize Stacker Select Byte in MPX Storage for Punch Stacker
Add 0100 to Address if Necessary (Carry Exist)
Column Binary Operation
Set F0 at Location in Main Storage Addressed by B-Star.
To Punch Data Loop
QG521
AH
Chart 111

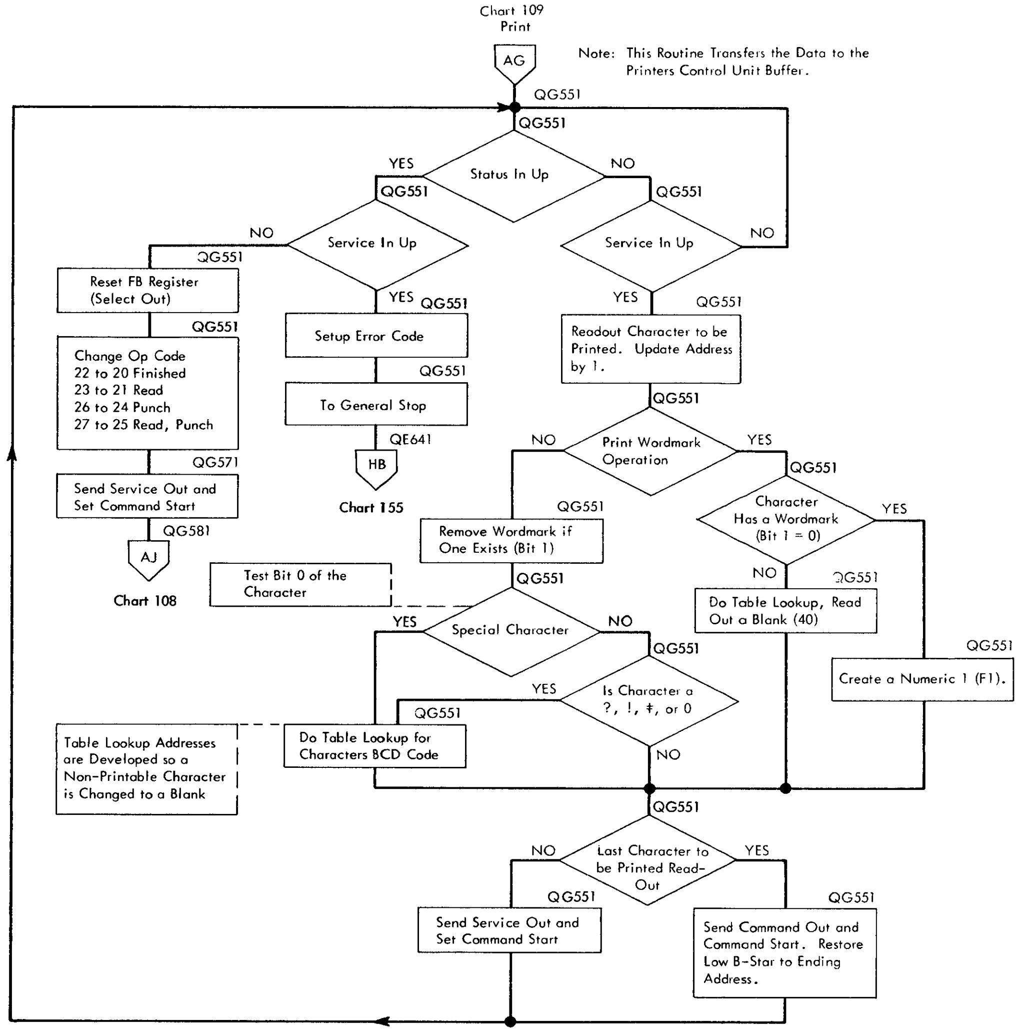
CLF 110 1402 Read Data Loop



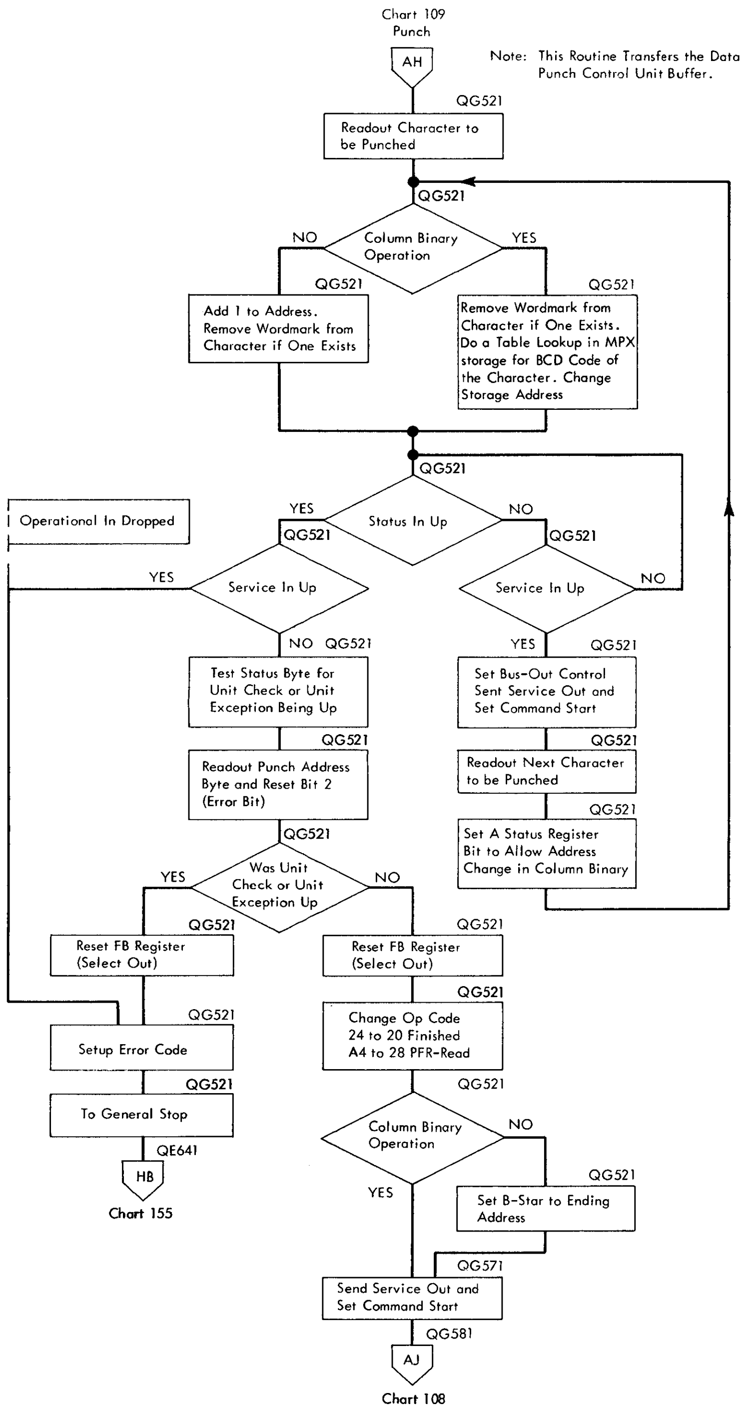
Note: This Routine Receives the Data from the Readers Control Unit Buffer, Check it for Valid Information and Store the Data in Main Storage.

CLF 111

1402 Punch 1403 Print Data Loops

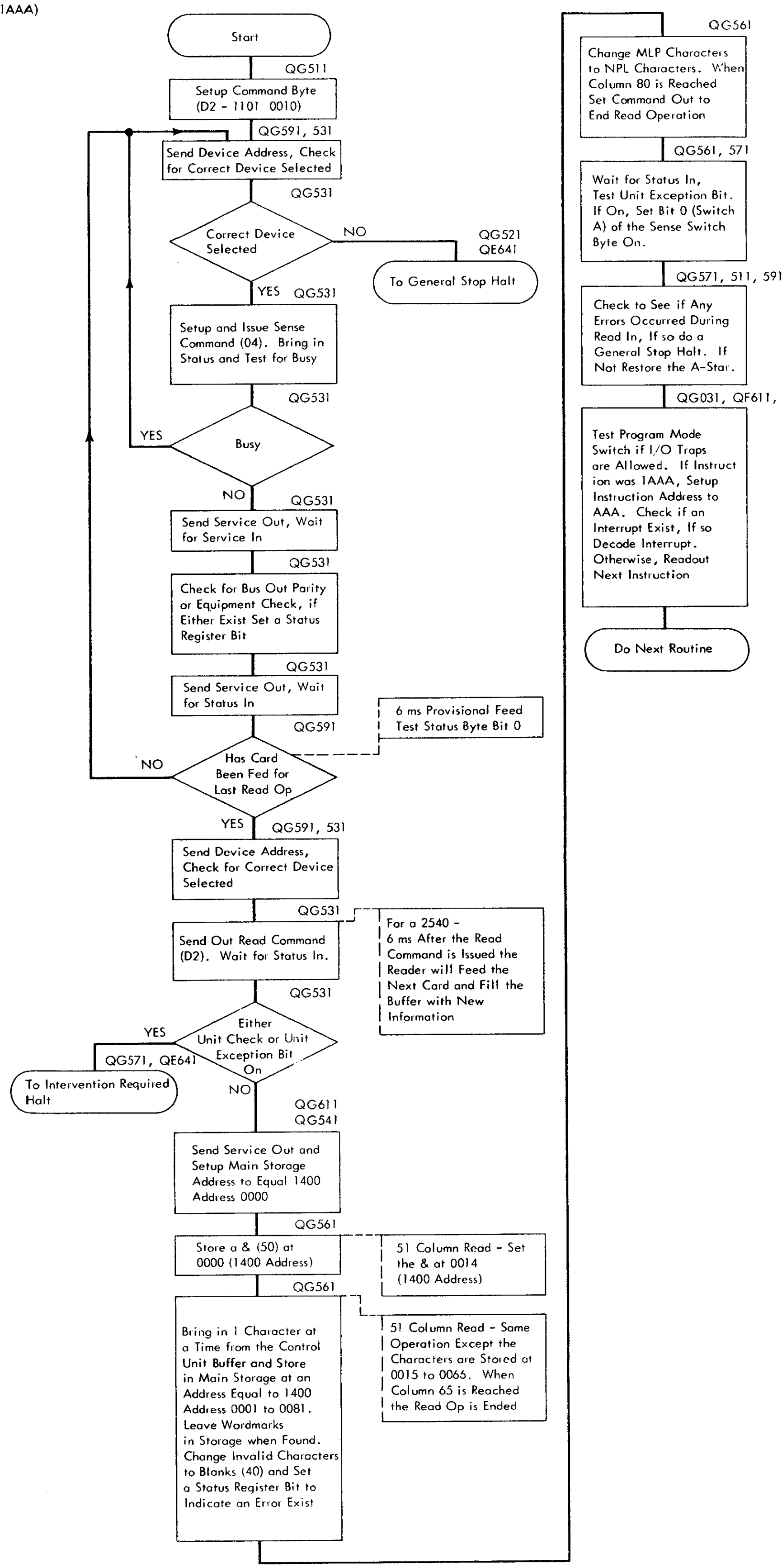


Note: This Routine Transfers the Data to the Printers Control Unit Buffer.



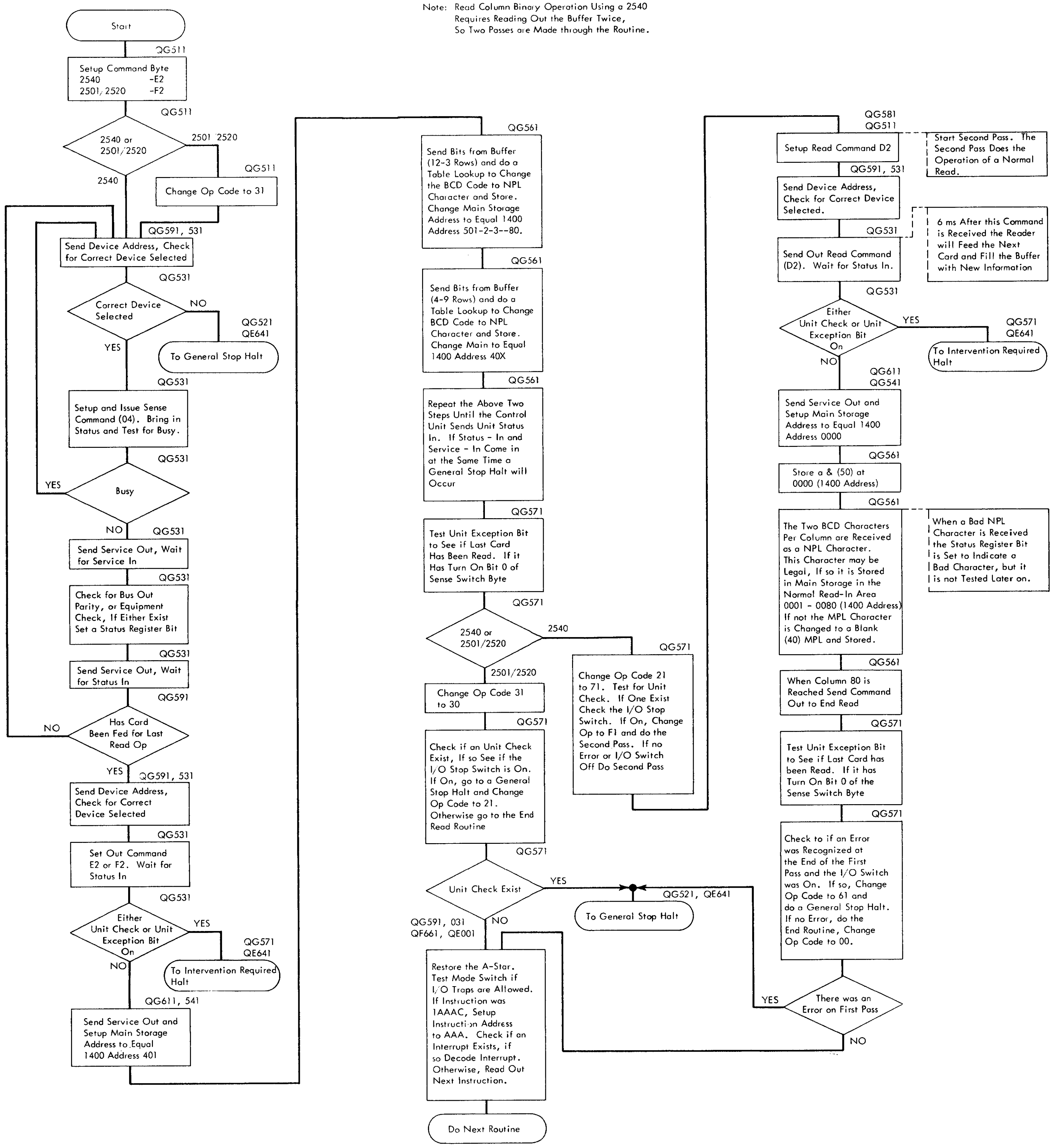
Note: This Routine Transfers the Data to the Punch Control Unit Buffer.

Instruction - Read (1 or 1AAA)
Bit Significant Op is 21

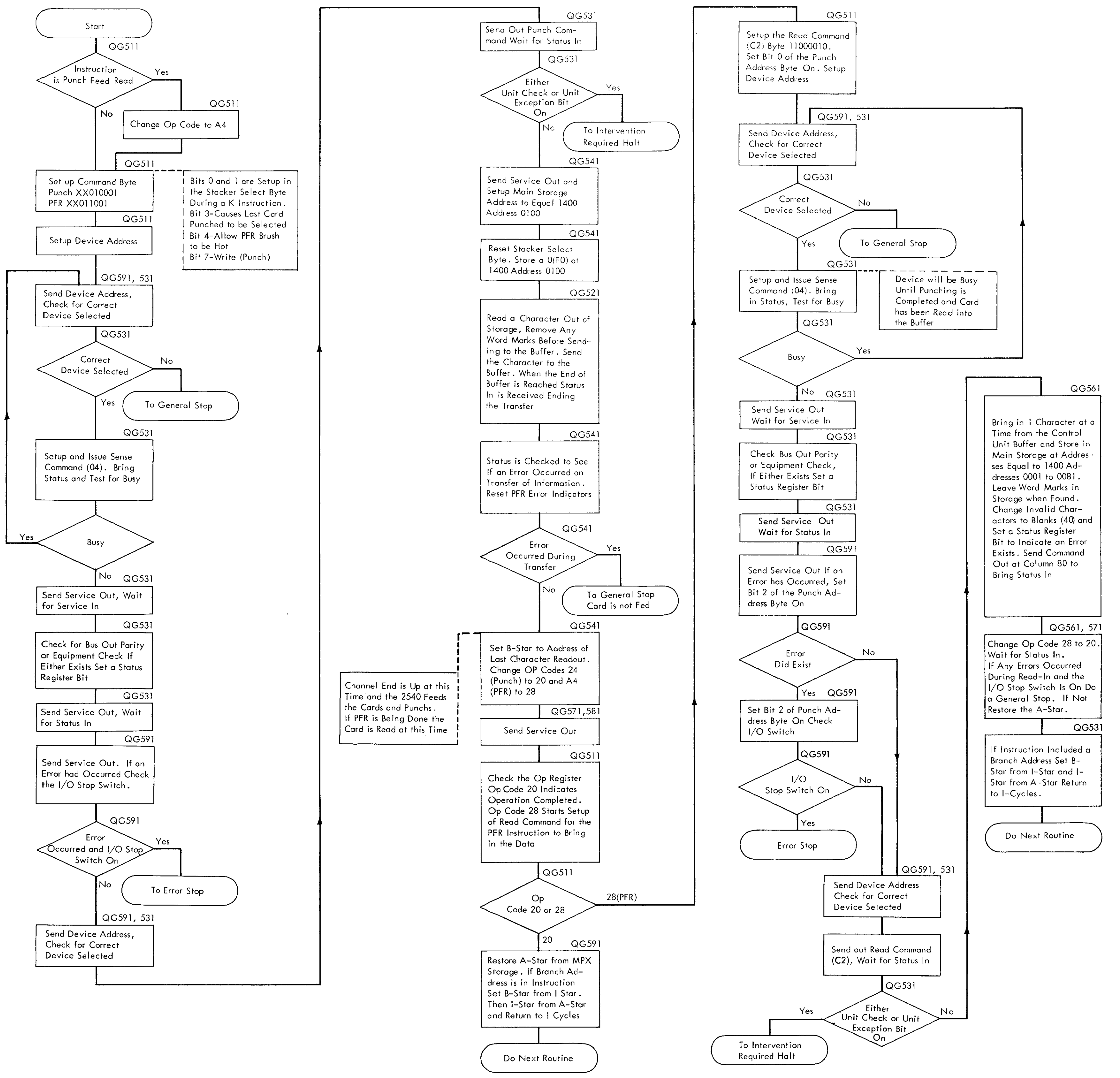


Instruction Read
Column Binary
(1C or 1AAAC)
Bit Significant
Op is 21

Note: Read Column Binary Operation Using a 2540 Requires Reading Out the Buffer Twice, So Two Passes are Made through the Routine.

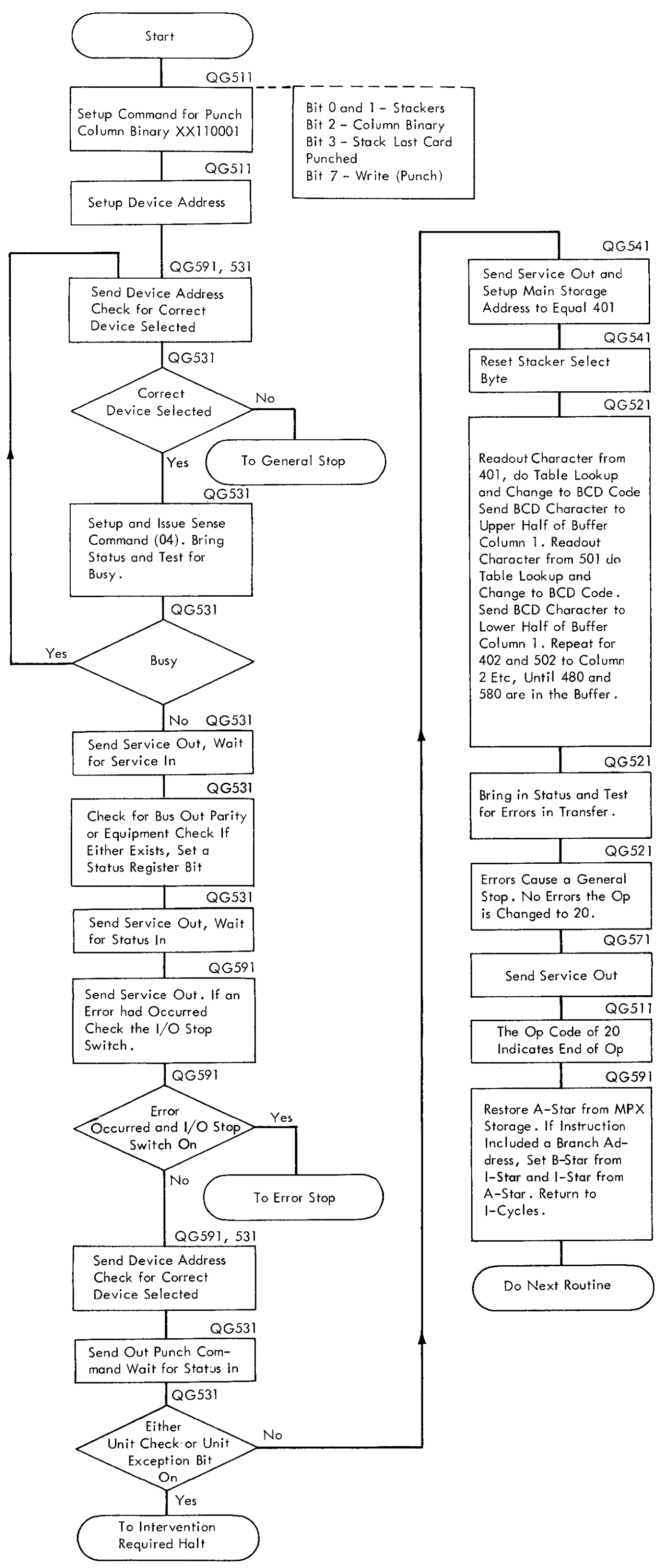


Instruction - 4, 4AAA_, 4R, or
4AAAR_.
Device - 2540

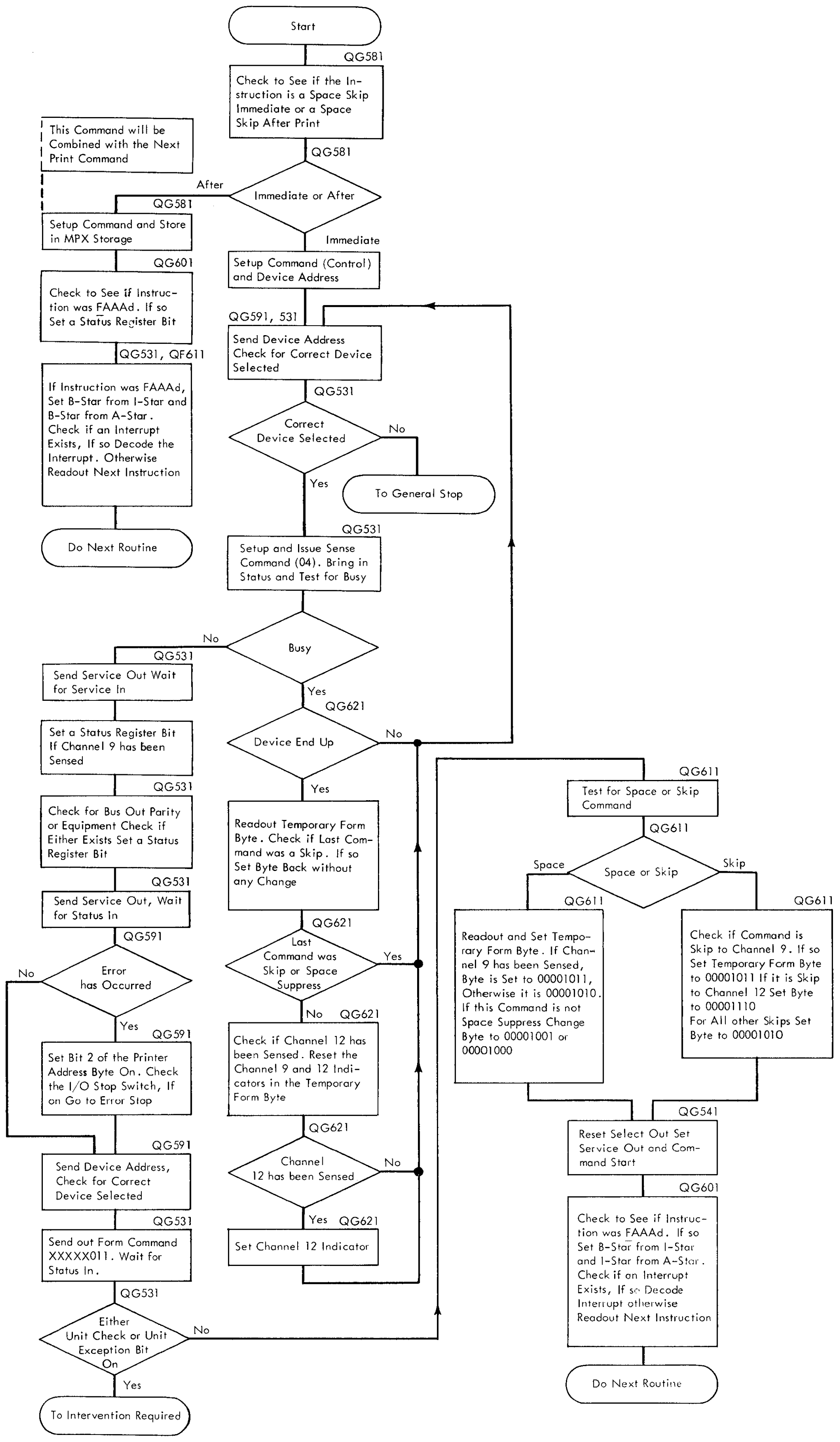


Instruction - 4C_ or 4AAAC_

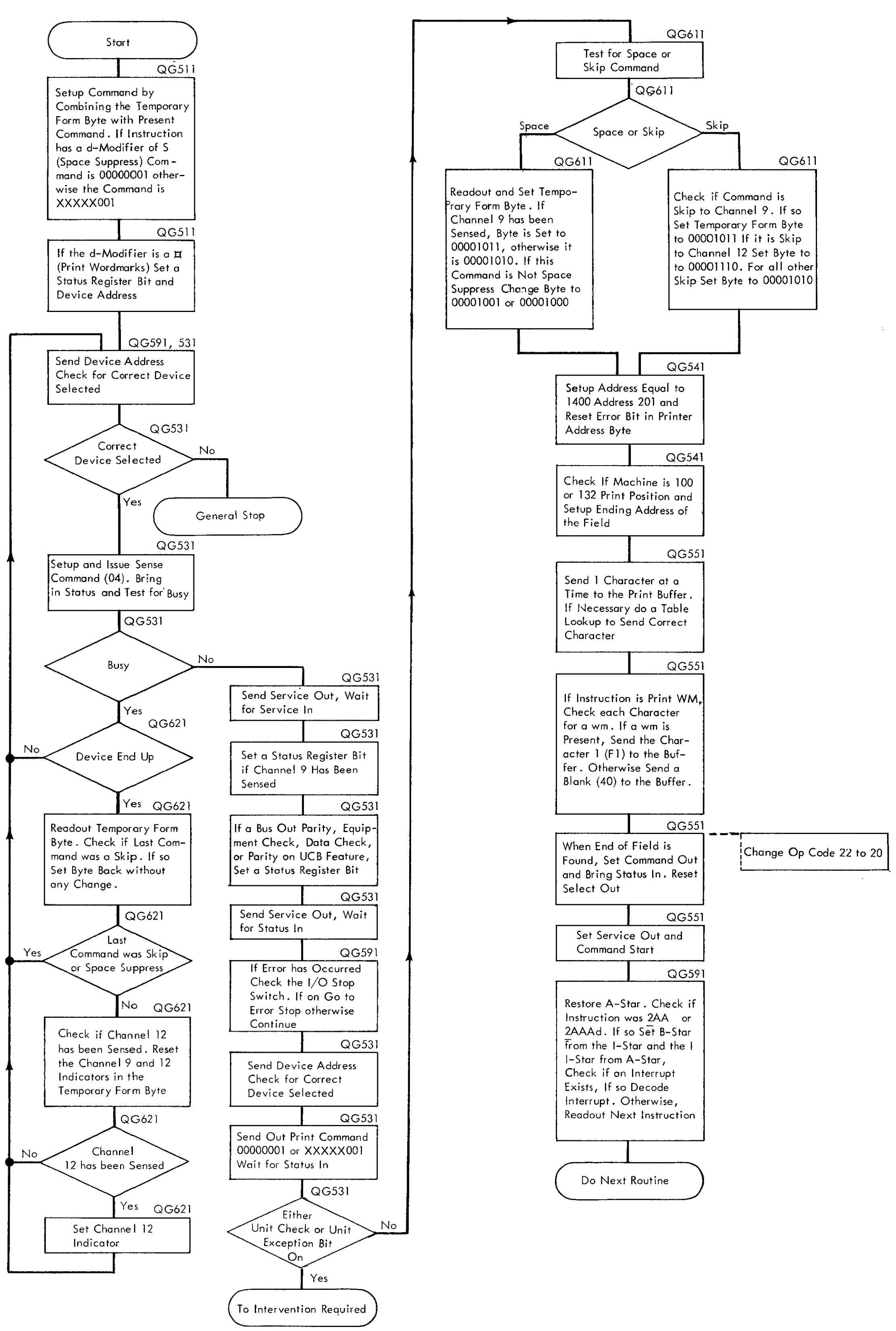
Device - 2540

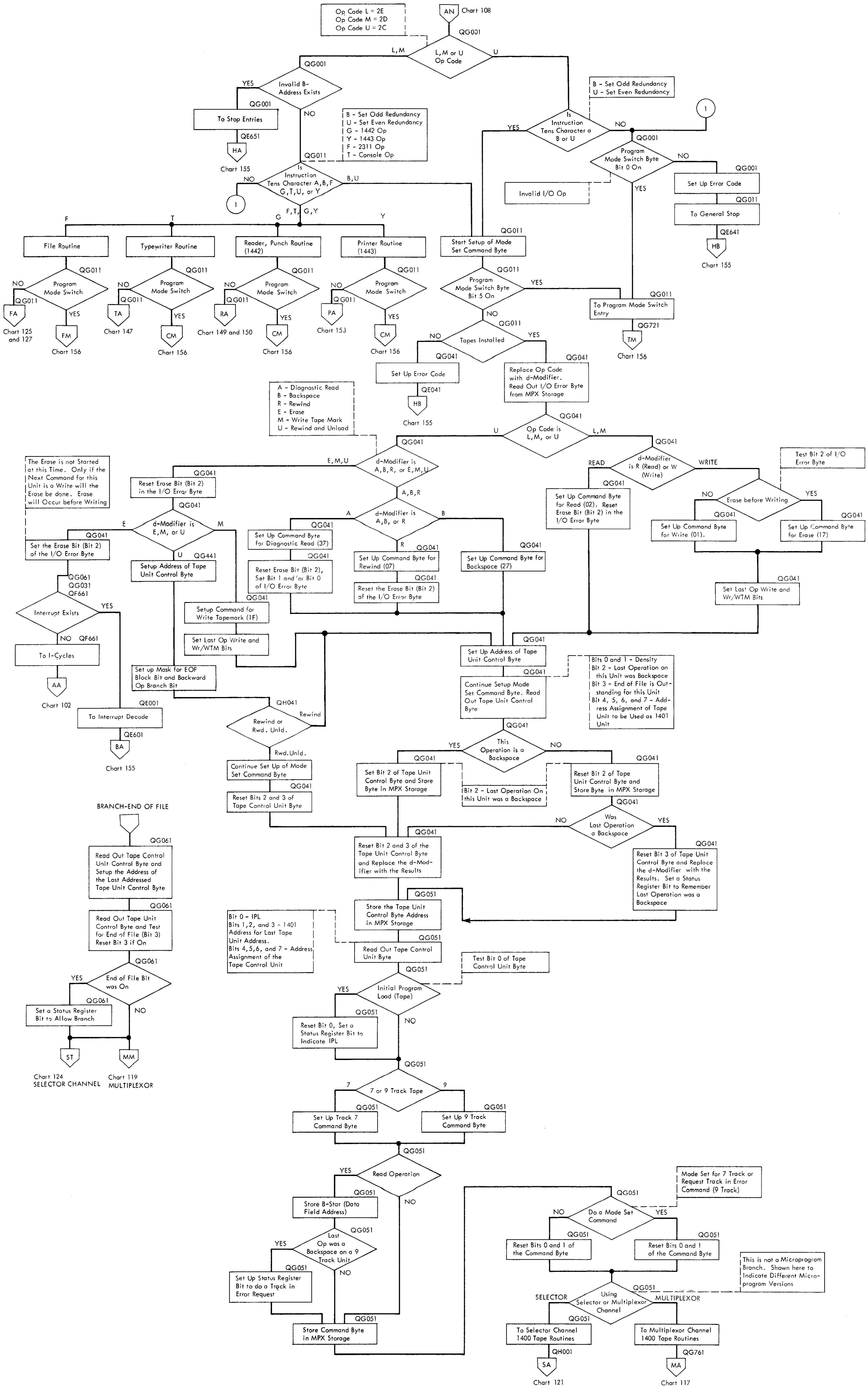


Forms Instruction Fd or FAAAd
Device 1403

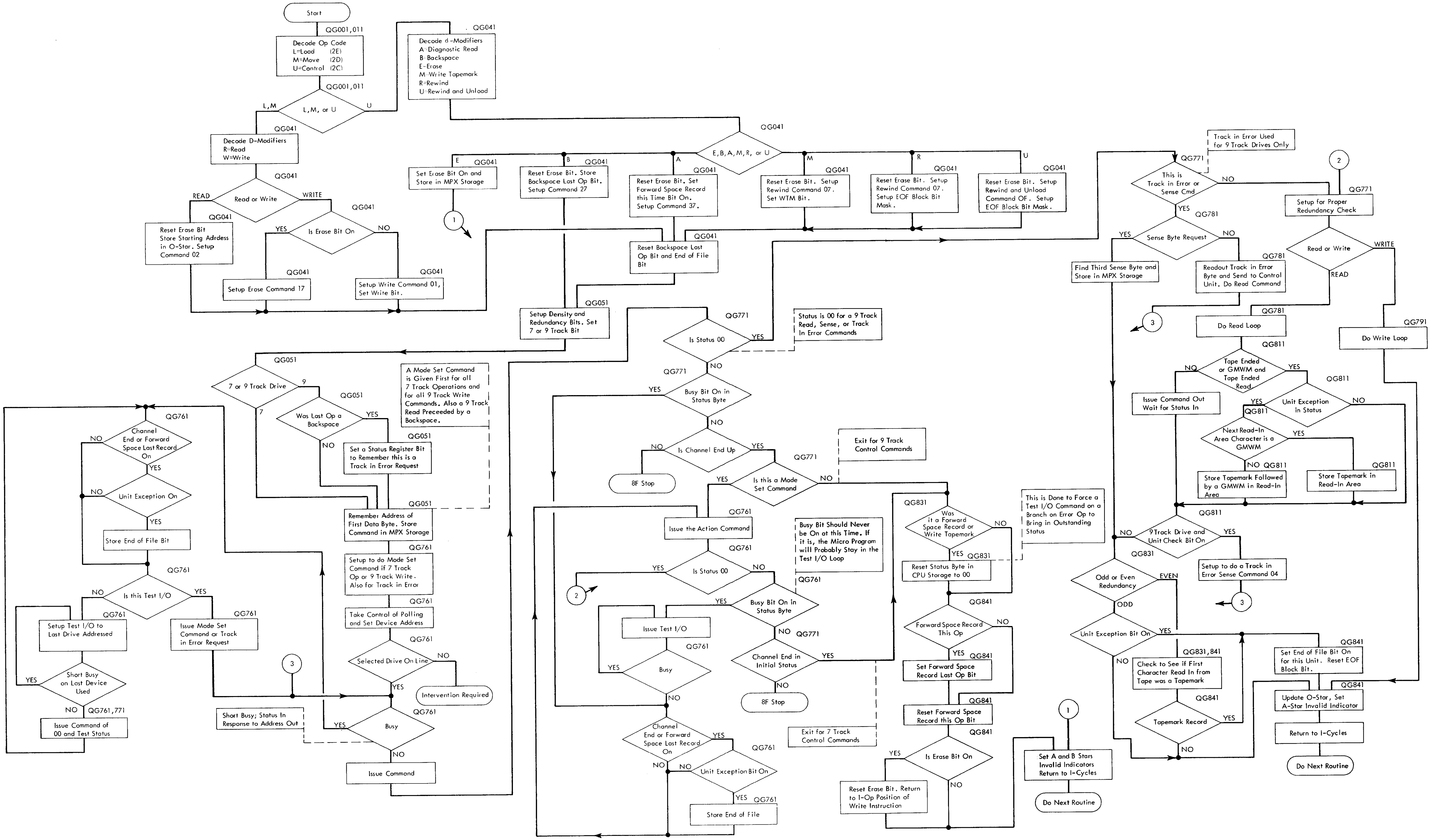


Print Instruction - 2, 2d, 2AAA, or 2AAAd
Device 1403

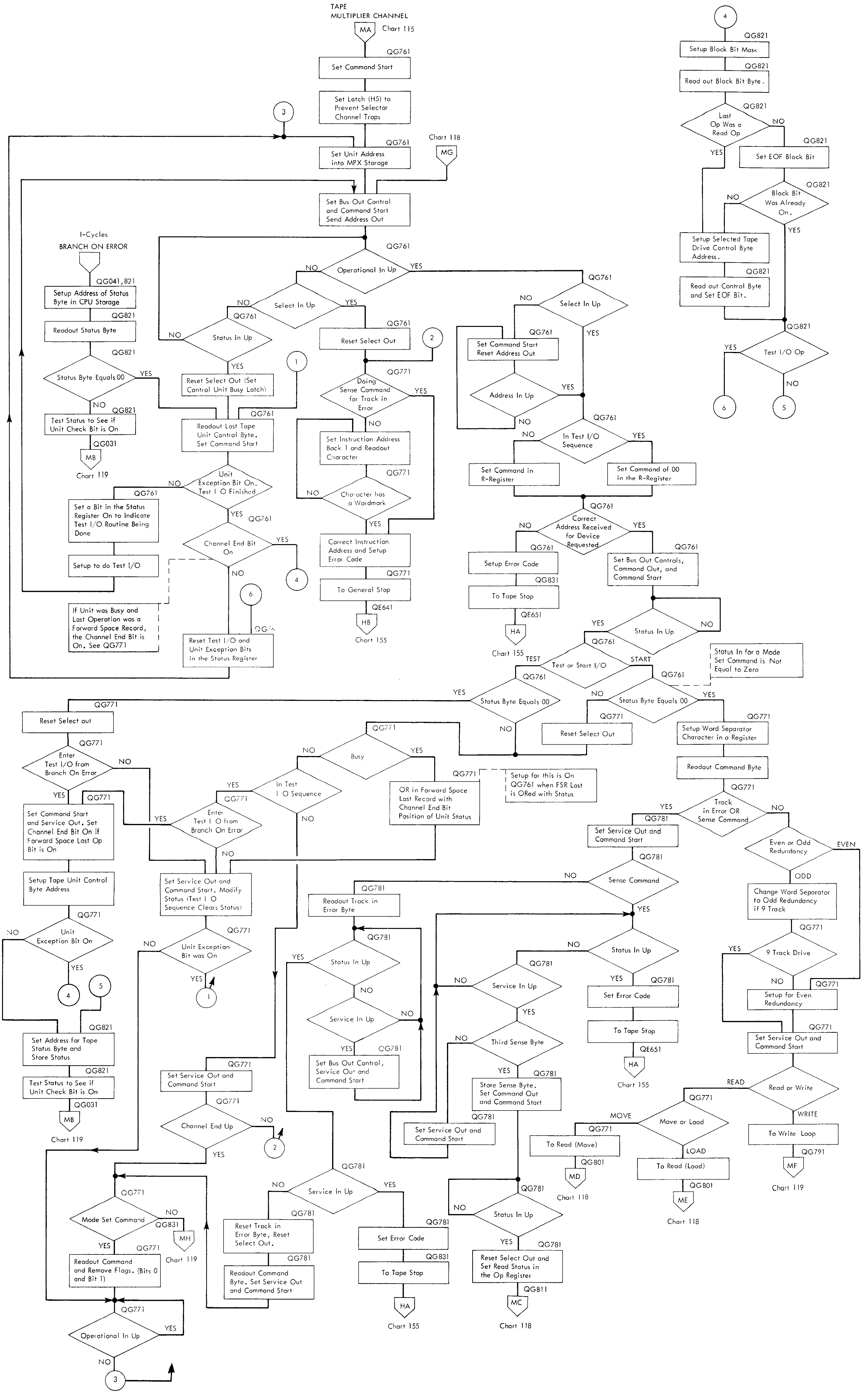




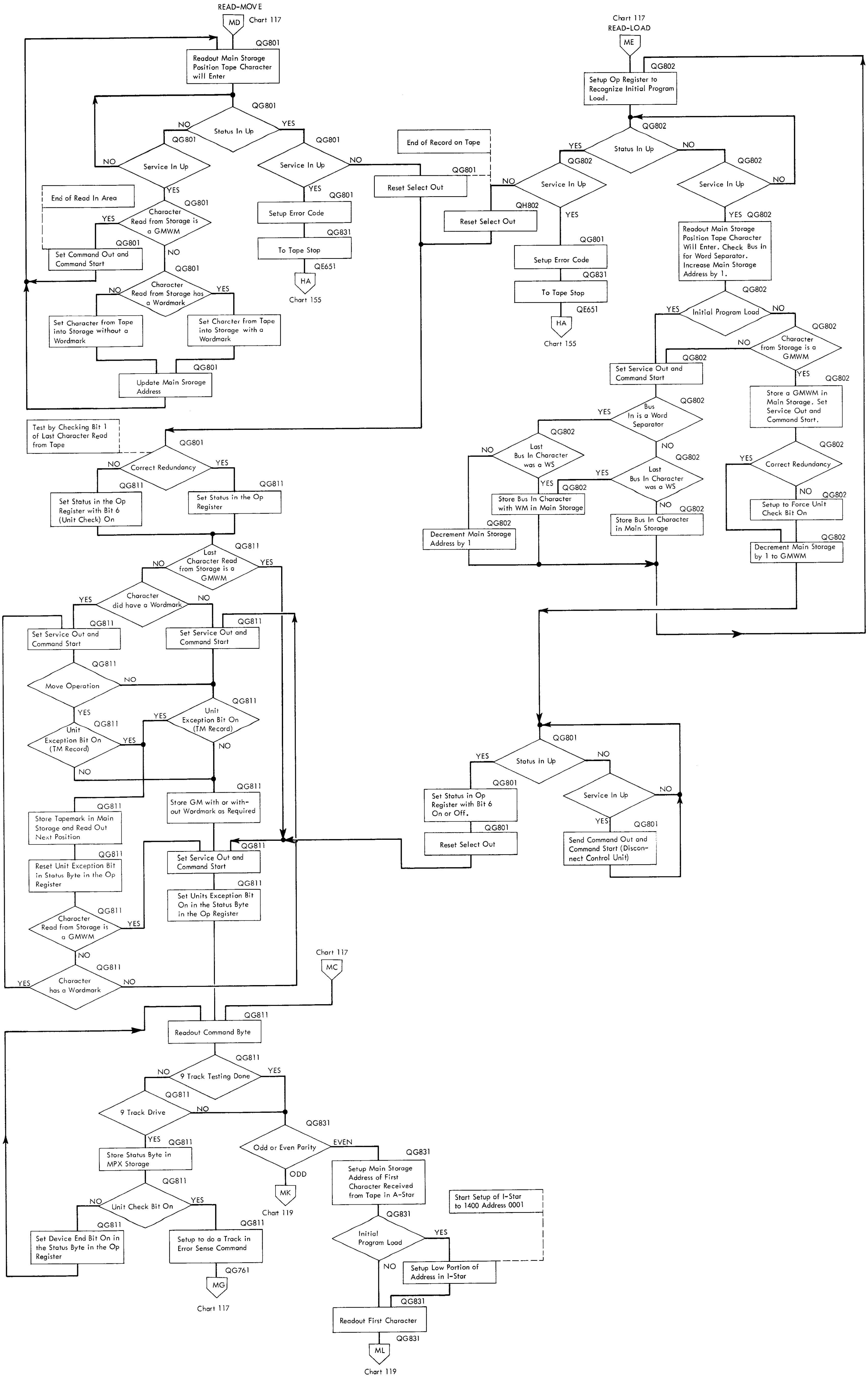
CLF 115 1400 Tape Common, Branch on EOF

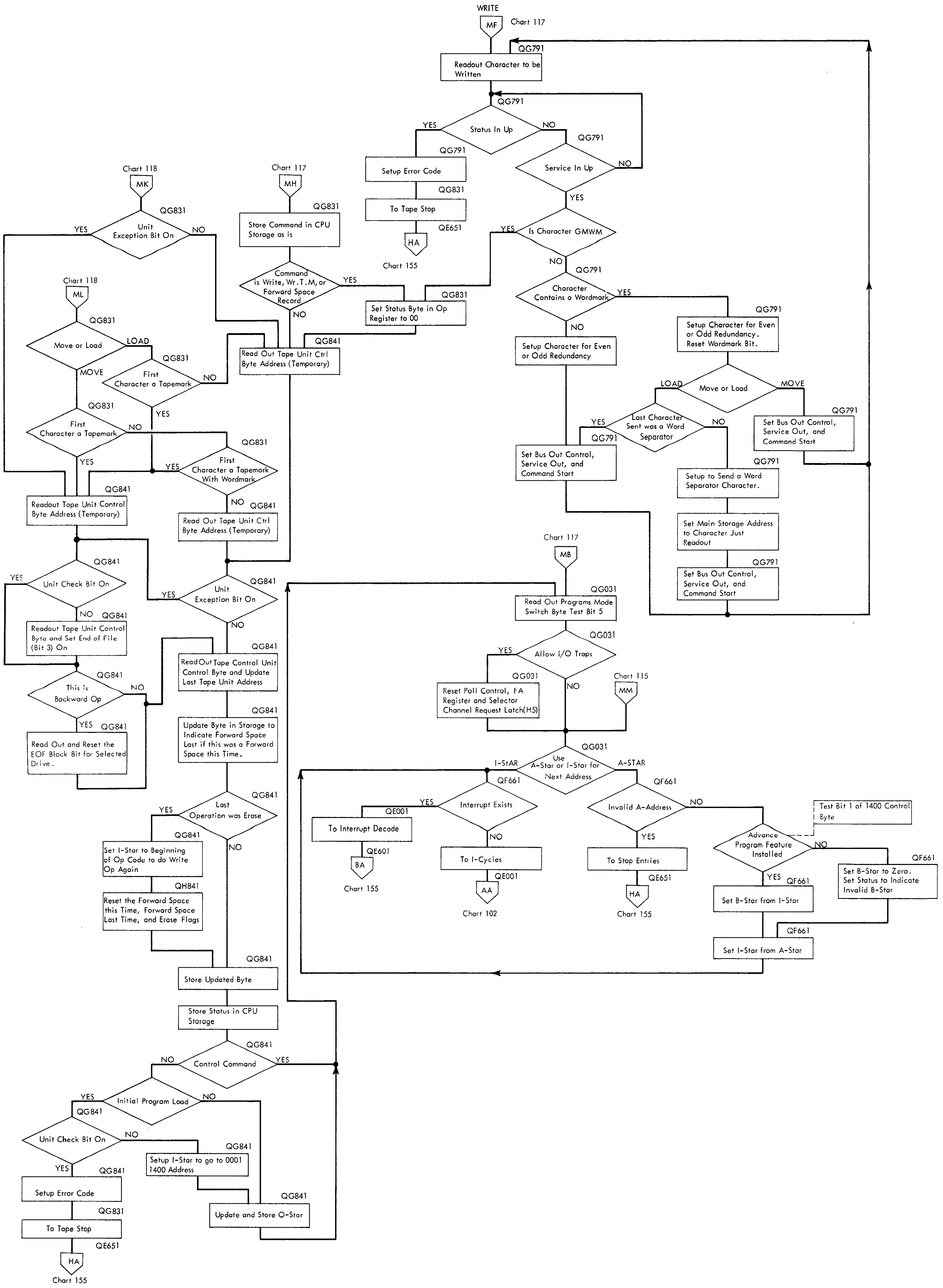


CLF 116 1400 Multiplexor Tape Objective

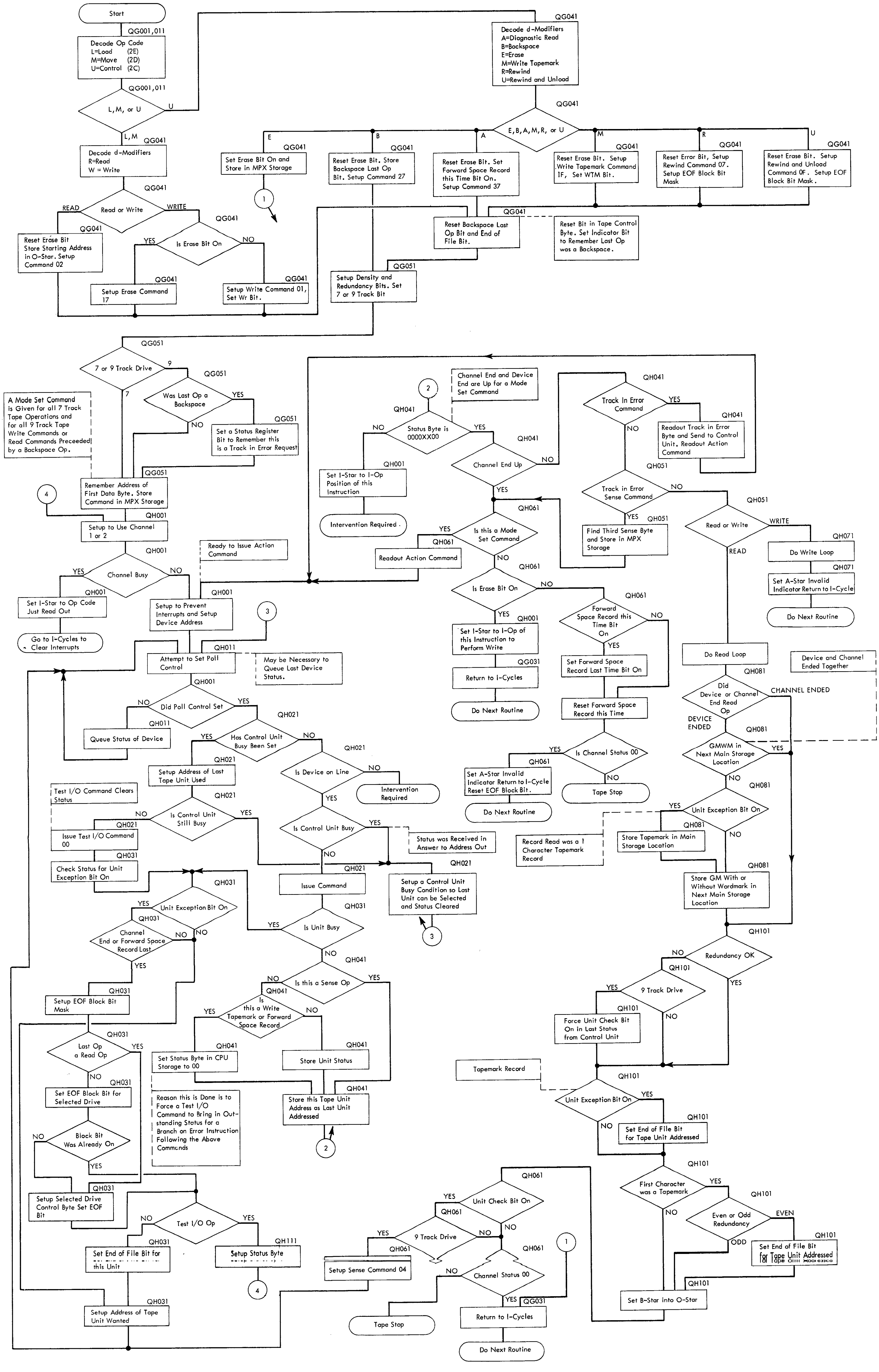


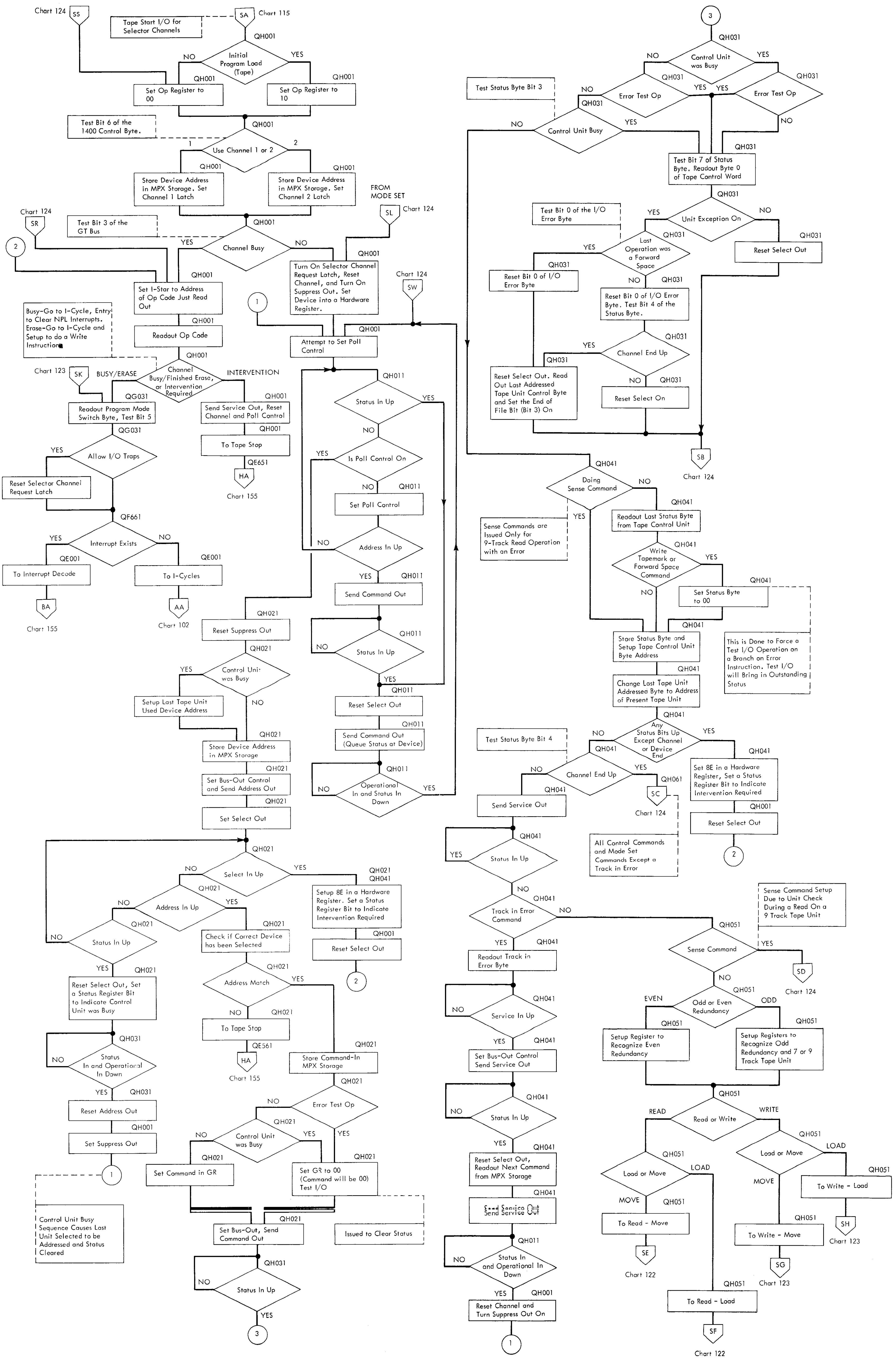
CLF 117 1400 Multiplexor Setup, Branch on Error


READ-MOVE
MD
Chart 117
QG801
Readout Main Storage Position Tape Character will Enter
QG801
Status In Up
NO
YES
QG801
Service In Up
NO
YES
End of Read In Area
QG801
Character Read from Storage is a GMWM
YES
NO
QG801
Set Command Out and Command Start
QG801
Character Read from Storage has a Wordmark
NO
YES
Set Character from Tape into Storage without a Wordmark
Set Charcter from Tape into Storage with a Wordmark
QG801
Update Main Srorage Address
QG801
Service In Up
NO
YES
QG801
Setup Error Code
QG831
To Tape Stop
QE651
HA
Chart 155
End of Record on Tape
QG801
Reset Select Out
Test by Checking Bit 1 of Last Character Read from Tape
QG801
Correct Redundancy
NO
YES
QG811
Set Status in the Op Register with Bit 6 (Unit Check) On
QG811
Set Status in the Op Register
QG811
Last Character Read from Storage is a GMWM
NO
YES
Character did have a Wordmark
YES
NO
QG811
Set Service Out and Command Start
QG811
Set Service Out and Command Start
QG811
Move Operation
NO
YES
QG811
Unit Exception Bit On (TM Record)
YES
NO
QG811
Unit Exception Bit On (TM Record)
YES
NO
QG811
Store GM with or without Wordmark as Required
QG811
Store Tapemark in Main Storage and Read Out Next Position
QG811
Reset Unit Exception Bit in Status Byte in the Op Register
QG811
Set Service Out and Command Start
QG811
Set Units Exception Bit On in the Status Byte in the Op Register
QG811
Character Read from Storage is a GMWM
YES
NO
QG811
Character has a Wordmark
YES
NO
Chart 117
MC
QG811
Readout Command Byte
QG811
9 Track Testing Done
NO
YES
QG811
9 Track Drive
NO
YES
QG811
Store Status Byte in MPX Storage
QG811
Unit Check Bit On
NO
YES
QG811
Set Device End Bit On in the Status Byte in the Op Register
QG811
Setup to do a Track in Error Sense Command
QG761
MG
Chart 117
QG831
Odd or Even Parity
EVEN
ODD
MK
Chart 119
QG831
Setup Main Storage Address of First Character Received from Tape in A-Star
QG831
Initial Program Load
YES
NO
Start Setup of I-Star to 1400 Address 0001
Setup Low Portion of Address in I-Star
QG831
Readout First Character
QG831
ML
Chart 119
Chart 117
READ-LOAD
ME
QG802
Setup Op Register to Recognize Initial Program Load.
QG802
Status In Up
YES
NO
QG802
Service In Up
NO
YES
QH802
Reset Select Out
QG801
Setup Error Code
QG831
To Tape Stop
QE651
HA
Chart 155
QG802
Service In Up
NO
YES
QG802
Readout Main Storage Position Tape Character Will Enter. Check Bus In for Word Separator. Increase Main Storage Address by 1.
QG802
Initial Program Load
YES
NO
QG802
Character from Storage is a GMWM
NO
YES
QG802
Set Service Out and Command Start
QG802
Store a GMWM in Main Storage. Set Service Out and Command Start.
QG802
Bus In is a Word Separator
YES
NO
QG802
Last Bus In Character was a WS
NO
YES
QG802
Last Bus In Character was a WS
YES
NO
QG802
Store Bus In Character with WM in Main Storage
QG802
Store Bus In Character in Main Storage
QG802
Decrement Main Storage Address by 1
QG802
Correct Redundancy
YES
NO
QG802
Setup to Force Unit Check Bit On
QG802
Decrement Main Storage by 1 to GMWM
QG801
Status In Up
YES
NO
QG801
Set Status in Op Register with Bit 6 On or Off.
QG801
Reset Select Out
Service In Up
NO
YES
QG801
Send Command Out and Command Start (Disconnect Control Unit)

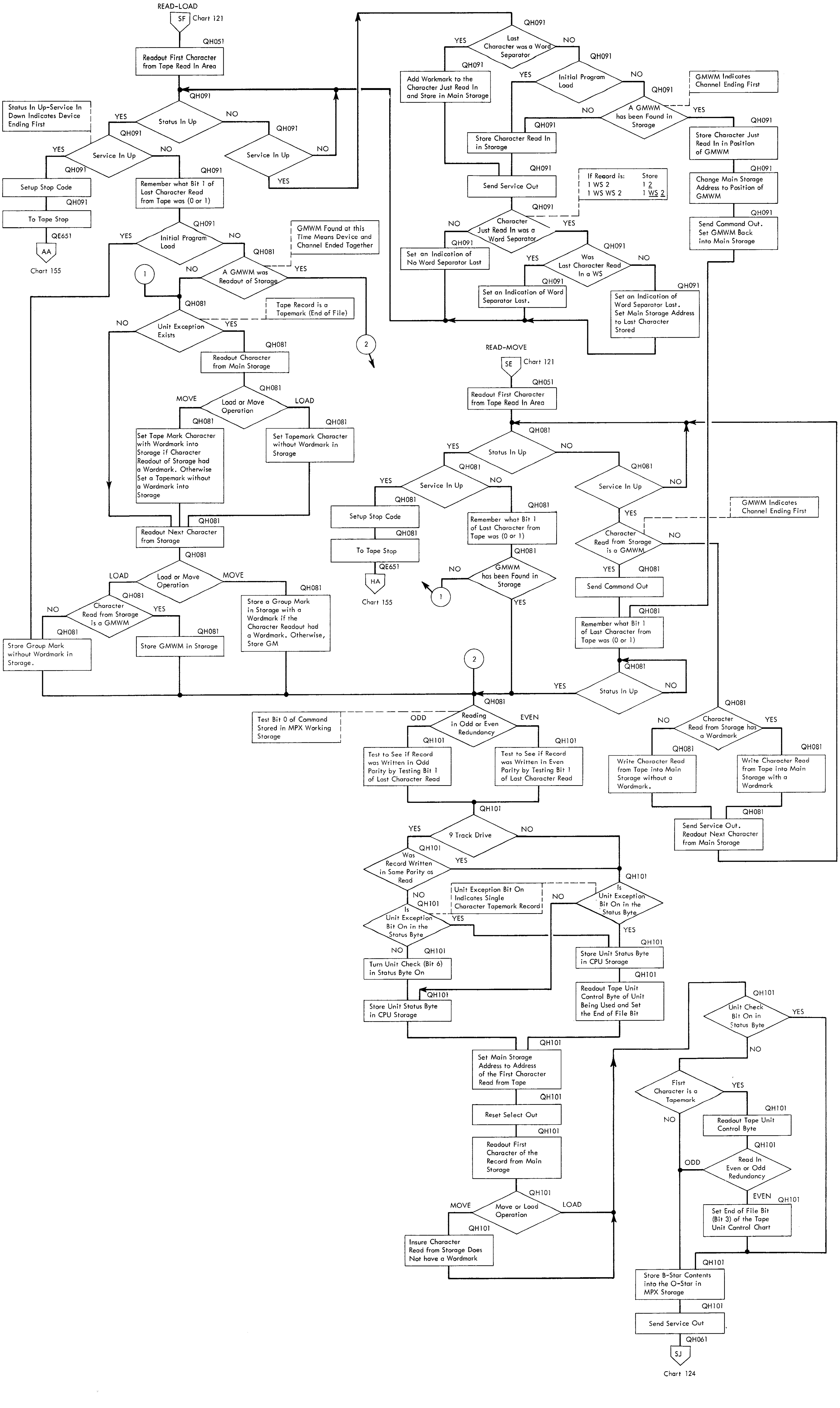


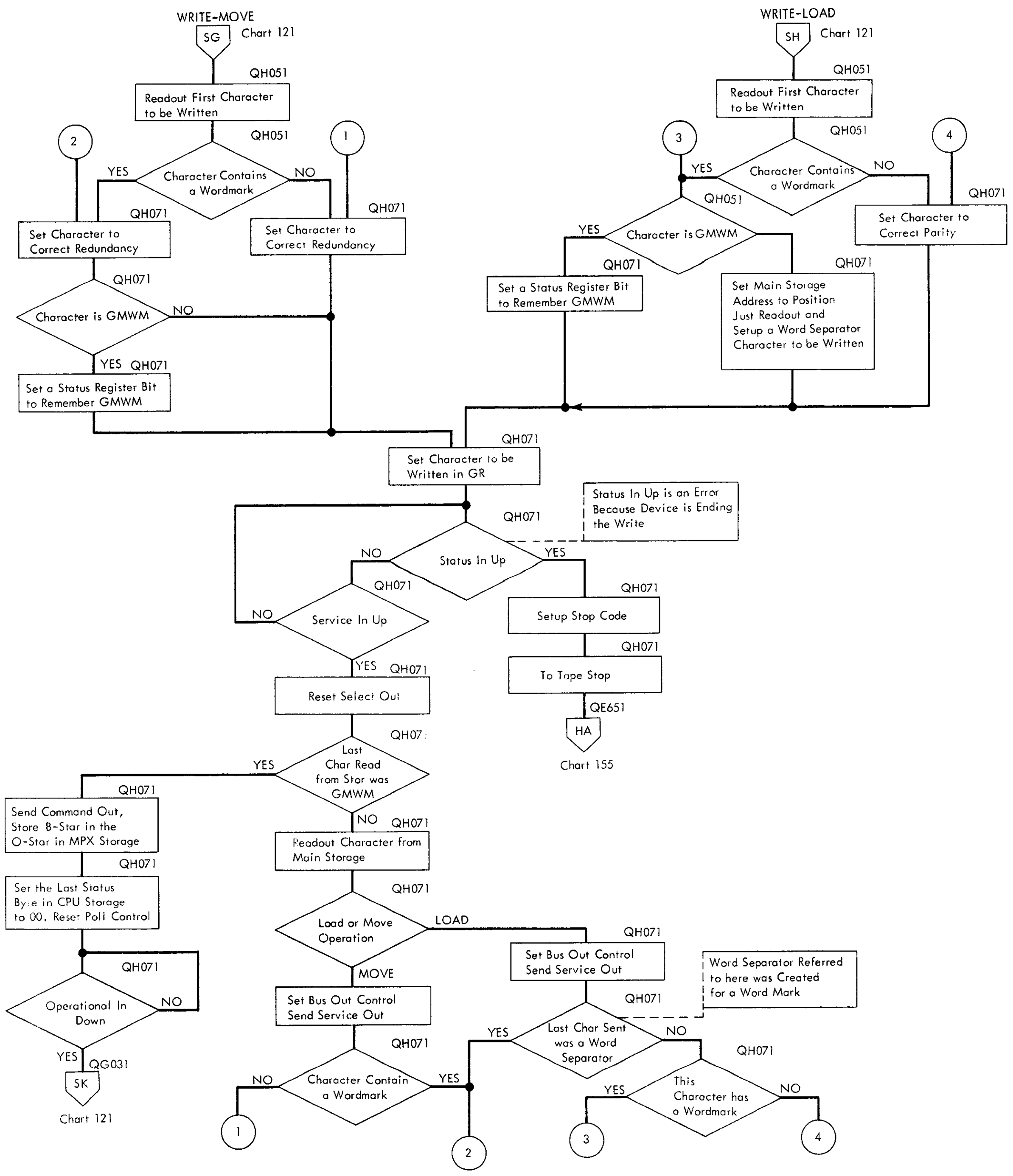
CLF 119 1400 Tape - Multiplexor Write, Tape Ending


Start
QG001,011
Decode Op Code
L=Load (2E)
M=Move (2D)
U=Control (2C)
QG001,011
L, M, or U
U
L,M
QG041
Decode d-Modifiers
R=Read
W = Write
QG041
Read or Write
READ
WRITE
QG041
Reset Erase Bit
Store Starting Address
in O-Star. Setup
Command 02
QG041
Is Erase Bit On
YES
NO
QG041
Setup Erase Command
17
QG041
Setup Write Command 01,
Set Wr Bit.
QG041
Decode d-Modifiers
A=Diagnostic Read
B=Backspace
E=Erase
M=Write Tapemark
R=Rewind
U=Rewind and Unload
QG041
E, B, A, M, R, or U
E
QG041
Set Erase Bit On and
Store in MPX Storage
1
B
QG041
Reset Erase Bit. Store
Backspace Last Op
Bit. Setup Command 27
A
Reset Erase Bit. Set
Forward Space Record
this Time Bit On.
Setup Command 37
M
QG041
Reset Erase Bit. Setup
Write Tapemark Command
1F, Set WTM Bit.
R
QG041
Reset Error Bit, Setup
Rewind Command 07.
Setup EOF Block Bit
Mask
U
QG041
Reset Erase Bit. Setup
Rewind and Unload
Command 0F. Setup EOF
Block Bit Mask.
QG041
Reset Backspace Last
Op Bit and End of
File Bit.
Reset Bit in Tape Control
Byte. Set Indicator Bit
to Remember Last Op
was a Backspace.
QG051
Setup Density and
Redundancy Bits. Set
7 or 9 Track Bit
QG051
7 or 9 Track Drive
9
7
QG051
Was Last Op a
Backspace
YES
NO
QG051
Set a Status Register
Bit to Remember this
is a Track in Error Request
A Mode Set Command
is Given for all 7 Track
Tape Operations and
for all 9 Track Tape
Write Commands or
Read Commands Preceeded
by a Backspace Op.
QG051
Remember Address of
First Data Byte. Store
Command in MPX Storage
4
QH001
Setup to Use Channel
1 or 2
QH001
Channel Busy
YES
NO
QH001
Set I-Star to Op Code
Just Read Out
Go to I-Cycles to
Clear Interrupts
Ready to Issue Action
Command
QH001
Setup to Prevent
Interrupts and Setup
Device Address
3
QH011
Attempt to Set Poll
Control
May be Necessary to
Queue Last Device
Status.
QH001
Did Poll Control Set
NO
YES
QH011
Queue Status of Device
QH021
Has Control Unit
Busy Been Set
YES
NO
QH021
Setup Address of Last
Tape Unit Used
QH021
Is Device on Line
NO
YES
Intervention
Required
Test I/O Command Clears
Status
QH021
Is Control Unit
Still Busy
NO
YES
QH021
Issue Test I/O Command
00
QH031
Check Status for Unit
Exception Bit On
QH021
Is Control Unit Busy
YES
NO
Status was Received in
Answer to Address Out
QH021
Issue Command
QH021
Setup a Control Unit
Busy Condition so Last
Unit can be Selected
and Status Cleared
3
QH031
Unit Exception Bit On
YES
NO
QH031
Channel
End or Forward Space
Record Last
NO
YES
QH031
Is Unit Busy
YES
NO
QH031
Setup EOF Block Bit
Mask
QH041
Is this a Sense Op
NO
YES
QH041
Is
this a Write
Tapemark or Forward
Space Record
YES
NO
QH031
Last Op
a Read Op
YES
NO
QH041
Set Status Byte in CPU
Storage to 00
QH041
Store Unit Status
QH031
Set EOF Block Bit for
Selected Drive
Reason this is Done is to
Force a Test I/O
Command to Bring in Out-
standing Status for a
Branch on Error Instruction
Following the Above
Commands
QH041
Store this Tape Unit
Address as Last Unit
Addressed
2
Block Bit
Was Already On
NO
YES
QH031
Setup Selected Drive
Control Byte Set EOF
Bit
Test I/O Op
NO
YES
QH031
Set End of File Bit for
this Unit
QH111
Setup Status Byte
4
QH031
Setup Address of Tape
Unit Wanted
2
Channel End and Device
End are Up for a Mode
Set Command
QH041
Status Byte is
0000XX00
NO
YES
QH001
Set I-Star to I-Op
Position of this
Instruction
Intervention Required
QH041
Channel End Up
NO
YES
QH061
Is this a Mode
Set Command
YES
NO
QH061
Readout Action Command
QH061
Is Erase Bit On
YES
NO
QH001
Set I-Star to I-Op of
this Instruction to
Perform Write
QG031
Return to I-Cycles
Do Next Routine
QH041
Track In Error
Command
YES
NO
QH041
Readout Track in Error
Byte and Send to Control
Unit. Readout Action
Command
QH051
Track in Error
Sense Command
YES
NO
QH051
Find Third Sense Byte
and Store in MPX
Storage
QH061
Forward
Space Record this
Time Bit
On
YES
NO
Set Forward Space
Record Last Time Bit On
Reset Forward Space
Record this Time
Is Channel Status 00
YES
NO
QH061
Set A-Star Invalid
Indicator Return to I-Cycle
Reset EOF Block Bit.
Do Next Routine
Tape Stop
QH051
Read or Write
WRITE
READ
QH071
Do Write Loop
QH071
Set A-Star Invalid
Indicator Return to I-Cycle
Do Next Routine
Do Read Loop
Device and Channel
Ended Together
QH081
Did
Device or Channel
End Read
Op
CHANNEL ENDED
DEVICE
ENDED
QH081
GMWM in
Next Main Storage
Location
YES
NO
QH081
Unit Exception Bit On
YES
NO
Record Read was a 1
Character Tapemark
Record
QH081
Store Tapemark in Main
Storage Location
QH081
Store GM With or
Without Wordmark in
Next Main Storage
Location
QH101
Redundancy OK
NO
YES
QH101
9 Track Drive
YES
NO
QH101
Force Unit Check Bit
On in Last Status
from Control Unit
Tapemark Record
QH101
Unit Exception Bit On
YES
NO
QH101
Set End of File Bit
for Tape Unit Addressed
QH101
First Character
was a Tapemark
YES
NO
QH101
Even or Odd
Redundancy
EVEN
ODD
QH101
Set End of File Bit
for Tape Unit Addressed
QH101
Set B-Star into O-Star
QH061
Unit Check Bit On
YES
NO
QH061
9 Track Drive
YES
NO
QH061
Setup Sense Command 04
QH061
Channel Status 00
NO
YES
1
Tape Stop
QG031
Return to I-Cycles
Do Next Routine

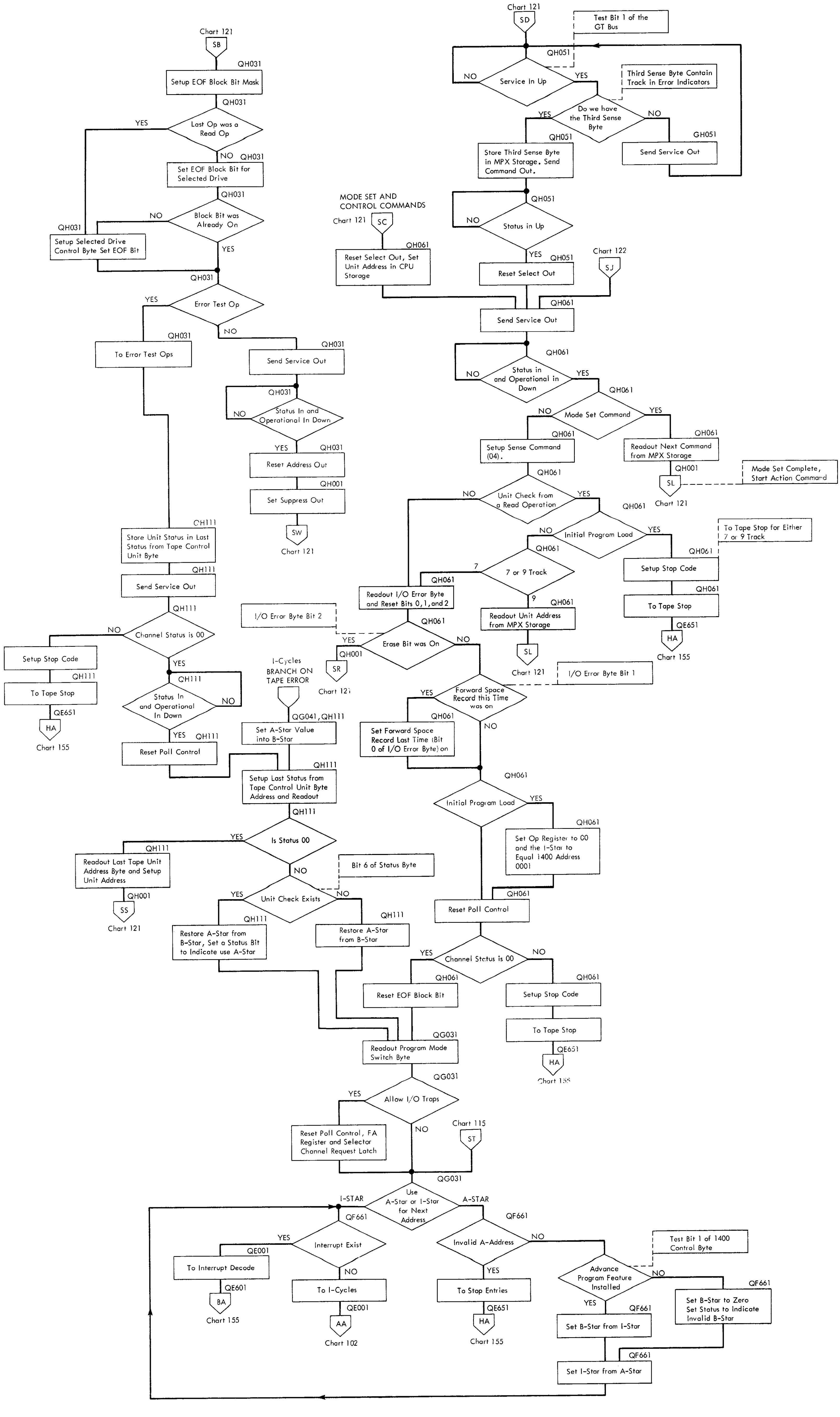


CLF 121 1400 Tape - Selector Setup


READ-LOAD
SF Chart 121
QH051
Readout First Character from Tape Read In Area
QH091
Status In Up-Service In Down Indicates Device Ending First
YES
Status In Up
NO
QH091
YES
Service In Up
NO
QH091
Service In Up
NO
YES
QH091
Setup Stop Code
QH091
To Tape Stop
QE651
AA
Chart 155
QH091
Remember what Bit 1 of Last Character Read from Tape was (0 or 1)
QH091
YES
Initial Program Load
NO
GMWM Found at this Time Means Device and Channel Ended Together
QH081
NO
A GMWM was Readout of Storage
YES
1
QH081
Tape Record is a Tapemark (End of File)
NO
Unit Exception Exists
YES
2
QH081
Readout Character from Main Storage
QH081
MOVE
Load or Move Operation
LOAD
QH081
Set Tape Mark Character with Wordmark into Storage if Character Readout of Storage had a Wordmark. Otherwise Set a Tapemark without a Wordmark into Storage
QH081
Set Tapemark Character without Wordmark in Storage
QH081
Readout Next Character from Storage
QH081
LOAD
Load or Move Operation
MOVE
QH081
NO
Character Read from Storage is a GMWM
YES
QH081
Store a Group Mark in Storage with a Wordmark if the Character Readout had a Wordmark. Otherwise, Store GM
QH081
Store Group Mark without Wordmark in Storage.
QH081
Store GMWM in Storage
QH091
YES
Last Character was a Word Separator
NO
QH091
Add Workmark to the Character Just Read In and Store in Main Storage
QH091
YES
Initial Program Load
NO
GMWM Indicates Channel Ending First
QH091
NO
A GMWM has been Found in Storage
YES
QH091
Store Character Read In in Storage
QH091
Store Character Just Read In in Position of GMWM
QH091
Send Service Out
QH091
Change Main Storage Address to Position of GMWM
If Record is: Store
1 WS 2 1 2
1 WS WS 2 1 WS 2
QH091
Send Command Out. Set GMWM Back into Main Storage
QH091
NO
Character Just Read In was a Word Separator
YES
QH091
Set an Indication of No Word Separator Last
QH091
YES
Was Last Character Read In a WS
NO
QH091
Set an Indication of Word Separator Last.
QH091
Set an Indication of Word Separator Last. Set Main Storage Address to Last Character Stored
READ-MOVE
SE Chart 121
QH051
Readout First Character from Tape Read In Area
QH081
YES
Status In Up
NO
QH081
YES
Service In Up
NO
QH081
Service In Up
NO
QH081
Setup Stop Code
QH081
Remember what Bit 1 of Last Character from Tape was (0 or 1)
GMWM Indicates Channel Ending First
YES
Character Read from Storage is a GMWM
NO
QH081
To Tape Stop
QH081
NO
GMWM has been Found in Storage
YES
QE651
HA
Chart 155
1
YES
QH081
Send Command Out
QH081
Remember what Bit 1 of Last Character from Tape was (0 or 1)
QH081
2
YES
Status In Up
NO
QH081
Test Bit 0 of Command Stored in MPX Working Storage
ODD
Reading in Odd or Even Redundancy
EVEN
QH101
Test to See if Record was Written in Odd Parity by Testing Bit 1 of Last Character Read
QH101
Test to See if Record was Written in Even Parity by Testing Bit 1 of Last Character Read
QH081
NO
Character Read from Storage has a Wordmark
YES
QH081
Write Character Read from Tape into Main Storage without a Wordmark.
QH081
Write Character Read from Tape into Main Storage with a Wordmark
QH081
Send Service Out. Readout Next Character from Main Storage
QH101
YES
9 Track Drive
NO
QH101
Was Record Written in Same Parity as Read
YES
QH101
NO
Unit Exception Bit On Indicates Single Character Tapemark Record
NO
Is Unit Exception Bit On in the Status Byte
QH101
Is Unit Exception Bit On in the Status Byte
YES
YES
NO
QH101
Turn Unit Check (Bit 6) in Status Byte On
QH101
Store Unit Status Byte in CPU Storage
QH101
Readout Tape Unit Control Byte of Unit Being Used and Set the End of File Bit
QH101
Store Unit Status Byte in CPU Storage
QH101
Unit Check Bit On in Status Byte
YES
NO
QH101
Set Main Storage Address to Address of the First Character Read from Tape
QH101
Reset Select Out
QH101
Readout First Character of the Record from Main Storage
First Character is a Tapemark
YES
NO
QH101
Readout Tape Unit Control Byte
QH101
ODD
Read In Even or Odd Redundancy
EVEN
QH101
Set End of File Bit (Bit 3) of the Tape Unit Control Chart
QH101
MOVE
Move or Load Operation
LOAD
QH101
Insure Character Read from Storage Does Not have a Wordmark
QH101
Store B-Star Contents into the O-Star in MPX Storage
QH101
Send Service Out
QH061
SJ
Chart 124

