IBM Field Engineering Maintenance Diagrams

Restricted Distribution

This manual is intended for internal use only and may not be used by other than IBM personnel without IBM's written permission.

2091 Processing Unit--Volume 4

I, FXP, FLP Operations

Y22-6674-1

IBM Field Engineering

Maintenance Diagrams

Restricted Distribution

This manual is intended for internal use only and may not be used by other than IBM personnel without IBM's written permission.

2091 Processing Unit--Volume 4

I, FXP, FLP Operations

Y22-6674-1

PREFACE

This is one of five volumes of the IBM 2091 Processing Unit, Field Engineering Maintenance Diagrams Manual (FEMDM). The organization of the FEMDM, the general content of each volume, and the form numbers of the five volumes are:

Title	Contents	
Volume 1 – Diagnostic Techniques, ECAD's (Form Y22-6671)	DIAGNOSTIC TECHNIQUES Diagrams 1-1 to 1-XX	
	ERROR CONDITIONS Diagrams 2-1 to 2-XX	
Volume 2 – Data Flow; I, FXP, FLP, Functional Units (Form Y22-6672)	DATA FLOW Diagram	
	3-1	System Data Flow
	3-2	I Unit Data Flow
	3-3	Fixed Point Data Flow
	3-4	Floating Point Data Flow
	3-5	MSCE Data Flow
	3-6	PSCE Data Flow
	3-7	MC Data Flow
	3-8	PSCE Storage Channel (SC) Data
	3-9	System Control Panel
	3-10	PSCE Control Panel
	FUNCTIONAL UNITS Diagrams	
	4-1 to 4-XX	I Unit
	4-100 to 4-1XX	Fixed Point Unit
	4-200 to 4-2XX	Floating Point Unit
Volume 3 – MSCE, PSCE, MC, Functional Units (Form Y22-6673)	FUNCTIONAL UNITS (Cont'd)	
	4-300 to 4-3XX	MSCE
	4-400 to 4-4XX	PSCE
	4-500 to 4-5XX	MC
Volume 4 – I, FXP, FLP Operations (Form Y22-6674)	OPERATIONS	
	5-1 to 5-XX	I Unit Operations
	5-100 to 5-1XX	Fixed Point Operations
	5-200 to 5-2XX	Floating Point Operations
Volume 5 – MSCE, PSCE, MC Operations; Power (Form Y22-6675)	OPERATIONS (Cont'd)	
	5-300 to 5-3XX	MSCE Operations
	5-400 to 5-4XX	PSCE Operations
	5-500 to 5-5XX	MC Operations
	POWER SUPPLIES 6-1 to 6-XX	

Diagrams contained in this manual are referenced from the seven 2091 Field Engineering Theory of Operation Manuals (FETOM's). References to FEMDM diagrams take the form "Diagram 5-103"; references to figures in a FETOM take the form "Figure 3-22." The seven 2091 FETOM's are:

IBM 2091 Processing Unit, FE Theory of Operation Manuals:

System Introduction, Instruction Processor, Form Y22-6622

Power Supplies and Control, Form Y22-6623

Console and Maintenance Features, Form Y22-6624

Fixed Point Execution Element, Form Y22-6625

Main Storage Control Element, Form Y22-6626

Peripheral Storage Control Element, Form Y22-6627

Floating Point Execution Element, Form Y22-6628

Other FE Manuals containing information pertinent to the 2091 are:

2091 Processing Unit, FE Maintenance Manual, Form Y22-6659

2091 Processing Unit, 2395 Processor Storage, FE Installation Manual, Form Y22-6634

Advanced Solid Logic Technology Packaging, Tools, Wiring Change and Repair Procedures, FE Theory-Maintenance Manual, Form Y22-6620

Solid Logic Technology, Packaging, Tools, Wiring Change Procedure FE Theory of Operation Manual, Form Y22-2800

Component Circuits -- SLT (Solid Logic Technology), SLD (Solid Logic Dense), ASLT (Advanced Solid Logic Technology), FE Theory of Operation Manual, Form Z22-2798 -- IBM Confidential

Power Supplies -- SLT (Solid Logic Technology), SLD (Solid Logic Dense), ASLT (Advanced Solid Logic Technology), FE Theory of Operation Manual, Form Y22-2799

Second Edition (March 1968)
This edition, Form Y22-6674-1, is a major revision of and obsoletes, Form Y22-6674-0 and FES Y22-6681. Diagrams 5-36, 5-131, and 5-132 have been added. Major changes have been made to Diagrams 5-1, 5-2, 5-6, and 5-122 through 5-126. Timing charts have been added to Diagrams 5-101, 5-102, 5-108, 5-109, and 5-112. Minor changes have been made to most of the remaining diagrams. New or changed diagrams are indicated by a page date of 3/68 and by a vertical bar to the left of the diagram number on the Contents page.

Changes are periodically made to the specifications herein; any such changes will be reported in subsequent revisions or FE Supplements.

This manual has been prepared by the IBM Systems Development Division, Product Publications, Dept. B95, PO Box 390, Poughkeepsie, N.Y. 12602. A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be sent to the above address.

CONTENTS

Diagram Title

I UNIT OPERATIONS

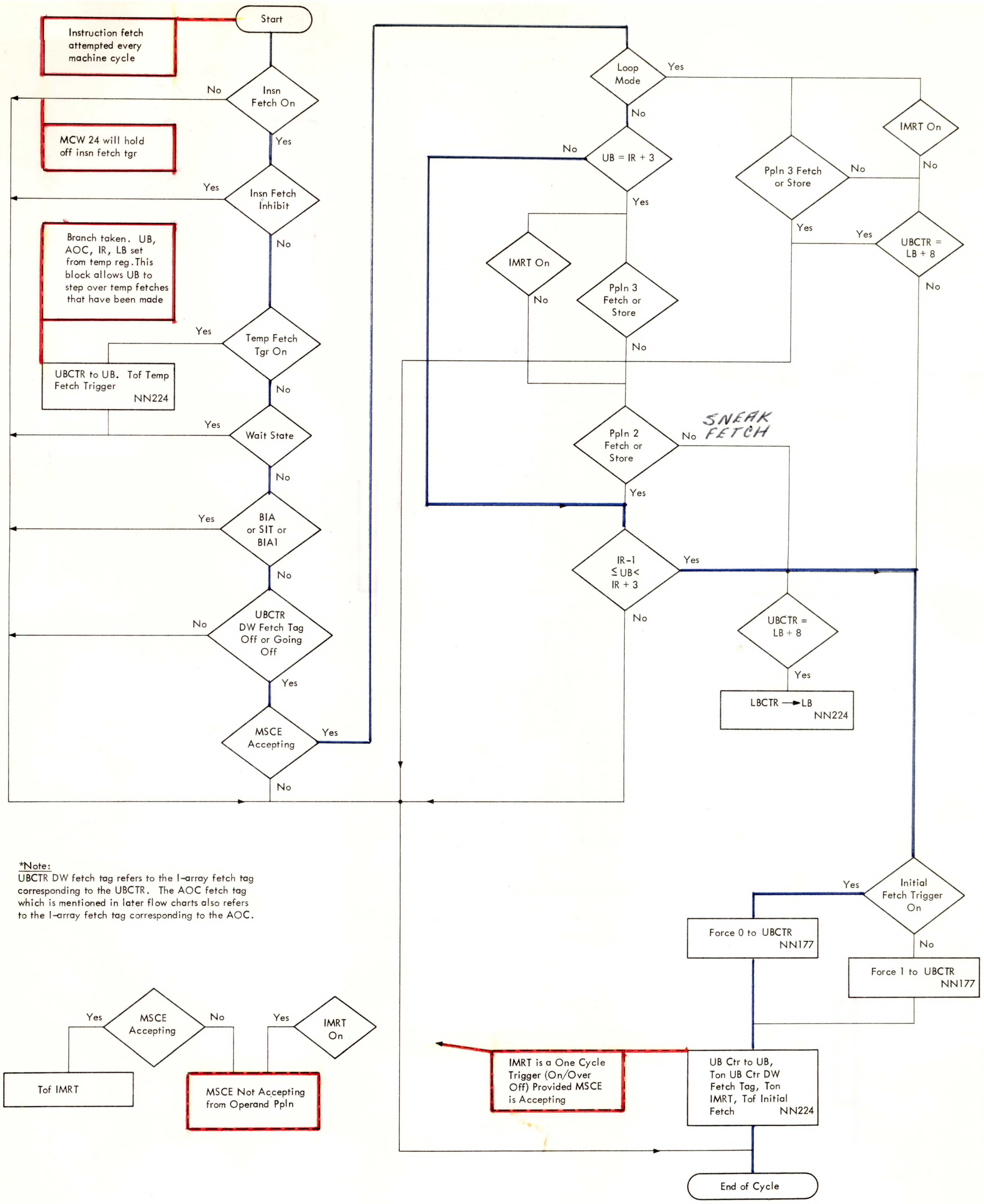
FIXED-POINT OPERATIONS

FLOATING-POINT OPERATIONS

ABBREVIATIONS

A	AND
AC	Address Check
Acc	Access; Accumulator
Acpt	Accept
Acptng	Accepting
Addr	Address
Adr	Address
Adv	Advance
AE	Address Exception
ALD	Automated Logic Diagram
Altr	Alteration
Amt	Amount
AOC	Array Out Counter
AR	Amplifier
Arg	Argument
Arg Wd	Argument Doubleword
AS	Accept Stack
ASLT	Advanced Solid Logic Technology
ATI	Auxiliary Tape Input
Avail	Available
B	Bit
BAB	Byte Address Buffer
BAC	Buffer Address Counter
BAL	(Mnemonic)Branch and Link (RX)
BALR	(Mnemonic) Branch and Link (RR)
BAR	Byte Address Register
BB	Bank Bit
BC	(Mnemonic) Branch on Condition; Bus Control
BCR	(Mnemonic) Branch on Condition (RR)
BCQT	Branch on Condition Quick Trigger
BCT	(Mnemonic) Branch on Count (RX)
BCTR	(Mnemonic) Branch on Count (RR)
BCU	Bus Control Unit
BCUNCONT	Unconditional Branch Trigger
Bd	Board
Bdy	Boundary
Bfr	Buffer
BIA	Branch In Array
BIAT	Back in Array Trigger
BOM	Basic Operating Memory
Br	Branch
BRT	Branch Trigger
BSM	Basic Storage Module
Bsy	Busy
BXH	(Mnemonic) Branch on Index High (RS)
BXLE	(Mnemonic) Branch on Index Low or Equal (RS)
BXQT	Branch on Index Quick Trigger
BZ	Busy
BZTP	Busy-to-Priority
BZTPSCE	Busy-to-PSCE
BZTR	Busy-to-Request
CAB	Channel Address Bus
CAR	Console Address Register;
CAR	Channel Address Register
CAW	Channel Address Word
C BACKL8 T	Condition Back Less than Eight Trigger
C BIA T	Conditional Back in Array Trigger
CBR	Console Buffer Register
CC	Command Counter; Condition Code
CCC	Common Channel Control
CCW	Channel Command Word
CD	Chain Data
CDB	Common Data Bus
CDBI	Console Data Bus In
CDBO	Console Data Bus Out
Ch	Channel
Chan	Channel
Ch Fr	Channel Frame
Chk	Check
Ck	Check
Chn	Chain
CIn	Carry In
CLC	(Mnemonic) Compare Logical (SS)
Clk	Clock
CM	Conditional Mode; Console Mode; Cripple Mode
Cncl	Cancel
Cndl	Conditional
Cnt	Count
CO	Conditional Op
Comp	Compare; Comparator
Cond	Condition
COut	Carry Out
CPA	Carry Propagate Adder
CPC	Cyclic Program Counter
CPE	Central Processing Element
Cpr	Computer

CPU	Central Processing Unit
C QUICK T	Conditional Quick Trigger
CR	Control Register
Crip	Cripple
CSA	Carry Save Adder
CSW	Channel Status Word
Ctl	Control
Ctr	Counter
Ctrl	Control
CV	Converter
CVB	(Mnemonic) Convert to Binary (RX)
CVD	(Mnemonic) Convert to Decimal (RF)
CXR	Console Auxiliary Register
D	Displacement
Dbl	Double
DC	Data Check; Display Check
Dcd	Decode
Dcdr	Decoder
Des	Designation
Det	Detection; Detector
DG	Display Gate
Diag	Diagnose
DIG	Data Ingate
Disp	Displacement
Dist	Distributor
Div	Divide
Dly	Delay, Delayed
Dlyd	Delayed
DM	Diagnostic Monitor
DOG	Data Outgate
DPC	Display Parity Check
Dsbl	Disable
Dt	Data
DW	Doubleword
DWC	Doubleword Counter
DWCR	Doubleword Count Register
EBA	Ending Byte Address
EBAR	Ending Byte Address Register
EBCDIC	Extended Binary Coded Decimal Interchange Code
EC	Engineering Change
ECAD	Error Check Analysis Diagram
ED	(Mnemonic) Edit (SS)
EDMK	(Mnemonic) Edit and Mark (SS)
EMS	Extended Main Storage (Same as LCS)
Eq	Equals
Err	Error
EX	(Mnemonic) Execute (RX)
Excpn	Exception
Exce	Execute
Exp	Exponent
FAU	Floating Point Add Unit
FE	Field Engineering
FEMDM	Field Engineering Maintenance Diagram Manual
FETOM	Field Engineering Theory of Operation Manual
FIFO	First-In, First-Out
Fir	First
FIWADFO	First-In-With-Available-Data, First-Out
FIWAMFO	First-In-With-Available-Memory, First-Out
FLA	Floating Point Area
FLB	Floating Point Buffer
FLBB	Floating Point Buffer Bus
Fld	Field
FLEU	Floating Point Execution Unit
FLIU	Floating Point Instruction Unit
FLOS	Floating Point Op Stack
FLP	Floating Point
FLR	Floating Point Register
FLRB	Floating Point Register Bus
FLU	Floating Point Unit
FMDU	Floating Point Multiply/Divide Unit
FP	Fetch Protect
FPA	Floating Point Area
Frm	Frame
Frac	Fraction
FS	False Start
FSB	Fixed Store Bus
Fth	Fetch
Fwd	Forward; Forwarding
FXA	Fixed Point/VFL Area
FXB	Fixed Point Buffer
FXEU	Fixed Point Execution Unit
FXIU	Fixed Point Instruction Unit
FXOS	Fixed Point Op Stack
FXP	Fixed Point



*Note:
UBCTR DW fetch tag refers to the I-array fetch tag corresponding to the UBCTR. The AOC fetch tag which is mentioned in later flow charts also refers to the I-array fetch tag corresponding to the AOC.

DIAGRAM 5-1. INSTRUCTION FETCH

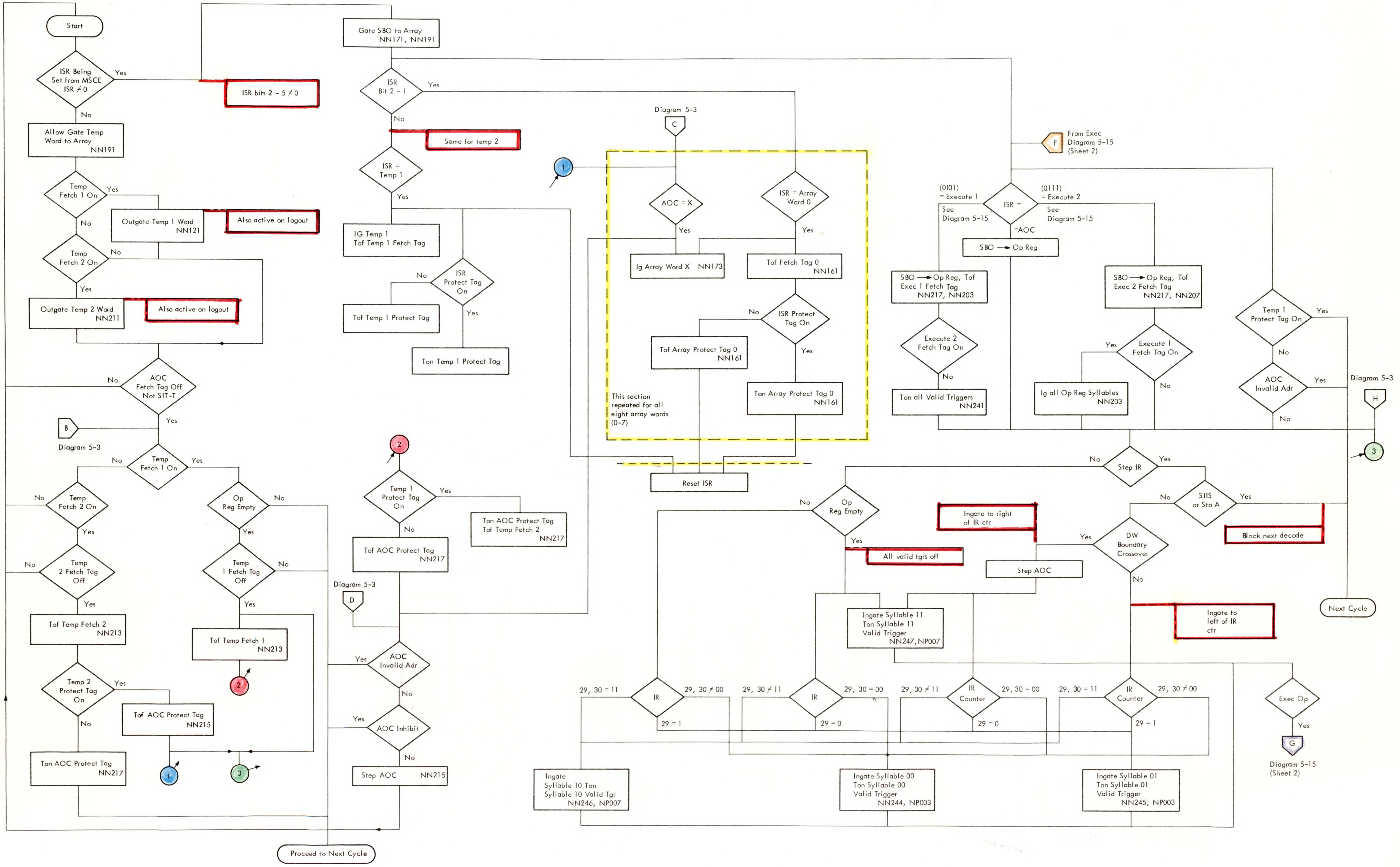


DIAGRAM 5-2. INSTRUCTION FETCH RETURN AND OP REGISTER INGATE CONTROLS

Objectives:

1. Outgate array word per AOC.
2. Generate the protect interrupt signal.

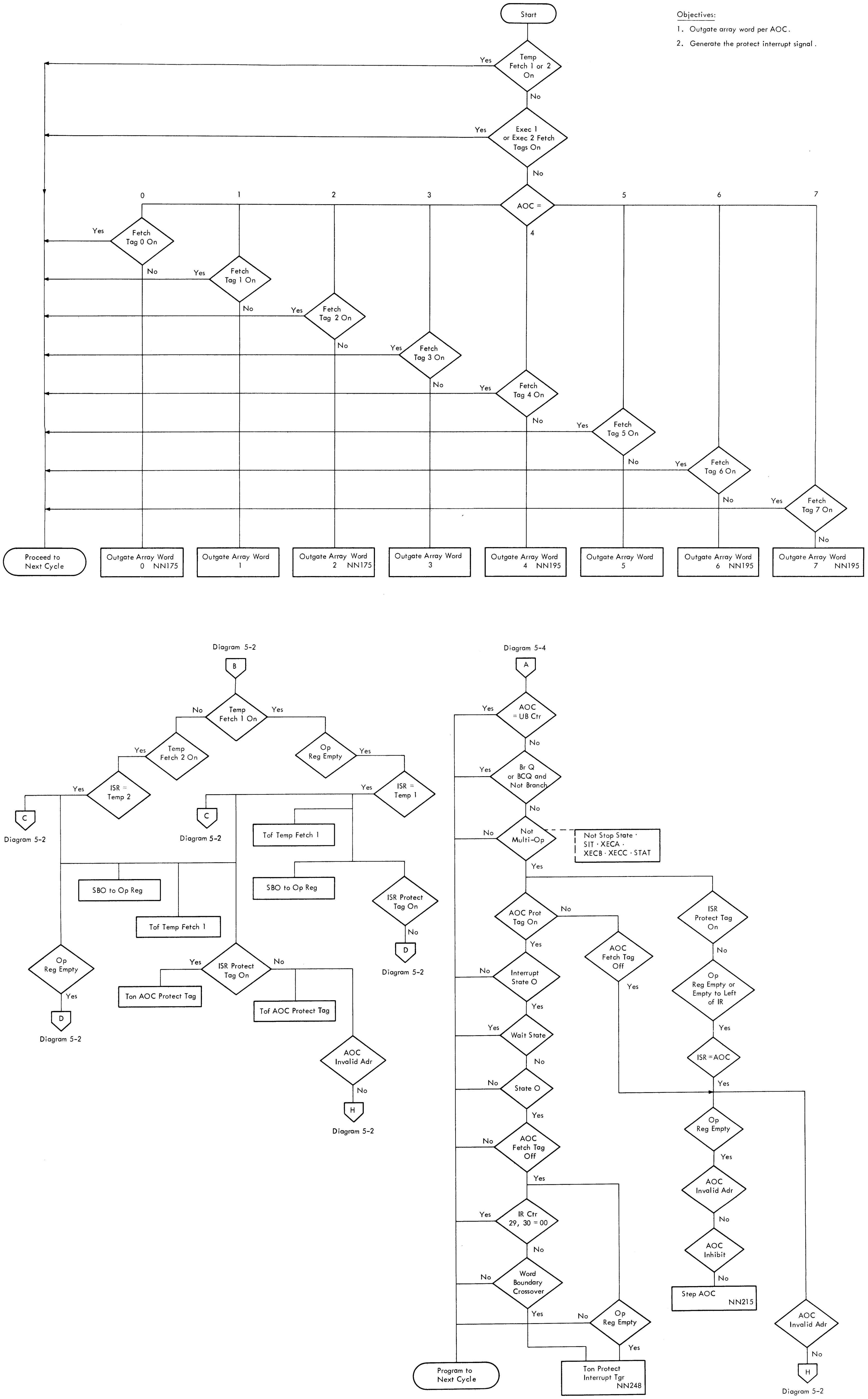
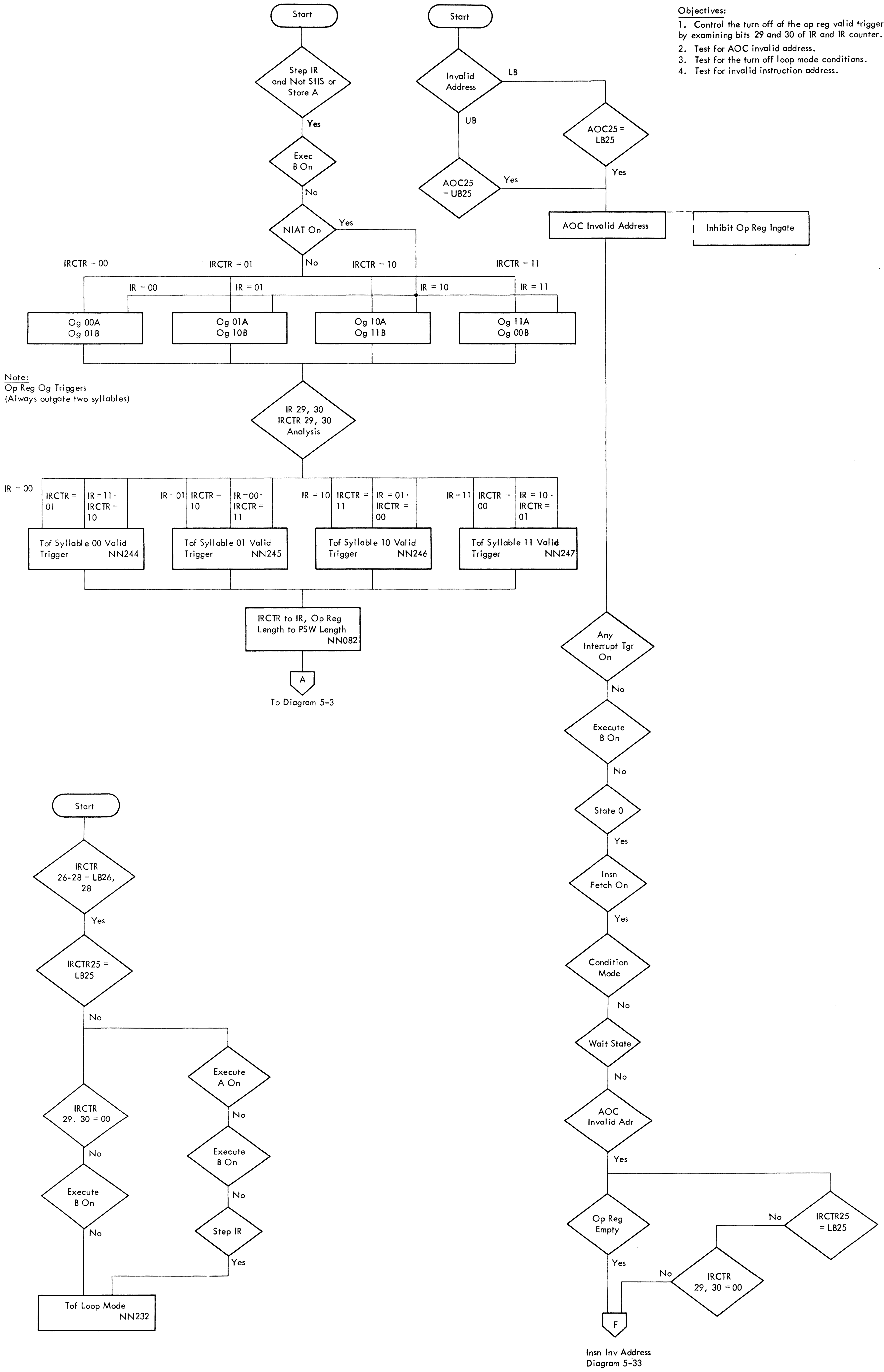


DIAGRAM 5-3. ARRAY WORD OUTGATE, FETCH PROTECT INTERRUPT AND STEP AOC, TEMP FETCH BYPASS

Objectives:

1. Control the turn off of the op reg valid triggers by examining bits 29 and 30 of IR and IR counter.
2. Test for AOC invalid address.
3. Test for the turn off loop mode conditions.
4. Test for invalid instruction address.



Note:
Op Reg Og Triggers
(Always outgate two syllables)

DIAGRAM 5-4. AOC INVALID ADDRESS, VALID TRIGGER TURN OFF AND TURN OFF LOOP MODE

Objectives:

1. Test for a store into array condition.

2. Set up to refetch the affected array word or to reinitialize the array.

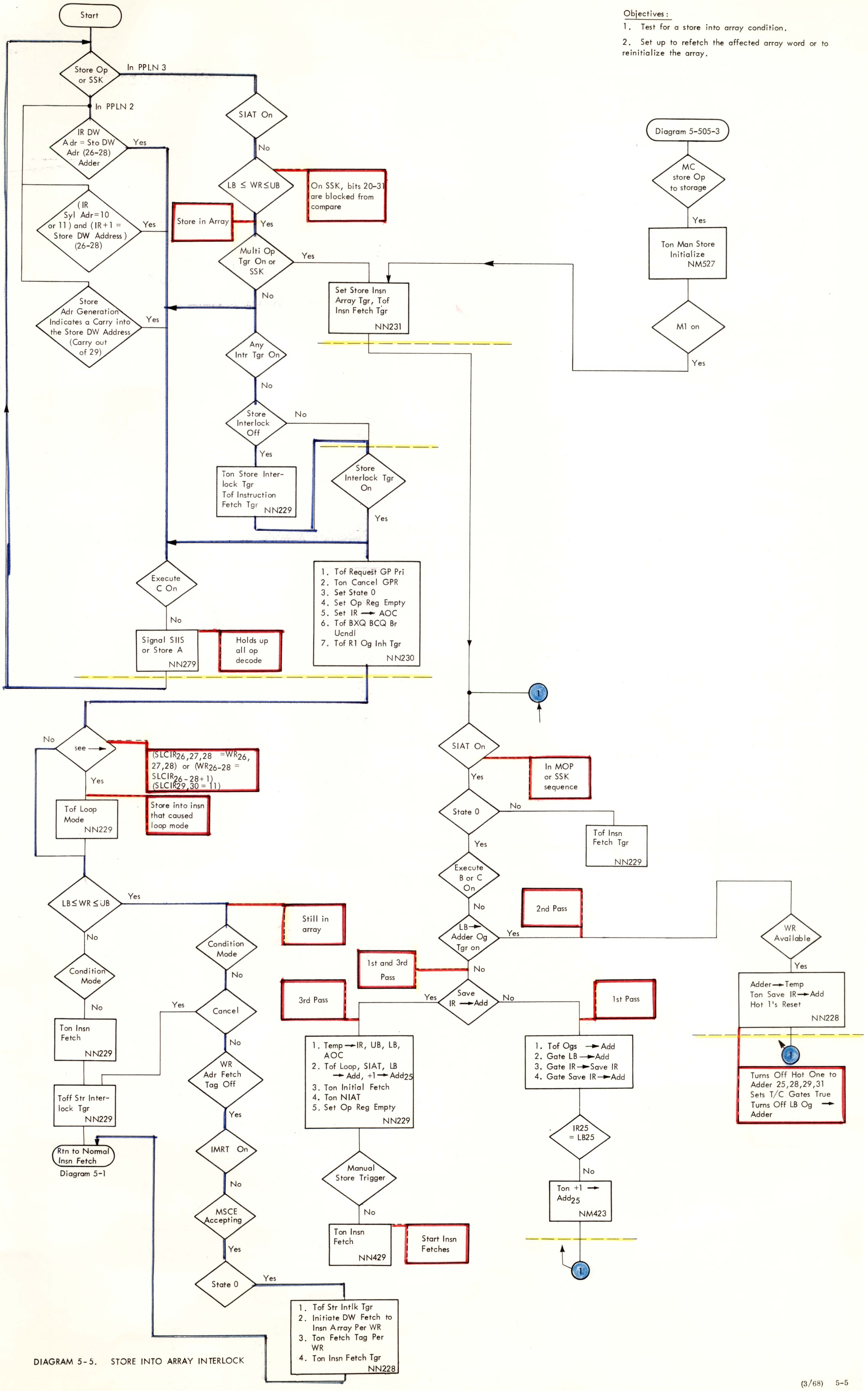


DIAGRAM 5-5. STORE INTO ARRAY INTERLOCK

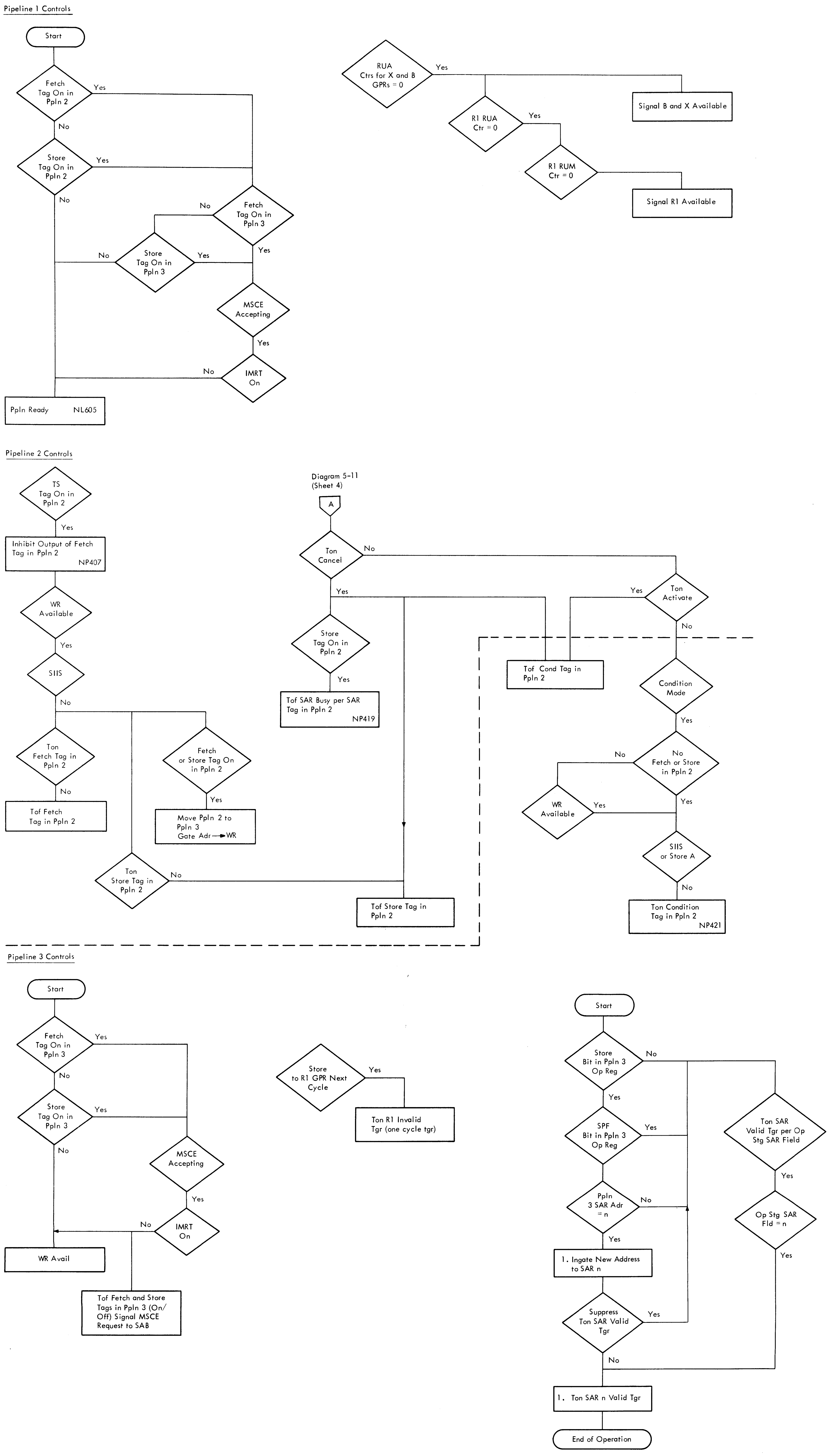


DIAGRAM 5-6. PIPELINE 2 AND 3 CONTROL

Objectives:

1. To check basic interlocks during decode cycle (state 0)

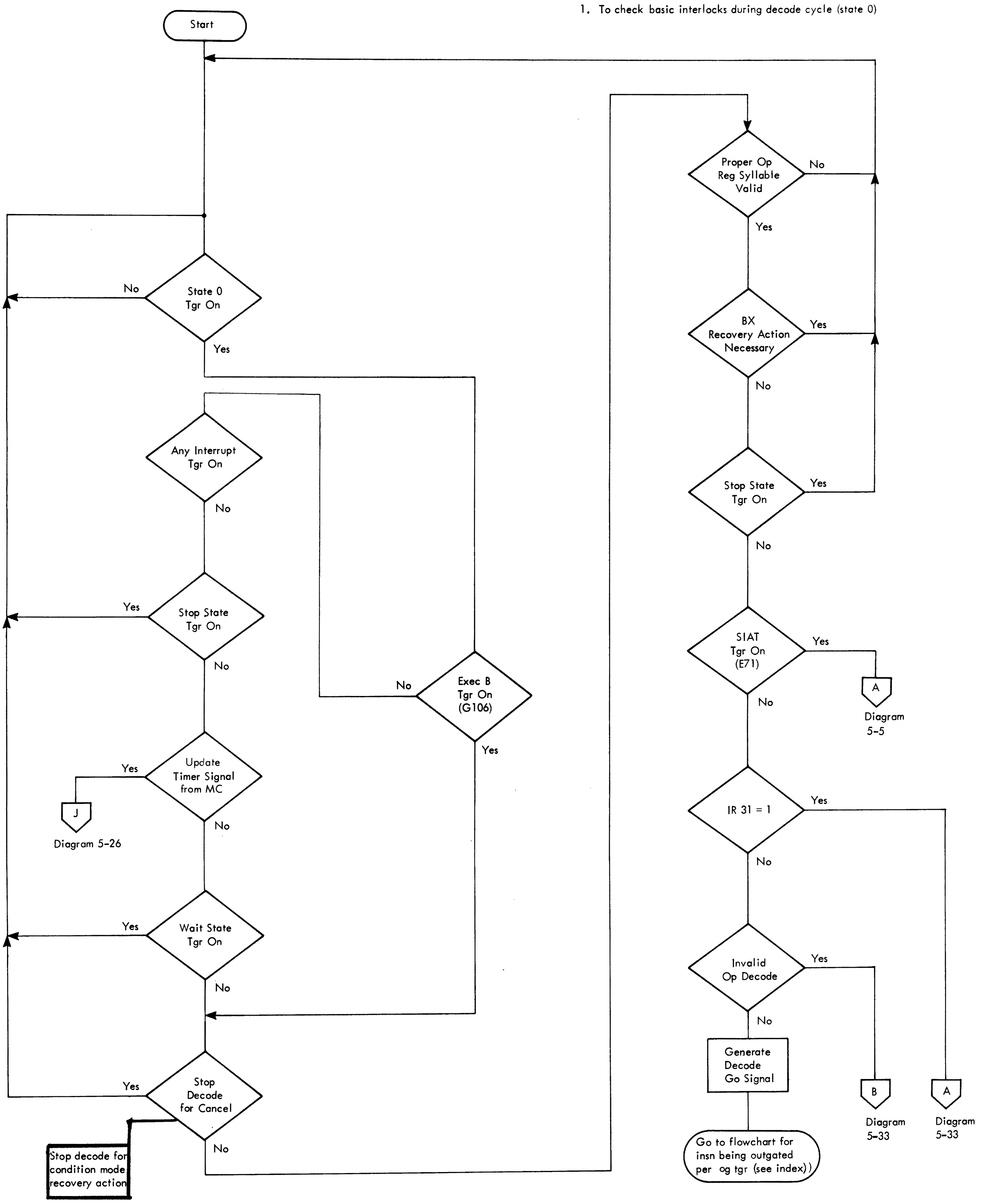


DIAGRAM 5-7. DECODE CYCLE (STATE 0) BASIC INTERLOCK CHECK

Objectives:

1. To decode and issue a floating point instruction.

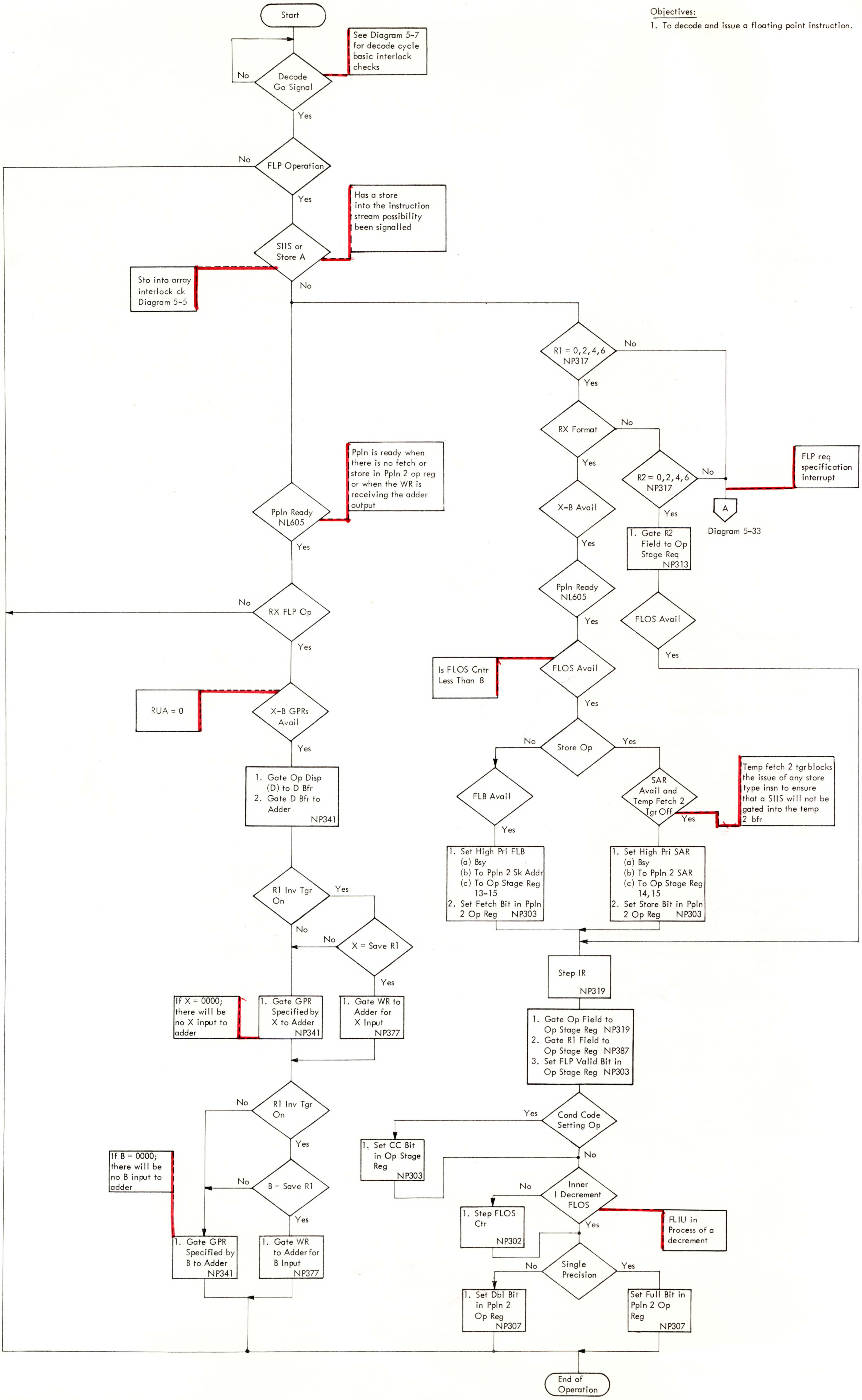

Start
See Diagram 5-7 for decode cycle basic interlock checks
No
Decode Go Signal
Yes
No
FLP Operation
Yes
Has a store into the instruction stream possibility been signalled
SIIS or Store A
Sto into array interlock ck Diagram 5-5
No
R1 = 0,2,4,6 NP317
No
Yes
RX Format
No
Yes
R2 = 0,2,4,6 NP317
No
FLP req specification interrupt
A
Diagram 5-33
Pplt is ready when there is no fetch or store in Ppln 2 op reg or when the WR is receiving the adder output
Ppln Ready NL605
Yes
X-B Avail
Yes
1. Gate R2 Field to Op Stage Req NP313
No
RX FLP Op
Yes
Ppln Ready NL605
Yes
FLOS Avail
Yes
Is FLOS Cntr Less Than 8
FLOS Avail
Yes
RUA = 0
X-B GPRs Avail
Yes
1. Gate Op Disp (D) to D Bfr
2. Gate D Bfr to Adder NP341
No
Store Op
Yes
FLB Avail
Yes
SAR Avail and Temp Fetch 2 Tgr Off
Yes
Temp fetch 2 tgr blocks the issue of any store type insn to ensure that a SIIS will not be gated into the temp 2 bfr
1. Set High Pri FLB (a) Bsy (b) To Ppln 2 Sk Addr (c) To Op Stage Reg 13-15
2. Set Fetch Bit in Ppln 2 Op Reg NP303
1. Set High Pri SAR (a) Bsy (b) To Ppln 2 SAR (c) To Op Stage Reg 14,15
2. Set Store Bit in Ppln 2 Op Reg NP303
R1 Inv Tgr On
Yes
No
No
X = Save R1
Yes
Step IR NP319
If X = 0000; there will be no X input to adder
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
1. Gate Op Field to Op Stage Reg NP319
2. Gate R1 Field to Op Stage Reg NP387
3. Set FLP Valid Bit in Op Stage Reg NP303
No
R1 Inv Tgr On
Yes
Yes
Cond Code Setting Op
No
1. Set CC Bit in Op Stage Reg NP303
If B = 0000; there will be no B input to adder
No
B = Save R1
Yes
No
Inner I Decrement FLOS
Yes
FLIU in Process of a decrement
1. Step FLOS Ctr NP302
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B Input NP377
No
Single Precision
Yes
1. Set Dbl Bit in Ppln 2 Op Reg NP307
Set Full Bit in Ppln 2 Op Reg NP307
End of Operation

DIAGRAM 5-8. FLOATING POINT ISSUE SEQUENCE

Objectives:

1. To decode and issue a FXP full word operation.

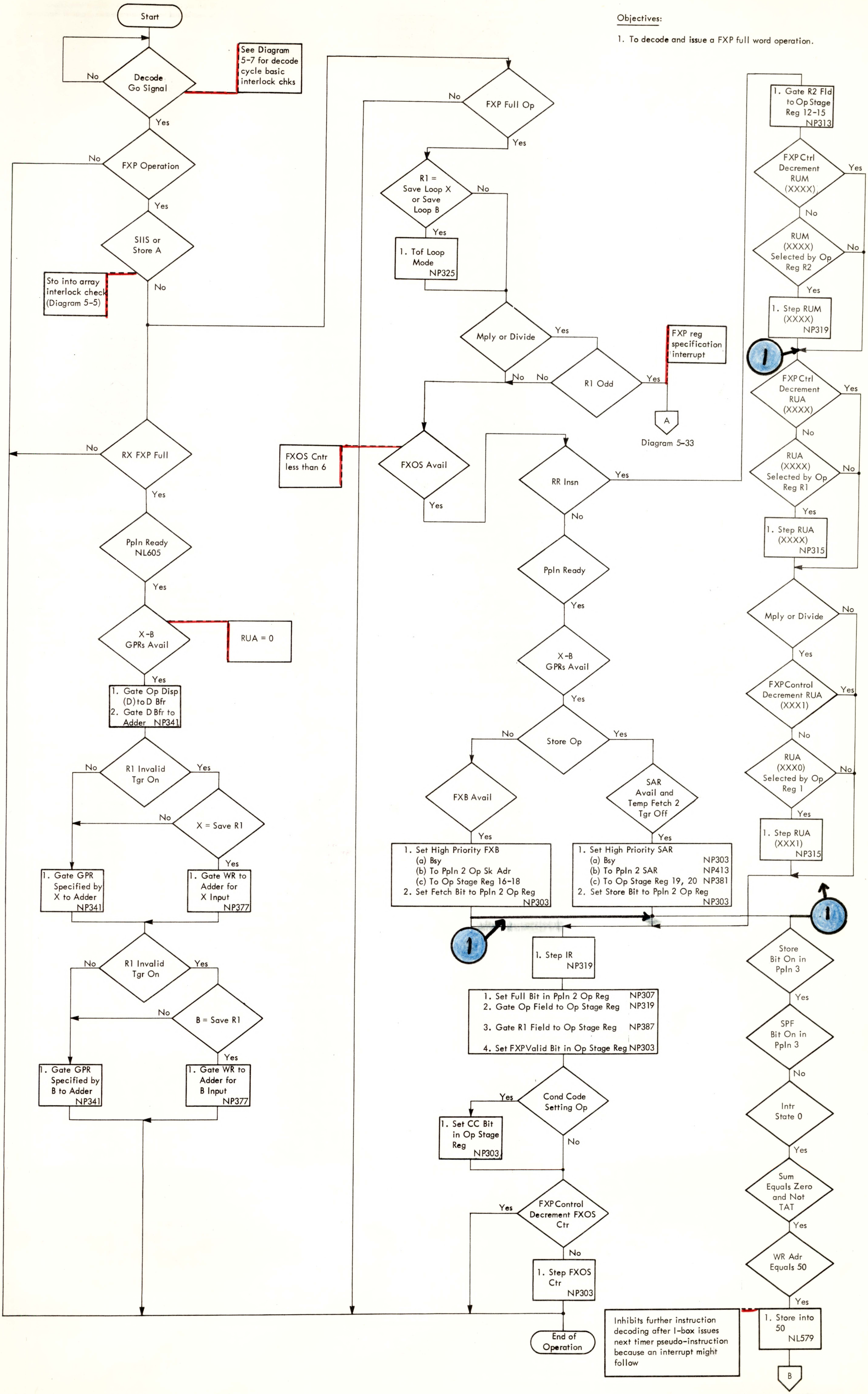
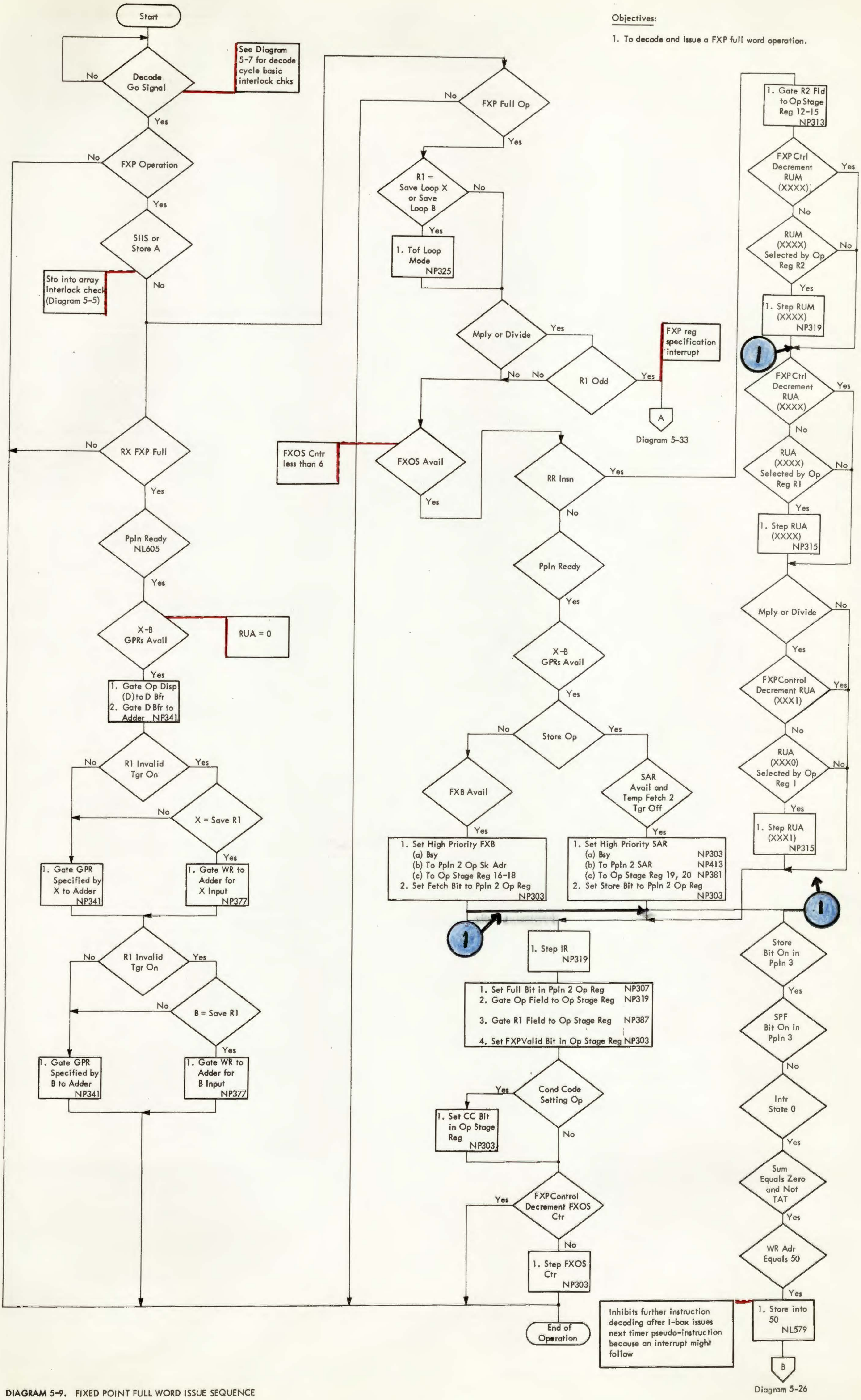

Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock chks
FXP Operation
No
Yes
SIIS or Store A
Sto into array interlock check (Diagram 5-5)
No
RX FXP Full
No
Yes
Ppln Ready NL605
Yes
X-B GPRs Avail
RUA = 0
Yes
1. Gate Op Disp (D) to D Bfr
2. Gate D Bfr to Adder NP341
R1 Invalid Tgr On
No
Yes
X = Save R1
No
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
R1 Invalid Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B Input NP377
FXP Full Op
No
Yes
R1 = Save Loop X or Save Loop B
No
Yes
1. Tof Loop Mode NP325
Mply or Divide
Yes
No
No
R1 Odd
Yes
FXP reg specification interrupt
A
Diagram 5-33
FXOS Cntr less than 6
FXOS Avail
Yes
RR Insn
Yes
No
Ppln Ready
Yes
X-B GPRs Avail
Yes
Store Op
No
Yes
FXB Avail
Yes
1. Set High Priority FXB
(a) Bsy
(b) To Ppln 2 Op Sk Adr
(c) To Op Stage Reg 16-18
2. Set Fetch Bit to Ppln 2 Op Reg NP303
SAR Avail and Temp Fetch 2 Tgr Off
Yes
1. Set High Priority SAR
(a) Bsy NP303
(b) To Ppln 2 SAR NP413
(c) To Op Stage Reg 19, 20 NP381
2. Set Store Bit to Ppln 2 Op Reg NP303
1
1
1
1. Step IR NP319
1. Set Full Bit in Ppln 2 Op Reg NP307
2. Gate Op Field to Op Stage Reg NP319
3. Gate R1 Field to Op Stage Reg NP387
4. Set FXPValid Bit in Op Stage Reg NP303
Cond Code Setting Op
Yes
No
1. Set CC Bit in Op Stage Reg NP303
FXP Control Decrement FXOS Ctr
Yes
No
1. Step FXOS Ctr NP303
End of Operation
1. Gate R2 Fld to Op Stage Reg 12-15 NP313
FXP Ctrl Decrement RUM (XXXX)
Yes
No
RUM (XXXX) Selected by Op Reg R2
No
Yes
1. Step RUM (XXXX) NP319
FXP Ctrl Decrement RUA (XXXX)
Yes
No
RUA (XXXX) Selected by Op Reg R1
No
Yes
1. Step RUA (XXXX) NP315
Mply or Divide
No
Yes
FXP Control Decrement RUA (XXX1)
Yes
No
RUA (XXX0) Selected by Op Reg 1
No
Yes
1. Step RUA (XXX1) NP315
Store Bit On in Ppln 3
Yes
SPF Bit On in Ppln 3
No
Intr State 0
Yes
Sum Equals Zero and Not TAT
Yes
WR Adr Equals 50
Yes
Inhibits further instruction decoding after I-box issues next timer pseudo-instruction because an interrupt might follow
1. Store into 50 NL579
B
Diagram 5-26

DIAGRAM 5-9. FIXED POINT FULL WORD ISSUE SEQUENCE

Objectives:

1. To decode and issue a FXP full word operation.



Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock chks
FXP Operation
No
Yes
SIIS or Store A
Sto into array interlock check (Diagram 5-5)
No
RX FXP Full
No
Yes
Ppln Ready NL605
Yes
X-B GPRs Avail
RUA = 0
Yes
1. Gate Op Disp (D) to D Bfr
2. Gate D Bfr to Adder NP341
R1 Invalid Tgr On
No
Yes
X = Save R1
No
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
R1 Invalid Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B Input NP377
FXP Full Op
No
Yes
R1 = Save Loop X or Save Loop B
No
Yes
1. Tof Loop Mode NP325
Mply or Divide
Yes
No
No
R1 Odd
Yes
FXP reg specification interrupt
A
Diagram 5-33
FXOS Cntr less than 6
FXOS Avail
Yes
RR Insn
Yes
No
Ppln Ready
Yes
X-B GPRs Avail
Yes
Store Op
No
Yes
FXB Avail
Yes
1. Set High Priority FXB
(a) Bsy
(b) To Ppln 2 Op Sk Adr
(c) To Op Stage Reg 16-18
2. Set Fetch Bit to Ppln 2 Op Reg NP303
SAR Avail and Temp Fetch 2 Tgr Off
Yes
1. Set High Priority SAR
(a) Bsy NP303
(b) To Ppln 2 SAR NP413
(c) To Op Stage Reg 19, 20 NP381
2. Set Store Bit to Ppln 2 Op Reg NP303
1
1. Step IR NP319
1. Set Full Bit in Ppln 2 Op Reg NP307
2. Gate Op Field to Op Stage Reg NP319
3. Gate R1 Field to Op Stage Reg NP387
4. Set FXPValid Bit in Op Stage Reg NP303
Cond Code Setting Op
Yes
No
1. Set CC Bit in Op Stage Reg NP303
FXPControl Decrement FXOS Ctr
Yes
No
1. Step FXOS Ctr NP303
End of Operation
1. Gate R2 Fld to Op Stage Reg 12-15 NP313
FXP Ctrl Decrement RUM (XXXX)
Yes
No
RUM (XXXX) Selected by Op Reg R2
No
Yes
1. Step RUM (XXXX) NP319
1
FXP Ctrl Decrement RUA (XXXX)
Yes
No
RUA (XXXX) Selected by Op Reg R1
No
Yes
1. Step RUA (XXXX) NP315
Mply or Divide
No
Yes
FXPControl Decrement RUA (XXX1)
Yes
No
RUA (XXX0) Selected by Op Reg 1
No
Yes
1. Step RUA (XXX1) NP315
1
Store Bit On in Ppln 3
Yes
SPF Bit On in Ppln 3
No
Intr State 0
Yes
Sum Equals Zero and Not TAT
Yes
WR Adr Equals 50
Yes
1. Store into 50 NL579
Inhibits further instruction decoding after I-box issues next timer pseudo-instruction because an interrupt might follow
B
Diagram 5-26

DIAGRAM 5-9. FIXED POINT FULL WORD ISSUE SEQUENCE

Objectives:

1. To decode and issue a fix point halfword instruction.
2. To decode and issue a insert and store character instruction.

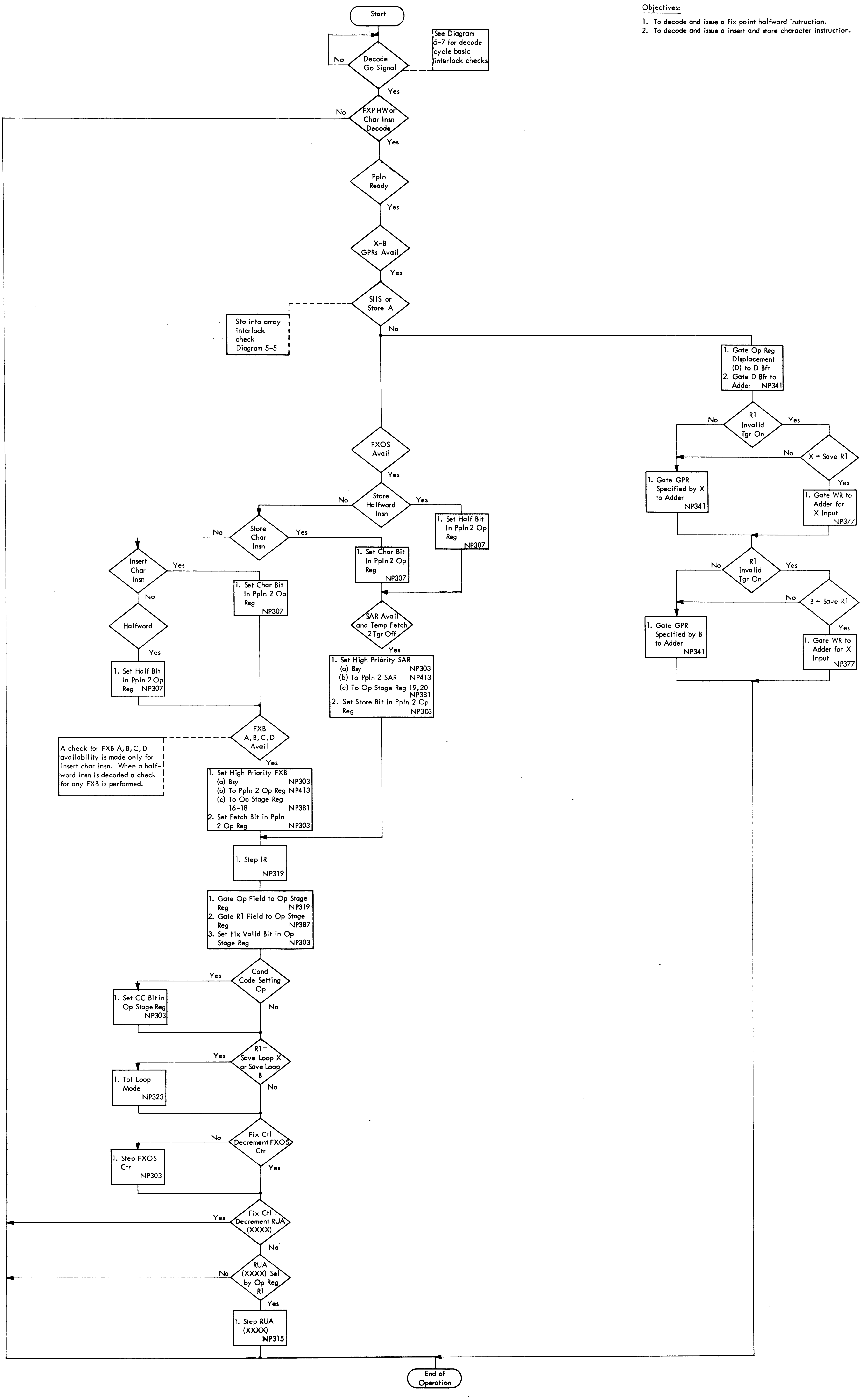
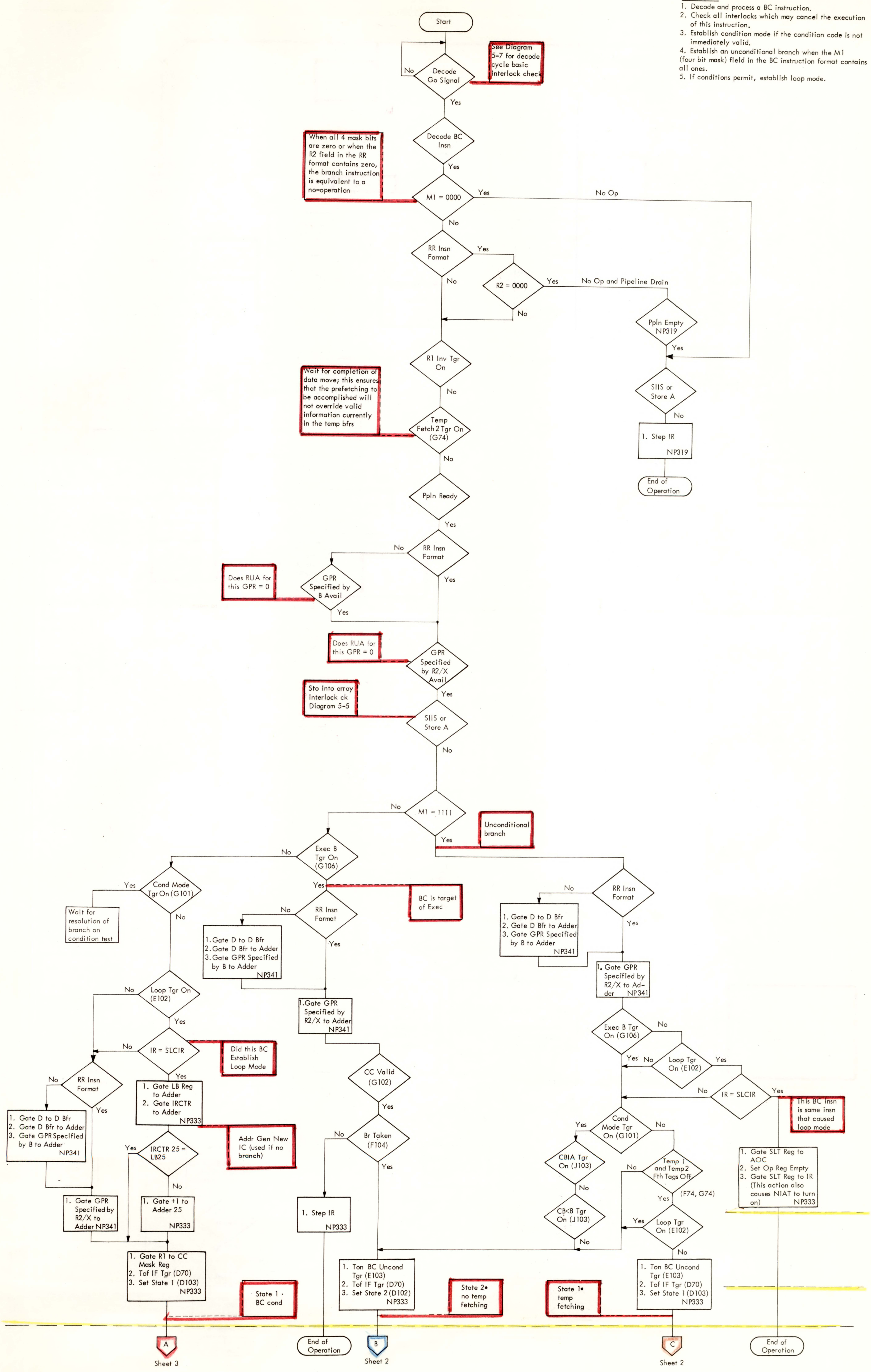

Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock checks
FXP HW or Char Insn Decode
No
Yes
Ppln Ready
Yes
X-B GPRs Avail
Yes
SIIS or Store A
No
Sto into array interlock check Diagram 5-5
1. Gate Op Reg Displacement (D) to D Bfr
2. Gate D Bfr to Adder NP341
R1 Invalid Tgr On
No
Yes
X = Save R1
No
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
R1 Invalid Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for X Input NP377
FXOS Avail
Yes
Store Halfword Insn
No
Yes
1. Set Half Bit In Ppln 2 Op Reg NP307
Store Char Insn
No
Yes
1. Set Char Bit In Ppln 2 Op Reg NP307
Insert Char Insn
Yes
No
1. Set Char Bit In Ppln 2 Op Reg NP307
Halfword
Yes
1. Set Half Bit in Ppln 2 Op Reg NP307
SAR Avail and Temp Fetch 2 Tgr Off
Yes
1. Set High Priority SAR
(a) Bsy NP303
(b) To Ppln 2 SAR NP413
(c) To Op Stage Reg 19, 20 NP381
2. Set Store Bit in Ppln 2 Op Reg NP303
FXB A, B, C, D Avail
Yes
A check for FXB A, B, C, D availability is made only for insert char insn. When a halfword insn is decoded a check for any FXB is performed.
1. Set High Priority FXB
(a) Bsy NP303
(b) To Ppln 2 Op Reg NP413
(c) To Op Stage Reg 16-18 NP381
2. Set Fetch Bit in Ppln 2 Op Reg NP303
1. Step IR NP319
1. Gate Op Field to Op Stage Reg NP319
2. Gate R1 Field to Op Stage Reg NP387
3. Set Fix Valid Bit in Op Stage Reg NP303
Cond Code Setting Op
Yes
No
1. Set CC Bit in Op Stage Reg NP303
R1 = Save Loop X or Save Loop B
Yes
No
1. Tof Loop Mode NP323
Fix Ctl Decrement FXOS Ctr
No
Yes
1. Step FXOS Ctr NP303
Fix Ctl Decrement RUA (XXXX)
Yes
No
RUA (XXXX) Sel by Op Reg R1
No
Yes
1. Step RUA (XXXX) NP315
End of Operation

DIAGRAM 5-10. FIXED POINT HALFWORD AND INSERT AND STORE CHARACTER ISSUE SEQUENCE

Objectives:

1. Decode and process a BC instruction.
2. Check all interlocks which may cancel the execution of this instruction.
3. Establish condition mode if the condition code is not immediately valid.
4. Establish an unconditional branch when the M1 (four bit mask) field in the BC instruction format contains all ones.
5. If conditions permit, establish loop mode.



Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock check
Decode BC Insn
When all 4 mask bits are zero or when the R2 field in the RR format contains zero, the branch instruction is equivalent to a no-operation
M1 = 0000
No Op
RR Insn Format
R2 = 0000
No Op and Pipeline Drain
Ppln Empty NP319
R1 Inv Tgr On
SIIS or Store A
1. Step IR NP319
End of Operation
Wait for completion of data move; this ensures that the prefetching to be accomplished will not override valid information currently in the temp bfrs
Temp Fetch 2 Tgr On (G74)
Ppln Ready
RR Insn Format
Does RUA for this GPR = 0
GPR Specified by B Avail
Does RUA for this GPR = 0
GPR Specified by R2/X Avail
Sto into array interlock ck Diagram 5-5
SIIS or Store A
M1 = 1111
Unconditional branch
Exec B Tgr On (G106)
BC is target of Exec
Cond Mode Tgr On (G101)
Wait for resolution of branch on condition test
RR Insn Format
1. Gate D to D Bfr 2. Gate D Bfr to Adder 3. Gate GPR Specified by B to Adder NP341
1. Gate GPR Specified by R2/X to Adder NP341
RR Insn Format
1. Gate D to D Bfr 2. Gate D Bfr to Adder 3. Gate GPR Specified by B to Adder NP341
1. Gate GPR Specified by R2/X to Ad-der NP341
Loop Tgr On (E102)
IR = SLCIR
Did this BC Establish Loop Mode
RR Insn Format
1. Gate D to D Bfr 2. Gate D Bfr to Adder 3. Gate GPR Specified by B to Adder NP341
1. Gate LB Reg to Adder 2. Gate IRCTR to Adder NP333
Addr Gen New IC (used if no branch)
IRCTR 25 = LB25
1. Gate GPR Specified by R2/X to Adder NP341
1. Gate +1 to Adder 25 NP333
1. Gate R1 to CC Mask Reg 2. Tof IF Tgr (D70) 3. Set State 1 (D103) NP333
State 1 · BC cond
CC Valid (G102)
Br Taken (F104)
1. Step IR NP333
1. Ton BC Uncond Tgr (E103) 2. Tof IF Tgr (D70) 3. Set State 2 (D102) NP333
State 2• no temp fetching
Exec B Tgr On (G106)
Loop Tgr On (E102)
IR = SLCIR
This BC insn is same insn that caused loop mode
Cond Mode Tgr On (G101)
CBIA Tgr On (J103)
CB<8 Tgr On (J103)
Temp 1 and Temp 2 Fth Tags Off
(F74, G74)
Loop Tgr On (E102)
1. Gate SLT Reg to AOC 2. Set Op Reg Empty 3. Gate SLT Reg to IR (This action also causes NIAT to turn on) NP333
State 1• temp fetching
1. Ton BC Uncond Tgr (E103) 2. Tof IF Tgr (D70) 3. Set State 1 (D103) NP333
A Sheet 3
End of Operation
B Sheet 2
C Sheet 2
End of Operation

DIAGRAM 5-11. BRANCH ON CONDITION SEQUENCE (SHEET 1 OF 5)

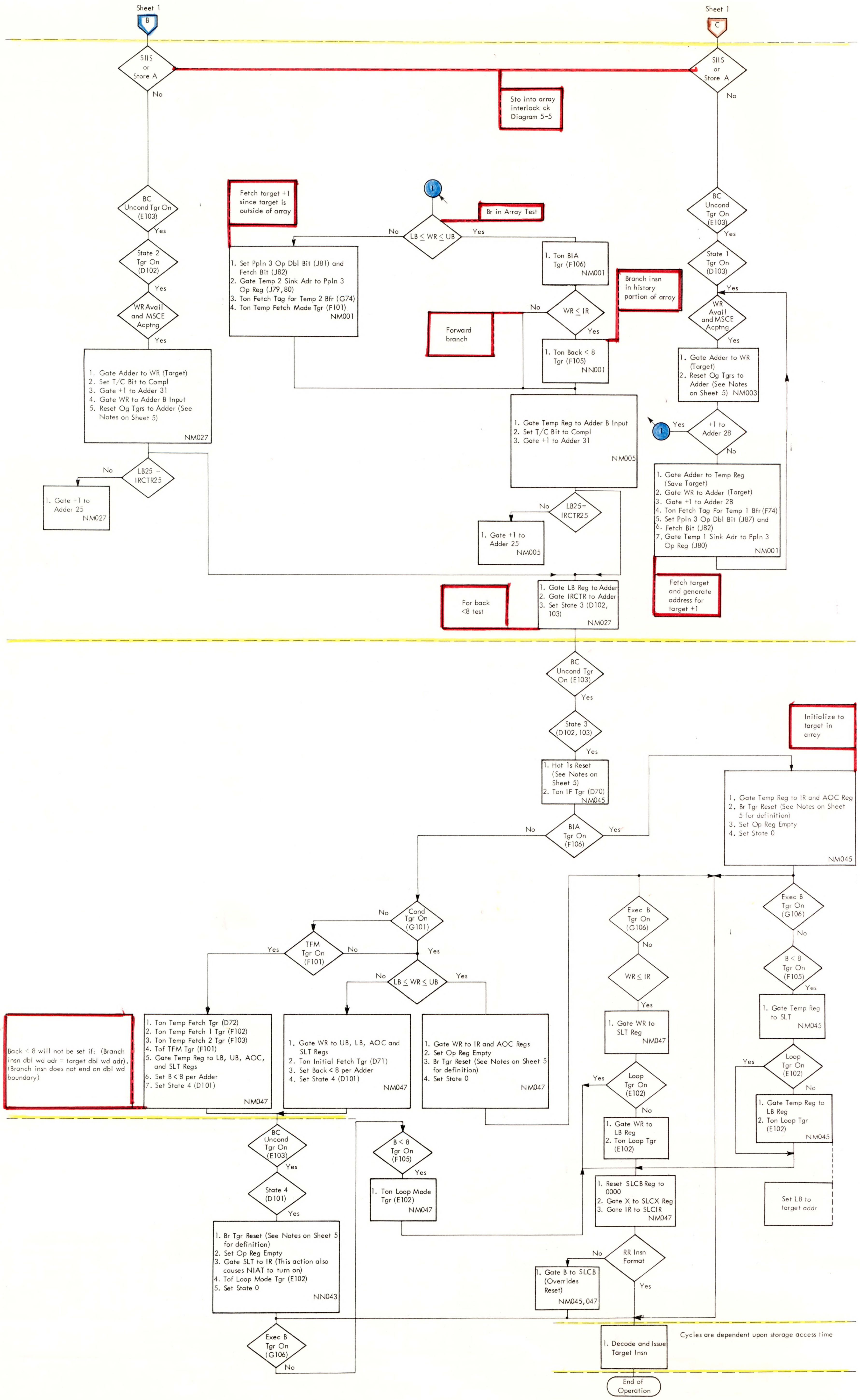

Sheet 1
B
Sheet 1
C
SIIS or Store A
SIIS or Store A
No
No
Sto into array interlock ck Diagram 5-5
BC Uncond Tgr On (E103)
Yes
State 2 Tgr On (D102)
Yes
WR Avail and MSCE Acptng
Yes
1. Gate Adder to WR (Target)
2. Set T/C Bit to Compl
3. Gate +1 to Adder 31
4. Gate WR to Adder B Input
5. Reset Og Tgrs to Adder (See Notes on Sheet 5)
NM027
No
LB25 = IRCTR25
1. Gate +1 to Adder 25
NM027
Fetch target +1 since target is outside of array
1
Br in Array Test
No
LB ≤ WR ≤ UB
Yes
1. Set Ppln 3 Op Dbl Bit (J81) and Fetch Bit (J82)
2. Gate Temp 2 Sink Adr to Ppln 3 Op Reg (J79,80)
3. Ton Fetch Tag for Temp 2 Bfr (G74)
4. Ton Temp Fetch Made Tgr (F101)
NM001
1. Ton BIA Tgr (F106)
NM001
Branch insn in history portion of array
No
WR ≤ IR
Yes
Forward branch
1. Ton Back < 8 Tgr (F105)
NN001
1. Gate Temp Reg to Adder B Input
2. Set T/C Bit to Compl
3. Gate +1 to Adder 31
NM005
No
LB25= IRCTR25
1. Gate +1 to Adder 25
NM005
For back <8 test
1. Gate LB Reg to Adder
2. Gate IRCTR to Adder
3. Set State 3 (D102, 103)
NM027
BC Uncond Tgr On (E103)
Yes
State 1 Tgr On (D103)
Yes
WR Avail and MSCE Acptng
Yes
1. Gate Adder to WR (Target)
2. Reset Og Tgrs to Adder (See Notes on Sheet 5) NM003
1
Yes
+1 to Adder 28
No
1. Gate Adder to Temp Reg (Save Target)
2. Gate WR to Adder (Target)
3. Gate +1 to Adder 28
4. Ton Fetch Tag For Temp 1 Bfr (F74)
5. Set Ppln 3 Op Dbl Bit (J87) and
6. Fetch Bit (J82)
7. Gate Temp 1 Sink Adr to Ppln 3 Op Reg (J80)
NM001
Fetch target and generate address for target +1
BC Uncond Tgr On (E103)
Yes
State 3 (D102, 103)
Yes
1. Hot 1s Reset (See Notes on Sheet 5)
2. Ton IF Tgr (D70)
NM045
No
BIA Tgr On (F106)
Yes
Initialize to target in array
1. Gate Temp Reg to IR and AOC Reg
2. Br Tgr Reset (See Notes on Sheet 5 for definition)
3. Set Op Reg Empty
4. Set State 0
NM045
No
Cond Tgr On (G101)
Yes
Yes
TFM Tgr On (F101)
No
No
LB ≤ WR ≤ UB
Yes
Back < 8 will not be set if: (Branch insn dbl wd adr = target dbl wd adr). (Branch insn does not end on dbl wd boundary)
1. Ton Temp Fetch Tgr (D72)
2. Ton Temp Fetch 1 Tgr (F102)
3. Ton Temp Fetch 2 Tgr (F103)
4. Tof TFM Tgr (F101)
5. Gate Temp Reg to LB, UB, AOC, and SLT Regs
6. Set B < 8 per Adder
7. Set State 4 (D101)
NM047
1. Gate WR to UB, LB, AOC and SLT Regs
2. Ton Initial Fetch Tgr (D71)
3. Set Back < 8 per Adder
4. Set State 4 (D101)
NM047
1. Gate WR to IR and AOC Regs
2. Set Op Reg Empty
3. Br Tgr Reset (See Notes on Sheet 5 for definition)
4. Set State 0
NM047
BC Uncond Tgr On (E103)
Yes
State 4 (D101)
Yes
1. Br Tgr Reset (See Notes on Sheet 5 for definition)
2. Set Op Reg Empty
3. Gate SLT to IR (This action also causes NIAT to turn on)
4. Tof Loop Mode Tgr (E102)
5. Set State 0
NN043
Exec B Tgr On (G106)
No
B < 8 Tgr On (F105)
Yes
1. Ton Loop Mode Tgr (E102)
NM047
Exec B Tgr On (G106)
No
WR ≤ IR
Yes
1. Gate WR to SLT Reg
NM047
Yes
Loop Tgr On (E102)
No
1. Gate WR to LB Reg
2. Ton Loop Tgr (E102)
1. Reset SLCB Reg to 0000
2. Gate X to SLCX Reg
3. Gate IR to SLCIR
NM047
No
RR Insn Format
Yes
1. Gate B to SLCB (Overrides Reset)
NM045,047
1. Decode and Issue Target Insn
Cycles are dependent upon storage access time
End of Operation
Exec B Tgr On (G106)
No
B < 8 Tgr On (F105)
Yes
1. Gate Temp Reg to SLT
NM045
Yes
Loop Tgr On (E102)
No
1. Gate Temp Reg to LB Reg
2. Ton Loop Tgr (E102)
NM045
Set LB to target addr

DIAGRAM 5-11. BRANCH ON CONDITION SEQUENCE (SHEET 2 OF 5)

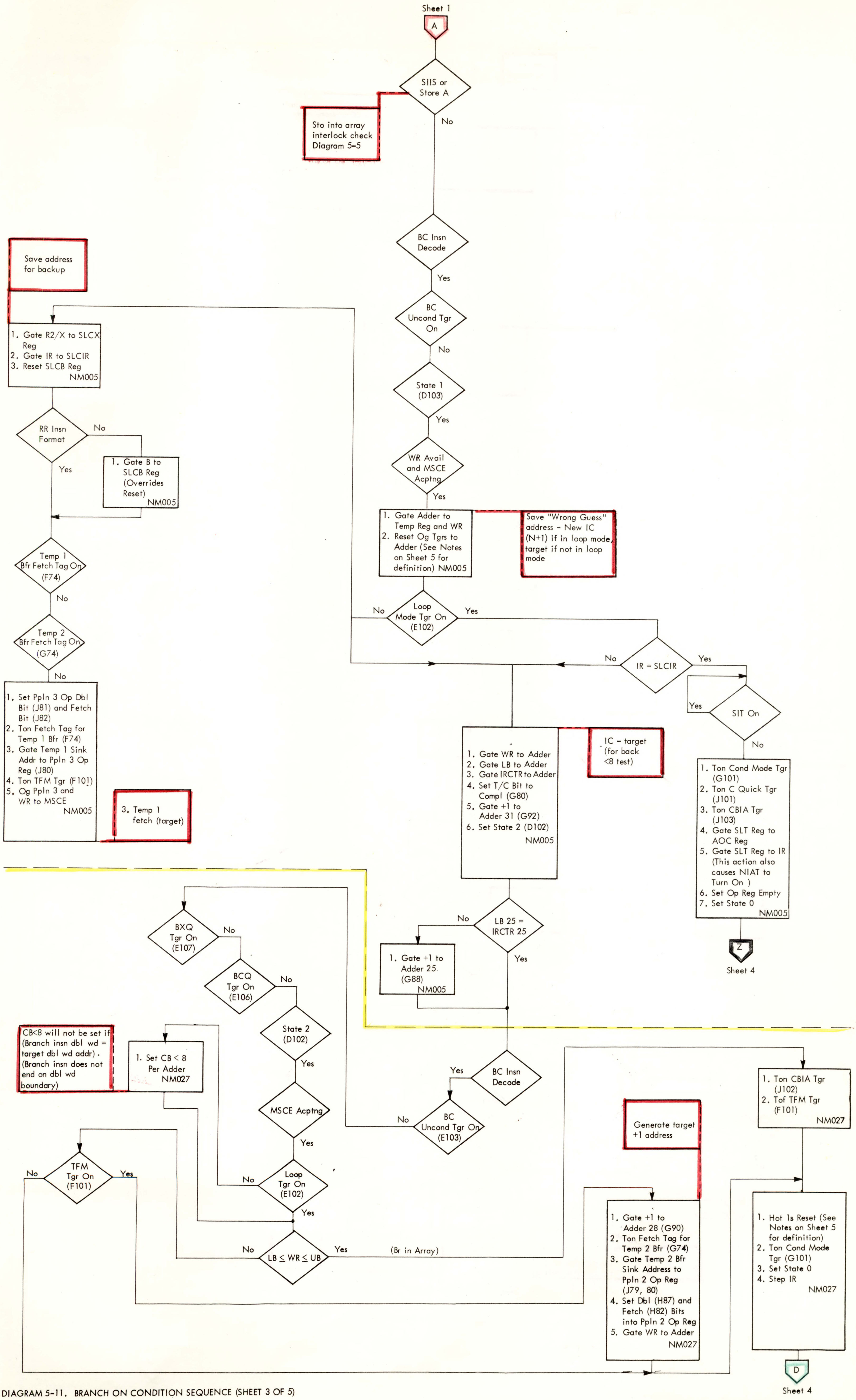
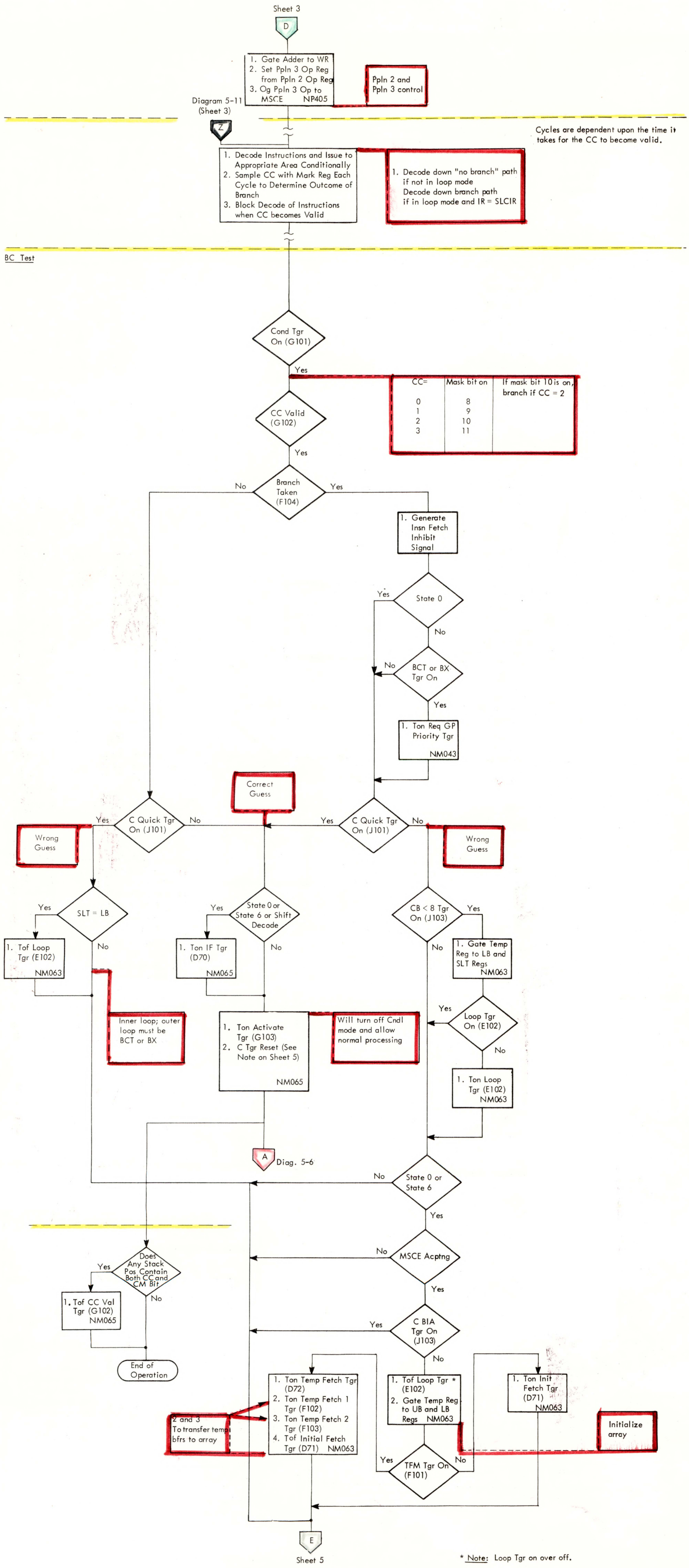

Sheet 1
A
SIIS or Store A
No
Sto into array interlock check Diagram 5-5
BC Insn Decode
Yes
BC Uncond Tgr On
No
State 1 (D103)
Yes
WR Avail and MSCE Acptng
Yes
1. Gate Adder to Temp Reg and WR
2. Reset Og Tgrs to Adder (See Notes on Sheet 5 for definition) NM005
Save "Wrong Guess" address - New IC (N+1) if in loop mode, target if not in loop mode
Loop Mode Tgr On (E102)
No
Yes
IR = SLCIR
No
Yes
SIT On
Yes
No
1. Ton Cond Mode Tgr (G101)
2. Ton C Quick Tgr (J101)
3. Ton CBIA Tgr (J103)
4. Gate SLT Reg to AOC Reg
5. Gate SLT Reg to IR (This action also causes NIAT to Turn On)
6. Set Op Reg Empty
7. Set State 0
NM005
Z
Sheet 4
1. Gate WR to Adder
2. Gate LB to Adder
3. Gate IRCTR to Adder
4. Set T/C Bit to Compl (G80)
5. Gate +1 to Adder 31 (G92)
6. Set State 2 (D102)
NM005
IC - target (for back <8 test)
LB 25 = IRCTR 25
No
Yes
1. Gate +1 to Adder 25 (G88)
NM005
Save address for backup
1. Gate R2/X to SLCX Reg
2. Gate IR to SLCIR
3. Reset SLCB Reg
NM005
RR Insn Format
No
Yes
1. Gate B to SLCB Reg (Overrides Reset)
NM005
Temp 1 Bfr Fetch Tag On (F74)
No
Temp 2 Bfr Fetch Tag On (G74)
No
1. Set Ppln 3 Op Dbl Bit (J81) and Fetch Bit (J82)
2. Ton Fetch Tag for Temp 1 Bfr (F74)
3. Gate Temp 1 Sink Addr to Ppln 3 Op Reg (J80)
4. Ton TFM Tgr (F101)
5. Og Ppln 3 and WR to MSCE
NM005
3. Temp 1 fetch (target)
BXQ Tgr On (E107)
No
BCQ Tgr On (E106)
No
State 2 (D102)
Yes
MSCE Acptng
Yes
Loop Tgr On (E102)
No
Yes
LB ≤ WR ≤ UB
No
Yes
(Br in Array)
BC Insn Decode
Yes
BC Uncond Tgr On (E103)
No
CB<8 will not be set if (Branch insn dbl wd = target dbl wd addr). (Branch insn does not end on dbl wd boundary)
1. Set CB < 8 Per Adder
NM027
TFM Tgr On (F101)
No
Yes
Generate target +1 address
1. Gate +1 to Adder 28 (G90)
2. Ton Fetch Tag for Temp 2 Bfr (G74)
3. Gate Temp 2 Bfr Sink Address to Ppln 2 Op Reg (J79, 80)
4. Set Dbl (H87) and Fetch (H82) Bits into Ppln 2 Op Reg
5. Gate WR to Adder
NM027
1. Ton CBIA Tgr (J102)
2. Tof TFM Tgr (F101)
NM027
1. Hot 1s Reset (See Notes on Sheet 5 for definition)
2. Ton Cond Mode Tgr (G101)
3. Set State 0
4. Step IR
NM027
D
Sheet 4

DIAGRAM 5-11. BRANCH ON CONDITION SEQUENCE (SHEET 3 OF 5)

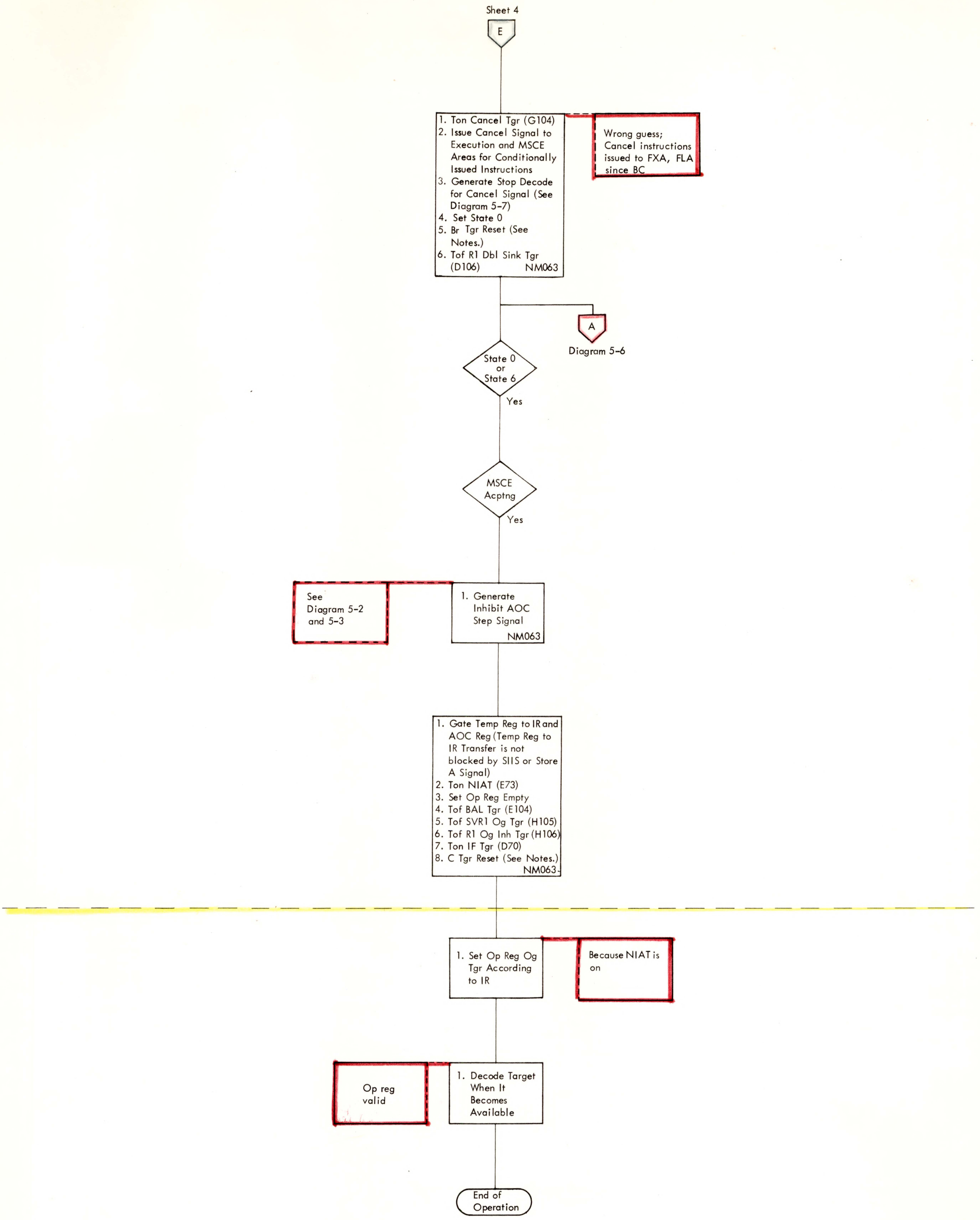

Sheet 3
D
1. Gate Adder to WR
2. Set Ppln 3 Op Reg from Ppln 2 Op Reg
3. Og Ppln 3 Op to MSCE NP405
Ppln 2 and Ppln 3 control
Diagram 5-11 (Sheet 3)
Z
1. Decode Instructions and Issue to Appropriate Area Conditionally
2. Sample CC with Mark Reg Each Cycle to Determine Outcome of Branch
3. Block Decode of Instructions when CC becomes Valid
1. Decode down "no branch" path if not in loop mode
Decode down branch path if in loop mode and IR = SLCIR
BC Test
Cond Tgr On (G101)
Yes
CC Valid (G102)
Yes
Branch Taken (F104)
No
Yes
1. Generate Insn Fetch Inhibit Signal
State 0
Yes
No
BCT or BX Tgr On
No
Yes
1. Ton Req GP Priority Tgr NM043
Correct Guess
C Quick Tgr On (J101)
Yes
No
Wrong Guess
C Quick Tgr On (J101)
Yes
No
Wrong Guess
SLT = LB
Yes
No
1. Tof Loop Tgr (E102) NM063
Inner loop; outer loop must be BCT or BX
State 0 or State 6 or Shift Decode
Yes
No
1. Ton IF Tgr (D70) NM065
1. Ton Activate Tgr (G103)
2. C Tgr Reset (See Note on Sheet 5) NM065
Will turn off Cndl mode and allow normal processing
CB &lt; 8 Tgr On (J103)
Yes
No
1. Gate Temp Reg to LB and SLT Regs NM063
Loop Tgr On (E102)
Yes
No
1. Ton Loop Tgr (E102) NM063
A
Diag. 5-6
State 0 or State 6
No
Yes
MSCE Acptng
No
Yes
C BIA Tgr On (J103)
Yes
No
Does Any Stack Pos Contain Both CC and CM Bit
Yes
No
1. Tof CC Val Tgr (G102) NM065
End of Operation
1. Ton Temp Fetch Tgr (D72)
2. Ton Temp Fetch 1 Tgr (F102)
3. Ton Temp Fetch 2 Tgr (F103)
4. Tof Initial Fetch Tgr (D71) NM063
2 and 3 To transfer temp bfrs to array
1. Tof Loop Tgr * (E102)
2. Gate Temp Reg to UB and LB Regs NM063
1. Ton Init Fetch Tgr (D71) NM063
Initialize array
TFM Tgr On (F101)
Yes
No
E
Sheet 5

CC=	Mask bit on	If mask bit 10 is on, branch if CC = 2
0	8	
1	9	
2	10	
3	11	

Cycles are dependent upon the time it takes for the CC to become valid.

* Note: Loop Tgr on over off.

DIAGRAM 5-11. BRANCH ON CONDITION SEQUENCE (SHEET 4 OF 5)


Sheet 4
E
1. Ton Cancel Tgr (G104)
2. Issue Cancel Signal to Execution and MSCE Areas for Conditionally Issued Instructions
3. Generate Stop Decode for Cancel Signal (See Diagram 5-7)
4. Set State 0
5. Br Tgr Reset (See Notes.)
6. Tof R1 Dbl Sink Tgr (D106)
NM063
Wrong guess; Cancel instructions issued to FXA, FLA since BC
A
Diagram 5-6
State 0 or State 6
Yes
MSCE Acptng
Yes
See Diagram 5-2 and 5-3
1. Generate Inhibit AOC Step Signal
NM063
1. Gate Temp Reg to IR and AOC Reg (Temp Reg to IR Transfer is not blocked by SIIS or Store A Signal)
2. Ton NIAT (E73)
3. Set Op Reg Empty
4. Tof BAL Tgr (E104)
5. Tof SVR1 Og Tgr (H105)
6. Tof R1 Og Inh Tgr (H106)
7. Ton IF Tgr (D70)
8. C Tgr Reset (See Notes.)
NM063
1. Set Op Reg Og Tgr According to IR
Because NIAT is on
Op reg valid
1. Decode Target When It Becomes Available
End of Operation

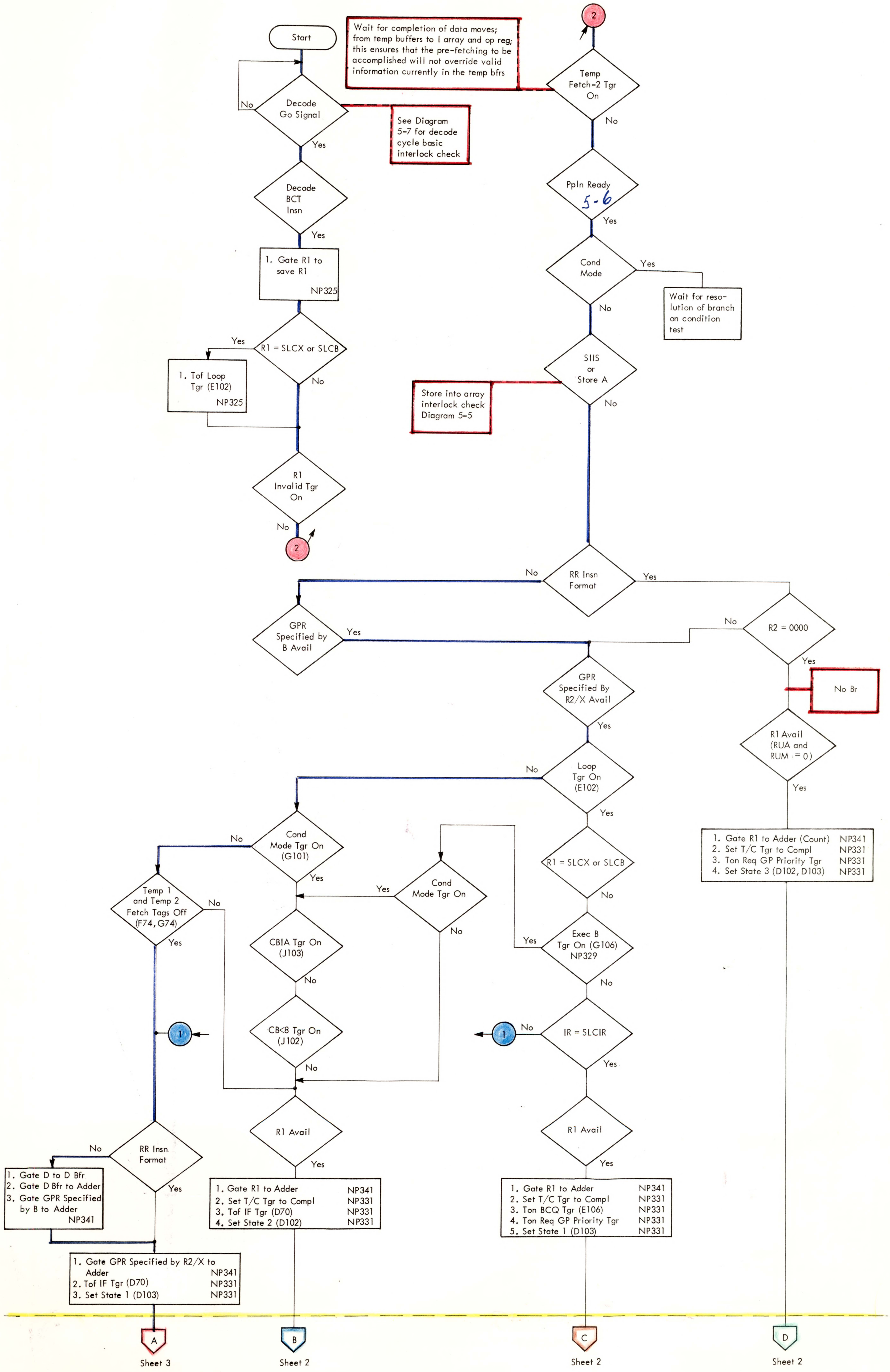
Notes:

1. Reset Og to Adder = GPR Reset and Hot 1s Reset
2. GPR Reset = (a) Tof R1 Og Tgrs
   (b) Tof X/R2; B; D; Og Tgrs
   (c) Tof WR Og Tgrs
   (d) Tof Temp Reg Og Tgrs
3. Hot is Reset = (a) Set T/C Tgr to T
   (b) Tof +1 to Adder 31; 29; 28; TAT; and 25
   (c) Tof SVIR Og Tgr
   (d) Tof LB Og Tgr
   (e) Tof IRCTR Og Tgr
4. Br Tgr Reset = (a) Tof Br Tgr
   (b) Tof BIA Tgr
   (c) Tof BIA 1 Tgr
   (d) Tof B<8 Tgr
   (e) Tof Exec Tgr
   (f) Tof BCUNCON Tgr
5. C Tgr Reset = (a) Tof Cond Tgr
   (b) Tof CBIA Tgr
   (c) Tof CB<8 Tgr
   (d) Tof C Quick Tgr
   (e) Tof TFM Tgr

DIAGRAM 5-11. BRANCH ON CONDITION SEQUENCE (SHEET 5 OF 5)

Objectives:

1. Decode and process a BCT instruction.
2. Check all interlocks which may cancel the execution of this instruction.
3. Condition mode must be removed before processing of the BCT instruction can be completed.
4. The BCT instruction can either break, establish, or retain a loop mode of operation.



Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock check
Decode BCT Insn
Yes
1. Gate R1 to save R1
NP325
R1 = SLCX or SLCB
Yes
No
1. Tof Loop Tgr (E102)
NP325
R1 Invalid Tgr On
No
2
Wait for completion of data moves; from temp buffers to I array and op reg; this ensures that the pre-fetching to be accomplished will not override valid information currently in the temp bfrs
2
Temp Fetch-2 Tgr On
No
Ppln Ready
5-6
Yes
Cond Mode
Yes
Wait for resolution of branch on condition test
No
SIIS or Store A
Store into array interlock check Diagram 5-5
No
RR Insn Format
No
Yes
GPR Specified by B Avail
Yes
R2 = 0000
No
Yes
No Br
GPR Specified By R2/X Avail
Yes
R1 Avail (RUA and RUM = 0)
Yes
1. Gate R1 to Adder (Count) NP341
2. Set T/C Tgr to Compl NP331
3. Ton Req GP Priority Tgr NP331
4. Set State 3 (D102, D103) NP331
Loop Tgr On (E102)
No
Yes
Cond Mode Tgr On (G101)
No
Yes
Cond Mode Tgr On
Yes
No
R1 = SLCX or SLCB
No
Exec B Tgr On (G106) NP329
Yes
No
Temp 1 and Temp 2 Fetch Tags Off (F74, G74)
No
Yes
CBIA Tgr On (J103)
No
CB<8 Tgr On (J102)
No
1
1
No
IR = SLCIR
Yes
R1 Avail
Yes
R1 Avail
Yes
RR Insn Format
No
Yes
1. Gate D to D Bfr
2. Gate D Bfr to Adder
3. Gate GPR Specified by B to Adder
NP341
1. Gate R1 to Adder NP341
2. Set T/C Tgr to Compl NP331
3. Tof IF Tgr (D70) NP331
4. Set State 2 (D102) NP331
1. Gate R1 to Adder NP341
2. Set T/C Tgr to Compl NP331
3. Ton BCQ Tgr (E106) NP331
4. Ton Req GP Priority Tgr NP331
5. Set State 1 (D103) NP331
1. Gate GPR Specified by R2/X to Adder NP341
2. Tof IF Tgr (D70) NP331
3. Set State 1 (D103) NP331
A
Sheet 3
B
Sheet 2
C
Sheet 2
D
Sheet 2

DIAGRAM 5-12. BRANCH ON COUNT SEQUENCE (SHEET 1 OF 5)

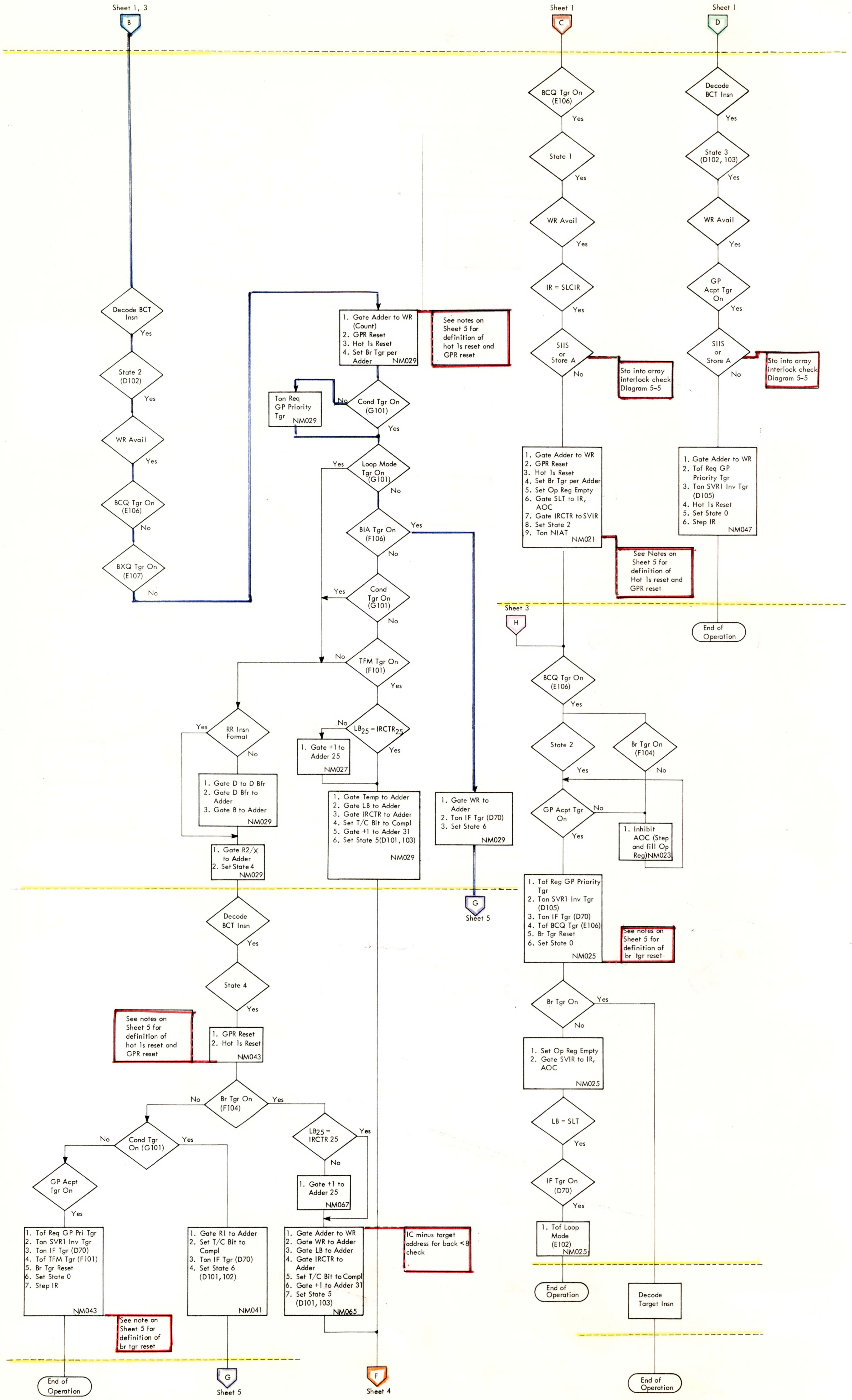


DIAGRAM 5-12. BRANCH ON COUNT SEQUENCE (SHEET 2 OF 5)

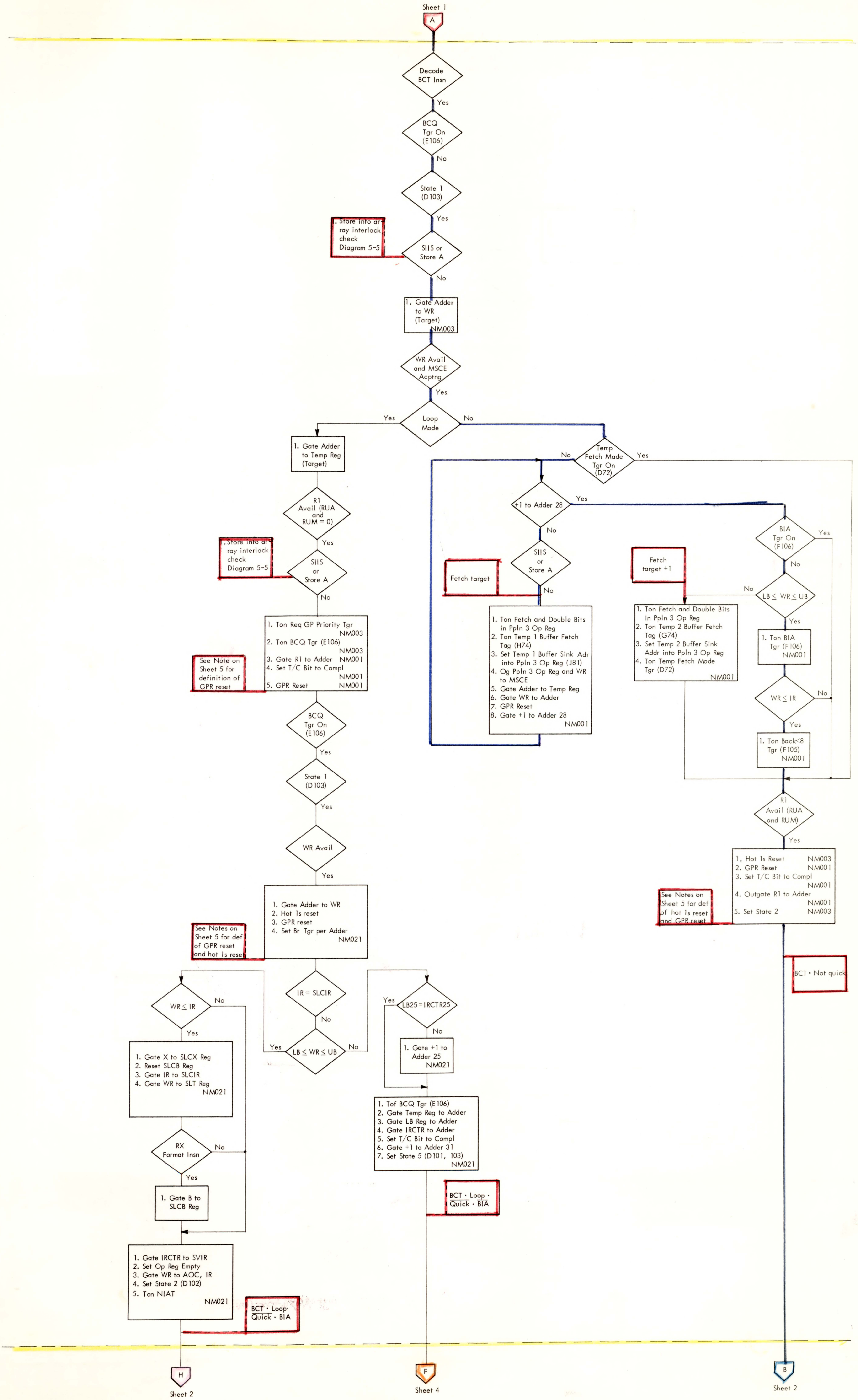

Sheet 1
A
Decode BCT Insn
Yes
BCQ Tgr On (E106)
No
State 1 (D103)
Yes
. Store into array interlock check Diagram 5-5
SIIS or Store A
No
1. Gate Adder to WR (Target) NM003
WR Avail and MSCE Acptng
Yes
Loop Mode
Yes
No
1. Gate Adder to Temp Reg (Target)
R1 Avail (RUA and RUM = 0)
Yes
. Store into array interlock check Diagram 5-5
SIIS or Store A
No
1. Ton Req GP Priority Tgr NM003
2. Ton BCQ Tgr (E106) NM003
3. Gate R1 to Adder NM001
4. Set T/C Bit to Compl NM001
5. GPR Reset NM001
See Note on Sheet 5 for definition of GPR reset
BCQ Tgr On (E106)
Yes
State 1 (D103)
Yes
WR Avail
Yes
1. Gate Adder to WR
2. Hot 1s reset
3. GPR reset
4. Set Br Tgr per Adder NM021
See Notes on Sheet 5 for def of GPR reset and hot 1s reset
WR ≤ IR
No
Yes
1. Gate X to SLCX Reg
2. Reset SLCB Reg
3. Gate IR to SLCIR
4. Gate WR to SLT Reg NM021
RX Format Insn
No
Yes
1. Gate B to SLCB Reg
1. Gate IRCTR to SVIR
2. Set Op Reg Empty
3. Gate WR to AOC, IR
4. Set State 2 (D102)
5. Ton NIAT NM021
BCT · Loop · Quick · BIA
H
Sheet 2
IR = SLCIR
No
Yes
LB ≤ WR ≤ UB
No
Yes
LB25 = IRCTR25
No
1. Gate +1 to Adder 25 NM021
1. Tof BCQ Tgr (E106)
2. Gate Temp Reg to Adder
3. Gate LB Reg to Adder
4. Gate IRCTR to Adder
5. Set T/C Bit to Compl
6. Gate +1 to Adder 31
7. Set State 5 (D101, 103) NM021
BCT · Loop · Quick · BIA
F
Sheet 4
Temp Fetch Made Tgr On (D72)
No
Yes
+1 to Adder 28
Yes
No
SIIS or Store A
No
Fetch target
1. Ton Fetch and Double Bits in Ppln 3 Op Reg
2. Ton Temp 1 Buffer Fetch Tag (H74)
3. Set Temp 1 Buffer Sink Adr into Ppln 3 Op Reg (J81)
4. Og Ppln 3 Op Reg and WR to MSCE
5. Gate Adder to Temp Reg
6. Gate WR to Adder
7. GPR Reset
8. Gate +1 to Adder 28 NM001
Fetch target +1
1. Ton Fetch and Double Bits in Ppln 3 Op Reg
2. Ton Temp 2 Buffer Fetch Tag (G74)
3. Set Temp 2 Buffer Sink Addr into Ppln 3 Op Reg
4. Ton Temp Fetch Mode Tgr (D72) NM001
BIA Tgr On (F106)
Yes
No
LB ≤ WR ≤ UB
No
Yes
1. Ton BIA Tgr (F106) NM001
WR ≤ IR
No
Yes
1. Ton Back<8 Tgr (F105) NM001
R1 Avail (RUA and RUM)
Yes
1. Hot 1s Reset NM003
2. GPR Reset NM001
3. Set T/C Bit to Compl NM001
4. Outgate R1 to Adder NM001
5. Set State 2 NM003
See Notes on Sheet 5 for def of hot 1s reset and GPR reset
BCT · Not quick
B
Sheet 2

DIAGRAM 5-12. BRANCH ON COUNT SEQUENCE (SHEET 3 OF 5)

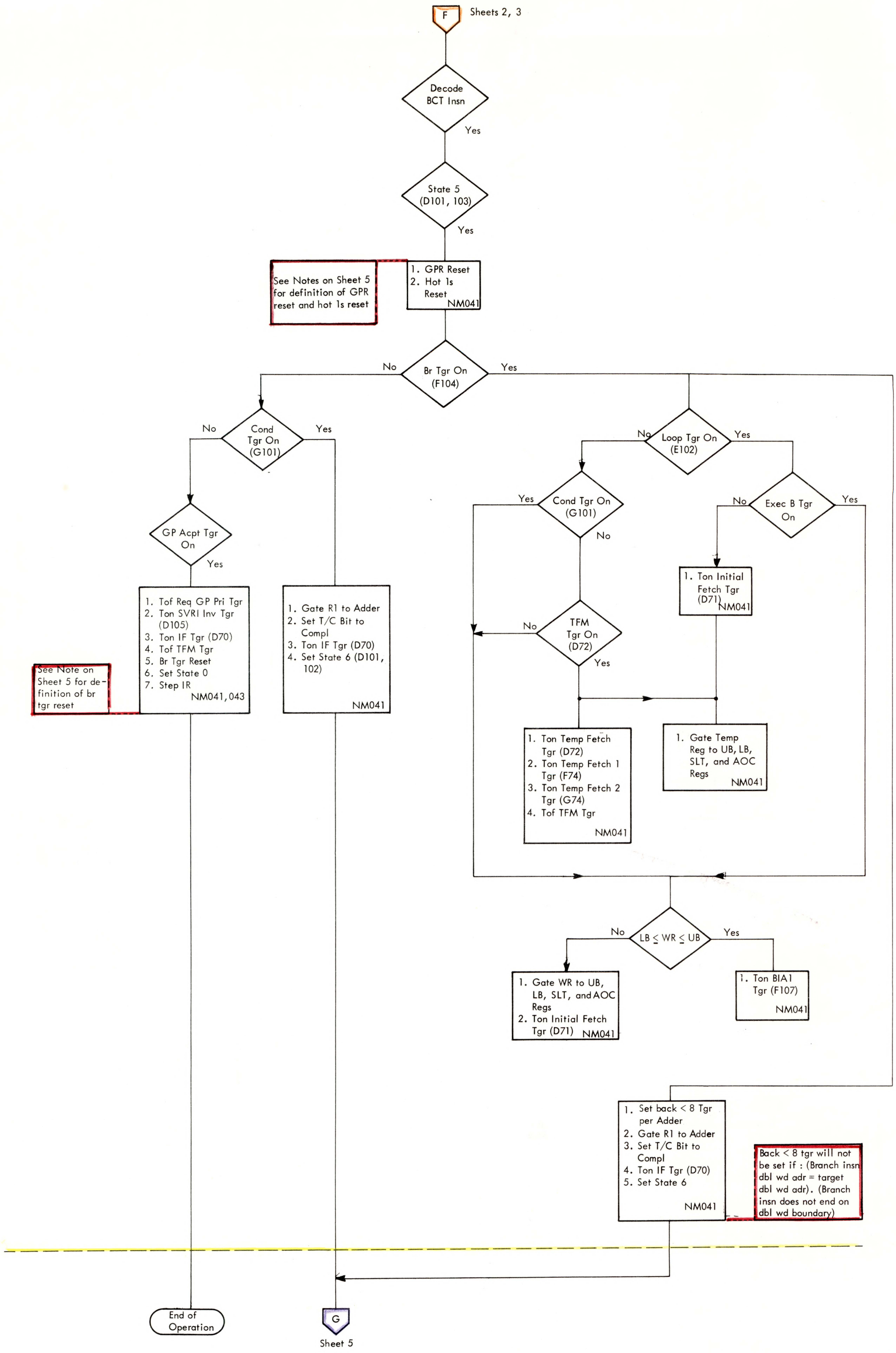
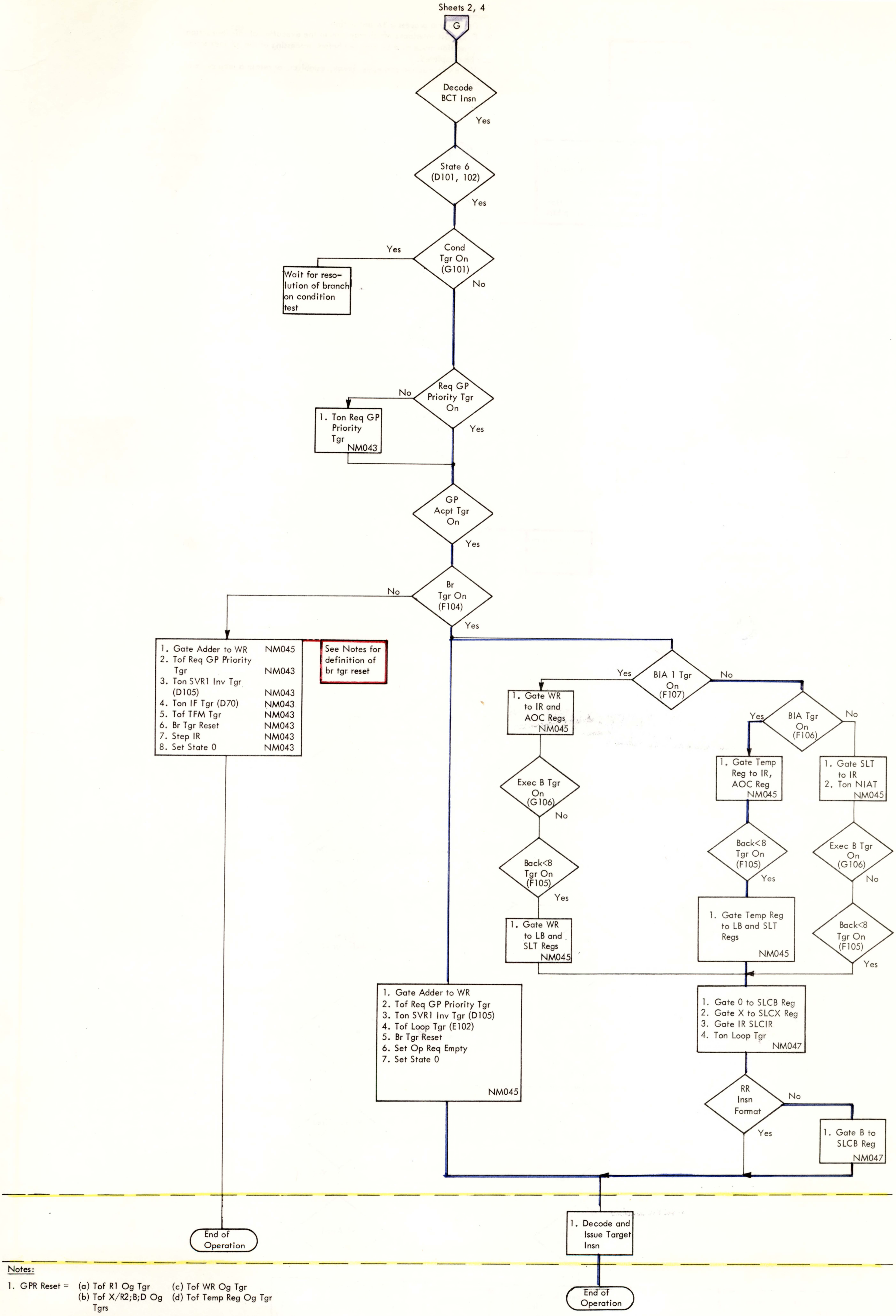


DIAGRAM 5-12. BRANCH ON COUNT SEQUENCE (SHEET 4 OF 5)



Notes:

1. GPR Reset = (a) Tof R1 Og Tgr (b) Tof X/R2;B;D Og Tgrs (c) Tof WR Og Tgr (d) Tof Temp Reg Og Tgr

2. Hot 1s Reset = (a) Set T/C Tgr to T (b) Tof SVIR Og Tgrs (c) Tof LB Og Tgrs (d) Tof IRCTR Og Tgr (e) Tof +1 to Adder 31, 29, 28, TAT, 25

3. Br Tgr Reset = (a) Tof Br Tgr (b) Tof BIA Tgr (c) Tof BIA1 Tgr (d) Tof Back<8 Tgr (e) Tof Exec B Tgr (f) Tof BCUNCONT

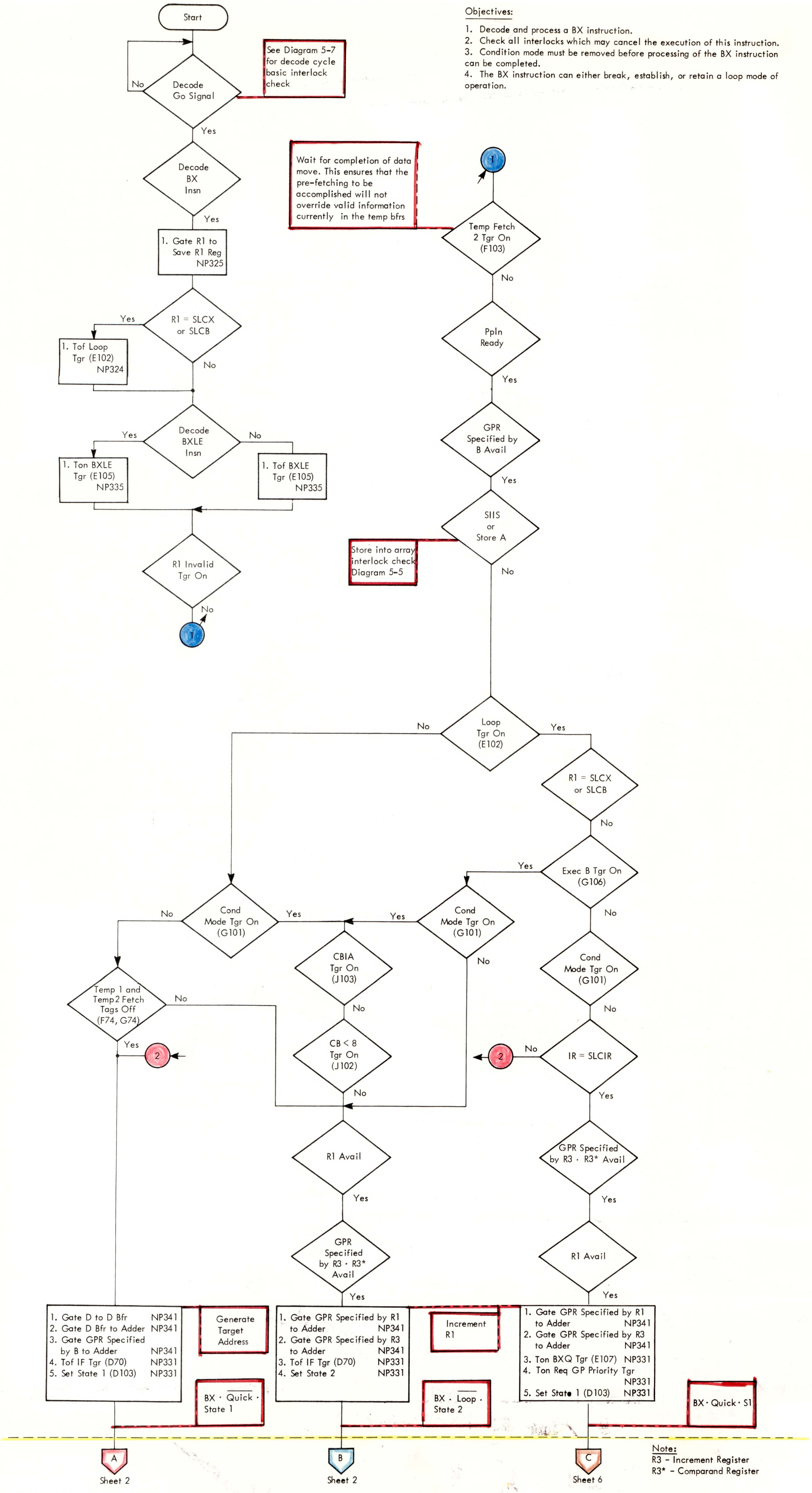
4. Minimum time from request to accept is 2 cycles.

5. CP request off and priority granted causes fixed point execution unit to gate WR into GPR.

DIAGRAM 5-12. BRANCH AND COUNT SEQUENCE (SHEET 5 OF 5)

Objectives:

1. Decode and process a BX instruction.
2. Check all interlocks which may cancel the execution of this instruction.
3. Condition mode must be removed before processing of the BX instruction can be completed.
4. The BX instruction can either break, establish, or retain a loop mode of operation.


Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock check
Decode BX Insn
Yes
1. Gate R1 to Save R1 Reg NP325
R1 = SLCX or SLCB
Yes
No
1. Tof Loop Tgr (E102) NP324
Decode BXLE Insn
Yes
No
1. Ton BXLE Tgr (E105) NP335
1. Tof BXLE Tgr (E105) NP335
R1 Invalid Tgr On
No
1
Wait for completion of data move. This ensures that the pre-fetching to be accomplished will not override valid information currently in the temp bfrs
1
Temp Fetch 2 Tgr On (F103)
No
Ppln Ready
Yes
GPR Specified by B Avail
Yes
SIIS or Store A
No
Store into array interlock check Diagram 5-5
Loop Tgr On (E102)
No
Yes
R1 = SLCX or SLCB
No
Exec B Tgr On (G106)
Yes
No
Cond Mode Tgr On (G101)
Yes
No
Cond Mode Tgr On (G101)
No
Yes
Cond Mode Tgr On (G101)
No
CBIA Tgr On (J103)
No
CB &lt; 8 Tgr On (J102)
No
Temp 1 and Temp 2 Fetch Tags Off (F74, G74)
No
Yes
2
2
No
IR = SLCIR
Yes
R1 Avail
Yes
GPR Specified by R3 · R3* Avail
Yes
GPR Specified by R3 · R3* Avail
Yes
R1 Avail
Yes
1. Gate D to D Bfr NP341
2. Gate D Bfr to Adder NP341
3. Gate GPR Specified by B to Adder NP341
4. Tof IF Tgr (D70) NP331
5. Set State 1 (D103) NP331
Generate Target Address
BX · Quick · State 1
1. Gate GPR Specified by R1 to Adder NP341
2. Gate GPR Specified by R3 to Adder NP341
3. Tof IF Tgr (D70) NP331
4. Set State 2 NP331
Increment R1
BX · Loop · State 2
1. Gate GPR Specified by R1 to Adder NP341
2. Gate GPR Specified by R3 to Adder NP341
3. Ton BXQ Tgr (E107) NP331
4. Ton Req GP Priority Tgr NP331
5. Set State 1 (D103) NP331
BX · Quick · S1
A
Sheet 2
B
Sheet 2
C
Sheet 6

Note:
R3 – Increment Register
R3* – Comparand Register

DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 1 OF 6)

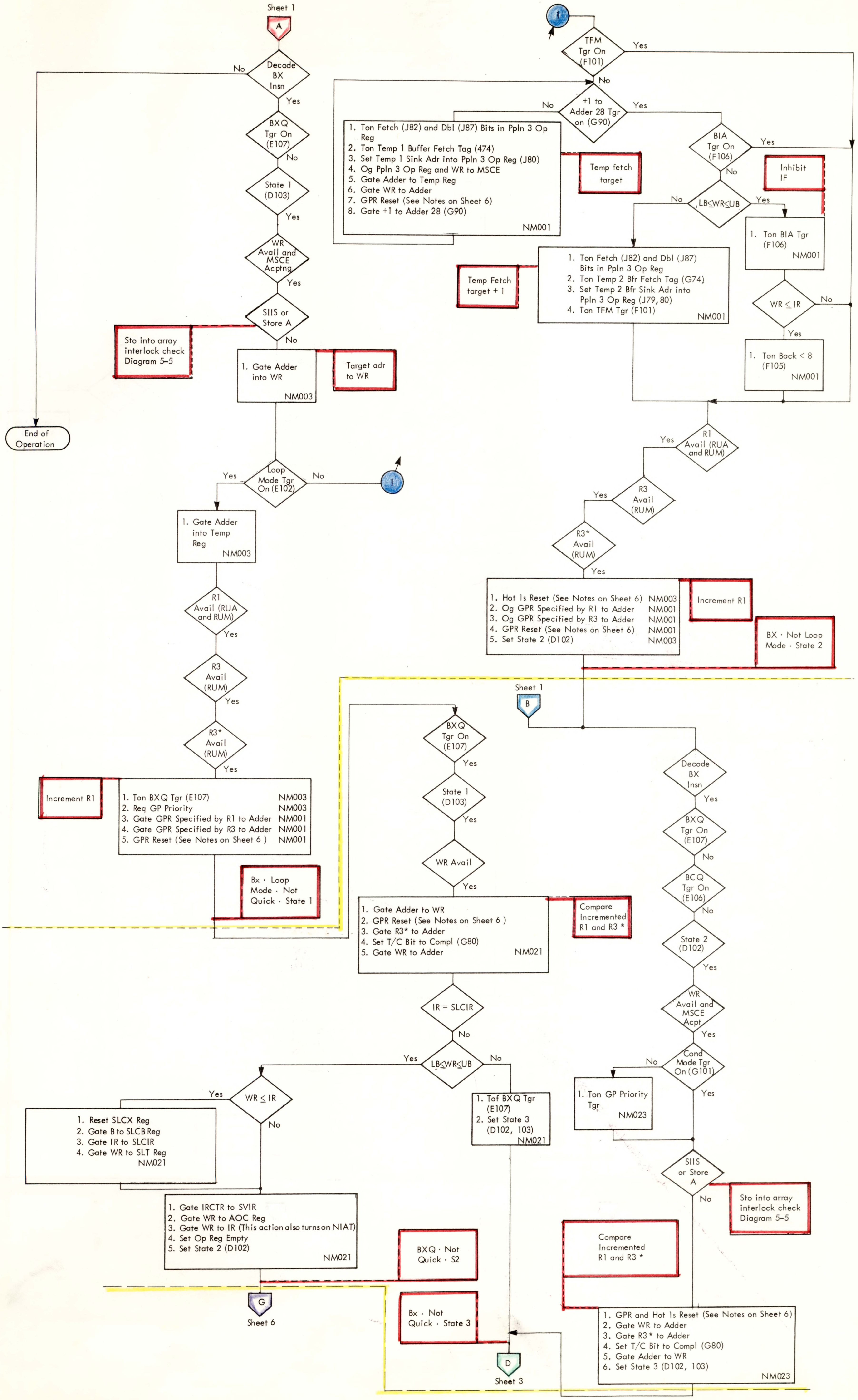

Sheet 1
A
Decode BX Insn
No
Yes
End of Operation
BXQ Tgr On (E107)
No
State 1 (D103)
Yes
WR Avail and MSCE Acptng
Yes
SIIS or Store A
Sto into array interlock check Diagram 5-5
No
1. Gate Adder into WR
NM003
Target adr to WR
Loop Mode Tgr On (E102)
Yes
No
1
1. Gate Adder into Temp Reg
NM003
R1 Avail (RUA and RUM)
Yes
R3 Avail (RUM)
Yes
R3* Avail (RUM)
Yes
Increment R1
1. Ton BXQ Tgr (E107) NM003
2. Req GP Priority NM003
3. Gate GPR Specified by R1 to Adder NM001
4. Gate GPR Specified by R3 to Adder NM001
5. GPR Reset (See Notes on Sheet 6) NM001
Bx · Loop Mode · Not Quick · State 1
1
TFM Tgr On (F101)
Yes
No
+1 to Adder 28 Tgr on (G90)
No
Yes
1. Ton Fetch (J82) and Dbl (J87) Bits in Ppln 3 Op Reg
2. Ton Temp 1 Buffer Fetch Tag (474)
3. Set Temp 1 Sink Adr into Ppln 3 Op Reg (J80)
4. Og Ppln 3 Op Reg and WR to MSCE
5. Gate Adder to Temp Reg
6. Gate WR to Adder
7. GPR Reset (See Notes on Sheet 6)
8. Gate +1 to Adder 28 (G90)
NM001
Temp fetch target
BIA Tgr On (F106)
Yes
No
Inhibit IF
LB≤WR≤UB
No
Yes
1. Ton BIA Tgr (F106)
NM001
Temp Fetch target + 1
1. Ton Fetch (J82) and Dbl (J87) Bits in Ppln 3 Op Reg
2. Ton Temp 2 Bfr Fetch Tag (G74)
3. Set Temp 2 Bfr Sink Adr into Ppln 3 Op Reg (J79, 80)
4. Ton TFM Tgr (F101)
NM001
WR ≤ IR
No
Yes
1. Ton Back < 8 (F105)
NM001
R1 Avail (RUA and RUM)
Yes
R3 Avail (RUM)
Yes
R3* Avail (RUM)
Yes
1. Hot 1s Reset (See Notes on Sheet 6) NM003
2. Og GPR Specified by R1 to Adder NM001
3. Og GPR Specified by R3 to Adder NM001
4. GPR Reset (See Notes on Sheet 6) NM001
5. Set State 2 (D102) NM003
Increment R1
BX · Not Loop Mode · State 2
Sheet 1
B
BXQ Tgr On (E107)
Yes
State 1 (D103)
Yes
WR Avail
Yes
1. Gate Adder to WR
2. GPR Reset (See Notes on Sheet 6)
3. Gate R3* to Adder
4. Set T/C Bit to Compl (G80)
5. Gate WR to Adder
NM021
Compare Incremented R1 and R3 *
IR = SLCIR
No
LB≤WR≤UB
Yes
No
WR ≤ IR
Yes
No
1. Reset SLCX Reg
2. Gate B to SLCB Reg
3. Gate IR to SLCIR
4. Gate WR to SLT Reg
NM021
1. Tof BXQ Tgr (E107)
2. Set State 3 (D102, 103)
NM021
1. Gate IRCTR to SVIR
2. Gate WR to AOC Reg
3. Gate WR to IR (This action also turns on NIAT)
4. Set Op Reg Empty
5. Set State 2 (D102)
NM021
BXQ · Not Quick · S2
G
Sheet 6
Bx · Not Quick · State 3
D
Sheet 3
Decode BX Insn
Yes
BXQ Tgr On (E107)
No
BCQ Tgr On (E106)
No
State 2 (D102)
Yes
WR Avail and MSCE Acpt
Yes
Cond Mode Tgr On (G101)
No
Yes
1. Ton GP Priority Tgr
NM023
SIIS or Store A
No
Sto into array interlock check Diagram 5-5
Compare Incremented R1 and R3 *
1. GPR and Hot 1s Reset (See Notes on Sheet 6)
2. Gate WR to Adder
3. Gate R3* to Adder
4. Set T/C Bit to Compl (G80)
5. Gate Adder to WR
6. Set State 3 (D102, 103)
NM023

DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 2 OF 6)

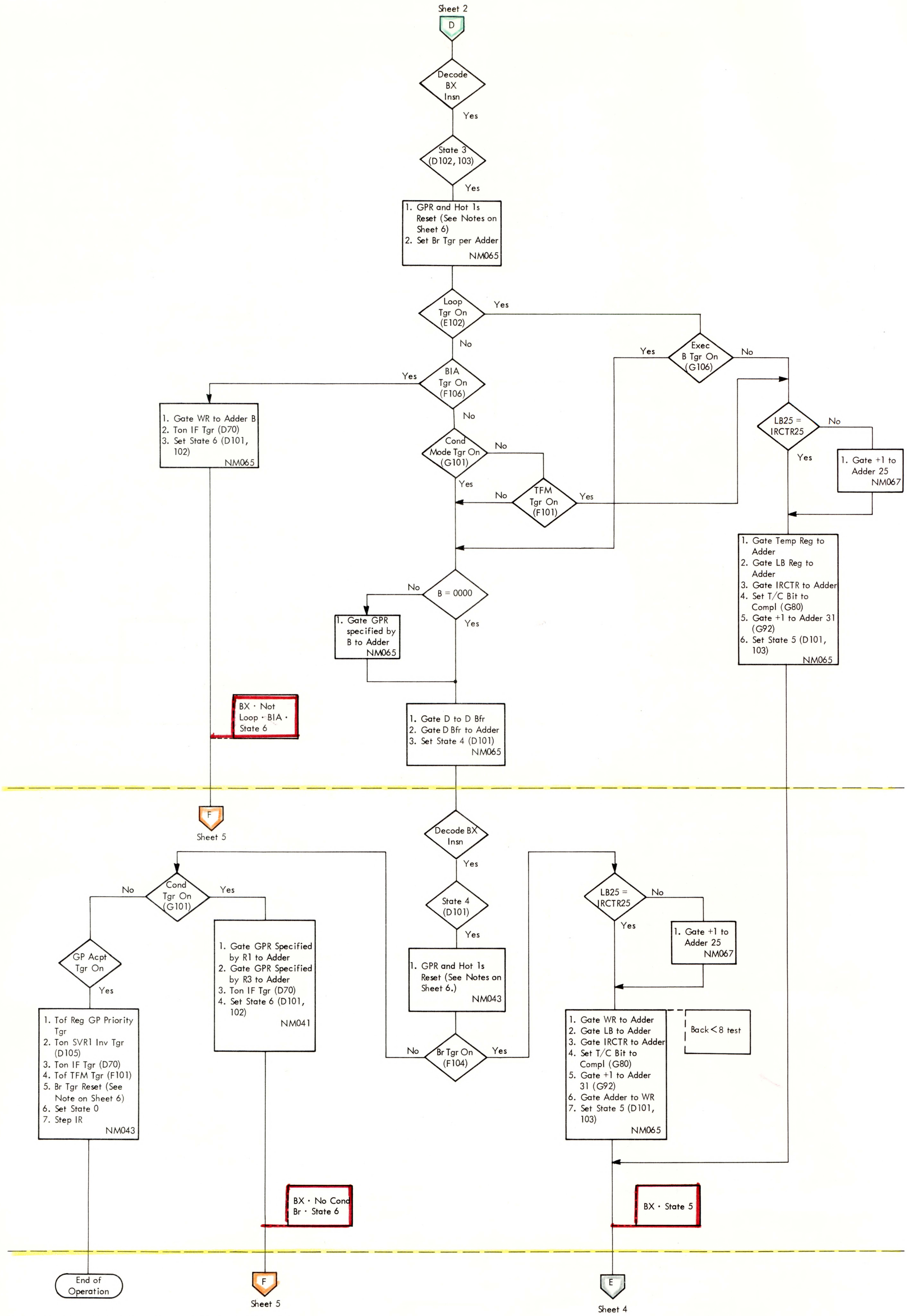


DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 3 OF 6)

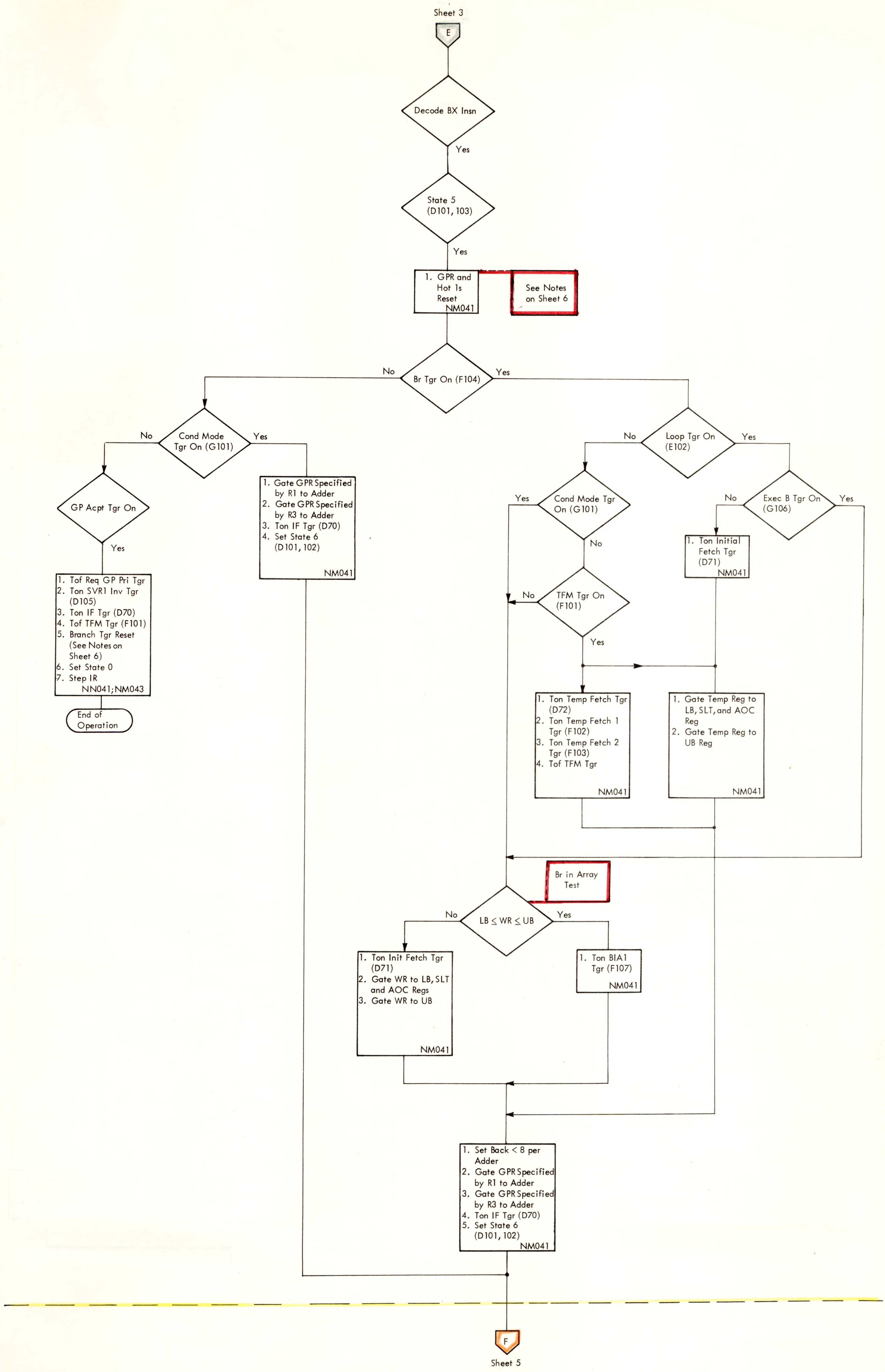


DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 4 OF 6)

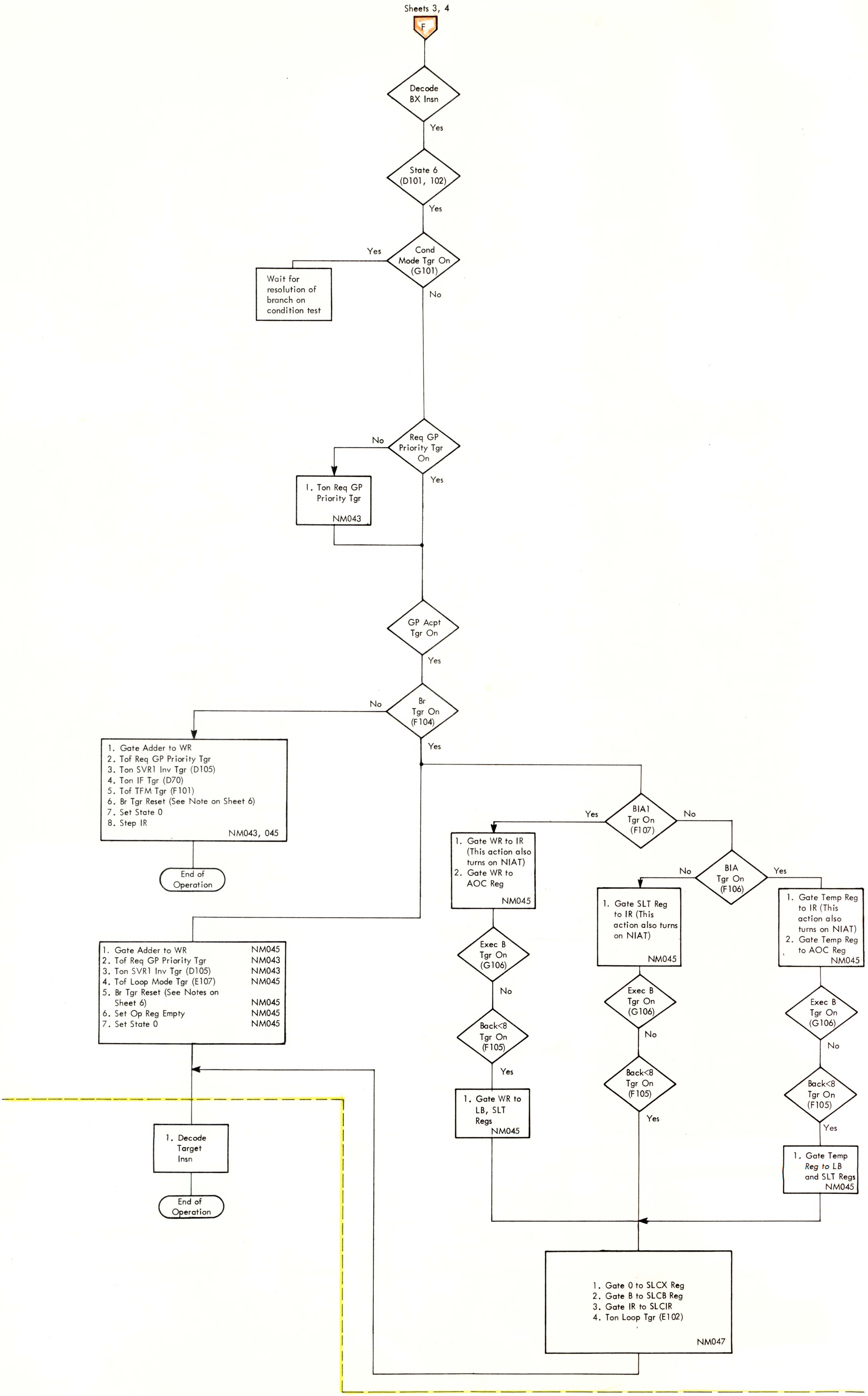
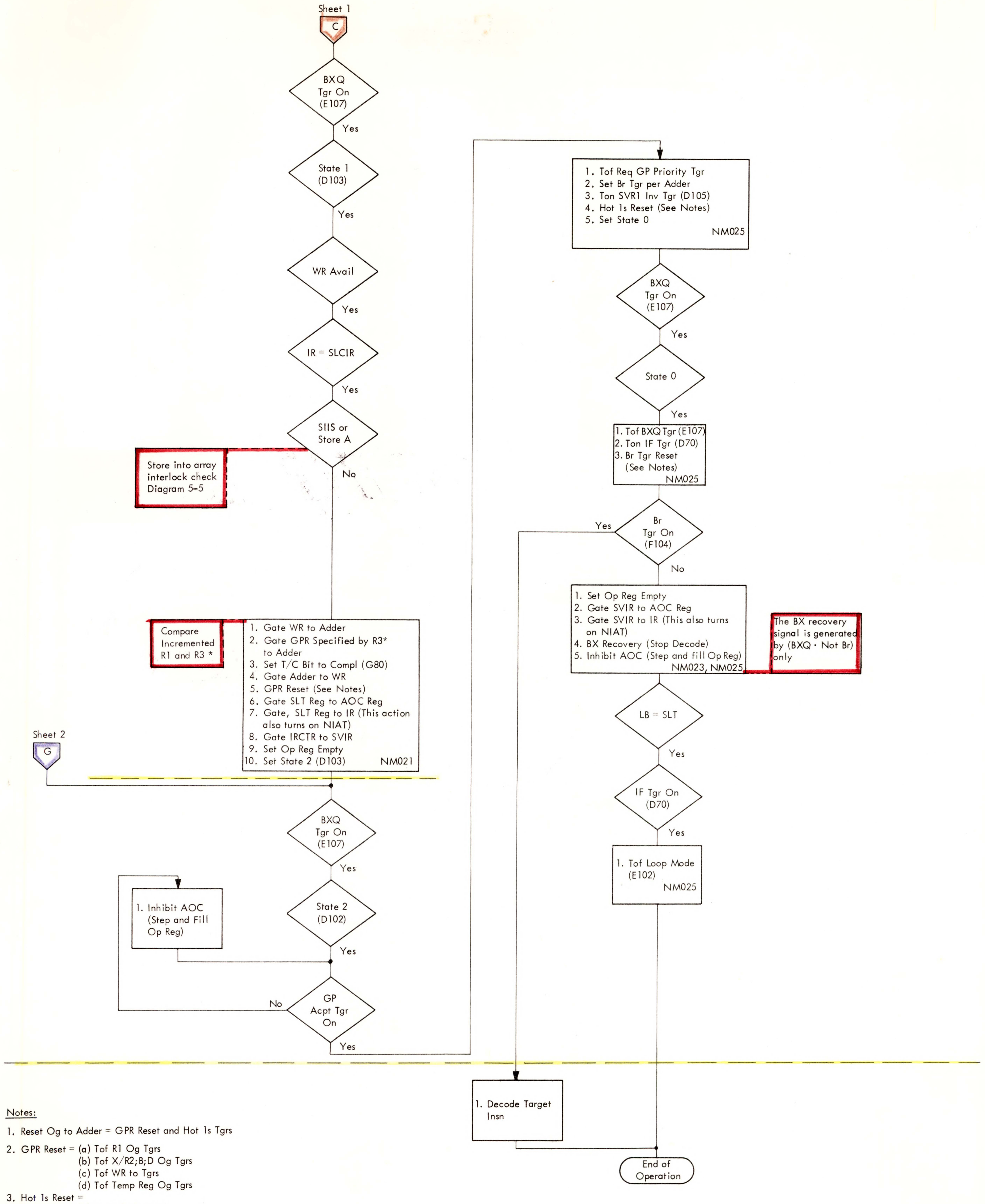


DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 5 OF 6)



Notes:

1. Reset Og to Adder = GPR Reset and Hot 1s Tgrs

2. GPR Reset = (a) Tof R1 Og Tgrs
(b) Tof X/R2;B;D Og Tgrs
(c) Tof WR to Tgrs
(d) Tof Temp Reg Og Tgrs

3. Hot 1s Reset =
(a) Set T/C Tgr to T
(b) Tof SVIR Og Tgr
(c) Tof LB Og Tgr
(d) Tof IRCTR Og Tgr
(e) Tof +1 to Adder 31, 29, 28, TAT, 25

4. Br Tgr Reset =
(a) Tof Br Tgr
(b) Tof BIA Tgr
(c) Tof BIA1 Tgr
(d) Tof Back <8 Tgr
(e) Tof Exec B Tgr
(f) Tof BCUNCONT

5. Minimum time from request to accept is 2 cycles

6. CP request off and priority granted, causes fixed point execution unit to gate WR into GPR

DIAGRAM 5-13. BRANCH ON INDEX SEQUENCE (SHEET 6 OF 6)

Objectives:

1. Decode and process a BAL instruction.
2. Check all interlocks which may cancel the execution of the instruction.
3. Set the rightmost 32 bits of the PSW, including the updated instruction address into the GPR staging area.
4. Notify FXA to read link data into GPR specified by R1.