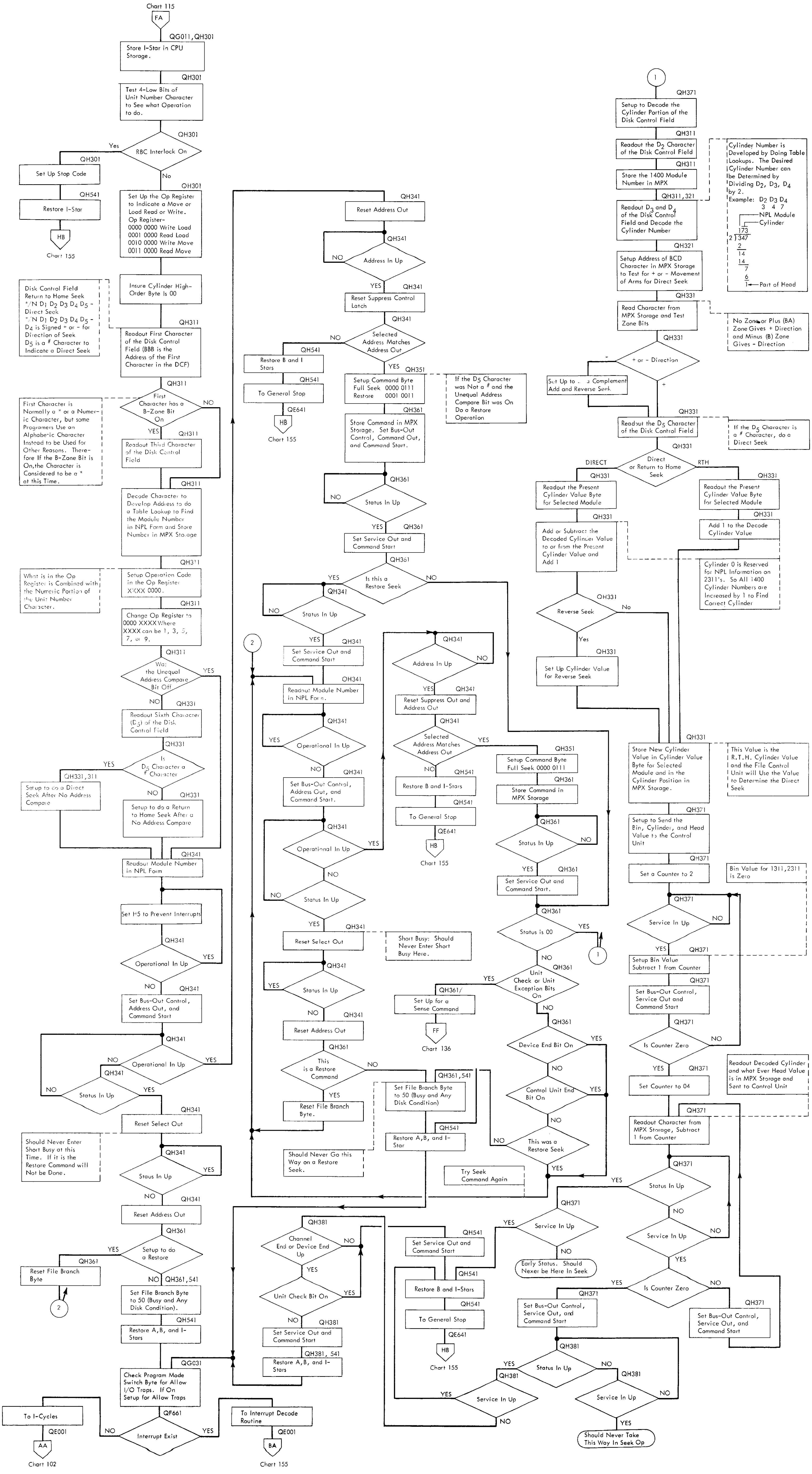


CLF 123 1400 Tape - Selector Write



CLF 124 1400 Tape - Selector Tape Ending, Branch on Error

Operation: M/L%F0 BBB R/W
Full Seek (Not Alternate Track)



CLF 125 1400 File - Seek Op

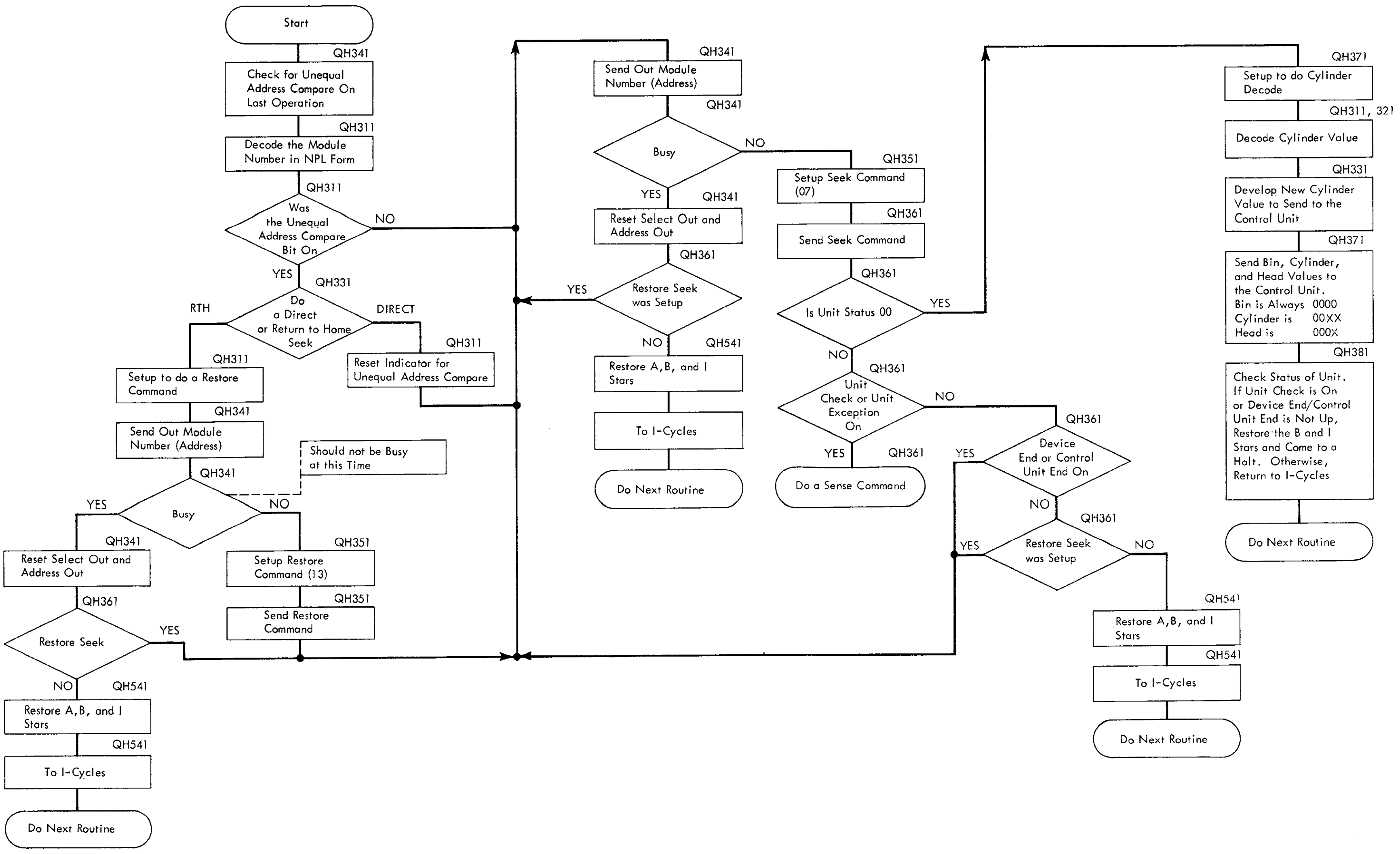
Seek Objectives:

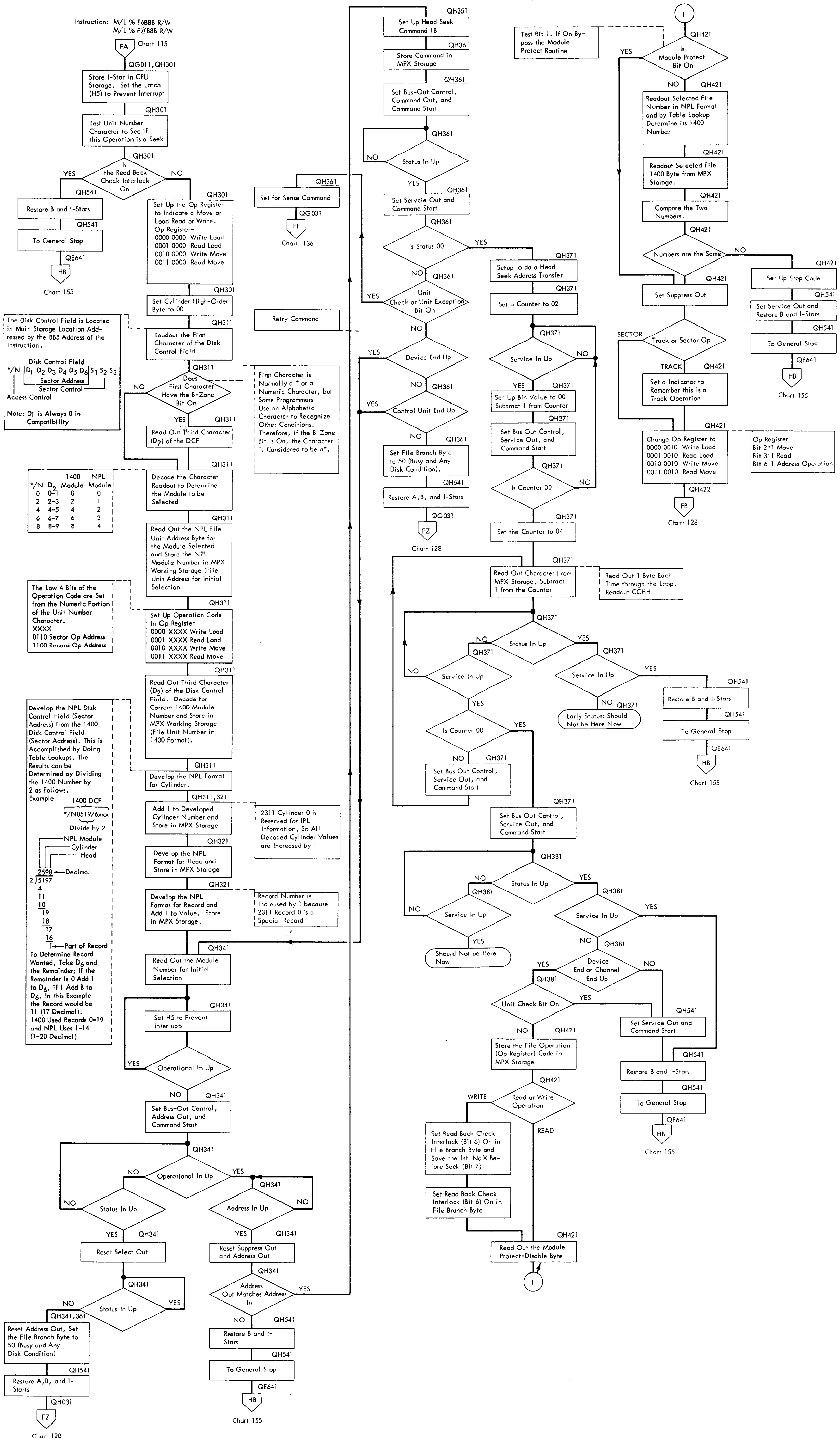
Direct Seek;	Decode the Disk Control Field for Module and Cylinder Values. Select the Module, Issue Seek Command (07), and Develop the Return to Home Seek Value to Send to the Control Unit. Send the Return to Home Value to the Control Unit and Check Unit Status.
Return to Home;	Decode the Disk Control Field for Module and Cylinder Values. Select the Module, Issue Seek Command (07), and Develop the Return to Home Seek Value to Send to the Control Unit. Send the Return to Home Value to the Control Unit and Check Unit Status.
Return to Home After a Unequal Address Compare;	Decode the Disk Control Field for Module and Cylinder Values. Issue a Restore Command (13), when Device End is Found Issue a Seek Command, and Develop the Return to Home Seek Value to Send to the Control Unit. Send Return to Home Value to the Control Unit and Check Unit Status.

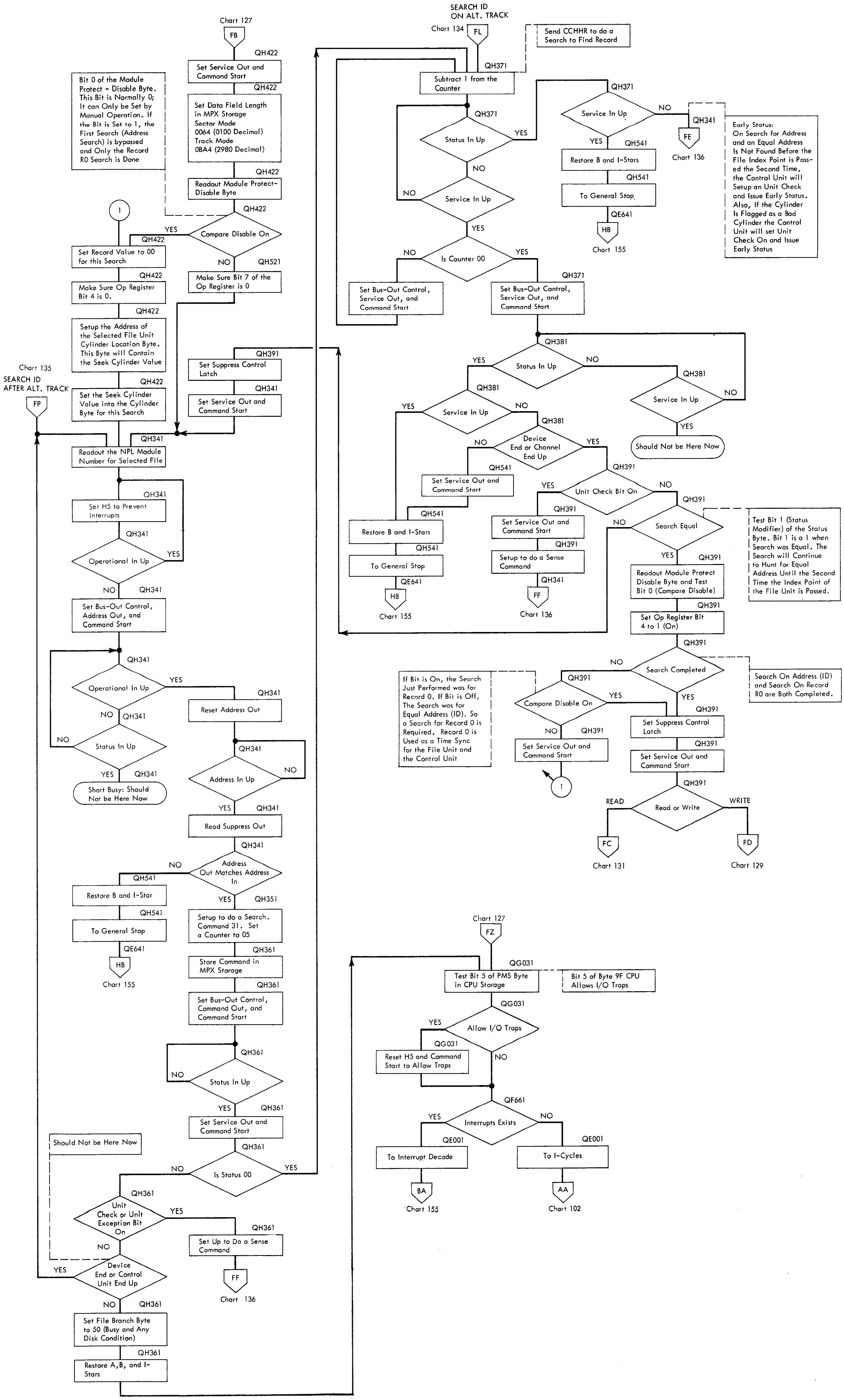
Note: The Cylinder Value is Increased by One Because Cylinder 0 is Reserved for IPL on 2311's. The format to Send Cylinder Value to the Control Unit is:

Bin	Bin	Cylinder	Cylinder	Head	Head
00	00	00	XX	00	0X

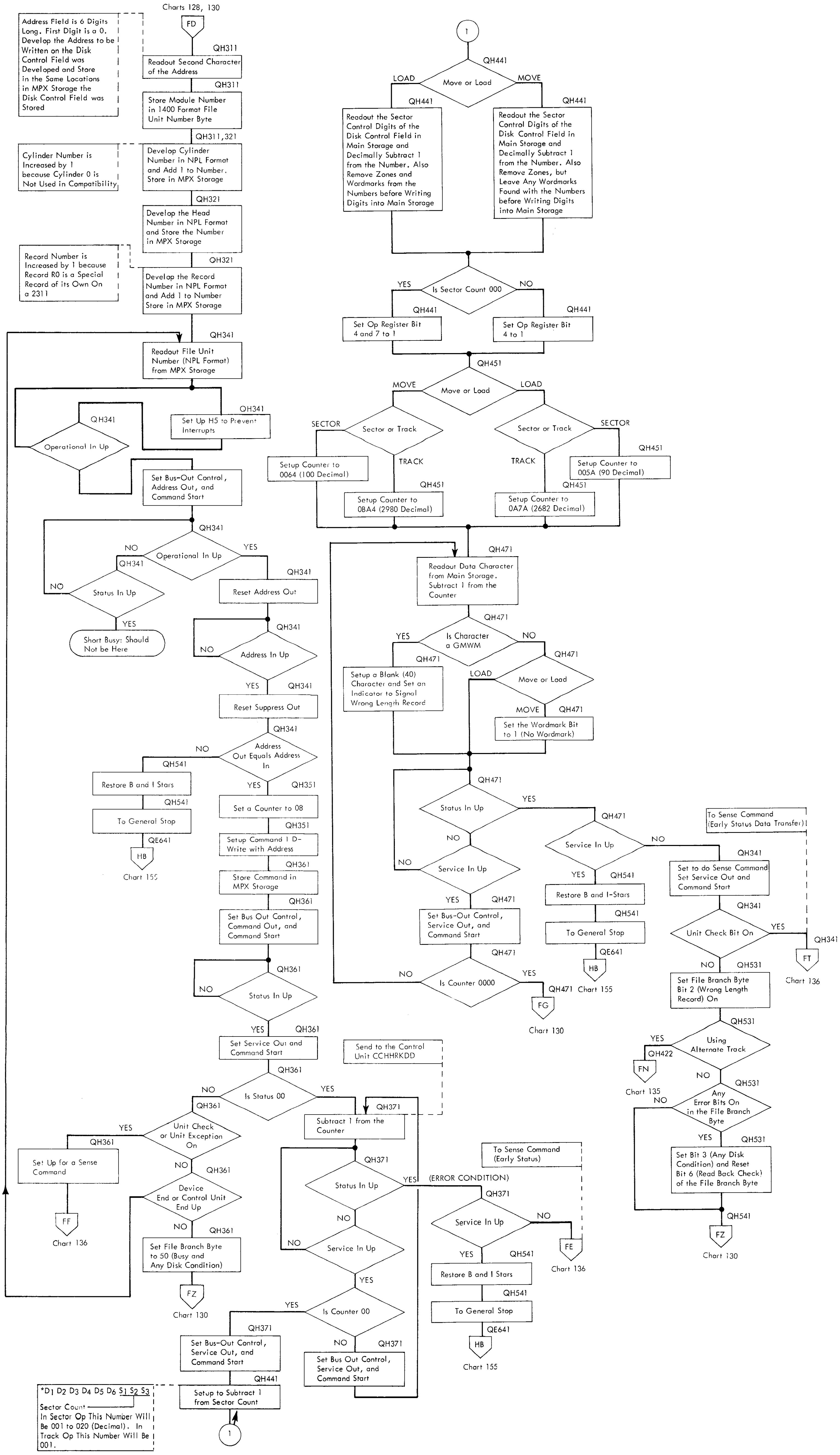
The Cylinder Value is Developed, the Head Value is whatever is in Head Location in MPX Storage. A Head Seek is Given on all Read or Write Operation so the Head Selected at this Time is not Important.



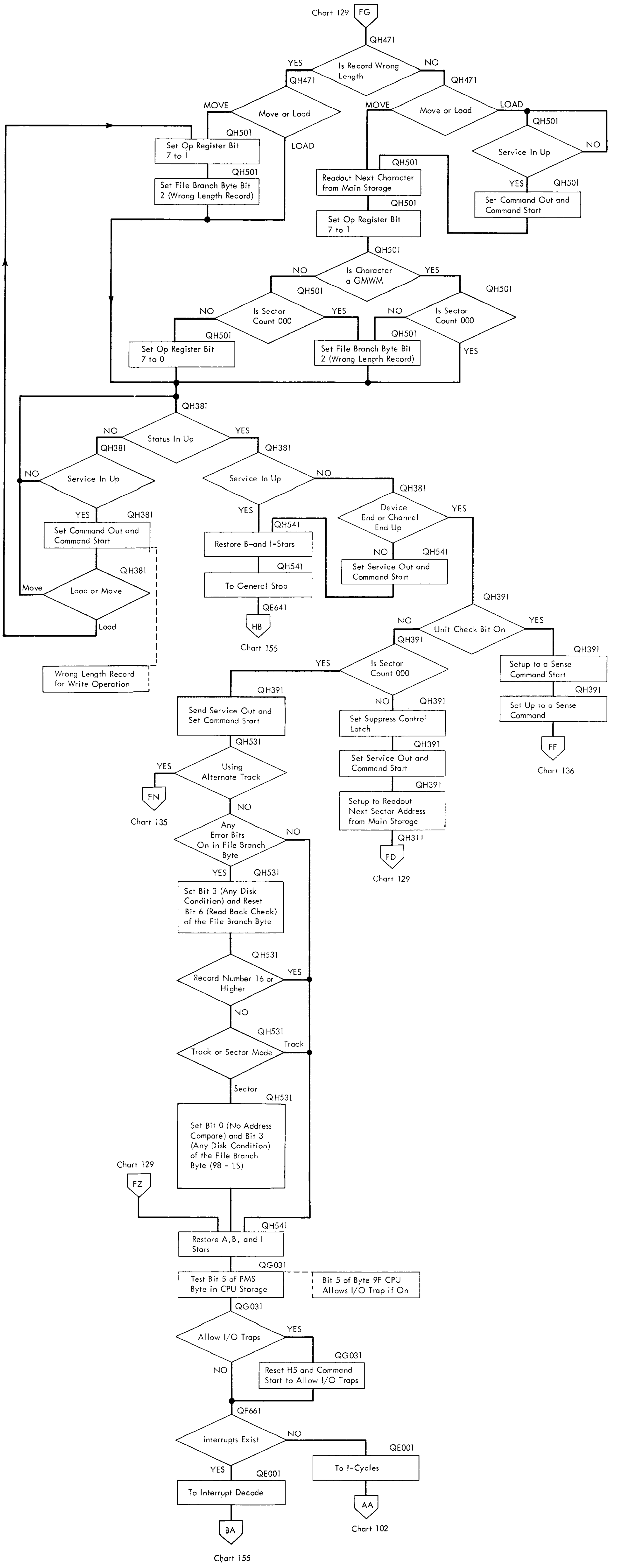

Instruction: M/L % F6BBB R/W
M/L % F@BBB R/W
FA
Chart 115
QG011,QH301
Store 1-Star in CPU Storage. Set the Latch (H5) to Prevent Interrupt
QH301
Test Unit Number Character to See if this Operation is a Seek
QH301
Is the Read Back Check Interlock On
YES
NO
QH541
Restore B and I-Stars
QH541
To General Stop
QE641
HB
Chart 155
QH301
Set Up the Op Register to Indicate a Move or Load Read or Write. Op Register–
0000 0000 Write Load
0001 0000 Read Load
0010 0000 Write Move
0011 0000 Read Move
QH301
Set Cylinder High-Order Byte to 00
QH311
Readout the First Character of the Disk Control Field
The Disk Control Field is Located in Main Storage Location Addressed by the BBB Address of the Instruction.
Disk Control Field
*/N D1 D2 D3 D4 D5 D6 S1 S2 S3
Sector Address
Sector Control
Access Control
Note: D1 is Always 0 in Compatibility
QH311
Does First Character Have the B-Zone Bit On
NO
YES
QH311
Read Out Third Character (D2) of the DCF
First Character is Normally a * or a Numeric Character, but Some Programmers Use an Alphabetic Character to Recognize Other Conditions. Therefore, if the B-Zone Bit is On, the Character is Considered to be a*.
QH311
Decode the Character Readout to Determine the Module to be Selected
*/N D2 1400 Module NPL Module
0 0-1 0 0
2 2-3 2 1
4 4-5 4 2
6 6-7 6 3
8 8-9 8 4
QH311
Read Out the NPL File Unit Address Byte for the Module Selected and Store the NPL Module Number in MPX Working Storage (File Unit Address for Initial Selection
The Low 4 Bits of the Operation Code are Set from the Numeric Portion of the Unit Number Character.
XXXX
0110 Sector Op Address
1100 Record Op Address
QH311
Set Up Operation Code in Op Register
0000 XXXX Write Load
0001 XXXX Read Load
0010 XXXX Write Move
0011 XXXX Read Move
QH311
Read Out Third Character (D2) of the Disk Control Field. Decode for Correct 1400 Module Number and Store in MPX Working Storage (File Unit Number in 1400 Format).
Develop the NPL Disk Control Field (Sector Address) from the 1400 Disk Control Field (Sector Address). This is Accomplished by Doing Table Lookups. The Results can be Determined by Dividing the 1400 Number by 2 as Follows.
Example 1400 DCF
*/N051976xxx
Divide by 2
NPL Module
Cylinder
Head
2598 Decimal
2) 5197
4
11
10
19
18
17
16
1 Part of Record
To Determine Record Wanted, Take D6 and the Remainder; If the Remainder is 0 Add 1 to D6, if 1 Add B to D6. In this Example the Record would be 11 (17 Decimal). 1400 Used Records 0-19 and NPL Uses 1-14 (1-20 Decimal)
QH311
Develop the NPL Format for Cylinder.
QH311,321
Add 1 to Developed Cylinder Number and Store in MPX Storage
2311 Cylinder 0 is Reserved for IPL Information. So All Decoded Cylinder Values are Increased by 1
QH321
Develop the NPL Format for Head and Store in MPX Storage
QH321
Develop the NPL Format for Record and Add 1 to Value. Store in MPX Storage.
Record Number is Increased by 1 because 2311 Record 0 is a Special Record
QH341
Read Out the Module Number for Initial Selection
QH341
Set H5 to Prevent Interrupts
YES
Operational In Up
NO
QH341
Set Bus-Out Control, Address Out, and Command Start
QH341
NO
Operational In Up
YES
NO
Status In Up
YES
QH341
Reset Select Out
QH341
Status In Up
NO
YES
QH341,361
Reset Address Out, Set the File Branch Byte to 50 (Busy and Any Disk Condition)
QH541
Restore A,B, and I-Starts
QH031
FZ
Chart 128
QH341
Address In Up
NO
YES
QH341
Reset Suppress Out and Address Out
QH341
Address Out Matches Address In
YES
NO
QH541
Restore B and I-Stars
QH541
To General Stop
QE641
HB
Chart 155
QH361
Set for Sense Command
QG031
FF
Chart 136
Retry Command
QH351
Set Up Head Seek Command 1B
QH361
Store Command in MPX Storage
QH361
Set Bus-Out Control, Command Out, and Command Start
QH361
NO
Status In Up
YES
QH361
Set Servcie Out and Command Start
QH361
Is Status 00
YES
NO
QH361
YES
Unit Check or Unit Exception Bit On
NO
YES
Device End Up
NO
QH361
YES
Control Unit End Up
NO
QH361
Set File Branch Byte to 50 (Busy and Any Disk Condition).
QH541
Restore A,B, and I-Stars
QG031
FZ
Chart 128
QH371
Setup to do a Head Seek Address Transfer
QH371
Set a Counter to 02
QH371
Service In Up
NO
YES
QH371
Set Up Bin Value to 00 Subtract 1 from Counter
QH371
Set Bus Out Control, Service Out, and Command Start
QH371
Is Counter 00
NO
QH371
Set the Counter to 04
QH371
Read Out Character From MPX Storage, Subtract 1 from the Counter
Read Out 1 Byte Each Time through the Loop. Readout CCHH
QH371
NO
Status In Up
YES
QH371
NO
Service In Up
YES
QH371
Service In Up
YES
NO
QH371
Early Status: Should Not be Here Now
QH541
Restore B and I-Stars
QH541
To General Stop
QE641
HB
Chart 155
Is Counter 00
YES
NO
QH371
Set Bus Out Control, Service Out, and Command Start
QH371
Set Bus Out Control, Service Out, and Command Start
QH381
NO
Status In Up
YES
QH381
NO
Service In Up
YES
Should Not be Here Now
QH381
Service In Up
YES
NO
QH381
Device End or Channel End Up
YES
NO
QH381
Unit Check Bit On
YES
NO
QH421
Store the File Operation (Op Register) Code in MPX Storage
QH421
WRITE
Read or Write Operation
READ
Set Read Back Check Interlock (Bit 6) On in File Branch Byte and Save the 1st NoX Before Seek (Bit 7).
Set Read Back Check Interlock (Bit 6) On in File Branch Byte
QH421
Read Out the Module Protect-Disable Byte
1
QH541
Set Service Out and Command Start
QH541
Restore B and I-Stars
QH541
To General Stop
QE641
HB
Chart 155
1
Test Bit 1. If On Bypass the Module Protect Routine
QH421
YES
Is Module Protect Bit On
NO
QH421
Readout Selected File Number in NPL Format and by Table Lookup Determine its 1400 Number
QH421
Readout Selected File 1400 Byte from MPX Storage.
QH421
Compare the Two Numbers.
QH421
Numbers are the Same
NO
QH421
Set Up Stop Code
QH541
Set Service Out and Restore B and I-Stars
QH541
To General Stop
QE641
HB
Chart 155
QH421
Set Suppress Out
SECTOR
Track or Sector Op
TRACK
QH421
Set a Indicator to Remember this is a Track Operation
QH421
Change Op Register to
0000 0010 Write Load
0001 0010 Read Load
0010 0010 Write Move
0011 0010 Read Move
Op Register
Bit 2=1 Move
Bit 3=1 Read
Bit 6=1 Address Operation
QH422
FB
Chart 128



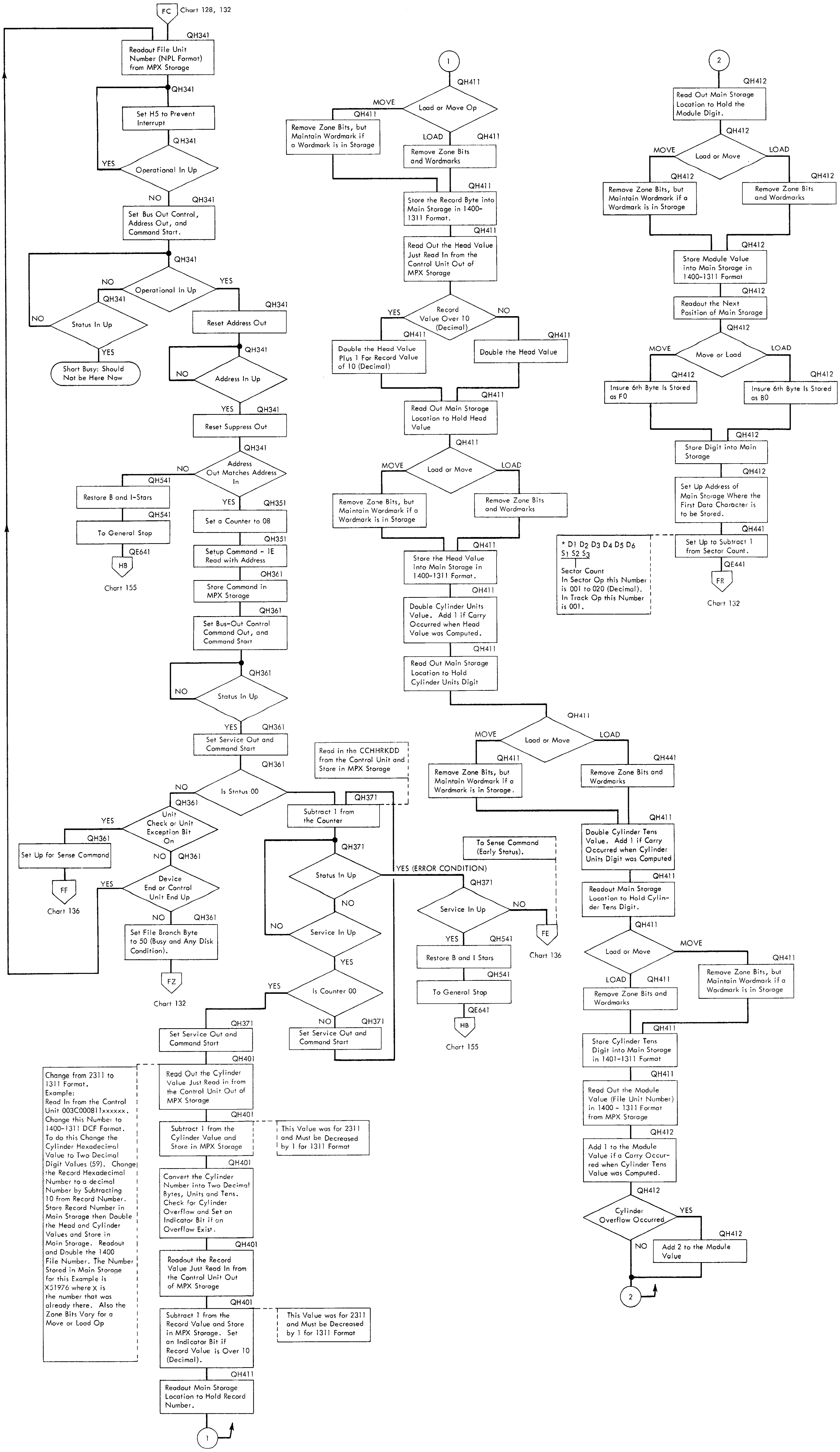
CLF 128 1400 File - R/W With Addresses - Sheet 2



CLF 129 1400 File - Write With Addresses Data Loop



CLF 130 1400 File - Write With Addresses Data Loop - Sheet 2



Read in the CCHHRKDD from the Control Unit and Store in MPX Storage

Change from 2311 to 1311 Format.
Example:
Read In from the Control Unit 003C00081 1xxxxxx. Change this Number to 1400-1311 DCF Format. To do this Change the Cylinder Hexadecimal Value to Two Decimal Digit Values (59). Change the Record Hexadecimal Number to a decimal Number by Subtracting 10 from Record Number. Store Record Number in Main Storage then Double the Head and Cylinder Values and Store in Main Storage. Readout and Double the 1400 File Number. The Number Stored in Main Storage for this Example is X51976 where X is the number that was already there. Also the Zone Bits Vary for a Move or Load Op

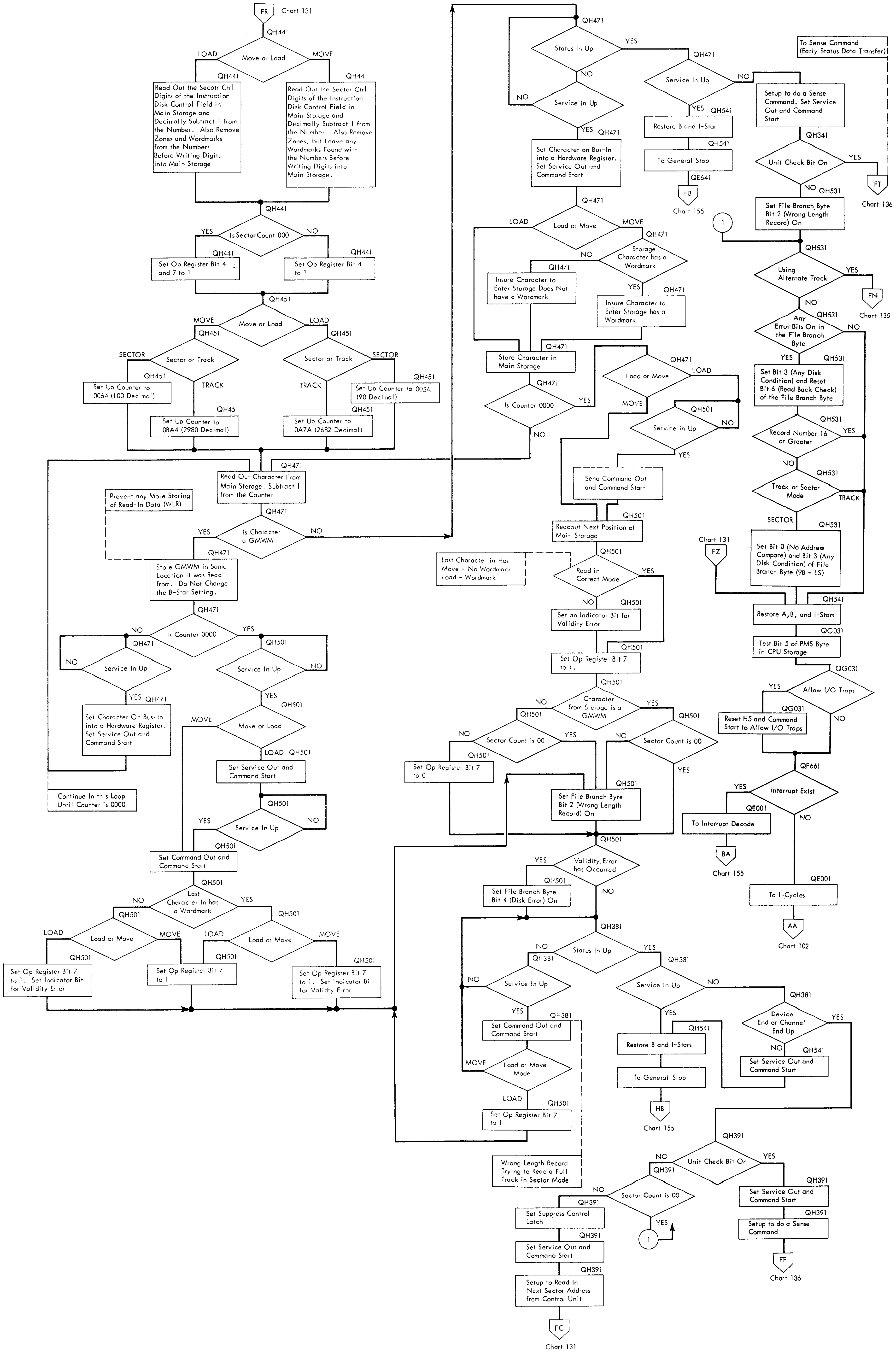
This Value was for 2311 and Must be Decreased by 1 for 1311 Format

This Value was for 2311 and Must be Decreased by 1 for 1311 Format

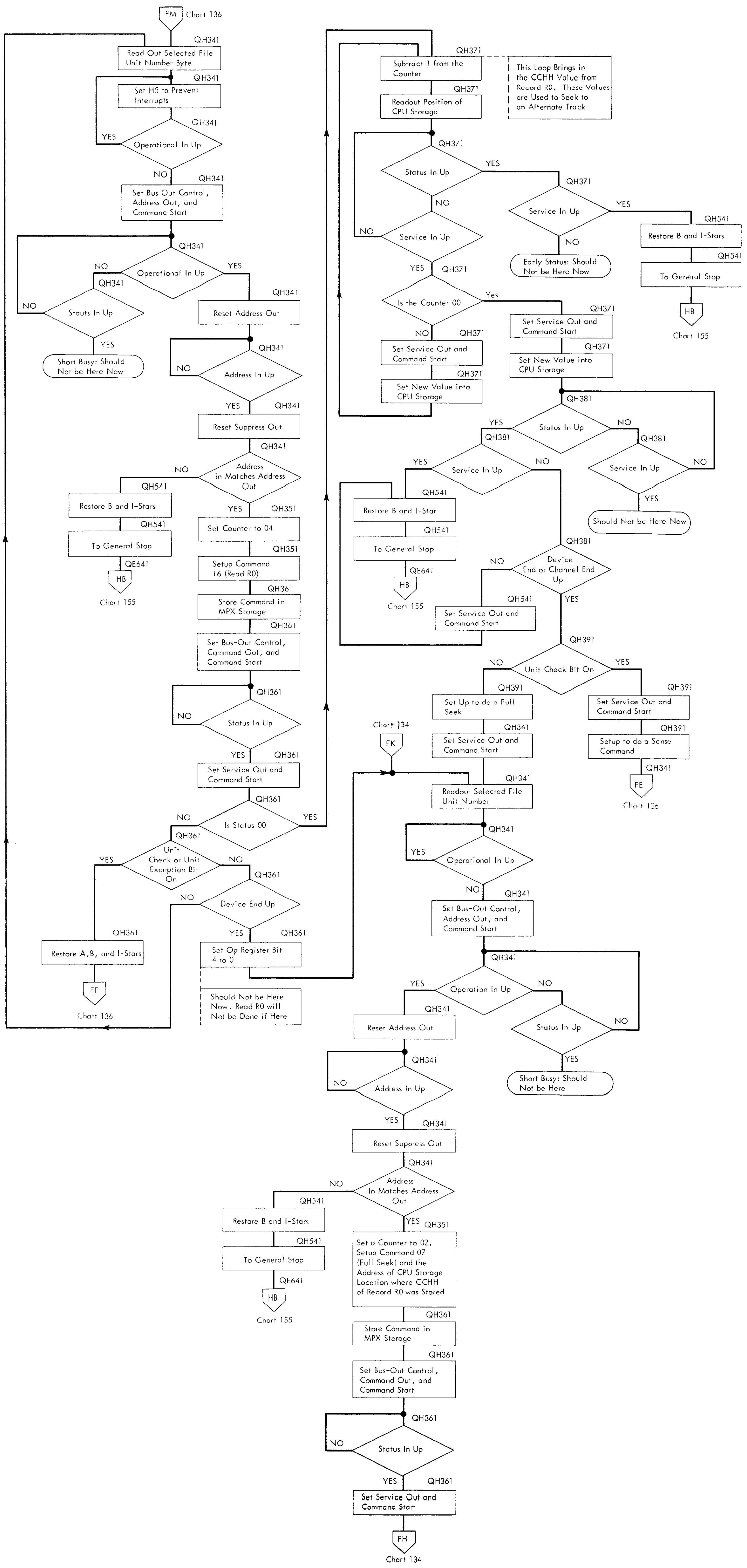
* D_1 D_2 D_3 D_4 D_5 D_6 $\underline{S_1}$ $\underline{S_2}$ $\underline{S_3}$

Sector Count
In Sector Op this Number is 001 to 020 (Decimal).
In Track Op this Number is 001.

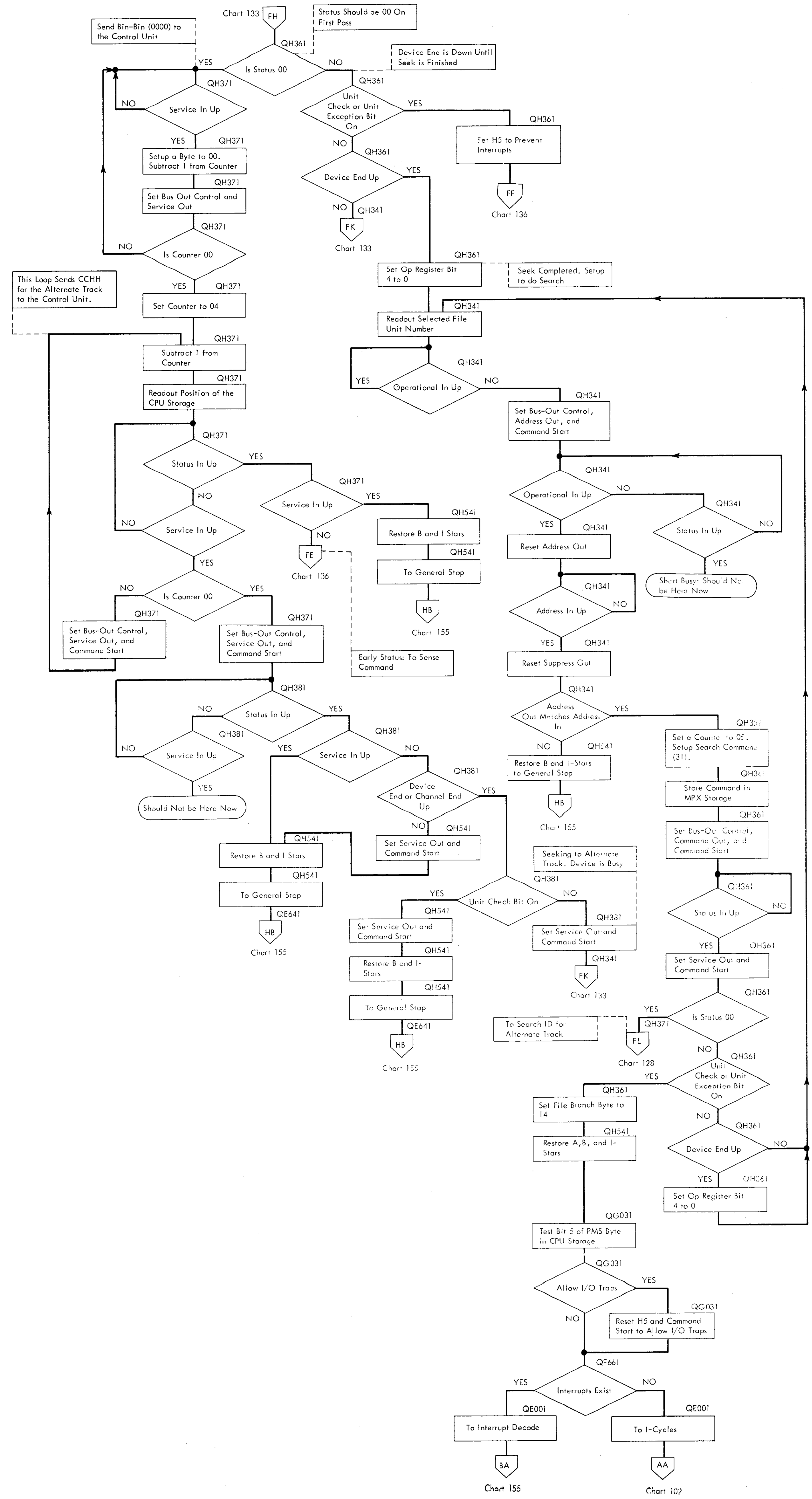
CLF 131 1400 File - Read With Addresses Data Loop



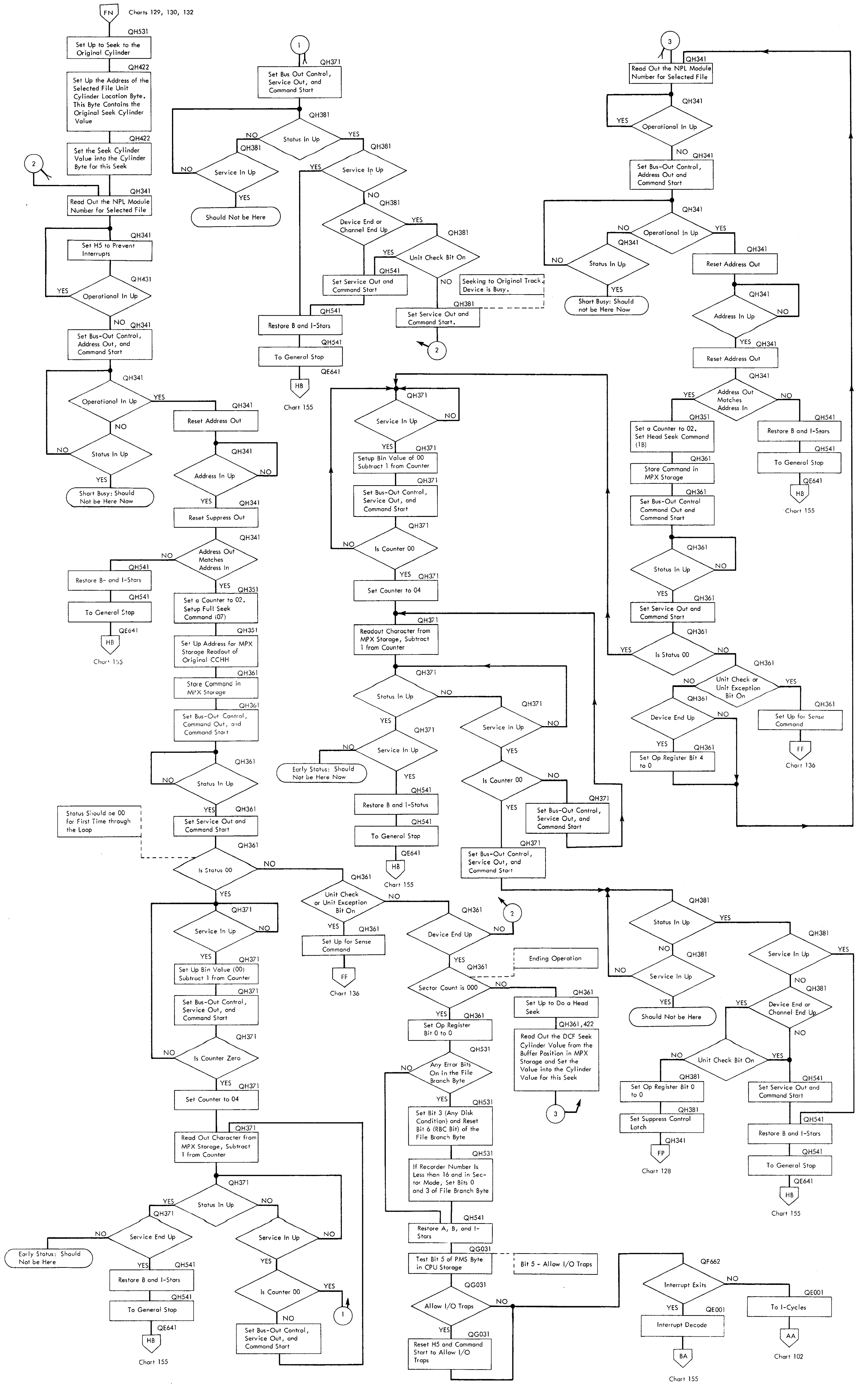
CLF 132   1400 File - Read With Addresses Data Loop - Sheet 2



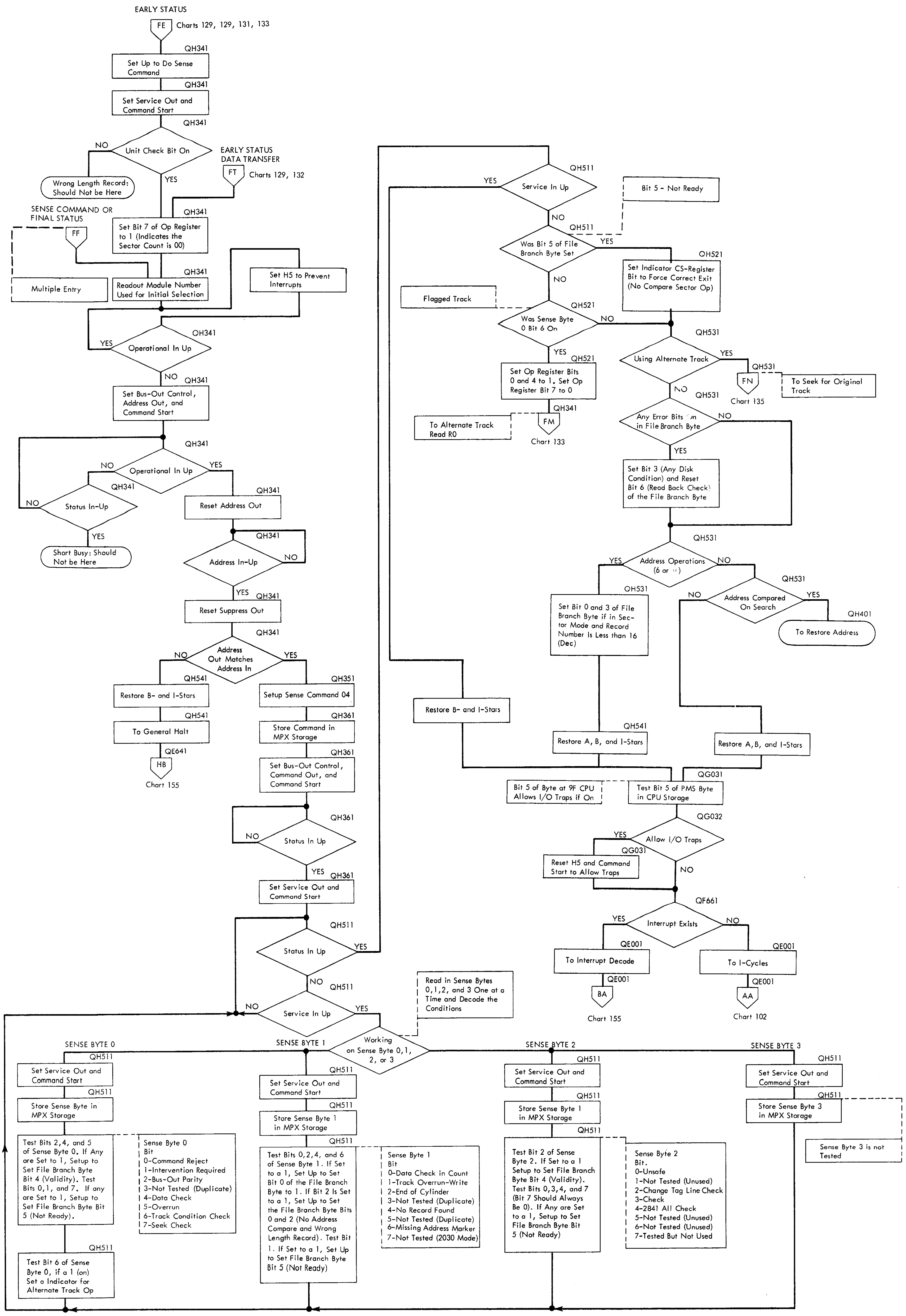
CLF 133 1400 File - Alternate Track Seek



CLF 134 1400 File - Alternate Track Seek - Sheet 2

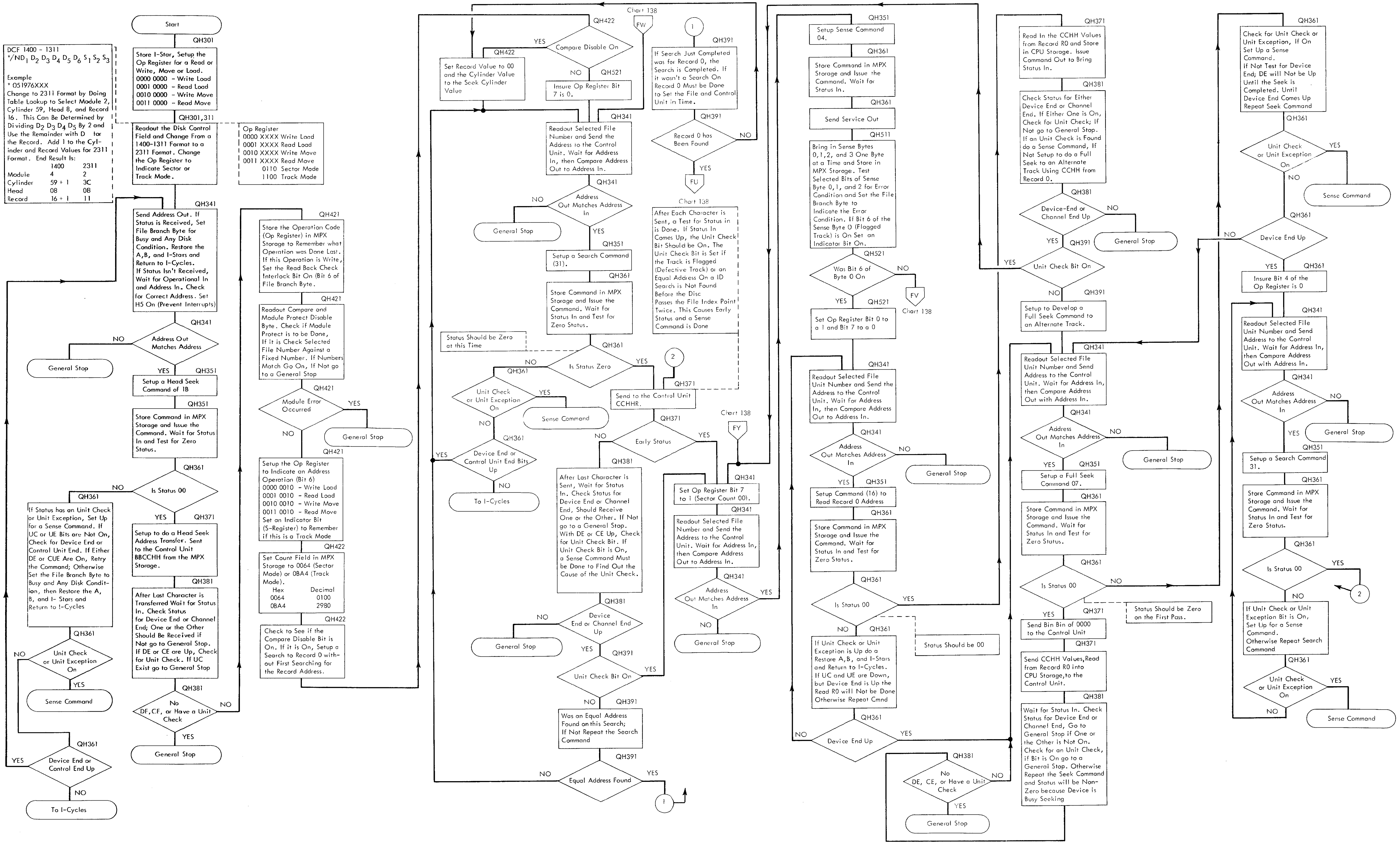


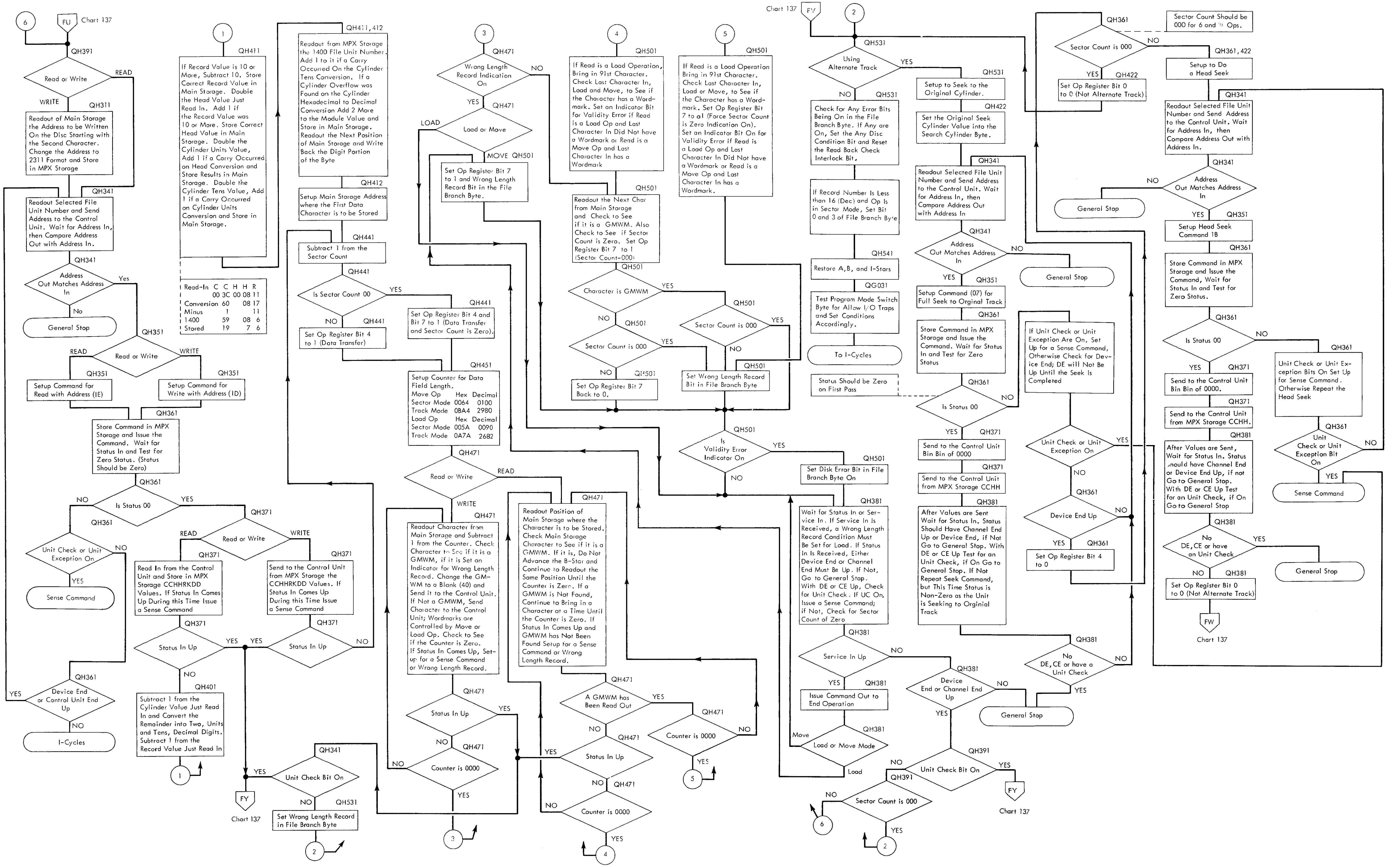
CLF 135 1400 File - Return to Original Track



CLF 136 1400 File - Sense Command for Unit Check

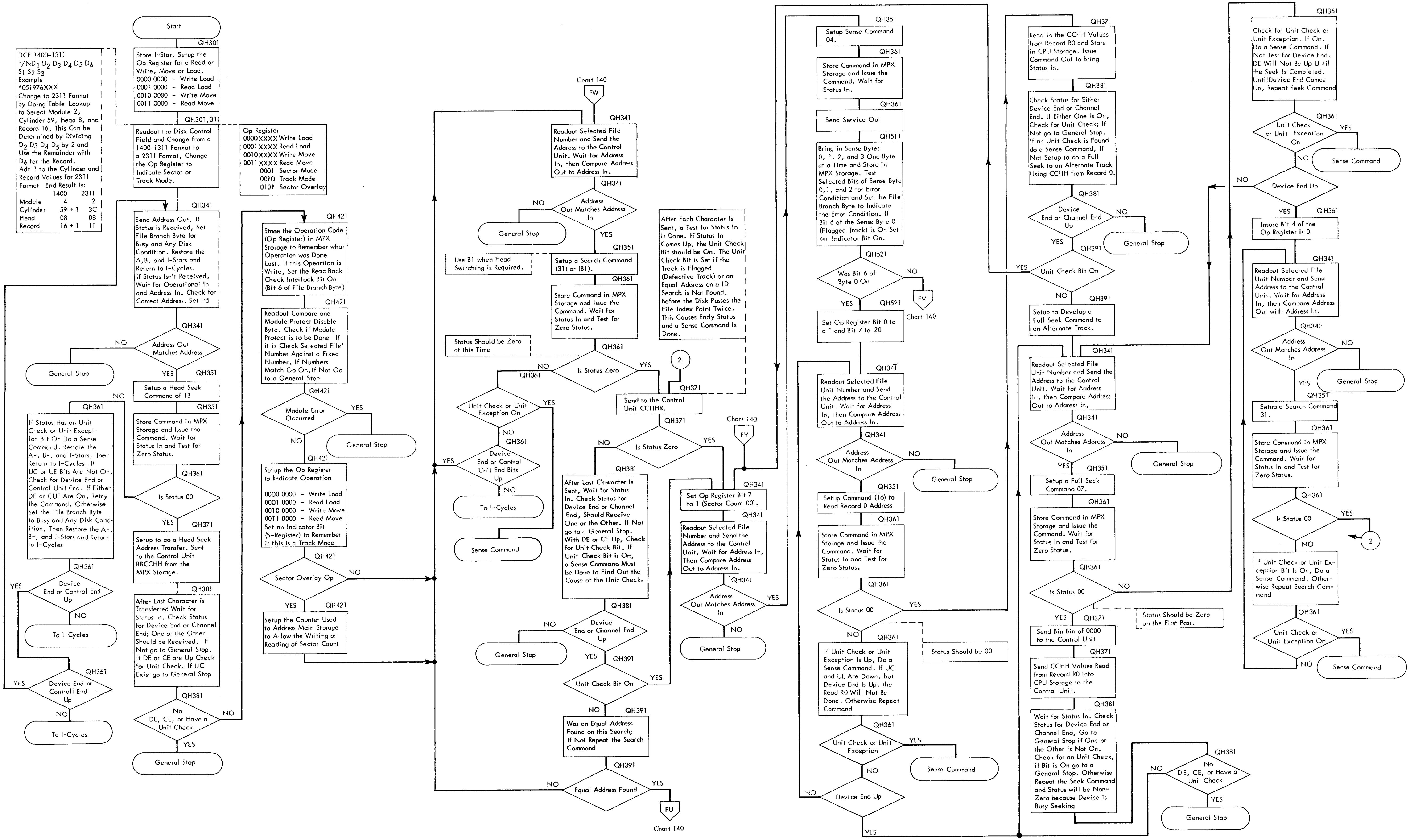
CLF 137 1400 File – R/W With Addresses Objectives



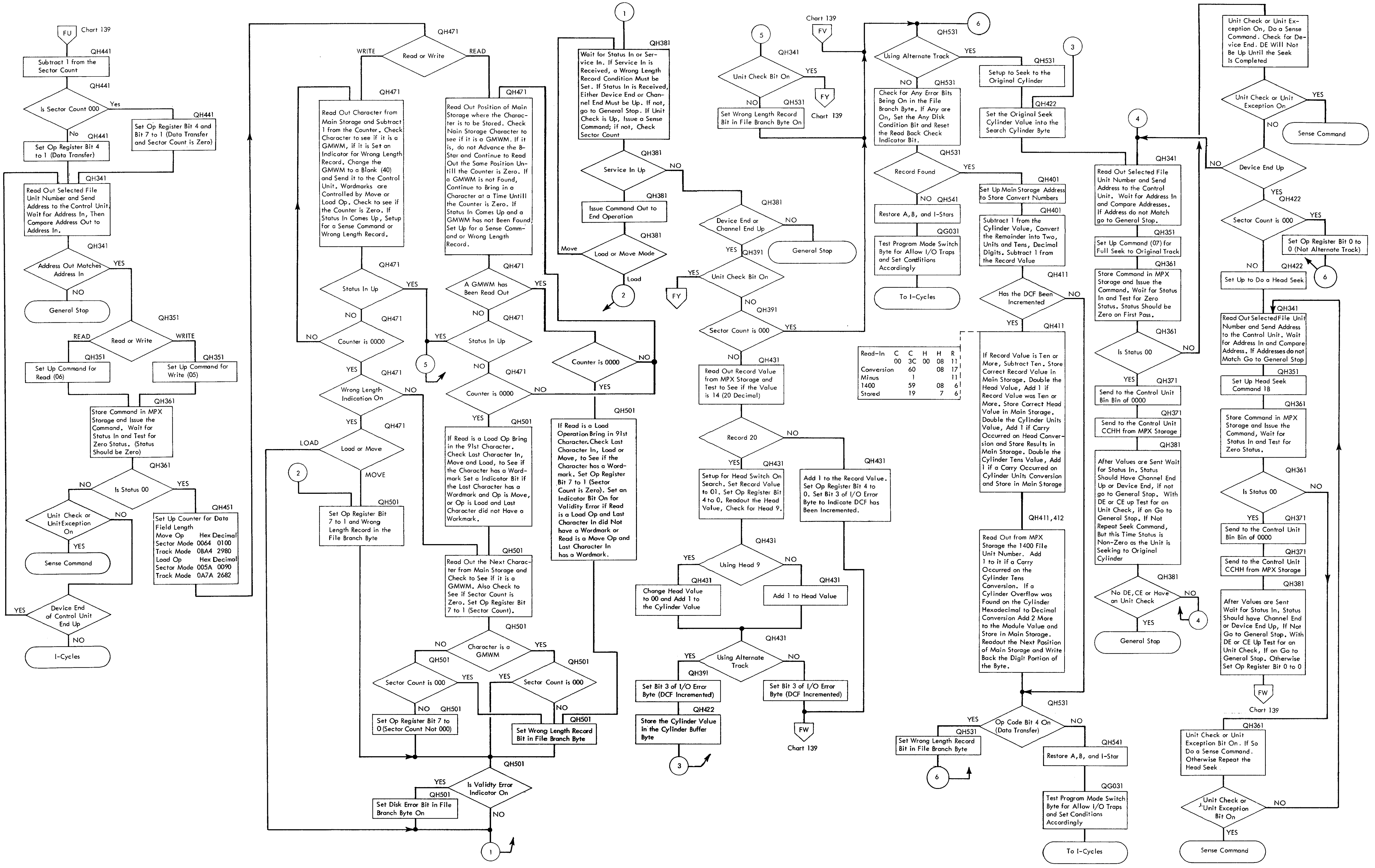


CLF 138 1400 File – R/W With Addresses Objectives – Sheet 2

CLF 139 1400 File – Ops 1, 2, and 5 Objectives



CLF 140 1400 File - Ops 1, 2, and 5 Objectives - Sheet 2



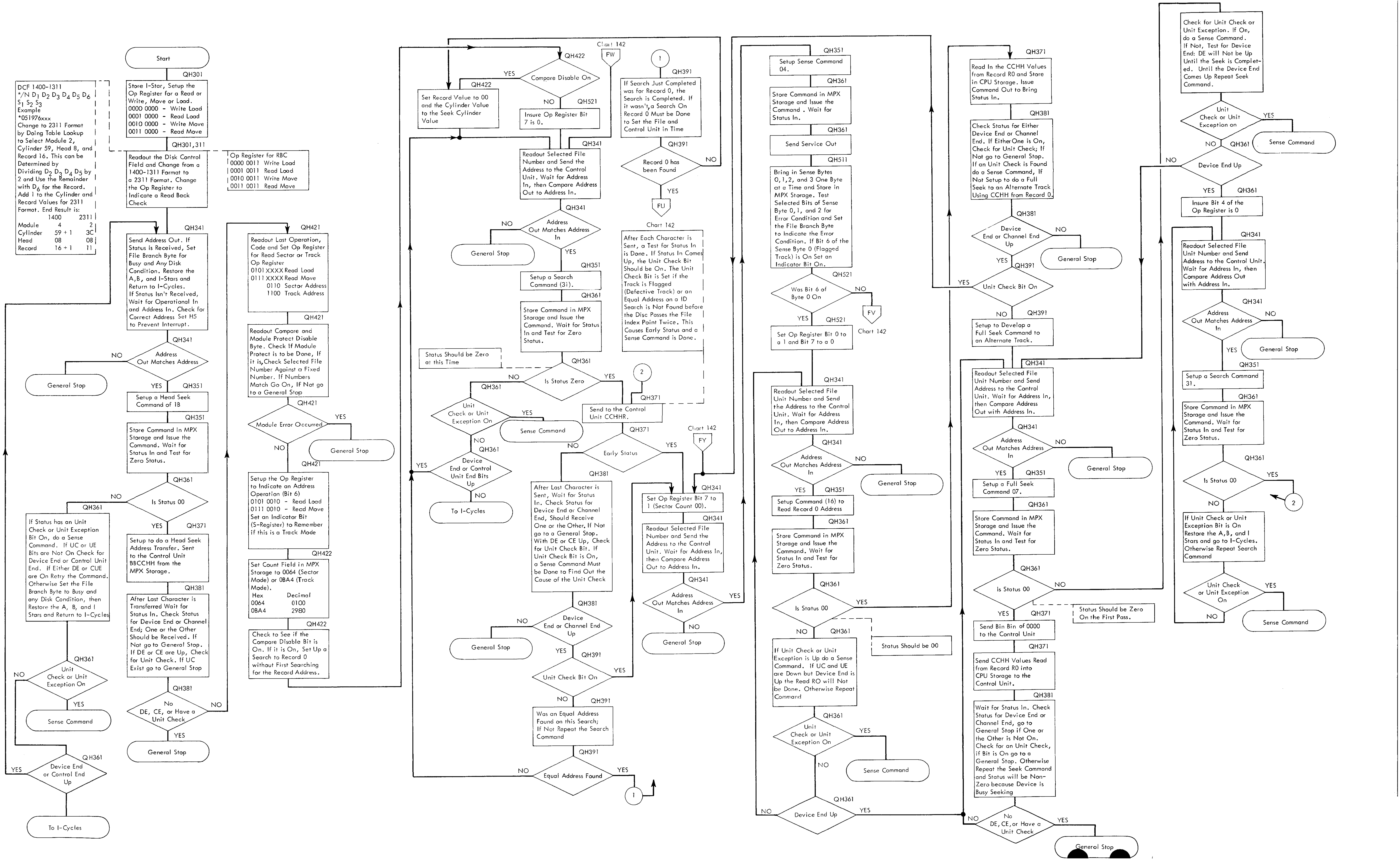
Set Up Counter for Data Field Length

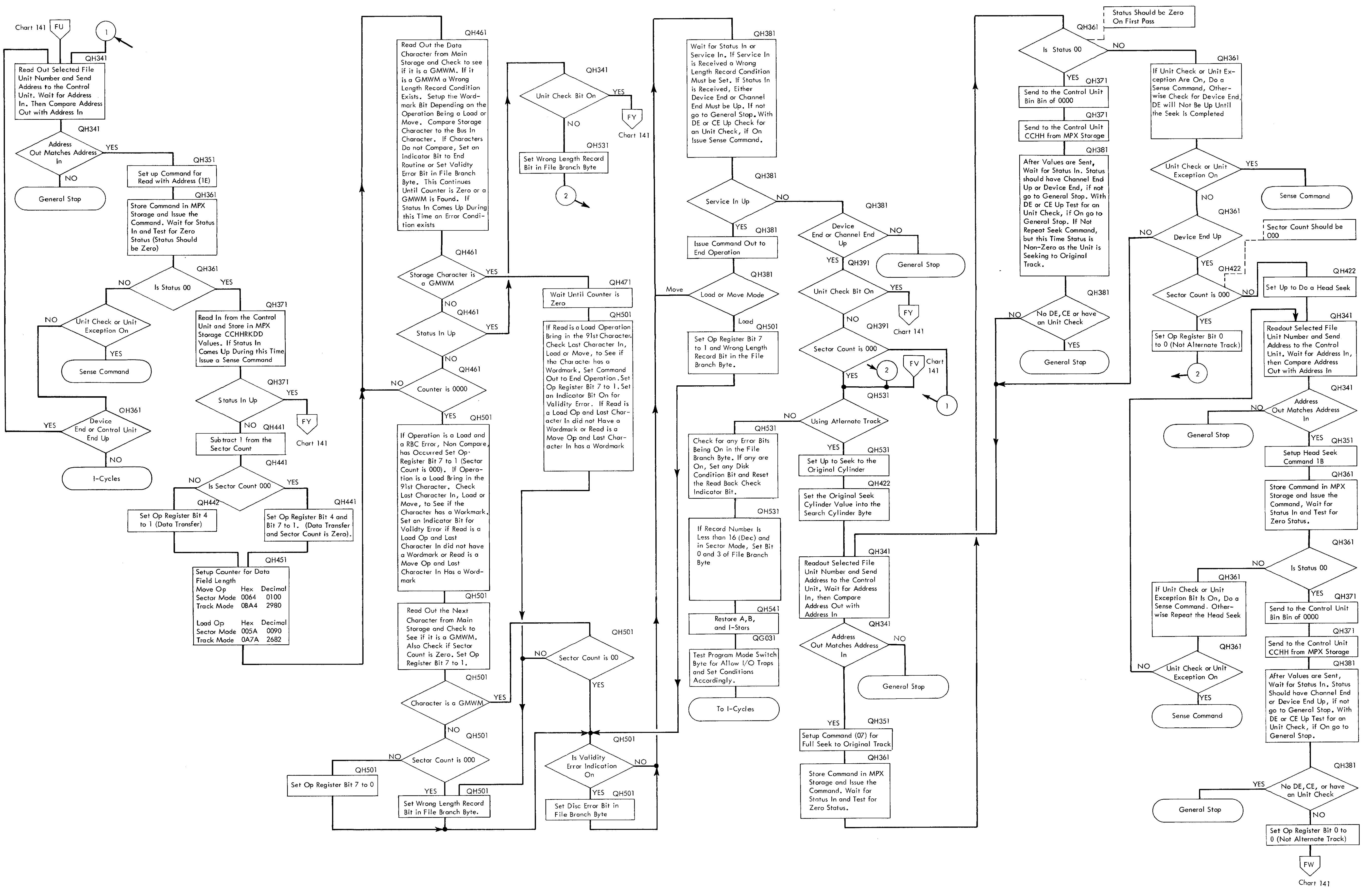
Move Op	Hex	Decimal
Sector Mode	0064	0100
Track Mode	0BA4	2980

Load Op	Hex	Decimal
Sector Mode	005A	0090
Track Mode	0A7A	2682

Read-In	C	C	H	H	R
	00	3C	00	08	11
Conversion		60		08	17
Minus		1			11
1400		59		08	6
Stored		19		7	6

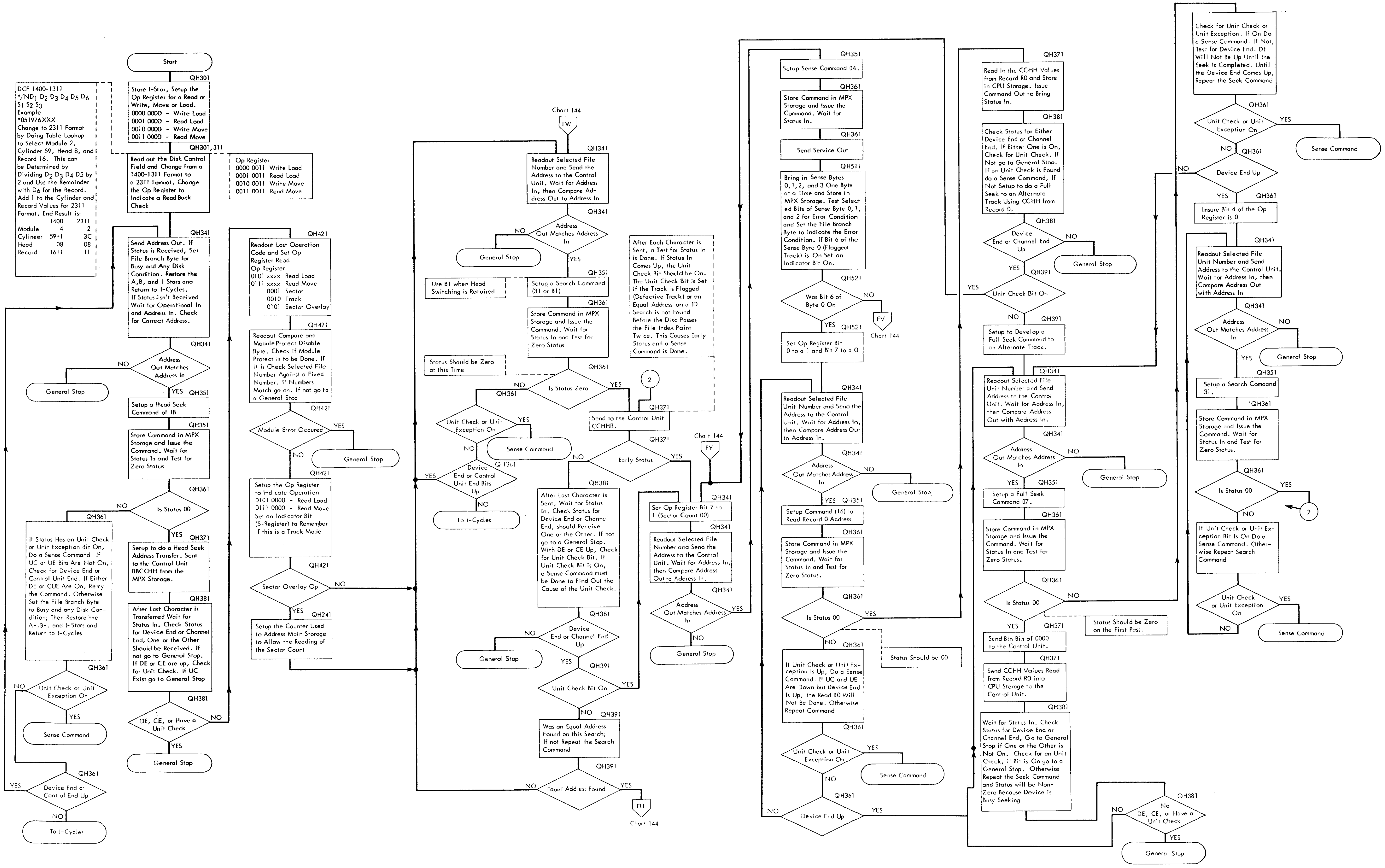
CLF 141 1400 File – RBC With Addresses Objectives