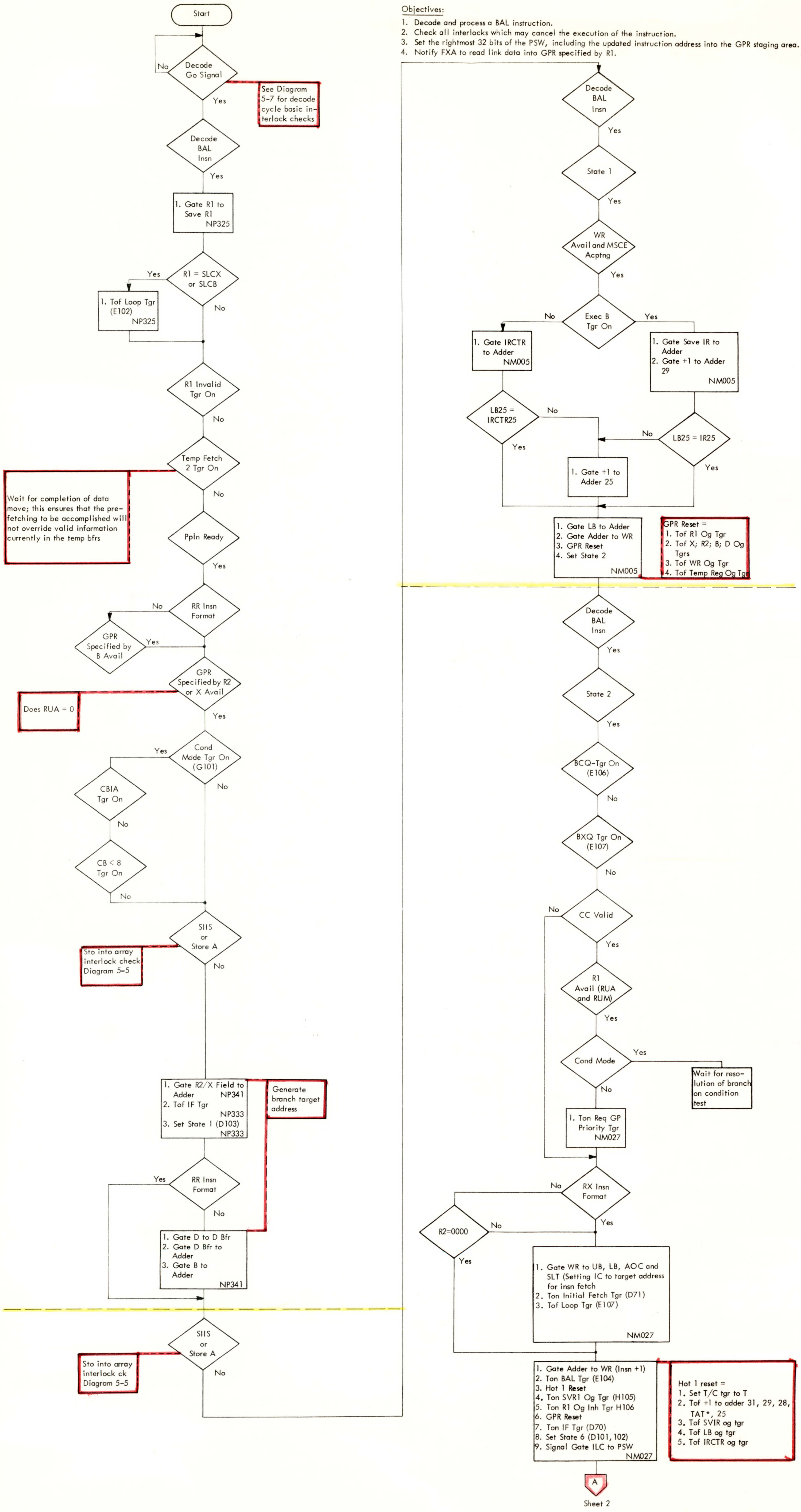


DIAGRAM 5-14. BRANCH AND LINK SEQUENCE (SHEET 1 OF 2)

Note:
* TAT - Timer Adr Tgr

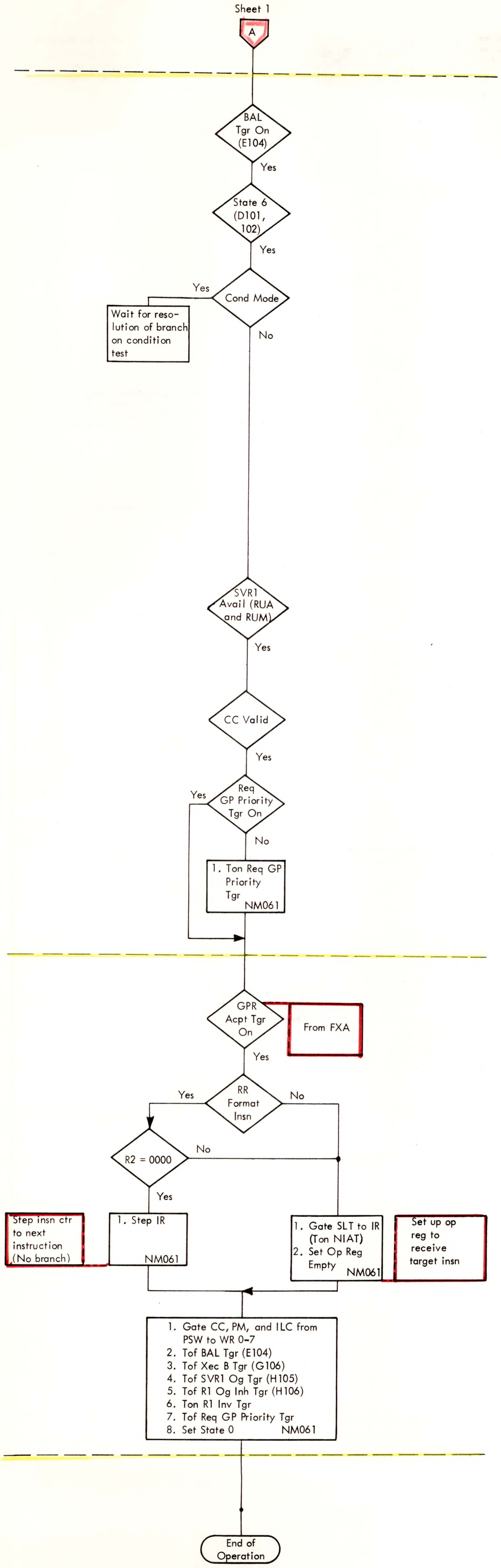


DIAGRAM 5-14. BRANCH AND LINK SEQUENCE (SHEET 2 OF 2)

Objectives:

1. To permit execution of an instruction that is not residing in the current instruction sequence.
2. To check all exceptions that can occur during the processing of an execute instruction.
3. Once the subject instruction is executed; update instruction counter, prepare op reg to recover to normal instruction processing, advance to next instruction, and resume instruction fetching.

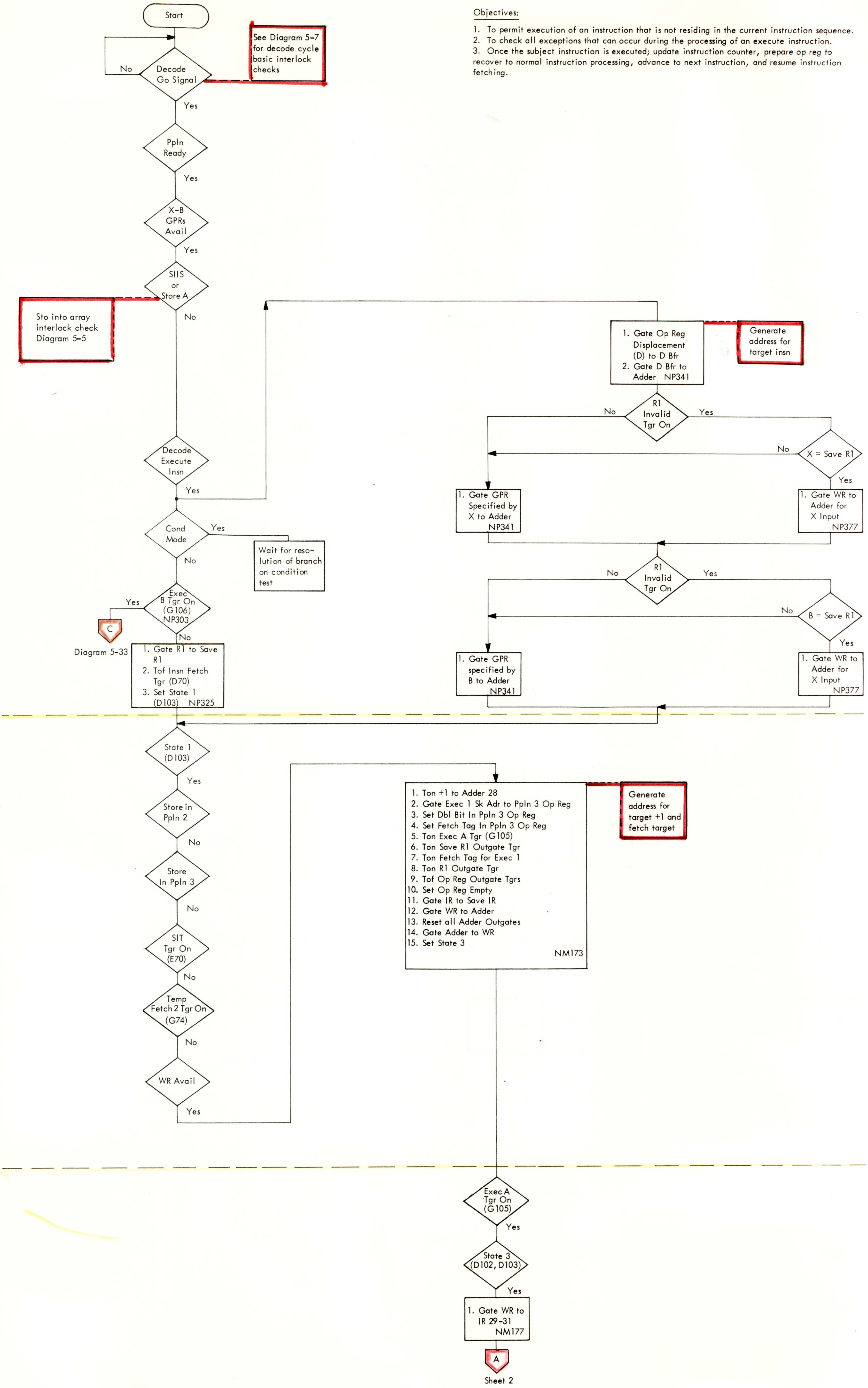

Start
No
Decode Go Signal
See Diagram 5-7 for decode cycle basic interlock checks
Yes
Ppln Ready
Yes
X-B GPRs Avail
Yes
SIIS or Store A
Sto into array interlock check Diagram 5-5
No
Decode Execute Insn
Yes
Cond Mode
Yes
Wait for resolution of branch on condition test
No
Exec B Tgr On (G106) NP303
Yes
C
Diagram 5-33
No
1. Gate R1 to Save R1
2. Tof Insn Fetch Tgr (D70)
3. Set State 1 (D103) NP325
1. Gate Op Reg Displacement (D) to D Bfr
2. Gate D Bfr to Adder NP341
Generate address for target insn
No
R1 Invalid Tgr On
Yes
No
X = Save R1
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
No
R1 Invalid Tgr On
Yes
No
B = Save R1
Yes
1. Gate GPR specified by B to Adder NP341
1. Gate WR to Adder for X Input NP377
State 1 (D103)
Yes
Store in Ppln 2
No
Store In Ppln 3
No
SIT Tgr On (E70)
No
Temp Fetch 2 Tgr On (G74)
No
WR Avail
Yes
1. Ton +1 to Adder 28
2. Gate Exec 1 Sk Adr to Ppln 3 Op Reg
3. Set Dbl Bit In Ppln 3 Op Reg
4. Set Fetch Tag In Ppln 3 Op Reg
5. Ton Exec A Tgr (G105)
6. Ton Save R1 Outgate Tgr
7. Ton Fetch Tag for Exec 1
8. Ton R1 Outgate Tgr
9. Tof Op Reg Outgate Tgrs
10. Set Op Reg Empty
11. Gate IR to Save IR
12. Gate WR to Adder
13. Reset all Adder Outgates
14. Gate Adder to WR
15. Set State 3
NM173
Generate address for target +1 and fetch target
Exec A Tgr On (G105)
Yes
State 3 (D102, D103)
Yes
1. Gate WR to IR 29-31 NM177
A
Sheet 2

DIAGRAM 5-15. EXECUTE SEQUENCE (SHEET 1 OF 3)

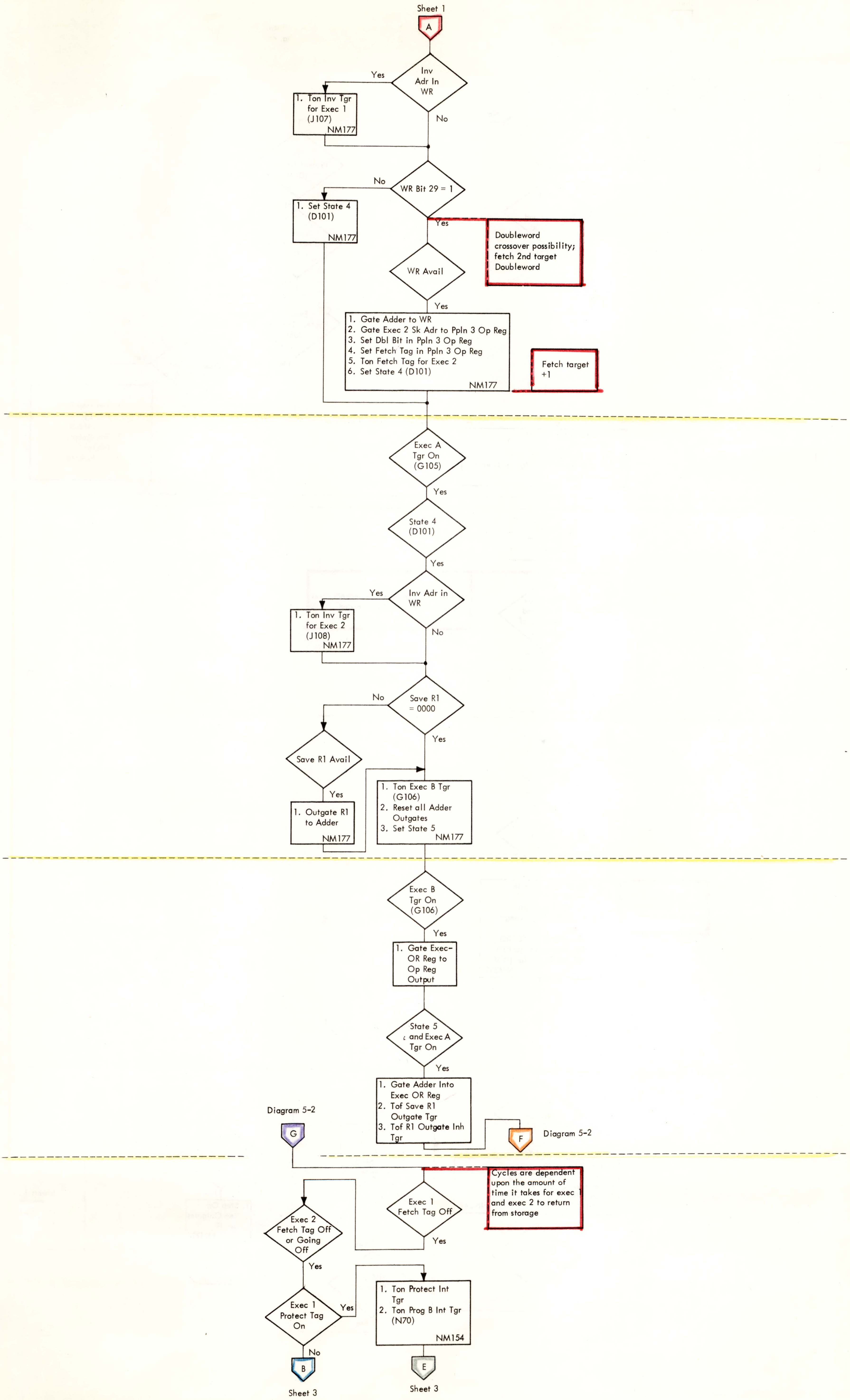

Sheet 1
A
Inv Adr In WR
Yes
No
1. Ton Inv Tgr for Exec 1 (J107)
NM177
WR Bit 29 = 1
No
Yes
1. Set State 4 (D101)
NM177
Doubleword crossover possibility; fetch 2nd target Doubleword
WR Avail
Yes
1. Gate Adder to WR
2. Gate Exec 2 Sk Adr to Ppln 3 Op Reg
3. Set Dbl Bit in Ppln 3 Op Reg
4. Set Fetch Tag in Ppln 3 Op Reg
5. Ton Fetch Tag for Exec 2
6. Set State 4 (D101)
NM177
Fetch target +1
Exec A Tgr On (G105)
Yes
State 4 (D101)
Yes
Inv Adr in WR
Yes
No
1. Ton Inv Tgr for Exec 2 (J108)
NM177
Save R1 = 0000
No
Yes
Save R1 Avail
Yes
1. Outgate R1 to Adder
NM177
1. Ton Exec B Tgr (G106)
2. Reset all Adder Outgates
3. Set State 5
NM177
Exec B Tgr On (G106)
Yes
1. Gate Exec-OR Reg to Op Reg Output
State 5 and Exec A Tgr On
Yes
1. Gate Adder Into Exec OR Reg
2. Tof Save R1 Outgate Tgr
3. Tof R1 Outgate Inh Tgr
Diagram 5-2
G
F
Diagram 5-2
Cycles are dependent upon the amount of time it takes for exec 1 and exec 2 to return from storage
Exec 1 Fetch Tag Off
Yes
Exec 2 Fetch Tag Off or Going Off
Yes
Exec 1 Protect Tag On
Yes
No
1. Ton Protect Int Tgr
2. Ton Prog B Int Tgr (N70)
NM154
B
Sheet 3
E
Sheet 3

DIAGRAM 5-15. EXECUTE SEQUENCE (SHEET 2 OF 3)

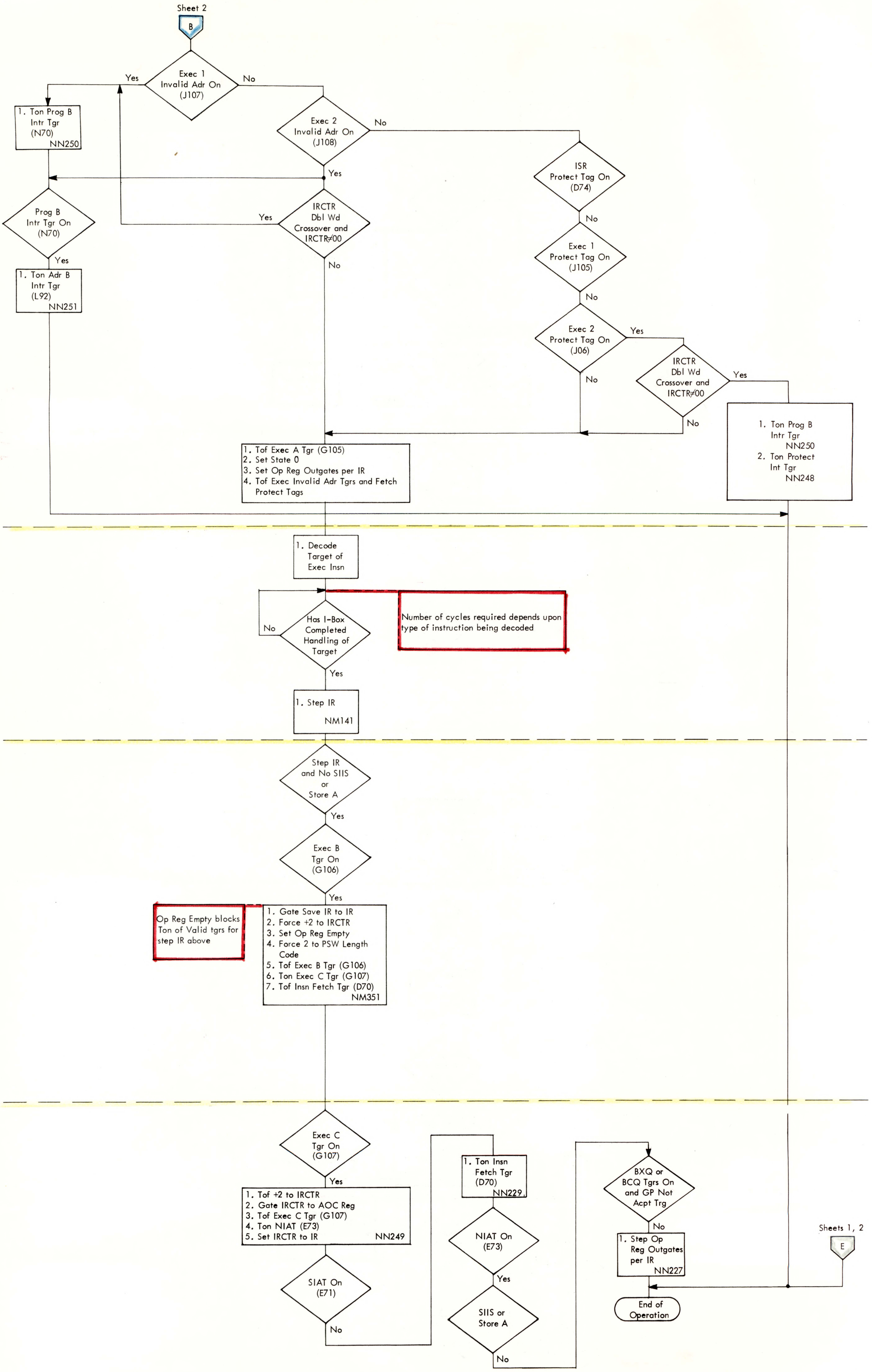

Sheet 2
B
Exec 1 Invalid Adr On (J107)
Yes
No
1. Ton Prog B Intr Tgr (N70) NN250
Exec 2 Invalid Adr On (J108)
No
Yes
ISR Protect Tag On (D74)
Prog B Intr Tgr On (N70)
Yes
IRCTR Dbl Wd Crossover and IRCTR≠00
No
No
Yes
1. Ton Adr B Intr Tgr (L92) NN251
Exec 1 Protect Tag On (J105)
No
Exec 2 Protect Tag On (J06)
Yes
IRCTR Dbl Wd Crossover and IRCTR≠00
Yes
No
No
1. Ton Prog B Intr Tgr NN250
2. Ton Protect Int Tgr NN248
1. Tof Exec A Tgr (G105)
2. Set State 0
3. Set Op Reg Outgates per IR
4. Tof Exec Invalid Adr Tgrs and Fetch Protect Tags
1. Decode Target of Exec Insn
No
Has I-Box Completed Handling of Target
Number of cycles required depends upon type of instruction being decoded
Yes
1. Step IR NM141
Step IR and No SIIS or Store A
Yes
Exec B Tgr On (G106)
Yes
Op Reg Empty blocks Ton of Valid tgrs for step IR above
1. Gate Save IR to IR
2. Force +2 to IRCTR
3. Set Op Reg Empty
4. Force 2 to PSW Length Code
5. Tof Exec B Tgr (G106)
6. Ton Exec C Tgr (G107)
7. Tof Insn Fetch Tgr (D70) NM351
Exec C Tgr On (G107)
Yes
1. Tof +2 to IRCTR
2. Gate IRCTR to AOC Reg
3. Tof Exec C Tgr (G107)
4. Ton NIAT (E73)
5. Set IRCTR to IR NN249
SIAT On (E71)
No
1. Ton Insn Fetch Tgr (D70) NN229
NIAT On (E73)
Yes
SIIS or Store A
No
BXQ or BCQ Tgrs On and GP Not Acpt Trg
No
1. Step Op Reg Outgates per IR NN227
End of Operation
Sheets 1, 2
E

DIAGRAM 5-15. EXECUTE SEQUENCE (SHEET 3 OF 3)

Objectives:

1. PSW located at storage address specified by B1 and D1 fields.
2. Operand address must have its three low-order bits zero to designate a double word.
3. CPE must be in supervisory state.
4. Privileged operation interrupt will occur if CPE is in problem state.
5. Load PSW is only instruction available for entering problem or wait states.

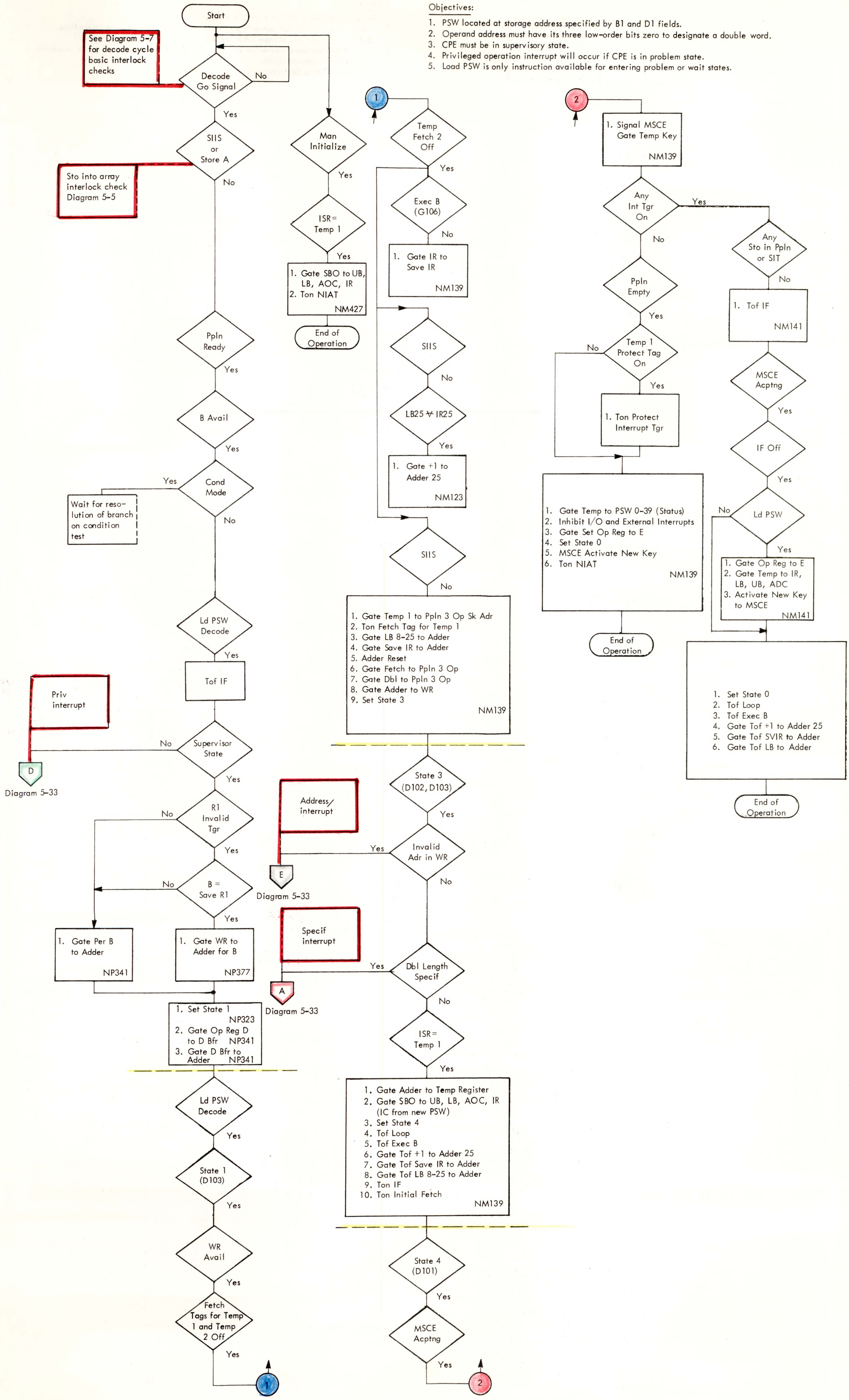


DIAGRAM 5-16. LOAD PSW SEQUENCE

Objectives:

1. Decode and issue a SPM or a SVC instruction.
2. Check all interlocks that can detain or cancel the execution of these instructions.
3. SPM - If no interrupt, set new mask and CC to PSW
4. SVC - When hardware into which the interrupt code is to be placed, become available, generate precise interrupt signal.
5. Step instruction to next instruction.

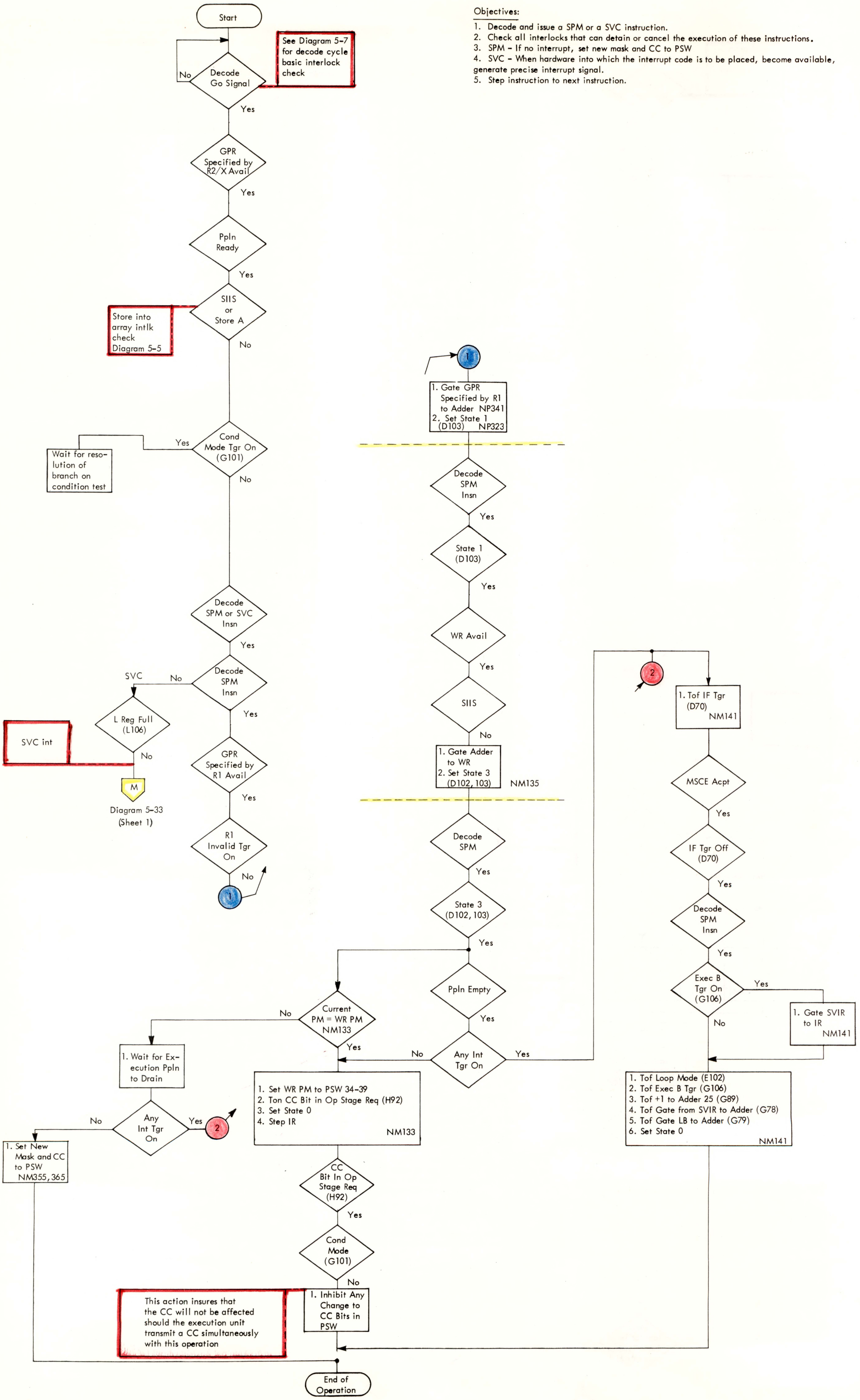

Start
No
Decode Go Signal
See Diagram 5-7 for decode cycle basic interlock check
Yes
GPR Specified by R2/X Avail
Yes
Ppln Ready
Yes
Store into array intlk check Diagram 5-5
SIIS or Store A
No
Yes
Wait for resolution of branch on condition test
Cond Mode Tgr On (G101)
No
Decode SPM or SVC Insn
Yes
SVC
No
Decode SPM Insn
Yes
L Reg Full (L106)
SVC int
No
M
Diagram 5-33 (Sheet 1)
GPR Specified by R1 Avail
Yes
R1 Invalid Tgr On
No
1
1
1. Gate GPR Specified by R1 to Adder NP341
2. Set State 1 (D103) NP323
Decode SPM Insn
Yes
State 1 (D103)
Yes
WR Avail
Yes
SIIS
No
1. Gate Adder to WR
2. Set State 3 (D102, 103)
NM135
Decode SPM
Yes
State 3 (D102, 103)
Yes
Ppln Empty
Yes
Current PM = WR PM NM133
No
Yes
1. Wait for Execution Ppln to Drain
Any Int Tgr On
No
Yes
2
1. Set New Mask and CC to PSW NM355, 365
Any Int Tgr On
No
Yes
1. Set WR PM to PSW 34-39
2. Ton CC Bit in Op Stage Req (H92)
3. Set State 0
4. Step IR
NM133
CC Bit In Op Stage Req (H92)
Yes
Cond Mode (G101)
No
This action insures that the CC will not be affected should the execution unit transmit a CC simultaneously with this operation
1. Inhibit Any Change to CC Bits in PSW
End of Operation
2
1. Tof IF Tgr (D70) NM141
MSCE Acpt
Yes
IF Tgr Off (D70)
Yes
Decode SPM Insn
Yes
Exec B Tgr On (G106)
Yes
1. Gate SVIR to IR NM141
No
1. Tof Loop Mode (E102)
2. Tof Exec B Tgr (G106)
3. Tof +1 to Adder 25 (G89)
4. Tof Gate from SVIR to Adder (G78)
5. Tof Gate LB to Adder (G79)
6. Set State 0
NM141

DIAGRAM 5-17. SET PROGRAM MASK AND SUPERVISOR CALL SEQUENCE

Objectives:

1. The byte located at the B1 and D1 instruction fields replaces system mask of current PSW.
2. VFLEU does byte selection for storage operand.
3. Correct system mask is stored because I/O and external interrupts are inhibited during instruction execution.

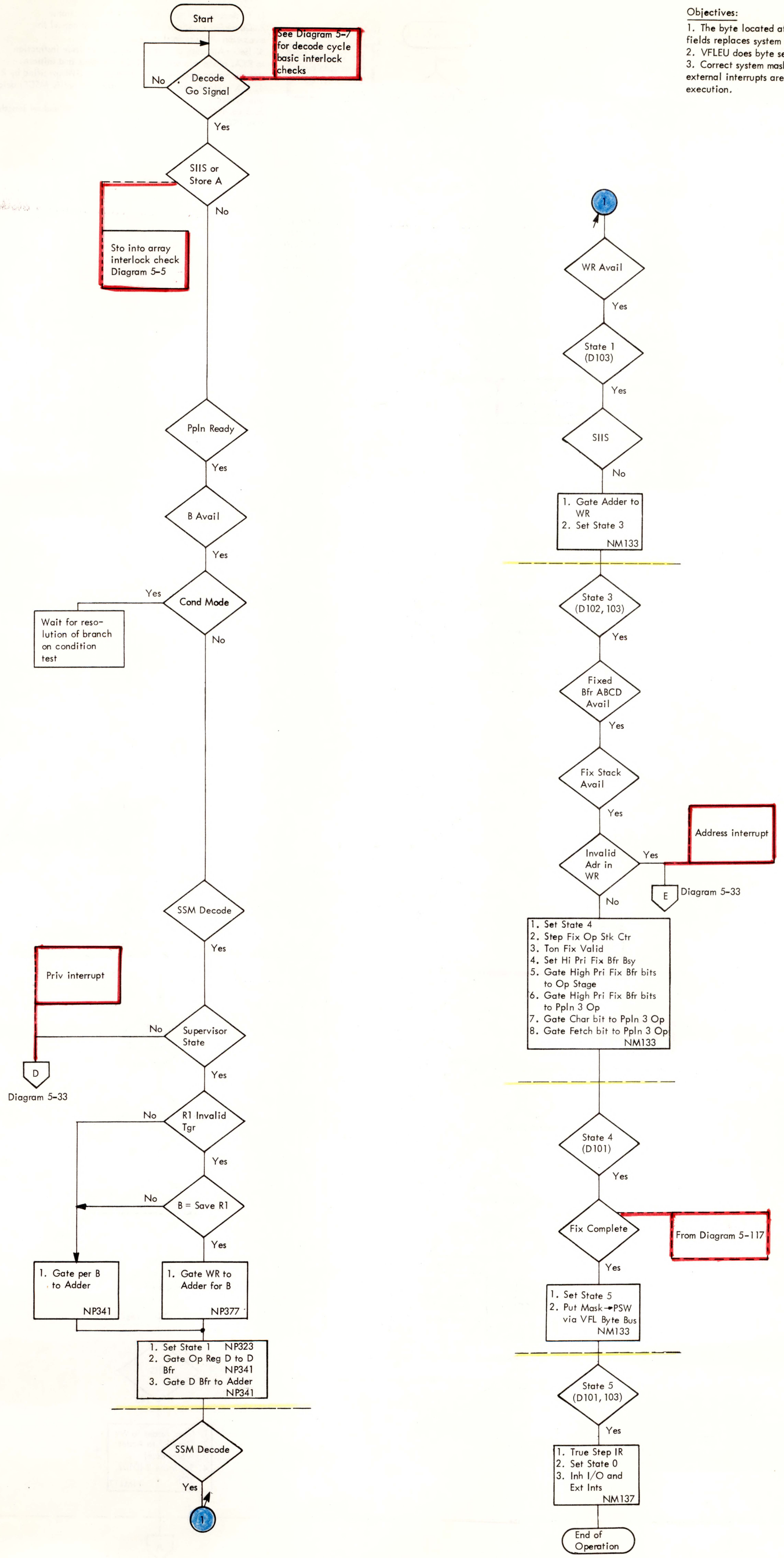


DIAGRAM 5-18. SET SYSTEM MASK SEQUENCE

Objectives:

1. To decode and issue a ISK or SSK instruction.
2. Check all interlocks that can detain or cancel the execution of these instructions.
3. ISK - Allocate buffers as appropriate; issue instruction to FXA; place address into MSCE interface and initiate storage fetch; establish sink interlock on GPR specified by R1.
4. SSK - Place data into MSCE interface; notify MSCE; delay one cycle.
5. Step instruction counter to next instruction and set length into PSW.

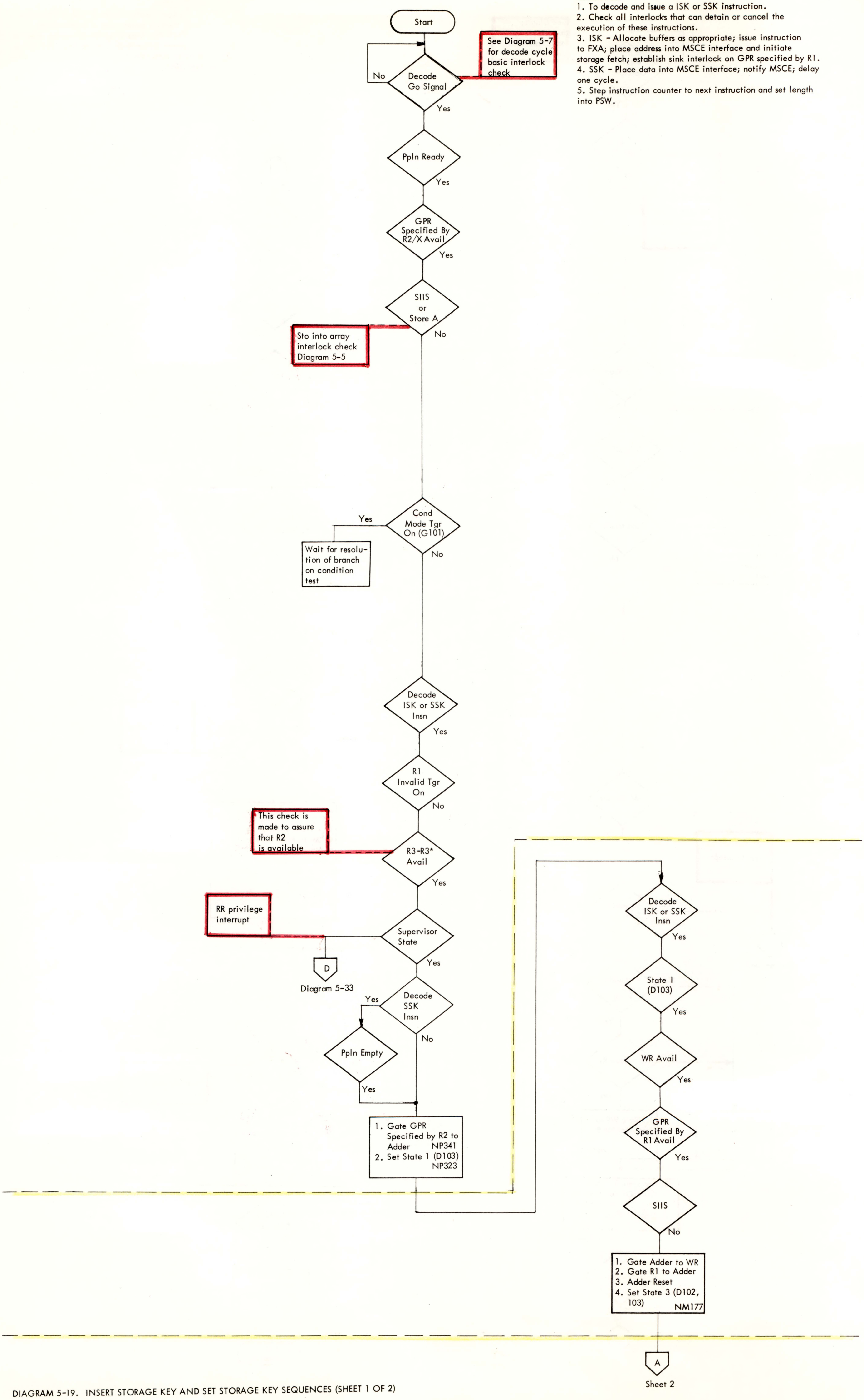


DIAGRAM 5-19. INSERT STORAGE KEY AND SET STORAGE KEY SEQUENCES (SHEET 1 OF 2)

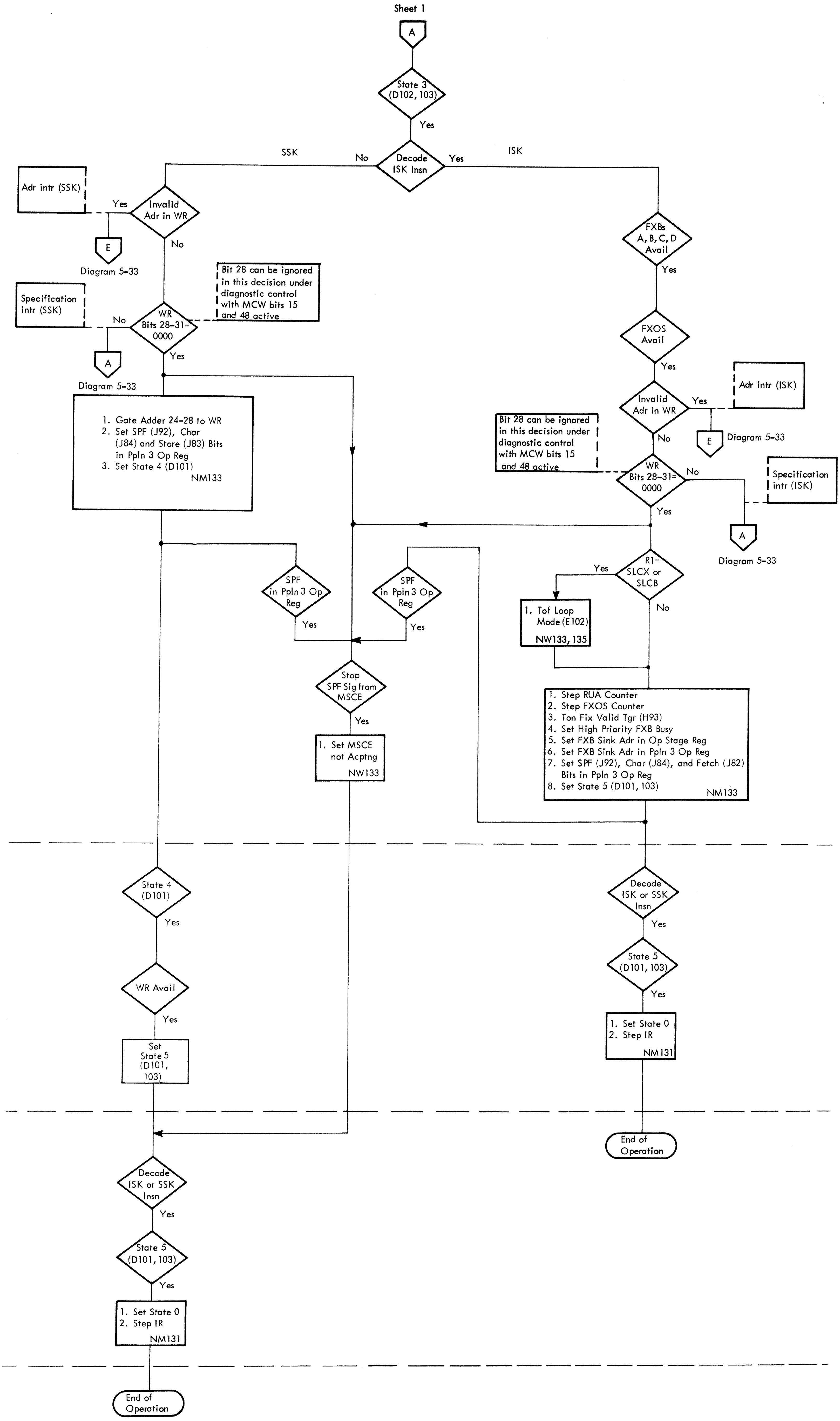


DIAGRAM 5-19. INSERT STORAGE KEY AND SET STORAGE KEY SEQUENCES (SHEET 2 OF 2)

Objectives:

1. Decode and issue a RDD or WRD instruction.
2. Check all interlocks that can detain or cancel these instructions.
3. RDD - Generate address and test for valid storage; check for FXOS and SAR availability; issue instruction to FXA; wait for FXA completion of this instruction before proceeding to decode cycle.
4. WRD - Generate address and test for valid storage; check for 'FXOS and FXB availability; issue instruction to FXA and proceed to decode cycle.

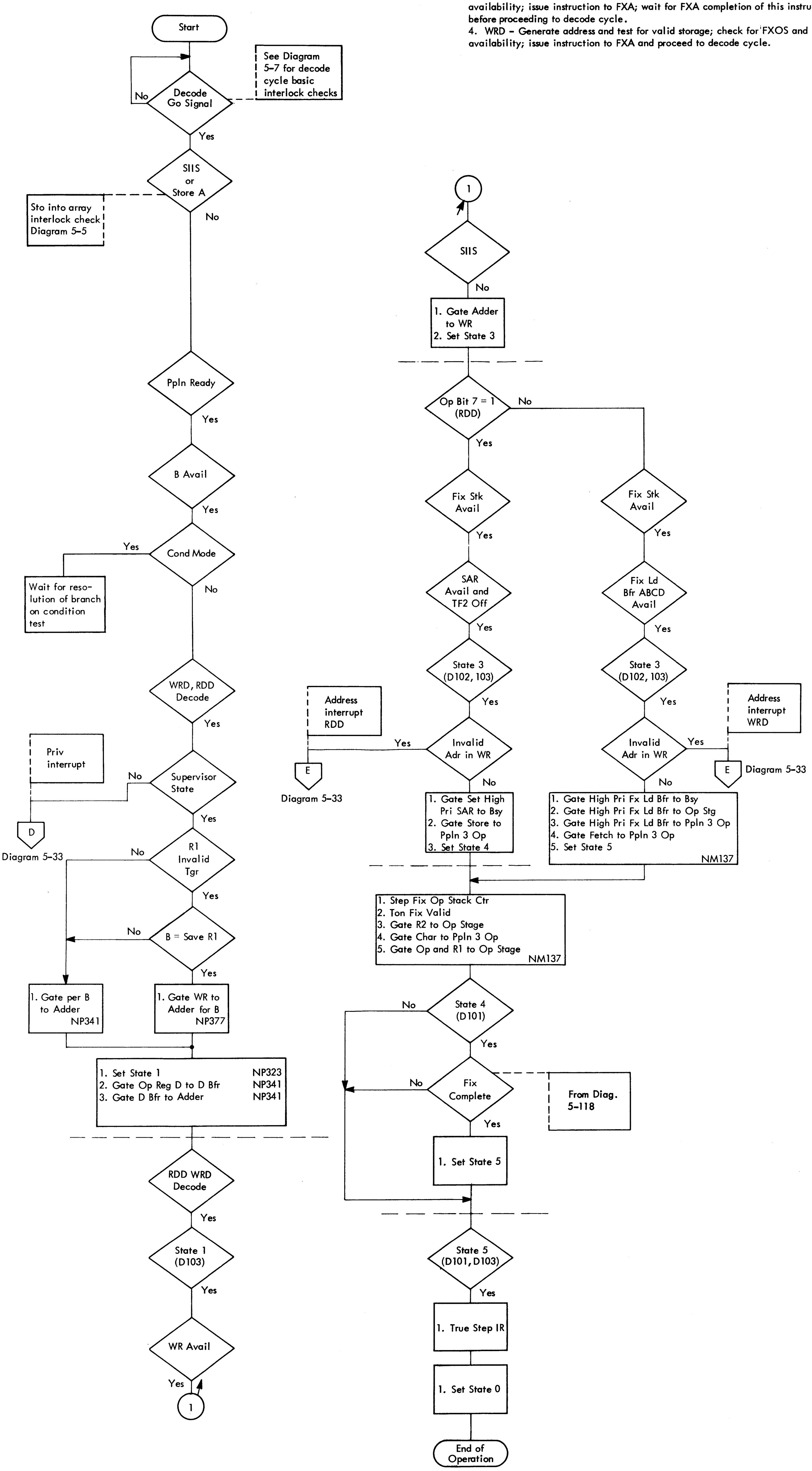


DIAGRAM 5-20. READ, WRITE DIRECT SEQUENCE

Objectives:

1. Decode and issue diagnose instruction.
2. Set MCW register which controls CPE and channel functions.
3. Verification of proper CPU equipment functions.

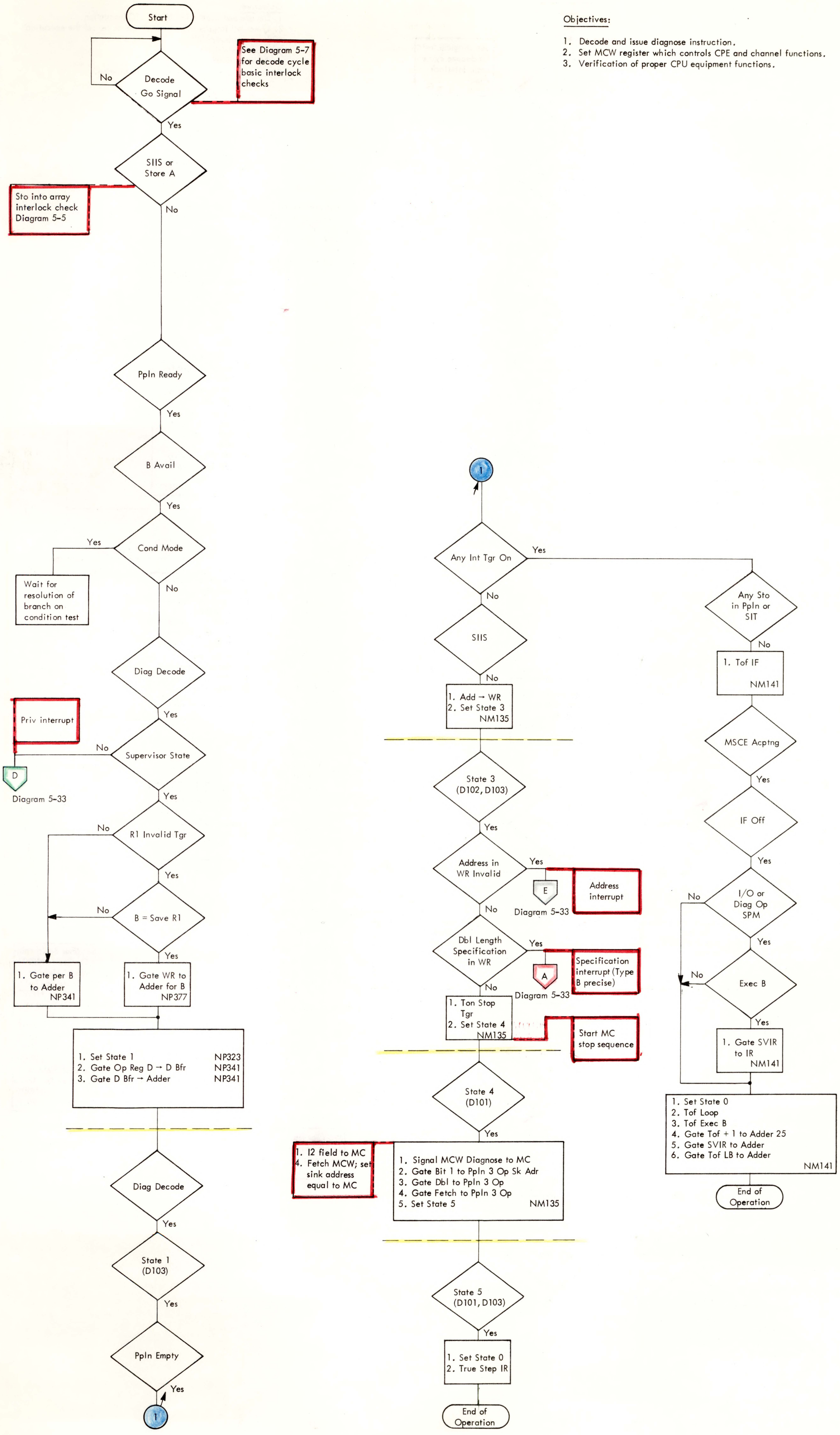
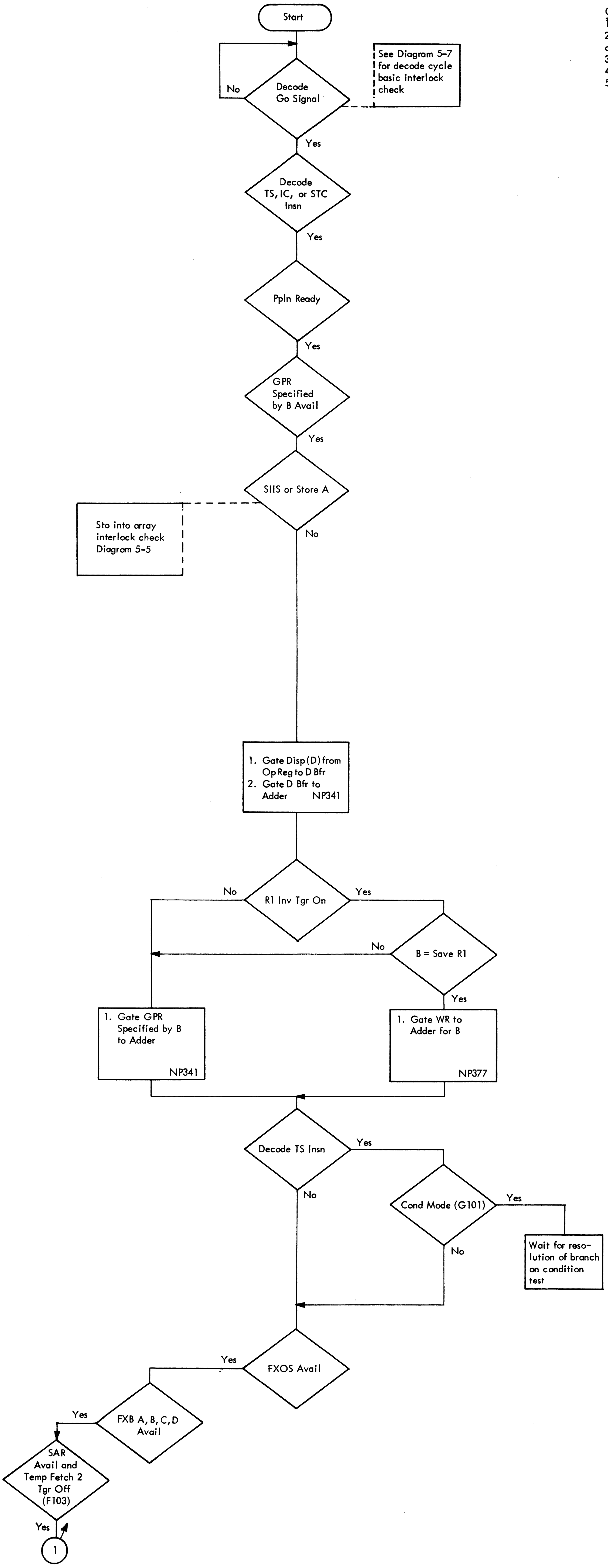


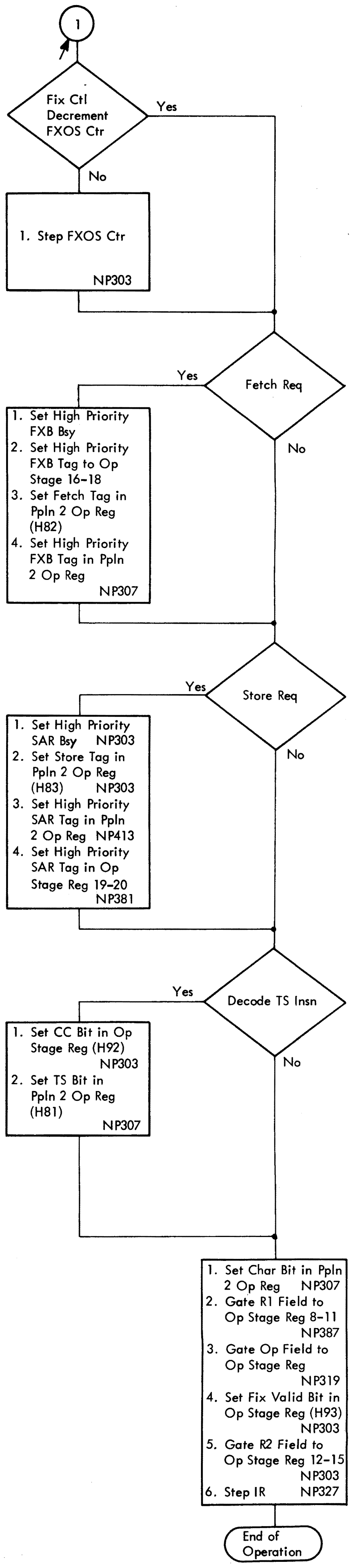
DIAGRAM 5-21. DIAGNOSE SEQUENCE

Objectives:

1. Decode and issue a TS, IC, or STC instruction.
2. Check all interlocks that can detain or cancel the execution of these instructions.
3. Issue instruction to FXA.
4. Allocate buffers as appropriate.
5. Step instruction counter to next instruction.



Start
Decode Go Signal
No
See Diagram 5-7 for decode cycle basic interlock check
Yes
Decode TS, IC, or STC Insn
Yes
Ppln Ready
Yes
GPR Specified by B Avail
Yes
SIIS or Store A
Sto into array interlock check Diagram 5-5
No
1. Gate Disp (D) from Op Reg to D Bfr
2. Gate D Bfr to Adder NP341
R1 Inv Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B NP377
Decode TS Insn
Yes
No
Cond Mode (G101)
Yes
No
Wait for resolution of branch on condition test
FXOS Avail
Yes
FXB A, B, C, D Avail
Yes
SAR Avail and Temp Fetch 2 Tgr Off (F103)
Yes
1

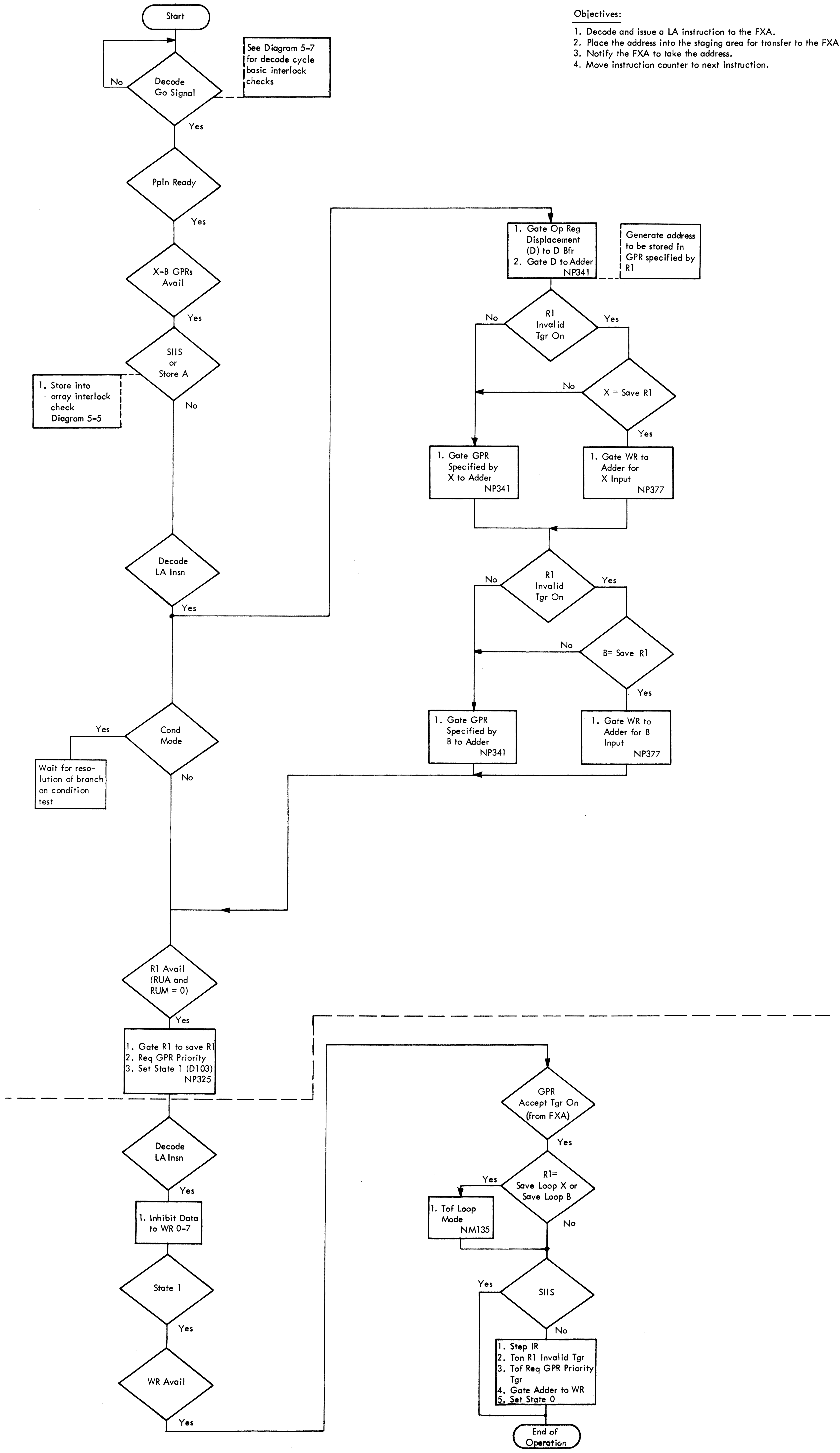


1
Fix Ctl Decrement FXOS Ctr
Yes
No
1. Step FXOS Ctr NP303
Fetch Req
Yes
No
1. Set High Priority FXB Bsy
2. Set High Priority FXB Tag to Op Stage 16-18
3. Set Fetch Tag in Ppln 2 Op Reg (H82)
4. Set High Priority FXB Tag in Ppln 2 Op Reg NP307
Store Req
Yes
No
1. Set High Priority SAR Bsy NP303
2. Set Store Tag in Ppln 2 Op Reg (H83) NP303
3. Set High Priority SAR Tag in Ppln 2 Op Reg NP413
4. Set High Priority SAR Tag in Op Stage Reg 19-20 NP381
Decode TS Insn
Yes
No
1. Set CC Bit in Op Stage Reg (H92) NP303
2. Set TS Bit in Ppln 2 Op Reg (H81) NP307
1. Set Char Bit in Ppln 2 Op Reg NP307
2. Gate R1 Field to Op Stage Reg 8-11 NP387
3. Gate Op Field to Op Stage Reg NP319
4. Set Fix Valid Bit in Op Stage Reg (H93) NP303
5. Gate R2 Field to Op Stage Reg 12-15 NP303
6. Step IR NP327
End of Operation

DIAGRAM 5-22. INSERT CHARACTER, STORE CHARACTER, AND TEST AND SET SEQUENCE

Objectives:

1. Decode and issue a LA instruction to the FXA.
2. Place the address into the staging area for transfer to the FXA.
3. Notify the FXA to take the address.
4. Move instruction counter to next instruction.



Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock checks
Ppln Ready
Yes
X-B GPRs Avail
Yes
SIIS or Store A
1. Store into array interlock check Diagram 5-5
No
Decode LA Insn
Yes
1. Gate Op Reg Displacement (D) to D Bfr
2. Gate D to Adder NP341
Generate address to be stored in GPR specified by R1
R1 Invalid Tgr On
No
Yes
X = Save R1
No
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP377
R1 Invalid Tgr On
No
Yes
B= Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B Input NP377
Cond Mode
Yes
Wait for resolution of branch on condition test
No
R1 Avail (RUA and RUM = 0)
Yes
1. Gate R1 to save R1
2. Req GPR Priority
3. Set State 1 (D103) NP325
Decode LA Insn
Yes
1. Inhibit Data to WR 0-7
State 1
Yes
WR Avail
Yes
GPR Accept Tgr On (from FXA)
Yes
R1= Save Loop X or Save Loop B
Yes
1. Tof Loop Mode NM135
No
SIIS
Yes
No
1. Step IR
2. Ton R1 Invalid Tgr
3. Tof Req GPR Priority Tgr
4. Gate Adder to WR
5. Set State 0
End of Operation

DIAGRAM 5-23. LOAD ADDRESS SEQUENCE

Objectives:

1. Integer part of operand shifted number of bits specified by second operand address.
2. Second operand low-order six bits indicate number of bit positions to be shifted.

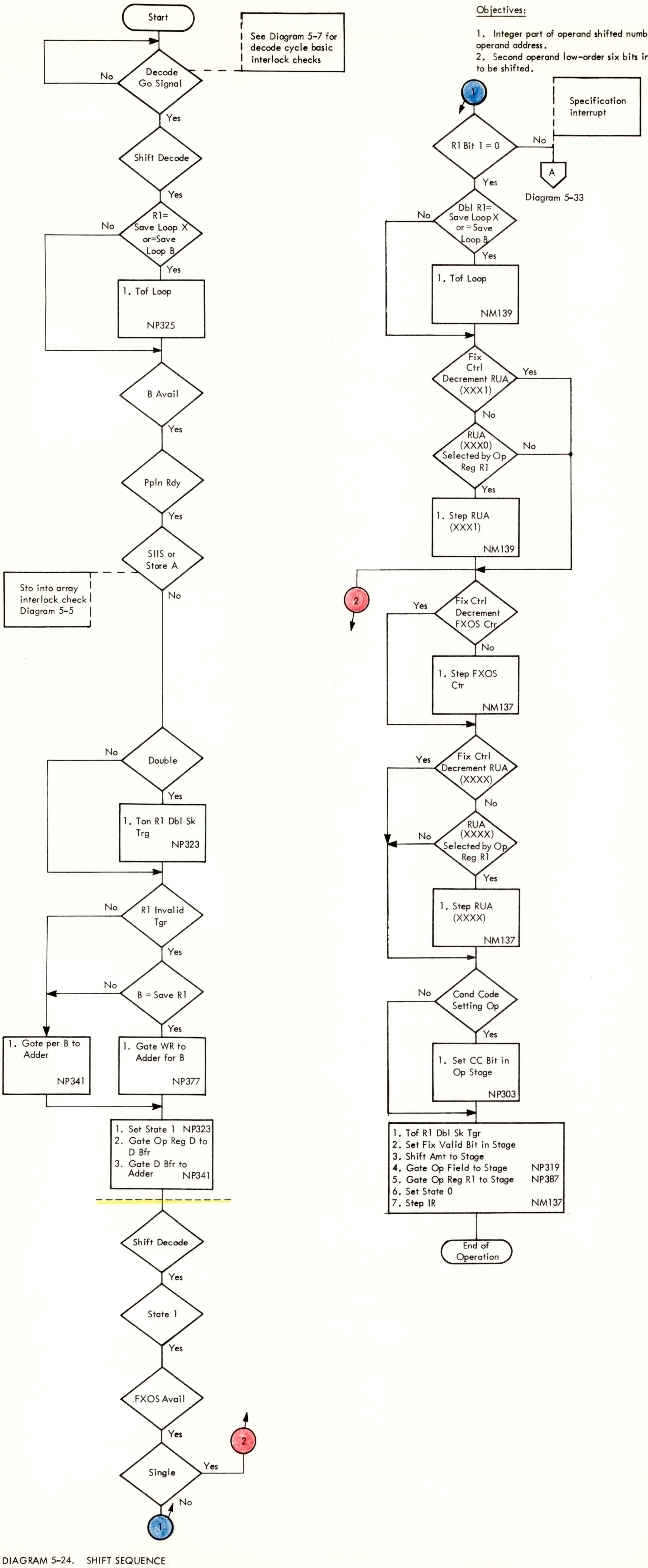


DIAGRAM 5-24. SHIFT SEQUENCE

Objectives:

1. Decode and issue the CVB or CVD instruction to the FXA
2. Place the sink address into the MSCE interface hardware and establish control signals as appropriate for storage.
3. Allocate FXP and FLP buffers as appropriate.
4. Establish the sink interlock on the R1 specified GPR.
5. Set length into the PSW.
6. Move instruction counter to next instruction.

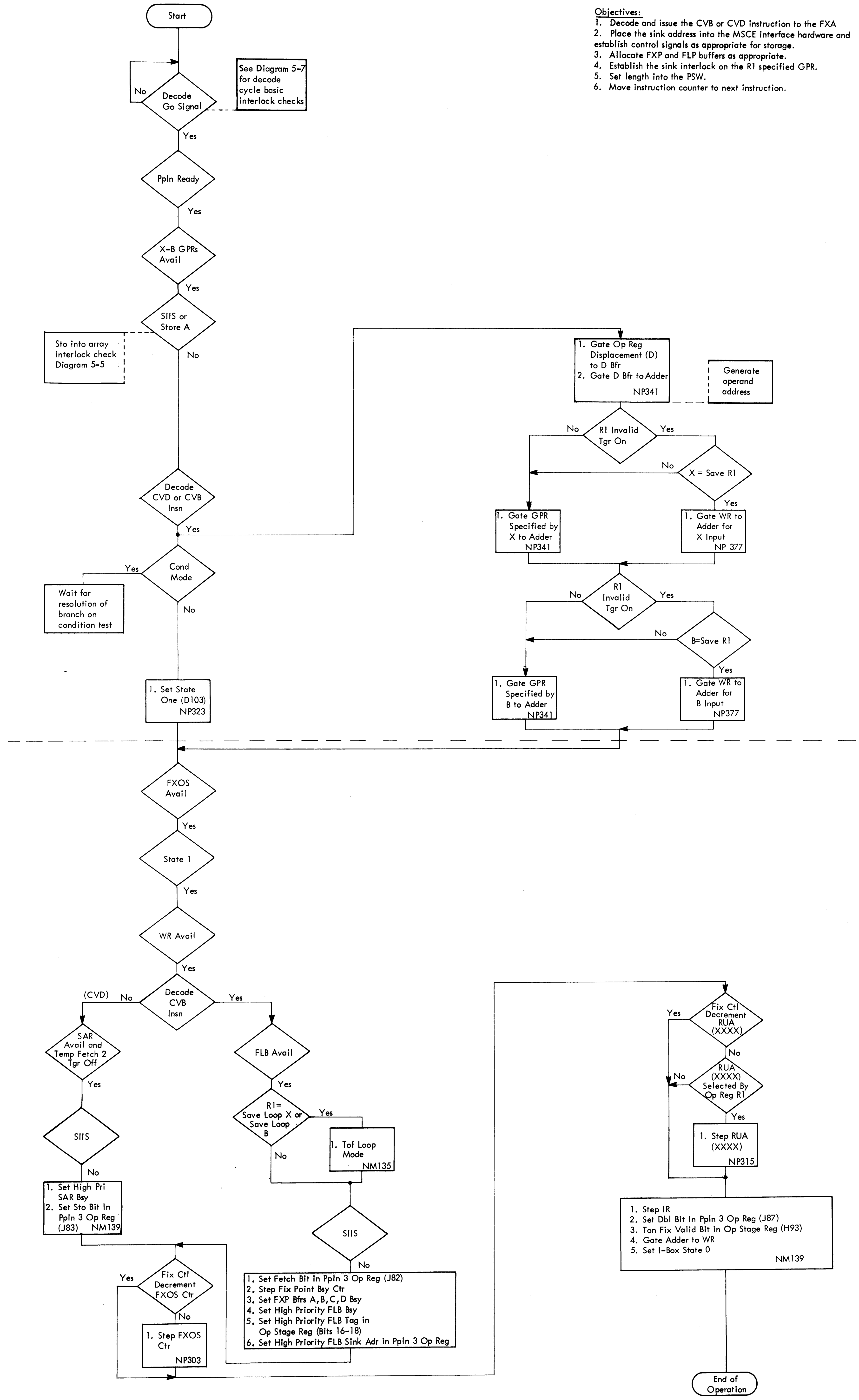

Start
No
Decode Go Signal
See Diagram 5-7 for decode cycle basic interlock checks
Yes
Ppln Ready
Yes
X-B GPRs Avail
Yes
SIIS or Store A
Sto into array interlock check Diagram 5-5
No
Decode CVD or CVB Insn
Yes
Yes
Cond Mode
Wait for resolution of branch on condition test
No
1. Set State One (D103) NP323
1. Gate Op Reg Displacement (D) to D Bfr
2. Gate D Bfr to Adder NP341
Generate operand address
No
R1 Invalid Tgr On
Yes
No
X = Save R1
Yes
1. Gate GPR Specified by X to Adder NP341
1. Gate WR to Adder for X Input NP 377
No
R1 Invalid Tgr On
Yes
No
B=Save R1
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B Input NP377
FXOS Avail
Yes
State 1
Yes
WR Avail
Yes
(CVD) No
Decode CVB Insn
Yes
SAR Avail and Temp Fetch 2 Tgr Off
Yes
SIIS
No
1. Set High Pri SAR Bsy
2. Set Sto Bit In Ppln 3 Op Reg (J83) NM139
Yes
Fix Ctl Decrement FXOS Ctr
No
1. Step FXOS Ctr NP303
FLB Avail
Yes
R1= Save Loop X or Save Loop B
Yes
1. Tof Loop Mode NM135
No
SIIS
No
1. Set Fetch Bit in Ppln 3 Op Reg (J82)
2. Step Fix Point Bsy Ctr
3. Set FXP Bfrs A,B,C,D Bsy
4. Set High Priority FLB Bsy
5. Set High Priority FLB Tag in Op Stage Reg (Bits 16-18)
6. Set High Priority FLB Sink Adr in Ppln 3 Op Reg
Yes
Fix Ctl Decrement RUA (XXXX)
No
No
RUA (XXXX) Selected By Op Reg R1
Yes
1. Step RUA (XXXX) NP315
1. Step IR
2. Set Dbl Bit In Ppln 3 Op Reg (J87)
3. Ton Fix Valid Bit in Op Stage Reg (H93)
4. Gate Adder to WR
5. Set I-Box State 0
NM139
End of Operation

DIAGRAM 5-25. CONVERT SEQUENCE

Objectives:
1. Timer update treated as fixed-point operation.
2. When I-box receives timer signal, it effectively initiates a fetch and sets up store for timer storage location.
3. FXA completes timer update and store operation.

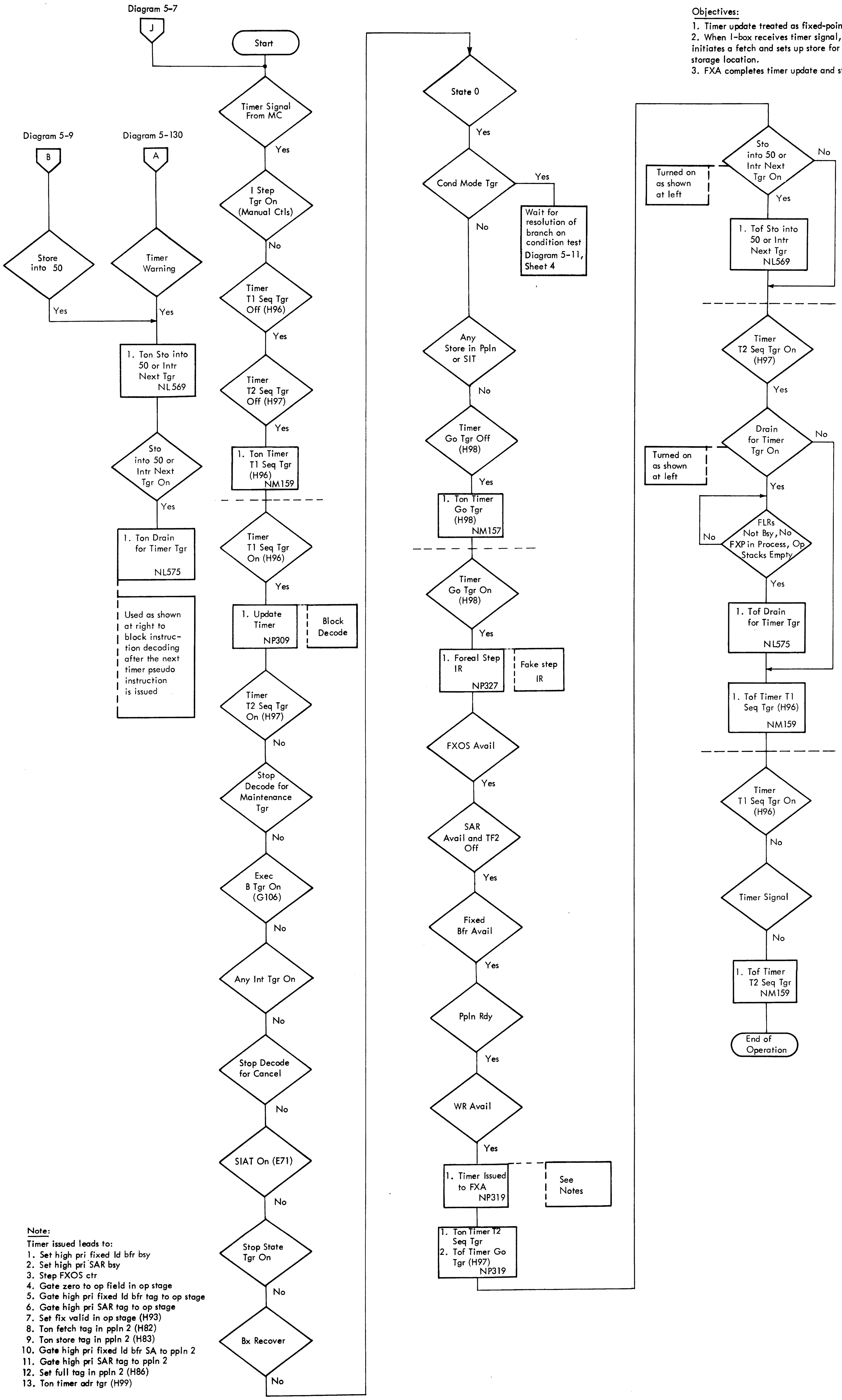


Diagram 5-7
J
Start
Timer Signal From MC
Yes
I Step Tgr On (Manual Ctls)
No
Timer T1 Seq Tgr Off (H96)
Yes
Timer T2 Seq Tgr Off (H97)
Yes
1. Ton Timer T1 Seq Tgr (H96) NM159
Timer T1 Seq Tgr On (H96)
Yes
1. Update Timer NP309
Block Decode
Timer T2 Seq Tgr On (H97)
No
Stop Decode for Maintenance Tgr
No
Exec B Tgr On (G106)
No
Any Int Tgr On
No
Stop Decode for Cancel
No
SIAT On (E71)
No
Stop State Tgr On
No
Bx Recover
No
Diagram 5-9
B
Store into 50
Yes
Diagram 5-130
A
Timer Warning
Yes
1. Ton Sto into 50 or Intr Next Tgr NL569
Sto into 50 or Intr Next Tgr On
Yes
1. Ton Drain for Timer Tgr NL575
Used as shown at right to block instruction decoding after the next timer pseudo instruction is issued
State 0
Yes
Cond Mode Tgr
Yes
Wait for resolution of branch on condition test Diagram 5-11, Sheet 4
No
Any Store in Ppln or SIT
No
Timer Go Tgr Off (H98)
Yes
1. Ton Timer Go Tgr (H98) NM157
Timer Go Tgr On (H98)
Yes
1. Foreal Step IR NP327
Fake step IR
FXOS Avail
Yes
SAR Avail and TF2 Off
Yes
Fixed Bfr Avail
Yes
Ppln Rdy
Yes
WR Avail
Yes
1. Timer Issued to FXA NP319
See Notes
1. Ton Timer T2 Seq Tgr
2. Tof Timer Go Tgr (H97) NP319
Turned on as shown at left
Sto into 50 or Intr Next Tgr On
No
Yes
1. Tof Sto into 50 or Intr Next Tgr NL569
Timer T2 Seq Tgr On (H97)
Yes
Turned on as shown at left
Drain for Timer Tgr On
No
Yes
No
FLRs Not Bsy, No FXP in Process, Op Stacks Empty
Yes
1. Tof Drain for Timer Tgr NL575
1. Tof Timer T1 Seq Tgr (H96) NM159
Timer T1 Seq Tgr On (H96)
No
Timer Signal
No
1. Tof Timer T2 Seq Tgr NM159
End of Operation

Note:
Timer issued leads to:
1. Set high pri fixed ld bfr bsy
2. Set high pri SAR bsy
3. Step FXOS ctr
4. Gate zero to op field in op stage
5. Gate high pri fixed ld bfr tag to op stage
6. Gate high pri SAR tag to op stage
7. Set fix valid in op stage (H93)
8. Ton fetch tag in ppln 2 (H82)
9. Ton store tag in ppln 2 (H83)
10. Gate high pri fixed ld bfr SA to ppln 2
11. Gate high pri SAR tag to ppln 2
12. Set full tag in ppln 2 (H86)
13. Ton timer adr tgr (H99)

DIAGRAM 5-26. TIMER UPDATE SEQUENCE

Objectives:

1. Decode and issue an I/O instruction.
2. Check all interlocks that can detain or cancel the execution of this instruction.
3. Gate bits 16-33 of B1 and D1 field (contained in WR) and a line specifying one of the I/O insn, to all channels.
4. When selected channel responds, set condition code and terminate instruction.

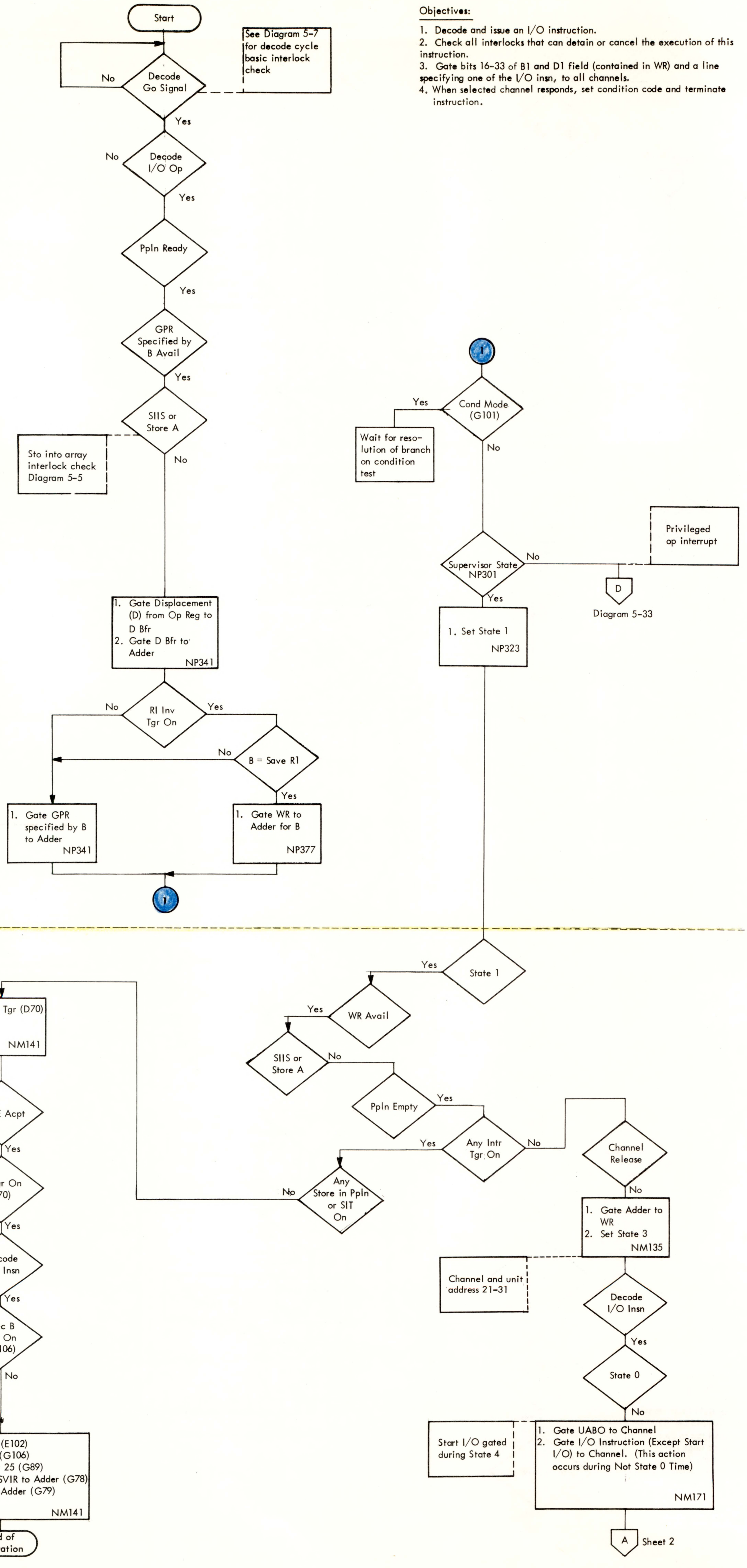


DIAGRAM 5-27. I/O SEQUENCE (SHEET 1 OF 2)

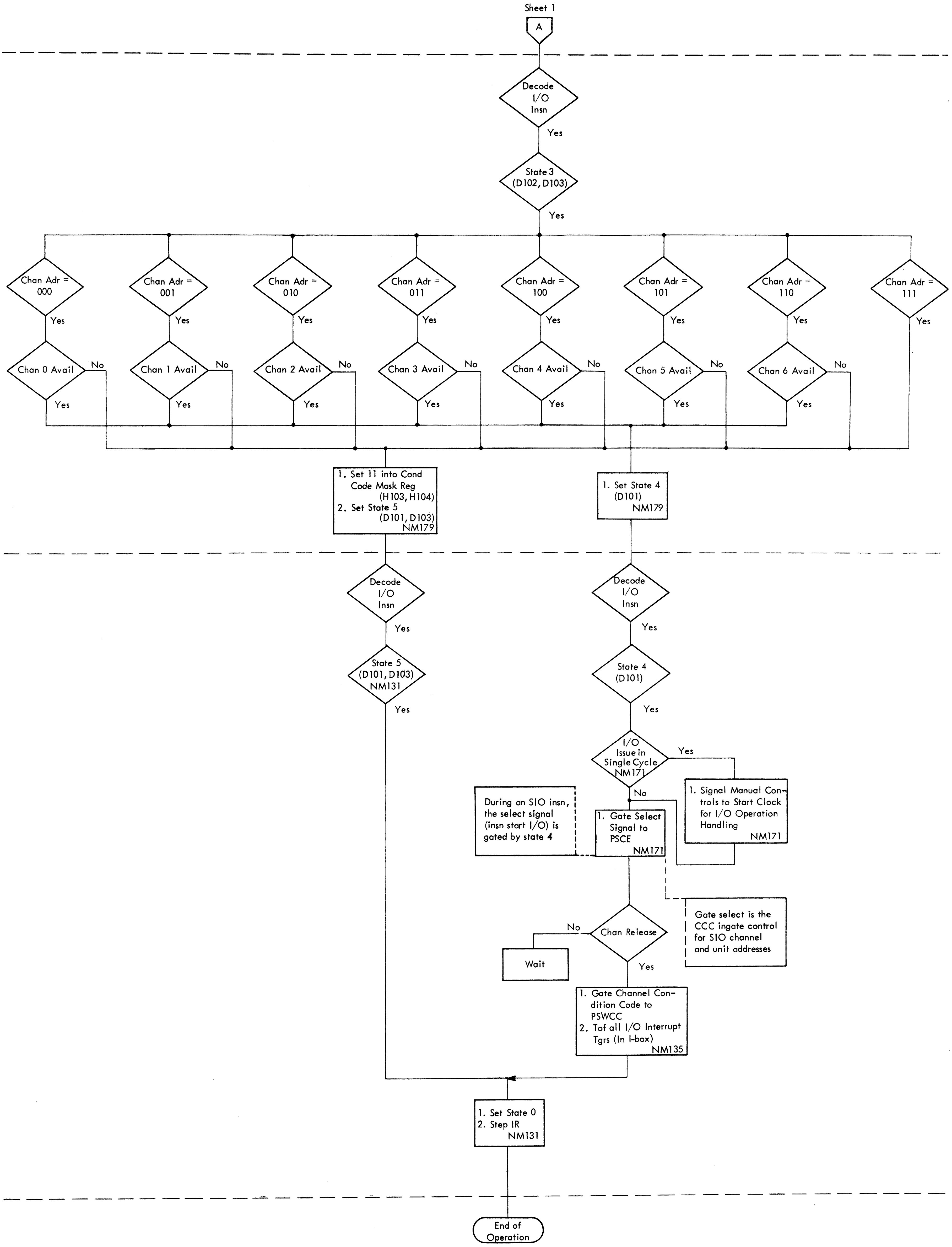
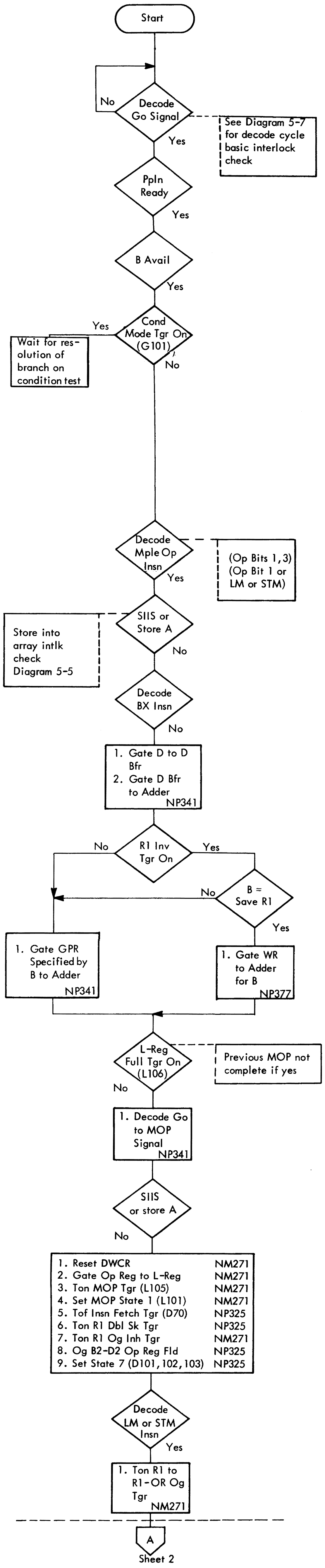

Sheet 1
A
Decode I/O Insn
Yes
State 3 (D102, D103)
Yes
Chan Adr = 000
Chan Adr = 001
Chan Adr = 010
Chan Adr = 011
Chan Adr = 100
Chan Adr = 101
Chan Adr = 110
Chan Adr = 111
Yes
Chan 0 Avail
Chan 1 Avail
Chan 2 Avail
Chan 3 Avail
Chan 4 Avail
Chan 5 Avail
Chan 6 Avail
No
Yes
1. Set 11 into Cond Code Mask Reg (H103, H104)
2. Set State 5 (D101, D103) NM179
1. Set State 4 (D101) NM179
Decode I/O Insn
Yes
State 5 (D101, D103) NM131
Yes
Decode I/O Insn
Yes
State 4 (D101)
Yes
I/O Issue in Single Cycle NM171
Yes
1. Signal Manual Controls to Start Clock for I/O Operation Handling NM171
No
During an SIO insn, the select signal (insn start I/O) is gated by state 4
1. Gate Select Signal to PSCE NM171
Gate select is the CCC ingate control for SIO channel and unit addresses
Chan Release
No
Wait
Yes
1. Gate Channel Condition Code to PSWCC
2. Tof all I/O Interrupt Tgrs (In I-box) NM135
1. Set State 0
2. Step IR NM131
End of Operation

DIAGRAM 5-27. I/O SEQUENCE (SHEET 2 OF 2)

Objectives:

1. Decode and issue LM or STM instruction.
2. Check all interlocks that might detain or cancel the instruction.
3. LM - Fetch a doubleword; inform FXA of the FLB assigned for this fetch, the position of the data within the doubleword; and the GPR(s) that the data should fill.
4. STM - Set up a store for a doubleword; inform FXA of the SAR assigned for this store, the position of the data within the doubleword, and the GPR(s) to be used as source(s) for the data.



Start
No
Decode Go Signal
Yes
See Diagram 5-7 for decode cycle basic interlock check
Ppln Ready
Yes
B Avail
Yes
Cond Mode Tgr On (G101)
Yes
Wait for resolution of branch on condition test
No
Decode Mple Op Insn
(Op Bits 1,3) (Op Bit 1 or LM or STM)
Yes
SIIS or Store A
Store into array intlk check Diagram 5-5
No
Decode BX Insn
No
1. Gate D to D Bfr
2. Gate D Bfr to Adder
NP341
R1 Inv Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder
NP341
1. Gate WR to Adder for B
NP377
L-Reg Full Tgr On (L106)
Previous MOP not complete if yes
No
1. Decode Go to MOP Signal
NP341
SIIS or store A
No
1. Reset DWCR NM271
2. Gate Op Reg to L-Reg NM271
3. Ton MOP Tgr (L105) NM271
4. Set MOP State 1 (L101) NM271
5. Tof Insn Fetch Tgr (D70) NP325
6. Ton R1 Dbl Sk Tgr NP325
7. Ton R1 Og Inh Tgr NM271
8. Og B2-D2 Op Reg Fld NP325
9. Set State 7 (D101,102,103) NP325
Decode LM or STM Insn
Yes
1. Ton R1 to R1-OR Og Tgr
NM271
A
Sheet 2

DIAGRAM 5-28. LM, STM SEQUENCE (SHEET 1 OF 3)

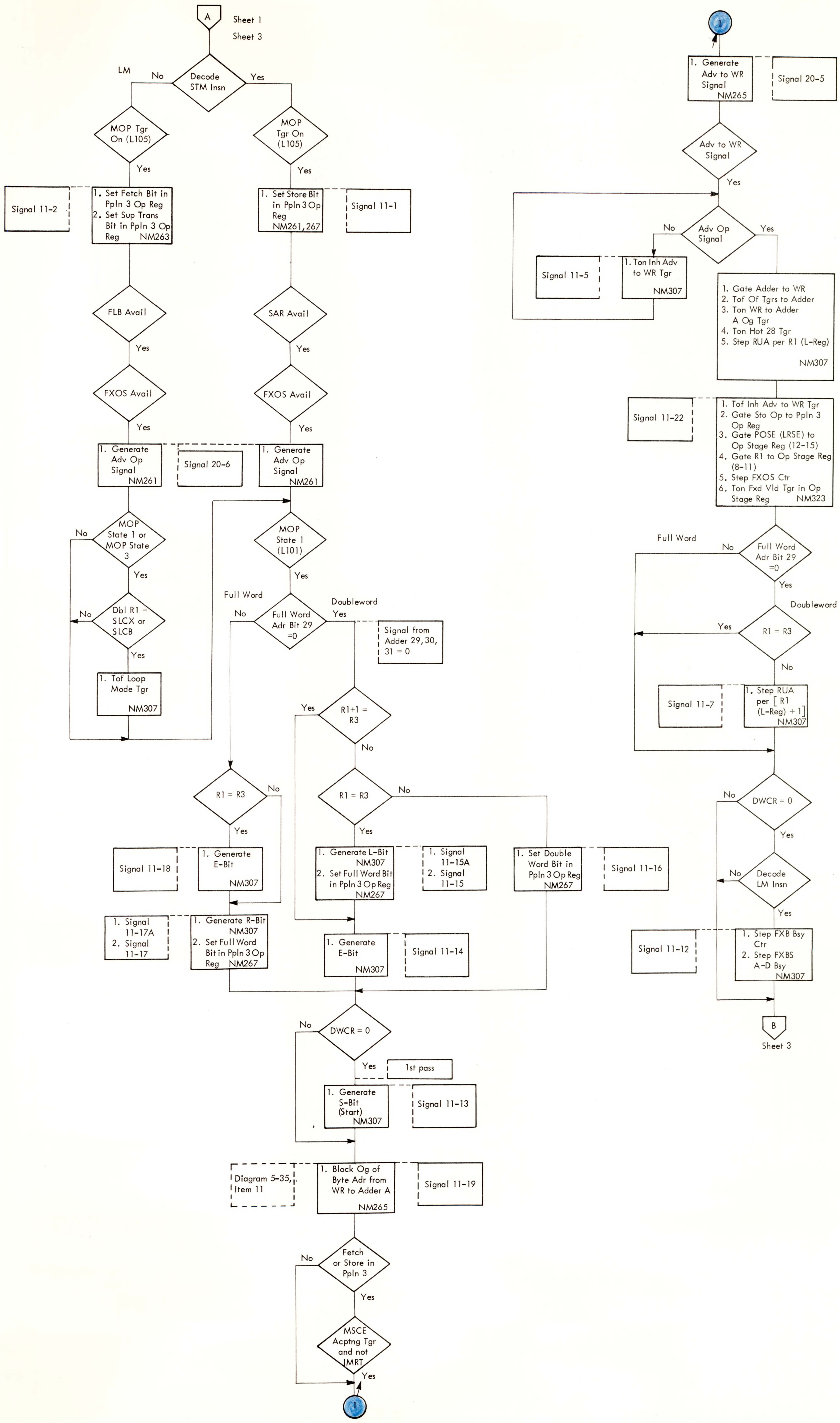


DIAGRAM 5-28. LM, STM SEQUENCE (SHEET 2 OF 3)

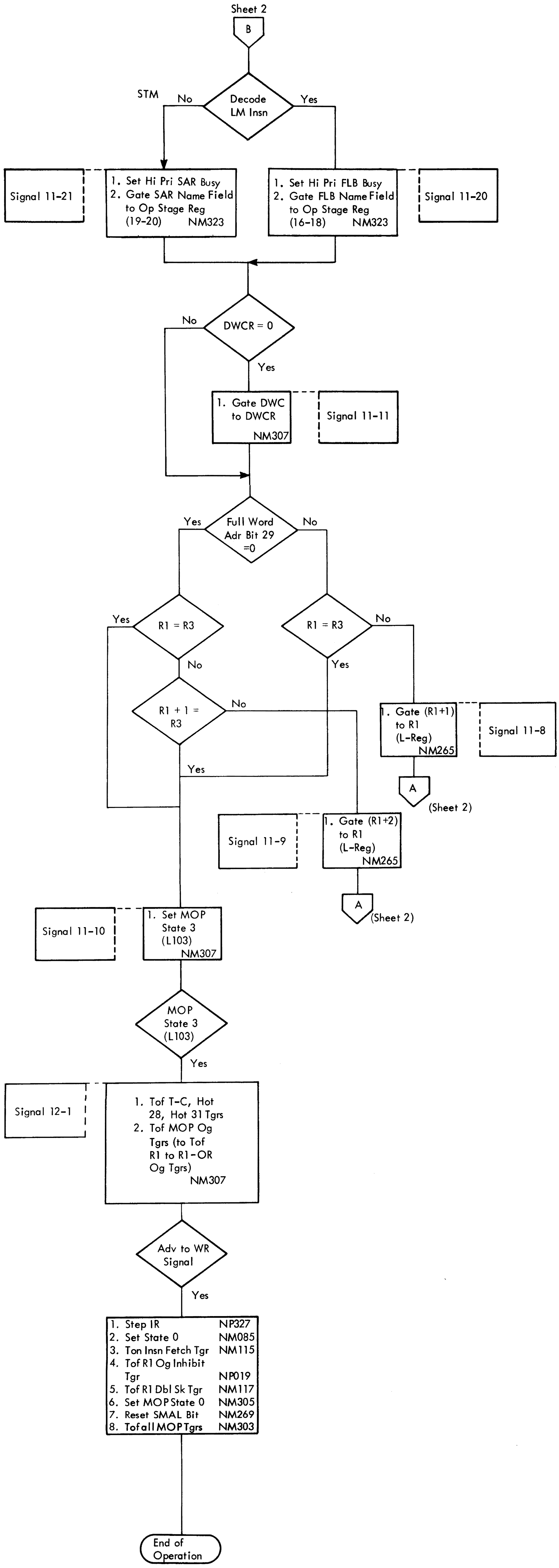
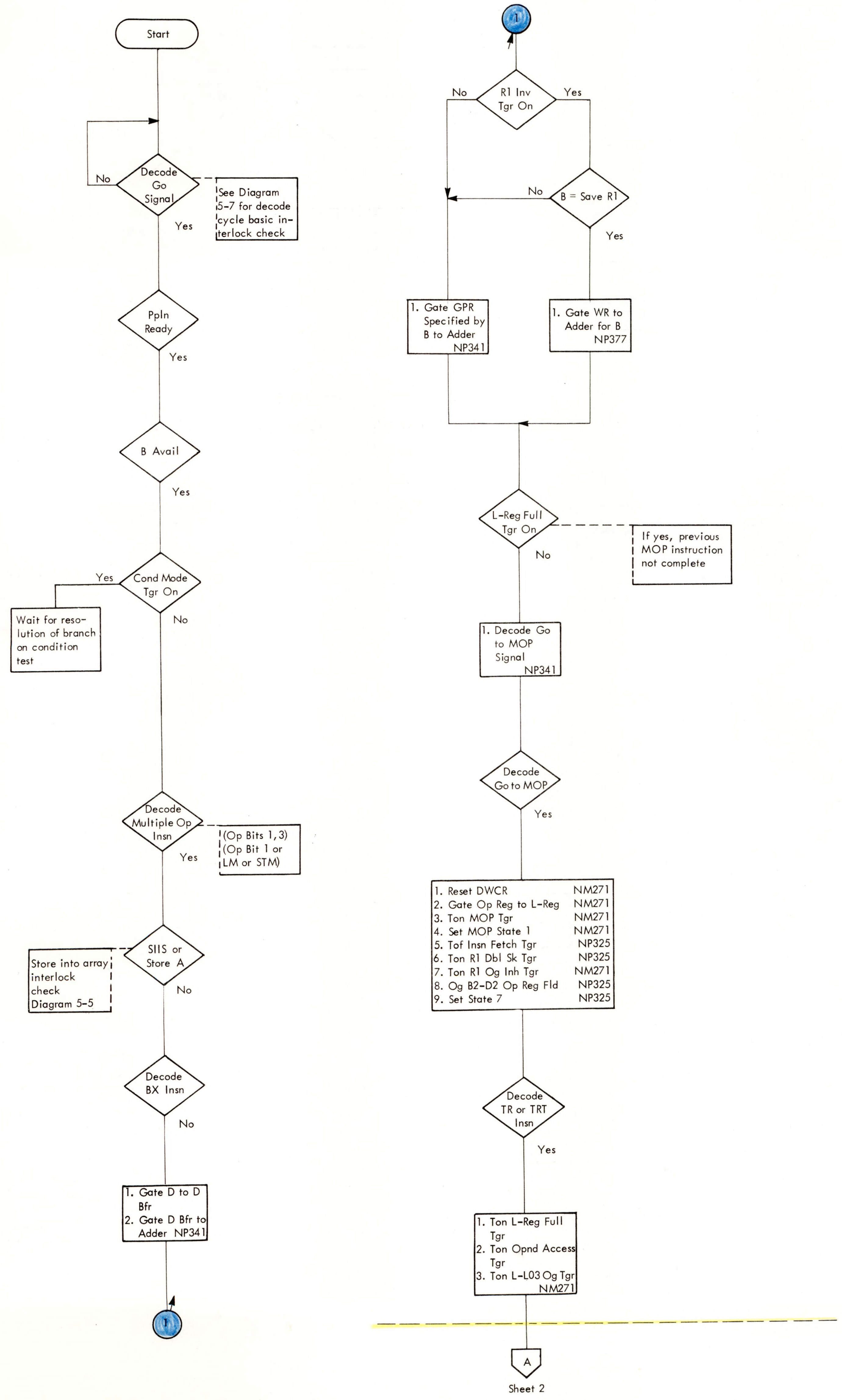

Sheet 2
B
STM
No
Decode LM Insn
Yes
Signal 11-21
1. Set Hi Pri SAR Busy
2. Gate SAR Name Field to Op Stage Reg (19-20) NM323
1. Set Hi Pri FLB Busy
2. Gate FLB Name Field to Op Stage Reg (16-18) NM323
Signal 11-20
No
DWCR = 0
Yes
1. Gate DWC to DWCR NM307
Signal 11-11
Yes
Full Word Adr Bit 29 =0
No
Yes
R1 = R3
No
R1 = R3
No
Yes
R1 + 1 = R3
No
1. Gate (R1+1) to R1 (L-Reg) NM265
Signal 11-8
Yes
A
(Sheet 2)
Signal 11-9
1. Gate (R1+2) to R1 (L-Reg) NM265
A
(Sheet 2)
Signal 11-10
1. Set MOP State 3 (L103) NM307
MOP State 3 (L103)
Yes
Signal 12-1
1. Tof T-C, Hot 28, Hot 31 Tgrs
2. Tof MOP Og Tgrs (to Tof R1 to R1-OR Og Tgrs) NM307
Adv to WR Signal
Yes
1. Step IR NP327
2. Set State 0 NM085
3. Ton Insn Fetch Tgr NM115
4. Tof R1 Og Inhibit Tgr NP019
5. Tof R1 Dbl Sk Tgr NM117
6. Set MOP State 0 NM305
7. Reset SMAL Bit NM269
8. Tof all MOP Tgrs NM303
End of Operation

DIAGRAM 5-28. LM, STM SEQUENCE (SHEET 3 OF 3)

Objectives:

1. Decode and issue TR or TRT instruction.
2. Check all interlocks that might detain or cancel the instruction.
3. TR-Fetch and set up store for argument double word(s), and inform FXA of the assigned FLB(s) and SAR(s); suppress MSCE multi-accessing; fetch table word(s), and inform FXA of the assigned FLB(s).
4. TRT-Fetch argument double word(s), and inform FXA of the assigned FLB(s); fetch table word(s) and inform FXA of the assigned FLB(s); generate required argument byte address and save it in GPR 1.



Start
Decode Go Signal
No
Yes
See Diagram 5-7 for decode cycle basic interlock check
Ppln Ready
Yes
B Avail
Yes
Cond Mode Tgr On
Yes
No
Wait for resolution of branch on condition test
Decode Multiple Op Insn
Yes
(Op Bits 1,3) (Op Bit 1 or LM or STM)
SIIS or Store A
No
Store into array interlock check Diagram 5-5
Decode BX Insn
No
1. Gate D to D Bfr
2. Gate D Bfr to Adder NP341
1
1
R1 Inv Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B NP377
L-Reg Full Tgr On
No
If yes, previous MOP instruction not complete
1. Decode Go to MOP Signal NP341
Decode Go to MOP
Yes
1. Reset DWCR NM271
2. Gate Op Reg to L-Reg NM271
3. Ton MOP Tgr NM271
4. Set MOP State 1 NM271
5. Tof Insn Fetch Tgr NP325
6. Ton R1 Dbl Sk Tgr NP325
7. Ton R1 Og Inh Tgr NM271
8. Og B2-D2 Op Reg Fld NP325
9. Set State 7 NP325
Decode TR or TRT Insn
Yes
1. Ton L-Reg Full Tgr
2. Ton Opnd Access Tgr
3. Ton L-L03 Og Tgr NM271
A
Sheet 2

DIAGRAM 5-29. TR, TRT SEQUENCE (SHEET 1 OF 5)

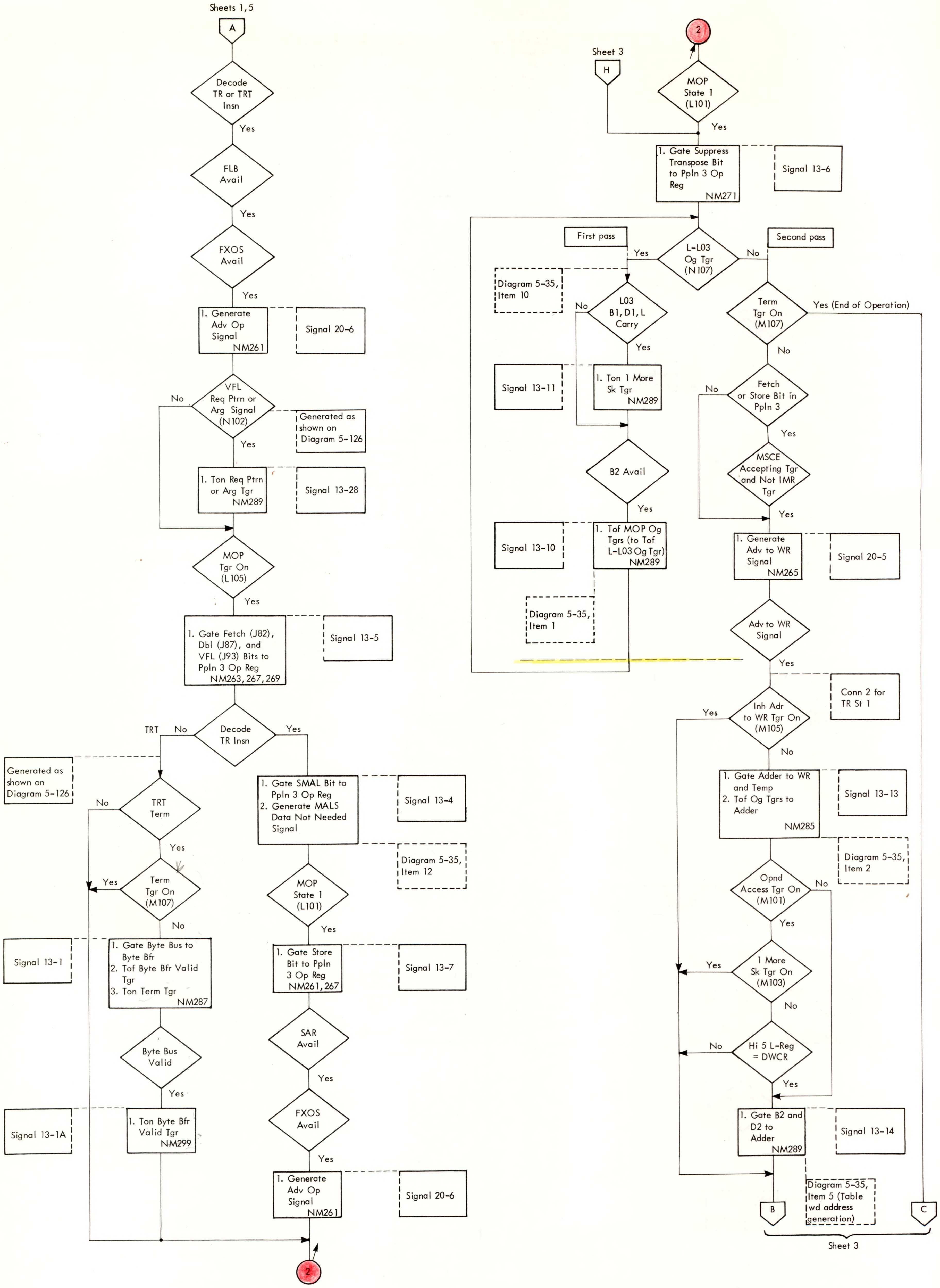

Sheets 1,5
A
Decode TR or TRT Insn
Yes
FLB Avail
Yes
FXOS Avail
Yes
1. Generate Adv Op Signal NM261
Signal 20-6
VFL Req Ptrn or Arg Signal (N102)
No
Generated as shown on Diagram 5-126
Yes
1. Ton Req Ptrn or Arg Tgr NM289
Signal 13-28
MOP Tgr On (L105)
Yes
1. Gate Fetch (J82), Dbl (J87), and VFL (J93) Bits to Ppln 3 Op Reg NM263, 267, 269
Signal 13-5
TRT
No
Decode TR Insn
Yes
Generated as shown on Diagram 5-126
No
TRT Term
Yes
Yes
Term Tgr On (M107)
No
1. Gate Byte Bus to Byte Bfr 2. Tof Byte Bfr Valid Tgr 3. Ton Term Tgr NM287
Signal 13-1
Byte Bus Valid
Yes
1. Ton Byte Bfr Valid Tgr NM299
Signal 13-1A
1. Gate SMAL Bit to Ppln 3 Op Reg 2. Generate MALS Data Not Needed Signal
Signal 13-4
Diagram 5-35, Item 12
MOP State 1 (L101)
Yes
1. Gate Store Bit to Ppln 3 Op Reg NM261, 267
Signal 13-7
SAR Avail
Yes
FXOS Avail
Yes
1. Generate Adv Op Signal NM261
Signal 20-6
2
2
Sheet 3
H
MOP State 1 (L101)
Yes
1. Gate Suppress Transpose Bit to Ppln 3 Op Reg NM271
Signal 13-6
First pass
Yes
L-L03 Og Tgr (N107)
No
Second pass
Diagram 5-35, Item 10
No
L03 B1, D1, L Carry
Yes
Signal 13-11
1. Ton 1 More Sk Tgr NM289
B2 Avail
Yes
Signal 13-10
1. Tof MOP Og Tgrs (to Tof L-L03 Og Tgr) NM289
Diagram 5-35, Item 1
Term Tgr On (M107)
Yes (End of Operation)
No
No
Fetch or Store Bit in Ppln 3
Yes
MSCE Accepting Tgr and Not IMR Tgr
Yes
1. Generate Adv to WR Signal NM265
Signal 20-5
Adv to WR Signal
Yes
Conn 2 for TR St 1
Yes
Inh Adr to WR Tgr On (M105)
No
1. Gate Adder to WR and Temp 2. Tof Og Tgrs to Adder NM285
Signal 13-13
Diagram 5-35, Item 2
Opnd Access Tgr On (M101)
No
Yes
Yes
1 More Sk Tgr On (M103)
No
No
Hi 5 L-Reg = DWCR
Yes
1. Gate B2 and D2 to Adder NM289
Signal 13-14
Diagram 5-35, Item 5 (Table wd address generation)
B
C
Sheet 3

DIAGRAM 5-29. TR, TRT SEQUENCE (SHEET 2 OF 5)

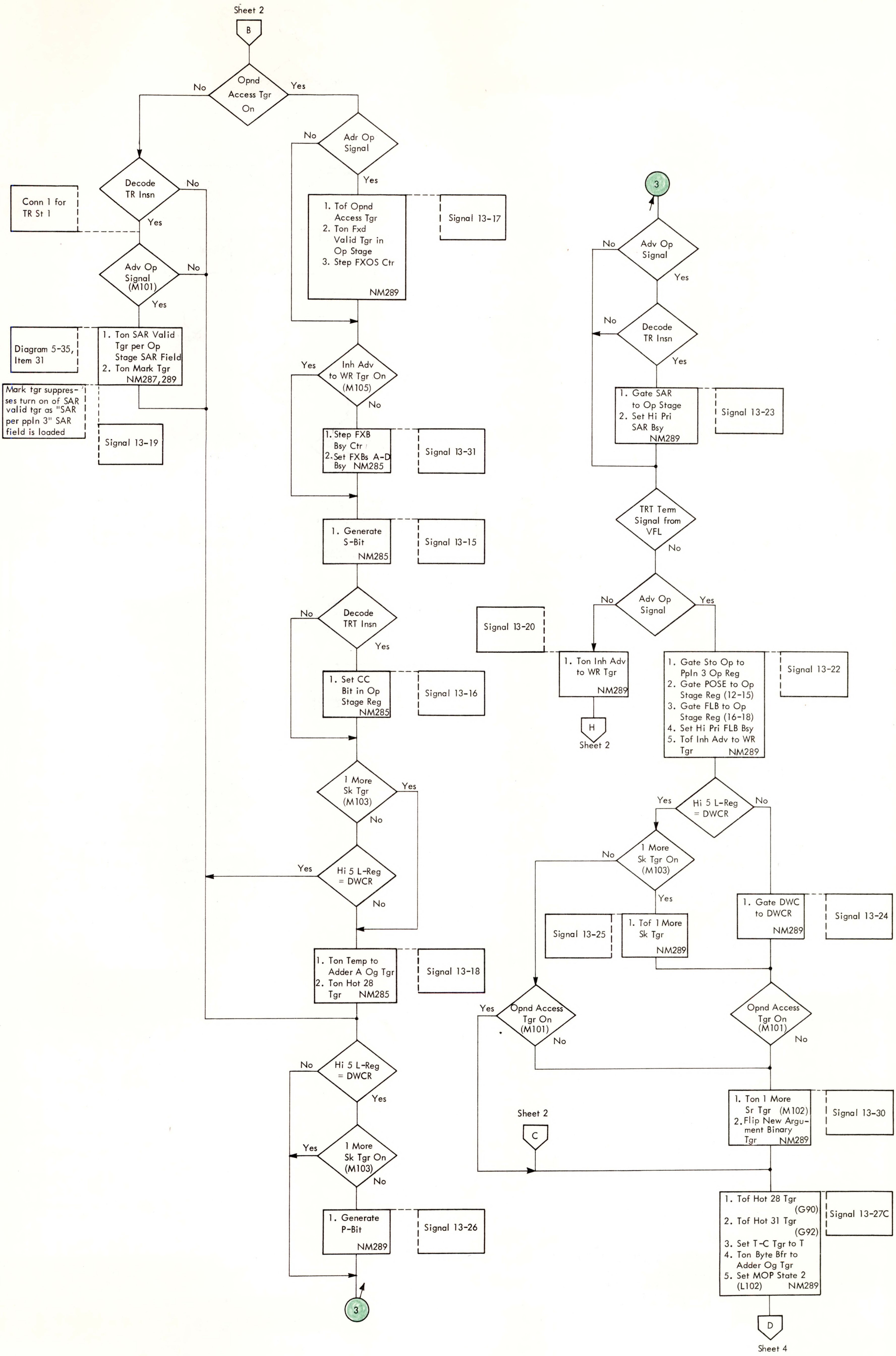


DIAGRAM 5-29. TR, TRT SEQUENCE (SHEET 3 OF 5)

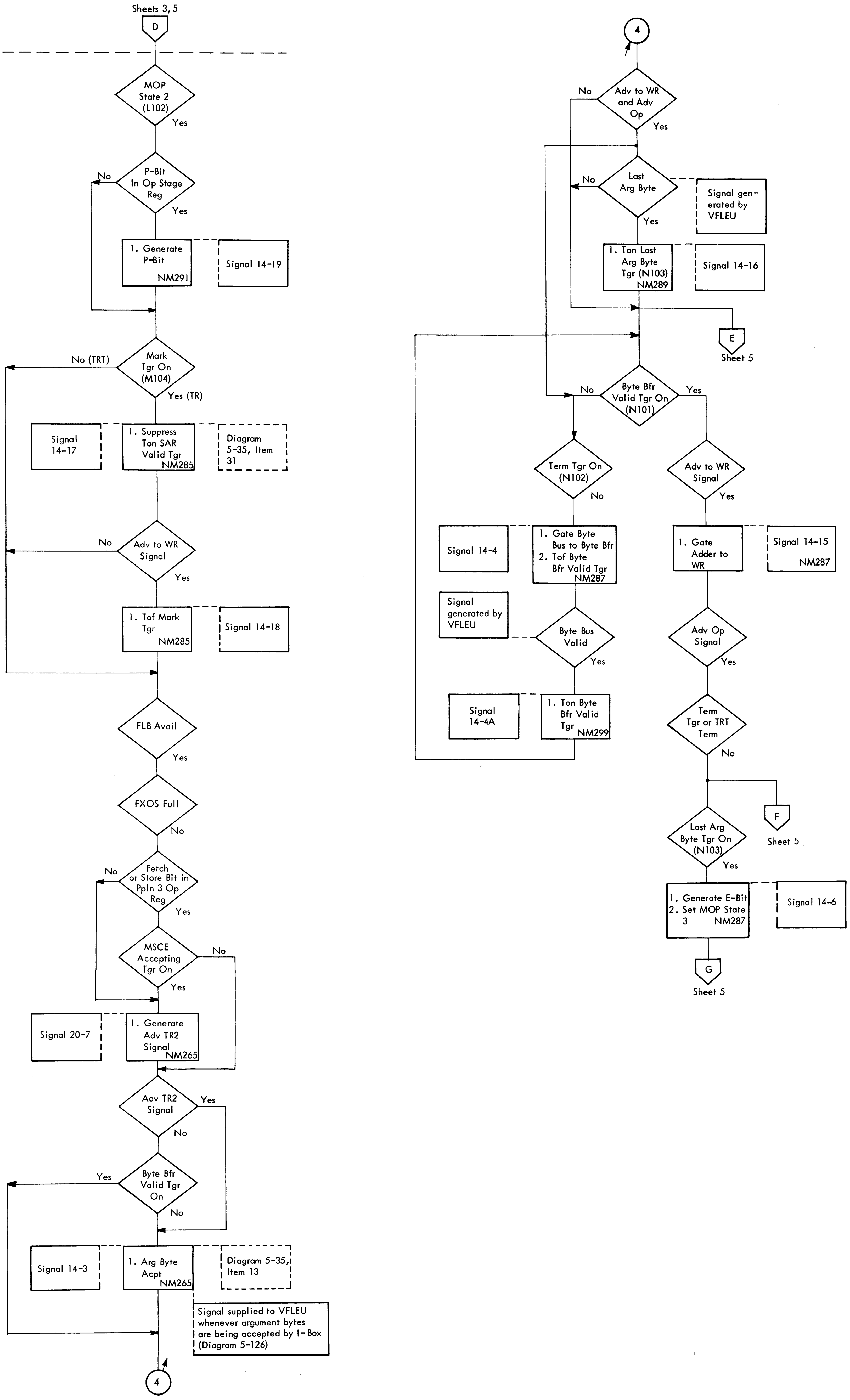


DIAGRAM 5-29. TR, TRT SEQUENCE (SHEET 4 OF 5)

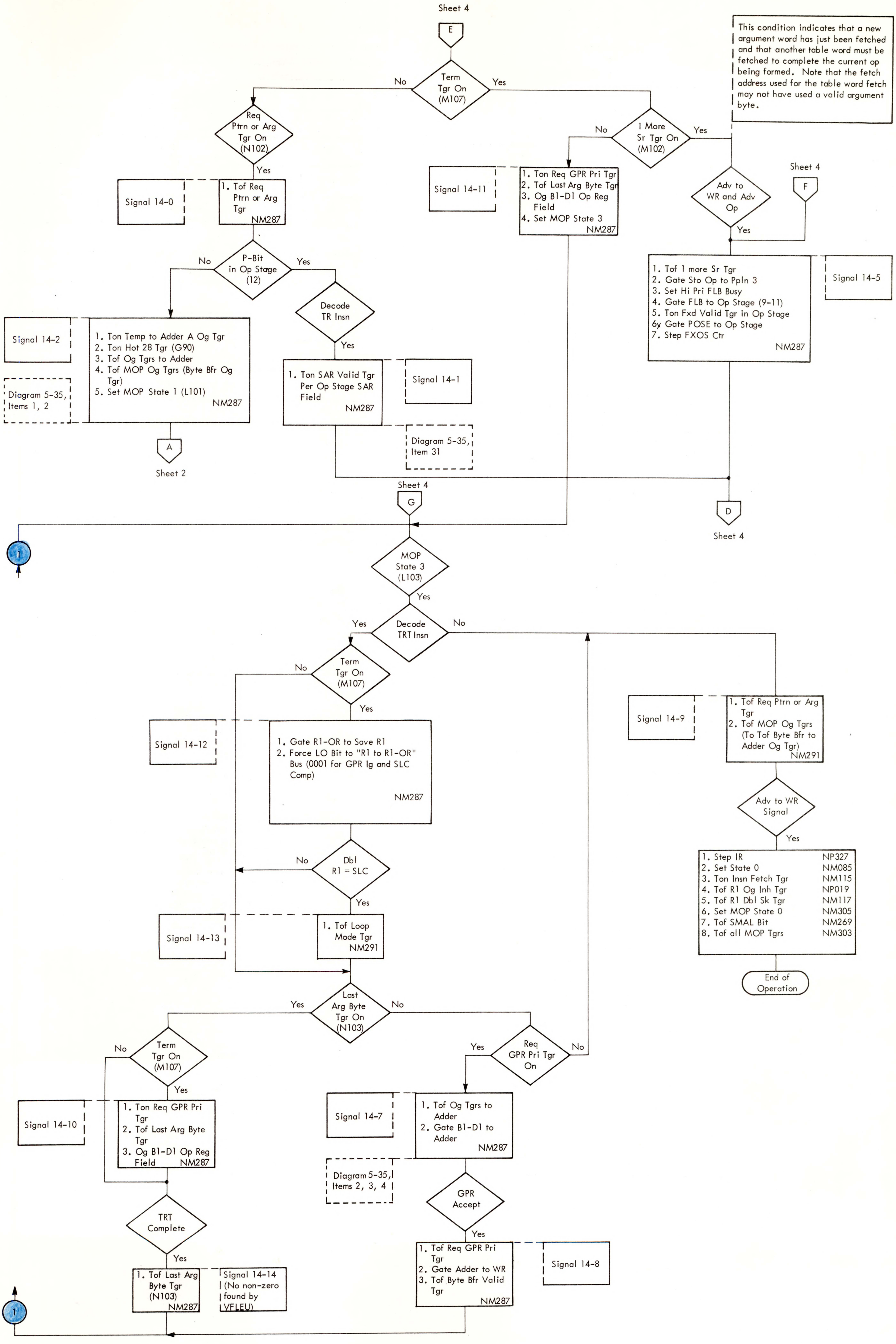


DIAGRAM 5-29. TR, TRT SEQUENCE (SHEET 5 OF 5)

Objective:

Decode and issue the following instructions:
NC, OC, XC, MVZ, MVN, and CLC Instructions (NOXCM Class)

NC	D4	L	B1	D1	B2	D2
	0 7	8 15	16 19	20 31	32 35	36 47
OC	D6	L	B1	D1	B2	D2
XC	D7	L	B1	D1	B2	D2
MVZ	D3	L	B1	D1	B2	D2
MVN	D1	L	B1	D1	B2	D2
CLC	D5	L	B1	D1	B2	D2

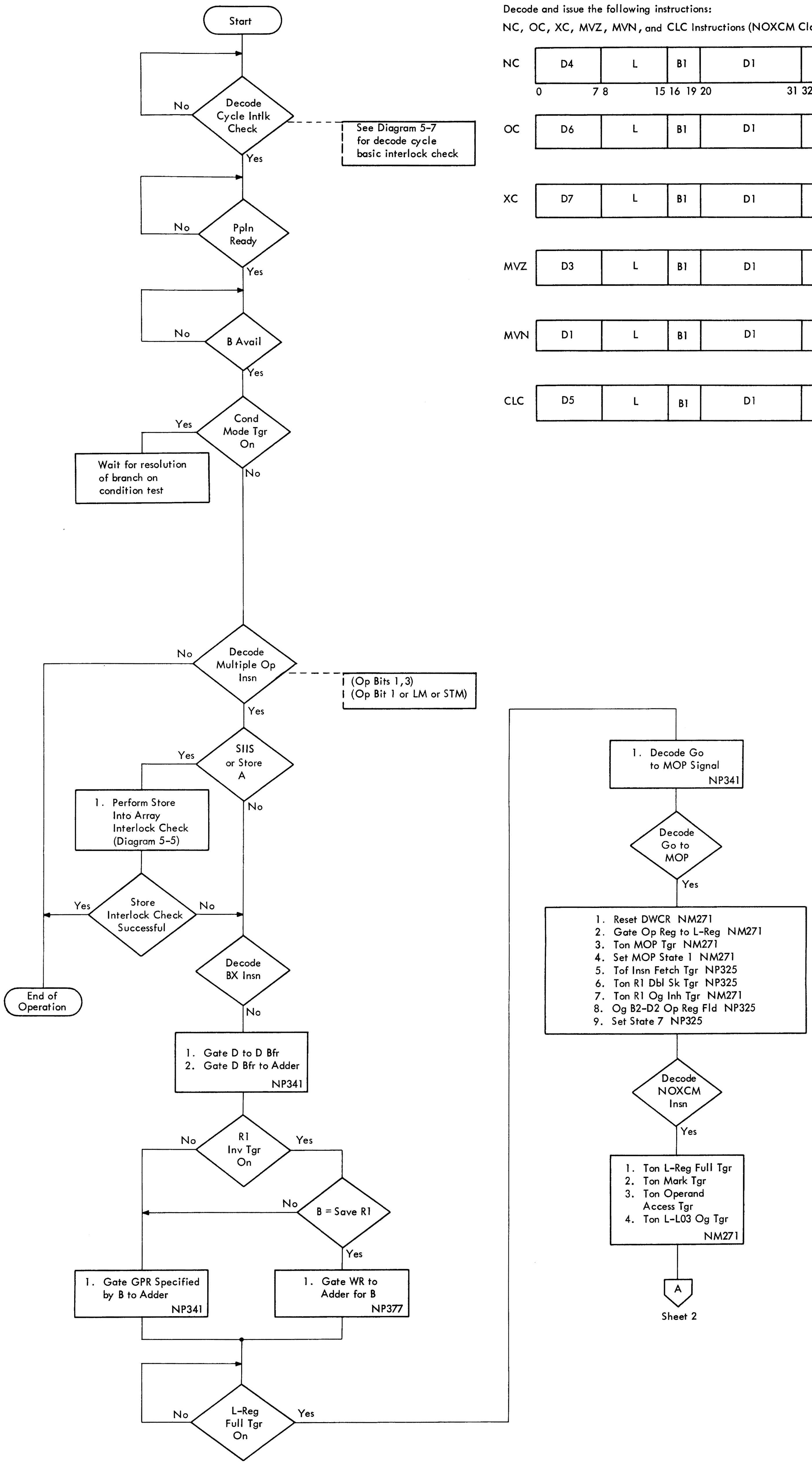


DIAGRAM 5-30. NOXCM SEQUENCE (SHEET 1 OF 3)

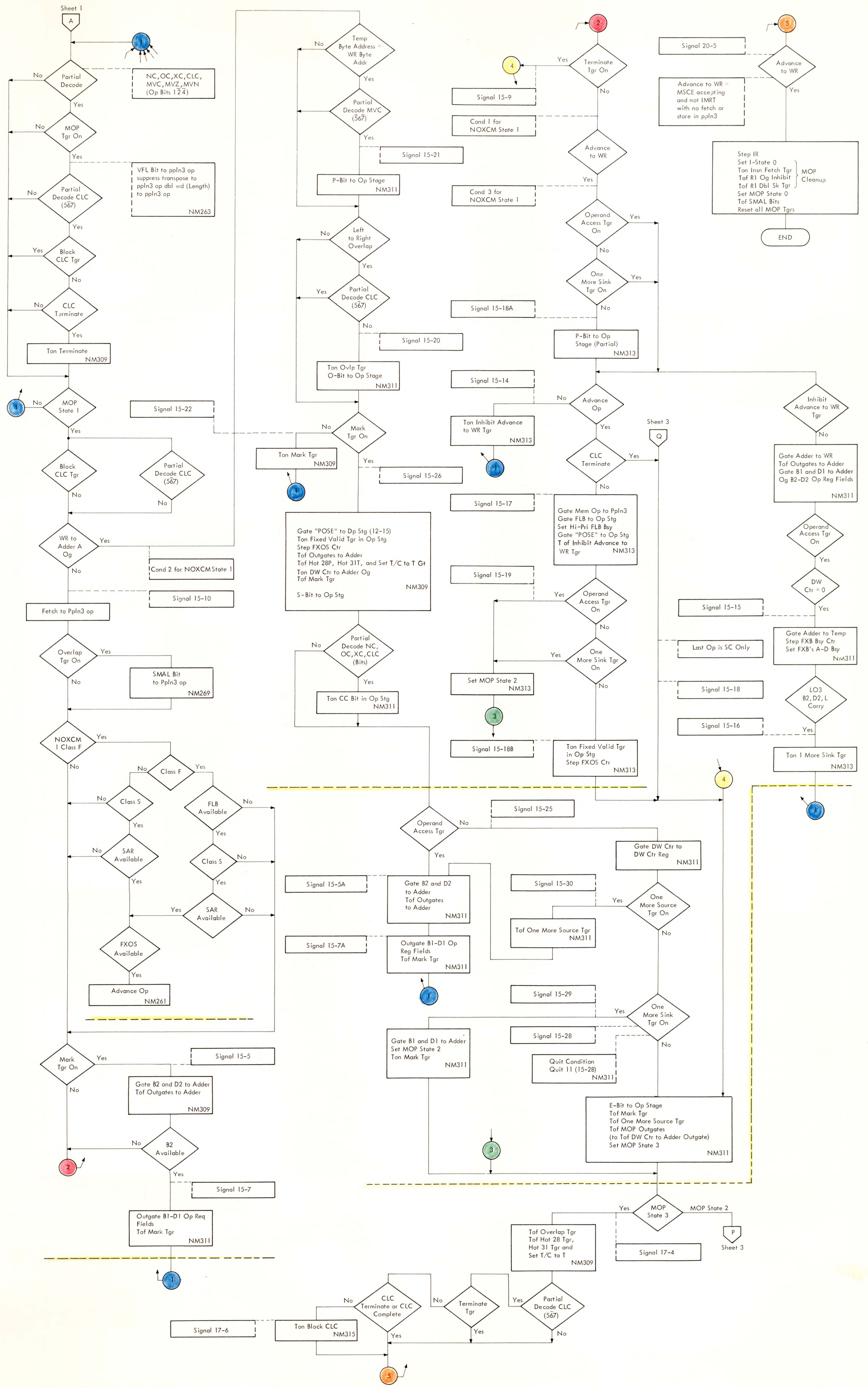


DIAGRAM 5-30. NOXCM SEQUENCE (SHEET 2 OF 3)

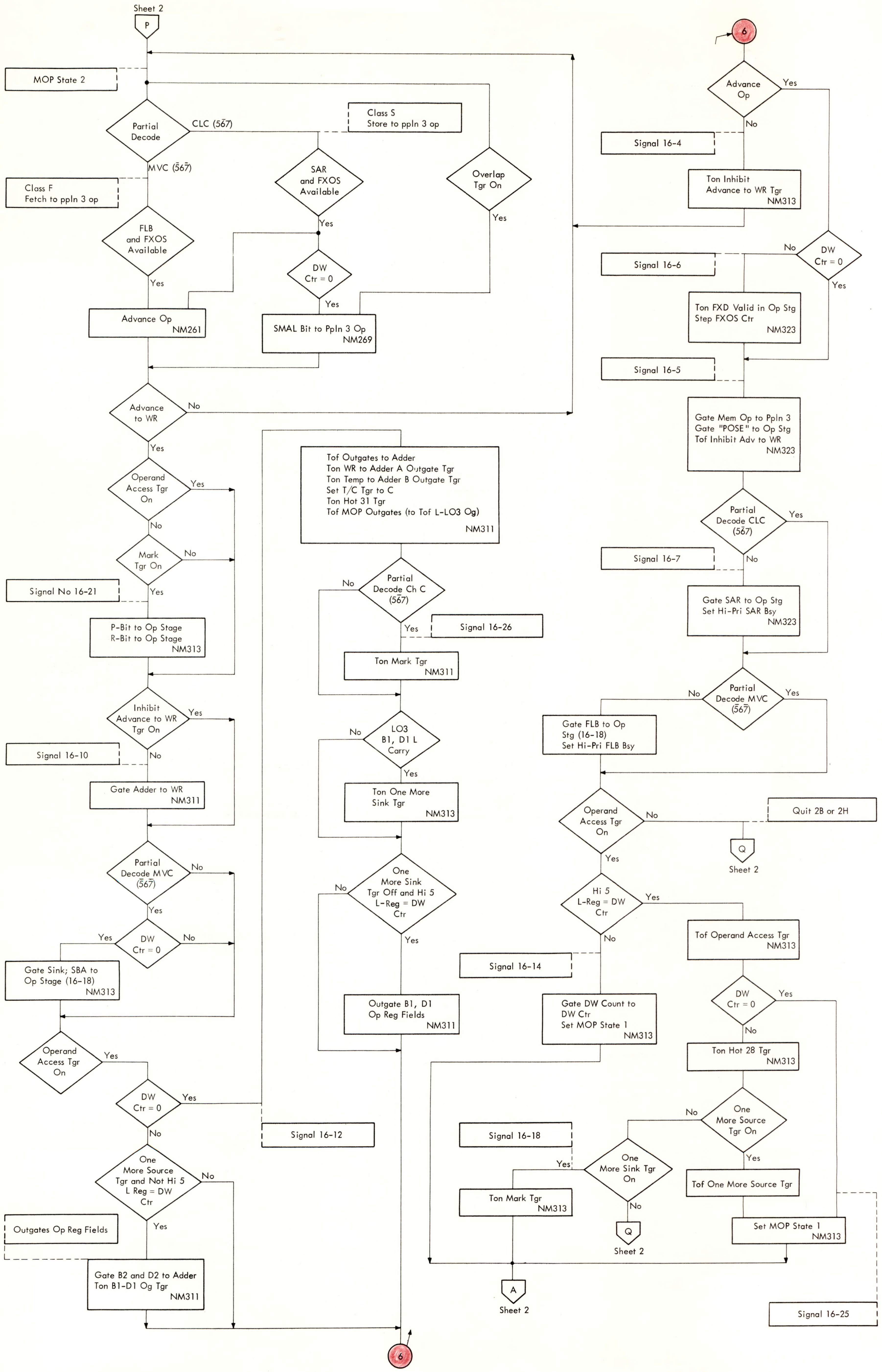
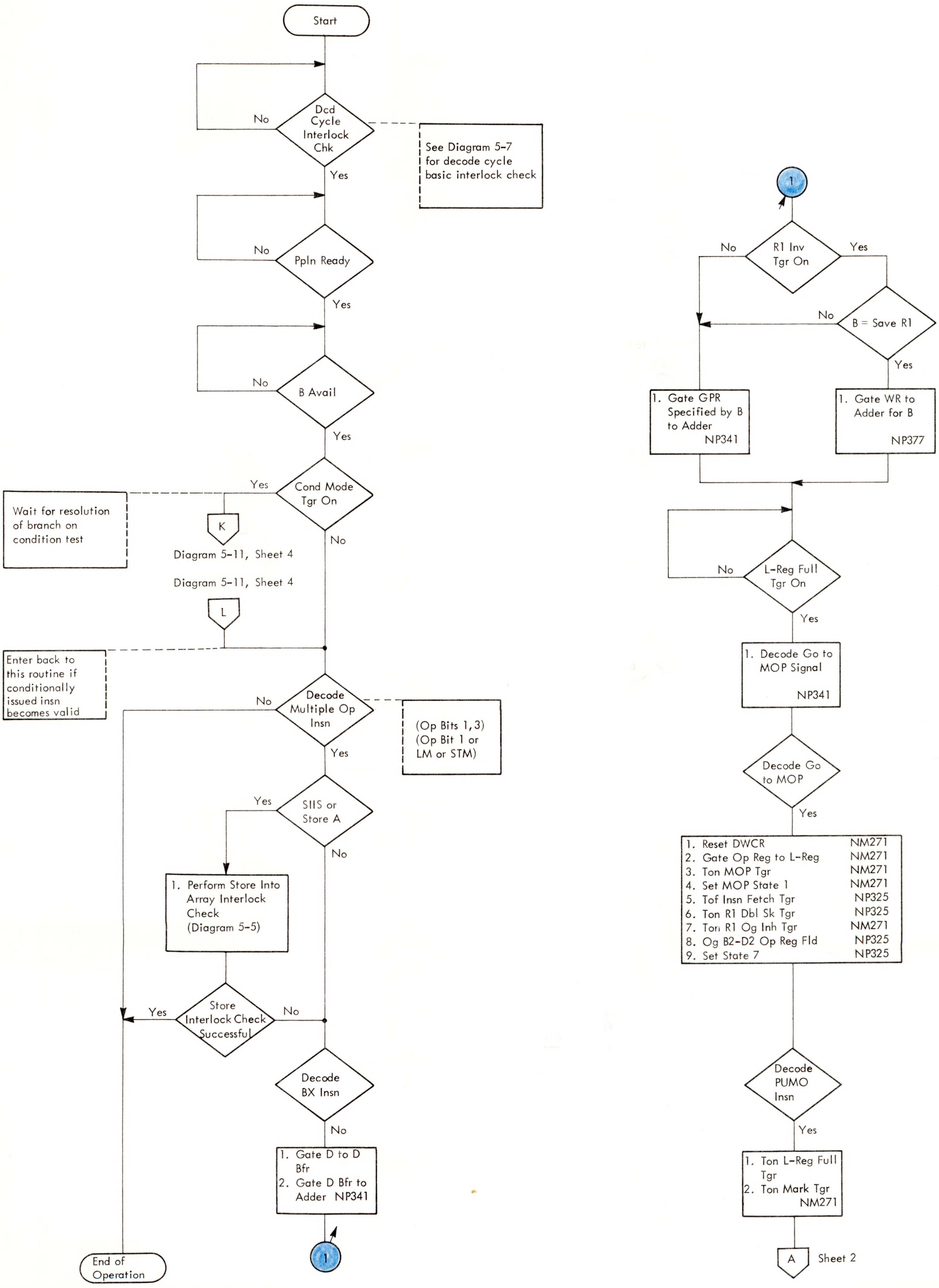


DIAGRAM 5-30. NOXCM SEQUENCE (SHEET 3 OF 3)

Objective:

Decode and issue the following instructions: Pack, Unpack, and Move With Ofset

	F2	L1	L2	B1	D1	B2	D2
Pack	0 7	8 11	12 15	16 19	20 31	32 35	36 47
	F3	L1	L2	B1	D1	B2	D2
Unpack	0 7	8 11	12 15	16 19	20 31	32 35	36 47
	F1	L1	L2	B1	D1	B2	D2
MVO	0 7	8 11	12 15	16 19	20 31	32 35	36 47



Start
Dcd Cycle Interlock Chk
No
Yes
See Diagram 5-7 for decode cycle basic interlock check
Ppln Ready
No
Yes
B Avail
No
Yes
Cond Mode Tgr On
Yes
No
Wait for resolution of branch on condition test
K
Diagram 5-11, Sheet 4
Diagram 5-11, Sheet 4
L
Enter back to this routine if conditionally issued insn becomes valid
Decode Multiple Op Insn
No
Yes
(Op Bits 1,3) (Op Bit 1 or LM or STM)
SIIS or Store A
Yes
No
1. Perform Store Into Array Interlock Check (Diagram 5-5)
Store Interlock Check Successful
Yes
No
Decode BX Insn
No
1. Gate D to D Bfr
2. Gate D Bfr to Adder NP341
1
End of Operation
1
R1 Inv Tgr On
No
Yes
B = Save R1
No
Yes
1. Gate GPR Specified by B to Adder NP341
1. Gate WR to Adder for B NP377
L-Reg Full Tgr On
No
Yes
1. Decode Go to MOP Signal NP341
Decode Go to MOP
Yes
1. Reset DWCR NM271
2. Gate Op Reg to L-Reg NM271
3. Ton MOP Tgr NM271
4. Set MOP State 1 NM271
5. Tof Insn Fetch Tgr NP325
6. Ton R1 Dbl Sk Tgr NP325
7. Ton R1 Og Inh Tgr NM271
8. Og B2-D2 Op Reg Fld NP325
9. Set State 7 NP325
Decode PUMO Insn
Yes
1. Ton L-Reg Full Tgr
2. Ton Mark Tgr NM271
A
Sheet 2

DIAGRAM 5-31. PUMO SEQUENCE (SHEET 1 OF 3)

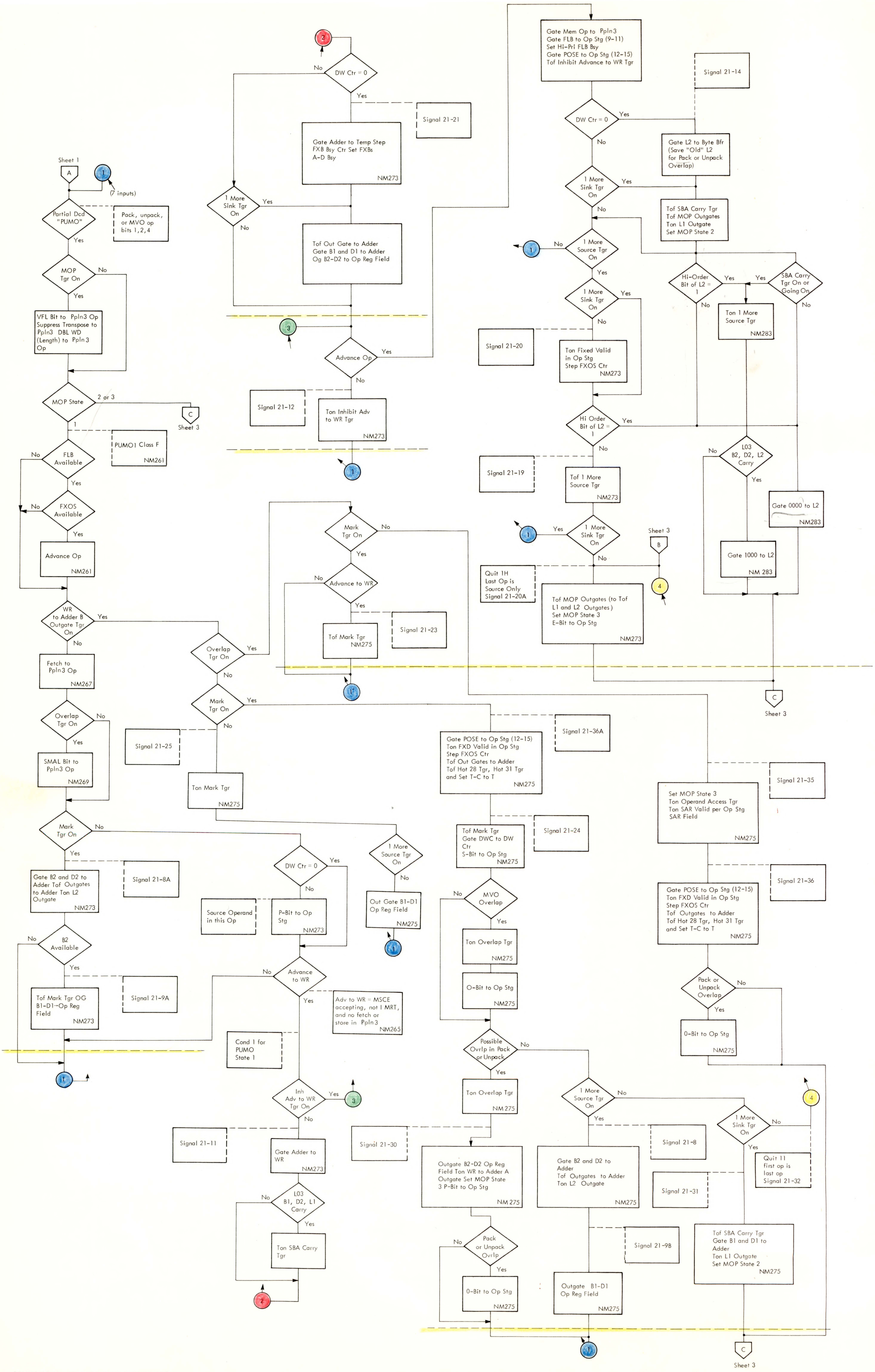

Sheet 1
A
1
(7 inputs)
Partial Dcd "PUMO"
Pack, unpack, or MVO op bits 1,2,4
Yes
MOP Tgr On
No
Yes
VFL Bit to Ppln3 Op Suppress Transpose to Ppln3 DBL WD (Length) to Ppln3 Op
MOP State
2 or 3
C
Sheet 3
1
PUMO1 Class F
NM261
FLB Available
No
Yes
FXOS Available
No
Yes
Advance Op
NM261
WR to Adder B Outgate Tgr On
Yes
No
Fetch to Ppln3 Op
NM267
Overlap Tgr On
No
Yes
SMAL Bit to Ppln3 Op
NM269
Mark Tgr On
No
Yes
Gate B2 and D2 to Adder Tof Outgates to Adder Ton L2 Outgate
NM273
Signal 21-8A
B2 Available
No
Yes
Tof Mark Tgr OG B1-D1→Op Reg Field
NM273
Signal 21-9A
1
2
No
DW Ctr = 0
Yes
Signal 21-21
Gate Adder to Temp Step FXB Bsy Ctr Set FXBs A-D Bsy
NM273
1 More Sink Tgr On
Yes
No
Tof Out Gate to Adder Gate B1 and D1 to Adder Og B2-D2 to Op Reg Field
3
Advance Op
Yes
No
Signal 21-12
Ton Inhibit Adv to WR Tgr
NM273
1
Overlap Tgr On
Yes
No
Mark Tgr On
Yes
No
Signal 21-25
Ton Mark Tgr
NM275
Mark Tgr On
No
Yes
Advance to WR
No
Yes
Tof Mark Tgr
NM275
Signal 21-23
1
DW Ctr = 0
Yes
No
Source Operand in this Op
P-Bit to Op Stg
NM273
1 More Source Tgr On
No
Out Gate B1-D1 Op Reg Field
NM275
1
Advance to WR
No
Yes
Adv to WR = MSCE accepting, not I MRT, and no fetch or store in Ppln3
NM265
Cond 1 for PUMO State 1
Inh Adv to WR Tgr On
Yes
3
No
Signal 21-11
Gate Adder to WR
NM273
L03 B1, D2, L1 Carry
No
Yes
Ton SBA Carry Tgr
2
Gate Mem Op to Ppln3 Gate FLB to Op Stg (9-11) Set Hi-Pri FLB Bsy Gate POSE to Op Stg (12-15) Tof Inhibit Advance to WR Tgr
Signal 21-14
DW Ctr = 0
Yes
No
Gate L2 to Byte Bfr (Save "Old" L2 for Pack or Unpack Overlap)
1 More Sink Tgr On
Yes
No
Tof SBA Carry Tgr Tof MOP Outgates Ton L1 Outgate Set MOP State 2
1
No
1 More Source Tgr On
Yes
1 More Sink Tgr On
Yes
No
Signal 21-20
Ton Fixed Valid in Op Stg Step FXOS Ctr
NM273
Hi-Order Bit of L2 = 1
Yes
No
SBA Carry Tgr On or Going On
Yes
No
Ton 1 More Source Tgr
NM283
Hi Order Bit of L2 = 1
Yes
No
Signal 21-19
Tof 1 More Source Tgr
NM273
L03 B2, D2, L2 Carry
No
Yes
Gate 0000 to L2
NM283
1
Yes
1 More Sink Tgr On
No
Sheet 3
B
4
Gate 1000 to L2
NM 283
Quit 1H Last Op is Source Only Signal 21-20A
Tof MOP Outgates (to Tof L1 and L2 Outgates) Set MOP State 3 E-Bit to Op Stg
NM273
C
Sheet 3
Gate POSE to Op Stg (12-15) Ton FXD Valid in Op Stg Step FXOS Ctr Tof Out Gates to Adder Tof Hot 28 Tgr, Hot 31 Tgr and Set T-C to T
NM275
Signal 21-36A
Tof Mark Tgr Gate DWC to DW Ctr S-Bit to Op Stg
NM275
Signal 21-24
MVO Overlap
No
Yes
Ton Overlap Tgr
NM275
O-Bit to Op Stg
NM275
Possible Ovrlp in Pack or Unpack
No
Yes
Ton Overlap Tgr
NM275
Signal 21-30
Outgate B2-D2 Op Reg Field Ton WR to Adder A Outgate Set MOP State 3 P-Bit to Op Stg
NM 275
Pack or Unpack Ovrlp
No
Yes
O-Bit to Op Stg
NM275
1
Set MOP State 3 Ton Operand Access Tgr Ton SAR Valid per Op Stg SAR Field
NM275
Signal 21-35
Gate POSE to Op Stg (12-15) Ton FXD Valid in Op Stg Step FXOS Ctr Tof Outgates to Adder Tof Hot 28 Tgr, Hot 31 Tgr and Set T-C to T
NM275
Signal 21-36
Pack or Unpack Overlap
No
Yes
O-Bit to Op Stg
NM275
1 More Source Tgr On
No
Yes
Signal 21-8
Gate B2 and D2 to Adder Tof Outgates to Adder Ton L2 Outgate
NM275
Signal 21-31
Signal 21-9B
Outgate B1-D1 Op Reg Field
NM275
4
1 More Sink Tgr On
No
Yes
Quit 11 first op is last op Signal 21-32
Tof SBA Carry Tgr Gate B1 and D1 to Adder Ton L1 Outgate Set MOP State 2
NM275
C
Sheet 3

DIAGRAM 5-31. PUMO SEQUENCE (SHEET 2 OF 3)

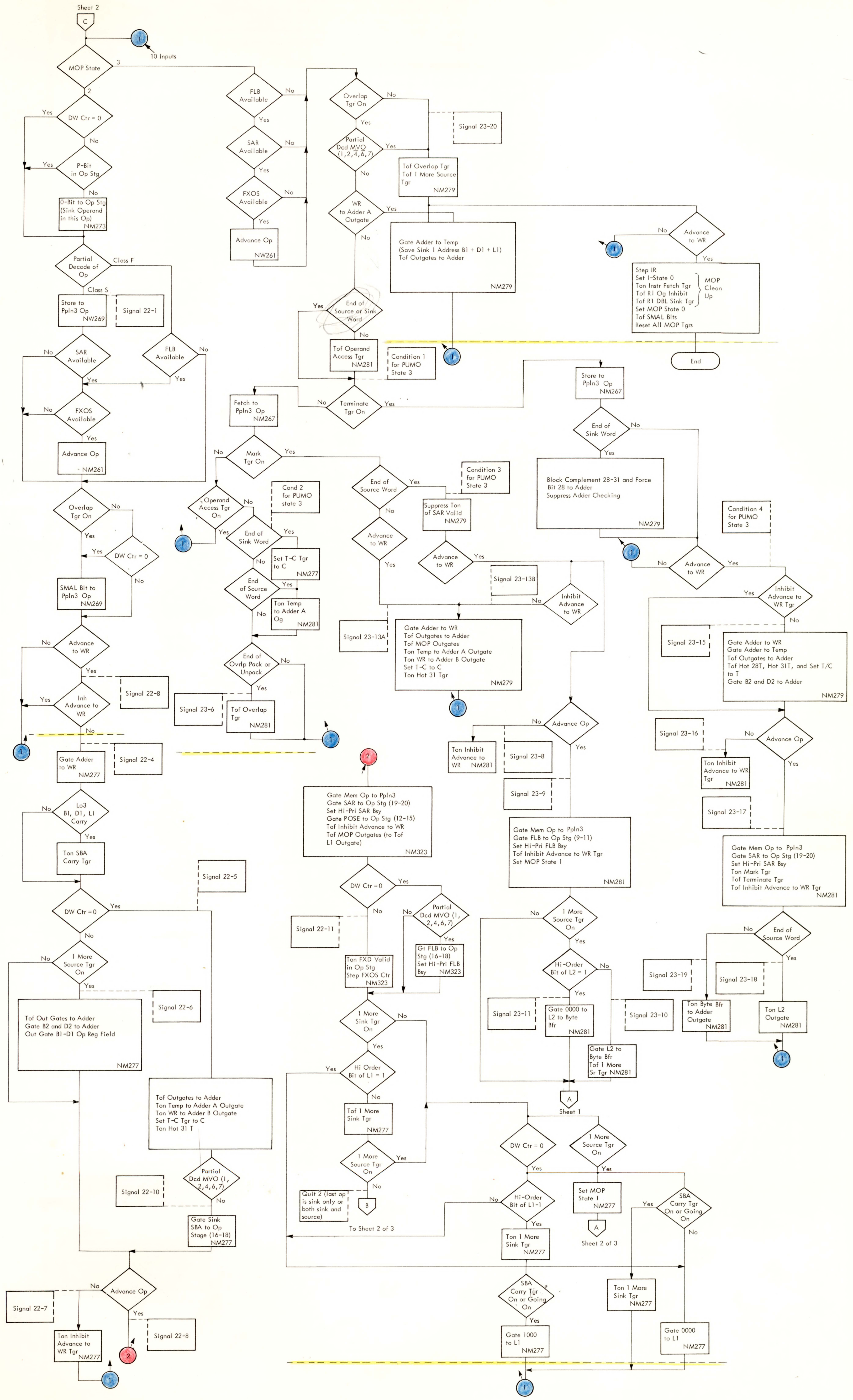


DIAGRAM 5-31. PUMO SEQUENCE (SHEET 3 OF 3)

Objectives:

1. Decode and issue an ED or EDMK instruction.
2. Check all interlocks that may detain or cancel the instruction.
3. Initially fetch two pattern and source words for VFLEU.
4. After first pattern (or source) word is requested by VFLEU, a subsequent reference for a pattern (or source) word is made each time the VFLEU finishes processing a pattern (or source) word.
5. VFLEU steps off an Op whenever it requests a pattern word, and the I-Box issues an op whenever it references a pattern word.
6. EDMK - Record byte address for first significant result digit by forming byte address and causing this byte address to be stored in GPR1.

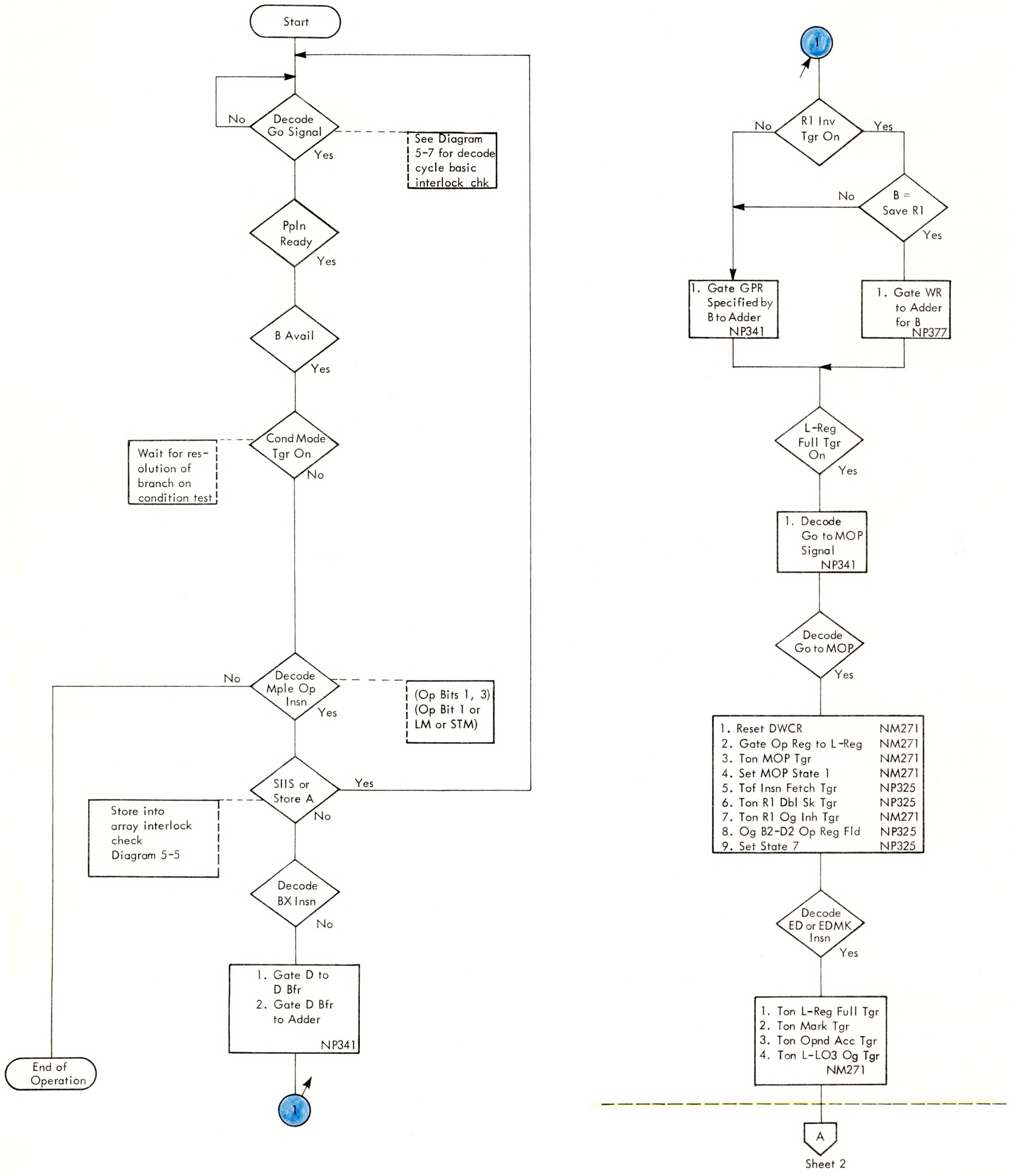


DIAGRAM 5-32. ED, EDMK SEQUENCE (SHEET 1 OF 4)

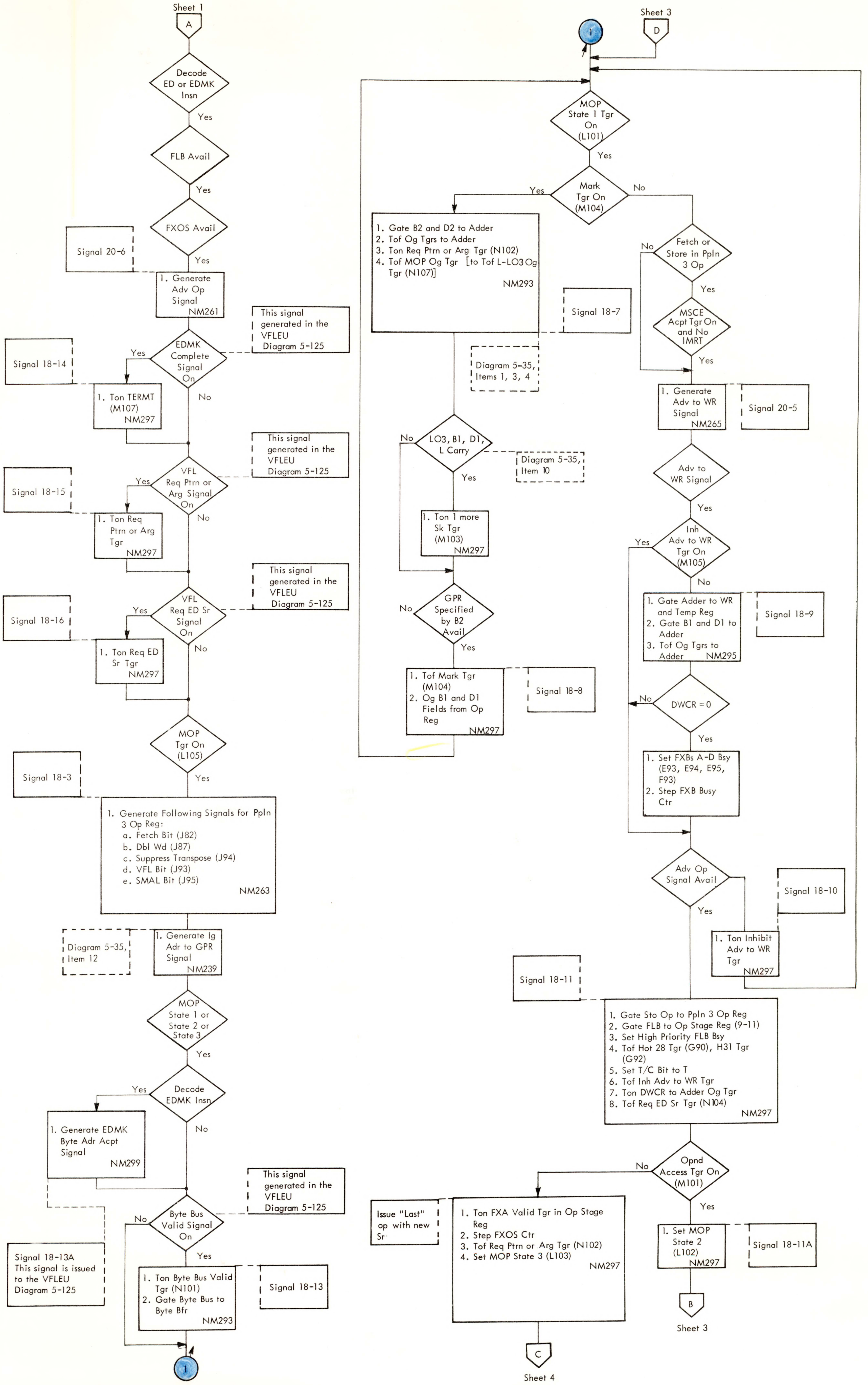

Sheet 1
A
Decode ED or EDMK Insn
Yes
FLB Avail
Yes
FXOS Avail
Yes
Signal 20-6
1. Generate Adv Op Signal
NM261
This signal generated in the VFLEU Diagram 5-125
EDMK Complete Signal On
Yes
No
Signal 18-14
1. Ton TERMT (M107)
NM297
This signal generated in the VFLEU Diagram 5-125
VFL Req Ptrn or Arg Signal On
Yes
No
Signal 18-15
1. Ton Req Ptrn or Arg Tgr
NM297
This signal generated in the VFLEU Diagram 5-125
VFL Req ED Sr Signal On
Yes
No
Signal 18-16
1. Ton Req ED Sr Tgr
NM297
MOP Tgr On (L105)
Yes
Signal 18-3
1. Generate Following Signals for Ppln 3 Op Reg:
a. Fetch Bit (J82)
b. Dbl Wd (J87)
c. Suppress Transpose (J94)
d. VFL Bit (J93)
e. SMAL Bit (J95)
NM263
Diagram 5-35, Item 12
1. Generate Ig Adr to GPR Signal
NM239
MOP State 1 or State 2 or State 3
Yes
Decode EDMK Insn
Yes
No
1. Generate EDMK Byte Adr Acpt Signal
NM299
This signal generated in the VFLEU Diagram 5-125
Byte Bus Valid Signal On
No
Yes
Signal 18-13A This signal is issued to the VFLEU Diagram 5-125
1. Ton Byte Bus Valid Tgr (N101)
2. Gate Byte Bus to Byte Bfr
NM293
Signal 18-13
1
Sheet 3
D
1
MOP State 1 Tgr On (L101)
Yes
Mark Tgr On (M104)
Yes
No
1. Gate B2 and D2 to Adder
2. Tof Og Tgrs to Adder
3. Ton Req Ptrn or Arg Tgr (N102)
4. Tof MOP Og Tgr [to Tof L-LO3 Og Tgr (N107)]
NM293
Signal 18-7
Diagram 5-35, Items 1, 3, 4
LO3, B1, D1, L Carry
No
Yes
Diagram 5-35, Item 10
1. Ton 1 more Sk Tgr (M103)
NM297
GPR Specified by B2 Avail
No
Yes
1. Tof Mark Tgr (M104)
2. Og B1 and D1 Fields from Op Reg
NM297
Signal 18-8
Fetch or Store in Ppln 3 Op
No
Yes
MSCE Acpt Tgr On and No IMRT
Yes
1. Generate Adv to WR Signal
NM265
Signal 20-5
Adv to WR Signal
Yes
Inh Adv to WR Tgr On (M105)
Yes
No
1. Gate Adder to WR and Temp Reg
2. Gate B1 and D1 to Adder
3. Tof Og Tgrs to Adder
NM295
Signal 18-9
DWCR = 0
No
Yes
1. Set FXBs A-D Bsy (E93, E94, E95, F93)
2. Step FXB Busy Ctr
Adv Op Signal Avail
Yes
Signal 18-10
1. Ton Inhibit Adv to WR Tgr
NM297
Signal 18-11
1. Gate Sto Op to Ppln 3 Op Reg
2. Gate FLB to Op Stage Reg (9-11)
3. Set High Priority FLB Bsy
4. Tof Hot 28 Tgr (G90), H31 Tgr (G92)
5. Set T/C Bit to T
6. Tof Inh Adv to WR Tgr
7. Ton DWCR to Adder Og Tgr
8. Tof Req ED Sr Tgr (N104)
NM297
Opnd Access Tgr On (M101)
No
Yes
Issue "Last" op with new Sr
1. Ton FXA Valid Tgr in Op Stage Reg
2. Step FXOS Ctr
3. Tof Req Ptrn or Arg Tgr (N102)
4. Set MOP State 3 (L103)
NM297
C
Sheet 4
1. Set MOP State 2 (L102)
NM297
Signal 18-11A
B
Sheet 3

DIAGRAM 5-32. ED, EDMK SEQUENCE (SHEET 2 OF 4)

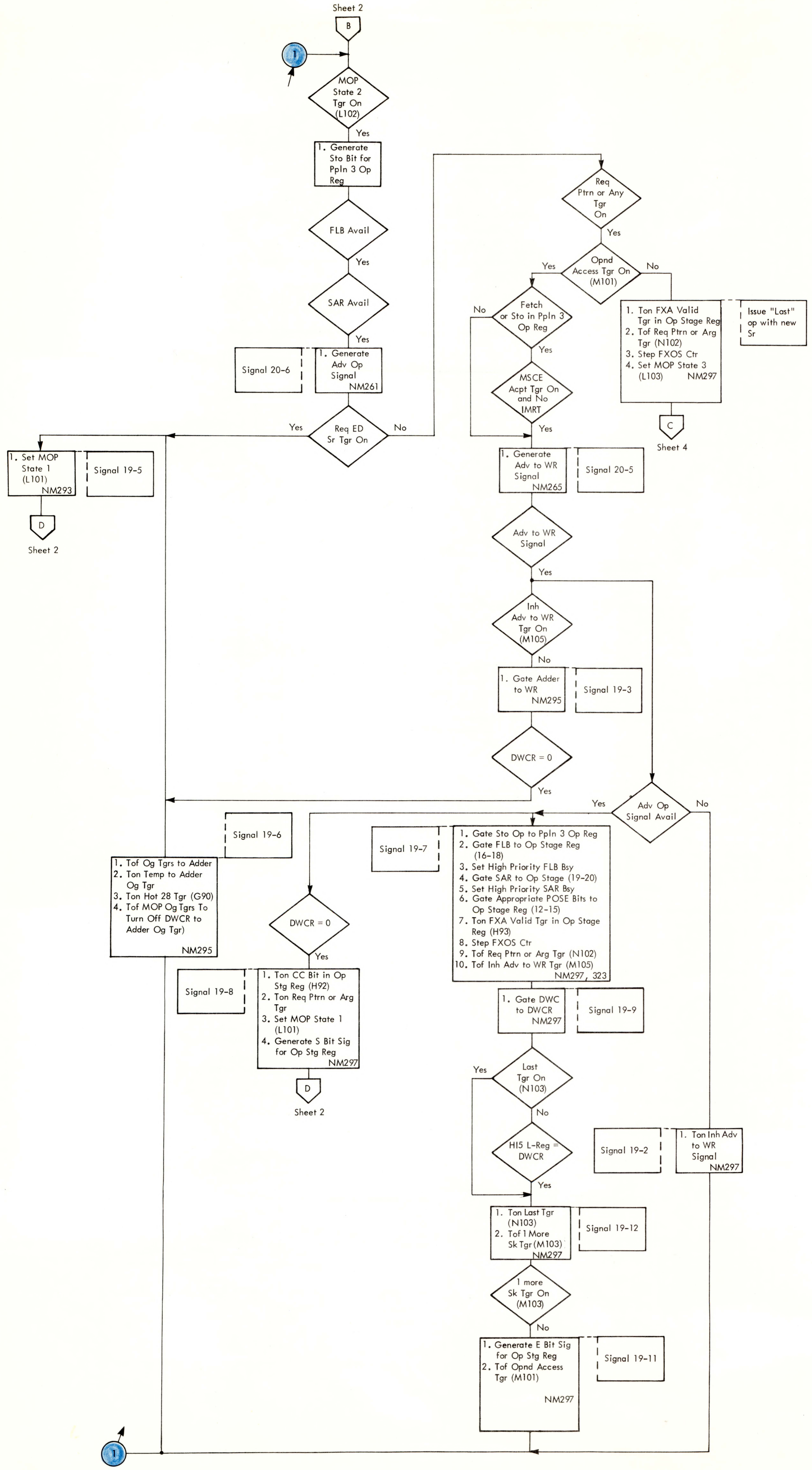

Sheet 2
B
1
MOP State 2 Tgr On (L102)
Yes
1. Generate Sto Bit for Ppln 3 Op Reg
FLB Avail
Yes
SAR Avail
Yes
Signal 20-6
1. Generate Adv Op Signal NM261
Yes
Req ED Sr Tgr On
No
1. Set MOP State 1 (L101) NM293
Signal 19-5
D
Sheet 2
Req Ptrn or Any Tgr On
Yes
Opnd Access Tgr On (M101)
Yes
No
1. Ton FXA Valid Tgr in Op Stage Reg
2. Tof Req Ptrn or Arg Tgr (N102)
3. Step FXOS Ctr
4. Set MOP State 3 (L103) NM297
Issue "Last" op with new Sr
C
Sheet 4
No
Fetch or Sto in Ppln 3 Op Reg
Yes
MSCE Acpt Tgr On and No IMRT
Yes
1. Generate Adv to WR Signal NM265
Signal 20-5
Adv to WR Signal
Yes
Inh Adv to WR Tgr On (M105)
No
1. Gate Adder to WR NM295
Signal 19-3
DWCR = 0
Yes
Yes
Adv Op Signal Avail
No
Signal 19-6
1. Tof Og Tgrs to Adder
2. Ton Temp to Adder Og Tgr
3. Ton Hot 28 Tgr (G90)
4. Tof MOP Og Tgrs To Turn Off DWCR to Adder Og Tgr)
NM295
DWCR = 0
Yes
Signal 19-8
1. Ton CC Bit in Op Stg Reg (H92)
2. Ton Req Ptrn or Arg Tgr
3. Set MOP State 1 (L101)
4. Generate S Bit Sig for Op Stg Reg
NM297
D
Sheet 2
Signal 19-7
1. Gate Sto Op to Ppln 3 Op Reg
2. Gate FLB to Op Stage Reg (16-18)
3. Set High Priority FLB Bsy
4. Gate SAR to Op Stage (19-20)
5. Set High Priority SAR Bsy
6. Gate Appropriate POSE Bits to Op Stage Reg (12-15)
7. Ton FXA Valid Tgr in Op Stage Reg (H93)
8. Step FXOS Ctr
9. Tof Req Ptrn or Arg Tgr (N102)
10. Tof Inh Adv to WR Tgr (M105)
NM297, 323
1. Gate DWC to DWCR NM297
Signal 19-9
Yes
Last Tgr On (N103)
No
HI5 L-Reg = DWCR
Yes
Signal 19-2
1. Ton Inh Adv to WR Signal NM297
1. Ton Last Tgr (N103)
2. Tof 1 More Sk Tgr (M103)
NM297
Signal 19-12
1 more Sk Tgr On (M103)
No
1. Generate E Bit Sig for Op Stg Reg
2. Tof Opnd Access Tgr (M101)
NM297
Signal 19-11
1

DIAGRAM 5-32. ED, EDMK SEQUENCE (SHEET 3 OF 4)

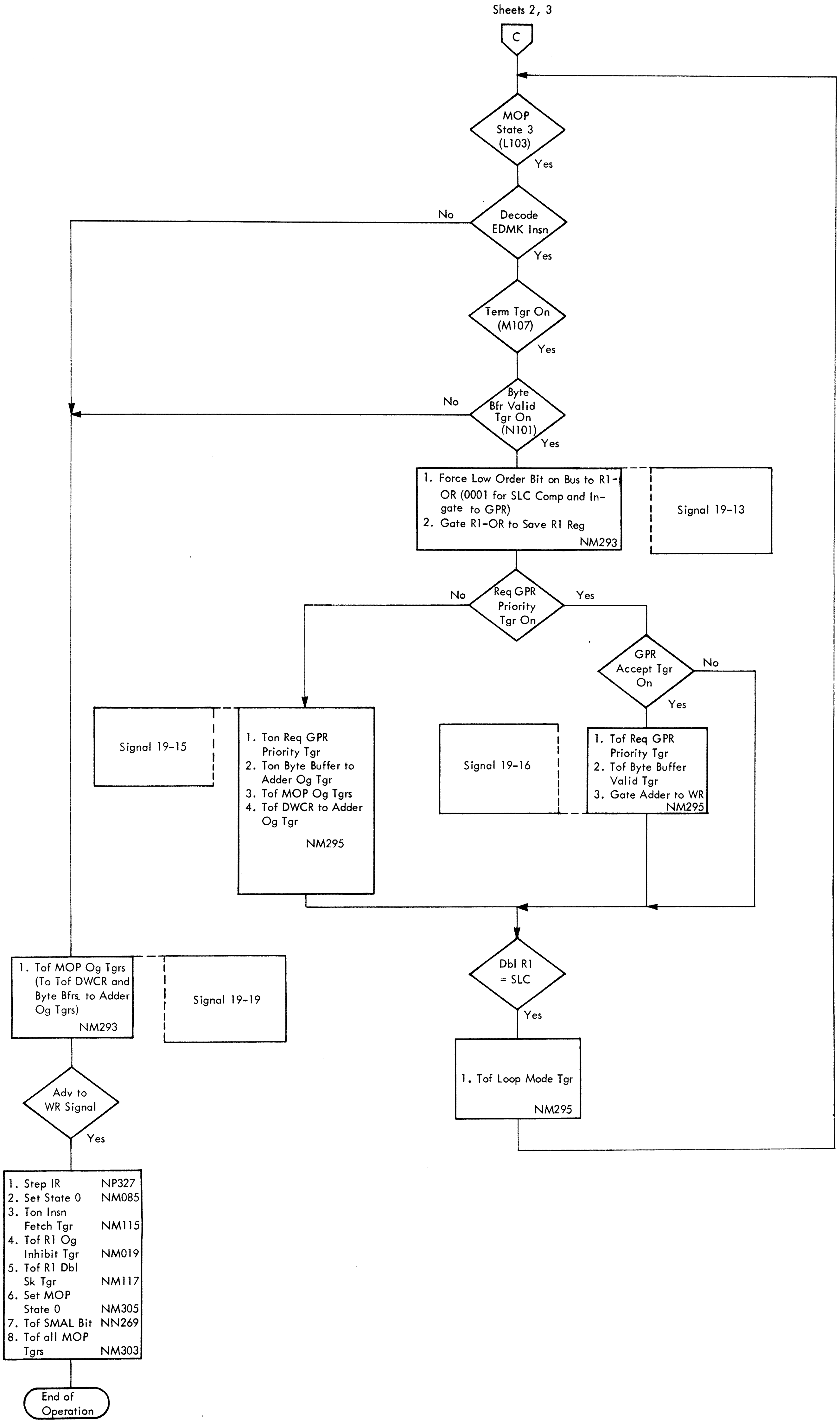

Sheets 2, 3
C
MOP State 3 (L103)
Yes
No
Decode EDMK Insn
Yes
Term Tgr On (M107)
Yes
No
Byte Bfr Valid Tgr On (N101)
Yes
1. Force Low Order Bit on Bus to R1-OR (0001 for SLC Comp and In-gate to GPR)
2. Gate R1-OR to Save R1 Reg
NM293
Signal 19-13
No
Req GPR Priority Tgr On
Yes
GPR Accept Tgr On
No
Yes
Signal 19-15
1. Ton Req GPR Priority Tgr
2. Ton Byte Buffer to Adder Og Tgr
3. Tof MOP Og Tgrs
4. Tof DWCR to Adder Og Tgr
NM295
Signal 19-16
1. Tof Req GPR Priority Tgr
2. Tof Byte Buffer Valid Tgr
3. Gate Adder to WR
NM295
1. Tof MOP Og Tgrs (To Tof DWCR and Byte Bfrs to Adder Og Tgrs)
NM293
Signal 19-19
Dbl R1 = SLC
Yes
1. Tof Loop Mode Tgr
NM295
Adv to WR Signal
Yes
1. Step IR NP327
2. Set State 0 NM085
3. Ton Insn Fetch Tgr NM115
4. Tof R1 Og Inhibit Tgr NM019
5. Tof R1 Dbl Sk Tgr NM117
6. Set MOP State 0 NM305
7. Tof SMAL Bit NN269
8. Tof all MOP Tgrs NM303
End of Operation

DIAGRAM 5-32. ED, EDMK SEQUENCE (SHEET 4 OF 4)

Precise Interrupt Signals (Program B)

Objectives:

1. Generate the various interrupt signals

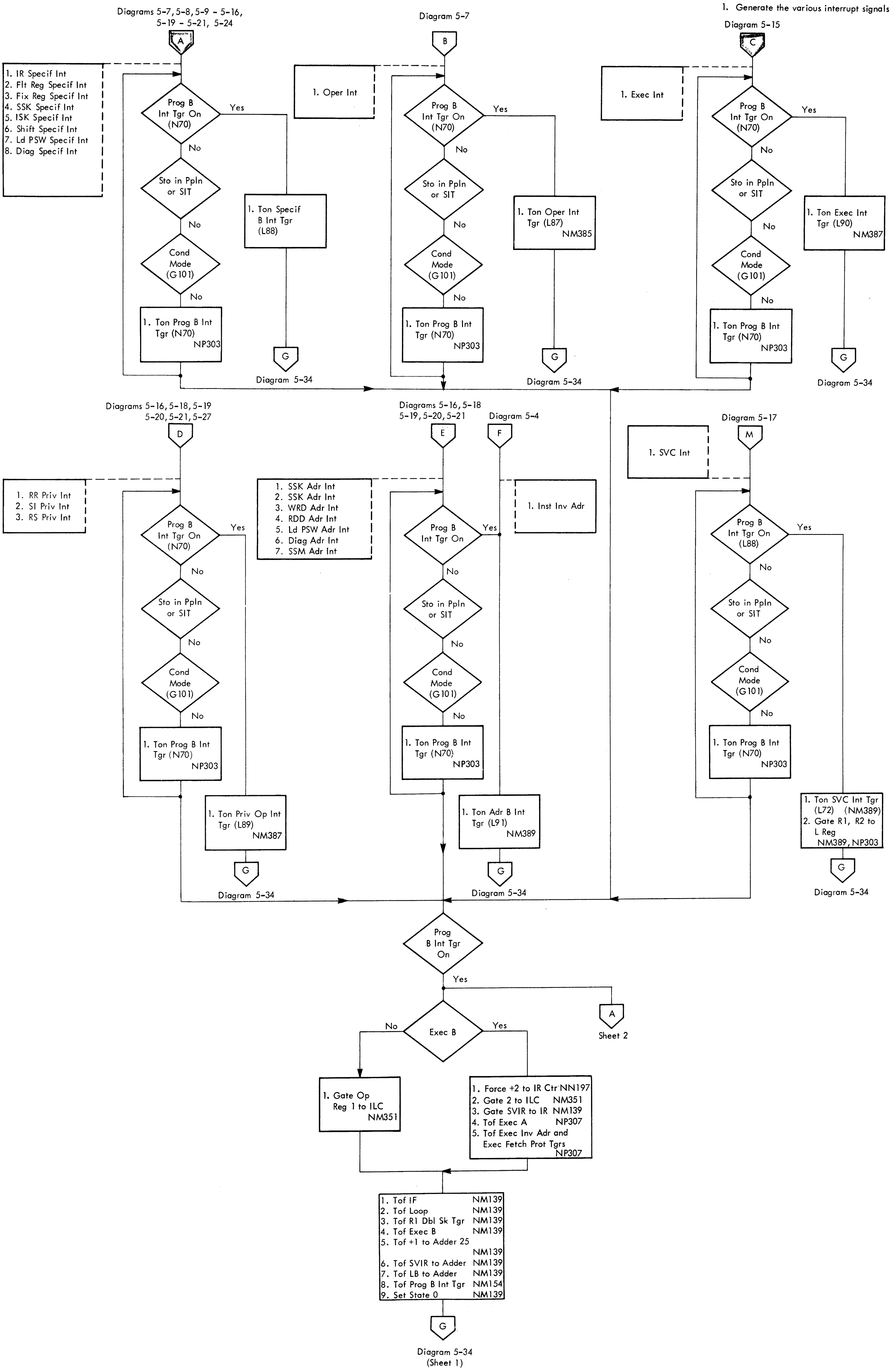
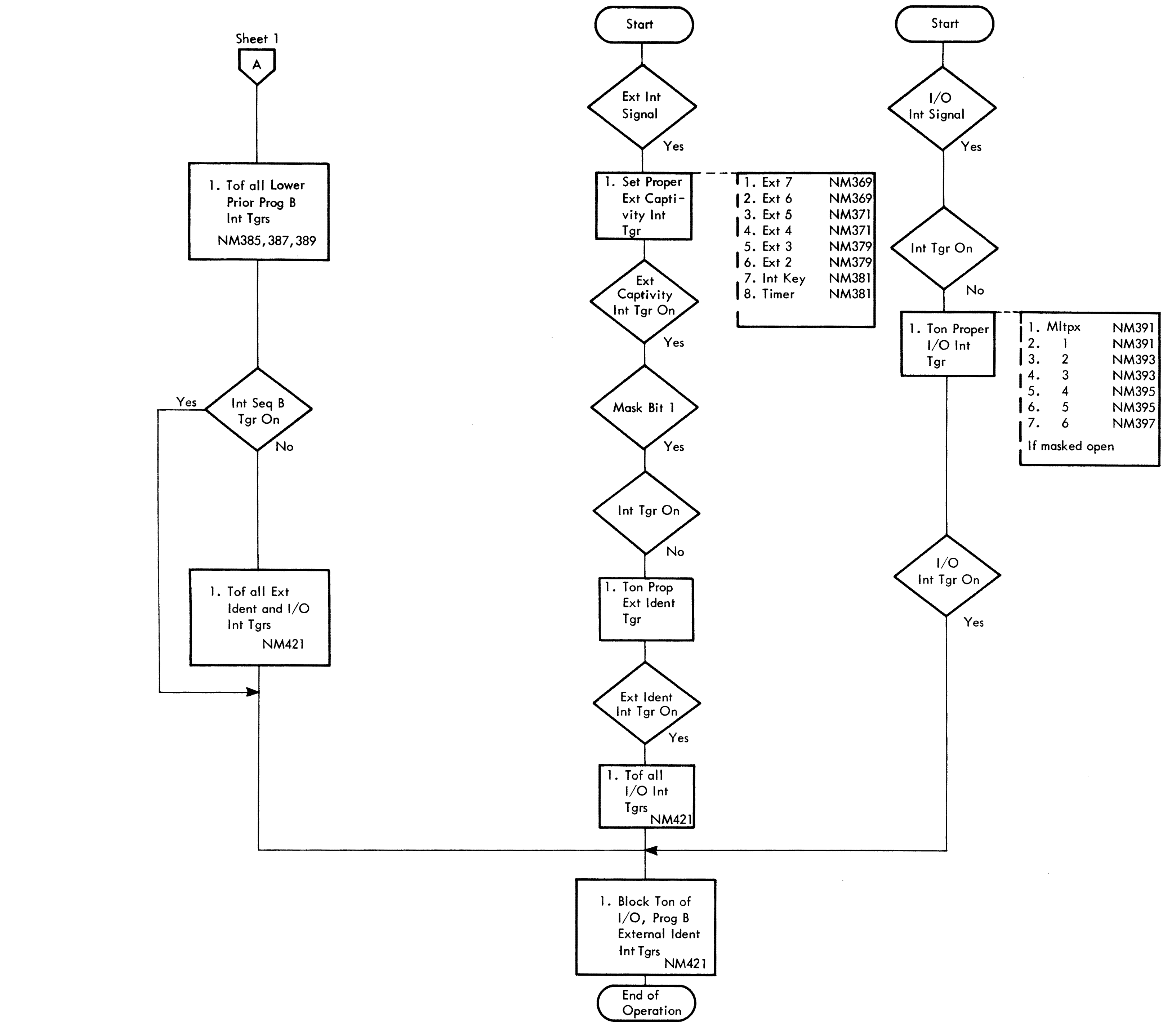


DIAGRAM 5-33. INTERRUPT SIGNALS (SHEET 1 OF 3)

Program B (Precise), External, and I/O Interrupt Signaling and Priority



Machine Check and Imprecise Interrupt Signals

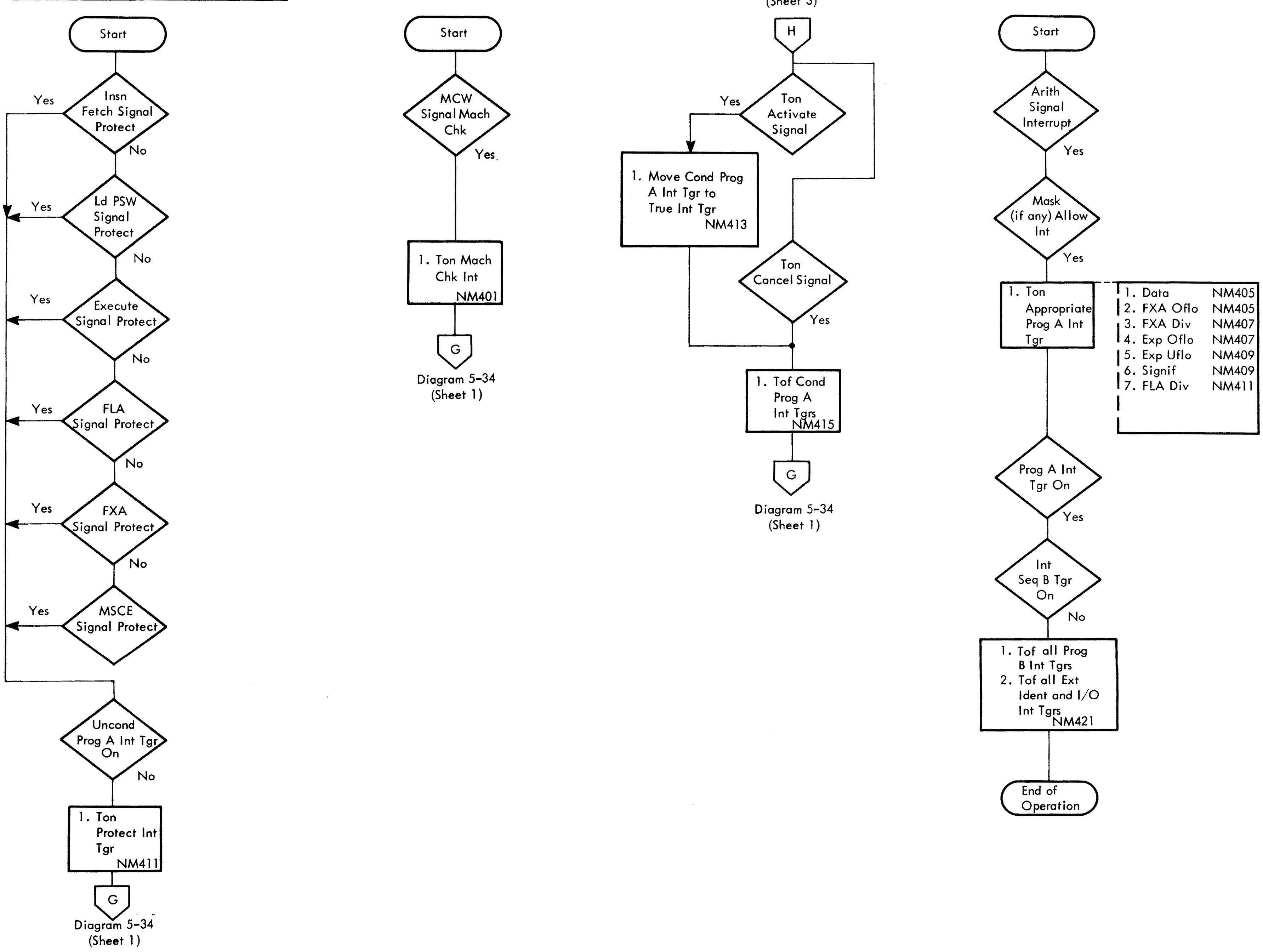
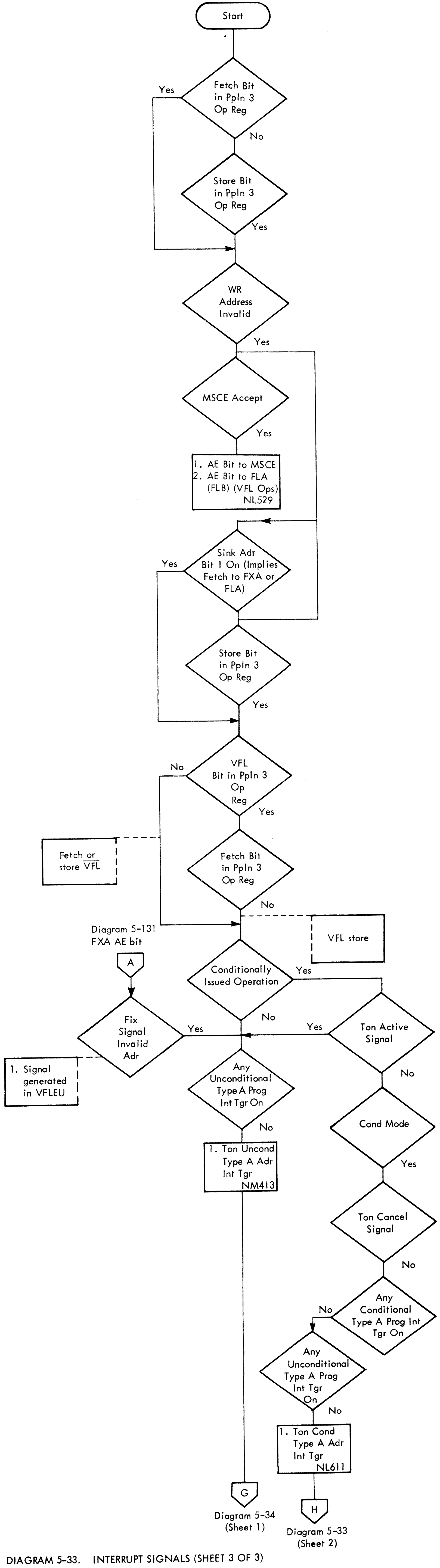


DIAGRAM 5-33. INTERRUPT SIGNALS (SHEET 2 OF 3)

Address Invalid Interrupt Signal Generation


Start
Fetch Bit in Ppln 3 Op Reg
Yes
No
Store Bit in Ppln 3 Op Reg
Yes
WR Address Invalid
Yes
MSCE Accept
Yes
1. AE Bit to MSCE
2. AE Bit to FLA (FLB) (VFL Ops)
NL529
Sink Adr Bit 1 On (Implies Fetch to FXA or FLA)
Yes
Store Bit in Ppln 3 Op Reg
Yes
VFL Bit in Ppln 3 Op Reg
No
Yes
Fetch or store VFL
Fetch Bit in Ppln 3 Op Reg
No
VFL store
Diagram 5-131 FXA AE bit
A
Fix Signal Invalid Adr
Yes
1. Signal generated in VFLEU
Conditionally Issued Operation
Yes
No
Ton Active Signal
Yes
No
Any Unconditional Type A Prog Int Tgr On
No
1. Ton Uncond Type A Adr Int Tgr
NM413
Cond Mode
Yes
Ton Cancel Signal
No
Any Conditional Type A Prog Int Tgr On
No
Any Unconditional Type A Prog Int Tgr On
No
1. Ton Cond Type A Adr Int Tgr
NL611
G
Diagram 5-34 (Sheet 1)
H
Diagram 5-33 (Sheet 2)

Imprecise Address Specification Interrupt Signal Generation

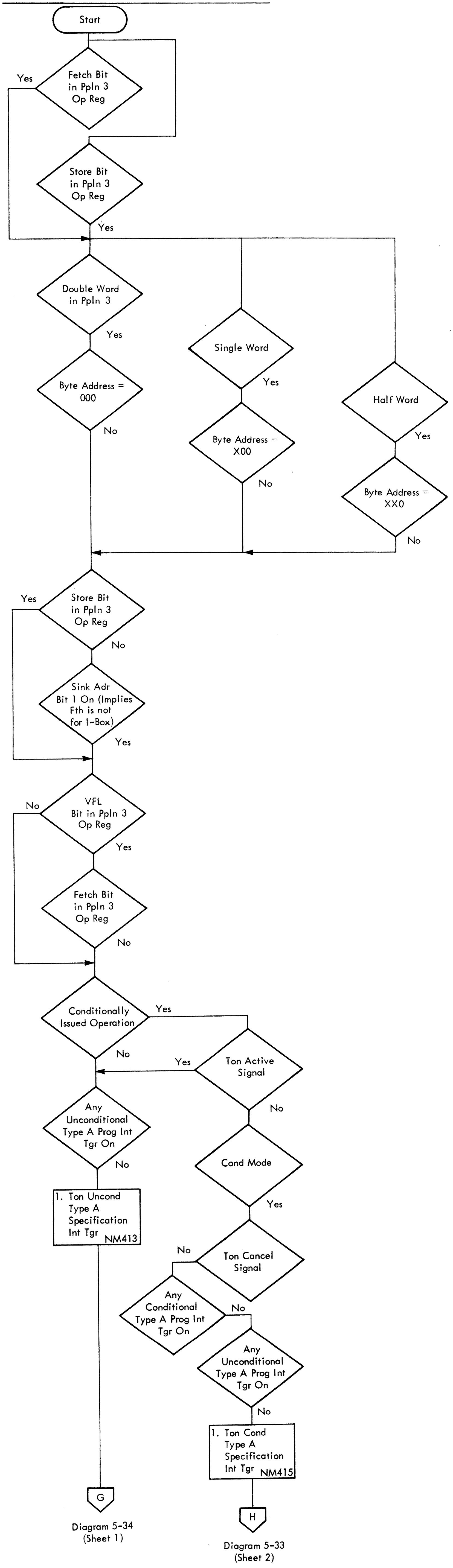

Start
Fetch Bit in Ppln 3 Op Reg
Yes
Store Bit in Ppln 3 Op Reg
Yes
Double Word in Ppln 3
Yes
Byte Address = 000
No
Single Word
Yes
Byte Address = X00
No
Half Word
Yes
Byte Address = XX0
No
Store Bit in Ppln 3 Op Reg
Yes
No
Sink Adr Bit 1 On (Implies Fth is not for I-Box)
Yes
VFL Bit in Ppln 3 Op Reg
No
Yes
Fetch Bit in Ppln 3 Op Reg
No
Conditionally Issued Operation
Yes
No
Ton Active Signal
Yes
No
Any Unconditional Type A Prog Int Tgr On
No
1. Ton Uncond Type A Specification Int Tgr
NM413
Cond Mode
Yes
Ton Cancel Signal
No
Any Conditional Type A Prog Int Tgr On
No
Any Unconditional Type A Prog Int Tgr On
No
1. Ton Cond Type A Specification Int Tgr
NM415
G
Diagram 5-34 (Sheet 1)
H
Diagram 5-33 (Sheet 2)

DIAGRAM 5-33. INTERRUPT SIGNALS (SHEET 3 OF 3)

Objectives:

1. Process all interruptions defined for the IBM System/360.
2. General steps during interruption sequencing are as follows.
   a. Stop I-Box decode
   b. Wait for I-Box to complete current instruction
   c. Fetch new PSW
   d. Initialize instruction fetch mechanism with instruction address in new PSW
   e. Wait for execution pipeline to empty.
   f. Store old PSW
   g. Return I-Box to decode
   h. Store IC of instruction (N)if interrupt is imprecise (ILC = 00)
   i. Store IC of instruction (N + 1) if interrupt is precise (ILC = True ILC)

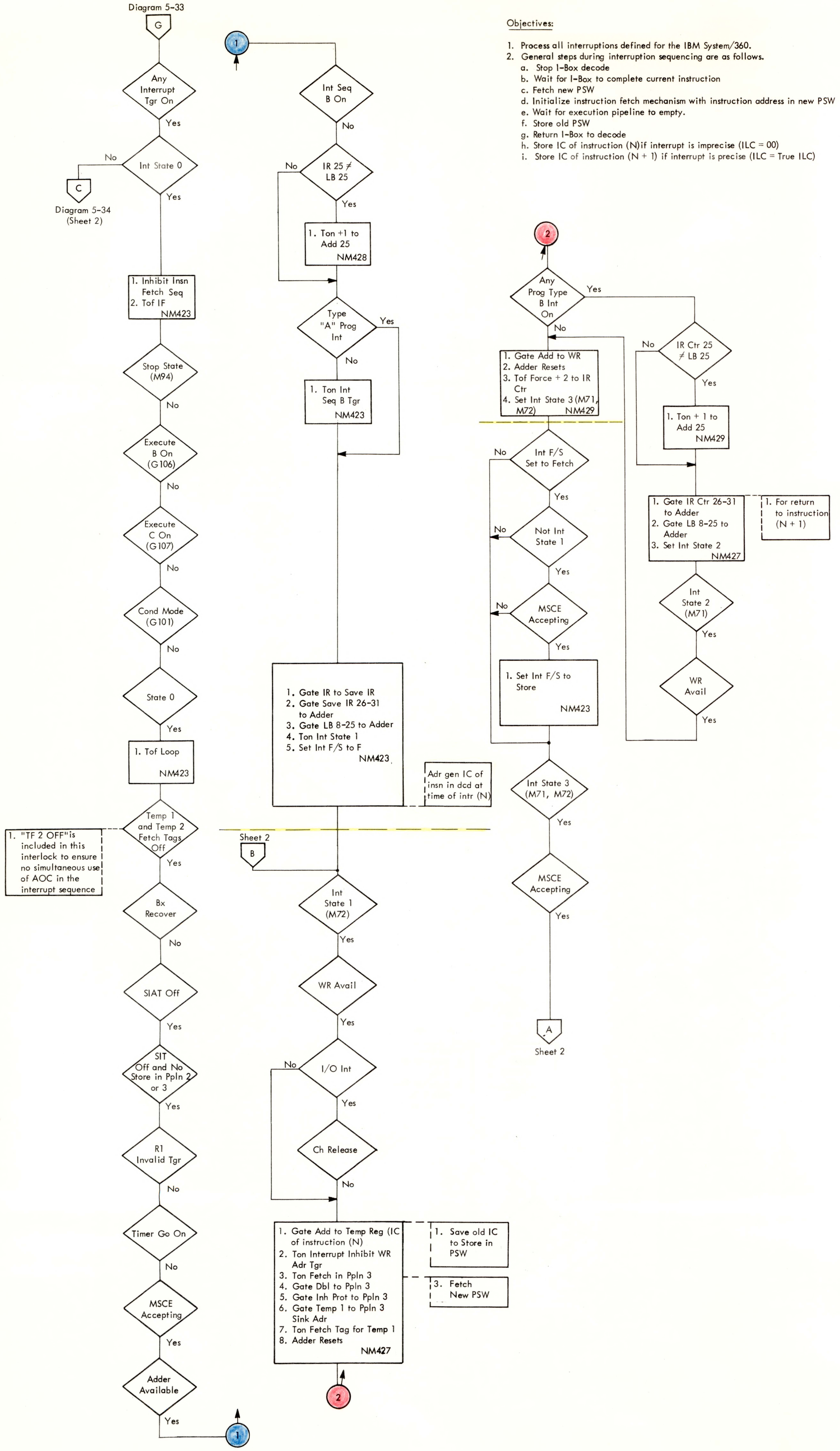


DIAGRAM 5-34. INTERRUPT SEQUENCING (SHEET 1 OF 2)

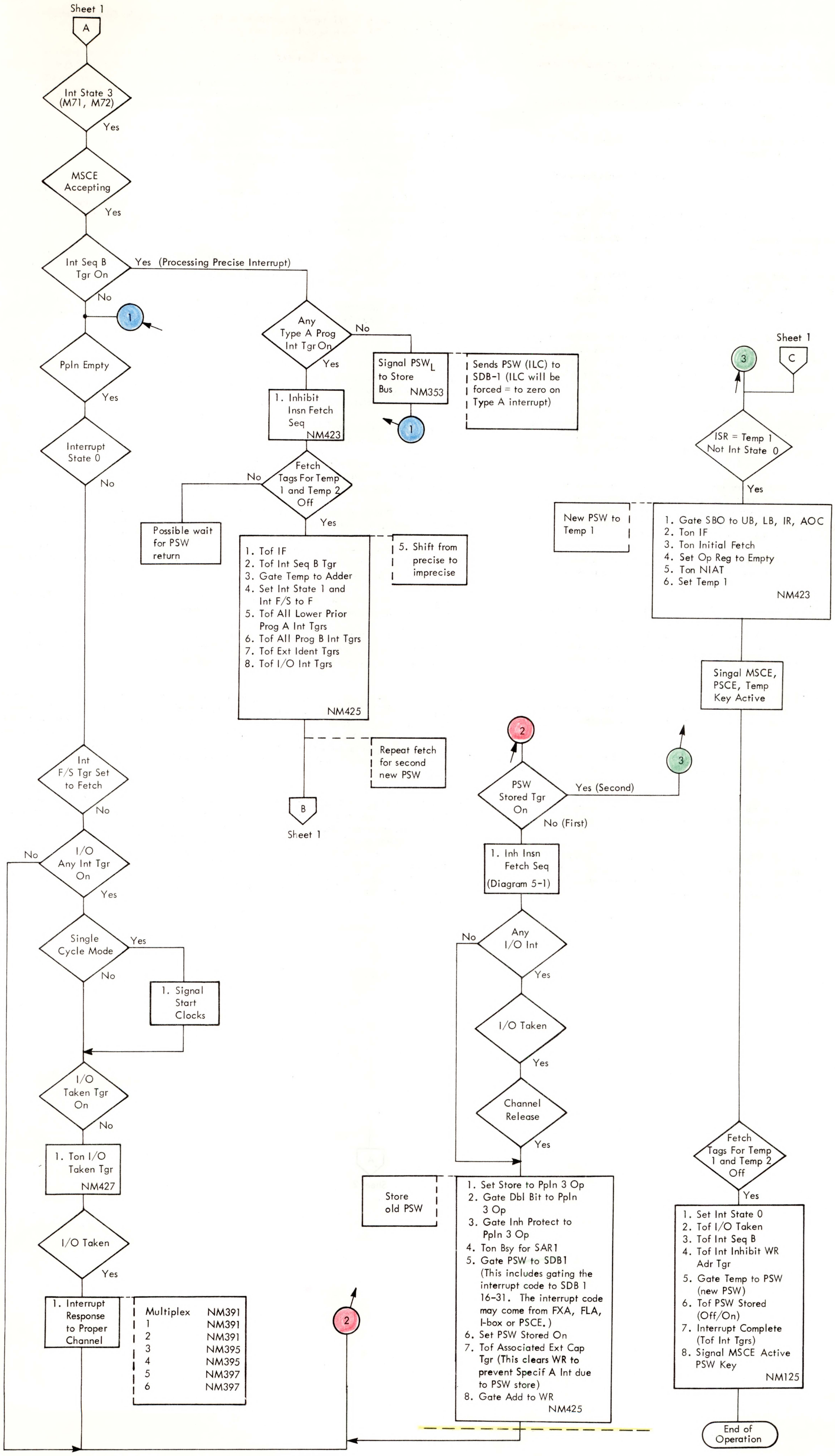

Sheet 1
A
Int State 3 (M71, M72)
Yes
MSCE Accepting
Yes
Int Seq B Tgr On
Yes (Processing Precise Interrupt)
No
1
Ppln Empty
Yes
Interrupt State 0
No
Any Type A Prog Int Tgr On
No
Yes
Signal PSW$_L$ to Store Bus NM353
Sends PSW (ILC) to SDB-1 (ILC will be forced = to zero on Type A interrupt)
1. Inhibit Insn Fetch Seq NM423
1
Fetch Tags For Temp 1 and Temp 2 Off
No
Possible wait for PSW return
Yes
1. Tof IF
2. Tof Int Seq B Tgr
3. Gate Temp to Adder
4. Set Int State 1 and Int F/S to F
5. Tof All Lower Prior Prog A Int Tgrs
6. Tof All Prog B Int Tgrs
7. Tof Ext Ident Tgrs
8. Tof I/O Int Tgrs
NM425
5. Shift from precise to imprecise
Repeat fetch for second new PSW
B
Sheet 1
Sheet 1
3
C
ISR = Temp 1 Not Int State 0
Yes
New PSW to Temp 1
1. Gate SBO to UB, LB, IR, AOC
2. Ton IF
3. Ton Initial Fetch
4. Set Op Reg to Empty
5. Ton NIAT
6. Set Temp 1
NM423
Singal MSCE, PSCE, Temp Key Active
2
3
PSW Stored Tgr On
Yes (Second)
No (First)
1. Inh Insn Fetch Seq (Diagram 5-1)
Any I/O Int
No
Yes
I/O Taken
Yes
Channel Release
Yes
Int F/S Tgr Set to Fetch
No
I/O Any Int Tgr On
No
Yes
Single Cycle Mode
Yes
No
1. Signal Start Clocks
I/O Taken Tgr On
No
1. Ton I/O Taken Tgr NM427
I/O Taken
Yes
1. Interrupt Response to Proper Channel
Multiplex NM391
1 NM391
2 NM391
3 NM395
4 NM395
5 NM397
6 NM397
2
Store old PSW
1. Set Store to Ppln 3 Op
2. Gate Dbl Bit to Ppln 3 Op
3. Gate Inh Protect to Ppln 3 Op
4. Ton Bsy for SAR1
5. Gate PSW to SDB1 (This includes gating the interrupt code to SDB 1 16-31. The interrupt code may come from FXA, FLA, I-box or PSCE.)
6. Set PSW Stored On
7. Tof Associated Ext Cap Tgr (This clears WR to prevent Specif A Int due to PSW store)
8. Gate Add to WR
NM425
Fetch Tags For Temp 1 and Temp 2 Off
Yes
1. Set Int State 0
2. Tof I/O Taken
3. Tof Int Seq B
4. Tof Int Inhibit WR Adr Tgr
5. Gate Temp to PSW (new PSW)
6. Tof PSW Stored (Off/On)
7. Interrupt Complete (Tof Int Tgrs)
8. Signal MSCE Active PSW Key
NM125
End of Operation

DIAGRAM 5-34. INTERRUPT SEQUENCING (SHEET 2 OF 2)

1. Tof MOP Og Tgrs - This line turns off the following outgate triggers: DWCR; Byte Buffer; R1 to R1 - OR; L1; L2; and L-L03.

2. Tof Og Tgrs to Adder - This line turns off the following outgate triggers: all "B" and "X" outgate triggers; D-Bfr; Temp to Adder A; Temp to Adder B; WR to Adder A; WR to Adder B.

3. Gate B1 (B2) to Adder - This line turns on the B outgate trigger which is specified by a decode of the B-field outgated from the Instruction Register.

4. Gate D1 (D2) to Adder - This line turns on the D-Bfr outgate trigger and ingates to the D-Bfr the D-Field which is outgated from the Instruction Register.

5. Gate B1 (B2) and D1 (D2) to Adder - This line combines the functions of the lines described in items 3 and 4.

6. Lf to Rt Olap - This signal is used in the NOXCM sequence to indicate an overlapping condition between the sink and source operand fields which requires special handling by the VFLEU. This line is conditioned by: 0 < [Sink Starting Address (to byte level) - Source Starting Address (to byte level)] < 8; i.e., 0 < [(B1 + D1) - (B2 + D2)] < 8.

7. MVO Olap - This signal is used in the PUMO sequence to indicate an overlapping condition between the sink and source operand fields which requires special handling by the VFLEU. This line is conditioned by: 0 < D < 8, where D = [(B2 + D2 + L2) - (B1 + D1 + L1)] = (Source Starting Address - Sink Starting Address).

8. Possible Olap in PACK or UNPK - This signal is used in the PUMO sequence to indicate an overlapping condition between the sink and source operand fields which may require special handling by the VFLEU. (See also item 9.) This line is conditioned by the following logical relationship: (Decode UNPK) (-8 ≤ D ≤ 0) or (0 < D < 16). (D is defined in item 7.) This condition will prompt the PUMO sequence to go into a special "hand-in-hand" relationship with the VFLEU.

9. PACK or UNPK Olap - This signal is used in the PUMO sequence to indicate that a particular source doubleword has the same address as a particular sink doubleword and that special handling of this pair of operands is required in the VFLEU. This line is conditioned when: (-8 ≤ D < 8) (Sink Dbl Wd Adr Bit, Bit 28) = (Source Dbl Wd Adr Bit, Bit 28); i.e., Sink Double Word Address = Source Double Wd Address. (D is defined in item 7.)

10. LO3, B, D, L Carry - This signal is used as operand fetching is begun, to aid in determining how many doublewords are involved in a particular operand stream. A three-bit sum is determined by the addition of the low-order three bits of the register specified by B to the low-order three bits of the D field. This sum is then added to the low-order three bits of the L-field; a carry out of the high-order position of this three-bit add is known as an "LO3, B, D, L Carry".

11. Block Og of Byte Adr from WR to Adder A - When this line is conditioned, it suppresses the outgating of bits 29-31 from the WR which would otherwise take place when the WR to Adder A Og Tgr is on.

12. MALS Data Not Needed - This line signals the MSCE that the data for a fetch are not needed.

13. Arg Byte Accp - This signal is generated by the MOP sequence for TR and TRT. Its presence indicates to the VFLEU that any argument byte which is presently on the Byte Bus may be taken off. Note that this signal consists of two lines to the VFLEU - the signal is present only when both lines are negative.

14. Blk End Cancel - This line is conditioned by the MOP logic whenever the Term Tgr is on. It is used in connection with the op cancelling process. When the line is conditioned, it prevents the VFLEU from completing its cancelling process until the MOP sequence can guarantee that no more ops will be issued to the FXOS.

15. Byte Bus Valid - This signal from the VFLEU indicates to the TR or TRT sequence that a valid argument byte is on the Byte Bus, or to the ED or EDMK sequence that a byte count is on the Byte Bus.

16. CLC Complete - This signal is generated by the VFLEU when it finds that the operands in a CLC instruction are equal. The signal is used in the NOXCM sequence as part of the control for the Blk CLC Tgr.

17. CLC Term - This signal is generated by the VFLEU based on the fact that a pair of CLC operand bytes are unequal. If the unequal byte pair is compared during cycle n, this signal is transmitted to the I-Box during cycle (n+1). CLC Term is used in the NOXCM sequence to prevent further operand accessing and as part of the control for the Blk CLC Tgr.

18. EDMK Byte Adr Accp - This signal is generated by the MOP sequence for ED and EDMK. Its presence indicates that the byte count has been gated into the Byte Buffer.

19. EDMK Complete - This signal is generated by the VFLEU and is transmitted to the I-Box in parallel with the transmission of the last pattern word to an SDB. It causes the ED or EDMK sequence to turn on the Term Tgr which will then lead to an exit from the sequence. This communication relative to the completion of EDMK execution is necessary since the I-Box must stand-by until there is no further chance that a byte address may have to be generated and stored in GPR1.

20. End of Olap PACK or UNPK - This signal is generated by the VFLEU in connection with the special approach used for particular overlap situations in PACK and UNPK. It is transmitted to the I-Box in parallel with the transmission of the last sink word to an SDB and is used in the PUMO sequence to motivate an exit.

21. End of Sr Wd - This signal, generated by the VFLEU, is used only in certain overlapped PACK or UNPK situations. It is transmitted to the I-Box during the cycle following the outgate of the last byte from the source word and is used in the PUMO sequence to initiate the fetch for the next source word and the setting up of another store for the present sink word.

22. End of Sk Wd - This signal, generated by the VFLEU, is used only in certain overlapped PACK or UNPK situations. It is transmitted to the I-Box during the cycle following the ingate of the last byte into the sink word, and is used in the PUMO sequence to initiate a store for the next sink word.

23. GPR Accept - This signal is the output of the GPR Accept Trigger which is a part of the GPR ingate priority scheme in the FXA. If the GPR Accept Tgr is turned on at the start of cycle n, data from the I-Box are gated into an addressed GPR at the start of cycle (n+2).

24. Ig Adr to GPR - This line is conditioned by the MOP logic whenever the MOP Tgr is on to block the ingate to the addressed GPR of the high-order eight bits from the I-Box. This covers the byte-address-saving situations which arise in TRT and EDMK, and requires the high-order eight bits of GPR1 to remain unchanged.

25. Last Arg Byte - This signal accompanies the last argument byte to be sent from the VFLEU to the I-Box in the course of a TR or TRT instruction.

26. TRT Complete - This signal is generated by the VFLEU when there is no non-zero function byte in the course of a TRT instruction. It is transmitted to the I-Box during the cycle following the examination of the last function byte and causes an exit from the TR or TRT sequence.

27. TRT Term - This signal is generated by the VFLEU based on the fact that a non-zero function byte has been found. It is transmitted to the I-Box during the cycle following the examination of the non-zero function byte, and in parallel with the transmission to the I-Box of the byte count necessary for the generation of the culprit argument byte address. TRT Term is recorded by the Term Tgr, and motivates argument byte address generation and storing into GPR1. It then leads to an exit from the TR or TRT sequence.

28. VFLEU - This signal is sent to the I-Box from the FXA during the FXOS decode cycle for an SS instruction. Its presence will cause the L-Reg Full Tgr to be turned off.

29. VFL Req ED Sr - This signal is sent to the I-Box by the VFLEU whenever, in the course of an ED or EDMK, it makes a request for a source word from an FLB. VFL Req ED Sr is recorded by MOP in the Req ED Sr Tgr and motivates the fetch of the next source word.

30. VFL Req Ptrn or Arg Tgr - This signal is generated by the VFLEU in connection with pattern word requests in ED or EDMK and argument word requests in TR or TRT. The signal is recorded by Req Ptrn or Arg Tgr. The TR or TRT sequence is notified for all argument word requests except the first; and the Req Ptrn or Arg Tgr initiates the fetch of the next argument word. The ED or EDMK sequence receives this signal for all pattern word requests except the first (and in one case the end of the pattern word from the "END" Op); and the Req Ptrn or Arg Tgr initiates the fetch of the next pattern word.

31. Ton SAR Valid Tgr - Associated with each of the three SARs is a valid trigger. A SAR is set "Valid" when a new address is set into the SAR. Once the SAR is valid, its address is compared with all incoming addresses. Situations arise in the multiple op sequences for handling TR and PACK or UNPK in which an address may be set into a SAR before it is desirable to subject it to comparisons. In such cases it becomes necessary to modify the general procedure for setting a SAR valid. The logic used to accomplish this modification makes use of the signals: Suppress Ton SAR Valid Tgr, and Ton SAR Valid Tgr per Op Stage SAR Field. The flow chart below shows how these special signals interact with the general procedure for setting a SAR valid.

DIAGRAM 5-35. MOP DEFINITIONS

Instruction Fetch Control Triggers

Array Fetch Protect Tags (Array Words 0-7, Temp 1 and 2) - NN161-167, 181-187, 207
Indicate when an instruction fetch has violated a protected block of storage. There are ten array protect tags corresponding to the 8 array doublewords and the two temp locations. Should an attempt be made to decode any portion of an array or a temp doubleword with a protect tag on, a fetch protect interrupt results. The tags are turned on or off respectively, as a storage return which violates or satisfies fetch protect is gated to the associated array word.

Array Fetch Tags (Array Words 0-7, Temp 1 and 2) - NN161-167, 181-187, 201
Indicate when a fetch for the corresponding doubleword of instruction array has been made and the data has not yet returned from storage. There are ten fetch tags, corresponding to the 8 array doublewords and the two temp locations. The tags are turned on by the instruction fetch mechanism and turned off on the same cycle that the data from the memory bus is ingated.

Initial Fetch - NN177
Used to gate a zero into the UBCTR, allowing an unmodified transfer of the UB register contents to the UBCTR output. This requirement is necessary because instruction fetching is initiated via the UB register even though the next instruction fetch address is always taken after UB is updated by the UBCTR output. The initial fetch trigger is used to gate a zero into the UBCTR. This allows an unmodified transfer of the UB register contents through the UBCTR and back to UB.

Instruction Fetch Trigger (IFT) - NM115
Master control trigger for instruction fetch logic. This trigger must be on for any fetches that take place using the UB register.

Instruction From Memory Request Trigger (IMRT) - NN225
Signals MSCE of an instruction fetch. This signal also inhibits operand fetching to give instruction fetching CPU priority to the address bus to storage. Initiating an instruction fetch turns IMRT on; MSCE accepting to the CPU turns IMRT off.

ISR Protect Tag - NN249
This trigger is turned on or off by the MSCE to indicate that the immediately following storage return to the I-Box has violated protected storage. It is used to turn on or off the appropriate array or execute protect tags.

Store Interlock Trigger (SIT) - NN230
Set on when it is detected that a store instruction in the ppln 3 op register has stored within the UB and LB boundaries. SIT effectively cancels the affected instruction array word by initiating a fetch to storage for the doubleword stored into. The trigger is turned off when the re-fetch has been initiated.

Store Into Array Trigger (SIAT) - NN231
On SS format store operation or a store multiple doubleword, if the store falls within UB and LB, multiple instruction array words may become invalid. SIAT simplifies the recovery control by initializing the whole array starting with the current instruction address. SIAT is turned on by the detection of a store between the UB and LB bounds when processing an SS store or a store multiple instruction. It goes off following the completion of the instruction and completion of the reinitialization.

Temp Fetch Trigger - NN226
This trigger is turned on when the instruction fetch area is initialized following a branch that did temp fetching. The trigger causes the updating of the UB without initiating storage fetches, thereby preventing re-fetching of branch targets already fetched to the temp 1 and temp 2 locations. The UBCTR output is then correctly positioned to fetch the instruction word after the doubleword fetched for temp 2. The temp fetch trigger goes off on completion of the UB update.

Temp Fetch 1 and 2 - NN205
Signify that the respective instruction array doubleword (temp 1 and temp 2) has been activated by a branch decision and is to be gated to the op register and also to the instruction array word designated by the AOC register. The triggers are turned on as a result of a successful branch that did temp fetching.

Pipeline Stage 1 (Instruction Decode) Control Triggers

Fixed Buffer Busy Counter - NP139
A three-bit control register that properly sequences the releasing of fixed operand buffers when an instruction requiring more than one of these buffers is encountered. These instructions require the fetching of 64 bit operands and therefore utilize the floating buffers as well as the fixed buffers. The fixed buffer busy counter is required when these instructions are stacked.

Fixed Buffer Busy Tags A-F (FXB-BSY) - NP343-353
A set of six triggers, one for each 32 bit fixed storage operand buffer, utilized to control allocation of buffers to storage fetches initiated for fixed operations. A buffer is made busy by turning its tag on as a fetch to the buffer is initiated. The tag is turned off when the fixed area utilizes the buffer contents, whereupon the buffer becomes available for re-assignment.

Fixed Operation Stack Count (FXOS) - NP131
A three-bit control register interlock on issuing to the fixed area. As an instruction is issued to the fixed area, the FXOS counter is incremented; when an instruction is completed by fixed execution, the counter is decremented. A maximum of six operations can be outstanding prior to the interlock preventing more issuing.

Floating Buffer Busy Tags 1-6 (FLB-BSY) - NP357-361
A set of six triggers, one for each 64 bit floating storage operand buffer, identical in function to the fixed buffer busy tags, except applying to the floating buffers.

Floating Operation Stack Count (FLOS) - NP135
A four-bit control register identical in function to the FXOS, except applying to the floating area. A maximum of eight operations can be outstanding prior to the interlock preventing more issuing.

New Instruction Address Trigger (NIAT) - NN227
Sets the op register outgate triggers according to the value in IR after a new instruction address has initialized the array counters and IR.

Op Register Outgate Triggers (2 per halfword) - NP009-015
Used to control the outgating of the op register.

Op Register Valid Triggers (one per halfword) - NN244-247
Used to indicate which of the four halfwords of the op register are valid and available for decoding. A particular valid trigger is turned on by ingating new data to its associated op register halfword and turned off when the instruction processing has utilized that halfword.

Register Unavailable For Addressing (RUA Tags) - NP145-285 Order is: UA0, 1; RUM 0, 1; RUA 2, 3; RUM 2, 3; etc.
A set of sixteen (one for each GPR) 3-bit control tags used to guarantee logically correct address generation by the I-Box. As an operation is issued to the fixed area, any GPR altered by that operation (R1) has its RUA tag incremented. The fixed area will decrement the tag upon execution of the operation. A nonzero tag indicates to any subsequent decode that the associated GPR is unavailable as an address generating parameter. Three bits are required to cover the number of operations that can be outstanding in the fixed area.

Register Unavailable For Modification (RUM Tags) - NP145-285 Order is: RUA 0, 1; RUM 0, 1; RUA 2, 3; RUM 2, 3; etc.
A set of sixteen (one for each GPR) 3-bit control tags used in conjunction with the RUA tags to guarantee the proper sequencing of I-Box results into the GPRs. As an operation is issued to the fixed area, any GPR required as a source parameter by that operation (R2) has its RUM tag incremented. The fixed area decrements the tag upon execution of the operation. A nonzero RUM or RUA indicates to any subsequent decode that the I-Box cannot change this register, as an outstanding operation requires the current contents. Three bits are required to cover the number of possible outstanding fixed area operations.

R1 Double Sink Trigger - NM117
Set by the decoder during all MOP instructions. It is meaningful only for LM and STM, double length shifts, and multiply/divide. It is used to make the R1 equal SLCX and R1 equal SLCB comparator compare not only R1 but R1 plus one, if R1 is even.

State Triggers (3) - NM083
General sequence control triggers. S0 is the decode state of all instructions. S7 is entered at the start of a multi-op instruction. MOP logic uses its own state triggers. The interrupt sequences are executed in S0 and have their own sequence triggers.

Store Address Register Busy Tags (SAR-BSY) - NP369

A set of three triggers, one for each store address buffer register. As a store is issued, one of the buffers is assigned to hold its address until executed. The buffer is freed following execution and becomes available for reassignment.

SVR1 Invalid Trigger (R1 Invalid) - NM113
Indicates, for one cycle, to the decoder that the GPR addressed by SVR1 has just been changed by the I-Box. If the address generate following this decode cycle used this GPR, the SVR1 invalid trigger substitutes the WR contents (WR contains the address parameter) for the GPR (which has not yet been updated).

Pipeline Stage 2 - Adder and Execution Staging Control Triggers

B Outgate Triggers - NP435-441
Sixteen triggers, one for each value of B. Allows gating the associated 32-bit GPR quantity to the A adder input 0-31.

D Outgate Trigger - NP473
Enables gating the 12-bit D buffer quantity to the D adder input 0-11.

Fixed Valid Trigger - NP385
Notifies the fixed area that the staging register contains an instruction for the fixed area. The trigger is turned on when an instruction is set into the stage and remains on for one cycle.

Floating Valid Trigger - NP385
Notifies the floating area that the staging register contains a floating point instruction. The trigger is turned on when an instruction is set into the stage and remains on for one cycle.

Hot Ones Triggers - NP467-481
Four triggers that allow gating particular single bit entries to the adder. The entry positions are into the D adder input.

1. Bit 31
2. Bit 29
3. Bit 28
4. Bit 25

IR Counter Outgate - NP471
Gates the five IR counter bits (26-30) to D adder input positions 26-30.

LB Outgate - NP471
Gates LB 8-25 to A adder input positions 8-25.

Set Condition Code Trigger - NP385
Notifies the execution areas that a condition code affecting instruction has just been issued or handled by the I-Box. The signal inhibits any outstanding instructions from changing the code. It is a one-cycle signal.

SVIR Outgate Trigger - NP479
Gates the five SVIR bits to the A adder input positions 26-30.

Temp Outgate Triggers - NP477
Two triggers allowing the 24-bit temp quantity to be gated to either the A or B adder inputs 8-31.

Timer Address Trigger (TAT) - NP483
Forces the address of the timer storage location into the adder when a timer update is required. Feeds hot inputs to D input positions 25 and 27.

True/Complement Trigger - NP467
Enables complementing the data gated to the B adder input 0-31.

WR Outgate Triggers - NP469
Two triggers allowing the 32-bit WR quantity to be gated to either the A or B adder inputs 0-31.

X Outgate Triggers - NP447-457
Sixteen triggers, one for each value of X. Allows gating the associated 32-bit GPR quantity to the B adder input 0-31.

Pipeline Stage 3 - Interface with Storage Control Triggers

MSCE Not Accepting - NL551
Indicates that no CPU address will be taken on the following cycle. Controlled by the MSCE.

Branch Control Triggers

Activate Trigger - NM091
Set for one cycle when test of CC shows branch condition guess is correct. It notifies both stacks to start processing conditional ops, i.e., render them normal.

Back Less Than Eight Trigger (BACK8T) - NM103
Indicates that the target of the current branch is back less than 8 doublewords. It can be set in two ways: (1) When LB $\leq$ WR $\leq$ UB and WR $\leq$ IR or (2) When the adder output indicates back less than 8 (LB plus IRCTR minus the target).

BAL Trigger - NM111
Indicates a BAL op has entered state 6.

BCQ Trigger - NM119
Set during the decode of a quick loop BCT instruction (loop trigger on and IR equals SLCIR).

Branch In Array Trigger (BIAT) - NM107
Indicates the target of a branch (not BC) is in the array. It is set by the LB ≤ WR ≤ UB comparator after the first temp fetch is made.

Branch In Array 1 Trigger (BIA1T) - NM107
Similar to BIAT but set when a temp fetch sequence has not been used.

Branch Trigger (BRT) - NM107
Indicates whether the branch in process will be taken. It is set according to the adder output and the op (BXLE, BXH, BCT).

BXLE Trigger - NM103
Turned on during the decode of a BXLE instruction, off by the decode of BXH. It tells the branch sequences how to interpret the adder output, i.e., whether a plus or D sum indicates branch or whether a minus sum indicates branch.

DIAGRAM 5-36. I-UNIT TRIGGER LIST (SHEET 1 OF 3)

BXQ Trigger - NM118

Set during the decode of a quick loop BX instruction (loop trigger on and IR equals SLCIR). It is also used for BX ops in loop (but not quick) mode.

Cancel GPR Request Trigger - NM109

A store in the instruction stream may occur after the request GPR priority trigger has been turned on. The cancel GPR request trigger allows the request GPR priority trigger to be turned off without ingating the GPRs.

Cancel Trigger - NM091

Set for one cycle when test of CC shows BC guess is wrong. It notifies both stacks, FXOS and FLOS, and MSCE to destroy all outstanding conditional ops.

CC Valid - NM095

Tells test sequence when CC is available for testing. It is a data trigger and is kept off by (1) any CC op outstanding in the FXA or FLA, (2) a CC bit on in the op stage and not conditional mode, and (3) by the turn on of the activate trigger for one cycle (set to on again if a CC bit was issued in conditional mode).

Conditional BACK8T - NM093

On during conditional mode caused by a BC operation whose target is back less than 8 doublewords.

Conditional BIAT - NM093

On during conditional mode caused by a BC whose target is in the instruction array.

Conditional Quick Trigger - NM089

On during conditional mode caused by a BC closing a loop in the instruction array.

Conditional Trigger - NM095

Indicates machine is in conditional mode, i.e., the I-Box has processed a BC op but the test of the CC has not yet been made.

Loop Trigger - NM087

Indicates machine in loop mode. May be reset by decode of any fixed op that changes the GPR pointed to by SLCB or SLCX.

R1 Outgate Inhibit Trigger - NP019

Inhibits the op register R1 fields from being outgated to the R1 decoder when a substitute field is utilizing the decode hardware.

Request GPR Priority Trigger - NM117

Requests the FXA to give the I-Box priority on ingating into the GPRs. The turn off of this trigger causes the WR to be set into the GPR addressed by the SVR1 register.

SVR1 Outgate Trigger - NM109

Outgates SVR1 to the R1 OR. It is set and reset by the XEC and BAL sequences.

Temp Fetch Made Trigger (TFMT) - NM089

Indicates that two temp fetches have been made for the branch in process. (Target is out of instruction array.)

Unconditional Branch Trigger (BCUNCONT) - NM118

Set during the decode of a BC with an all ones mask field to distinguish it from a conditional branch.

Execute Control Triggers

Execute A Trigger - NM152

Indicates that an execute instruction is being processed. Required because the op register will be changed by the target fetch during the processing. This trigger is reset after target returns from MSCE.

Execute B Trigger - NM103

Indicates that the op in process is the target of an XEC instruction.

Execute C Trigger - NN249

Controls housekeeping following the completion of an execute target that has not resulted in a branch being taken. It controls the adjusting of the AOC and IR register contents to point to the instruction subsequent to the execute instruction, and activates instruction fetching. It is turned on for one cycle as a result of the completion of the target.

Execute Fetch Protect Tags - NN209

Two execute fetch protect tags are turned on as a result of fetching the execute target from a protected array. An interrupt will result from an attempt to use any portion of a protected word. Turn on and off is identical to that of the array protect tags.

Execute 1 and 2 Fetch Tags - NN203

Indicate outstanding fetches for the execute target. When the data arrives, it will be routed only to the op register and not to any array location. Two fetches may be required because the XEC target may cross a doubleword boundary.

Execute 1 Invalid Trigger - NM152

Turned on when the fetch for an execute target is to an invalid address.

Execute 2 Invalid - NM155

Indicates the second fetch for execute target is invalid. This fetch is made because the target instruction may cross a doubleword boundary. No action results from this indicator unless the fetch is utilized.

Plus 2 To IRCTR Trigger - NN197

Used to set the IRCTR to the correct value after an execute has been processed. The IR must be updated by two halfwords following the target completion irrespective of the target length. It is turned on for one cycle following the completion of the execute target.

Interrupt Control Triggers

Channel Release Trigger - NM383

Captures the release signal from the channels to provide a timing base on which to terminate I/O instructions and I/O interrupts. Trigger is off whenever there is no turn-on signal.

External Identity Triggers

Set of eight triggers used to indicate which of the external, interrupt key, and timer interrupts are currently being processed.

Inhibit WR Address Trigger - NM551

Blocks the output of the WR from the address bus to MSCE during interrupt sequencing. This action allows the PSW addresses to be forced via the same bus.

Interrupt Fetch/Store Trigger - NM153

Indicates and causes the PSW fetch or store address to be formed based on the interrupt class being sequenced.

Interrupt Sequence B Trigger - NM160

Indicates a precise interrupt is currently being processed. Enables imprecise interrupts that occur during the process to usurp the control and cancel the precise interrupt.

Interrupt Sequence 1 and 2 Triggers - NM151

Provide clocking for handling interrupt sequencing.

I/O Interrupt Taken Trigger - NM158

Signals the channels that an I/O interrupt has been honored and that status may be stored.

I/O Interrupt Triggers - NM391-397

Set of seven triggers, one for each channel, turned on when the associated channel signals interrupt.

Pipeline Not Empty Trigger - NM158

Used whenever logical correctness demands a pipeline drain. The requirements for pipeline empty are:

1. FXOS and FLOS empty
2. I-Box state zero
3. All FXBs, FLBs, and SARs available
4. Not step IR
5. FLRs not busy
6. No FXA execution in process
7. Instruction fetch trigger on
8. All SDBs empty
9. Ppln empty from PSCE
   a. No CPE request in CPE buffer
   b. No active CPU request in any queue register.
10. CC valid

Program A Interrupt Triggers

Set of triggers identifying the various imprecise interrupts. In order of priority the imprecise interrupts are:

1. Data - NM417
2. Fixed Overflow - NM417
3. Divide - NM407
4. Exponent Overflow - NM407
5. Exponent Underflow - NM409
6. Significance - NM409
7. Floating Divide - NM411
8. Protect - NM411
9. Specification A - NM413
10. Address A - NM413

Two additional triggers indicate imprecise interrupts on conditionally issued instructions. These triggers turn on the normal program A interrupt triggers if the conditional instructions are activated.

1. Conditional Specification A - NM415
2. Conditional Address A - NM415

Program B Interrupt Triggers

Set of triggers identifying the various program exceptions and the supervisor call exception, all of which are precise. In order of priority:

1. Operation - NM385
2. Specification B - NM385
3. Privileged Op - NM387
4. Execute - NM387
5. Address B - NM389
6. Supervisor Call - NM389

Program B Interrupt Trigger - NM154

Indicates a precise interrupt condition has been signaled. Establishes correct timing platform on which to begin sequencing the interrupt.

PSW Stored Trigger - NM157

Indicates that the PSW has been stored to prevent a second store for a single interrupt.

Multiple Operation (MOP) and Timer Control Triggers

Block CLC Trigger (BLK CLCT) - NM259

Used to block the MOP sequence for CLC (the NOXCM sequence) in case a prior CLC has not yet been terminated or completed. It is turned on when a MOP sequence completes a CLC prior to its completion or termination by the VFLEU. It is turned off when the VFLEU signals completion or termination.

Byte Buffer Outgate Trigger - NM319

Outgates the 8 bits of the byte buffer to input B, bits 24-31 of the address adder. This outgate function is used in the TR or TRT, ED, EDMK and PUMO sequences.

Byte Buffer Valid Trigger - NM301

Turned on whenever the byte bus is gated into the byte buffer and the byte bus valid line from the VFLEU is conditioned. This trigger goes off whenever the data in the byte buffer has been used and no new data has been gated in. TR, TRT (in TR or TRT sequence) and EDMK (in ED, EDMK sequence) instructions make use of this trigger.

DWCR To Adder Outgate Trigger - NM319

Outgates the 5 bits of the doubleword count register to input B, bits 24-28 of the address adder. This outgate function is used in the NOXCM and ED, EDMK sequences.

Inhibit Advance to Working Register Trigger - NM248

Prevents ingating the address adder sum to the working register (or temp). This trigger is turned on whenever an address is ingated to the WR but the associated ppln op bits are not set into the ppln 3 register. This condition occurs during operand accessing whenever advance to WR is present but advance is not. The trigger is turned off as the ppln op bits are set into the ppln 3 op register.

Last or SBA Carry Trigger (LAST) - NM247

Used in three contexts. It is used in the TR or TRT sequence to record the last argument byte signal which accompanies the last byte of the argument stream from the VFLEU to the I-Box. It is used in the ED, EDMK sequence to aid in sequencing and addressing the last pattern word fetch. It is used in the PUMO sequence to record the occurrence of a LO3 B, D, L, carry to set the one more source trigger or the one more sink trigger correctly.

L1 Outgate - NM319

Outgates the L1 field from the L register (bits 0-3) to input B, bits 28-31 of the address adder, and also gates the low-order 3 bits of the L1 field to the LO3 carry propagate logic (see LO3, B, D, L, carry in the MOP definitions - Diagram 5-35, item 10). This outgate function is used in the PUMO sequence (covering all MOP instructions which have two 4-bit L fields, L1 and L2).

L2 Outgate - NM319

Outgates the L2 field from the L register (bits 4-7) to input B, bits 28-31 of the address adder and also gates the low-order 3 bits of the L2 field to the LO3 carry propagate logic (see LO3, B, D, L, carry in the MOP definitions - Diagram 5-35, item 10). This outgate function is used in the PUMO sequence (covering all MOP instructions which have two 4-bit L fields, L1 and L2).

L-LO3 Outgate - NM201

Outgates the low-order 3 bits of the L register to the carry propagate logic (see LO3, B, D, L, carry in the MOP definitions - Diagram 5-35, item 10). This outgate function is used in the TR or TRT, NOXCM and ED, EDMK sequences (covering all MOP instructions which have a single 8-bit L field).

L Register Full Trigger - NM245

Turned on when the L (or L1-L2) field is set into the L register at the beginning of each MOP sequence except LM or STM. While on, it serves as an interlock to prevent the I-Box from beginning a subsequent MOP instruction. It is turned off on the cycle following the FXOS decode of the first op of the associated MOP instruction.

Mark Trigger - NM243

Provides an assortment of special functions in all MOP sequences except LM or STM.

Multiple Op State Triggers - NM305

Define the 3 major states used in the MOP (multiple op) sequences. There are five MOP sequences: LM or STM; TR or TRT; NOXCM which covers NC, OC, XC, CLC, MVC, MVZ and MVN; ED, EDMK; and PUMO which covers PACK, UNPK, and MVO. Each of the sequences is entered in MOP state 1 and is completed in MOP state 3.

DIAGRAM 5-36. I-UNIT TRIGGER LIST (SHEET 2 OF 3)

1. MOP State 1 - This trigger defines state 1 for each MOP sequence.

2. MOP State 2 - This trigger defines state 2 for all MOP sequences. (Note that the LM or STM sequence does not use state 2.)

3. MOP State 3 - This trigger defines state 3 for all MOP sequences.

Multiple Op Trigger - NM239
Turned on as each MOP sequence is entered and is turned off as each MOP sequence is completed. While on, it: aids in control of ingating to the op stage and ppln 3 op; modifies the handling of the store into instruction array situation; and calls for partial (24-bit address field only) GPR ingating whenever required.

New Argument Binary Trigger (NABT) - NM260
Used in TRT to aid in the cancellation of an unused argument buffer in case the TRT instruction terminates rather than completes. The NABT is off as the I-Box TRT sequence initiates the first argument fetch and is then alternately turned on and off as succeeding argument doubleword fetches are initiated. When NABT is on, it forces a bit to op stage position 13 (in the POSE field).

One More Sink Trigger - NM253
Used in all MOP sequences except LM or STM as an aid in counting the doublewords involved in the sink operand field.

One More Source Trigger - NM251
Used in NOXCM and PUMO sequences as an aid in counting the doublewords involved in the source operand field. This trigger is also used in the TR or TRT sequence as described on the TR and TRT flowchart (Diagram 5-126).

Operand Access Trigger - NM241
Used in all MOP sequences except LM or STM. It derives its name from its use in the NOXCM and ED or EDMK sequences wherein it is used (in conjunction with one more source trigger and one more sink trigger) as a part of the operand doubleword fetching mechanism. In this context, the operand access trigger will be on as fetching is begun and will be turned off when operand accessing is almost complete. In the TR or TRT and PUMO sequences, this trigger provides general sequencing functions.

Overlap Trigger (OLAPT) - NM257
Used as an aid to handle certain overlapping operand situations that may be discovered during the NOXCM and PUMO sequences. It is turned on whenever the operand fields overlap in such a way that special handling will be (or may be, in case of PACK and UNPK) necessary in the VFLEU.

Request Edit Source Trigger - NM301
Used in the ED, EDMK sequence to record that the VFLEU has requested a source word from an FLB. When on, it causes the ED, EDMK sequence to fetch the next source doubleword.

Request Pattern or Argument Trigger - NM301
Used in the TR or TRT sequence to record that the VFLEU has requested an argument word from an FLB. When on, it causes a fetch for the next argument word. This trigger is used in the ED, EDMK sequence to record that the VFLEU has requested a pattern word from an FLB; and then it causes the fetching of the next pattern doubleword, and the issuance of an associated op to the FXOS.

R1 to R1 OR Outgate Trigger - NM201
Used only in the LM or STM sequence. Gates the R1 field from the L register (bits 0-3) to the R1 OR. The decoded output of this 4-bit OR is used to step the appropriate RUA counter(s) when stepping is signaled by the LM or STM sequence.

Terminate Trigger (TERMT) - NM255
Used to record a termination (not to be confused with the architecturally defined termination) signal from the VFLEU. Termination of a TRT or a CLC instruction prevents further operand accessing. This trigger is also used to record an EDMK execution complete signal from the VFLEU in the ED, EDMK sequence; and for a special sequence function for handling PACK or UNPK overlap situations (in the PUMO sequence).

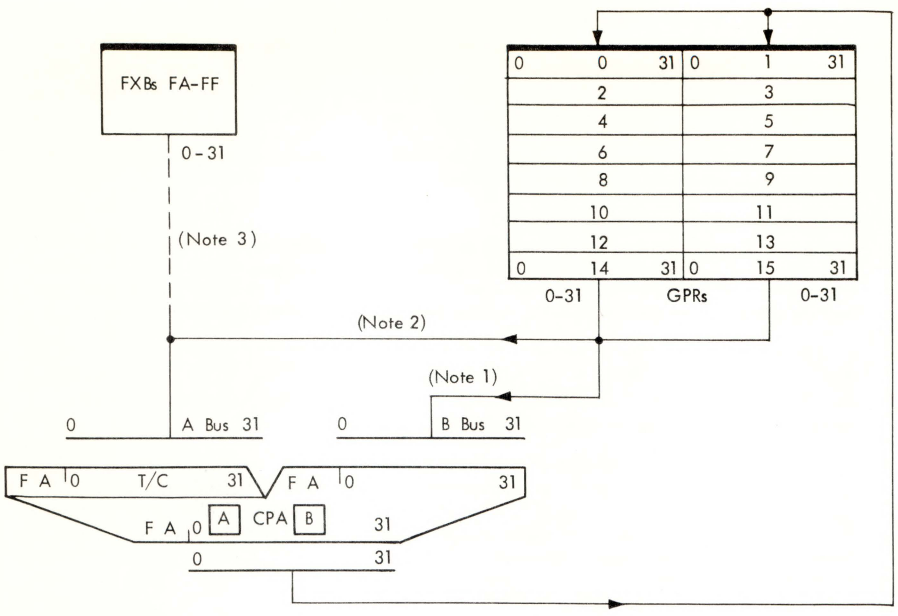
Timer Go Trigger - NM157
Turned on when basic interlocks for issuing the timer instruction are satisfied. When the trigger is on, it initiates the issuance of the instruction and the formation of the proper address.

Timer T1 and T2 Triggers - NM158
Indicates timer update signal has been received. (The two triggers guarantee only one update per signal.)

MOP Definitions and Control Lines Between MOP Sequences and Fixed Area

Refer to MOP Definitions, Diagram 5-35.

DIAGRAM 5-36. I-UNIT TRIGGER LIST (SHEET 3 OF 3)



Notes:

1. R1 Field Data (RR and RX Formats).

2. R2 Field Data (RR Format Only).

3. FXB Field Data (RX Format Only).

1A	AR	Add
1B	SR	Subtract
1E	ALR	Add Logical
1F	SLR	Subtract Logical

	0	7 8	11 12	15 16	20
RR	1A,1B,1E,1F	R1	R2	▨	

CONDITION CODE

Arithmetic Ops:

0 = Results are zero
1 = Results less than zero
2 = Results greater than zero
3 = Overflow

Logical Ops:

0 = Results are zero (no carry)
1 = Results are not zero (no carry)
2 = Results are zero (carry)
3 = Results are not zero (carry)

4A	AH	Add Halfword
4B	SH	Subtract Halfword
5A	A	Add
5B	S	Subtract
5E	AL	Add Logical
5F	SL	Subtract Logical

	0	7 8	11 12	15 16 18	19 20
RX	4A,4B,5A,5B,5E,5F	R1	▨	FXB	

1. Add and subtract operations gate data from the GPRs or storage (FXB) to the carry propagate adder.

2. Adder circuits perform all the add and subtract functions.

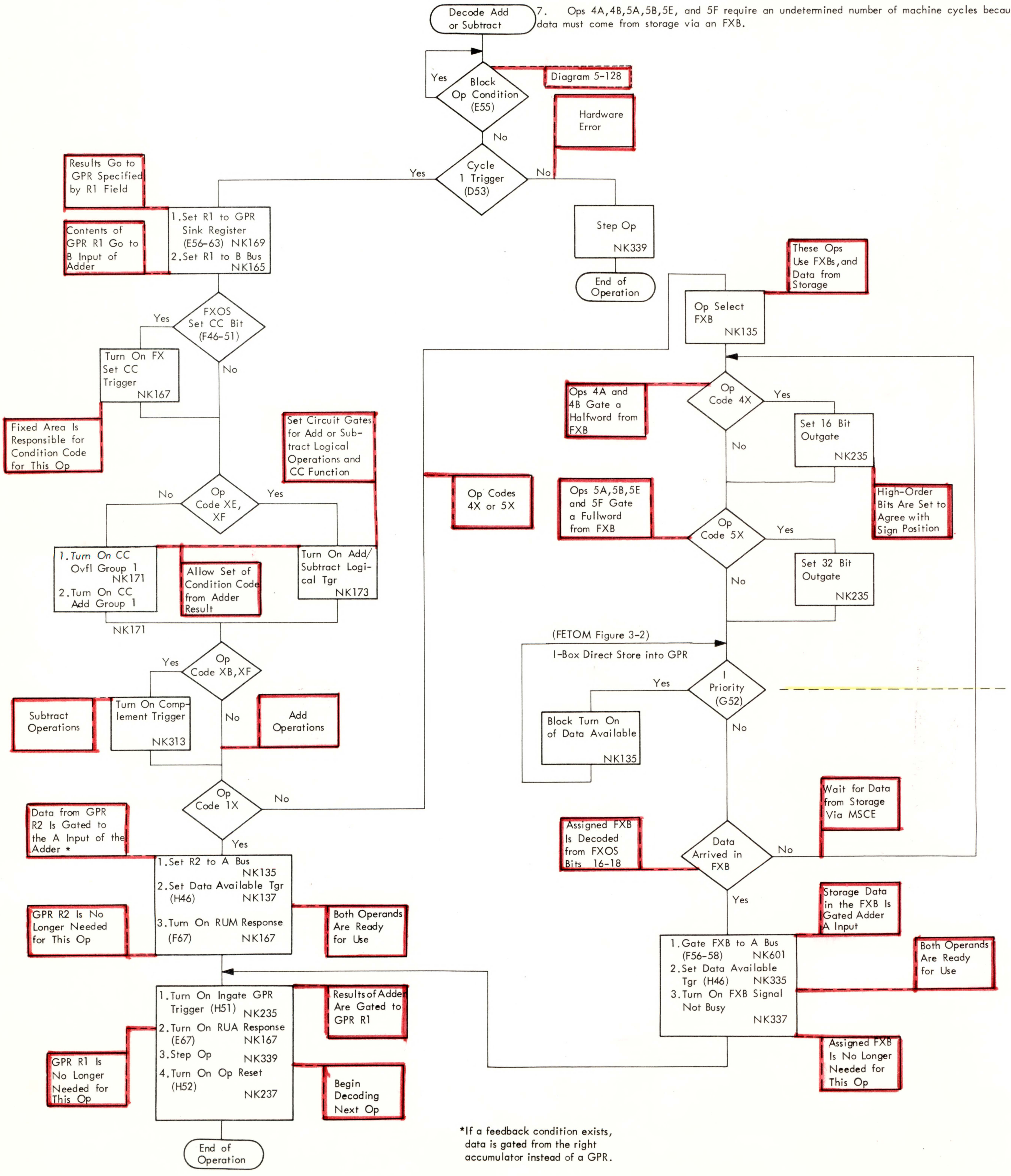
3. Results are placed back in the R1 GPR.

4. Logical operations treat the data as 32 bit values without signs.

5. In halfword operations, 16 bits of data from the FXB are gated to the adder. The high-order bits of the word are changed to agree with the halfword sign.

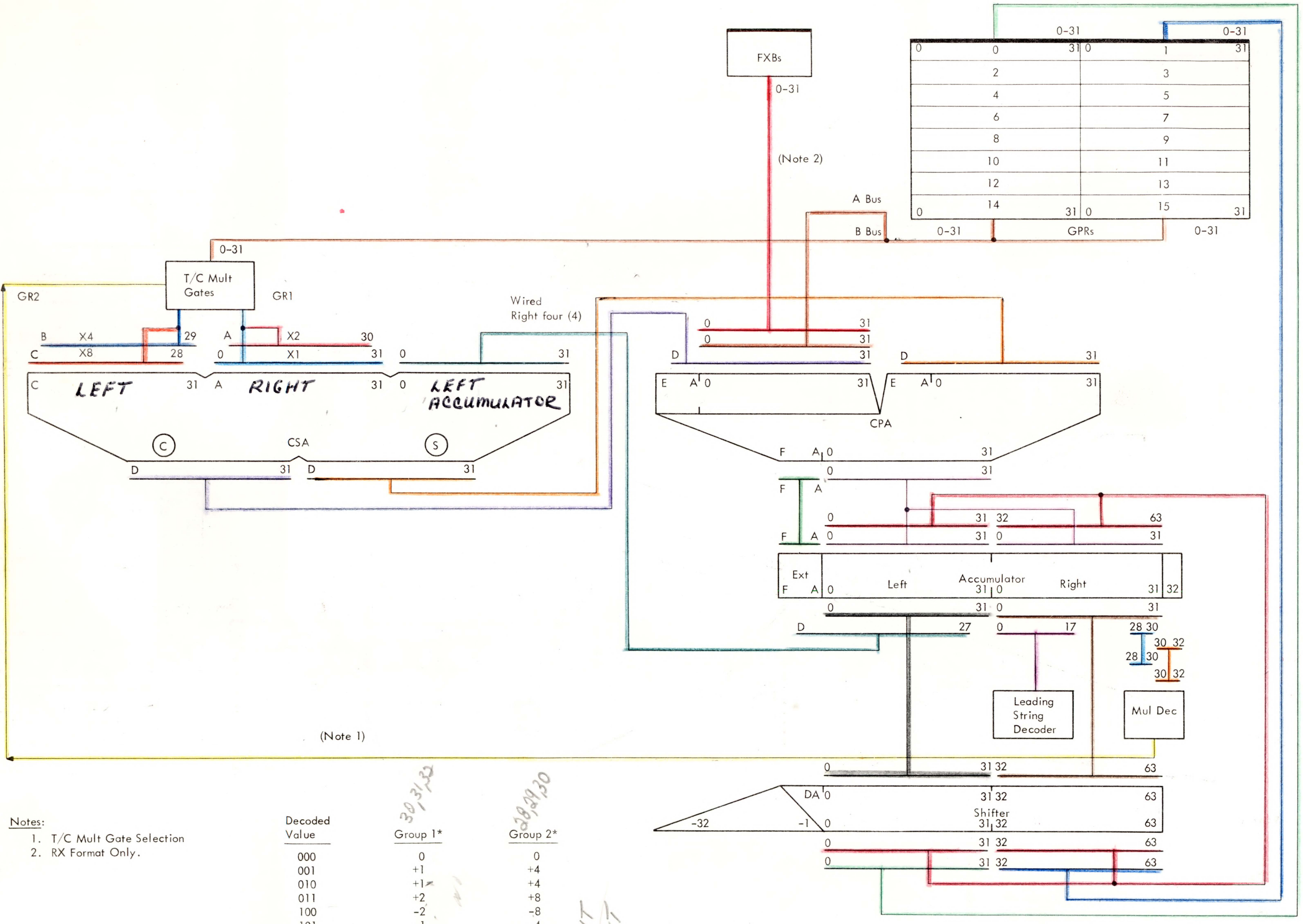
6. Normal execution time for ops 1A, 1B, 1E, and 1F is one machine cycle.

7. Ops 4A,4B,5A,5B,5E, and 5F require an undetermined number of machine cycles because input data must come from storage via an FXB.


Decode Add or Subtract
Yes
Block Op Condition (E55)
Diagram 5-128
No
Hardware Error
Results Go to GPR Specified by R1 Field
Yes
Cycle 1 Trigger (D53)
No
1. Set R1 to GPR Sink Register (E56-63) NK169
2. Set R1 to B Bus NK165
Contents of GPR R1 Go to B Input of Adder
Step Op NK339
End of Operation
These Ops Use FXBs, and Data from Storage
Op Select FXB NK135
Yes
FXOS Set CC Bit (F46-51)
Turn On FX Set CC Trigger NK167
No
Fixed Area Is Responsible for Condition Code for This Op
Set Circuit Gates for Add or Subtract Logical Operations and CC Function
Ops 4A and 4B Gate a Halfword from FXB
Op Code 4X
Yes
Set 16 Bit Outgate NK235
No
High-Order Bits Are Set to Agree with Sign Position
Op Codes 4X or 5X
Ops 5A, 5B, 5E and 5F Gate a Fullword from FXB
Op Code 5X
Yes
Set 32 Bit Outgate NK235
No
Op Code XE, XF
Yes
1. Turn On CC Ovfl Group 1 NK171
2. Turn On CC Add Group 1
Allow Set of Condition Code from Adder Result
Turn On Add/Subtract Logical Tgr NK173
NK171
(FETOM Figure 3-2)
I-Box Direct Store into GPR
Yes
I Priority (G52)
Op Code XB, XF
Subtract Operations
Turn On Complement Trigger NK313
No
Add Operations
Block Turn On of Data Available NK135
No
Op Code 1X
No
Yes
Data from GPR R2 Is Gated to the A Input of the Adder *
1. Set R2 to A Bus NK135
2. Set Data Available Tgr (H46) NK137
3. Turn On RUM Response (F67) NK167
Wait for Data from Storage Via MSCE
Assigned FXB Is Decoded from FXOS Bits 16-18
Data Arrived in FXB
No
Yes
GPR R2 Is No Longer Needed for This Op
Both Operands Are Ready for Use
Storage Data in the FXB Is Gated Adder A Input
1. Gate FXB to A Bus (F56-58) NK601
2. Set Data Available Tgr (H46) NK335
3. Turn On FXB Signal Not Busy NK337
Both Operands Are Ready for Use
1. Turn On Ingate GPR Trigger (H51) NK235
2. Turn On RUA Response (E67) NK167
3. Step Op NK339
4. Turn On Op Reset (H52) NK237
Results of Adder Are Gated to GPR R1
GPR R1 Is No Longer Needed for This Op
Begin Decoding Next Op
Assigned FXB Is No Longer Needed for This Op
End of Operation

*If a feedback condition exists, data is gated from the right accumulator instead of a GPR.

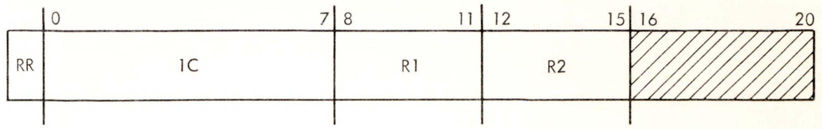
DIAGRAM 5-100. ADD AND SUBTRACT



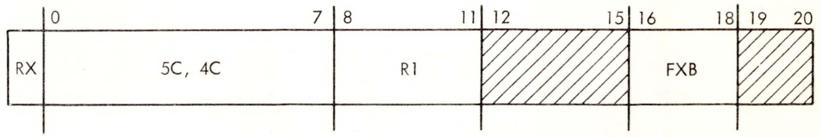
Notes:
1. T/C Mult Gate Selection
2. RX Format Only.

The condition code remains unchanged during multiply ops

1C MR MULTIPLY (RR)



5C M MULTIPLY (RX)
4C MH MULTIPLY HALFWORD (RX)



Objectives:

1. During MULTIPLY operations, information from GPR's or Storage (FXB's) is multiplied and the product placed in a GPR.
2. The multiplicand is specified by the R1 field.
3. The multiplier is specified by the R2 field for RR multiply and by the FXB field for RX multiply or multiply halfword.
4. Actual multiplication is performed by the True/Complement multiply gates at the input of the carry save adder (CSA), Each hex digit of the multiplier is decoded in turn. The decoded gates determine the multiplied value of the multiplicand entered into the partial product during each iteration.
5. T/C Mult. gates multiply by shifting the multiplicand 0, 1, 2 or 3 positions as it is gated into the CSA. By gating into the CSA in true or complement form and shifting the correct number of positions, any value of multiplication can be produced from 0 to 16 (the binary value of the multiplier hex digit).
6. The partial product is held in the left side of the accumulator and shifted right 4 positions for each iteration. The final product occupies both the left and right sides of the accumulator.
7. At the end of the multiply operation the final product is transferred to the GPR's.
8. Multiply operations can be performed in less than the normal 8 iterations (11 cycles) if the multiplier high order positions are an unbroken string of 9 or 17 zero bits or one bits.
9. During the last cycle the product is shifted the correct number of bits to position it in the sink GPR's.

Decoded Value	30,31,32 Group 1*	28,29,30 Group 2*
000	0	0
001	+1	+4
010	+1	+4
011	+2	+8
100	-2	-8
101	-1	-4
110	-1	-4
111	0	0

* True and complement form indicated by (+) and (-), respectively. Numbers indicate effective multiplications made for a particular decoded value.

T/C Multiply Gate Decoding

∅ NO INPUT
1 = NO SHIFT
2 = 1 SHIFT
4 = 2 SHIFT
8 = 3 SHIFT

DIAGRAM 5-101. MULTIPLY (SHEET 1 OF 3)

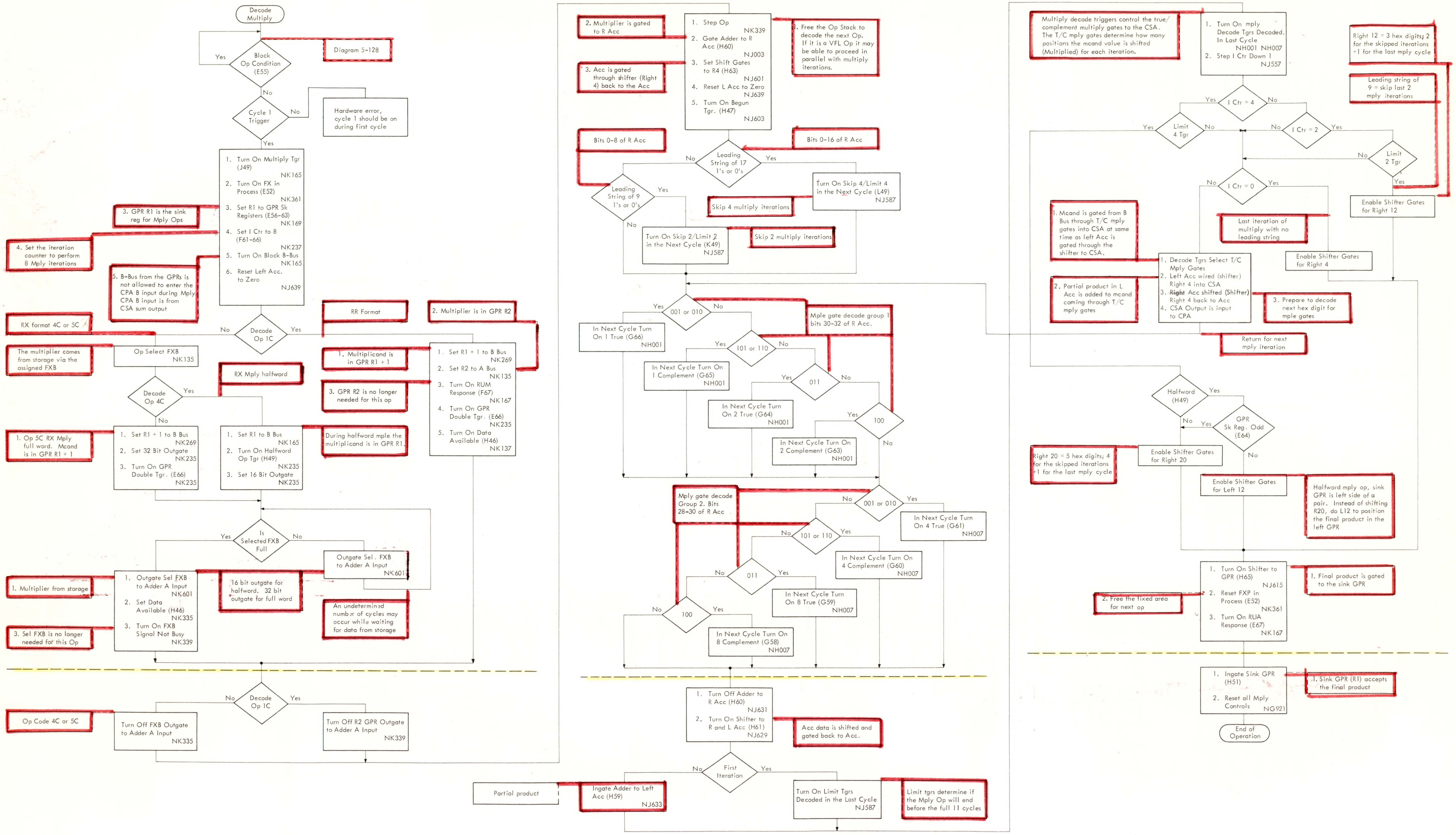
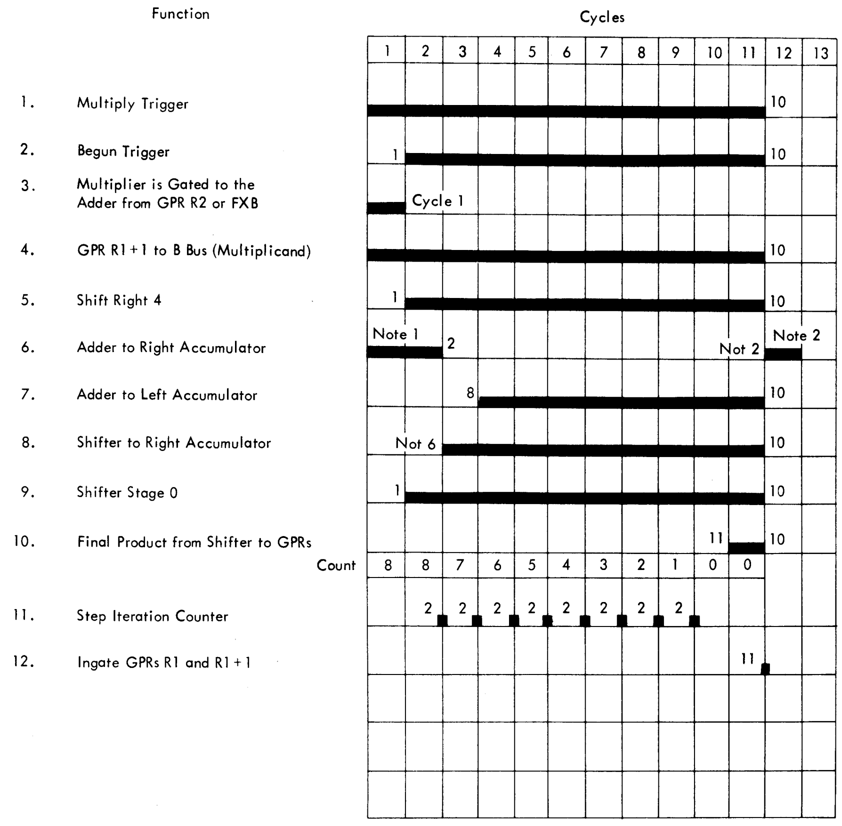
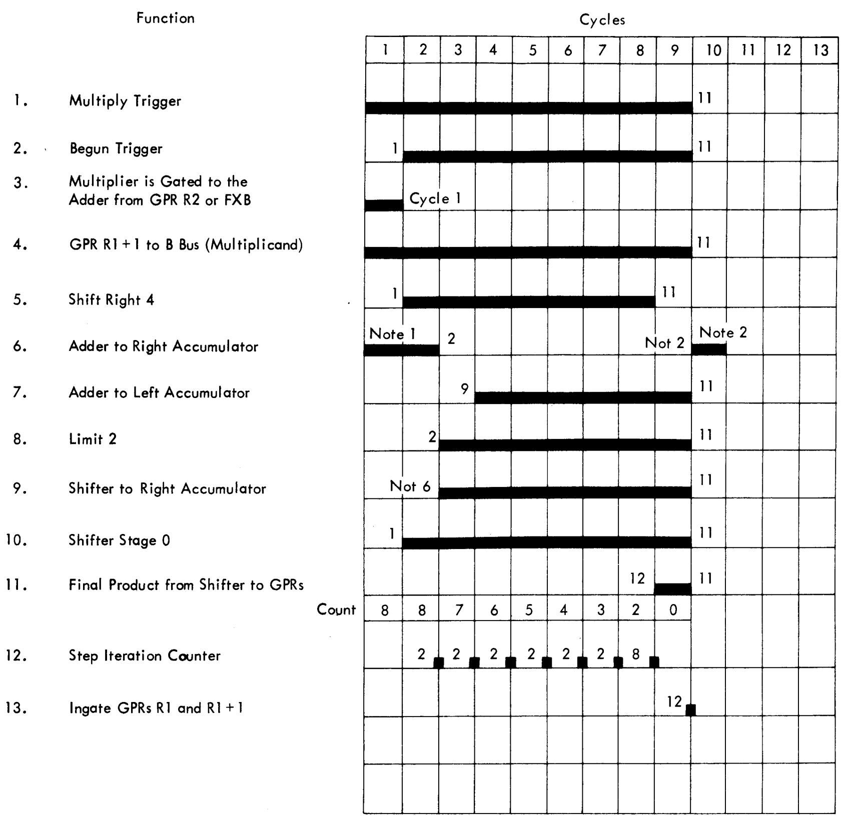

Decode Multiply
Yes
Block Op Condition (E55)
Diagram 5-128
No
Cycle 1 Trigger
No
Hardware error, cycle 1 should be on during first cycle
Yes
1. Turn On Multiply Tgr (J49) NK165
2. Turn On FX in Process (E52) NK361
3. Set R1 to GPR Sk Registers (E56-63) NK169
4. Set I Ctr to 8 (F61-66) NK237
5. Turn On Block B-Bus NK165
6. Reset Left Acc. to Zero NJ639
3. GPR R1 is the sink reg for Mply Ops
4. Set the iteration counter to perform 8 Mply iterations
5. B-Bus from the GPRs is not allowed to enter the CPA B input during Mply. CPA B input is from CSA sum output
RX format 4C or 5C
The multiplier comes from storage via the assigned FXB
No
Decode Op 1C
Yes
RR Format
2. Multiplier is in GPR R2
Op Select FXB NK135
RX Mply halfword
1. Multiplicand is in GPR R1 + 1
3. GPR R2 is no longer needed for this op
1. Set R1 + 1 to B Bus NK269
2. Set R2 to A Bus NK135
3. Turn On RUM Response (F67) NK167
4. Turn On GPR Double Tgr. (E66) NK235
5. Turn On Data Available (H46) NK137
Decode Op 4C
Yes
No
1. Op 5C RX Mply full word. Mcand is in GPR R1 + 1
1. Set R1 + 1 to B Bus NK269
2. Set 32 Bit Outgate NK235
3. Turn On GPR Double Tgr. (E66) NK235
1. Set R1 to B Bus NK165
2. Turn On Halfword Op Tgr (H49) NK235
3. Set 16 Bit Outgate NK235
During halfword mple the multiplicand is in GPR R1.
Yes
Is Selected FXB Full
No
Outgate Sel. FXB to Adder A Input NK601
1. Multiplier from storage
1. Outgate Sel FXB to Adder A Input NK601
2. Set Data Available (H46) NK335
3. Turn On FXB Signal Not Busy NK339
16 bit outgate for halfword. 32 bit outgate for full word
An undetermined number of cycles may occur while waiting for data from storage
3. Sel FXB is no longer needed for this Op
No
Decode Op 1C
Yes
Op Code 4C or 5C
Turn Off FXB Outgate to Adder A Input NK335
Turn Off R2 GPR Outgate to Adder A Input NK339
2. Multiplier is gated to R Acc
3. Acc is gated through shifter (Right 4) back to the Acc
1. Step Op NK339
2. Gate Adder to R Acc (H60) NJ003
3. Set Shift Gates to R4 (H63) NJ601
4. Reset L Acc to Zero NJ639
5. Turn On Begun Tgr. (H47) NJ603
1. Free the Op Stack to decode the next Op. If it is a VFL Op it may be able to proceed in parallel with multiply iterations.
Bits 0-8 of R Acc
Bits 0-16 of R Acc
No
Leading String of 17 1's or 0's
Yes
Leading String of 9 1's or 0's
Yes
No
Turn On Skip 4/Limit 4 in the Next Cycle (L49) NJ587
Skip 4 multiply iterations
Turn On Skip 2/Limit 2 in the Next Cycle (K49) NJ587
Skip 2 multiply iterations
Yes
001 or 010
No
In Next Cycle Turn On 1 True (G66) NH001
Mple gate decode group 1 bits 30-32 of R Acc.
Yes
101 or 110
No
In Next Cycle Turn On 1 Complement (G65) NH001
Yes
011
No
In Next Cycle Turn On 2 True (G64) NH001
Yes
100
No
In Next Cycle Turn On 2 Complement (G63) NH001
Mply gate decode Group 2. Bits 28-30 of R Acc
No
001 or 010
Yes
In Next Cycle Turn On 4 True (G61) NH007
No
101 or 110
Yes
In Next Cycle Turn On 4 Complement (G60) NH007
No
011
Yes
In Next Cycle Turn On 8 True (G59) NH007
No
100
Yes
In Next Cycle Turn On 8 Complement (G58) NH007
1. Turn Off Adder to R Acc (H60) NJ631
2. Turn On Shifter to R and L Acc (H61) NJ629
Acc data is shifted and gated back to Acc.
No
First Iteration
Yes
Partial product
Ingate Adder to Left Acc (H59) NJ633
Turn On Limit Tgrs Decoded in the Last Cycle NJ587
Limit tgrs determine if the Mply Op will end before the full 11 cycles
Multiply decode triggers control the true/complement multiply gates to the CSA. The T/C mply gates determine how many positions the mcand value is shifted (Multiplied) for each iteration.
1. Turn On mply Decode Tgrs Decoded. In Last Cycle NH001 NH007
2. Step I Ctr Down 1 NJ557
Right 12 = 3 hex digits; 2 for the skipped iterations +1 for the last mply cycle
Leading string of 9 = skip last 2 mply iterations
Yes
I Ctr = 4
No
Yes
Limit 4 Tgr
No
No
I Ctr = 2
Yes
No
Limit 2 Tgr
Yes
Enable Shifter Gates for Right 12
No
I Ctr = 0
Yes
1. Mcand is gated from B Bus through T/C mply gates into CSA at same time as left Acc is gated through the shifter to CSA.
Last iteration of multiply with no leading string
Enable Shifter Gates for Right 4
1. Decode Tgrs Select T/C Mply Gates
2. Left Acc wired (shifter) Right 4 into CSA
3. Right Acc shifted (Shifter) Right 4 back to Acc
4. CSA Output is input to CPA
2. Partial product in L Acc is added to mcand coming through T/C mply gates
3. Prepare to decode next hex digit for mple gates
Return for next mply iteration
Halfword (H49)
Yes
No
Yes
GPR Sk Reg. Odd (E64)
No
Right 20 = 5 hex digits; 4 for the skipped iterations +1 for the last mply cycle
Enable Shifter Gates for Right 20
Enable Shifter Gates for Left 12
Halfword mply op, sink GPR is left side of a pair. Instead of shifting R20, do L12 to position the final product in the left GPR
1. Turn On Shifter to GPR (H65) NJ615
2. Reset FXP in Process (E52) NK361
3. Turn On RUA Response (E67) NK167
1. Final product is gated to the sink GPR
2. Free the fixed area for next op
1. Ingate Sink GPR (H51)
2. Reset all Mply Controls NG921
1. Sink GPR (R1) accepts the final product
End of Operation

DIAGRAM 5-101. MULTIPLY (SHEET 2 OF 3)

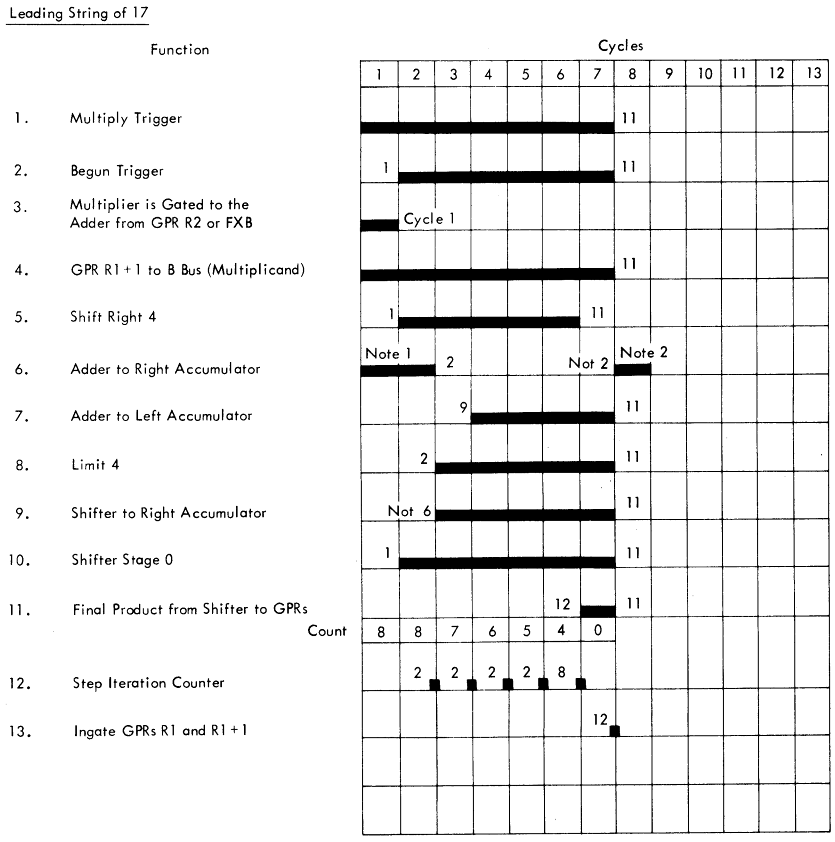
With No Leading String


Function
Cycles
1 2 3 4 5 6 7 8 9 10 11 12 13
1. Multiply Trigger
10
2. Begun Trigger
1
10
3. Multiplier is Gated to the Adder from GPR R2 or FXB
Cycle 1
4. GPR R1 + 1 to B Bus (Multiplicand)
10
5. Shift Right 4
1
10
6. Adder to Right Accumulator
Note 1
2
Not 2
Note 2
7. Adder to Left Accumulator
8
10
8. Shifter to Right Accumulator
Not 6
10
9. Shifter Stage 0
1
10
10. Final Product from Shifter to GPRs
11
10
Count
8 8 7 6 5 4 3 2 1 0 0
11. Step Iteration Counter
2 2 2 2 2 2 2 2
12. Ingate GPRs R1 and R1 + 1
11

Leading String of 9


Function
Cycles
1 2 3 4 5 6 7 8 9 10 11 12 13
1. Multiply Trigger
11
2. Begun Trigger
1
11
3. Multiplier is Gated to the Adder from GPR R2 or FXB
Cycle 1
4. GPR R1 + 1 to B Bus (Multiplicand)
11
5. Shift Right 4
1
11
6. Adder to Right Accumulator
Note 1
2
Not 2
Note 2
7. Adder to Left Accumulator
9
11
8. Limit 2
2
11
9. Shifter to Right Accumulator
Not 6
11
10. Shifter Stage 0
1
11
11. Final Product from Shifter to GPRs
12
11
Count
8 8 7 6 5 4 3 2 0
12. Step Iteration Counter
2 2 2 2 2 2 2 8
13. Ingate GPRs R1 and R1 + 1
12

Leading String of 17


Function
Cycles
1 2 3 4 5 6 7 8 9 10 11 12 13
1. Multiply Trigger
11
2. Begun Trigger
1
11
3. Multiplier is Gated to the Adder from GPR R2 or FXB
Cycle 1
4. GPR R1 + 1 to B Bus (Multiplicand)
11
5. Shift Right 4
1
11
6. Adder to Right Accumulator
Note 1
2
Not 2
Note 2
7. Adder to Left Accumulator
9
11
8. Limit 4
2
11
9. Shifter to Right Accumulator
Not 6
11
10. Shifter Stage 0
1
11
11. Final Product from Shifter to GPRs
12
11
Count
8 8 7 6 5 4 0
12. Step Iteration Counter
2 2 2 2 8
13. Ingate GPRs R1 and R1 + 1
12

Notes:

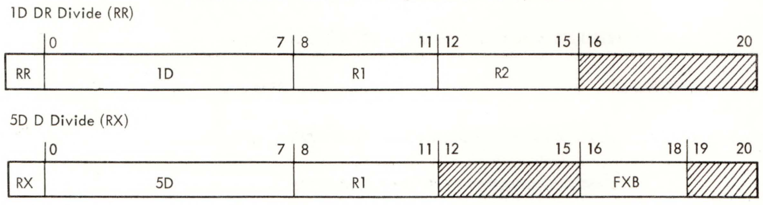
1. Depends on previous op.
2. Depends on following op.

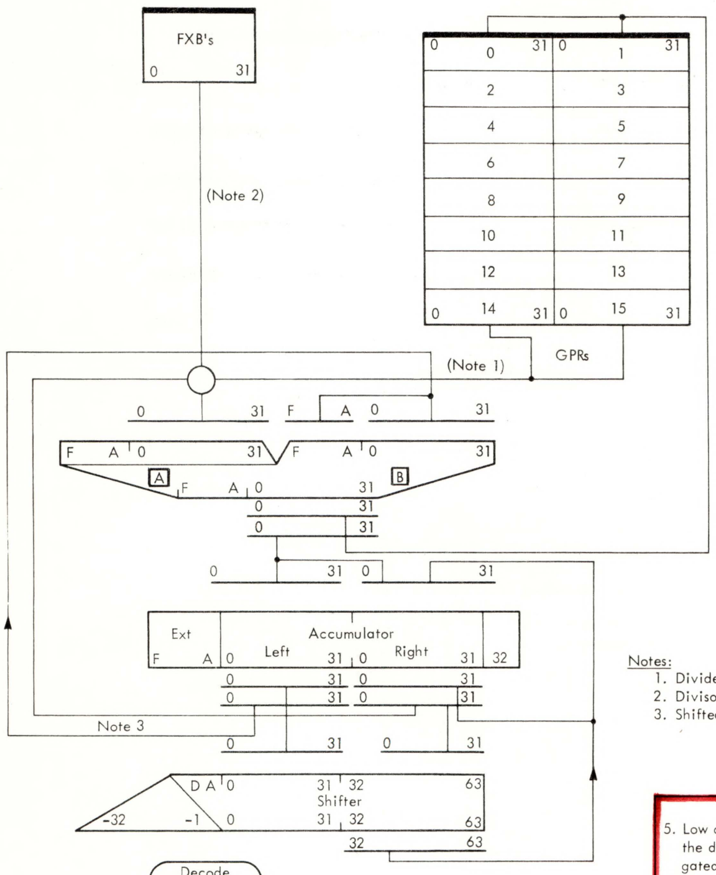
DIAGRAM 5-101. MULTIPLY (SHEET 3 OF 3)

Objectives:

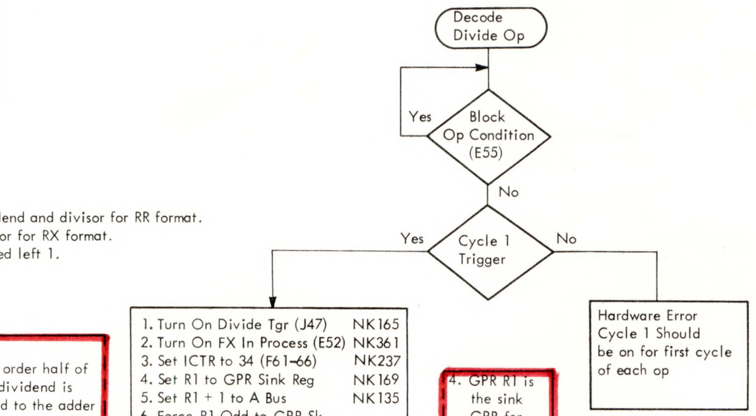
1. During divide operations the value in GPR R1 and R1+1 (Dividend) is divided by the value (Divisor) in either GPR R2 (RR) or an assigned FXB (RX). The results are placed back in GPR R1 (Remainder) and GPR R1+1 (Quotient).
2. Division is performed by repetitively subtracting the divisor from the dividend and at the same time shifting 1 bit position for each of 32 iterations.
3. The quotient is constructed bit by bit from the results of each subtraction.
4. In the last iteration, subtraction may cause the remainder to go negative (less than zero). In this case the divisor must be added back in once to correct the remainder.
5. Remainder, Divisor and Dividend signs determine the ending sequence which adjusts the results to conform to standard mathematical sign rules.
6. During the ending sequence the quotient is transferred to GPR R1+1 and the remainder is transferred to GPR R1.
7. Divide operations may be aborted if the factors exceed the capacity of machine circuits.
8. A complete divide operation requires 36 or 37 machine cycles depending on how many cycles are required in the ending sequence.

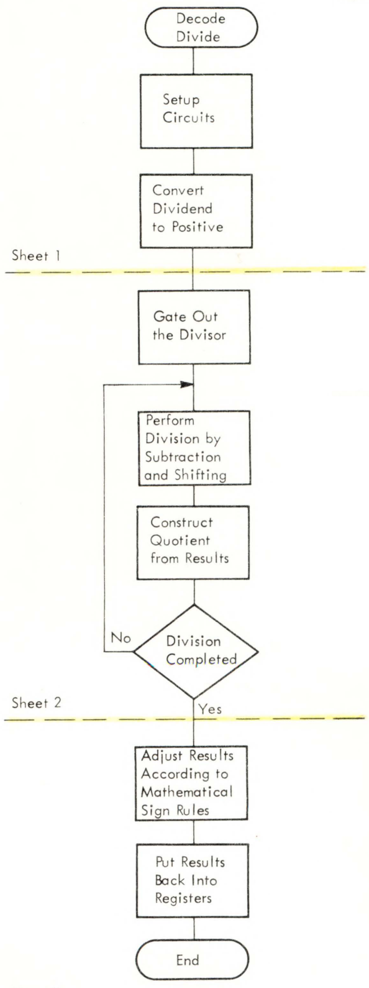
Condition Code Remains Unchanged For Divide Operations.





Notes:
1. Dividend and divisor for RR format.
2. Divisor for RX format.
3. Shifted left 1.


Decode Divide Op
Yes
Block Op Condition (E55)
No
Yes
Cycle 1 Trigger
No
Hardware Error Cycle 1 Should be on for first cycle of each op
5. Low order half of the dividend is gated to the adder
1. Turn On Divide Tgr (J47) NK165
2. Turn On FX In Process (E52) NK361
3. Set ICTR to 34 (F61-66) NK237
4. Set R1 to GPR Sink Reg NK169
5. Set R1 + 1 to A Bus NK135
6. Force R1 Odd to GPR Sk Reg (E64) NK309
4. GPR R1 is the sink GPR for this op


Decode Divide
Setup Circuits
Convert Dividend to Positive
Sheet 1
Gate Out the Divisor
Perform Division by Subtraction and Shifting
Construct Quotient from Results
No
Division Completed
Sheet 2
Yes
Adjust Results According to Mathematical Sign Rules
Put Results Back Into Registers
End
Sheet 3

Basic Flow of Divide Operations

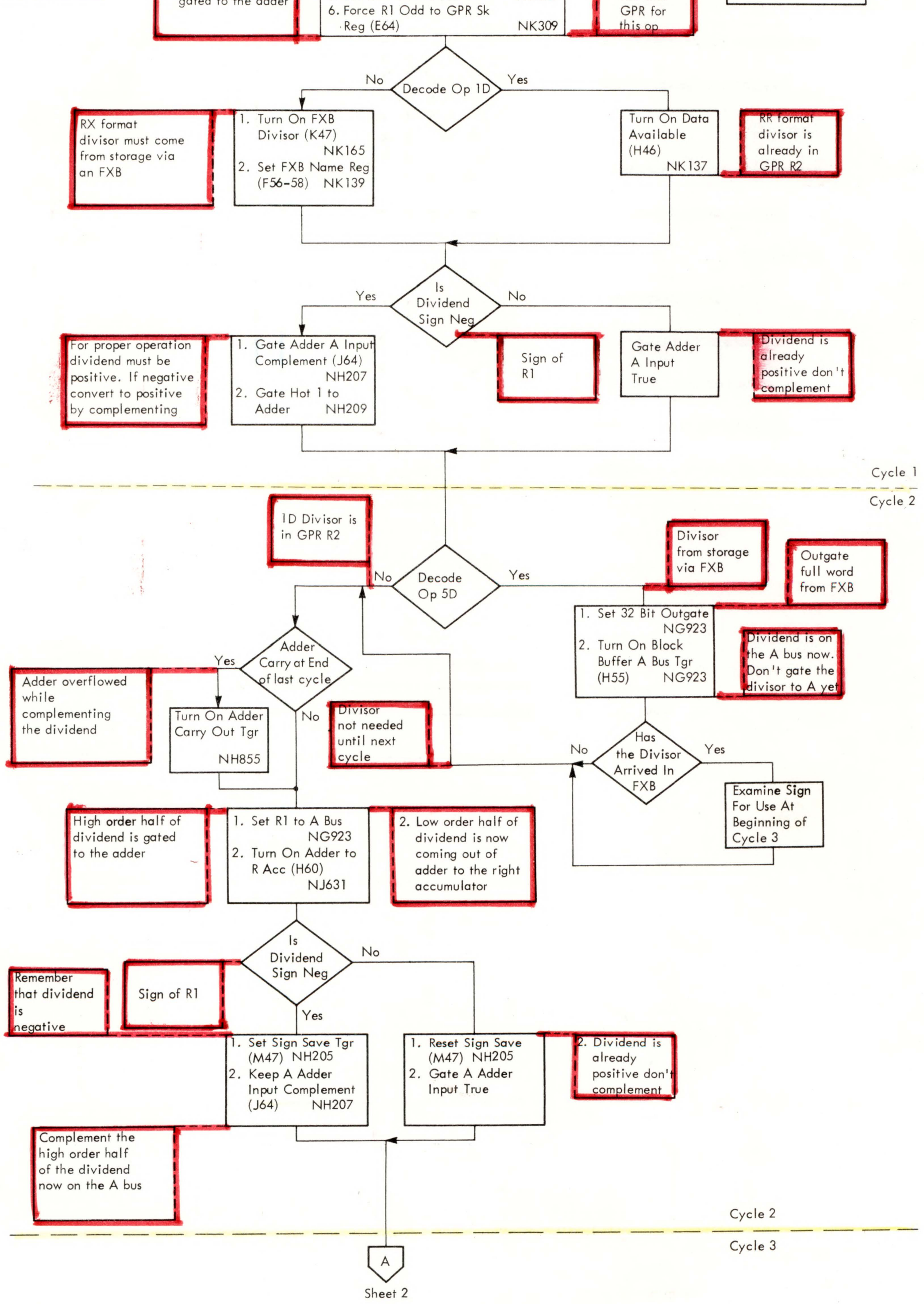

No
Decode Op 1D
Yes
RX format divisor must come from storage via an FXB
1. Turn On FXB Divisor (K47) NK165
2. Set FXB Name Reg (F56-58) NK139
Turn On Data Available (H46) NK137
RR format divisor is already in GPR R2
Yes
Is Dividend Sign Neg
No
For proper operation dividend must be positive. If negative convert to positive by complementing
1. Gate Adder A Input Complement (J64) NH207
2. Gate Hot 1 to Adder NH209
Sign of R1
Gate Adder A Input True
Dividend is already positive don't complement
Cycle 1
Cycle 2
1D Divisor is in GPR R2
No
Decode Op 5D
Yes
Divisor from storage via FXB
Outgate full word from FXB
1. Set 32 Bit Outgate NG923
2. Turn On Block Buffer A Bus Tgr (H55) NG923
Dividend is on the A bus now. Don't gate the divisor to A yet
Adder Carry at End of last cycle
Yes
No
Adder overflowed while complementing the dividend
Turn On Adder Carry Out Tgr NH855
Divisor not needed until next cycle
No
Has the Divisor Arrived In FXB
Yes
Examine Sign For Use At Beginning of Cycle 3
High order half of dividend is gated to the adder
1. Set R1 to A Bus NG923
2. Turn On Adder to R Acc (H60) NJ631
2. Low order half of dividend is now coming out of adder to the right accumulator
Is Dividend Sign Neg
No
Yes
Remember that dividend is negative
Sign of R1
1. Set Sign Save Tgr (M47) NH205
2. Keep A Adder Input Complement (J64) NH207
1. Reset Sign Save (M47) NH205
2. Gate A Adder Input True
2. Dividend is already positive don't complement
Complement the high order half of the dividend now on the A bus
Cycle 2
Cycle 3
A
Sheet 2

DIAGRAM 5-102. DIVIDE (SHEET 1 OF 4)

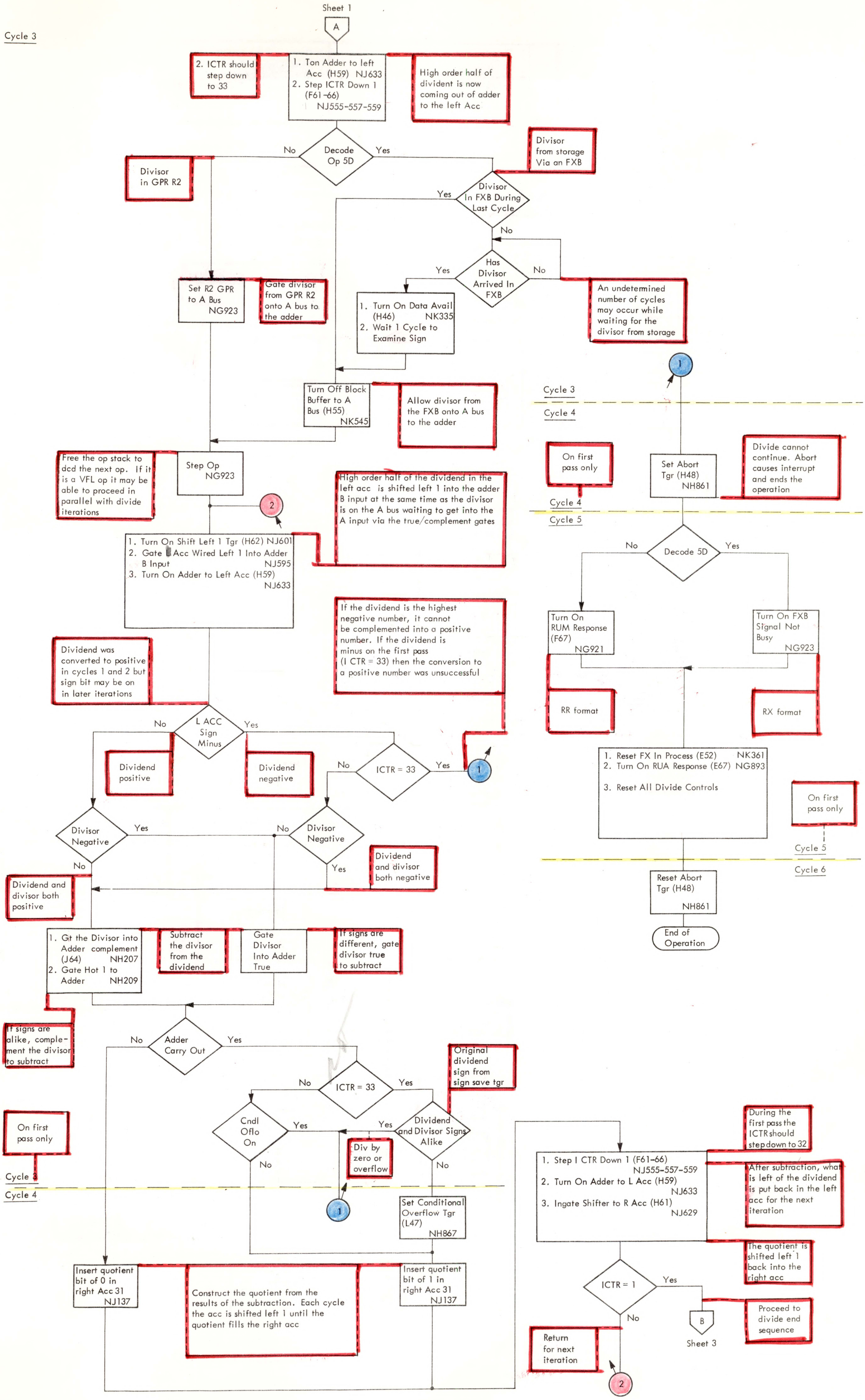
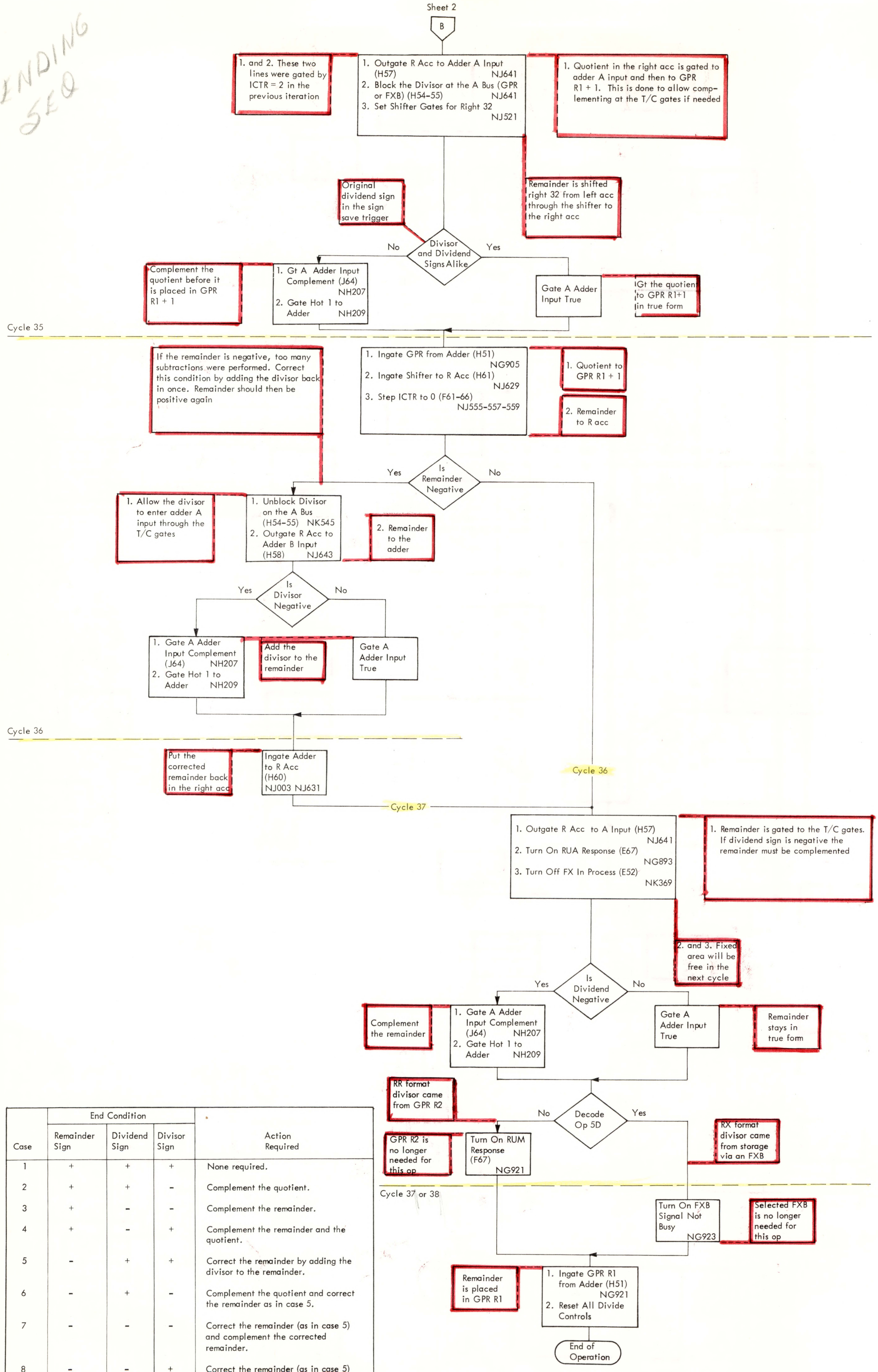

Cycle 3
Sheet 1
A
2. ICTR should step down to 33
1. Ton Adder to left Acc (H59) NJ633
2. Step ICTR Down 1 (F61-66) NJ555-557-559
High order half of divident is now coming out of adder to the left Acc
No
Decode Op 5D
Yes
Divisor from storage Via an FXB
Divisor in GPR R2
Yes
Divisor In FXB During Last Cycle
No
Yes
Has Divisor Arrived In FXB
No
Set R2 GPR to A Bus NG923
Gate divisor from GPR R2 onto A bus to the adder
1. Turn On Data Avail (H46) NK335
2. Wait 1 Cycle to Examine Sign
An undetermined number of cycles may occur while waiting for the divisor from storage
1
Turn Off Block Buffer to A Bus (H55) NK545
Allow divisor from the FXB onto A bus to the adder
Cycle 3
Cycle 4
Free the op stack to dcd the next op. If it is a VFL op it may be able to proceed in parallel with divide iterations
Step Op NG923
2
High order half of the dividend in the left acc is shifted left 1 into the adder B input at the same time as the divisor is on the A bus waiting to get into the A input via the true/complement gates
On first pass only
Cycle 4
Cycle 5
Set Abort Tgr (H48) NH861
Divide cannot continue. Abort causes interrupt and ends the operation
1. Turn On Shift Left 1 Tgr (H62) NJ601
2. Gate L Acc Wired Left 1 Into Adder B Input NJ595
3. Turn On Adder to Left Acc (H59) NJ633
No
Decode 5D
Yes
If the dividend is the highest negative number, it cannot be complemented into a positive number. If the dividend is minus on the first pass (I CTR = 33) then the conversion to a positive number was unsuccessful
Turn On RUM Response (F67) NG921
Turn On FXB Signal Not Busy NG923
Dividend was converted to positive in cycles 1 and 2 but sign bit may be on in later iterations
RR format
RX format
No
L ACC Sign Minus
Yes
No
ICTR = 33
Yes
1
Dividend positive
Dividend negative
1. Reset FX In Process (E52) NK361
2. Turn On RUA Response (E67) NG893
3. Reset All Divide Controls
On first pass only
Cycle 5
Cycle 6
Divisor Negative
Yes
No
Divisor Negative
No
Yes
Dividend and divisor both negative
Dividend and divisor both positive
Reset Abort Tgr (H48) NH861
End of Operation
1. Gt the Divisor into Adder complement (J64) NH207
2. Gate Hot 1 to Adder NH209
Subtract the divisor from the dividend
Gate Divisor Into Adder True
If signs are different, gate divisor true to subtract
If signs are alike, complement the divisor to subtract
No
Adder Carry Out
Yes
Original dividend sign from sign save tgr
No
ICTR = 33
Yes
On first pass only
Cndl Oflo On
Yes
Yes
Dividend and Divisor Signs Alike
Div by zero or overflow
During the first pass the ICTR should step down to 32
No
No
Cycle 3
Cycle 4
1
Set Conditional Overflow Tgr (L47) NH867
1. Step I CTR Down 1 (F61-66) NJ555-557-559
2. Turn On Adder to L Acc (H59) NJ633
3. Ingate Shifter to R Acc (H61) NJ629
After subtraction, what is left of the dividend is put back in the left acc for the next iteration
The quotient is shifted left 1 back into the right acc
Insert quotient bit of 0 in right Acc 31 NJ137
Construct the quotient from the results of the subtraction. Each cycle the acc is shifted left 1 until the quotient fills the right acc
Insert quotient bit of 1 in right Acc 31 NJ137
ICTR = 1
Yes
No
B
Sheet 3
Proceed to divide end sequence
Return for next iteration
2

DIAGRAM 5-102. DIVIDE (SHEET 2 OF 4)

ENDING SEQ

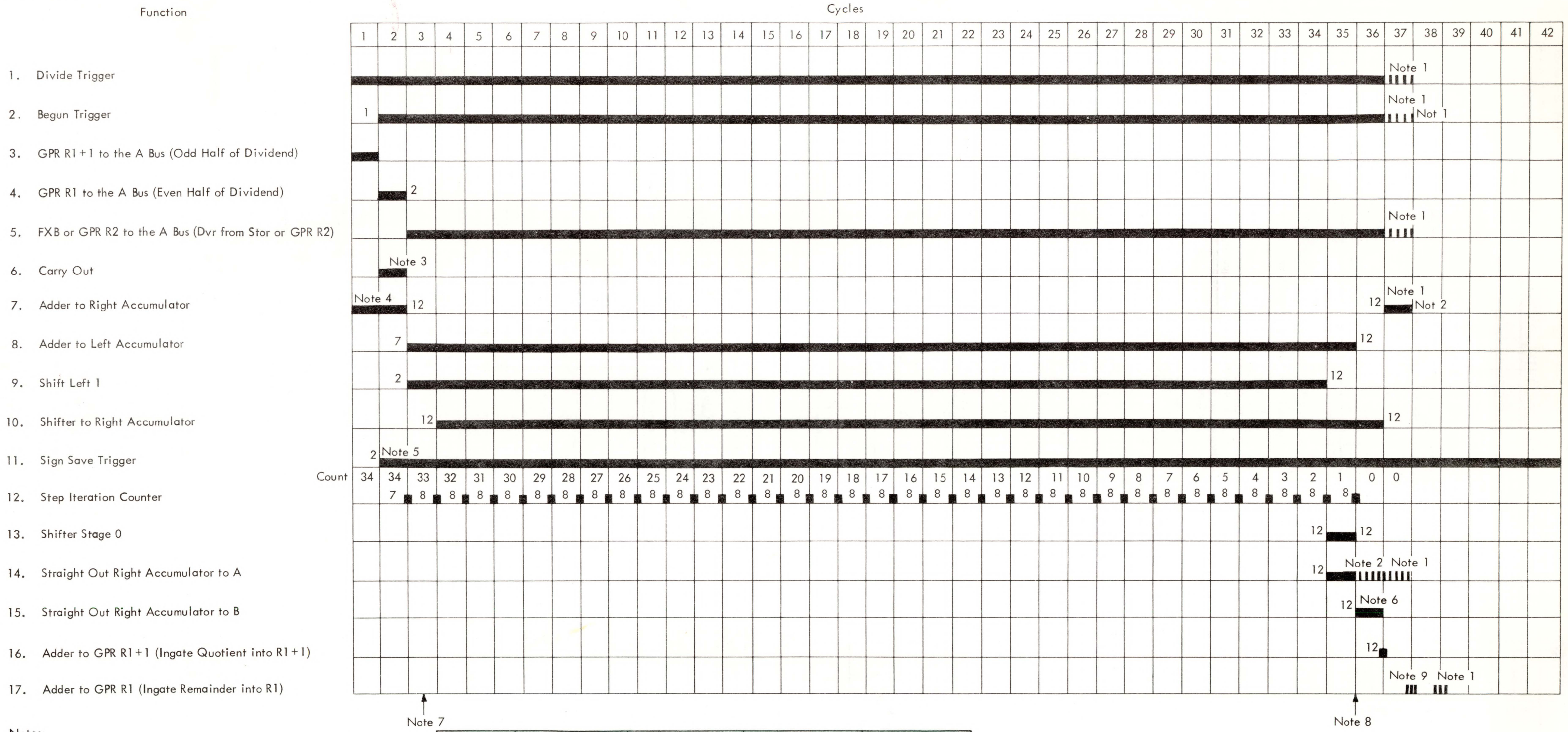


Sheet 2
B
1. and 2. These two lines were gated by ICTR = 2 in the previous iteration
1. Outgate R Acc to Adder A Input (H57) NJ641
2. Block the Divisor at the A Bus (GPR or FXB) (H54-55) NJ641
3. Set Shifter Gates for Right 32 NJ521
1. Quotient in the right acc is gated to adder A input and then to GPR R1 + 1. This is done to allow complementing at the T/C gates if needed
Remainder is shifted right 32 from left acc through the shifter to the right acc
Original dividend sign in the sign save trigger
Divisor and Dividend Signs Alike
No
Yes
Complement the quotient before it is placed in GPR R1 + 1
1. Gt A Adder Input Complement (J64) NH207
2. Gate Hot 1 to Adder NH209
Gate A Adder Input True
Gt the quotient to GPR R1+1 in true form
Cycle 35
If the remainder is negative, too many subtractions were performed. Correct this condition by adding the divisor back in once. Remainder should then be positive again
1. Ingate GPR from Adder (H51) NG905
2. Ingate Shifter to R Acc (H61) NJ629
3. Step ICTR to 0 (F61-66) NJ555-557-559
1. Quotient to GPR R1 + 1
2. Remainder to R acc
Is Remainder Negative
Yes
No
1. Allow the divisor to enter adder A input through the T/C gates
1. Unblock Divisor on the A Bus (H54-55) NK545
2. Outgate R Acc to Adder B Input (H58) NJ643
2. Remainder to the adder
Is Divisor Negative
Yes
No
1. Gate A Adder Input Complement (J64) NH207
2. Gate Hot 1 to Adder NH209
Add the divisor to the remainder
Gate A Adder Input True
Cycle 36
Put the corrected remainder back in the right acc
Ingate Adder to R Acc (H60) NJ003 NJ631
Cycle 36
Cycle 37
1. Outgate R Acc to A Input (H57) NJ641
2. Turn On RUA Response (E67) NG893
3. Turn Off FX In Process (E52) NK369
1. Remainder is gated to the T/C gates. If dividend sign is negative the remainder must be complemented
2. and 3. Fixed area will be free in the next cycle
Is Dividend Negative
Yes
No
Complement the remainder
1. Gate A Adder Input Complement (J64) NH207
2. Gate Hot 1 to Adder NH209
Gate A Adder Input True
Remainder stays in true form
RR format divisor came from GPR R2
Decode Op 5D
No
Yes
RX format divisor came from storage via an FXB
GPR R2 is no longer needed for this op
Turn On RUM Response (F67) NG921
Cycle 37 or 38
Turn On FXB Signal Not Busy NG923
Selected FXB is no longer needed for this op
Remainder is placed in GPR R1
1. Ingate GPR R1 from Adder (H51) NG921
2. Reset All Divide Controls
End of Operation

	End Condition			
Case	Remainder Sign	Dividend Sign	Divisor Sign	Action Required
1	+	+	+	None required.
2	+	+	-	Complement the quotient.
3	+	-	-	Complement the remainder.
4	+	-	+	Complement the remainder and the quotient.
5	-	+	+	Correct the remainder by adding the divisor to the remainder.
6	-	+	-	Complement the quotient and correct the remainder as in case 5.
7	-	-	-	Correct the remainder (as in case 5) and complement the corrected remainder.
8	-	-	+	Correct the remainder (as in case 5) and complement the quotient and the corrected remainder.

DIAGRAM 5-102. DIVIDE (SHEET 3 OF 4)

Divide Timing



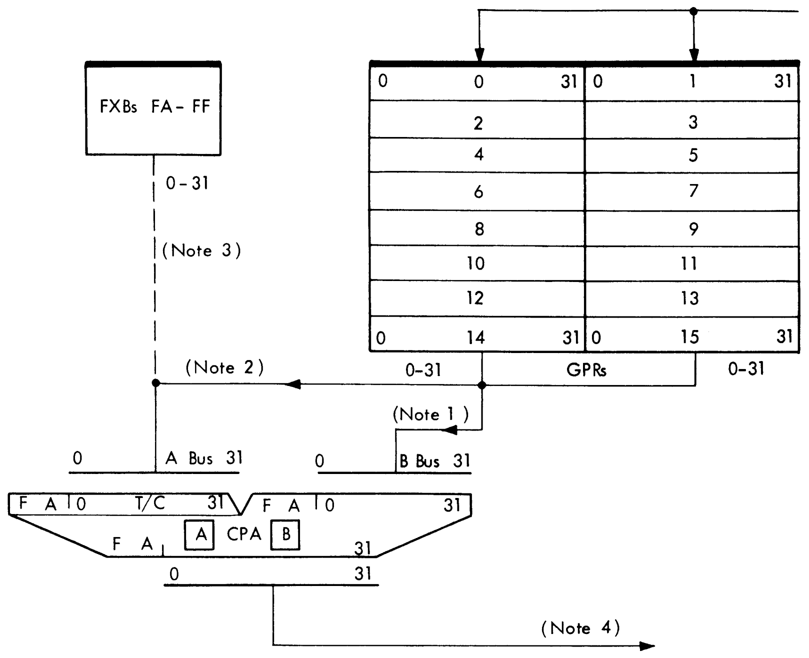
Notes:

1. On only for cases 7 and 8.
2. On only for cases 1 to 4.
3. On only if there is a carry between halves of the dividend.
4. Depends on previous op.
5. On only if dividend is negative (Not reset at end of op.)
6. On only for cases 5 to 8.
7. Wait during this cycle until data is available.
8. Last iteration into the accumulator occurs at this cycle boundary.
9. On only for cases 1 to 6.

Case	Remainder	Dividend	Divisor	Action
1	Positive	Positive	Positive	None
2	Positive	Positive	Negative	CQ
3	Positive	Negative	Negative	CR
4	Positive	Negative	Positive	CQ CR
5	Negative	Positive	Positive	CC
6	Negative	Positive	Negative	CQ CC
7	Negative	Negative	Negative	CC CR
8	Negative	Negative	Positive	CQ CC CR

CQ - Complement Quotient
CR - Complement Remainder
CC - Correction Cycle

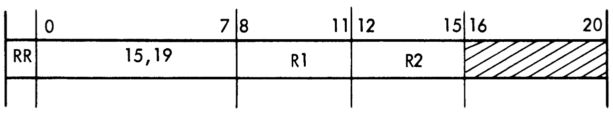
DIAGRAM 5-102. DIVIDE (SHEET 4 OF 4)



Notes:

1. R1 Field Data (RR and RX Formats).
2. R2 Field Data (RR Format Only).
3. FXB Field Data (RX Format Only).
4. Results set the CC but are not gated back to a GPR.