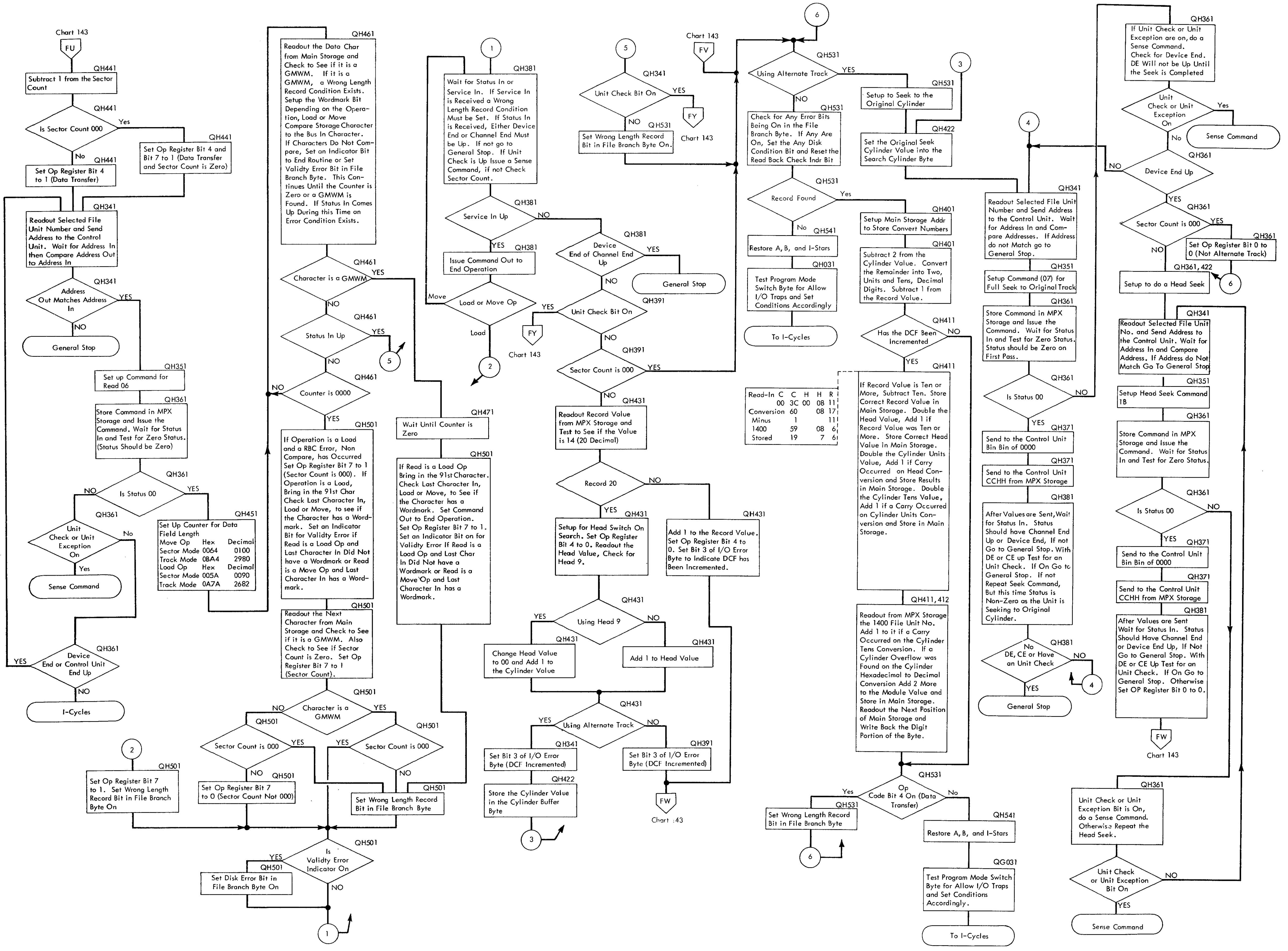


CLF 142 1400 File - RBC With Addresses Objectives - Sheet 2

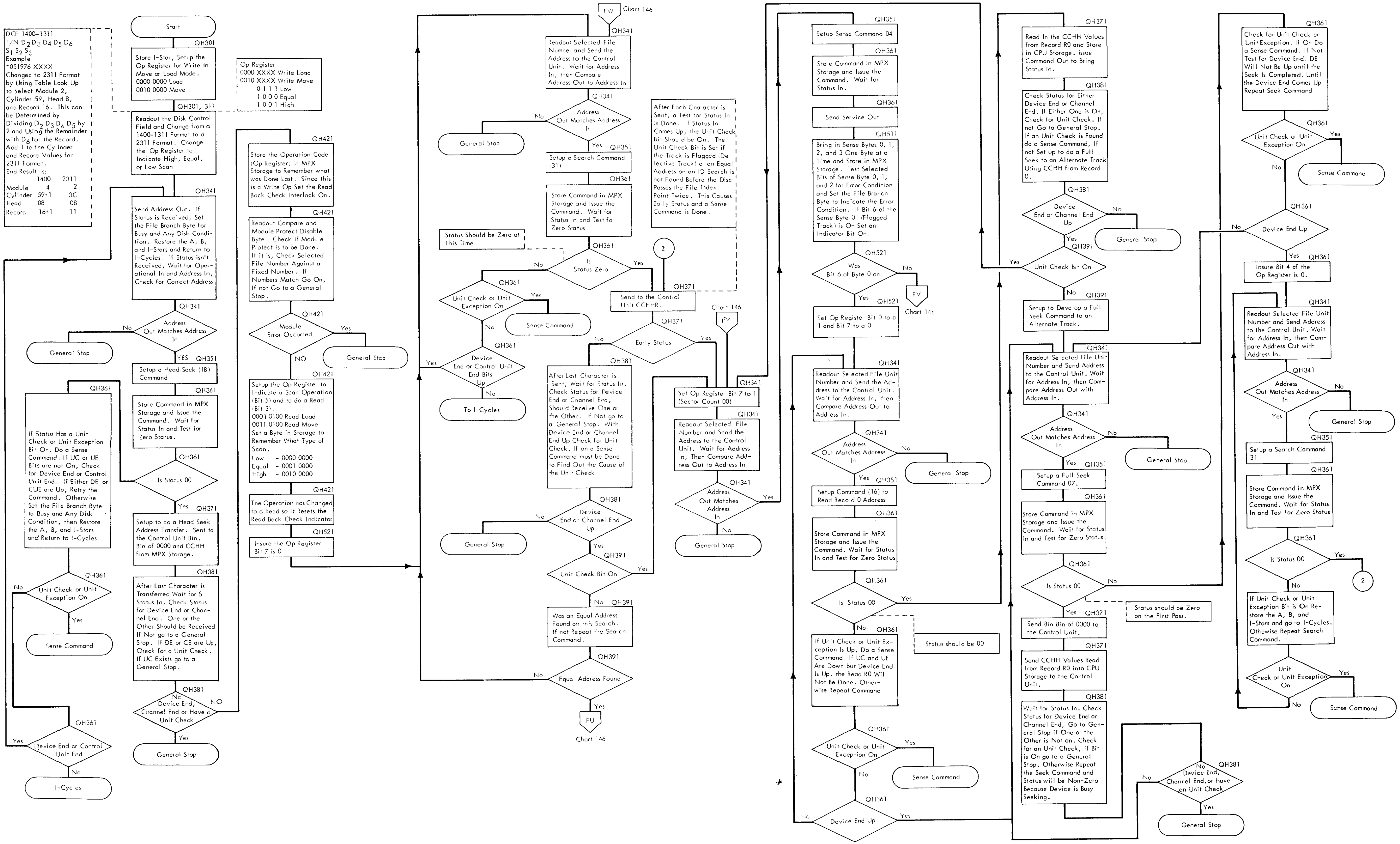
CLF 143 1400 File - RBC for 1, 2, and 5 Ops Objectives

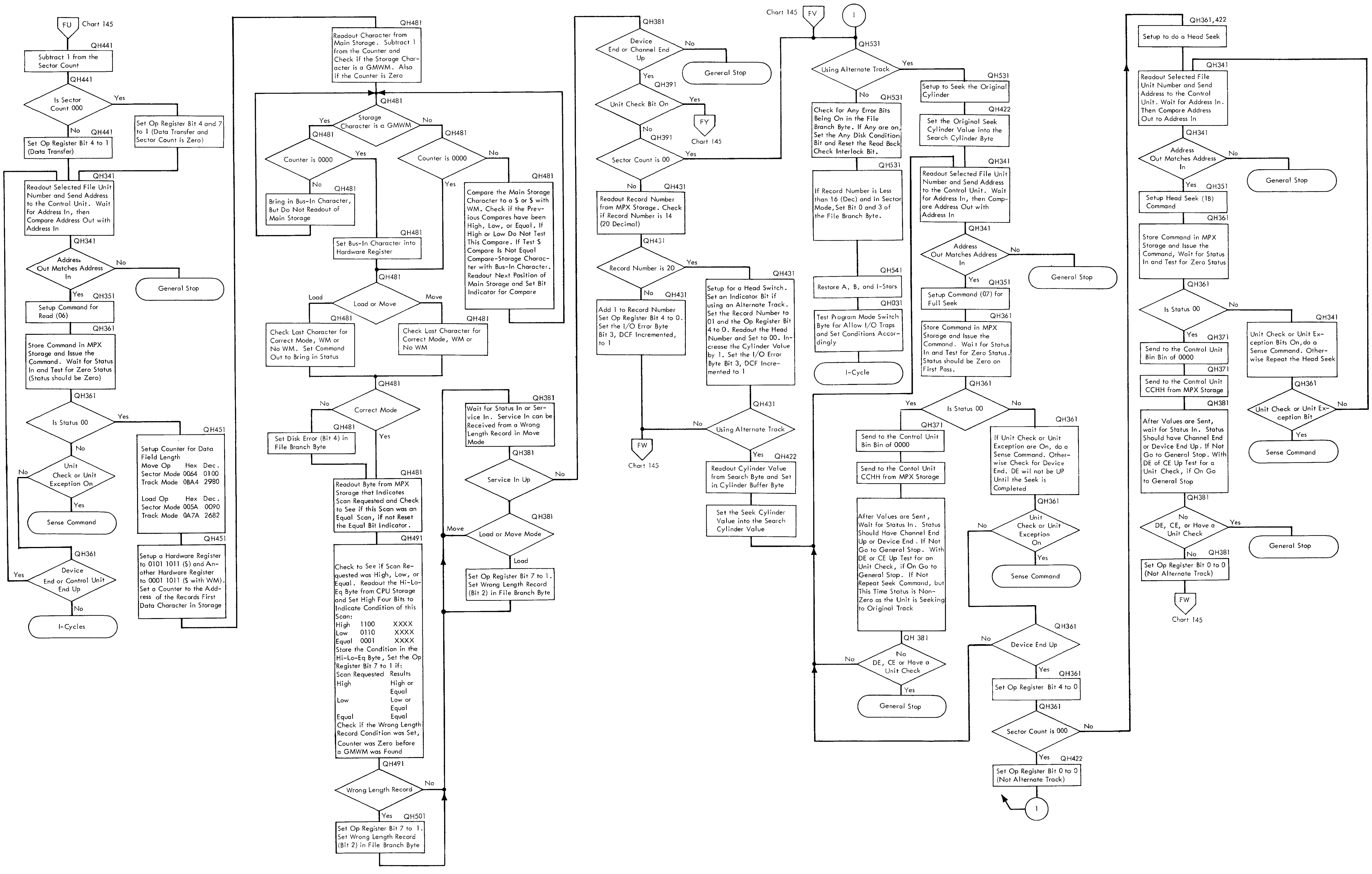


CLF 144 1400 File - RBC for 1, 2, and 5 Ops Objectives - Sheet 2

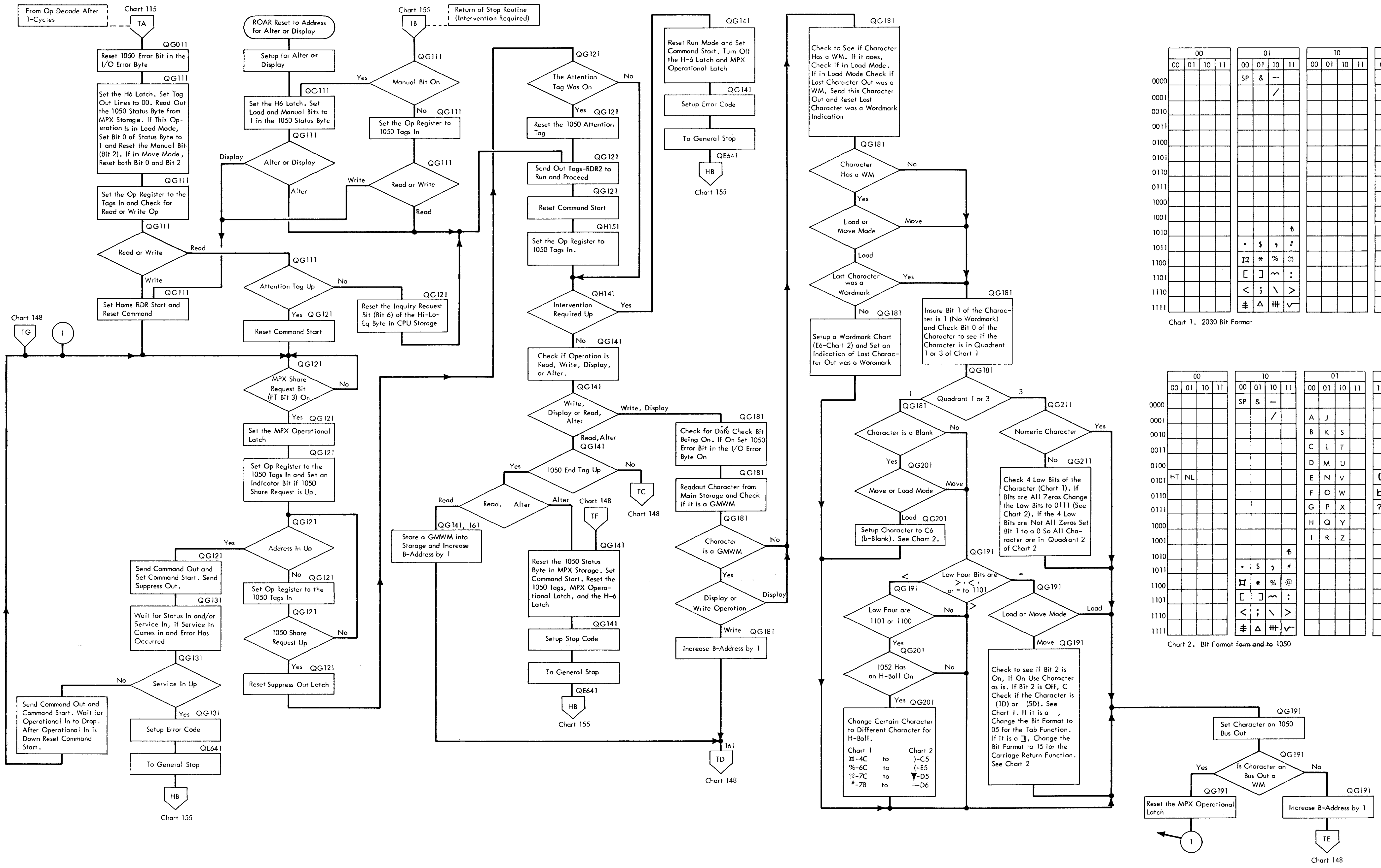


1400 File - Scan Op Objectives





CLF 146 1400 File - Scan Op Objectives - Sheet 2



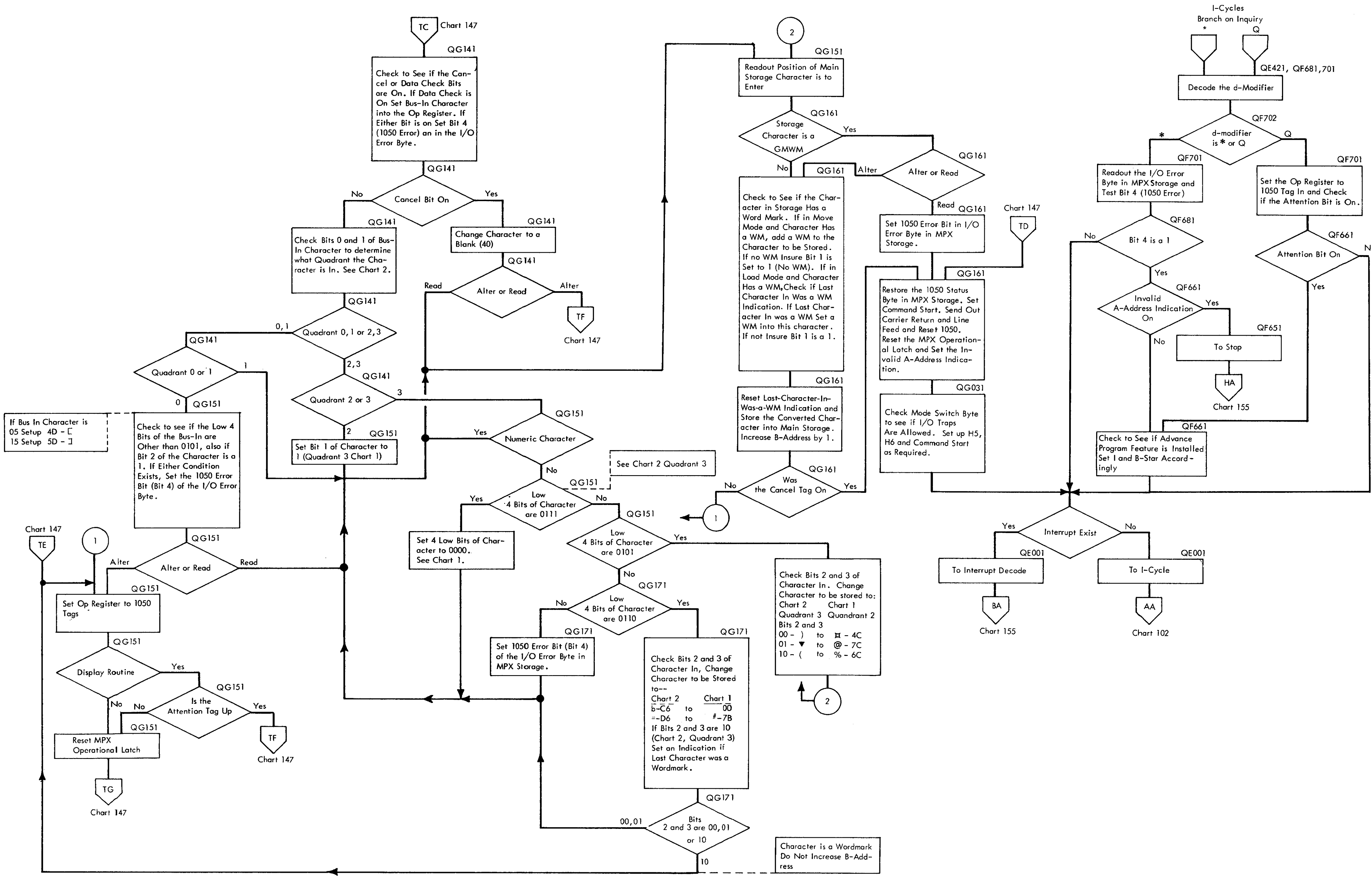
	00				01				10				11			
	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0000					SP	&	—						?	!	‡	0
0001							/						A	J		1
0010													B	K	S	2
0011													C	L	T	3
0100													D	M	U	4
0101													E	N	V	5
0110													F	O	W	6
0111													G	P	X	7
1000													H	Q	Y	8
1001													I	R	Z	9
1010								ƀ								
1011					.	$	,	#								
1100					⌑	*	%	@								
1101					[	]	⌢	:								
1110					<	;	\	>								
1111					⧧	Δ	⧻	√								

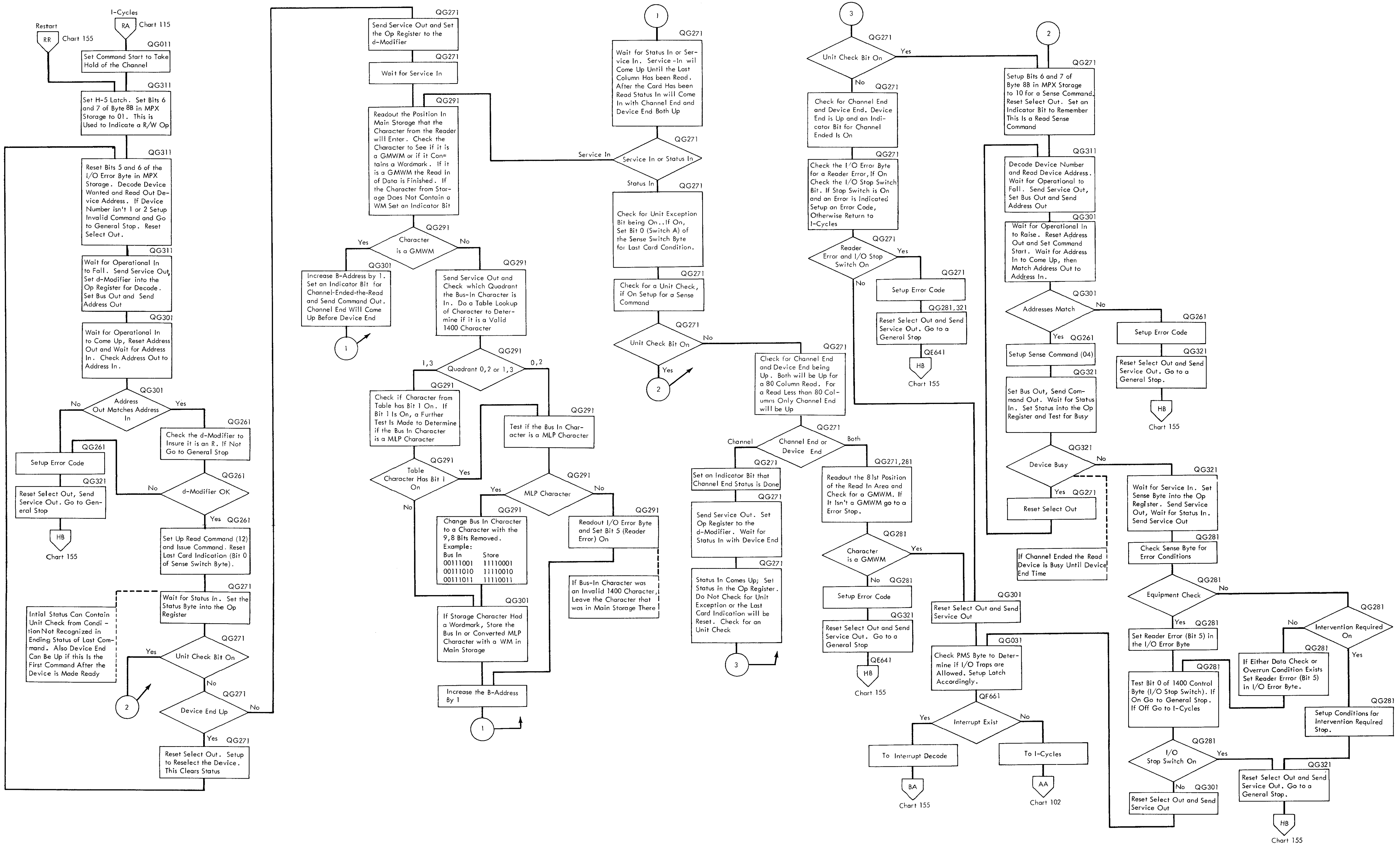
Chart 1. 2030 Bit Format

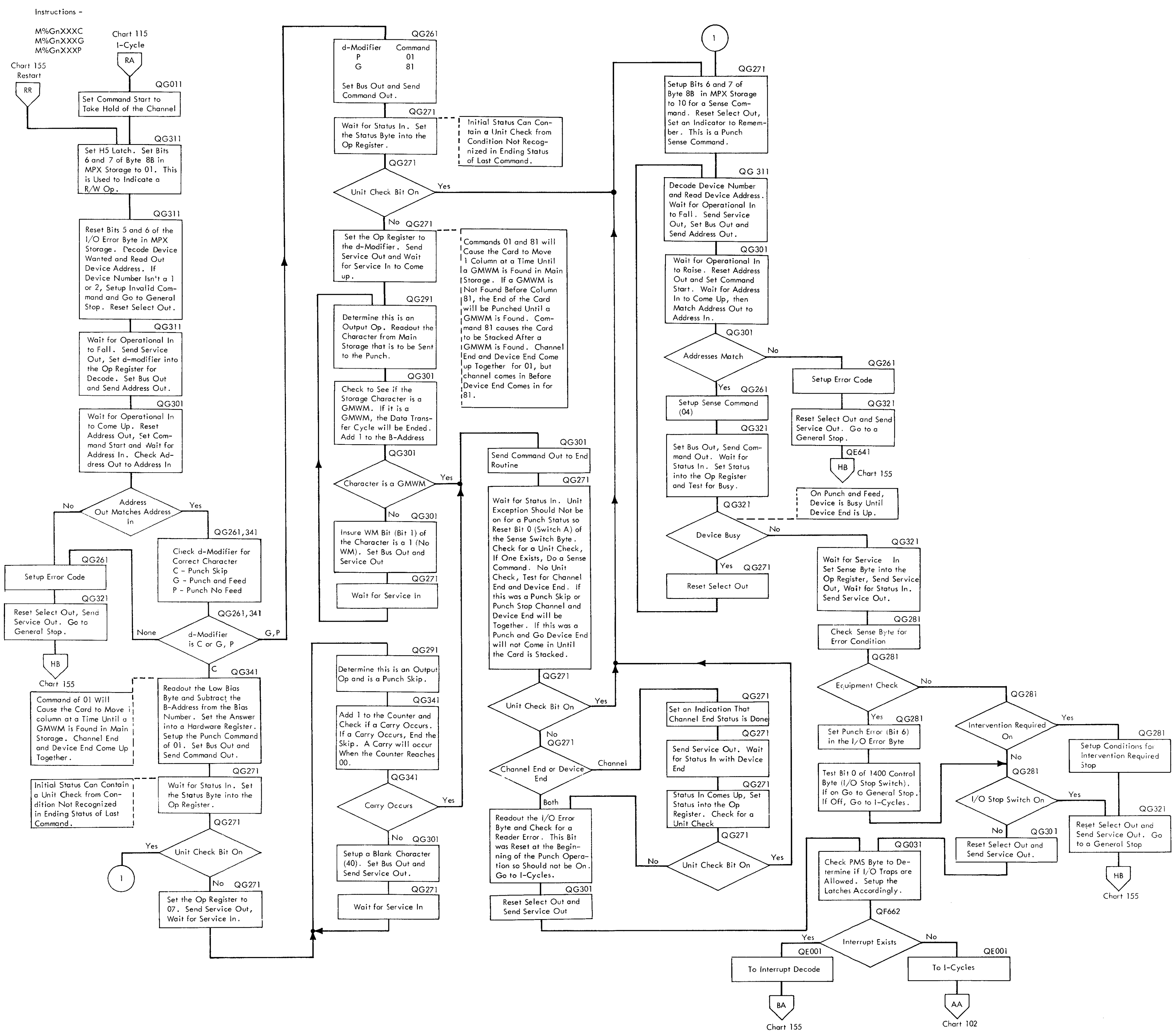
	00				10				01				11			
	00	01	10	11	00	01	10	11	00	01	10	11	10	01	10	11
0000					SP	&	—									0
0001							/		A	J						1
0010									B	K	S					2
0011									C	L	T					3
0100									D	M	U					4
0101	HT	NL							E	N	V		(	▼	)	5
0110									F	O	W		b	=	∨	6
0111									G	P	X		?	!	‡	7
1000									H	Q	Y					8
1001									I	R	Z					9
1010								ƀ								
1011					.	$	,	#								
1100					⌑	*	%	@								
1101					[	]	⌢	:								
1110					<	;	\	>								
1111					⧧	Δ	⧻	√								

Chart 2. Bit Format form and to 1050

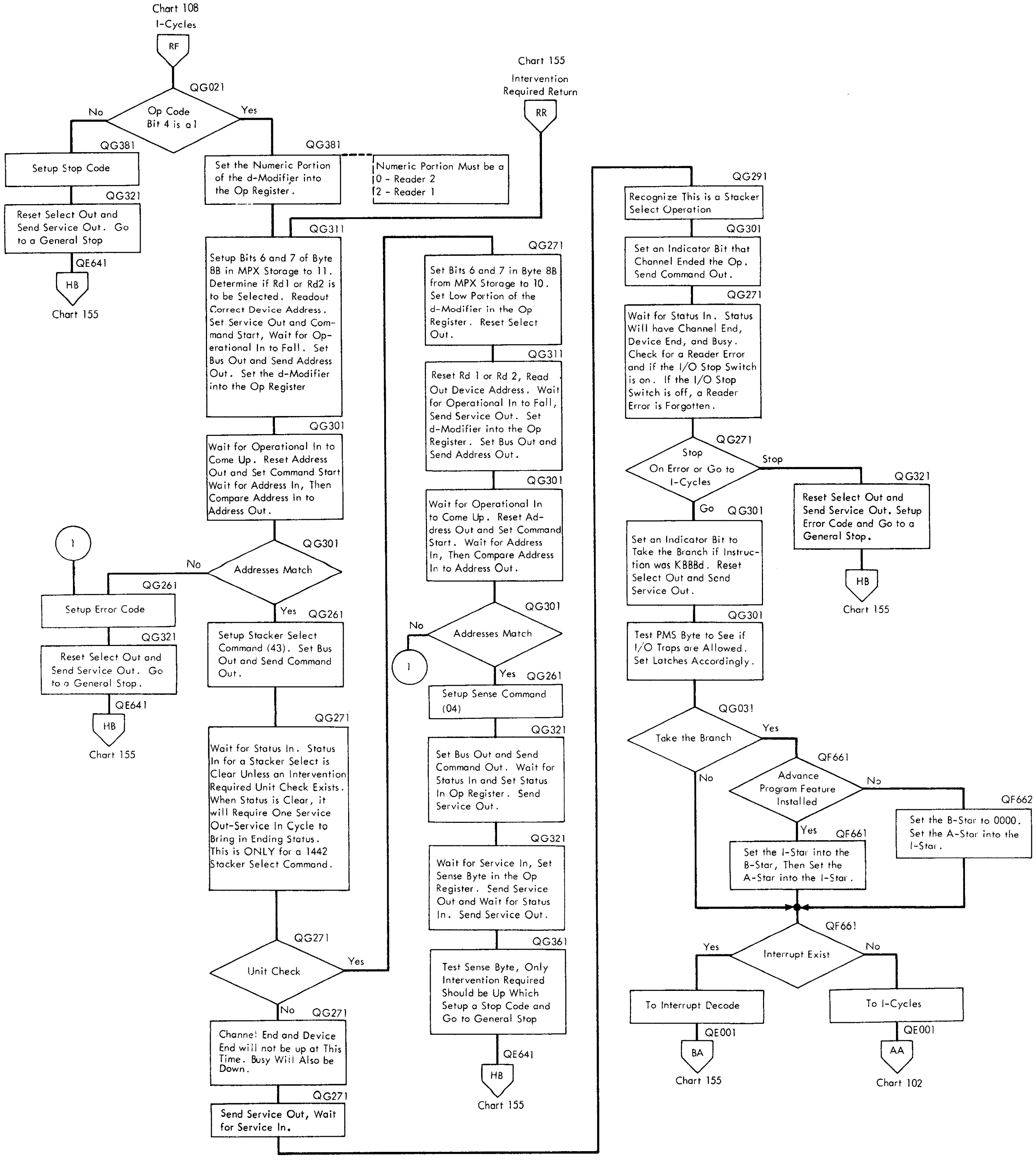
CLF 148 1400 - Console and 1050, Branch on Inquiry




Restart
RR
Chart 155
I-Cycles
RA
Chart 115
QG011
Set Command Start to Take Hold of the Channel
QG311
Set H-5 Latch. Set Bits 6 and 7 of Byte 8B in MPX Storage to 01. This is Used to Indicate a R/W Op
QG311
Reset Bits 5 and 6 of the I/O Error Byte in MPX Storage. Decode Device Wanted and Read Out Device Address. If Device Number isn't 1 or 2 Setup Invalid Command and Go to General Stop. Reset Select Out.
QG311
Wait for Operational In to Fall. Send Service Out, Set d-Modifier into the Op Register for Decode. Set Bus Out and Send Address Out
QG301
Wait for Operational In to Come Up, Reset Address Out and Wait for Address In. Check Address Out to Address In.
QG301
Address Out Matches Address In
No
Yes
QG261
Setup Error Code
QG321
Reset Select Out, Send Service Out. Go to General Stop
HB
Chart 155
QG261
Check the d-Modifier to Insure it is an R. If Not Go to General Stop
QG261
d-Modifier OK
No
Yes
QG261
Set Up Read Command (12) and Issue Command. Reset Last Card Indication (Bit 0 of Sense Switch Byte).
QG271
Wait for Status In. Set the Status Byte into the Op Register
Intial Status Can Contain Unit Check from Condition Not Recognized in Ending Status of Last Command. Also Device End Can Be Up if this Is the First Command After the Device is Made Ready
QG271
Unit Check Bit On
Yes
No
2
QG271
Device End Up
No
Yes
QG271
Reset Select Out. Setup to Reselect the Device. This Clears Status
QG271
Send Service Out and Set the Op Register to the d-Modifier
QG271
Wait for Service In
QG291
Readout the Position In Main Storage that the Character from the Reader will Enter. Check the Character to See if it is a GMWM or if it Contains a Wordmark. If it is a GMWM the Read In of Data is Finished. If the Character from Storage Does Not Contain a WM Set an Indicator Bit
QG291
Character is a GMWM
Yes
No
QG301
Increase B-Address by 1. Set an Indicator Bit for Channel-Ended-the-Read and Send Command Out. Channel End Will Come Up Before Device End
1
QG291
Send Service Out and Check which Quadrant the Bus-In Character is In. Do a Table Lookup of Character to Determine if it is a Valid 1400 Character
QG291
Quadrant 0,2 or 1,3
1,3
0,2
QG291
Check if Character from Table has Bit 1 On. If Bit 1 Is On, a Further Test Is Made to Determine if the Bus In Character is a MLP Character
QG291
Table Character Has Bit 1 On
Yes
No
QG291
Test if the Bus In Character is a MLP Character
QG291
MLP Character
Yes
No
QG291
Change Bus In Character to a Character with the 9,8 Bits Removed.
Example:
Bus In Store
00111001 11110001
00111010 11110010
00111011 11110011
QG291
Readout I/O Error Byte and Set Bit 5 (Reader Error) On
If Bus-In Character was an Invalid 1400 Character, Leave the Character that was in Main Storage There
QG301
If Storage Character Had a Wordmark, Store the Bus In or Converted MLP Character with a WM in Main Storage
Increase the B-Address By 1
1
1
QG271
Wait for Status In or Service In. Service-In will Come Up Until the Last Column Has been Read. After the Card Has been Read Status In will Come In with Channel End and Device End Both Up
QG271
Service In
Service In or Status In
Status In
QG271
Check for Unit Exception Bit being On. If On, Set Bit 0 (Switch A) of the Sense Switch Byte for Last Card Condition.
QG271
Check for a Unit Check, if On Setup for a Sense Command
QG271
Unit Check Bit On
No
Yes
2
QG271
Check for Channel End and Device End being Up. Both will be Up for a 80 Column Read. For a Read Less than 80 Columns Only Channel End will be Up
QG271
Channel End or Device End
Channel
Both
QG271
Set an Indicator Bit that Channel End Status is Done
QG271
Send Service Out. Set Op Register to the d-Modifier. Wait for Status In with Device End
QG271
Status In Comes Up; Set Status in the Op Register. Do Not Check for Unit Exception or the Last Card Indication will be Reset. Check for an Unit Check
3
QG271,281
Readout the 81st Position of the Read In Area and Check for a GMWM. If It Isn't a GMWM go to a Error Stop.
QG281
Character is a GMWM
Yes
No
QG281
Setup Error Code
QG321
Reset Select Out and Send Service Out. Go to a General Stop
QE641
HB
Chart 155
3
QG271
Unit Check Bit On
Yes
No
QG271
Check for Channel End and Device End. Device End is Up and an Indicator Bit for Channel Ended Is On
QG271
Check the I/O Error Byte for a Reader Error, If On Check the I/O Stop Switch Bit. If Stop Switch is On and an Error is Indicated Setup an Error Code, Otherwise Return to I-Cycles
QG271
Reader Error and I/O Stop Switch On
Yes
No
QG271
Setup Error Code
QG281,321
Reset Select Out and Send Service Out. Go to a General Stop
QE641
HB
Chart 155
QG301
Reset Select Out and Send Service Out
QG031
Check PMS Byte to Determine if I/O Traps are Allowed. Setup Latch Accordingly.
QF661
Interrupt Exist
Yes
No
To Interrupt Decode
BA
Chart 155
To I-Cycles
AA
Chart 102
2
QG271
Setup Bits 6 and 7 of Byte 8B in MPX Storage to 10 for a Sense Command. Reset Select Out. Set an Indicator Bit to Remember This Is a Read Sense Command
QG311
Decode Device Number and Read Device Address. Wait for Operational to Fall. Send Service Out, Set Bus Out and Send Address Out
QG301
Wait for Operational In to Raise. Reset Address Out and Set Command Start. Wait for Address In to Come Up, then Match Address Out to Address In.
QG301
Addresses Match
No
Yes
QG261
Setup Sense Command (04)
QG321
Set Bus Out, Send Command Out. Wait for Status In. Set Status into the Op Register and Test for Busy
QG261
Setup Error Code
QG321
Reset Select Out and Send Service Out. Go to a General Stop.
HB
Chart 155
QG321
Device Busy
No
Yes
QG271
Reset Select Out
If Channel Ended the Read Device is Busy Until Device End Time
QG321
Wait for Service In. Set Sense Byte into the Op Register. Send Service Out, Wait for Status In. Send Service Out
QG281
Check Sense Byte for Error Conditions
QG281
Equipment Check
No
Yes
QG281
Set Reader Error (Bit 5) in the I/O Error Byte
QG281
Intervention Required On
No
Yes
QG281
If Either Data Check or Overrun Condition Exists Set Reader Error (Bit 5) in I/O Error Byte.
QG281
Setup Conditions for Intervention Required Stop.
QG281
Test Bit 0 of 1400 Control Byte (I/O Stop Switch). If On Go to General Stop. If Off Go to I-Cycles
QG281
I/O Stop Switch On
Yes
No
QG301
Reset Select Out and Send Service Out
QG321
Reset Select Out and Send Service Out. Go to a General Stop.
HB
Chart 155

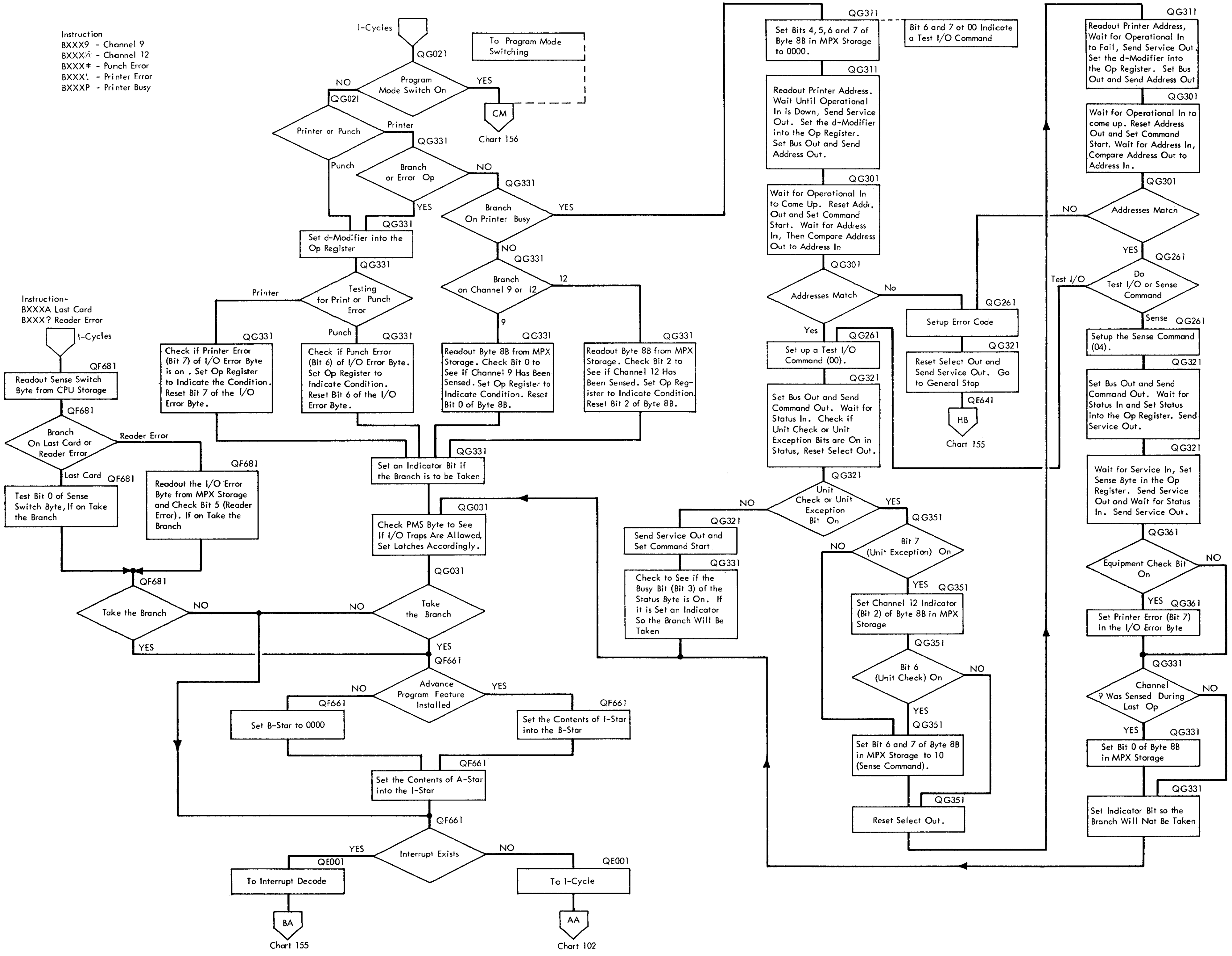

Instructions -
M%GnXXXC
M%GnXXXG
M%GnXXXP
Chart 155 Restart
RR
Chart 115 I-Cycle
RA
QG011
Set Command Start to Take Hold of the Channel
QG311
Set H5 Latch. Set Bits 6 and 7 of Byte 8B in MPX Storage to 01. This is Used to Indicate a R/W Op.
QG311
Reset Bits 5 and 6 of the I/O Error Byte in MPX Storage. Decode Device Wanted and Read Out Device Address. If Device Number Isn't a 1 or 2, Setup Invalid Command and Go to General Stop. Reset Select Out.
QG311
Wait for Operational In to Fall. Send Service Out, Set d-modifier into the Op Register for Decode. Set Bus Out and Send Address Out.
QG301
Wait for Operational In to Come Up. Reset Address Out, Set Command Start and Wait for Address In. Check Address Out to Address In
Address Out Matches Address In
No
Yes
QG261
Setup Error Code
QG321
Reset Select Out, Send Service Out. Go to General Stop.
HB
Chart 155
Command of 01 Will Cause the Card to Move 1 column at a Time Until a GMWM is Found in Main Storage. Channel End and Device End Come Up Together.
Initial Status Can Contain a Unit Check from Condition Not Recognized in Ending Status of Last Command.
QG261,341
Check d-Modifier for Correct Character
C - Punch Skip
G - Punch and Feed
P - Punch No Feed
QG261,341
d-Modifier is C or G, P
None
G,P
C
QG341
Readout the Low Bias Byte and Subtract the B-Address from the Bias Number. Set the Answer into a Hardware Register. Setup the Punch Command of 01. Set Bus Out and Send Command Out.
QG271
Wait for Status In. Set the Status Byte into the Op Register.
QG271
Unit Check Bit On
Yes
1
No
QG271
Set the Op Register to 07. Send Service Out, Wait for Service In.
QG261
d-Modifier Command
P 01
G 81
Set Bus Out and Send Command Out.
QG271
Wait for Status In. Set the Status Byte into the Op Register.
Initial Status Can Contain a Unit Check from Condition Not Recognized in Ending Status of Last Command.
QG271
Unit Check Bit On
Yes
No
QG271
Set the Op Register to the d-Modifier. Send Service Out and Wait for Service In to Come up.
Commands 01 and 81 will Cause the Card to Move 1 Column at a Time Until a GMWM is Found in Main Storage. If a GMWM is Not Found Before Column 81, the End of the Card will be Punched Until a GMWM is Found. Command 81 causes the Card to be Stacked After a GMWM is Found. Channel End and Device End Come up Together for 01, but channel comes in Before Device End Comes in for 81.
QG291
Determine this is an Output Op. Readout the Character from Main Storage that is to be Sent to the Punch.
QG301
Check to See if the Storage Character is a GMWM. If it is a GMWM, the Data Transfer Cycle will be Ended. Add 1 to the B-Address
QG301
Character is a GMWM
Yes
No
QG301
Insure WM Bit (Bit 1) of the Character is a 1 (No WM). Set Bus Out and Service Out
QG271
Wait for Service In
QG291
Determine this is an Output Op and is a Punch Skip.
QG341
Add 1 to the Counter and Check if a Carry Occurs. If a Carry Occurs, End the Skip. A Carry will occur When the Counter Reaches 00.
QG341
Carry Occurs
Yes
No
QG301
Setup a Blank Character (40). Set Bus Out and Send Service Out.
QG271
Wait for Service In
QG301
Send Command Out to End Routine
QG271
Wait for Status In. Unit Exception Should Not be on for a Punch Status so Reset Bit 0 (Switch A) of the Sense Switch Byte. Check for a Unit Check, If One Exists, Do a Sense Command. No Unit Check, Test for Channel End and Device End. If this was a Punch Skip or Punch Stop Channel and Device End will be Together. If this was a Punch and Go Device End will not Come in Until the Card is Stacked.
QG271
Unit Check Bit On
Yes
No
QG271
Channel End or Device End
Channel
Both
Readout the I/O Error Byte and Check for a Reader Error. This Bit was Reset at the Beginning of the Punch Operation so Should not be On. Go to I-Cycles.
QG301
Reset Select Out and Send Service Out
QG271
Set an Indication That Channel End Status is Done
QG271
Send Service Out. Wait for Status In with Device End
QG271
Status In Comes Up, Set Status into the Op Register. Check for a Unit Check
QG271
Unit Check Bit On
No
Yes
1
QG271
Setup Bits 6 and 7 of Byte 8B in MPX Storage to 10 for a Sense Command. Reset Select Out, Set an Indicator to Remember. This is a Punch Sense Command.
QG 311
Decode Device Number and Read Device Address. Wait for Operational In to Fall. Send Service Out, Set Bus Out and Send Address Out.
QG301
Wait for Operational In to Raise. Reset Address Out and Set Command Start. Wait for Address In to Come Up, then Match Address Out to Address In.
QG301
Addresses Match
No
Yes
QG261
Setup Sense Command (04)
QG321
Set Bus Out, Send Command Out. Wait for Status In. Set Status into the Op Register and Test for Busy.
QG321
Device Busy
Yes
No
QG271
Reset Select Out
QG261
Setup Error Code
QG321
Reset Select Out and Send Service Out. Go to a General Stop.
QE641
HB
Chart 155
On Punch and Feed, Device is Busy Until Device End is Up.
QG321
Wait for Service In Set Sense Byte into the Op Register, Send Service Out, Wait for Status In. Send Service Out.
QG281
Check Sense Byte for Error Condition
QG281
Equipment Check
Yes
No
QG281
Set Punch Error (Bit 6) in the I/O Error Byte
Test Bit 0 of 1400 Control Byte (I/O Stop Switch). If on Go to General Stop. If Off, Go to I-Cycles.
QG281
Intervention Required On
Yes
No
QG281
Setup Conditions for Intervention Required Stop
QG321
Reset Select Out and Send Service Out. Go to a General Stop
HB
Chart 155
QG281
I/O Stop Switch On
Yes
No
QG301
Reset Select Out and Send Service Out.
QG031
Check PMS Byte to Determine if I/O Traps are Allowed. Setup the Latches Accordingly.
QF662
Interrupt Exists
Yes
No
QE001
To Interrupt Decode
BA
Chart 155
QE001
To I-Cycles
AA
Chart 102

Instruction Kd, KBBBd



CLF 151 1442 - Stacker Select Objective

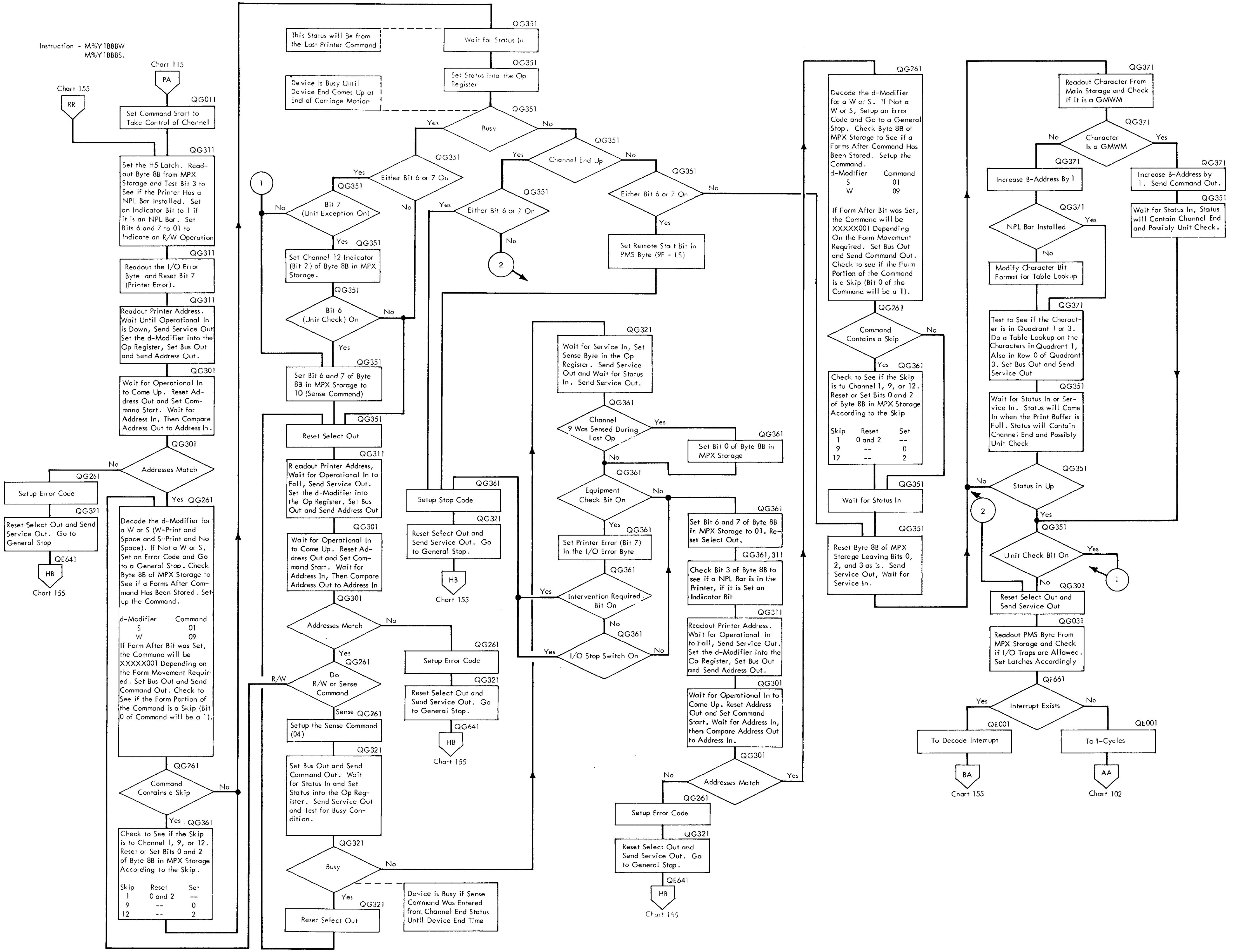
CLF 152 1442-1443 Branch on Condition

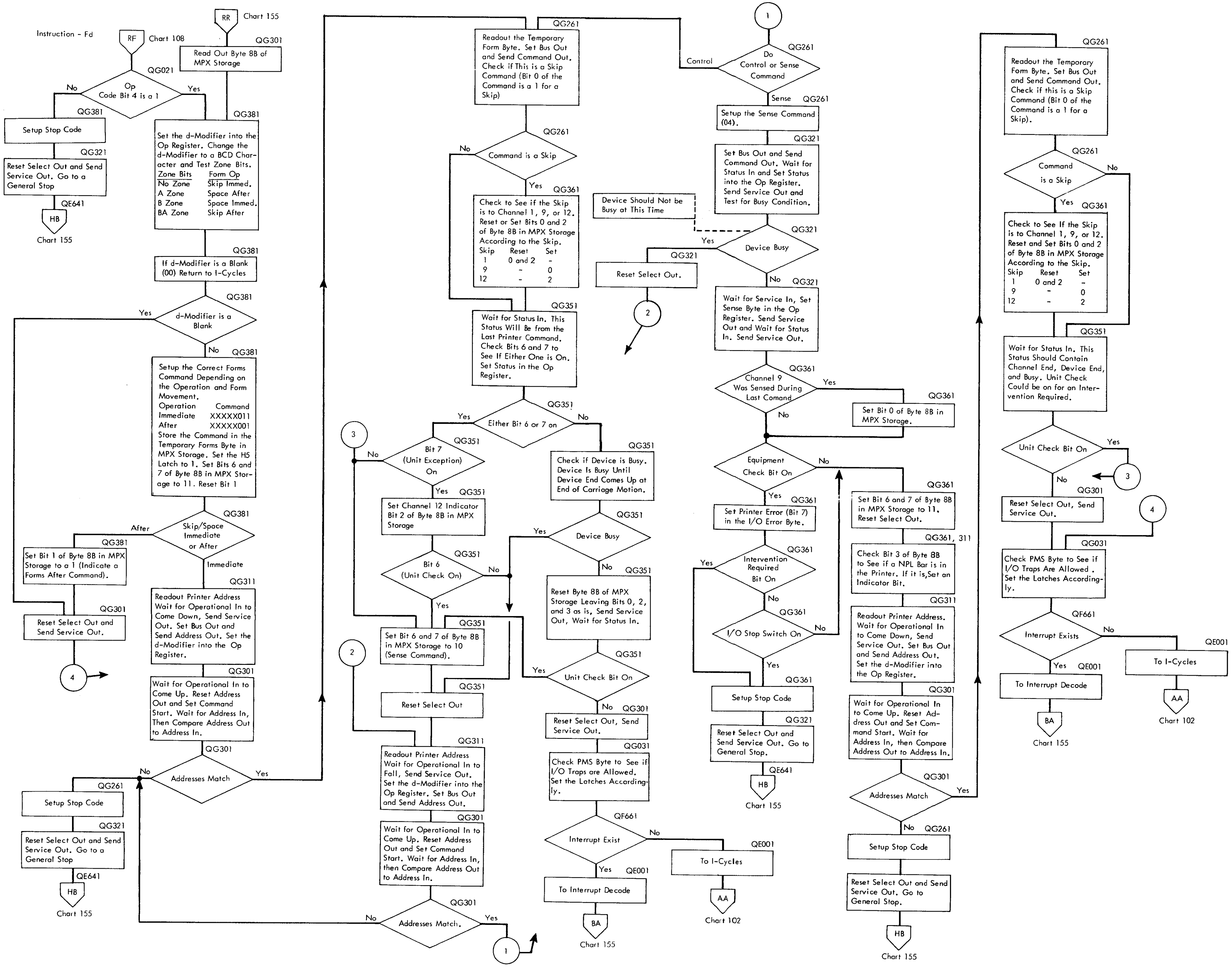


Instruction
BXXX9 - Channel 9
BXXX@ - Channel 12
BXXX# - Punch Error
BXXX! - Printer Error
BXXXP - Printer Busy

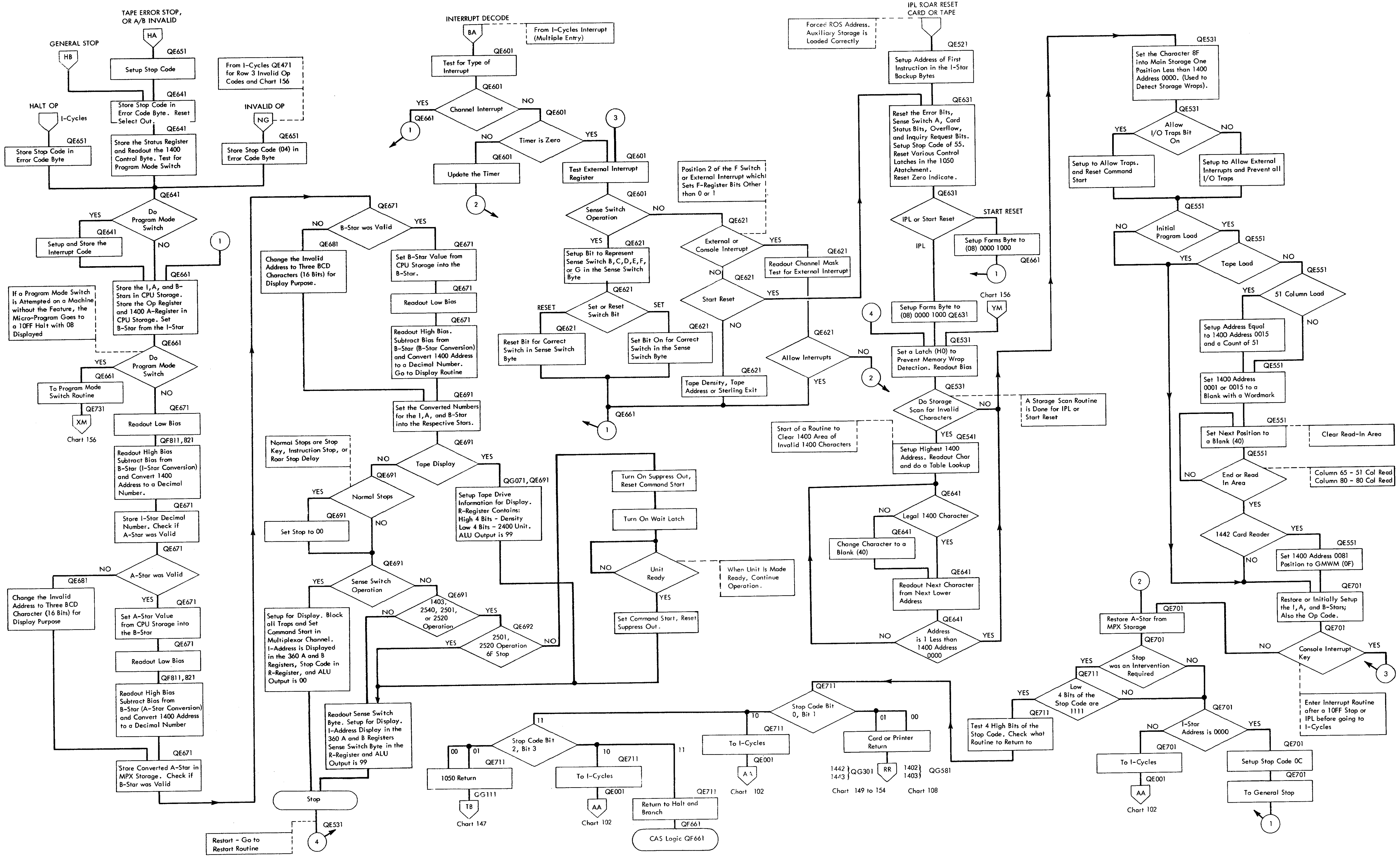
Instruction-
BXXXA Last Card
BXXX? Reader Error

1443 Print Objective

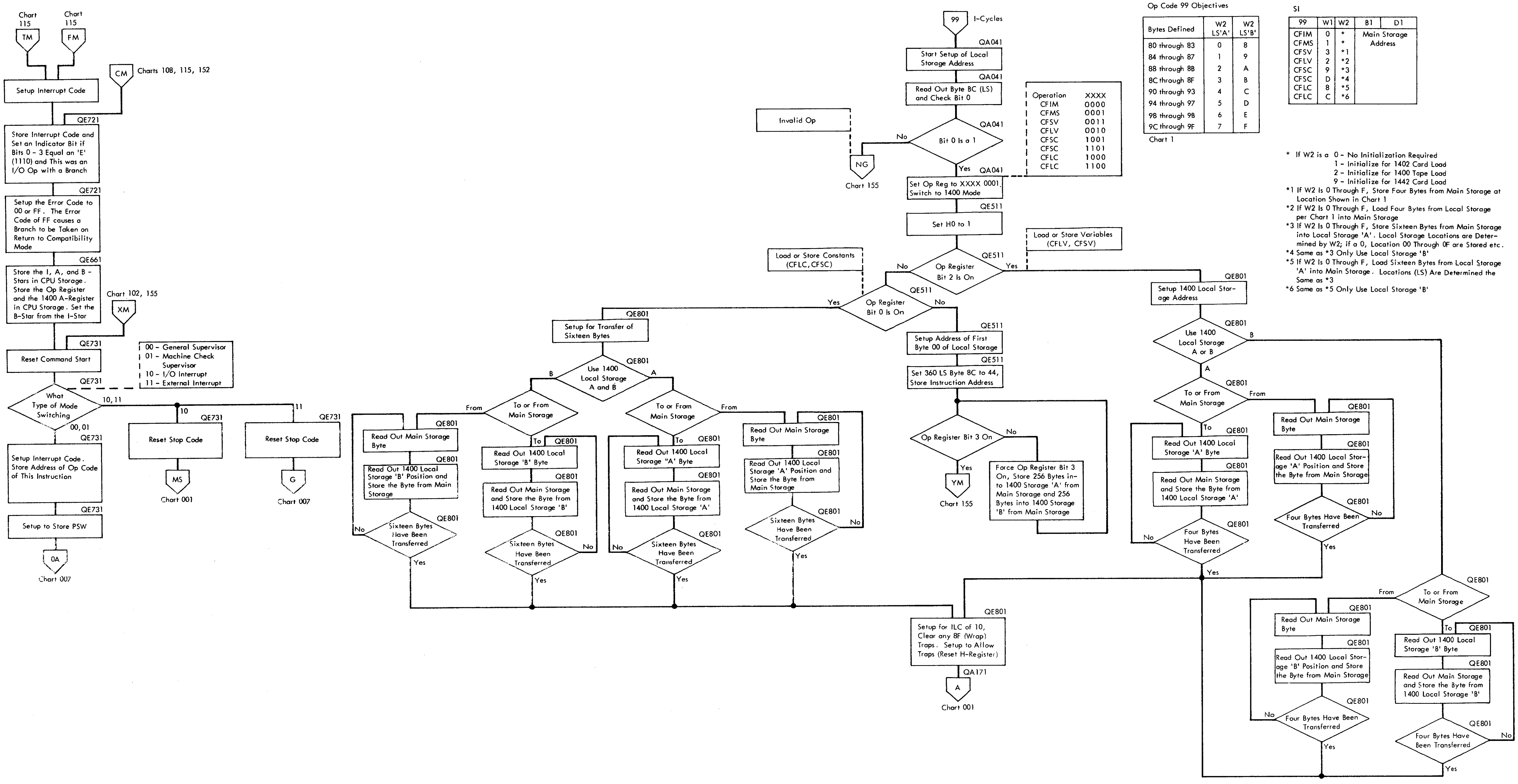



Instruction - Fd
RF
Chart 108
QG021
Op Code Bit 4 is a 1
No
Yes
QG381
Setup Stop Code
QG321
Reset Select Out and Send Service Out. Go to a General Stop
QE641
HB
Chart 155
RR
Chart 155
QG301
Read Out Byte 8B of MPX Storage
QG381
Set the d-Modifier into the Op Register. Change the d-Modifier to a BCD Character and Test Zone Bits.
Zone Bits / Form Op
No Zone / Skip Immed.
A Zone / Space After
B Zone / Space Immed.
BA Zone / Skip After
QG381
If d-Modifier is a Blank (00) Return to I-Cycles
QG381
d-Modifier is a Blank
Yes
No
QG381
Setup the Correct Forms Command Depending on the Operation and Form Movement.
Operation / Command
Immediate / XXXXX011
After / XXXXX001
Store the Command in the Temporary Forms Byte in MPX Storage. Set the H5 Latch to 1. Set Bits 6 and 7 of Byte 8B in MPX Storage to 11. Reset Bit 1
QG381
Skip/Space Immediate or After
After
Immediate
QG381
Set Bit 1 of Byte 8B in MPX Storage to a 1 (Indicate a Forms After Command).
QG301
Reset Select Out and Send Service Out.
4
QG311
Readout Printer Address Wait for Operational In to Come Down, Send Service Out. Set Bus Out and Send Address Out. Set the d-Modifier into the Op Register.
QG301
Wait for Operational In to Come Up. Reset Address Out and Set Command Start. Wait for Address In, Then Compare Address Out to Address In.
QG301
Addresses Match
No
Yes
QG261
Setup Stop Code
QG321
Reset Select Out and Send Service Out. Go to a General Stop
QE641
HB
Chart 155
QG261
Readout the Temporary Form Byte. Set Bus Out and Send Command Out. Check if This is a Skip Command (Bit 0 of the Command is a 1 for a Skip)
QG261
Command is a Skip
No
Yes
QG361
Check to See if the Skip is to Channel 1, 9, or 12. Reset or Set Bits 0 and 2 of Byte 8B in MPX Storage According to the Skip.
Skip / Reset / Set
1 / 0 and 2 / -
9 / - / 0
12 / - / 2
QG351
Wait for Status In. This Status Will Be from the Last Printer Command. Check Bits 6 and 7 to See If Either One is On. Set Status in the Op Register.
QG351
Either Bit 6 or 7 on
Yes
No
3
QG351
Bit 7 (Unit Exception) On
No
Yes
QG351
Set Channel 12 Indicator Bit 2 of Byte 8B in MPX Storage
QG351
Bit 6 (Unit Check On)
No
Yes
QG351
Set Bit 6 and 7 of Byte 8B in MPX Storage to 10 (Sense Command).
QG351
Reset Select Out
2
QG311
Readout Printer Address Wait for Operational In to Fall, Send Service Out. Set the d-Modifier into the Op Register. Set Bus Out and Send Address Out.
QG301
Wait for Operational In to Come Up. Reset Address Out and Set Command Start. Wait for Address In, then Compare Address Out to Address In.
QG301
Addresses Match.
No
Yes
1
QG351
Check if Device is Busy. Device Is Busy Until Device End Comes Up at End of Carriage Motion.
QG351
Device Busy
Yes
No
QG351
Reset Byte 8B of MPX Storage Leaving Bits 0, 2, and 3 as is, Send Service Out, Wait for Status In.
QG351
Unit Check Bit On
Yes
No
QG301
Reset Select Out, Send Service Out.
QG031
Check PMS Byte to See if I/O Traps are Allowed. Set the Latches Accordingly.
QF661
Interrupt Exist
No
Yes
QE001
To Interrupt Decode
BA
Chart 155
QE001
To I-Cycles
AA
Chart 102
1
QG261
Do Control or Sense Command
Control
Sense
QG261
Setup the Sense Command (04).
QG321
Set Bus Out and Send Command Out. Wait for Status In and Set Status into the Op Register. Send Service Out and Test for Busy Condition.
Device Should Not be Busy at This Time
QG321
Device Busy
Yes
No
QG321
Reset Select Out.
2
QG321
Wait for Service In, Set Sense Byte in the Op Register. Send Service Out and Wait for Status In. Send Service Out.
QG361
Channel 9 Was Sensed During Last Comand
Yes
No
QG361
Set Bit 0 of Byte 8B in MPX Storage.
Equipment Check Bit On
No
Yes
QG361
Set Printer Error (Bit 7) in the I/O Error Byte.
QG361
Intervention Required Bit On
Yes
No
QG361
I/O Stop Switch On
No
Yes
QG361
Setup Stop Code
QG321
Reset Select Out and Send Service Out. Go to General Stop.
QE641
HB
Chart 155
QG361
Set Bit 6 and 7 of Byte 8B in MPX Storage to 11. Reset Select Out.
QG361, 311
Check Bit 3 of Byte 8B to See if a NPL Bar is in the Printer. If it is, Set an Indicator Bit.
QG311
Readout Printer Address. Wait for Operational In to Come Down, Send Service Out. Set Bus Out and Send Address Out. Set the d-Modifier into the Op Register.
QG301
Wait for Operational In to Come Up. Reset Address Out and Set Command Start. Wait for Address In, then Compare Address Out to Address In.
QG301
Addresses Match
Yes
No
QG261
Setup Stop Code
Reset Select Out and Send Service Out. Go to General Stop.
HB
Chart 155
QG261
Readout the Temporary Form Byte. Set Bus Out and Send Command Out. Check if this is a Skip Command (Bit 0 of the Command is a 1 for a Skip).
QG261
Command is a Skip
No
Yes
QG361
Check to See If the Skip is to Channel 1, 9, or 12. Reset and Set Bits 0 and 2 of Byte 8B in MPX Storage According to the Skip.
Skip / Reset / Set
1 / 0 and 2 / -
9 / - / 0
12 / - / 2
QG351
Wait for Status In. This Status Should Contain Channel End, Device End, and Busy. Unit Check Could be on for an Intervention Required.
Unit Check Bit On
Yes
No
3
QG301
Reset Select Out, Send Service Out.
4
QG031
Check PMS Byte to See if I/O Traps Are Allowed. Set the Latches Accordingly.
QF661
Interrupt Exists
No
Yes
QE001
To Interrupt Decode
BA
Chart 155
QE001
To I-Cycles
AA
Chart 102

CLF 155 1400 Stops, Interrupts, Resets and IPL



CLF 156 Mode Switching, 99 Op



Operation	XXXX
CFIM	0000
CFMS	0001
CFSV	0011
CFLV	0010
CFSC	1001
CFSC	1101
CFLC	1000
CFLC	1100

Op Code 99 Objectives

Bytes Defined	W2 LS'A'	W2 LS'B'
80 through 83	0	8
84 through 87	1	9
88 through 8B	2	A
8C through 8F	3	B
90 through 93	4	C
94 through 97	5	D
98 through 9B	6	E
9C through 9F	7	F

Chart 1

SI

99	W1	W2	B1	D1
CFIM	0	*	Main Storage Address	
CFMS	1	*		
CFSV	3	*1		
CFLV	2	*2		
CFSC	9	*3		
CFSC	D	*4		
CFLC	8	*5		
CFLC	C	*6		

* If W2 is a 0 - No Initialization Required
1 - Initialize for 1402 Card Load
2 - Initialize for 1400 Tape Load
9 - Initialize for 1442 Card Load

*1 If W2 Is 0 Through F, Store Four Bytes from Main Storage at Location Shown in Chart 1

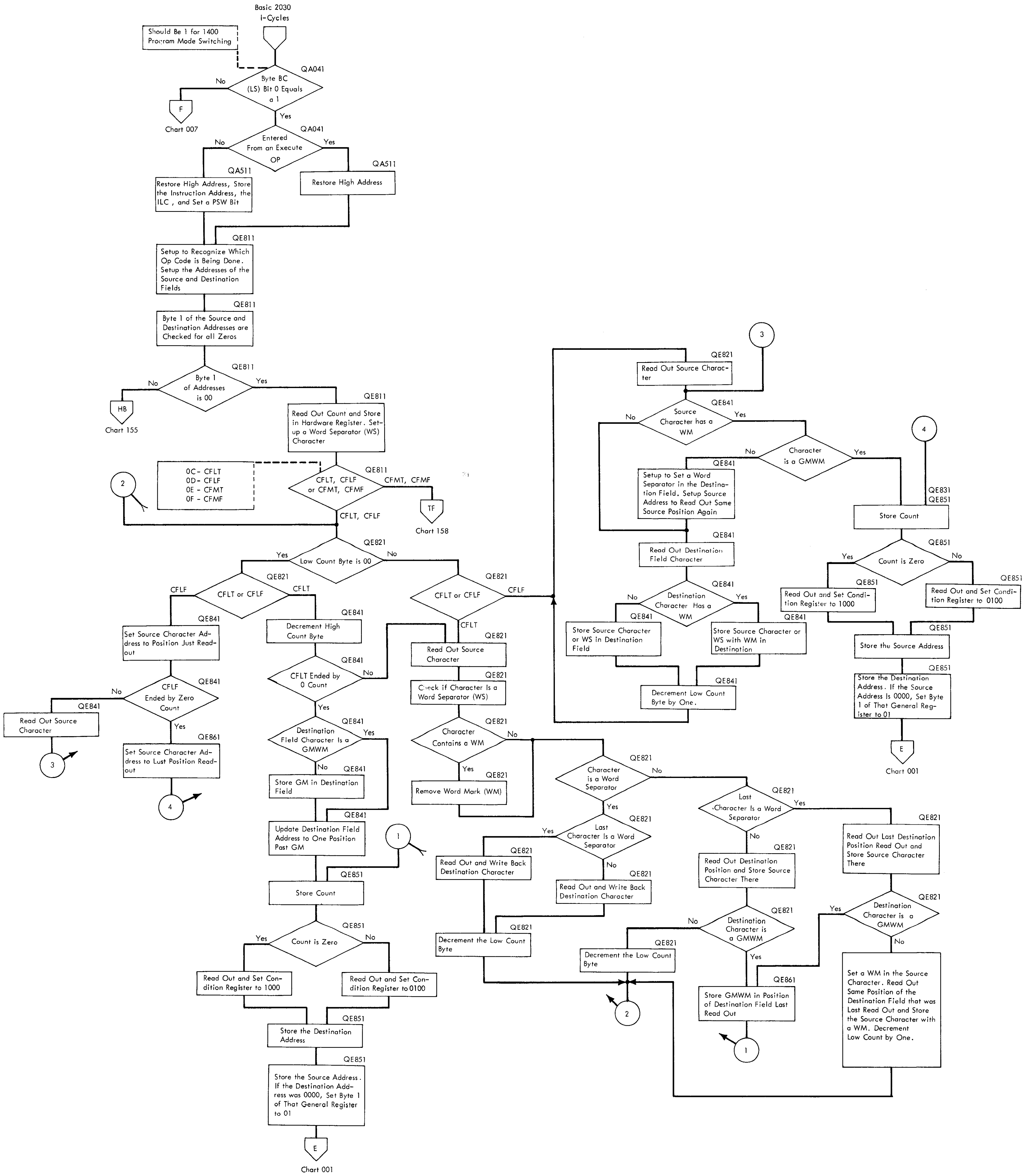
*2 If W2 Is 0 Through F, Load Four Bytes from Local Storage per Chart 1 into Main Storage

*3 If W2 Is 0 Through F, Store Sixteen Bytes from Main Storage into Local Storage 'A'. Local Storage Locations are Determined by W2; if a 0, Location 00 Through 0F are Stored etc.

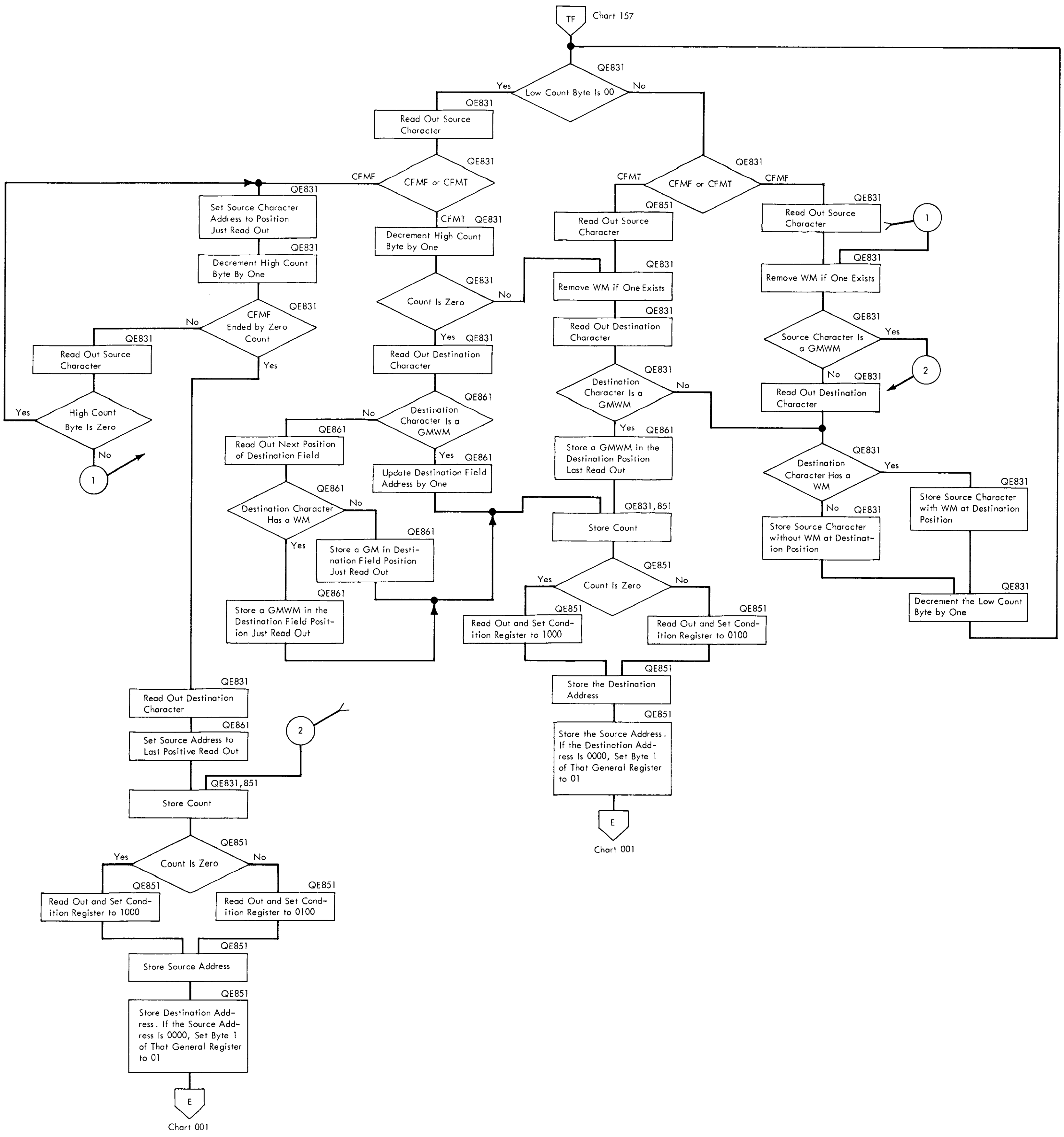
*4 Same as *3 Only Use Local Storage 'B'

*5 If W2 Is 0 Through F, Load Sixteen Bytes from Local Storage 'A' into Main Storage. Locations (LS) Are Determined the Same as *3

*6 Same as *5 Only Use Local Storage 'B'



CLF 157 CFMT, CFMF, CFLT, and CFLF Instructions, Sheet 1

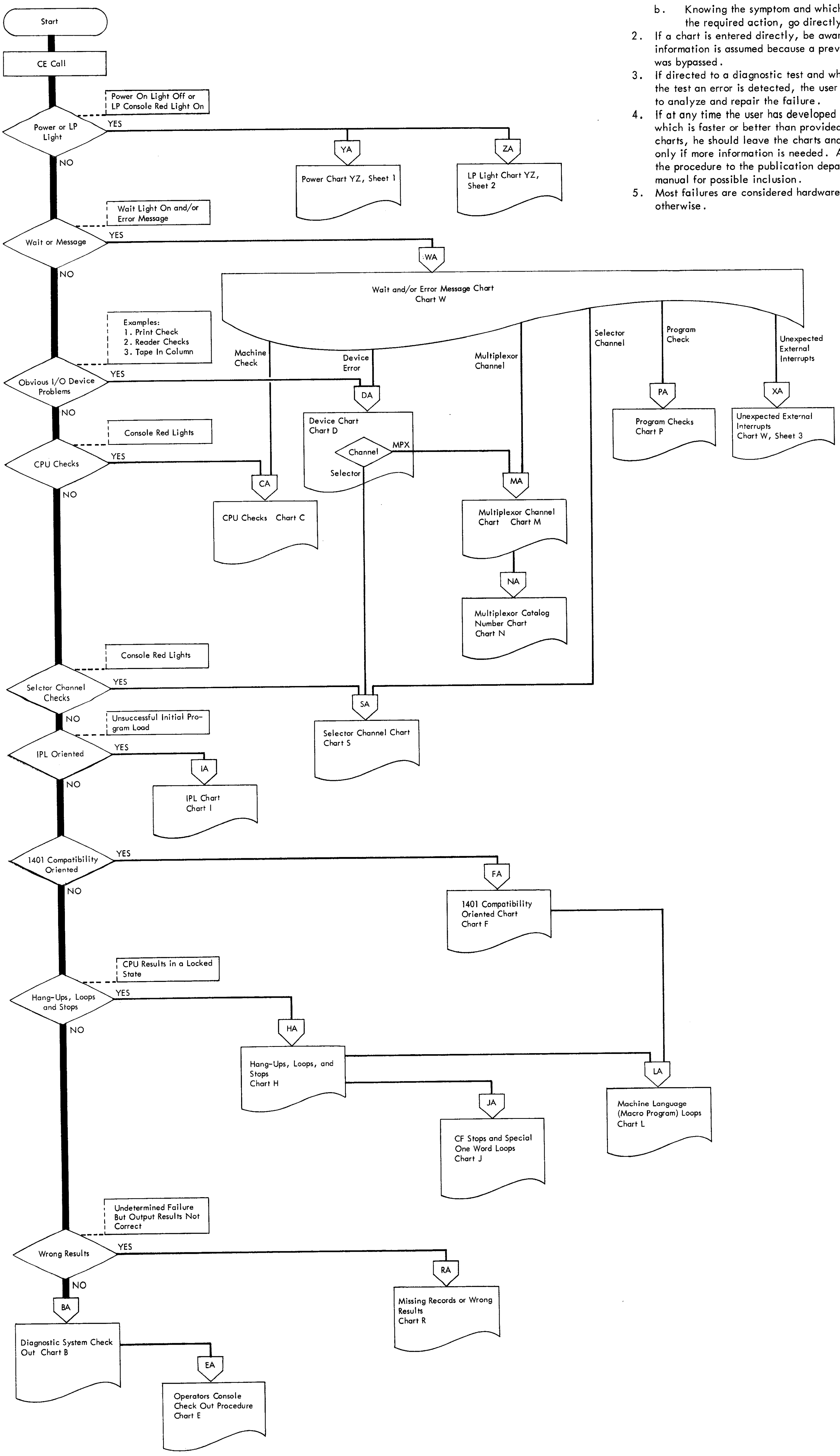


CLF 158 CFMT, CFMF, CFLT, and CFLF Instructions, Sheet 2

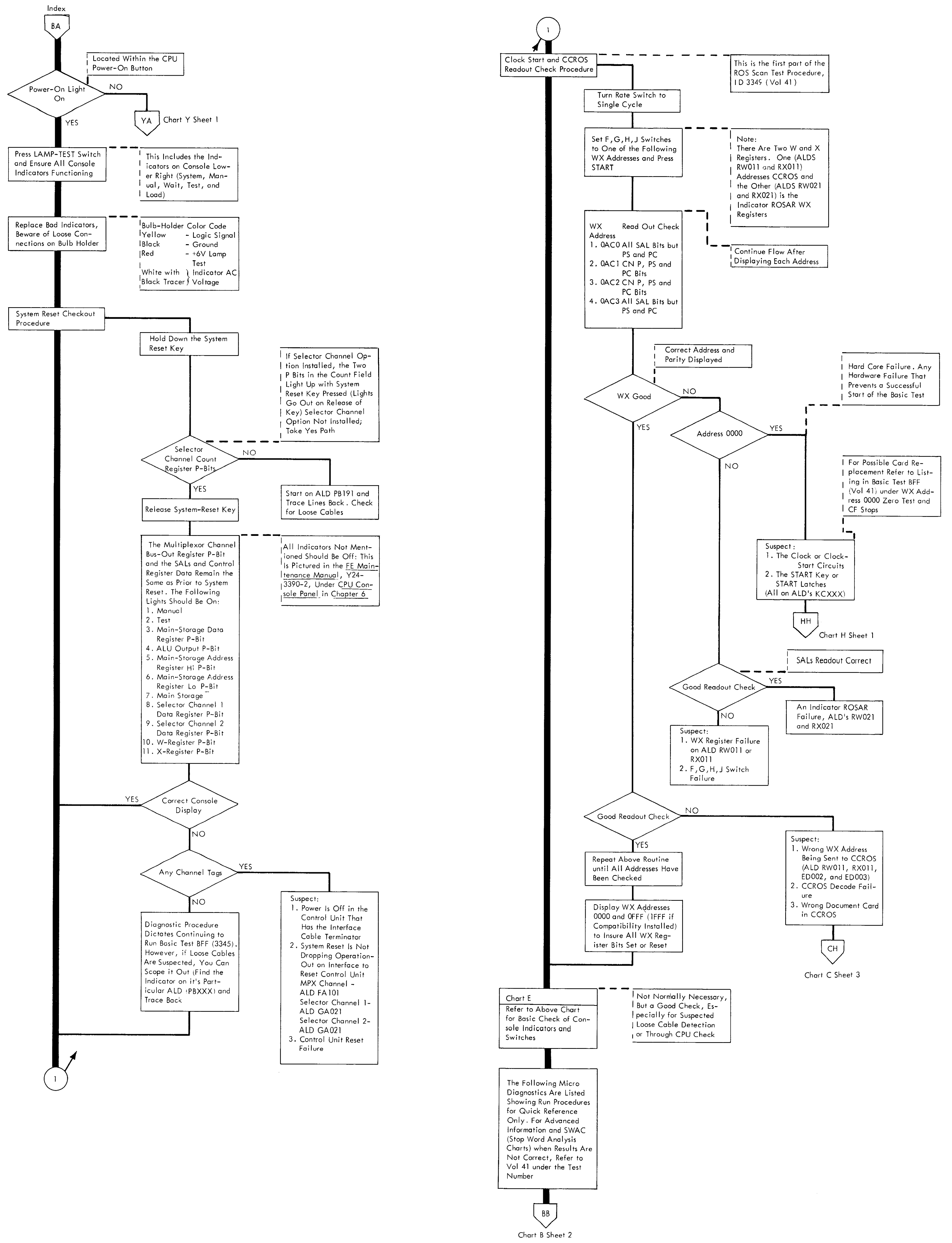
Diagnostic Techniques Charts.