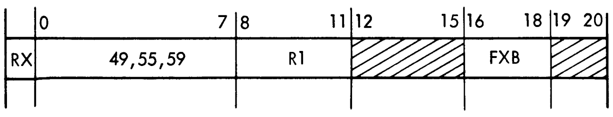
15	CLR	Compare Logical
19	CR	Compare



Condition Code

0 = Operands are equal
1 = First Operand is low
2 = First operand is high

49	CH	Compare Halfword
55	CL	Compare Logical
59	C	Compare



1. Compare operations gate data from the GPR's or Storage (FXB) to the carry propagate adder.

2. The value of R2 is subtracted from the value of R1 and the results set the condition code.

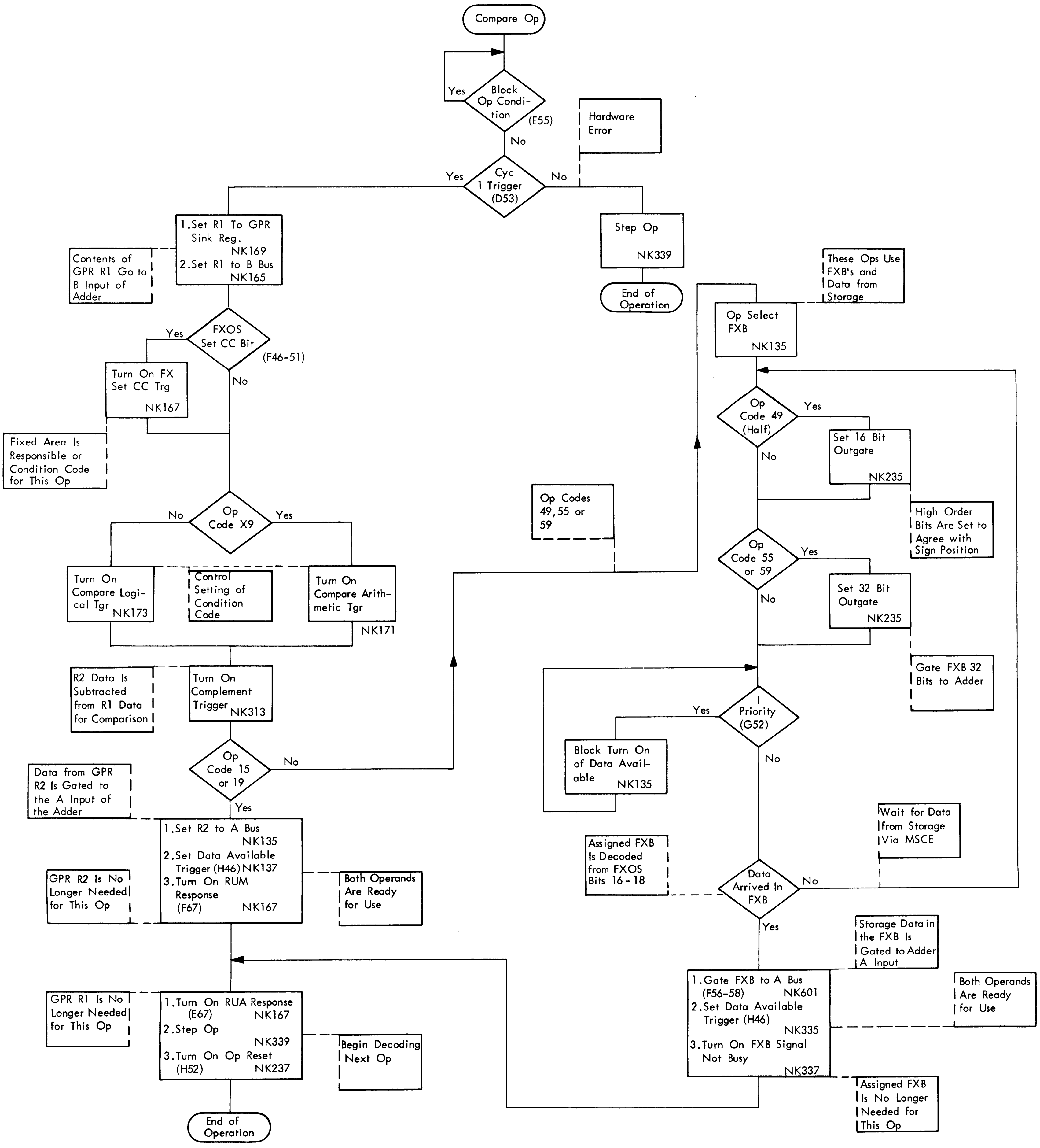
Operands are not changed by the compare operations.

4. Logical compares treat the data as 32 bit values without signs.

5. During halfword compare, 16 bits of data from the FXB are gated to the adder. The high order bits of the word are changed to agree with the halfword sign.

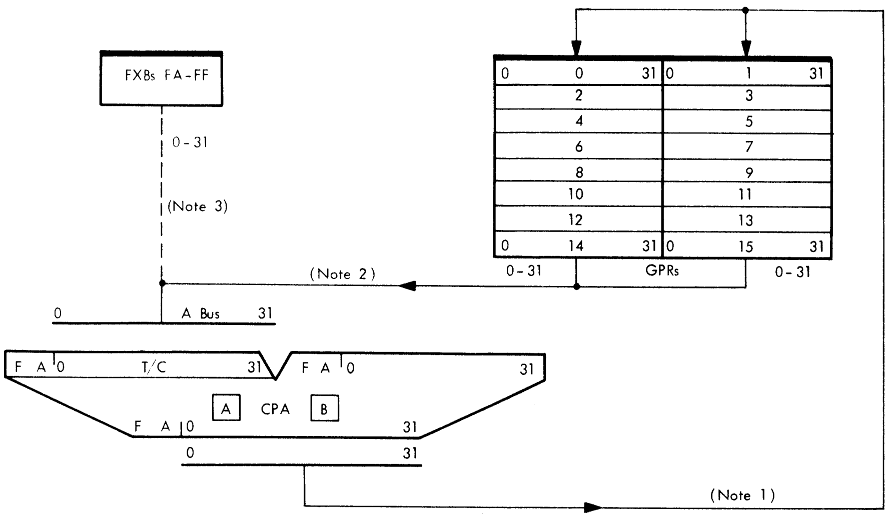
6. Normal execution time for ops 15 and 19 is 1 machine cycle.

7. Ops 49, 55 and 59 require an undetermined number of machine cycles because input data must come from storage via an FXB.



Compare Op
Yes
Block Op Condition (E55)
No
Hardware Error
Yes
Cyc 1 Trigger (D53)
No
1. Set R1 To GPR Sink Reg. NK169
2. Set R1 to B Bus NK165
Contents of GPR R1 Go to B Input of Adder
Step Op NK339
End of Operation
These Ops Use FXB's and Data from Storage
Op Select FXB NK135
Yes
FXOS Set CC Bit (F46-51)
No
Turn On FX Set CC Trg NK167
Fixed Area Is Responsible or Condition Code for This Op
Op Code 49 (Half)
Yes
No
Set 16 Bit Outgate NK235
High Order Bits Are Set to Agree with Sign Position
No
Op Code X9
Yes
Op Codes 49,55 or 59
Op Code 55 or 59
Yes
No
Set 32 Bit Outgate NK235
Turn On Compare Logical Tgr NK173
Control Setting of Condition Code
Turn On Compare Arithmetic Tgr NK171
Gate FXB 32 Bits to Adder
R2 Data Is Subtracted from R1 Data for Comparison
Turn On Complement Trigger NK313
Yes
Priority (G52)
Block Turn On of Data Available NK135
No
Data from GPR R2 Is Gated to the A Input of the Adder
Op Code 15 or 19
No
Yes
1. Set R2 to A Bus NK135
2. Set Data Available Trigger (H46) NK137
3. Turn On RUM Response (F67) NK167
Wait for Data from Storage Via MSCE
Assigned FXB Is Decoded from FXOS Bits 16-18
Data Arrived In FXB
No
GPR R2 Is No Longer Needed for This Op
Both Operands Are Ready for Use
Yes
Storage Data in the FXB Is Gated to Adder A Input
1. Gate FXB to A Bus (F56-58) NK601
2. Set Data Available Trigger (H46) NK335
3. Turn On FXB Signal Not Busy NK337
Both Operands Are Ready for Use
GPR R1 Is No Longer Needed for This Op
1. Turn On RUA Response (E67) NK167
2. Step Op NK339
3. Turn On Op Reset (H52) NK237
Begin Decoding Next Op
Assigned FXB Is No Longer Needed for This Op
End of Operation

DIAGRAM 5-103. COMPARE



Notes:

1. R1 Field Data (RR and RX Formats).
2. R2 Field Data (RR Format Only).
3. FXB Field Data (RX Format Only).

10	LPR	Load Positive
11	LNR	Load Negative
12	LTR	Load and Test
13	LCR	Load Complement
18	LR	Load

	0 7	8 11	12 15	16 20
RR	10, 11, 12, 13, 18	R1	R2	▨

48	LH	Load Halfword
58	L	Load

	0 7	8 11	12 15	16 18	19 20
RX	48, 58	R1	▨	FXB	▨

For ops 18, 48 and 58 the CC remains unchanged.

Ops 10, 11, 12 and 13 set the CC as follows:

CC Adder Condition

00 = Result is zero
01 = Result less than zero
10 = Result greater than zero
11 = Overflow

1. Load operations place information in a GPR (R1) from either another GPR (R2) or from storage via an FXB.
2. I-unit initiates the storage fetch (ops 48 and 58) and assigns an FXB to accept the data from MSCE.
3. All data is routed through the A input of the carry propagate adder to R1 to allow complementing when required (for ops 10, 11 and 13).
4. Load operations (except LH) move 32 bits of information in parallel into GPR R1.
5. LH moves 16 bits of information into the low order half of R1 and propagates the sign bit through the high order half of R1.
6. Operations 10, 11, 12, 13 and 18 require 1 machine cycle.
7. Operations 48 and 58 must wait for storage data and therefore require an undetermined number of machine cycles.

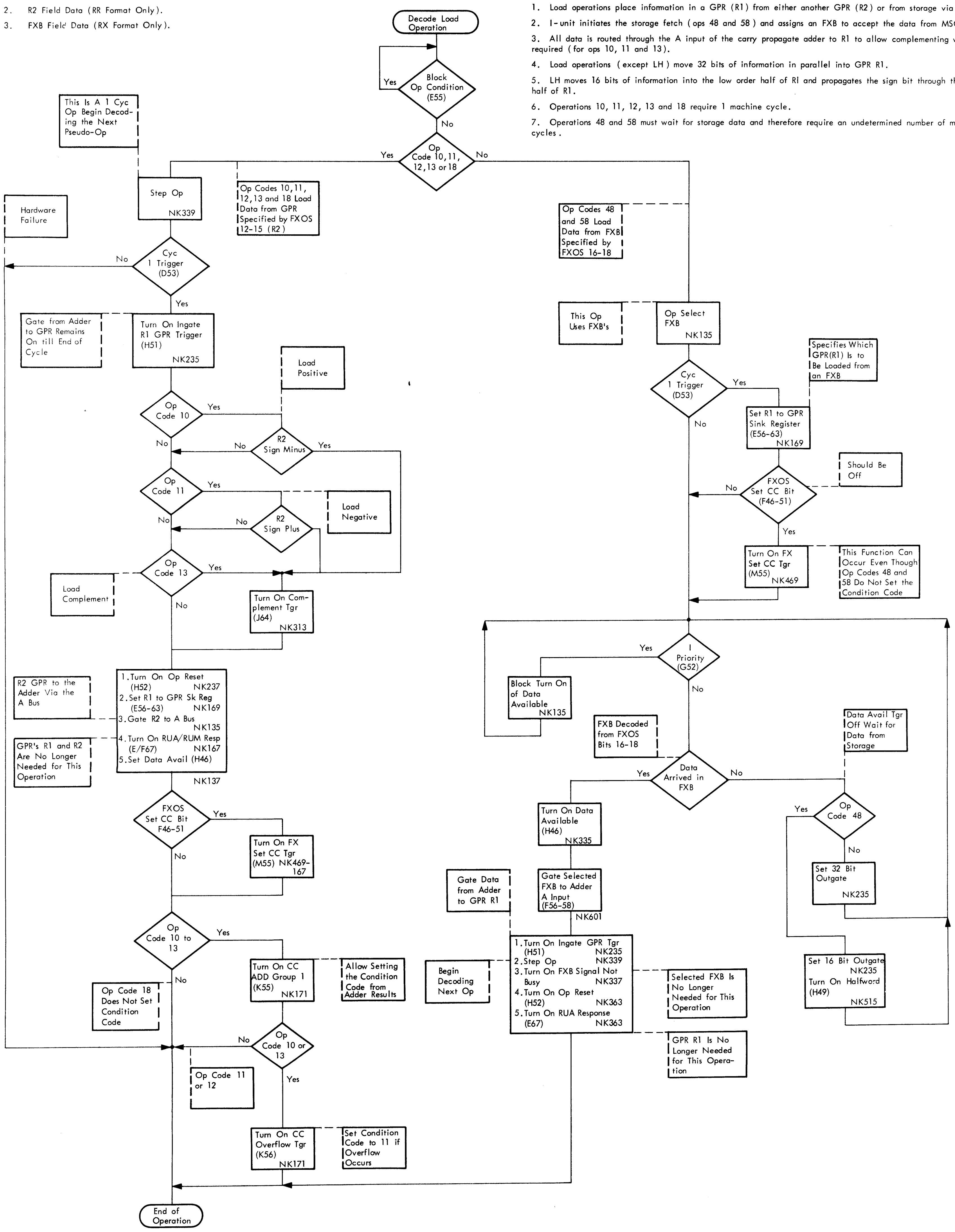
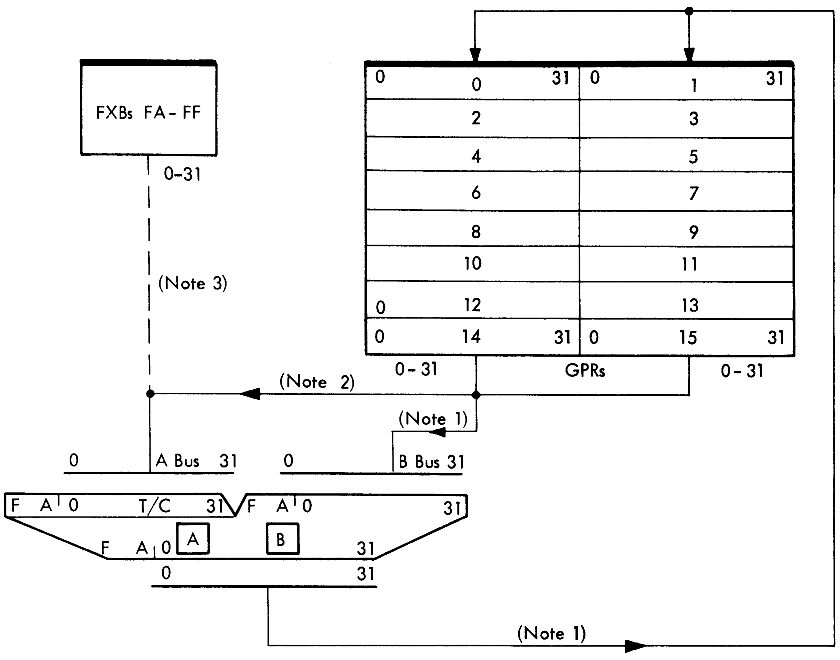

Decode Load Operation
Yes
Block Op Condition (E55)
No
This Is A 1 Cyc Op Begin Decoding the Next Pseudo-Op
Yes
Op Code 10, 11, 12, 13 or 18
No
Step Op NK339
Op Codes 10, 11, 12, 13 and 18 Load Data from GPR Specified by FXOS 12-15 (R2)
Hardware Failure
No
Cyc 1 Trigger (D53)
Yes
Gate from Adder to GPR Remains On till End of Cycle
Turn On Ingate R1 GPR Trigger (H51) NK235
Load Positive
Op Code 10
Yes
No
No
R2 Sign Minus
Yes
Op Code 11
Yes
Load Negative
No
No
R2 Sign Plus
Op Code 13
Yes
Load Complement
No
Turn On Complement Tgr (J64) NK313
R2 GPR to the Adder Via the A Bus
GPR's R1 and R2 Are No Longer Needed for This Operation
1. Turn On Op Reset (H52) NK237
2. Set R1 to GPR Sk Reg (E56-63) NK169
3. Gate R2 to A Bus NK135
4. Turn On RUA/RUM Resp (E/F67) NK167
5. Set Data Avail (H46)
NK137
FXOS Set CC Bit F46-51
Yes
No
Turn On FX Set CC Tgr (M55) NK469-167
Op Code 10 to 13
Yes
No
Op Code 18 Does Not Set Condition Code
Turn On CC ADD Group 1 (K55) NK171
Allow Setting the Condition Code from Adder Results
No
Op Code 10 or 13
Yes
Op Code 11 or 12
Turn On CC Overflow Tgr (K56) NK171
Set Condition Code to 11 if Overflow Occurs
End of Operation
Op Codes 48 and 58 Load Data from FXB Specified by FXOS 16-18
This Op Uses FXB's
Op Select FXB NK135
Specifies Which GPR(R1) Is to Be Loaded from an FXB
Cyc 1 Trigger (D53)
Yes
No
Set R1 to GPR Sink Register (E56-63) NK169
Should Be Off
No
FXOS Set CC Bit (F46-51)
Yes
Turn On FX Set CC Tgr (M55) NK469
This Function Can Occur Even Though Op Codes 48 and 58 Do Not Set the Condition Code
Yes
I Priority (G52)
No
Block Turn On of Data Available NK135
FXB Decoded from FXOS Bits 16-18
Data Avail Tgr Off Wait for Data from Storage
Yes
Data Arrived in FXB
No
Turn On Data Available (H46) NK335
Yes
Op Code 48
No
Set 32 Bit Outgate NK235
Gate Data from Adder to GPR R1
Gate Selected FXB to Adder A Input (F56-58) NK601
Set 16 Bit Outgate NK235
Turn On Halfword (H49) NK515
Begin Decoding Next Op
1. Turn On Ingate GPR Tgr (H51) NK235
2. Step Op NK339
3. Turn On FXB Signal Not Busy NK337
4. Turn On Op Reset (H52) NK363
5. Turn On RUA Response (E67) NK363
Selected FXB Is No Longer Needed for This Operation
GPR R1 Is No Longer Needed for This Operation

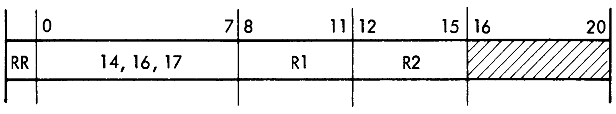
DIAGRAM 5-104. LOADS



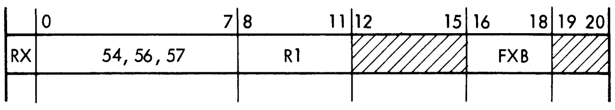
Notes:

1. R1 Field Data (RR and RX Formats)
2. R2 Field Data (RR Format Only)
3. FXB Field Data (RX Format Only)

14	NR	AND
16	OR	OR
17	XR	Exclusive OR



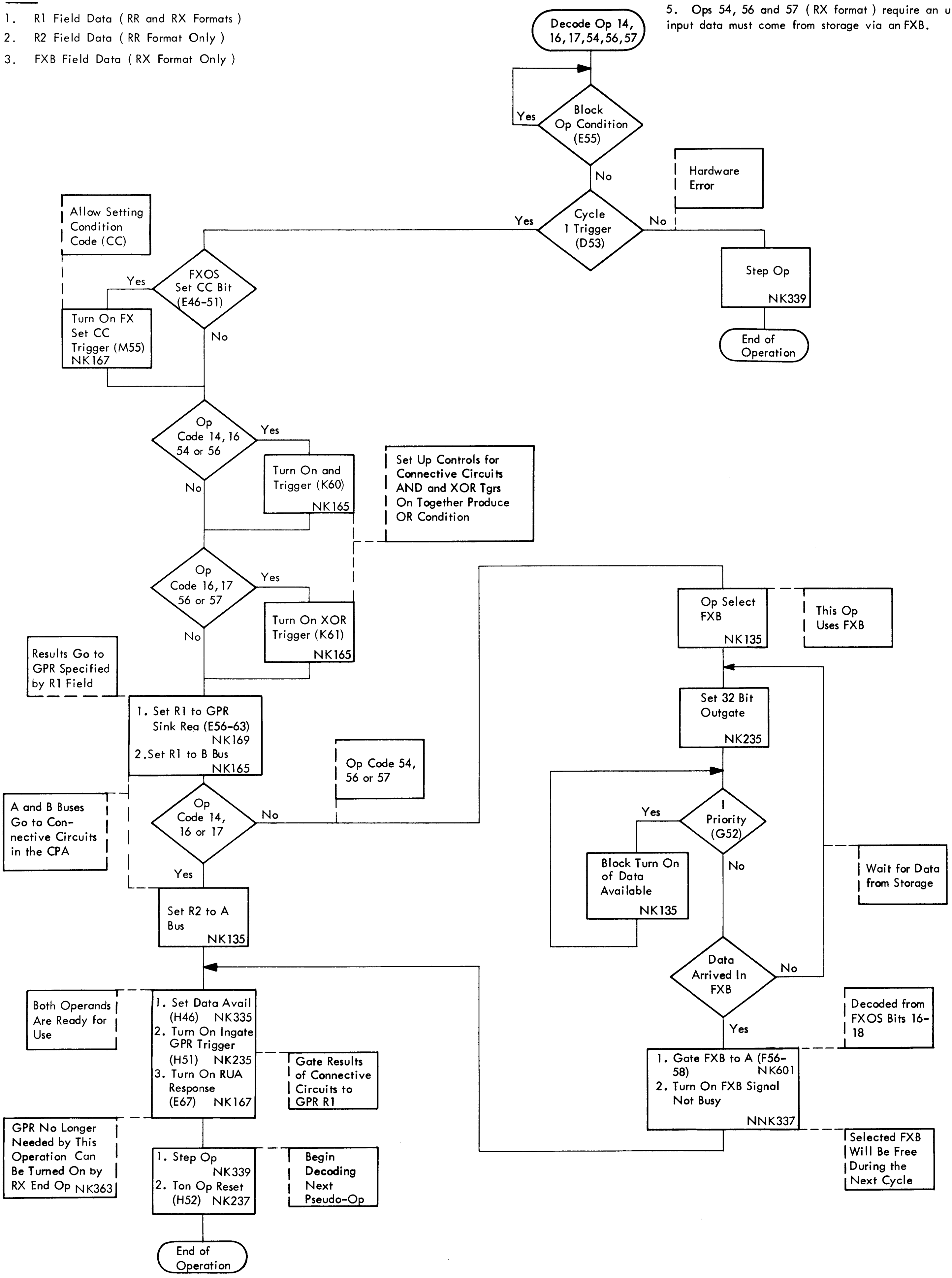
54	N	AND
56	O	OR
57	X	Exclusive OR



Condition Code

CC = 0: Results Zero
CC = 1: Results Not Zero

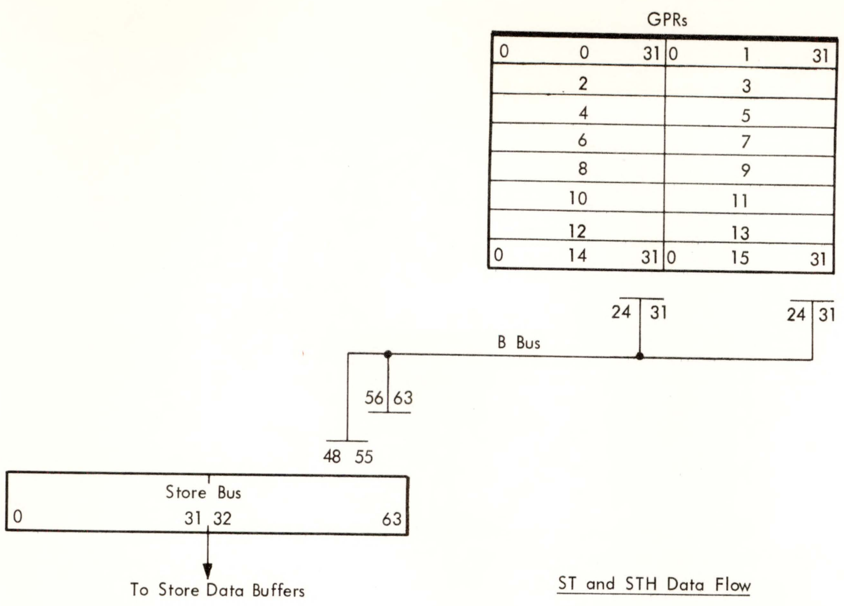
1. AND, OR, and exclusive OR operations gate data from the GPR's or storage (FXB) to connective circuits in the carry propagate adder.
2. The connective circuits perform all the AND, OR, and exclusive OR functions.
3. Results are placed back in the R1 GPR.
4. Normal execution time for ops 14, 16 and 17 (RR format) is 1 machine cycle.
5. Ops 54, 56 and 57 (RX format) require an undertermined number of cycles because input data must come from storage via an FXB.



Decode Op 14, 16, 17, 54, 56, 57
Block Op Condition (E55)
Yes
No
Hardware Error
Cycle 1 Trigger (D53)
Yes
No
Step Op NK339
End of Operation
Allow Setting Condition Code (CC)
FXOS Set CC Bit (E46-51)
Yes
No
Turn On FX Set CC Trigger (M55) NK167
Op Code 14, 16 54 or 56
Yes
No
Turn On and Trigger (K60) NK165
Set Up Controls for Connective Circuits AND and XOR Tgrs On Together Produce OR Condition
Op Code 16, 17 56 or 57
Yes
No
Turn On XOR Trigger (K61) NK165
Results Go to GPR Specified by R1 Field
1. Set R1 to GPR Sink Req (E56-63) NK169
2. Set R1 to B Bus NK165
Op Code 54, 56 or 57
A and B Buses Go to Connective Circuits in the CPA
Op Code 14, 16 or 17
No
Yes
Set R2 to A Bus NK135
Op Select FXB NK135
This Op Uses FXB
Set 32 Bit Outgate NK235
I Priority (G52)
Yes
No
Block Turn On of Data Available NK135
Wait for Data from Storage
Data Arrived In FXB
No
Yes
Decoded from FXOS Bits 16-18
1. Gate FXB to A (F56-58) NK601
2. Turn On FXB Signal Not Busy NNK337
Selected FXB Will Be Free During the Next Cycle
Both Operands Are Ready for Use
1. Set Data Avail (H46) NK335
2. Turn On Ingate GPR Trigger (H51) NK235
3. Turn On RUA Response (E67) NK167
Gate Results of Connective Circuits to GPR R1
GPR No Longer Needed by This Operation Can Be Turned On by RX End Op NK363
1. Step Op NK339
2. Ton Op Reset (H52) NK237
Begin Decoding Next Pseudo-Op
End of Operation

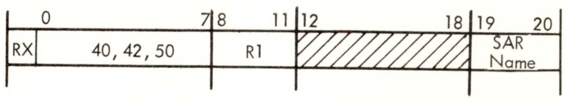
DIAGRAM 5-105. AND, OR, AND EXCLUSIVE OR

STC Data Flow



CC Remains Unchanged

40	STH	Store Halfword
42	STC	Store Character
50	ST	Store (Word)



1. Data is transferred from the GPR, specified by R1, to storage via the fixed area store bus and MSCE storage data buffers (SDB).

2. During a store operation, all 32 bits of R1 are placed in storage via bus positions 0-31 and 32-63.

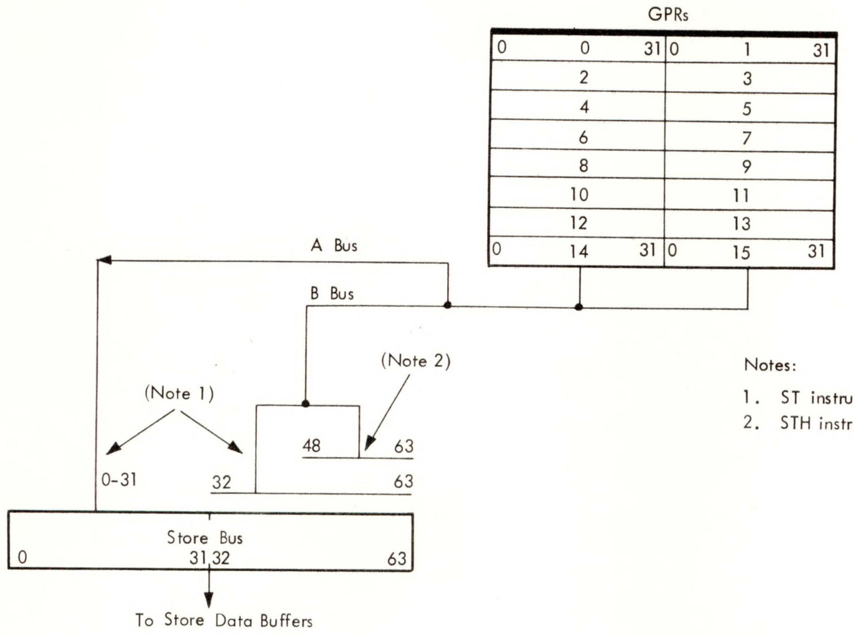
3. During a store halfword operation, bits 16-31 of R1 are placed in storage via bus positions 48-63.

4. During a store character operation, bits 24-31 of R1 are placed in storage via bus positions 48-55 and 56-63.

5. I-unit notifies MSCE what portion of the SDB data is to be stored.

6. Normal execution time is 1 machine cycle.

ST and STH Data Flow



Notes:

1. ST instruction only.
2. STH instruction only.

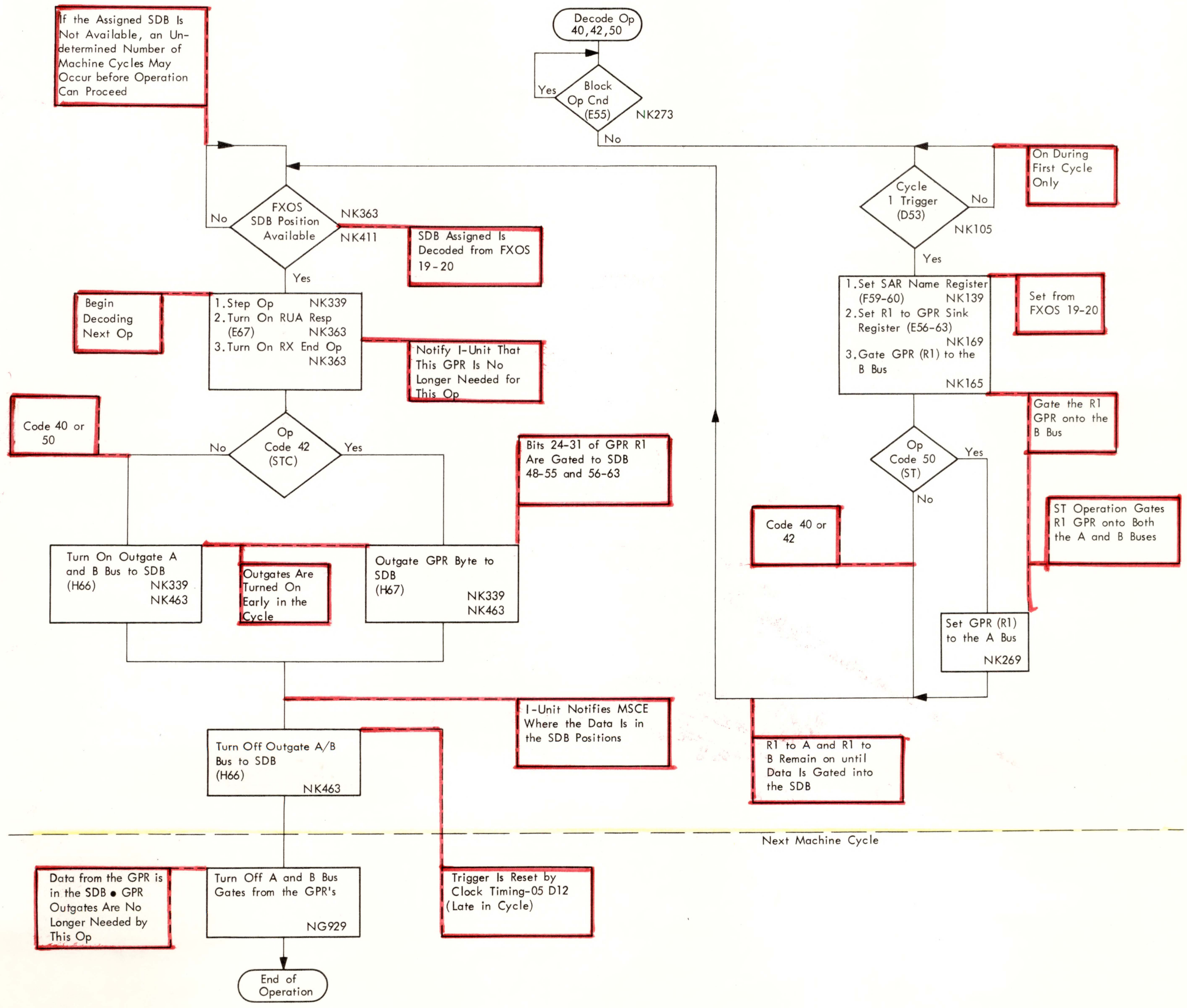
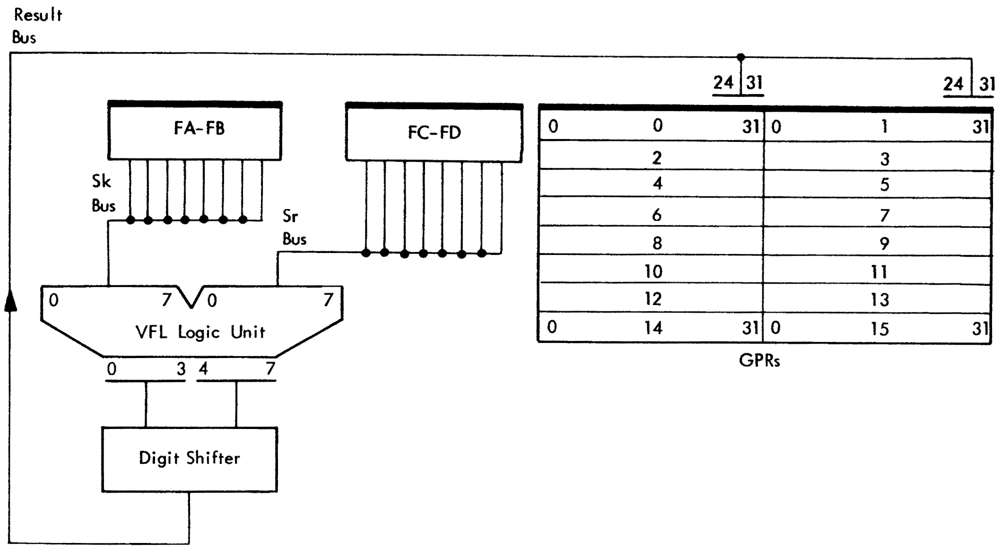
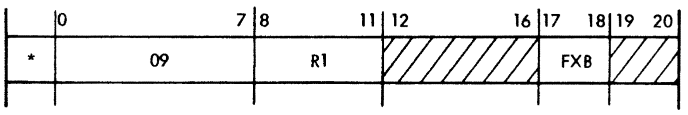

Decode Op 40,42,50
Block Op Cnd (E55)
Yes
No
NK273
If the Assigned SDB Is Not Available, an Undetermined Number of Machine Cycles May Occur before Operation Can Proceed
FXOS SDB Position Available
No
Yes
NK363
NK411
SDB Assigned Is Decoded from FXOS 19-20
Begin Decoding Next Op
1. Step Op NK339
2. Turn On RUA Resp (E67) NK363
3. Turn On RX End Op NK363
Notify I-Unit That This GPR Is No Longer Needed for This Op
Code 40 or 50
Op Code 42 (STC)
No
Yes
Bits 24-31 of GPR R1 Are Gated to SDB 48-55 and 56-63
Turn On Outgate A and B Bus to SDB (H66) NK339 NK463
Outgates Are Turned On Early in the Cycle
Outgate GPR Byte to SDB (H67) NK339 NK463
I-Unit Notifies MSCE Where the Data Is in the SDB Positions
Turn Off Outgate A/B Bus to SDB (H66) NK463
Next Machine Cycle
Data from the GPR is in the SDB • GPR Outgates Are No Longer Needed by This Op
Turn Off A and B Bus Gates from the GPR's NG929
Trigger Is Reset by Clock Timing-05 D12 (Late in Cycle)
End of Operation
Cycle 1 Trigger (D53)
No
Yes
NK105
On During First Cycle Only
1. Set SAR Name Register (F59-60) NK139
2. Set R1 to GPR Sink Register (E56-63) NK169
3. Gate GPR (R1) to the B Bus NK165
Set from FXOS 19-20
Gate the R1 GPR onto the B Bus
Op Code 50 (ST)
Yes
No
ST Operation Gates R1 GPR onto Both the A and B Buses
Code 40 or 42
Set GPR (R1) to the A Bus NK269
R1 to A and R1 to B Remain on until Data Is Gated into the SDB

DIAGRAM 5-106. STORE, STORE HALFWORD AND STORE CHARACTER



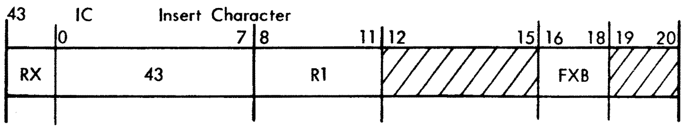
Condition Code
Remains Unchanged

09 ISK Insert Storage Key



* Special RR Format

43 IC Insert Character



1. Insert operations move a byte of information from storage into bits 24-31 of GPR R1.

2. ISK obtains the information from storage protect memory (a 5 bit storage key).

3. IC fetches a byte from a storage address.

4. Data path for the information is from storage through the FXB, VFL logic unit, digit shifter (straight through), and the result bus into GPR R1.

5. Insert ops require an undetermined number of machine cycles because data must come from storage.

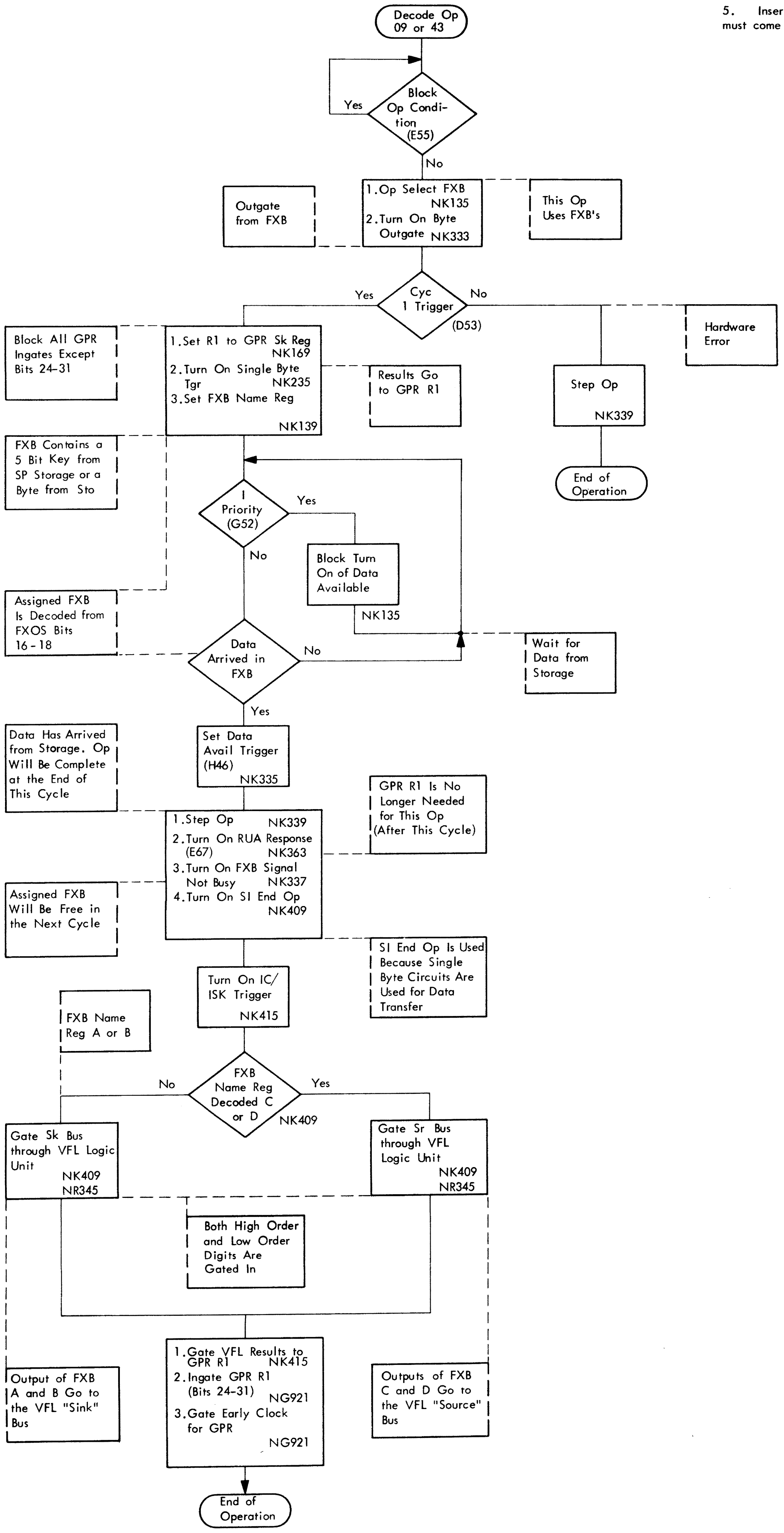
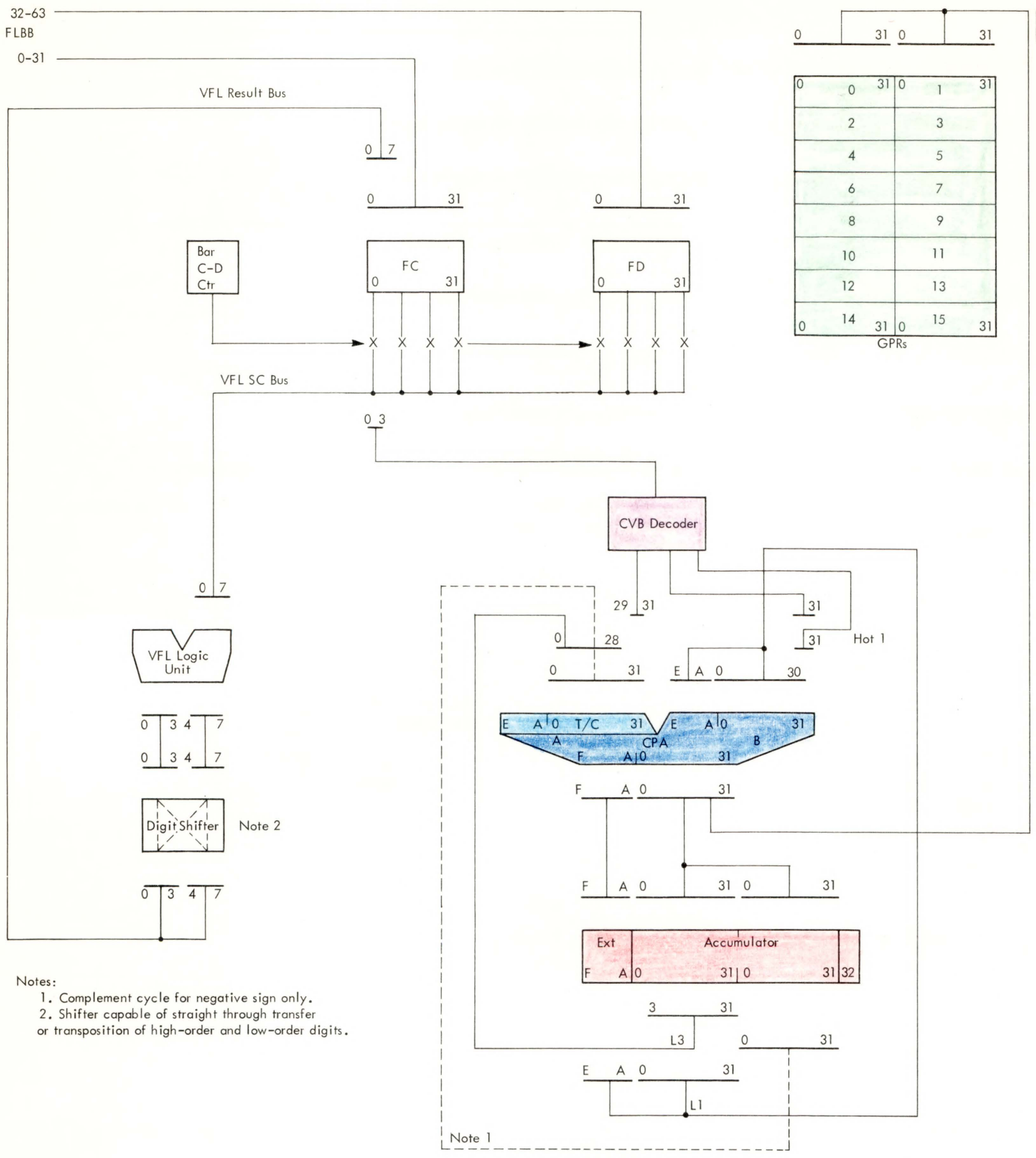


DIAGRAM 5-107. INSERT

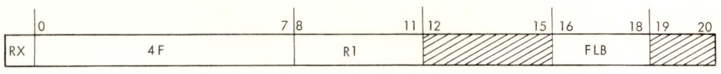
CVB Data Flow



Notes:
1. Complement cycle for negative sign only.
2. Shifter capable of straight through transfer or transposition of high-order and low-order digits.

The Condition Remains Unchanged for CVB

4 F CVB Convert to Binary



Objectives:

1. Convert to Binary Processes a doubleword of packed decimal information from storage and converts the information to Binary.
2. The doubleword from storage is held temporarily in an FLB in the floating point unit before being transferred to FXB C and D in the fixed point unit.
3. Each digit (4 bits) of the doubleword is moved in turn into the high order 4 bits of FXB C only, decoded and added into the binary value in the accumulator.
4. As each digit is decoded and added in, the binary value is multiplied by 10_{10} to simulate shifting 1 decimal position. The high order digit is decoded first so it will be multiplied by 10 (or shifted 1) for each following iteration (maximum of 15).
5. The largest number that can be held in packed decimal form in a doubleword would produce a binary number larger than a 32 bit register can hold. An overflow detection circuit monitors left accumulator positions 0, 1 and 2 to detect this condition. When results are too big to be stored in a GPR, FXP Divide Interrupt occurs.
6. If any digit in the source doubleword (except sign) contains a number higher than 9 (hexidecimal A to F) the operation will abort and cause an interrupt.
7. At the end of the operation the sign digit of the double word is examined. If the sign digit is a legitimate negative, the binary value is complemented to a negative form.
8. The resulting binary number is placed in the GPR specified by the R1 field of the pseudo-op.

Digit Value	CPA A Bit Positions 29	30	31	CPA B Bit Position 31	CPA B Hot 1 Bit 31
0	0	0	0	0	0
1	0	0	1	0	0
2	0	1	0	0	0
3	0	1	1	0	0
4	1	0	0	0	0
5	1	0	1	0	0
6	1	1	0	0	0
7	1	1	1	0	0
8	1	1	1	1	0
9	1	1	1	1	1

CVB Decoder Chart

Digit values are decoded and added into the binary result in the accumulator during each iteration of a CVB operation (objective 3).

GROUP 1 ≤ 7
GROUP 2 = 8
GROUP 3 = 9

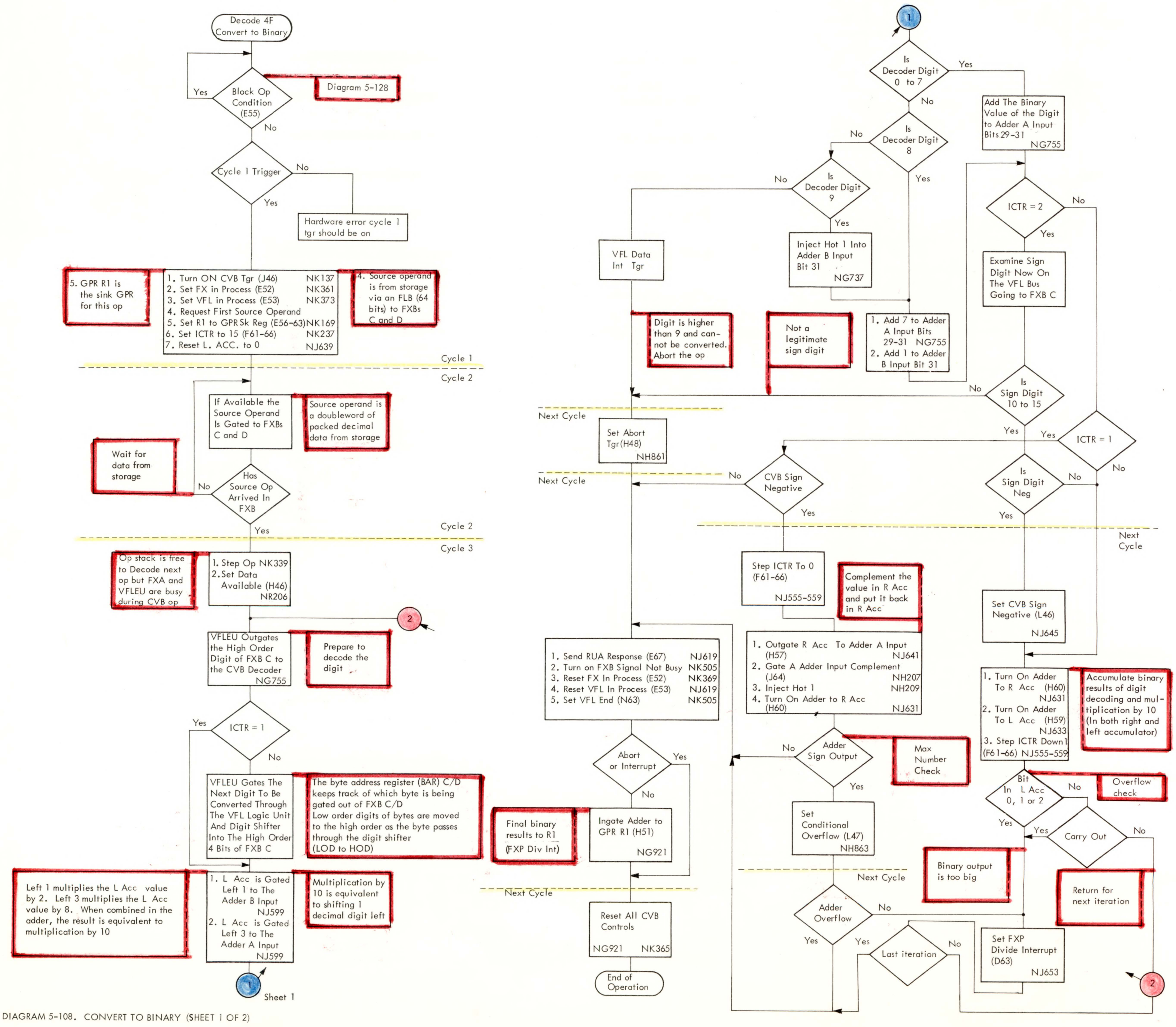
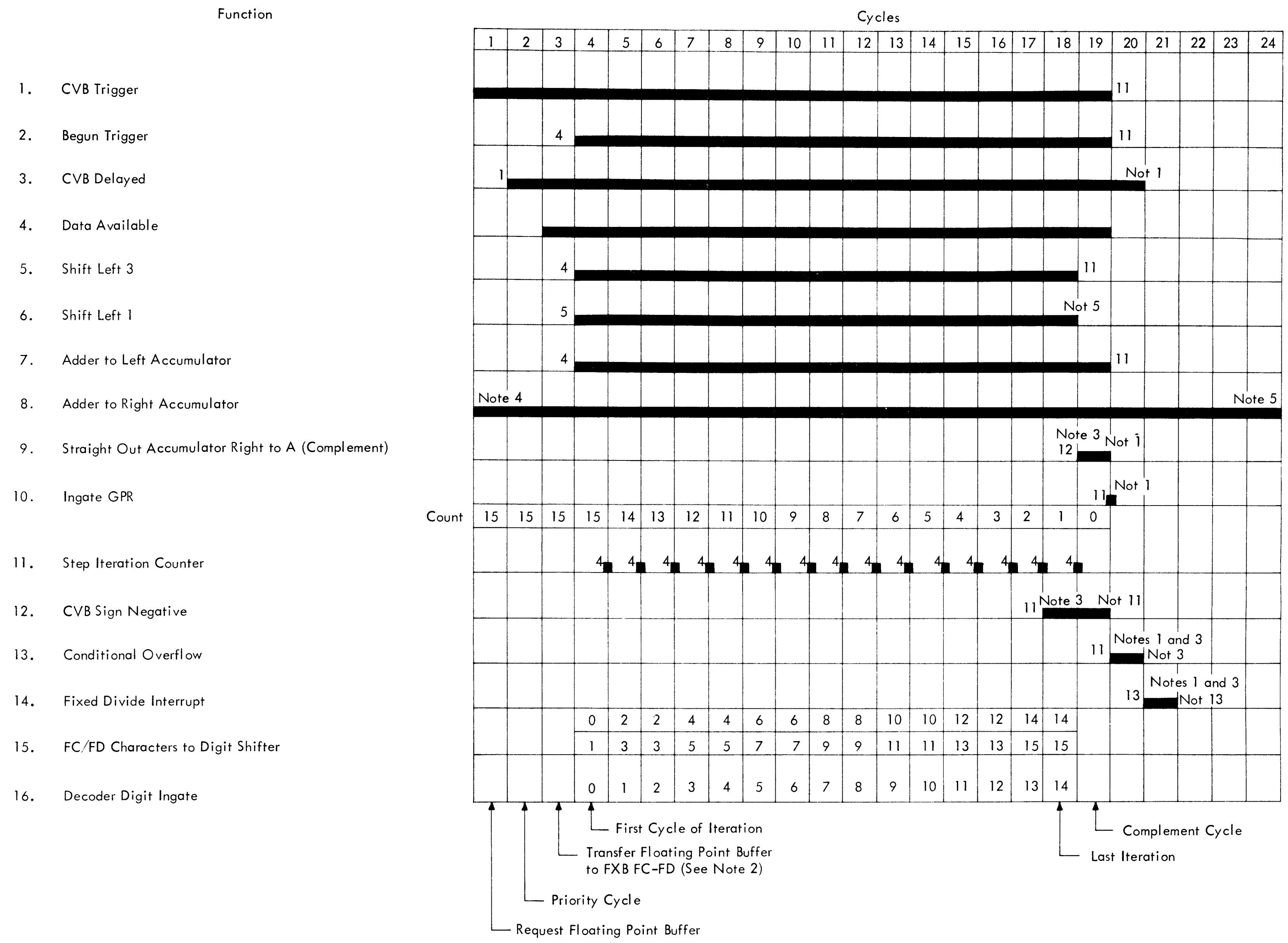
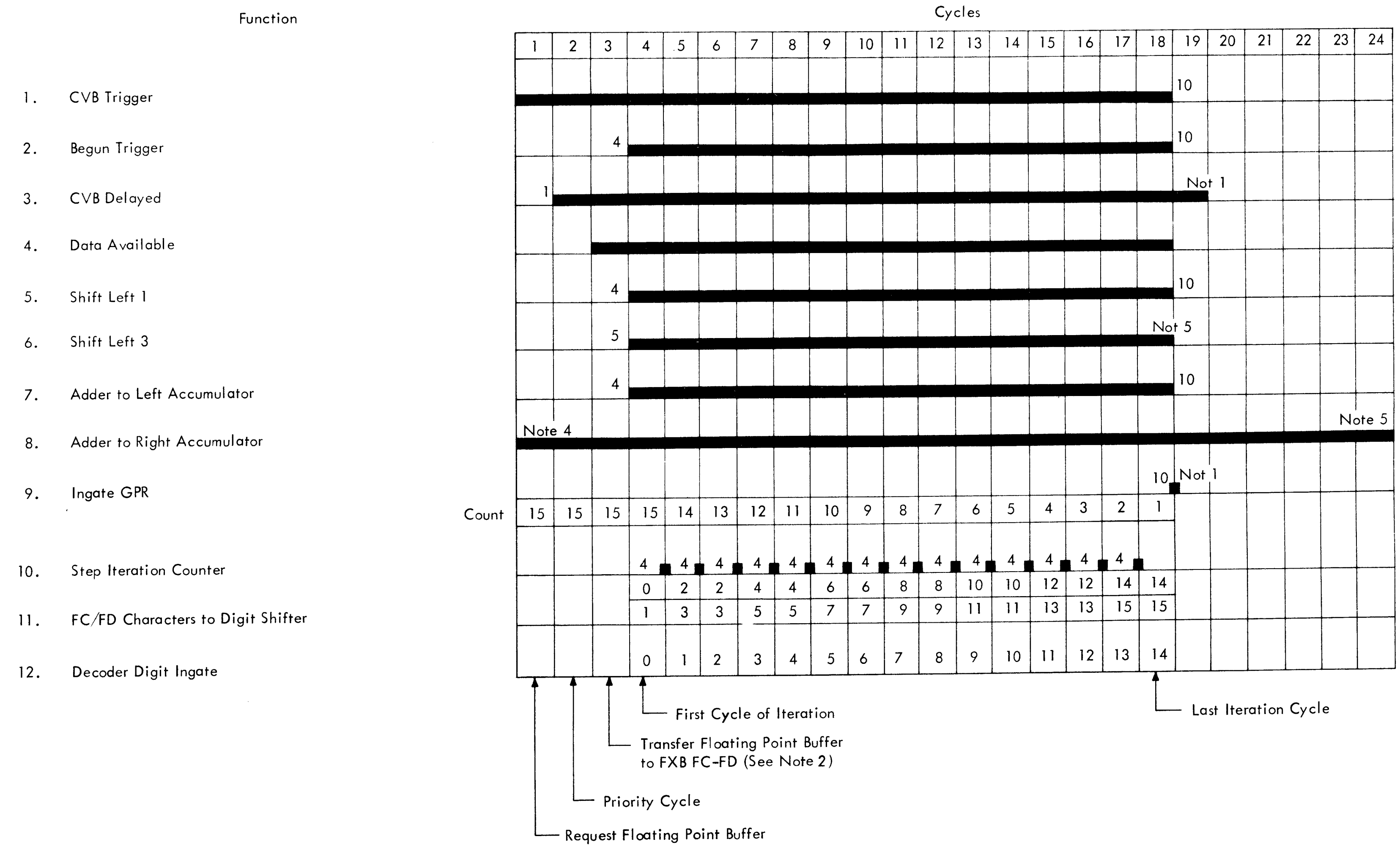

Decode 4F
Convert to Binary
Yes
Block Op Condition (E55)
No
Diagram 5-128
Cycle 1 Trigger
No
Yes
Hardware error cycle 1 tgr should be on
5. GPR R1 is the sink GPR for this op
1. Turn ON CVB Tgr (J46) NK137
2. Set FX in Process (E52) NK361
3. Set VFL in Process (E53) NK373
4. Request First Source Operand
5. Set R1 to GPR Sk Reg (E56-63) NK169
6. Set ICTR to 15 (F61-66) NK237
7. Reset L. ACC. to 0 NJ639
4. Source operand is from storage via an FLB (64 bits) to FXBs C and D
Cycle 1
Cycle 2
If Available the Source Operand Is Gated to FXBs C and D
Source operand is a doubleword of packed decimal data from storage
Wait for data from storage
No
Has Source Op Arrived In FXB
Yes
Cycle 2
Cycle 3
Op stack is free to Decode next op but FXA and VFLEU are busy during CVB op
1. Step Op NK339
2. Set Data Available (H46) NR206
2
VFLEU Outgates the High Order Digit of FXB C to the CVB Decoder NG755
Prepare to decode the digit
Yes
ICTR = 1
No
VFLEU Gates The Next Digit To Be Converted Through The VFL Logic Unit And Digit Shifter Into The High Order 4 Bits of FXB C
The byte address register (BAR) C/D keeps track of which byte is being gated out of FXB C/D
Low order digits of bytes are moved to the high order as the byte passes through the digit shifter
(LOD to HOD)
Left 1 multiplies the L Acc value by 2. Left 3 multiplies the L Acc value by 8. When combined in the adder, the result is equivalent to multiplication by 10
1. L Acc is Gated Left 1 to The Adder B Input NJ599
2. L Acc is Gated Left 3 to The Adder A Input NJ599
Multiplication by 10 is equivalent to shifting 1 decimal digit left
1
Sheet 1
1
Is Decoder Digit 0 to 7
Yes
No
Add The Binary Value of the Digit to Adder A Input Bits 29-31 NG755
No
Is Decoder Digit 8
Yes
No
Is Decoder Digit 9
Yes
VFL Data Int Tgr
Inject Hot 1 Into Adder B Input Bit 31 NG737
ICTR = 2
No
Yes
Examine Sign Digit Now On The VFL Bus Going to FXB C
Digit is higher than 9 and can-not be converted. Abort the op
Not a legitimate sign digit
1. Add 7 to Adder A Input Bits 29-31 NG755
2. Add 1 to Adder B Input Bit 31
No
Is Sign Digit 10 to 15
Yes
Next Cycle
Set Abort Tgr (H48) NH861
Yes
ICTR = 1
No
Next Cycle
No
CVB Sign Negative
Yes
Is Sign Digit Neg
No
Yes
Next Cycle
Step ICTR To 0 (F61-66) NJ555-559
Complement the value in R Acc and put it back in R Acc
Set CVB Sign Negative (L46) NJ645
1. Send RUA Response (E67) NJ619
2. Turn on FXB Signal Not Busy NK505
3. Reset FX In Process (E52) NK369
4. Reset VFL In Process (E53) NJ619
5. Set VFL End (N63) NK505
1. Outgate R Acc To Adder A Input (H57) NJ641
2. Gate A Adder Input Complement (J64) NH207
3. Inject Hot 1 NH209
4. Turn On Adder to R Acc (H60) NJ631
1. Turn On Adder To R Acc (H60) NJ631
2. Turn On Adder To L Acc (H59) NJ633
3. Step ICTR Down 1 (F61-66) NJ555-559
Accumulate binary results of digit decoding and mul-tiplication by 10 (In both right and left accumulator)
Abort or Interrupt
Yes
No
No
Adder Sign Output
Yes
Max Number Check
Bit In L Acc 0, 1 or 2
No
Overflow check
Yes
Final binary results to R1 (FXP Div Int)
Ingate Adder to GPR R1 (H51) NG921
Set Conditional Overflow (L47) NH863
Yes
Carry Out
No
Binary output is too big
Return for next iteration
Next Cycle
Next Cycle
Reset All CVB Controls
NG921 NK365
Adder Overflow
No
Yes
Yes
Last iteration
No
Set FXP Divide Interrupt (D63) NJ653
2
End of Operation

DIAGRAM 5-108. CONVERT TO BINARY (SHEET 1 OF 2)

CVB Timing, Sign Minus



CVB Timing, Sign Plus



Notes:

1. These triggers are turned on during last possible case of a FXA divide interrupt (overflow situation).
2. This operation occurs if data is available in the floating point buffer; if data is not available, second logical cycle continues indefinitely until data becomes available.
3. Occurs only if operand is negative.
4. Depends on previous op.
5. Depends on following op.

DIAGRAM 5-108. CONVERT TO BINARY (SHEET 2 OF 2)

The Condition Code Remains Unchanged During CVD Operations

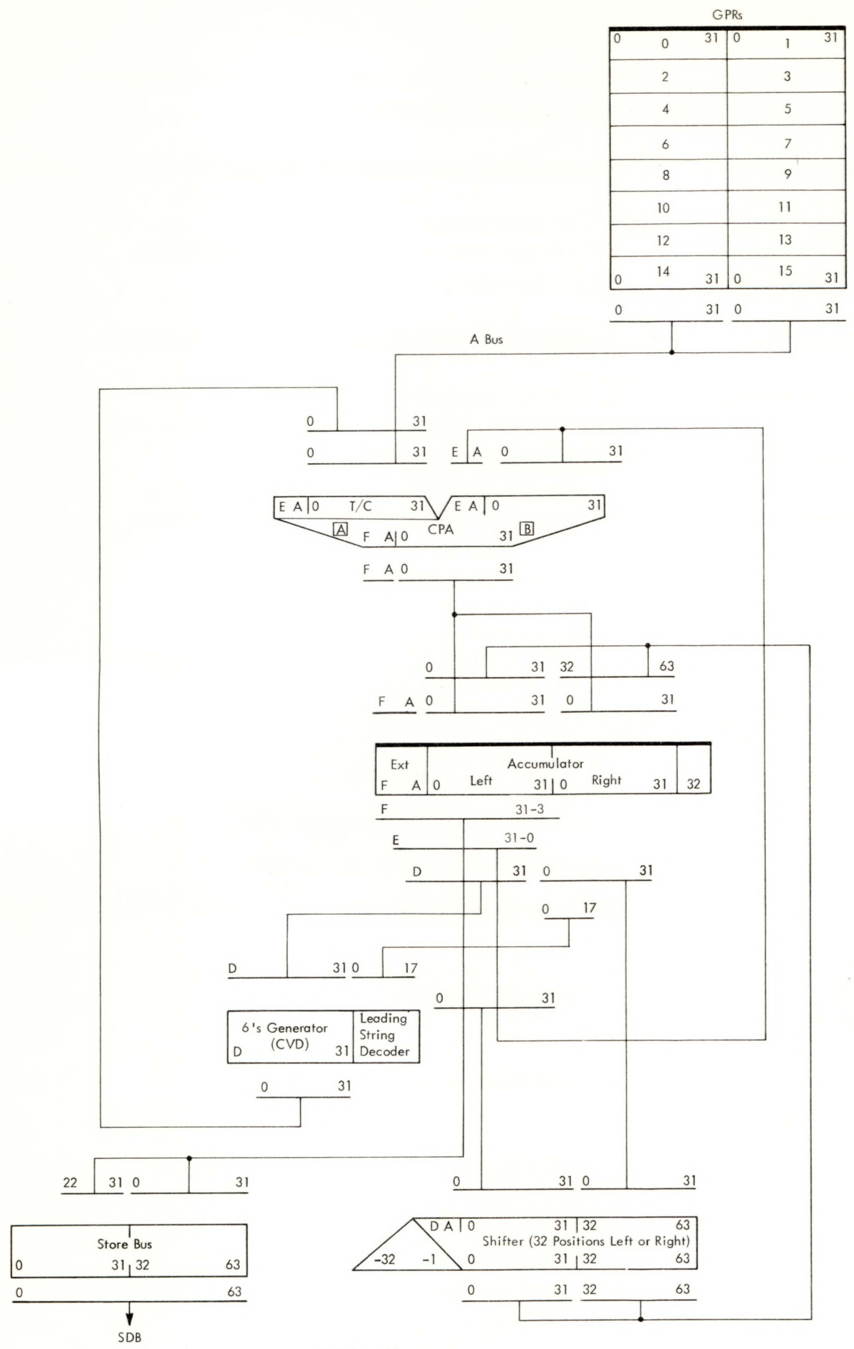
4E CVD Convert to Decimal

	0 7	8 11	12 18	19 20
RX	4E	R1	▨	SAR Name

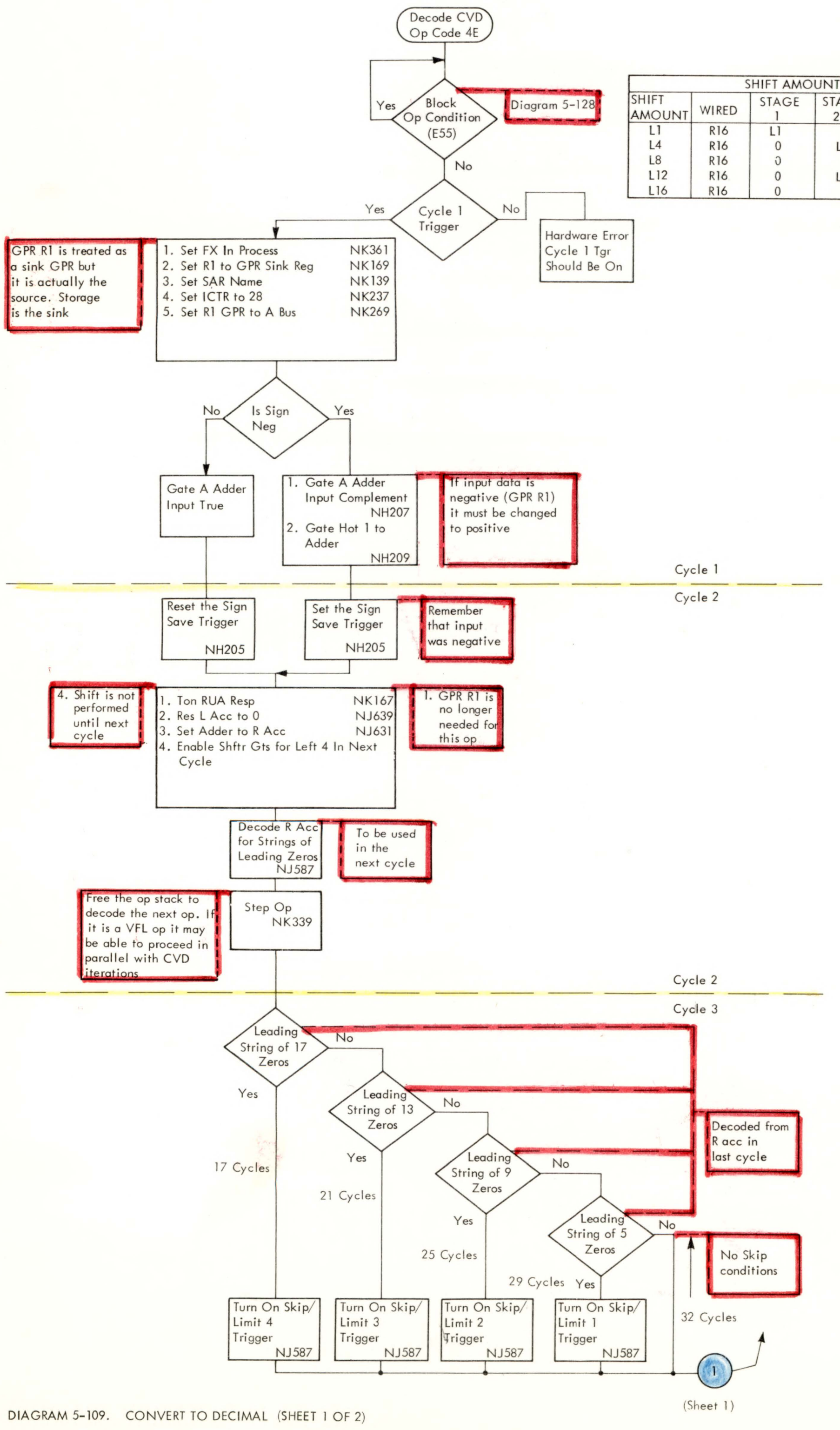
Objectives:

1. A CVD operation converts the data in GPR R1 from binary to packed decimal and stores the results via the SDB.
2. The address in the R1 field is normally a sink GPR but for CVD it is used as a source. Storage is the sink for CVD.
3. The sign of R1 is stored in the sign save trigger. The sign digit of the converted data is set to agree with the sign save trigger and format specified by PSW 12 (PSW 12 ON for ASCII; OFF for EBCDIC)
4. Conversion is done by decoding the output of the accumulator and inserting a 6 in the adder for each digit position that contains 5 or greater. The result will be a 38 bit decimal output from the adder back to the accumulator. (with sign bits the result is 42 bits)
5. The 42 bit result is transferred to storage in a doubleword (64 bits) via the SDB.
6. Timing of CVD depends on the number of leading zeros in the high order of R1 as follows:

Leading Zeros	Iterations	Total Cycles
0–4	No Skip = 28	32
5–8	Skip 4 = 24	29
9–12	Skip 8 = 20	25
13–16	Skip 12 = 16	21
17 or More	Skip 16 = 12	17



CVD Data Flow


Decode CVD Op Code 4E
Block Op Condition (E55)
Yes
No
Diagram 5-128
Cycle 1 Trigger
Yes
No
Hardware Error Cycle 1 Tgr Should Be On
GPR R1 is treated as a sink GPR but it is actually the source. Storage is the sink
1. Set FX In Process NK361
2. Set R1 to GPR Sink Reg NK169
3. Set SAR Name NK139
4. Set ICTR to 28 NK237
5. Set R1 GPR to A Bus NK269
Is Sign Neg
No
Yes
Gate A Adder Input True
1. Gate A Adder Input Complement NH207
2. Gate Hot 1 to Adder NH209
If input data is negative (GPR R1) it must be changed to positive
Cycle 1
Cycle 2
Reset the Sign Save Trigger NH205
Set the Sign Save Trigger NH205
Remember that input was negative
4. Shift is not performed until next cycle
1. Ton RUA Resp NK167
2. Res L Acc to 0 NJ639
3. Set Adder to R Acc NJ631
4. Enable Shftr Gts for Left 4 In Next Cycle
1. GPR R1 is no longer needed for this op
Decode R Acc for Strings of Leading Zeros NJ587
To be used in the next cycle
Free the op stack to decode the next op. If it is a VFL op it may be able to proceed in parallel with CVD iterations
Step Op NK339
Cycle 2
Cycle 3
Leading String of 17 Zeros
No
Yes
17 Cycles
Leading String of 13 Zeros
No
Yes
21 Cycles
Leading String of 9 Zeros
No
Yes
25 Cycles
Leading String of 5 Zeros
No
Yes
29 Cycles
Decoded from R acc in last cycle
No Skip conditions
32 Cycles
Turn On Skip/ Limit 4 Trigger NJ587
Turn On Skip/ Limit 3 Trigger NJ587
Turn On Skip/ Limit 2 Trigger NJ587
Turn On Skip/ Limit 1 Trigger NJ587
1
(Sheet 1)

SHIFT AMOUNTS IN CVD					
SHIFT AMOUNT	WIRED	STAGE 1	STAGE 2	STAGE 3	STAGE 4
L1	R16	L1	0	L16	0
L4	R16	0	L4	L16	0
L8	R16	0	0	0	L24
L12	R16	0	L4	0	L24
L16	R16	0	0	L8	L24

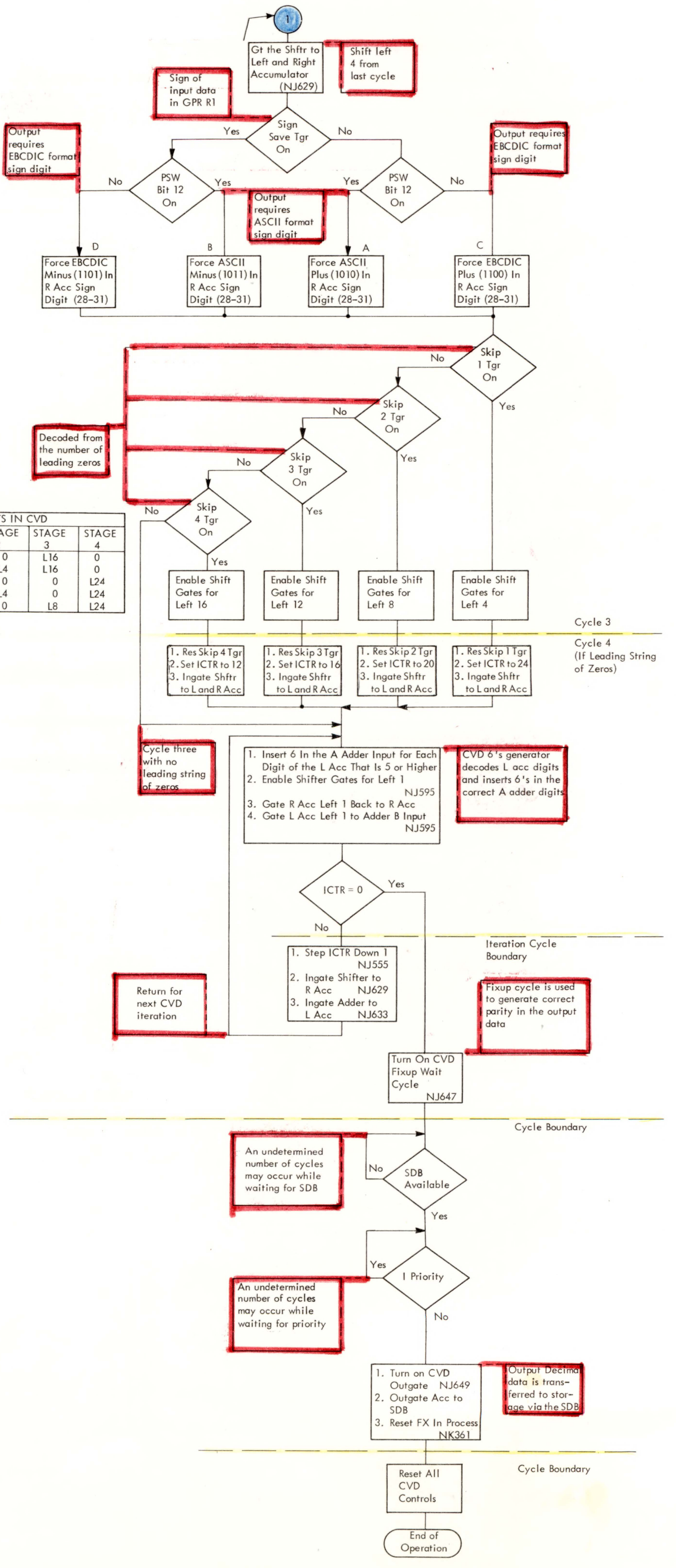
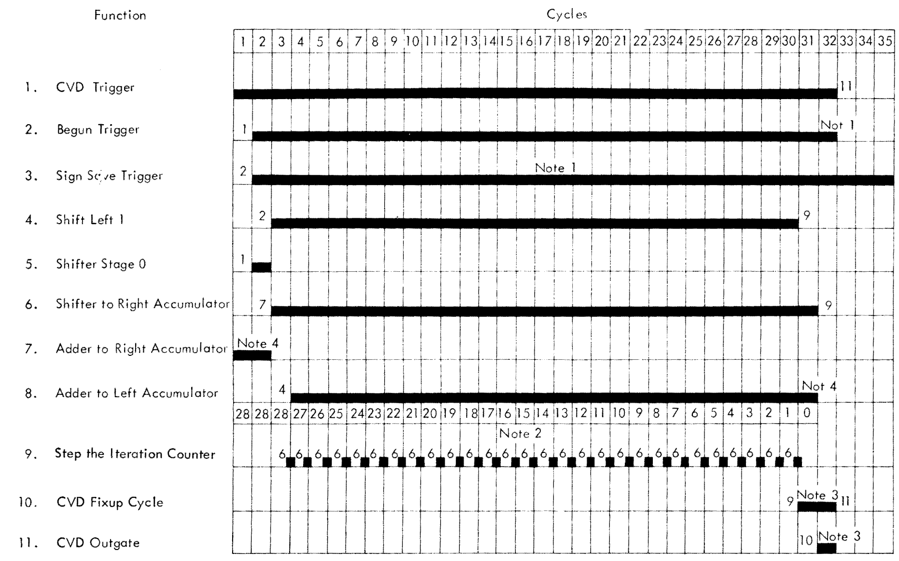
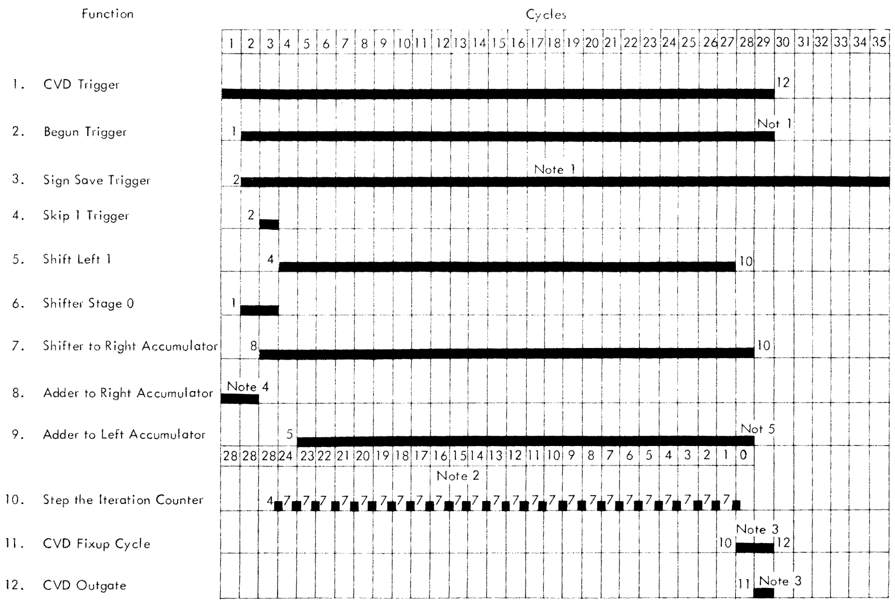

1
Gt the Shftr to Left and Right Accumulator (NJ629)
Shift left 4 from last cycle
Sign of input data in GPR R1
Sign Save Tgr On
Yes
No
Output requires EBCDIC format sign digit
PSW Bit 12 On
No
Yes
Output requires ASCII format sign digit
PSW Bit 12 On
Yes
No
Output requires EBCDIC format sign digit
D
Force EBCDIC Minus (1101) In R Acc Sign Digit (28–31)
B
Force ASCII Minus (1011) In R Acc Sign Digit (28–31)
A
Force ASCII Plus (1010) In R Acc Sign Digit (28–31)
C
Force EBCDIC Plus (1100) In R Acc Sign Digit (28–31)
Skip 1 Tgr On
No
Yes
Skip 2 Tgr On
No
Yes
Decoded from the number of leading zeros
Skip 3 Tgr On
No
Yes
Skip 4 Tgr On
No
Yes
Enable Shift Gates for Left 16
Enable Shift Gates for Left 12
Enable Shift Gates for Left 8
Enable Shift Gates for Left 4
Cycle 3
Cycle 4 (If Leading String of Zeros)
1. Res Skip 4 Tgr 2. Set ICTR to 12 3. Ingate Shftr to L and R Acc
1. Res Skip 3 Tgr 2. Set ICTR to 16 3. Ingate Shftr to L and R Acc
1. Res Skip 2 Tgr 2. Set ICTR to 20 3. Ingate Shftr to L and R Acc
1. Res Skip 1 Tgr 2. Set ICTR to 24 3. Ingate Shftr to L and R Acc
Cycle three with no leading string of zeros
1. Insert 6 In the A Adder Input for Each Digit of the L Acc That Is 5 or Higher
2. Enable Shifter Gates for Left 1 NJ595
3. Gate R Acc Left 1 Back to R Acc
4. Gate L Acc Left 1 to Adder B Input NJ595
CVD 6's generator decodes L acc digits and inserts 6's in the correct A adder digits
ICTR = 0
Yes
No
Iteration Cycle Boundary
1. Step ICTR Down 1 NJ555
2. Ingate Shifter to R Acc NJ629
3. Ingate Adder to L Acc NJ633
Return for next CVD iteration
Fixup cycle is used to generate correct parity in the output data
Turn On CVD Fixup Wait Cycle NJ647
Cycle Boundary
An undetermined number of cycles may occur while waiting for SDB
SDB Available
No
Yes
I Priority
Yes
No
An undetermined number of cycles may occur while waiting for priority
1. Turn on CVD Outgate NJ649
2. Outgate Acc to SDB
3. Reset FX In Process NK361
Output Decimal data is transferred to storage via the SDB
Cycle Boundary
Reset All CVD Controls
End of Operation

DIAGRAM 5-109. CONVERT TO DECIMAL (SHEET 1 OF 2)

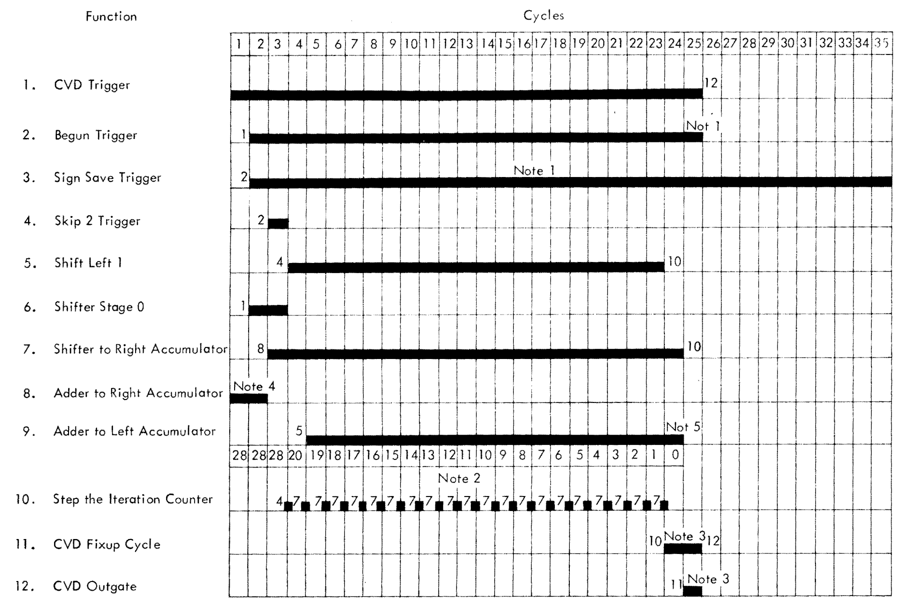
No Leading String



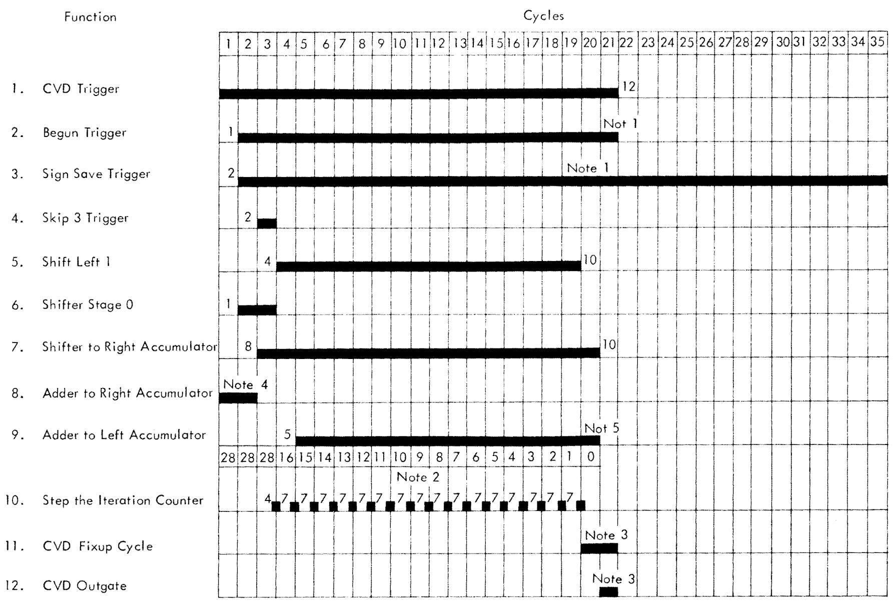
Leading String of 5



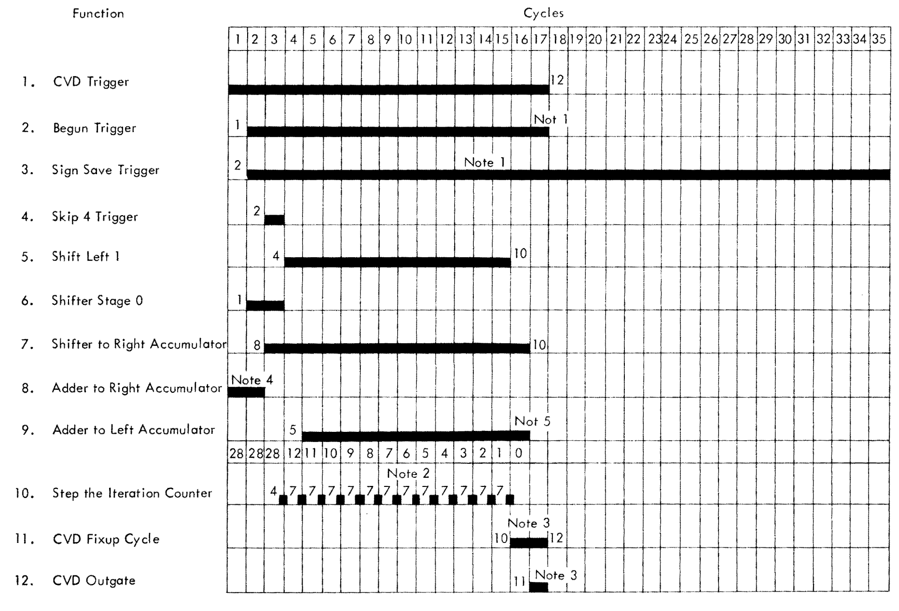
Leading String of 9



Leading String of 13



Leading String of 17



Notes:

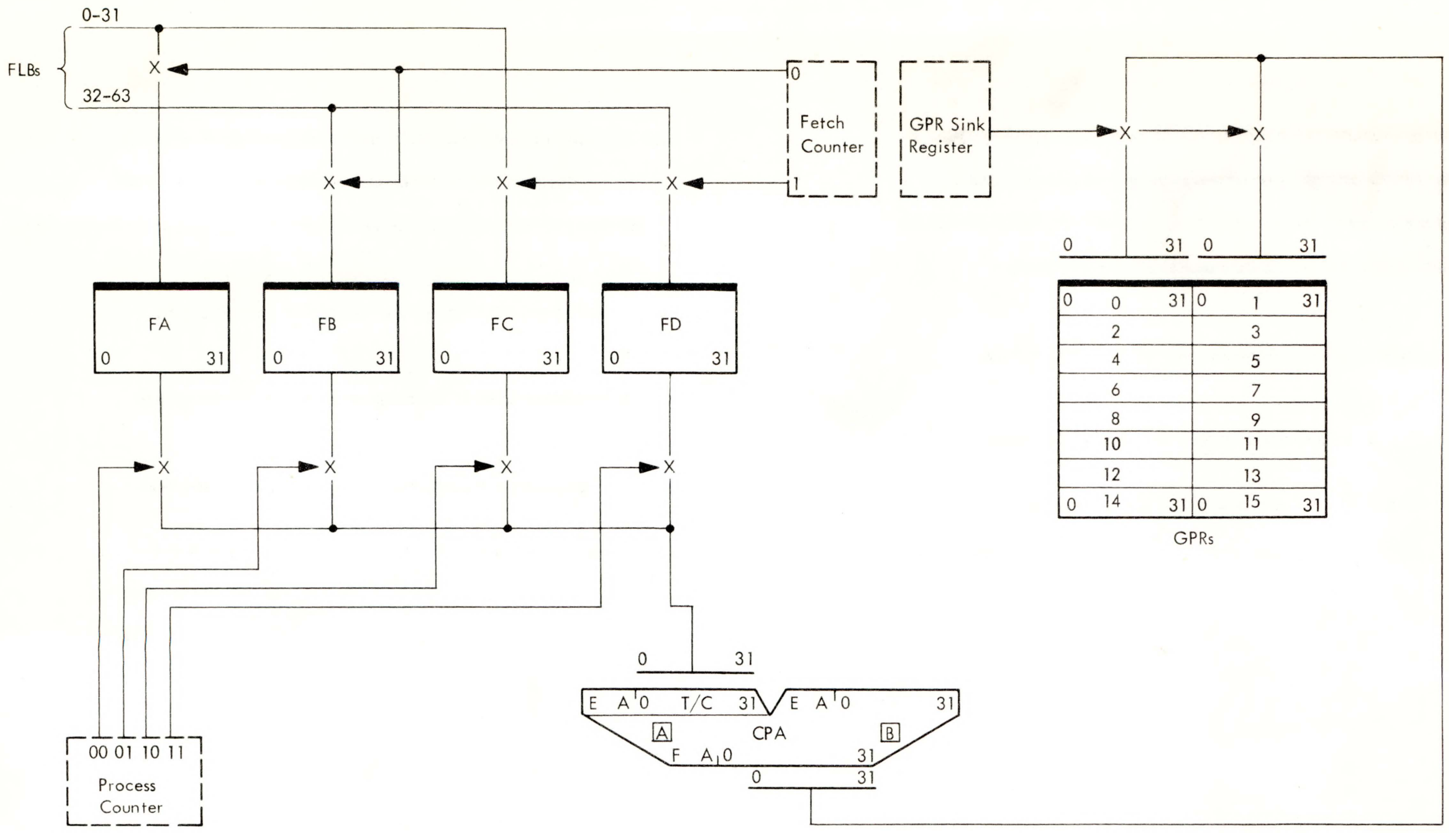
1. Conditional upon the sign of the number converted. Is turned on only for negative number and is not reset at end of operation. (Updating occurs only at beginning of an operation.)

2. The numbers show the counter value for each cycle. The counter is always initialized at 28. The first decrement (after step ICTR goes on) decrements the counter to the starting value appropriate for the leading string of zeros. All subsequent decrements are -1.

3. The starting time for CVD fixup is always as shown, however, CVD fixup cannot turn off until one cycle after CVD outgate is turned on. CVD outgate cannot turn on until at least one cycle after CVD fixup turns on and until an SDB is available. The representation of CVD fixup and CVD outgate shown here is the optimum case (SDB available immediately).

4. Depends on previous op.

DIAGRAM 5-109. CONVERT TO DECIMAL (SHEET 2 OF 2)



98 LM Load Multiple

		0 – 7	8 – 11	12	13	14	15	16 – 18	19 – 20
CC Remains Unchanged	RS	98	R1	L	R	S	E	FLB	▨

1. Load multiple transmits data from storage through the floating point buffers, the fixed buffers, and the adder, to the GPRs.

2. Any number of GPRs, from 1 to 16 can be loaded with one LM operation.

3. I-unit issues pseudo op to the fixed area for each doubleword of data fetched from storage.

4. Start (S) or end (E) bits indicate the first and last pseudo ops of the operation.

5. The first and last pseudo ops may transmit only one word of data. In these cases the left (L) and the right (R) bits indicate which half of the doubleword fetch is to be used.

6. FXB A-B and C-D are used as doubleword pairs to accept data from the specified FLB. FXB E and F are not used for LM.

7. FXB are outgated, one at a time, through the adder to each GPR in sequence.

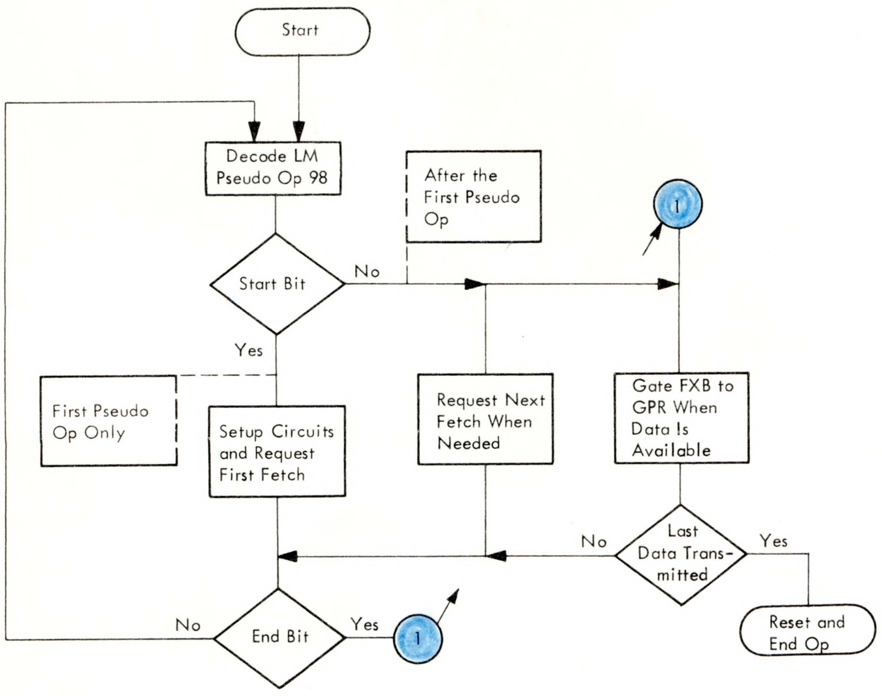
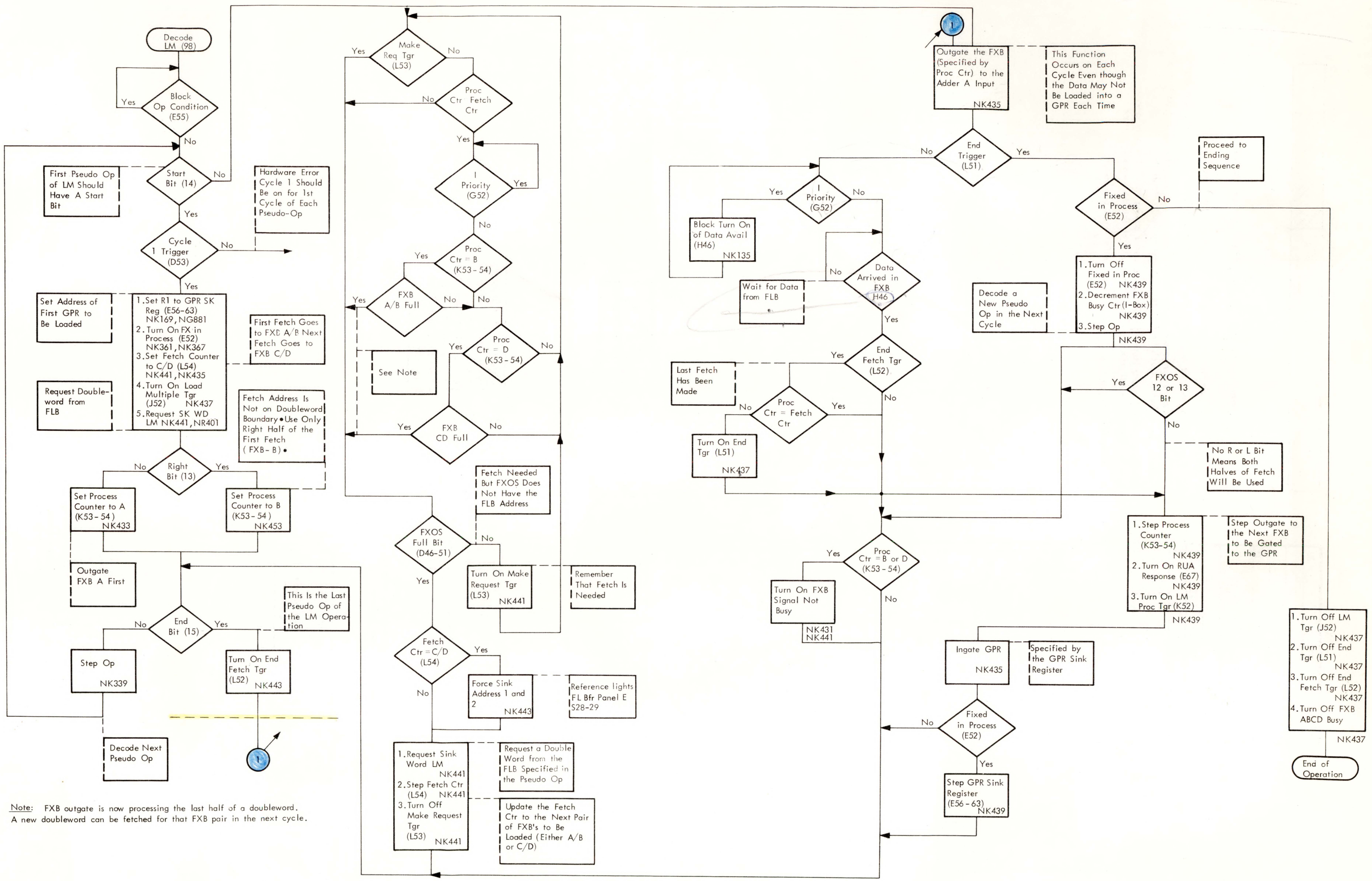

Start
Decode LM Pseudo Op 98
After the First Pseudo Op
Start Bit
No
Yes
First Pseudo Op Only
Setup Circuits and Request First Fetch
Request Next Fetch When Needed
1
Gate FXB to GPR When Data Is Available
Last Data Transmitted
No
Yes
Reset and End Op
End Bit
No
Yes
1

DIAGRAM 5-110. LOAD MULTIPLE (SHEET 1 OF 2)


Decode LM (98)
Block Op Condition (E55)
Yes
No
First Pseudo Op of LM Should Have A Start Bit
Start Bit (14)
No
Yes
Hardware Error Cycle 1 Should Be on for 1st Cycle of Each Pseudo-Op
Cycle 1 Trigger (D53)
No
Yes
Set Address of First GPR to Be Loaded
1. Set R1 to GPR SK Reg (E56-63) NK169, NG881
2. Turn On FX in Process (E52) NK361, NK367
3. Set Fetch Counter to C/D (L54) NK441, NK435
4. Turn On Load Multiple Tgr (J52) NK437
5. Request SK WD LM NK441, NR401
Request Double-word from FLB
First Fetch Goes to FXB A/B Next Fetch Goes to FXB C/D
Fetch Address Is Not on Doubleword Boundary • Use Only Right Half of the First Fetch (FXB-B) •
Right Bit (13)
No
Yes
Set Process Counter to A (K53-54) NK433
Set Process Counter to B (K53-54) NK453
Outgate FXB A First
This Is the Last Pseudo Op of the LM Operation
End Bit (15)
No
Yes
Step Op NK339
Turn On End Fetch Tgr (L52) NK443
Decode Next Pseudo Op
1
Make Req Tgr (L53)
Yes
No
Proc Ctr Fetch Ctr
No
Yes
I Priority (G52)
Yes
No
Proc Ctr = B (K53-54)
Yes
No
FXB A/B Full
Yes
No
See Note
Proc Ctr = D (K53-54)
Yes
No
FXB CD Full
Yes
No
Fetch Needed But FXOS Does Not Have the FLB Address
FXOS Full Bit (D46-51)
No
Yes
Turn On Make Request Tgr (L53) NK441
Remember That Fetch Is Needed
Fetch Ctr = C/D (L54)
Yes
No
Force Sink Address 1 and 2 NK443
Reference lights FL Bfr Panel E S28-29
1. Request Sink Word LM NK441
2. Step Fetch Ctr (L54) NK441
3. Turn Off Make Request Tgr (L53) NK441
Request a Double Word from the FLB Specified in the Pseudo Op
Update the Fetch Ctr to the Next Pair of FXB's to Be Loaded (Either A/B or C/D)
1
Outgate the FXB (Specified by Proc Ctr) to the Adder A Input NK435
This Function Occurs on Each Cycle Even though the Data May Not Be Loaded into a GPR Each Time
End Trigger (L51)
No
Yes
Proceed to Ending Sequence
I Priority (G52)
Yes
No
Block Turn On of Data Avail (H46) NK135
Data Arrived in FXB H46
No
Yes
Wait for Data from FLB
Fixed in Process (E52)
No
Yes
1. Turn Off Fixed in Proc (E52) NK439
2. Decrement FXB Busy Ctr (I-Box) NK439
3. Step Op
NK439
Decode a New Pseudo Op in the Next Cycle
End Fetch Tgr (L52)
Yes
No
Last Fetch Has Been Made
Proc Ctr = Fetch Ctr
No
Yes
Turn On End Tgr (L51) NK437
FXOS 12 or 13 Bit
Yes
No
No R or L Bit Means Both Halves of Fetch Will Be Used
1. Step Process Counter (K53-54) NK439
2. Turn On RUA Response (E67) NK439
3. Turn On LM Proc Tgr (K52)
NK439
Step Outgate to the Next FXB to Be Gated to the GPR
Proc Ctr = B or D (K53-54)
Yes
No
Turn On FXB Signal Not Busy
NK431 NK441
Ingate GPR NK435
Specified by the GPR Sink Register
Fixed in Process (E52)
No
Yes
Step GPR Sink Register (E56-63) NK439
1. Turn Off LM Tgr (J52) NK437
2. Turn Off End Tgr (L51) NK437
3. Turn Off End Fetch Tgr (L52) NK437
4. Turn Off FXB ABCD Busy
NK437
End of Operation

Note: FXB outgate is now processing the last half of a doubleword. A new doubleword can be fetched for that FXB pair in the next cycle.

DIAGRAM 5-110. LOAD MULTIPLE (SHEET 2 OF 2)

STM Store Multiple CC Remains Unchanged

	0 7	8 11	12	13	14 18	19 20
RS	90	R1	L	R	▨	SAR NAME

1. A block of data is transferred from the GPR's to storage via the fixed area storage bus.

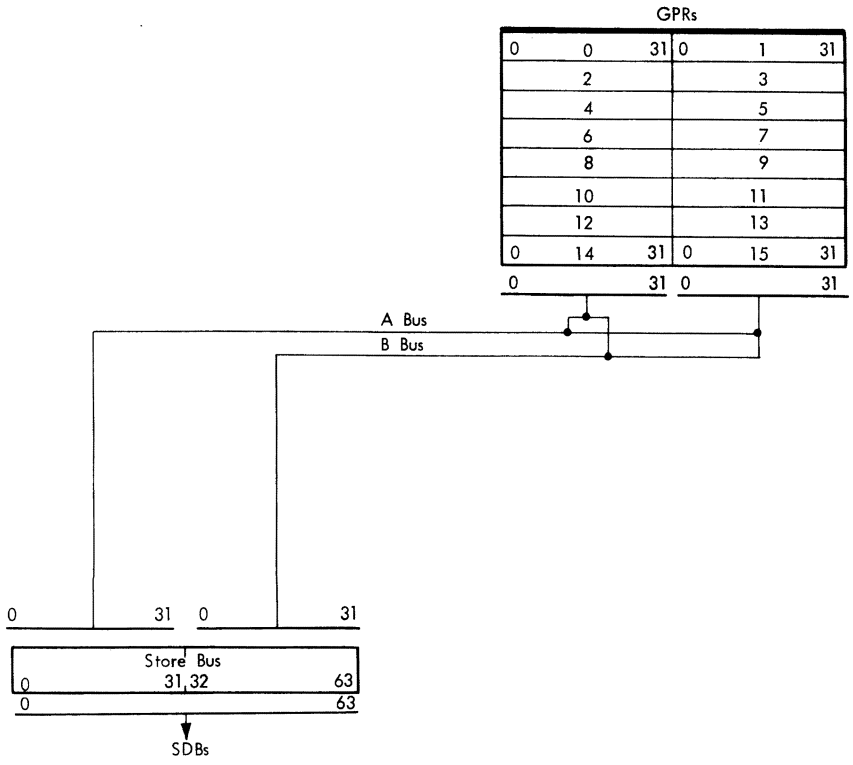
2. The R1 field specifies the first GPR of the block to be stored.

3. A pseudo-op will be issued, by the I-unit, for each doubleword of the data block.

4. Singleword stores can occur during the first and last pseudo-ops. The L (left) and R (right) bits indicate which half of the doubleword is to be stored.

5. During a singleword store two GPR's are gated out to MSCE but only one is actually stored. I-unit notifies MSCE which half of the doubleword to store.

6. Normal execution time is 1 machine cycle for each pseudo-op.



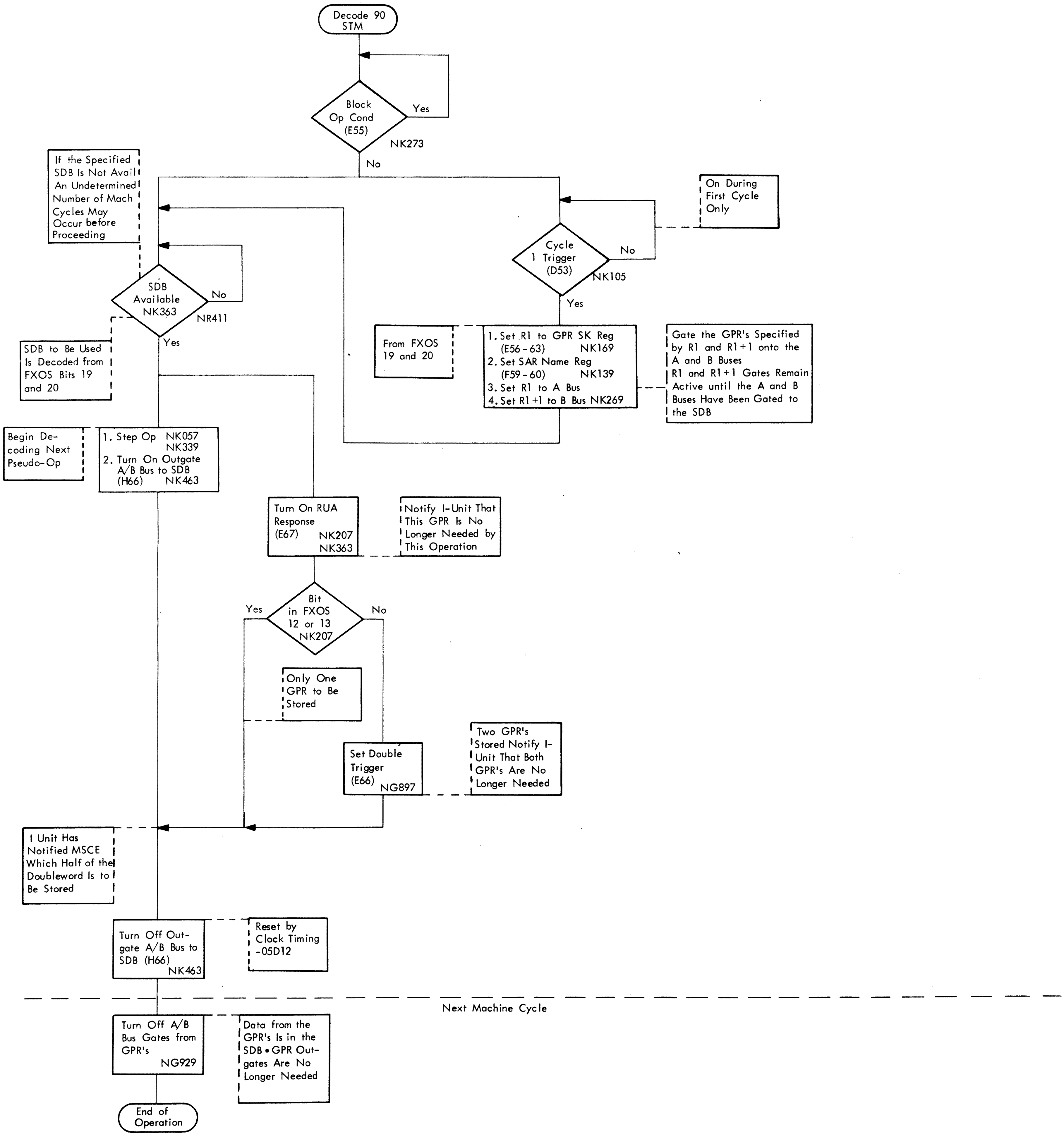
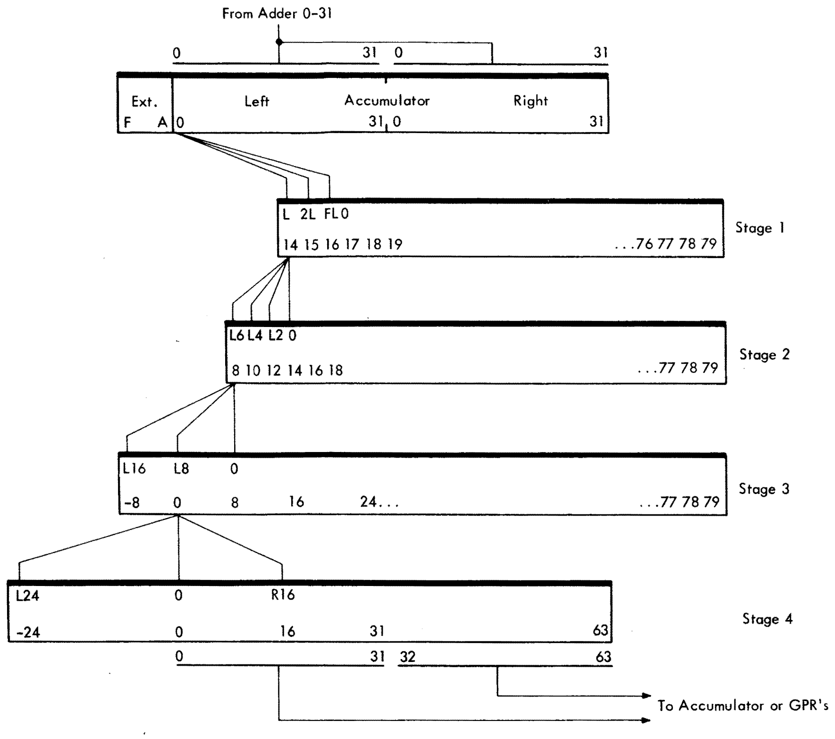

Decode 90 STM
Block Op Cond (E55)
Yes
NK273
No
If the Specified SDB Is Not Avail An Undetermined Number of Mach Cycles May Occur before Proceeding
On During First Cycle Only
Cycle 1 Trigger (D53)
No
NK105
Yes
SDB Available NK363
No
NR411
Yes
SDB to Be Used Is Decoded from FXOS Bits 19 and 20
From FXOS 19 and 20
1. Set R1 to GPR SK Reg (E56-63) NK169
2. Set SAR Name Reg (F59-60) NK139
3. Set R1 to A Bus
4. Set R1 +1 to B Bus NK269
Gate the GPR's Specified by R1 and R1+1 onto the A and B Buses
R1 and R1+1 Gates Remain Active until the A and B Buses Have Been Gated to the SDB
Begin Decoding Next Pseudo-Op
1. Step Op NK057 NK339
2. Turn On Outgate A/B Bus to SDB (H66) NK463
Turn On RUA Response (E67) NK207 NK363
Notify I-Unit That This GPR Is No Longer Needed by This Operation
Bit in FXOS 12 or 13 NK207
Yes
No
Only One GPR to Be Stored
Set Double Trigger (E66) NG897
Two GPR's Stored Notify I-Unit That Both GPR's Are No Longer Needed
I Unit Has Notified MSCE Which Half of the Doubleword Is to Be Stored
Turn Off Outgate A/B Bus to SDB (H66) NK463
Reset by Clock Timing -05D12
Next Machine Cycle
Turn Off A/B Bus Gates from GPR's NG929
Data from the GPR's Is in the SDB • GPR Outgates Are No Longer Needed
End of Operation

DIAGRAM 5-111. STORE MULTIPLE



Force 0s in:
Bits 74-79 on L6
Bits 76-79 on L4
Bits 78-79 on L2

Force 0s in bits 56-63 on L24

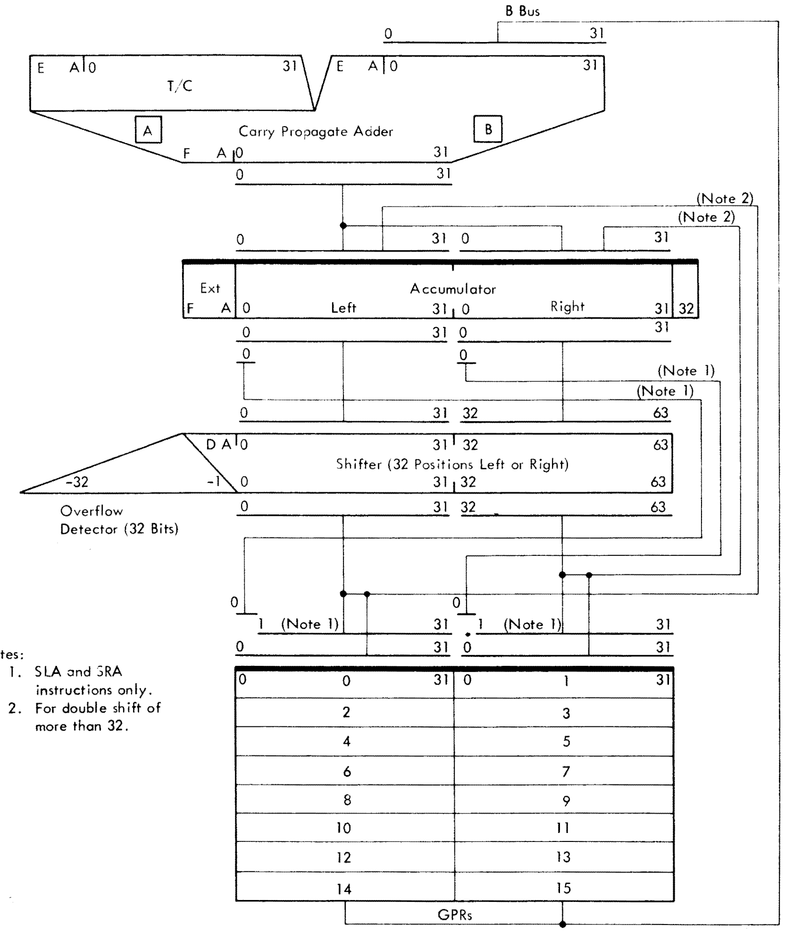
Left Shift

Shift Amount	Wired	Stage 1	Stage 2	Stage 3	Stage 4	Shift Amount	Wired	Stage 1	Stage 2	Stage 3	Stage 4
0	R16	L0	L0	L16	L0	16	R16	L0	L0	L8	L24
1	R16	L1	L0	L16	L0	17	R16	L1	L0	L8	L24
2	R16	L0	L2	L16	L0	18	R16	L0	L2	L8	L24
3	R16	L1	L2	L16	L0	19	R16	L1	L2	L8	L24
4	R16	L0	L4	L16	L0	20	R16	L0	L4	L8	L24
5	R16	L1	L4	L16	L0	21	R16	L1	L4	L8	L24
6	R16	L0	L6	L16	L0	22	R16	L0	L6	L8	L24
7	R16	L1	L6	L16	L0	23	R16	L1	L6	L8	L24
8	R16	L0	L0	L0	L24	24	R16	L0	L0	L16	L24
9	R16	L1	L0	L0	L24	25	R16	L1	L0	L16	L24
10	R16	L0	L2	L0	L24	26	R16	L0	L2	L16	L24
11	R16	L1	L2	L0	L24	27	R16	L1	L2	L16	L24
12	R16	L0	L4	L0	L24	28	R16	L0	L4	L16	L24
13	R16	L1	L4	L0	L24	29	R16	L1	L4	L16	L24
14	R16	L0	L6	L0	L24	30	R16	L0	L6	L16	L24
15	R16	L1	L6	L0	L24	31	R16	L1	L6	L16	L24
						32	R16	L2	L6	L16	L24

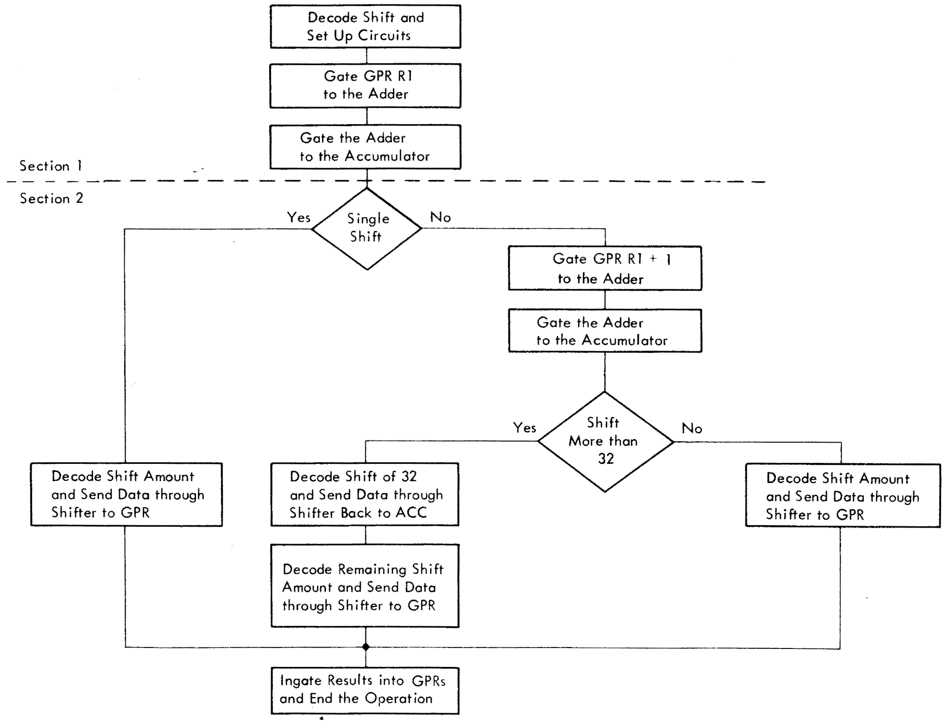
Right Shift

Shift Amount	Wired	Stage 1	Stage 2	Stage 3	Stage 4	Shift Amount	Wired	Stage 1	Stage 2	Stage 3	Stage 4
0	R16	L2	L6	L8	L0	16	R16	L2	L6	L8	R16
1	R16	L1	L6	L8	L0	17	R16	L1	L6	L8	R16
2	R16	L0	L6	L8	L0	18	R16	L0	L6	L8	R16
3	R16	L1	L4	L8	L0	19	R16	L1	L4	L8	R16
4	R16	L0	L4	L8	L0	20	R16	L0	L4	L8	R16
5	R16	L1	L2	L8	L0	21	R16	L1	L2	L8	R16
6	R16	L0	L2	L8	L0	22	R16	L0	L2	L8	R16
7	R16	L1	L0	L8	L0	23	R16	L1	L0	L8	R16
8	R16	L2	L6	L0	L0	24	R16	L2	L6	L0	R16
9	R16	L1	L6	L0	L0	25	R16	L1	L6	L0	R16
10	R16	L0	L6	L0	L0	26	R16	L0	L6	L0	R16
11	R16	L1	L4	L0	L0	27	R16	L1	L4	L0	R16
12	R16	L0	L4	L0	L0	28	R16	L0	L4	L0	R16
13	R16	L1	L2	L0	L0	29	R16	L1	L2	L0	R16
14	R16	L0	L2	L0	L0	30	R16	L0	L2	L0	R16
15	R16	L1	L0	L0	L0	31	R16	L1	L0	L0	R16
						32	R16	L0	L0	L0	R16

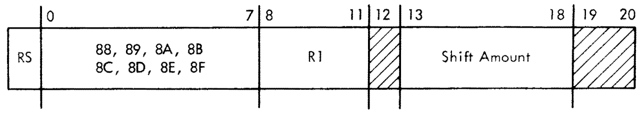
DIAGRAM 5-112. SHIFT INSTRUCTIONS (SHEET 1 OF 3)



Notes:
1. SLA and SRA instructions only.
2. For double shift of more than 32.


Decode Shift and Set Up Circuits
Gate GPR R1 to the Adder
Gate the Adder to the Accumulator
Section 1
Section 2
Yes
Single Shift
No
Gate GPR R1 + 1 to the Adder
Gate the Adder to the Accumulator
Yes
Shift More than 32
No
Decode Shift Amount and Send Data through Shifter to GPR
Decode Shift of 32 and Send Data through Shifter Back to ACC
Decode Shift Amount and Send Data through Shifter to GPR
Decode Remaining Shift Amount and Send Data through Shifter to GPR
Ingate Results into GPRs and End the Operation

88	SRL	Shift Right Single Logical
89	SLL	Shift Left Single Logical
8A	SRA	Shift Right Single (Arithmetic)
8B	SLA	Shift Left Single (Arithmetic)
8C	SRDL	Shift Right Double Logical
8D	SLDL	Shift Left Double Logical
8E	SRDA	Shift Right Double (Arithmetic)
8F	SLDA	Shift Left Double (Arithmetic)



Objectives:

1. Data in the GPR specified by R1 is shifted right or left the specified amount and replaced in the GPR.
2. Logical shifts move bits 0 to 31 as an unsigned value.
3. Arithmetic shifts move bits 1 to 31 and treat position 0 as a fixed sign.
4. Single shifts move the data from a single GPR.
5. Double shifts move the data from an even odd pair of GPRs as one value.
6. Data path for shift operations is from the GPRs through the carry propagate adder, the accumulator, and shifter back to the GPRs.
7. The shift amount generates gates to control the shifter.
8. Single shift operations are completed in two machine cycles with some ending functions performed in the first cycle of the following operation (Cycle 3).
9. Double shift operations are completed in three machine cycles if the shift amount is more than 32. Some ending functions are performed in the first cycle of the following operation (Cycles 4 or 5).

Condition code remains unchanged for logical shift operations

Condition code is set as follows for arithmetic shift operations:

CC	Condition
0	Result is zero
1	Result is less than zero
2	Result is greater than zero
3	Overflow * (SLA and SLDA only)

* Overflow detection circuits are attached to the left end of the shifter. (See data flow)

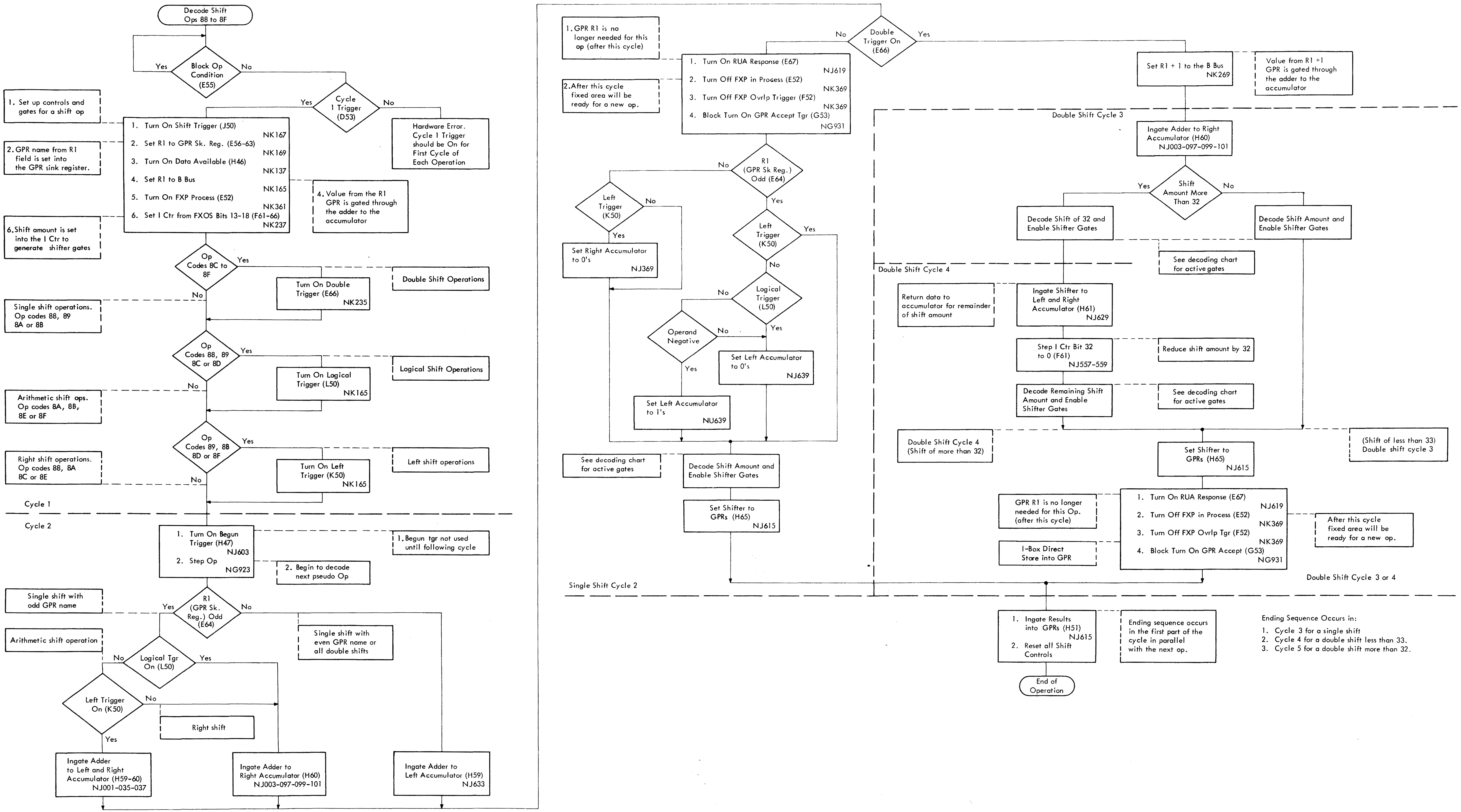
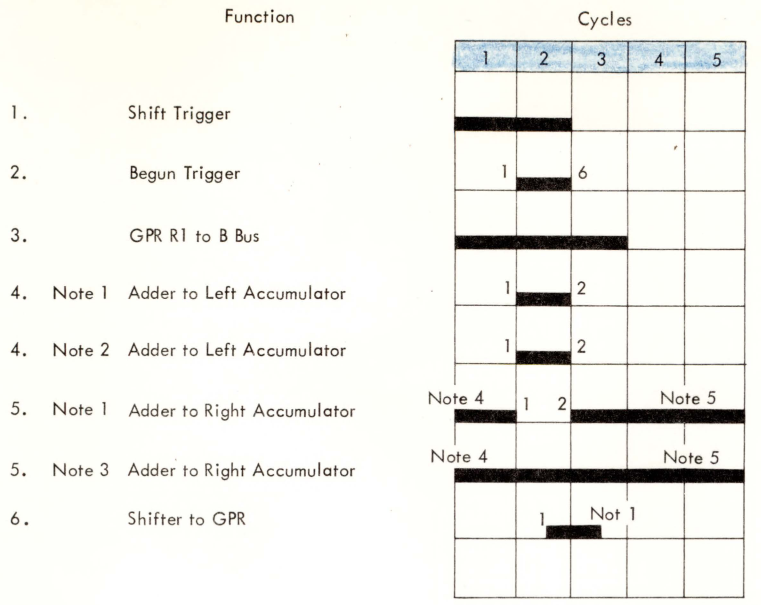
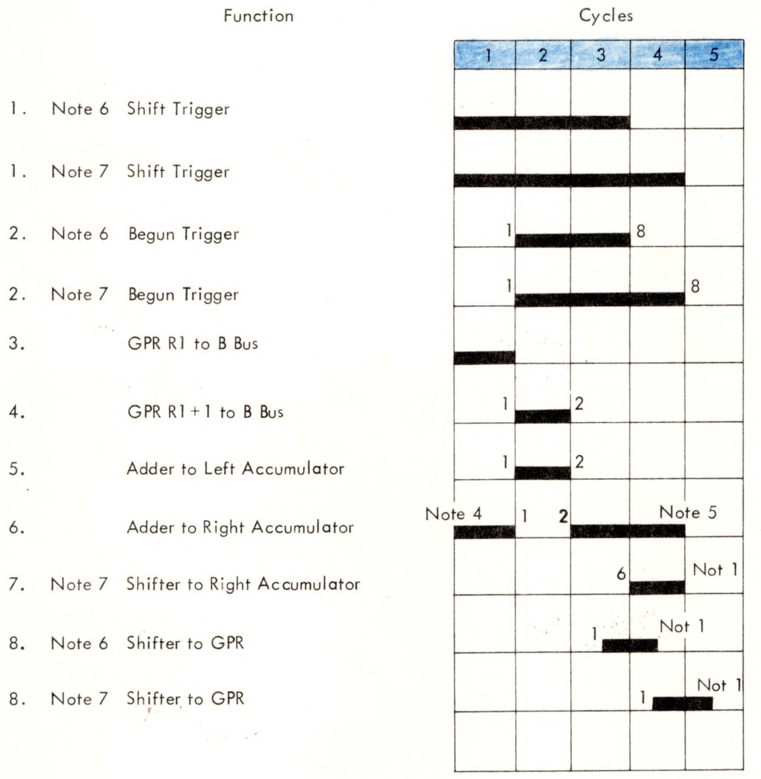

Decode Shift Ops 88 to 8F
Block Op Condition (E55)
Yes
No
Cycle 1 Trigger (D53)
Yes
No
Hardware Error. Cycle 1 Trigger should be On for First Cycle of Each Operation
1. Set up controls and gates for a shift op
2. GPR name from R1 field is set into the GPR sink register.
6. Shift amount is set into the I Ctr to generate shifter gates
1. Turn On Shift Trigger (J50) NK167
2. Set R1 to GPR Sk. Reg. (E56-63) NK169
3. Turn On Data Available (H46) NK137
4. Set R1 to B Bus NK165
5. Turn On FXP Process (E52) NK361
6. Set I Ctr from FXOS Bits 13-18 (F61-66) NK237
4. Value from the R1 GPR is gated through the adder to the accumulator
Op Codes 8C to 8F
Yes
No
Turn On Double Trigger (E66) NK235
Double Shift Operations
Single shift operations. Op codes 88, 89 8A or 8B
Op Codes 88, 89 8C or 8D
Yes
No
Turn On Logical Trigger (L50) NK165
Logical Shift Operations
Arithmetic shift ops. Op codes 8A, 8B, 8E or 8F
Op Codes 89, 8B 8D or 8F
Yes
No
Turn On Left Trigger (K50) NK165
Left shift operations
Right shift operations. Op codes 88, 8A 8C or 8E
Cycle 1
Cycle 2
1. Turn On Begun Trigger (H47) NJ603
2. Step Op NG923
1. Begun tgr not used until following cycle
2. Begin to decode next pseudo Op
Single shift with odd GPR name
R1 (GPR Sk. Reg.) Odd (E64)
Yes
No
Single shift with even GPR name or all double shifts
Arithmetic shift operation
Logical Tgr On (L50)
No
Yes
Left Trigger On (K50)
No
Yes
Right shift
Ingate Adder to Left and Right Accumulator (H59-60) NJ001-035-037
Ingate Adder to Right Accumulator (H60) NJ003-097-099-101
Ingate Adder to Left Accumulator (H59) NJ633
1. GPR R1 is no longer needed for this op (after this cycle)
2. After this cycle fixed area will be ready for a new op.
Double Trigger On (E66)
No
Yes
1. Turn On RUA Response (E67) NJ619
2. Turn Off FXP in Process (E52) NK369
3. Turn Off FXP Ovrlp Trigger (F52) NK369
4. Block Turn On GPR Accept Tgr (G53) NG931
R1 (GPR Sk Reg.) Odd (E64)
No
Yes
Left Trigger (K50)
No
Yes
Set Right Accumulator to 0's NJ369
Left Trigger (K50)
Yes
No
Logical Trigger (L50)
No
Yes
Operand Negative
No
Yes
Set Left Accumulator to 0's NJ639
Set Left Accumulator to 1's NU639
See decoding chart for active gates
Decode Shift Amount and Enable Shifter Gates
Set Shifter to GPRs (H65) NJ615
Single Shift Cycle 2
Set R1 + 1 to the B Bus NK269
Value from R1 +1 GPR is gated through the adder to the accumulator
Double Shift Cycle 3
Ingate Adder to Right Accumulator (H60) NJ003-097-099-101
Shift Amount More Than 32
Yes
No
Decode Shift of 32 and Enable Shifter Gates
Decode Shift Amount and Enable Shifter Gates
See decoding chart for active gates
Double Shift Cycle 4
Return data to accumulator for remainder of shift amount
Ingate Shifter to Left and Right Accumulator (H61) NJ629
Step I Ctr Bit 32 to 0 (F61) NJ557-559
Reduce shift amount by 32
Decode Remaining Shift Amount and Enable Shifter Gates
See decoding chart for active gates
Double Shift Cycle 4 (Shift of more than 32)
(Shift of less than 33) Double shift cycle 3
Set Shifter to GPRs (H65) NJ615
GPR R1 is no longer needed for this Op. (after this cycle)
I-Box Direct Store into GPR
1. Turn On RUA Response (E67) NJ619
2. Turn Off FXP in Process (E52) NK369
3. Turn Off FXP Ovrlp Tgr (F52) NK369
4. Block Turn On GPR Accept Tgr (G53) NG931
After this cycle fixed area will be ready for a new op.
Double Shift Cycle 3 or 4
1. Ingate Results into GPRs (H51) NJ615
2. Reset all Shift Controls
Ending sequence occurs in the first part of the cycle in parallel with the next op.
End of Operation
Ending Sequence Occurs in:
1. Cycle 3 for a single shift
2. Cycle 4 for a double shift less than 33.
3. Cycle 5 for a double shift more than 32.

DIAGRAM 5-112. SHIFT INSTRUCTIONS (SHEET 2 OF 3)

Single Shift Timing


Function
Cycles
1
2
3
4
5
1. Shift Trigger
2. Begun Trigger
1
6
3. GPR R1 to B Bus
4. Note 1 Adder to Left Accumulator
1
2
4. Note 2 Adder to Left Accumulator
1
2
5. Note 1 Adder to Right Accumulator
Note 4
1
2
Note 5
5. Note 3 Adder to Right Accumulator
Note 4
Note 5
6. Shifter to GPR
1
Not 1

Double Shift Timing



Notes:

1. Address of sink GPR is even.
2. Address of sink GPR is odd during an arithmetic left shift.
3. Address of sink GPR is odd.
4. Depends on previous op.
5. Depends on following op.
6. Shift amount is equal to or less than 32.
7. Shift amount is more than 32.

DIAGRAM 5-112. SHIFT INSTRUCTIONS (SHEET 3 OF 3)

Objectives:

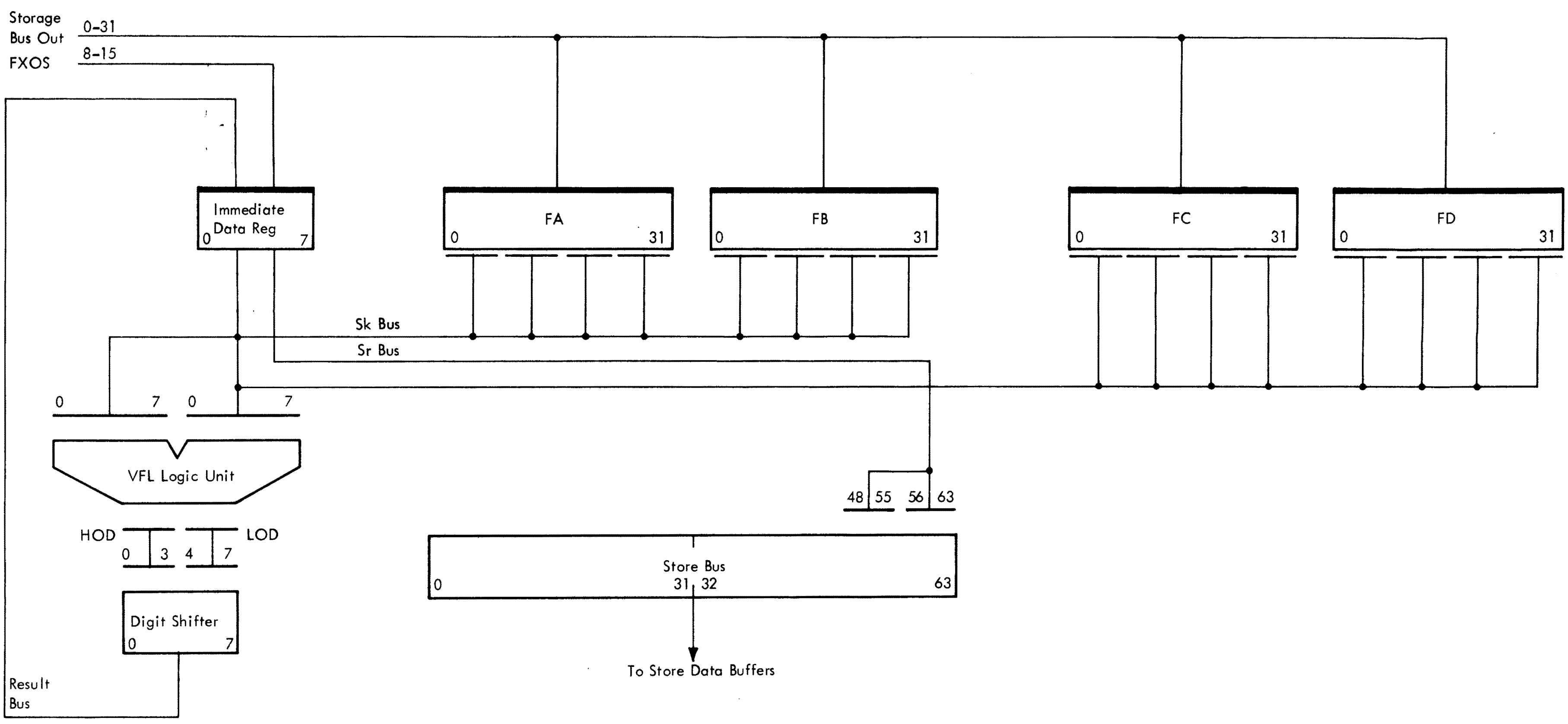
AND, OR, Exclusive OR
NI, OI, XI

	0	1–7	8–15	16–17	18–20
FXOS Format	SI	94,96,97	I Field	FXB Name	SAR Name

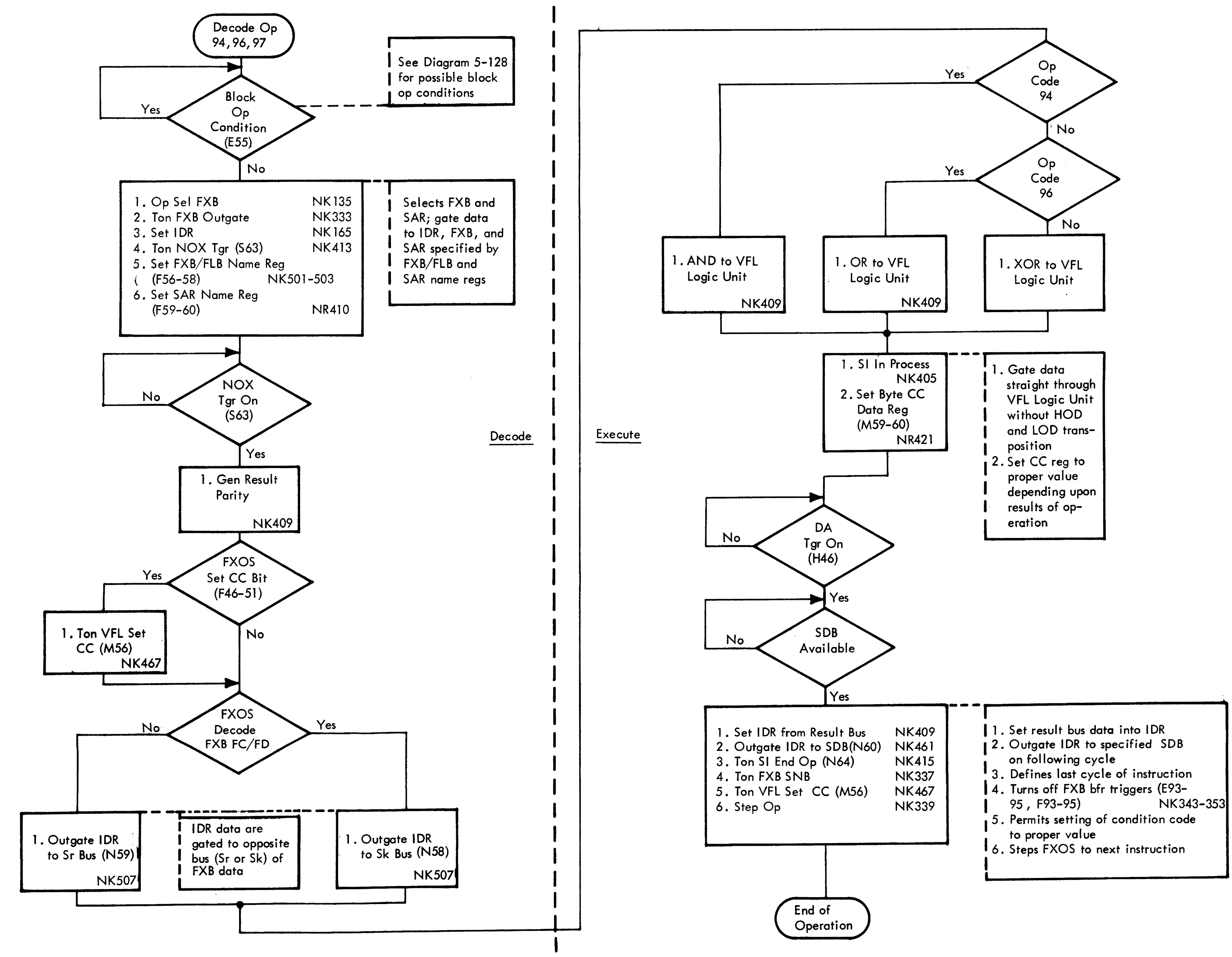
NI,OI, or XI Condition Codes:
0-Result is 0
1-Result is not 0
2-(Not applicable)
3-(Not applicable)

1. Specified FXB byte is ANDed (94), ORed (96) or Exclusive ORed (97) with immediate data field in VFL Logic Unit.
2. FXB data and immediate data are gated to VFL Logic Unit over opposite buses (Sk or Sr).
3. Results are outgated to specified SDB.

Data Flow



Flow Chart



Decode Op 94,96,97
Block Op Condition (E55)
Yes
No
See Diagram 5-128 for possible block op conditions
1. Op Sel FXB NK135
2. Ton FXB Outgate NK333
3. Set IDR NK165
4. Ton NOX Tgr (S63) NK413
5. Set FXB/FLB Name Reg (F56-58) NK501-503
6. Set SAR Name Reg (F59-60) NR410
Selects FXB and SAR; gate data to IDR, FXB, and SAR specified by FXB/FLB and SAR name regs
NOX Tgr On (S63)
No
Yes
1. Gen Result Parity NK409
FXOS Set CC Bit (F46-51)
Yes
No
1. Ton VFL Set CC (M56) NK467
FXOS Decode FXB FC/FD
No
Yes
1. Outgate IDR to Sr Bus (N59) NK507
IDR data are gated to opposite bus (Sr or Sk) of FXB data
1. Outgate IDR to Sk Bus (N58) NK507
Decode
Execute
Op Code 94
Yes
No
Op Code 96
Yes
No
1. AND to VFL Logic Unit NK409
1. OR to VFL Logic Unit NK409
1. XOR to VFL Logic Unit
1. SI In Process NK405
2. Set Byte CC Data Reg (M59-60) NR421
1. Gate data straight through VFL Logic Unit without HOD and LOD transposition
2. Set CC reg to proper value depending upon results of operation
DA Tgr On (H46)
No
Yes
SDB Available
No
Yes
1. Set IDR from Result Bus NK409
2. Outgate IDR to SDB(N60) NK461
3. Ton SI End Op (N64) NK415
4. Ton FXB SNB NK337
5. Ton VFL Set CC (M56) NK467
6. Step Op NK339
1. Set result bus data into IDR
2. Outgate IDR to specified SDB on following cycle
3. Defines last cycle of instruction
4. Turns off FXB bfr triggers (E93-95, F93-95) NK343-353
5. Permits setting of condition code to proper value
6. Steps FXOS to next instruction
End of Operation

DIAGRAM 5-113. NI, OI, AND XI INSTRUCTIONS

Objectives:

Test Under Mask
TM

	0	1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19	19 20
FXOS Format	SI	91	I Field	FXB Name	▨

1. State of bits in FXB byte selected by mask in immediate data field is used to set condition code.
2. Mask bit of 1 indicates corresponding FXB bit is to be tested; mask bit of 0 indicates corresponding bit is to be ignored.
3. FXB data and immediate data are gated to VFL Logic Unit over opposite buses (Sk or Sr).

TM Condition Code:
0 - Selected bits and mask are all 0
1 - Selected bits are mixed 0 and 1
2 - (Not applicable)
3 - Selected bits are all 1

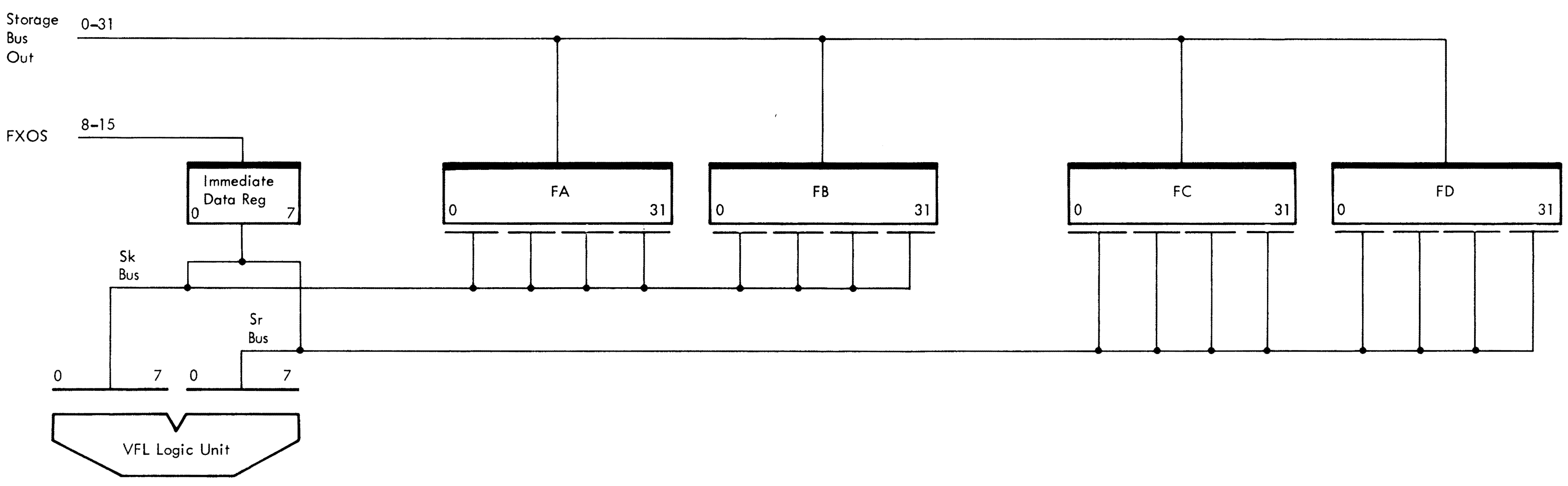
Compare Logical
CLI

	0	1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18 19	19 20
FXOS Format	SI	95	I Field	FXB Name	▨

1. Specified FXB byte is compared with immediate data field in VFL Logic Unit.
2. FXB data and immediate data are gated to VFL Logic Unit over opposite buses (Sk or Sr).
3. Results of operation are indicated in condition code.

CLI Condition Code:
0 - Operands are equal
1 - FXB data is low
2 - FXB data is high
3 - (Not applicable)

TM Data Flow



CLI Data Flow

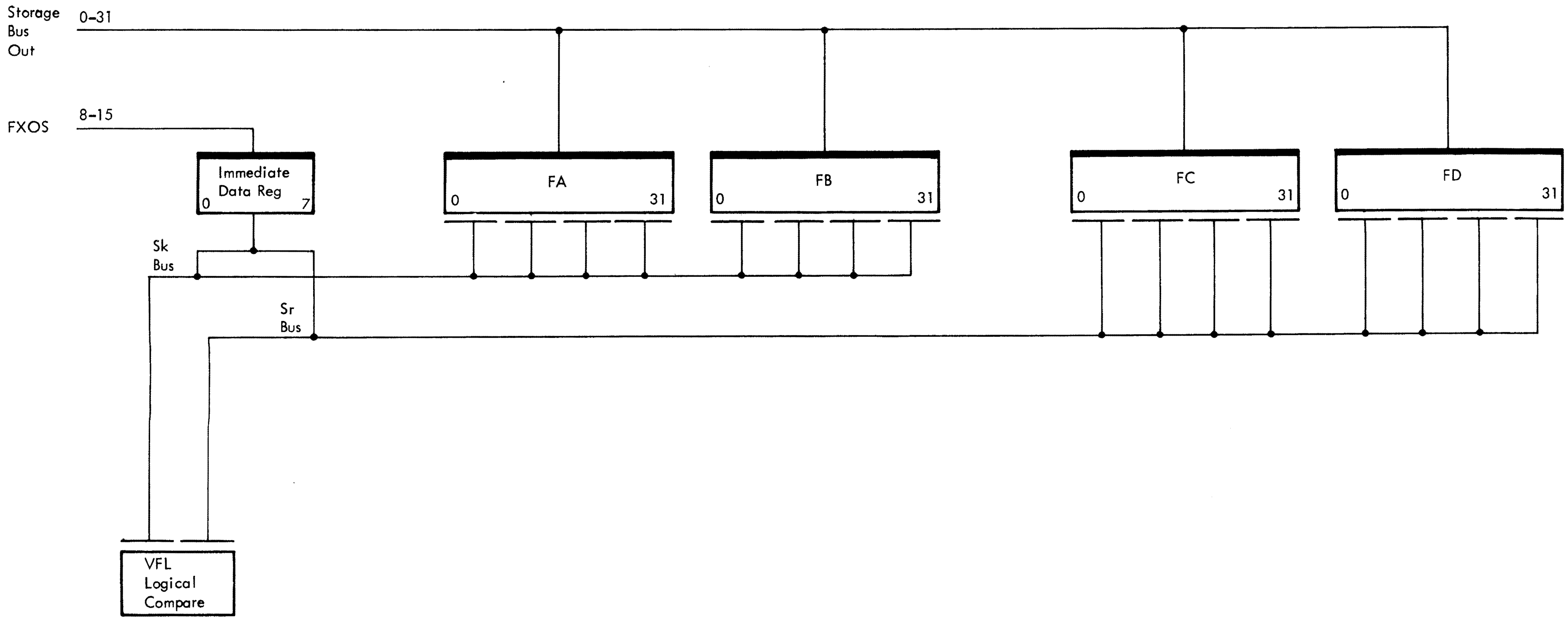


DIAGRAM 5-114. TM AND CLI INSTRUCTIONS (SHEET 1 OF 2)

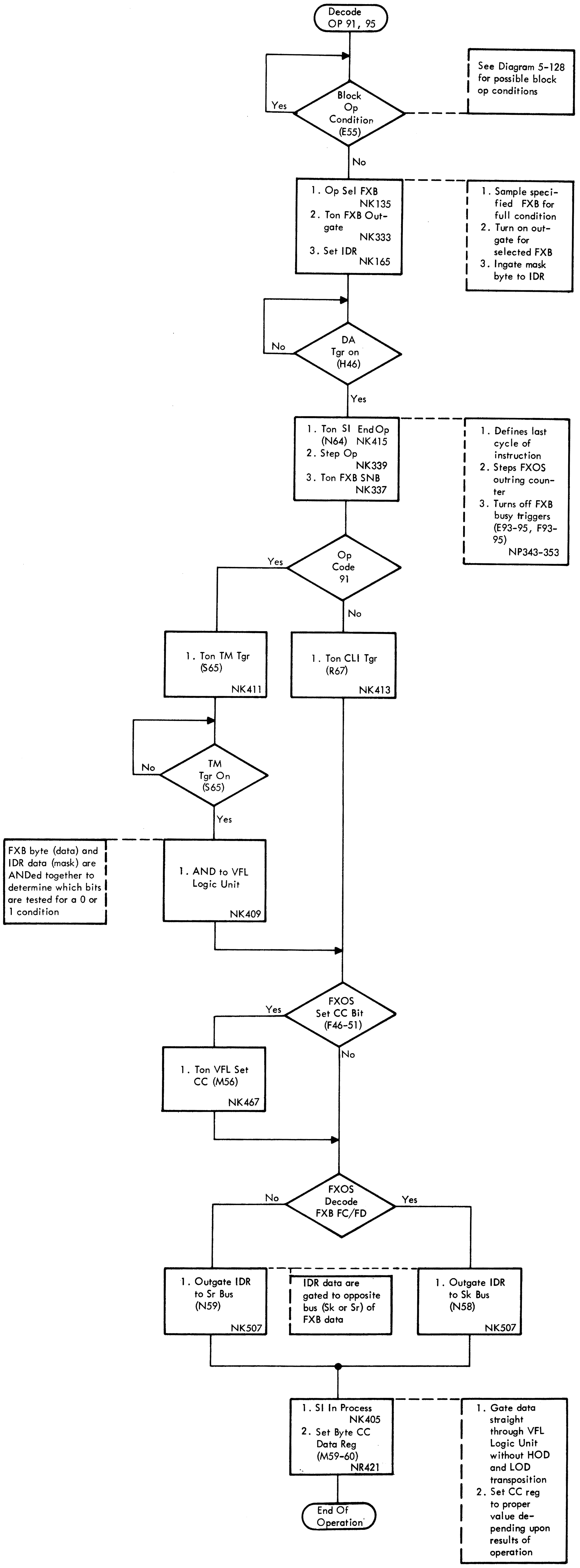
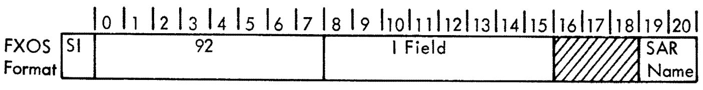

Decode
OP 91, 95
See Diagram 5-128 for possible block op conditions
Yes
Block Op Condition (E55)
No
1. Op Sel FXB NK135
2. Ton FXB Out-gate NK333
3. Set IDR NK165
1. Sample specified FXB for full condition
2. Turn on out-gate for selected FXB
3. Ingate mask byte to IDR
No
DA Tgr on (H46)
Yes
1. Ton SI EndOp (N64) NK415
2. Step Op NK339
3. Ton FXB SNB NK337
1. Defines last cycle of instruction
2. Steps FXOS outring counter
3. Turns off FXB busy triggers (E93-95, F93-95) NP343-353
Yes
Op Code 91
No
1. Ton TM Tgr (S65) NK411
1. Ton CLI Tgr (R67) NK413
No
TM Tgr On (S65)
Yes
FXB byte (data) and IDR data (mask) are ANDed together to determine which bits are tested for a 0 or 1 condition
1. AND to VFL Logic Unit NK409
Yes
FXOS Set CC Bit (F46-51)
No
1. Ton VFL Set CC (M56) NK467
No
FXOS Decode FXB FC/FD
Yes
1. Outgate IDR to Sr Bus (N59) NK507
IDR data are gated to opposite bus (Sk or Sr) of FXB data
1. Outgate IDR to Sk Bus (N58) NK507
1. SI In Process NK405
2. Set Byte CC Data Reg (M59-60) NR421
1. Gate data straight through VFL Logic Unit without HOD and LOD transposition
2. Set CC reg to proper value depending upon results of operation
End Of Operation

DIAGRAM 5-114. TM AND CLI INSTRUCTIONS (SHEET 2 OF 2)

Objectives:

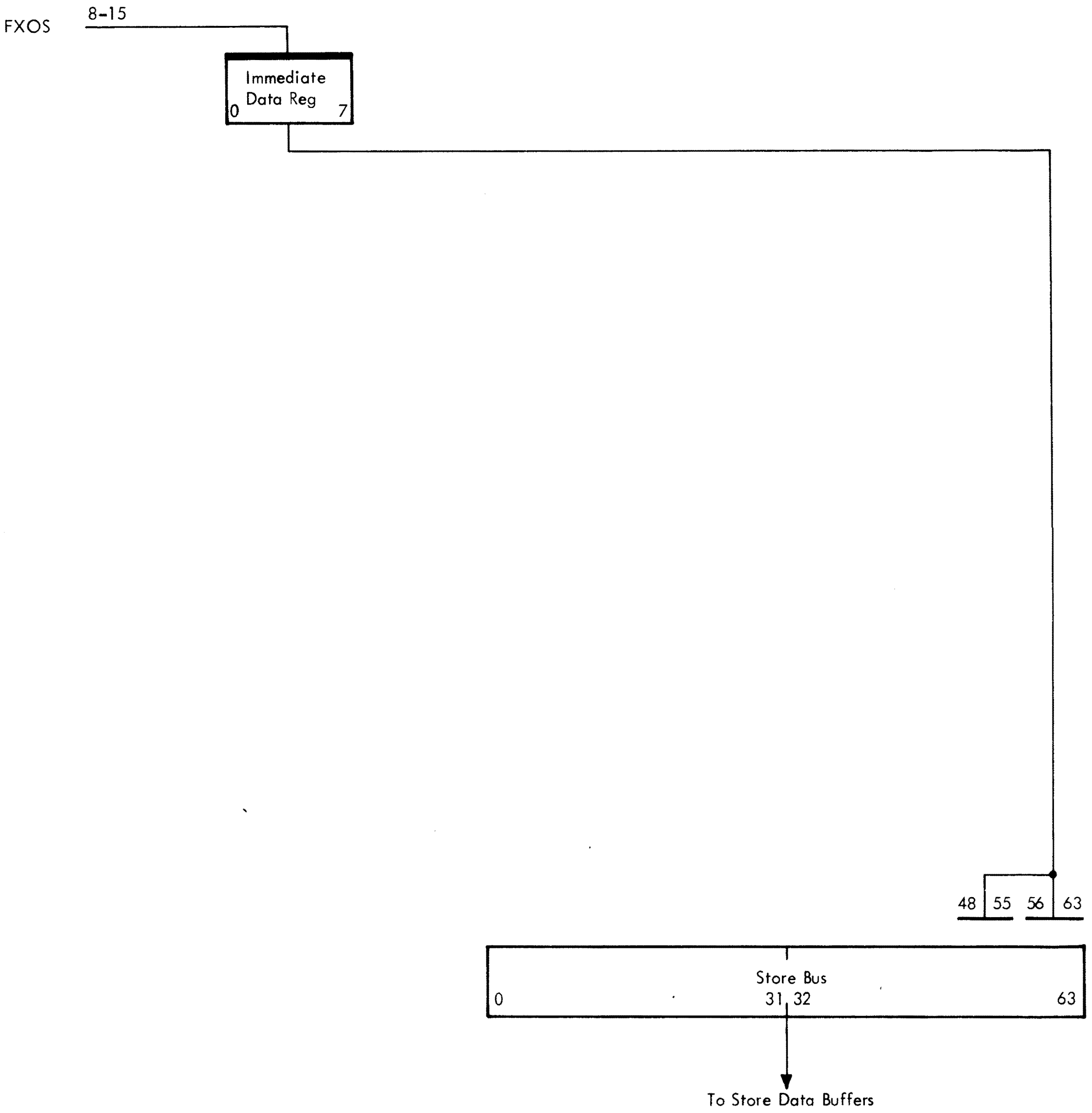
Move
MVI



1. IDR data are moved to a specified SDB
2. IDR data are gated onto store bus positions 48 through 55 and 56 through 63.

MVI Condition Code:
Code remains unchanged

Data Flow



Flow Chart

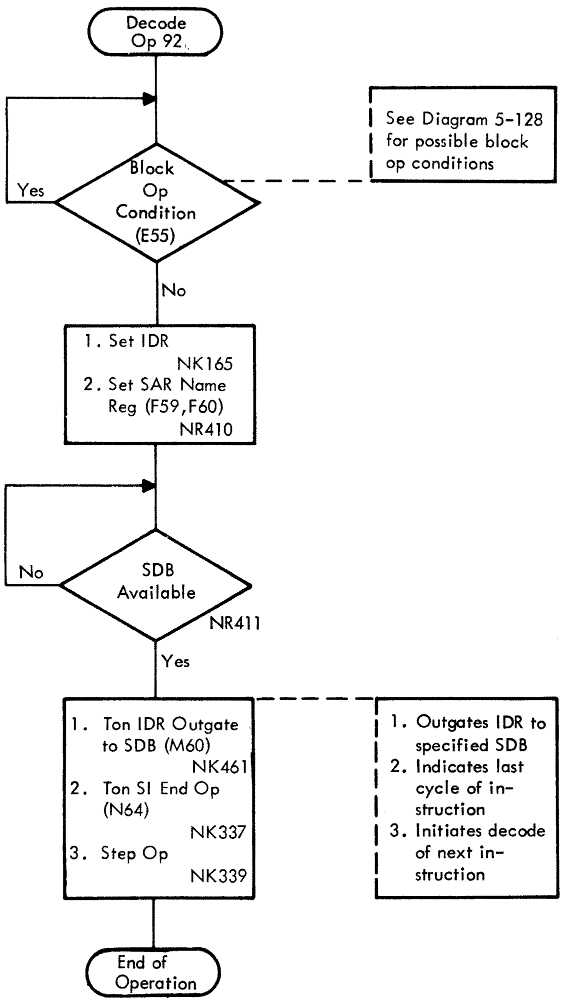

Decode Op 92
Yes
Block Op Condition (E55)
See Diagram 5-128 for possible block op conditions
No
1. Set IDR
NK165
2. Set SAR Name Reg (F59,F60)
NR410
No
SDB Available
NR411
Yes
1. Ton IDR Outgate to SDB (M60)
NK461
2. Ton SI End Op (N64)
NK337
3. Step Op
NK339
1. Outgates IDR to specified SDB
2. Indicates last cycle of instruction
3. Initiates decode of next instruction
End of Operation

DIAGRAM 5-115. MVI INSTRUCTION

Objectives:

Test and Set
TS

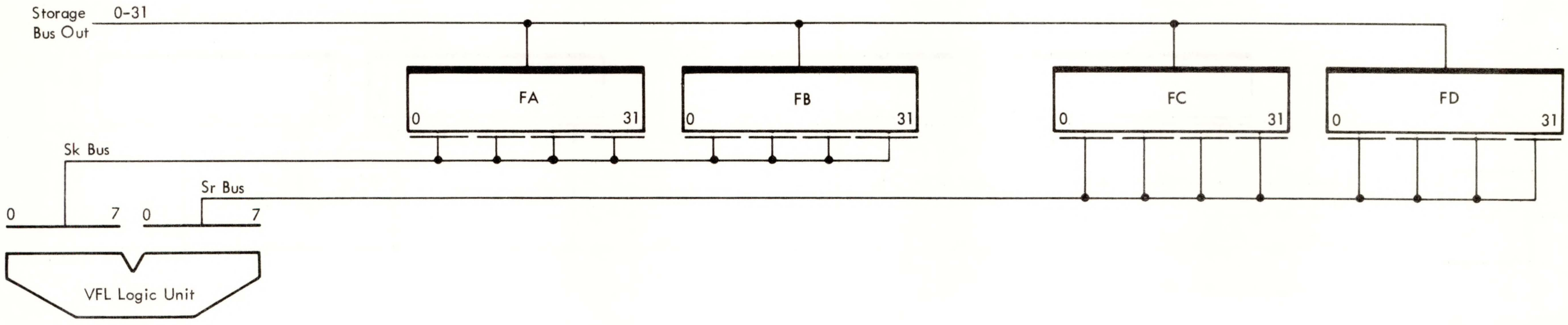
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

SI	93	▨	FXB Name	▨

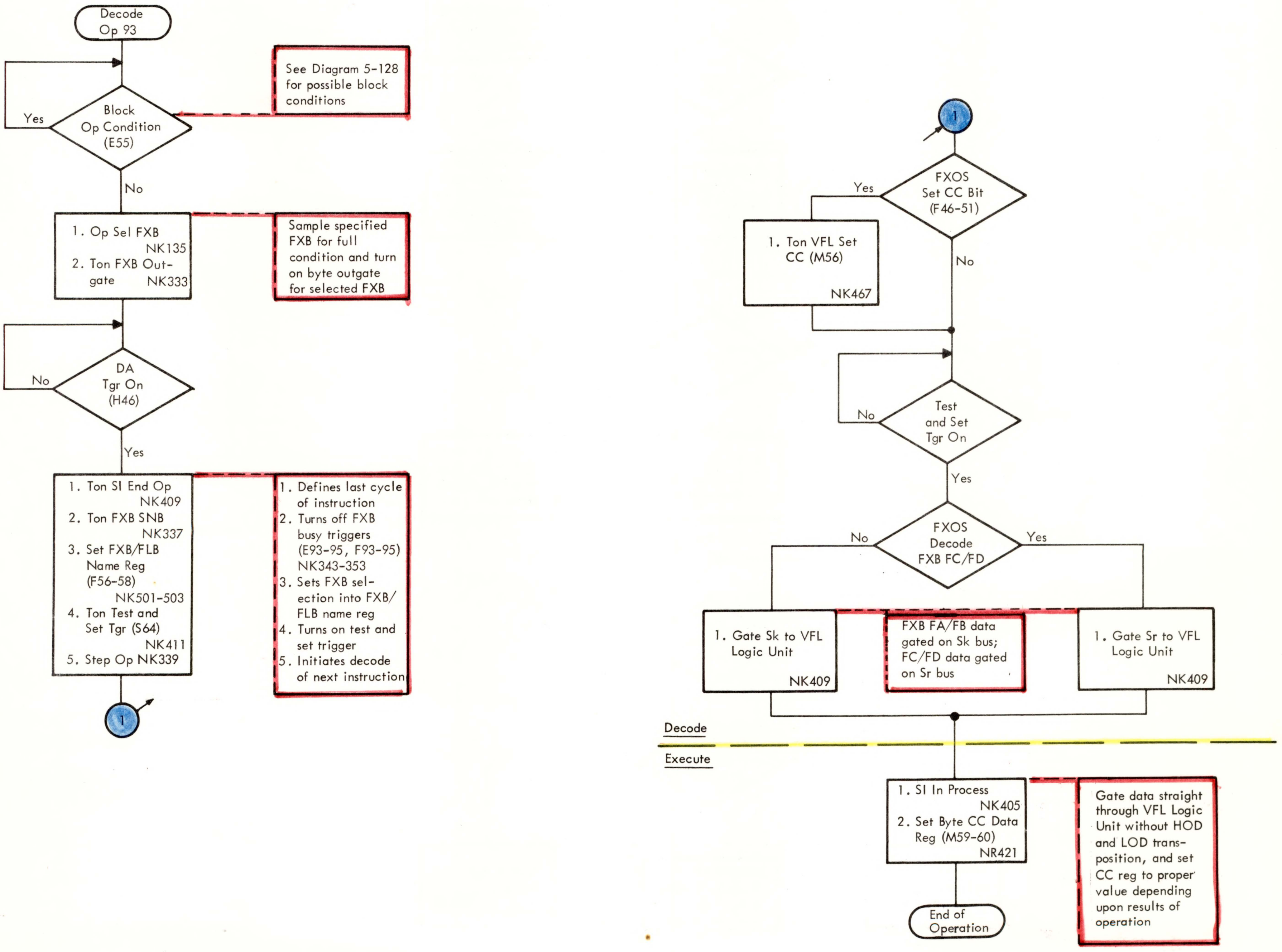
1. VFLEU only performs part of instruction. MSCE performs other part. MSCE outgates a word from storage and (1) sends it to VFLEU, and (2) sets specified byte of word to all 1s and places it back in storage.
2. Specified byte is gated from FXB to VFL Logic Unit where bit 0 of byte is tested for a 0 or 1 condition. Condition code is set accordingly.

TS Condition Code:
0 - Bit 0 of specified byte is zero
1 - Bit 1 of specified byte is one
2 - (Not applicable)
3 - (Not applicable)

Data Flow



Flow Chart

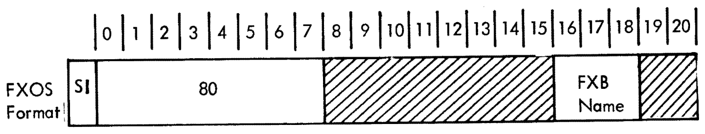


Decode Op 93
Block Op Condition (E55)
Yes
No
See Diagram 5-128 for possible block conditions
1. Op Sel FXB NK135
2. Ton FXB Out-gate NK333
Sample specified FXB for full condition and turn on byte outgate for selected FXB
DA Tgr On (H46)
No
Yes
1. Ton SI End Op NK409
2. Ton FXB SNB NK337
3. Set FXB/FLB Name Reg (F56-58) NK501-503
4. Ton Test and Set Tgr (S64) NK411
5. Step Op NK339
1. Defines last cycle of instruction
2. Turns off FXB busy triggers (E93-95, F93-95) NK343-353
3. Sets FXB selection into FXB/FLB name reg
4. Turns on test and set trigger
5. Initiates decode of next instruction
1
1
FXOS Set CC Bit (F46-51)
Yes
No
1. Ton VFL Set CC (M56) NK467
Test and Set Tgr On
No
Yes
FXOS Decode FXB FC/FD
No
Yes
1. Gate Sk to VFL Logic Unit NK409
FXB FA/FB data gated on Sk bus; FC/FD data gated on Sr bus
1. Gate Sr to VFL Logic Unit NK409
Decode
Execute
1. SI In Process NK405
2. Set Byte CC Data Reg (M59-60) NR421
Gate data straight through VFL Logic Unit without HOD and LOD transposition, and set CC reg to proper value depending upon results of operation
End of Operation

DIAGRAM 5-116. TS INSTRUCTION

Objective:

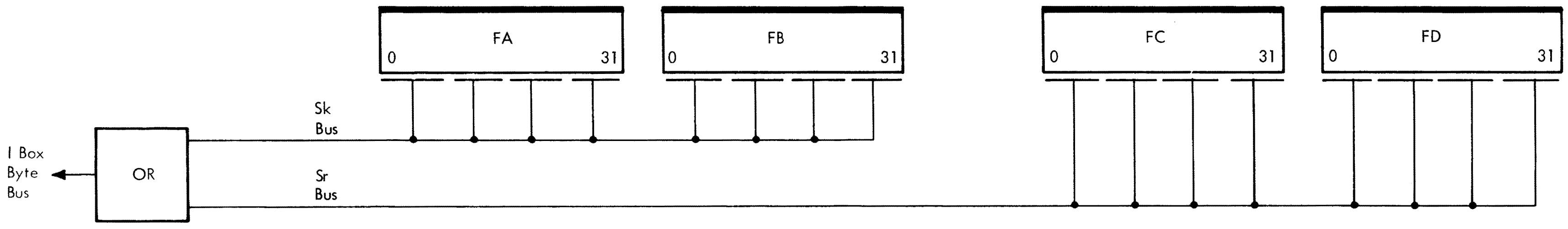
Set System Mask
SSM



1. Byte in specified FXB replaces system mask bits in current PSW.
2. FXB byte is sent to I-Box via byte bus.
3. FX complete signal is sent to I-Box causing current PSW to ingate new mask byte.

SSM Condition Code:
Code remains unchanged

Data Flow



Flow Chart

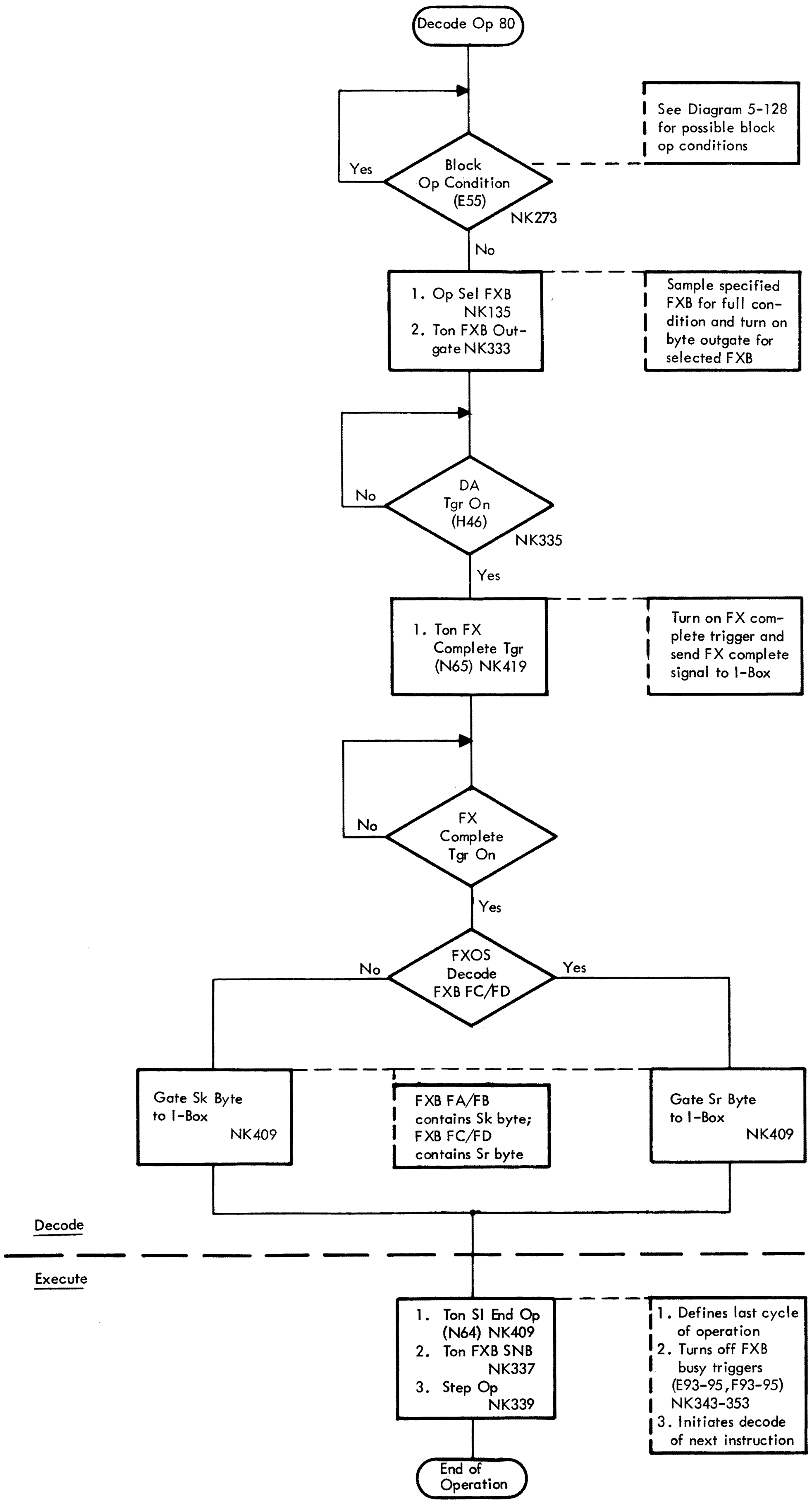
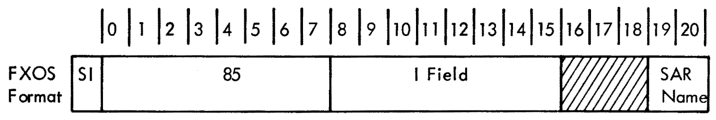

Decode Op 80
Block Op Condition (E55)
NK273
Yes
No
See Diagram 5-128 for possible block op conditions
1. Op Sel FXB NK135
2. Ton FXB Outgate NK333
Sample specified FXB for full condition and turn on byte outgate for selected FXB
DA Tgr On (H46)
NK335
No
Yes
1. Ton FX Complete Tgr (N65) NK419
Turn on FX complete trigger and send FX complete signal to I-Box
FX Complete Tgr On
No
Yes
FXOS Decode FXB FC/FD
No
Yes
Gate Sk Byte to I-Box NK409
FXB FA/FB contains Sk byte; FXB FC/FD contains Sr byte
Gate Sr Byte to I-Box NK409
Decode
Execute
1. Ton SI End Op (N64) NK409
2. Ton FXB SNB NK337
3. Step Op NK339
1. Defines last cycle of operation
2. Turns off FXB busy triggers (E93-95,F93-95) NK343-353
3. Initiates decode of next instruction
End of Operation

DIAGRAM 5-117. SSM INSTRUCTION

Objectives:

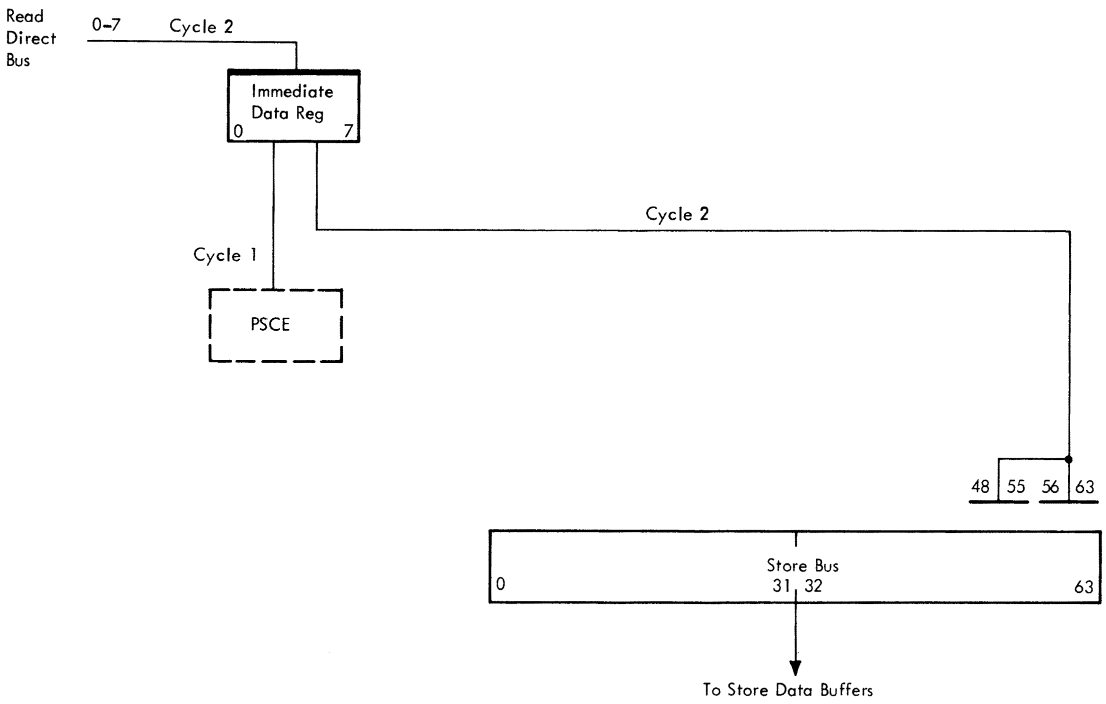
Read Direct
RDD



1. IDR data are made available as signal-out timing signals; direct-in data are placed in IDR for subsequent transfer to specified SDB.
2. IDR data are gated to singleshots and direct-in data are set into IDR via result bus after singleshots are activated.
3. Direct-in data are transferred from IDR to specified SDB.

RDD Condition Code:
Code remains unchanged

Data Flow



Flow Chart

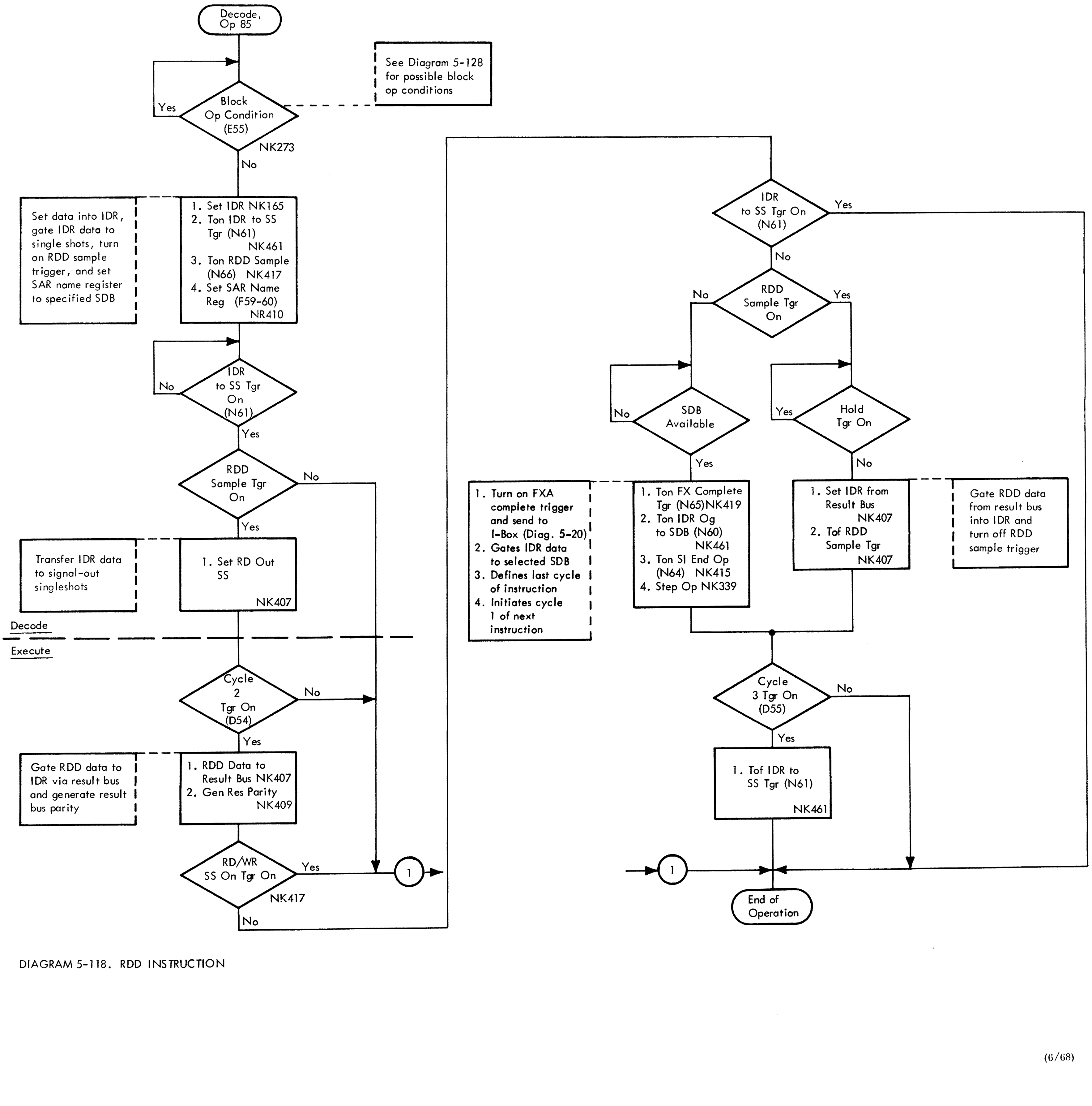

Decode, Op 85
Block Op Condition (E55)
NK273
Yes
No
See Diagram 5-128 for possible block op conditions
Set data into IDR, gate IDR data to single shots, turn on RDD sample trigger, and set SAR name register to specified SDB
1. Set IDR NK165
2. Ton IDR to SS Tgr (N61) NK461
3. Ton RDD Sample (N66) NK417
4. Set SAR Name Reg (F59-60) NR410
IDR to SS Tgr On (N61)
No
Yes
RDD Sample Tgr On
No
Yes
Transfer IDR data to signal-out singleshots
1. Set RD Out SS
NK407
Decode
Execute
Cycle 2 Tgr On (D54)
No
Yes
Gate RDD data to IDR via result bus and generate result bus parity
1. RDD Data to Result Bus NK407
2. Gen Res Parity NK409
RD/WR SS On Tgr On
NK417
Yes
No
1
IDR to SS Tgr On (N61)
Yes
No
RDD Sample Tgr On
No
Yes
SDB Available
No
Yes
Hold Tgr On
Yes
No
1. Turn on FXA complete trigger and send to I-Box (Diag. 5-20)
2. Gates IDR data to selected SDB
3. Defines last cycle of instruction
4. Initiates cycle 1 of next instruction
1. Ton FX Complete Tgr (N65)NK419
2. Ton IDR Og to SDB (N60) NK461
3. Ton SI End Op (N64) NK415
4. Step Op NK339
1. Set IDR from Result Bus NK407
2. Tof RDD Sample Tgr NK407
Gate RDD data from result bus into IDR and turn off RDD sample trigger
Cycle 3 Tgr On (D55)
No
Yes
1. Tof IDR to SS Tgr (N61)
NK461
1
End of Operation

DIAGRAM 5-118. RDD INSTRUCTION

Objectives:

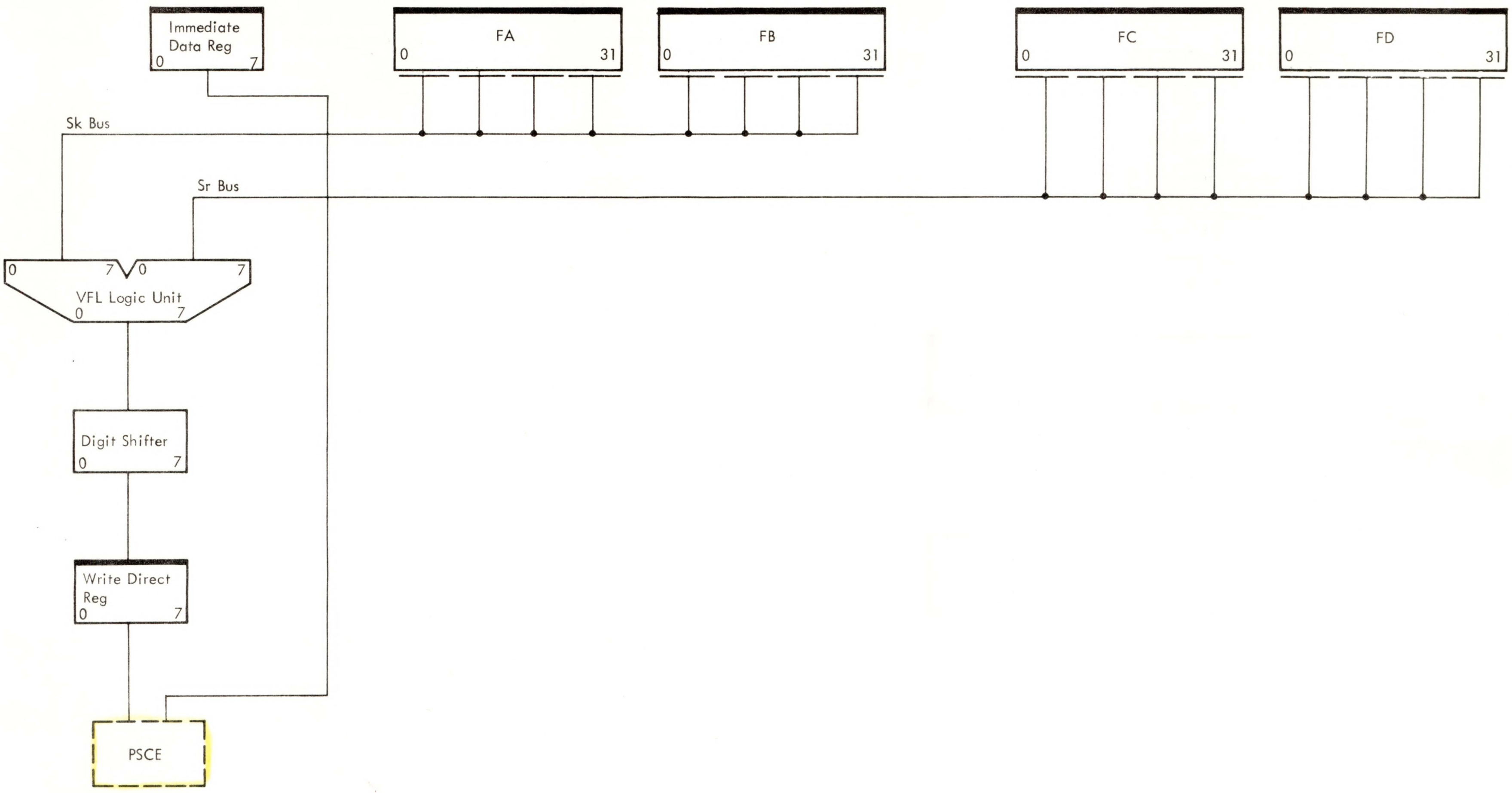
Write Direct
WRD

	0	1 2 3 4 5 6 7	8 9 10 11 12 13 14 15	16 17 18	19 20
FXOS Format	SI	84	I Field	FXB Name	▨

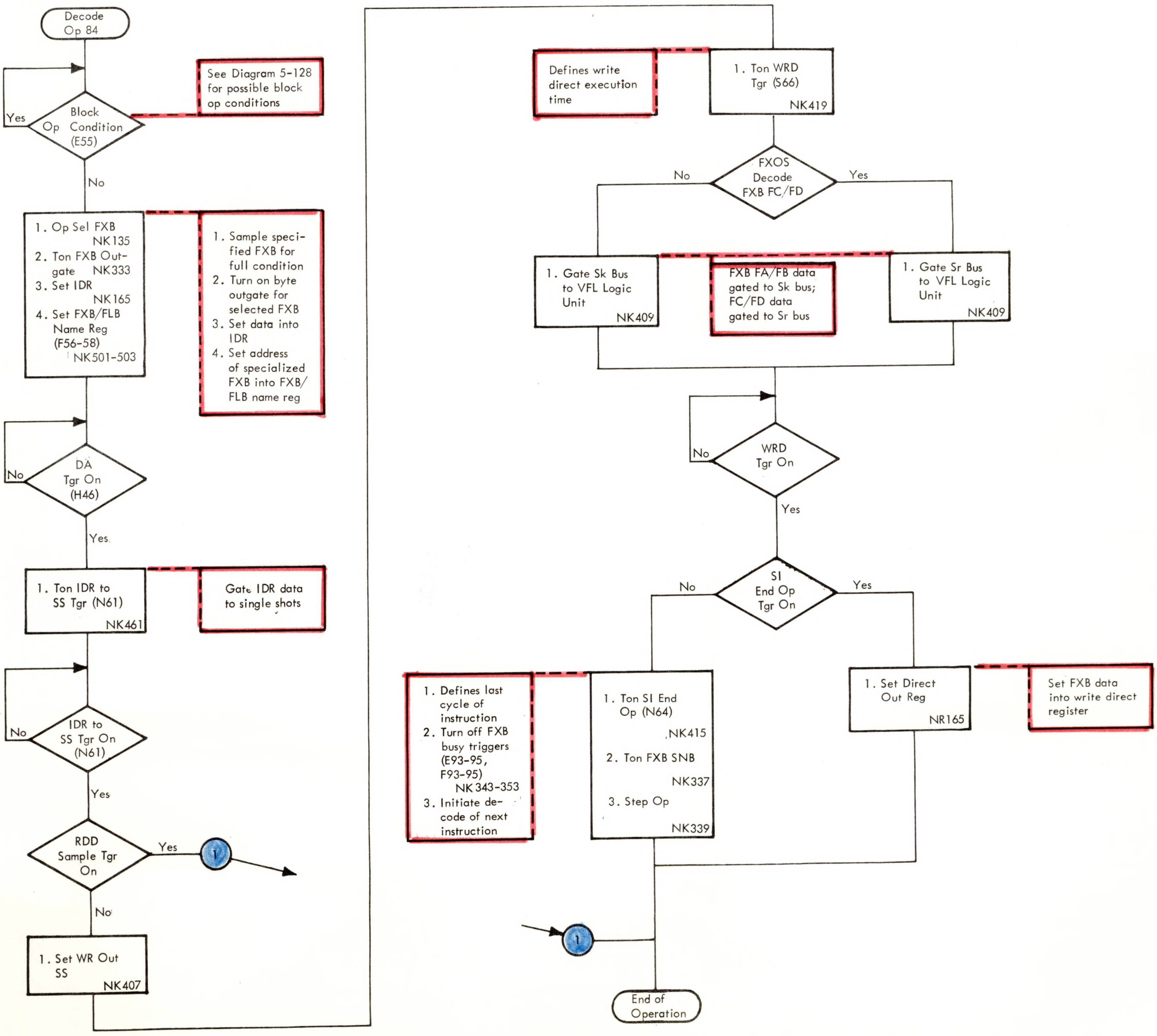
1. A byte in the specified FXB is made available as a set of direct-out static signals and IDR data are made available as signal-out timing signals.
2. When instruction is decoded, IDR data are gated to singleshots.
3. After activation of singleshots, FXB data are gated to write direct register.

WRD Condition Code:
Code remains unchanged

Data Flow



Flow Chart



Decode Op 84
Block Op Condition (E55)
Yes
No
See Diagram 5-128 for possible block op conditions
1. Op Sel FXB NK135
2. Ton FXB Outgate NK333
3. Set IDR NK165
4. Set FXB/FLB Name Reg (F56-58) NK501-503
1. Sample specified FXB for full condition
2. Turn on byte outgate for selected FXB
3. Set data into IDR
4. Set address of specialized FXB into FXB/FLB name reg
DA Tgr On (H46)
No
Yes
1. Ton IDR to SS Tgr (N61) NK461
Gat‹ IDR data to single shots
IDR to SS Tgr On (N61)
No
Yes
RDD Sample Tgr On
Yes
No
1
1. Set WR Out SS NK407
Defines write direct execution time
1. Ton WRD Tgr (S66) NK419
FXOS Decode FXB FC/FD
No
Yes
1. Gate Sk Bus to VFL Logic Unit NK409
FXB FA/FB data gated to Sk bus; FC/FD data gated to Sr bus
1. Gate Sr Bus to VFL Logic Unit NK409
WRD Tgr On
No
Yes
SI End Op Tgr On
No
Yes
1. Defines last cycle of instruction
2. Turn off FXB busy triggers (E93-95, F93-95) NK343-353
3. Initiate decode of next instruction
1. Ton SI End Op (N64) NK415
2. Ton FXB SNB NK337
3. Step Op NK339
1. Set Direct Out Reg NR165
Set FXB data into write direct register
1
End of Operation

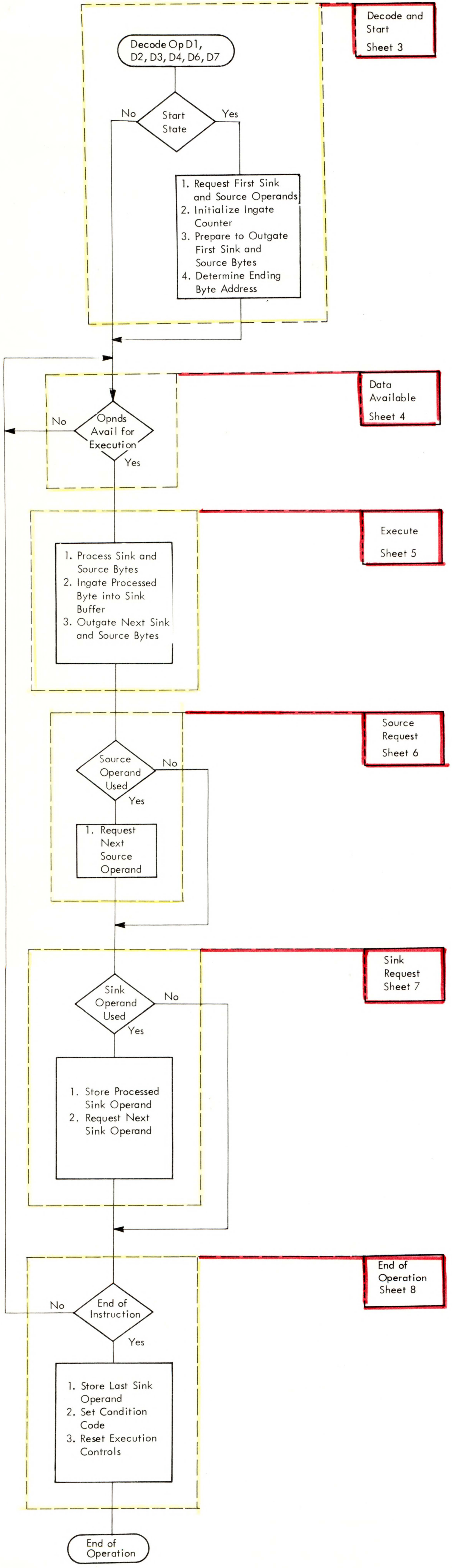
DIAGRAM 5-119. WRD INSTRUCTION

Objectives:

AND, OR, Exclusive OR, Move Numerics, Move Zones
NC, OC, XC, MVN, MVZ

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
SS	D4, D6, D7, D1, D3								▨	FLB Source			P	O	S	E	FLB Sink			SAR Name	

Simplified Flow

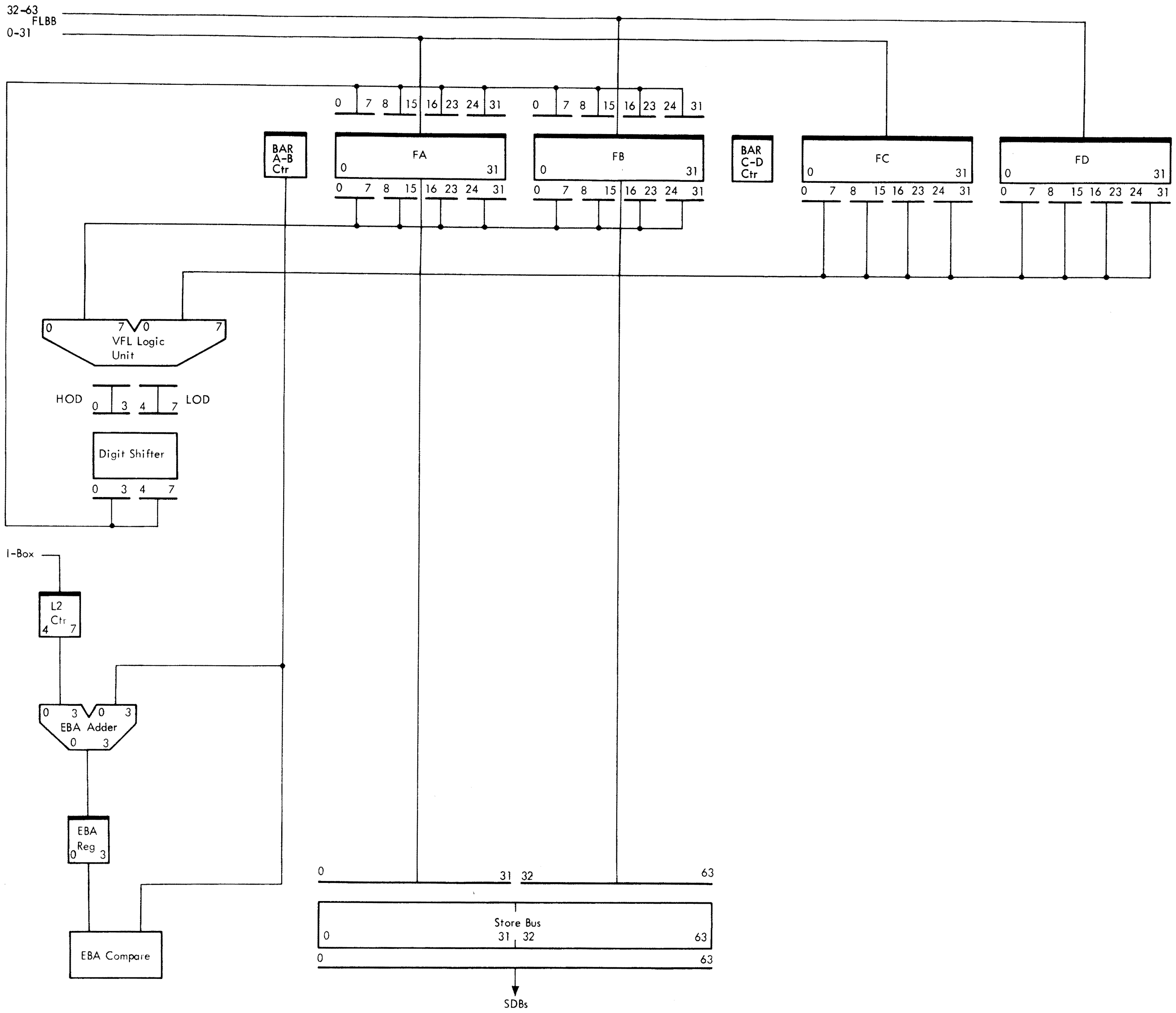


Decode Op D1, D2, D3, D4, D6, D7
Start State
No
Yes
1. Request First Sink and Source Operands
2. Initialize Ingate Counter
3. Prepare to Outgate First Sink and Source Bytes
4. Determine Ending Byte Address
Decode and Start
Sheet 3
Opnds Avail for Execution
No
Yes
Data Available
Sheet 4
1. Process Sink and Source Bytes
2. Ingate Processed Byte into Sink Buffer
3. Outgate Next Sink and Source Bytes
Execute
Sheet 5
Source Operand Used
No
Yes
1. Request Next Source Operand
Source Request
Sheet 6
Sink Operand Used
No
Yes
1. Store Processed Sink Operand
2. Request Next Sink Operand
Sink Request
Sheet 7
End of Instruction
No
Yes
1. Store Last Sink Operand
2. Set Condition Code
3. Reset Execution Controls
End of Operation
Sheet 8
End of Operation

1. Sink and source doublewords are gated from FLBs to FXBs FA/FB and FC/FD, respectively.
2. Sink and source bytes are gated to the VFL logic unit and are processed as follows:
AND (NC) - Logical product is obtained between the sink and source bytes.
OR (OC) - Logical sum is obtained between the sink and source bytes.
Exclusive OR (XC) - Module-two sum is obtained between the sink and source bytes.
Move Numerics (MVN) - LOD of source byte is placed in LOD position of sink byte.
Move Zones (MVZ) - HOD of source byte is placed in HOD of sink byte.
3. Resultant bytes are placed in specified positions of sink FXB (FA/FB). When FA/FB are filled with processed bytes, FA/FB data are stored in a specified SDB.

DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 1 OF 8)

MVN and MVZ Data Flow



NC, OC and XC Data Flow

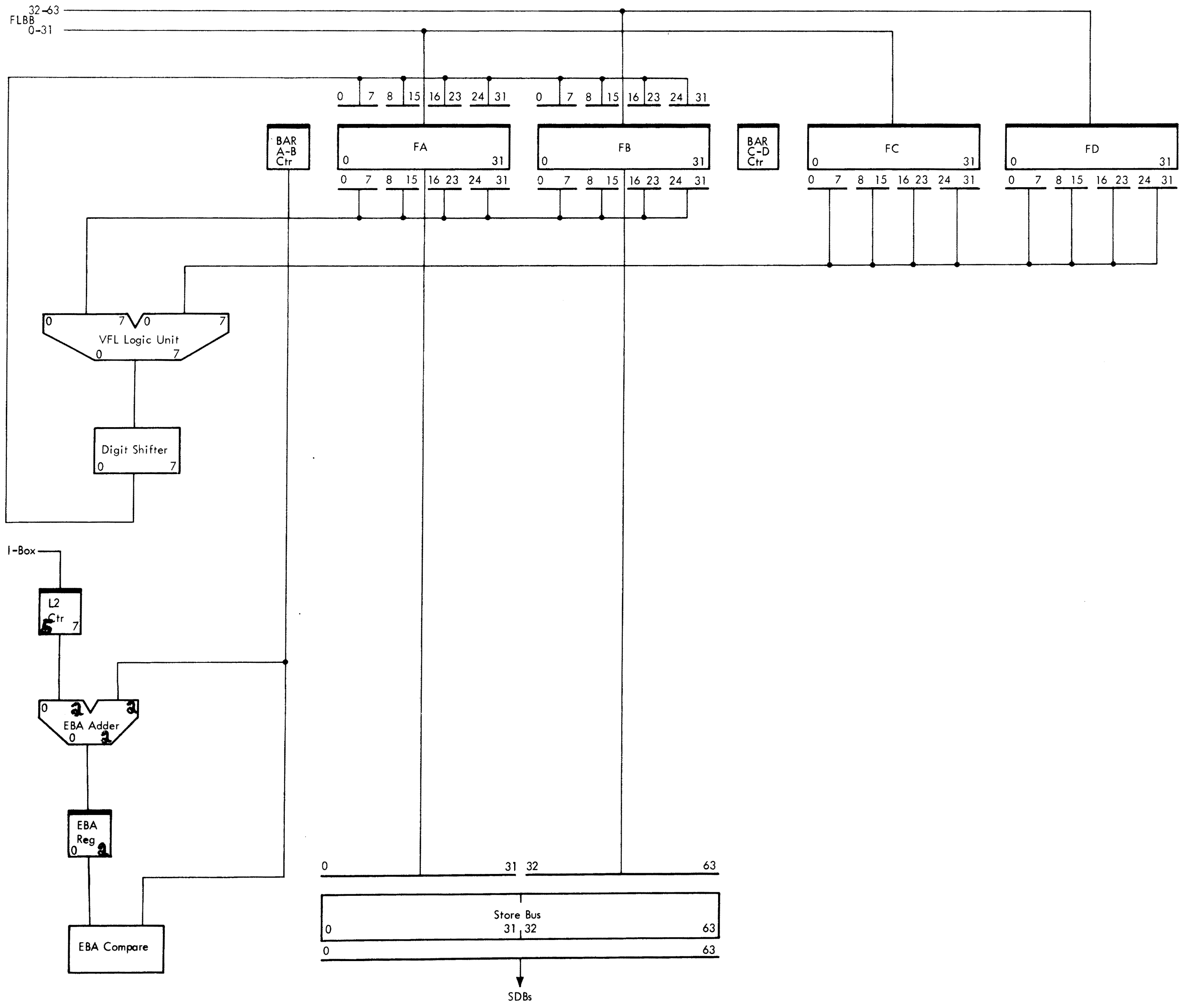


DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 2 OF 8)

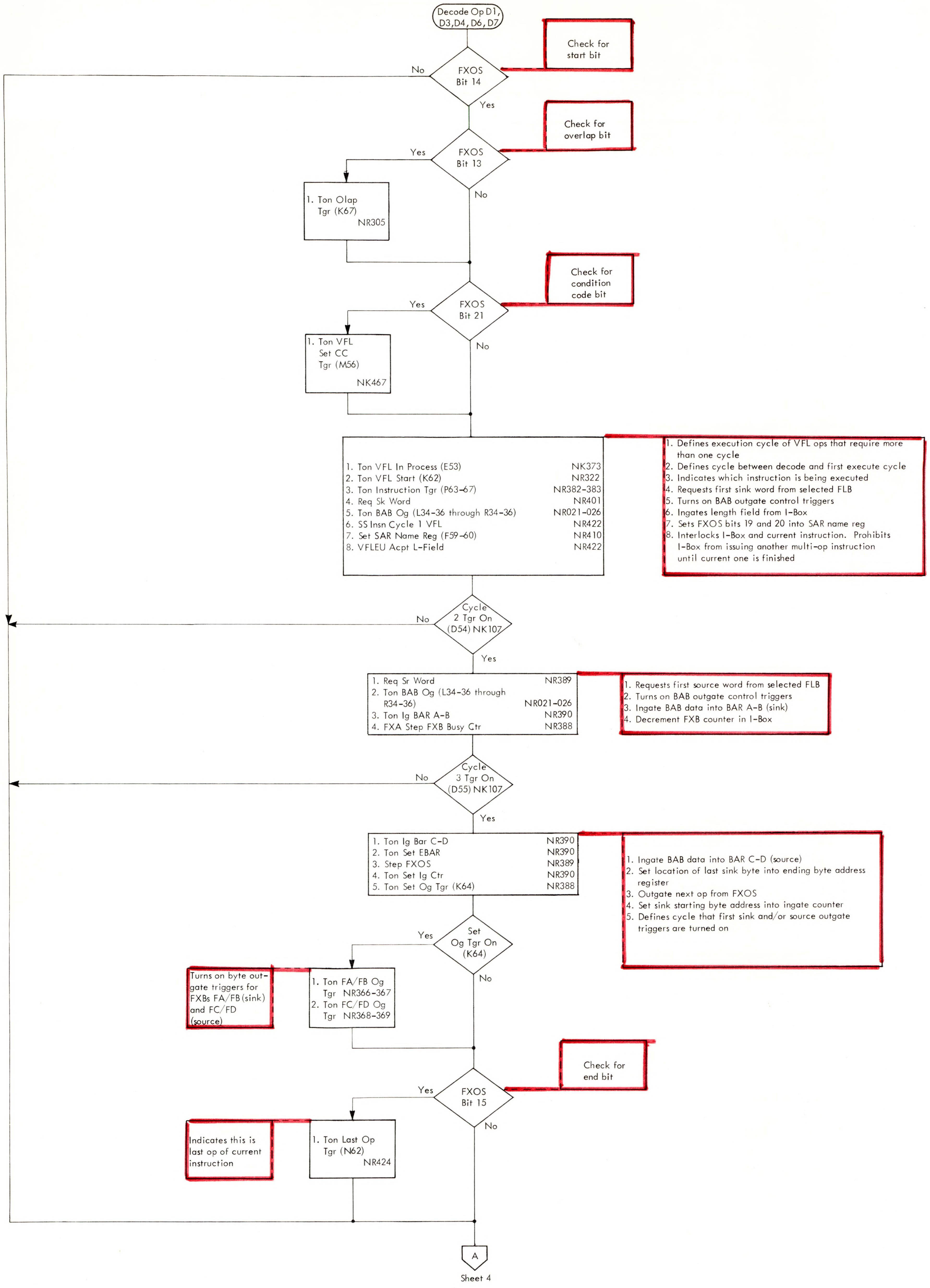

Decode Op D1, D3, D4, D6, D7
Check for start bit
FXOS Bit 14
No
Yes
Check for overlap bit
FXOS Bit 13
Yes
No
1. Ton Olap Tgr (K67) NR305
Check for condition code bit
FXOS Bit 21
Yes
No
1. Ton VFL Set CC Tgr (M56) NK467
1. Ton VFL In Process (E53) NK373
2. Ton VFL Start (K62) NR322
3. Ton Instruction Tgr (P63-67) NR382-383
4. Req Sk Word NR401
5. Ton BAB Og (L34-36 through R34-36) NR021-026
6. SS Insn Cycle 1 VFL NR422
7. Set SAR Name Reg (F59-60) NR410
8. VFLEU Acpt L-Field NR422
1. Defines execution cycle of VFL ops that require more than one cycle
2. Defines cycle between decode and first execute cycle
3. Indicates which instruction is being executed
4. Requests first sink word from selected FLB
5. Turns on BAB outgate control triggers
6. Ingates length field from I-Box
7. Sets FXOS bits 19 and 20 into SAR name reg
8. Interlocks I-Box and current instruction. Prohibits I-Box from issuing another multi-op instruction until current one is finished
Cycle 2 Tgr On (D54) NK107
No
Yes
1. Req Sr Word NR389
2. Ton BAB Og (L34-36 through R34-36) NR021-026
3. Ton Ig BAR A-B NR390
4. FXA Step FXB Busy Ctr NR388
1. Requests first source word from selected FLB
2. Turns on BAB outgate control triggers
3. Ingate BAB data into BAR A-B (sink)
4. Decrement FXB counter in I-Box
Cycle 3 Tgr On (D55) NK107
No
Yes
1. Ton Ig Bar C-D NR390
2. Ton Set EBAR NR390
3. Step FXOS NR389
4. Ton Set Ig Ctr NR390
5. Ton Set Og Tgr (K64) NR388
1. Ingate BAB data into BAR C-D (source)
2. Set location of last sink byte into ending byte address register
3. Outgate next op from FXOS
4. Set sink starting byte address into ingate counter
5. Defines cycle that first sink and/or source outgate triggers are turned on
Set Og Tgr On (K64)
Yes
No
Turns on byte outgate triggers for FXBs FA/FB (sink) and FC/FD (source)
1. Ton FA/FB Og Tgr NR366-367
2. Ton FC/FD Og Tgr NR368-369
Check for end bit
FXOS Bit 15
Yes
No
Indicates this is last op of current instruction
1. Ton Last Op Tgr (N62) NR424
A
Sheet 4

DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 3 OF 8)

Data Available

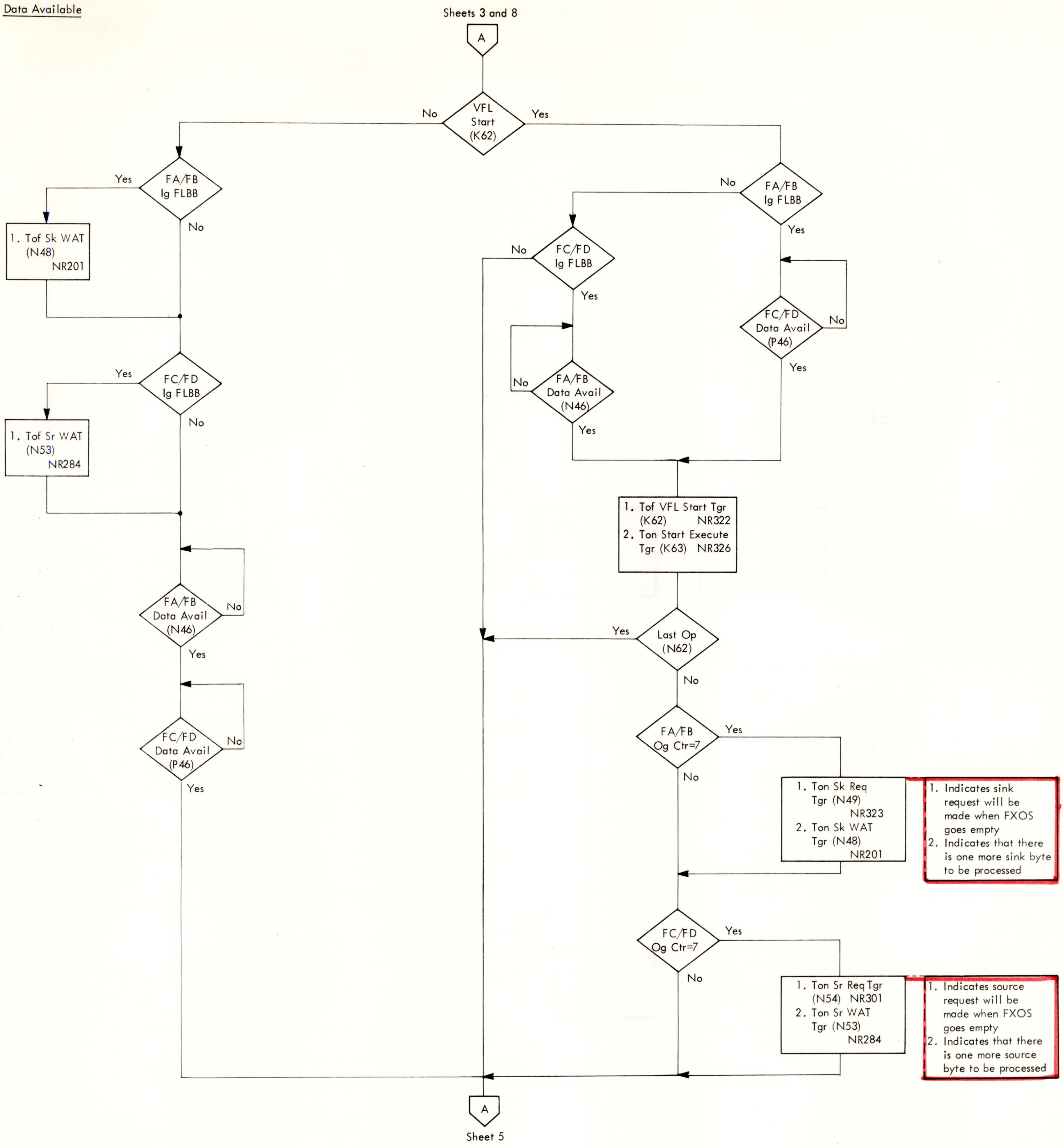

Sheets 3 and 8
A
VFL Start (K62)
No
Yes
FA/FB Ig FLBB
Yes
No
1. Tof Sk WAT (N48) NR201
FC/FD Ig FLBB
Yes
No
1. Tof Sr WAT (N53) NR284
FA/FB Data Avail (N46)
No
Yes
FC/FD Data Avail (P46)
No
Yes
FA/FB Ig FLBB
No
Yes
FC/FD Ig FLBB
No
Yes
FA/FB Data Avail (N46)
No
Yes
FC/FD Data Avail (P46)
No
Yes
1. Tof VFL Start Tgr (K62) NR322
2. Ton Start Execute Tgr (K63) NR326
Last Op (N62)
Yes
No
FA/FB Og Ctr=7
Yes
No
1. Ton Sk Req Tgr (N49) NR323
2. Ton Sk WAT Tgr (N48) NR201
1. Indicates sink request will be made when FXOS goes empty
2. Indicates that there is one more sink byte to be processed
FC/FD Og Ctr=7
Yes
No
1. Ton Sr Req Tgr (N54) NR301
2. Ton Sr WAT Tgr (N53) NR284
1. Indicates source request will be made when FXOS goes empty
2. Indicates that there is one more source byte to be processed
A
Sheet 5

DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 4 OF 8)

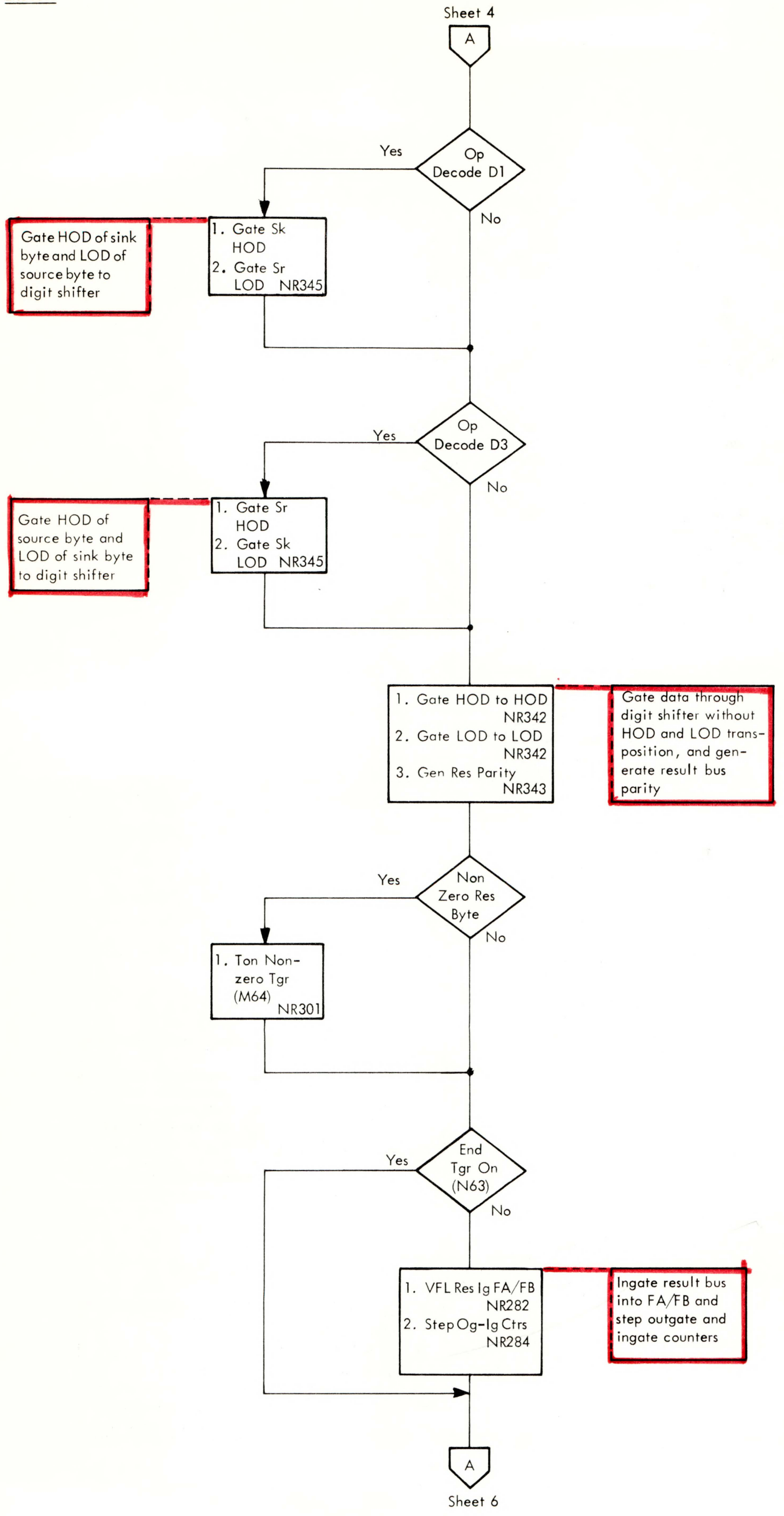

Sheet 4
A
Op Decode D1
Yes
No
1. Gate Sk HOD
2. Gate Sr LOD NR345
Gate HOD of sink byte and LOD of source byte to digit shifter
Op Decode D3
Yes
No
1. Gate Sr HOD
2. Gate Sk LOD NR345
Gate HOD of source byte and LOD of sink byte to digit shifter
1. Gate HOD to HOD NR342
2. Gate LOD to LOD NR342
3. Gen Res Parity NR343
Gate data through digit shifter without HOD and LOD transposition, and generate result bus parity
Non Zero Res Byte
Yes
No
1. Ton Non-zero Tgr (M64) NR301
End Tgr On (N63)
Yes
No
1. VFL Res Ig FA/FB NR282
2. Step Og-Ig Ctrs NR284
Ingate result bus into FA/FB and step outgate and ingate counters
A
Sheet 6

DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 5 OF 8)

Sink Request

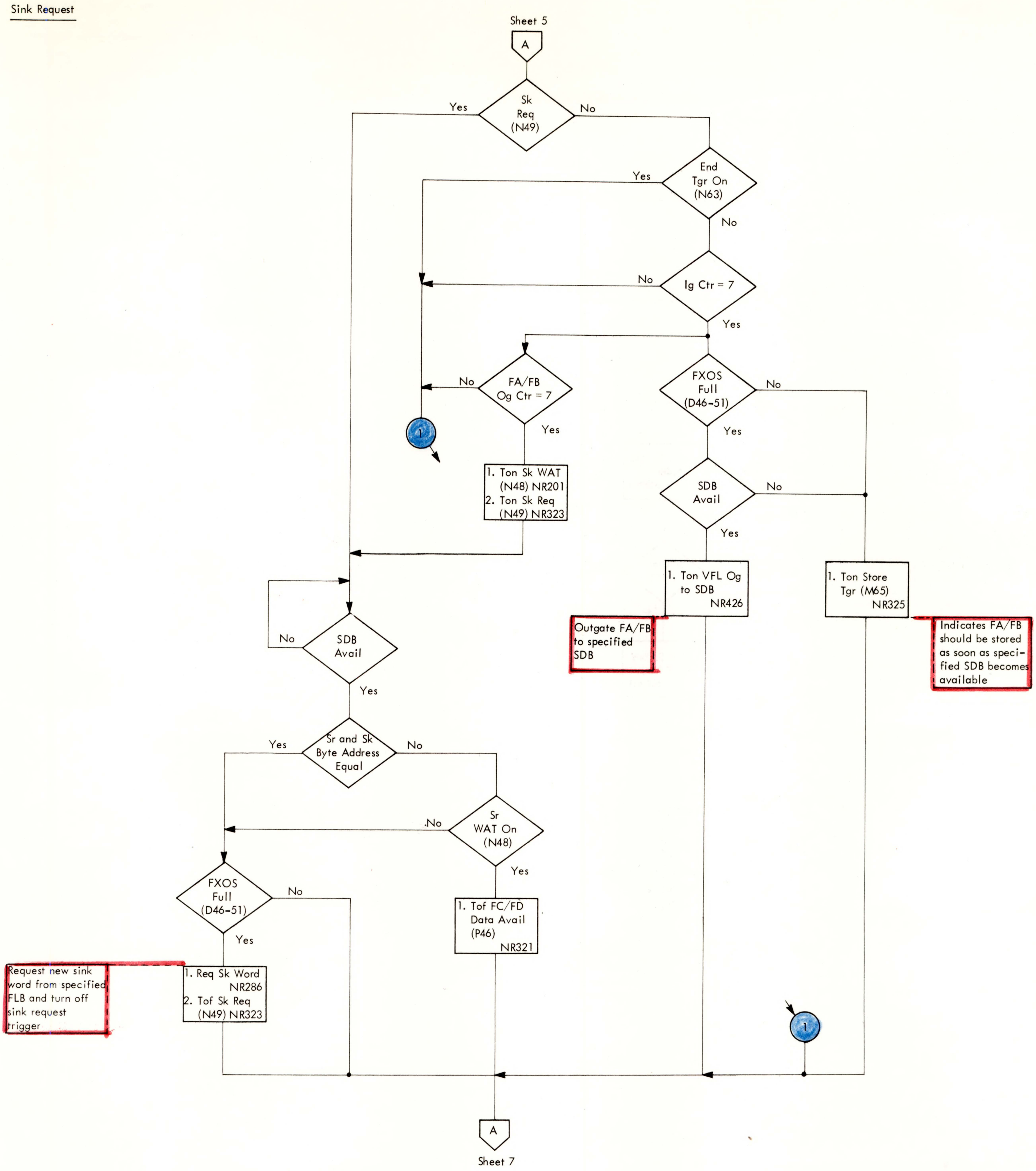

Sheet 5
A
Sk Req (N49)
Yes
No
End Tgr On (N63)
Yes
No
Ig Ctr = 7
No
Yes
FA/FB Og Ctr = 7
No
Yes
FXOS Full (D46-51)
No
Yes
1
1. Ton Sk WAT (N48) NR201
2. Ton Sk Req (N49) NR323
SDB Avail
No
Yes
1. Ton VFL Og to SDB NR426
1. Ton Store Tgr (M65) NR325
Outgate FA/FB to specified SDB
Indicates FA/FB should be stored as soon as specified SDB becomes available
SDB Avail
No
Yes
Sr and Sk Byte Address Equal
Yes
No
Sr WAT On (N48)
.No
Yes
FXOS Full (D46-51)
No
Yes
1. Tof FC/FD Data Avail (P46) NR321
Request new sink word from specified FLB and turn off sink request trigger
1. Req Sk Word NR286
2. Tof Sk Req (N49) NR323
1
A
Sheet 7

DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 6 OF 8)

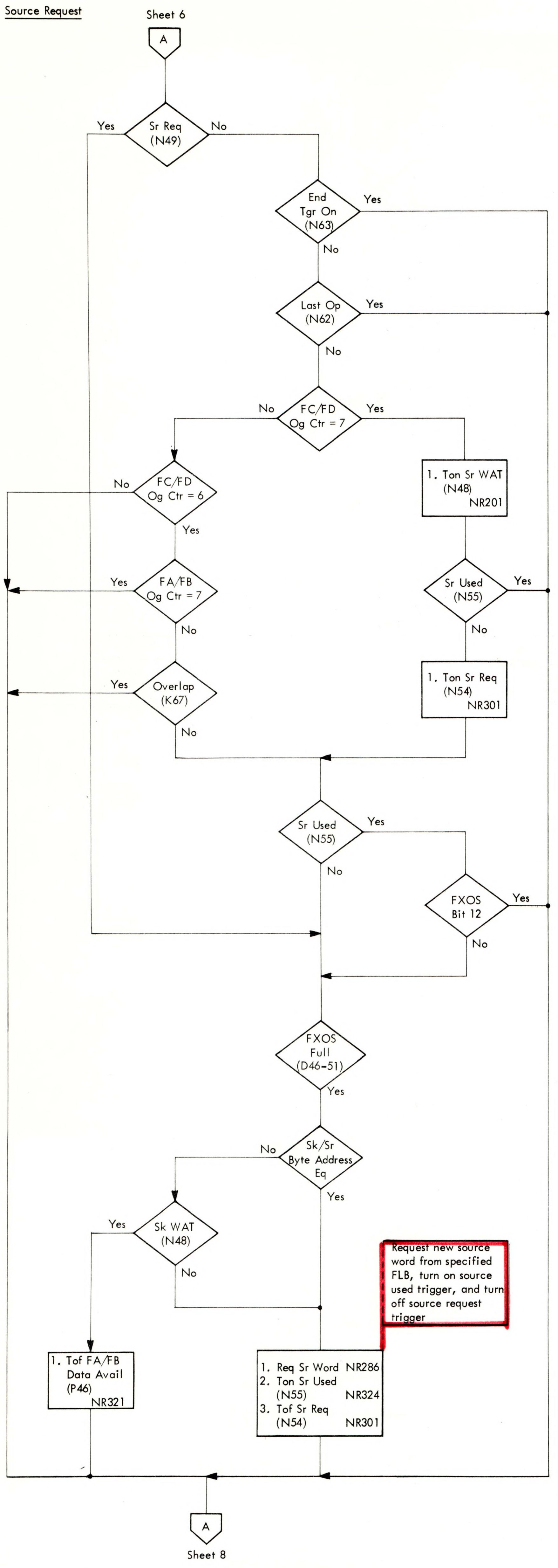


DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 7 OF 8)

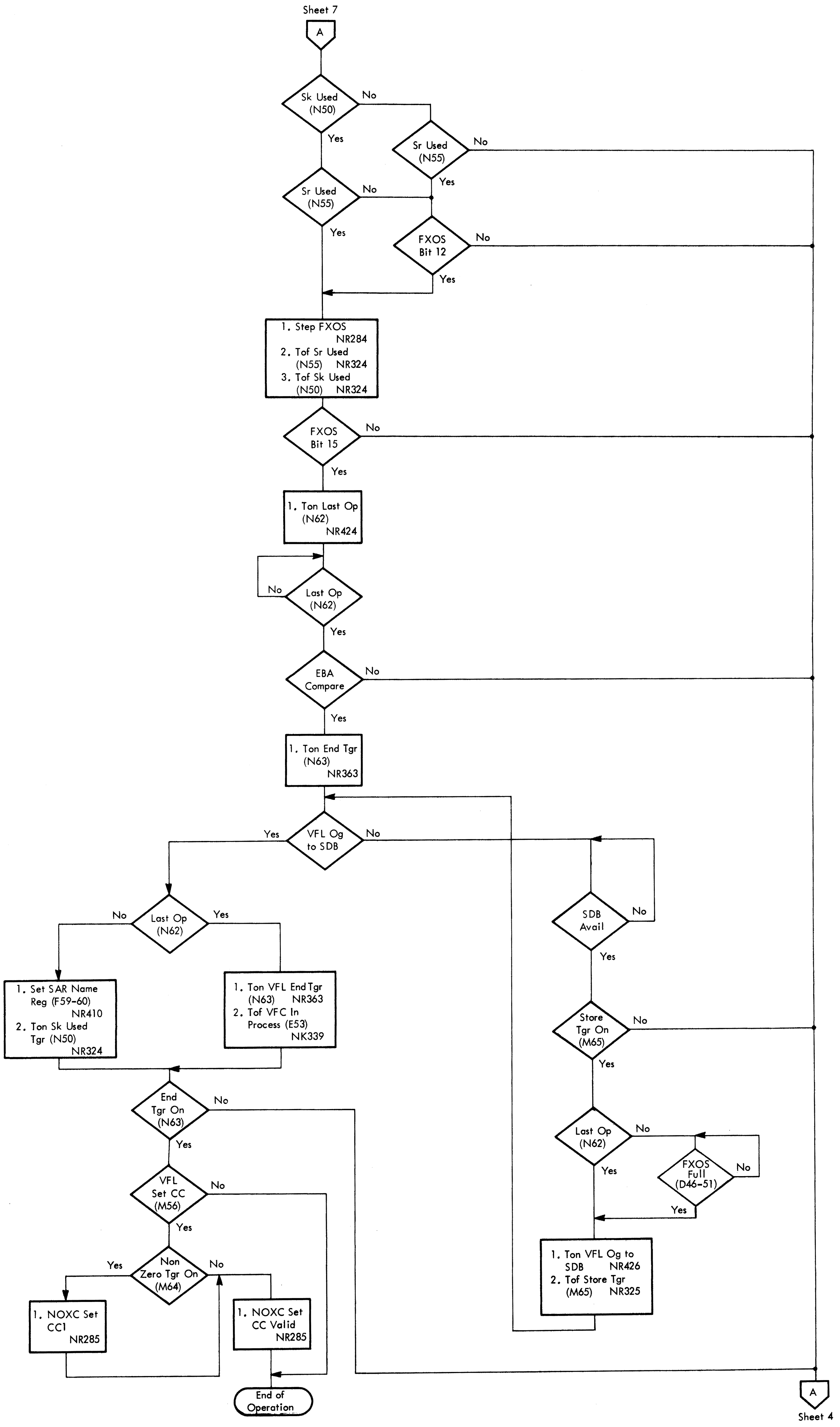


DIAGRAM 5-120. NC, OC, XC, MVN, AND MVZ INSTRUCTIONS (SHEET 8 OF 8)

Objectives:

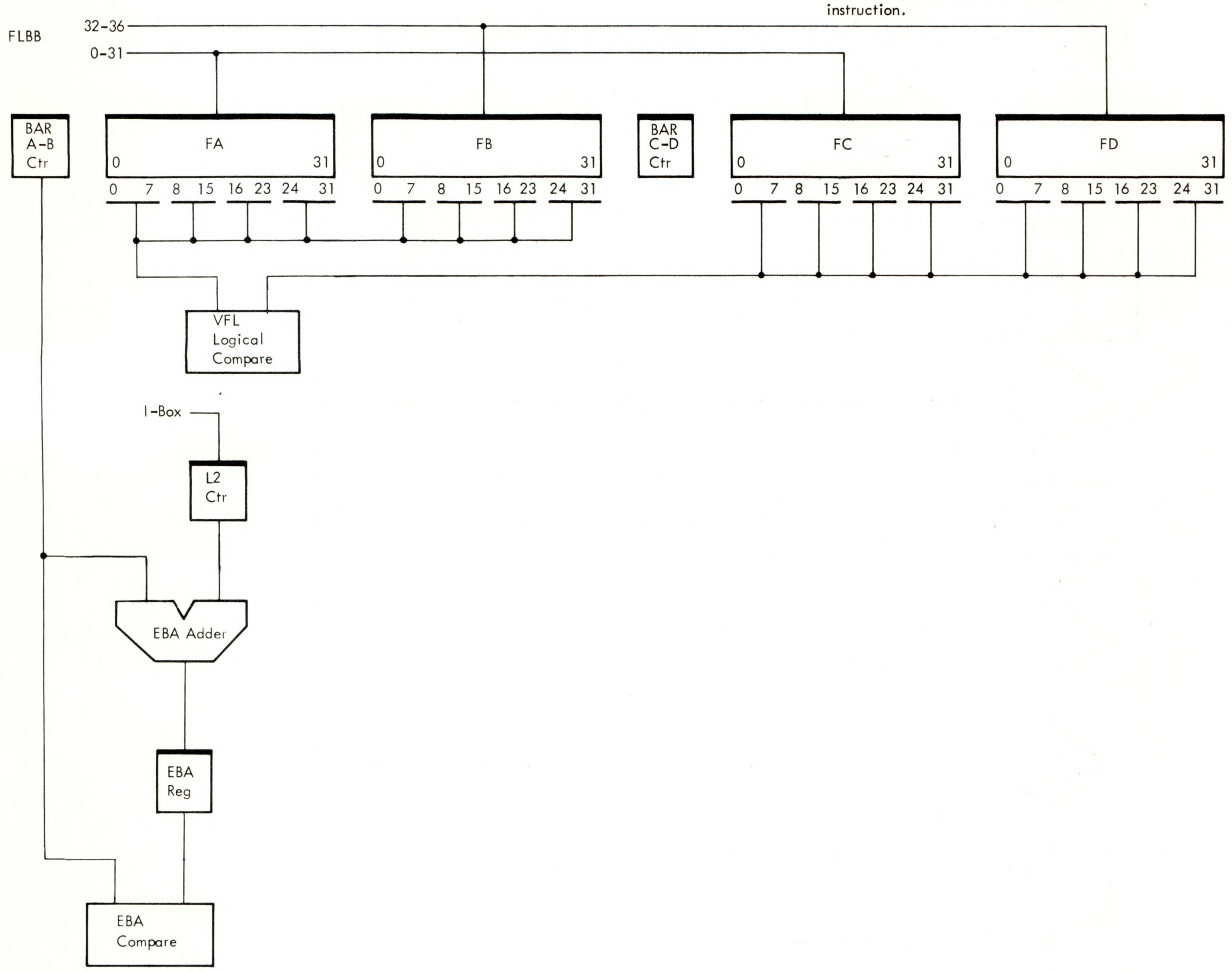
Compare Logical
CLC

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
SS	D5								▨	FLB Source			P	O	S	E	FLB Sink			▨	

1. Binary comparison is made between sink and source bytes.
2. Sink and source words are gated from FLBs to FXBs, FA/FB and FC/FD, respectively.
3. Bytes are gated to VFL comparison unit where bits are binarily compared.
4. Results of comparison are used to set condition code.
5. An unequal compare terminates instruction.

CLC Condition Code:
0 – Operands are equal
1 – Sink operand is low
2 – Sink operand is high
3 – (Not applicable)

Data Flow



Simplified Flow Chart

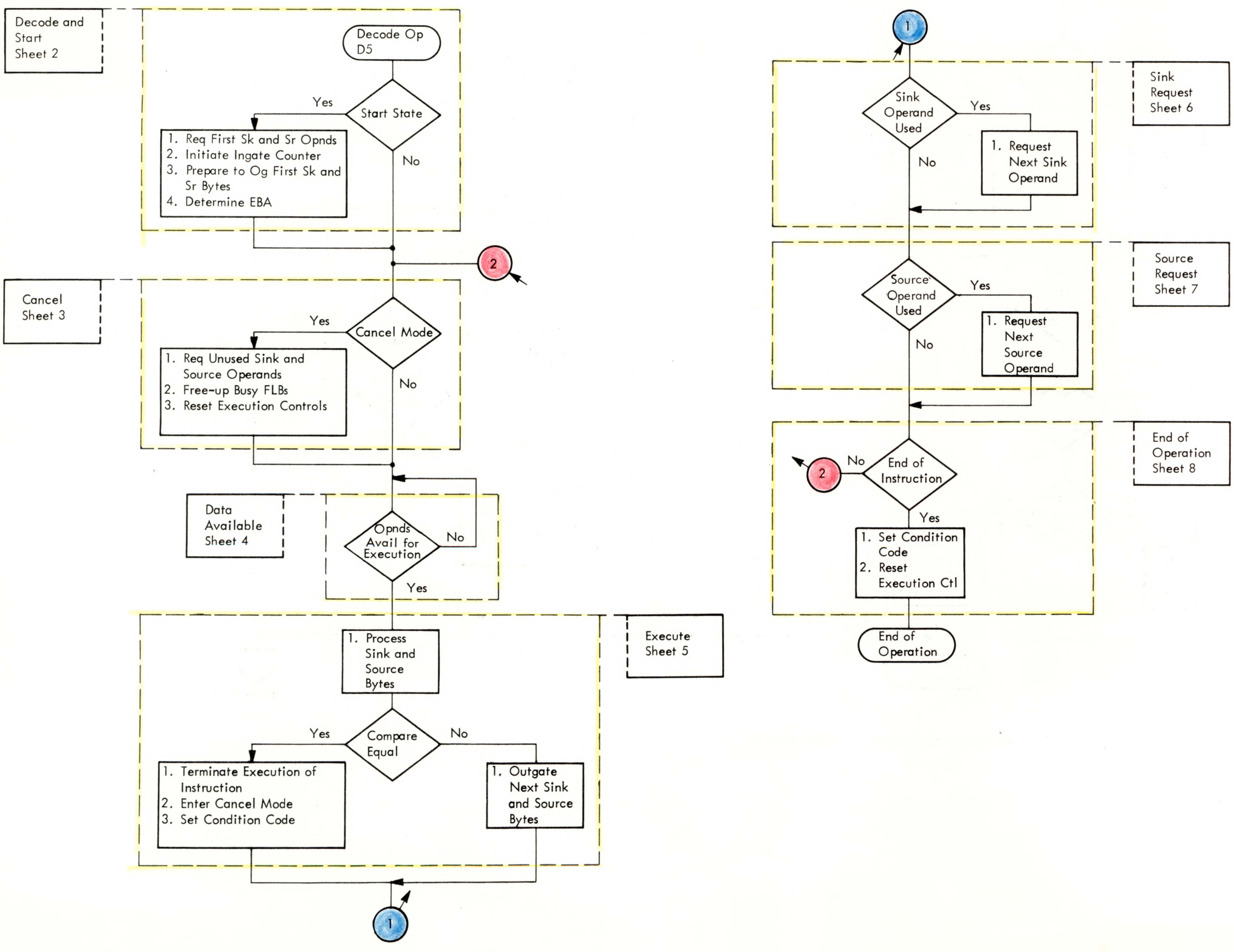


DIAGRAM 5-121. CLC INSTRUCTION (SHEET 1 OF 8)

Decode and Start

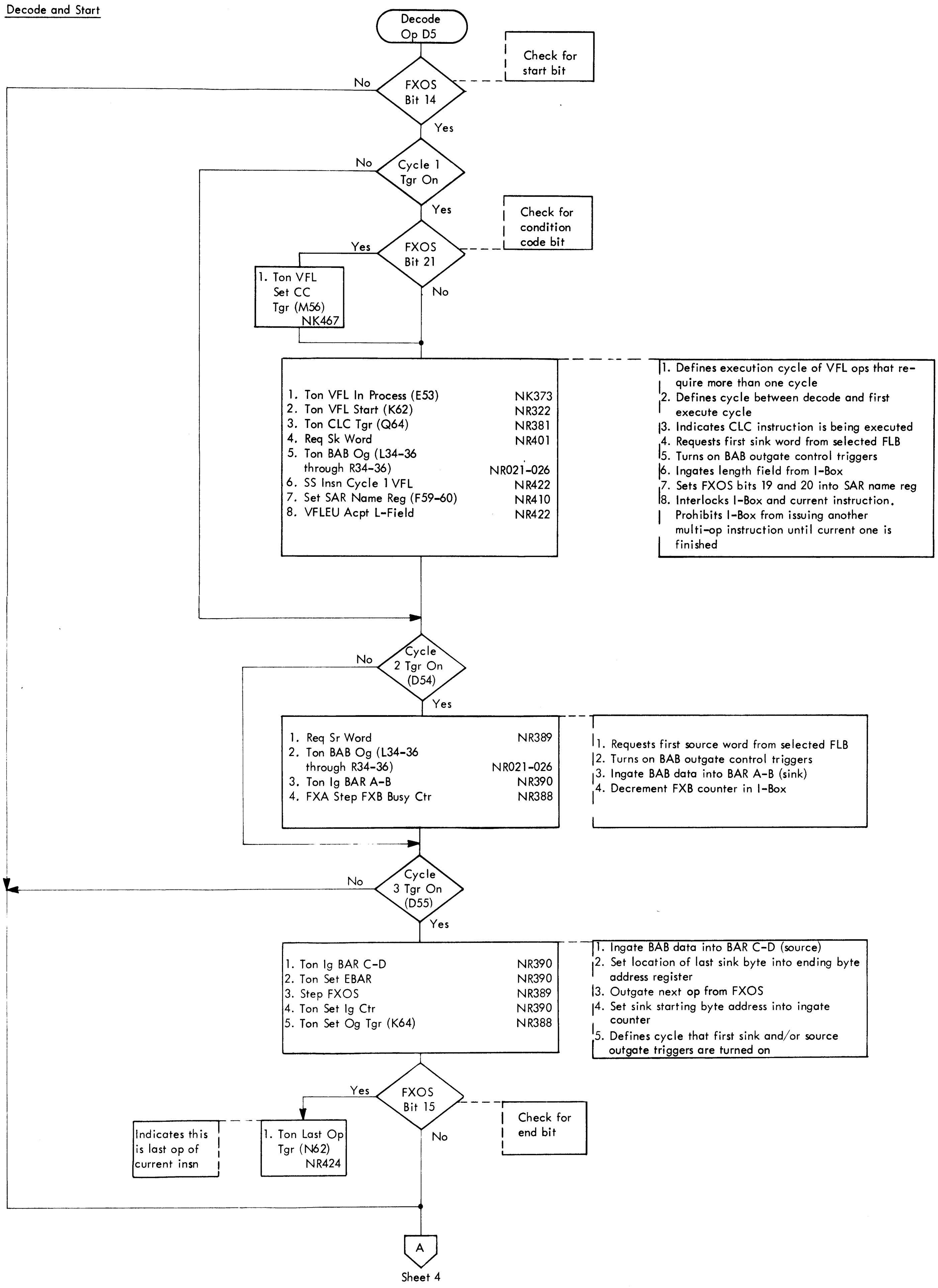

Decode Op D5
Check for start bit
FXOS Bit 14
No
Yes
Cycle 1 Tgr On
No
Yes
Check for condition code bit
FXOS Bit 21
Yes
No
1. Ton VFL Set CC Tgr (M56) NK467
1. Ton VFL In Process (E53) NK373
2. Ton VFL Start (K62) NR322
3. Ton CLC Tgr (Q64) NR381
4. Req Sk Word NR401
5. Ton BAB Og (L34-36 through R34-36) NR021-026
6. SS Insn Cycle 1 VFL NR422
7. Set SAR Name Reg (F59-60) NR410
8. VFLEU Acpt L-Field NR422
1. Defines execution cycle of VFL ops that require more than one cycle
2. Defines cycle between decode and first execute cycle
3. Indicates CLC instruction is being executed
4. Requests first sink word from selected FLB
5. Turns on BAB outgate control triggers
6. Ingates length field from I-Box
7. Sets FXOS bits 19 and 20 into SAR name reg
8. Interlocks I-Box and current instruction. Prohibits I-Box from issuing another multi-op instruction until current one is finished
Cycle 2 Tgr On (D54)
No
Yes
1. Req Sr Word NR389
2. Ton BAB Og (L34-36 through R34-36) NR021-026
3. Ton Ig BAR A-B NR390
4. FXA Step FXB Busy Ctr NR388
1. Requests first source word from selected FLB
2. Turns on BAB outgate control triggers
3. Ingate BAB data into BAR A-B (sink)
4. Decrement FXB counter in I-Box
Cycle 3 Tgr On (D55)
No
Yes
1. Ton Ig BAR C-D NR390
2. Ton Set EBAR NR390
3. Step FXOS NR389
4. Ton Set Ig Ctr NR390
5. Ton Set Og Tgr (K64) NR388
1. Ingate BAB data into BAR C-D (source)
2. Set location of last sink byte into ending byte address register
3. Outgate next op from FXOS
4. Set sink starting byte address into ingate counter
5. Defines cycle that first sink and/or source outgate triggers are turned on
FXOS Bit 15
Yes
No
Check for end bit
Indicates this is last op of current insn
1. Ton Last Op Tgr (N62) NR424
A
Sheet 4

DIAGRAM 5-121. CLC INSTRUCTION (SHEET 2 OF 8)

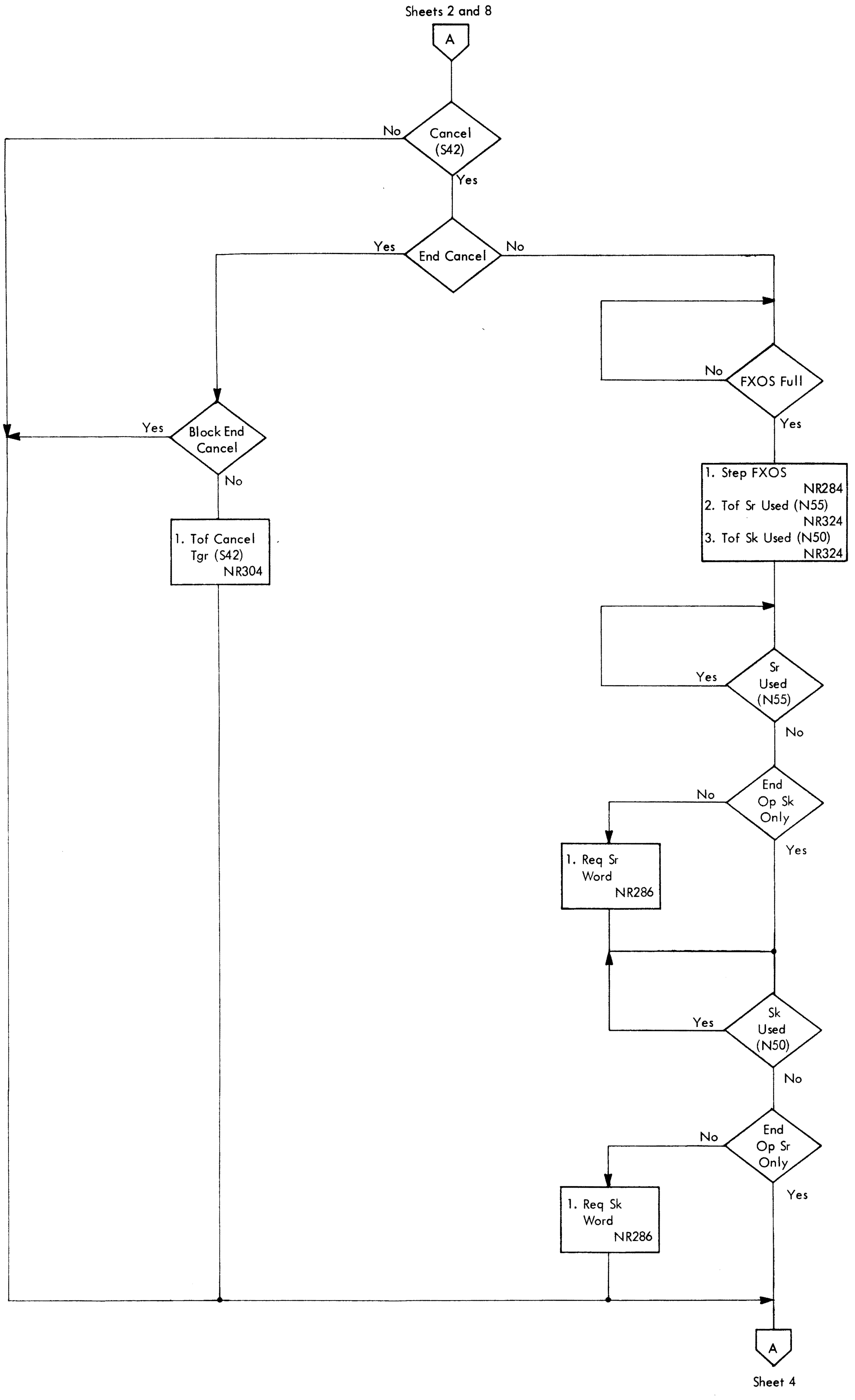


DIAGRAM 5-121. CLC INSTRUCTION (SHEET 3 OF 8)

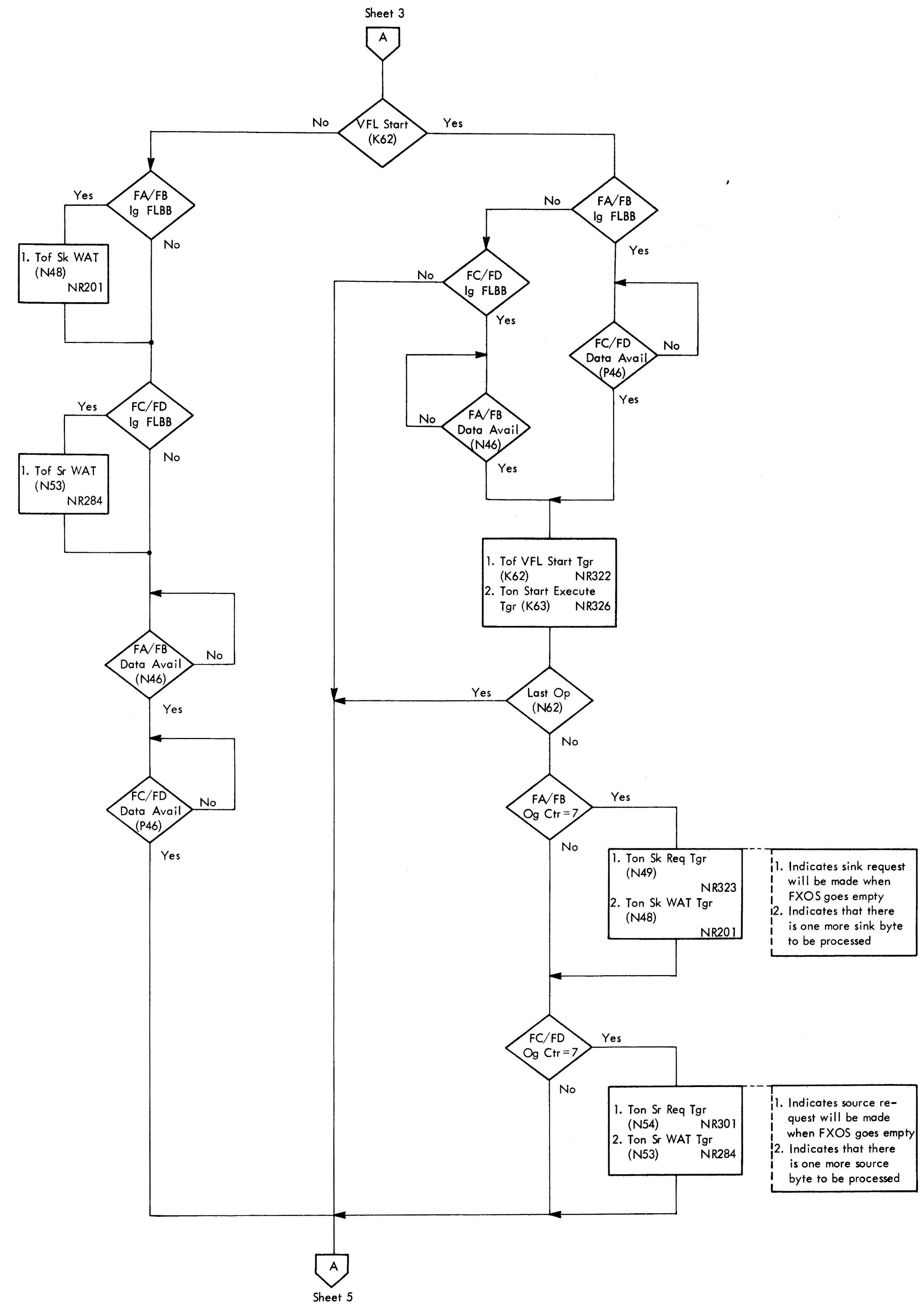

Sheet 3
A
VFL Start (K62)
No
Yes
FA/FB Ig FLBB
Yes
No
1. Tof Sk WAT (N48) NR201
FC/FD Ig FLBB
Yes
No
1. Tof Sr WAT (N53) NR284
FA/FB Data Avail (N46)
No
Yes
FC/FD Data Avail (P46)
No
Yes
FA/FB Ig FLBB
No
Yes
FC/FD Ig FLBB
No
Yes
FA/FB Data Avail (N46)
No
Yes
FC/FD Data Avail (P46)
No
Yes
1. Tof VFL Start Tgr (K62) NR322
2. Ton Start Execute Tgr (K63) NR326
Last Op (N62)
Yes
No
FA/FB Og Ctr = 7
Yes
No
1. Ton Sk Req Tgr (N49) NR323
2. Ton Sk WAT Tgr (N48) NR201
1. Indicates sink request will be made when FXOS goes empty
2. Indicates that there is one more sink byte to be processed
FC/FD Og Ctr = 7
Yes
No
1. Ton Sr Req Tgr (N54) NR301
2. Ton Sr WAT Tgr (N53) NR284
1. Indicates source request will be made when FXOS goes empty
2. Indicates that there is one more source byte to be processed
A
Sheet 5

DIAGRAM 5-121. CLC INSTRUCTION (SHEET 4 OF 8)

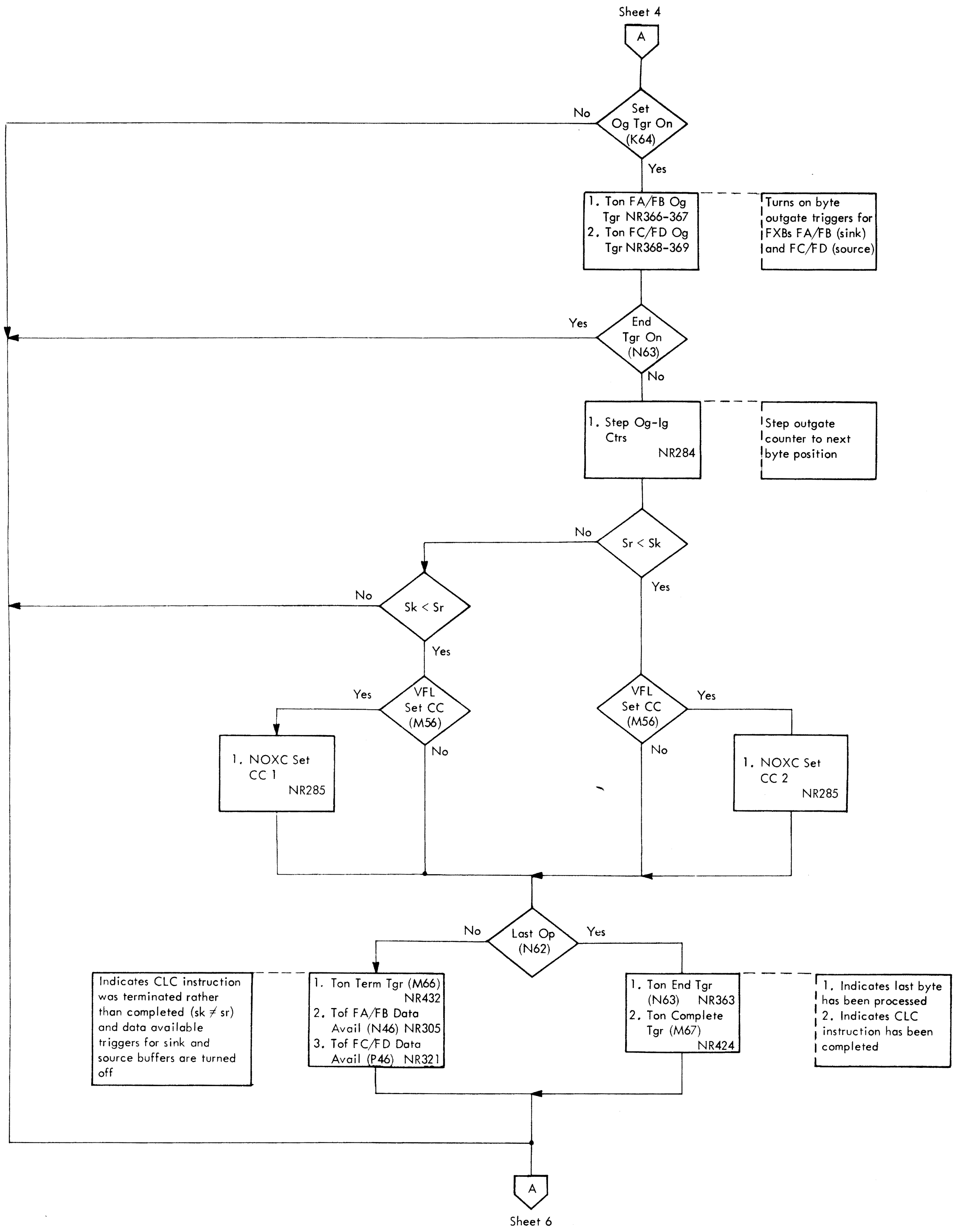
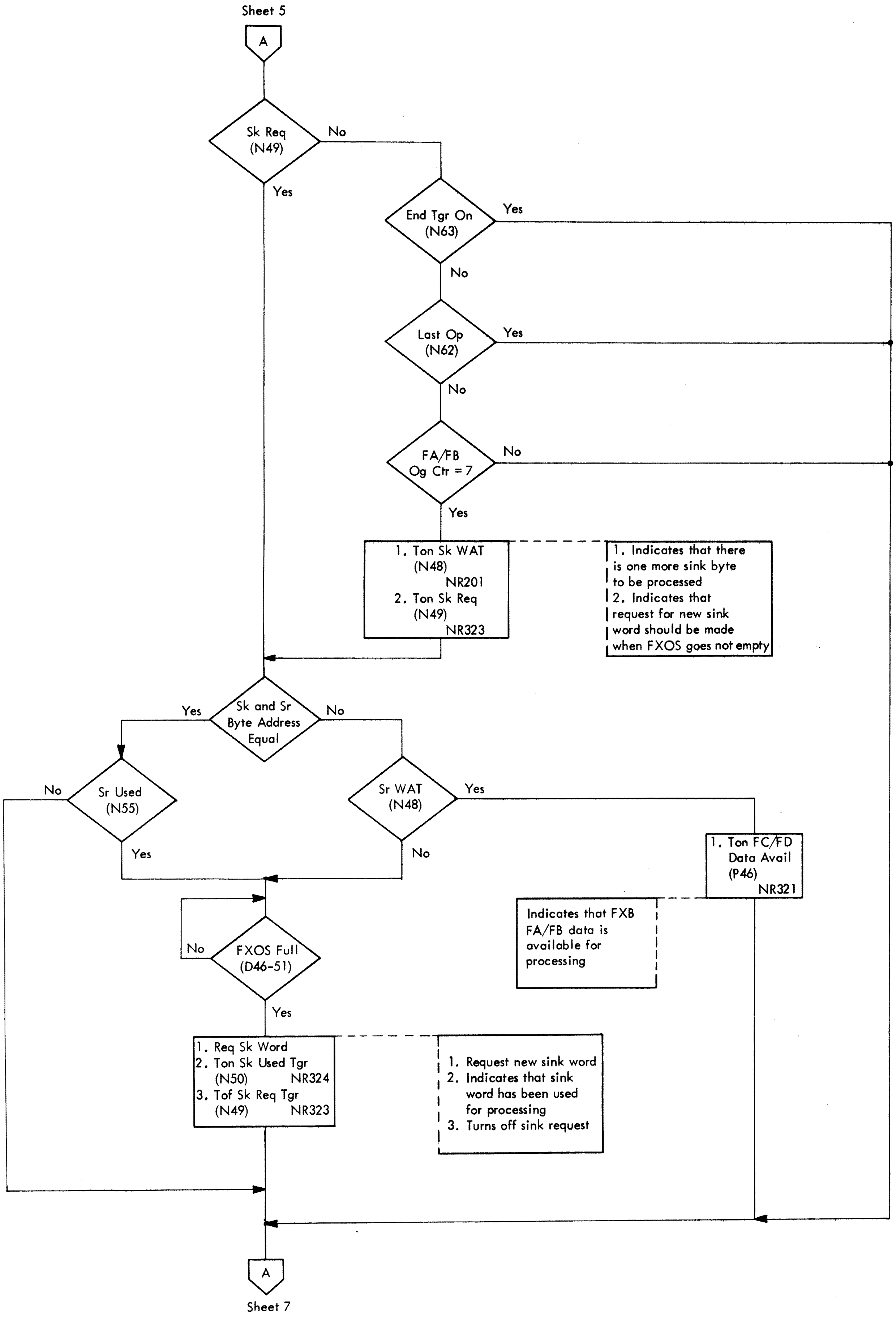

Sheet 4
A
Set Og Tgr On (K64)
No
Yes
1. Ton FA/FB Og Tgr NR366-367
2. Ton FC/FD Og Tgr NR368-369
Turns on byte outgate triggers for FXBs FA/FB (sink) and FC/FD (source)
End Tgr On (N63)
Yes
No
1. Step Og-Ig Ctrs NR284
Step outgate counter to next byte position
Sr < Sk
No
Yes
Sk < Sr
No
Yes
VFL Set CC (M56)
Yes
No
1. NOXC Set CC 1 NR285
VFL Set CC (M56)
Yes
No
1. NOXC Set CC 2 NR285
Last Op (N62)
No
Yes
Indicates CLC instruction was terminated rather than completed (sk ≠ sr) and data available triggers for sink and source buffers are turned off
1. Ton Term Tgr (M66) NR432
2. Tof FA/FB Data Avail (N46) NR305
3. Tof FC/FD Data Avail (P46) NR321
1. Ton End Tgr (N63) NR363
2. Ton Complete Tgr (M67) NR424
1. Indicates last byte has been processed
2. Indicates CLC instruction has been completed
A
Sheet 6

DIAGRAM 5-121. CLC INSTRUCTION (SHEET 5 OF 8)

Sink Request



Sheet 5
A
Sk Req (N49)
No
Yes
End Tgr On (N63)
Yes
No
Last Op (N62)
Yes
No
FA/FB Og Ctr = 7
No
Yes
1. Ton Sk WAT (N48) NR201
2. Ton Sk Req (N49) NR323
1. Indicates that there is one more sink byte to be processed
2. Indicates that request for new sink word should be made when FXOS goes not empty
Sk and Sr Byte Address Equal
Yes
No
Sr Used (N55)
No
Yes
Sr WAT (N48)
Yes
No
1. Ton FC/FD Data Avail (P46) NR321
Indicates that FXB FA/FB data is available for processing
FXOS Full (D46-51)
No
Yes
1. Req Sk Word
2. Ton Sk Used Tgr (N50) NR324
3. Tof Sk Req Tgr (N49) NR323
1. Request new sink word
2. Indicates that sink word has been used for processing
3. Turns off sink request
A
Sheet 7

DIAGRAM 5-121. CLC INSTRUCTION (SHEET 6 OF 8)

Source Request

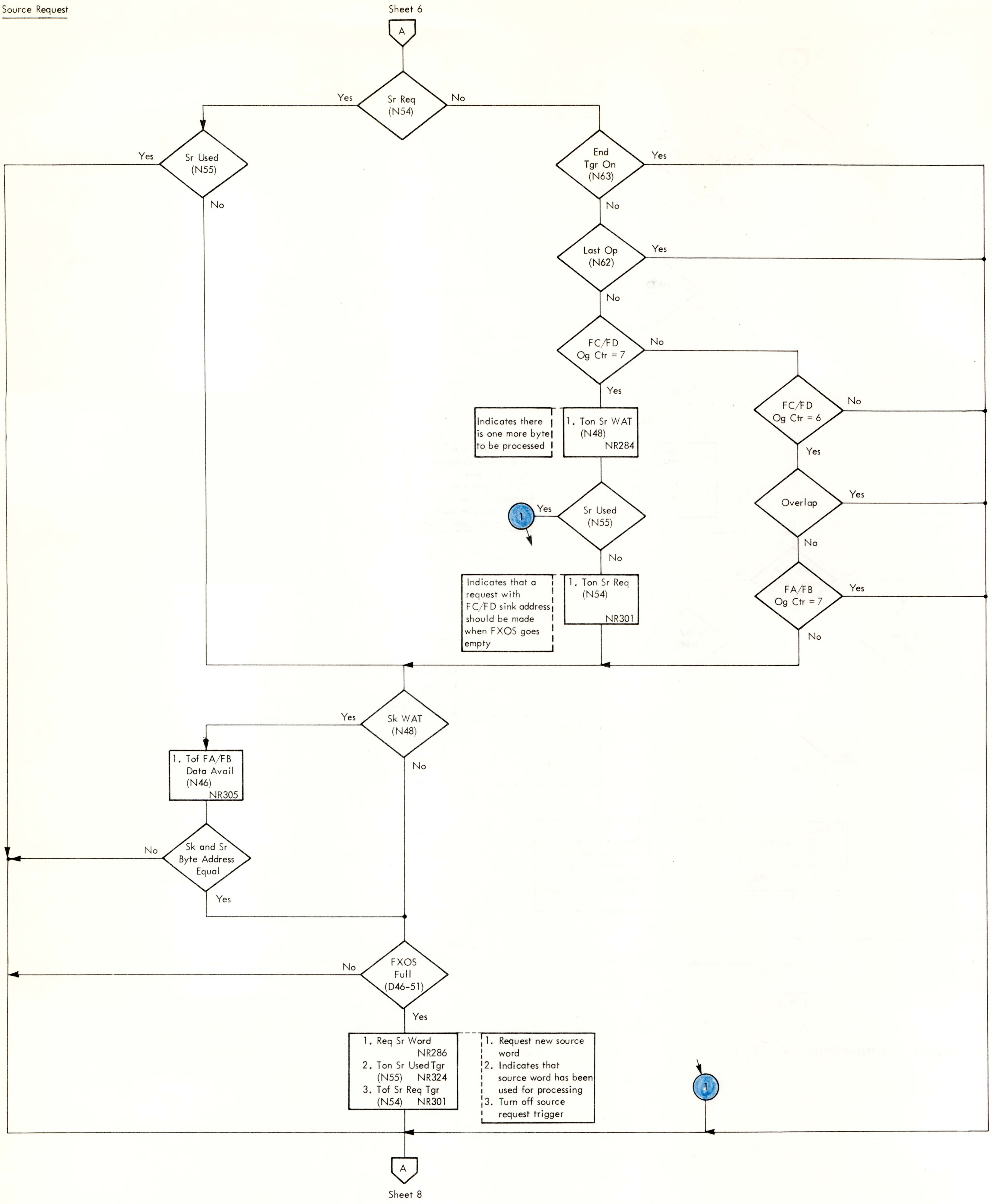

Sheet 6
A
Sr Req (N54)
Yes
No
Sr Used (N55)
Yes
No
End Tgr On (N63)
Yes
No
Last Op (N62)
Yes
No
FC/FD Og Ctr = 7
No
Yes
Indicates there is one more byte to be processed
1. Ton Sr WAT (N48) NR284
FC/FD Og Ctr = 6
No
Yes
Overlap
Yes
No
1
Yes
Sr Used (N55)
No
Indicates that a request with FC/FD sink address should be made when FXOS goes empty
1. Ton Sr Req (N54) NR301
FA/FB Og Ctr = 7
Yes
No
Sk WAT (N48)
Yes
No
1. Tof FA/FB Data Avail (N46) NR305
Sk and Sr Byte Address Equal
No
Yes
FXOS Full (D46-51)
No
Yes
1. Req Sr Word NR286
2. Ton Sr Used Tgr (N55) NR324
3. Tof Sr Req Tgr (N54) NR301
1. Request new source word
2. Indicates that source word has been used for processing
3. Turn off source request trigger
1
A
Sheet 8

DIAGRAM 5-121. CLC INSTRUCTION (SHEET 7 OF 8)

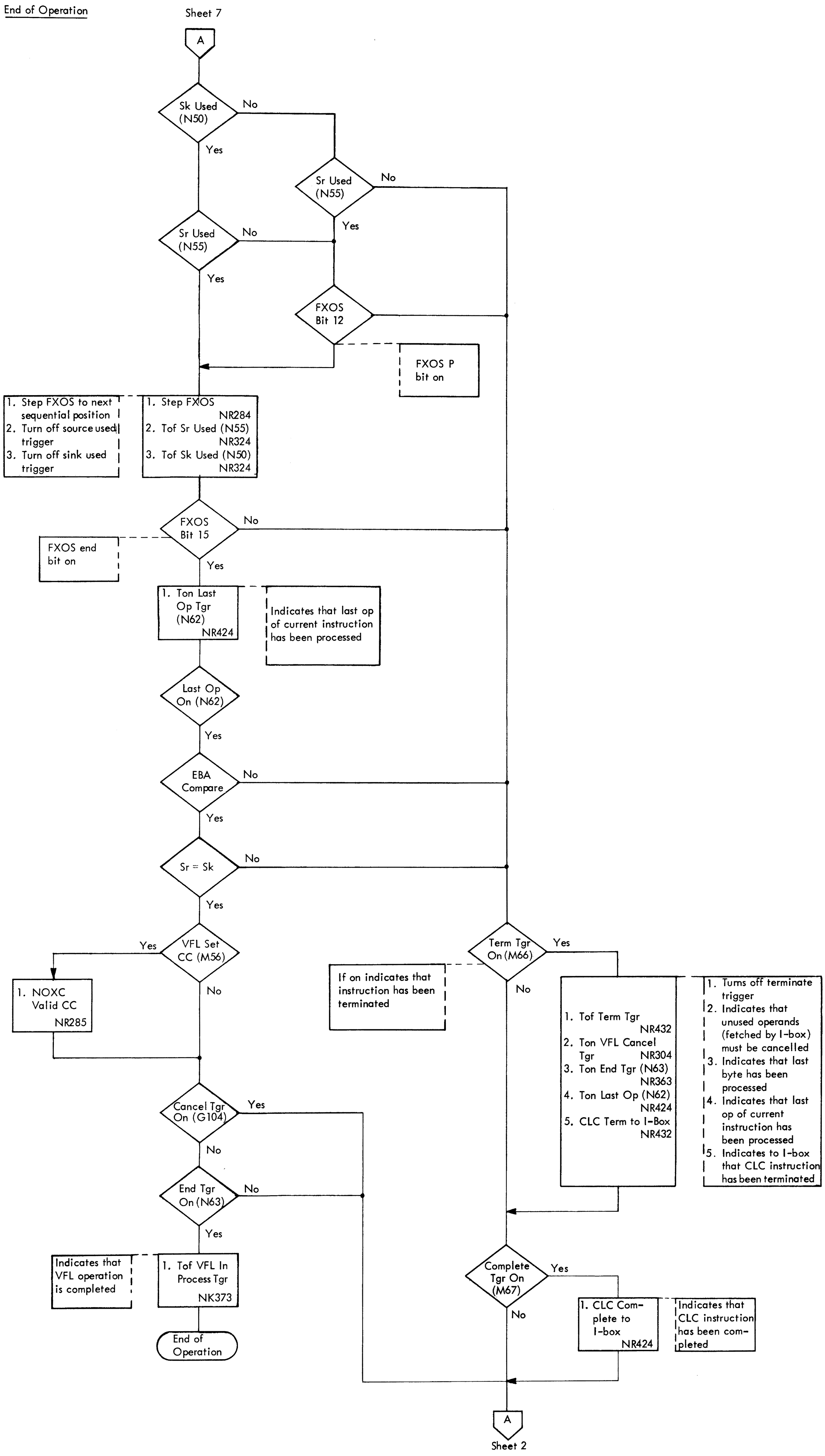

End of Operation
Sheet 7
A
Sk Used (N50)
No
Yes
Sr Used (N55)
No
Yes
Sr Used (N55)
No
Yes
FXOS Bit 12
FXOS P bit on
1. Step FXOS to next sequential position
2. Turn off source used trigger
3. Turn off sink used trigger
1. Step FXOS NR284
2. Tof Sr Used (N55) NR324
3. Tof Sk Used (N50) NR324
FXOS Bit 15
No
FXOS end bit on
Yes
1. Ton Last Op Tgr (N62) NR424
Indicates that last op of current instruction has been processed
Last Op On (N62)
Yes
EBA Compare
No
Yes
Sr = Sk
No
Yes
VFL Set CC (M56)
Yes
1. NOXC Valid CC NR285
No
Cancel Tgr On (G104)
Yes
No
End Tgr On (N63)
No
Yes
Indicates that VFL operation is completed
1. Tof VFL In Process Tgr NK373
End of Operation
Term Tgr On (M66)
Yes
If on indicates that instruction has been terminated
No
1. Tof Term Tgr NR432
2. Ton VFL Cancel Tgr NR304
3. Ton End Tgr (N63) NR363
4. Ton Last Op (N62) NR424
5. CLC Term to I-Box NR432
1. Turns off terminate trigger
2. Indicates that unused operands (fetched by I-box) must be cancelled
3. Indicates that last byte has been processed
4. Indicates that last op of current instruction has been processed
5. Indicates to I-box that CLC instruction has been terminated
Complete Tgr On (M67)
Yes
No
1. CLC Complete to I-box NR424
Indicates that CLC instruction has been completed
A
Sheet 2

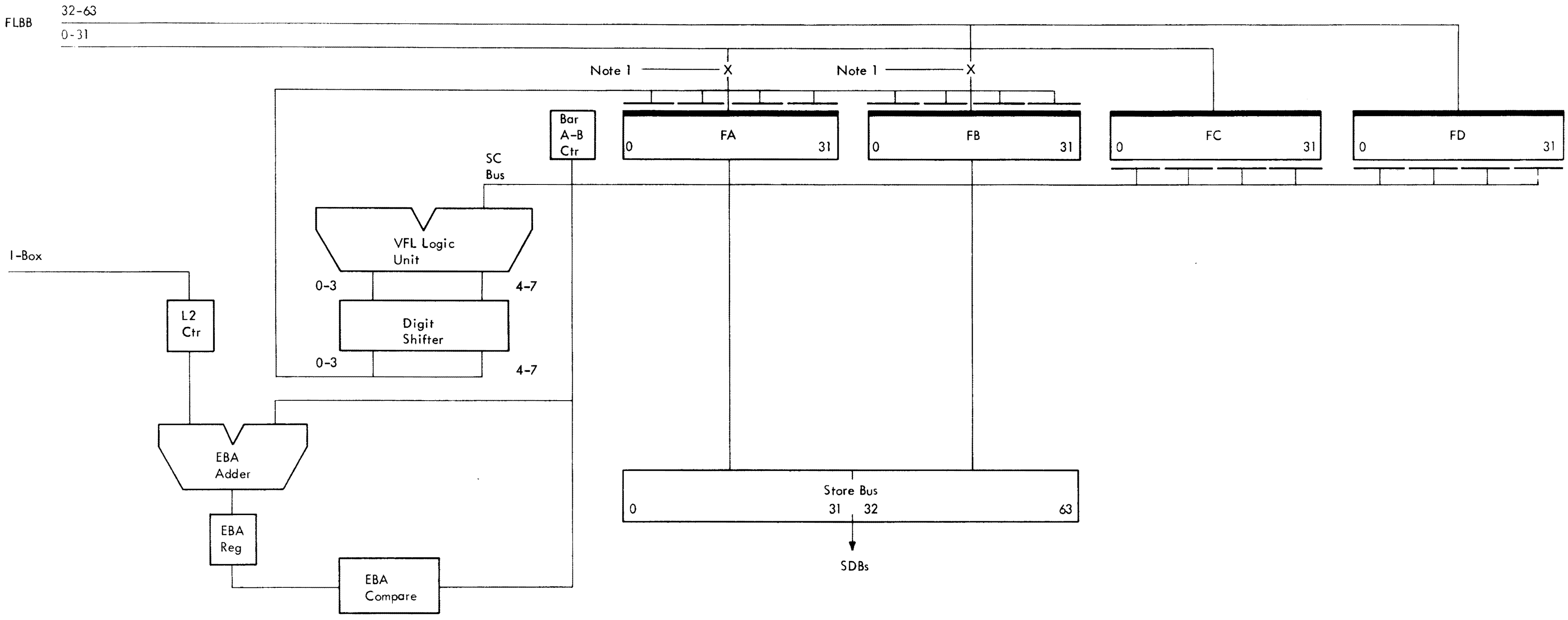
DIAGRAM 5-121. CLC INSTRUCTION (SHEET 8 OF 8)

Objectives:
Move Character
MVC

0	1 2 3 4 5 6 7	8	9 10 11	12	13	14	15	16 17 18	19 20
SS	D2		FLB Name	P	O	S	E	Starting Byte Adr	SAR Name

1. Specified FLB is gated to FA/FB and FC/FD during parallel operation (P bit = 1) or FC/FD only during serial operation (P bit = 0).
2. Data is moved one byte at a time from FC/FD to FA/FB during serial operation.
3. When all byte positions of FA/FB are filled, FA/FB data is gated to specified SDB.

Data Flow



Notes:
1. Parallel processing only

Sheet Layout

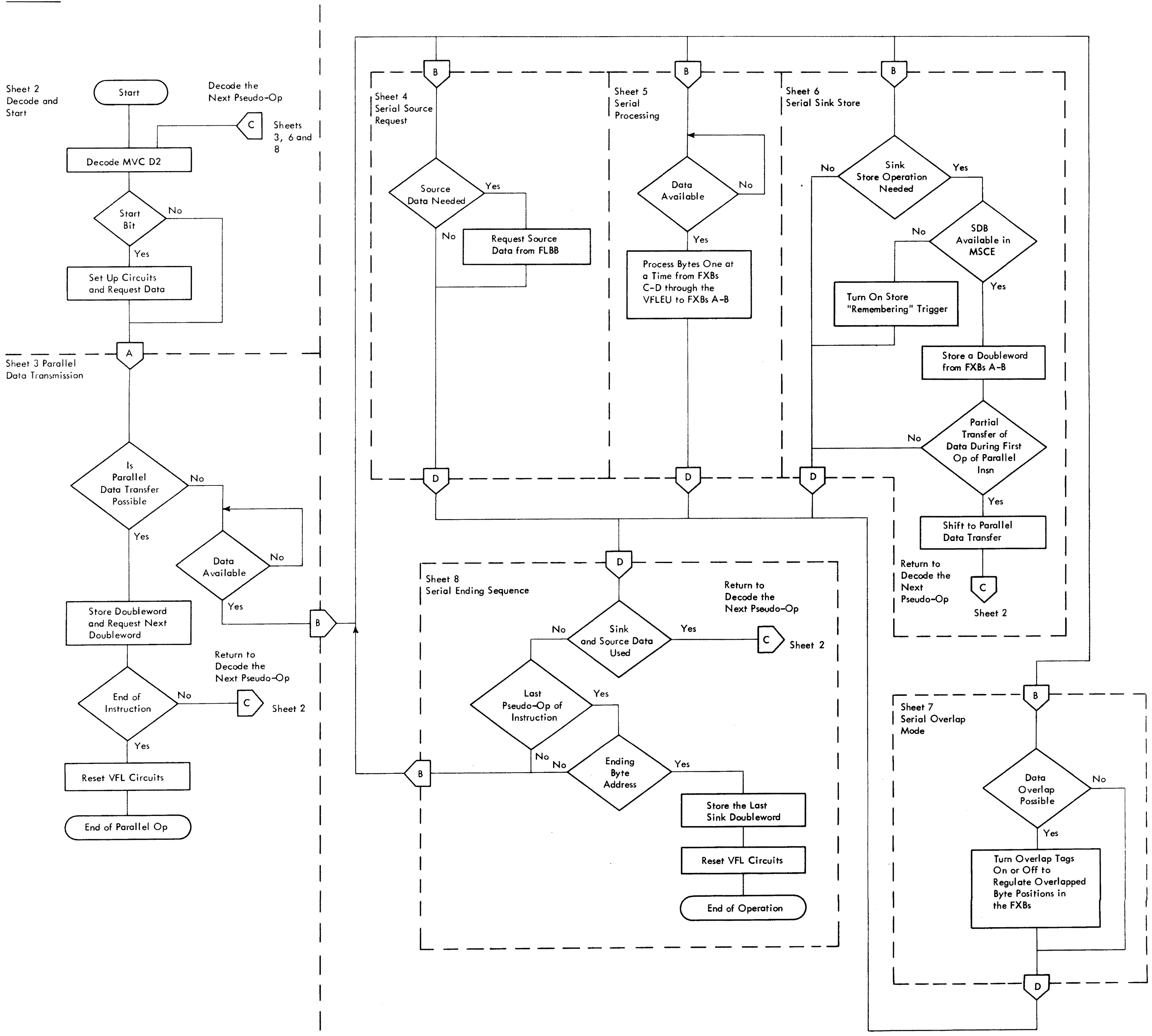


DIAGRAM 5-122. MVC INSTRUCTION (SHEET 1 OF 8)

Decode and Start.

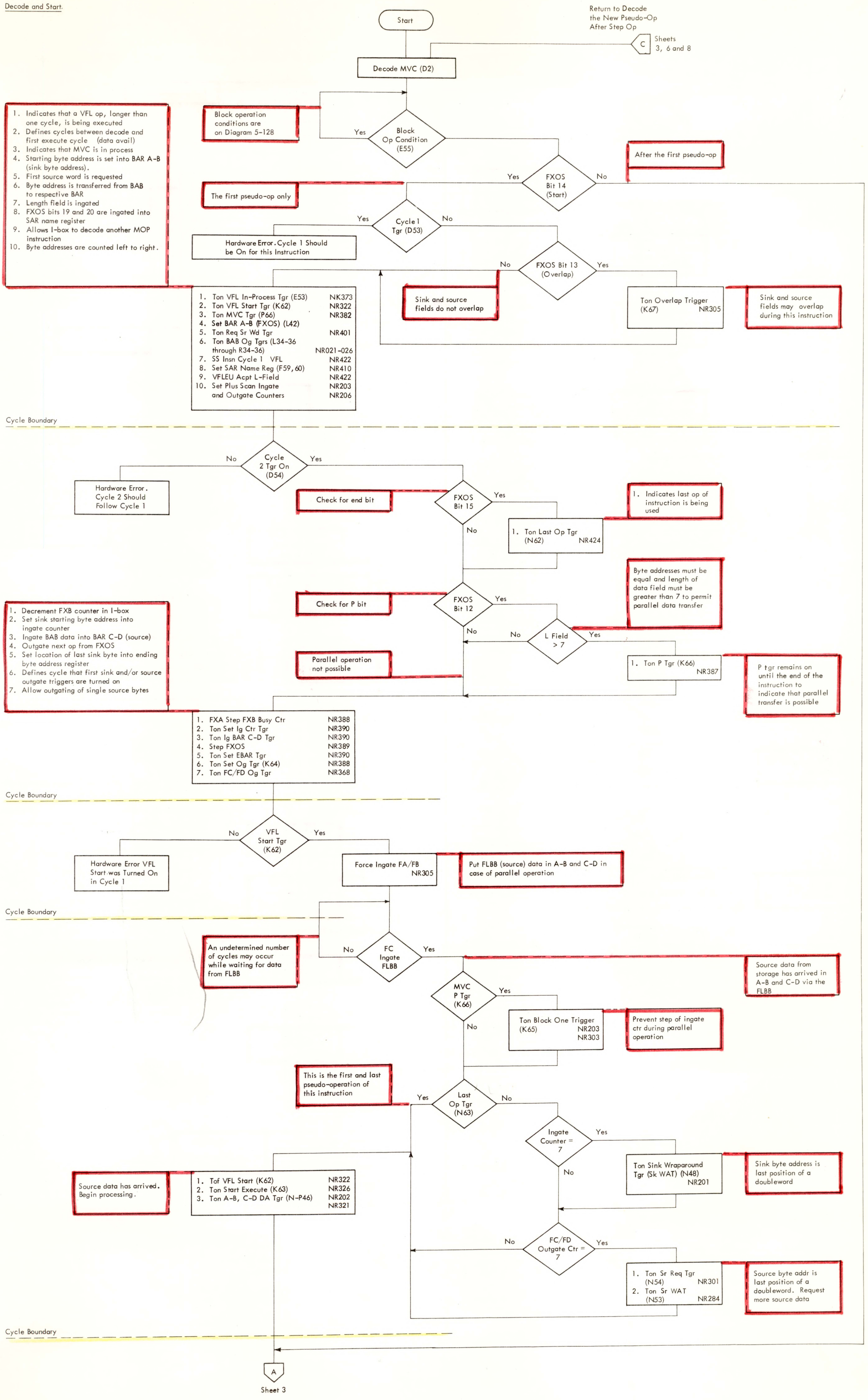
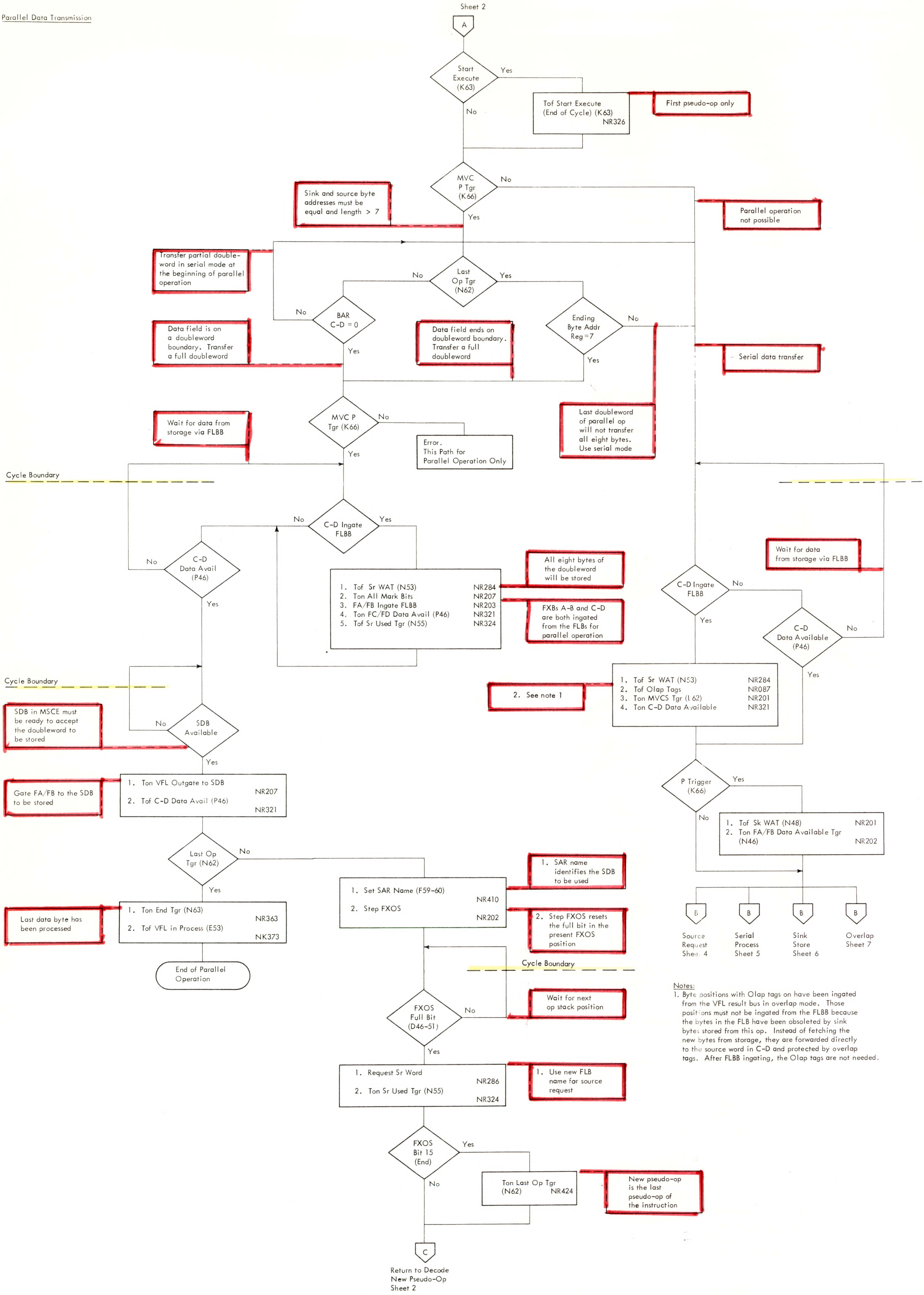

Start
Return to Decode
the New Pseudo-Op
After Step Op
C
Sheets
3, 6 and 8
Decode MVC (D2)
1. Indicates that a VFL op, longer than one cycle, is being executed
2. Defines cycles between decode and first execute cycle (data avail)
3. Indicates that MVC is in process
4. Starting byte address is set into BAR A-B (sink byte address).
5. First source word is requested
6. Byte address is transferred from BAB to respective BAR
7. Length field is ingated
8. FXOS bits 19 and 20 are ingated into SAR name register
9. Allows I-box to decode another MOP instruction
10. Byte addresses are counted left to right.
Block operation conditions are on Diagram 5-128
Yes
Block
Op Condition
(E55)
After the first pseudo-op
Yes
FXOS
Bit 14
(Start)
No
The first pseudo-op only
Yes
Cycle 1
Tgr (D53)
No
Hardware Error. Cycle 1 Should be On for this Instruction
No
FXOS Bit 13
(Overlap)
Yes
Sink and source fields do not overlap
Ton Overlap Trigger
(K67) NR305
Sink and source fields may overlap during this instruction
1. Ton VFL In-Process Tgr (E53) NK373
2. Ton VFL Start Tgr (K62) NR322
3. Ton MVC Tgr (P66) NR382
4. Set BAR A-B (FXOS) (L42)
5. Ton Req Sr Wd Tgr NR401
6. Ton BAB Og Tgrs (L34-36 through R34-36) NR021-026
7. SS Insn Cycle 1 VFL NR422
8. Set SAR Name Reg (F59,60) NR410
9. VFLEU Acpt L-Field NR422
10. Set Plus Scan Ingate and Outgate Counters NR203 NR206
Cycle Boundary
No
Cycle
2 Tgr On
(D54)
Yes
Hardware Error. Cycle 2 Should Follow Cycle 1
Check for end bit
FXOS
Bit 15
Yes
1. Indicates last op of instruction is being used
No
1. Ton Last Op Tgr (N62) NR424
Byte addresses must be equal and length of data field must be greater than 7 to permit parallel data transfer
Check for P bit
FXOS
Bit 12
Yes
1. Decrement FXB counter in I-box
2. Set sink starting byte address into ingate counter
3. Ingate BAB data into BAR C-D (source)
4. Outgate next op from FXOS
5. Set location of last sink byte into ending byte address register
6. Defines cycle that first sink and/or source outgate triggers are turned on
7. Allow outgating of single source bytes
No
No
L Field
> 7
Yes
Parallel operation not possible
1. Ton P Tgr (K66) NR387
P tgr remains on until the end of the instruction to indicate that parallel transfer is possible
1. FXA Step FXB Busy Ctr NR388
2. Ton Set Ig Ctr Tgr NR390
3. Ton Ig BAR C-D Tgr NR390
4. Step FXOS NR389
5. Ton Set EBAR Tgr NR390
6. Ton Set Og Tgr (K64) NR388
7. Ton FC/FD Og Tgr NR368
Cycle Boundary
No
VFL
Start Tgr
(K62)
Yes
Hardware Error VFL Start was Turned On in Cycle 1
Force Ingate FA/FB NR305
Put FLBB (source) data in A-B and C-D in case of parallel operation
Cycle Boundary
An undetermined number of cycles may occur while waiting for data from FLBB
No
FC
Ingate
FLBB
Yes
Source data from storage has arrived in A-B and C-D via the FLBB
MVC
P Tgr
(K66)
Yes
No
Ton Block One Trigger (K65) NR203 NR303
Prevent step of ingate ctr during parallel operation
This is the first and last pseudo-operation of this instruction
Yes
Last
Op Tgr
(N63)
No
Ingate
Counter =
7
Yes
Ton Sink Wraparound Tgr (Sk WAT) (N48) NR201
Sink byte address is last position of a doubleword
No
Source data has arrived. Begin processing.
1. Tof VFL Start (K62) NR322
2. Ton Start Execute (K63) NR326
3. Ton A-B, C-D DA Tgr (N-P46) NR202 NR321
No
FC/FD
Outgate Ctr =
7
Yes
1. Ton Sr Req Tgr (N54) NR301
2. Ton Sr WAT (N53) NR284
Source byte addr is last position of a doubleword. Request more source data
Cycle Boundary
A
Sheet 3

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 2 OF 8)


Sheet 2
A
Start Execute (K63)
Yes
No
Tof Start Execute (End of Cycle) (K63) NR326
First pseudo-op only
MVC P Tgr (K66)
No
Yes
Sink and source byte addresses must be equal and length > 7
Parallel operation not possible
Transfer partial doubleword in serial mode at the beginning of parallel operation
Last Op Tgr (N62)
No
Yes
BAR C-D = 0
No
Yes
Ending Byte Addr Reg = 7
No
Yes
Data field is on a doubleword boundary. Transfer a full doubleword
Data field ends on doubleword boundary. Transfer a full doubleword
Serial data transfer
Last doubleword of parallel op will not transfer all eight bytes. Use serial mode
MVC P Tgr (K66)
No
Yes
Error. This Path for Parallel Operation Only
Wait for data from storage via FLBB
Cycle Boundary
C-D Ingate FLBB
No
Yes
C-D Data Avail (P46)
No
Yes
1. Tof Sr WAT (N53) NR284
2. Ton All Mark Bits NR207
3. FA/FB Ingate FLBB NR203
4. Ton FC/FD Data Avail (P46) NR321
5. Tof Sr Used Tgr (N55) NR324
All eight bytes of the doubleword will be stored
FXBs A-B and C-D are both ingated from the FLBs for parallel operation
Wait for data from storage via FLBB
C-D Ingate FLBB
No
Yes
C-D Data Available (P46)
No
Yes
1. Tof Sr WAT (N53) NR284
2. Tof Olap Tags NR087
3. Ton MVCS Tgr (L62) NR201
4. Ton C-D Data Available NR321
2. See note 1
Cycle Boundary
SDB in MSCE must be ready to accept the doubleword to be stored
SDB Available
No
Yes
1. Ton VFL Outgate to SDB NR207
2. Tof C-D Data Avail (P46) NR321
Gate FA/FB to the SDB to be stored
P Trigger (K66)
Yes
No
1. Tof Sk WAT (N48) NR201
2. Ton FA/FB Data Available Tgr (N46) NR202
B
Source Request Sheet 4
B
Serial Process Sheet 5
B
Sink Store Sheet 6
B
Overlap Sheet 7
Last Op Tgr (N62)
No
Yes
1. Ton End Tgr (N63) NR363
2. Tof VFL in Process (E53) NK373
Last data byte has been processed
End of Parallel Operation
1. Set SAR Name (F59-60) NR410
2. Step FXOS NR202
1. SAR name identifies the SDB to be used
2. Step FXOS resets the full bit in the present FXOS position
Cycle Boundary
FXOS Full Bit (D46-51)
No
Yes
Wait for next op stack position
1. Request Sr Word NR286
2. Ton Sr Used Tgr (N55) NR324
1. Use new FLB name for source request
FXOS Bit 15 (End)
Yes
No
Ton Last Op Tgr (N62) NR424
New pseudo-op is the last pseudo-op of the instruction
C
Return to Decode New Pseudo-Op Sheet 2

Notes:
1. Byte positions with Olap tags on have been ingated from the VFL result bus in overlap mode. Those positions must not be ingated from the FLBB because the bytes in the FLB have been obsoleted by sink bytes stored from this op. Instead of fetching the new bytes from storage, they are forwarded directly to the source word in C-D and protected by overlap tags. After FLBB ingating, the Olap tags are not needed.

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 3 OF 8)

Serial Source Request

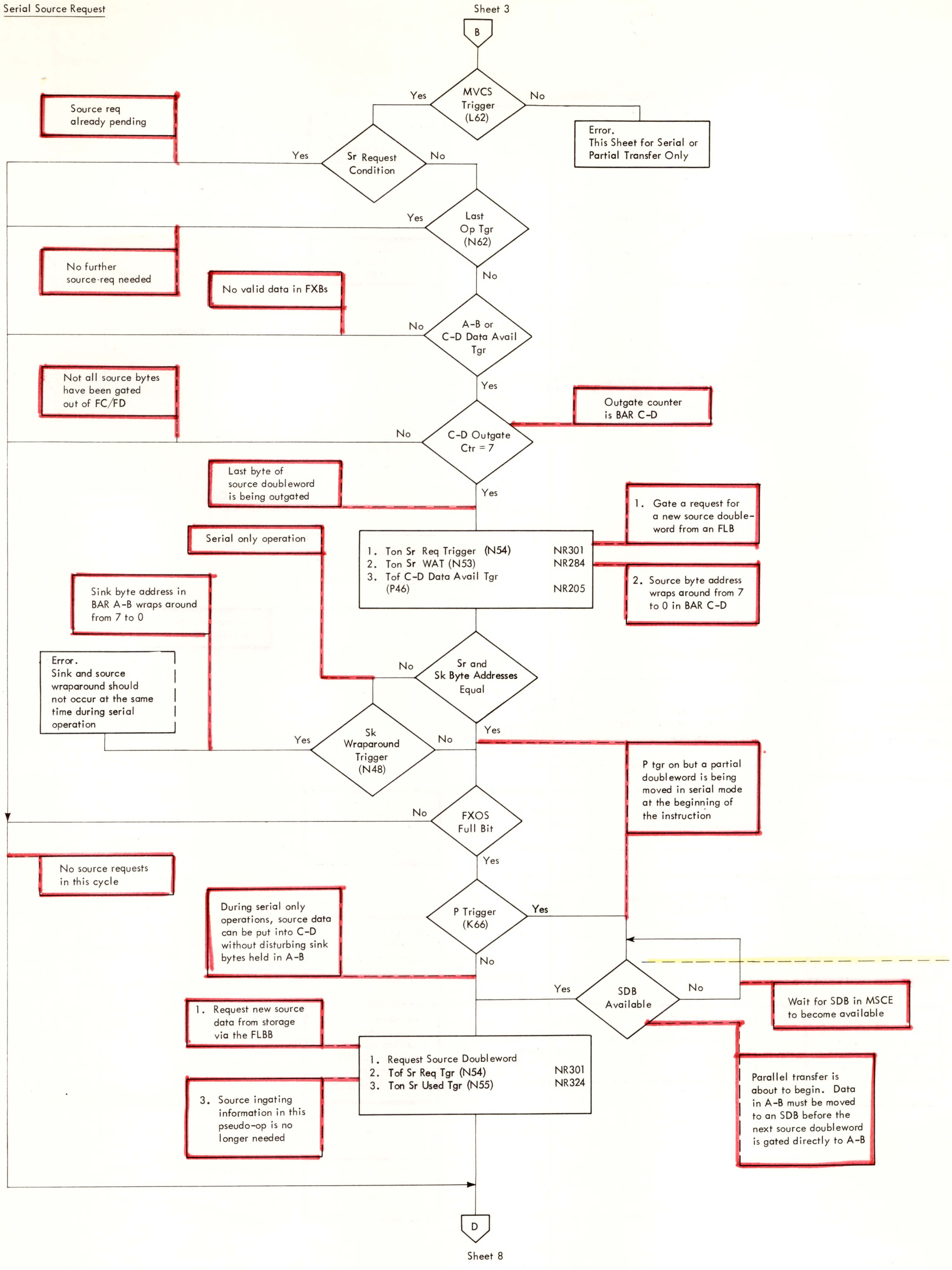

Sheet 3
B
MVCS Trigger (L62)
Yes
No
Error. This Sheet for Serial or Partial Transfer Only
Source req already pending
Sr Request Condition
Yes
No
Last Op Tgr (N62)
Yes
No
No further source-req needed
No valid data in FXBs
A-B or C-D Data Avail Tgr
No
Yes
Not all source bytes have been gated out of FC/FD
Outgate counter is BAR C-D
C-D Outgate Ctr = 7
No
Yes
Last byte of source doubleword is being outgated
1. Gate a request for a new source doubleword from an FLB
Serial only operation
1. Ton Sr Req Trigger (N54) NR301
2. Ton Sr WAT (N53) NR284
3. Tof C-D Data Avail Tgr (P46) NR205
2. Source byte address wraps around from 7 to 0 in BAR C-D
Sink byte address in BAR A-B wraps around from 7 to 0
Error. Sink and source wraparound should not occur at the same time during serial operation
Sr and Sk Byte Addresses Equal
No
Yes
Sk Wraparound Trigger (N48)
Yes
No
P tgr on but a partial doubleword is being moved in serial mode at the beginning of the instruction
FXOS Full Bit
No
Yes
No source requests in this cycle
During serial only operations, source data can be put into C-D without disturbing sink bytes held in A-B
P Trigger (K66)
Yes
No
SDB Available
Yes
No
Wait for SDB in MSCE to become available
1. Request new source data from storage via the FLBB
1. Request Source Doubleword
2. Tof Sr Req Tgr (N54) NR301
3. Ton Sr Used Tgr (N55) NR324
Parallel transfer is about to begin. Data in A-B must be moved to an SDB before the next source doubleword is gated directly to A-B
3. Source ingating information in this pseudo-op is no longer needed
D
Sheet 8

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 4 OF 8)

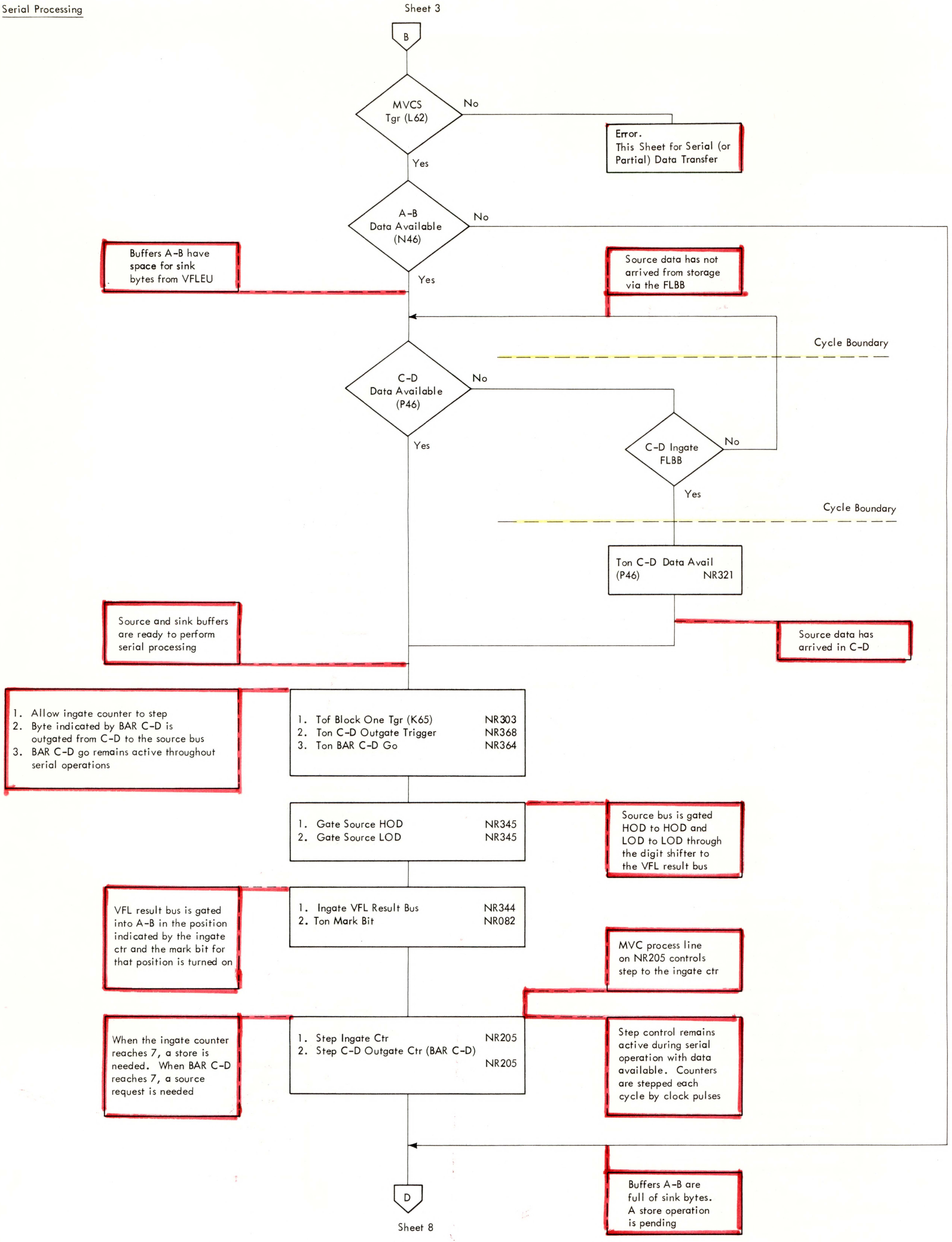

Sheet 3
B
MVCS Tgr (L62)
No
Error. This Sheet for Serial (or Partial) Data Transfer
Yes
A-B Data Available (N46)
No
Buffers A-B have space for sink bytes from VFLEU
Yes
Source data has not arrived from storage via the FLBB
Cycle Boundary
C-D Data Available (P46)
No
C-D Ingate FLBB
No
Yes
Yes
Cycle Boundary
Ton C-D Data Avail (P46) NR321
Source and sink buffers are ready to perform serial processing
Source data has arrived in C-D
1. Allow ingate counter to step
2. Byte indicated by BAR C-D is outgated from C-D to the source bus
3. BAR C-D go remains active throughout serial operations
1. Tof Block One Tgr (K65) NR303
2. Ton C-D Outgate Trigger NR368
3. Ton BAR C-D Go NR364
1. Gate Source HOD NR345
2. Gate Source LOD NR345
Source bus is gated HOD to HOD and LOD to LOD through the digit shifter to the VFL result bus
VFL result bus is gated into A-B in the position indicated by the ingate ctr and the mark bit for that position is turned on
1. Ingate VFL Result Bus NR344
2. Ton Mark Bit NR082
MVC process line on NR205 controls step to the ingate ctr
When the ingate counter reaches 7, a store is needed. When BAR C-D reaches 7, a source request is needed
1. Step Ingate Ctr NR205
2. Step C-D Outgate Ctr (BAR C-D) NR205
Step control remains active during serial operation with data available. Counters are stepped each cycle by clock pulses
D
Sheet 8
Buffers A-B are full of sink bytes. A store operation is pending

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 5 OF 8)

Serial Sink Store

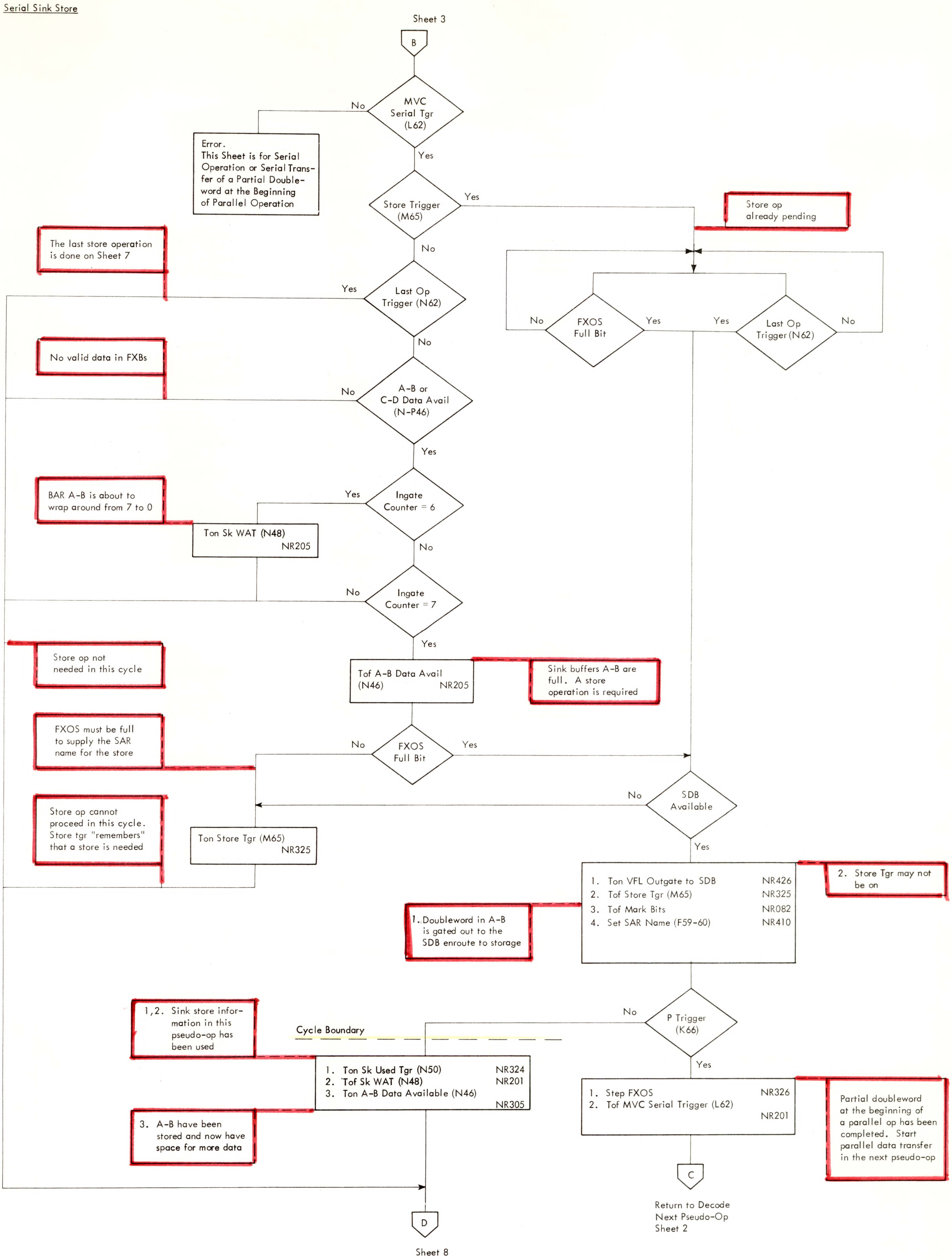

Sheet 3
B
MVC Serial Tgr (L62)
No
Yes
Error. This Sheet is for Serial Operation or Serial Transfer of a Partial Doubleword at the Beginning of Parallel Operation
Store Trigger (M65)
Yes
Store op already pending
No
The last store operation is done on Sheet 7
Yes
Last Op Trigger (N62)
No
FXOS Full Bit
Yes
Yes
Last Op Trigger (N62)
No
No
No valid data in FXBs
No
A-B or C-D Data Avail (N-P46)
Yes
BAR A-B is about to wrap around from 7 to 0
Yes
Ingate Counter = 6
Ton Sk WAT (N48) NR205
No
No
Ingate Counter = 7
Yes
Store op not needed in this cycle
Tof A-B Data Avail (N46) NR205
Sink buffers A-B are full. A store operation is required
FXOS must be full to supply the SAR name for the store
No
FXOS Full Bit
Yes
Store op cannot proceed in this cycle. Store tgr "remembers" that a store is needed
No
SDB Available
Ton Store Tgr (M65) NR325
Yes
1. Ton VFL Outgate to SDB NR426
2. Tof Store Tgr (M65) NR325
3. Tof Mark Bits NR082
4. Set SAR Name (F59-60) NR410
2. Store Tgr may not be on
1. Doubleword in A-B is gated out to the SDB enroute to storage
1,2. Sink store information in this pseudo-op has been used
Cycle Boundary
No
P Trigger (K66)
Yes
1. Ton Sk Used Tgr (N50) NR324
2. Tof Sk WAT (N48) NR201
3. Ton A-B Data Available (N46) NR305
1. Step FXOS NR326
2. Tof MVC Serial Trigger (L62) NR201
Partial doubleword at the beginning of a parallel op has been completed. Start parallel data transfer in the next pseudo-op
3. A-B have been stored and now have space for more data
C
Return to Decode Next Pseudo-Op Sheet 2
D
Sheet 8

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 6 OF 8)

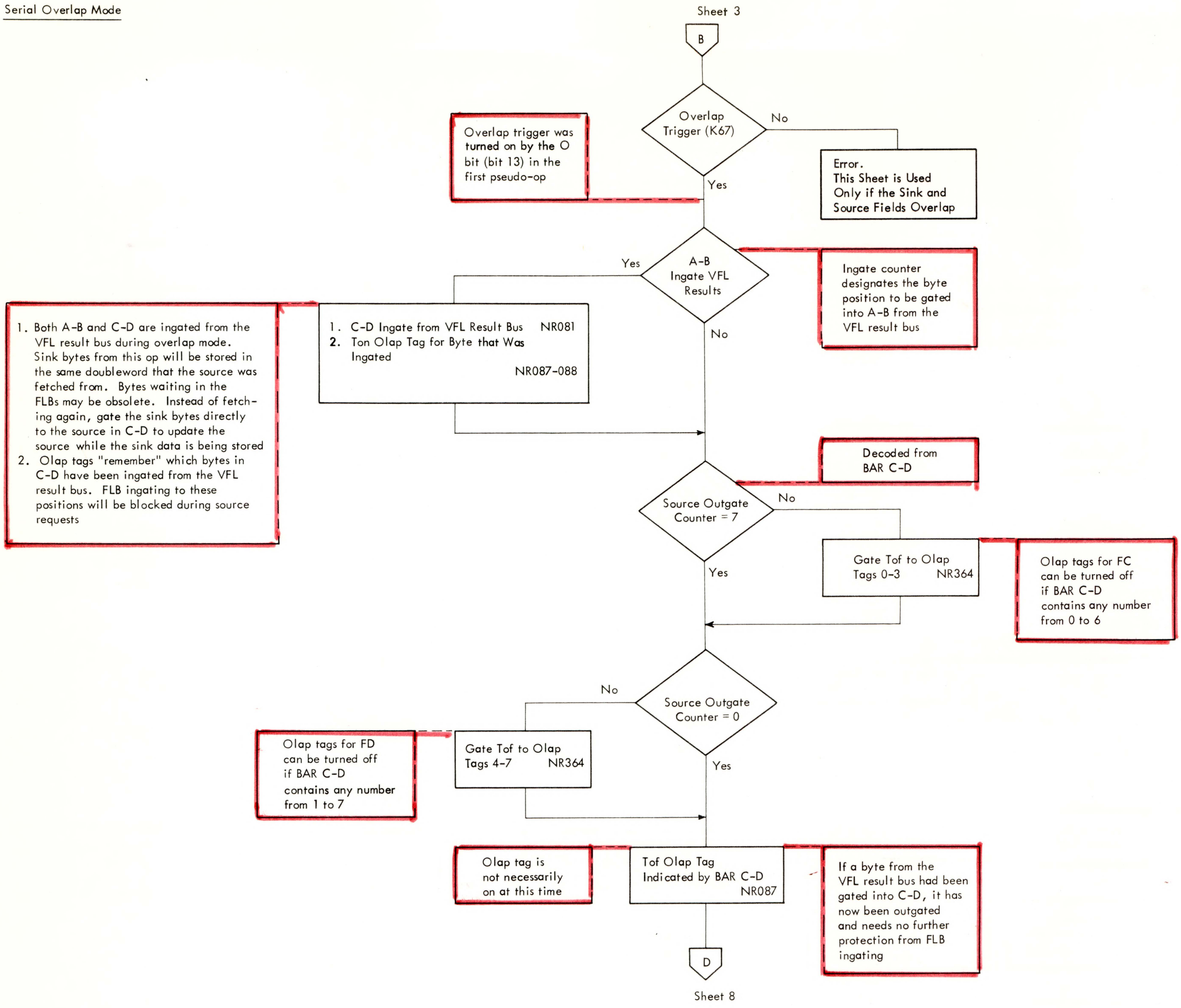

Sheet 3
B
Overlap Trigger (K67)
No
Yes
Overlap trigger was turned on by the O bit (bit 13) in the first pseudo-op
Error. This Sheet is Used Only if the Sink and Source Fields Overlap
A-B Ingate VFL Results
Yes
No
Ingate counter designates the byte position to be gated into A-B from the VFL result bus
1. C-D Ingate from VFL Result Bus NR081
2. Ton Olap Tag for Byte that Was Ingated NR087-088
1. Both A-B and C-D are ingated from the VFL result bus during overlap mode. Sink bytes from this op will be stored in the same doubleword that the source was fetched from. Bytes waiting in the FLBs may be obsolete. Instead of fetching again, gate the sink bytes directly to the source in C-D to update the source while the sink data is being stored
2. Olap tags "remember" which bytes in C-D have been ingated from the VFL result bus. FLB ingating to these positions will be blocked during source requests
Decoded from BAR C-D
Source Outgate Counter = 7
No
Yes
Gate Tof to Olap Tags 0-3 NR364
Olap tags for FC can be turned off if BAR C-D contains any number from 0 to 6
Source Outgate Counter = 0
No
Yes
Gate Tof to Olap Tags 4-7 NR364
Olap tags for FD can be turned off if BAR C-D contains any number from 1 to 7
Olap tag is not necessarily on at this time
Tof Olap Tag Indicated by BAR C-D NR087
If a byte from the VFL result bus had been gated into C-D, it has now been outgated and needs no further protection from FLB ingating
D
Sheet 8

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 7 OF 8)

Serial Ending Sequence

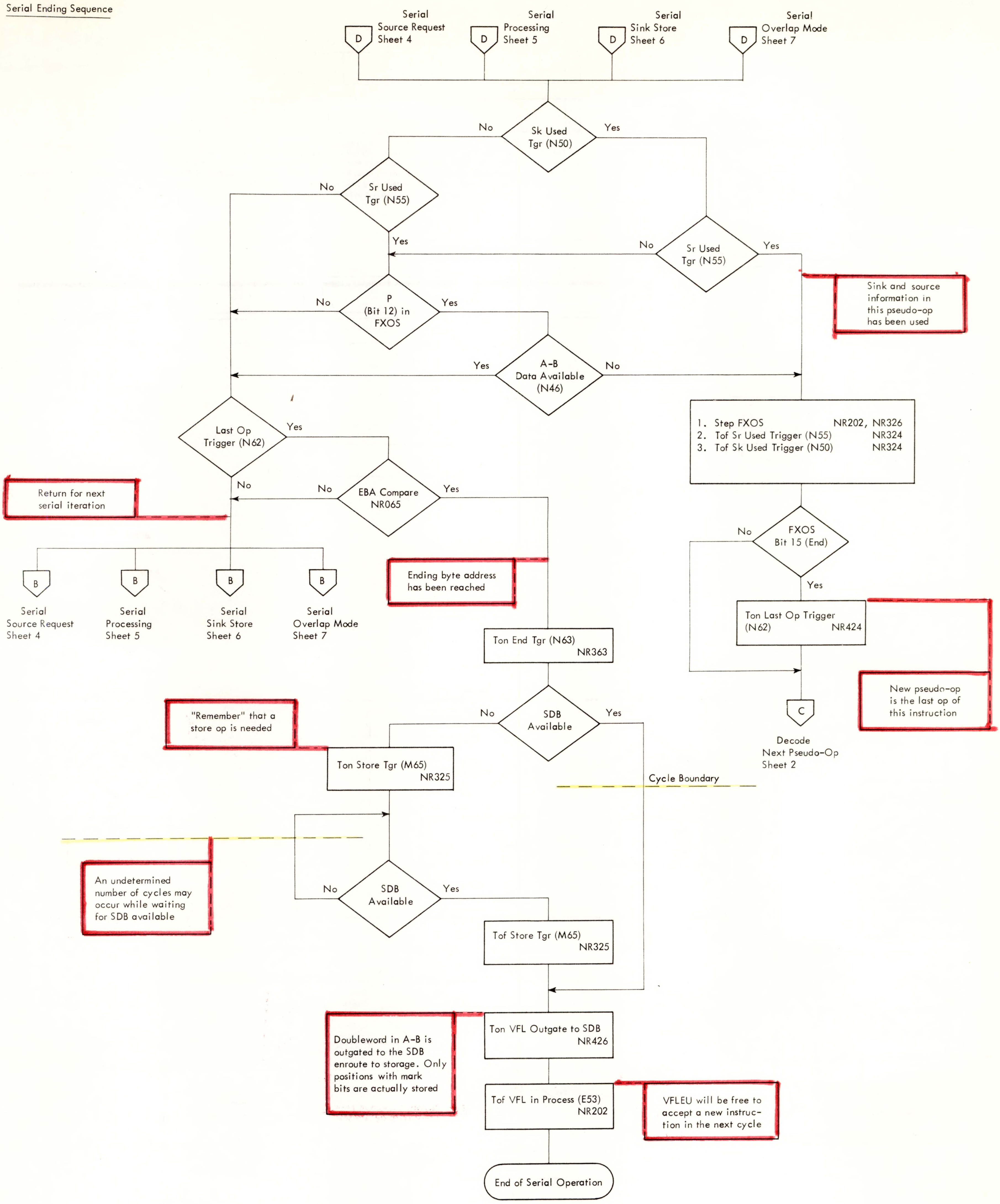

D Serial Source Request Sheet 4
D Serial Processing Sheet 5
D Serial Sink Store Sheet 6
D Serial Overlap Mode Sheet 7
Sk Used Tgr (N50)
No
Yes
Sr Used Tgr (N55)
No
Yes
Sr Used Tgr (N55)
No
Yes
Sink and source information in this pseudo-op has been used
P (Bit 12) in FXOS
No
Yes
A-B Data Available (N46)
Yes
No
1. Step FXOS NR202, NR326
2. Tof Sr Used Trigger (N55) NR324
3. Tof Sk Used Trigger (N50) NR324
Last Op Trigger (N62)
Yes
No
Return for next serial iteration
EBA Compare NR065
No
Yes
FXOS Bit 15 (End)
No
Yes
B Serial Source Request Sheet 4
B Serial Processing Sheet 5
B Serial Sink Store Sheet 6
B Serial Overlap Mode Sheet 7
Ending byte address has been reached
Ton Last Op Trigger (N62) NR424
Ton End Tgr (N63) NR363
New pseudo-op is the last op of this instruction
C Decode Next Pseudo-Op Sheet 2
"Remember" that a store op is needed
SDB Available
No
Yes
Ton Store Tgr (M65) NR325
Cycle Boundary
An undetermined number of cycles may occur while waiting for SDB available
SDB Available
No
Yes
Tof Store Tgr (M65) NR325
Doubleword in A-B is outgated to the SDB enroute to storage. Only positions with mark bits are actually stored
Ton VFL Outgate to SDB NR426
Tof VFL in Process (E53) NR202
VFLEU will be free to accept a new instruction in the next cycle
End of Serial Operation

DIAGRAM 5-122. MVC INSTRUCTION (SHEET 8 OF 8)

Data Flow

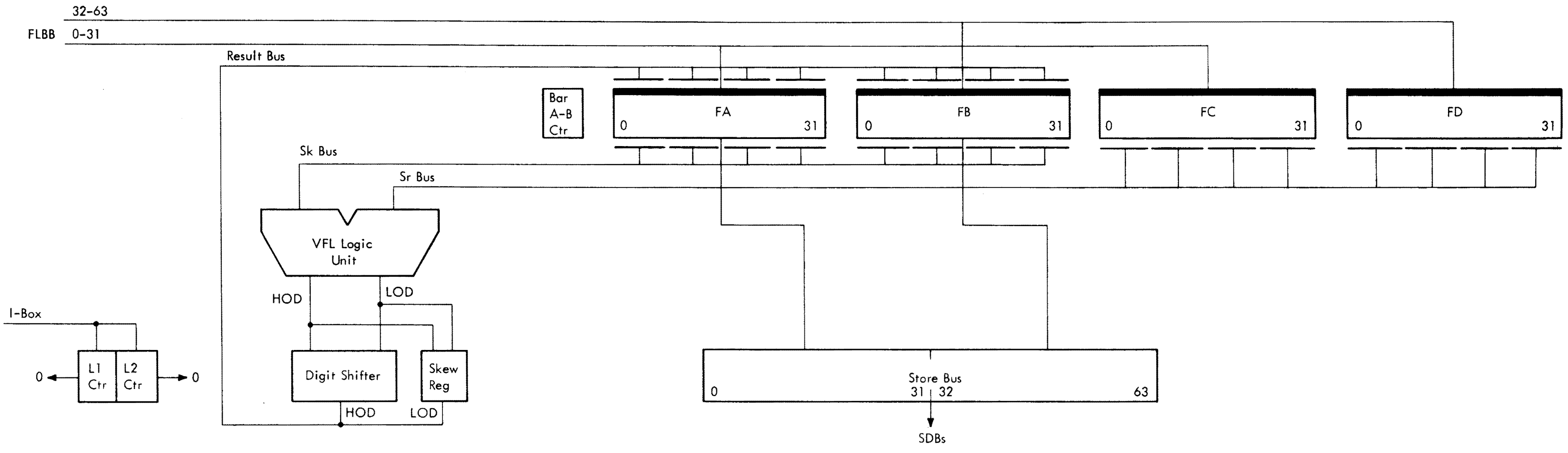
Objectives:
Move with Offset
MVO

0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20

* SS	F1	▨	FLB Name	P	O	S	E	FLB Name	SAR Name
** SS	F1	▨	FLB Name	P	O	S	E	▨	SAR Name

* First op only
** Any op after first op

1. Instruction offsets source word one digit position toward high-order-byte position.
2. First result byte: Low-order digit of sink byte is combined with high-order digit of source byte (digits retain same relative digit position).
3. Subsequent result bytes: Low-order digit of source byte is combined with high-order digit of succeeding source byte.
4. Bytes are processed until sink and source fields are used; however, if sink is used first, resultant sink word is stored and cancel mode is initiated.
If source word is used first, zeros are placed in remaining sink word byte positions and resultant sink word is stored.
5. First op specifies sink and source operand; subsequent ops specify source operands only.