How to Use.

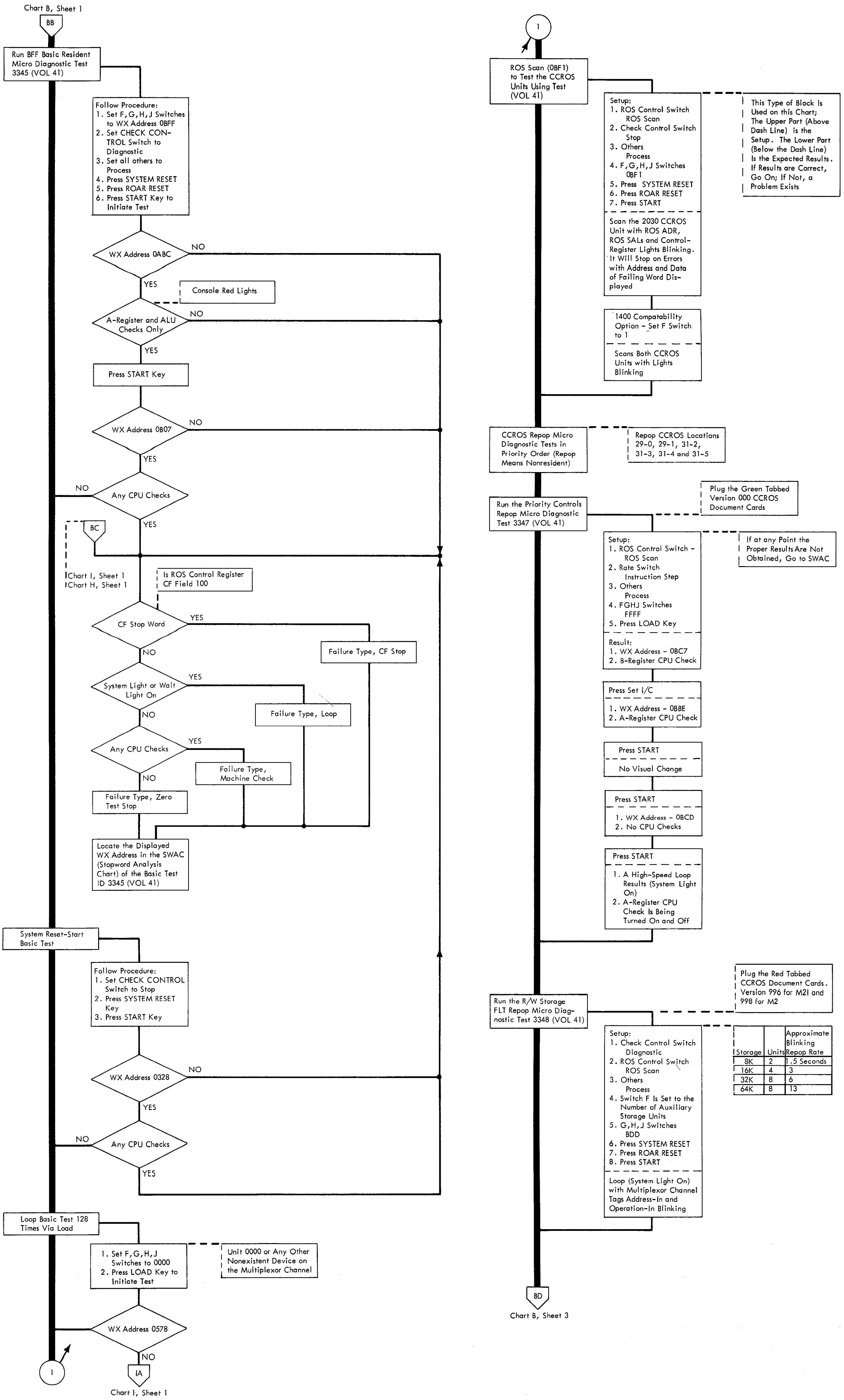
1. Initial entry can be by:
    a. Starting on this chart (chart A) find a symptom and find which chart shows the required action needed.
    b. Knowing the symptom and which chart shows the required action, go directly to that chart.
2. If a chart is entered directly, be aware that some information is assumed because a previous chart was bypassed.
3. If directed to a diagnostic test and while running the test an error is detected, the user is expected to analyze and repair the failure.
4. If at any time the user has developed an approach which is faster or better than provided in the charts, he should leave the charts and link back only if more information is needed. Also, send the procedure to the publication department of this manual for possible inclusion.
5. Most failures are considered hardware until proven otherwise.



DT Chart A Action Index

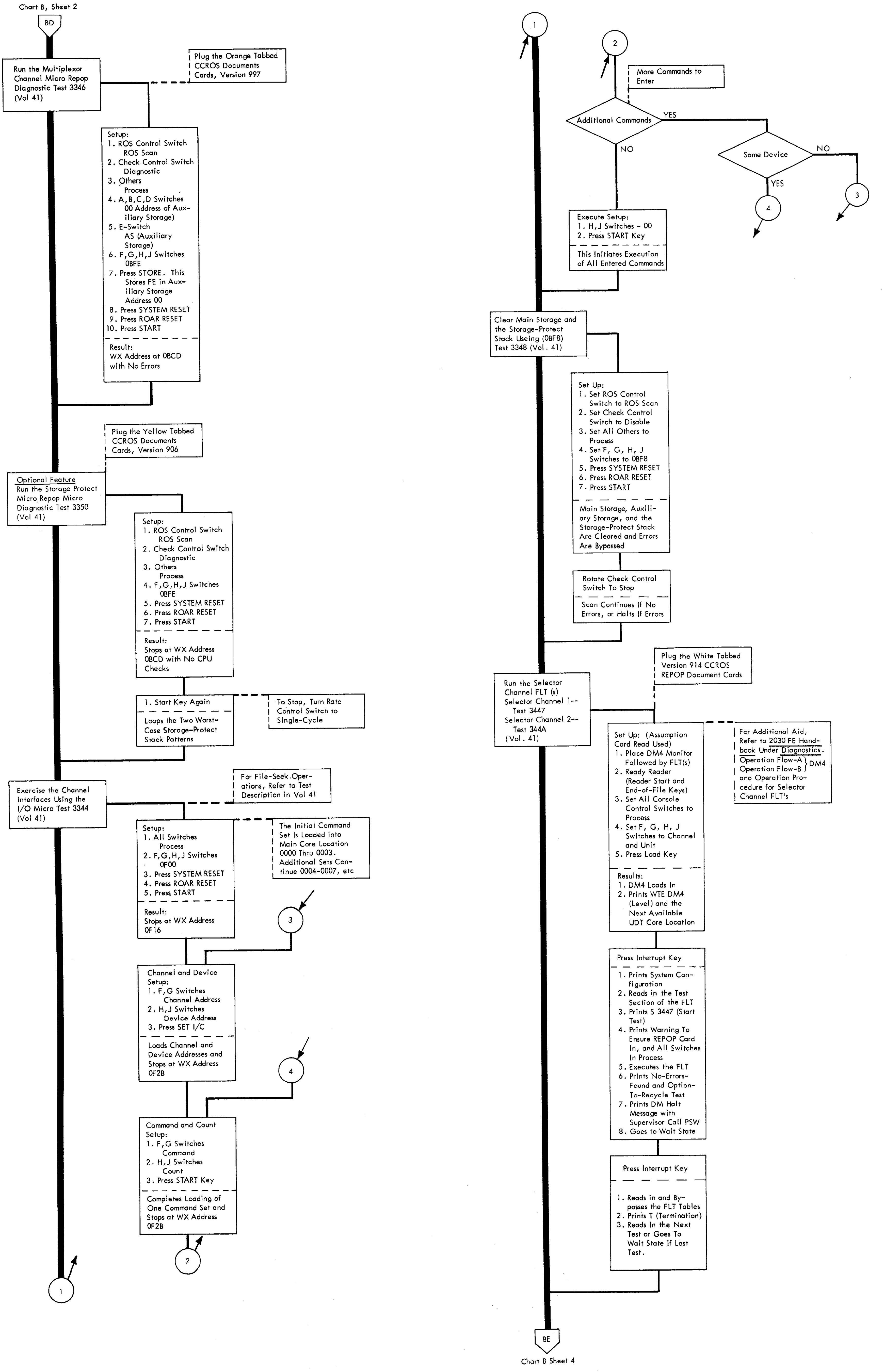


DT Chart B Diagnostic Check Out (Sheet 1



Storage	Units	Approximate Blinking Repop Rate
8K	2	1.5 Seconds
16K	4	3
32K	8	6
64K	8	13

DT Chart B Diagnostic Check Out (Sheet 2)



Run the Multiplexor Channel Micro Repop Diagnostic Test 3346 (Vol 41)

Plug the Orange Tabbed CCROS Documents Cards, Version 997

Setup:
1. ROS Control Switch
ROS Scan
2. Check Control Switch
Diagnostic
3. Others
Process
4. A,B,C,D Switches
00 Address of Auxiliary Storage)
5. E-Switch
AS (Auxiliary Storage)
6. F,G,H,J Switches
0BFE
7. Press STORE. This Stores FE in Auxiliary Storage Address 00
8. Press SYSTEM RESET
9. Press ROAR RESET
10. Press START

Result:
WX Address at 0BCD with No Errors

Plug the Yellow Tabbed CCROS Documents Cards, Version 906

Optional Feature
Run the Storage Protect Micro Repop Micro Diagnostic Test 3350 (Vol 41)

Setup:
1. ROS Control Switch
ROS Scan
2. Check Control Switch
Diagnostic
3. Others
Process
4. F,G,H,J Switches
0BFE
5. Press SYSTEM RESET
6. Press ROAR RESET
7. Press START

Result:
Stops at WX Address 0BCD with No CPU Checks

1. Start Key Again

Loops the Two Worst-Case Storage-Protect Stack Patterns

To Stop, Turn Rate Control Switch to Single-Cycle

Exercise the Channel Interfaces Using the I/O Micro Test 3344 (Vol 41)

For File-Seek Operations, Refer to Test Description in Vol 41

Setup:
1. All Switches
Process
2. F,G,H,J Switches
0F00
3. Press SYSTEM RESET
4. Press ROAR RESET
5. Press START

Result:
Stops at WX Address 0F16

The Initial Command Set Is Loaded into Main Core Location 0000 Thru 0003. Additional Sets Continue 0004-0007, etc

Channel and Device Setup:
1. F,G Switches
Channel Address
2. H,J Switches
Device Address
3. Press SET I/C

Loads Channel and Device Addresses and Stops at WX Address 0F2B

Command and Count Setup:
1. F,G Switches
Command
2. H,J Switches
Count
3. Press START Key

Completes Loading of One Command Set and Stops at WX Address 0F2B

More Commands to Enter

Additional Commands

Same Device

Execute Setup:
1. H,J Switches - 00
2. Press START Key

This Initiates Execution of All Entered Commands

Clear Main Storage and the Storage-Protect Stack Useing (0BF8) Test 3348 (Vol. 41)

Set Up:
1. Set ROS Control Switch to ROS Scan
2. Set Check Control Switch to Disable
3. Set All Others to Process
4. Set F, G, H, J Switches to 0BF8
5. Press SYSTEM RESET
6. Press ROAR RESET
7. Press START

Main Storage, Auxiliary Storage, and the Storage-Protect Stack Are Cleared and Errors Are Bypassed

Rotate Check Control Switch To Stop

Scan Continues If No Errors, or Halts If Errors

Plug the White Tabbed Version 914 CCROS REPOP Document Cards

Run the Selector Channel FLT (s)
Selector Channel 1--
Test 3447
Selector Channel 2--
Test 344A
(Vol. 41)

Set Up: (Assumption Card Read Used)
1. Place DM4 Monitor Followed by FLT(s)
2. Ready Reader (Reader Start and End-of-File Keys)
3. Set All Console Control Switches to Process
4. Set F, G, H, J Switches to Channel and Unit
5. Press Load Key

Results:
1. DM4 Loads In
2. Prints WTE DM4 (Level) and the Next Available UDT Core Location

For Additional Aid, Refer to 2030 FE Handbook Under Diagnostics.
Operation Flow-A } DM4
Operation Flow-B }
and Operation Procedure for Selector Channel FLT's

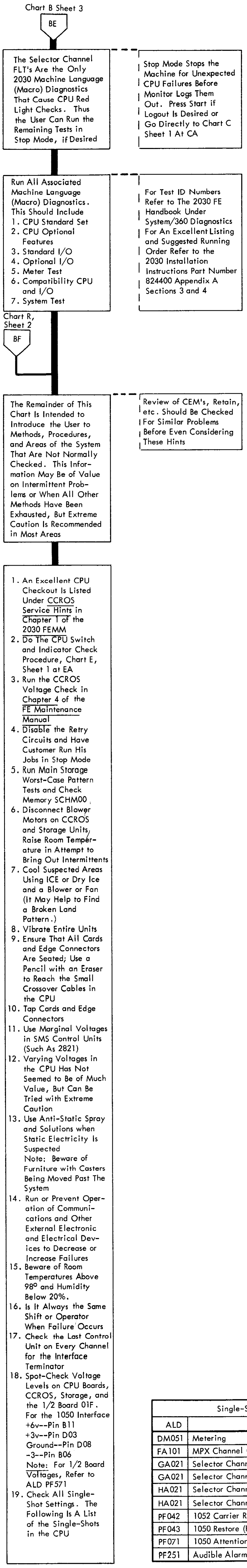
Press Interrupt Key

1. Prints System Configuration
2. Reads in the Test Section of the FLT
3. Prints S 3447 (Start Test)
4. Prints Warning To Ensure REPOP Card In, and All Switches In Process
5. Executes the FLT
6. Prints No-Errors-Found and Option-To-Recycle Test
7. Prints DM Halt Message with Supervisor Call PSW
8. Goes to Wait State

Press Interrupt Key

1. Reads in and By-passes the FLT Tables
2. Prints T (Termination)
3. Reads In the Next Test or Goes To Wait State If Last Test.

DT Chart B Diagnostic Check Out (Sheet 3)



The Selector Channel FLT's Are the Only 2030 Machine Language (Macro) Diagnostics That Cause CPU Red Light Checks. Thus the User Can Run the Remaining Tests in Stop Mode, if Desired

Stop Mode Stops the Machine for Unexpected CPU Failures Before Monitor Logs Them Out. Press Start if Logout Is Desired or Go Directly to Chart C Sheet 1 At CA

Run All Associated Machine Language (Macro) Diagnostics. This Should Include

1. CPU Standard Set
2. CPU Optional Features
3. Standard I/O
4. Optional I/O
5. Meter Test
6. Compatibility CPU and I/O
7. System Test

For Test ID Numbers Refer to The 2030 FE Handbook Under System/360 Diagnostics For An Excellent Listing and Suggested Running Order Refer to the 2030 Installation Instructions Part Number 824400 Appendix A Sections 3 and 4

The Remainder of This Chart Is Intended to Introduce the User to Methods, Procedures, and Areas of the System That Are Not Normally Checked. This Information May Be of Value on Intermittent Problems or When All Other Methods Have Been Exhausted, But Extreme Caution Is Recommended in Most Areas

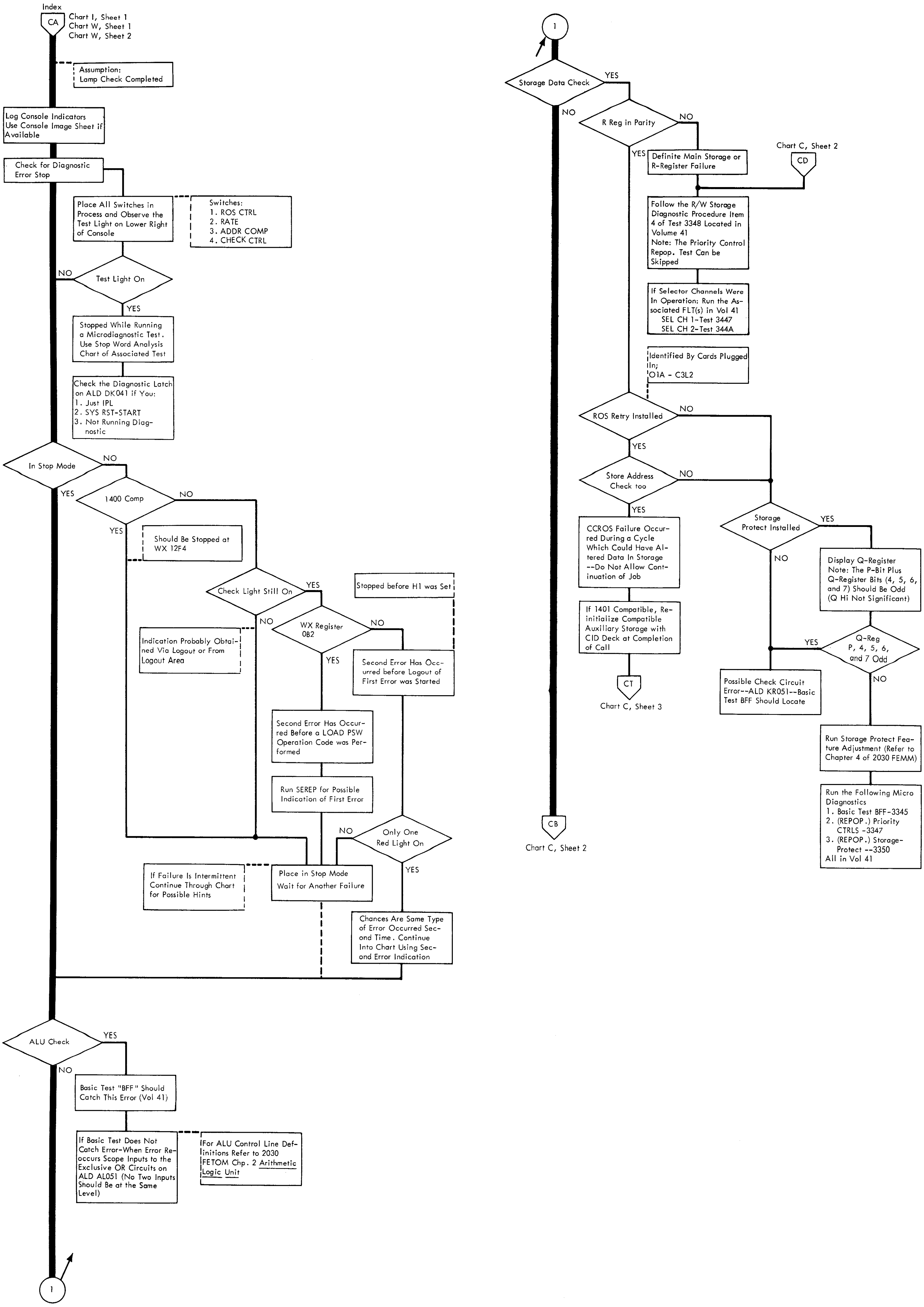
Review of CEM's, Retain, etc. Should Be Checked For Similar Problems Before Even Considering These Hints

1. An Excellent CPU Checkout Is Listed Under CCROS Service Hints in Chapter 1 of the 2030 FEMM
2. Do The CPU Switch and Indicator Check Procedure, Chart E, Sheet 1 at EA
3. Run the CCROS Voltage Check in Chapter 4 of the FE Maintenance Manual
4. Disable the Retry Circuits and Have Customer Run His Jobs in Stop Mode
5. Run Main Storage Worst-Case Pattern Tests and Check Memory SCHM00
6. Disconnect Blower Motors on CCROS and Storage Units, Raise Room Temperature in Attempt to Bring Out Intermittents
7. Cool Suspected Areas Using ICE or Dry Ice and a Blower or Fan (It May Help to Find a Broken Land Pattern.)
8. Vibrate Entire Units
9. Ensure That All Cards and Edge Connectors Are Seated; Use a Pencil with an Eraser to Reach the Small Crossover Cables in the CPU
10. Tap Cards and Edge Connectors
11. Use Marginal Voltages in SMS Control Units (Such As 2821)
12. Varying Voltages in the CPU Has Not Seemed to Be of Much Value, But Can Be Tried with Extreme Caution
13. Use Anti-Static Spray and Solutions when Static Electricity Is Suspected
    Note: Beware of Furniture with Casters Being Moved Past The System
14. Run or Prevent Operation of Communications and Other External Electronic and Electrical Devices to Decrease or Increase Failures
15. Beware of Room Temperatures Above 98° and Humidity Below 20%.
16. Is It Always the Same Shift or Operator When Failure Occurs
17. Check the Last Control Unit on Every Channel for the Interface Terminator
18. Spot-Check Voltage Levels on CPU Boards, CCROS, Storage, and the 1/2 Board 01F. For the 1050 Interface
    +6v--Pin B11
    +3v--Pin D03
    Ground--Pin D08
    -3--Pin B06
    Note: For 1/2 Board Voltages, Refer to ALD PF571
19. Check All Single-Shot Settings. The Following Is A List of the Single-Shots in the CPU

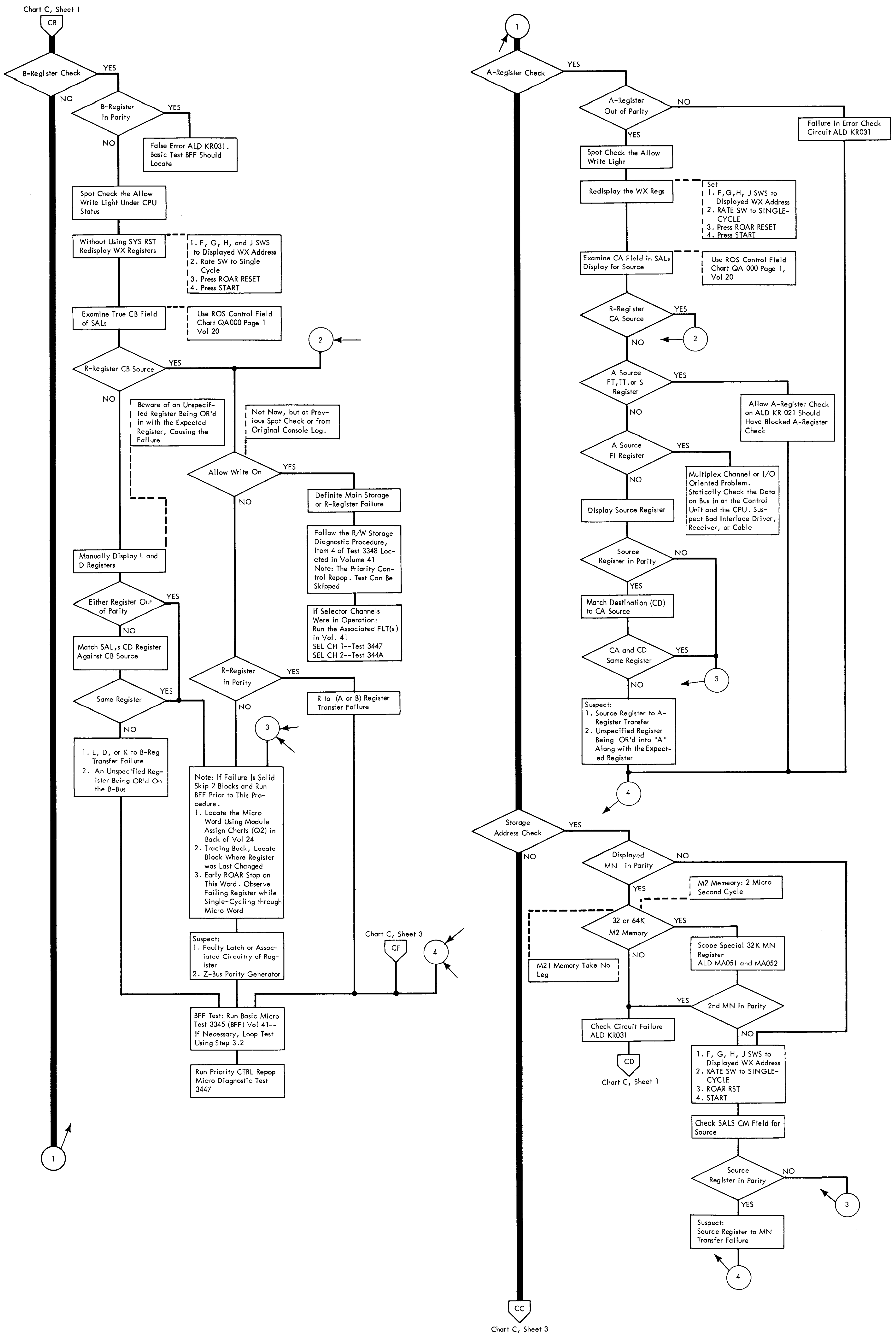
Single-Shots in CPU

ALD	Function
DM051	Metering
FA101	MPX Channel Operation-Out Reset
GA021	Selector Channel 1 Operation-Out Reset
GA021	Selector Channel 1 Hold-Out Interlock
HA021	Selector Channel 2 Operation-Out Reset
HA021	Selector Channel 2 Hold-Out Interlock
PF042	1052 Carrier Return Line Feed
PF043	1050 Restore (Reset)
PF071	1050 Attention (Request)
PF251	Audible Alarm (Optional)

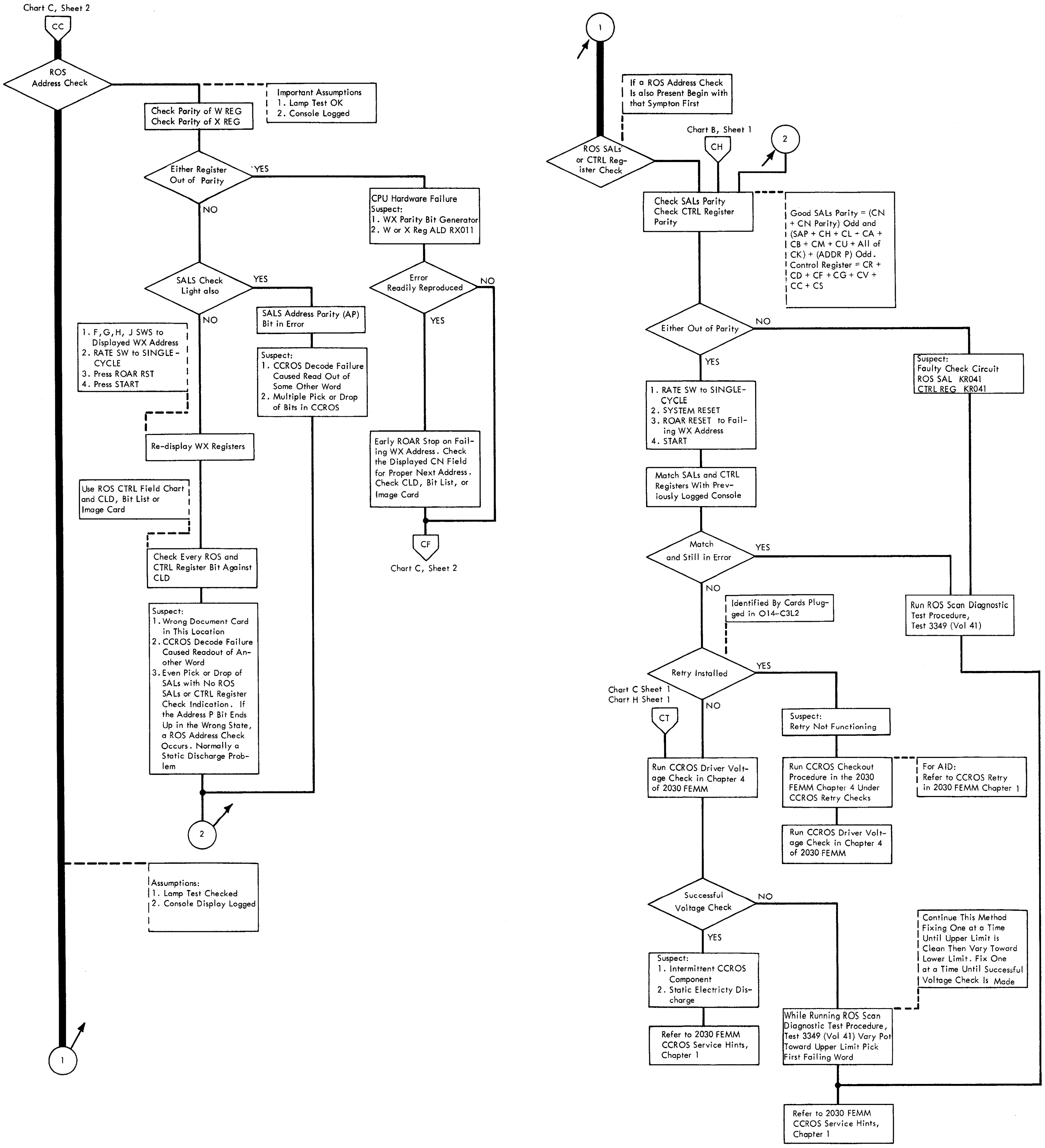
DT Chart B Diagnostic Check Out (Sheet 4)



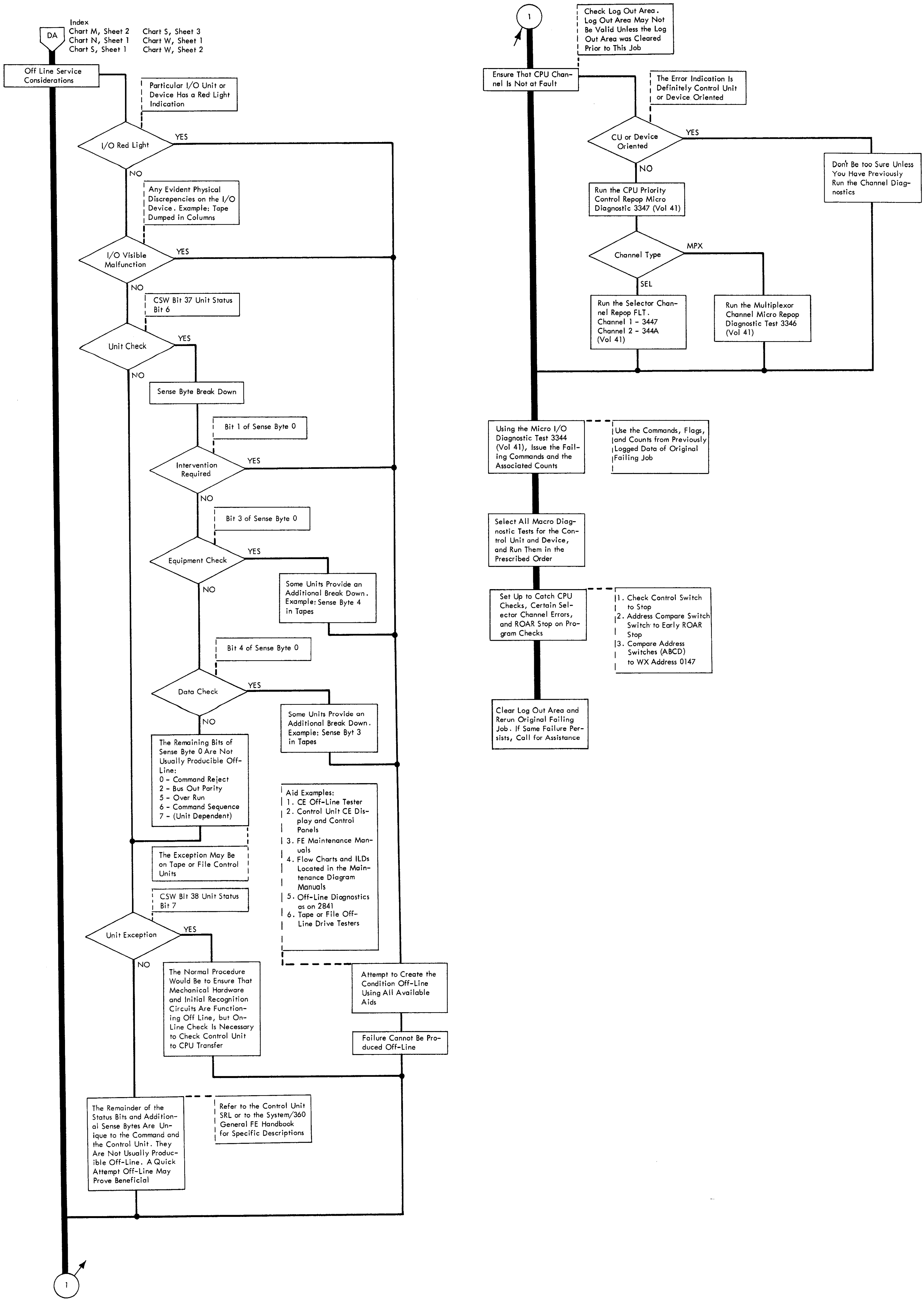
DT Chart C CPU Checks (Sheet 1)



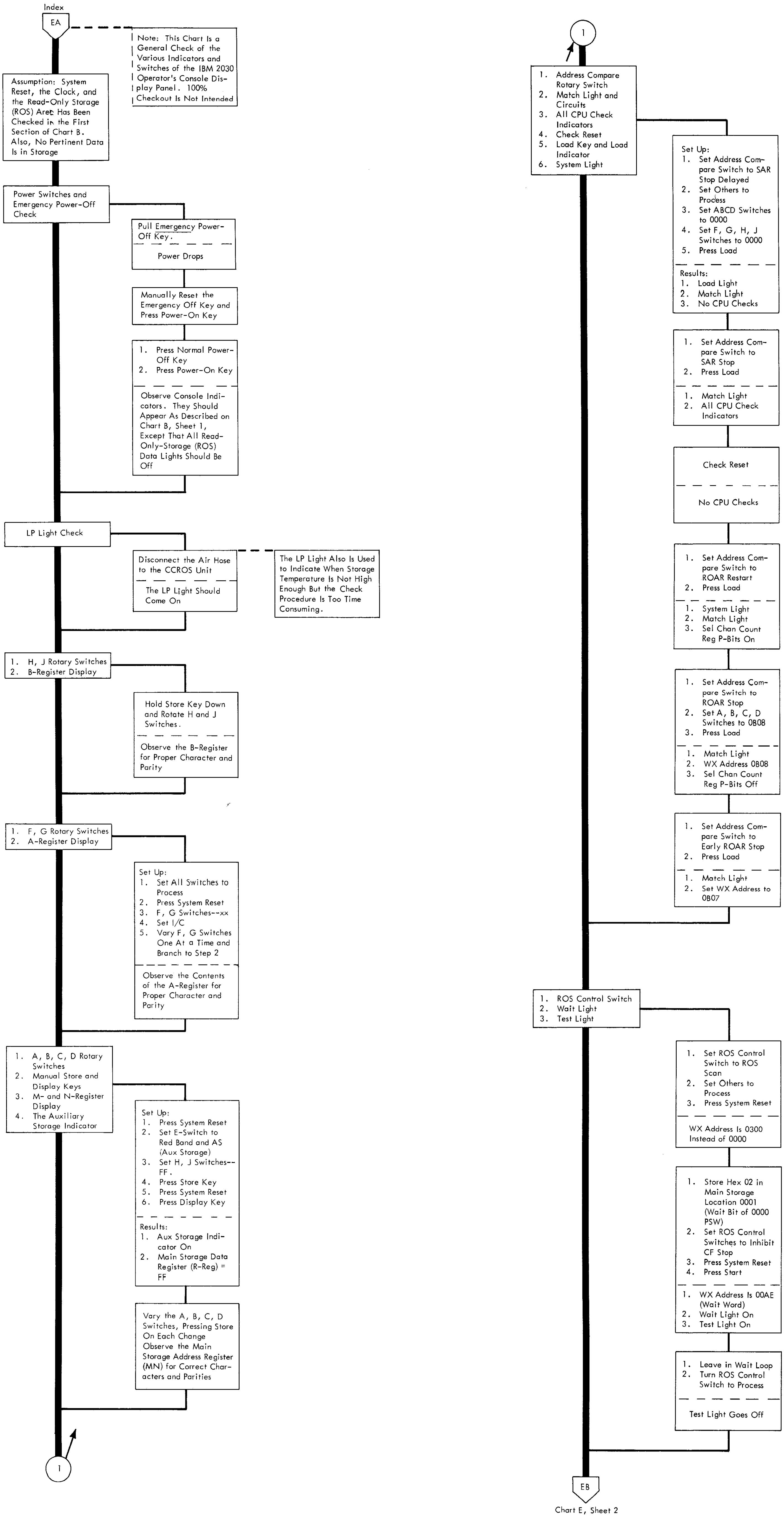
DT Chart C CPU Checks (Sheet 2)



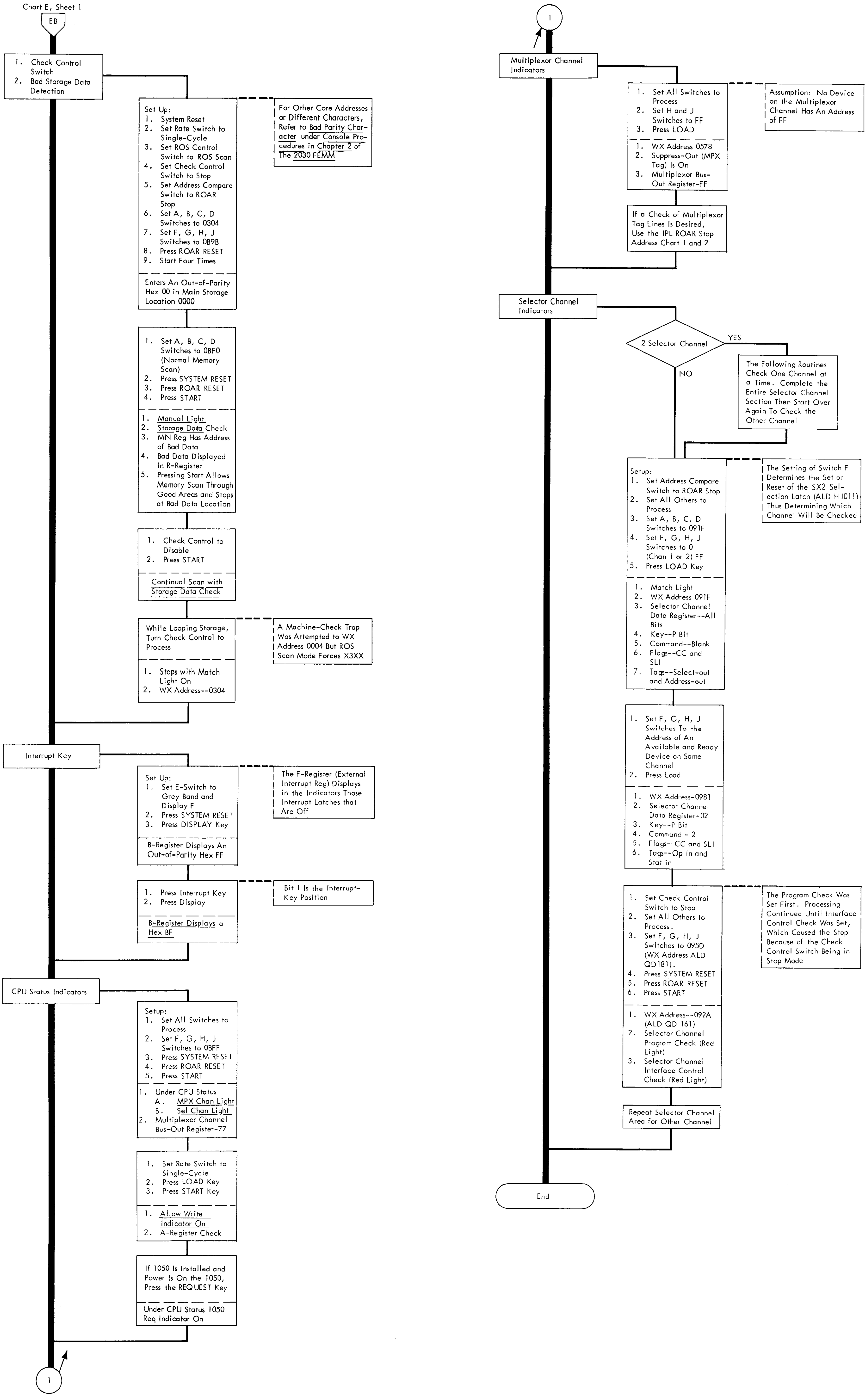
DT Chart C CPU Checks (Sheet 3)



DT Chart D Device Chart



DT Chart E Operators Console Check Out (Sheet 1)



1. Check Control Switch
2. Bad Storage Data Detection

Set Up:
1. System Reset
2. Set Rate Switch to Single-Cycle
3. Set ROS Control Switch to ROS Scan
4. Set Check Control Switch to Stop
5. Set Address Compare Switch to ROAR Stop
6. Set A, B, C, D Switches to 0304
7. Set F, G, H, J Switches to 0B9B
8. Press ROAR RESET
9. Start Four Times

Enters An Out-of-Parity Hex 00 in Main Storage Location 0000

For Other Core Addresses or Different Characters, Refer to Bad Parity Character under Console Procedures in Chapter 2 of The 2030 FEMM

1. Set A, B, C, D Switches to 0BF0 (Normal Memory Scan)
2. Press SYSTEM RESET
3. Press ROAR RESET
4. Press START

1. Manual Light
2. Storage Data Check
3. MN Reg Has Address of Bad Data
4. Bad Data Displayed in R-Register
5. Pressing Start Allows Memory Scan Through Good Areas and Stops at Bad Data Location

1. Check Control to Disable
2. Press START

Continual Scan with Storage Data Check

While Looping Storage, Turn Check Control to Process

1. Stops with Match Light On
2. WX Address--0304

A Machine-Check Trap Was Attempted to WX Address 0004 But ROS Scan Mode Forces X3XX

Interrupt Key

Set Up:
1. Set E-Switch to Grey Band and Display F
2. Press SYSTEM RESET
3. Press DISPLAY Key

B-Register Displays An Out-of-Parity Hex FF

The F-Register (External Interrupt Reg) Displays in the Indicators Those Interrupt Latches that Are Off

1. Press Interrupt Key
2. Press Display

B-Register Displays a Hex BF

Bit 1 Is the Interrupt-Key Position

CPU Status Indicators

Setup:
1. Set All Switches to Process
2. Set F, G, H, J Switches to 0BFF
3. Press SYSTEM RESET
4. Press ROAR RESET
5. Press START

1. Under CPU Status
   A. MPX Chan Light
   B. Sel Chan Light
2. Multiplexor Channel Bus-Out Register-77

1. Set Rate Switch to Single-Cycle
2. Press LOAD Key
3. Press START Key

1. Allow Write Indicator On
2. A-Register Check

If 1050 Is Installed and Power Is On the 1050, Press the REQUEST Key

Under CPU Status 1050 Req Indicator On

Multiplexor Channel Indicators

1. Set All Switches to Process
2. Set H and J Switches to FF
3. Press LOAD

1. WX Address 0578
2. Suppress-Out (MPX Tag) Is On
3. Multiplexor Bus-Out Register-FF

Assumption: No Device on the Multiplexor Channel Has An Address of FF

If a Check of Multiplexor Tag Lines Is Desired, Use the IPL ROAR Stop Address Chart 1 and 2

Selector Channel Indicators

The Following Routines Check One Channel at a Time. Complete the Entire Selector Channel Section Then Start Over Again To Check the Other Channel

Setup:
1. Set Address Compare Switch to ROAR Stop
2. Set All Others to Process
3. Set A, B, C, D Switches to 091F
4. Set F, G, H, J Switches to 0 (Chan 1 or 2) FF
5. Press LOAD Key

1. Match Light
2. WX Address 091F
3. Selector Channel Data Register--All Bits
4. Key--P Bit
5. Command--Blank
6. Flags--CC and SLI
7. Tags--Select-out and Address-out

The Setting of Switch F Determines the Set or Reset of the SX2 Selection Latch (ALD HJ011) Thus Determining Which Channel Will Be Checked

1. Set F, G, H, J Switches To the Address of An Available and Ready Device on Same Channel
2. Press Load

1. WX Address-0981
2. Selector Channel Data Register-02
3. Key--P Bit
4. Command - 2
5. Flags--CC and SLI
6. Tags--Op in and Stat in

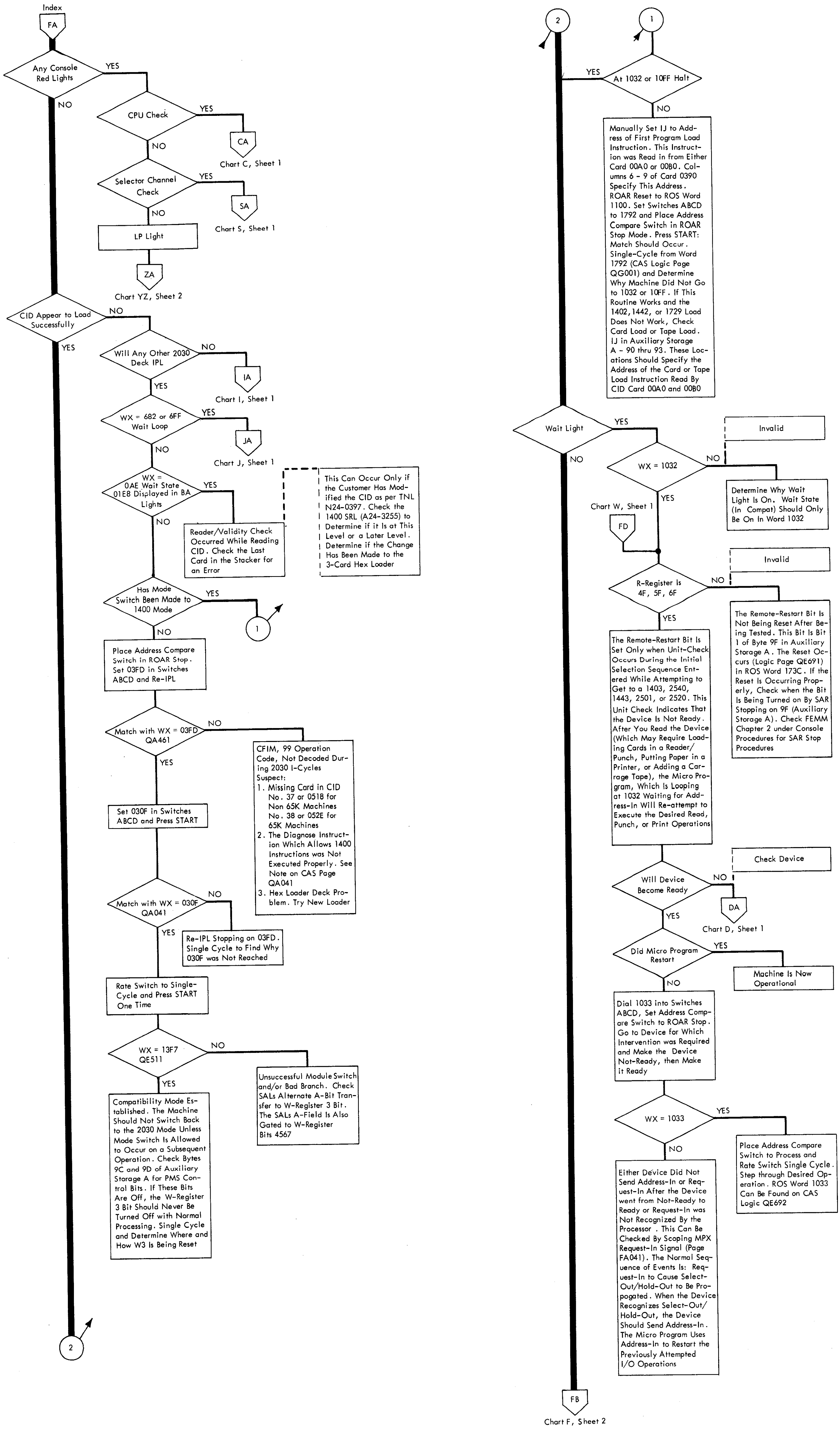
1. Set Check Control Switch to Stop
2. Set All Others to Process.
3. Set F, G, H, J Switches to 095D (WX Address ALD QD181).
4. Press SYSTEM RESET
5. Press ROAR RESET
6. Press START

1. WX Address--092A (ALD QD 161)
2. Selector Channel Program Check (Red Light)
3. Selector Channel Interface Control Check (Red Light)

The Program Check Was Set First. Processing Continued Until Interface Control Check Was Set, Which Caused the Stop Because of the Check Control Switch Being in Stop Mode

Repeat Selector Channel Area for Other Channel

DT Chart E Operators Console Check Out (Sheet 2)



Reader/Validity Check Occurred While Reading CID. Check the Last Card in the Stacker for an Error

This Can Occur Only if the Customer Has Modified the CID as per TNL N24-0397. Check the 1400 SRL (A24-3255) to Determine if it Is at This Level or a Later Level. Determine if the Change Has Been Made to the 3-Card Hex Loader

Place Address Compare Switch in ROAR Stop. Set 03FD in Switches ABCD and Re-IPL

CFIM, 99 Operation Code, Not Decoded During 2030 I-Cycles
Suspect:
1. Missing Card in CID No. 37 or 0518 for Non 65K Machines No. 38 or 052E for 65K Machines
2. The Diagnose Instruction Which Allows 1400 Instructions was Not Executed Properly. See Note on CAS Page QA041
3. Hex Loader Deck Problem. Try New Loader

Set 030F in Switches ABCD and Press START

Re-IPL Stopping on 03FD. Single Cycle to Find Why 030F was Not Reached

Rate Switch to Single-Cycle and Press START One Time

Unsuccessful Module Switch and/or Bad Branch. Check SALs Alternate A-Bit Transfer to W-Register 3 Bit. The SALs A-Field Is Also Gated to W-Register Bits 4567

Compatibility Mode Established. The Machine Should Not Switch Back to the 2030 Mode Unless Mode Switch Is Allowed to Occur on a Subsequent Operation. Check Bytes 9C and 9D of Auxiliary Storage A for PMS Control Bits. If These Bits Are Off, the W-Register 3 Bit Should Never Be Turned Off with Normal Processing. Single Cycle and Determine Where and How W3 Is Being Reset

Manually Set IJ to Address of First Program Load Instruction. This Instruction was Read in from Either Card 00A0 or 00B0. Columns 6 - 9 of Card 0390 Specify This Address. ROAR Reset to ROS Word 1100. Set Switches ABCD to 1792 and Place Address Compare Switch in ROAR Stop Mode. Press START: Match Should Occur. Single-Cycle from Word 1792 (CAS Logic Page QG001) and Determine Why Machine Did Not Go to 1032 or 10FF. If This Routine Works and the 1402,1442, or 1729 Load Does Not Work, Check Card Load or Tape Load. IJ in Auxiliary Storage A - 90 thru 93. These Locations Should Specify the Address of the Card or Tape Load Instruction Read By CID Card 00A0 and 00B0

Determine Why Wait Light Is On. Wait State (In Compat) Should Only Be On In Word 1032

The Remote-Restart Bit Is Set Only when Unit-Check Occurs During the Initial Selection Sequence Entered While Attempting to Get to a 1403, 2540, 1443, 2501, or 2520. This Unit Check Indicates That the Device Is Not Ready. After You Read the Device (Which May Require Loading Cards in a Reader/Punch, Putting Paper in a Printer, or Adding a Carrage Tape), the Micro Program, Which Is Looping at 1032 Waiting for Address-In Will Re-attempt to Execute the Desired Read, Punch, or Print Operations

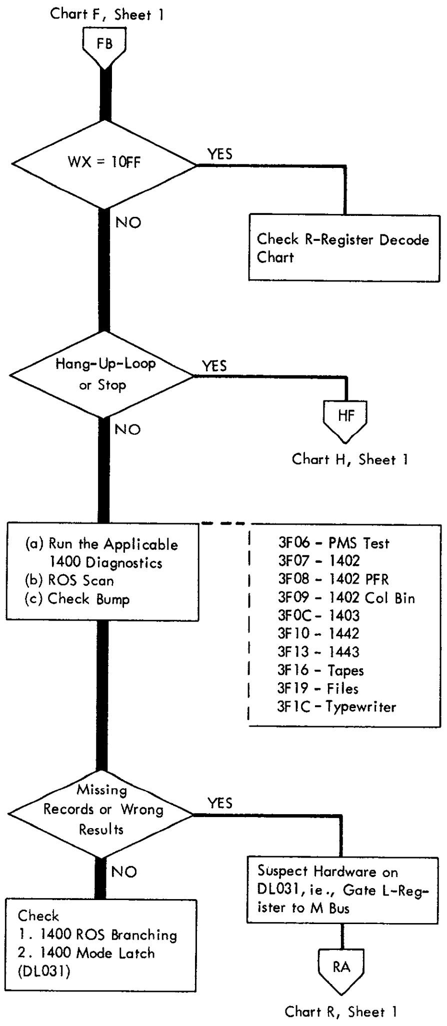
The Remote-Restart Bit Is Not Being Reset After Being Tested. This Bit Is Bit 1 of Byte 9F in Auxiliary Storage A. The Reset Occurs (Logic Page QE691) in ROS Word 173C. If the Reset Is Occurring Properly, Check when the Bit Is Being Turned on By SAR Stopping on 9F (Auxiliary Storage A). Check FEMM Chapter 2 under Console Procedures for SAR Stop Procedures

Dial 1033 into Switches ABCD, Set Address Compare Switch to ROAR Stop. Go to Device for Which Intervention was Required and Make the Device Not-Ready, then Make it Ready

Place Address Compare Switch to Process and Rate Switch Single Cycle. Step through Desired Operation. ROS Word 1033 Can Be Found on CAS Logic QE692

Either Device Did Not Send Address-In or Request-In After the Device went from Not-Ready to Ready or Request-In was Not Recognized By the Processor . This Can Be Checked By Scoping MPX Request-In Signal (Page FA041). The Normal Sequence of Events Is: Request-In to Cause Select-Out/Hold-Out to Be Propogated. When the Device Recognizes Select-Out/Hold-Out, the Device Should Send Address-In. The Micro Program Uses Address-In to Restart the Previously Attempted I/O Operations

DT Chart F 1400 Compatibility Oriented (Sheet 1)



DT Chart F 1400 Compatibility Oriented (Sheet 2)

On all stops at ROS address 10FF, a coded digit is displayed in the Main Storage Data Register. In the case of a set IC operation, the contents of the main storage location to which the instruction counter has been set are displayed. In the case of a sense switch operation, the resulting sense switch byte (98 of Aux. Stor. A) is displayed. In all other cases, the displayed digit has the significance shown in the R-Register Decode chart.

In addition to the coded digits which are detailed below, the following 1400 addresses are available on a 10FF halt.

(1) The decimal instruction address is displayed in the BA-registers.

(2) The decimal A-storage-address register is displayed by the MSDR lights.

(3) The decimal B-storage-address register is contained in the U-and V- registers and must be displayed manually.

The hexadecimal equivalents of these storage-address registers are also available and can be obtained by displaying the following Aux. Stor. A locations:

(1) The hexadecimal 1400 instruction address in locations 88 and 89.

(2) The hexadecimal 1400 A-storage-address register in locations 8A and 8B.

(3) The hexadecimal 1400 B-storage-address register in locations 8C and 8D.

The S-register is stored in byte 8F of Aux. Stor. A. If either S6 or S7 or both S6 and S7 equal 1, the respective 1400 storage-address register, A or B or both A and B, is displayed in hexadecimal rather than decimal form. A 1 for S6 or S7 denotes an invalid address for addressing storage. No hexadecimal-decimal conversion is attempted on an invalid address. For example: M%U1500R; - the contents of the A-storage-address register (%U1) is considered an invalid A-address.

FC Chart H, Sheet 1

R-REGISTER DECODE CHART

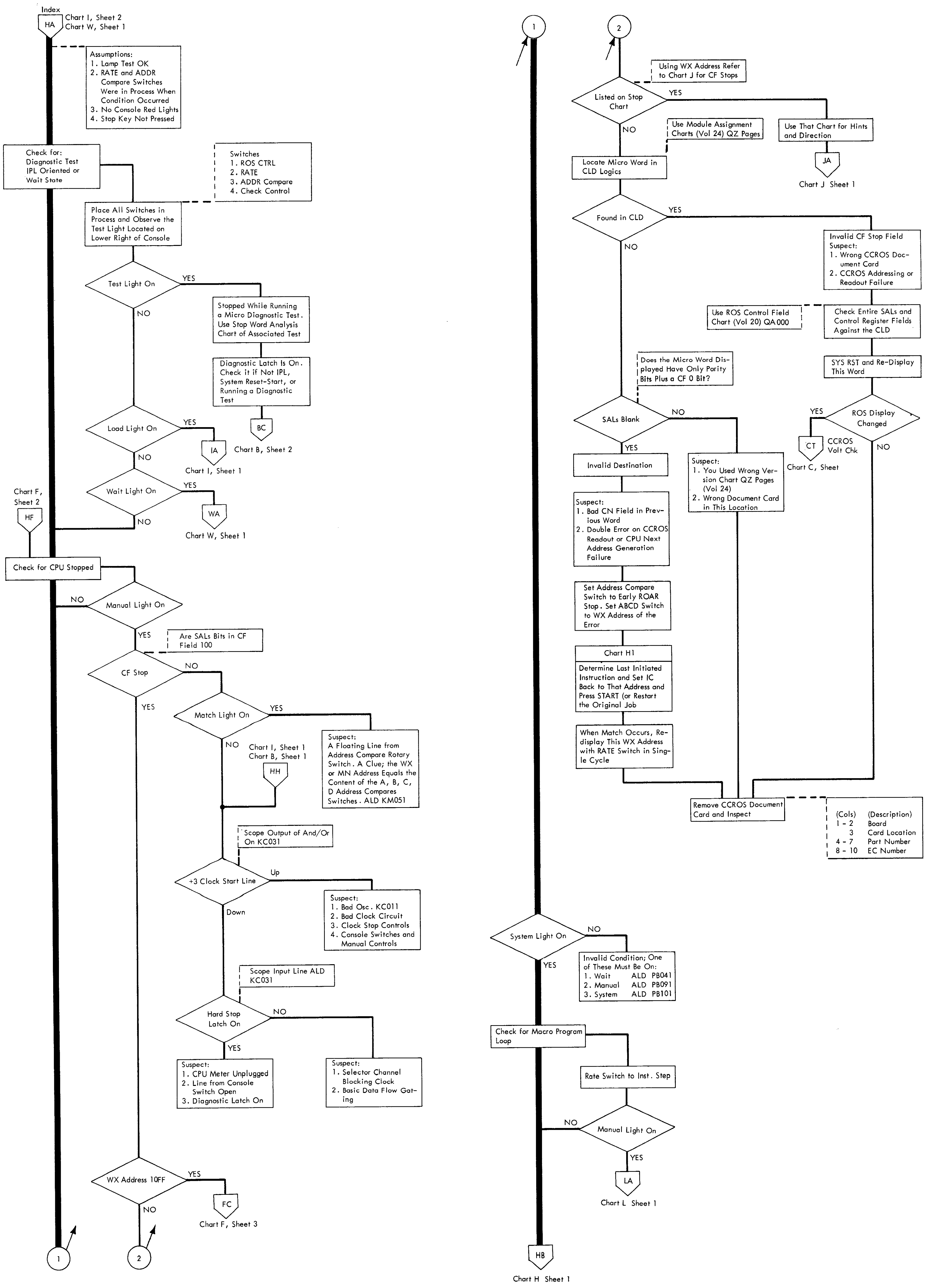
Digit Displayed In MSDR	Reason For Halt	Comments
00	Normal stop. Appears when the stop is caused by pressing the stop key, ending an instruction executed in instruction-step mode, or a match occurs in SAR Delayed-Stop mode.	
01	The program attempts to use an invalid 1400 B-address.	The address displayed in the BA-register lights is the address of the next instruction character which has a word mark. Working back from this next instruction, assemble the failing instruction and attempt to determine why the B-address was invalid. Use a program listing if one is available. Was the instruction read into storage improperly?
02	The program attempted to use an invalid 1400 A-address.	Same comments apply as for character 01 halt.
04	The program attempted to use an invalid 1400 operation code.	The decimal address displayed in the BA-register lights is the address of the next character with a word mark. Working back from this address, determine the failing op-code. From a program listing, attempt to determine what the op code should be. Also investigate for a misplaced word mark.
05	An invalid I/O operation was attempted; either the unit selection or the unit number was invalid.	Check the A-field of the previous op-code. The address of the previous op-code should be 7 less than the address displayed in the BA-lights. This halt identifies a problem with the hundreds and/or units position of the A-field. In the case of M%U1500R, the difficulty would be with the U or the 1.
06	A storage-wrap condition occurred when an address outside of the system capacity was used to address core storage.	Check the op-code to ensure that valid addresses are being used. If these addresses are valid, check the storage size as defined in the CID. Cards 0320, 0390, 0420, 0500, and the last card of the IBM Standard CID contain references to storage size. If these addresses are invalid, attempt to determine what the address should be from a program listing.
07	Storage protection occurred in 1400 mode.	Check the Q-register. Bits 0 through 3 should be either 0000 or equal to bits 4 thru 7. If these bits do not compare as indicated above, check the protection key in the last loaded PSW. This key must be equal to Q-register bits 4 thru 7. If these two differ, call for programming assistance. Check that the conditions specified by the last loaded PSW exist while the machine is operating in 1400 mode. If the key in the PSW is 0000 or equal to bits 4 thru 7 of the Q-register and this halt occurs, a hardware malfunction has probably occurred. See Chart P-PE (Storage Protection).
08	An attempt was made to switch to 2030 mode without the PMS feature. May also be caused by erroneous interrupts.	Check the PMS control bytes for proper definition. Bytes 9C and 9D of Aux. Storage A contain bits that define the program-mode-switch subfeature. If the feature is not installed on the machine, these definition bits must be zero. The bytes are 9C 9D 0. Mode Sw On Inv I/O Ops 1. Mode Sw On Console Ops 2. Mode Sw On Printer Ops 3. Mode Sw On Inv Ops Mode Sw On Rdr/Pch Ops 4. 5. Mode Sw On Halts Mode Sw On Tape Ops 6. Mode Sw On File Ops 7. Mode Sw On Error Stops Also check if an erroneous interrupt is forcing a branch to word 1102 (CAS logic page QE601). ROAR stopping on this word and displaying the F-register indicates what type of interrupt has occurred. Note: All bits of the F-reg are normally on. Bit 0 being off indicates a timer interrupt has occurred. Bit 1 off indicates a console interrupt. Bits 2-7 indicate that an external interrupt has occurred.
09	An invalid source or destination address was specified on one of the following special 2030 PMS instructions: CFMT - 0E R1,R2 - COMPAT FEAT MOVE TO CFMF - 0F R1,R2 - COMPAT FEAT MOVE FROM CFLT - 0C R1,R2 - COMPAT FEAT LOAD TO CFLF - 0D R1,R2 - COMPAT FEAT LOAD FROM	The third byte of the source and or destination address is non-zero. The next Basic 2030 instruction should be stored in Basic 2030 LS A9 and AA. Working back from that address, check the specified registers to determine why the addresses are invalid.
0A	The microprogram attempted to convert to BCD an address that was less than the bias address. This occurred during execution of a clear-storage or store-address-register operation.	This situation usually appears when the execution of an operation is attempted where the A- and/or B-fields are obtained from the backup locations; for example, M H aaa. When execution of the Move operation is attempted, the A- and B-addresses are fetched from LT and UV backup locations. These addresses are in hexadecimal format. When the clear-or store-star operation is attempted, these addresses are converted to BCD. If the address is less than the bias, the 0A-halt occurs. When the PMS feature is installed, the programmer can alter these locations. Therefore, checking the 2030 op-code which can access these locations may point to the error. These op-codes are of the SI format: 99 W1¦W2 B1¦D1
0B	Storage wrap low; the 2030 storage location 1 less than 1400 000 was addressed.	The BA-register may be invalid at this halt, ie., it may not contain the decimal address of the next instruction. If this is the case, repeat the job and ROAR stop on word 1002. When a match occurs, the next instruction address may be obtained by displaying bytes 83 and 84 of Aux. Storage A. Working back from this instruction, determine the failing operation. Suspect misplaced wordmarks.
0E	The program attempted to index without advanced programming being defined by CID.	Check bit 1 of byte 9C in Aux. Storage A. This bit should equal 1. Card 0390 of the standard CID contains this definition.
11	Some device other than the 1050 attempted to take a multiplexor-channel data-cycle while preparing to enter the 1050 data loop.	While waiting for TREQ to come up, another device sent its address in an attempt to take a data cycle. CAS logic page QG121, ROS word 1944 detects the Address-In signal. This routine is normally used to stack any outstanding status so that the channel is free to operate with the 1050. Check to ensure that this data-transfer request is not from a device which was initiated in Basic 2030 mode prior to a mode switch to 1400 mode. This is defined as an invalid operation in the PMS portion of the IBM System/360 Model 30 1401/1440/1460 Compatibility Feature Systems Reference Libary. If the operation was initiated in 1400 mode, a hardware problem probably exists with the device whose address was sent in. Also check that the proper signals are received by the device which was forced to stack its status.
21	A word mark was missing from the 1400 op-code during I-Cycles.	Usually occurs when a branch is made to a location not containing a word mark. The decimal BA-address points 1 beyond this location. There is no easy method to determine the branch from address. Several steps which may be taken in order to further define the problem include: (1) SAR stopping on the location to which the branch has occurred to determine how it is changed during program initialization and execution. (2) Determine the operation when the failure occurs (perhaps through observing an I/O operation prior to the failure) and then operating in instruction mode to pinpoint the failing instruction.
22	A wrong address was sent back from the channel during initial selection of a tape operation.	Restart the operation. If tapes are on the multiplexor channel, ROAR stop on address 1AA0 (CAS logic page QG761). The desired unit address can be found in the D-register. Execute the next micro word 1AA4; this sends Address-Out. Check the TCU to determine that the proper address was received. Place rate switch to process mode and early ROAR stop on word 1A93. If entry to 1A93 is via 1A92, Address-In came up. This address may be determined by displaying FI. If this address is different than the contents of the D-Register, a TCU problem probably exists. If entry to 1A93 is via 1A41, an invalid channel condition exists; for example: Op-in and Select-In up simultaneously. If an address-mismatch condition does not exist at this time, single-cycle through the next several words to ensure that the proper address-compare branch takes place in word 1A98 or 1A99. If tapes are on a selector channel, ROAR stop on word 1A94 (CAS logic page QH021). The desired unit address can be found in the MSDR. This address is sent out to the TCU with the execution of ROS word 1A97. In word 1AA0 the address sent from the channel is compared to the address sent out. The address-compare branch takes place in word 1AA2.
23	GMWM in first position of tape write field.	A tape write command was issued. The first character to be written was a GMWM, which terminated the operation. Check the first data location to determine if there actually was a GMWM in this position. The decimal address in UV at this halt points 1 beyond the GMWM location. The B-field of the previous op-code points at the first data location. Determine how this GMWM was inserted in this field. Suspect tape drive failure or invalid programming. If the same situation were to occur on a 1400 system, the tape drive would have 'run away'. Check that this is not a delibertate operation by the customer.
31	A word mark was detected in the A-address of an I/O instruction, for example: M%U1500R	SAR stops on the main storage location containing the erroneous word mark. Determine when and why it is being set. Remember when SAR-stopping in compatibility mode, the hexadecimal Basic 2030 main storage address must be used rather than the 1400 decimal address. This can be obtained by performing a set IC operation to the decimal address and reading the hexadecimal address in the MN-register lights.
3F	A 2540 or 2501 reader error, a 2520 PFR error, or an invalid character has occurred.	Check the read-in area (001-080); any invalid characters have been replaced by a blank. Check the last card in the stacker for an error. If no invalid character has occurred, then unit check has been sent with the channel-end status from the reader; this means that an equipment-check condition, such as a hole-count error, has been detected.
41	An 8F character was detected at an address other than the bias address while in 1400 mode. For a 65K machine, this may indicate a 1400 high-wrap condition because 256 8F characters are placed between the upper limits of 1400 storage and the highest core address.	Detecting this character while in 1400 mode forces a microprogram trap to ROS word 1002. The displays at WX = 10FF cannot be guaranteed with this halt. In order to determine the location of the 8F character, check the BA-address and determine if it contains a valid address. Is there a word mark in the BA-address location? If BA contains a valid instruction address, determine what the failing op-code was; this may define the storage areas where the 8F was detected. Also check the A and B stars; these may point directly at the BF character or 1 position on either side of it. If these methods fail, ROAR stop on word 1002. When a match occurs, the failing location is displayed in the MN lights in hexadecimal form and the previous instruction address is in bytes 88 and 89 of Aux. Storage A (in hexadecimal). Once the failing address is known, SAR stopping on this should point out how the 8F was generated. It may occur as a result of a disk-out-of-sync condition or it may be read in via a Basic 2030 prepared tape where 8F is a valid character.
42	Invalid channel status was detected following a tape-read operation (Selector Channel only).	ROS word 1A30 gates the GJ-assembler containing the channel status on to the Z-bus. The following word branches on Z = 0; if Z is non-zero, this halt occurs. The bit configuration of GJ at word 1A30 is: 0 - Program controlled interruption 1 - Incorrect length 2 - Program check 3 - Protection check 4 - Channel data check 5 - Channel control check 6 - Interface control check 7 - Chaining check If this halt occurs while operating in 1400 mode, a channel hardware problem probably exists. Check the selector channel chart. Chart S at SA.
4F	1442 Reader or Punch intervention required.	This stop indicates that a Not Ready condition exists at the 1442. This condition usually results from: (1) an empty hopper (2) a full stacker, or (3) an open cover.