Sheet Layout

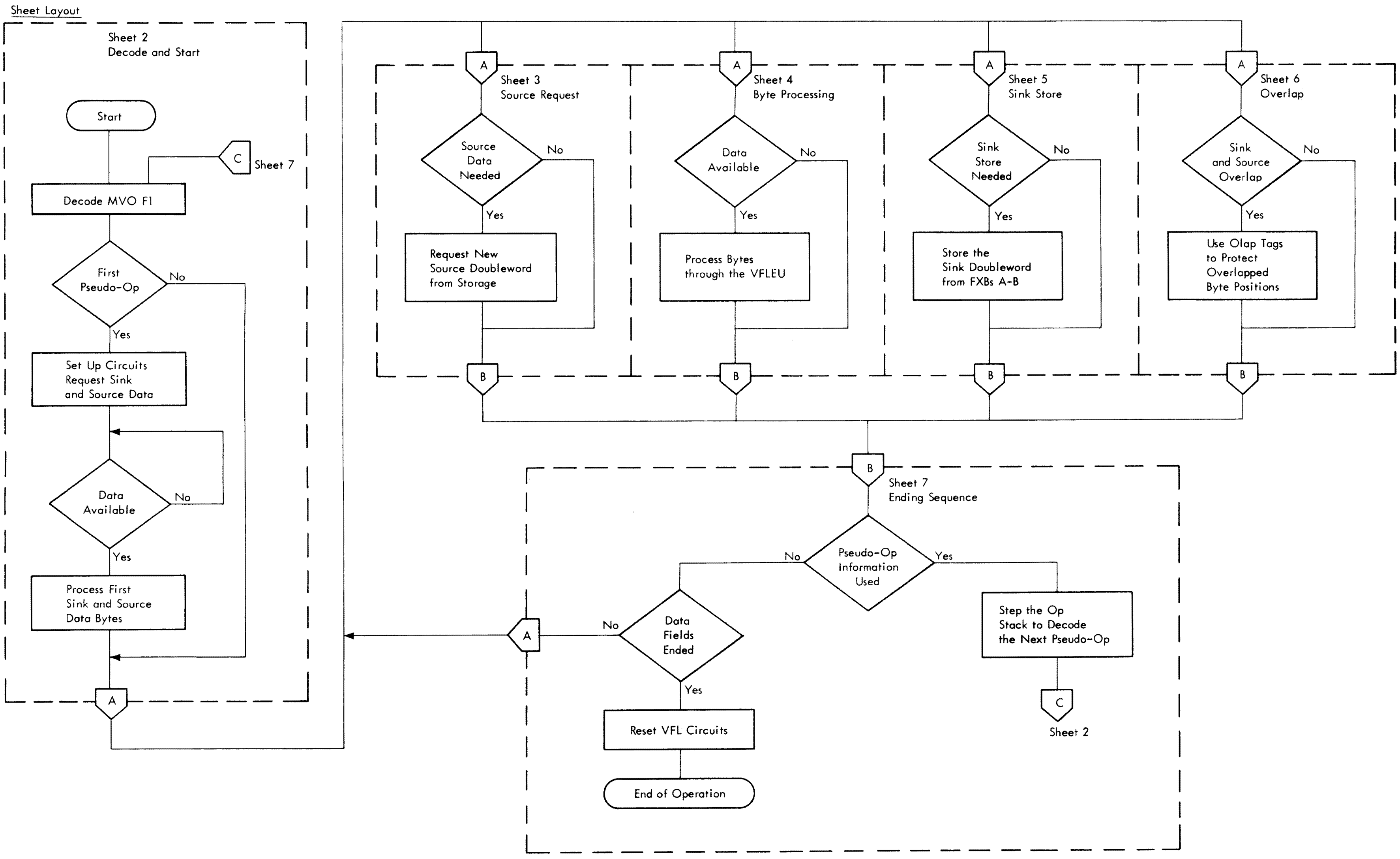


DIAGRAM 5-123. MVO INSTRUCTION (SHEET 1 OF 7)

Decode and Start

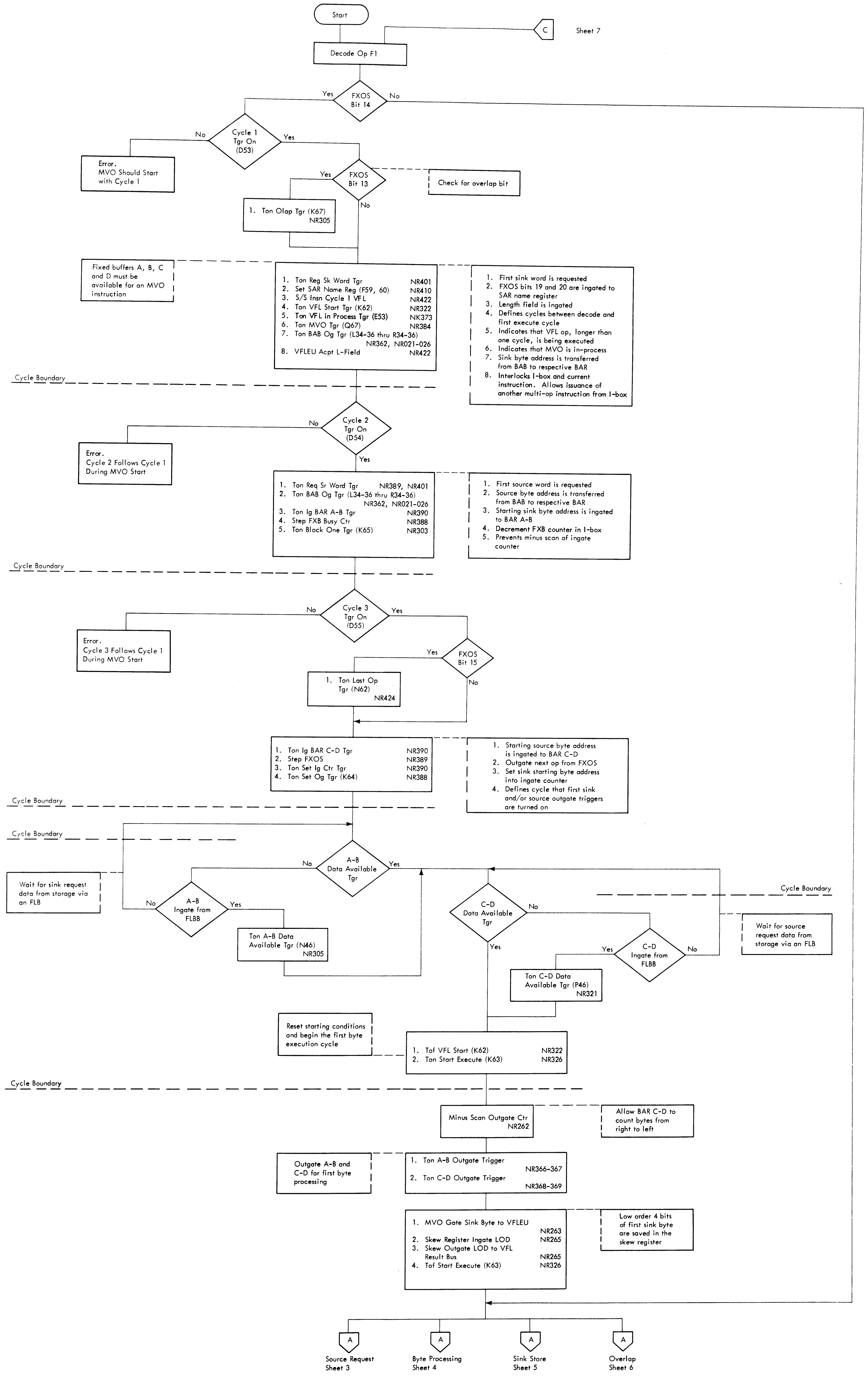

Start
C
Sheet 7
Decode Op F1
FXOS Bit 14
Yes
No
Cycle 1 Tgr On (D53)
No
Yes
Error. MVO Should Start with Cycle 1
FXOS Bit 13
Yes
No
Check for overlap bit
1. Ton Olap Tgr (K67) NR305
Fixed buffers A, B, C and D must be available for an MVO instruction
1. Ton Reg Sk Word Tgr NR401
2. Set SAR Name Reg (F59, 60) NR410
3. S/S Insn Cycle 1 VFL NR422
4. Ton VFL Start Tgr (K62) NR322
5. Ton VFL in Process Tgr (E53) NK373
6. Ton MVO Tgr (Q67) NR384
7. Ton BAB Og Tgr (L34-36 thru R34-36) NR362, NR021-026
8. VFLEU Acpt L-Field NR422
1. First sink word is requested
2. FXOS bits 19 and 20 are ingated to SAR name register
3. Length field is ingated
4. Defines cycles between decode and first execute cycle
5. Indicates that VFL op, longer than one cycle, is being executed
6. Indicates that MVO is in-process
7. Sink byte address is transferred from BAB to respective BAR
8. Interlocks I-box and current instruction. Allows issuance of another multi-op instruction from I-box
Cycle Boundary
Cycle 2 Tgr On (D54)
No
Yes
Error. Cycle 2 Follows Cycle 1 During MVO Start
1. Ton Req Sr Word Tgr NR389, NR401
2. Ton BAB Og Tgr (L34-36 thru R34-36) NR362, NR021-026
3. Ton Ig BAR A-B Tgr NR390
4. Step FXB Busy Ctr NR388
5. Ton Block One Tgr (K65) NR303
1. First source word is requested
2. Source byte address is transferred from BAB to respective BAR
3. Starting sink byte address is ingated to BAR A-B
4. Decrement FXB counter in I-box
5. Prevents minus scan of ingate counter
Cycle Boundary
Cycle 3 Tgr On (D55)
No
Yes
Error. Cycle 3 Follows Cycle 1 During MVO Start
FXOS Bit 15
Yes
No
1. Ton Last Op Tgr (N62) NR424
1. Ton Ig BAR C-D Tgr NR390
2. Step FXOS NR389
3. Ton Set Ig Ctr Tgr NR390
4. Ton Set Og Tgr (K64) NR388
1. Starting source byte address is ingated to BAR C-D
2. Outgate next op from FXOS
3. Set sink starting byte address into ingate counter
4. Defines cycle that first sink and/or source outgate triggers are turned on
Cycle Boundary
Cycle Boundary
A-B Data Available Tgr
No
Yes
Wait for sink request data from storage via an FLB
A-B Ingate from FLBB
No
Yes
Ton A-B Data Available Tgr (N46) NR305
C-D Data Available Tgr
No
Yes
Cycle Boundary
C-D Ingate from FLBB
Yes
No
Wait for source request data from storage via an FLB
Ton C-D Data Available Tgr (P46) NR321
Reset starting conditions and begin the first byte execution cycle
1. Tof VFL Start (K62) NR322
2. Ton Start Execute (K63) NR326
Cycle Boundary
Minus Scan Outgate Ctr NR262
Allow BAR C-D to count bytes from right to left
Outgate A-B and C-D for first byte processing
1. Ton A-B Outgate Trigger NR366-367
2. Ton C-D Outgate Trigger NR368-369
1. MVO Gate Sink Byte to VFLEU NR263
2. Skew Register Ingate LOD NR265
3. Skew Outgate LOD to VFL Result Bus NR265
4. Tof Start Execute (K63) NR326
Low order 4 bits of first sink byte are saved in the skew register
A
Source Request Sheet 3
A
Byte Processing Sheet 4
A
Sink Store Sheet 5
A
Overlap Sheet 6

DIAGRAM 5-123. MVO INSTRUCTION (SHEET 2 OF 7)

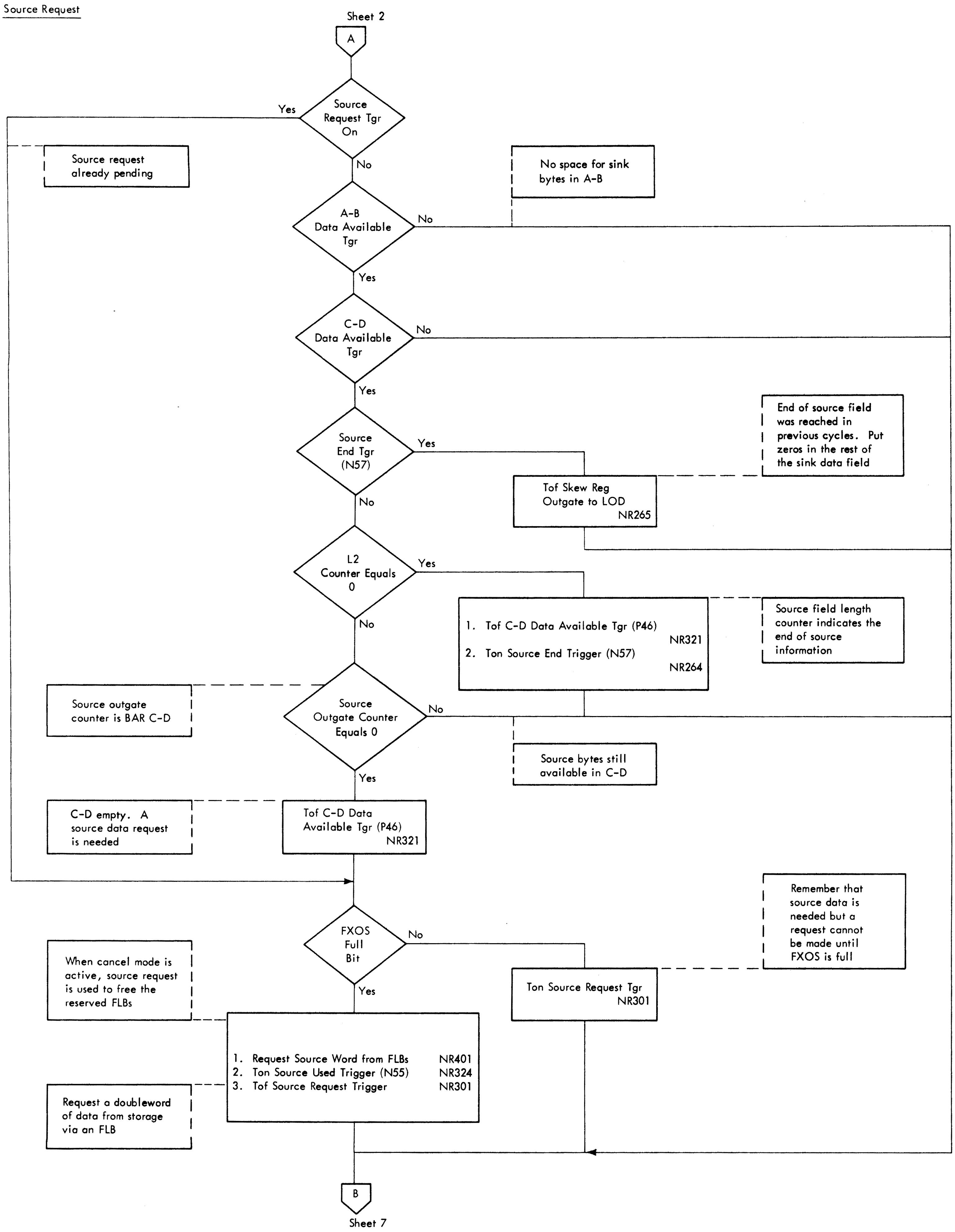

Source Request
Sheet 2
A
Source Request Tgr On
Yes
Source request already pending
No
No space for sink bytes in A-B
A-B Data Available Tgr
No
Yes
C-D Data Available Tgr
No
Yes
End of source field was reached in previous cycles. Put zeros in the rest of the sink data field
Source End Tgr (N57)
Yes
Tof Skew Reg Outgate to LOD NR265
No
L2 Counter Equals 0
Yes
1. Tof C-D Data Available Tgr (P46) NR321
2. Ton Source End Trigger (N57) NR264
Source field length counter indicates the end of source information
No
Source outgate counter is BAR C-D
Source Outgate Counter Equals 0
No
Source bytes still available in C-D
Yes
C-D empty. A source data request is needed
Tof C-D Data Available Tgr (P46) NR321
Remember that source data is needed but a request cannot be made until FXOS is full
FXOS Full Bit
No
When cancel mode is active, source request is used to free the reserved FLBs
Yes
Ton Source Request Tgr NR301
1. Request Source Word from FLBs NR401
2. Ton Source Used Trigger (N55) NR324
3. Tof Source Request Trigger NR301
Request a doubleword of data from storage via an FLB
B
Sheet 7

DIAGRAM 5-123. MVO INSTRUCTION (SHEET 3 OF 7)

Byte Processing

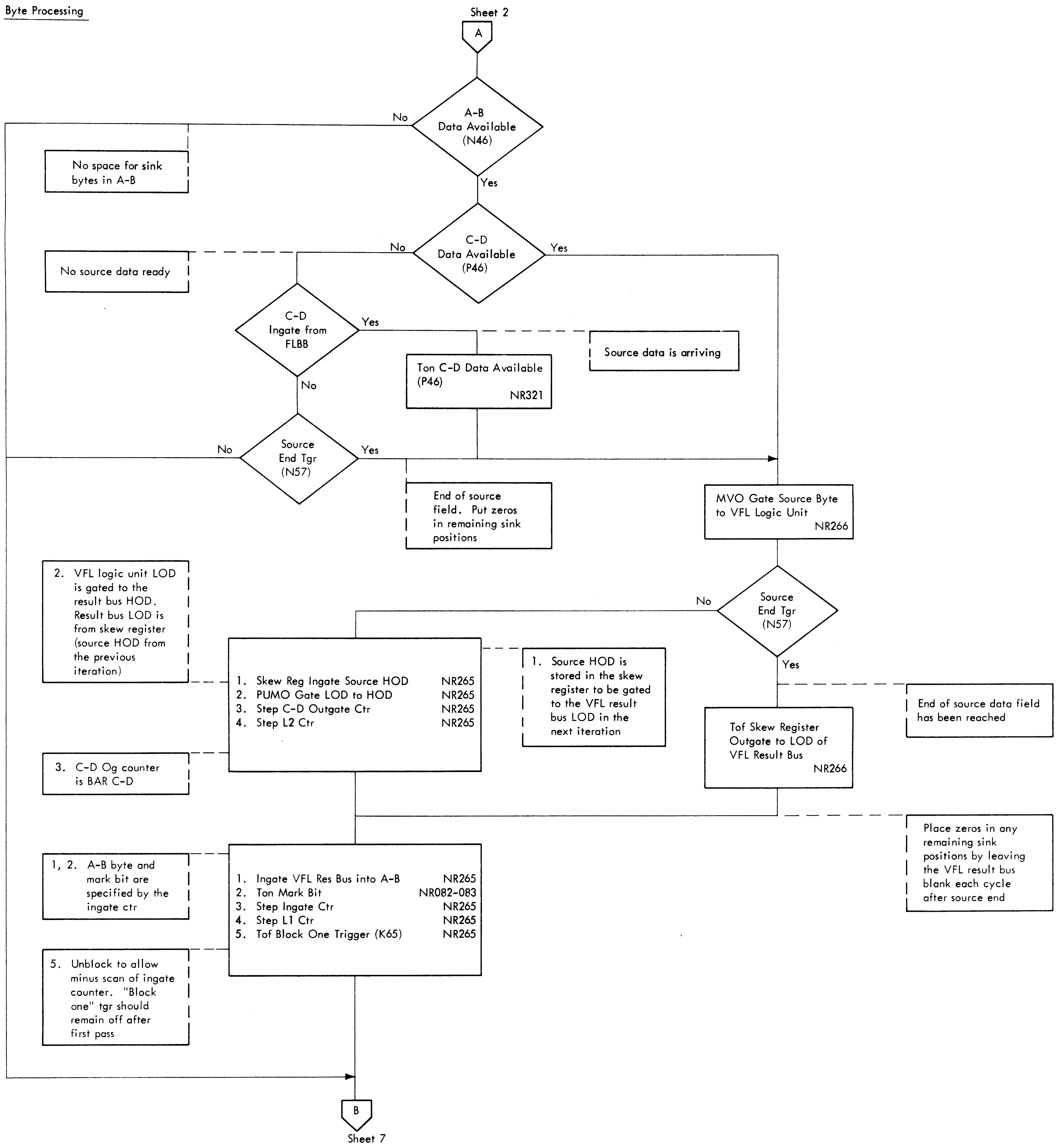

Sheet 2
A
A-B Data Available (N46)
No
Yes
No space for sink bytes in A-B
C-D Data Available (P46)
No
Yes
No source data ready
C-D Ingate from FLBB
Yes
No
Source data is arriving
Ton C-D Data Available (P46) NR321
Source End Tgr (N57)
No
Yes
End of source field. Put zeros in remaining sink positions
MVO Gate Source Byte to VFL Logic Unit NR266
2. VFL logic unit LOD is gated to the result bus HOD. Result bus LOD is from skew register (source HOD from the previous iteration)
Source End Tgr (N57)
No
Yes
1. Skew Reg Ingate Source HOD NR265
2. PUMO Gate LOD to HOD NR265
3. Step C-D Outgate Ctr NR265
4. Step L2 Ctr NR265
1. Source HOD is stored in the skew register to be gated to the VFL result bus LOD in the next iteration
End of source data field has been reached
Tof Skew Register Outgate to LOD of VFL Result Bus NR266
3. C-D Og counter is BAR C-D
Place zeros in any remaining sink positions by leaving the VFL result bus blank each cycle after source end
1, 2. A-B byte and mark bit are specified by the ingate ctr
1. Ingate VFL Res Bus into A-B NR265
2. Ton Mark Bit NR082-083
3. Step Ingate Ctr NR265
4. Step L1 Ctr NR265
5. Tof Block One Trigger (K65) NR265
5. Unblock to allow minus scan of ingate counter. "Block one" tgr should remain off after first pass
B
Sheet 7

DIAGRAM 5-123. MVO INSTRUCTION (SHEET 4 OF 7)

Sink Store

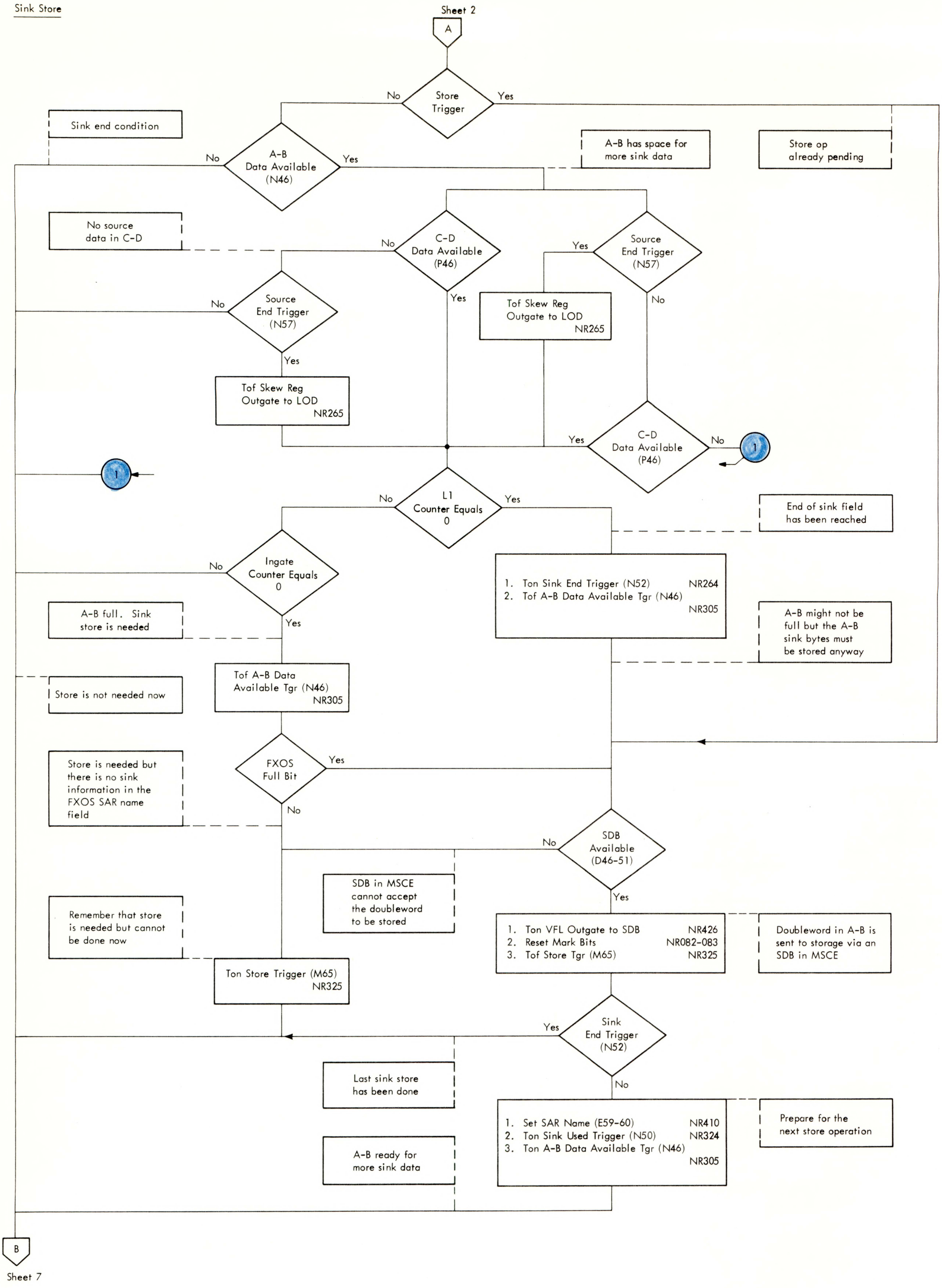

Sheet 2
A
Store Trigger
No
Yes
Sink end condition
A-B Data Available (N46)
No
Yes
A-B has space for more sink data
Store op already pending
No source data in C-D
C-D Data Available (P46)
No
Yes
Source End Trigger (N57)
Yes
No
Source End Trigger (N57)
No
Yes
Tof Skew Reg Outgate to LOD NR265
Tof Skew Reg Outgate to LOD NR265
C-D Data Available (P46)
Yes
No
1
1
L1 Counter Equals 0
No
Yes
End of sink field has been reached
Ingate Counter Equals 0
No
Yes
1. Ton Sink End Trigger (N52) NR264
2. Tof A-B Data Available Tgr (N46) NR305
A-B full. Sink store is needed
A-B might not be full but the A-B sink bytes must be stored anyway
Store is not needed now
Tof A-B Data Available Tgr (N46) NR305
Store is needed but there is no sink information in the FXOS SAR name field
FXOS Full Bit
Yes
No
SDB Available (D46-51)
No
Yes
SDB in MSCE cannot accept the doubleword to be stored
Remember that store is needed but cannot be done now
1. Ton VFL Outgate to SDB NR426
2. Reset Mark Bits NR082-083
3. Tof Store Tgr (M65) NR325
Doubleword in A-B is sent to storage via an SDB in MSCE
Ton Store Trigger (M65) NR325
Sink End Trigger (N52)
Yes
No
Last sink store has been done
1. Set SAR Name (E59-60) NR410
2. Ton Sink Used Trigger (N50) NR324
3. Ton A-B Data Available Tgr (N46) NR305
Prepare for the next store operation
A-B ready for more sink data
B
Sheet 7

DIAGRAM 5-123. MVO INSTRUCTION (SHEET 5 OF 7)

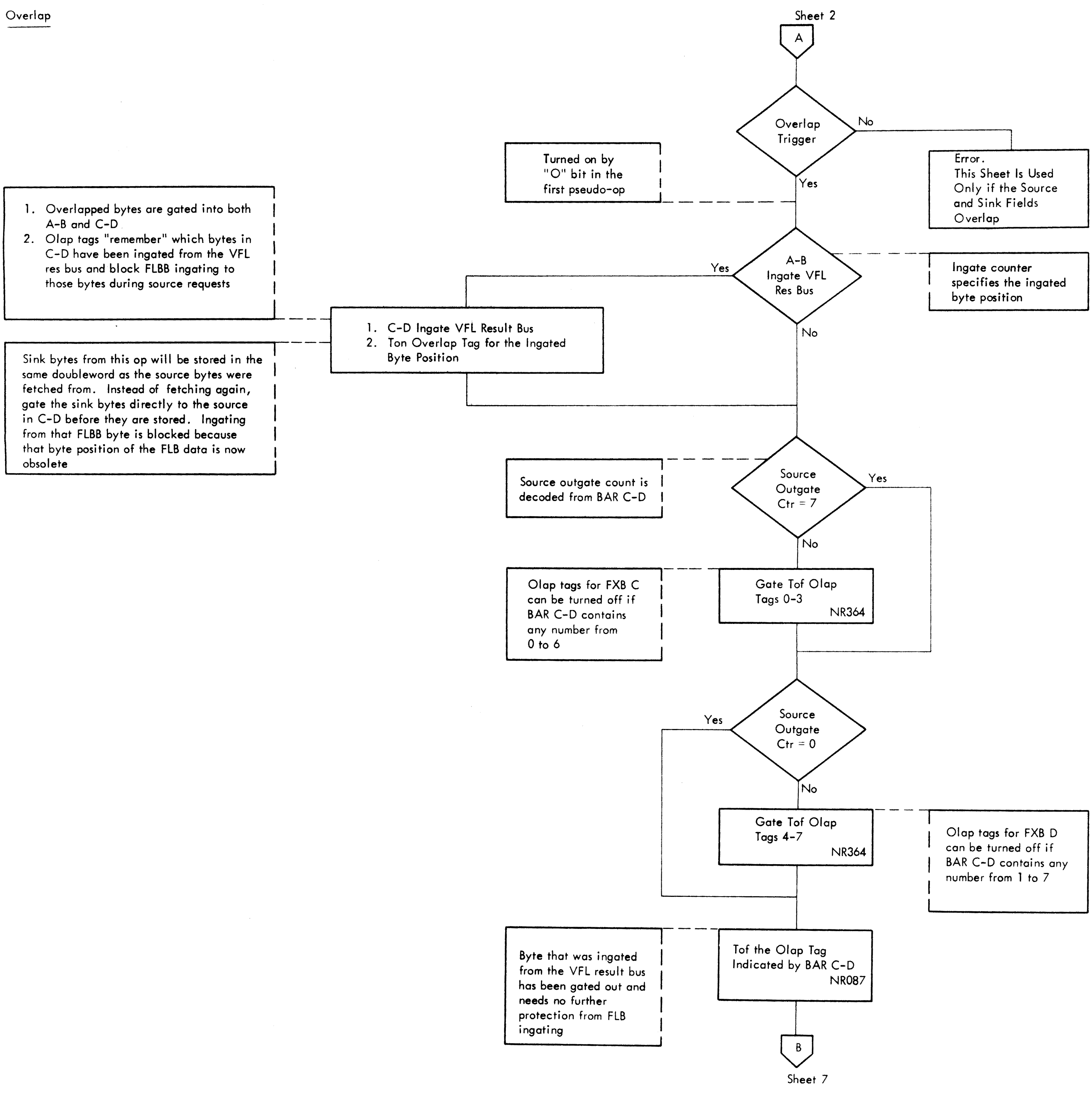

Overlap
Sheet 2
A
Overlap Trigger
No
Yes
Turned on by "O" bit in the first pseudo-op
Error. This Sheet Is Used Only if the Source and Sink Fields Overlap
1. Overlapped bytes are gated into both A-B and C-D
2. Olap tags "remember" which bytes in C-D have been ingated from the VFL res bus and block FLBB ingating to those bytes during source requests
A-B Ingate VFL Res Bus
Yes
No
Ingate counter specifies the ingated byte position
1. C-D Ingate VFL Result Bus
2. Ton Overlap Tag for the Ingated Byte Position
Sink bytes from this op will be stored in the same doubleword as the source bytes were fetched from. Instead of fetching again, gate the sink bytes directly to the source in C-D before they are stored. Ingating from that FLBB byte is blocked because that byte position of the FLB data is now obsolete
Source outgate count is decoded from BAR C-D
Source Outgate Ctr = 7
Yes
No
Olap tags for FXB C can be turned off if BAR C-D contains any number from 0 to 6
Gate Tof Olap Tags 0-3
NR364
Source Outgate Ctr = 0
Yes
No
Gate Tof Olap Tags 4-7
NR364
Olap tags for FXB D can be turned off if BAR C-D contains any number from 1 to 7
Byte that was ingated from the VFL result bus has been gated out and needs no further protection from FLB ingating
Tof the Olap Tag Indicated by BAR C-D
NR087
B
Sheet 7

DIAGRAM 5-123. MVO INSTRUCTION (SHEET 6 OF 7)

Ending Sequence

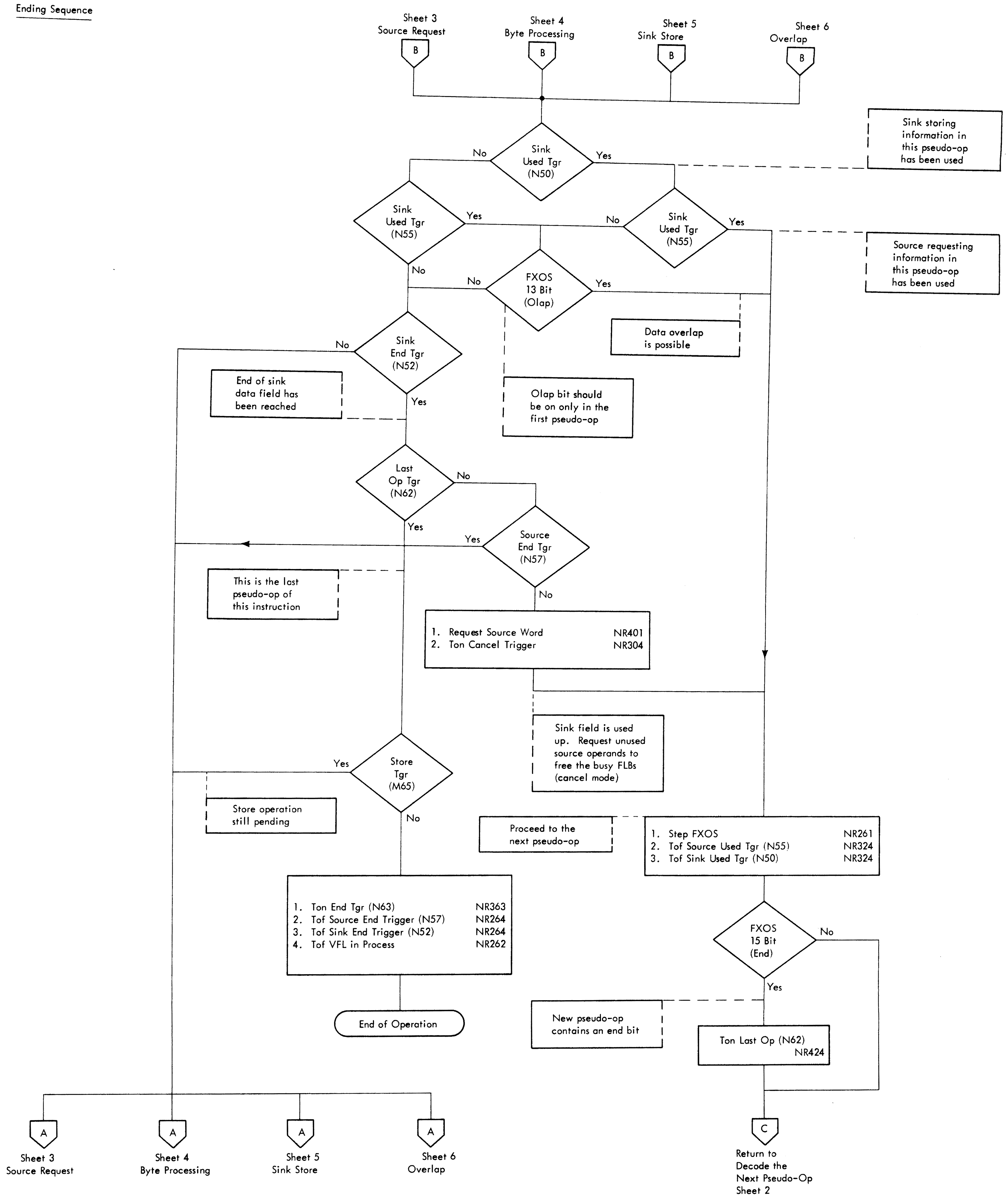

Sheet 3
Source Request
B
Sheet 4
Byte Processing
B
Sheet 5
Sink Store
B
Sheet 6
Overlap
B
Sink Used Tgr (N50)
No
Yes
Sink storing information in this pseudo-op has been used
Sink Used Tgr (N55)
Yes
No
Sink Used Tgr (N55)
Yes
Source requesting information in this pseudo-op has been used
No
FXOS 13 Bit (Olap)
No
Yes
Data overlap is possible
Sink End Tgr (N52)
No
Yes
End of sink data field has been reached
Olap bit should be on only in the first pseudo-op
Last Op Tgr (N62)
No
Yes
Source End Tgr (N57)
Yes
No
This is the last pseudo-op of this instruction
1. Request Source Word NR401
2. Ton Cancel Trigger NR304
Sink field is used up. Request unused source operands to free the busy FLBs (cancel mode)
Store Tgr (M65)
Yes
No
Store operation still pending
Proceed to the next pseudo-op
1. Step FXOS NR261
2. Tof Source Used Tgr (N55) NR324
3. Tof Sink Used Tgr (N50) NR324
1. Ton End Tgr (N63) NR363
2. Tof Source End Trigger (N57) NR264
3. Tof Sink End Trigger (N52) NR264
4. Tof VFL in Process NR262
FXOS 15 Bit (End)
No
Yes
New pseudo-op contains an end bit
Ton Last Op (N62) NR424
End of Operation
A
Sheet 3
Source Request
A
Sheet 4
Byte Processing
A
Sheet 5
Sink Store
A
Sheet 6
Overlap
C
Return to Decode the Next Pseudo-Op Sheet 2

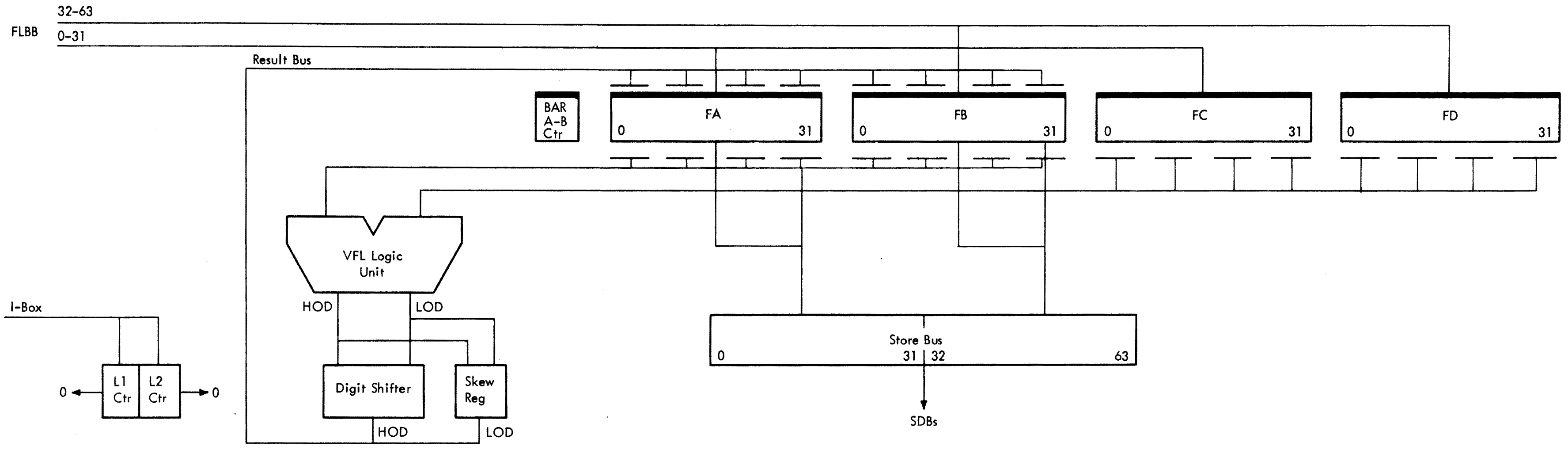
DIAGRAM 5-123. MVO INSTRUCTION (SHEET 7 OF 7)

Objectives:
Pack and Unpack
PACK, UNPK

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
SS	F2, F3								▨	FLB Name			P	O	S	E	Starting Byte Addr			SAR Name	

1. Source words are transferred from a specified FLB (FLB name) to source buffer (FC/FD).
2. Bytes are processed one at a time. During overlap processing result bytes are placed in sink and source buffers.
3. Processing is as follows:
Pack - The LOD from two adjacent bytes are combined into a single, all digit result byte.
HOD from these bytes (zone bits) are dropped.
Unpack - The HOD and LOD of a byte are separated and placed in the LOD positions of two result bytes.
The HOD of the result bytes (zone bits) are supplied by circuits within the VFLEU.
4. Processing continues until both fields (sink and source) are used.

Data Flow



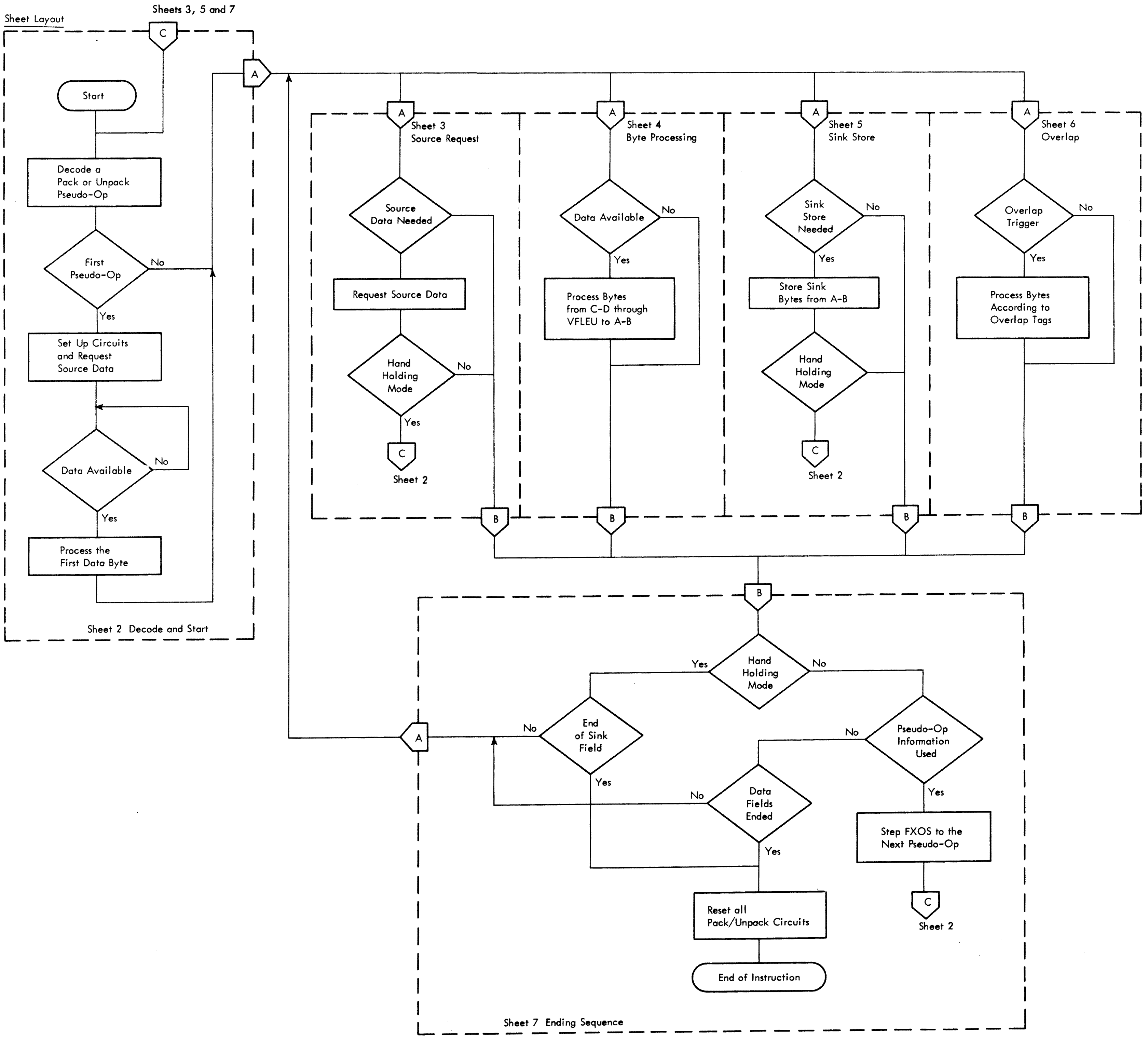


DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 1 OF 7)

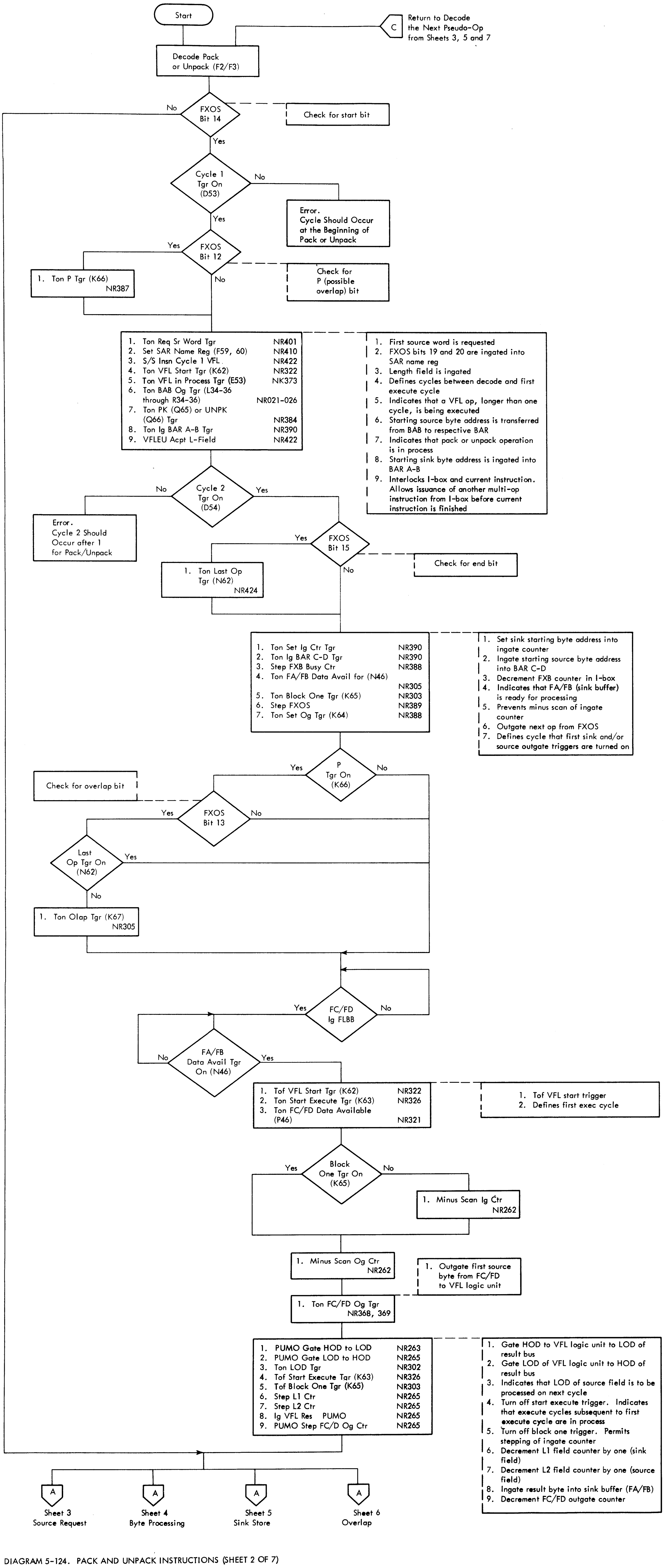

Start
Return to Decode the Next Pseudo-Op from Sheets 3, 5 and 7
C
Decode Pack or Unpack (F2/F3)
FXOS Bit 14
No
Yes
Check for start bit
Cycle 1 Tgr On (D53)
No
Error. Cycle Should Occur at the Beginning of Pack or Unpack
Yes
FXOS Bit 12
Yes
No
1. Ton P Tgr (K66) NR387
Check for P (possible overlap) bit
1. Ton Req Sr Word Tgr NR401
2. Set SAR Name Reg (F59, 60) NR410
3. S/S Insn Cycle 1 VFL NR422
4. Ton VFL Start Tgr (K62) NR322
5. Ton VFL in Process Tgr (E53) NK373
6. Ton BAB Og Tgr (L34-36 through R34-36) NR021-026
7. Ton PK (Q65) or UNPK (Q66) Tgr NR384
8. Ton Ig BAR A-B Tgr NR390
9. VFLEU Acpt L-Field NR422
1. First source word is requested
2. FXOS bits 19 and 20 are ingated into SAR name reg
3. Length field is ingated
4. Defines cycles between decode and first execute cycle
5. Indicates that a VFL op, longer than one cycle, is being executed
6. Starting source byte address is transferred from BAB to respective BAR
7. Indicates that pack or unpack operation is in process
8. Starting sink byte address is ingated into BAR A-B
9. Interlocks I-box and current instruction. Allows issuance of another multi-op instruction from I-box before current instruction is finished
No
Cycle 2 Tgr On (D54)
Yes
Error. Cycle 2 Should Occur after 1 for Pack/Unpack
Yes
FXOS Bit 15
No
Check for end bit
1. Ton Last Op Tgr (N62) NR424
1. Ton Set Ig Ctr Tgr NR390
2. Ton Ig BAR C-D Tgr NR390
3. Step FXB Busy Ctr NR388
4. Ton FA/FB Data Avail for (N46) NR305
5. Ton Block One Tgr (K65) NR303
6. Step FXOS NR389
7. Ton Set Og Tgr (K64) NR388
1. Set sink starting byte address into ingate counter
2. Ingate starting source byte address into BAR C-D
3. Decrement FXB counter in I-box
4. Indicates that FA/FB (sink buffer) is ready for processing
5. Prevents minus scan of ingate counter
6. Outgate next op from FXOS
7. Defines cycle that first sink and/or source outgate triggers are turned on
Yes
P Tgr On (K66)
No
Check for overlap bit
Yes
FXOS Bit 13
No
Last Op Tgr On (N62)
Yes
No
1. Ton Olap Tgr (K67) NR305
Yes
FC/FD Ig FLBB
No
FA/FB Data Avail Tgr On (N46)
No
Yes
1. Tof VFL Start Tgr (K62) NR322
2. Ton Start Execute Tgr (K63) NR326
3. Ton FC/FD Data Available (P46) NR321
1. Tof VFL start trigger
2. Defines first exec cycle
Yes
Block One Tgr On (K65)
No
1. Minus Scan Ig Ctr NR262
1. Minus Scan Og Ctr NR262
1. Outgate first source byte from FC/FD to VFL logic unit
1. Ton FC/FD Og Tgr NR368, 369
1. PUMO Gate HOD to LOD NR263
2. PUMO Gate LOD to HOD NR265
3. Ton LOD Tgr NR302
4. Tof Start Execute Tgr (K63) NR326
5. Tof Block One Tgr (K65) NR303
6. Step L1 Ctr NR265
7. Step L2 Ctr NR265
8. Ig VFL Res PUMO NR265
9. PUMO Step FC/D Og Ctr NR265
1. Gate HOD to VFL logic unit to LOD of result bus
2. Gate LOD of VFL logic unit to HOD of result bus
3. Indicates that LOD of source field is to be processed on next cycle
4. Turn off start execute trigger. Indicates that execute cycles subsequent to first execute cycle are in process
5. Turn off block one trigger. Permits stepping of ingate counter
6. Decrement L1 field counter by one (sink field)
7. Decrement L2 field counter by one (source field)
8. Ingate result byte into sink buffer (FA/FB)
9. Decrement FC/FD outgate counter
A
Sheet 3 Source Request
A
Sheet 4 Byte Processing
A
Sheet 5 Sink Store
A
Sheet 6 Overlap

DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 2 OF 7)

Source Request

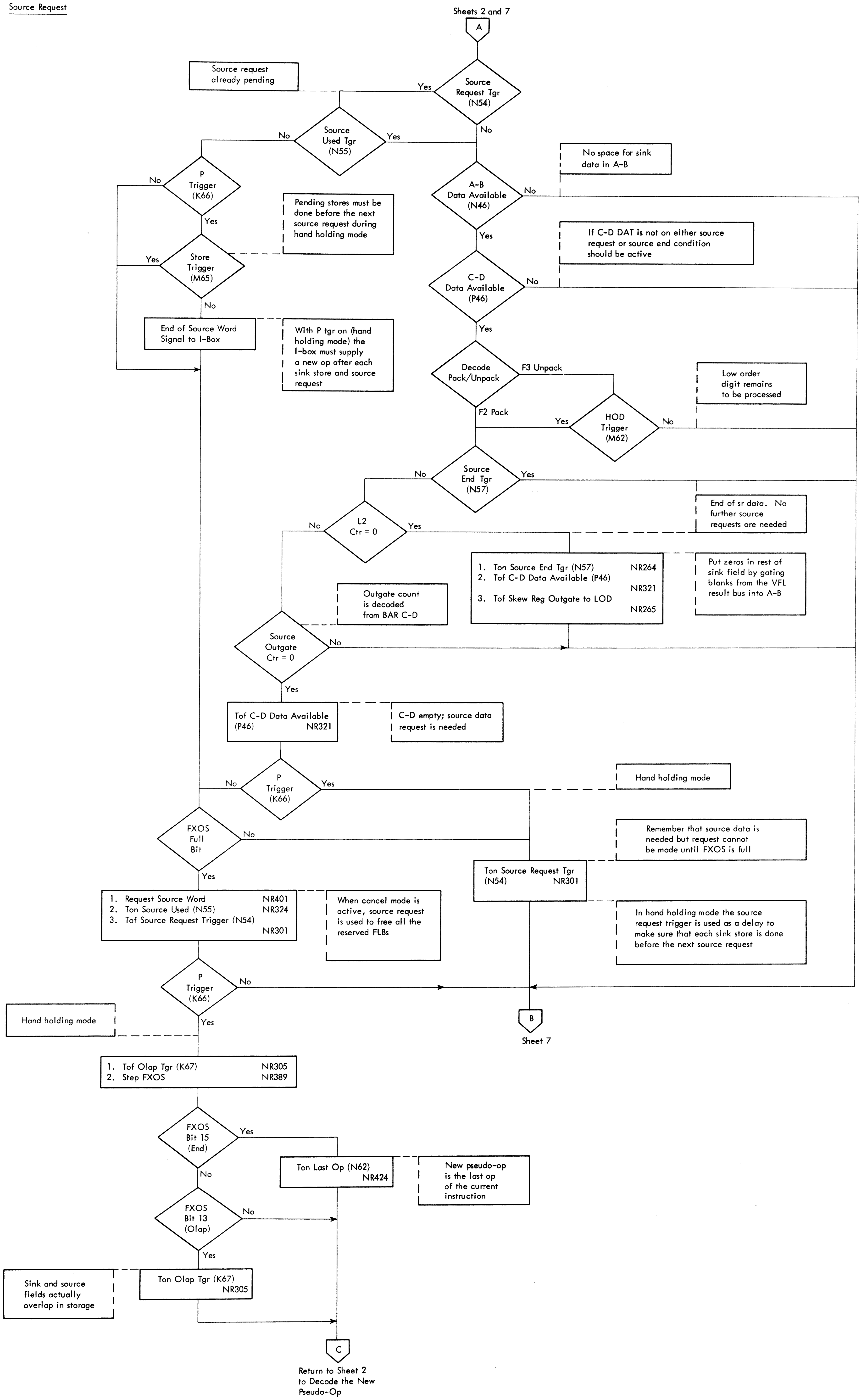

Sheets 2 and 7
A
Source request already pending
Yes
Source Request Tgr (N54)
No
Source Used Tgr (N55)
No
Yes
No space for sink data in A-B
P Trigger (K66)
No
Yes
A-B Data Available (N46)
No
Yes
Pending stores must be done before the next source request during hand holding mode
If C-D DAT is not on either source request or source end condition should be active
Store Trigger (M65)
Yes
No
C-D Data Available (P46)
No
Yes
End of Source Word Signal to I-Box
With P tgr on (hand holding mode) the I-box must supply a new op after each sink store and source request
Decode Pack/Unpack
F3 Unpack
F2 Pack
Low order digit remains to be processed
HOD Trigger (M62)
Yes
No
Source End Tgr (N57)
No
Yes
End of sr data. No further source requests are needed
L2 Ctr = 0
No
Yes
1. Ton Source End Tgr (N57) NR264
2. Tof C-D Data Available (P46) NR321
3. Tof Skew Reg Outgate to LOD NR265
Put zeros in rest of sink field by gating blanks from the VFL result bus into A-B
Outgate count is decoded from BAR C-D
Source Outgate Ctr = 0
No
Yes
Tof C-D Data Available (P46) NR321
C-D empty; source data request is needed
P Trigger (K66)
No
Yes
Hand holding mode
FXOS Full Bit
No
Yes
Remember that source data is needed but request cannot be made until FXOS is full
Ton Source Request Tgr (N54) NR301
1. Request Source Word NR401
2. Ton Source Used (N55) NR324
3. Tof Source Request Trigger (N54) NR301
When cancel mode is active, source request is used to free all the reserved FLBs
In hand holding mode the source request trigger is used as a delay to make sure that each sink store is done before the next source request
P Trigger (K66)
No
Yes
Hand holding mode
B
Sheet 7
1. Tof Olap Tgr (K67) NR305
2. Step FXOS NR389
FXOS Bit 15 (End)
Yes
No
Ton Last Op (N62) NR424
New pseudo-op is the last op of the current instruction
FXOS Bit 13 (Olap)
No
Yes
Sink and source fields actually overlap in storage
Ton Olap Tgr (K67) NR305
C
Return to Sheet 2 to Decode the New Pseudo-Op

DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 3 OF 7)

Byte Processing

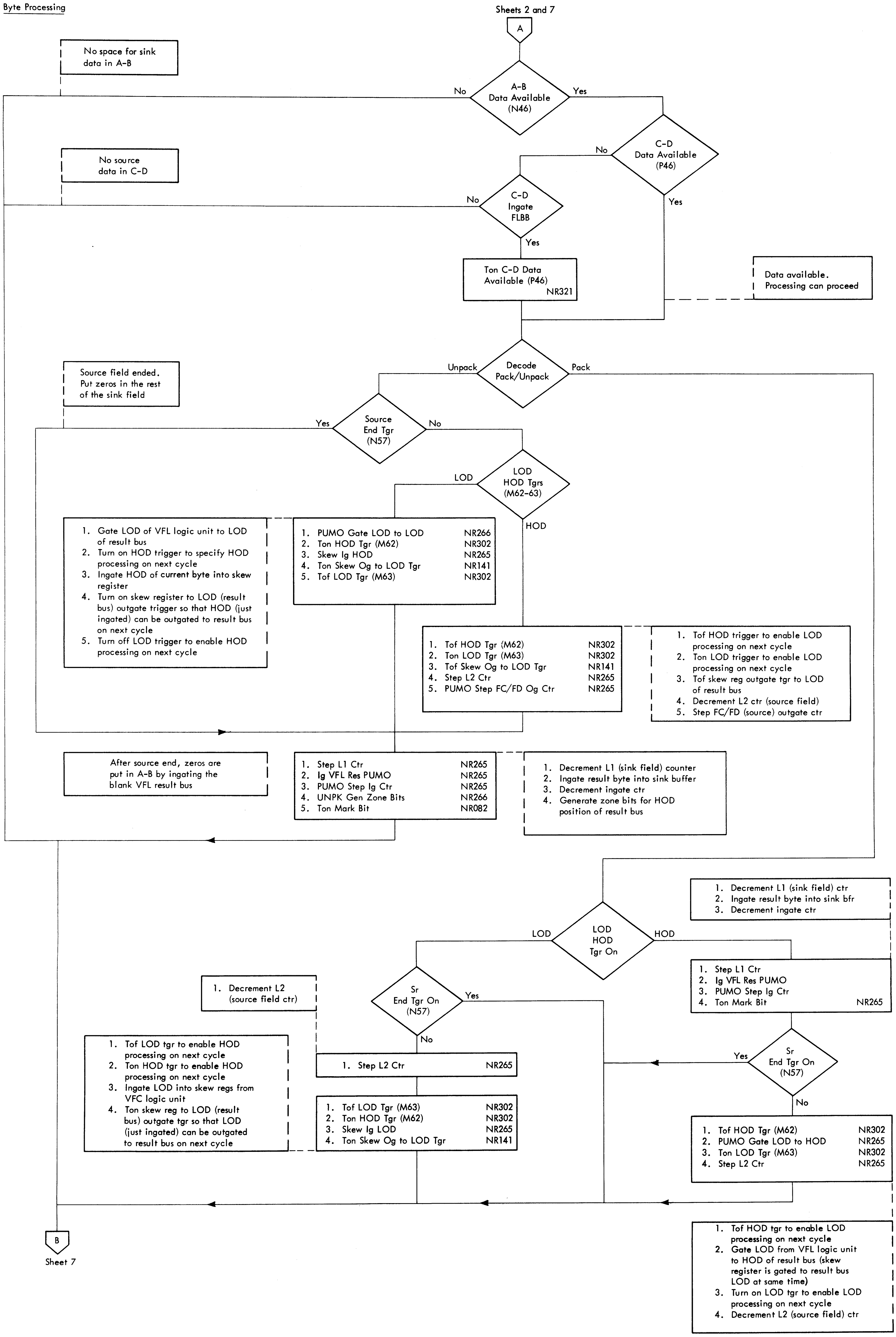

Sheets 2 and 7
A
No space for sink data in A-B
No
A-B Data Available (N46)
Yes
No source data in C-D
No
C-D Data Available (P46)
No
C-D Ingate FLBB
Yes
Yes
Ton C-D Data Available (P46) NR321
Data available. Processing can proceed
Source field ended. Put zeros in the rest of the sink field
Unpack
Decode Pack/Unpack
Pack
Yes
Source End Tgr (N57)
No
LOD
LOD HOD Tgrs (M62-63)
HOD
1. Gate LOD of VFL logic unit to LOD of result bus
2. Turn on HOD trigger to specify HOD processing on next cycle
3. Ingate HOD of current byte into skew register
4. Turn on skew register to LOD (result bus) outgate trigger so that HOD (just ingated) can be outgated to result bus on next cycle
5. Turn off LOD trigger to enable HOD processing on next cycle
1. PUMO Gate LOD to LOD NR266
2. Ton HOD Tgr (M62) NR302
3. Skew Ig HOD NR265
4. Ton Skew Og to LOD Tgr NR141
5. Tof LOD Tgr (M63) NR302
1. Tof HOD Tgr (M62) NR302
2. Ton LOD Tgr (M63) NR302
3. Tof Skew Og to LOD Tgr NR141
4. Step L2 Ctr NR265
5. PUMO Step FC/FD Og Ctr NR265
1. Tof HOD trigger to enable LOD processing on next cycle
2. Ton LOD trigger to enable LOD processing on next cycle
3. Tof skew reg outgate tgr to LOD of result bus
4. Decrement L2 ctr (source field)
5. Step FC/FD (source) outgate ctr
After source end, zeros are put in A-B by ingating the blank VFL result bus
1. Step L1 Ctr NR265
2. Ig VFL Res PUMO NR265
3. PUMO Step Ig Ctr NR265
4. UNPK Gen Zone Bits NR266
5. Ton Mark Bit NR082
1. Decrement L1 (sink field) counter
2. Ingate result byte into sink buffer
3. Decrement ingate ctr
4. Generate zone bits for HOD position of result bus
1. Decrement L1 (sink field) ctr
2. Ingate result byte into sink bfr
3. Decrement ingate ctr
LOD
LOD HOD Tgr On
HOD
1. Decrement L2 (source field ctr)
Sr End Tgr On (N57)
Yes
No
1. Step L1 Ctr
2. Ig VFL Res PUMO
3. PUMO Step Ig Ctr
4. Ton Mark Bit NR265
1. Tof LOD tgr to enable HOD processing on next cycle
2. Ton HOD tgr to enable HOD processing on next cycle
3. Ingate LOD into skew regs from VFC logic unit
4. Ton skew reg to LOD (result bus) outgate tgr so that LOD (just ingated) can be outgated to result bus on next cycle
1. Step L2 Ctr NR265
Yes
Sr End Tgr On (N57)
No
1. Tof LOD Tgr (M63) NR302
2. Ton HOD Tgr (M62) NR302
3. Skew Ig LOD NR265
4. Ton Skew Og to LOD Tgr NR141
1. Tof HOD Tgr (M62) NR302
2. PUMO Gate LOD to HOD NR265
3. Ton LOD Tgr (M63) NR302
4. Step L2 Ctr NR265
B
Sheet 7
1. Tof HOD tgr to enable LOD processing on next cycle
2. Gate LOD from VFL logic unit to HOD of result bus (skew register is gated to result bus LOD at same time)
3. Turn on LOD tgr to enable LOD processing on next cycle
4. Decrement L2 (source field) ctr

DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 4 OF 7)

Sink and Store

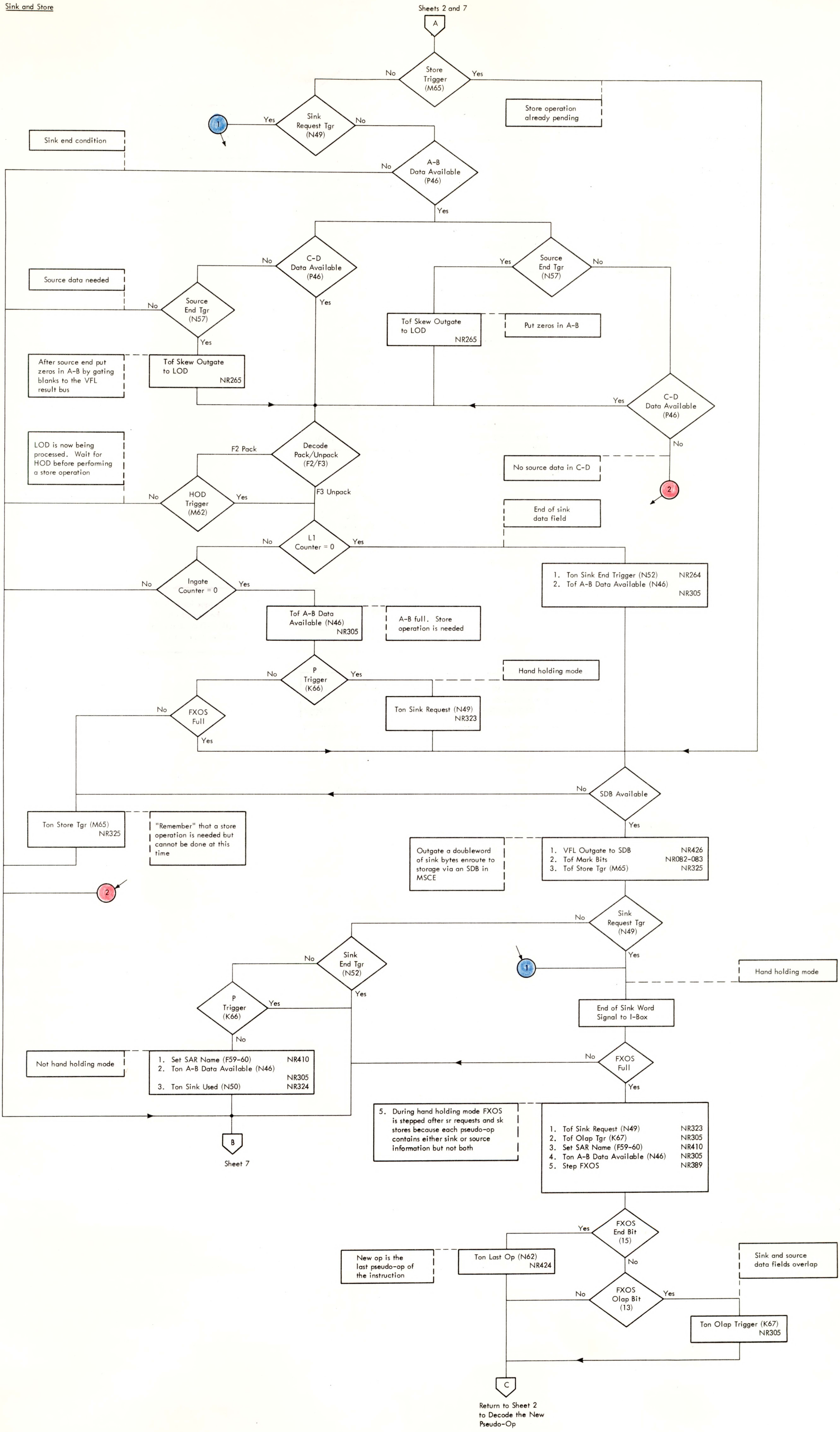

Sheets 2 and 7
A
Store Trigger (M65)
No
Yes
Store operation already pending
Sink Request Tgr (N49)
Yes
No
1
Sink end condition
A-B Data Available (P46)
No
Yes
C-D Data Available (P46)
No
Yes
Source End Tgr (N57)
Yes
No
Source data needed
Source End Tgr (N57)
No
Yes
Tof Skew Outgate to LOD NR265
Put zeros in A-B
After source end put zeros in A-B by gating blanks to the VFL result bus
Tof Skew Outgate to LOD NR265
C-D Data Available (P46)
Yes
No
LOD is now being processed. Wait for HOD before performing a store operation
Decode Pack/Unpack (F2/F3)
F2 Pack
F3 Unpack
HOD Trigger (M62)
No
Yes
No source data in C-D
2
End of sink data field
L1 Counter = 0
No
Yes
Ingate Counter = 0
No
Yes
1. Ton Sink End Trigger (N52) NR264
2. Tof A-B Data Available (N46) NR305
Tof A-B Data Available (N46) NR305
A-B full. Store operation is needed
P Trigger (K66)
No
Yes
Hand holding mode
FXOS Full
No
Yes
Ton Sink Request (N49) NR323
SDB Available
No
Yes
Ton Store Tgr (M65) NR325
"Remember" that a store operation is needed but cannot be done at this time
2
Outgate a doubleword of sink bytes enroute to storage via an SDB in MSCE
1. VFL Outgate to SDB NR426
2. Tof Mark Bits NR082-083
3. Tof Store Tgr (M65) NR325
Sink Request Tgr (N49)
No
Yes
Sink End Tgr (N52)
No
Yes
1
Hand holding mode
P Trigger (K66)
Yes
No
End of Sink Word Signal to I-Box
Not hand holding mode
1. Set SAR Name (F59-60) NR410
2. Ton A-B Data Available (N46) NR305
3. Ton Sink Used (N50) NR324
FXOS Full
No
Yes
B
Sheet 7
5. During hand holding mode FXOS is stepped after sr requests and sk stores because each pseudo-op contains either sink or source information but not both
1. Tof Sink Request (N49) NR323
2. Tof Olap Tgr (K67) NR305
3. Set SAR Name (F59-60) NR410
4. Ton A-B Data Available (N46) NR305
5. Step FXOS NR389
FXOS End Bit (15)
Yes
No
New op is the last pseudo-op of the instruction
Ton Last Op (N62) NR424
Sink and source data fields overlap
FXOS Olap Bit (13)
No
Yes
Ton Olap Trigger (K67) NR305
C
Return to Sheet 2 to Decode the New Pseudo-Op

DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 5 OF 7)

Overlap

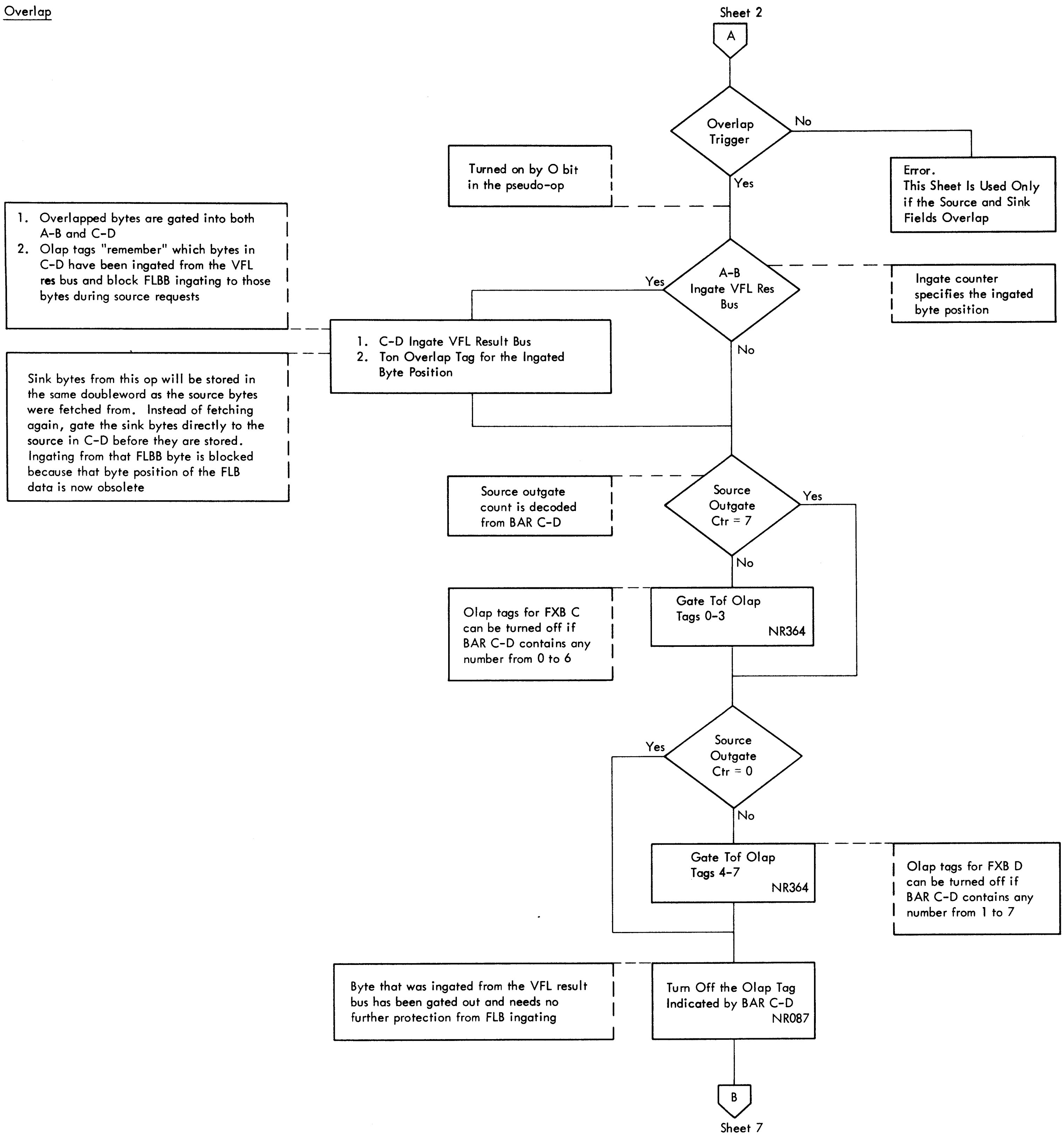

Sheet 2
A
Overlap Trigger
No
Error.
This Sheet Is Used Only if the Source and Sink Fields Overlap
Yes
Turned on by O bit in the pseudo-op
1. Overlapped bytes are gated into both A-B and C-D
2. Olap tags "remember" which bytes in C-D have been ingated from the VFL res bus and block FLBB ingating to those bytes during source requests
A-B Ingate VFL Res Bus
Yes
Ingate counter specifies the ingated byte position
1. C-D Ingate VFL Result Bus
2. Ton Overlap Tag for the Ingated Byte Position
No
Sink bytes from this op will be stored in the same doubleword as the source bytes were fetched from. Instead of fetching again, gate the sink bytes directly to the source in C-D before they are stored. Ingating from that FLBB byte is blocked because that byte position of the FLB data is now obsolete
Source outgate count is decoded from BAR C-D
Source Outgate Ctr = 7
Yes
No
Olap tags for FXB C can be turned off if BAR C-D contains any number from 0 to 6
Gate Tof Olap Tags 0-3
NR364
Source Outgate Ctr = 0
Yes
No
Gate Tof Olap Tags 4-7
NR364
Olap tags for FXB D can be turned off if BAR C-D contains any number from 1 to 7
Byte that was ingated from the VFL result bus has been gated out and needs no further protection from FLB ingating
Turn Off the Olap Tag Indicated by BAR C-D
NR087
B
Sheet 7

DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 6 OF 7)

Ending Sequence

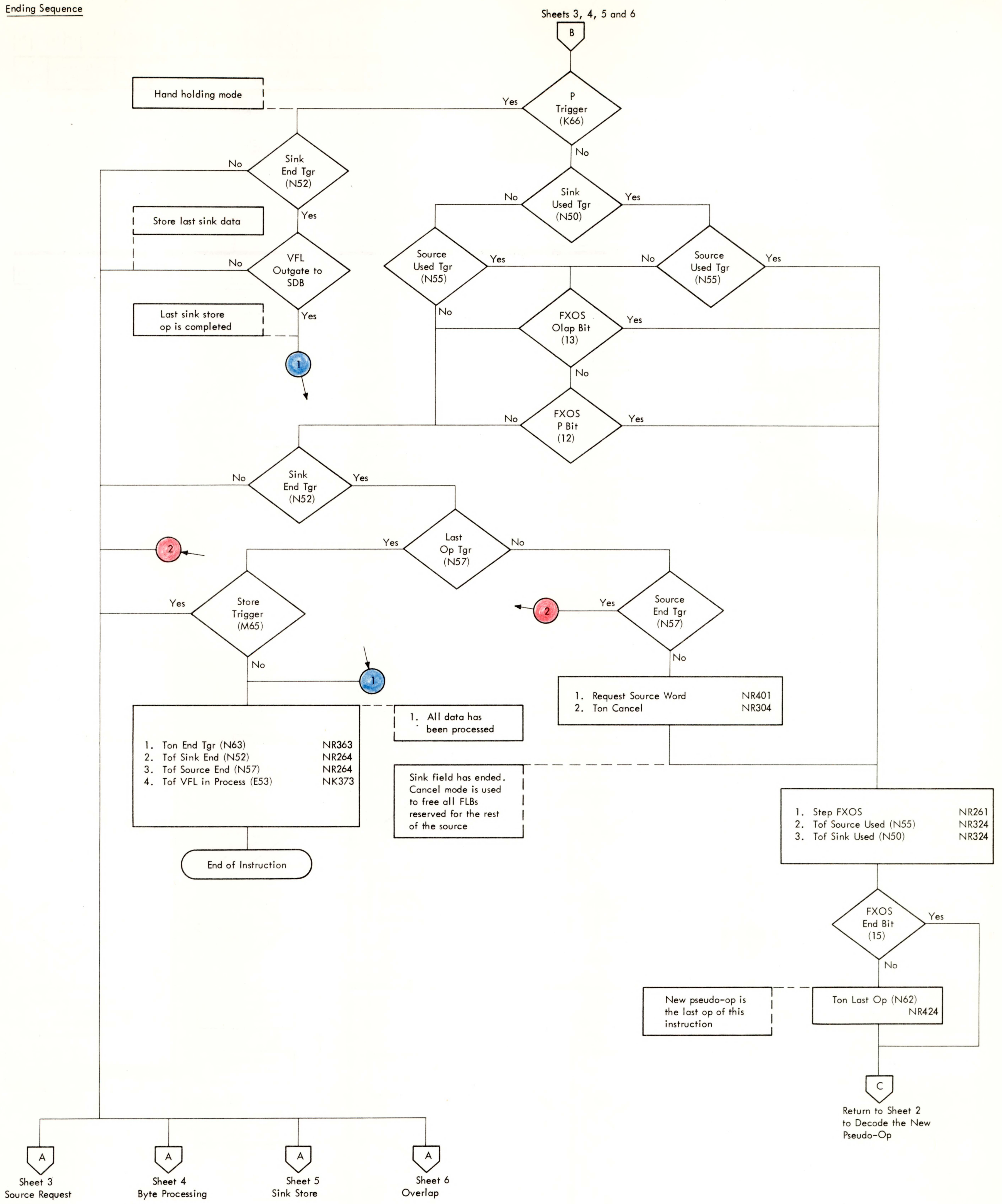


DIAGRAM 5-124. PACK AND UNPACK INSTRUCTIONS (SHEET 7 OF 7)

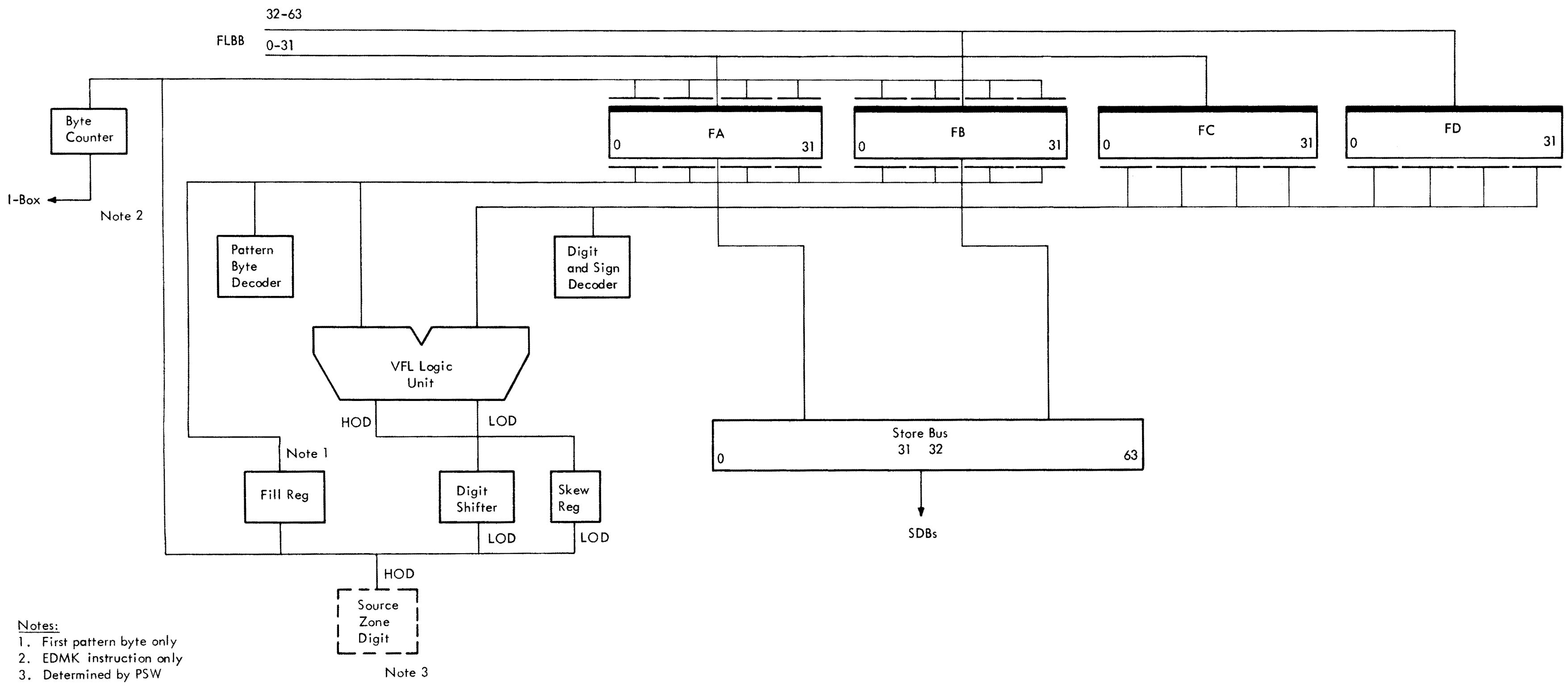
Data Flow

Objectives:
Edit, Edit and Mark
ED, EDMK

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

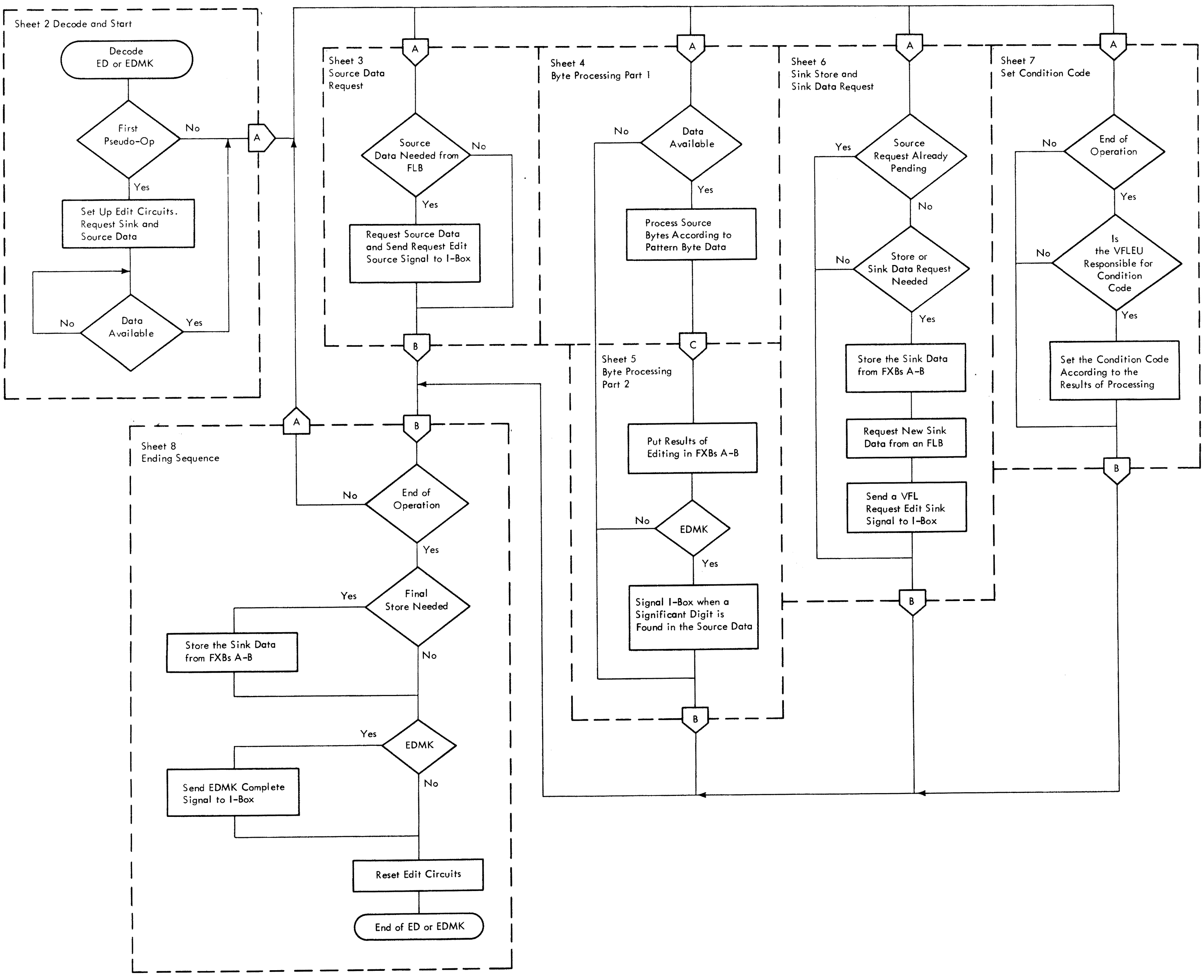
SS	DE, DF	▨	FLB Name (Source)	▨	S	E	FLB Name (Sink)	SAR Name

1. Source data are changed from packed to zone format and are edited under control of a pattern word (sink).
2. Edit and mark sends byte address of first significant result digit to I-box.
3. During editing process:
   a. Source digit is expended to zone format.
   b. Pattern byte is left unchanged.
   c. Fill character from fill register (first sink byte) is used.



Notes:
1. First pattern byte only
2. EDMK instruction only
3. Determined by PSW

Sheet Layout



Sheet 2 Decode and Start
Decode ED or EDMK
First Pseudo-Op
No
Yes
Set Up Edit Circuits. Request Sink and Source Data
Data Available
No
Yes
A
Sheet 3 Source Data Request
A
Source Data Needed from FLB
No
Yes
Request Source Data and Send Request Edit Source Signal to I-Box
B
Sheet 4 Byte Processing Part 1
A
Data Available
No
Yes
Process Source Bytes According to Pattern Byte Data
C
Sheet 5 Byte Processing Part 2
Put Results of Editing in FXBs A-B
EDMK
No
Yes
Signal I-Box when a Significant Digit is Found in the Source Data
B
Sheet 6 Sink Store and Sink Data Request
A
Source Request Already Pending
Yes
No
Store or Sink Data Request Needed
No
Yes
Store the Sink Data from FXBs A-B
Request New Sink Data from an FLB
Send a VFL Request Edit Sink Signal to I-Box
B
Sheet 7 Set Condition Code
A
End of Operation
No
Yes
Is the VFLEU Responsible for Condition Code
No
Yes
Set the Condition Code According to the Results of Processing
B
A
B
Sheet 8 Ending Sequence
End of Operation
No
Yes
Final Store Needed
Yes
No
Store the Sink Data from FXBs A-B
EDMK
Yes
No
Send EDMK Complete Signal to I-Box
Reset Edit Circuits
End of ED or EDMK

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 1 OF 8)

Decode and Start

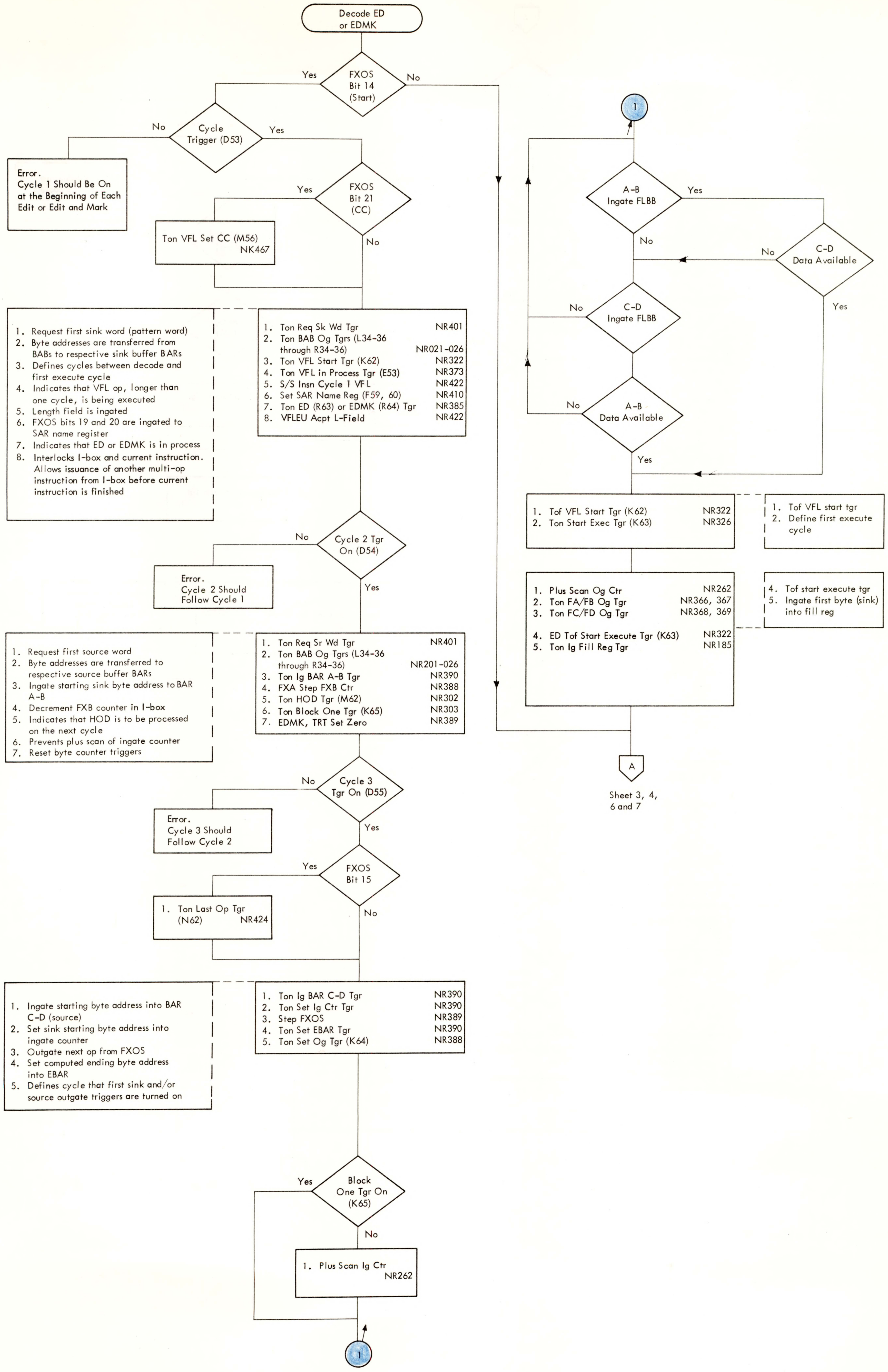

Decode ED or EDMK
FXOS Bit 14 (Start)
Yes
No
Cycle Trigger (D53)
No
Yes
Error.
Cycle 1 Should Be On at the Beginning of Each Edit or Edit and Mark
FXOS Bit 21 (CC)
Yes
No
Ton VFL Set CC (M56) NK467
1. Request first sink word (pattern word)
2. Byte addresses are transferred from BABs to respective sink buffer BARs
3. Defines cycles between decode and first execute cycle
4. Indicates that VFL op, longer than one cycle, is being executed
5. Length field is ingated
6. FXOS bits 19 and 20 are ingated to SAR name register
7. Indicates that ED or EDMK is in process
8. Interlocks I-box and current instruction. Allows issuance of another multi-op instruction from I-box before current instruction is finished
1. Ton Req Sk Wd Tgr NR401
2. Ton BAB Og Tgrs (L34-36 through R34-36) NR021-026
3. Ton VFL Start Tgr (K62) NR322
4. Ton VFL in Process Tgr (E53) NR373
5. S/S Insn Cycle 1 VFL NR422
6. Set SAR Name Reg (F59, 60) NR410
7. Ton ED (R63) or EDMK (R64) Tgr NR385
8. VFLEU Acpt L-Field NR422
Cycle 2 Tgr On (D54)
No
Yes
Error.
Cycle 2 Should Follow Cycle 1
1. Request first source word
2. Byte addresses are transferred to respective source buffer BARs
3. Ingate starting sink byte address to BAR A-B
4. Decrement FXB counter in I-box
5. Indicates that HOD is to be processed on the next cycle
6. Prevents plus scan of ingate counter
7. Reset byte counter triggers
1. Ton Req Sr Wd Tgr NR401
2. Ton BAB Og Tgrs (L34-36 through R34-36) NR201-026
3. Ton Ig BAR A-B Tgr NR390
4. FXA Step FXB Ctr NR388
5. Ton HOD Tgr (M62) NR302
6. Ton Block One Tgr (K65) NR303
7. EDMK, TRT Set Zero NR389
Cycle 3 Tgr On (D55)
No
Yes
Error.
Cycle 3 Should Follow Cycle 2
FXOS Bit 15
Yes
No
1. Ton Last Op Tgr (N62) NR424
1. Ingate starting byte address into BAR C-D (source)
2. Set sink starting byte address into ingate counter
3. Outgate next op from FXOS
4. Set computed ending byte address into EBAR
5. Defines cycle that first sink and/or source outgate triggers are turned on
1. Ton Ig BAR C-D Tgr NR390
2. Ton Set Ig Ctr Tgr NR390
3. Step FXOS NR389
4. Ton Set EBAR Tgr NR390
5. Ton Set Og Tgr (K64) NR388
Block One Tgr On (K65)
Yes
No
1. Plus Scan Ig Ctr NR262
1
1
A-B Ingate FLBB
Yes
No
C-D Data Available
No
Yes
C-D Ingate FLBB
No
A-B Data Available
No
Yes
1. Tof VFL Start Tgr (K62) NR322
2. Ton Start Exec Tgr (K63) NR326
1. Tof VFL start tgr
2. Define first execute cycle
1. Plus Scan Og Ctr NR262
2. Ton FA/FB Og Tgr NR366, 367
3. Ton FC/FD Og Tgr NR368, 369
4. ED Tof Start Execute Tgr (K63) NR322
5. Ton Ig Fill Reg Tgr NR185
4. Tof start execute tgr
5. Ingate first byte (sink) into fill reg
A
Sheet 3, 4, 6 and 7

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 2 OF 8)

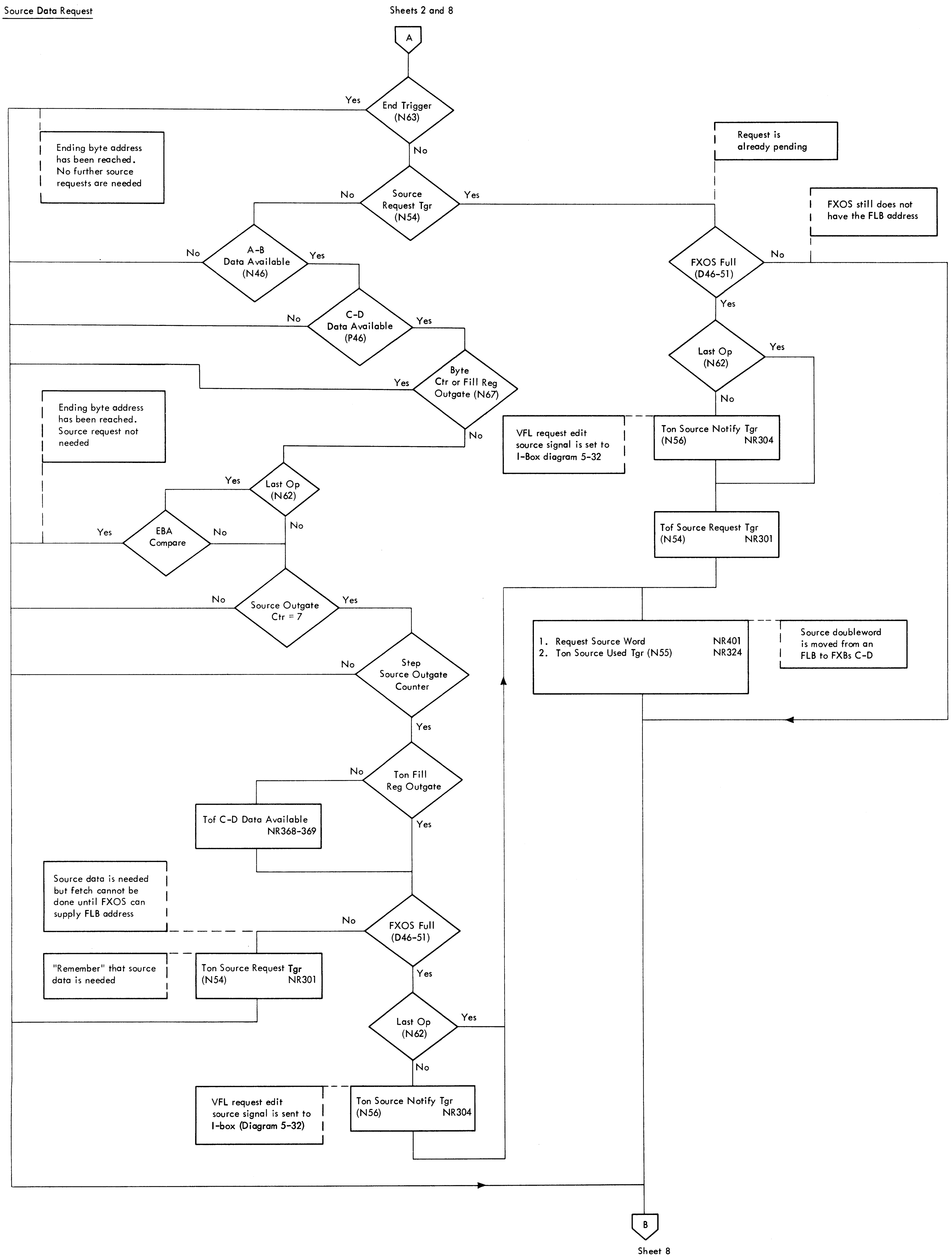

Source Data Request
Sheets 2 and 8
A
End Trigger (N63)
Yes
No
Ending byte address has been reached. No further source requests are needed
Source Request Tgr (N54)
No
Yes
Request is already pending
FXOS still does not have the FLB address
A-B Data Available (N46)
No
Yes
FXOS Full (D46-51)
No
Yes
C-D Data Available (P46)
No
Yes
Last Op (N62)
Yes
No
Byte Ctr or Fill Reg Outgate (N67)
Yes
No
Ending byte address has been reached. Source request not needed
VFL request edit source signal is set to I-Box diagram 5-32
Ton Source Notify Tgr (N56) NR304
Last Op (N62)
Yes
No
EBA Compare
Yes
No
Tof Source Request Tgr (N54) NR301
Source Outgate Ctr = 7
No
Yes
1. Request Source Word NR401
2. Ton Source Used Tgr (N55) NR324
Source doubleword is moved from an FLB to FXBs C-D
Step Source Outgate Counter
No
Yes
Ton Fill Reg Outgate
No
Yes
Tof C-D Data Available NR368-369
Source data is needed but fetch cannot be done until FXOS can supply FLB address
FXOS Full (D46-51)
No
Yes
"Remember" that source data is needed
Ton Source Request Tgr (N54) NR301
Last Op (N62)
Yes
No
VFL request edit source signal is sent to I-box (Diagram 5-32)
Ton Source Notify Tgr (N56) NR304
B
Sheet 8

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 3 OF 8)

Byte Processing Part 2

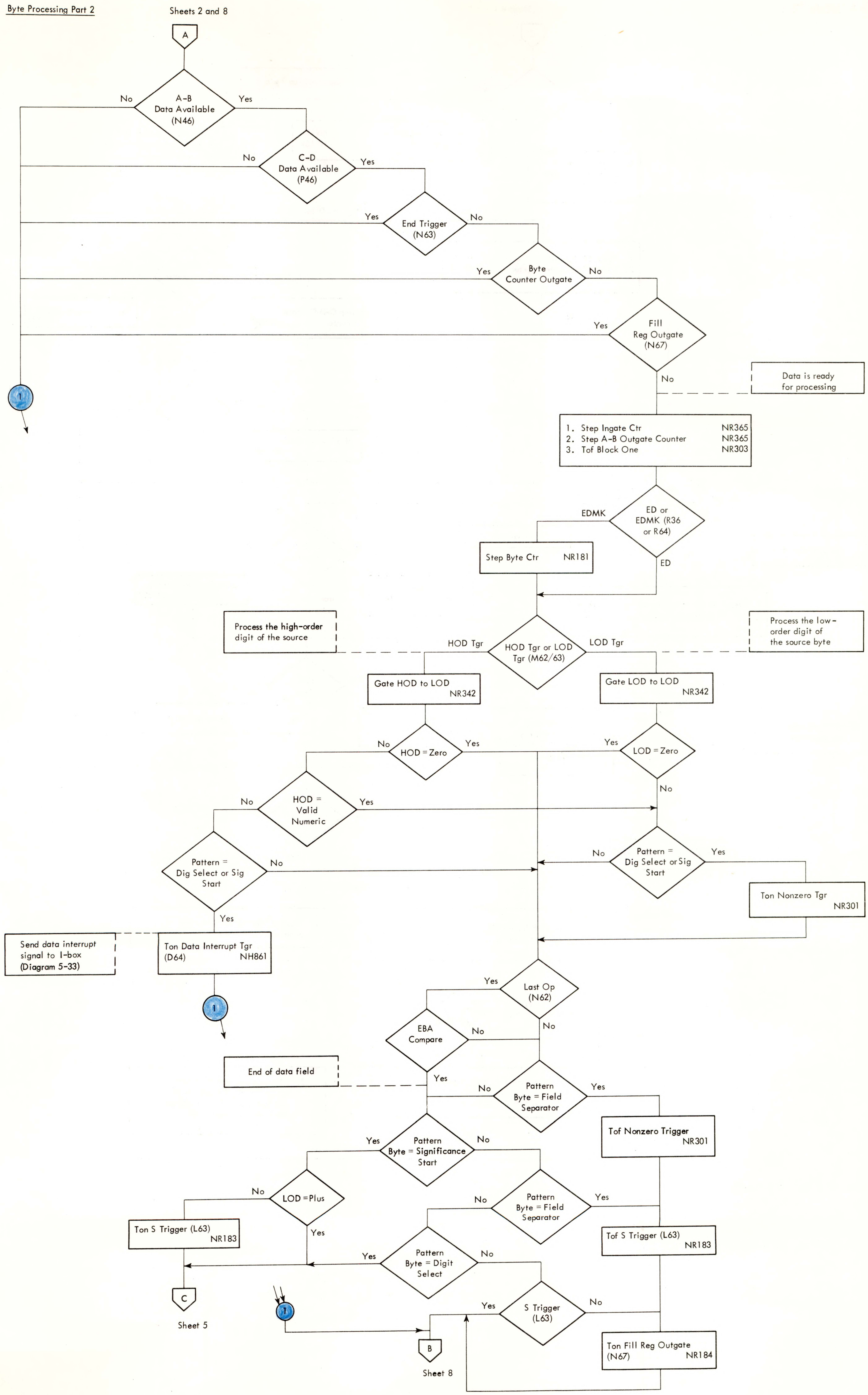

Sheets 2 and 8
A
No
A-B Data Available (N46)
Yes
No
C-D Data Available (P46)
Yes
Yes
End Trigger (N63)
No
Yes
Byte Counter Outgate
No
Yes
Fill Reg Outgate (N67)
No
Data is ready for processing
1
1. Step Ingate Ctr NR365
2. Step A-B Outgate Counter NR365
3. Tof Block One NR303
EDMK
ED or EDMK (R36 or R64)
ED
Step Byte Ctr NR181
Process the high-order digit of the source
HOD Tgr
HOD Tgr or LOD Tgr (M62/63)
LOD Tgr
Process the low-order digit of the source byte
Gate HOD to LOD NR342
Gate LOD to LOD NR342
No
HOD = Zero
Yes
Yes
LOD = Zero
No
No
HOD = Valid Numeric
Yes
Pattern = Dig Select or Sig Start
No
No
Pattern = Dig Select or Sig Start
Yes
Ton Nonzero Tgr NR301
Yes
Send data interrupt signal to I-box (Diagram 5-33)
Ton Data Interrupt Tgr (D64) NH861
1
Yes
Last Op (N62)
No
EBA Compare
No
End of data field
Yes
No
Pattern Byte = Field Separator
Yes
Tof Nonzero Trigger NR301
Yes
Pattern Byte = Significance Start
No
No
LOD = Plus
Yes
Ton S Trigger (L63) NR183
No
Pattern Byte = Field Separator
Yes
Tof S Trigger (L63) NR183
Yes
Pattern Byte = Digit Select
No
C
Sheet 5
1
Yes
S Trigger (L63)
No
B
Sheet 8
Ton Fill Reg Outgate (N67) NR184

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 4 OF 8)

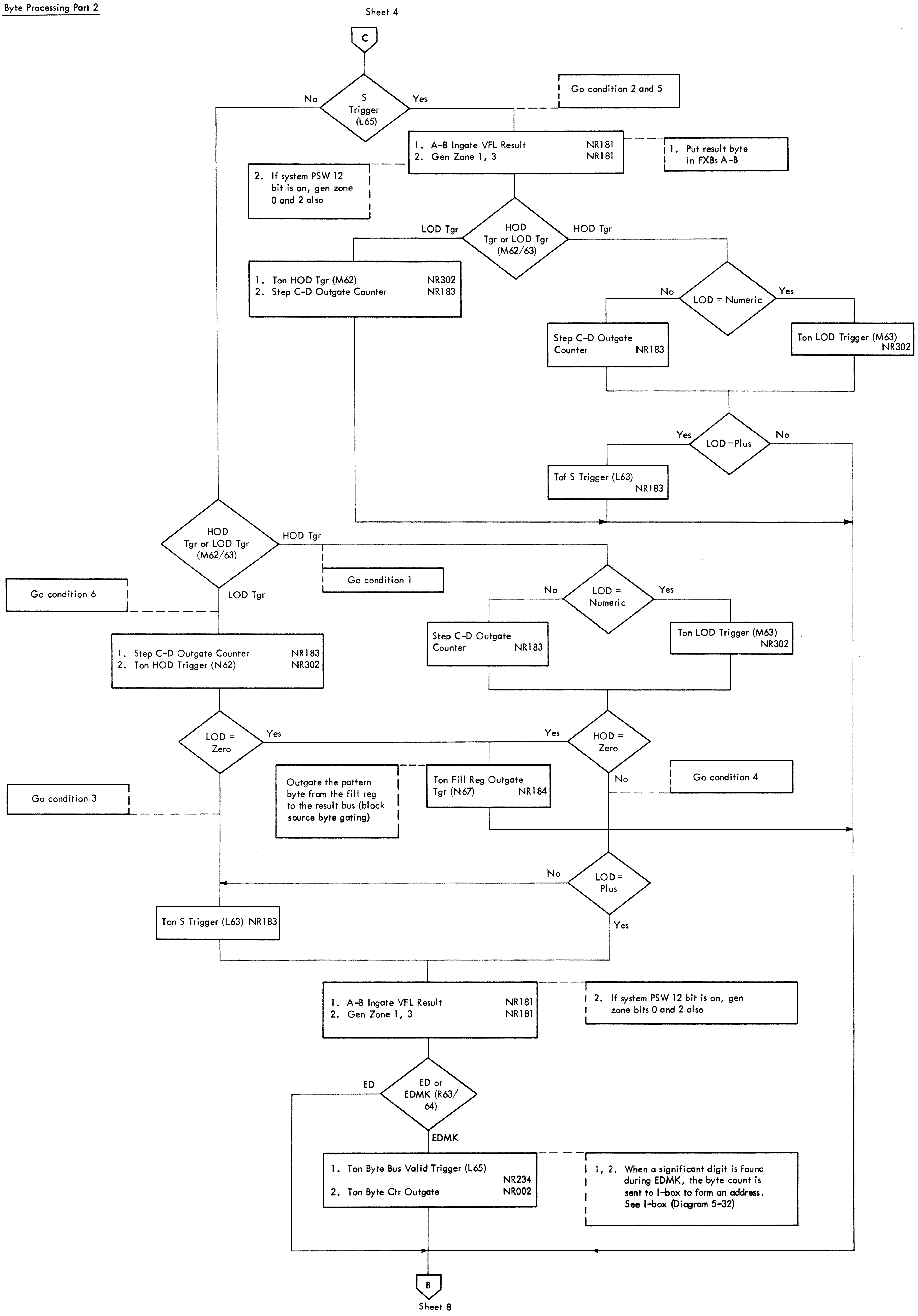

Sheet 4
C
S Trigger (L65)
No
Yes
Go condition 2 and 5
1. A-B Ingate VFL Result NR181
2. Gen Zone 1, 3 NR181
1. Put result byte in FXBs A-B
2. If system PSW 12 bit is on, gen zone 0 and 2 also
HOD Tgr or LOD Tgr (M62/63)
LOD Tgr
HOD Tgr
1. Ton HOD Tgr (M62) NR302
2. Step C-D Outgate Counter NR183
LOD = Numeric
No
Yes
Step C-D Outgate Counter NR183
Ton LOD Trigger (M63) NR302
LOD =Plus
Yes
No
Tof S Trigger (L63) NR183
HOD Tgr or LOD Tgr (M62/63)
HOD Tgr
LOD Tgr
Go condition 1
Go condition 6
1. Step C-D Outgate Counter NR183
2. Ton HOD Trigger (N62) NR302
LOD = Numeric
No
Yes
Step C-D Outgate Counter NR183
Ton LOD Trigger (M63) NR302
LOD = Zero
Yes
HOD = Zero
Yes
No
Go condition 3
Outgate the pattern byte from the fill reg to the result bus (block source byte gating)
Ton Fill Reg Outgate Tgr (N67) NR184
Go condition 4
LOD= Plus
No
Yes
Ton S Trigger (L63) NR183
1. A-B Ingate VFL Result NR181
2. Gen Zone 1, 3 NR181
2. If system PSW 12 bit is on, gen zone bits 0 and 2 also
ED or EDMK (R63/64)
ED
EDMK
1. Ton Byte Bus Valid Trigger (L65) NR234
2. Ton Byte Ctr Outgate NR002
1, 2. When a significant digit is found during EDMK, the byte count is sent to I-box to form an address. See I-box (Diagram 5-32)
B
Sheet 8

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 5 OF 8)

Sink Store and Sink Data Request

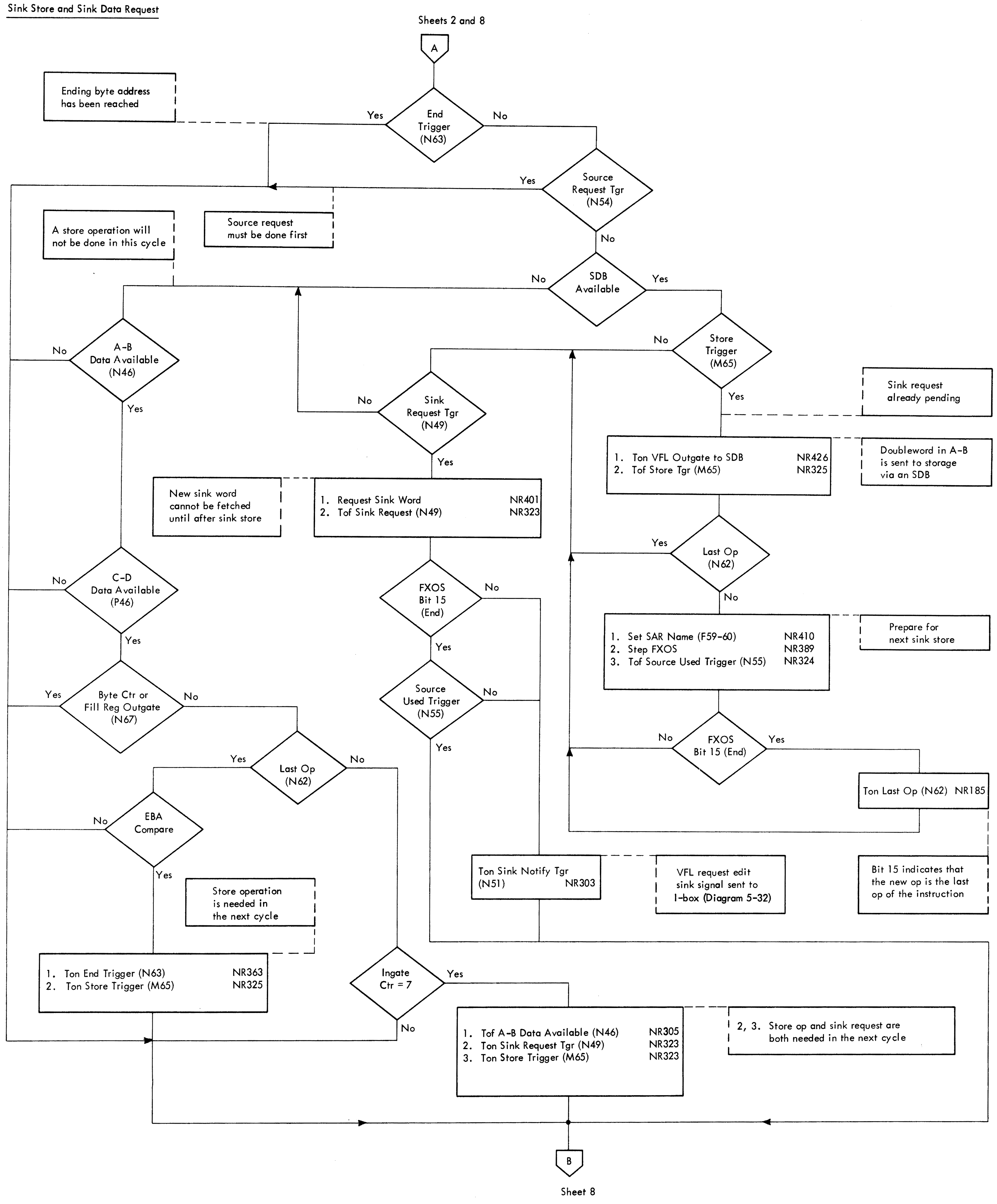

Sheets 2 and 8
A
Ending byte address has been reached
End Trigger (N63)
Yes
No
Source Request Tgr (N54)
Yes
No
Source request must be done first
A store operation will not be done in this cycle
SDB Available
No
Yes
A-B Data Available (N46)
No
Yes
Store Trigger (M65)
No
Yes
Sink request already pending
Sink Request Tgr (N49)
No
Yes
1. Ton VFL Outgate to SDB NR426
2. Tof Store Tgr (M65) NR325
Doubleword in A-B is sent to storage via an SDB
New sink word cannot be fetched until after sink store
1. Request Sink Word NR401
2. Tof Sink Request (N49) NR323
Last Op (N62)
Yes
No
C-D Data Available (P46)
No
Yes
FXOS Bit 15 (End)
No
Yes
1. Set SAR Name (F59-60) NR410
2. Step FXOS NR389
3. Tof Source Used Trigger (N55) NR324
Prepare for next sink store
Byte Ctr or Fill Reg Outgate (N67)
Yes
No
Source Used Trigger (N55)
No
Yes
FXOS Bit 15 (End)
No
Yes
Last Op (N62)
Yes
No
Ton Last Op (N62) NR185
EBA Compare
No
Yes
Ton Sink Notify Tgr (N51) NR303
VFL request edit sink signal sent to I-box (Diagram 5-32)
Bit 15 indicates that the new op is the last op of the instruction
Store operation is needed in the next cycle
1. Ton End Trigger (N63) NR363
2. Ton Store Trigger (M65) NR325
Ingate Ctr = 7
Yes
No
1. Tof A-B Data Available (N46) NR305
2. Ton Sink Request Tgr (N49) NR323
3. Ton Store Trigger (M65) NR323
2, 3. Store op and sink request are both needed in the next cycle
B
Sheet 8

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 6 OF 8)

Set Condition Code

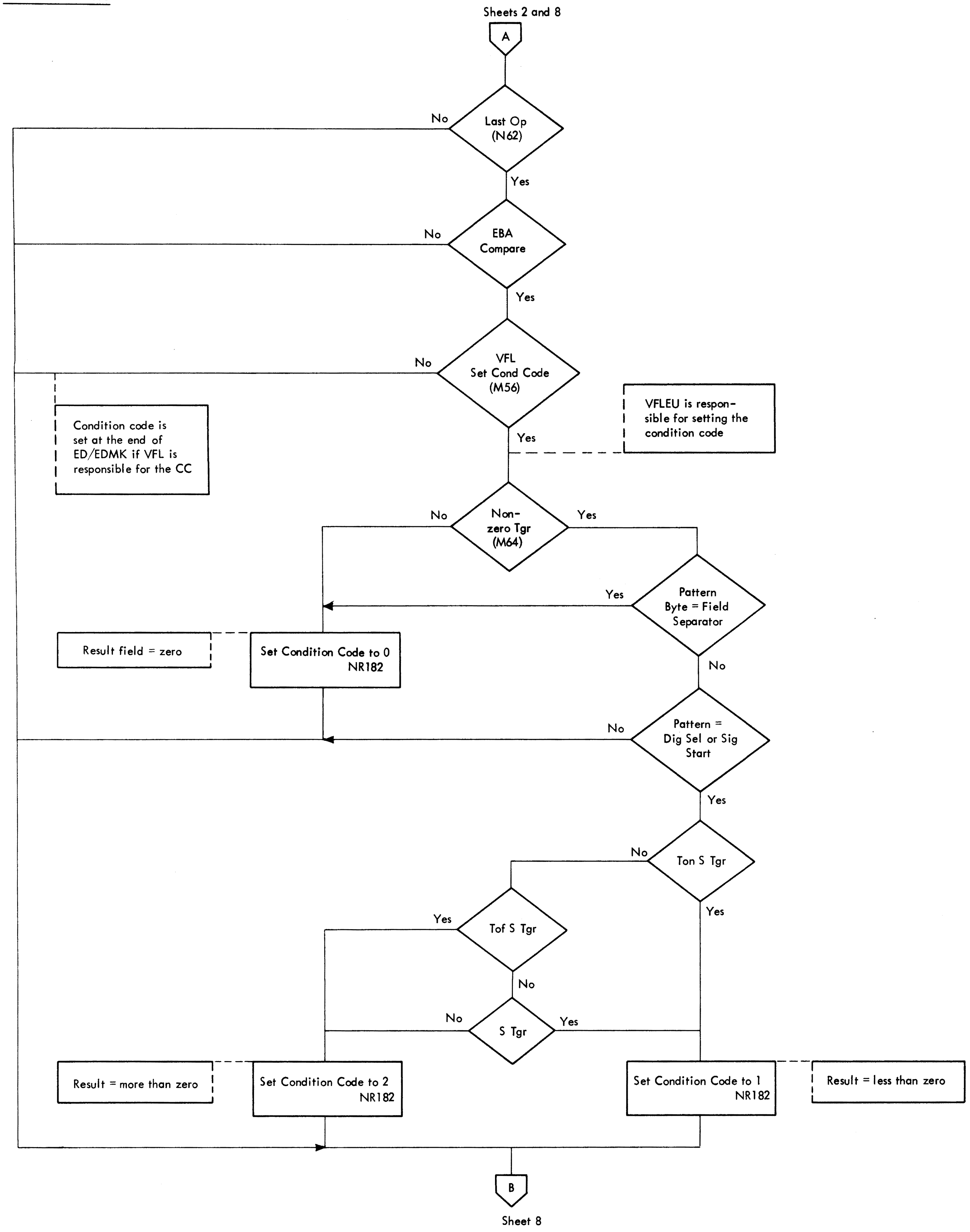

Sheets 2 and 8
A
Last Op (N62)
No
Yes
EBA Compare
No
Yes
VFL Set Cond Code (M56)
No
Yes
Condition code is set at the end of ED/EDMK if VFL is responsible for the CC
VFLEU is responsible for setting the condition code
Non-zero Tgr (M64)
No
Yes
Pattern Byte = Field Separator
Yes
No
Result field = zero
Set Condition Code to 0 NR182
Pattern = Dig Sel or Sig Start
No
Yes
Ton S Tgr
No
Yes
Tof S Tgr
Yes
No
S Tgr
No
Yes
Result = more than zero
Set Condition Code to 2 NR182
Set Condition Code to 1 NR182
Result = less than zero
B
Sheet 8

DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 7 OF 8)

Ending Sequence

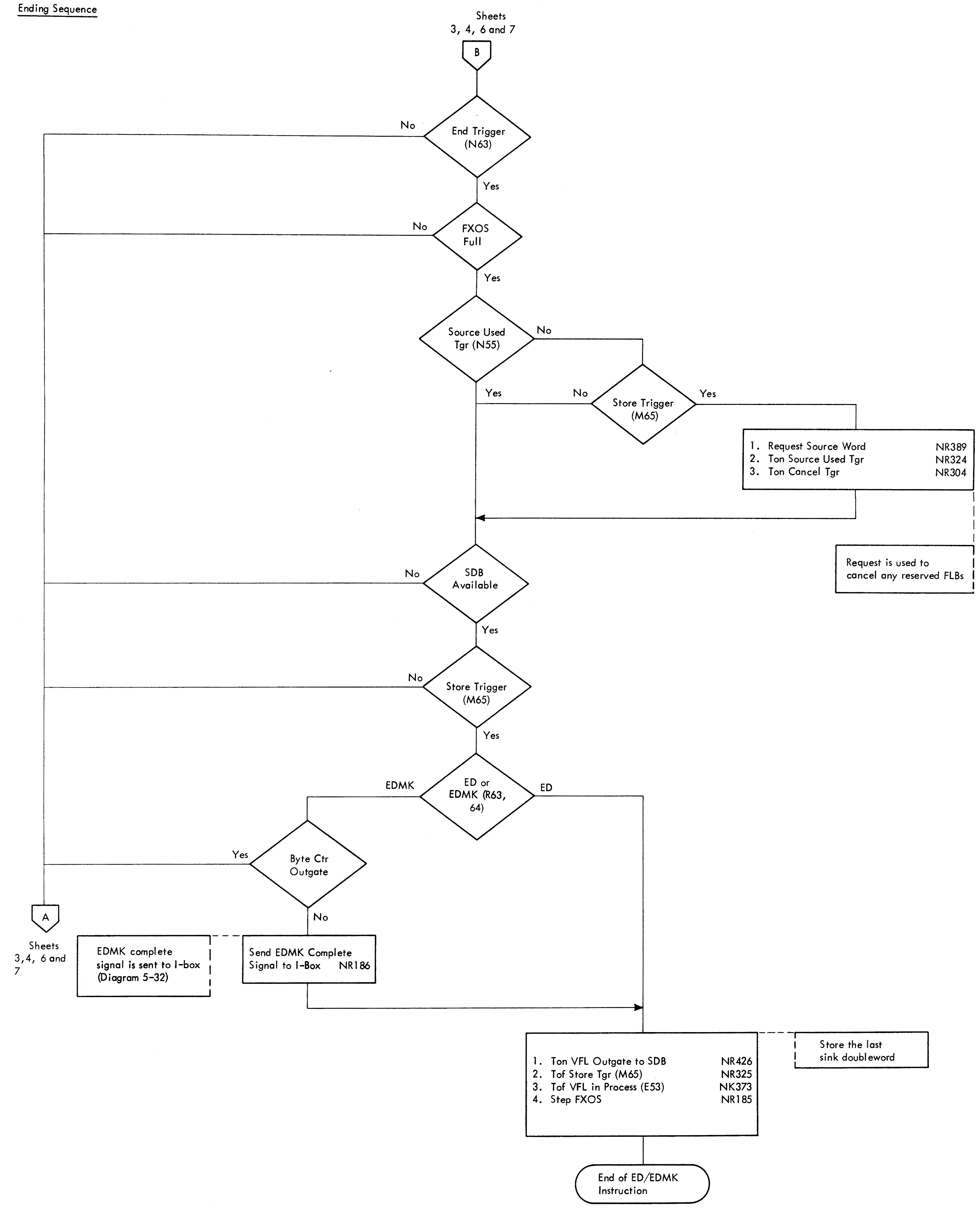
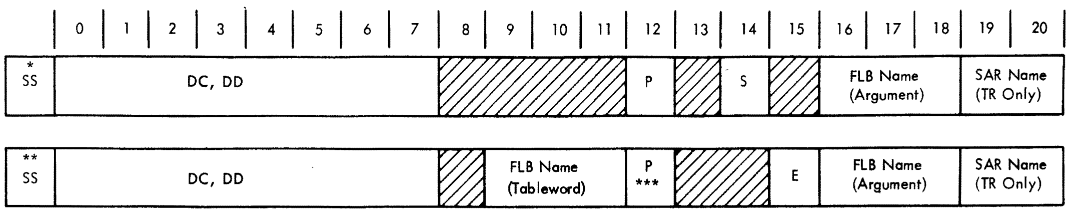


DIAGRAM 5-125. ED AND EDMK INSTRUCTION (SHEET 8 OF 8)

Objectives:

Translate, Translate and Test

TR, TRT



*First op only **Subsequent ops ***P bit not present in end op

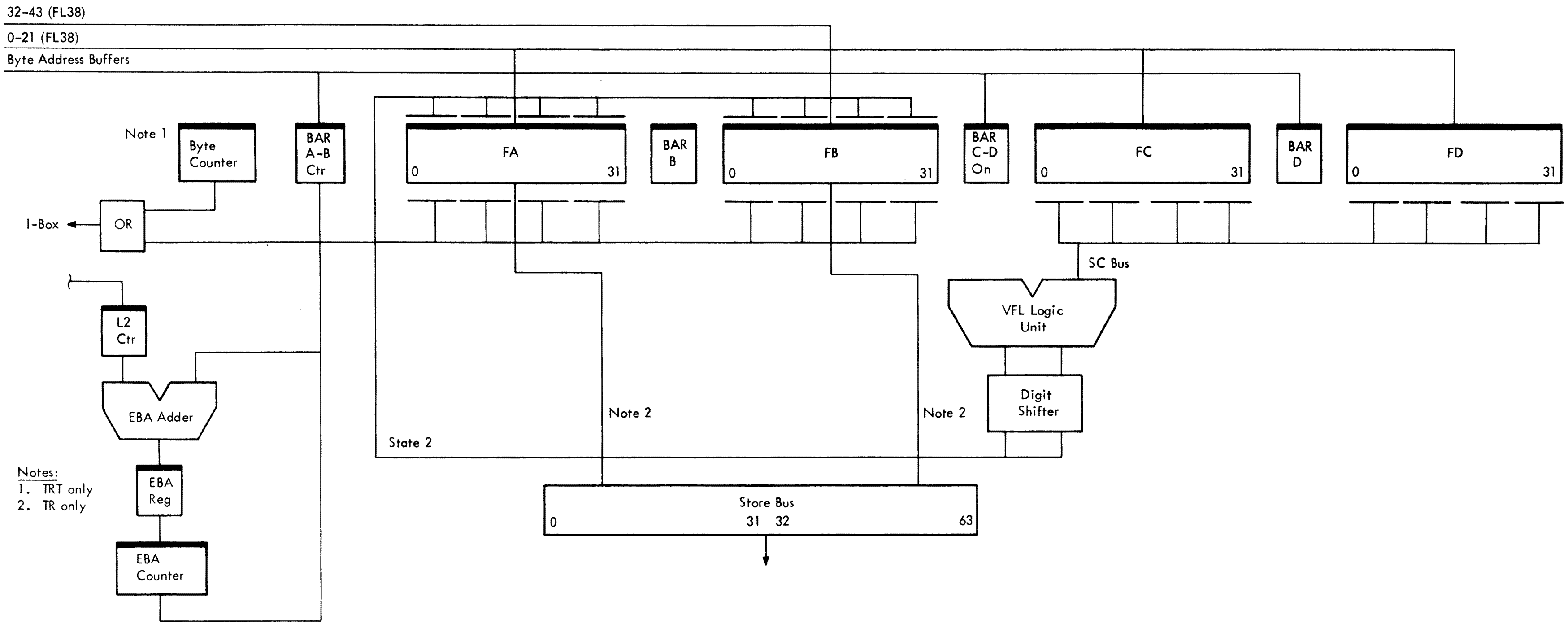
1. Data bytes from the argument word are used to reference data bytes in the tableword.
2. TR operation replaces the original argument bytes with the tableword bytes.
3. TRT operation checks tableword bytes for zero/nonzero condition. Nonzero condition causes instruction to terminate.
4. Basic operation is as follows:
   a. Argument bytes are sent to I-box to form address of tableword containing desired table bytes.
   b. Addressed tablewords are alternately gated to FC and FD.
   c. Desired table bytes are outgated from tablewords in FC and FD to VFL logic unit. in TR operation, these bytes replace the argument word. In TRT operation, these bytes are checked and the condition code is set accordingly.