DT Chart F 1400 Compatibility Oriented (Sheet 3)

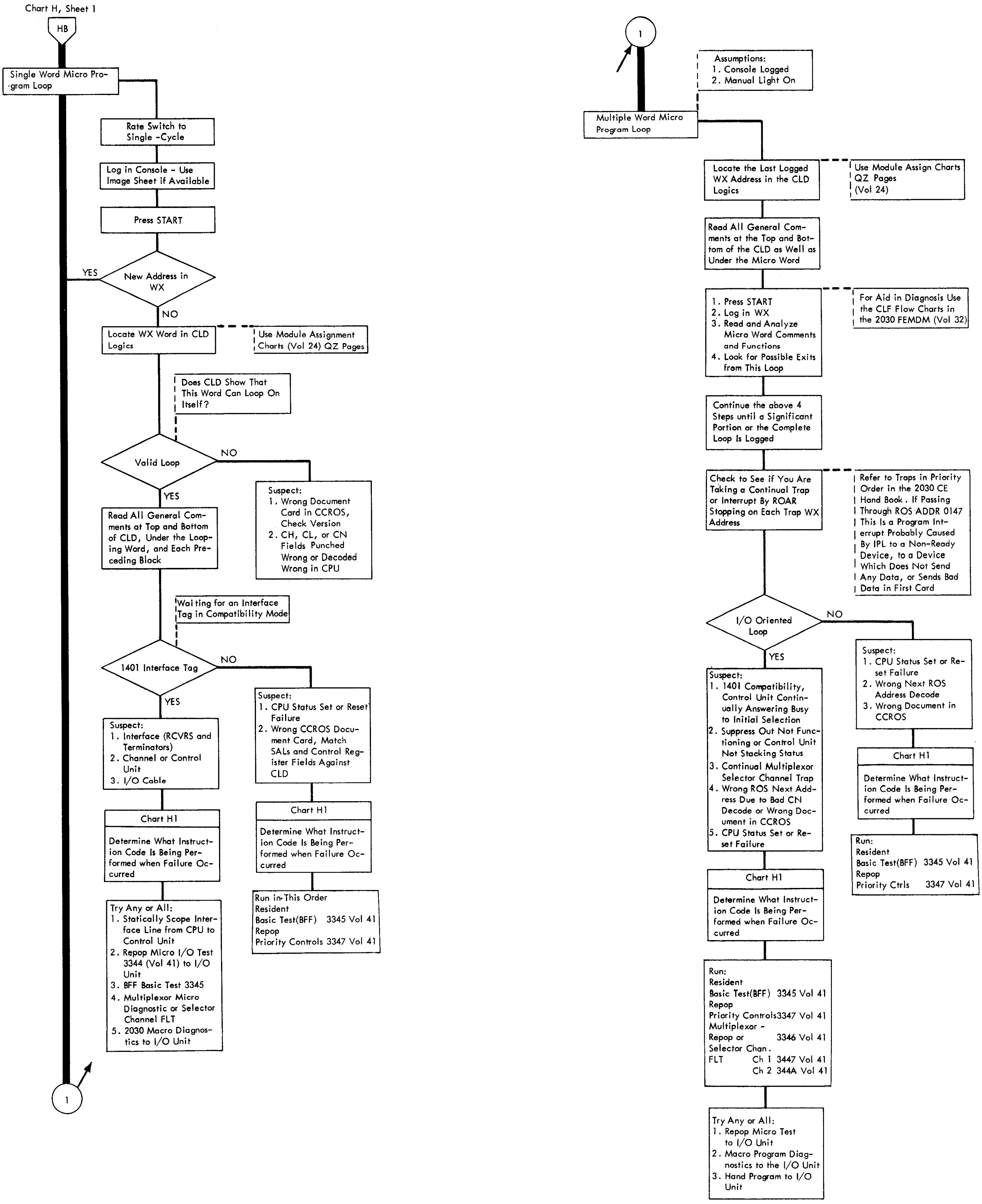
Digit Displayed In MSDR	Reason For Halt	Comments
51	The program attempted to perform an I/O operation on a device for which the compatibility feature is not installed.	Working back from the BA-address, determine what the invalid operation is, for example, M%Y1201W for a machine with 1402/1403 subfeature. If this occurs while running a diagnostic test, check the TAD(s) to determine that the proper types of I/O devices are described. If this occurs while loading CID, the IJ backup locations that point to the first 1400 instruction may be erroneously set. Columns 28 and 29 of card 0380 specify the Basic 2030 main storage locations from which the first 1400 instruction is read. This instruction is contained on card 00A0 or 00B0 of the CID and is loaded into the 2030 main storage. If the halt occurs while attempting a 1729, 1402, or 1442 load, the card-load or tape-load IJ address specified in bytes 90 through 93 of Aux. Storage A may be erroneously set. These locations are used to set IJ when 1729, 1402, or 1442 load is attempted. These locations should contain the addresses of a card-read or tape-read instruction located in Basic 2030 storage. These addresses are loaded from card 0390 and are contained in columns 6, 7, 8, and 9 for card load and 11, 12, 13, and 14 for tape load.
52	A device-end signal was received before a GMWM was encountered on a tape-write operation. This halt can occur on both selector and multiplexor channels.	This is probably a tape-drive or control-unit problem. If tapes are on a selector channel, refer to CAS logic page QH071. Word 1763 branches on Status-In and Service-In. If Status-In occurs before the group-mark word-mark, which normally terminates the operation, this error halt occurs. If tapes are on the multiplexor channel, refer to CAS logic page QG791. Word 1A40 or 19C4 branches on Status-In and Service-In. Do not attempt to ROAR stop on a word containing the Status-In, Service-In branch; these are part of the data loop and cause overrun conditions, etc., when the match occurs.
55	A 1400 start reset function was performed.	
62	For tapes on the selector channel, this stop means that Status-In and Service-In came up at the same time during a tape-write operation.	This is probably a TCU problem. CAS logic page QH071 contains the word where the branch is made.
62	For tapes on the multiplexor channel, this stop means that the TCU disconnected during a tape-write operation.	This is probably a hardware problem. ROS word 1A40 or 19C4 (CAS logic page QG791) contains the branching conditions which detects this failure. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
7F	The program has issued a stacker-select command to a 2540 more than 6 ms after the previous card-read instruction was issued.	Check the program to determine if a lengthy I/O op was issued between the Read command and the stacker select. Also check the 2821 to ensure that the attention bit is sent to the processor during the 6 ms provisional feed time.
80	If this halt is used in conjunction with the 1403/2540 or 2501/2520/1403, it means either that there was no address compare (Address-In was not the same as Address-Out) or that a punch-transfer error occurred.	Early ROAR stop on word 1892. If the previous address displayed is 181B, a punch-transfer error has occurred; that is, an error occurred while filling the punch buffer. It can either be a Bus-Out check or an error internal to the punch which would result in bad parity data being transferred into the buffer. If the previous address was either 1896 or 1897, a no-compare condition exists between Address-Out and Address-In. This is probably a hardware problem in either the channel or the device. Check that the proper address was sent out and received by the device. Address-Out is sent during the execution of word 1801 (CAS logic page QG531). Also check the address sent in to verify the comparison and branch executed by the microprogram. The address compare is made in ROS word 1898 (CAS logic page QG531) and the branch is made in the following word.
80	When this halt occurs on a machine that has the 1442/1443 compatibility subfeature installed, one of the following situations has occurred: (1) A no-compare condition exists between Address-Out and Address-In. (2) A data-transfer error has occurred during a printer operation. (3) The print field was not terminated by a GMWM.	Repeat the operation and ROAR stop on ROS address 188A (CAS logic page QG261). If a match occurs, Address-In does not match Address-Out. The address sent out is contained in R-Register in its crossed form at this point. The address sent in can be displayed in FI. This is probably an I/O hardware problem. Check that Address-Out is properly received and decoded by the device; Address-Out is sent in word 1821 (CAS logic page QG311) and the address comparison is made in ROS word 1887 (CAS logic page QG301). If a match does not occur, there has been an error during data transmission, a unit check has occurred at channel end. This is probably either a Bus-Out error or a device error associated with receiving the data. This stop is set up on CAS logic page QG351 word 1842. Also, check if a GMWM (0F Hex) is in the print field; the absence of this character can also result in a unit check at channel end.
82	For tapes on the selector channel, this stop means that Status-In and Service-In came up simultaneously on a read-move operation.	This is probably a TCU problem. CAS logic page QH081 contains the word where the branch takes place.
8F	Tape-unit intervention is required for tapes on the multiplexor channel or on the selector channel. At this halt, IJ has been decremented to the op just attempted. BA points to this decimal address. Pressing START allows the operation to be tried again.	Unit check is in the status response to Command-Out during initial selection. Check that the TCU is on-line, that the desired tape drive is ready, that Aux. Storage properly defines the drives and control unit. (See the description for the 92 halt for proper definition), that a write operation is not being attempted on a file-protected drive.
90	The program has attempted to use an invalid d-modifier on a 1442/1443 operation.	Check the program residing in core storage to determine if the d-modifier is actually valid. If it is invalid, attempt to determine how it was introduced into the instruction stream. If it is valid, check the microprogram decode. The stop code is set in ROS words 1816, 188B and 1893 (CAS logic page QG261).
90	Operational-In disconnect on 2540 or 2501 reader.	Op-in dropped during a read operation. This is probably a hardware problem. ROS word 18CD (CAS logic page QG561) contains the branching conditions which detects this failure. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
92	A tape error occurred on a tape initial program load.	A unit check is contained in the read ending-status of a 1729 tape load. This end status can be located by displaying FI after executing word 1A46 (CAS logic page QG802) when the tapes are on the multiplexor channel. When the tapes are on the selector channel, this ending status is displayed in GR following the execution of word 19D5 (CAS logic page QH081). When performing a tape IPL (1729 in switches FGHJ--Press SYSTEM RESET - ROAR RESET - START), the machine is forced to ROS word 1729 (CAS logic page QE521). This routine loads IJ-backup with the contents of tape-load-IJ (bytes 92 and 93 of Aux. Stor A). The instruction located at the address specified by tape-load-IJ is L%U1001R and is read into the machine from card 00A0 of the CID. After performing a start-reset function, the microprogram IJ-backup is loaded into IJ, and I-cycles are entered. After completing the read, IJ is forced to 1400 storage location 001 and processing of the data just read begins. Check that this is done correctly if the 92 halt occurs. Also check that the proper drive assignments have been made. Byte 81 of Aux. Stor. A must properly define the Basic 2030 drive containing the initial program: Bits 0 and 1 00 - 7-track @ 200 bpi 01 - 7-track @ 556 bpi 10 - 7-track @ 800 bpi 11 - 9-track Bits 2 and 3 should be zeros. Bits 4 - 7 contain the Basic 2030 drive number which is to be assigned as 1400 drive #1. Bits 4 - 7 of Byte 80 of Aux. Stor. A contains the TCU address. In addition, bytes 97 and BB of Aux. Stor. A contain 7- or 9-track status information and must be initialized as follows: Bit 0 of each byte has 7- or 9-track status for tape unit 0; bit 1 of both bytes is used for this status for tape unit 1, etc. If a bit of the first byte (position 97) is set to 1, the associated tape unit is equipped with 9-track read/write facilities. The unit is a 7-track drive if the associated bit in position 97 is set to 0. If the corresponding bit of the second byte (position BB) is set to 1, the associated tape unit is a 9-track phase-encoded drive with density set at 1600 bpi. If this bit is set to 0, the associated tape unit has a density setting of 800 bpi and is either a 7-track or 9-track unit.
A0	An Operational-In disconnect occurred during a 1403 print operation.	Op-in dropped during the data-transfer portion of a print operation. This is probably a hardware problem. ROS word 18BC (CAS logic page QG551) contains the branching conditions which detects this failure. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
A0	There was no GMWM in storage to terminate a 1442 read operation.	On a 1442 read operation, a GMWM (0F hex) is required to terminate the operation. Manually check the read-in area for this character; if there is a GMWM in read-in area, this is probably a hardware problem with the GMWM detection latch (ALD page MB031). If this halt occurs after mode switching from Basic 2030 mode to 1400 mode during initialization, check CID card 0384, columns 28 and 29 and card 0390, columns 8 and 9. These columns should each contain AD. Also check card 37. For non-65K mach, columns 18 and 19 should be 09. For 65K machines, check card 38, columns 18 and 19. These should also contain 09. These cards cause a GMWM to be inserted in storage location 0082 of 1400 storage when compatibility mode is initialized via the 99 op code.
A2	Invalid selector-channel status was received on a 1400 branch-if-tape-error instruction.	Non-zero channel status was sent back from the channel on a TIO; Word 1A12 gates channel status on to the Z-bus (QH111). This is probably a channel hardware problem. Also check that the proper command was sent to the channel. Command-Out is sent during the execution of word 1AA4 (CAS logic page QH021). The command byte sent should be 00. Word 1AA9 (CAS logic page QH021) should reset GR prior to issuing Command-Out in word 1AA4. Check the selector channel chart (Chart S at SA) for additional assistance.
B0	An Operational-In disconnect occurred during a 2520 or 2540 punch command.	Op-in dropped during the data-transfer portion of a punch operation. This is probably a hardware problem. ROS word 1820 (CAS logic page QG521) contains the branching conditions which detects this failure. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
B0	A printer error on a print operation or a 1442 error on a read or punch operation has been detected.	If on a read command, this signifies either an equipment check or an invalid card code. For the punch command, this halt identifies an equipment-check condition. ROAR stopping on word 1877 (CAS logic page QG281) allows displaying the sense byte in the G-register. The sense byte makeup is: 0 - Command reject 1 - Not ready 2 - Bus-Out check 3 - Equip check (a) reader check (b) punch check (c) invalid card code punched (d) data error at channel end 4 - Data Check - invalid card code on read 5 - Overrun check If performing a print operation, the sense byte may be displayed in the G-register by ROAR stopping on word 1876 (CAS logic page QG361). The sense byte content is: 0 - Command reject 1 - Intervention required--printer not ready because forms check, stop key, or carriage-stop key pressed , or cover interlock open. 2 - Bus-Out parity check 3 - Equipment check 4 and 5 - Typebar selection 6 - Channel 9 7 - Channel 12 If a 1403 is operated with the 1443 microprograms, the sense byte is: 0 - Command Reject 1 - Intervention Required--Not Ready 2 - Bus-out Parity Check 3 - Equipment Check--Parity Error in CU or device 4 - Data Check 5 - Not Used 6 - Channel 9 7 - Channel 12

Digit Displayed In MSDR	Reason For Halt	Comments
B2	Status-In and Service-In occurred simultaneously on a 1400 read-load operation, for example: L%UXBBBR This halt is valid only when tapes are on the selector channel.	ROS word 1A63 (CAS logic page QH091) has this branch. Both Status-In and Service-In up at the same time probably indicates a channel or device hardware problem. Also check the set and reset of S4 and S5. These are the status indicators on which the branch is made. Use the selector channel chart (Chart S at SA) for checking the channel.
C2	Operational-In disconnect on a multiplexor channel tape-read operation.	Op-in dropped on a tape-read operation. For a read-move operation, the branch is found on CAS logic page QG801; for a read-load operation (page QG802). This is probably a channel or device hardware problem. Also check the Status-In and Service-In branches. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
CE	This halt indicates that a premature data disconnect occurred on a read operation. Pressing the start key causes processing at the next sequential instruction but loses the record just read. To re-read, the operator must set the instruction counter to the address of a backspace-re-read routine.	The following conditions have occurred when this halt is reached: (1) Unit check in read end-status. (2) 12 or fewer characters have been transferred into storage. (3) The noise bit has been set in the TCU and has been recognized as bit 0 of the 2nd sense byte sent from the TCU. The noise bit being turned on implies that after the data-disconnect additional bits were found before the IRG was detected. Tape movement was continued until an IRG was recognized. Therefore, the data encountered after data transfer was halted has been skipped over. This halt generally indicates a problem with the tape drives. Check the following: (1) IRG (2) Start/Stop adjustment (3) Dirty idlers/prolays (4) Ungrounded or improperly grounded drives.
CF	1050 intervention is required.	Check for an out-of-paper, power-off, or off-line condition. Bit 5 of the 1050 tags indicates this intervention-required condition. These tags are gated to the G-register in ROS word 1946 (CAS logic page QG121). Branch on G5 to check for the intervention-required condition (CAS logic page QG141 word 197A).
D2	A microprogram initiated sense operation was prematurely ended. This halt can only occur if tapes are on the multiplexor channel.	This is probably a hardware problem. ROS word 1A8D (CAS logic page QG781) contains the branching conditions which detects this failure. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
E2	Operational-In disconnected on a microprogram-initiated mode-set operation with tapes on the multiplexor channel. This mode-set command is issued in order to send the track-in-error byte previously stored to the TCU.	ROS word 1A6C (CAS page QG781) contains the Status-In and Service-In branches for the track in-error mode-set command. If Op-In drops, this error halt occurs. This is probably a channel or device hardware problem. Taking the 1,1 branch does not mean that Status-In and Service-In came up simultaneously, but rather that Op-In dropped. Check the applicable hardware on ALD pages FA092 and FA041. Also refer to Chart D at DA.
F0	A 1400 halt instruction has been properly executed.	Not an error halt.
F1	A 1400 halt instruction with an invalid B-address has been successfully executed.	Not an error halt.
F2	A 1400 halt instruction with an invalid A-address has been successfully executed.	Not an error halt.
F3	A 1400 halt instruction with an invalid A- and B-address has been successfully executed.	Not an error halt.
FF	A 1400 halt-and-branch instruction has been properly executed.	Not an error halt.
10	Read-back check stop. A disk write instruction was followed by a disk instruction other than a read-back check.	If the I/O stop was caused by operator intervention, press SYSTEM RESET to clear the read-back check interlock, set IC to the beginning of the write operation, and press START to continue processing. If the I/O stop is caused by programming--a write disk without a subsequent read-back check--the program is invalid and should be corrected.
20	Channel-end or device-end not received during ending status.	This condition is tested for on QH381, location J3. At this time, a 2030 disk command should have been completed, and ending status posted by the file control unit. Ending status was detected at location J1 on QH381. The last 2030 disk command issued can be read from the B Aux. Storage XXBE. To recreate the 1400 disk operation that led to this coded stop, find the address of the next sequential instruction displayed in the B-star and A-star registers. The beginning of the disk instruction is this address minus 8. It is necessary to set up the disk control field correctly each time before executing the 1400 disk operation. To set up the disk control field prior to executing the operation, a disk control field can be moved into its working location with a move data or load data op-code. Note: If the disk operation is a multi-sector operation, a microprogram stop at the J3 location (QH381) causes time disorientation. It is necessary to reset and restart at the beginning of the disk operation.
30	Wrong address sent back from the channel during initial selection.	This stop is probably caused by a hardware malfunction in the multiplexor channel bus cabling or associated hardware. A test I/O instruction to the 2311 in question can be used to determine if the problem exists in Basic 2030 mode. The 2841 file control unit hardware that receives the Address-Out from the processor is located on page GA071 (2841 ALD Page). The 2841 FCU initial selection microprogram is on QB010. During initial selection, the microprogram moves from location E2 to A6, where Address-In is raised. It is possible to single-step both the processor and the 2841 microprograms through the initial selection sequence. The disk compatibility initial selection microprogram is on page QH341 (2030 CLD Page), locations G5 through L8.
40	Unit-check status response to seek command, following address transfer.	This stop happens if the seek address sent to the file control unit is outside the range of addresses acceptable to the FCU. The address-transfer portion of a full seek is decoded to binary and stored in the B-auxiliary storage in addresses XXAD, E, F, and XXB0. The bytes stored in these locations for a seek operation are the CCHH values. The highest legitimate value that can be stored, is hex 00 CB 00 09, which represents cylinder 203, head 9. If for any reason, a previous read operation placed non-zero values in addresses XXAD or XXAF, which were not removed by a built-in microprogram housekeeping program, the seek check can be removed, as an interim measure, by manually performing the following steps: (1) Note the B-and A-reg display of the 1400 program next address. (2) Press SYSTEM RESET to clear 2841 microprogram notation. (3) Set the processor to 1400 mode. (4) Set IC to I-next minus 8. (5) Press START to retry the seek operation. This type of stop has been eliminated for all known cases used by customer programs.
50	Operational interlock	This stop occurs when the control unit is disconnected from the processor, and the processor microprogram is at a microaddress, where it should be only when the two boxes are logically connected. The processor disk-compatibility microprogram can exit to the 50 stop from several places. This exit is taken whenever the processor microprogram is at a 4-way branch, waiting for service-in or status-in. If the control unit is disconnected, the 11 branch is taken to the 50-coded stop. During the execution of disk operations, the file control unit should stay connected until the operation is completed. The most probable time at which this stop can occur is when the disk compatibility feature is updated and a wrong EC level or incorrect CCROSS card is inserted. A wrong next-address in a microprogram block can put the microprogram into one of the 4-way branch blocks that leads to a 50-coded stop. It might be possible to correct this condition rapidly by checking the EC level and version of CCROSS cards installed during change activity. Another approach is to use EARLY ROAR STOP on the processor, and work back to the wrong CCROSS card. This approach is based on the assumption that 2030 disk diagnostics had run successfully, and that the file control unit is working correctly. Disk compatibility CCROSS cards are from location 79-0 to 84-7.
60	Module overflow detected. The disk module value in Aux. Storage B B011, 2, 3, 4, or 5 is different from the module value specified by the disk control field of the disk instruction.	(Limited Usage). Change disk packs, if program depends upon module overflow to indicate pack replacement. All others: Set B010 Aux. Storage B to 40 hex. Set IC to 1400 displayed address -8, press console START to continue. In the IBM System/360, Mod 30 1401/1440/1460, Compatibility Feature Manual, Form No. A24-3255, see Appendix F card 410 - Disk File Functions, to determine whether this stop can be disabled for customer application. See Appendix D, Storage B Byte 10 Bit 1 on Module-Overflow Detection usage. Most users can disable this stop, and must disable this stop if they use labels on drives other than drive 0.

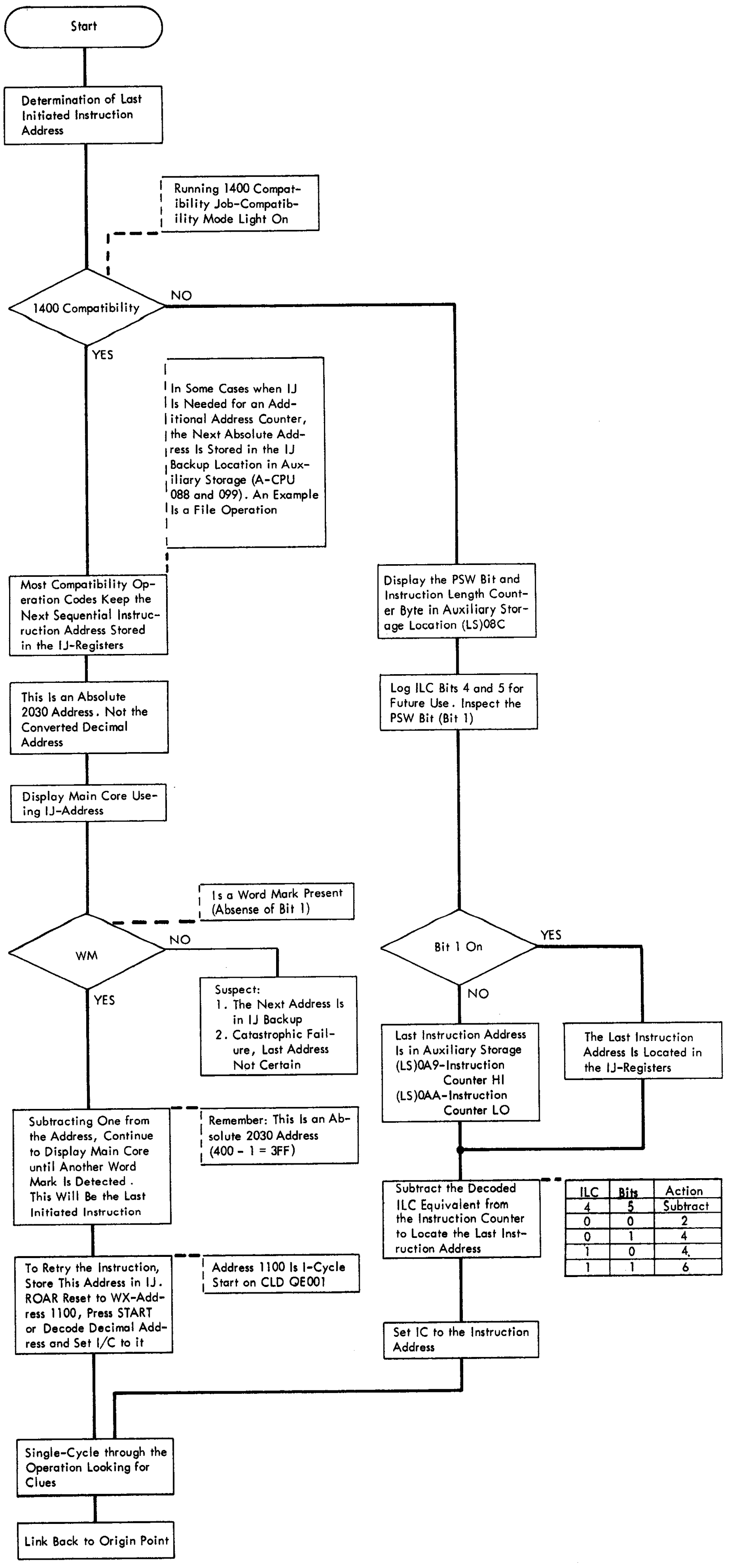
DT Chart F 1400 Compatibility Oriented (Sheet 5)



DT Chart H Hang Ups, Loops, and Stops (Sheet 1)

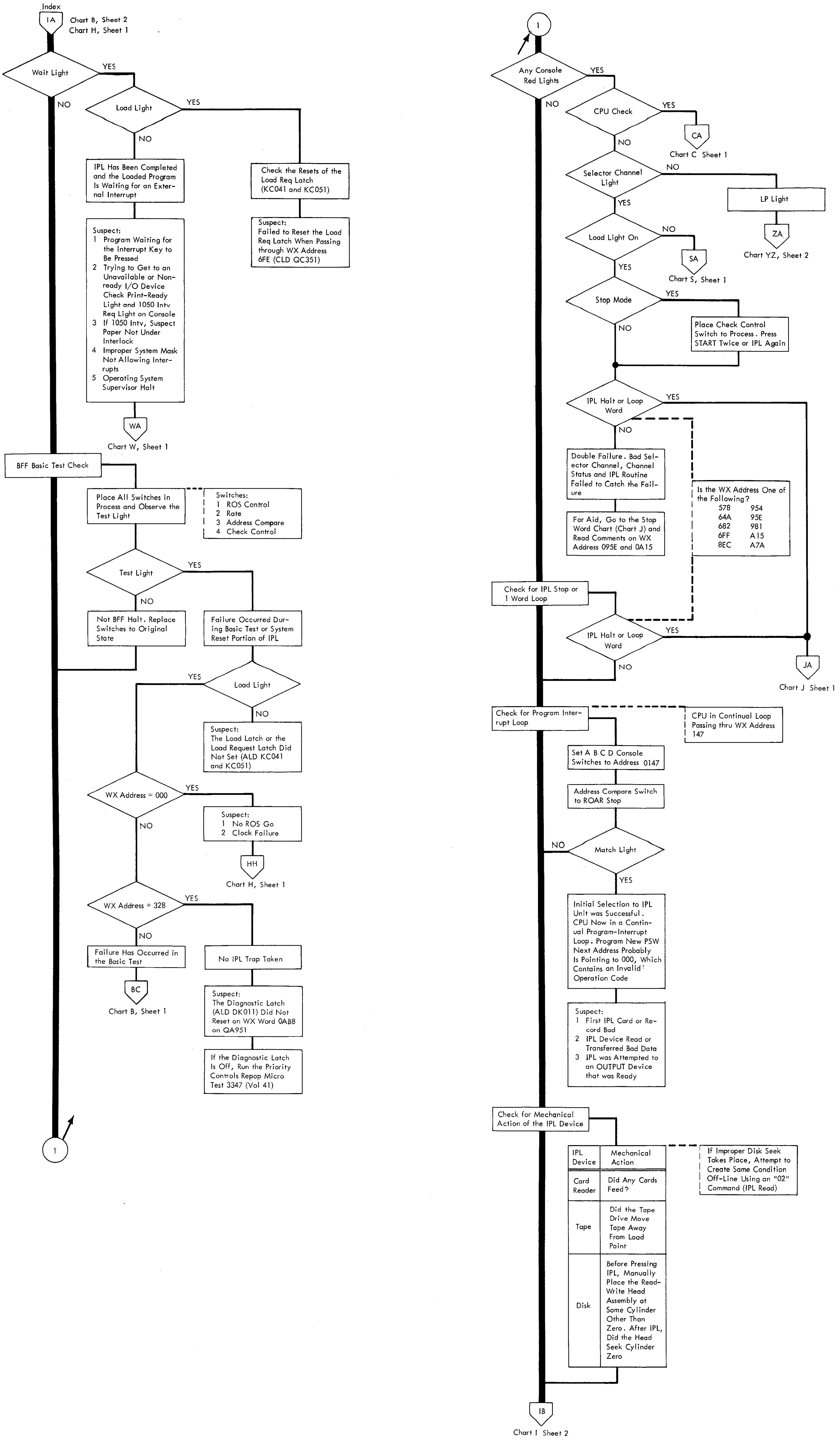


DT Chart H Hang Ups, Loops, and Stops (Sheet 2)

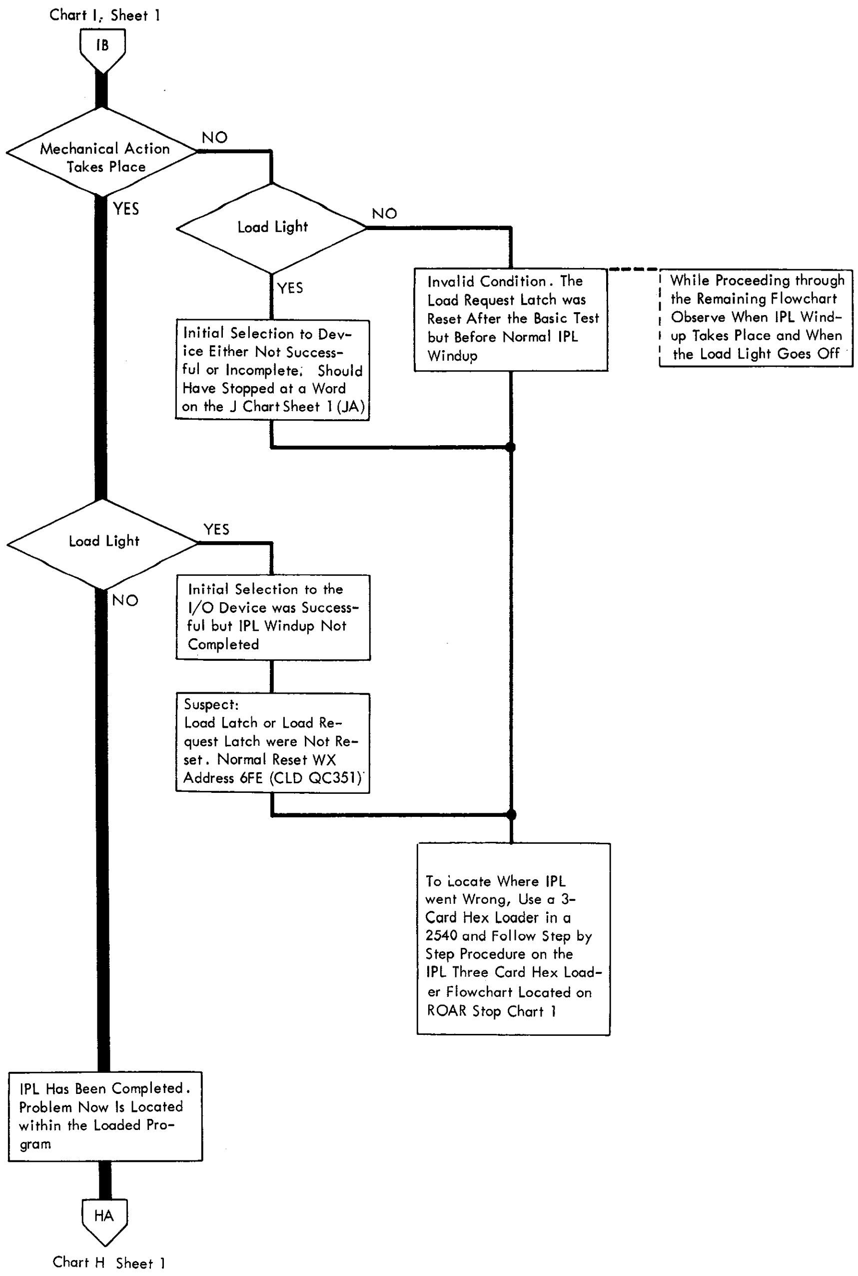


ILC	Bits	Action
4	5	Subtract
0	0	2
0	1	4
1	0	4.
1	1	6

DT Chart H1 Last Initiated Address



DT Chart I IPL (Sheet 1)



DT Chart I IPL (Sheet 2)

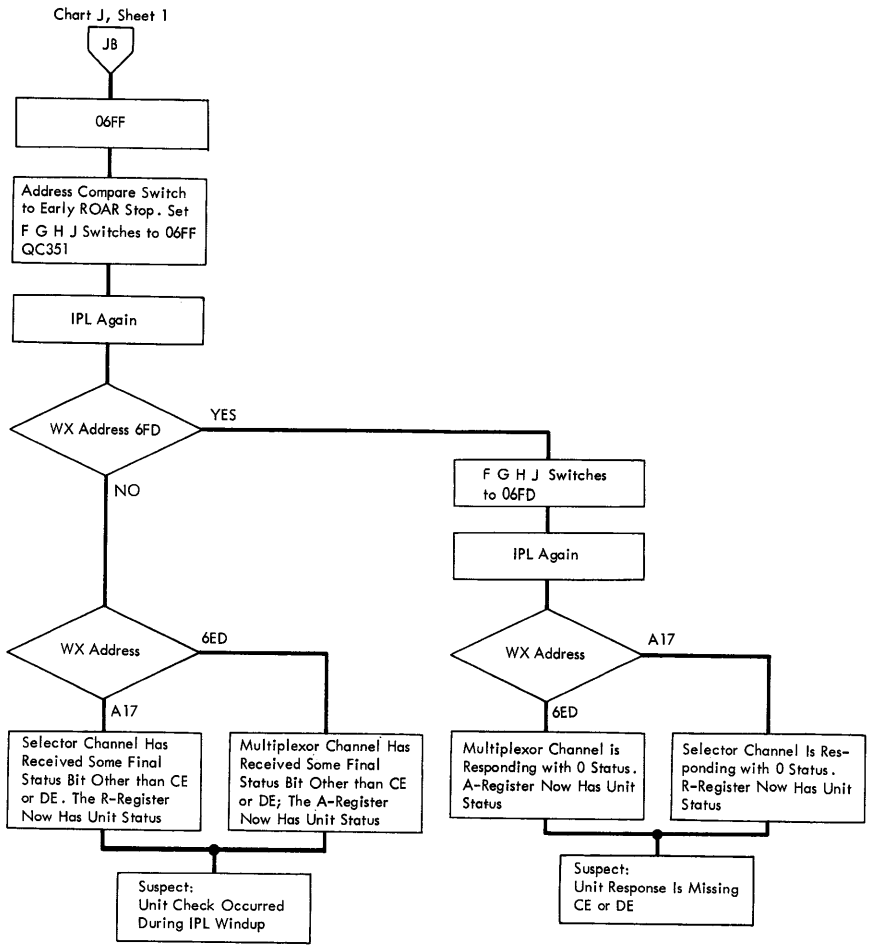
JA

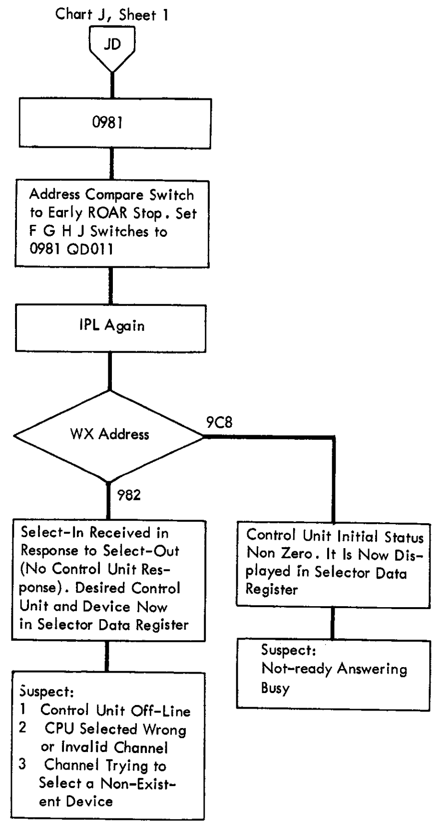
Index
Chart I, Sheet 1
Chart H, Sheet 1

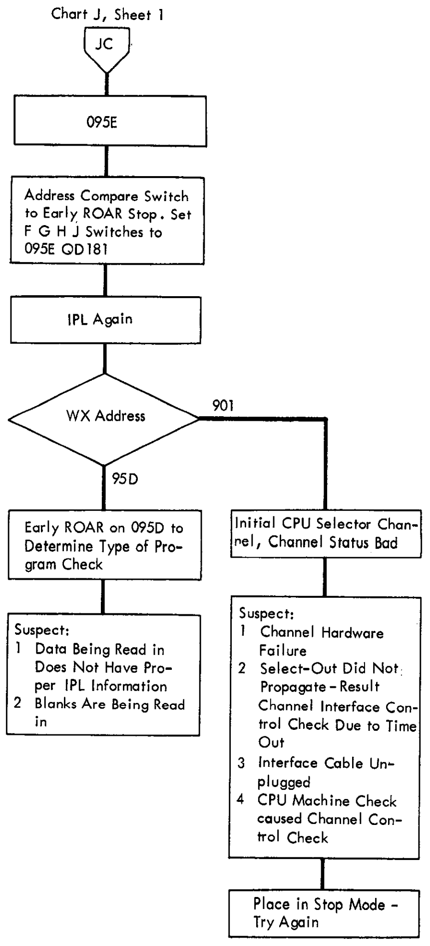
Note 1: All CPU CF stops are listed except diagnostic test stops and 1620 feature stops.
* For additional aid in locating IPL failures, use a three-card hex loader in a 2540 and follow the step-by-step ROAR-STOP procedure on the IPL three-card hex loader flowchart located on ROAR stop Chart 1

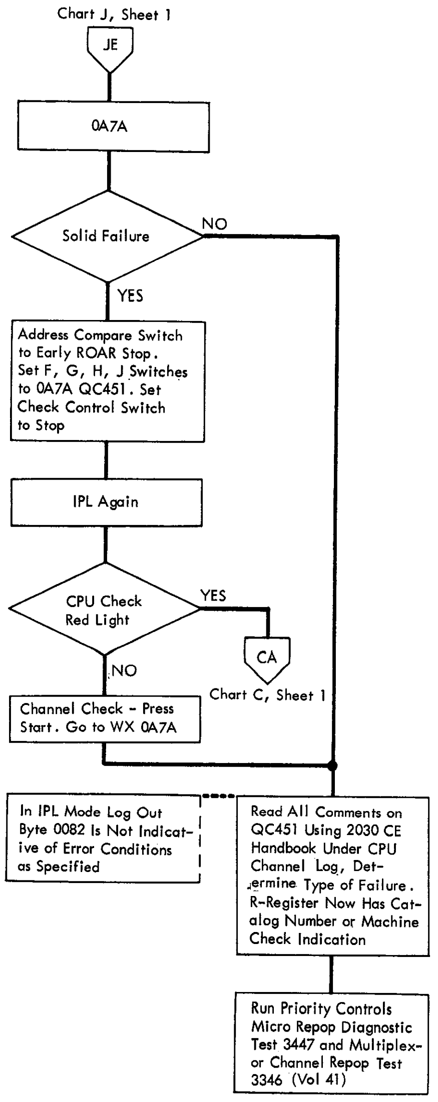
WX ADDR	Type	CLD Page No	Normal Cause	Hints and Aids
00B2	Stop	QA961	A second machine check occurred with check control switch not in stop, before the first machine check could be logged.	Check control switch to stop; try again or go directly to CPU check. Chart C, Sheet 1 at CA
00FF	Stop	QA941	Soft stop occurred because STOP key was pressed. Rate switch was in instruction step, or match occurred using SAR delayed stop mode.	Check process stop latch on ALD KC081
0328	Stop	QA951	System reset followed by pressing the START key once -- basic test BFF has run successfully 128 times.	If IPL initiated, suspect the multiplexor channel. Trap was not taken at previous WX 00EA, best action run priority controls repop micro test 3447
0578	IPL Stop	QC091	Attempted to IPL to an unavailable or non-existent device on the multiplexor channel, Select-In received in response to Select-Out.	Suspect: * 1. Desired unit is off line 2. Desired unit does not recognize its address 3. Wrong address being sent out on channel Do: ROAR stop on ROS address 05A0 - QC051 Check Bus-Out and control unit statically
0649	IPL Stop	QC061	IPL was attempted to a non-ready device on the multiplexor channel.	Suspect: * Control unit answers busy to initial selection Do: ROAR stop on address 05AC, rate switch to single cycle, press START, A-Register now contains status from device.
0682	IPL Loop	QC351	1. During IPL on the multiplexor channel, the control unit failed to send Request-In or the CPU did not recognize Request-In, which is necessary to cause a multiplexor channel share request trap 2. First card read did not have proper IPL data or data read in was blank. This is one way of causing a program check that causes this loop	Suspect: * 1. If only one card fed into the stacker, suspect that it is not the proper IPL card 2. Request-In from control unit not getting to CPU. Check Request-In on ALD FA082 3. CPU failure to take multiplexor channel trap to WX 0010. 4. Operational-In appeared dropped to CPU 5. Reader clutch failed to energize Do: Run priority controls micro repop diagnostic test 3447 and multiplexor channel repop test 3346 (Vol 41)
06FF	IPL Stop	QC351	Bad final status received from source unit during IPL on the multiplexor or selector channel	Do: * Go to Chart J, Sheet 2 at JB
08EC	IPL Stop	QC511	IPL attempted using address assigned to the 1052 console KB-Printer	1. ROAR stop on WX Address 00EA - QA951 and IPL again - If no match, place rate switch to single cycle and IPL again, scope diagnostic latch on DK011. This latch is not on while executing basic test causing a premature IPL trap to 0002 * 2. If a match occurred in 1, ROAR stop on 053A - QC041 and single-cycle through address formation for clues
0954	IPL Loop	QD091	Selector channel wait loop, waiting for selector channel chaining trap or share cycles.	Suspect: * 1. CPU failed to take selector channel trap 2. Control unit failed to bring up Status-In or Service-In 3. Selector channel poll control problem Do: Run priority controls micro repop diagnostic test 3447 and selector FLT 3447 - 344A (Vol 41)
095E	IPL Stop	QD181	1. Initial internal channel status not equal to zero on selector channel IPL; or 2. Program check occurred during IPL on selector channel.	Do: * Go to Chart J, Sheet 2 at JC
0981	IPL Stop	QD011	1. Initial status from source unit during IPL on selector channel is non zero, or 2. IPL attempted to non existent or unavailable selector channel	Do: * Go to Chart J, Sheet 2 at JD
0A15	IPL Stop	QD121	Non-zero channel status detected at completion of IPL on the selector channel	ROAR stop on WX Address 0A01 - QD121, rate switch to single-cycle press START. R-Register now contains ending channel status *
0A7A	IPL Loop	QC451	1. A machine check occurred during IPL while check control switch was in process 2. A multiplexor channel control check occurred during IPL 3. Interface time-out occurred during IPL	Do: * Go to Chart J, Sheet 2 at JE
10FF	Stop	QE691	1401 Compatibility coded stop, R-Register hexadecimal character determines cause	Do: Go to Chart F, Sheet 3
12F4	Stop	QE561	1. Machine check occurred in Compatibility mode with check control switch in process; or 2. Second machine check occurred in Compatibility mode using mode switch on machine checks	Place in stop mode and catch error when it occurs, or go to Chart C, Sheet 1 at CA
1E4F	Stop	QG532	An RPQ operation (branch on next card column binary) was attempted but reader was not ready	Ready reader or ROAR stop on WX Address 1E4E - CLD QG532 and observe Status-In in multiplexor channel and control unit

DT Chart J CF Stops and Special One Word Loops (Sheet 1)

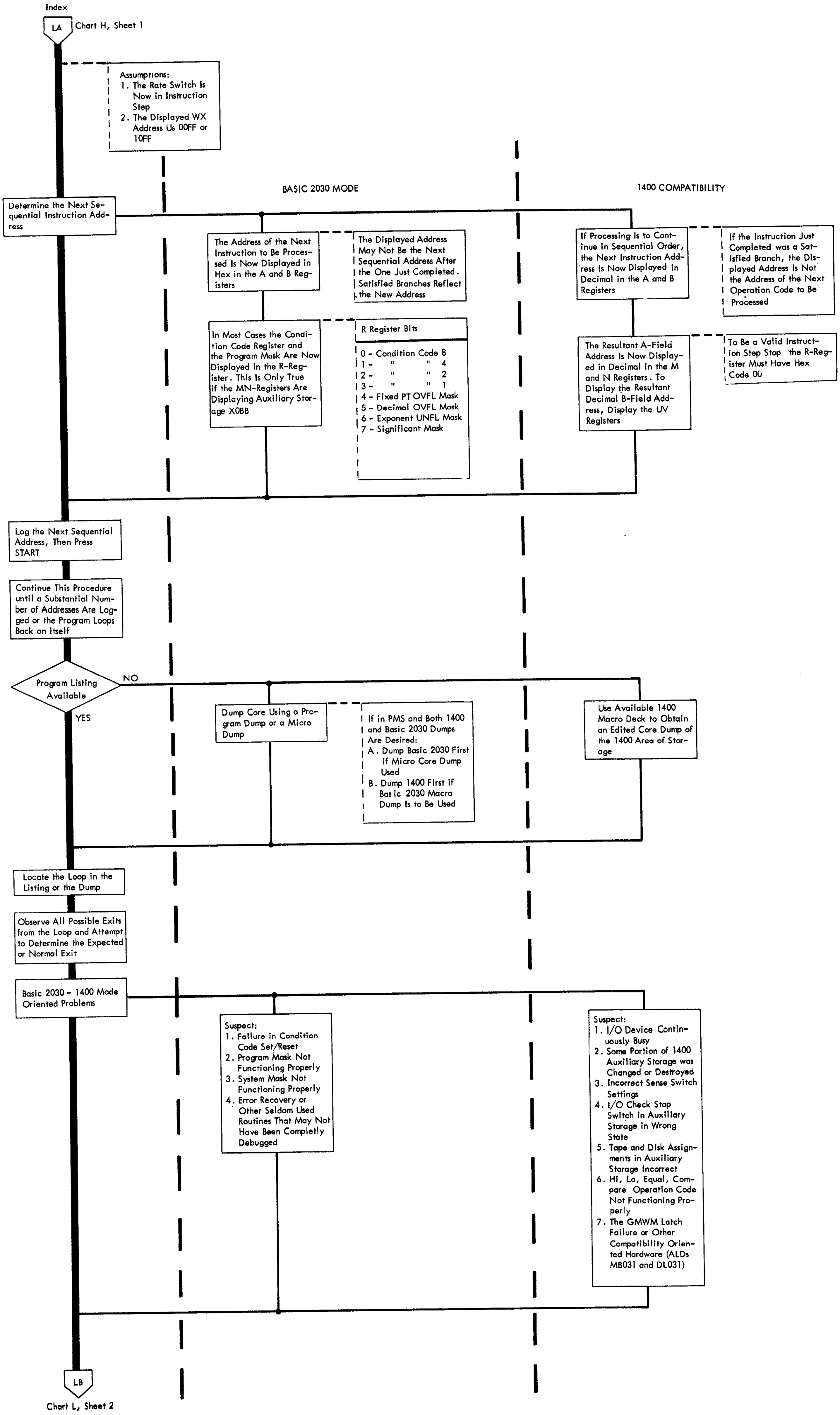




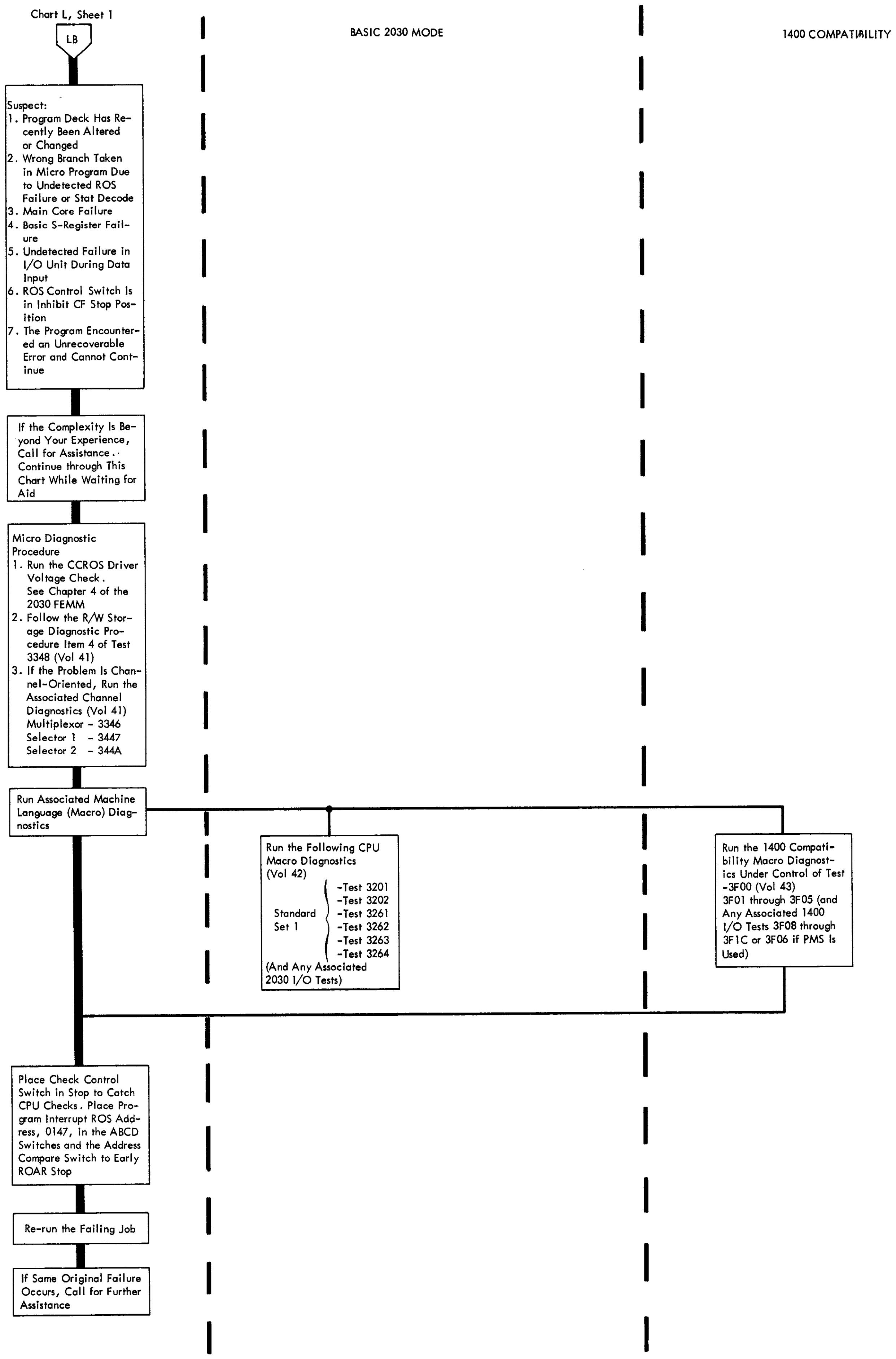




DT Chart J CF Stops and Special One Word Loops (Sheet 2)



DT Chart L Machine Language (Macro Program) Loops (Sheet 1)



Suspect:
1. Program Deck Has Recently Been Altered or Changed
2. Wrong Branch Taken in Micro Program Due to Undetected ROS Failure or Stat Decode
3. Main Core Failure
4. Basic S-Register Failure
5. Undetected Failure in I/O Unit During Data Input
6. ROS Control Switch Is in Inhibit CF Stop Position
7. The Program Encountered an Unrecoverable Error and Cannot Continue

If the Complexity Is Beyond Your Experience, Call for Assistance. Continue through This Chart While Waiting for Aid

Micro Diagnostic Procedure
1. Run the CCROS Driver Voltage Check. See Chapter 4 of the 2030 FEMM
2. Follow the R/W Storage Diagnostic Procedure Item 4 of Test 3348 (Vol 41)
3. If the Problem Is Channel-Oriented, Run the Associated Channel Diagnostics (Vol 41)
   Multiplexor - 3346
   Selector 1 - 3447
   Selector 2 - 344A

Run Associated Machine Language (Macro) Diagnostics

Run the Following CPU Macro Diagnostics (Vol 42)
Standard Set 1
-Test 3201
-Test 3202
-Test 3261
-Test 3262
-Test 3263
-Test 3264
(And Any Associated 2030 I/O Tests)

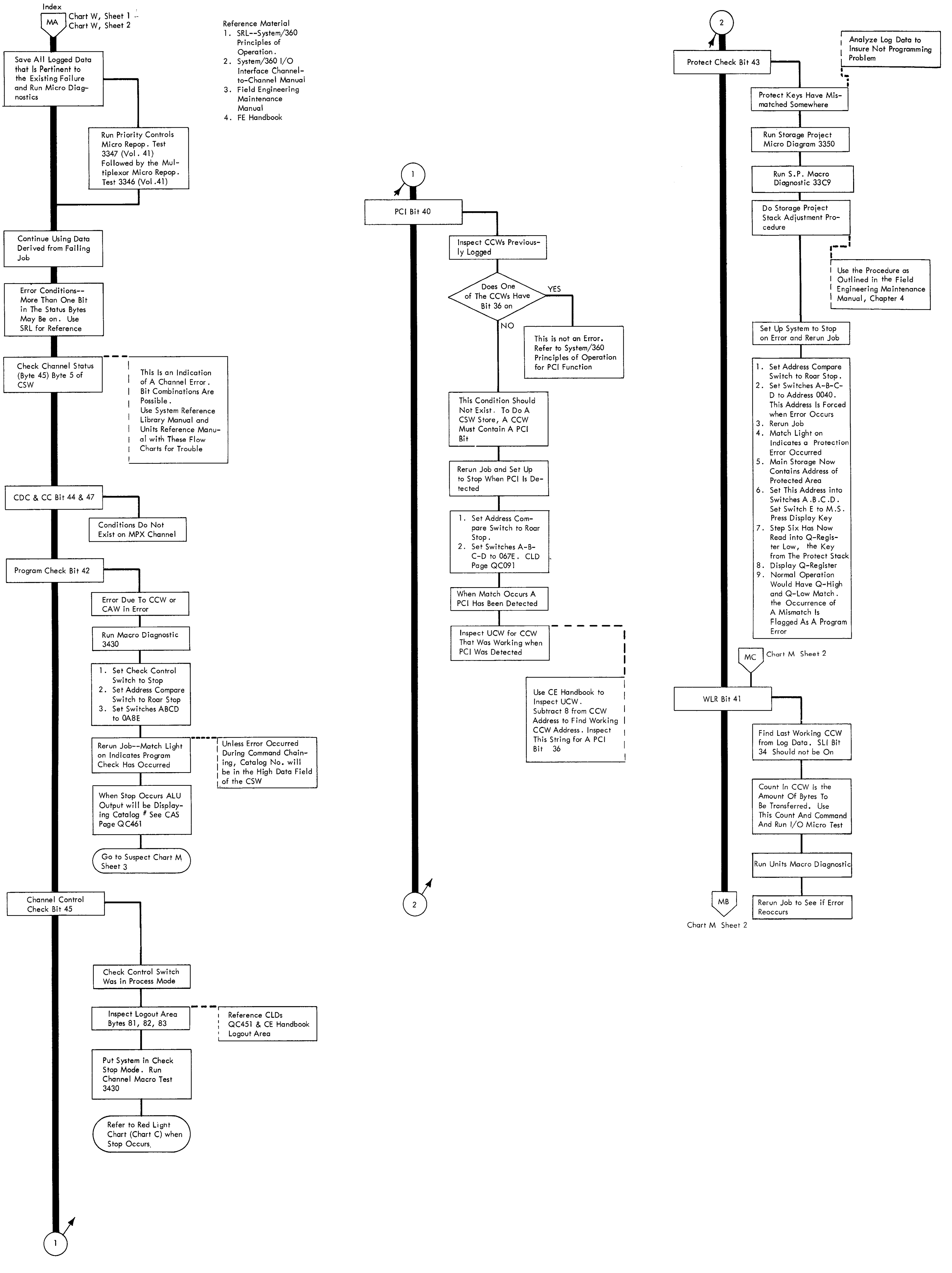
Run the 1400 Compatibility Macro Diagnostics Under Control of Test -3F00 (Vol 43) 3F01 through 3F05 (and Any Associated 1400 I/O Tests 3F08 through 3F1C or 3F06 if PMS Is Used)

Place Check Control Switch in Stop to Catch CPU Checks. Place Program Interrupt ROS Address, 0147, in the ABCD Switches and the Address Compare Switch to Early ROAR Stop

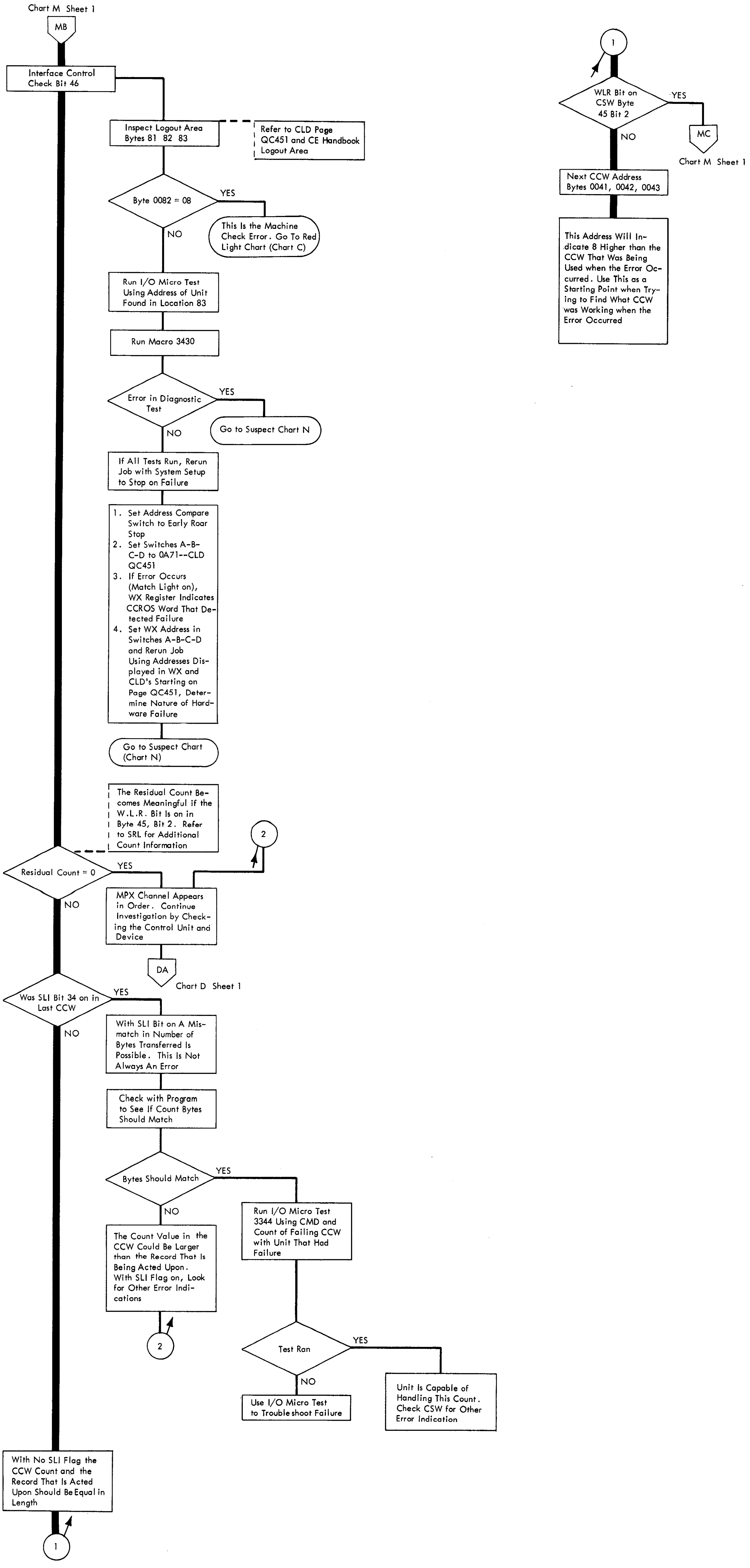
Re-run the Failing Job

If Same Original Failure Occurs, Call for Further Assistance

DT Chart L Machine Language (Macro Program) Loops (Sheet 2)

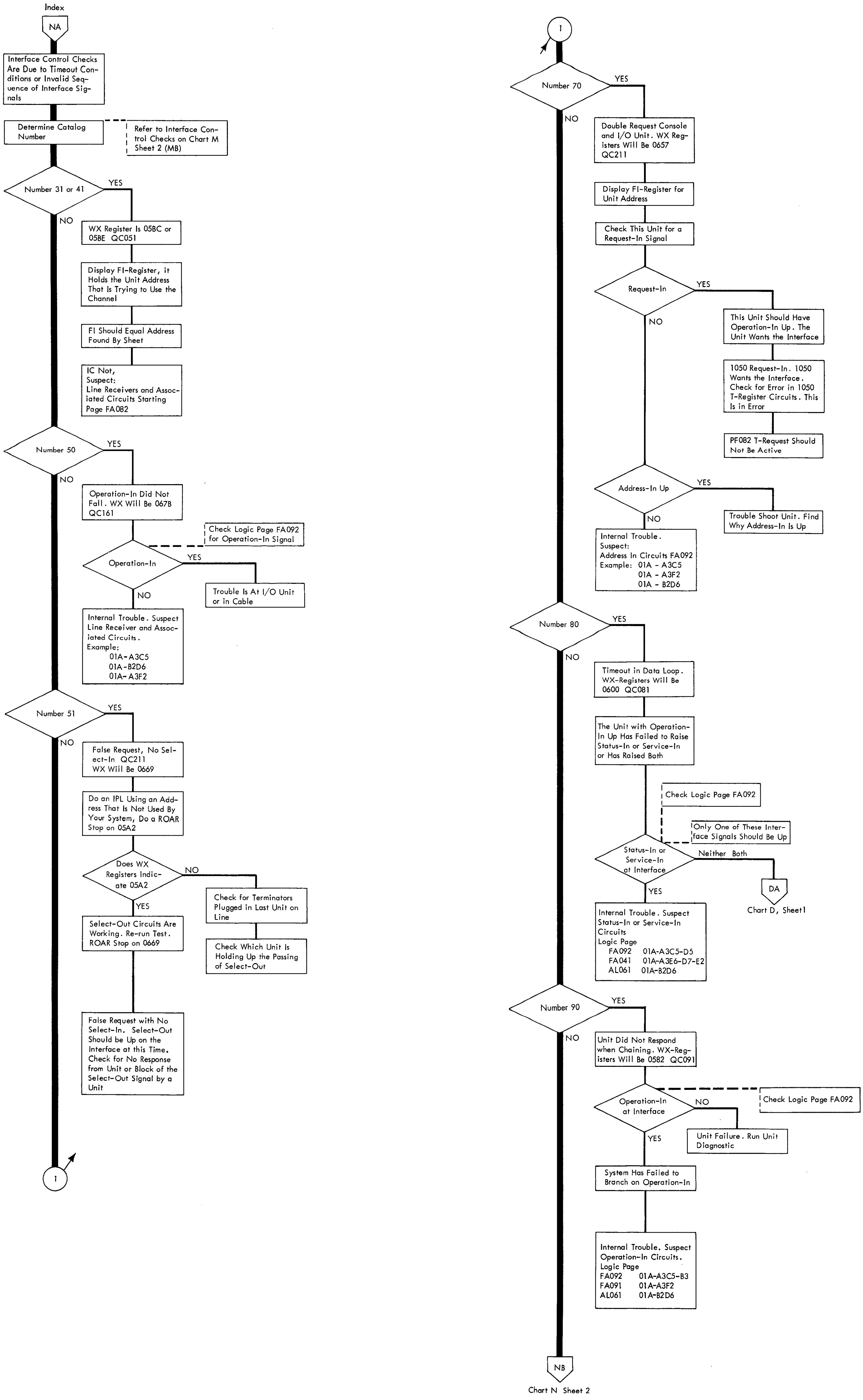


DT Chart M Multiplexor Channel (Sheet 1)

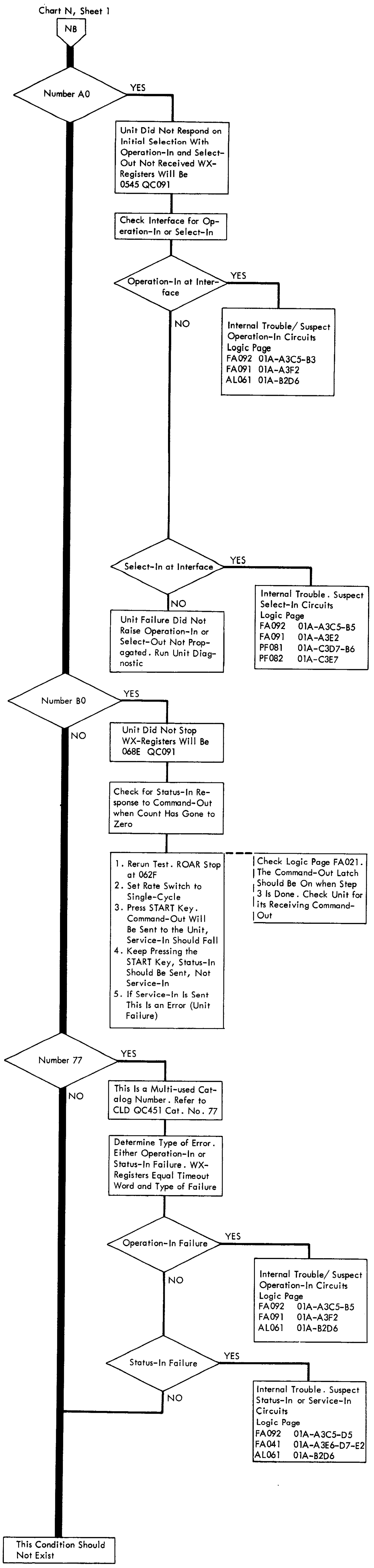


DT Chart M Multiplexor Channel (Sheet 2)

L-Register	CLD Page	Early ROAR Stop at	Do WX Registers Read at	Hints
01	QC021	0A89	052C	1. Zero count in CCW Byte 6 and 7 Check CCW display DL-Registers now holding the count.
02	QC021	0A89	0590	1. Invalid flags in Byte 4 of CCW. Display L-Register. Low-order three bits must be zero.
04	QC021	0A89	0515	1. Initial operation has a TIC or 2. Two TICs in a row. 3. Inspect CCW string for this condition.
05	QC021	0A89	051D	1. Invalid command Byte, check Byte 0 of CCW.
06	QX021	051E	051B	1. Byte 1 of CCW is not zero. Check CCW for this Condition.
07	QC011	058A	0572	1. Protection error in CCW. 2. CCW address too big for memory size. 3. Byte 0049 must be zero for 2030. 4. Check CAW format using CE Handbook or SRL Manual.
08	QC021	0A8B	0517	1. Two successive TICs. I- and J-Registers now have address of CCW that had the second TIC as its command. Inspect CCW string for this condition.
09	QC011	0526	0522	1. The new CCW address designated by a TIC command was not located on a doubleword boundary.
0A	QC061	0A8F	0611/ 0613	1. Memory wrap when CCW was being updated during a chaining operation
0B	QC101	0A8B	069C	1. Zero count detected while data chaining or invalid flag Byte. 2. Check L- and D-Registers for count. 3. Check S-Register for 5 bit being on. If 5 bit is off, an invalid flag Byte was detected. 4. Check CCW for these conditions.
0C	QC101	058B	053C	1. Memory wrap on next CCW address while CDA. 2. I- and J-Registers now should indicate zero.
0D	QC021	0A8F	052D	1. Zero count detected during a command chain. Check L- and D-Registers for count. 2. Check CCW string for these conditions.
0E	QC031	0A8F	052A	1. Invalid flag Byte while command chining. 2. L-Register contains flag Byte. 3. Check CCW string for this condition
1F	QC131	0A8F	06BC	1. Memory wrap detected while command chaining. Next CCW was located outside of memory. I- and J-Registers are 0000. 2. Check CCW string for a CCW located in the last 8 locations in memory. This one is chaining out of memory.
2F	QC131	0A8F		1. Command Byte all zeros detected while command chaining. U-Register now holds the command. 2. I- and J-Registers now contain the failing CCW address plus 1. 3. Check CCW string for this address and the indicated error condition.

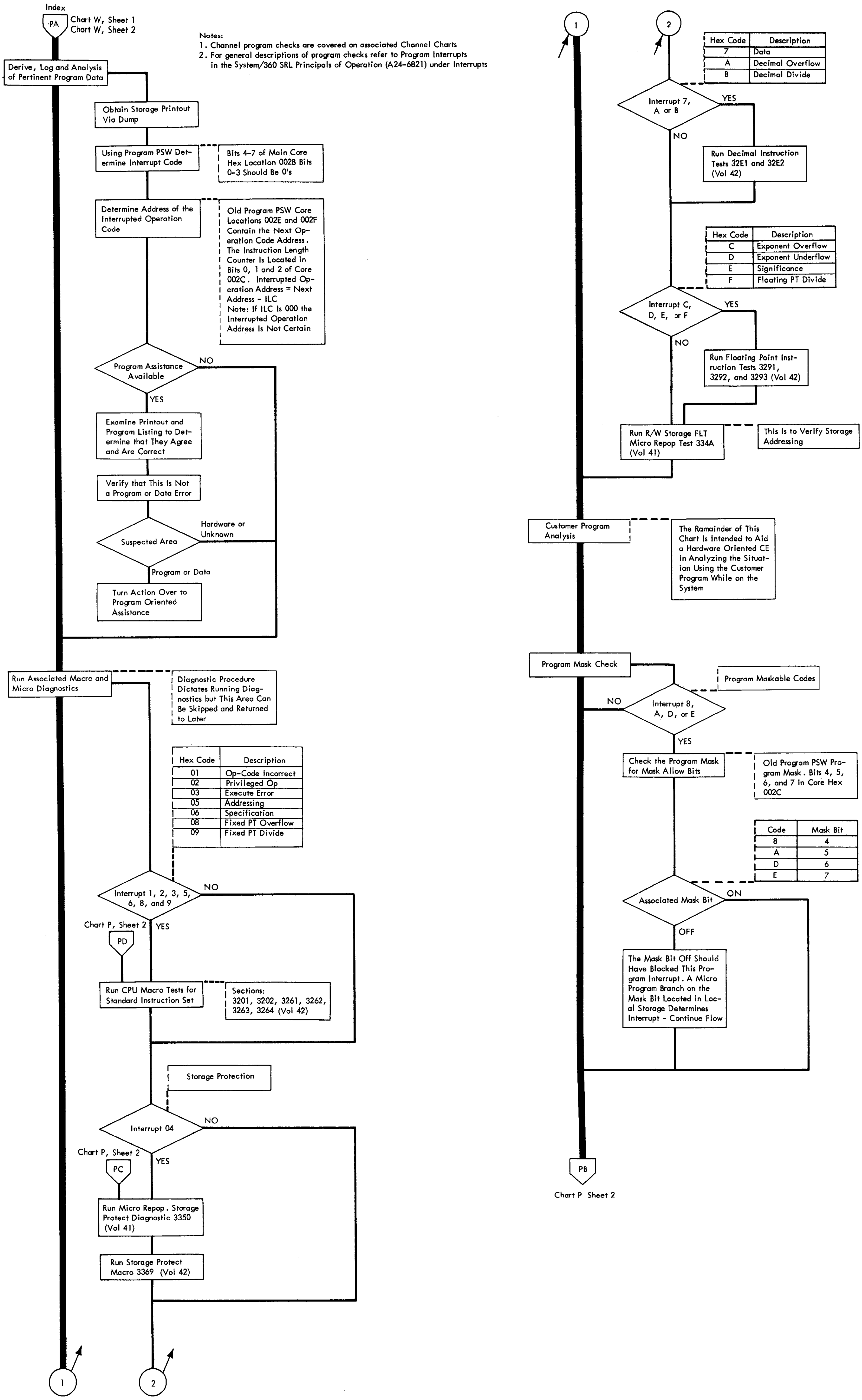


DT Chart N Multiplexor Catalog Numbers (Sheet 1)



DT Chart N Multiplexor Catalog Numbers (Sheet 2)

Notes:
1. Channel program checks are covered on associated Channel Charts
2. For general descriptions of program checks refer to Program Interrupts in the System/360 SRL Principals of Operation (A24-6821) under Interrupts



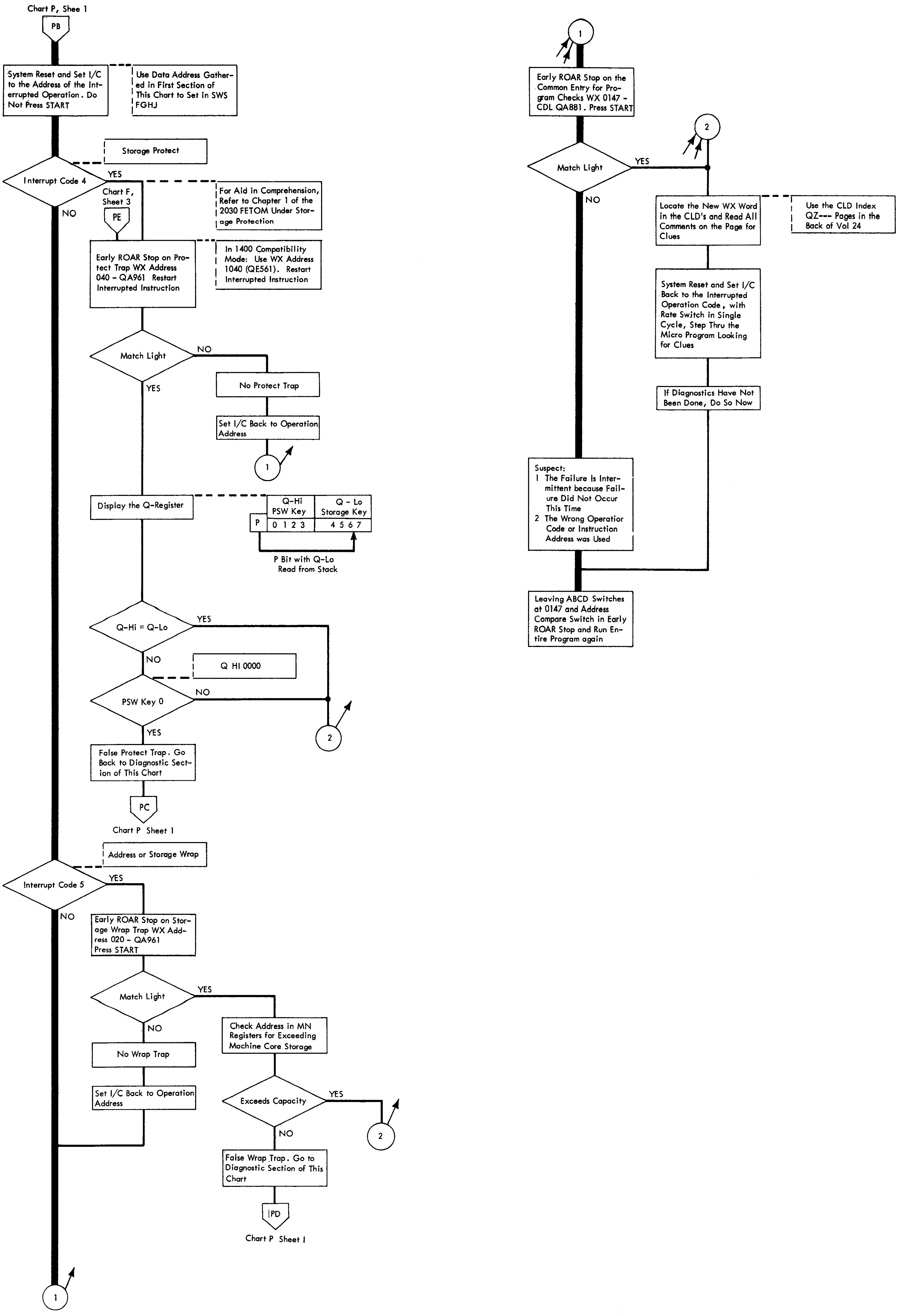
Hex Code	Description
01	Op-Code Incorrect
02	Privileged Op
03	Execute Error
05	Addressing
06	Specification
08	Fixed PT Overflow
09	Fixed PT Divide

Hex Code	Description
7	Data
A	Decimal Overflow
B	Decimal Divide

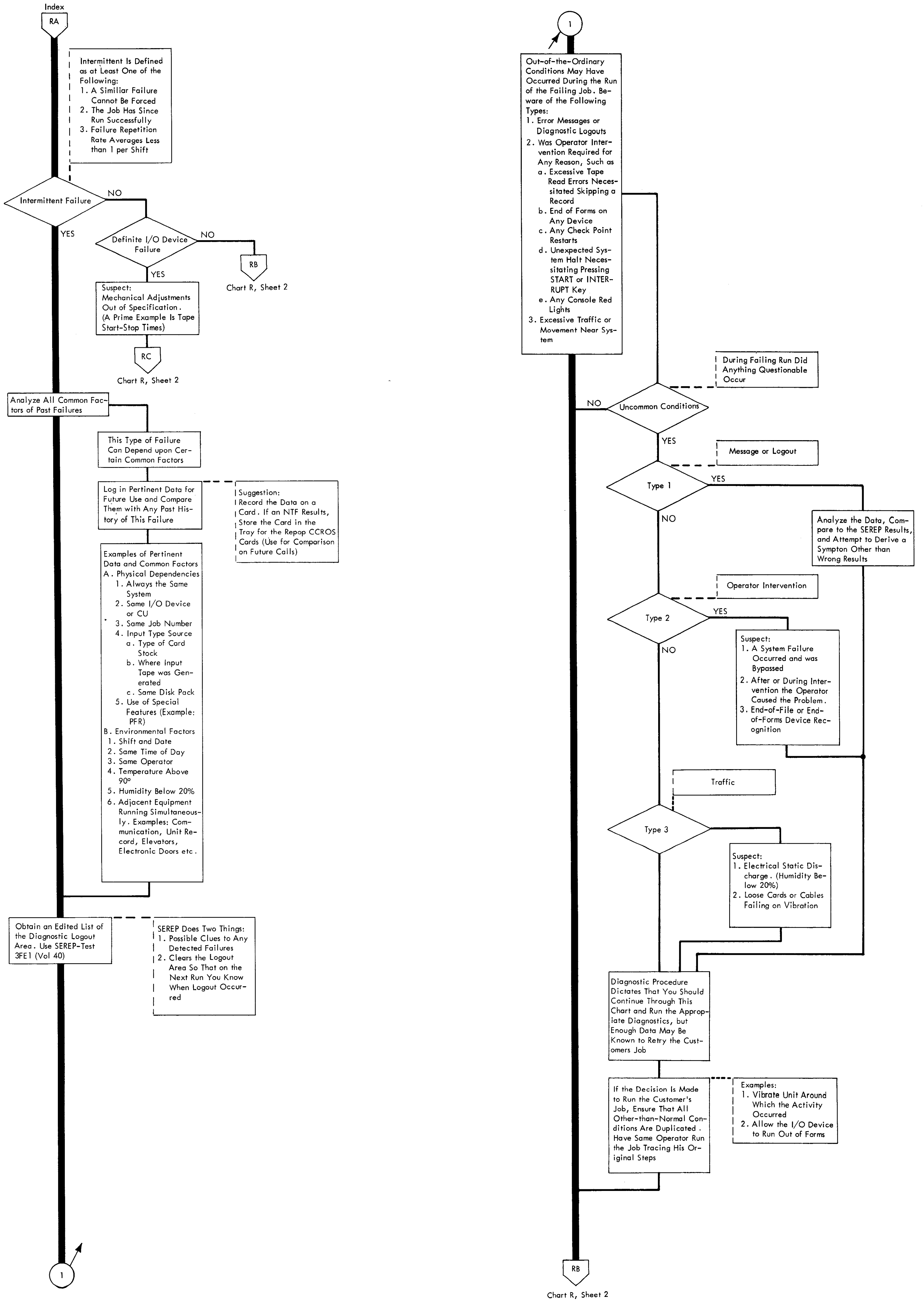
Hex Code	Description
C	Exponent Overflow
D	Exponent Underflow
E	Significance
F	Floating PT Divide

Code	Mask Bit
8	4
A	5
D	6
E	7

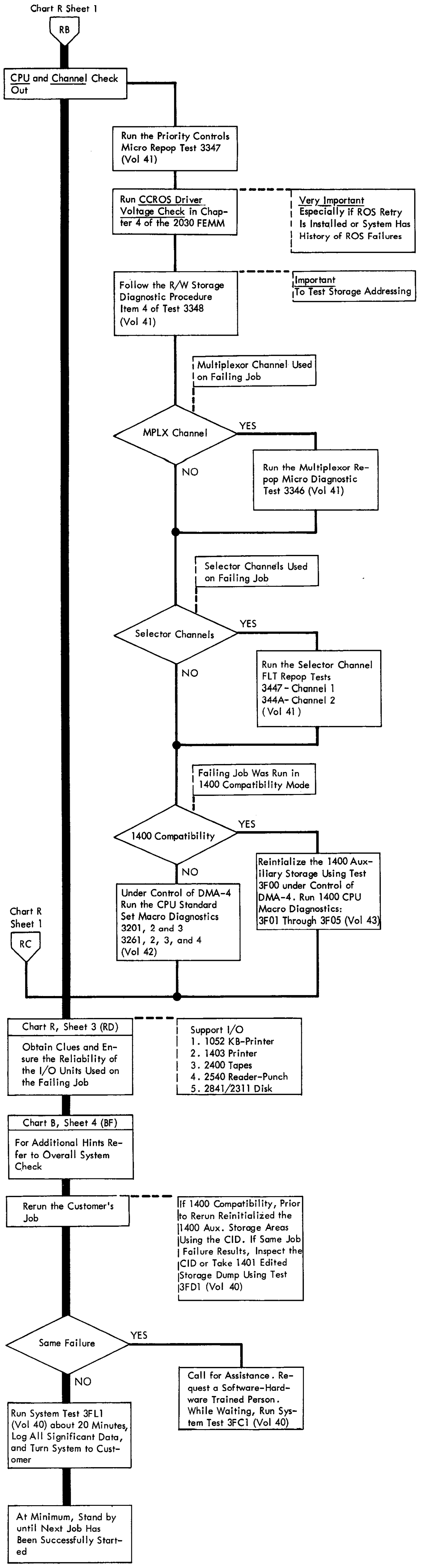
DT Chart P Program Checks (Sheet 1)



DT Chart P Program Checks (Sheet 2)



DT Chart R Missing Records or Wrong Results (Sheet 1)



DT Chart R Missing Records or Wrong Results (Sheet 2)

Chart R, Sheet 2

RD

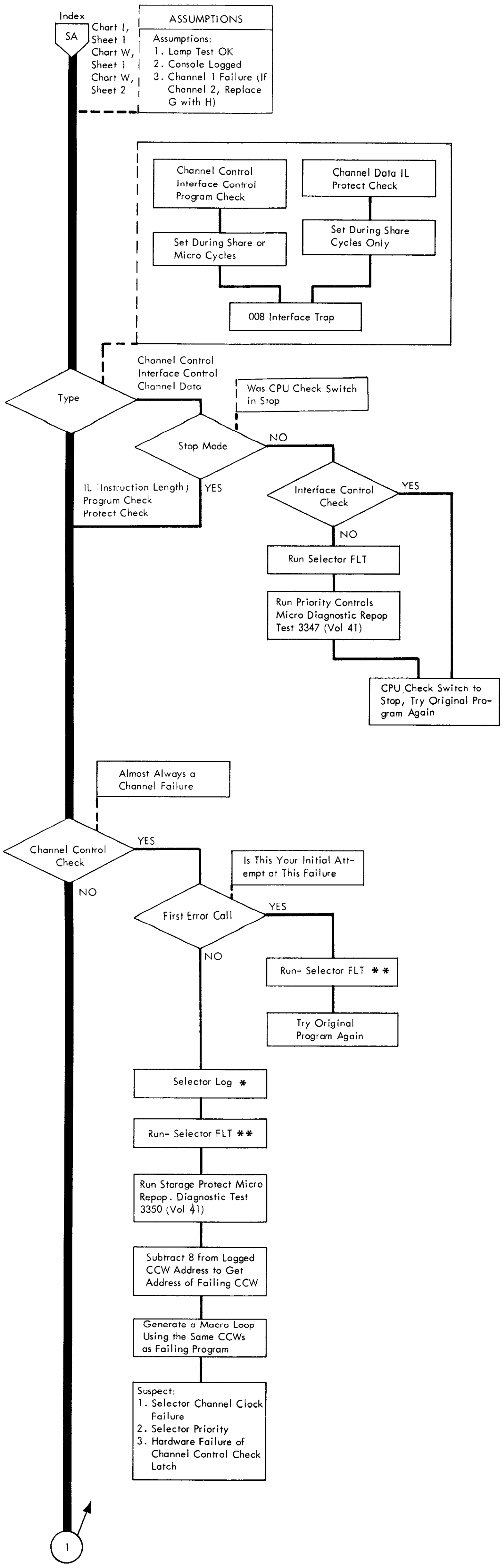
Unit Description	Symptom or Cause	Notes and Hints	Suggested Minimum Quick Check
1052 Key Board	1. Missing characters caused by console attachment failing to read punch-1-clutch signal	Check interface cables in the 1051, the half board 01F-A1, and in the CPU on boards 01A-C2 and C3 (ALD PF 521)	Run the read inquiry macro diagnostic 902 Vol 43
	2. Extra character in storage	Console request-in line may not be resetting properly (ALD PF021)	
	3. Wrong data in storage; good data on 1052 printer	Check PT & T to EBDIC Translate starting at PF 291 through PF 331	
	4. Wrong data on printer; good data in storage	Definite 1051 or 1052 printer failure	
1052 Printer	1. Missing characters--caused by 1051 home loop timer running slower than the console attachment clock--or console attachment clock faster than 1051 timer	1. Check attachment clock timing; refer to note on ALD PF011 2. Check 1051 home loop single shots in 1051 manual page 131TM	Run the basic write function macro diagnostic 900 Vol 43
	2. Only one character of a message prints. This may also be the wrong first character	Posssible that 1050-Request-in line not being reset (ALD PF071)	
	3. 1050 intervention required would cause no message to be accepted by 1052 printer (indicator on lower right of console under CPU status)	1. Is printer paper under interlock 2. Is CPU switch off 3. Check EPO switch inside back cover of 1051	
	4. Incorrect data	1. Turn the 1052 CPU switch off. Type A through Z. If incorrect character prints, 1051 or 1052 is at fault. 2. On-line check the EBCDIC to PT & T Translate on ALD PF 191 through 222	
1403 Printer	1. Wrong chain on 1403 or wrong UCB image	Check job for proper chain and UCB deck	Run printer macro diagnostics 1. F 832 (UCS printers) (2821 Vol 8) 2. F 836 ripple print (2821 Vol 8)
	2. Hammer check circuit malfunction not catching extra or missing hammer fire, or failure to receive data check on unprintable UCS character	Diagnostic F832 will check this	
	3. Interface or 2821 failure causing data record to be cut short		
	4. Multiple pick or drop of bits in UCB	Reinitialize UCB	
	5. Multiple pick or drop of bits in print buffer		
	6. UCB parity-check circuit failure		
	7. Print buffer parity-check circuit failure		
	8. Sync-check error circuit failure	To ensure back in sync, open and close T-casting or initialize the UCB.	
	9. End-of-forms indication malfunction	Test off-line	
	10. Failure to recognize data checks could be caused by the BLOCK DATA CHECK LATCH or the FOLDING LATCH in 2821 (in UCB only)	Diagnostic F832 will check this	
	11. Missing slug on chain or broken hammer	One character or one print position	
	12. 1401 compatibility missing last 32 positions of print line	Missing bit 0 in COMP AUX storage-B of byte 8C. This bit is reset on 8K machines if operator presses the load key with 1402 in the F, G, H, and J switches; then does system reset, ROAR reset, and start	
2400 Tapes	1. Drive start-stop time maladjustment	Most probable cause	Run the following macro diagnostics to check tape start-stop times: 16K systems --F521 and F522 8K Systems--3523 and 3524 only All systems--for Diagnostic Check F510 All tests are in MDP Tape/TCU (Vol 3)
	2. Check to see if the forward stop delay noise trigger ever comes on. Records can be skipped if it does	This is usually a result of maladjusted drive start-stop times	
	3. Tape being used was generated bad at a previous date, possibly on another system	Check tape source, and the age and condition of the tape	
	4. Detected hard errors not being handled	Run job with error hold plugged on TAU	
	5. Failures in special recognition circuits and command codes such as single tape mark records, noise bits, EOF, erase, backspace and forward space commands		
	6. 1400-compatibility mode: check the operation of the GMWM latch on CPU ALD MB 022		
	7. Failures with data convert, translate, and densities, either hardware or program failure		