Condition Code:

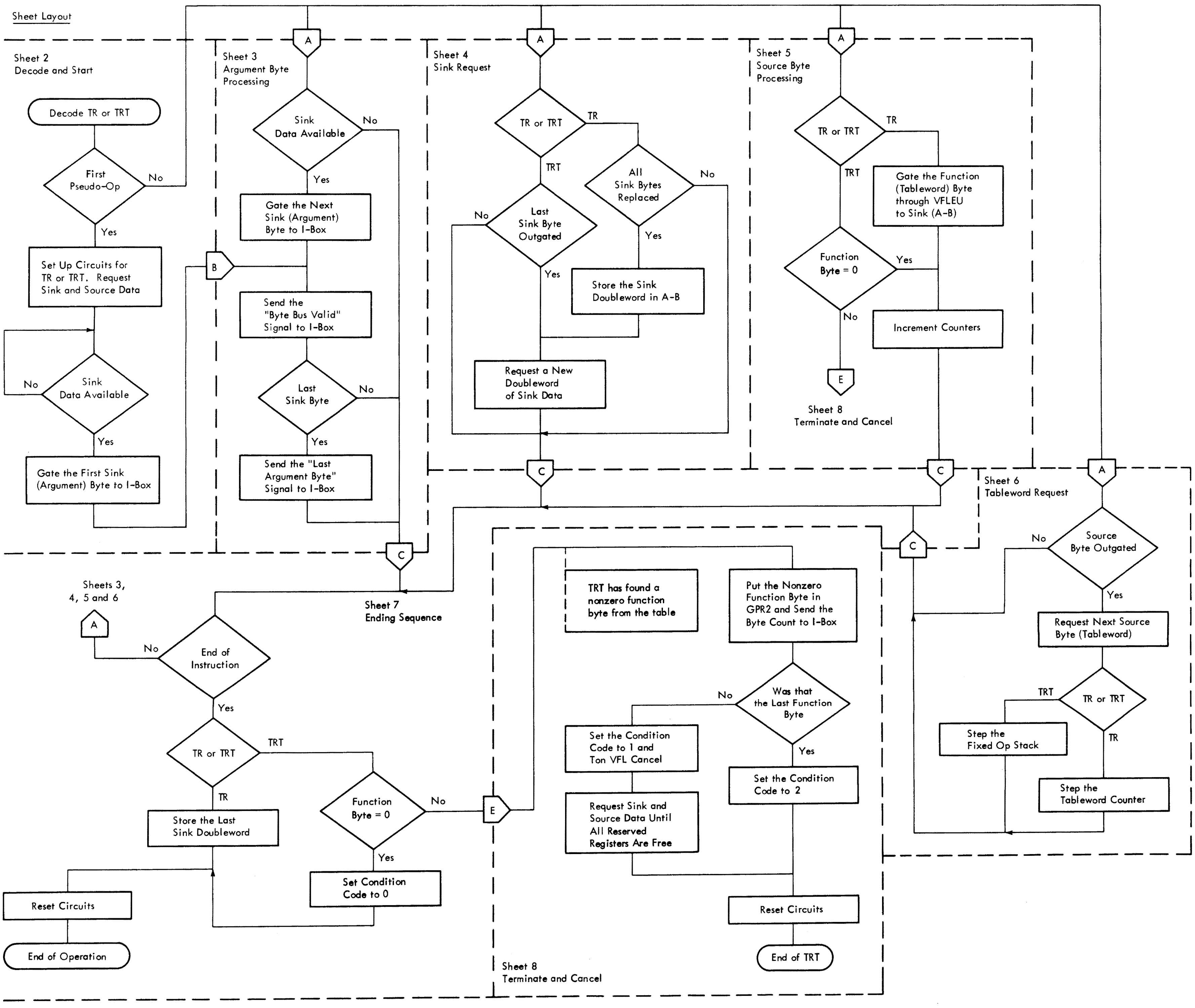
TR - Code remains unchanged

TRT - 0 All function bytes are zero
1 Nonzero function byte before the first operand is exhausted
2 Last function byte is nonzero

Data Flow



Sheet Layout



Sheet 2
Decode and Start
Decode TR or TRT
First Pseudo-Op
No
Yes
Set Up Circuits for TR or TRT. Request Sink and Source Data
Sink Data Available
No
Yes
Gate the First Sink (Argument) Byte to I-Box
Sheet 3
Argument Byte Processing
A
Sink Data Available
No
Yes
Gate the Next Sink (Argument) Byte to I-Box
B
Send the "Byte Bus Valid" Signal to I-Box
Last Sink Byte
No
Yes
Send the "Last Argument Byte" Signal to I-Box
Sheet 4
Sink Request
A
TR or TRT
TR
TRT
Last Sink Byte Outgated
No
Yes
All Sink Bytes Replaced
No
Yes
Store the Sink Doubleword in A-B
Request a New Doubleword of Sink Data
C
Sheet 5
Source Byte Processing
A
TR or TRT
TR
TRT
Gate the Function (Tableword) Byte through VFLEU to Sink (A-B)
Function Byte = 0
Yes
No
Increment Counters
E
Sheet 8
Terminate and Cancel
C
Sheet 6
Tableword Request
A
Source Byte Outgated
No
Yes
Request Next Source Byte (Tableword)
TRT or TR
TRT
TR
Step the Fixed Op Stack
Step the Tableword Counter
C
C
Sheet 7
Ending Sequence
Sheets 3, 4, 5 and 6
A
End of Instruction
No
Yes
TR or TRT
TRT
TR
Store the Last Sink Doubleword
Function Byte = 0
No
Yes
Set Condition Code to 0
Reset Circuits
End of Operation
E
TRT has found a nonzero function byte from the table
Put the Nonzero Function Byte in GPR2 and Send the Byte Count to I-Box
Was that the Last Function Byte
No
Yes
Set the Condition Code to 1 and Ton VFL Cancel
Request Sink and Source Data Until All Reserved Registers Are Free
Set the Condition Code to 2
Reset Circuits
End of TRT
Sheet 8
Terminate and Cancel

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 1 OF 8)

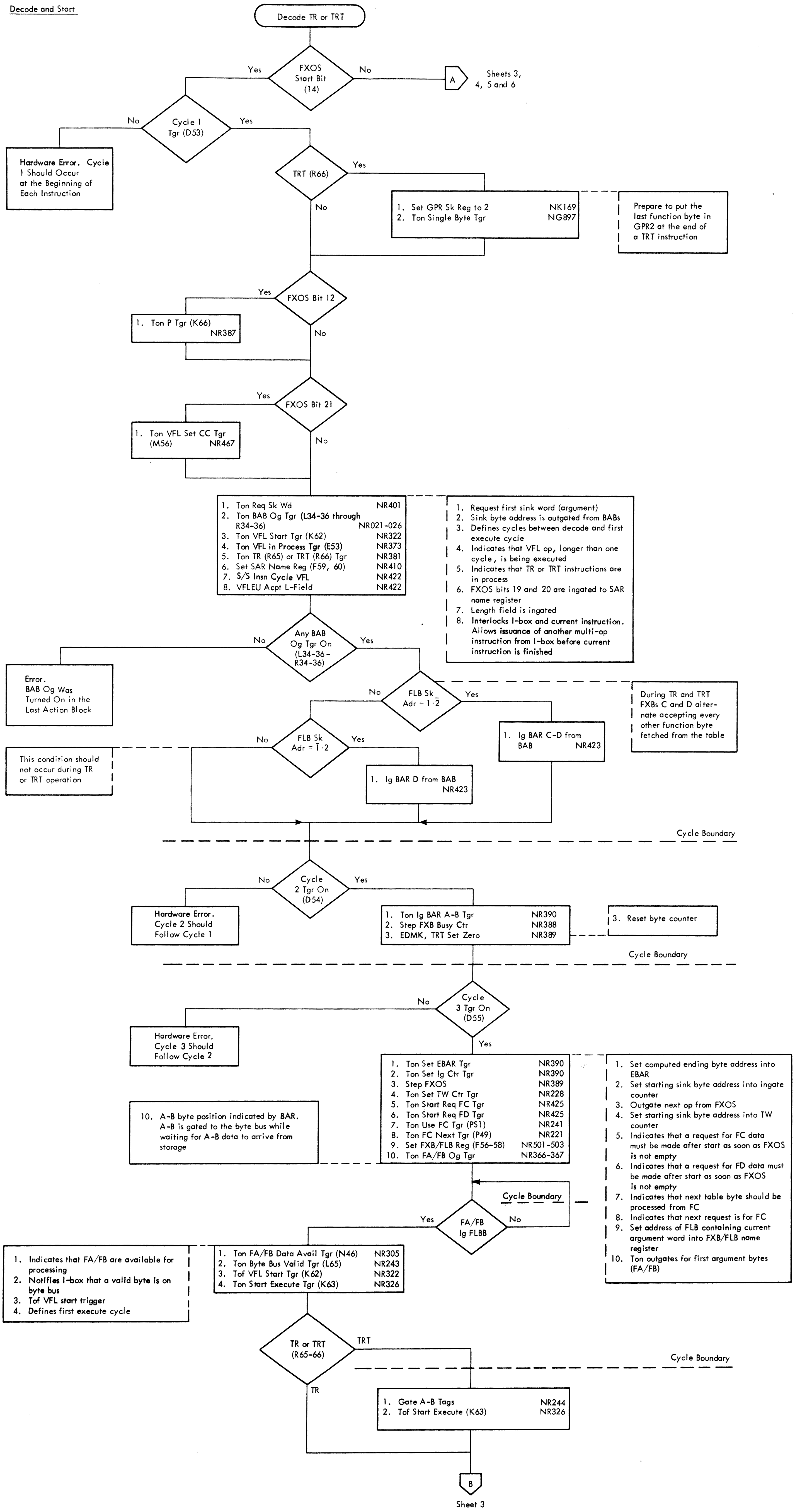

Decode and Start
Decode TR or TRT
FXOS Start Bit (14)
Yes
No
A
Sheets 3, 4, 5 and 6
Cycle 1 Tgr (D53)
No
Yes
Hardware Error. Cycle 1 Should Occur at the Beginning of Each Instruction
TRT (R66)
Yes
No
1. Set GPR Sk Reg to 2 NK169
2. Ton Single Byte Tgr NG897
Prepare to put the last function byte in GPR2 at the end of a TRT instruction
FXOS Bit 12
Yes
No
1. Ton P Tgr (K66) NR387
FXOS Bit 21
Yes
No
1. Ton VFL Set CC Tgr (M56) NR467
1. Ton Req Sk Wd NR401
2. Ton BAB Og Tgr (L34-36 through R34-36) NR021-026
3. Ton VFL Start Tgr (K62) NR322
4. Ton VFL in Process Tgr (E53) NR373
5. Ton TR (R65) or TRT (R66) Tgr NR381
6. Set SAR Name Reg (F59, 60) NR410
7. S/S Insn Cycle VFL NR422
8. VFLEU Acpt L-Field NR422
1. Request first sink word (argument)
2. Sink byte address is outgated from BABs
3. Defines cycles between decode and first execute cycle
4. Indicates that VFL op, longer than one cycle, is being executed
5. Indicates that TR or TRT instructions are in process
6. FXOS bits 19 and 20 are ingated to SAR name register
7. Length field is ingated
8. Interlocks I-box and current instruction. Allows issuance of another multi-op instruction from I-box before current instruction is finished
Any BAB Og Tgr On (L34-36 - R34-36)
No
Yes
Error. BAB Og Was Turned On in the Last Action Block
FLB Sk Adr = 1·2̄
No
Yes
During TR and TRT FXBs C and D alternate accepting every other function byte fetched from the table
1. Ig BAR C-D from BAB NR423
FLB Sk Adr = 1̄·2
No
Yes
This condition should not occur during TR or TRT operation
1. Ig BAR D from BAB NR423
Cycle Boundary
Cycle 2 Tgr On (D54)
No
Yes
Hardware Error. Cycle 2 Should Follow Cycle 1
1. Ton Ig BAR A-B Tgr NR390
2. Step FXB Busy Ctr NR388
3. EDMK, TRT Set Zero NR389
3. Reset byte counter
Cycle Boundary
Cycle 3 Tgr On (D55)
No
Yes
Hardware Error. Cycle 3 Should Follow Cycle 2
1. Ton Set EBAR Tgr NR390
2. Ton Set Ig Ctr Tgr NR390
3. Step FXOS NR389
4. Ton Set TW Ctr Tgr NR228
5. Ton Start Req FC Tgr NR425
6. Ton Start Req FD Tgr NR425
7. Ton Use FC Tgr (PS1) NR241
8. Ton FC Next Tgr (P49) NR221
9. Set FXB/FLB Reg (F56-58) NR501-503
10. Ton FA/FB Og Tgr NR366-367
10. A-B byte position indicated by BAR. A-B is gated to the byte bus while waiting for A-B data to arrive from storage
1. Set computed ending byte address into EBAR
2. Set starting sink byte address into ingate counter
3. Outgate next op from FXOS
4. Set starting sink byte address into TW counter
5. Indicates that a request for FC data must be made after start as soon as FXOS is not empty
6. Indicates that a request for FD data must be made after start as soon as FXOS is not empty
7. Indicates that next table byte should be processed from FC
8. Indicates that next request is for FC
9. Set address of FLB containing current argument word into FXB/FLB name register
10. Ton outgates for first argument bytes (FA/FB)
Cycle Boundary
FA/FB Ig FLBB
Yes
No
1. Ton FA/FB Data Avail Tgr (N46) NR305
2. Ton Byte Bus Valid Tgr (L65) NR243
3. Tof VFL Start Tgr (K62) NR322
4. Ton Start Execute Tgr (K63) NR326
1. Indicates that FA/FB are available for processing
2. Notifies I-box that a valid byte is on byte bus
3. Tof VFL start trigger
4. Defines first execute cycle
TR or TRT (R65-66)
TRT
TR
Cycle Boundary
1. Gate A-B Tags NR244
2. Tof Start Execute (K63) NR326
B
Sheet 3

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 2 OF 8) —

Argument Byte Processing

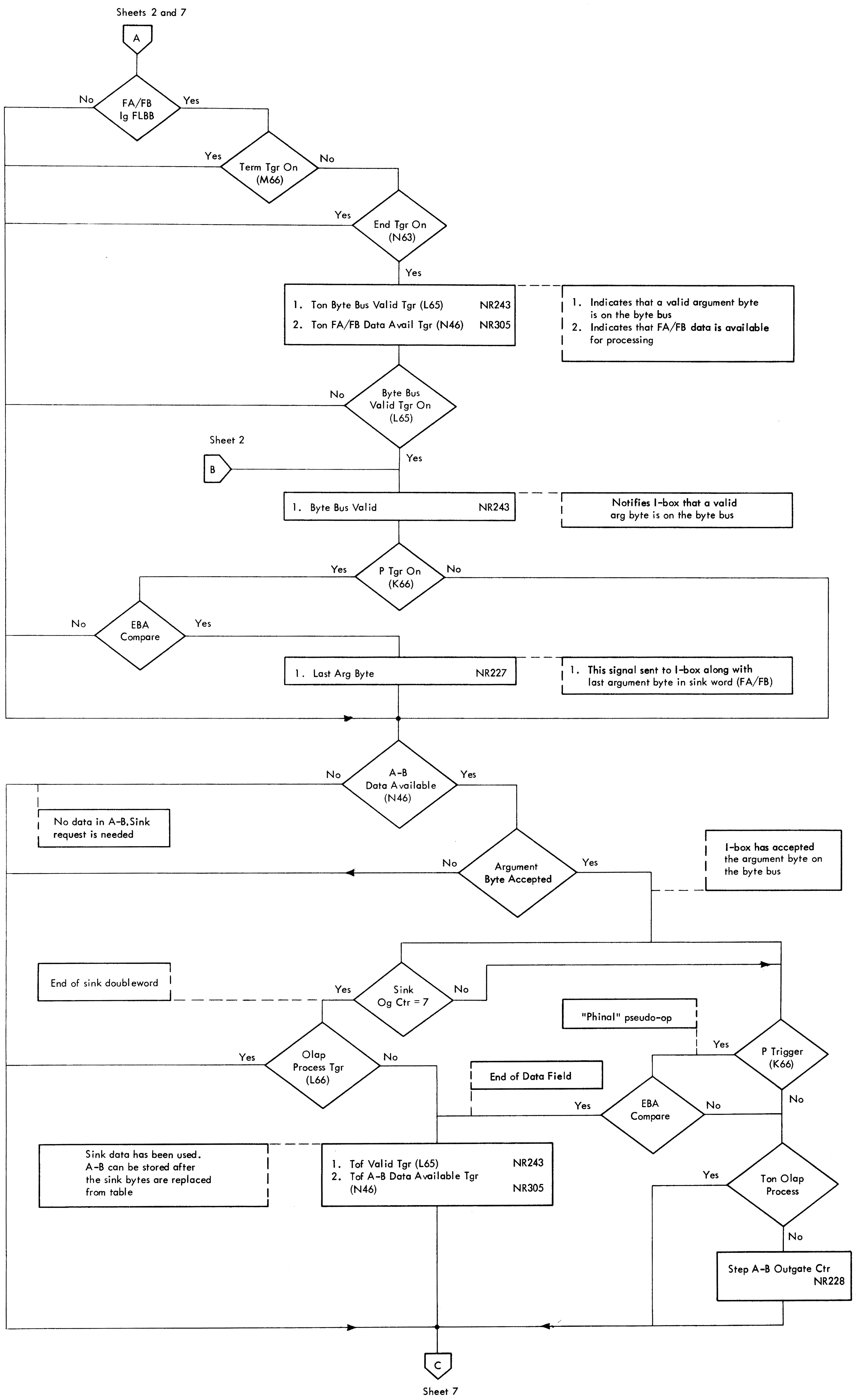
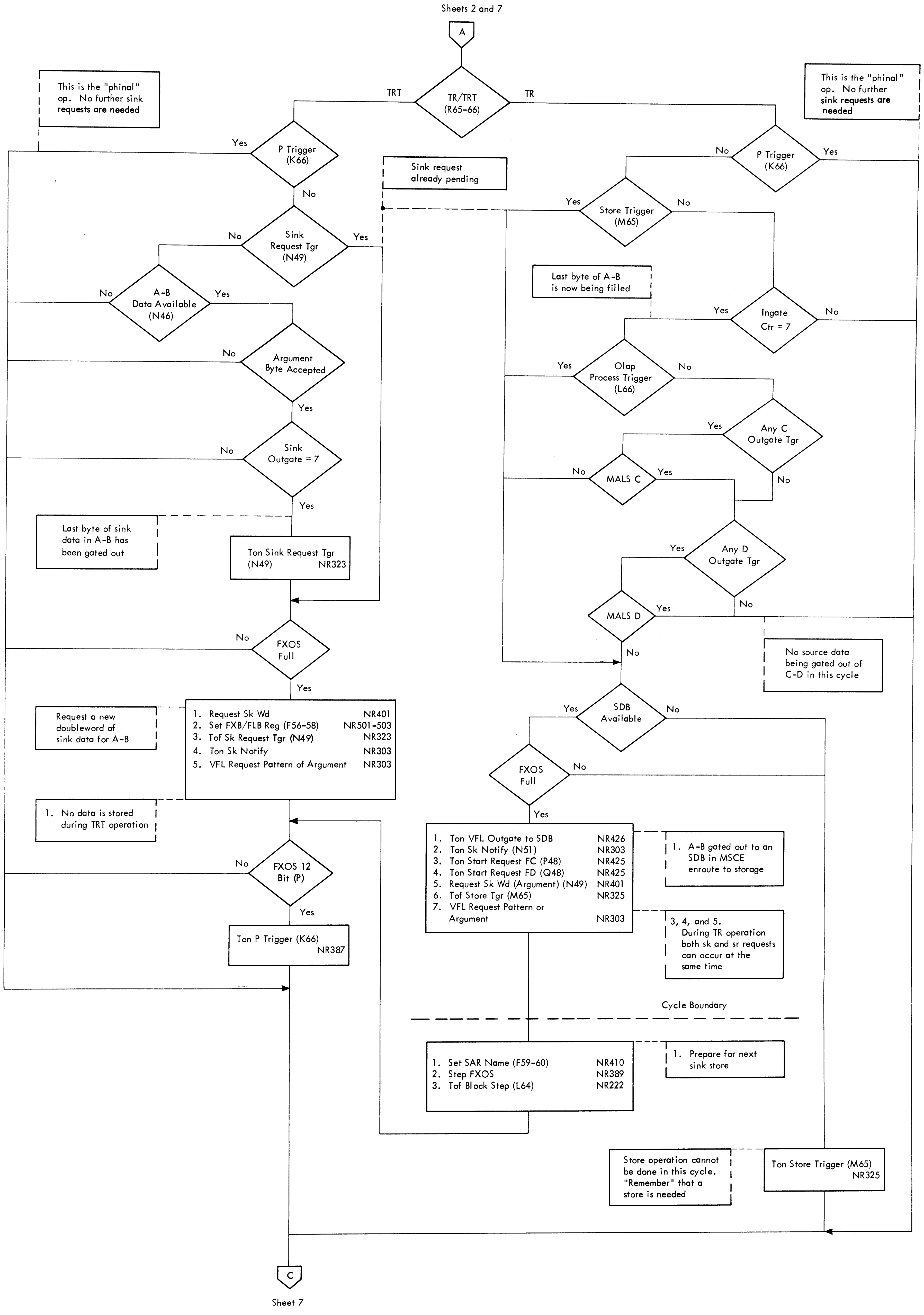

Sheets 2 and 7
A
FA/FB Ig FLBB
No
Yes
Term Tgr On (M66)
Yes
No
End Tgr On (N63)
Yes
Yes
1. Ton Byte Bus Valid Tgr (L65) NR243
2. Ton FA/FB Data Avail Tgr (N46) NR305
1. Indicates that a valid argument byte is on the byte bus
2. Indicates that FA/FB data is available for processing
Byte Bus Valid Tgr On (L65)
No
Yes
Sheet 2
B
1. Byte Bus Valid NR243
Notifies I-box that a valid arg byte is on the byte bus
P Tgr On (K66)
Yes
No
EBA Compare
No
Yes
1. Last Arg Byte NR227
1. This signal sent to I-box along with last argument byte in sink word (FA/FB)
A-B Data Available (N46)
No
Yes
No data in A-B, Sink request is needed
Argument Byte Accepted
No
Yes
I-box has accepted the argument byte on the byte bus
End of sink doubleword
Sink Og Ctr = 7
Yes
No
"Phinal" pseudo-op
Olap Process Tgr (L66)
Yes
No
P Trigger (K66)
Yes
No
End of Data Field
EBA Compare
Yes
No
Sink data has been used. A-B can be stored after the sink bytes are replaced from table
1. Tof Valid Tgr (L65) NR243
2. Tof A-B Data Available Tgr (N46) NR305
Ton Olap Process
Yes
No
Step A-B Outgate Ctr NR228
C
Sheet 7

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 3 OF 8)

Sink Request



Sheets 2 and 7
A
TR/TRT (R65-66)
TRT
TR
This is the "phinal" op. No further sink requests are needed
P Trigger (K66)
Yes
No
Sink request already pending
Sink Request Tgr (N49)
No
Yes
A-B Data Available (N46)
No
Yes
Argument Byte Accepted
No
Yes
Sink Outgate = 7
No
Yes
Last byte of sink data in A-B has been gated out
Ton Sink Request Tgr (N49) NR323
FXOS Full
No
Yes
Request a new doubleword of sink data for A-B
1. Request Sk Wd NR401
2. Set FXB/FLB Reg (F56-58) NR501-503
3. Tof Sk Request Tgr (N49) NR323
4. Ton Sk Notify NR303
5. VFL Request Pattern of Argument NR303
1. No data is stored during TRT operation
FXOS 12 Bit (P)
No
Yes
Ton P Trigger (K66) NR387
This is the "phinal" op. No further sink requests are needed
P Trigger (K66)
No
Yes
Store Trigger (M65)
Yes
No
Last byte of A-B is now being filled
Ingate Ctr = 7
Yes
No
Olap Process Trigger (L66)
Yes
No
Any C Outgate Tgr
Yes
No
MALS C
No
Yes
Any D Outgate Tgr
Yes
No
MALS D
Yes
No
No source data being gated out of C-D in this cycle
SDB Available
Yes
No
FXOS Full
No
Yes
1. Ton VFL Outgate to SDB NR426
2. Ton Sk Notify (N51) NR303
3. Ton Start Request FC (P48) NR425
4. Ton Start Request FD (Q48) NR425
5. Request Sk Wd (Argument) (N49) NR401
6. Tof Store Tgr (M65) NR325
7. VFL Request Pattern or Argument NR303
1. A-B gated out to an SDB in MSCE enroute to storage
3, 4, and 5. During TR operation both sk and sr requests can occur at the same time
Cycle Boundary
1. Set SAR Name (F59-60) NR410
2. Step FXOS NR389
3. Tof Block Step (L64) NR222
1. Prepare for next sink store
Store operation cannot be done in this cycle. "Remember" that a store is needed
Ton Store Trigger (M65) NR325
C
Sheet 7

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 4 OF 8)

Source Byte Processing

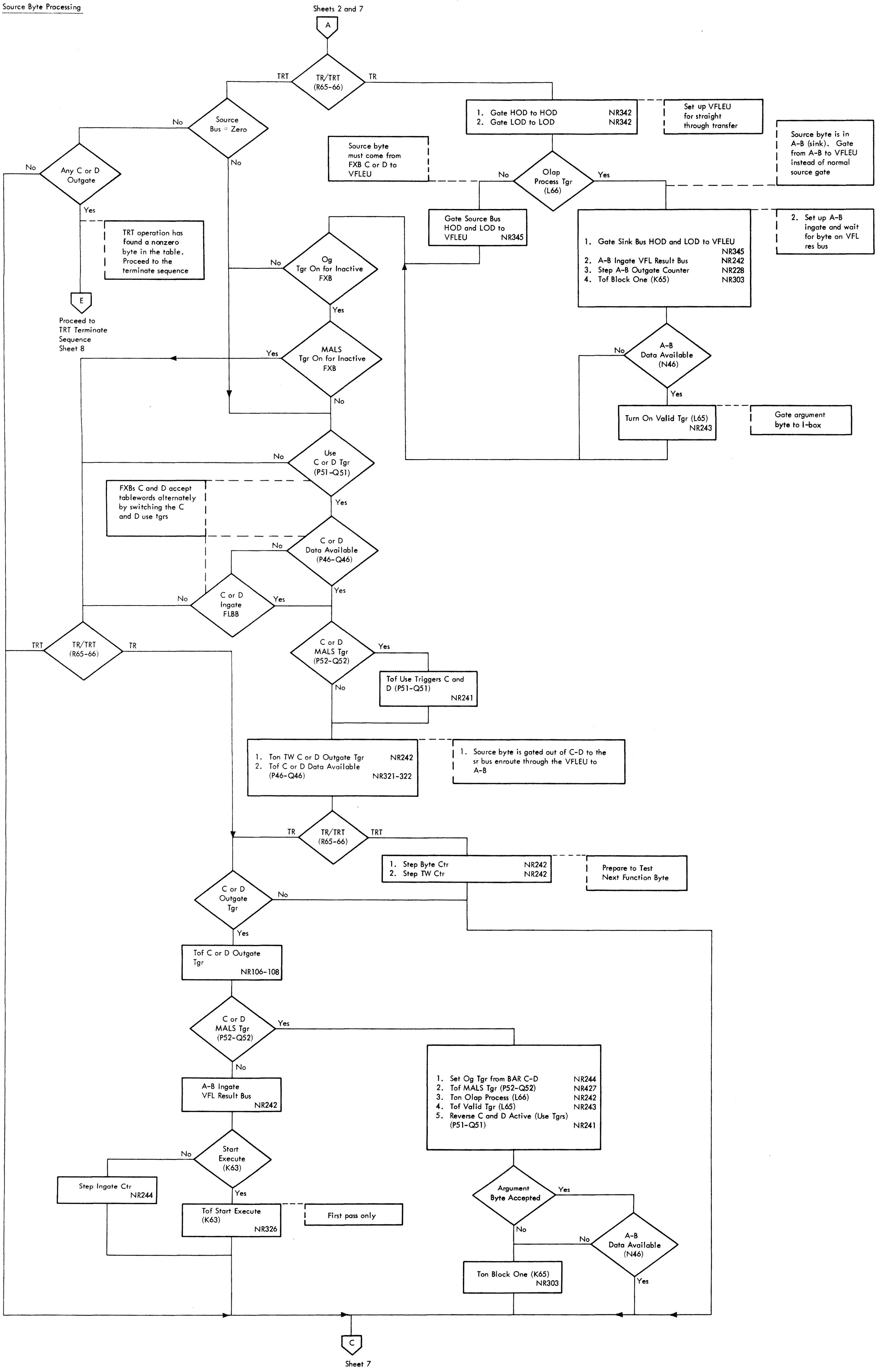

Sheets 2 and 7
A
TR/TRT (R65-66)
TRT
TR
Source Bus = Zero
No
1. Gate HOD to HOD NR342
2. Gate LOD to LOD NR342
Set up VFLEU for straight through transfer
Source byte is in A-B (sink). Gate from A-B to VFLEU instead of normal source gate
Any C or D Outgate
No
Yes
Source byte must come from FXB C or D to VFLEU
Olap Process Tgr (L66)
No
Yes
Gate Source Bus HOD and LOD to VFLEU NR345
1. Gate Sink Bus HOD and LOD to VFLEU NR345
2. A-B Ingate VFL Result Bus NR242
3. Step A-B Outgate Counter NR228
4. Tof Block One (K65) NR303
2. Set up A-B ingate and wait for byte on VFL res bus
TRT operation has found a nonzero byte in the table. Proceed to the terminate sequence
Og Tgr On for Inactive FXB
No
Yes
E
Proceed to TRT Terminate Sequence Sheet 8
MALS Tgr On for Inactive FXB
Yes
No
A-B Data Available (N46)
No
Yes
Turn On Valid Tgr (L65) NR243
Gate argument byte to I-box
Use C or D Tgr (P51-Q51)
No
Yes
FXBs C and D accept tablewords alternately by switching the C and D use tgrs
C or D Data Available (P46-Q46)
No
Yes
C or D Ingate FLBB
No
Yes
TR/TRT (R65-66)
TRT
TR
C or D MALS Tgr (P52-Q52)
Yes
No
Tof Use Triggers C and D (P51-Q51) NR241
1. Ton TW C or D Outgate Tgr NR242
2. Tof C or D Data Available (P46-Q46) NR321-322
1. Source byte is gated out of C-D to the sr bus enroute through the VFLEU to A-B
TR/TRT (R65-66)
TR
TRT
1. Step Byte Ctr NR242
2. Step TW Ctr NR242
Prepare to Test Next Function Byte
C or D Outgate Tgr
No
Yes
Tof C or D Outgate Tgr NR106-108
C or D MALS Tgr (P52-Q52)
Yes
No
1. Set Og Tgr from BAR C-D NR244
2. Tof MALS Tgr (P52-Q52) NR427
3. Ton Olap Process (L66) NR242
4. Tof Valid Tgr (L65) NR243
5. Reverse C and D Active (Use Tgrs) (P51-Q51) NR241
A-B Ingate VFL Result Bus NR242
Start Execute (K63)
No
Yes
Step Ingate Ctr NR244
Tof Start Execute (K63) NR326
First pass only
Argument Byte Accepted
Yes
No
A-B Data Available (N46)
No
Yes
Ton Block One (K65) NR303
C
Sheet 7

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 5 OF 8)

Tableword Request

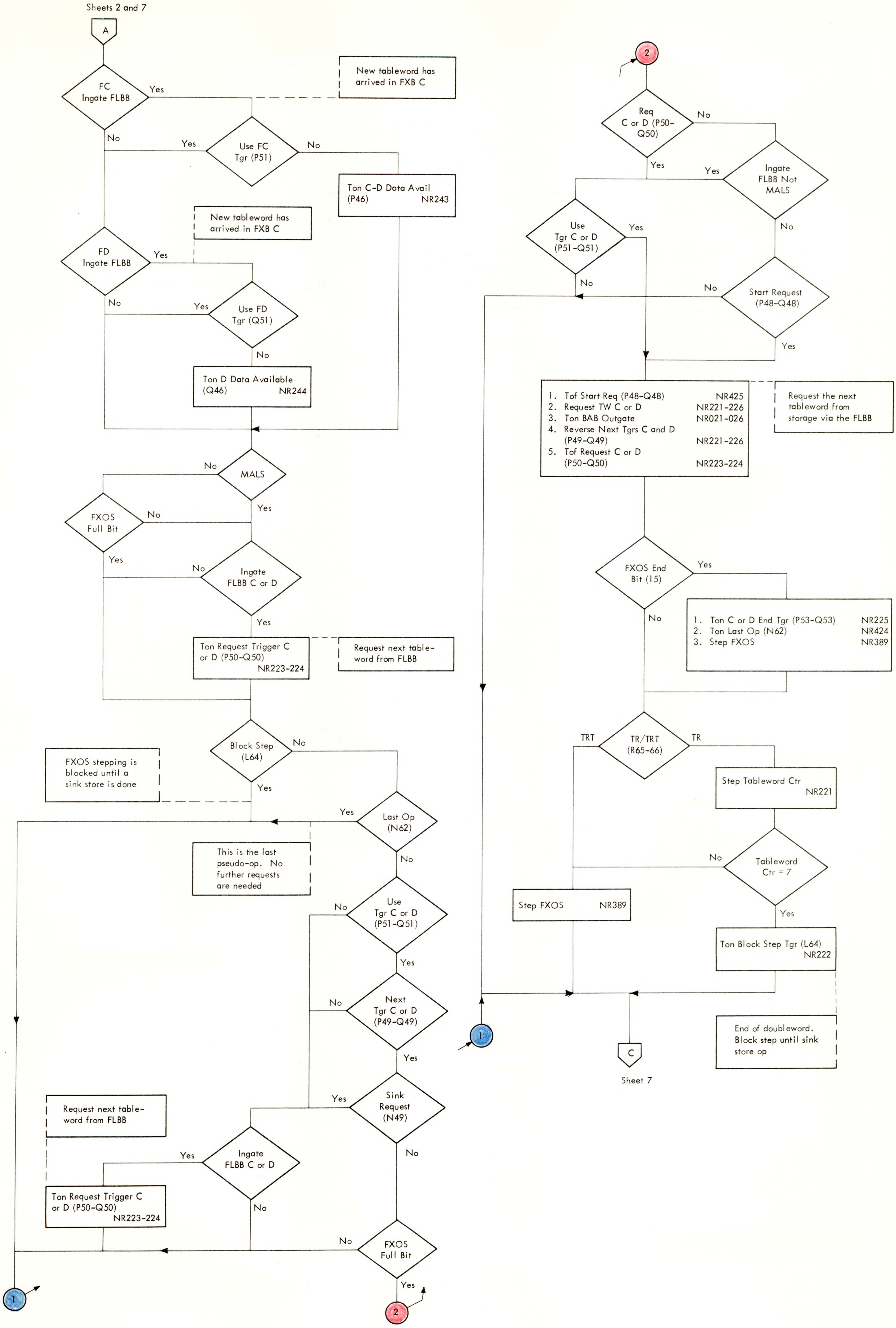

Sheets 2 and 7
A
FC Ingate FLBB
Yes
New tableword has arrived in FXB C
No
Yes
Use FC Tgr (P51)
No
Ton C-D Data Avail (P46) NR243
New tableword has arrived in FXB C
FD Ingate FLBB
Yes
No
Yes
Use FD Tgr (Q51)
No
Ton D Data Available (Q46) NR244
No
MALS
Yes
FXOS Full Bit
No
Yes
No
Ingate FLBB C or D
Yes
Ton Request Trigger C or D (P50-Q50) NR223-224
Request next tableword from FLBB
Block Step (L64)
No
FXOS stepping is blocked until a sink store is done
Yes
Yes
Last Op (N62)
This is the last pseudo-op. No further requests are needed
No
No
Use Tgr C or D (P51-Q51)
Yes
No
Next Tgr C or D (P49-Q49)
Yes
Yes
Sink Request (N49)
Request next tableword from FLBB
Yes
Ingate FLBB C or D
No
Ton Request Trigger C or D (P50-Q50) NR223-224
No
No
FXOS Full Bit
Yes
1
2
2
Req C or D (P50-Q50)
No
Yes
Yes
Ingate FLBB Not MALS
Use Tgr C or D (P51-Q51)
Yes
No
No
No
Start Request (P48-Q48)
Yes
1. Tof Start Req (P48-Q48) NR425
2. Request TW C or D NR221-226
3. Ton BAB Outgate NR021-026
4. Reverse Next Tgrs C and D (P49-Q49) NR221-226
5. Tof Request C or D (P50-Q50) NR223-224
Request the next tableword from storage via the FLBB
FXOS End Bit (15)
Yes
1. Ton C or D End Tgr (P53-Q53) NR225
2. Ton Last Op (N62) NR424
3. Step FXOS NR389
No
TRT
TR/TRT (R65-66)
TR
Step Tableword Ctr NR221
No
Tableword Ctr = 7
Step FXOS NR389
Yes
Ton Block Step Tgr (L64) NR222
End of doubleword. Block step until sink store op
1
C
Sheet 7

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 6 OF 8)

Ending Sequence

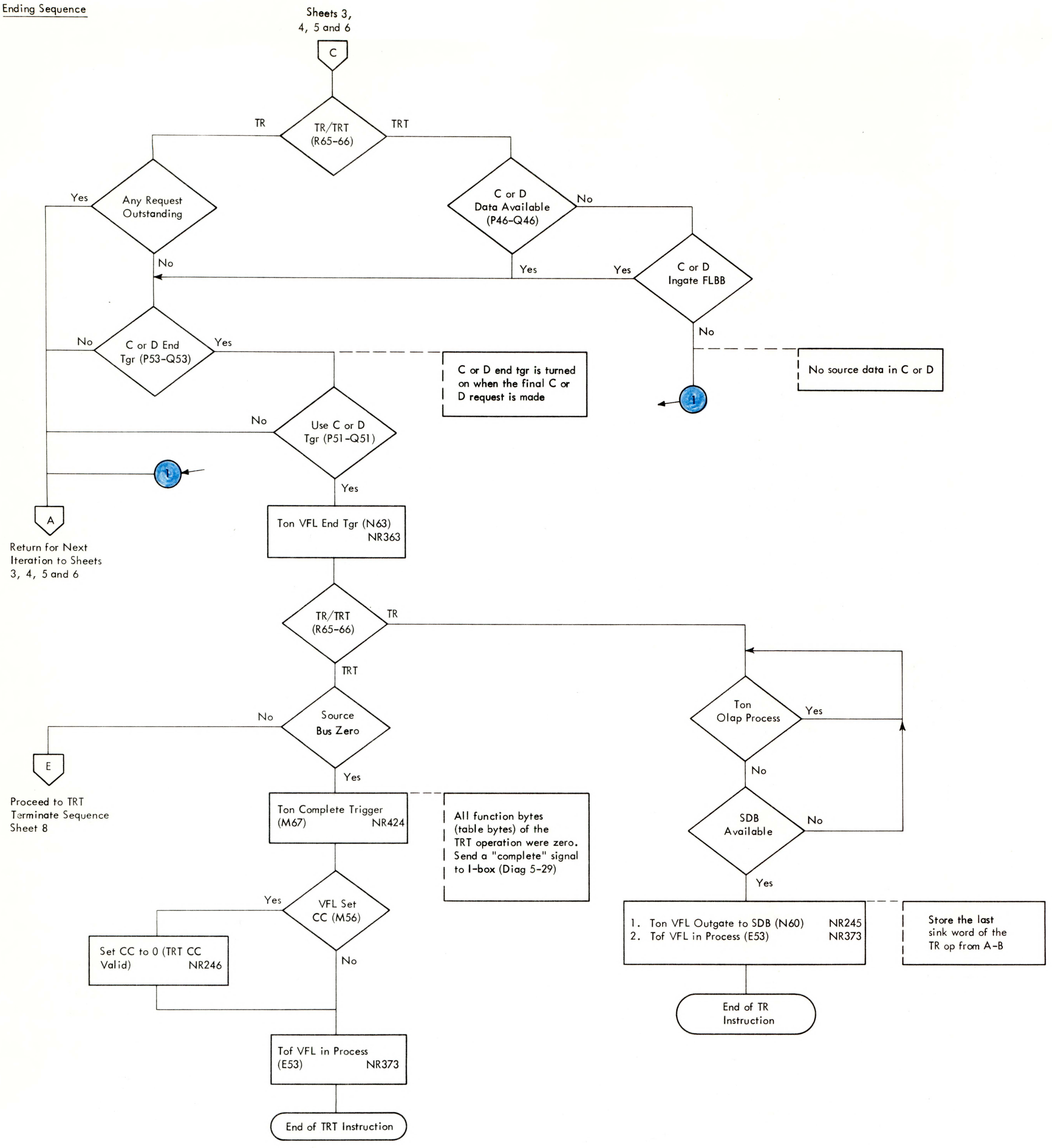

Sheets 3, 4, 5 and 6
C
TR/TRT (R65-66)
TR
TRT
Any Request Outstanding
Yes
No
C or D Data Available (P46-Q46)
No
Yes
C or D Ingate FLBB
Yes
No
No source data in C or D
C or D End Tgr (P53-Q53)
No
Yes
C or D end tgr is turned on when the final C or D request is made
Use C or D Tgr (P51-Q51)
No
Yes
1
1
A
Return for Next Iteration to Sheets 3, 4, 5 and 6
Ton VFL End Tgr (N63) NR363
TR/TRT (R65-66)
TR
TRT
Source Bus Zero
No
Yes
E
Proceed to TRT Terminate Sequence Sheet 8
Ton Complete Trigger (M67) NR424
All function bytes (table bytes) of the TRT operation were zero. Send a "complete" signal to I-box (Diag 5-29)
VFL Set CC (M56)
Yes
No
Set CC to 0 (TRT CC Valid) NR246
Tof VFL in Process (E53) NR373
End of TRT Instruction
Ton Olap Process
Yes
No
SDB Available
No
Yes
1. Ton VFL Outgate to SDB (N60) NR245
2. Tof VFL in Process (E53) NR373
Store the last sink word of the TR op from A-B
End of TR Instruction

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 7 OF 8)

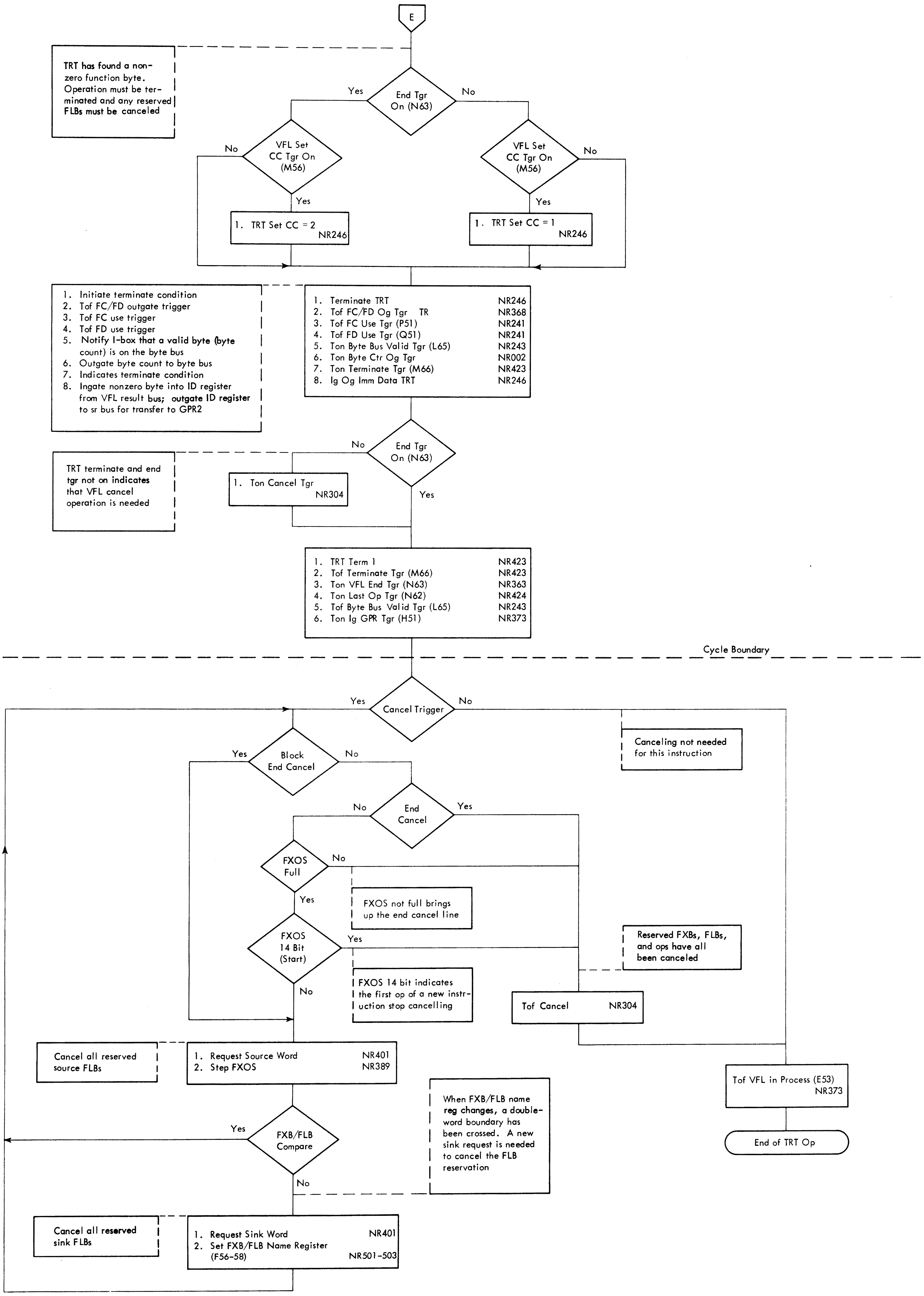
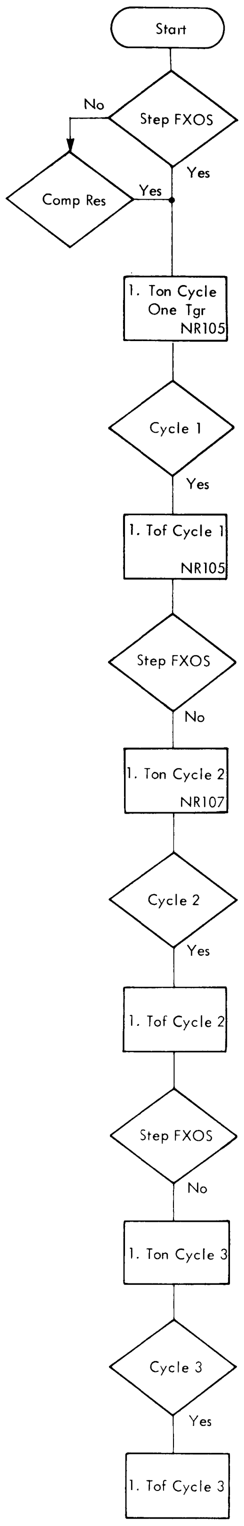

E
TRT has found a nonzero function byte. Operation must be terminated and any reserved FLBs must be canceled
End Tgr On (N63)
Yes
No
VFL Set CC Tgr On (M56)
No
Yes
1. TRT Set CC = 2 NR246
VFL Set CC Tgr On (M56)
No
Yes
1. TRT Set CC = 1 NR246
1. Initiate terminate condition
2. Tof FC/FD outgate trigger
3. Tof FC use trigger
4. Tof FD use trigger
5. Notify I-box that a valid byte (byte count) is on the byte bus
6. Outgate byte count to byte bus
7. Indicates terminate condition
8. Ingate nonzero byte into ID register from VFL result bus; outgate ID register to sr bus for transfer to GPR2
1. Terminate TRT NR246
2. Tof FC/FD Og Tgr TR NR368
3. Tof FC Use Tgr (P51) NR241
4. Tof FD Use Tgr (Q51) NR241
5. Ton Byte Bus Valid Tgr (L65) NR243
6. Ton Byte Ctr Og Tgr NR002
7. Ton Terminate Tgr (M66) NR423
8. Ig Og Imm Data TRT NR246
TRT terminate and end tgr not on indicates that VFL cancel operation is needed
End Tgr On (N63)
No
Yes
1. Ton Cancel Tgr NR304
1. TRT Term 1 NR423
2. Tof Terminate Tgr (M66) NR423
3. Ton VFL End Tgr (N63) NR363
4. Ton Last Op Tgr (N62) NR424
5. Tof Byte Bus Valid Tgr (L65) NR243
6. Ton Ig GPR Tgr (H51) NR373
Cycle Boundary
Cancel Trigger
Yes
No
Canceling not needed for this instruction
Block End Cancel
Yes
No
End Cancel
No
Yes
FXOS Full
No
Yes
FXOS not full brings up the end cancel line
FXOS 14 Bit (Start)
Yes
No
FXOS 14 bit indicates the first op of a new instruction stop cancelling
Reserved FXBs, FLBs, and ops have all been canceled
Tof Cancel NR304
Tof VFL in Process (E53) NR373
End of TRT Op
Cancel all reserved source FLBs
1. Request Source Word NR401
2. Step FXOS NR389
FXB/FLB Compare
Yes
No
When FXB/FLB name reg changes, a double-word boundary has been crossed. A new sink request is needed to cancel the FLB reservation
Cancel all reserved sink FLBs
1. Request Sink Word NR401
2. Set FXB/FLB Name Register (F56-58) NR501-503

DIAGRAM 5-126. TR AND TRT INSTRUCTION (SHEET 8 OF 8)

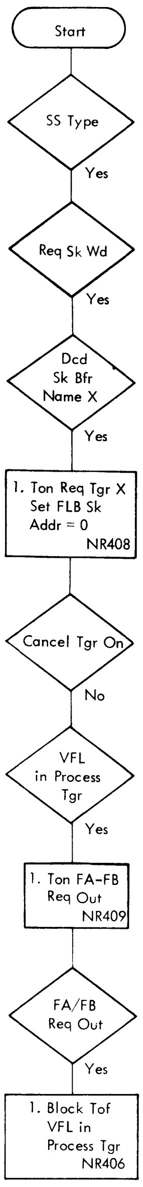
Objectives:

1. FA/FB Request sequence is executed whenever a sink word request is generated.
2. FC/FD Request sequence is executed whenever a source request or TWC/TWD (TRT instruction only) request is generated.
3. Cycle Control sequence is executed each time FXOS is stepped.
4. SDB Available sequence is used to determine which SDB is available for receiving data from the VFL area.
5. VFL Reset sequence is executed to reset various VFL control triggers.
6. Block VFL sequence prohibits VFL operations when certain conditions exist.

Cycle Control

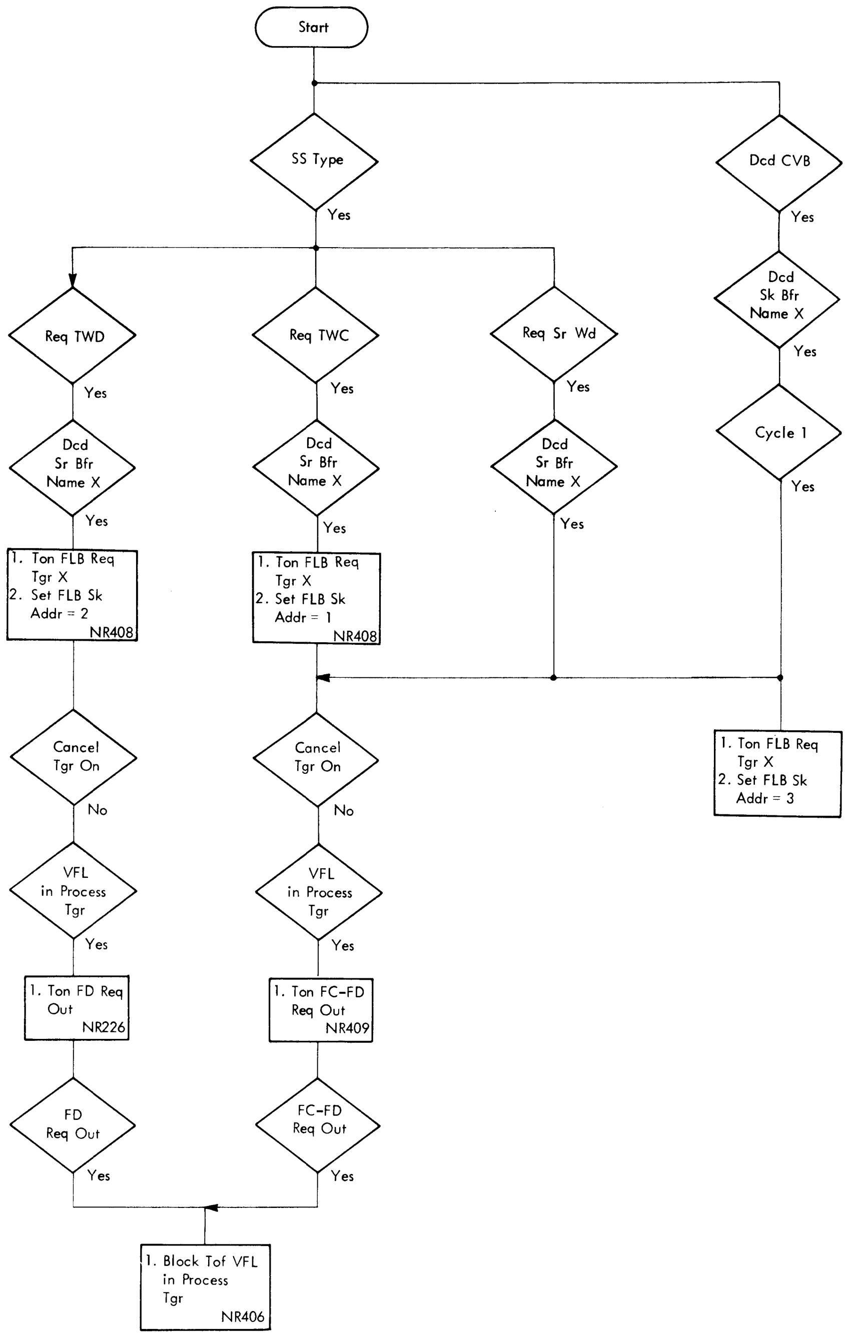


FA/FB Request

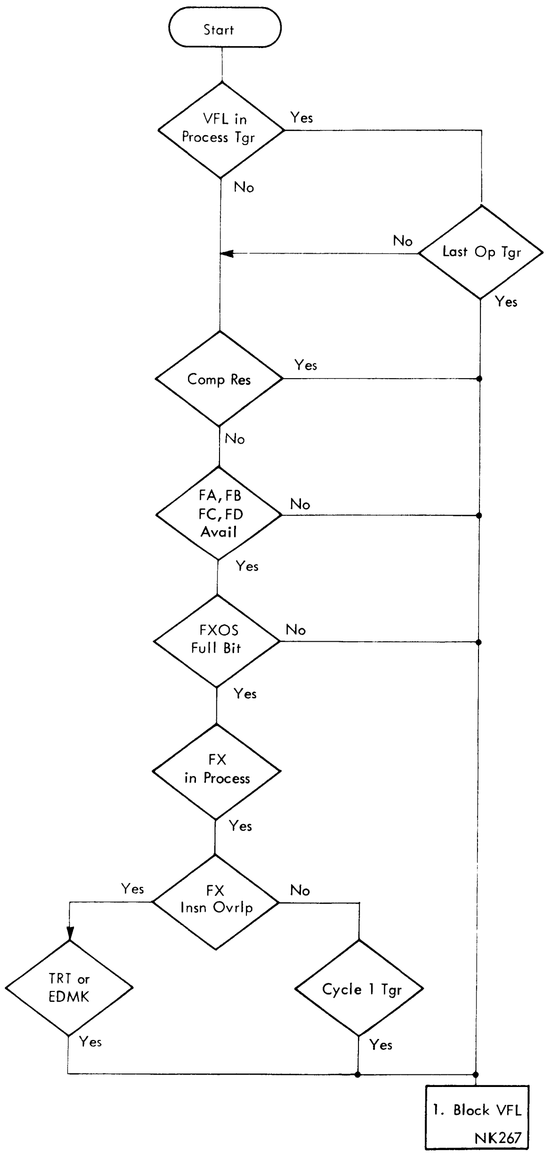


Start
SS Type
Yes
Req Sk Wd
Yes
Dcd Sk Bfr Name X
Yes
1. Ton Req Tgr X Set FLB Sk Addr = 0 NR408
Cancel Tgr On
No
VFL in Process Tgr
Yes
1. Ton FA-FB Req Out NR409
FA/FB Req Out
Yes
1. Block Tof VFL in Process Tgr NR406

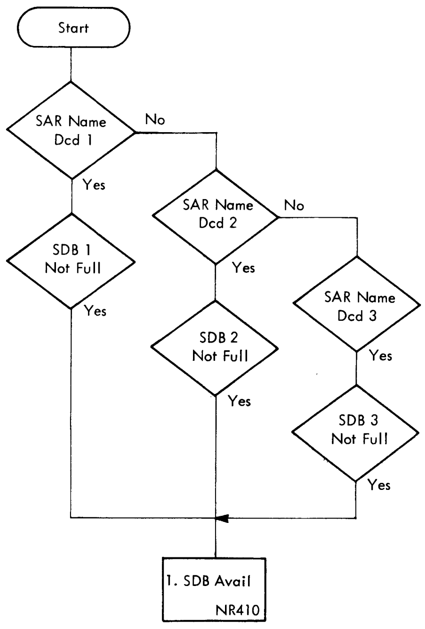
FC/FD Request



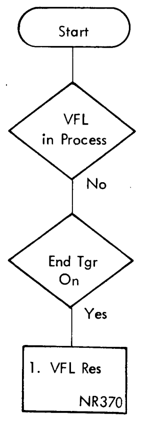
Block VFL



SDB Available



VFL Reset



DIAGPAM 5-127. VFL CONTROLS

Objectives:

1. As each operation is decoded from the op stack, a sequence of conditions must be checked to see if the operation will proceed or be blocked.
2. If no interfering conditions exist, the operation can proceed with decoding and processing.
3. If any block condition exists, the block op trigger is turned on and the operation must wait until the block condition is cleared before proceeding.

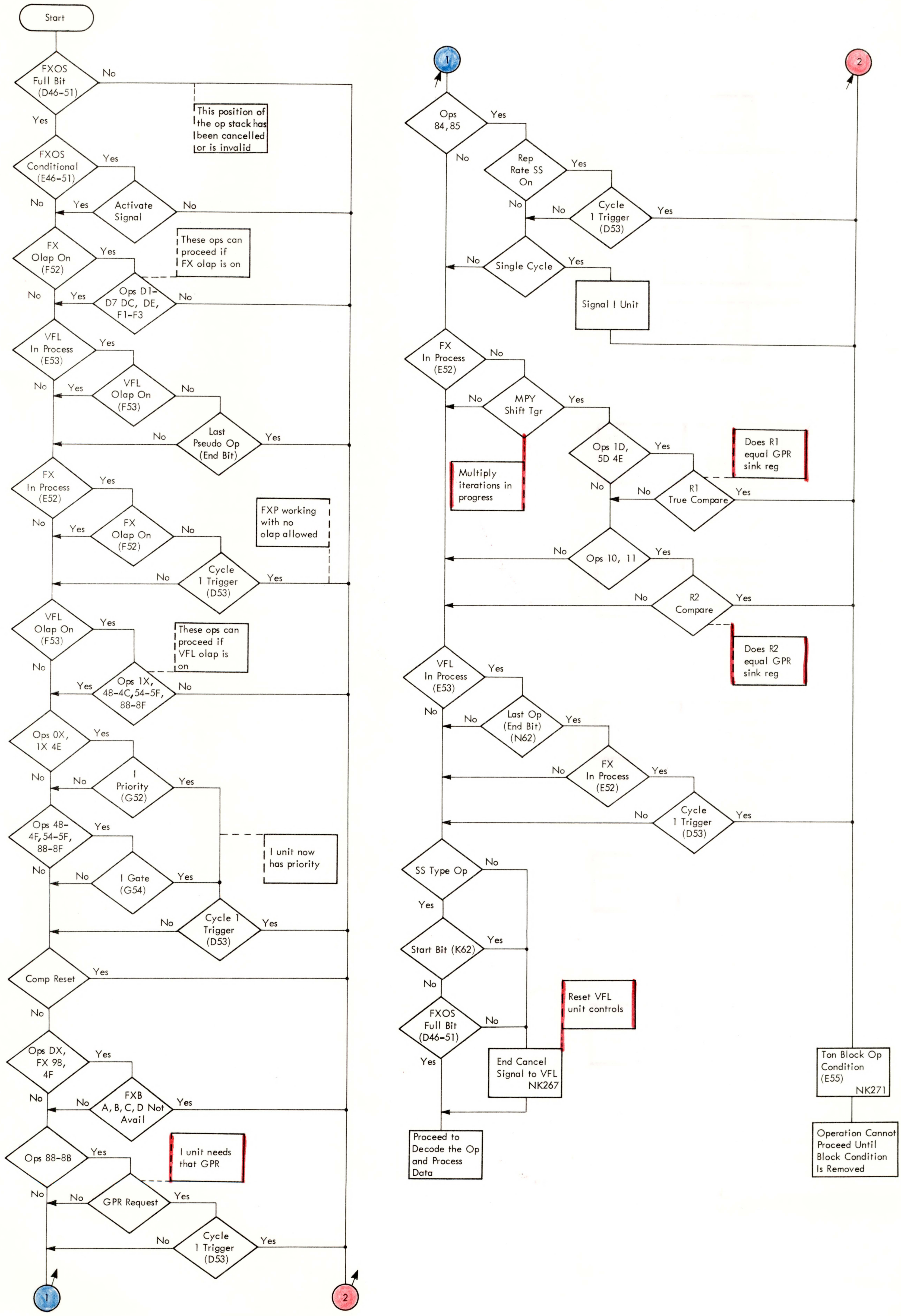


DIAGRAM 5-128. BLOCK OPERATION

Cancelled Operation Processing

Objectives:

1. A pseudo op in the FXOS can be cancelled by resetting the corresponding FXOS full bit.
2. When an operation is cancelled, all circuits set up for that operation must be restored or reset.
3. Signals are sent to I unit that all FXB's and GPR's reserved for the cancelled operation will no longer be needed by that operation.
4. FXOS steps to decode the next pseudo-op.

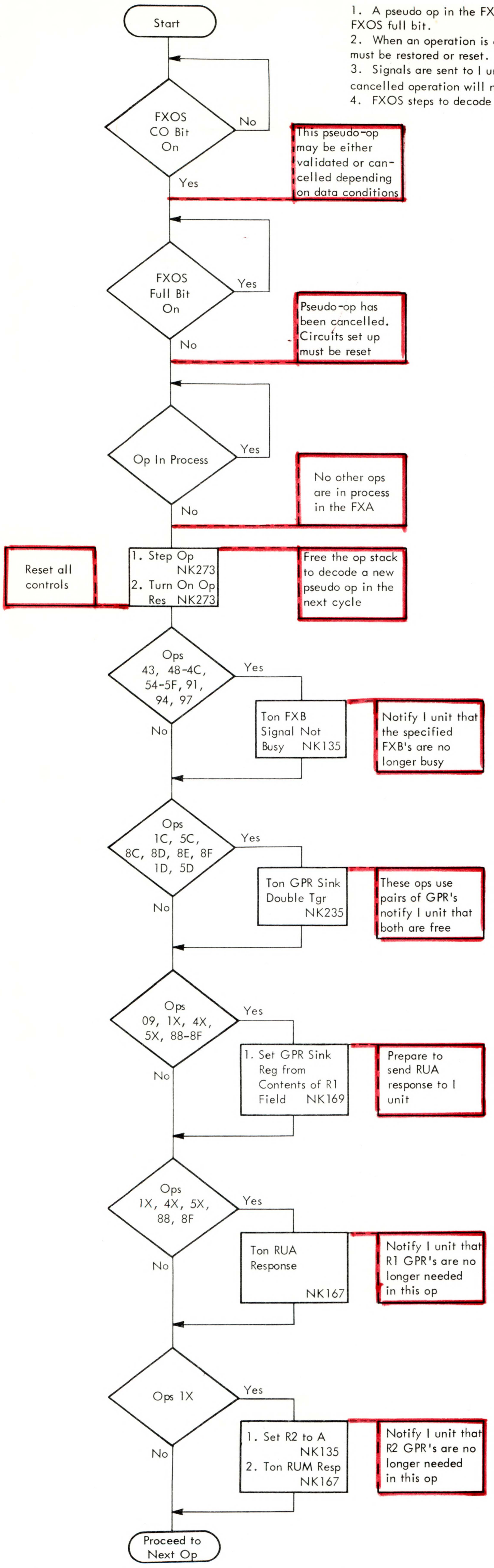
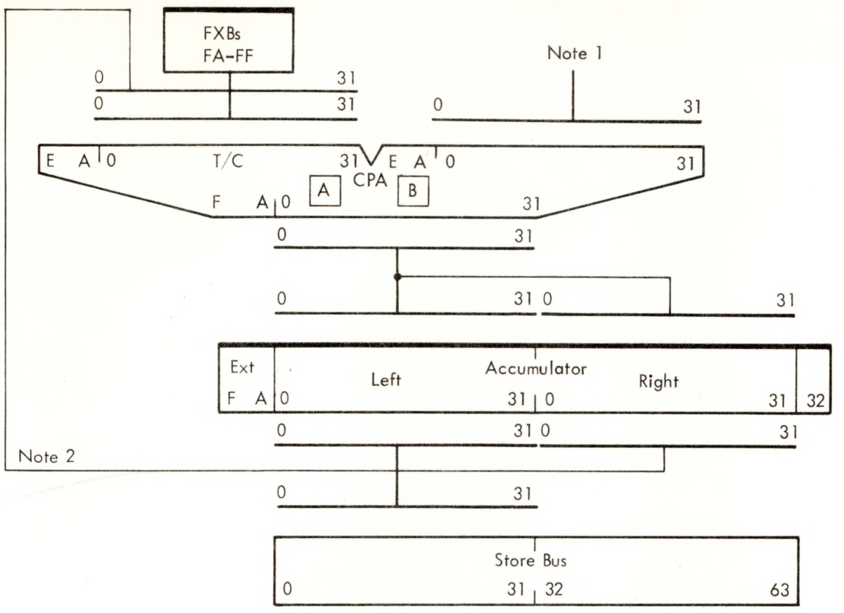
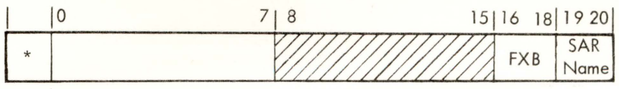

Start
FXOS CO Bit On
No
Yes
This pseudo-op may be either validated or cancelled depending on data conditions
FXOS Full Bit On
Yes
No
Pseudo-op has been cancelled. Circuits set up must be reset
Op In Process
Yes
No
No other ops are in process in the FXA
Reset all controls
1. Step Op NK273
2. Turn On Op Res NK273
Free the op stack to decode a new pseudo op in the next cycle
Ops 43, 48-4C, 54-5F, 91, 94, 97
Yes
No
Ton FXB Signal Not Busy NK135
Notify I unit that the specified FXB's are no longer busy
Ops 1C, 5C, 8C, 8D, 8E, 8F 1D, 5D
Yes
No
Ton GPR Sink Double Tgr NK235
These ops use pairs of GPR's notify I unit that both are free
Ops 09, 1X, 4X, 5X, 88-8F
Yes
No
1. Set GPR Sink Reg from Contents of R1 Field NK169
Prepare to send RUA response to I unit
Ops 1X, 4X, 5X, 88, 8F
Yes
No
Ton RUA Response NK167
Notify I unit that R1 GPR's are no longer needed in this op
Ops 1X
Yes
No
1. Set R2 to A NK135
2. Ton RUM Resp NK167
Notify I unit that R2 GPR's are no longer needed in this op
Proceed to Next Op

DIAGRAM 5-129. CANCELLED OPERATION PROCESSING



Timer Operation



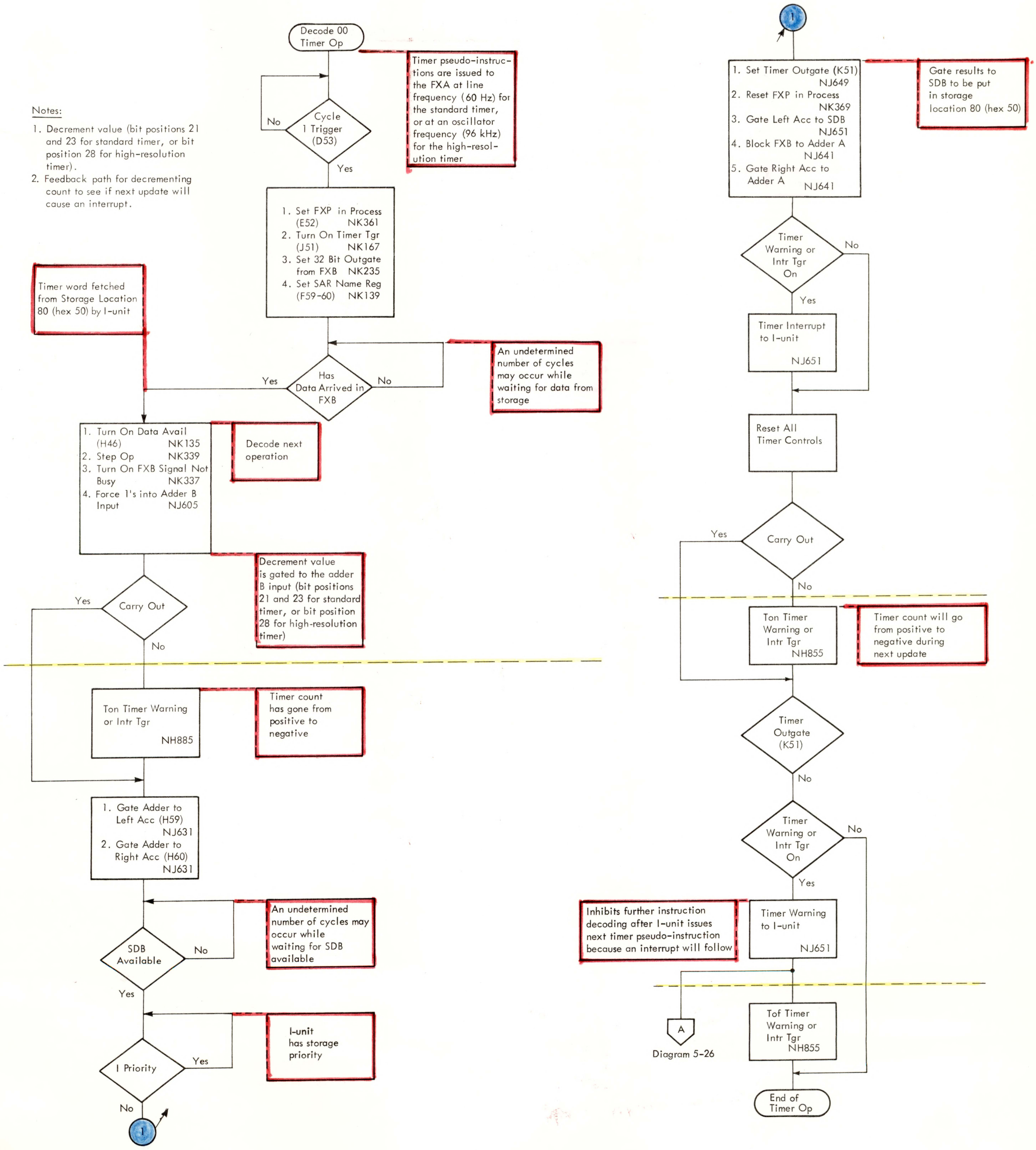
* Special RR Format

Objectives:

1. Timer operation fetches a timer count word from storage location 80 (hexadecimal 50), decrements the count by a specified amount, and stores the count word back in the same location.
2. Timer pseudo instructions are issued to the FXA at a rate of 60 per second (every 16.7 ms) for the standard timer, or at a rate of 9,600 per second (every 104 usec) for the high-resolution timer.
3. When the decremented count becomes negative (has passed through zero), a timer interrupt occurs to signal the program that a particular period of time has passed.
4. When the FXA detects that the next decrement will cause the count to become negative (pass through zero), it sends a timer warning signal to the I-unit to instruction decoding after the I-unit issues the next timer pseudo-op.

Notes:

1. Decrement value (bit positions 21 and 23 for standard timer, or bit position 28 for high-resolution timer).
2. Feedback path for decrementing count to see if next update will cause an interrupt.



Decode 00
Timer Op
Timer pseudo-instructions are issued to the FXA at line frequency (60 Hz) for the standard timer, or at an oscillator frequency (96 kHz) for the high-resolution timer
No
Cycle
1 Trigger
(D53)
Yes
1. Set FXP in Process (E52) NK361
2. Turn On Timer Tgr (J51) NK167
3. Set 32 Bit Outgate from FXB NK235
4. Set SAR Name Reg (F59-60) NK139
Timer word fetched from Storage Location 80 (hex 50) by I-unit
Yes
Has Data Arrived in FXB
No
An undetermined number of cycles may occur while waiting for data from storage
1. Turn On Data Avail (H46) NK135
2. Step Op NK339
3. Turn On FXB Signal Not Busy NK337
4. Force 1's into Adder B Input NJ605
Decode next operation
Decrement value is gated to the adder B input (bit positions 21 and 23 for standard timer, or bit position 28 for high-resolution timer)
Yes
Carry Out
No
Ton Timer Warning or Intr Tgr
NH885
Timer count has gone from positive to negative
1. Gate Adder to Left Acc (H59) NJ631
2. Gate Adder to Right Acc (H60) NJ631
An undetermined number of cycles may occur while waiting for SDB available
SDB Available
No
Yes
I-unit has storage priority
I Priority
Yes
No
1
1
1. Set Timer Outgate (K51) NJ649
2. Reset FXP in Process NK369
3. Gate Left Acc to SDB NJ651
4. Block FXB to Adder A NJ641
5. Gate Right Acc to Adder A NJ641
Gate results to SDB to be put in storage location 80 (hex 50)
Timer Warning or Intr Tgr On
No
Yes
Timer Interrupt to I-unit
NJ651
Reset All Timer Controls
Yes
Carry Out
No
Ton Timer Warning or Intr Tgr
NH855
Timer count will go from positive to negative during next update
Timer Outgate (K51)
No
Timer Warning or Intr Tgr On
No
Yes
Inhibits further instruction decoding after I-unit issues next timer pseudo-instruction because an interrupt will follow
Timer Warning to I-unit
NJ651
A
Diagram 5-26
Tof Timer Warning or Intr Tgr
NH855
End of Timer Op

DIAGRAM 5-130. TIMER OPERATION

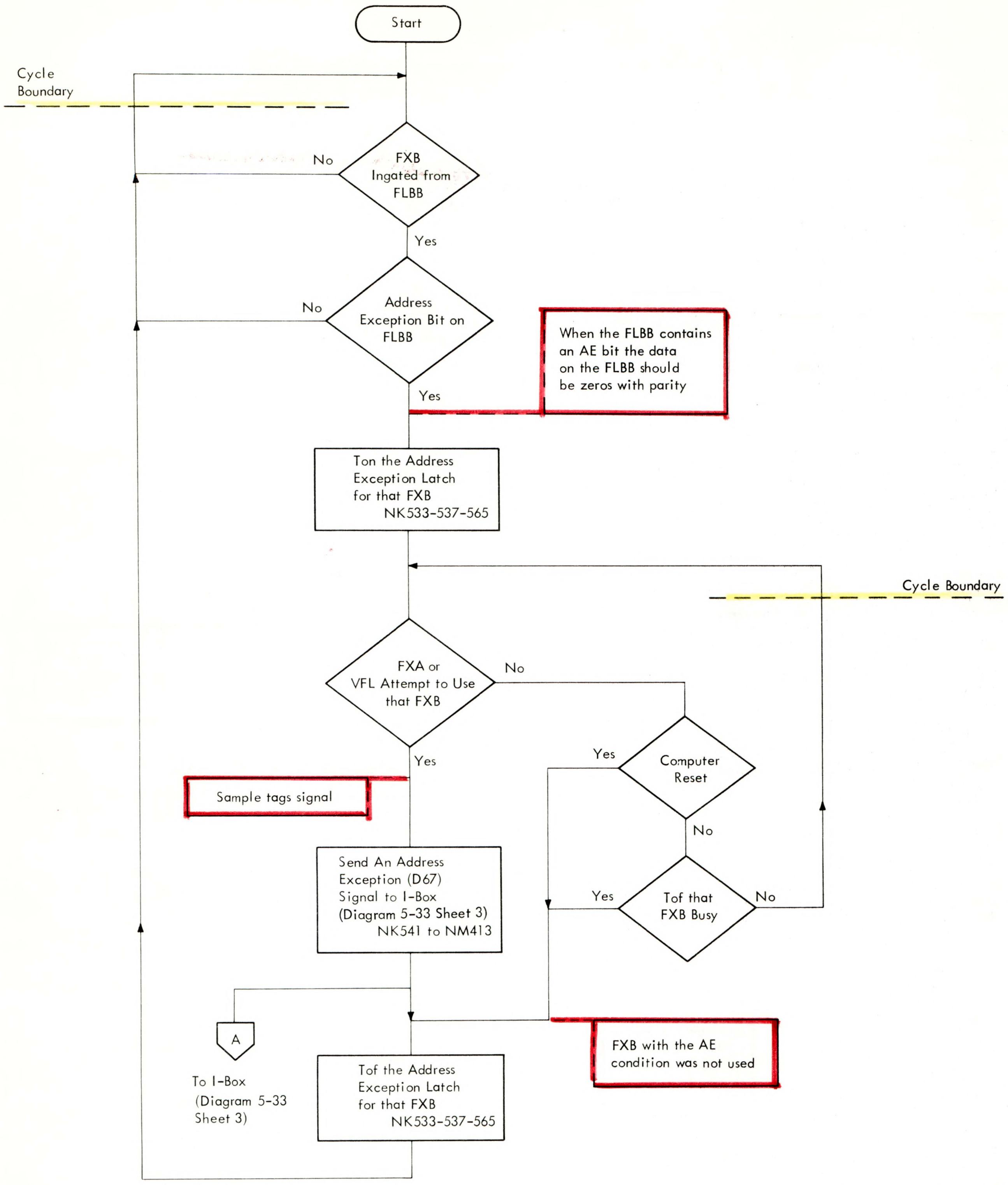

Start
Cycle Boundary
FXB Ingated from FLBB
No
Yes
Address Exception Bit on FLBB
No
Yes
When the FLBB contains an AE bit the data on the FLBB should be zeros with parity
Ton the Address Exception Latch for that FXB NK533-537-565
Cycle Boundary
FXA or VFL Attempt to Use that FXB
No
Yes
Sample tags signal
Computer Reset
Yes
No
Tof that FXB Busy
Yes
No
Send An Address Exception (D67) Signal to I-Box (Diagram 5-33 Sheet 3) NK541 to NM413
A
To I-Box (Diagram 5-33 Sheet 3)
Tof the Address Exception Latch for that FXB NK533-537-565
FXB with the AE condition was not used

DIAGRAM 5-131. ADDRESS EXCEPTION INTERRUPT

A Bus Sign - NH857
Defines the sign of the A bus into the adder.

Abort - NH861
Used to define an abort condition in divide or convert to binary. In divide, quotient has overflowed; in convert to binary, the definition of the decimal digit processed is incorrect.

Add Logical, Subtract Logical - NH853
Defines conditions used to set the condition code on the cycle following the execution cycle.

Add, Subtract, Logical - NK173
Defines execution cycle of AL, ALR, SL, and SLR for CC setting.

Adder Carry Out - NH855
Defines a carry out of bit 0 in the adder.

Adder Output All Zeros - NH859
Set when the adder output is all zeros. Used for condition code setting.

Adder Output Sign - NH863
Defines the adder output sign.

Adder Overflow - NH855
Defines an overflow out of the adder.

Adder to Accumulator, Left - NJ625
Gates the adder output into the left accumulator.

Adder to Accumulator, Right - NJ627
Gates the adder output into the right accumulator.

AE Interrupt - NK541
One cycle trigger to signal an AE interrupt to the I-Box.

AND - NG927
Defines the execution time of N, NR, O, OR.

AND, EXOR - NH851
Defines instructions that set condition code.

BAR AB - NR061, 063
Byte address register AB initially contains the starting byte address of the sink operand, and thereafter holds the encoded value (0-7) of the sink byte position being used. BAR AB is a three-bit register.

BAR B - NR125
Byte address register B is a two-bit register that contains the byte position of FB to be used in execution.

BAR CD - NR103, 105
Byte address register CD initially contains the starting byte address of the source operand, and thereafter holds the encoded value (0-7) of the source byte position being used. It is a four-bit register, with the extra bit being added for the execution of TR.

BAR D - NR126
Byte address register D is a three-position register that contains the byte position of FD to be used in execution.

BAR E2 - NK643
Byte address register, 2 bit of FXB E.

BAR F2 - NK643
Byte address register, 2 bit of FXB F.

B Bus Sign - NH857
Used to define the sign of the B bus input to the adder.

Begun - NJ603
Defines second through last cycle of multicycle ops (i.e., divide, multiply, shift, CVB, CVD, and timer).

Block B Bus - NK543
Blocks the bus from GPRs to CPA B bus.

Block Buffer A Bus - NK545
Blocks the bus from FXBs to CPA A bus.

Block GPR A Bus - NK545
Blocks the bus from GPRs to CPA A bus.

Block One Trigger - NR303
Used to force the ingate counter from counting up or down. In particular, it is used for executing overlapping TR.

Block Step Trigger - NR222
Prevents the FXOS from stepping.

Byte Address Buffers - NR021, 026
The byte address buffers (BABs) are a group of six three-bit buffers. Each BAB is associated with a specific floating buffer, and holds the byte address associated with the data contained in that buffer.

Byte Address Buffer Outgates - NR021, 026
Six triggers, one associated with each byte address buffer. These triggers control the outgating of one of six BABs to the byte address registers.

Byte Counter - NR004, 007
The byte counter is an eight-bit counter used for the execution of TRT and EDMK. It is initially set to zero and then incremented by one every time a result byte is processed.

Byte Counter Outgate - NR002
Outgates the byte counter to the byte bus.

Carry Error Byte 1 - NH773
Records a carry error in byte 1.

Carry Error Byte 2 - NH775
Records a carry error in byte 2.

Carry Error Byte 3 - NH777
Records a carry error in byte 3.

Carry Error Byte 4 - NH779
Records a carry error in byte 4.

CC Add Group 1 - NK171
Defines execution cycle of LPR, LNR, LTR, LCR, A, AH, AR, S, SH, and SR for CC setting.

CC Overflow Group 1 - NK169
Defines execution cycle of LPR, LCR, A, AH, AR, S, SH, and SR for CC setting.

CLI - NK413
Defines compare immediate execution time.

Compare Arithmetic - NH853
Defines instruction used to set condition code on the cycle following the execution cycle.

Compare Logical - NH851
Defines instruction to set condition code on the cycle following the execution cycle.

Compare Arithmetic - NK171
Defines execution cycle of C, CH, and CR for CC setting.

Complement - NH203
Used to complement the A input of the adder.

Complement if Feedback Sign Negative - NH201
Used to complement the A input of the adder if the feedback from the adder on the previous cycle was negative.

Complement if Feedback Sign Plus - NH201
Used to complement the A adder input if the feedback from the adder on the previous cycle was positive.

Compare Logical - NK173
Defines execution cycle of CL and CLR for CC setting.

Complete - NR424
Notifies the I-Box of completion as opposed to termination of a VFL instruction.

Conditional Overflow - NH867
Used in divide and CVB to remember a conditional overflow whose ambiguity will be resolved on a later cycle.

CVB - NJ611
Defines duration of convert to binary.

CVB Delayed - NH863
Comes on one cycle after CVB is turned on and goes off one cycle after CVB goes off. Used in setting overflow interrupt for CVB.

CVB Sign Negative - NJ645
Remembers that the sign digit in CVB is negative.

CVD - NJ607
Defines duration of fixed point convert to decimal (CVD).

CVD Fixup - NJ647
Allows a cycle delay in CVD to fix up parity on result before sending the result to the store bus.

CVD Outgate - NJ649
Outgates left accumulator bits F-31 and right accumulator bits (0-3) to the store bus at the end of CVD.

Cycle 1 - NK105
Defines the decode cycle of any op.

Cycle 2 - NK107
One-cycle trigger defining the cycle after the decode cycle provided there is no step op.

Cycle 3 - NK107
One-cycle trigger defining the cycle after cycle 2 provided there is no step op.

Data Available - NK335
Defines that all operands are available for FXP and SI executions.

Data Interrupt - NH867
Used to define one cycle data interrupt signal to the I-Box. May be set by CVB and VFL operations.

Double Ingate - NG897
Defines that an even-odd pair of GPRs are to be ingated.

Decoder Error - NH771
Used to record a multiply decode error.

Divide - NJ613
Defines duration of fixed point divide.

EBAR - NR064, 065
The ending byte address register is a three-bit register that contains (in an encoded form) the position of the last sink byte to be processed.

End Fetch - NK443
Defines that the last memory operand has been fetched for LM.

FA-FB Data Available Trigger - NR305
Indicates the availability of registers FA-FB for processing. For instructions that fetch the sink word, it means that data is available. For instructions that do not fetch the sink word, it means that the register is not filled with result bytes. Therefore, processing of bytes may continue.

FA-FB Request Out - NR407
Indicates that there is an outstanding FLB request with a sink address of registers FA-FB.

FC End Trigger (FD End Trigger) - NR225
Indicates that the last table word will be in register FC (FD).

FC-FD Data Available Trigger - NR321
Indicates the availability of data in registers FC-FD for processing. When executing TR or TRT, it indicates the availability of data in register FC.

FC-FD Request Out - NR406
Indicates that there is an outstanding FLB request with a sink address of registers FC-FD or register FC.

FC Next (FD Next) - NR221, 226
Indicates the priority of table word requests.

FD Data Availability Trigger - NR322
Indicates the availability of data in register FD. Used only when executing TR or TRT.

FD Request Out - NR405
Indicates an outstanding FLB request with a sink address of register FD.

Feedback Sign - NH203
Used to record the sign output of the adder on the previous cycle.

Fetch - NK435
Defines which FXB pair (FA-FB or FC-FD) is next to receive data from the FLBs for LM.

Fill Register - NR042, 044
The fill register is a nine-bit register (including parity) that saves the first byte of the sink field for ED and EDMK instructions. The contents of the fill register are outgated to the result bus when necessary.

Fill Register Outgate Trigger - NR184
Outgates the fill register to the result bus.

Fixed Divide Interrupt - NJ653
A one-cycle trigger that signals an interrupt to I-Box if divide or CVB overflows.

FLB Request Trigger - NR403, 404
Six triggers, one associated with each FLB, used to notify the floating buffer controls to outgate one of the six buffers to the VFLEU.

FLB Sk Address Trigger - NR409
Two triggers that hold (in an encoded form) the sink address information that accompanies the FLB request signal.

DIAGRAM 5-132. FIXED POINT TRIGGER LIST (SHEET 1 OF 3)

Fetch Protect Interrupt - NK541
One-cycle trigger to signal a fetch protect interrupt to the I-box.

FXA A CC - NK469
One-cycle trigger (to I-Box) that sets the CC from the CC A bus. (For shift operations.)

FXA B CC - NK467
One-cycle trigger (to I-Box) that sets the CC from the CC B bus. (For all FXP operations, but not shift VFL operation.

FXA In Process - NK379
One-cycle trigger continually sampling 'FXP in process' and 'VFL in process.' Needed to extend 'in process' condition by one cycle.

FXB AE - NK533, 535, 565
Three triggers (for FA, FC, FD) that indicate an address exception (AE) violation for the data currently in the FXB. Condition can only come from the FLB bus.

FXB Divisor - NG933
Defines RX divide execution time.

FXB/FLB Register - NK501, 503
The FXB/FLB register is a three-bit register used for the execution of TRT. It holds the binary value signifying which floating buffer (1-6) the current argument word came from.

FXB FP - NK531, 575
Six triggers (one per FXB) used to indicate a fetch protect (FP) violation for the data currently in the FXB.

FXB Full - NK531, 575
Six triggers (one per FXB) that define when there is data in the FXB.

FXB SNB - NK531, 575
Six triggers (one per FXB) called the 'signal not busy' triggers. Used to turn off the FXB busy trigger in the I-Box.

FXP Error Generate - NK105
One-cycle trigger used to invert parity of selected bytes of the CPA during RX load (58).

FXP In Process - NK367
Defines when FXP op is being executed when execution is longer than one cycle.

FXOS Conditional - NK071, 083
Six triggers (one per FXOS position) to define the status of that FXOS position.

FXOS Decrement - NK109
One-cycle trigger to decrement the FXOS counter in the I-Box.

FXOS Full - NK071, 083
Six triggers (one per FXOS position) to define the status of that FXOS position.

FXOS Outgate Ring - NK041, 057
Six triggers making an outgate ring to outgate the FXOS (one FXOS position at any one time).

FXOS Parity Error - NK205
Three triggers to hold parity error conditions from the FXOS.

FXOS Set CC - NK041, 057
Six triggers (one per FXOS position) to define the set CC (condition code) ability of that op.

FXP Complete - NK419
A two-cycle signal to the I-Box at the end of SSM and RDD execution.

FXP Overlap - NK371
Defines that portion of a multicycle FXP execution (MPY, DIV, SHIFT) during which a VFL op (not DD, DF) could be issued.

FXP Set CC - NK469
Defines that the FXP op in process is to set the CC.

GPR Accept - NG931
Signals the I-Box that it has been given control for ingates to the GPRs.

GPR A Bus Outgates - NG801, 869
Sixteen outgate triggers (one per GPR) to outgate bits 0-31 of the GPR to bit positions 0-31 of the A bus.

GPR B Bus Outgates - NG801, 869
Sixteen outgate triggers (one per GPR) to outgate bits 0-31 of the GPR to bit positions 0-31 of the B bus.

GPR Sk Register - NG881, 897
Eight triggers that hold the address of the GPR pair (even-odd pair) that will be the sink of the current op in execution.

Half-Sum Error Byte 0 - NH523
Used to record a half-sum error in byte 0.

Half-Sum Error Byte 1 - NH521
Used to record a half-sum error in byte 1.

Half-Sum Error Byte 2 - NH521
Used to record a half-sum error in byte 2.

Half-Sum Error Byte 3 - NH519
Used to record a half-sum error in byte 3.

Half-Sum Error Byte 4 - NH519
Used to record a half-sum error in byte 4.

Halfword Op - NK515
Defines execution time of halfword ops.

Hardstop - NH789
Turned on (if not disabled) by a fixed point error. When on the trigger prevents starting the next cycle.

HO Cross Outgates - NK601, 613
Six outgate triggers (one per FXB) to outgate bits 0-15 of the FXB to bit positions 16-31 of the A bus.

HOD Trigger - NR302
Indicates that the high-order digit of the source field is being processed.

HO Straight Outgates - NK601, 613
Six outgate triggers (one per FXB) to outgate bits 0-15 of the FXB to bit positions 0-15 of the A bus.

I Counter Bit 1 - NJ653
The 2^0 bit of the iteration counter.

I Counter Bit 2 - NJ551
The 2^1 bit of the iteration counter.

I Counter Bit 4 - NJ553
The 2^2 bit of the iteration counter.

I Counter Bit 8 - NJ553
The 2^3 bit of the iteration counter.

I Counter Bit 16 - NJ555
The 2^4 bit of the iteration counter.

I Counter Bit 32 - NJ557
The 2^5 bit of the iteration counter.

IC-ISK - NK415
Defines the cycle of IC or ISK execution that gates data to the GPRs.

Ingate Control - NR167, 169
The ingate counter is a three-bit counter that points to the byte position to be ingated next. The counter is initially set with the sink starting byte address, and then decremented or incremented by one depending on the instruction.

Ingate GPR - NR373
One-cycle trigger that causes a GPR ingate.

I Priority - NG931
One-cycle trigger continually sampling the GPR request line from the I-Box.

Immediate Data Register - NR001, 003
The immediate data register holds the immediate field before an SI instruction is executed, and contains the result byte after processing of the instruction is completed. It is a nine-bit register (including parity) and has outgates to (1) the source bus, (2) the sink bus, (3) the store data bus, and (4) the singleshots.

Instruction Triggers - NR381, 385
Fourteen triggers, one per instruction. These triggers indicate which SS instruction the VFLEU is currently executing.

Last Op Trigger - NR424
Indicates that the last op of the instruction currently being executed has been passed.

Left - NG925
Data trigger to define left shifts (off defines right shifts).

LM End - NK437
Defines the last two cycles of a LM execution.

LM Process - NK439
One-cycle trigger that defines the transfer of one FXB to a GPR for LM.

Load Complement, Load Negative, Load and Test, Add, Subtract, Load Positive - NH853
Defines specific operations used to set the condition code on the cycle following the execution cycle.

Load Multiple - NK437
Defines LM execution time.

LOD Trigger - NR302
Indicates that the low-order digit of the source field is being processed.

Logical - NG925
Data trigger to define logical shifts (off defines arithmetic shifts).

LO Straight Outgates - NK601, 613
Six outgate triggers (one per FXB) to outgate bits 16-31 of the FXB to bit positions 16-31 of the A bus.

L Register - NR067, 068, 101, 102
The L register is an eight-bit register and counter combination. The register is initially set with the length field, and for all the VFL instructions except for the double L type, the low-order three bits are used to determine the ending byte address. For the double L type instructions, the eight-bit unit is broken into two four-bit counters, one counter for the sink length field and the other for the source length field. The counters are decremented by one when necessary during execution.

Make Request - NK443
Remembers that an operand fetch (for LM) is required (happens if there is a break in the string of ops).

MALS Trigger C (MALS Trigger D) - NR427
Indicates that the table word in register FC (FD) is from the argument word currently being processed (overlapping TR).

Mark Field - NR082, 084
The mark field controls which bytes of the resulting sink doubleword are to be stored. The mark field is an eight-bit field, one trigger in control of each result byte.

Multiply Decode of Zero, Group 1/2 - NH001
Used in checking multiply decode.

Multiply Decode of Zero, Group 4/8 - NH013
Used in checking multiply decode.

Multiply Decode 1 Complement - NH003
Used to record multiplier decode.

Multiply Decode 1 True - NH003
Used to record multiplier decode.

Multiply Decode 2 Complement - NH005
Used to record multiplier decode.

Multiply Decode 2 True - NH005
Used to record multiplier decode.

Multiply Decode 4 Complement - NH009
Used to record multiplier decode.

Multiply Decode 4 True - NH009
Used to record multiplier decode.

Multiply Decode 8 Complement - NH011
Used to record multiplier decode.

Multiply Decode 8 True - NH011
Used to record multiplier decode.

Multiply - NJ609
Defines duration of fixed point multiply (MUL).

MVC Serial Trigger - NR201
Indicates that MVC should be executed serially.

Nonzero Trigger - NR301
Indicates that during the execution, a nonzero result byte(s) was generated. (Used to set the condition code.)

NOX - NK413
Defines execution time of NI, OI, XI.

Odd - NG891
Defines which GPR of the even-odd pair is to be ingated.

Outgate A-B To SDB - NK463
Outgates the A and B bus to the SDBs.

Outgate GPR Byte To SDB - NK463
Outgates B bus 24-31 to bit positions 48-55 and 56-63 of the SDBs.

DIAGRAM 5-132. FIXED POINT TRIGGER LIST (SHEET 2 OF 3)

Outgate ID To Source - NK507
Outgates ID register to the source bus.

Outgate ID To SDB - NK461
Outgates ID register to bit positions 48-55 and 56-63 of the SDBs.

Outgate ID To Sink - NK507
Outgates ID register to the sink bus.

Outgate ID To SS - NK461
Outgates ID register to fire the signal out singleshots in the PSCE.

Overlap Process Trigger - NR242
Defines the cycle during which an overlapped byte is handled when executing TR.

Overlap Tags - NR087, 088
The overlap tags facilitate the processing of overlap in VFL. There are eight tags (triggers), one associated with each source byte. An overlap tag on blocks the ingating of its respective byte into FC-FD from the floating buffer bus.

Overlap Trigger - NR305
Indicates overlapping source and sink fields.

Op Reset - NK367
Defines the last cycle of most RR and RX ops. It also defines a cancel op cycle.

Parity Error Byte 0 - NH787
Used to record a parity error in byte 0.

Parity Error Byte 1 - NH787
Used to record a parity error in byte 1.

Parity Error Byte 2 - NH787
Used to record a parity error in byte 2.

Parity Error Byte 3 - NH787
Used to record a parity error in byte 3.

Parity Error Byte 4 - NH787
Used to record a parity error in byte 4.

Process Counter - NK433
Two-position counter that identifies which FXB is being processed for LM execution.

P Trigger - NR387
Indicates that bit 12 in FXOS was on.

RDD Sample - NK417
Used to define a single cycle during which the 'direct in' lines are to be sampled.

Request FC Trigger (FD Trigger) - NR223, 224
Indicates that as soon as the FXOS goes not empty, a request with a sink address of FC (FD) should be made.

Right Accumulator Bit 32 - NH013
Used to save one bit of the multiplier for decode overlap.

RUA Response - NG893
One-cycle trigger to condition RUA response lines.

RUM Response - NG903
One-cycle trigger to condition RUM response lines.

SAR Name Register - NR410
The SAR name register is a two-bit register that holds FXOS bits 19 and 20.

SBO Fetch Protect - NK541
One-cycle trigger continually sampling SBO fetch protect (FP) bit. (FP bit leads data by one cycle.)

SBO Sink Bit - NK509
Three-bit register (sink bits 3, 4, and 5) continually sampling SBO sink bus. (Sink address leads data by one cycle.)

Set Outgate - NR388
Defines the cycle during which the first source and/or sink outgate triggers are turned on.

Shift - NJ625
Defines duration of fixed point shift instruction.

Shifter Output All Zeros - NJ483
Defines the shifter output as all zeros on a shift instruction.

Shifter Overflow - NJ485
Defines a shifter overflow on a shift instruction.

Shifter Sign Out - NJ483
Defines the sign output of the shifter.

Shifter Stage 0 - NJ599
Used to define shift of zero in the first stage of the shifter.

Shifter to Accumulator, Right - NJ627
Gates the shifter output (bits 32-63) into the right accumulator.

Shifter to GPR - NJ615
Gates output of shifter (64 bits) to GPRs.

Shift Left 1 - NJ595
Gates the left accumulator, shifted left 1, into the B input of the adder.

Shift Right 4 - NJ597
Gates the left accumulator, shifted right 4, into the carry save adder.

SI End Op - NK415
Defines the last cycle of all SI executions.

Sign-Save - NH205
Used to remember the sign of the dividend in divide and the sign of the operand in CVD.

Single Byte - NG897
Defines that only bits 24-31 of the GPR are to be ingated.

Single Pulse Control - NH771
Used in single pulse mode to force the hardstop trigger on and off to give single sample pulses.

Sink Bus Outgates - NR121, 124
Eight triggers, one associated with each sink byte, used to control the outgating of one of eight bytes from registers FA-FB to the execution unit.

Sink End Trigger - NR264
Indicates, that for a double L op (PACK, UNPK, MVO), the L1 field is depleted.

Sink Notify Trigger - NR303
Notifies the I-Box that another pattern or argument word has been requested by the VFLEU while executing ED, EDMK, TR or TRT.

Sink Request Trigger - NR323
Indicates, that as soon as the FXOS goes not empty, a request should be made with sink address of FA-FB.

Sink Used Trigger - NR324
Indicates that the sink doubleword and/or SAR name from the outgated op has been used.

Sink Wraparound Trigger - NR201
Indicates one more byte to be processed in the sink word in registers FA-FB.

Skew Register - NR151, 152
The skew register is a four-bit register used for the execution of pack, unpack, and move with offset. It saves either the HOD or the LOD portion of the source byte for one cycle, and then outgates it onto the LOD portion of the result bus.

Skew Register Outgate - NR141
Outgates the skew register to the result bus.

Skip 1 - NJ587
Used in CVD to remember a leading string of 5 zeros (right accumulator bits 0-4).

Skip 2/Limit Equals 2 - NJ589
Used in CVD to remember a leading string of 9 zeros (right accumulator bits 0-8). Also used in multiply to remember a leading string of 9 zeros or 9 ones (right accumulator bits 0-8).

Skip 3 - NJ591
Used in CVD to remember a leading string of 13 zeros (right accumulator bits 0-12).

Skip 4/Limit Equals 4 - NJ593
Used in CVD to remember leading string of 17 zeros (right accumulator bits 0-16).
Used in multiply to remember a leading string of 17 zeros or 17 ones (right accumulator bits 0-16).

Source Bus Outgates - NR106, 108
Eight triggers, one associated with each source byte. These triggers control the outgating of one of eight bytes from registers FC-FD to the execution unit.

Source End Trigger - NR264
Indicates, that for a double L op (PACK, UNPK, MVO), the L2 field is depleted.

Source Notify Trigger - NR304
Notifies the I-Box that another source word has been requested by the VFLEU while executing ED or EDMK.

Source Request Trigger - NR301
Indicates, that as soon as the FXOS goes not empty, a request should be made with sink address of FC-FD.

Source Used Trigger - NR324
Indicates that the source doubleword from the outgated op has been used (requested).

Source Wraparound Trigger - NR284
Indicates one byte left to be processed in the source word in registers FC-FD.

Start Execute Trigger - NR326
Defines the first execution cycle.

Start Request FC Trigger (FD Trigger) - NR425
Indicates that a request must be made after start as soon as the FXOS goes not empty.

S Trigger - NR183
Defines start of significant data during edit operation.

Store Trigger - NR325
Indicates that registers FA-FB should be stored as soon as the SDB becomes available.

Straight Out Accumulator, Right to A - NJ641
Gates the right accumulator unshifted into the A adder input.

Straight Out Accumulator, Right to B - NJ643
Gates the right accumulator unshifted into the B adder input.

Terminate Trigger - NR423
Indicates the termination (as opposed to completion) of TRT and CLC.

Timer - NJ605
Defines duration of timer update.

Timer Interrupt - NH855
Defines timer interrupt signal (one cycle) to the I-Box.

Timer Outgate - NJ651
Outgates left accumulator (bits 0-31) to store bus at end of CVD.

TM - NK411
Defines test under mask execution time.

Tof Overlap Tags - NR306
Defines the cycle during which all 8 overlap tags are turned off.

T and S - NK411
Defines test and set execution time.

TW Counter - NR085, 086
The table word counter is a three-bit counter used in the execution of TR and TRT. It is initially set with the sink starting byte address, and then incremented by one when necessary during execution.

Use FC Trigger (FD Trigger) - NR241
Indicates that the next table byte should be processed from register FC (FD).

Valid Trigger - NR243
Notifies the I-Box that there is a valid byte on the byte bus.

VFL Cancel Trigger - NR304
Indicates that the I-Box fetched operands that have not been used during execution and must be canceled. (Associated FLB must be freed.)

VFL End Trigger - NR363
Indicates that the last byte has been processed.

VFL In Process - NK373
Defines when VFL op is being executed when execution is longer than 1 cycle.

VFL Outgate To SDB - NR426
Outgates registers FA-FB to the SDB.

VFL Overlap - NK371
Defines the last op portion of VFL ops (not DD, DF) during which some FXP ops can be issued.

VFL Set CC - NK467
Defines that the VFL op in process is to set the CC.

VFL Start - NR322
Defines the cycles between the decode cycle and the first execute cycle.

WRD - NK419
Defines write direct execution time.

Write Direct Register - NR161, 162
The write direct register is an eight-bit register that holds the direct-out static signals for the write direct instruction.

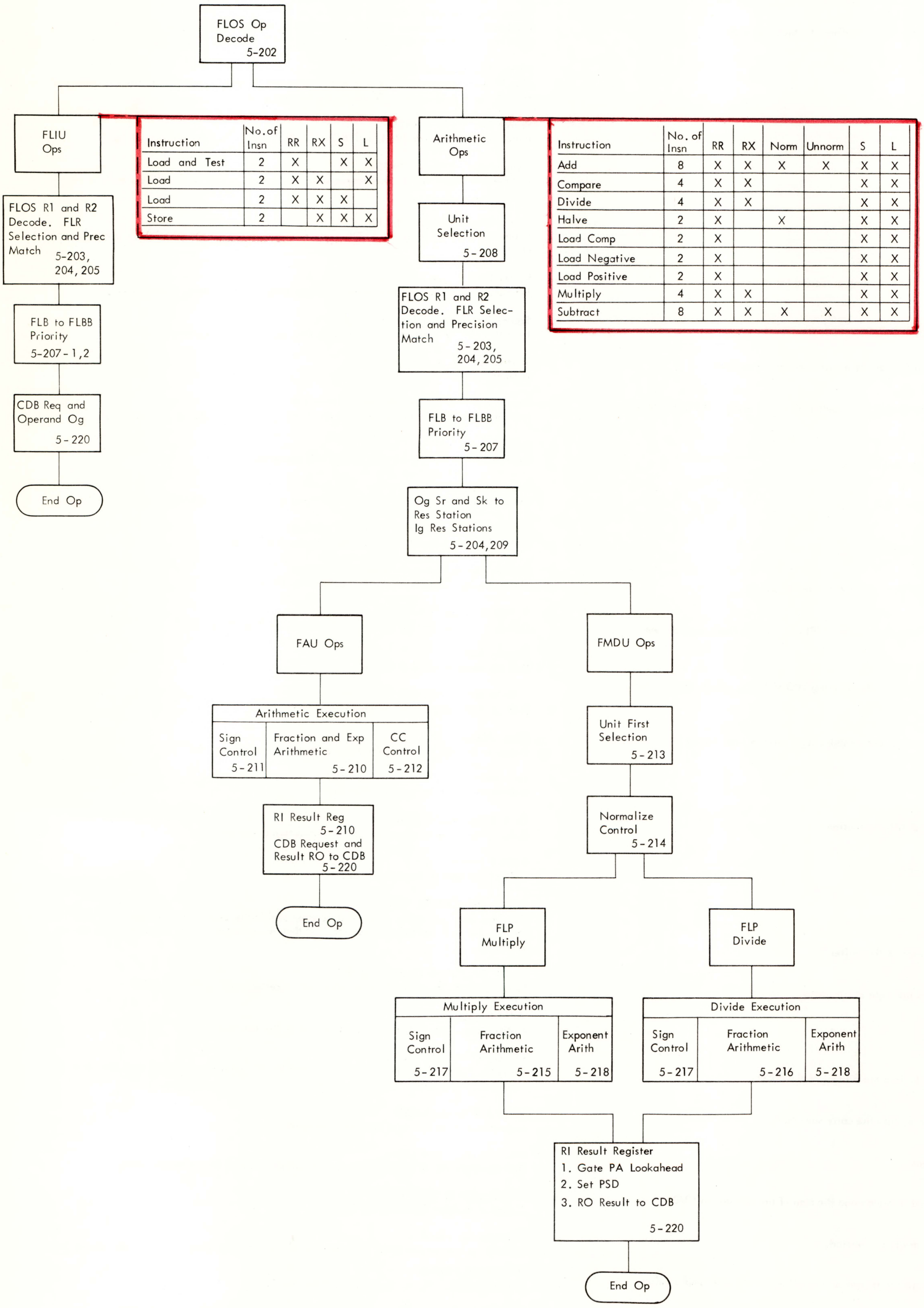
XOR - NG927
Defines the execution time of O, OR, X, XR.

DIAGRAM 5-132. FIXED POINT TRIGGER LIST (SHEET 3 OF 3)

FLIU		Decode	Data 1	Data 2			
Ops				CDB Cycle			
FAU	RR	Decode	Select Data 1	Data 2	Execute 1	Execute 2	CDB Cycle
Ops	RX	Decode	Select	Execute 1	Execute 2	CDB Cycle	
FMDU	RR	Decode	Select Data 1	Data 2 *Possible Normalize	*Normalize	Execute Cycles (3-11)	CDB Cycle
Ops	RX	Decode	Select	*Normalize	*Normalize	Execute Cycles (3-11)	CDB Cycle

*A normalize cycle occurs only for an operand that is set into a reservation station unnormalized. Execute cycles follow the select or data 2 cycle when both operands arrive normalized.

Simplified Instruction Sequencing.


FLOS Op Decode 5-202
FLIU Ops
FLOS R1 and R2 Decode. FLR Selection and Prec Match 5-203, 204, 205
FLB to FLBB Priority 5-207-1,2
CDB Req and Operand Og 5-220
End Op
Arithmetic Ops
Unit Selection 5-208
FLOS R1 and R2 Decode. FLR Selection and Precision Match 5-203, 204, 205
FLB to FLBB Priority 5-207
Og Sr and Sk to Res Station Ig Res Stations 5-204, 209
FAU Ops
Arithmetic Execution
Sign Control 5-211
Fraction and Exp Arithmetic 5-210
CC Control 5-212
RI Result Reg 5-210 CDB Request and Result RO to CDB 5-220
End Op
FMDU Ops
Unit First Selection 5-213
Normalize Control 5-214
FLP Multiply
FLP Divide
Multiply Execution
Sign Control 5-217
Fraction Arithmetic 5-215
Exponent Arith 5-218
Divide Execution
Sign Control 5-217
Fraction Arithmetic 5-216
Exponent Arith 5-218
RI Result Register
1. Gate PA Lookahead
2. Set PSD
3. RO Result to CDB
5-220
End Op

Instruction	No. of Insn	RR	RX	S	L
Load and Test	2	X		X	X
Load	2	X	X		X
Load	2	X	X	X	
Store	2		X	X	X

Instruction	No. of Insn	RR	RX	Norm	Unnorm	S	L
Add	8	X	X	X	X	X	X
Compare	4	X	X			X	X
Divide	4	X	X			X	X
Halve	2	X		X		X	X
Load Comp	2	X				X	X
Load Negative	2	X				X	X
Load Positive	2	X				X	X
Multiply	4	X	X			X	X
Subtract	8	X	X	X	X	X	X

DIAGRAM 5-200. FLOATING POINT OPERATIONS

Objectives:

1. Controls ingating, storing and outgating of op codes from the I-Box for use by the arithmetic units.

2. Signals the I-Box to deliver more ops to replace those which have been processed.

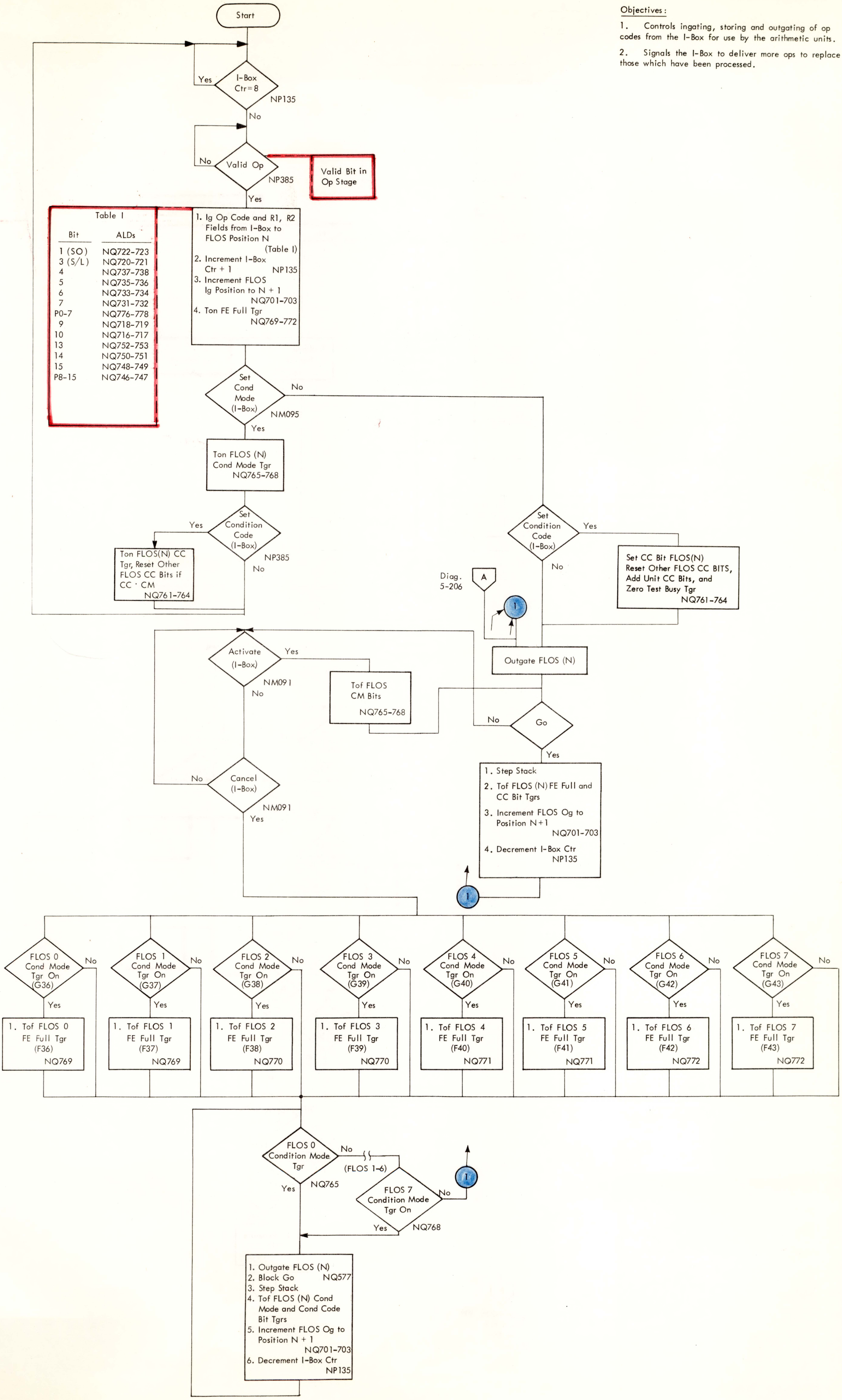

Start
I-Box Ctr=8
Yes
No
NP135
Valid Op
No
Yes
NP385
Valid Bit in Op Stage
1. Ig Op Code and R1, R2 Fields from I-Box to FLOS Position N (Table I)
2. Increment I-Box Ctr + 1 NP135
3. Increment FLOS Ig Position to N + 1 NQ701-703
4. Ton FE Full Tgr NQ769-772
Set Cond Mode (I-Box)
No
NM095
Yes
Ton FLOS (N) Cond Mode Tgr NQ765-768
Set Condition Code (I-Box)
Yes
NP385
No
Ton FLOS(N) CC Tgr, Reset Other FLOS CC Bits if CC · CM NQ761-764
Set Condition Code (I-Box)
Yes
No
Set CC Bit FLOS(N) Reset Other FLOS CC BITS, Add Unit CC Bits, and Zero Test Busy Tgr NQ761-764
Diag. 5-206
A
1
Outgate FLOS (N)
Activate (I-Box)
Yes
NM091
No
Tof FLOS CM Bits NQ765-768
No
Go
Yes
1. Step Stack
2. Tof FLOS (N) FE Full and CC Bit Tgrs
3. Increment FLOS Og to Position N+1 NQ701-703
4. Decrement I-Box Ctr NP135
No
Cancel (I-Box)
NM091
Yes
1
FLOS 0 Cond Mode Tgr On (G36)
No
Yes
1. Tof FLOS 0 FE Full Tgr (F36) NQ769
FLOS 1 Cond Mode Tgr On (G37)
No
Yes
1. Tof FLOS 1 FE Full Tgr (F37) NQ769
FLOS 2 Cond Mode Tgr On (G38)
No
Yes
1. Tof FLOS 2 FE Full Tgr (F38) NQ770
FLOS 3 Cond Mode Tgr On (G39)
No
Yes
1. Tof FLOS 3 FE Full Tgr (F39) NQ770
FLOS 4 Cond Mode Tgr On (G40)
No
Yes
1. Tof FLOS 4 FE Full Tgr (F40) NQ771
FLOS 5 Cond Mode Tgr On (G41)
No
Yes
1. Tof FLOS 5 FE Full Tgr (F41) NQ771
FLOS 6 Cond Mode Tgr On (G42)
No
Yes
1. Tof FLOS 6 FE Full Tgr (F42) NQ772
FLOS 7 Cond Mode Tgr On (G43)
No
Yes
1. Tof FLOS 7 FE Full Tgr (F43) NQ772
FLOS 0 Condition Mode Tgr
No
(FLOS 1-6)
NQ765
Yes
FLOS 7 Condition Mode Tgr On
No
Yes
NQ768
1
1. Outgate FLOS (N)
2. Block Go NQ577
3. Step Stack
4. Tof FLOS (N) Cond Mode and Cond Code Bit Tgrs
5. Increment FLOS Og to Position N + 1 NQ701-703
6. Decrement I-Box Ctr NP135

Table I

Bit	ALDs
1 (SO)	NQ722-723
3 (S/L)	NQ720-721
4	NQ737-738
5	NQ735-736
6	NQ733-734
7	NQ731-732
P0-7	NQ776-778
9	NQ718-719
10	NQ716-717
13	NQ752-753
14	NQ750-751
15	NQ748-749
P8-15	NQ746-747

DIAGRAM 5-201. FLOS CONTROLS

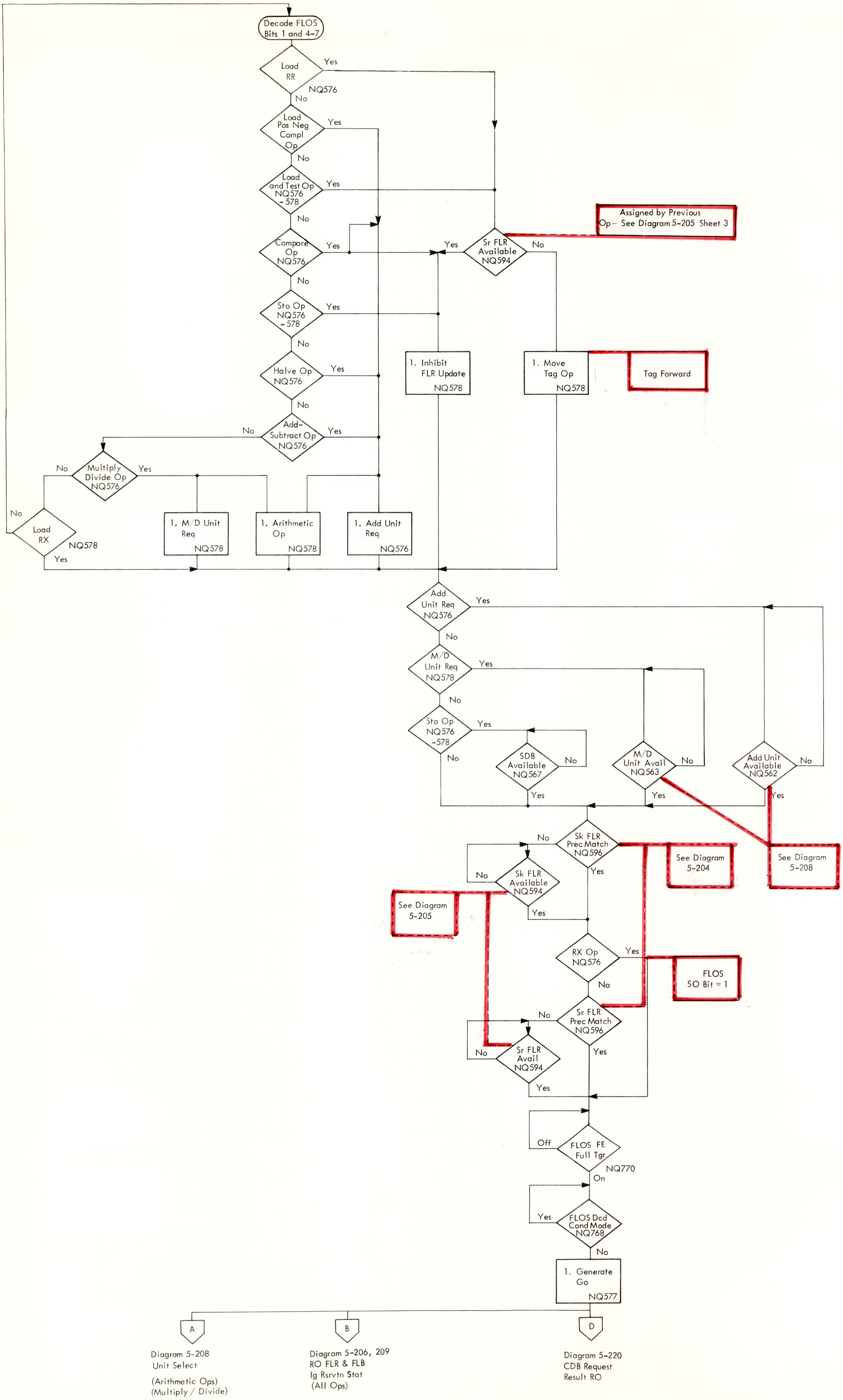


DIAGRAM 5-202. FLOS OP DECODE - GO GENERATION

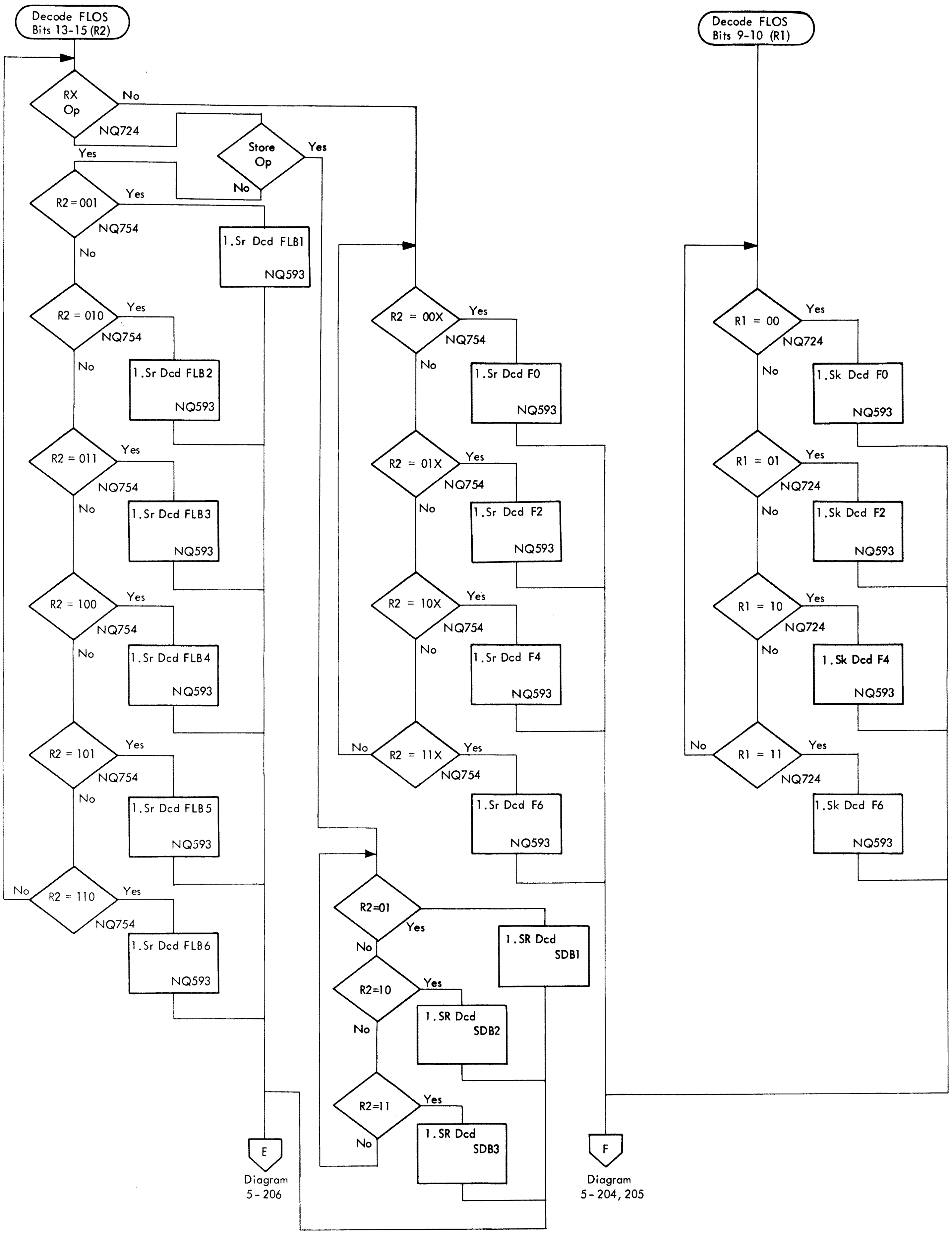


DIAGRAM 5-203. FLOS R1 AND R2 DECODE

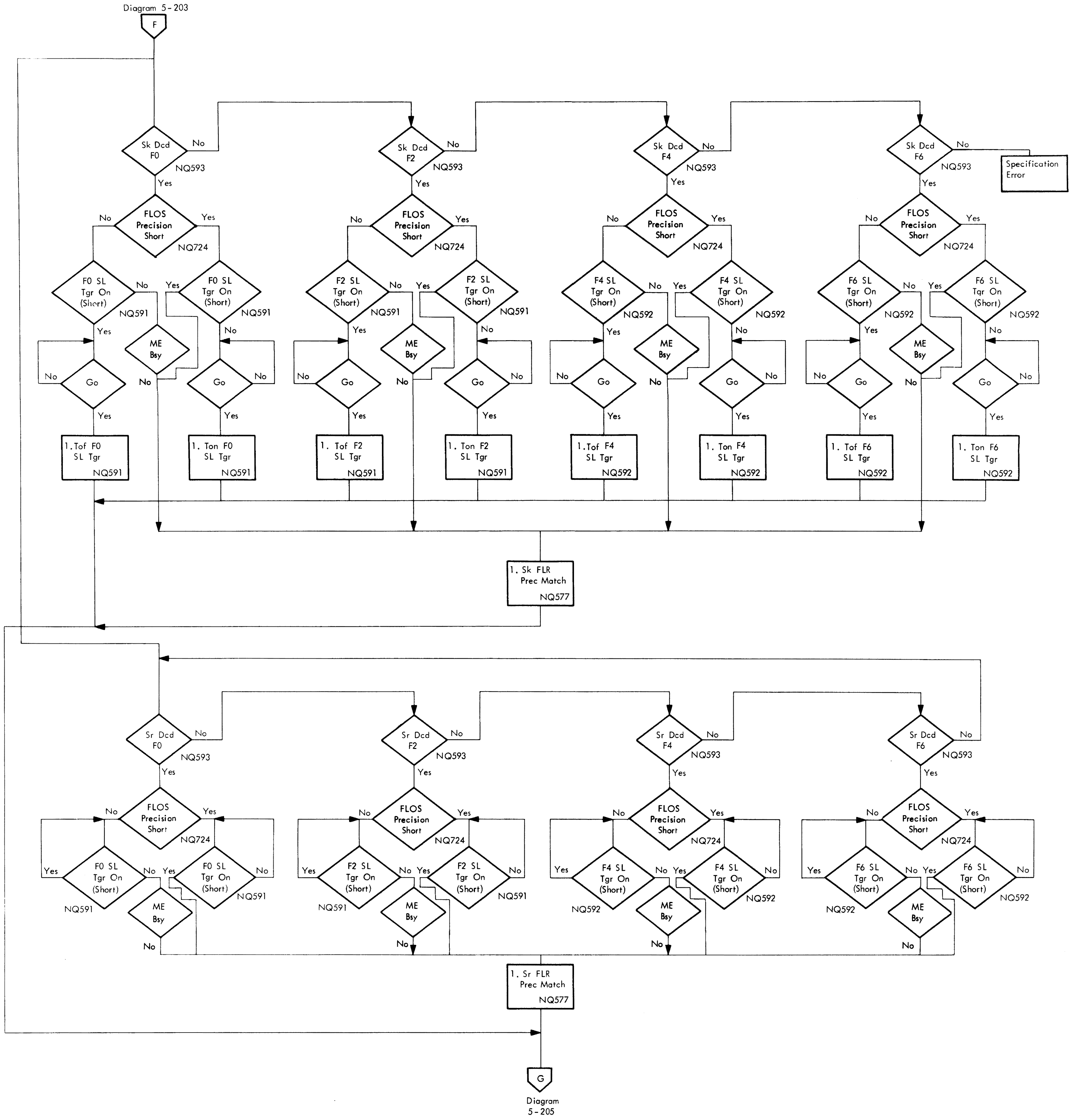


DIAGRAM 5-204. FLR PRECISION MATCH

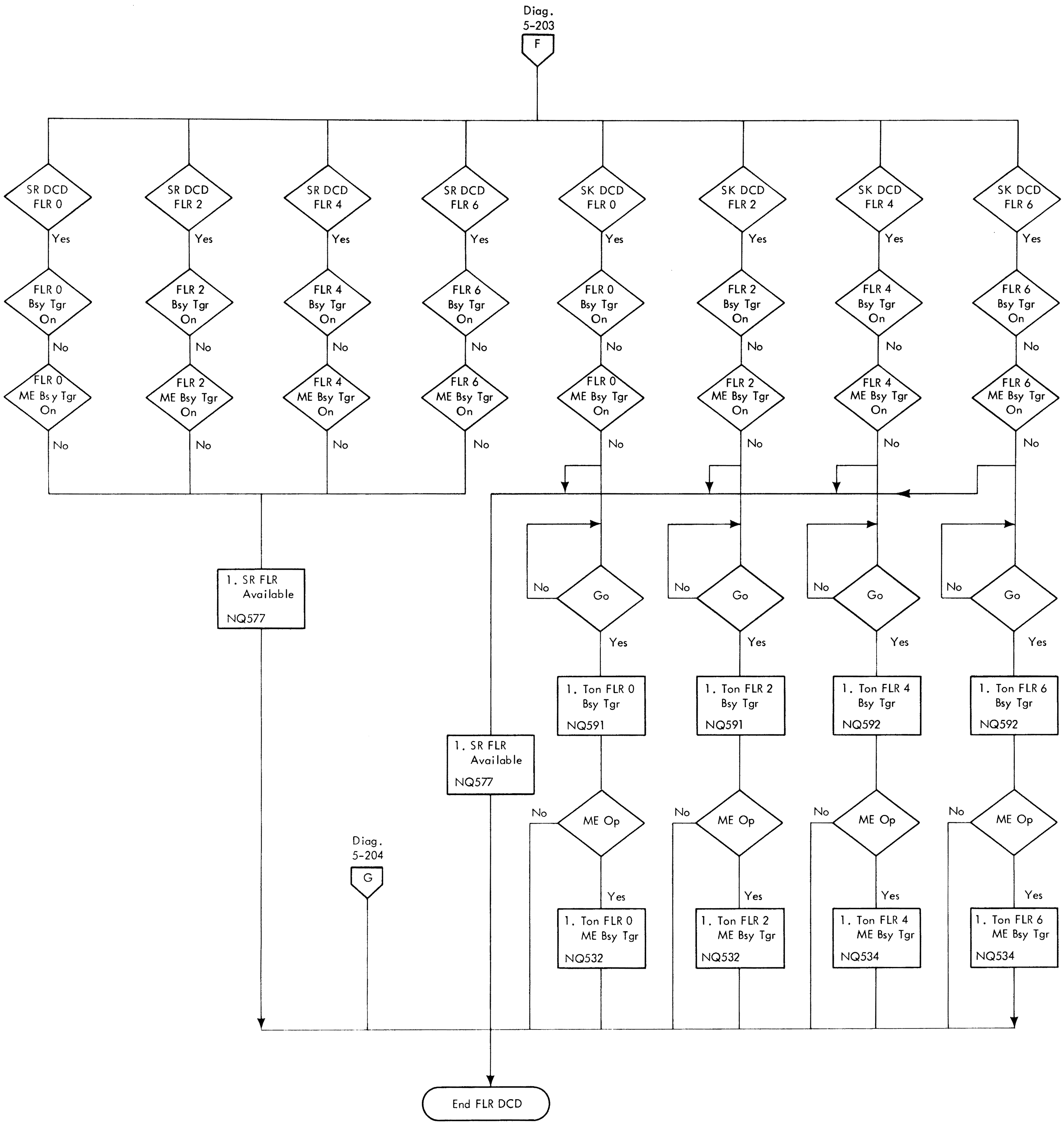


DIAGRAM 5-205. FLR AVAILABILITY

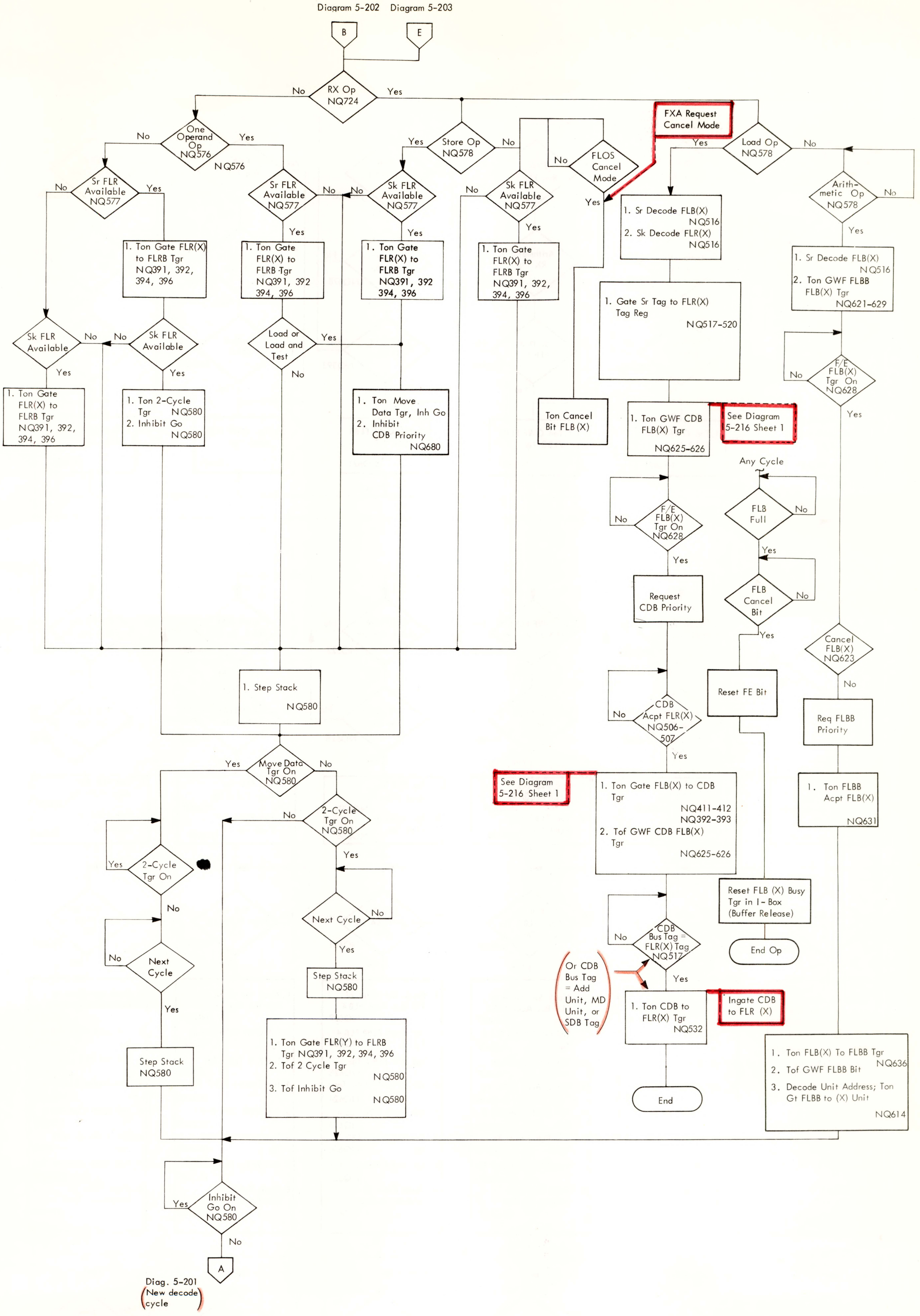


Diagram 5-202
Diagram 5-203
B
E
RX Op NQ724
No
Yes
One Operand Op NQ576
NQ576
Store Op NQ578
FXA Request Cancel Mode
Load Op NQ578
Sr FLR Available NQ577
Sr FLR Available NQ577
Sk FLR Available NQ577
Sk FLR Available NQ577
FLOS Cancel Mode
Arith-metic Op NQ578
1. Ton Gate FLR(X) to FLRB Tgr NQ391, 392, 394, 396
1. Ton Gate FLR(X) to FLRB Tgr NQ391, 392 394, 396
1. Ton Gate FLR(X) to FLRB Tgr NQ391, 392 394, 396
1. Ton Gate FLR(X) to FLRB Tgr NQ391, 392, 394, 396
1. Sr Decode FLB(X) NQ516
2. Sk Decode FLR(X) NQ516
1. Sr Decode FLB(X) NQ516
2. Ton GWF FLBB FLB(X) Tgr NQ621-629
1. Gate Sr Tag to FLR(X) Tag Reg NQ517-520
Sk FLR Available
Sk FLR Available
Load or Load and Test
F/E FLB(X) Tgr On NQ628
1. Ton Gate FLR(X) to FLRB Tgr NQ391, 392, 394, 396
1. Ton 2-Cycle Tgr NQ580
2. Inhibit Go NQ580
1. Ton Move Data Tgr, Inh Go
2. Inhibit CDB Priority NQ680
Ton Cancel Bit FLB (X)
1. Ton GWF CDB FLB(X) Tgr NQ625-626
See Diagram 5-216 Sheet 1
Any Cycle
F/E FLB(X) Tgr On NQ628
FLB Full
FLB Cancel Bit
Request CDB Priority
Cancel FLB(X) NQ623
Reset FE Bit
Req FLBB Priority
1. Step Stack NQ580
CDB Acpt FLR(X) NQ506-507
1. Ton FLBB Acpt FLB(X) NQ631
Move Data Tgr On NQ580
See Diagram 5-216 Sheet 1
1. Ton Gate FLB(X) to CDB Tgr NQ411-412 NQ392-393
2. Tof GWF CDB FLB(X) Tgr NQ625-626
2-Cycle Tgr On NQ580
2-Cycle Tgr On
Reset FLB (X) Busy Tgr in I-Box (Buffer Release)
Next Cycle
End Op
Next Cycle
CDB Bus Tag = FLR(X) Tag NQ517
Or CDB Bus Tag = Add Unit, MD Unit, or SDB Tag
Step Stack NQ580
1. Ton CDB to FLR(X) Tgr NQ532
Ingate CDB to FLR (X)
1. Ton Gate FLR(Y) to FLRB Tgr NQ391, 392, 394, 396
2. Tof 2 Cycle Tgr NQ580
3. Tof Inhibit Go NQ580
Step Stack NQ580
1. Ton FLB(X) To FLBB Tgr NQ636
2. Tof GWF FLBB Bit
3. Decode Unit Address; Ton Gt FLBB to (X) Unit NQ614
End
Inhibit Go On NQ580
A
Diag. 5-201
New decode cycle

DIAGRAM 5-206. FLR OUTGATING

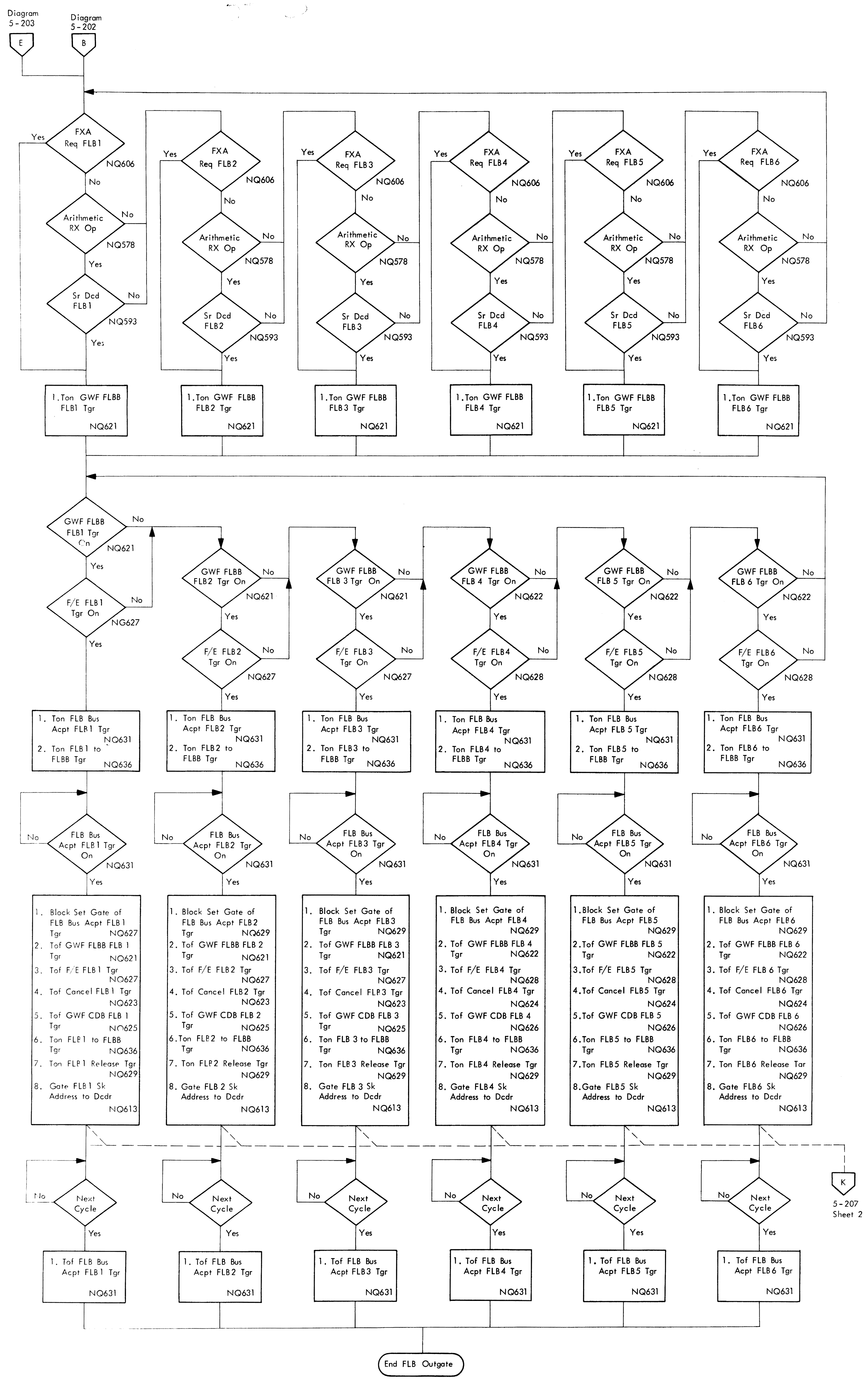


Diagram 5-203
E
Diagram 5-202
B
FXA Req FLB1
Yes
NQ606
No
Arithmetic RX Op
No
NQ578
Yes
Sr Dcd FLB1
No
NQ593
Yes
1.Ton GWF FLBB FLB1 Tgr NQ621
FXA Req FLB2
NQ606
Arithmetic RX Op
NQ578
Sr Dcd FLB2
NQ593
1.Ton GWF FLBB FLB2 Tgr NQ621
FXA Req FLB3
NQ606
Arithmetic RX Op
NQ578
Sr Dcd FLB3
NQ593
1.Ton GWF FLBB FLB3 Tgr NQ621
FXA Req FLB4
NQ606
Arithmetic RX Op
NQ578
Sr Dcd FLB4
NQ593
1.Ton GWF FLBB FLB4 Tgr NQ621
FXA Req FLB5
NQ606
Arithmetic RX Op
NQ578
Sr Dcd FLB5
NQ593
1.Ton GWF FLBB FLB5 Tgr NQ621
FXA Req FLB6
NQ606
Arithmetic RX Op
NQ578
Sr Dcd FLB6
NQ593
1.Ton GWF FLBB FLB6 Tgr NQ621
GWF FLBB FLB1 Tgr On
NQ621
F/E FLB1 Tgr On
NG627
GWF FLBB FLB2 Tgr On
NQ621
F/E FLB2 Tgr On
NQ627
GWF FLBB FLB 3 Tgr On
NQ621
F/E FLB3 Tgr On
NQ627
GWF FLBB FLB 4 Tgr On
NQ622
F/E FLB4 Tgr On
NQ628
GWF FLBB FLB 5 Tgr On
NQ622
F/E FLB5 Tgr On
NQ628
GWF FLBB FLB 6 Tgr On
NQ622
F/E FLB6 Tgr On
NQ628
1. Ton FLB Bus Acpt FLB1 Tgr NQ631
2. Ton FLB1 to FLBB Tgr NQ636
1. Ton FLB Bus Acpt FLB2 Tgr NQ631
2. Ton FLB2 to FLBB Tgr NQ636
1. Ton FLB Bus Acpt FLB3 Tgr NQ631
2. Ton FLB3 to FLBB Tgr NQ636
1. Ton FLB Bus Acpt FLB4 Tgr NQ631
2. Ton FLB4 to FLBB Tgr NQ636
1. Ton FLB Bus Acpt FLB 5 Tgr NQ631
2. Ton FLB5 to FLBB Tgr NQ636
1. Ton FLB Bus Acpt FLB6 Tgr NQ631
2. Ton FLB6 to FLBB Tgr NQ636
FLB Bus Acpt FLB1 Tgr On
NQ631
FLB Bus Acpt FLB2 Tgr On
NQ631
FLB Bus Acpt FLB3 Tgr On
NQ631
FLB Bus Acpt FLB4 Tgr On
NQ631
FLB Bus Acpt FLB5 Tgr On
NQ631
FLB Bus Acpt FLB6 Tgr On
NQ631
1. Block Set Gate of FLB Bus Acpt FLB1 Tgr NQ627
2. Tof GWF FLBB FLB 1 Tgr NQ621
3. Tof F/E FLB1 Tgr NQ627
4. Tof Cancel FLB1 Tgr NQ623
5. Tof GWF CDB FLB 1 Tgr NQ625
6. Ton FLB1 to FLBB Tgr NQ636
7. Ton FLB1 Release Tgr NQ629
8. Gate FLB1 Sk Address to Dcdr NQ613
1. Block Set Gate of FLB Bus Acpt FLB2 Tgr NQ629
2. Tof GWF FLBB FLB 2 Tgr NQ621
3. Tof F/E FLB2 Tgr NQ627
4. Tof Cancel FLB2 Tgr NQ623
5. Tof GWF CDB FLB 2 Tgr NQ625
6.Ton FLB2 to FLBB Tgr NQ636
7. Ton FLB2 Release Tgr NQ629
8. Gate FLB 2 Sk Address to Dcdr NQ613
1. Block Set Gate of FLB Bus Acpt FLB3 Tgr NQ629
2. Tof GWF FLBB FLB 3 Tgr NQ621
3. Tof F/E FLB3 Tgr NQ627
4. Tof Cancel FLB3 Tgr NQ623
5. Tof GWF CDB FLB 3 Tgr NQ625
6. Ton FLB 3 to FLBB Tgr NQ636
7. Ton FLB3 Release Tgr NQ629
8. Gate FLB 3 Sk Address to Dcdr NQ613
1. Block Set Gate of FLB Bus Acpt FLB4 NQ629
2. Tof GWF FLBB FLB 4 Tgr NQ622
3. Tof F/E FLB4 Tgr NQ628
4. Tof Cancel FLB4 Tgr NQ624
5. Tof GWF CDB FLB 4 NQ626
6. Ton FLB4 to FLBB Tgr NQ636
7. Ton FLB4 Release Tgr NQ629
8. Gate FLB4 Sk Address to Dcdr NQ613
1.Block Set Gate of FLB Bus Acpt FLB5 NQ629
2.Tof GWF FLBB FLB 5 Tgr NQ622
3.Tof F/E FLB5 Tgr NQ628
4.Tof Cancel FLB5 Tgr NQ624
5.Tof GWF CDB FLB 5 NQ626
6.Ton FLB5 to FLBB Tgr NQ636
7.Ton FLB5 Release Tgr NQ629
8.Gate FLB5 Sk Address to Dcdr NQ613
1. Block Set Gate of FLB Bus Acpt FLB6 NQ629
2. Tof GWF FLBB FLB 6 Tgr NQ622
3. Tof F/E FLB 6 Tgr NQ628
4. Tof Cancel FLB6 Tgr NQ624
5. Tof GWF CDB FLB 6 NQ626
6. Ton FLB6 to FLBB Tgr NQ636
7. Ton FLB6 Release Tar NQ629
8. Gate FLB6 Sk Address to Dcdr NQ613
Next Cycle
1. Tof FLB Bus Acpt FLB1 Tgr NQ631
1. Tof FLB Bus Acpt FLB2 Tgr NQ631
1. Tof FLB Bus Acpt FLB3 Tgr NQ631
1. Tof FLB Bus Acpt FLB4 Tgr NQ631
1. Tof FLB Bus Acpt FLB5 Tgr NQ631
1. Tof FLB Bus Acpt FLB6 Tgr NQ631
K
5-207 Sheet 2
End FLB Outgate

DIAGRAM 5-207. FLBB PRIORITY AND FLB OUTGATING (SHEET 1 OF 2)

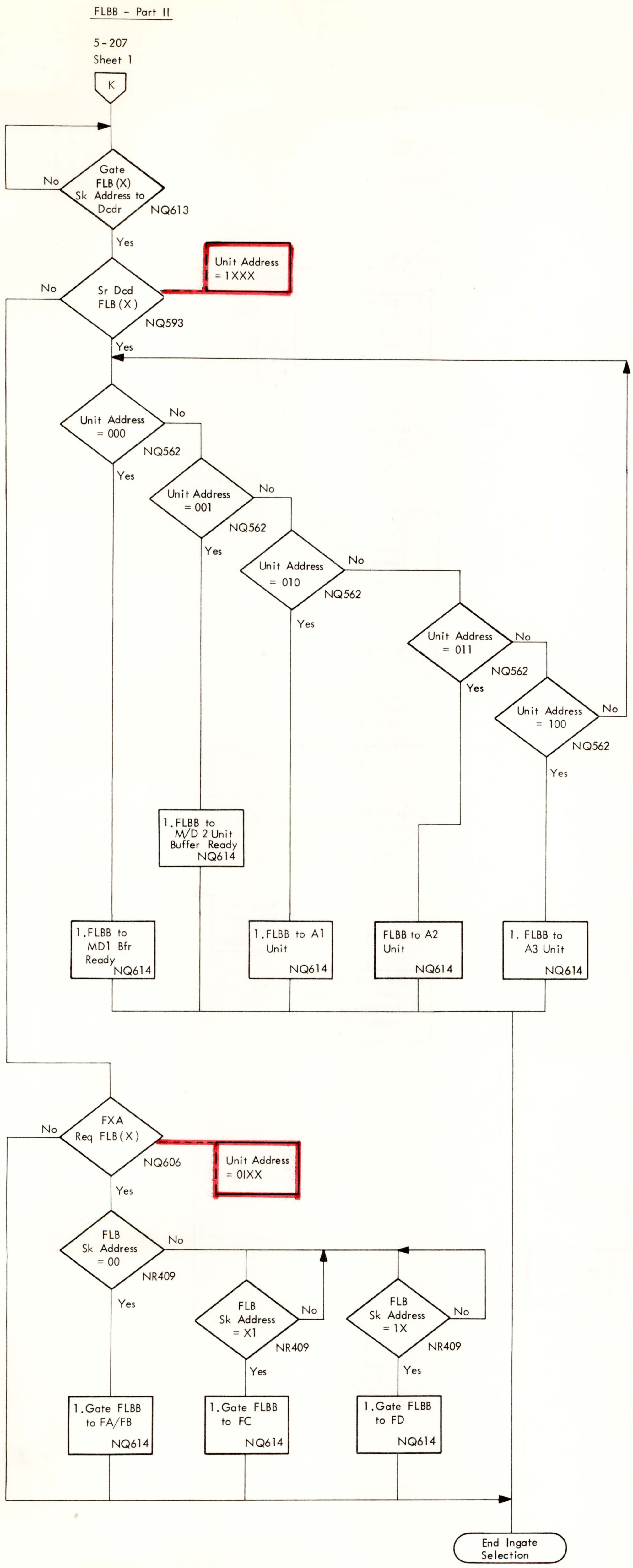


DIAGRAM 5-207. FLBB PRIORITY AND FLB OUTGATING (SHEET 2 OF 2)

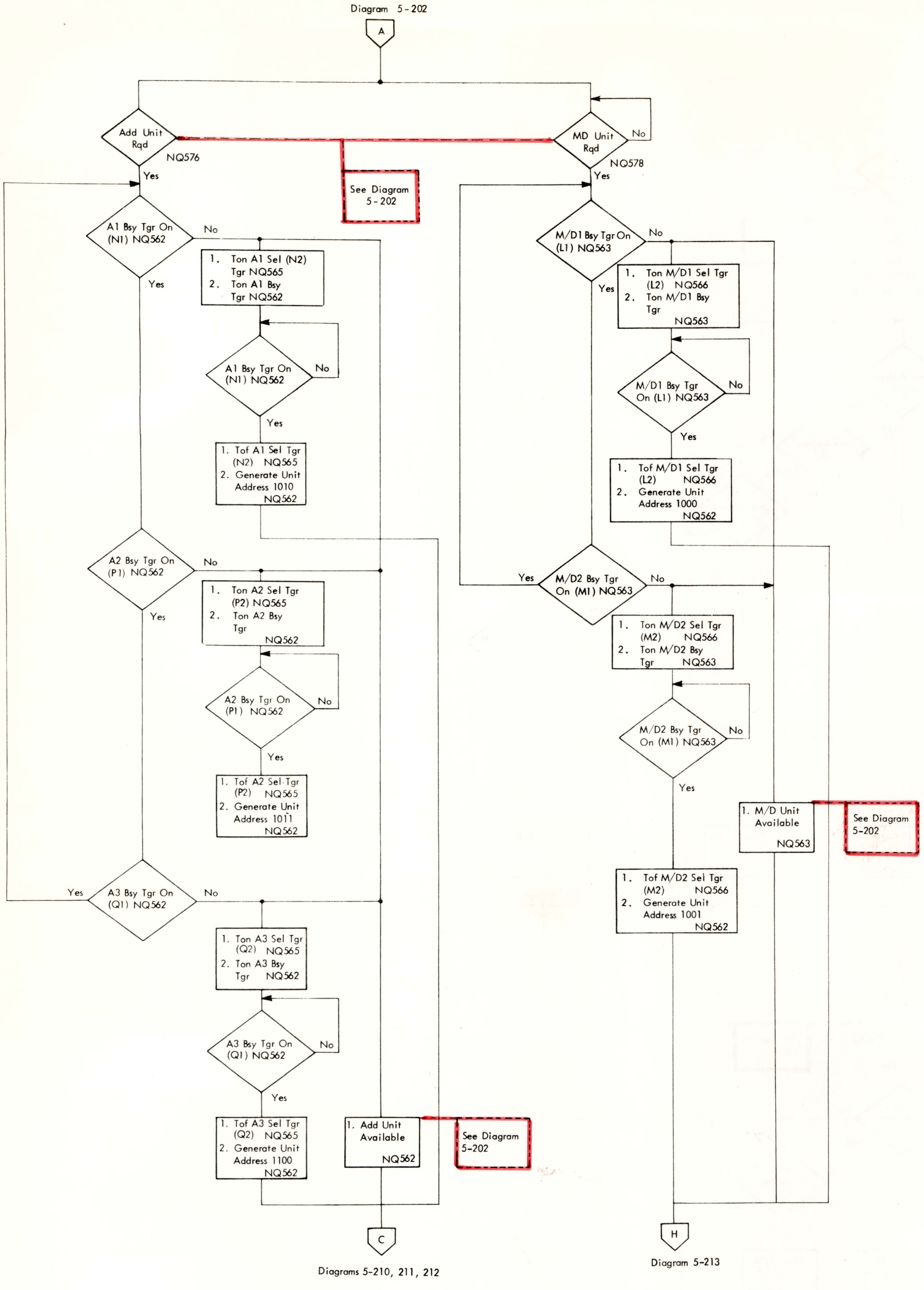


Diagram 5-202
A
Add Unit Rqd
NQ576
No
See Diagram 5-202
MD Unit Rqd
No
NQ578
Yes
Yes
A1 Bsy Tgr On (N1) NQ562
No
Yes
1. Ton A1 Sel (N2) Tgr NQ565
2. Ton A1 Bsy Tgr NQ562
A1 Bsy Tgr On (N1) NQ562
No
Yes
1. Tof A1 Sel Tgr (N2) NQ565
2. Generate Unit Address 1010 NQ562
A2 Bsy Tgr On (P1) NQ562
No
Yes
1. Ton A2 Sel Tgr (P2) NQ565
2. Ton A2 Bsy Tgr NQ562
A2 Bsy Tgr On (P1) NQ562
No
Yes
1. Tof A2 Sel Tgr (P2) NQ565
2. Generate Unit Address 1011 NQ562
Yes
A3 Bsy Tgr On (Q1) NQ562
No
1. Ton A3 Sel Tgr (Q2) NQ565
2. Ton A3 Bsy Tgr NQ562
A3 Bsy Tgr On (Q1) NQ562
No
Yes
1. Tof A3 Sel Tgr (Q2) NQ565
2. Generate Unit Address 1100 NQ562
1. Add Unit Available NQ562
See Diagram 5-202
C
Diagrams 5-210, 211, 212
M/D1 Bsy Tgr On (L1) NQ563
No
Yes
1. Ton M/D1 Sel Tgr (L2) NQ566
2. Ton M/D1 Bsy Tgr NQ563
M/D1 Bsy Tgr On (L1) NQ563
No
Yes
1. Tof M/D1 Sel Tgr (L2) NQ566
2. Generate Unit Address 1000 NQ562
Yes
M/D2 Bsy Tgr On (M1) NQ563
No
1. Ton M/D2 Sel Tgr (M2) NQ566
2. Ton M/D2 Bsy Tgr NQ563
M/D2 Bsy Tgr On (M1) NQ563
No
Yes
1. M/D Unit Available NQ563
See Diagram 5-202
1. Tof M/D2 Sel Tgr (M2) NQ566
2. Generate Unit Address 1001 NQ562
H
Diagram 5-213

DIAGRAM 5-208. UNIT SELECTION

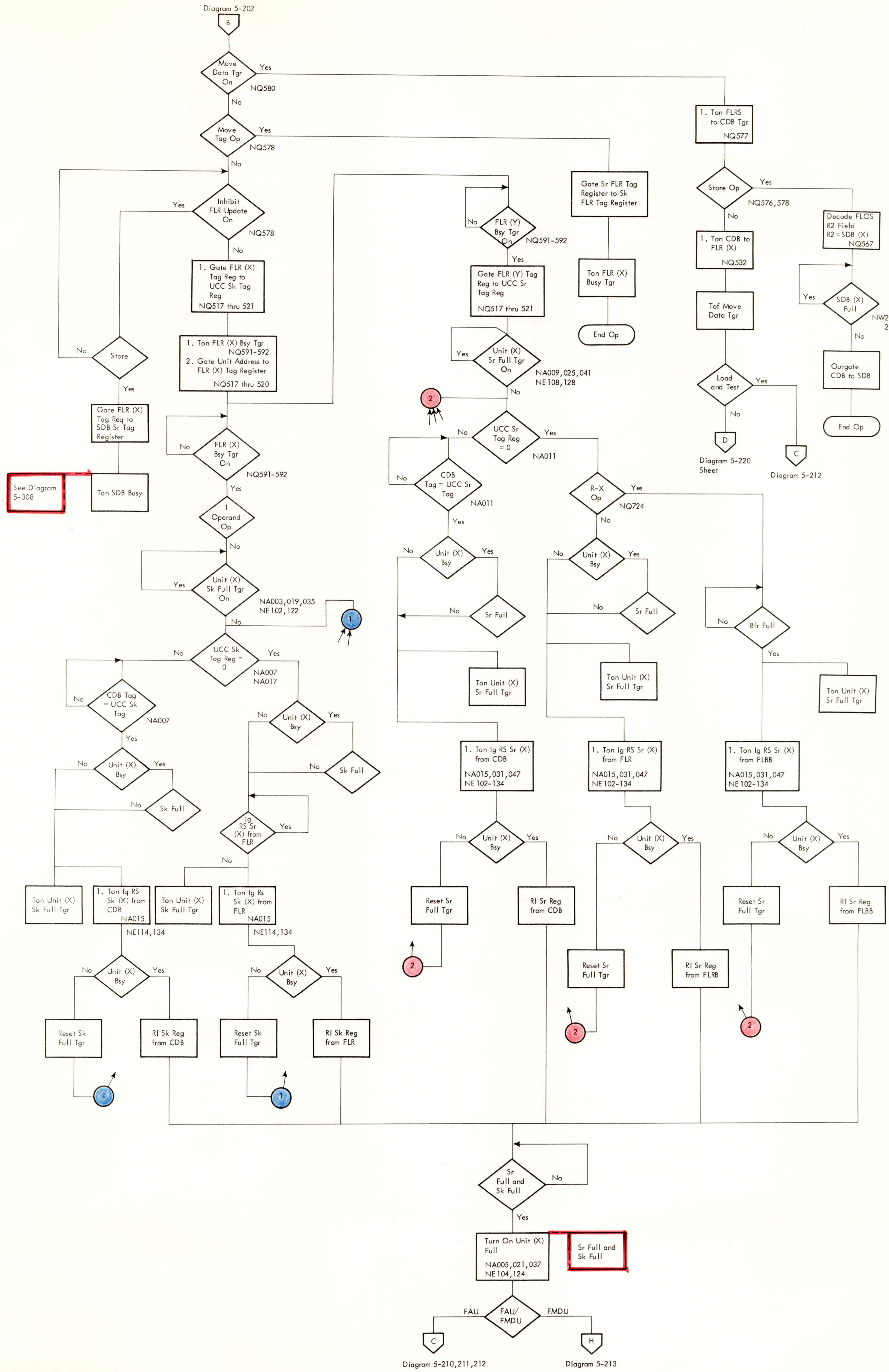


DIAGRAM 5-209. RESERVATION STATION INGATING AND TAG FORWARDING

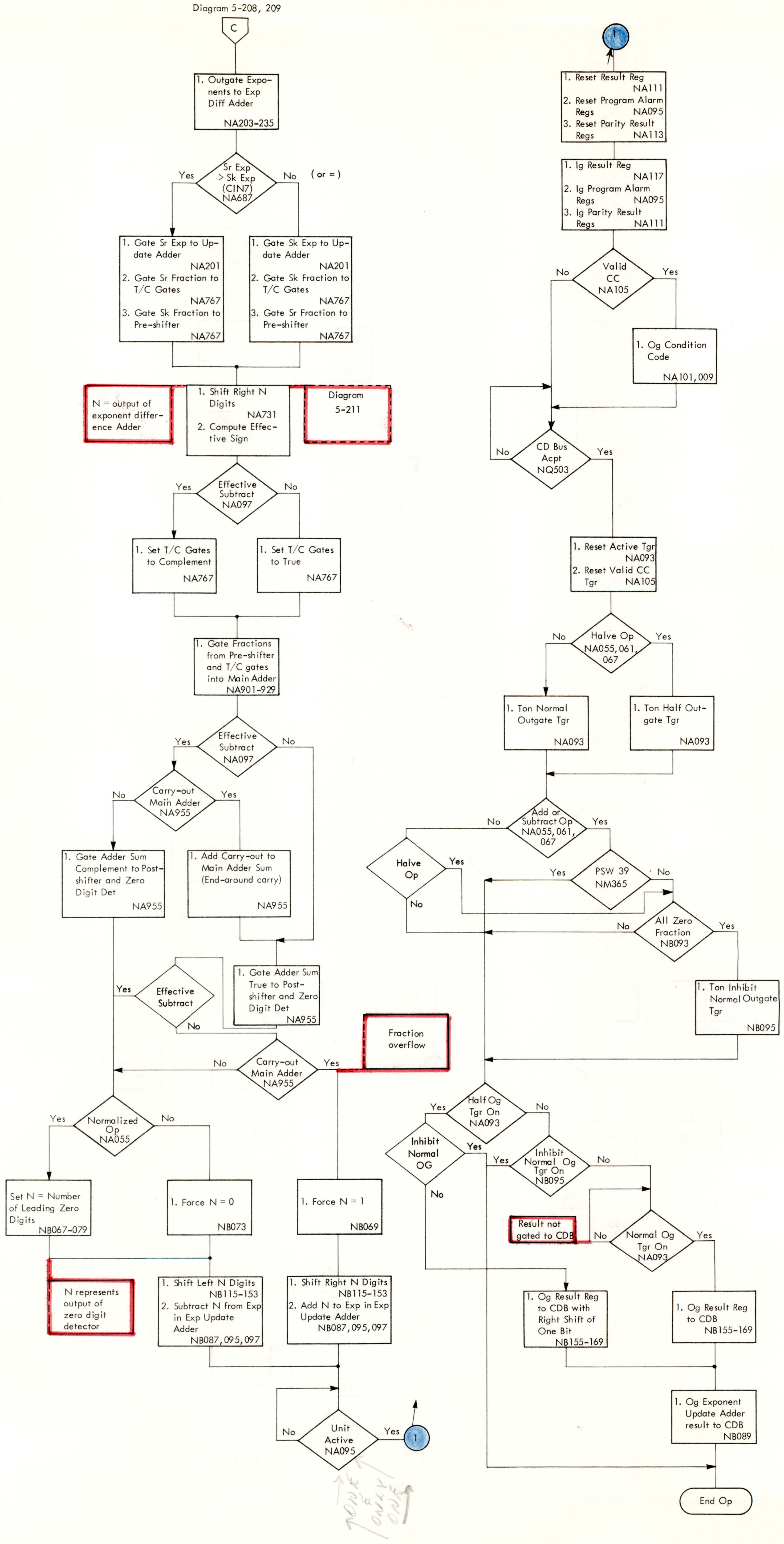


Diagram 5-208, 209
C
1. Outgate Exponents to Exp Diff Adder NA203-235
Sr Exp > Sk Exp (CIN7) NA687
Yes
No
(or =)
1. Gate Sr Exp to Update Adder NA201
2. Gate Sr Fraction to T/C Gates NA767
3. Gate Sk Fraction to Pre-shifter NA767
1. Gate Sk Exp to Update Adder NA201
2. Gate Sk Fraction to T/C Gates NA767
3. Gate Sr Fraction to Pre-shifter NA767
N = output of exponent difference Adder
1. Shift Right N Digits NA731
2. Compute Effective Sign
Diagram 5-211
Effective Subtract NA097
Yes
No
1. Set T/C Gates to Complement NA767
1. Set T/C Gates to True NA767
1. Gate Fractions from Pre-shifter and T/C gates into Main Adder NA901-929
Effective Subtract NA097
Yes
No
Carry-out Main Adder NA955
No
Yes
1. Gate Adder Sum Complement to Post-shifter and Zero Digit Det NA955
1. Add Carry-out to Main Adder Sum (End-around carry) NA955
Effective Subtract
Yes
No
1. Gate Adder Sum True to Post-shifter and Zero Digit Det NA955
Fraction overflow
Carry-out Main Adder NA955
No
Yes
Normalized Op NA055
Yes
No
Set N = Number of Leading Zero Digits NB067-079
1. Force N = 0 NB073
1. Force N = 1 NB069
N represents output of zero digit detector
1. Shift Left N Digits NB115-153
2. Subtract N from Exp in Exp Update Adder NB087, 095, 097
1. Shift Right N Digits NB115-153
2. Add N to Exp in Exp Update Adder NB087, 095, 097
Unit Active NA095
No
Yes
1
TONE & ONLY ONE
1
1. Reset Result Reg NA111
2. Reset Program Alarm Regs NA095
3. Reset Parity Result Regs NA113
1. Ig Result Reg NA117
2. Ig Program Alarm Regs NA095
3. Ig Parity Result Regs NA111
Valid CC NA105
No
Yes
1. Og Condition Code NA101, 009
CD Bus Acpt NQ503
No
Yes
1. Reset Active Tgr NA093
2. Reset Valid CC Tgr NA105
Halve Op NA055, 061, 067
No
Yes
1. Ton Normal Outgate Tgr NA093
1. Ton Half Outgate Tgr NA093
Add or Subtract Op NA055, 061, 067
No
Yes
Halve Op
Yes
No
PSW 39 NM365
Yes
No
All Zero Fraction NB093
No
Yes
1. Ton Inhibit Normal Outgate Tgr NB095
Half Og Tgr On NA093
Yes
No
Inhibit Normal OG
Yes
No
Inhibit Normal Og Tgr On NB095
Yes
No
Result not gated to CDB
Normal Og Tgr On NA093
No
Yes
1. Og Result Reg to CDB with Right Shift of One Bit NB155-169
1. Og Result Reg to CDB NB155-169
1. Og Exponent Update Adder result to CDB NB089
End Op

DIAGRAM 5-210. FAU EXECUTION - FRACTION AND EXPONENT

Diagrams 5-208,209

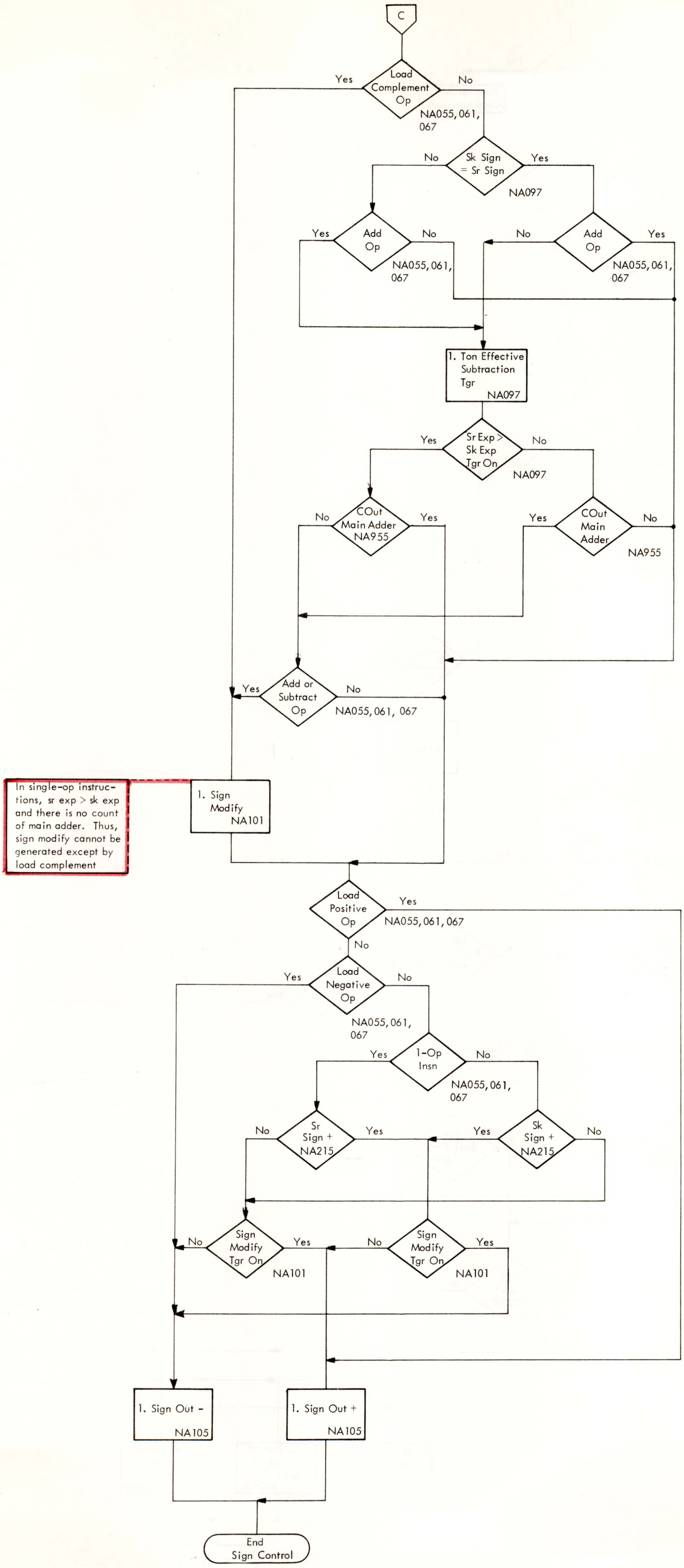


DIAGRAM 5-211. FAU SIGN CONTROL

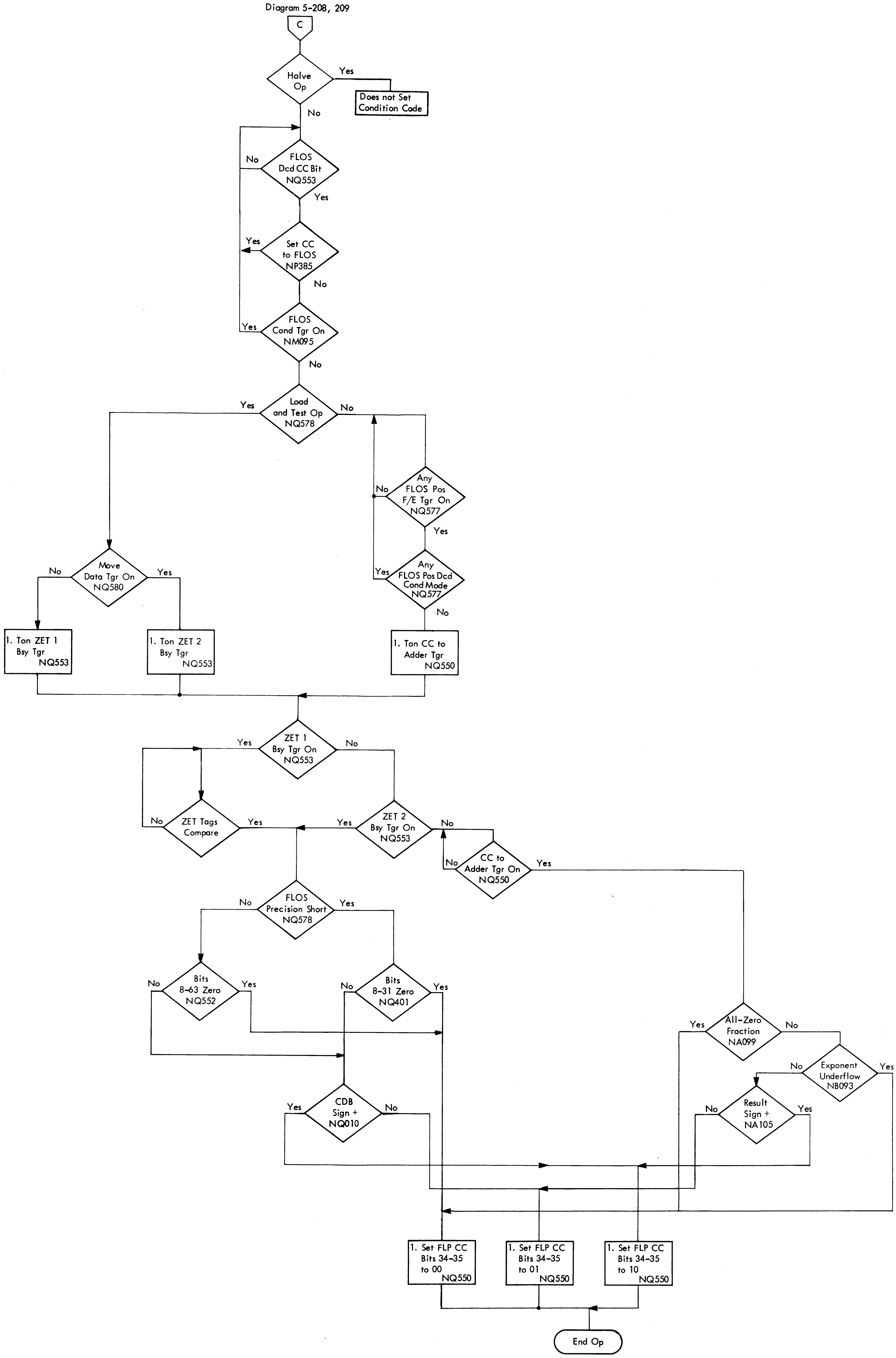


DIAGRAM 5-212. FAU AND ZET CONDITION CODE CONTROL

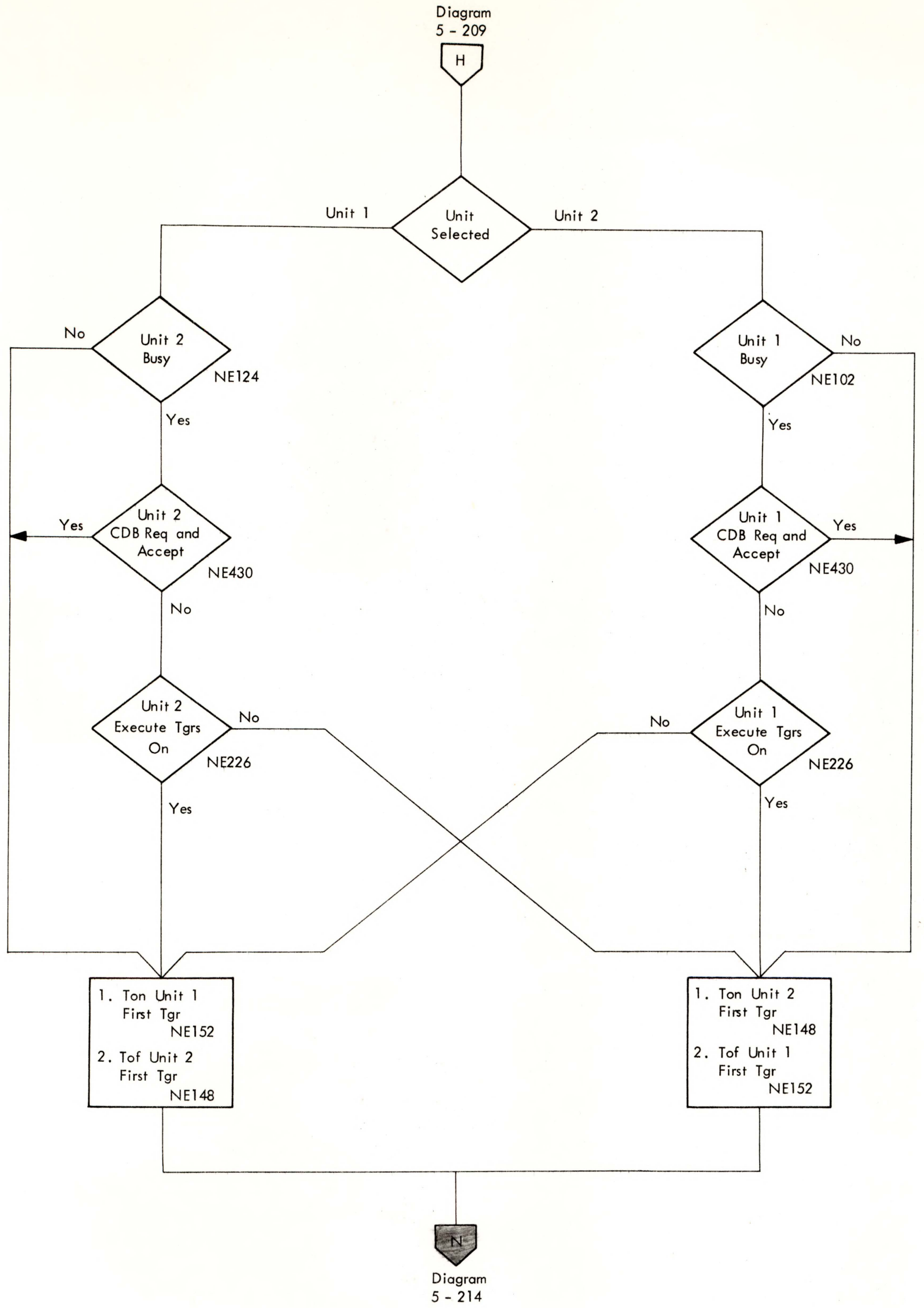


DIAGRAM 5-213. FMDU UNIT FIRST SELECTION

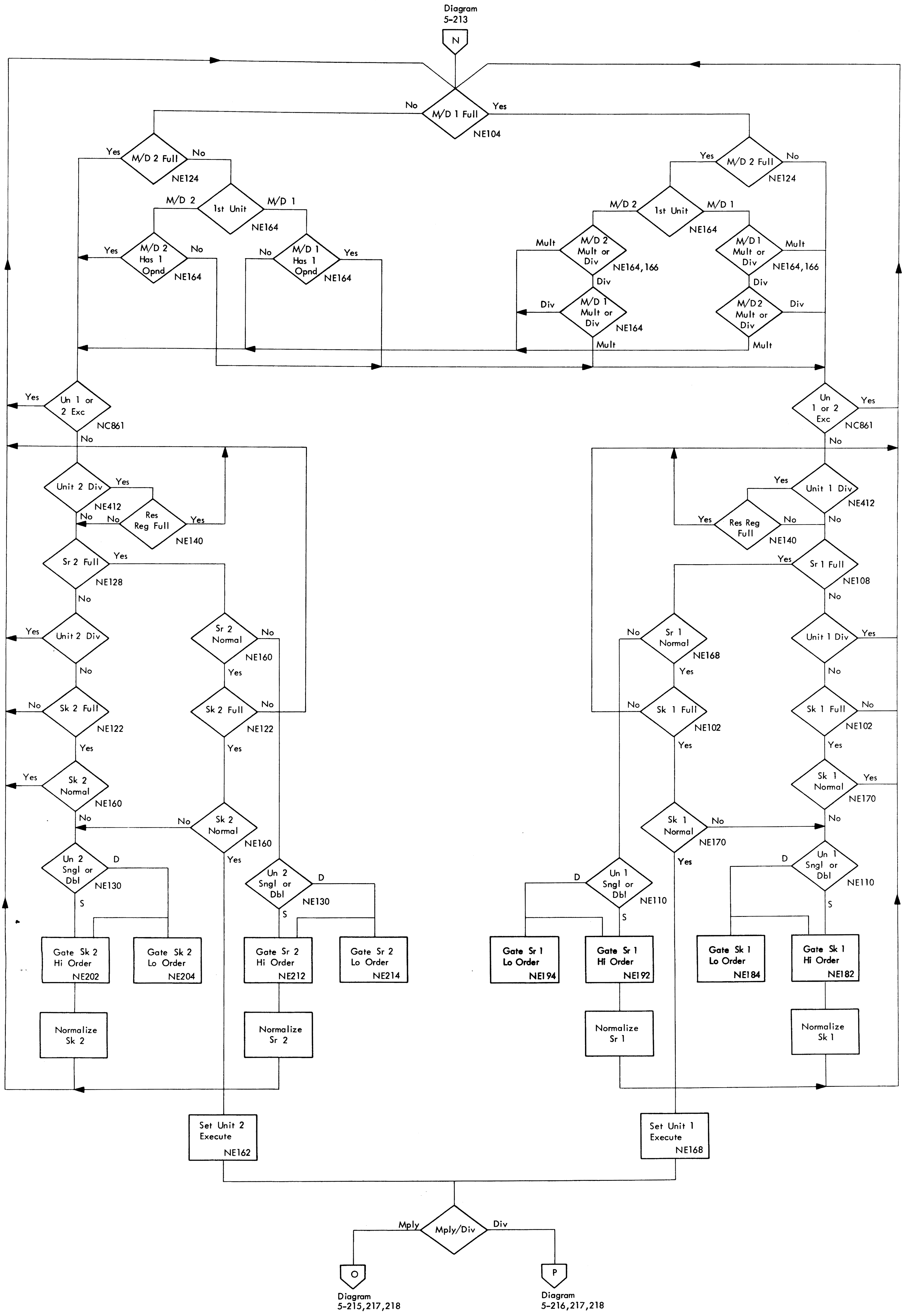


DIAGRAM 5-214. FMDU NORMALIZE CONTROL

Arithmetic Sequencing and Result Outgate Controls

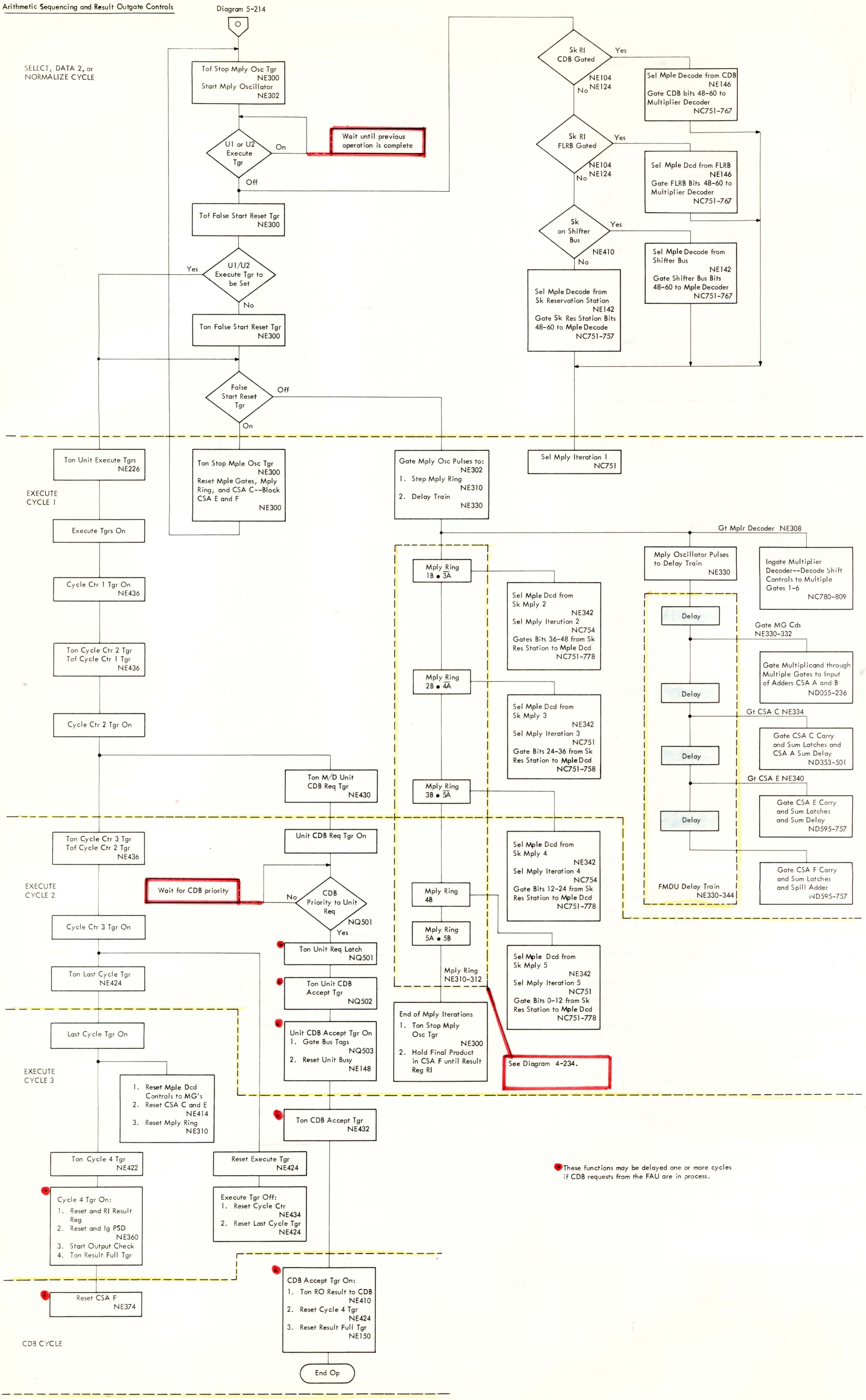
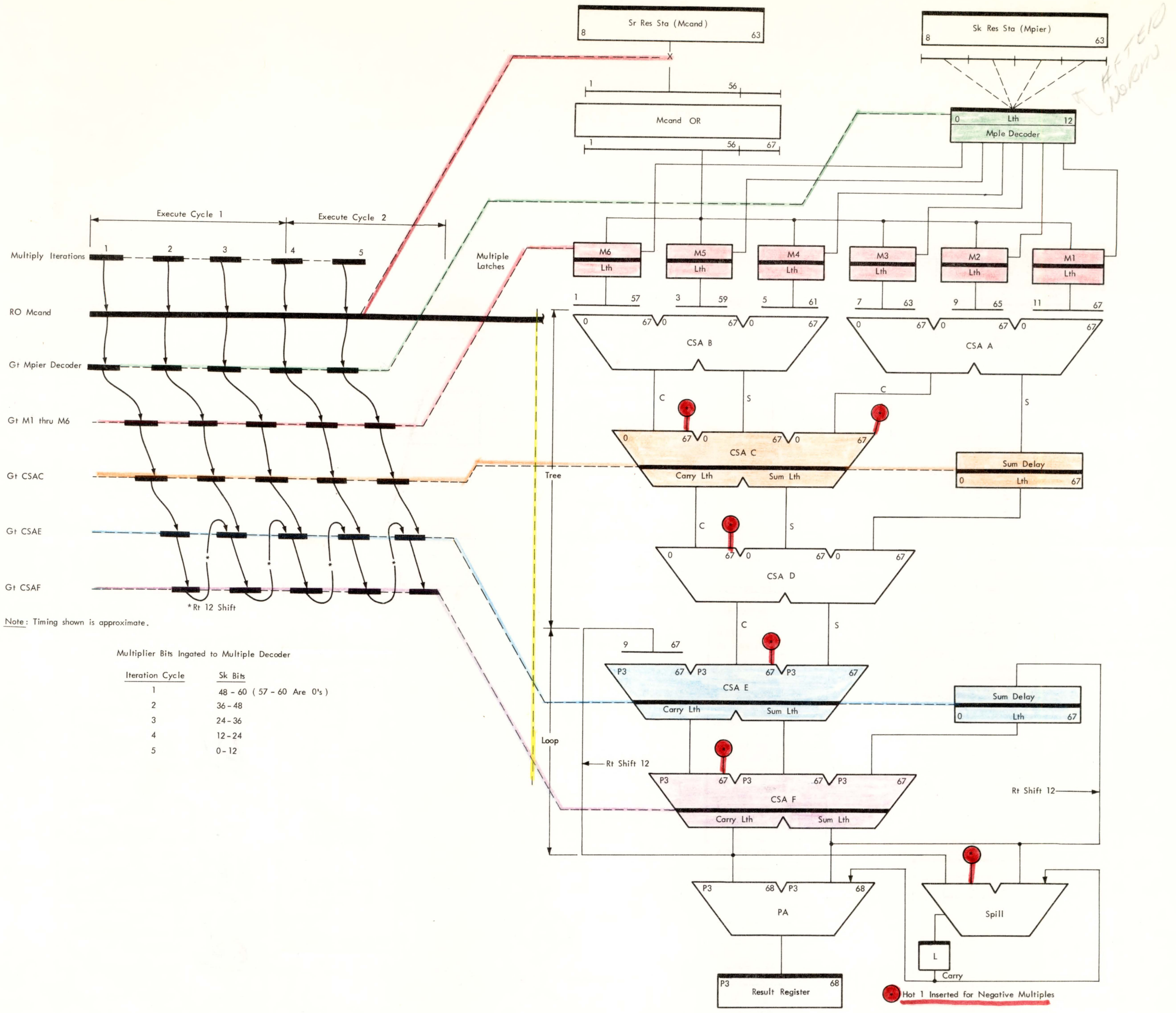


Diagram 5-214
O
SELECT, DATA 2, or NORMALIZE CYCLE
Tof Stop Mply Osc Tgr NE300
Start Mply Oscillator NE302
U1 or U2 Execute Tgr
On
Off
Wait until previous operation is complete
Tof False Start Reset Tgr NE300
U1/U2 Execute Tgr to be Set
Yes
No
Ton False Start Reset Tgr NE300
False Start Reset Tgr
Off
On
Sk RI CDB Gated
Yes
No
NE104 NE124
Sel Mple Decode from CDB NE146
Gate CDB bits 48-60 to Multiplier Decoder NC751-767
Sk RI FLRB Gated
Yes
No
NE104 NE124
Sel Mple Dcd from FLRB NE146
Gate FLRB Bits 48-60 to Multiplier Decoder NC751-767
Sk on Shifter Bus
Yes
No
NE410
Sel Mple Decode from Shifter Bus NE142
Gate Shifter Bus Bits 48-60 to Mple Decoder NC751-767
Sel Mple Decode from Sk Reservation Station NE142
Gate Sk Res Station Bits 48-60 to Mple Decode NC751-757
EXECUTE CYCLE 1
Ton Unit Execute Tgrs NE226
Ton Stop Mple Osc Tgr NE300
Reset Mple Gates, Mply Ring, and CSA C--Block CSA E and F NE300
Gate Mply Osc Pulses to: NE302
1. Step Mply Ring NE310
2. Delay Train NE330
Sel Mply Iteration 1 NC751
Execute Tgrs On
Cycle Ctr 1 Tgr On NE436
Ton Cycle Ctr 2 Tgr
Tof Cycle Ctr 1 Tgr NE436
Cycle Ctr 2 Tgr On
Ton M/D Unit CDB Req Tgr NE430
Gt Mplr Decoder NE308
Mply Ring 1B • 3A
Sel Mple Dcd from Sk Mply 2 NE342
Sel Mply Iteration 2 NC754
Gates Bits 36-48 from Sk Res Station to Mple Dcd NC751-778
Mply Ring 2B • 4A
Sel Mple Dcd from Sk Mply 3 NE342
Sel Mply Iteration 3 NC751
Gate Bits 24-36 from Sk Res Station to Mple Dcd NC751-758
Mply Ring 3B • 5A
Sel Mple Dcd from Sk Mply 4 NE342
Sel Mply Iteration 4 NC754
Gate Bits 12-24 from Sk Res Station to Mple Dcd NC751-778
Mply Ring 4B
Mply Ring 5A • 5B
Mply Ring NE310-312
Sel Mple Dcd from Sk Mply 5 NE342
Sel Mply Iteration 5 NC751
Gate Bits 0-12 from Sk Res Station to Mple Dcd NC751-778
End of Mply Iterations
1. Ton Stop Mply Osc Tgr NE300
2. Hold Final Product in CSA F until Result Reg RI
See Diagram 4-234.
Mply Oscillator Pulses to Delay Train NE330
Ingate Multiplier Decoder--Decode Shift Controls to Multiple Gates 1-6 NC780-809
Delay
Gate MG Cds NE330-332
Gate Multiplicand through Multiple Gates to Input of Adders CSA A and B ND055-236
Delay
Gt CSA C NE334
Gate CSA C Carry and Sum Latches and CSA A Sum Delay ND353-501
Delay
Gt CSA E NE340
Gate CSA E Carry and Sum Latches and Sum Delay ND595-757
Delay
FMDU Delay Train NE330-344
Gate CSA F Carry and Sum Latches and Spill Adder ND595-757
EXECUTE CYCLE 2
Ton Cycle Ctr 3 Tgr
Tof Cycle Ctr 2 Tgr NE436
Unit CDB Req Tgr On
Wait for CDB priority
No
CDB Priority to Unit Req NQ501
Yes
Cycle Ctr 3 Tgr On
Ton Unit Req Latch NQ501
Ton Last Cycle Tgr NE424
Ton Unit CDB Accept Tgr NQ502
Last Cycle Tgr On
Unit CDB Accept Tgr On
1. Gate Bus Tags NQ503
2. Reset Unit Busy NE148
EXECUTE CYCLE 3
1. Reset Mple Dcd Controls to MG's
2. Reset CSA C and E NE414
3. Reset Mply Ring NE310
Ton CDB Accept Tgr NE432
Ton Cycle 4 Tgr NE422
Reset Execute Tgr NE424
Cycle 4 Tgr On:
1. Reset and RI Result Reg
2. Reset and Ig PSD NE360
3. Start Output Check
4. Ton Result Full Tgr
Execute Tgr Off:
1. Reset Cycle Ctr NE434
2. Reset Last Cycle Tgr NE424
Reset CSA F NE374
CDB Accept Tgr On:
1. Ton RO Result to CDB NE410
2. Reset Cycle 4 Tgr NE424
3. Reset Result Full Tgr NE150
CDB CYCLE
End Op
These functions may be delayed one or more cycles if CDB requests from the FAU are in process.

DIAGRAM 5-215. MULTIPLY EXECUTION (SHEET 1 OF 2)

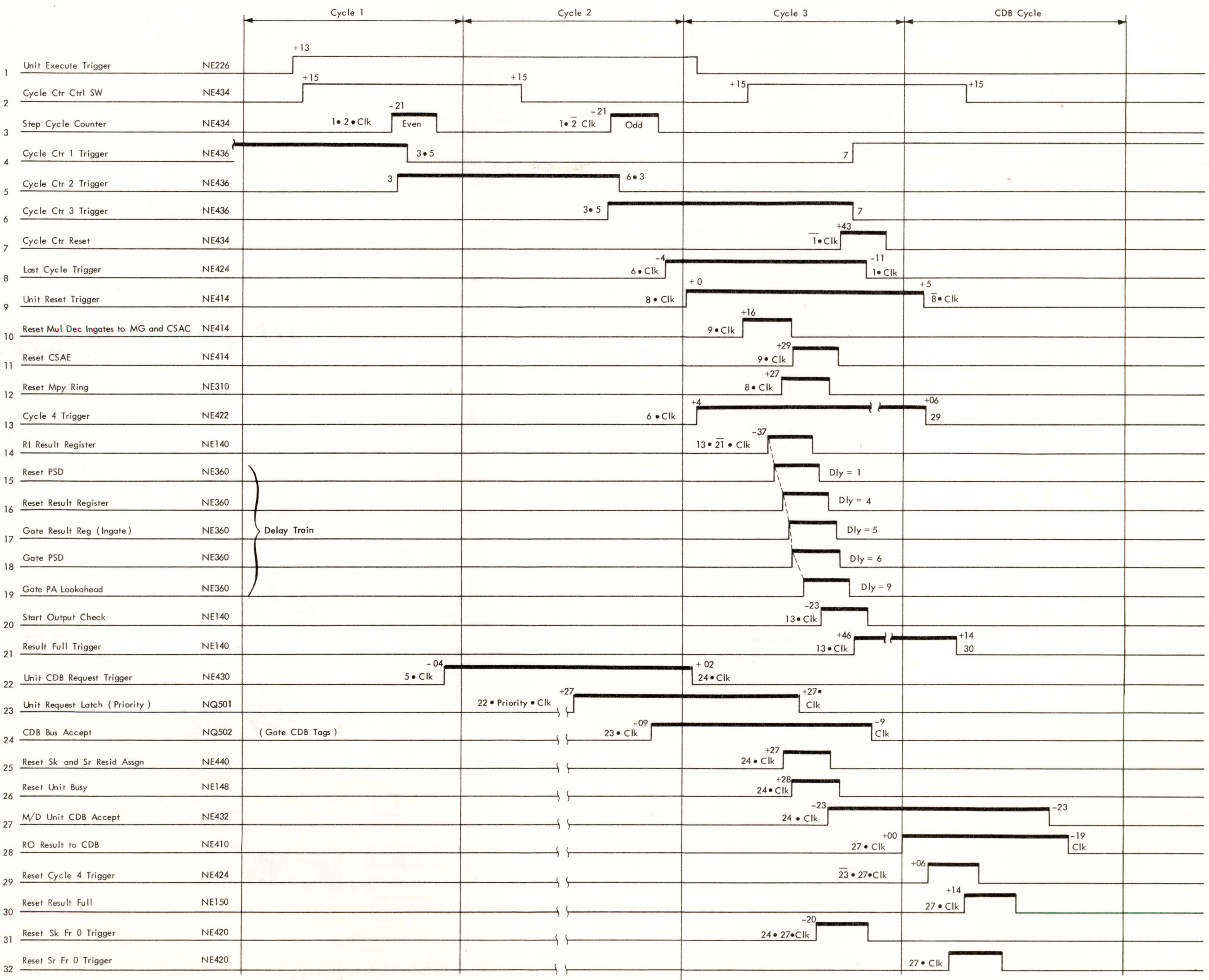


Note: Timing shown is approximate.

Multiplier Bits Ingated to Multiple Decoder

Iteration Cycle	Sk Bits
1	48 - 60 (57 - 60 Are 0's)
2	36 - 48
3	24 - 36
4	12 - 24
5	0 - 12

Multiply Sequencing and Outgate Controls



Cycle 1
Cycle 2
Cycle 3
CDB Cycle
1 Unit Execute Trigger NE226
+13
2 Cycle Ctr Ctrl SW NE434
+15
+15
+15
+15
3 Step Cycle Counter NE434
1 • 2 • Clk
-21
Even
1 • $\overline{2}$ Clk
-21
Odd
4 Cycle Ctr 1 Trigger NE436
3 • 5
7
5 Cycle Ctr 2 Trigger NE436
3
6 • 3
6 Cycle Ctr 3 Trigger NE436
3 • 5
7
7 Cycle Ctr Reset NE434
+43
$\overline{1}$ • Clk
8 Last Cycle Trigger NE424
-4
6 • Clk
-11
1 • Clk
9 Unit Reset Trigger NE414
+0
8 • Clk
+5
$\overline{8}$ • Clk
10 Reset Mul Dec Ingates to MG and CSAC NE414
+16
9 • Clk
11 Reset CSAE NE414
+29
9 • Clk
12 Reset Mpy Ring NE310
+27
8 • Clk
13 Cycle 4 Trigger NE422
+4
6 • Clk
+06
29
14 RI Result Register NE140
-37
13 • $\overline{21}$ • Clk
15 Reset PSD NE360
Dly = 1
16 Reset Result Register NE360
Dly = 4
17 Gate Result Reg (Ingate) NE360
Delay Train
Dly = 5
18 Gate PSD NE360
Dly = 6
19 Gate PA Lookahead NE360
Dly = 9
20 Start Output Check NE140
-23
13 • Clk
21 Result Full Trigger NE140
+46
13 • Clk
+14
30
22 Unit CDB Request Trigger NE430
-04
5 • Clk
+02
24 • Clk
23 Unit Request Latch (Priority) NQ501
22 • Priority • Clk
+27
+27 •
Clk
24 CDB Bus Accept NQ502
(Gate CDB Tags)
-09
23 • Clk
-9
Clk
25 Reset Sk and Sr Resid Assgn NE440
+27
24 • Clk
26 Reset Unit Busy NE148
+28
24 • Clk
27 M/D Unit CDB Accept NE432
-23
24 • Clk
-23
28 RO Result to CDB NE410
+00
27 • Clk
-19
Clk
29 Reset Cycle 4 Trigger NE424
+06
$\overline{23}$ • 27 • Clk
30 Reset Result Full NE150
+14
27 • Clk
31 Reset Sk Fr 0 Trigger NE420
-20
24 • 27 • Clk
32 Reset Sr Fr 0 Trigger NE420
27 • Clk

Note: The timing shown is approximate. For each signal line, the originating clock timing is indicated; however, because of inherent circuit delays a given signal may occur several nanoseconds later.

DIAGRAM 5-215. MULTIPLY EXECUTION (SHEET 2 OF 2)

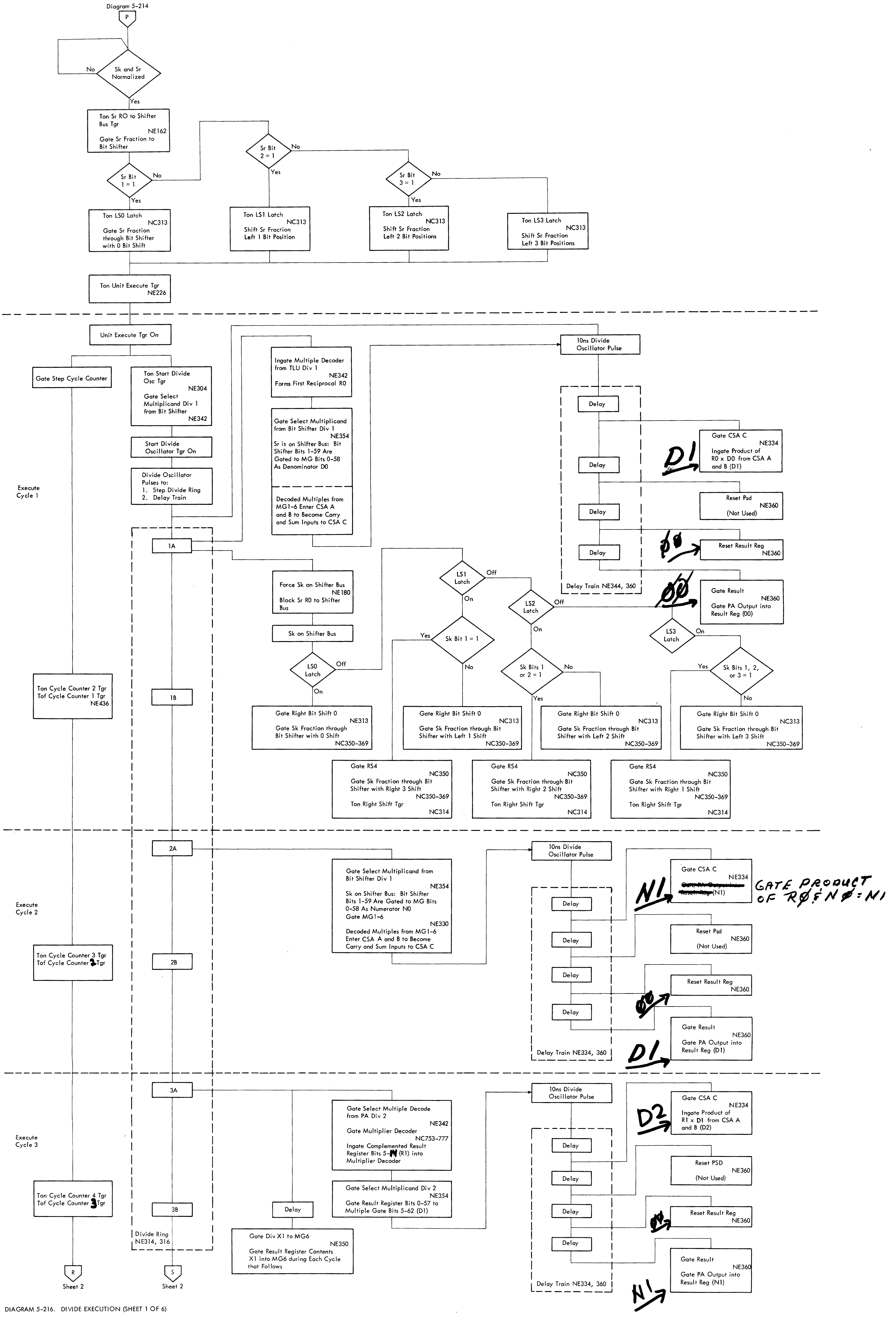


Diagram 5-214
P
Sk and Sr Normalized
No
Yes
Ton Sr RO to Shifter Bus Tgr
NE162
Gate Sr Fraction to Bit Shifter
Sr Bit 1 = 1
No
Yes
Sr Bit 2 = 1
No
Yes
Sr Bit 3 = 1
No
Yes
Ton LS0 Latch
NC313
Gate Sr Fraction through Bit Shifter with 0 Bit Shift
Ton LS1 Latch
NC313
Shift Sr Fraction Left 1 Bit Position
Ton LS2 Latch
NC313
Shift Sr Fraction Left 2 Bit Positions
Ton LS3 Latch
NC313
Shift Sr Fraction Left 3 Bit Positions
Ton Unit Execute Tgr
NE226
Unit Execute Tgr On
Gate Step Cycle Counter
Ton Start Divide Osc Tgr
NE304
Gate Select Multiplicand Div 1 from Bit Shifter
NE342
Start Divide Oscillator Tgr On
Divide Oscillator Pulses to:
1. Step Divide Ring
2. Delay Train
Execute Cycle 1
1A
Ingate Multiple Decoder from TLU Div 1
NE342
Forms First Reciprocal R0
Gate Select Multiplicand from Bit Shifter Div 1
NE354
Sr is on Shifter Bus: Bit Shifter Bits 1-59 Are Gated to MG Bits 0-58 As Denominator D0
Decoded Multiples from MG1-6 Enter CSA A and B to Become Carry and Sum Inputs to CSA C
10ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE344, 360
Gate CSA C
NE334
Ingate Product of R0 x D0 from CSA A and B (D1)
D1
Reset Psd
NE360
(Not Used)
Reset Result Reg
NE360
ØØ
Gate Result
NE360
Gate PA Output into Result Reg (00)
ØØ
Force Sk on Shifter Bus
NE180
Block Sr R0 to Shifter Bus
Sk on Shifter Bus
LS0 Latch
Off
On
LS1 Latch
Off
On
LS2 Latch
Off
On
LS3 Latch
On
Sk Bit 1 = 1
Yes
No
Sk Bits 1 or 2 = 1
No
Yes
Sk Bits 1, 2, or 3 = 1
Yes
No
Ton Cycle Counter 2 Tgr
Tof Cycle Counter 1 Tgr
NE436
1B
Gate Right Bit Shift 0
NE313
Gate Sk Fraction through Bit Shifter with 0 Shift
NC350-369
Gate Right Bit Shift 0
NC313
Gate Sk Fraction through Bit Shifter with Left 1 Shift
NC350-369
Gate Right Bit Shift 0
NC313
Gate Sk Fraction through Bit Shifter with Left 2 Shift
NC350-369
Gate Right Bit Shift 0
NC313
Gate Sk Fraction through Bit Shifter with Left 3 Shift
NC350-369
Gate RS4
NC350
Gate Sk Fraction through Bit Shifter with Right 3 Shift
NC350-369
Ton Right Shift Tgr
NC314
Gate RS4
NC350
Gate Sk Fraction through Bit Shifter with Right 2 Shift
NC350-369
Ton Right Shift Tgr
NC314
Gate RS4
NC350
Gate Sk Fraction through Bit Shifter with Right 1 Shift
NC350-369
Ton Right Shift Tgr
NC314
Execute Cycle 2
2A
Gate Select Multiplicand from Bit Shifter Div 1
NE354
Sk on Shifter Bus: Bit Shifter Bits 1-59 Are Gated to MG Bits 0-58 As Numerator N0
Gate MG1-6
NE330
Decoded Multiples from MG1-6 Enter CSA A and B to Become Carry and Sum Inputs to CSA C
10ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE334, 360
Gate CSA C
NE334
~~Gate PA Output into~~
~~Result Reg~~ (N1)
N1
GATE PRODUCT OF RØ & NØ = N1
Reset Psd
NE360
(Not Used)
Reset Result Reg
NE360
ØØ
Gate Result
NE360
Gate PA Output into Result Reg (D1)
D1
Ton Cycle Counter 3 Tgr
Tof Cycle Counter 2 Tgr
2B
Execute Cycle 3
3A
Gate Select Multiple Decode from PA Div 2
NE342
Gate Multiplier Decoder
NC753-777
Ingate Complemented Result Register Bits 5-N (R1) into Multiplier Decoder
Gate Select Multiplicand Div 2
NE354
Gate Result Register Bits 0-57 to Multiple Gate Bits 5-62 (D1)
Delay
Gate Div X1 to MG6
NE350
Gate Result Register Contents X1 into MG6 during Each Cycle that Follows
10ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE334, 360
Gate CSA C
NE334
Ingate Product of R1 x D1 from CSA A and B (D2)
D2
Reset PSD
NE360
(Not Used)
Reset Result Reg
NE360
ØØ
Gate Result
NE360
Gate PA Output into Result Reg (N1)
N1
Ton Cycle Counter 4 Tgr
Tof Cycle Counter 3 Tgr
3B
Divide Ring NE314, 316
R
Sheet 2
S
Sheet 2

DIAGRAM 5-216. DIVIDE EXECUTION (SHEET 1 OF 6)

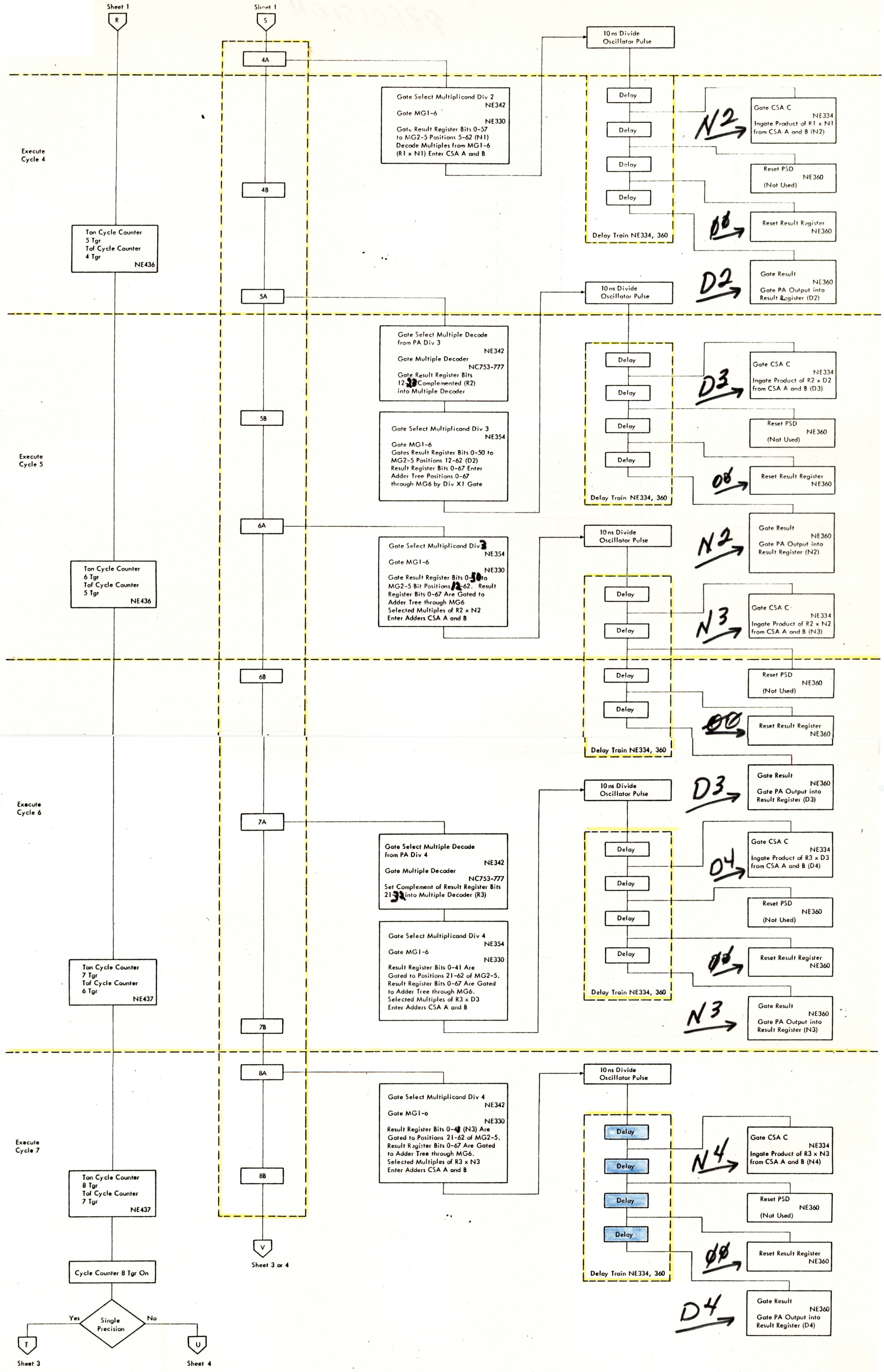

Sheet 1
R
Sheet 1
S
Execute Cycle 4
4A
Gate Select Multiplicand Div 2 NE342
Gate MG1-6 NE330
Gate Result Register Bits 0-57 to MG2-5 Positions 5-62 (N1) Decode Multiples from MG1-6 (R1 x N1) Enter CSA A and B
10 ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE334, 360
N2
Gate CSA C NE334 Ingate Product of R1 x N1 from CSA A and B (N2)
Reset PSD NE360 (Not Used)
Ø0
Reset Result Register NE360
D2
Gate Result NE360 Gate PA Output into Result Register (D2)
4B
Ton Cycle Counter 5 Tgr
Tof Cycle Counter 4 Tgr
NE436
5A
Execute Cycle 5
Gate Select Multiple Decode from PA Div 3 NE342
Gate Multiple Decoder NC753-777
Gate Result Register Bits 12-23 Complemented (R2) into Multiple Decoder
5B
Gate Select Multiplicand Div 3 NE354
Gate MG1-6
Gates Result Register Bits 0-50 to MG2-5 Positions 12-62 (D2)
Result Register Bits 0-67 Enter Adder Tree Positions 0-67 through MG6 by Div X1 Gate
10 ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE334, 360
D3
Gate CSA C NE334 Ingate Product of R2 x D2 from CSA A and B (D3)
Reset PSD NE360 (Not Used)
00
Reset Result Register NE360
N2
Gate Result NE360 Gate PA Output into Result Register (N2)
6A
Ton Cycle Counter 6 Tgr
Tof Cycle Counter 5 Tgr
NE436
Gate Select Multiplicand Div 3 NE354
Gate MG1-6 NE330
Gate Result Register Bits 0-50 to MG2-5 Bit Positions 12-62. Result Register Bits 0-67 Are Gated to Adder Tree through MG6 Selected Multiples of R2 x N2 Enter Adders CSA A and B
10 ns Divide Oscillator Pulse
Delay
Delay
N3
Gate CSA C NE334 Ingate Product of R2 x N2 from CSA A and B (N3)
6B
Delay
Delay
Reset PSD NE360 (Not Used)
ØØ
Reset Result Register NE360
Delay Train NE334, 360
Execute Cycle 6
D3
Gate Result NE360 Gate PA Output into Result Register (D3)
10 ns Divide Oscillator Pulse
7A
Gate Select Multiple Decode from PA Div 4 NE342
Gate Multiple Decoder NC753-777
Set Complement of Result Register Bits 21-32 into Multiple Decoder (R3)
Delay
Delay
Delay
Delay
Delay Train NE334, 360
D4
Gate CSA C NE334 Ingate Product of R3 x D3 from CSA A and B (D4)
Reset PSD NE360 (Not Used)
Ø0
Reset Result Register NE360
Gate Select Multiplicand Div 4 NE354
Gate MG1-6 NE330
Result Register Bits 0-41 Are Gated to Positions 21-62 of MG2-5. Result Register Bits 0-67 Are Gated to Adder Tree through MG6. Selected Multiples of R3 x D3 Enter Adders CSA A and B
Ton Cycle Counter 7 Tgr
Tof Cycle Counter 6 Tgr
NE437
N3
Gate Result NE360 Gate PA Output into Result Register (N3)
7B
8A
Execute Cycle 7
Gate Select Multiplicand Div 4 NE342
Gate MG1-6 NE330
Result Register Bits 0-41 (N3) Are Gated to Positions 21-62 of MG2-5. Result Register Bits 0-67 Are Gated to Adder Tree through MG6. Selected Multiples of R3 x N3 Enter Adders CSA A and B
10 ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay Train NE334, 360
N4
Gate CSA C NE334 Ingate Product of R3 x N3 from CSA A and B (N4)
Reset PSD NE360 (Not Used)
ØØ
Reset Result Register NE360
D4
Gate Result NE360 Gate PA Output into Result Register (D4)
8B
Ton Cycle Counter 8 Tgr
Tof Cycle Counter 7 Tgr
NE437
V
Sheet 3 or 4
Cycle Counter 8 Tgr On
Single Precision
Yes
No
T
Sheet 3
U
Sheet 4

DIAGRAM 5-216. DIVIDE EXECUTION (SHEET 2 OF 6)

SHORT PRECISION

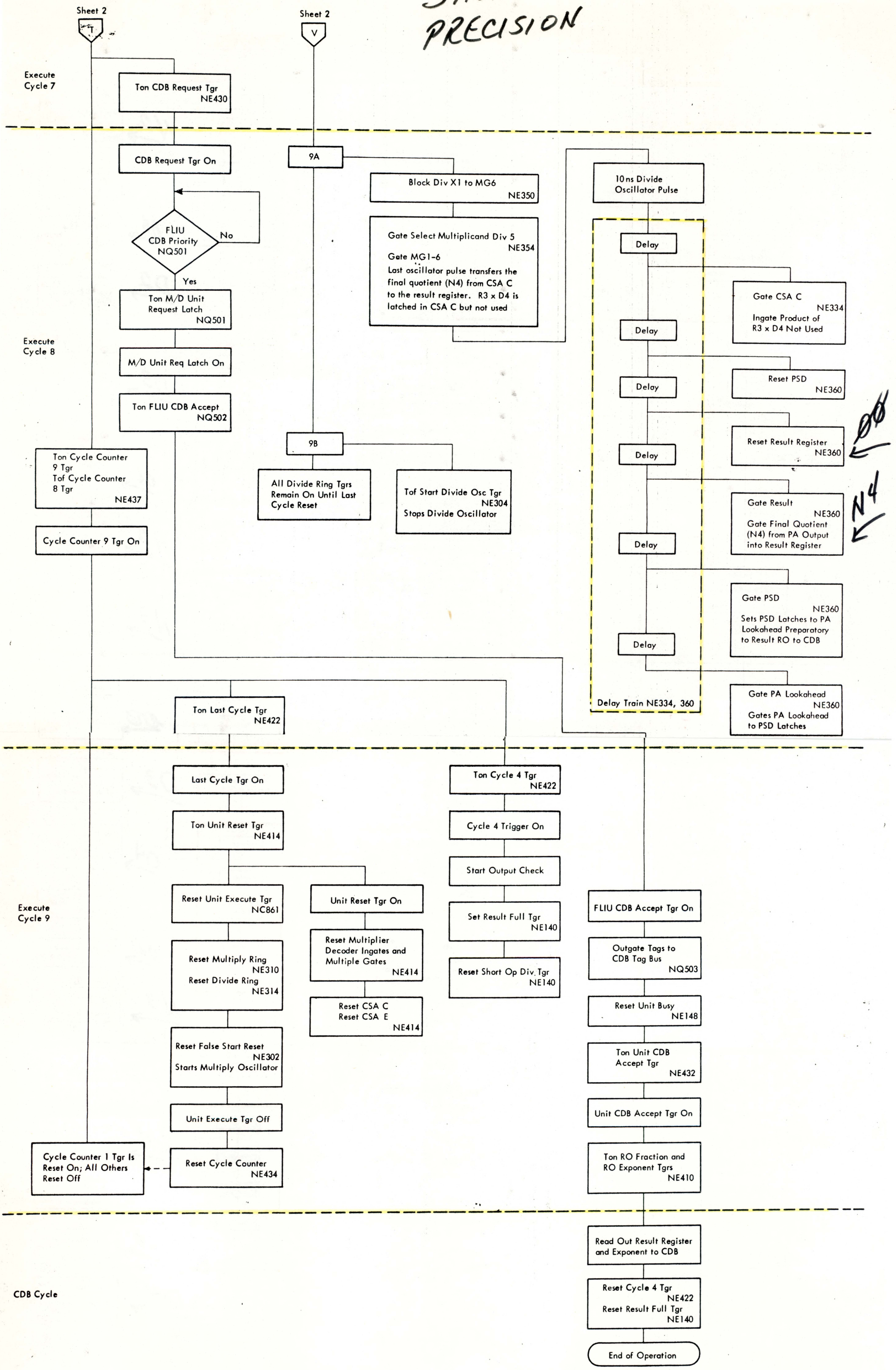

Sheet 2
T
Sheet 2
V
Execute Cycle 7
Ton CDB Request Tgr NE430
CDB Request Tgr On
FLIU CDB Priority NQ501
No
Yes
Ton M/D Unit Request Latch NQ501
Execute Cycle 8
M/D Unit Req Latch On
Ton FLIU CDB Accept NQ502
Ton Cycle Counter 9 Tgr Tof Cycle Counter 8 Tgr NE437
Cycle Counter 9 Tgr On
9A
Block Div X1 to MG6 NE350
Gate Select Multiplicand Div 5 NE354
Gate MG1-6
Last oscillator pulse transfers the final quotient (N4) from CSA C to the result register. R3 x D4 is latched in CSA C but not used
9B
All Divide Ring Tgrs Remain On Until Last Cycle Reset
Tof Start Divide Osc Tgr NE304 Stops Divide Oscillator
10 ns Divide Oscillator Pulse
Delay
Delay
Delay
Delay
Delay
Delay
Delay Train NE334, 360
Gate CSA C NE334 Ingate Product of R3 x D4 Not Used
Reset PSD NE360
Reset Result Register NE360
ØØ
Gate Result NE360 Gate Final Quotient (N4) from PA Output into Result Register
N4
Gate PSD NE360 Sets PSD Latches to PA Lookahead Preparatory to Result RO to CDB
Gate PA Lookahead NE360 Gates PA Lookahead to PSD Latches
Ton Last Cycle Tgr NE422
Execute Cycle 9
Last Cycle Tgr On
Ton Unit Reset Tgr NE414
Reset Unit Execute Tgr NC861
Reset Multiply Ring NE310 Reset Divide Ring NE314
Reset False Start Reset NE302 Starts Multiply Oscillator
Unit Execute Tgr Off
Reset Cycle Counter NE434
Cycle Counter 1 Tgr Is Reset On; All Others Reset Off
Unit Reset Tgr On
Reset Multiplier Decoder Ingates and Multiple Gates NE414
Reset CSA C Reset CSA E NE414
Ton Cycle 4 Tgr NE422
Cycle 4 Trigger On
Start Output Check
Set Result Full Tgr NE140
Reset Short Op Div. Tgr NE140
FLIU CDB Accept Tgr On
Outgate Tags to CDB Tag Bus NQ503
Reset Unit Busy NE148
Ton Unit CDB Accept Tgr NE432
Unit CDB Accept Tgr On
Ton RO Fraction and RO Exponent Tgrs NE410
CDB Cycle
Read Out Result Register and Exponent to CDB
Reset Cycle 4 Tgr NE422 Reset Result Full Tgr NE140
End of Operation

DIAGRAM 5-216. DIVIDE EXECUTION (SHEET 3 OF 6)

LONG PRECISION

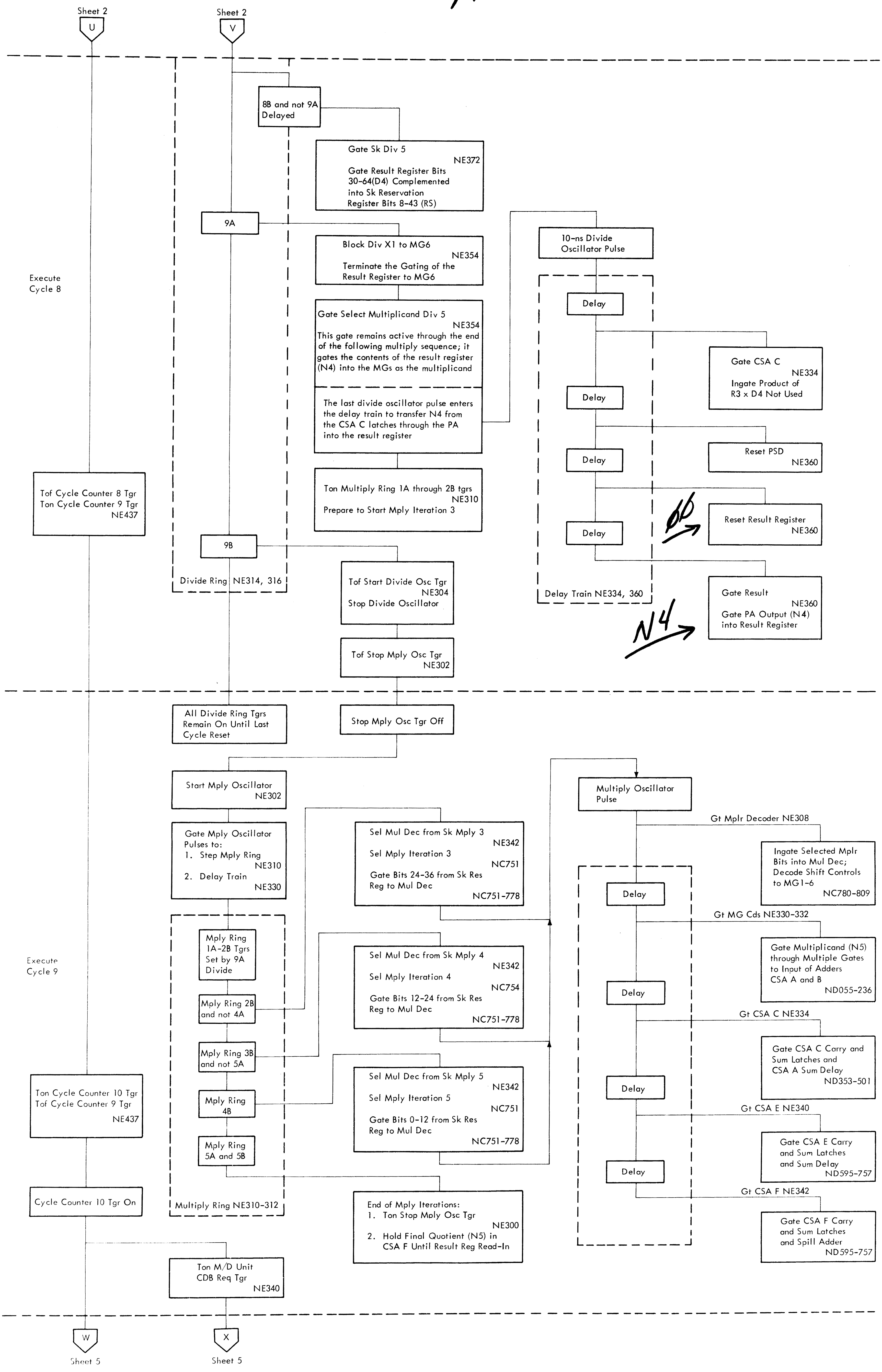
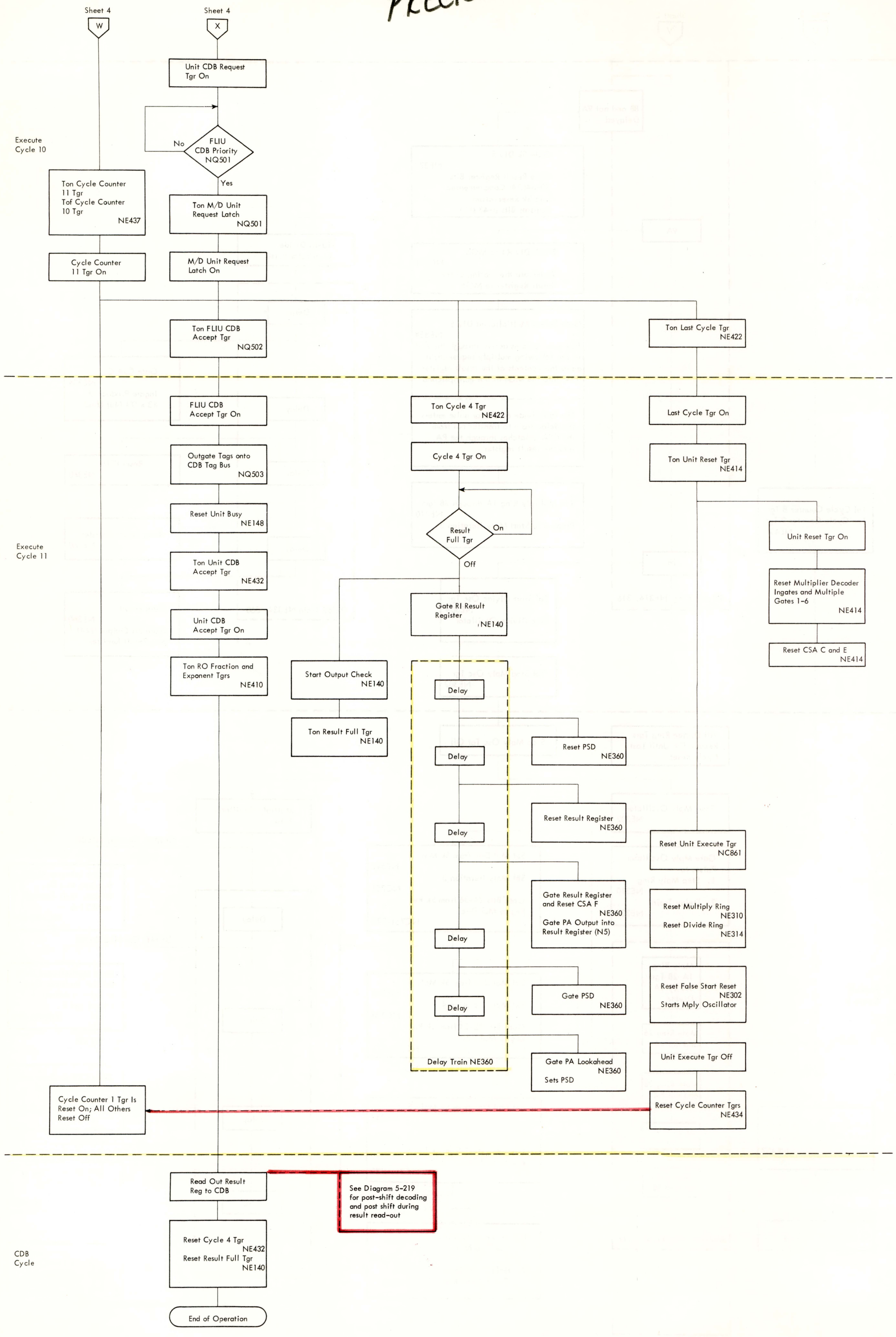

Sheet 2
U
Sheet 2
V
Execute Cycle 8
8B and not 9A Delayed
Gate Sk Div 5 NE372
Gate Result Register Bits 30-64(D4) Complemented into Sk Reservation Register Bits 8-43 (RS)
9A
Block Div X1 to MG6 NE354
Terminate the Gating of the Result Register to MG6
10-ns Divide Oscillator Pulse
Gate Select Multiplicand Div 5 NE354
This gate remains active through the end of the following multiply sequence; it gates the contents of the result register (N4) into the MGs as the multiplicand
The last divide oscillator pulse enters the delay train to transfer N4 from the CSA C latches through the PA into the result register
Delay
Gate CSA C NE334
Ingate Product of R3 x D4 Not Used
Delay
Delay
Reset PSD NE360
Tof Cycle Counter 8 Tgr
Ton Cycle Counter 9 Tgr
NE437
Ton Multiply Ring 1A through 2B tgrs NE310
Prepare to Start Mply Iteration 3
6b
Reset Result Register NE360
Delay
9B
Divide Ring NE314, 316
Tof Start Divide Osc Tgr NE304
Stop Divide Oscillator
Delay Train NE334, 360
Gate Result NE360
Gate PA Output (N4) into Result Register
N4
Tof Stop Mply Osc Tgr NE302
All Divide Ring Tgrs Remain On Until Last Cycle Reset
Stop Mply Osc Tgr Off
Start Mply Oscillator NE302
Multiply Oscillator Pulse
Gt Mplr Decoder NE308
Gate Mply Oscillator Pulses to:
1. Step Mply Ring NE310
2. Delay Train NE330
Sel Mul Dec from Sk Mply 3 NE342
Sel Mply Iteration 3 NC751
Gate Bits 24-36 from Sk Res Reg to Mul Dec NC751-778
Ingate Selected Mplr Bits into Mul Dec; Decode Shift Controls to MG1-6 NC780-809
Delay
Gt MG Cds NE330-332
Mply Ring 1A-2B Tgrs Set by 9A Divide
Execute Cycle 9
Sel Mul Dec from Sk Mply 4 NE342
Sel Mply Iteration 4 NC754
Gate Bits 12-24 from Sk Res Reg to Mul Dec NC751-778
Gate Multiplicand (N5) through Multiple Gates to Input of Adders CSA A and B ND055-236
Mply Ring 2B and not 4A
Delay
Gt CSA C NE334
Mply Ring 3B and not 5A
Gate CSA C Carry and Sum Latches and CSA A Sum Delay ND353-501
Sel Mul Dec from Sk Mply 5 NE342
Sel Mply Iteration 5 NC751
Gate Bits 0-12 from Sk Res Reg to Mul Dec NC751-778
Delay
Ton Cycle Counter 10 Tgr
Tof Cycle Counter 9 Tgr
NE437
Mply Ring 4B
Gt CSA E NE340
Gate CSA E Carry and Sum Latches and Sum Delay ND595-757
Mply Ring 5A and 5B
Delay
Gt CSA F NE342
Cycle Counter 10 Tgr On
Multiply Ring NE310-312
End of Mply Iterations:
1. Ton Stop Mply Osc Tgr NE300
2. Hold Final Quotient (N5) in CSA F Until Result Reg Read-In
Gate CSA F Carry and Sum Latches and Spill Adder ND595-757
Ton M/D Unit CDB Req Tgr NE340
W
Sheet 5
X
Sheet 5

DIAGRAM 5-216. DIVIDE EXECUTION (SHEET 4 OF 6)

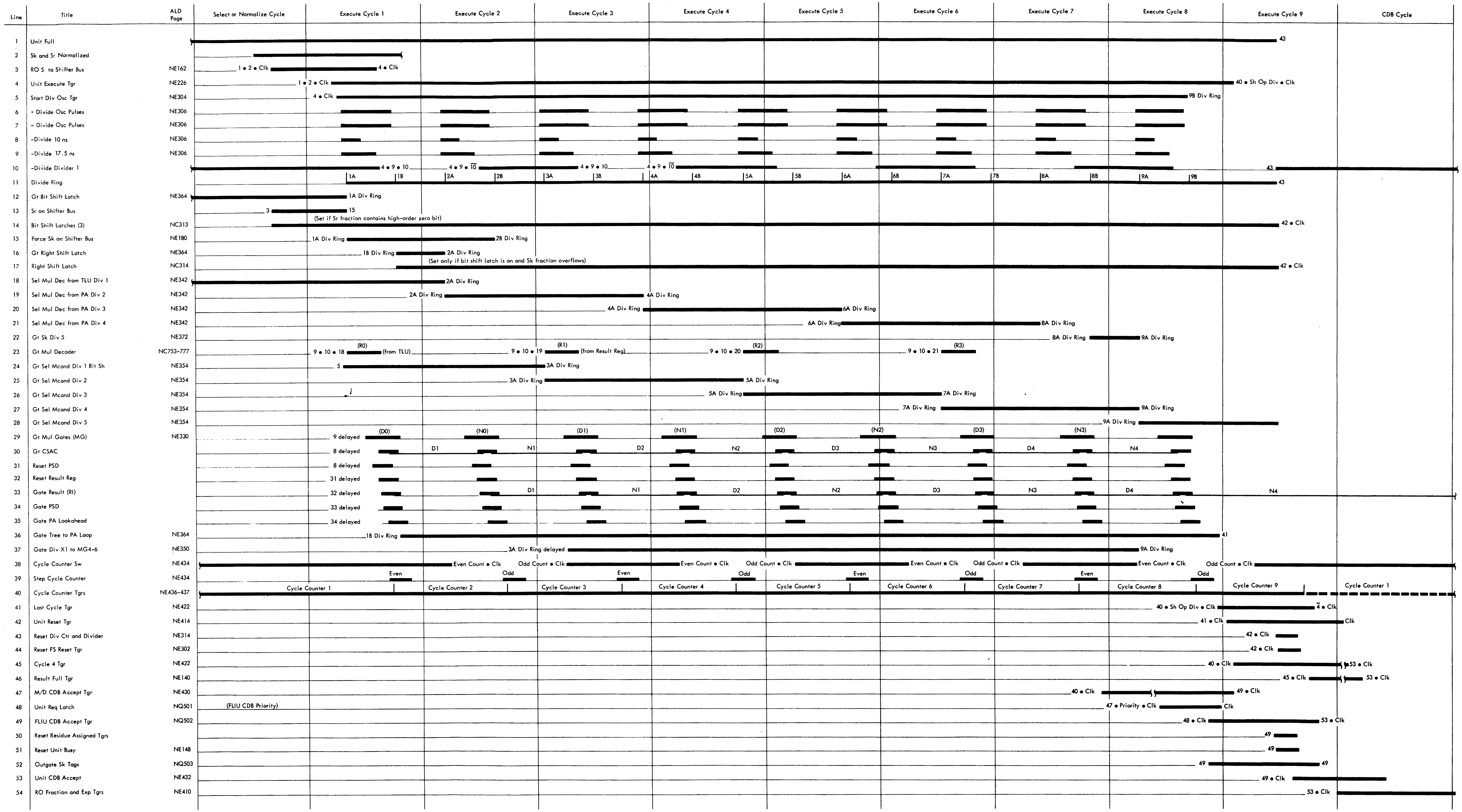
LONG PRECISION



Sheet 4
W
Sheet 4
X
Execute Cycle 10
Unit CDB Request Tgr On
No
FLIU CDB Priority NQ501
Yes
Ton Cycle Counter 11 Tgr Tof Cycle Counter 10 Tgr NE437
Ton M/D Unit Request Latch NQ501
Cycle Counter 11 Tgr On
M/D Unit Request Latch On
Ton FLIU CDB Accept Tgr NQ502
Ton Last Cycle Tgr NE422
FLIU CDB Accept Tgr On
Ton Cycle 4 Tgr NE422
Last Cycle Tgr On
Outgate Tags onto CDB Tag Bus NQ503
Cycle 4 Tgr On
Ton Unit Reset Tgr NE414
Reset Unit Busy NE148
Result Full Tgr
On
Off
Unit Reset Tgr On
Execute Cycle 11
Ton Unit CDB Accept Tgr NE432
Gate RI Result Register NE140
Reset Multiplier Decoder Ingates and Multiple Gates 1-6 NE414
Unit CDB Accept Tgr On
Ton RO Fraction and Exponent Tgrs NE410
Start Output Check NE140
Reset CSA C and E NE414
Delay
Ton Result Full Tgr NE140
Delay
Reset PSD NE360
Delay
Reset Result Register NE360
Reset Unit Execute Tgr NC861
Gate Result Register and Reset CSA F NE360
Gate PA Output into Result Register (N5)
Reset Multiply Ring NE310
Reset Divide Ring NE314
Delay
Gate PSD NE360
Reset False Start Reset NE302
Starts Mply Oscillator
Delay
Unit Execute Tgr Off
Delay Train NE360
Gate PA Lookahead NE360
Sets PSD
Cycle Counter 1 Tgr Is Reset On; All Others Reset Off
Reset Cycle Counter Tgrs NE434
Read Out Result Reg to CDB
See Diagram 5-219 for post-shift decoding and post shift during result read-out
CDB Cycle
Reset Cycle 4 Tgr NE432
Reset Result Full Tgr NE140
End of Operation

Note:
Functions generated through the delay train, shown in execute cycle 9, extend into execute cycle 10.

DIAGRAM 5-216. DIVIDE EXECUTION (SHEET 5 OF 6)



Note: Because of circuit delays, the timing shown is approximate.

DIAGRAM 5-216. DIVIDE EXECUTION -- TIMING CHART (SHEET 6 OF 6)

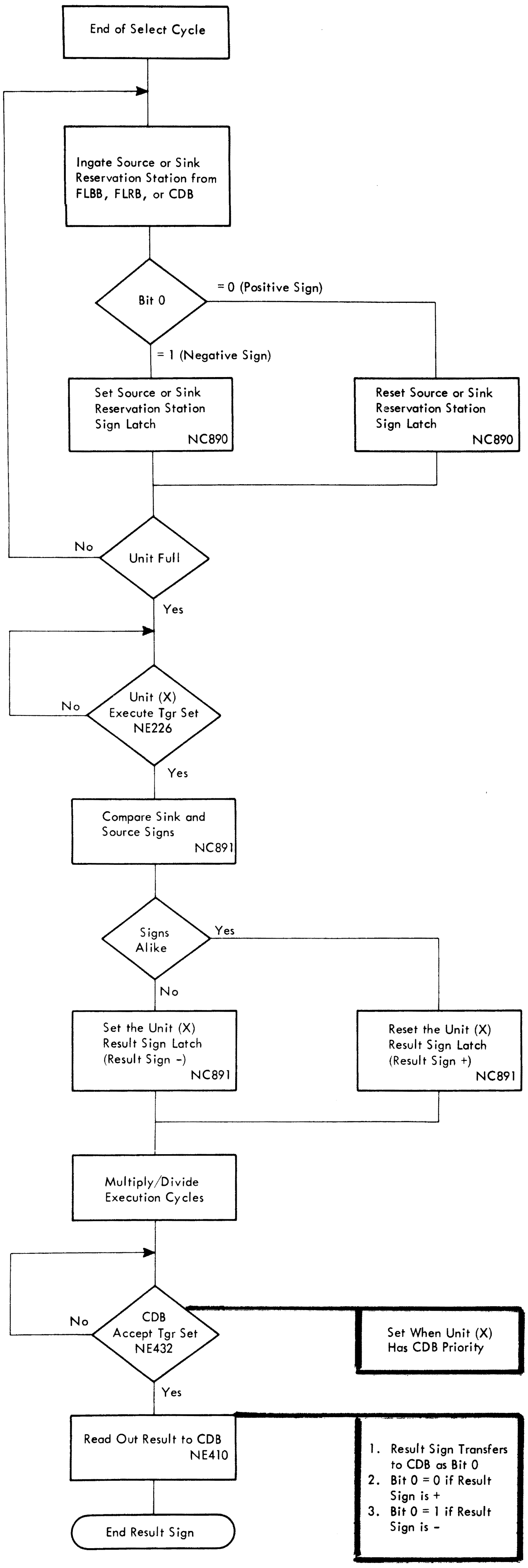
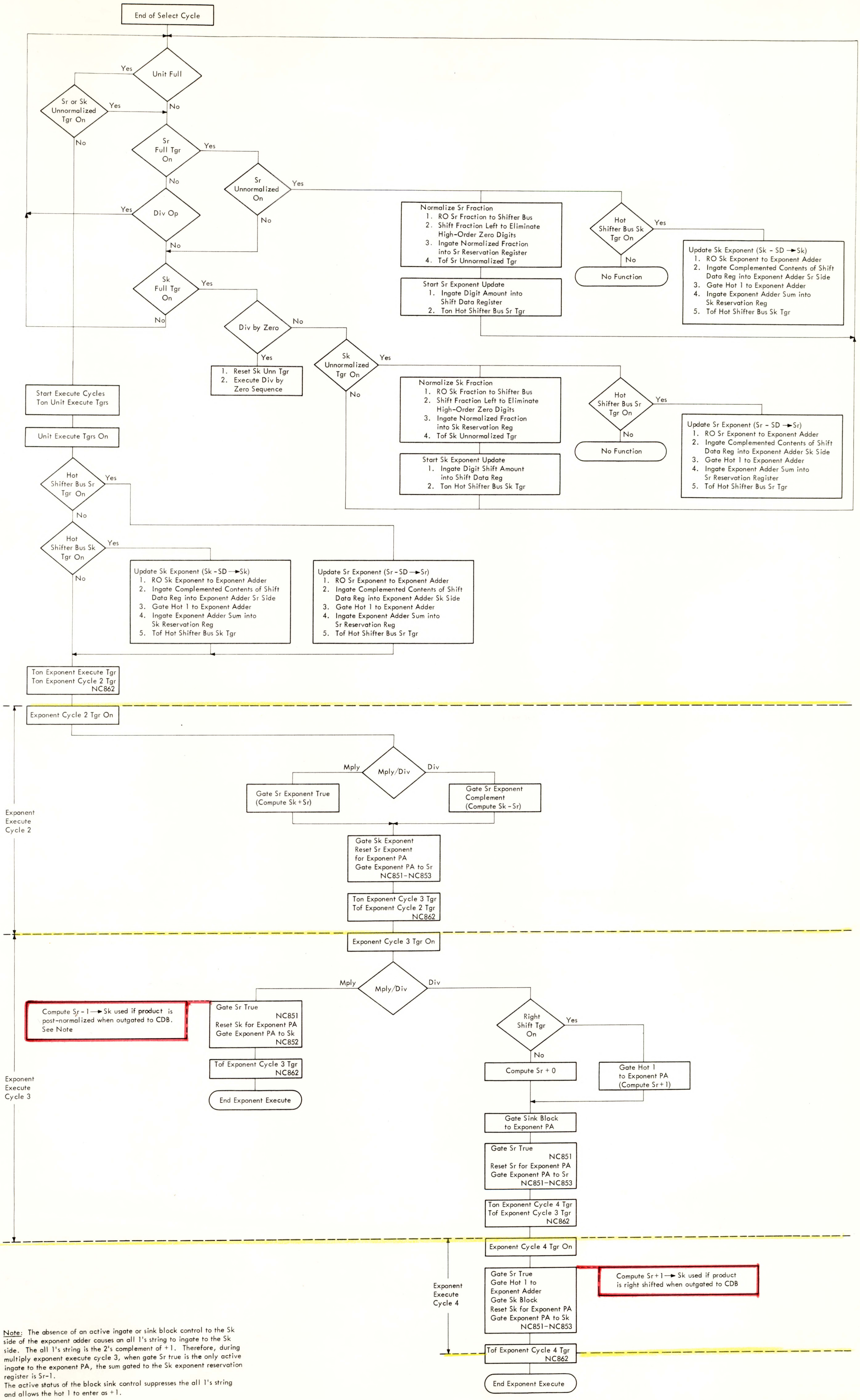
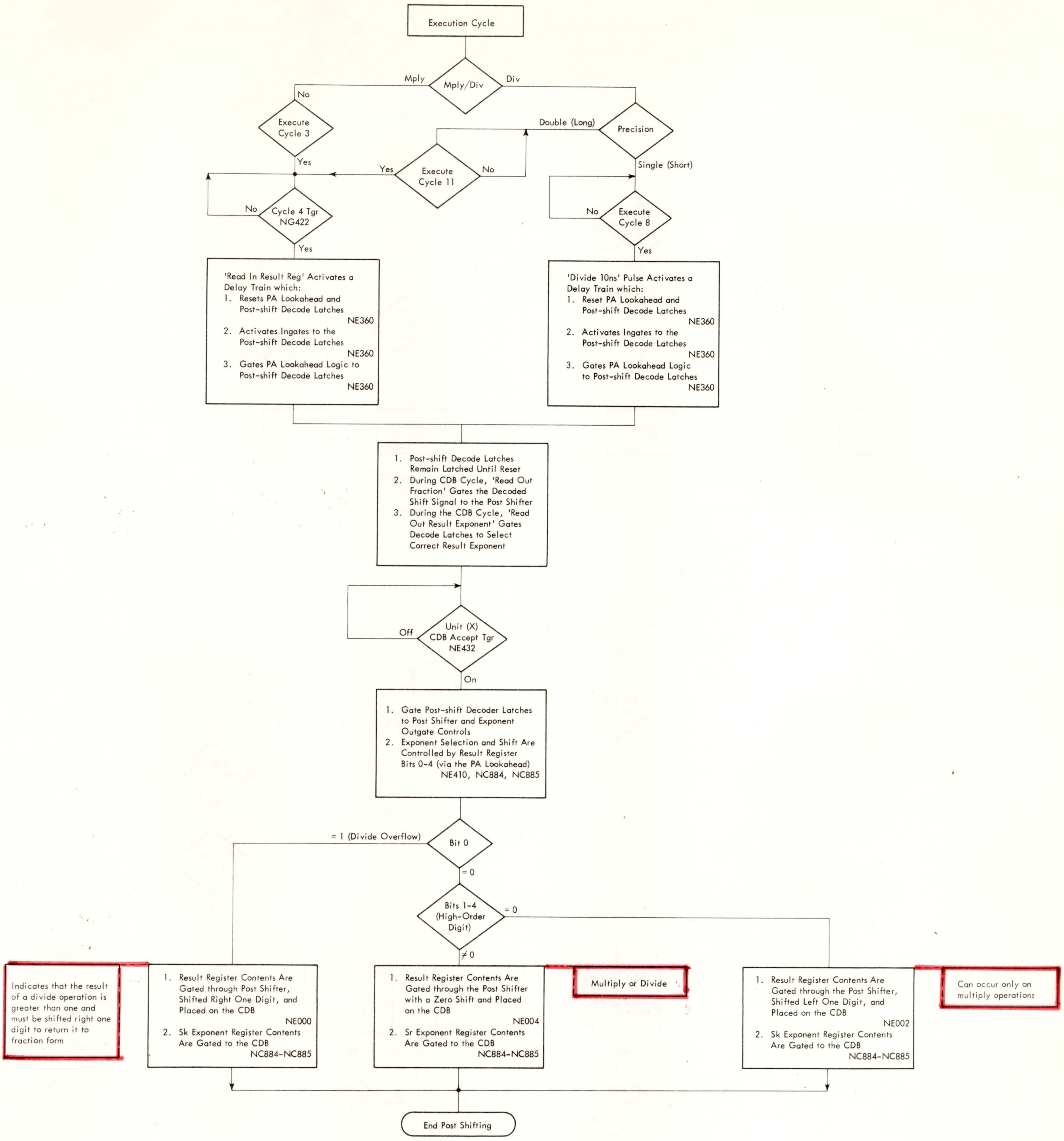

End of Select Cycle
Ingate Source or Sink Reservation Station from FLBB, FLRB, or CDB
Bit 0
= 0 (Positive Sign)
= 1 (Negative Sign)
Set Source or Sink Reservation Station Sign Latch NC890
Reset Source or Sink Reservation Station Sign Latch NC890
No
Unit Full
Yes
No
Unit (X) Execute Tgr Set NE226
Yes
Compare Sink and Source Signs NC891
Signs Alike
Yes
No
Set the Unit (X) Result Sign Latch (Result Sign -) NC891
Reset the Unit (X) Result Sign Latch (Result Sign +) NC891
Multiply/Divide Execution Cycles
No
CDB Accept Tgr Set NE432
Set When Unit (X) Has CDB Priority
Yes
Read Out Result to CDB NE410
1. Result Sign Transfers to CDB as Bit 0
2. Bit 0 = 0 if Result Sign is +
3. Bit 0 = 1 if Result Sign is -
End Result Sign

DIAGRAM 5-217. FMDU SIGN CONTROL


End of Select Cycle
Unit Full
Yes
No
Sr or Sk Unnormalized Tgr On
Yes
No
Sr Full Tgr On
Yes
No
Sr Unnormalized On
Yes
No
Div Op
Yes
No
Sk Full Tgr On
Yes
No
Normalize Sr Fraction
1. RO Sr Fraction to Shifter Bus
2. Shift Fraction Left to Eliminate High-Order Zero Digits
3. Ingate Normalized Fraction into Sr Reservation Register
4. Tof Sr Unnormalized Tgr
Start Sr Exponent Update
1. Ingate Digit Amount into Shift Data Register
2. Ton Hot Shifter Bus Sr Tgr
Hot Shifter Bus Sk Tgr On
Yes
No
No Function
Update Sk Exponent (Sk - SD ⟶ Sk)
1. RO Sk Exponent to Exponent Adder
2. Ingate Complemented Contents of Shift Data Reg into Exponent Adder Sr Side
3. Gate Hot 1 to Exponent Adder
4. Ingate Exponent Adder Sum into Sk Reservation Reg
5. Tof Hot Shifter Bus Sk Tgr
Div by Zero
No
Yes
1. Reset Sk Unn Tgr
2. Execute Div by Zero Sequence
Sk Unnormalized Tgr On
Yes
No
Normalize Sk Fraction
1. RO Sk Fraction to Shifter Bus
2. Shift Fraction Left to Eliminate High-Order Zero Digits
3. Ingate Normalized Fraction into Sk Reservation Reg
4. Tof Sk Unnormalized Tgr
Start Sk Exponent Update
1. Ingate Digit Shift Amount into Shift Data Reg
2. Ton Hot Shifter Bus Sk Tgr
Hot Shifter Bus Sr Tgr On
Yes
No
No Function
Update Sr Exponent (Sr - SD ⟶ Sr)
1. RO Sr Exponent to Exponent Adder
2. Ingate Complemented Contents of Shift Data Reg into Exponent Adder Sk Side
3. Gate Hot 1 to Exponent Adder
4. Ingate Exponent Adder Sum into Sr Reservation Register
5. Tof Hot Shifter Bus Sr Tgr
Start Execute Cycles
Ton Unit Execute Tgrs
Unit Execute Tgrs On
Hot Shifter Bus Sr Tgr On
Yes
No
Hot Shifter Bus Sk Tgr On
Yes
No
Update Sk Exponent (Sk - SD ⟶ Sk)
1. RO Sk Exponent to Exponent Adder
2. Ingate Complemented Contents of Shift Data Reg into Exponent Adder Sr Side
3. Gate Hot 1 to Exponent Adder
4. Ingate Exponent Adder Sum into Sk Reservation Reg
5. Tof Hot Shifter Bus Sk Tgr
Update Sr Exponent (Sr - SD ⟶ Sr)
1. RO Sr Exponent to Exponent Adder
2. Ingate Complemented Contents of Shift Data Reg into Exponent Adder Sk Side
3. Gate Hot 1 to Exponent Adder
4. Ingate Exponent Adder Sum into Sr Reservation Reg
5. Tof Hot Shifter Bus Sr Tgr
Ton Exponent Execute Tgr
Ton Exponent Cycle 2 Tgr
NC862
Exponent Cycle 2 Tgr On
Exponent Execute Cycle 2
Mply
Mply/Div
Div
Gate Sr Exponent True (Compute Sk + Sr)
Gate Sr Exponent Complement (Compute Sk - Sr)
Gate Sk Exponent
Reset Sr Exponent for Exponent PA
Gate Exponent PA to Sr
NC851-NC853
Ton Exponent Cycle 3 Tgr
Tof Exponent Cycle 2 Tgr
NC862
Exponent Cycle 3 Tgr On
Mply
Mply/Div
Div
Compute Sr - 1 ⟶ Sk used if product is post-normalized when outgated to CDB. See Note
Gate Sr True
NC851
Reset Sk for Exponent PA
Gate Exponent PA to Sk
NC852
Tof Exponent Cycle 3 Tgr
NC862
End Exponent Execute
Exponent Execute Cycle 3
Right Shift Tgr On
Yes
No
Compute Sr + 0
Gate Hot 1 to Exponent PA (Compute Sr + 1)
Gate Sink Block to Exponent PA
Gate Sr True
NC851
Reset Sr for Exponent PA
Gate Exponent PA to Sr
NC851-NC853
Ton Exponent Cycle 4 Tgr
Tof Exponent Cycle 3 Tgr
NC862
Exponent Cycle 4 Tgr On
Exponent Execute Cycle 4
Gate Sr True
Gate Hot 1 to Exponent Adder
Gate Sk Block
Reset Sk for Exponent PA
Gate Exponent PA to Sk
NC851-NC853
Compute Sr + 1 ⟶ Sk used if product is right shifted when outgated to CDB
Tof Exponent Cycle 4 Tgr
NC862
End Exponent Execute

Note: The absence of an active ingate or sink block control to the Sk side of the exponent adder causes an all 1's string to ingate to the Sk side. The all 1's string is the 2's complement of + 1. Therefore, during multiply exponent execute cycle 3, when gate Sr true is the only active ingate to the exponent PA, the sum gated to the Sk exponent reservation register is Sr-1.
The active status of the block sink control suppresses the all 1's string and allows the hot 1 to enter as + 1.

DIAGRAM 5-218. FMDU EXPONENT EXECUTION


Execution Cycle
Mply
Mply/Div
Div
No
Execute Cycle 3
Double (Long)
Precision
Yes
Yes
Execute Cycle 11
No
Single (Short)
No
Cycle 4 Tgr NG422
No
Execute Cycle 8
Yes
Yes
'Read In Result Reg' Activates a Delay Train which:
1. Resets PA Lookahead and Post-shift Decode Latches NE360
2. Activates Ingates to the Post-shift Decode Latches NE360
3. Gates PA Lookahead Logic to Post-shift Decode Latches NE360
'Divide 10ns' Pulse Activates a Delay Train which:
1. Reset PA Lookahead and Post-shift Decode Latches NE360
2. Activates Ingates to the Post-shift Decode Latches NE360
3. Gates PA Lookahead Logic to Post-shift Decode Latches NE360
1. Post-shift Decode Latches Remain Latched Until Reset
2. During CDB Cycle, 'Read Out Fraction' Gates the Decoded Shift Signal to the Post Shifter
3. During the CDB Cycle, 'Read Out Result Exponent' Gates Decode Latches to Select Correct Result Exponent
Off
Unit (X) CDB Accept Tgr NE432
On
1. Gate Post-shift Decoder Latches to Post Shifter and Exponent Outgate Controls
2. Exponent Selection and Shift Are Controlled by Result Register Bits 0-4 (via the PA Lookahead) NE410, NC884, NC885
= 1 (Divide Overflow)
Bit 0
= 0
Bits 1-4 (High-Order Digit)
= 0
≠ 0
1. Result Register Contents Are Gated through Post Shifter, Shifted Right One Digit, and Placed on the CDB NE000
2. Sk Exponent Register Contents Are Gated to the CDB NC884-NC885
1. Result Register Contents Are Gated through the Post Shifter with a Zero Shift and Placed on the CDB NE004
2. Sr Exponent Register Contents Are Gated to the CDB NC884-NC885
1. Result Register Contents Are Gated through the Post Shifter, Shifted Left One Digit, and Placed on the CDB NE002
2. Sk Exponent Register Contents Are Gated to the CDB NC884-NC885
End Post Shifting

Indicates that the result of a divide operation is greater than one and must be shifted right one digit to return it to fraction form

Multiply or Divide

Can occur only on multiply operations

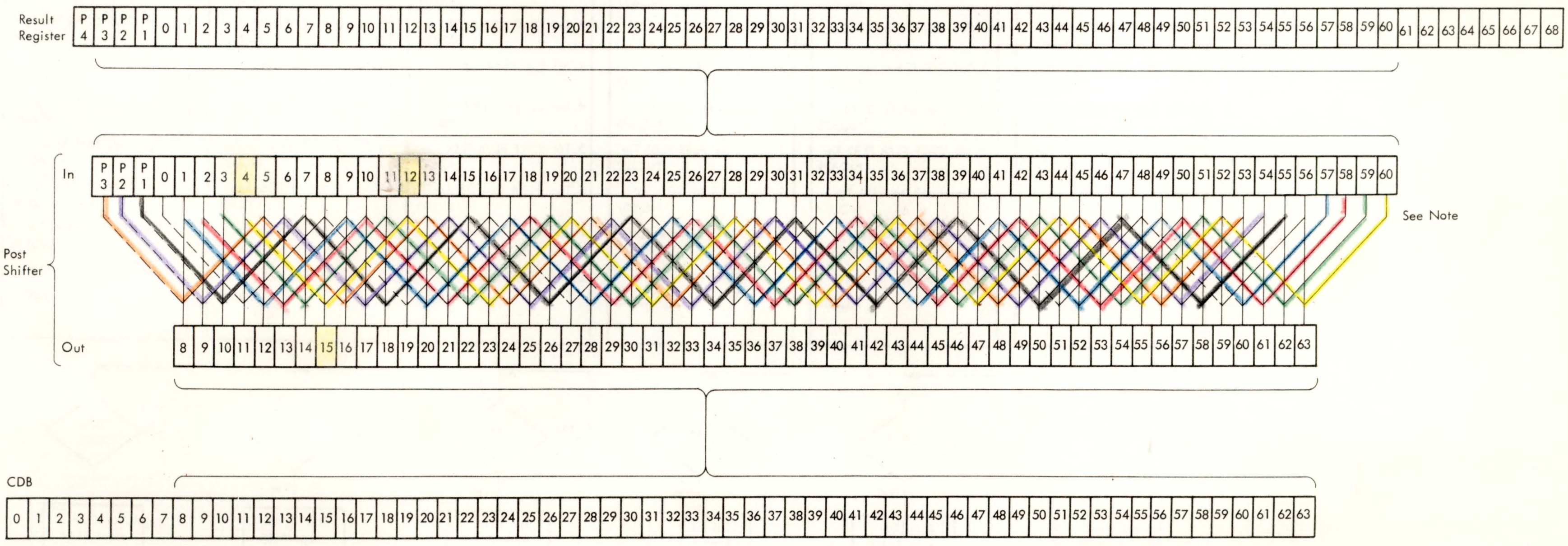

Result Register
P4 P3 P2 P1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68
Post Shifter
In
P3 P2 P1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
See Note
Out
8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63
CDB
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63

DIAGRAM 5-219. FMDU POST-SHIFT DECODING AND POST SHIFTING

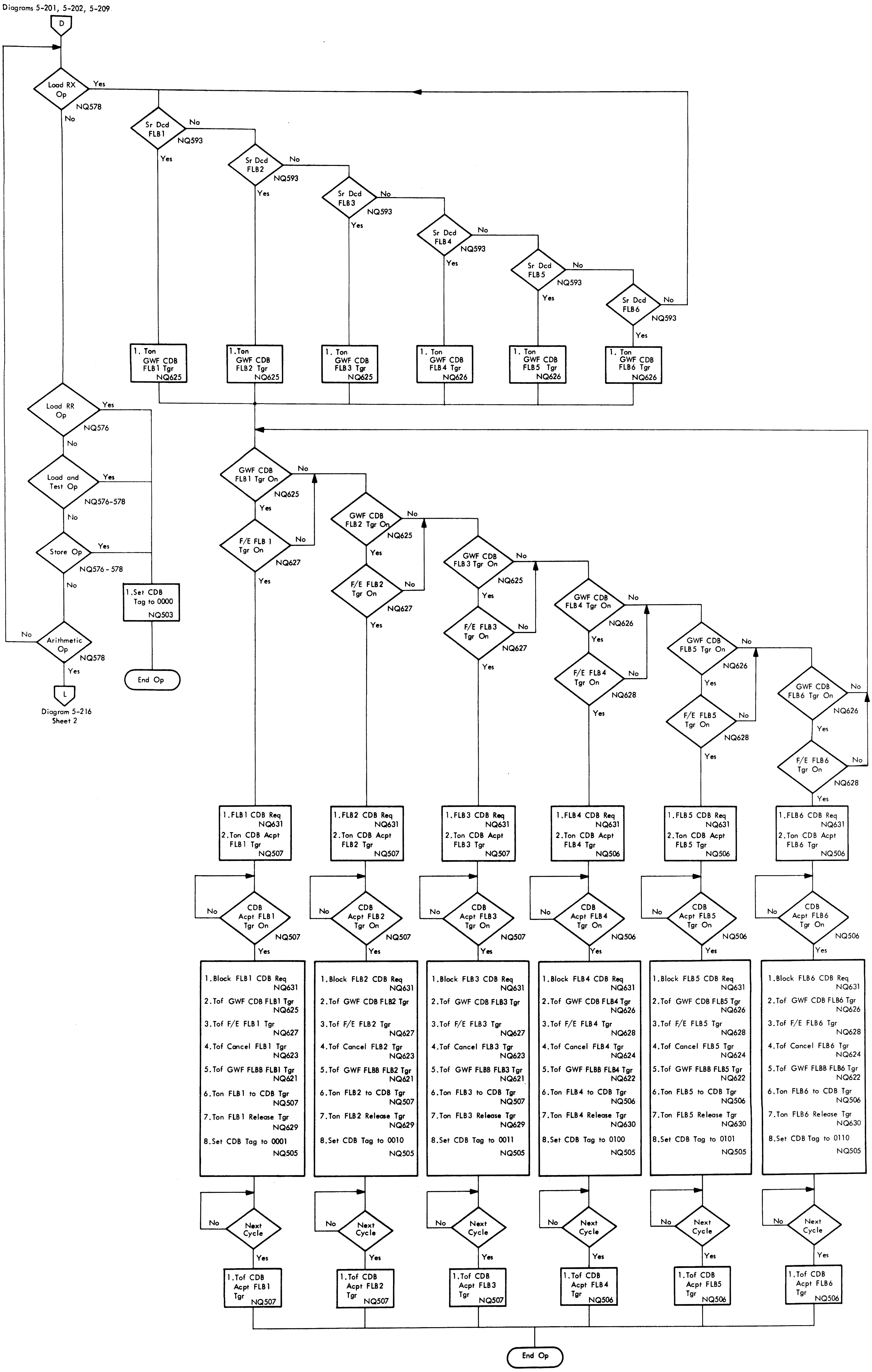

Diagrams 5-201, 5-202, 5-209
D
Load RX Op
NQ578
Yes
No
Sr Dcd FLB1
NQ593
Sr Dcd FLB2
NQ593
Sr Dcd FLB3
NQ593
Sr Dcd FLB4
NQ593
Sr Dcd FLB5
NQ593
Sr Dcd FLB6
NQ593
1. Ton GWF CDB FLB1 Tgr NQ625
1. Ton GWF CDB FLB2 Tgr NQ625
1. Ton GWF CDB FLB3 Tgr NQ625
1. Ton GWF CDB FLB4 Tgr NQ626
1. Ton GWF CDB FLB5 Tgr NQ626
1. Ton GWF CDB FLB6 Tgr NQ626
Load RR Op
NQ576
Load and Test Op
NQ576-578
Store Op
NQ576 - 578
1. Set CDB Tag to 0000 NQ503
End Op
Arithmetic Op
NQ578
L
Diagram 5-216 Sheet 2
GWF CDB FLB1 Tgr On
NQ625
F/E FLB 1 Tgr On
NQ627
GWF CDB FLB2 Tgr On
NQ625
F/E FLB2 Tgr On
NQ627
GWF CDB FLB3 Tgr On
NQ625
F/E FLB3 Tgr On
NQ627
GWF CDB FLB4 Tgr On
NQ626
F/E FLB4 Tgr On
NQ628
GWF CDB FLB5 Tgr On
NQ626
F/E FLB5 Tgr On
NQ628
GWF CDB FLB6 Tgr On
NQ626
F/E FLB6 Tgr On
NQ628
1. FLB1 CDB Req NQ631
2. Ton CDB Acpt FLB1 Tgr NQ507
1. FLB2 CDB Req NQ631
2. Ton CDB Acpt FLB2 Tgr NQ507
1. FLB3 CDB Req NQ631
2. Ton CDB Acpt FLB3 Tgr NQ507
1. FLB4 CDB Req NQ631
2. Ton CDB Acpt FLB4 Tgr NQ506
1. FLB5 CDB Req NQ631
2. Ton CDB Acpt FLB5 Tgr NQ506
1. FLB6 CDB Req NQ631
2. Ton CDB Acpt FLB6 Tgr NQ506
CDB Acpt FLB1 Tgr On
NQ507
CDB Acpt FLB2 Tgr On
NQ507
CDB Acpt FLB3 Tgr On
NQ507
CDB Acpt FLB4 Tgr On
NQ506
CDB Acpt FLB5 Tgr On
NQ506
CDB Acpt FLB6 Tgr On
NQ506
1. Block FLB1 CDB Req NQ631
2. Tof GWF CDB FLB1 Tgr NQ625
3. Tof F/E FLB1 Tgr NQ627
4. Tof Cancel FLB1 Tgr NQ623
5. Tof GWF FLBB FLB1 Tgr NQ621
6. Ton FLB1 to CDB Tgr NQ507
7. Ton FLB1 Release Tgr NQ629
8. Set CDB Tag to 0001 NQ505
1. Block FLB2 CDB Req NQ631
2. Tof GWF CDB FLB2 Tgr
3. Tof F/E FLB2 Tgr NQ627
4. Tof Cancel FLB2 Tgr NQ623
5. Tof GWF FLBB FLB2 Tgr NQ621
6. Ton FLB2 to CDB Tgr NQ507
7. Ton FLB2 Release Tgr NQ629
8. Set CDB Tag to 0010 NQ505
1. Block FLB3 CDB Req NQ631
2. Tof GWF CDB FLB3 Tgr
3. Tof F/E FLB3 Tgr NQ627
4. Tof Cancel FLB3 Tgr NQ623
5. Tof GWF FLBB FLB3 Tgr NQ621
6. Ton FLB3 to CDB Tgr NQ507
7. Ton FLB3 Release Tgr NQ629
8. Set CDB Tag to 0011 NQ505
1. Block FLB4 CDB Req NQ631
2. Tof GWF CDB FLB4 Tgr NQ626
3. Tof F/E FLB4 Tgr NQ628
4. Tof Cancel FLB4 Tgr NQ624
5. Tof GWF FLBB FLB4 Tgr NQ622
6. Ton FLB4 to CDB Tgr NQ506
7. Ton FLB4 Release Tgr NQ630
8. Set CDB Tag to 0100 NQ505
1. Block FLB5 CDB Req NQ631
2. Tof GWF CDB FLB5 Tgr NQ626
3. Tof F/E FLB5 Tgr NQ628
4. Tof Cancel FLB5 Tgr NQ624
5. Tof GWF FLBB FLB5 Tgr NQ622
6. Ton FLB5 to CDB Tgr NQ506
7. Ton FLB5 Release Tgr NQ630
8. Set CDB Tag to 0101 NQ505
1. Block FLB6 CDB Req NQ631
2. Tof GWF CDB FLB6 Tgr NQ626
3. Tof F/E FLB6 Tgr NQ628
4. Tof Cancel FLB6 Tgr NQ624
5. Tof GWF FLBB FLB6 Tgr NQ622
6. Ton FLB6 to CDB Tgr NQ506
7. Ton FLB6 Release Tgr NQ630
8. Set CDB Tag to 0110 NQ505
Next Cycle
1. Tof CDB Acpt FLB1 Tgr NQ507
1. Tof CDB Acpt FLB2 Tgr NQ507
1. Tof CDB Acpt FLB3 Tgr NQ507
1. Tof CDB Acpt FLB4 Tgr NQ506
1. Tof CDB Acpt FLB5 Tgr NQ506
1. Tof CDB Acpt FLB6 Tgr NQ506
End Op

DIAGRAM 5 - 220. OUTGATES TO CDB (SHEET 1 OF 2)

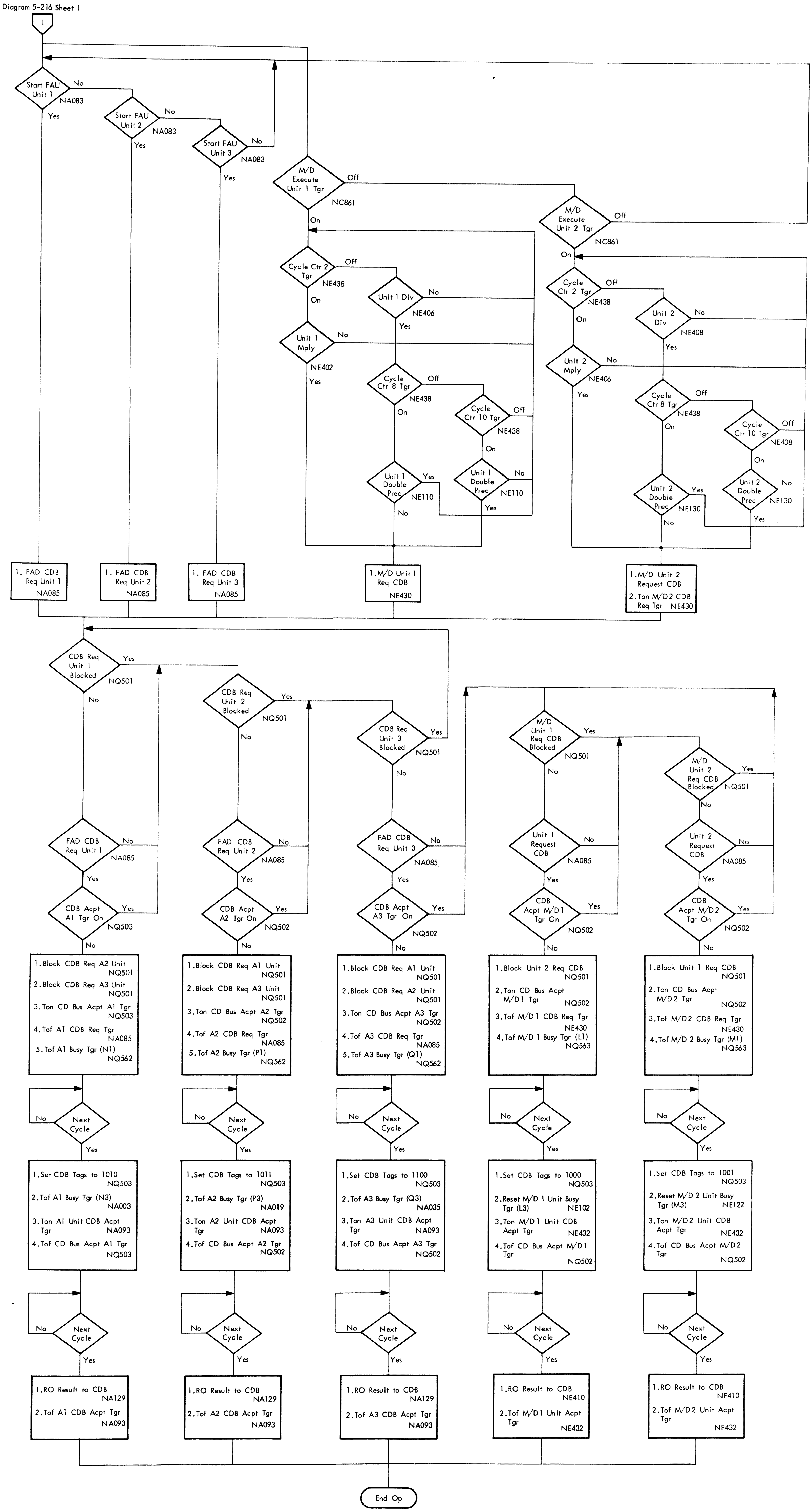
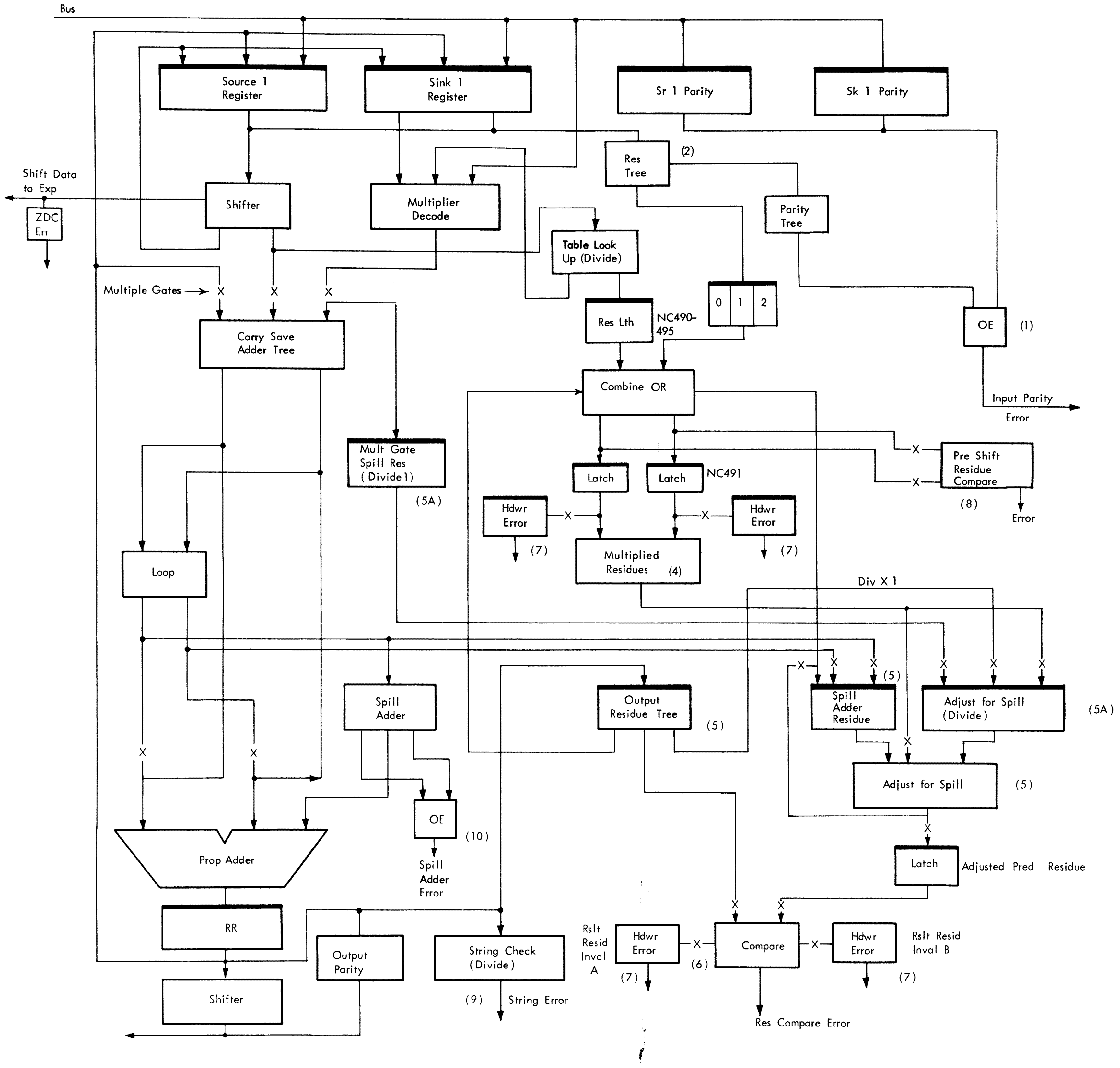


Diagram 5-216 Sheet 1
L
Start FAU Unit 1
NA083
No
Yes
Start FAU Unit 2
NA083
No
Yes
Start FAU Unit 3
NA083
No
Yes
M/D Execute Unit 1 Tgr
NC861
Off
On
M/D Execute Unit 2 Tgr
NC861
Off
On
Cycle Ctr 2 Tgr
NE438
Off
On
Unit 1 Div
NE406
No
Yes
Unit 1 Mply
NE402
No
Yes
Cycle Ctr 8 Tgr
NE438
Off
On
Cycle Ctr 10 Tgr
NE438
Off
On
Unit 1 Double Prec
NE110
Yes
No
Unit 1 Double Prec
NE110
No
Yes
Cycle Ctr 2 Tgr
NE438
Off
On
Unit 2 Div
NE408
No
Yes
Unit 2 Mply
NE406
No
Yes
Cycle Ctr 8 Tgr
NE438
Off
On
Cycle Ctr 10 Tgr
NE438
Off
On
Unit 2 Double Prec
NE130
Yes
No
Unit 2 Double Prec
NE130
No
Yes
1. FAD CDB Req Unit 1 NA085
1. FAD CDB Req Unit 2 NA085
1. FAD CDB Req Unit 3 NA085
1.M/D Unit 1 Req CDB NE430
1.M/D Unit 2 Request CDB
2.Ton M/D2 CDB Req Tgr NE430
CDB Req Unit 1 Blocked
NQ501
Yes
No
CDB Req Unit 2 Blocked
NQ501
Yes
No
CDB Req Unit 3 Blocked
NQ501
Yes
No
M/D Unit 1 Req CDB Blocked
NQ501
Yes
No
M/D Unit 2 Req CDB Blocked
NQ501
Yes
No
FAD CDB Req Unit 1
NA085
No
Yes
FAD CDB Req Unit 2
NA085
No
Yes
FAD CDB Req Unit 3
NA085
No
Yes
Unit 1 Request CDB
NA085
No
Yes
Unit 2 Request CDB
NA085
No
Yes
CDB Acpt A1 Tgr On
NQ503
Yes
No
CDB Acpt A2 Tgr On
NQ502
Yes
No
CDB Acpt A3 Tgr On
NQ502
Yes
No
CDB Acpt M/D 1 Tgr On
NQ502
Yes
No
CDB Acpt M/D 2 Tgr On
NQ502
Yes
No
1.Block CDB Req A2 Unit NQ501
2.Block CDB Req A3 Unit NQ501
3.Ton CD Bus Acpt A1 Tgr NQ503
4.Tof A1 CDB Req Tgr NA085
5.Tof A1 Busy Tgr (N1) NQ562
1.Block CDB Req A1 Unit NQ501
2.Block CDB Req A3 Unit NQ501
3.Ton CD Bus Acpt A2 Tgr NQ502
4.Tof A2 CDB Req Tgr NA085
5.Tof A2 Busy Tgr (P1) NQ562
1.Block CDB Req A1 Unit NQ501
2.Block CDB Req A2 Unit NQ501
3.Ton CD Bus Acpt A3 Tgr NQ502
4.Tof A3 CDB Req Tgr NA085
5.Tof A3 Busy Tgr (Q1) NQ562
1.Block Unit 2 Req CDB NQ501
2.Ton CD Bus Acpt M/D 1 Tgr NQ502
3.Tof M/D 1 CDB Req Tgr NE430
4.Tof M/D 1 Busy Tgr (L1) NQ563
1.Block Unit 1 Req CDB NQ501
2.Ton CD Bus Acpt M/D 2 Tgr NQ502
3.Tof M/D2 CDB Req Tgr NE430
4.Tof M/D 2 Busy Tgr (M1) NQ563
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
1.Set CDB Tags to 1010 NQ503
2.Tof A1 Busy Tgr (N3) NA003
3.Ton A1 Unit CDB Acpt Tgr NA093
4.Tof CD Bus Acpt A1 Tgr NQ503
1.Set CDB Tags to 1011 NQ503
2.Tof A2 Busy Tgr (P3) NA019
3.Ton A2 Unit CDB Acpt Tgr NA093
4.Tof CD Bus Acpt A2 Tgr NQ502
1.Set CDB Tags to 1100 NQ503
2.Tof A3 Busy Tgr (Q3) NA035
3.Ton A3 Unit CDB Acpt Tgr NA093
4.Tof CD Bus Acpt A3 Tgr NQ502
1.Set CDB Tags to 1000 NQ503
2.Reset M/D 1 Unit Busy Tgr (L3) NE102
3.Ton M/D1 Unit CDB Acpt Tgr NE432
4.Tof CD Bus Acpt M/D1 Tgr NQ502
1.Set CDB Tags to 1001 NQ503
2.Reset M/D 2 Unit Busy Tgr (M3) NE122
3.Ton M/D2 Unit CDB Acpt Tgr NE432
4.Tof CD Bus Acpt M/D2 Tgr NQ502
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
Next Cycle
No
Yes
1.RO Result to CDB NA129
2.Tof A1 CDB Acpt Tgr NA093
1.RO Result to CDB NA129
2.Tof A2 CDB Acpt Tgr NA093
1.RO Result to CDB NA129
2.Tof A3 CDB Acpt Tgr NA093
1.RO Result to CDB NE410
2.Tof M/D1 Unit Acpt Tgr NE432
1.RO Result to CDB NE410
2.Tof M/D2 Unit Acpt Tgr NE432
End Op

DIAGRAM 5-220. OUTGATES TO CDB (SHEET 2 OF 2)

Fraction Residue and Parity



Exponent Error Checks

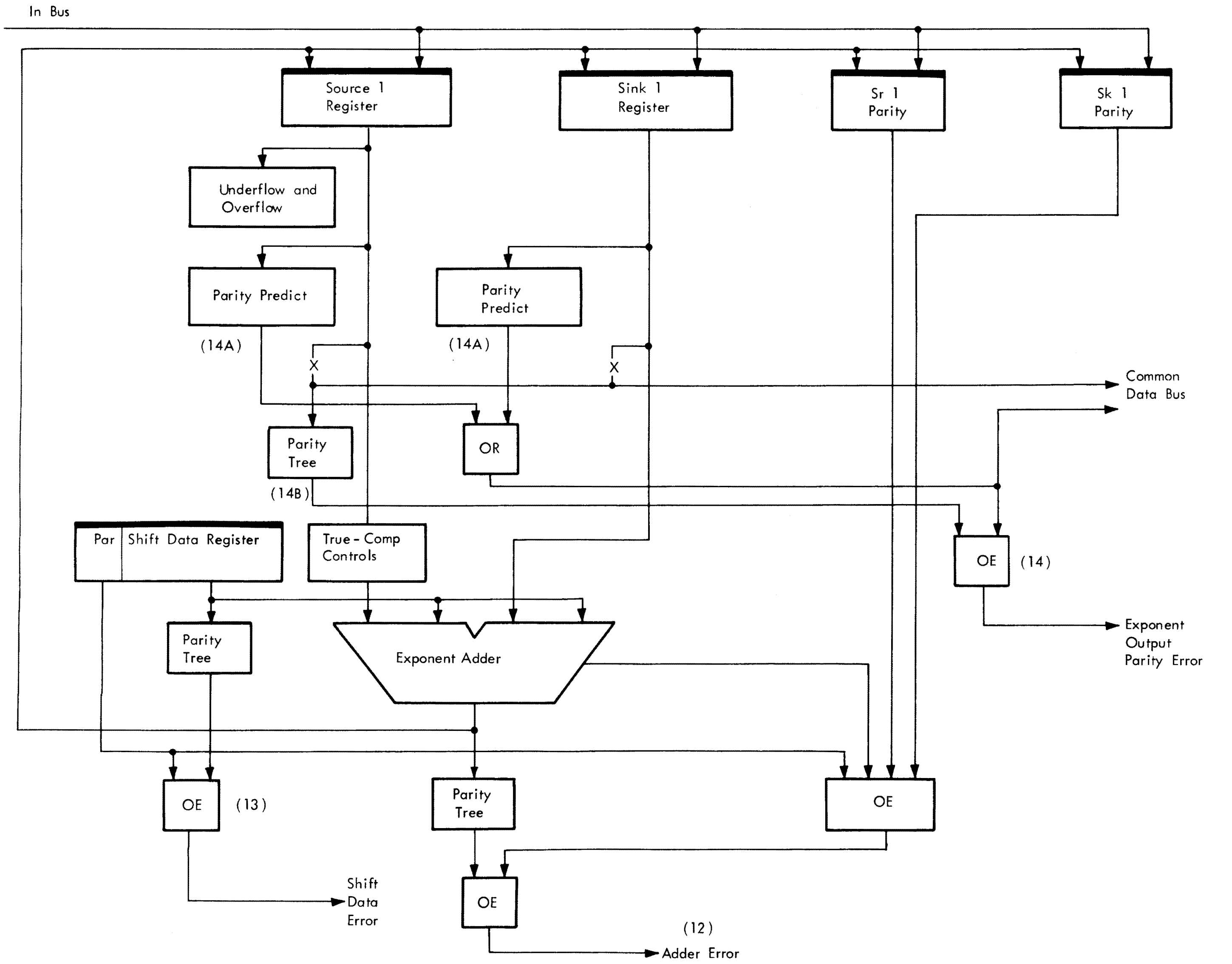
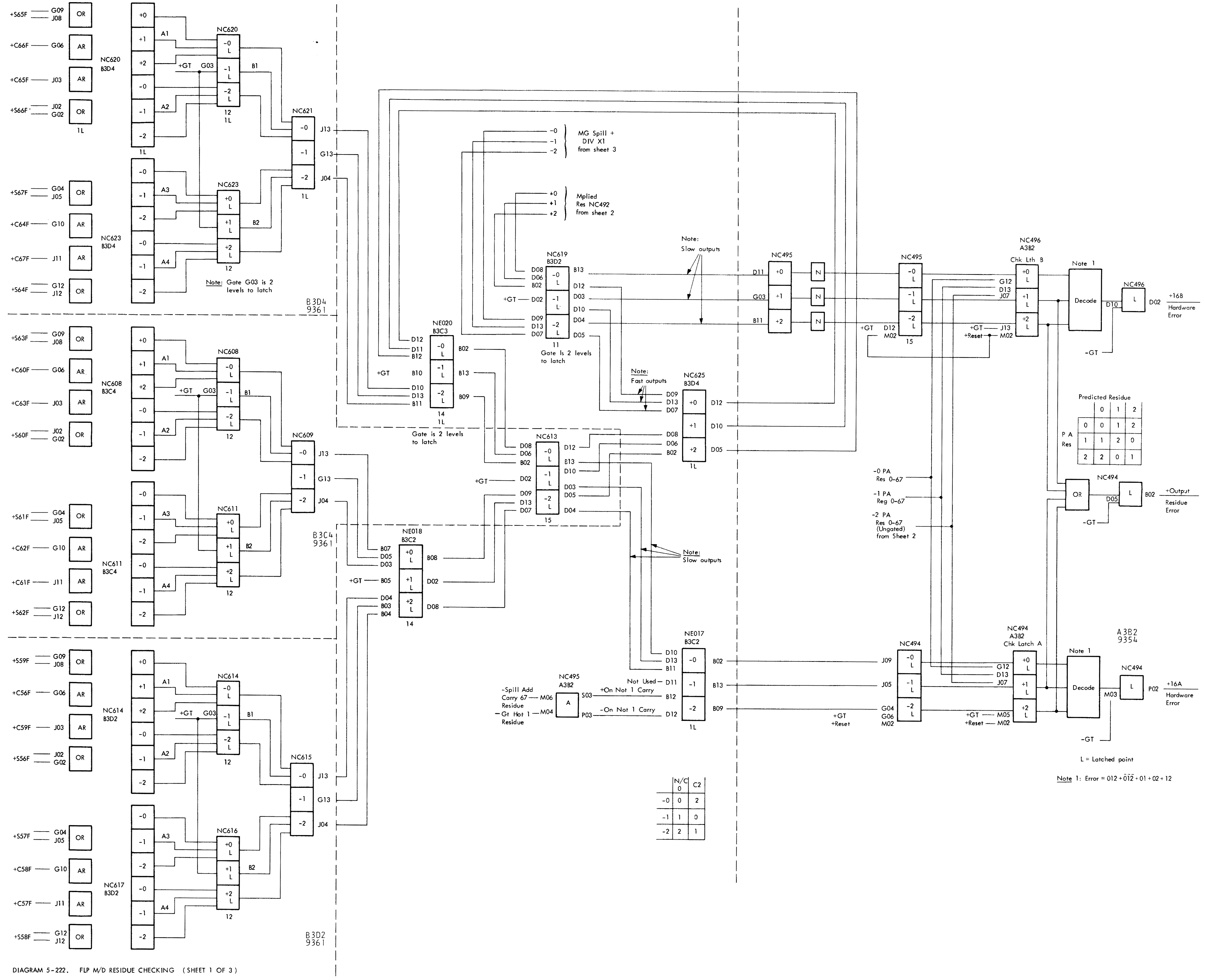


DIAGRAM 5-221. MULTIPLY/DIVIDE ERROR CHECKING



Predicted Residue

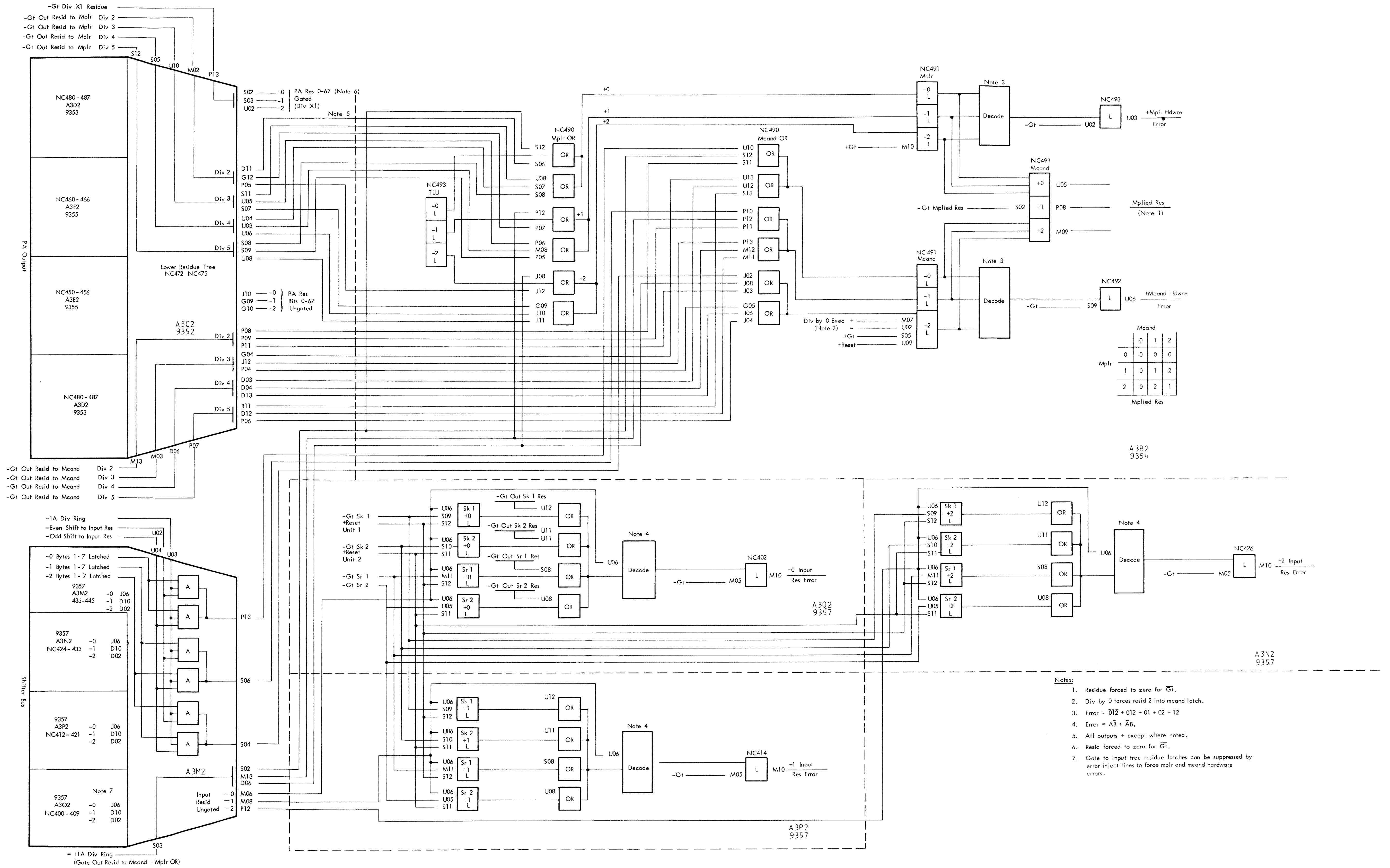
P A Res	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

	N/C 0	C2
-0	0	2
-1	1	0
-2	2	1

L = Latched point

Note 1: Error = $012 + \overline{0}\overline{1}\overline{2} + 01 + 02 + 12$

DIAGRAM 5-222. FLP M/D RESIDUE CHECKING (SHEET 1 OF 3)

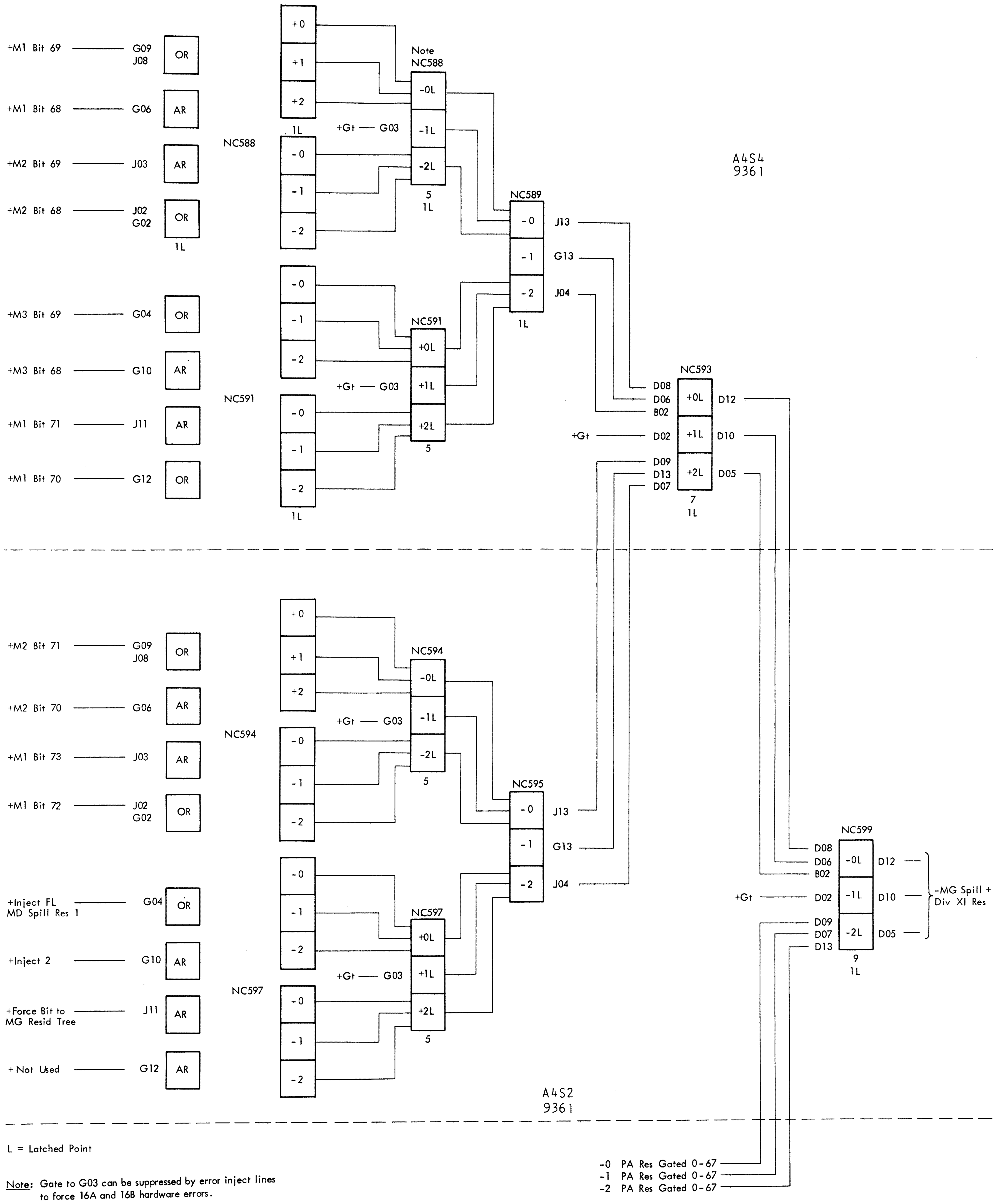


		Mcand		
		0	1	2
Mplr	0	0	0	0
	1	0	1	2
	2	0	2	1
		Mplied Res		

Notes:

1. Residue forced to zero for $\overline{\text{Gt}}$.
2. Div by 0 torces resid 2 into mcand latch.
3. Error = $\bar{0}\bar{1}\bar{2}$ + 012 + 01 + 02 + 12
4. Error = $A\bar{B} + \bar{A}B$.
5. All outputs + except where noted.
6. Resid forced to zero for $\overline{\text{Gt}}$.
7. Gate to input tree residue latches can be suppressed by error inject lines to force mplr and mcand hardware errors.

DIAGRAM 5-222. FLP M/D RESIDUE CHECKING (SHEET 2 OF 3)



L = Latched Point

Note: Gate to G03 can be suppressed by error inject lines to force 16A and 16B hardware errors.

DIAGRAM 5-222. FLP M/D RESIDUE CHECKING (SHEET 3 OF 3)

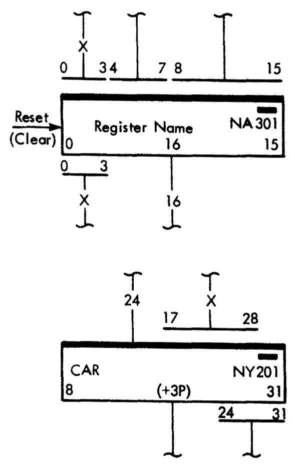
Gen	General; Generate
GP Acpt	General Purpose Register Accept
GPR	General Purpose Register
Gr	Group
Gt	Gate
Gtd	Gated
GWFCDB	Go When Full Common Data Bus
GWFFLBB	Go When Full Floating Buffer Bus
HIO	(Mnemonic) Halt I/O (SI)
HPMS	High Performance Main Storage
HOD	High Order Digit
HS	Half Sum
HSB	High Speed Bus
HW	Halfword
I	Instruction
I-Box	Instruction Processor
IC	(Mnemonic)Insert Character (RX); Instruction Count
IDR	Immediate Data Register
IF	Instruction Fetch
IFT	Instruction Fetch Trigger
Ig	Ingate
ILC	Instruction Length Code
IMRT	Instruction from Memory Request Trigger
Incr	Increment
Ind	Indication; Indicator
Inh	Inhibit
Init	Initialize
In Pri	Input Priority
Insn	Instruction
Int	Internal
Intr	Interrupt
Inv	Invalid
I/O	Input/Output
IOC	I/O Channel
IPL	Initial Program Load
IR	Instruction Register
IRCTR	Instruction Register Counter
ISK	(Mnemonic) Insert Storage Key (RR)
ISR	Instruction Sink Register
IWC	Indicator Word Counter
K	Thousand
L	Operand Length
LA	(Mnemonic) Load Address (RX)
Last	Last Trigger
LB	Lower Bound; Loop Block
LBCTR	Lower Bound Counter
L Cnt	Length Count
LCS	Large Capacity Storage (Same as EMS)
Ld	Load
LM	(Mnemonic) Load Multiple (RS)
LO	Low Order
LOD	Low Order Digit
LPSW	(Mnemonic) Load PSW (SI)
LSN	Load Multiple (LM), Store Multiple (STM), and NOXCM Instructions
Lth	Latch
MA	Multi-Access
MAC	Multi-Access Code
MALS	Multi-Access Link Suppressed
Man	Manual
MAR	Memory Address Register (Same as SAR)
MAT	Multi-Access Trigger
MC	Maintenance Console; Megacycle; Marginal Checking
Mcand	Multiplicand
MCW	Maintenance Control Word
M/D	Multiply/Divide
MDR	Memory Data Register (Same as SDR)
ME	Multiply (Short)
Mem	Memory
MG	Motor Generator; Multiple Gate
MOP	Multiple Operation
Mplr	Multiplier
Mple	Multiple
Mod	Modifier
Mply	Muliply
MS	Main Storage (Same as MWS)
MSB	Medium Speed Bus
MSC	Monolithic Storage Cell
MSCE	Main Storage Control Element
MSM	Main Storage Module
MTBF	Mean Time Between Failures
Mul Dec	Multiplier Decoder
MVC	(Mnemonic) Move (SS)
MVN	(Mnemonic) Move Numerics (SS)
MVO	(Mnemonic) Move with Offset (SS)
MVZ	(Mnemonic) Move Zones (SS)
MWS	Main Working Storage (Same as MS)

N	Inverter
NC	Mnemonic AND (SS)
Neg	Negative
NIAT	New Instruction Address Trigger
No.	Number
N Op	No Operation
Norm	Normalize
NOXCM	(Mnemonic) NC (AND) OC (OR) XC (Exclusive OR) CLC (Compare Logical) MVC (Move) MVZ (Move Zone) MVN (Move Numeric)
Ns	Nanosecond
NSI	Next Sequential Instruction
NUBAT	New Upper Bound Address Trigger
OC	(Mnemonic) OR (SS)
Oflo	Overflow
Og	Outgate
Olap	Overlap
Op	Operation
Opnd	Operand
OR	Outring (Line Name Only)
Out Pri	Output Priority
Ord	Order
Osc	Oscillator
Ovrd	Override
Ovrlp	Overlap
P	Parity; Position; Priority
PA	Propagate Adder
PACK	(Mnemonic) Pack (SS)
Par	Parity
PAR	Position Address Register
PAW	Position Address Word
PC	Parity Check
PDU	Power Distribution Unit
PG	Parity Generate
PH	Polarity Hold
PK	Protection Key
PM	Program Mask; Protect Memory (Same as PS)
Pos	Position; Positive
PPE	Peripheral Processor Element
PPln	Pipeline
Prec	Precision
Pred	Predict
Pri	Primary; Priority
Prob	Problem
Prog	Program
Prop	Propagate
Prot	Protect; Protection
PS	Protect Storage (Same as PM); Power Supply
PSCE	Peripheral Storage Control Element
PSW	Program Status Word
Ptrn	Pattern
Pty	Parity
PUMO	(Mnemonic) PACK (Pack) UNPK (Unpack) MVO (Move with Offset)
PV	Protection Violation
Pwd	Powered
Q	Queue
Qx	Queue (any number)
R	Ready
Rd	Read
RDD	(Mnemonic) Read Direct (SI)
Rdy	Ready
Rec	Record
Reg	Register
Rel	Release
Req	Request
Res	Reset; Residue
Resd	Reserved
Resp	Response
Rgen	Regenerate; Regeneration
RI	Read In
R/L	Remote/Local
RO	Read Out
RR	(Instruction Format) Both Operands from GPR's
RS	Request Stack; (Instruction Format) One Operand from a GPR, the Other from Storage
Rslt	Result
Rsrvtn	Reservation
Rt	Right
Rtn	Return
RUA	Register Unavailable for Address Generation
RUM	Register Unavailable for Modification

RX	(Instruction Format) One Operand from a GPR, the Other from an Indexed Storage Location
S0	State Zero
SA	Sink Address
SAA	Storage Address Alteration
SAB	Storage Address Bus
SAP	Storage Address Protection
SAR	Storage Address Register (Same as MAR); Store Address Register
SB	Sink Address Bus
SBI	Storage Bus In
SBO	Storage Bus Out
SC	Single-Cycle; Storage Channel; Sequence Complete
Sc	Source
SDB	Storage Data Buffer
SDE	Storage Distribution Element
SDR	Storage Data Register (Same as MAR)
Sel	Select
SERR	CPE Status Recording Program
SEVA	Systems Evaluation Program
S/F	Store/Fetch
Sh	Shift
Shftr	Shifter
SI	(Instruction Format) One Operand from Storage, the Other Is Immediate
SIAT	Store Into Array Trigger
SIIS	Store into Instruction Stream
SIO	(Mnemonic) Start I/O (SI)
SIT	Store Interlock Trigger
SK	Storage Key
Sk	Sink
S/L	Short/Long Precision
SLA	(Mnemonic) Shift Left Single (RS)
SLCB	Save Loop Close B Register
SLC	Save Loop Close
SLCIR	Save Loop Close Instruction Register
SLDA	(Mnemonic) Shift Left Double (RS)
SLCX	Save Loop Close - X Register
SLI	Suppress-Length-Indication
SLT	Save Loop Target; Solid Logic Technology
SM	Storage Module
SMAL	Suppress Multi-Access Link
Sng	Single
SO	Storage Operand
SP	Storage Protect; Single Pulse
SPAD	Select Parity and Display Counter
SPAR	Storage Protect Address Register
SPC	Storage Protect Check
SPF	Storage Protect Feature
SPM	(Mnemonic) Set Program Mask (RR); Storage Protect Memory
SP91	Protect Storage for System/360 Model 91
Sr	Source
SRA	(Mnemonic) Shift Right Single (RS)
SRDA	(Mnemonic) Shift Right Double (RS)
SS	Snapshot Register; Storage-to-Storage; Stepping Switch
S/S	Source/Sink
SSC	Selector Subchannel
SSK	(Mnemonic) Set Storage Key (RR)
SSM	(Mnemonic) Set System Mask (SI)
ST	(Mnemonic) Store (RX)
Stat	Station
STC	(Mnemonic) Store Character (RX)
Stg	Stage; Storage
Stk	Stack
Sto	Store; Storage
STOOP	Storage Operation
Stor	Store; Storage
Stp	Stop
STR	Source Tag Register
Sup	Suppress
SVC	(Mnemonic) Supervisor Call (RR)

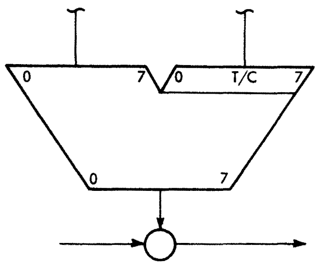
SVIR	Save Instruction Register
SVR1	Save R1 Register
Sw	Switch; Switch Enabled
SW	Single Word
Syl	Syllable
Sync	Synchronize
Sys	System
T	Time
TAT	Time Address Trigger
Tbl Wd	Table Word
TCH	(Mnemonic) Test Channel (SI)
T/CT	True/Complement Trigger
TD	Time Delay
Temp	Temporary
TERMT	Terminate Trigger
TFMT	Temporary Fetch Made Trigger
Tgr	Trigger
TI	Terminate Immediate
TIO	(Mnemonic) Test I/O (SI)
TM	(Mnemonic) Test under Mask (SI)
Tof	Turn Off
Ton	Turn on
Tot	Total
TR	(Mnemonic) Translate (SS)
Trans	Transpose
Trnsps	Transpose
TRT	(Mnemonic) Translate and Test (SS)
TS	(Mnemonic) Test and Set (SI); Timing Stack
T&S	Test and Set
U1	Unit 1
U2	Unit 2
UB	Upper Bound
UABI	Unit Address Bus In
UABO	Unit Address Bus Out
UBCTR	Upper Bound Counter
UCC	Unit Communications Control
Ucndl	Unconditional
Uncond	Uncondition
UNPK	(Mnemonic) Unpack (SS)
Val	Valid
Var	Variable
VFL	Variable Field Length
VFLEU	Variable Field Length Execution Unit
Viol	Violate; Violation
WAM	With Available Memory
WC	Word Counter
Wd	Word
Wd Bdy	Word Boundary
WR	Working Register
WRD	(Mnemonic) Write Direct (SI)
XC	(Mnemonic) Exclusive OR (SS)
Xec	Execute
XOR	Exclusive OR
ZAP	(Mnemonic) Zero and Add (SS)
ZET	Zero Test Unit
1A2	SAR 1 Loaded after SAR 2
1B2	RS 1 Loaded before RS 2
1B3	RS 1 Loaded before RS 3
1B4	RS 1 Loaded before RS 4
1C2	SAR 1 Address Compares with SAR 2 Address
2A3	SAR 2 Loaded after SAR 3
2B3	RS 2 Loaded before RS 3
2B4	RS 2 Loaded before RS 4
2C3	SAR 2 Address Compares with SAR 3 Address
3A1	SAR 3 Loaded after SAR 1
3B4	RS 3 Loaded before RS 4
3C1	SAR 3 Address Compares with SAR 1 Address

1. Data Flow Diagrams



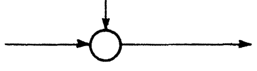
Register, Counter

Heavy line indicates input side of a functional unit that can store information. A partial transfer of contents is shown by numbered input/output lines. An input or output line connected directly to the register denotes a complete transfer of contents. An X placed in an input or an output line means that a gating condition is required to activate the transfer path. A number in a transfer path denotes the number of lines. A bar in the upper right corner means that the status of register positions is shown in indicator lights. An ALD page reference is given under the indicator bar. The bottom line within a register gives either the register position numbers or a single number indicating register size. Where register position numbers are given, a symbol such as (+3P) indicates that the register also contains 3 parity bits.

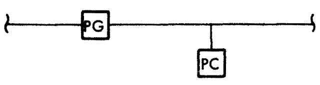


Adder, Incrementer

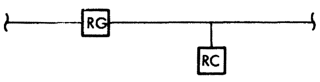
Notches across top divide logical inputs. T/C indicates that the associated input can be entered in either true or complement form.



Functional or Switching OR

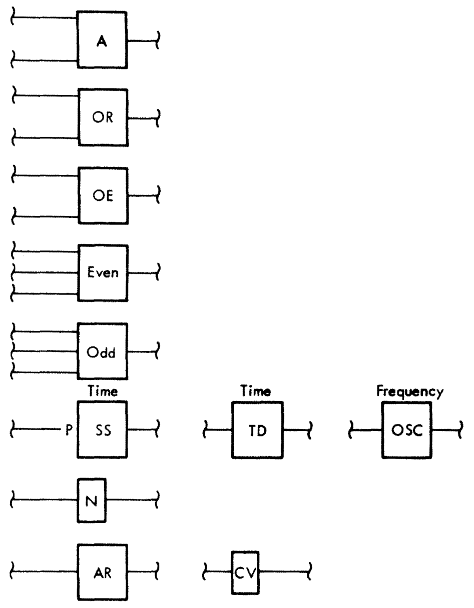


Parity Generate, Parity Check



Residue Generate, Residue Check

2. Positive Logic Diagrams



AND
The output is active only when all inputs are active.

OR
The output is active if one or more inputs is active.

Exclusive OR
The output is active only when one input is a active and the other is not.

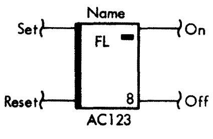
Even
The output is active only when an even number of inputs is active.

Odd
The output is active only when an odd number of inputs is active.

Singleshot, Time Delay, Oscillator

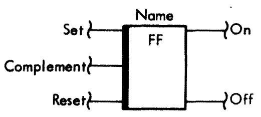
Inverter, or Negator
The output is active only when the input is not active.

Amplifier, Signal Mode Converter



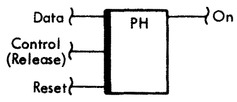
Flip Latch

Input side is denoted by thick line. Bar means that an indicator is connected to this latch. The number in the lower right corner means that this symbol represents 8 actual flip latches in the machine. An ALD page reference may be given below the block. A set input activates the on output and deactivates the off output; a reset input activates the off output and deactivates the on output. The device holds its outputs between active inputs. Simultaneous set and reset inputs activate both on and off outputs until one input is deactivated.



Flip-Flop

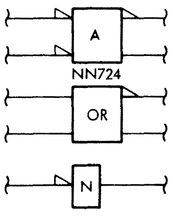
Same as flip latch except that an active complement input causes the device to switch to the opposite state. Also, on and off outputs can never be active simultaneously: Simultaneous set and reset inputs are equivalent to a complement input.



Polarity Hold

The PH output follows tbe data input when the control (release) input is activated. Between control inputs, the PH holds the previously sampled state of the data line. The reset input (when used) deactivates the output.

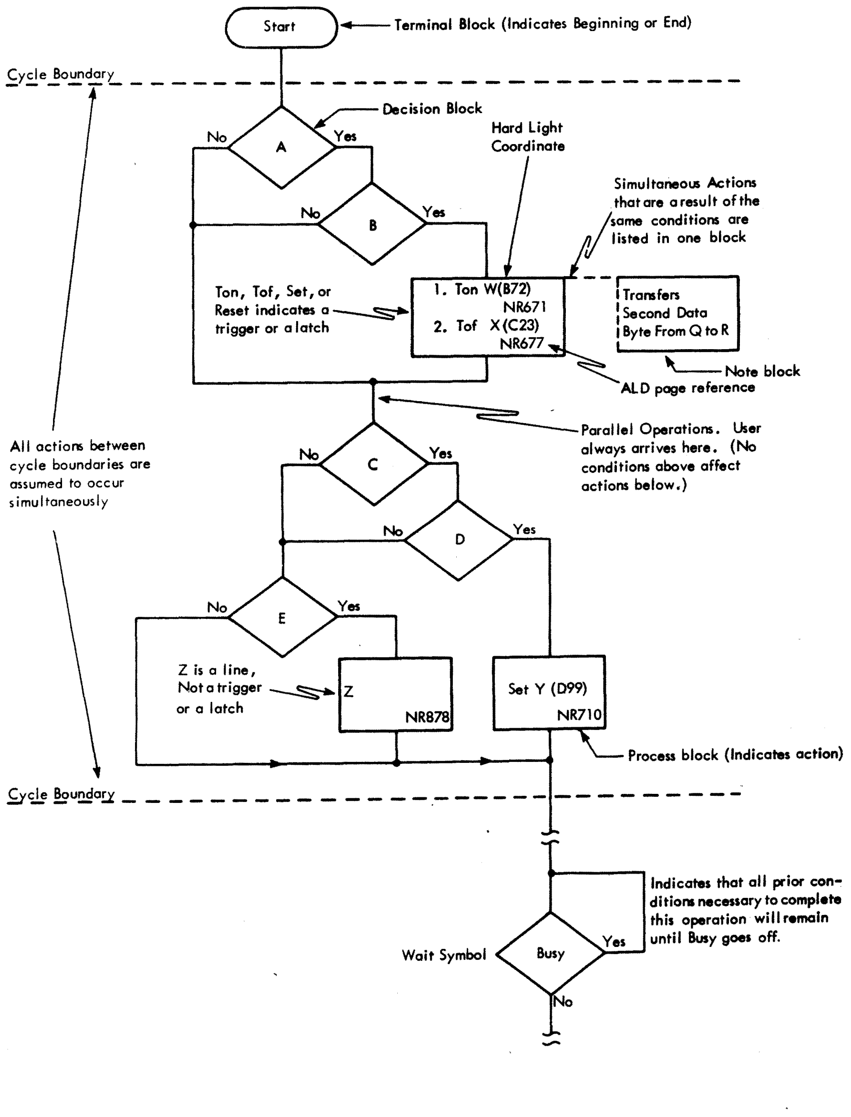
3. Simplified Logic Diagrams



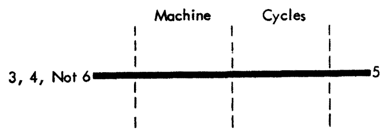
Input wedges mean that the more negative line level is required to activate the circuit; output wedges mean that the more negative line level is present when the circuit is activated. Lack of wedges indicate the more positive level. Blocks may have more than one output line. All line titles are preceded by + or - to indicate line level.

Note: Additional SLD symbology used only on ECAD's is shown in Volume 1.

4. Flowcharts

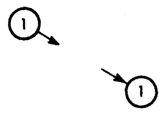

Start
Terminal Block (Indicates Beginning or End)
Cycle Boundary
Decision Block
No
A
Yes
Hard Light Coordinate
No
B
Yes
Simultaneous Actions that are a result of the same conditions are listed in one block
Ton, Tof, Set, or Reset indicates a trigger or a latch
1. Ton W(B72)
NR671
2. Tof X(C23)
NR677
Transfers Second Data Byte From Q to R
Note block
ALD page reference
Parallel Operations. User always arrives here. (No conditions above affect actions below.)
All actions between cycle boundaries are assumed to occur simultaneously
No
C
Yes
No
D
Yes
No
E
Yes
Z is a line, Not a trigger or a latch
Z
NR878
Set Y (D99)
NR710
Process block (Indicates action)
Cycle Boundary
Indicates that all prior conditions necessary to complete this operation will remain until Busy goes off.
Wait Symbol
Busy
Yes
No

5. Timing Charts



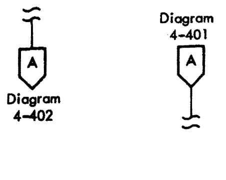
Heavy bar indicates active state. Numbers at beginning and end of the bar identify the signal(s) (also on the same chart) that activate and deactivate this line. "Not" preceding a number means that the deactive signal conditions this line.

6. General



On-Page Connector

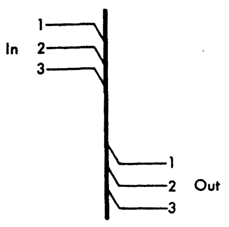
Indicates connection between two points on the same diagram. Arrow leaving symbol points to symbol with the same number.



Off-Page Connector

Indicates connection between two points located on separate pages. Where the connection is between two pages of a multipage diagram, a reference such as "Sheet 2" is given instead of a diagram number.

Text Reference Point (Reference from FETOM)



Multiple Line Transfer

READER'S COMMENT FORM

IBM 2091 Processing Unit – Volume 4
I, FXP, FLP Operations, FEMDM

Y22-6674-1

- How did you use this publication?

As a reference source □
As a classroom text □
As .... □

- Based on your own experience, rate this publication...

As a reference source:	...... Very Good	...... Good	...... Fair	...... Poor	.... Very Poor
As a text:	...... Very Good	...... Good	...... Fair	...... Poor	.... Very Poor

- What is your occupation? ....

- We would appreciate your other comments; please give specific page and line references where appropriate. If you wish a reply, be sure to include your name and address.

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

READER'S COMMENT FORM

IBM 2091 Processing Unit – Volume 4
I, FXP, FLP Operations, FEMDM

Y22-6674-1

- How did you use this publication?

As a reference source □
As a classroom text □
As .... □

- Based on your own experience, rate this publication...

As a reference source: ...... Very Good ...... Good ...... Fair ...... Poor .... Very Poor
As a text: ...... Very Good ...... Good ...... Fair ...... Poor .... Very Poor

- What is your occupation? ....

- We would appreciate your other comments; please give specific page and line references where appropriate. If you wish a reply, be sure to include your name and address.

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

YOUR COMMENTS PLEASE . . .

Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

fold fold

FIRST CLASS
PERMIT NO. 419
POUGHKEEPSIE, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM CORPORATION
P.O. BOX 390
POUGHKEEPSIE, N.Y. 12602

ATTENTION: FE MANUALS, DEPT. B95

fold

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601

YOUR COMMENTS PLEASE . . .

Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in utilizing your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

fold fold

FIRST CLASS
PERMIT NO. 419
POUGHKEEPSIE, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM CORPORATION
P.O. BOX 390
POUGHKEEPSIE, N.Y. 12602

ATTENTION: FE MANUALS, DEPT. B95

fold

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N.Y. 10601

IBM FE Supplement

System S/360 Model 91

Re: Form No. Y22-6674-1

This Supplement No. Y22-6730

Date June 14, 1968

Previous Supplement Nos. None

Field Engineering Maintenance Diagrams Manual, Volume 4
IBM System/360 Model 91

This supplement provides replacement pages and figures for IBM System/360 Model 91 FE Maintenance Diagrams Manual, Volume 4, Form Y22-6674-1. Pages and figures to be inserted and/or removed are listed below:

REMOVE	INSERT
Contents	Contents
Abbreviations	Abbreviations
5-19 (sheet 2) and 5-20	5-19 (sheet 2) and 5-20
5-118	5-118
5-132 (sheet 3) through 5-212	5-132 (sheet 3) through 5-212
5-214 through 5-216 (sheet 1)	5-214 through 5-216 (sheet 1)
5-216 (sheet 4) and 5-216 (sheet 5)	5-216 (sheet 4) and 5-216 (sheet 5)
5-219 through 5-221	5-219 through 5-221

New and changed diagrams are indicated by a FES notice in the upper outside corner, a page date of 6/68, and by a bar on the revised contents page.

Summary of Amendments

Minor changes to diagrams have been made throughout these pages.

File this cover letter at the back of the manual to provide a record of changes.

International Business Machines Corp., Product Publications Dept., Poughkeepsie, N. Y. 12602

PRINTED IN U.S.A.

FE
System
Maintenance
Library

System

cut here

Y22-6674-1

Printed in U.S.A. Y22-6674-1

IBM

International Business Machines Corporation
Field Engineering Division
112 East Post Road, White Plains, N. Y. 10601