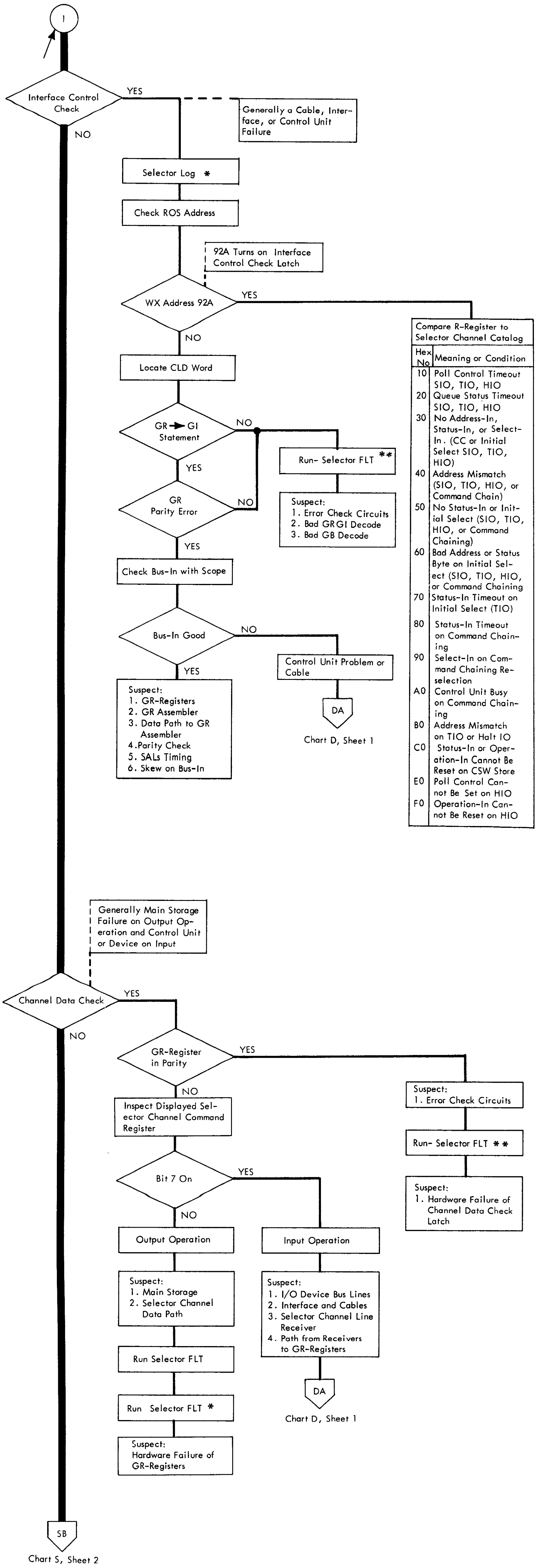
Unit Description	Symptom or Cause	Notes and Hints	Suggested Minimum Quick Check
2540 Reader	1. Two cards feeding piggy-back through reader	Check reader throat adjustment; if near .020", this is a possibility	Run macro diagnostics 1. F821 routine 1--(punch 27 cards) 2821 Vol 8 2. F811 routine 1--(read same 27 cards) 2821 Vol 8
	2. Interface or 2821 failure causing data record to be cut short		
	3. Any error check circuit not functioning would fail to catch input errors	Running cards with one set of brushes removed will check one possibility	
	4. Multiple pick or drop in reader buffer		
	5. Cards way off registration	Very remote	
2540 Punch	1. Failure to give dummy write to punch at end of job allows one card to be left in punch. This card is not checked when nonprocess runout key is pressed	Check program for proper punch end procedure	Same as 2540 Reader
	2. Using EOF key on other than PFR can cause a missing record when cards are allowed to run out, new ones placed in hopper, and job continued	Use EOF on PFR operations only	
	3. Punch check circuit failure in conjunction with valid punch check	1. Lift punch brushes, check for errors when punched cards are fed 2. Check the PCH BRUSH CL DELAY INT signal	
	4. See No. 2 on reader		
	5. Punch buffer pick or drop of even bits		
2841/2311 Files	1. Intermittently missing data or receiving no-record-found	Head alignment problems can cause this. But if adjustment is made beware of causing new problems while trying to read packs written before adjustment	1. Run the 2841 nonresident micro test--read/write 100 address marks. 2. Run the 2841 macro test 602 (MDP Vol A02) 3. Run the 2841 interchangability macro test 613 (MDP Vol A01)
	2. Missing records resulting in no-record-found	1. Have customer temporarily add a patch to his program to do a read-back check after every file-write command 2. Suggested programming procedure after no-record-found a. Read home address to ensure correct cylinder b. If not correct cylinder, a recalibrate command should be issued c. If correct cylinder, retry the original record at least 10 times Note: STEP C may have to be patched in some operating systems 3. Address marker detection circuit failure	
	3. Missing or short-length records with no indications	1. Run 2841 in check-stop mode to ensure that all detected errors are caught and handled 2. Address marker detection circuit failure and the program is not checking record identification 3. 2841 metering-in line is not forcing CPU metering-out when the CPU is stopped and the file command is still being executed. Check -0 channel to meter line on ALD KU011	
	4. 1400-compatibility count fields and data fields located on the wrong cylinder.	The compare-disable bit 1400 AUX storage B UCW (XX10) bit 0 was possibly on when data records were written. This bit should be on only during the original 1400 formatting of the disk	

DT Chart R Missing Records or Wrong Results (Sheet 3)

SELECTOR CHANNEL

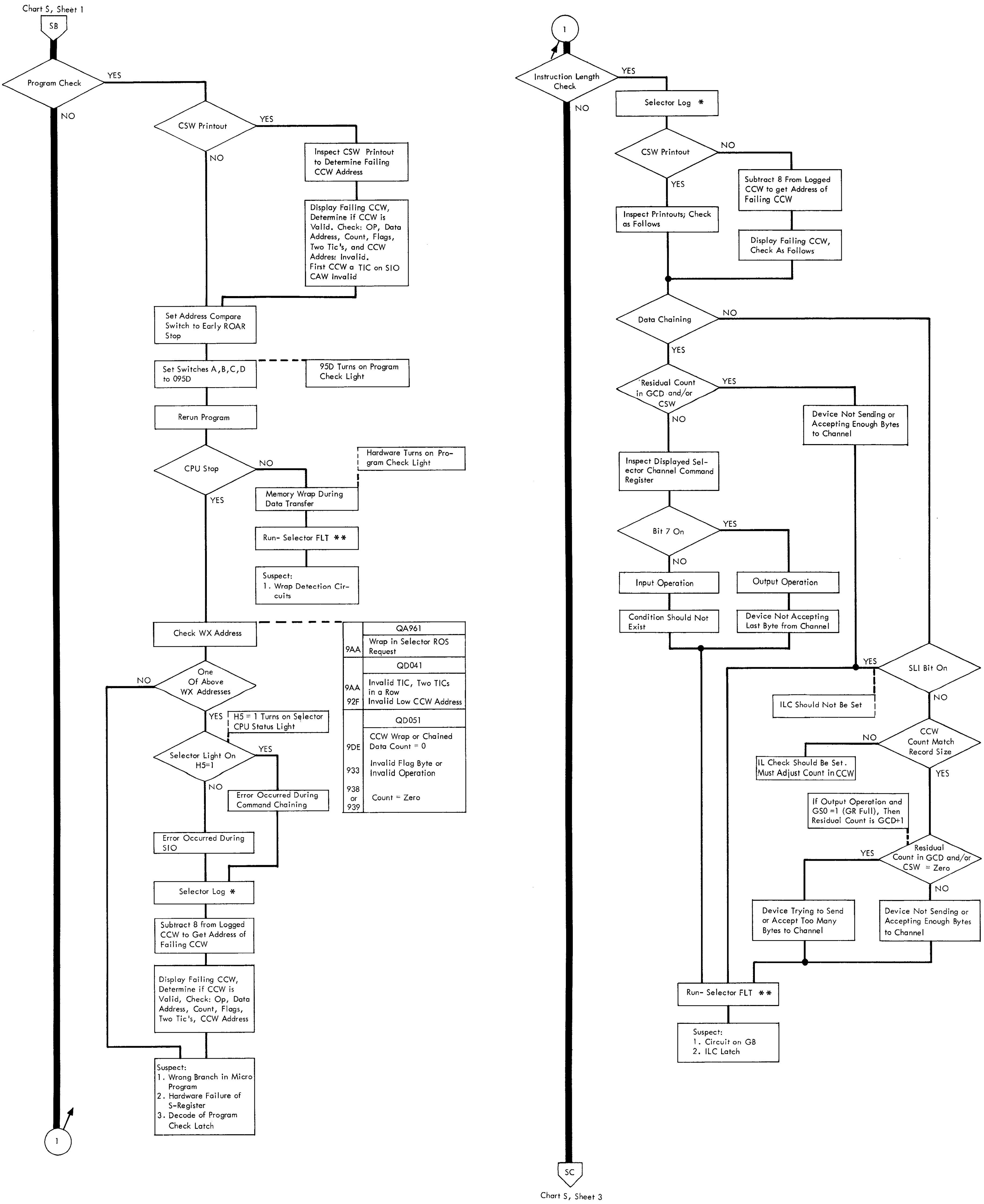
* SELECTOR LOG DATA	** SELECTOR CHANNEL FLT
1. GUV - Data Address	Channel 1 - 3447 Vol. 41
2. GCD - Count	Channel 2 - 344A Vol. 41
3. GS - Status, Control, Error Conditions	
4. GT - Inbound Tags, Control Information	
5. Next CCW Address from Local Storage. Channel 1 - 8E - 8F Channel 2 - AE - AF	



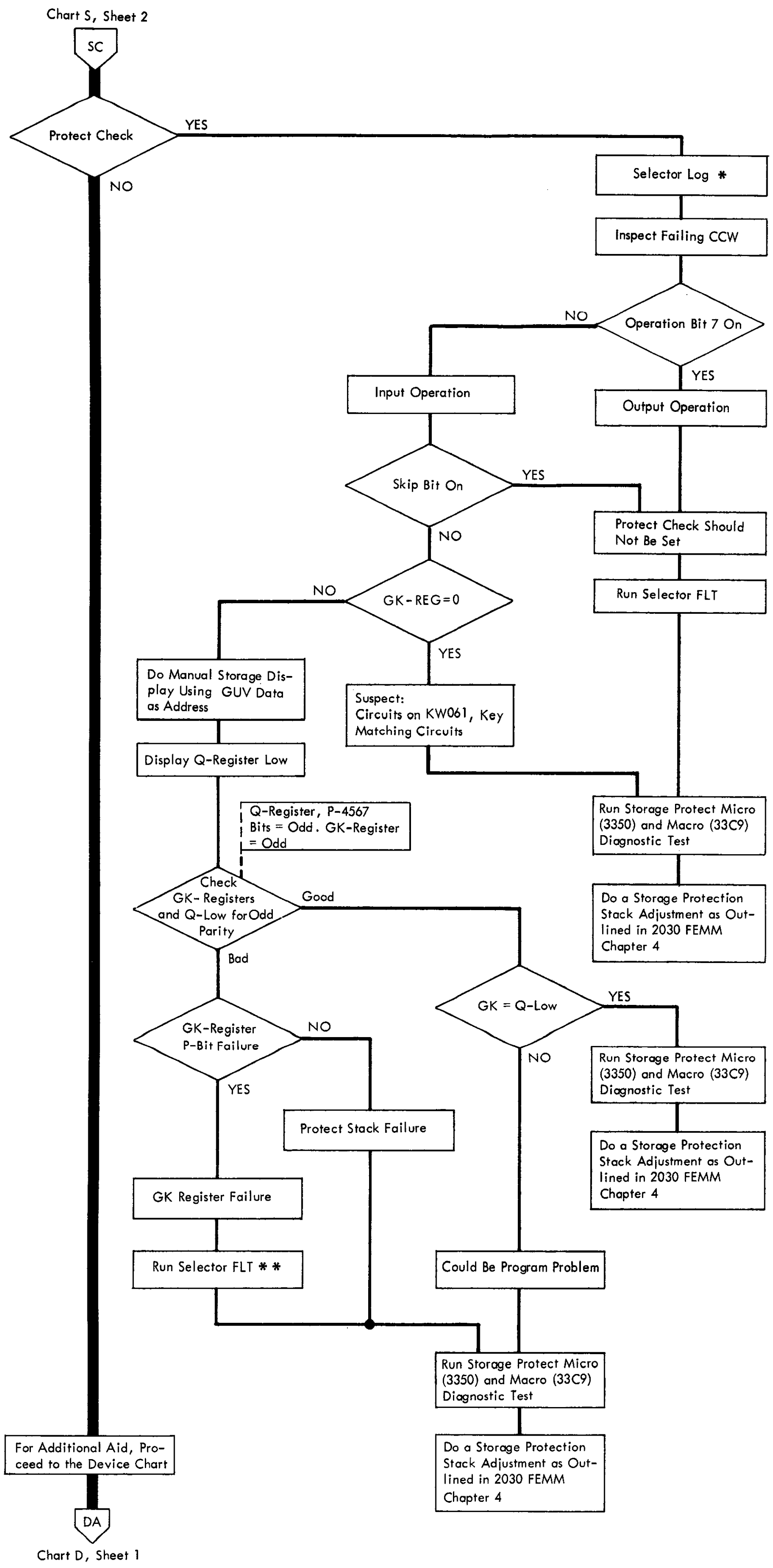


Compare R-Register to Selector Channel Catalog

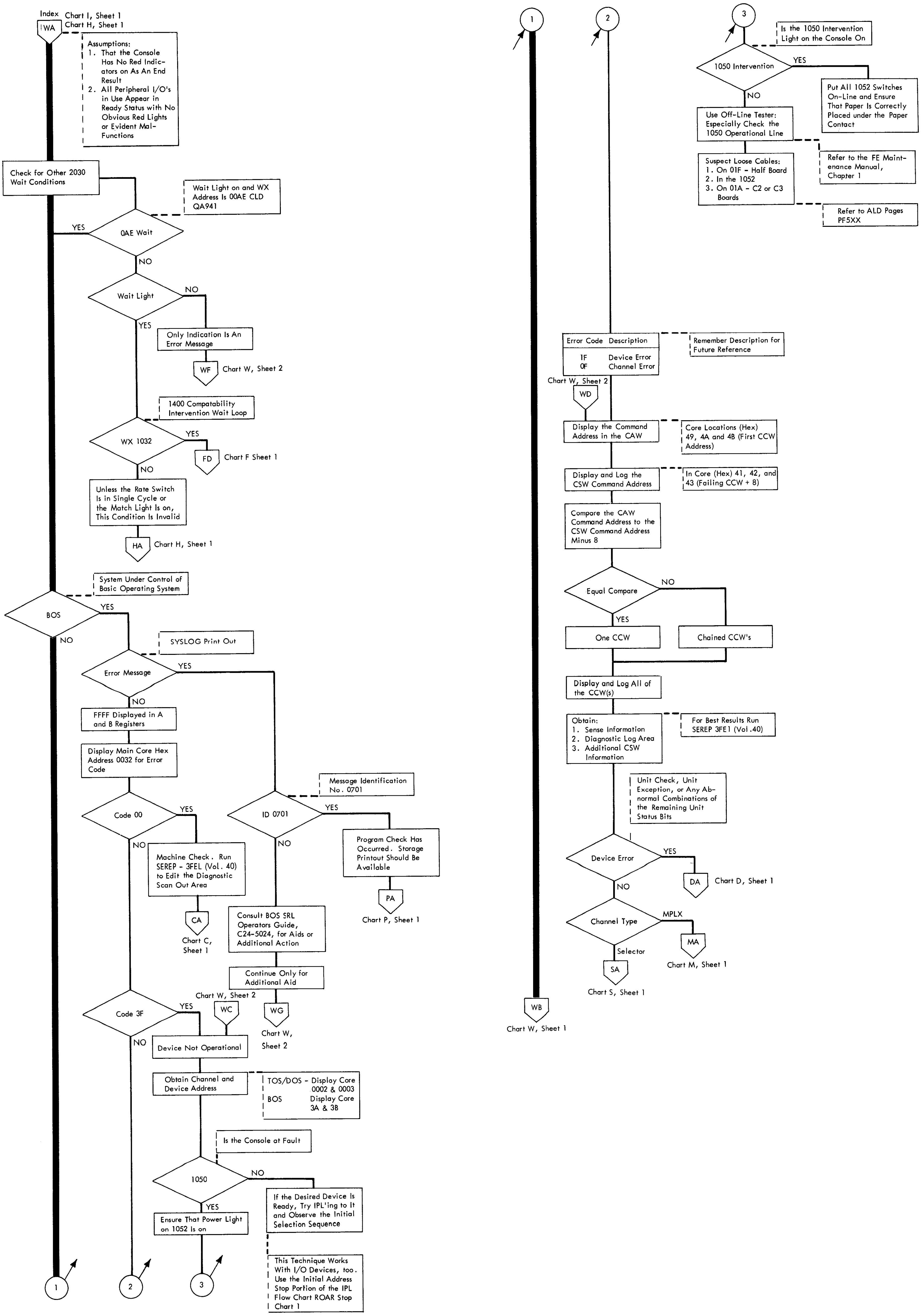
Hex No	Meaning or Condition
10	Poll Control Timeout SIO, TIO, HIO
20	Queue Status Timeout SIO, TIO, HIO
30	No Address-In, Status-In, or Select-In. (CC or Initial Select SIO, TIO, HIO)
40	Address Mismatch (SIO, TIO, HIO, or Command Chain)
50	No Status-In or Initial Select (SIO, TIO, HIO, or Command Chaining)
60	Bad Address or Status Byte on Initial Select (SIO, TIO, HIO, or Command Chaining
70	Status-In Timeout on Initial Select (TIO)
80	Status-In Timeout on Command Chaining
90	Select-In on Command Chaining Reselection
A0	Control Unit Busy on Command Chaining
B0	Address Mismatch on TIO or Halt IO
C0	Status-In or Operation-In Cannot Be Reset on CSW Store
E0	Poll Control Cannot Be Set on HIO
F0	Operation-In Cannot Be Reset on HIO



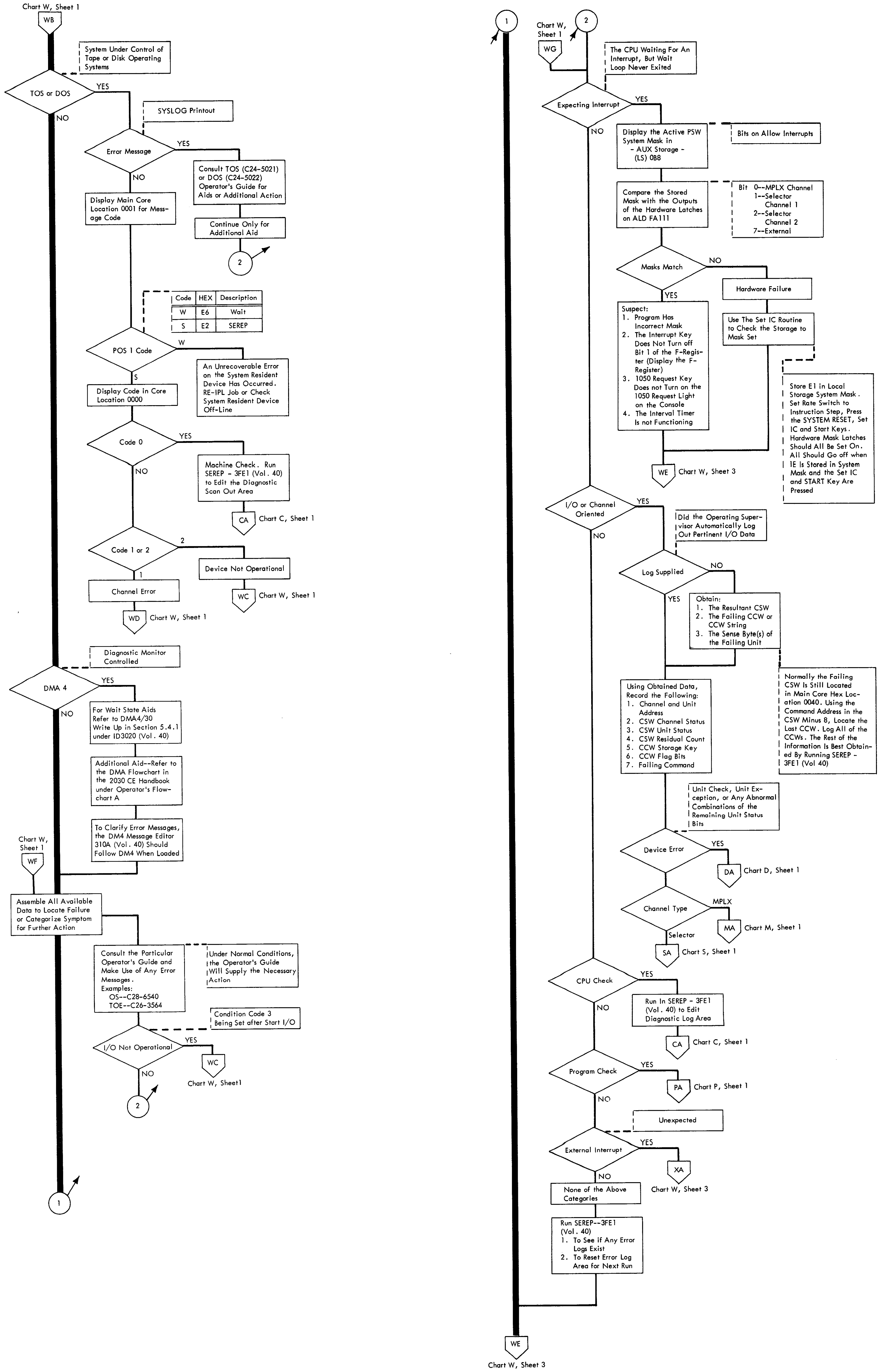
DT Chart S Selector Channel (Sheet 2)



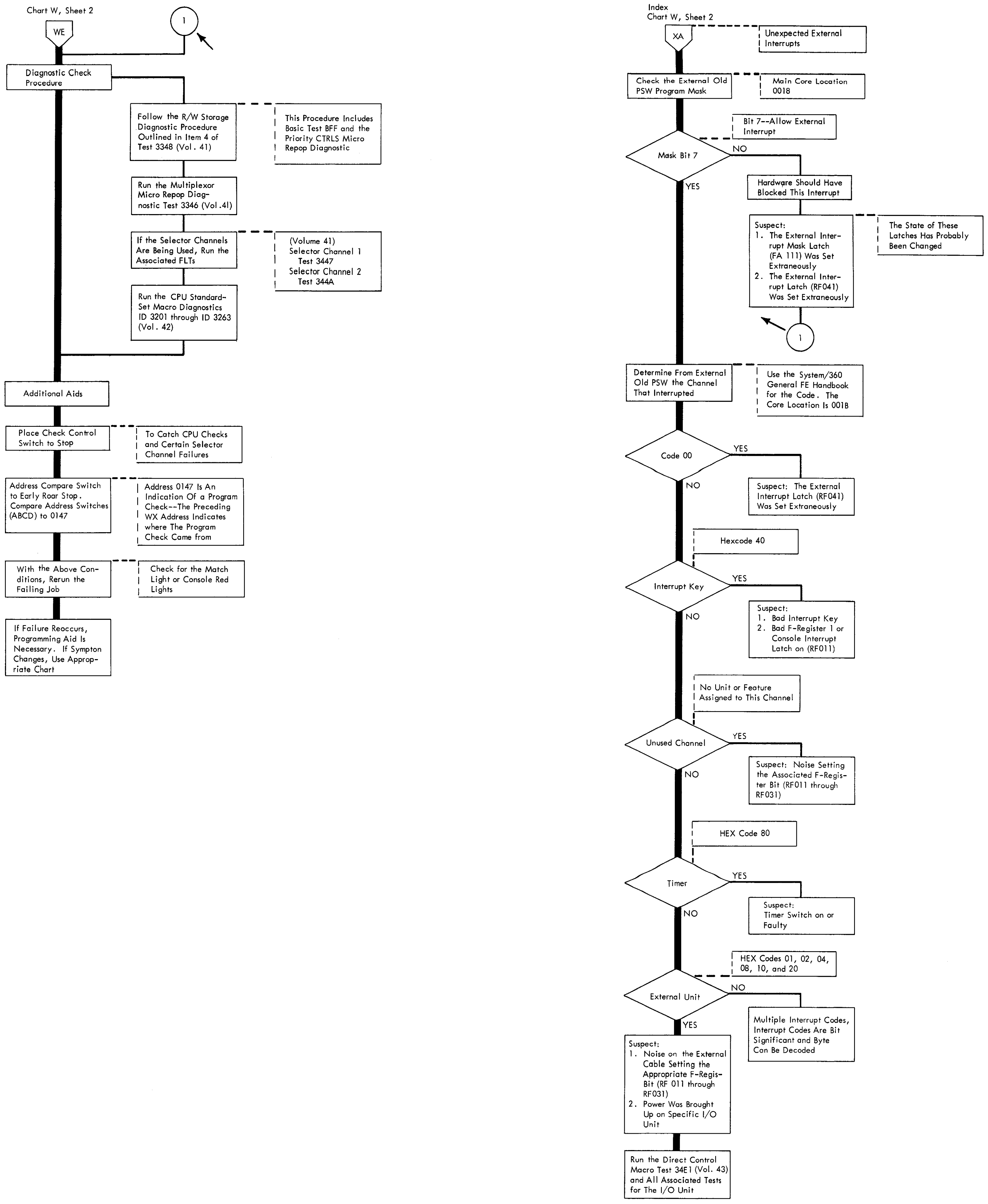
DT Chart S Selector Channel (Sheet 3)



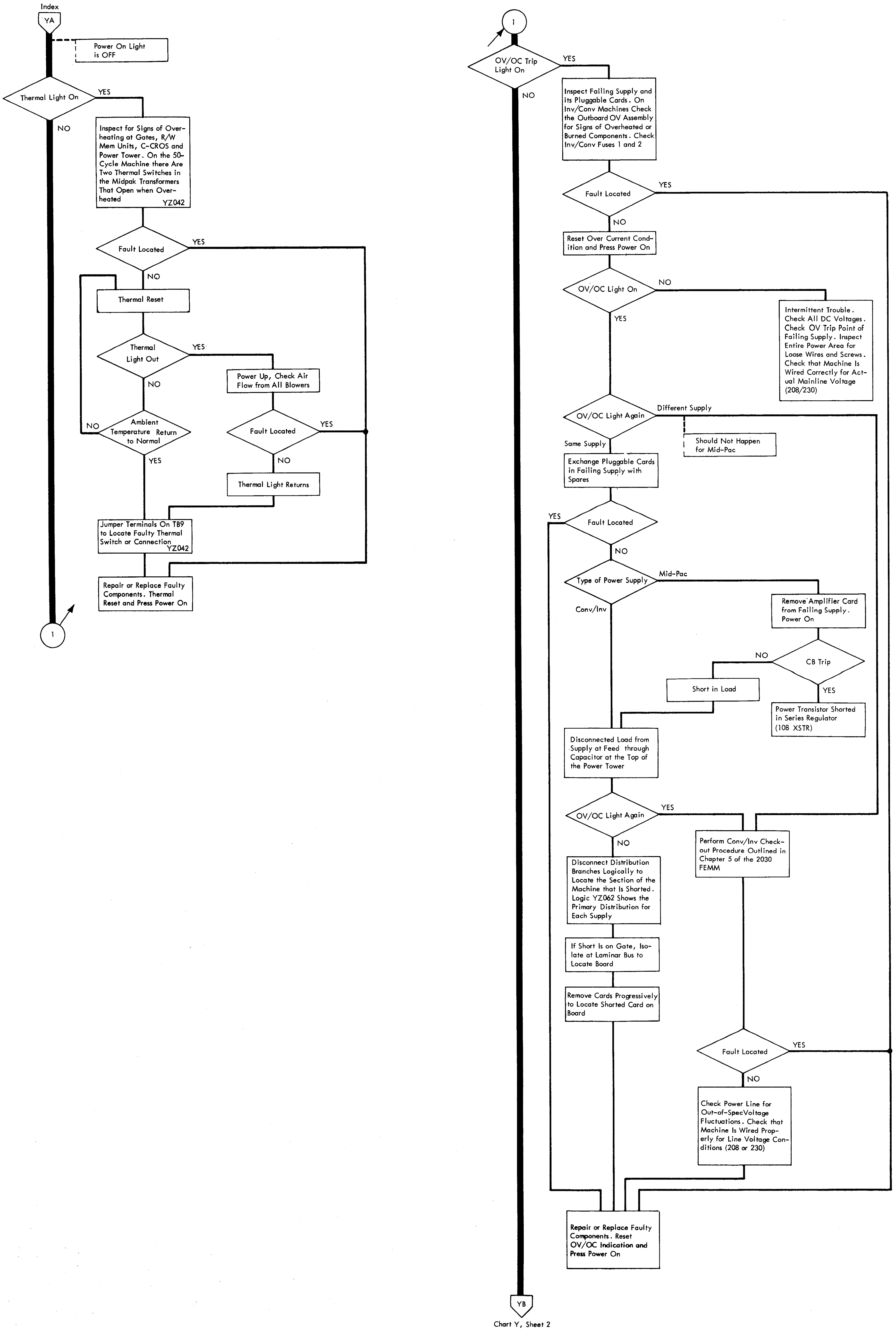
DT Chart WX Wait and/or Error Messages, Unexpected External Interrupt (Sheet 1)



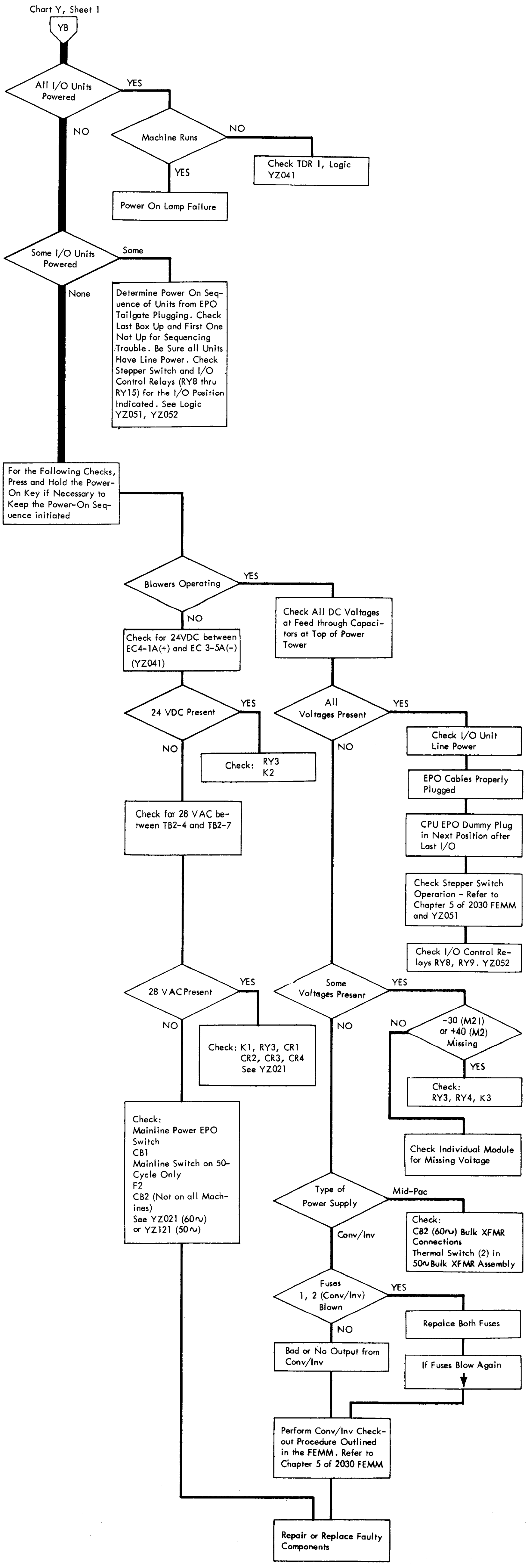
DT Chart WX Wait and/or Error Messages, Unexpected External Interrupt (Sheet 2)

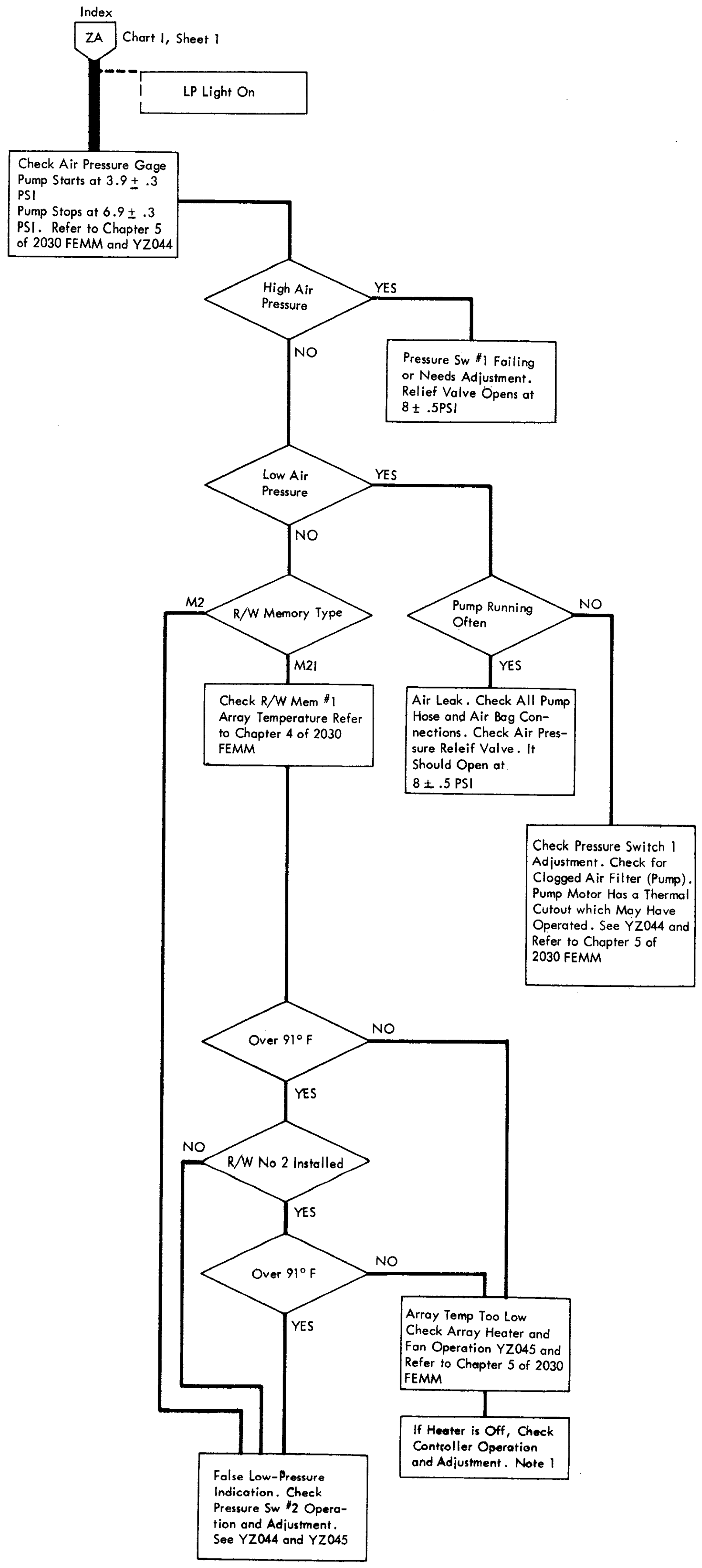


DT Chart WX Wait and/or Error Messages, Unexpected External Interrupt (Sheet 3)



DT Chart YZ Power and LP Light (Sheet 1)

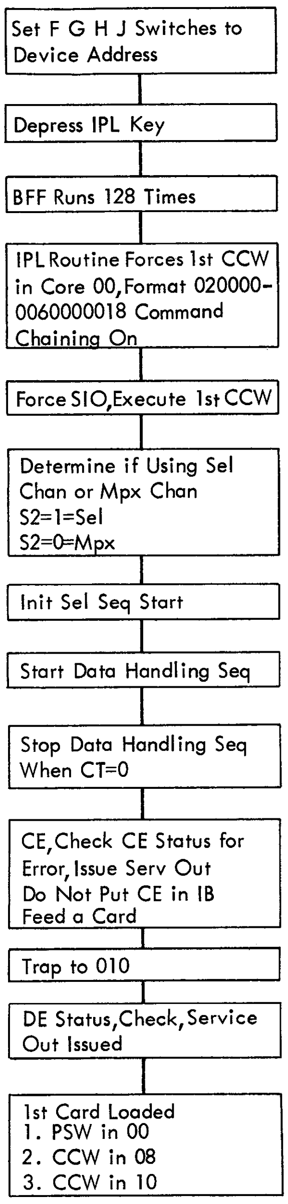




DT Chart YZ Power and LP Light (Sheet 2)

IPL THREE CARD HEX LOADER - MPX CHANNEL BURST MODE

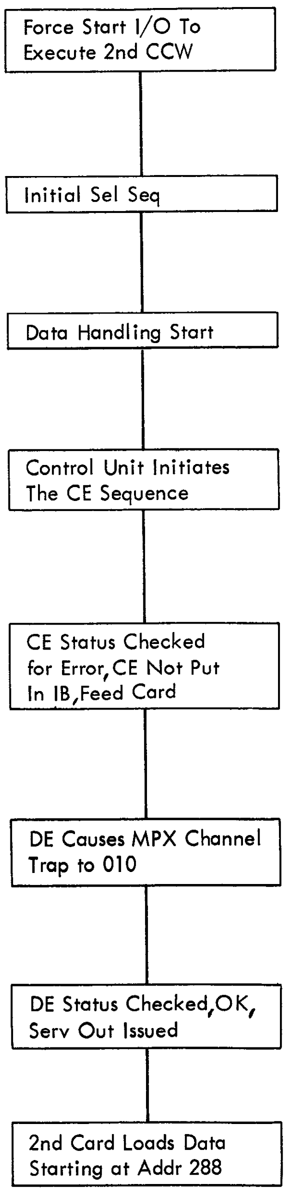
OVERALL SEQUENCE FOR 1st CARD



DETAILED SEQUENCE FOR 1st CARD

INFORMATION	ROS ADDRESS	LOGIC PAGE
MPX or SEL Channel S2=1=Sel S2=0=Mpx	588	QC011
Addr Out-Supp Out	5AO	QC051
Addr Out-Sel Out Op In	59D	QC051
Op In-Addr In	5A8	QC051
Stat In-Op In	5B0	QC051
Enter Data Loop	6C1	QC231
Op In-Serv In Count=0	62F	QC081
Op In-Stat In FI=CE 00001000 Feed Card	61D	QC081
Supp Out-Stat In-Op In FI=CE Issue Serv Out	638	QC121
Supp Out	67A	QC161
Trap to 010 Caused by DE Int	010	QC201
Check for Stat In	6C1	QC231
Serv Out Issued to Stat In	6B1	QC131
Start Getting Next CCW	6B9	QC131
Finish with CCW Ready to Execute	5E9	QC031

OVERALL SEQUENCE FOR DATA FLOW 2nd CARD

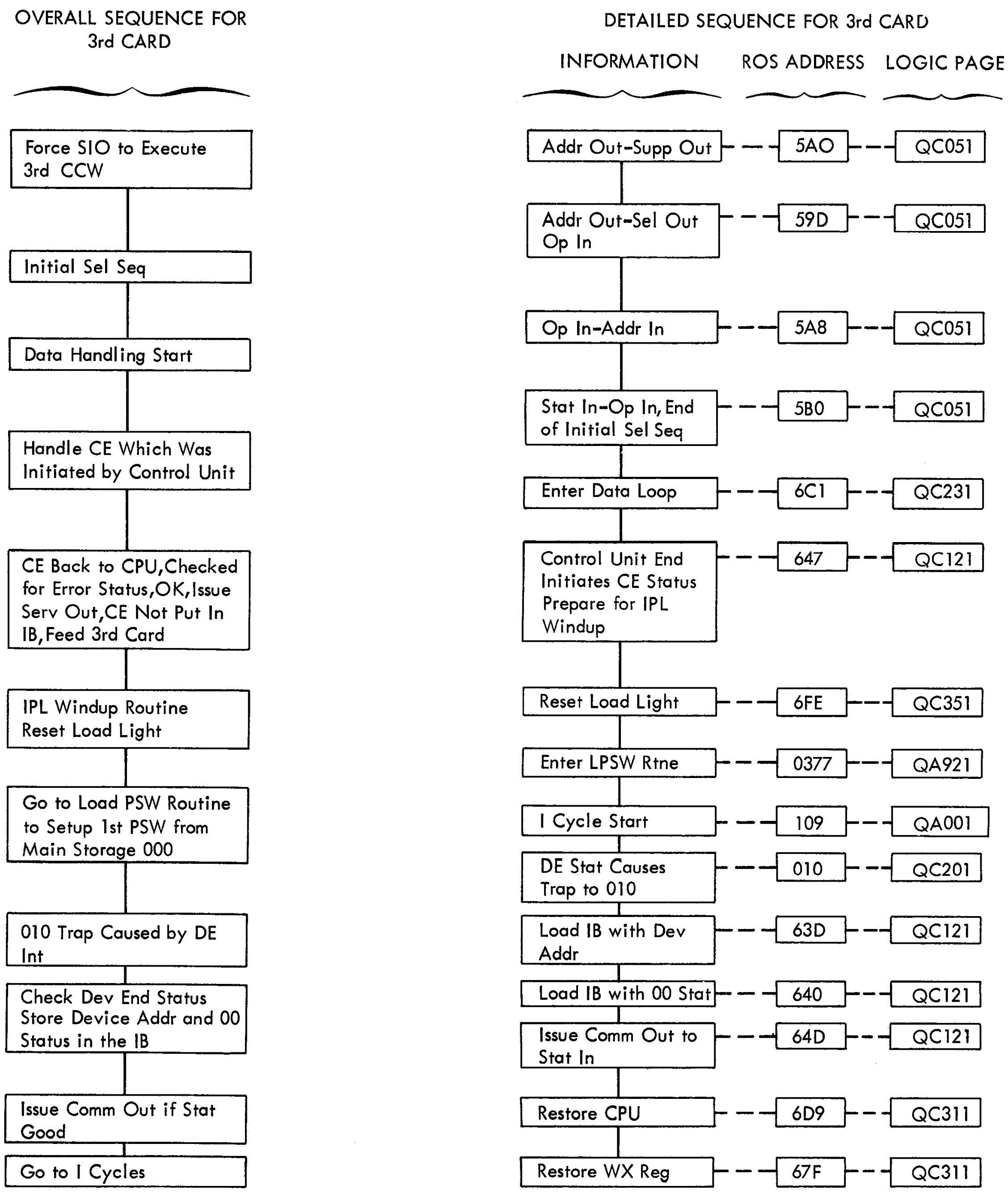


DETAILED SEQUENCE FOR 2nd CARD

INFORMATION	ROS ADDRESS	LOGIC PAGE
Addr Out-Supp Out	5AO	QC051
Addr Out-Sel Out Op In	59D	QC051
Op In-Addr In	5A8	QC051
Stat In-Op In End of Initial Sel Seq	5B0	QC051
Enter Data Loop	6C1	QC231
Cont Unit Ends Data Handling	6A7	QC111
Supp Out-Stat In Op In FI=CE Issue Serv Out	638	QC121
Supp Out	67A	QC161
Trap to 010 Caused by DE Int	010	QC201
Check for Stat In	6C1	QC231
Serv Out Issued	6B1	QC131
Start Getting Next CCW	6B9	QC131
Finish with CCW Ready to Execute	5E9	QC031

IPL Three Card Hex Loader--MPX Channel Burst Mode RS Chart 1 (Sheet 1)

IPL THREE CARD HEX LOADER - MPX CHANNEL BURST MODE



OVERALL SEQUENCE FOR 3rd CARD

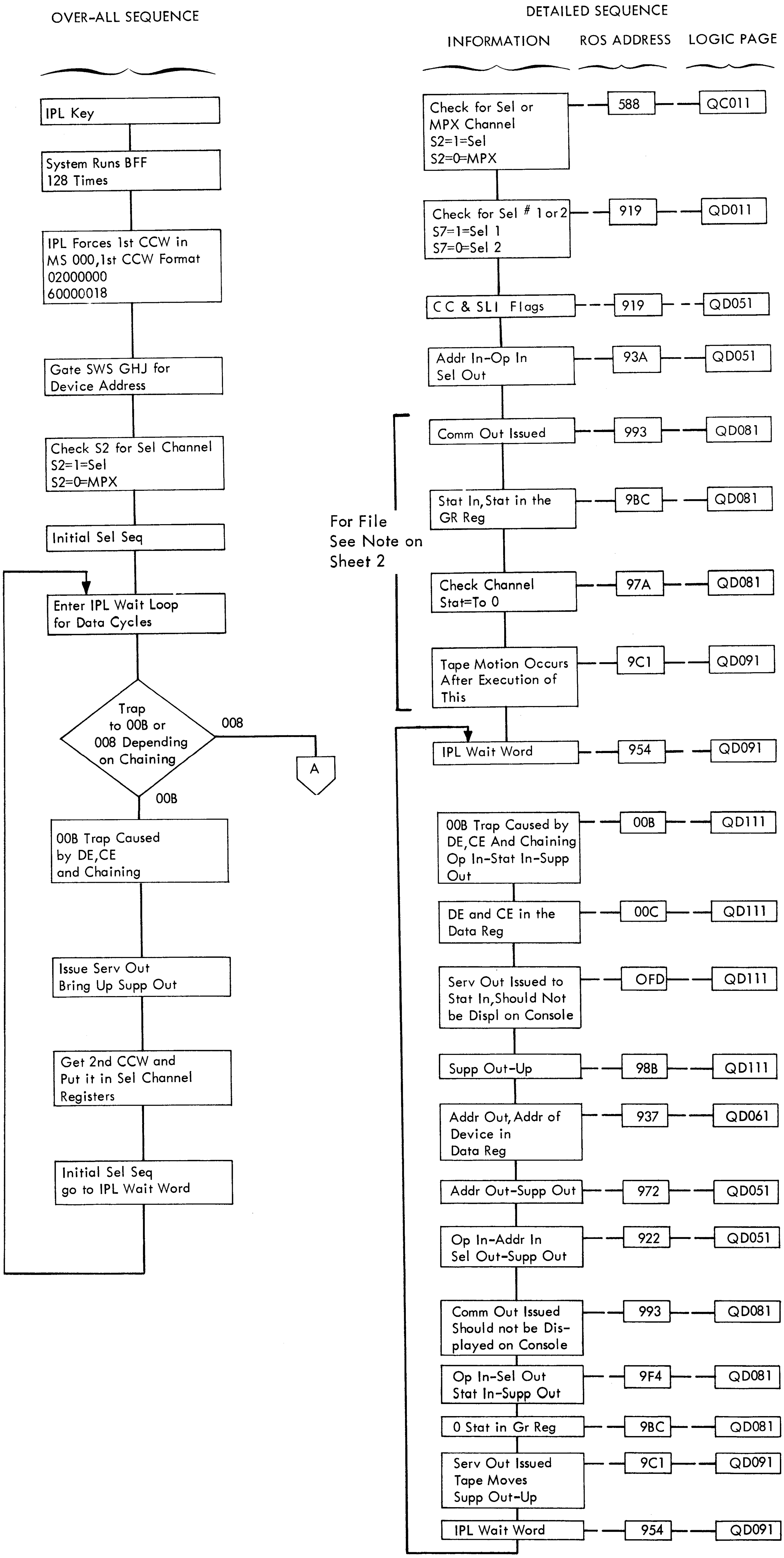
Force SIO to Execute 3rd CCW
Initial Sel Seq
Data Handling Start
Handle CE Which Was Initiated by Control Unit
CE Back to CPU,Checked for Error Status,OK,Issue Serv Out,CE Not Put In IB,Feed 3rd Card
IPL Windup Routine Reset Load Light
Go to Load PSW Routine to Setup 1st PSW from Main Storage 000
010 Trap Caused by DE Int
Check Dev End Status Store Device Addr and 00 Status in the IB
Issue Comm Out if Stat Good
Go to I Cycles

DETAILED SEQUENCE FOR 3rd CARD

INFORMATION	ROS ADDRESS	LOGIC PAGE
Addr Out-Supp Out	5AO	QC051
Addr Out-Sel Out Op In	59D	QC051
Op In-Addr In	5A8	QC051
Stat In-Op In, End of Initial Sel Seq	5B0	QC051
Enter Data Loop	6C1	QC231
Control Unit End Initiates CE Status Prepare for IPL Windup	647	QC121
Reset Load Light	6FE	QC351
Enter LPSW Rtne	0377	QA921
I Cycle Start	109	QA001
DE Stat Causes Trap to 010	010	QC201
Load IB with Dev Addr	63D	QC121
Load IB with 00 Stat	640	QC121
Issue Comm Out to Stat In	64D	QC121
Restore CPU	6D9	QC311
Restore WX Reg	67F	QC311

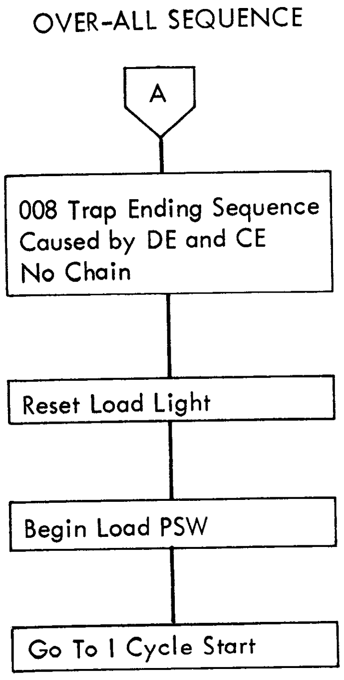
IPL Three Card Hex Loader--MPX Channel Burst Mode RS Chart 1 (Sheet 2)

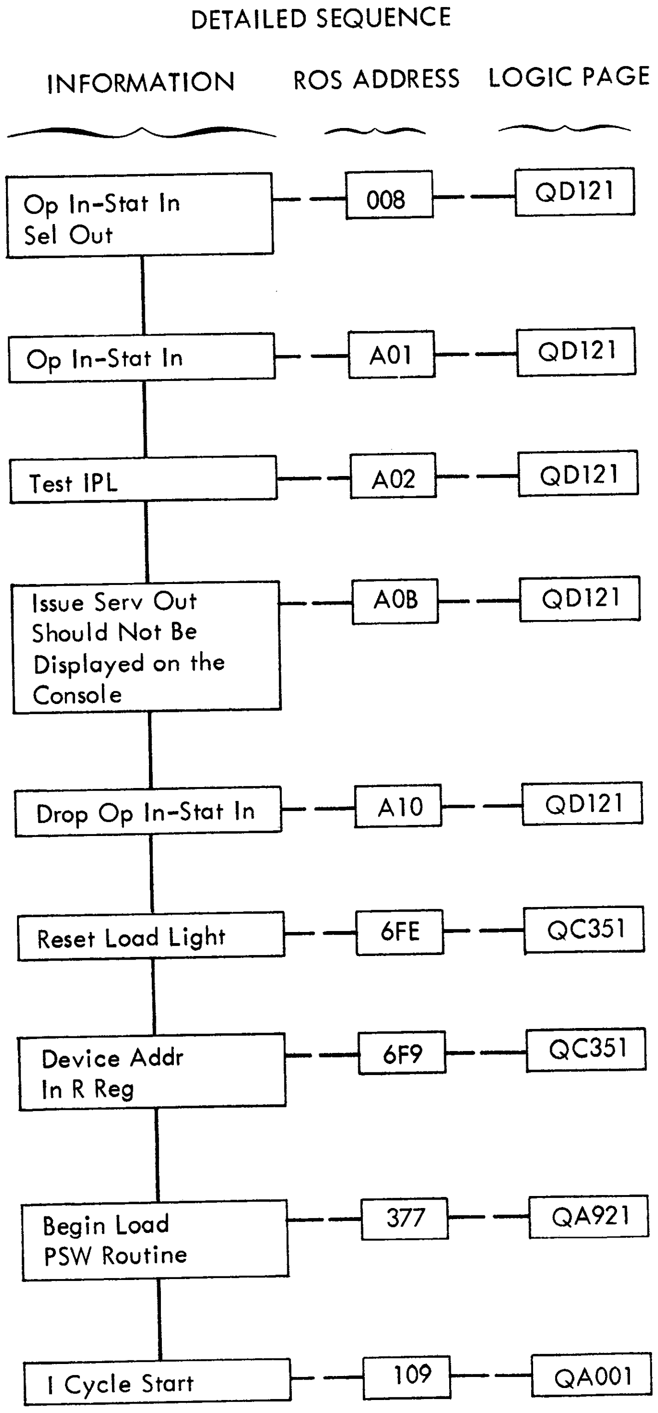
SELECTOR CHANNEL - TAPE & FILE IPL



Selector Channel--Tape and File IPL RS Chart 2 (Sheet 1)

SELECTOR CHANNEL - TAPE & FILE IPL





NOTE:

1. File Control Unit Decodes the 02 as a Read IPL
2. Control Unit Causes a Seek to Cylinder Zero
3. Selects Head Zero
4. Search Address Mark on Record One
5. Reads Record One Data Field (24 Byte Loader)
6. Present Ending Status
7. Channel takes an 00B Trap if Chaining is designated in the Selector Channel GF Register

I OBJECTIVE APPROACH TO CHANNELS

A. Enter the following program using the illustrated variables desired to simulate your failure. Enter the program by using either manual store or BF6.

NOTE: Clear Memory using BF8 prior to entering program.

Review of BF6 procedure:

1. System Reset
2. Set IC to 1st Address to be Altered
3. Roar Reset to BF6
4. Start
5. Set Switches HJ to Desired Character
6. Start
7. Repeat Step 5 & 6 until Data Entry is Complete
8. Set IC to 500; Depress Start

By insertion of the following instructions in front of the I/O program, we can simulate customer operation. This is accomplished by introducing a time constant prior to execution of the I/O program.

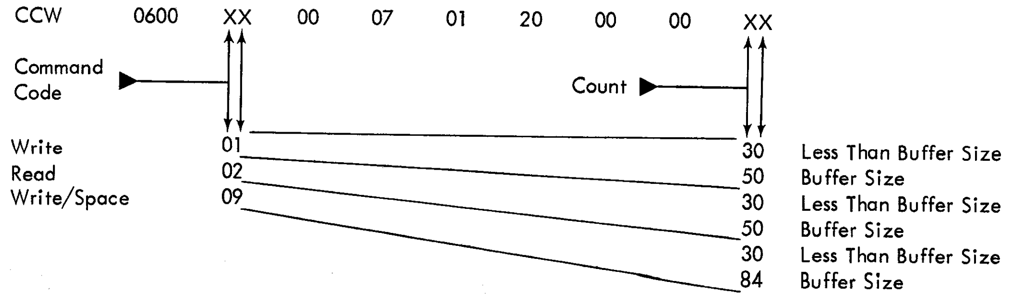
MAIN STORAGE ADDRESS

```
Load Reg          04F8   41   10   01   00

Branch on Count   04FC   46   10   04   FC

                  0500 - Beginning of I/O
                         Program
```

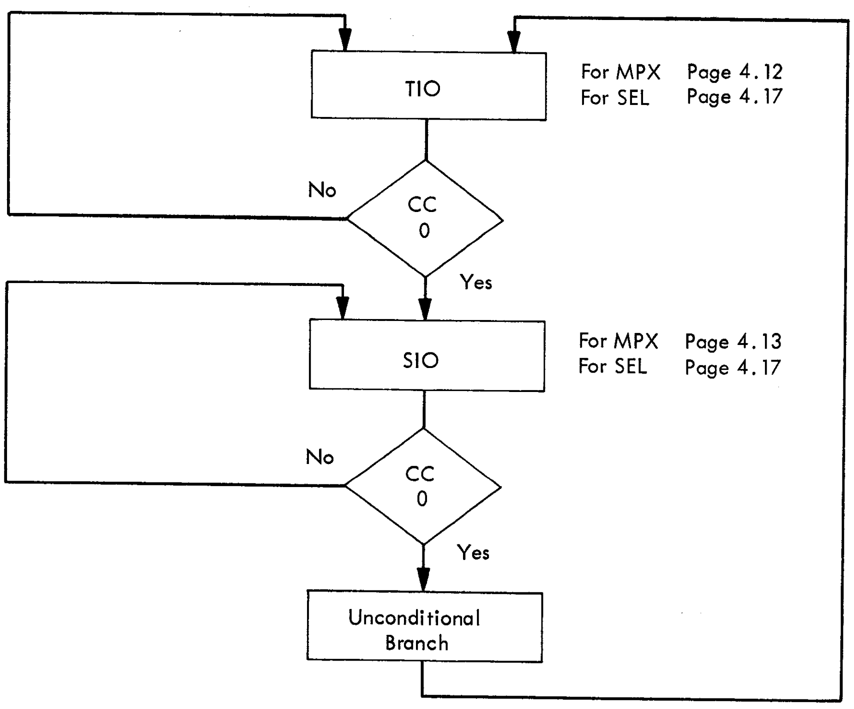
In the above example, we load register 1 with a value of Hex 100 which in turn causes the Branch on Count instruction to cycle 256 times prior to entering the I/O program. It should be noted that by changing the address in the Load Reg instruction varies the cycle time on the Branch on Count instruction.

MAIN STORAGE ADDRESS

```
Test I/O    0500   9D   00   0X   XX  :
                                      :
            0504   47   70   05   00  :......Channel And
                                      :      Device Address
Start I/O   0508   9C   00   0X   XX  :

            050C   47   70   05   08

            0510   47   F0   05   00


CAW         0048   00   00   06   00

CCW         0600   XX   00   07   01   20   00   00   XX
```



PROGRAM FLOW IN INSTRUCTION STEP FOR SELECTOR/MPX CHANNELS



Address of Next Instruction Displayed in A & B Reg in Instruction Step.

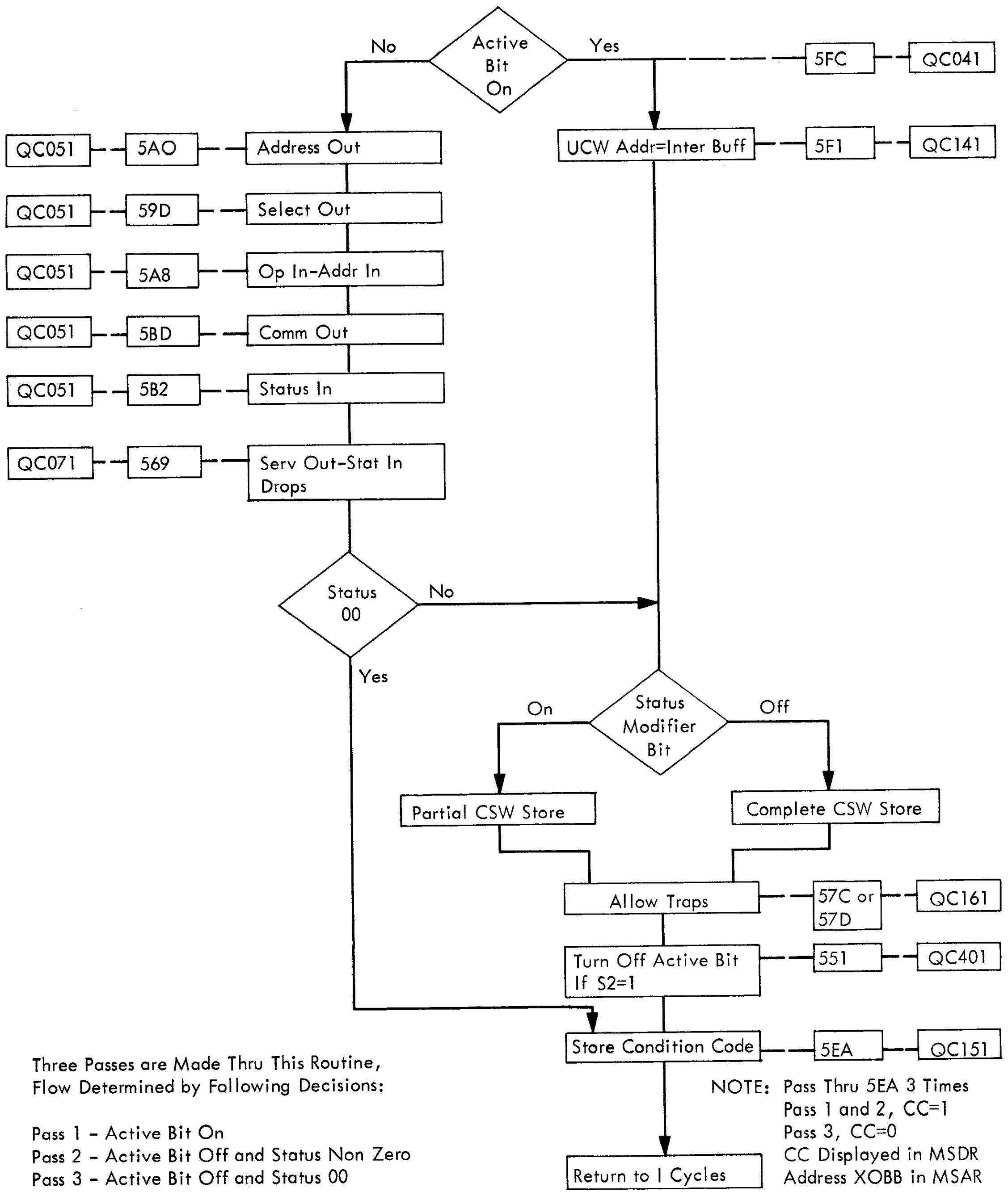
NOTE: Initial Test I/O after System Reset has:
CC = 0,
IB = 00
CSW = 00

Condition Code settings shown in charts are after execution of instruction.

MPX - Channel Address Trace	500	504	500	504	500	504	508	50C	510
Condition Code	1		1		0		0		
Interrupt - Buffer	00		00		00		08		
CSW - Byte 44	08	08	04	04	04	04	04	04	04
CSW - Byte 45	00	00	00	00	00	00	00	00	00

SEL- Channel Address Trace	500	504	500	504	508	50C	510
Condition Code	1		1	0	0	0	0
CSW - Byte 44	0C	0C	0C	0C	0C	0C	0C
CSW - Byte 45	00	00	00	00	00	00	00

TEST I/O - MPX - PRINTER - READER - PUNCH

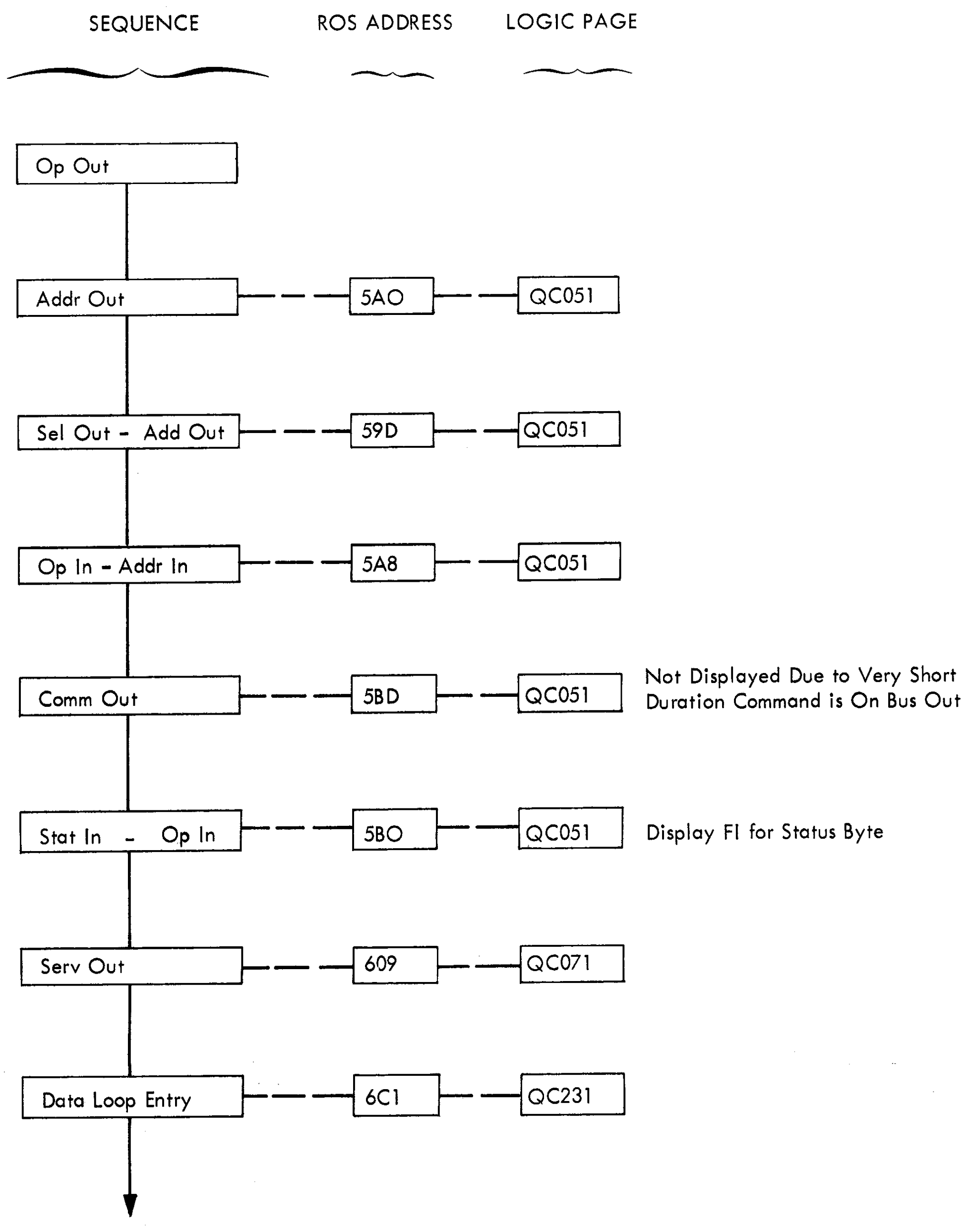


NOTE: Pass Thru 5EA 3 Times
Pass 1 and 2, CC=1
Pass 3, CC=0
CC Displayed in MSDR
Address XOBB in MSAR

Three Passes are Made Thru This Routine, Flow Determined by Following Decisions:

Pass 1 - Active Bit On
Pass 2 - Active Bit Off and Status Non Zero
Pass 3 - Active Bit Off and Status 00

MPX - SIO INITIAL SELECTION

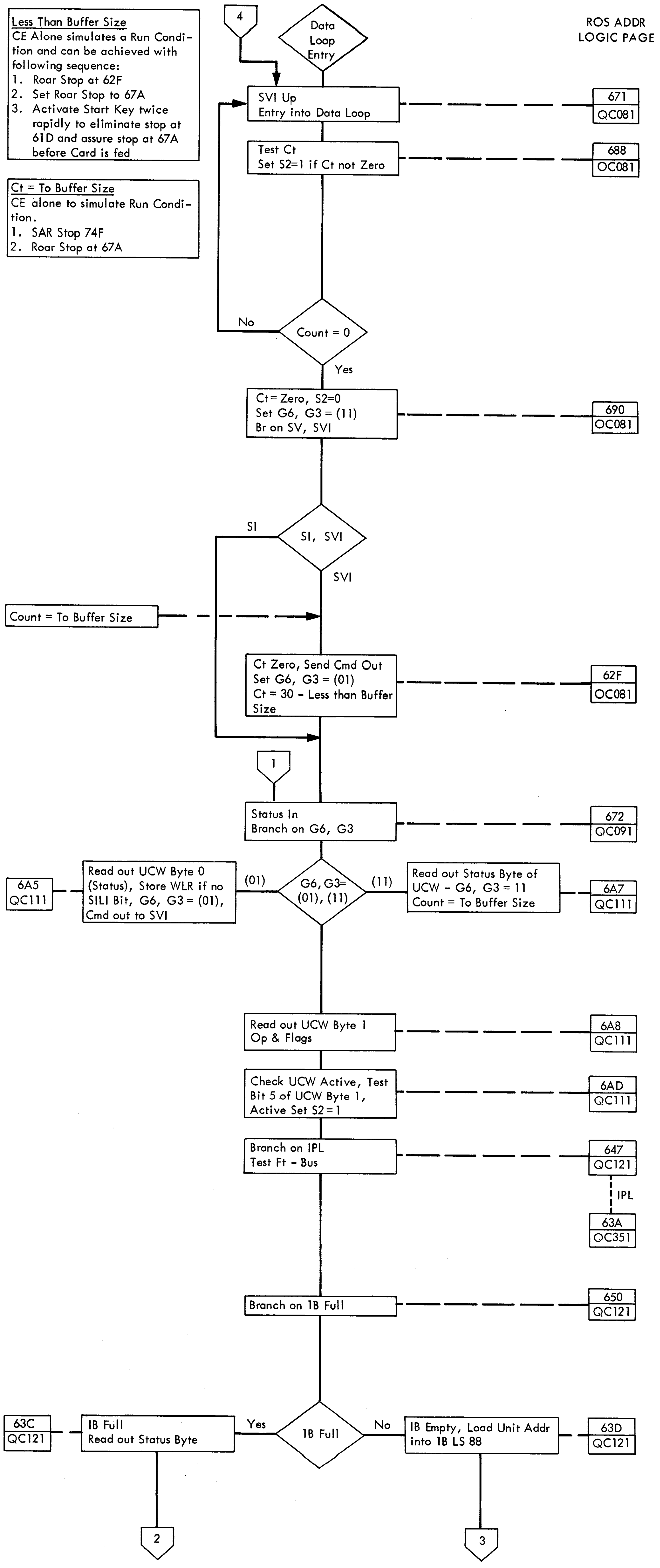


Not Displayed Due to Very Short Duration Command is On Bus Out

Display FI for Status Byte

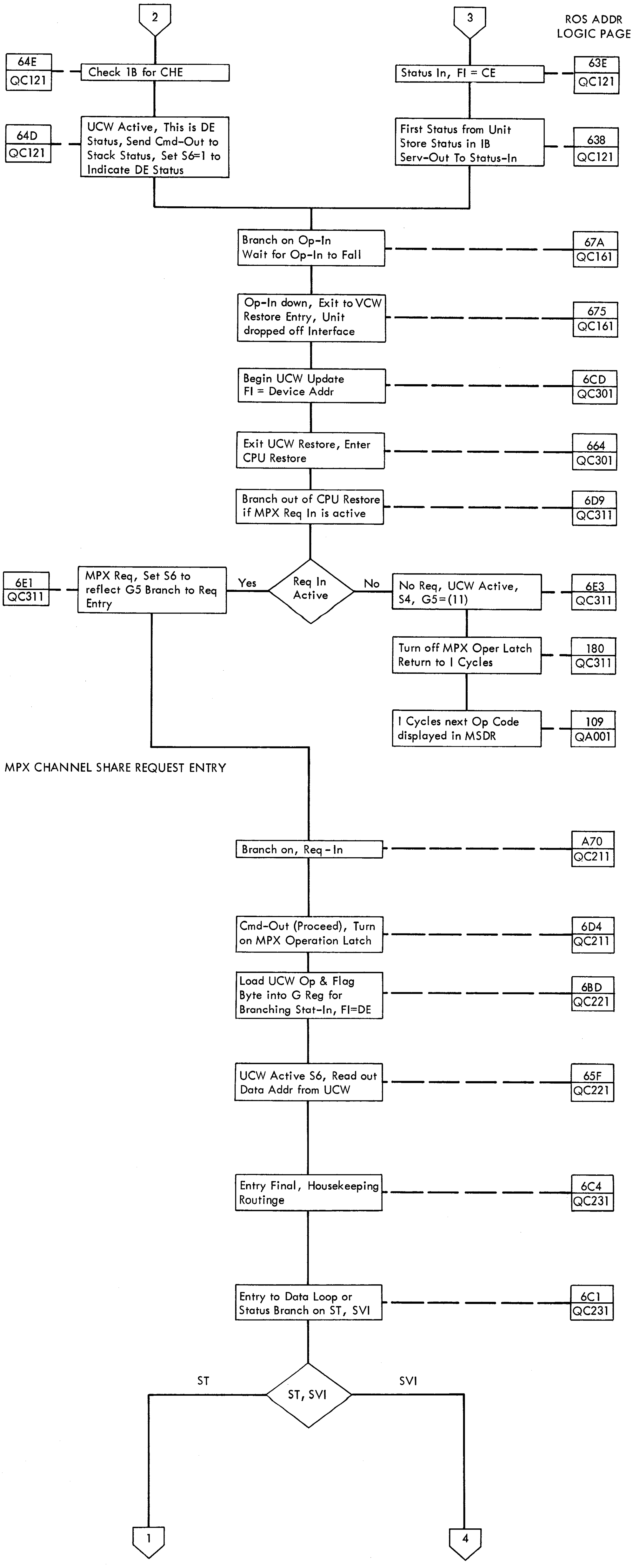
Exit to Proper Routine is Dependent on the Count Fld and Mode Setting

MPX - SIO INPUT/OUTPUT BURST OR BYTE MODE CHART REPRESENTS = TO AND LESS THAN BUFFER SIZE

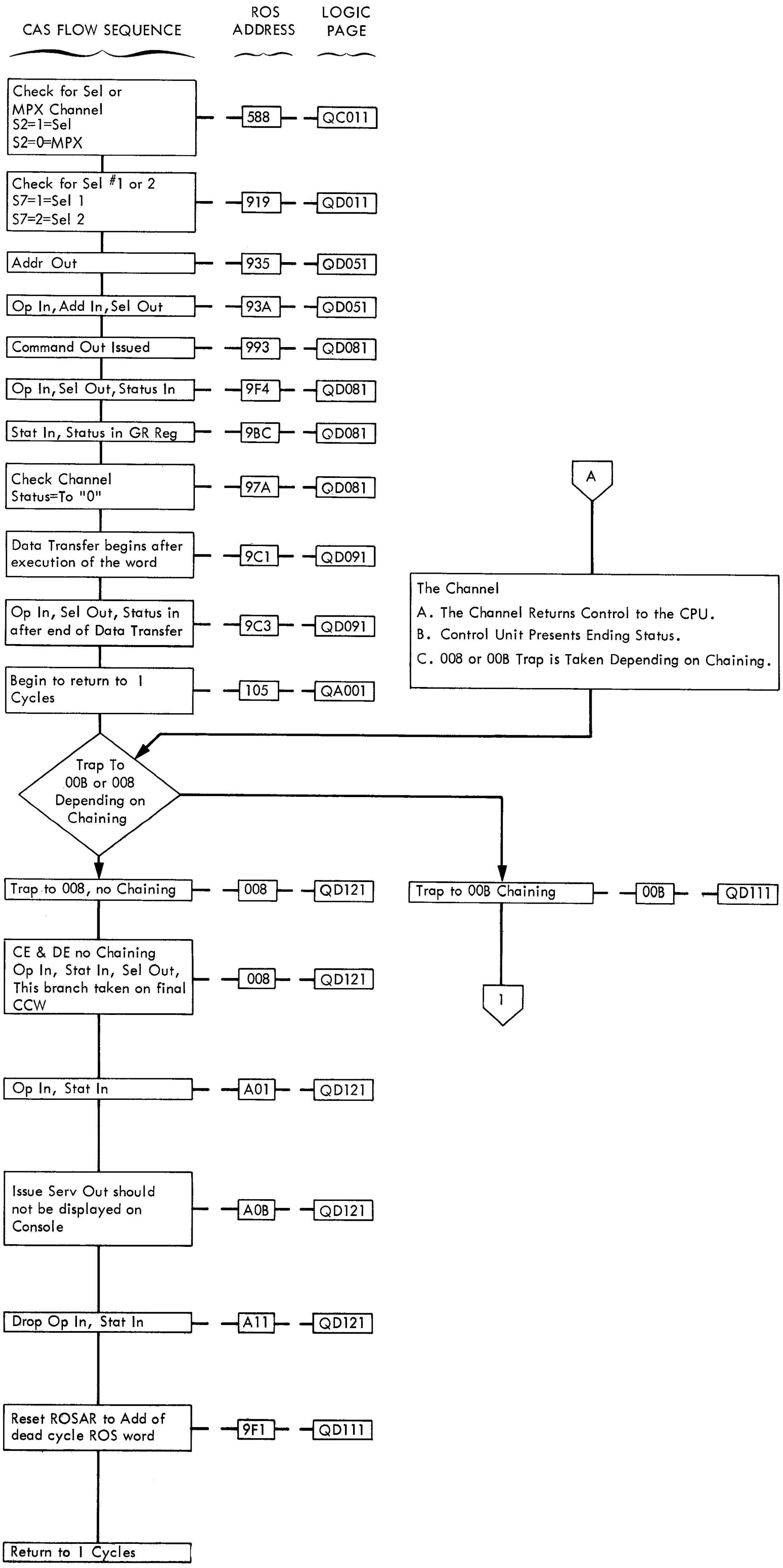


MPX--SIO Input/Output Burst or Byte Mode RS Chart 4 (Sheet 1)

MPX - SIO INPUT/OUTPUT OR BYTE MODE CHART REPRESENTS = TO AND LESS THAN BUFFER SIZE

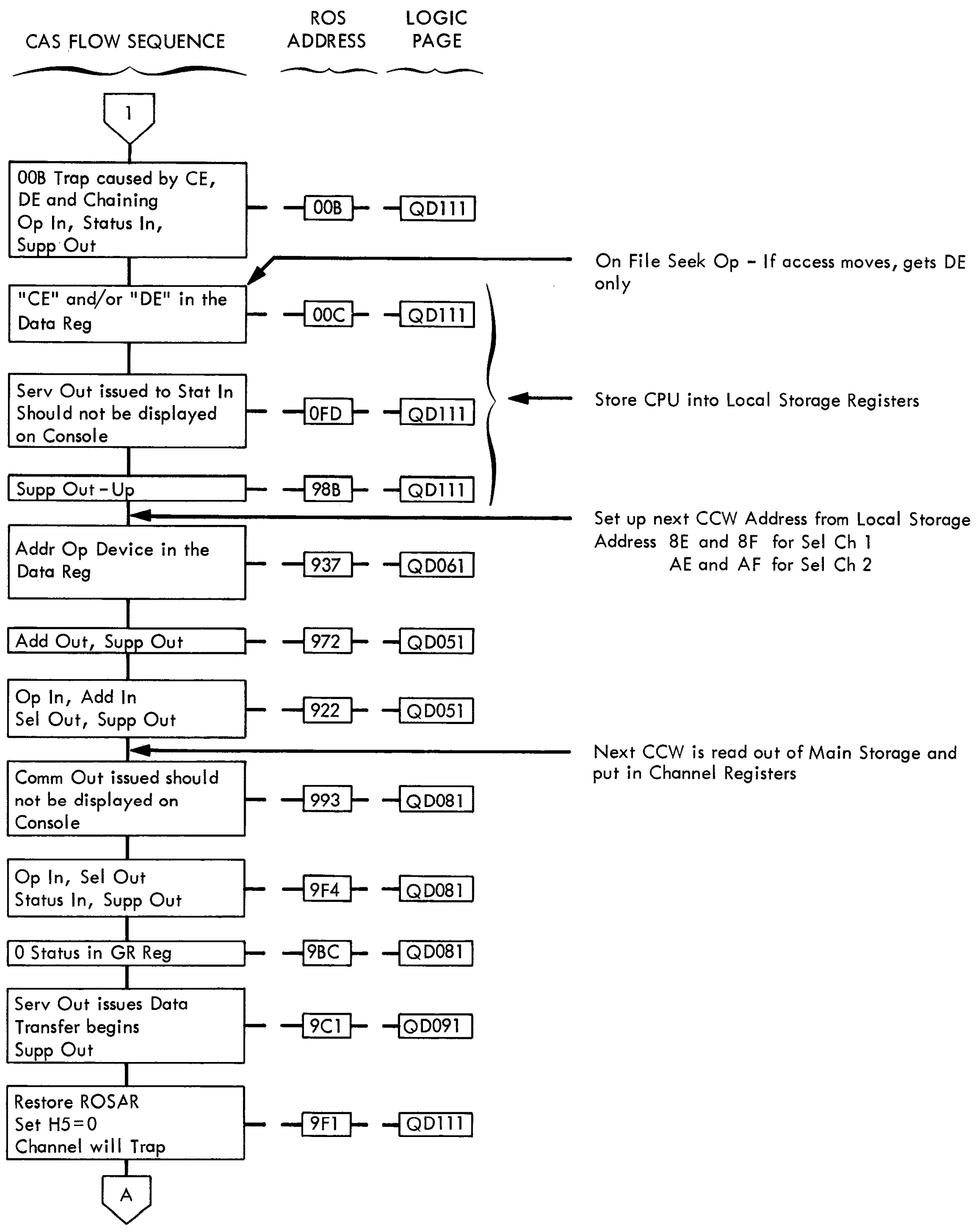


MPX--SIO Input/Output Burst or Byte Mode RS Chart 4 (Sheet 2)

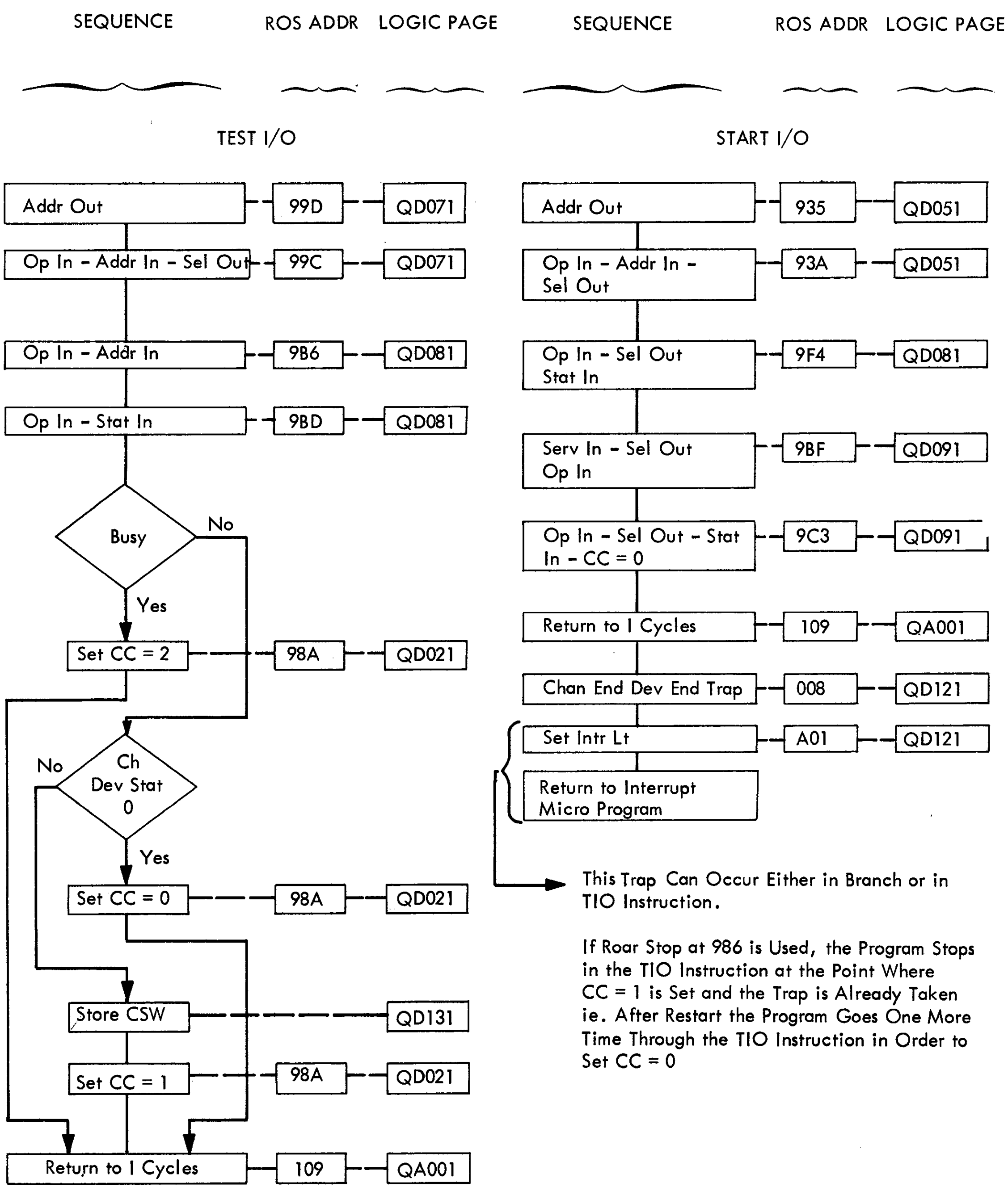


Selector Channel RS Chart 5 (Sheet 1)

TRAP, CHAINING ON SELECTOR CHANNEL



Selector Channel RS Chart 5 (Sheet 2)



This Trap Can Occur Either in Branch or in TIO Instruction.

If Roar Stop at 986 is Used, the Program Stops in the TIO Instruction at the Point Where CC = 1 is Set and the Trap is Already Taken ie. After Restart the Program Goes One More Time Through the TIO Instruction in Order to Set CC = 0

Selector Channel TIO and SIO RS Chart 6 (Sheet 1)

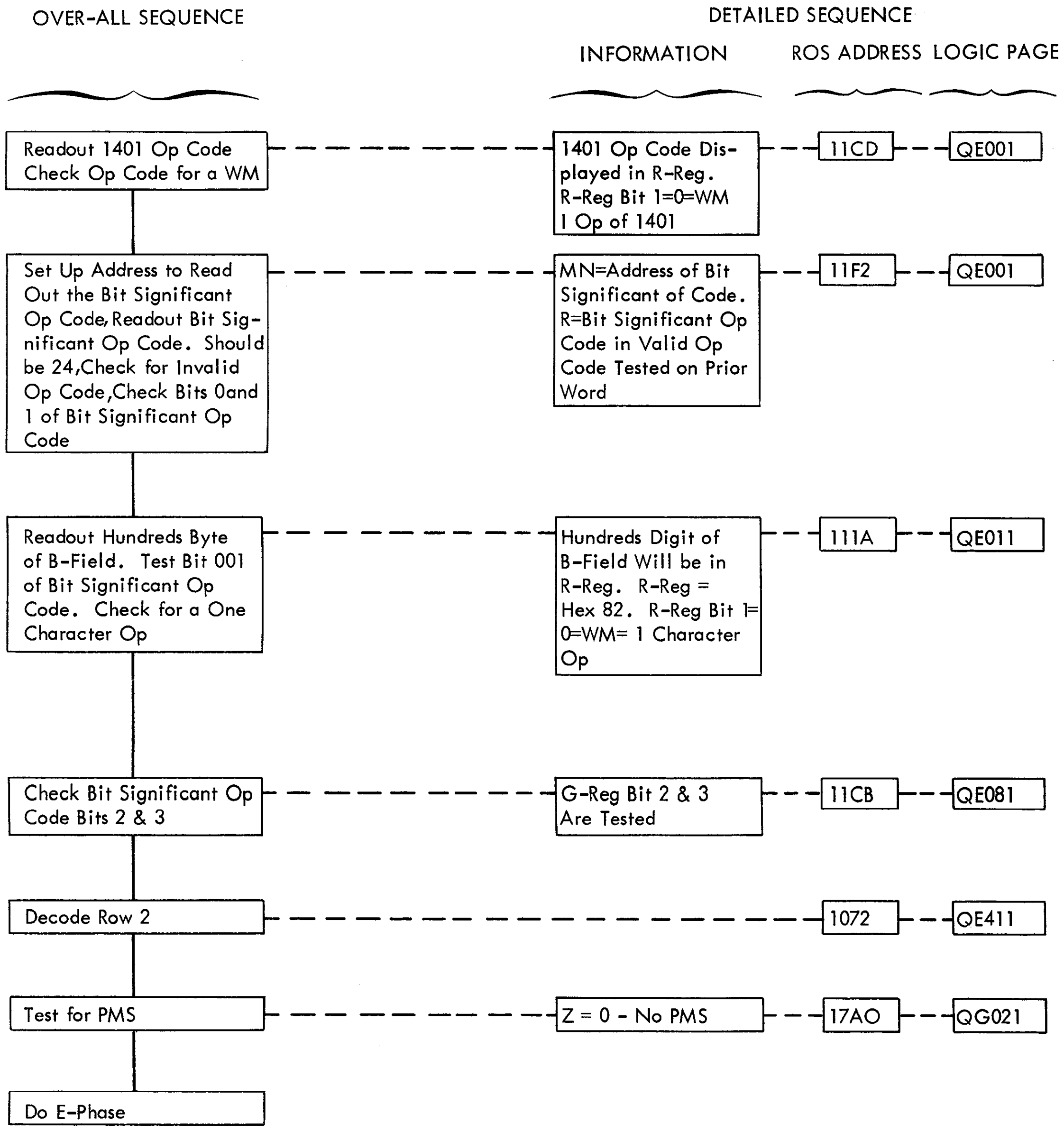
1400 COMPATIBILITY I CYCLE
500 M 555 666
507 B
555
666 1

OVER-ALL SEQUENCE	DETAILED SEQUENCE INFORMATION	ROS ADDRESS	LOGIC PAGE
Readout OP Code Check OP Code for WM	1400 OP Code = 94 displayed in R-Reg R-Reg Bit 1=0=WM I Op of I Cycles	11CD	QE001
Set Up Addr To Readout the Bit Signfc. OP Code readout Bit Signfc OP code should be 80 Check for inval. OP Code Check Bits 0 & 1 of Bit Signfc. OP Code	MN= Addr of Bit Signfc. OP Code,R= Bit Signfc. OP Code inval. OP Code tested on prior word	11F2	QE001
Store B Star in Aux A Storage 8A	Save B Star for possible Store B-Star Op Code	1110	QE011
Readout Hundr. Byte of A-Field Addr	Hundred Digit of A-Field will be in R-Reg	111B	QE011
Readout Tens Byte of A-Field Addr	Tens Digit of A-Field will be in R-Reg	112F	QE011
Readout Units Byte of A-Field Addr	Units Digit of A-Field will be in R-Reg	15E4	QE051
Readout Hundred Byte of B-Field Addr	Hundred Digit of B-Field will be in R-Reg	119E	QE061
Xfer of B-Star to A-Star	I.E. A-Field Hex equiv. Addr is now in LT Reg	11A5	QE071
Readout Tens Byte of B-Field Addr	Tens Digit of B-Field will be in R-Reg Check for Index	112F	QE011
Readout Units Byte of B-Field Addr	Units Digit of B-Field will be in R-Reg B-Field Hex Equival. Addr is now in UV Reg	15E4	QE051
B Star Complete		1198	QE061
Readout Possible Modifier or next OP Code,Check for WM	In this program Example = B	119C	QE061
Do E Phase of Move OP			

1400 Compatibility I Cycle RS Chart 7 (Sheet 1)

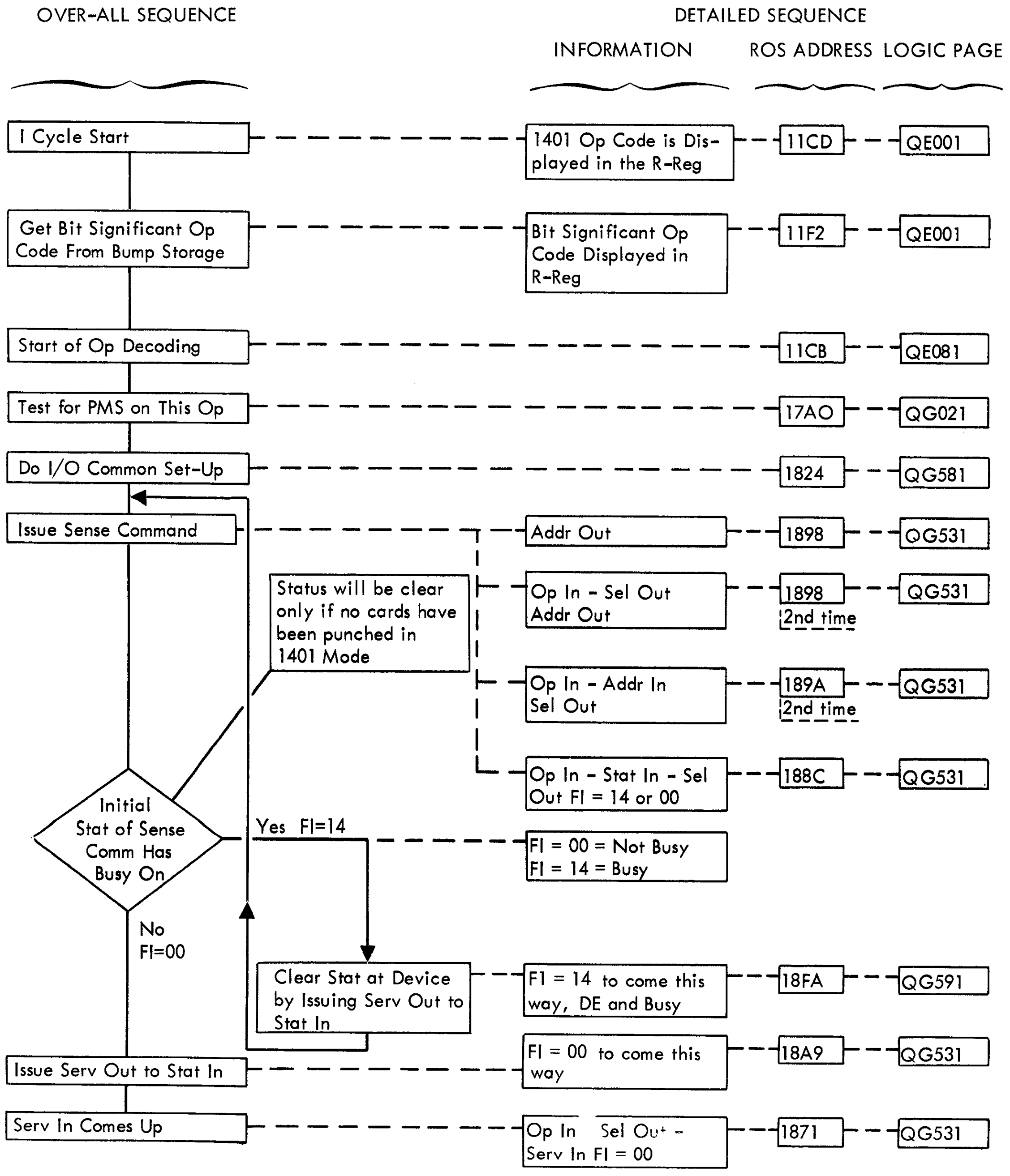
1401 COMPATIBILITY PUNCH OPERATION - I-CYCLES FOR 4 OP

444 4
445 B̄ 444
449 .̄



1401 COMPATIBILITY PUNCH OPERATION - EXECUTE

444 4
445 B̄ 444
449 .̄



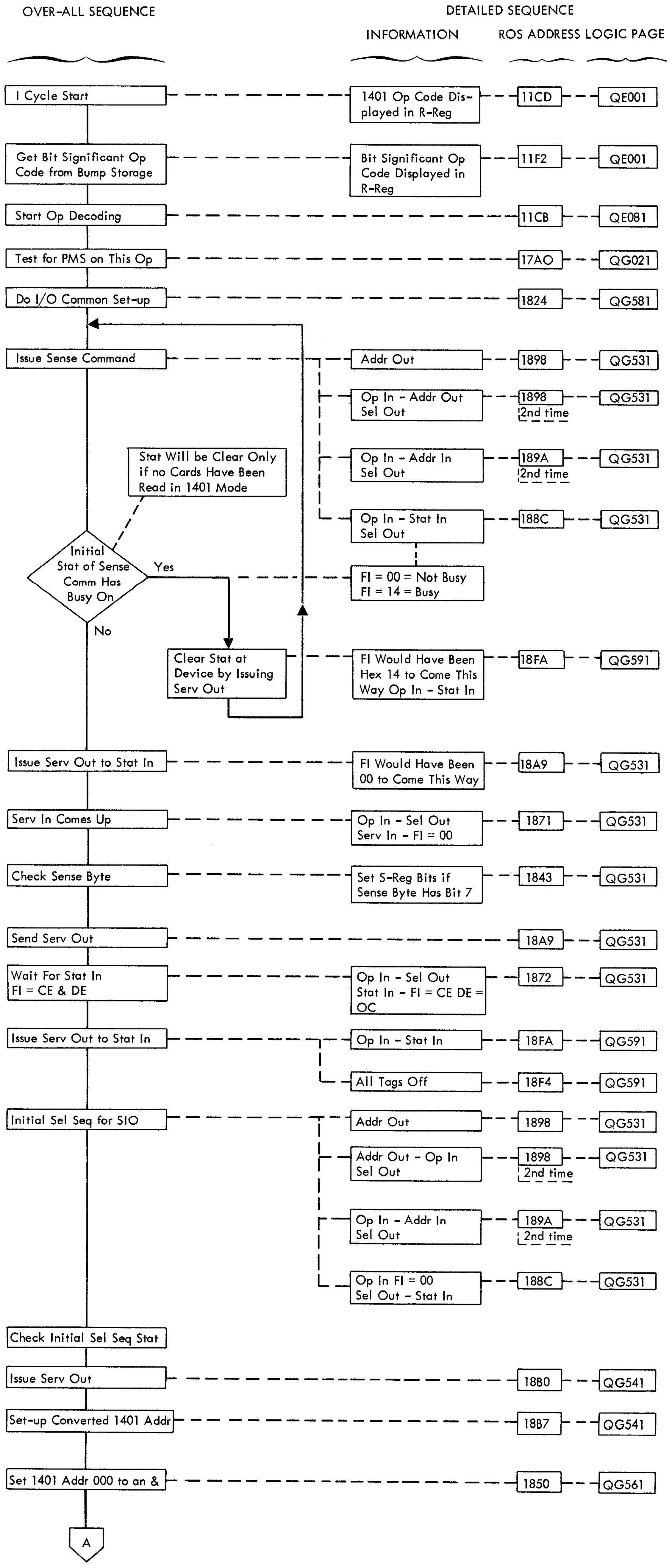
1401 COMPATIBILITY PUNCH OPERATION - EXECUTE

OVER-ALL SEQUENCE	DETAILED SEQUENCE		
	INFORMATION	ROS ADDRESS	LOGIC PAGE
Check Sense Byte	Set S-Reg Bits if Sense Byte has Bit 7 On	1843	QG531
Send Serv Out		18A9	QG531
Wait For Stat In FI = CE & DE	Op In - Sel Out Stat In FI = OC = CE & DE	1872	QG531
Issue Serv Out to Stat In	Op In - Stat In	18FA	QG591
	All Tags Off	18F4	QG591
Initial Sel Seq for SIO	Addr Out	1898	QG531
	Op In - Sel Out Addr Out	1898 2nd time	QG531
	Op In - Sel Out Addr In	189A 2nd time	QG531
	Op In - Sel Out Stat In FI = 00	188C	QG531
Check Initial Sel Seq Stat		1867	QG541
Issue Serv Out		18B0	QG541
Set Up Converted 1401 Addr - Set F0 Into 1401 Addr 100	MN = Converted 1401 B-Star Addr R = FO	186C	QG541
Go Into Data Handling Routine		18F3	QG521
Control Unit Ends Op by Presenting CE Stat	Op In - Stat In Sel Out FI = CE	1822	QG521
Check Ending Stat for Errors,Unit Check or Unit Exception		181B	QG521
Reset Sel Out	Op In - Stat In FI = CE	1819	QG521
Change Bit Significant Op Code in G Reg from 24 to 20		1890	QG521
Update B Addr to 1401 Addr of 180		1819	QG521
Send Serv Out In Response to Stat In	After Execution of This Word All Tags Will be Off	1818	QG571
Return to I Cycles		1100	QE001

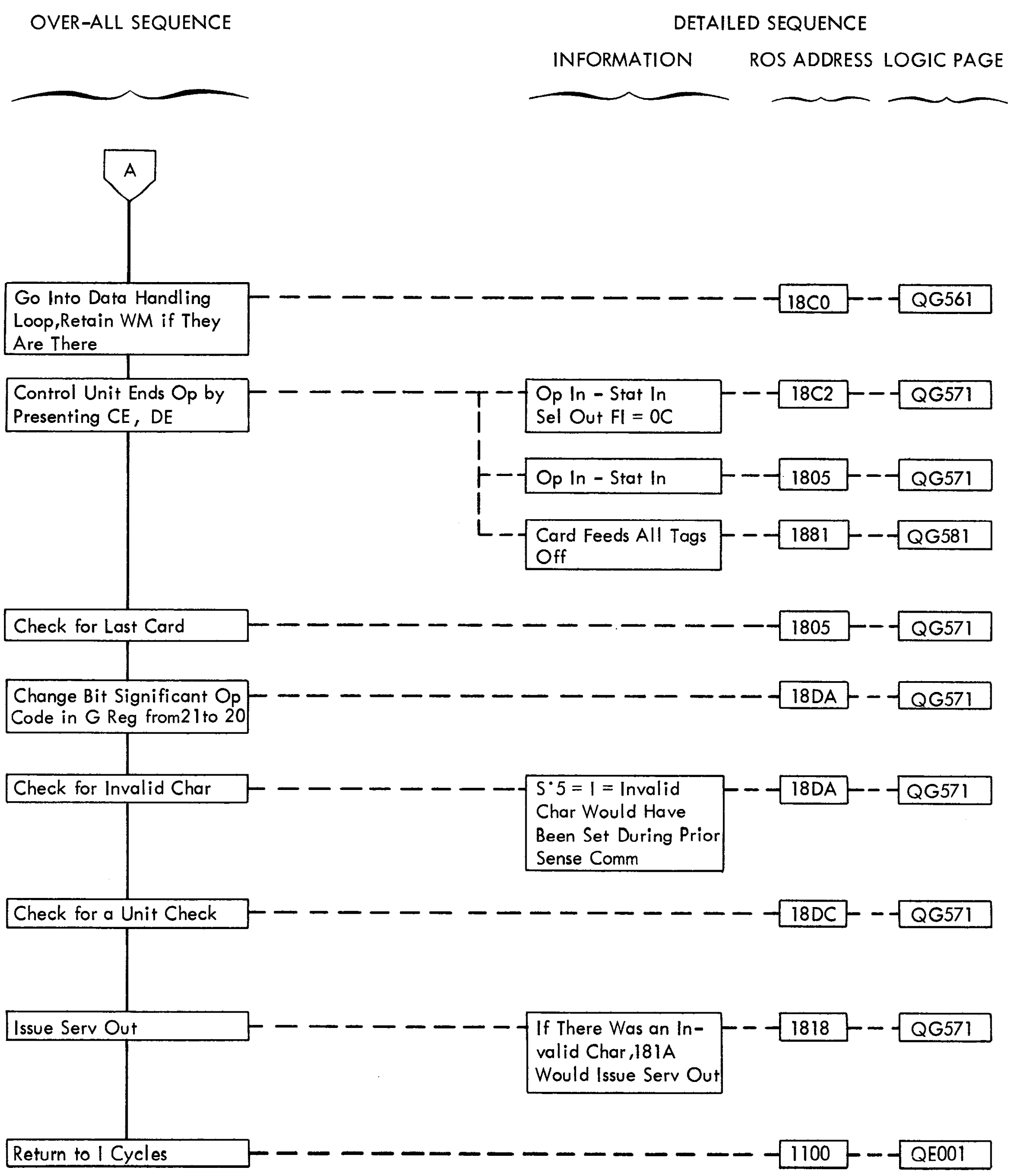
1401 Compatibility Punch Operation RS Chart 8 (Sheet 2)

1401 COMPATIBILITY READ OPERATION

444 1
445 B̄ 444
449 .̄



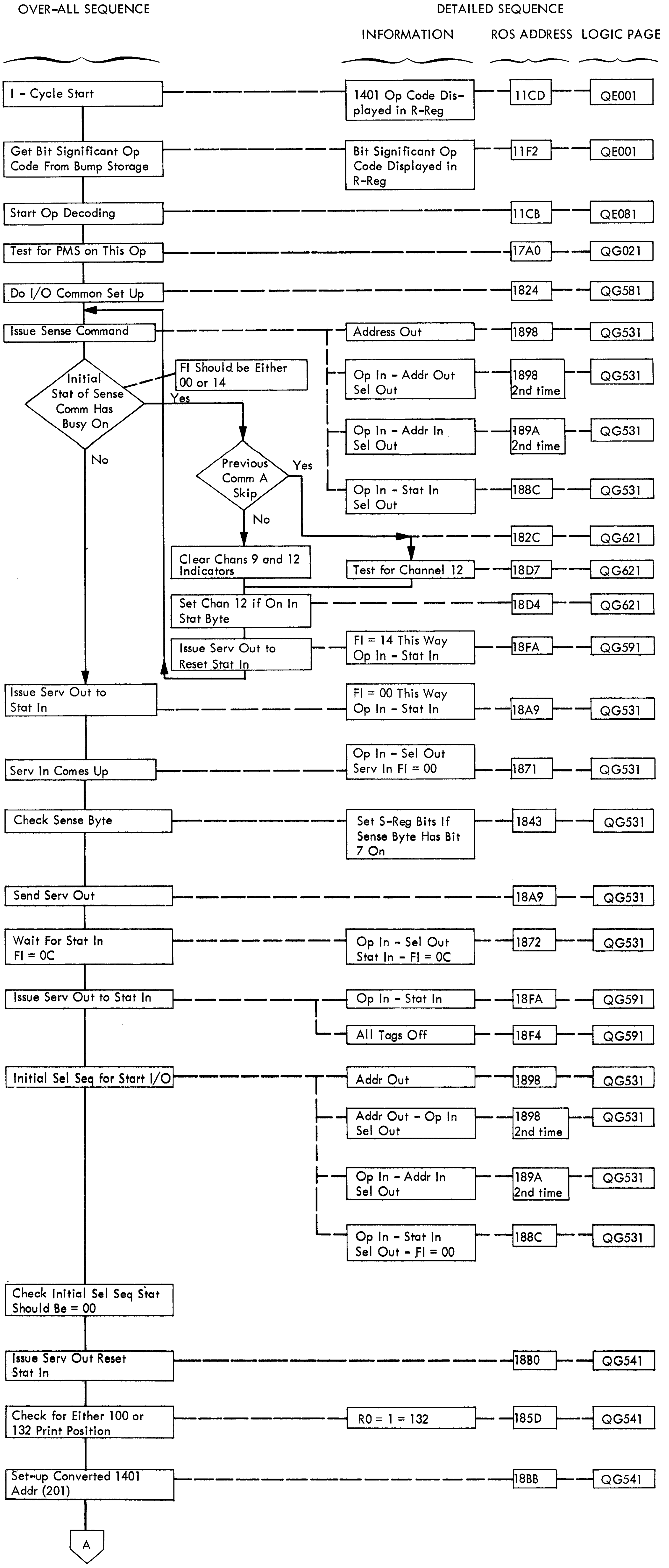
1401 Compatibility Read Operation RS Chart 9 (Sheet 1)



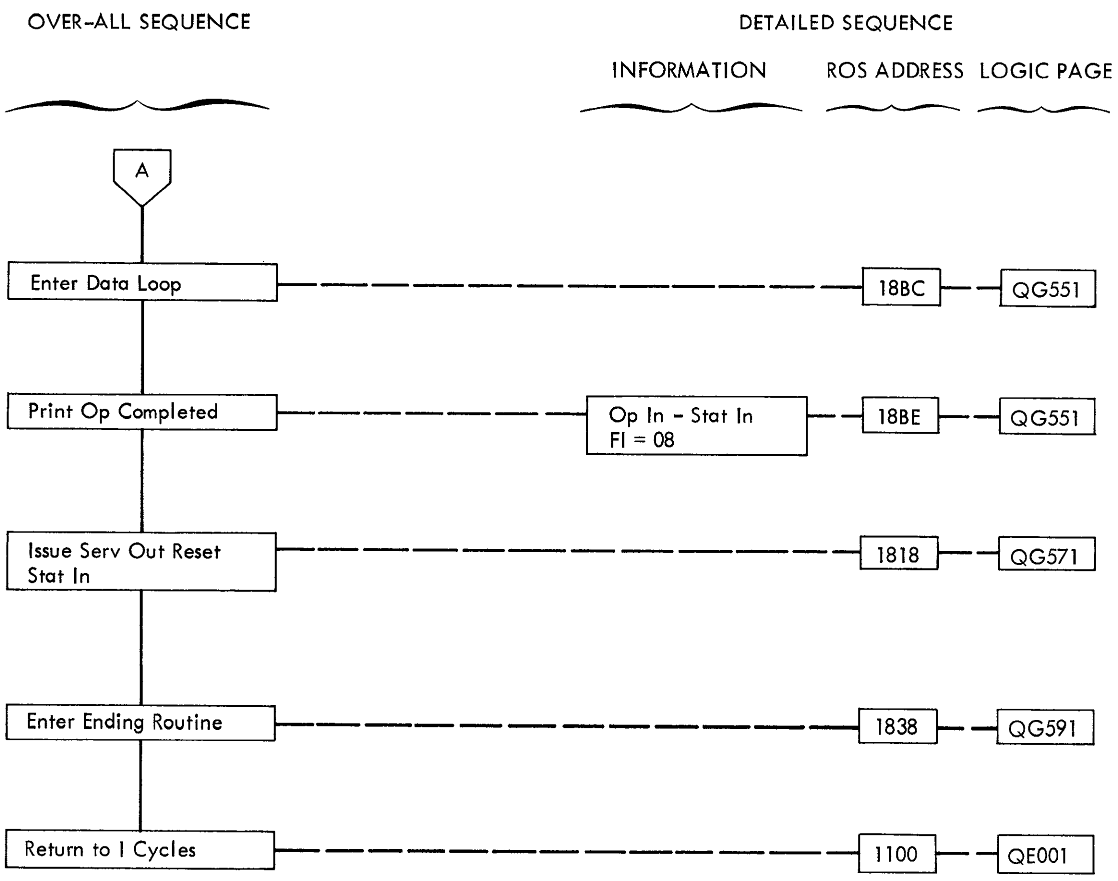
1401 Compatibility Read Operation RS Chart 9 (Sheet 2)

1401 COMPATIBILITY PRINT OPERATION

444 2
445 B 444
449 .



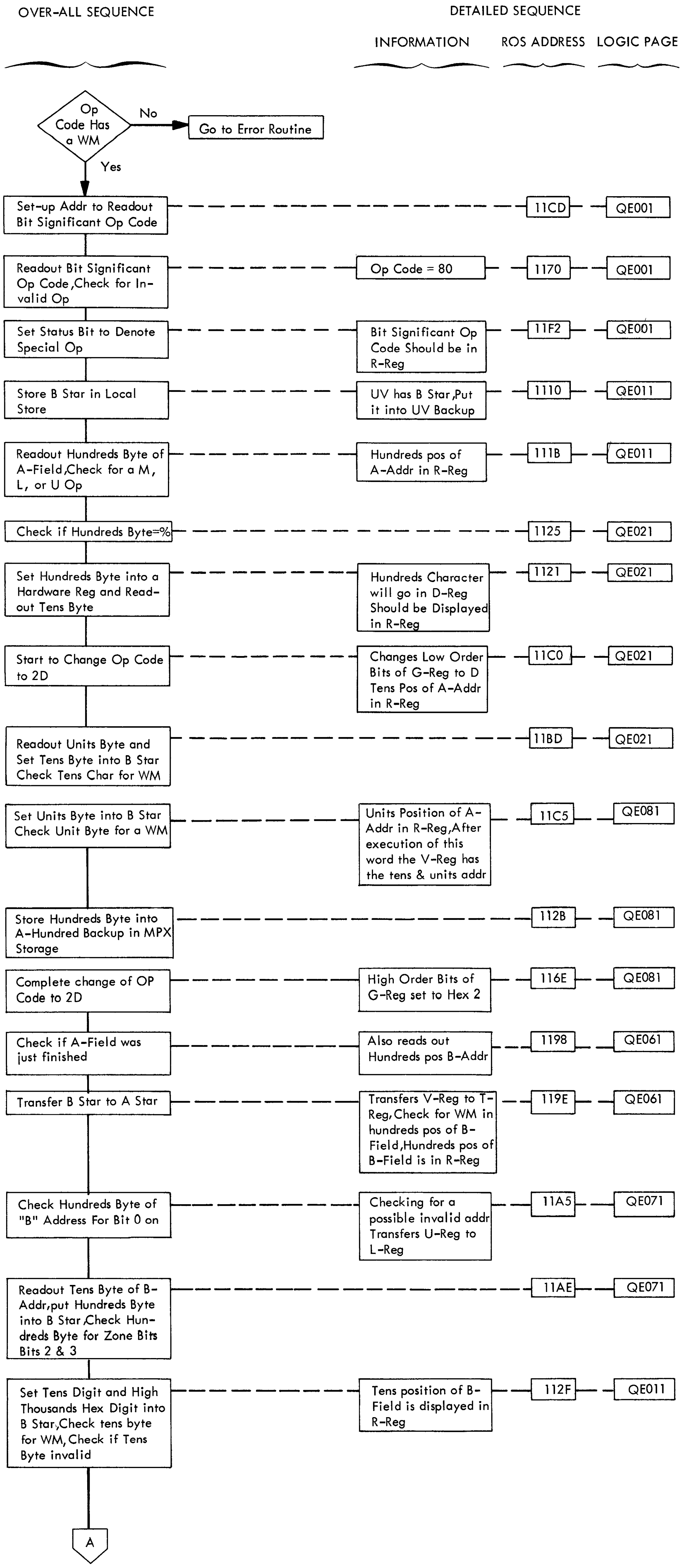
1401 Compatibility Print Operation RS Chart 10 (Sheet 1)



1401 Compatibility Print Operation RS Chart 10 (Sheet 2)

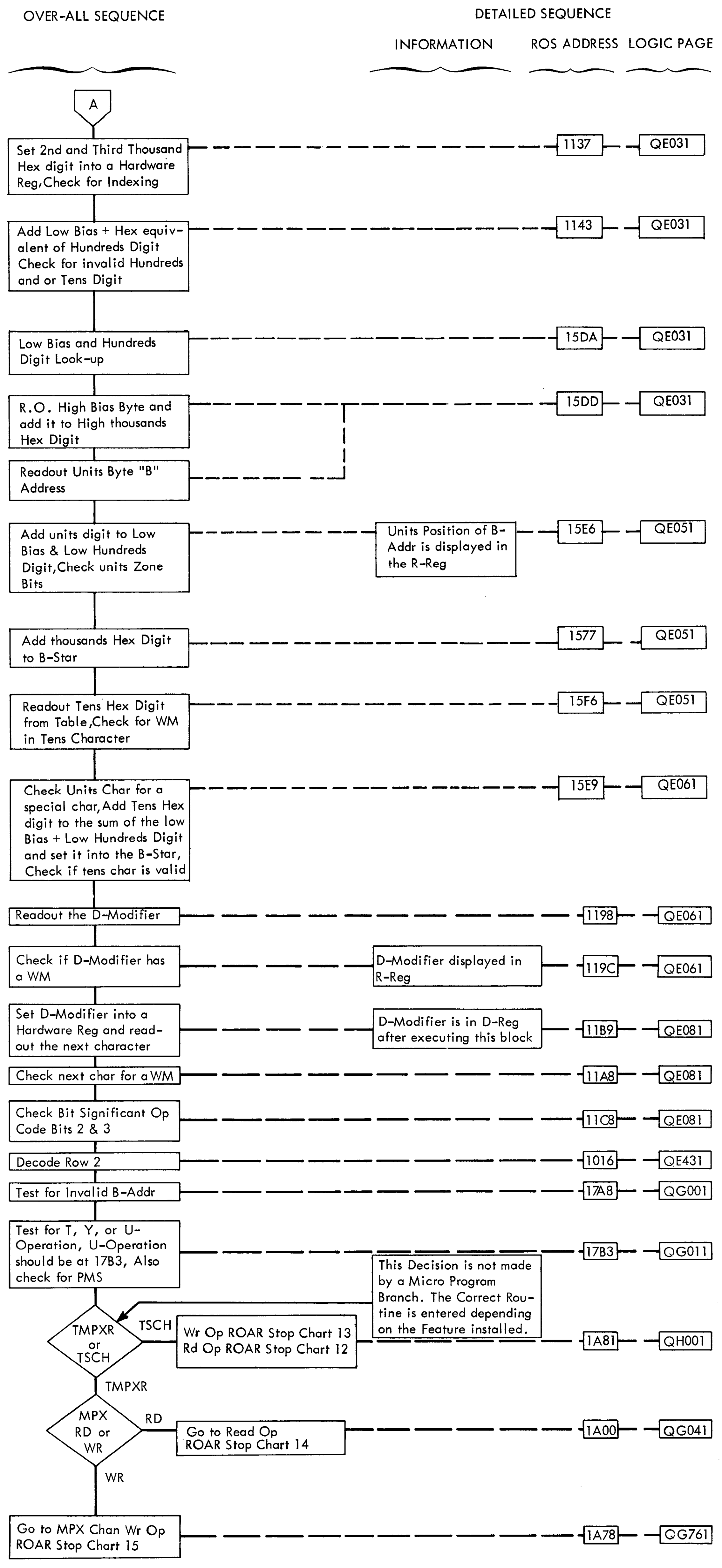
1400 TAPE OPERATION - TAPES ON MPX CHANNEL OR SELECTOR CHANNEL - I CYCLES

444 M % U 1 500W
452 $\bar{B}$ 444
456 $\bar{.}$



1400 Tape Operation-Tapes on MPX Channel or Selector Channel--I Cycles RS Chart 11 (Sheet 1)

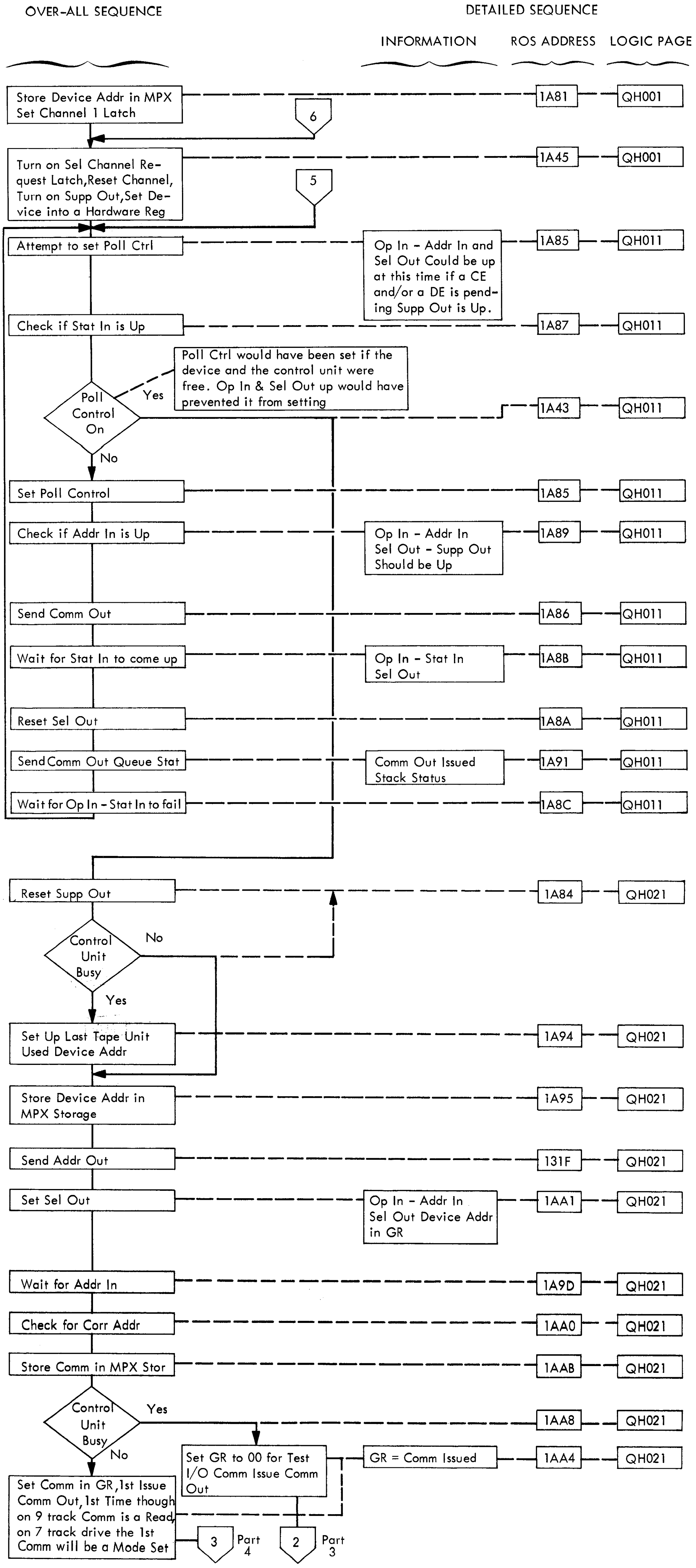
1400 TAPE OPERATION - TAPES ON MPX CHANNEL OR SELECTOR CHANNEL - I CYCLES



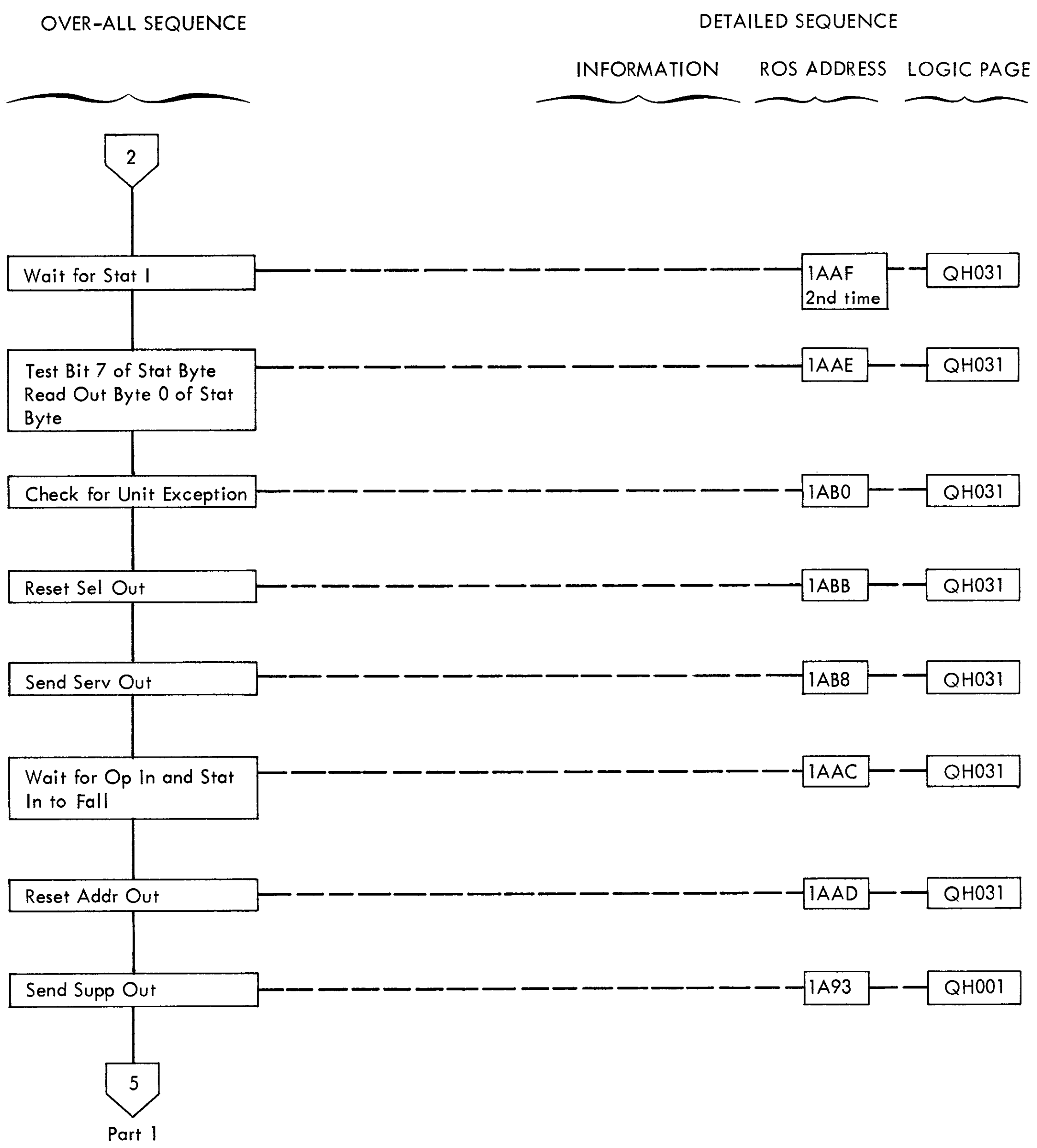
1400 Tape Operation-Tapes on MPX Channel or Selector Channel--I Cycles RS Chart 11 (Sheet 2)

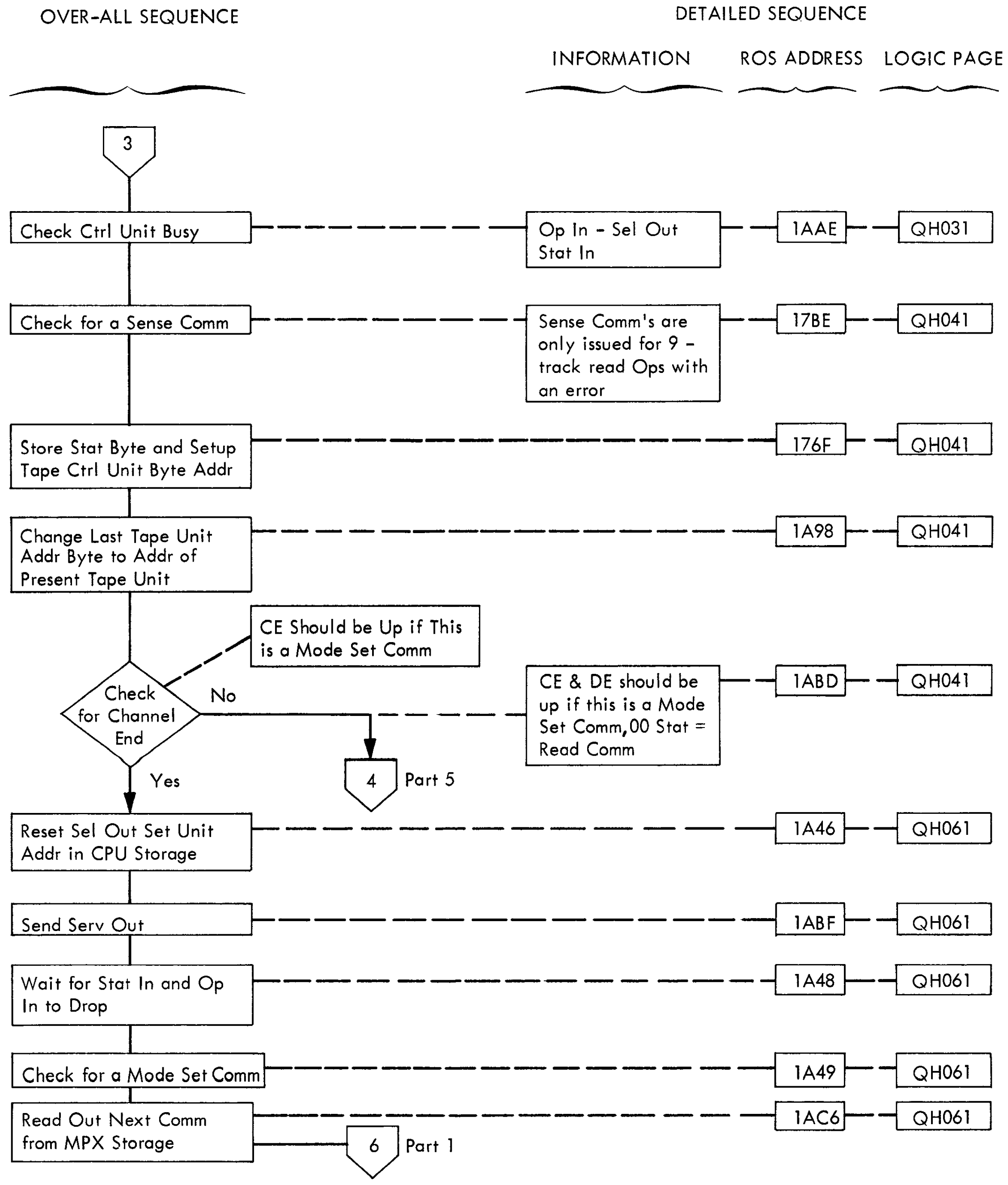
1400 COMPATIBILITY TAPE READ OPERATION - SELECTOR CHANNEL

444 M %U1 500R
452 B 444
456 -

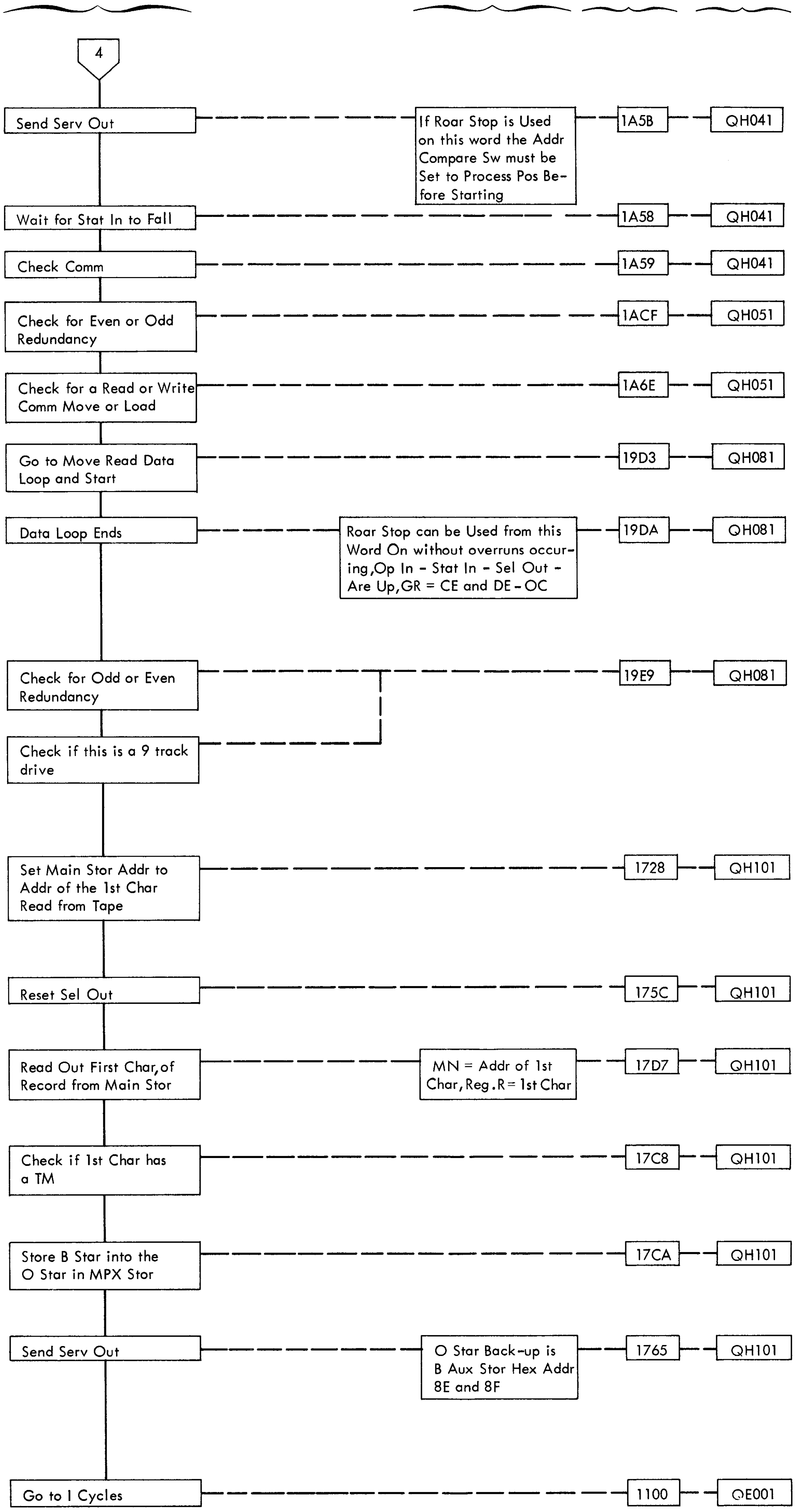


1400 Compatibility Tape Read Operation--Selector Channel RS Chart 12 (Sheet 1)


OVER-ALL SEQUENCE
DETAILED SEQUENCE
INFORMATION
ROS ADDRESS
LOGIC PAGE
2
Wait for Stat I
1AAF 2nd time
QH031
Test Bit 7 of Stat Byte Read Out Byte 0 of Stat Byte
1AAE
QH031
Check for Unit Exception
1AB0
QH031
Reset Sel Out
1ABB
QH031
Send Serv Out
1AB8
QH031
Wait for Op In and Stat In to Fall
1AAC
QH031
Reset Addr Out
1AAD
QH031
Send Supp Out
1A93
QH001
5
Part 1


OVER-ALL SEQUENCE
DETAILED SEQUENCE
INFORMATION
ROS ADDRESS
LOGIC PAGE
3
Check Ctrl Unit Busy
Op In - Sel Out Stat In
1AAE
QH031
Check for a Sense Comm
Sense Comm's are only issued for 9 - track read Ops with an error
17BE
QH041
Store Stat Byte and Setup Tape Ctrl Unit Byte Addr
176F
QH041
Change Last Tape Unit Addr Byte to Addr of Present Tape Unit
1A98
QH041
CE Should be Up if This is a Mode Set Comm
Check for Channel End
No
CE & DE should be up if this is a Mode Set Comm, 00 Stat = Read Comm
1ABD
QH041
4
Part 5
Yes
Reset Sel Out Set Unit Addr in CPU Storage
1A46
QH061
Send Serv Out
1ABF
QH061
Wait for Stat In and Op In to Drop
1A48
QH061
Check for a Mode Set Comm
1A49
QH061
Read Out Next Comm from MPX Storage
1AC6
QH061
6
Part 1

1400 COMPATIBILITY TAPE READ OPERATION - SELECTOR CHANNEL

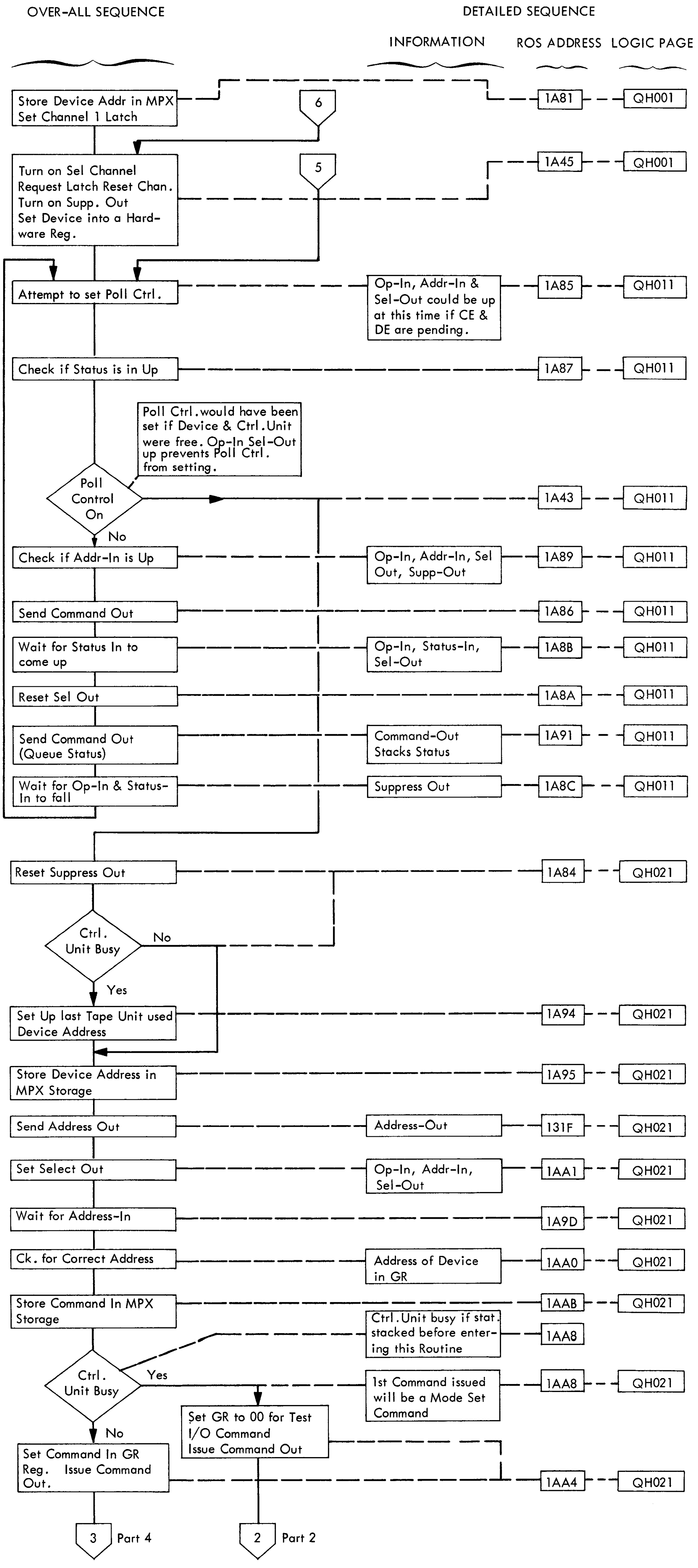


OVER-ALL SEQUENCE	DETAILED SEQUENCE INFORMATION	ROS ADDRESS	LOGIC PAGE
4			
Send Serv Out	If Roar Stop is Used on this word the Addr Compare Sw must be Set to Process Pos Before Starting	1A5B	QH041
Wait for Stat In to Fall		1A58	QH041
Check Comm		1A59	QH041
Check for Even or Odd Redundancy		1ACF	QH051
Check for a Read or Write Comm Move or Load		1A6E	QH051
Go to Move Read Data Loop and Start		19D3	QH081
Data Loop Ends	Roar Stop can be Used from this Word On without overruns occuring, Op In - Stat In - Sel Out - Are Up, GR = CE and DE - OC	19DA	QH081
Check for Odd or Even Redundancy Check if this is a 9 track drive		19E9	QH081
Set Main Stor Addr to Addr of the 1st Char Read from Tape		1728	QH101
Reset Sel Out		175C	QH101
Read Out First Char, of Record from Main Stor	MN = Addr of 1st Char, Reg. R = 1st Char	17D7	QH101
Check if 1st Char has a TM		17C8	QH101
Store B Star into the O Star in MPX Stor		17CA	QH101
Send Serv Out	O Star Back-up is B Aux Stor Hex Addr 8E and 8F	1765	QH101
Go to I Cycles		1100	QE001

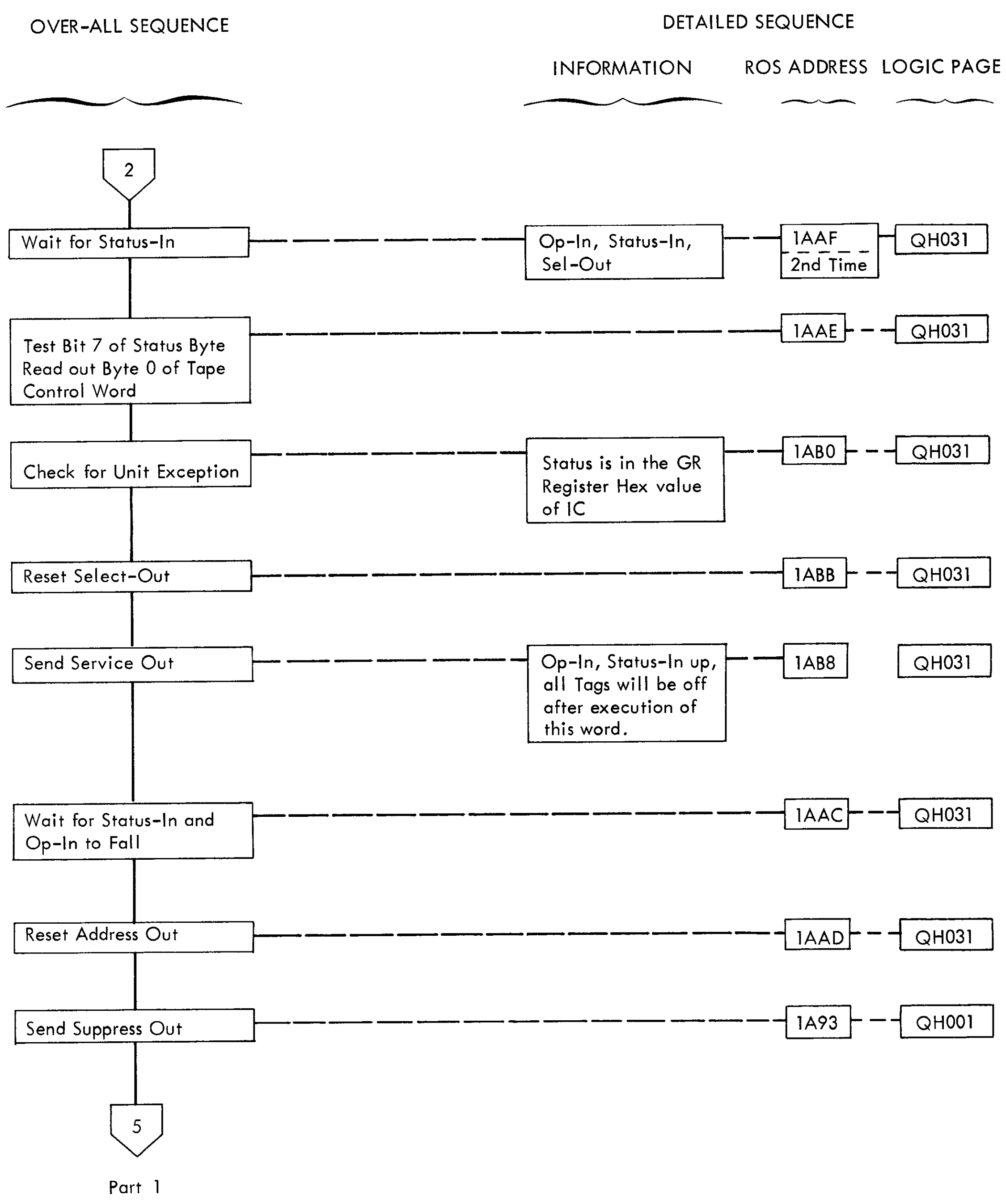
1400 Compatibility Tape Read Operation - Selector Channel RS 12 (Sheet 3)

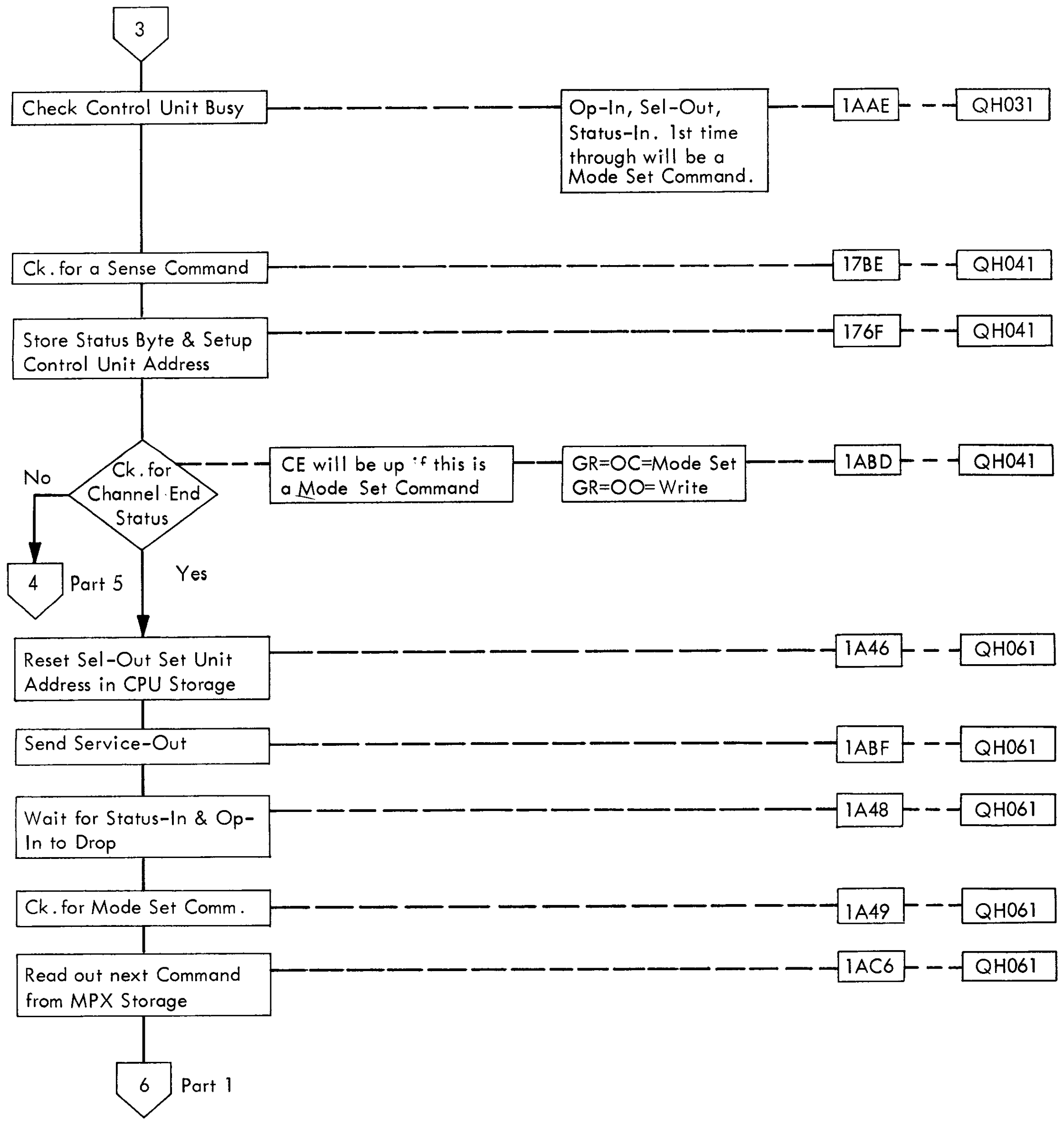
1400 COMPATIBILITY - TAPE SELECTOR SETUP - TAPE WRITE OP 9 TRACK DRIVE

444 M % U 1 500 W
452 B̄ 444
456 .̄



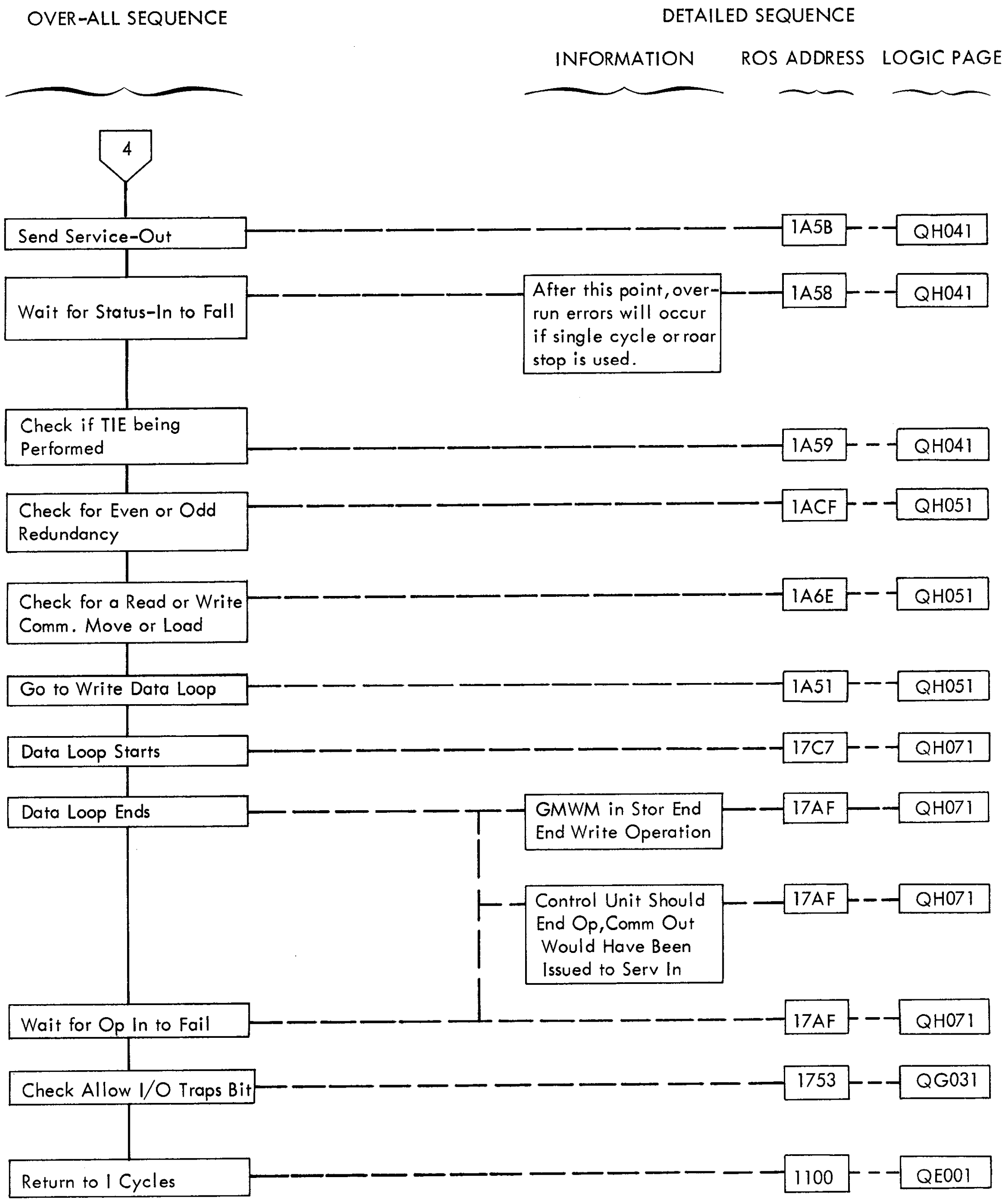
1400 Compatibility Tape Selector Setup-Tape Write Op 9 Track Drive RS Chart 13 (Sheet 1)





1400 Compatibility Tape Selector Setup-Tape Write Op 9 Track Drive RS Chart 13 (Sheet 2)

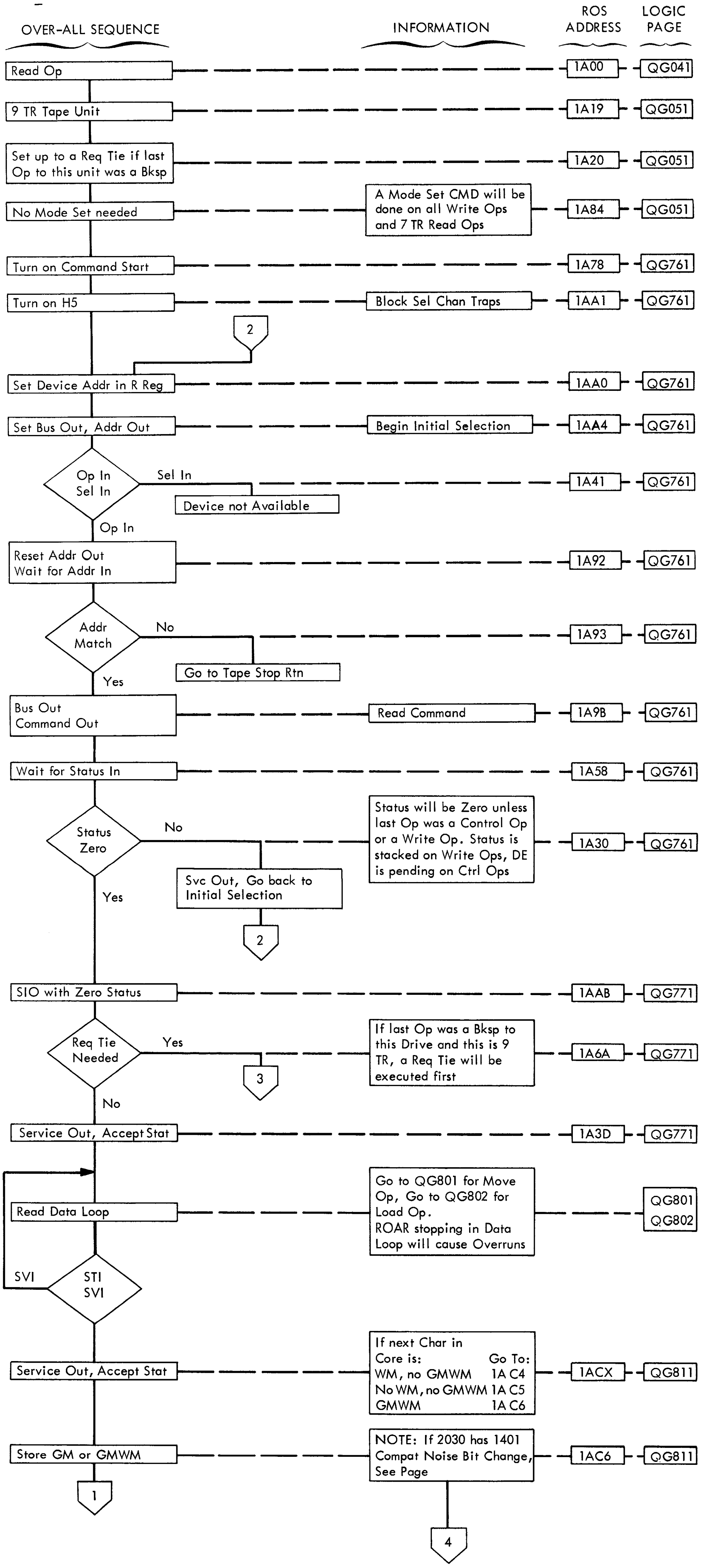
1400 COMPATIBILITY - TAPE SELECTOR SETUP - TAPE WRITE OP 9 TRACK DRIVE



1400 Compatibility Tape Selector Setup-Tape Write Op 9 Track Drive RS Chart 13 (Sheet 3)

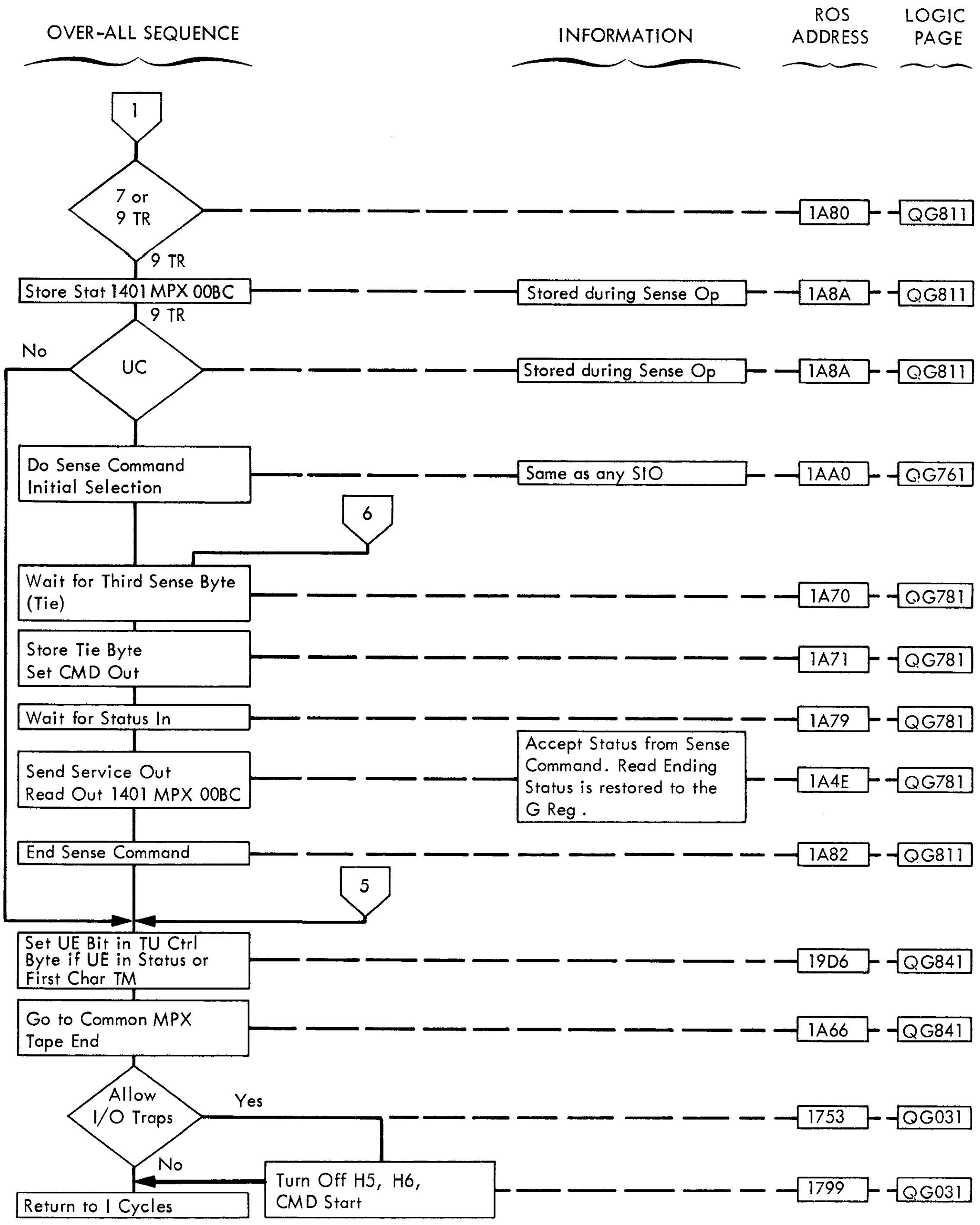
1400 COMPATIBILITY TAPE READ OPERATION TMPXR

```
444  M % U I 500R
452  B  417 L
457  B̄  400
461  Ū  % U I B
466  B̄  400
470  .̄
```



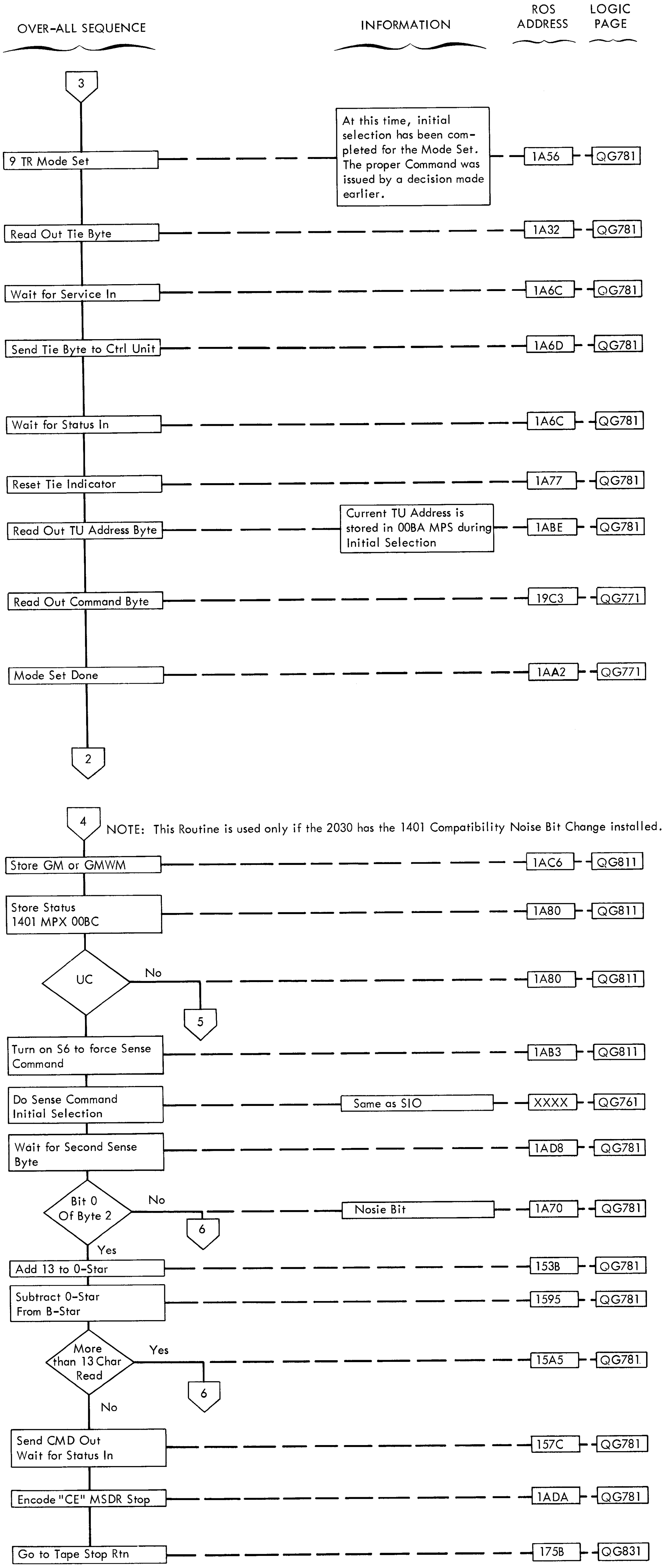
1400 Compatibility Tape Read Operation TMPXR RS Chart 14 (Sheet 1)

1400 COMPATIBILITY TAPE READ OPERATION TMPXR



1400 Compatibility Tape Read Operation TMPXR RS Chart 14 (Sheet 2)

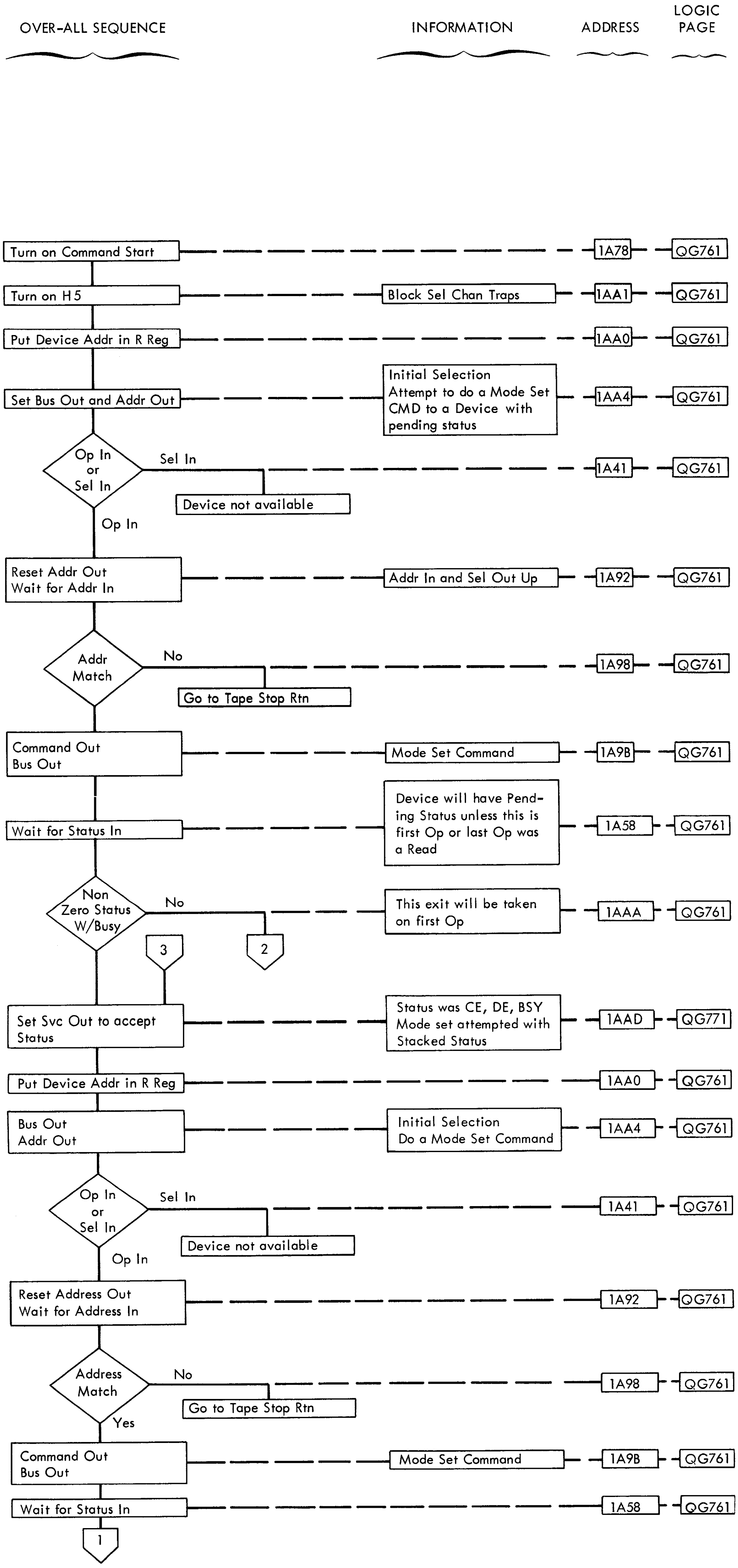
1400 COMPATIBILITY TAPE READ OPERATION TMPXR



1400 Compatibility Tape Read Operation TMPXR RS Chart 14 (Sheet 3)

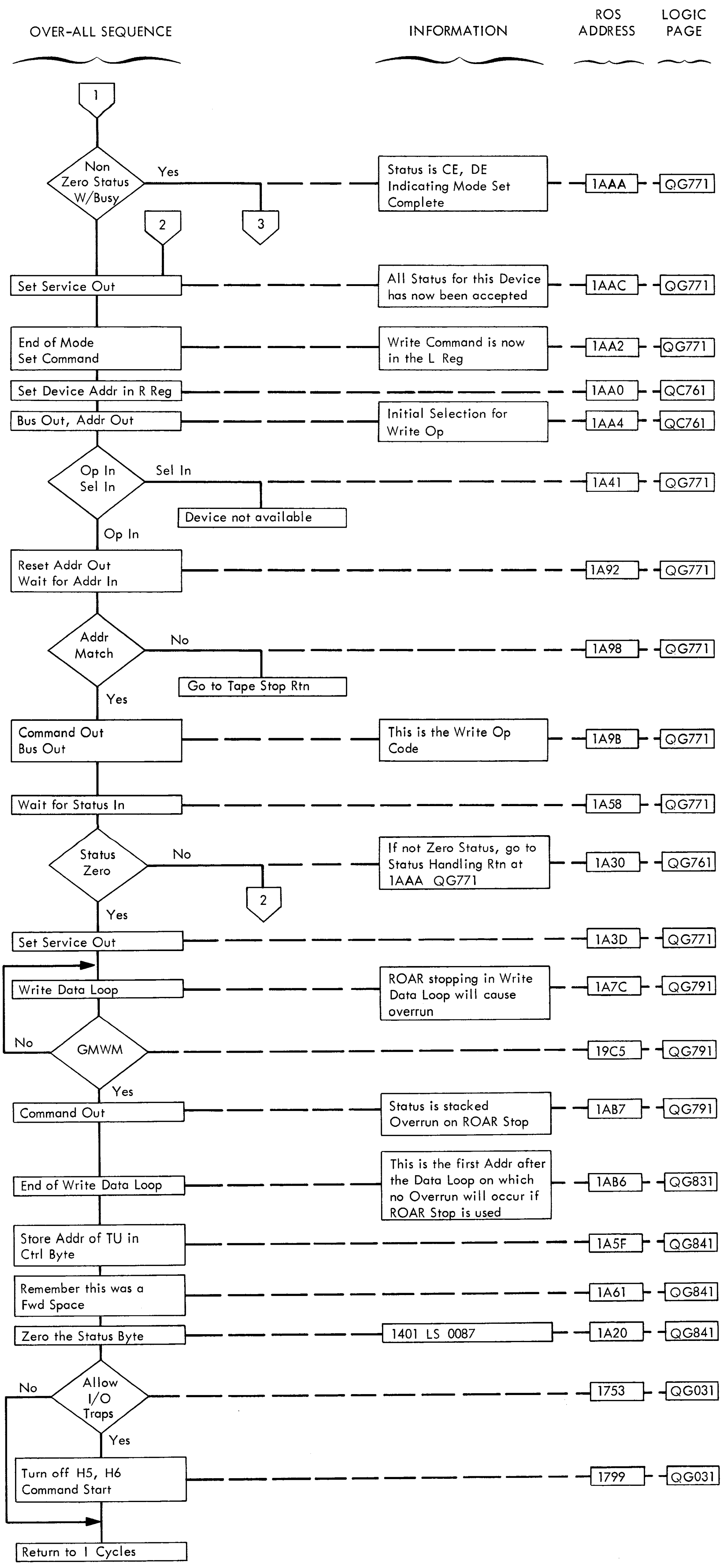
1400 COMPATIBILITY - TAPES ON MPX SET UP - TAPE WRITE OP 9 TRACK DRIVE

444 M % U 1 500W
452 B 400
456 .



1400 Compatibility-Tapes on MPX Set Up--Tape Write Op 9 Track Drive RS Chart 15 (Sheet 1)

1400 COMPATIBILITY TAPES ON MPX SET UP - TAPE WRITE OP 9 TRACK DRIVE



1400 Compatibility-Tapes on MPX Set Up--Tape Write Op 9 Track Drive RS Chart 15 (Sheet 2)

I 1400 FILE COMPATIBILITY

A. To allow proper operation of the included programs and stop words, the following initial preparation is necessary.

1. Initialize Disc Pack in 360 Mode writing Home addresses and record zero on all tracks using UT069.

   Surface analysis should also be done to flag defective tracks.

2. Load Auxiliary Storage using either CID or 3F00.

3. Write 1400 addresses with Clear Disc Program.

   Item 3 can be bypassed if 1400 Write Address Operation is to be performed first with Compare Disable Bit on. This will write addresses on a given track and allow other tests to be run on that track.

B. When restarting any of the example programs, a second error may occur if the following is not performed.

1. Do 1400 Start Reset

2. Start program at an instruction that will restore the DCF.

C. A Validity Check during Read or Write will break chaining. This will force a Sector Count 000 Indicator (Bit . 7 of Operation Register). The number of sectors not processed will be indicated in the 1400 DCF.

D. Sector Read and Write with addresses will post a No Address Compare (X) if the last Record Address is not 16 or greater. This test is made to check for missing 2311 address marks. (QH531, Word 1D1D).

   This may cause confusion if a Validity Check breaks chaining before 16 records are processed. Result is that X and V are posted when V is the actual cause.

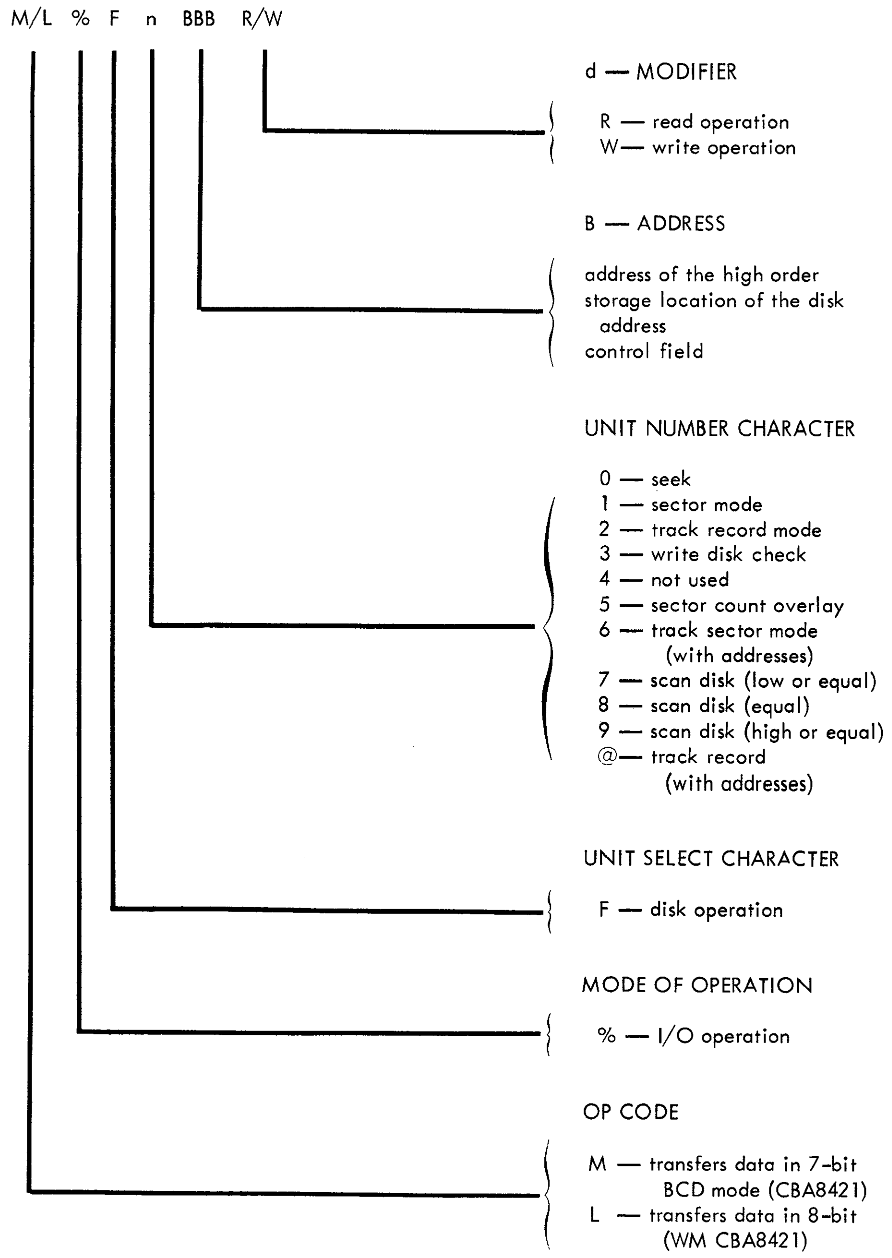
E. If an Alternate Track is assigned, the 2311 Carriage will be restored at the end of each completed 1400 Operation or 20 Sectors. This means that a RBC following a Write Operation will have to seek to the Alternate Track and return as did the Write Operation. Also, if a number of tracks are to be processed (sector count greater than 020) and more than one is flagged, the 2311 Carriage will be restored after each Alternate Track Read or Write is completed. Therefore, an Alternate Track Seek can occur more than once during a single 1400 Operation.

F. Auxiliary Storage B - 90, 92, 94, 96 and 98 (File Unit Addresses) must be in sequence when doing Address Operations or Alternate Track Operations or a No Address Compare will occur. The data in these addresses is always stored in K29 MPX on Initial Selection. When the above two operations are performed, a table look-up is done using the value in K29 MPX to determine the Auxiliary Storage Address that the actual cylinder value is stored in 91, 93, 95, 97 or 99. An Out of Sequence Unit Address will result in looking up cylinder value for the wrong file. This procedure is necessary because some program applications use other than normal addresses on certain cylinders, such as labels. If addresses are abnormal, it becomes impossible to read 360 Record 0 or return to the proper track.

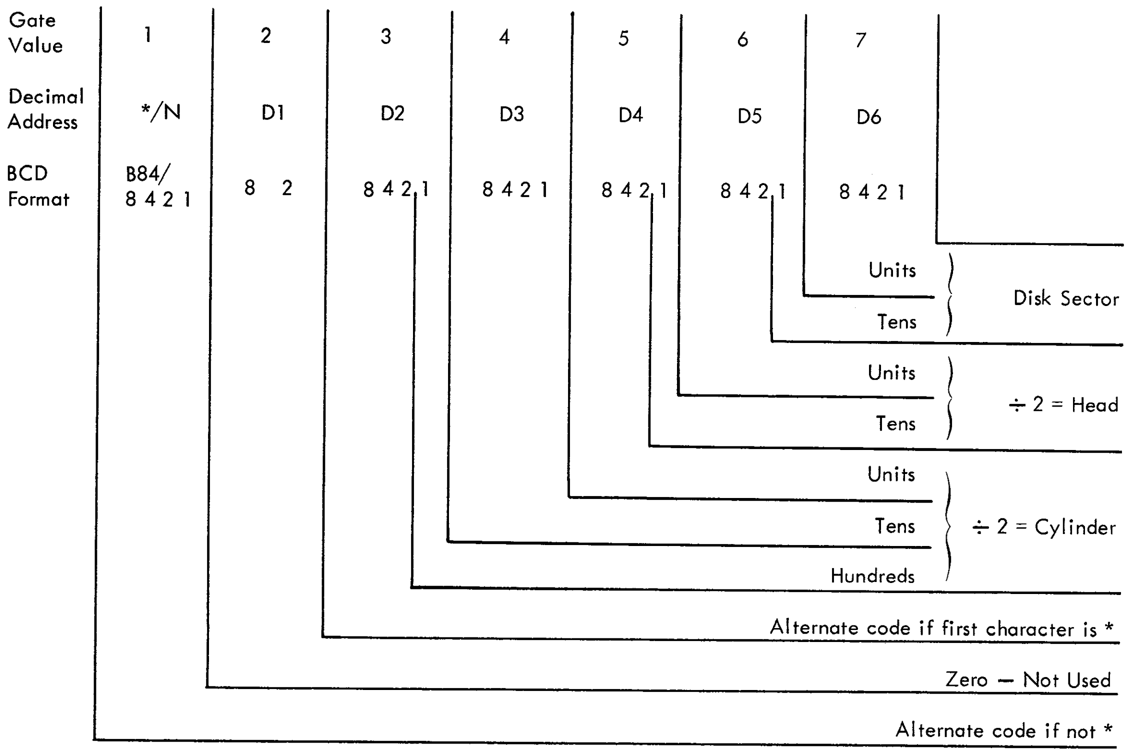
G. Module Protect Check is defined on Head Seek flow chart.

H. STOP CHECK is indicated on several flow charts. This is the first stop point since 1BB0 of the search command that will allow continuation of the operation without an overrun. Check FI = CE• DE, Op In and Stat In up, command code in Aux B - BE, updated CCHHR stored in Aux B - AD thru B1.

1311 DISK INSTRUCTION FORMAT



1311 ADDRESS CONTROL FIELD



PERMANENT FILE AUXILIARY STORAGE LOCATIONS

AUXILIARY STORAGE B

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	TLU for Head and Cylinder 10's Decode															
	00	05	01	06	02	07	03	08	04	09						
1	Note 1	File Mod 00	File Mod 02	File Mod 04	File Mod 06	File Mod 08										
8	0	Sense 1	Bytes 2	3												
9	File Unit 0		File Unit 1		File Unit 2		File Unit 3		File Unit 4		Note 2					
	Addr 00	Cyl Loc	Addr 01	Cyl Loc	Addr 02	Cyl Loc	Addr 03	3 Cyl Loc	Addr 04	Cyl Loc						
A	TLU for Cylinder 100's Decode										Pre-vious File Op			Always 00		
	Numeric															
	00	50	10	60	20	70	30	80	40	90						
	Hex															
	00	32	0A	3C	14	46	1E	50	28	5A				Cyl 00	Cyl	Head 00
B	Head	Re-cord	K	D	D				Note 5	1400 File Mod No.				File Unit Addr Init Sel	Last File Comm and	Scan Cond Note 3

AUXILIARY STORAGE A

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
9										Scan Re-sult Note 3		Note 4				
A	Alternate Track Addr Read From R0 of Flagged Track															
	Cyl	Cyl	Head	Head												

NOTE 1

BIT 0 – COMP DISABLE
1 – MOD PROTECT

NOTE 2

BIT 3 = 1 = DCF HAS INCREMENTED

NOTE 3

K31 UCW SCAN	OP WT	K9H CPU SCAN	RESULT WT
SCAN LO 7	0	HI	12
	0	LO	6
	0	EQ	1
SCAN EQ 8	1	HI	12
	1	LO	6
	1	EQ	1
SCAN HI 9	2	HI	12
	2	LO	6
	2	EQ	1

NOTE 4

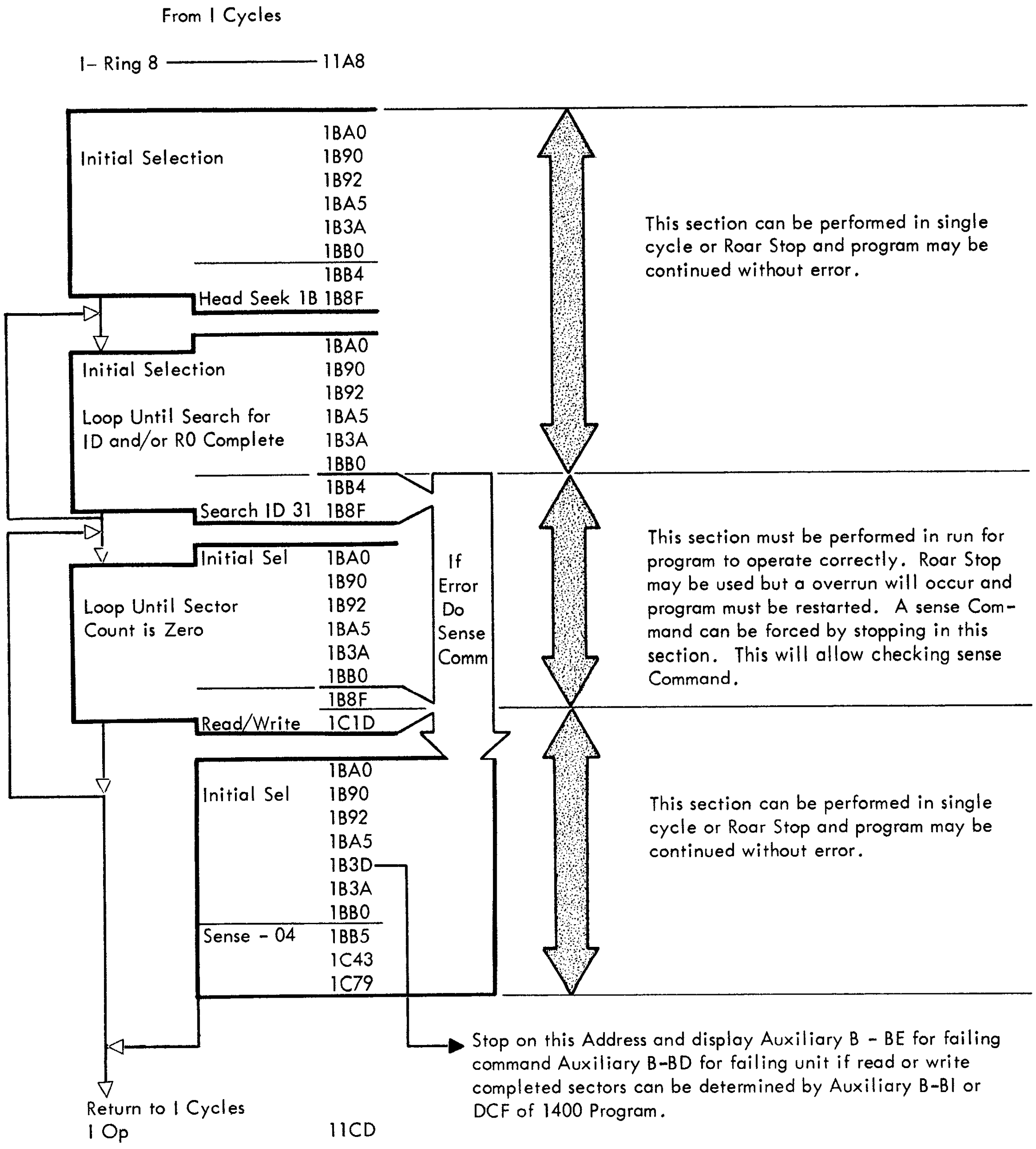
1400 BR BYTE
0 - NO ADDR COMP
1 - BUSY
2 - WLR
3 - ANY
4 - PARITY
5 - NOT RDY
6 - RBC
7 - ON-Checked for Recalibrate Seek Necessary

NOTE 5

On alternate track operation, original cylinder value in AE(K22) is stored here for return seek

FILE READ/WRITE GENERAL ADDRESS FLOW

Details on the following address are on the individual charts for each operation.



This section can be performed in single cycle or Roar Stop and program may be continued without error.

This section must be performed in run for program to operate correctly. Roar Stop may be used but a overrun will occur and program must be restarted. A sense Command can be forced by stopping in this section. This will allow checking sense Command.

This section can be performed in single cycle or Roar Stop and program may be continued without error.

Stop on this Address and display Auxiliary B - BE for failing command Auxiliary B-BD for failing unit if read or write completed sectors can be determined by Auxiliary B-BI or DCF of 1400 Program.

II SECTOR OPERATIONS IN COMPATIBILITY MODE

Note the following characteristics of Sector OPs in compatibility mode:

1. Sector address is incremented in 360 CCHHR not 1400 DCF.
2. Updated CCHHR is not converted and stored in DCF until sector count 000 is detected.
3. Sector count is decremented in 1400 DCF.
4. In event of error sector count could appear decreased and sector address may not be increased in the DCF. CCHHR in aux. B - AD thru B1 should be checked for actual sector address value.
5. DCF incremented bit - aux. B-9 A-Bit 3
   Bit 3 = 1 indicates CCHHR has been incremented (sector count more than one) therefore CCHHR must be converted 1400 format and zone bits stripped.
6. Zone bits will remain in DCF only if initial sector count is 001.

III SECTOR OPERATIONS - READ/WRITE/RBC

The following programs should be used when possible as they are tested and provide predictable results:

<u>Read Op</u>

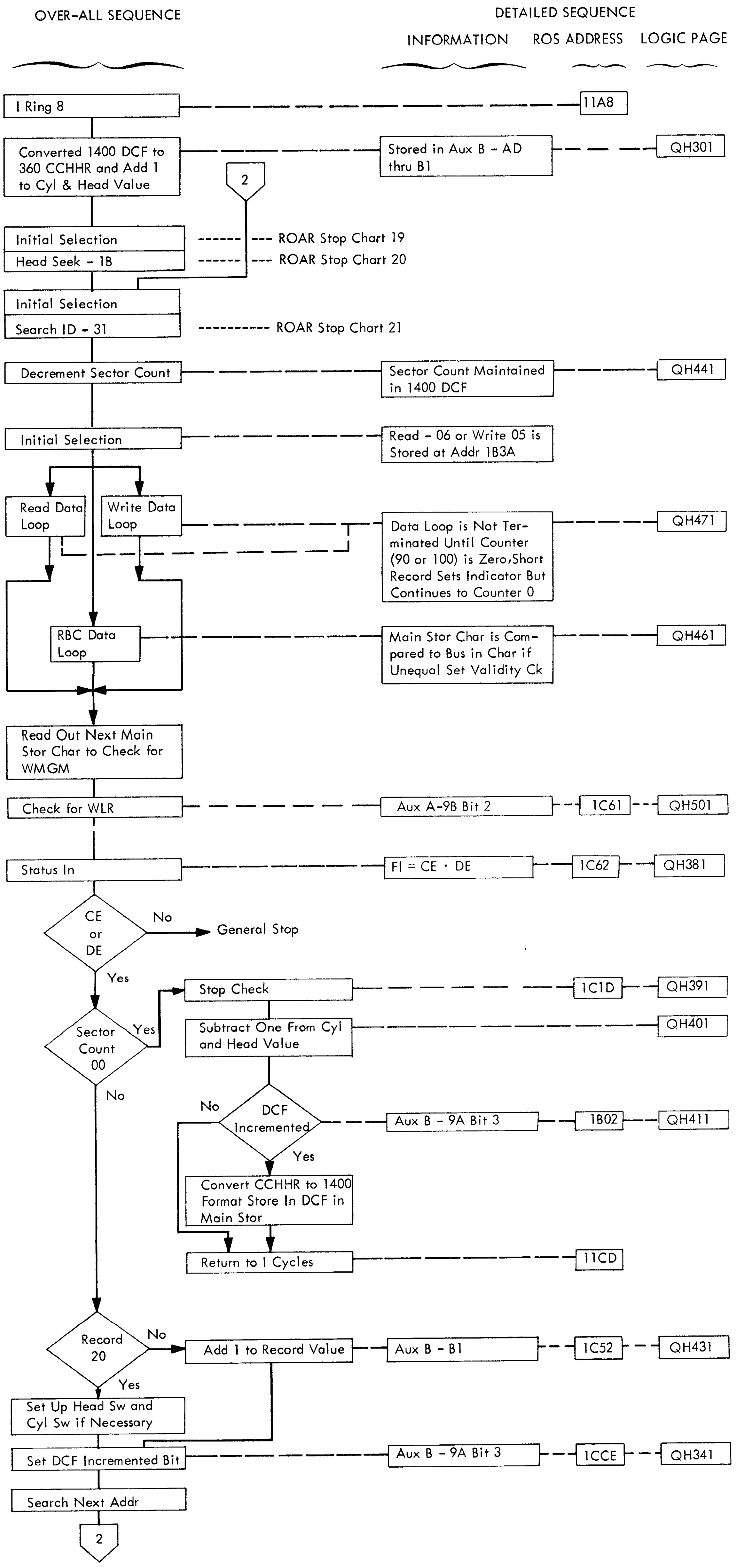
```
500 M % F 0 800 R         Seek
508 M % F 1 800 R         Read
516 M 709 809             Restore DCF
523 B 532 Y               Branch on any condition
528 B 508                 Loop
532 . 500
536 .
```

<u>Write Op</u>

```
500 M % F 0 800 R         Seek
508 M % F 1 800 W         Write
516 M 709 809             Restore DCF
523 M % F 3 800 R         RBC
531 M 709 809             Restore DCF
538 B 557 Y               Branch on any condition
543 B 508                 Loop
547 . 500
551 .
```

```
700 - * 010000 001        Can be increased from 1
800 - * 010000 001        thru 20 depending on the
910 - ‡                   number of sector opera-
                          tions desired.
```

Add 100 addresses for each sector over one (1) to be processed.



1400 Sector Read/Write/RBC RS Chart 17 (Sheet 2)

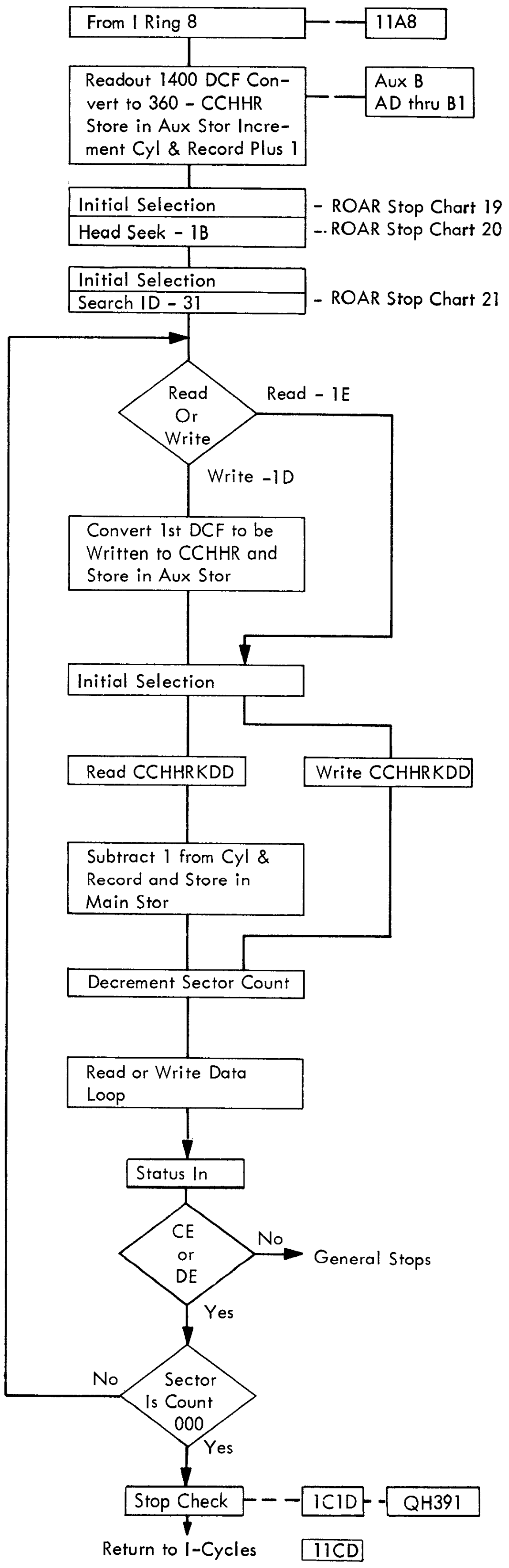
PROGRAM EXAMPLE FOR SECTOR WRITE WITH ADDRESSES (Write 20 Sectors with Addresses)

```
 500 M % F 0 800 R       Seek to Cyl 050 (1400)
 508 M % F 6 800 W       Write
 516 M 709 809           Restore DCF
 523 M % F 3 800 R       RBC
 531 M 709 809           Restore DCF
 538 B 547 Y             Branch on any Condition
 543 B 508               Loop
 547 . 500               Stop/Seek Restore after
                          Error
 551 .
 700 * 01 0000 020
 800 * 01 0000 020 01 0000
2824 0 10 019
2930 ±
```

PROGRAM EXAMPLE FOR SECTOR READ WITH ADDRESSES

```
 500 M % F 0 800 R       Seek
 508 M % F 6 800 R       Read
 516 M 709 809           Restore DCF
 523 B 532 Y             Branch on any Condition
 528 B 508               Loop
 532 . 500               Stop/Seek Restore after
                          Error
 536 .
 700 * 01 0000 020
 800 * 01 0000 020
2930 ±
```

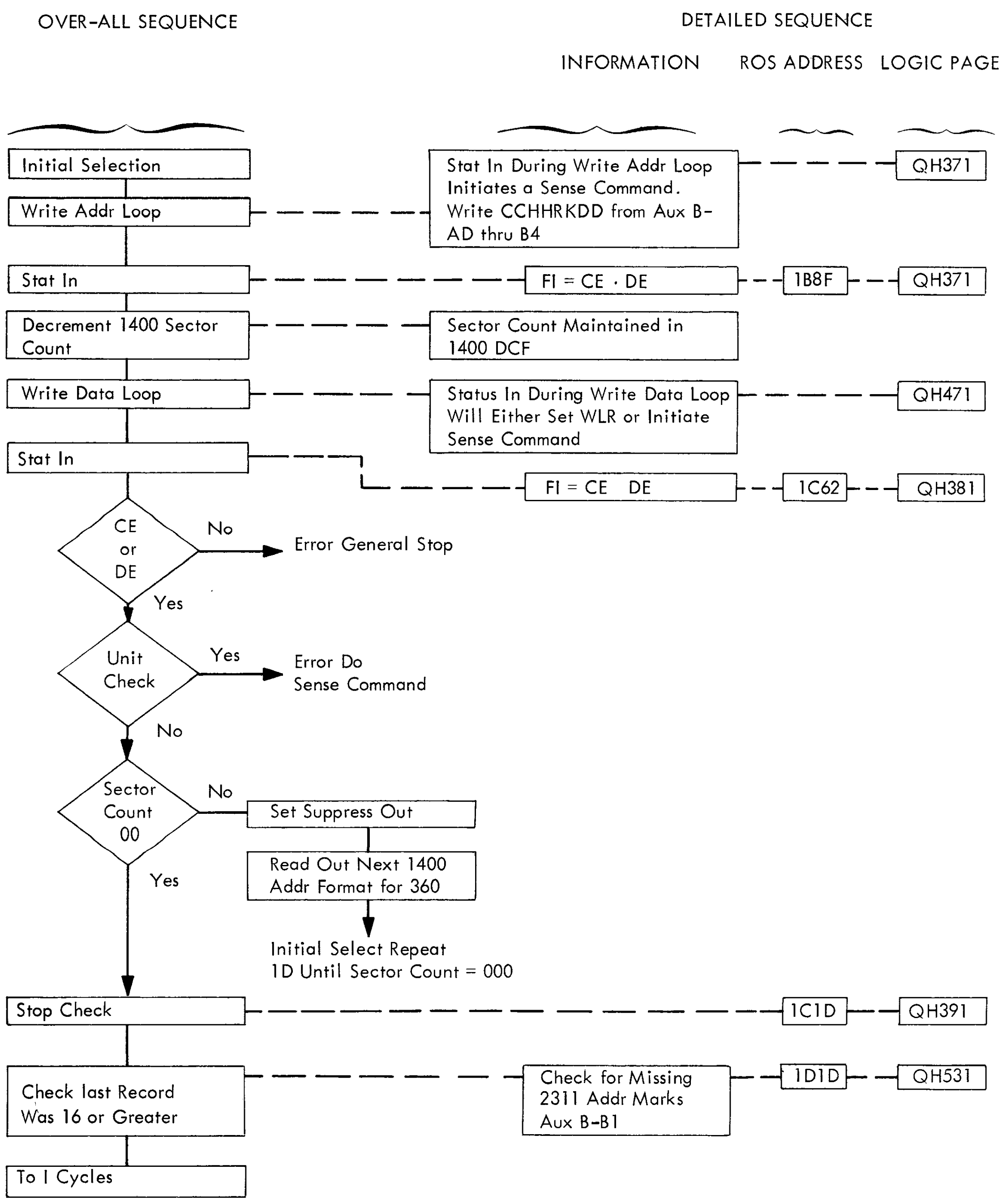
SECTOR READ WRITE WITH ADDRESSES - OBJECTIVES



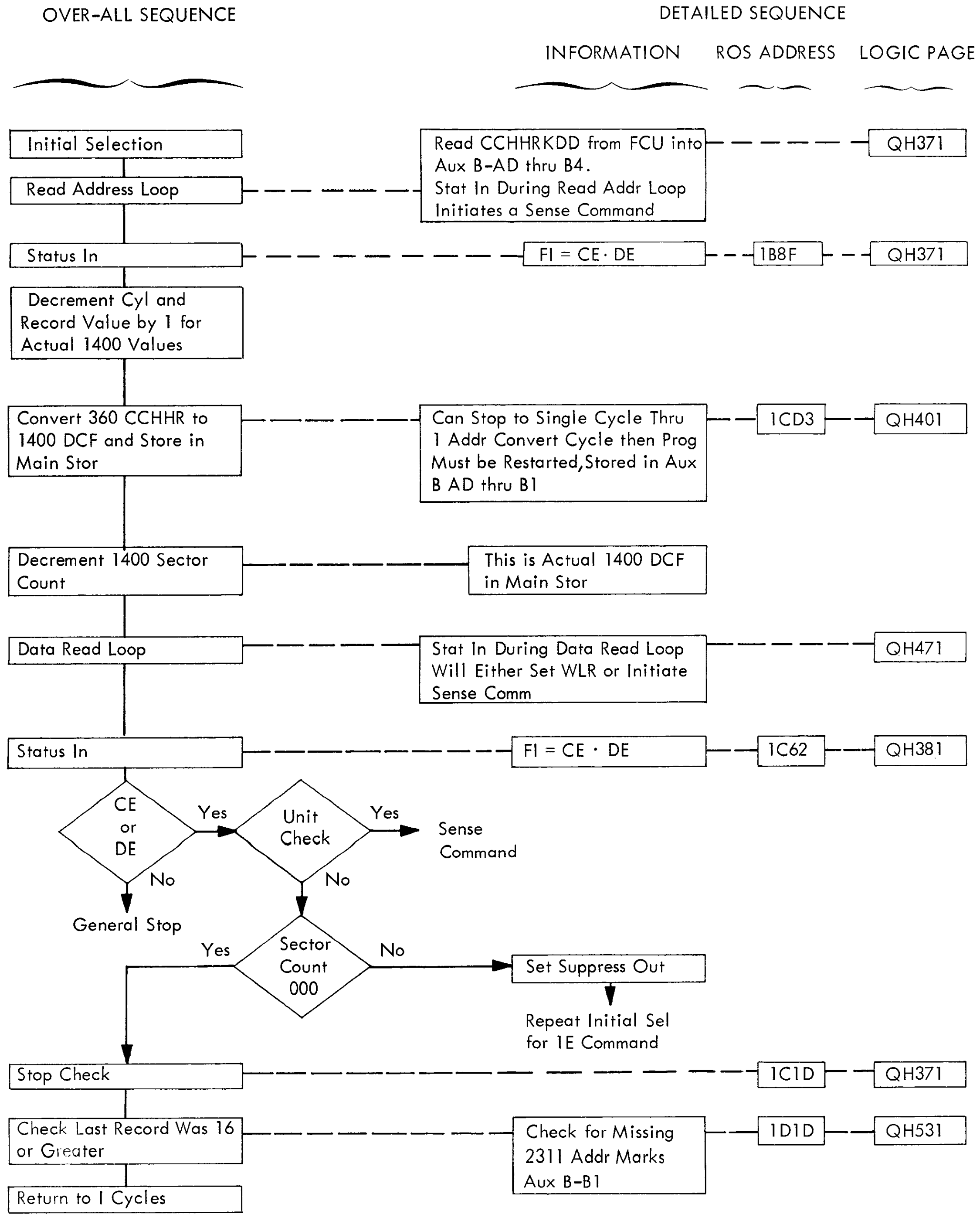
1400 Sector Read/Write/With Address RS Chart 18 (Sheet 1)

WRITE ADDRESS OPERATION 1D

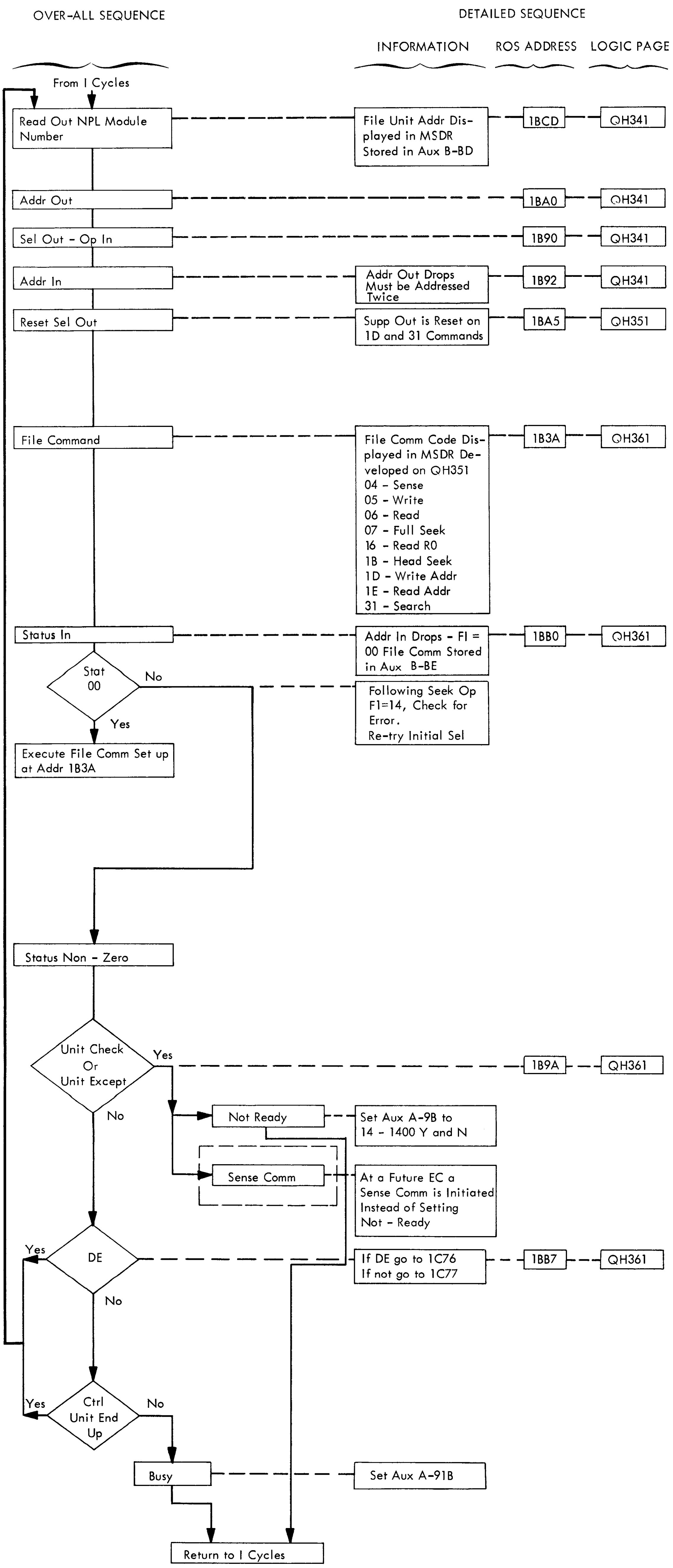
Initial File Address to be Written Has Been Developed with the Cylinder and Record Count Incremented Plus 1 to Conform to 360 Format and Stored as CCHHRKDD in Aux B AD thru B4.



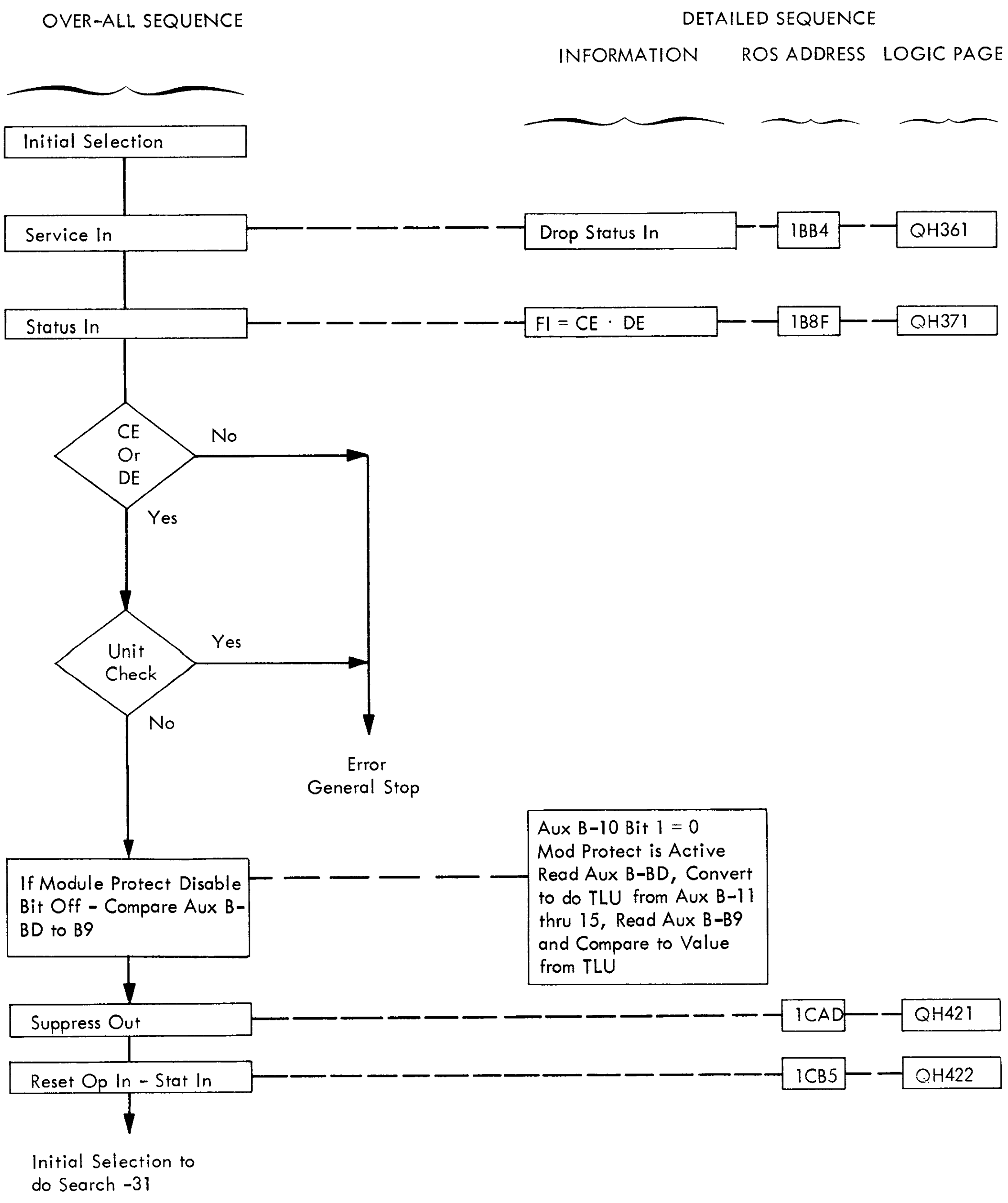
READ ADDRESS OPERATION 1E



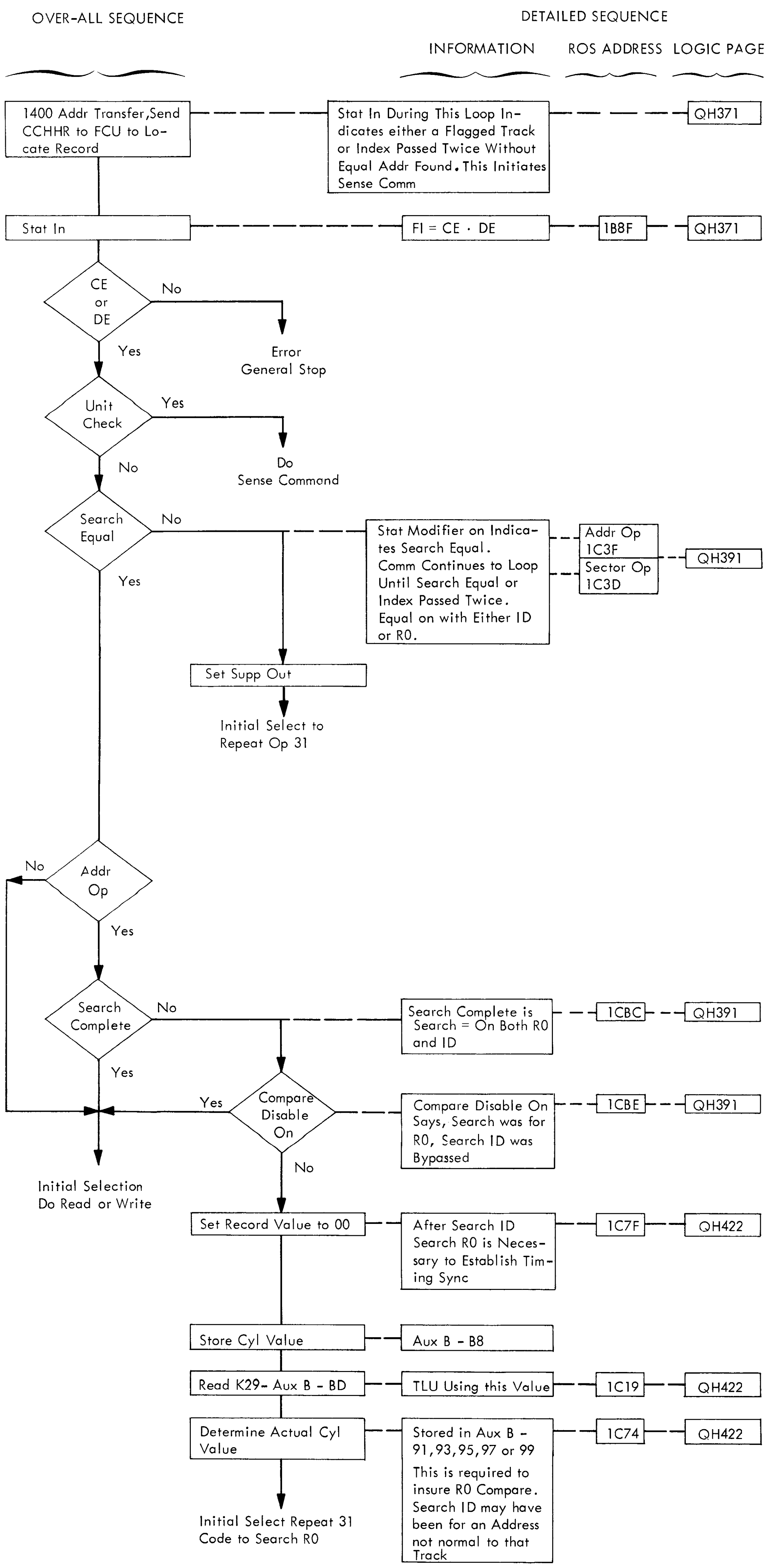
1400 Sector Read/Write/With Address RS Chart 18 (Sheet 2)



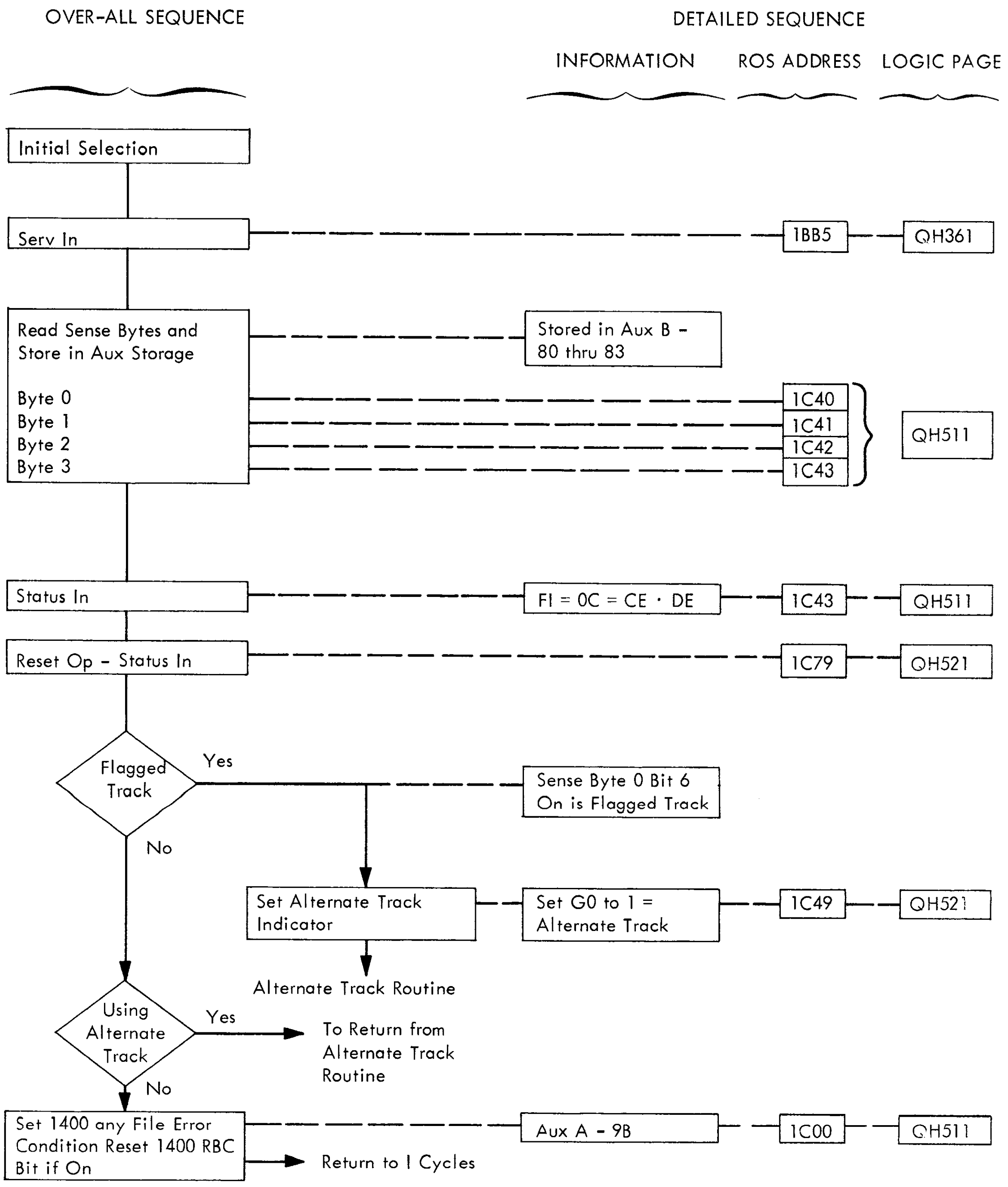
Initial Selection File Commands RS Chart 19 (Sheet 1)



Head Seek 1B RS Chart 20 (Sheet 1)



Search ID--31 Command RS Chart 21 (Sheet 1)


OVER-ALL SEQUENCE
DETAILED SEQUENCE
INFORMATION
ROS ADDRESS
LOGIC PAGE
Initial Selection
Serv In
1BB5
QH361
Read Sense Bytes and Store in Aux Storage
Stored in Aux B - 80 thru 83
Byte 0
Byte 1
Byte 2
Byte 3
1C40
1C41
1C42
1C43
QH511
Status In
FI = 0C = CE · DE
1C43
QH511
Reset Op - Status In
1C79
QH521
Flagged Track
Yes
No
Sense Byte 0 Bit 6 On is Flagged Track
Set Alternate Track Indicator
Set G0 to 1 = Alternate Track
1C49
QH521
Alternate Track Routine
Using Alternate Track
Yes
No
To Return from Alternate Track Routine
Set 1400 any File Error Condition Reset 1400 RBC Bit if On
Aux A - 9B
1C00
QH511
Return to I Cycles

IV SEEK OPERATION

A. Initialize Auxiliary Storage

B. Enter following program

C Ready the 1st File

D. 1400 Start Reset

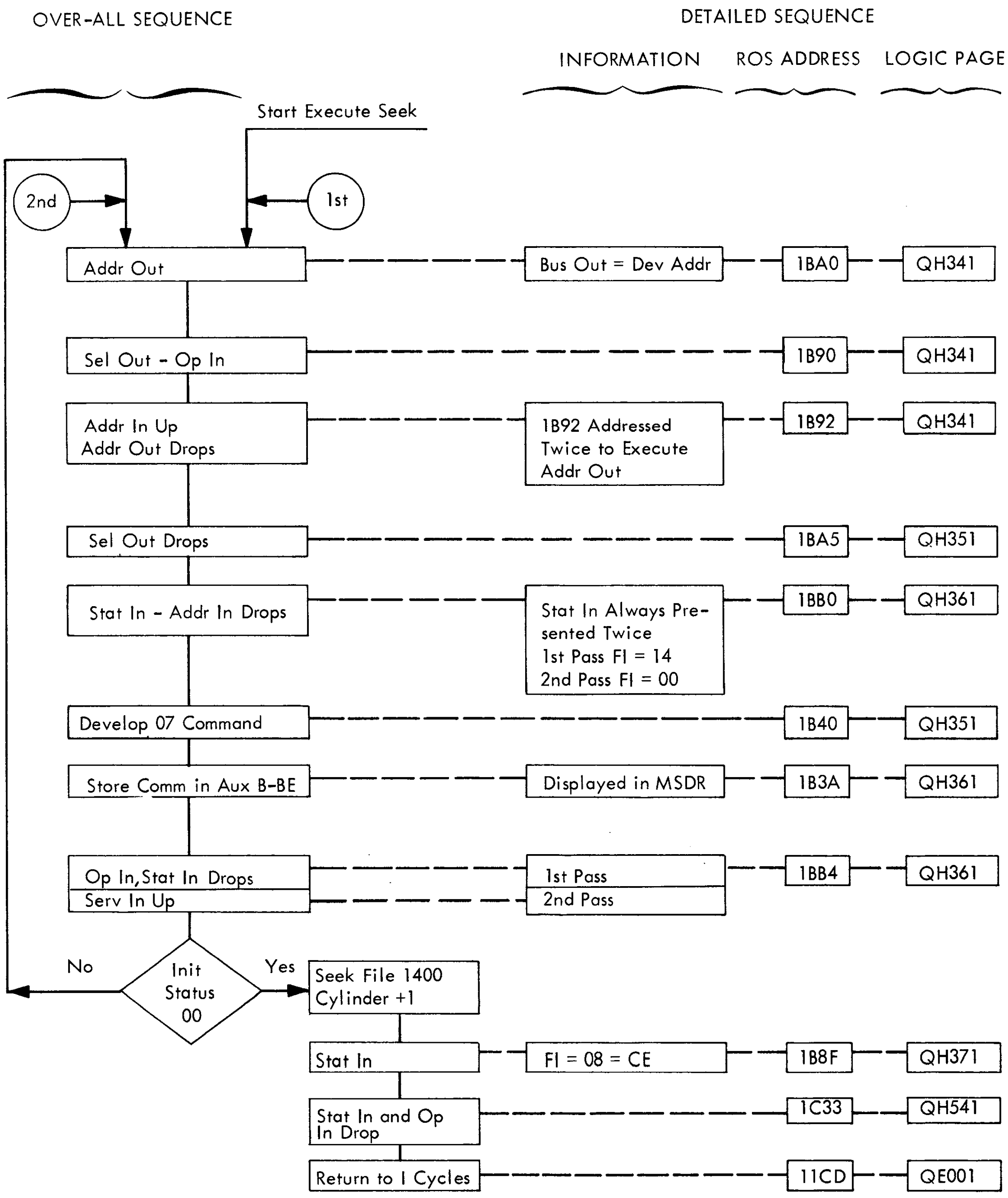
E. Start Program (Set 1C to 500)

F. Execute program in single cycle or Roar Stop on address provided in flow chart for Seek Operation.

This program should do repetitive seeks between 1400 cylinders 000 and 050. The actual physical location on the Disc Pack are cylinders 001 and 051. The compatibility feature increments all 1400 cylinders plus one because cylinder 000 is reserved for 360 use.

```
500   M % F 0 600 R    Seek to Cylinder 00
508   B 5 0 0 \        Branch on Busy
513   M % F 0 700 R    Seek to Cylinder 50
521   B 5 1 3 \        Branch on Busy
526   B 5 0 0 .        Return to Seek
600   * 0 0 0 0 0 0
700   * 0 1 0 0 0 0
```

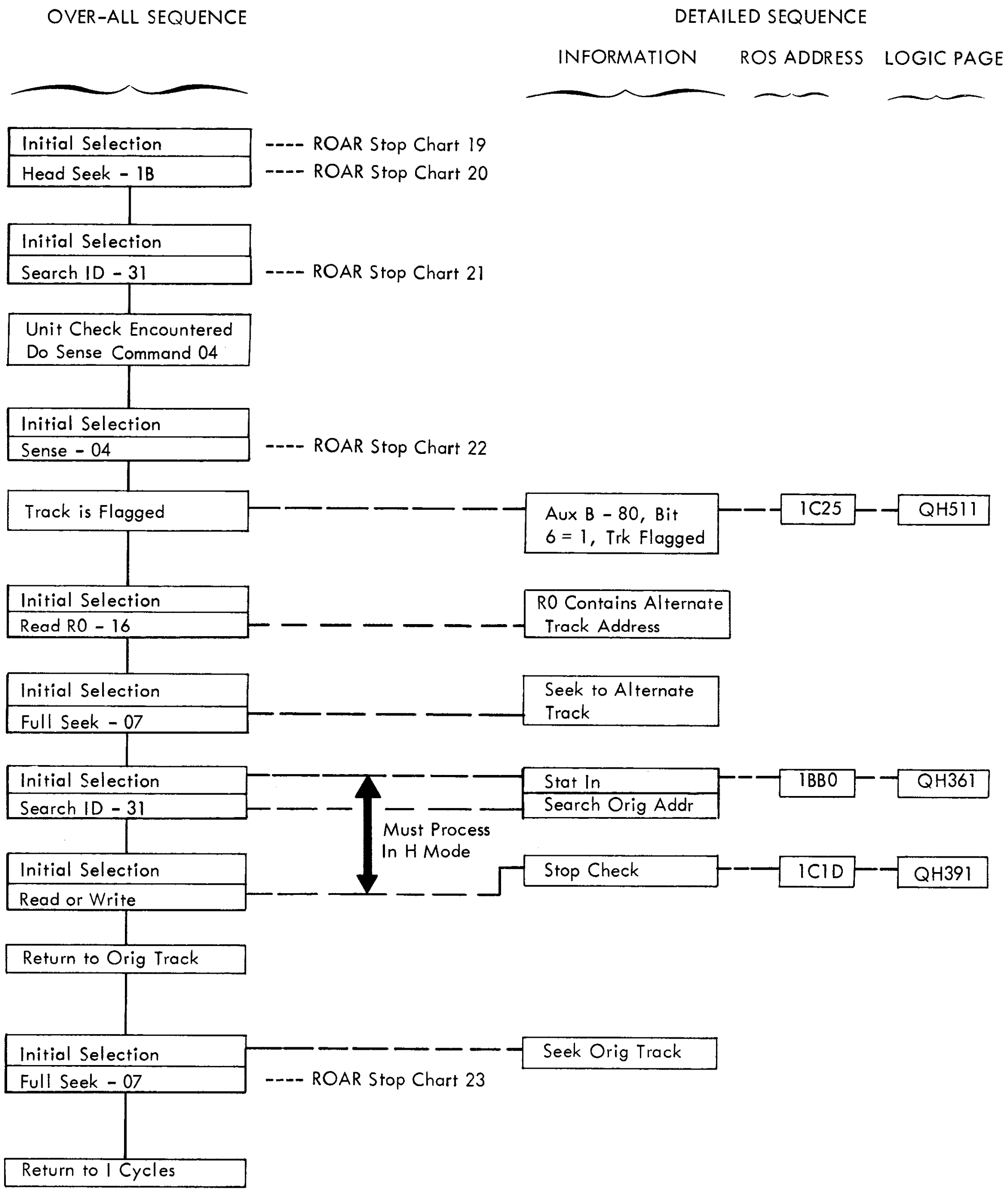
SEEK OPERATION -07



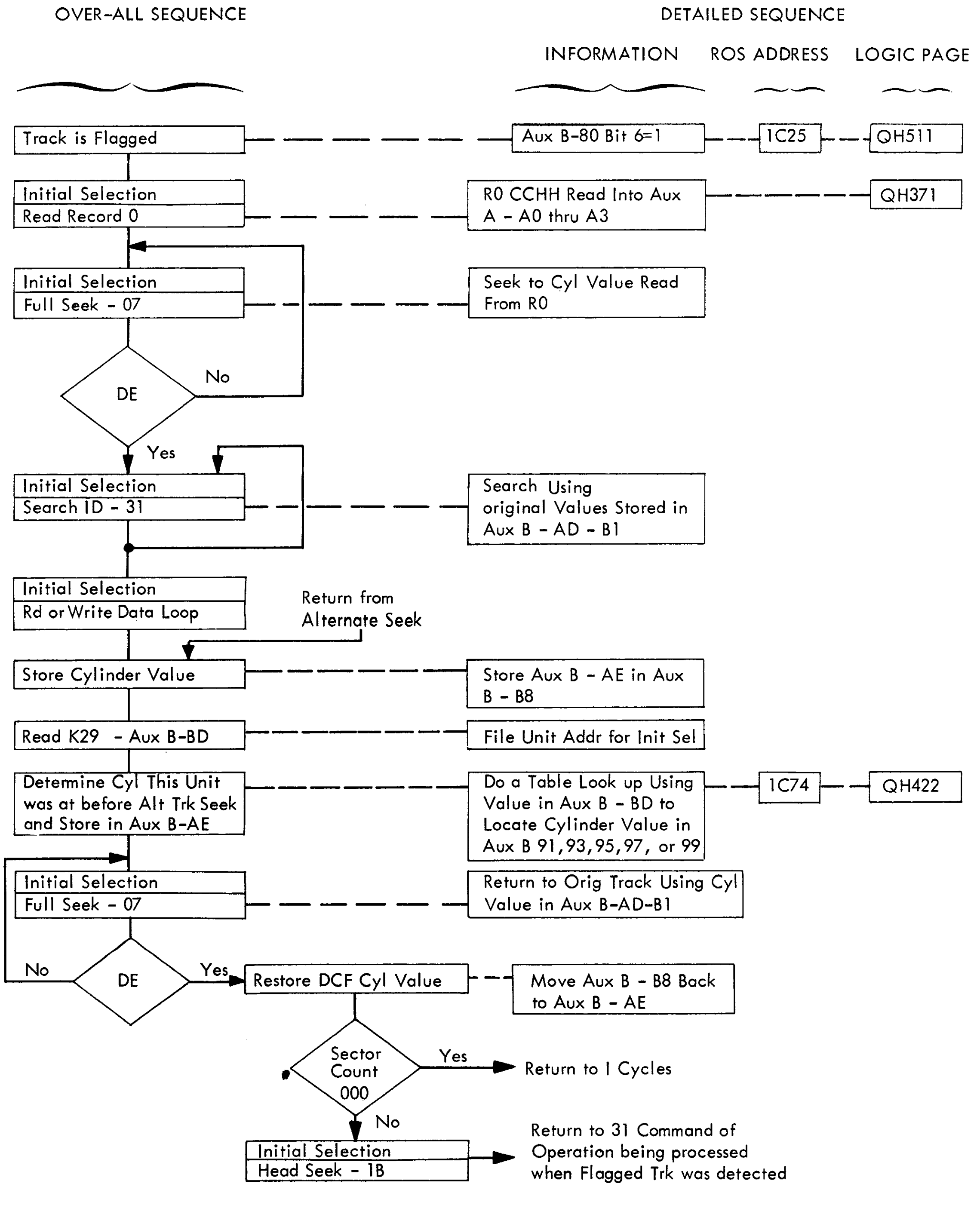
Seek Operation RS Chart 23 (Sheet 1)

ALTERNATE TRACK ENTRY AND EXIT OBJECTIVE

Refer to Individual Flow Charts for Details on the Following Command Codes.



ALTERNATE TRACK ROUTINE



Alternate Track Entry and Exit Objective RS Chart 24 (Sheet 1)

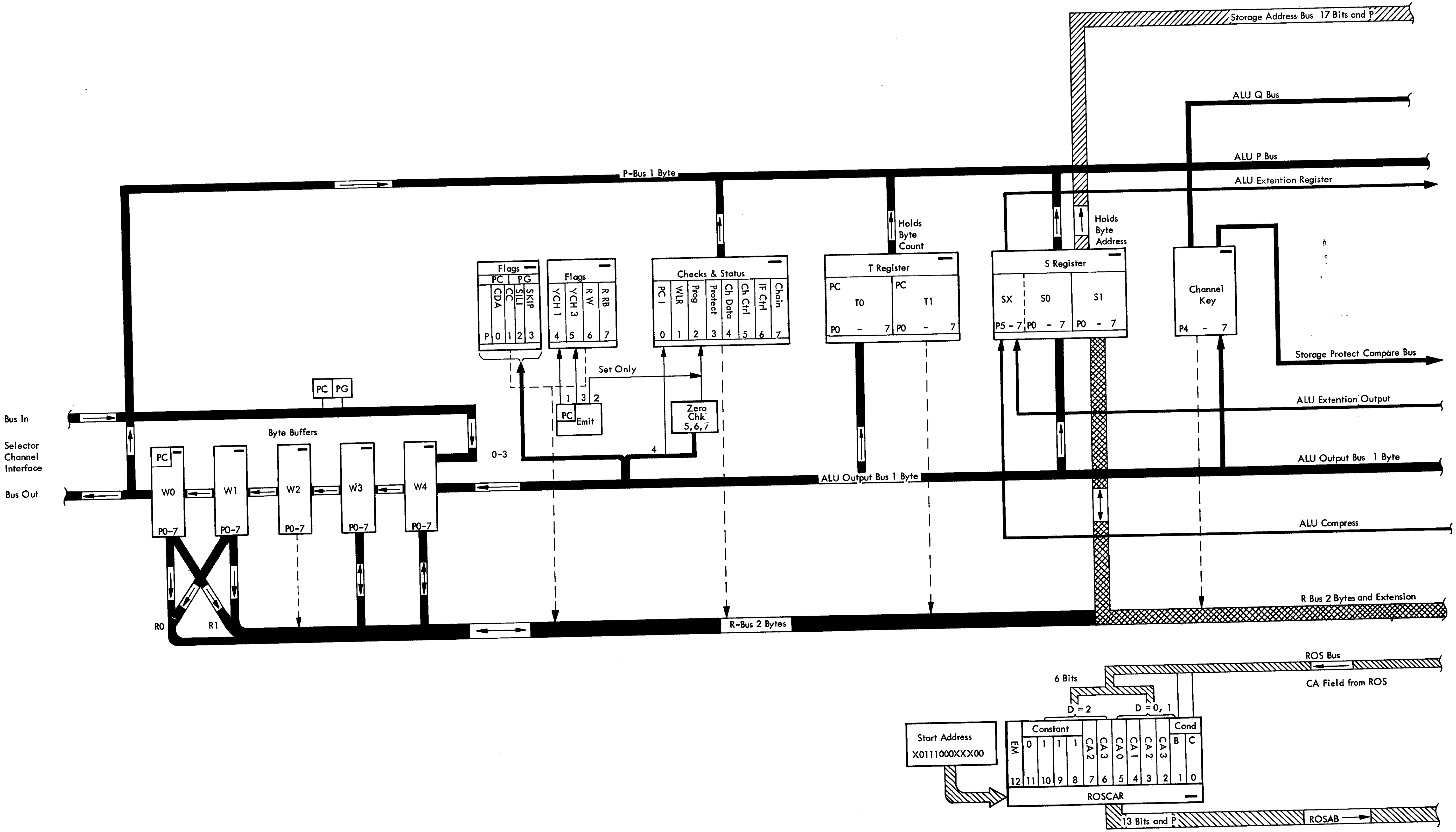


FIGURE 011. SELECTOR CHANNEL DATA FLOW

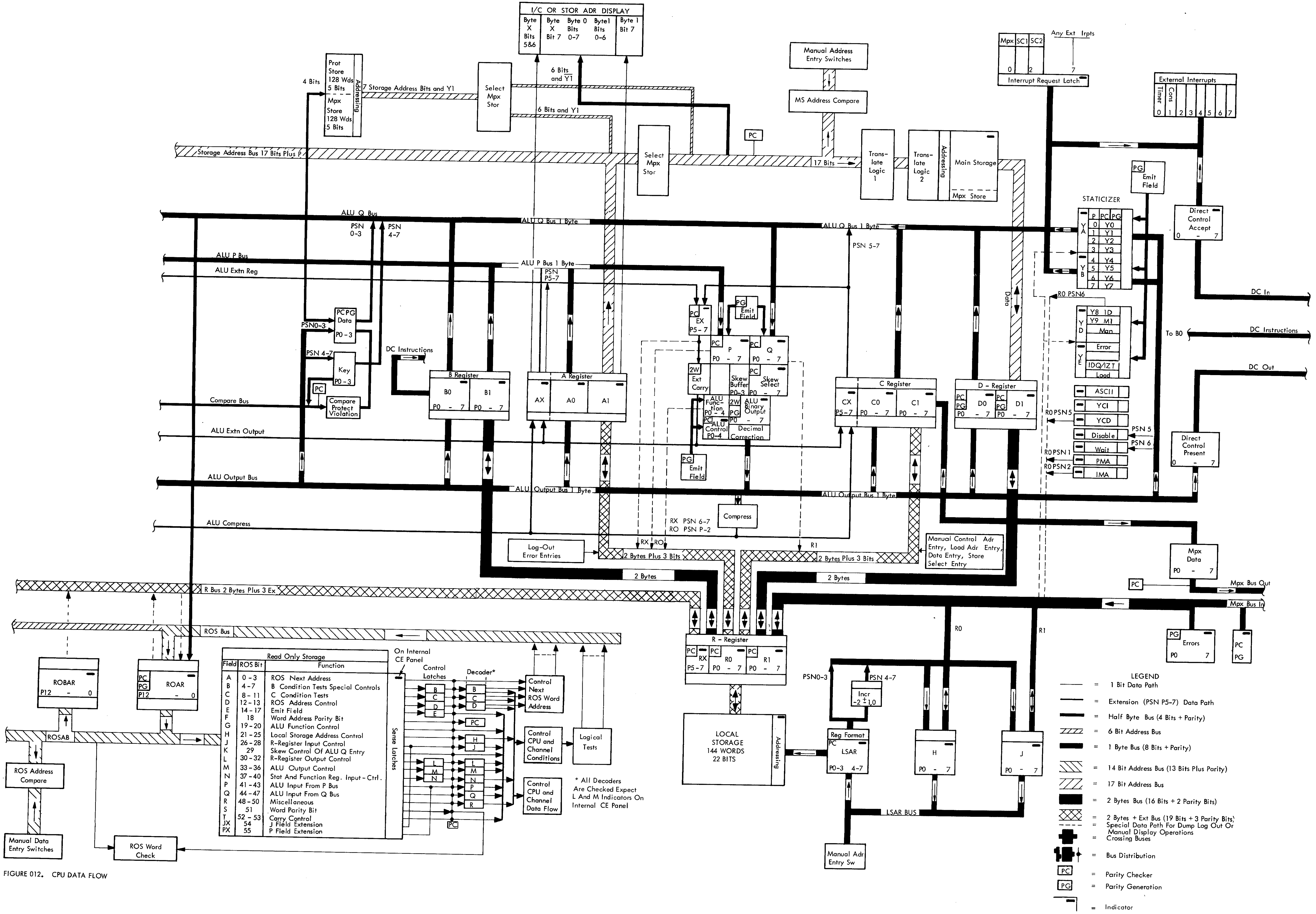
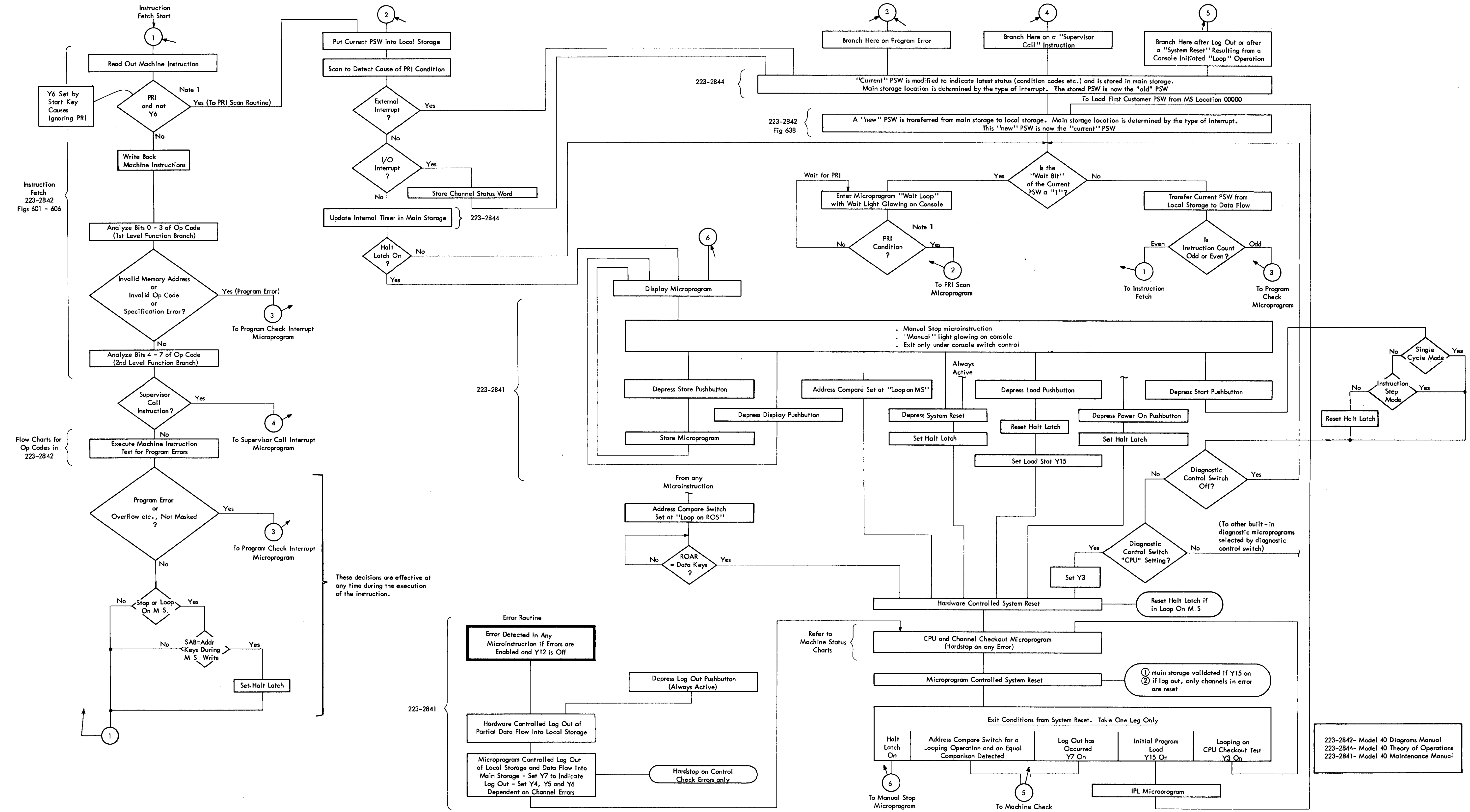


FIGURE 012. CPU DATA FLOW



Note 1. Conditions that Cause Program Interrupt (PRI)

① Channel interrupt if masked to allow

② 4 bit timer nonzero (cancelled by disable interval timer switch)

③ Halt latch on
- a. Stop key
- b. Instruction step mode
- c. Stop on MS } via address compare switch
- d. Loop on MS } via address compare switch
- e. System reset pushbutton
- f. Power on Pushbutton

④ External interrupts masked to allow
- a. Console attention (interrupt pushbutton on console depressed)
- b. Interval timer in main storage has timed out
- c. External interrupts 3 through 7

223-2844

FIGURE 014. CPU MICROPROGRAM FLOW CHART (CHANNEL DATA SERVICE NOT SHOWN)

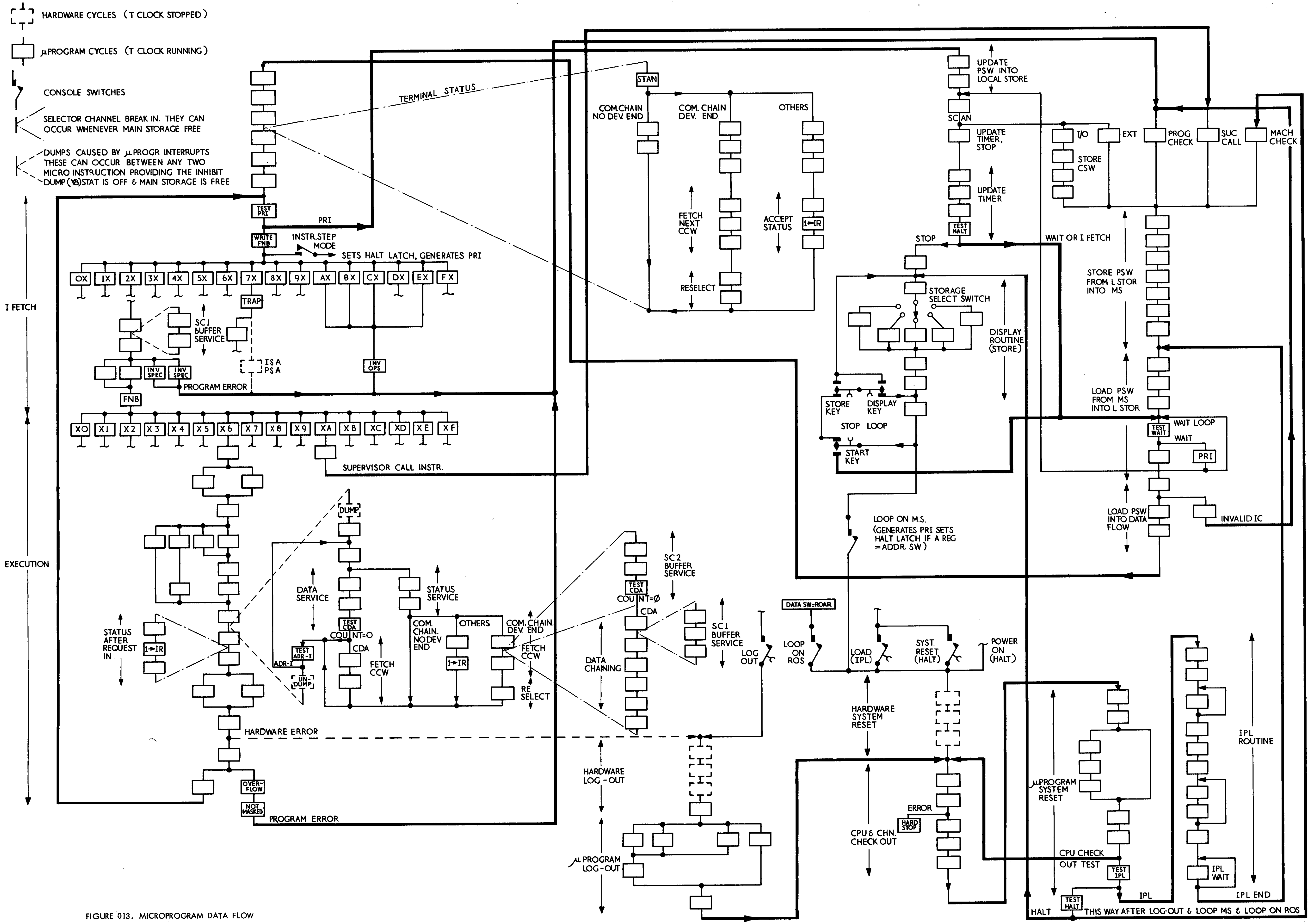


FIGURE 013. MICROPROGRAM DATA FLOW

Index

READER'S COMMENT FORM

IBM 2030 Processing Unit System/360 Model 30
Field Engineering Diagram Manual Y24-3466-2

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. Comments and suggestions become the property of IBM.

	Yes	No
• Does this publication meet your needs?	☐	☐
• Did you find the material:		
Easy to read and understand?	☐	☐
Organized for convenient use?	☐	☐
Complete?	☐	☐
Well illustrated?	☐	☐
Written for your technical level?	☐	☐

- What is your occupation? ____________________
- How do you use this publication?
  - As an introduction to the subject? ☐
  - For advanced knowledge of the subject? ☐
  - For information about operating procedures? ☐
  - As an instructor in a class? ☐
  - As a student in a class? ☐
  - As a reference manual? ☐

  Other ____________________
- Please give specific page and line references with your comments when appropriate. If you wish a reply, be sure to include your name and address.

COMMENTS:

- Thank you for your cooperation. No postage necessary if mailed in the U. S. A.

READER'S COMMENT FORM

IBM 2030 Processing Unit System/360 Model 30
Field Engineering Diagram Manual Y24-3466-2

- Your comments, accompanied by answers to the following questions, help us produce better publications for your use. If your answer to a question is "No" or requires qualification, please explain in the space provided below. Comments and suggestions become the property of IBM.

	Yes	No
• Does this publication meet your needs?	☐	☐
• Did you find the material:		
Easy to read and understand?	☐	☐
Organized for convenient use?	☐	☐
Complete?	☐	☐
Well illustrated?	☐	☐
Written for your technical level?	☐	☐

- What is your occupation? ____________________
- How do you use this publication?
  - As an introduction to the subject? ☐
  - For advanced knowledge of the subject? ☐
  - For information about operating procedures? ☐
  - As an instructor in a class? ☐
  - As a student in a class? ☐
  - As a reference manual? ☐

  Other ____________________
- Please give specific page and line references with your comments when appropriate. If you wish a reply, be sure to include your name and address.

COMMENTS:

- Thank you for your cooperation. No postage necessary if mailed in the U. S. A.

Staple

Fold

Fold

FIRST CLASS
PERMIT NO. 170
ENDICOTT, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
P. O. Box 6
Endicott, N. Y. 13760

Attention: Product Publications, Dept. 171

Fold

Fold

Cut Along Line

International Business Machines Corporation
Field Engineering Division
[illegible]

Staple

Fold

Fold

FIRST CLASS
PERMIT NO. 170
ENDICOTT, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
P. O. Box 6
Endicott, N. Y. 13760

Attention: Product Publications, Dept. 171

Fold

Fold

Cut Along Line

International Business Machines Corporation
Field Engineering Division
[illegible] East [illegible] Road, White [illegible]ns, N.[illegible] 1060[illegible]

Additional Comments: