Contract No. FA64WA-5223

IBM Field Engineering

Maintenance Diagrams

7201-02 Computing Element

SFN-0202-1

PREFACE

This manual contains the maintenance-oriented and recall diagrams referenced in the companion 7201-02 Computing Element FETOM (Form SFN-0201) and in the 7201-02 Computing Element FEMM (Form SFN-0203).

The diagrams in this manual are arranged into six categories:

Category 1. Diagnostic Techniques
Category 2. Overall Data Flow
Category 3. Data Flow by Instruction Class
Category 4. Functional Units
Category 5. Operations
Category 6. Manual Controls and Maintenance Facilities

All diagrams are in numerical order. The first digit of the diagram number reflects the category; for example, Diagram 4-210 belongs to Category 4, Functional Units. A category may be further subdivided into functional groups; for example, in Category 4, the diagrams have been grouped as follows:

Group 1. Timing and Clock Control
Group 2. ROS
Group 3. Data and Control Registers
Group 4. Local Storage
Group 5. Serial and Parallel Adders
Group 6. Status and Control Triggers
Group 7. SCI

Prerequisite and companion manuals are:

Prerequisite Manuals

9020E System Introduction, Theory of Operation Manual, Form SFN-0103
9020D System Introduction, Theory of Operation Manual, Form SFN-0104

Companion Manuals

7201-02 Computing Element, Theory of Operation Manual, Form SFN-0201
7201-02 Computing Element, Maintenance Manual, Form SFN-0203
7201-02 Computing Element, Installation Manual, Form SFN-0204
7201-02 Parts Catalog, Form SFN-0205
9020 D/E Power Controls and Distribution, Theory of Operation Manual, Form SFN-0105.

First Edition (July, 1970)

This manual has been prepared by IBM Product Publications, Kingston, N.Y.

CONTENTS

DIAGNOSTIC TECHNIQUES
Maintenance Strategy Diagram (2 Sheets) . . . 1-1
ROS Test Flowchart (3 Sheets) . . . 1-2
FLT Flowchart . . . 1-3

OVERALL DATA FLOW
7201-02 Computing Element Data Flow . . . 2-1

DATA FLOW BY INSTRUCTION CLASS
Fixed-Point Instruction Data Flow . . . 3-1
Floating-Point Instruction Data Flow . . . 3-2
Decimal and Logical Instruction Data Flow . . . 3-3
Branching Instruction Data Flow . . . 3-4
Status Switching Instruction Data Flow . . . 3-5
Input/Output Instruction Data Flow . . . 3-6
Multiple Computing Element Instruction Data Flow . . . 3-7
Display Instruction Data Flow . . . 3-8

FUNCTIONAL UNITS
Group 1: Timing and Clock Control
Clock Control Logic . . . 4-1
Reference Oscillator . . . 4-2
CE Clock Signal Generator . . . 4-3

Group 2: ROS
ROSAR (0–5) Logic . . . 4-101
ROSAR (6–9) Logic . . . 4-102
ROSAR (10) Logic . . . 4-103
ROSAR (11) Logic . . . 4-104
ROS Addressing and Data Flow (2 Sheets) . . . 4-105
Array Drivers . . . 4-106
ROS Data Register . . . 4-107

Group 3: Data and Control Registers
Q-Register B-Field Transfer Controls . . . 4-201
R-Register Transfer to LAL . . . 4-202
E-Register Incrementer, Bits 14 and 15 . . . 4-203
E-Register Parity Prediction after Incrementing . . . 4-204
Parity Adjustment for IC (21, 22) Stepping . . . 4-205
S-Register, Bits 15 and 16 . . . 4-206
AB Byte Counter . . . 4-207
ST Byte Counter . . . 4-208
Mark Trigger Logic . . . 4-209
CCR Output Logic and Control Paths (3 Sheets) . . . 4-210
LM to XY Reformatting via Mixer (2 Sheets) . . . 4-211
XY Register Parity Prediction Logic . . . 4-212
Select Register - Select Signal Generation and Response Reset . . . 4-213

Group 4: Local Storage
Local Storage Read/Write Controls . . . 4-301
9020 Out Bus to LS Data Bus Gating Logic . . . 4-302
LS Bus Parity Generation or Check . . . 4-303

Group 5: Serial and Parallel Adders
Serial Adder Input Bus Logic . . . 4-401
Carry Lookahead Logic, SAL(0–3) . . . 4-402
Decimal Add 6 Logic . . . 4-403
Decimal Correction Logic for SAL(0–3) . . . 4-404
Invalid Digit Logic . . . 4-405
Logical Functions, SAL(0) . . . 4-406
Serial Adder Parity Predict Logic . . . 4-407
Serial Adder Product-Quotient Bit Logic . . . 4-408
Gate Control Triggers for 'B+T' Micro-order . . . 4-409
Parallel Adder Bit-Position Logic (Bit 47) . . . 4-410
Parallel Adder Carry Lookahead Logic . . . 4-411
Parity Generation, PAL(48–55) . . . 4-412
Parallel Adder Half-Sum Checking Logic, PA(48–55) . . . 4-413
Parallel Adder Full-Sum Checking Logic, PA(48–55) . . . 4-414
Parallel Adder Excess 6 Logic . . . 4-415
Parallel Adder Set-Condition-Code Logic . . . 4-416

Group 6: Status and Control Triggers
STAT B Logic . . . 4-501

Group 7: SCI
SCI Request Sensing and Gating Logic . . . 4-601
Address Decode and Gating Logic . . . 4-602
SCI Control Logic for CE Clock . . . 4-603
Invalid Address and Frame Stopped Logic (2 Sheets) . . . 4-604
Storage Timeout Logic . . . 4-605
Error Handling Logic . . . 4-606
PSBAR Step Control Logic . . . 4-607
PSBAR Operations . . . 4-608
Page Control Logic and Timing . . . 4-609
SAB Parity Conversion Logic . . . 4-610
Detailed SCI Functional Sequence (2 Sheets) . . . 4-611
Servicing of Storage Requests in Single-Cycle Mode . . . 4-612

OPERATIONS
Group 1: I-Fetch
Operand Prefetching During End Op . . . 5-1
Instruction Requests During End Op . . . 5-2
Instruction Requests During Early End Op . . . 5-3
Branch Requests . . . 5-4
Selection of I-Fetch Sequence . . . 5-5
RR I-Fetch . . . 5-6
One-Cycle RR I-Fetch . . . 5-7
Two-Cycle RR I-Fetch . . . 5-8
RX I-Fetch . . . 5-9
One-Cycle RX, RS, and SI I-Fetch . . . 5-10
Two-Cycle Indexed RX I-Fetch . . . 5-11
Two-Cycle Non-Indexed RX, RS, and SI I-Fetch . . . 5-12
RS and SI I-Fetch . . . 5-13
SS I-Fetch (2 Sheets) . . . 5-14
I-Fetch Sequencers (2 Sheets) . . . 5-15
Block I-Fetch Trigger . . . 5-16
Timer Exceptional Condition . . . 5-17
CPU Store in Progress Exceptional Condition . . . 5-18
Machine Check Interruption . . . 5-19
Non-Branch Setting of Interrupt Code Triggers . . . 5-20
SPEC Y-Branch Setting of Interrupt Code Triggers . . . 5-21
Program Interruption . . . 5-22
Supervisor Call Interruption . . . 5-23
External Interruption . . . 5-24
I/O Interruption (2 Sheets) . . . 5-25
Common Interruption Routine . . . 5-26
Manual Control Exceptional Conditions . . . 5-27
Program Store Compare Exceptional Condition . . . 5-28
Invalid Instruction Address Test Exceptional Condition (2 Sheets) . . . 5-29
Test for Q-Register Refill Exceptional Condition . . . 5-30

Group 2: Fixed-Point Instructions
Load, LR (18); Load, L (58) . . . 5-101
Load Halfword, LH (48) . . . 5-102
Load and Test, LTR (12) . . . 5-103
Load Complement, LCR (13) . . . 5-104
Load Positive, LPR (10) . . . 5-105
Load Negative, LNR (11) . . . 5-106
Load Multiple, LM (98) . . . 5-107
Fixed-Point Add-Type Instructions (2 Sheets) . . . 5-108
Fixed-Point Multiply (3 Sheets) . . . 5-109
Fixed-Point Divide (6 Sheets) . . . 5-110
Convert to Binary, CVB (4F) (2 Sheets) . . . 5-111
Convert to Decimal, CVD (4E) . . . 5-112
Store, ST (50) . . . 5-113
Store Halfword, STH (40) . . . 5-114
Store Multiple, STM (90) (2 Sheets) . . . 5-115
Shift Left Single, SLA (8B) (2 Sheets) . . . 5-116
Shift Left Double, SLDA (8F) (4 Sheets) . . . 5-117
Shift Right Single, SRA (8A) (3 Sheets) . . . 5-118
Shift Right Double, SRDA (8E) (4 Sheets) . . . 5-119

Group 3: Floating-Point Instructions
Save Signs and Insert Sign Functions, and CC Setting . . . 5-201
Load, LER (38) – Short Operands; Load, LDR (28) – Long Operands . . . 5-202
Load, LE (78) – Short Operands; Load, LD (68) – Long Operands . . . 5-203
Load Positive, LPER (30); Load Negative, LNER (31); Load and Test, LTER (32); Load Complement, LCER (33) – Short Operands . . . 5-204
Load Positive, LPDR (20); Load Negative, LNDR (21); Load and Test, LTDR (22); Load Complement, LCDR (23) – Long Operands . . . 5-205
Floating-Point Add, Subtract, and Compare – Short Operands (5 Sheets) . . . 5-206
Floating-Point Add, Subtract, and Compare – Long Operands (5 Sheets) . . . 5-207
Halve, HER (34) – Short Operands . . . 5-208
Halve, HDR (24) – Long Operands . . . 5-209
Floating-Point Multiply Data Paths . . . 5-210
Floating-Point Multiply, Short Operands (4 Sheets) . . . 5-211
Floating-Point Multiply, Long Operands (4 Sheets) . . . 5-212
Floating-Point Divide Data Paths . . . 5-213
Floating-Point Divide, Short Operands (4 Sheets) . . . 5-214
Floating-Point Divide, Long Operands (5 Sheets) . . . 5-215
Store, STE (70) – Short Operands; Store, STD (60) – Long Operands . . . 5-216

Group 4: Decimal Instructions
GIS for Decimal Add, Subtract, and Compare . . . 5-301
True Add Sequence for Decimal Add, Subtract, and Compare (3 Sheets) . . . 5-302
Complement Add Sequence for Decimal Add, Subtract, and Compare (3 Sheets) . . . 5-303
Zero and Add (4 Sheets) . . . 5-304
Decimal Multiply (7 Sheets) . . . 5-305
Decimal Divide (9 Sheets) . . . 5-306
GIS for Pack, Unpack, and Move With Offset . . . 5-307
Pack, Not Word Overlap Sequence . . . 5-308
Pack, Word Overlap Sequence . . . 5-309
Unpack, Not Word Overlap Sequence . . . 5-310
Unpack, Word Overlap Sequence . . . 5-311
Move With Offset, Not Word Overlap Sequence . . . 5-312
Move With Offset, Word Overlap Sequence . . . 5-313

Group 5: Logical Instructions
GIS for Logical Instructions . . . 5-401
Logical Move Instructions . . . 5-402
Logical Compare Instructions . . . 5-403
Logical AND Instructions . . . 5-404
Logical OR Instructions . . . 5-405
Logical Exclusive-OR Instructions . . . 5-406
Test Under Mask, TM (91) . . . 5-407
Insert Character, IC (43); Store Character, STC (42) . . . 5-408
Load Address, LA (41) . . . 5-409
Translate, TR (DC); Translate and Test, TRT (DD) . . . 5-410
Edit, ED (DE); Edit and Mark, EDMK (DF) . . . 5-411
Logical Shift Instructions . . . 5-412

Group 6: Branching Instructions
Branch On Condition, BCR (07); BC (47) (2 Sheets) . . . 5-501
Branch and Link, BALR (05) (2 Sheets) . . . 5-502
Branch and Link, BAL (45) (2 Sheets) . . . 5-503
Branch On Count, BCTR (06); BCT (46) (2 Sheets) . . . 5-504

Branch on Index High, BXH (86); Branch on Index Low or Equal, BXLE (87) (3 Sheets) . . . 5-505
Execute, EX (44) (2 Sheets) . . . 5-506

Group 7: Status Switching Instructions
Load PSW, LPSW (82) . . . 5-601
Set Program Mask, SPM (04) . . . 5-602
Set System Mask, SSM (80) . . . 5-603
Supervisor Call, SVC (0A) . . . 5-604
Set Storage Key, SSK (08) . . . 5-605
Insert Storage Key, ISK (09) . . . 5-606
Write Direct, WRD (84) . . . 5-607
Read Direct, RDD (85) . . . 5-608
Diagnose (83) (3 Sheets) . . . 5-609

Group 8: I/O Instructions
I/O Instructions . . . 5-701

Group 9: Multiple Computing Element Instructions
Load Identity, LI (0C) . . . 5-801
Insert ATR, IATR (0E) . . . 5-802
Delay, DLY (0B) . . . 5-803
Store PSBAR, SPSB (A0) . . . 5-804
Load PSBAR, LPSB (A1) . . . 5-805
Move Word, MVW (D8) (3 Sheets) . . . 5-806
Start I/O Processor, SIOP (9A) . . . 5-807
Set Address Translator, SATR (0D), Execution in Issuing CE (6 Sheets) . . . 5-808
Set Address Translator, SATR (0D), Execution in Receiving CE (3 Sheets) . . . 5-809
Set Configuration, SCON (01) (6 Sheets) . . . 5-810
Test and Set, TS (93) . . . 5-811

Group 10: Display Instructions
Repack Symbols, Simplified Flow Chart . . . 5-901
Repack Symbols, RPSB (0F) (21 Sheets) . . . 5-902
Convert and Sort Symbols, CSS (02) (10 Sheets) . . . 5-903
Convert Weather Lines, Simplified Flowchart . . . 5-904
Convert Weather Lines, CVWL (03) (9 Sheets) . . . 5-905
Load Chain, LC (52) . . . 5-906

MANUAL CONTROLS AND MAINTENANCE FEATURES
Group 1: 7201-02 CE Console Controls
CE Control Panel (2 Sheets) . . . 6-1
CE Roller Switch Indicators (2 Sheets) . . . 6-2
Pushbutton Signal Generation (2 Sheets) . . . 6-3
Stop Loop Routine (2 Sheets) . . . 6-4
Stop Loop Monitored Pushbutton Gating . . . 6-5
Stop, Manual, Address Compare Triggers, and Block Interrupt Latch (2 Sheets) . . . 6-6
CE Machine Reset and Force Address . . . 6-7
System Operation: IPL or PSW Restart . . . 6-8A
Subsystem Operation: IPL or PSW Restart . . . 6-8B
Common Routine: IPL or PSW Restart (2 Sheets) . . . 6-9
STORAGE SELECT Switch Gating . . . 6-10
DEFEAT INTERLEAVING Switch Gating . . . 6-11
RATE Switch Logic . . . 6-12
Instruction Step Routine . . . 6-13
Single-Cycle and Single-Cycle-Inhibit Routine . . . 6-14
Repeat Instruction Switch Logic . . . 6-15
Repeat Instruction Switch Routine . . . 6-16
ROS TRANSFER and REPEAT ROS ADDRESS Switch Gating . . . 6-17
Storage Ripple Loop (Store and Display) Routine . . . 6-18
Wait State Gating . . . 6-19
Wait State Microprogram Routine . . . 6-20
Disable Interval Timer Logic . . . 6-21
CE Check Control and Inhibit CE Hardstop Switches, Logic and Error Controls . . . 6-22
Pulse Mode Controls . . . 6-23
Pulse Mode Operation . . . 6-24
LOG OUT Pushbutton Logic . . . 6-25
SCAN MODE, ROS/PROC/FLT Switch Logic . . . 6-26
FLT BACKSPACE Pushbutton Logic and Flow . . . 6-27
1052 Adapter Unit . . . 6-28*
1052 Adapter Initial Selection – Read, Write, Sense . . . 6-29*
1052 Adapter Data Transfer – Write . . . 6-30*
1052 Adapter Data Transfer – Read . . . 6-31*
1052 Adapter Ending Sequence . . . 6-32*
1052 Adapter Sense and Status Bytes . . . 6-33*

Group 2: Maintenance Features
Scan Data and Control . . . 6-101
Scan Clock . . . 6-102
FLT Clock . . . 6-103
Scan Counter Latches and Decrementer . . . 6-104
Scan Storage Address Generator . . . 6-105
FLT Counter Decrementing . . . 6-106
Scan-Out Bus Data Flow . . . 6-107
Logout Control Logic . . . 6-108
Scan-Out Path For One Bit . . . 6-109
Maintenance Mode Stop Clock Logic . . . 6-110
Scan Control Triggers . . . 6-111
Scan Control of ROS Microbranching . . . 6-112
CE Scan/IOCE Interface . . . 6-113
Logout Sequence (2 Sheets) . . . 6-114
ROS Test Sequence (5 Sheets) . . . 6-115
FLT Sequence (5 Sheets) . . . 6-116
CE Logword Formats (3 Sheets) . . . 6-117
SE Logword Formats . . . 6-118
DE Logword Formats . . . 6-119

Group 3: DE Wrap Operation
DE Wrap Bus Controls . . . 6-201

INDEX . . . X-1

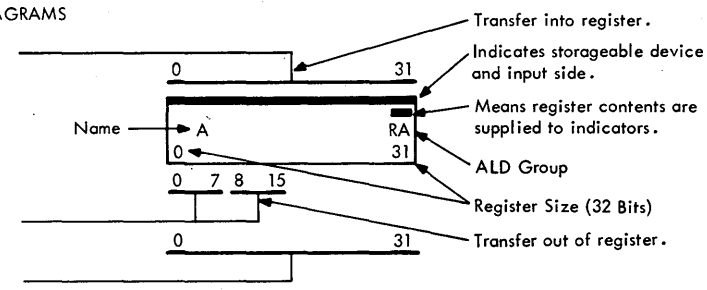
*Note: 1052 Adapter is used only with the 9020E configuration.

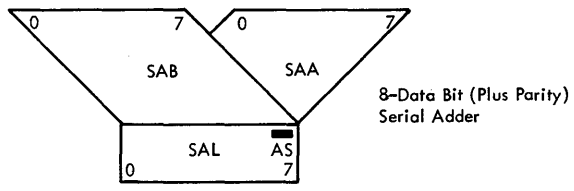
ABBREVIATIONS

ABC	AB register byte counter
ac	alternating current
ACR	Automatic Carrier Return
adr	address, addressed, addressing
ALD	automated logic diagram
ALTN	Alternate
amp	ampere
APSA	alternate preferential storage area
ASC	address store compare
ATC	air traffic control
ATN	alternate test number
ATR	address translation register
Attn	attention
Aux	Auxiliary Magnet
BCD	binary-coded decimal
BCU	bus control unit (alternate terminology for SCI)
BL	blink
BR	brightness
BSM	basic storage module
C	capacitor
CAS	control automation system
CAW	channel address word
CB	circuit breaker
CC	condition code, also Configuration Console
CCC	Central Computer Complex
CCR	configuration control register
CCW	channel command word
CE	Computing Element
Charistic	Characteristic
CLD	control automation system logic diagram
Cmd	command
CPU	Central Processing Unit (alternate terminology for CE)
CR	diode or Carrier Return
CROS	capacitive read-only storage
CSW	channel status word
CT	conditional terminate
CTC	channel-to-channel
CU	Control Unit
CVG	Character Vector Generator
DA	dash
DAR	diagnose accessible register
DARM	diagnose accessible register mask
DAU	Data Adapter Unit
dc	direct current
DCP	Display Channel Processor
DE	Display Element
dec	decimal
dec div	decimal divide
dec ovflo	decimal overflow
DG	Display Generator
Disc	disconnect
dly	delay
Dply	display
dsbl	disable
DX	first byte in a series of destination bytes
DX+1	second byte in a series of destination bytes
DX+2	third byte in a series of destination bytes
ELC	element check
end op	end operation
EOB	end of block
EOL	End-of-Line
EPO	emergency power off
ERSLT	expected result
EXC	Executive Control Program
exp ovflo	exponent overflow
exp unflo	exponent underflow
F	fuse
FEMDM	Field Engineering Maintenance Diagrams Manual
FEMI	Field Engineering Manual of Instruction
FEMM	Field Engineering Maintenance Manual
FETOM	Field Engineering Theory of Operation Manual
fix-pt ovflo	fixed-point overflow
FLT	fault locating test
flt-pt div	floating-point divide
FMTN	Format New
FMTO	Format Old
FMTW	Format Weather
FPR	Floating-point register
fract	fraction

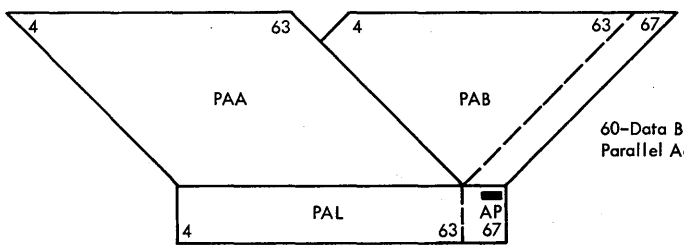
GIS	general initialization sequence
GPR	general-purpose register
hex	hexadecimal
Hz	Hertz
IC	instruction counter
ICR	inhibit carrier return
IDES	inhibit display element stop
I-Fetch	instruction fetching
ILC	instruction length code
ILOS	inhibit logout stop
Init	initial
I/O	input/output
IOCE	Input/Output Control Element
IPL	initial program load
K	kilo; also relay
kHz	kilohertz
LAB	logical address bus
LADS	Logic Automation Documentation System
LAL	local storage address latches
LAR	local storage address register
LC	lower case
LF	line feed
LOS	logout stop
LS	local store
LSWR	local storage working register
MACH	maintenance and channel (storage)
max	maximum
MC	machine check
MCW	maintenance control word
mHz	megahertz
MMSC	maintenance mode stop clock
Mple	Multiple
MPR	multiplier
MPX	multiplex
ms	millisecond
NDT	new descriptor tables
no op	no operation
NRM	new refresh memory
NRMA	new refresh memory address
ns	nanosecond
OBS	on battery signal
ODT	old descriptor tables
op code	operation code
oper	operation
opr	operand
ORM	old refresh memory
ORMA	old refresh memory address
OTC	out of tolerance check
P	parity
PAA	parallel adder A-side
PAB	parallel adder B-side
PAL	parallel adder latch
PB	pushbutton
pf	picofarad
PK	power contactor
PP	partial product
PQ	partial quotient
priv oper	privileged operation
proc	process
prog	program
PROSAR A	previous read-only storage address register A
PROSAR B	previous read-only storage address register B
prot	protection
PS	power supply
PSA	preferential storage address
PSBA	preferential storage base address
PSBAR	preferential storage base address register
PSW	program status word
PVD	Plan View Display
R	resistor
RCU	Reconfiguration Control Unit
reg	register
RKM	Radar Keyboard Multiplexor
ROS	read-only storage
ROSAR	read-only storage address register
ROSBR	read-only storage backup register
ROSDR	read-only storage data register
RST	Reset
SAA	serial adder A-side
SAB	storage address bus, also serial adder B-side
SAL	serial adder latch
SATR	set Address Translation Register
SBA	serial adder bus A
SBB	serial adder bus B
SC	System Console
SCI	storage control interface
SCON	set Configuration Control Register
SCOPEX	scoping index
SCR	silicon-controlled rectifier
SDBI	storage data bus in
SDBO	storage data bus out
SE	Storage Element
Sel	select
Serv	service
signif	significance
SLT	solid logic technology
SMMC	system maintenance monitor console
SMS	standard modular system
SOROS	scan out read-only storage
spec	specification
SRL	Systems Reference Library
SSU	storage switching unit
STAT	status trigger
STC	ST register byte counter
stg	storage
SU	switch unit
sync	synchronizing
T	transformer
TC	time clock (interval timer)
TCU	tape control unit
T(DX)	table byte specified by DX
T(DX+1)	table byte specified by DX+1
TIC	transfer in channel
TN	test number
T/R	tilt/rotate
TU	tape unit
uc	upper case
uf	microfarad
usec	microsecond
UT	unconditional terminate
V	volt
VFL	variable-field length
VFR	visual flight rules
Xlat	translate
$\geq$	greater than or equal to
$\geq$	greater than or equal to
$\leq$	less than or equal to
$\leq$	less than or equal to
=	equal to
$\neq$	not equal to
&	and

LOGIC DIAGRAMS

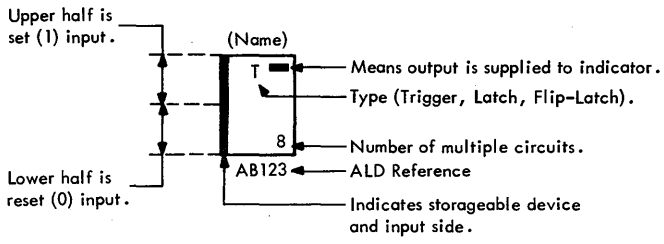


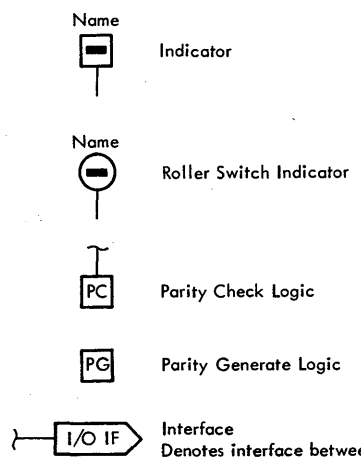


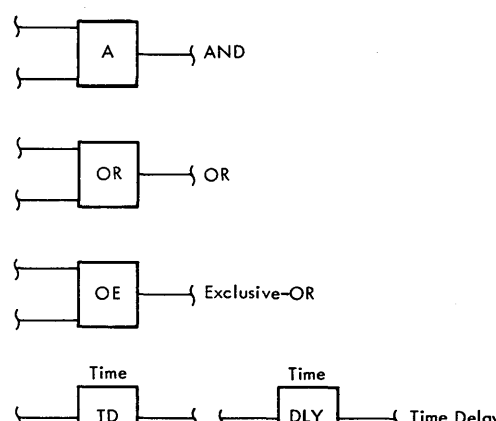
8-Data Bit (Plus Parity) Serial Adder

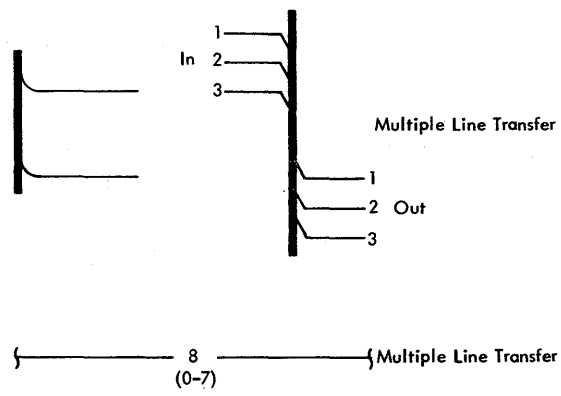


60-Data Bit (Plus Parity) Parallel Adder

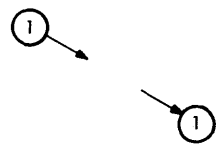








GENERAL

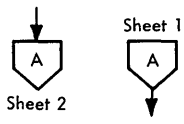


On-Page Connector

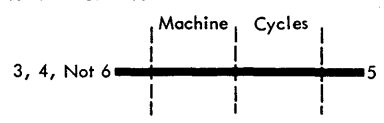
Indicates connection between two parts of the same diagram. Arrow leaving symbol points (line of sight) to correspondingly-numbered symbol.

Off-Page Connector

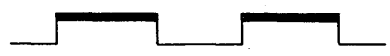
Indicates connection between sheets of the same diagram.



TIMING CHARTS

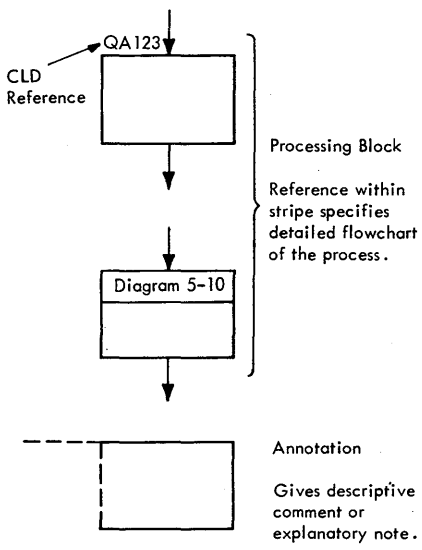


Heavy bar indicates active state. Number(s) at beginning and end of the bar identify the signal(s) (also on the same chart) that activate and deactivate this line. "Not" preceding a number means that the deactive signal conditions this line.



Waveshape
Heavy bar indicates active state.

FLOWCHARTS

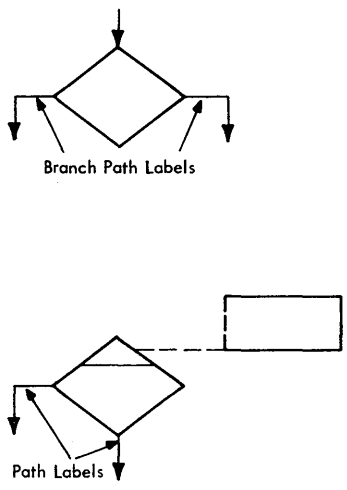


Processing Block

Reference within stripe specifies detailed flowchart of the process.

Annotation

Gives descriptive comment or explanatory note.



Decision Block

Indicates point in flowchart where branch to alternate paths is possible. Diagram reference within annotation block connected to stripe specifies detailed flowchart of the decision.

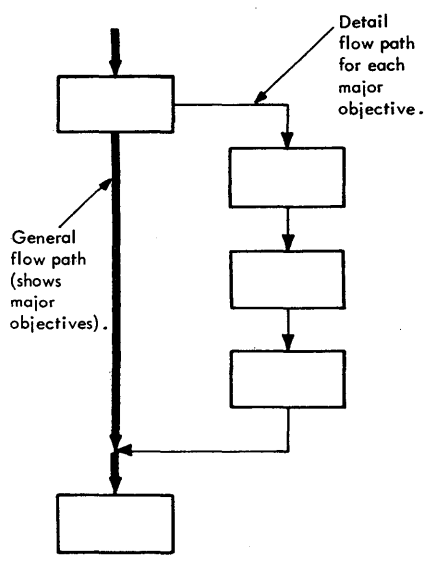


Diagram 1-1. Maintenance Strategy Diagram (Sheet 1 of 2)

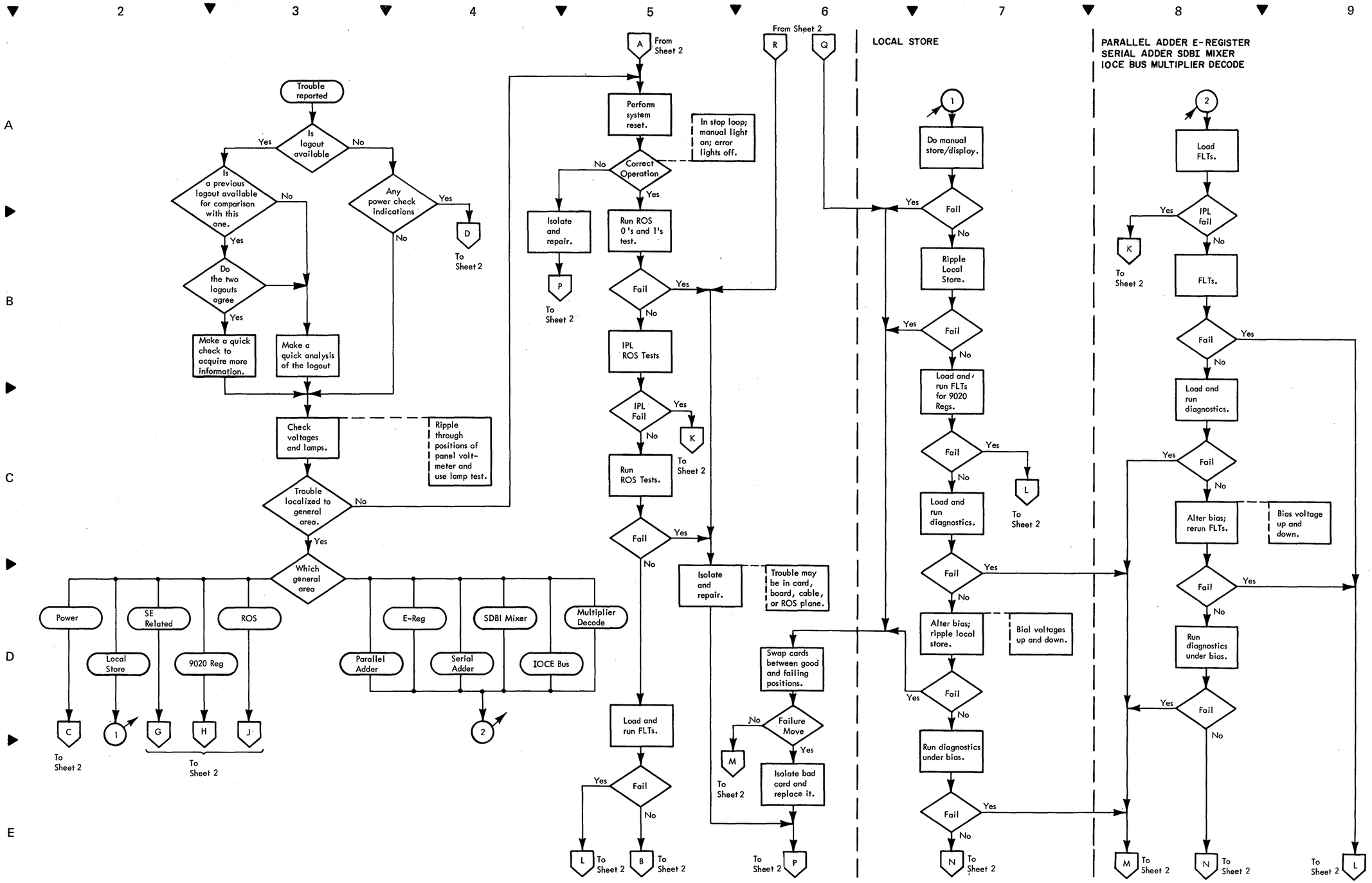
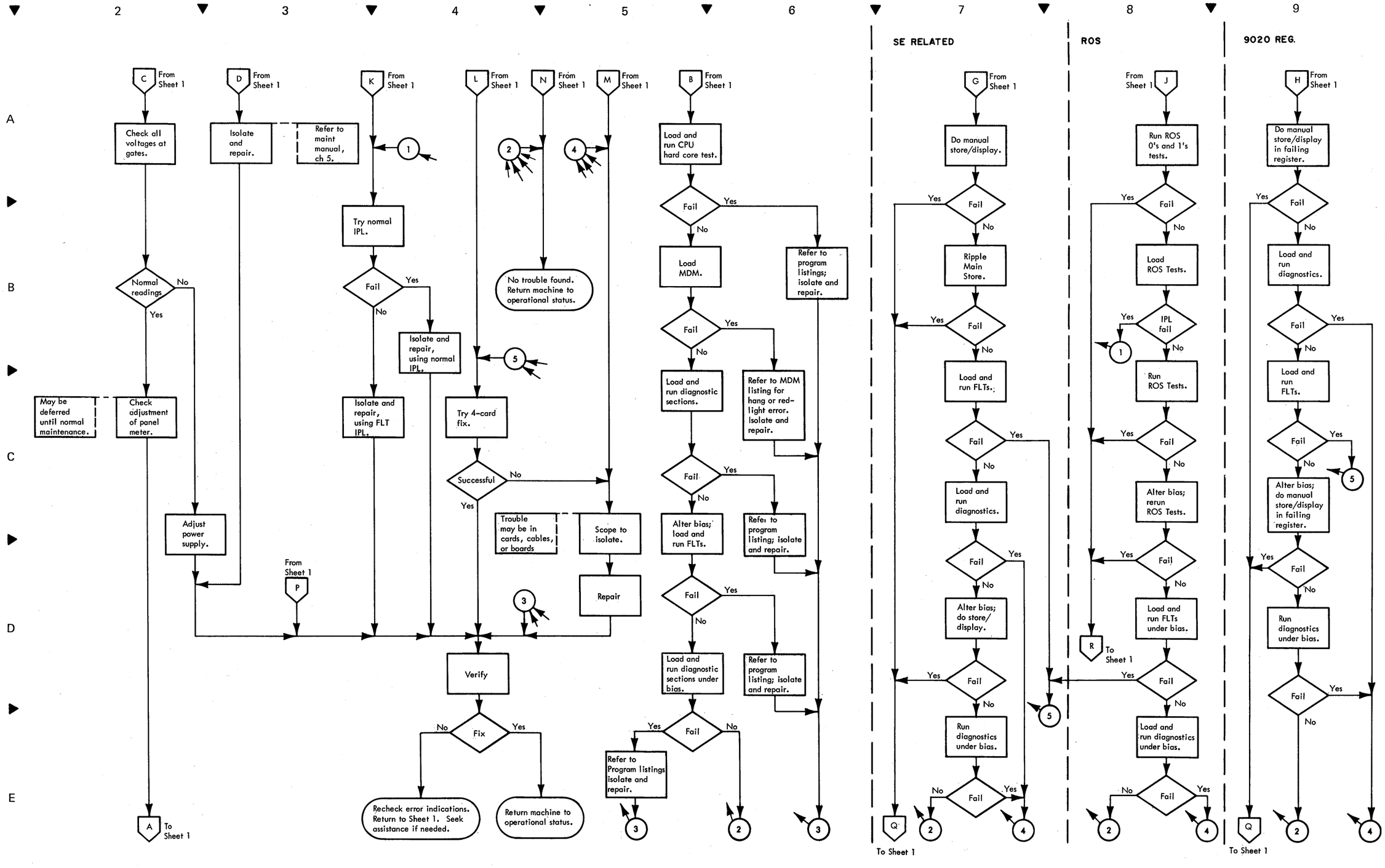


Diagram 1-1. Maintenance Strategy Diagram (Sheet 2 of 2)



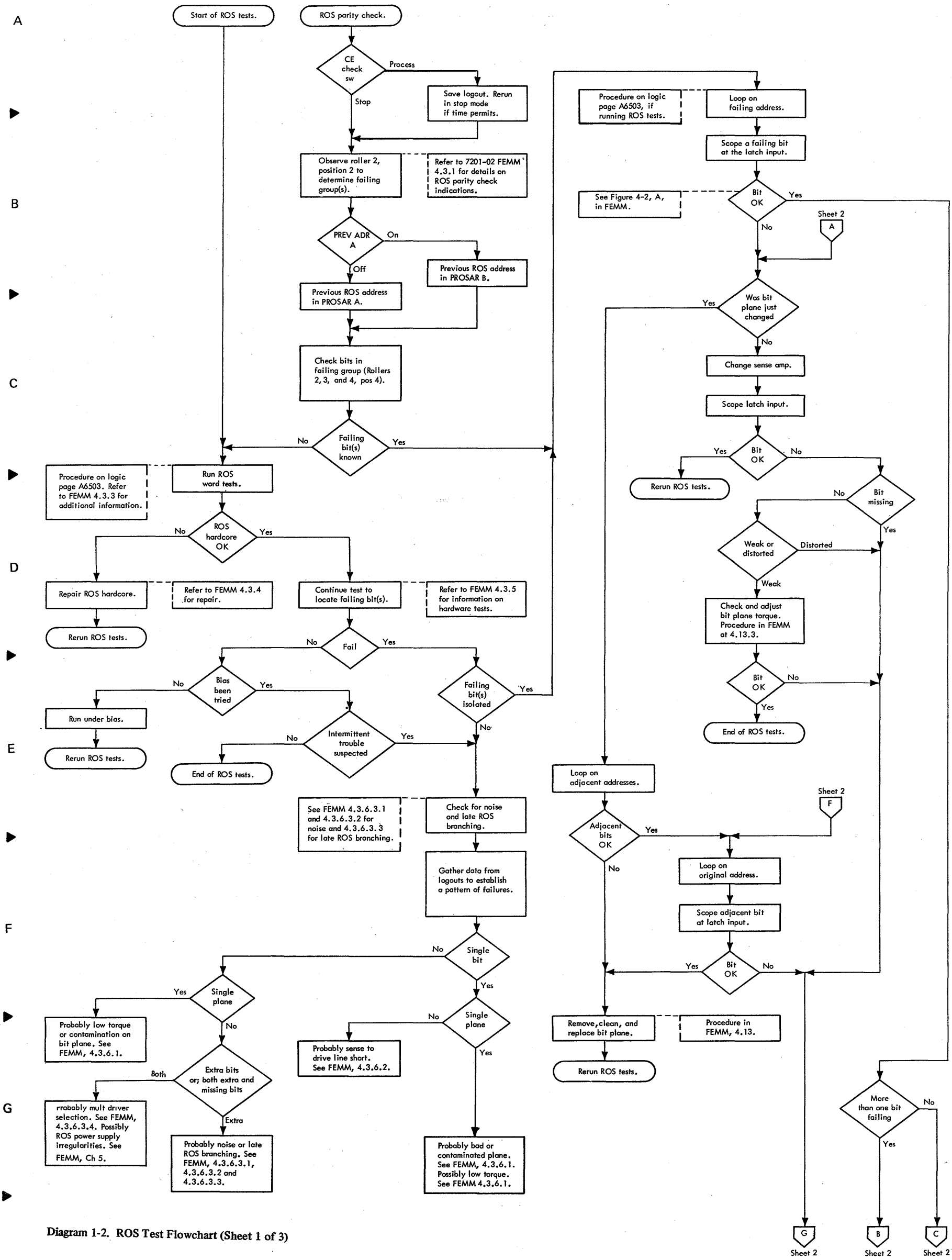


Diagram 1-2. ROS Test Flowchart (Sheet 1 of 3)

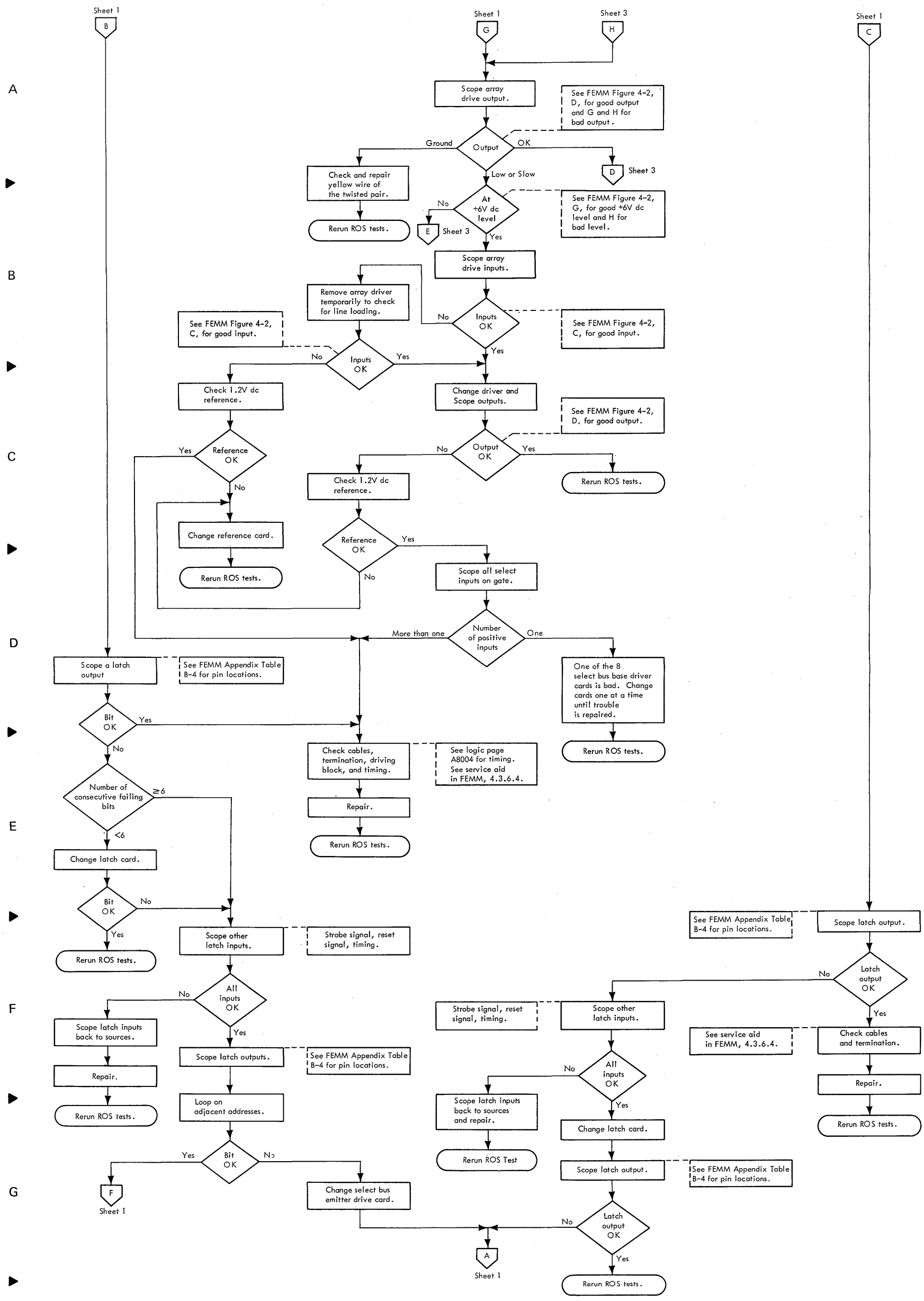
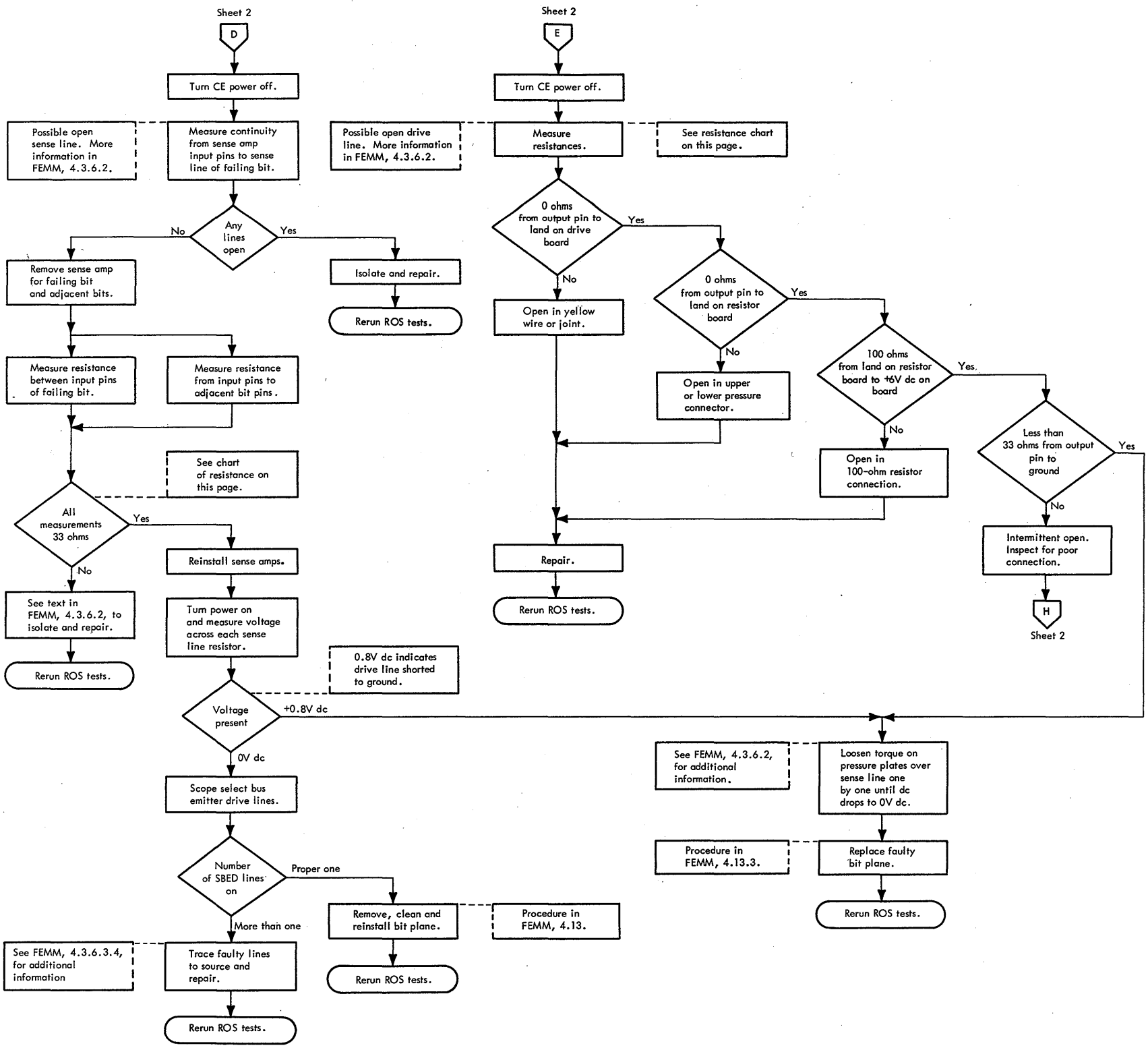


Diagram 1-2. ROS Test Flowchart (Sheet 2 of 3)



Resistance Readings

Condition	From Sense Amp Input to	Resistance
Normal	DC return. Any other sense amp input. Any drive/balance line.	16.5 ohms 33.0 ohms Open
Sense - sense line short	DC return. Input of sense amp to which it is shorted. Any other sense amp input. Any drive/balance line.	8.25 ohms 0 ohm 24.75 ohms Open
Sense - DC return short	DC return. Any other sense amp input. Any drive/balance line.	0 ohm 16.5 ohms Open
Sense - drive/balance line short	DC return. Drive/balance line to which it is shorted. Any drive/balance line in same or opposite plane, except line to which it is shorted. Any drive/balance line not in same or opposite plane.	16.5 ohms 0 ohm 200 ohms plus short resistance Open

Diagram 1-2. ROS Test Flowchart (Sheet 3 of 3)

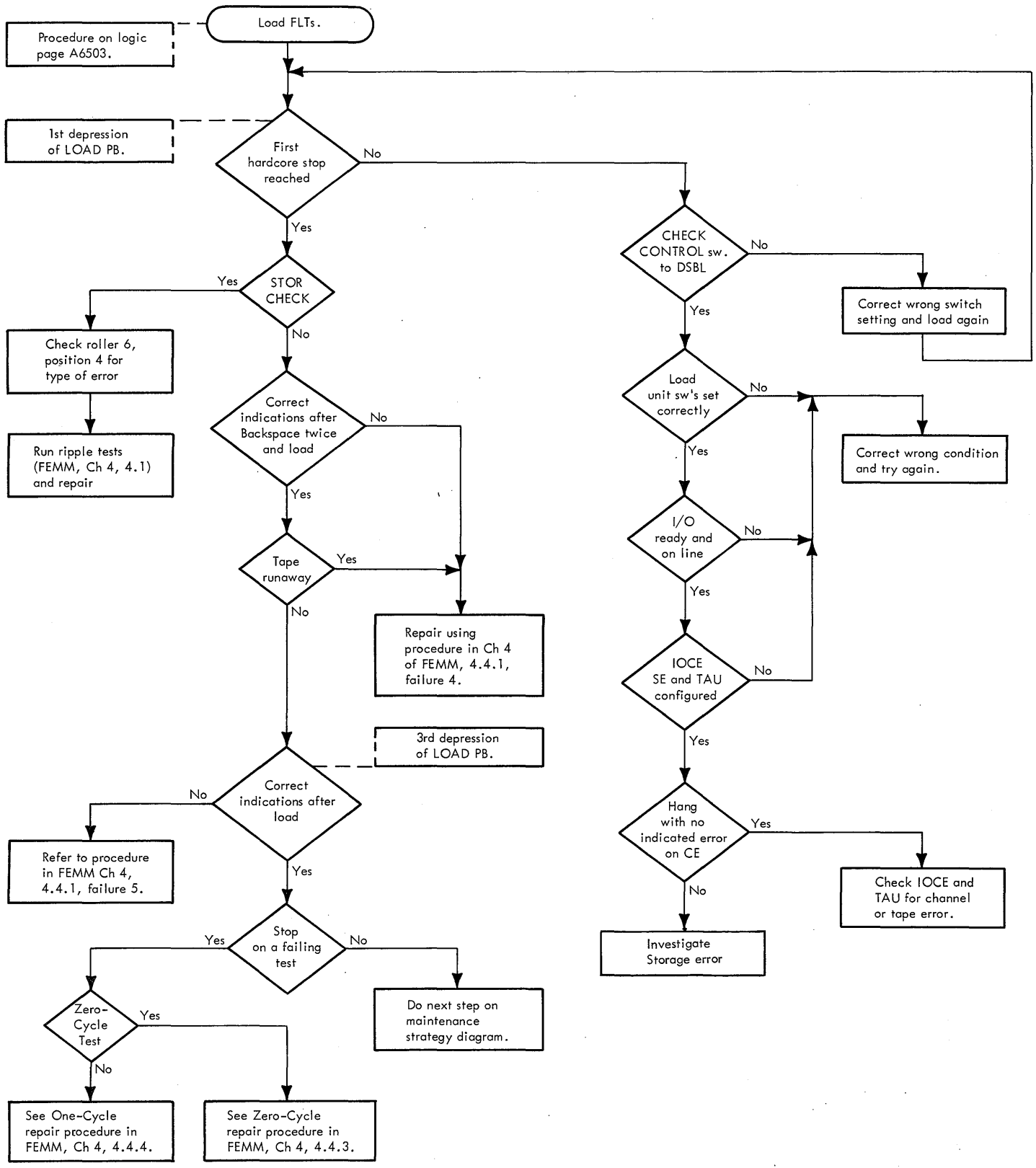


Diagram 1-3. FLT Flowchart

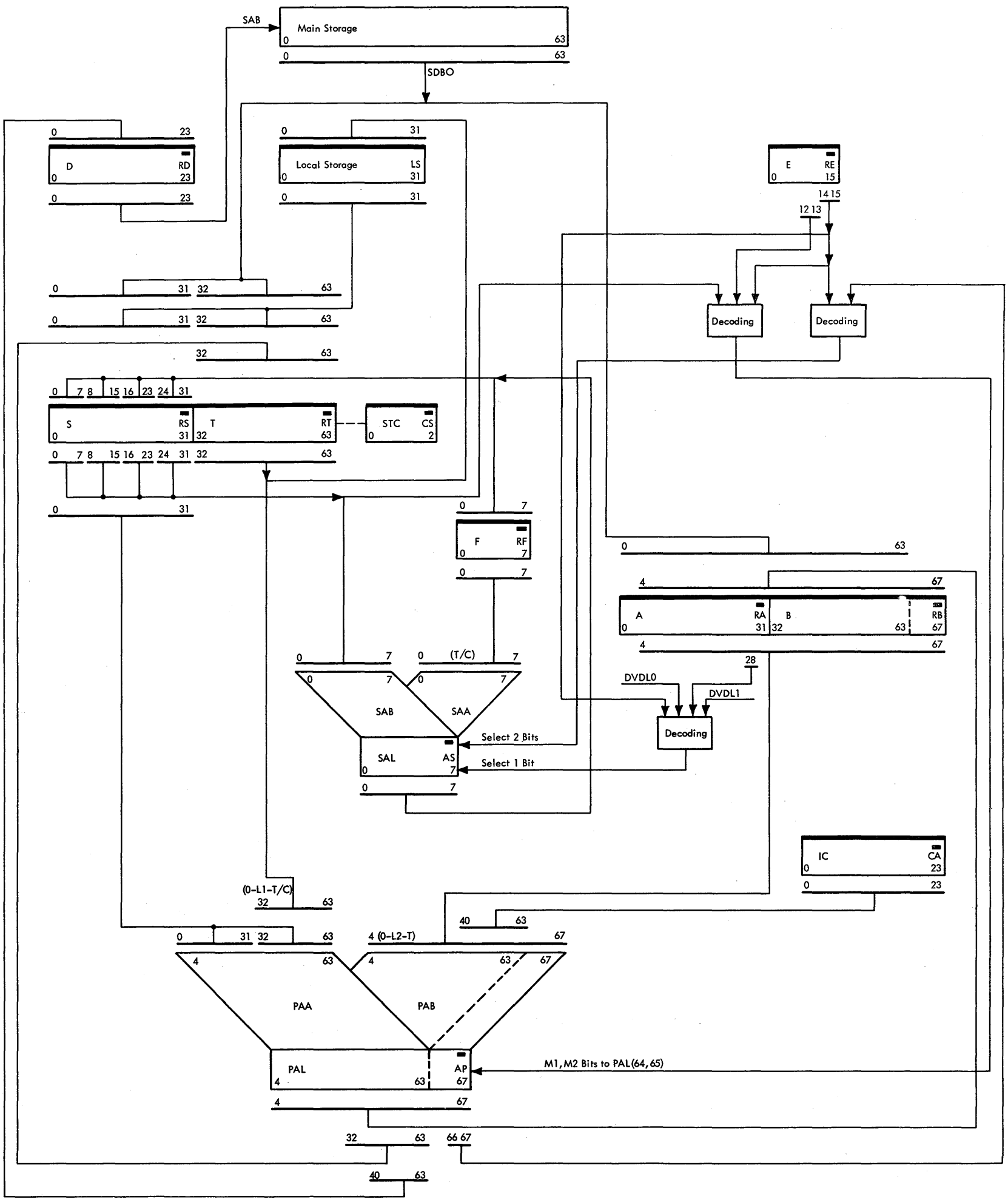


Diagram 3-1. Fixed Point Instruction Data Flow

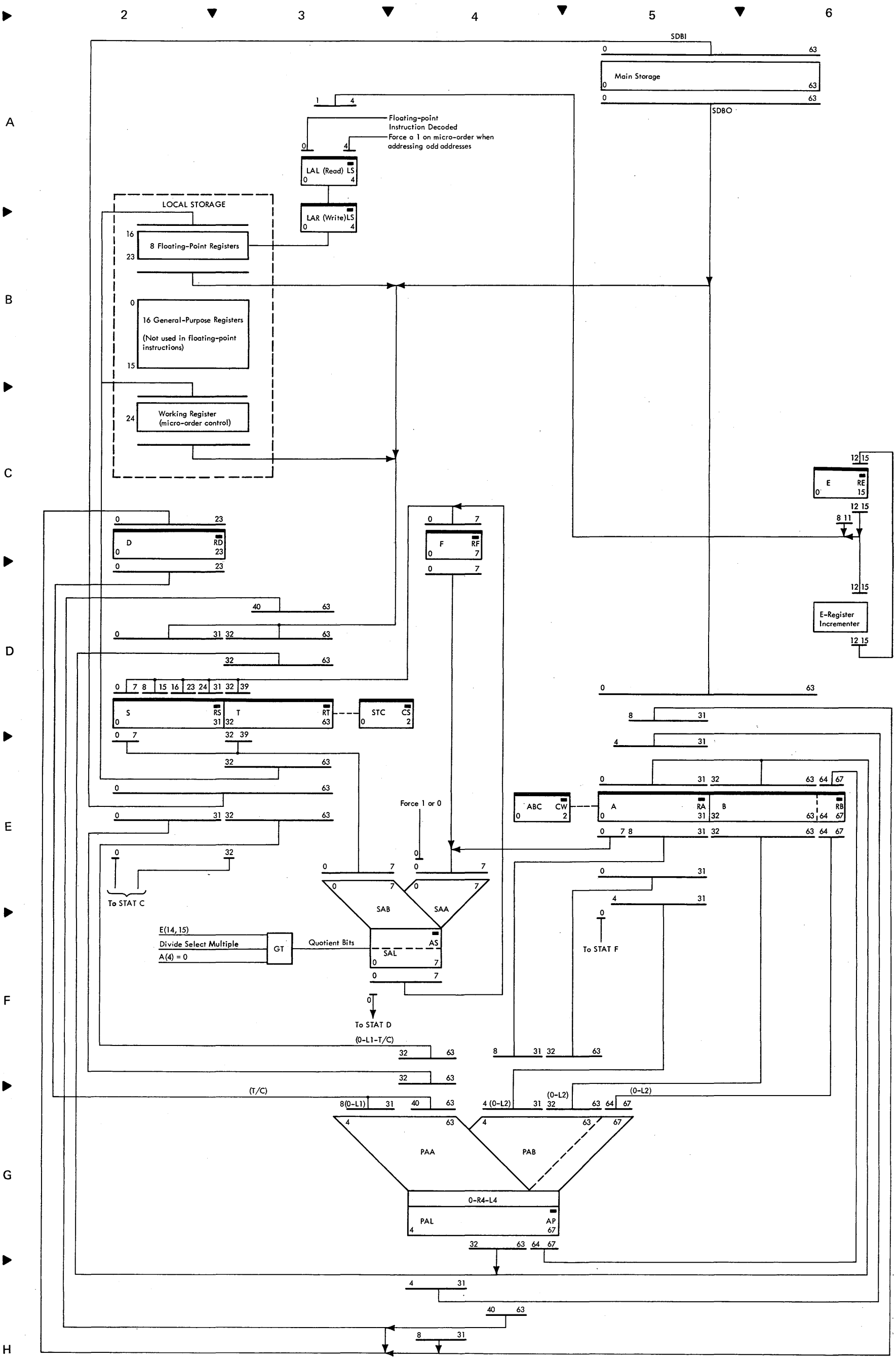


Diagram 3-2. Floating-Point Instruction Data Flow

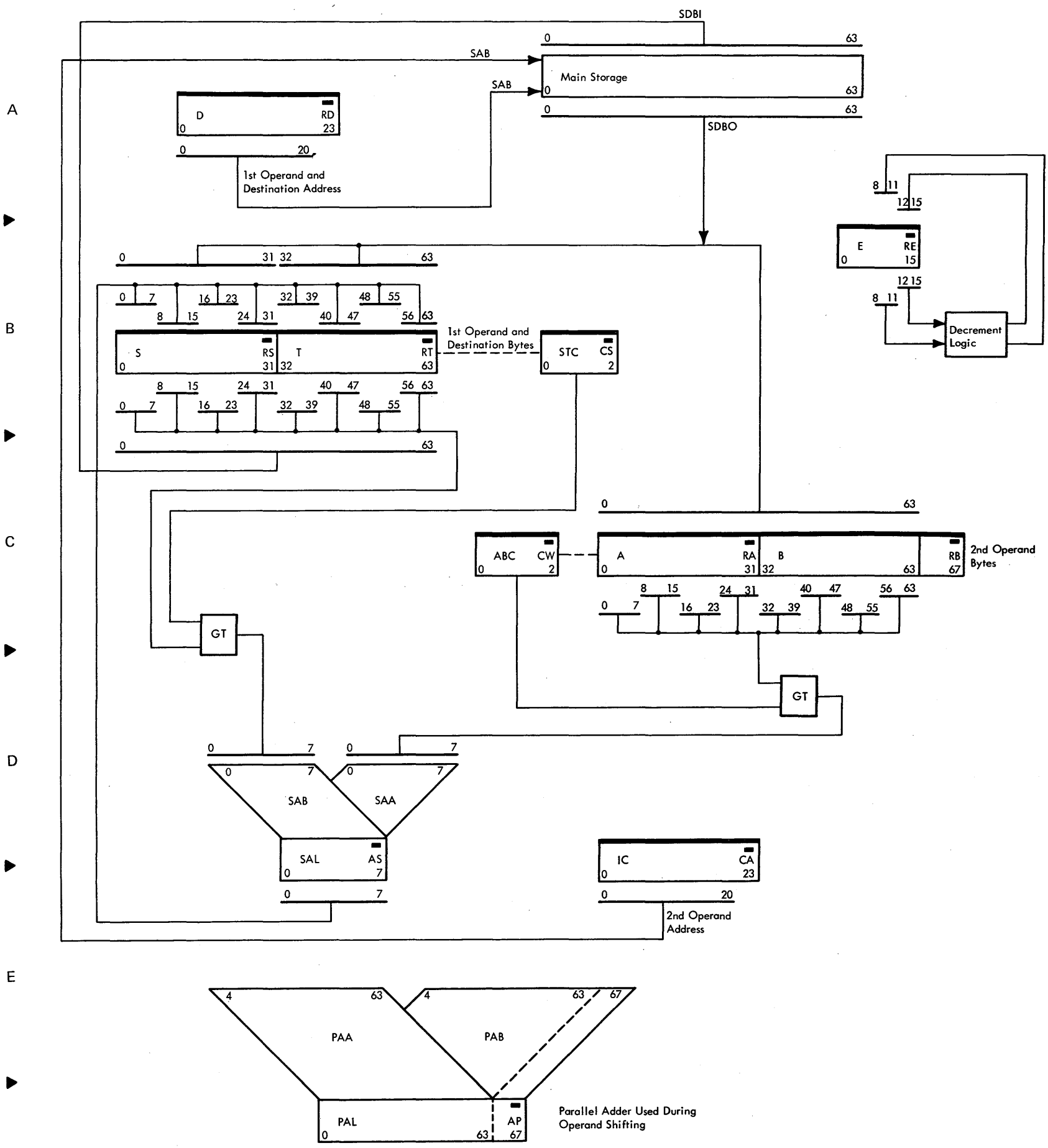


Diagram 3-3. Decimal and Logical Instruction Data Flow

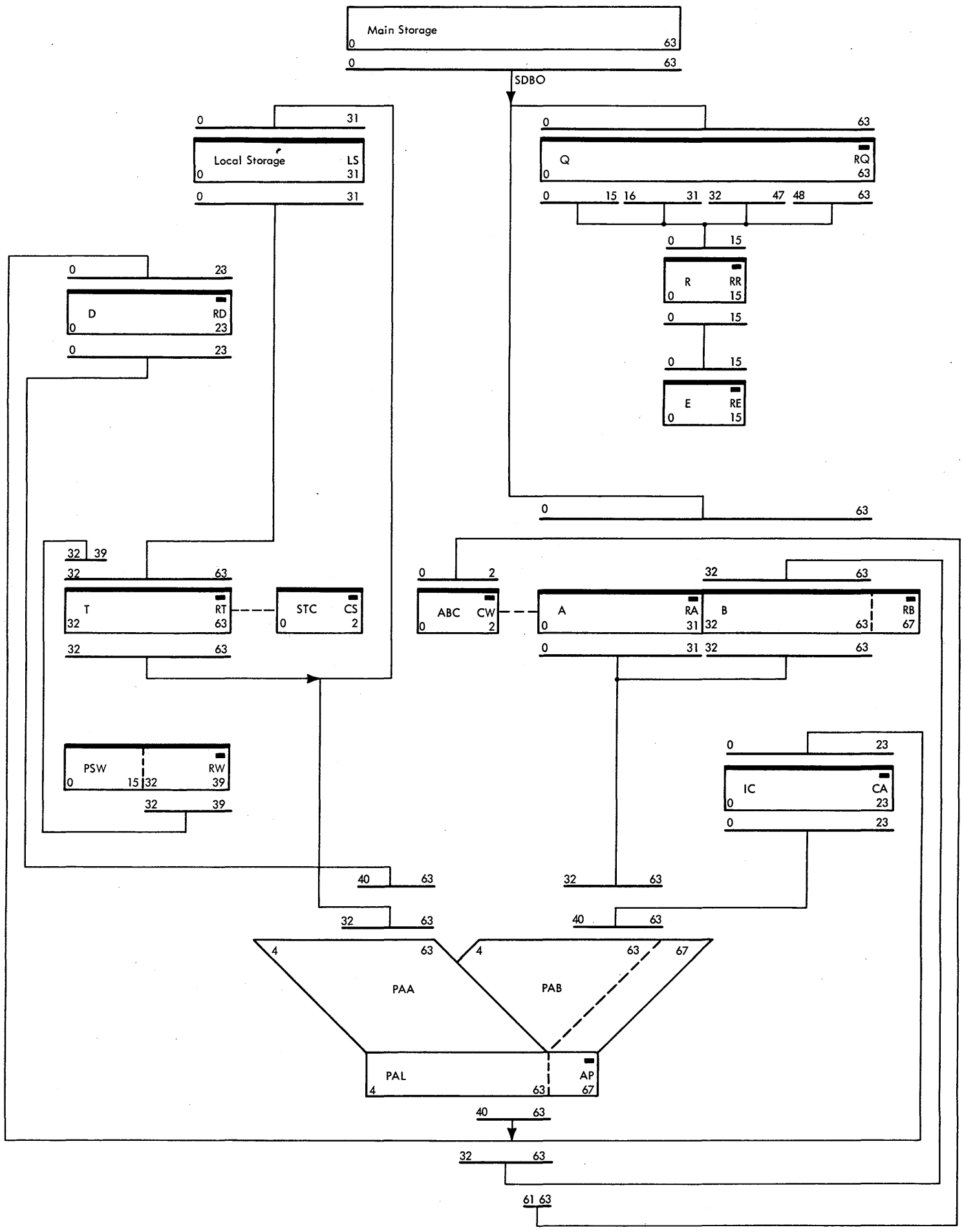


Diagram 3-4. Branching Instruction Data Flow

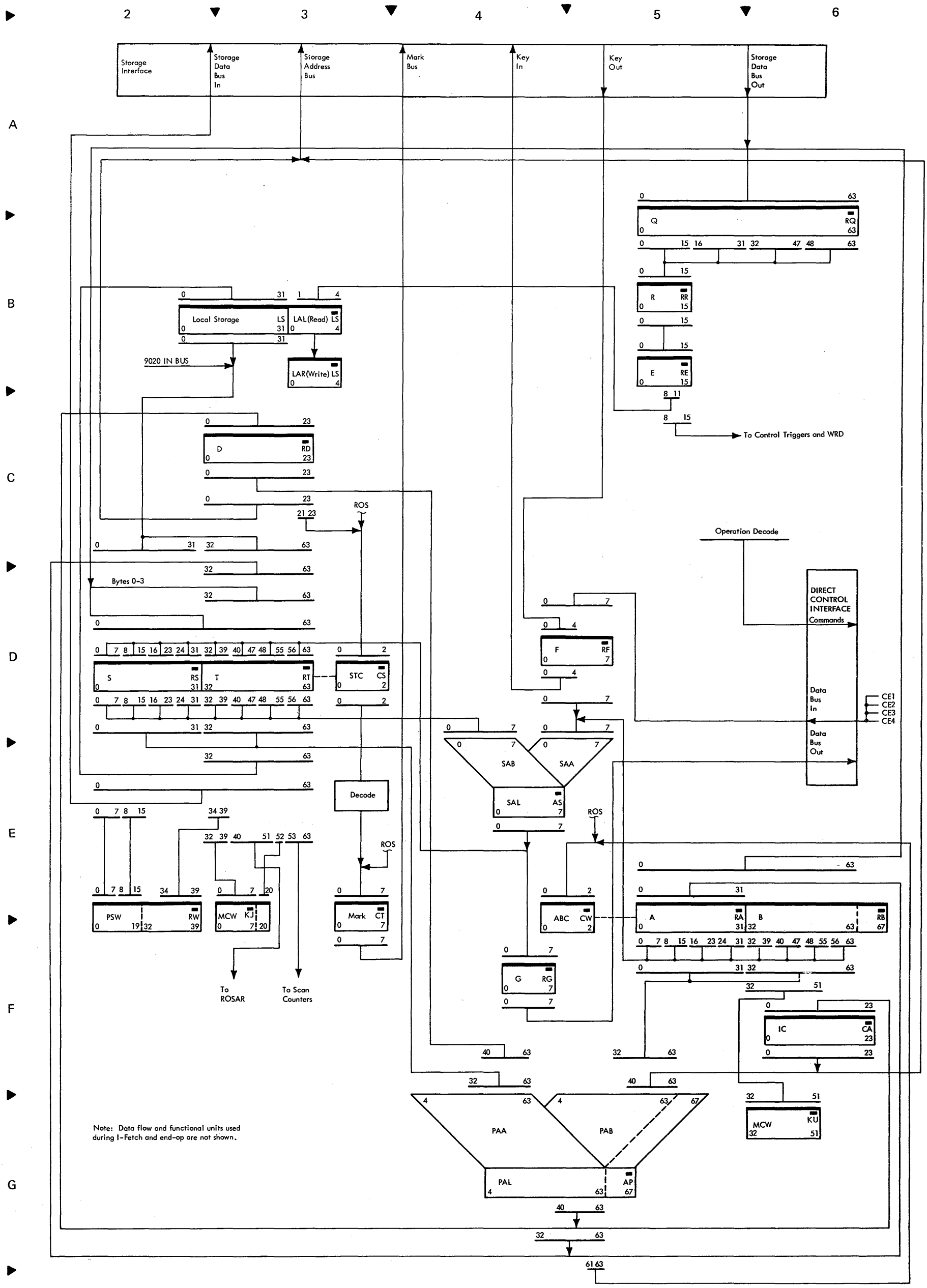


Diagram 3-5. Status Switching Instruction Data Flow

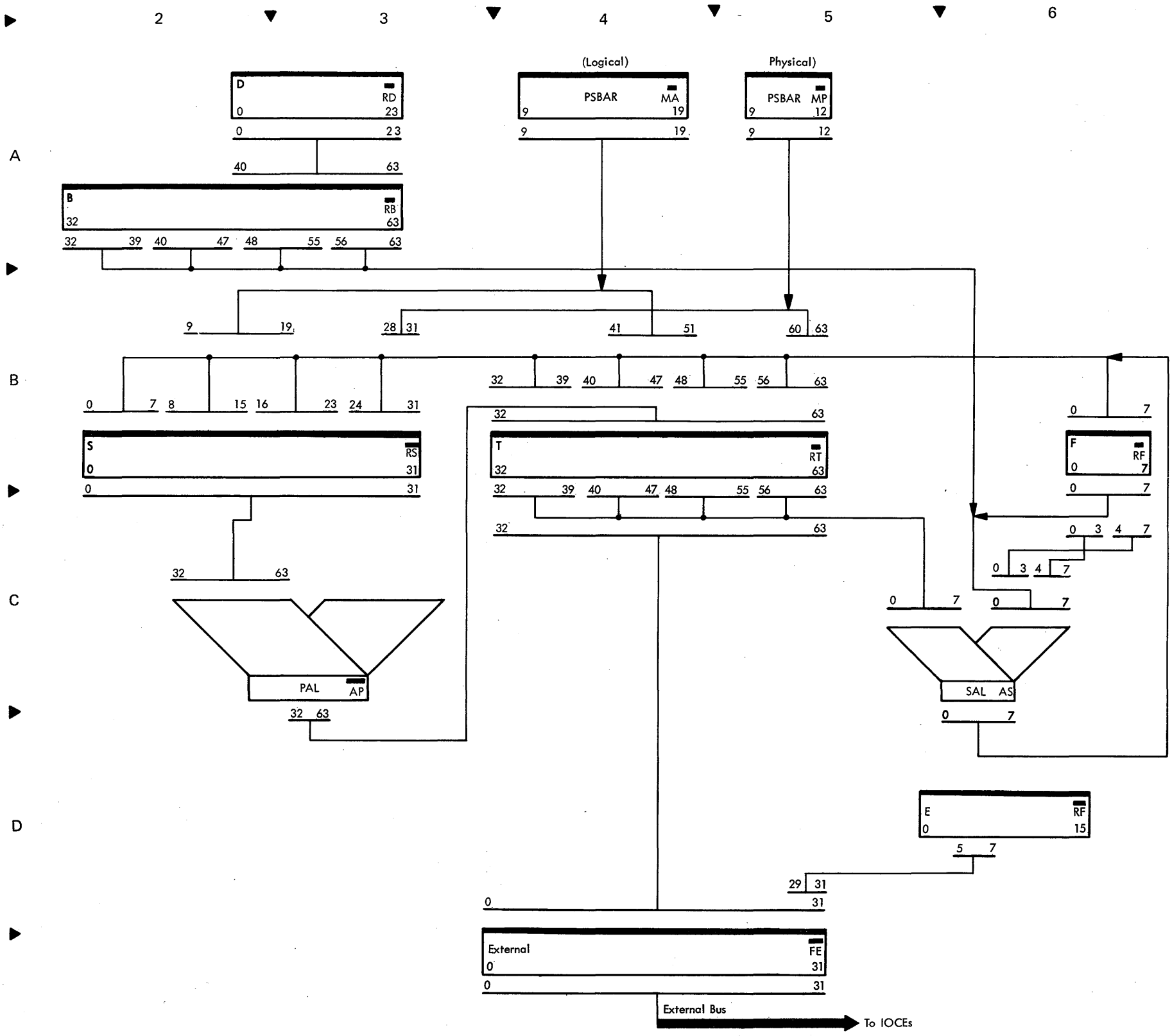


Diagram 3-6. Input/Output Instruction Data Flow

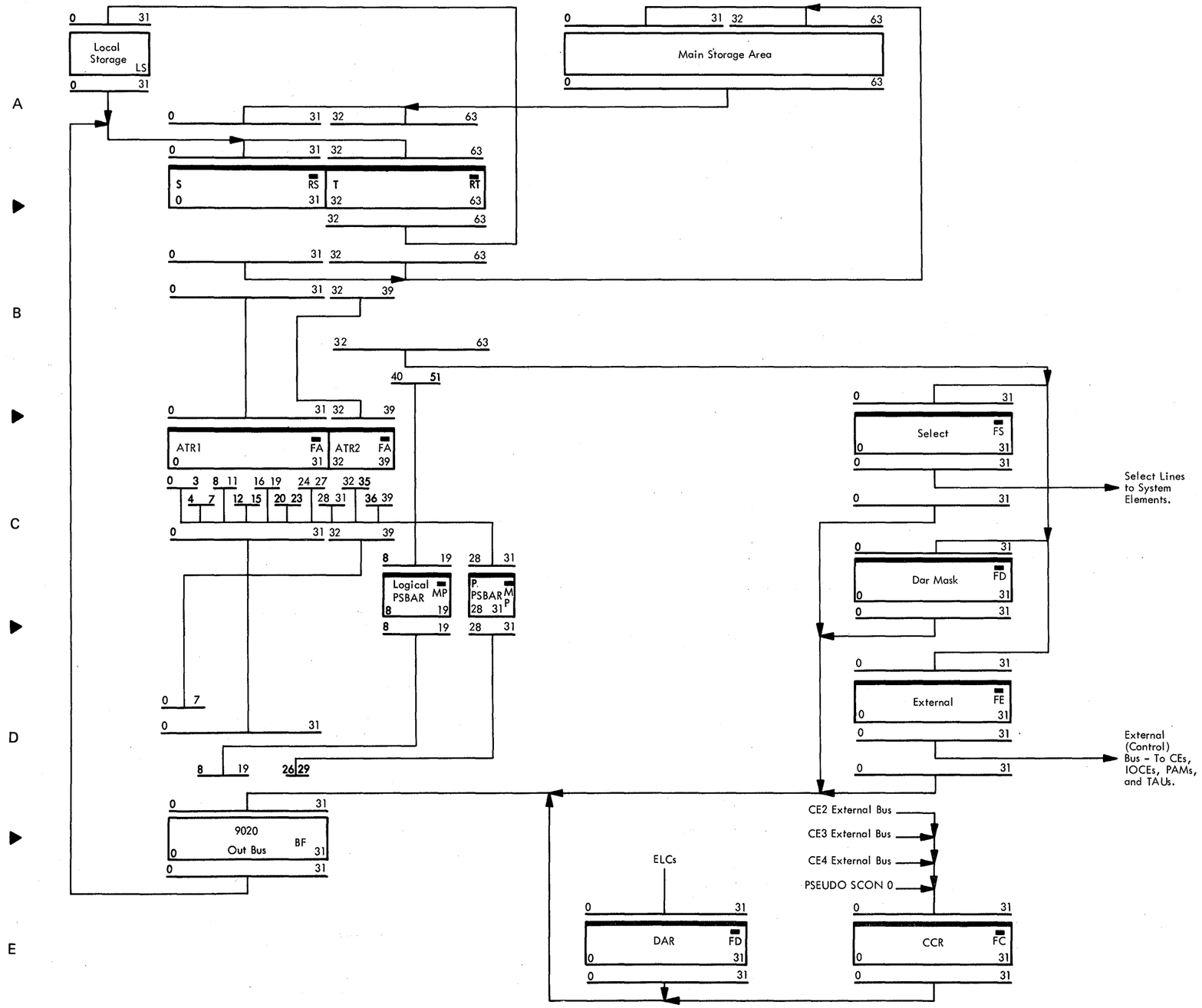


Diagram 3-7. Multiple Computing Element Instruction Data Flow

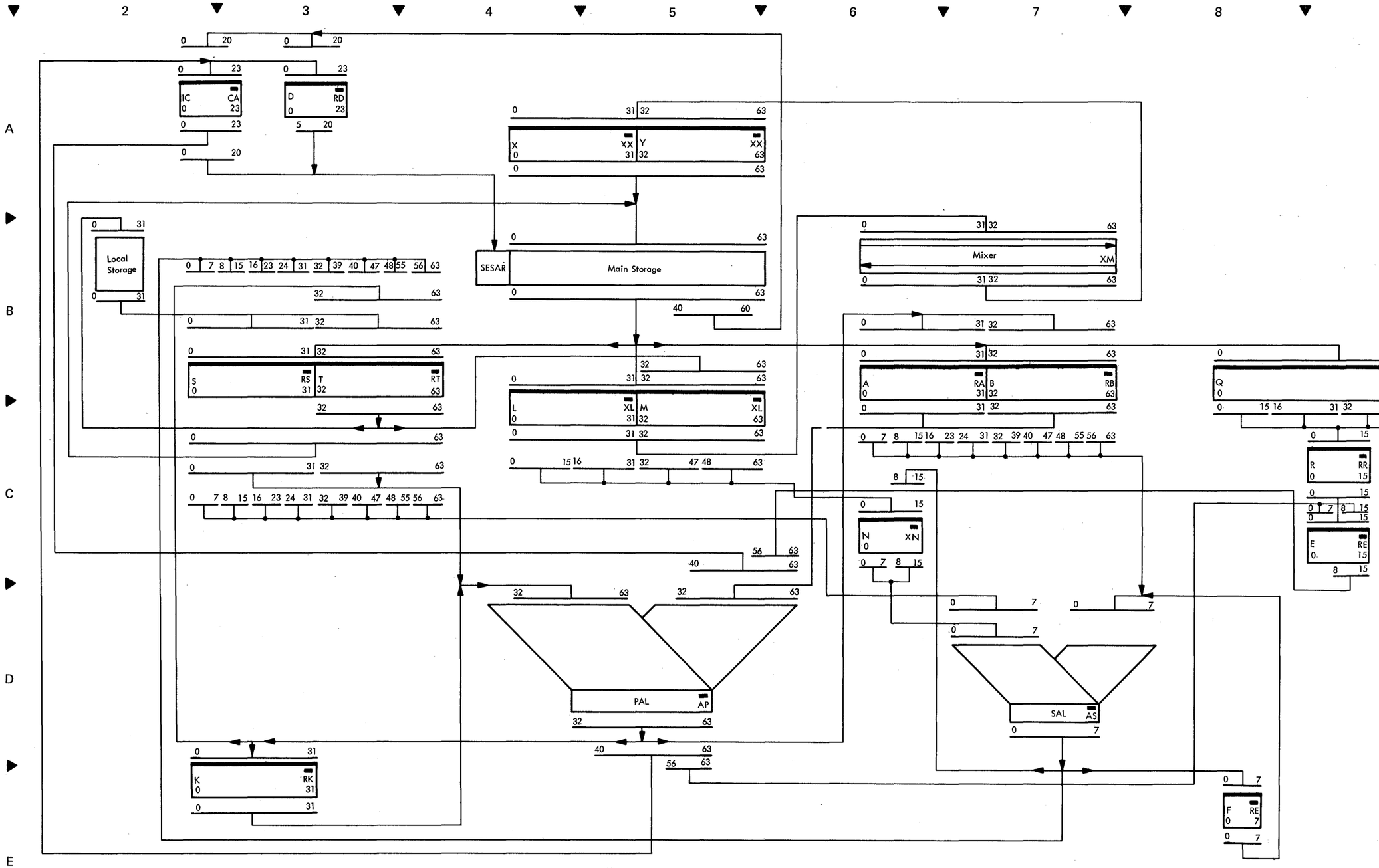
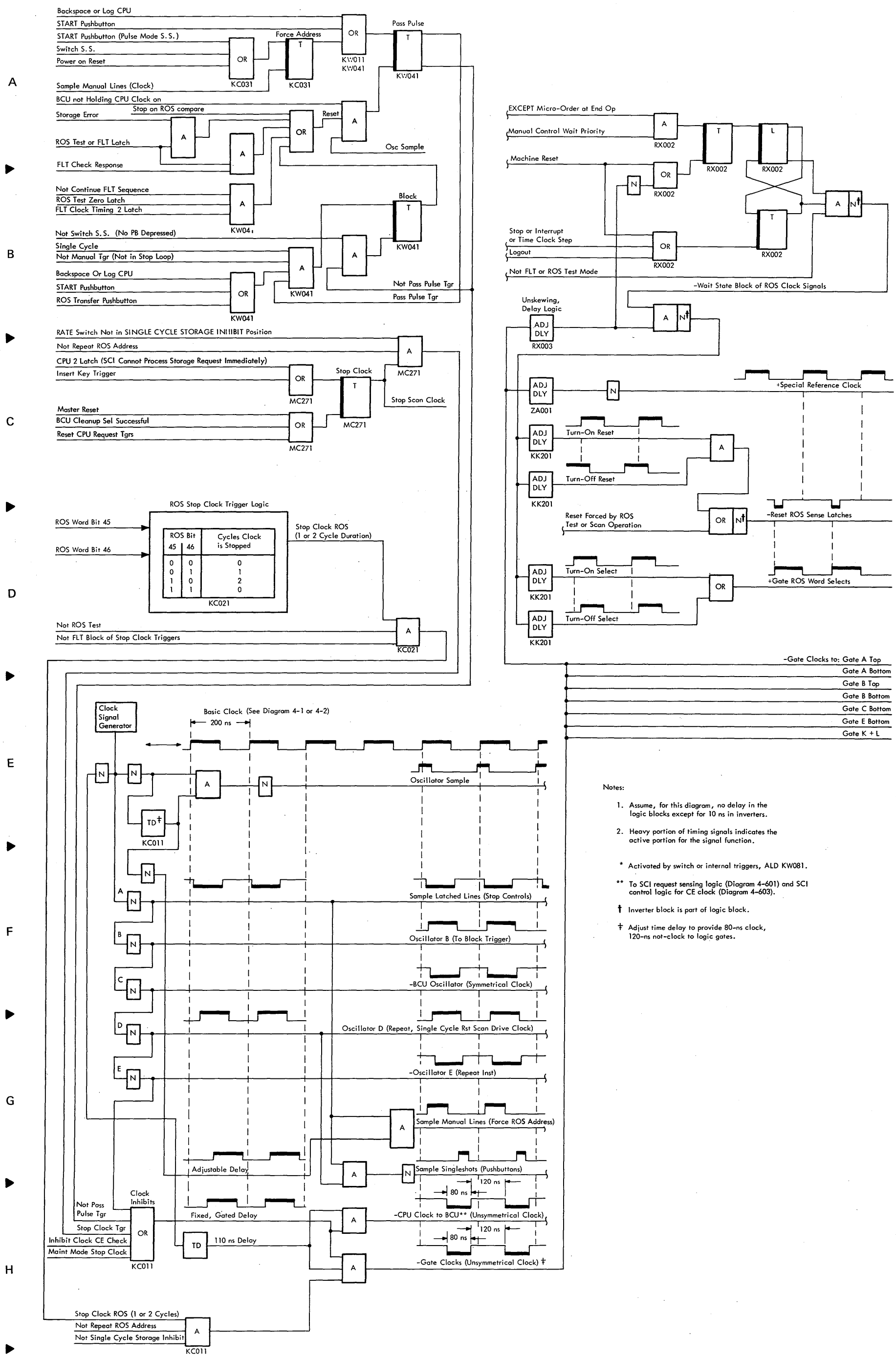


Diagram 3-8. Display Instruction Data Flow



Notes:

1. Assume, for this diagram, no delay in the logic blocks except for 10 ns in inverters.
2. Heavy portion of timing signals indicates the active portion for the signal function.

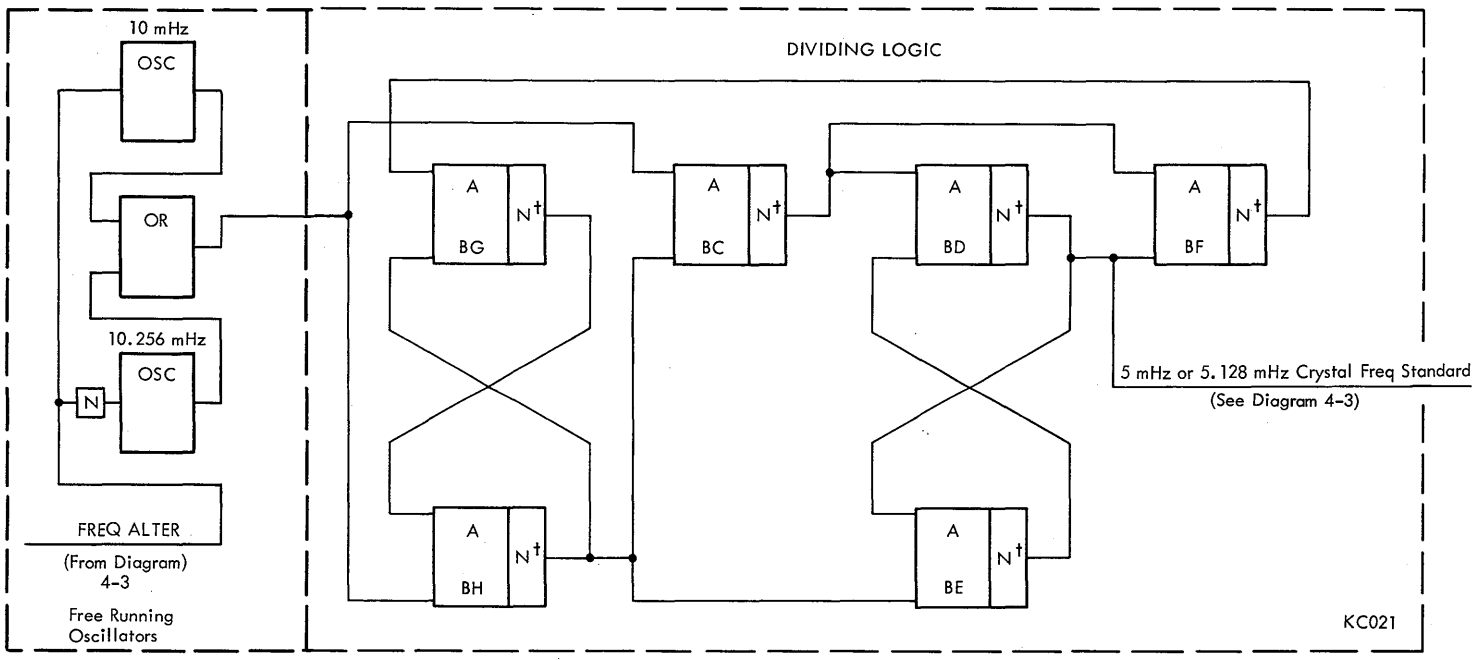
* Activated by switch or internal triggers, ALD KW081.

** To SCI request sensing logic (Diagram 4-601) and SCI control logic for CE clock (Diagram 4-603).

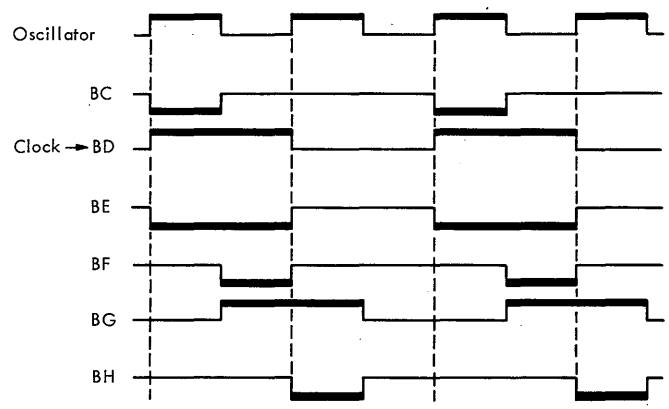
† Inverter block is part of logic block.

‡ Adjust time delay to provide 80-ns clock, 120-ns not-clock to logic gates.

Diagram 4-1. Clock Control Logic



Ideal Waveform At Output Of Each AND After Inversion:



Notes:

1. Heavy portion of timing signals indicates the active portion for the signal function.
2. The two letter notation within the AND's is the block serial number on ALD KC021.

†Inverter block is part of logic block.

Diagram 4-2. Reference Oscillator

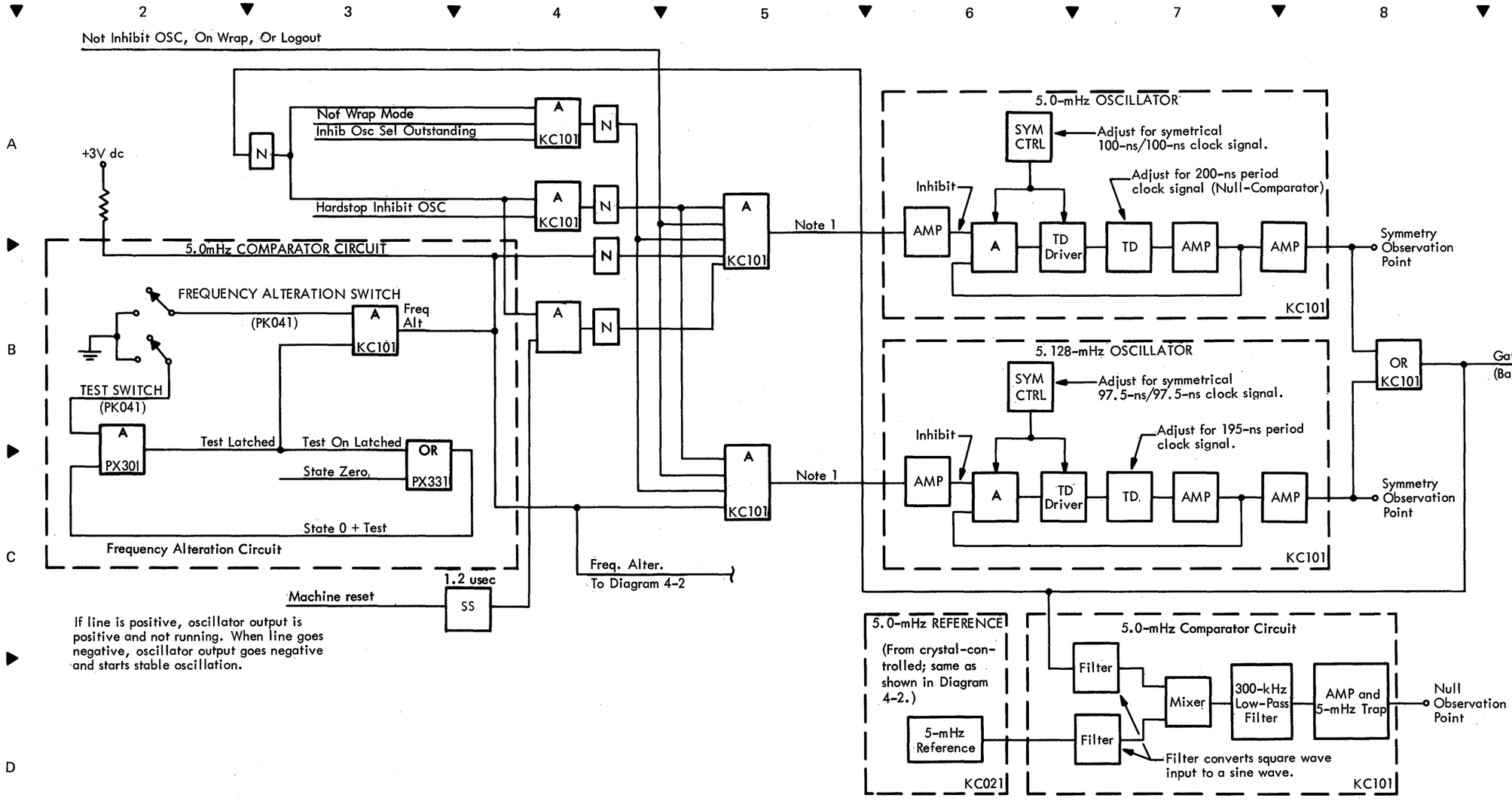


Diagram 4-3. CE Clock Signal Generator

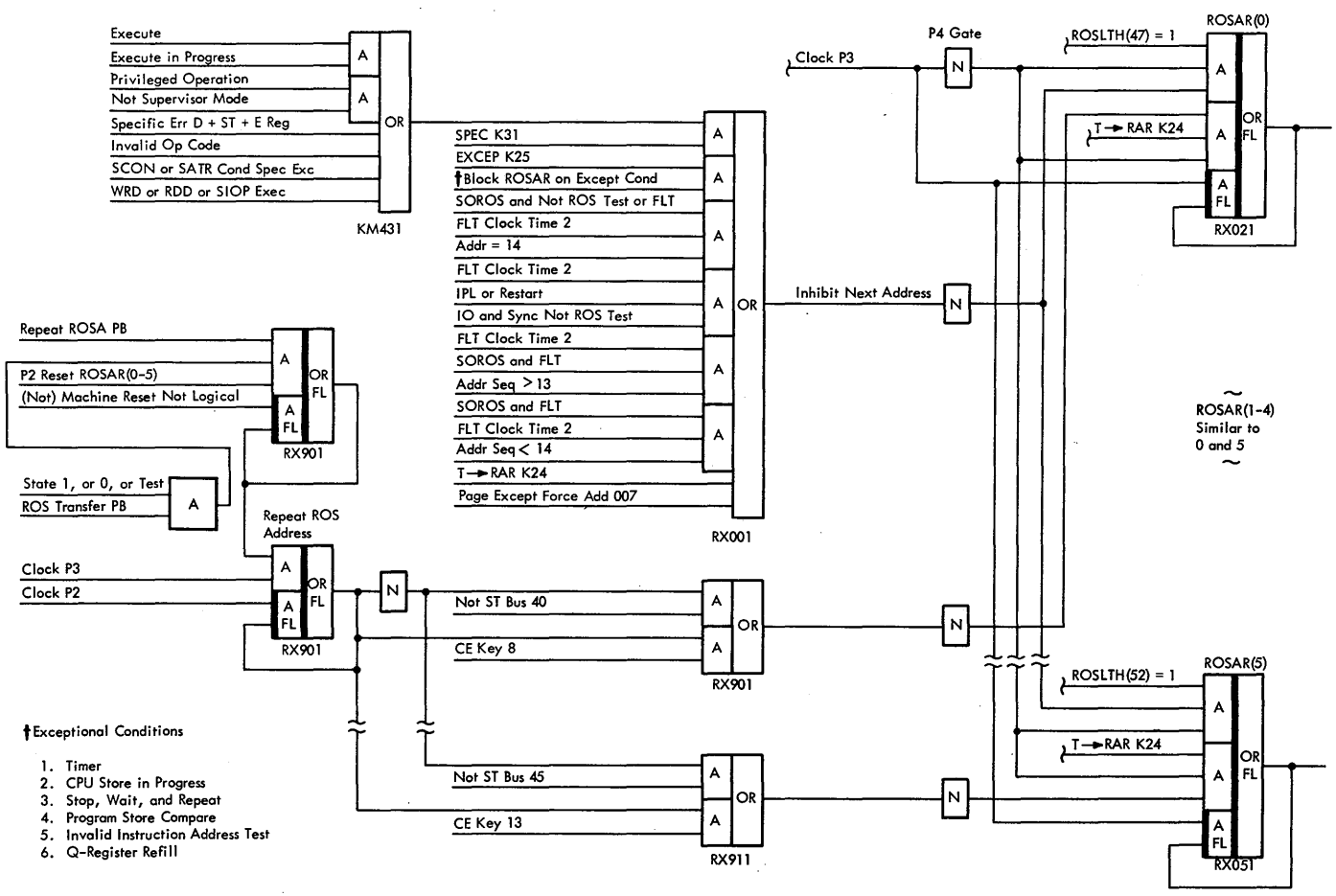


Diagram 4-101. ROSAR (0-5) Logic

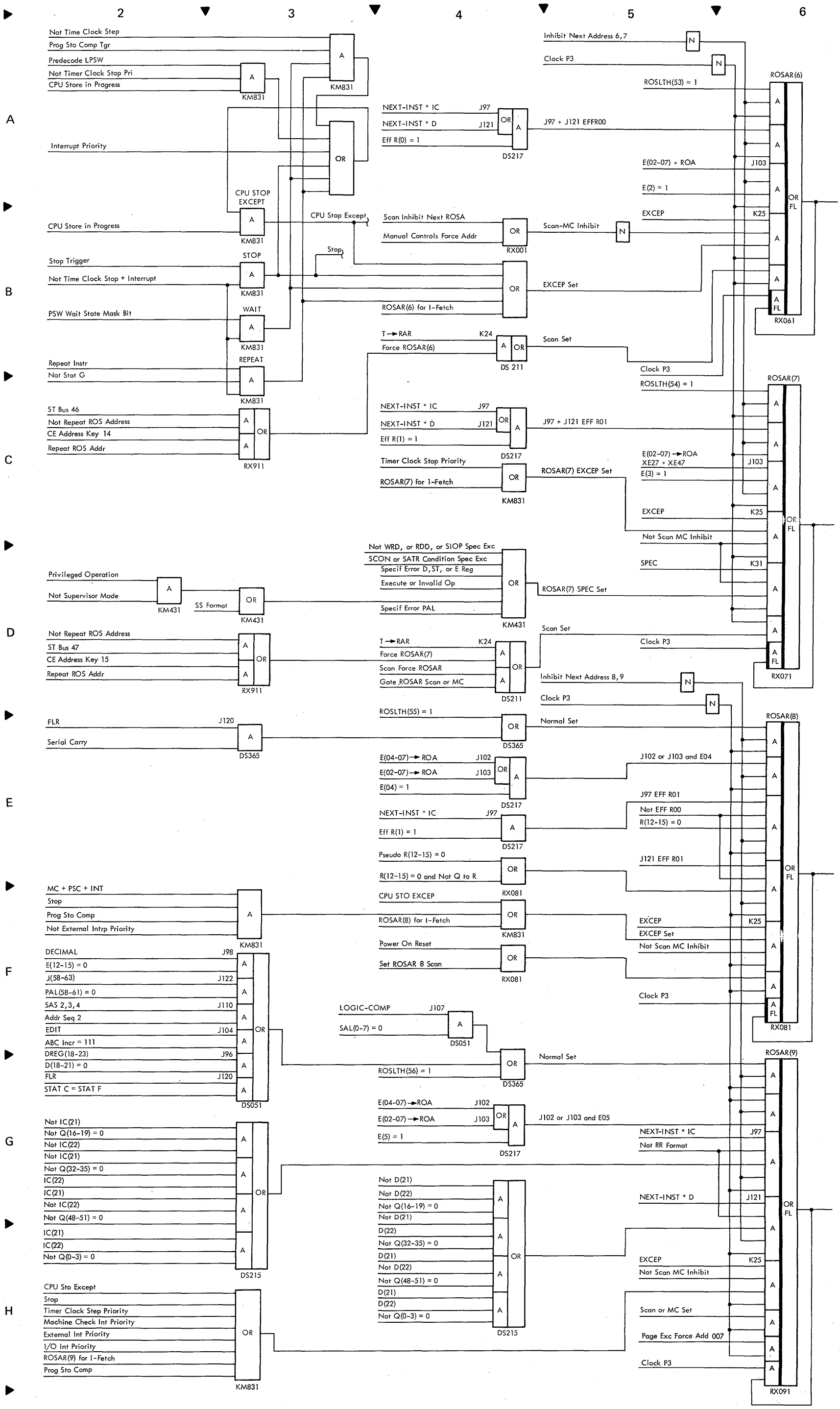


Diagram 4-102. ROSAR (6-9) Logic

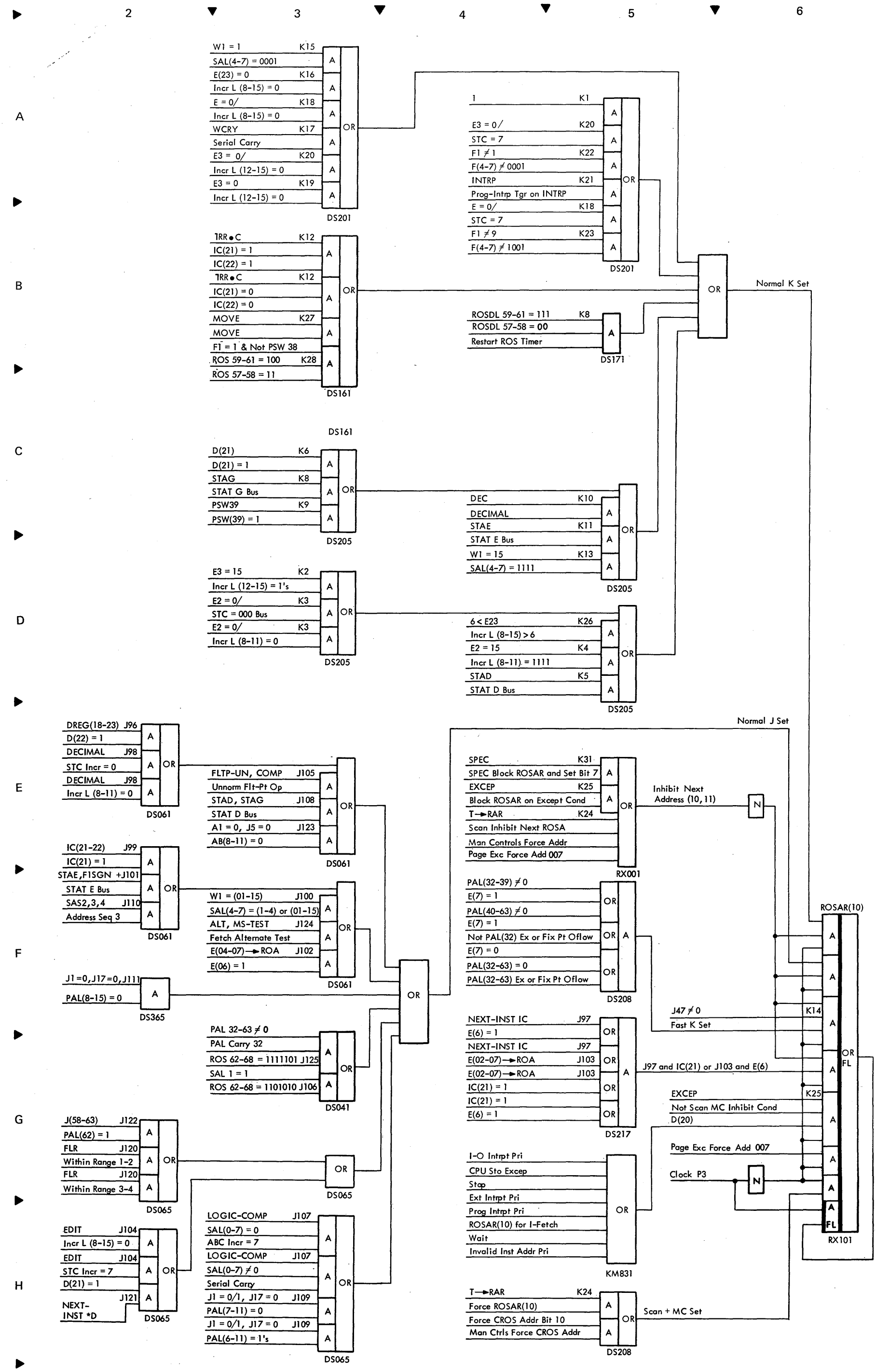


Diagram 4-103. ROSAR (10) Logic

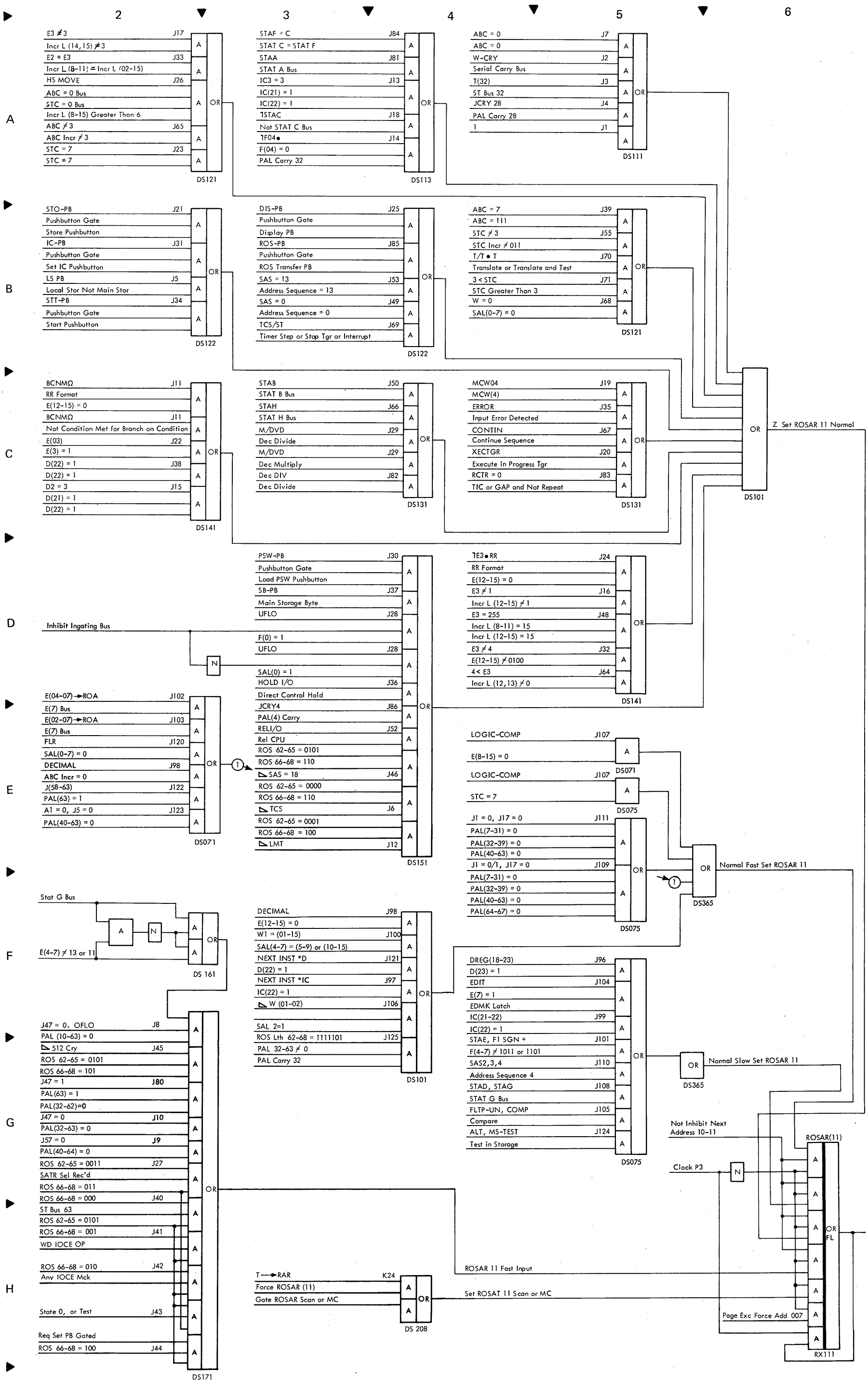


Diagram 4-104. ROSAR (11) Logic

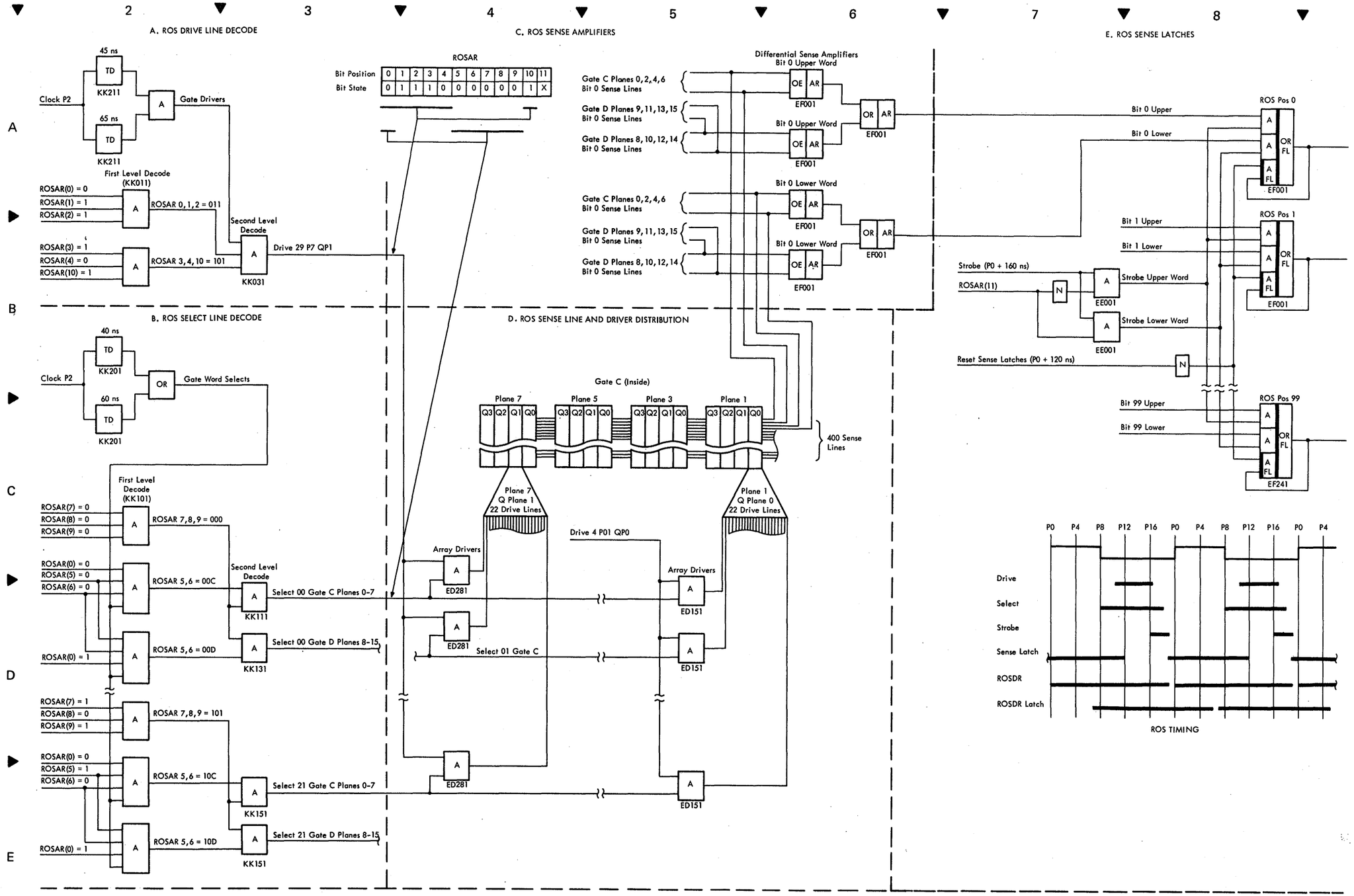


Diagram 4-105. ROS Addressing and Data Flow (Sheet 1 of 2)

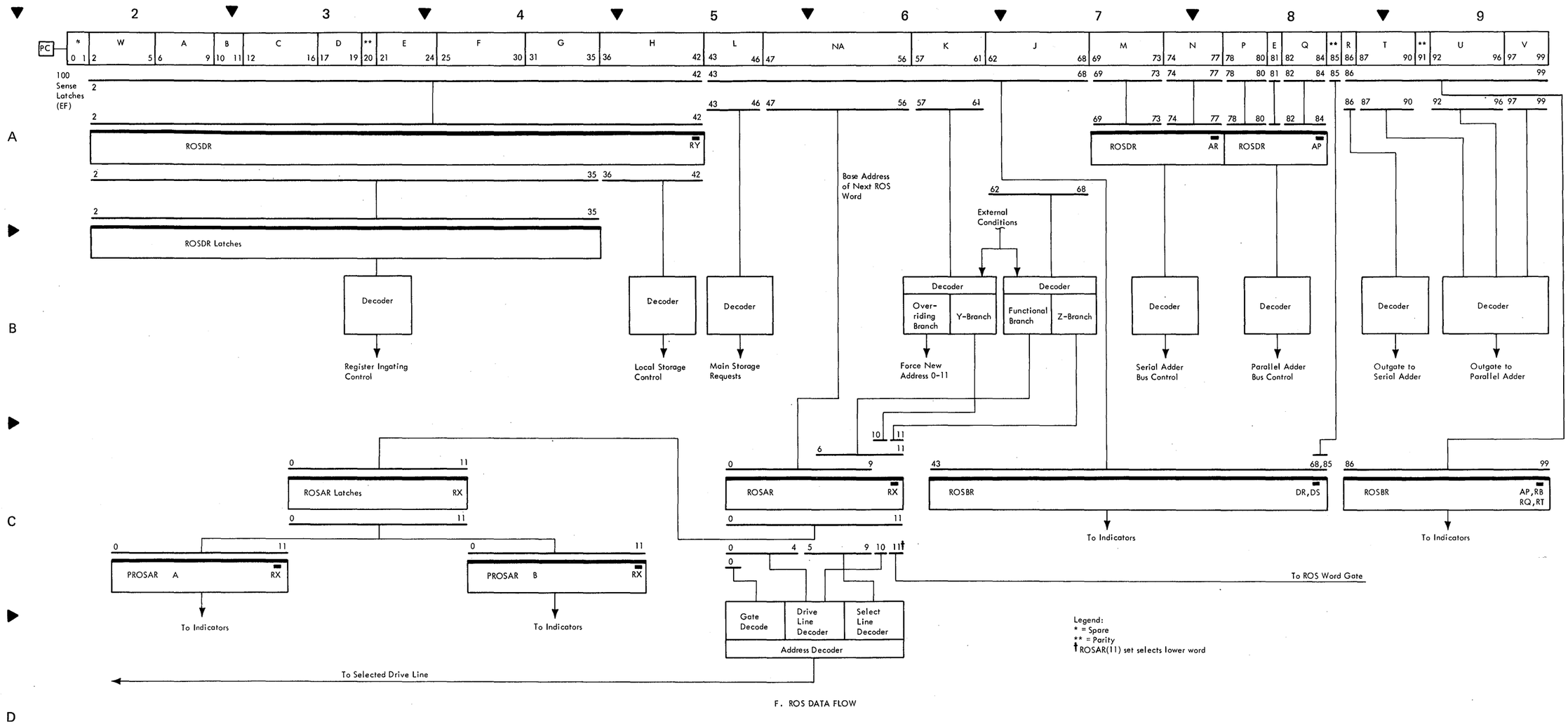


Diagram 4-105. ROS Addressing and Data Flow (Sheet 2 of 2)

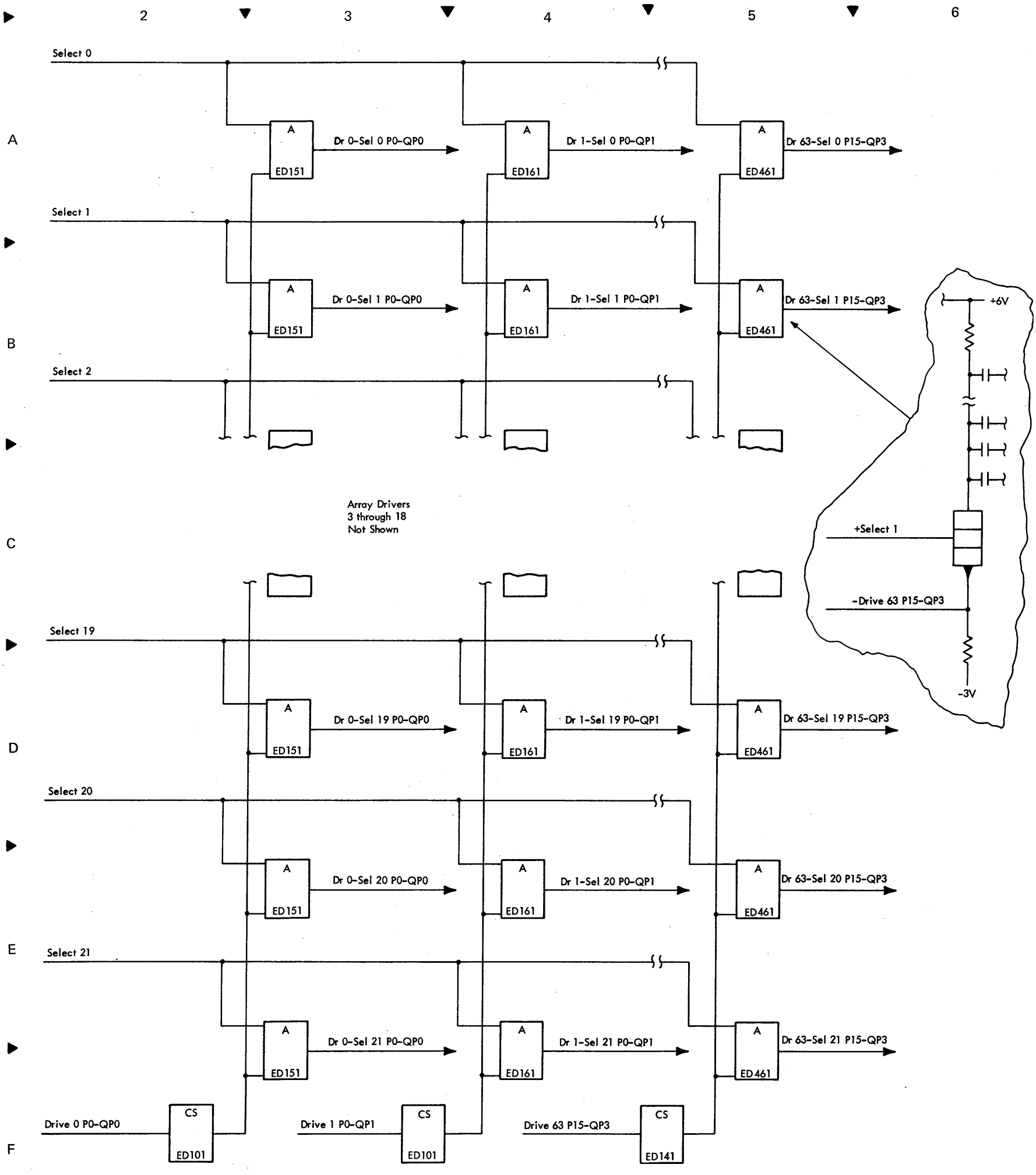
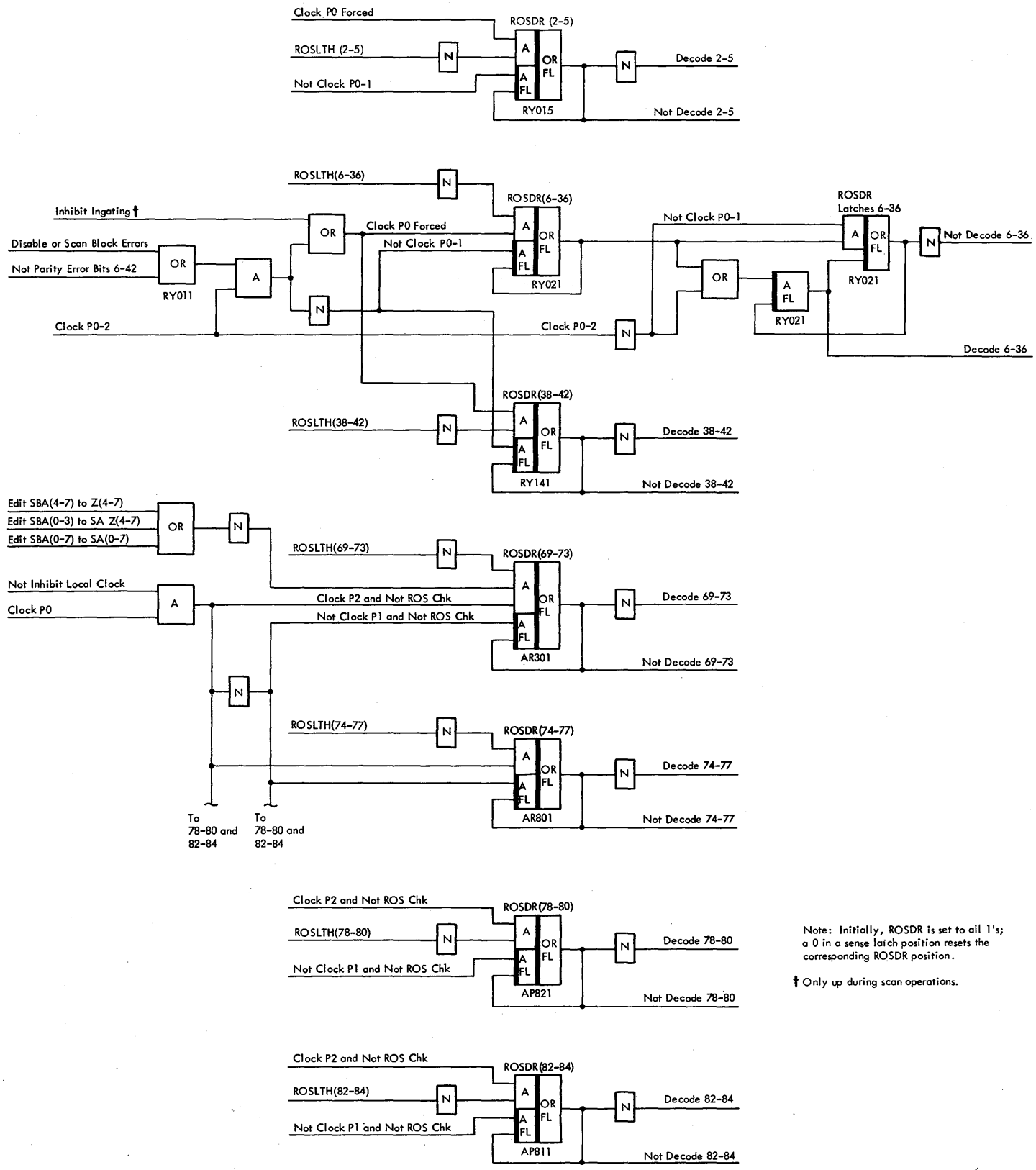


Diagram 4-106. Array Drivers



Note: Initially, ROSDR is set to all 1's; a 0 in a sense latch position resets the corresponding ROSDR position.

† Only up during scan operations.

Diagram 4-107. ROS Data Register

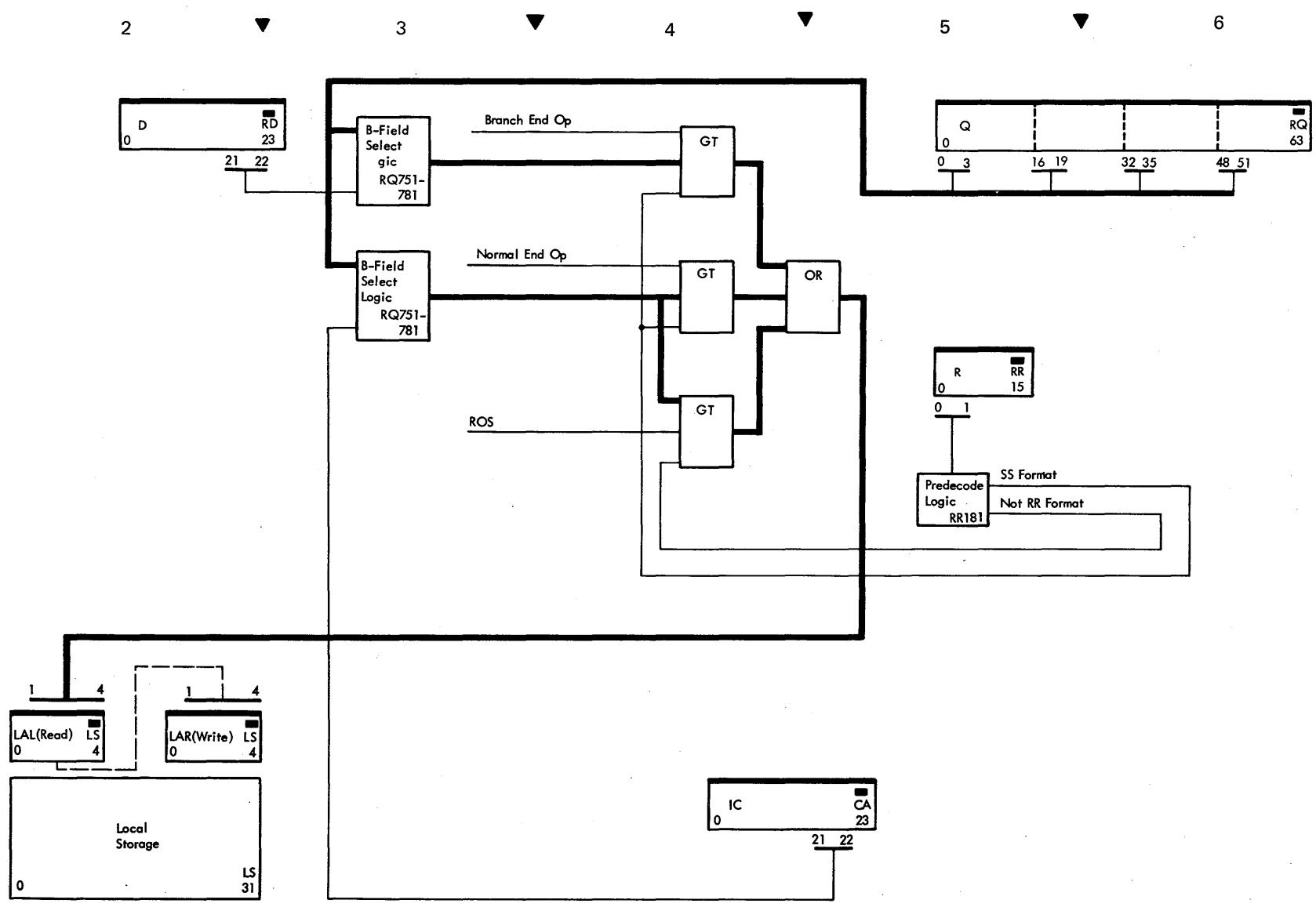


Diagram 4-201. Q-Register B-Field Transfer Controls

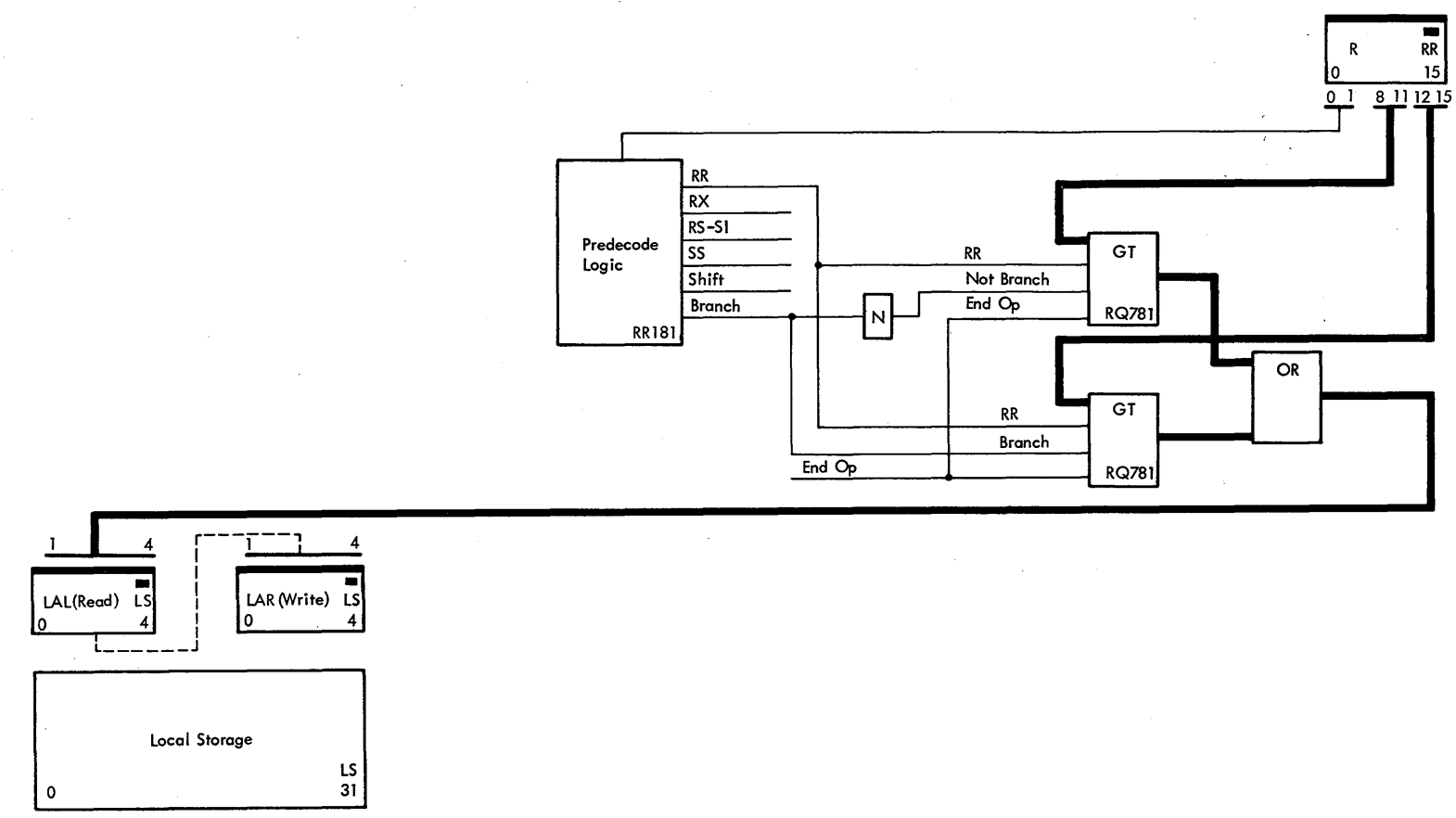


Diagram 4-202. R-Register Transfer to LAL

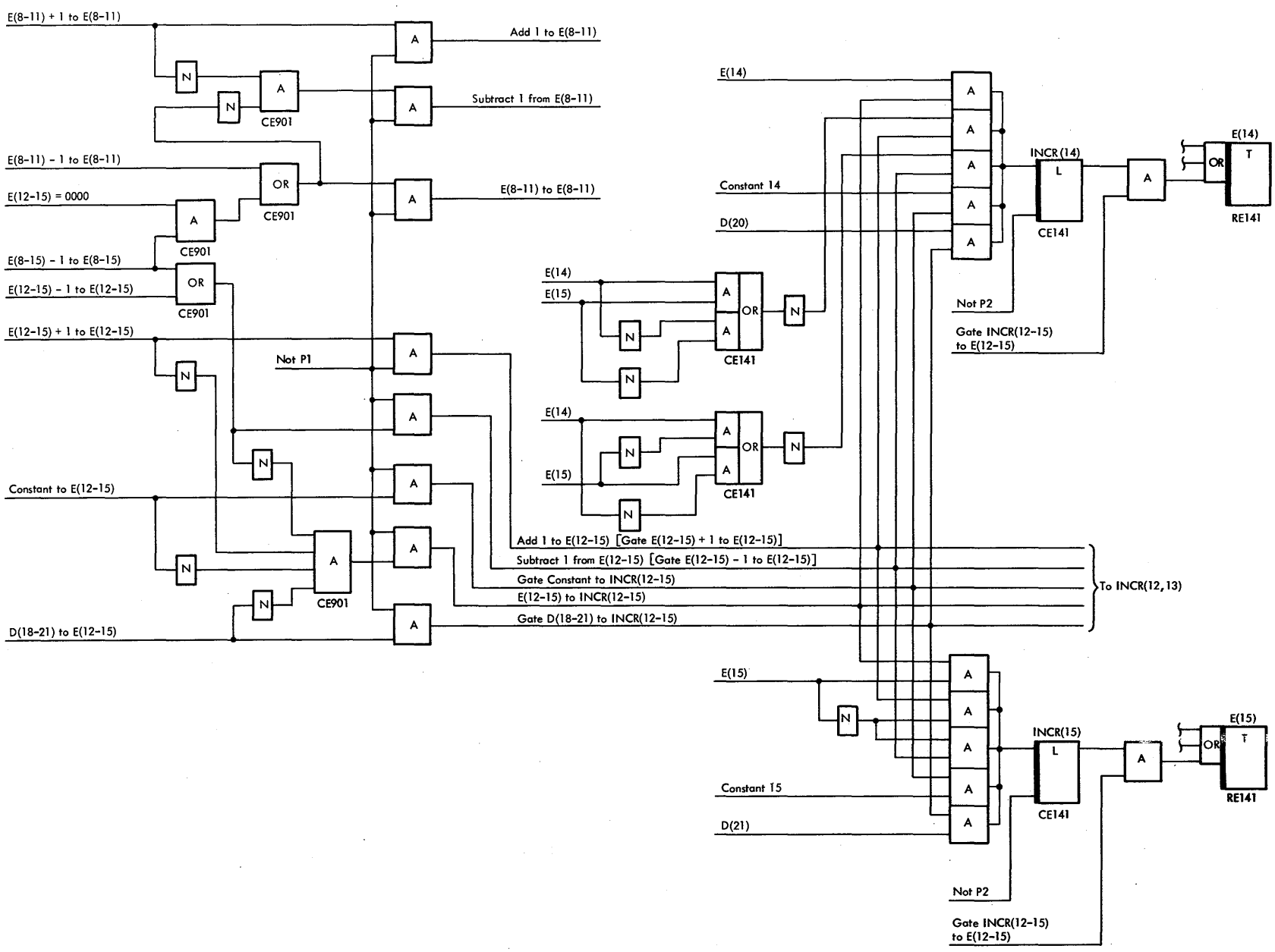


Diagram 4-203. E-Register Incrementer, Bits 14 and 15

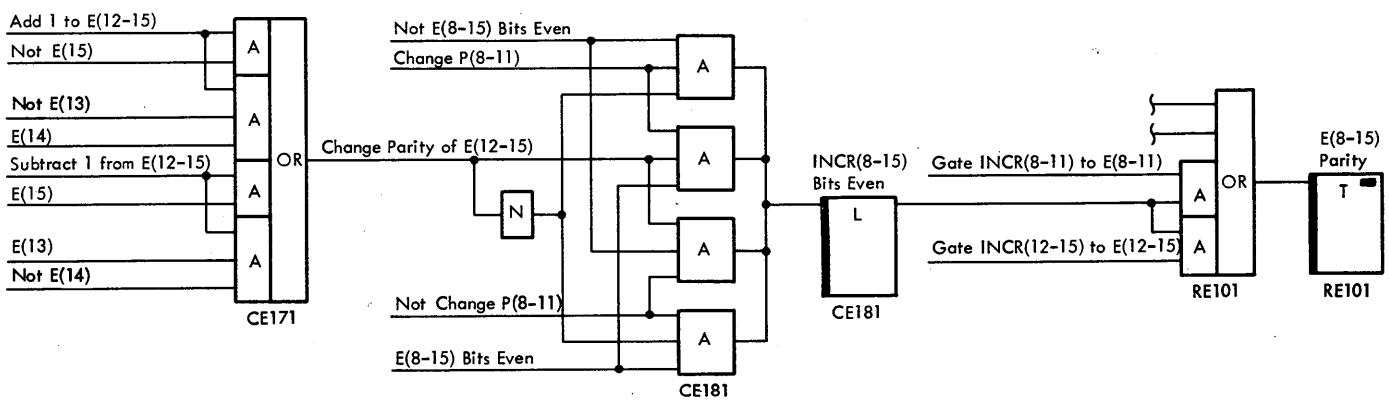


Diagram 4-204. E-Register Parity Prediction after Incrementing

B

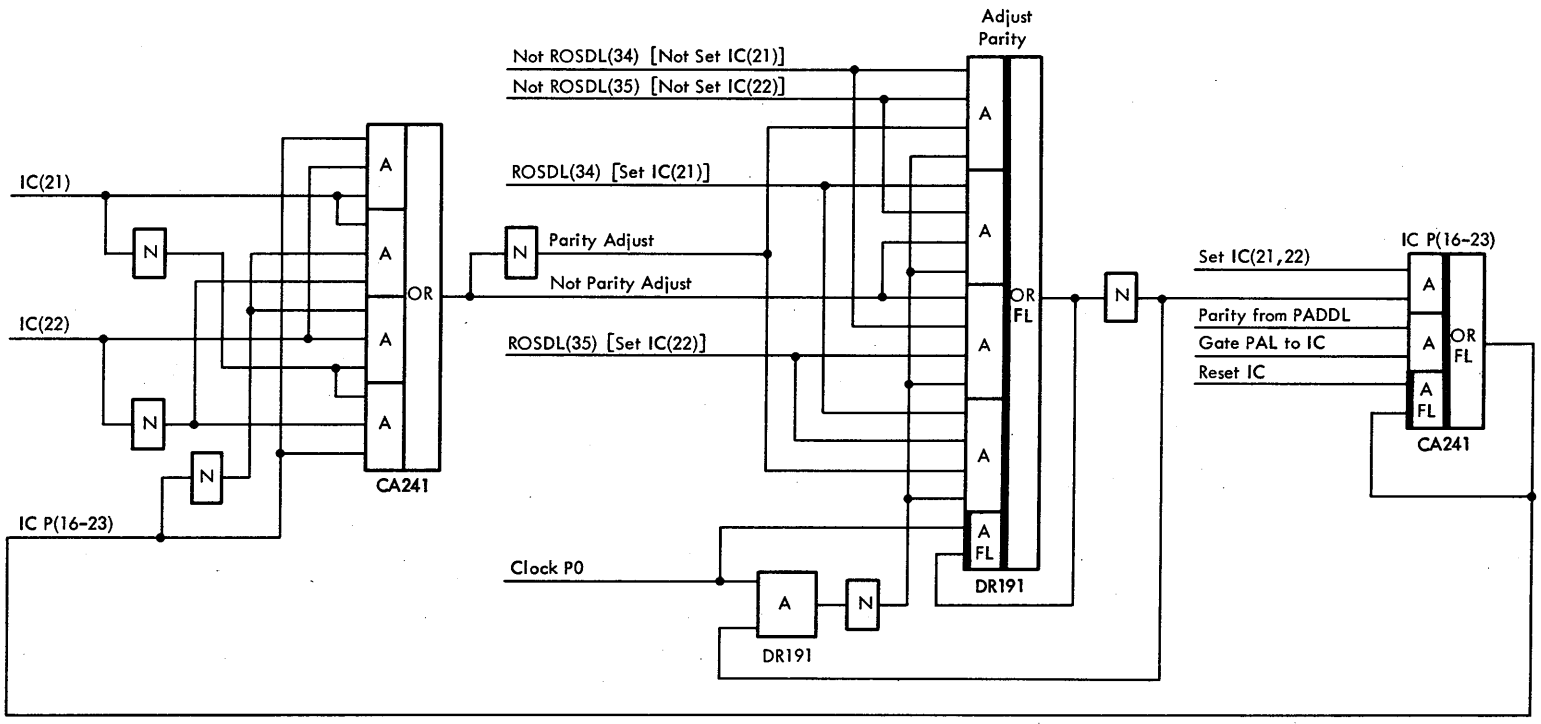
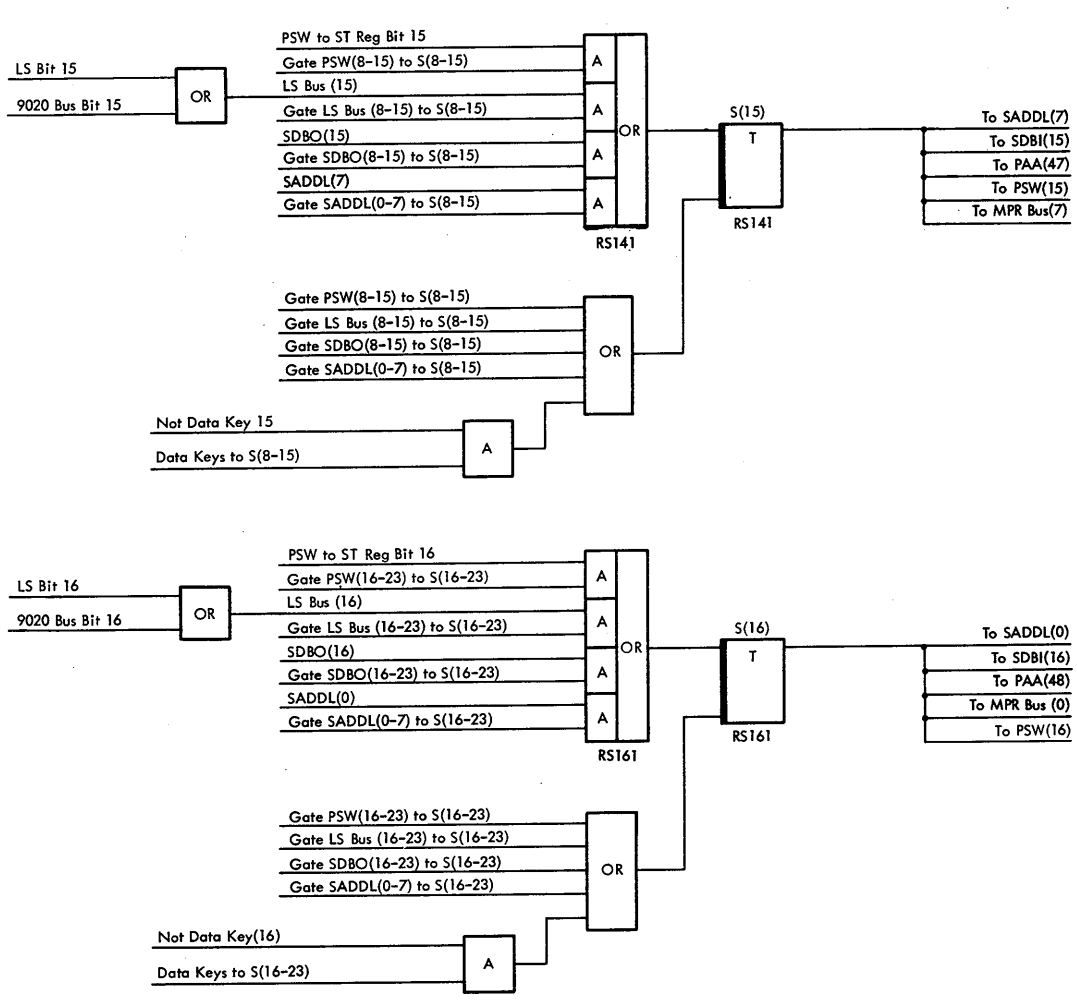


Diagram 4-205. Parity Adjustment for IC (21, 22) Stepping

E

F



G

H

Diagram 4-206. S-Register, Bits 15 and 16

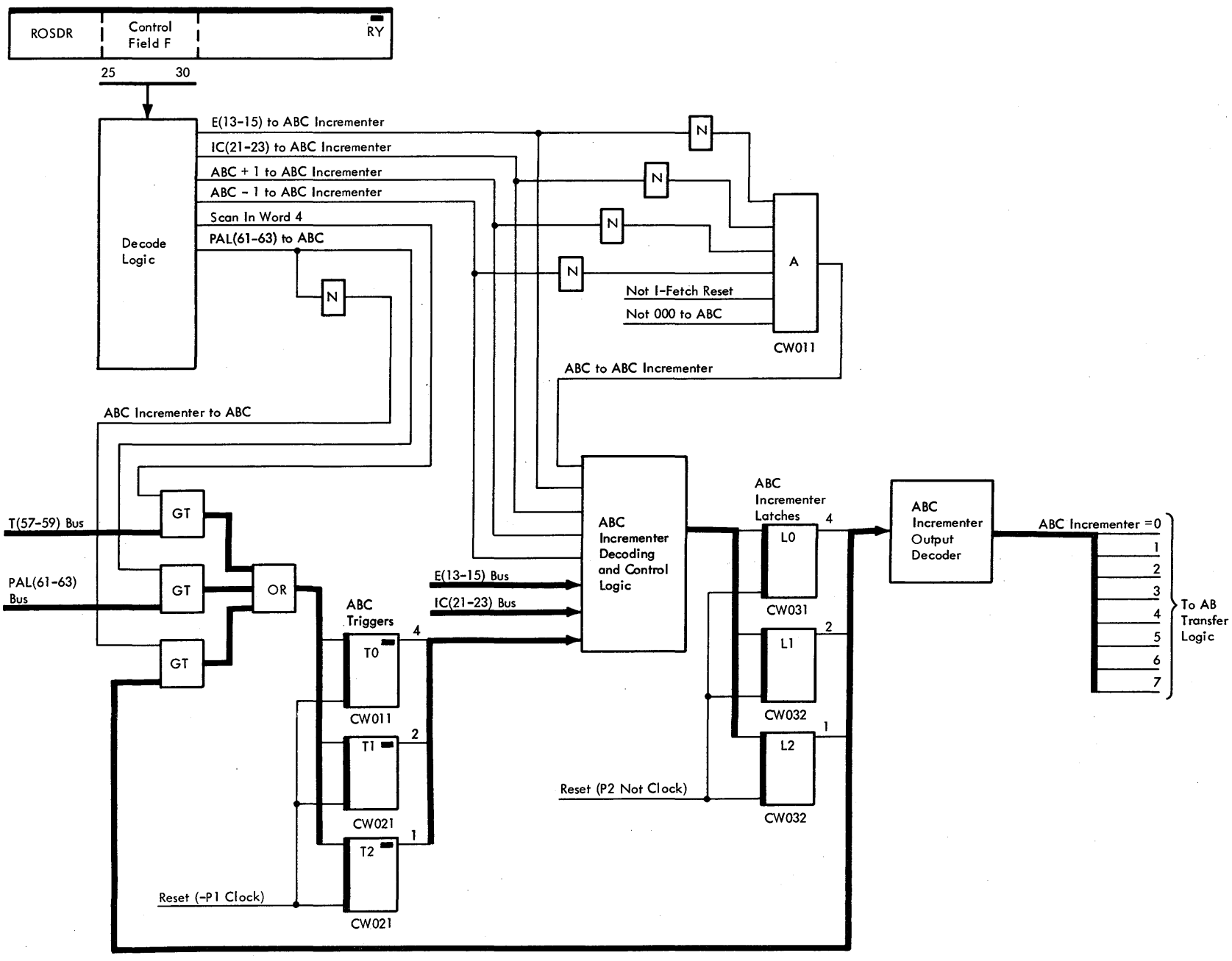


Diagram 4-207. AB Byte Counter

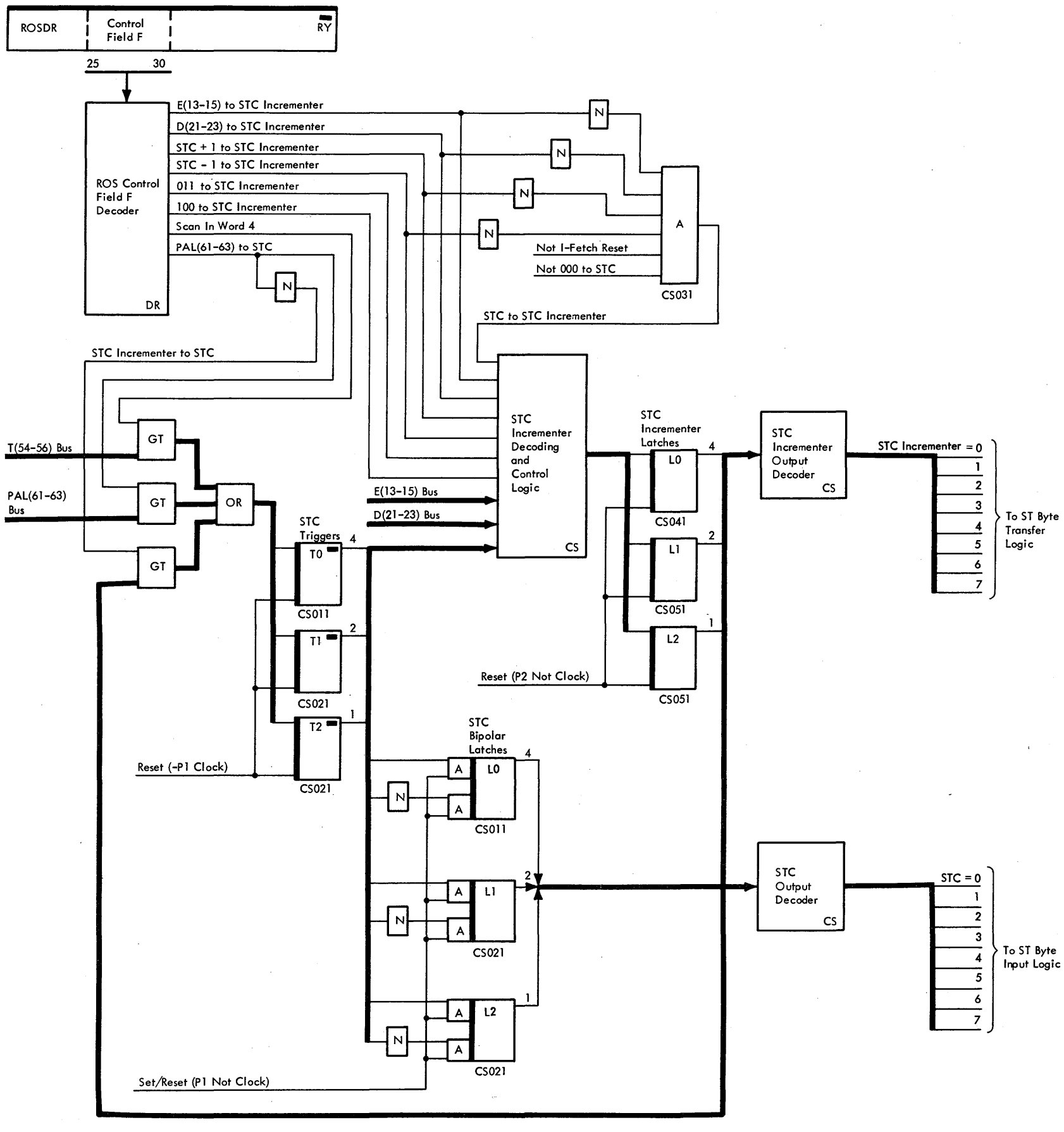


Diagram 4-208. ST Byte Counter

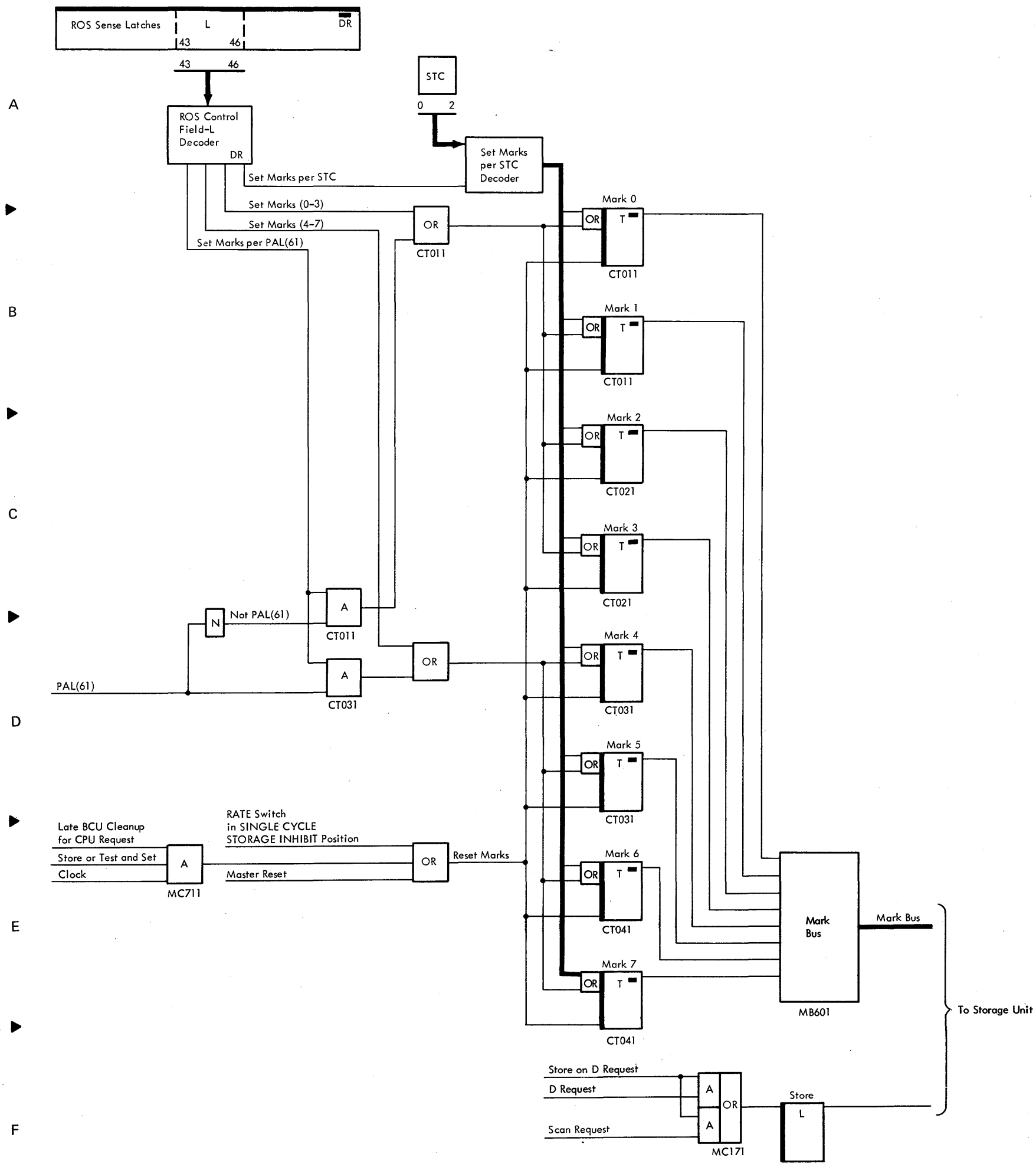


Diagram 4-209. Mark Trigger Logic

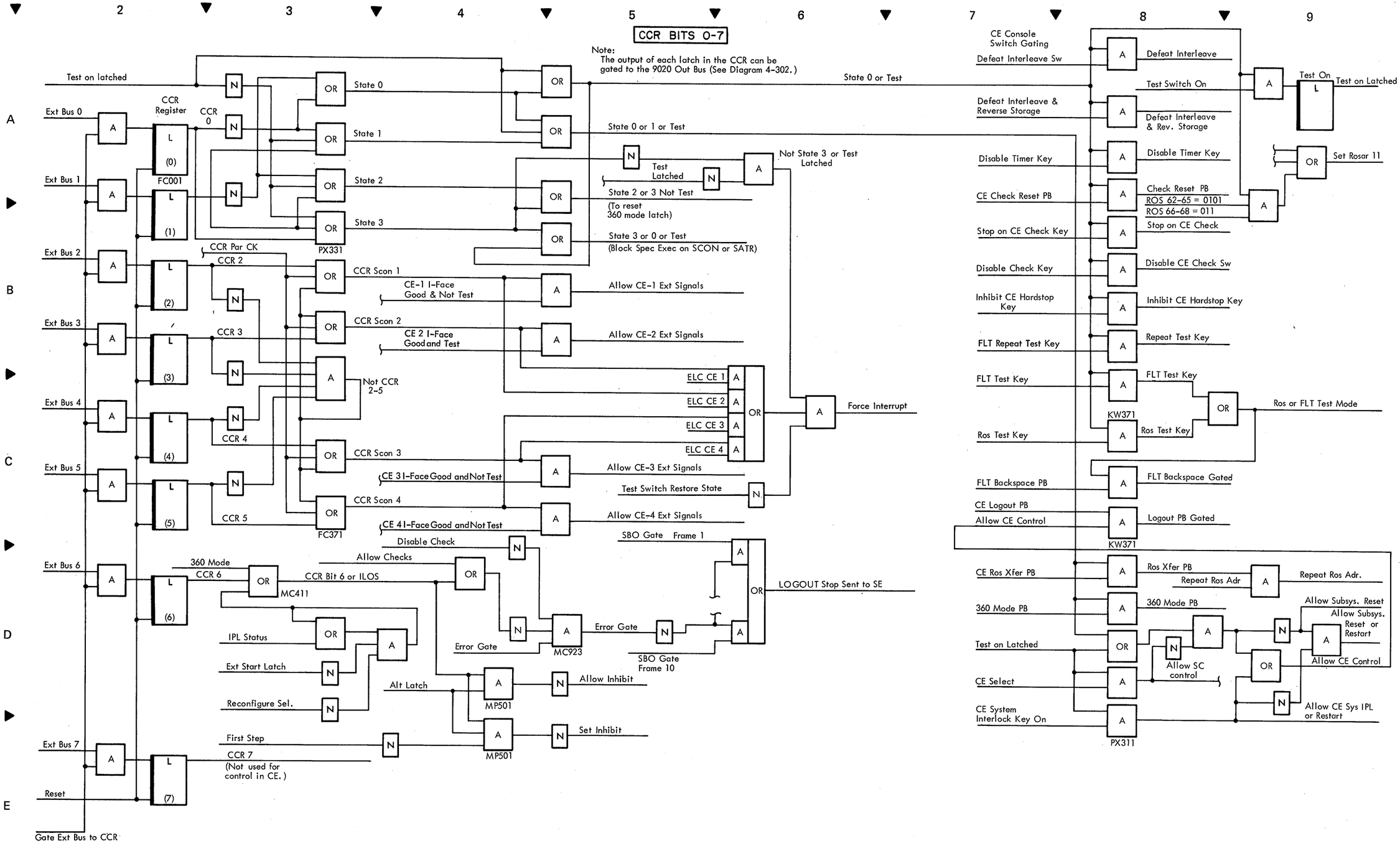


Diagram 4-210. CCR Output Logic and Control Paths (Sheet 1 of 3)

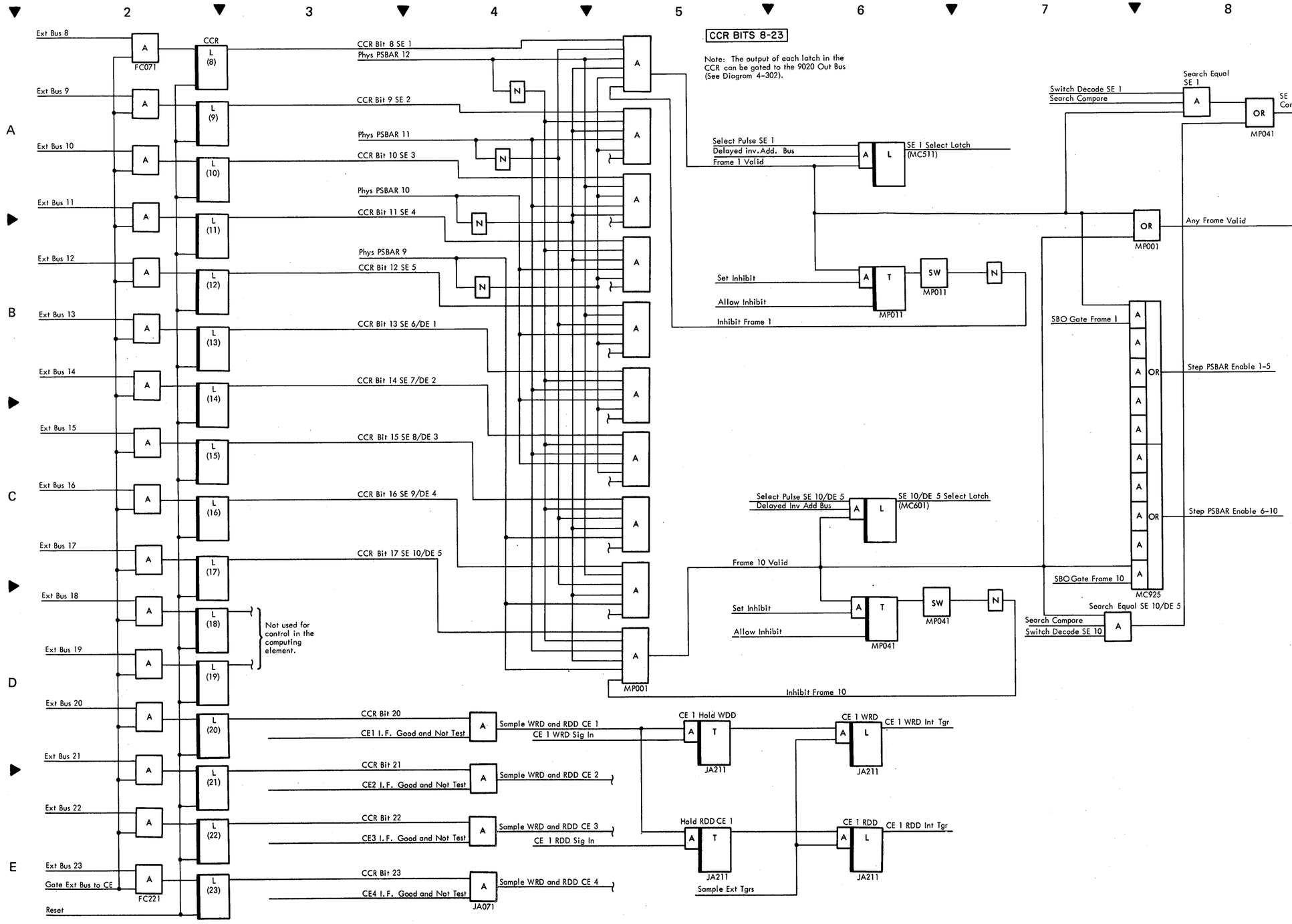


Diagram 4-210. CCR Output Logic and Control Paths (Sheet 2 of 3)

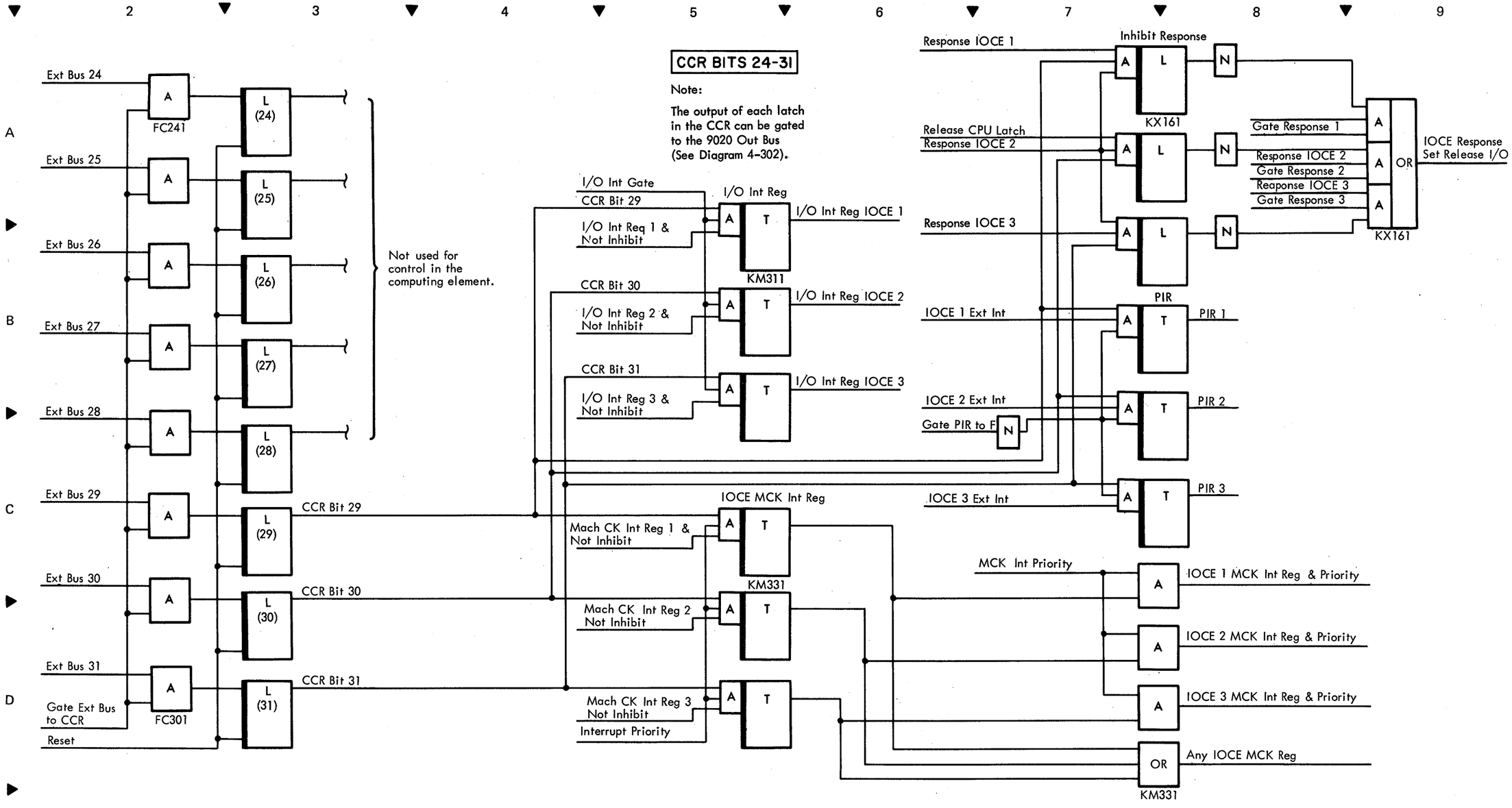
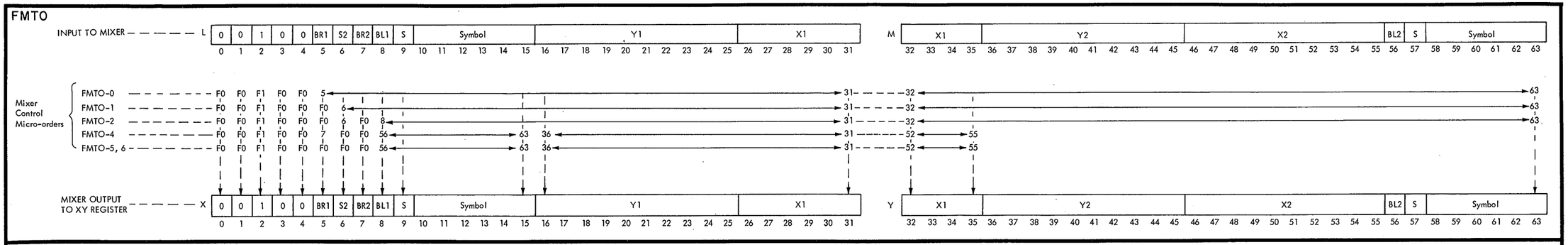


Diagram 4-210. CCR Output Logic and Control Paths (Sheet 3 of 3)

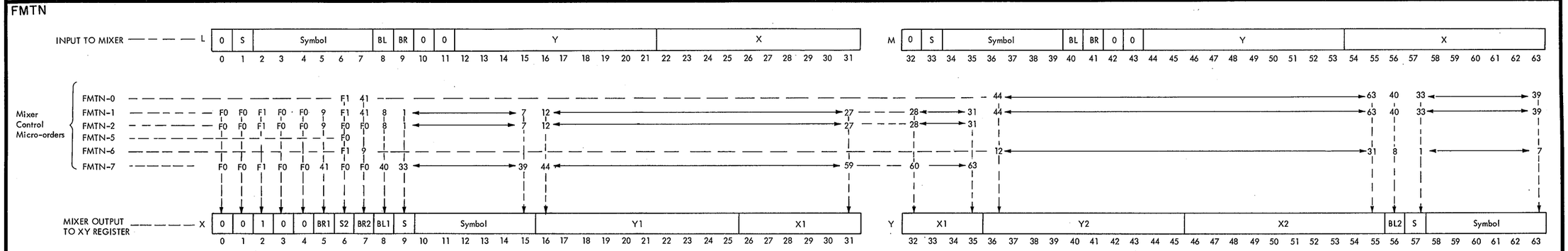
2 3 4 5 6 7 8 9

RPSB Instruction

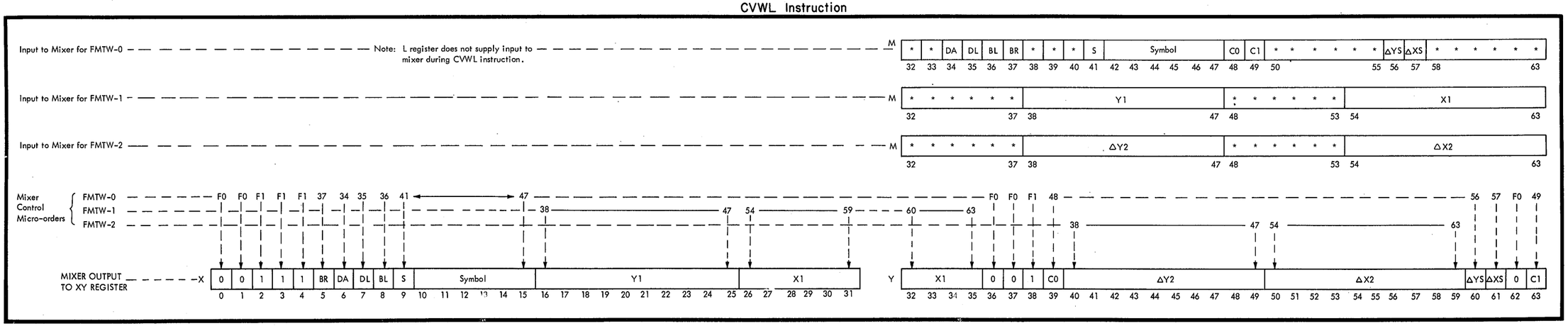
FMTO



FMTN



CVWL Instruction



Notes:

1. To trace the origin of a mixer output bit: (1) select chart for specific instruction, RPSB or CVWL; (2) if RPSB is selected, refer to FMTO or FMTN portion of chart; (3) find bit to be traced in XY register and; (4) follow the line up to the micro-order being issued. For example, assume an RPSB instruction and an FMTN micro-order and assume bit 8 is to be traced. Refer to the RPSB instruction chart, FMTN portion, XY register. Find bit 8 in the XY register and follow the line up to the specific micro-order. If the specific micro-order is an FMTN-7, bit 8 originates from bit 40 of the LM-register. Note also that FMTN-0, 5, and 6 do not set bit 8 and that FMTN-1 and 2 set bit 8 from the input bit 8 in the LM register.
2. Parity for all XY bytes is predicted from LM data parity and LM parity bits.
3. XY register is reset on receipt of either FMTO; FMTN-1, -2, -7; or FMTW-0.

Legend:

1. F0 = Bit forced to 0
2. F1 = Bit forced to 1
3. BR = Brightness bit
   BL = Blink bit
   S = Symbol size
   DA = Dash
   DL = Dash length
   ΔYS = Sign of ΔY2
   ΔXS = Sign of ΔX2
   C0 = Character at major position
   C1 = Character at secondary position
   * = Not used

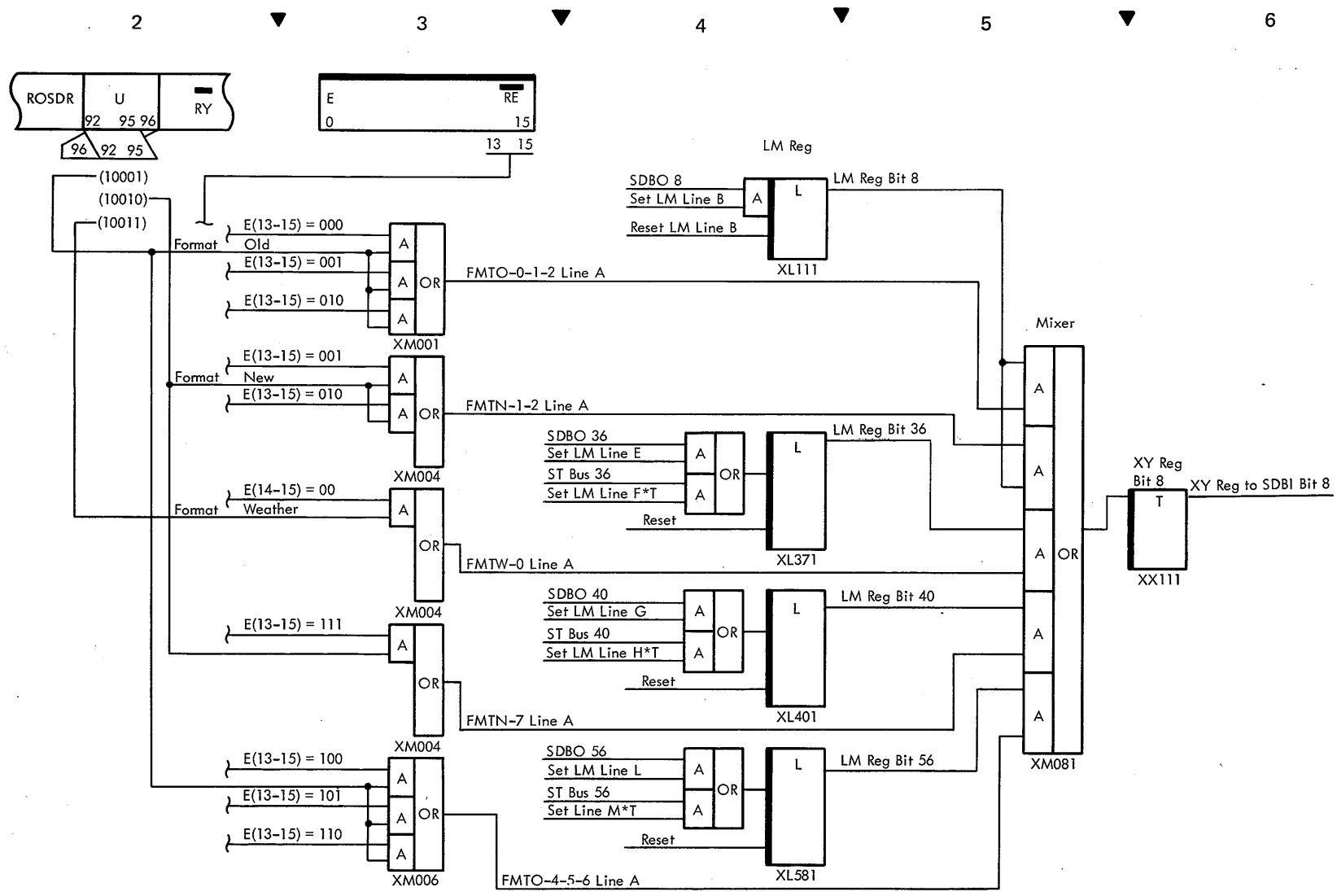


Diagram 4-211. LM to XY Reformatting via Mixer (Sheet 2 of 2)

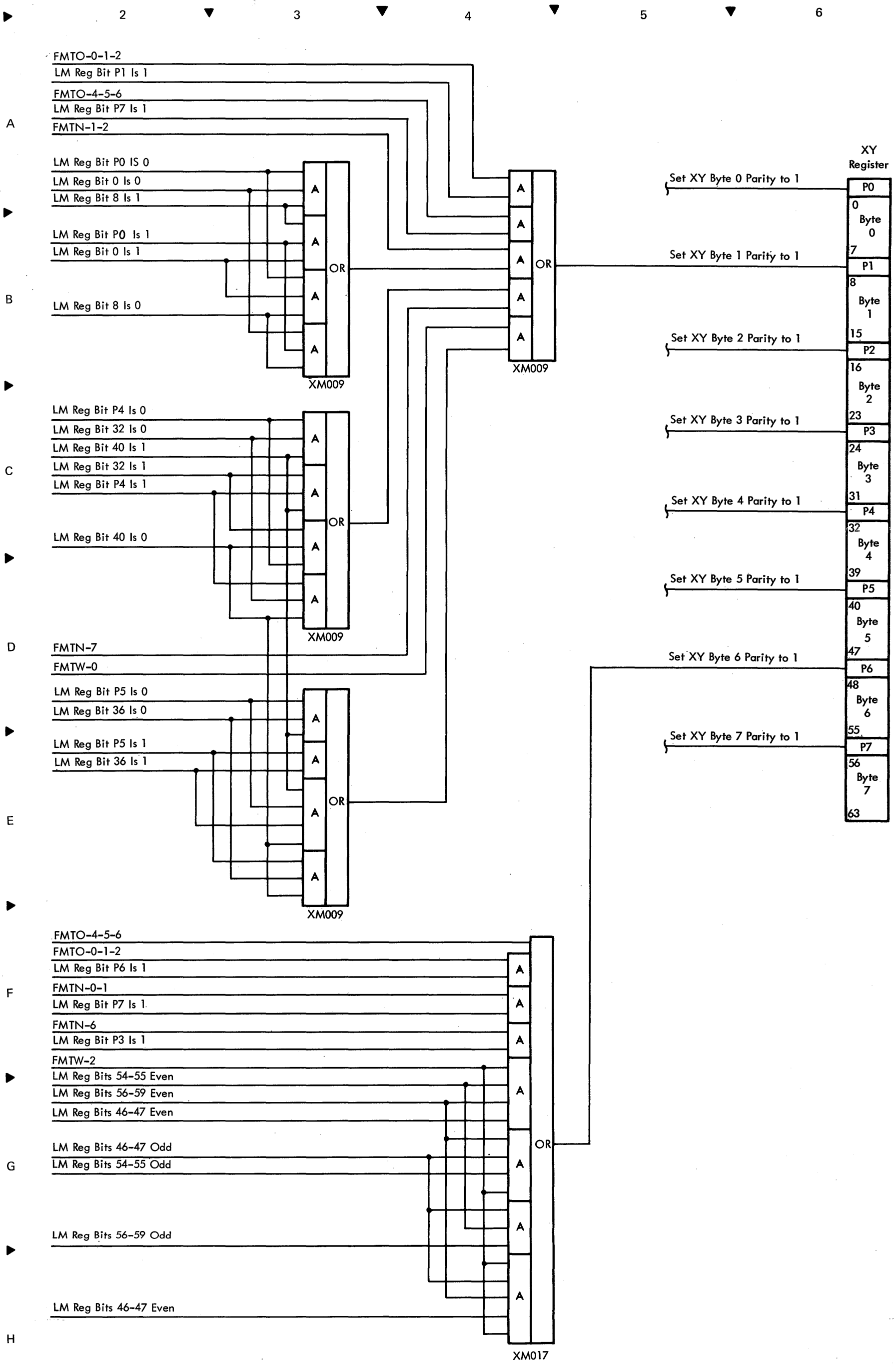


Diagram 4-212. XY Register Parity Prediction Logic

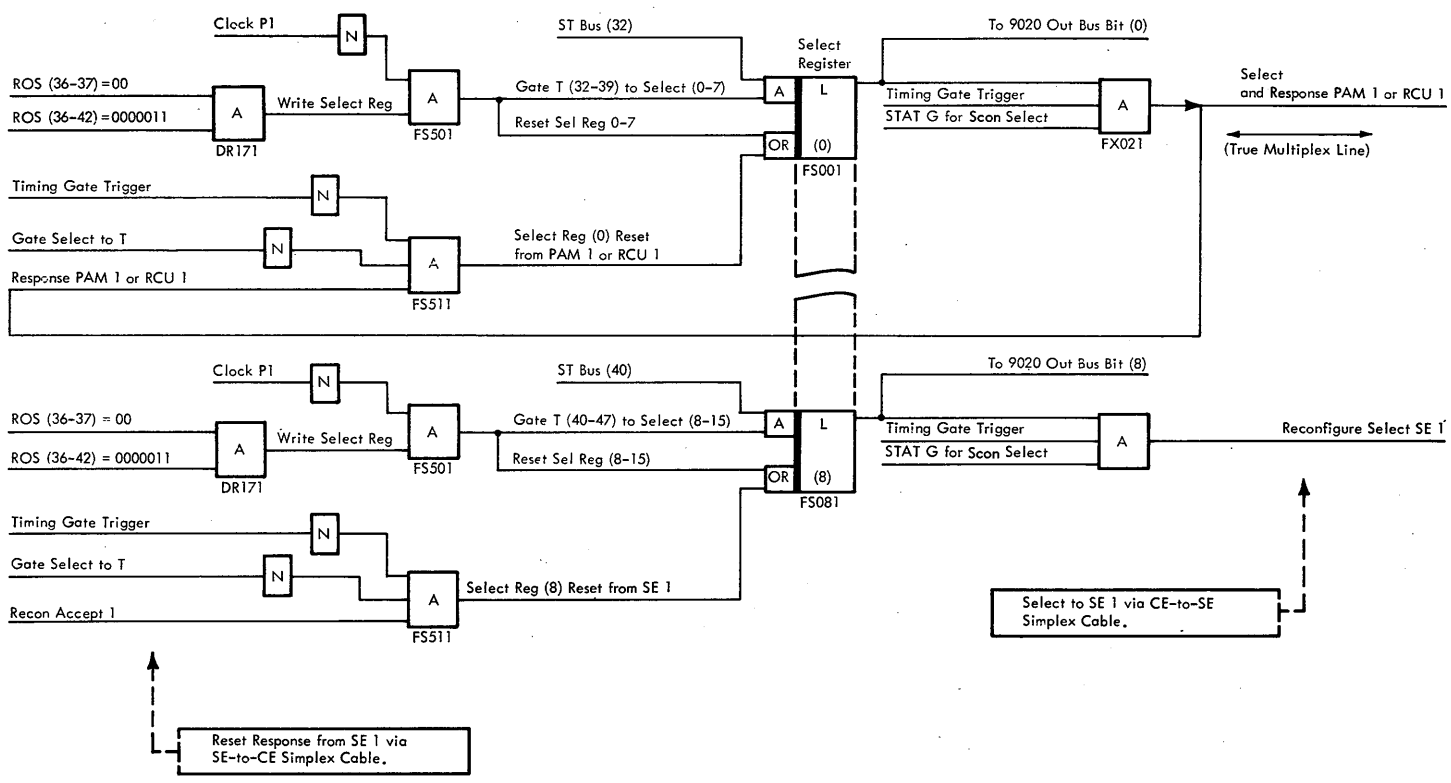


Diagram 4-213. Select Register - Select Signal Generation and Response Reset

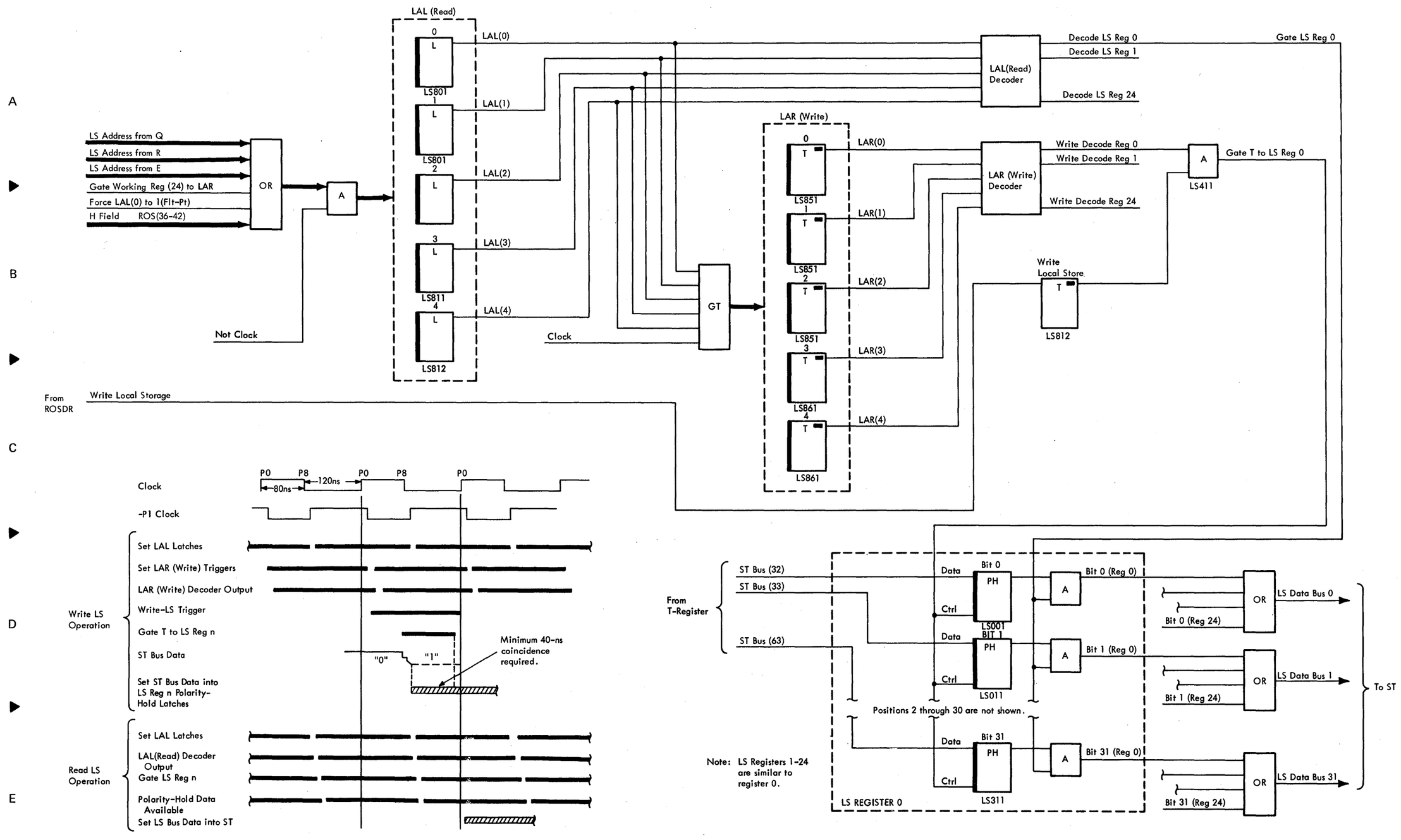


Diagram 4-301. Local Storage Read/Write Controls

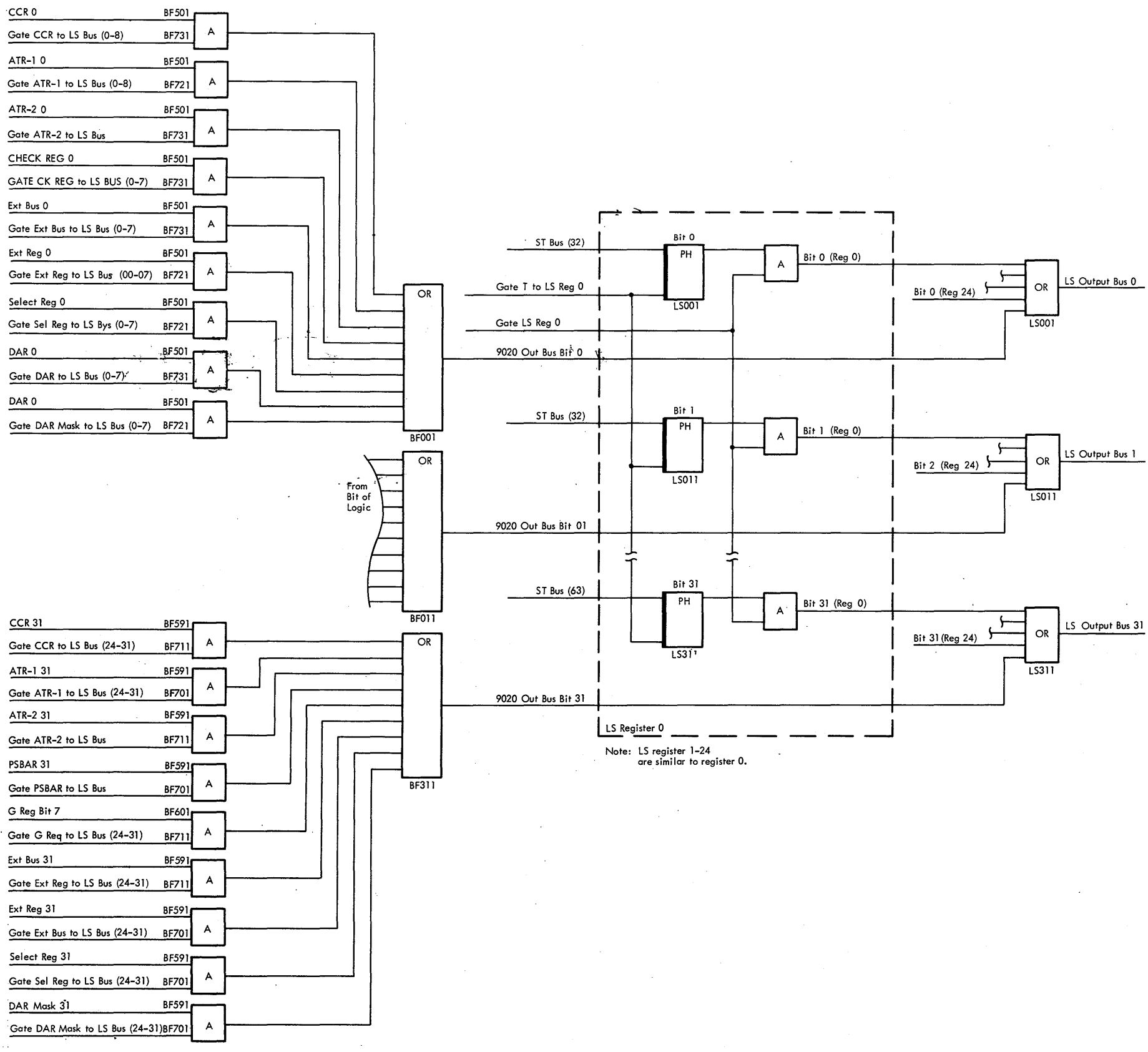


Diagram 4-302. 9020 Out Bus to LS Data Bus Gating Logic

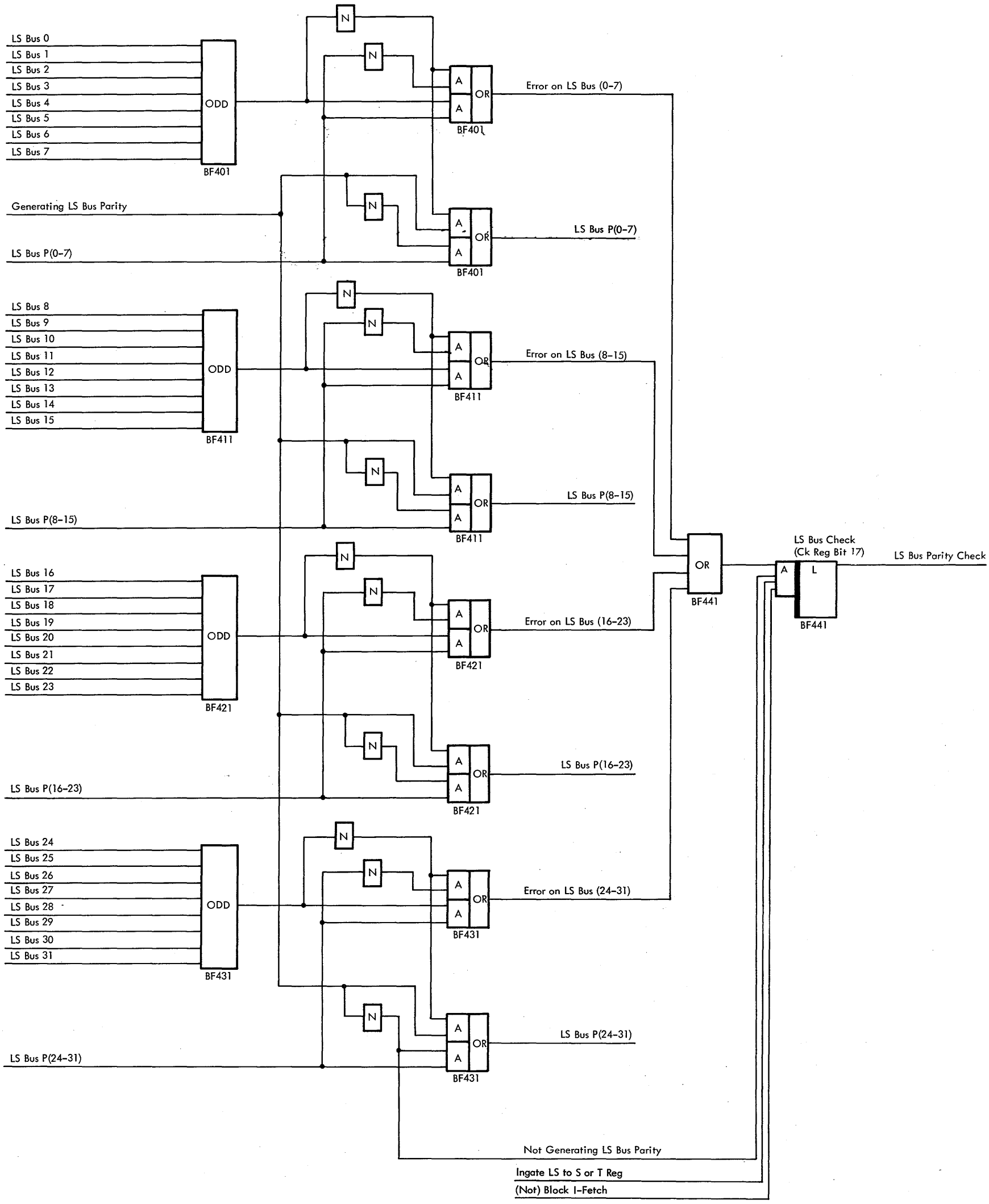


Diagram 4-303. LS Bus Parity Generation or Check

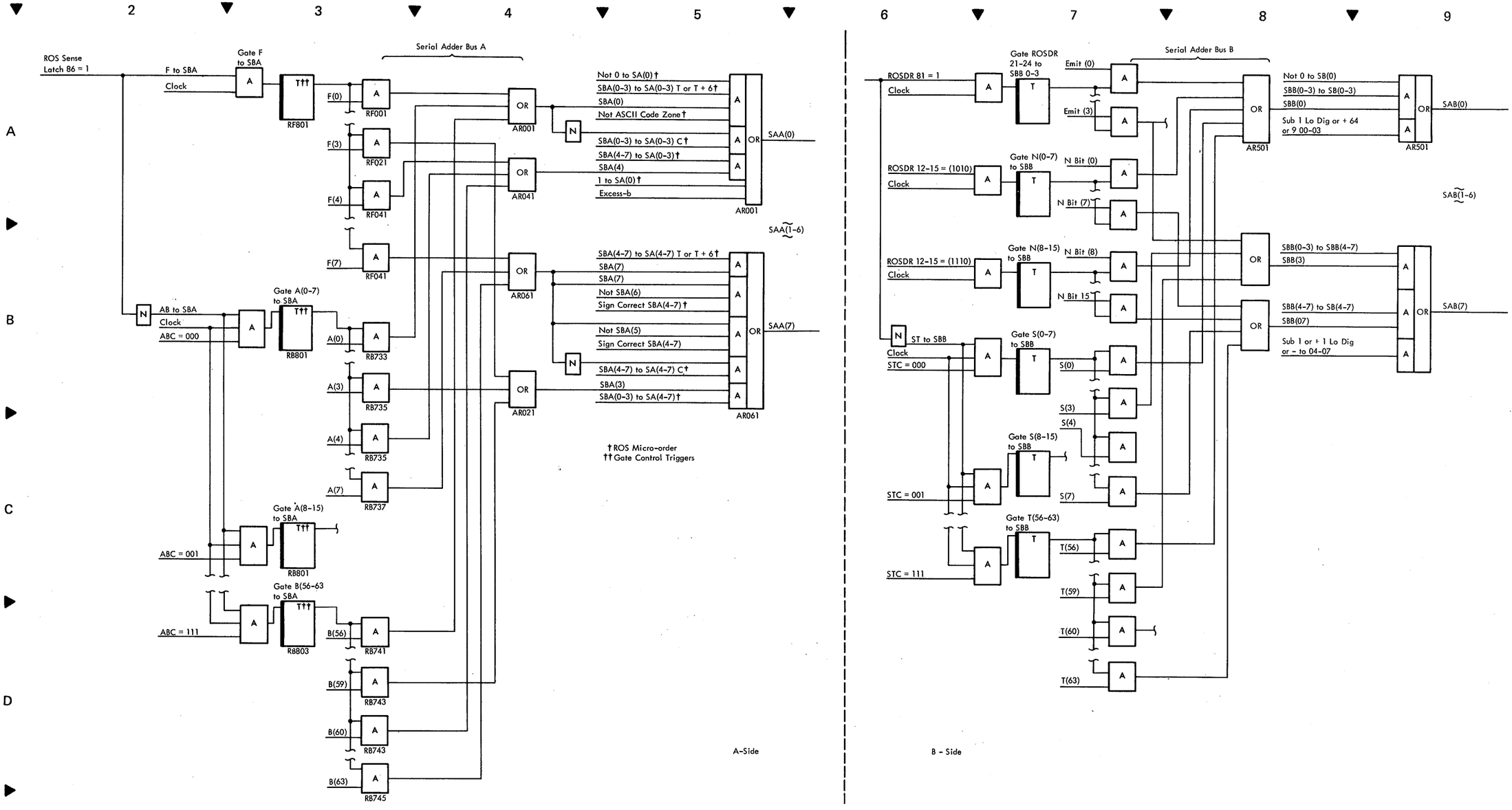


Diagram 4-401. Serial Adder Input Bus Logic

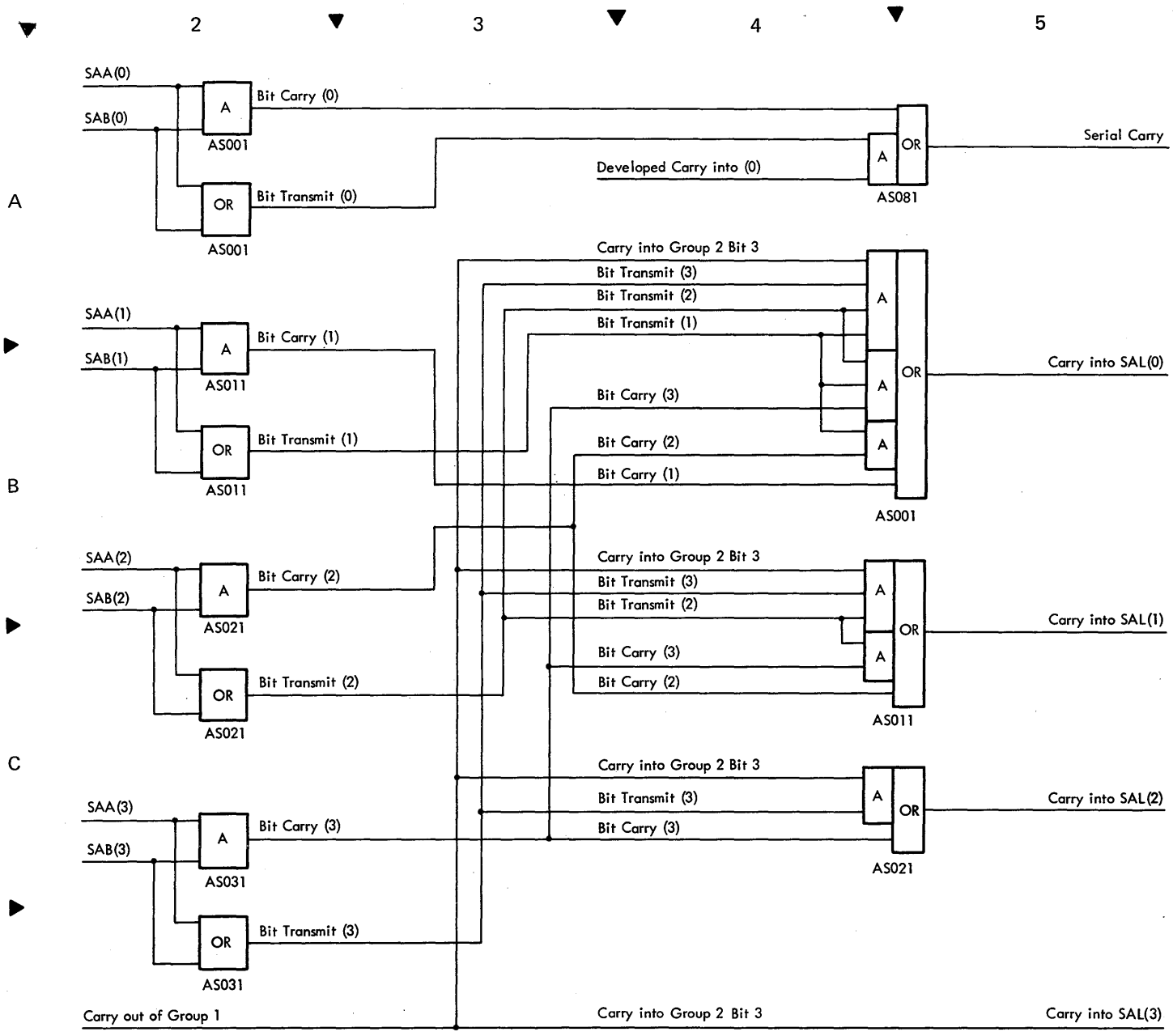


Diagram 4-402. Carry Lookahead Logic, SAL(0-3)

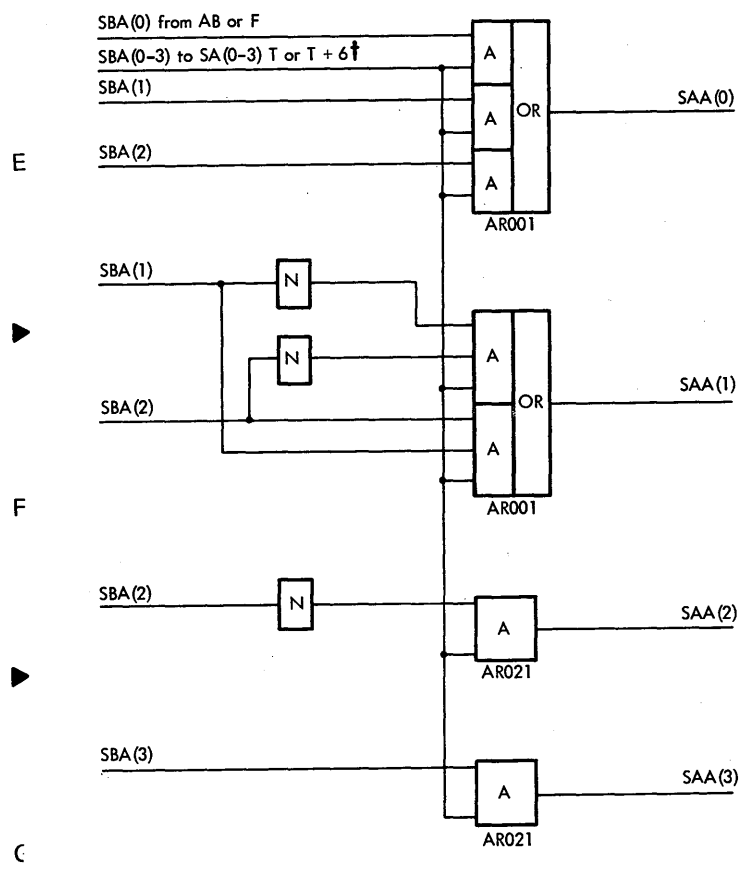


Diagram 4-403. Decimal Add 6 Logic

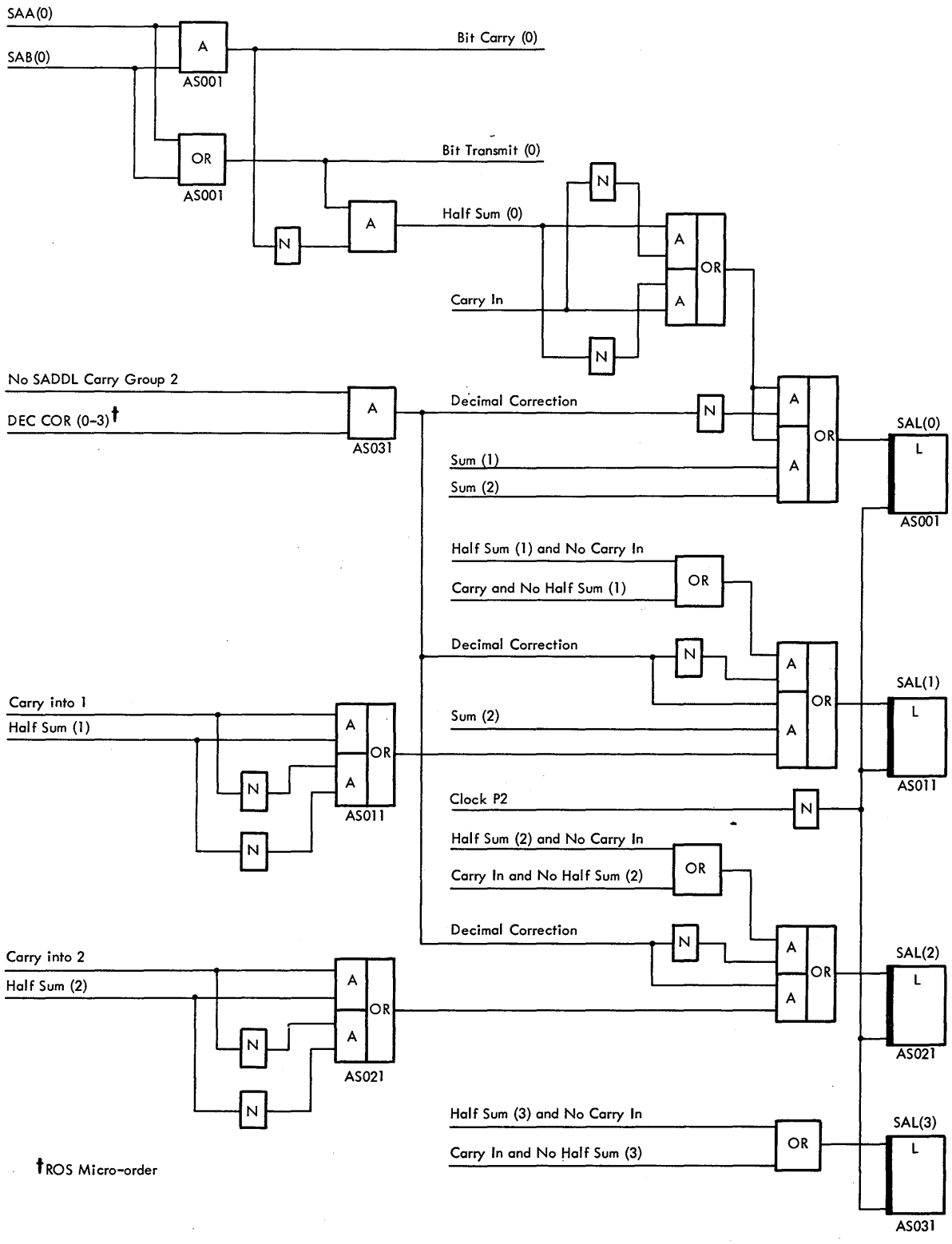


Diagram 4-404. Decimal Correction Logic For SAL (0-3)

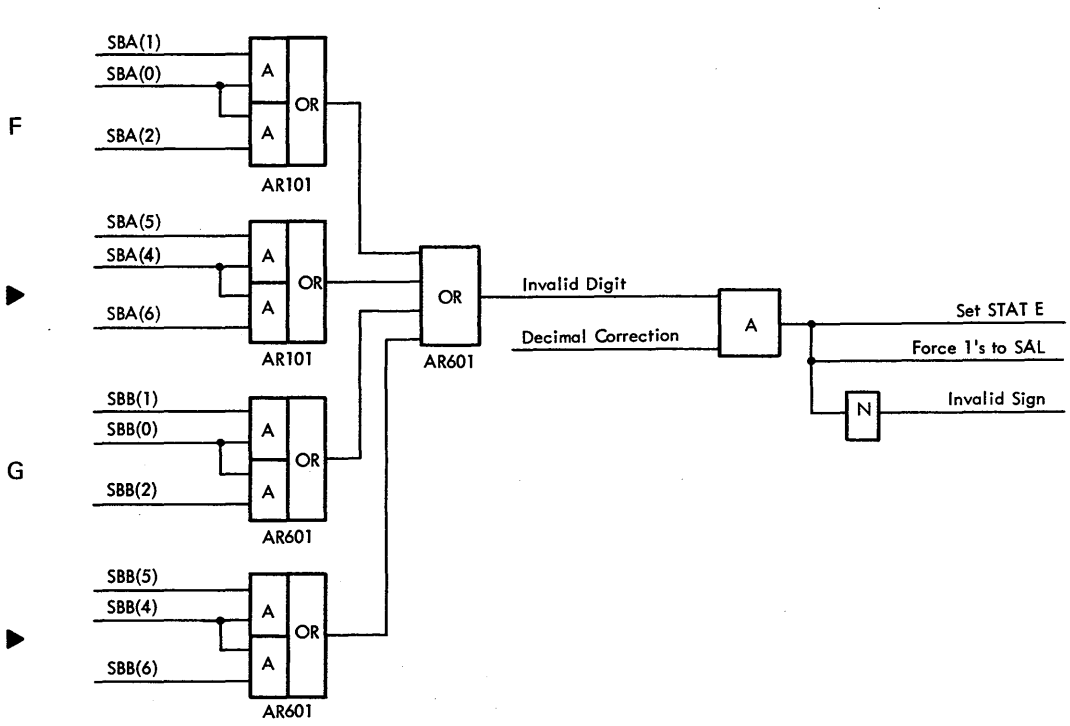


Diagram 4-405. Invalid Digit Logic

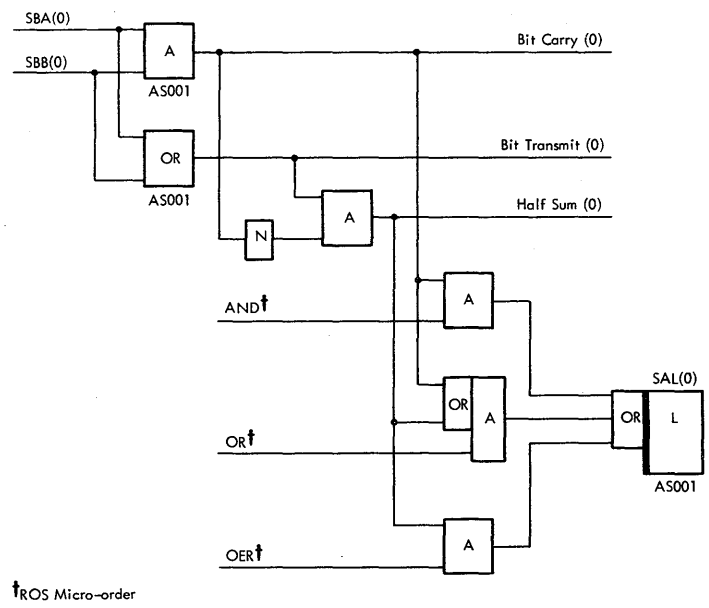


Diagram 4-406. Logical Functions, SAL (0)

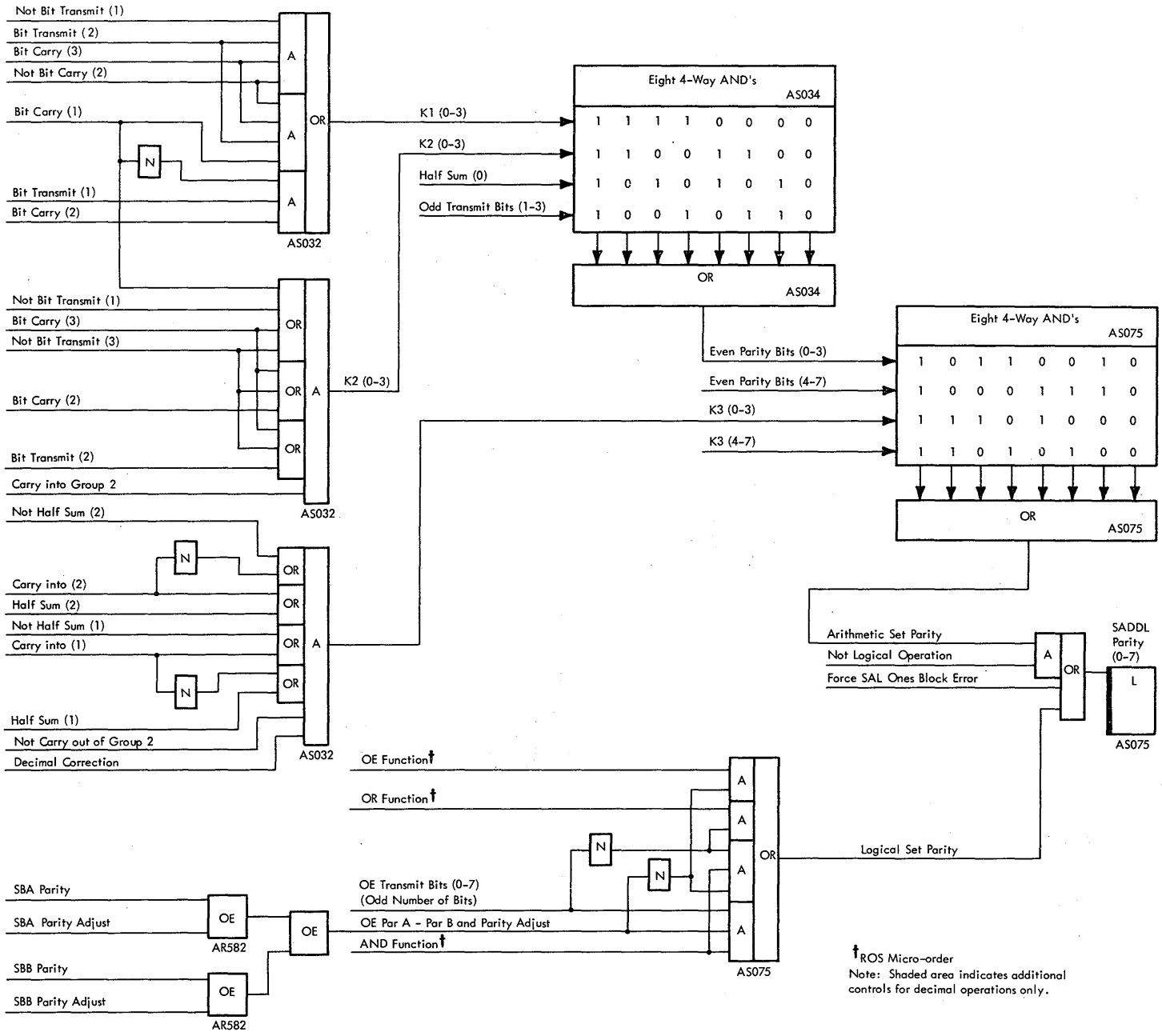


Diagram 4-407. Serial Adder Parity Predict Logic

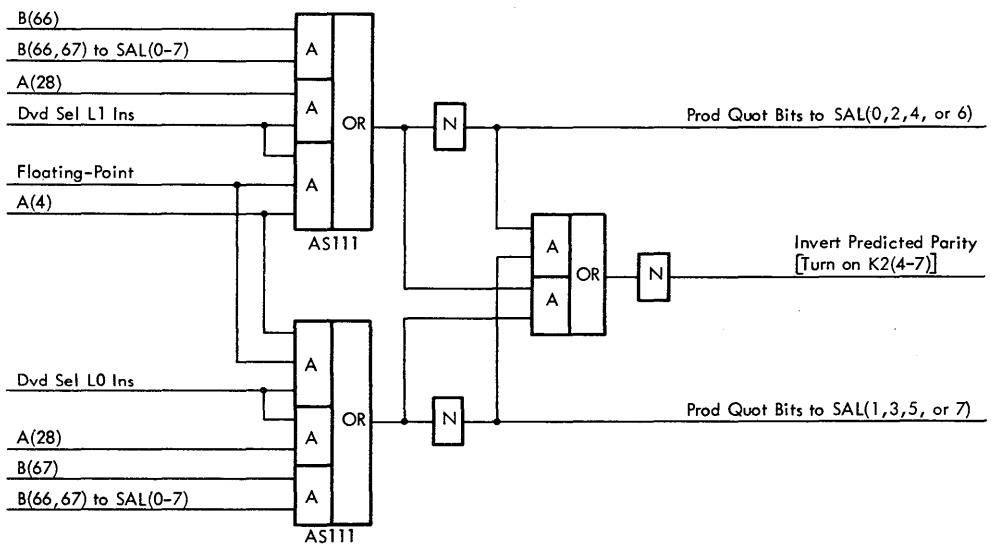
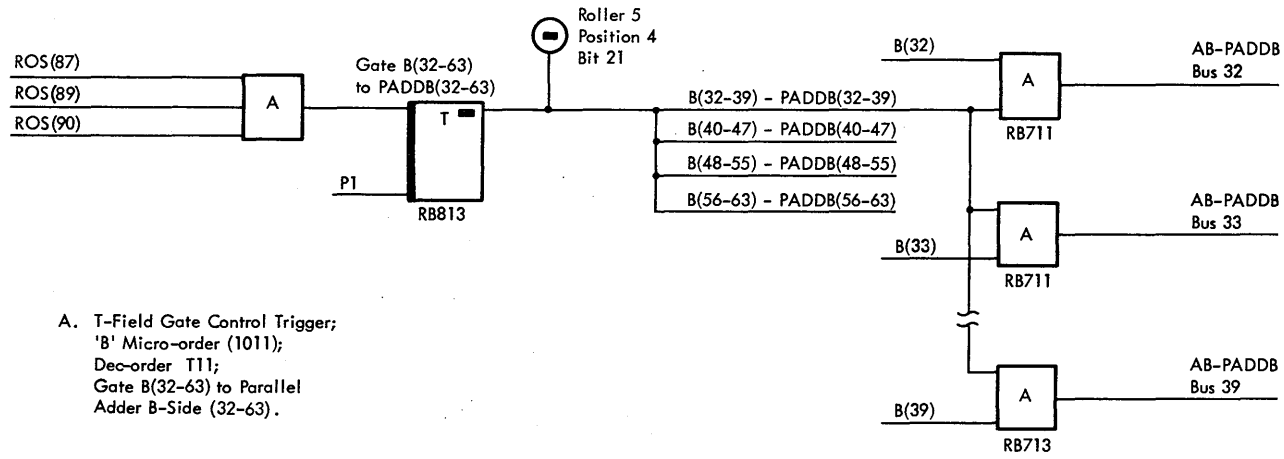


Diagram 4-408. Serial Adder Product-Quotient Bit Logic



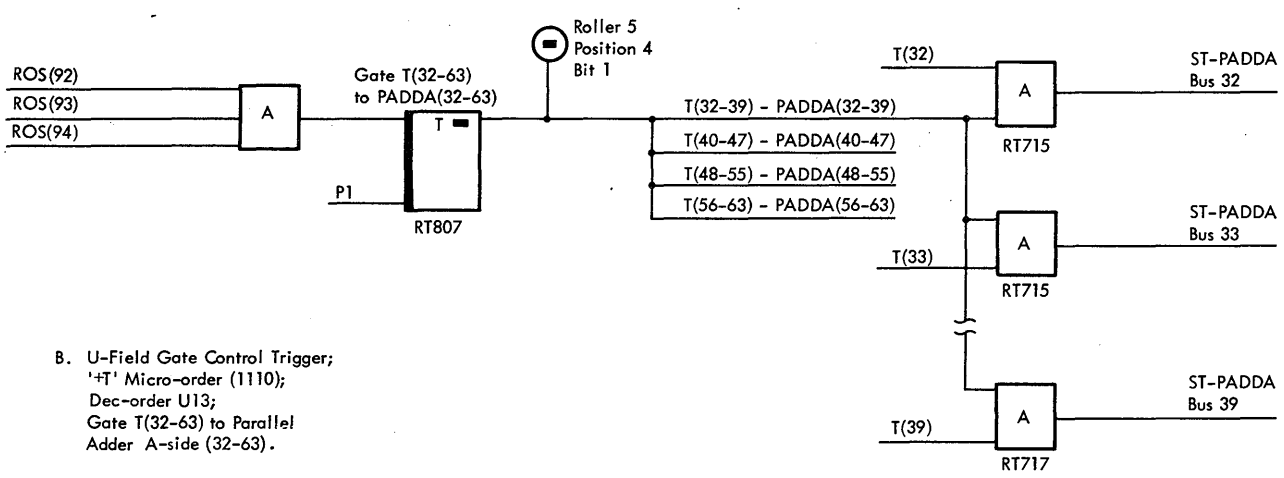
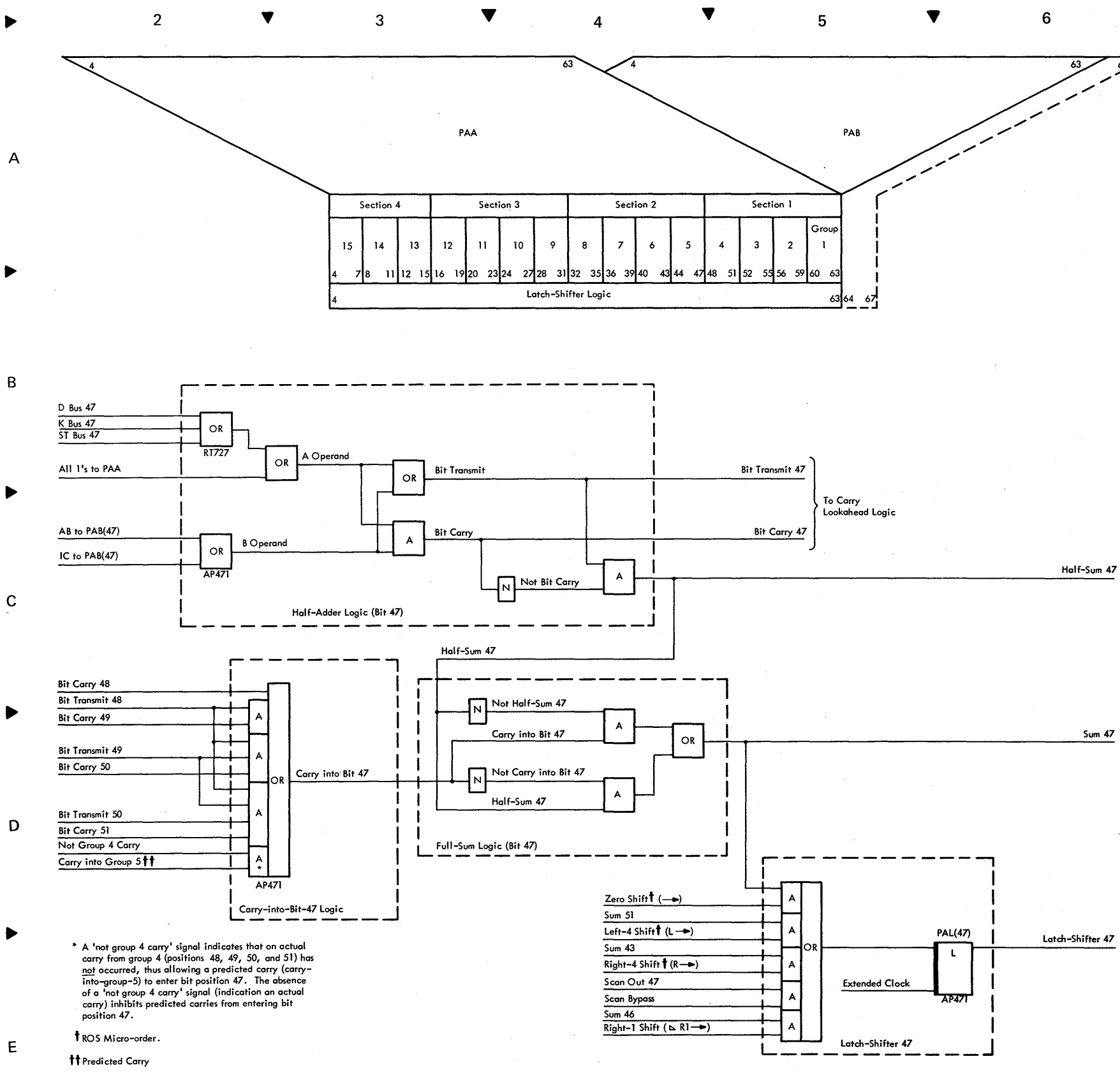


Diagram 4-409. Gate Control Triggers for 'B + T' Micro-order



* A 'not group 4 carry' signal indicates that on actual carry from group 4 (positions 48, 49, 50, and 51) has not occurred, thus allowing a predicted carry (carry-into-group-5) to enter bit position 47. The absence of a 'not group 4 carry' signal (indication an actual carry) inhibits predicted carries from entering bit position 47.

† ROS Micro-order.

†† Predicted Carry

Diagram 4-410. Parallel Adder Bit-Position Logic (Bit 47)

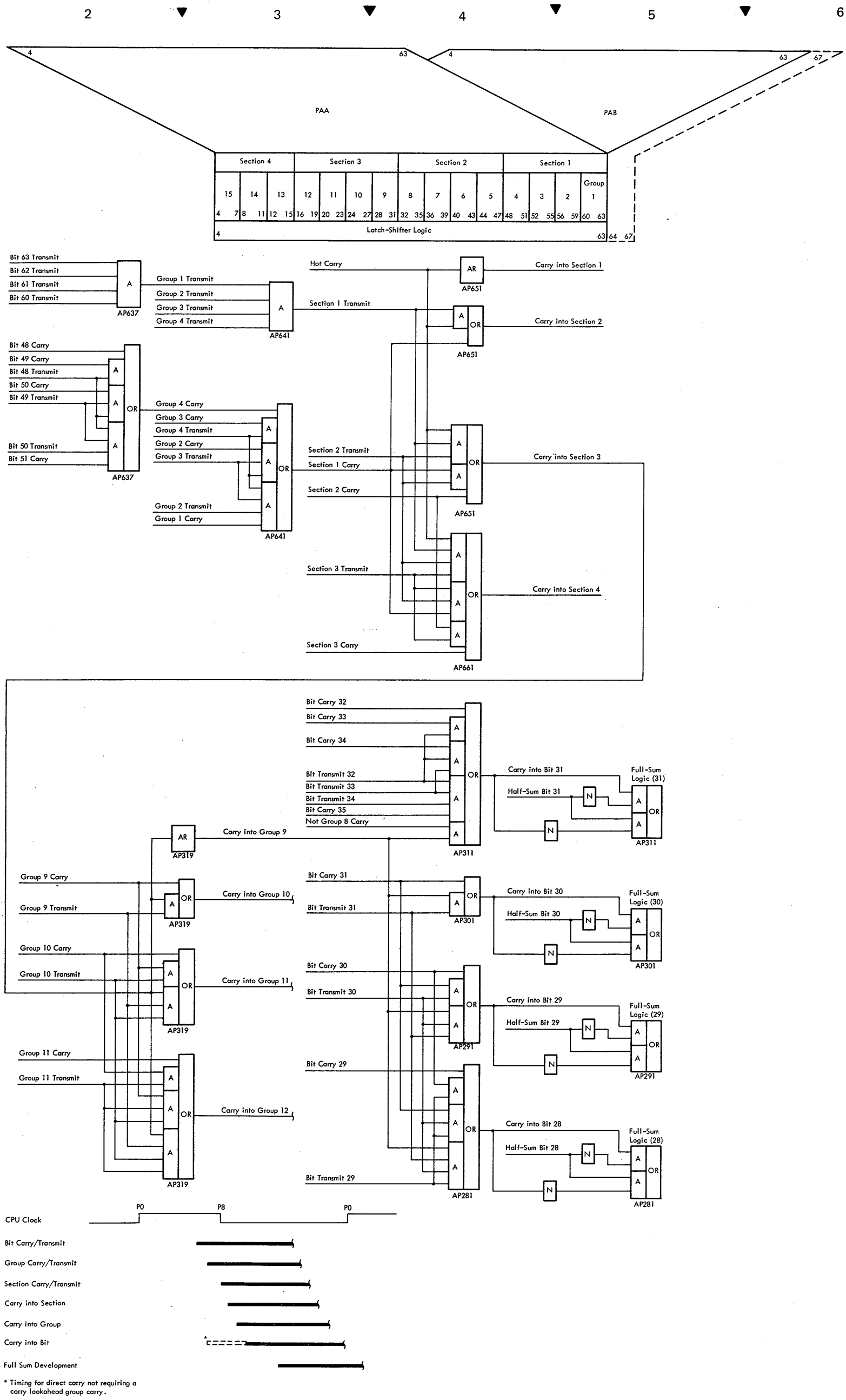


Diagram 4-411. Parallel Adder Carry Lookahead Logic

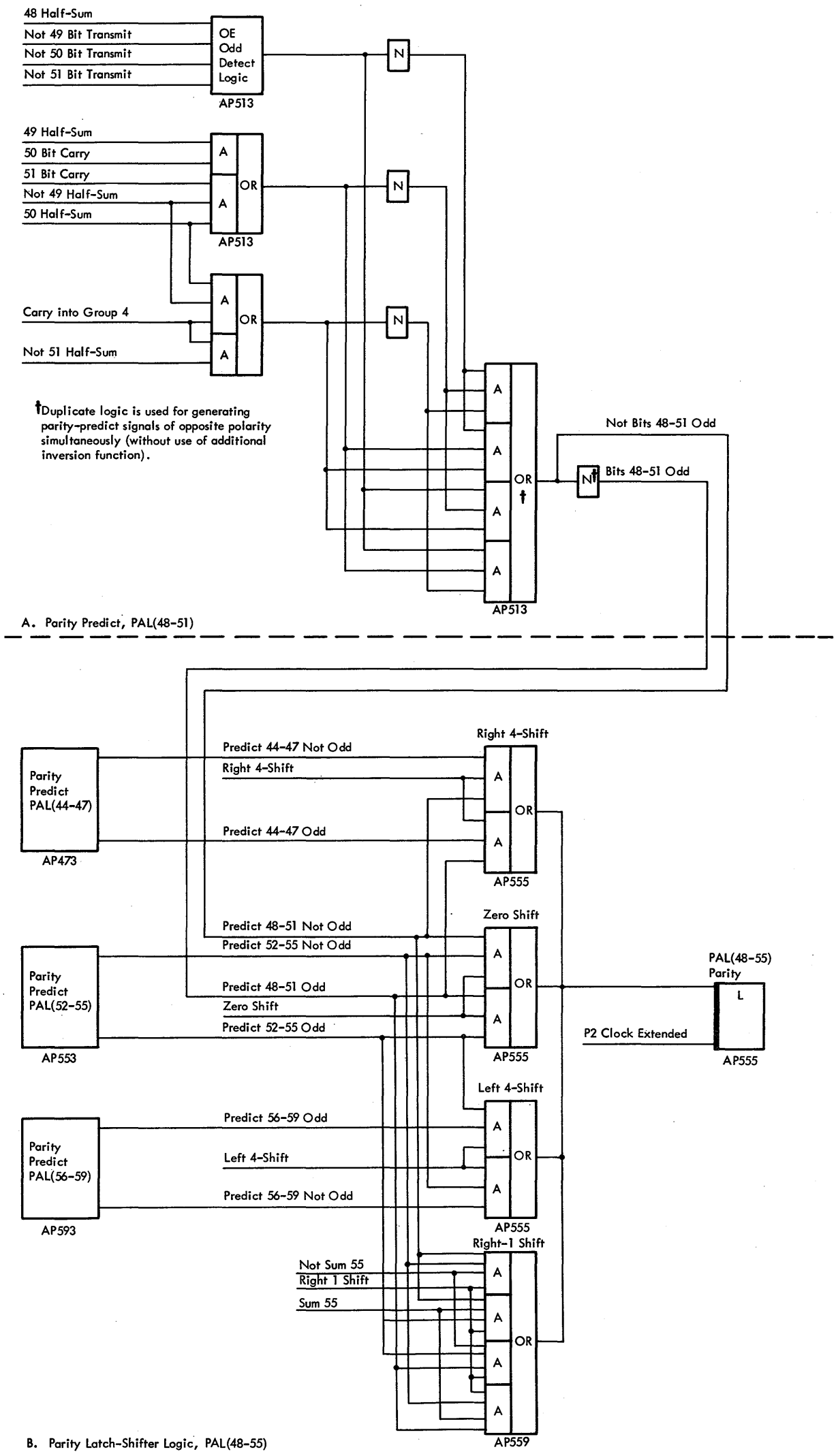


Diagram 4-412. Parity Generation, PAL (48-55)

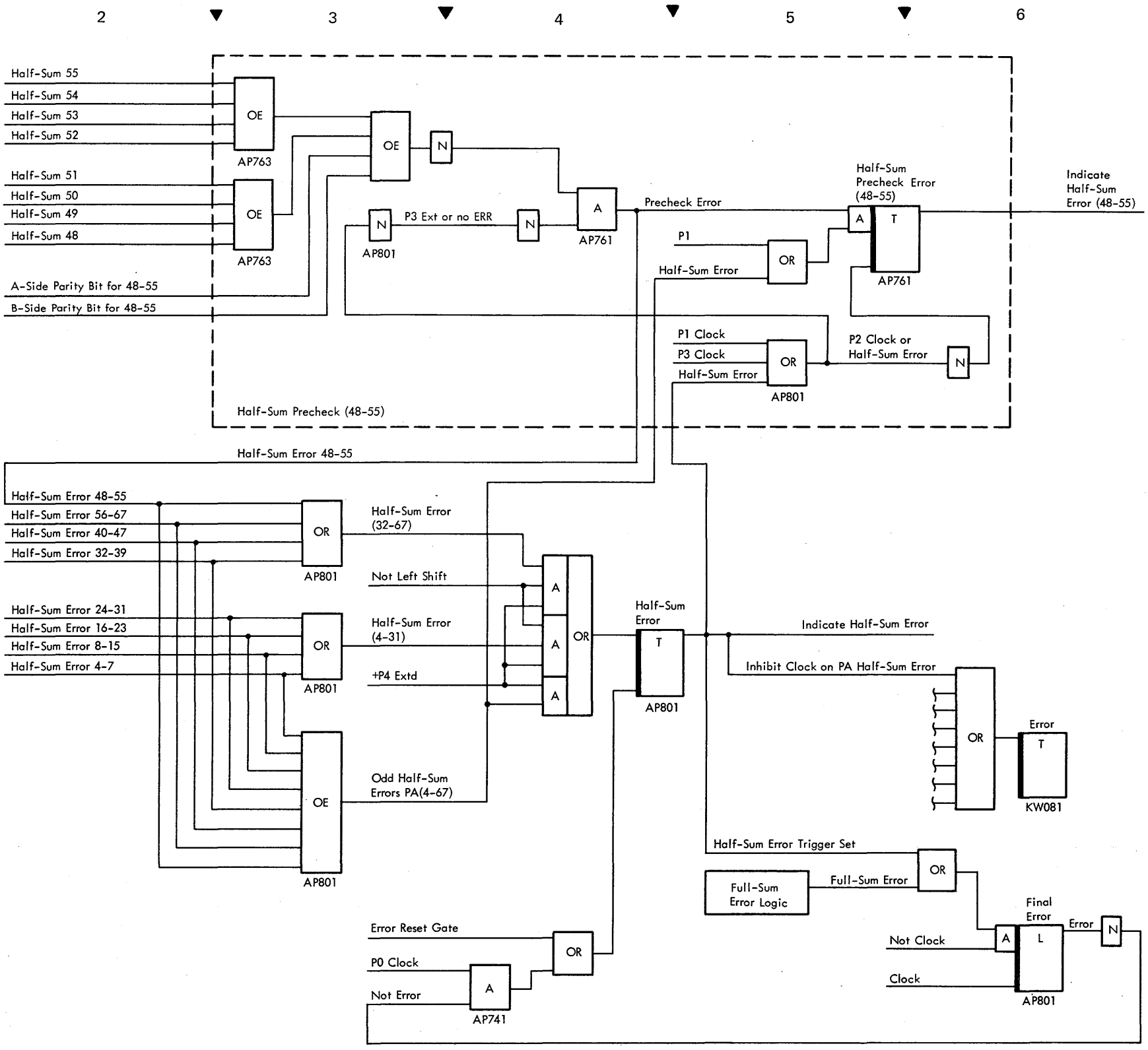


Diagram 4-413. Parallel Adder Half-Sum Checking Logic, PA (48-55)

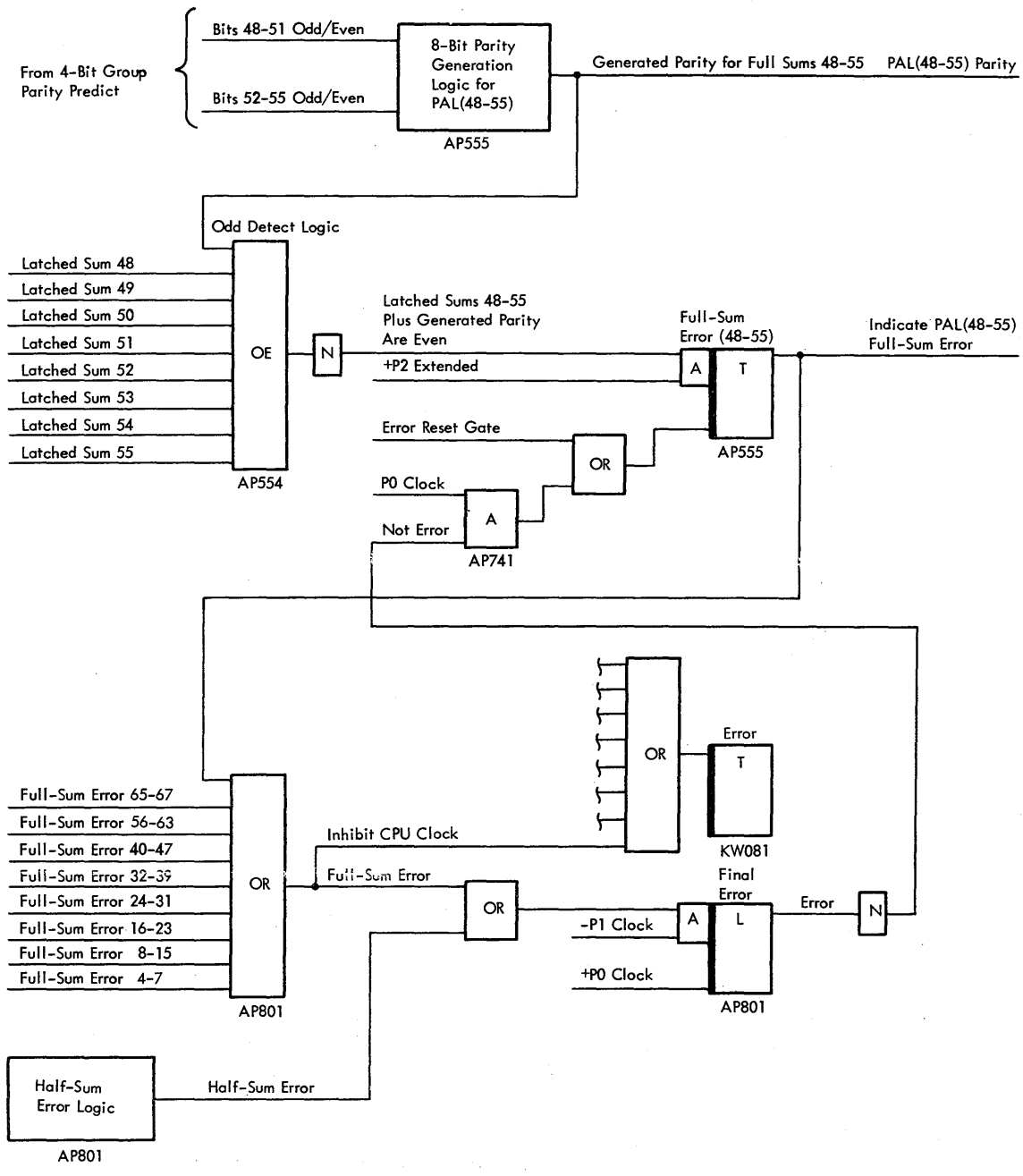
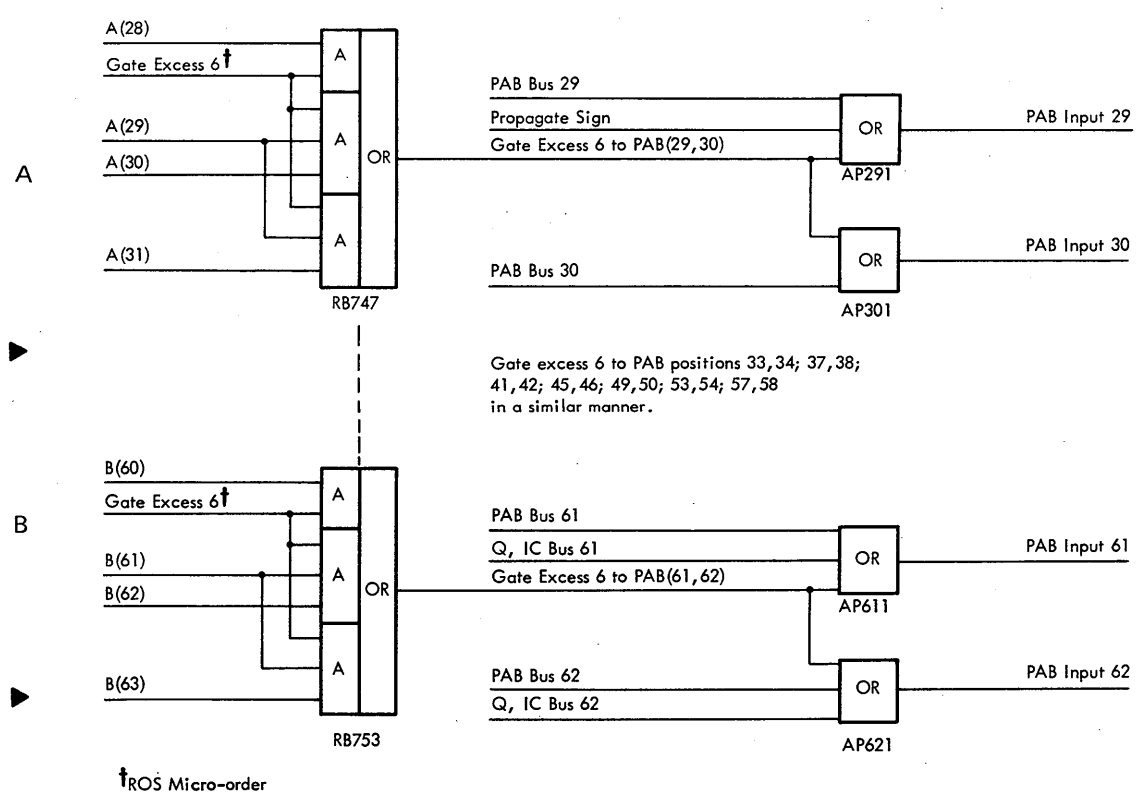


Diagram 4-414. Parallel Adder Full-Sum Checking Logic, PA(48-55)



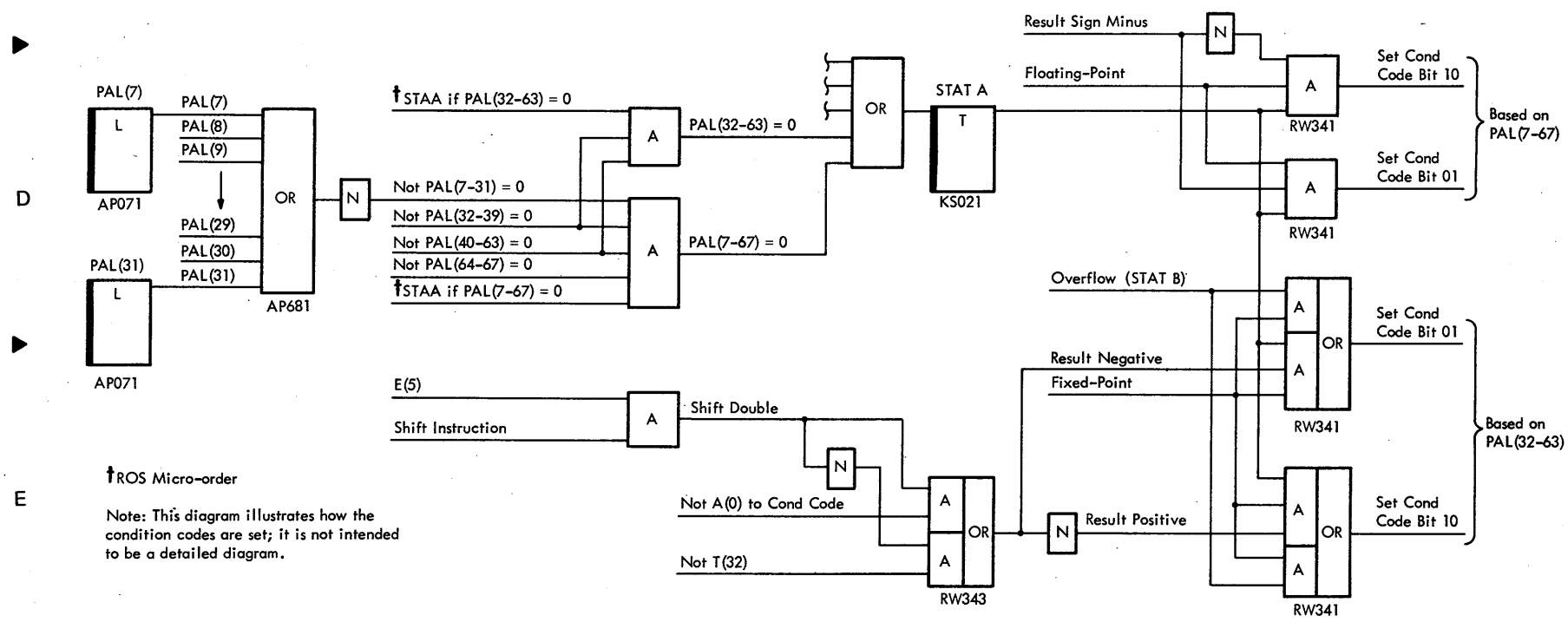


Diagram 4-416. Parallel Adder Set-Condition-Code Logic

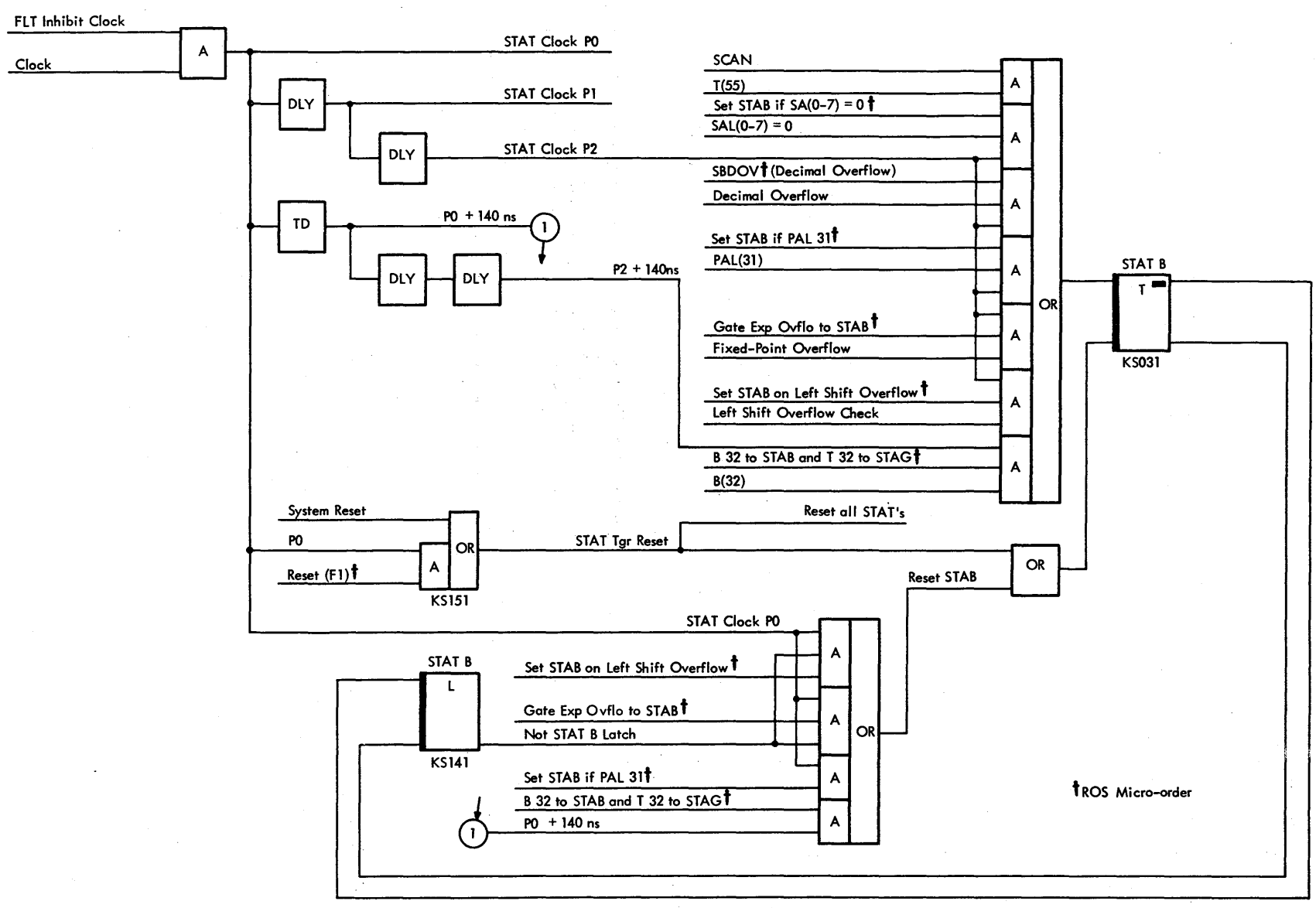
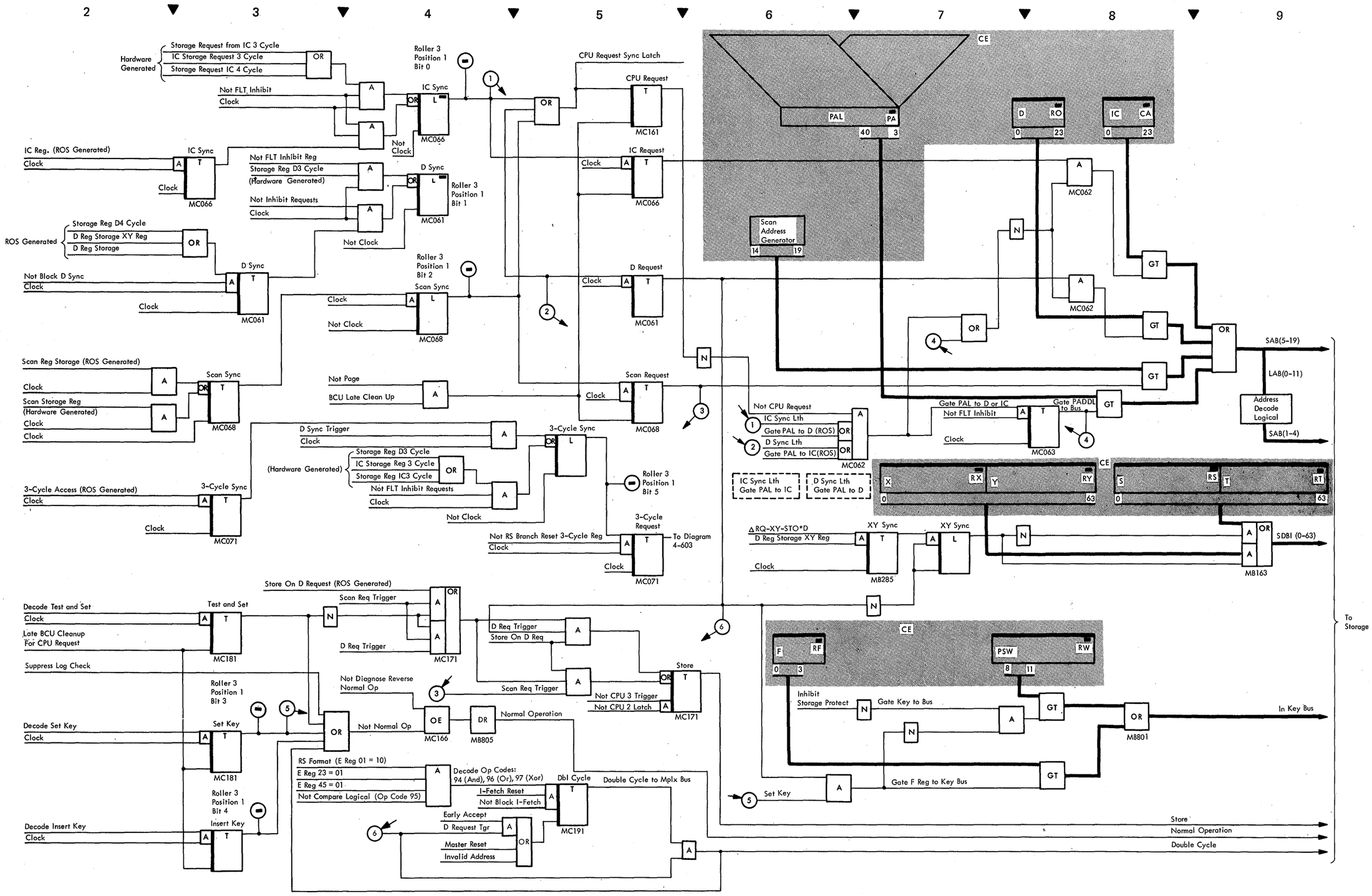


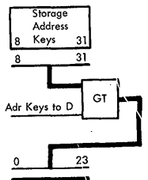
Diagram 4-501. STAT B Logic

Diagram 4-601. SCI Request Sensing and Gating Logic



CE

SCI



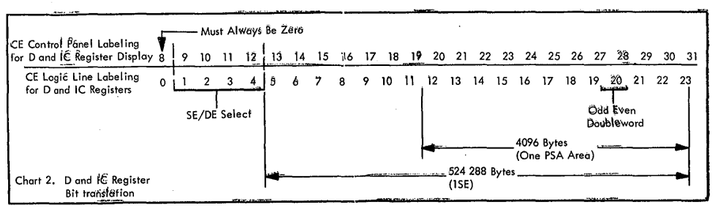


Chart 2. D and IC Register Bit translation

Chart 1. Range of Addresses

	9020D		9020E	
	From	To	From	To
SE1	000,000	524,287	000,000	524,287
SE2	524,288	1,048,575	524,288	1,048,575
SE3	1,048,576	1,572,863	1,048,576	1,572,863
SE4	1,572,864	2,097,151	1,572,864	2,097,151
SE5	2,097,152	2,621,439	2,097,152	2,621,439
SE/DE6	2,621,440	3,145,727	2,621,440	2,883,583
SE/DE7	3,145,728	3,670,015	3,145,728	3,407,871
SE/DE8	3,670,016	4,194,303	3,670,016	3,932,159
SE/DE9	4,194,304	4,718,591	4,194,304	4,456,447
SE/DE10	4,718,592	5,242,879	4,718,592	4,980,735

Diagram 4-602. Address Decode and Gating Logic

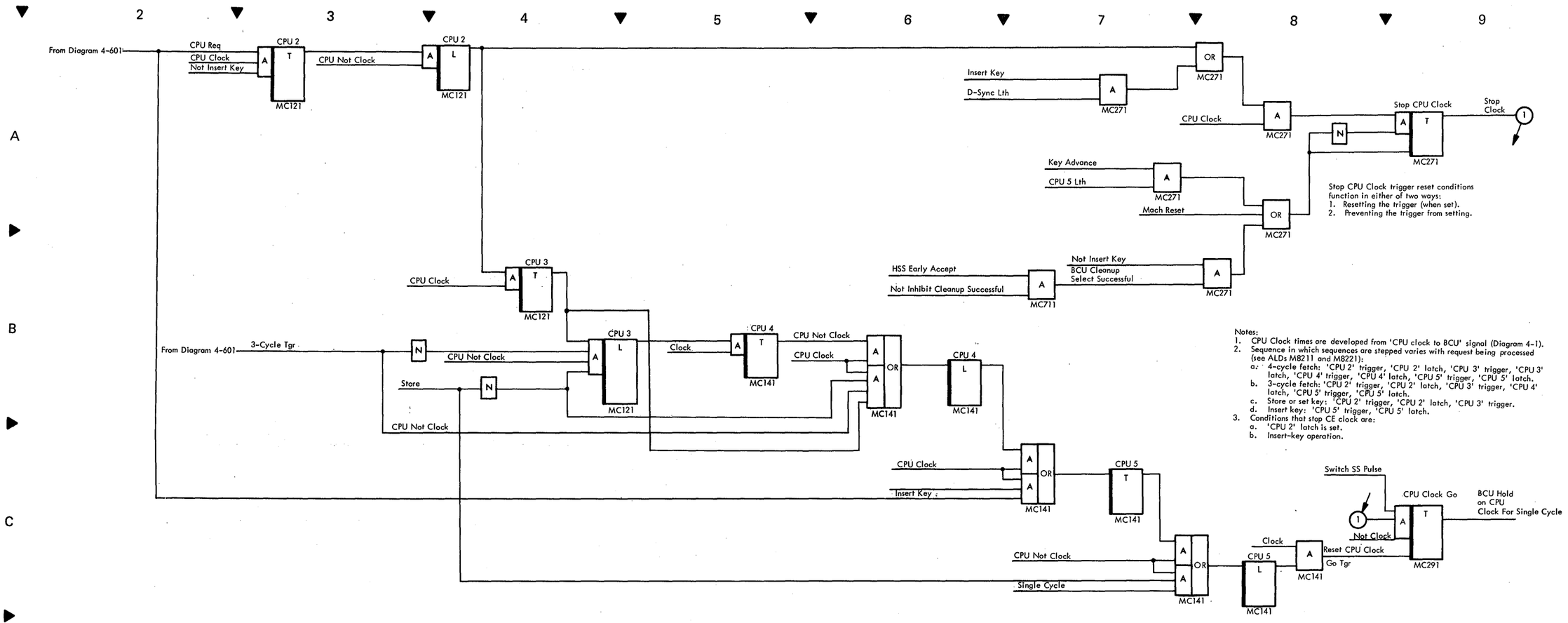


Diagram 4-603. SCI Control Logic for CE Clock

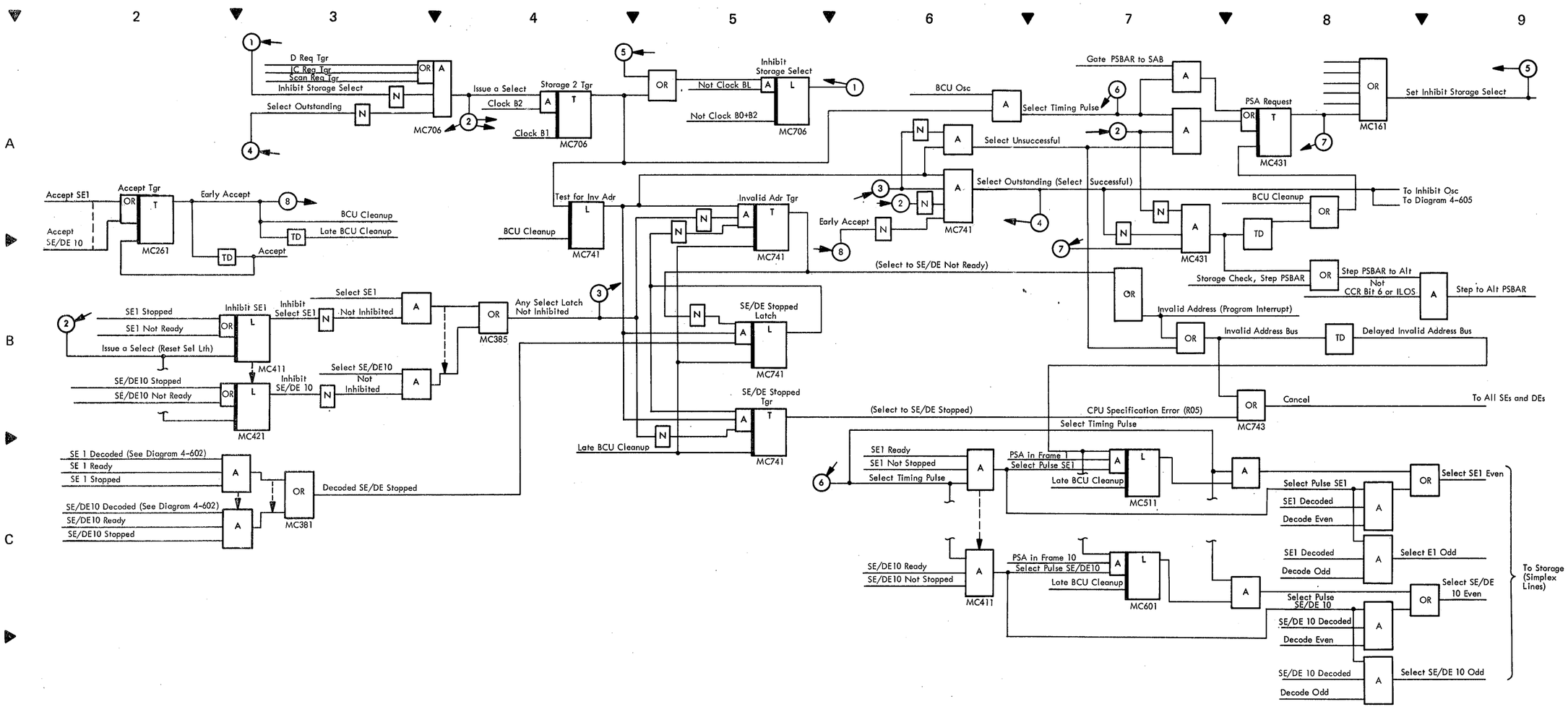


Diagram 4-604. Invalid Address and Frame Stopped Logic (Sheet 1 of 2)

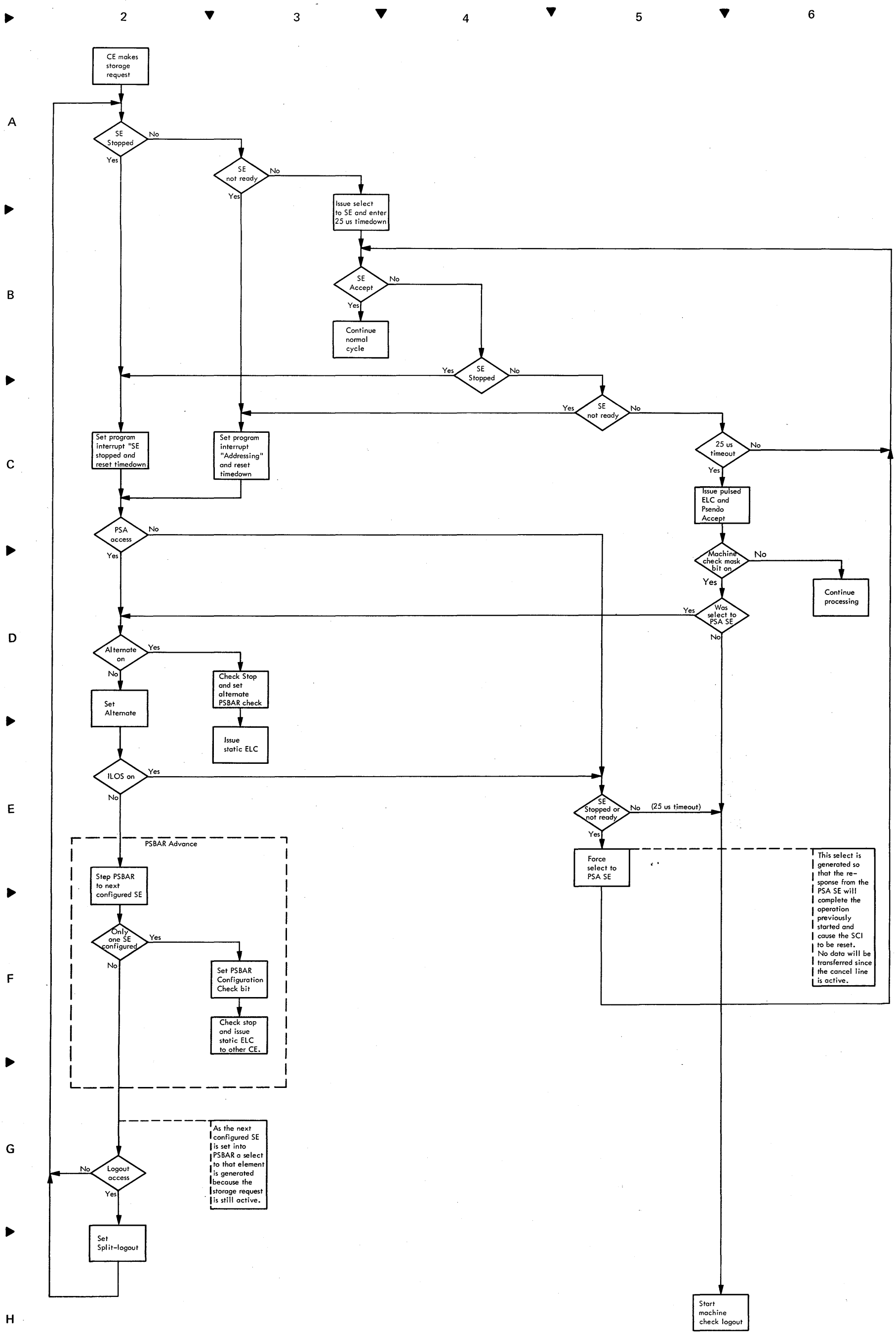
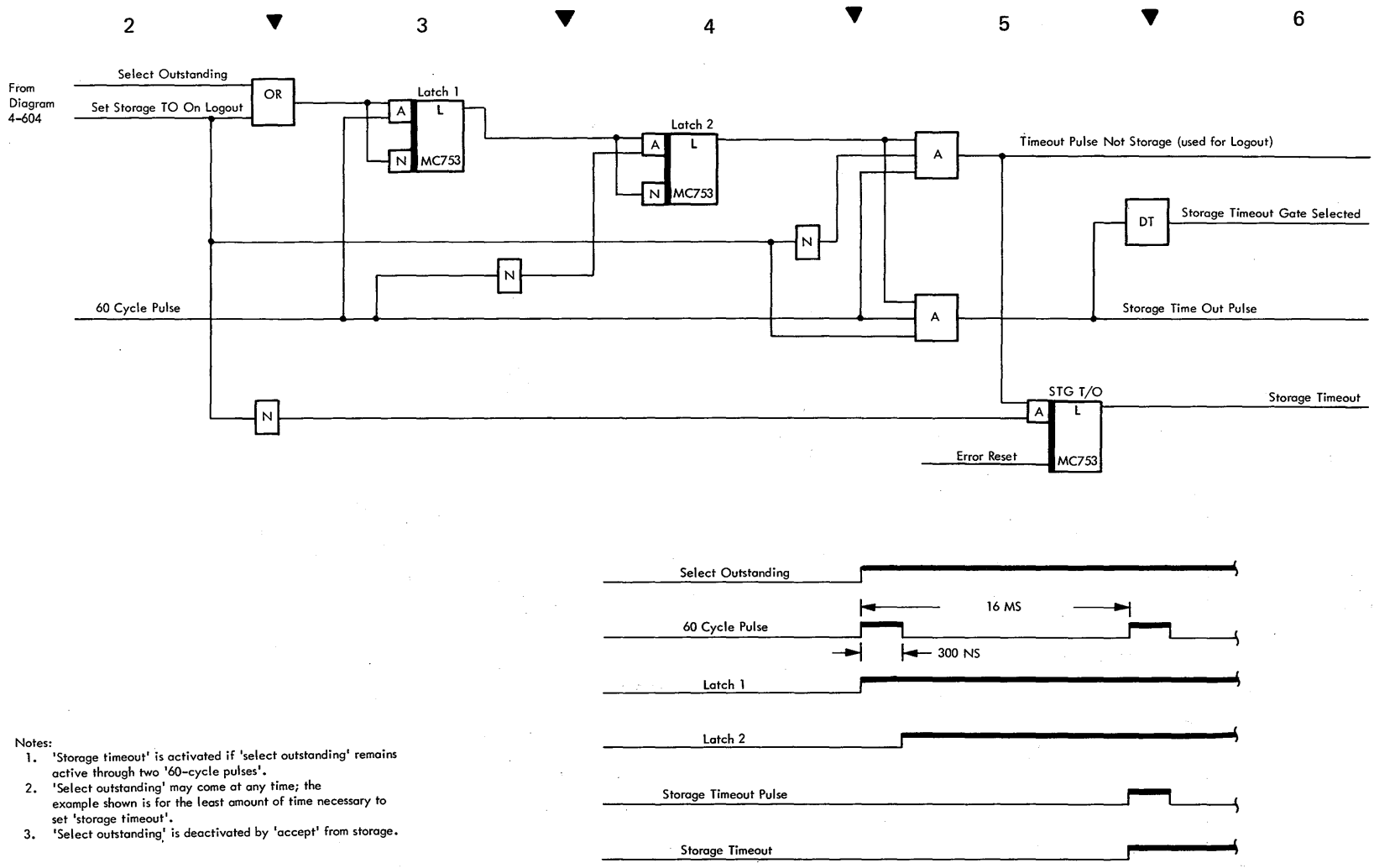


Diagram 4-604. Invalid Address and Frame Stopped Logic (Sheet 2 of 2)



Notes:

1. 'Storage timeout' is activated if 'select outstanding' remains active through two '60-cycle pulses'.
2. 'Select outstanding' may come at any time; the example shown is for the least amount of time necessary to set 'storage timeout'.
3. 'Select outstanding' is deactivated by 'accept' from storage.

Diagram 4-605. Storage Timeout Logic

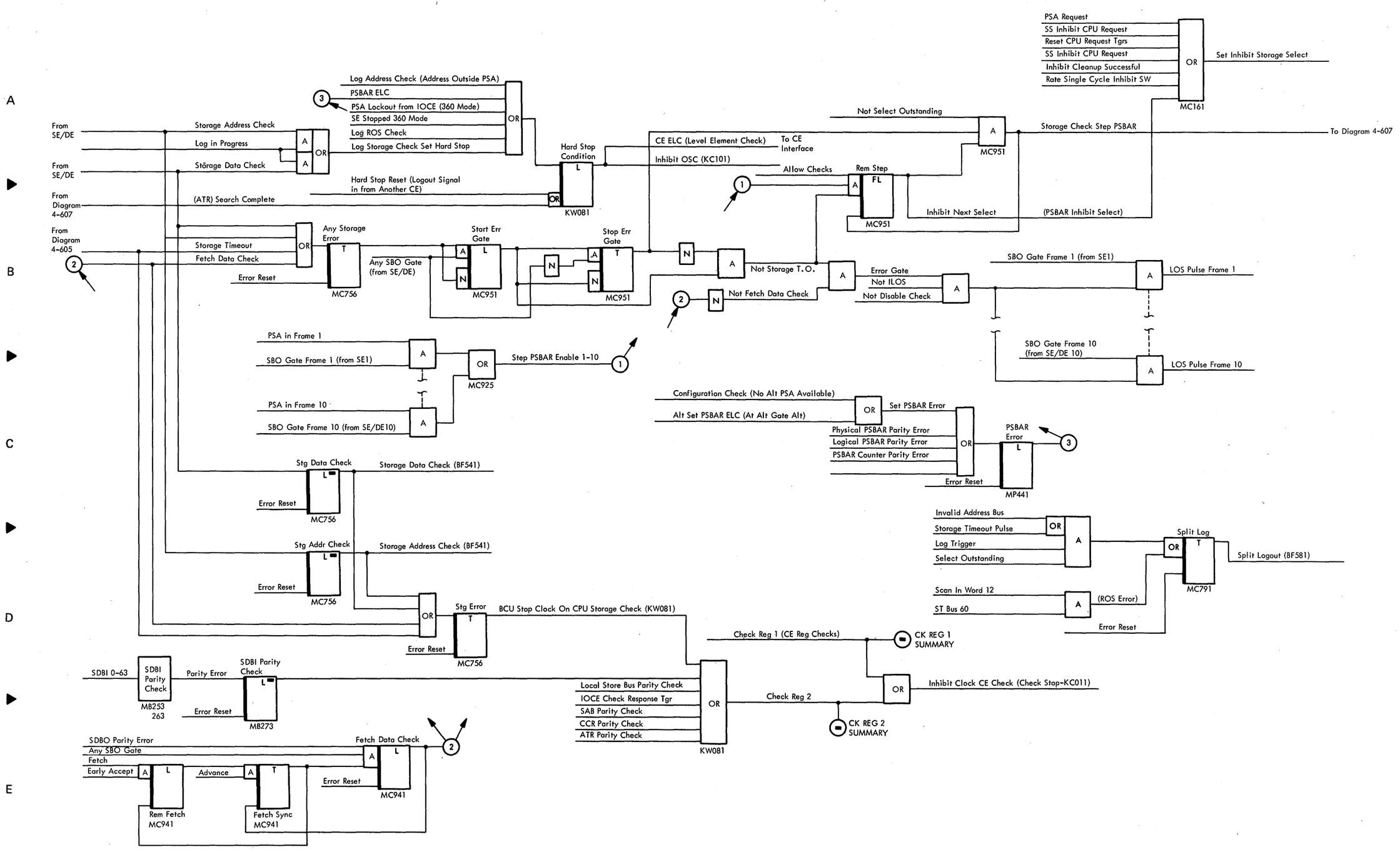
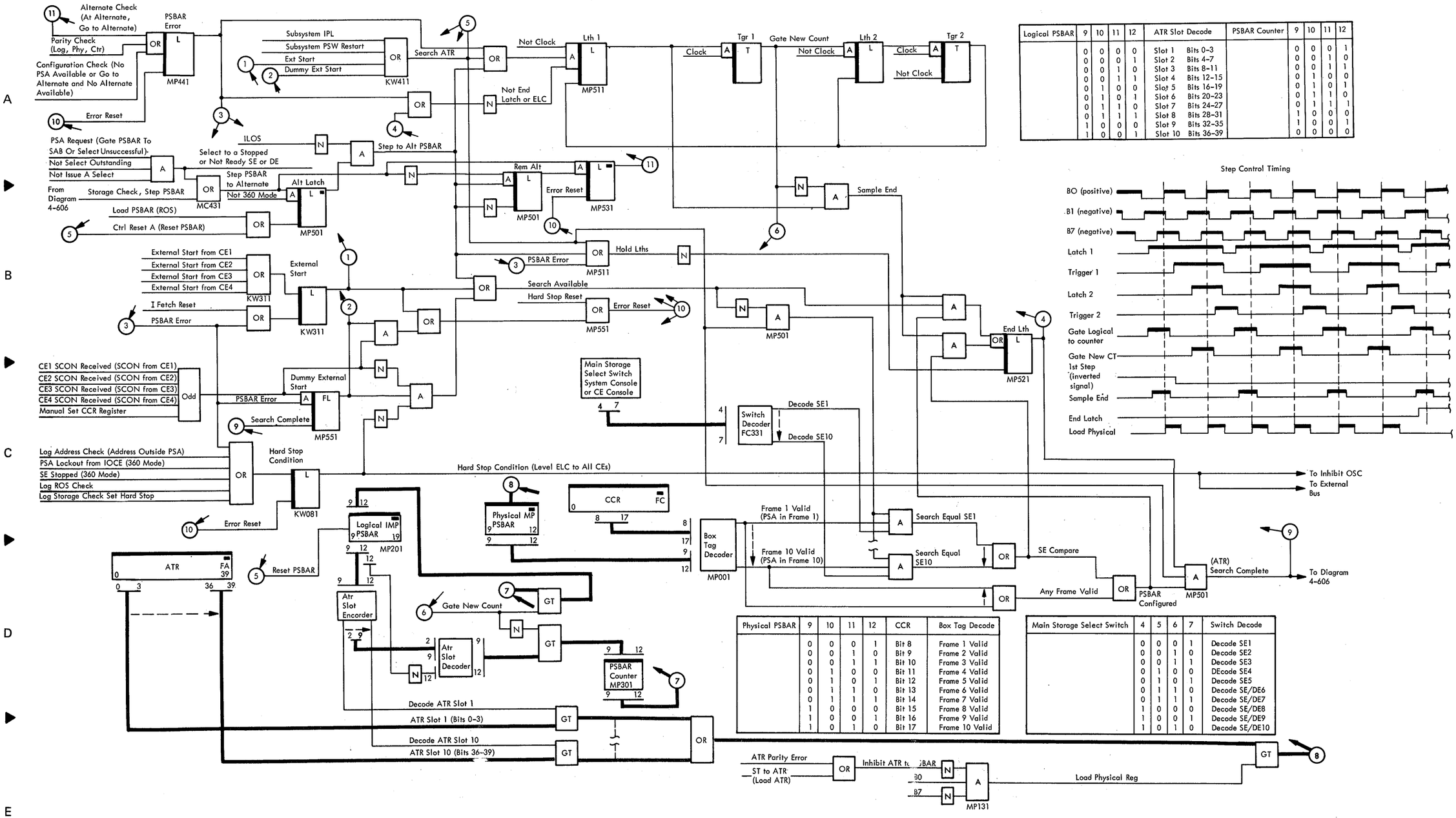


Diagram 4-606. Error Handling Logic



Logical PSBAR	9	10	11	12	ATR Slot Decode	PSBAR Counter	9	10	11	12
	0	0	0	0	Slot 1 Bits 0–3		0	0	0	1
	0	0	0	1	Slot 2 Bits 4–7		0	0	1	0
	0	0	1	0	Slot 3 Bits 8–11		0	0	1	1
	0	0	1	1	Slot 4 Bits 12–15		0	1	0	0
	0	1	0	0	Slot 5 Bits 16–19		0	1	0	1
	0	1	0	1	Slot 6 Bits 20–23		0	1	1	0
	0	1	1	0	Slot 7 Bits 24–27		0	1	1	1
	0	1	1	1	Slot 8 Bits 28–31		1	0	0	0
	1	0	0	0	Slot 9 Bits 32–35		1	0	0	1
	1	0	0	1	Slot 10 Bits 36–39		1	0	1	0

Physical PSBAR	9	10	11	12	CCR	Box Tag Decode
	0	0	0	1	Bit 8	Frame 1 Valid
	0	0	1	0	Bit 9	Frame 2 Valid
	0	0	1	1	Bit 10	Frame 3 Valid
	0	1	0	0	Bit 11	Frame 4 Valid
	0	1	0	1	Bit 12	Frame 5 Valid
	0	1	1	0	Bit 13	Frame 6 Valid
	0	1	1	1	Bit 14	Frame 7 Valid
	1	0	0	0	Bit 15	Frame 8 Valid
	1	0	0	1	Bit 16	Frame 9 Valid
	1	0	1	0	Bit 17	Frame 10 Valid

Main Storage Select Switch	4	5	6	7	Switch Decode
	0	0	0	1	Decode SE1
	0	0	1	0	Decode SE2
	0	0	1	1	Decode SE3
	0	1	0	0	DEcode SE4
	0	1	0	1	Decode SE5
	0	1	1	0	Decode SE/DE6
	0	1	1	1	Decode SE/DE7
	1	0	0	0	Decode SE/DE8
	1	0	0	1	Decode SE/DE9
	1	0	1	0	Decode SE/DE10

Diagram 4-607. PSBAR Step Control Logic

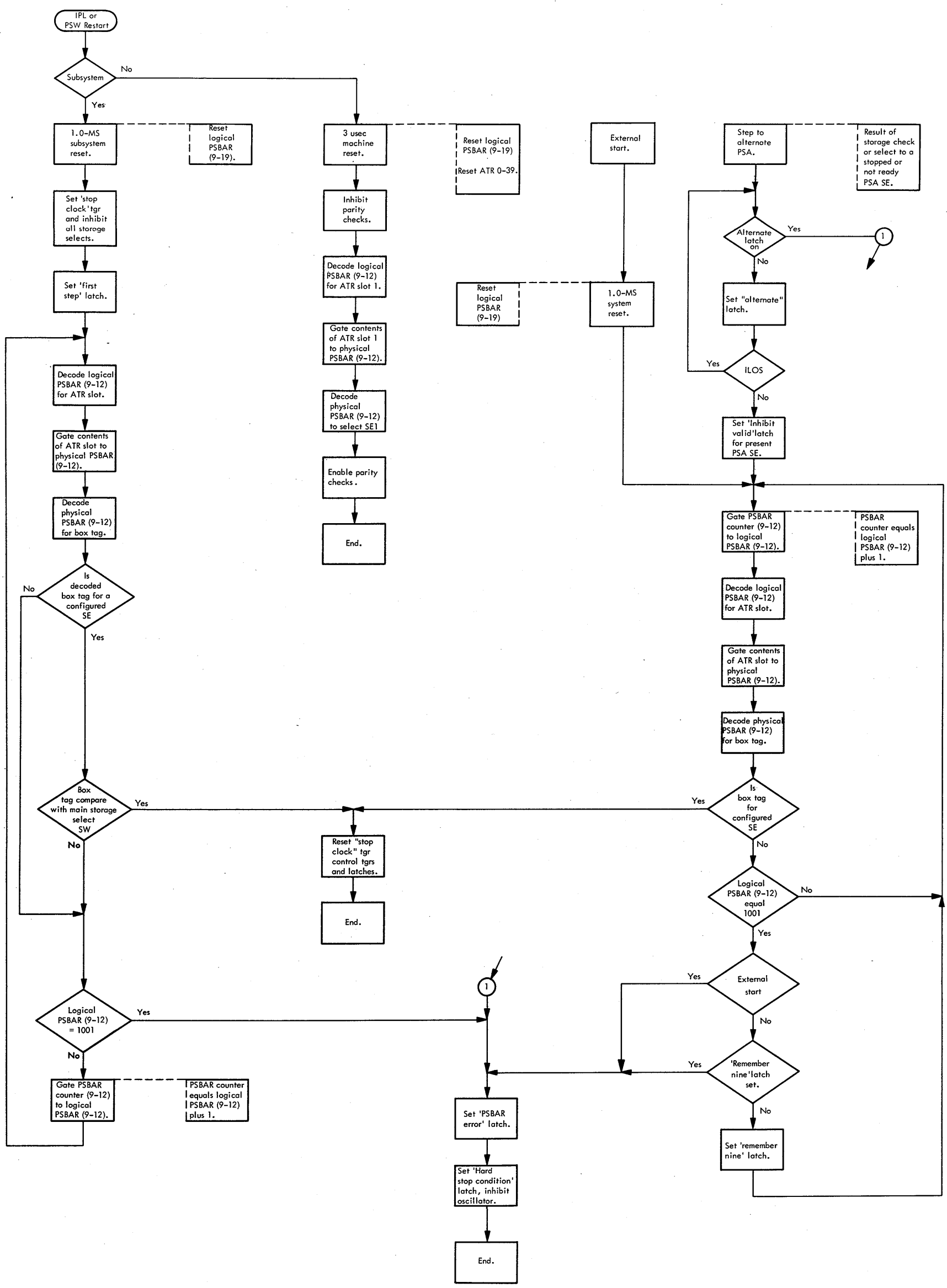
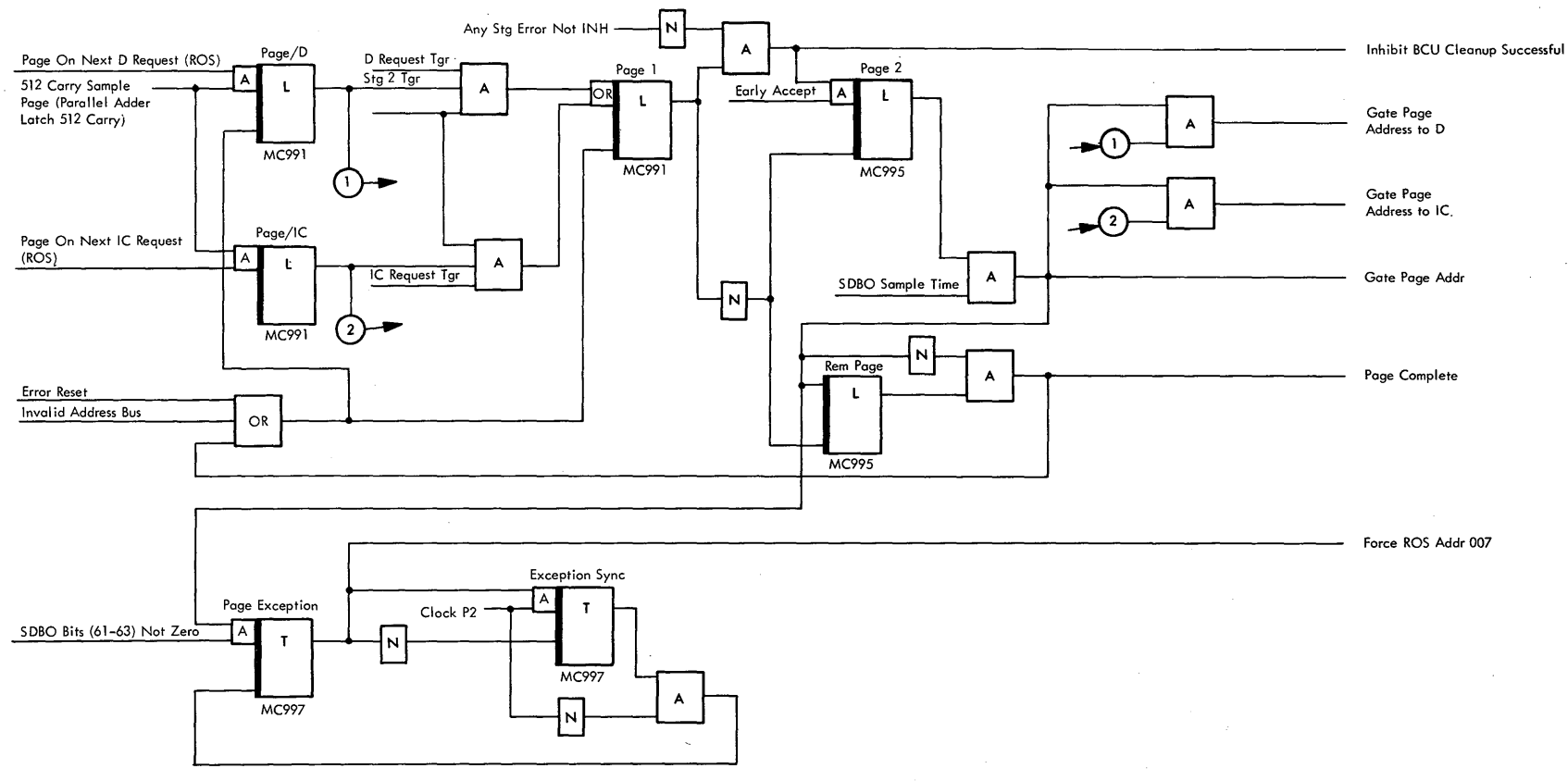


Diagram 4-608. PSBAR Operations



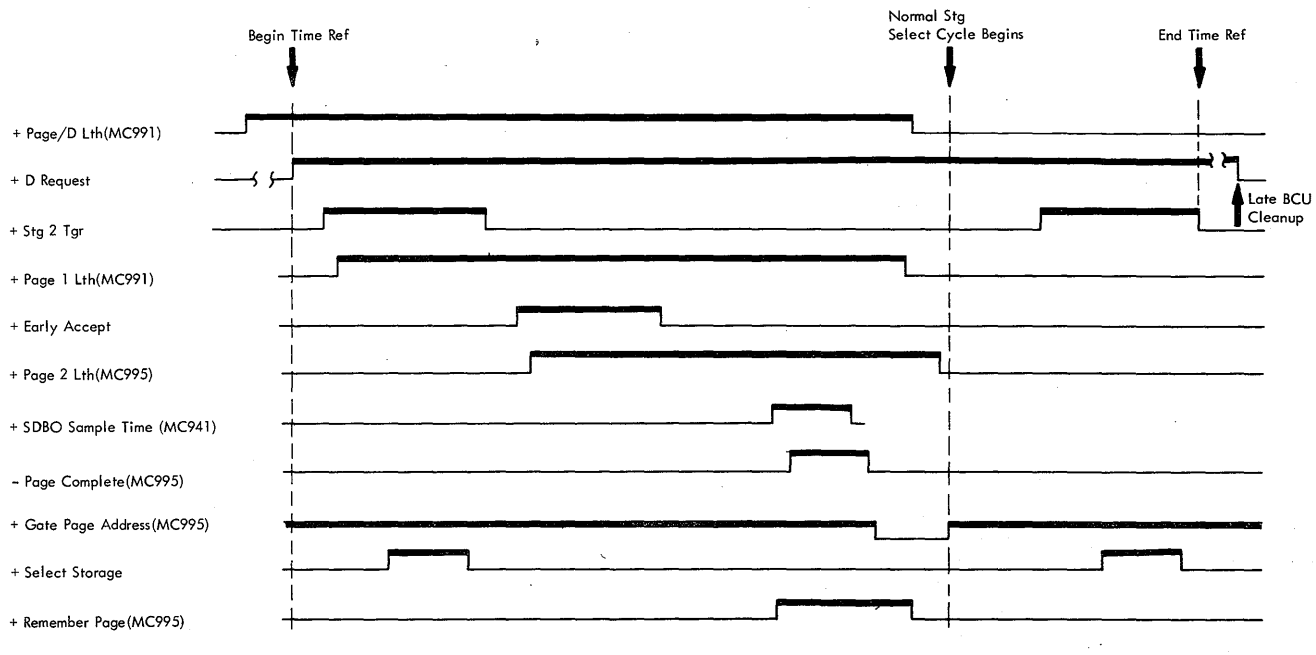


Diagram 4-609. Page Control Logic and Timing

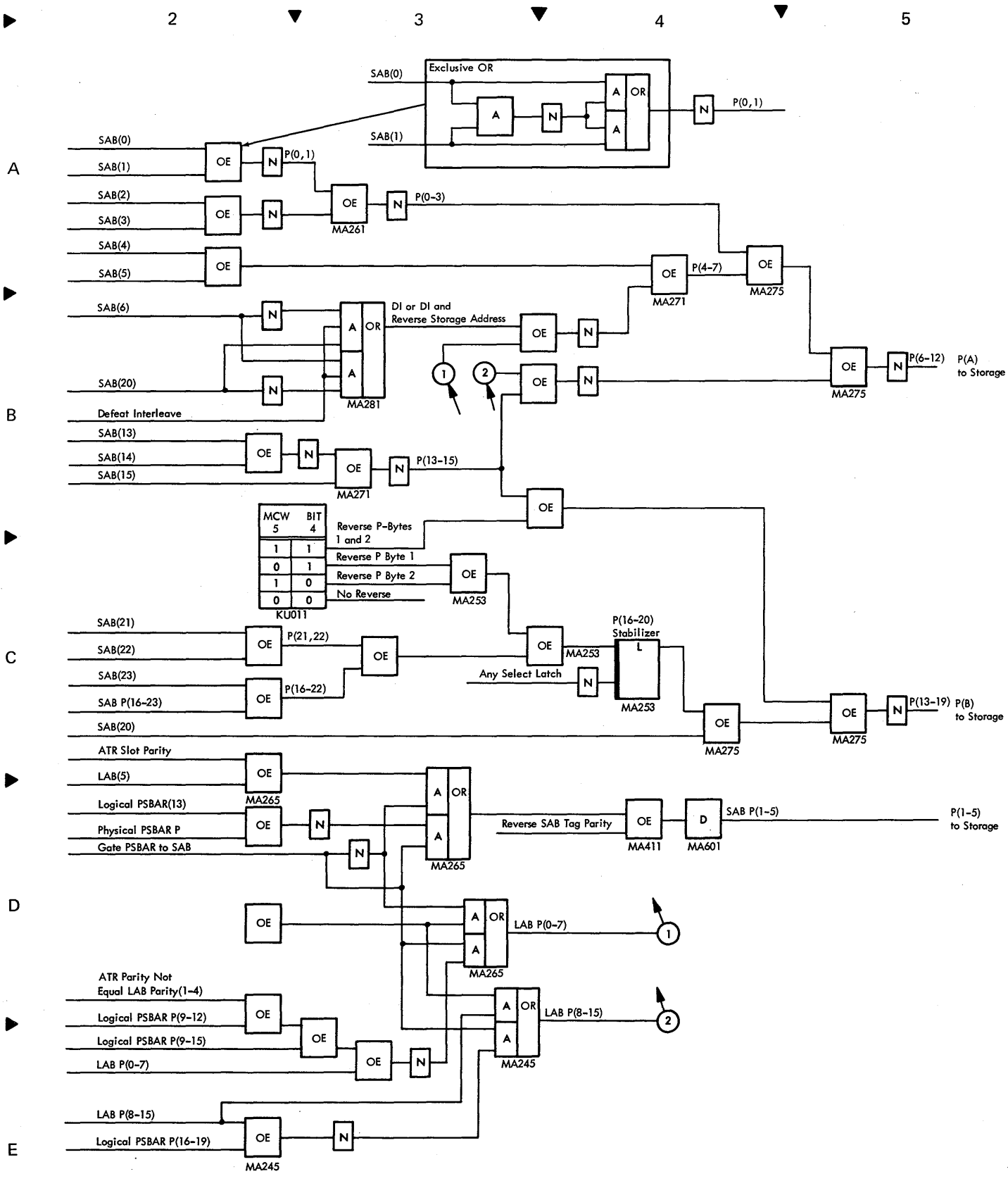


Diagram 4-610. SAB Parity Conversion Logic

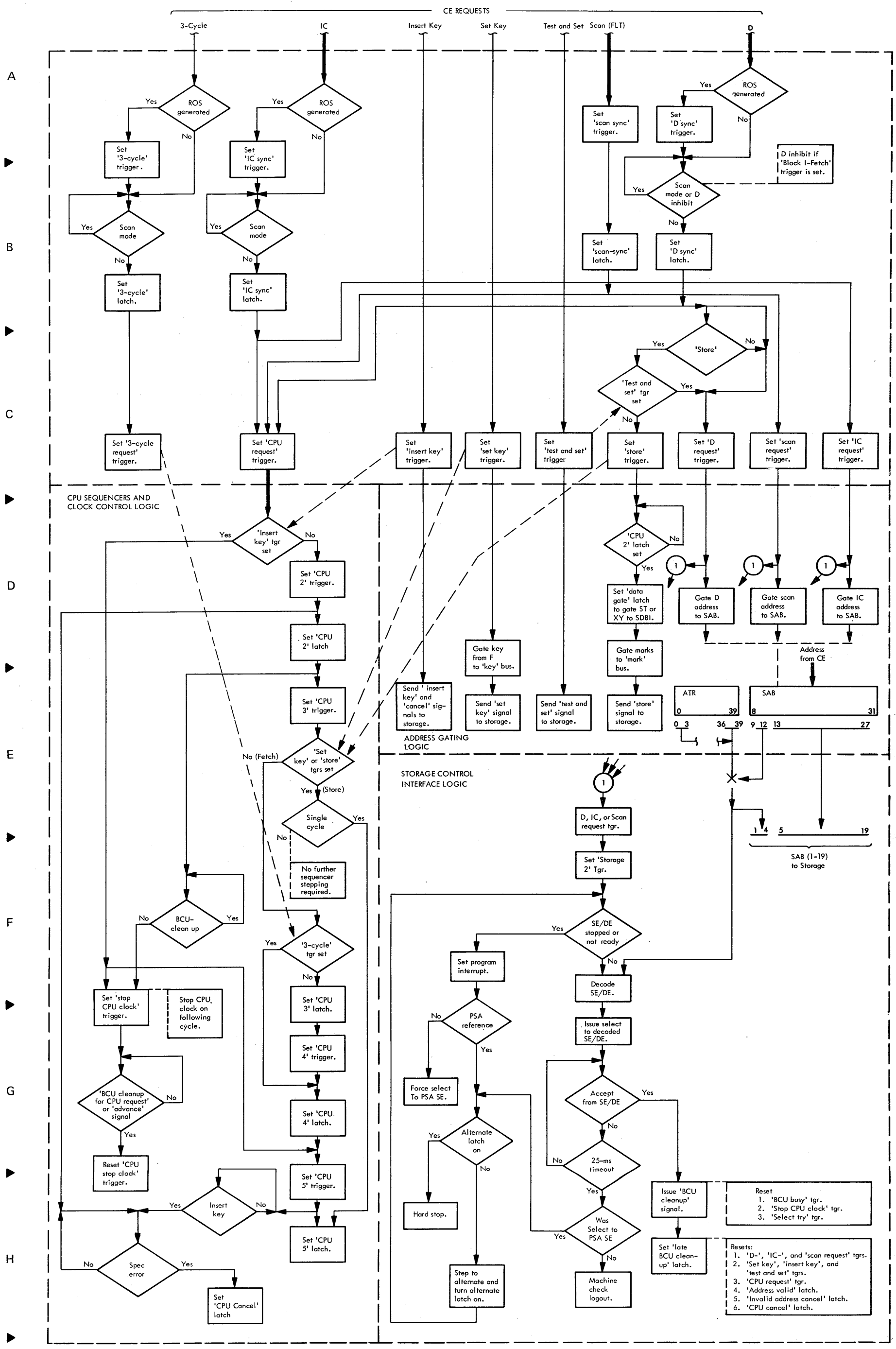


Diagram 4-611. Detailed SCI Functional Sequence (Sheet 1 of 2)

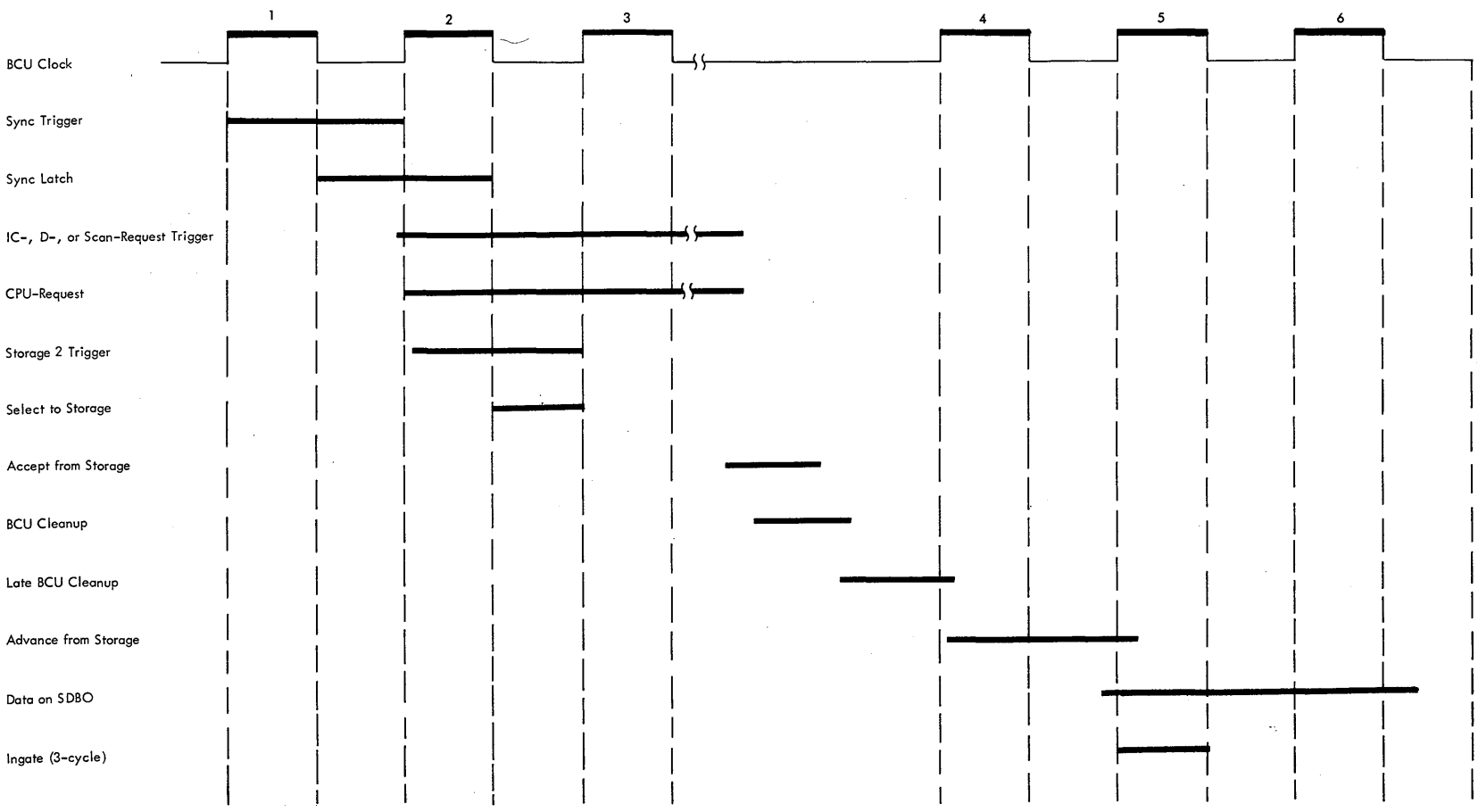


Diagram 4-611. Servicing of Storage Requests in Single-Cycle Mode (Sheet 2 of 2)

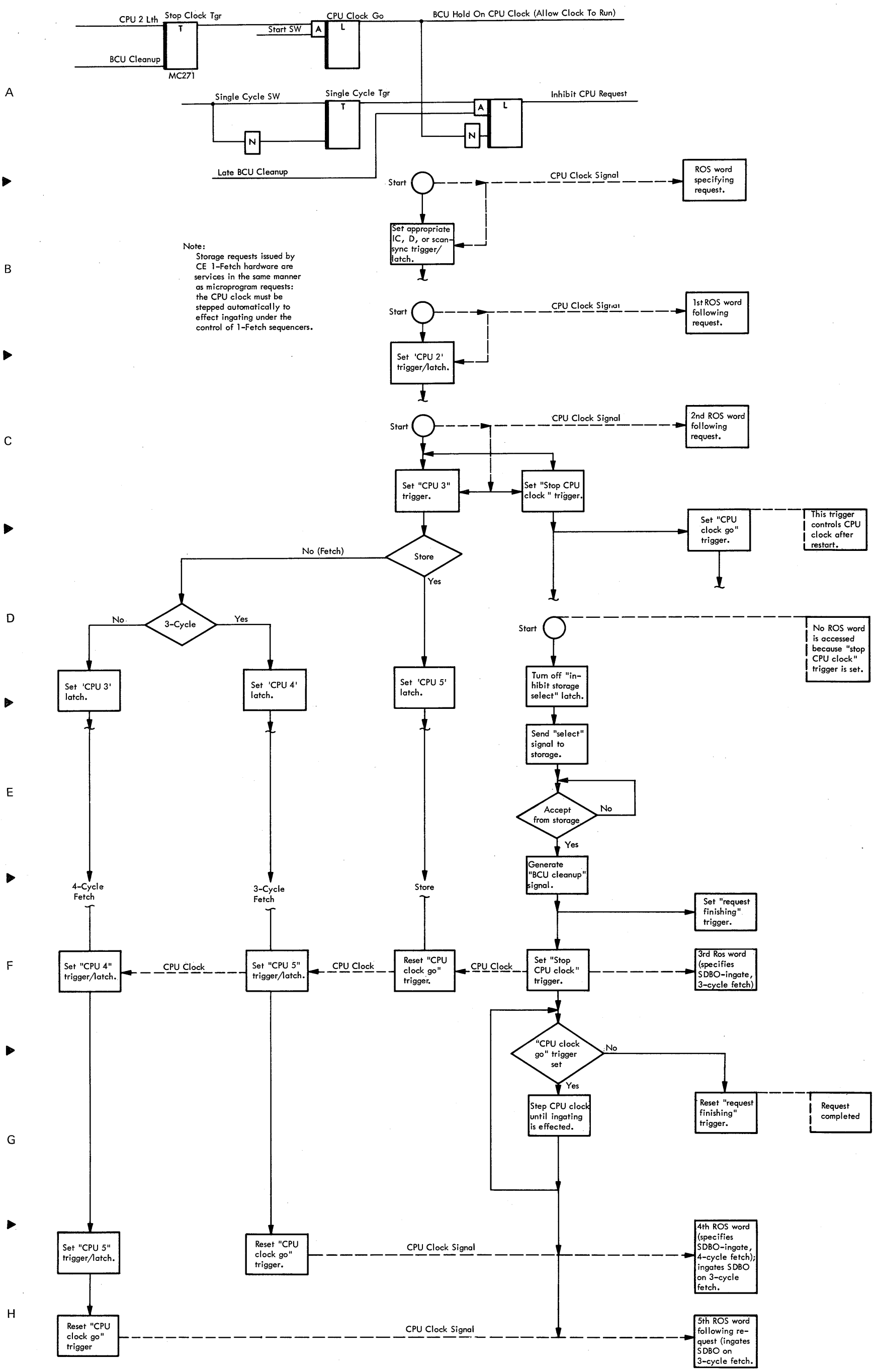


Diagram 4-612. Servicing of Storage Requests in Single-Cycle Mode

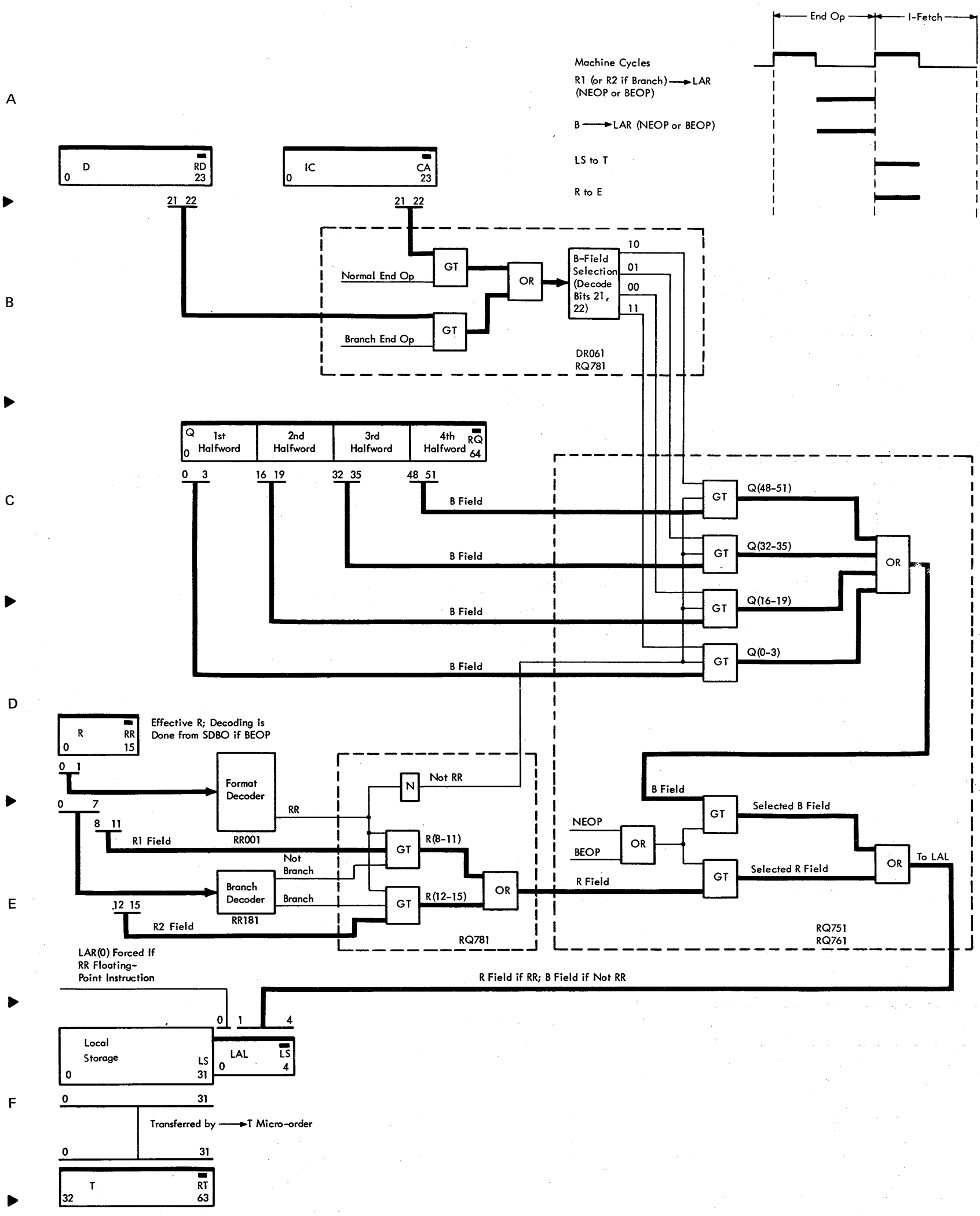


Diagram 5-1. Operand Prefetching During End Op

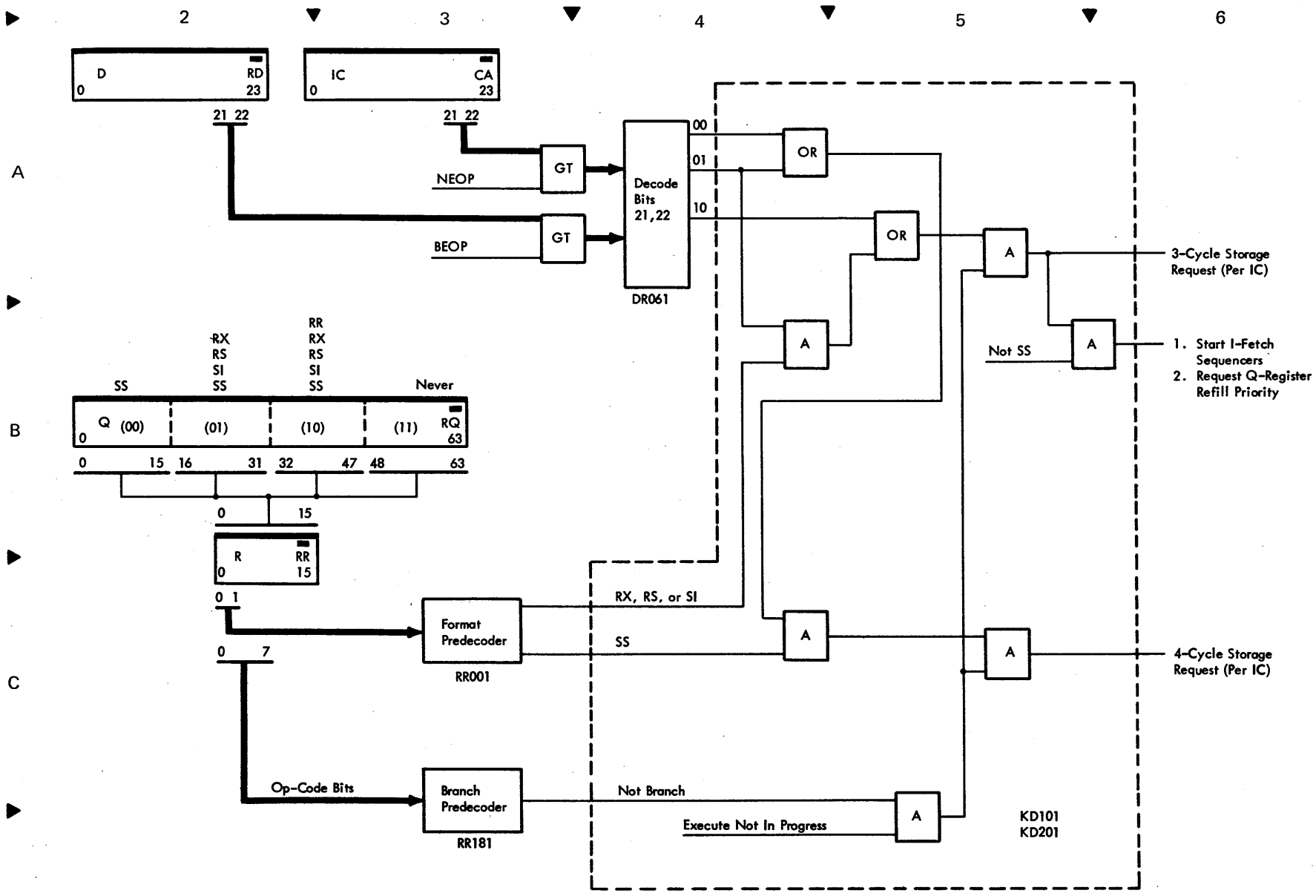


Diagram 5-2. Instruction Requests During End Op

DIAGRAM 5-2. INSTRUCTION REQUESTS DURING END OP

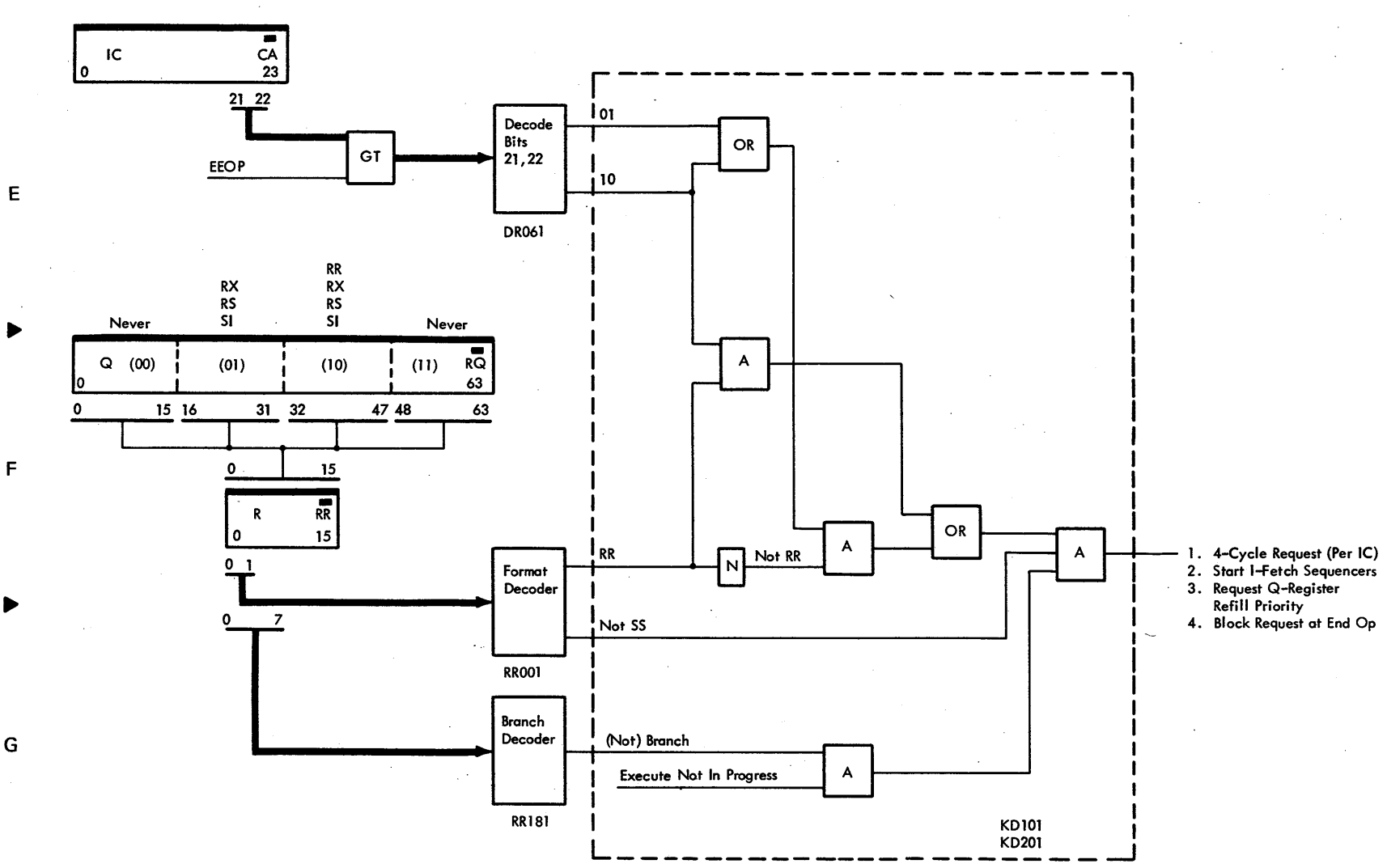


Diagram 5-3. Instruction Requests During Early End Op

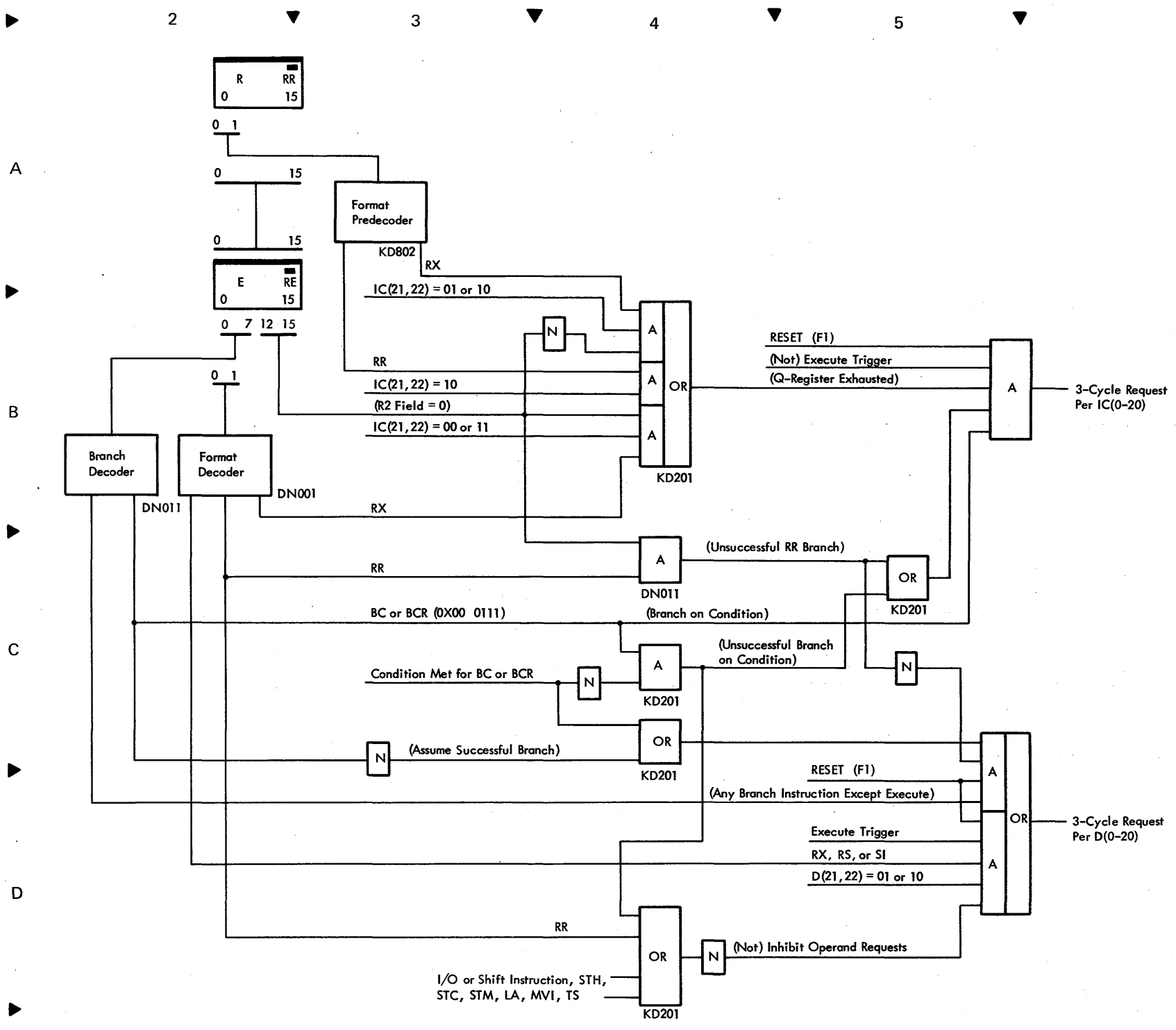


Diagram 5-4. Branch Requests

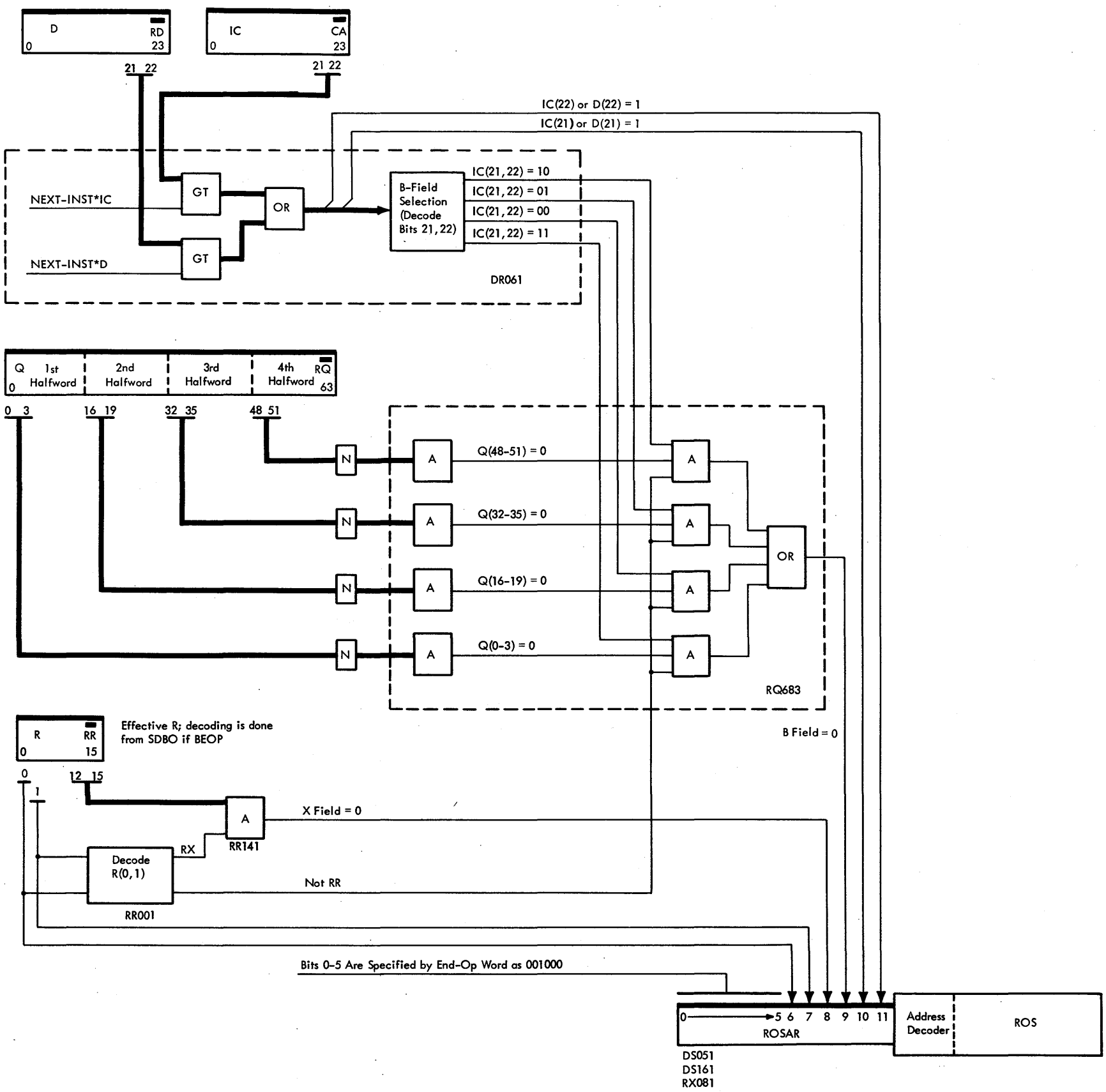
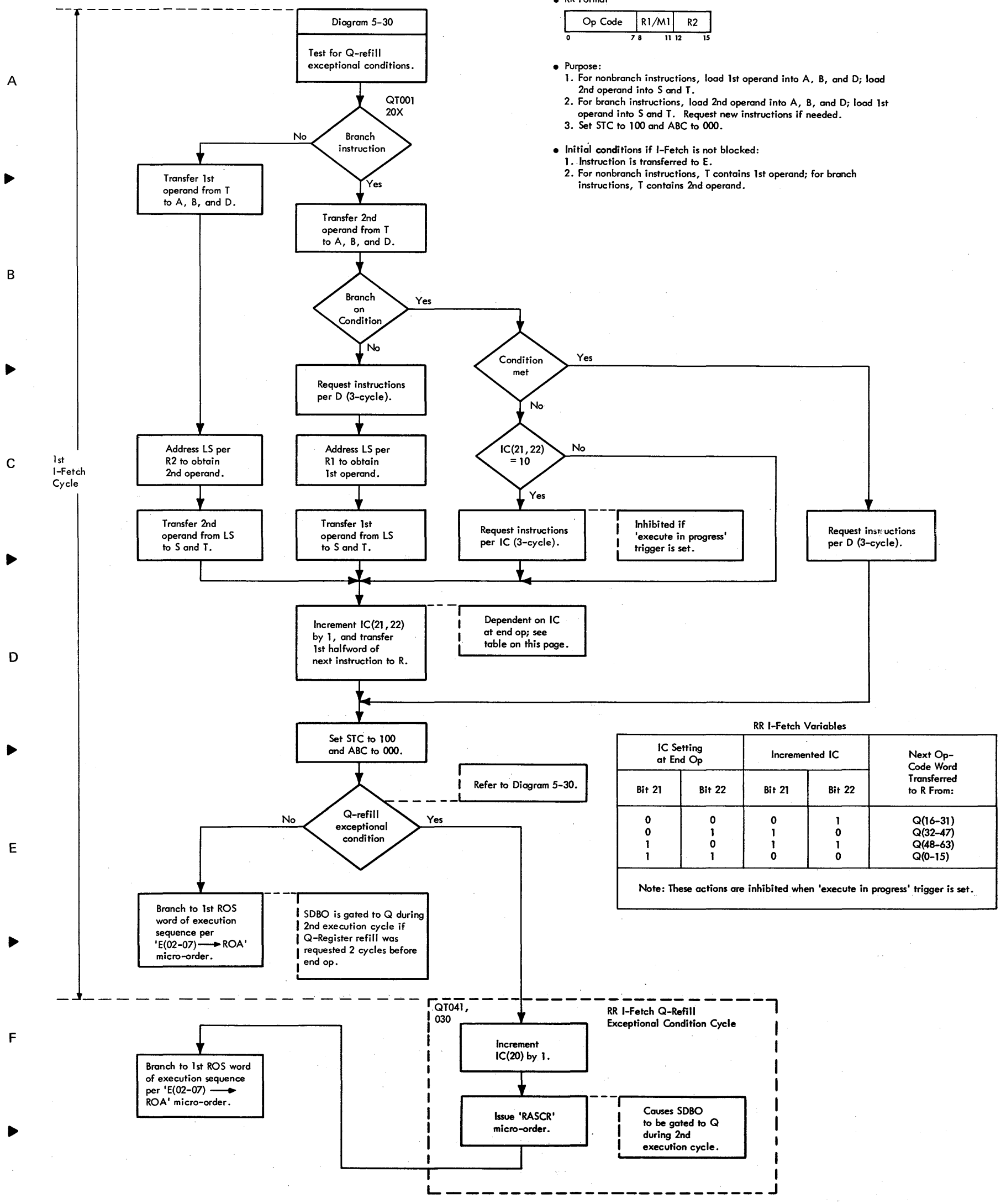


Diagram 5-5. Selection of I-Fetch Sequence



- RR Format

Op Code	R1/M1	R2
0 7	8 11	12 15

- Purpose:
  1. For nonbranch instructions, load 1st operand into A, B, and D; load 2nd operand into S and T.
  2. For branch instructions, load 2nd operand into A, B, and D; load 1st operand into S and T. Request new instructions if needed.
  3. Set STC to 100 and ABC to 000.

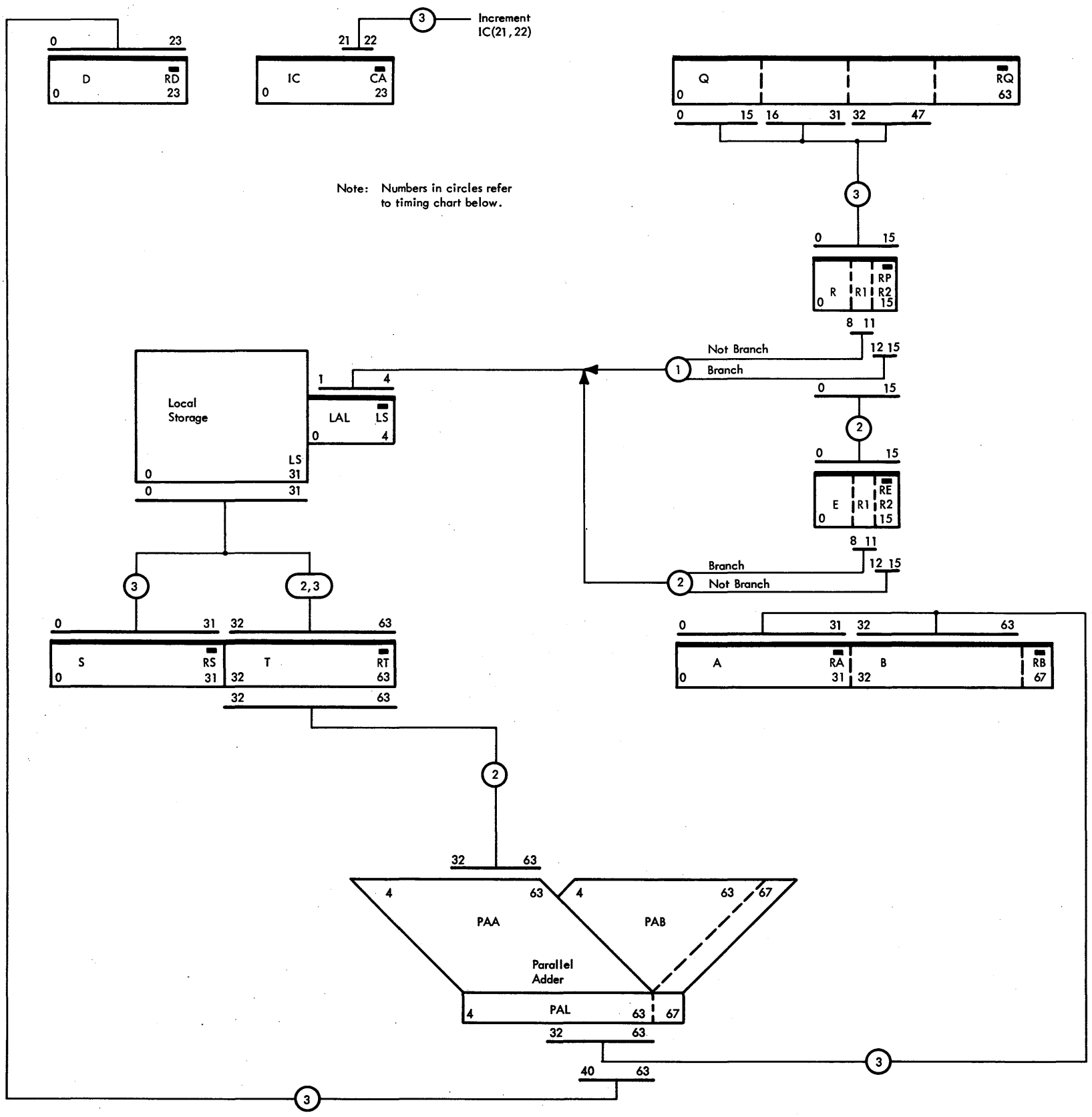
- Initial conditions if I-Fetch is not blocked:
  1. Instruction is transferred to E.
  2. For nonbranch instructions, T contains 1st operand; for branch instructions, T contains 2nd operand.

RR I-Fetch Variables

IC Setting at End Op		Incremented IC		Next Op-Code Word Transferred to R From:
Bit 21	Bit 22	Bit 21	Bit 22	
0	0	0	1	Q(16-31)
0	1	1	0	Q(32-47)
1	0	1	1	Q(48-63)
1	1	0	0	Q(0-15)

Note: These actions are inhibited when 'execute in progress' trigger is set.

Diagram 5-6. RR I-Fetch



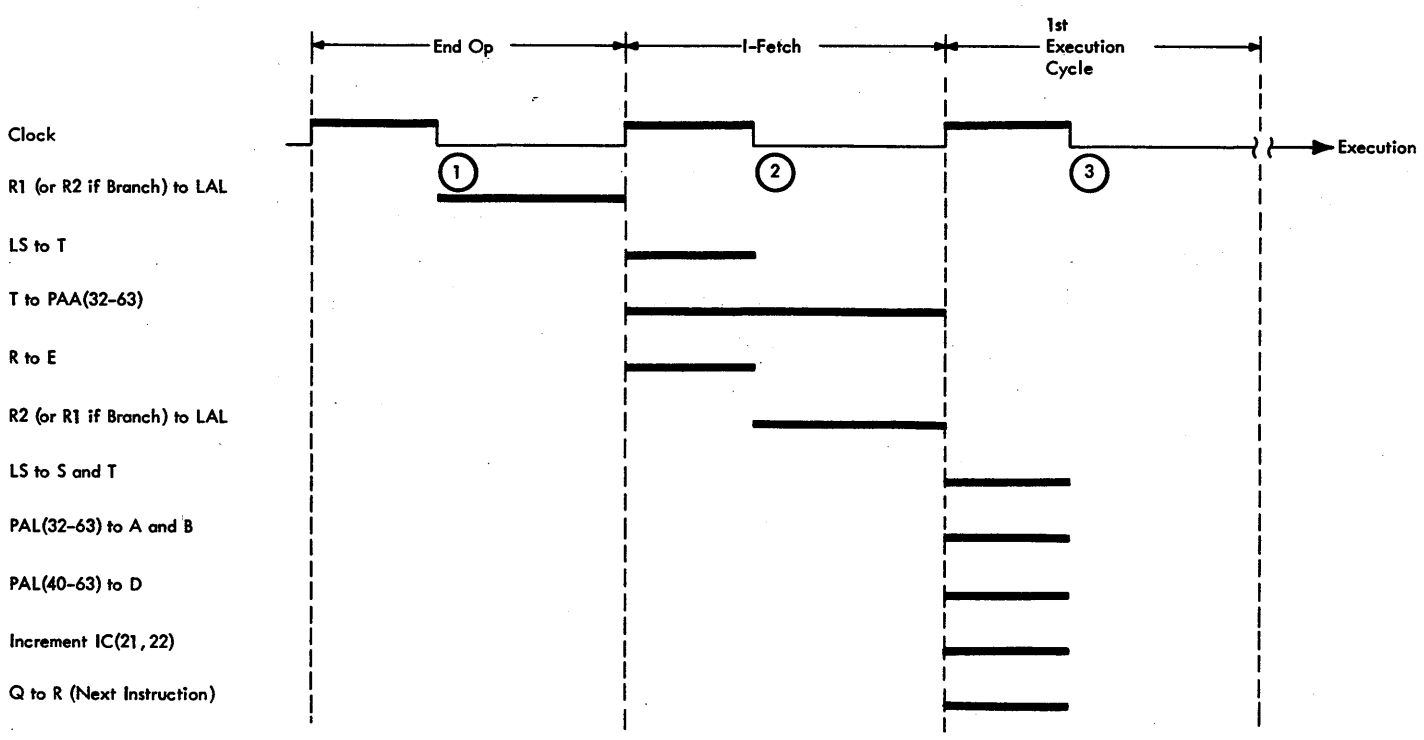


Diagram 5-7. One-Cycle RR I-Fetch

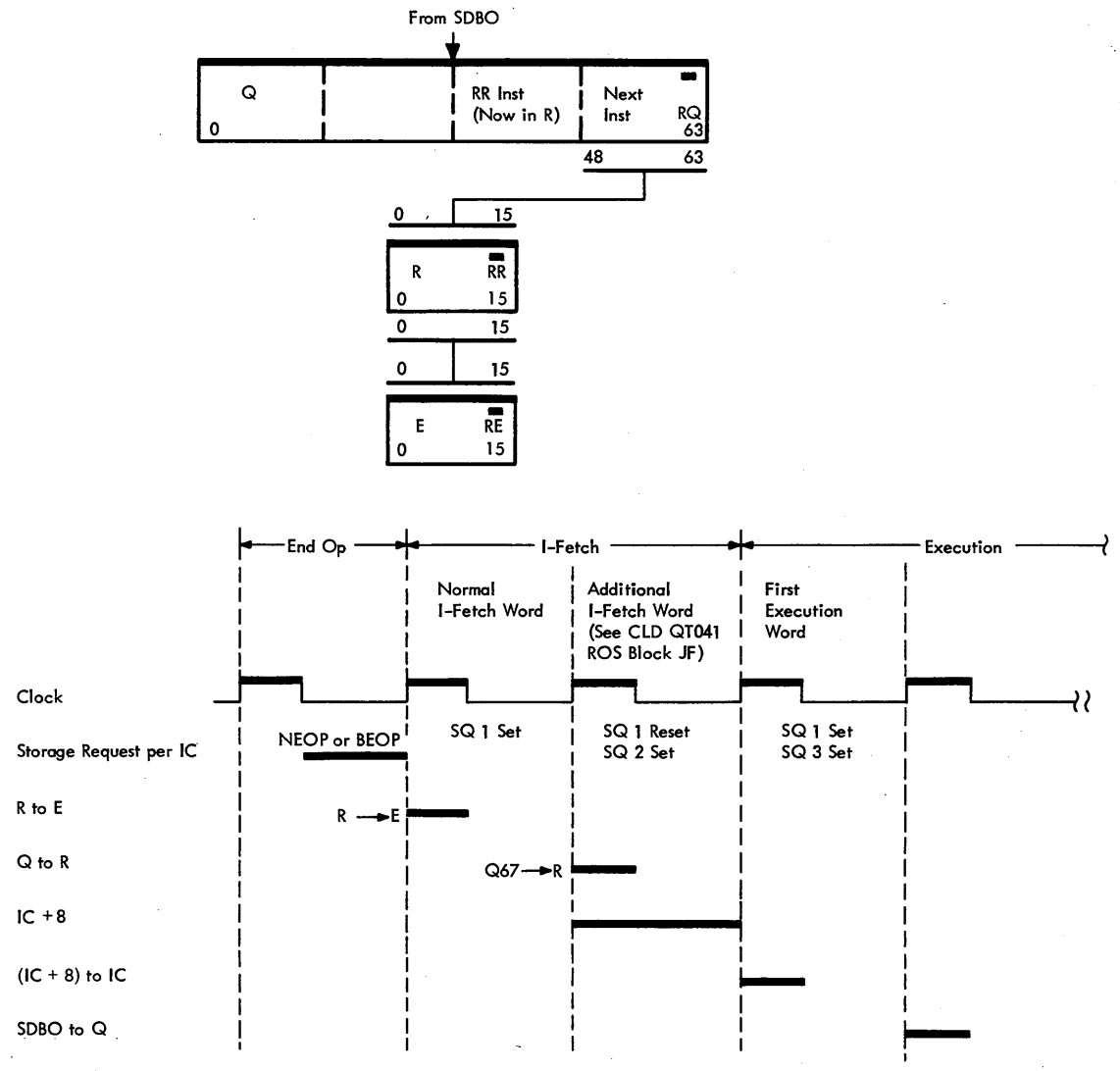
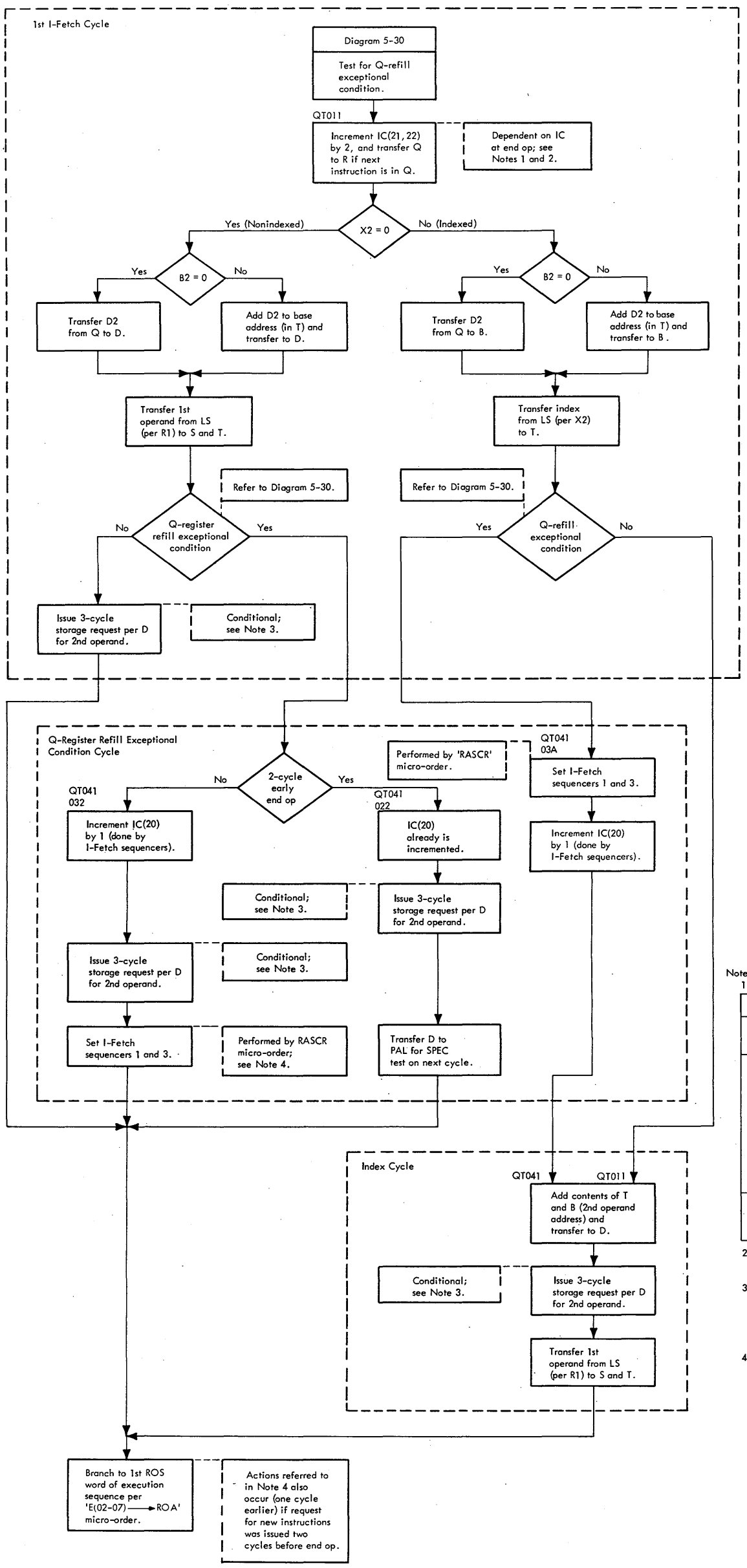
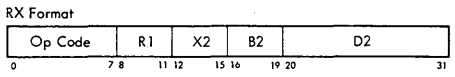


Diagram 5-8. Two-Cycle RR I-Fetch



- RX Format

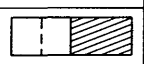
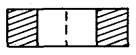
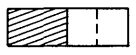
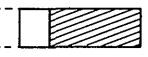


- Purpose:
  1. Transfer 1st operand to S and T.
  2. Compute address of 2nd operand and transfer to D.
  3. Request 2nd operand from main storage (see Note 3).

- Initial conditions if I-Fetch is not blocked:
  1. 1st halfword of instruction is transferred to E; 2nd halfword is in Q.
  2. Contents of LS register specified by B2 are transferred to T.

Notes:

1. IC Incrementing and Q to R Transfer:

Conditions at End Op			Incremented IC(21,22)	Next Op-Code Word Transferred to R From:
IC(21,22)		Position in Q		
00			10	Q(32-47)
01	†		11	Q(48-63)
10	†		00	Not Yet in Q
11			01	Q(16-31)

† Q-register refill is requested by end-op word; Q-register refill priority does not result if RX is indexed and request is issued two cycles early.

2. Incrementing of IC, transfer of Q to R, and the refilling of Q are inhibited if 'execute in progress' trigger is set.

3. 'RESET' micro-order:
   a. Inhibits setting of 'D sync' latch if LA, STC, STH, or unsuccessful BC.
   b. Sets 'IC sync' latch if unsuccessful BC.
   c. Resets STAT's, Edit controls, STC, and ABC.
   d. Sets LAR(0) if floating-point instruction.

4. I-Fetch sequencers 1 and 3 gate SDBO to Q during 2nd execution cycle, and transfer Q(0-15) to R if IC(21, 22) = 10 during previous end op.

Diagram 5-9. RX I-Fetch

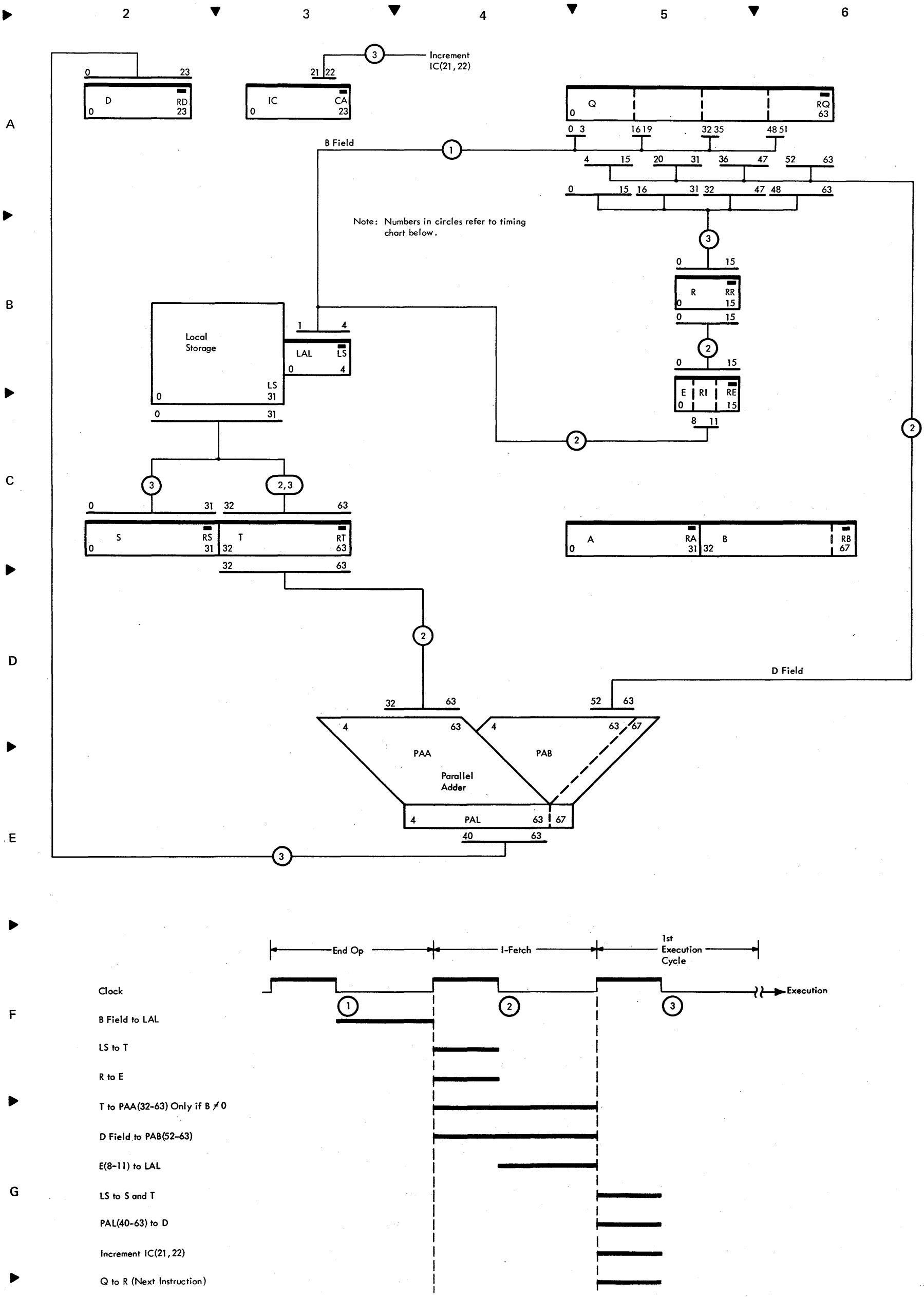


Diagram 5-10. One-Cycle RX, RS, and SI I-Fetch

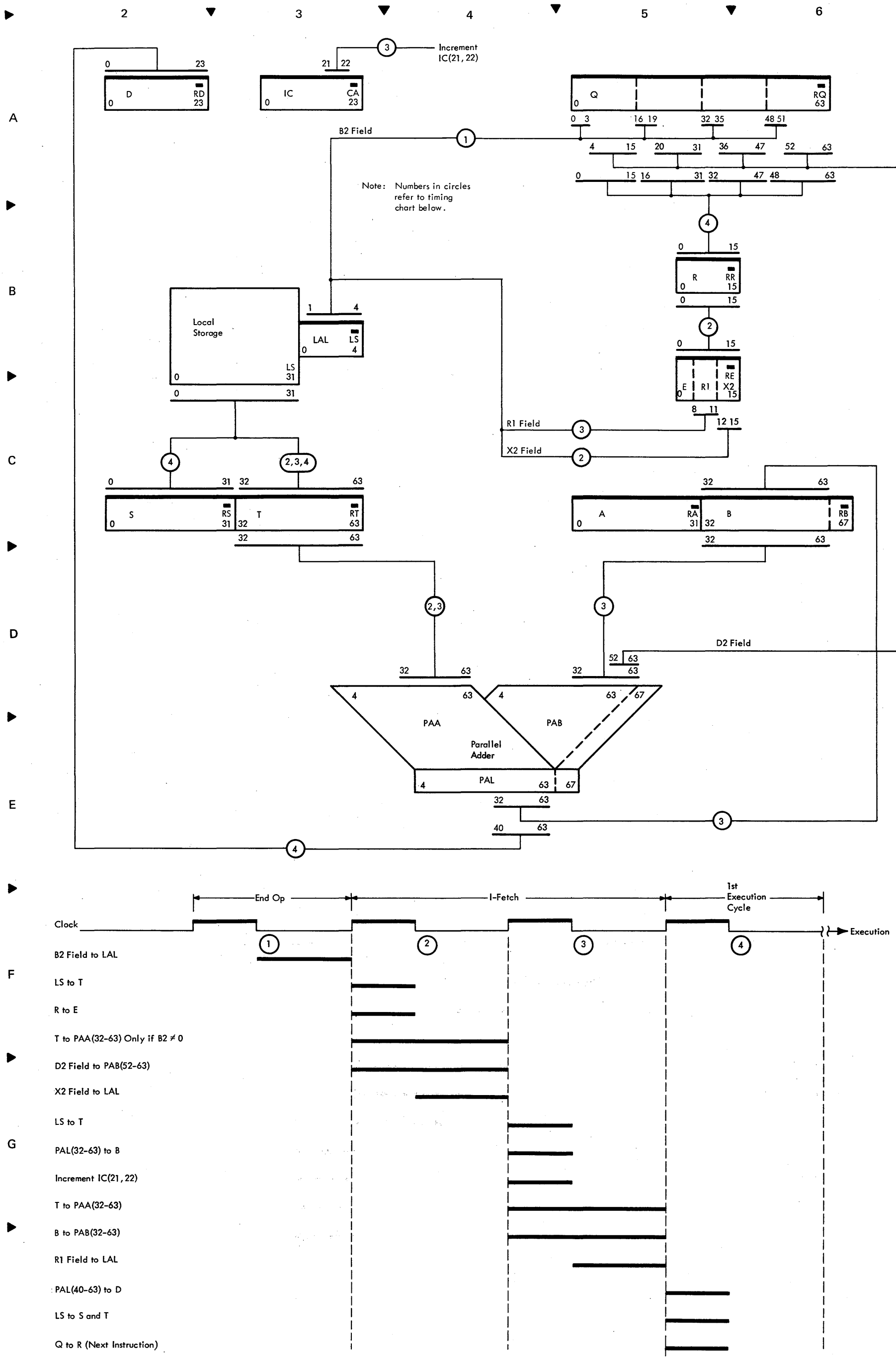


Diagram 5-11. Two-Cycle Indexed RX I-Fetch

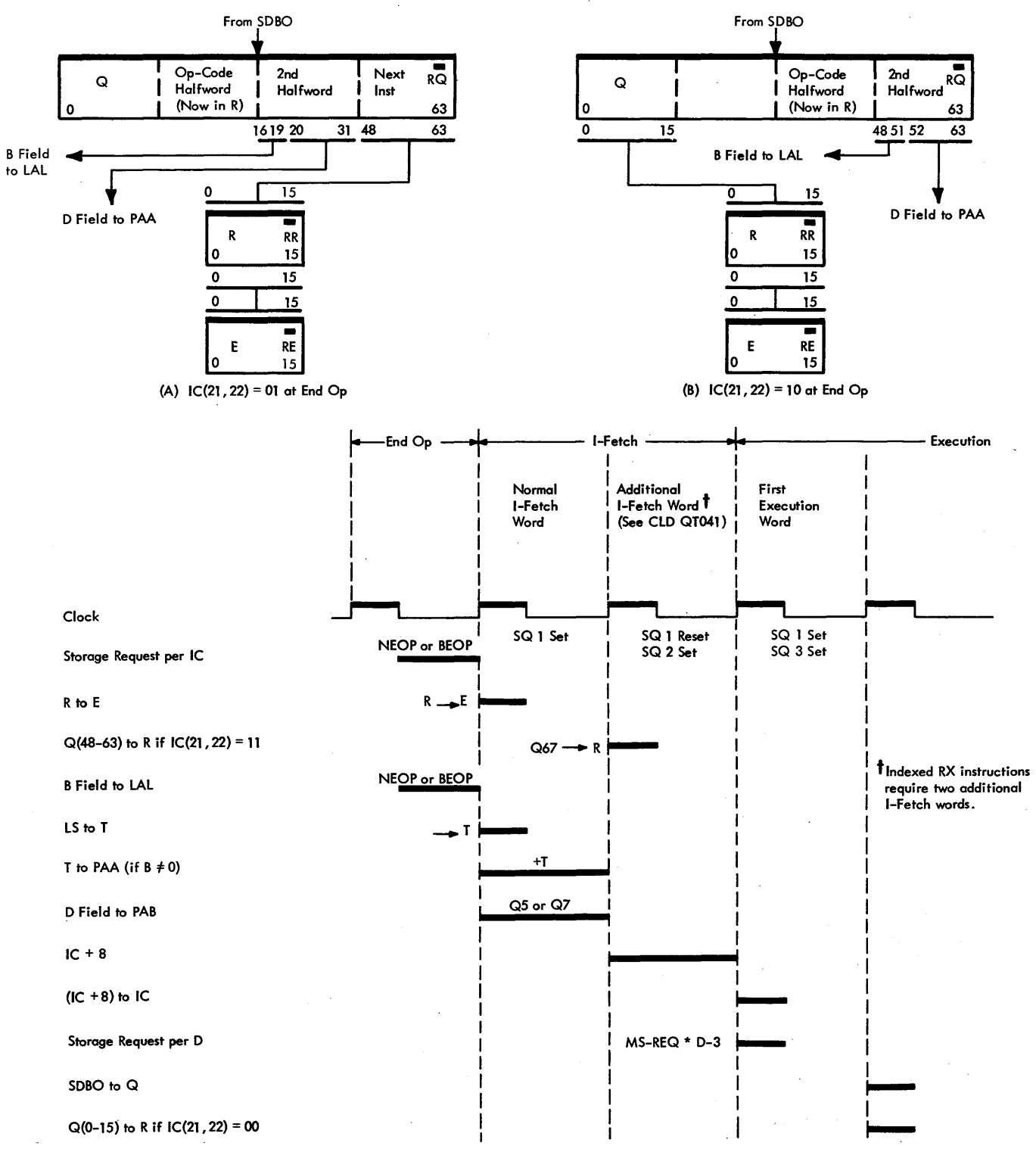


Diagram 5-12. Two-Cycle Non-Indexed RX, RS, and SI I-Fetch

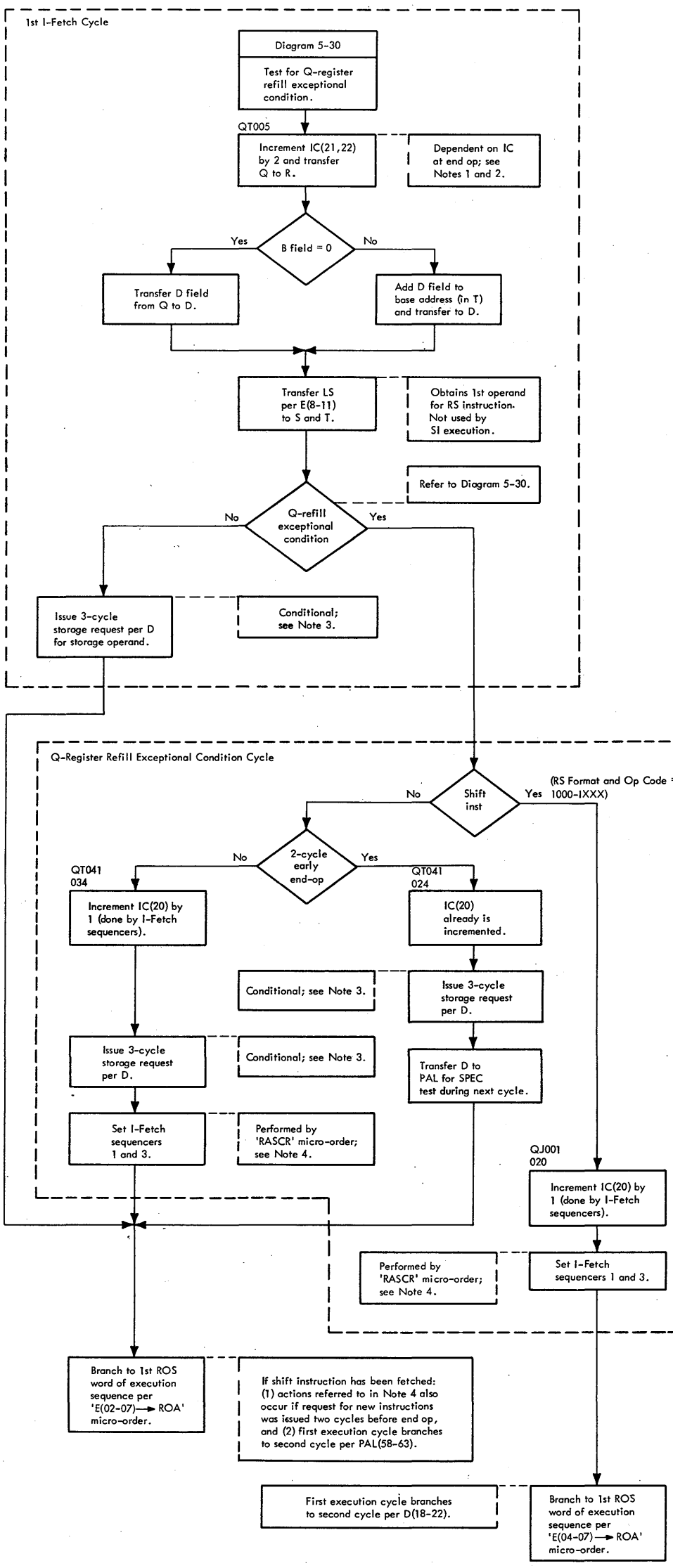
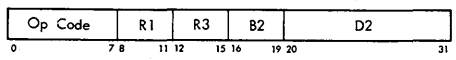
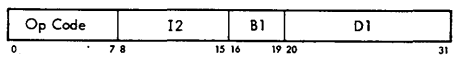


Diagram 5-13. RS and SI I-Fetch

- RS Format:



- SI Format:



- Purpose:
  1. Transfer contents of LS register specified by E(8-11) to S and T. (This is 1st operand for RS instructions only.)
  2. Compute address of storage operand and transfer to D. (For RS instructions, this is address of 2nd operand.)
  3. Request operand from main storage (see Note 3).

- Initial conditions if I-Fetch is not blocked:
  1. First halfword of instruction is transferred to E; 2nd halfword is in Q.
  2. Contents of LS register specified by B field are transferred to T.

Notes : 1. IC Incrementing and Q to R Transfer:

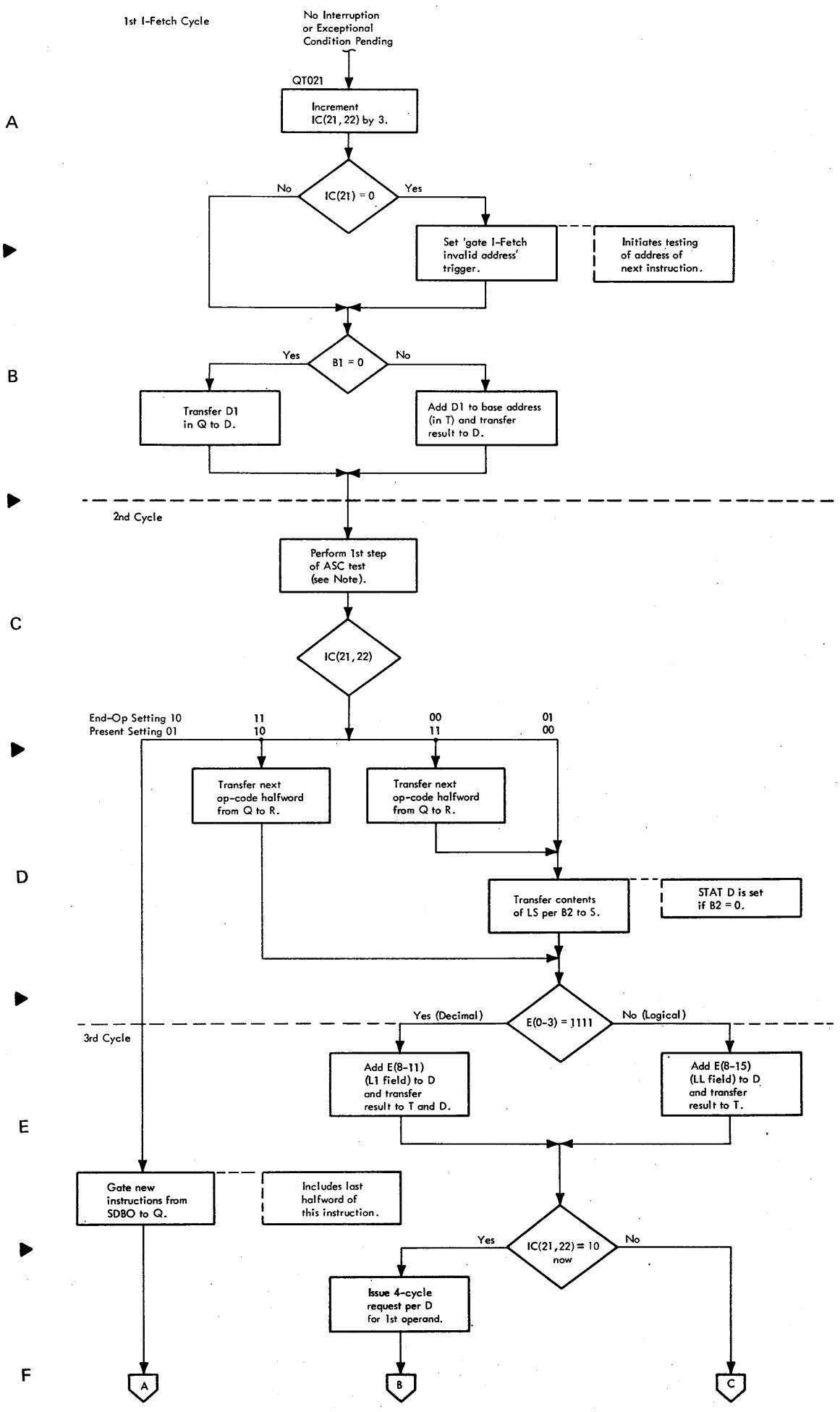
Conditions at End Op		Incremented IC(21,22)	Next Op-Code Word Transferred to R From:
IC(21,22)	Position in Q		
00		10	Q(32-47)
01	†	11	Q(48-63)
10	†	00	Not Yet in Q
11		01	Q(16-31)

†Q-register refill requested by end-op word; Q-register refill priority does not result if shift instruction is being fetched and request is issued two cycles early.

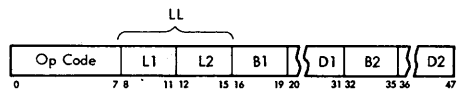
2. Incrementing of IC, transferring of Q to R, and setting of I-Fetch sequencers are inhibited if 'execute in progress' trigger is set.

3. 'RESET' micro-order:
   a. Inhibits setting of 'D sync' trigger if fetching MVI, STM, TS, shift, or I/O instruction.
   b. Resets '3-cycle request' trigger (causing 4-cycle request) if fetching BXH or BXLE instruction.
   c. Resets STAT's, Edit controls, STC, and ABC.

4. I-Fetch sequencers 1 and 3 gate SDBO to Q during 2nd execution cycle, and transfer Q(0-15) to R if IC(21,22) = 10 during previous end op.

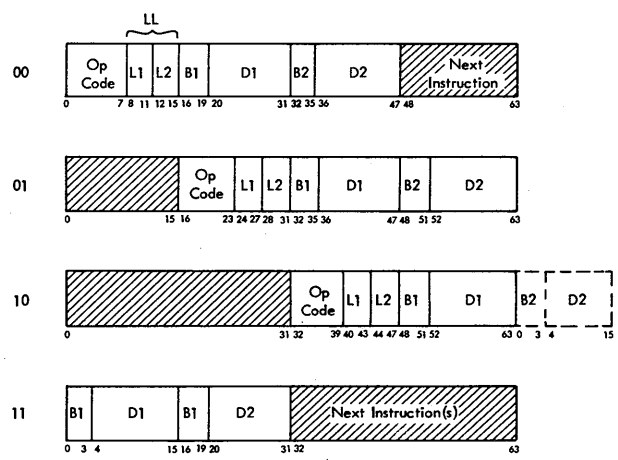


- SS Format:



- Purpose:
  1. Transfer op-code halfword of next instruction to R; update IC and place into LSWR.
  2. Transfer computed address of 1st operand (destination) per instruction class to D; request destination operand from main storage (gated into CPU during 2nd execution cycle):
     a. Lowest destination address for logical instructions = base address (per B1) + D1.
     b. Highest destination address for decimal instructions = base address (per B1) + D1 + L1.
  3. Transfer computed address of 2nd operand (source) to IC and T. Lowest source address = base address (per B2) + D2.
  4. Perform ASC test (and invalid instruction address test if complete instruction is in Q).

- Initial conditions if I-Fetch is not blocked:
  1. Contents of Q per setting of IC(21, 22) during preceding end of cycle:



  2. First halfword of instruction is transferring to E.
  3. Base address (per B1) is in T.
  4. Main storage request for more instructions was generated per IC during preceding end op if 'execute in progress' trigger is not set and IC(21) = 0 (4-cycle) or if IC(21, 22) = 10 (3-cycle). (Branch end op tests D instead of IC.)

Note: Details of 1st step of ASC test:

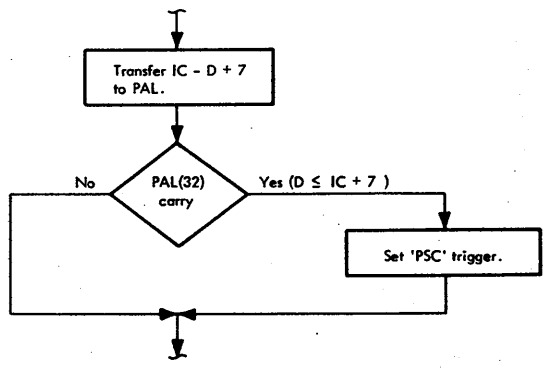


Diagram 5-14. SS I-Fetch (Sheet 1 of 2)

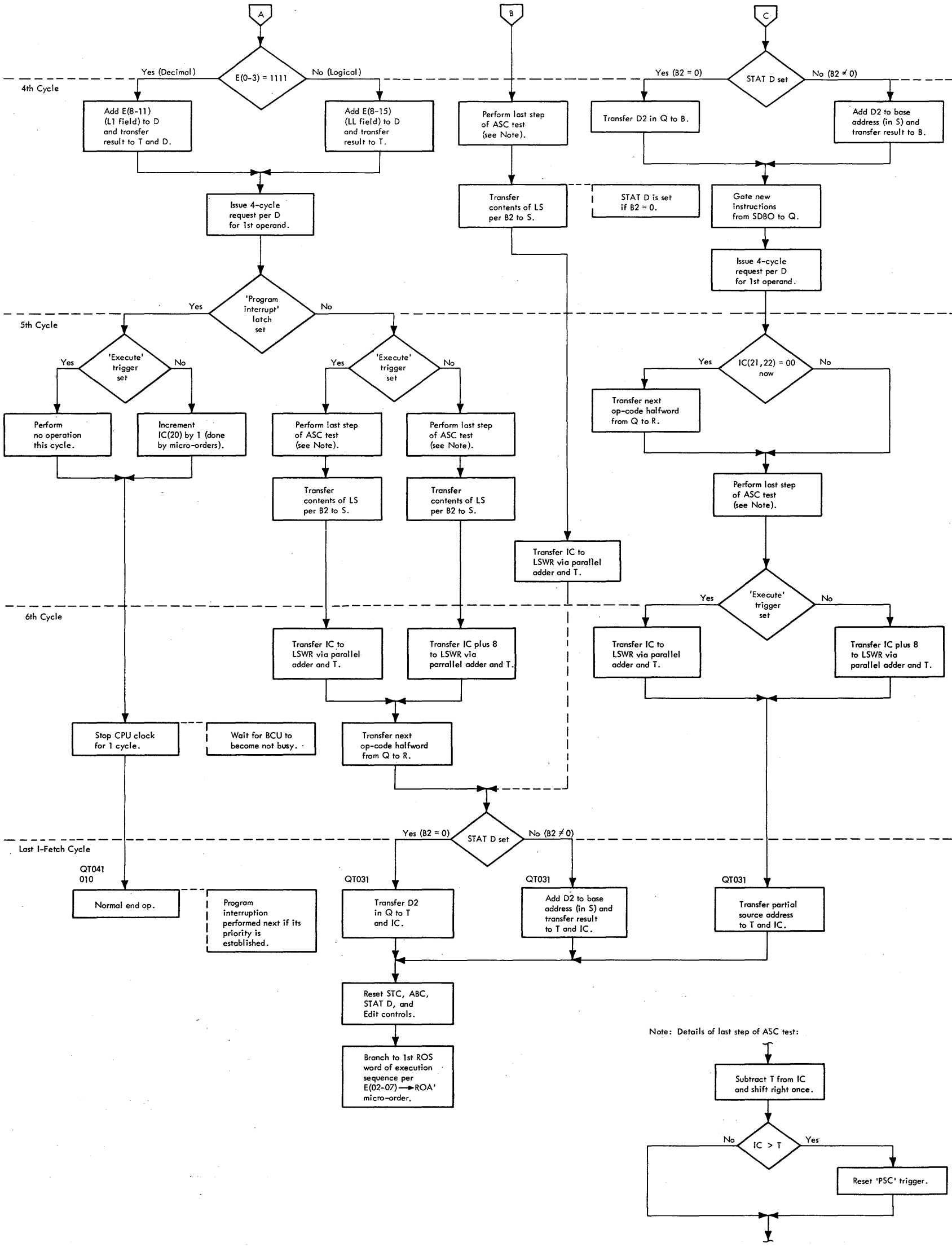
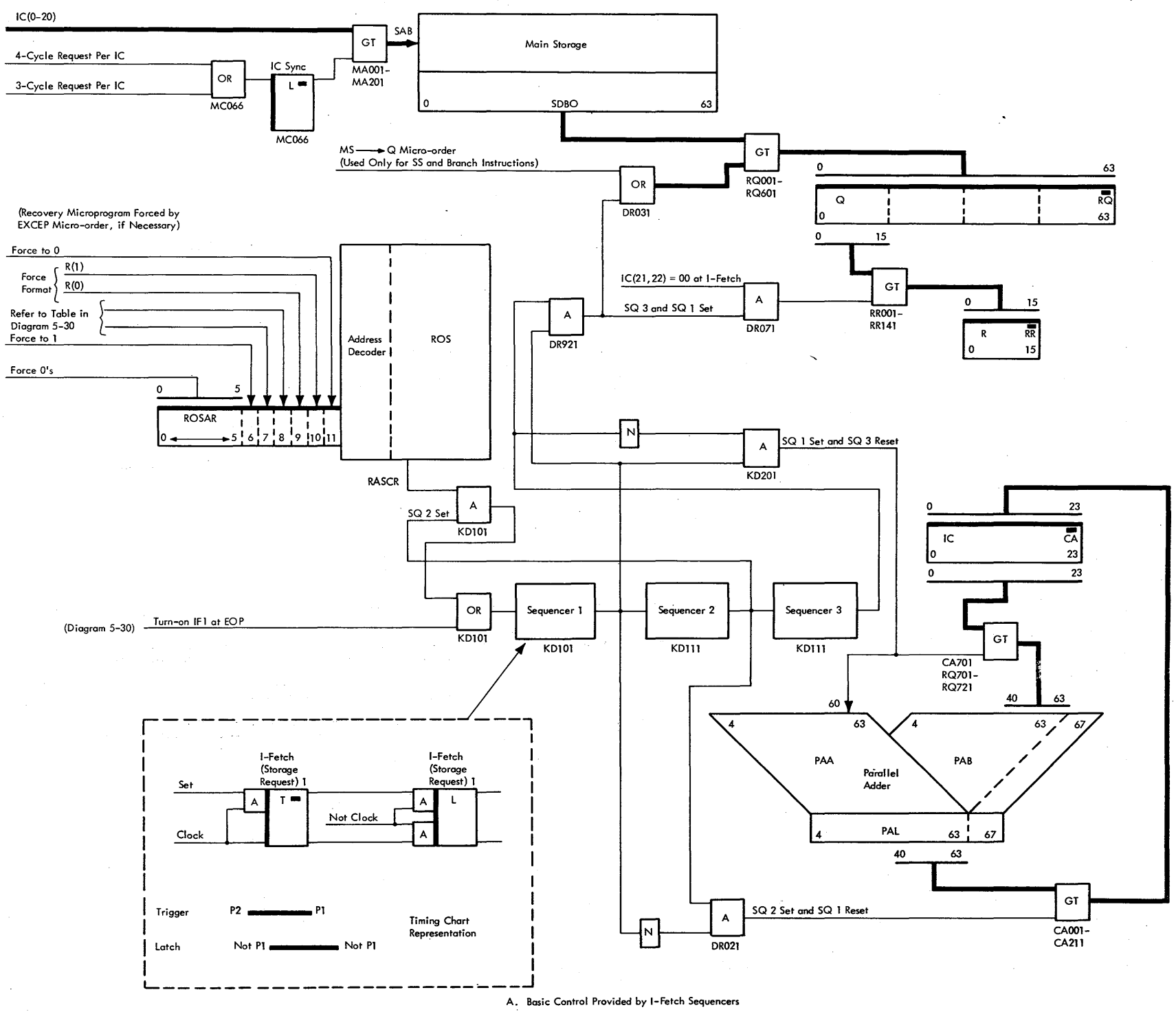
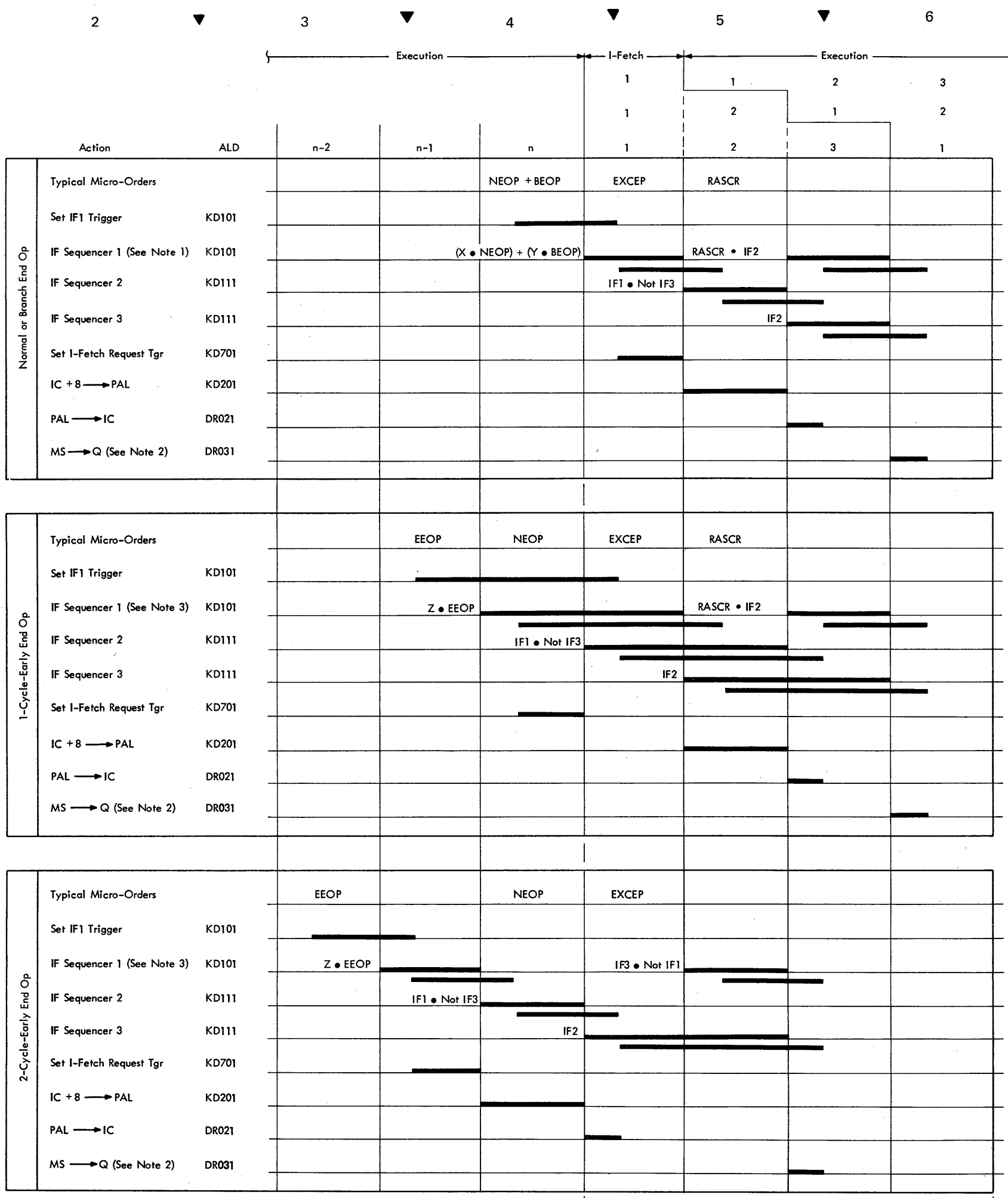


Diagram 5-14. SS I-Fetch (Sheet 2 of 2)



A. Basic Control Provided by I-Fetch Sequencers

Diagram 5-15. I-Fetch Sequences (Sheet 1 of 2)



Notes: 1. X = Z • Not IF2, where Z is defined in Note 3.
Y = [Not (Predecode branch + Predecode SS + Execute)] • [D21,22 = 10 + (D21,22 = 01 • Predecode not RR)].

2. Also, gate Q(0-15) to R(0-15) if IC(21,22) = 00 (ALD DR071).

3. Z = [Not (Predecode branch + Predecode SS + Execute)] • [IC21,22 = 10 + (IC 21,22 = 01 • Predecode not RR).

B. I-Fetch Sequencer Timing Chart

Diagram 5-15. I-Fetch Sequencers (Sheet 2 of 2)

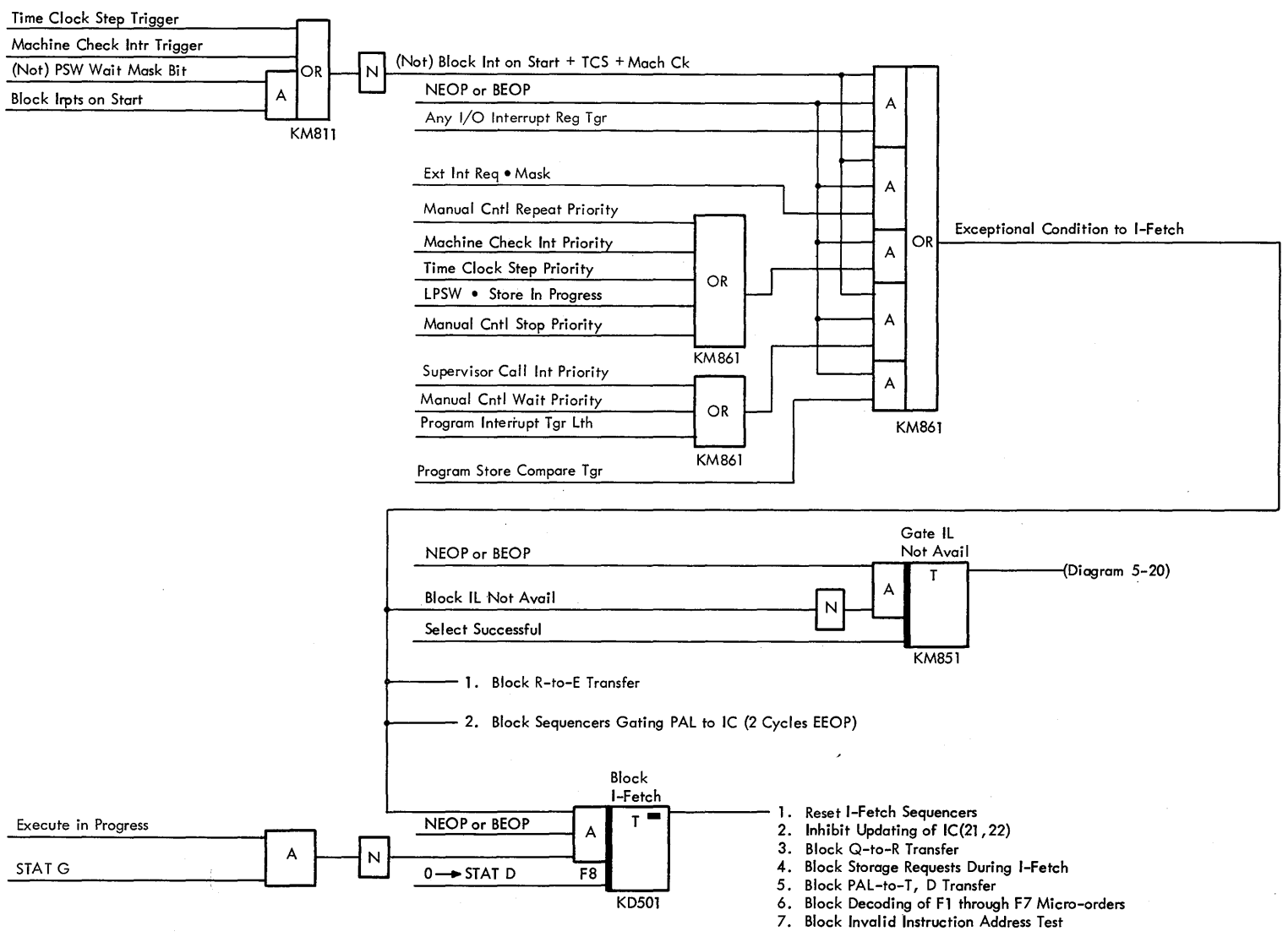
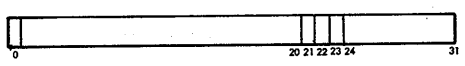
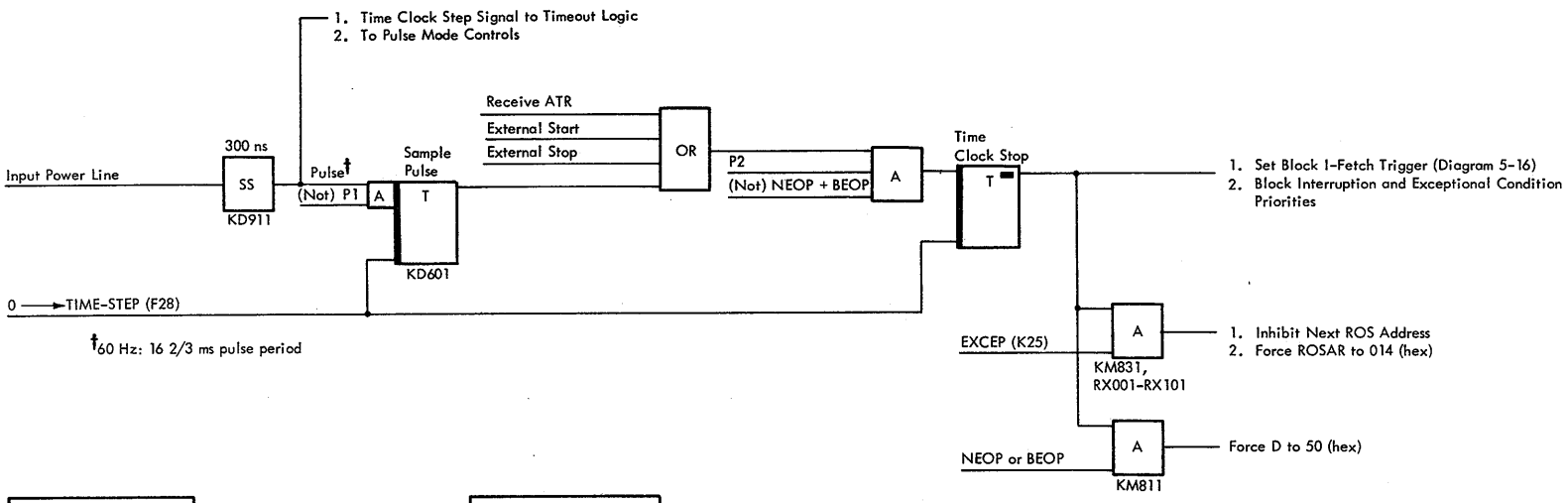


Diagram 5-16. Block I-Fetch Trigger

• Timer format, location 50 (hex):



Bits 21 and 23 are decremented with 60 Hz line frequency.
Bit 0 (sign)=0 if positive; 1 if negative (2's complement form)



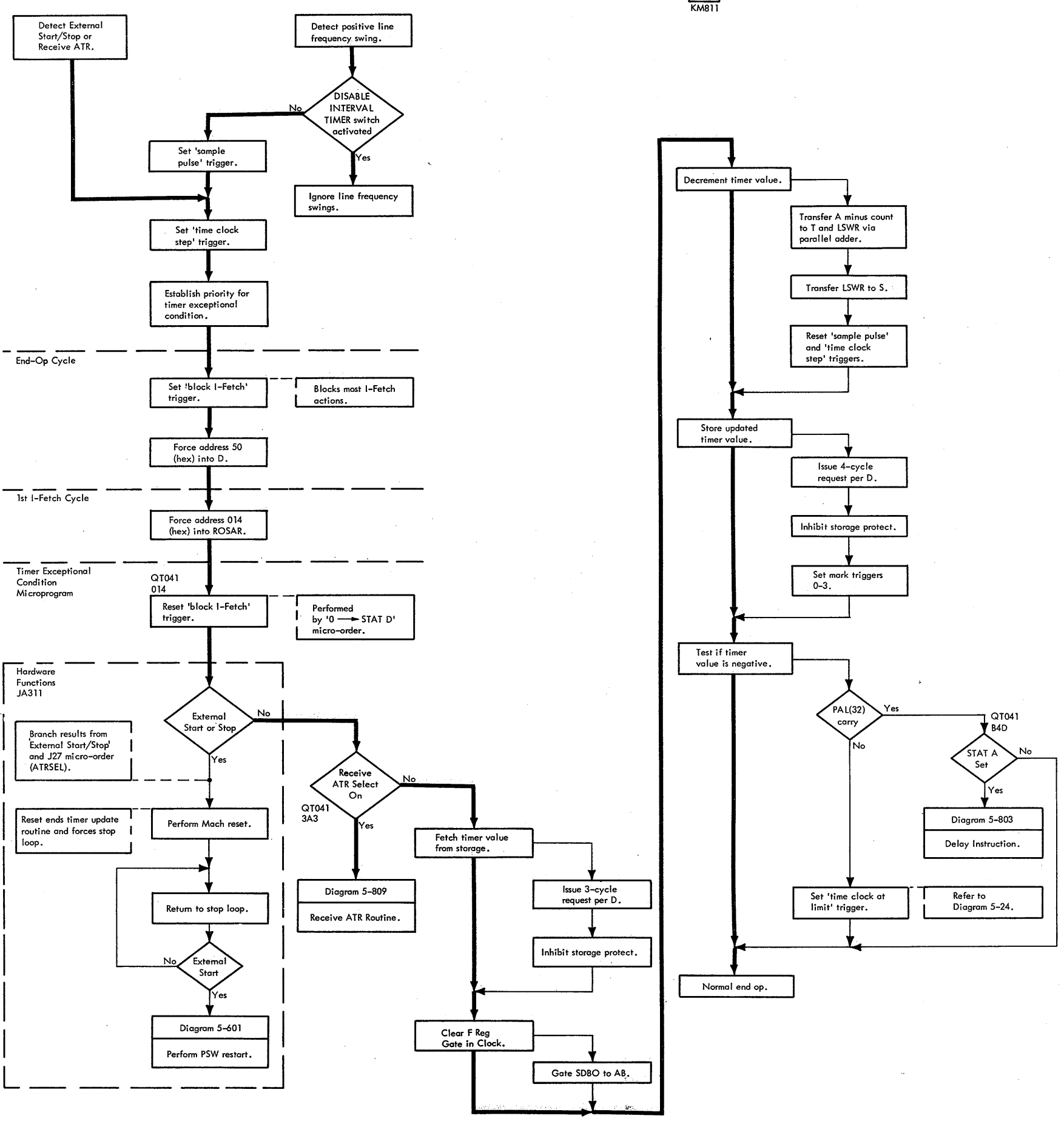
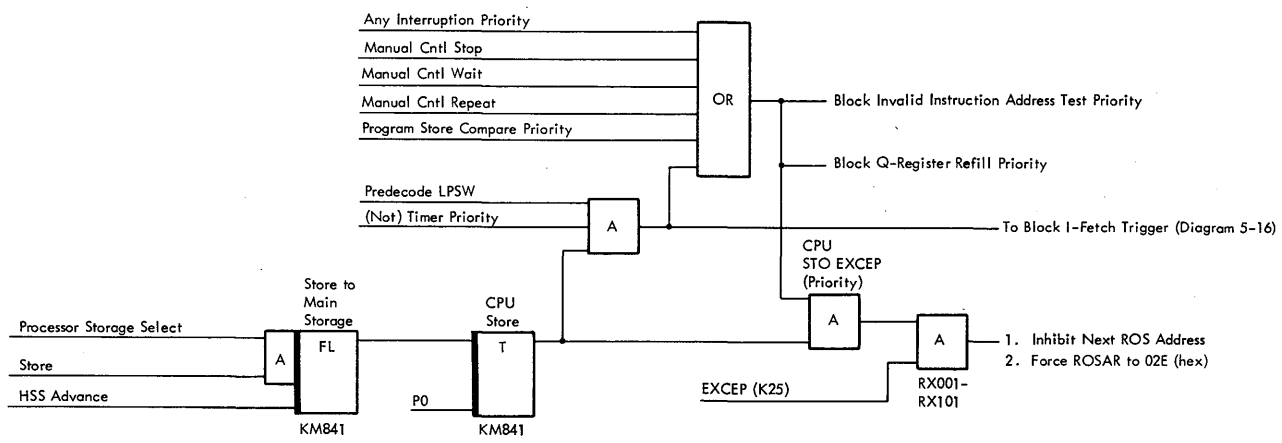


Diagram 5-17. Timer Exceptional Condition



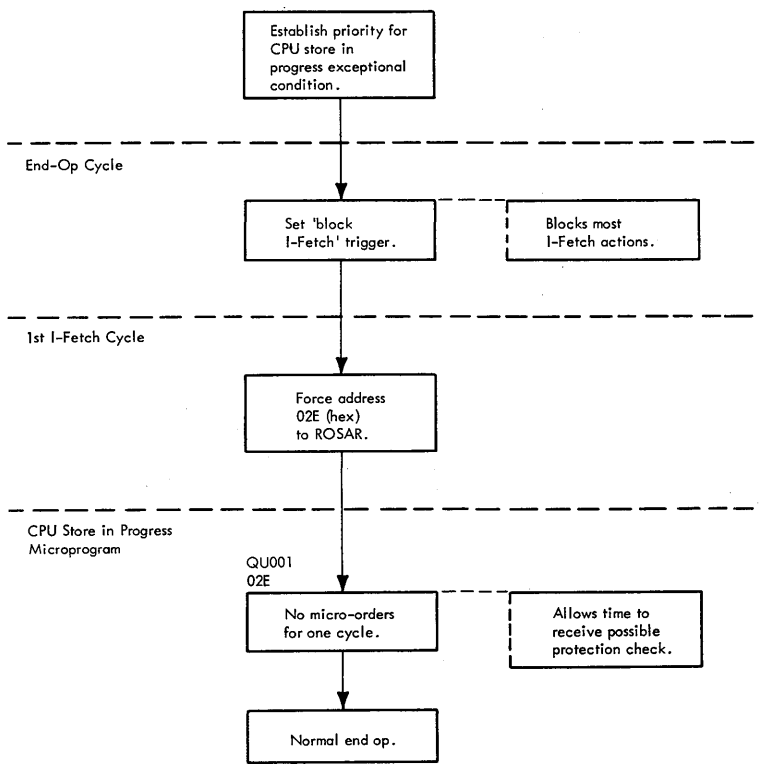


Diagram 5-18. CPU Store In Progress Exceptional Condition

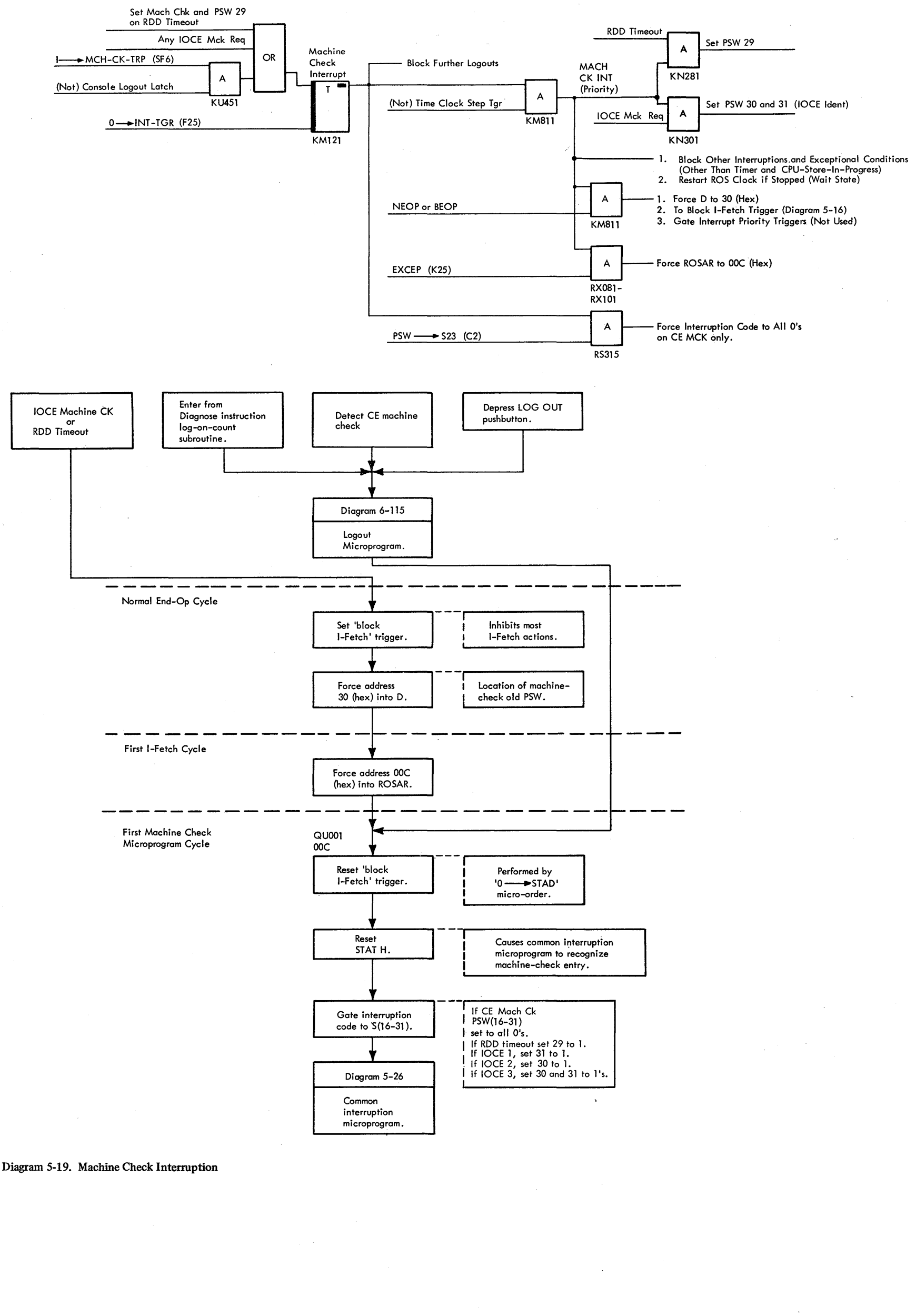
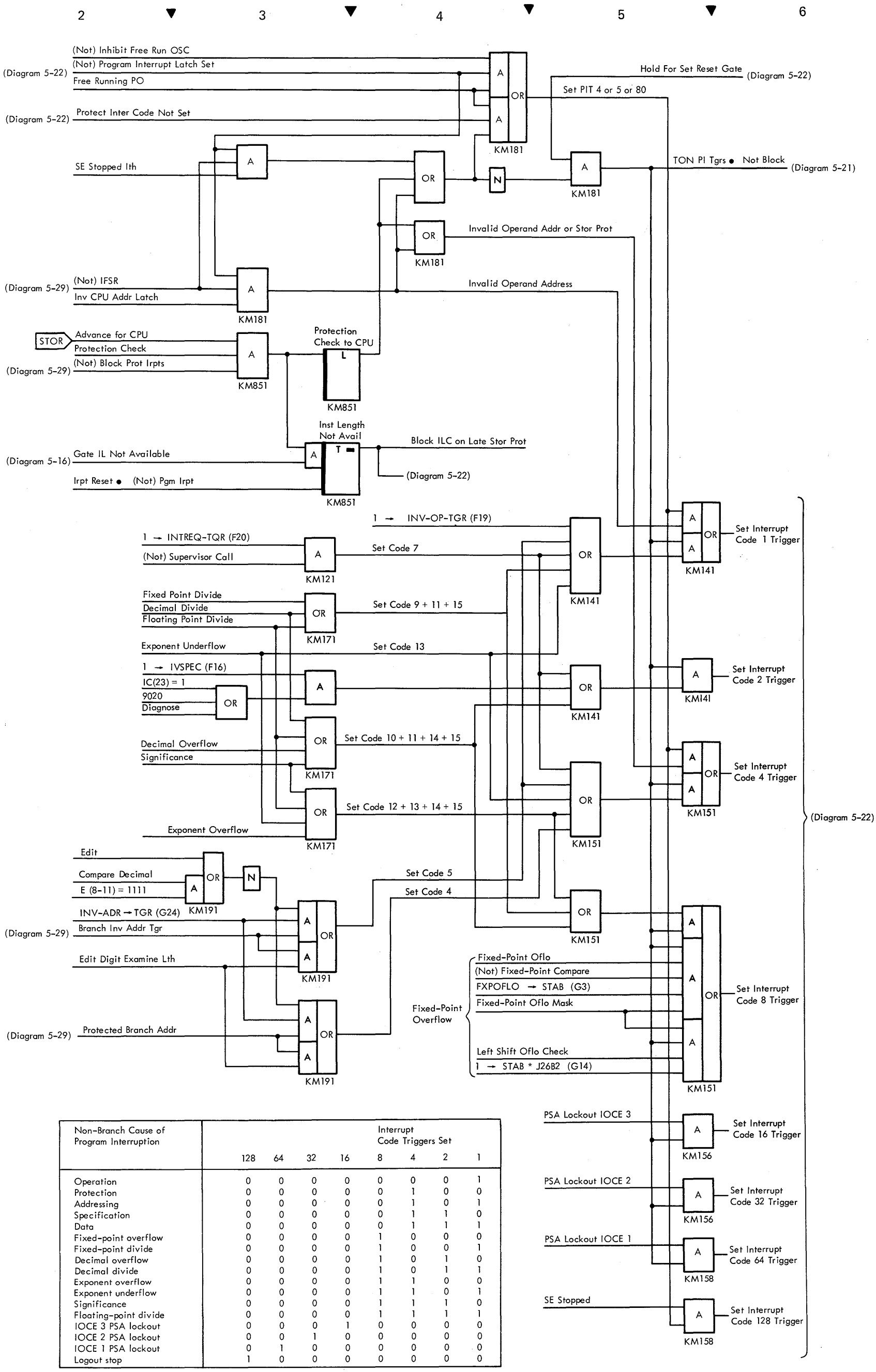
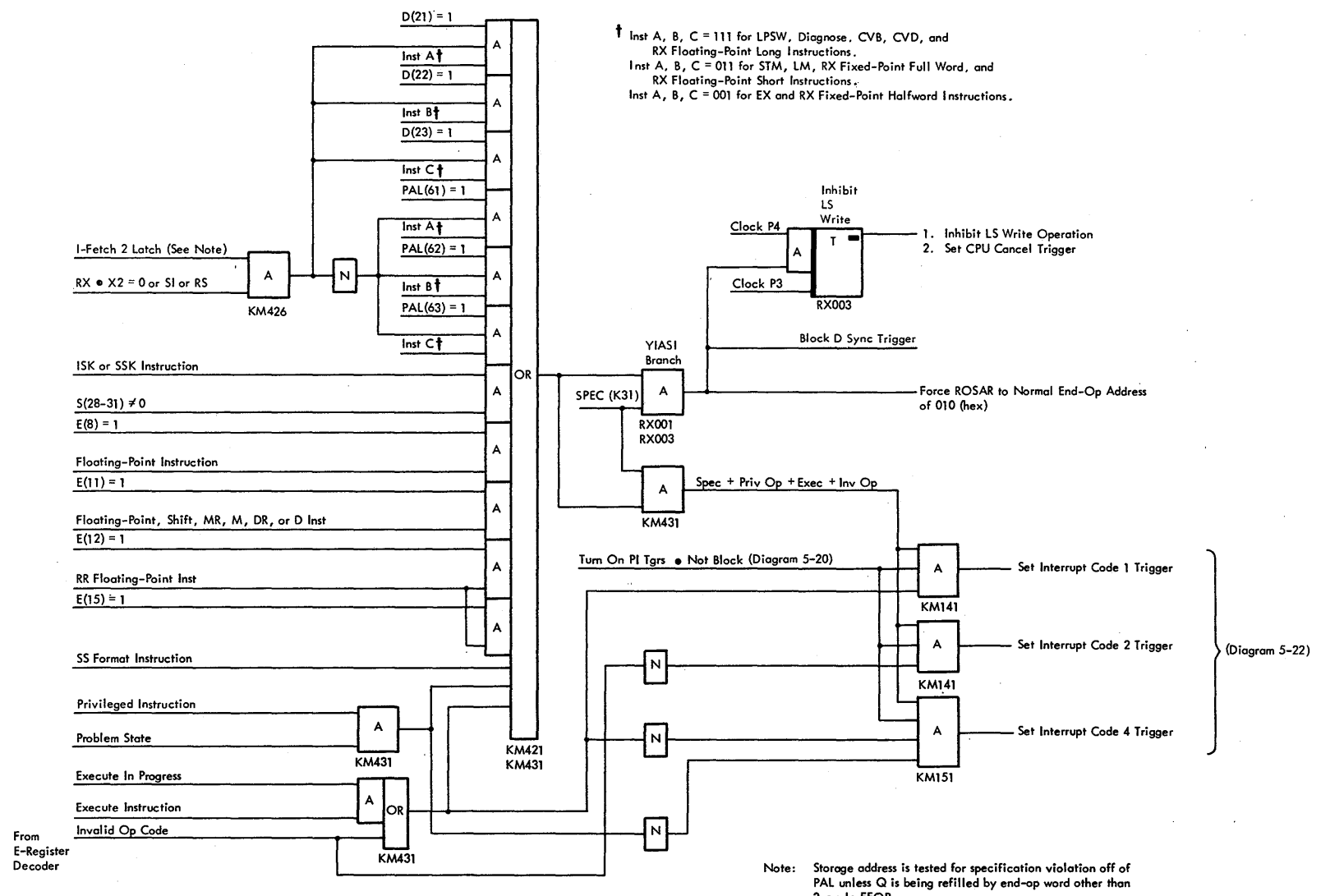


Diagram 5-19. Machine Check Interruption



Non-Branch Cause of Program Interruption	Interrupt Code Triggers Set							
	128	64	32	16	8	4	2	1
Operation	0	0	0	0	0	0	0	1
Protection	0	0	0	0	0	1	0	0
Addressing	0	0	0	0	0	1	0	1
Specification	0	0	0	0	0	1	1	0
Data	0	0	0	0	0	1	1	1
Fixed-point overflow	0	0	0	0	1	0	0	0
Fixed-point divide	0	0	0	0	1	0	0	1
Decimal overflow	0	0	0	0	1	0	1	0
Decimal divide	0	0	0	0	1	0	1	1
Exponent overflow	0	0	0	0	1	1	0	0
Exponent underflow	0	0	0	0	1	1	0	1
Significance	0	0	0	0	1	1	1	0
Floating-point divide	0	0	0	0	1	1	1	1
IOCE 3 PSA lockout	0	0	0	1	0	0	0	0
IOCE 2 PSA lockout	0	0	1	0	0	0	0	0
IOCE 1 PSA lockout	0	1	0	0	0	0	0	0
Logout stop	1	0	0	0	0	0	0	0

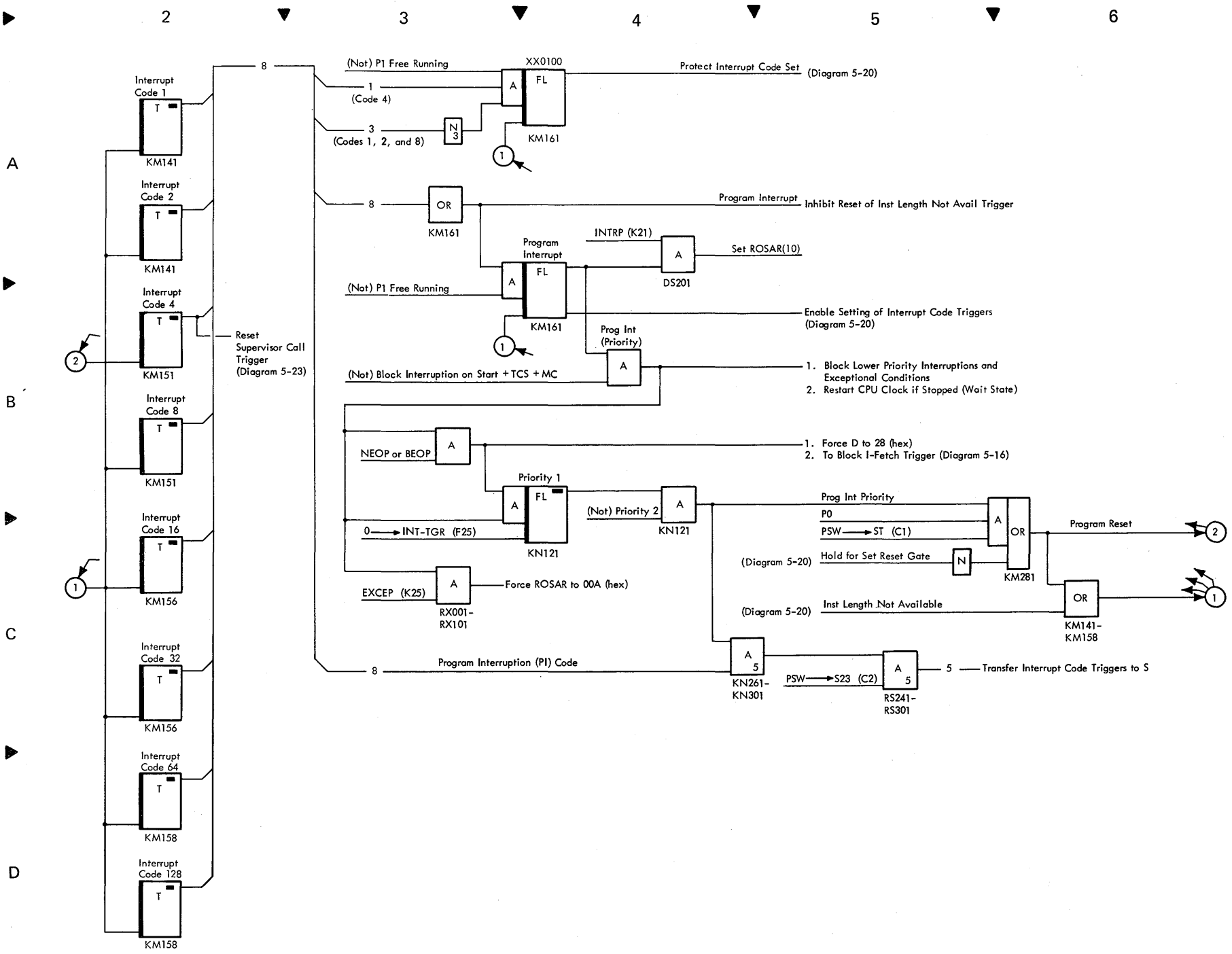
Diagram 5-20. Non-Branch Setting of Interrupt Code Triggers



Note: Storage address is tested for specification violation off of PAL unless Q is being refilled by end-op word other than 2-cycle EEOP.

SPEC Y-Branch Cause	Interrupt Code Triggers Set				
	16	8	4	2	1
Operation	0	0	0	0	1
Privileged Operation	0	0	0	1	0
Execute	0	0	0	1	1
Specification	0	0	1	1	0

Diagram 5-21. SPEC Y-Branch Setting of Interrupt Code Triggers



See Diagrams 5-20 and 5-21 for conditions that set 'interrupt code' triggers

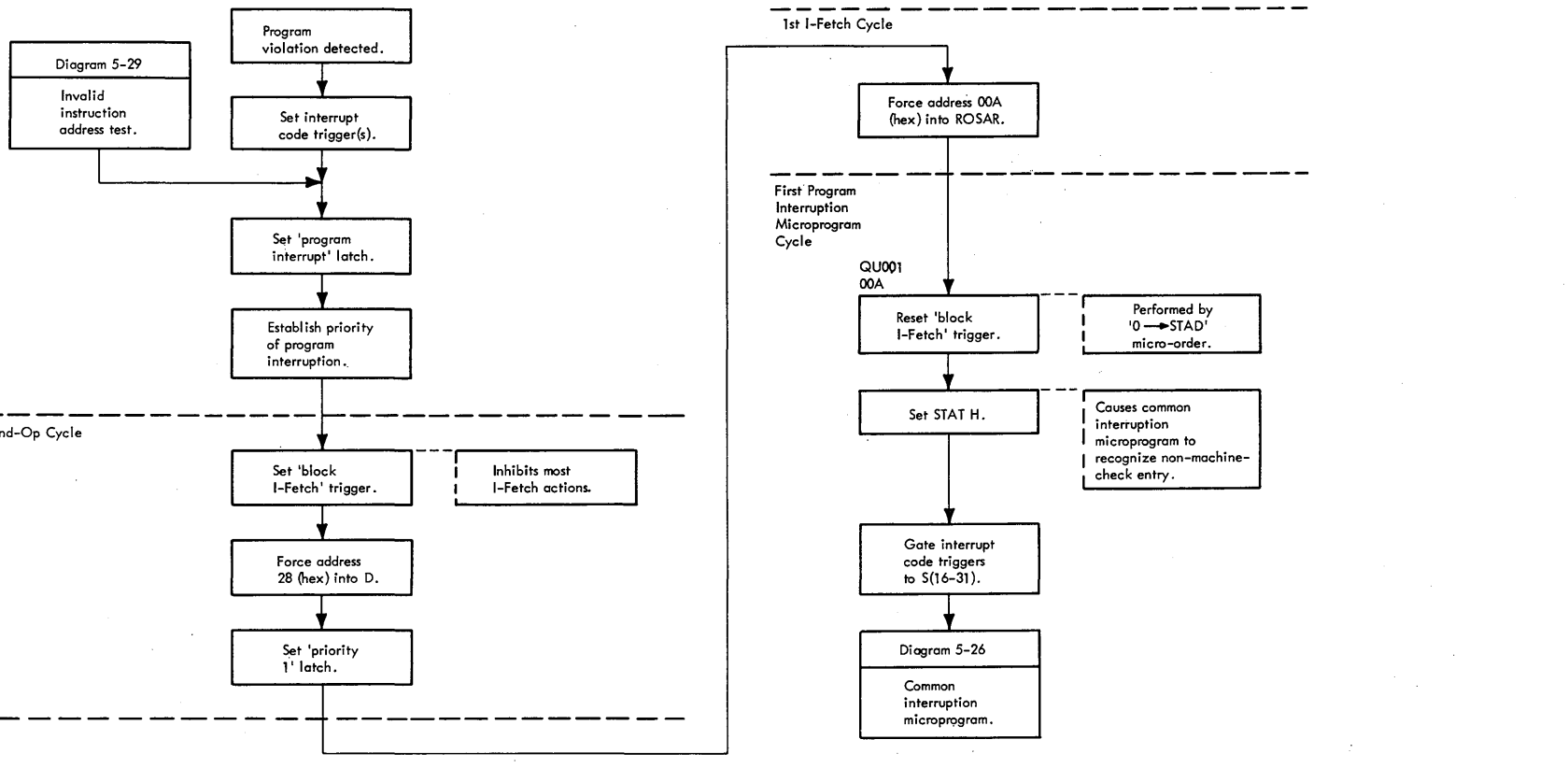
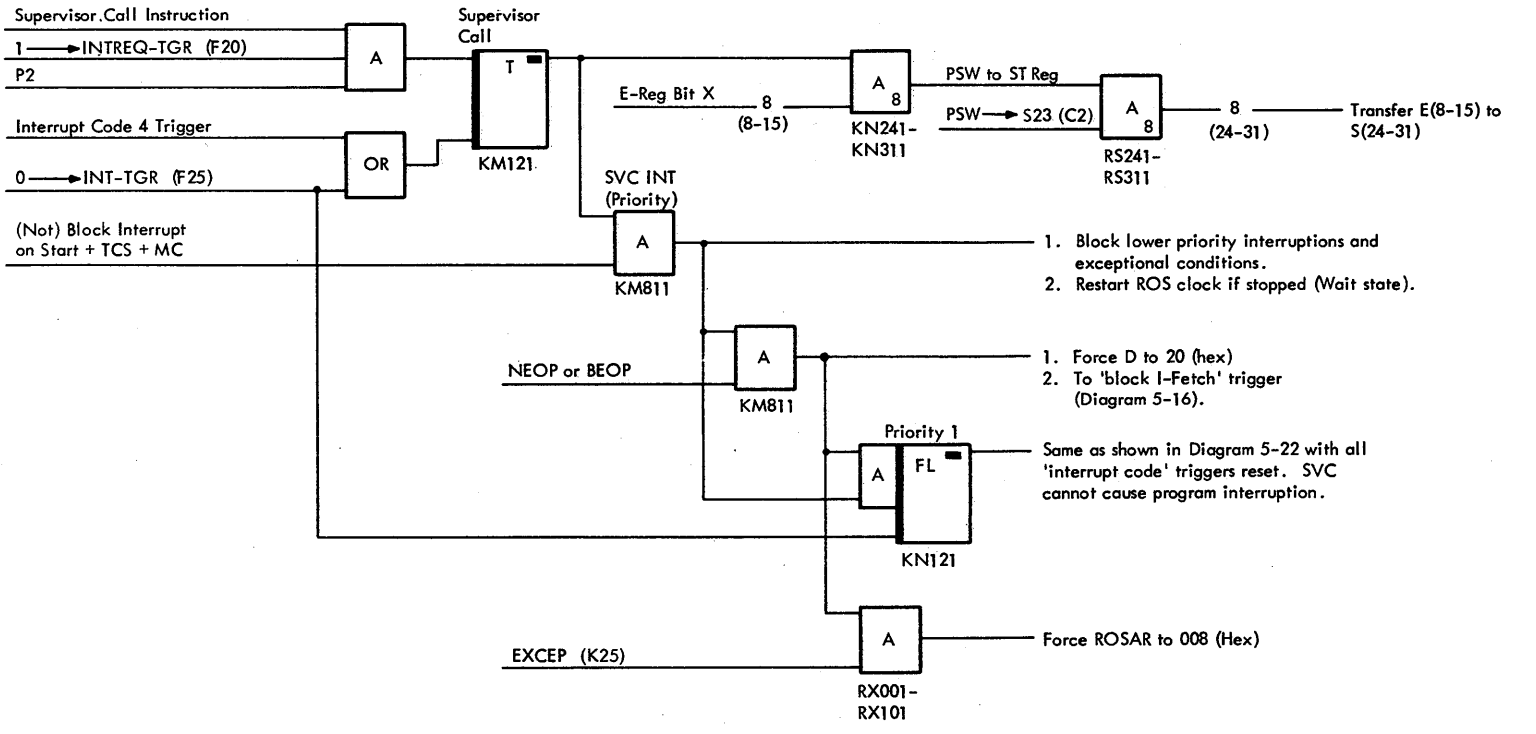
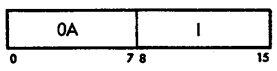


Diagram 5-22. Program Interruption



C

- RR Format:



- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand (not used) is in A, B, and D.
  3. 2nd operand (not used) is in S and T.

D

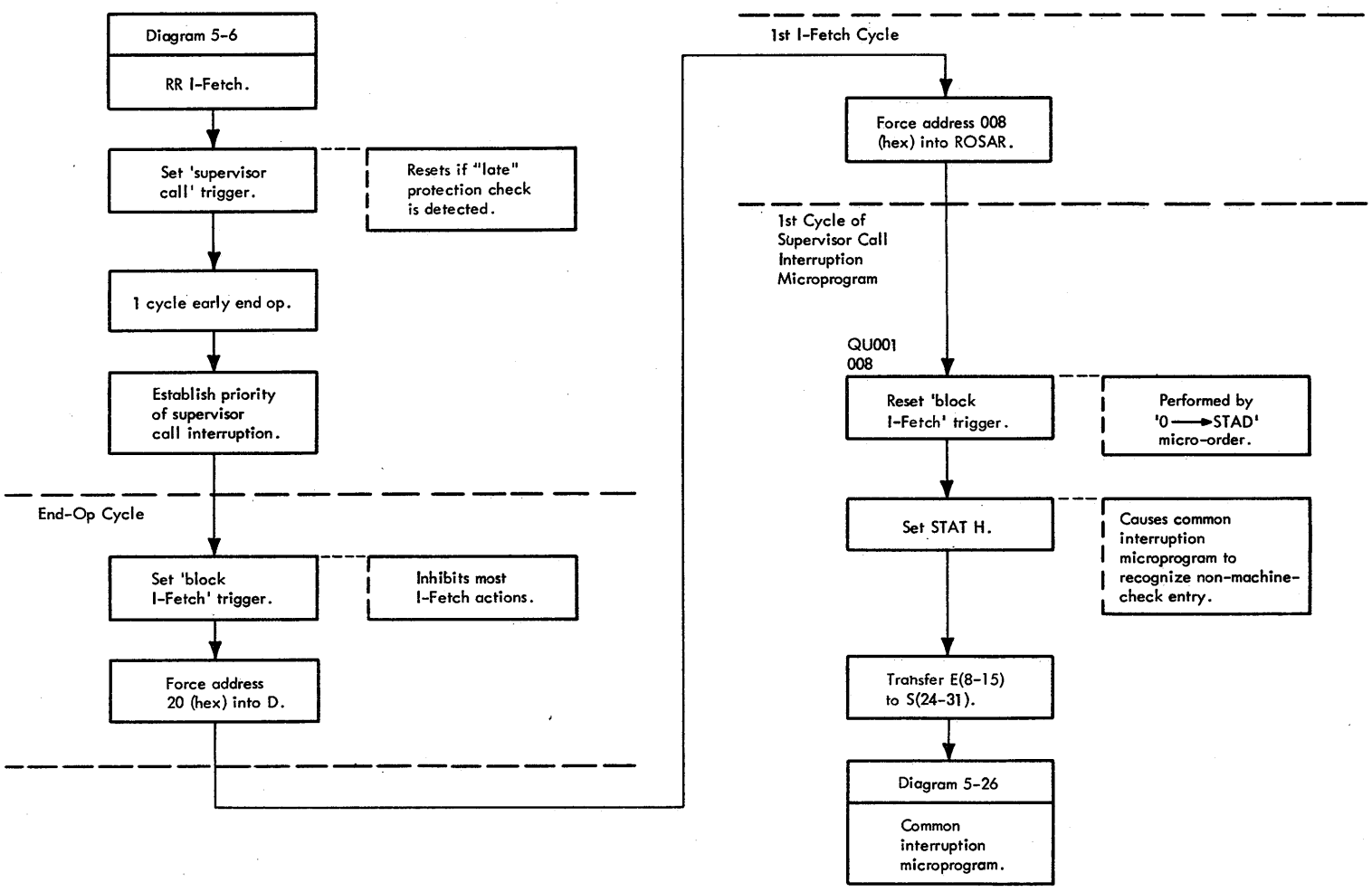


Diagram 5-23. Supervisor Call Interruption

G

H

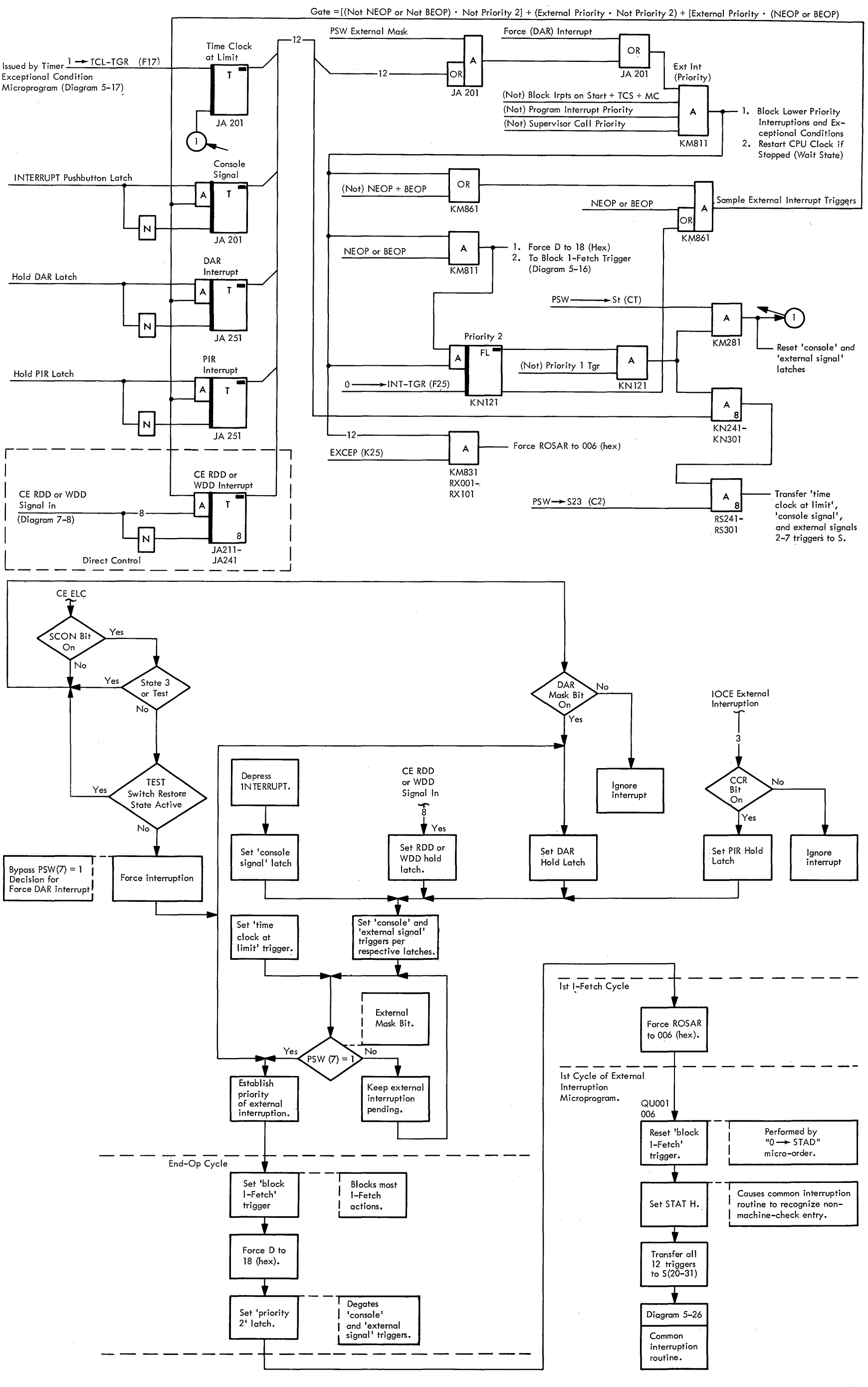


Diagram 5-24. External Interruption

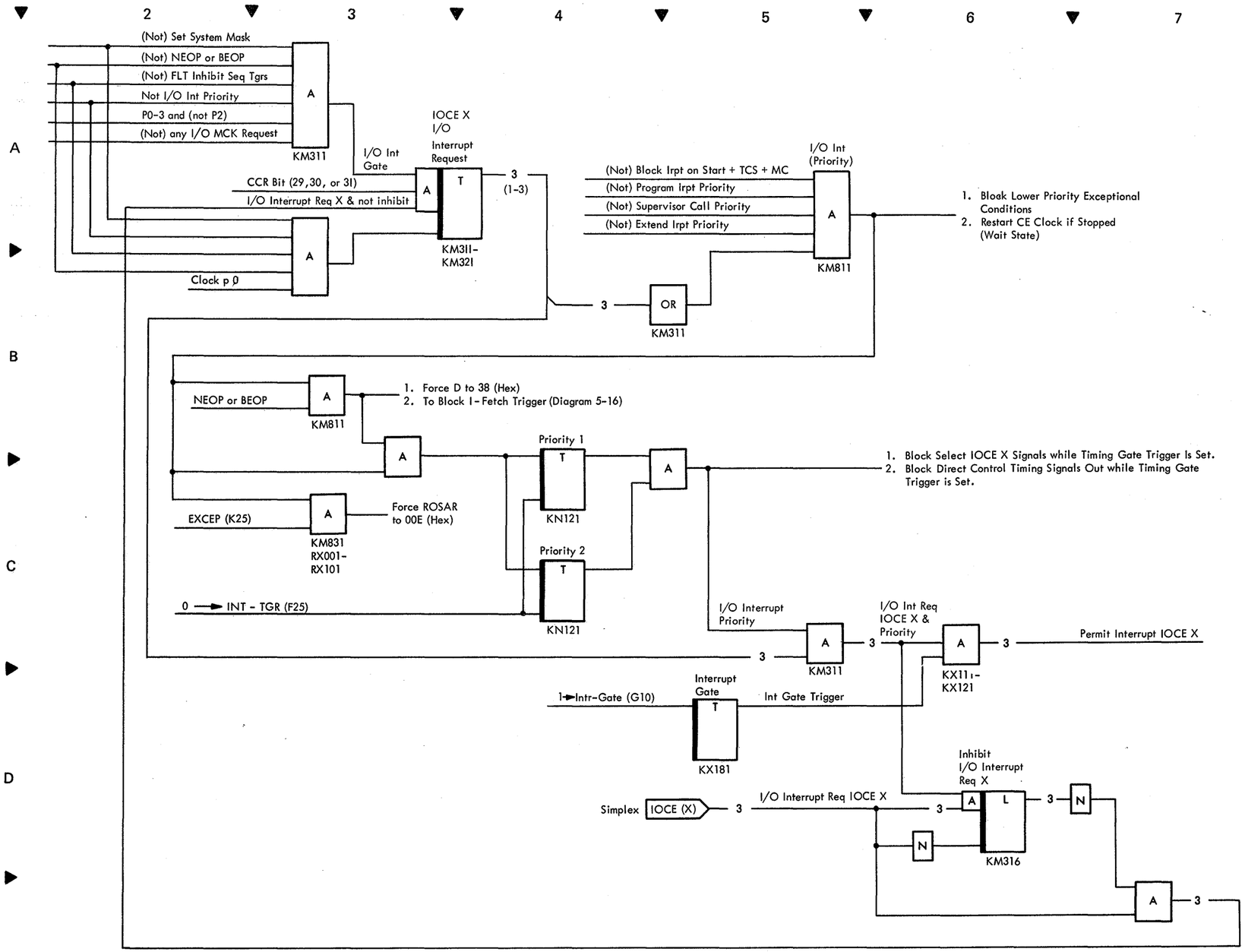


Diagram 5-25. I/O Interruption (Sheet 1 of 2)

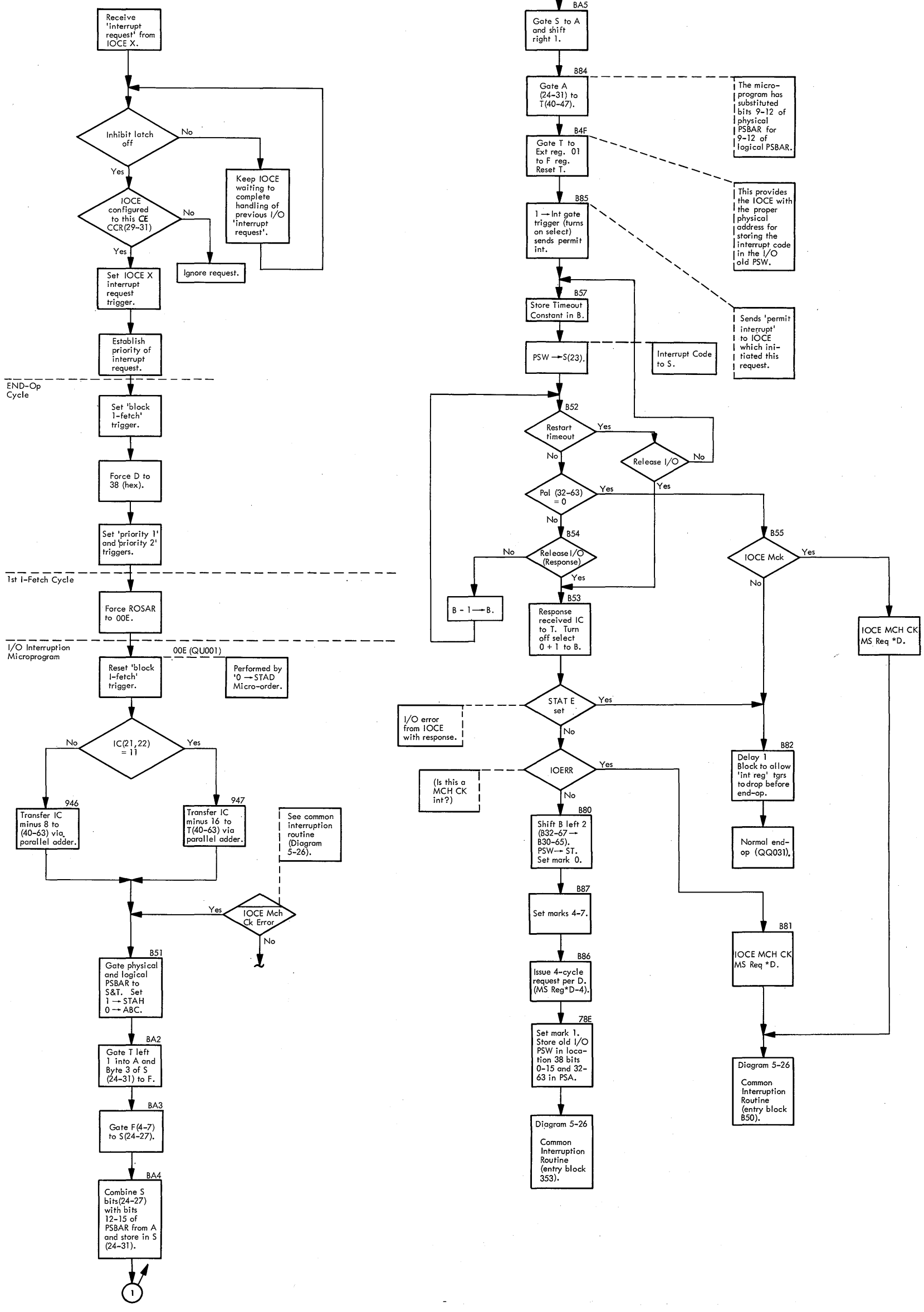
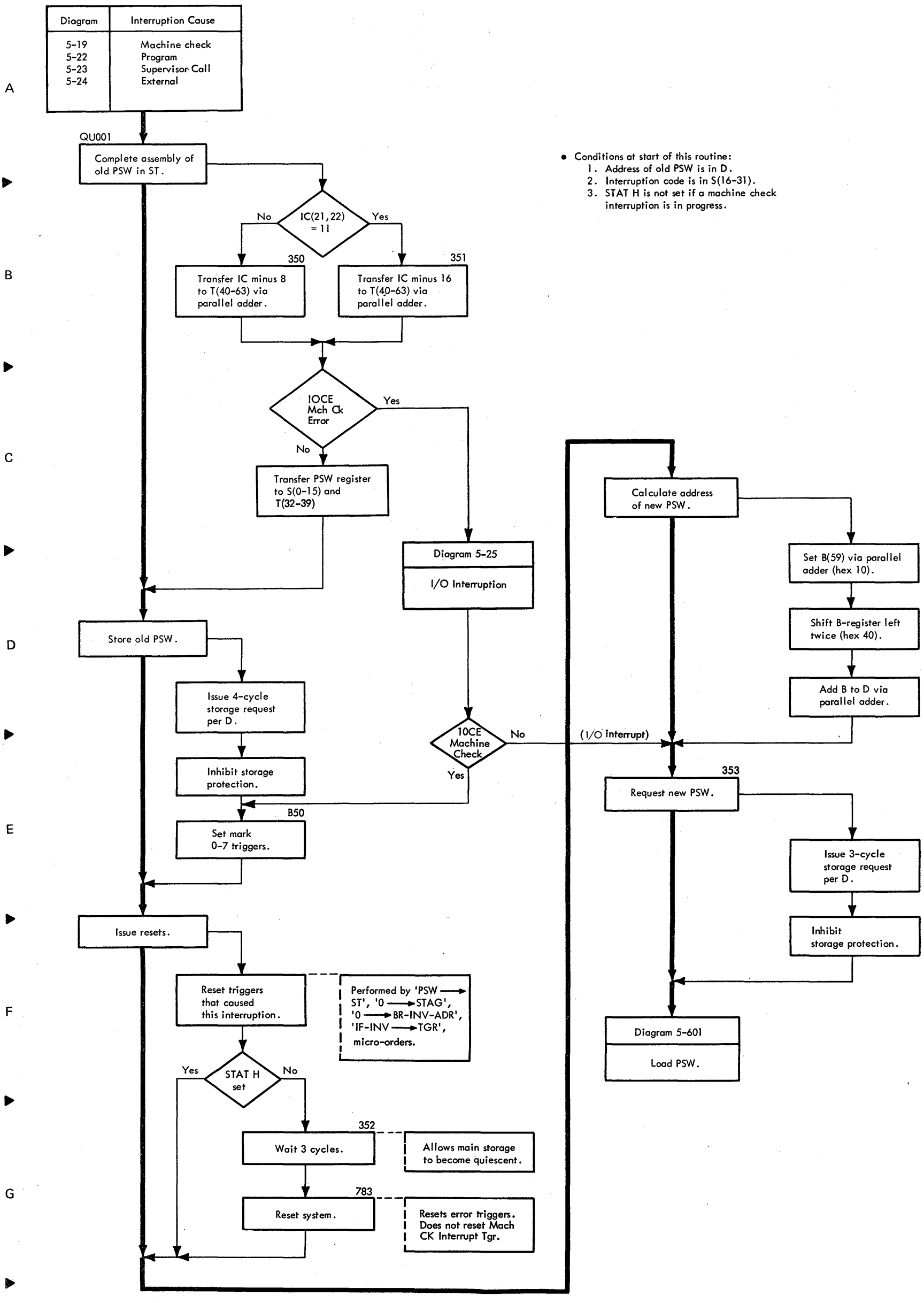


Diagram 5-25. I/O Interruption (Sheet 2 of 2)



- Conditions at start of this routine:
  1. Address of old PSW is in D.
  2. Interruption code is in S(16-31).
  3. STAT H is not set if a machine check interruption is in progress.

Diagram 5-26. Common Interruption Routine

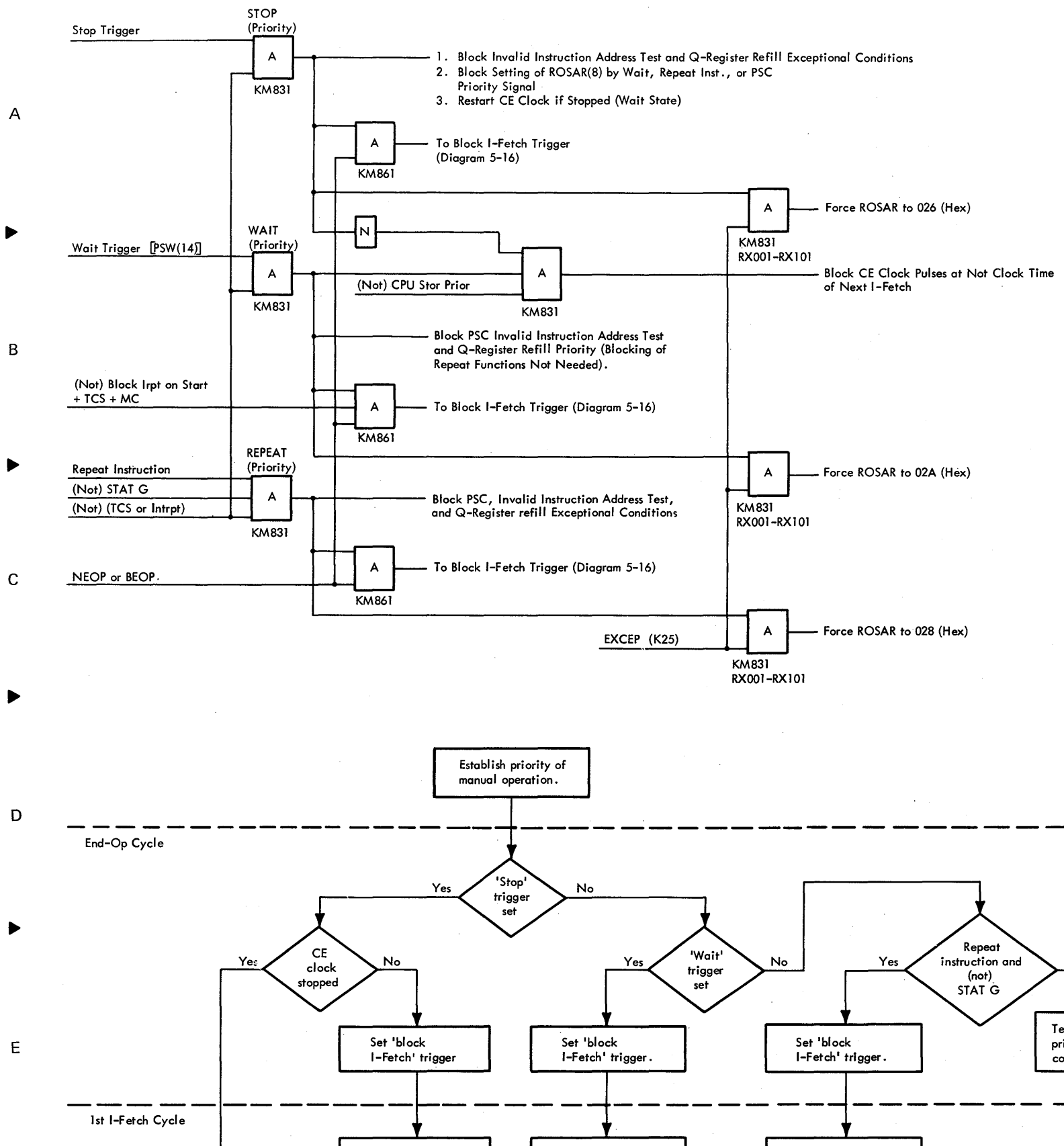
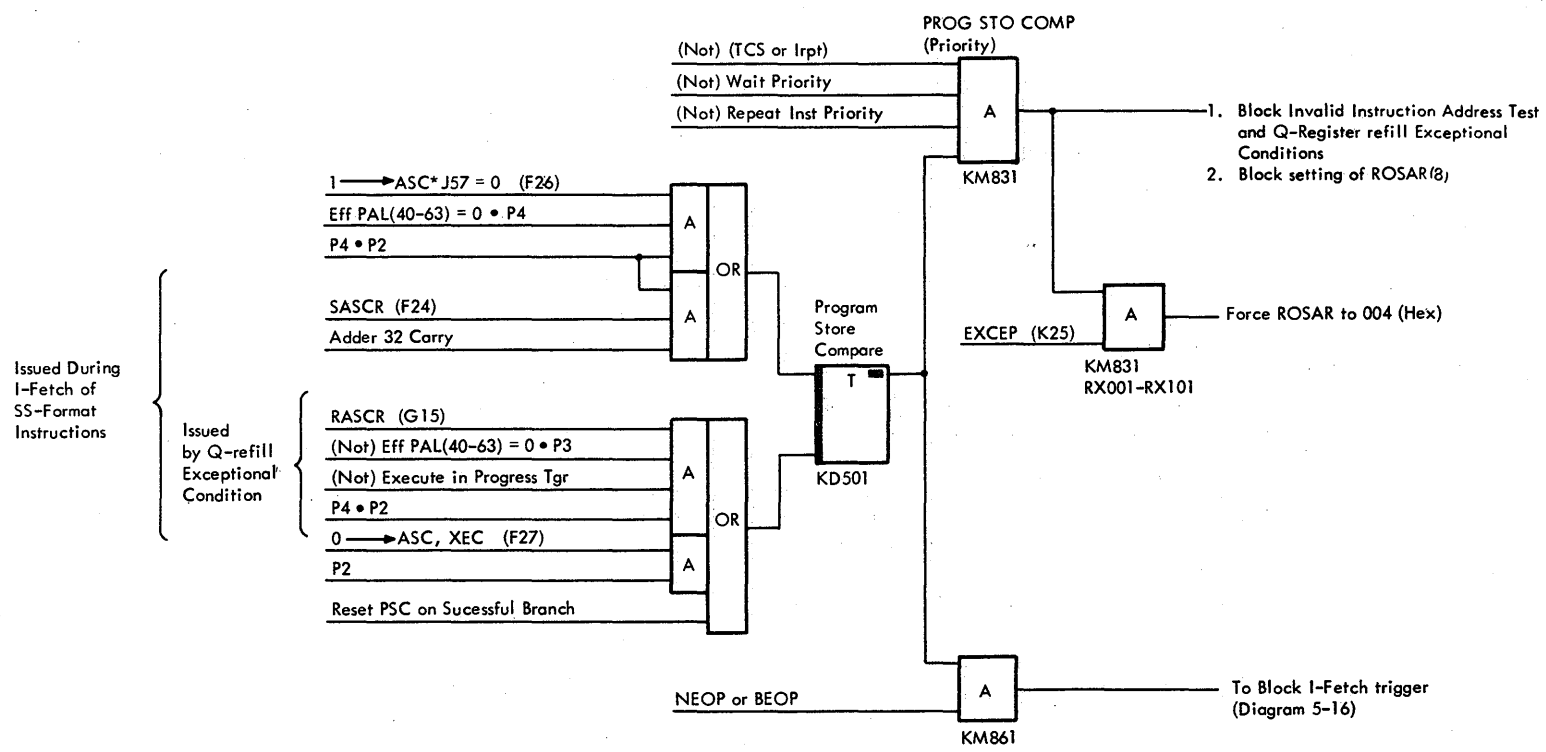


Diagram 5-27. Manual Control Exceptional Conditions



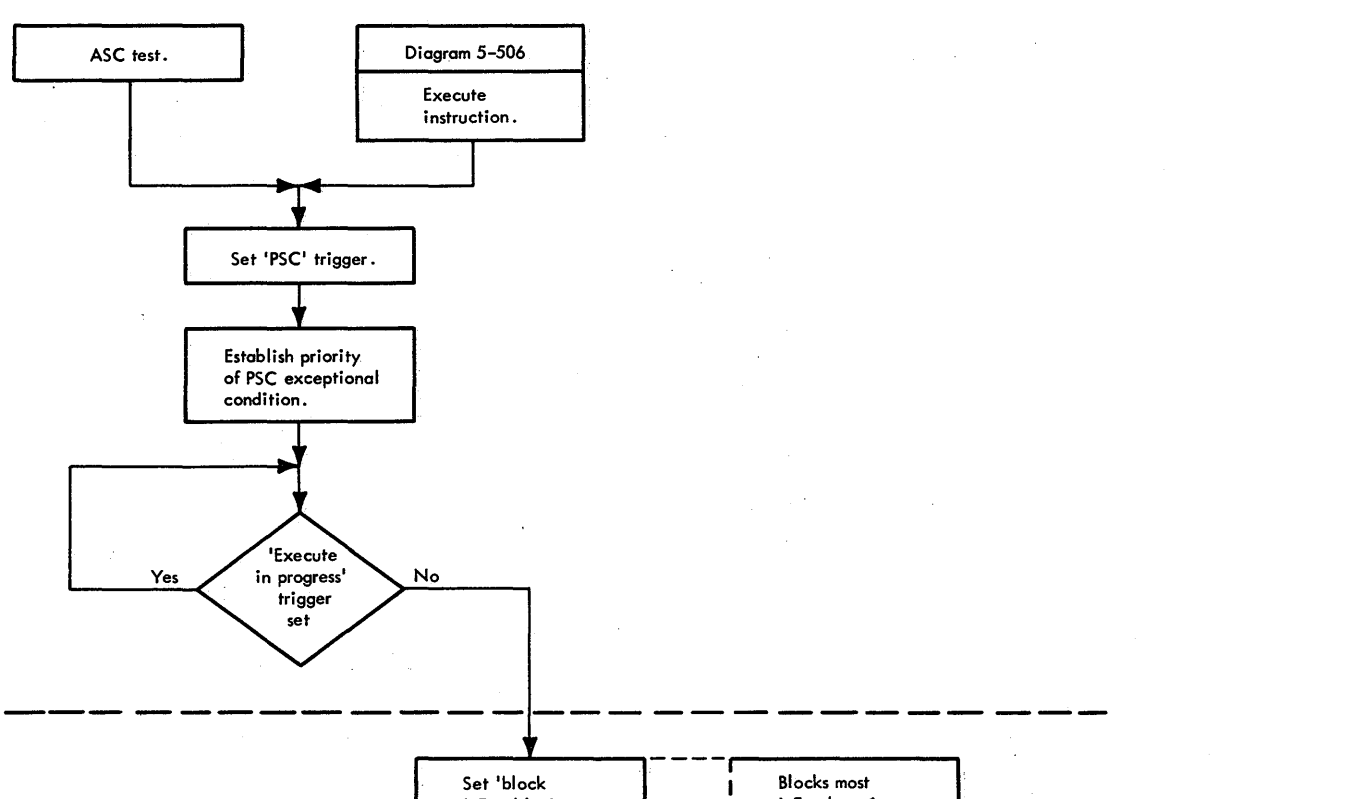


Diagram 5-28. Program Store Compare Exceptional Condition

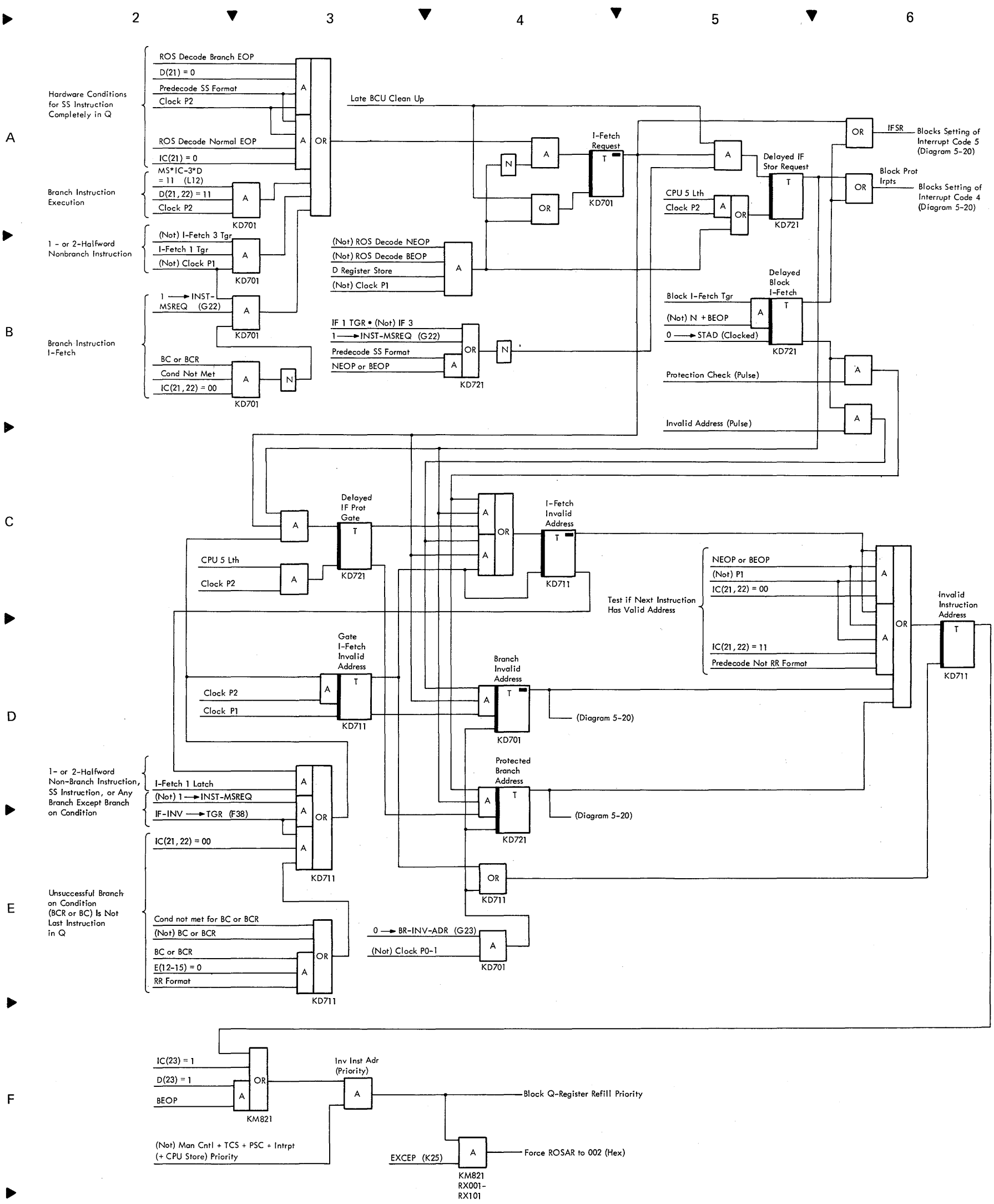


Diagram 5-29. Invalid Instruction Address Test Exceptional Condition (Sheet 1 of 2)

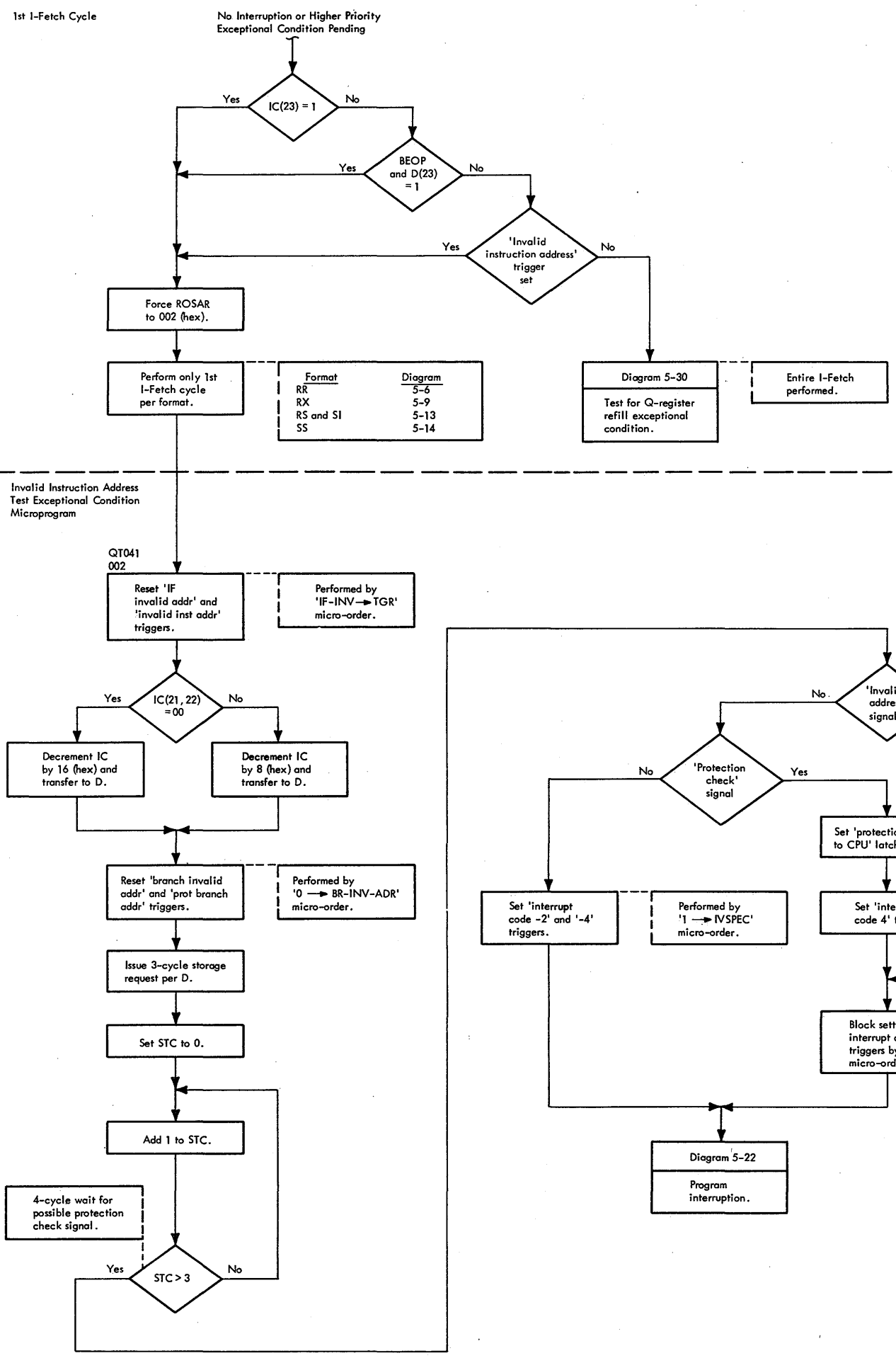
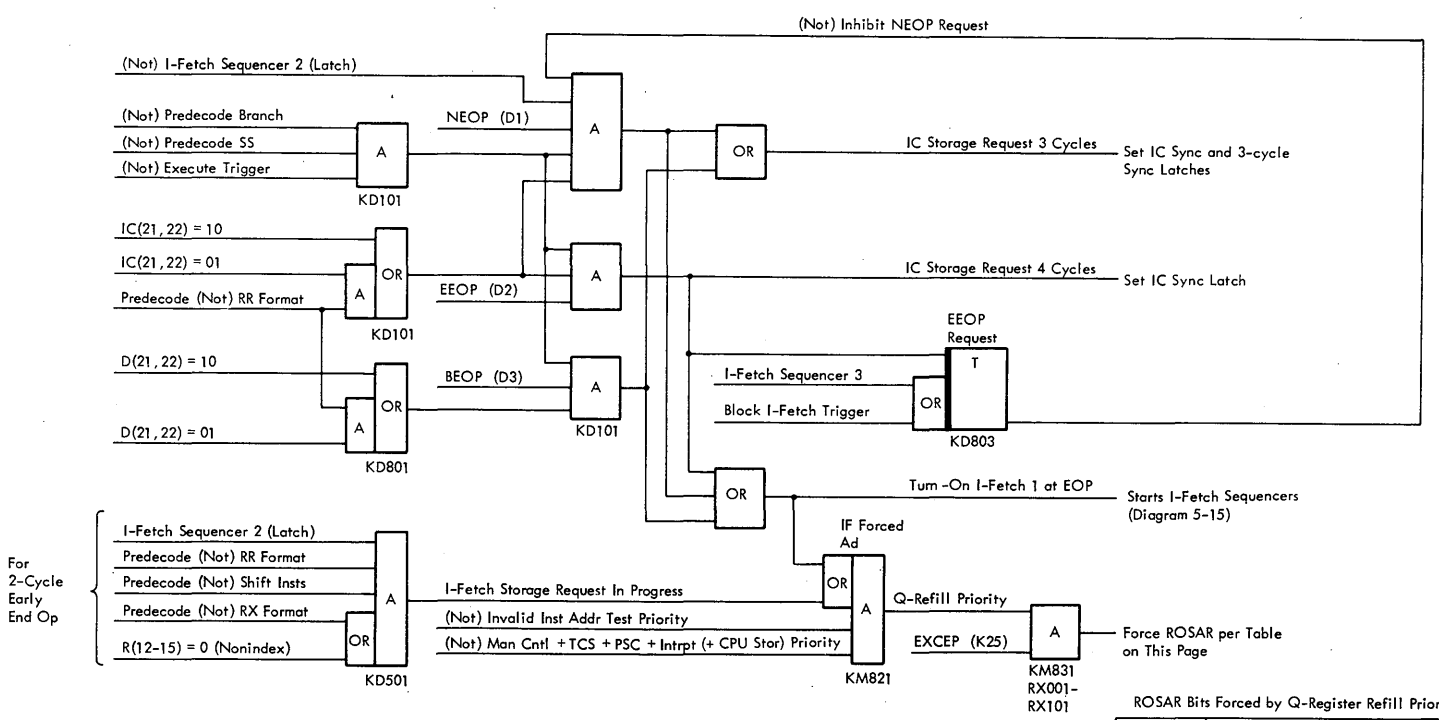


Diagram 5-29. Invalid Instruction Address Test Exceptional Condition (Sheet 2 of 2)

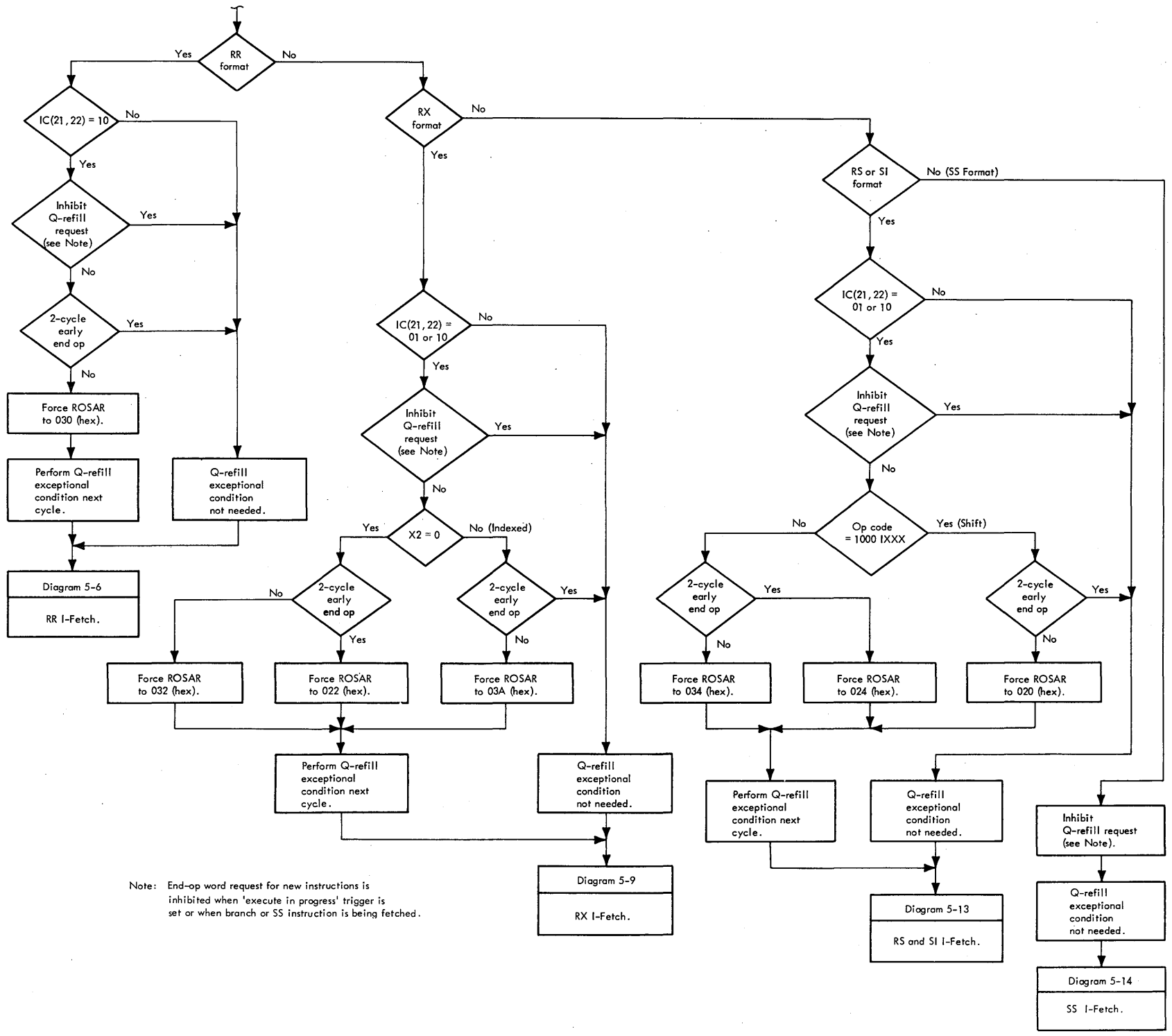


ROSAR Bits Forced by Q-Register Refill Priority and EXCEP Micro-Order

Bit	Setting Condition
0-5	0's (unconditional)
6	1 (unconditional)
7	Not shift • Not 2-cycle early end op
8	Not shift • Not 2-cycle early end op • indexed RX format
9	Not shift • R(0) = 1
10	Not shift • R(1) = 1
11	0 (unconditional)

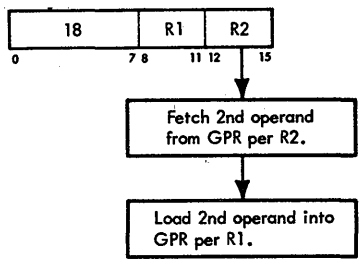
1st I-Fetch Cycle

No Interruption, Timer, CPU Store in Progress, Manual Control, Program Store Compare, or Invalid Instruction Address Test Exceptional Condition Pending



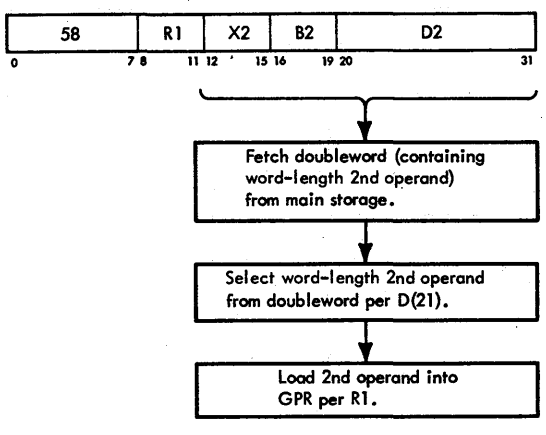
Note: End-op word request for new instructions is inhibited when 'execute in progress' trigger is set or when branch or SS instruction is being fetched.

Diagram 5-30. Test for Q-Register Refill Exceptional Condition



- RR format.
- Purpose: Load 2nd operand (in GPR, per R2) into 1st operand location (in GPR, per R1)

A. Load, LR (18)

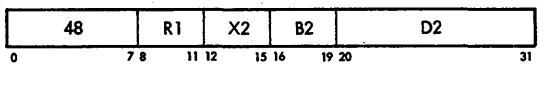


- RX format.
- Purpose: Load 2nd operand (in storage) into 1st operand location (in GPR, per R1).

B. Load, L (58)

Diagram 5-101. Load, LR (18); Load, L (58)

- RX format:



- Purpose: Load halfword 2nd operand (in storage) into 1st operand location (in GPR, per R1).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T (not used).
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D.

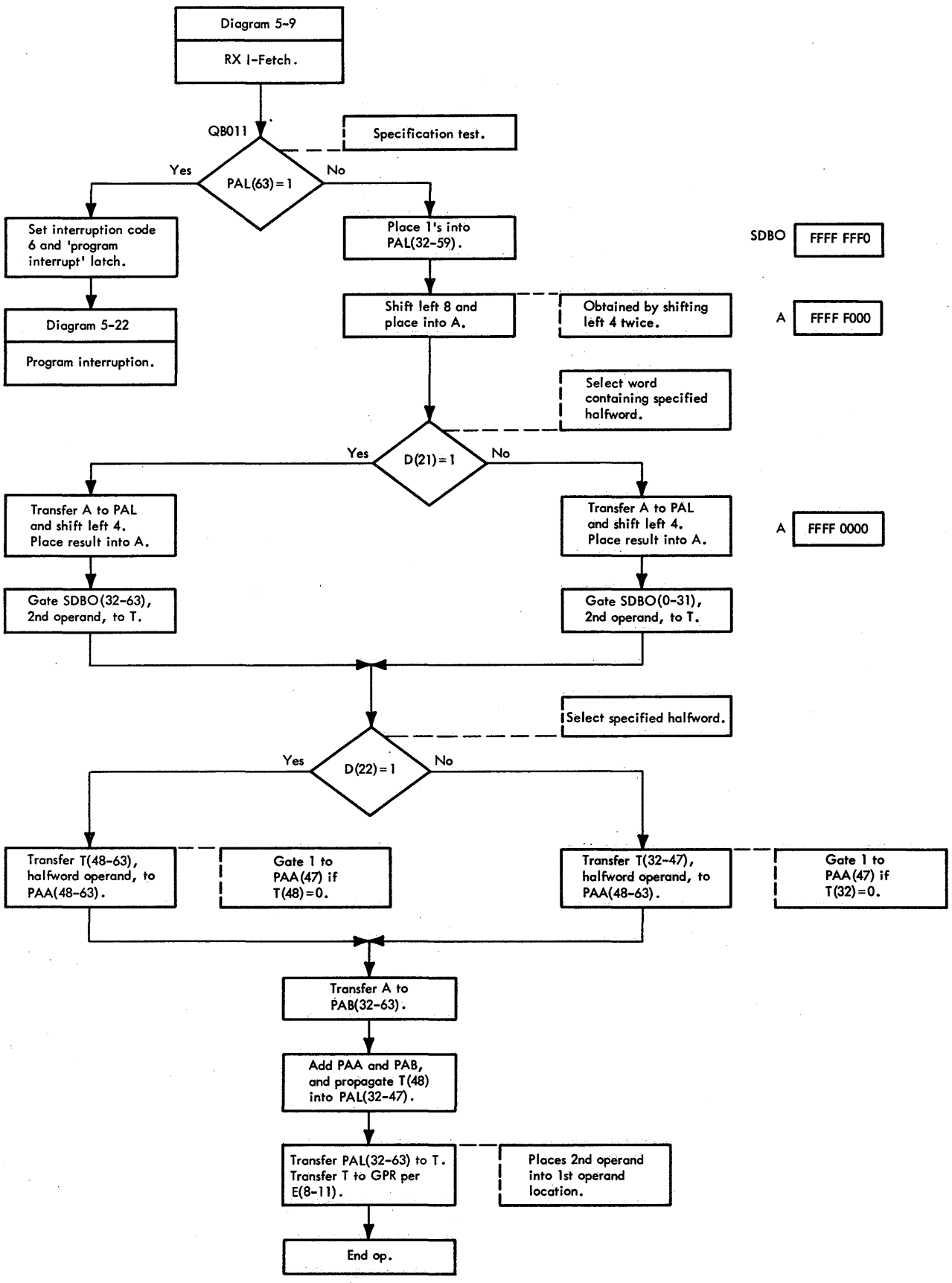
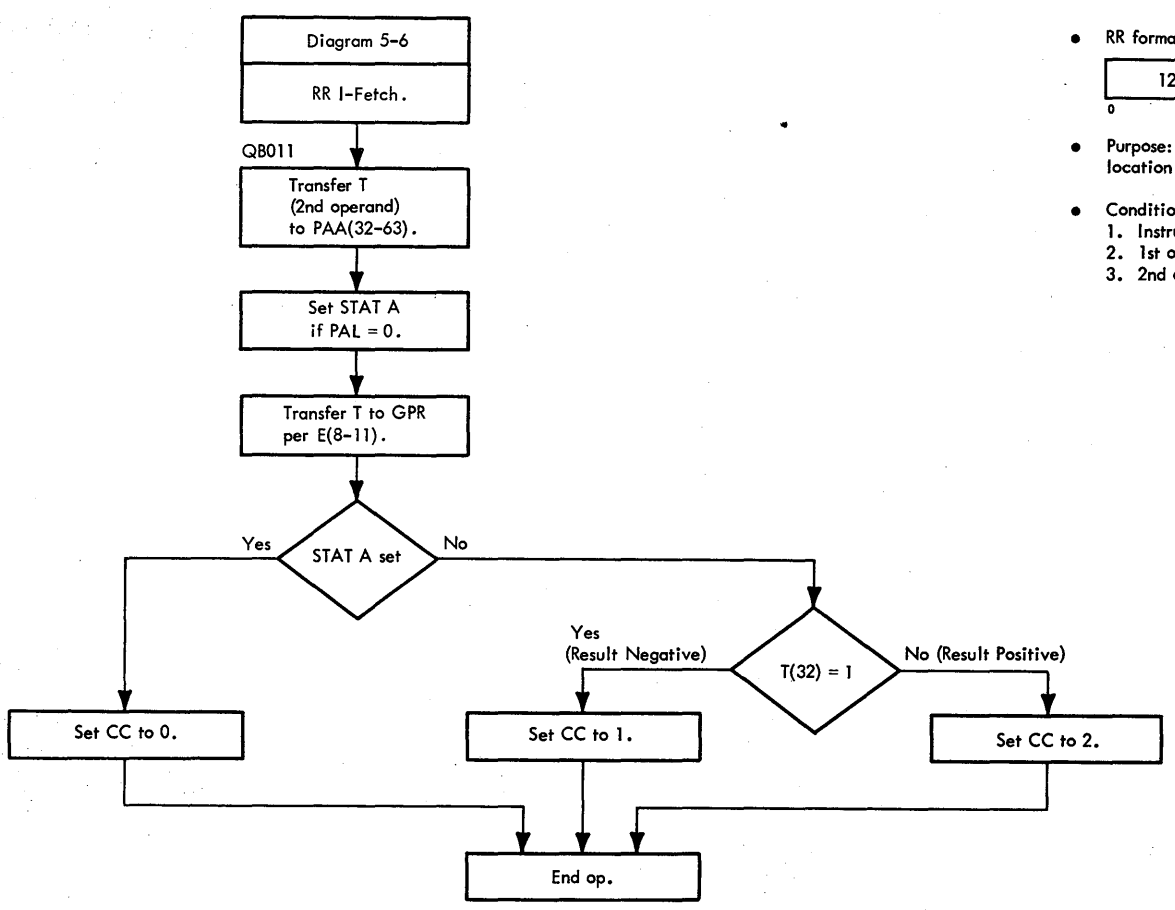
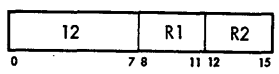


Diagram 5-102. Load Halfword, LH (48)

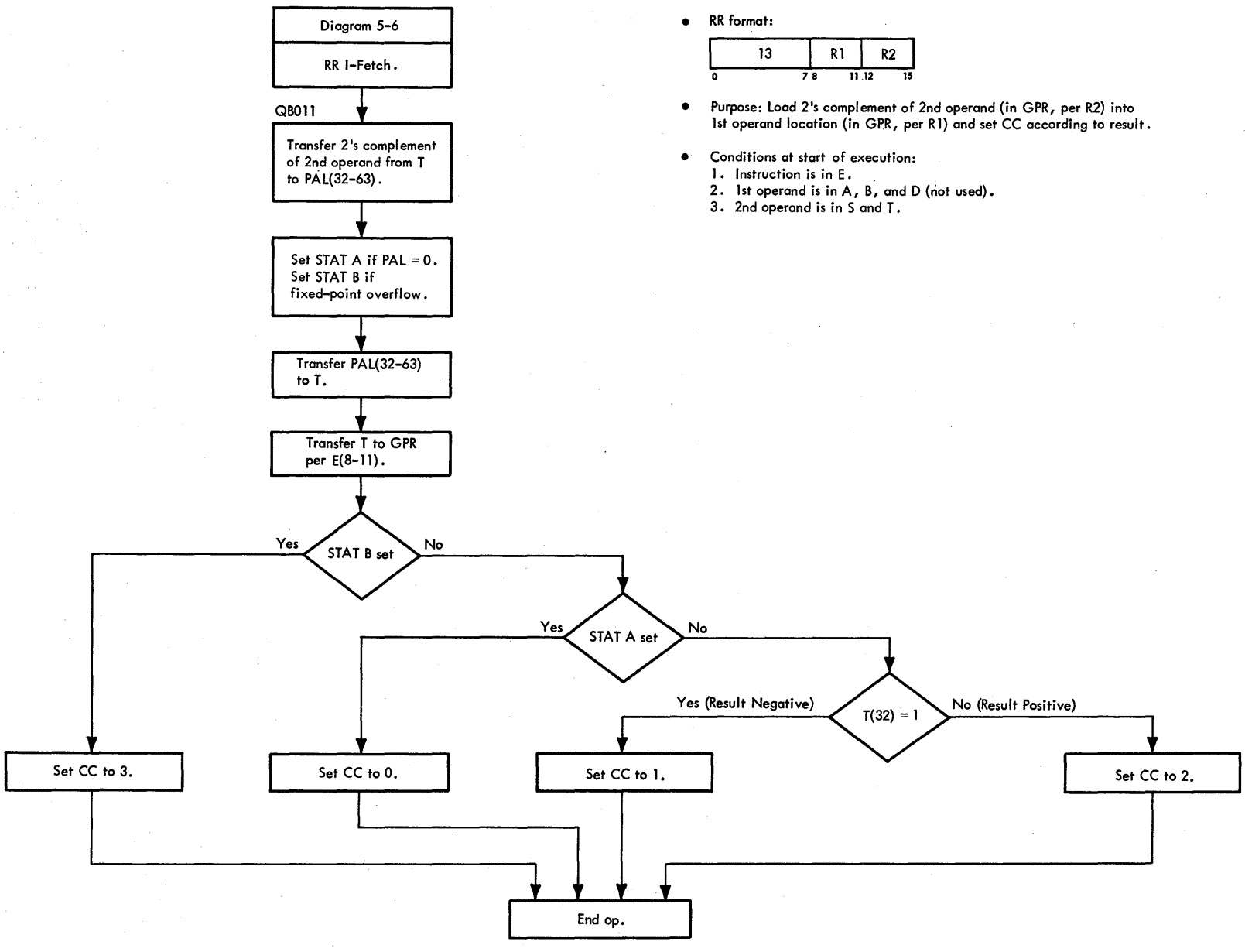


- RR format:



- Purpose: Load 2nd operand (in GPR, per R2) into 1st operand location (in GPR, per R1) and set CC according to result.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (not used).
  3. 2nd operand is in S and T.

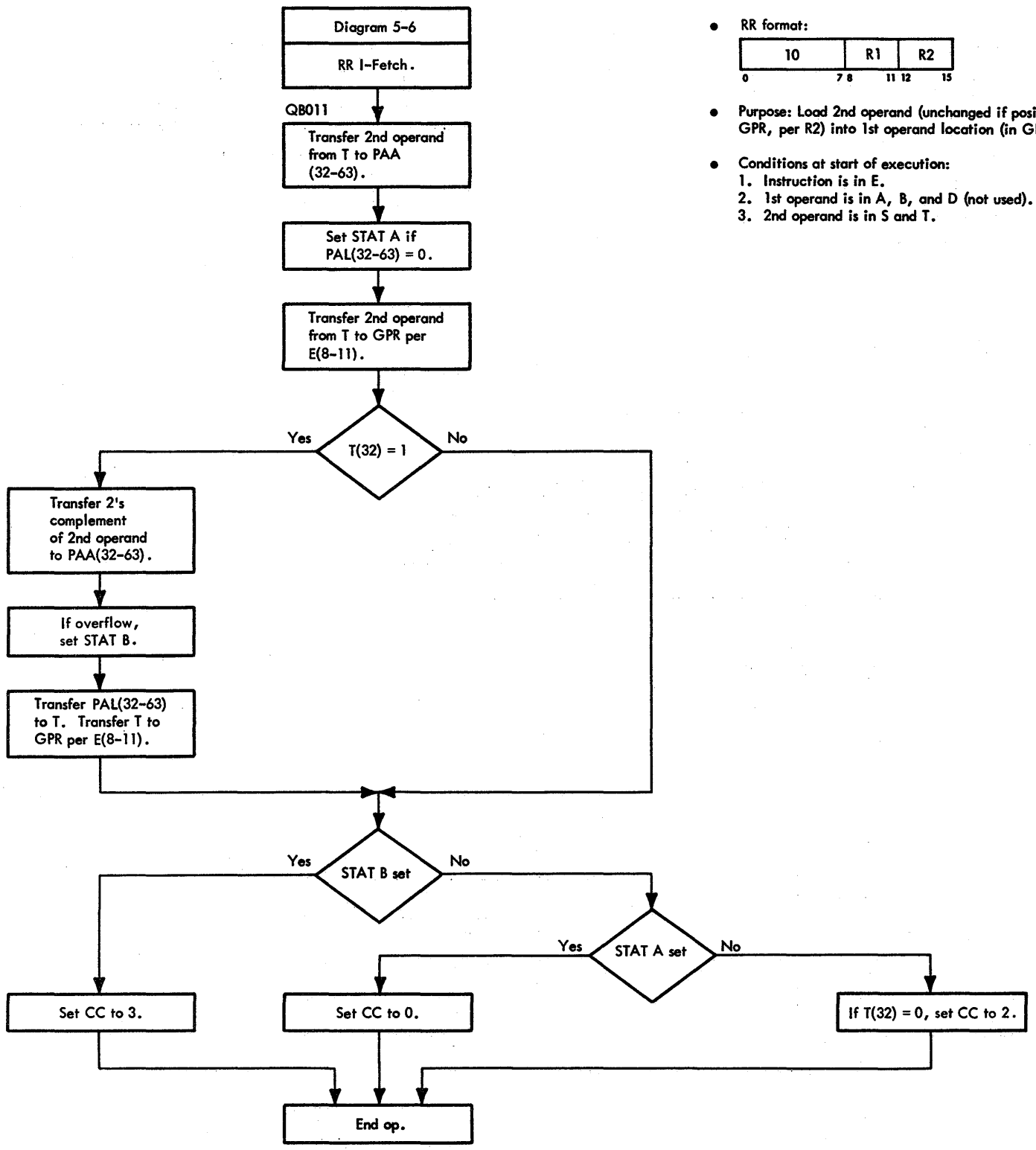
Diagram 5-103. Load and Test, LTR (12)



- RR format:

- Purpose: Load 2's complement of 2nd operand (in GPR, per R2) into 1st operand location (in GPR, per R1) and set CC according to result.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (not used).
  3. 2nd operand is in S and T.

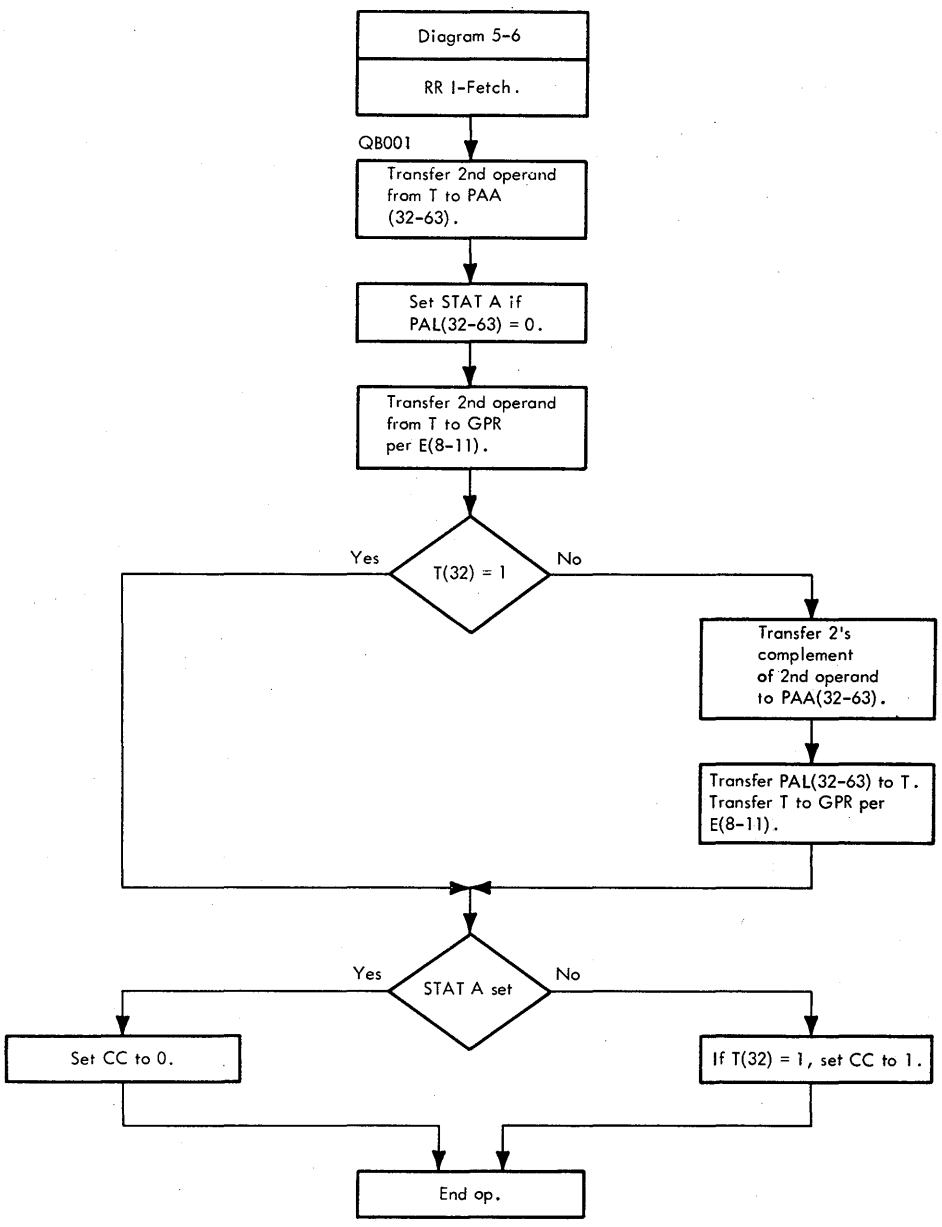
Diagram 5-104. Load Complement, LCR (13)



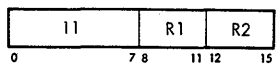
- RR format:

- Purpose: Load 2nd operand (unchanged if positive, 2's complemented if negative; in GPR, per R2) into 1st operand location (in GPR, per R1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (not used).
  3. 2nd operand is in S and T.

Diagram 5-105. Load Positive, LPR (10)



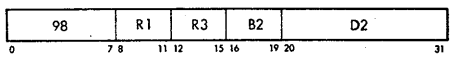
- RR format:



- Purpose: Load 2nd operand (unchanged if negative, 2's complemented if positive; in GPR per R2) into 1st operand location (in GPR, per R1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (not used).
  3. 2nd operand is in S and T.

Diagram 5-106. Load Negative, LNR (11)

- RS format:



- Purpose: Load 2nd operand (as many words as required; in storage) into GPR's, in ascending order, starting with 1st operand location (per R1) and ending with 3rd operand location (per R3).

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T (not used).
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D.

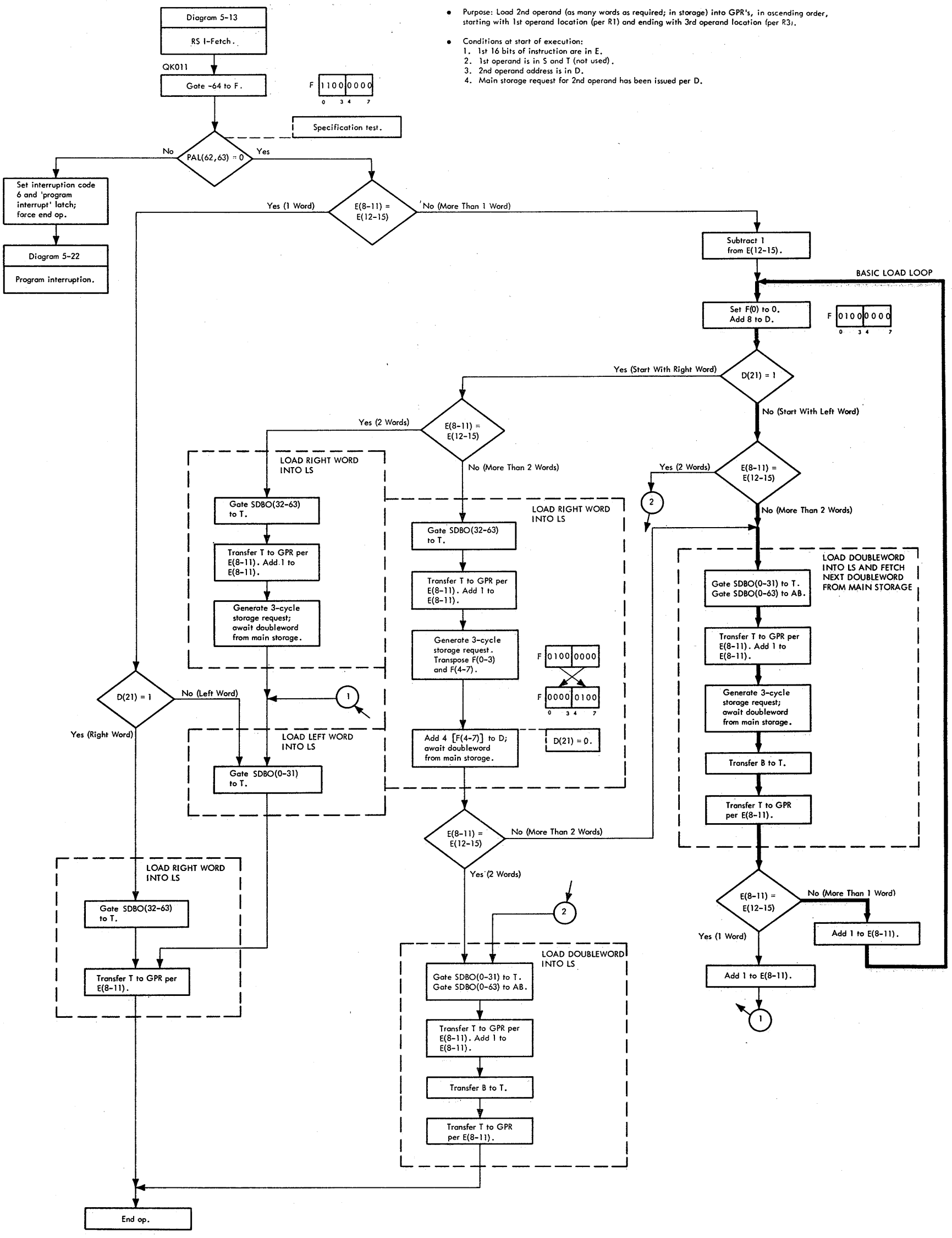
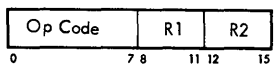


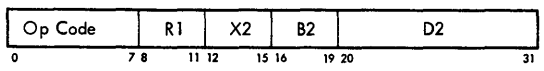
Diagram 5-107. Load Multiple, LM (98)

- RR format: AR, ALR, SR, SLR, CR.



- Purpose:
  1. AR, ALR, SR, SLR - Algebraically add (subtract) 2nd operand (in GPR, per R2) to (from) 1st operand (in GPR, per R1) and place result into 1st operand location.
  2. CR - Algebraically compare 1st operand (in GPR, per R1) with 2nd operand (in GPR, per R2) and set CC according to result.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D.
  3. 2nd operand is in S and T.
- Op codes:
  1. Add, AR = 1A.
  2. Add Logical, ALR = 1E.
  3. Subtract, SR = 1B.
  4. Subtract Logical, SLR = 1F.
  5. Compare, CR = 19.

- RX format: A, AH, AL, S, SH, SL, C, CH.



- Purpose:
  1. A, AL, S, SL - Algebraically add (subtract) 2nd operand (in storage) to (from) 1st operand (in GPR, per R1) and place result into 1st operand location.
  2. AH, SH - Algebraically add (subtract) halfword 2nd operand (in storage) to (from) 1st operand (in GPR, per R1) and place result into 1st operand location.
  3. C - Algebraically compare 1st operand (in GPR, per R1) with 2nd operand (in storage) and set CC according to result.
  4. CH - Algebraically compare 1st operand (in GPR, per R1) with halfword 2nd operand (in storage) and set CC according to result.
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D.
- Op codes:
  1. Add, A = 5A.
  2. Add Halfword, AH = 4A.
  3. Add Logical, AL = 5E.
  4. Subtract, S = 5B.
  5. Subtract Halfword, SH = 4B.
  6. Subtract Logical, SL = 5F.
  7. Compare, C = 59.
  8. Compare Halfword, CH = 49.

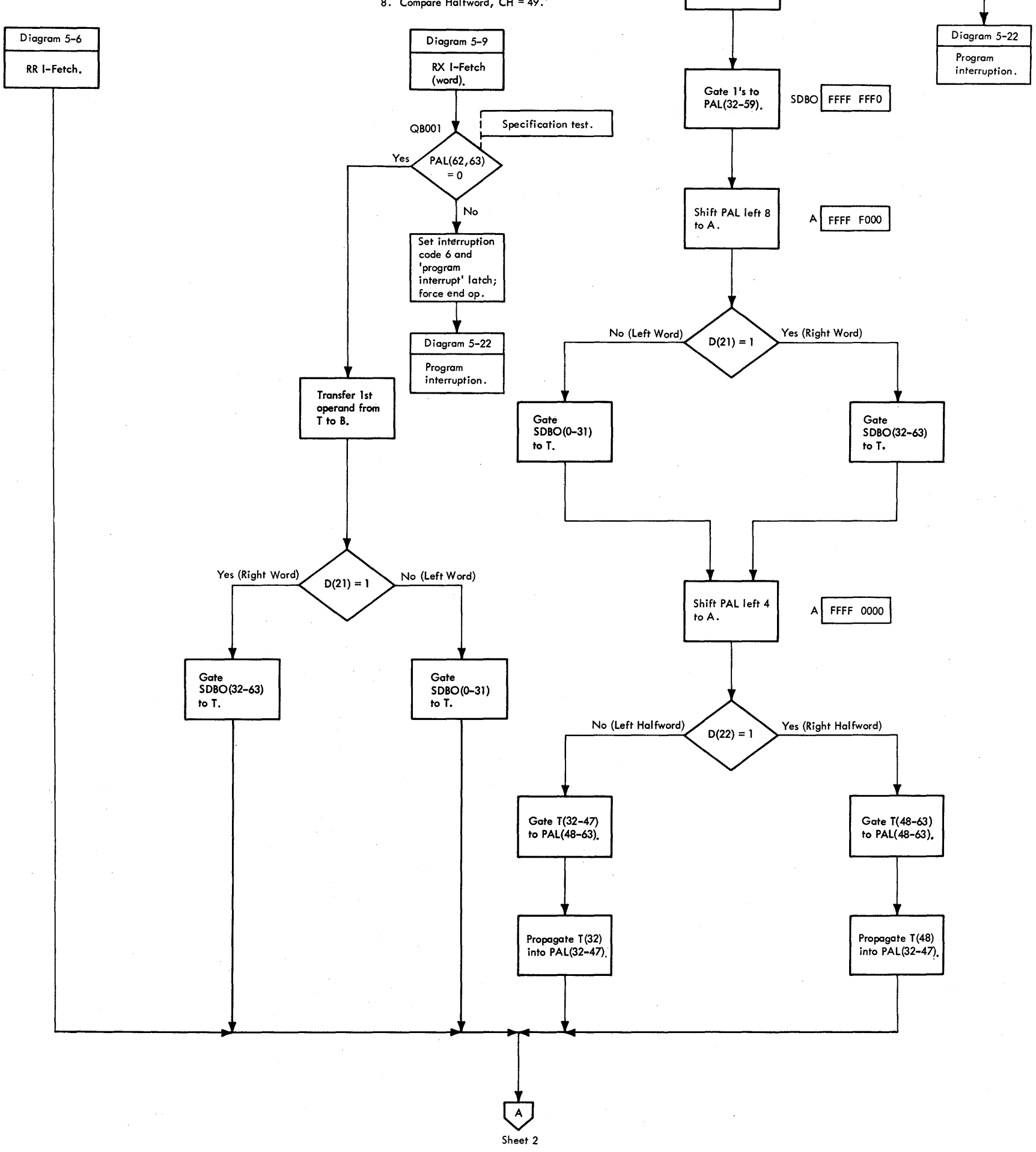
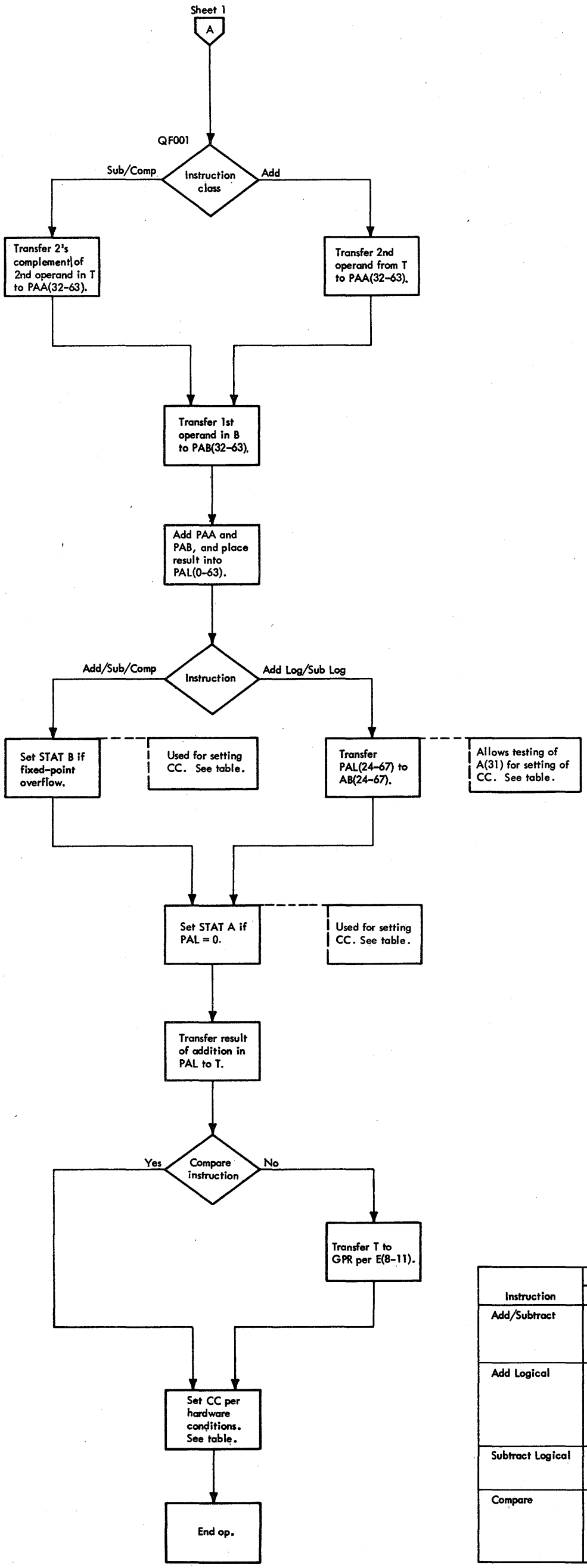


Diagram 5-108. Fixed-Point Add-Type Instructions (Sheet 1 of 2)

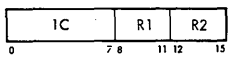


Instruction	STAT A	STAT B	State of Bit Tested	CC	Comments
Add/Subtract	1	0	T(32) = 0	0	PAL equals zero.
	0	0	T(32) = 1	1	PAL is less than zero.
	0	0	T(32) = 0	2	PAL is greater than zero.
	0	1	T(32) = 0	3	Overflow.
Add Logical	1	0	A(31) = 0	0	PAL equals zero and no carry from PAL(32).
	0	0	A(31) = 0	1	PAL does not equal zero and no carry from PAL(32).
	1	0	A(31) = 1	2	PAL equals zero and carry from PAL(32).
	0	0	A(31) = 1	3	PAL does not equal zero and carry from PAL(32).
Subtract Logical	0	0	A(31) = 0	1	Same as Add Logical.
	1	0	A(31) = 1	2	Same as Add Logical.
	0	0	A(31) = 1	3	Same as Add Logical.
Compare	1	0	T(32) = 0	0	Operands are equal.
	0	1	T(32) = 0	1	1st operand is less than 2nd operand.
	0	0	T(32) = 1	1	1st operand is less than 2nd operand.
	0	0	T(32) = 0	2	1st operand is greater than 2nd operand.
	0	1	T(32) = 1	2	1st operand is greater than 2nd operand.

Diagram 5-108. Fixed-Point Add-Type Instructions (Sheet 2 of 2)

Multiply (MR) 1C

- RR format:

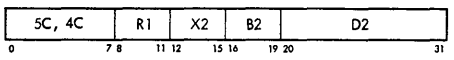


- Purpose: Multiply 1st operand (in GPR, per R1 + 1) by 2nd operand (in GPR, per R2) and place 64-bit result into 1st operand location (in GPR, per R1 and R1 + 1).

- Conditions at start of execution:
  1. Instruction is in E.
  2. Contents of even-address GPR specified by R1 is in A, B, and D (not used).
  3. Multiplicand (1st operand) is in odd-address GPR specified by R1 + 1.
  4. Multiplier (2nd operand) is in S and T.

Multiply (M), 5C and Multiply Halfword (MH), 4C

- RX format:



- Purpose:
  1. M - Multiply 1st operand (in GPR, per R1 + 1) and 2nd operand (in storage) and place 64-bit result into 1st operand location (in GPR, per R1 and R1 + 1).
  2. MH - Multiply 1st operand (in GPR, per R1) and halfword 2nd operand (in storage) and place low-order 32 bits of result into 1st operand location.

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. Contents of even-address GPR specified by R1 is in S and T for M instruction (not used).
  3. 1st operand for M instruction is in odd-address GPR specified by R1 + 1.
  4. 1st operand for MH instruction is in S and T.
  5. 2nd operand address is in D.
  6. Main storage request for 2nd operand has been issued per D.

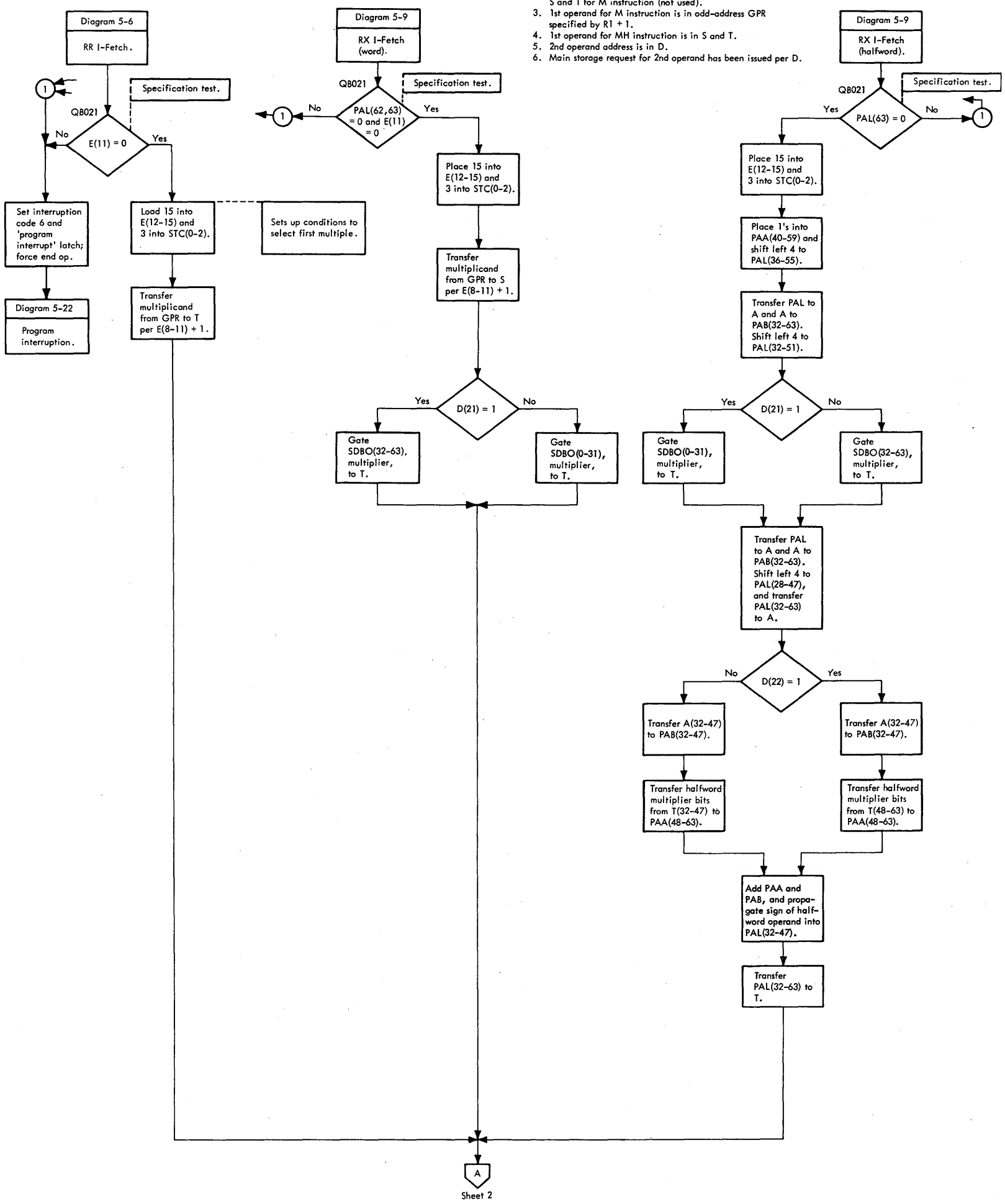
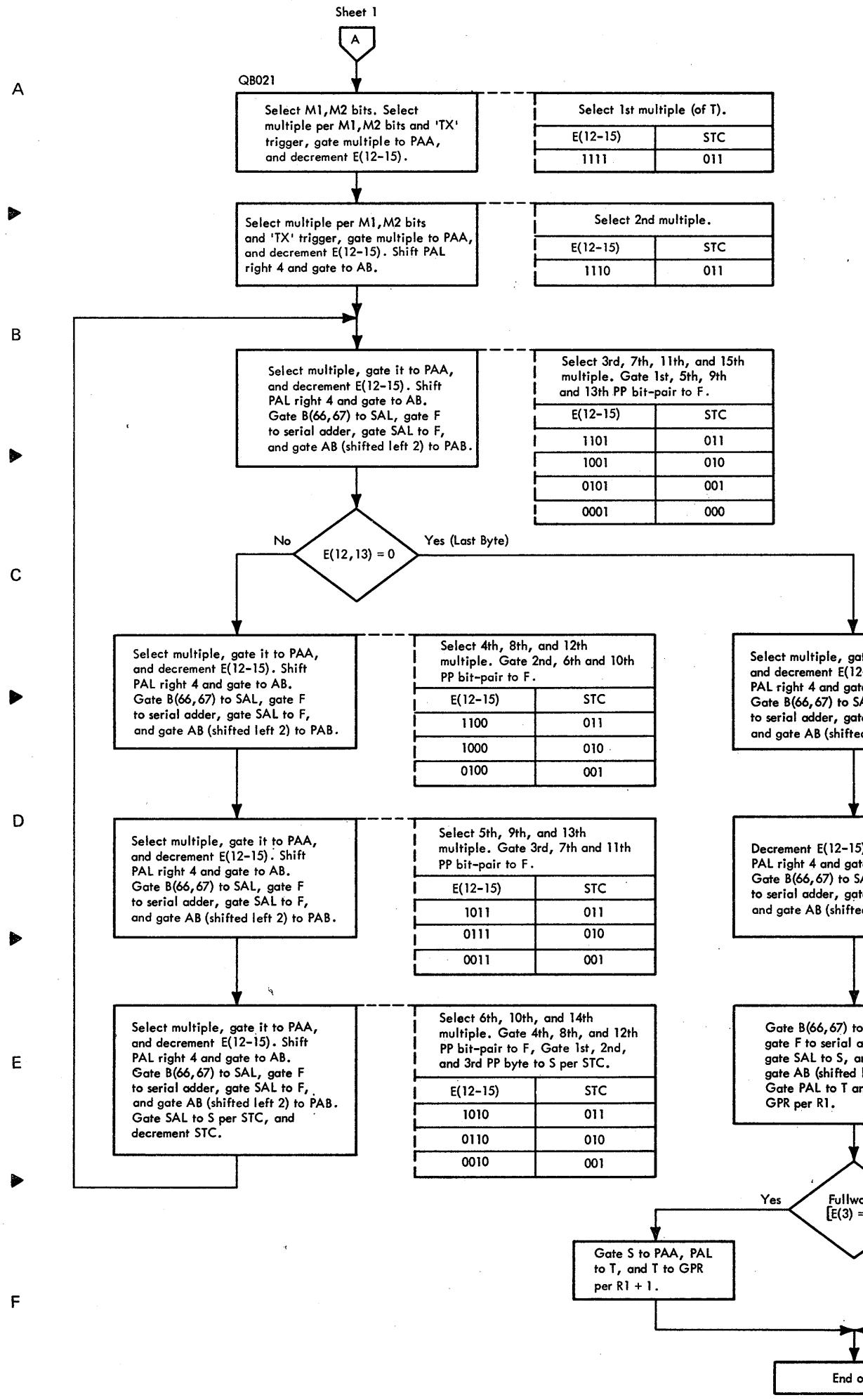


Diagram 5-109. Fixed-Point Multiply (Sheet 1 of 3)



Objectives:

1. Select multiple (of T).
   a. Select M1,M2 bits from S per E(12-15) (A of Sheet 3).
   b. Select multiple (of T) per M1,M2 bits and 'TX' trigger (Table 1, Sheet 3).

2. Develop partial product (PP) bit-pair.
   a. Add multiple to PP in AB (shifted right 2) (B of Sheet 3).
   b. Gate new PP to AB.
   c. Gate PP bit-pair [B(66,67)] to SAL per E(14,15) (C of Sheet 3).

3. Develop PP byte.
   Add SAL to F.

4. Develop low-order PP in S.
   a. When PP byte is complete, gate SAL to S per STC.
   b. When S is full (4 bytes of PP has been loaded), low-order product is in S and high-order product is in PAL.

5. Store product.
   a. Store high-order product (in PAL) into GPR per R1 (even register).
   b. Store low-order product (in S) into GPR per R1 + 1 (odd register) (per R1 for MH instruction).

Diagram 5-109. Fixed-Point Multiply (Sheet 2 of 3)

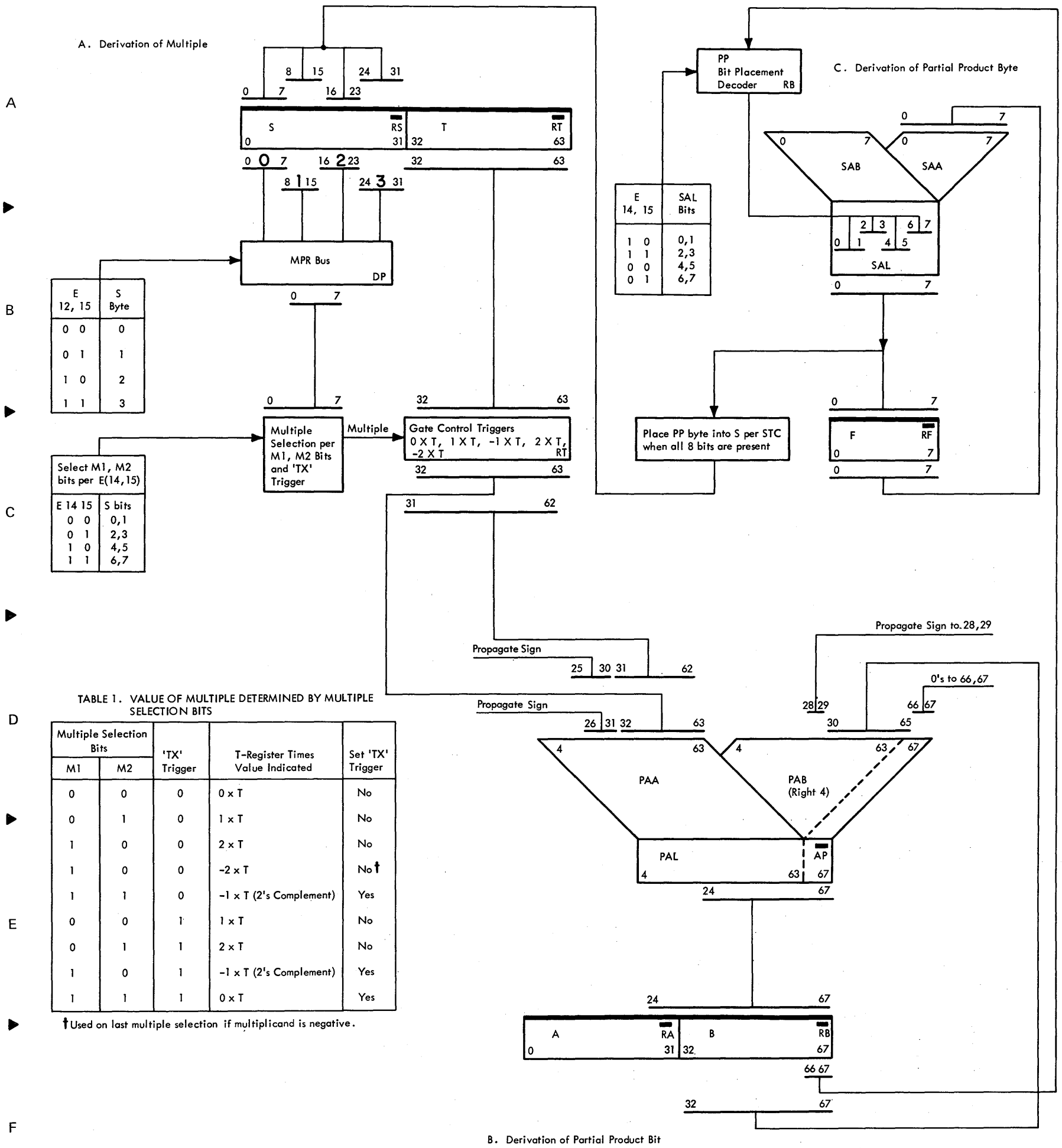
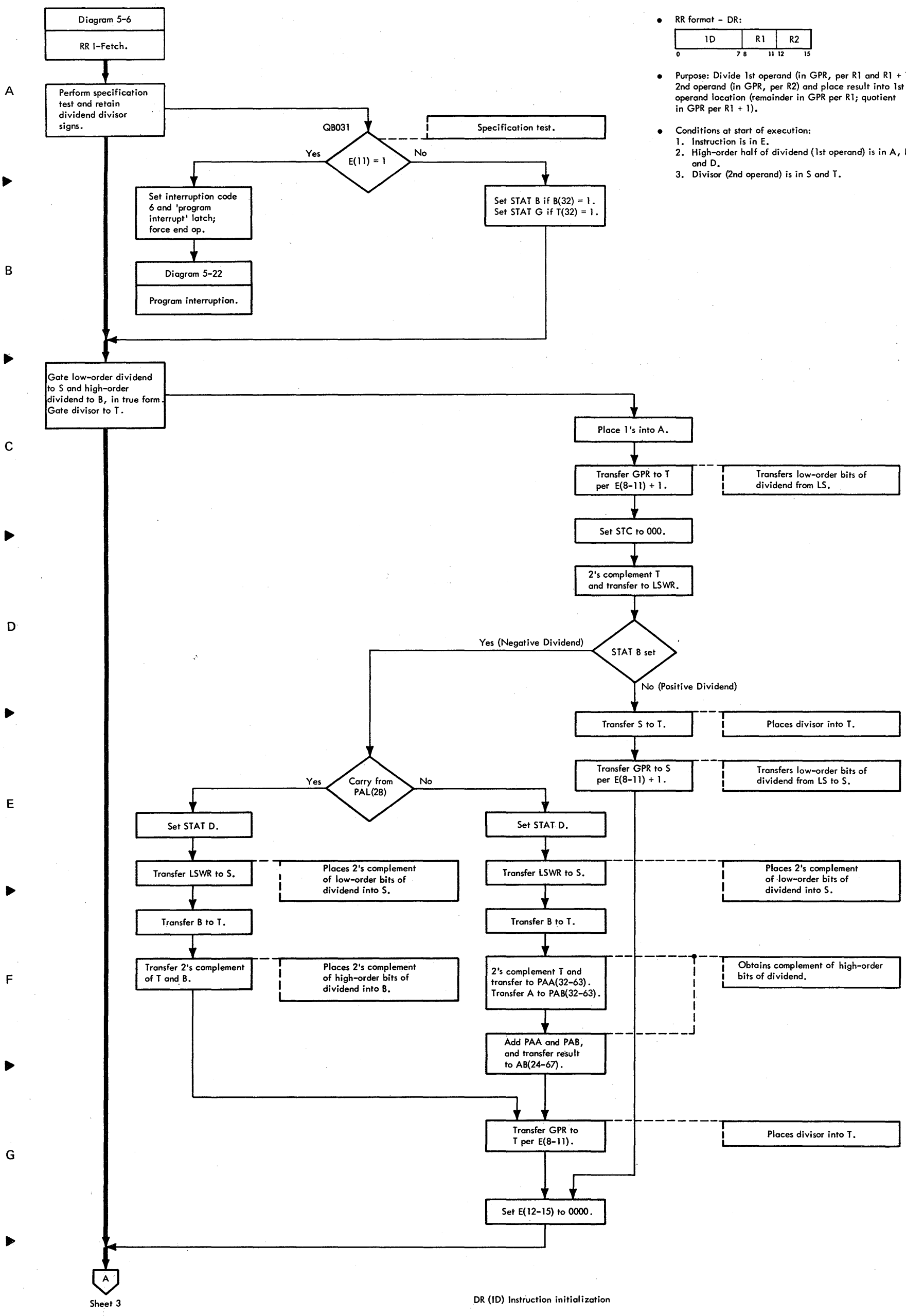


TABLE 1. VALUE OF MULTIPLE DETERMINED BY MULTIPLE SELECTION BITS

Multiple Selection Bits		'TX' Trigger	T-Register Times Value Indicated	Set 'TX' Trigger
M1	M2			
0	0	0	0 x T	No
0	1	0	1 x T	No
1	0	0	2 x T	No
1	0	0	-2 x T	No†
1	1	0	-1 x T (2's Complement)	Yes
0	0	1	1 x T	No
0	1	1	2 x T	No
1	0	1	-1 x T (2's Complement)	Yes
1	1	1	0 x T	Yes

†Used on last multiple selection if multiplicand is negative.

Diagram 5-109. Fixed-Point Multiply (Sheet 3 of 3)

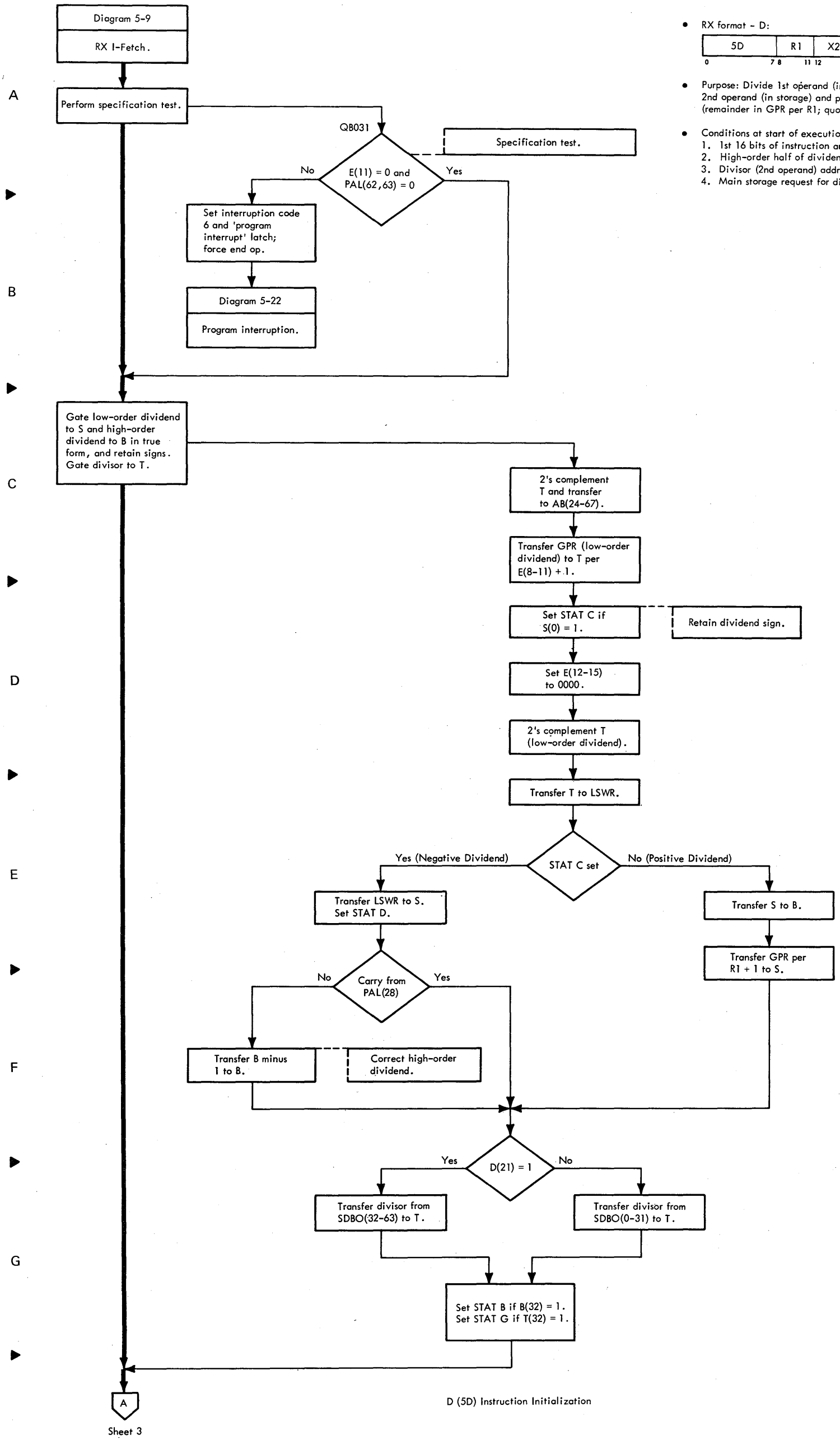


- RR format - DR:

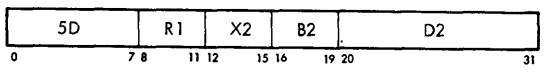
- Purpose: Divide 1st operand (in GPR, per R1 and R1 + 1) by 2nd operand (in GPR, per R2) and place result into 1st operand location (remainder in GPR per R1; quotient in GPR per R1 + 1).

- Conditions at start of execution:
  1. Instruction is in E.
  2. High-order half of dividend (1st operand) is in A, B, and D.
  3. Divisor (2nd operand) is in S and T.

Diagram 5-110. Fixed-Point Divide (Sheet 1 of 6)



- RX format - D:



- Purpose: Divide 1st operand (in GPR, per R1 and R1 + 1) by 2nd operand (in storage) and place result into 1st operand location (remainder in GPR per R1; quotient in GPR per R1 + 1).

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. High-order half of dividend (1st operand) is in S and T.
  3. Divisor (2nd operand) address is in D.
  4. Main storage request for divisor has been issued per D.

Diagram 5-110. Fixed-Point Divide (Sheet 2 of 6)

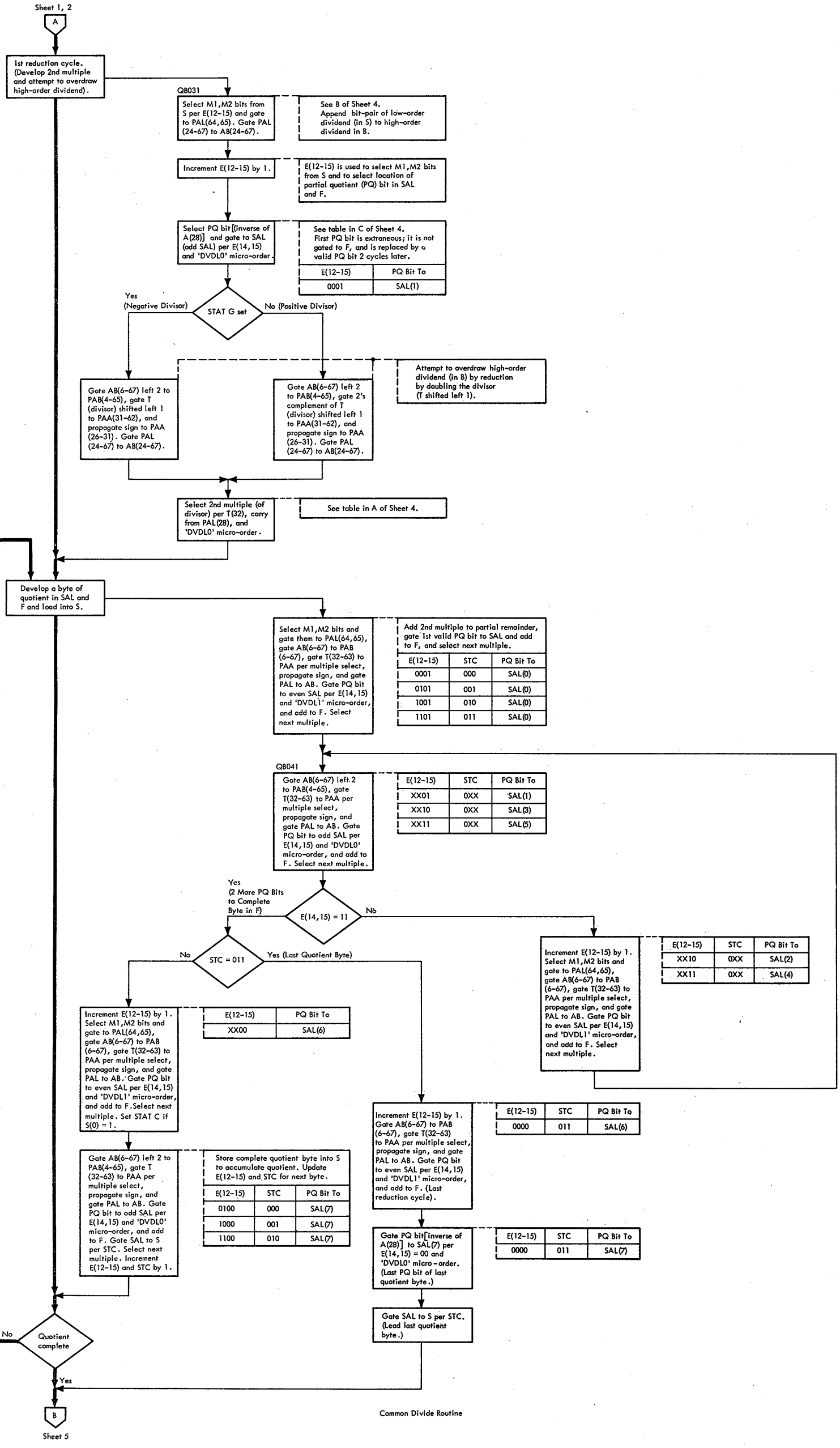


Diagram 5-110. Fixed-Point Divide (Sheet 3 of 6)

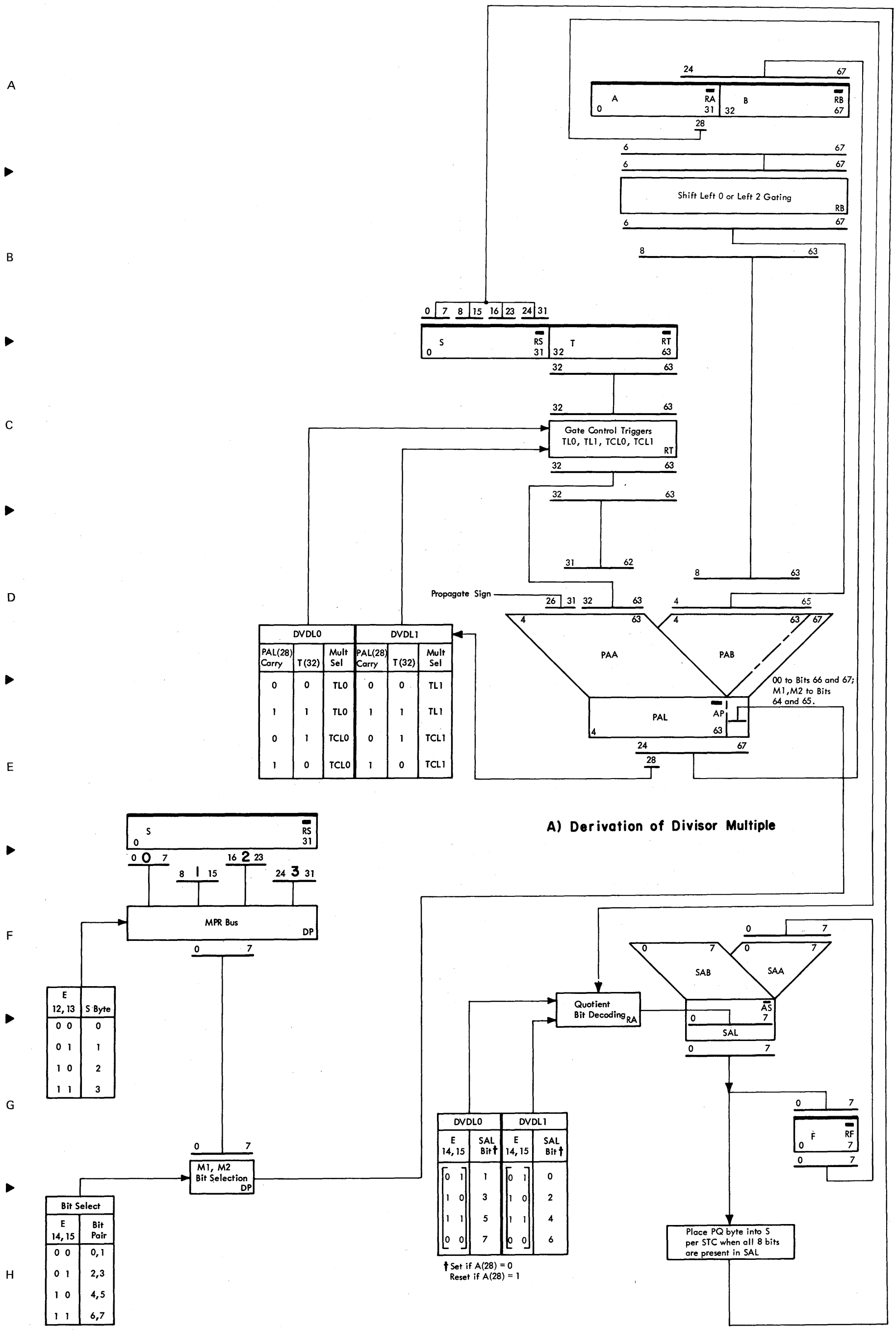
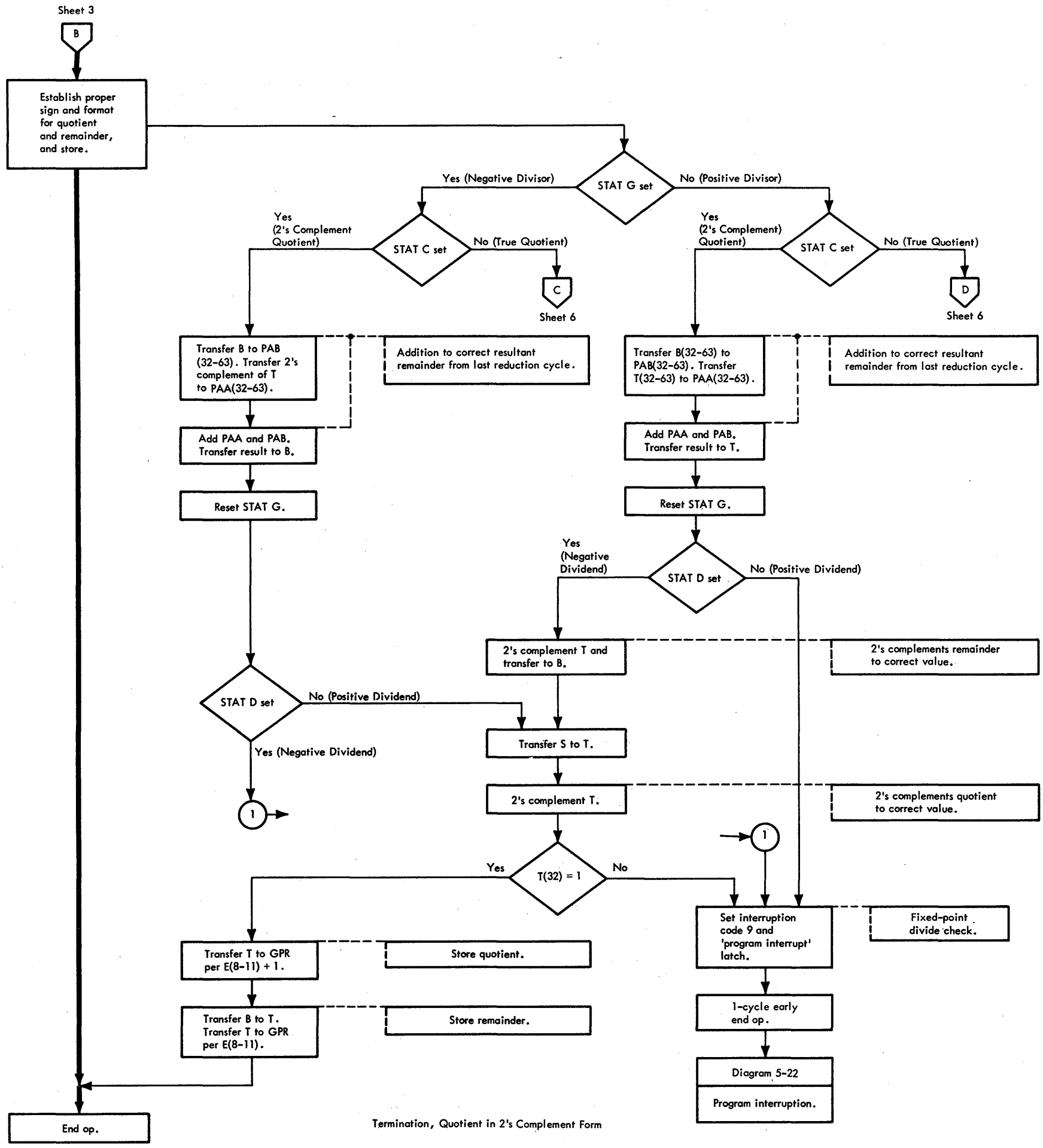


Diagram 5-110. Fixed-Point Divide (Sheet 4 of 6)



Termination, Quotient in 2's Complement Form

Diagram 5-110. Fixed-Point Divide (Sheet 5 of 6)

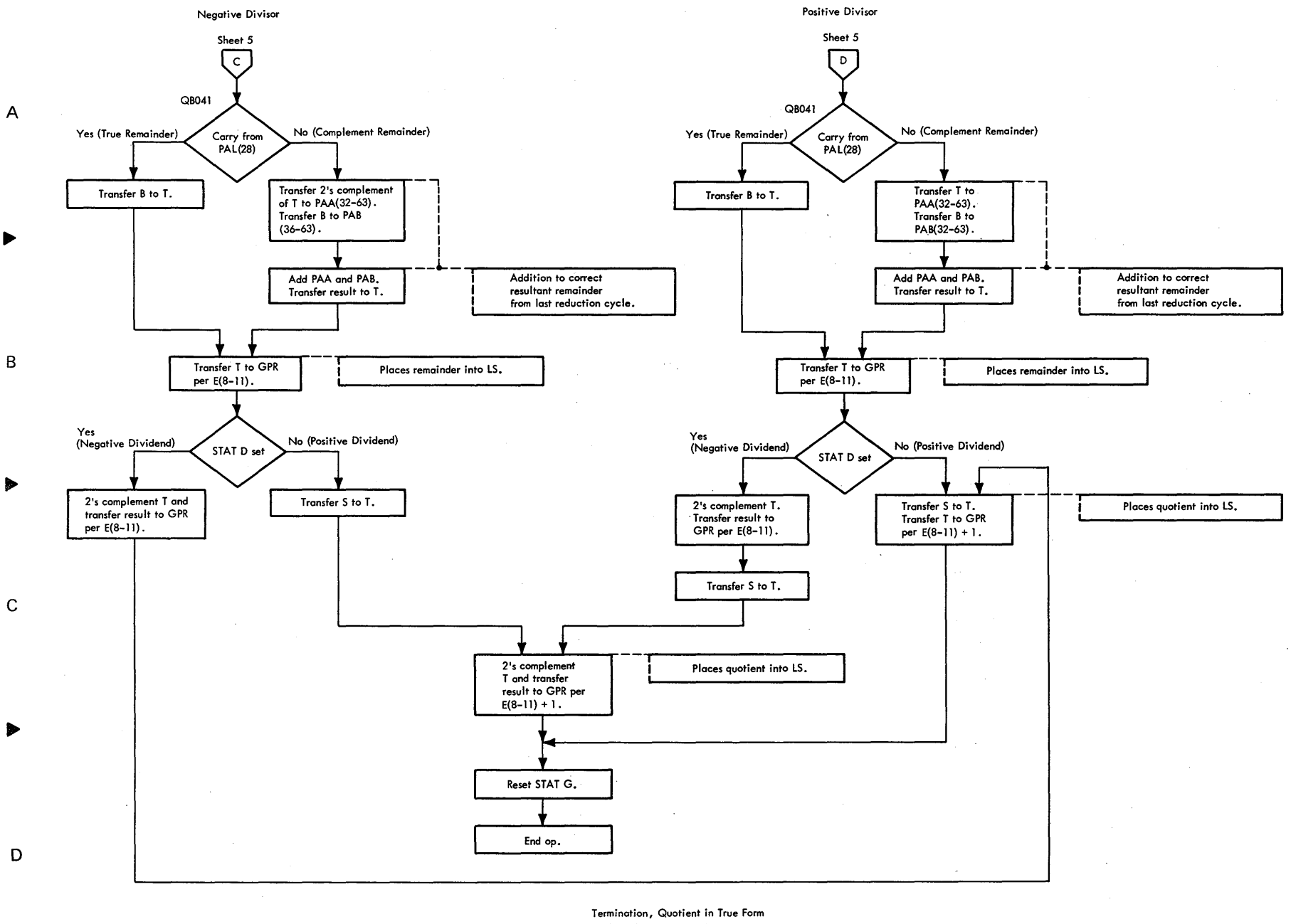


Diagram 5-110. Fixed-Point Divide (Sheet 6 of 6)

- RX format:

4F	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

- Purpose: Convert radix of 2nd operand (in storage) from decimal to binary and place result into 1st operand location (in GPR, per R1).

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T (not used).
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D.

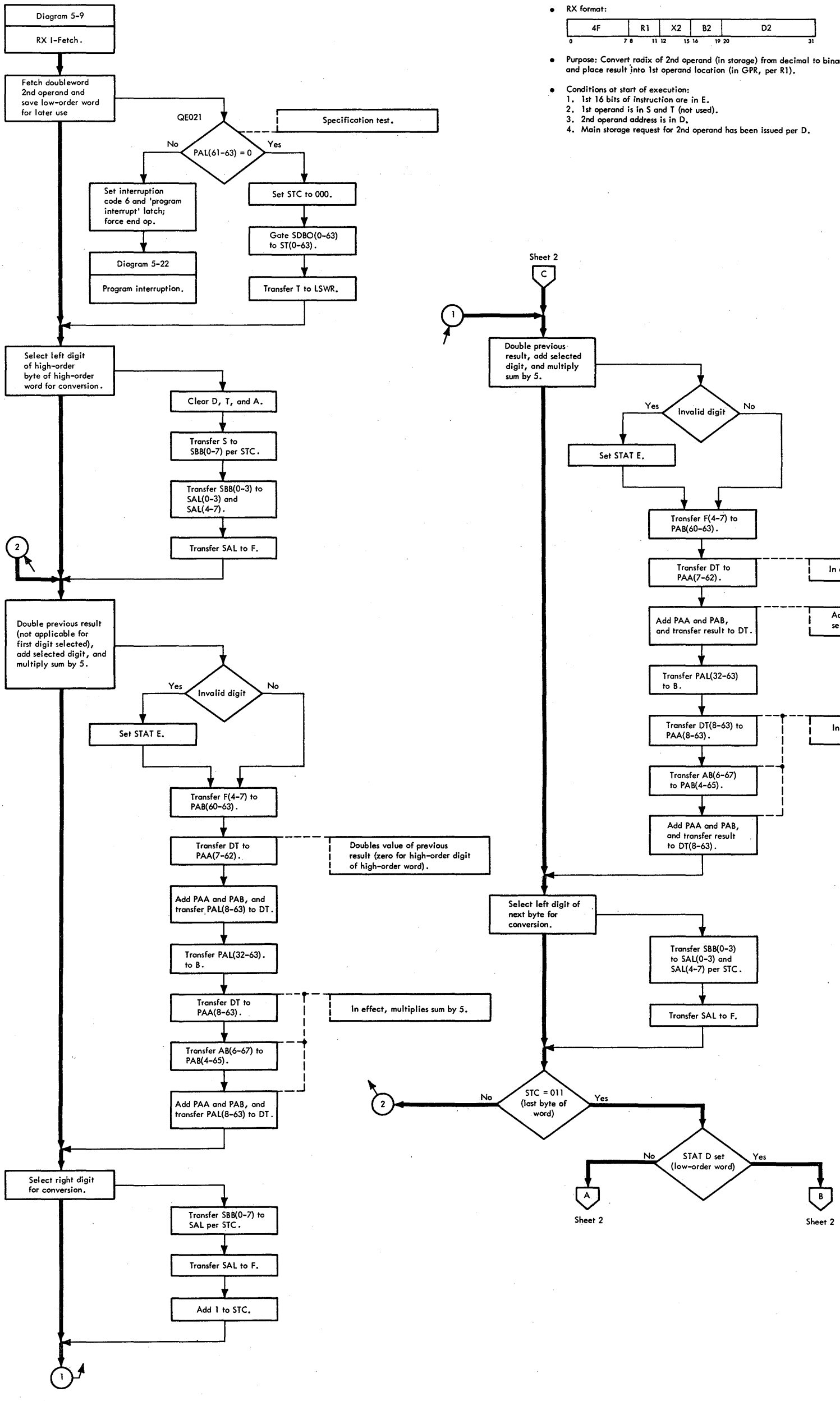


Diagram 5-111. Convert to Binary, CVB (4F) (Sheet 1 of 2)

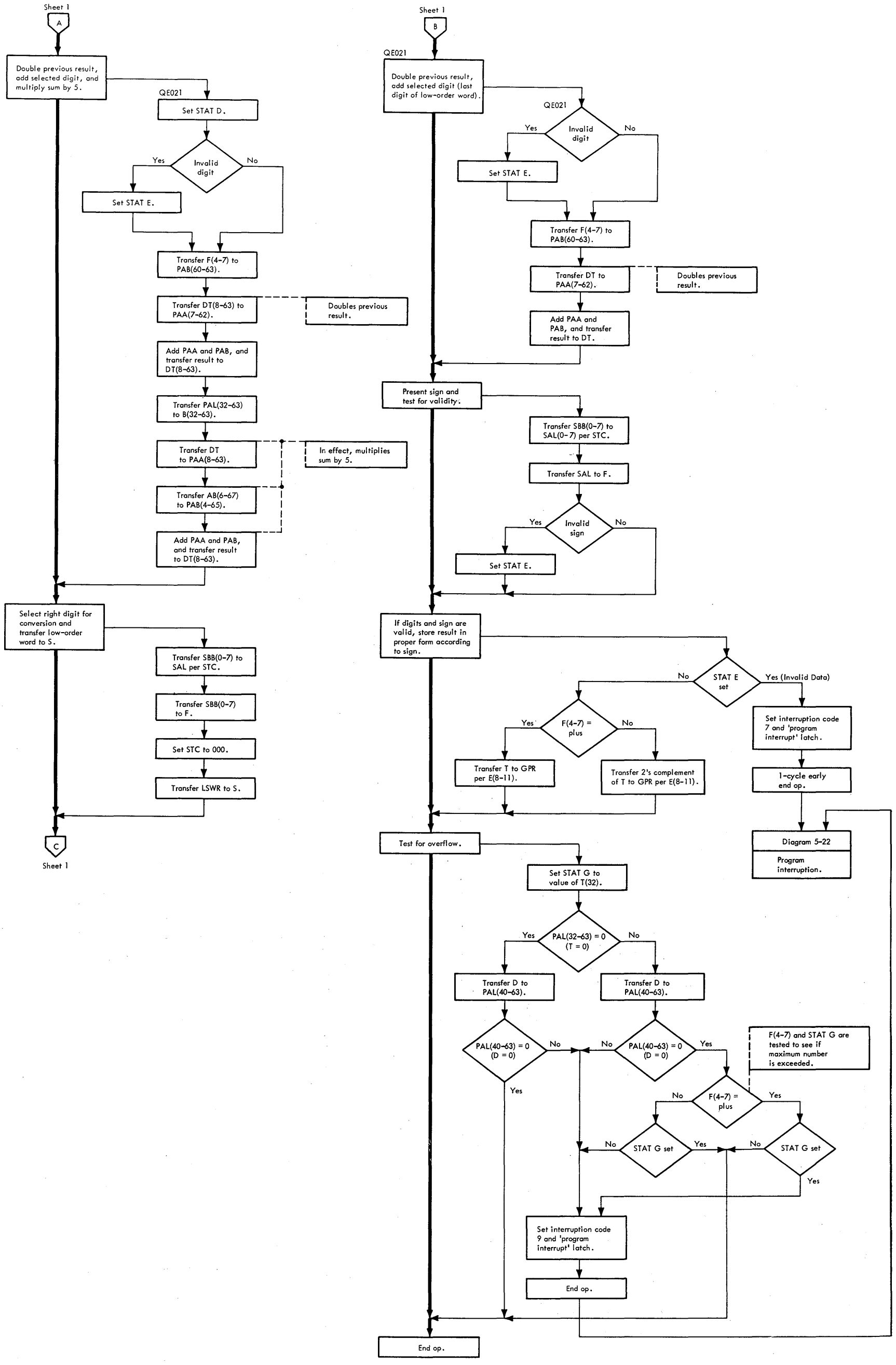
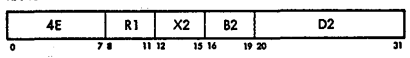


Diagram 5-111. Convert to Binary, CVG (4F) (Sheet 2 of 2)

- RX format:



- Convert radix of 1st operand (in GPR, per R1) from binary to decimal and place result into 2nd operand location (in storage).

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D (not used).

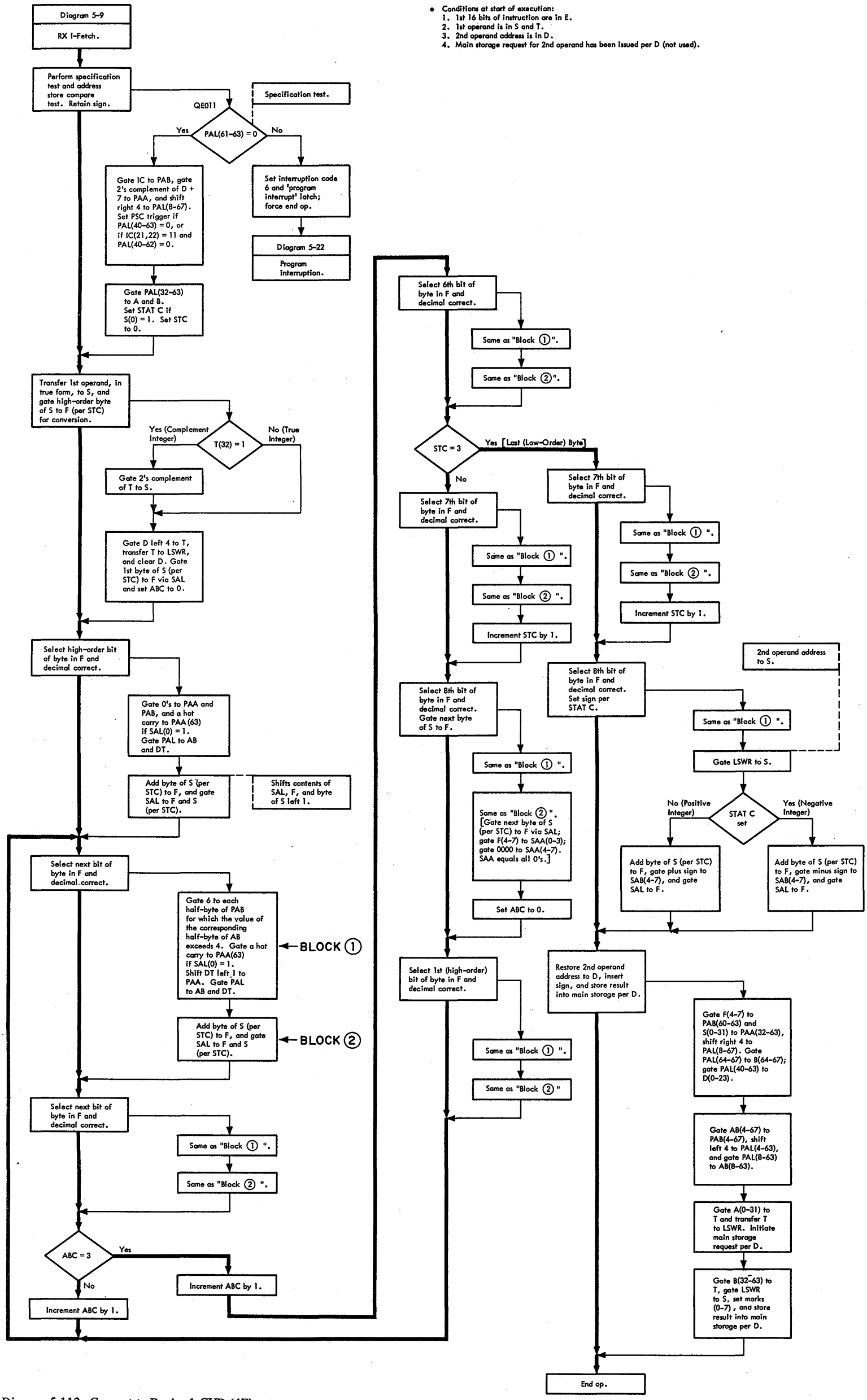
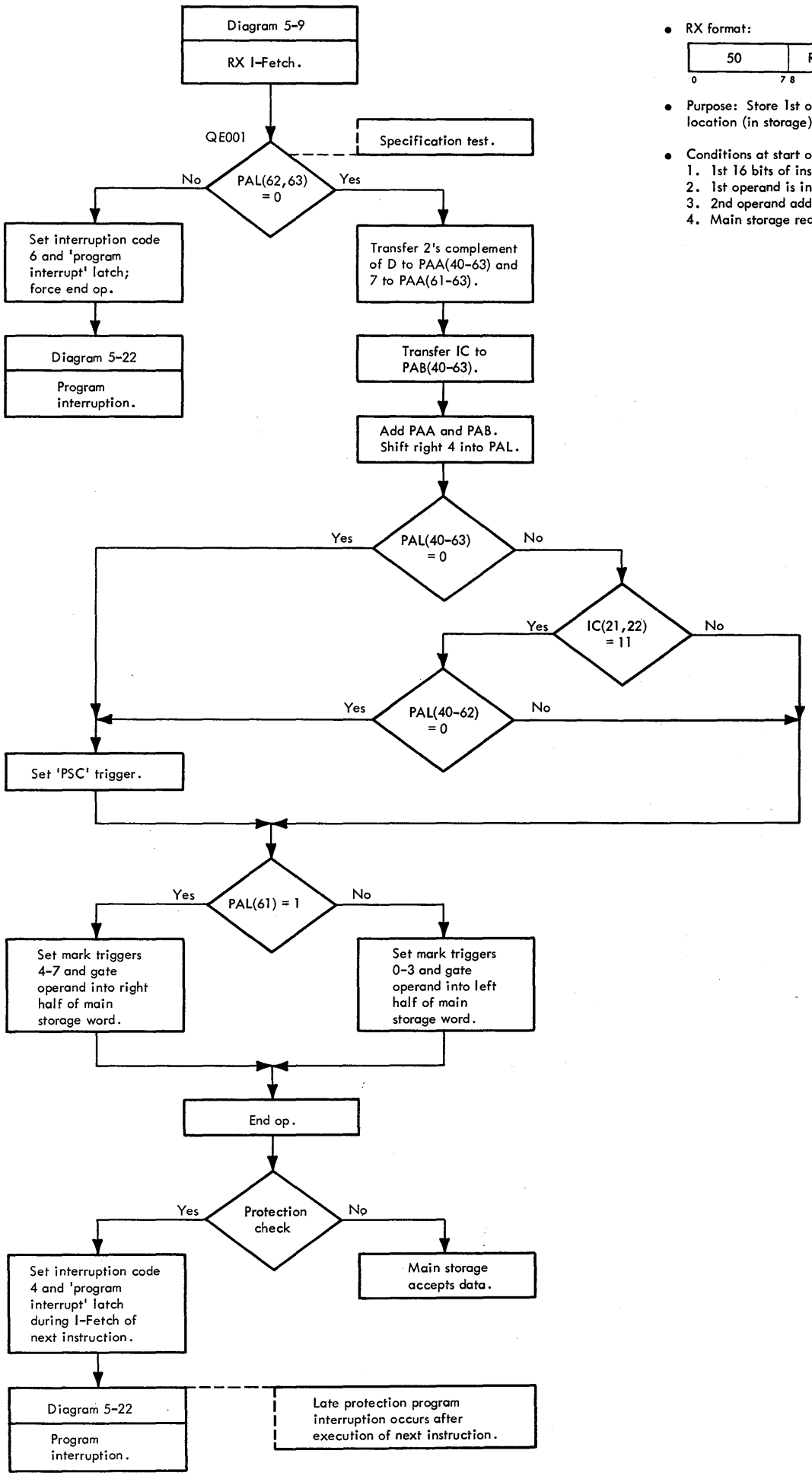
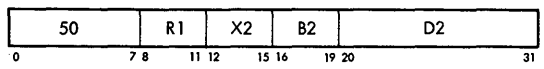


Diagram 5-112. Convert to Decimal, CVD (4E)

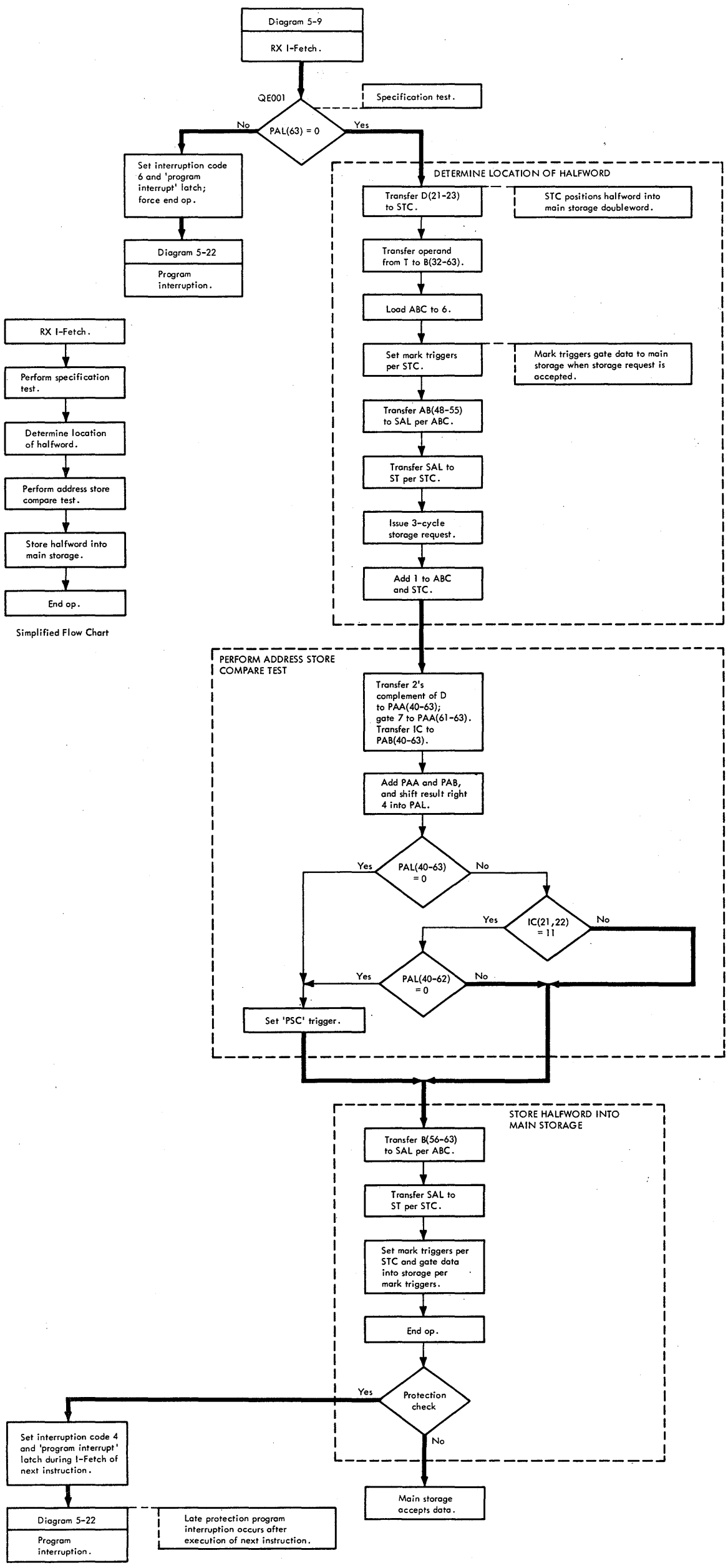


- RX format:

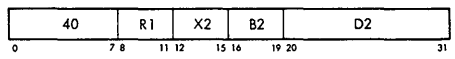


- Purpose: Store 1st operand (in GPR, per R1) into 2nd operand location (in storage).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D (not used).

Diagram 5-113. Store, ST (50)

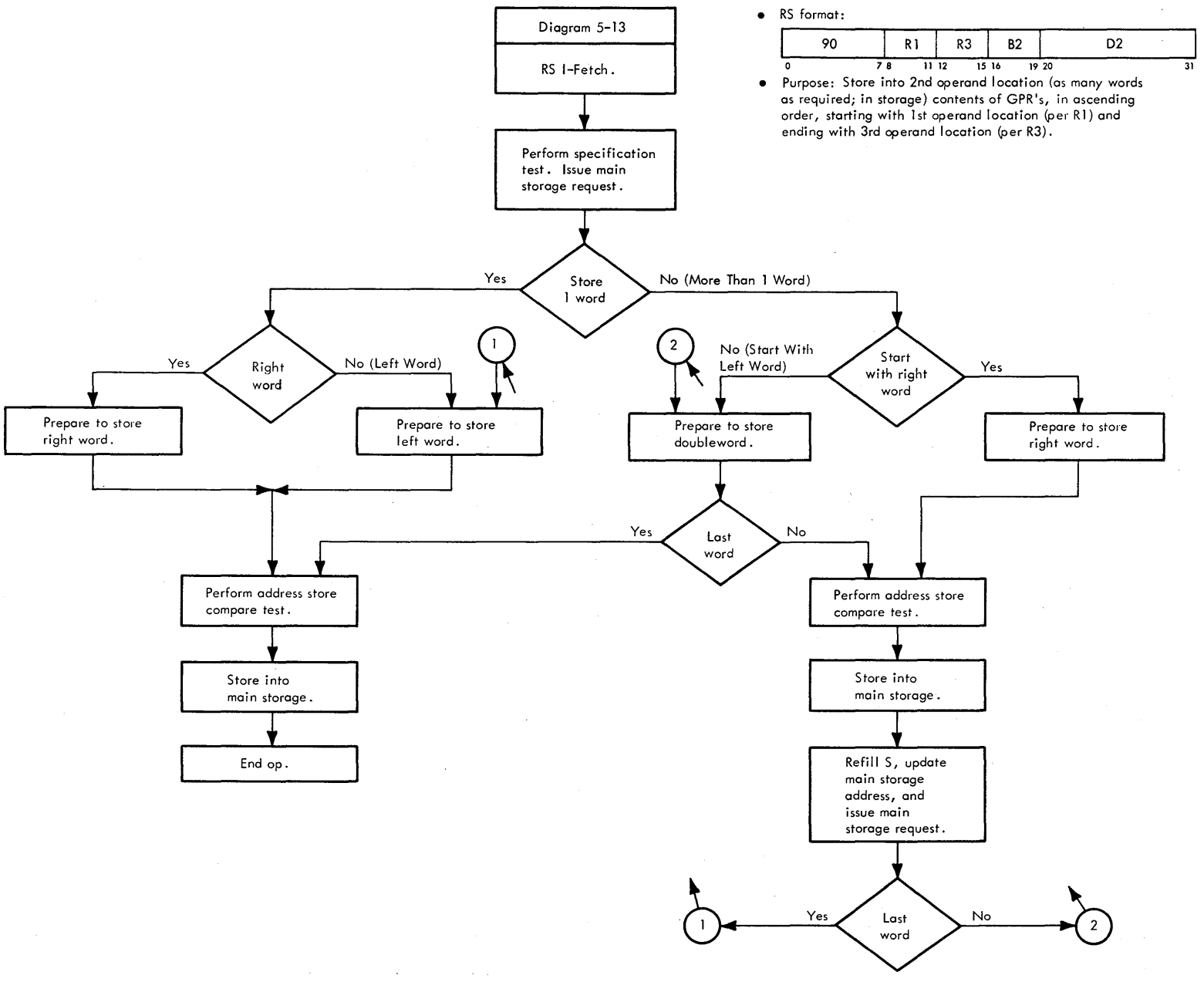


- RX format:



- Purpose: Store halfword 1st operand (in GPR, per R1) into 2nd operand location (in storage).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D (not used).

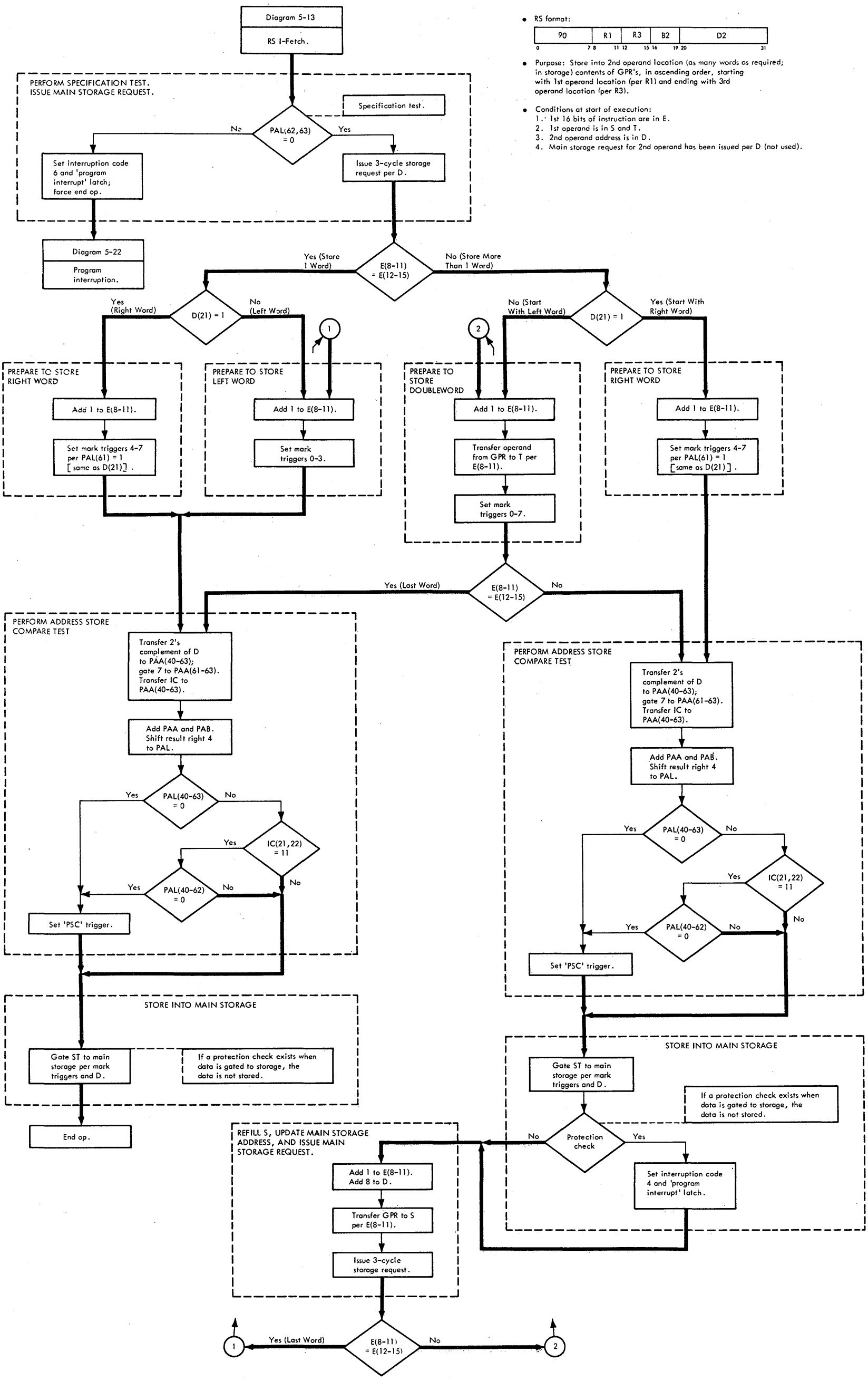
Diagram 5-114. Store Halfword, STH (40)



- RS format:

- Purpose: Store into 2nd operand location (as many words as required; in storage) contents of GPR's, in ascending order, starting with 1st operand location (per R1) and ending with 3rd operand location (per R3).

Diagram 5-115. Store Multiple, STM (90) (Sheet 1 of 2)

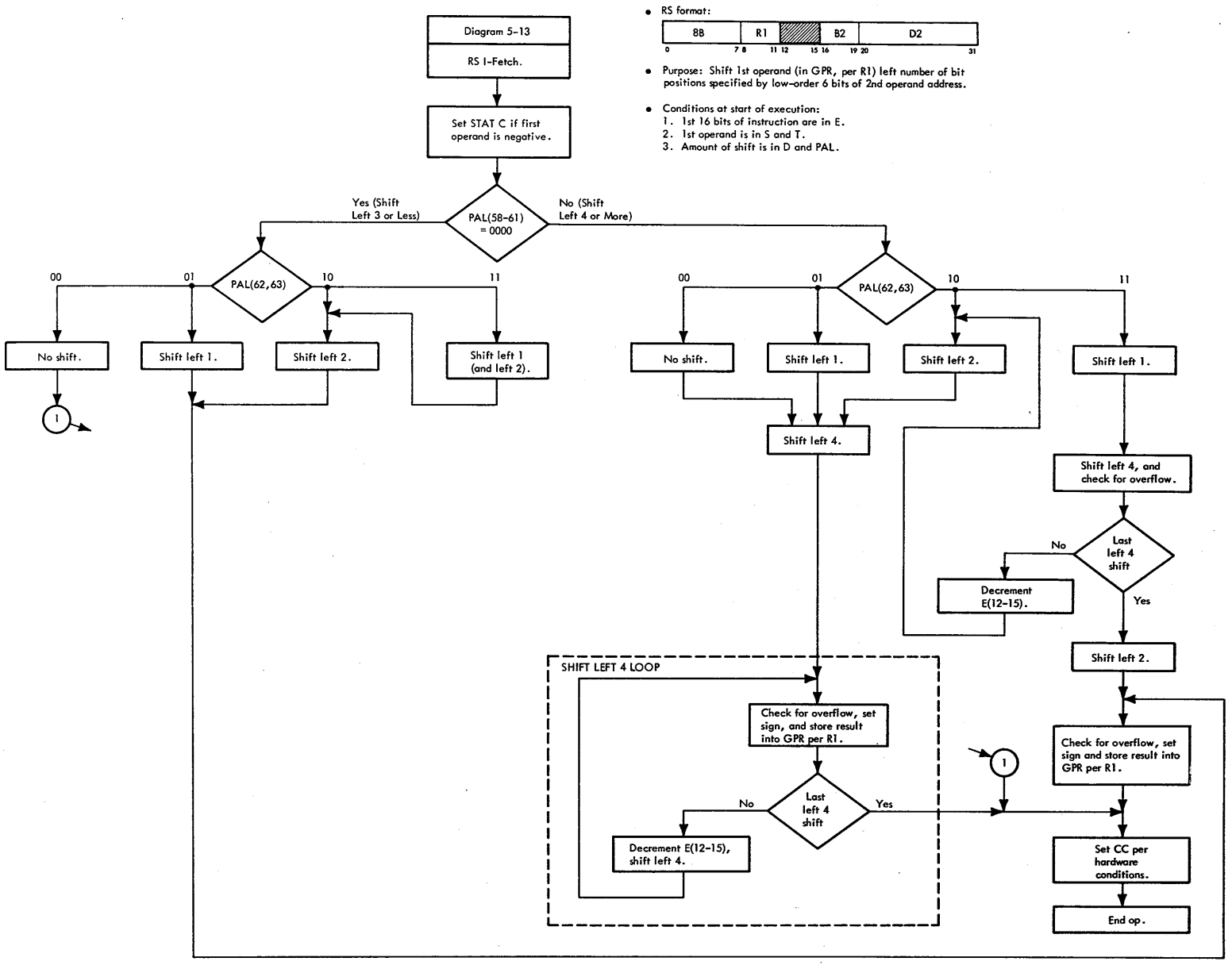


- RS format:

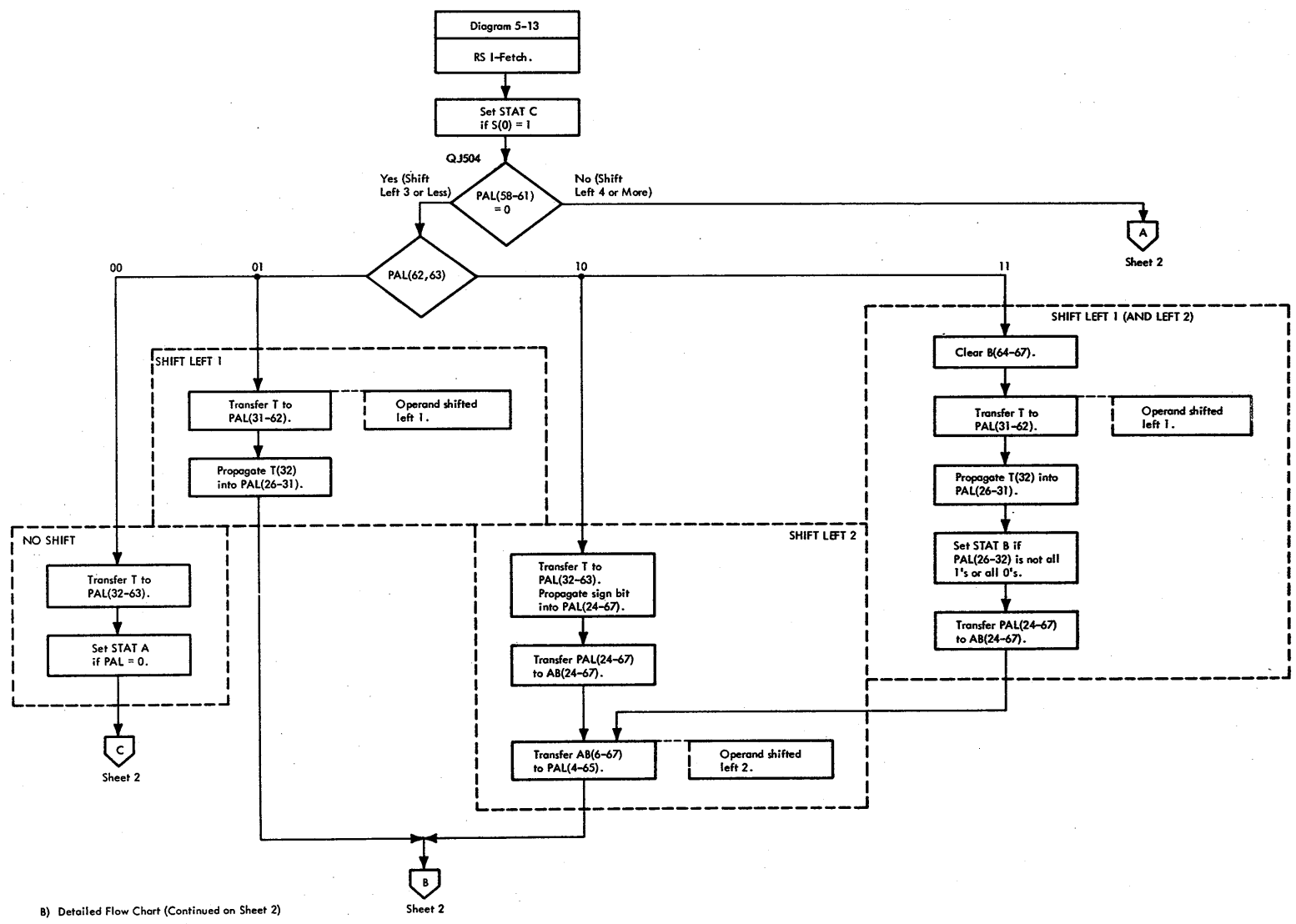
90	R1	R3	B2	D2
0 7	8 11	12 15	16 19	20 31

- Purpose: Store into 2nd operand location (as many words as required; in storage) contents of GPR's, in ascending order, starting with 1st operand location (per R1) and ending with 3rd operand location (per R3).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. 2nd operand address is in D.
  4. Main storage request for 2nd operand has been issued per D (not used).

Diagram 5-115. Store Multiple, STM (90) (Sheet 2 of 2)



A) Simplified Flow Chart



B) Detailed Flow Chart (Continued on Sheet 2)

Diagram 5-116. Shift Left Single, SLA (8B) (Sheet 1 of 2)

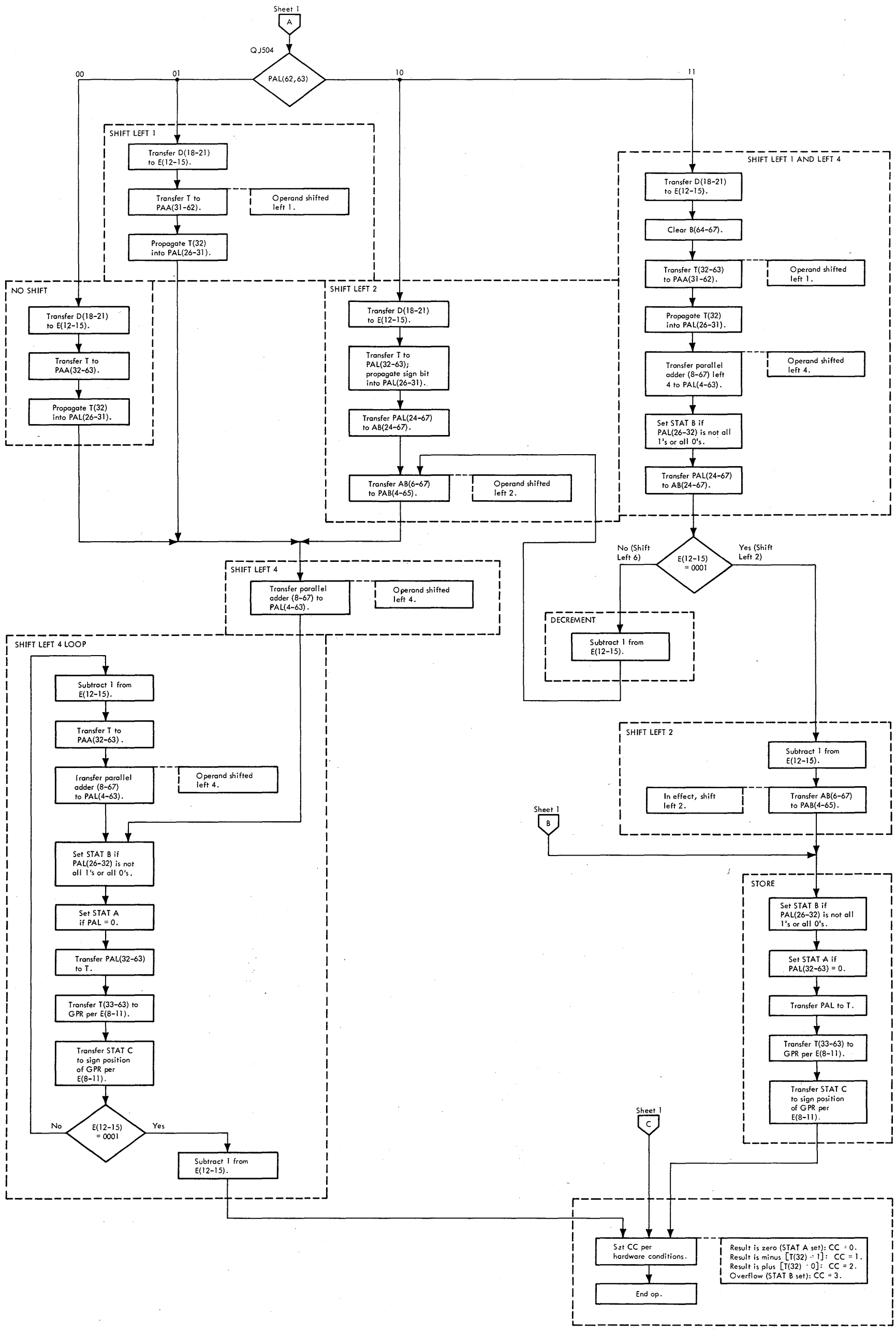
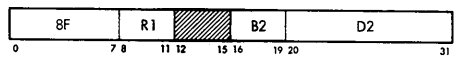


Diagram 5-116. Shift Left Single, SLA (8B) (Sheet 2 of 2)

Diagram 5-13
RS I-Fetch.

- RS format:



- Purpose: Shift 1st operand (in GPR, per R1 and R1 + 1) left number of bit positions specified by low-order 6 bits of 2nd operand address.

Specification check
Yes
Specification program interruption.
No
Set STAT C if first operand is negative. Transfer low-order word to S and T.

PAL(62,63)
00
PAL(58-61) = 0
Yes (No Shift)
Test low-order word for zeros. Transfer high-order word to T.
Test high-order word for zeros.
No (Shift Left 4, Left 8, and So On)
Shift low-order word left 4; gate overflow bits to B(64-67); transfer high-order word to T.

01
PAL(58-61) = 0
Yes (Shift Left 1)
Shift low-order word left 1; gate overflow bit to B(64-67); store low-order word into GPR per R1 + 1.
Shift overflow bit left 4, and transfer high-order word to T.
Shift high-order word left 1, add overflow bit, test for overflow from high-order word, and test for zeros.
1
No (Shift Left 5, Left 9, and So On)
Shift low-order word left 1; gate overflow bit to B(64-67); store low-order word into GPR per R1 + 1.
Shift overflow bit left 4, and transfer high-order word to T.
Shift high-order word left 1, add overflow bit, and test for overflow from high-order word.
Transfer low-order word to S.

10
(Shift Left 2, Left 6, and So On)
Shift low-order word left 2; gate overflow bits to B(64-67); store low-order word into GPR per R1 + 1.
Shift overflow bits left 2.
Transfer high-order word to T.
Transfer low-order word to S.

11
(Shift Left 3, Left 7, and So On)
Shift low-order word left 1.
Shift low-order word left 2; gate overflow bits to B(64-67); store low-order word into GPR per R1 + 1.
Transfer high-order word to T.
Shift high-order word left 1, shift overflow bits left 2, and transfer low-order word to S.

Shift high-order word and overflow bits left 2; test for overflow from high-order word and test for zeros.
Last left 4 shift
No
Shift low-order word left 4; gate overflow bits to B(64-67).
Yes
1
Insert sign per STAT C; store high-order word into GPR per R1.

SHIFT LEFT 4 LOOP
Shift high-order word and overflow bits left 4, insert sign per STAT C, test for overflow from high-order word, test for zeros, and store high-order word into GPR per R1.
Last left 4 shift
Yes
Store low-order word into GPR per R1 + 1, and test for zeros.
No
Shift low-order word left 4, gate overflow bits to B(64-67), and decrement left 4 count.

Set CC.
End op.

Diagram 5-117. Shift Left Double, SLDA (8F) (Sheet 1 of 4)

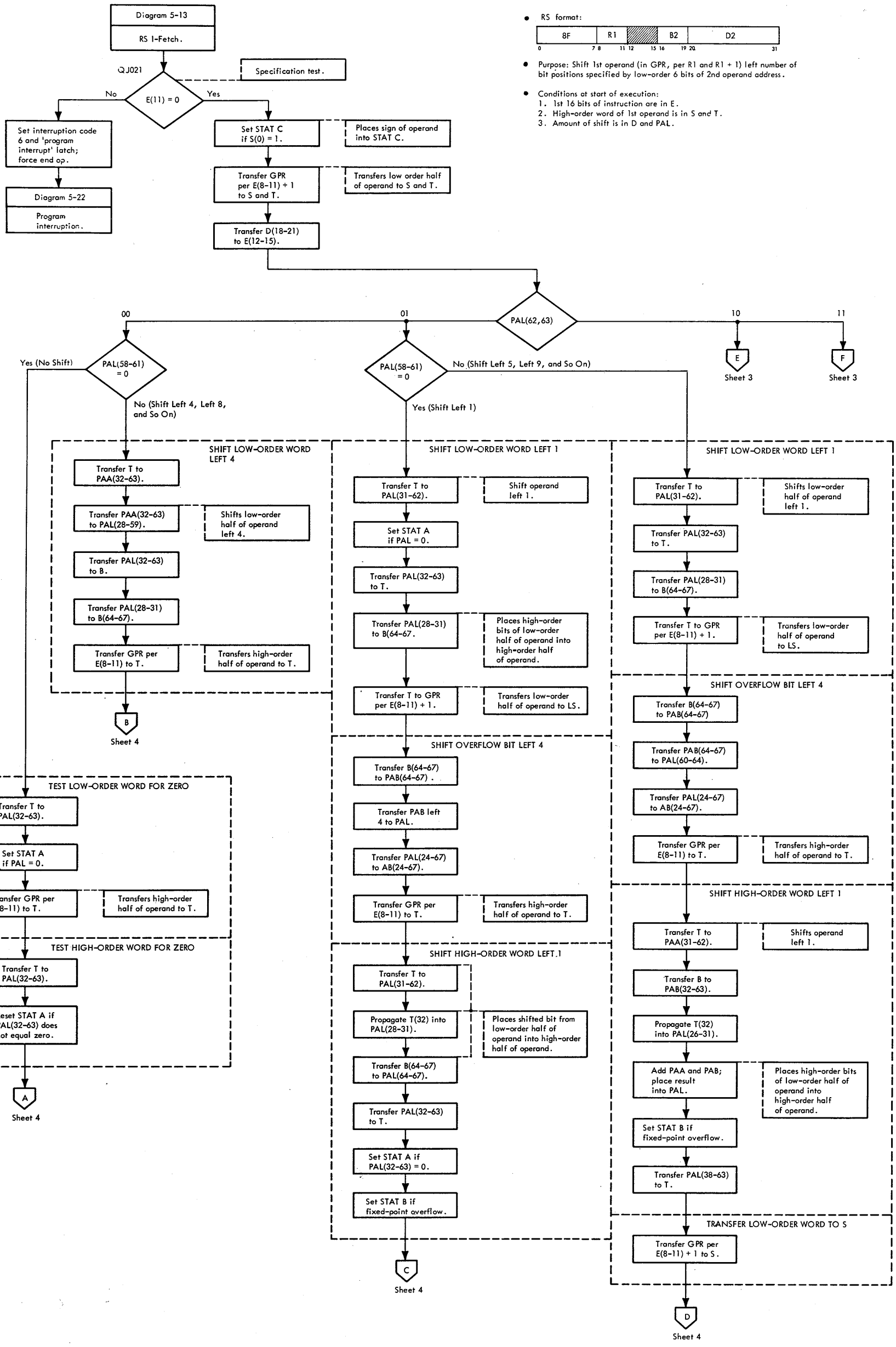


Diagram 5-117. Shift Left Double, SLDA (8F) (Sheet 2 of 4)

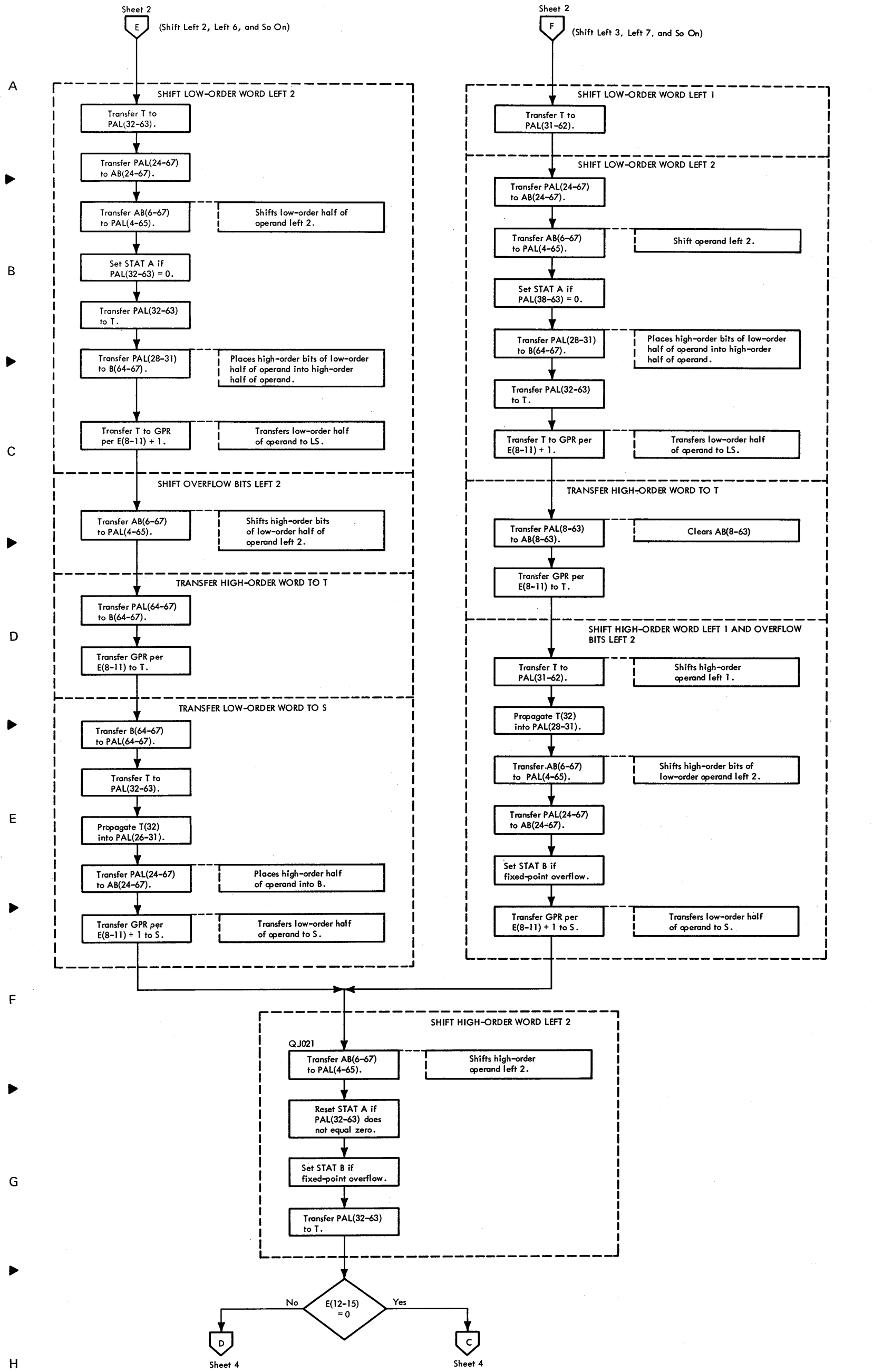


Diagram 5-117. Shift Left Double, SLDA (8F) (Sheet 3 of 4)

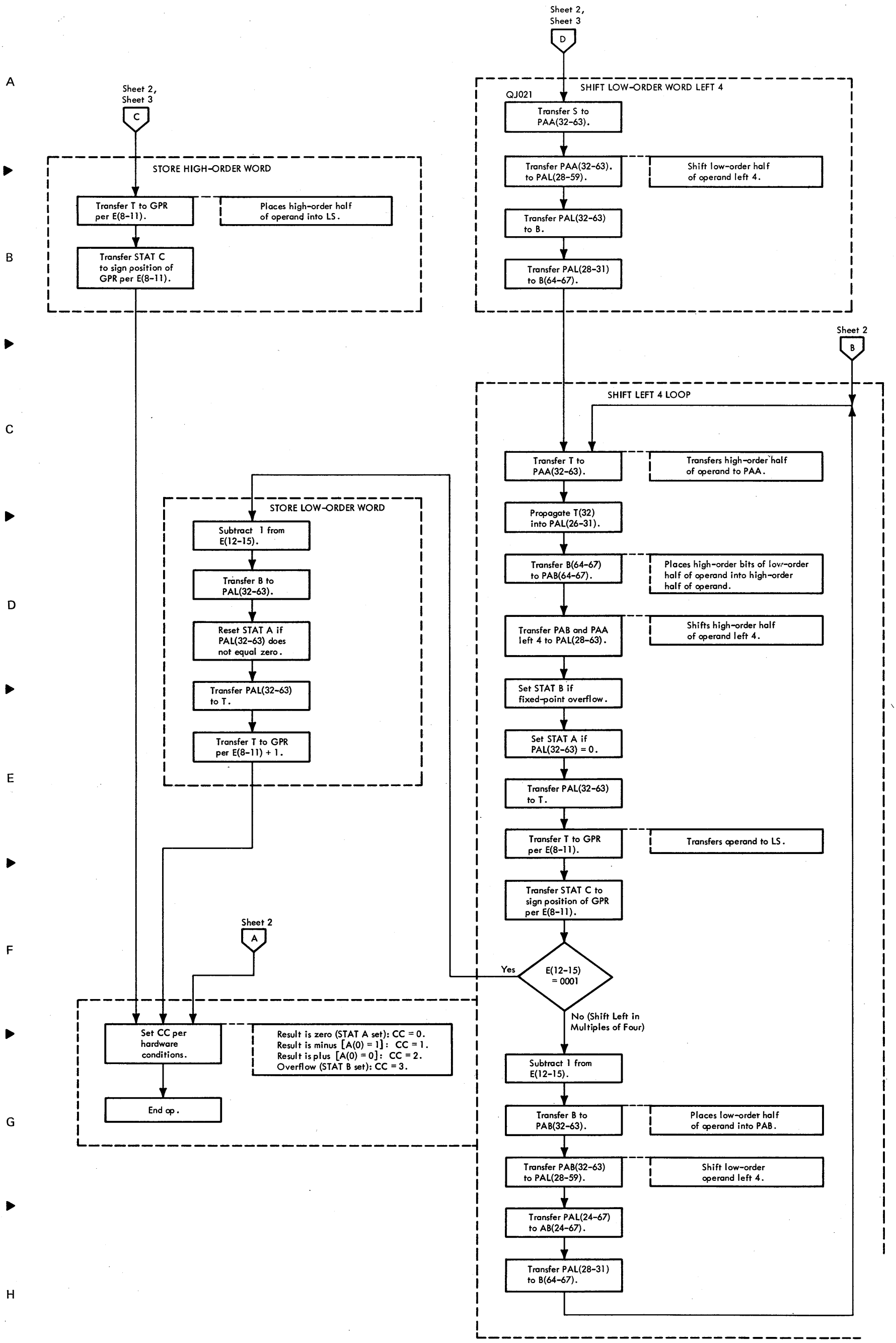


Diagram 5-117. Shift Left Double, SLDA (8F) (Sheet 4 of 4)

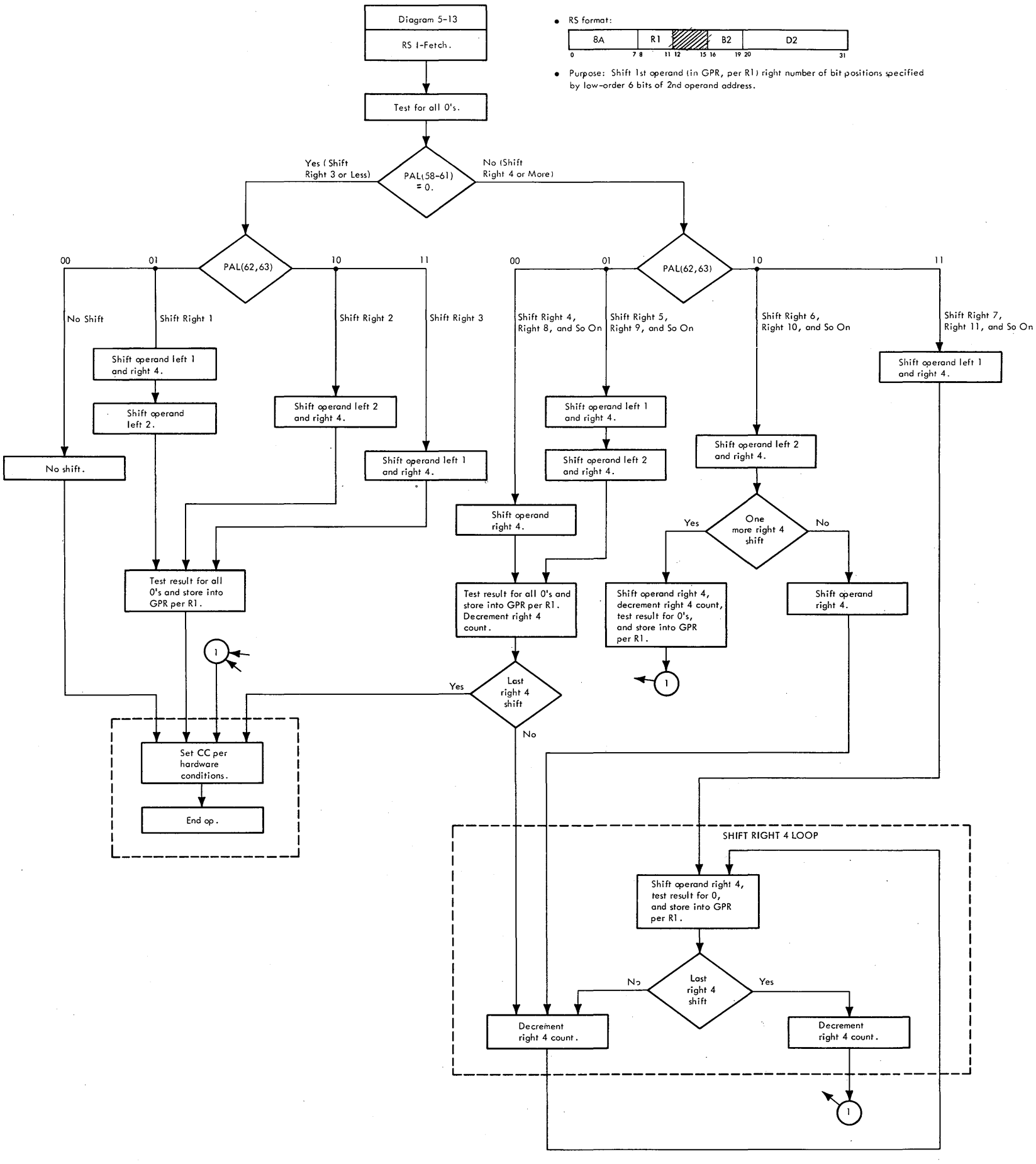
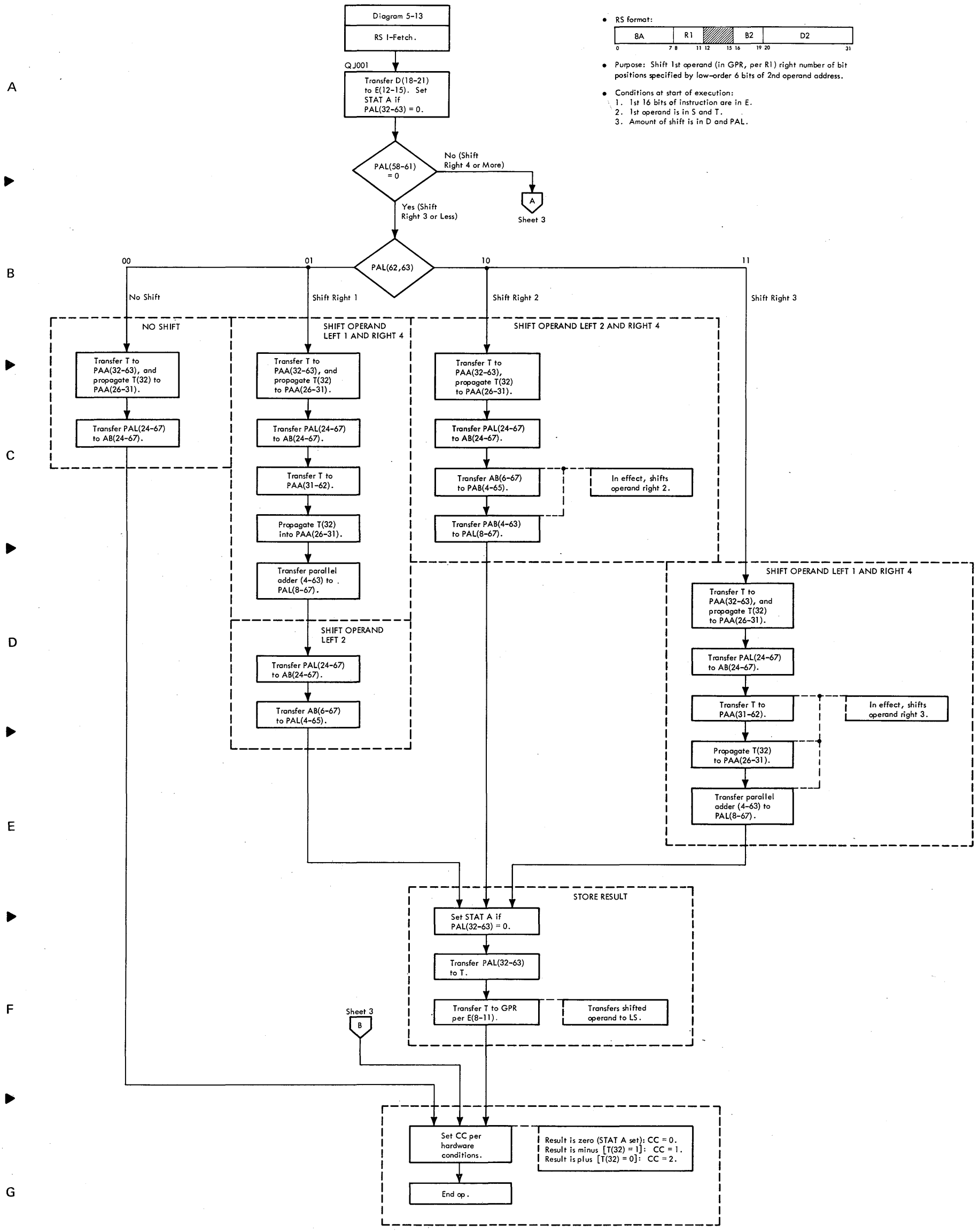


Diagram 5-118. Shift Right Single, SRA (8A) (Sheet 1 of 3)



- RS format:

- Purpose: Shift 1st operand (in GPR, per R1) right number of bit positions specified by low-order 6 bits of 2nd operand address.
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Amount of shift is in D and PAL.

Diagram 5-118. Shift Right Single, SRA (8A) (Sheet 2 of 3)

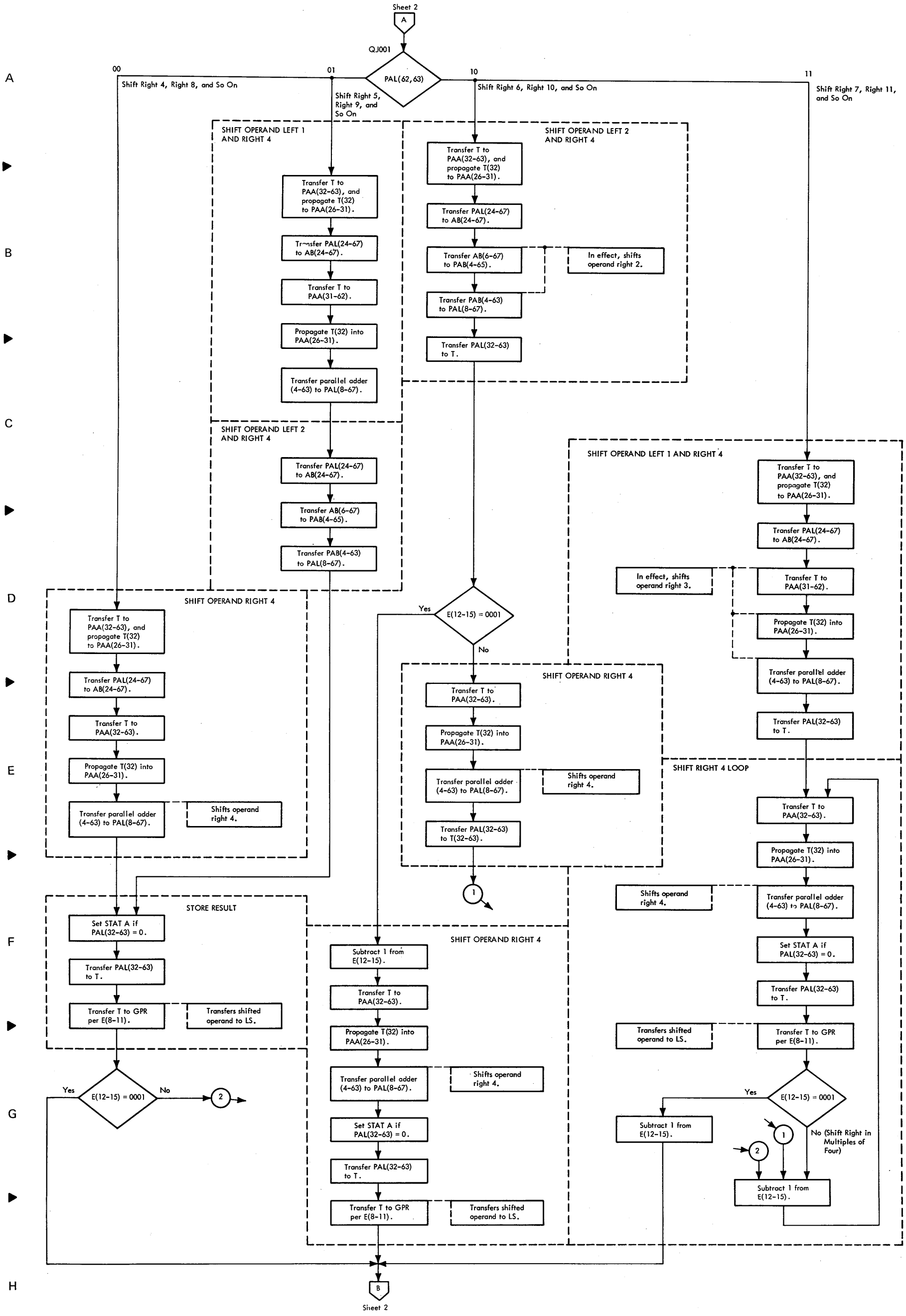
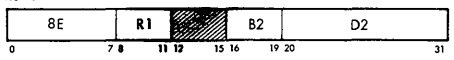


Diagram 5-118. Shift Right Single, SRA (8A) (Sheet 3 of 3)

Diagram 5-13
RS I-Fetch.

- RS format:



- Purpose: Shift 1st operand (in GPR, per R1 and R1 + 1) right number of bit positions specified by low-order 6 bits of 2nd operand address.

Specification check
Yes
Specification program interruption.
No
Test high-order word for all 0's.
PAL(58-61) = 0
Yes (Shift Right 3 or Less)
No (Shift Right 4 or More)

PAL(62,63)
00
01
10
11
No Shift
Shift Right 1
Shift Right 2
Shift Right 3

Shift high-order word left 1 and right 4, and gate underflow bits to B(64-67).
Shift high-order word left 2, test result for zero, and store into GPR per R1.
Transfer low-order word from GPR per R1 + 1.
Shift low-order word left 1, and shift low-order word plus underflow bits right 4.
Shift low-order word left 2, test result for zero, and store into GPR per R1 + 1.
Test low-order word for all 0's.

Shift high-order word right 4, and gate underflow bits to B(64-67).
Shift high-order word left 2, test result for zero, and store into GPR per R1.
Shift low-order word plus underflow bits right 4.
Last right 4 shift
Yes
No
Shift low-order word left 2, test result for zero, and store into GPR per R1 + 1.
1
2

Shift high-order word left 1 and right 4, and gate underflow bits to B(64-67).
Transfer low-order word from GPR per R1 + 1.
Shift low-order word left 1, and shift low-order word plus underflow bits right 4.
Test result for zero, and store low-order word into GPR per R1 + 1.
Store high-order word into GPR per R1.

PAL(62,63)
00
01
10
11
Shift Right 4, Right 8, and So On
Shift Right 5, Right 9, and So On
Shift Right 6, Right 10, and So On
Shift Right 7, Right 11, and So On

Shift high-order word left 1 and right 4, and gate underflow bits to B(64-67).
Shift high-order word right 4, and gate underflow bits to B(64-67).
Shift high-order word left 2.
Transfer low-order word from GPR per R1 + 1.
Shift low-order word left 1, and shift low-order word plus underflow bits right 4.
Shift low-order word left 2.
1

Shift high-order word left 1 and right 4, and gate underflow bits to B(64-67).
Transfer low-order word from GPR per R1 + 1.
Shift low-order word left 1, and shift low-order word plus underflow bits right 4.

Shift high-order word right 4, and gate underflow bits to B(64-67).
Shift low-order word plus underflow bits right 4.
Last right 4 shift
Yes
No

2
Shift low-order word left 2.
Shift high-order word right 4, test result for zero, store into GPR per R1, and gate underflow bits to B(64-67).
Shift low-order word plus underflow bits right 4, and test result for zero.

SHIFT RIGHT 4 LOOP
Last right 4 shift
No
Yes
Decrement right 4 count, shift high-order word right 4, gate underflow bits to B(64-67), and test result for zero.
Shift low-order word plus underflow bits right 4, and test result for zero.
Last right 4 shift
No
Yes

Store low-order word into GPR per R1 + 1.
Set CC per hardware conditions.
End op.

Diagram 5-119. Shift Right Double, SRDA (8E) (Sheet 1 of 4)

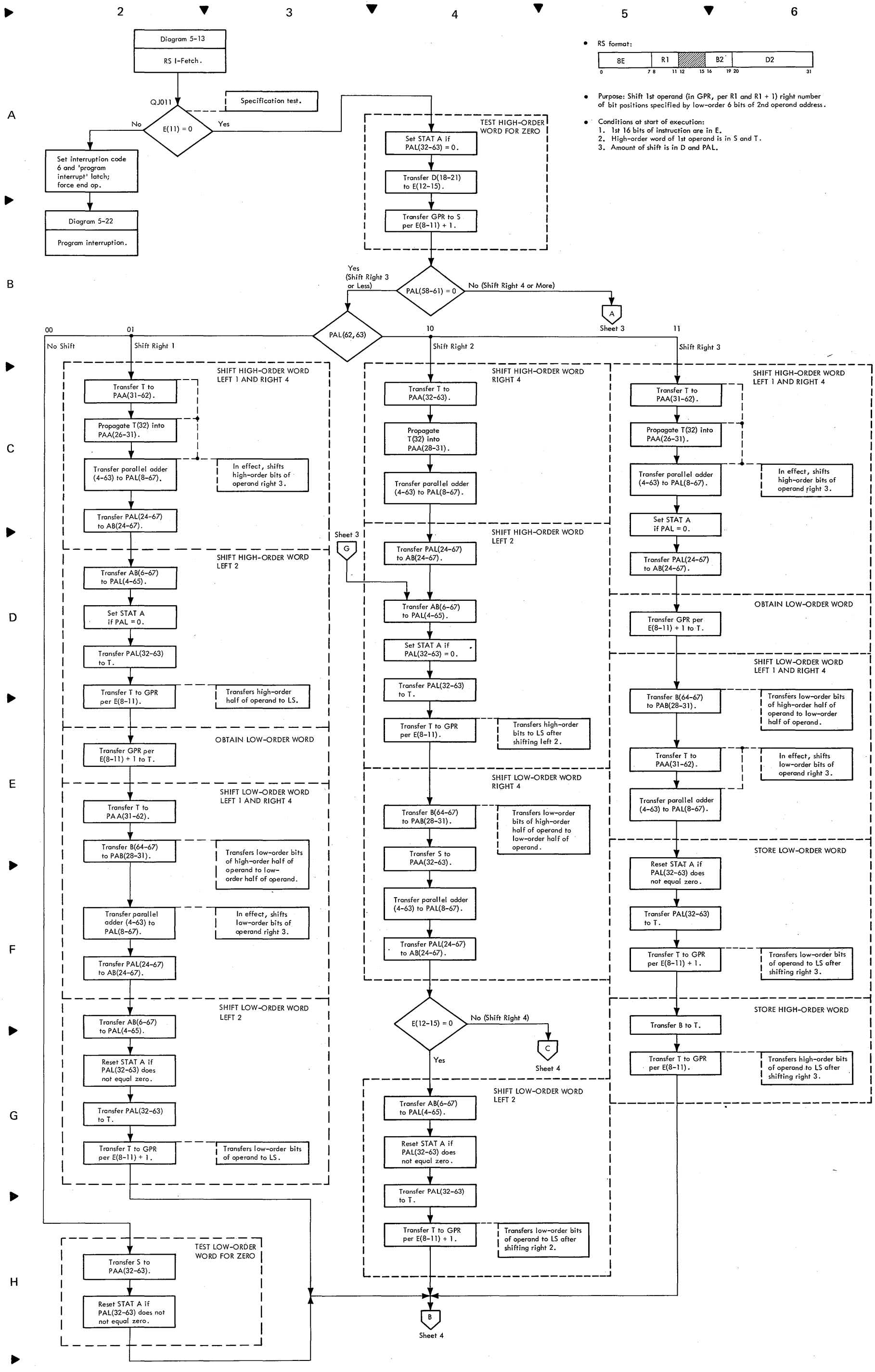


Diagram 5-119. Shift Right Double, SRDA (8E) (Sheet 2 of 4)

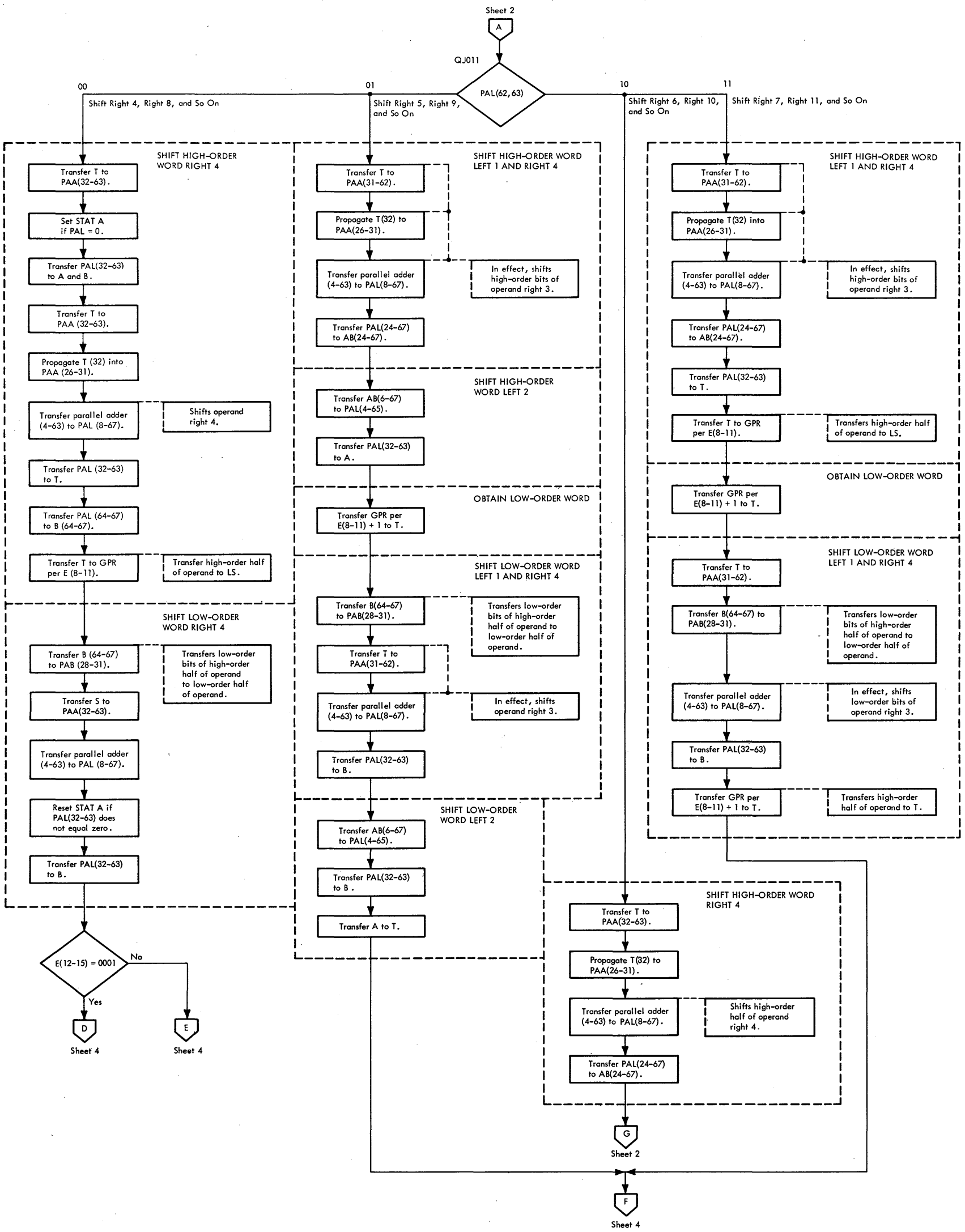


Diagram 5-119. Shift Right Double, SRDA (8E) (Sheet 3 of 4)

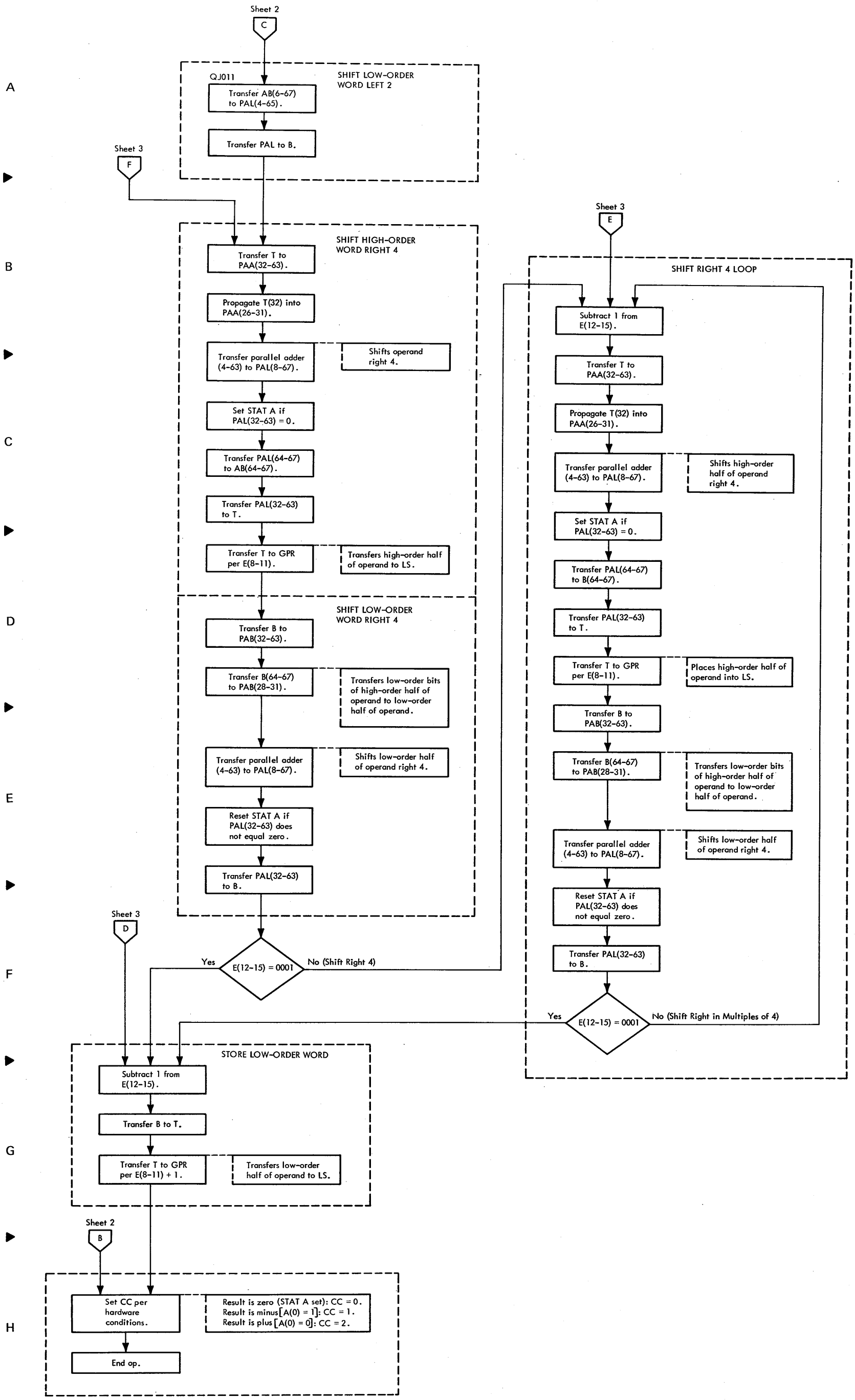


Diagram 5-119. Shift Right Double, SRDA (8E) (Sheet 4 of 4)

DIAGRAM 5-119. SHIFT RIGHT DOUBLE, SRDA (8E) (SHEET 4 OF 4)

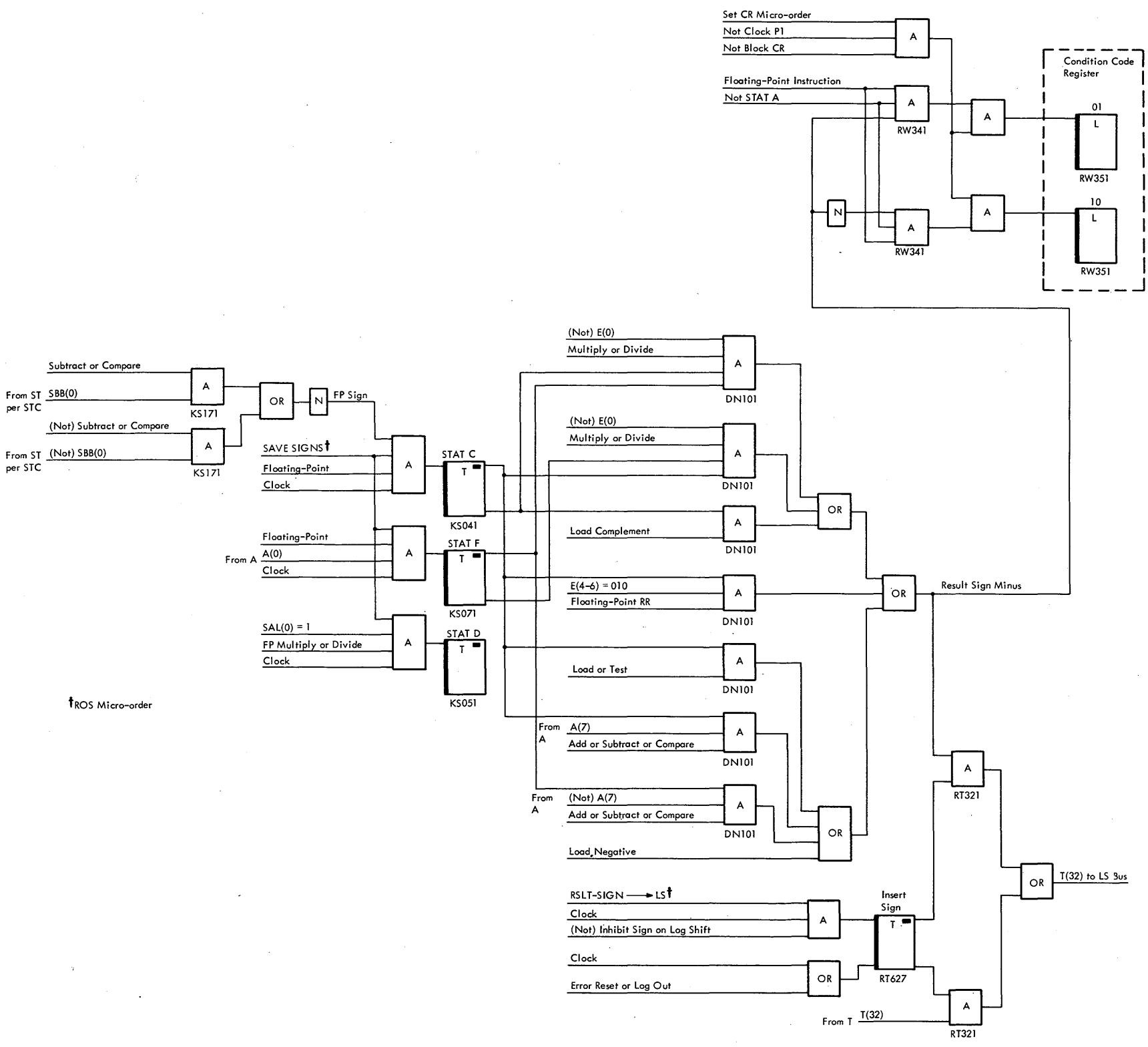
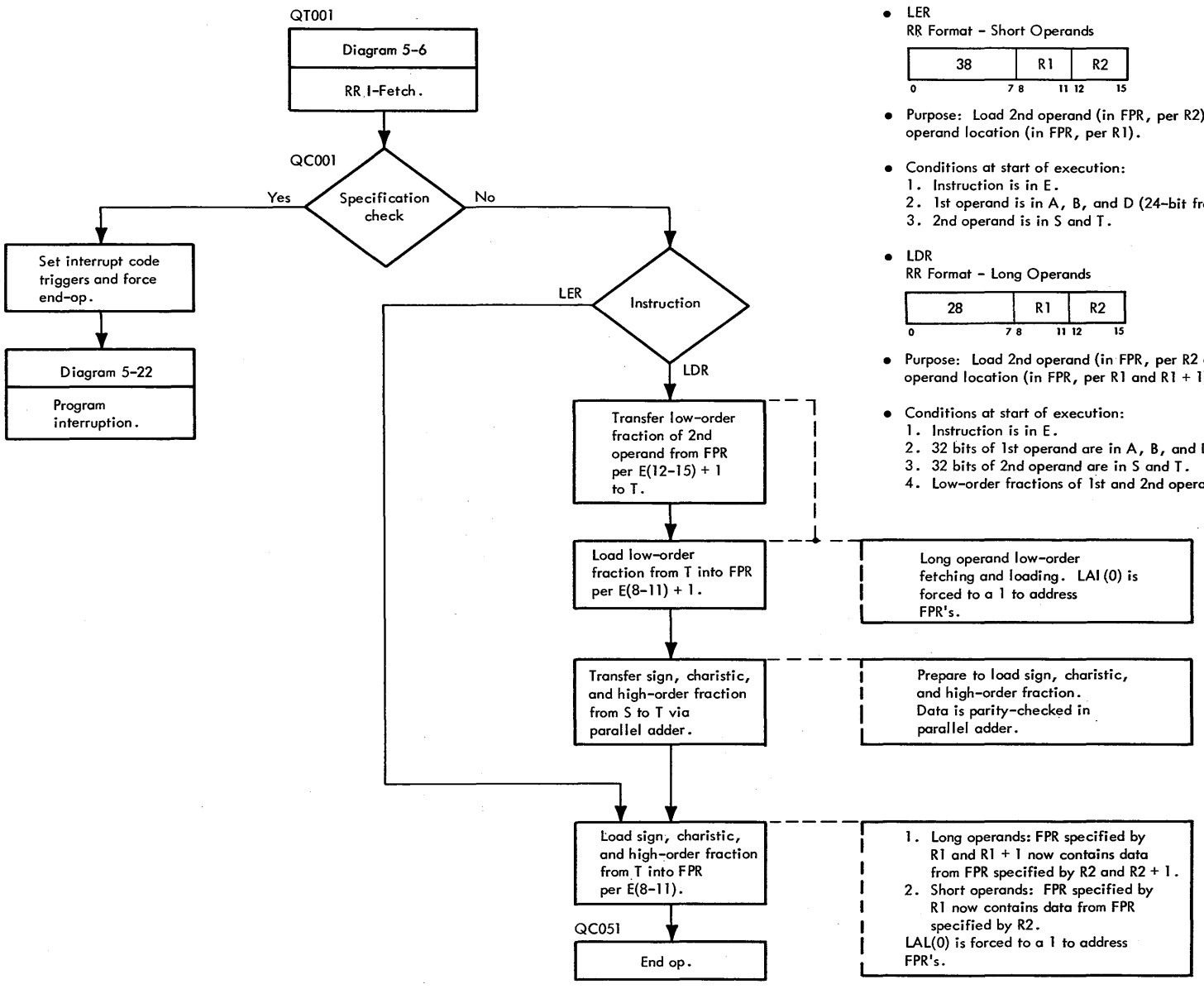


Diagram 5-201. Save Signs and Insert Sign Functions, and CC Setting

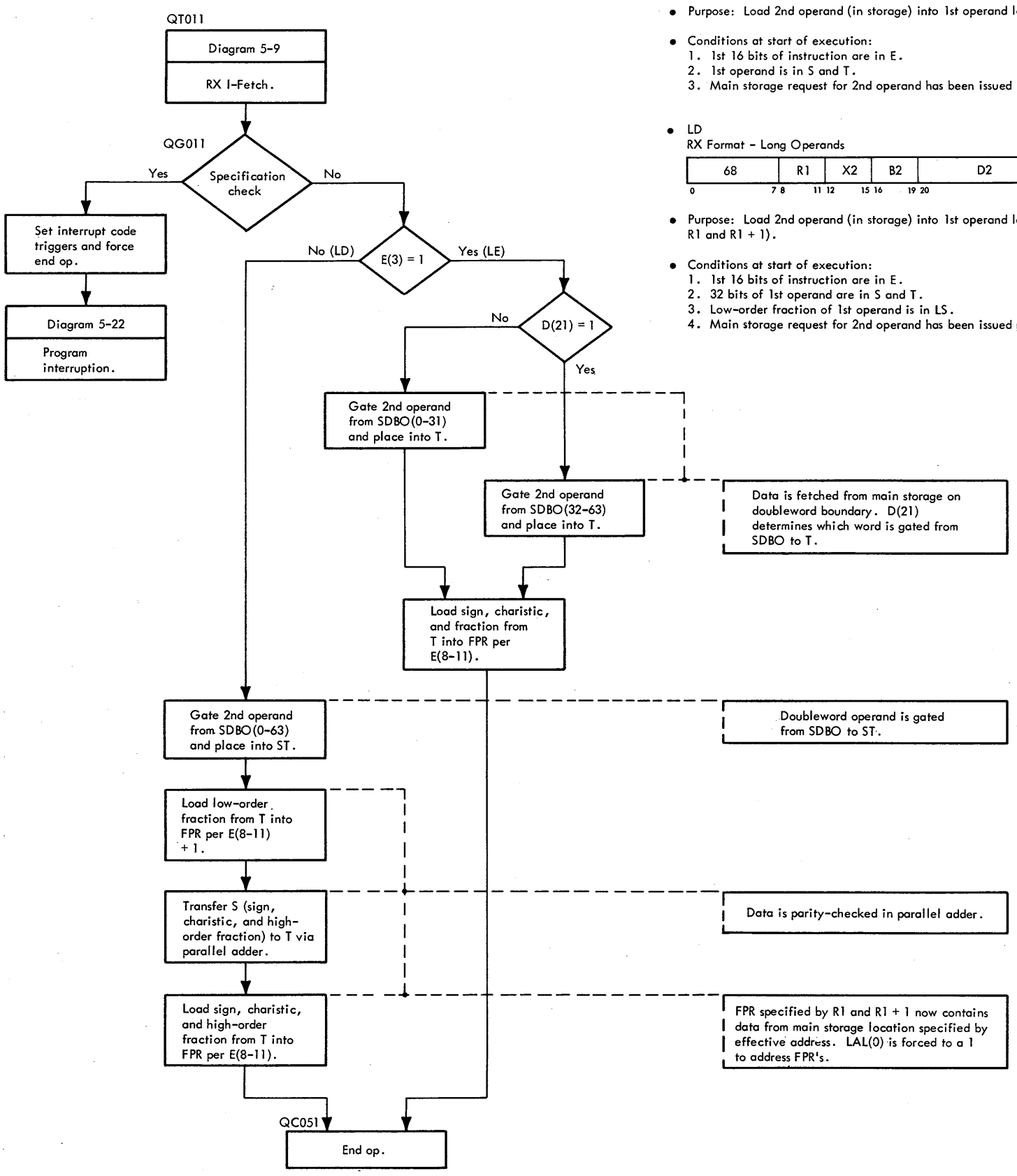


- LER
  RR Format - Short Operands

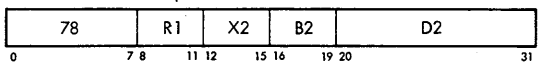
- Purpose: Load 2nd operand (in FPR, per R2) into 1st operand location (in FPR, per R1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (24-bit fraction only).
  3. 2nd operand is in S and T.
- LDR
  RR Format - Long Operands

- Purpose: Load 2nd operand (in FPR, per R2 and R2 + 1) into 1st operand location (in FPR, per R1 and R1 + 1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 32 bits of 1st operand are in A, B, and D (24-bit fraction only)
  3. 32 bits of 2nd operand are in S and T.
  4. Low-order fractions of 1st and 2nd operands are in LS.

Diagram 5-202. Load, LER (38) - Short Operands; Load, LDR (28) - Long Operands



- LE
  RX Format - Short Operands



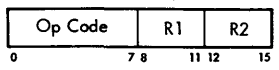
- Purpose: Load 2nd operand (in storage) into 1st operand location (in FPR, per R1).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Main storage request for 2nd operand has been issued per D.

- LD
  RX Format - Long Operands

- Purpose: Load 2nd operand (in storage) into 1st operand location (in FPR, per R1 and R1 + 1).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 32 bits of 1st operand are in S and T.
  3. Low-order fraction of 1st operand is in LS.
  4. Main storage request for 2nd operand has been issued per D.

Diagram 5-203. Load, LE (78) - Short Operands; Load, LD (68) - Long Operands

- RR Format - Short Operands



- Purpose: Load 2nd operand (in FPR, per R2) into 1st operand location (in FPR, per R1): LTER, with original sign; LCER, with sign complemented; LNER, with sign made minus; LPER, with sign made plus.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (24-bit fraction only).
  3. 2nd operand is in S and T.
- Op codes:
  1. LCER = 33.
  2. LNER = 31.
  3. LPER = 30.
  4. LTER = 32.

Objectives:

1. Fetch operand.
2. Insert sign.
3. Store 2nd operand into 1st operand location.
4. Test for zero result.

QT001

Diagram 5-6
RR I-Fetch.

QC001

Specification check

Yes

Set interrupt code triggers and force end op.

Diagram 5-22
Program interruption.

No

Transfer 2nd operand fraction from T to D.

Save sign of 2nd operand in STAT C.

Insert sign on LS bus per instruction and STAT C.

1. LPER: Plus sign to LS.
2. LNER: Minus sign to LS.
3. LTER: Original sign to LS.
4. LCER: Complement of original sign to LS.

See Diagram 5-201.

Store charistic and fraction from T into FPR per E(8-11).

1st operand location now contains 2nd operand (specified by R2) with correct sign. Sign is loaded from LS bus. LAL(0) is forced to 1 to address FPR's.

Transfer fraction from D to PAL.

Prepare to test for zero fraction. Data is parity-checked in parallel adder.

Set STAT A if fraction = 0 (PAL = 0).

Objectives:

1. Set CC per hardware conditions.
2. End op.

QC051

Result = 0 (STAT A = 1)

Yes

Set CC to 0 (fraction = 0).

No

Instruction

LPER

LTER

LCER

LNER

Original sign plus (STAT C = 0)

Yes

No

Original sign plus (STAT C = 0)

Yes

No

Set CC to 2 (2nd operand > 0).

Set CC to 1 (2nd operand < 0).

End op.

Diagram 5-204. Load Positive, LPER (30); Load Negative, LNER (31); Load and Test, LTER (32); Load Complement, LCER (33) - Short Operand

- RR Format - Long Operands

Op Code	R1	R2
0 7	8 11	12 15

- Purpose: Load 2nd operand (in FPR, per R2 and R2 + 1) into 1st operand location (in FPR, per R1 and R1 + 1); LTDR, with original sign; LCDR, with sign complemented; LNDR, with sign made minus; LPDR, with sign made plus.

- Conditions at start of execution:
  1. Instruction is in E.
  2. 32 bits of 1st operand are in A, B, and D (24-bit fraction only).
  3. 32 bits of 2nd operand are in S and T.
  4. Low-order fractions of 1st and 2nd operands are in LS.

- Op codes:
  1. LCDR = 23.
  2. LNDR = 21.
  3. LPDR = 20.
  4. LTDR = 22.

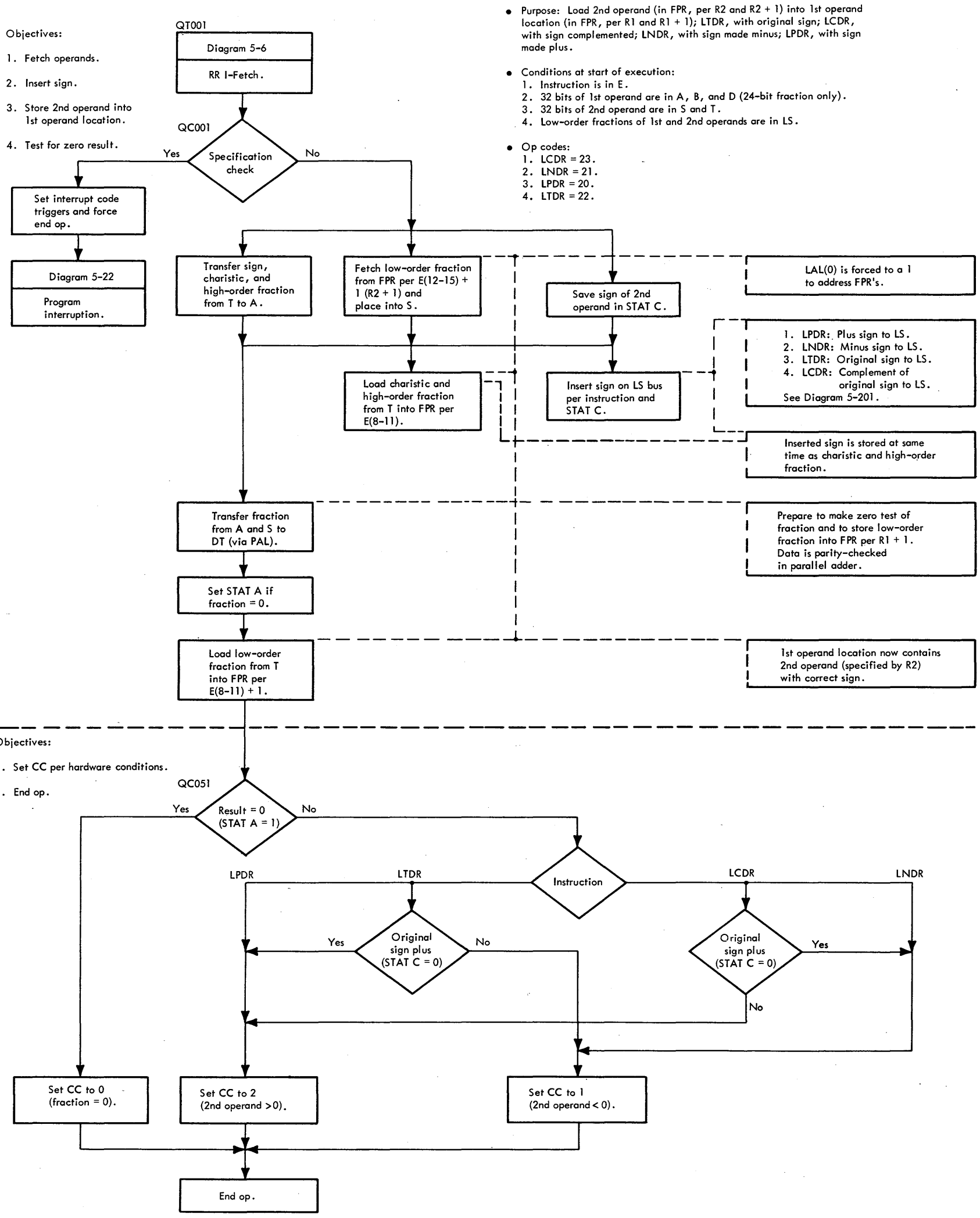


Diagram 5-205. Load Positive, LPDR (20); Load Negative, LNDR (21); Load and Test, LTDR (22); Load Complement, LCDR (23) - Long Operands

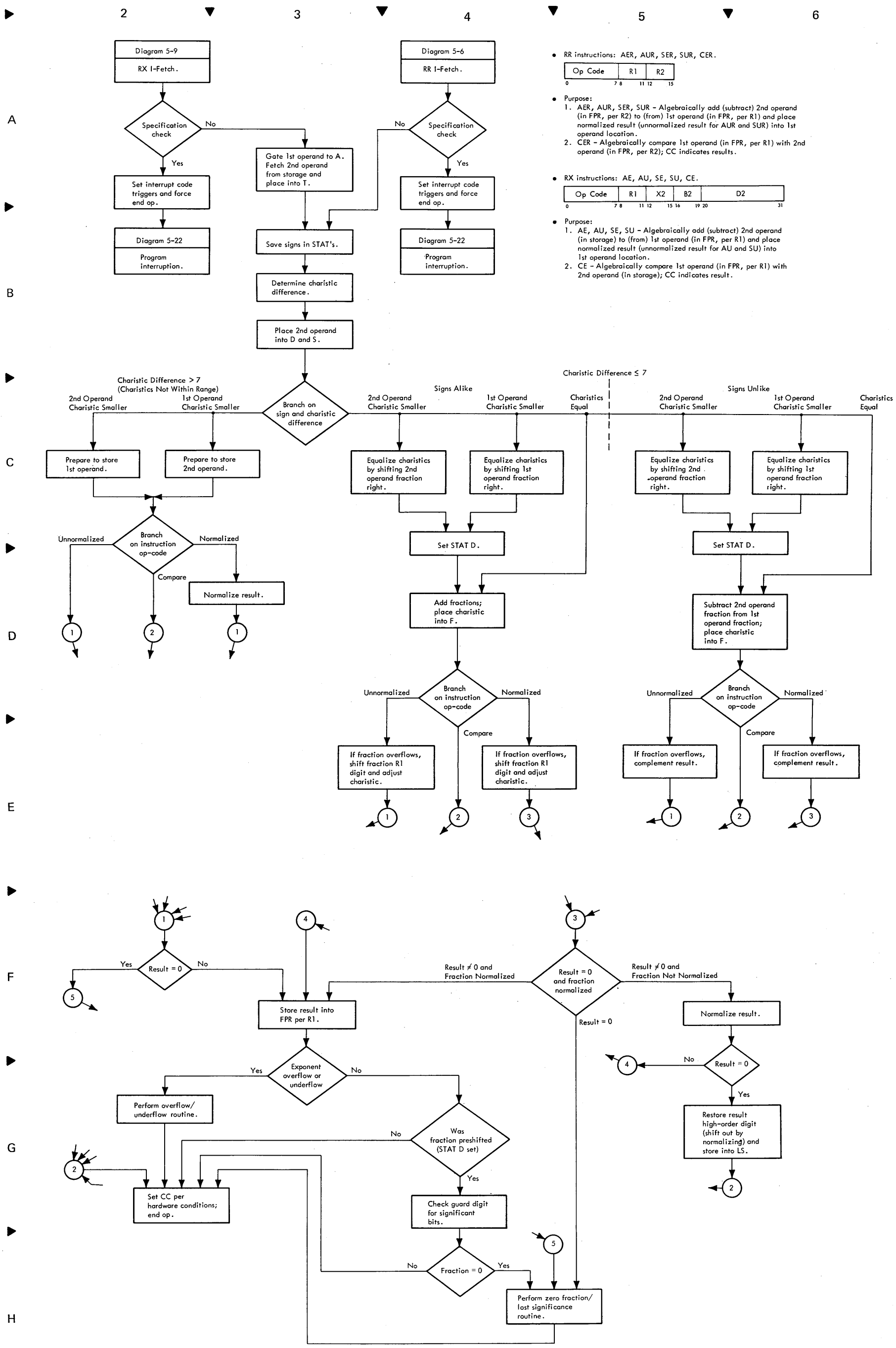
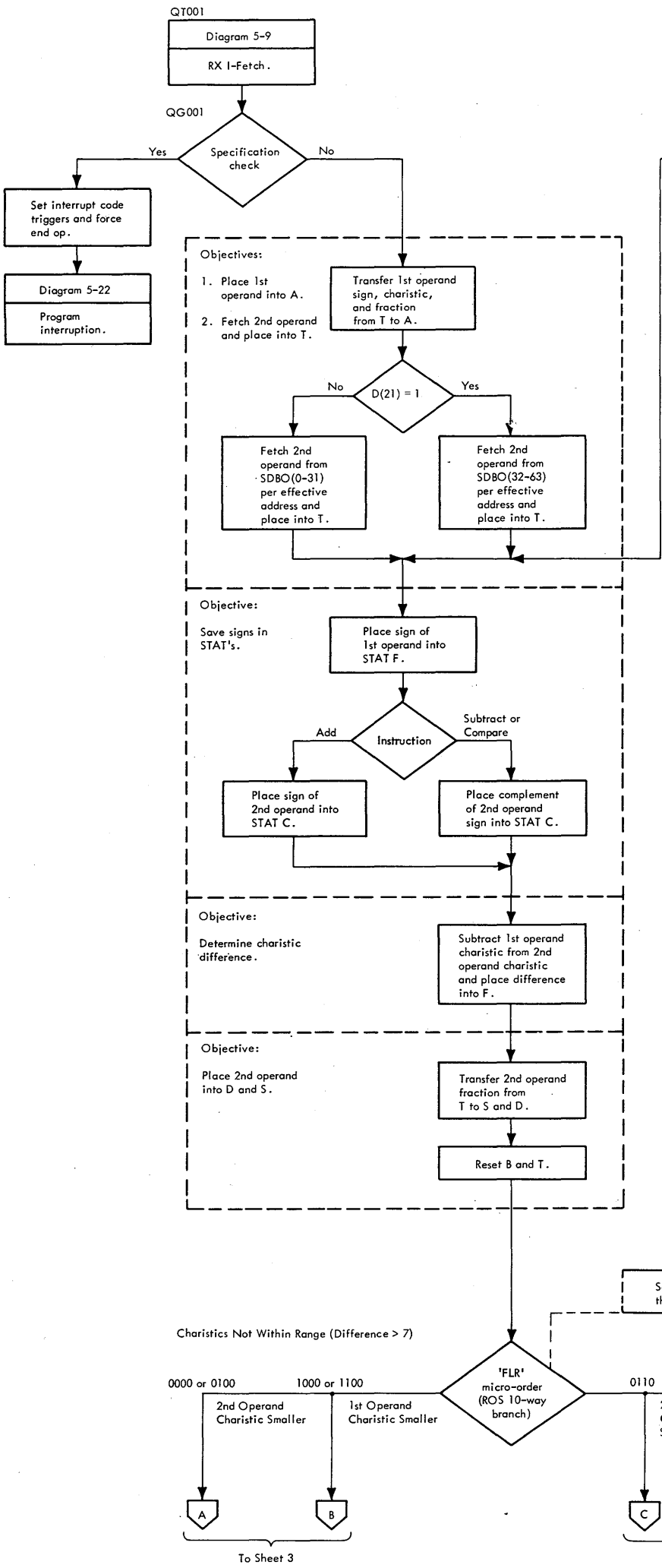
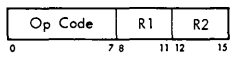


Diagram 5-206. Floating-Point Add, Subtract, and Compare - Short Operands (Sheet 1 of 5)



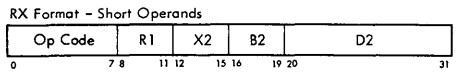
- RR Format - Short Operands



- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (24-bit fraction only).
  3. 2nd operand is in S and T.
  4. STC = 4 ('RESET' micro-order during I-Fetch).

- Op Code:
  1. AER = 3A.
  2. AUR = 3E.
  3. SER = 3B.
  4. SUR = 3F.
  5. CER = 39.

- RX Format - Short Operands



- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Effective address of 2nd operand is in D.
  4. 2nd operand is in main storage.
  5. STC = 4 ('RESET' micro-order during I-Fetch).

- Op Codes:
  1. AE = 7A.
  2. AU = 7E.
  3. SE = 7B.
  4. SU = 7F.
  5. CE = 79.

= 1 if a serial adder carry occurred (2nd operand charistic ≥ 1st operand charistic).
= 1 if signs are alike (add fractions).
= 1 if charistics are within range (charistic difference ≤ 7):
  a. Serial adder carry and SAL(0-4) = 0.
  b. No serial adder carry and SAL(0-4) = F (hex).
= 1 if SAL = 0 (charistics equal).

ROSAR				Operation Performed
8	9	10	11	
0	0	0	0	Store 1st operand.
0	0	1	0	Preshift 2nd operand and subtract 2nd operand from 1st operand.
0	1	0	0	Store 1st operand.
0	1	1	0	Preshift 2nd operand and add 2nd operand to 1st operand.
1	0	0	0	Store 2nd operand.
1	0	1	0	Preshift 1st operand and subtract 2nd operand from 1st operand.
1	0	1	1	Subtract 2nd operand from 1st operand (charistics equal).
1	1	0	0	Store 2nd operand.
1	1	1	0	Preshift 1st operand and add 2nd operand to 1st operand.
1	1	1	1	Add 2nd operand to 1st operand (charistics equal).

Diagram 5-206. Floating-Point Add, Subtract, and Compare - Short Operands (Sheet 2 of 5)

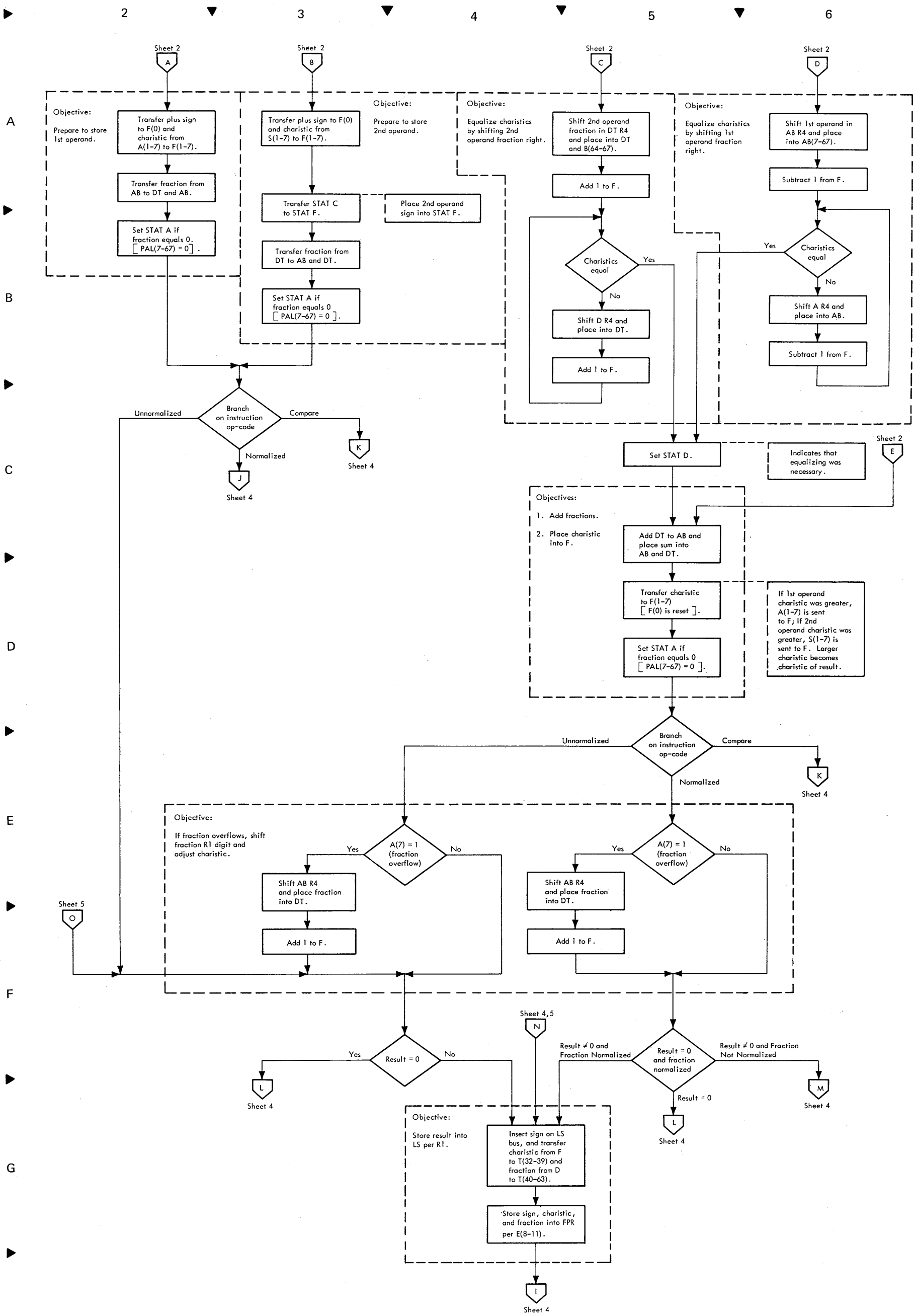


Diagram 5-206. Floating-Point Add, Subtract, and Compare - Short Operands (Sheet 3 of 5)

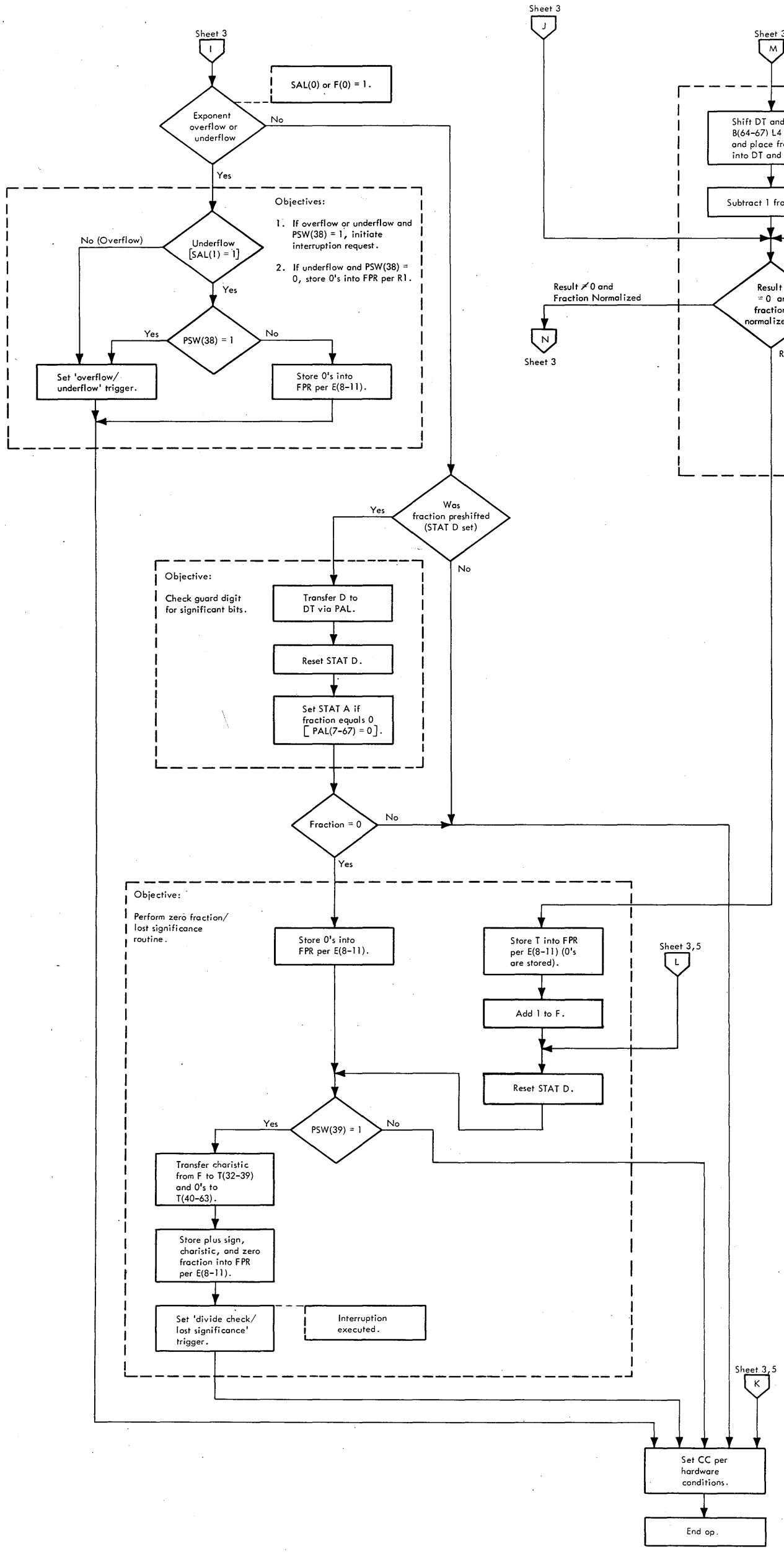


Diagram 5-206. Floating-Point Add, Subtract, and Compare - Short Operands (Sheet 4 of 5)

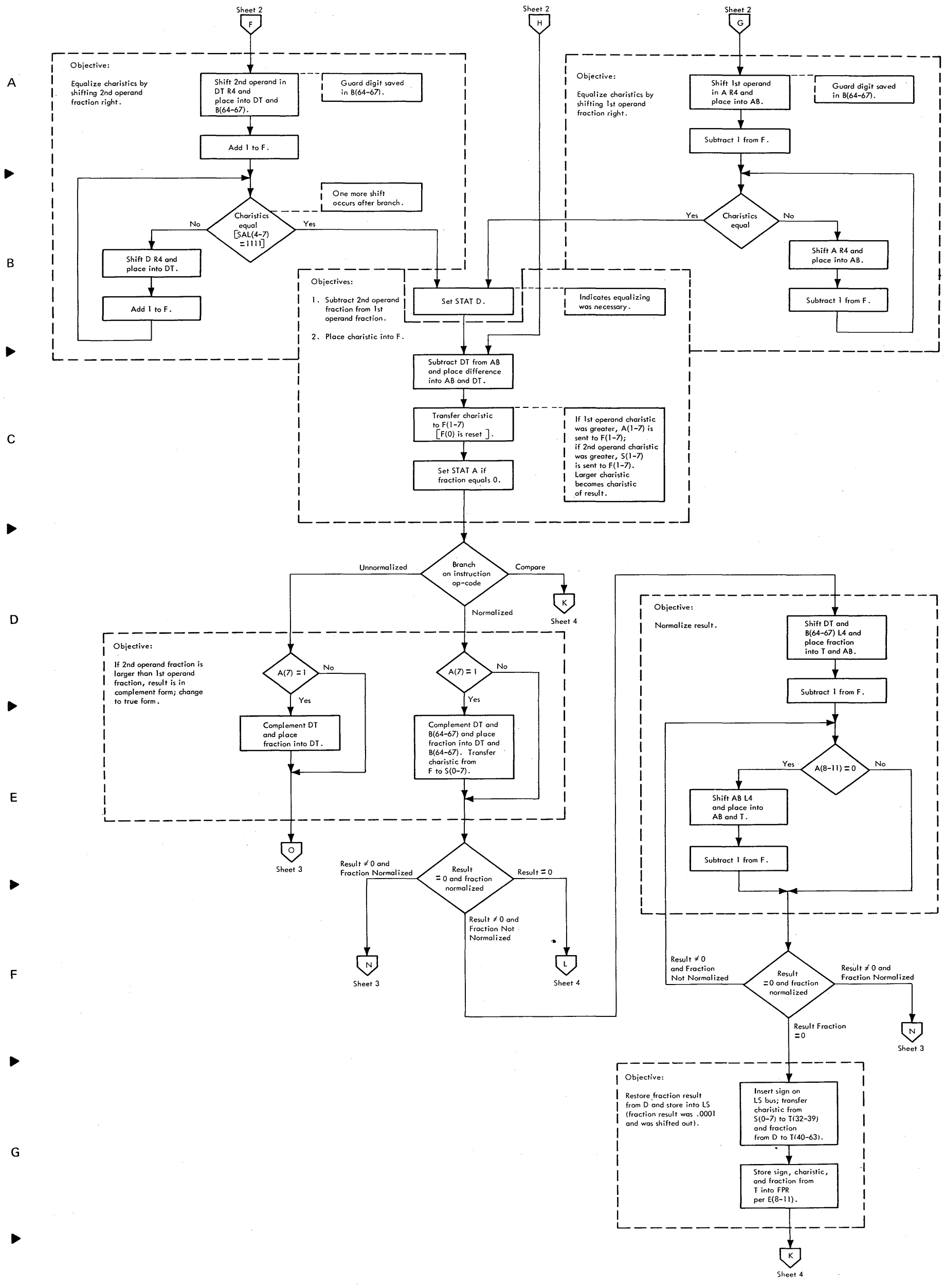
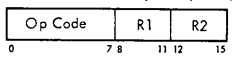


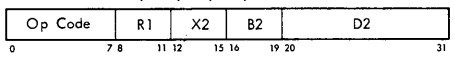
Diagram 5-206. Floating-Point Add, Subtract, and Compare - Short Operands (Sheet 5 of 5)

- RR instructions: ADR, AWR, SDR, SWR, CDR



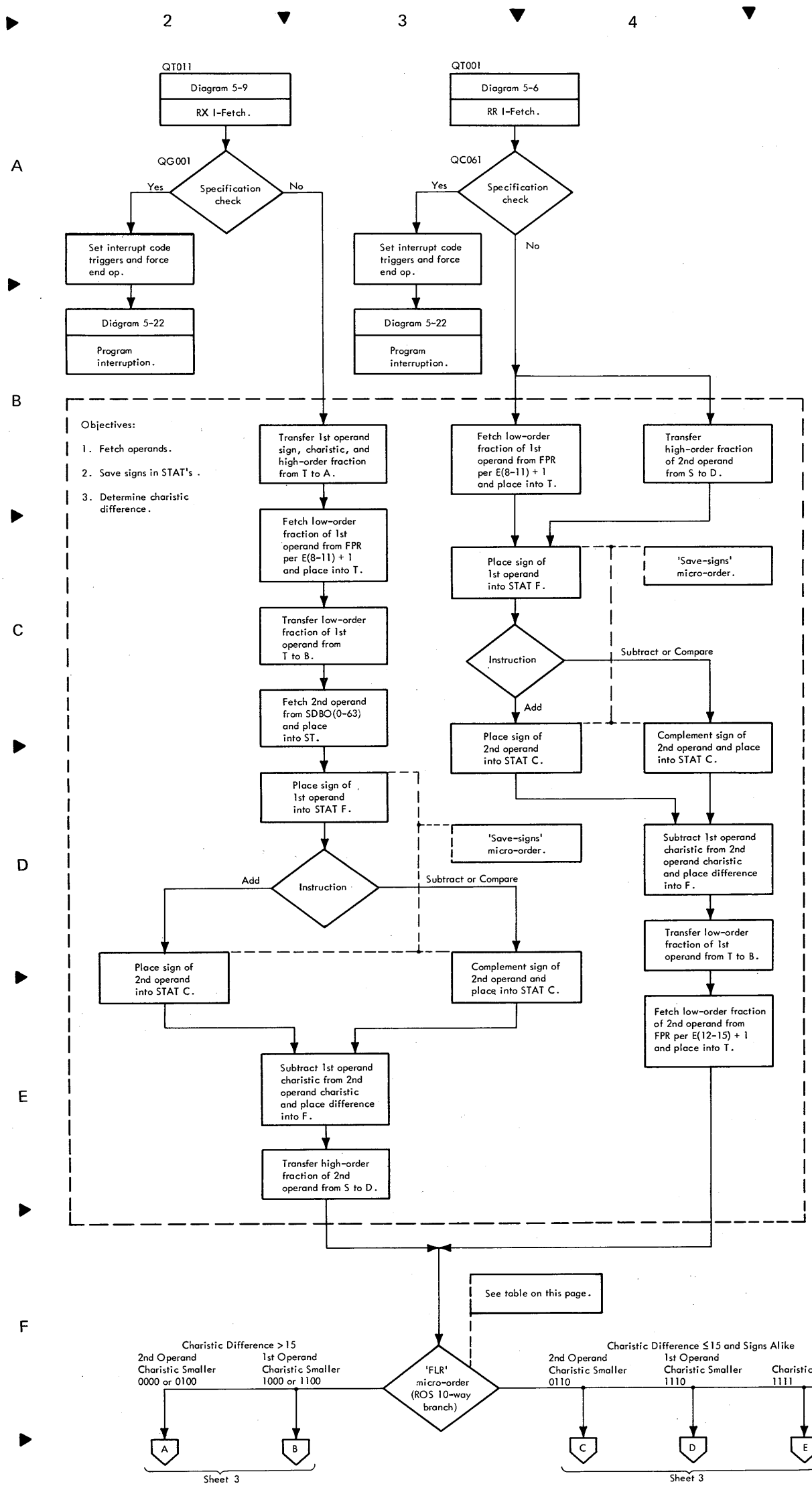
- Purpose:
  1. ADR, AWR, SDR, SWR - Algebraically add (subtract) 2nd operand (in FPR, per R2 and R2 + 1) to (from) 1st operand (in FPR, per R1 and R1 + 1) and place normalized result (unnormalized result for AWR and SWR) into 1st operand location.
  2. CDR - Algebraically compare 1st operand (in FPR, per R1 and R1 + 1) with 2nd operand (in FPR, per R2 and R2 + 1); CC indicates result.

- RX instructions: AD, AW, SD, SW, CD



- Purpose:
  1. AD, AW, SD, SW - Algebraically add (subtract) 2nd operand (in storage) to (from) 1st operand (in FPR, per R1 and R1 + 1) and place normalized result (unnormalized result for AW and SW) into 1st operand location.
  2. CD - Algebraically compare 1st operand (in FPR, per R1 and R1 + 1) with 2nd operand (in storage); CC indicates result.

Diagram 5-207. Floating-Point Add, Subtract, and Compare - Long Operands (Sheet 1 of 5)



- RR Format - Long Operands

Op Code	R1	R2
0 7	8 11	12 15

- Conditions at start of execution:
  1. Instruction is in E.
  2. 32 bits of 1st operand are in A, B, and D (24-bit fraction only).
  3. 32 bits of 2nd operand are in S and T.
  4. Low-order fraction of 1st and 2nd operand is in LS.
  5. STC = 4.

- Op Codes:
  1. ADR = 2A.
  2. AWR = 2E.
  3. SDR = 2B.
  4. SWR = 2F.
  5. CDR = 29.

- RX Format - Long Operands

Op Code	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 32 bits of 1st operand are in S and T.
  3. Low-order fraction of 1st operand is in LS.
  4. Effective address of 2nd operand is in D.
  5. 2nd operand is in main storage.

- Op Codes:
  1. AD = 6A.
  2. AW = 6E.
  3. SD = 6B.
  4. SW = 6F.
  5. CD = 69.

= 1 if a serial adder carry (2nd operand ≥ 1st operand).

= 1 if signs are alike (add fractions).

= 1 if preshifting is necessary (= 0 if charistics are not within range):
a. Serial adder carry and SAL(0-3) = 0.
b. No serial adder carry and SAL(0-3) = 1.

= 1 if SAL = 0 (charistics equal).

ROSAR				Operation Performed
8	9	10	11	
0	0	0	0	Store 1st operand.
0	0	1	0	Preshift 2nd operand and subtract 2nd operand from 1st operand.
0	1	0	0	Store 1st operand.
0	1	1	0	Preshift 2nd operand and add 2nd operand to 1st operand.
1	0	0	0	Store 2nd operand.
1	0	1	0	Preshift 1st operand and subtract 2nd operand from 1st operand.
1	0	1	1	Subtract 2nd operand from 1st operand.
1	1	0	0	Store 2nd operand.
1	1	1	0	Preshift 1st operand and add 2nd operand to 1st operand.
1	1	1	1	Add 2nd operand to 1st operand.

Diagram 5-207. Floating-Point Add, Subtract, and Compare - Long Operands (Sheet 2 of 5)

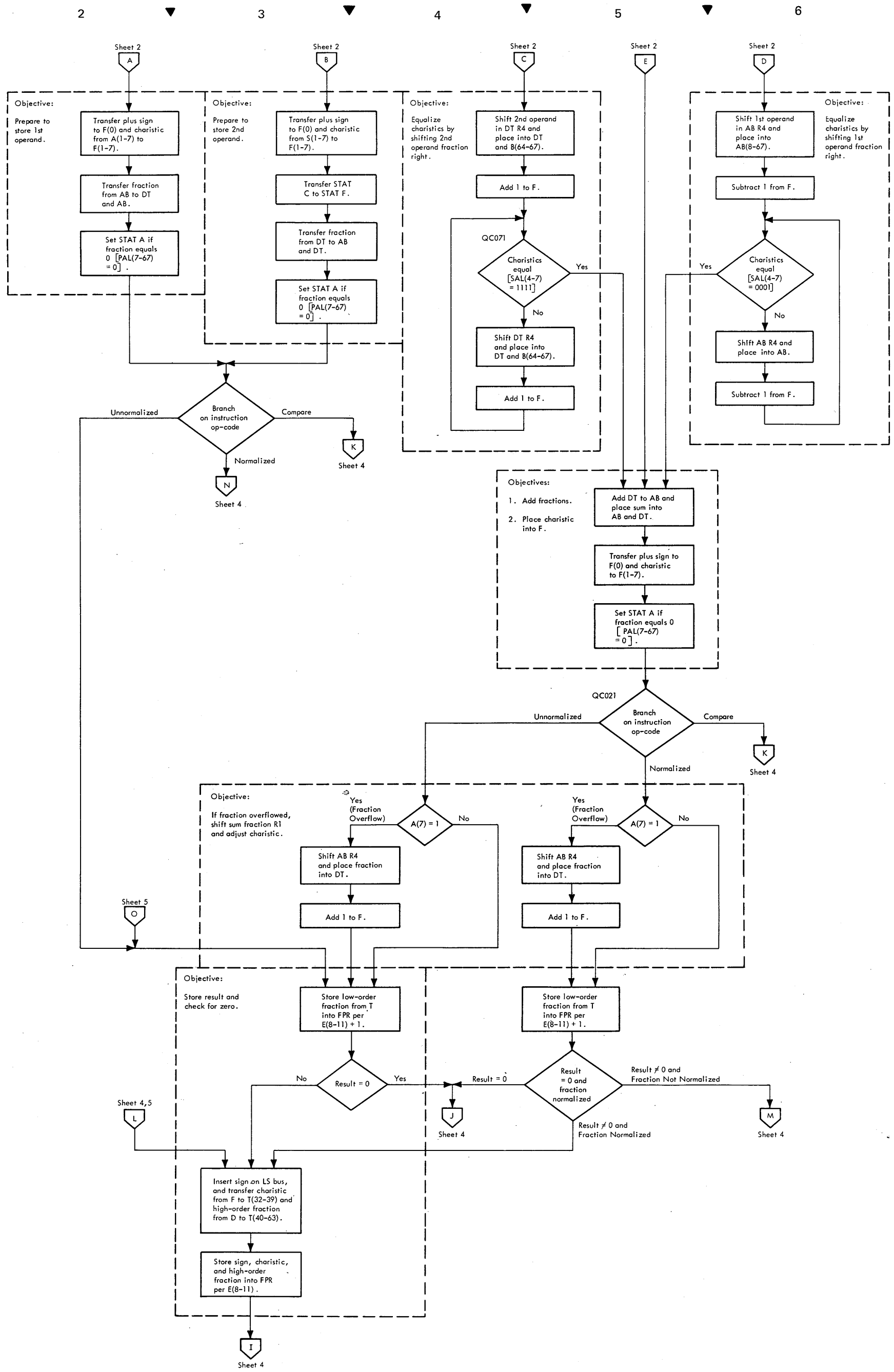


Diagram 5-207. Floating-Point Add, Subtract, and Compare - Long Operands (Sheet 3 of 5)

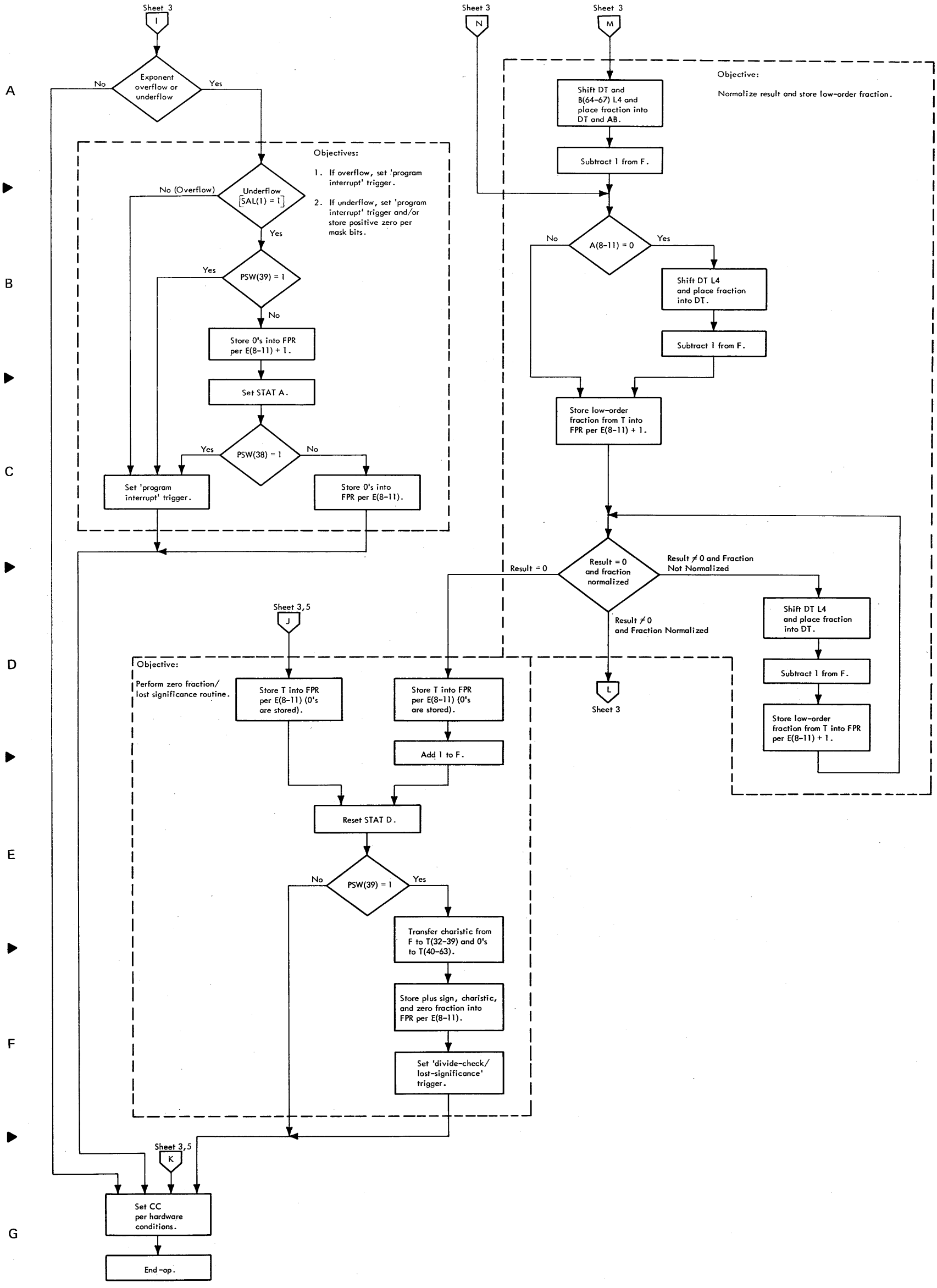


Diagram 5-207. Floating-Point Add, Subtract, and Compare - Long Operands (Sheet 4 of 5)

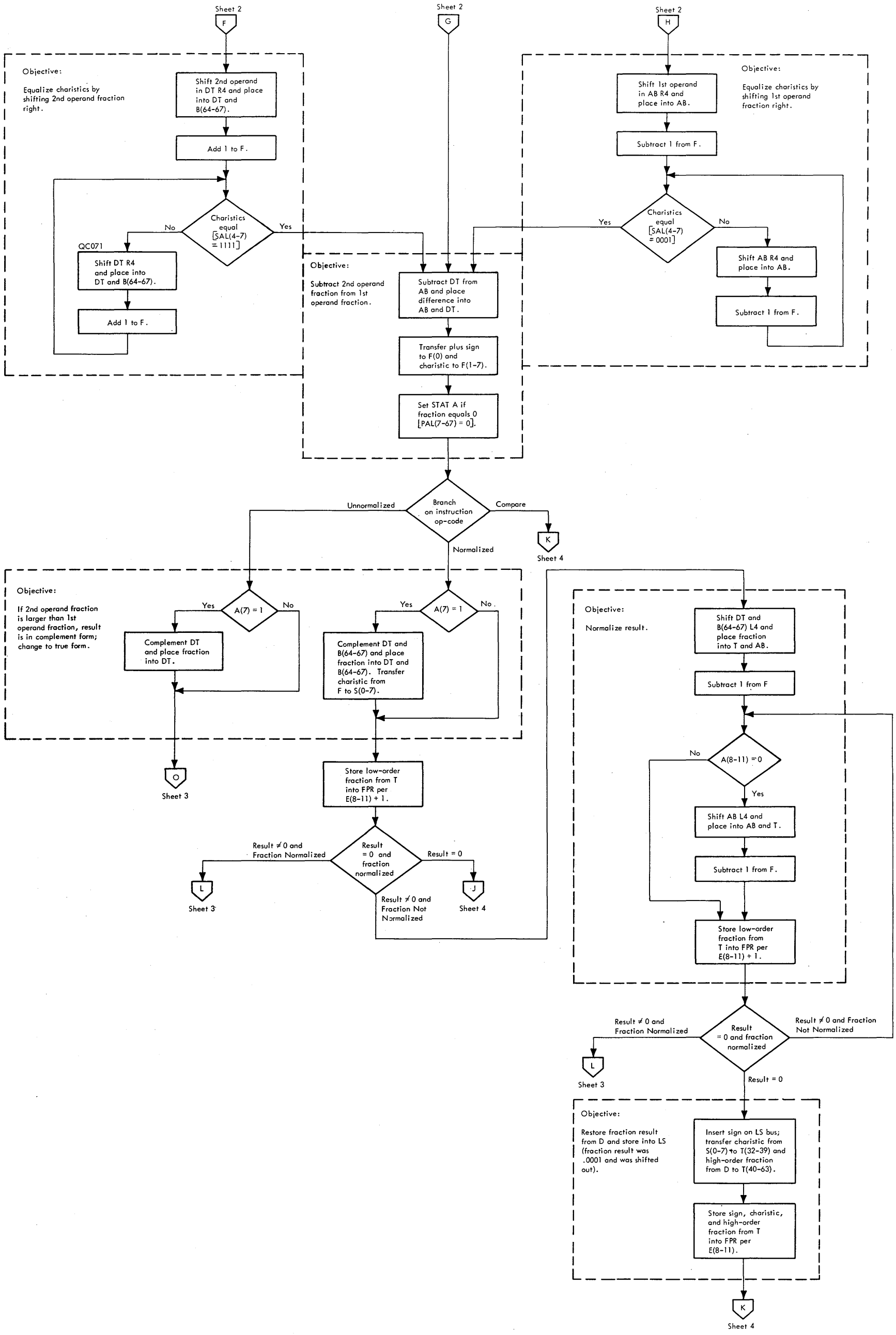


Diagram 5-207. Floating-Point Add, Subtract, and Compare - Long Operands (Sheet 5 of 5)

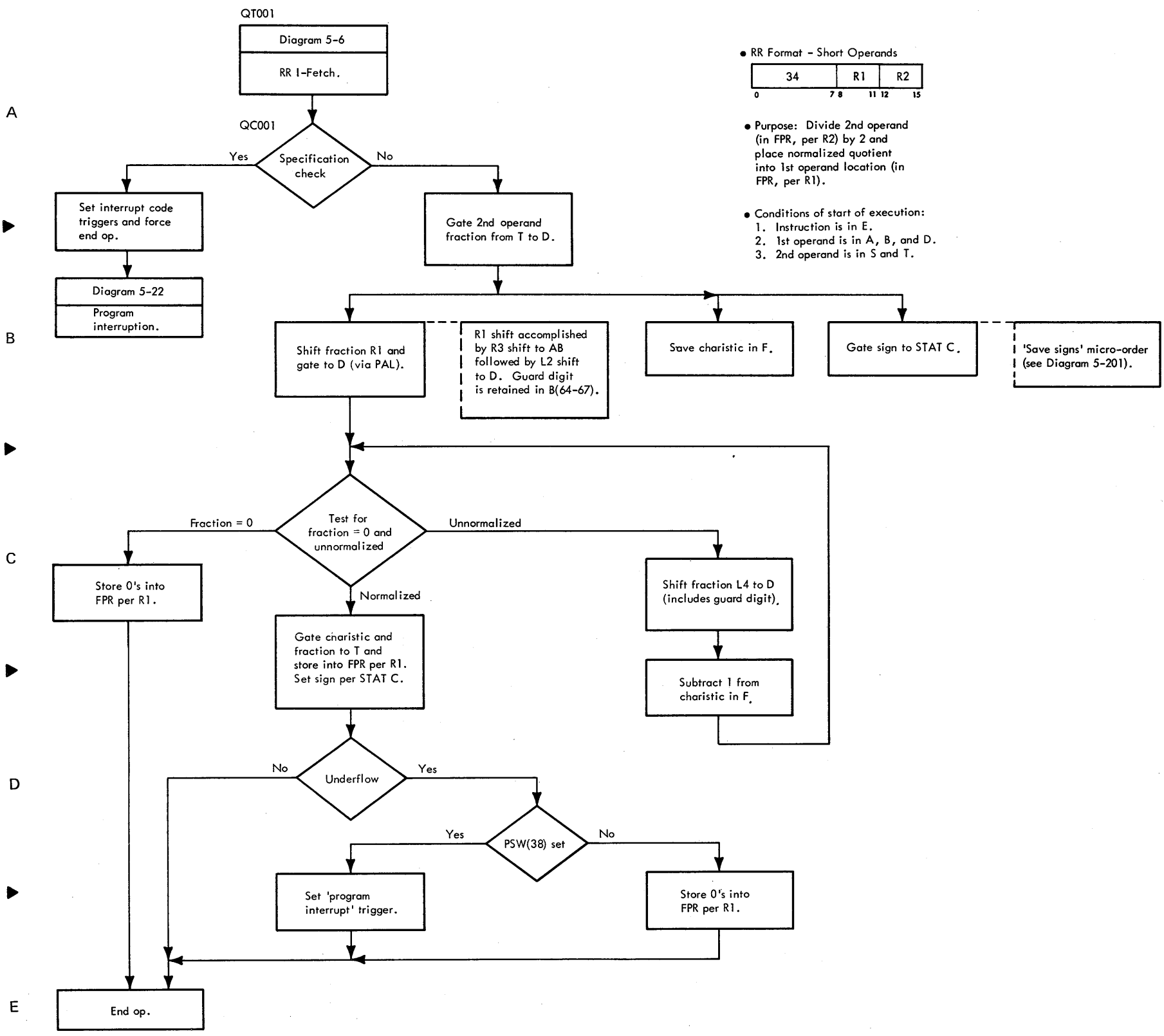
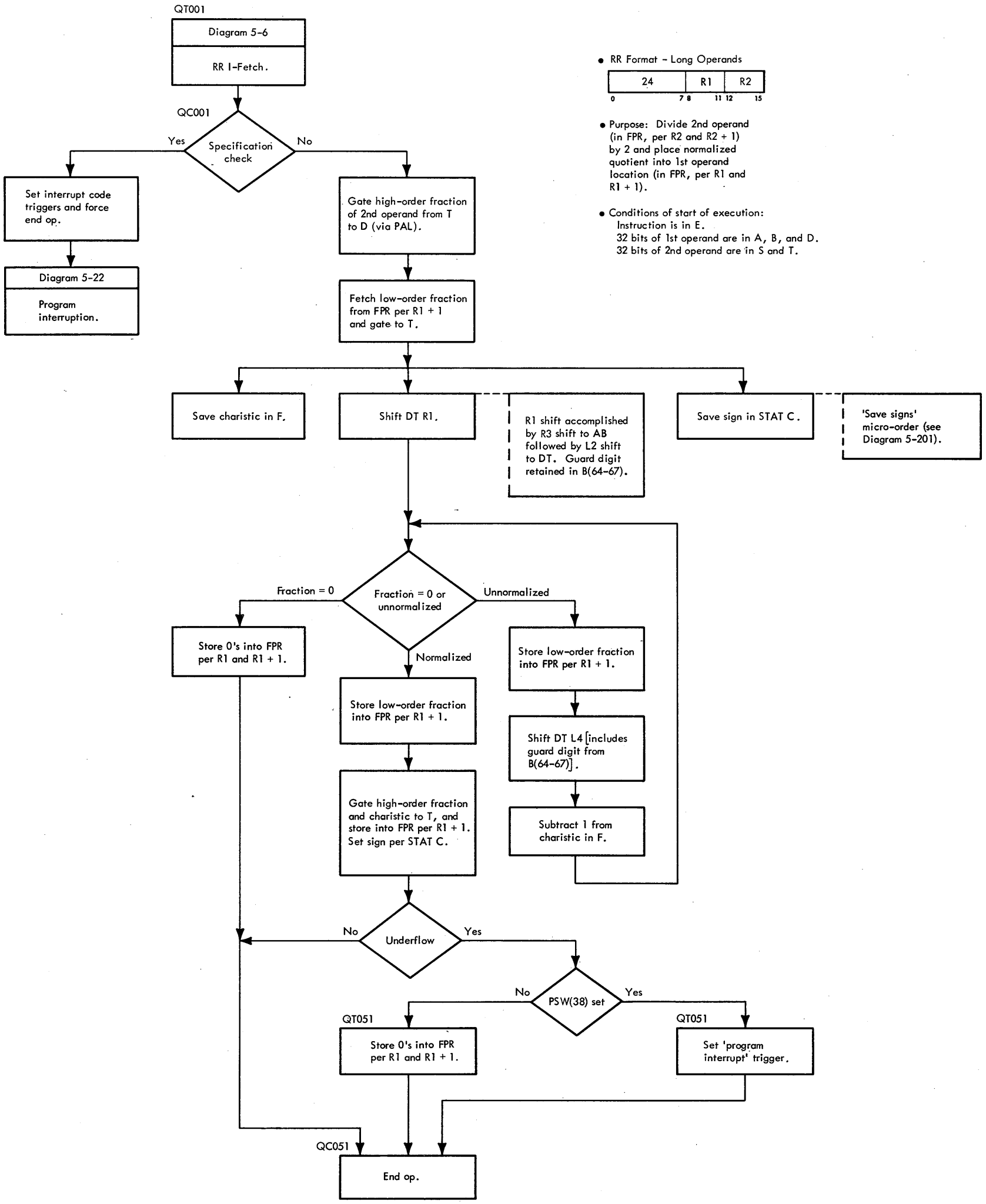


Diagram 5-208. Halve, HER (34) - Short Operands



- RR Format - Long Operands

- Purpose: Divide 2nd operand (in FPR, per R2 and R2 + 1) by 2 and place normalized quotient into 1st operand location (in FPR, per R1 and R1 + 1).
- Conditions of start of execution:
  Instruction is in E.
  32 bits of 1st operand are in A, B, and D.
  32 bits of 2nd operand are in S and T.

Diagram 5-209. Halve, HDR (24) - Long Operands

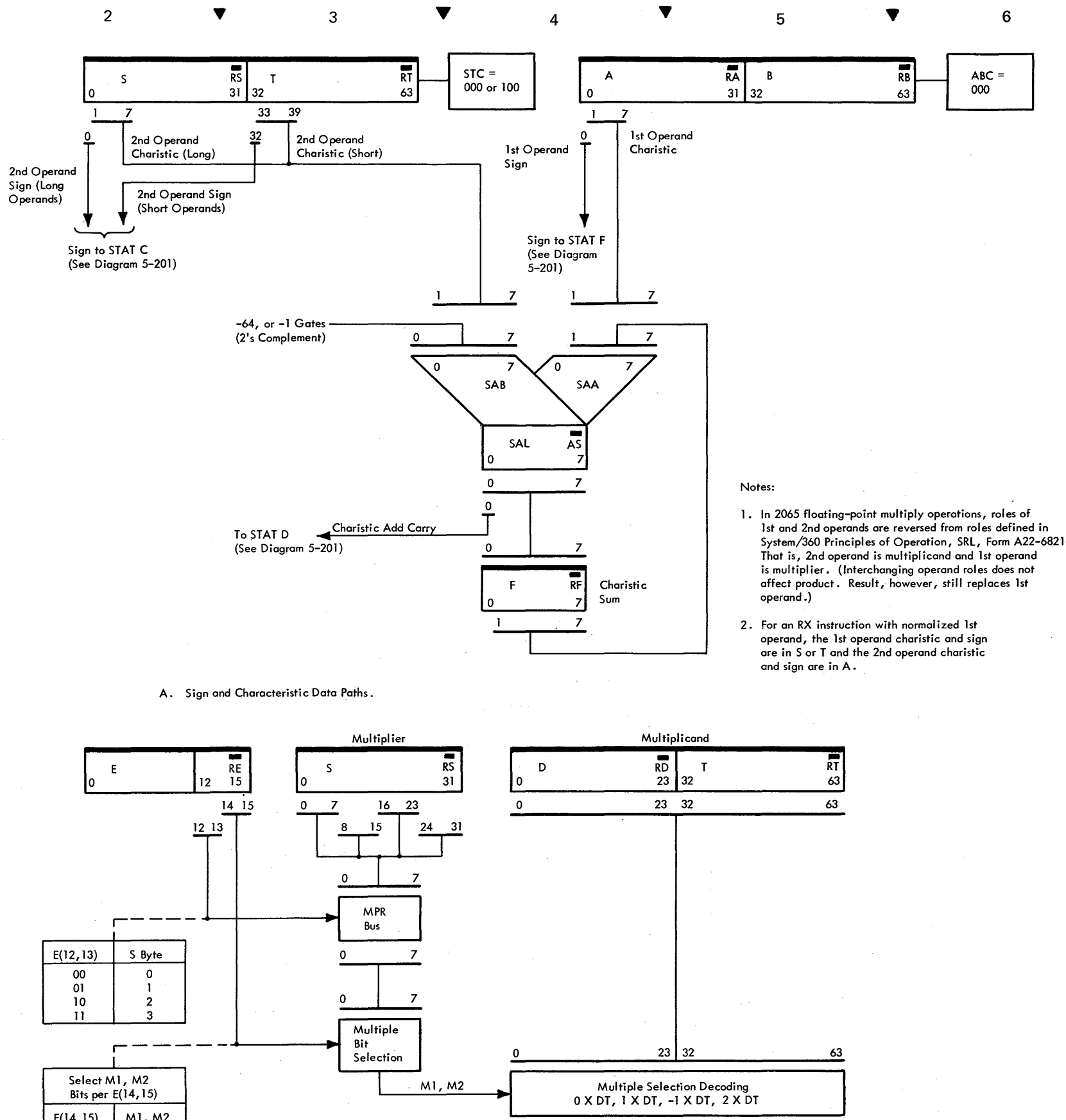
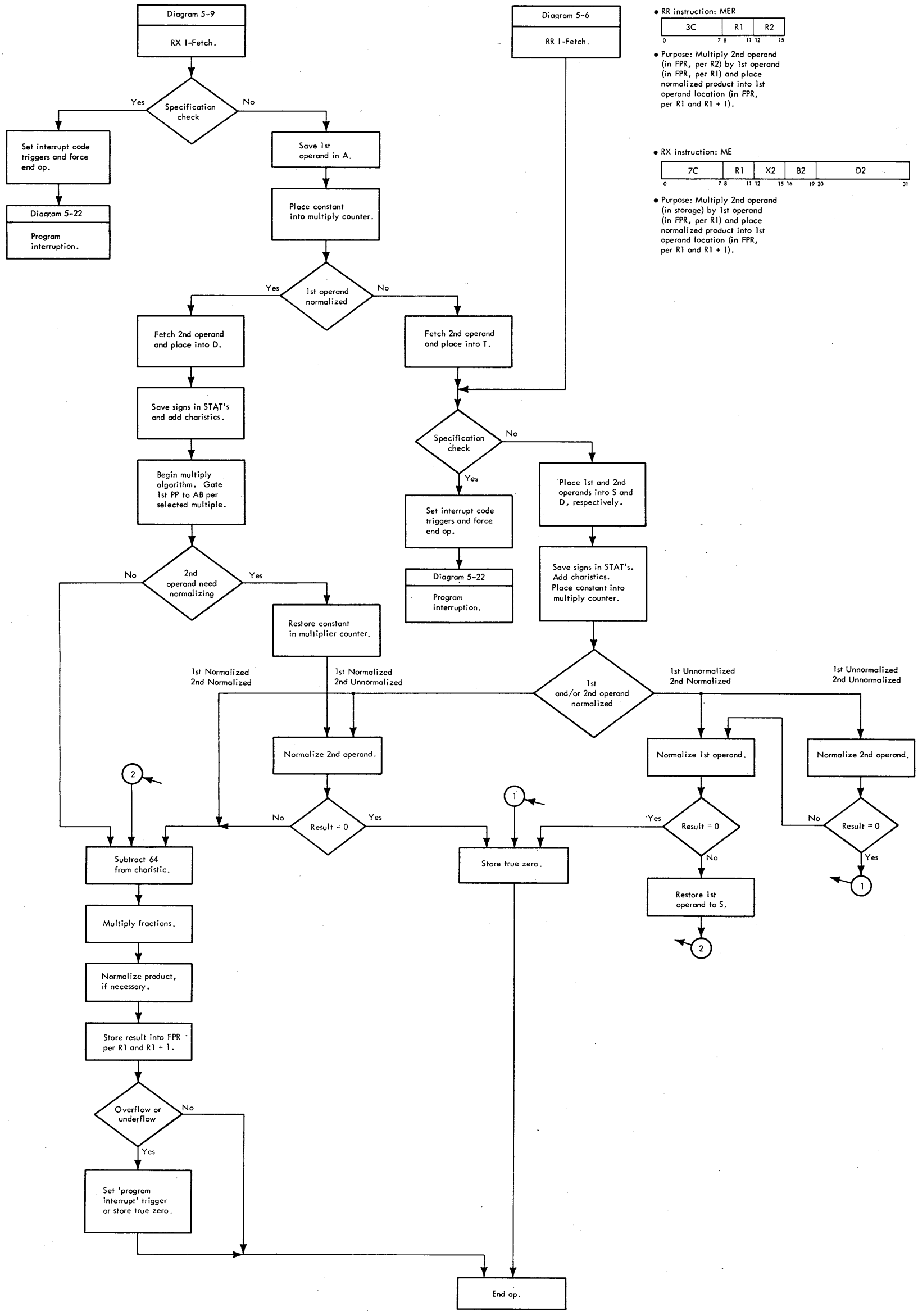


Diagram 5-210. Floating-Point Multiply Data Paths



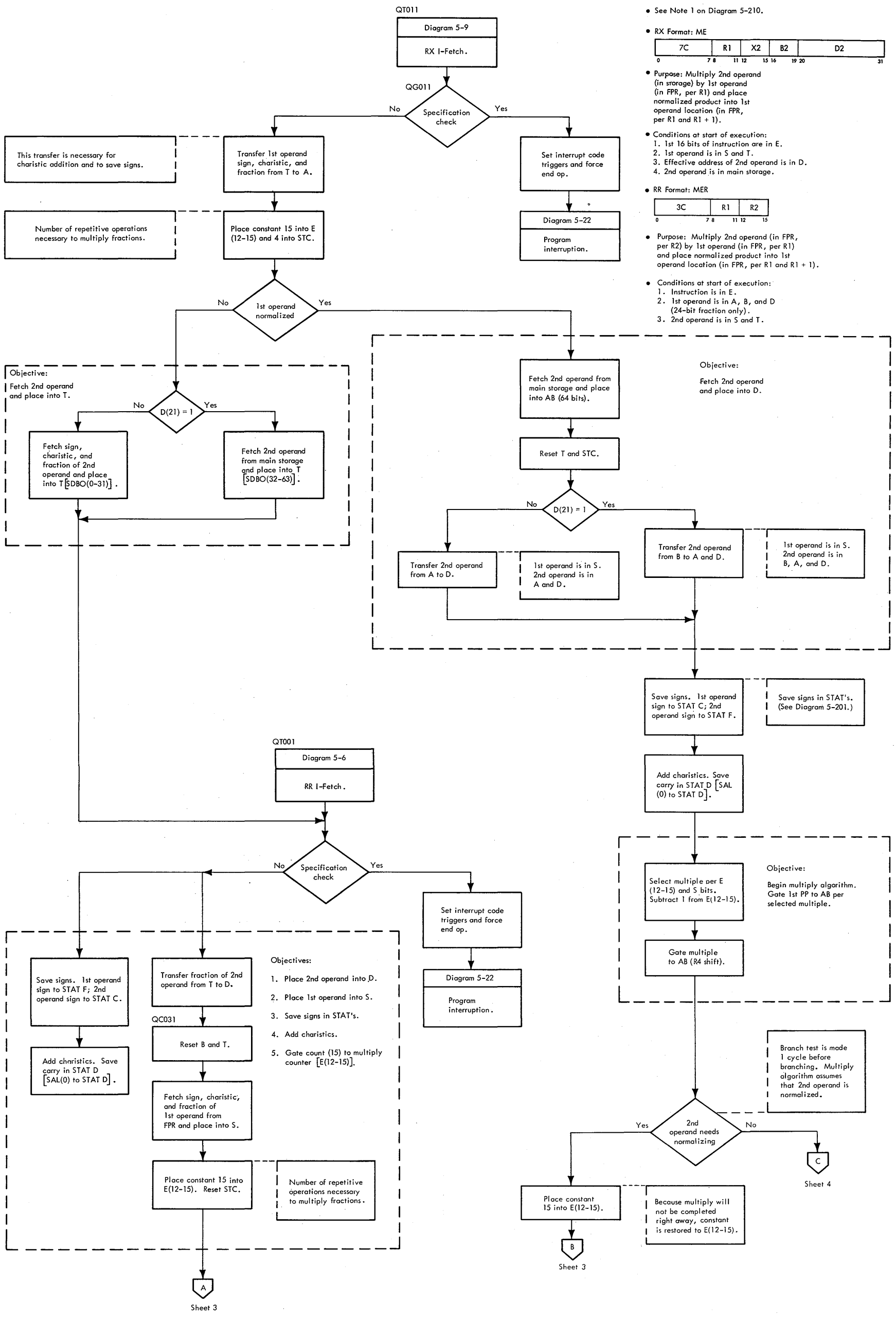
- RR instruction: MER

- Purpose: Multiply 2nd operand (in FPR, per R2) by 1st operand (in FPR, per R1) and place normalized product into 1st operand location (in FPR, per R1 and R1 + 1).

- RX instruction: ME

- Purpose: Multiply 2nd operand (in storage) by 1st operand (in FPR, per R1) and place normalized product into 1st operand location (in FPR, per R1 and R1 + 1).

Diagram 5-211. Floating-Point Multiply, Short Operands (Sheet 1 of 4)



- See Note 1 on Diagram 5-210.
- RX Format: ME

7C	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

- Purpose: Multiply 2nd operand (in storage) by 1st operand (in FPR, per R1) and place normalized product into 1st operand location (in FPR, per R1 and R1 + 1).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Effective address of 2nd operand is in D.
  4. 2nd operand is in main storage.
- RR Format: MER

3C	R1	R2
0 7	8 11	12 15

- Purpose: Multiply 2nd operand (in FPR, per R2) by 1st operand (in FPR, per R1) and place normalized product into 1st operand location (in FPR, per R1 and R1 + 1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (24-bit fraction only).
  3. 2nd operand is in S and T.

Diagram 5-211. Floating-Point Multiply, Short Operands (Sheet 2 of 4)

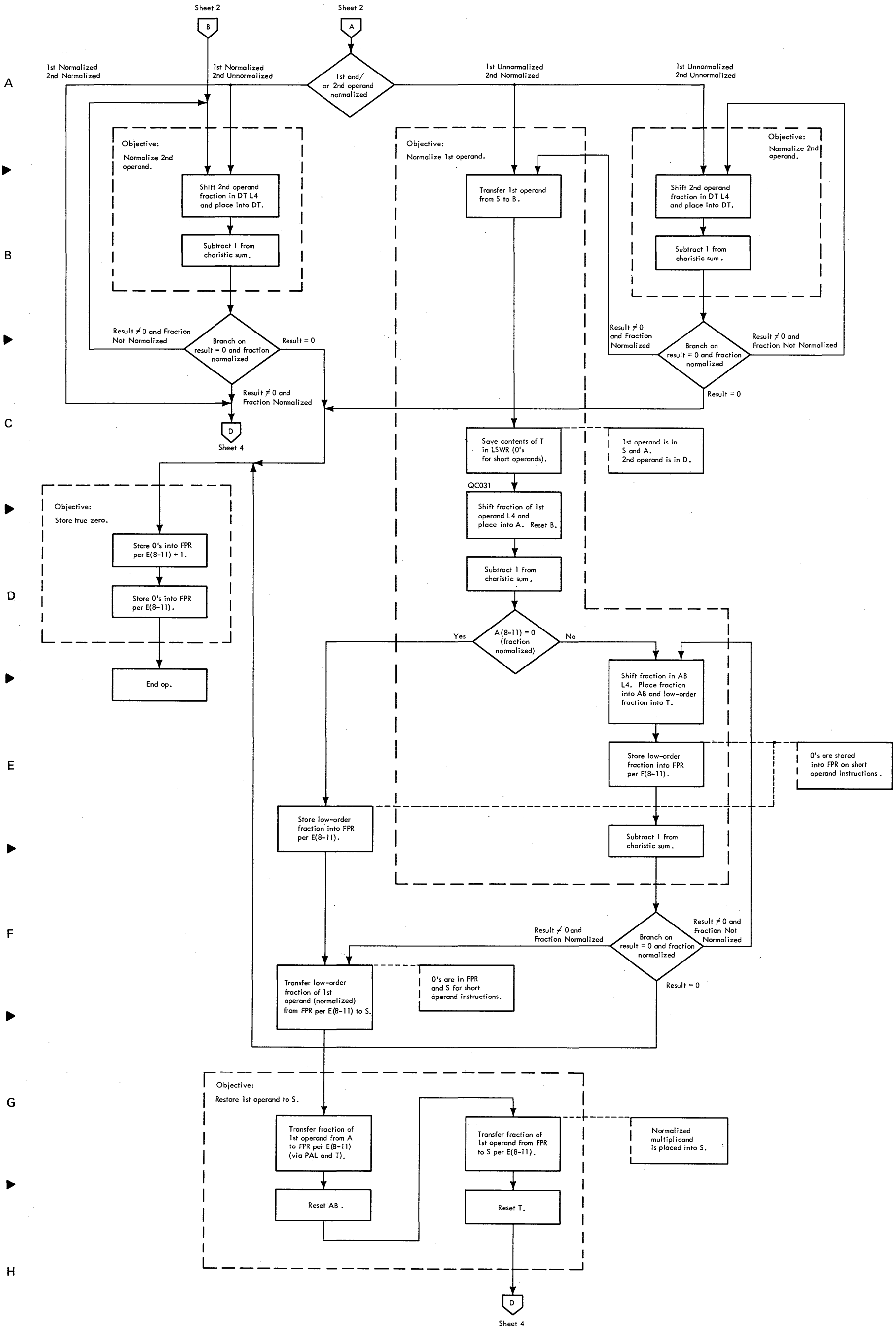
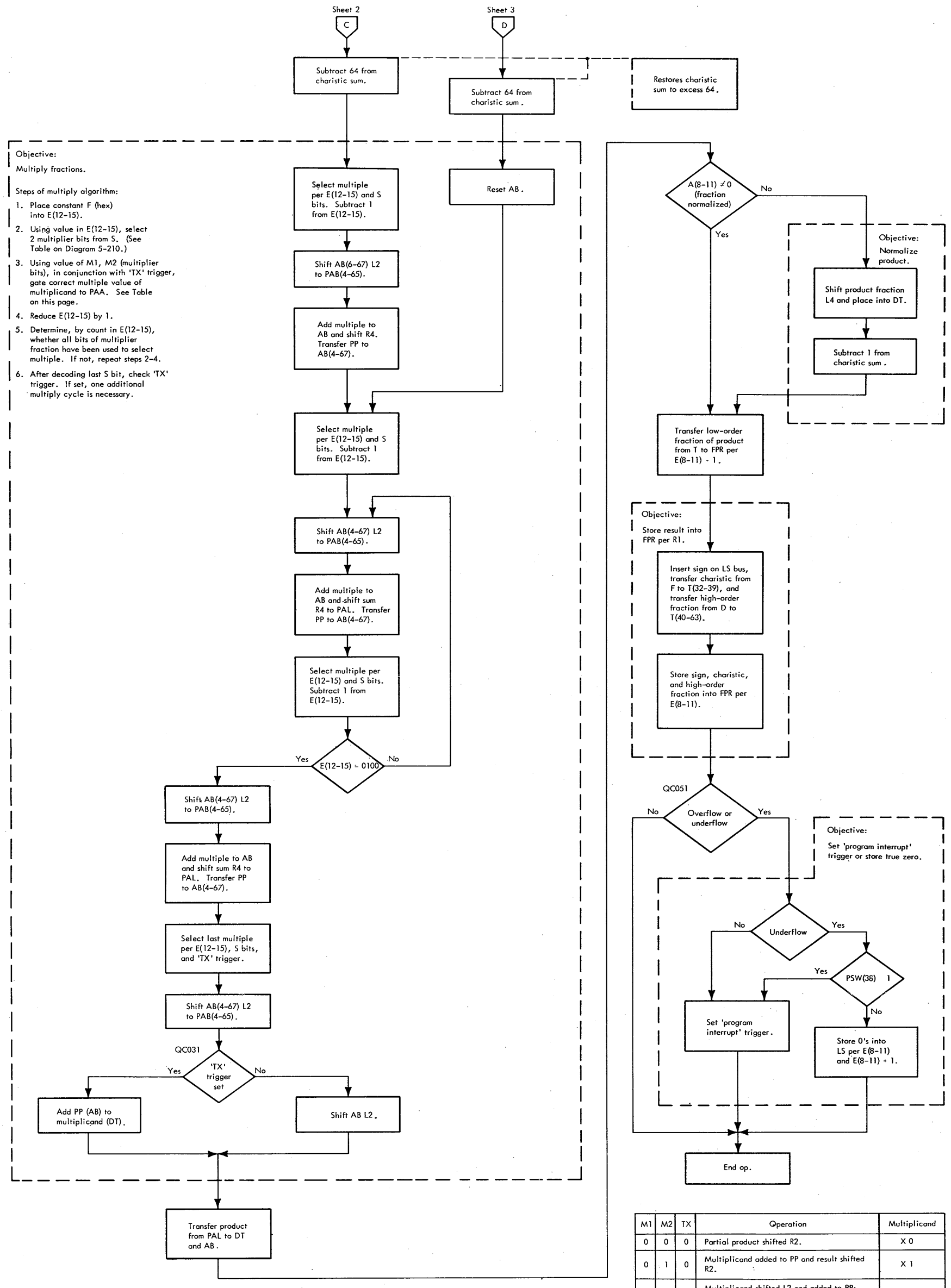


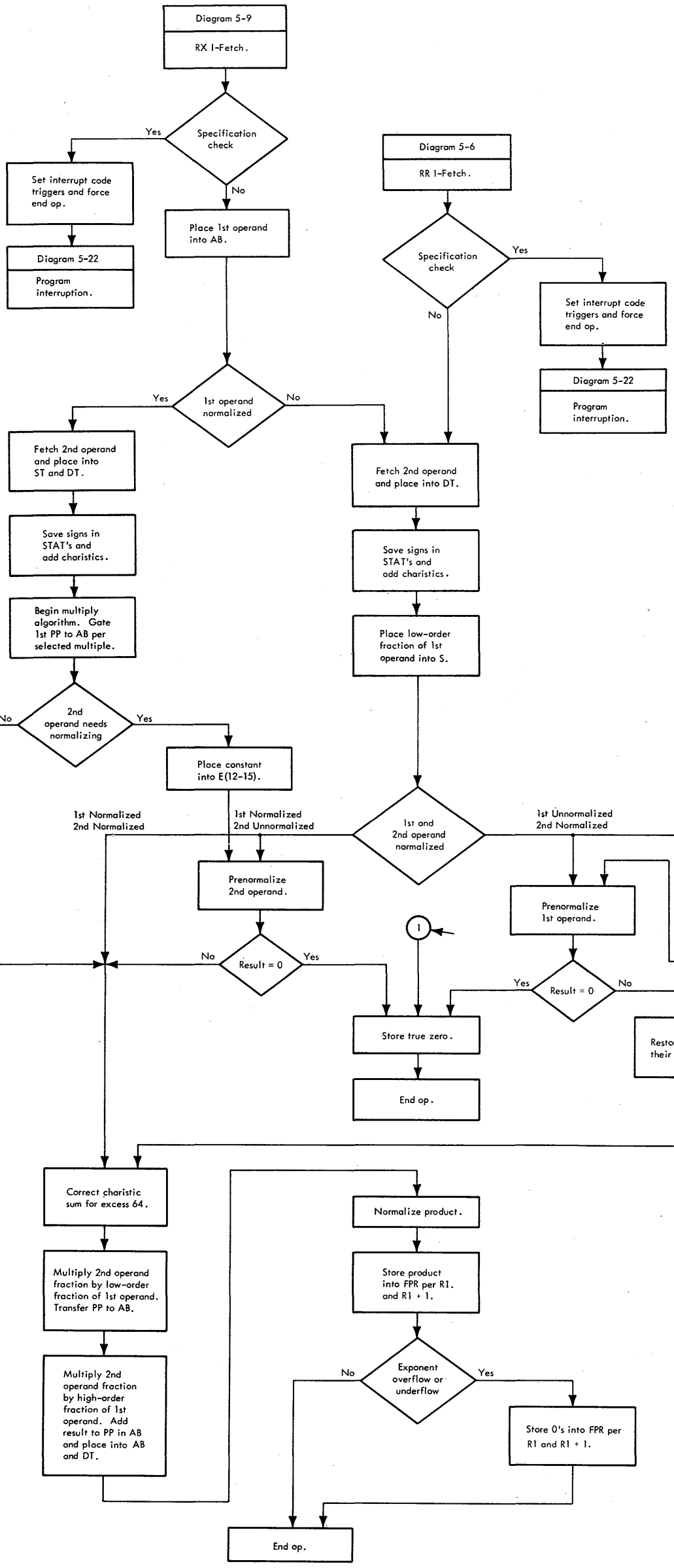
Diagram 5-211. Floating-Point, Short Operands (Sheet 3 of 4)



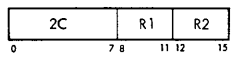
M1	M2	TX	Operation	Multiplicand
0	0	0	Partial product shifted R2.	X 0
0	1	0	Multiplicand added to PP and result shifted R2.	X 1
1	0	0	Multiplicand shifted L2 and added to PP; result shifted R2.	X 2
1	1	0	2's complement of multiplicand added to partial product, 'TX' trigger set, and result shifted R2.	X 3 (X -1 and 'TX')
0	0	1	Multiplicand added to PP and result shifted R2.	X 1
0	1	1	Multiplicand shifted L2 and added to PP; result shifted R2.	X 2
1	0	1	2's complement of multiplicand added to PP, 'TX' trigger set, and result shifted R2.	X 3 (X -1 and 'TX')
1	1	1	Partial product shifted R2 and 'TX' trigger set	X 0 and 'TX'

Note: See Note 1 on Diagram 5-210.

Diagram 5-211. Floating-Point Multiply, Short Operands (Sheet 4 of 4)

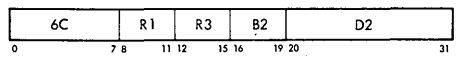


- RR instruction: MDR



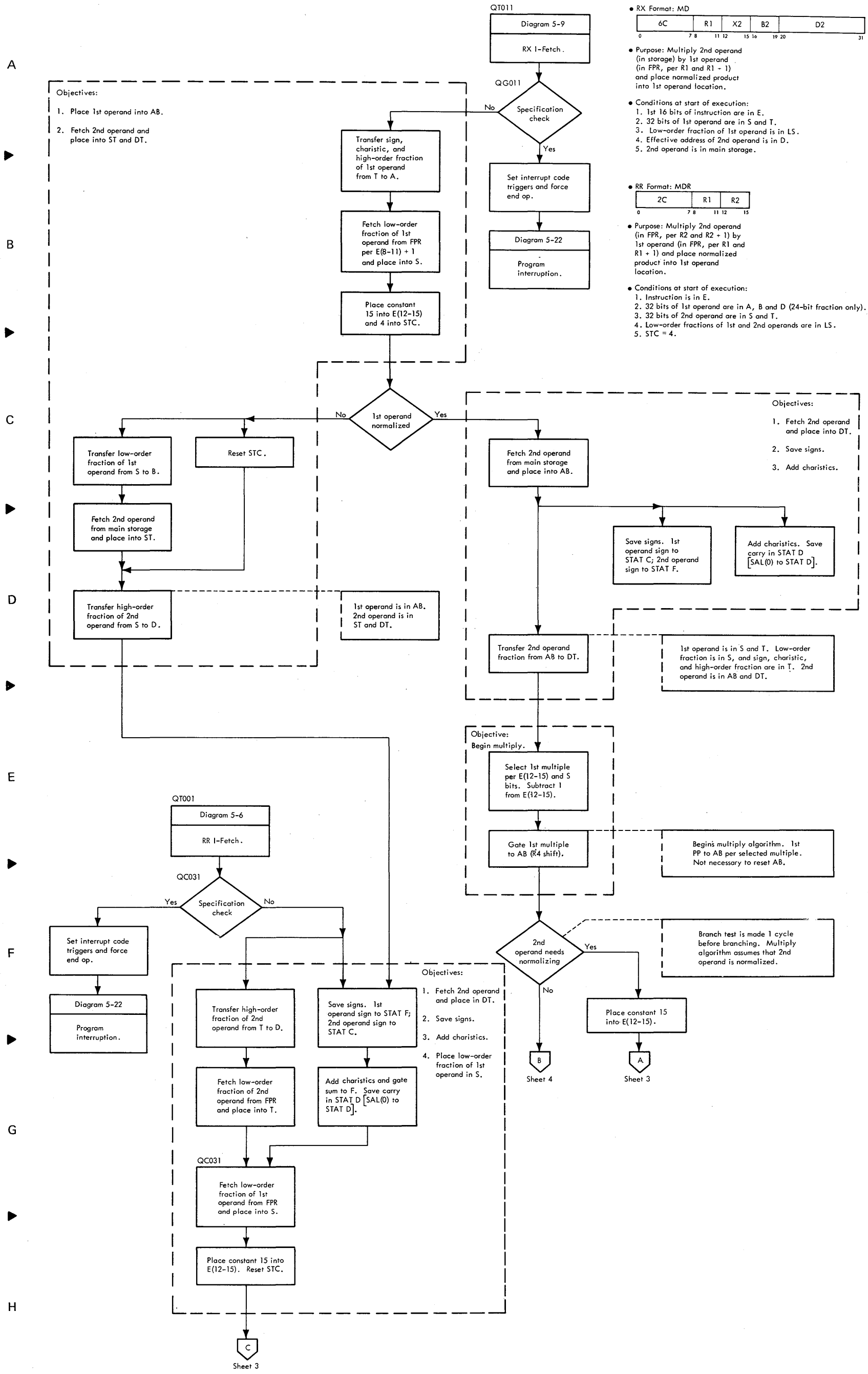
- Purpose: Multiply 2nd operand (in FPR, per R2 and R2 + 1) by 1st operand (in FPR, per R1 and R1 + 1) and place normalized product into 1st operand location.

- RX instruction: MD



- Purpose: Multiply 2nd operand (in storage) by 1st operand (in FPR, per R1 and R1 + 1) and place normalized product into 1st operand location.

Diagram 5-212. Floating-Point Multiply, Long Operands (Sheet 1 of 4)



- See Note 1 on Diagram 5-210.
- RX Format: MD

- Purpose: Multiply 2nd operand (in storage) by 1st operand (in FPR, per R1 and R1 + 1) and place normalized product into 1st operand location.
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 32 bits of 1st operand are in S and T.
  3. Low-order fraction of 1st operand is in LS.
  4. Effective address of 2nd operand is in D.
  5. 2nd operand is in main storage.

- RR Format: MDR

- Purpose: Multiply 2nd operand (in FPR, per R2 and R2 + 1) by 1st operand (in FPR, per R1 and R1 + 1) and place normalized product into 1st operand location.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 32 bits of 1st operand are in A, B and D (24-bit fraction only).
  3. 32 bits of 2nd operand are in S and T.
  4. Low-order fractions of 1st and 2nd operands are in LS.
  5. STC = 4.

Diagram 5-212. Floating-Point Multiply, Long Operands (Sheet 2 of 4)

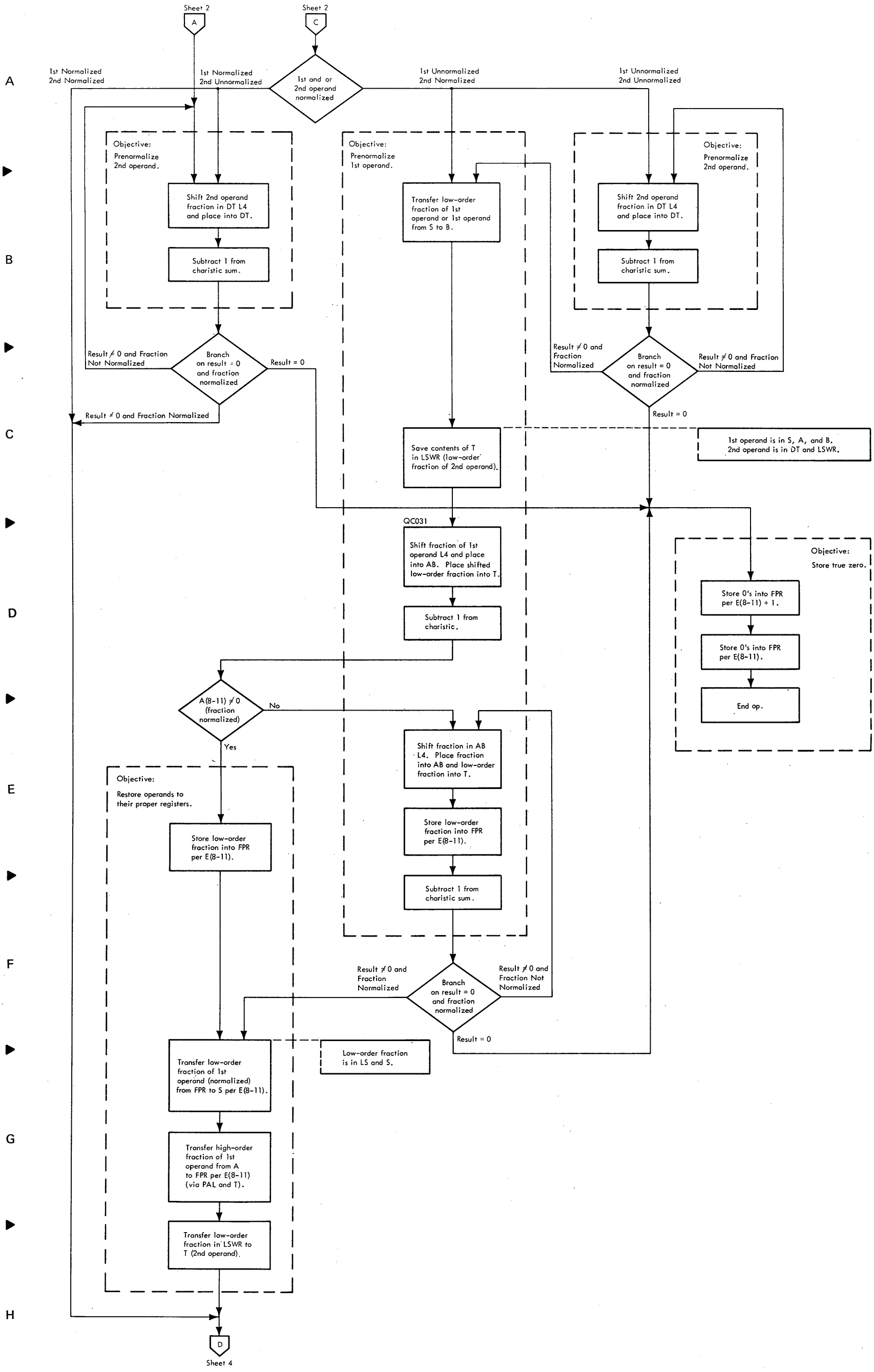
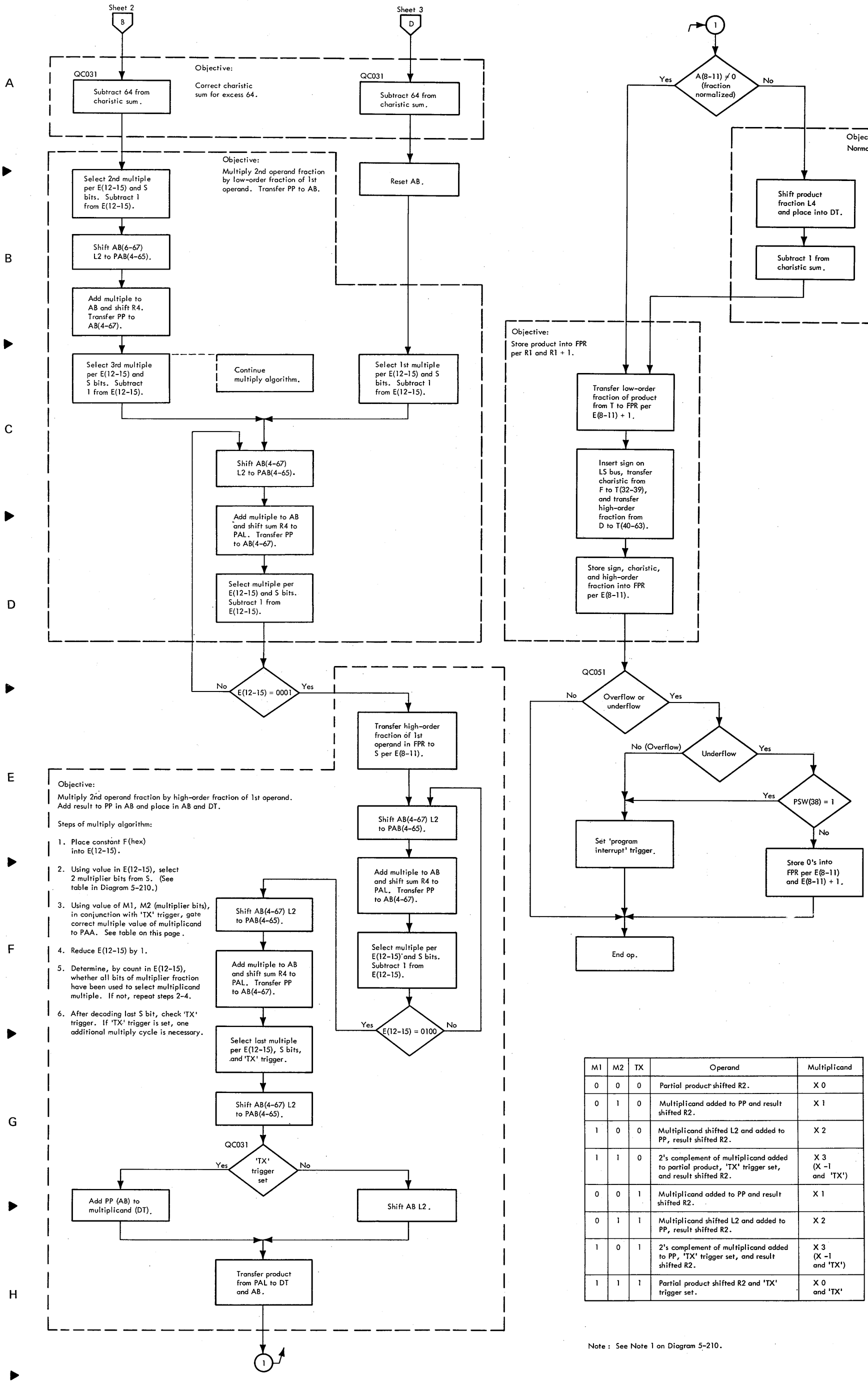


Diagram 5-212. Floating-Point Multiply, Long Operands (Sheet 3 of 4)



M1	M2	TX	Operand	Multiplicand
0	0	0	Partial product shifted R2.	X 0
0	1	0	Multiplicand added to PP and result shifted R2.	X 1
1	0	0	Multiplicand shifted L2 and added to PP, result shifted R2.	X 2
1	1	0	2's complement of multiplicand added to partial product, 'TX' trigger set, and result shifted R2.	X 3 (X -1 and 'TX')
0	0	1	Multiplicand added to PP and result shifted R2.	X 1
0	1	1	Multiplicand shifted L2 and added to PP, result shifted R2.	X 2
1	0	1	2's complement of multiplicand added to PP, 'TX' trigger set, and result shifted R2.	X 3 (X -1 and 'TX')
1	1	1	Partial product shifted R2 and 'TX' trigger set.	X 0 and 'TX'

Note : See Note 1 on Diagram 5-210.

Diagram 5-212. Floating-Point Multiply, Long Opernads (Sheet 4 of 4)

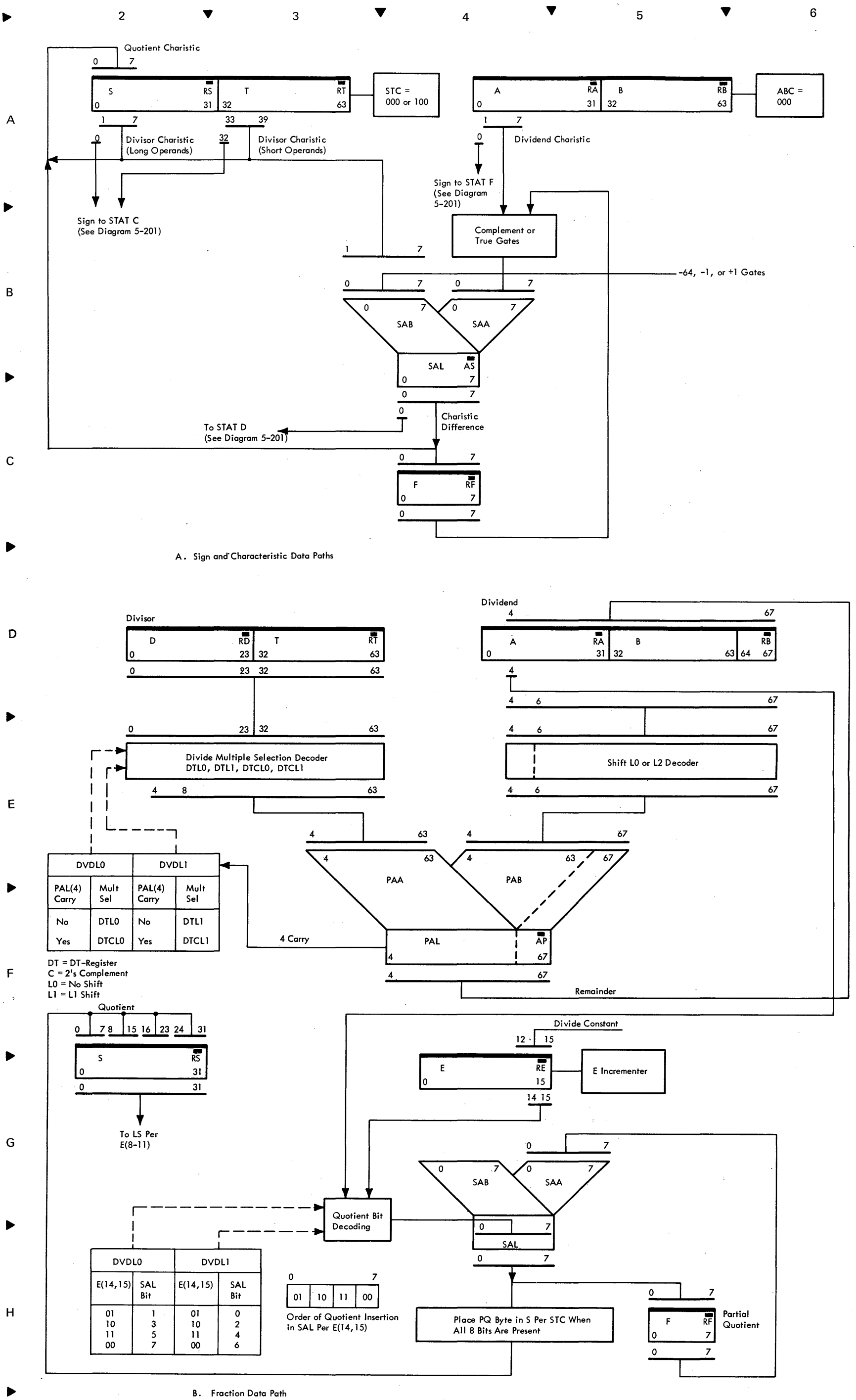
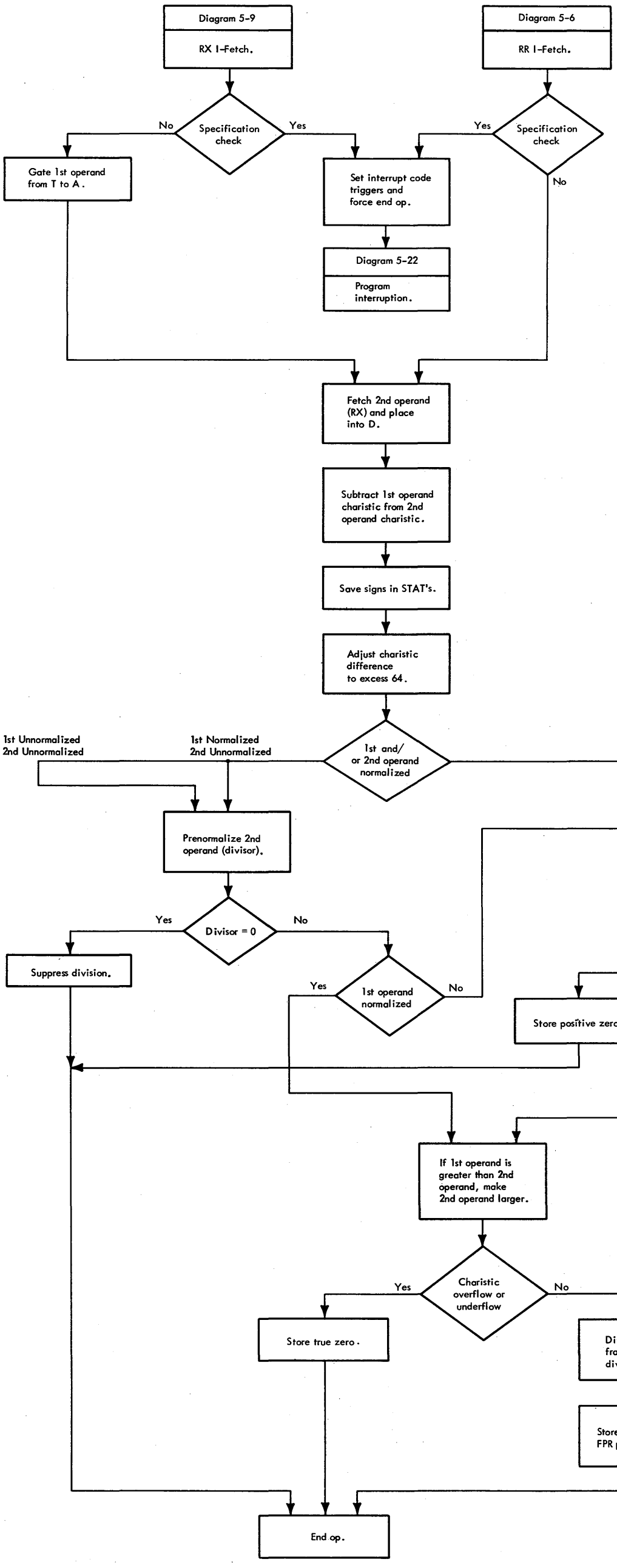
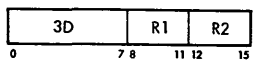


Diagram 5-213. Floating-Point Divide Data Paths

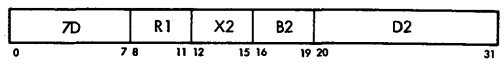


- RR instruction: DER



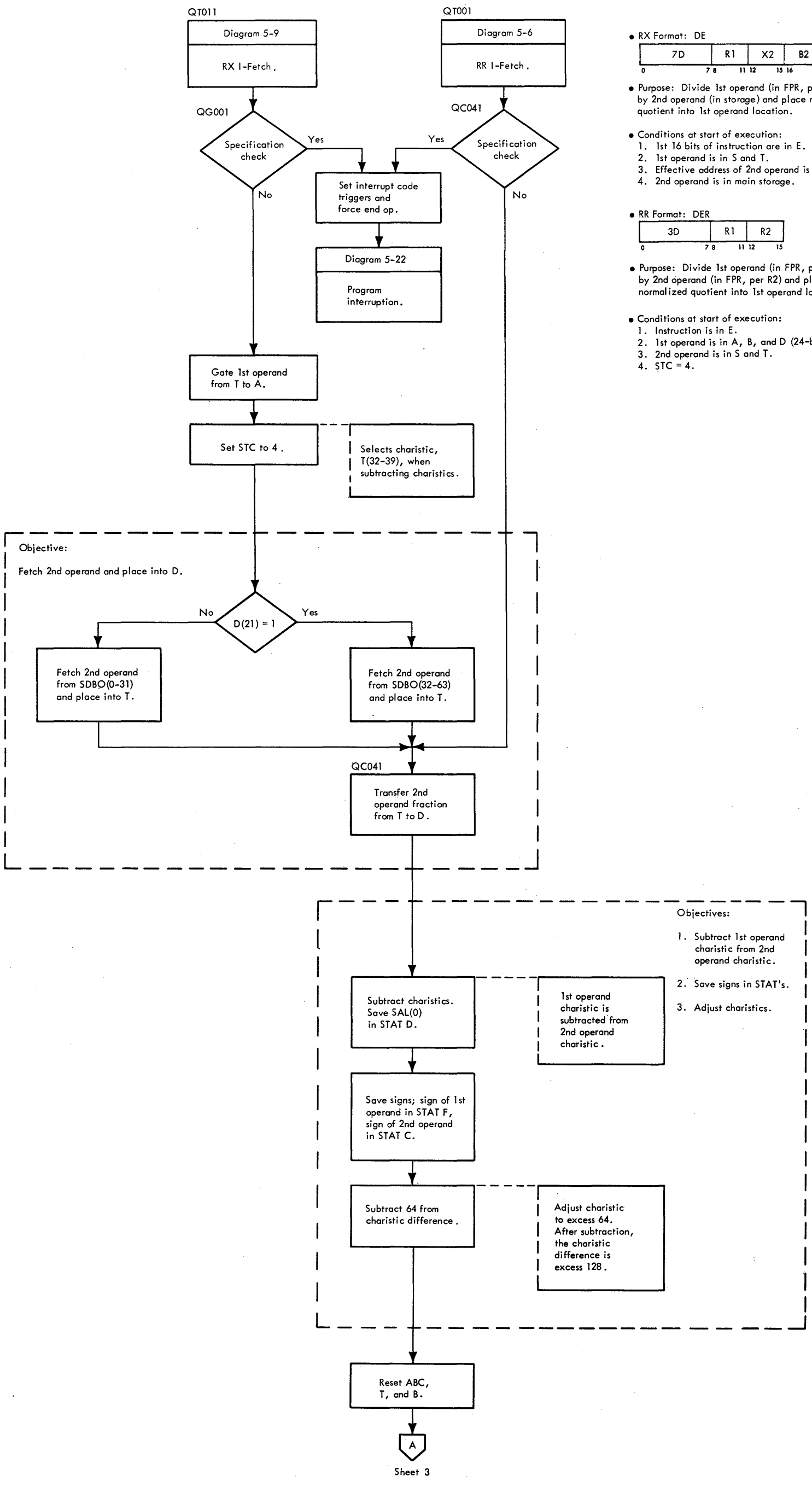
- Purpose: Divide 1st operand (in FPR, per R1) by 2nd operand (in FPR, per R2) and place normalized quotient into 1st operand location.

- RX instruction: DE



- Purpose: Divide 1st operand (in FPR, per R1) by 2nd operand (in storage) and place normalized quotient into 1st operand location.

Diagram 5-214. Floating-Point Divide, Short Operands (Sheet 1 of 4)



- RX Format: DE

- Purpose: Divide 1st operand (in FPR, per R1) by 2nd operand (in storage) and place normalized quotient into 1st operand location.

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Effective address of 2nd operand is in D.
  4. 2nd operand is in main storage.

- RR Format: DER

- Purpose: Divide 1st operand (in FPR, per R1) by 2nd operand (in FPR, per R2) and place normalized quotient into 1st operand location.

- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D (24-bit fraction only).
  3. 2nd operand is in S and T.
  4. STC = 4.

Figure 5-214. Floating-Point Divide, Short Operands (Sheet 2 of 4)

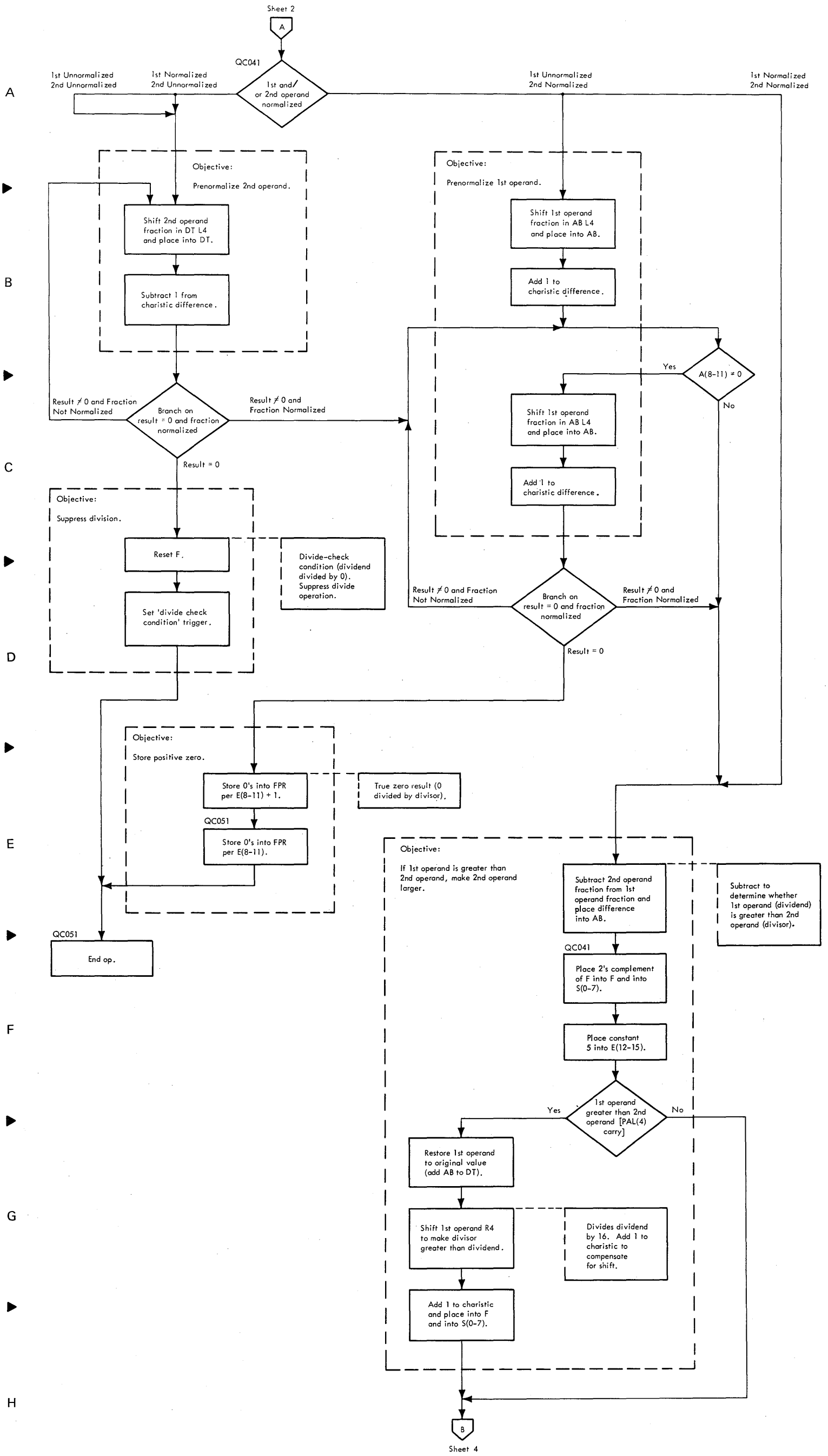


Diagram 5-214. Floating-Point Divide, Short Operands (Sheet 3 of 4)

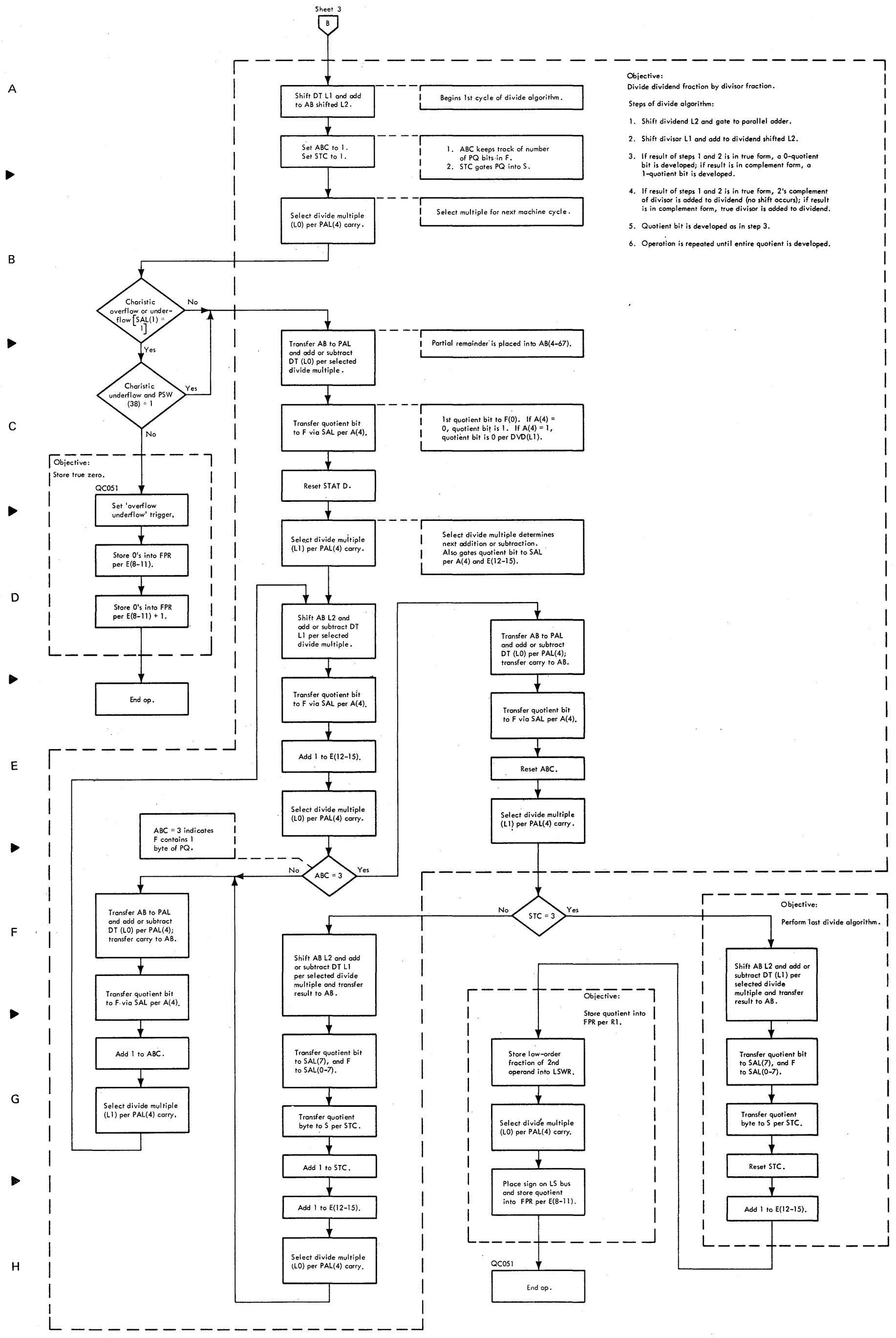
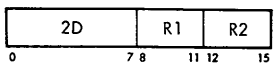
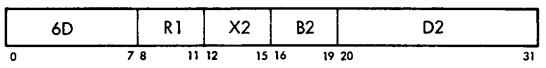


Diagram 5-214. Floating-Point Divide, Short Operands (Sheet 4 of 4)

- RR instruction: DDR

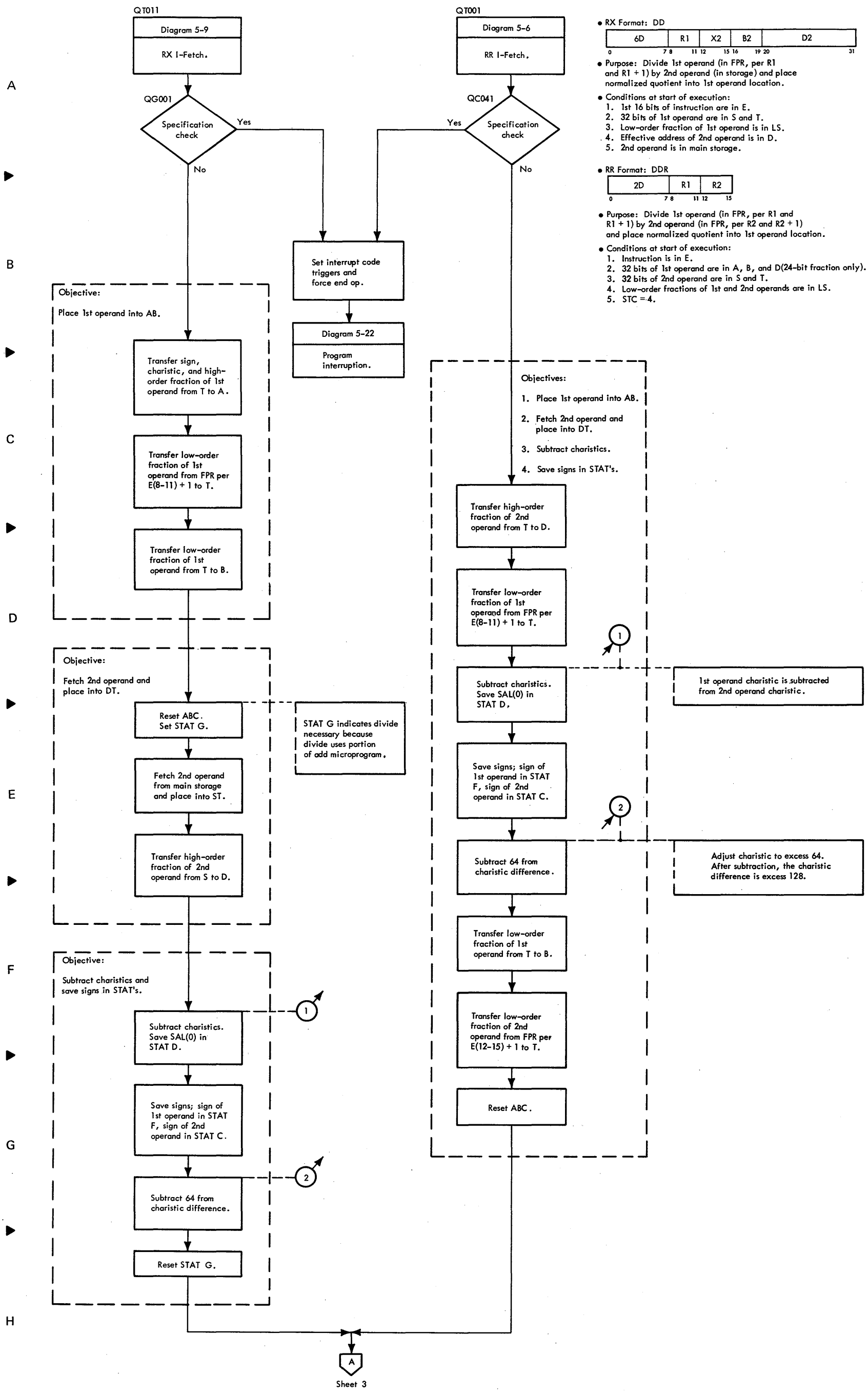


- Purpose: Divide 1st operand (in FPR, per R1 and R1 + 1) by 2nd operand (in FPR, per R2 and R2 + 1) and place normalized quotient into 1st operand location.
- RX instruction: DD



- Purpose: Divide 1st operand (in FPR, per R1 and R1 + 1) by 2nd operand (in storage) and place normalized quotient into 1st operand location.

Diagram 5-215. Floating-Point Divide, Long Operands (Sheet 1 of 5)



- RX Format: DD

- Purpose: Divide 1st operand (in FPR, per R1 and R1 + 1) by 2nd operand (in storage) and place normalized quotient into 1st operand location.
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 32 bits of 1st operand are in S and T.
  3. Low-order fraction of 1st operand is in LS.
  4. Effective address of 2nd operand is in D.
  5. 2nd operand is in main storage.

- RR Format: DDR

- Purpose: Divide 1st operand (in FPR, per R1 and R1 + 1) by 2nd operand (in FPR, per R2 and R2 + 1) and place normalized quotient into 1st operand location.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 32 bits of 1st operand are in A, B, and D(24-bit fraction only).
  3. 32 bits of 2nd operand are in S and T.
  4. Low-order fractions of 1st and 2nd operands are in LS.
  5. STC = 4.

Diagram 5-215. Floating-Point Divide, Long Operands (Sheet 2 of 5)

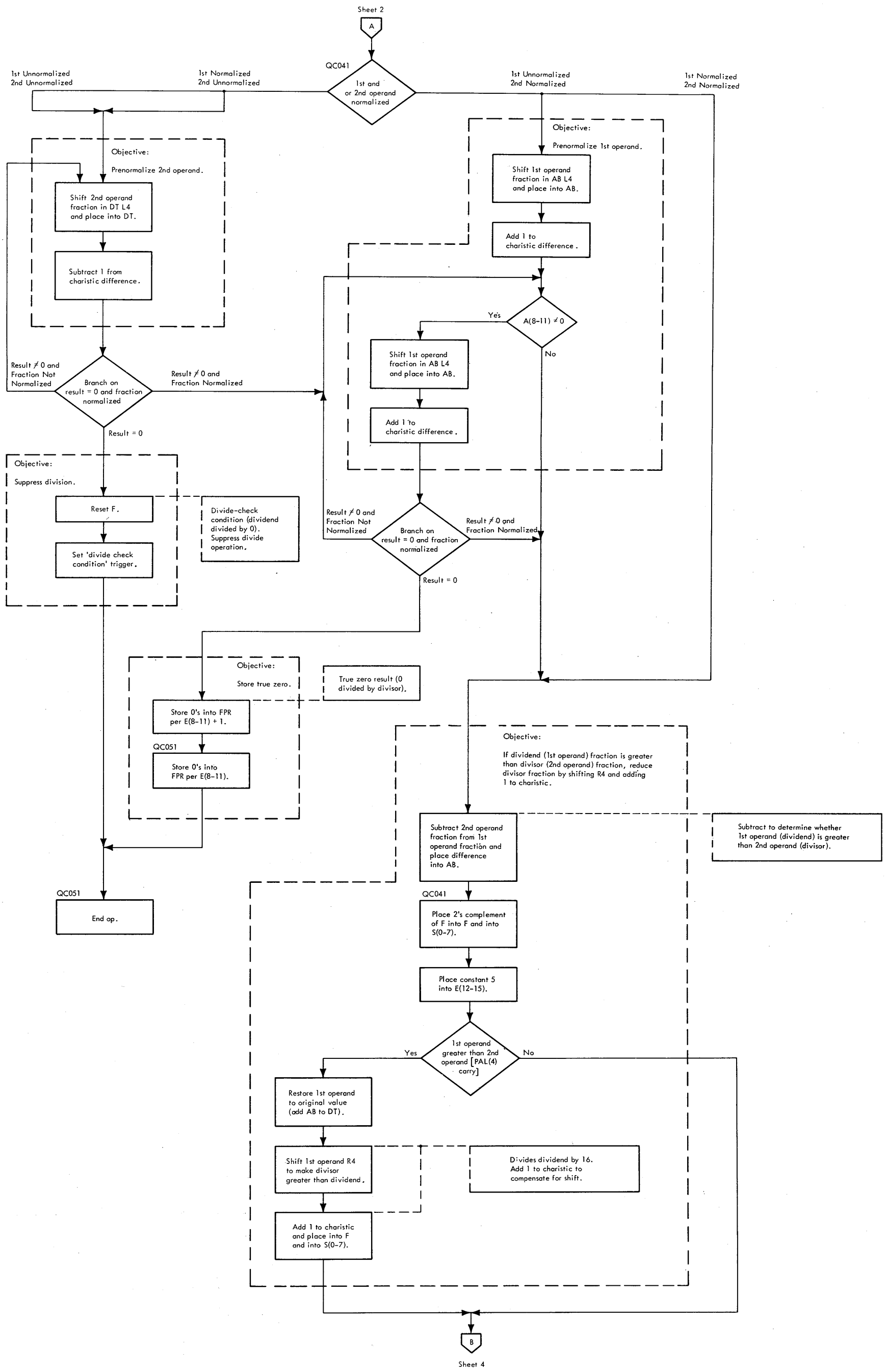


Diagram 5-215. Floating-Point Divide, Long Operands (Sheet 3 of 5)

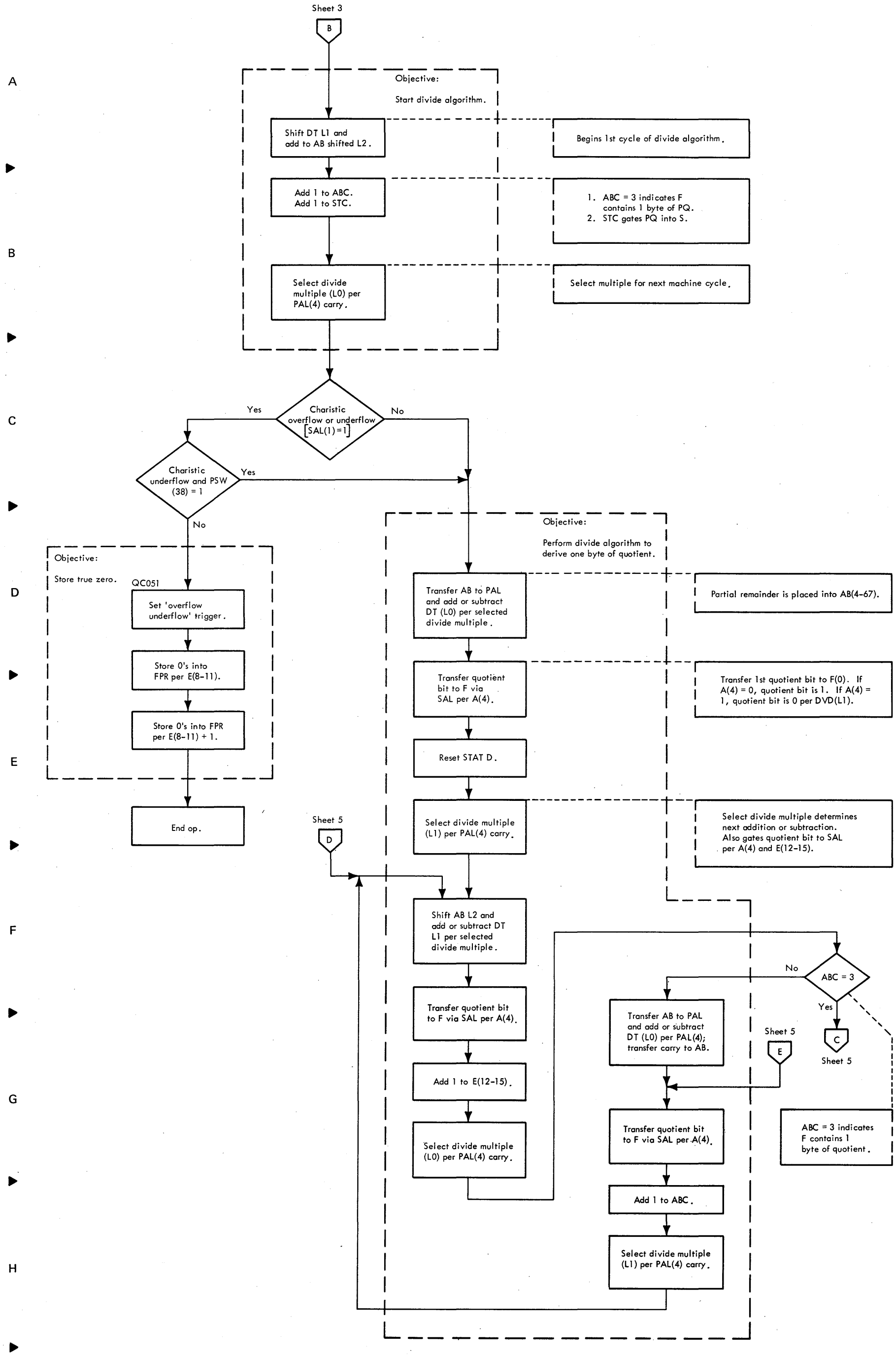


Diagram 5-215. Floating-Point Divide, Long Operands (Sheet 4 of 5)

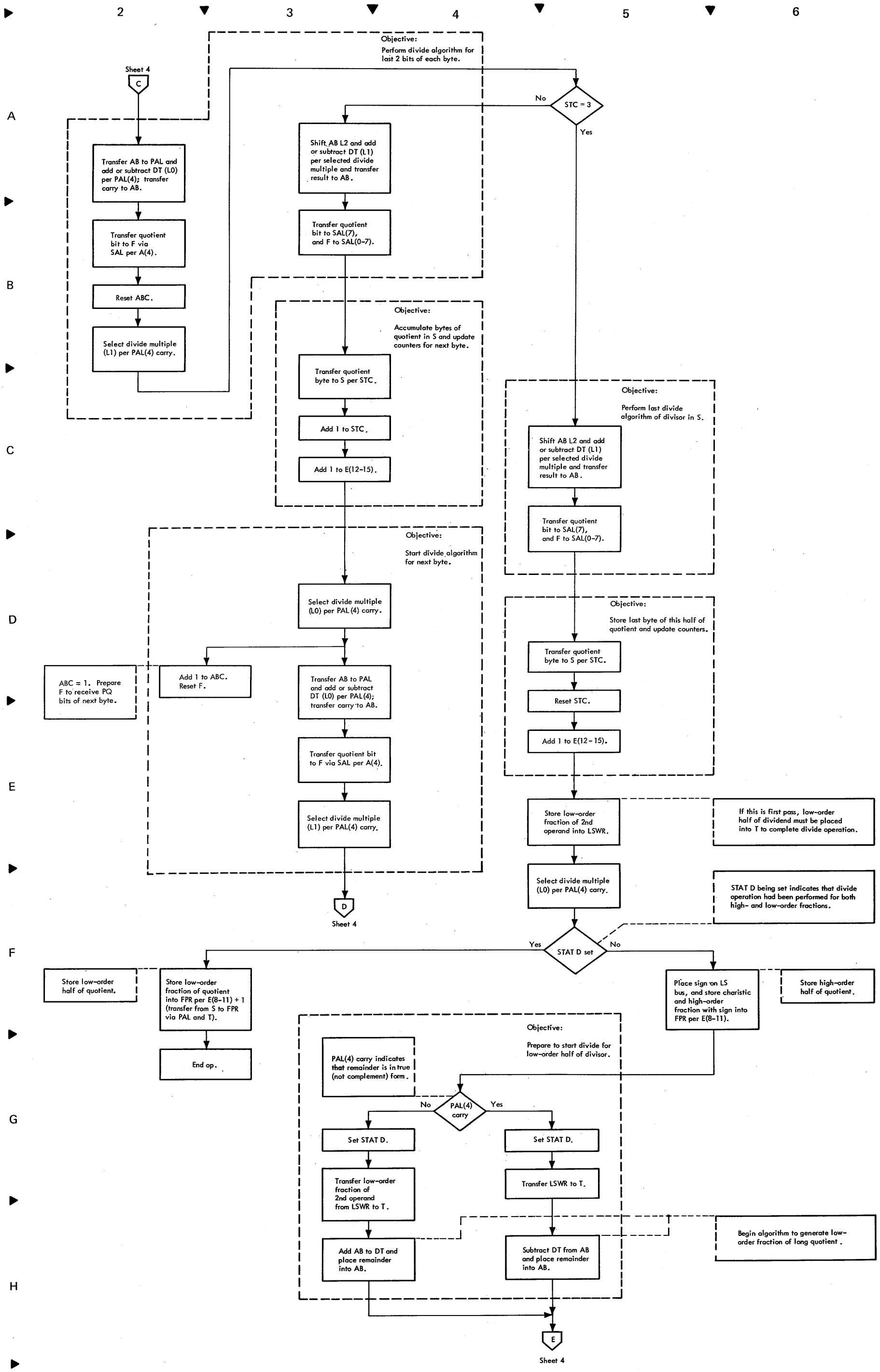
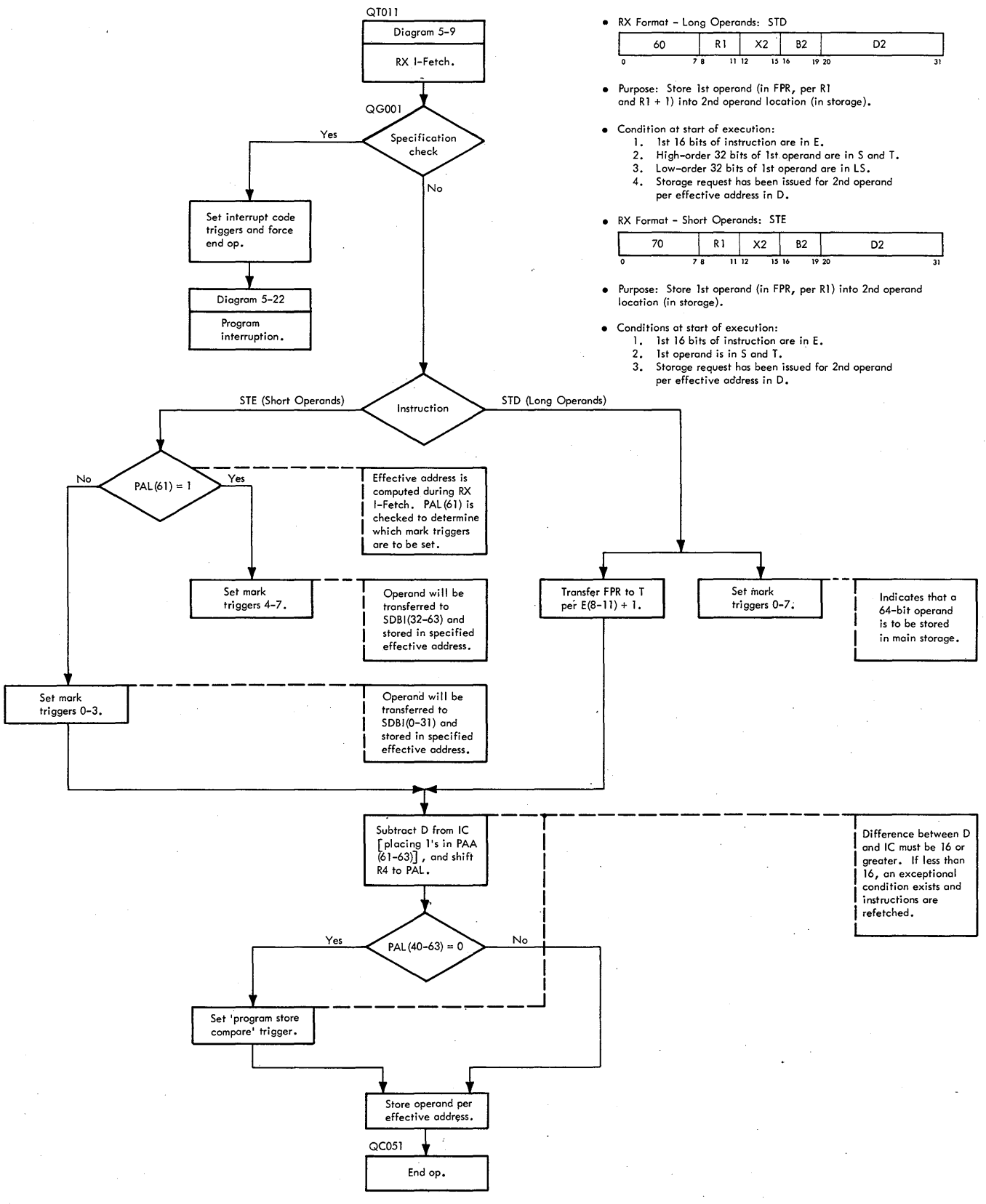


Diagram 5-215. Floating-Point Divide, Long Operands (Sheet 5 of 5)



- RX Format - Long Operands: STD

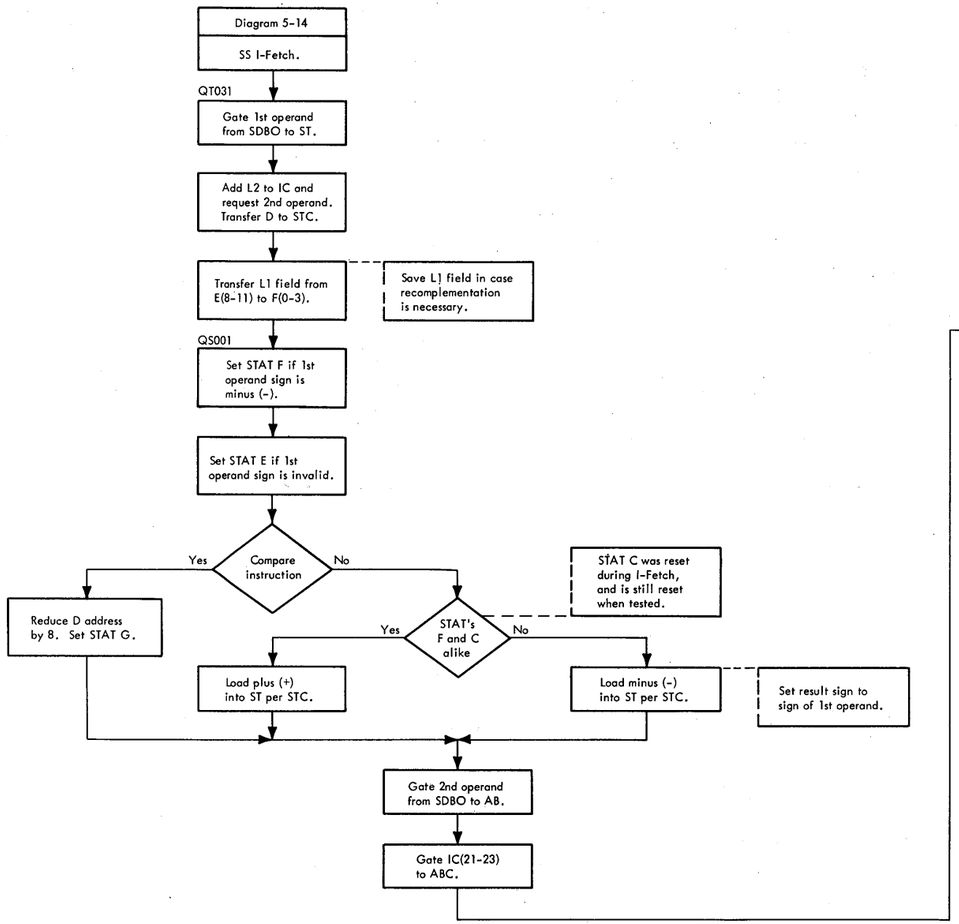
60	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

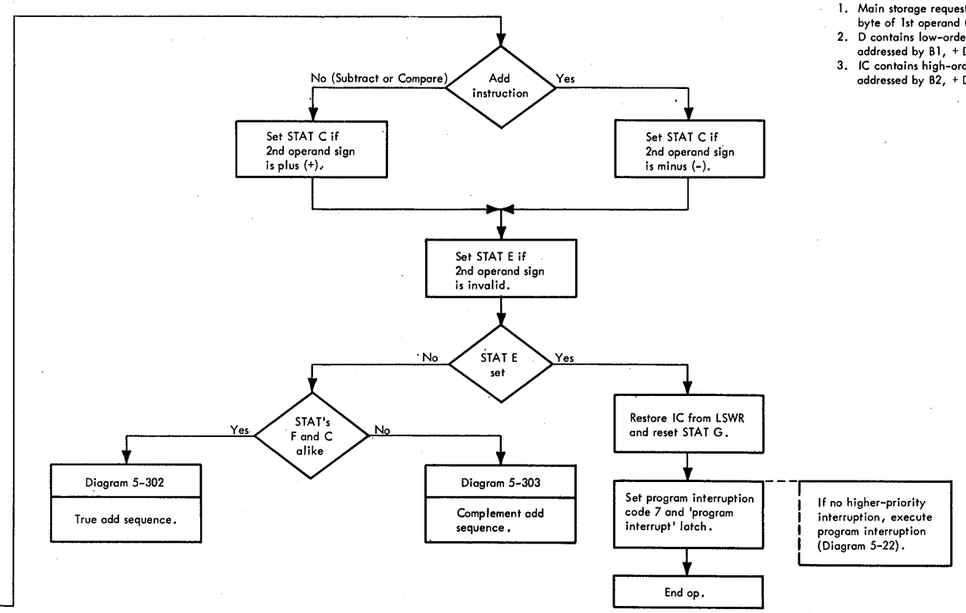
- Purpose: Store 1st operand (in FPR, per R1 and R1 + 1) into 2nd operand location (in storage).
- Condition at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. High-order 32 bits of 1st operand are in S and T.
  3. Low-order 32 bits of 1st operand are in LS.
  4. Storage request has been issued for 2nd operand per effective address in D.
- RX Format - Short Operands: STE

70	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

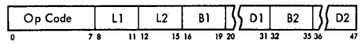
- Purpose: Store 1st operand (in FPR, per R1) into 2nd operand location (in storage).
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. 1st operand is in S and T.
  3. Storage request has been issued for 2nd operand per effective address in D.

Diagram 5-216. Store, STE (70) – Short Operands; Store, STD (60) – Long Operands





- SS Format:



Op Code:
1. Add Decimal (AP) - FA.
2. Subtract Decimal (SP) - FB.
3. Compare Decimal (CP) - F9.

- Purpose:
1. Add Decimal, Subtract Decimal - Algebraically add (subtract) 2nd operand (in storage) to (from) 1st operand (in storage) and place result into 1st operand location.
2. Compare Decimal - Algebraically compare 1st operand (in storage) with 2nd operand (in storage) and set CC according to result.

- Conditions at end of I-Fetch:
1. Main storage request for doubleword containing low-order byte of 1st operand (destination) has been issued per D.
2. D contains low-order byte address (contents of GPR addressed by B1, + D1 + L1) of 1st operand.
3. IC contains high-order byte address (contents of GPR addressed by B2, + D2) of 2nd operand.

Decimal Add Instruction

Sign of 1st Operand	Sign of 2nd Operand	STAT F	STAT C at 1st Test	STAT F Equals STAT C	Set Sign of Result	STAT C at 2nd Test	STAT F Equals STAT C	Form of 1st Operand to SAB	Form of 2nd Operand to SAA	Form of Result† 1st Operand ≧ 2nd Operand	Form of Result 1st Operand < 2nd Operand
+	+	0	0	Yes	+	0	Yes	True	True +6	True	True
+	-	0	0	Yes	+	1	No	True	2's complement	True	10's complement*
-	+	1	0	No	-	0	No	True	2's complement	True	10's complement*
-	-	1	0	No	-	1	Yes	True	True +6	True	True

Decimal Subtract Instruction

Sign of 1st Operand	Sign of 2nd Operand	STAT F	STAT C at 1st Test	STAT F Equals STAT C	Set Sign of Result	STAT C at 2nd Test	STAT F Equals STAT C	Form of 1st Operand to SAB	Form of 2nd Operand to SAA	Form of Result† 1st Operand ≧ 2nd Operand	Form of Result 1st Operand < 2nd Operand
+	+	0	0	Yes	+	1	No	True	2's complement	True	10's complement*
+	-	0	0	Yes	+	0	Yes	True	True +6	True	True
-	+	1	0	No	-	1	Yes	True	True +6	True	True
-	-	1	0	No	-	0	No	True	2's complement	True	10's complement*

†Result sign set to plus for zero result.

*Result recomplemented and sign inverted.

Diagram 5-301. GIS for Decimal Add, Subtract, and Compare

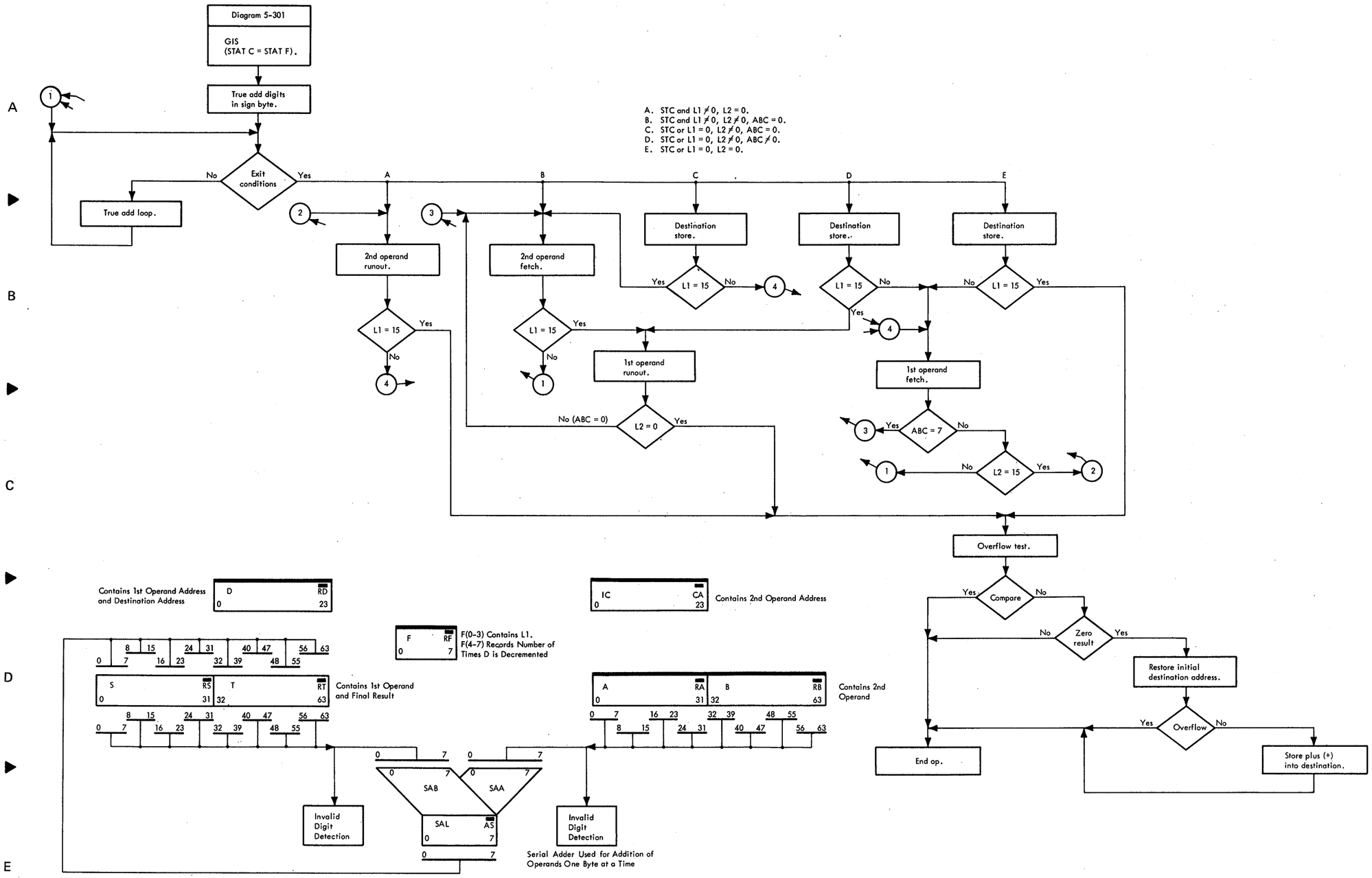


Diagram 5-302. True Add Sequence for Decimal Add, Subtract, and Compare (Sheet 1 of 3)

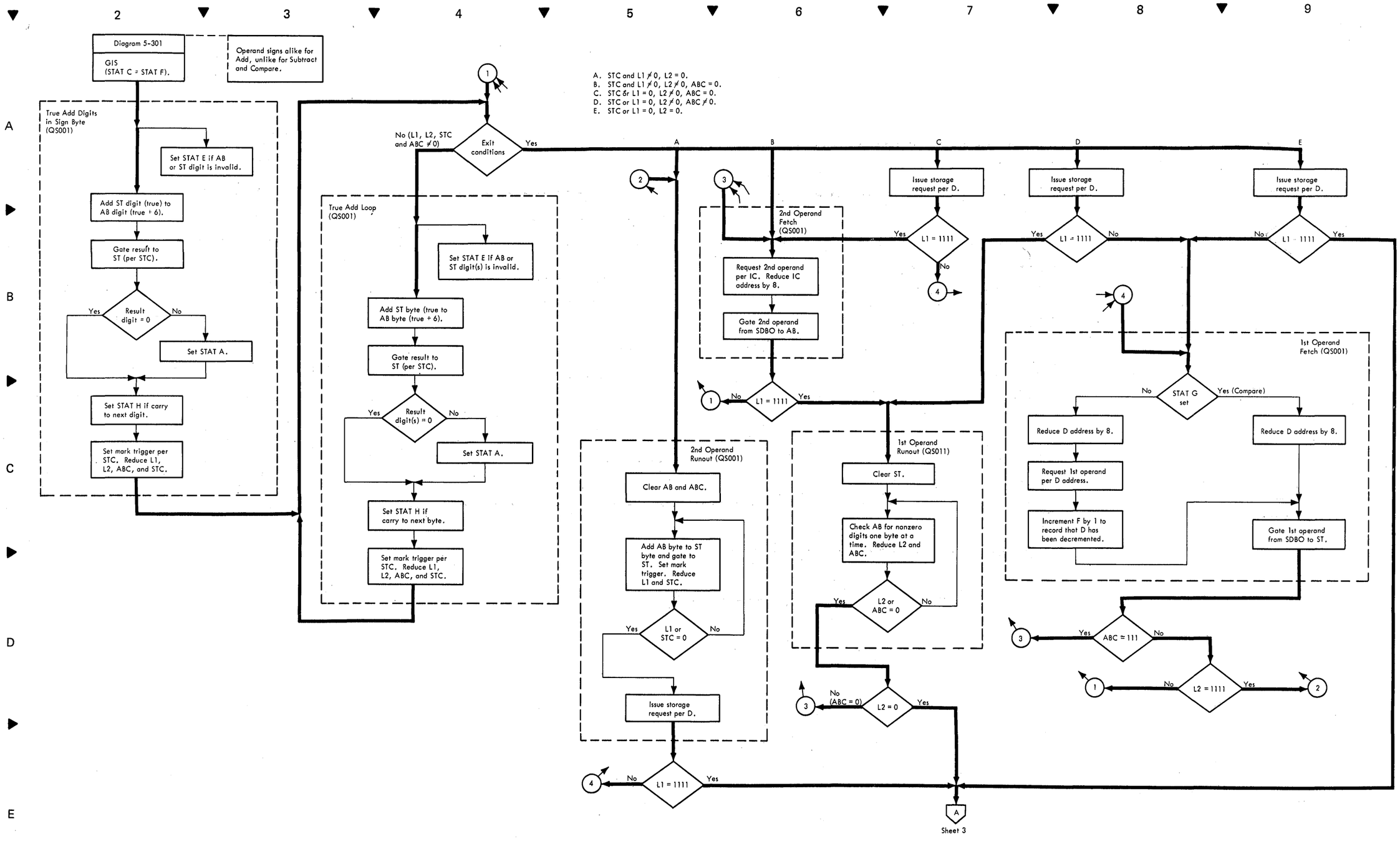


Diagram 5-302. True Add Sequence for Decimal Add, Subtract, and Compare (Sheet 2 of 3)

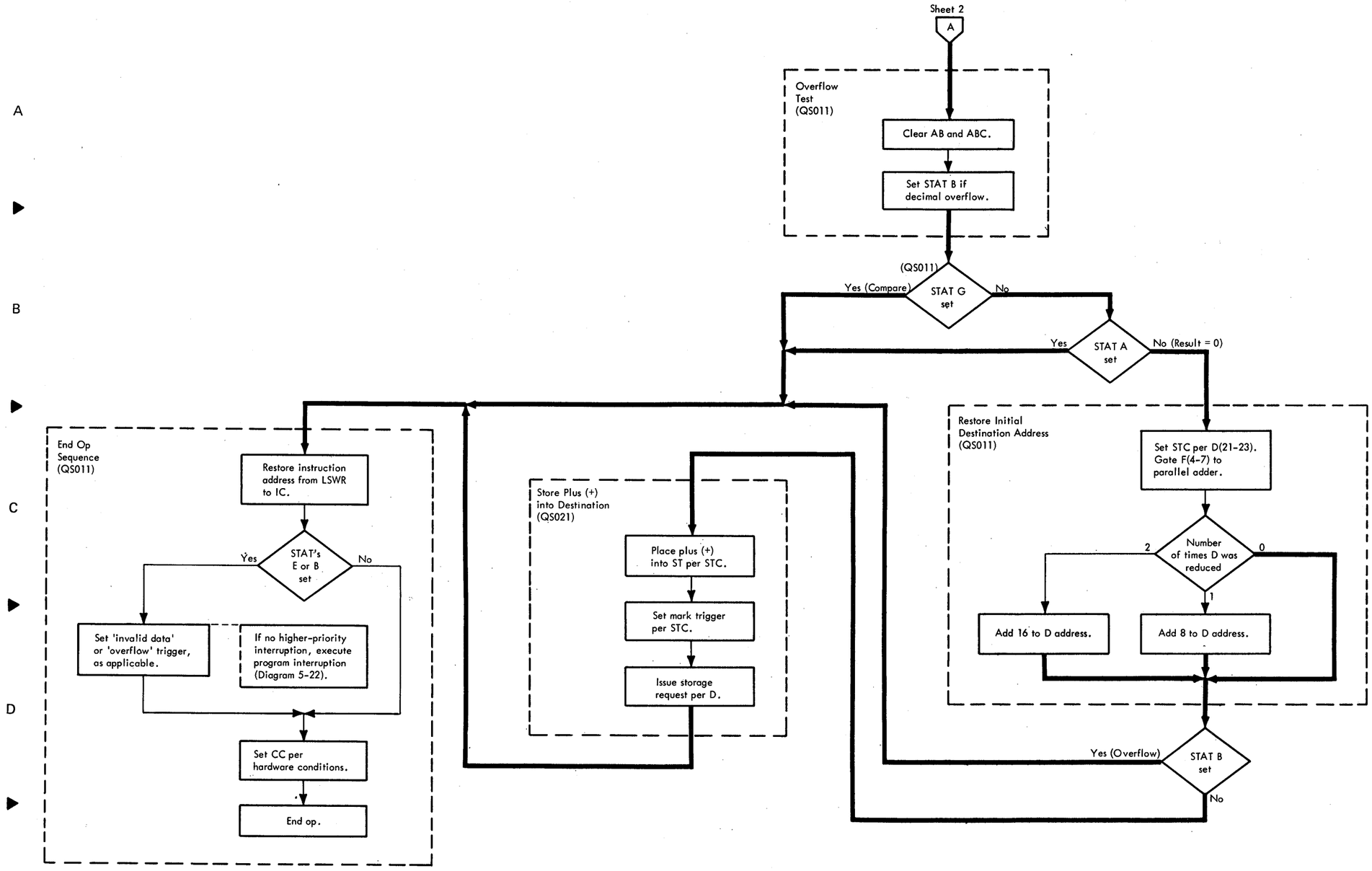
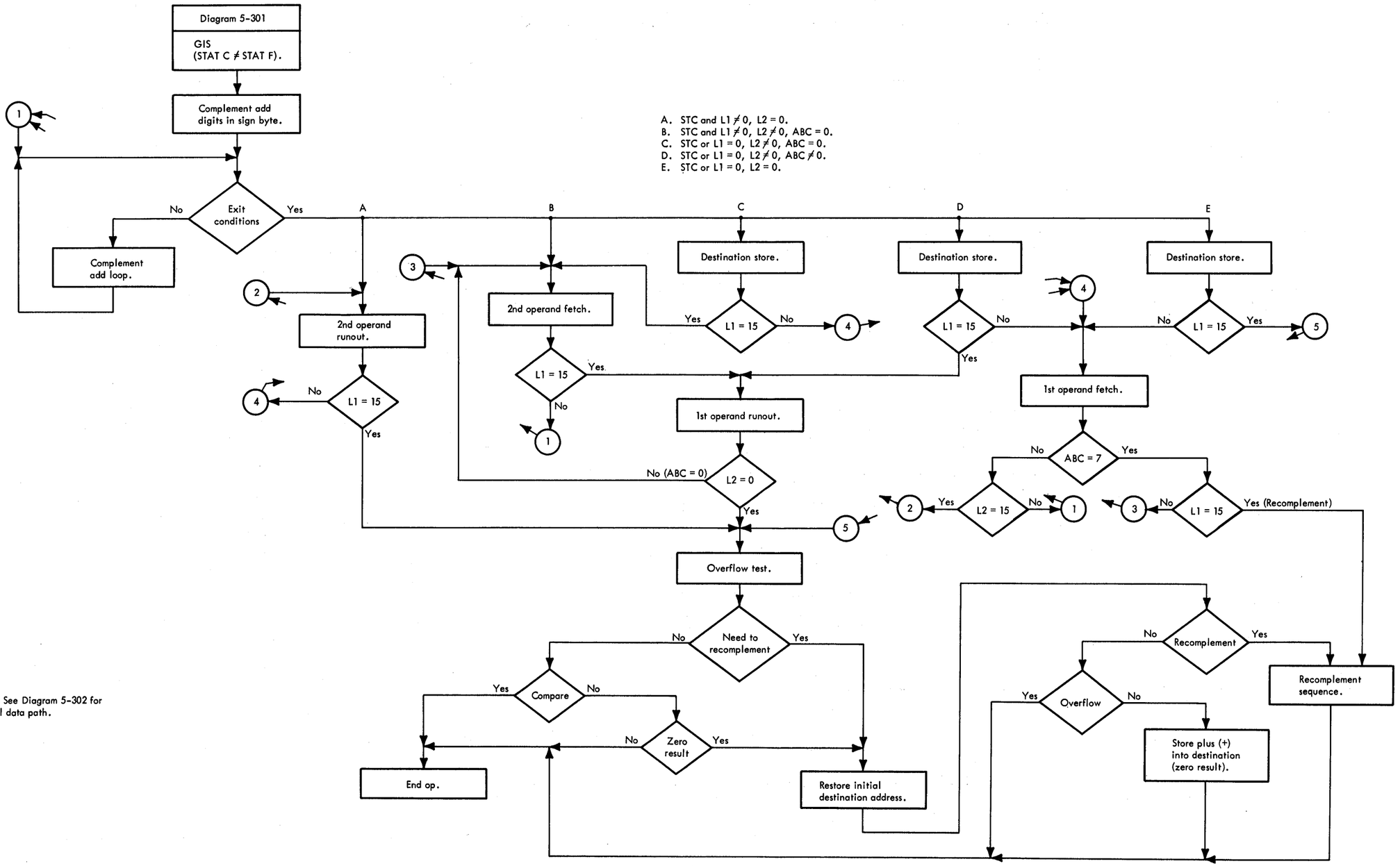


Diagram 5-302. True Add Sequence for Decimal Add, Subtract, and Compare (Sheet 3 of 3)



Note: See Diagram 5-302 for general data path.

Diagram 5-303. Complement Add Sequence for Decimal Add, Subtract, and Compare (Sheet 1 of 3)

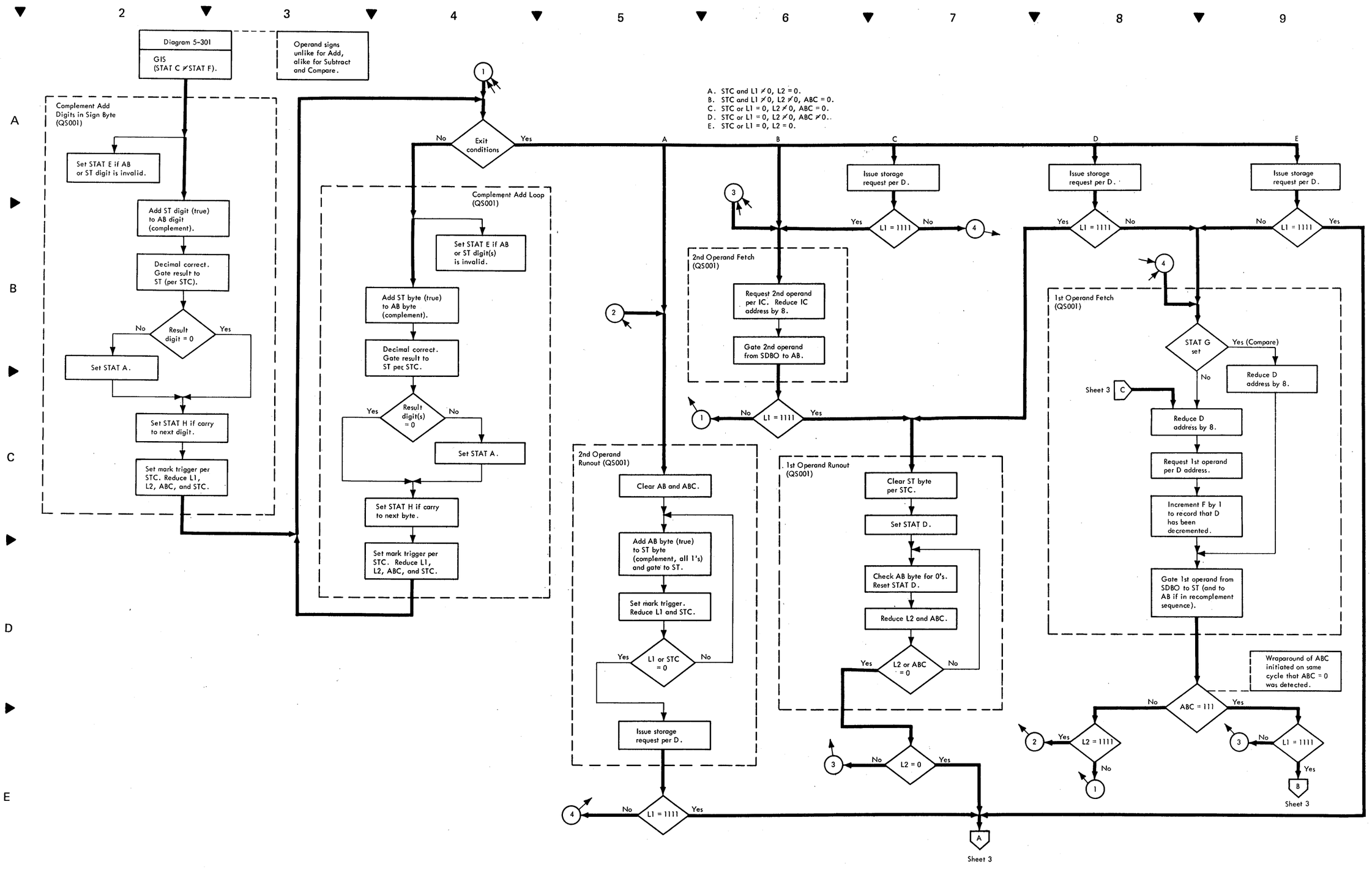


Diagram 5-303. Complement Add Sequence for Decimal Add, Subtract, and Compare (Sheet 2 of 3)

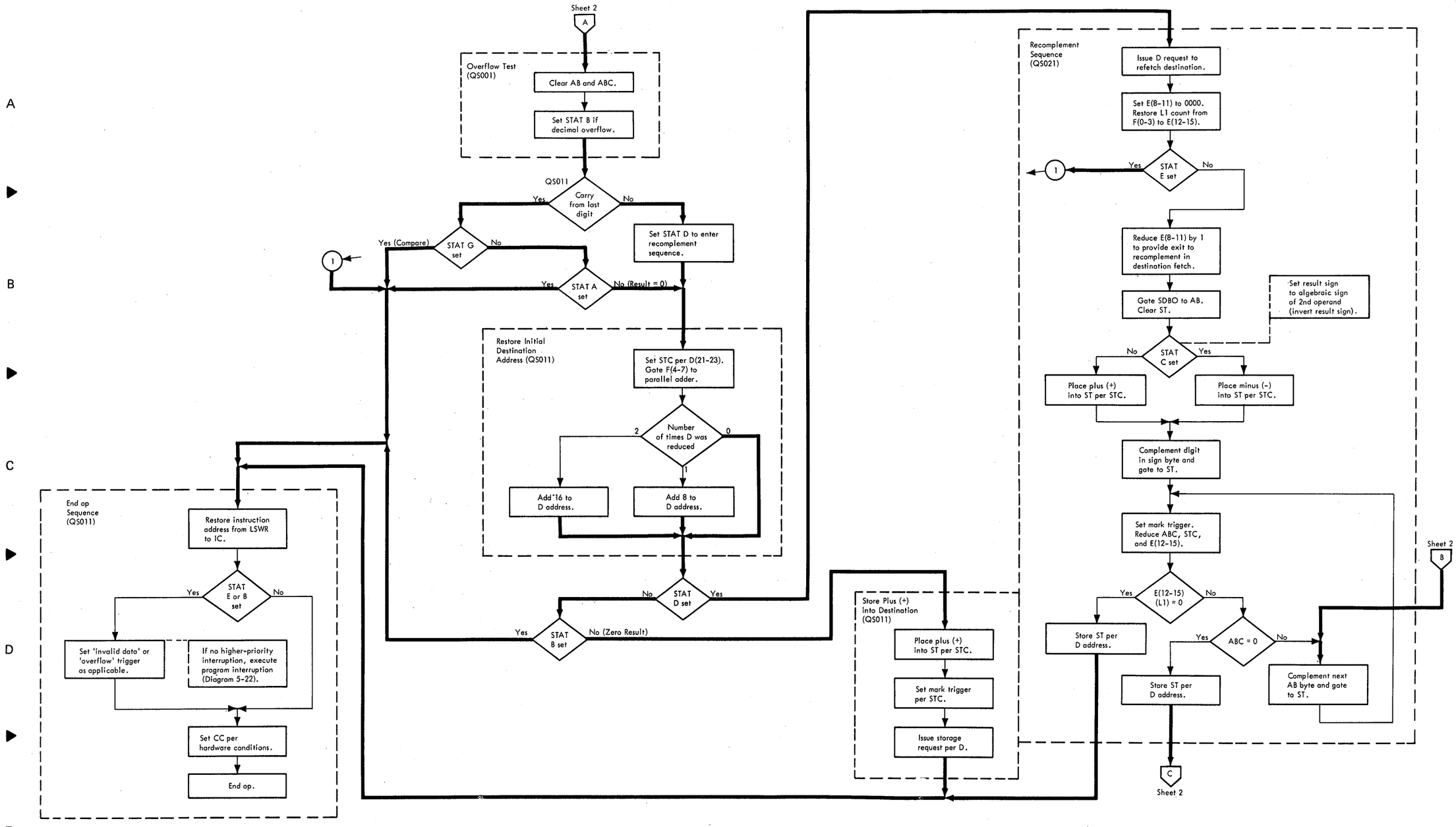
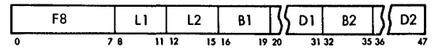


Diagram 5-303. Complement Add Sequence for Decimal Add, Subtract, and Compare (Sheet 3 of 3)

2 3 4 5 6 7 8 9

A B C D E

- SS format:

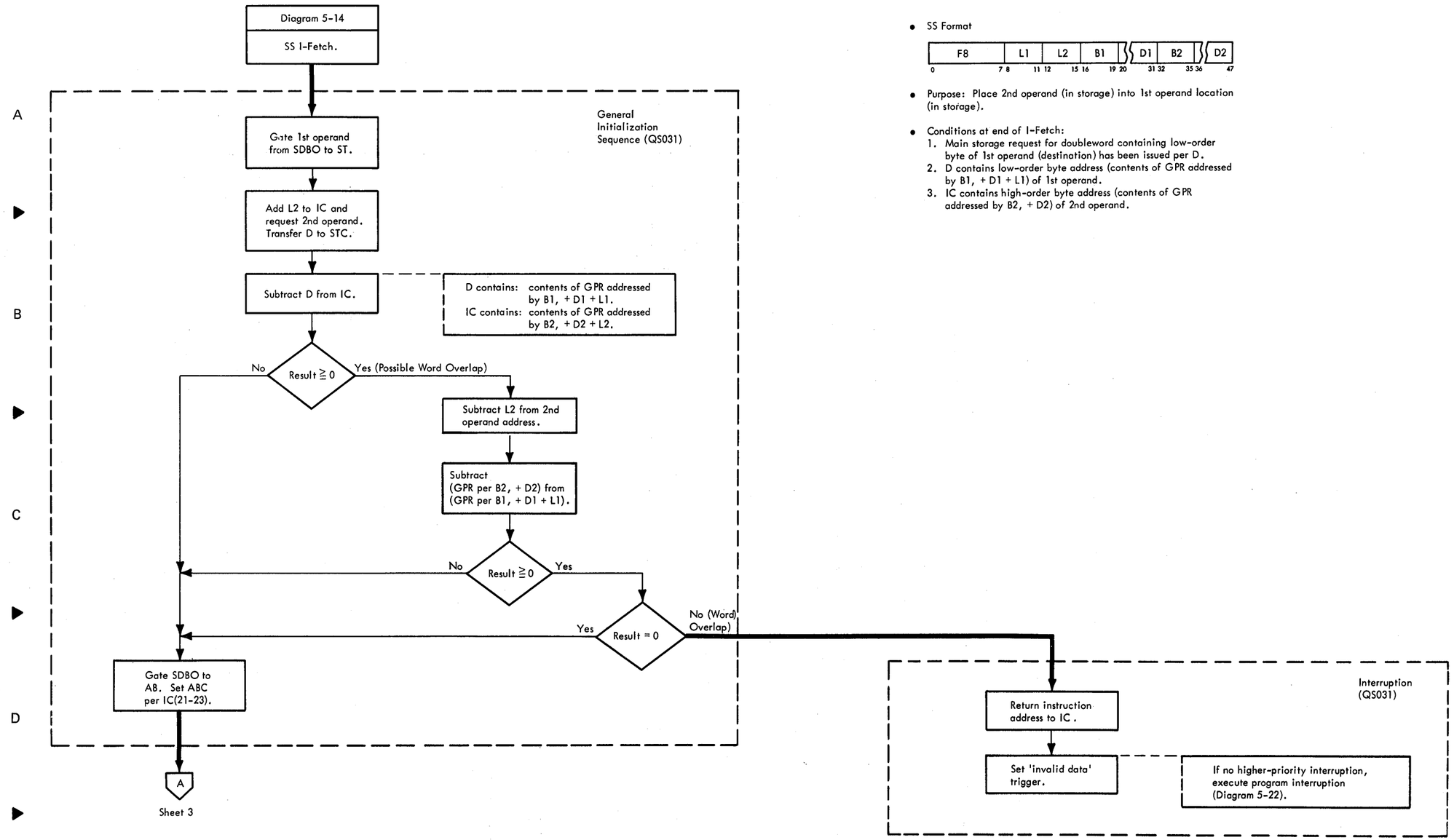


- Purpose: Place 2nd operand (in storage) into 1st operand location (in storage).

A. STC and L1 ≠ 0, L2 = 0.
B. STC and L1 ≠ 0, L2 ≠ 0, ABC = 0.
C. STC or L1 = 0, L2 ≠ 0, ABC = 0.
D. STC or L1 = 0, L2 ≠ 0, ABC ≠ 0.
E. STC or L1 = 0, L2 = 0.

Note: See Diagram 5-302 for general data path.

Diagram 5-304. Zero and Add (Sheet 1 of 4)



- SS Format

- Purpose: Place 2nd operand (in storage) into 1st operand location (in storage).

- Conditions at end of I-Fetch:
  1. Main storage request for doubleword containing low-order byte of 1st operand (destination) has been issued per D.
  2. D contains low-order byte address (contents of GPR addressed by B1, + D1 + L1) of 1st operand.
  3. IC contains high-order byte address (contents of GPR addressed by B2, + D2) of 2nd operand.

Diagram 5-304. Zero and Add (Sheet 2 of 4)

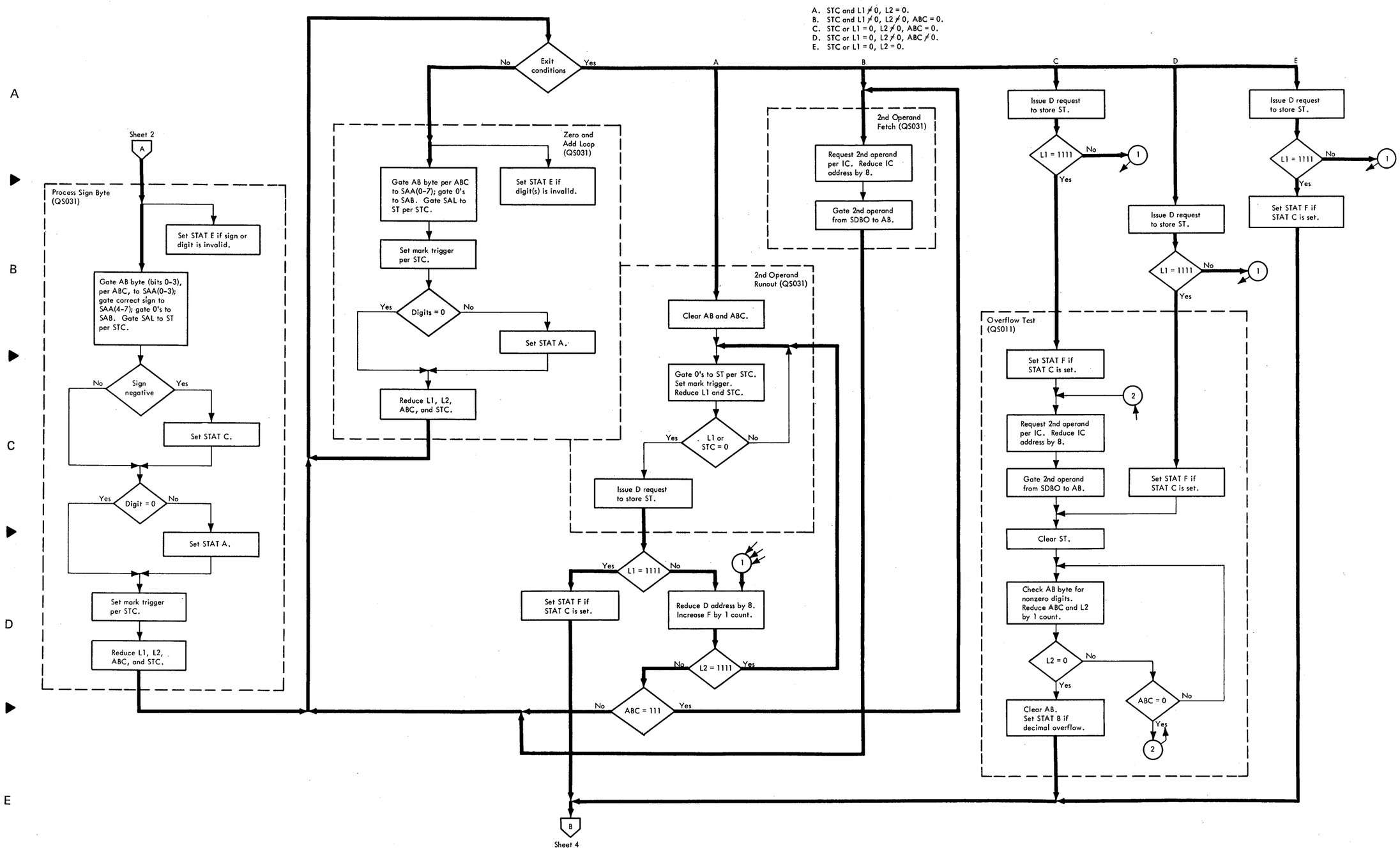


Diagram 5-304. Zero and Add (Sheet 3 of 4)

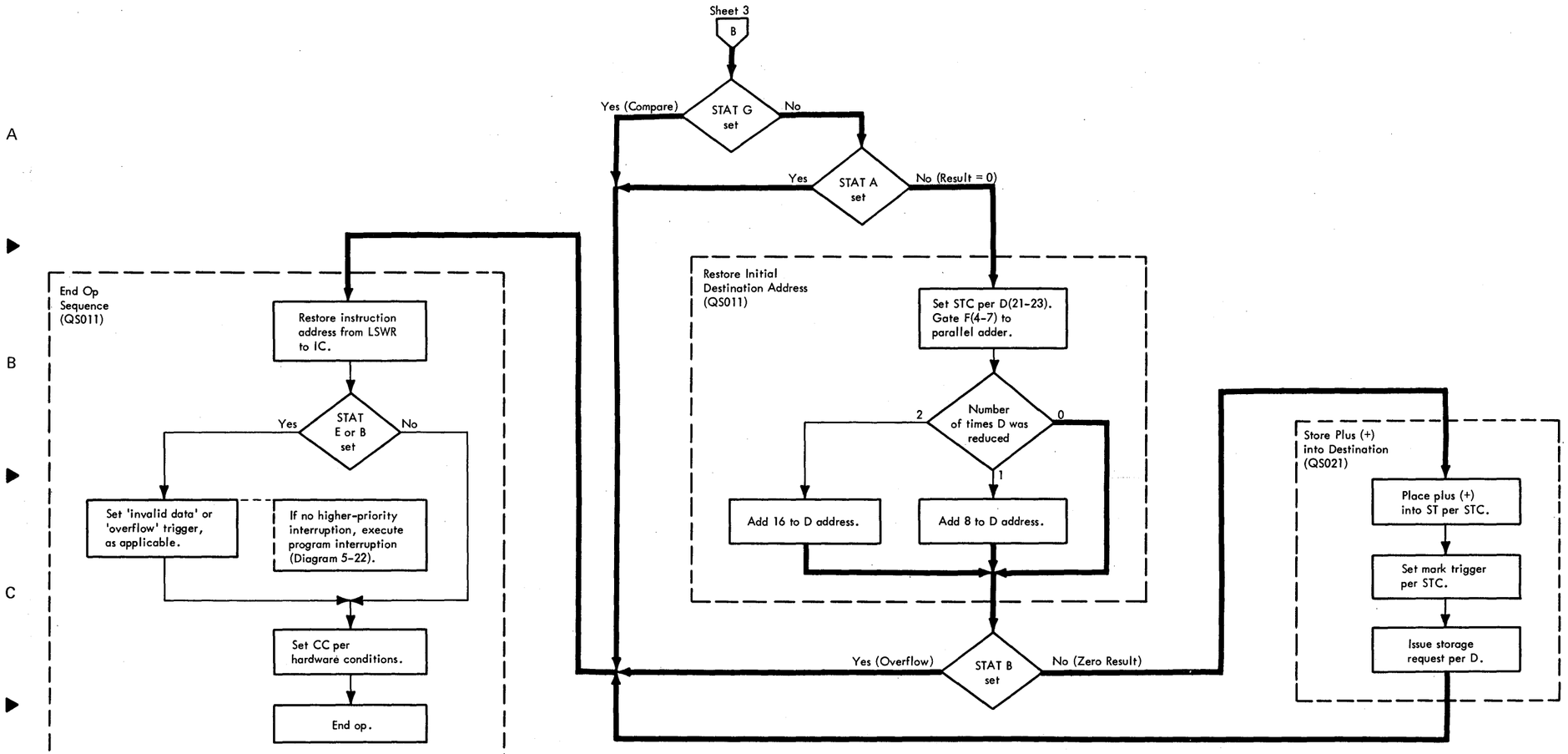
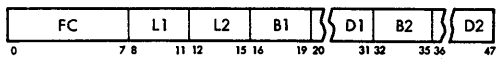


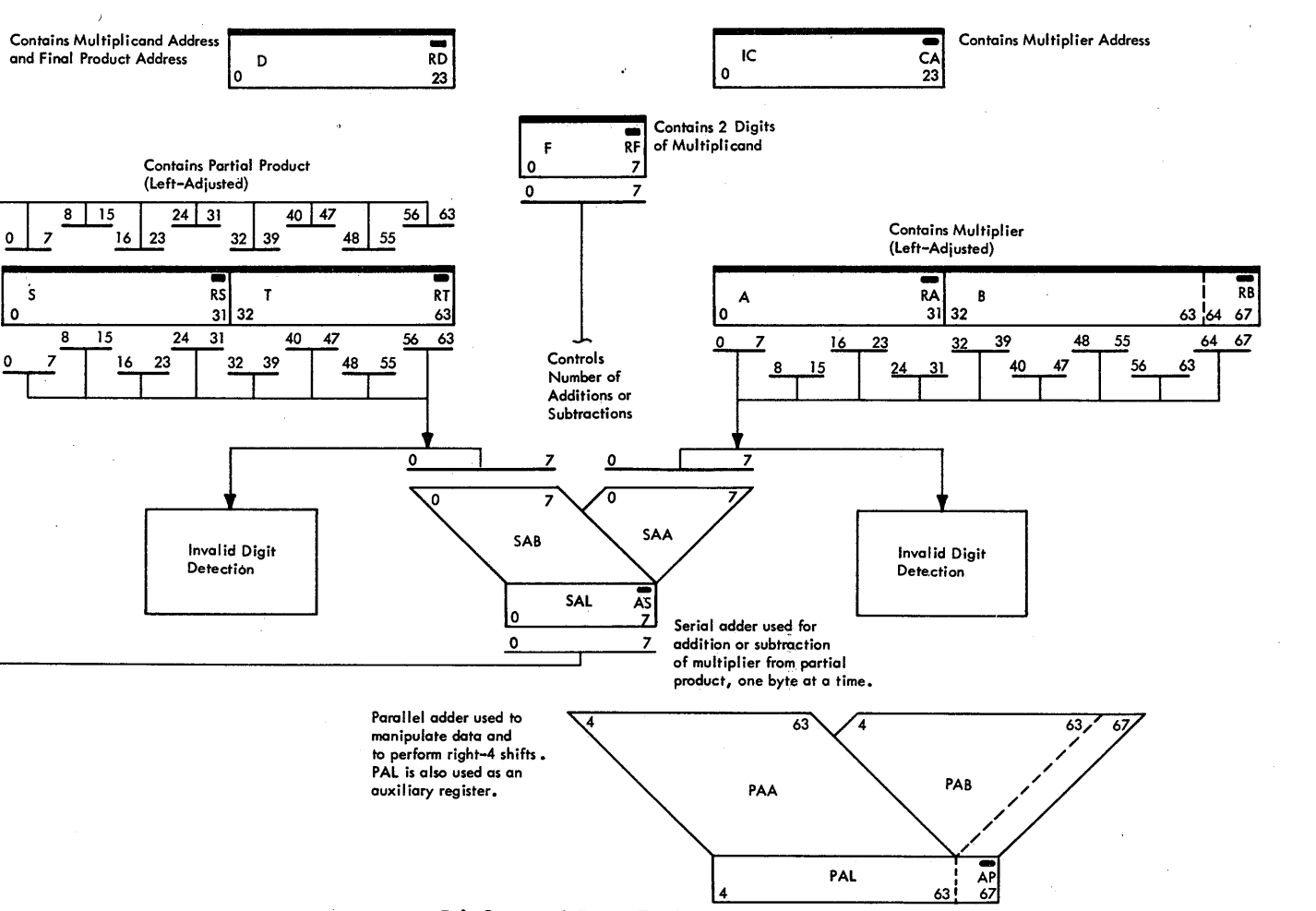
Diagram 5-304. Zero and Add (Sheet 4 of 4)

- SS Format:



- Purpose: Multiply 1st operand (in storage) by 2nd operand (in storage) and place result into 1st operand location.

A) Overall Flow Chart



B) General Data Path

Diagram 5-305. Decimal Multiply (Sheet 1 of 7)

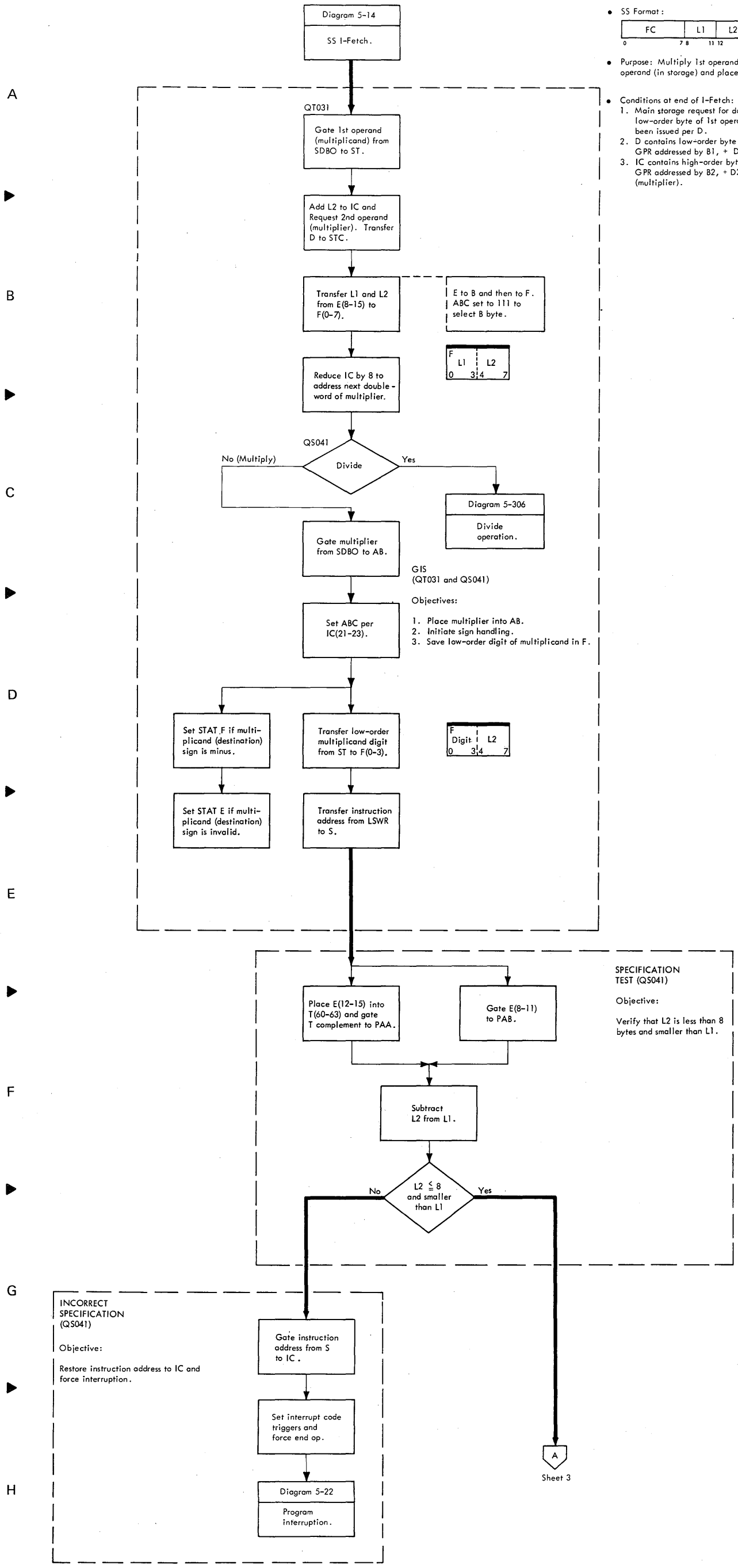


Diagram 5-305. Decimal Multiply (Sheet 2 of 7)

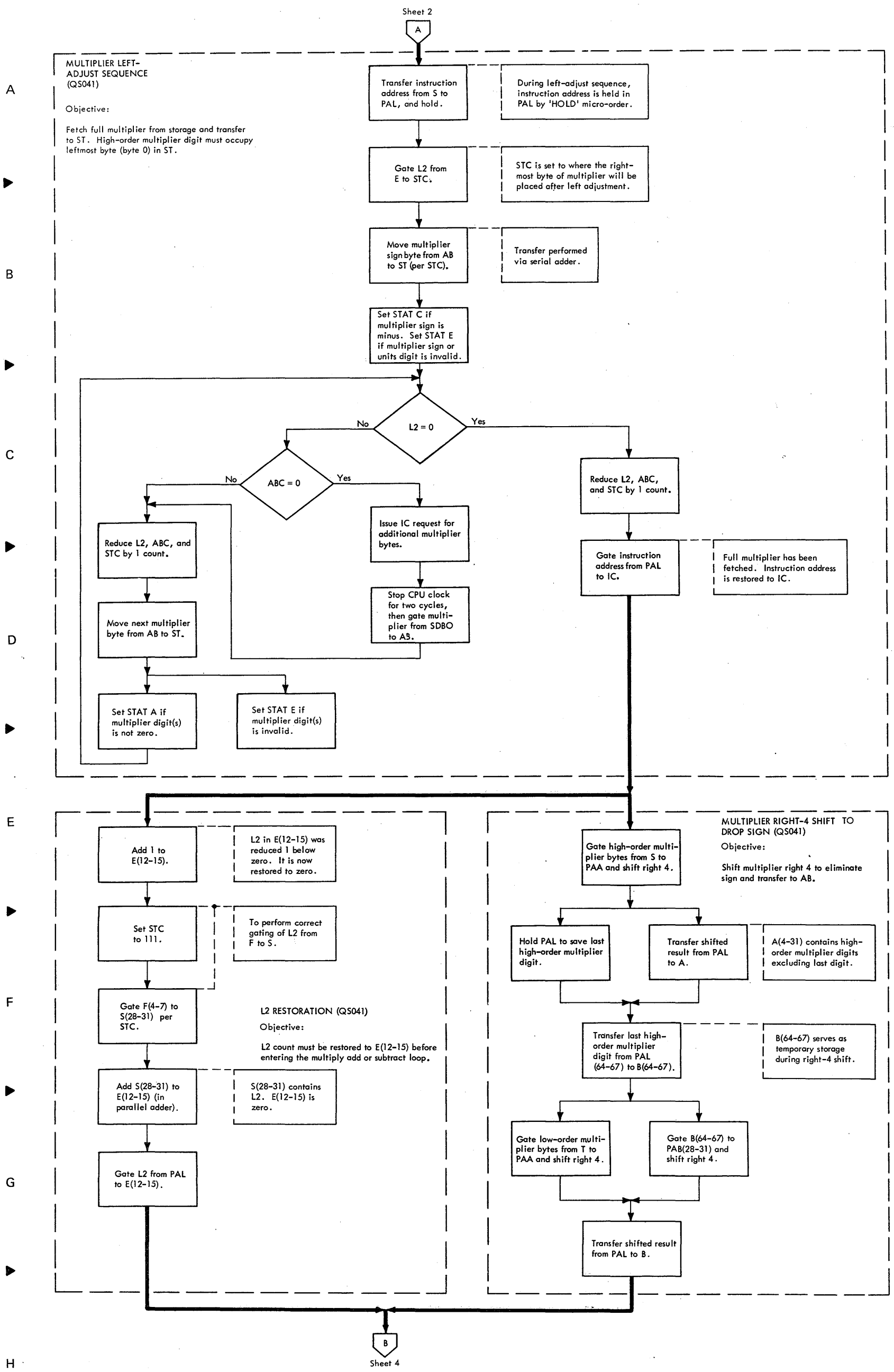


Diagram 5-305. Decimal Multiply (Sheet 3 of 7)

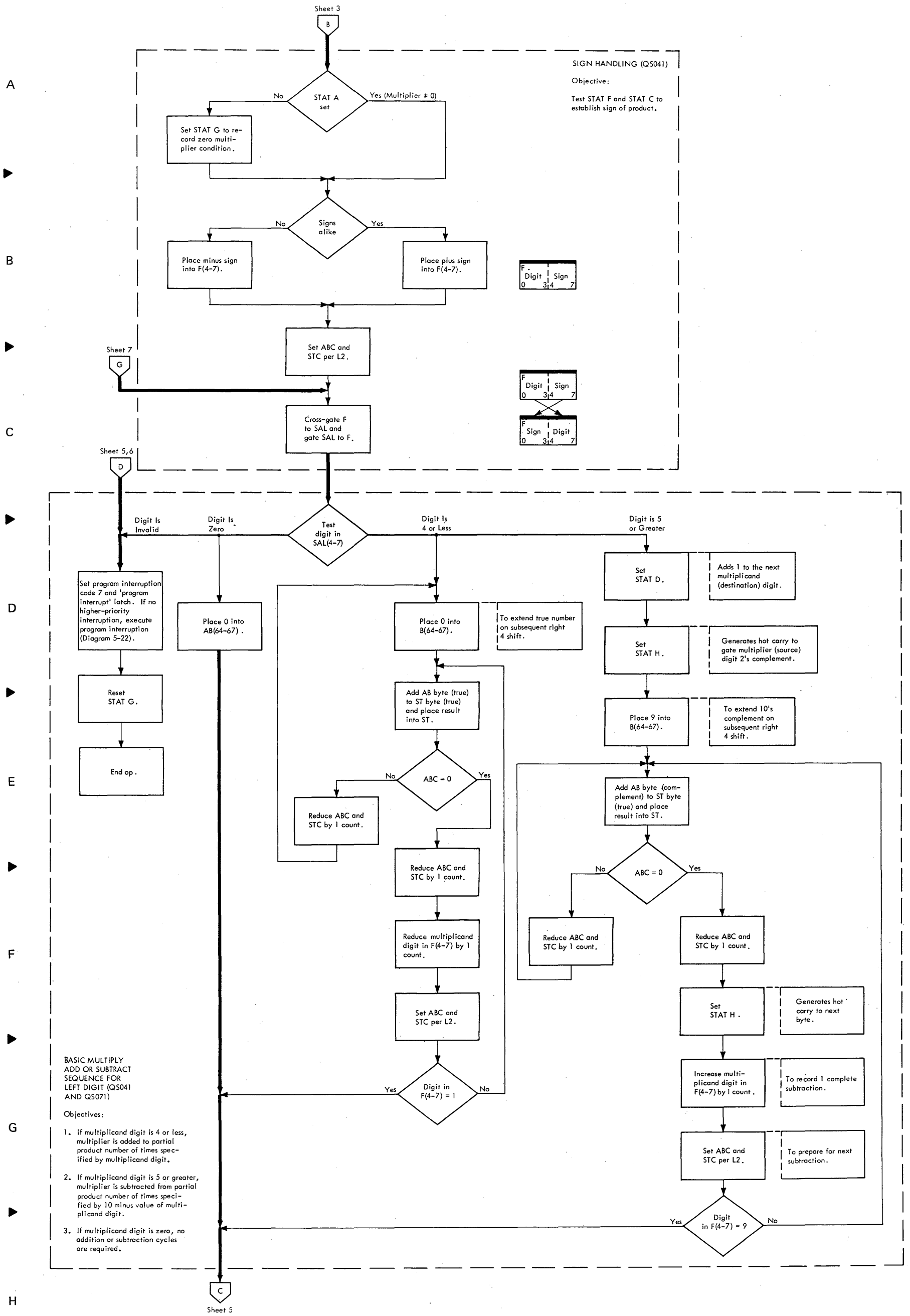


Diagram 5-305. Decimal Multiply (Sheet 4 of 7)

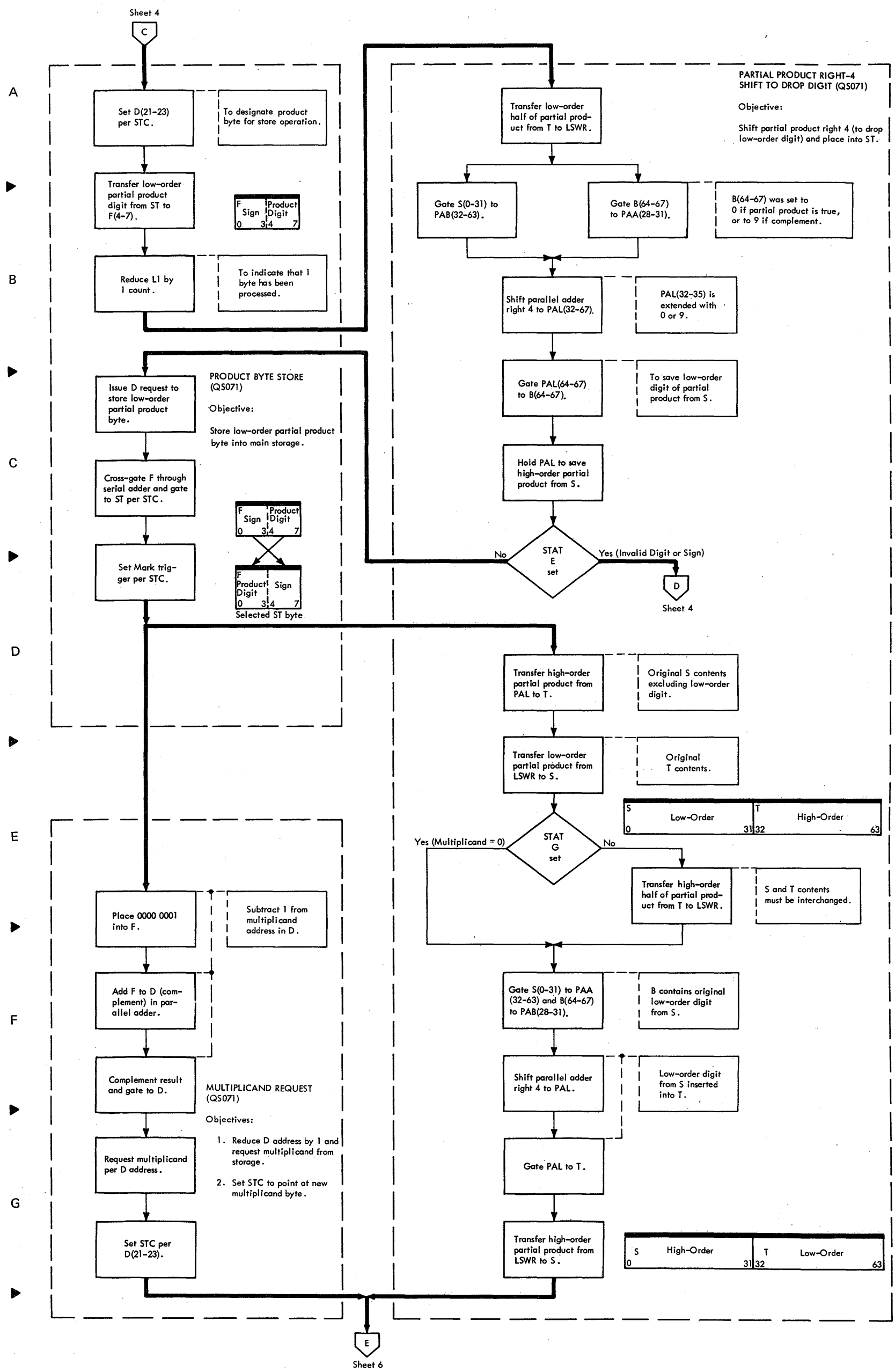


Diagram 5-305. Decimal Multiply (Sheet 5 of 7)

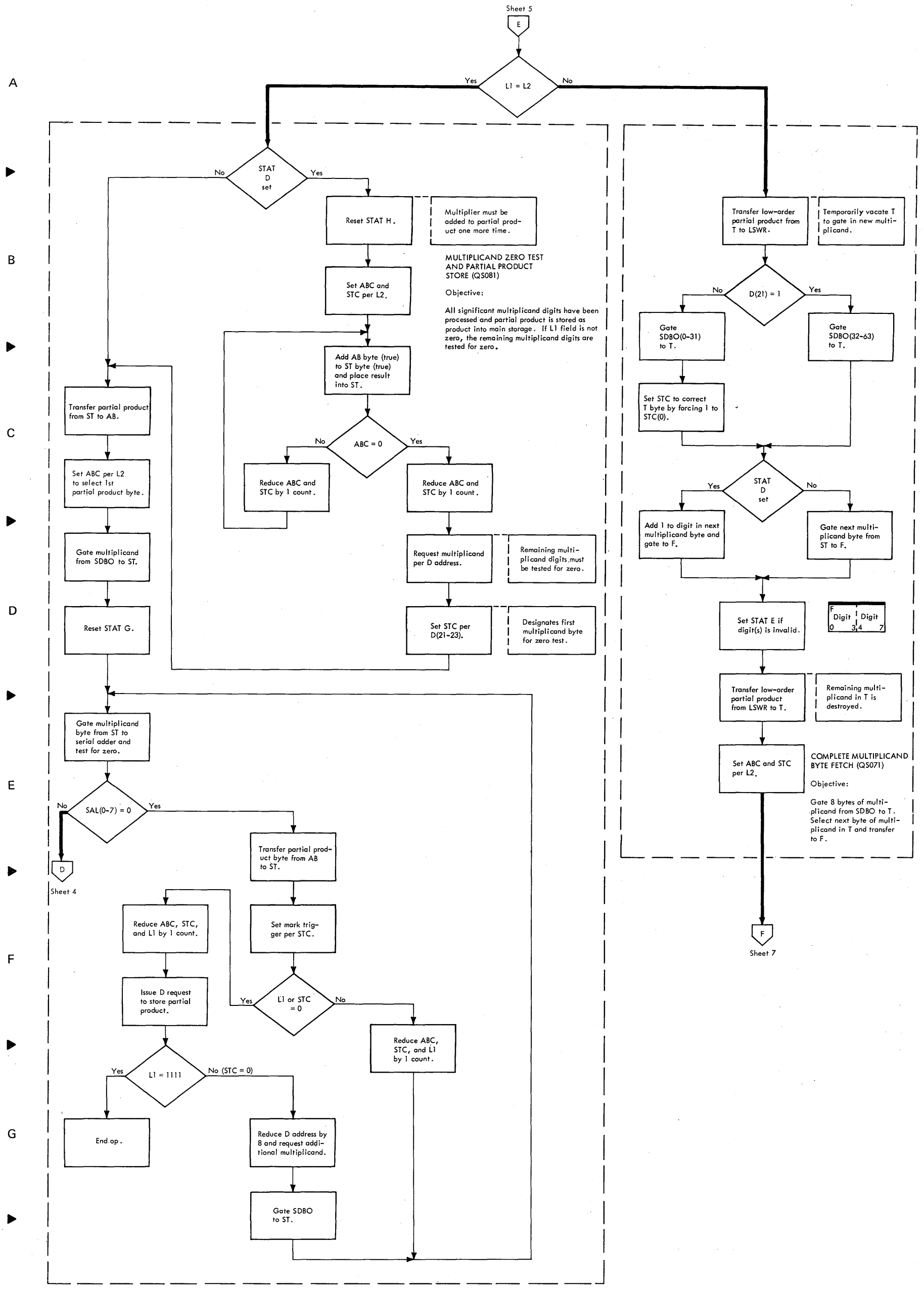


Diagram 5-305. Decimal Multiply (Sheet 6 of 7)

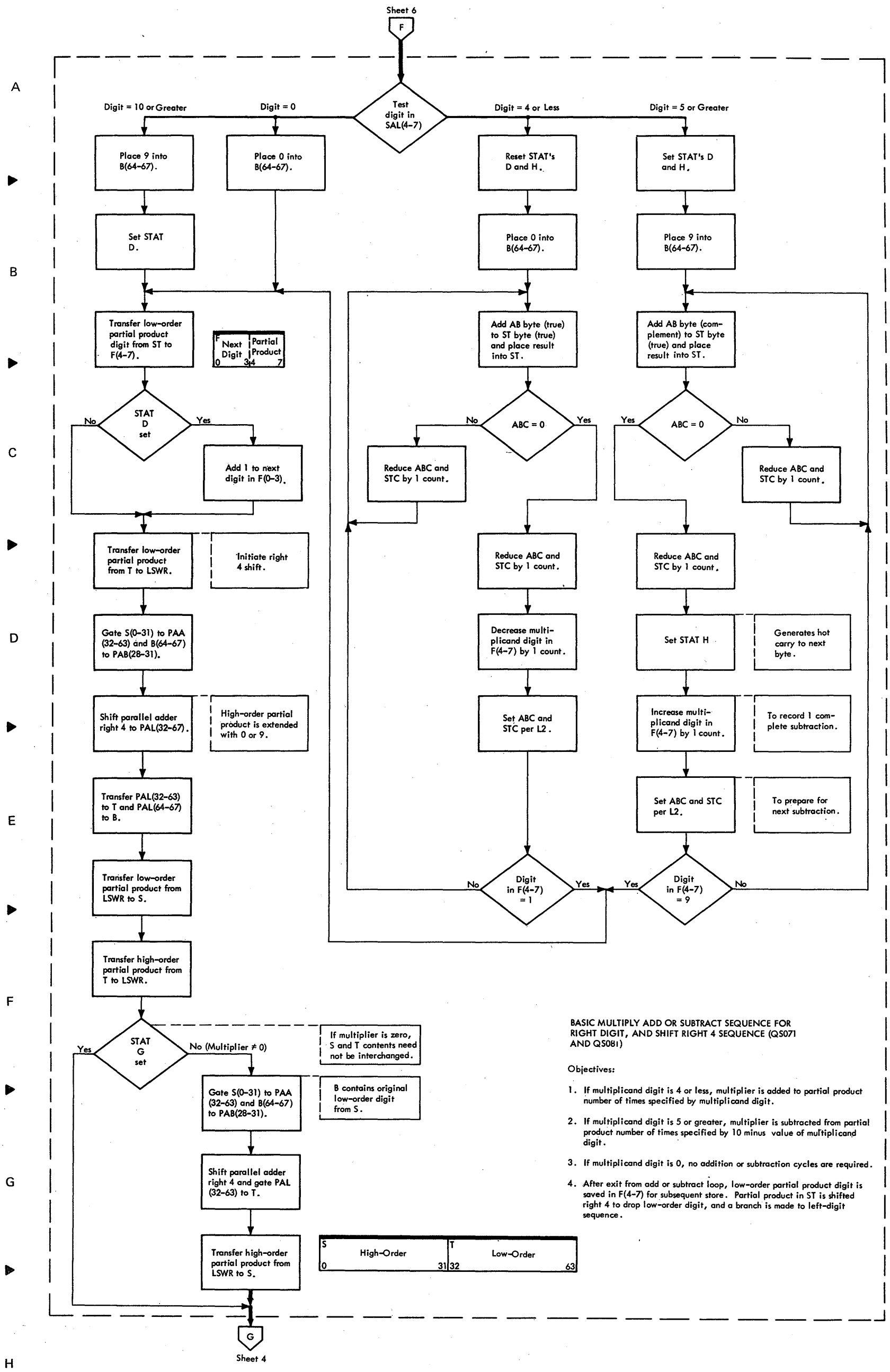


Diagram 5-305. Decimal Multiply (Sheet 7 of 7)

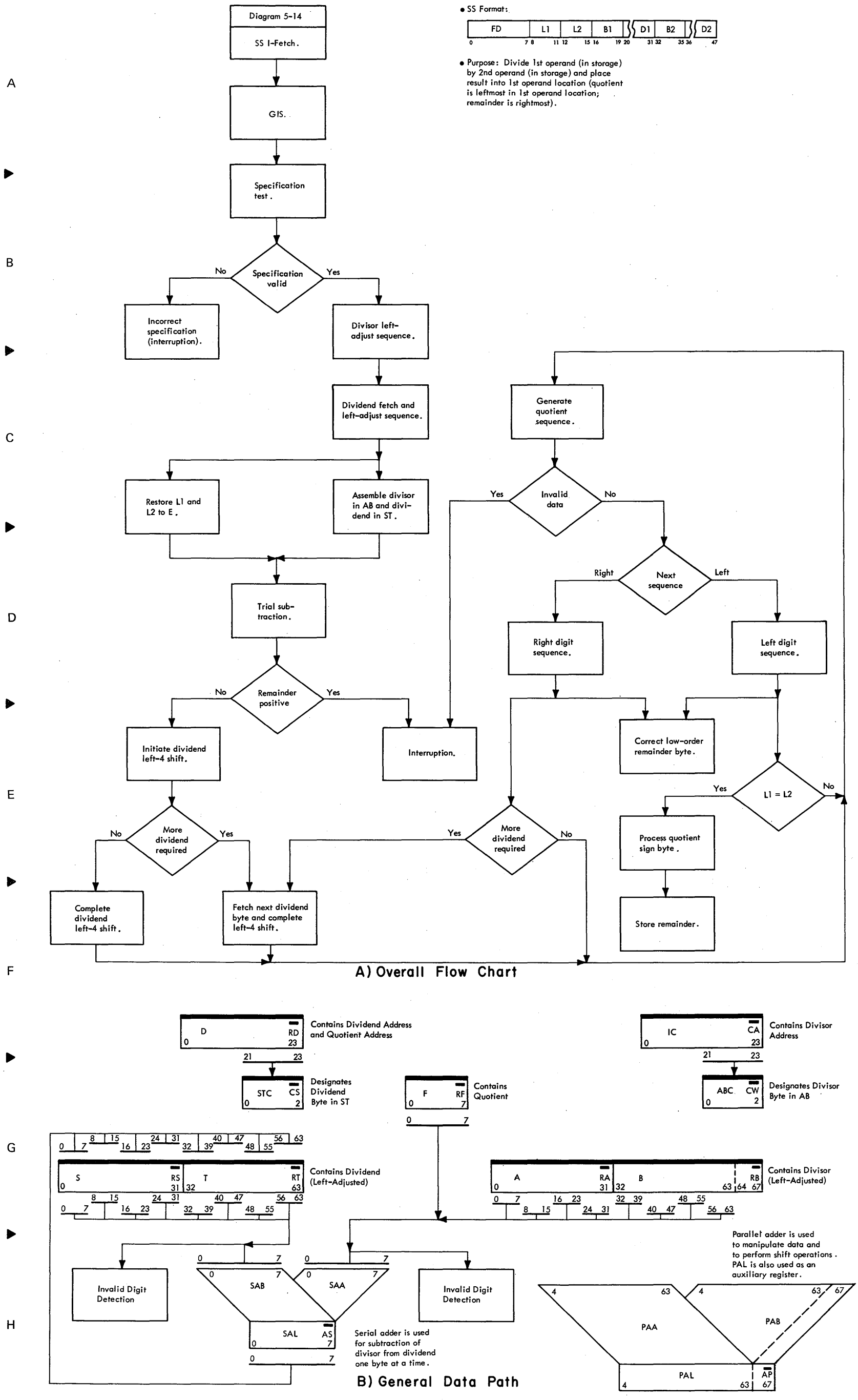
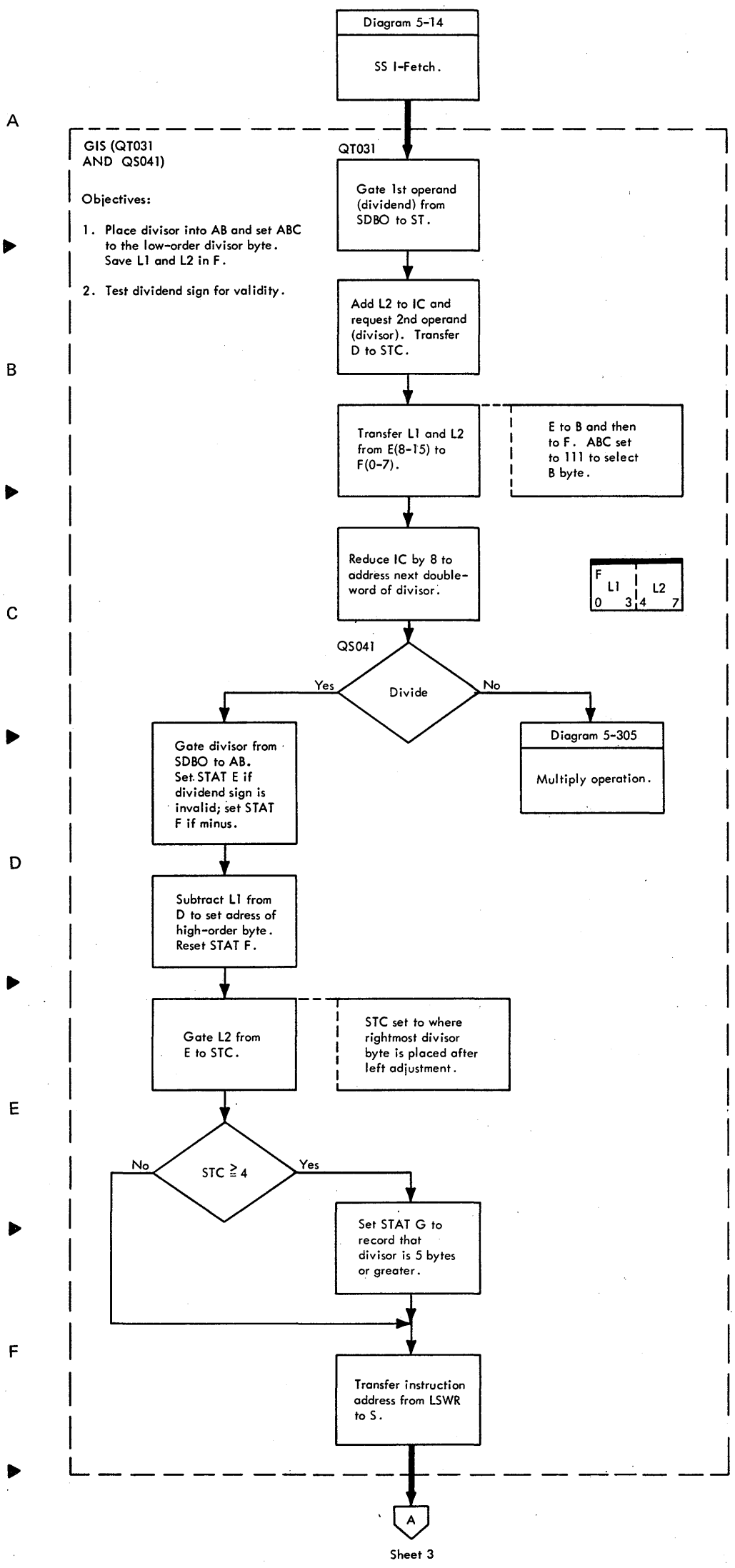
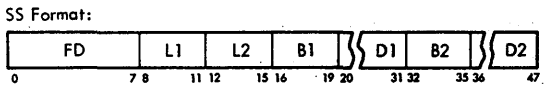


Diagram 5-306. Decimal Divide (Sheet 1 of 9)



- SS Format:



- Purpose: Divide 1st operand (in storage) by 2nd operand (in storage) and place result into 1st operand location (quotient is leftmost in 1st operand location; remainder is rightmost).

- Conditions at end of I-Fetch:
  1. Main storage request for doubleword containing low-order byte of 1st operand (dividend) has been issued per D.
  2. D contains low-order byte address (contents of GPR addressed by B1, + D1 + L1) of 1st operand.
  3. IC contains high-order byte address (contents of GPR addressed by B2, + D2) of 2nd operand (divisor).

Diagram 5-306. Decimal Divide (Sheet 2 of 9)

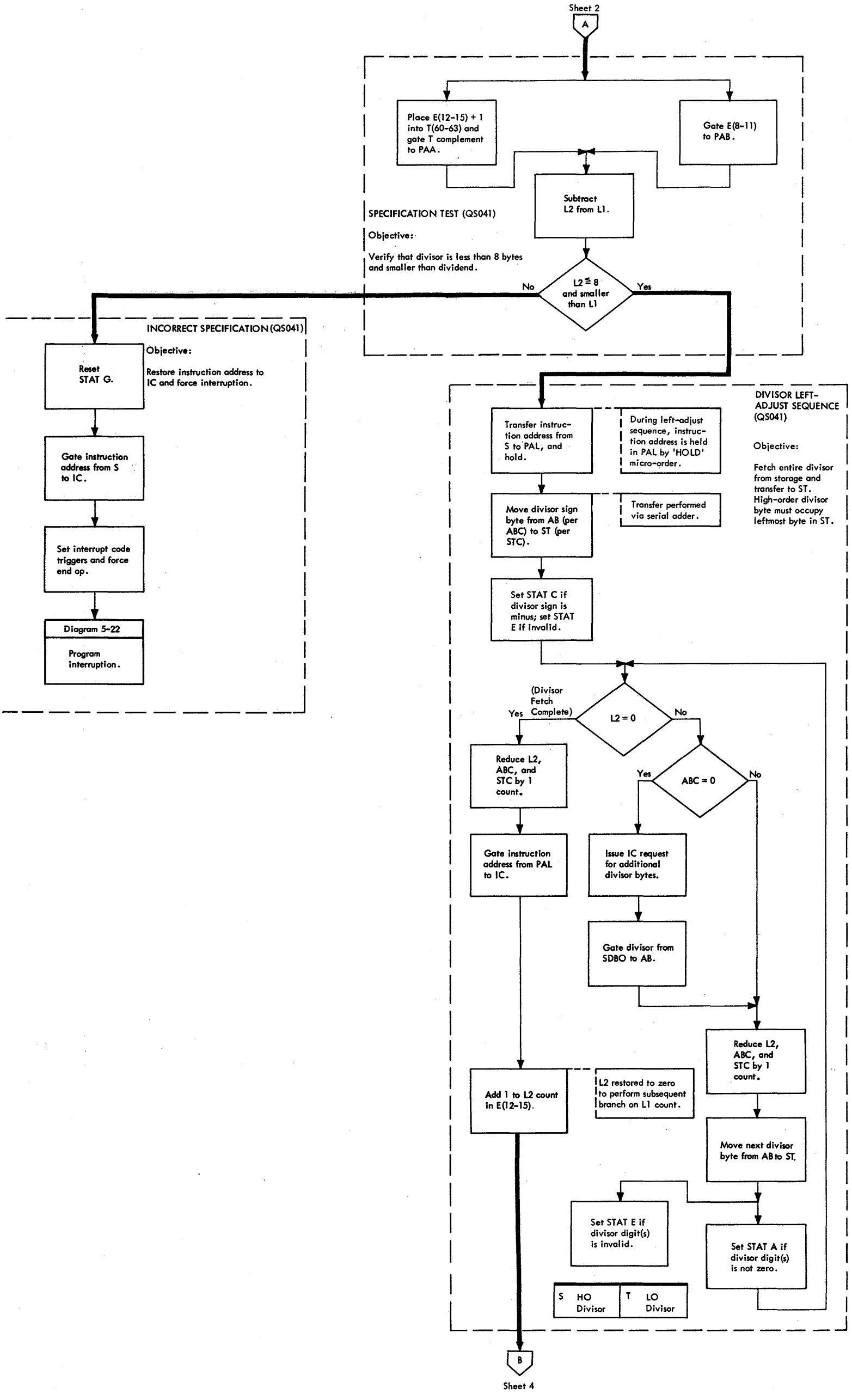


Diagram 5-306. Decimal Divide (Sheet 3 of 9)

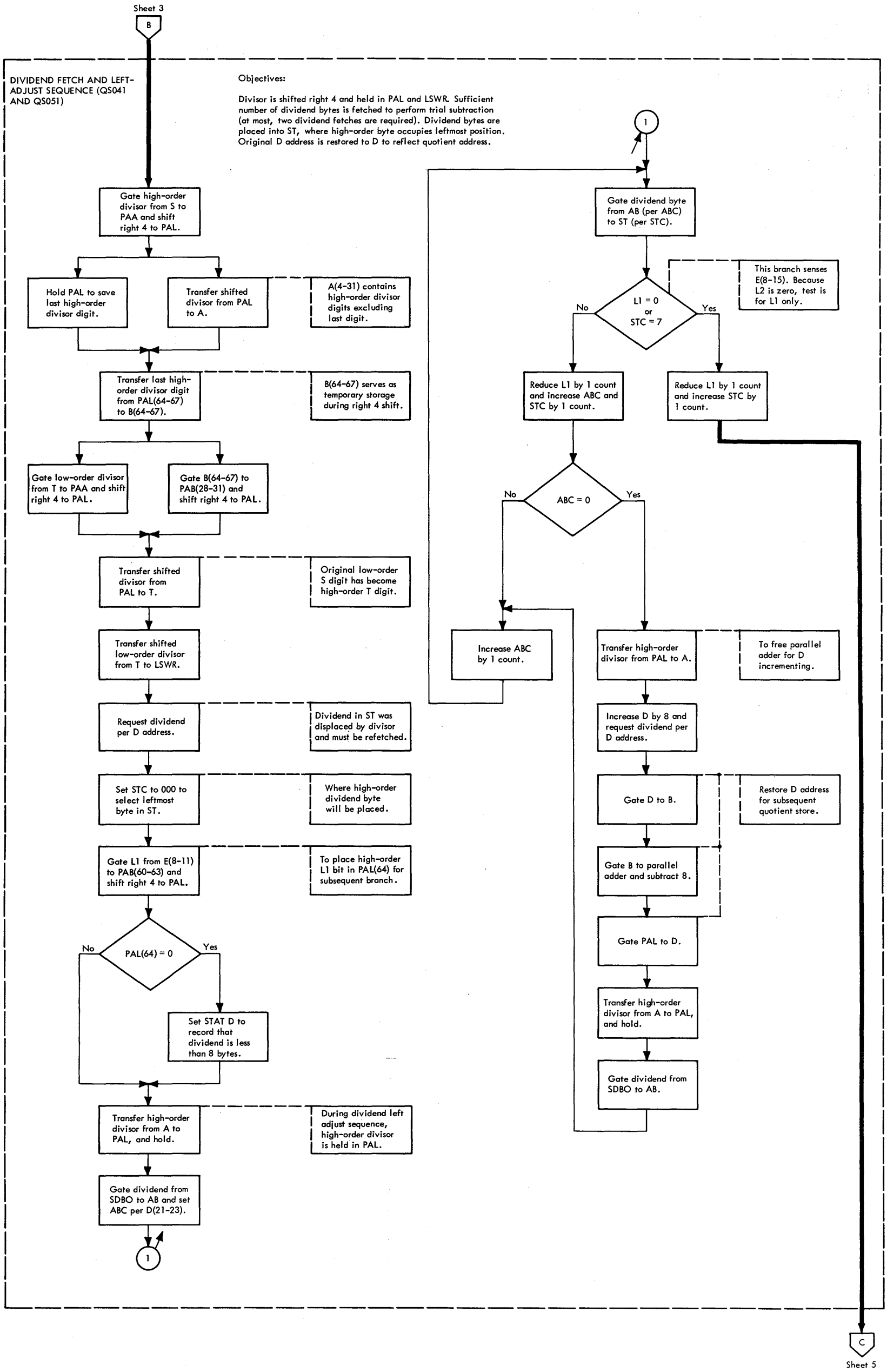


Diagram 5-306. Decimal Divide (Sheet 4 of 9)

RESTORE L1 AND L2 TO E (QS051)

Objective:

Left adjustment of 1st and 2nd operands has destroyed original L1 and L2 counts in E. These counts are restored to E.

Transfer L2 count from F(4-7) to B(64-67).

F(4-7) to PAB(60-63). Shift right 4 to PAL. Gate PAL(64-67) to B.

F | L1 (0-3) | L2 (4-7)

B | L2 (63-64-67)

Cross-gate F through serial adder and gate to F.

F | L1 (0-3) | L2 (4-7)

F | L2 (0-3) | L1 (4-7)

Transfer L1 count from F(4-7) to E(12-15).

F(4-7) to PAB (60-63). Gate PAL(56-63) to E(8-15).

E | 0000 (7-8-11) | L1 (12-15)

STAT G set

No

Yes (Divisor is 5 Bytes or Greater)

Gate L1 from E(12-15) to PAB(60-63).

Gate L2 from B(64-67) to PAB(64-67).

Shift left 4 to PAL and gate PAL(56-63) to E(8-15).

E | L1 (7-8-11) | L2 (12-15)

D

Sheet 4

C

ASSEMBLE DIVISOR IN AB AND DIVIDEND IN ST (QS051)

Objectives:

High-order divisor is transferred from PAL to A. If divisor exceeds 4 bytes (STAT G set), low-order divisor is in LSWR and must be transferred to B.

Transfer high-order divisor from PAL to A.

Reset STAT G.

Transfer low-order dividend from T to B.

A HO Divisor | B LO Dividend

Transfer high-order dividend from S to T.

S | T HO Dividend

Transfer low-order divisor from LSWR to S.

S LO Divisor | T HO Dividend

Transfer high-order dividend from T to LSWR.

S LO Divisor | T

Transfer low-order dividend from B to T.

S LO Divisor | T LO Dividend

Transfer low-order divisor from S to B.

A HO Divisor | B LO Divisor

Transfer high-order dividend from LSWR to S.

S HO Dividend | T LO Dividend

Sheet 6

Diagram 5-306. Decimal Divide (Sheet 5 of 9)

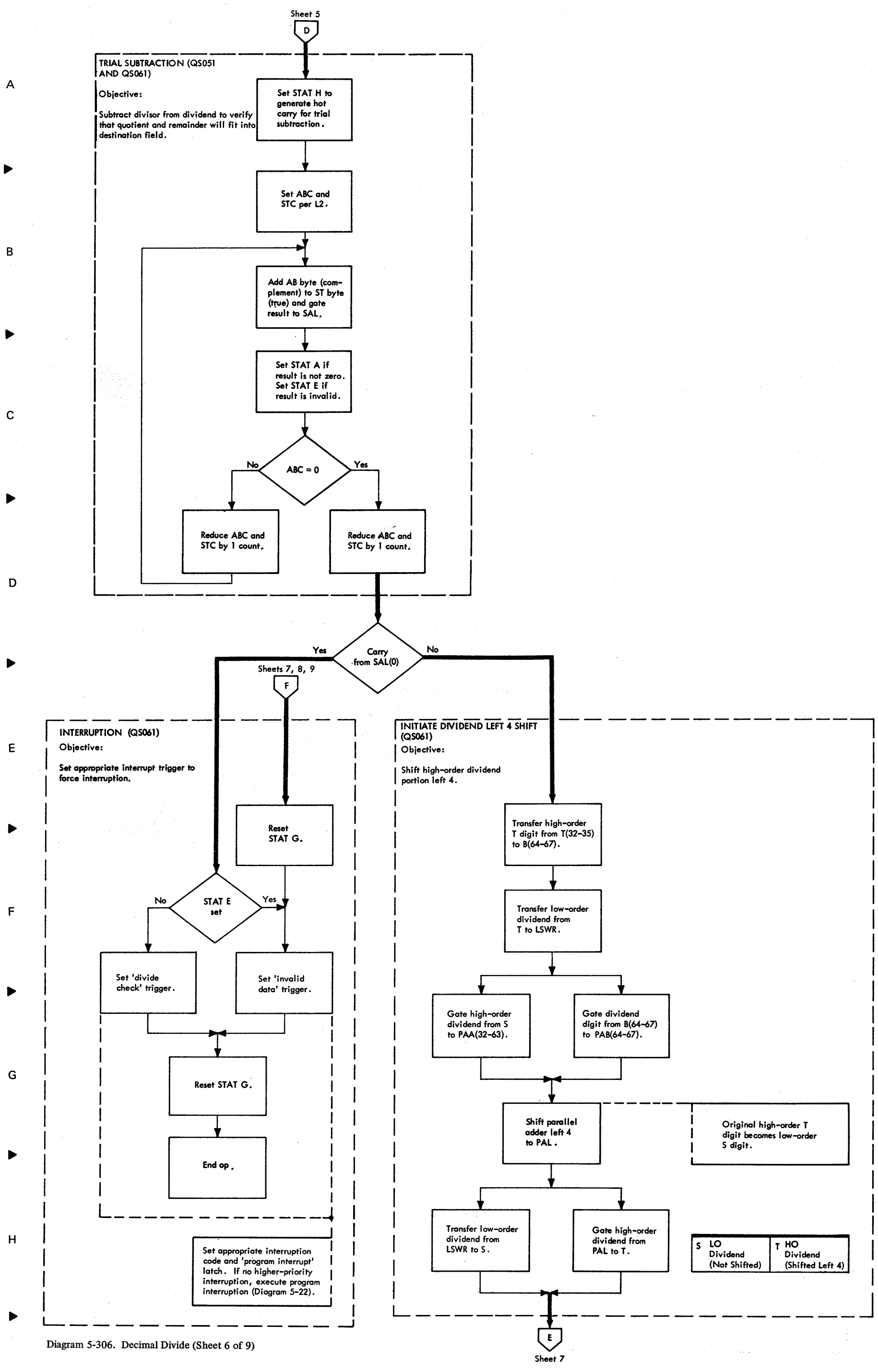


Diagram 5-306. Decimal Divide (Sheet 6 of 9)

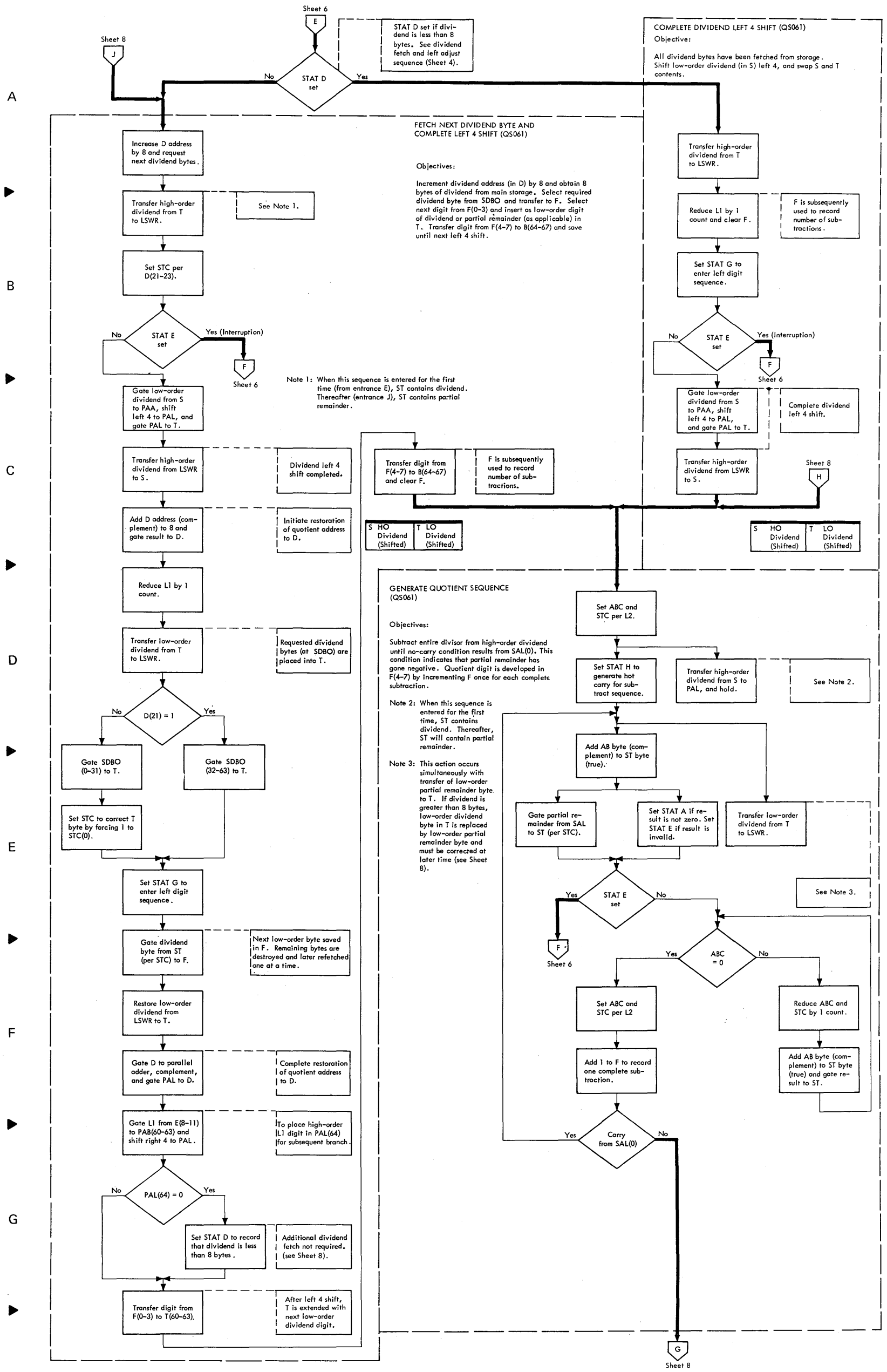


Diagram 5-306. Decimal Divide (Sheet 7 of 9)

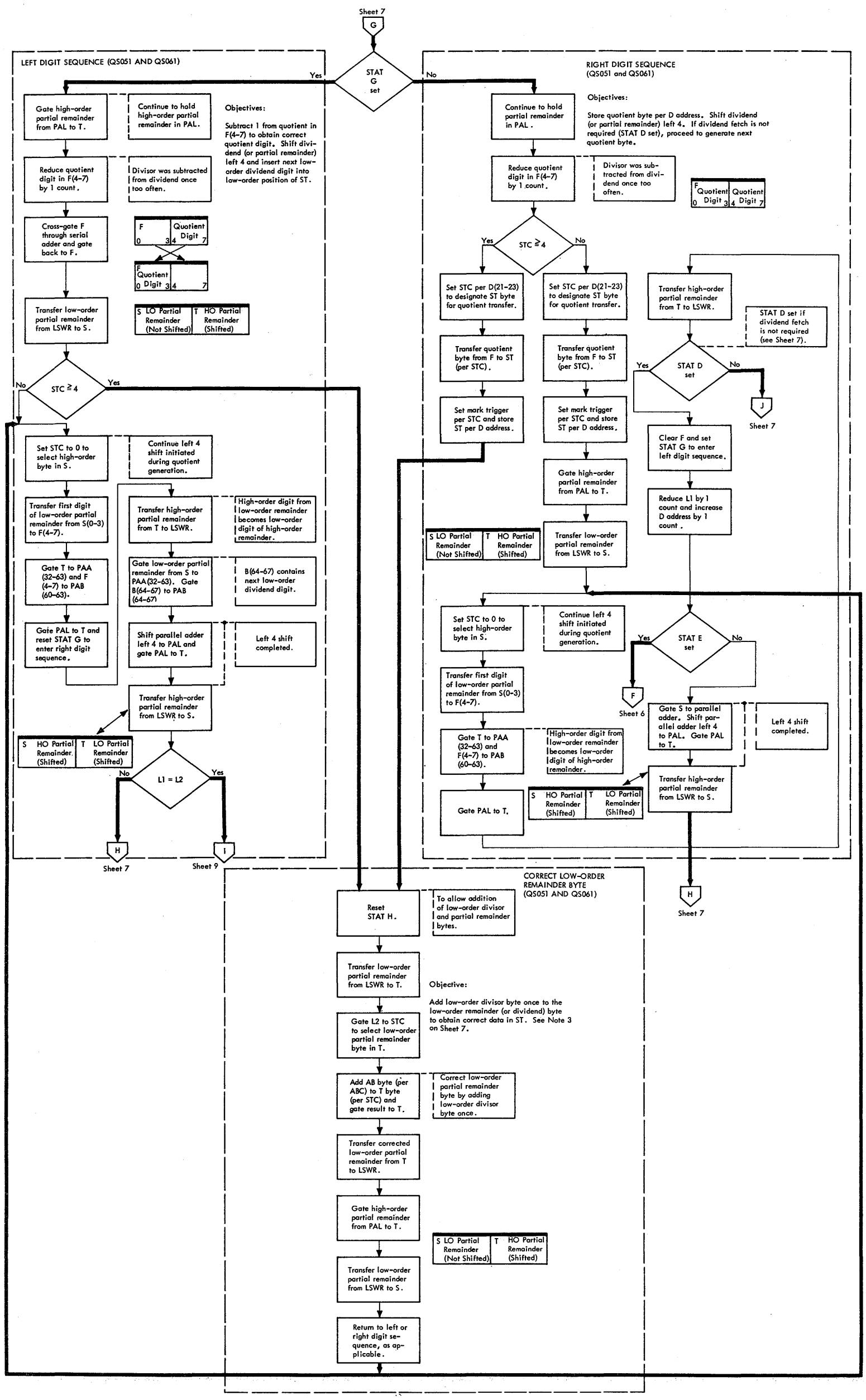


Diagram 5-306. Decimal Divide (Sheet 8 of 9)

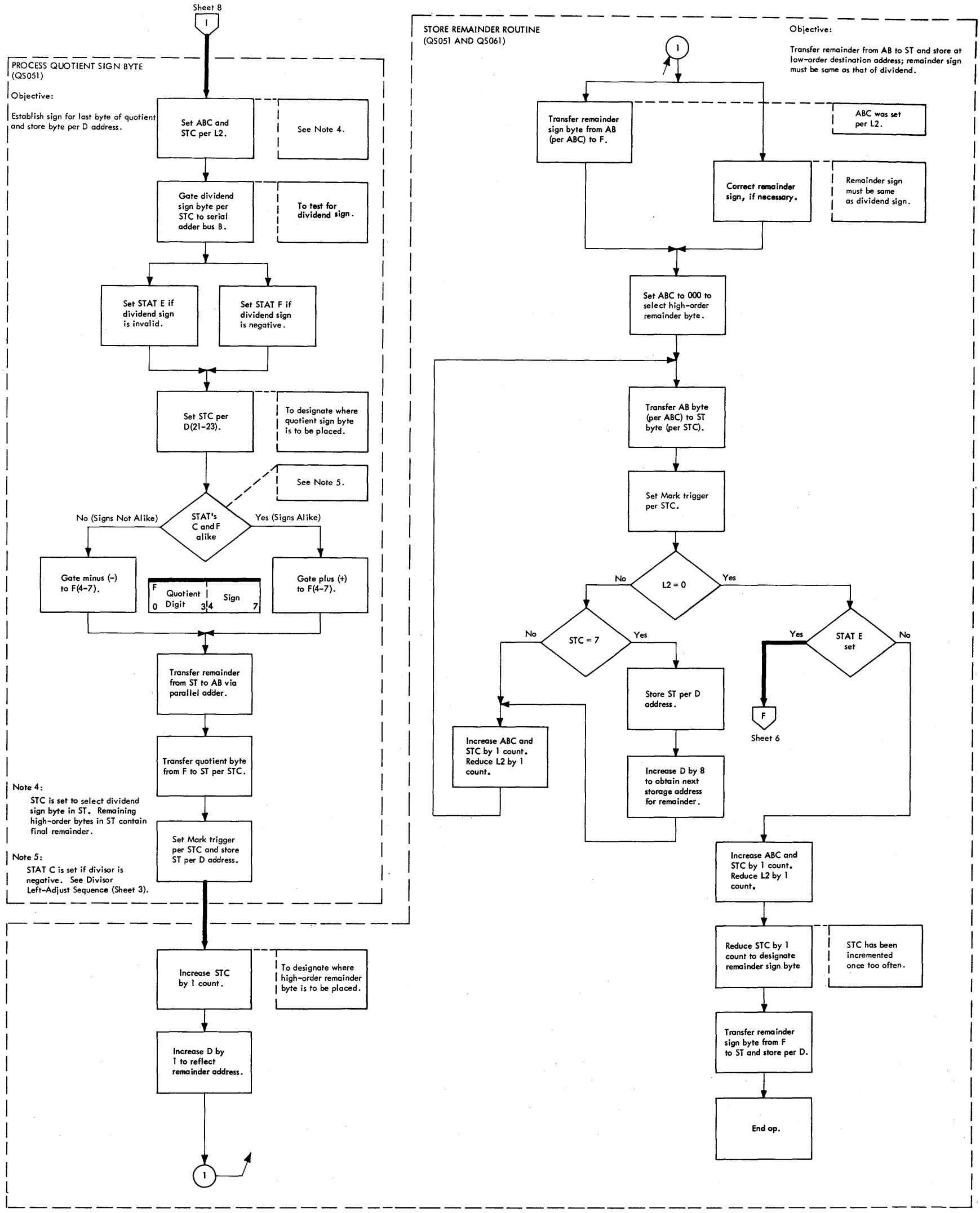
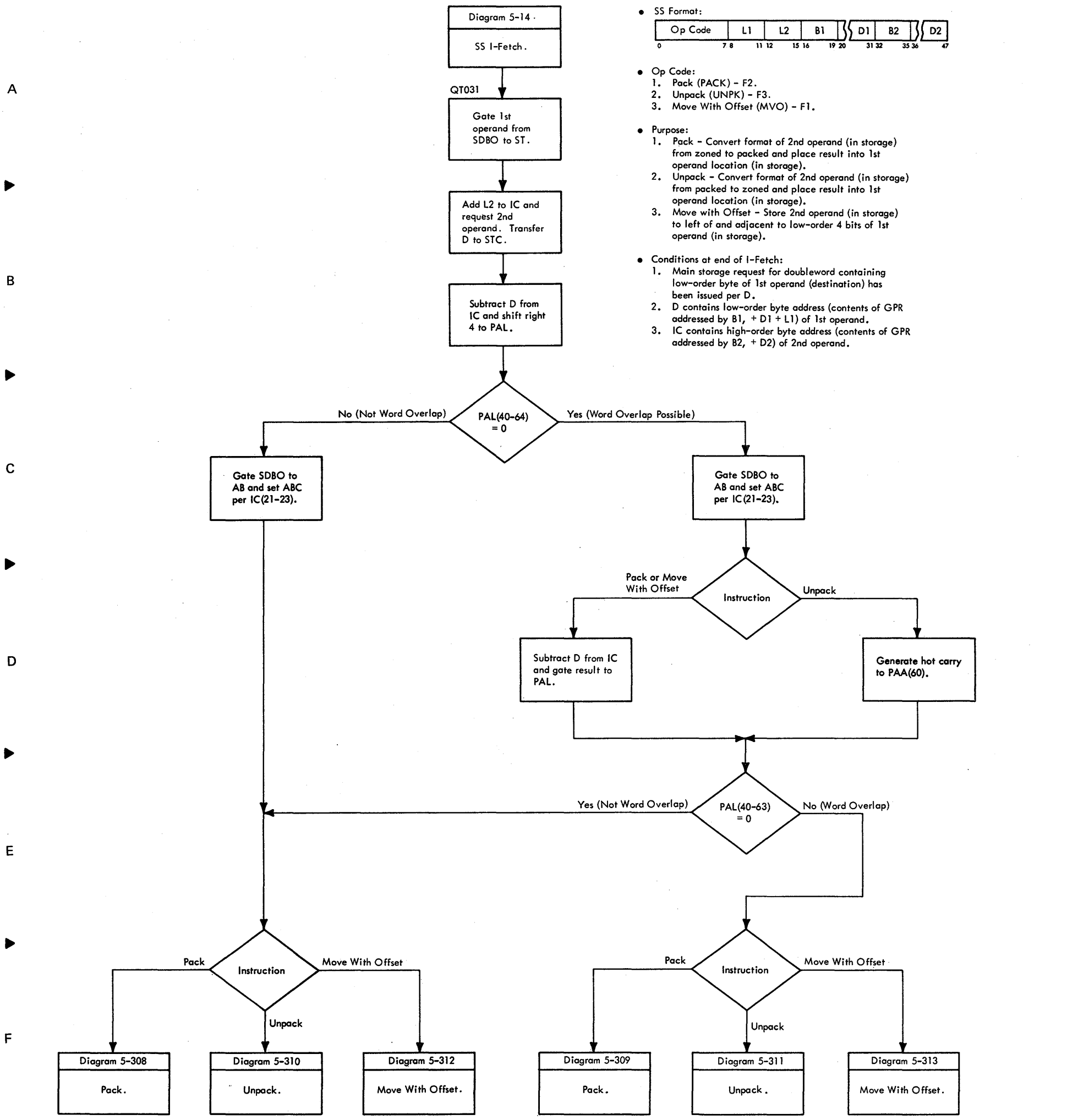


Diagram 5-306. Decimal Divide (Sheet 9 of 9)



- SS Format:

- Op Code:
  1. Pack (PACK) - F2.
  2. Unpack (UNPK) - F3.
  3. Move With Offset (MVO) - F1.

- Purpose:
  1. Pack - Convert format of 2nd operand (in storage) from zoned to packed and place result into 1st operand location (in storage).
  2. Unpack - Convert format of 2nd operand (in storage) from packed to zoned and place result into 1st operand location (in storage).
  3. Move with Offset - Store 2nd operand (in storage) to left of and adjacent to low-order 4 bits of 1st operand (in storage).

- Conditions at end of I-Fetch:
  1. Main storage request for doubleword containing low-order byte of 1st operand (destination) has been issued per D.
  2. D contains low-order byte address (contents of GPR addressed by B1, + D1 + L1) of 1st operand.
  3. IC contains high-order byte address (contents of GPR addressed by B2, + D2) of 2nd operand.

Diagram 5-307. GIS for Pack, Unpack, and Move With Offset

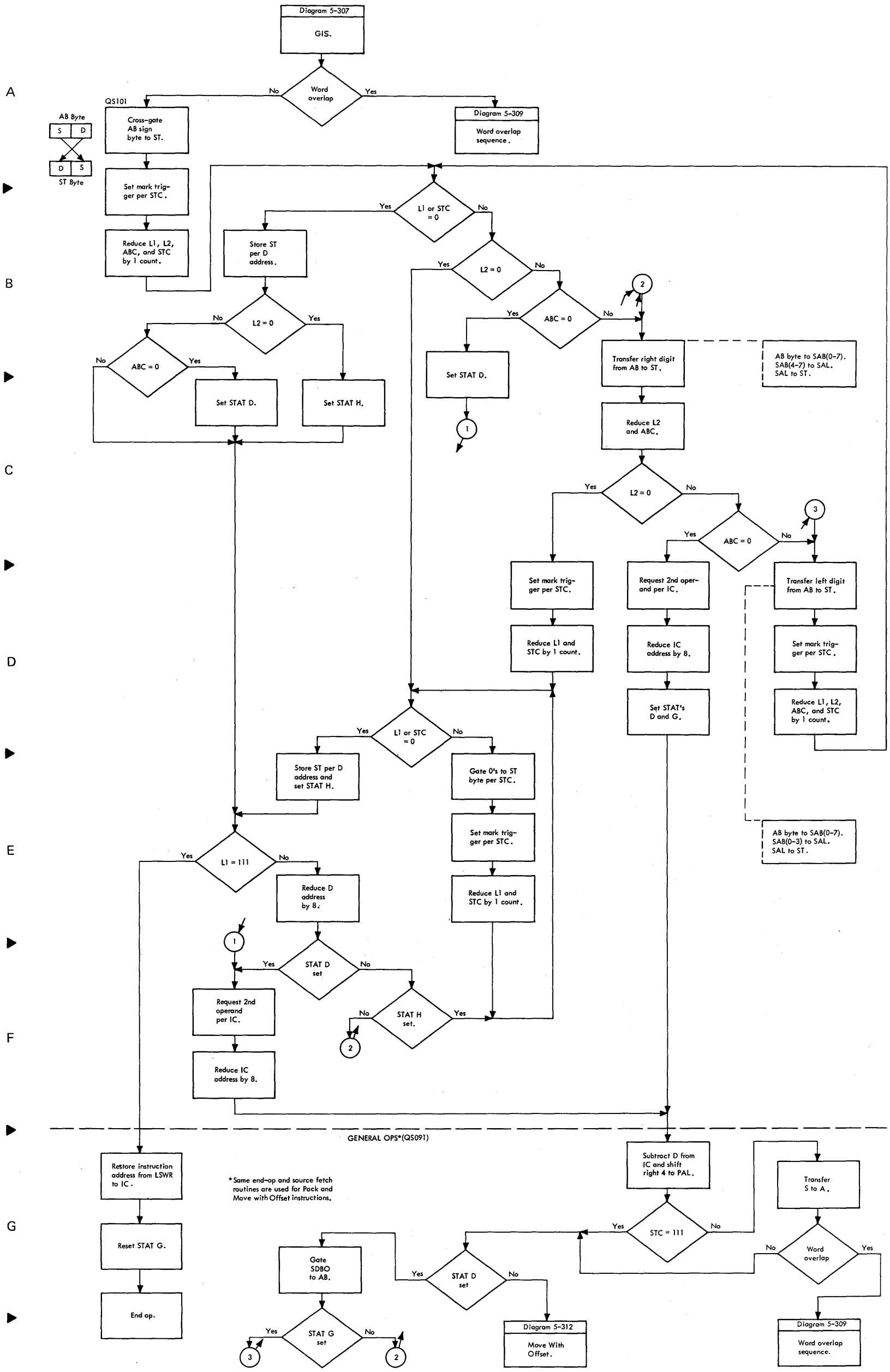


Diagram 5-308. Pack, Not Word Overlap Sequence

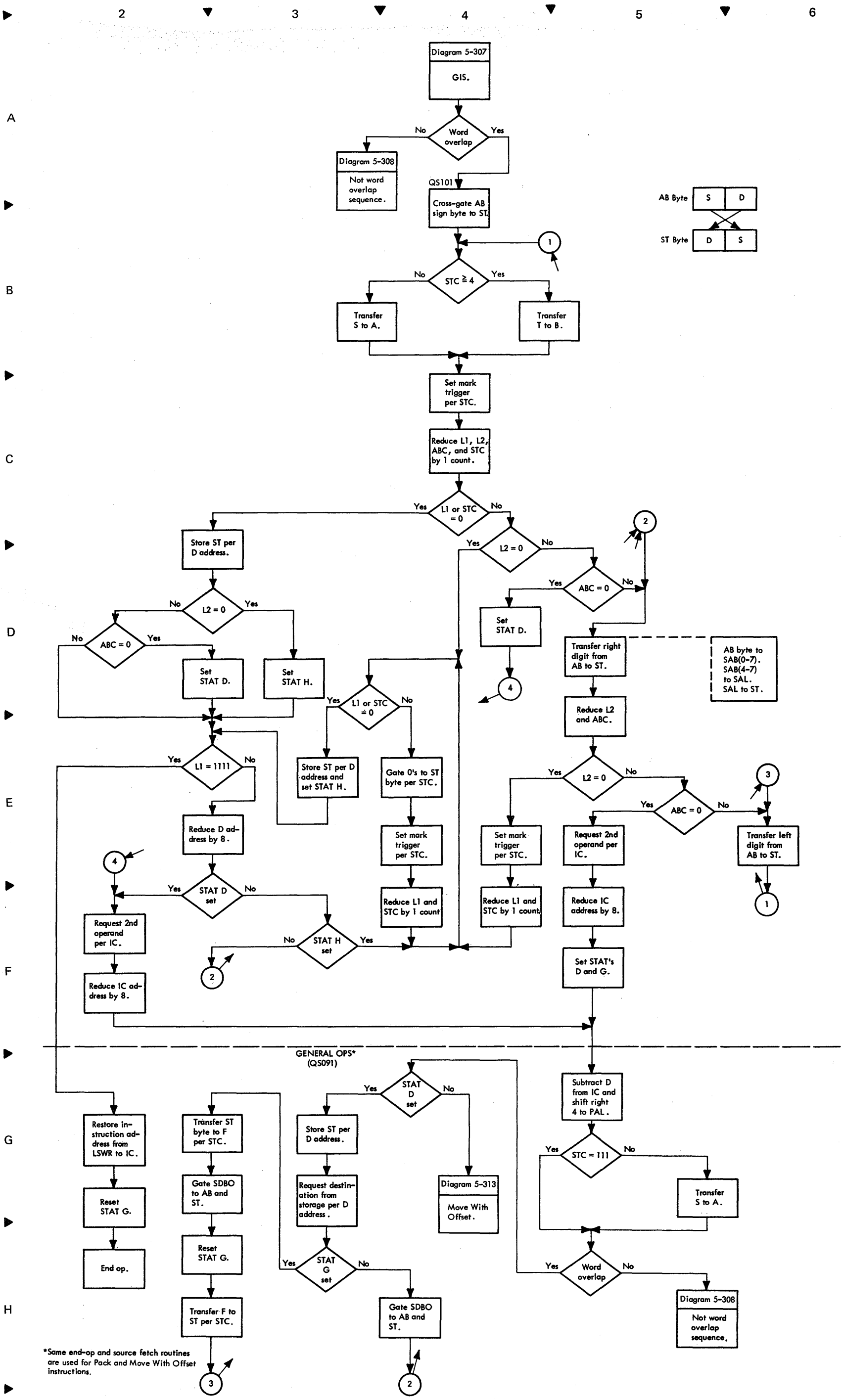


Diagram 5-309. Pack, Word Overlap Sequence

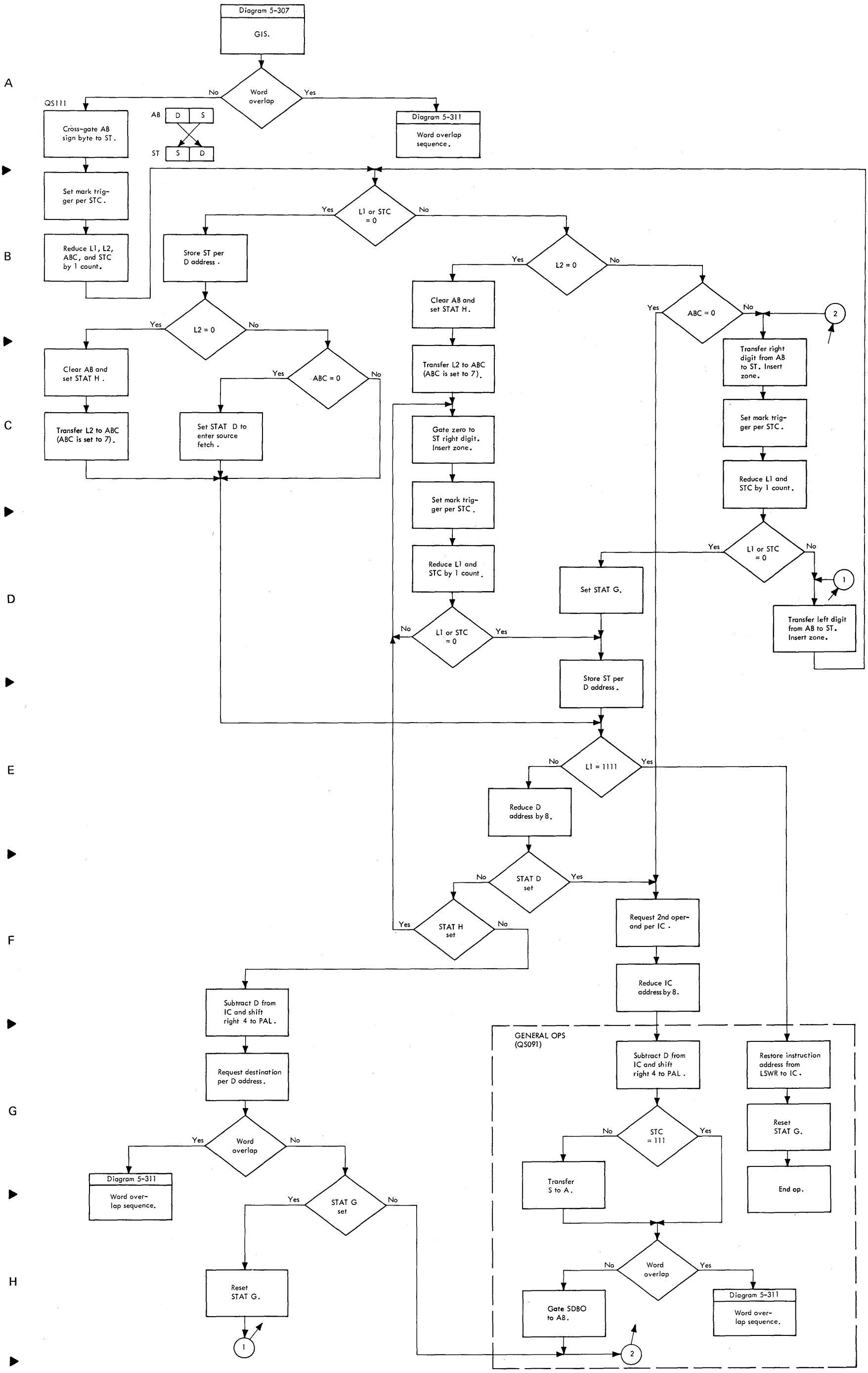


Diagram 5-310. Unpack, Not Word Overlap Sequence

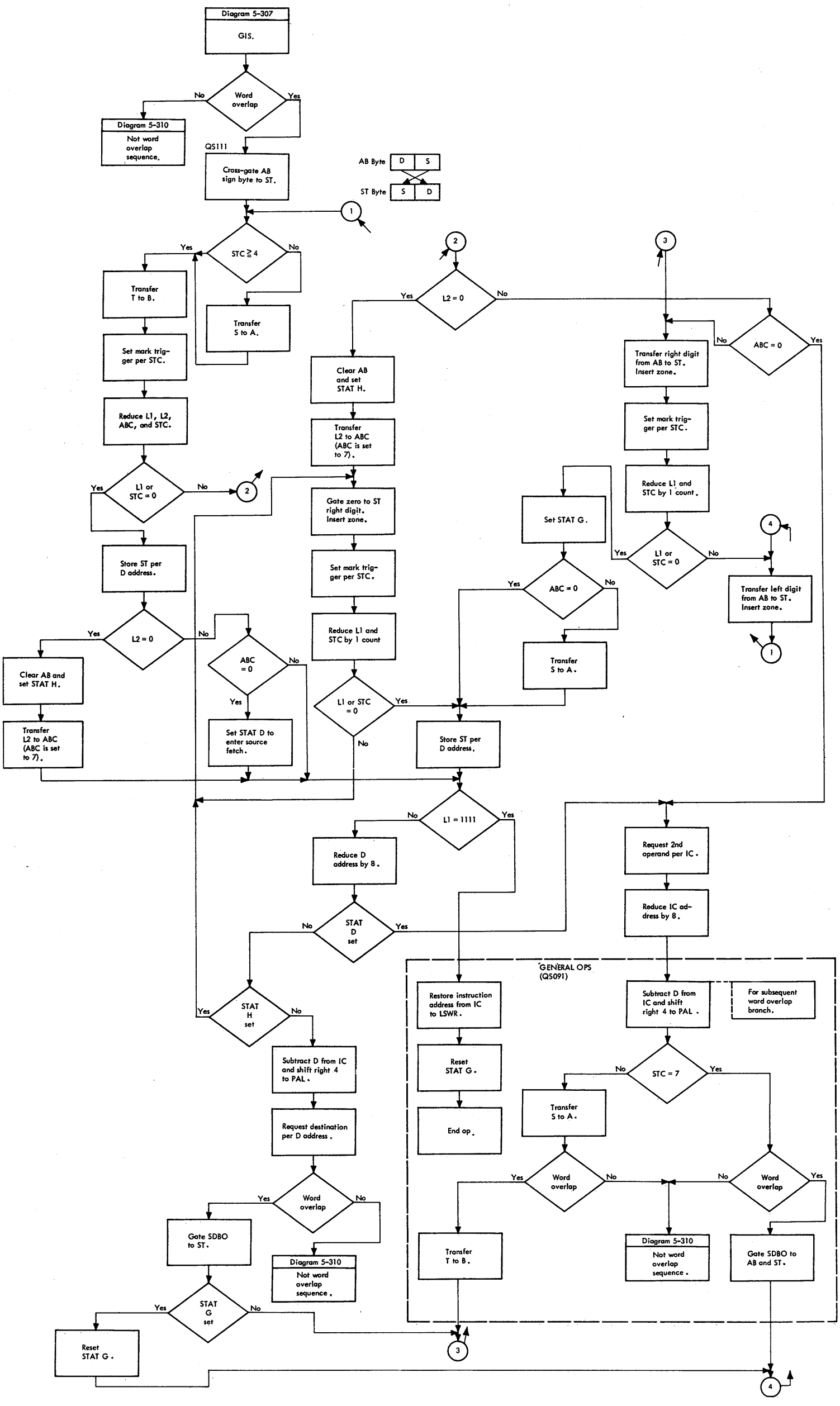


Diagram 5-311. Unpack, Word Overlap Sequence

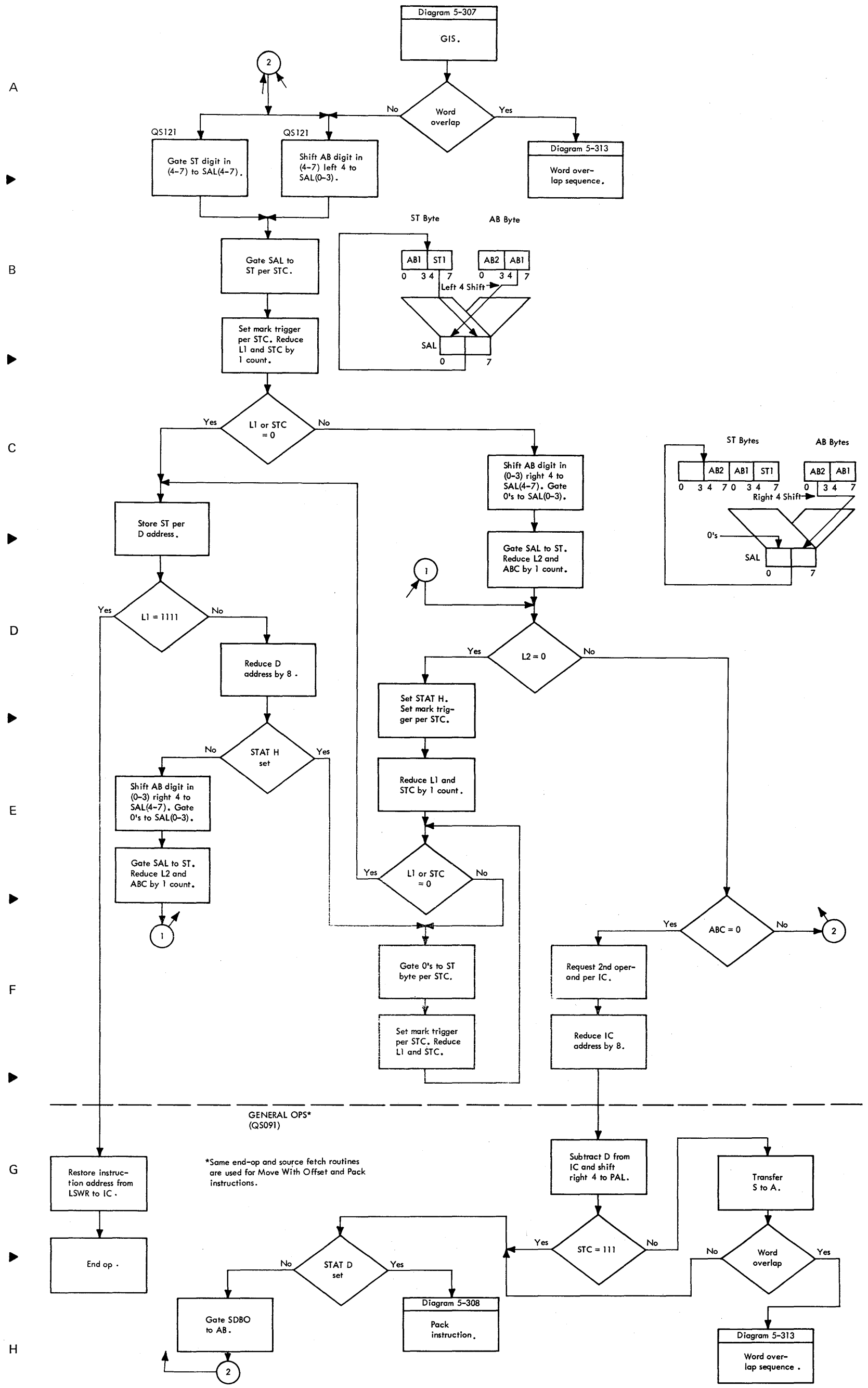


Diagram 5-312. Move With Offset, Not Word Overlap Sequence

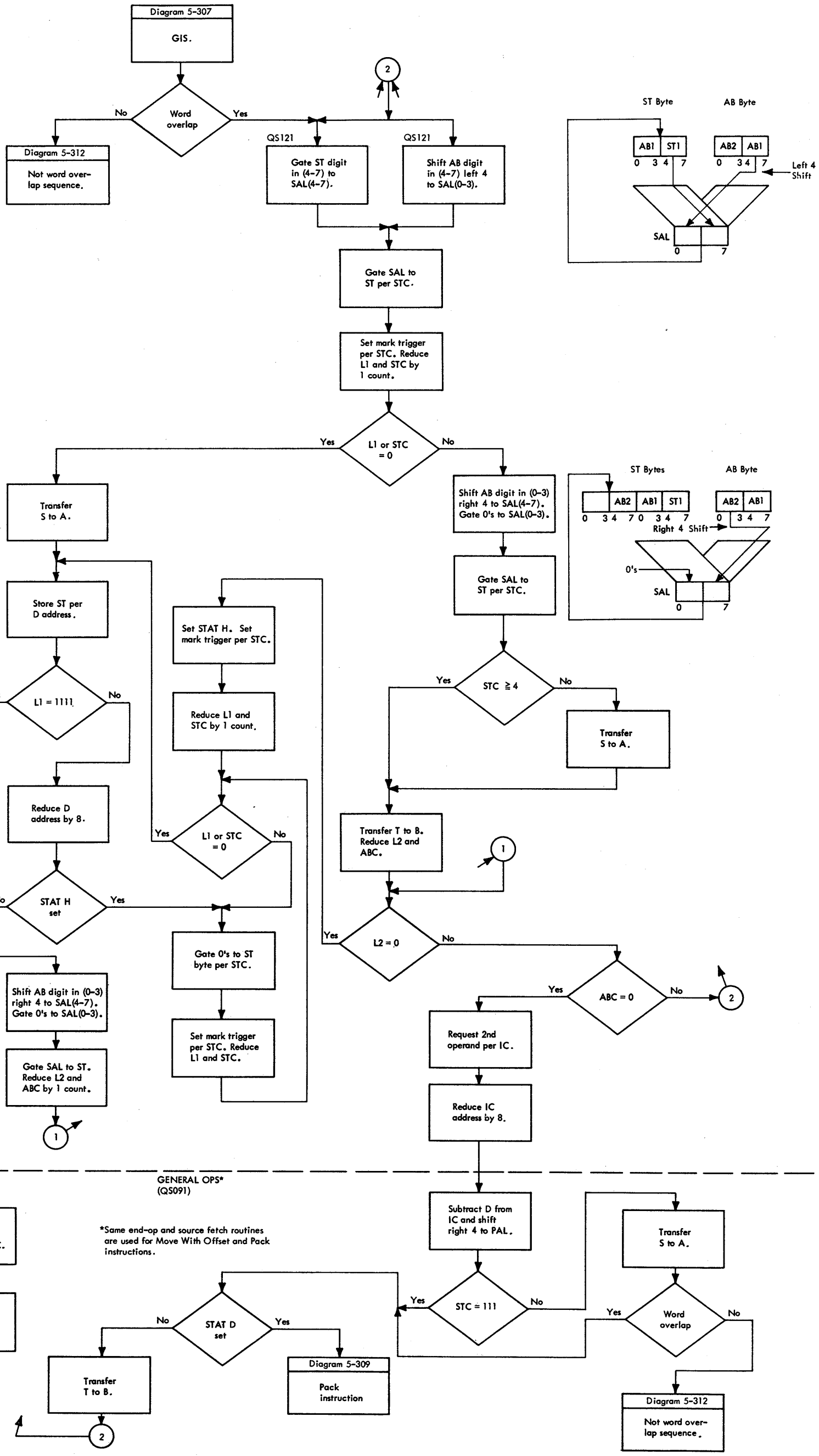


Diagram 5-313. Move With Offset, Word Overlap Sequence

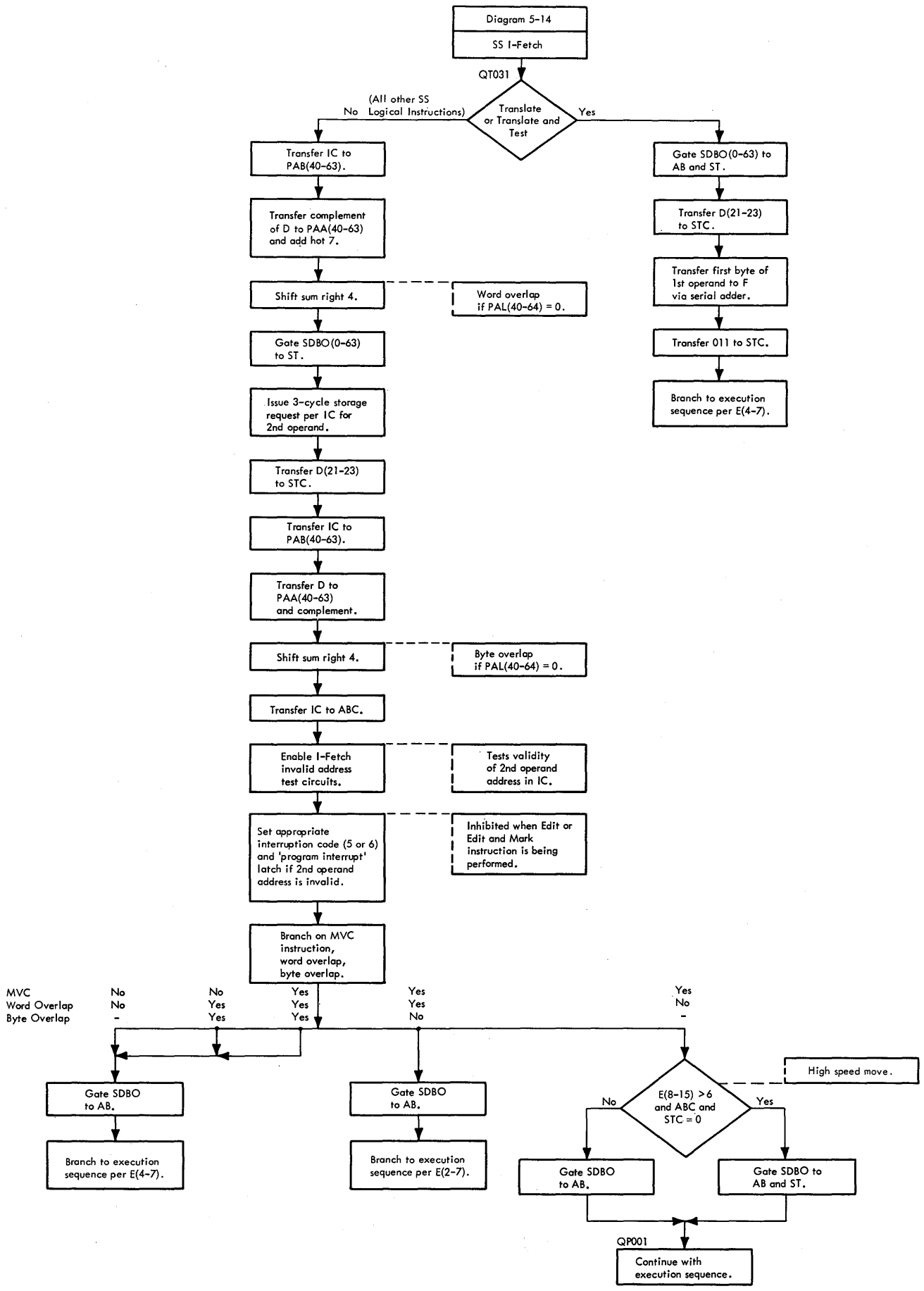
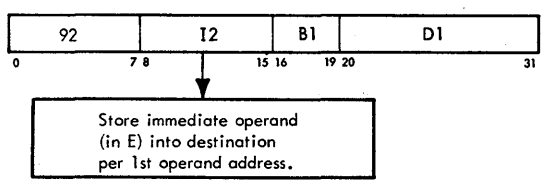
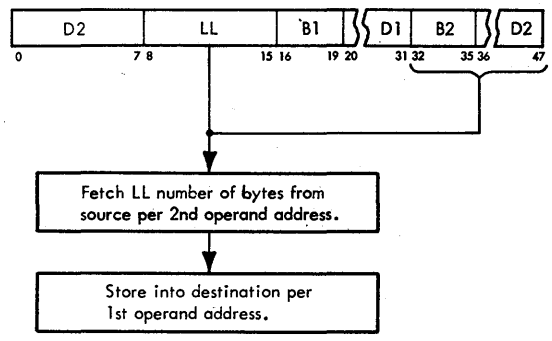


Diagram 5-401. GIS for Logical Instructions



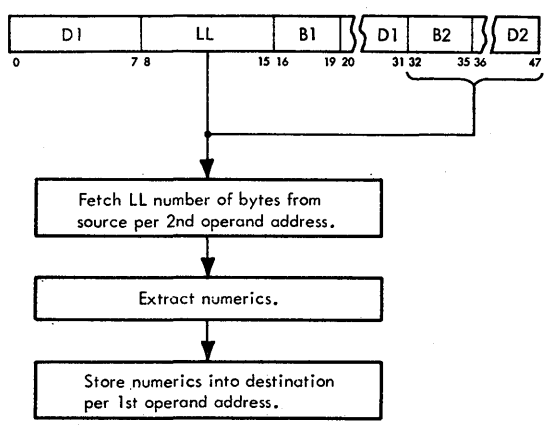
- SI format.
- Purpose: Place immediate operand (I2 of instruction) into 1st operand location (in storage).

A. Move, MVI (92)



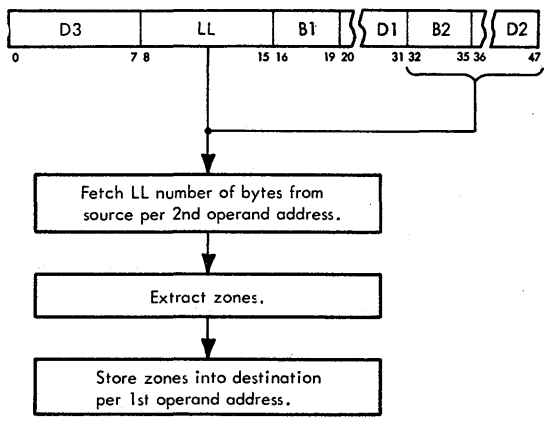
- SS format.
- Purpose: Place 2nd operand (in storage) into 1st operand location (in storage).

B. Move, MVC (D2)



- SS format.
- Purpose: Place numeric portion (low-order 4 bits) of each byte of 2nd operand (in storage) into low-order 4 bits of corresponding byte of 1st operand (in storage).

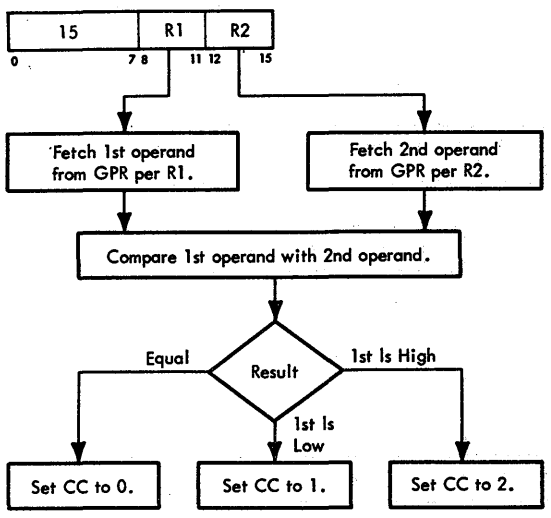
C. Move Numerics, MVN (D1)



- SS format.
- Purpose: Place zone portion (high-order 4 bits) of each byte of 2nd operand (in storage) into high-order 4 bits of corresponding byte of 1st operand (in storage).

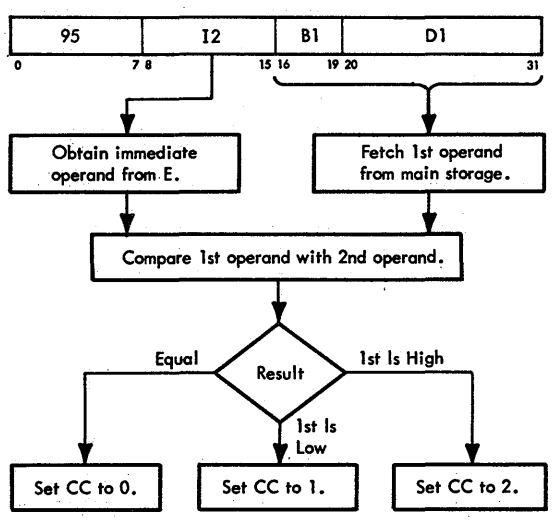
D. Move Zones, MVZ (D3)

Diagram 5-402. Logical Move Instructions



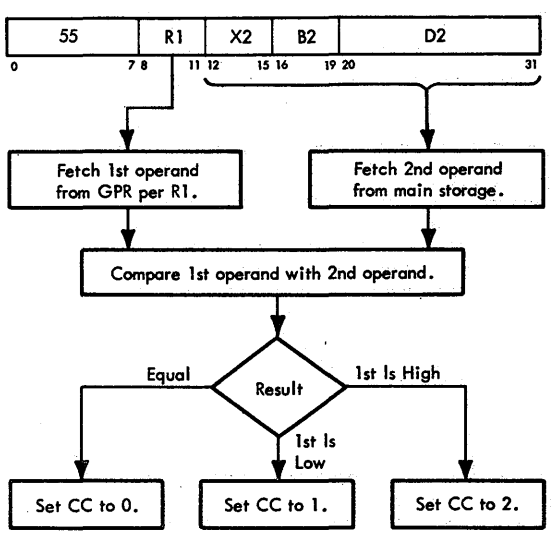
- RR format.
- Purpose: Binarily compare 1st operand (in GPR, per R1) with 2nd operand (in GPR, per R2) and set CC according to result.

A. Compare Logical, CLR (15)



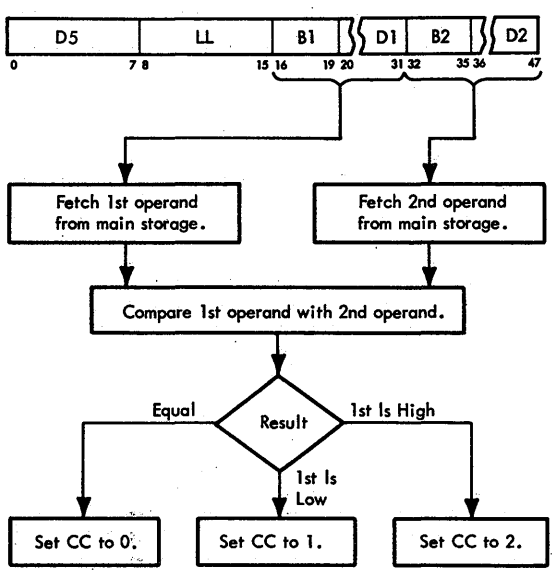
- SI format.
- Purpose: Binarily compare 1st operand (in storage) with immediate operand (I2 of instruction) and set CC according to result.

C. Compare Logical, CLI (95)



- RX format.
- Purpose: Binarily compare 1st operand (in GPR, per R1) with 2nd operand (in storage) and set CC according to result.

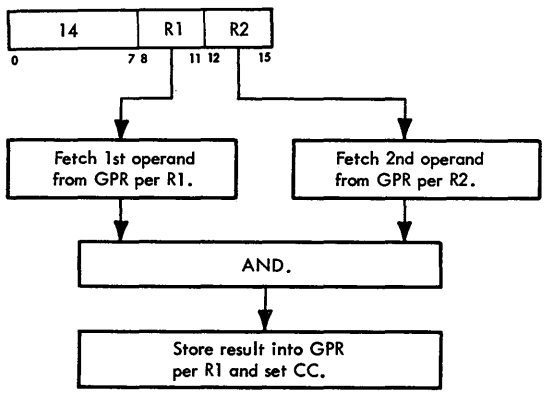
B. Compare Logical, CL (55)



- SS format.
- Purpose: Binarily compare 1st operand (in storage) with 2nd operand (in storage) and set CC according to result.

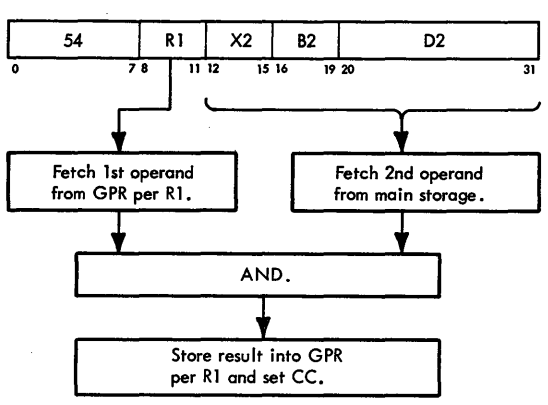
D. Compare Logical, CLC (D5)

Diagram 5-403. Logical Compare Instructions



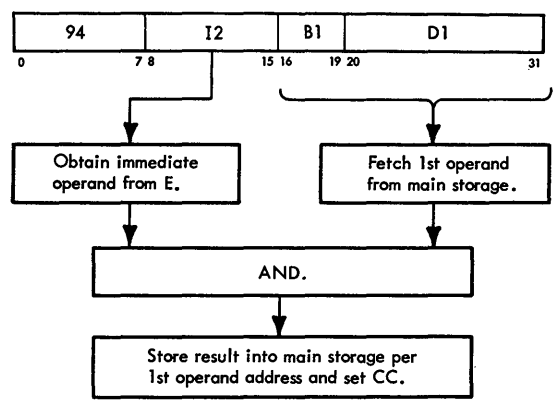
- RR format.
- Purpose: AND 1st operand (in GPR, per R1) with 2nd operand (in GPR, per R2) and place result into 1st operand location.

A. AND, NR (14)



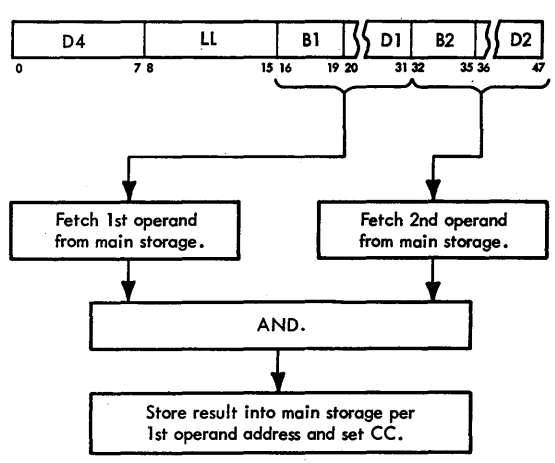
- RX format.
- Purpose: AND 1st operand (in GPR, per R1) with 2nd operand (in storage) and place result into 1st operand location.

B. AND, N (54)



- SI format.
- Purpose: AND immediate operand (I2 of instruction) with 1st operand (in storage) and place result into 1st operand location.

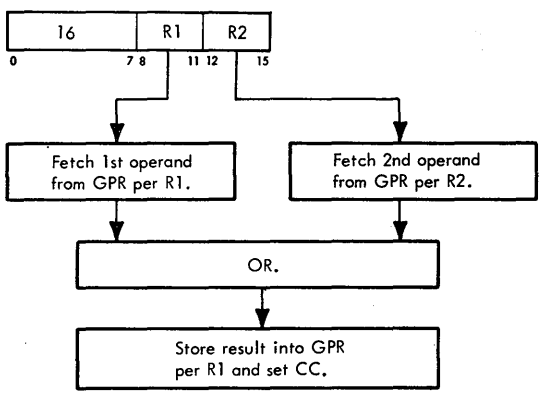
C. AND, NI (94)



- SS format.
- Purpose: AND 1st operand (in storage) with 2nd operand (in storage) and place result into 1st operand location.

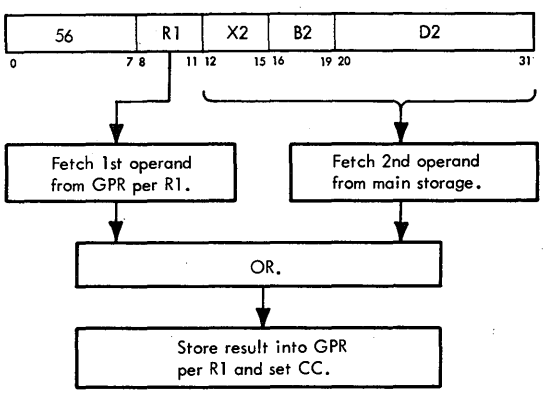
D. AND, NC (D4)

Diagram 5-404. Logical AND Instructions



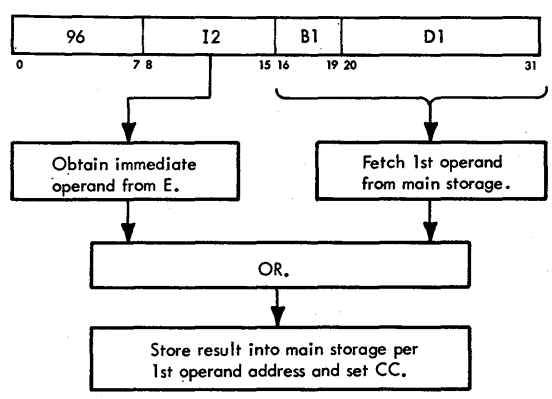
- RR format.
- Purpose: OR 1st operand (in GPR, per R1) with 2nd operand (in GPR, per R2) and place result into 1st operand location.

A. OR, OR (16)



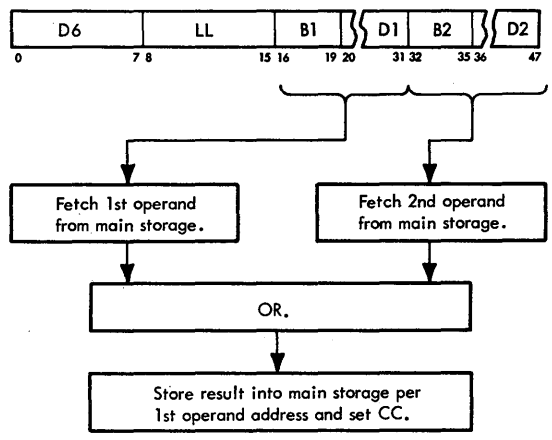
- RX format.
- Purpose: OR 1st operand (in GPR, per R1) with 2nd operand (in storage) and place result into 1st operand location.

B. OR, O (56)



- SI format.
- Purpose: OR immediate operand (I2 of instruction) with 1st operand (in storage) and place result into 1st operand location.

C. OR, OI (96)

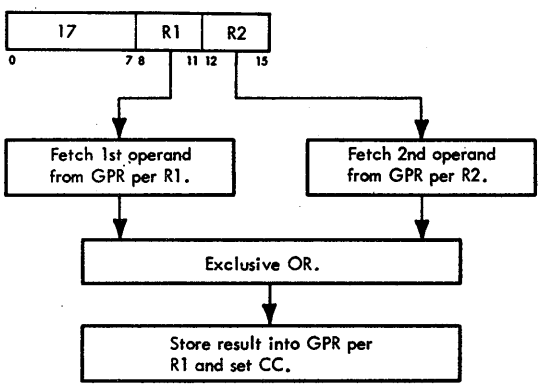


- SS format.
- Purpose: OR 1st operand (in storage) with 2nd operand (in storage) and place result into 1st operand location.

D. OR, OC (D6)

Diagram 5-405. Logical OR Instructions

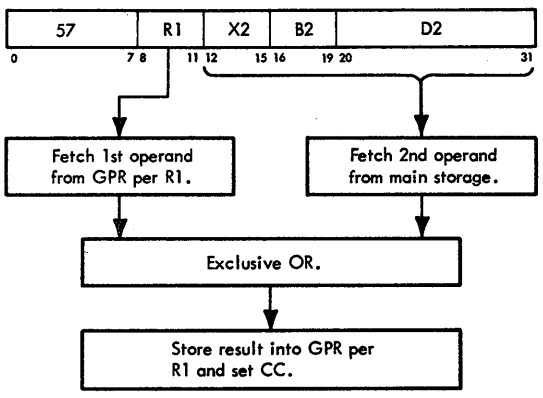
A



- RR format.
- Purpose: Exclusive-OR 1st operand (in GPR, per R1) with 2nd operand (in GPR, per R2) and place result into 1st operand location.

A. Exclusive-OR, XR (17)

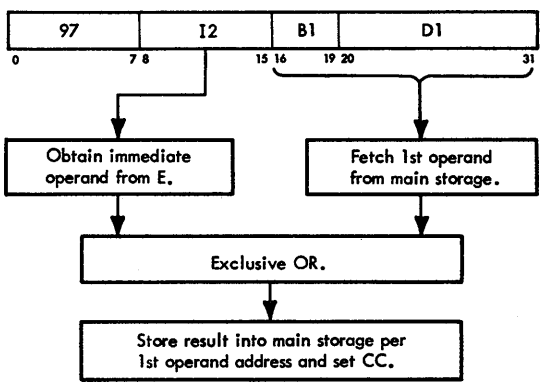
B



- RX format.
- Purpose: Exclusive-OR 1st operand (in GPR, per R1) with 2nd operand (in storage) and place result into 1st operand location.

B. Exclusive-OR, X (57)

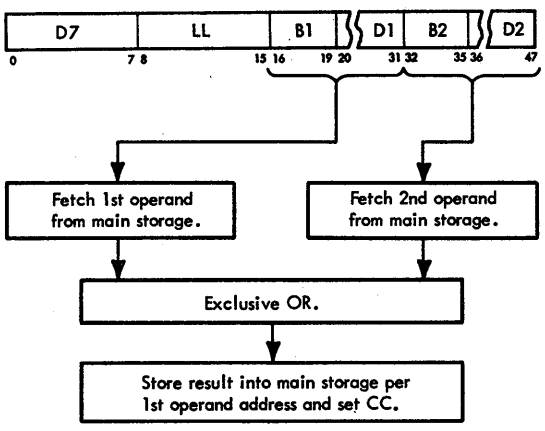
C



- SI format.
- Purpose: Exclusive-OR immediate operand (I2 of instruction) with 1st operand (in storage) and place result into 1st operand location.

C. Exclusive-OR, XI (97)

D

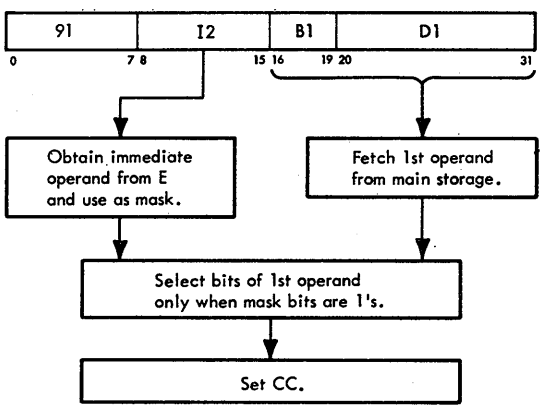


- SS format.
- Purpose: Exclusive-OR 1st operand (in storage) with 2nd operand (in storage) and place result into 1st operand location.

D. Exclusive-OR, XC (D7)

Diagram 5-406. Logical Exclusive-OR Instructions

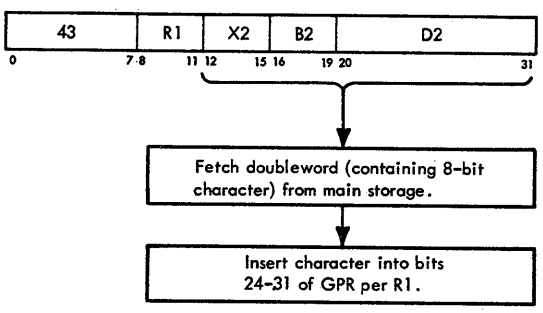
E



- SI format.
- Purpose: Set CC according to state of 1st operand bits (in storage) selected by mask bits (I2 of instruction).

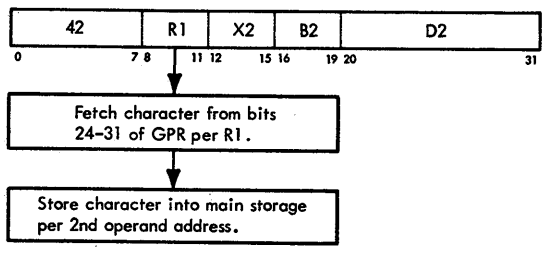
Diagram 5-407. Test Under Mask, TM (91)

F



- RX format.
- Purpose: Insert 2nd operand (byte; in storage) into bits 24-31 of 1st operand location (in GPR, per R1).

A. Insert Character, IC (43)

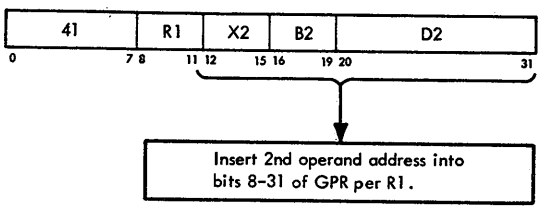


- RX format.
- Purpose: Store bits 24-31 of 1st operand (in GPR, per R1) into 2nd operand location (in storage).

B. Store Character, STC (42)

Diagram 5-408. Insert Character, IC (43); Store Character, STC (42)

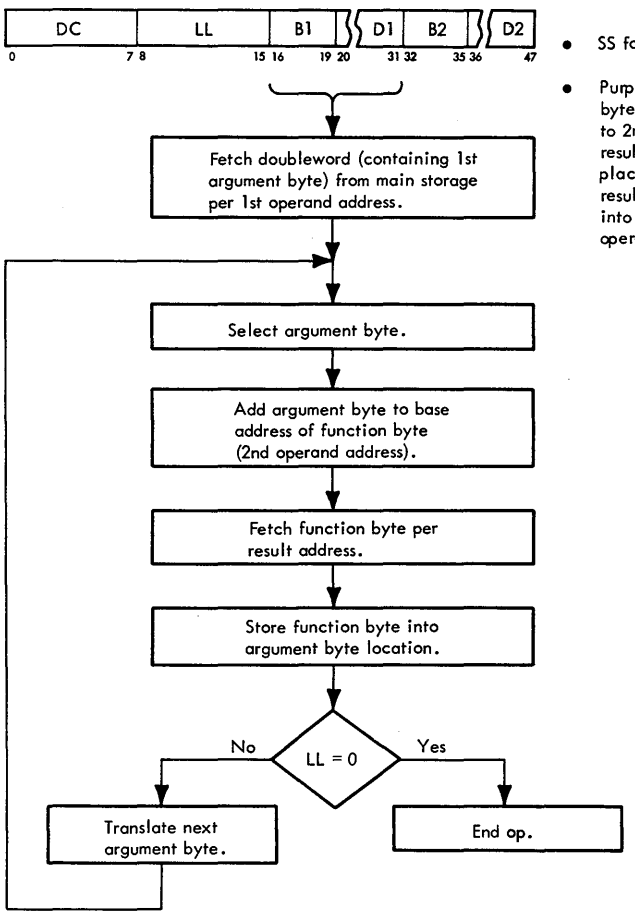
G



- RX format.
- Purpose: Insert 2nd operand address into bits 8-31 of GPR specified by R1.

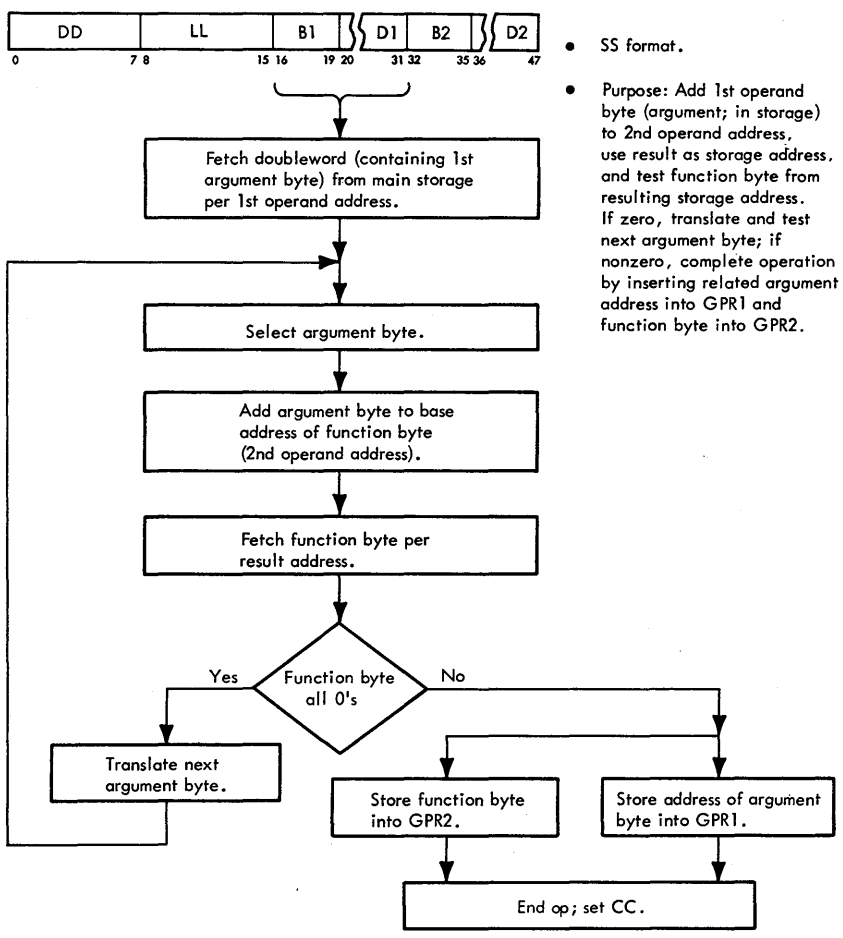
Diagram 5-409. Load Address, LA (41)

H



A. Translate, TR (DC)

- SS format.
- Purpose: Add 1st operand byte (argument; in storage) to 2nd operand address, use result as storage address, and place function byte from resulting storage address into corresponding 1st operand byte location.

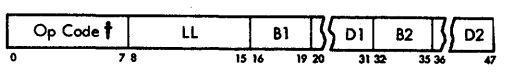


B. Translate and Test, TRT (DD)

- SS format.
- Purpose: Add 1st operand byte (argument; in storage) to 2nd operand address, use result as storage address, and test function byte from resulting storage address. If zero, translate and test next argument byte; if nonzero, complete operation by inserting related argument address into GPR1 and function byte into GPR2.

Diagram 5-410. Translate, TR (DC); Translate and Test, TRT (DD)

- SS format:



† DE for Edit
DF for Edit and Mark

- Purpose:

1. Edit: change format of source (2nd operand; in storage) from packed to zoned, edit source under control of pattern (1st operand; in storage), and place result into 1st operand location

2. Edit and Mark: same as Edit, but, in addition, place location of each 1st significant digit into GPR1.

- Conditions at end of GIS:

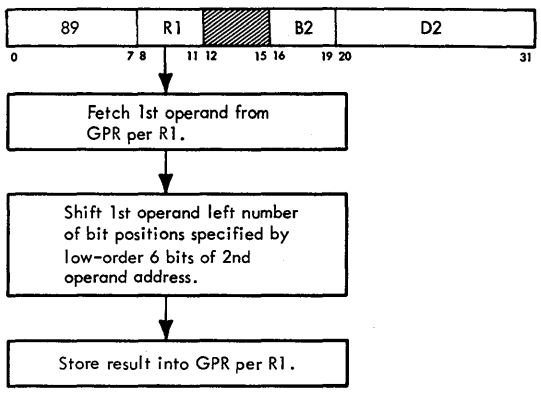
1. Pattern (destination operand) is in ST.
2. Pattern address is in D.
3. Source address (contents of GPR per B2, + D2) is in IC.
4. 1st 16 bits of instruction are in E.

Diagram 5-401
Logical GIS.
Store 1st pattern character (fill) into F.
QP041
Gate pattern character to SBB.
Decode pattern character
Digit Select (20) or Significant Start (21)
Field Separator (22)
Other Character
Examine source digit and gate AB byte to SAB.
Appropriate interruption code (5 or 6) and 'program interrupt' latch are set now if 2nd operand address is invalid.
Reset 'S' trigger and STAT A.
Invalid digit in SAB(0-3)
Yes
No
Set STAT E.
'S' trigger set
Source digit = 0
Significant start (21)
Set 'S' trigger.
Sign in SAB (4-7) or 'right digit' trigger set
Reset 'right digit' trigger and increase ABC by 1.
Set 'right digit' trigger to examine right digit.
Reset 'S' trigger if source sign is plus (+).
Set 'EDMRK' latch.
Set 'S' trigger and STAT A.
'S' trigger set.
Insert fill character into ST. Set mark trigger per STC.
Leave pattern character in ST. Set mark trigger per STC.
'Right digit' trigger set
Gate right digit to ST and insert zone.
Gate left digit to ST and insert zone.
Increase STC and D by 1. Reduce LL by 1.
Set mark trigger per STC. Reduce LL by 1.
Pattern field exhausted or LL = 0 or STC = 0
Store ST per D address.
'EDMRK' latch set
Store D contents into GPR1.
Address of last result byte in ST.
STAT E set
Set interruption code 7 and 'program interrupt' latch.
LL = 1111
Set CC and take end op.
CC setting: 1 if STAT A and 'S' trigger set; 2 if STAT A set and 'S' trigger reset.
Fetch new pattern to ST.
Source field exhausted or ABC = 0
Increase IC by 8 and fetch new source to AB.

Diagram 5-411. Edit, ED (DE); Edit and Mark, EDMK (DF)

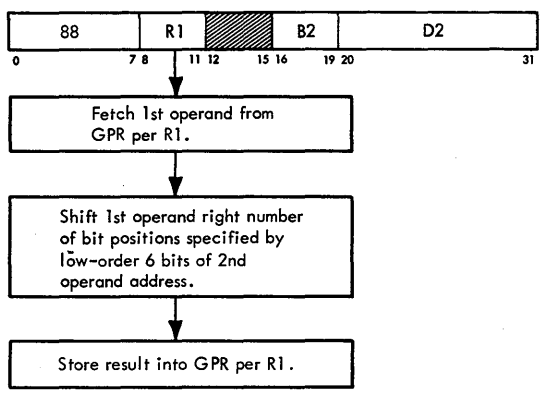
2 3 4 5 6

A



A. Shift Left Single, SLL (89)

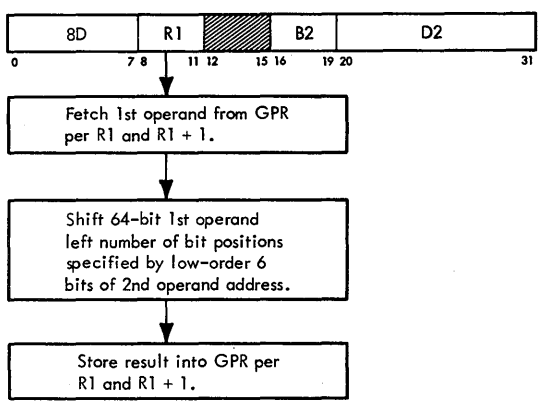
- RS format.
- Purpose: Shift 1st operand (in GPR, per R1) left number of bit positions specified by low-order 6 bits of 2nd operand address.



C. Shift Right Single, SRL (88)

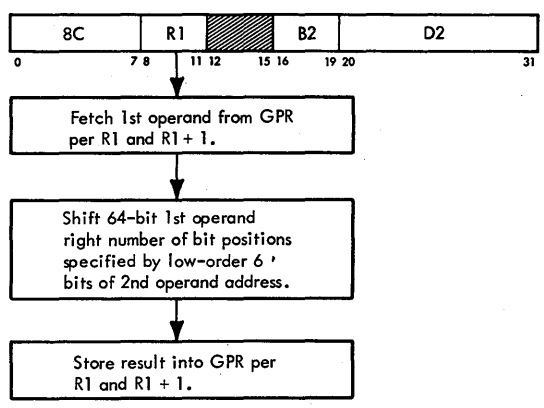
- RS format.
- Purpose: Shift 1st operand (in GPR, per R1) right number of bit positions specified by low-order 6 bits of 2nd operand address.

B



B. Shift Left Double, SLDL (8D)

- RS format.
- Purpose: Shift 1st operand (in GPR, per R1 and R1 + 1) left number of bit positions specified by low-order 6 bits of 2nd operand address.



D. Shift Right Double, SRDL (8C)

- RS format.
- Purpose: Shift 1st operand (in GPR, per R1 and R1 + 1) right number of bit positions specified by low-order 6 bits of 2nd operand address.

Diagram 5-412. Logical Shift Instructions

C D E F G H

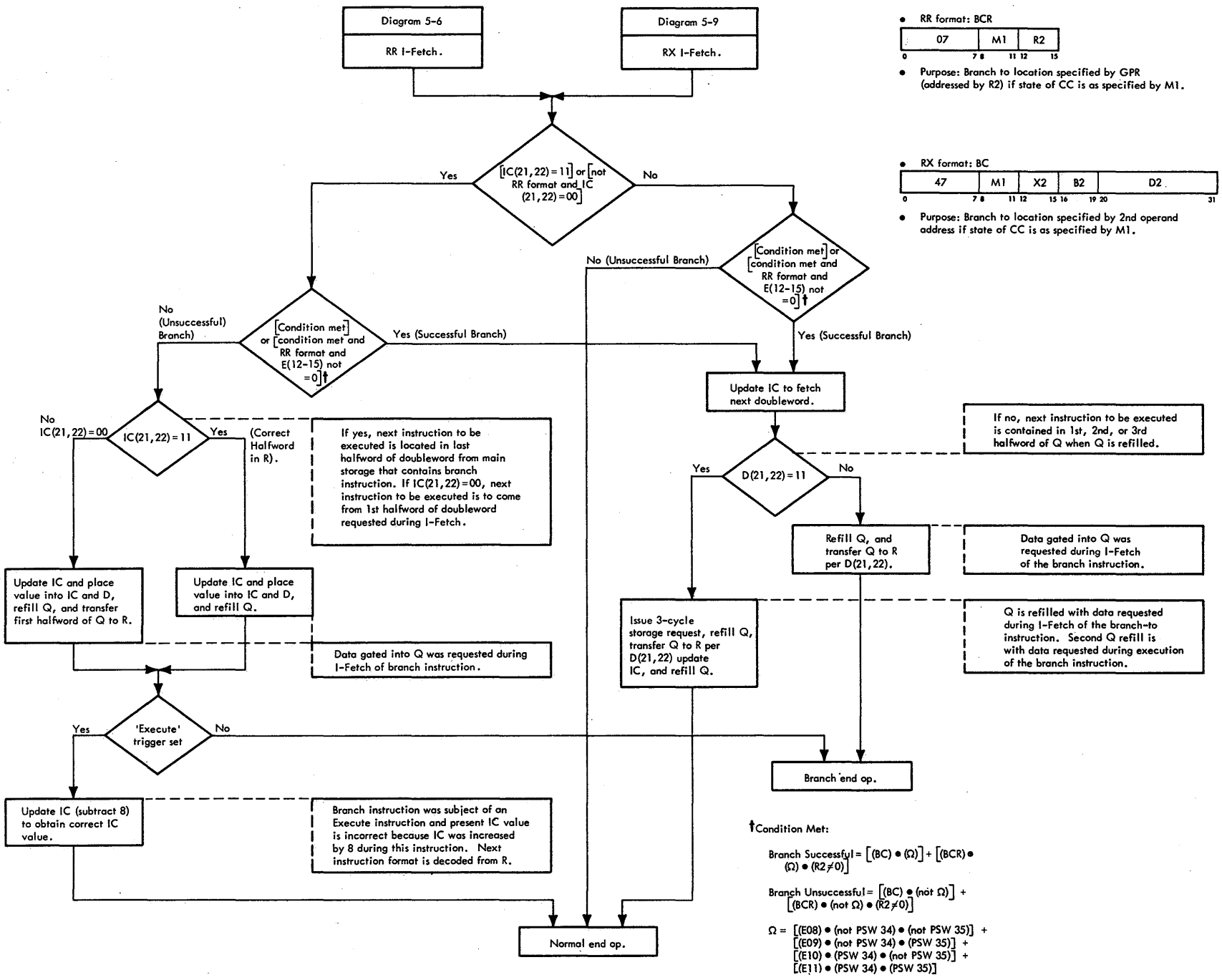
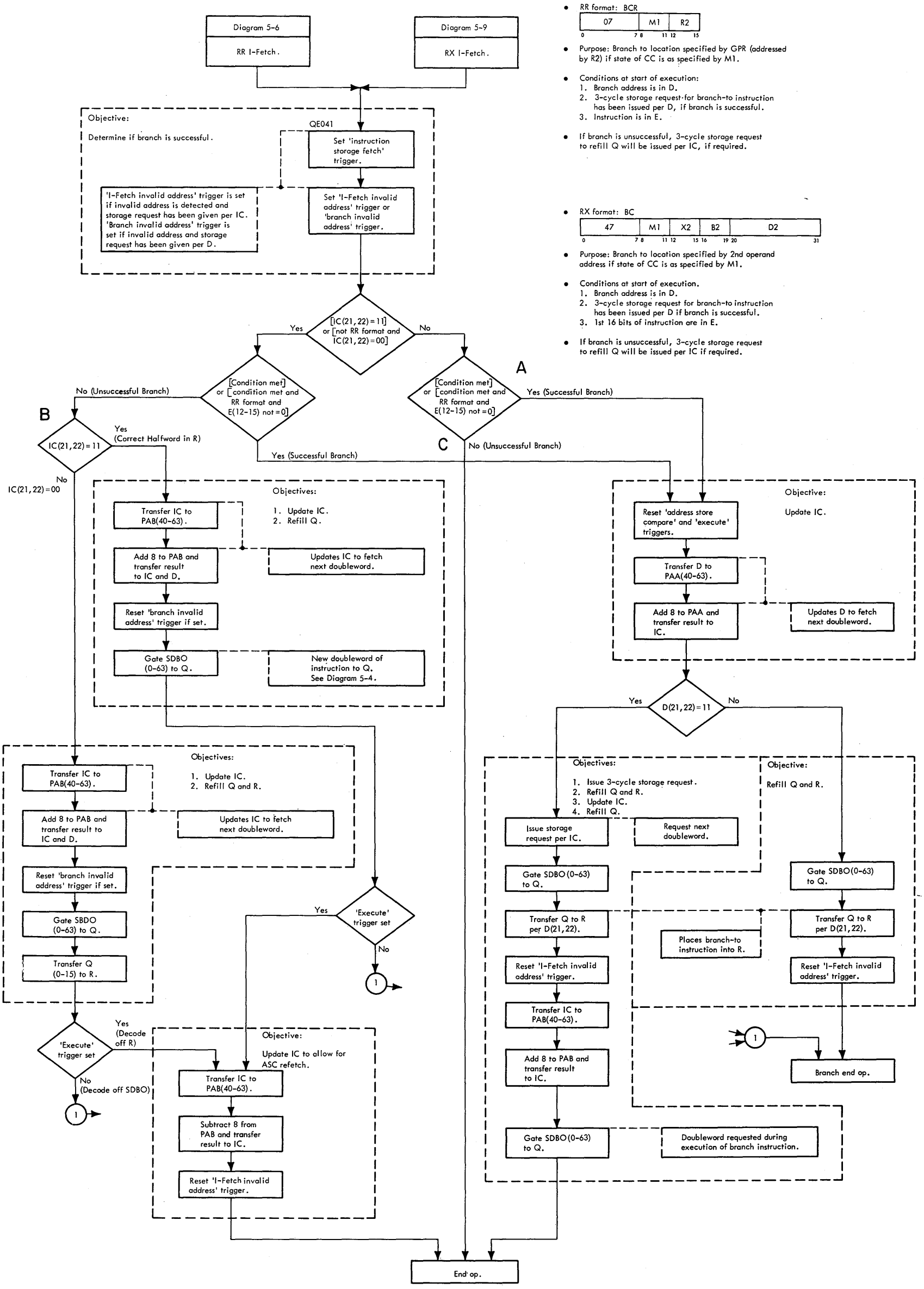


Diagram 5-501. Branch On Condition, BCR (07); BC (47) (Sheet 1 of 2)



- RR format: BCR

- Purpose: Branch to location specified by GPR (addressed by R2) if state of CC is as specified by M1.
- Conditions at start of execution:
  1. Branch address is in D.
  2. 3-cycle storage request for branch-to instruction has been issued per D, if branch is successful.
  3. Instruction is in E.
- If branch is unsuccessful, 3-cycle storage request to refill Q will be issued per IC, if required.

- RX format: BC

- Purpose: Branch to location specified by 2nd operand address if state of CC is as specified by M1.
- Conditions at start of execution.
  1. Branch address is in D.
  2. 3-cycle storage request for branch-to instruction has been issued per D if branch is successful.
  3. 1st 16 bits of instruction are in E.
- If branch is unsuccessful, 3-cycle storage request to refill Q will be issued per IC if required.

Diagram 5-501. Branch On Condition, BCR (07); BC (47) (Sheet 2 of 2)

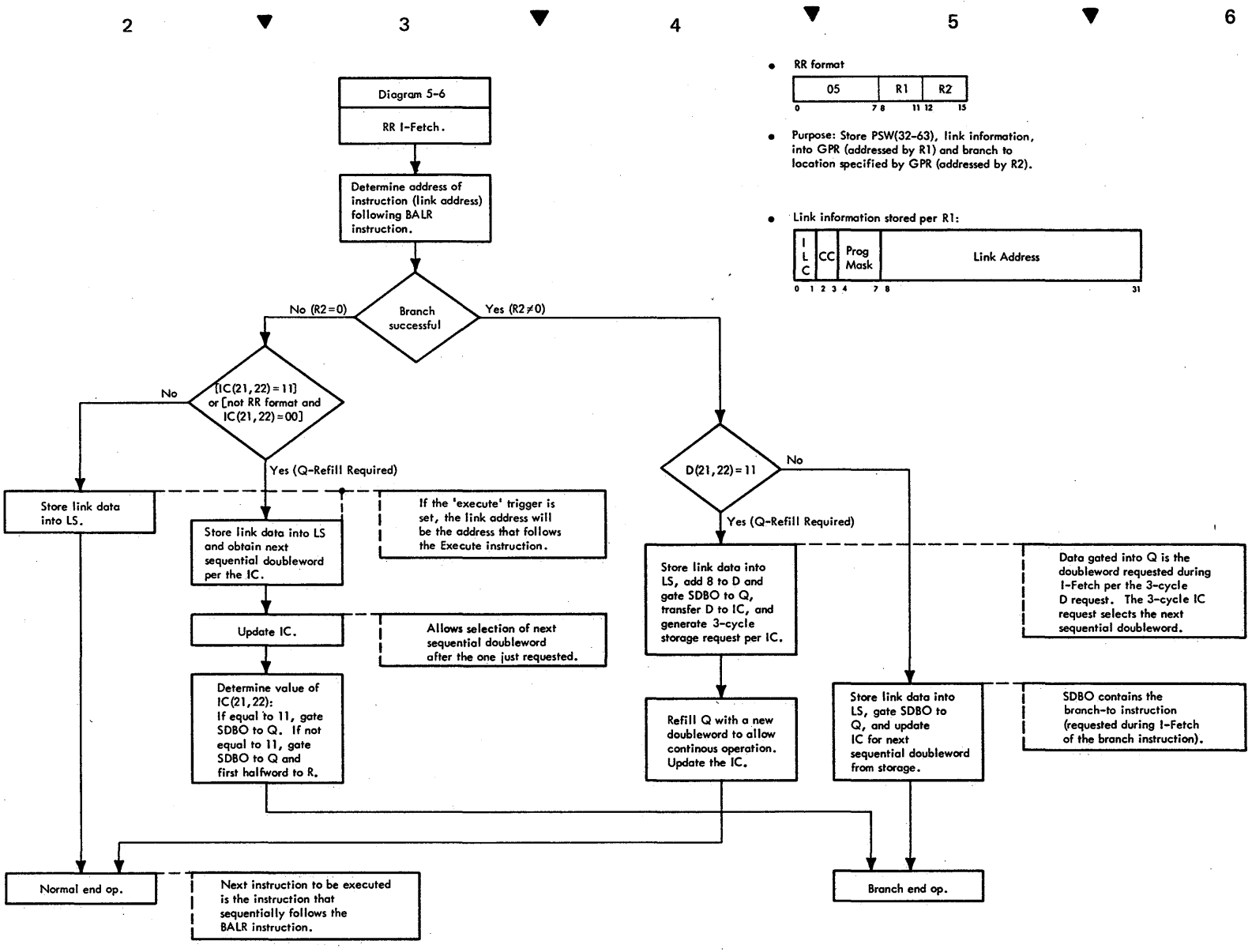
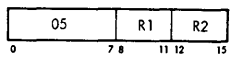


Diagram 5-502. Branch and Link, BALR (05) (Sheet 1 of 2)

- RR format:



- Purpose: Store PSW(32-63), link information, into GPR (addressed by R1) and branch to location specified by GPR (addressed by R2).

- Conditions at start of execution:
  1. 2nd operand is in A, B, and D.
  2. 3-cycle storage request has been issued per D for branch-to instruction.
  3. Instruction is in E.

Diagram 5-6
RR I-Fetch.

Objective:
Determine address to be stored as link address.

QE031

IC(21,22) = 11? Yes / No

'Execute' trigger set? Yes / No

Yes: Transfer IC to PAB(40-63). → Subtract 16 from PAB(40-63). → Transfer PAL to T.

No: Transfer IC to PAB(40-63). → Subtract 8 from PAB(40-63). → Transfer PAL to T.

Set 'instruction storage fetch' trigger.

Branch successful? Yes (R2=0) / No (R2≠0)

Yes (R2=0): D(21,22) = 11? Yes / No

Yes:

Objectives:
1. Store link data into LS.
2. Refill Q.
3. Update IC.
4. Generate 3-cycle storage request.

Transfer PSW(32-39) to T(32-39).
Gate SDBO(0-63) to Q. — Gates branch-to instruction to Q.
Transfer Q to R per D(21,22).
Transfer D to PAA(40-63). — Updates IC for next sequential doubleword.
Add 8 to PAA and transfer result to IC.
Issue 3-cycle storage request per IC.
Set 'I-Fetch invalid address' trigger. — Places link information into LS.
Reset 'I-Fetch invalid address' trigger.
Transfer T to GPR per E(8-11).

QE041
Transfer IC to PAB(40-63).
Add 8 to PAB and transfer result to IC.
Reset 'address store compare' and 'execute' triggers.

Objective:
Refill Q.

QE031
Gate SDBO(0-63) to Q.

No:

Objectives:
1. Store link data into LS.
2. Refill Q.
3. Update IC.

Transfer PSW(32-39) to T(32-39).
Gates branch-to instruction to Q. — Gate SDBO(0-63) to Q.
Transfer Q to R per D(21,22).
Updates IC for next sequential doubleword. — Transfer D to PAA(40-63).
Add 8 to PAA and transfer result to IC.
Reset 'I-Fetch invalid address', 'program store compare', and 'execute' triggers.
Places link information into LS. — Transfer T to GPR per E(8-11).

No (R2≠0): [IC(21,22) = 11] or [not RR format and IC(21,22) = 00]? No / Yes (Q-Refill Required)

No:

Objectives:
1. Store link data into LS.
2. Obtain next sequential doubleword.

Transfer PSW(32-39) to T(32-39).
Transfer T to GPR per E(8-11). — Places link information into LS.

QE041
Issue 3-cycle storage request per IC.
Set 'instruction storage fetch' trigger.
Set 'I-Fetch invalid address' trigger if invalid address.
Reset 'branch invalid address' trigger if set.

'Execute' trigger set? Yes / No

Yes:

Objective:
Store link data into LS.

Transfer PSW(32-39) to T(32-39).
Places link information into LS. — Transfer T to GPR per E(8-11).

Reset 'I-Fetch invalid address' trigger.

Normal end op.

No:

Objectives:
1. Update IC.
2. Transfer IC value to D.

Transfer IC to PAB(40-63).
Add 8 to PAB and transfer result to IC and D.

Objectives:
1. Determine value of IC(21,22).
2. Refill Q and/or R.

IC(21,22) = 11? No / Yes

No: Gate SDBO(0-63) to Q. → Transfer Q(0-15) to R.

Yes: Gate SDBO(0-63) to Q.

Branch end op.

Diagram 5-502. Branch and Link BALR (05) (Sheet 2 of 2)

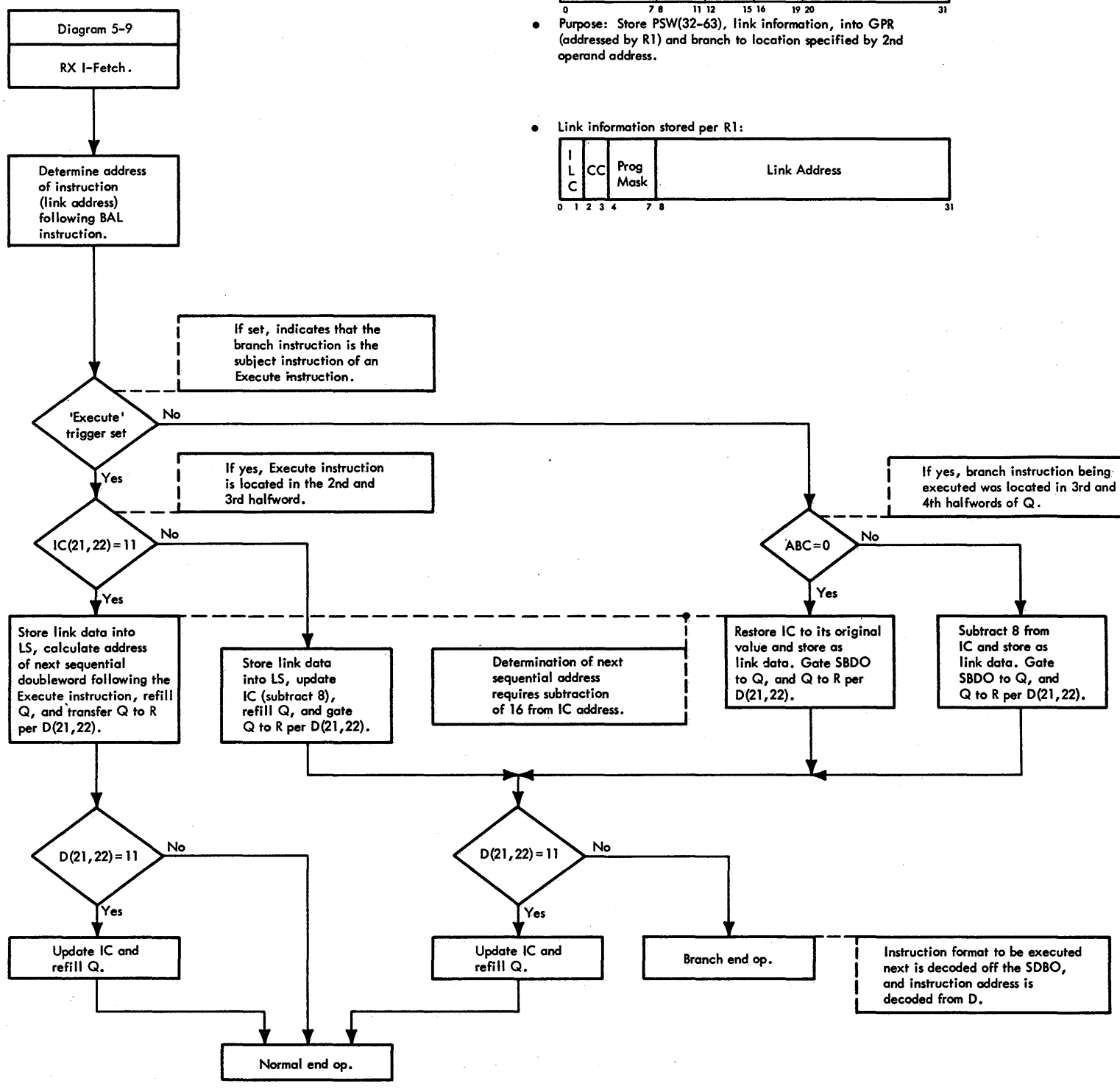


Diagram 5-503. Branch and Link, BAL (45) (Sheet 1 of 2)

- RX format:

45	R1	X2	B2	D2
0 7	8 11	12 15	16 19	20 31

- Purpose: Store PSW(32-63), link information, into GPR (addressed by R1) and branch to location specified by 2nd operand address.
- Conditions at start of execution:
  1. Branch address is in D.
  2. 3-cycle storage request has been issued per D for branch-to instruction.
  3. 1st 16 bits of instruction are in E.

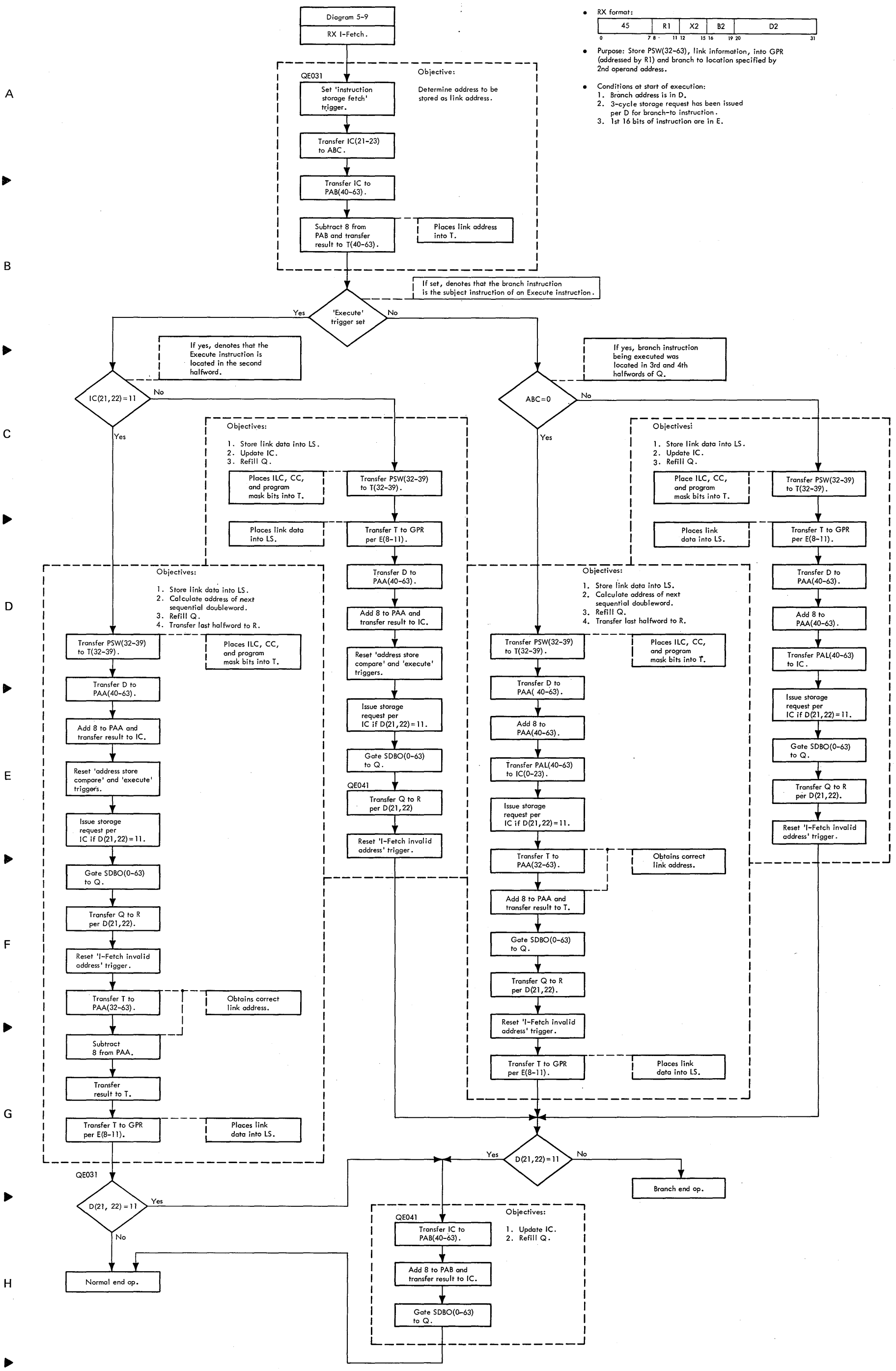
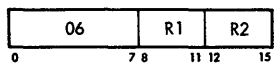


Diagram 5-503. Branch and Link, BAL (45) (Sheet 2 of 2)

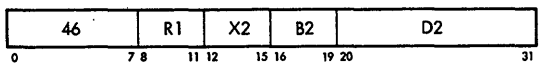
A

- RR format: BCTR



- Purpose: Subtract 1 from 1st operand (in GPR, per R1) and, if result ≠0, branch to location specified by GPR (addressed by R2).

- RX format: BCT



- Purpose: Subtract 1 from 1st operand (in GPR, per R1) and, if result ≠0, branch to location specified by 2nd operand address.

B

C

D

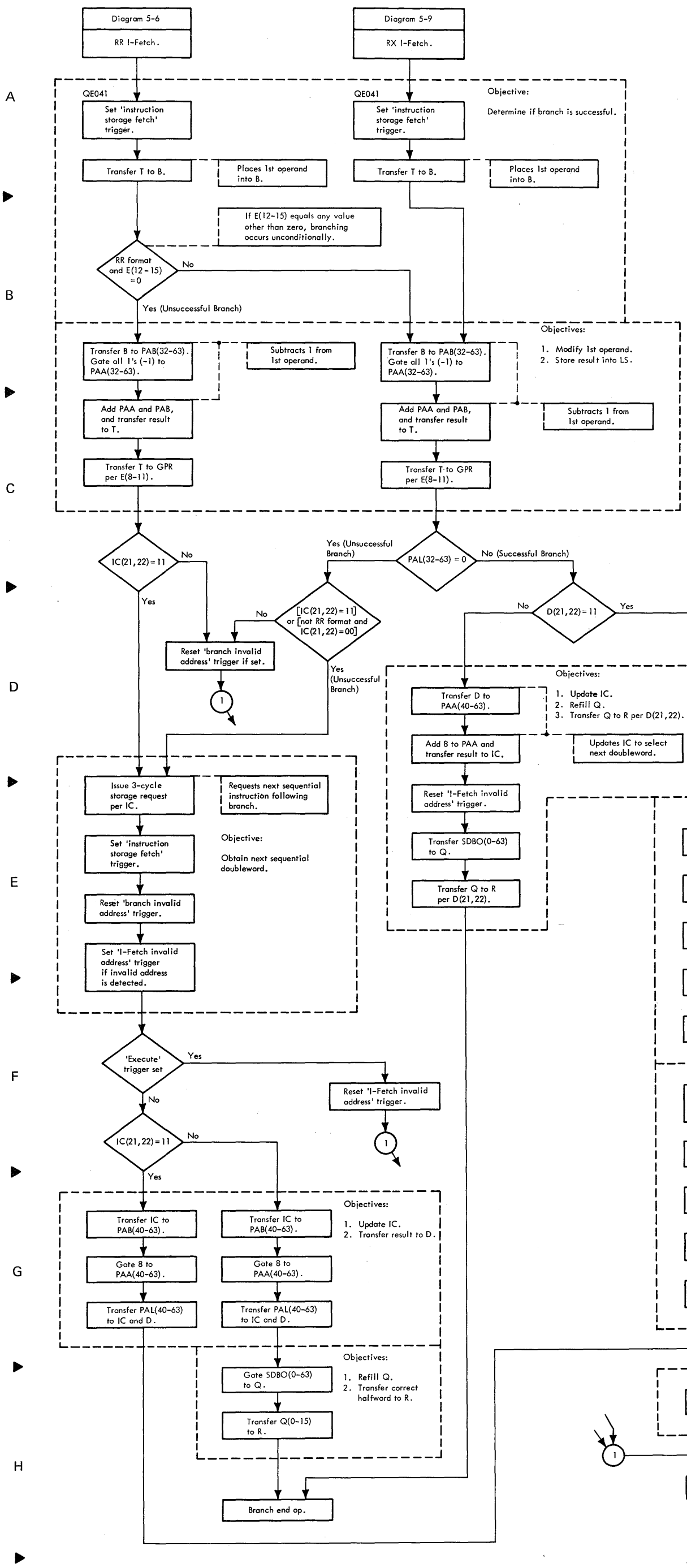
E

F

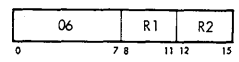
G

H

Diagram 5-504. Branch On Count, BCTR (06); BCT (46) (Sheet 1 of 2)

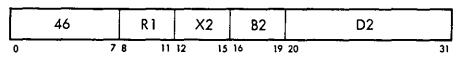


- RR format: BCTR



- Purpose: Subtract 1 from 1st operand (in GPR, per R1) and, if result ≠0, branch to location specified by GPR (addressed by R2).
- Conditions at start of execution:
  1. Branch address is in D.
  2. 3-cycle storage request has been issued per D for branch-to instruction.
  3. 1st operand is in S and T.
  4. Instruction is in E.

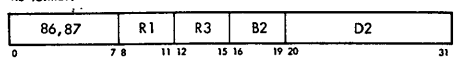
- RX format: BCT



- Purpose: Subtract 1 from 1st operand (in GPR, per R1) and, if result ≠0, branch to location specified by 2nd operand address.
- Conditions at start of execution:
  1. Branch address is in D.
  2. 3-cycle storage request has been issued per D for branch-to instruction.
  3. 1st 16 bits of instruction are in E.

Diagram 5-504. Branch On Count, BCTR (06); BCT (46) (Sheet 2 of 2)

- RS format:

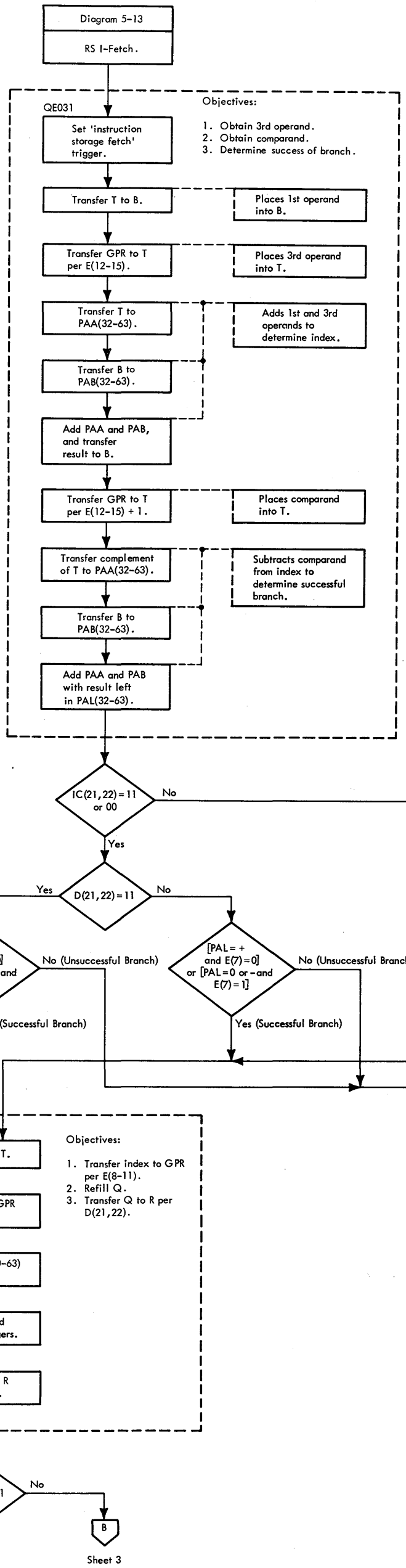


- Purpose:

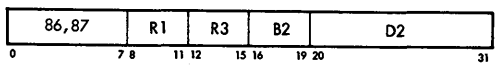
1. BXH - Add increment (3rd operand; in GPR, per R3) to 1st operand (in GPR, per R1), algebraically compare result (index) with comparand (in odd-address GPR specified by R3 or R3 + 1), and, if index is greater than comparand, branch to location specified by 2nd operand address.

2. BXLE - Add increment (3rd operand; in GPR, per R3) to 1st operand (in GPR, per R1), algebraically compare result (index) with comparand (in odd-address GPR specified by R3 or R3 + 1), and, if index is equal to or less than comparand, branch to location specified by 2nd operand address.

Diagram 5-505. Branch on Index High, BXH (86); Branch on Index Low or Equal, BXLE (87) (Sheet 1 of 3)



- RS format:



- Purpose:

1. BXH - Add increment (3rd operand; in GPR, per R3) to 1st operand (in GPR, per R1), algebraically compare result (index) with comparand (in odd-address GPR specified by R3 or R3 + 1), and, if index is greater than comparand, branch to location specified by 2nd operand address.

2. BXLE - Add increment (3rd operand; in GPR, per R3) to 1st operand (in GPR, per R1), algebraically compare result (index) with comparand (in odd-address GPR specified by R3 or R3 + 1), and, if index is equal to or less than comparand, branch to location specified by 2nd operand address.

- Conditions at start of execution:

1. 1st operand is in S and T.
2. Branch address is in D.
3. 3-cycle storage request has been issued per D for the branch-to instruction.
4. 1st 16 bits of instruction are in E.

Diagram 5-505. Branch on Index High, BXH (86); Branch on Index Low or Equal, BXLE (87) (Sheet 2 of 3)

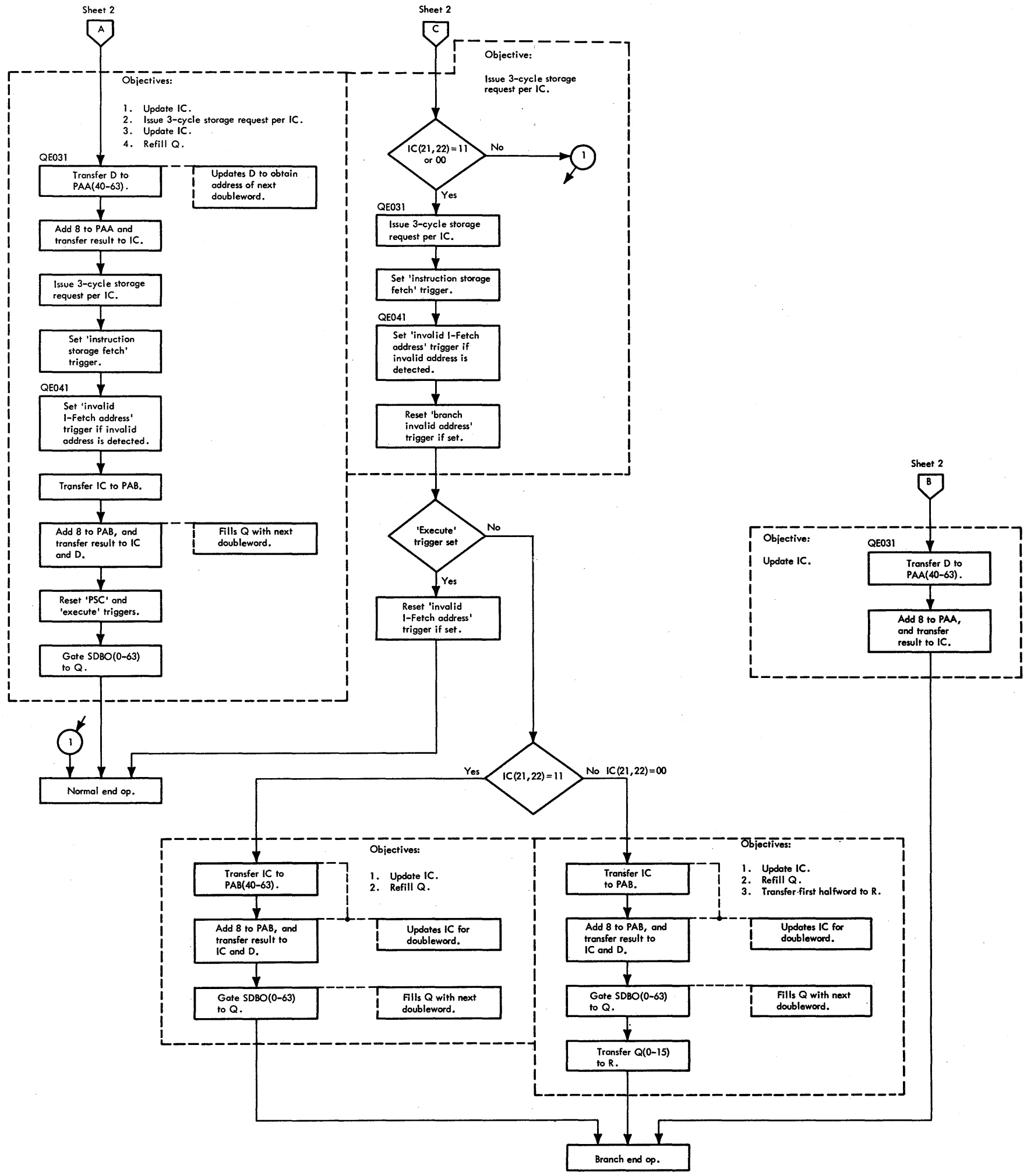
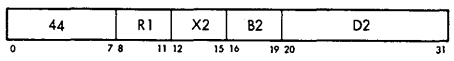


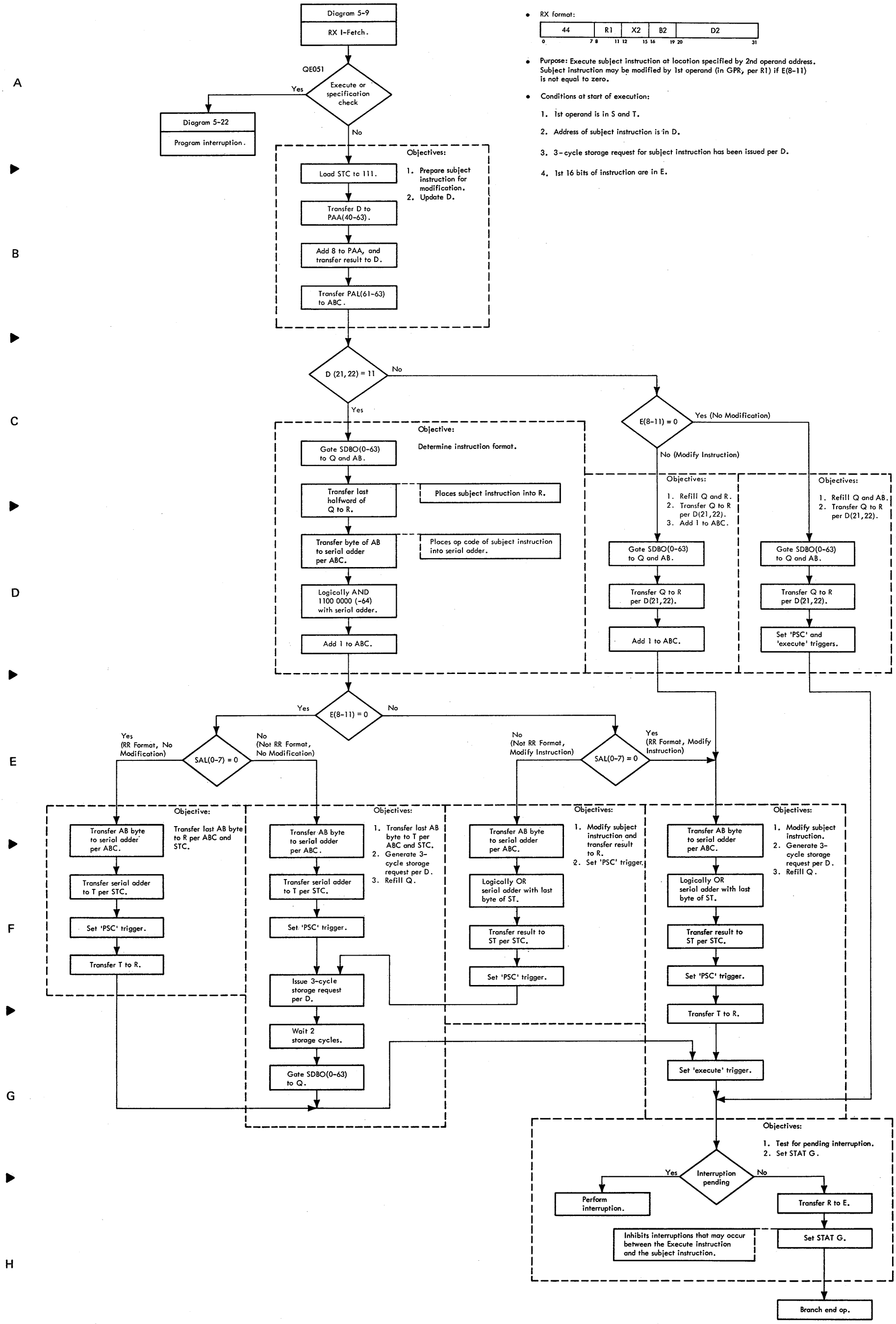
Diagram 5-505. Branch on Index High, BXH (86); Branch on Index Low or Equal, BXLE (87) (Sheet 3 of 3)

- RX format:



- Purpose: Execute subject instruction at location specified by 2nd operand address. Subject instruction may be modified by 1st operand (in GPR, per R1) if E(8-11) is not equal to zero.

Diagram 5-506. Execute, EX (44) (Sheet 1 of 2)



- RX format:

- Purpose: Execute subject instruction at location specified by 2nd operand address. Subject instruction may be modified by 1st operand (in GPR, per R1) if E(8-11) is not equal to zero.
- Conditions at start of execution:
  1. 1st operand is in S and T.
  2. Address of subject instruction is in D.
  3. 3-cycle storage request for subject instruction has been issued per D.
  4. 1st 16 bits of instruction are in E.

Diagram 5-506. Execute, EX (44) (Sheet 2 of 2)

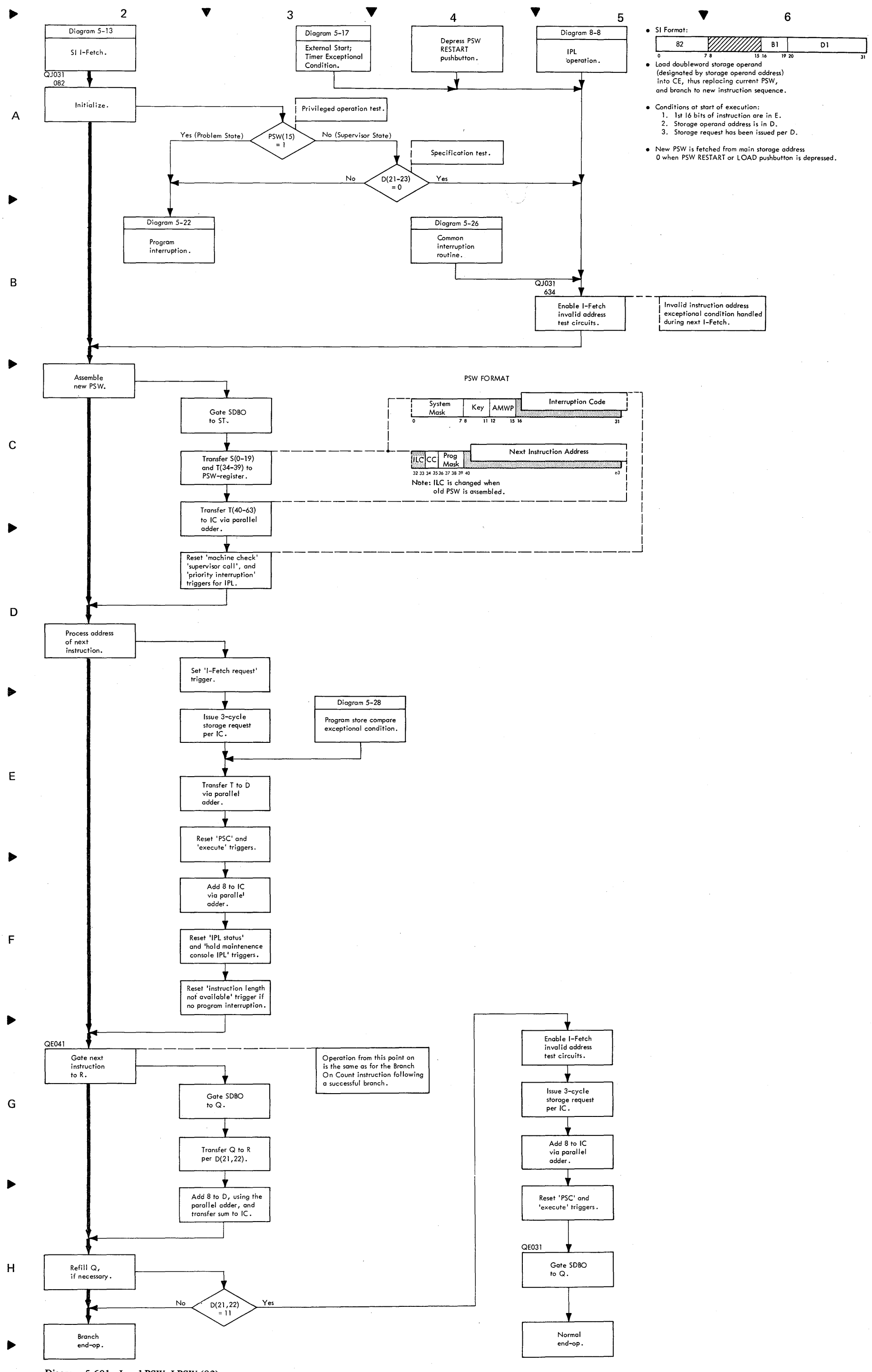


Diagram 5-601. Load PSW, LPSW (82)

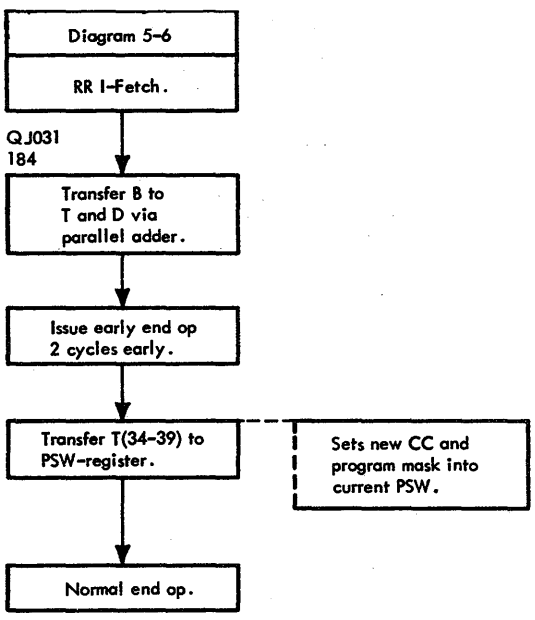
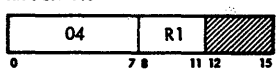
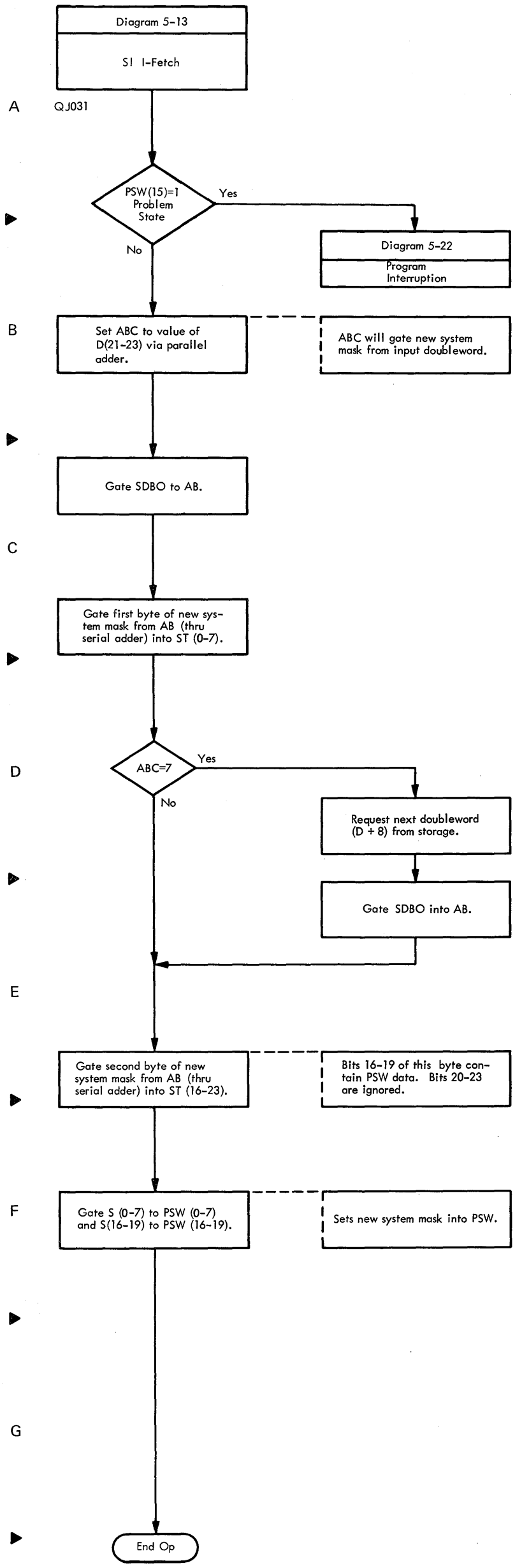


Diagram 5-602. Set Program Mask, SPM (04)

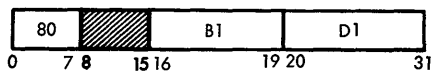
- RR Format:



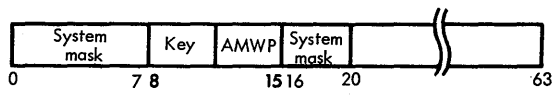
- Replace CC and program mask (bits 34-39) of current PSW with bits 2-7 of 1st operand (in GPR, per R1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D.
  3. 2nd operand is not used.
- Bits 2-7 of 1st operand may have been loaded from PSW-register by a previous Branch and Link instruction.
- Program mask format (set mask bit permits interruption):
  Bit 36 - Fixed-point overflow mask.
  Bit 37 - Decimal overflow mask.
  Bit 38 - Exponent underflow (floating-point) mask.
  Bit 39 - Significance (floating-point) mask.



- `SI Format



- Purpose: Replace system mask (bits 0-7 and 16-19) of current PSW with byte designated by storage operand address and bits 0-3 of the following byte.
- Conditions at start of execution:
  1. First 16 bits of instruction are in E.
  2. Storage operand address is in D.
  3. Storage request has been issued per D.
- System mask format:



0 - Multiplexer channel A mask bit
1 - Selector channel 1A mask bit
2 - Selector channel 2A mask bit
3 - Selector channel 3A mask bit
4 - Multiplexer channel B mask bit
5 - Selector channel 1B mask bit
6 - Selector channel 2B mask bit
7 - External mask bit
16 - Selector channel 3B mask bit
17 - Multiplexer channel C mask bit
18 - Selector channel 1C mask bit
19 - Selector channel 2C mask bit

Diagram 5-603. Set System Mask, SSM (80)

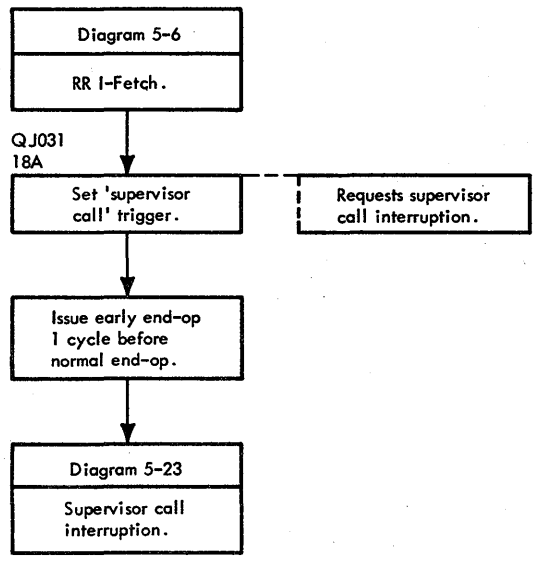
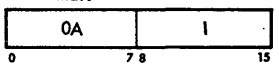


Diagram 5-604. Supervisor Call, SVC (0A)

- RR Format:



- Cause supervisor call interruption; replace old-PSW(24-31) with I-field (bits 8-15) of instruction, providing interruption code.
- Conditions at start of execution:
  1. Instruction is in E.
  2. E(8-15) contains interruption code.

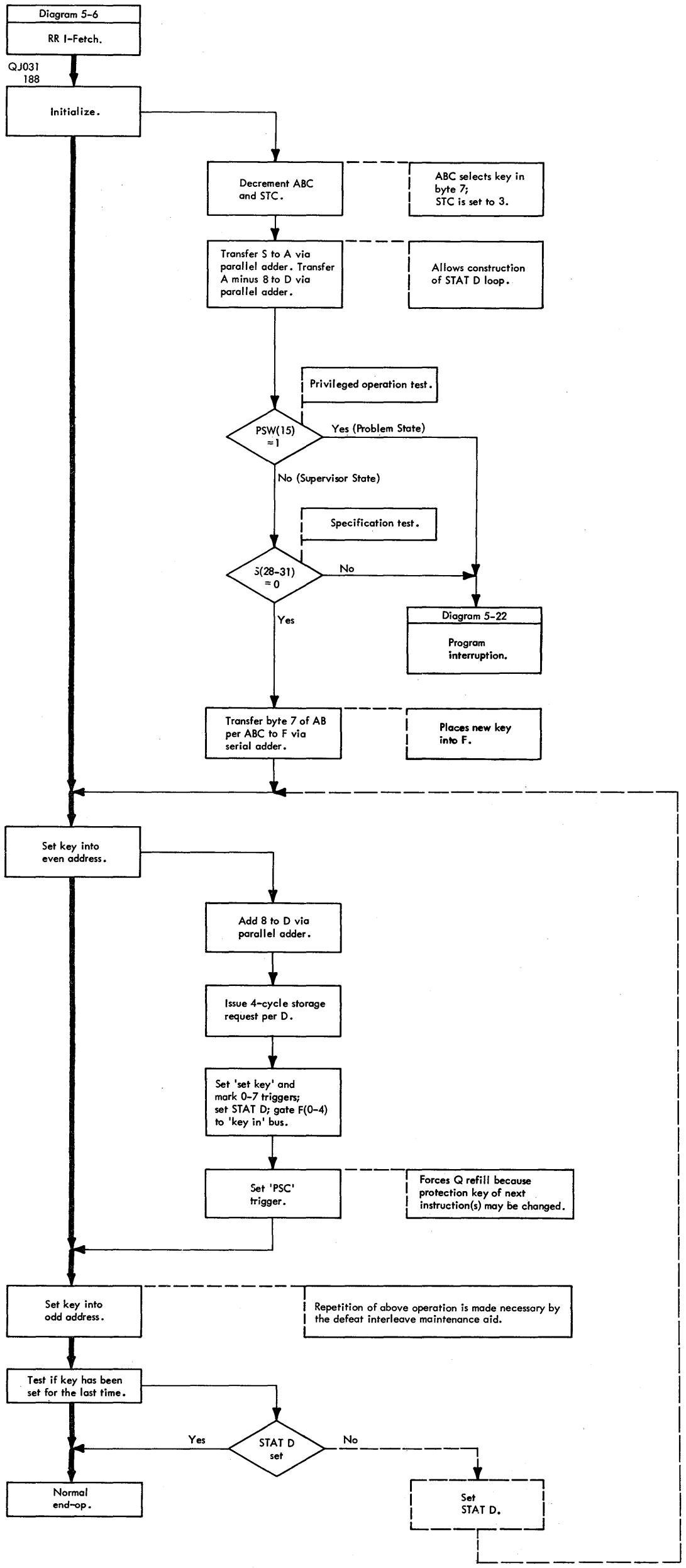
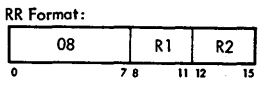
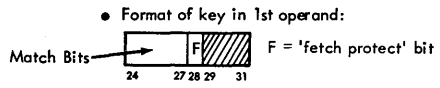


Diagram 5-605. Set Storage Key, SSK (08)

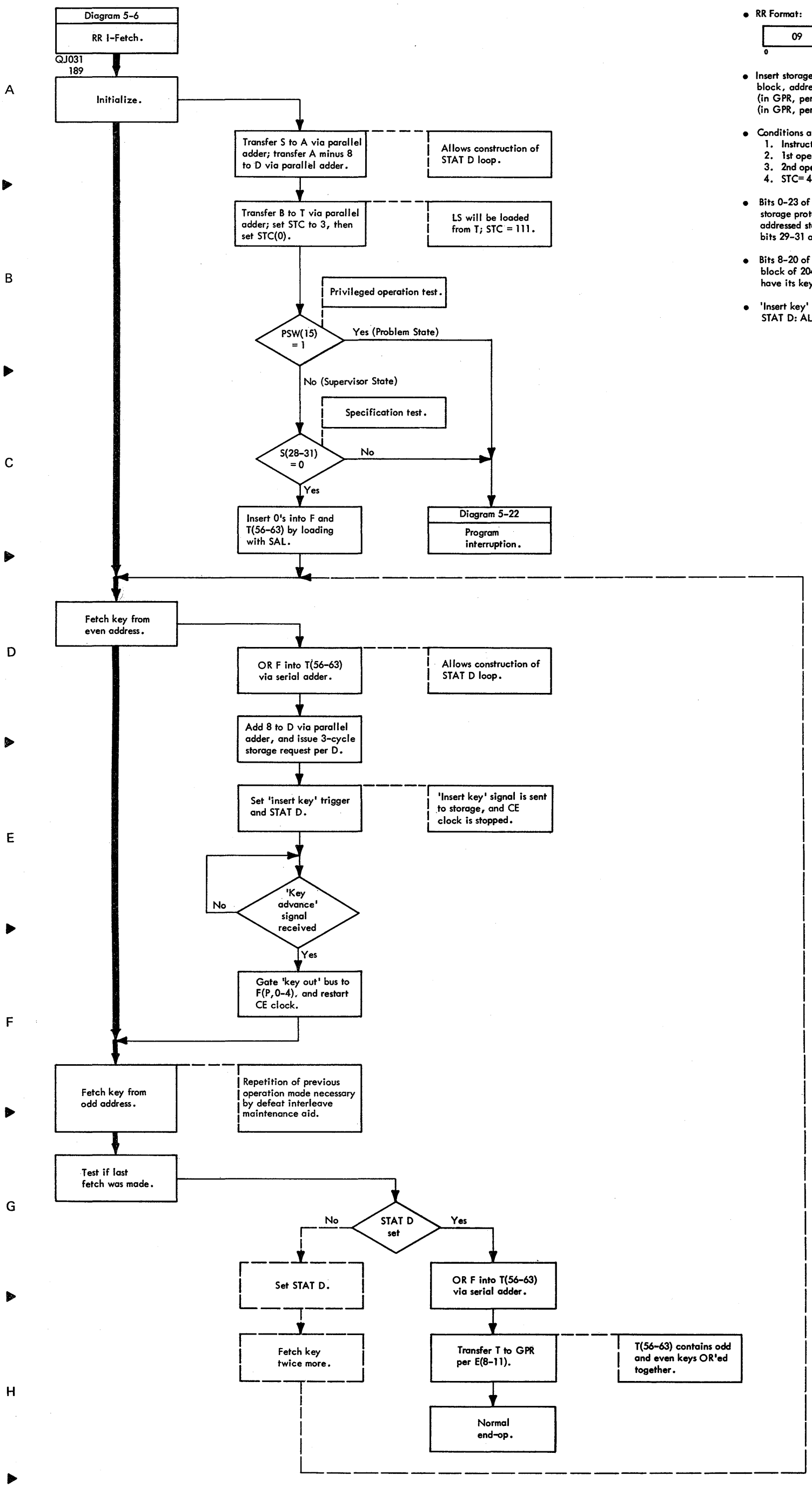
- RR Format:



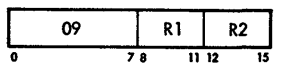
- Set storage key (bits 24–28 of 1st GPR, per R1) for 2048-byte storage block (addressed by bits 8–20 of 2nd operand, in GPR per R2) into storage protection logic in main storage.
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D.
  3. 2nd operand is in S and T.
  4. STC = 4 and ABC = 0.
- Format of key in 1st operand:



- Bits 8–20 of 2nd operand designate which block of 2048 bytes in main storage is to have its key set.
- 'Set key' trigger: ALD MC181; STAT D: ALD KS051.

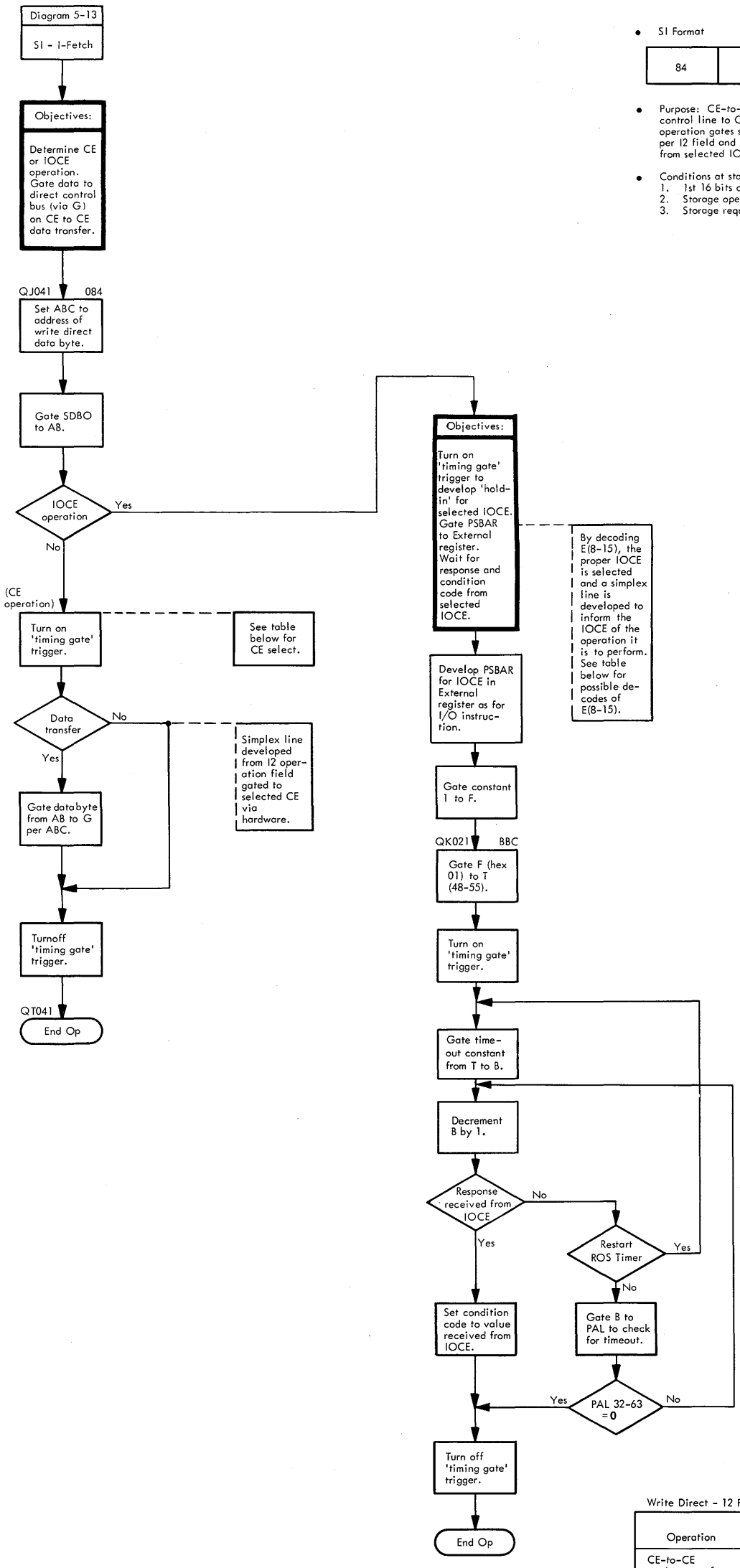


- RR Format:



- Insert storage protection key for 2048-byte storage block, addressed by bits 8-20 of 2nd operand (in GPR, per R2), into bits 24-28 of 1st operand (in GPR, per R1).
- Conditions at start of execution:
  1. Instruction is in E.
  2. 1st operand is in A, B, and D.
  3. 2nd operand is in S and T.
  4. STC= 4 and ABC = 0.
- Bits 0-23 of 1st operand are not changed; storage protection key from block of addressed storage is inserted into bits 24-28; bits 29-31 are set to 0's.
- Bits 8-20 of 2nd operand designate which block of 2048 bytes in main storage is to have its key inserted into 1st operand.
- 'Insert key' trigger: ALD MC 181; STAT D: ALD KS051.

Diagram 5-606. Insert Storage Key, ISK (09)



- SI Format

84	I2	B1	D1

- Purpose: CE-to-CE operation gates data byte or simplex control line to CE selected per I2 field. CE-to-IOCE operation gates simplex control line to IOCE selected per I2 field and sets condition code to value received from selected IOCE.

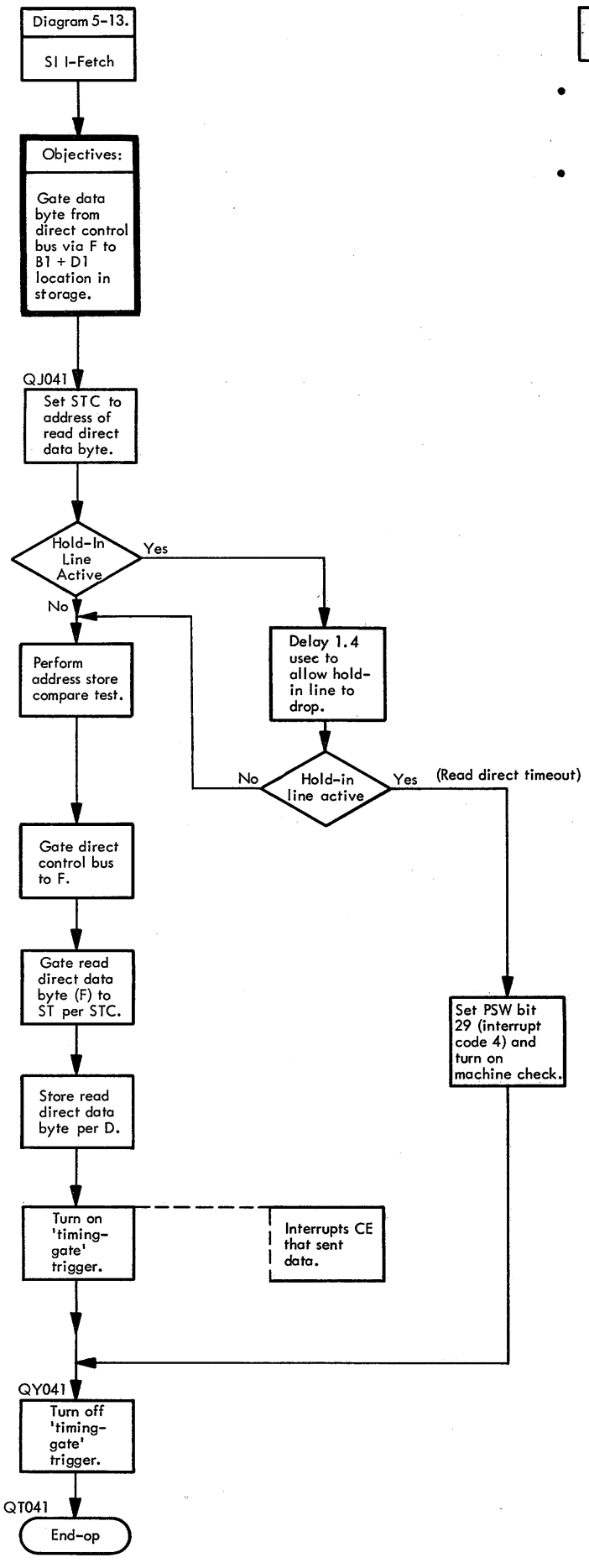
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. Storage operand address in D.
  3. Storage request has been issued per D.

Write Direct - 12 Field - E(8-15) Decode

Operation	Operation 8	9	10	11	Element Selected 12	13	14	15
CE-to-CE data transfer	0	0	0	0	CE 1	CE 2	CE 3	CE 4
CE external start	0	0	0	1	CE 1	CE 2	CE 3	CE 4
CE logout	0	0	1	0	CE 1	CE 2	CE 3	CE 4
IOCE logout	0	0	1	1	IOCE 1	IOCE 2	IOCE 3	0
CE external stop	0	1	0	0	CE 1	CE 2	CE 3	CE 4
IOCE processor start	0	1	0	1	IOCE 1	IOCE 2	IOCE 3	0
IOCE processor stop	0	1	1	0	IOCE 1	IOCE 2	IOCE 3	0
IOCE processor interrupt	0	1	1	1	IOCE 1	IOCE 2	IOCE 3	0

Note:
12 field decoding is done by hardware, and the select to the proper element is also hardware-developed.

Diagram 5-607. Write Direct, WRD (84)



- SI Format

85	I2	B1	D1

- Purpose: Gates a data byte from the control bus to the byte location (specified by the B1, D1 fields) in storage.

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. Storage operand address is in D.
  3. Storage request has been issued per D.

Diagram 5-608. Read Direct, RDD (85)

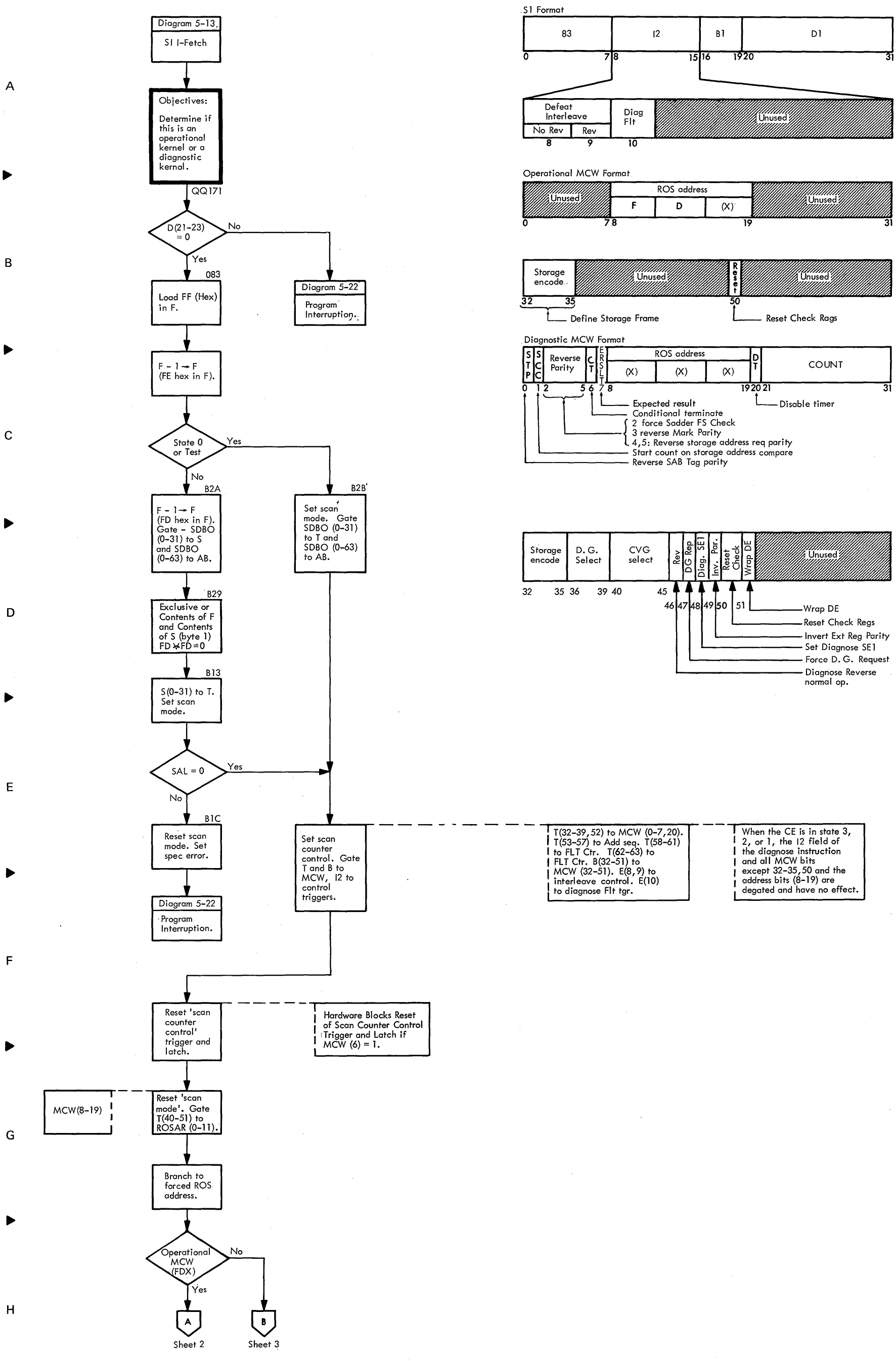
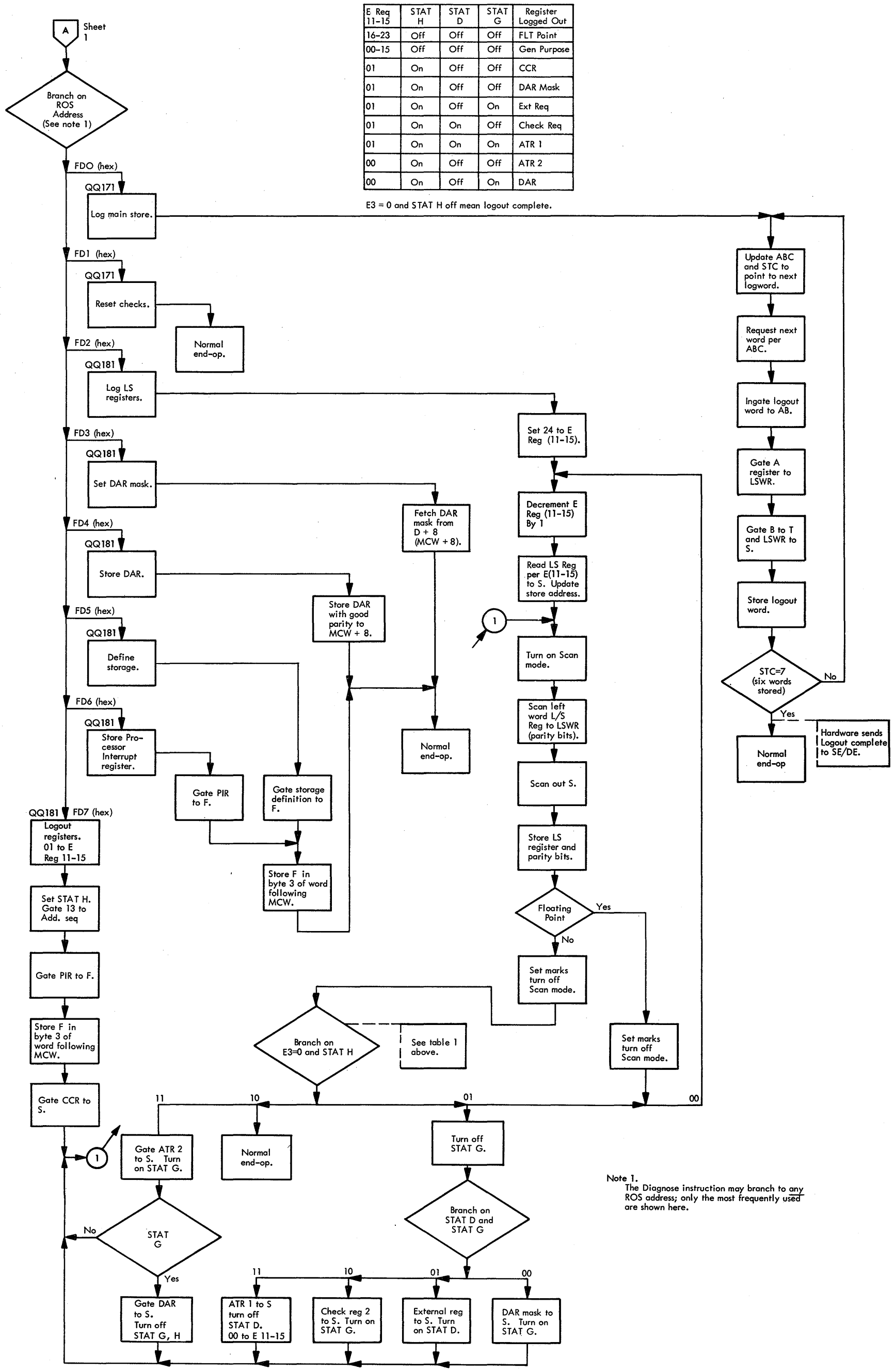


Diagram 5-609. Diagnose (83) (Sheet 1 of 3)

Table 1

E Req 11-15	STAT H	STAT D	STAT G	Register Logged Out
16-23	Off	Off	Off	FLT Point
00-15	Off	Off	Off	Gen Purpose
01	On	Off	Off	CCR
01	On	Off	Off	DAR Mask
01	On	Off	On	Ext Req
01	On	On	Off	Check Req
01	On	On	On	ATR 1
00	On	Off	Off	ATR 2
00	On	Off	On	DAR

E3 = 0 and STAT H off mean logout complete.



Note 1.
The Diagnose instruction may branch to any ROS address; only the most frequently used are shown here.

Diagram 5-609. Diagnose (83) (Sheet 2 of 3)

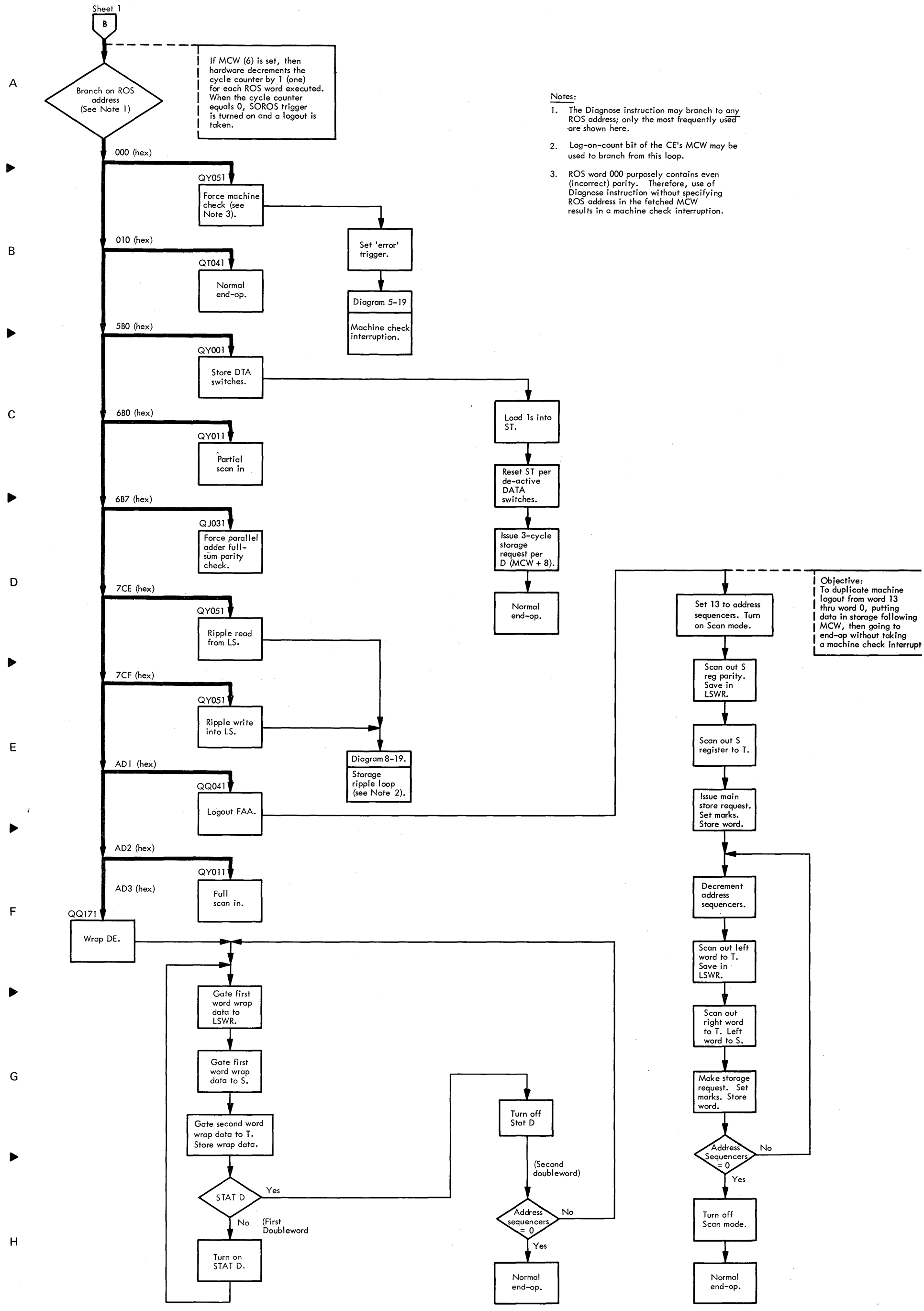
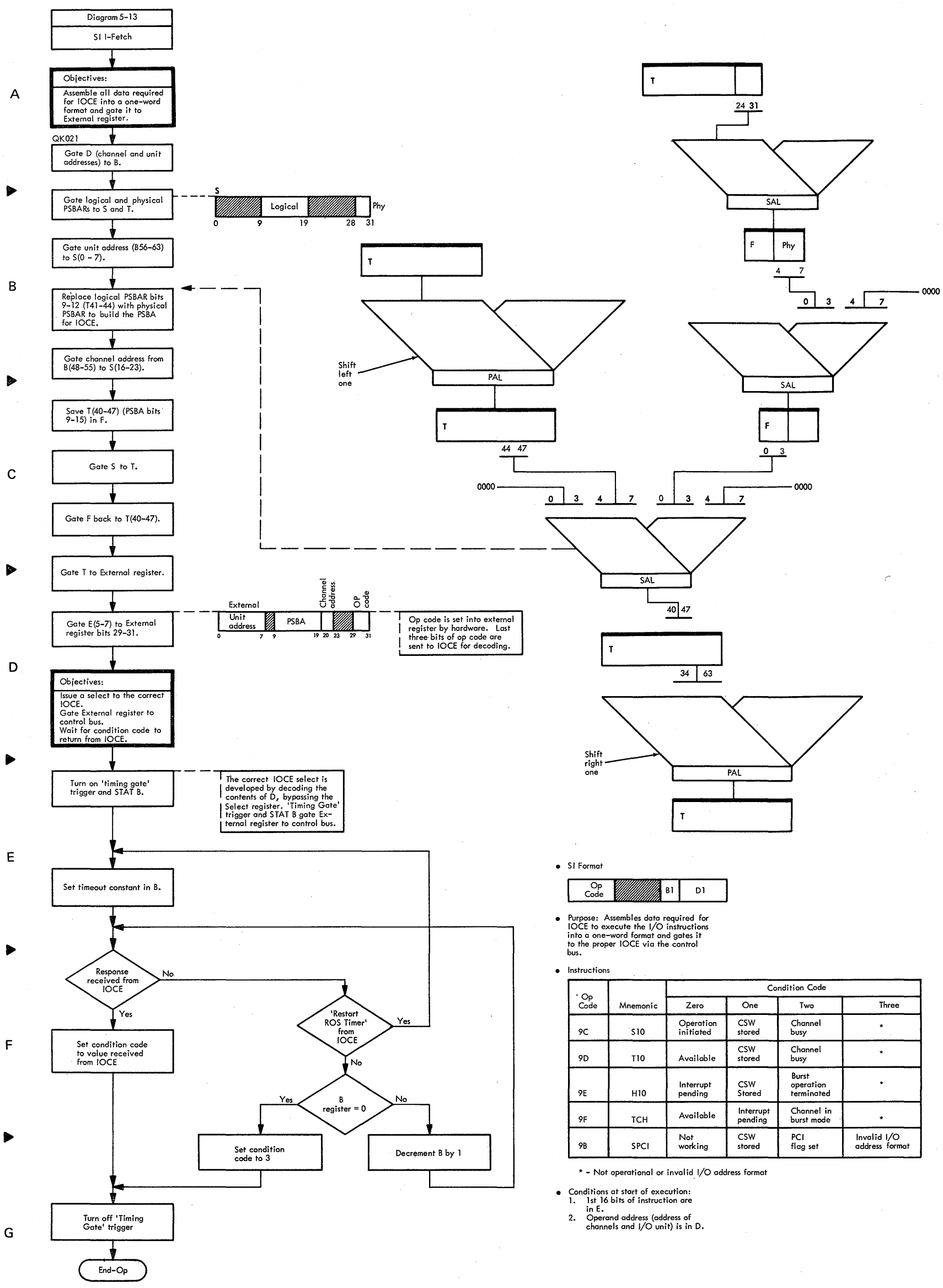


Diagram 5-609. Diagnose (83) (Sheet 3 of 3)



- SI Format

Op Code		B1	D1

- Purpose: Assembles data required for IOCE to execute the I/O instructions into a one-word format and gates it to the proper IOCE via the control bus.
- Instructions

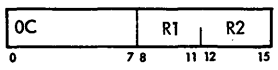
Op Code	Mnemonic	Condition Code			
		Zero	One	Two	Three
9C	S10	Operation initiated	CSW stored	Channel busy	*
9D	T10	Available	CSW stored	Channel busy	*
9E	H10	Interrupt pending	CSW Stored	Burst operation terminated	*
9F	TCH	Available	Interrupt pending	Channel in burst mode	*
9B	SPCI	Not working	CSW stored	PCI flag set	Invalid I/O address format

* - Not operational or invalid I/O address format

- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. Operand address (address of channels and I/O unit) is in D.

Diagram 5-701. I/O Instructions

- RR Format: L1



- Purpose: Loads identity of CE executing instruction into bits 28-31 of GPR specified by R1 field.

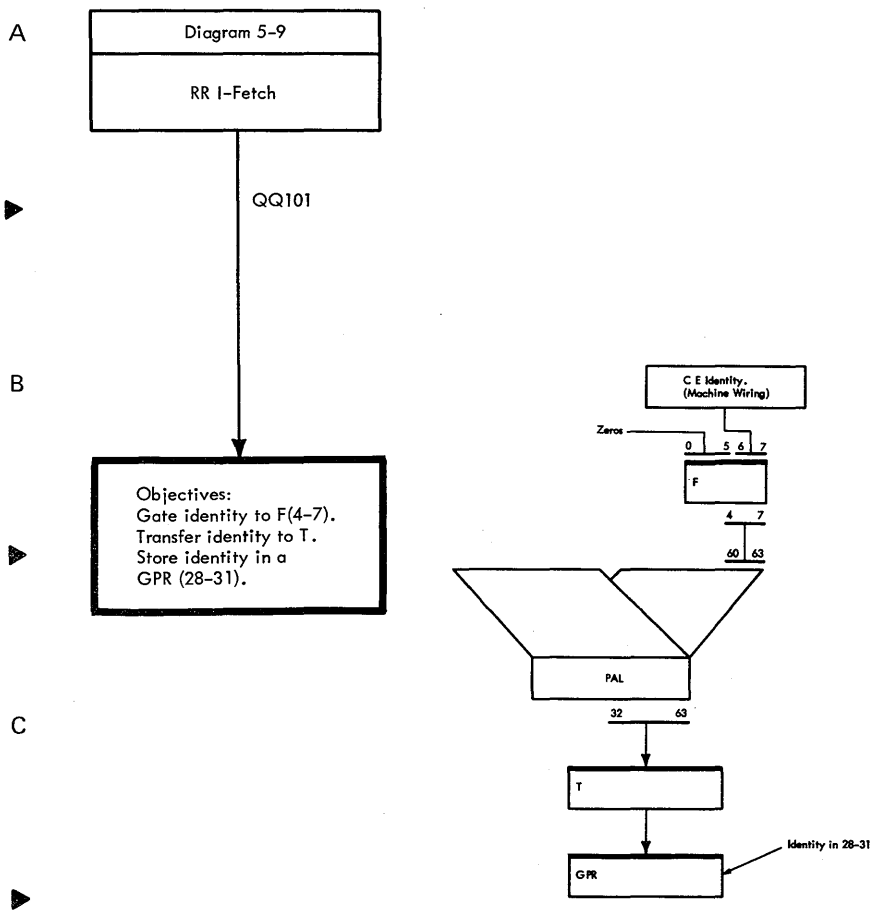
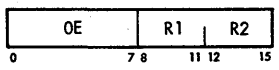


Diagram 5-801. Load Identity, LI (0C)

- RR Format



- Purpose: Places contents of ATR1 and ATR2 into two GPR's specified by R1 and R2 fields.

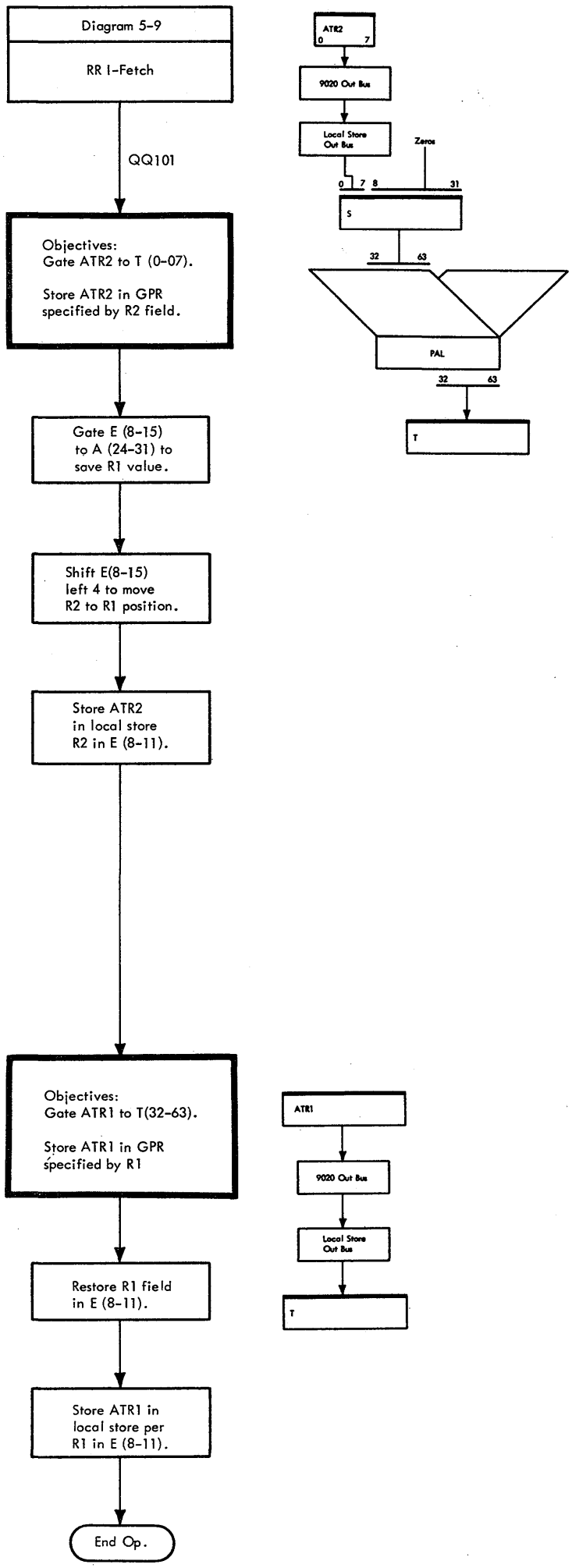
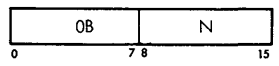


Diagram 5-802. Insert ATR, IATR (0E)

- RR Format



- Purpose: Provides a variable delay (256 usec x N) dependent on the value of N.
- Conditions at the beginning of execution: Instruction and N are in E.

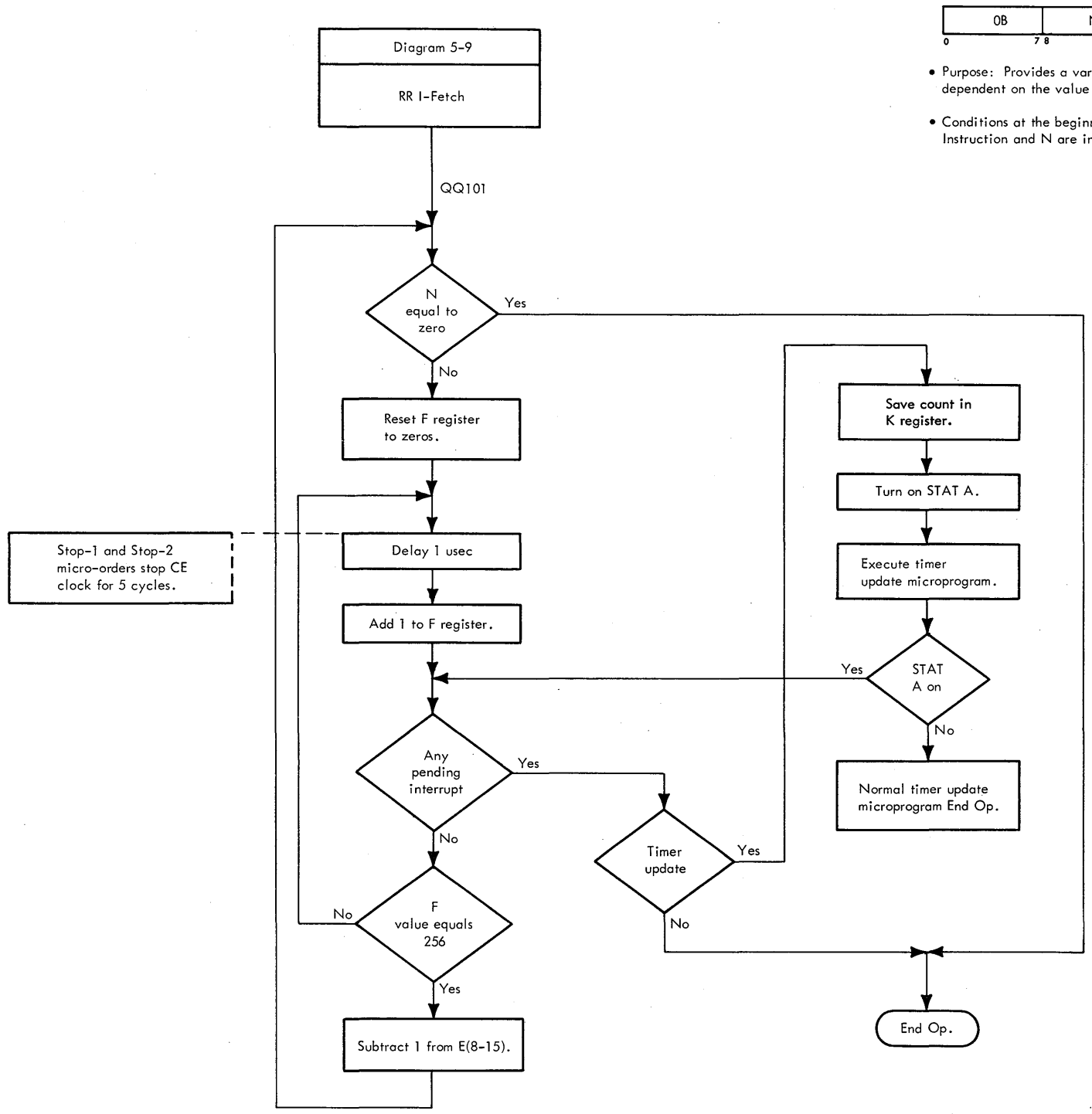
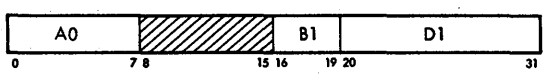


Diagram 5-803. Delay, DLY (0B)

- SI Format



- Purpose: Store contents of logical and physical PSBAR in bit positions 9 - 19 and 28 - 31, respectively, at the word location in main storage (specified by 1st operand address).

- Conditions at the beginning of execution:
  1. 1st 16 bits of instruction are in E.
  2. Operand address (address to store PSBA) is in D.

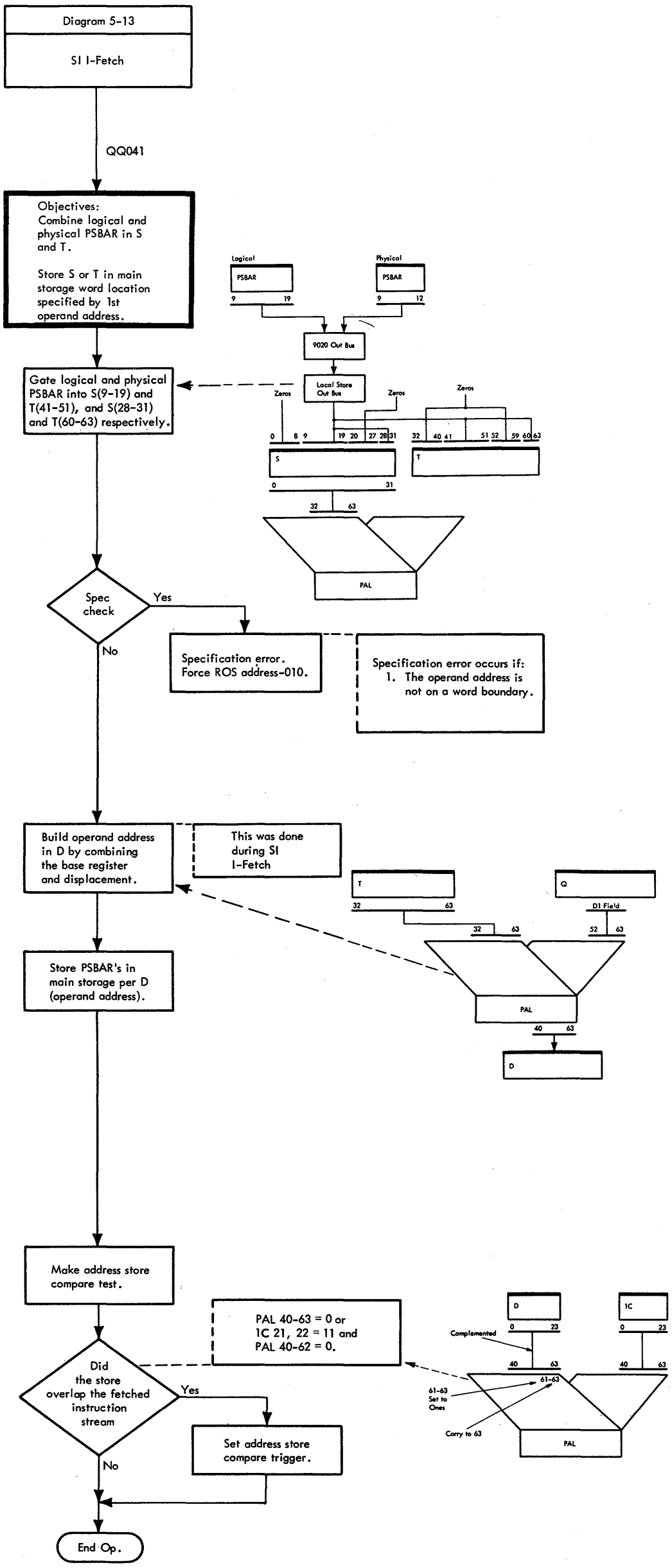
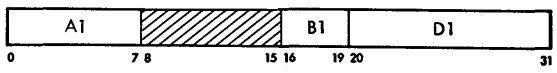


Diagram 5-804. Store PSBAR, SPSB (A0)

- SI Format



- Purpose: Load logical PSBAR from a storage location pointed to by the operand. Load physical PSBAR from an ATR slot (selected by bits 9-12 of the newly fetched logical PSBA).

- Conditions at beginning of execution:
  1. 1st 16 bits of instruction are in E.
  2. Operand address (address of new PSBA) is in D.

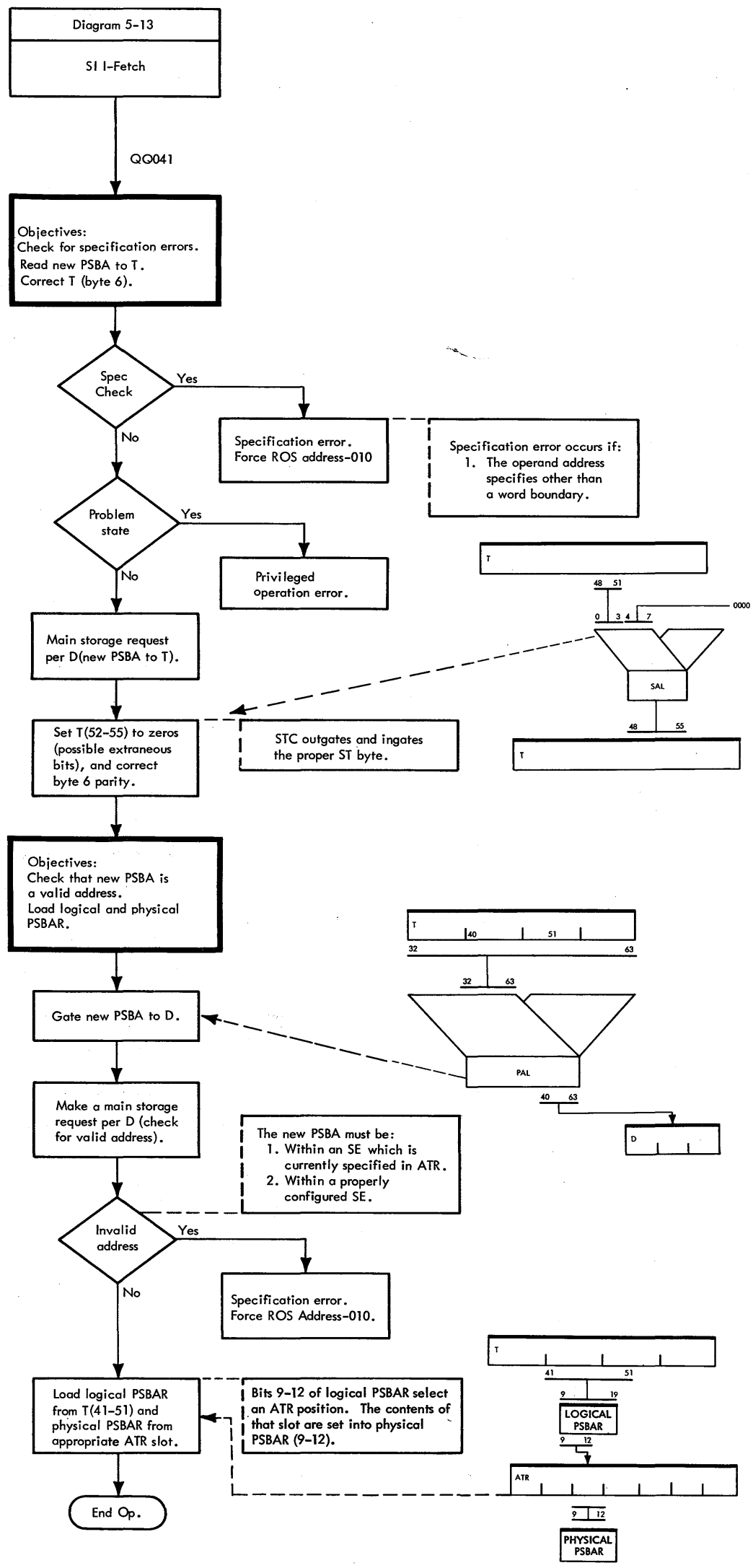
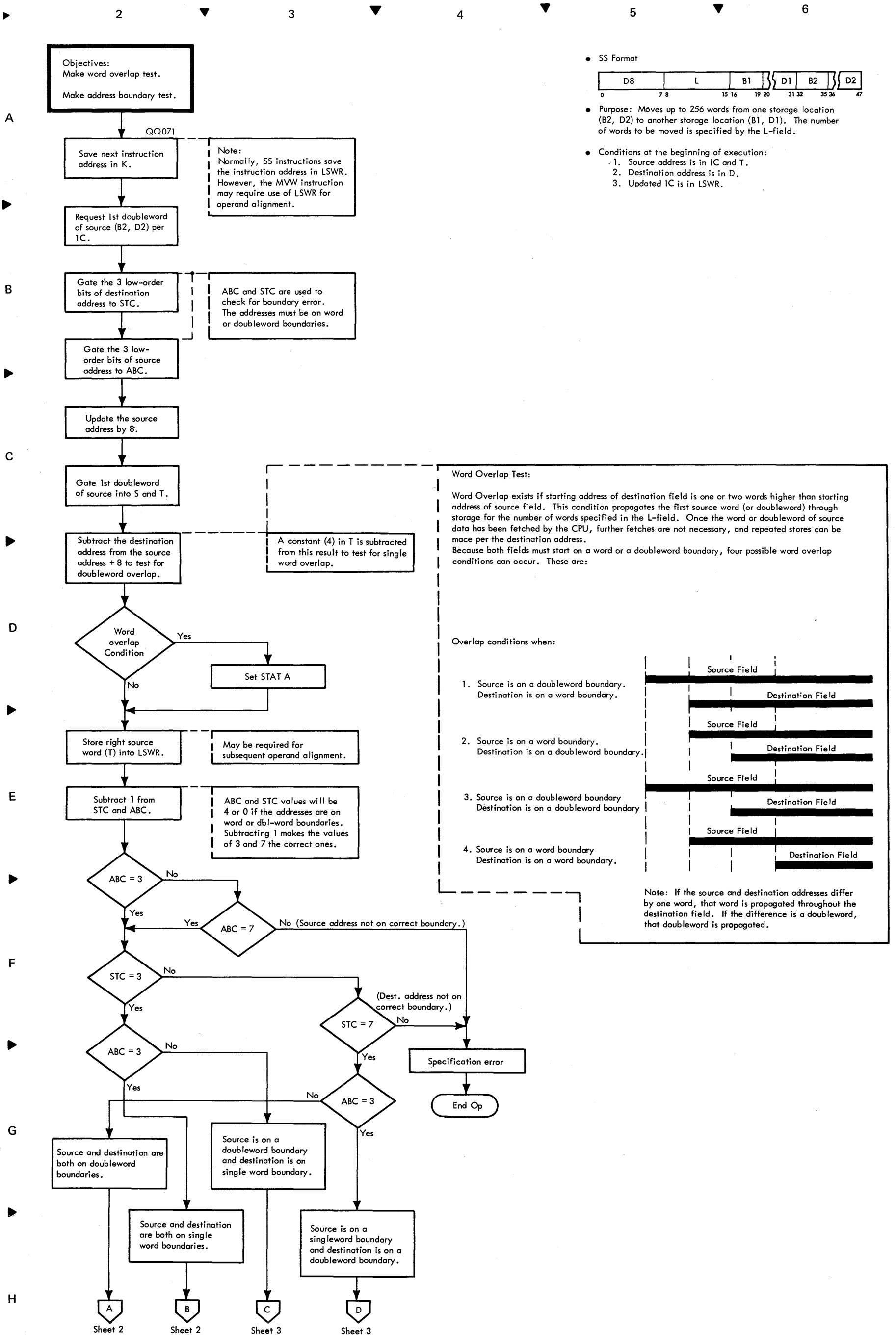


Diagram 5-805. Load PSBAR, LPSB (A1)



- SS Format

- Purpose: Moves up to 256 words from one storage location (B2, D2) to another storage location (B1, D1). The number of words to be moved is specified by the L-field.
- Conditions at the beginning of execution:
  1. Source address is in IC and T.
  2. Destination address is in D.
  3. Updated IC is in LSWR.

Diagram 5-806. Move Word, MVW (D8) (Sheet 1 of 3)

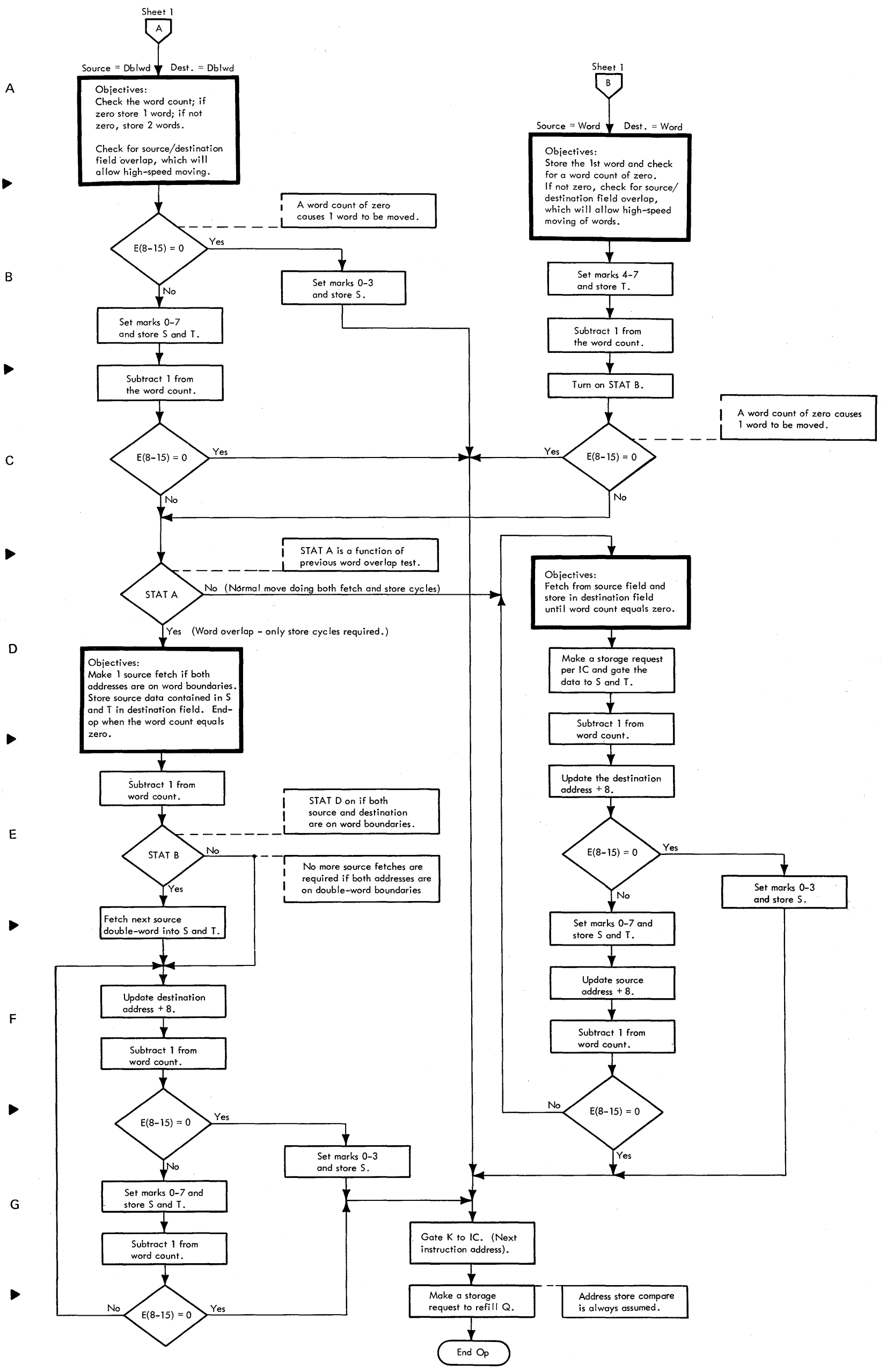


Diagram 5-806. Move Word, MVW (D8) (Sheet 2 of 3)

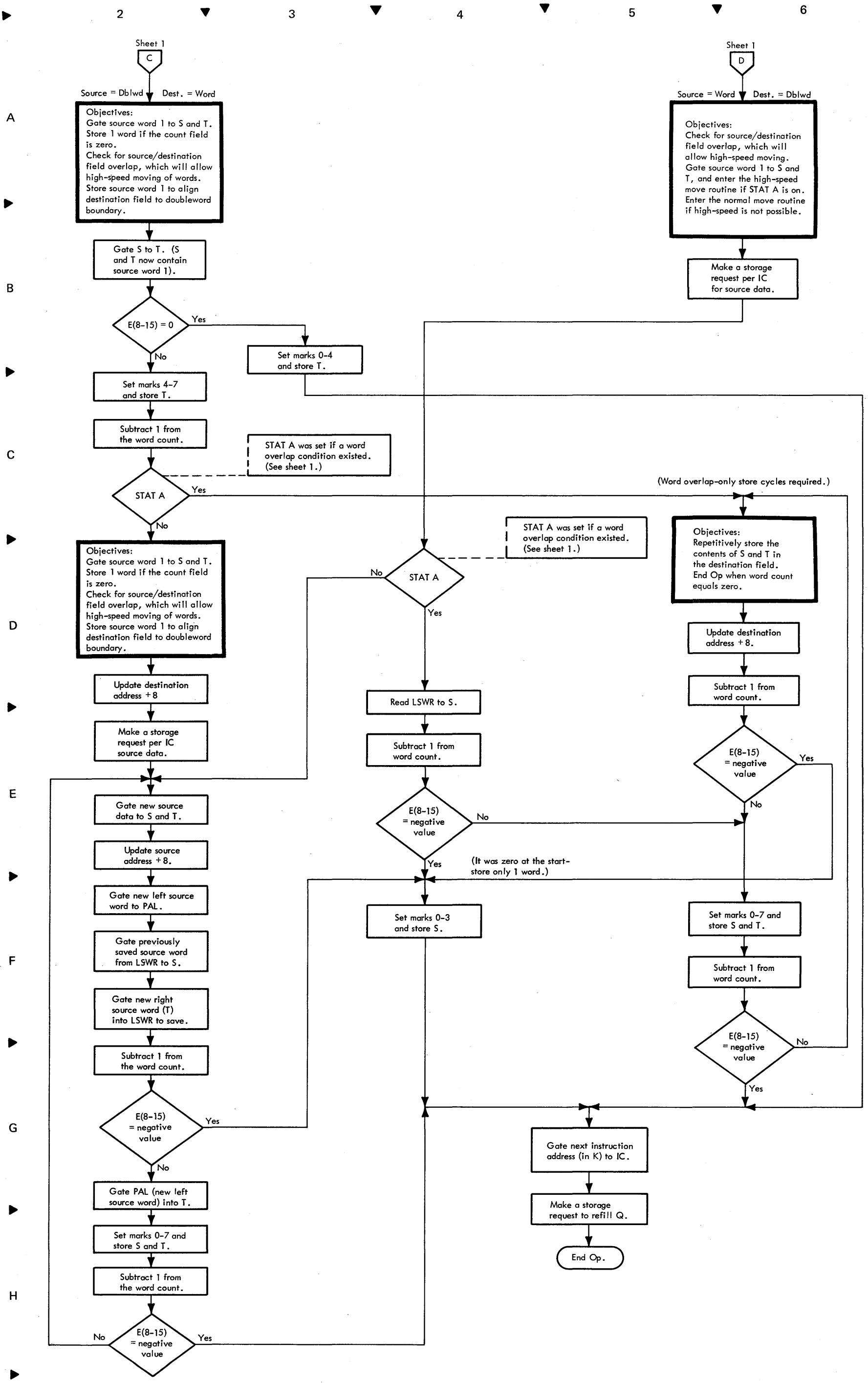
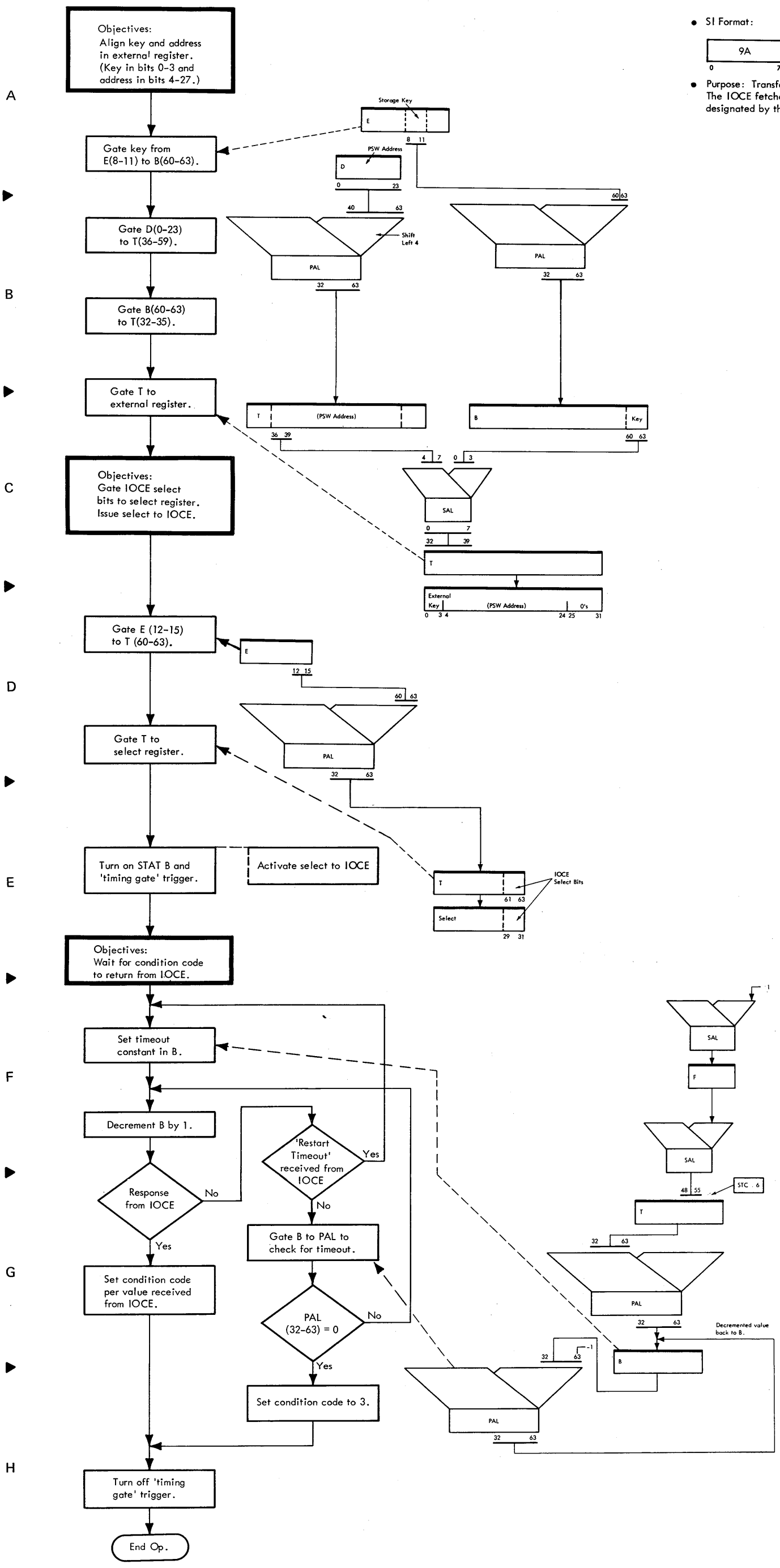
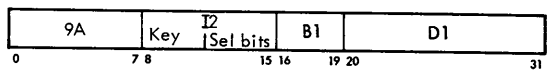


Diagram 5-806. Move Word, MVW (D8) (Sheet 3 of 3)

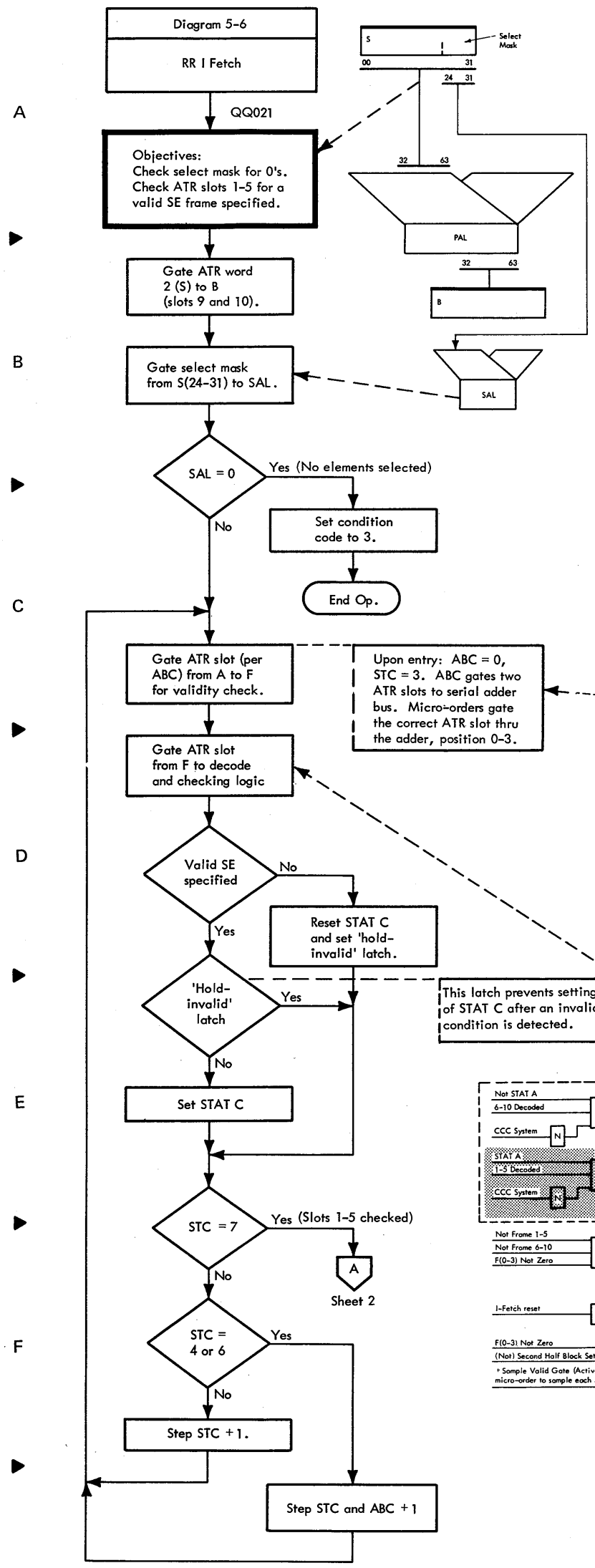


- SI Format:



- Purpose: Transfers storage key and PSW address to selected IOCE. The IOCE fetches the new PSW and proceeds with the operation designated by the PSW.

Diagram 5-807. Start I/O Processor, SIOP (9A)



- RR Format

0	D	R1	R2

- Purpose: Loads the address translation register: (0-31) from the GPR specified by R1, and (32-39) from byte 0 of the GPR specified by R2. Byte 3 of the GPR specified by R2 is the selection mask, and the response bits are set into byte 2.

- Format of GPR specified by:
  1. R1

SE	SE	SE	SE	SE	SE or DE *	SE or DE *	SE or DE *

  2. R2

SE or DE *	SE or DE *	Ignored	Response Bits **	Selection Mask **

* = 9020 E only

** =

Response Bits	Selection Mask	CE or IOCE Designated
16	24	CE 1
17	25	CE 2
18	26	CE 3
19	27	CE 4
20	28	not used
21	29	IOCE 1
22	30	IOCE 2
23	31	IOCE 3

- Conditions at the start of execution:
  1. Instruction is in E.
  2. ATR word 1 is in A and B
  3. ATR word 2 is in S and T.

Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 1 of 6)

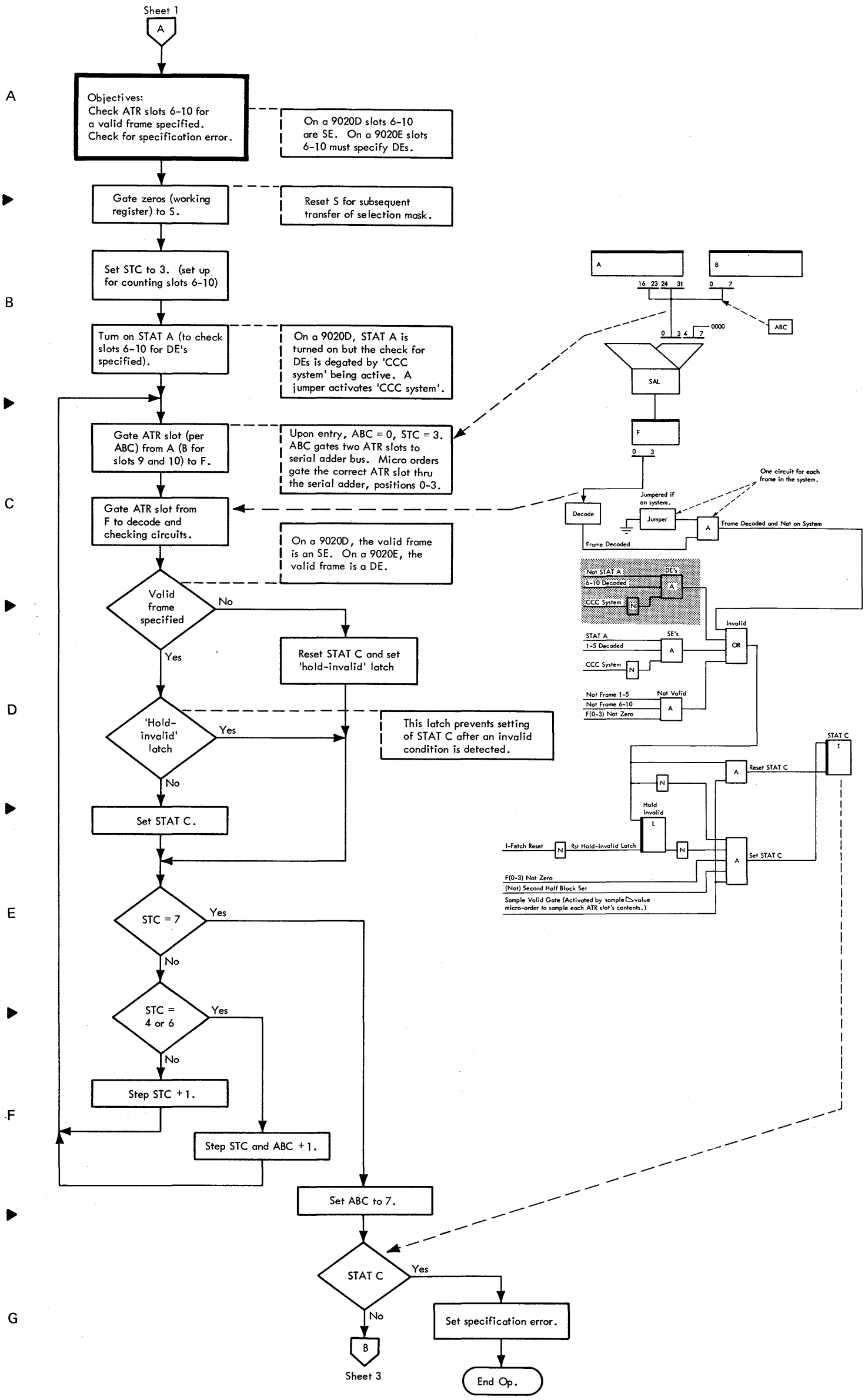


Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 2 of 6)

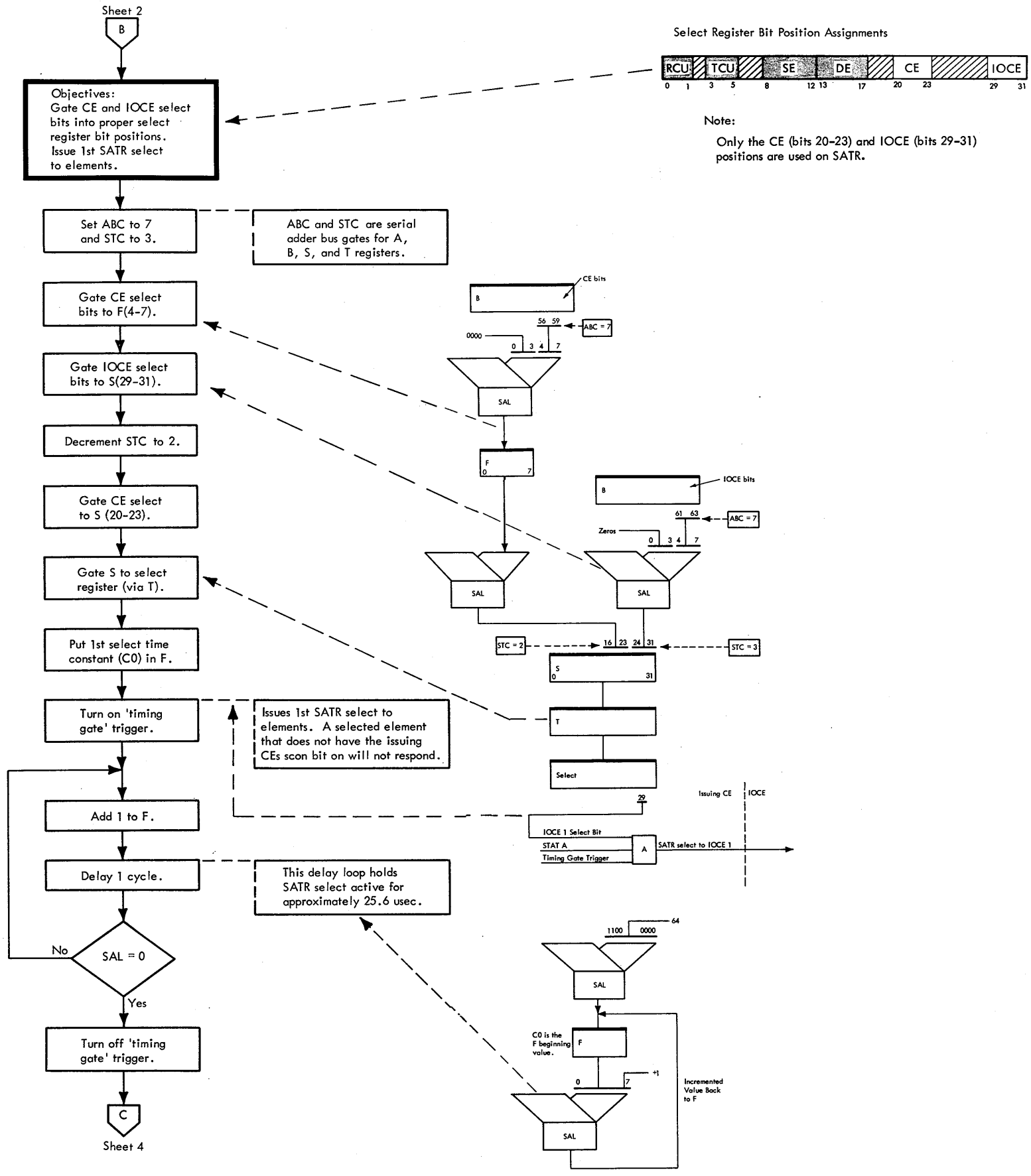


Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 3 of 6)

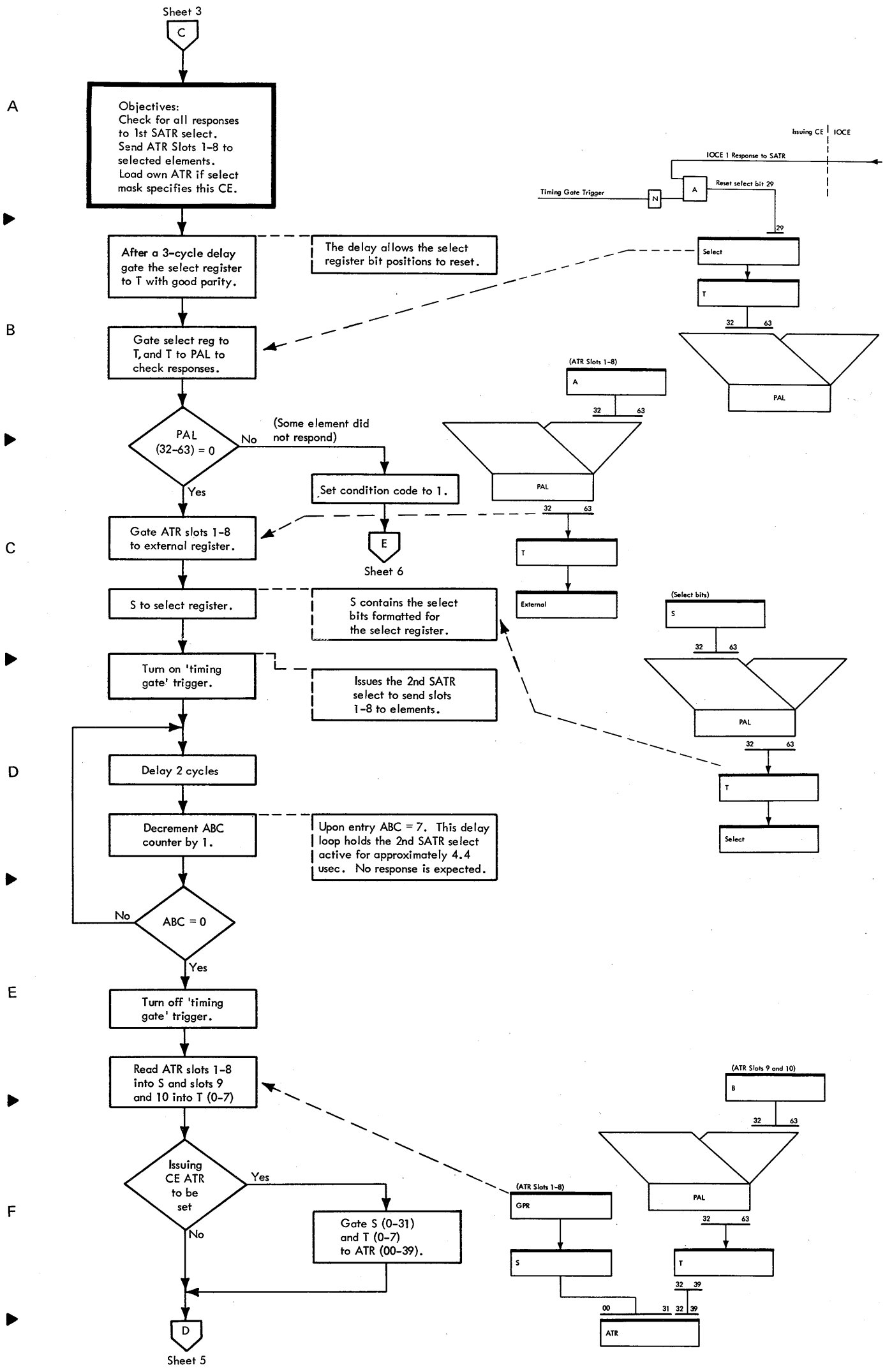


Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 4 of 6)

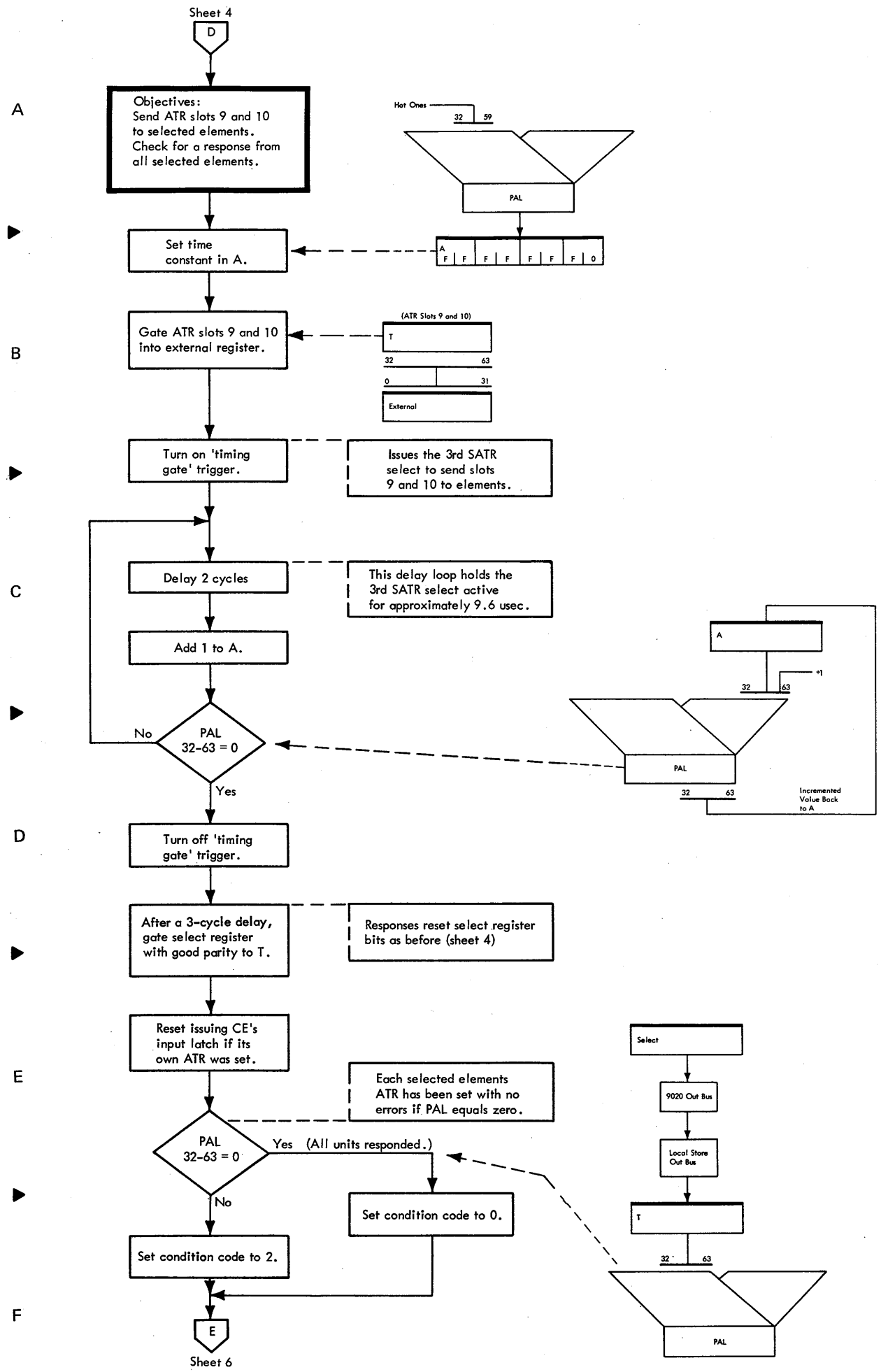
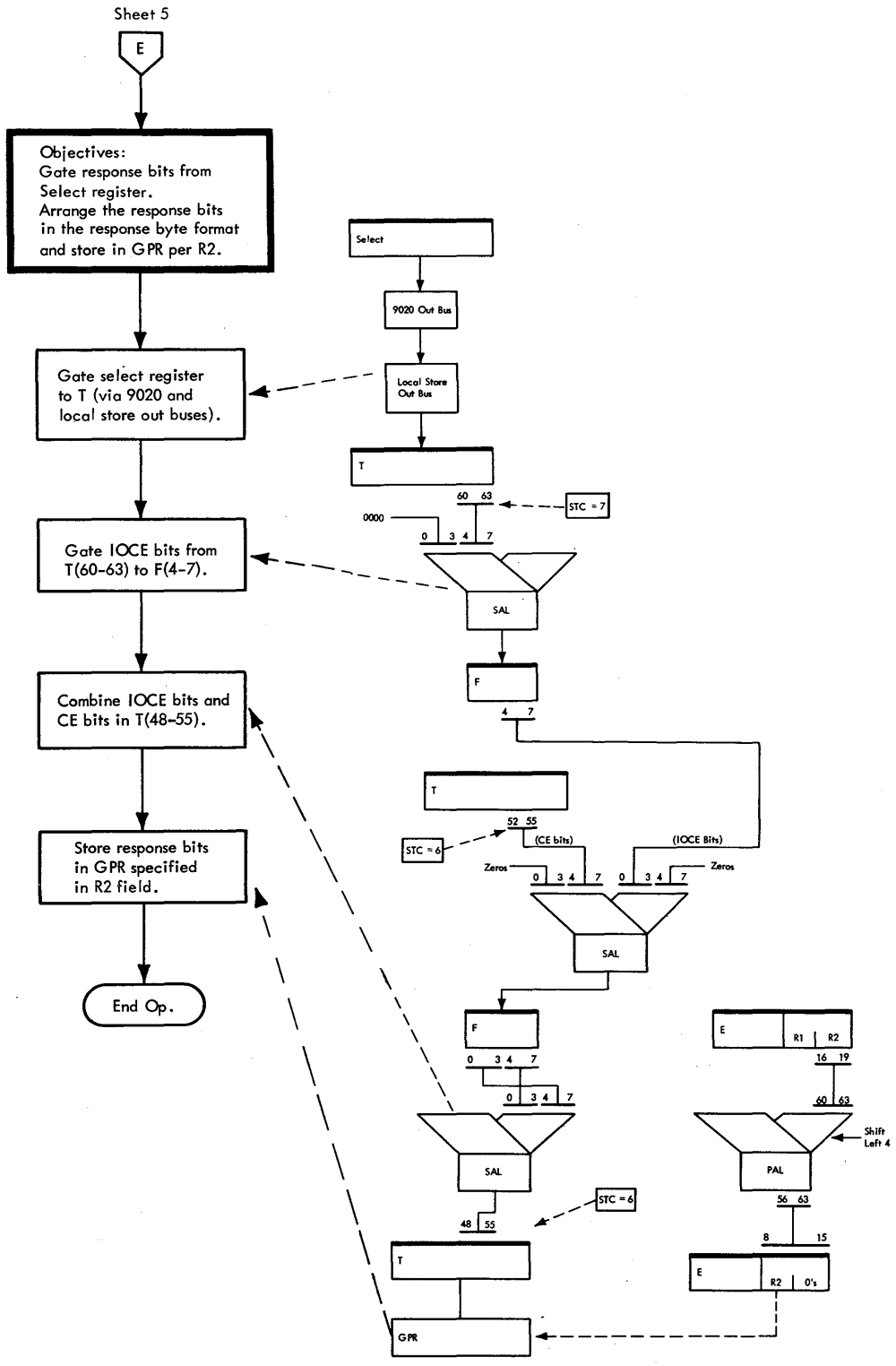
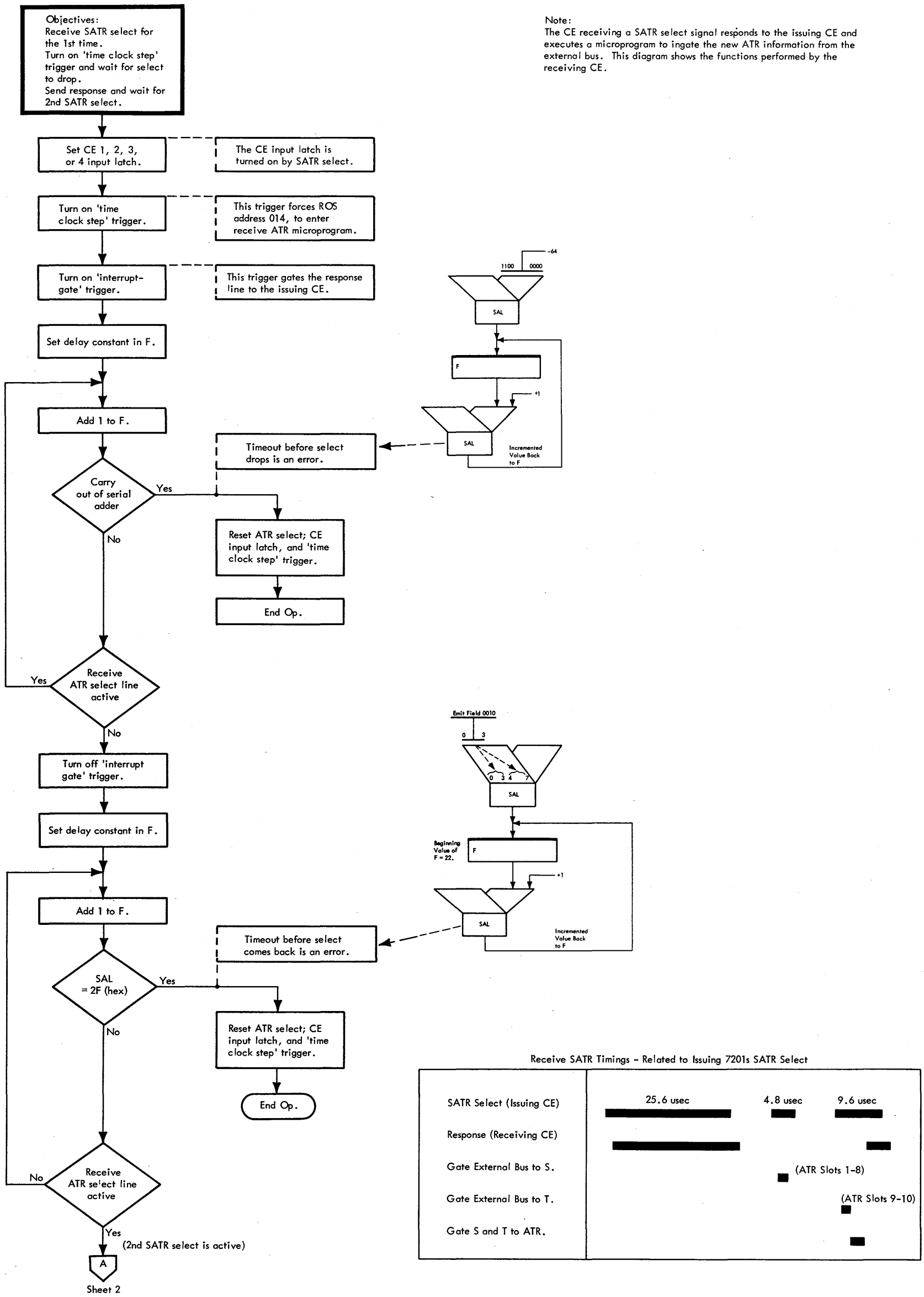


Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 5 of 6)



Note:
A CE receiving a SATR select signal from another CE must execute a microprogram to ingate the new ATR information. Refer to Diagram 9, MDM for explanation of the SATR instruction in the receiving CE.

Diagram 5-808. Set Address Translator, SATR (0D), Execution in Issuing CE (Sheet 6 of 6)



Note:
The CE receiving a SATR select signal responds to the issuing CE and executes a microprogram to ingate the new ATR information from the external bus. This diagram shows the functions performed by the receiving CE.

Diagram 5-809. Set Address Translator, SATR (0D), Execution in Receiving CE (Sheet 1 of 3)

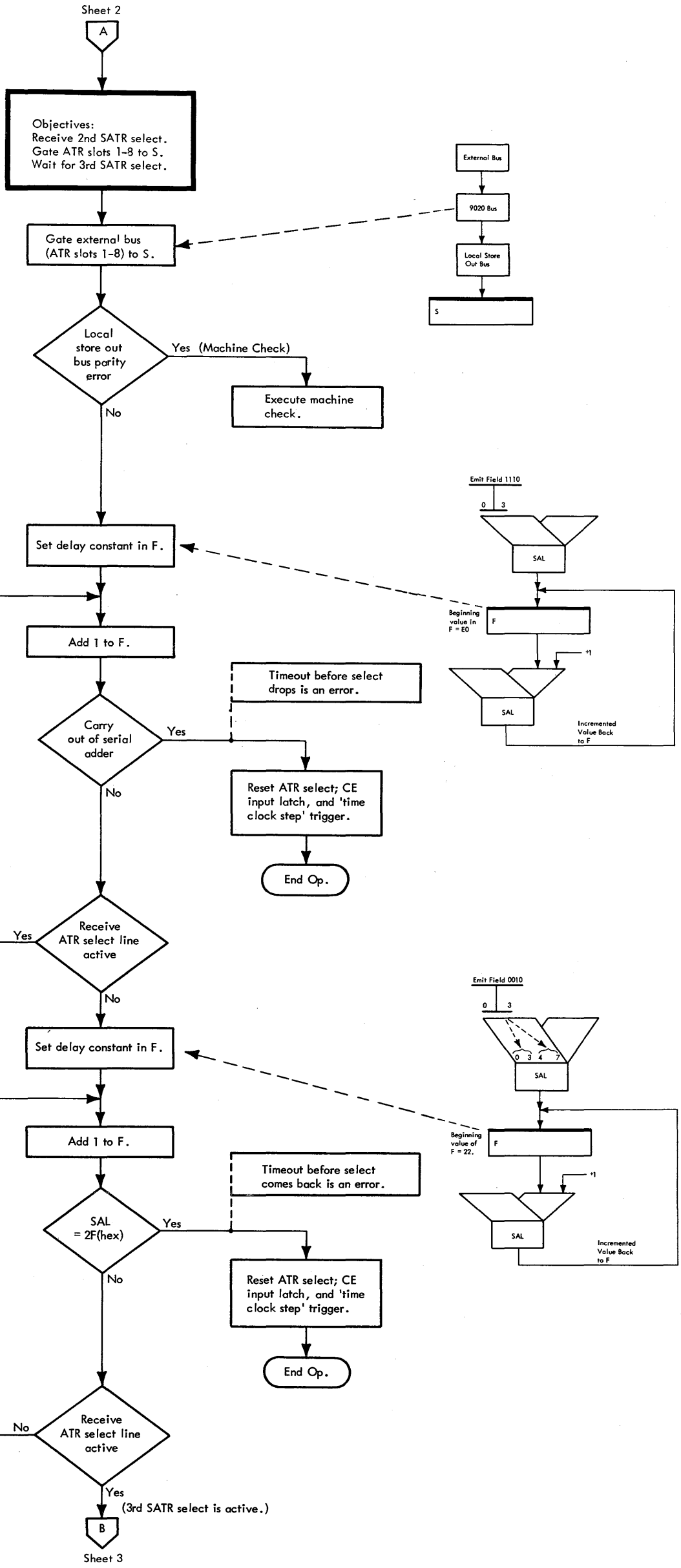


Diagram 5-809. Set Address Translator, SATR (0D), Execution in Receiving CE (Sheet 2 of 3)

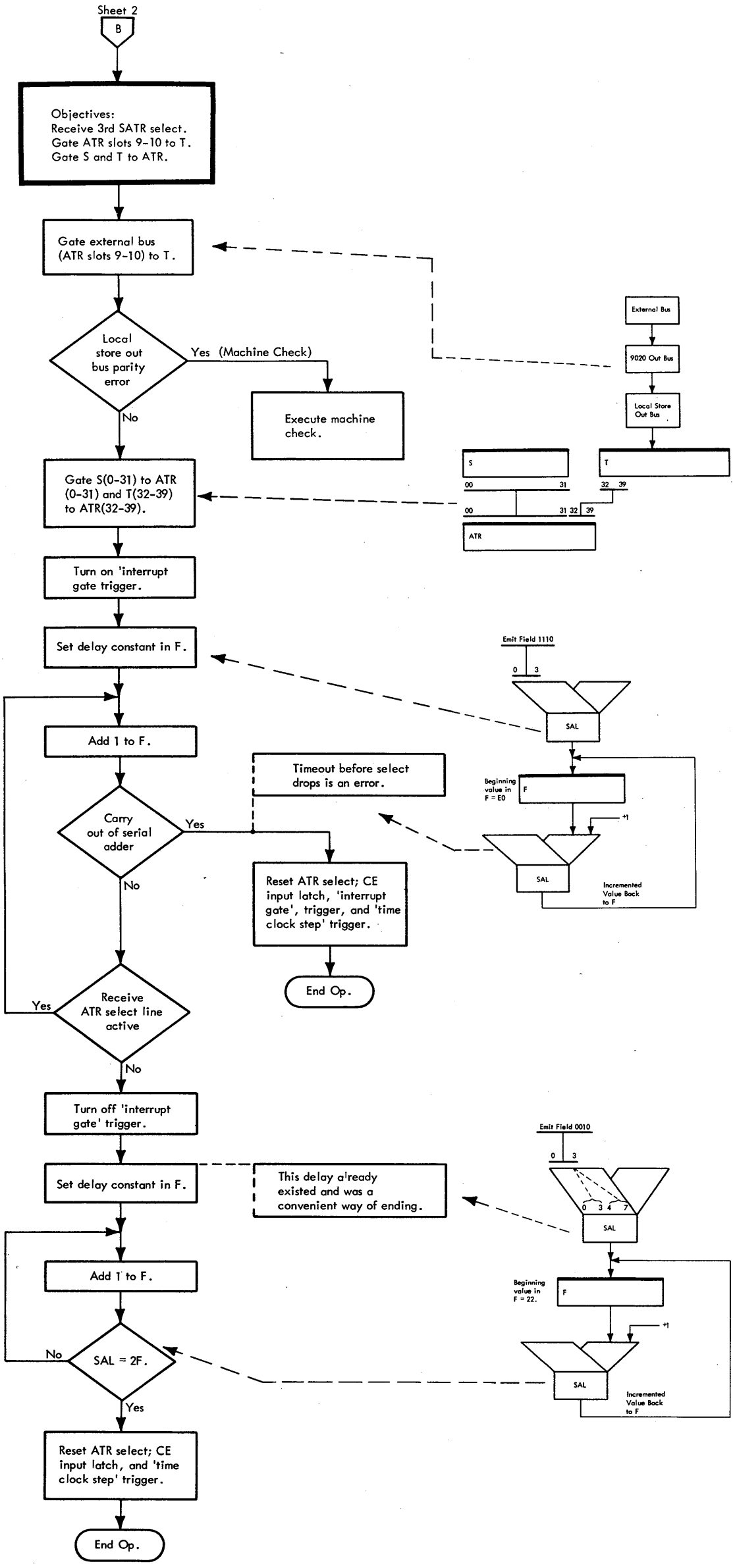
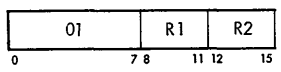


Diagram 5-809. Set Address Translator, SATR (0D), Execution in Receiving CE (Sheet 3 of 3)

- RR Format



- Purpose: To transfer configuration mask to CCR of element(s) specified by selection mask.

- Conditions at the end of I-Fetch:
  1. A-reg has 1st word of configuration mask (contents of GPR specified by R1).
  2. T-reg has selection mask (contents of GPR specified by R2).

- Configuration and selection mask formats:
  1. 9020D System

Configuration Mask

1st Word

State		Scon						SE												CE								IOCE		
S1	S2	1	2	3	-	‡	*	1	2	3	4	5	6	7	8	-	-	*	*	1	2	3	-	*	*	*	*	*	1	2	3
0	1	2			5	6	7	8									17	18	19	20			23	24				28	29		31

2nd Word

(Not Used)
0 31

Selection Mask

PAM			TCU					SE												CE								IOCE		
1	2	3	1	2	3	*	*	1	2	3	4	5	6	7	8	-	-	*	*	1	2	3	-	*	*	*	*	*	1	2	3
0		2	3		5	6	7	8									17	18	19	20			23	24				28	29		31

  2. 9020E System

Configuration Mask

1st Word

State		Scon						SE					DE							CE								IOCE		
S1	S2	1	2	3	-	‡	‡	1	2	3	4	-	1	2	3	4	-	*	*	1	2	3	-	*	*	*	*	*	1	2	3
0	1	2			5	6	7	8				12	13				17	18	19	20			23	24				28	29		31

2nd Word

DG																(Not Used)
1A	1B	2B	2C	3C	3D	4D	4A	5A	5B	6B	6C	7C	7D	8D	8A	
0															15	16 31

Selection Mask

RCU			TCU					SE					DE							CE								IOCE		
1	2	*	1	2	-	*	*	1	2	3	4	-	1	2	3	4	-	*	*	1	2	3	-	*	*	*	*	*	1	2	-
0	1	2	3		5	6	7	8				12	13				17	18	19	20			23	24				28	29		31

Legend:
- * Spare Bit
- \- Reserved Bit
- ‡ Inhibit Logout Stop Bit
- ⁜ Inhibit DE Stop Bit

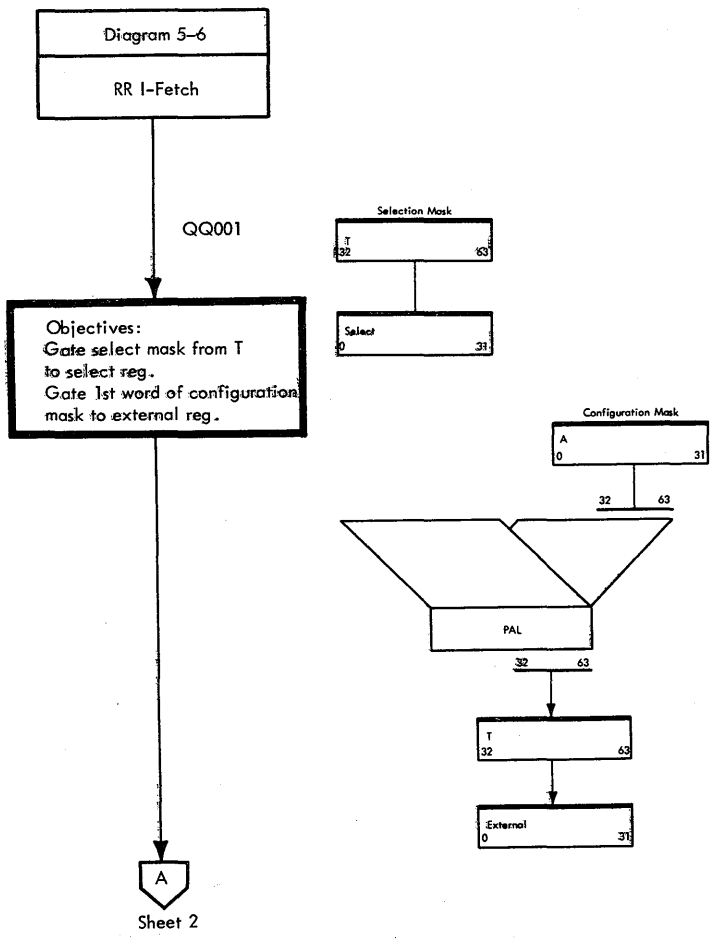


Diagram 5-810. Set Configuration, SCON (01) (Sheet 1 of 6)

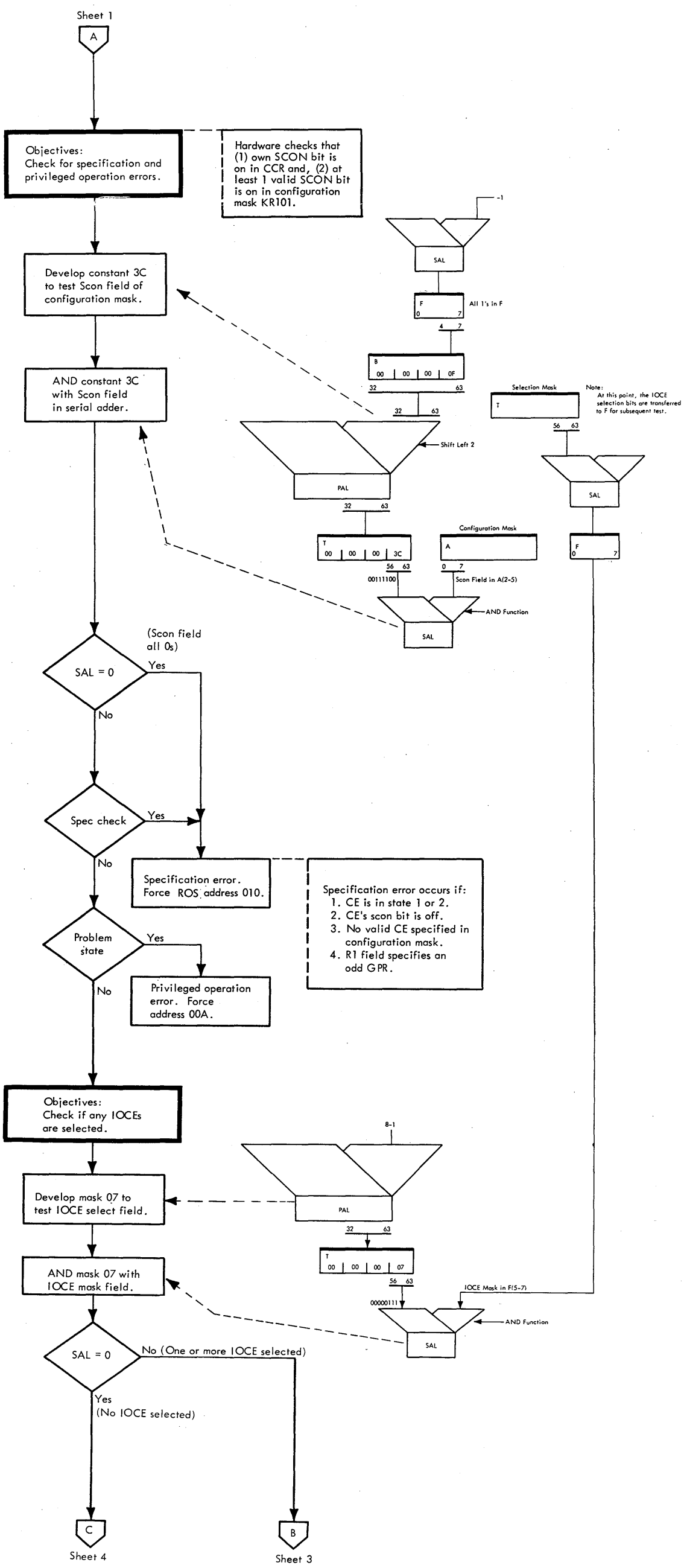


Diagram 5-810. Set Configuration, SCON (01) (Sheet 2 of 6)

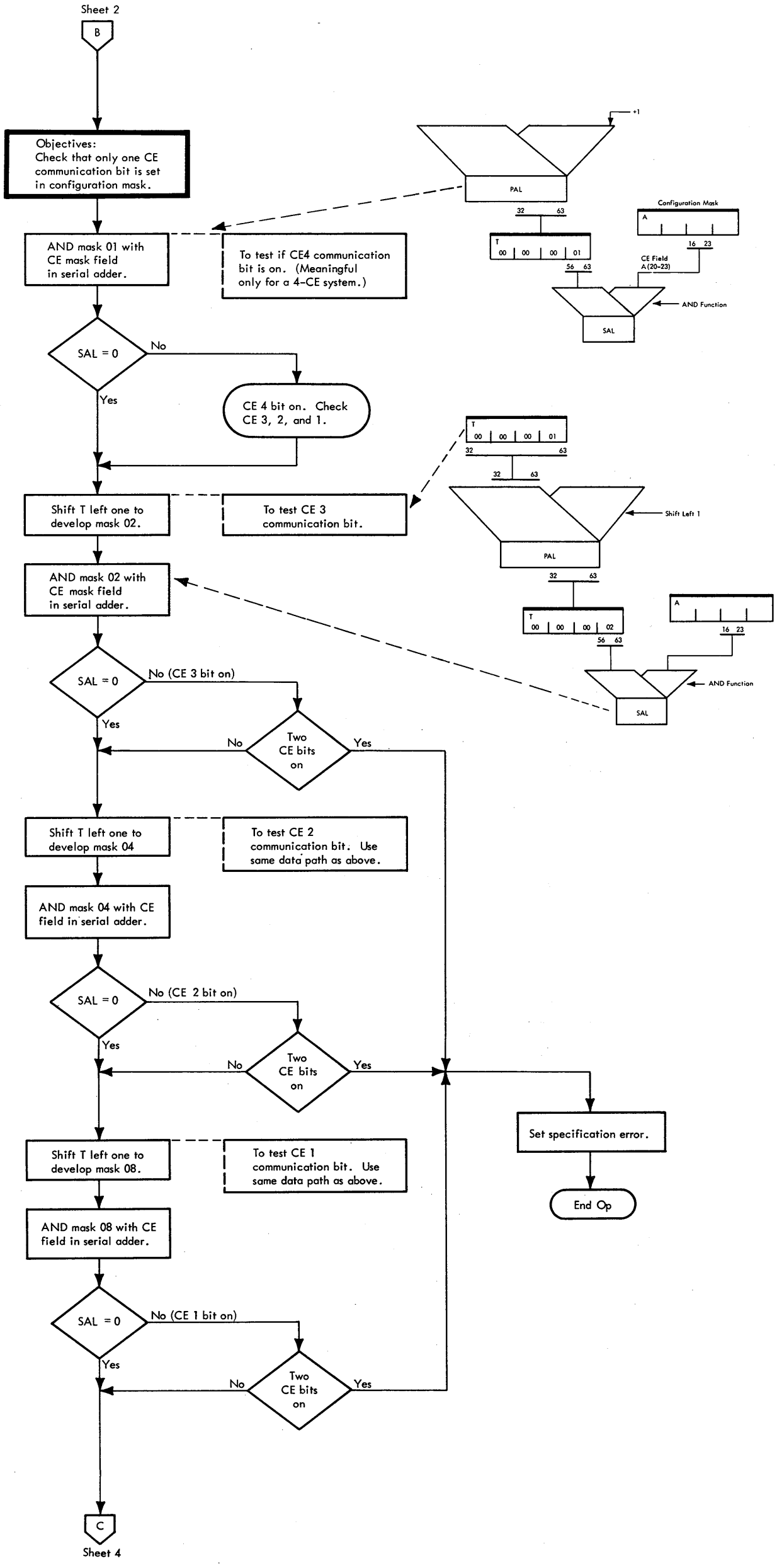
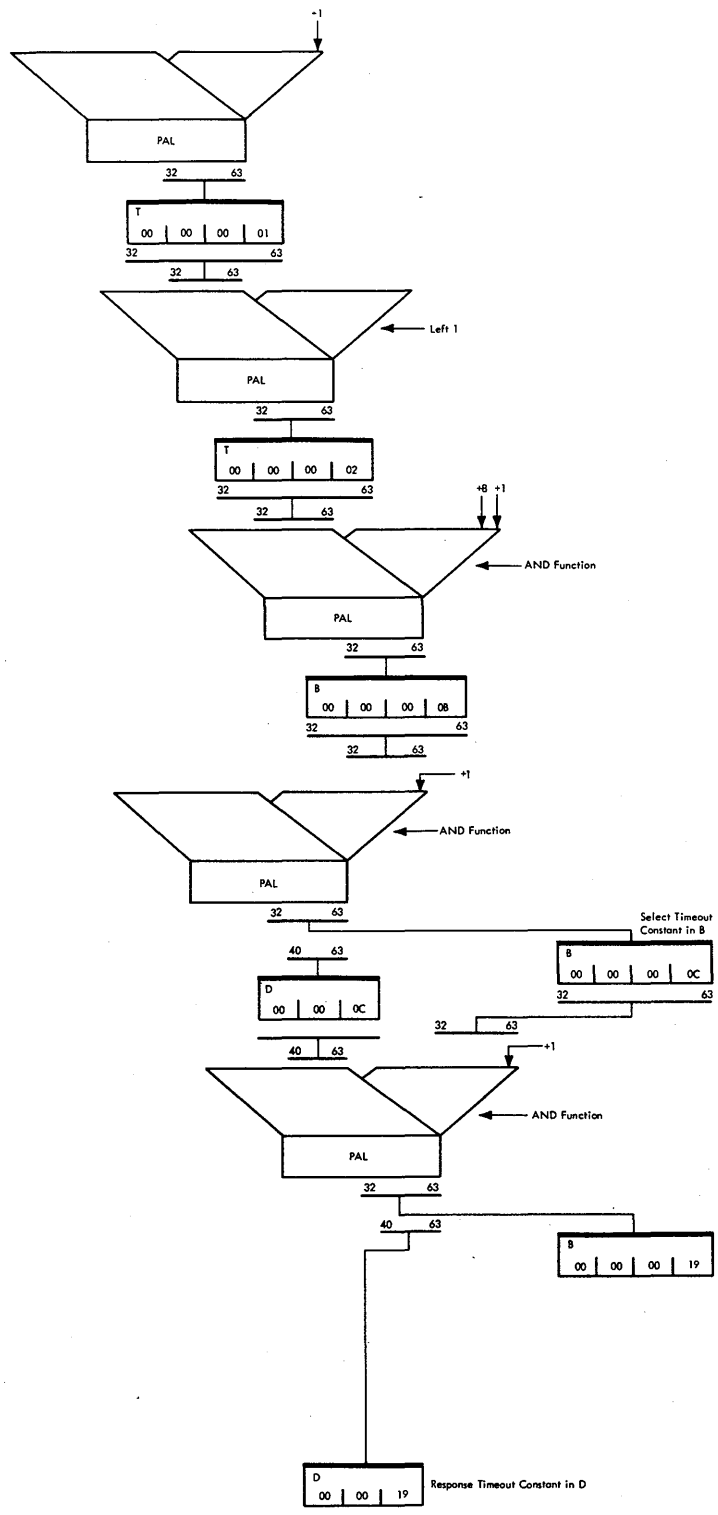


Diagram 5-810. Set Configuration, SCON (01) (Sheet 3 of 6)

Sheet 3
C

Objectives:
Set count of 19 (hex) in B for 'select timeout' (5.0 µsec).
Set count of 19 in D for 'response timeout' (5.0 µsec).



D
Sheet 5

Diagram 5-810. Set Configuration, SCON (01) (Sheet 4 of 6)

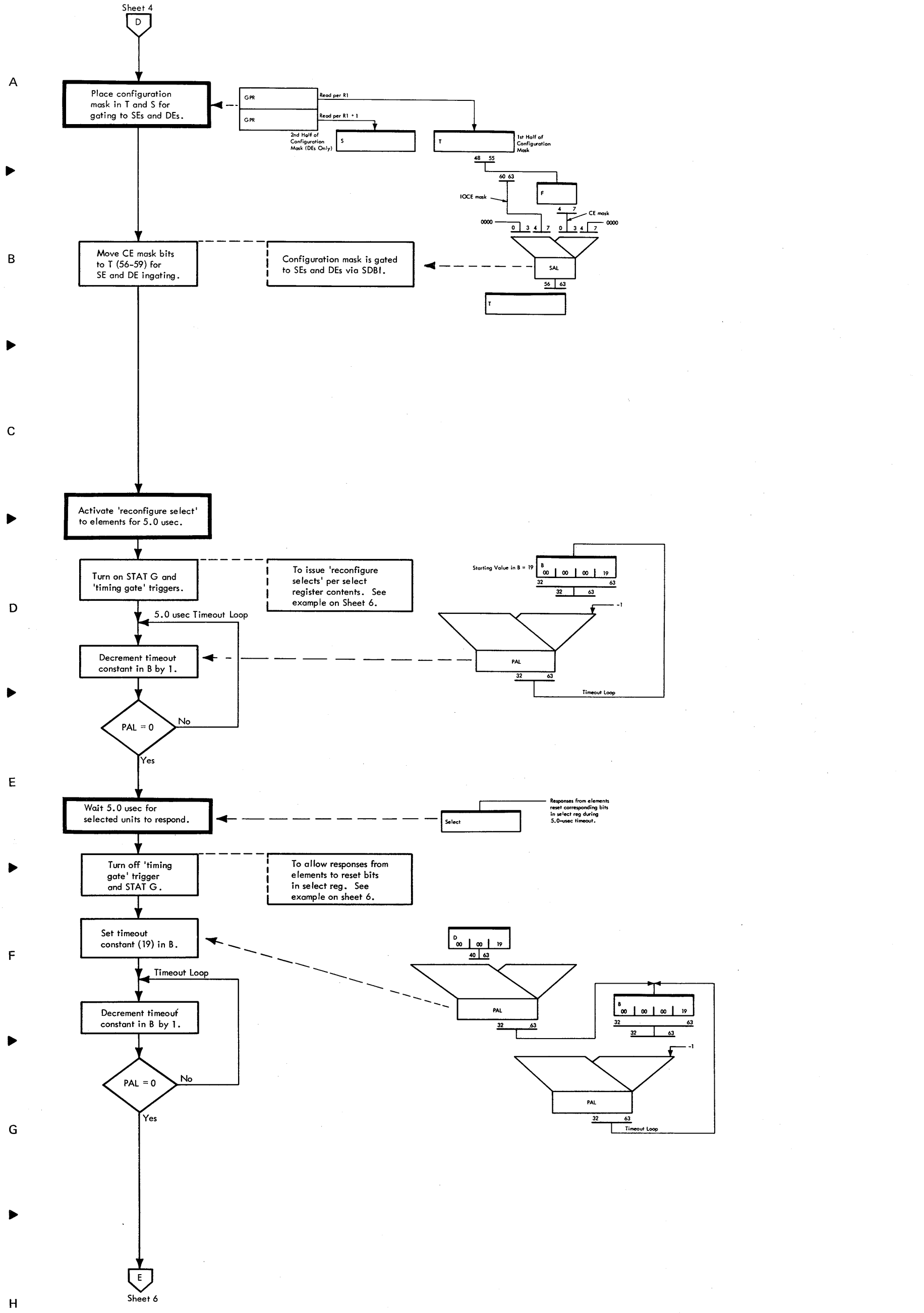
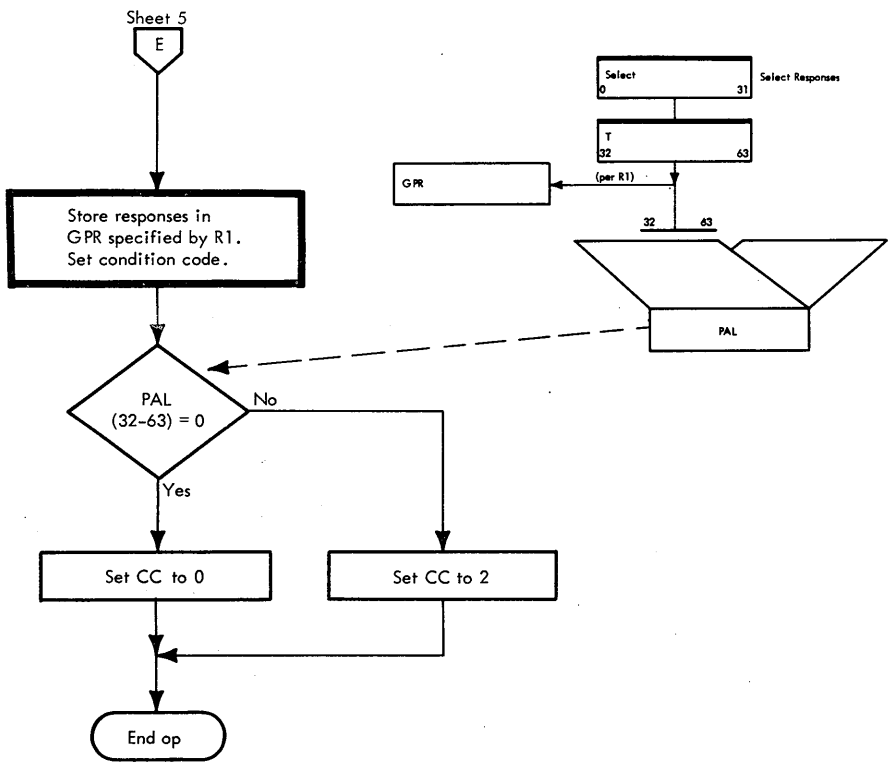


Diagram 5-810. Set Configuration, SCON (01) (Sheet 5 of 6)



The example below shows the basic logic active during the select and response timeouts. CE1 is executing the SCON instruction, and CE2 is receiving the 'scon select'. Similar select and response logic is used for all elements in the system.

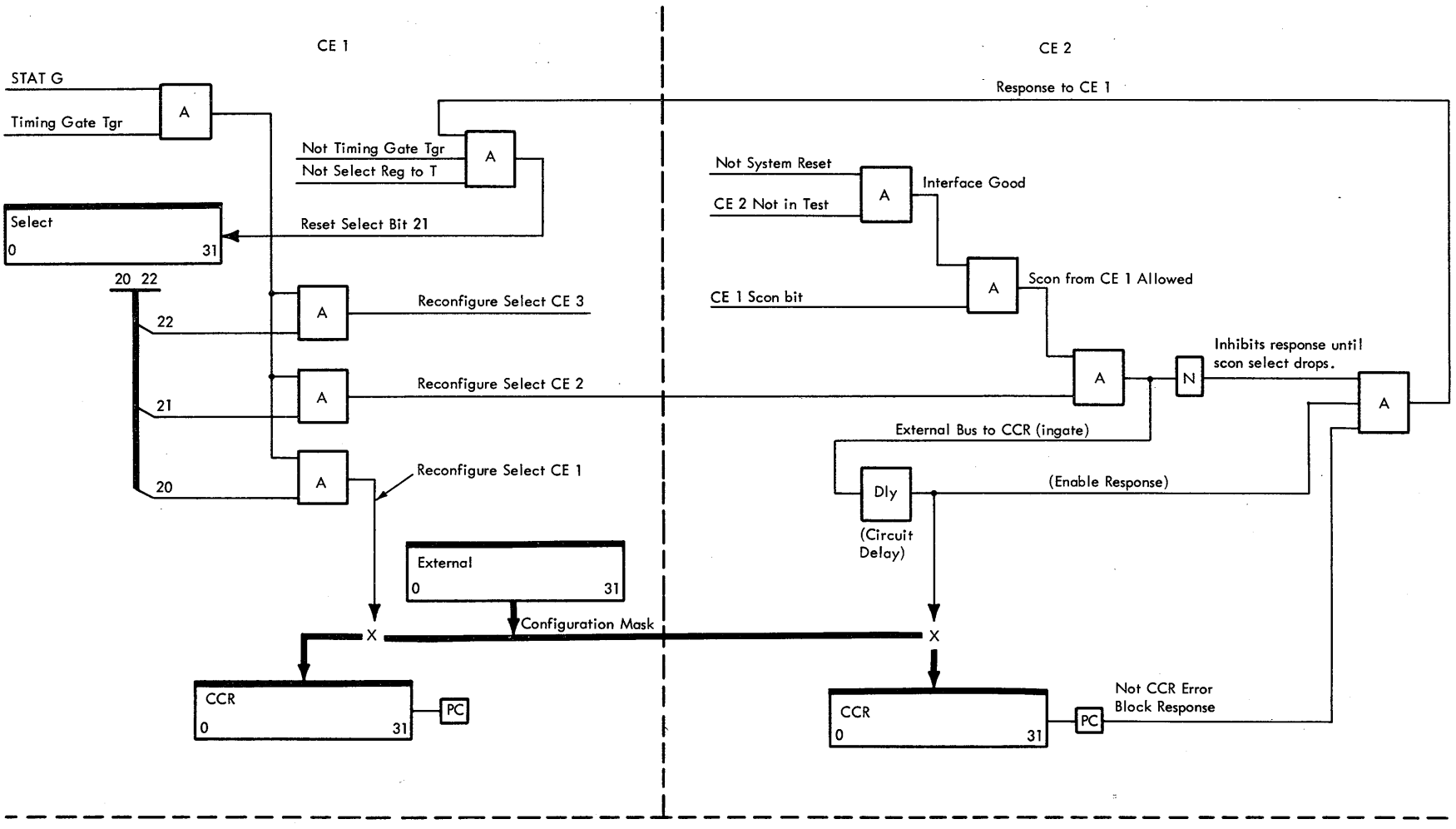
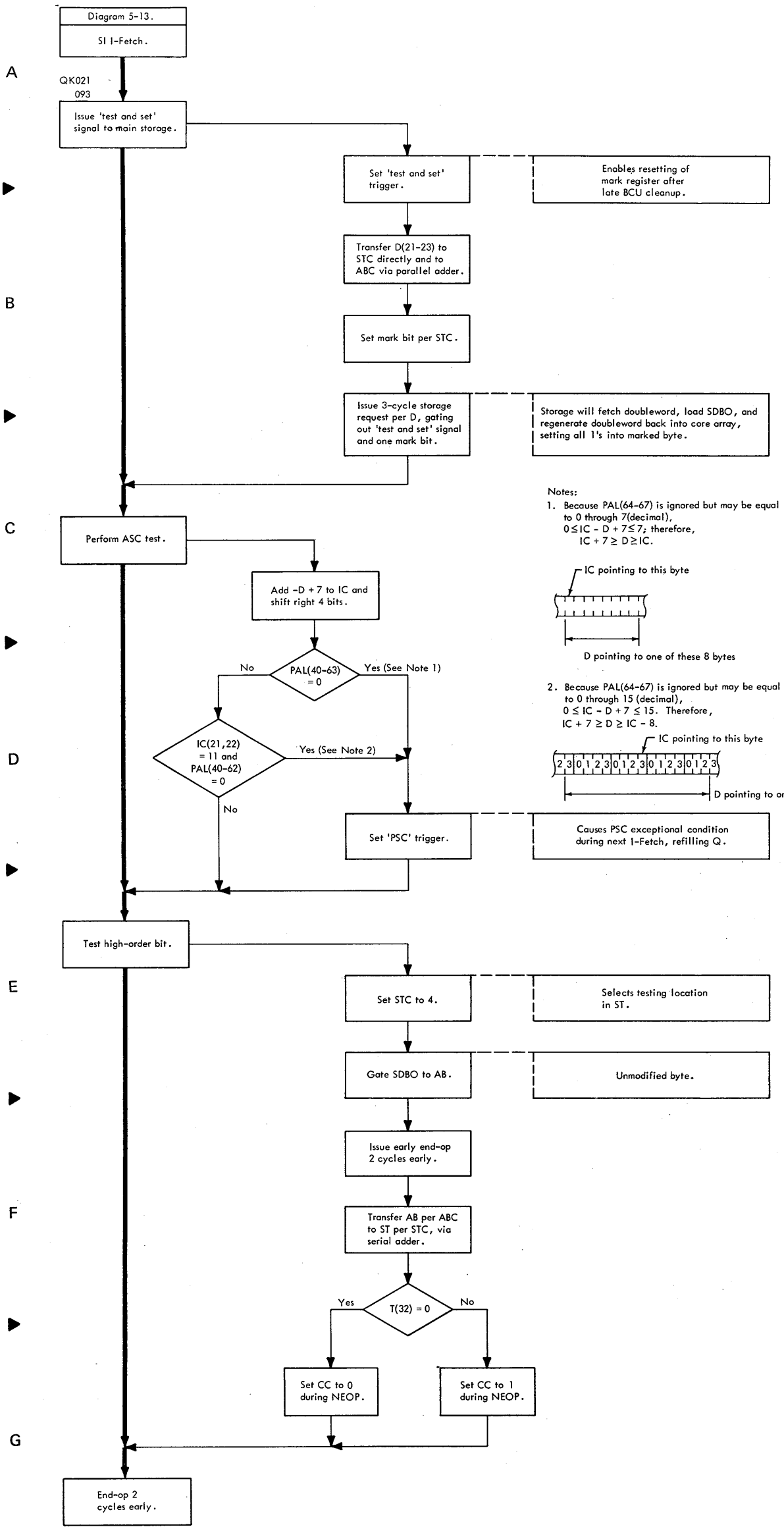
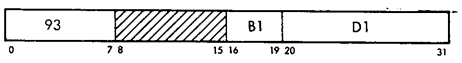


Diagram 5-810. Set Configuration, SCON (01) (Sheet 6 of 6)



- SI Format:



- Test high-order bit (bit 0) of storage operand byte (in storage), set CC according to state of tested bit, and set addressed byte back into storage as all 1's.
- Conditions at start of execution:
  1. 1st 16 bits of instruction are in E.
  2. Storage operand address is in D.
  3. Storage request per D is blocked during I-Fetch ('D sync' latch blocked) because data byte could change before executing this instruction.
- 'Test and set' trigger: ALD MC181.

Diagram 5-811. Test and Set, TS (93)

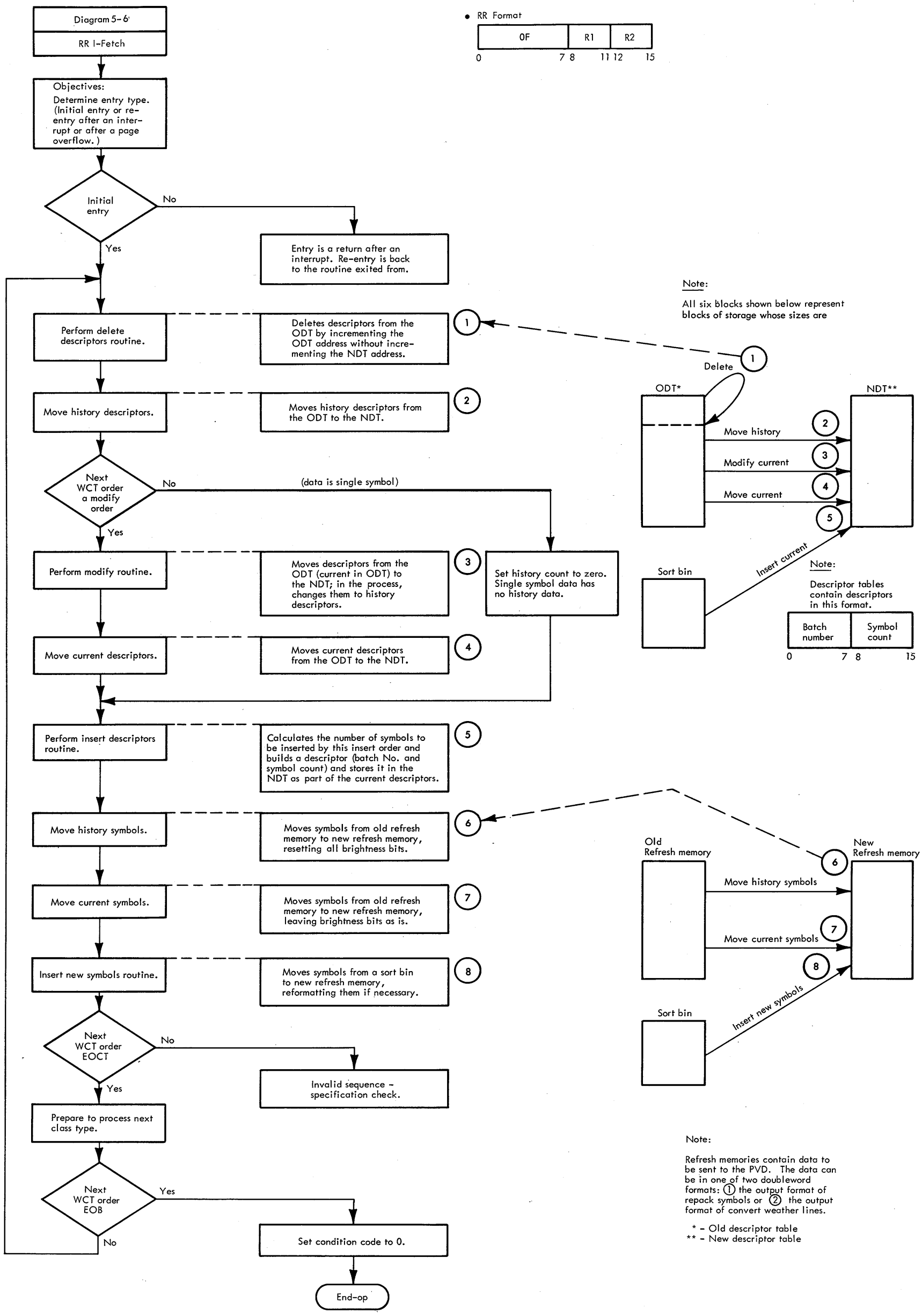
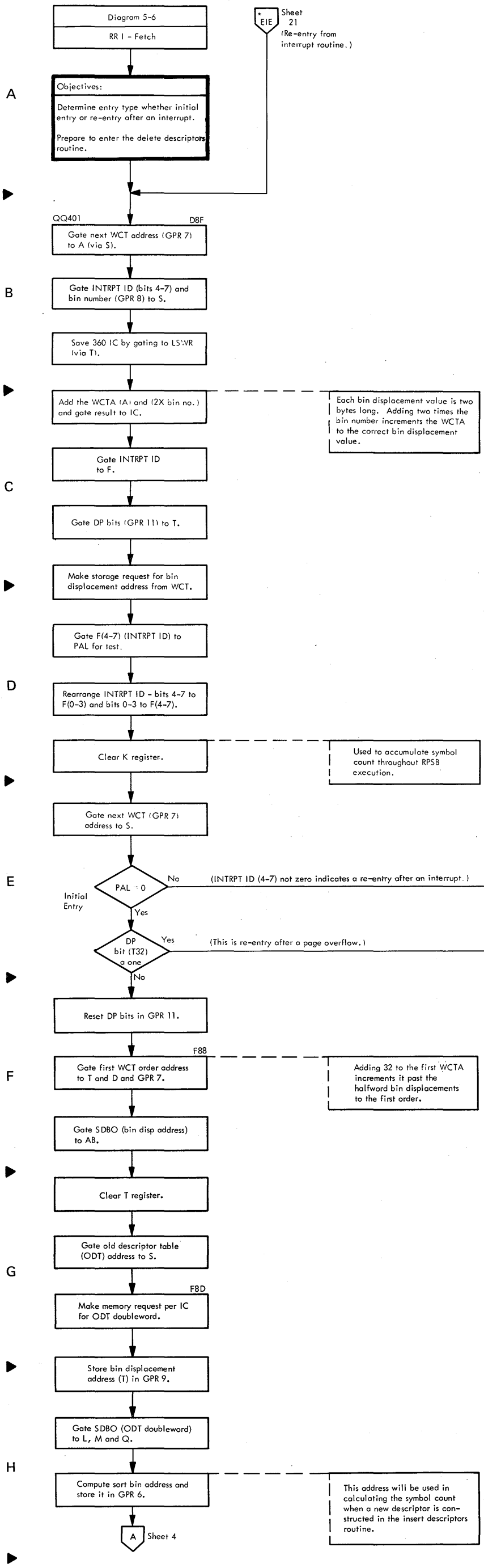
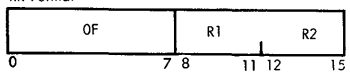


Diagram 5-901. Repack Symbols, Simplified Flow Chart

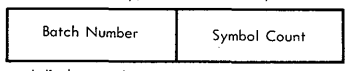


- RR Format

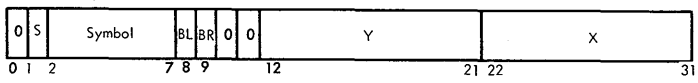


- Purpose: Assembles a new updated display image (refresh memory) for 1/16 of a PVD's area It accomplishes this by deleting, from the old display image, all information which is no longer to be displayed, by moving (and modifying, if necessary) the data remaining in old refresh memory (ORM) to new refresh memory (NRM), and by adding new data to be displayed by inserting data from a sort bin (created by CSS execution) into NRM.

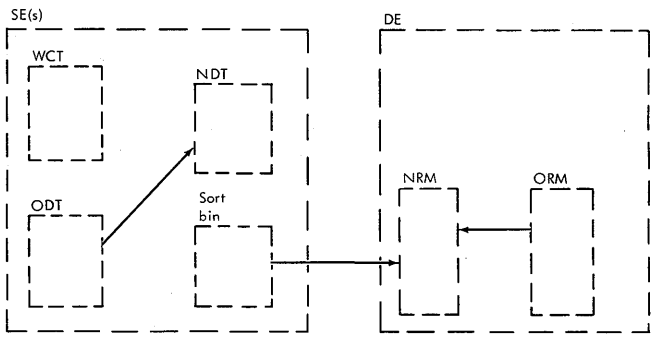
- RPSB moves two types of data, descriptors



and display words,



and is logically divided into two sections. The first builds a new descriptor table by using the applicable previous descriptors (from the ODT) and by generating new descriptors for the new input data (from CSS execution(s)) located in a sort bin. This section operates only in SEs. The second section builds a new display image for a sort bin (called new refresh memory, NRM), using the symbol counts accumulated while building the new descriptor table (NDT) which actually describes NRM. This section operates in both SEs and DEs as shown below.



* 3-letter designations identify actual line edge connectors on CAS page.

Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 1 of 21)

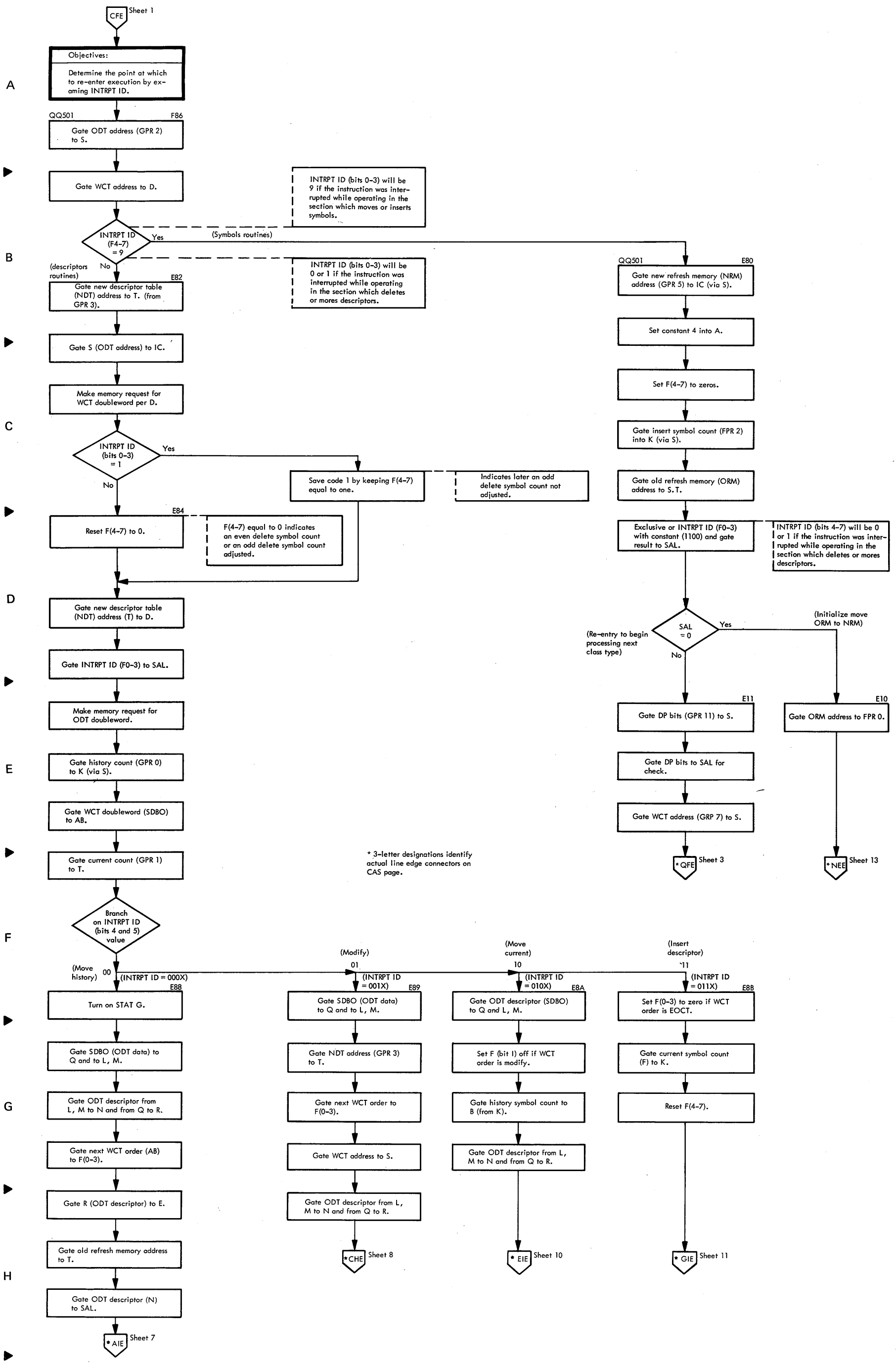


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 2 of 21)

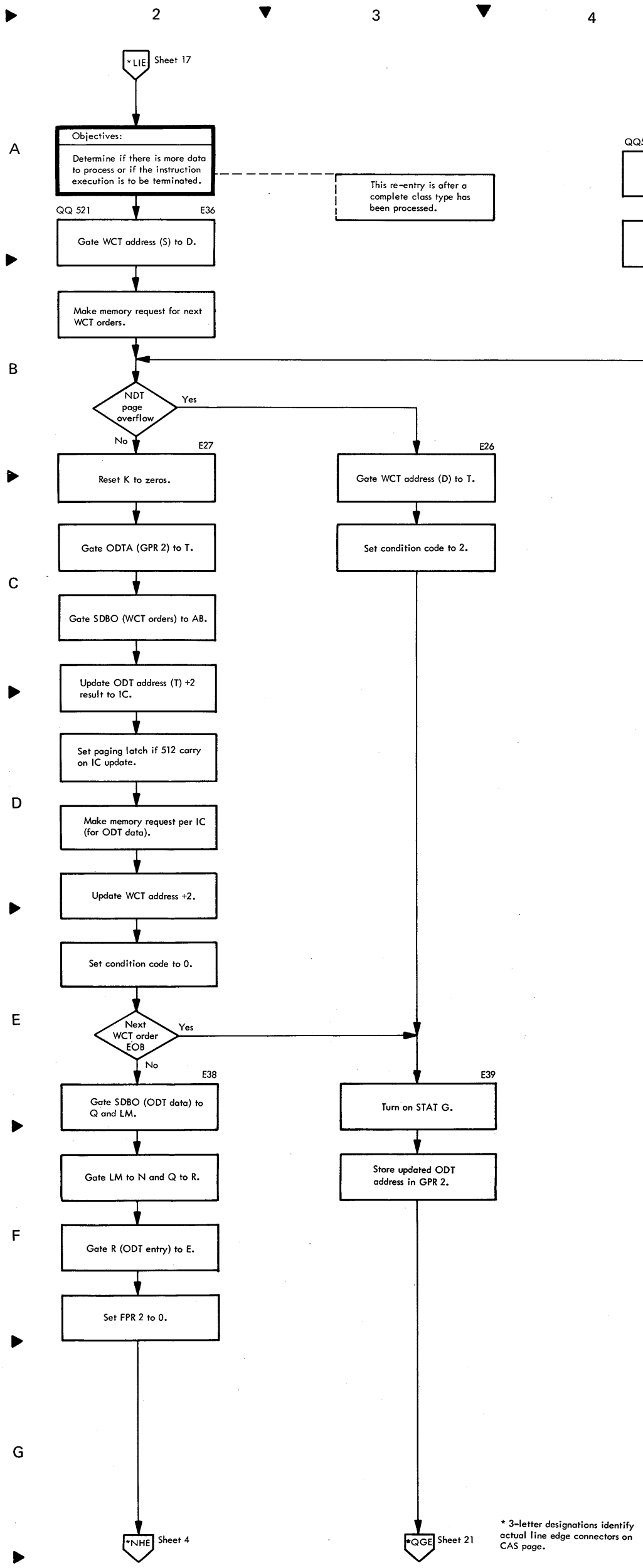


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 3 of 21)

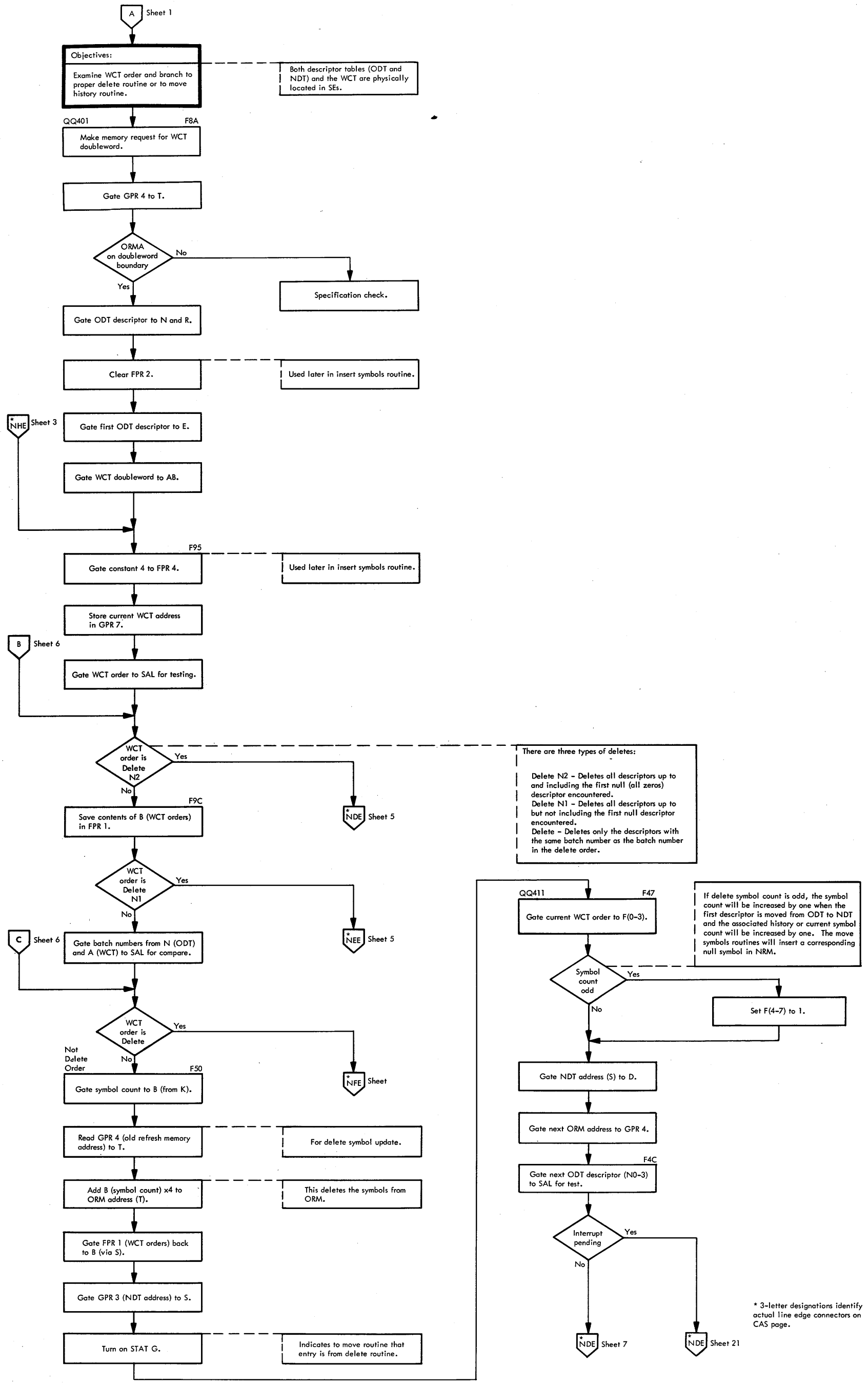


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 4 of 21)

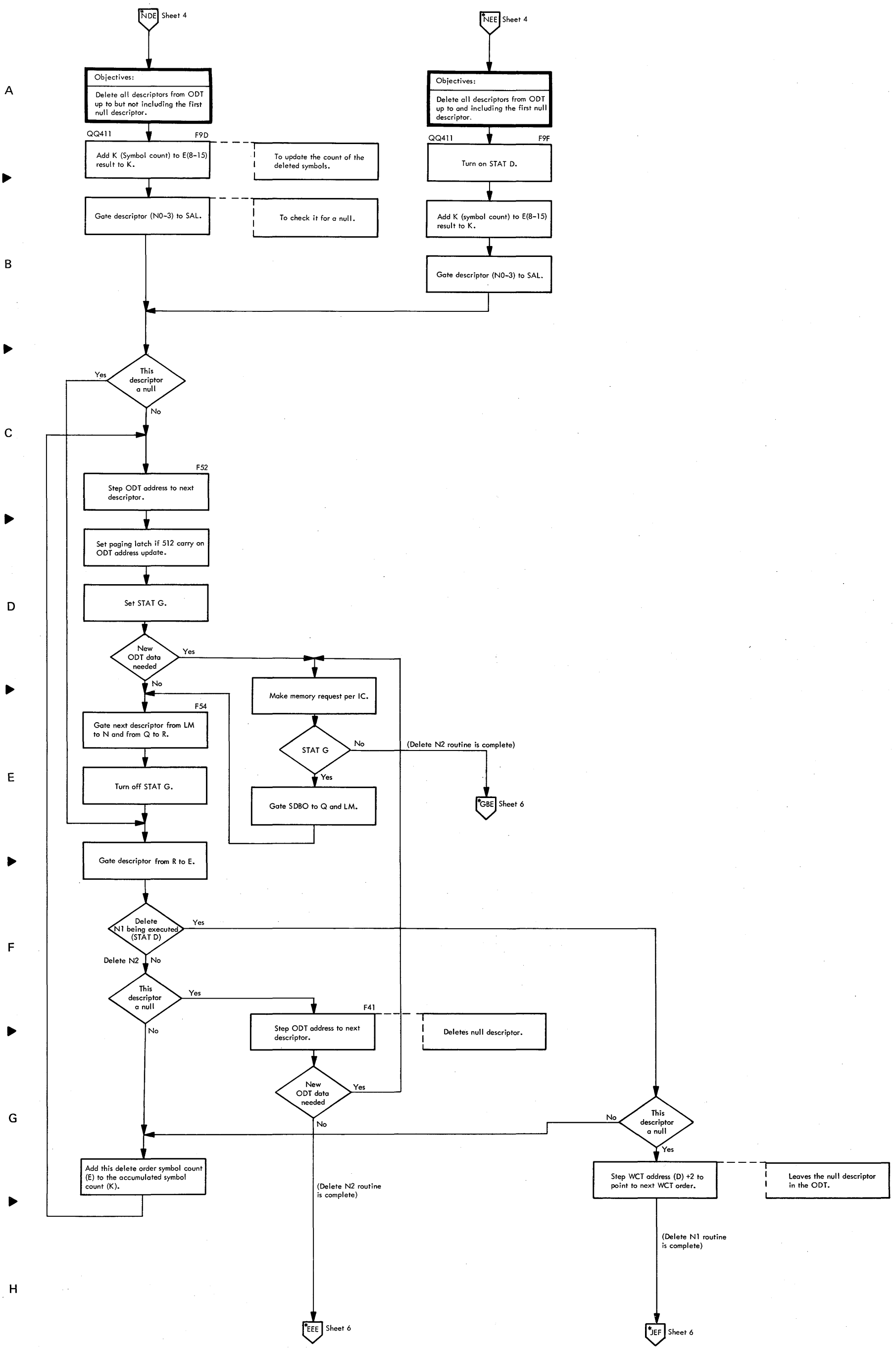
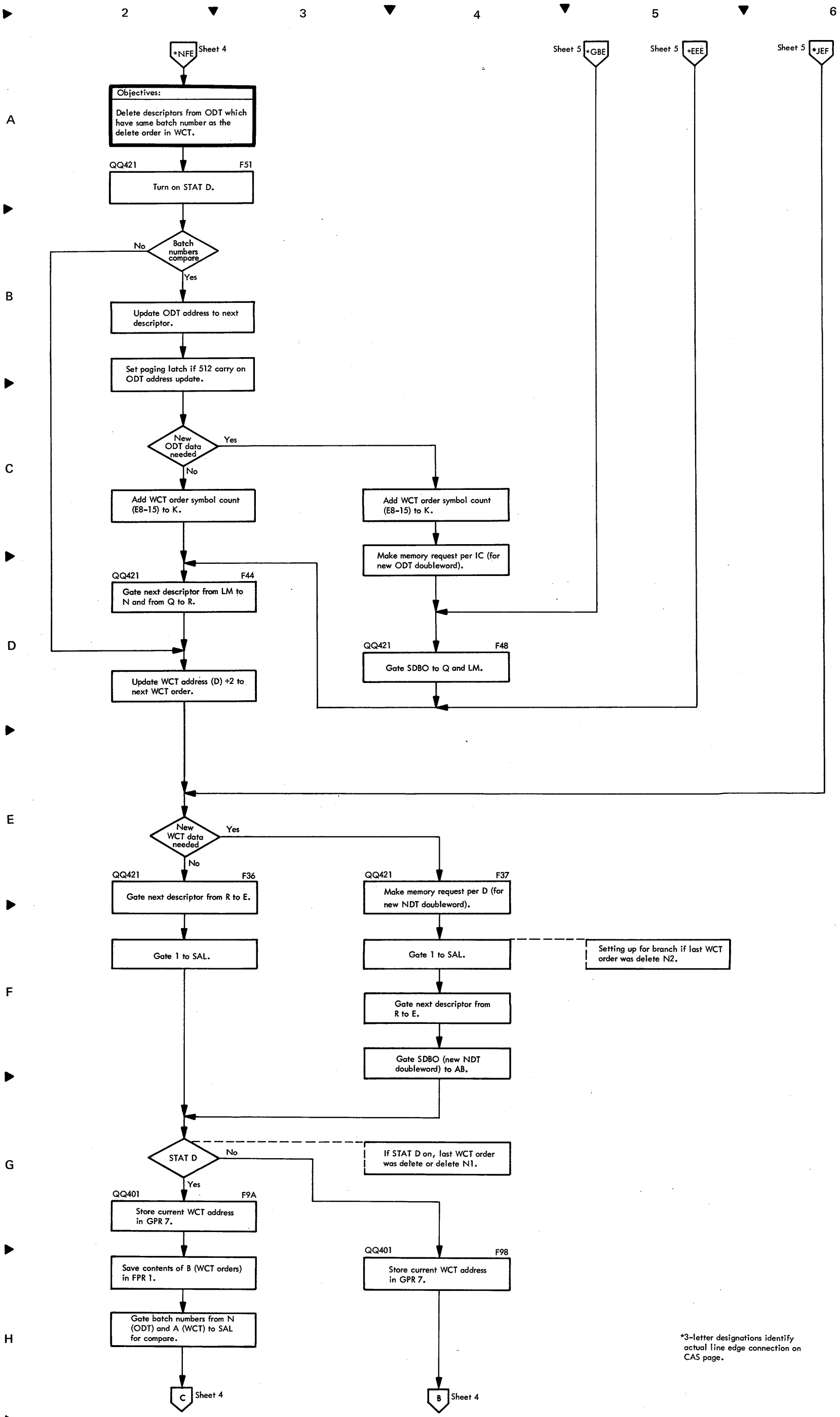


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 5 of 21)



*3-letter designations identify actual line edge connection on CAS page.

Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 6 of 21)

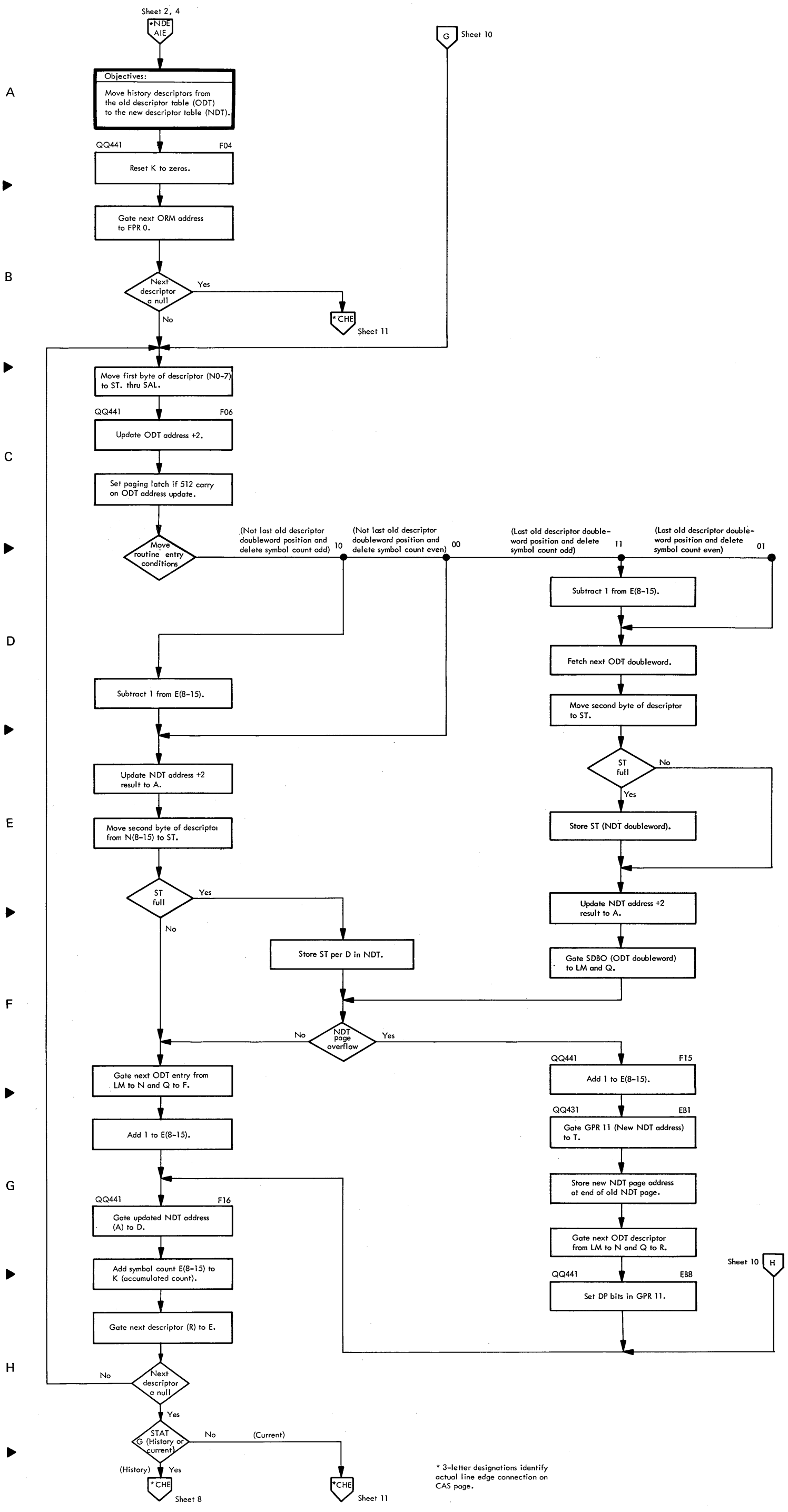


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 7 of 21)

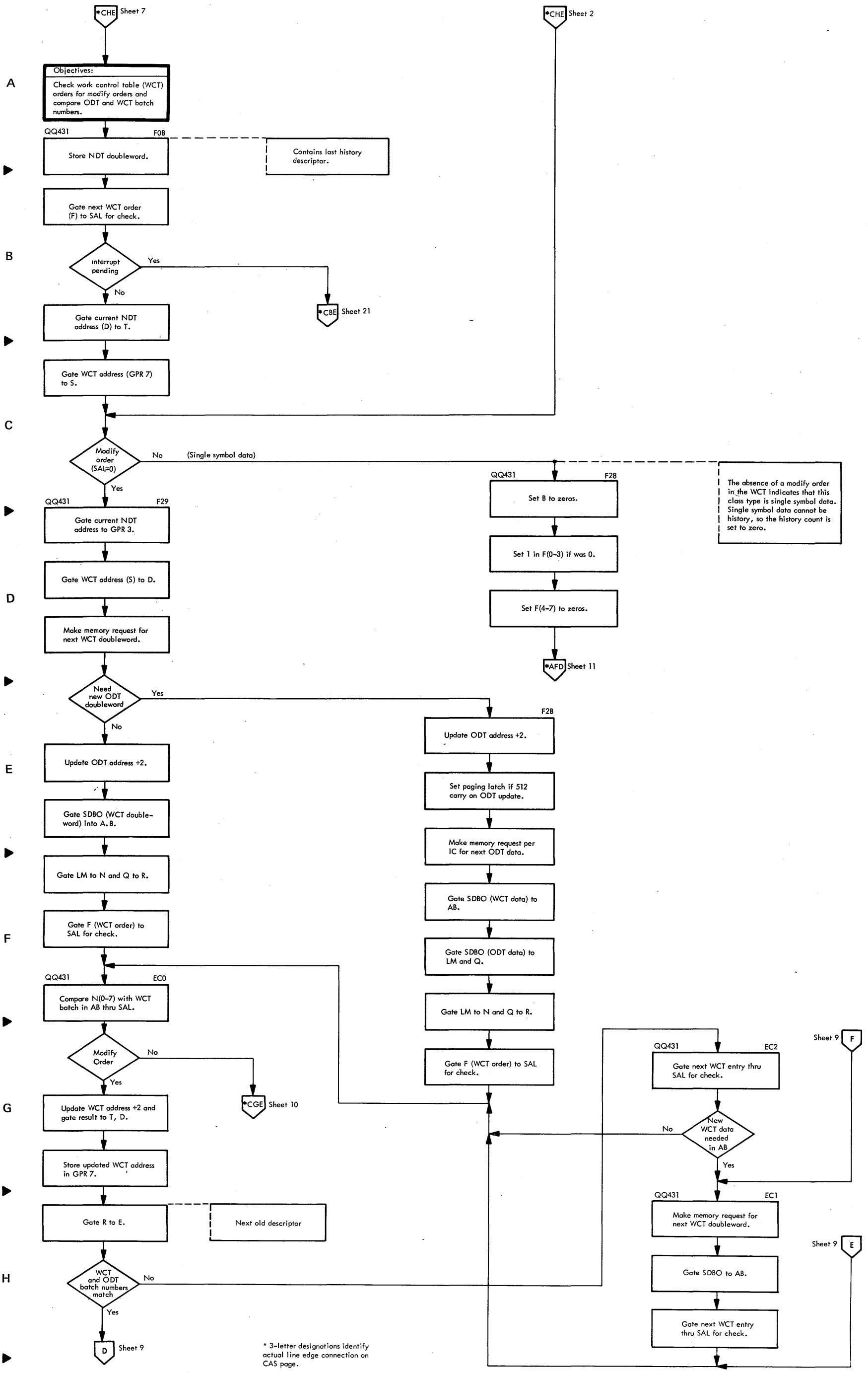


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 8 of 21)

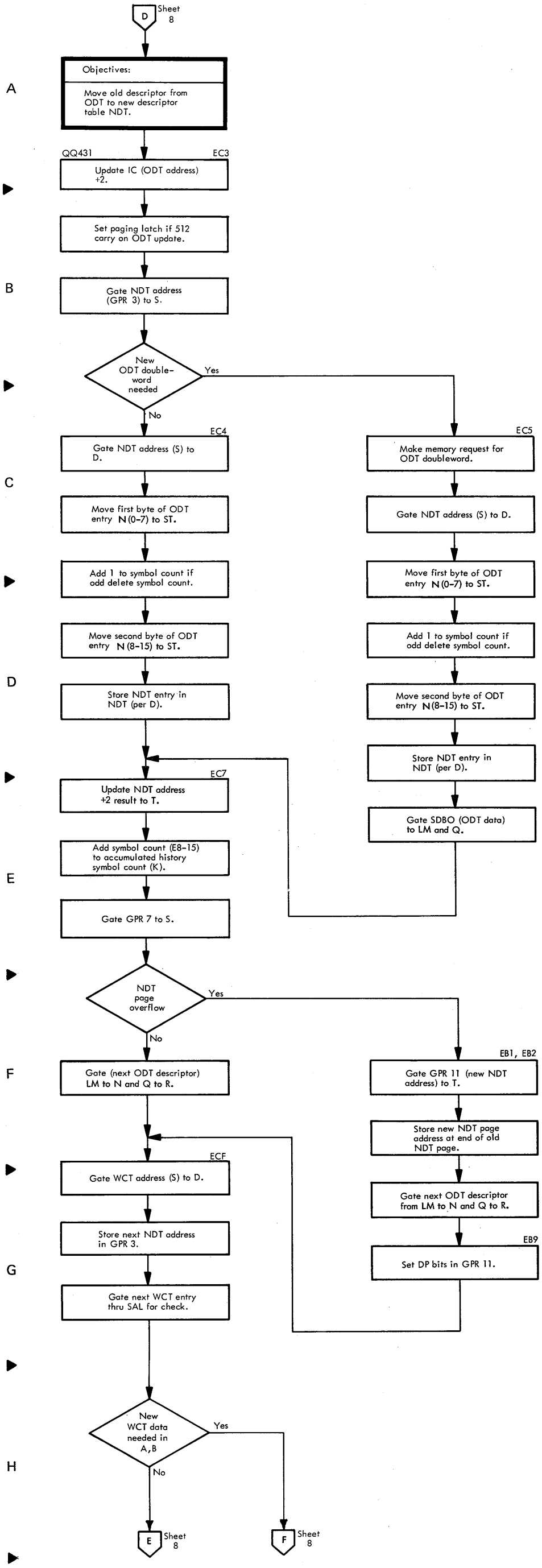


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 9 of 21)

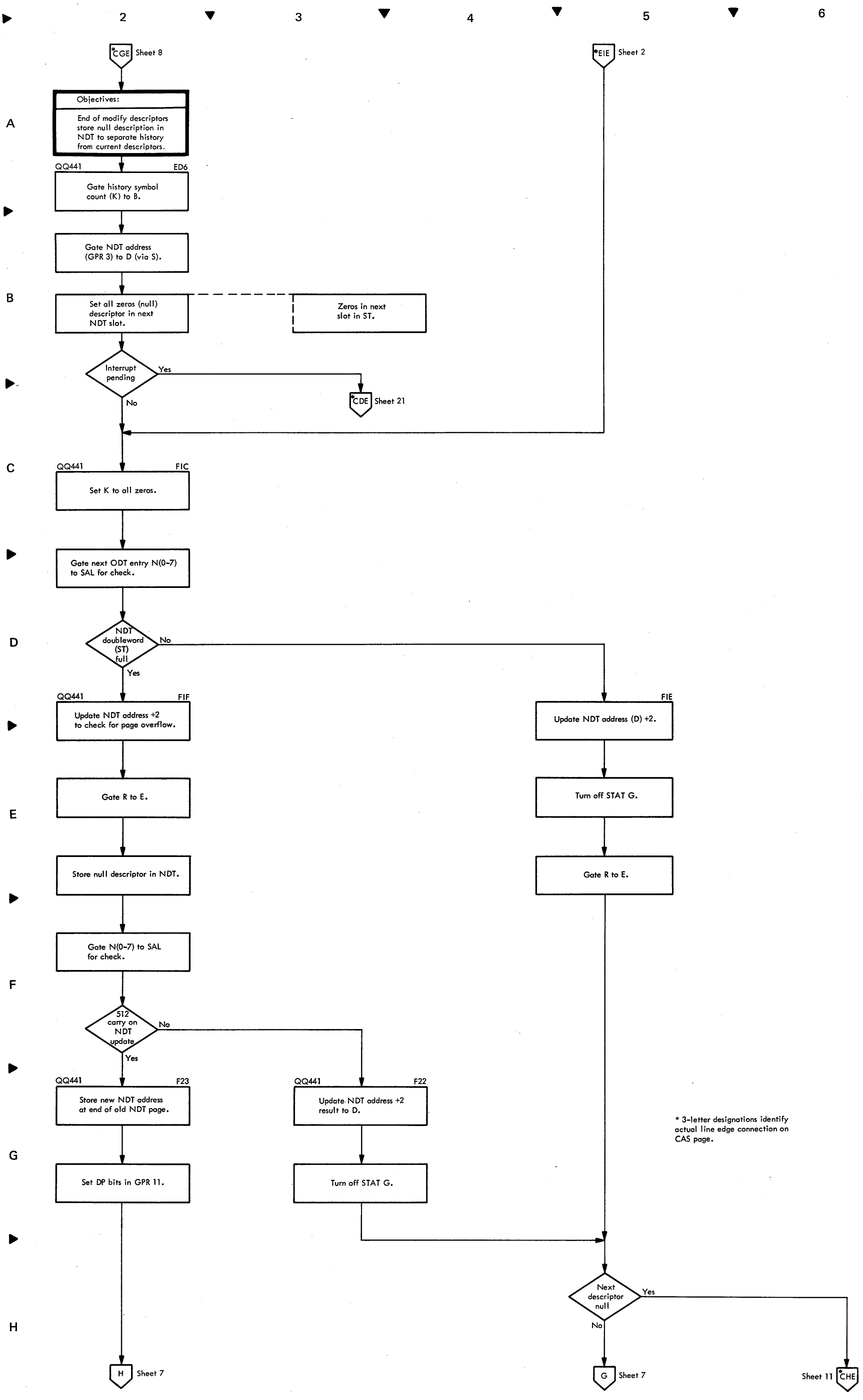


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 10 of 21)

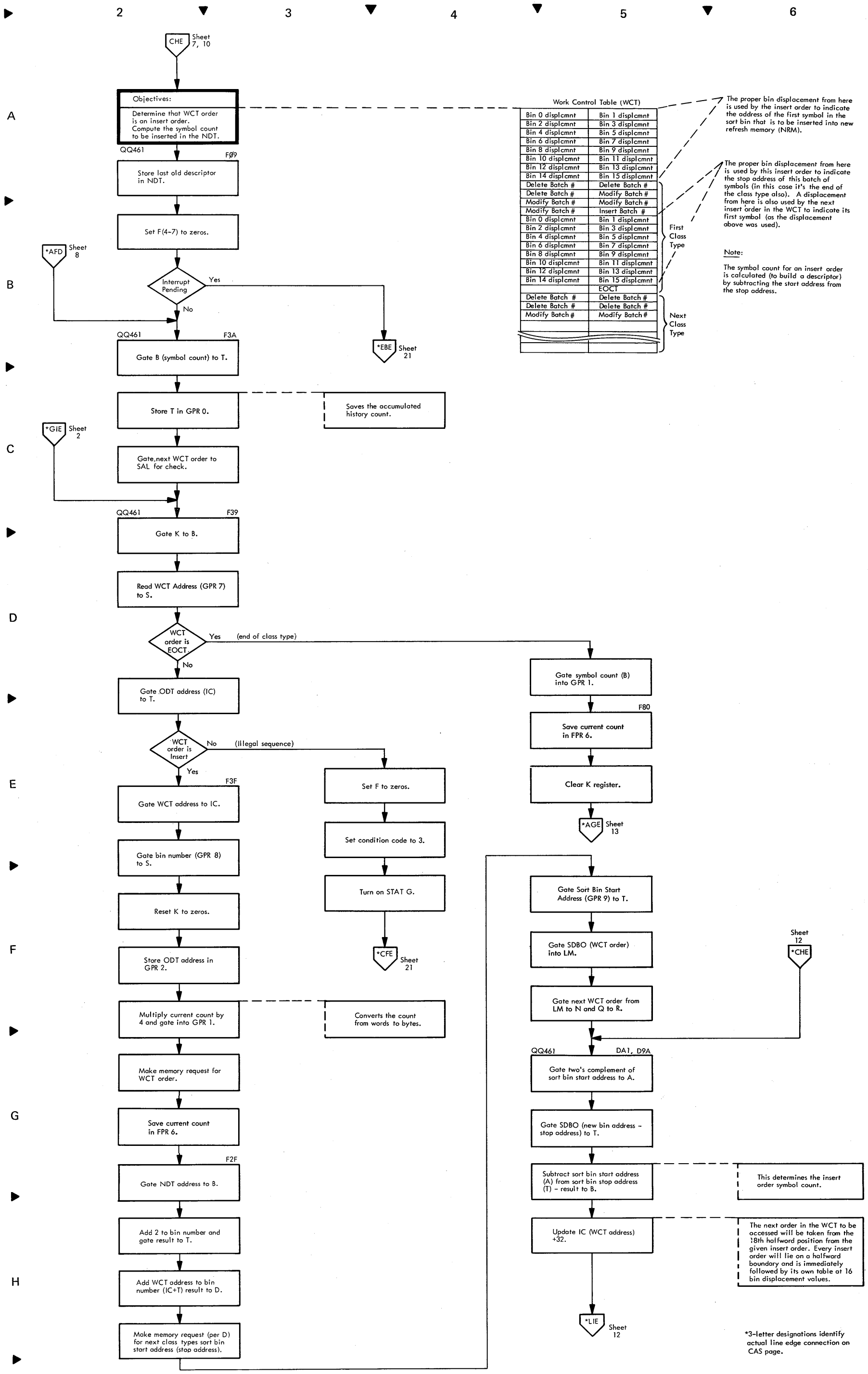


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 11 of 21)

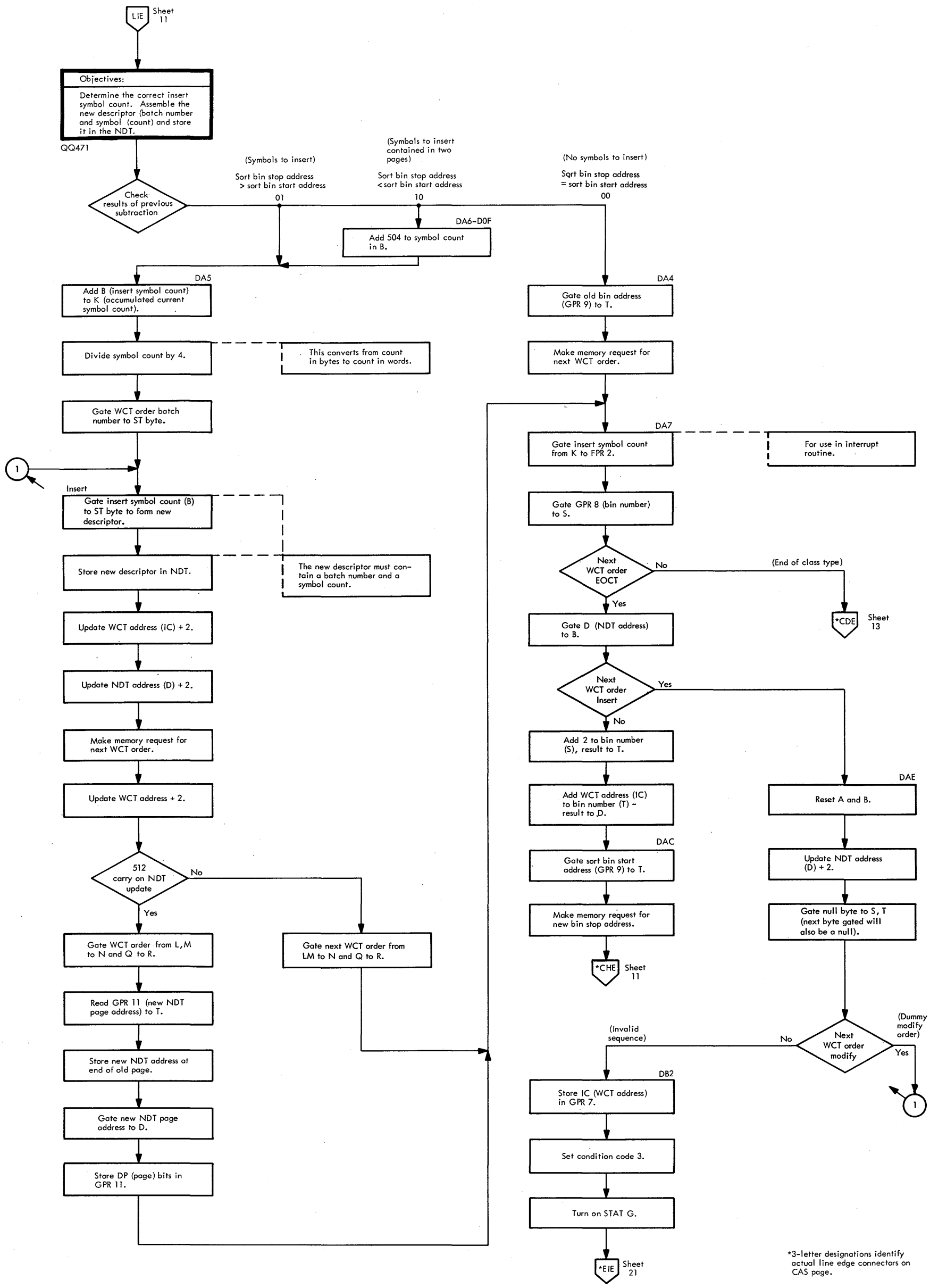


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 12 of 21)

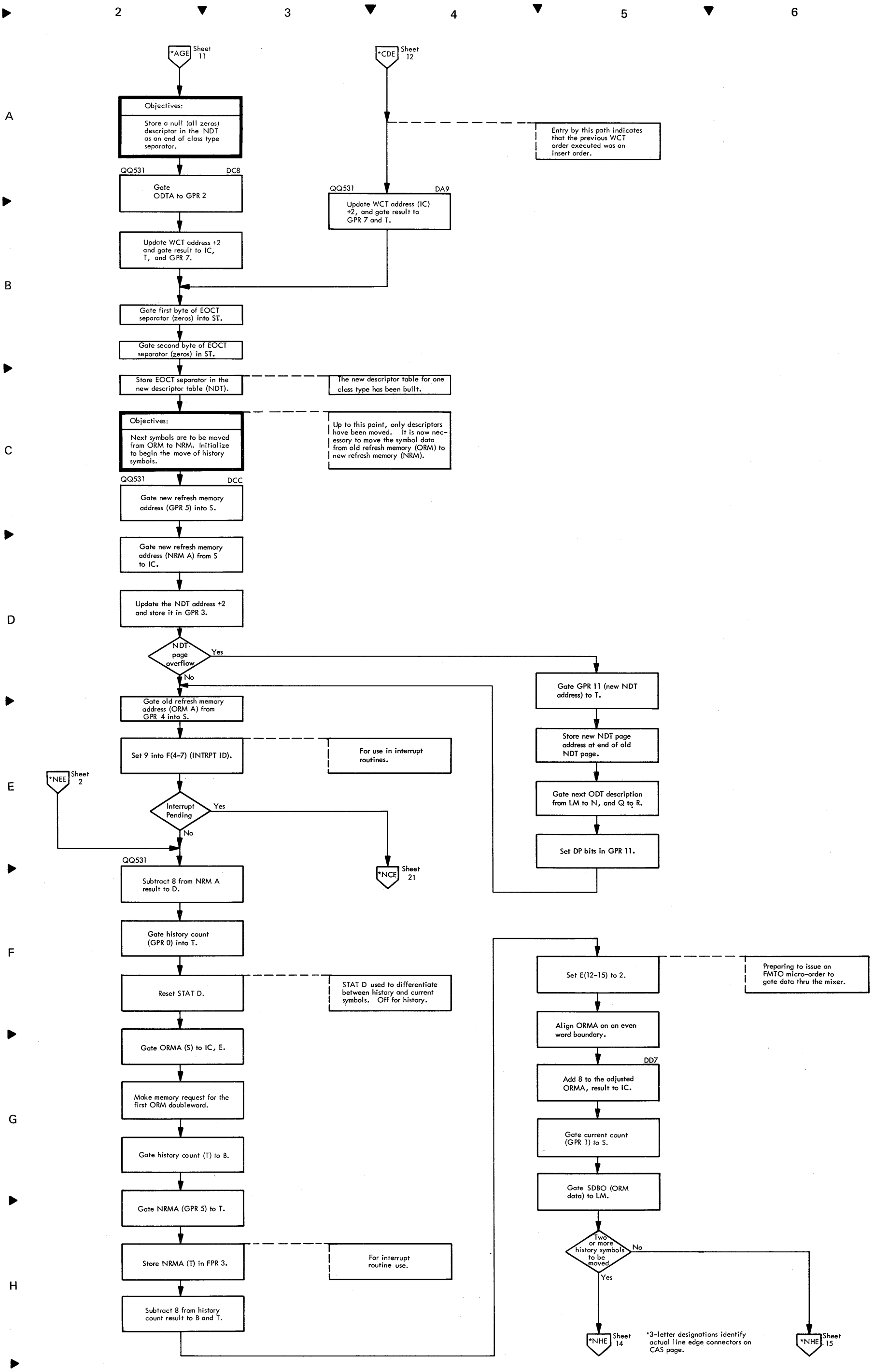


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 13 of 21)

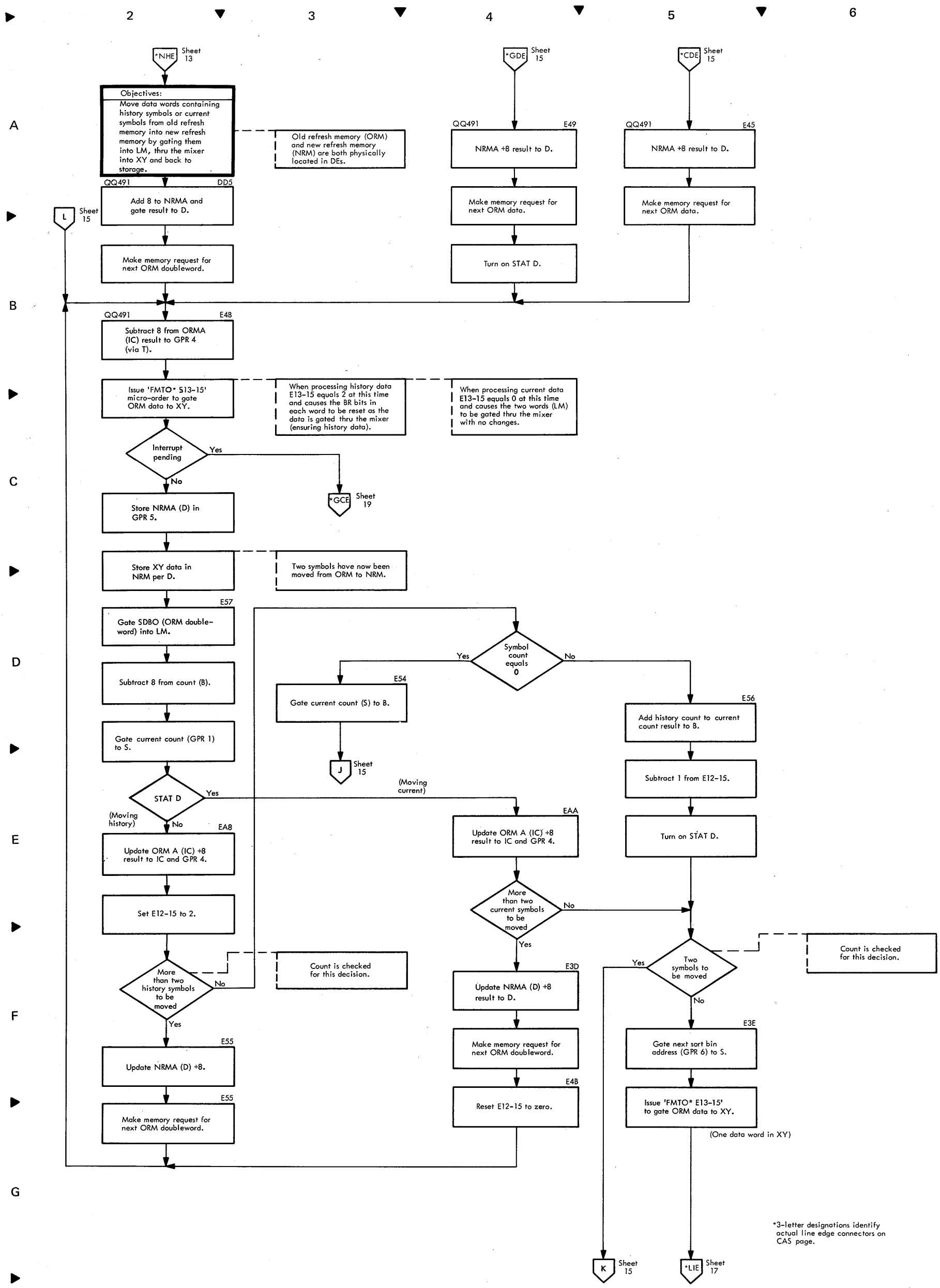


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 14 of 21)

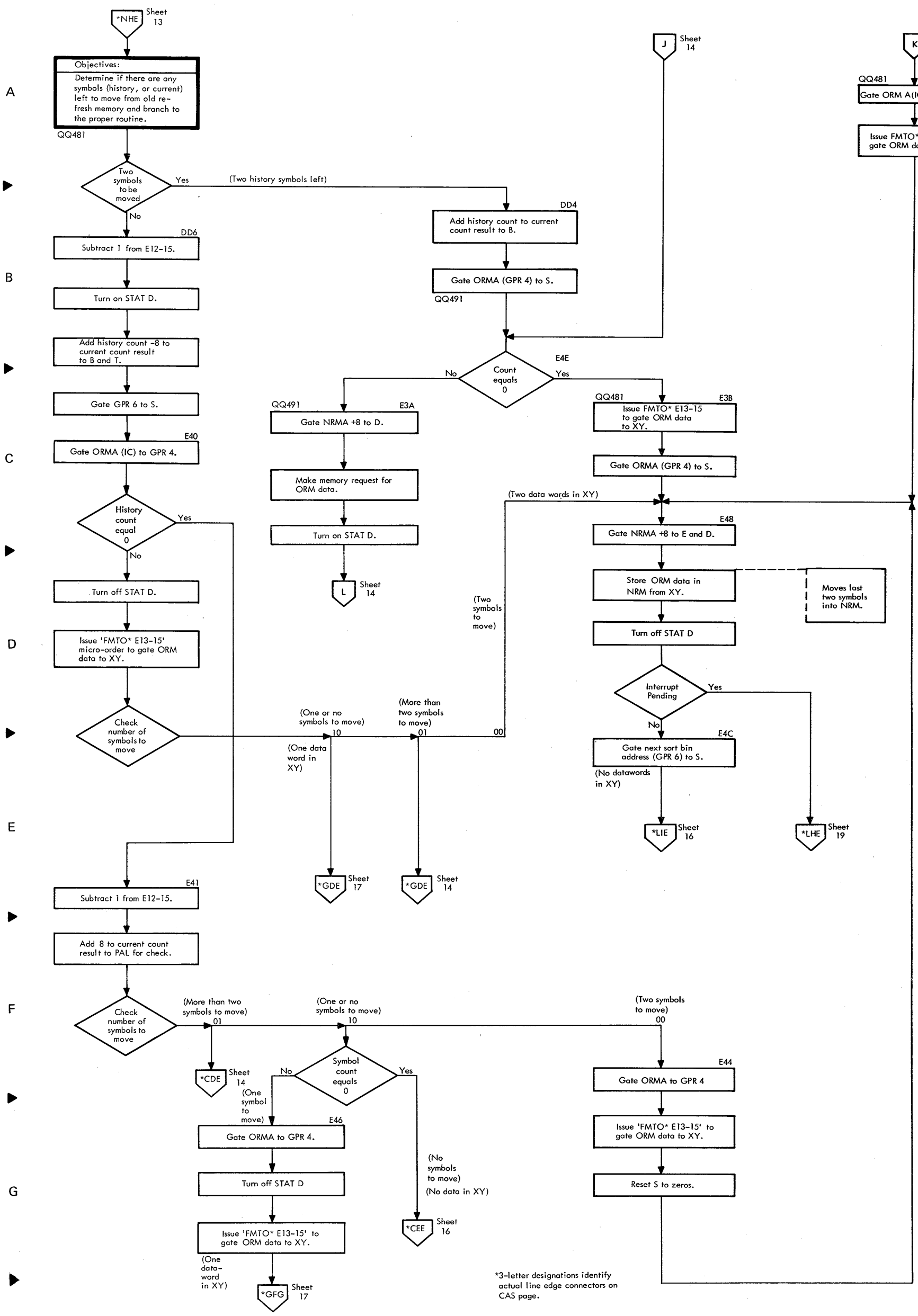


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 15 of 21)

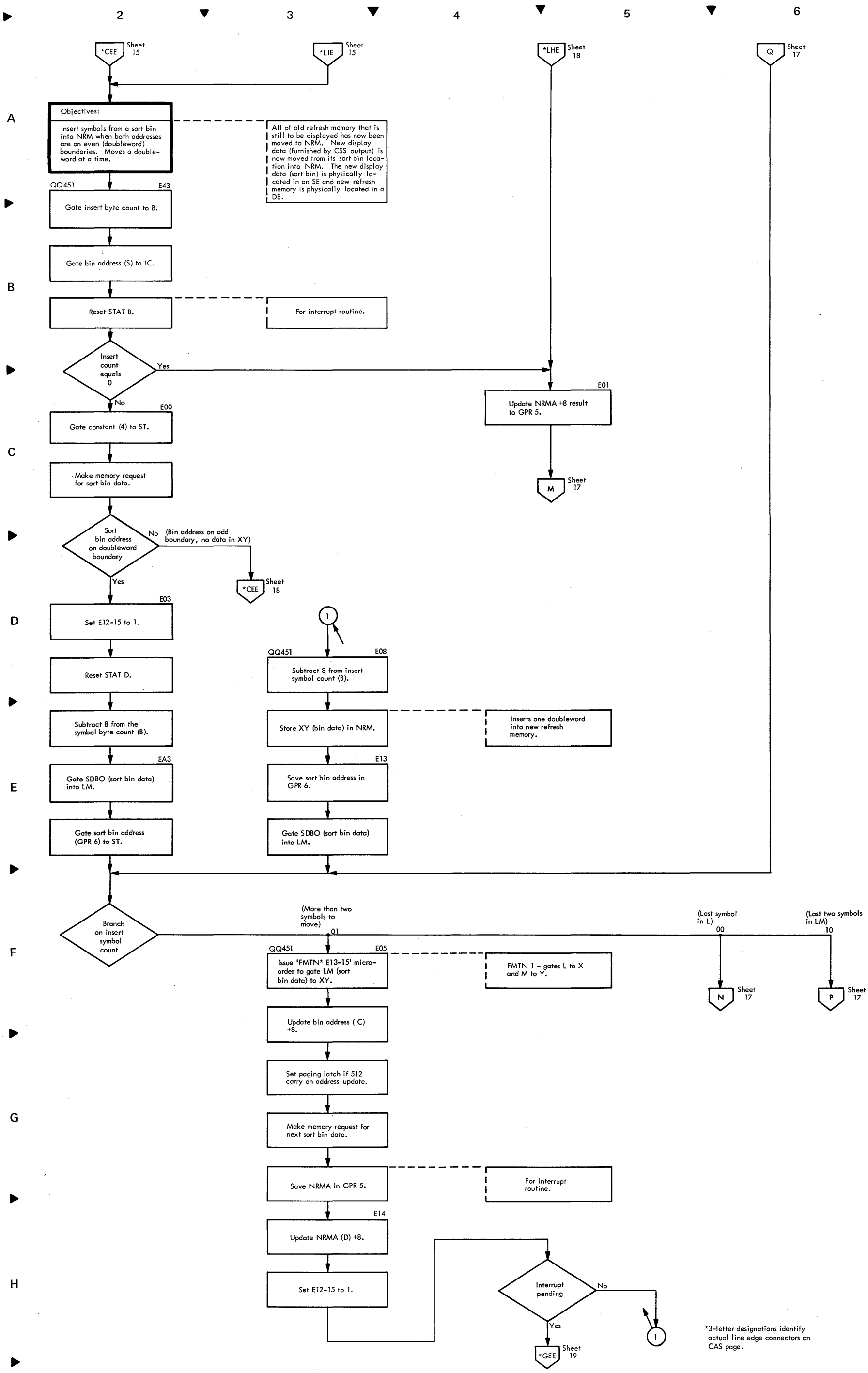
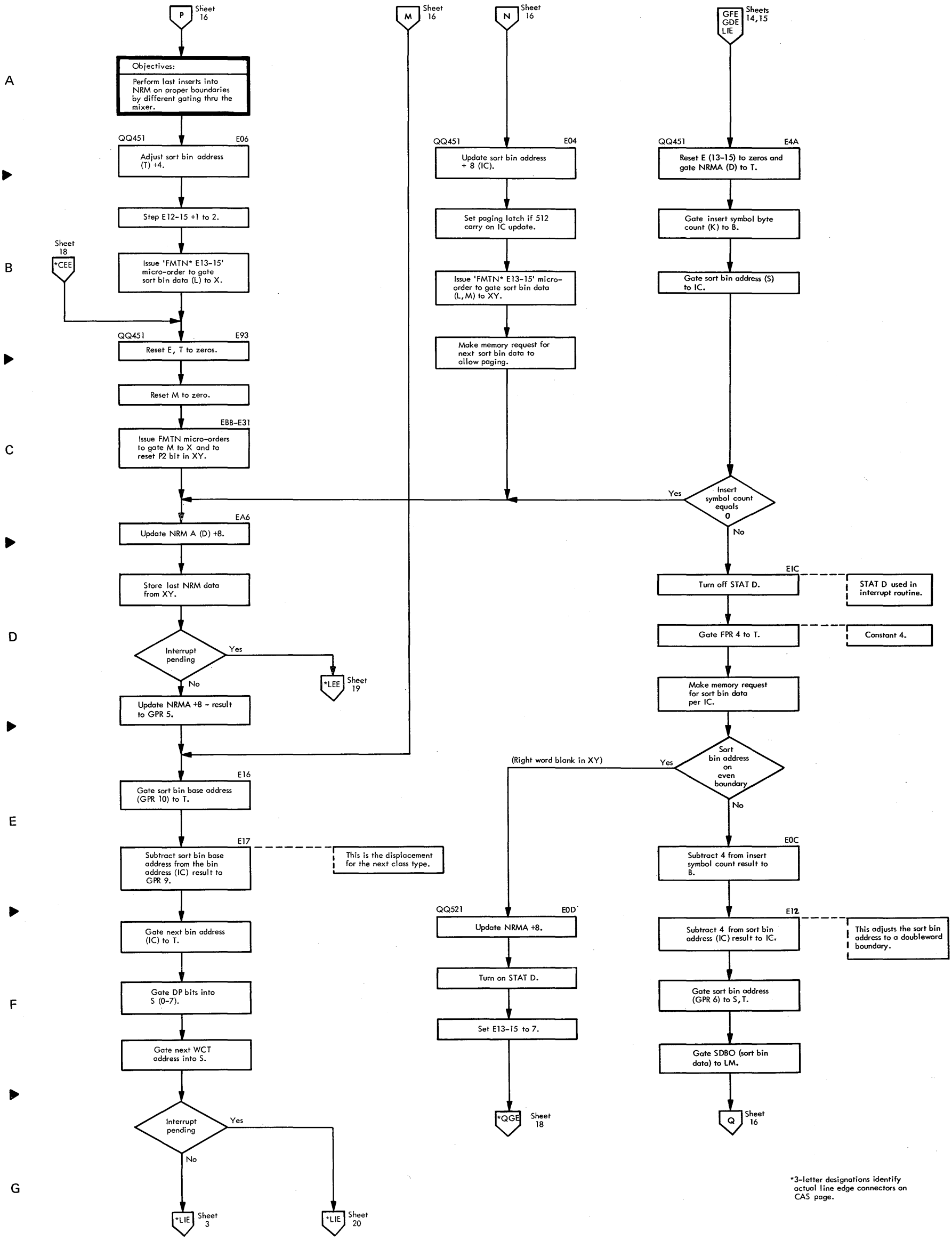


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 16 of 21)



*3-letter designations identify actual line edge connectors on CAS page.

Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 17 of 21)

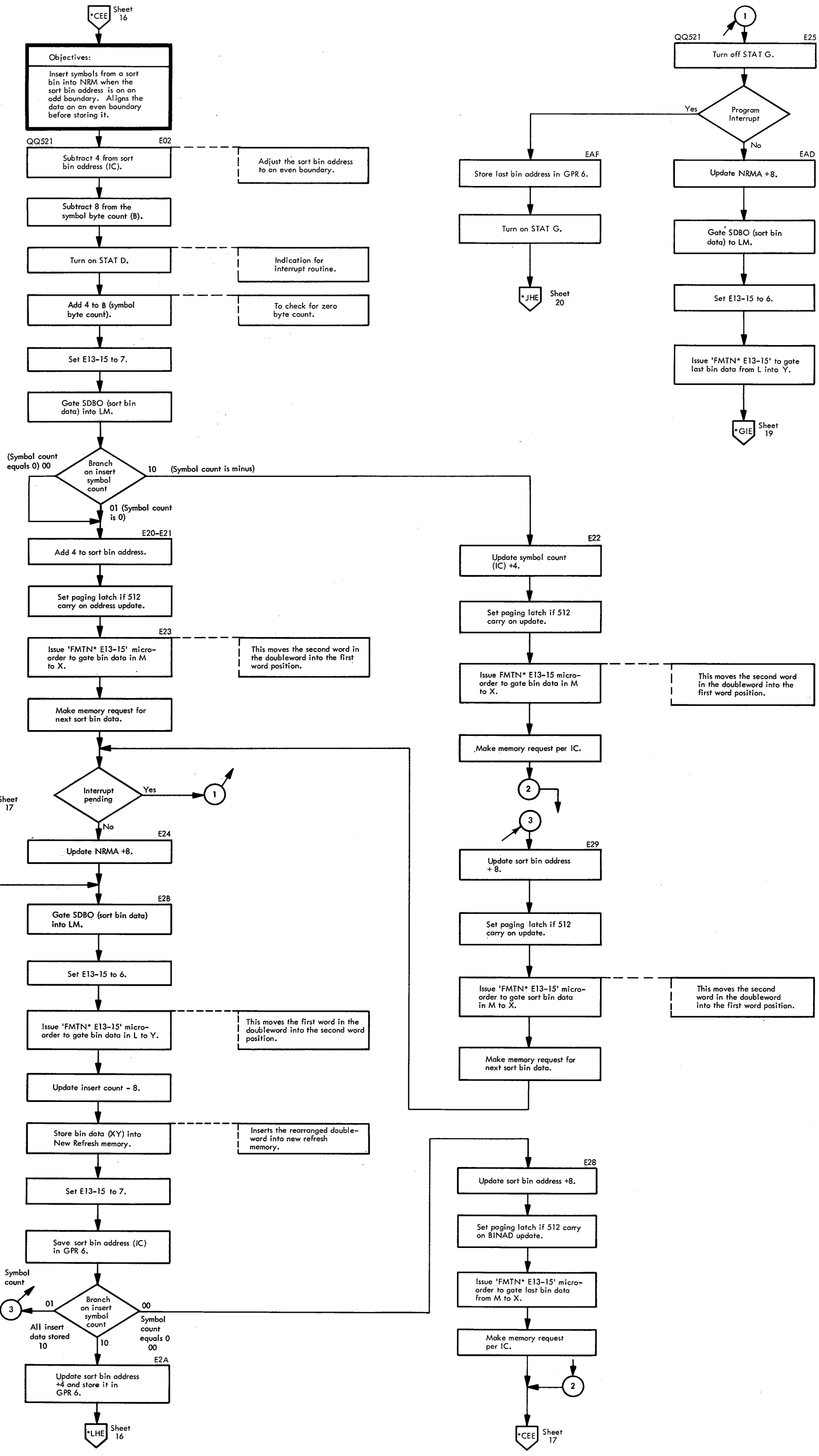


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 18 of 21)

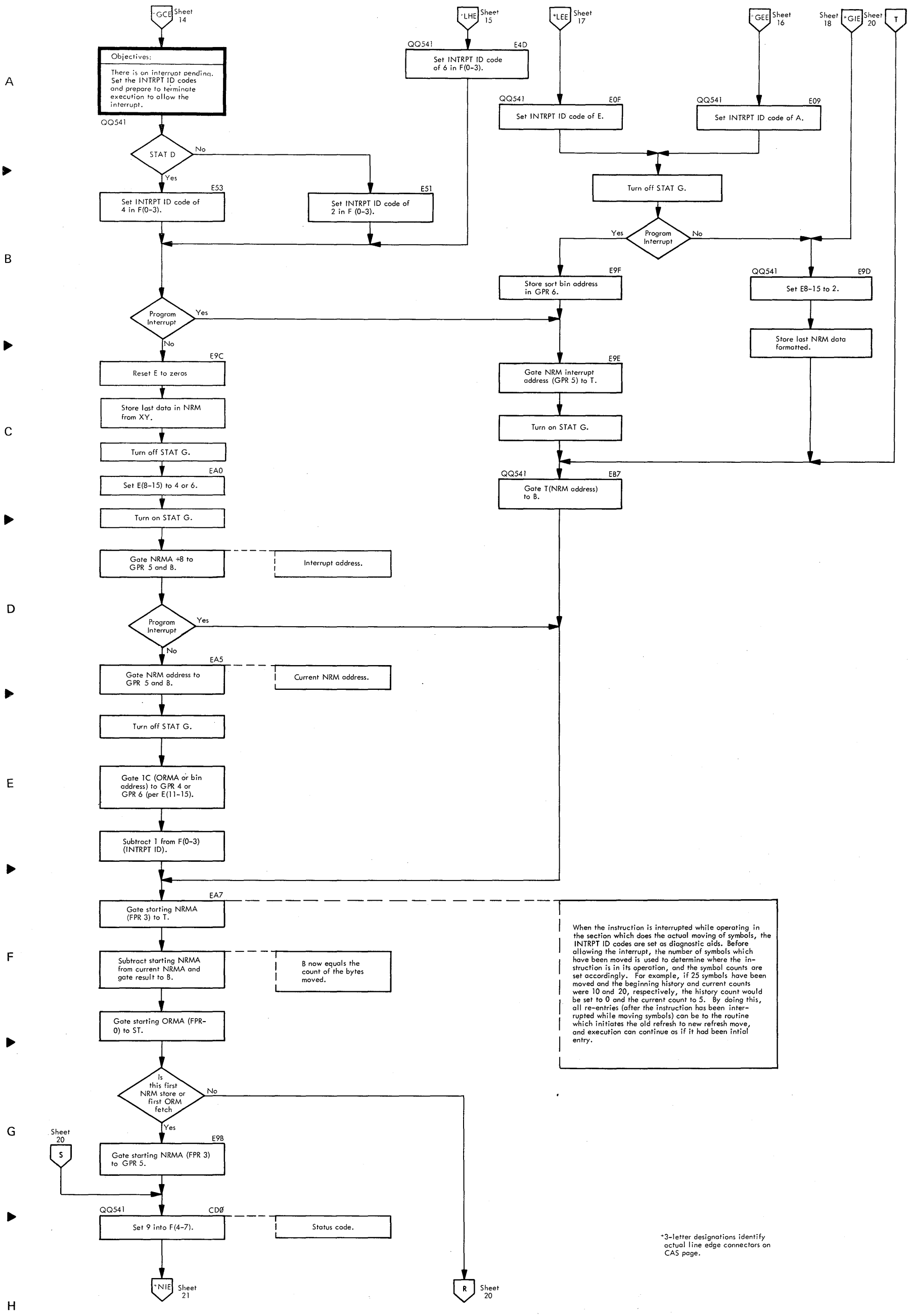


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 19 of 21)

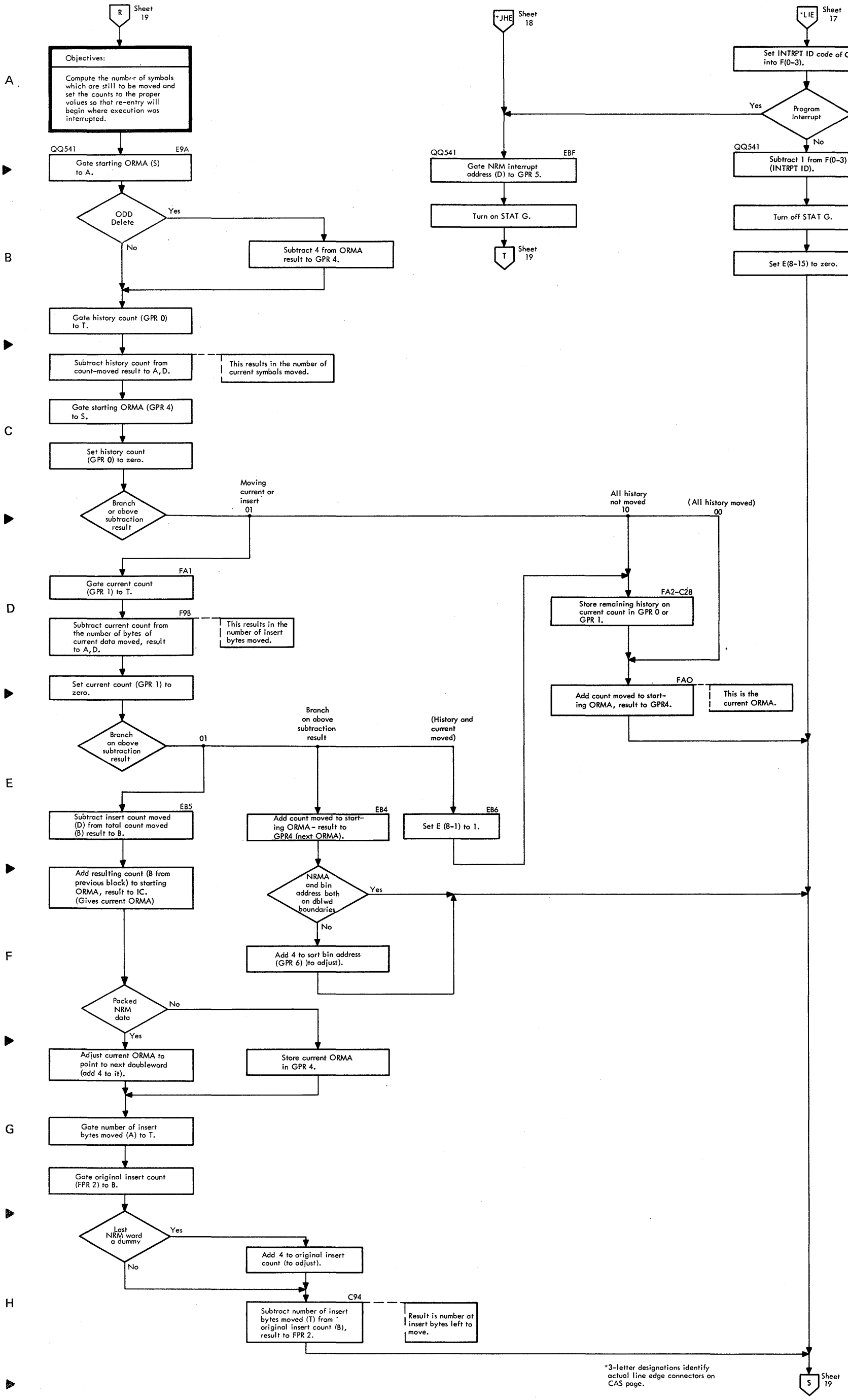


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 20 of 21)

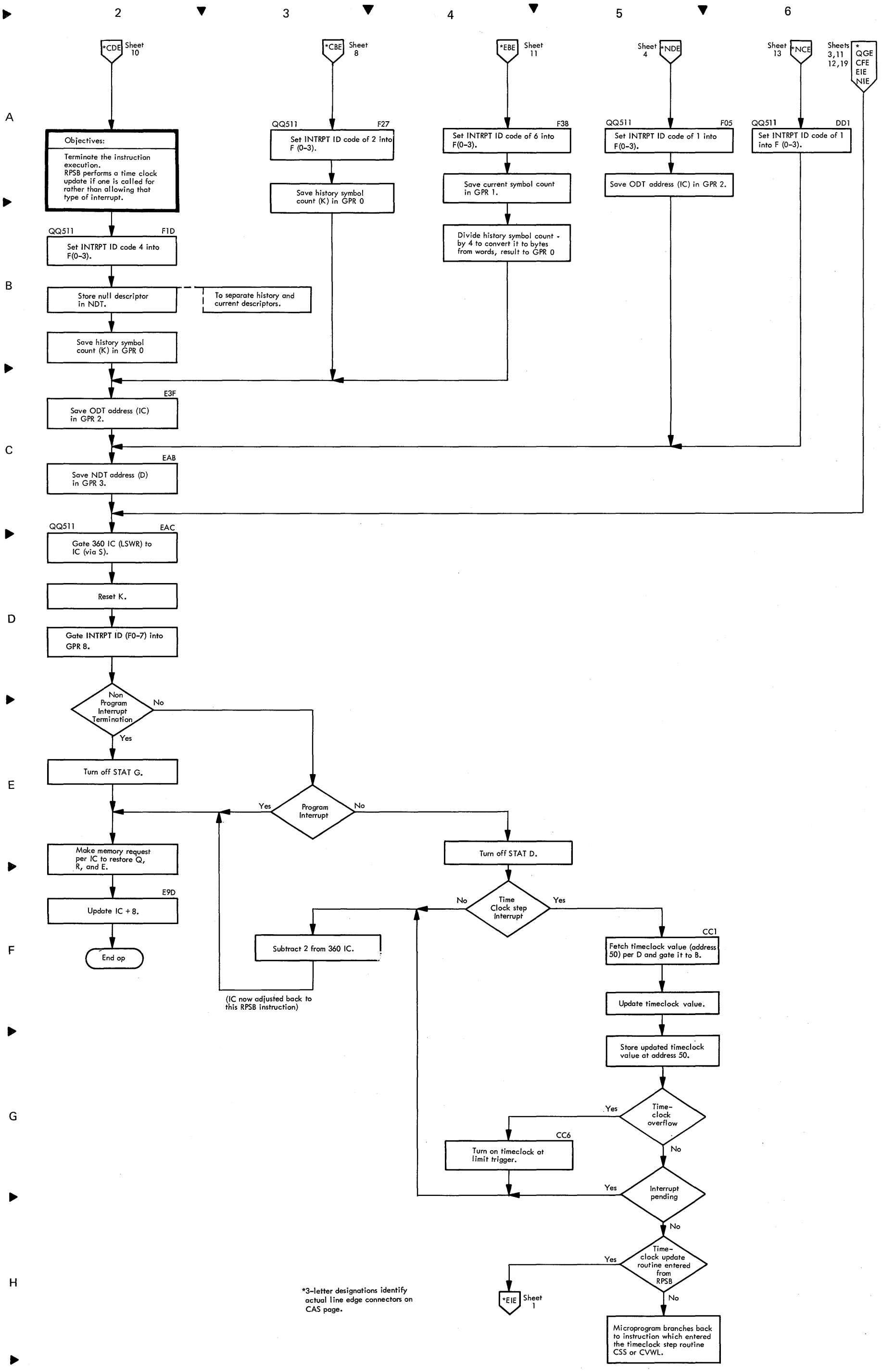
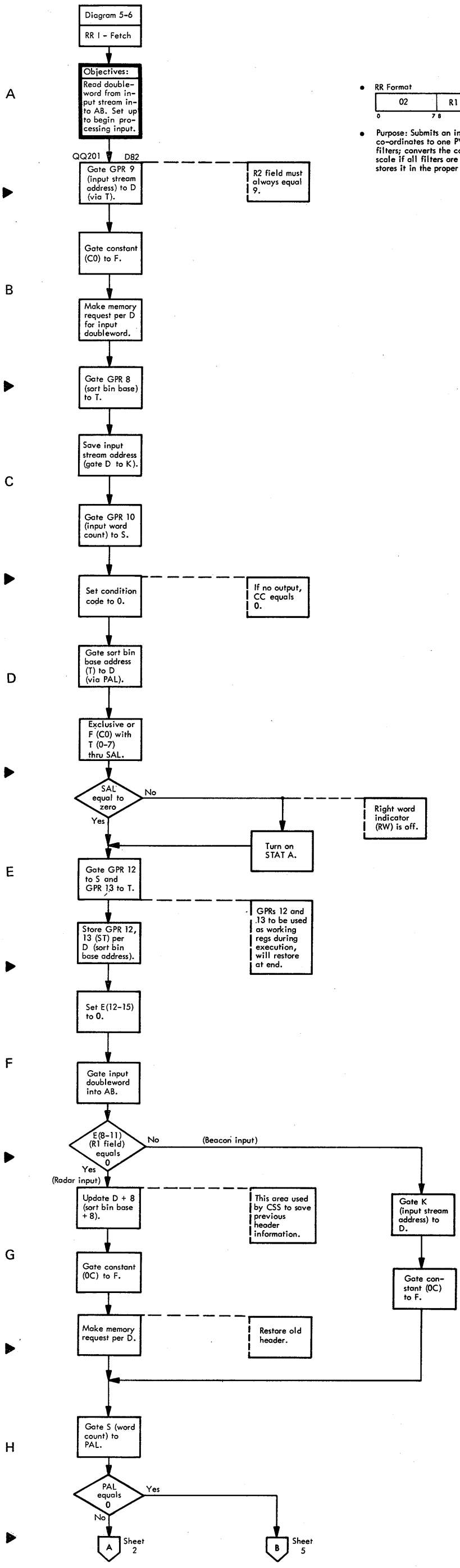


Diagram 5-902. Repack Symbols, RPSB (0F) (Sheet 21 of 21)



- RR Format

- Purpose: Submits an input stream of either Radar or Beacon co-ordinates to one PVD's geographic and sterile area filters; converts the co-ordinates from system scale to PVD scale if all filters are passed; assembles an output word and stores it in the proper sort bin location.

Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 1 of 10)

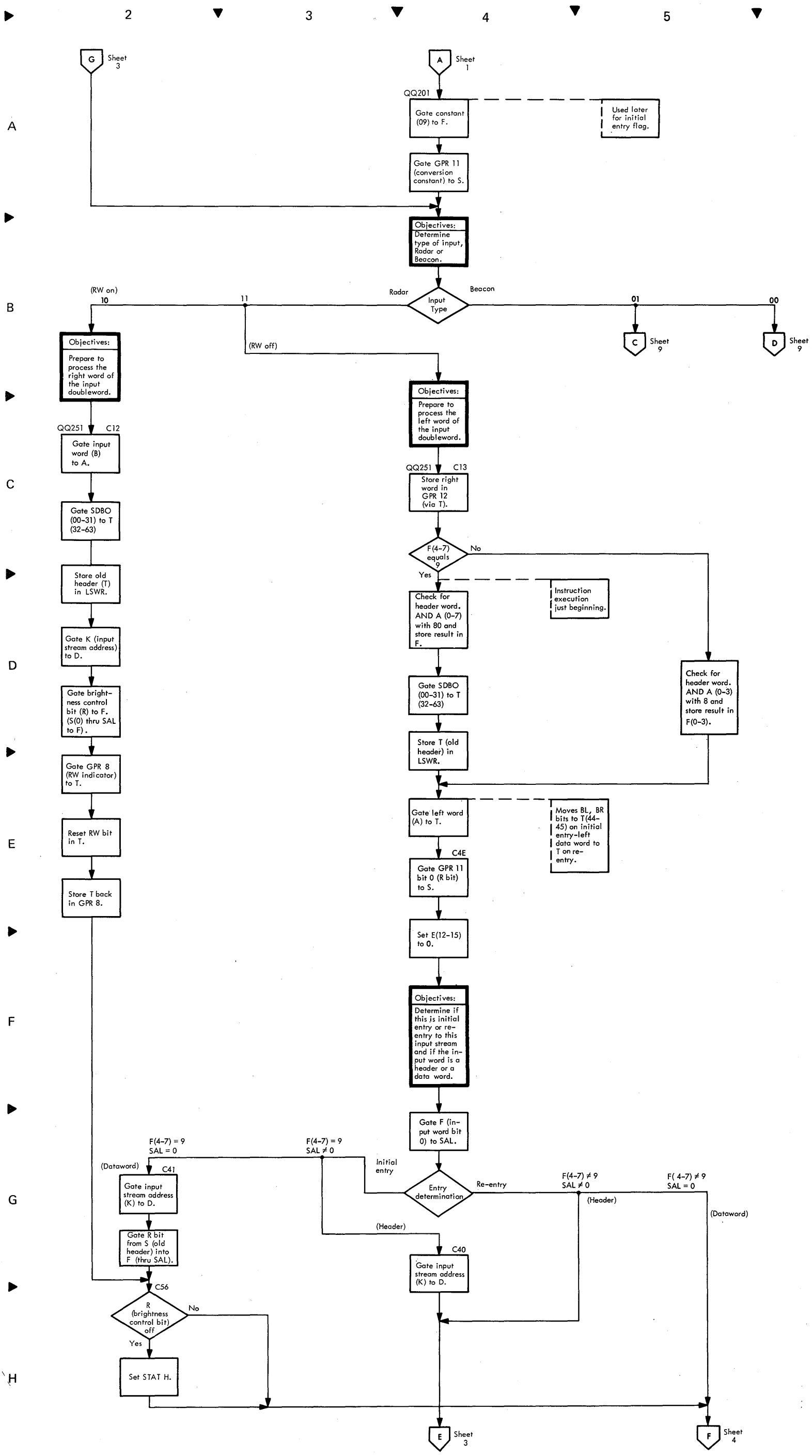


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 2 of 10)

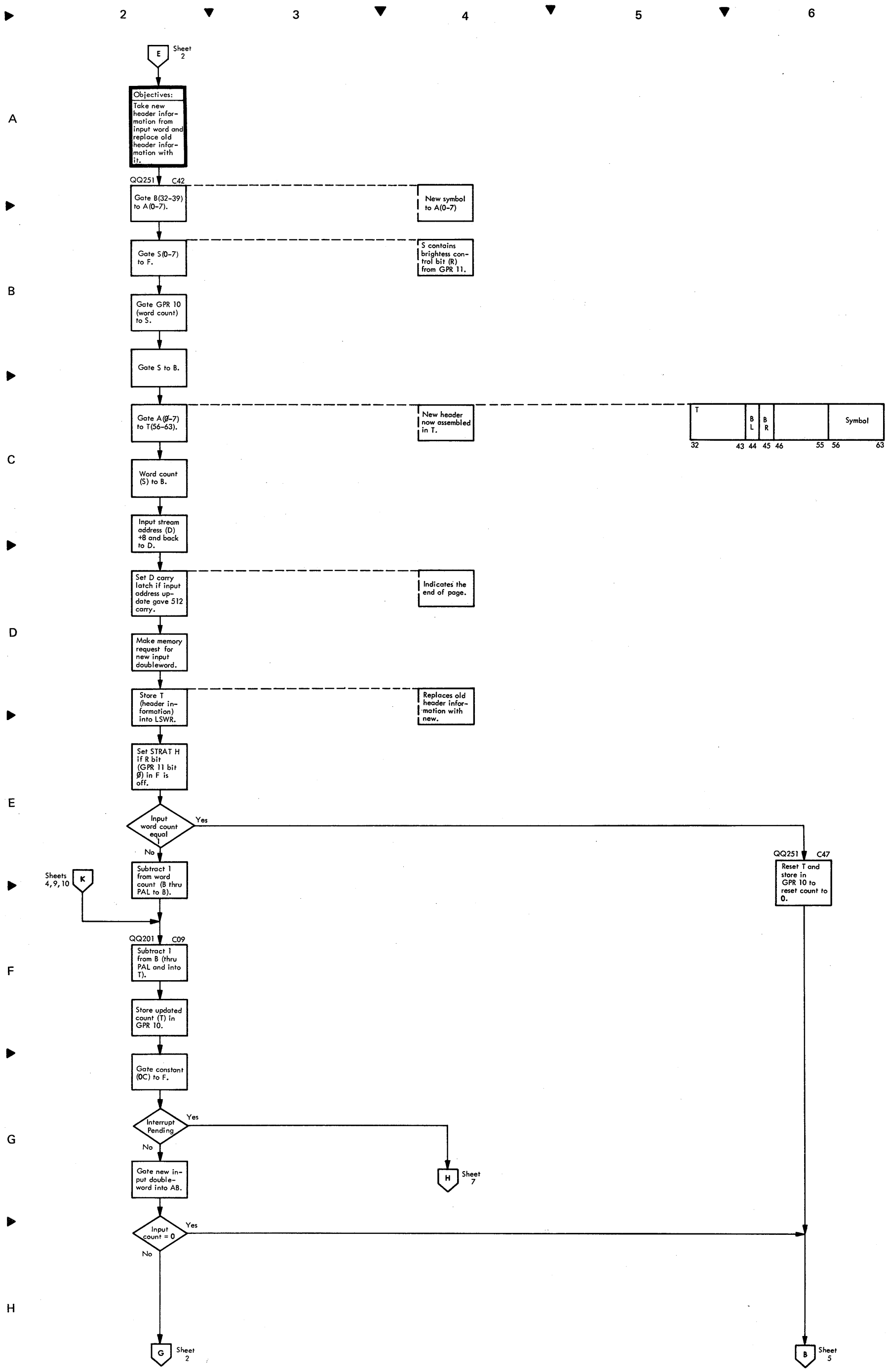


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 3 of 10)

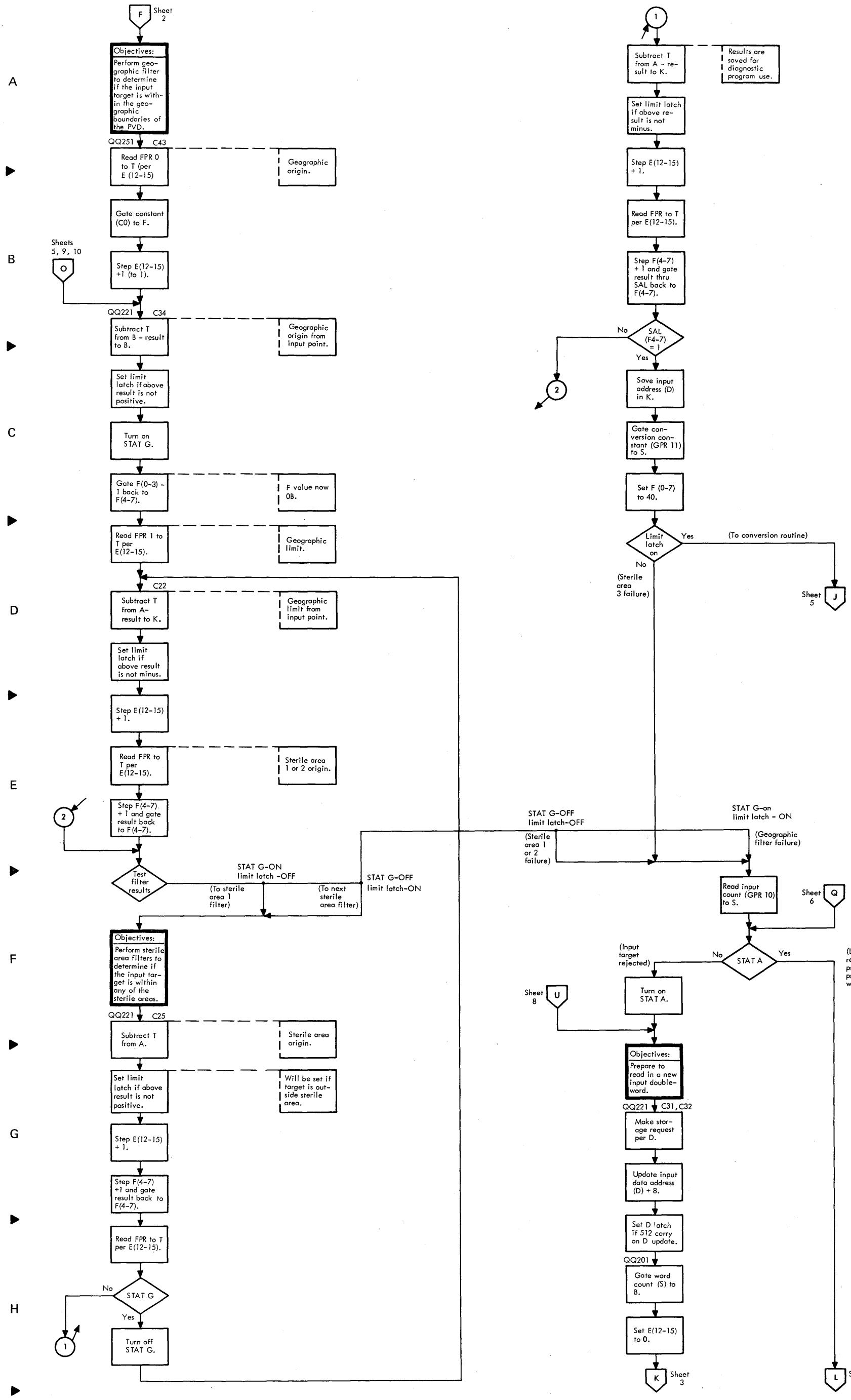


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 4 of 10)

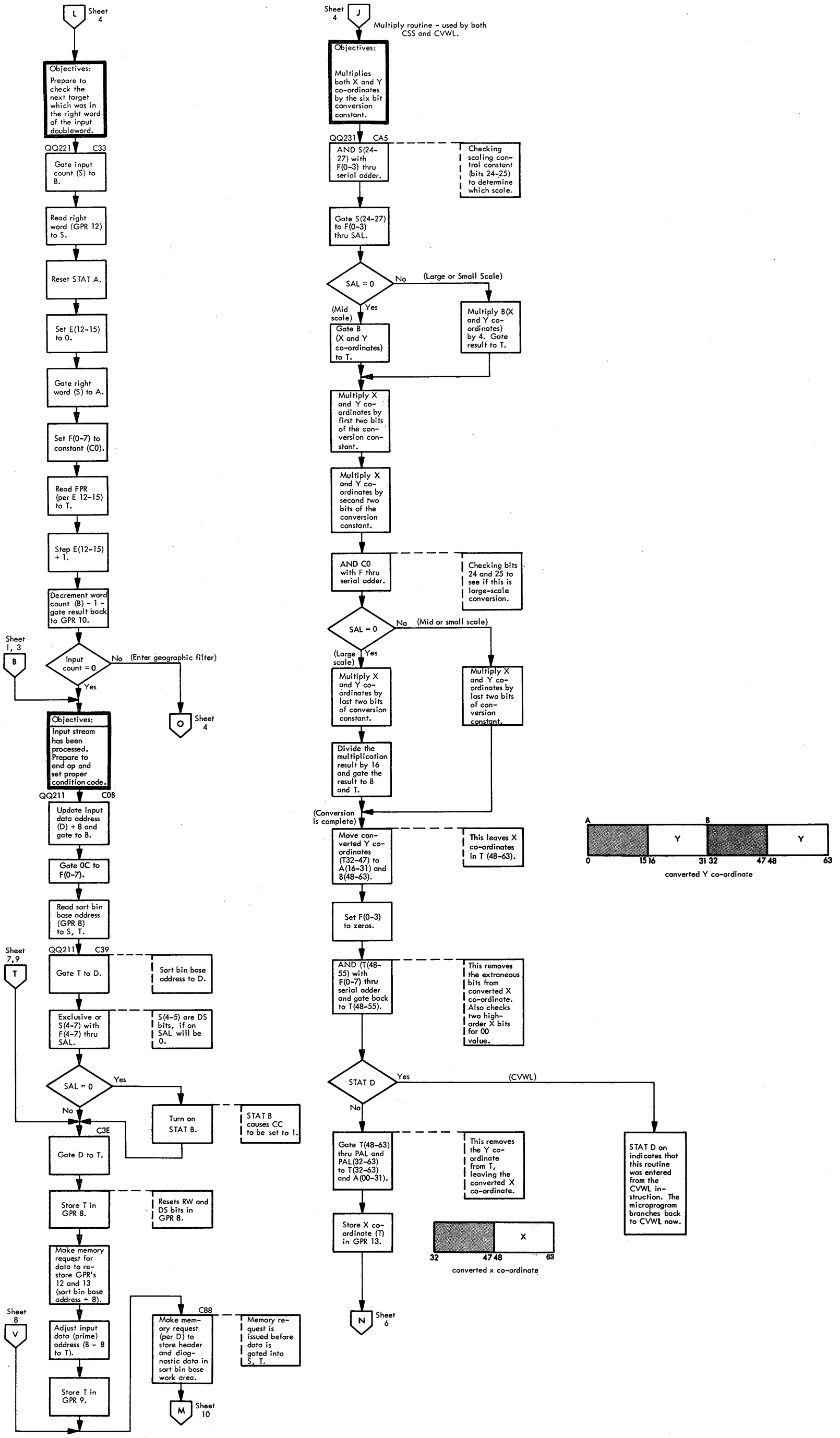


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 5 of 10)

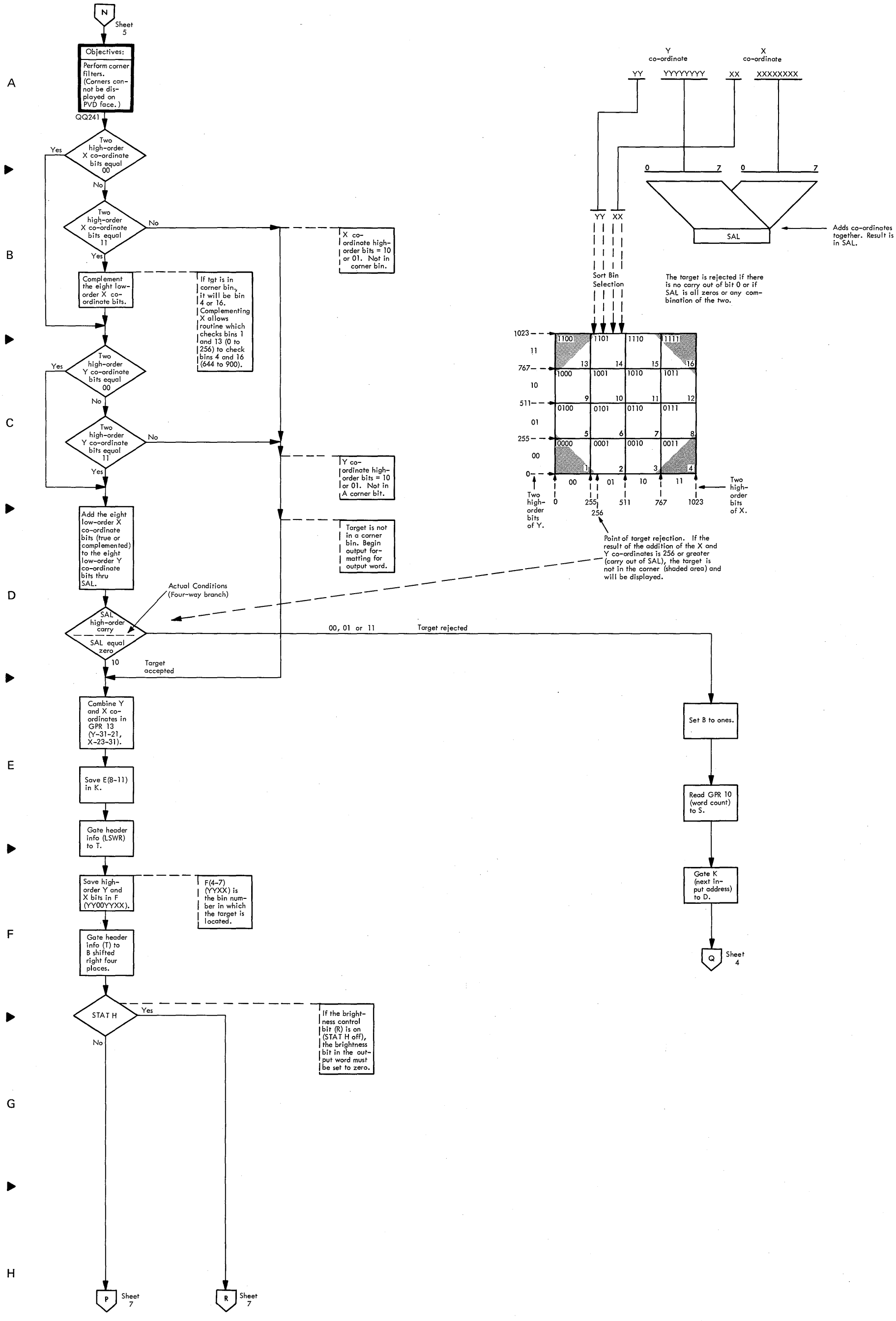


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 6 of 10)

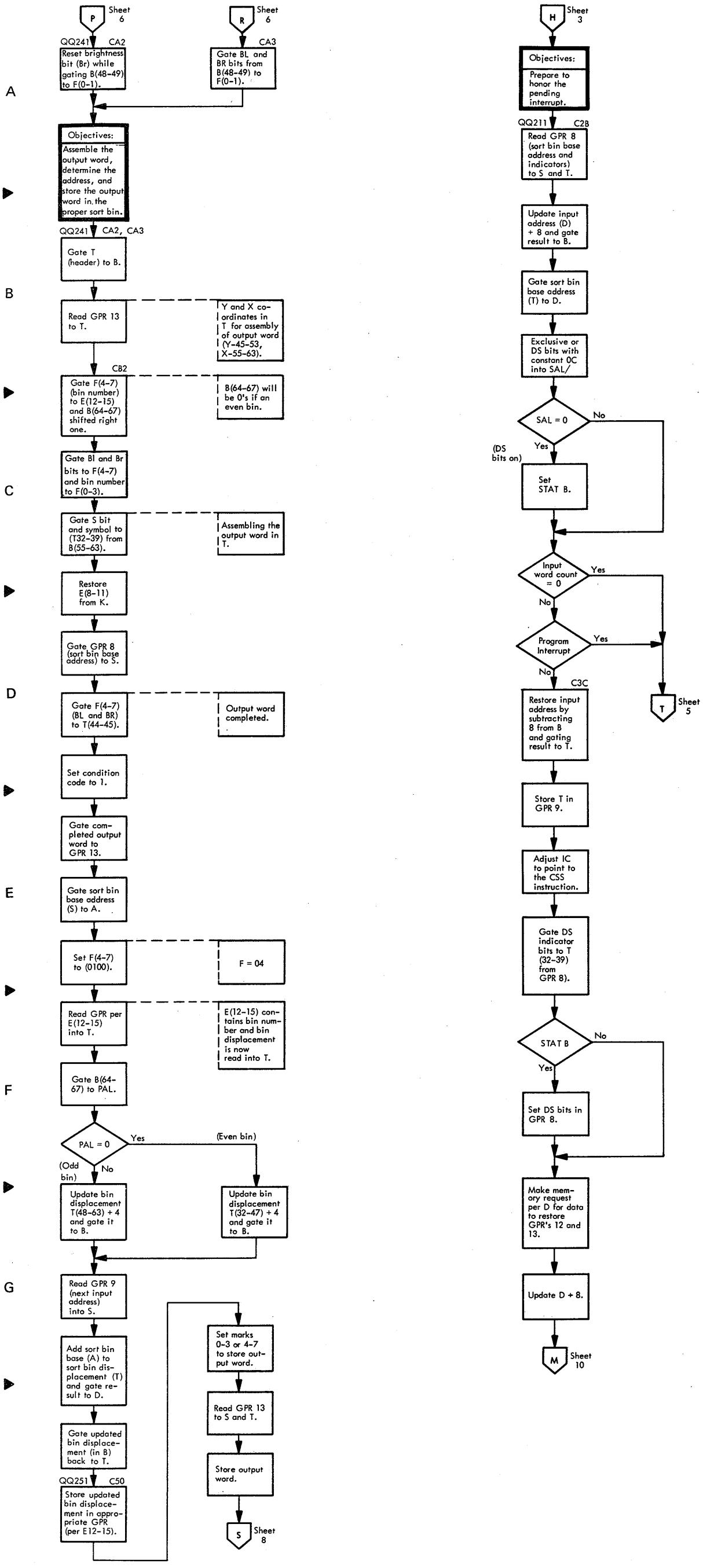


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 7 of 10)

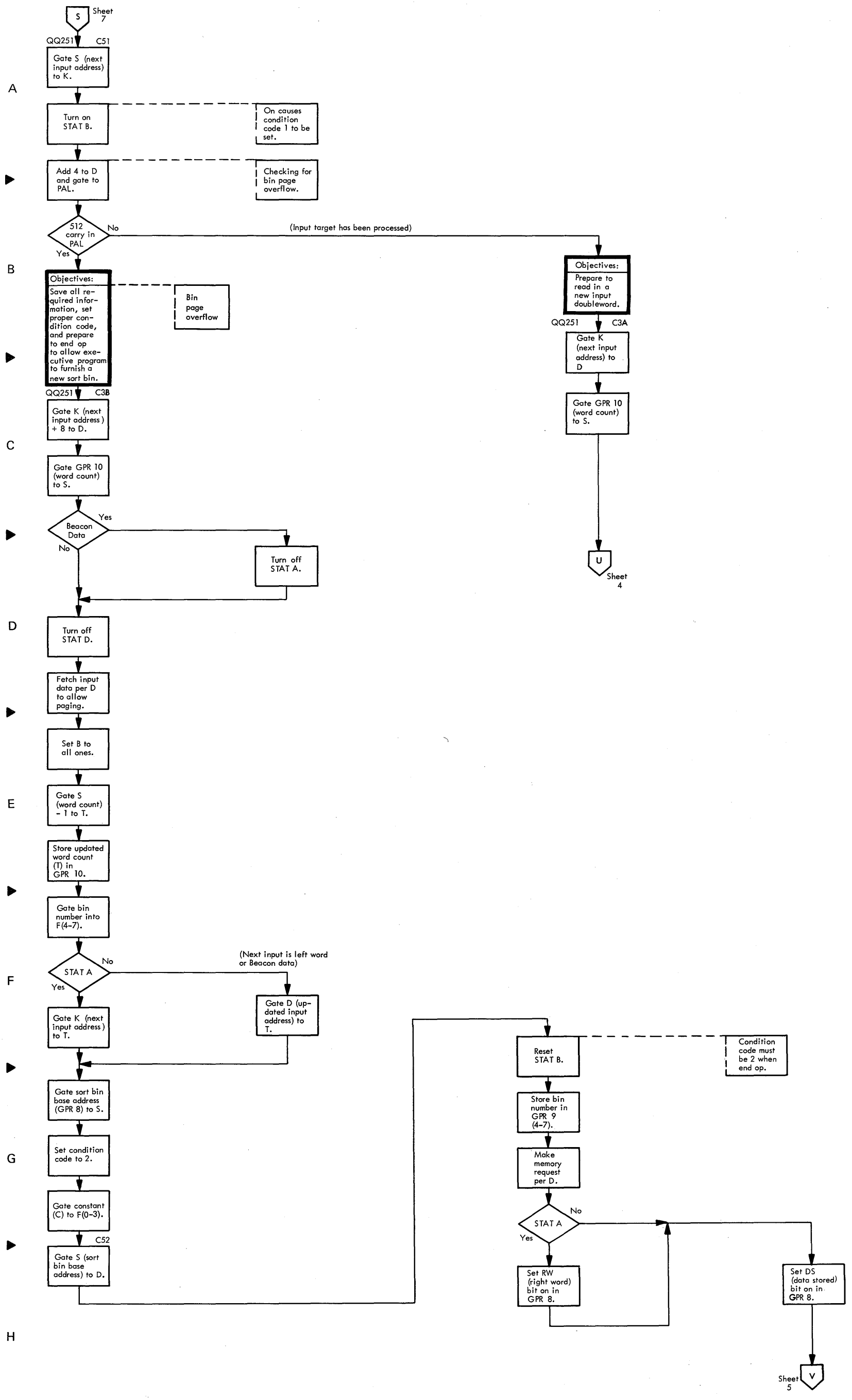


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 8 of 10)

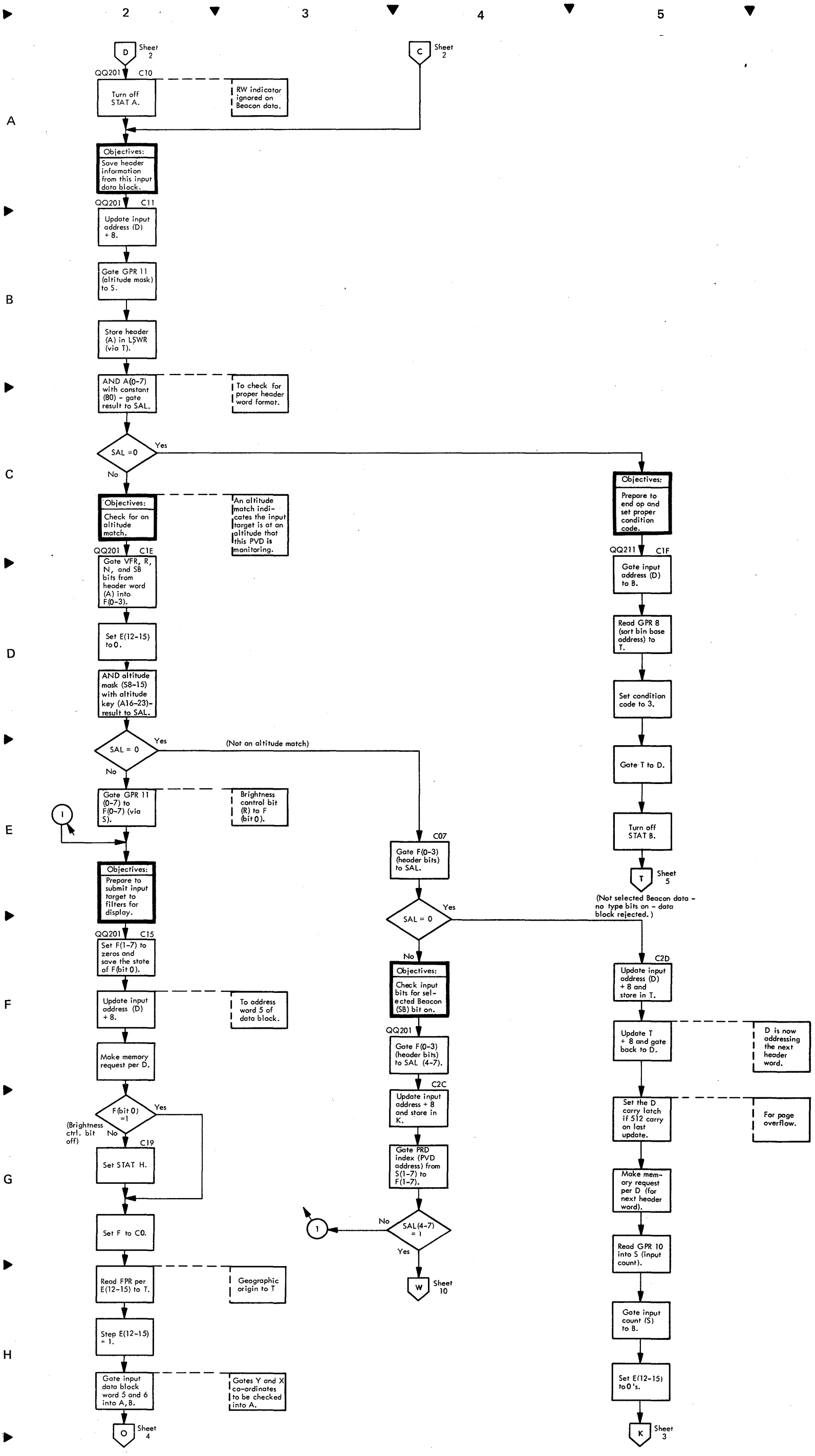


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 9 of 10)

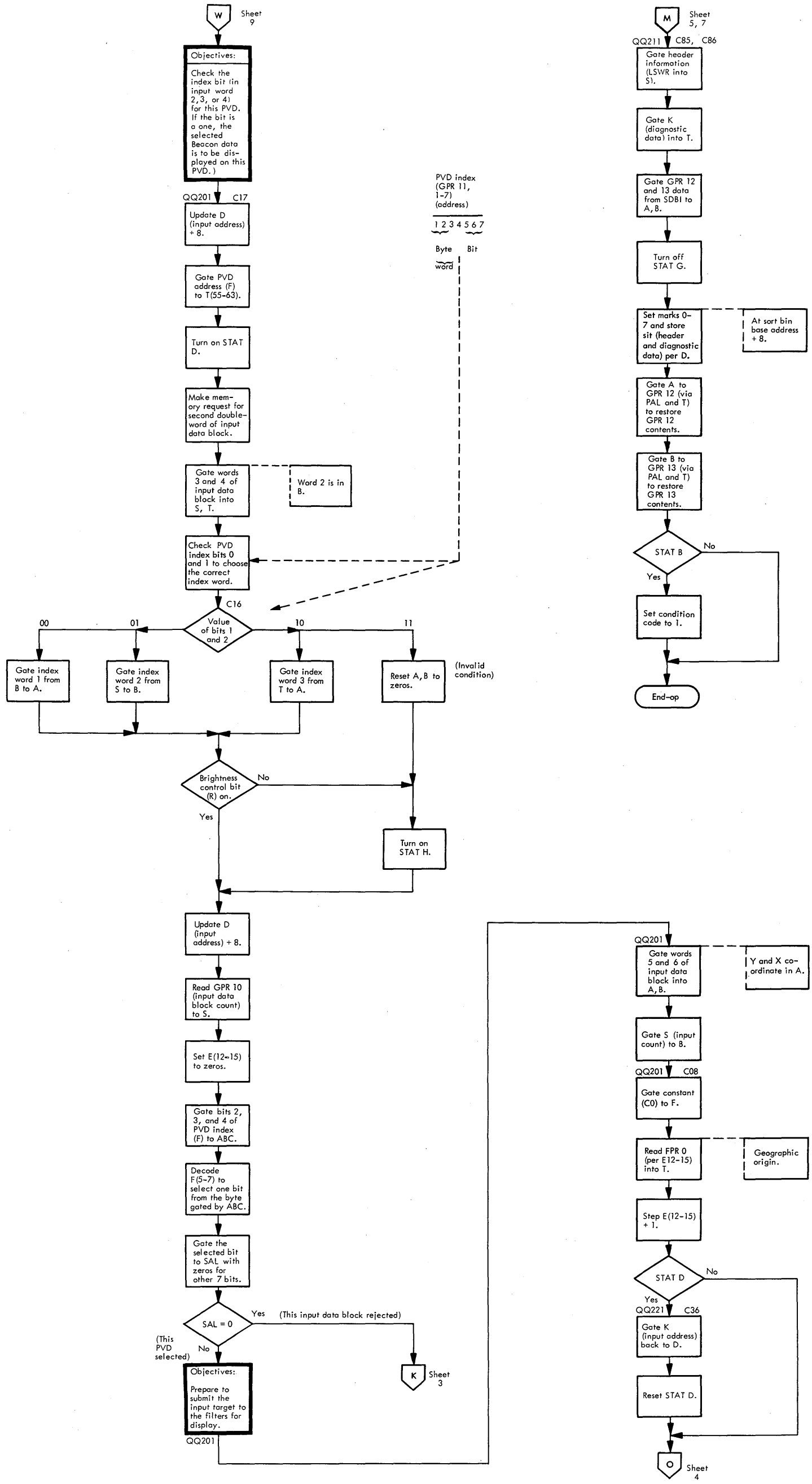


Diagram 5-903. Convert and Sort Symbols, CSS (02) (Sheet 10 of 10)

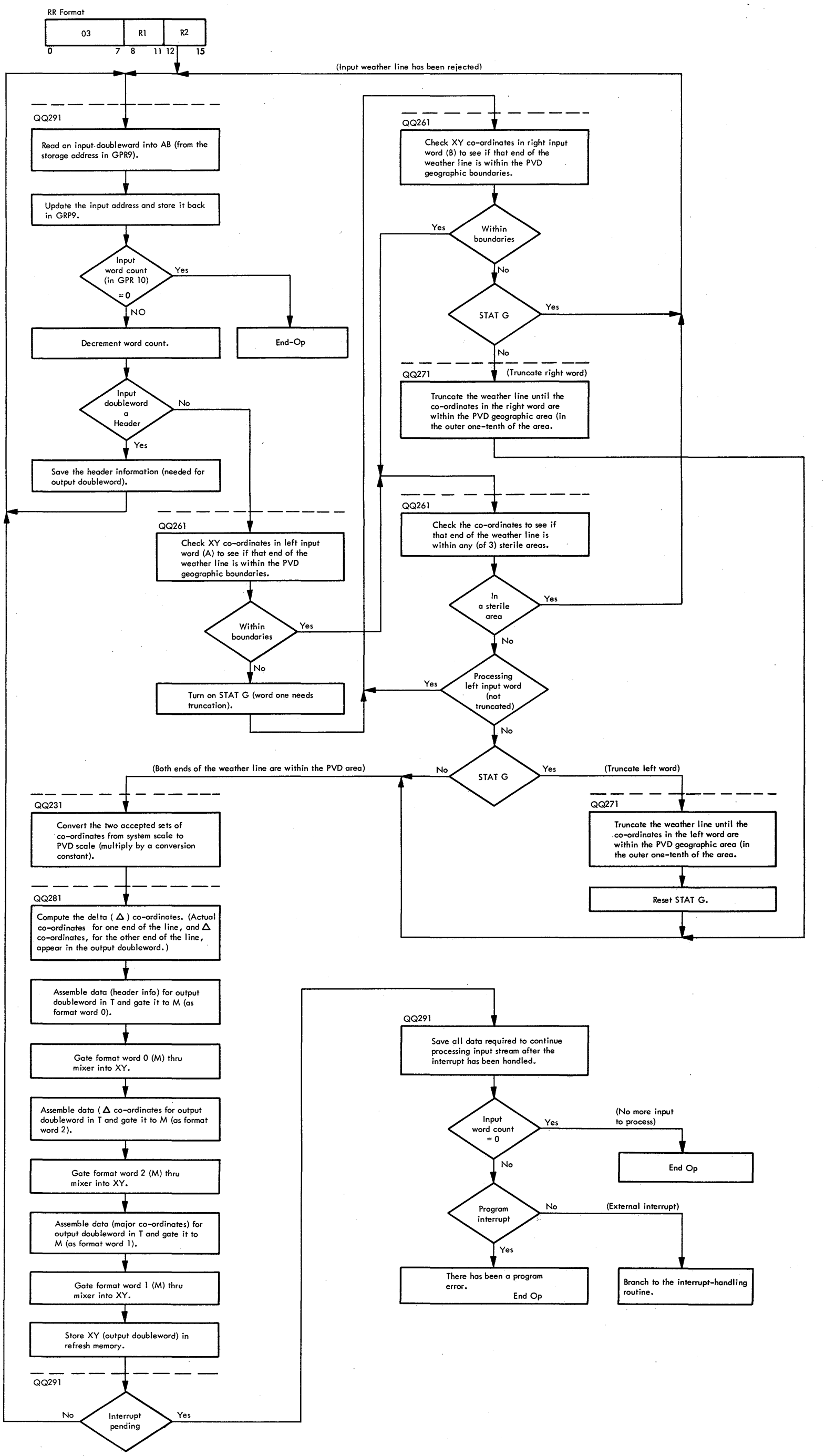
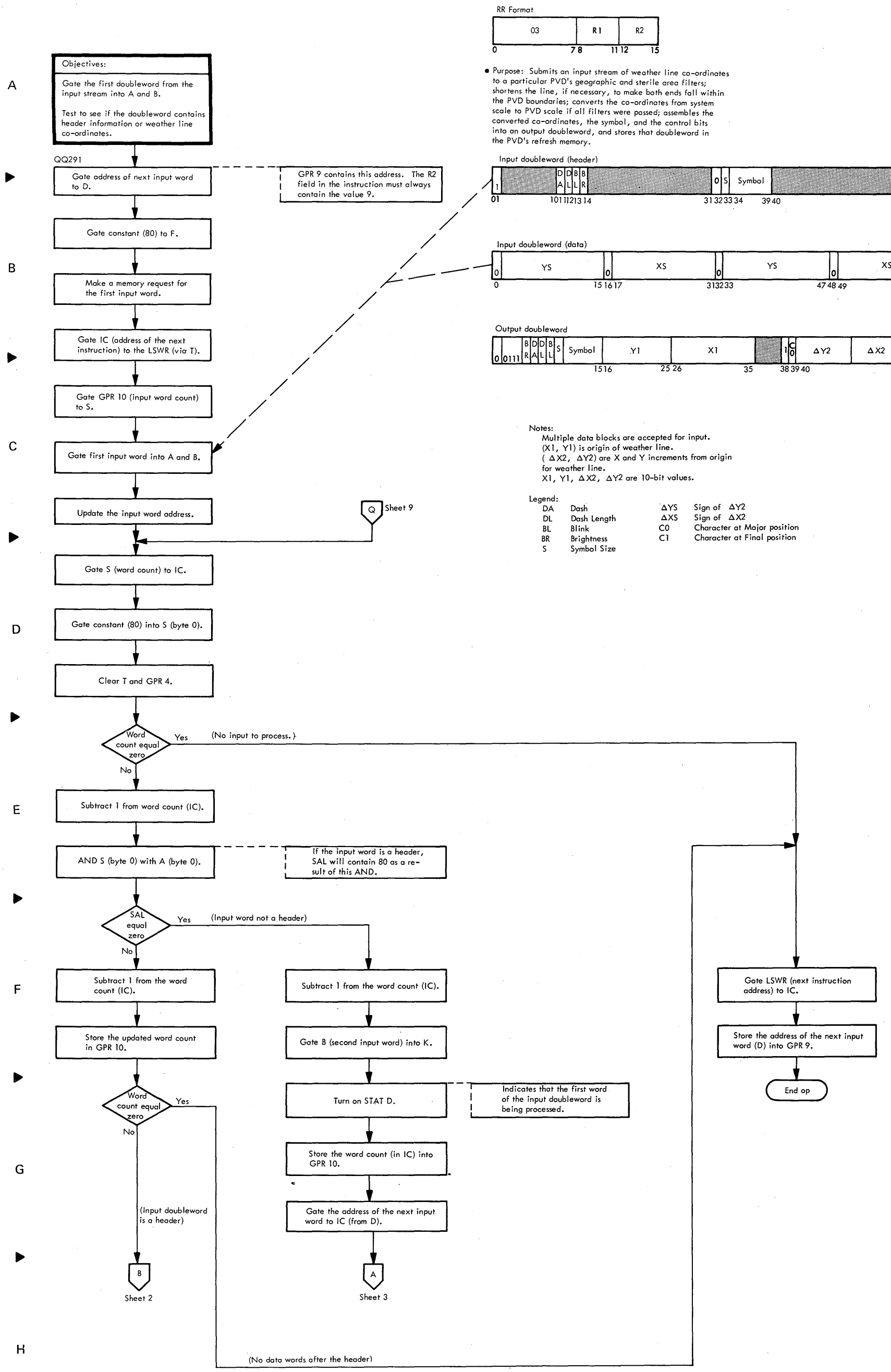


Diagram 5-904. Convert Weather Lines, Simplified Flowchart



Notes:
Multiple data blocks are accepted for input.
(X1, Y1) is origin of weather line.
(ΔX2, ΔY2) are X and Y increments from origin for weather line.
X1, Y1, ΔX2, ΔY2 are 10-bit values.

Legend:

DA	Dash	ΔYS	Sign of ΔY2
DL	Dash Length	ΔXS	Sign of ΔX2
BL	Blink	C0	Character at Major position
BR	Brightness	C1	Character at Final position
S	Symbol Size		

- Purpose: Submits an input stream of weather line co-ordinates to a particular PVD's geographic and sterile area filters; shortens the line, if necessary, to make both ends fall within the PVD boundaries; converts the co-ordinates from system scale to PVD scale if all filters were passed; assembles the converted co-ordinates, the symbol, and the control bits into an output doubleword, and stores that doubleword in the PVD's refresh memory.

Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 1 of 9)

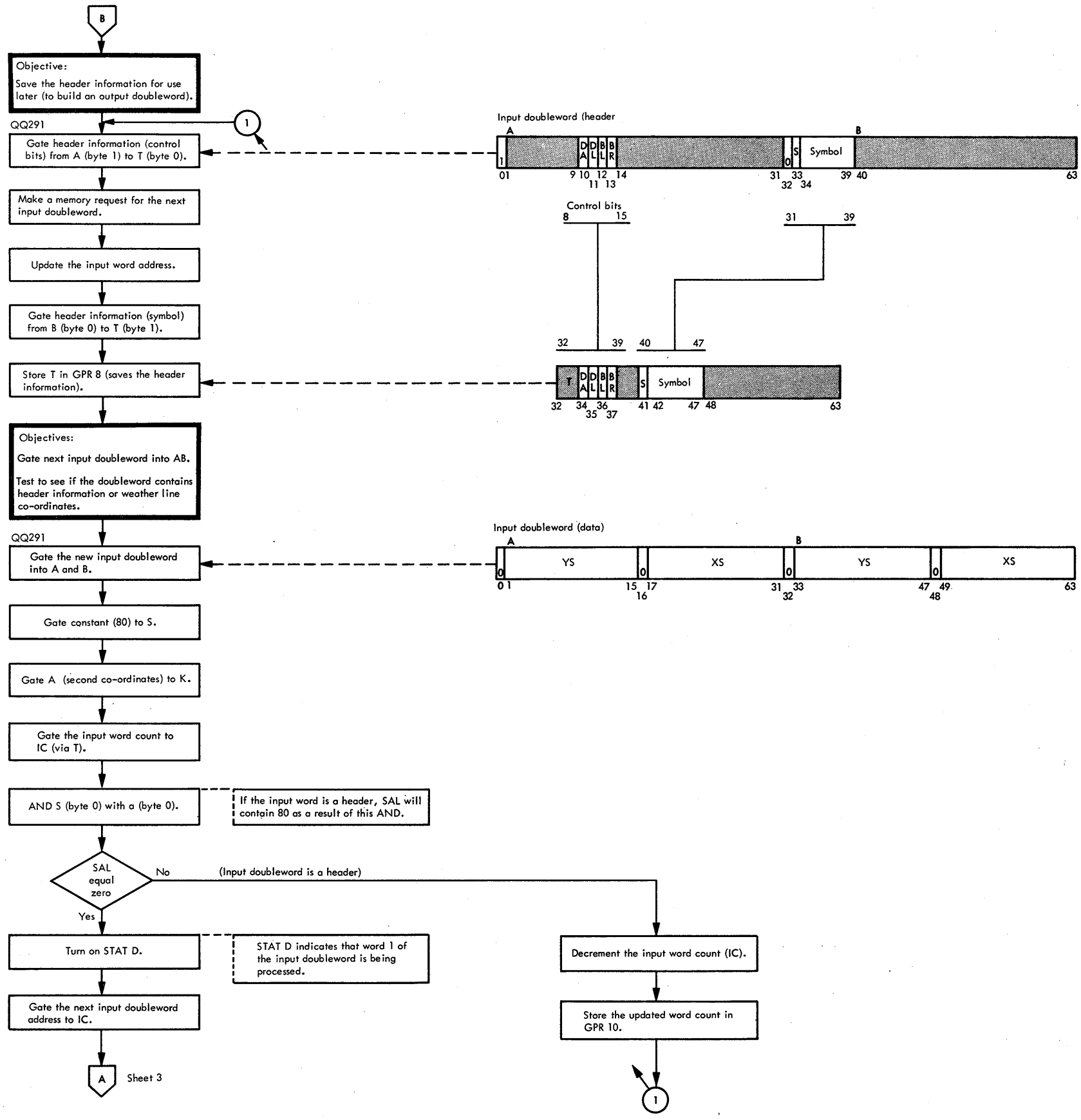


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 2 of 9)

A

Objectives:

Perform the geographic filter (determine if the input point is within the PVD geographic area).
Perform the first and second sterile area filters (determine if the input point is within a sterile area on the PVD screen).

QQ261

Gate the PVD geographic origin co-ordinates (in FPR0) to T.

B

Subtract T (origin) from the first input co-ordinates (in A) and put the result in B.

If the co-ordinates do not require truncation, this result becomes the major or secondary position.

Set the Limit Latch if the subtraction result is not positive.

Gate the PVD geographic Limit (FPR1) to T.

C

Subtract the geographic Limit (in T) from the first co-ordinates (in A).

Set the Limit Latch if the subtraction result is not minus.

Limit latch on

Yes — (This point failed geographic filter)

No — (Sterile area filter)

D

Gate the first sterile area origin (FPR 2) to T.

If either end of a weather line (original or truncated point) is within a sterile area, the line is rejected.

Subtract T (origin) from the first input co-ordinates (in A).

Set the limit latch if the subtraction result is not positive.

E

Gate the first sterile area limit (FPR 3) to T.

Subtract T (Limit) from the first input co-ordinates (in A).

Set the Limit latch if the subtraction result is not minus.

F

Limit latch on

Yes — (This point failed the first sterile area filter.)

No — (Sterile area filter)

Gate the second sterile area origin (FPR 4) to T.

Subtract T (origin) from the first input co-ordinates (in A).

Set the Limit latch if the subtraction result is not positive.

G

Gate the second sterile area Limit (FPR 5) to T.

Subtract T (Limit) from the input co-ordinates (in A).

Set the Limit latch if the subtraction result is not minus.

H

F

Sheet 4

E

Sheet 4

Set 2 into GPR 4.

Indicates to later routine that truncation has been performed.

STATD

No — (Word two is being processed and has failed the geographic filter.)

Yes — (Input word 1 is being processed.)

D

Sheet 4

Objectives:

Save the first word values (for possible truncation later).
Set up to gate the second word co-ordinates thru the filters.

QQ261

STAT G indicates that input word 1 failed the geographic filter and requires truncation.

Reset STAT D and turn on STAT G.

Store first input co-ordinates in GRP 0.

Store the first subtraction result (in B) in GPR 1.

Gate K (second word of input doubleword) to A.

1

Two sets of co-ordinates are used to describe the geographic boundaries of a PVD. Two sets are also used to describe each sterile area within a PVD's geographic boundaries. The set of co-ordinates which describe the lowest left point of an area will be referred to as the geographic origin. The other set, describing the uppermost right point of an area, will be called the geographic limit.

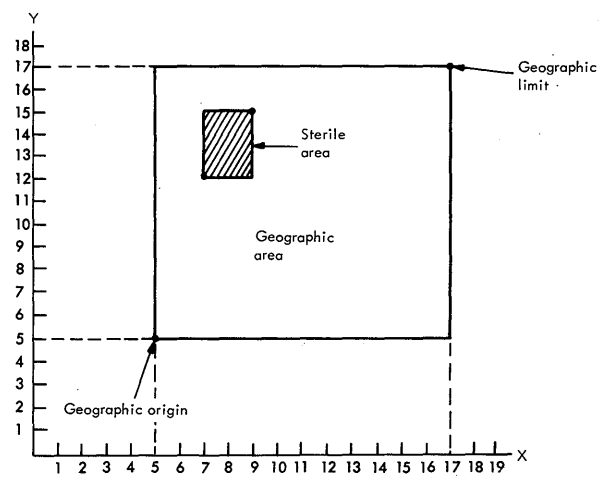


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 3 of 9)

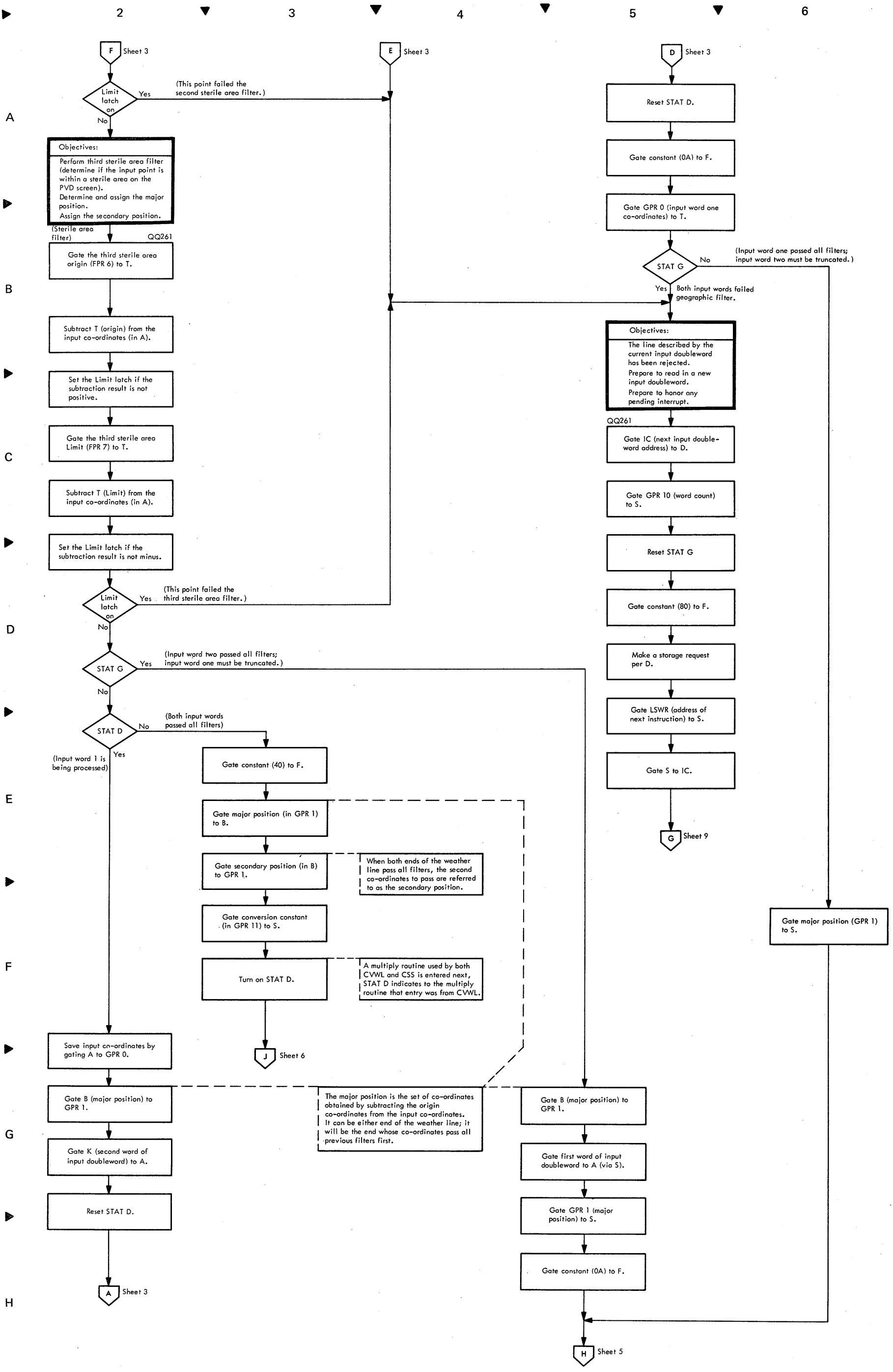


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 4 of 9)

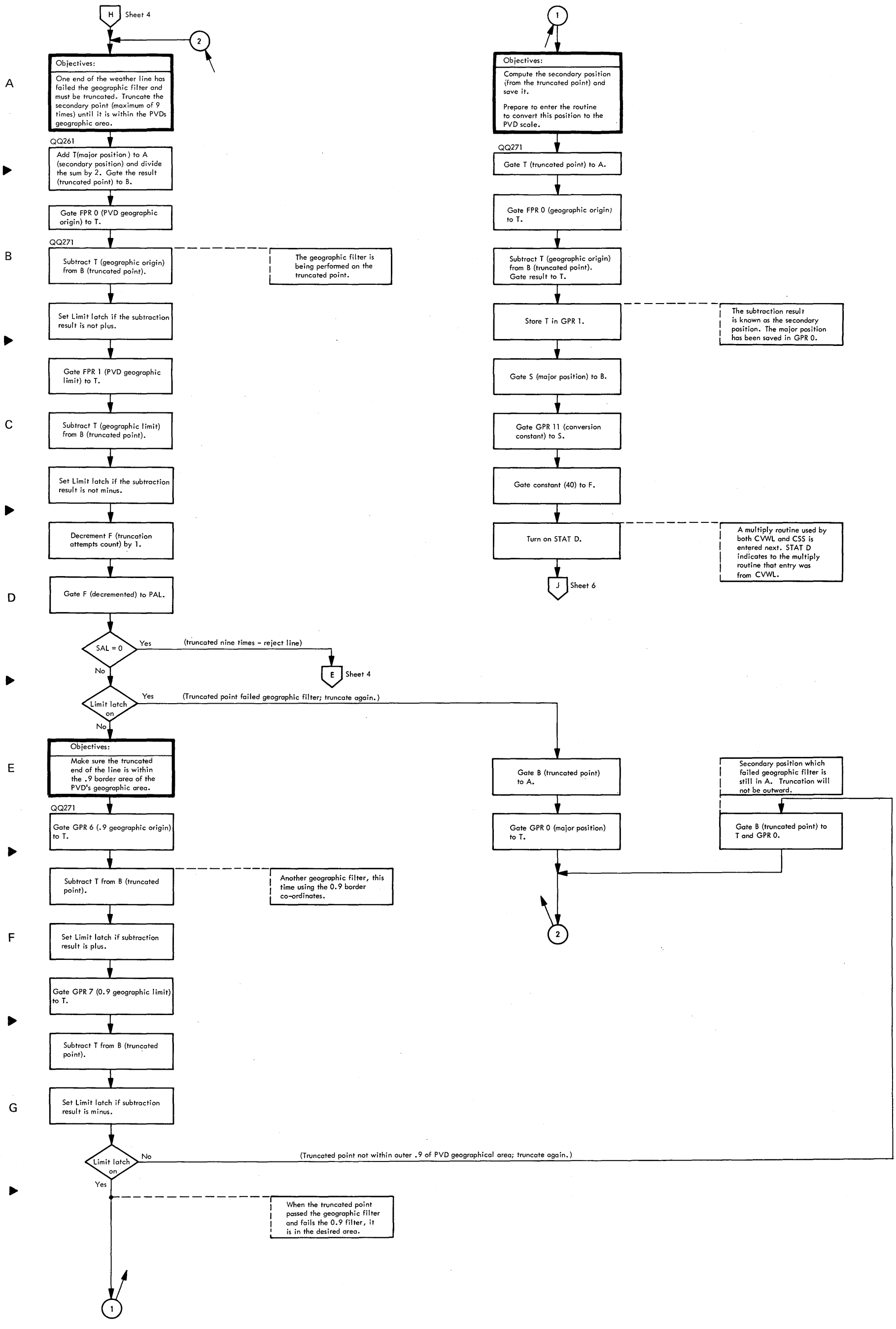


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 5 of 9)

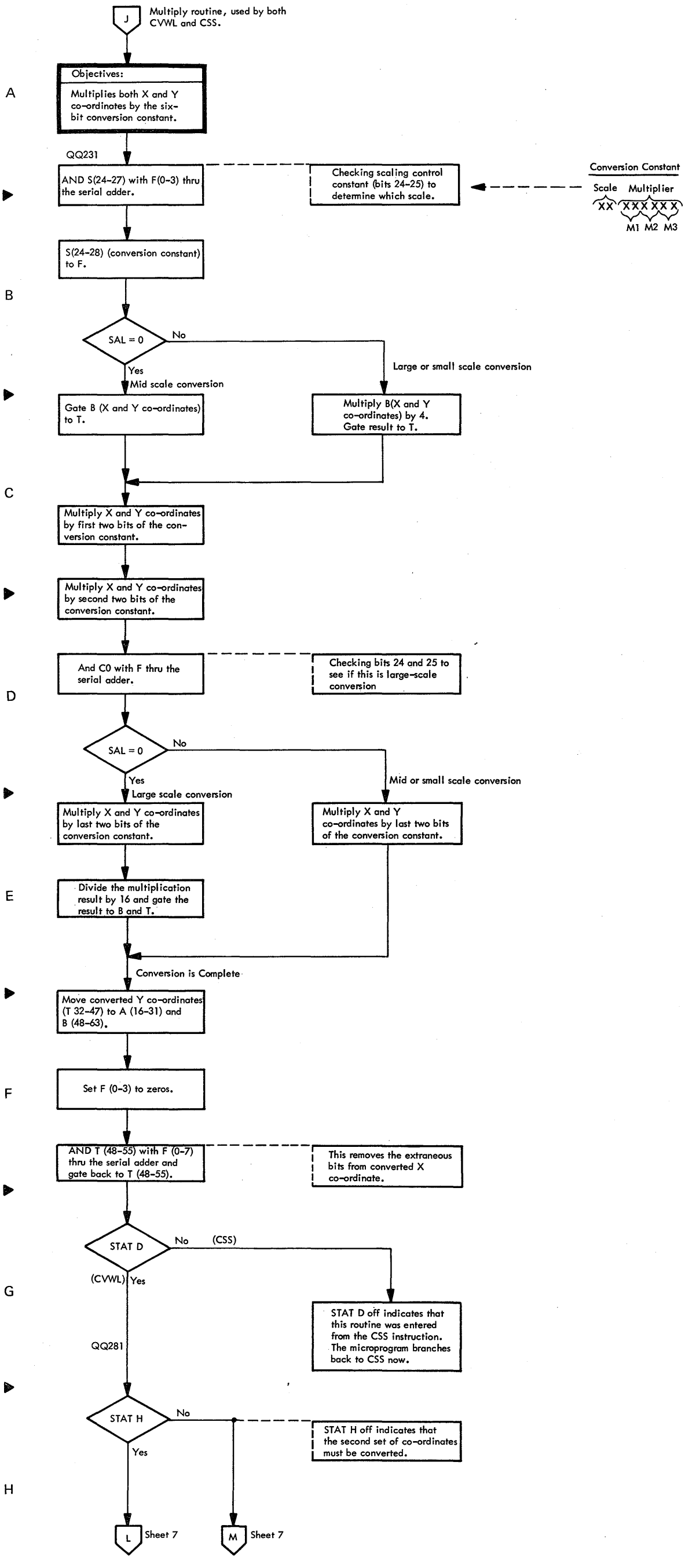


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 6 of 9)

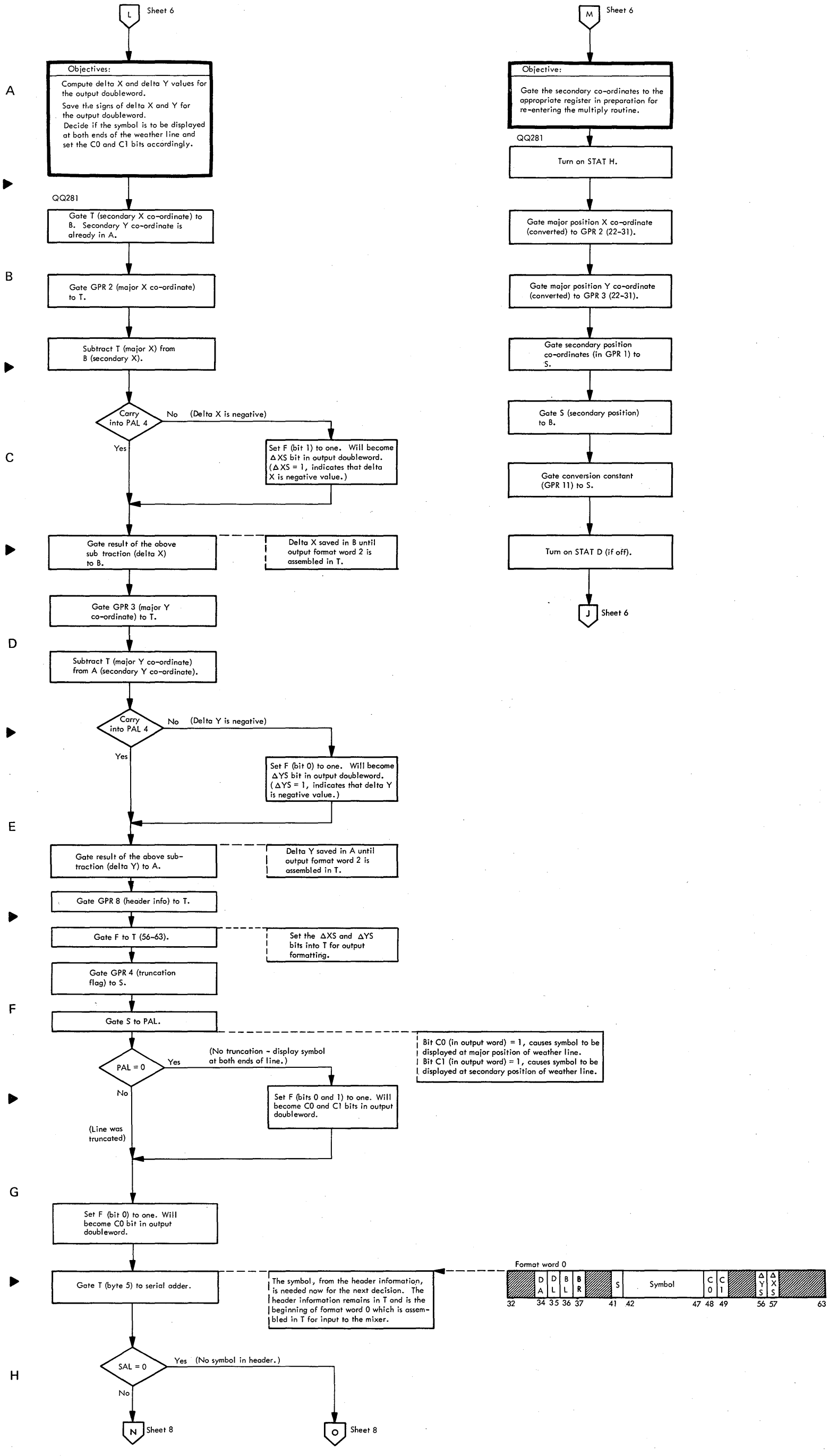


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 7 of 9)

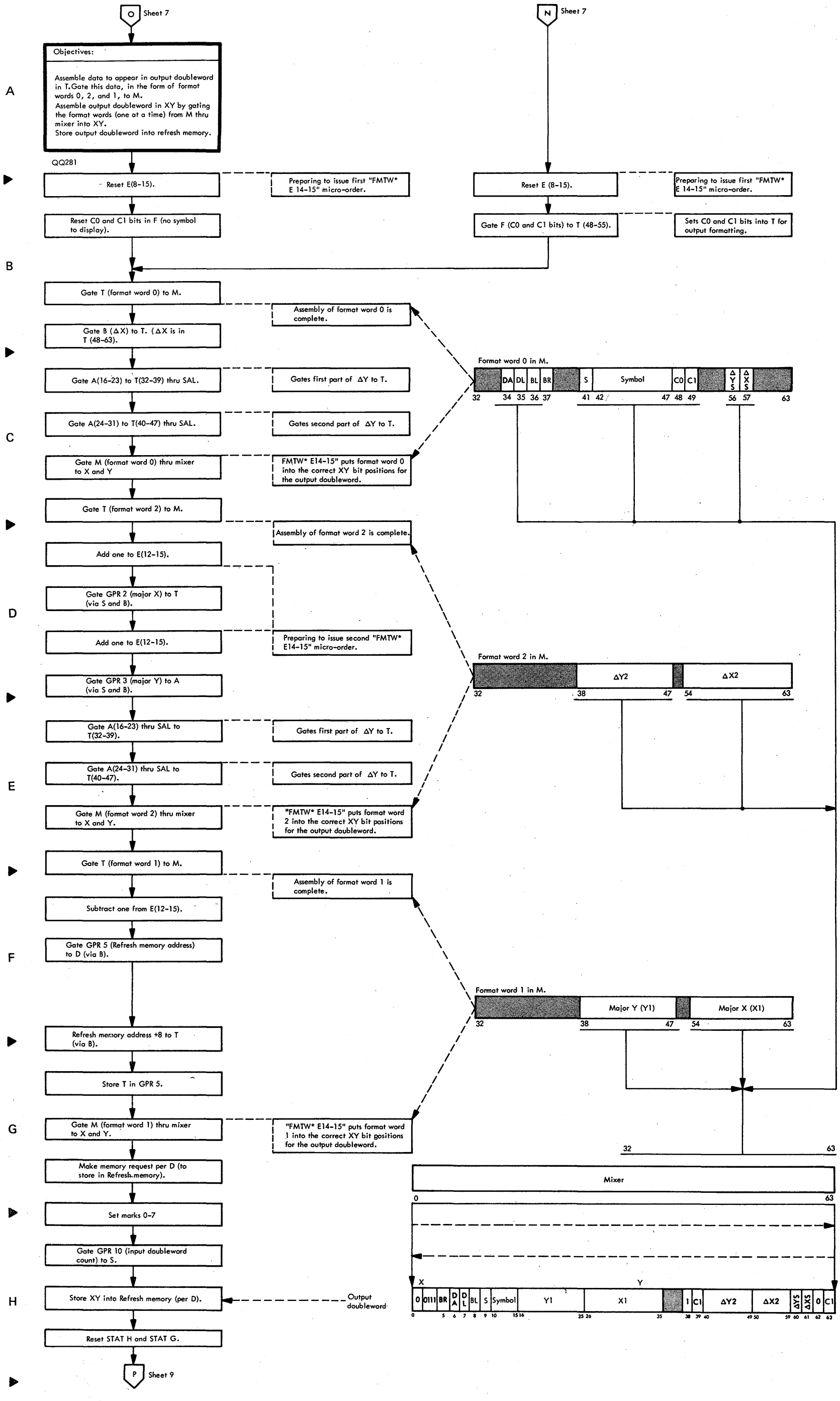


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 8 of 9)

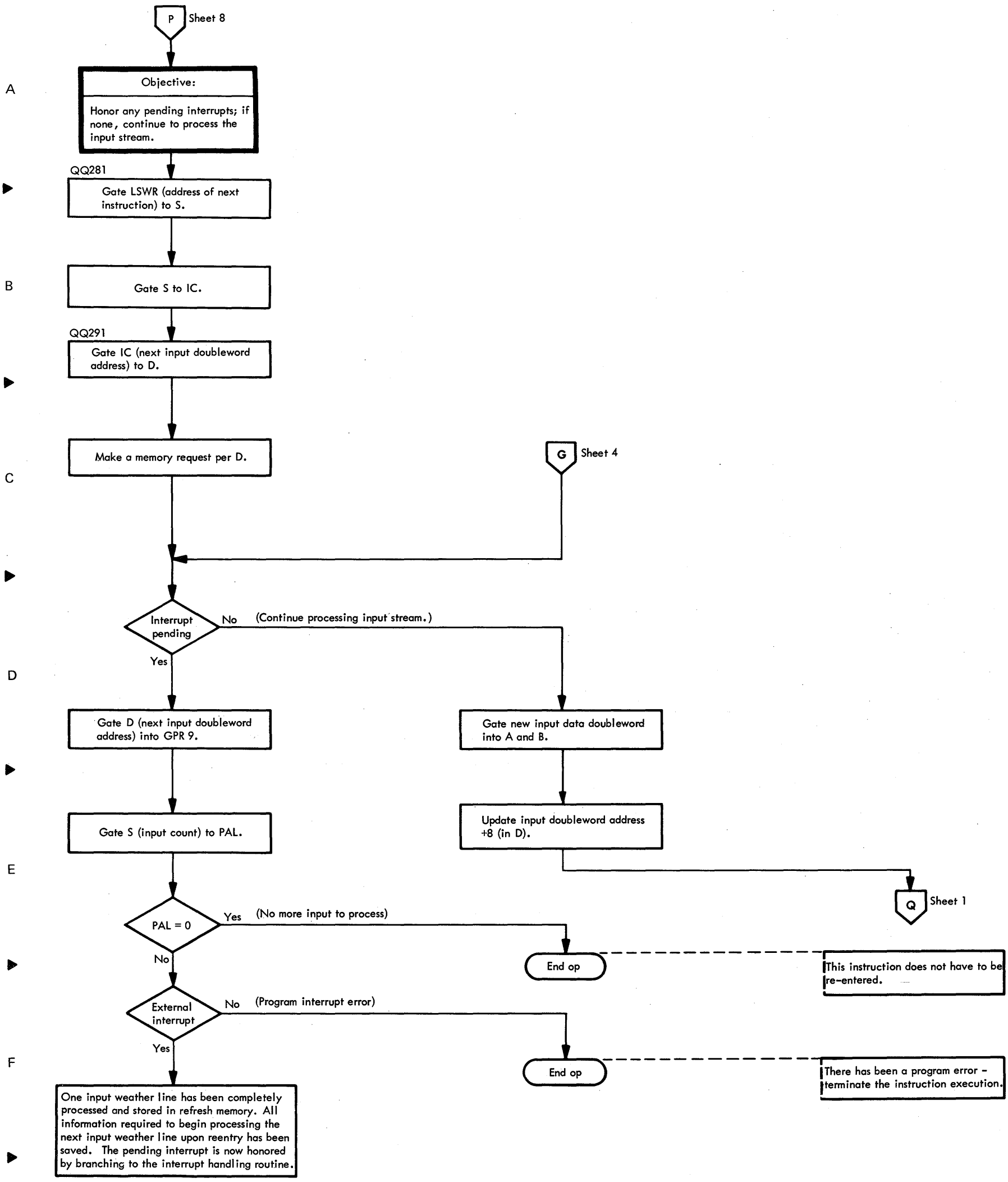


Diagram 5-905. Convert Weather Lines, CVWL (03) (Sheet 9 of 9)

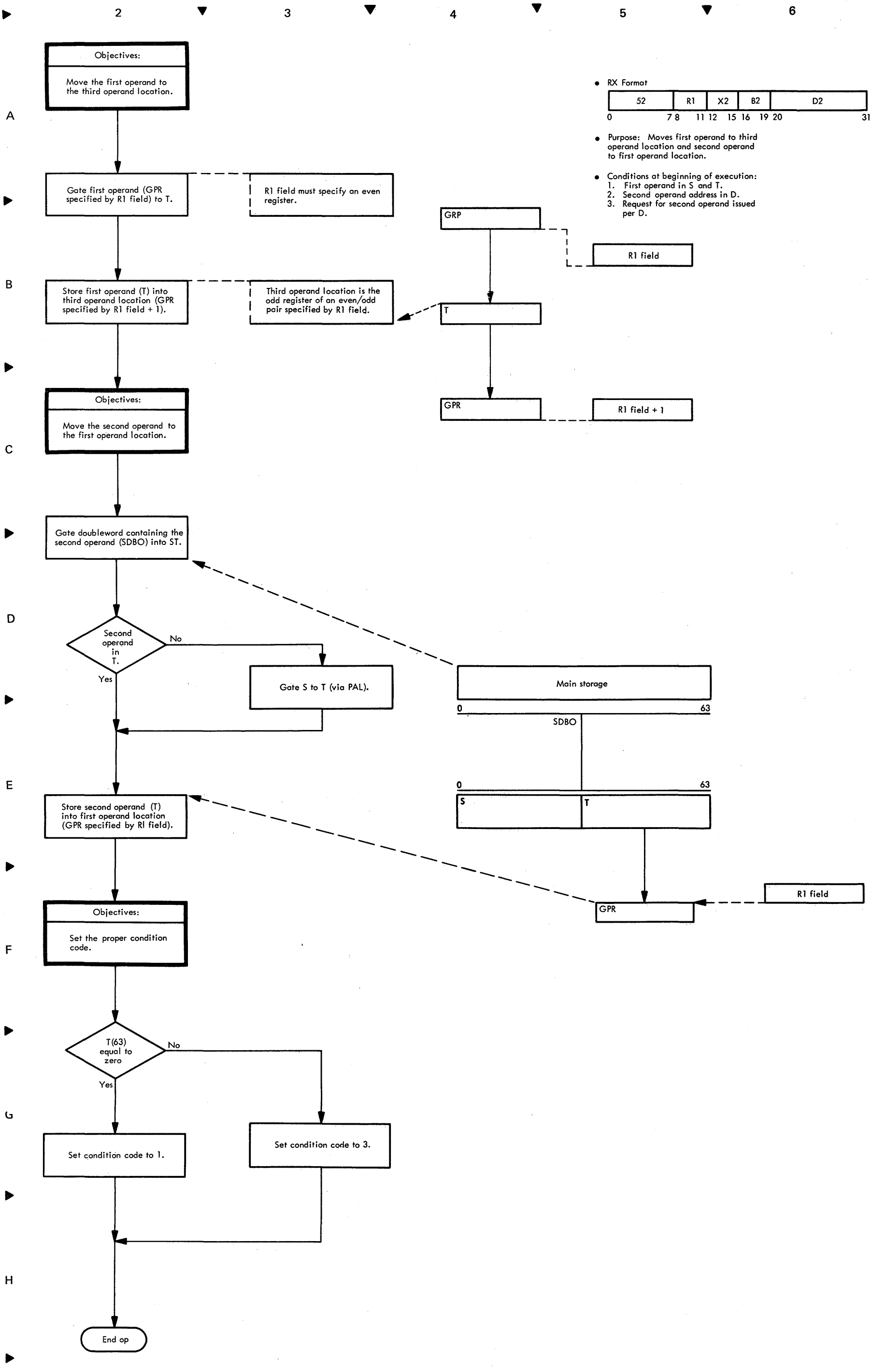


Diagram 5-906. Load Chain, LC (52)

Diagram 6-1. CE Control Panel (Sheet 1 of 2)

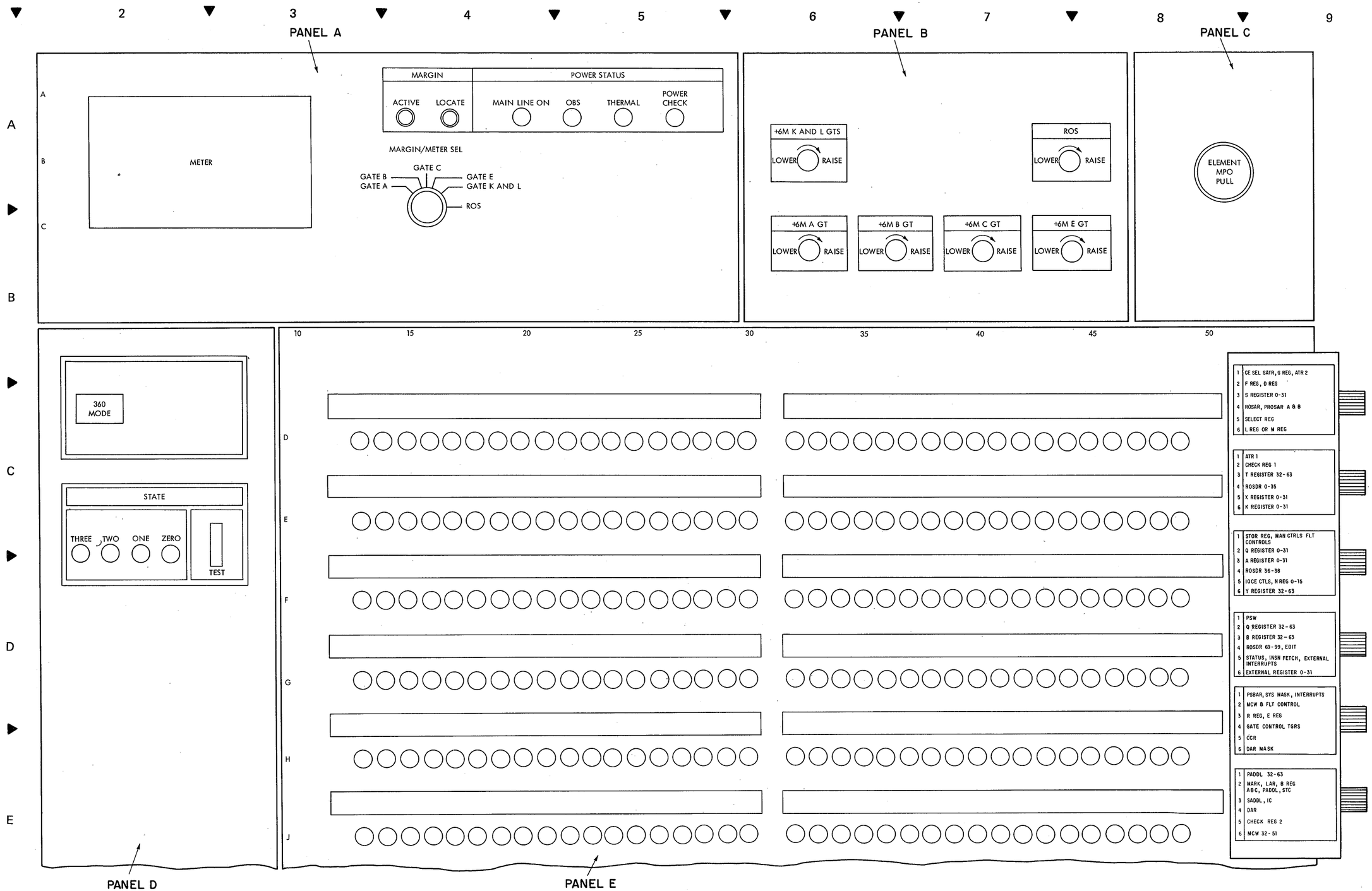
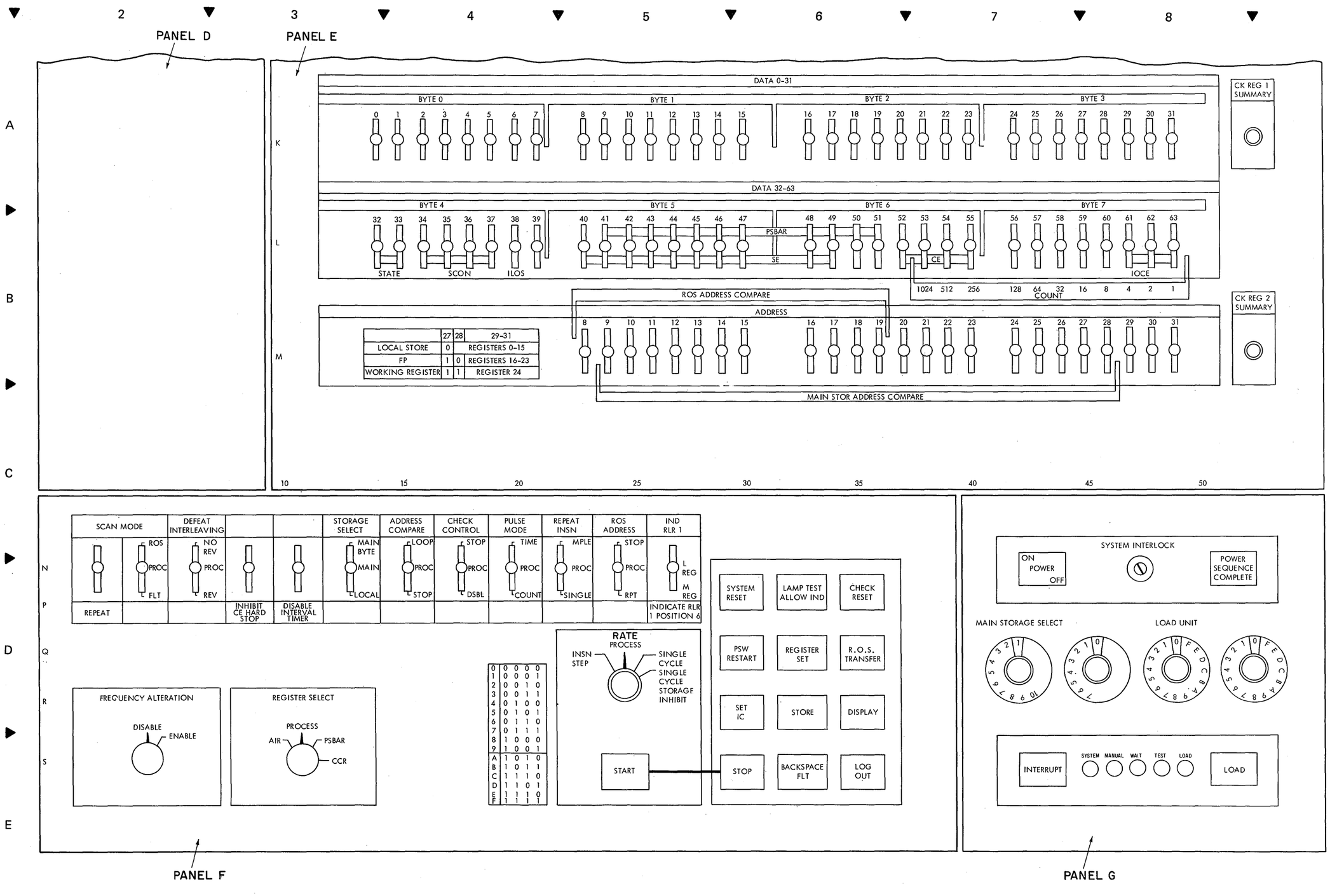


Diagram 6-1. CE Control Panel (Sheet 2 of 2)



DISPLAY NO. 1

	2	3	4	5	6	7	
1	CE SEL ATR 1 2 3 4; T C STEP			G REGISTER (WRITE DIRECT) P 0-7 0 1 2 3 4 5 6 7		ATR-2 P 32-39 32 33 34 35 36 37 38 39	1
2	F REGISTER (READ DIRECT) P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15			D REGISTER P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31			2
A 3	S REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						3
4	READ ONLY STORAGE ADDRESS REGISTER 0 1 2 3 4 5 6 7 8 9 10 11	ROS PREVIOUS ADDRESS REGISTER A 0 1 2 3 4 5 6 7 8 9 10 11			ROS PREVIOUS ADDRESS REGISTER B 0 1 2 3 4 5 6 7 8 9 10 11		4
▶ 5	PAM/RCU P 0-7 1 2 3; TCU 1 2 3; P 8-15; SE 1 2; SELECT REGISTER 3 4 5 6 7 8; SE/DE P 16-23 9 10; CE 1 2 3 4; P 24-31; IOCE 1 2 3						5
6*	L REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						6
B 6*	M REGISTER P 32-39 32 33 34 35 36 37 38 39 P 40-47 40 41 42 43 44 45 46 47 P 48-55 48 49 50 51 52 53 54 55 P 56-68 56 57 58 59 60 61 62 63						6

DISPLAY 1

1	CE SEL SATR, G REG, ATR-2
2	F REG, D REG
3	S REG (0-31)
4	ROSAR, PROSAR A AND B
5	SELECT REG
6	L REG*
6	M REG*

*Selected via INDICATE ON ROLLER 1 POSITION 6 switch.

DISPLAY NO. 2

	2	3	4	5	6	7	
1	ADDRESS TRANSLATION REGISTER 1 P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						1
▶ 2	E REG PTY; PADD FULL SUM 4-7 8-15 16-23 24-31 32-39 40-47 48-55 56-63 64-67			MLPR DECODE PTY; PADD HALF-SUM 4-7 8-15 16-23 24-31 32-39 40-47 48-55 56-67; ERROR TGR		ROS PTY 2-42 43-68 69-99; SADD HALF SUM FULL SUM	2
3	T REGISTER P 32-39 32 33 34 35 36 37 38 39 P 40-47 40 41 42 43 44 45 46 47 P 48-55 48 49 50 51 52 53 54 55 P 56-63 56 57 58 59 60 61 62 63						3
C 4	0 1; W FAA CONTROLS 2 3 4 5; A AB, IC INGATES 6 7 8 9; B LS-T 10; LS-S 11; C ST, D, Q, G, PSW INGATES 12 13 14 15 16; D FLT 17; F INGT AND EOP A FIELD 18; SD 19; P 6-42; E EMIT E, R INGATES 21 22 23 24; F MISC CONTROL PART 2 FLT B FIELD 25 26 27 28 29 30; SF; G MISC CONTROL PART 1 FLT C FIELD 31 32 33 34 35; SG						4
5	X REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						5
▶ 6	K REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						6

DISPLAY 2

1	ATR 1
2	CHECK REG 1
3	T REG (32-63)
4	ROSDR (0-35)
5	X REG (0-31)
6	K REG (0-31)

DISPLAY NO. 3

	2	3	4	5	6	7	
1	STORAGE REQUEST IC D FLT SET KEY INSERT KEY 3 CYCLE; PAGE 1 LTH; MANUAL CONTROLS RPT INSTR ADJ INIT; PULSE MODE ADJ INIT STOP BLOCK PASS PULSE BLOCK INTRPT FORCE ADDR INSTR STEP SINGLE CYCLE CHK SUMM INHIB CLOCK; FLT CONTROLS RPT TEST FLT TEST ROS TEST SCC SYNC; DEFEAT INTLV NO REV REV; BACK SPACE CNSL LOG SOROS DIAG DIAG RELEASE TIC GAP						1
D 2	Q REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						2
3	A REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15 P 16-23 16 17 18 19 20 21 22 23 P 24-31 24 25 26 27 28 29 30 31						3
▶ 4	HW EXTD 36 37; H LS LAR CONTROL 38 39 40 41 42; L STOR REQ SET MARK 43 44 45 46; NA, AD 47; NEXT ROS BASE ADDRESS 48 49 50 51 52 53 54 55 56; K Y BRANCH 57 58 59 60 61; J X OR Z BRANCH 62 63 64 65 66 67 68; PREV ADR A; FLT MODE						4
5	PERMIT IOCE 1 2 3; IOCE MACH CHK 1 2 3; SEL IOCE 1 2 3; IOCE INTRPT REQ MC REQ; INTRPT GATE TGR; N REGISTER P 0-7 0 1 2 3 4 5 6 7 P 8-15 8 9 10 11 12 13 14 15						5
6	Y REGISTER P 32-39 32 33 34 35 36 37 38 39 P 40-47 40 41 42 43 44 45 46 47 P 48-55 48 49 50 51 52 53 54 55 P 56-63 56 57 58 59 60 61 62 63						6

DISPLAY 3

1	STOR REQ, MAN CTLS, FLT CTLS
2	Q REG (0-31)
3	A REG (0-31)
4	ROSDR (36-68)
5	IOCE CTLS, N REG (0-15)
6	Y REG (32-63)

E

Diagram 6-2. CE Roller Switch Indicators (Sheet 1 of 2)

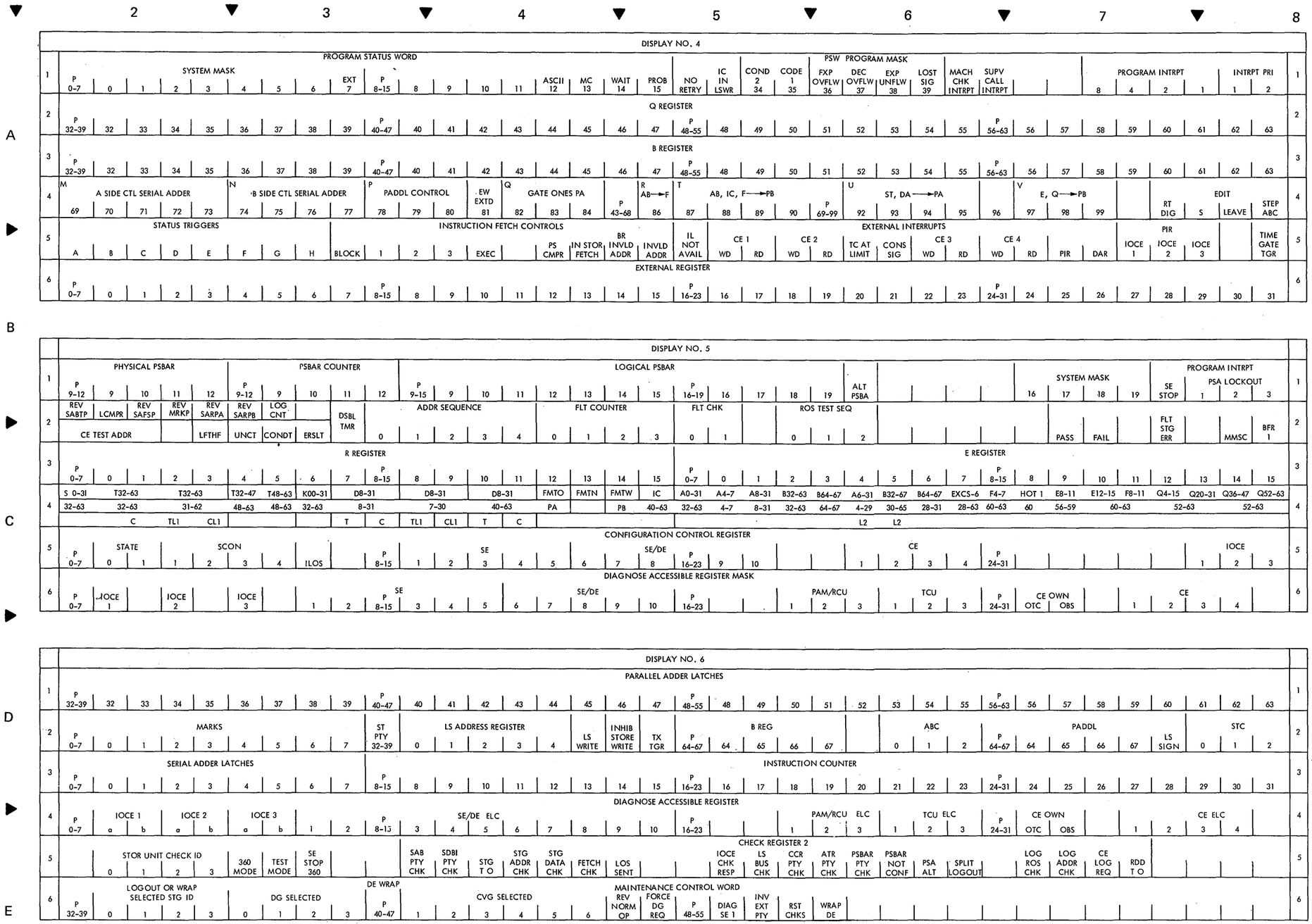


Diagram 6-2. CE Roller Switch Indicators (Sheet 2 of 2)

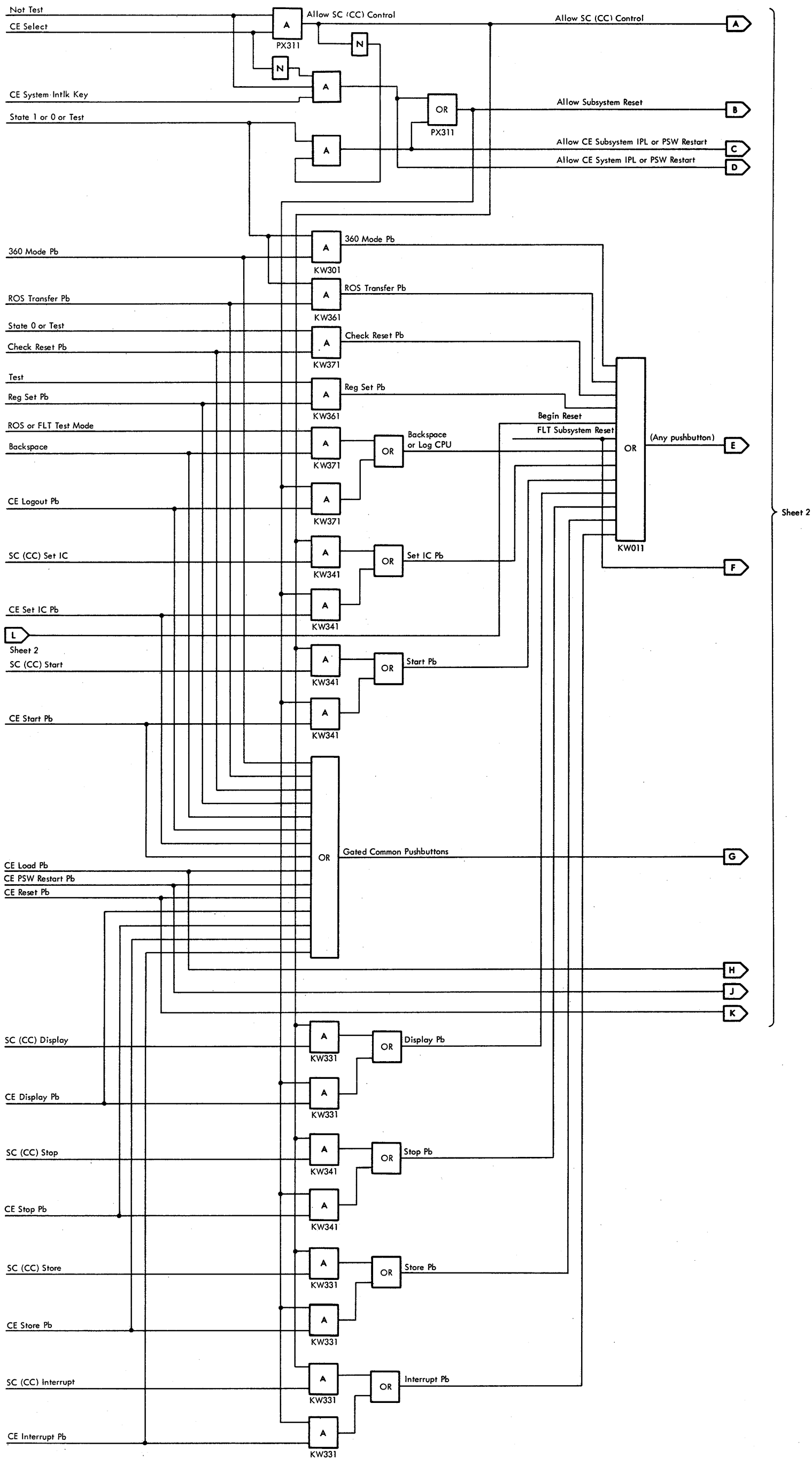


Diagram 6-3. Pushbutton Signal Generation (Sheet 1 of 2)

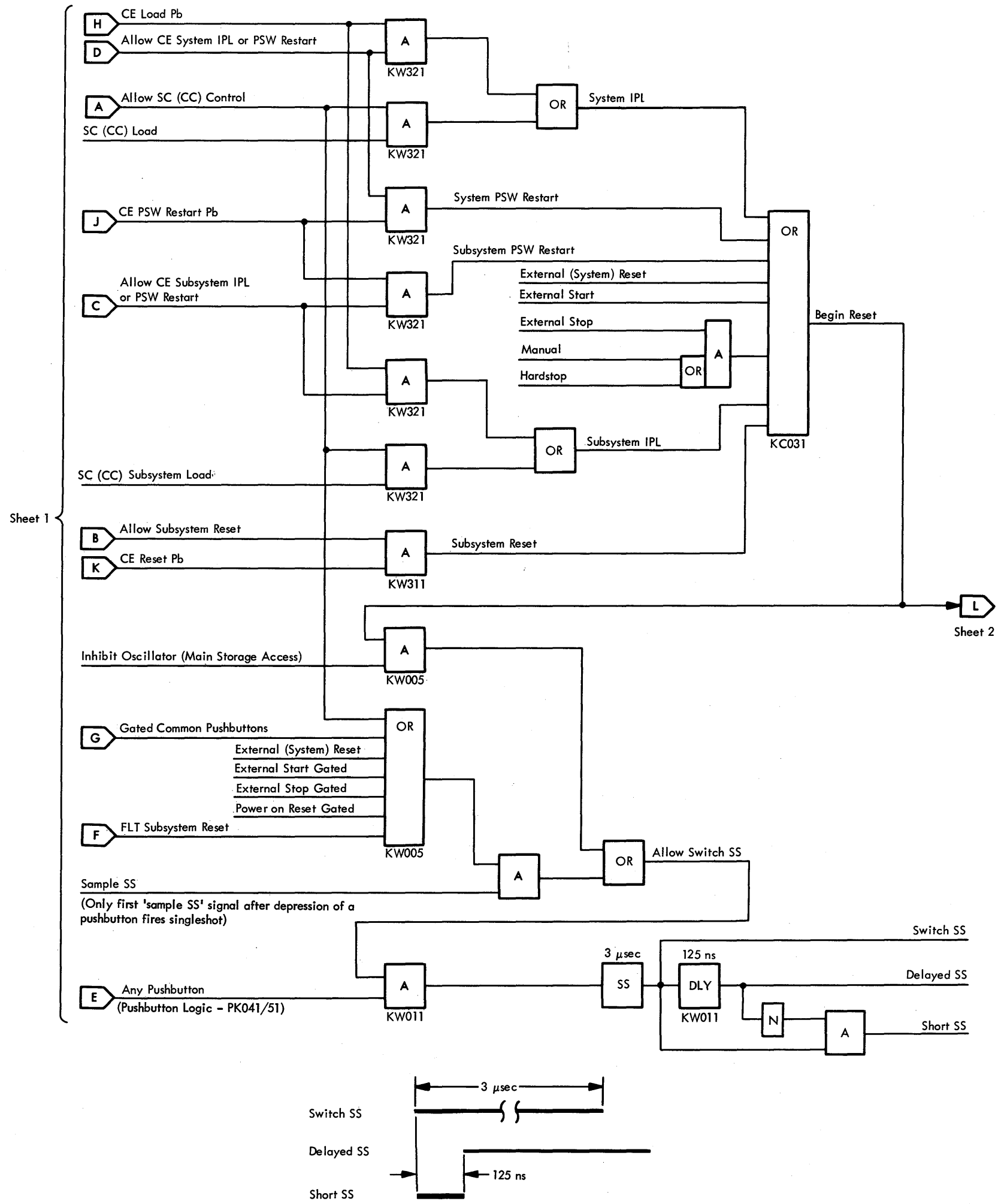


Diagram 6-3. Pushbutton Signal Generation (Sheet 2 of 2)

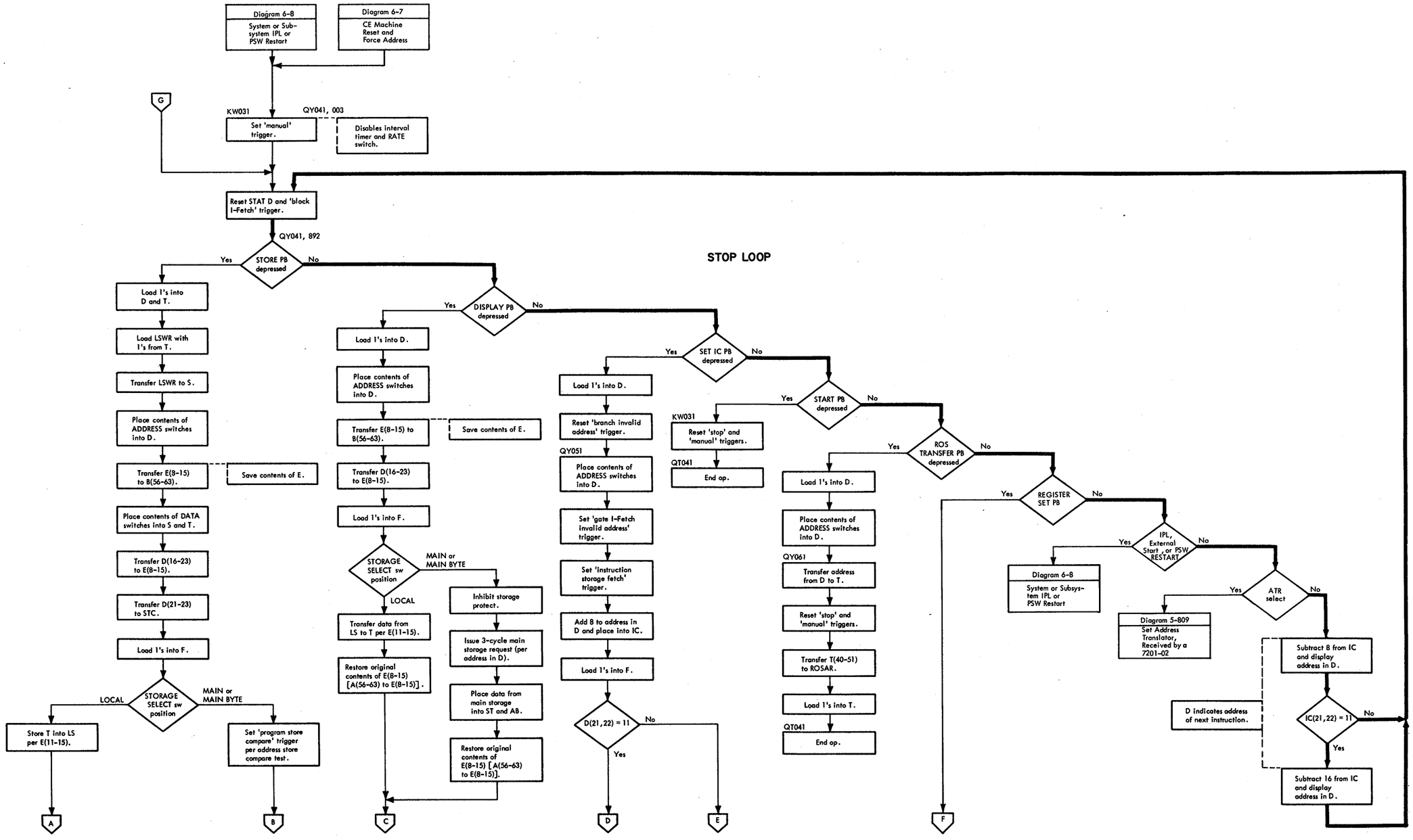


Diagram 6-4. Stop Loop Routine (Sheet 1 of 2)

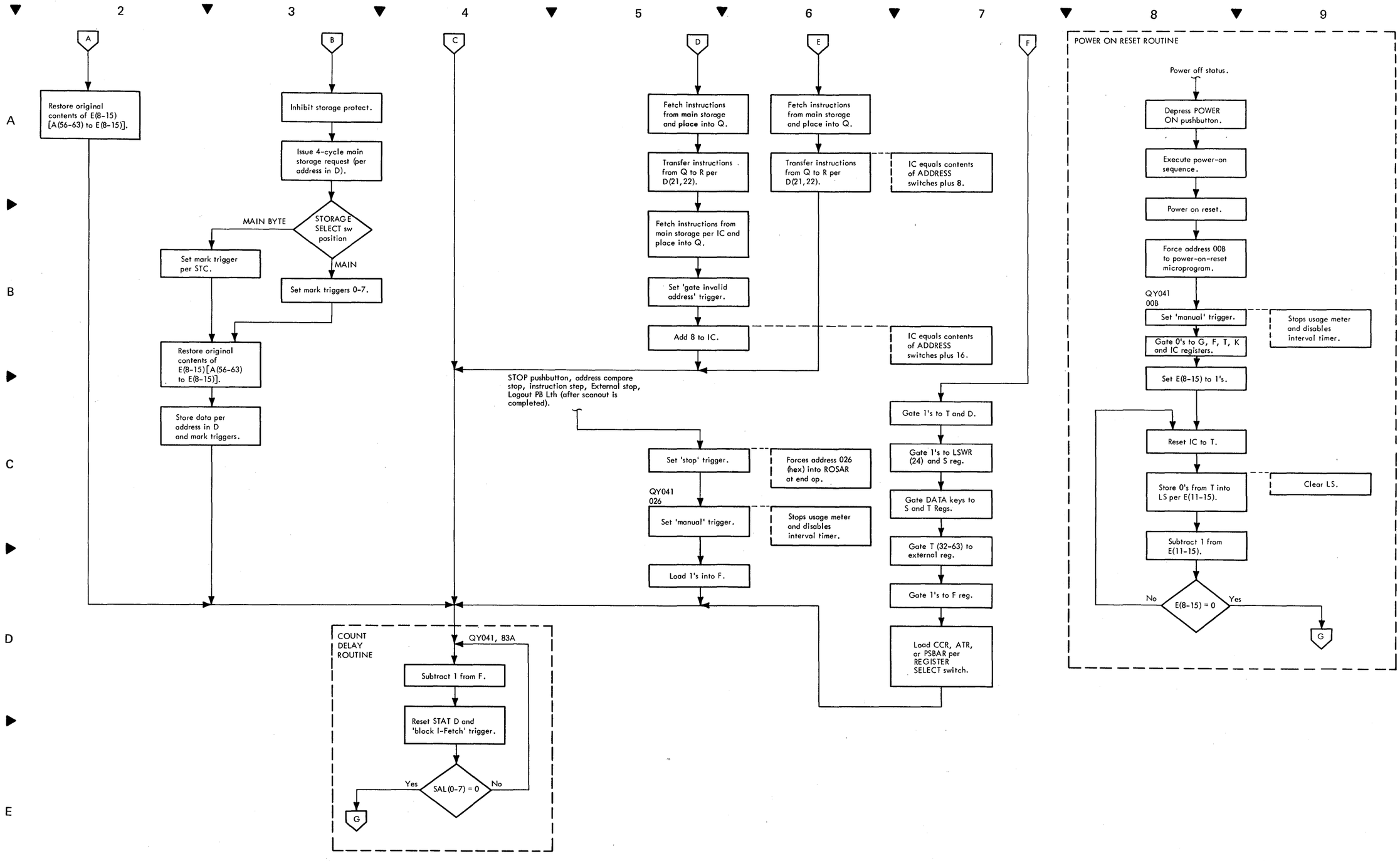


Diagram 6-4. Stop Loop Routine (Sheet 2 of 2)

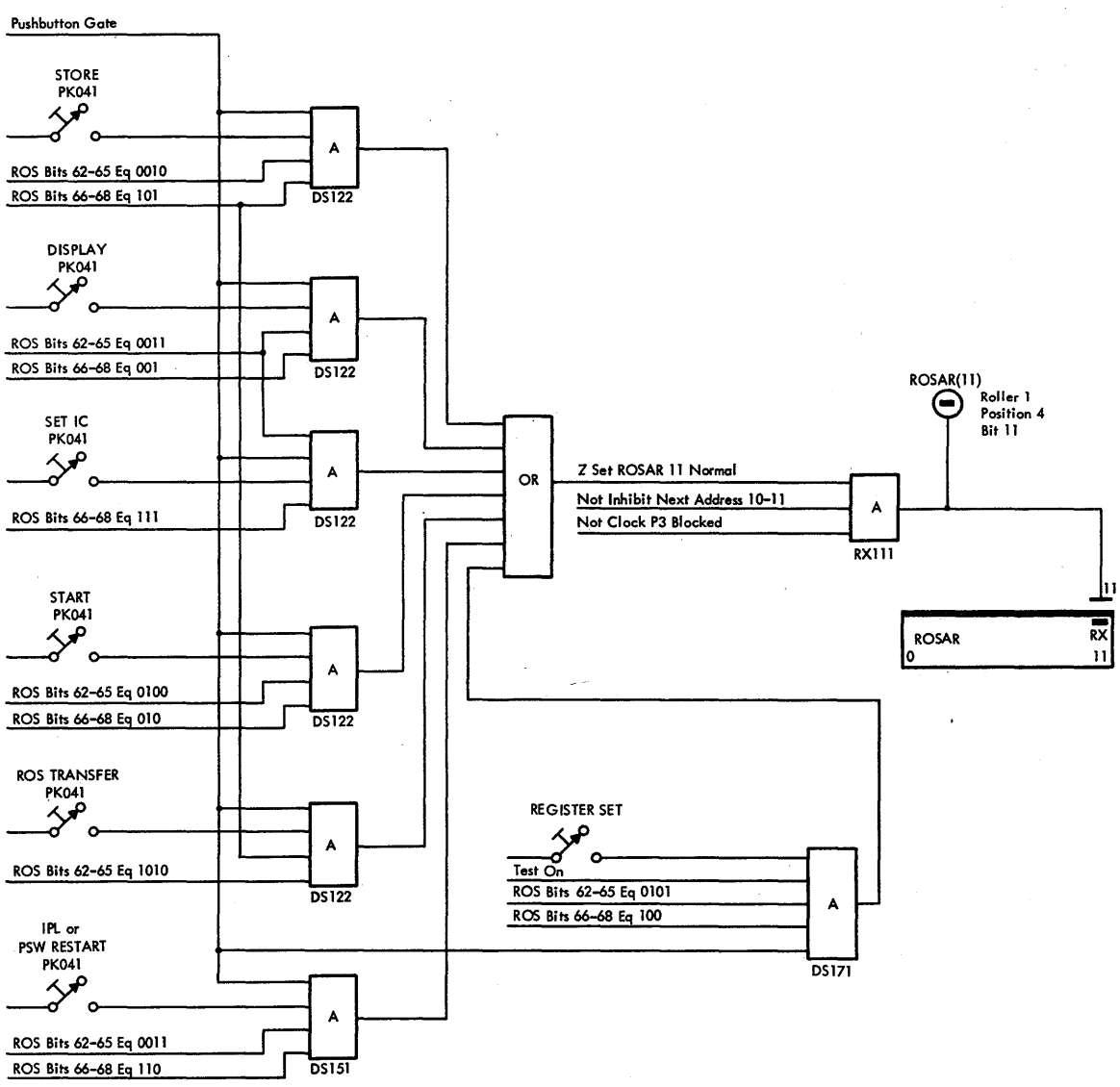


Diagram 6-5. Stop Loop Monitored Pushbutton Gating

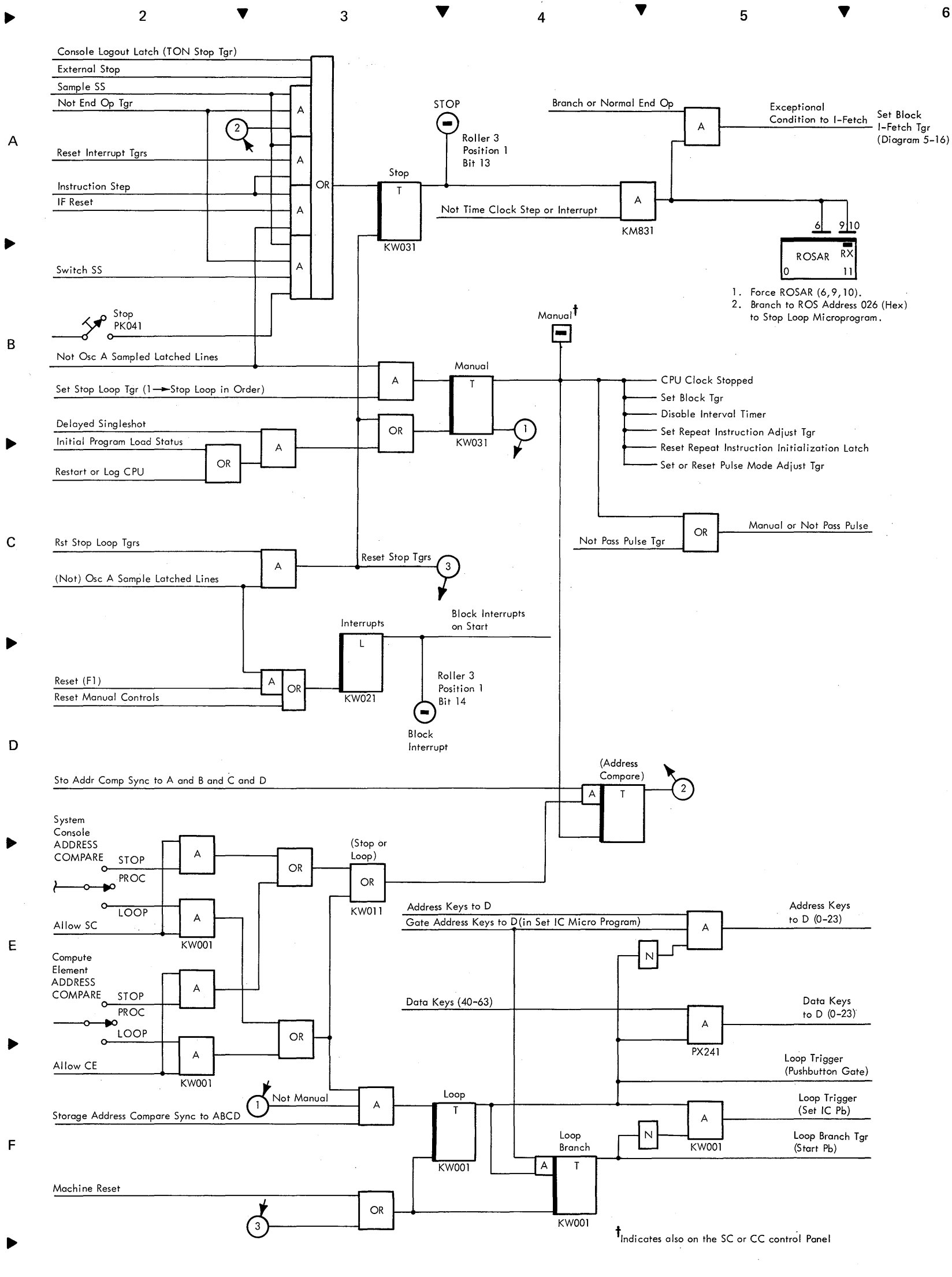


Diagram 6-6. Stop, Manual, Address Compare Triggers, and Block Interrupt Latch (Sheet 1 of 2)

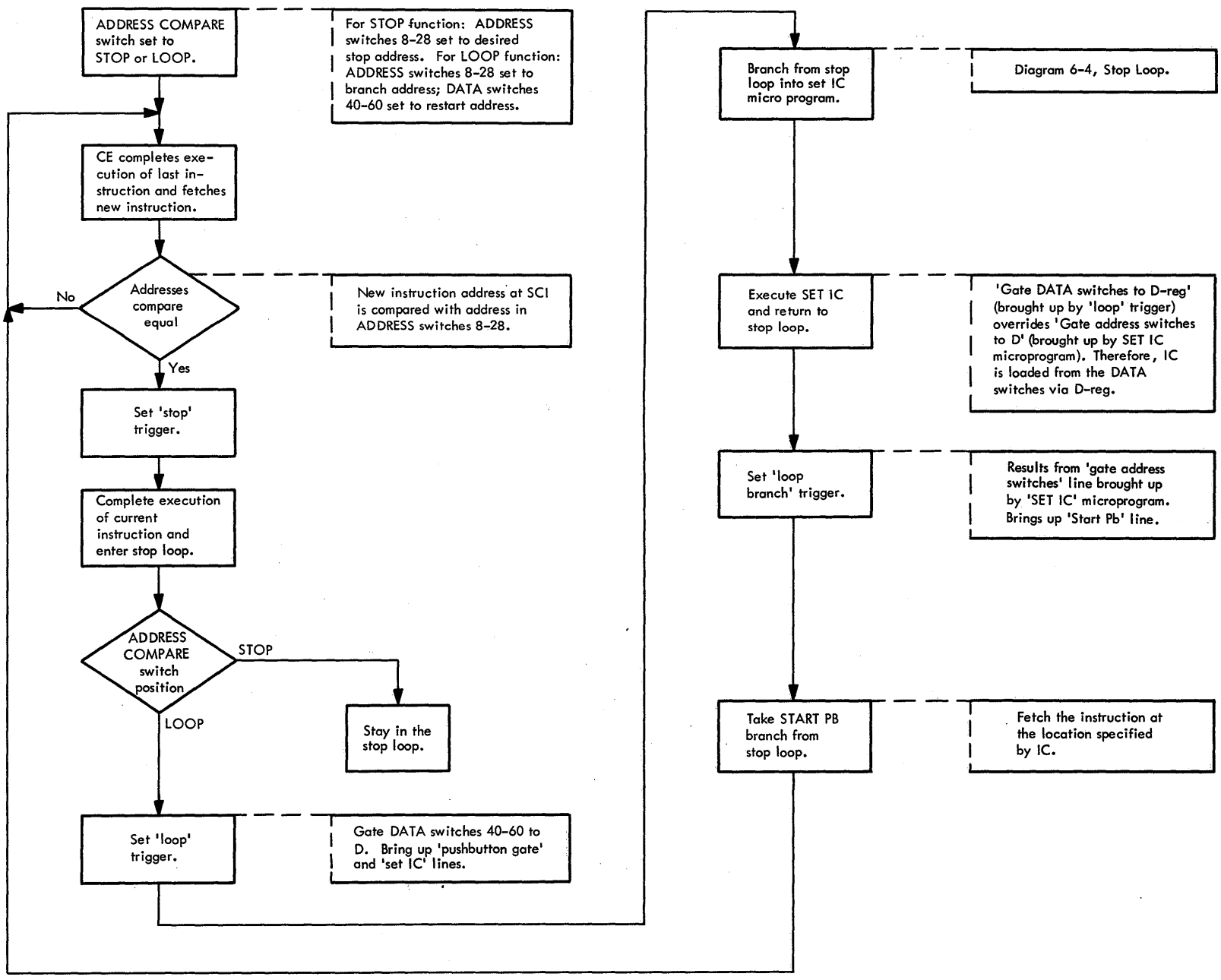


Diagram 6-6. Stop, Manual, Address Compare Triggers, and Block Interrupt Latch (Sheet 2 of 2)

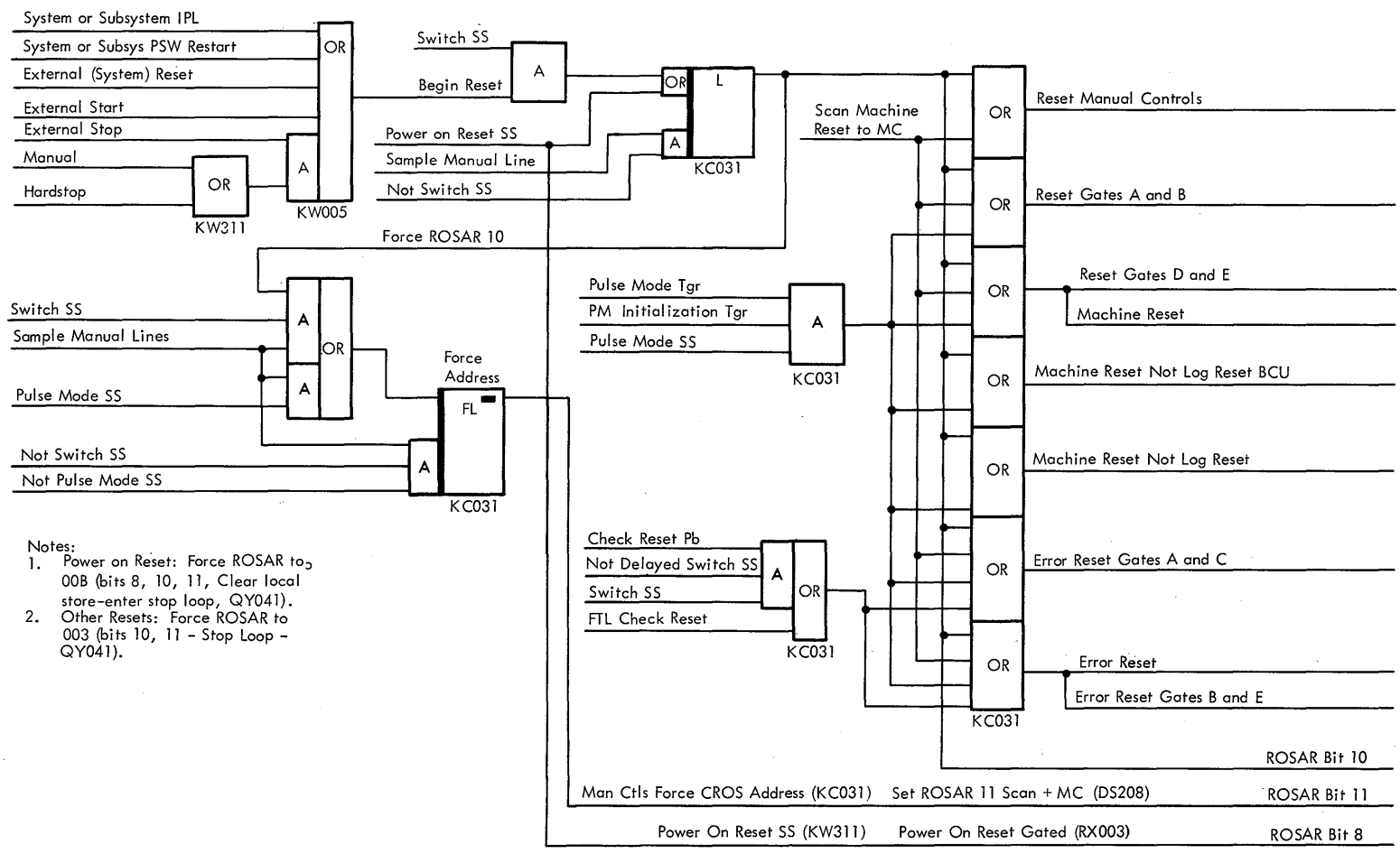
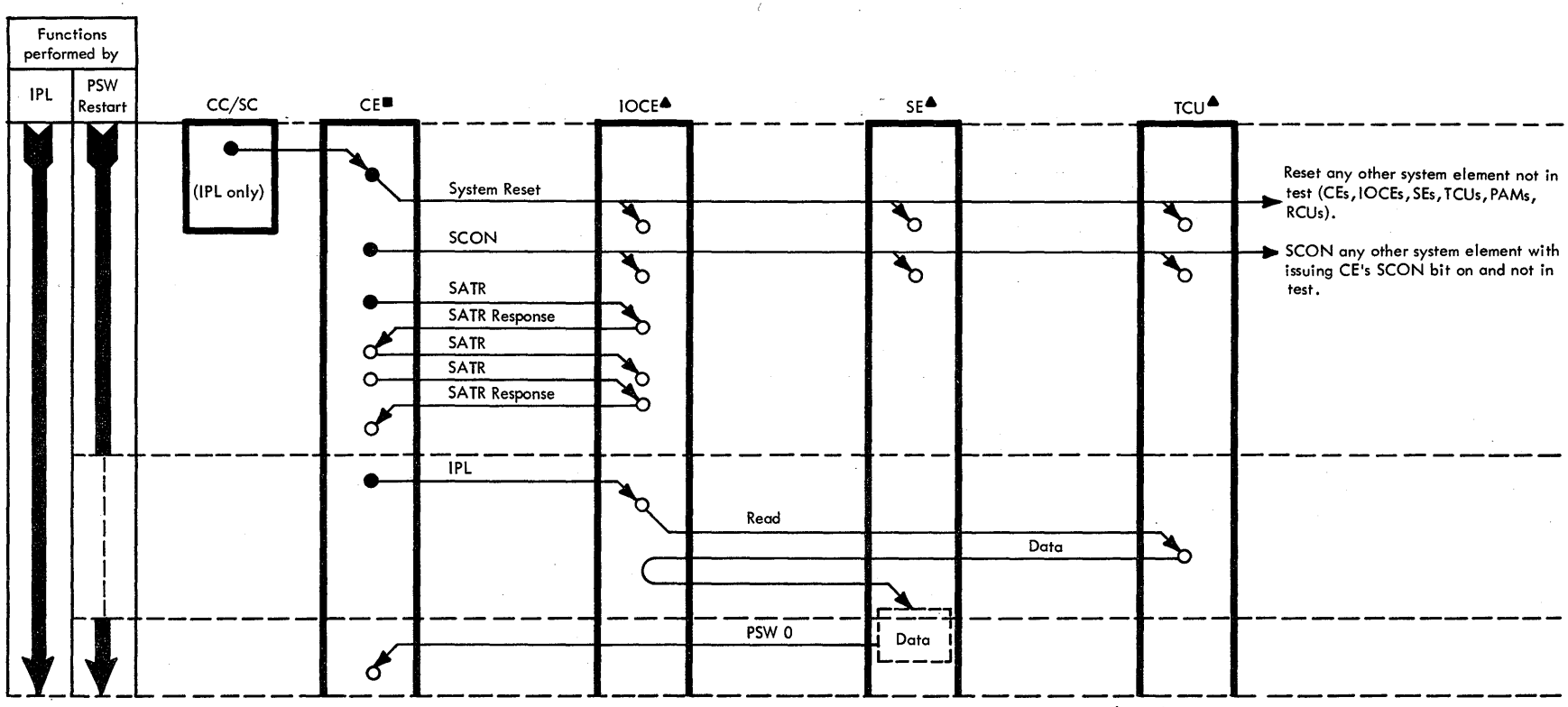


Diagram 6-7. CE Machine Reset and Force Address



Summary of System Operations: IPL or PSW Restart

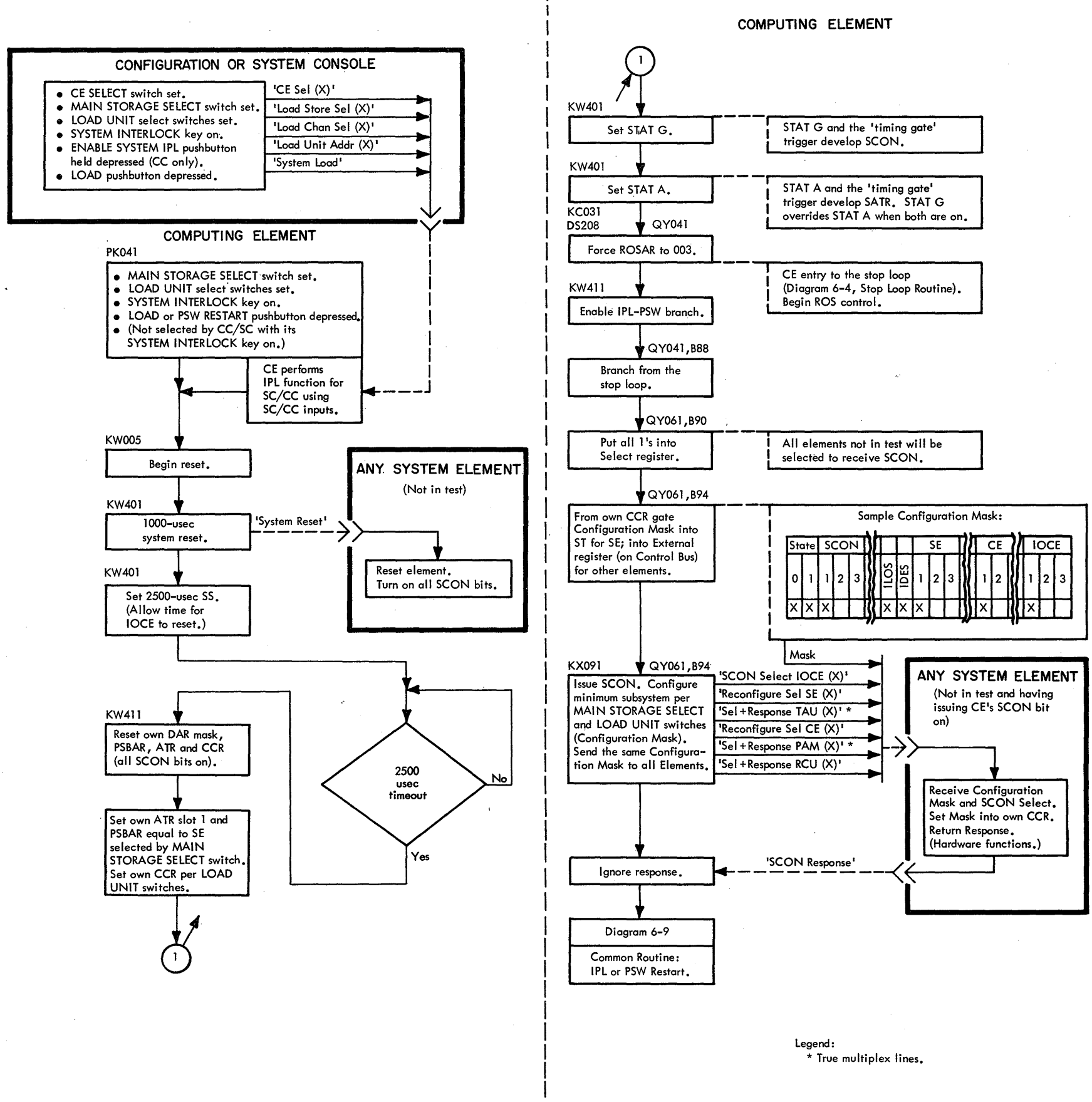
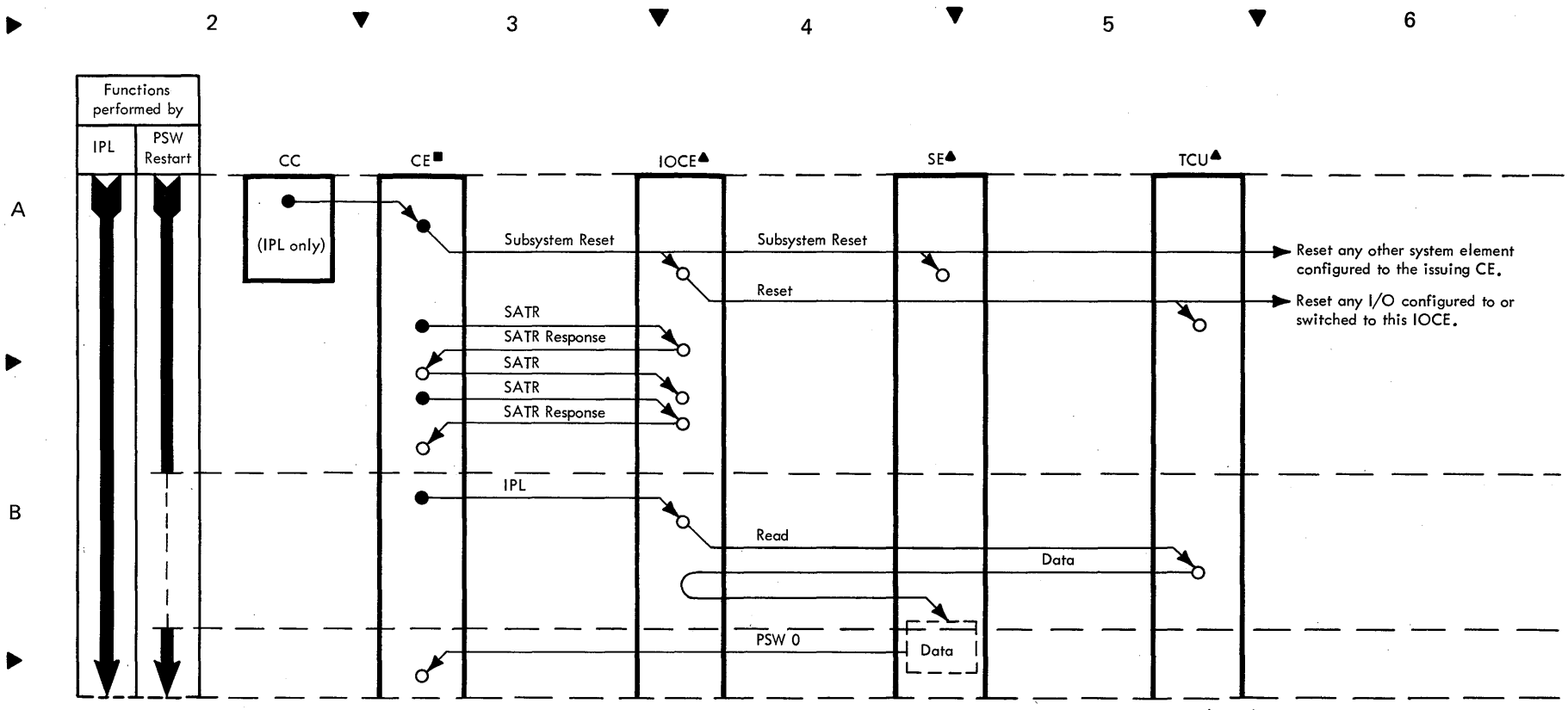


Diagram 6-8A. System Operation: IPL or PSW Restart



Summary of Subsystem Operations: IPL or PSW Restart.

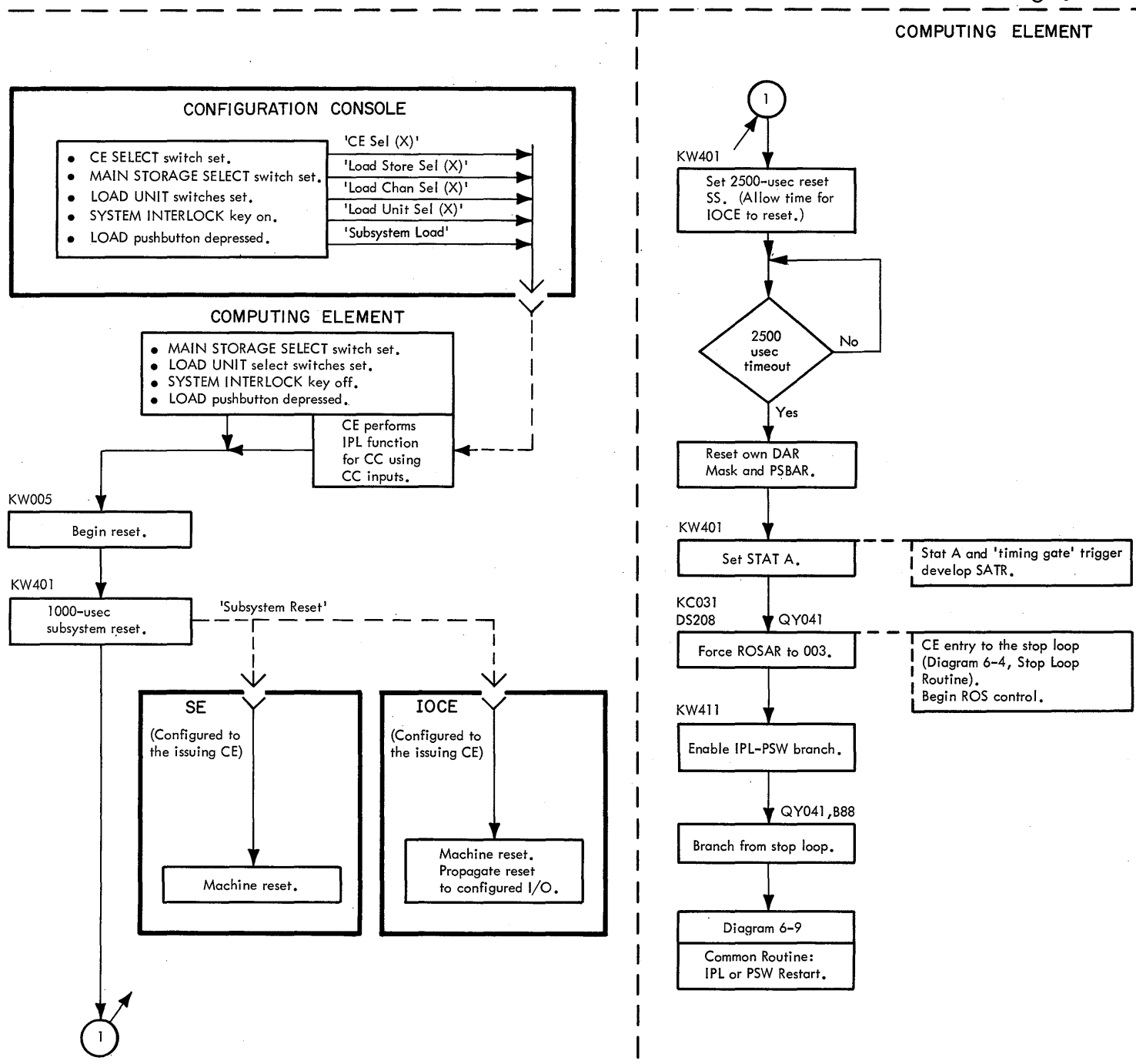


Diagram 6-8B. Subsystem Operation: IPL or PSW Restart

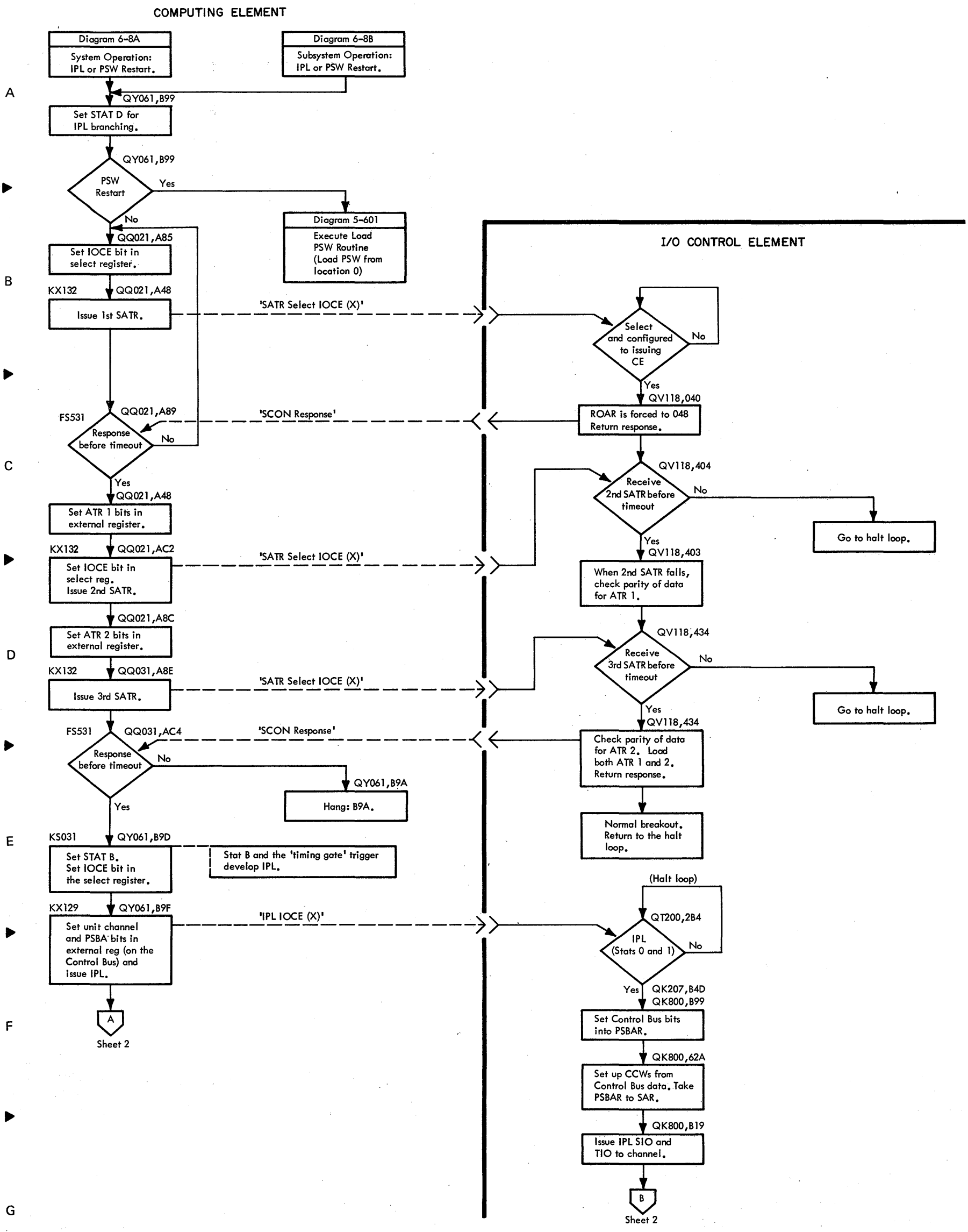


Diagram 6-9. Common Routine: IPL or PSW Restart (Sheet 1 of 2)

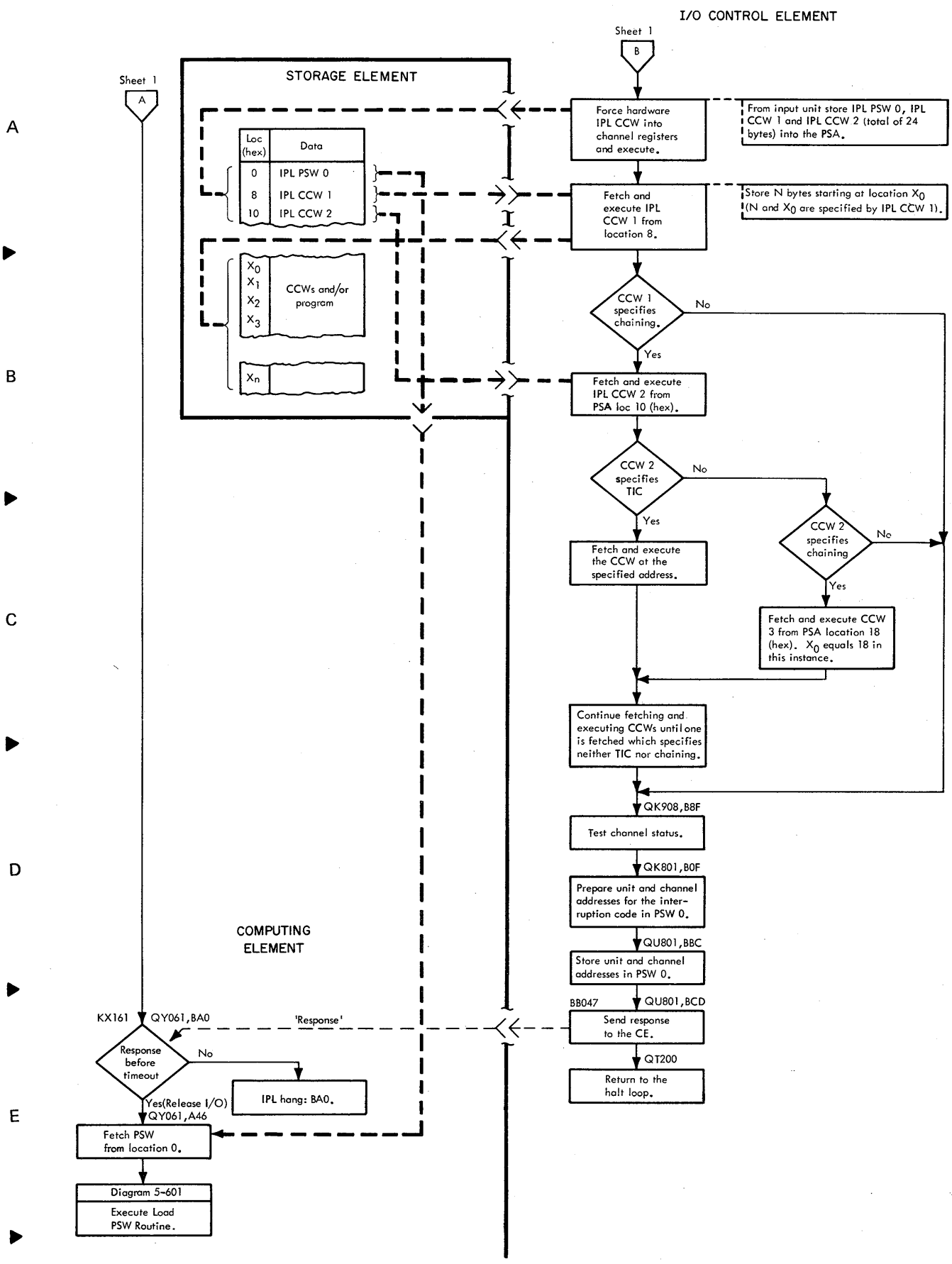


Diagram 6-9. Common Routine: IPL or PSW Restart (Sheet 2 of 2)

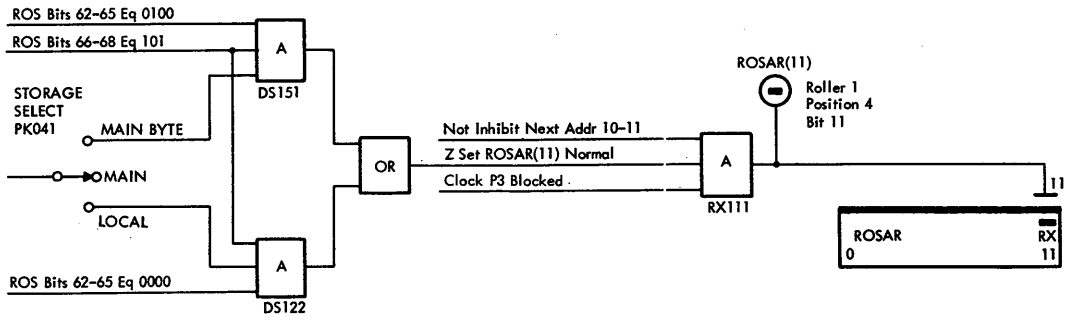


Diagram 6-10. STORAGE SELECT Switch Gating

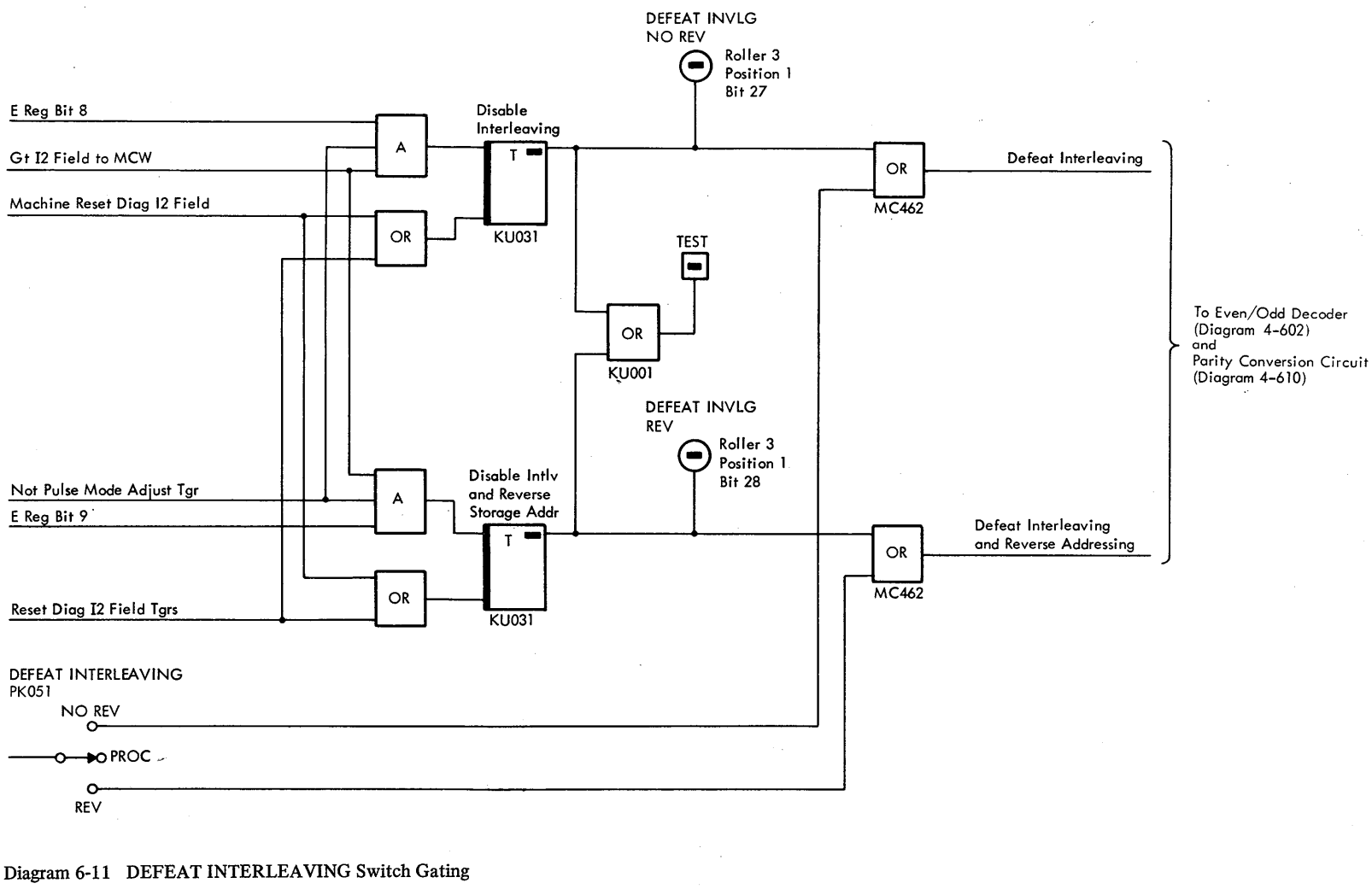


Diagram 6-11 DEFEAT INTERLEAVING Switch Gating

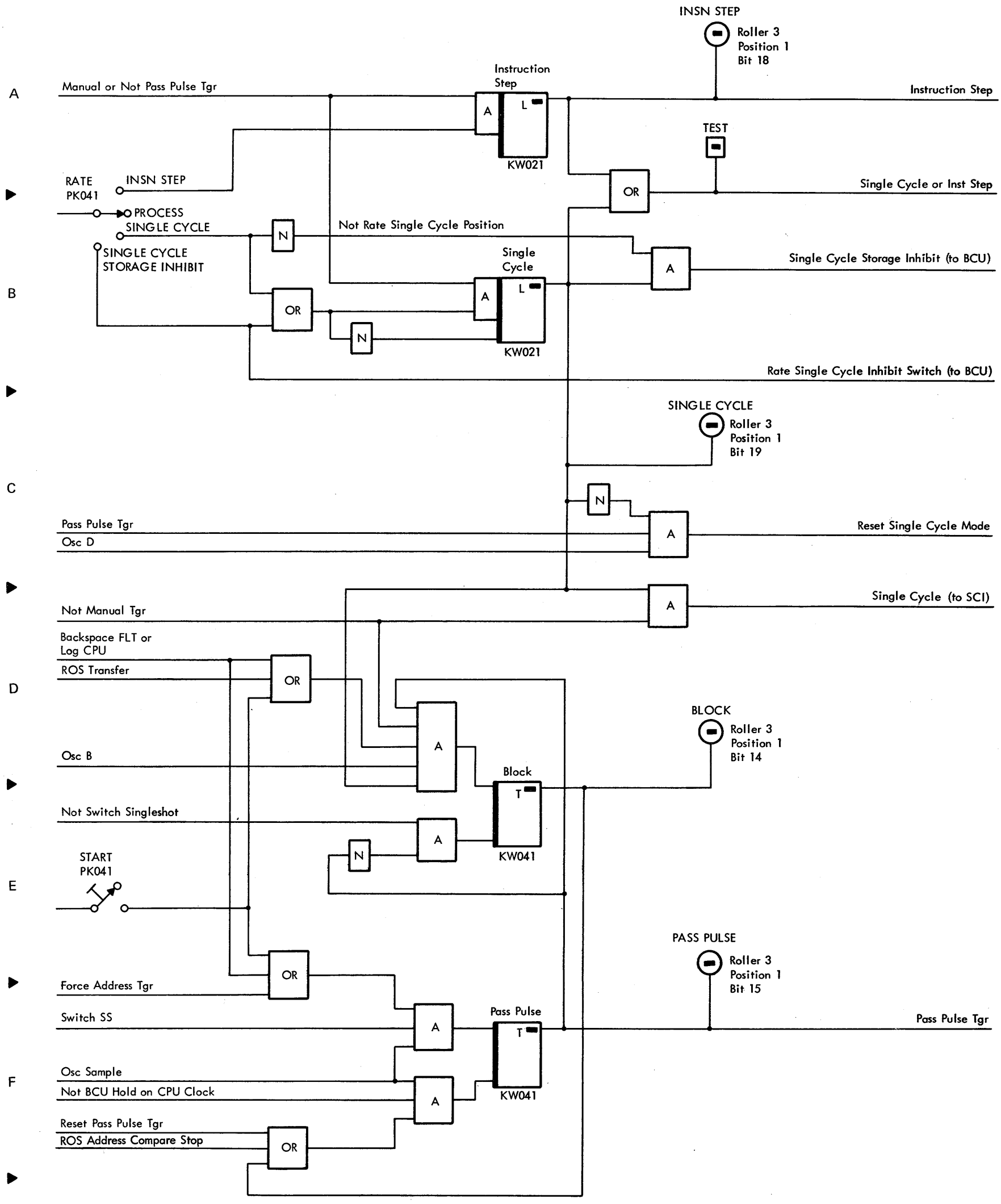


Diagram 6-12. RATE Switch Logic

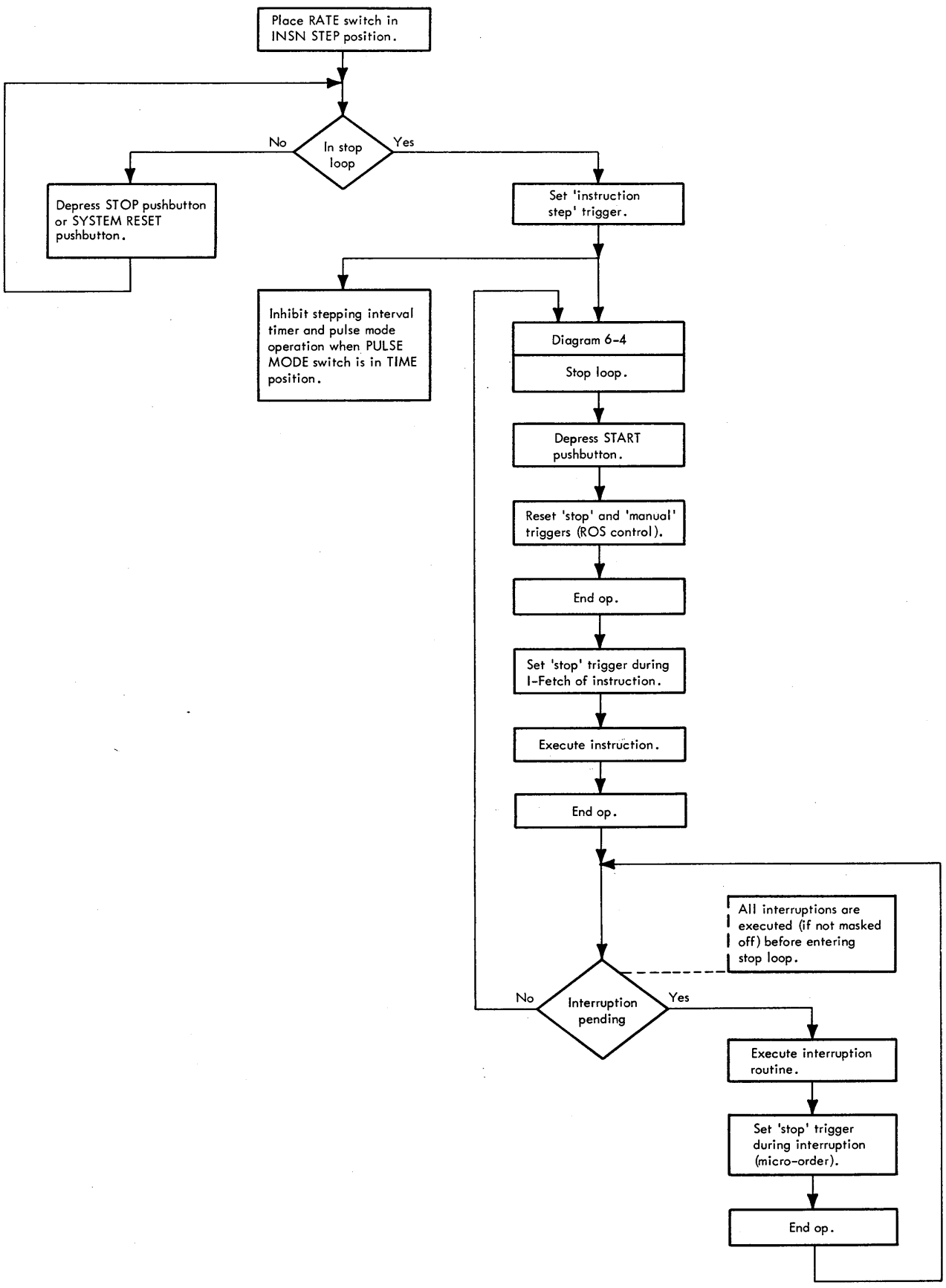


Diagram 6-13. Instruction Step Routine

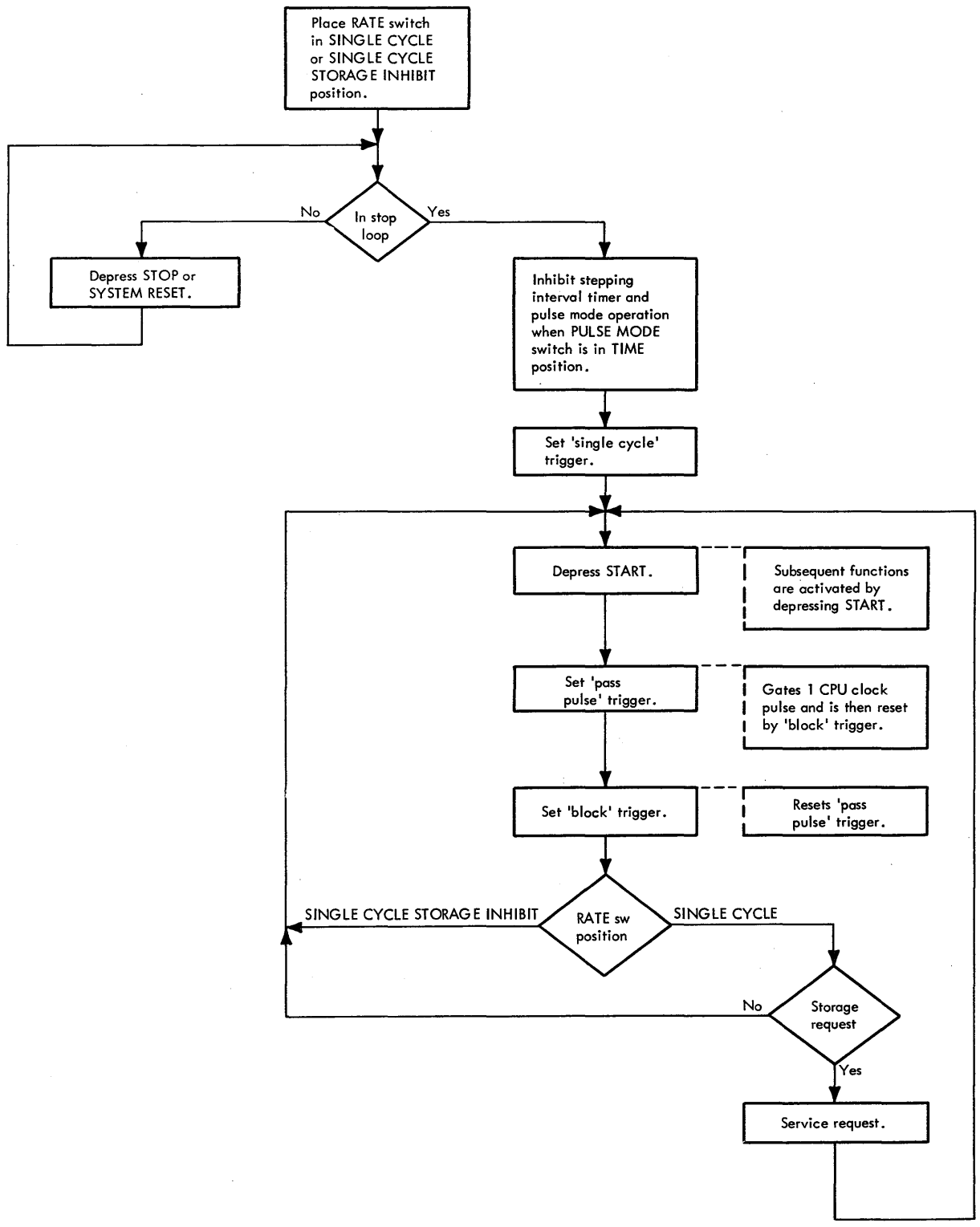


Diagram 6-14. Single-Cycle and Single-Cycle-Inhibit Routine

RPT INSN

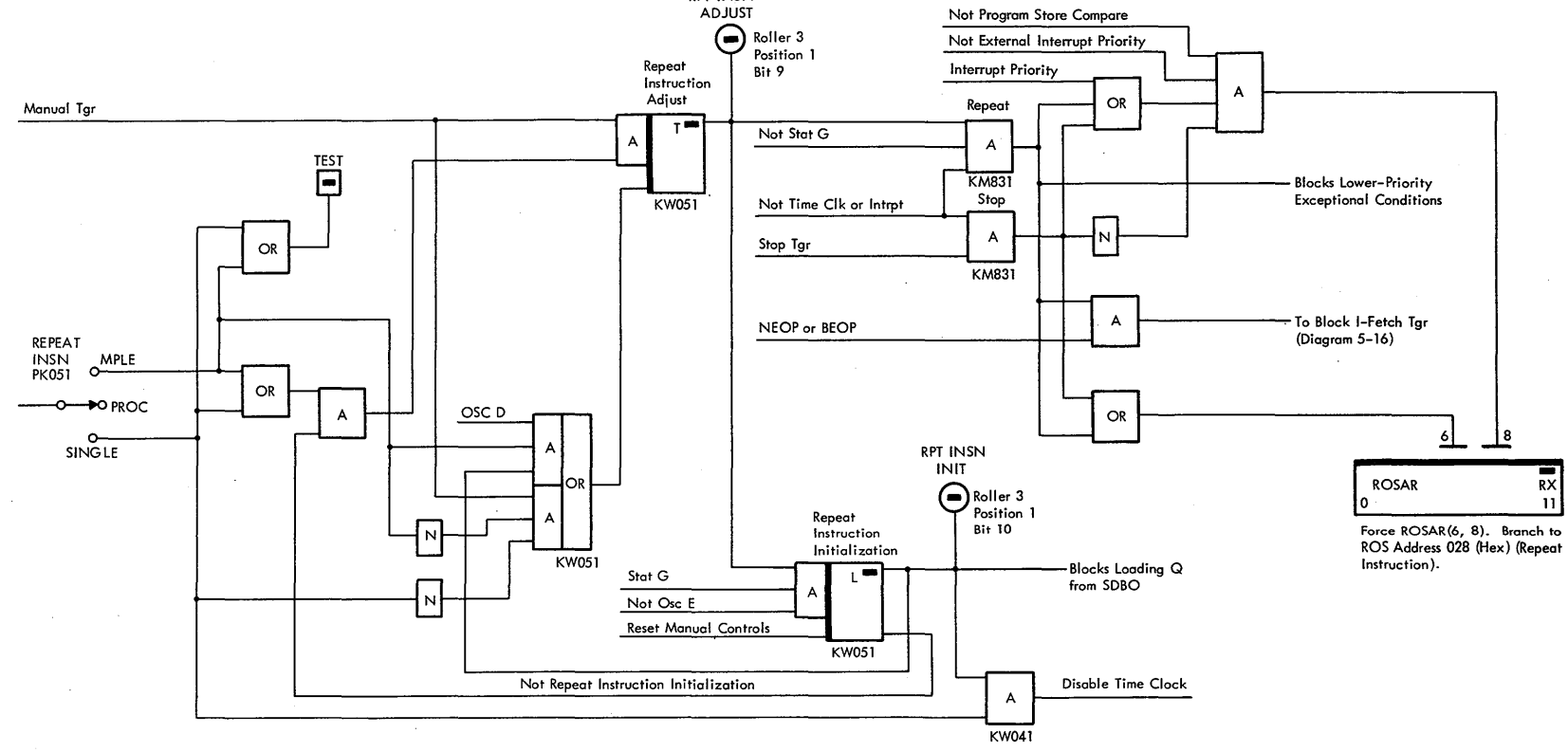


Diagram 6-15. Repeat Instruction Switch Logic

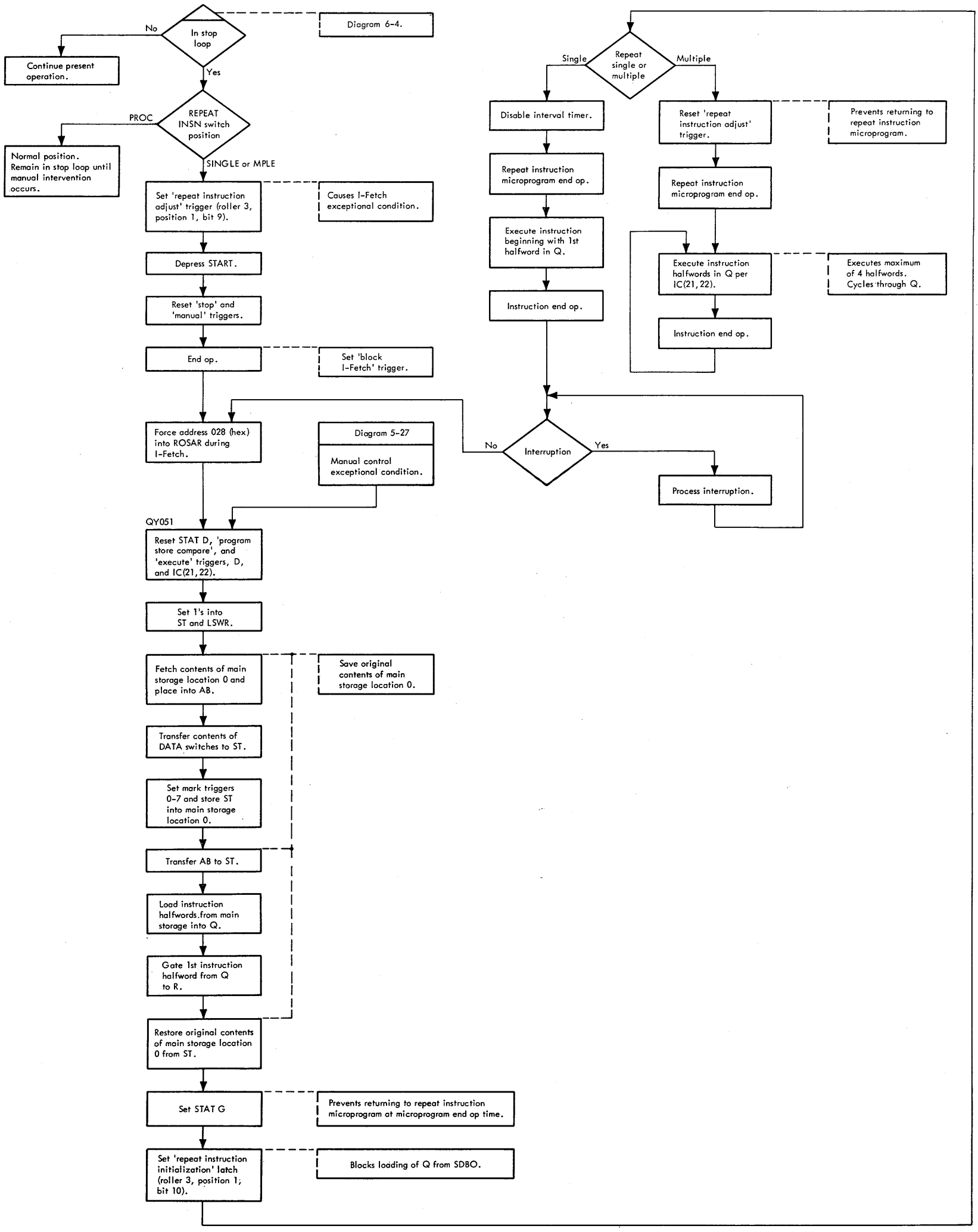


Diagram 6-16. Repeat Instruction Switch Routine

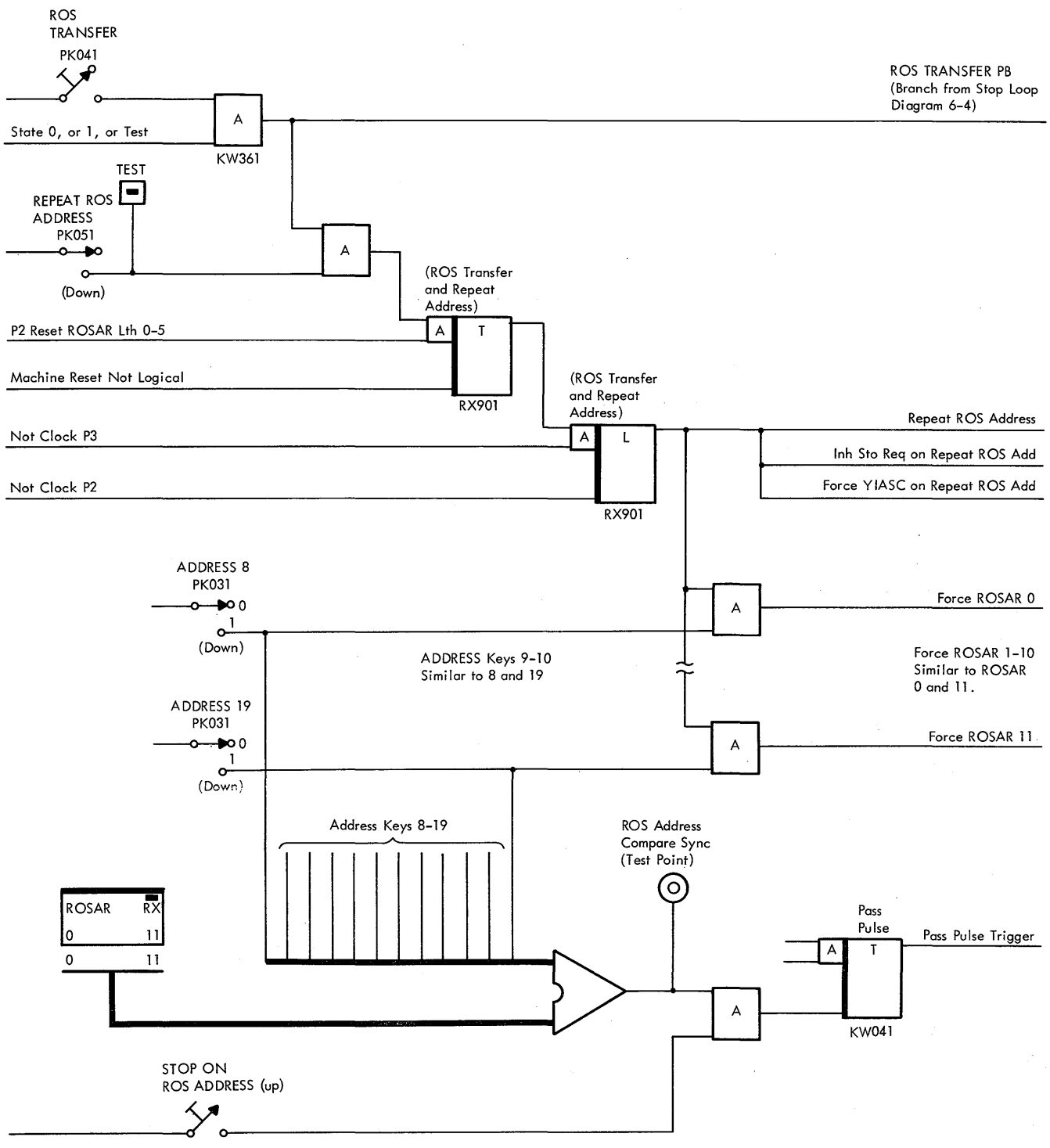
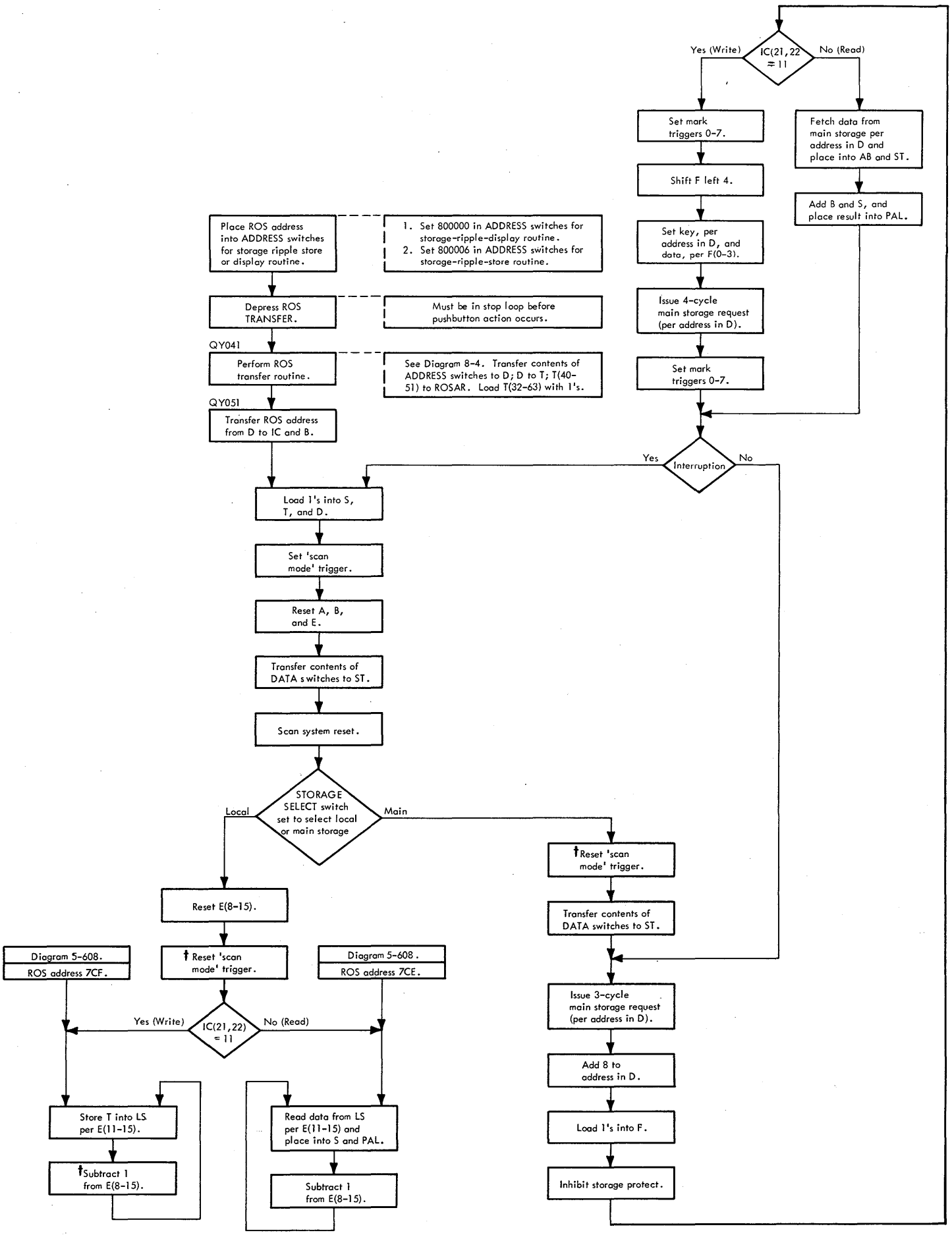


Diagram 6-17. ROS TRANSFER and REPEAT ROS ADDRESS Switch Gating



Note: Manual intervention (e.g., system reset or load) is required to get out of storage ripple loop.

† Following this block, an FLT-2, -3, or -4 microprogram is executed. These microprograms are inserted at this point to exercise circuits; they do not affect the storage ripple test.

Diagram 6-18. Storage Ripple Loop (Store and Display) Routine

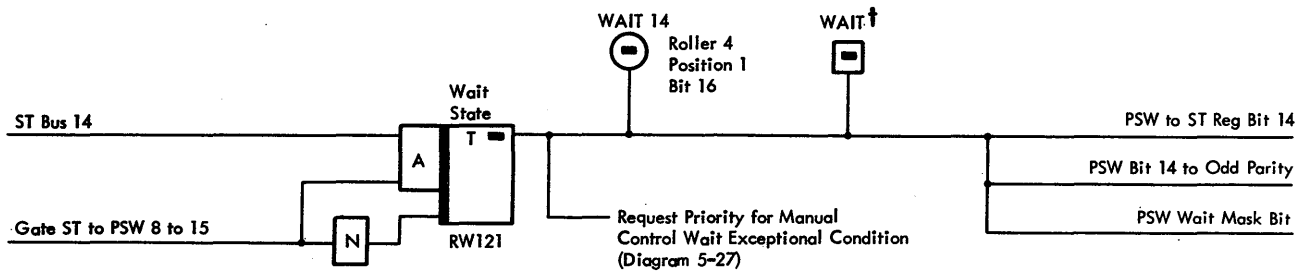


Diagram 6-19. Wait State Gating

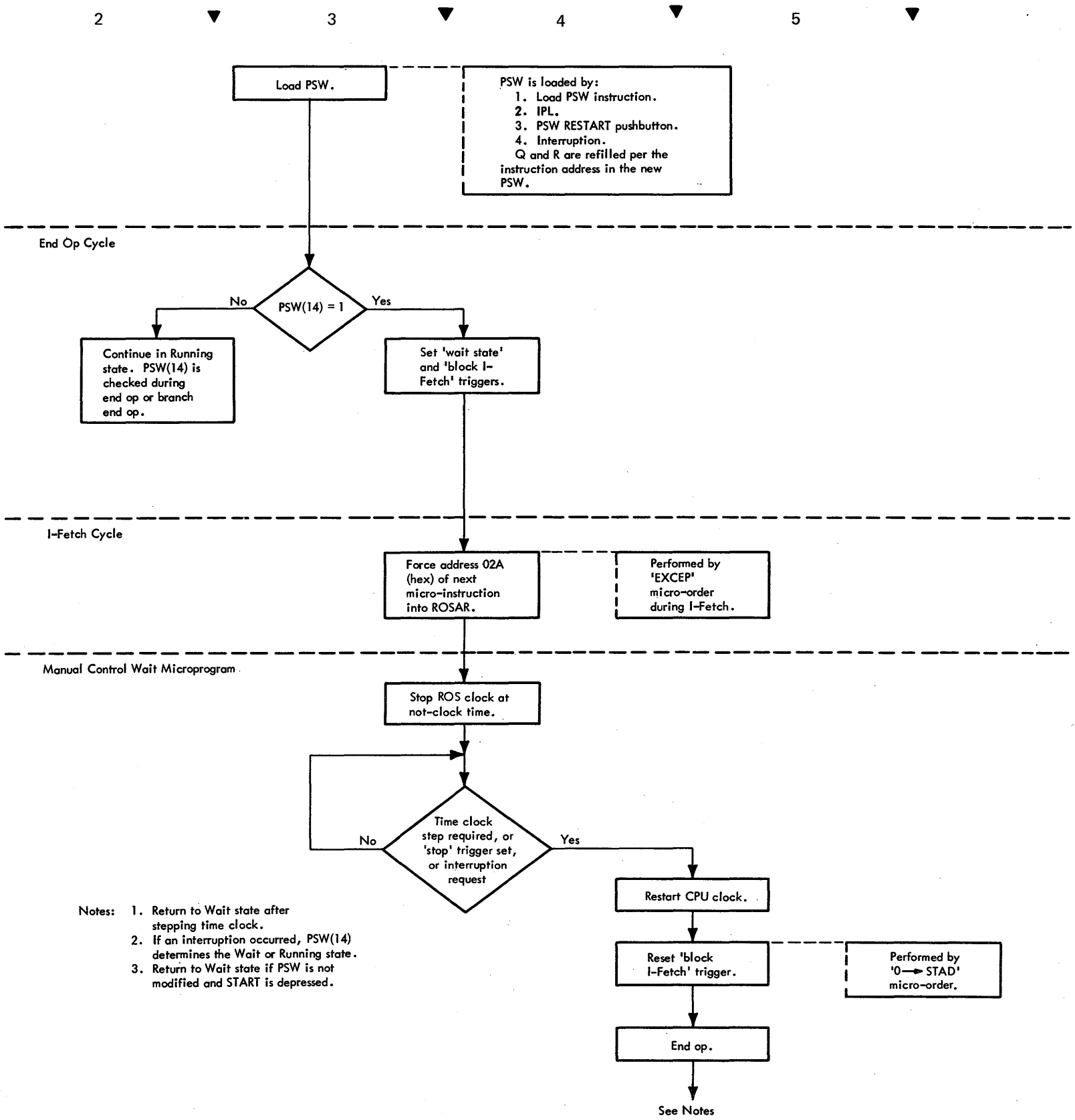


Diagram 6-20. Wait State Microprogram Routine

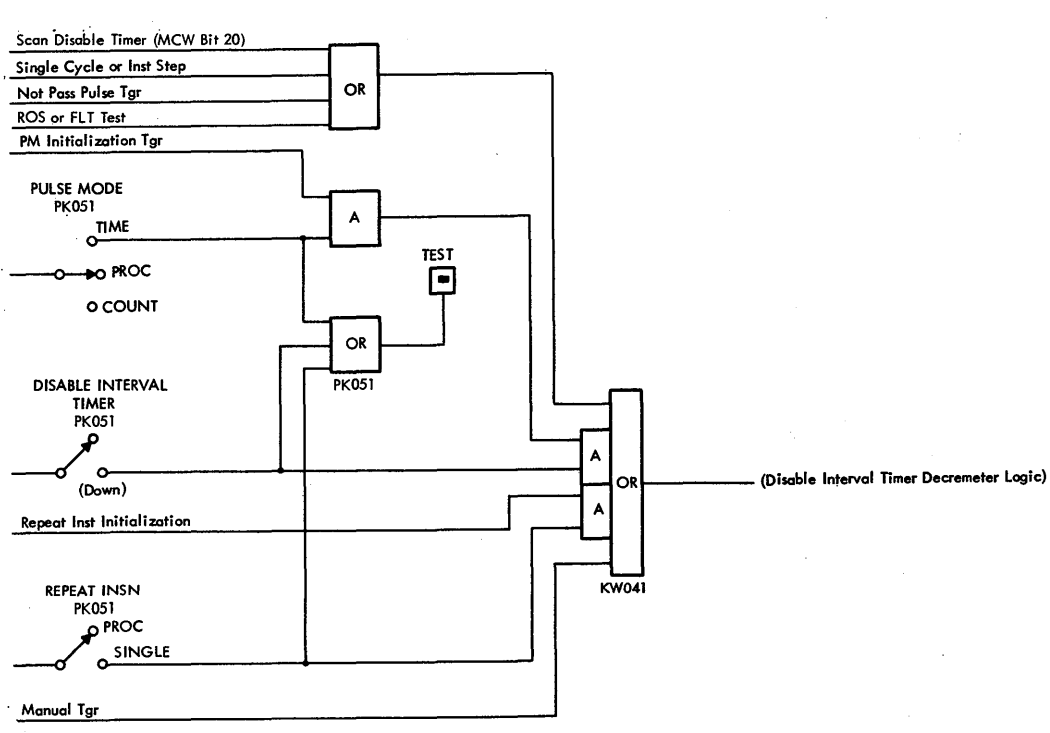


Diagram 6-21. Disable Interval Timer Logic

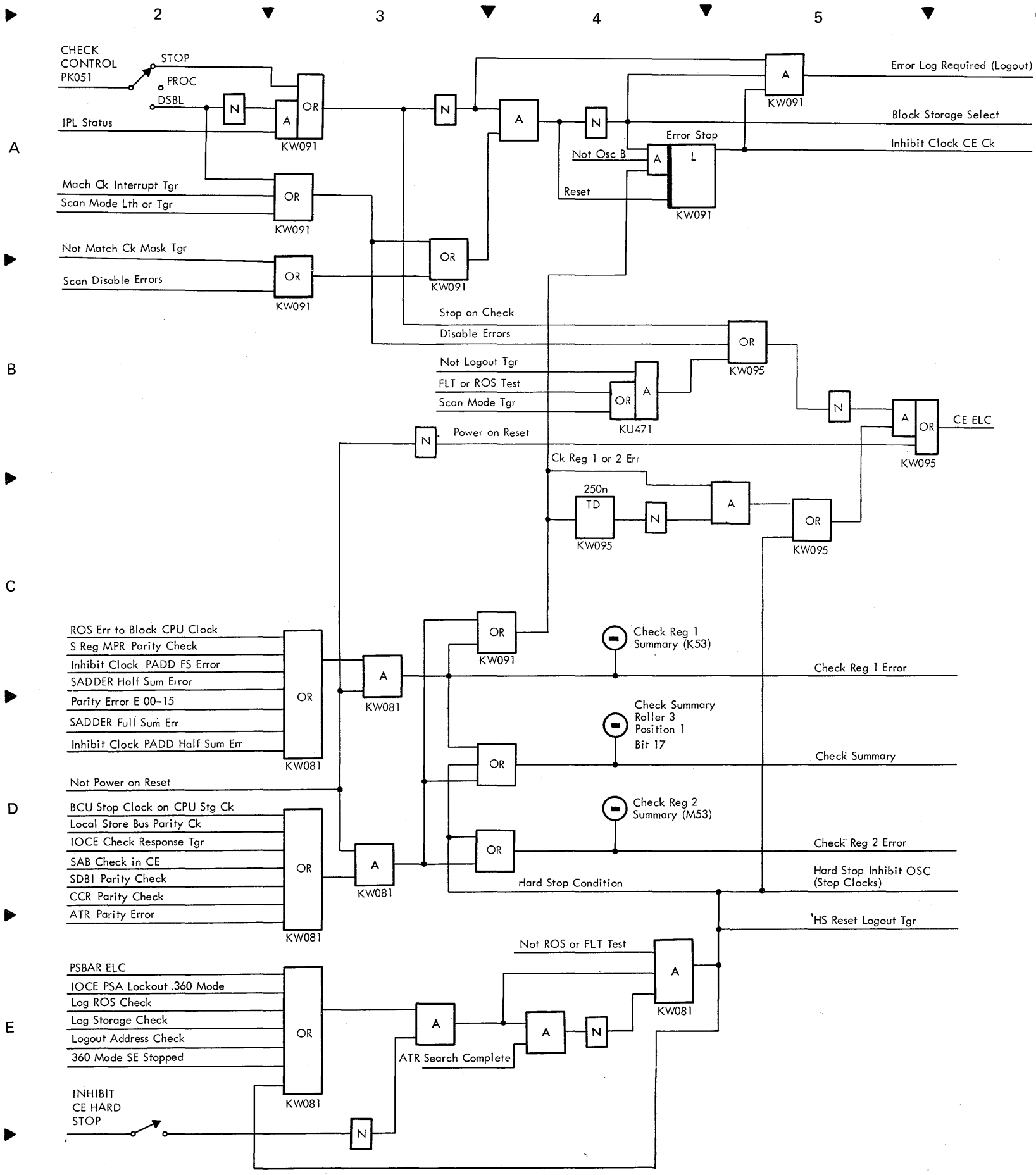


Diagram 6-22. CE Check Control and Inhibit CE Hardstop Switches, Logic and Error Controls

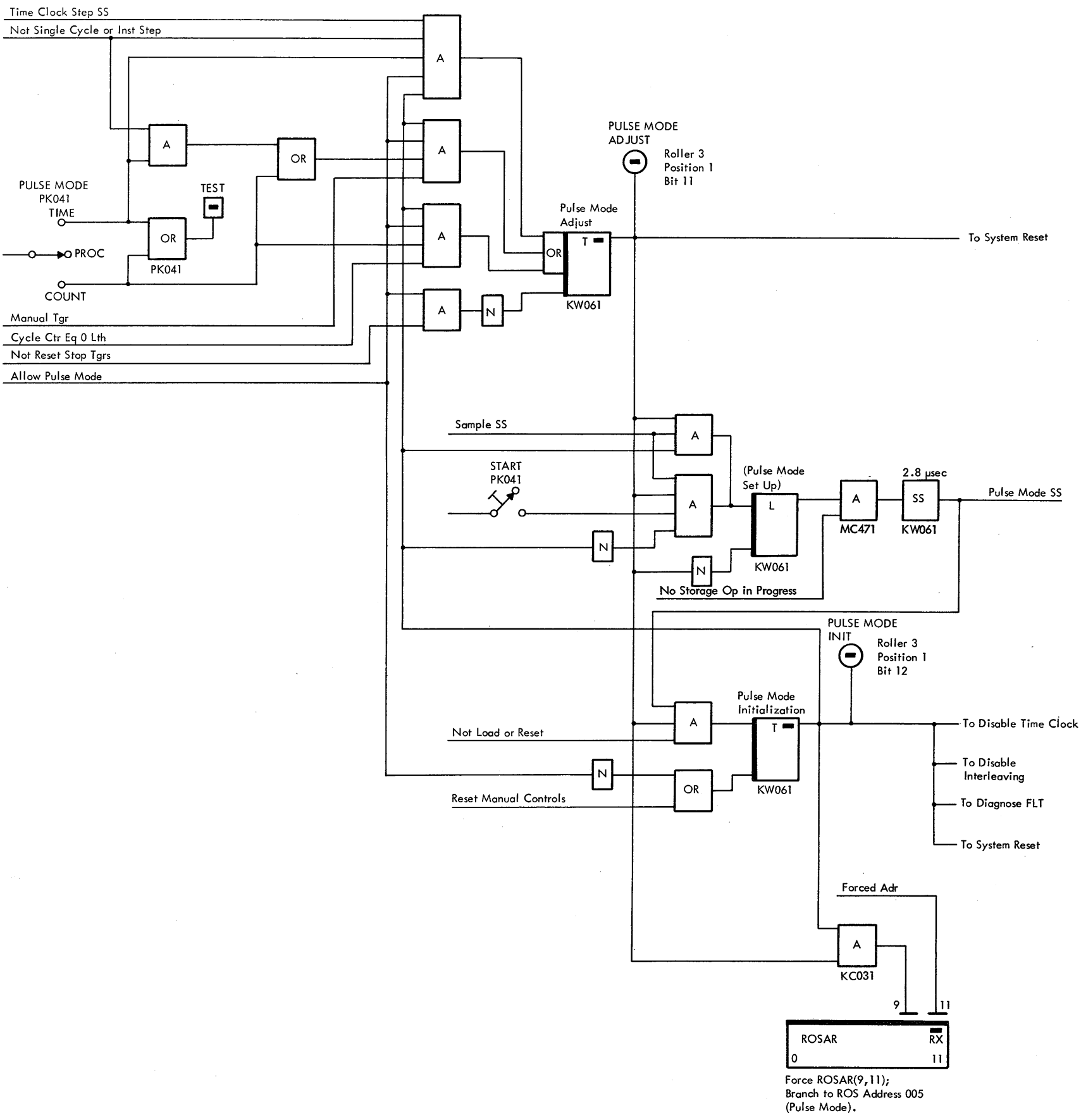
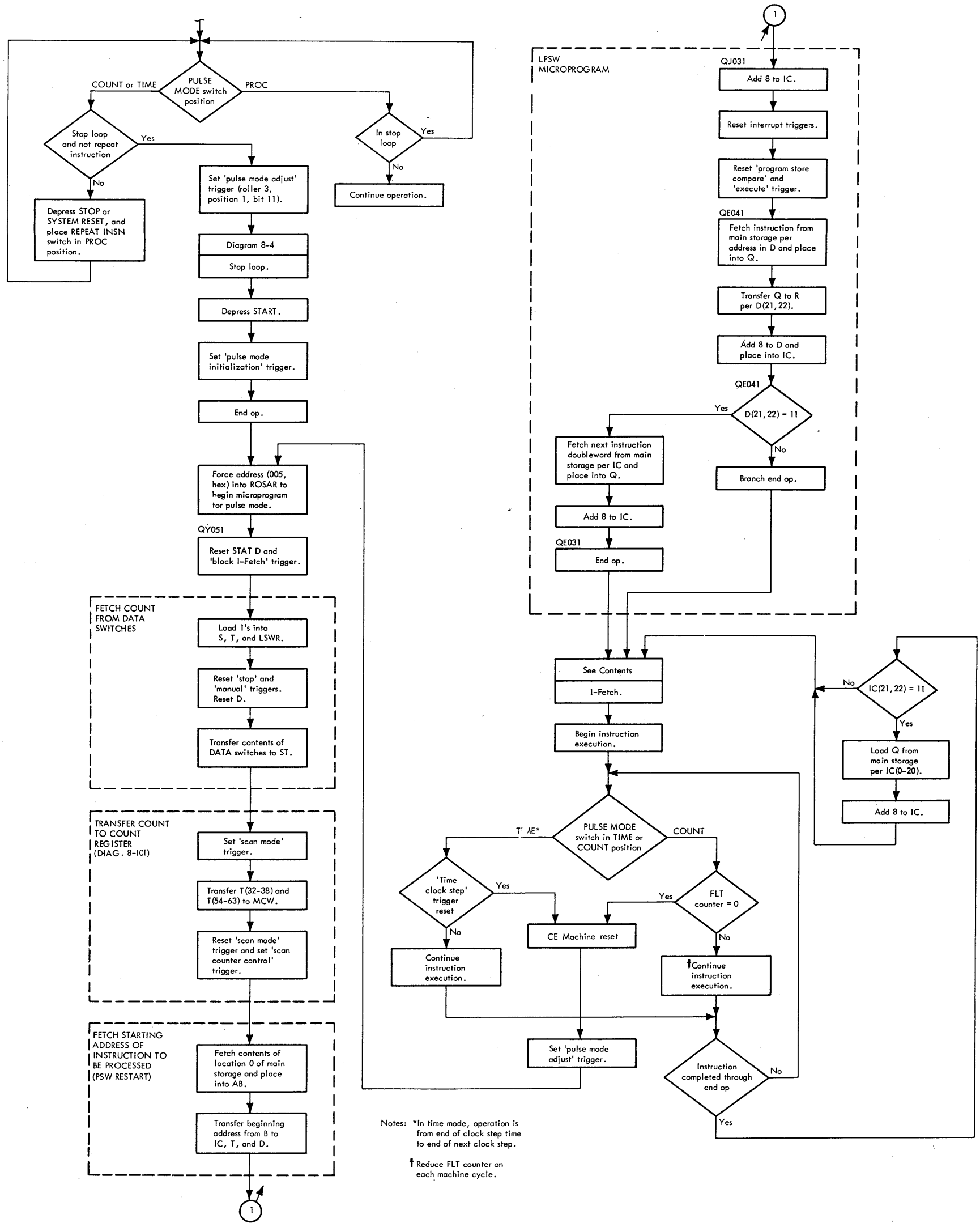


Diagram 6-23. Pulse Mode Controls



Notes: *In time mode, operation is from end of clock step time to end of next clock step.

† Reduce FLT counter on each machine cycle.

Diagram 6-24. Pulse Mode Operation

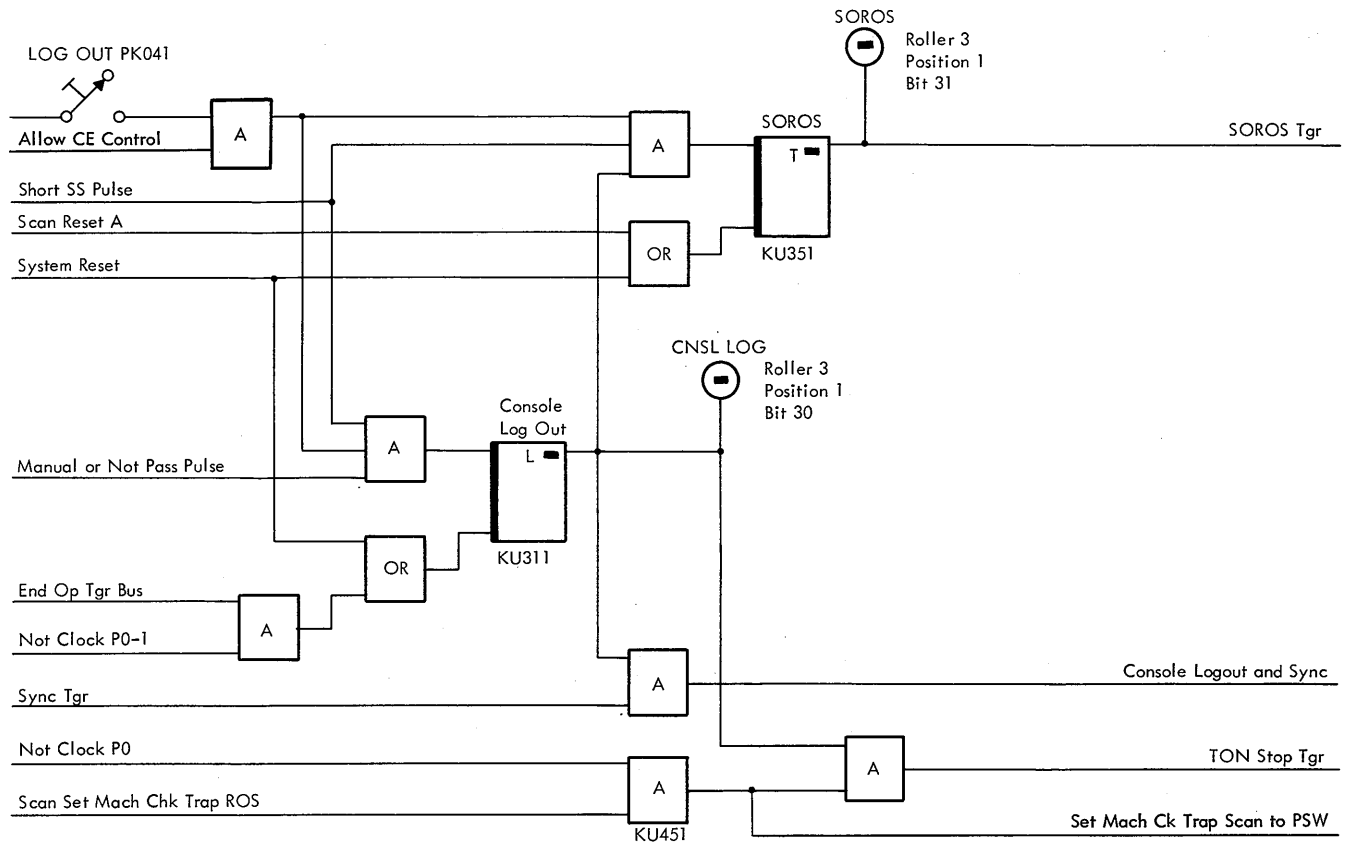


Diagram 6-25. LOG OUT Pushbutton Logic

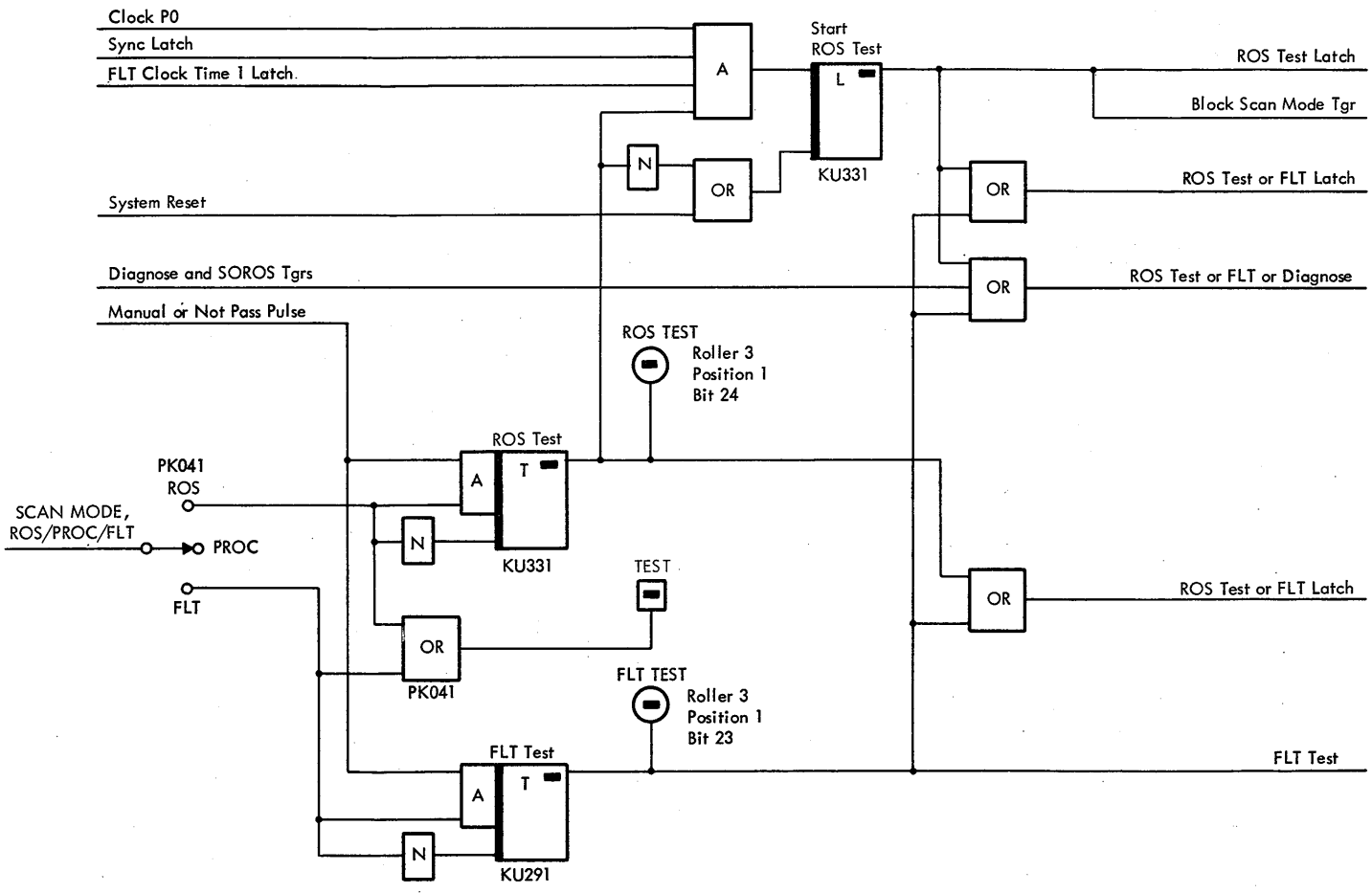


Diagram 6-26. SCAN MODE, ROS/PROC/FLT Switch Logic

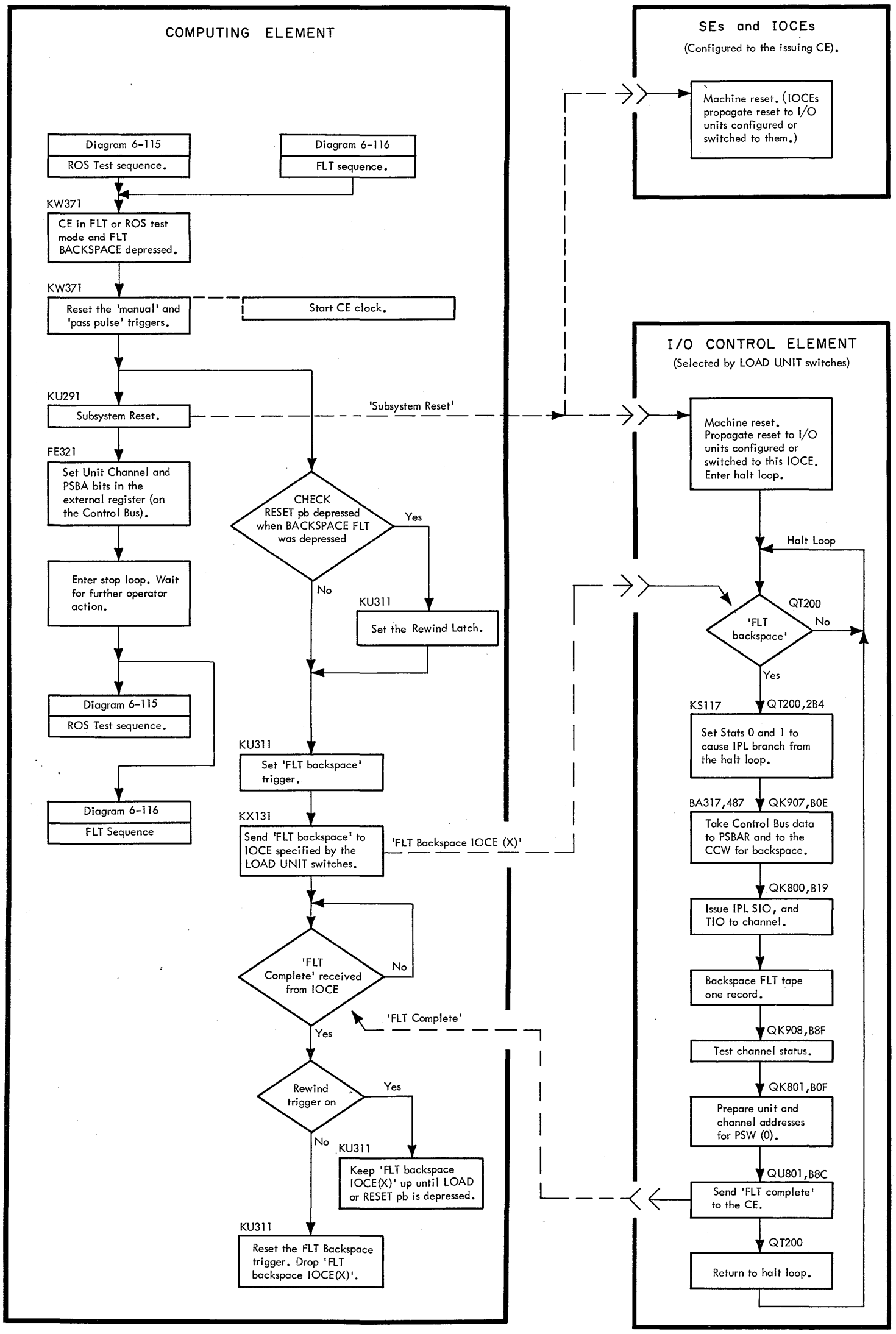


Diagram 6-27. FLT BACKSPACE Pushbutton Logic and Flow

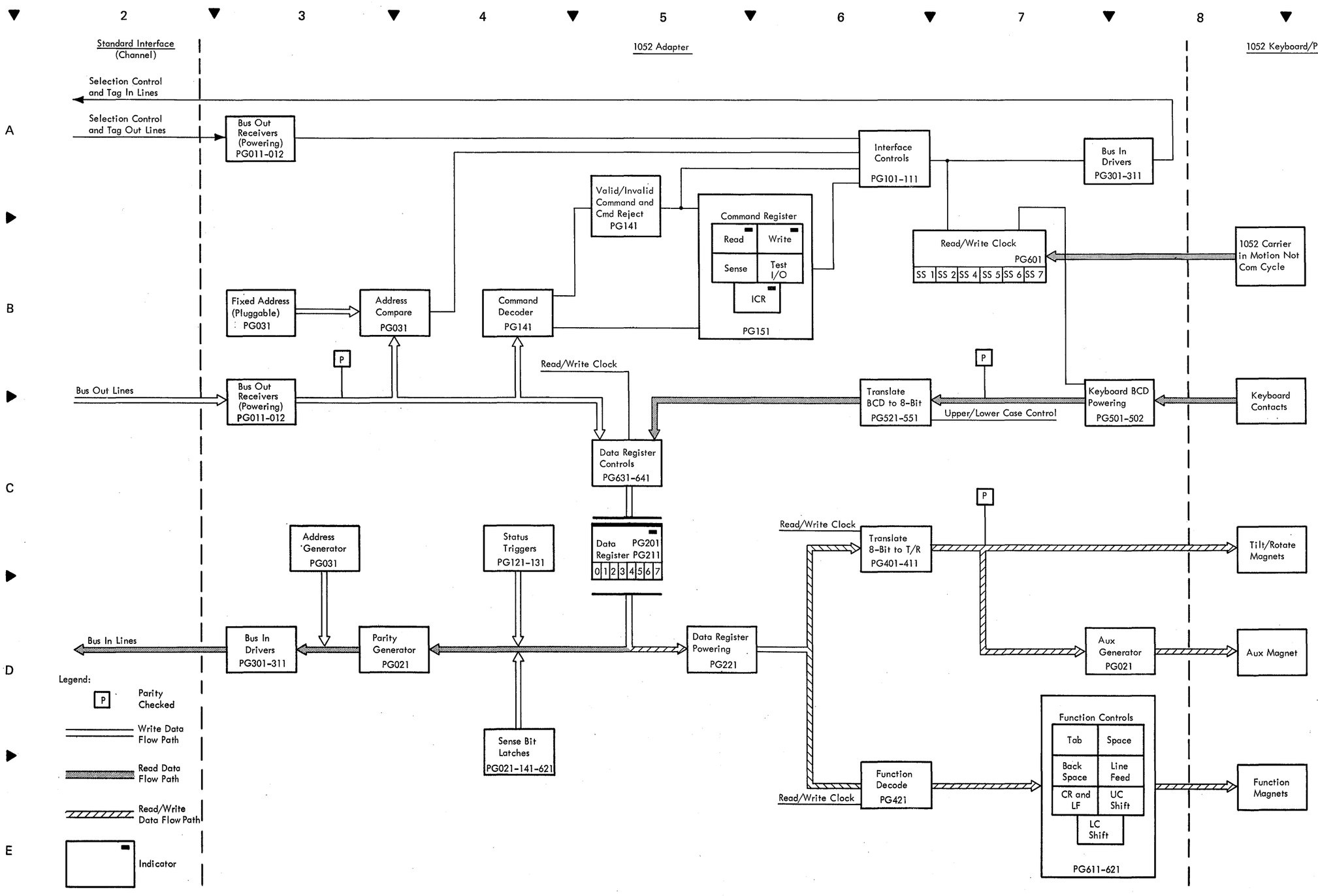


Diagram 6-28. 1052 Adapter Unit

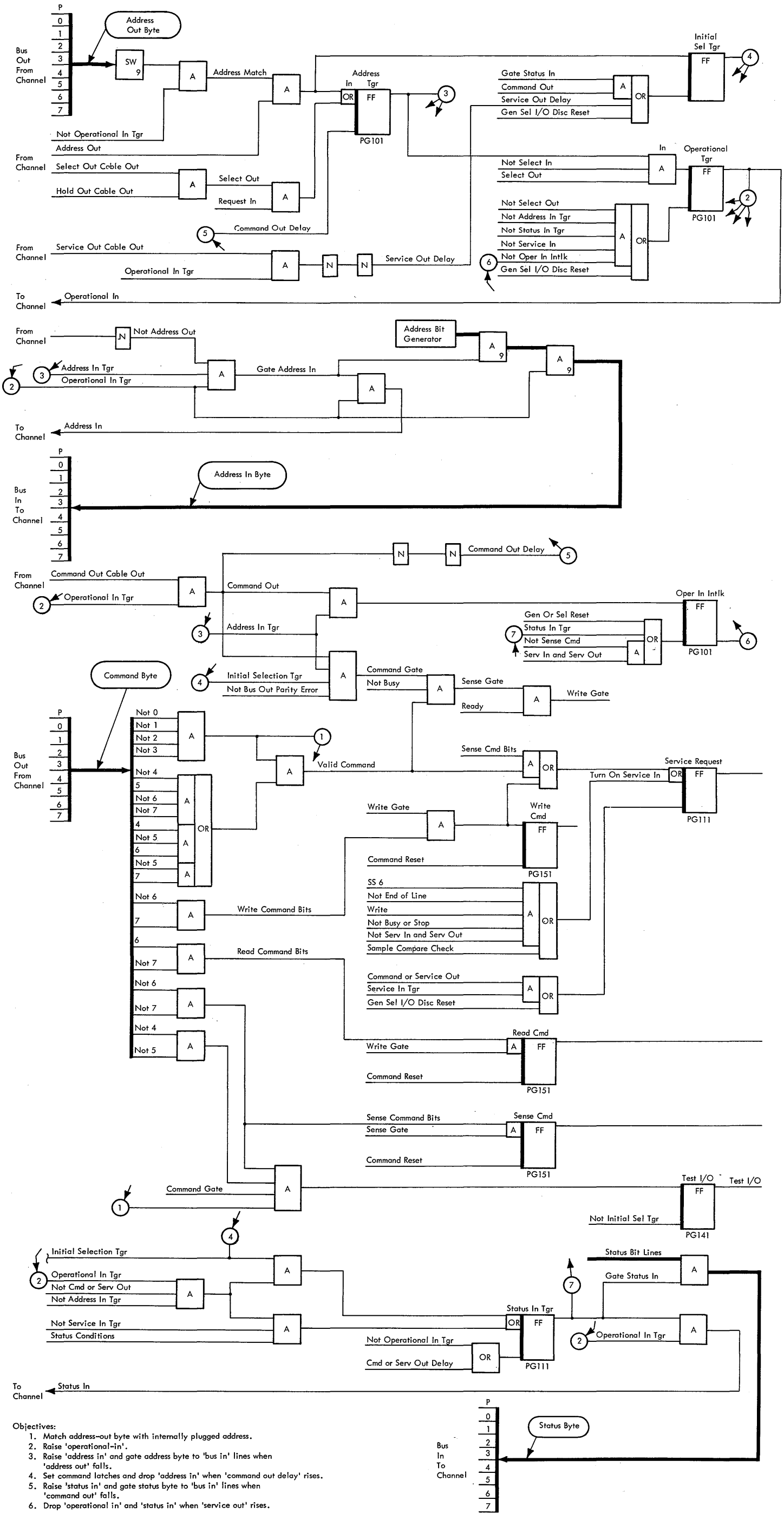


Diagram 6-29. 1052 Adapter Initial Selection – Read, Write, Sense

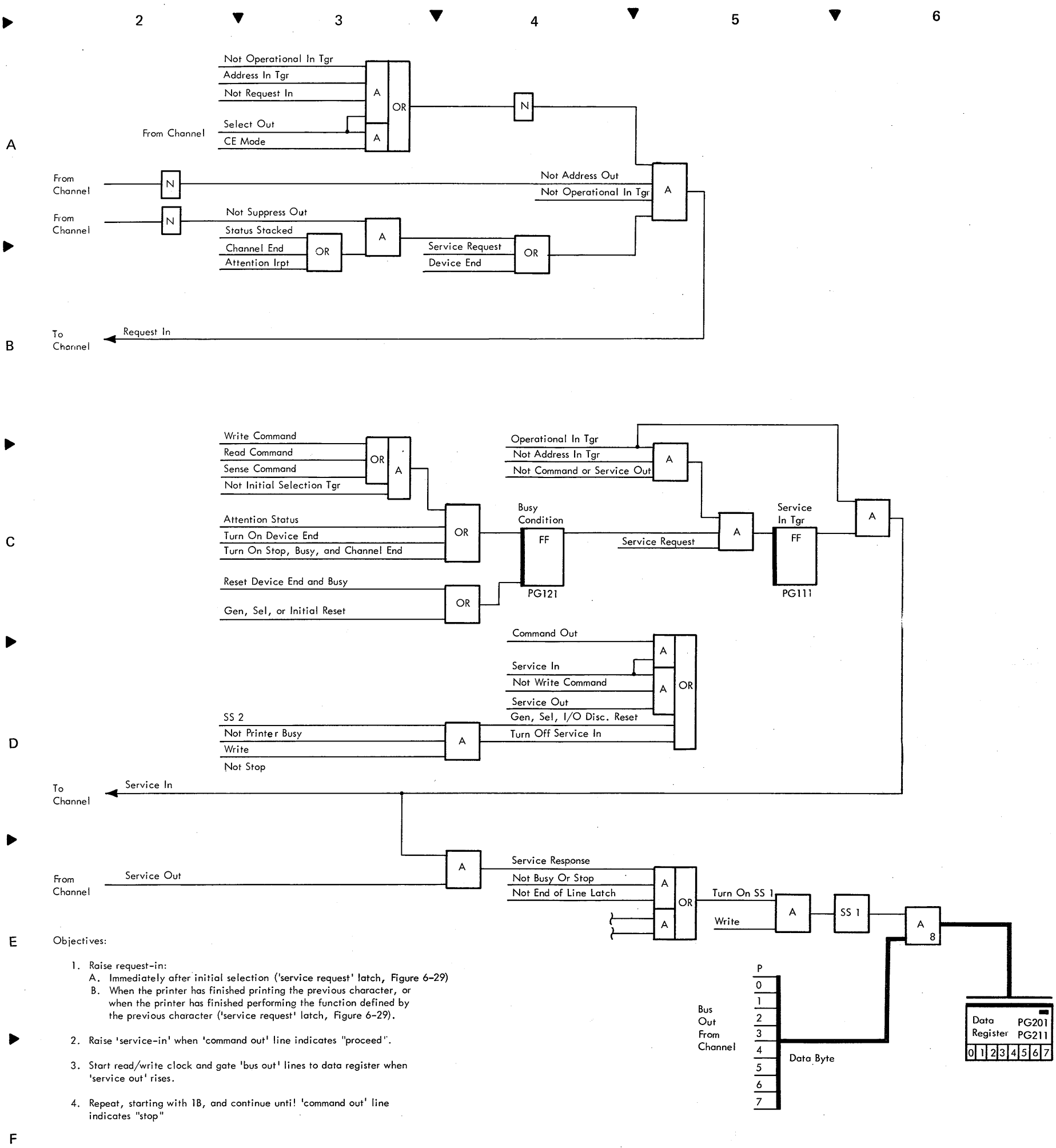
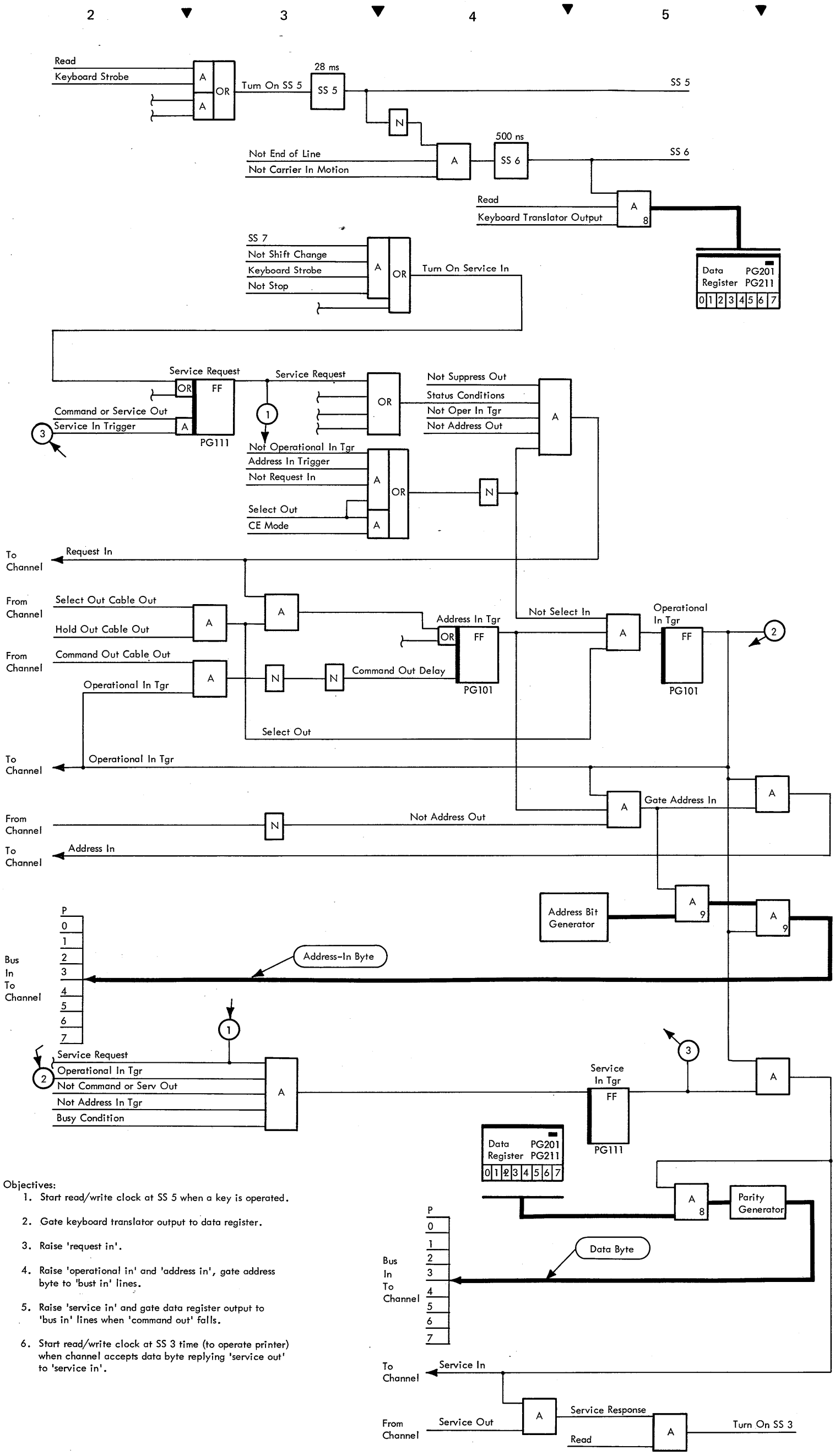


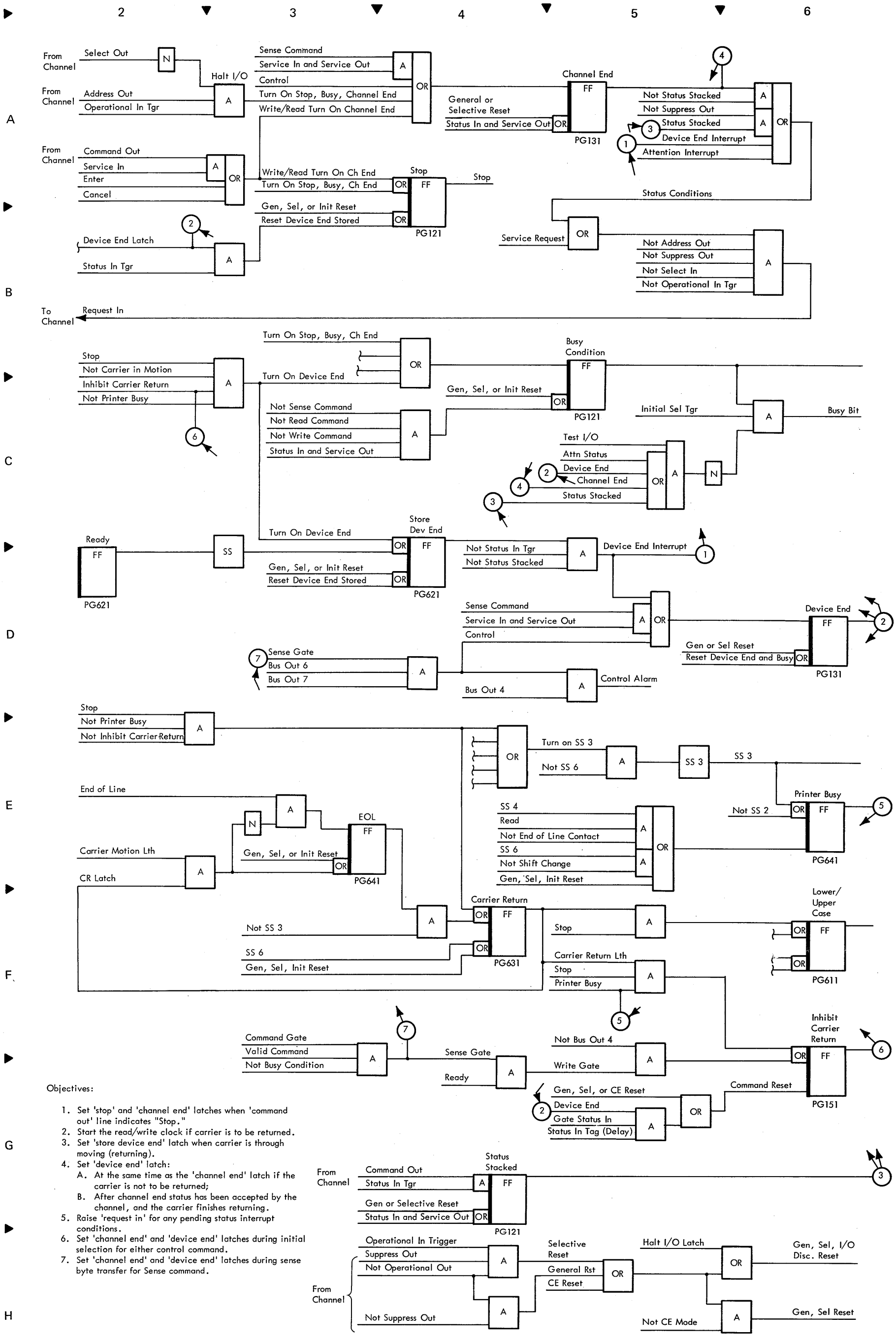
Diagram 6-30. 1052 Adapter Data Transfer -- Write



Objectives:

1. Start read/write clock at SS 5 when a key is operated.
2. Gate keyboard translator output to data register.
3. Raise 'request in'.
4. Raise 'operational in' and 'address in', gate address byte to 'bust in' lines.
5. Raise 'service in' and gate data register output to 'bus in' lines when 'command out' falls.
6. Start read/write clock at SS 3 time (to operate printer) when channel accepts data byte replying 'service out' to 'service in'.

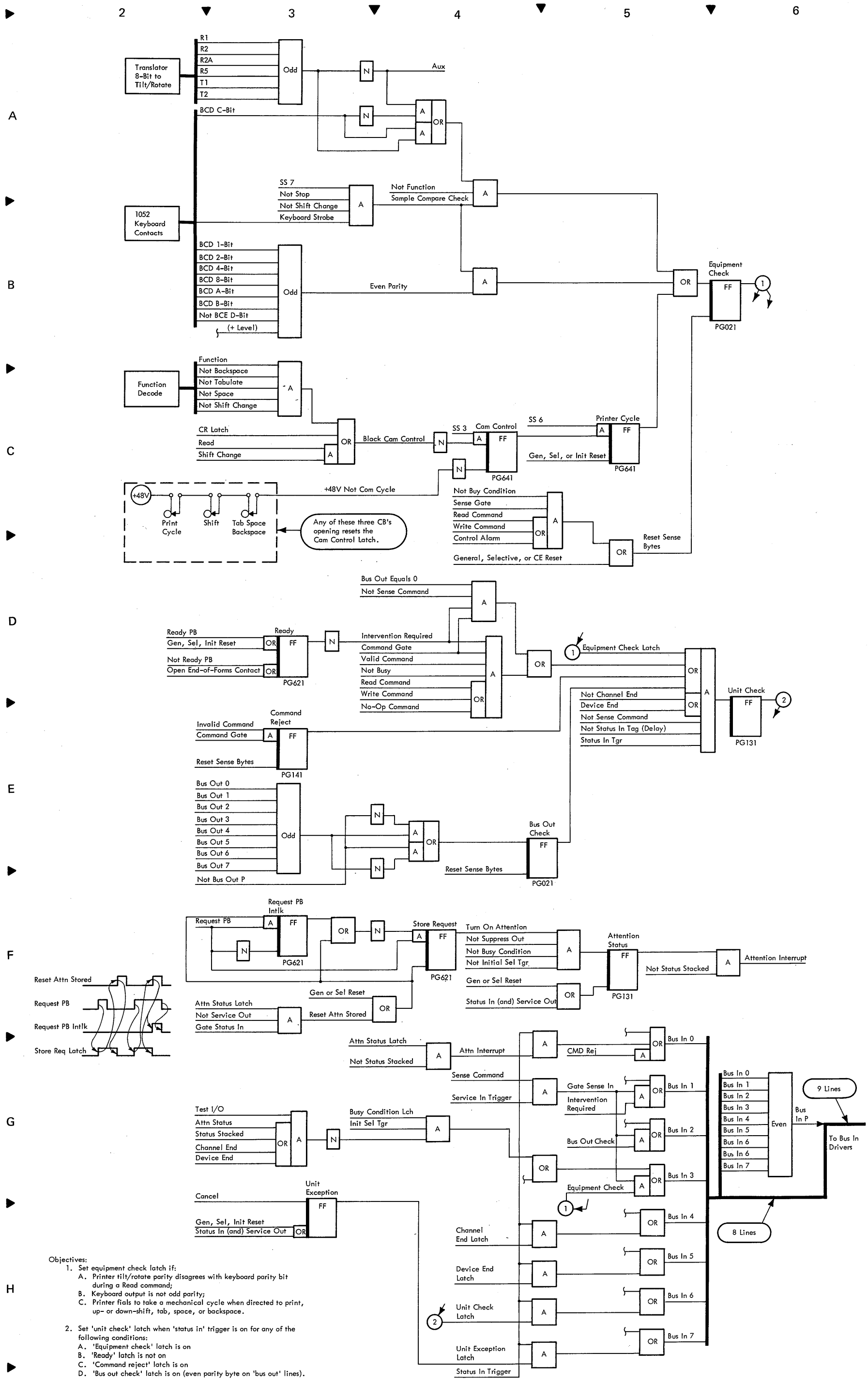
Diagram 6-31. 1052 Adapter Data Transfer – Read



Objectives:

1. Set 'stop' and 'channel end' latches when 'command out' line indicates "Stop."
2. Start the read/write clock if carrier is to be returned.
3. Set 'store device end' latch when carrier is through moving (returning).
4. Set 'device end' latch:
   A. At the same time as the 'channel end' latch if the carrier is not to be returned;
   B. After channel end status has been accepted by the channel, and the carrier finishes returning.
5. Raise 'request in' for any pending status interrupt conditions.
6. Set 'channel end' and 'device end' latches during initial selection for either control command.
7. Set 'channel end' and 'device end' latches during sense byte transfer for Sense command.

Diagram 6-32. 1052 Adapter Ending Sequence

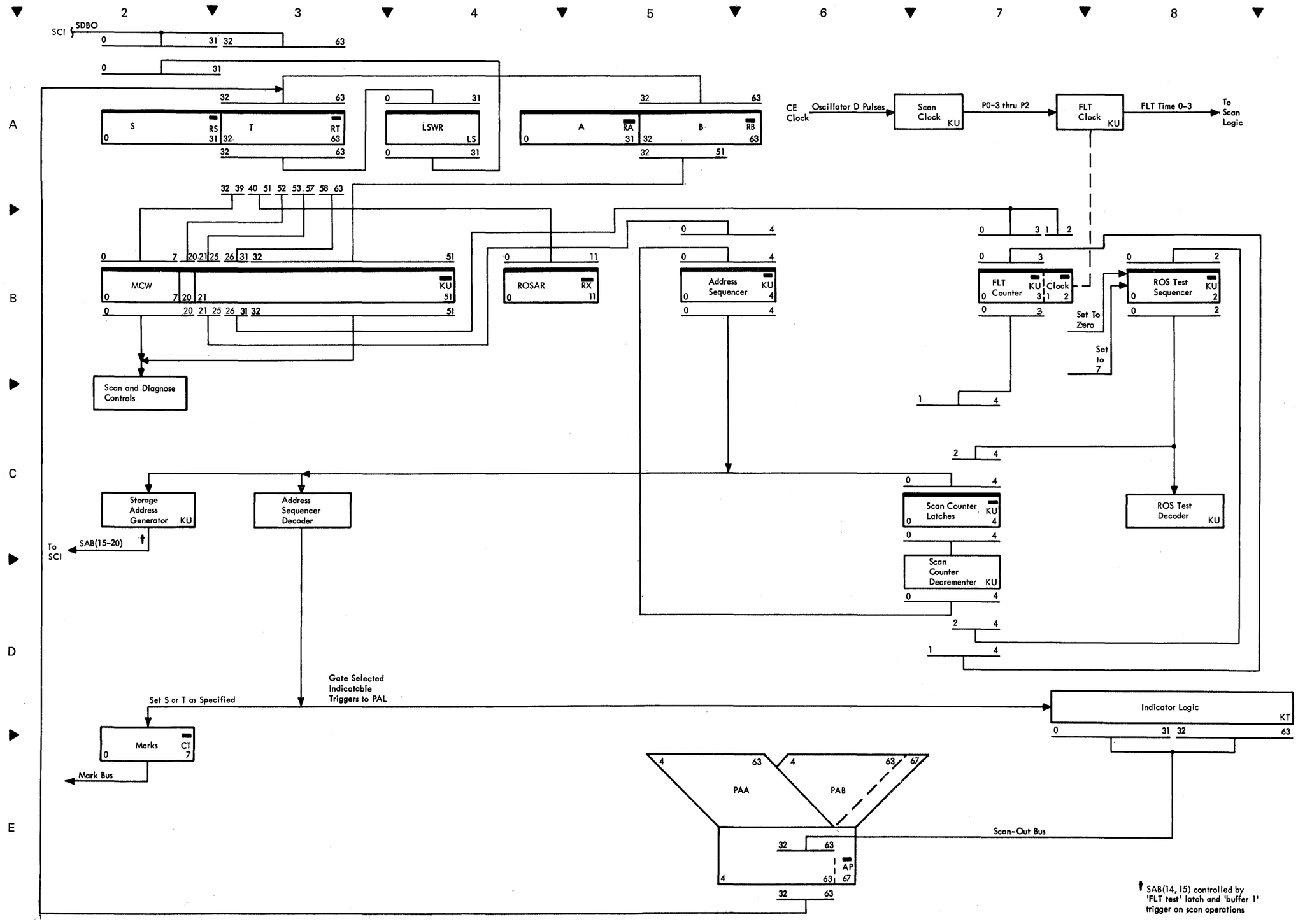


Objectives:

1. Set equipment check latch if:
   A. Printer tilt/rotate parity disagrees with keyboard parity bit during a Read command;
   B. Keyboard output is not odd parity;
   C. Printer fials to take a mechanical cycle when directed to print, up- or down-shift, tab, space, or backspace.

2. Set 'unit check' latch when 'status in' trigger is on for any of the following conditions:
   A. 'Equipment check' latch is on
   B. 'Ready' latch is not on
   C. 'Command reject' latch is on
   D. 'Bus out check' latch is on (even parity byte on 'bus out' lines).

Diagram 6-33. 1052 Adapter Sense and Status Bytes


2
3
4
5
6
7
8
9
A
B
C
D
E
SCI
SDBO
S
RS
T
RT
LSWR
LS
A
RA
B
RB
MCW
KU
ROSAR
RX
Address Sequencer
KU
CE Clock
Oscillator D Pulses
Scan Clock
KU
P0-3 thru P2
FLT Clock
KU
FLT Time 0-3
To Scan Logic
FLT Counter
KU
Clock
Set To Zero
Set to 7
ROS Test Sequencer
KU
ROS Test Decoder
KU
Scan and Diagnose Controls
Storage Address Generator
KU
To SCI
SAB(15-20)
Address Sequencer Decoder
Scan Counter Latches
KU
Scan Counter Decrementer
KU
Gate Selected Indicatable Triggers to PAL
Set S or T as Specified
Marks
CT
Mark Bus
Indicator Logic
KT
PAA
PAB
AP
Scan-Out Bus
SAB(14, 15) controlled by 'FLT test' latch and 'buffer 1' trigger on scan operations

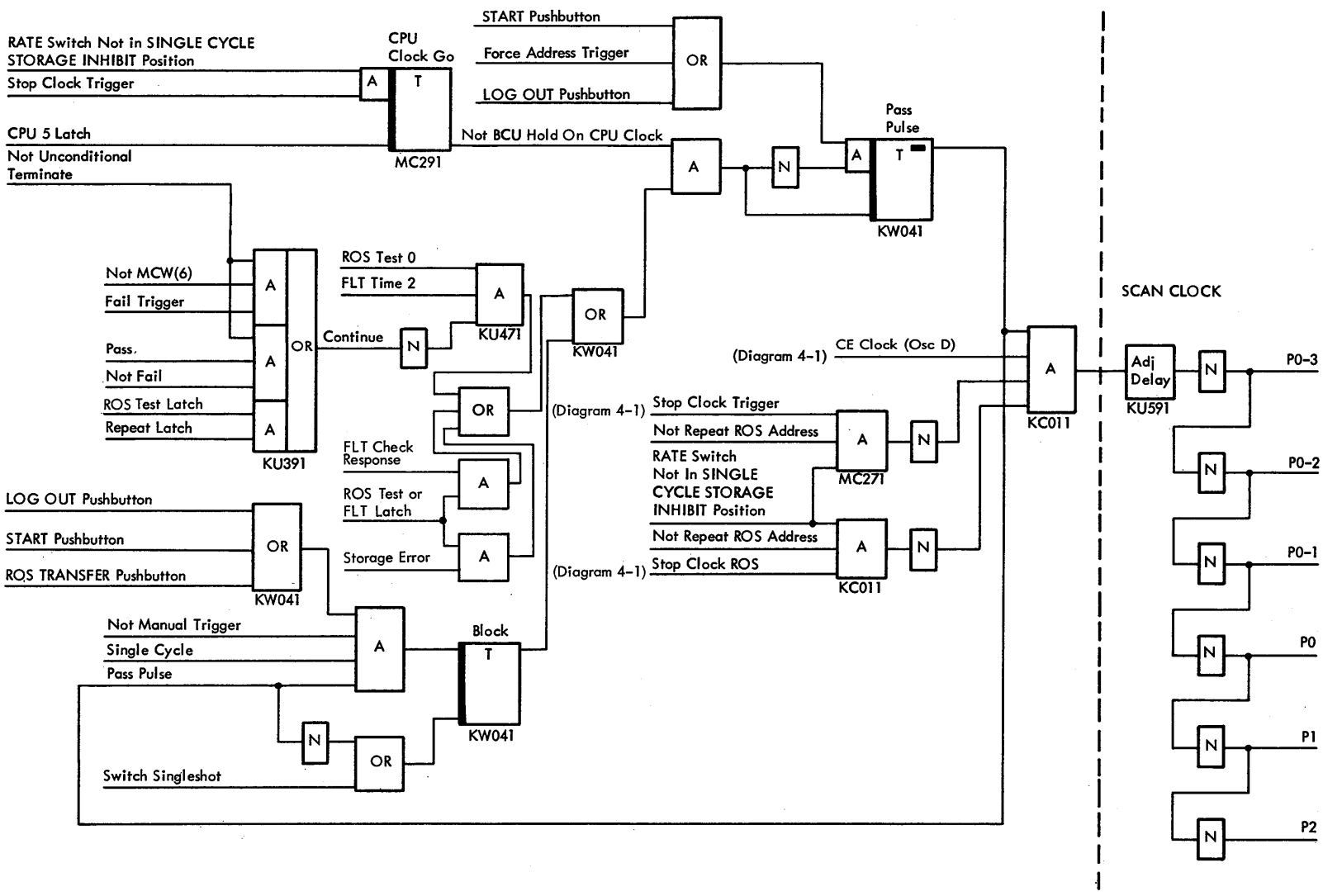
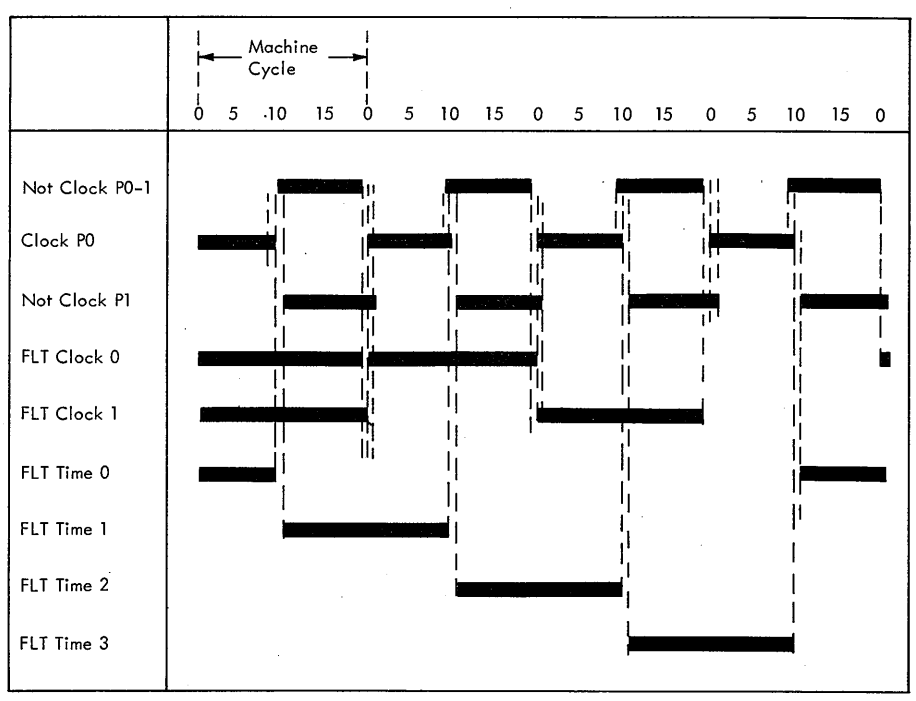


Diagram 6-102. Scan Clock



Note: All clock and not-clock signals to FLT clock are from scan clock.

† In the ALD, this signal is 'gate ST bus to scan ctr'.

Diagram 6-103. FLT Clock

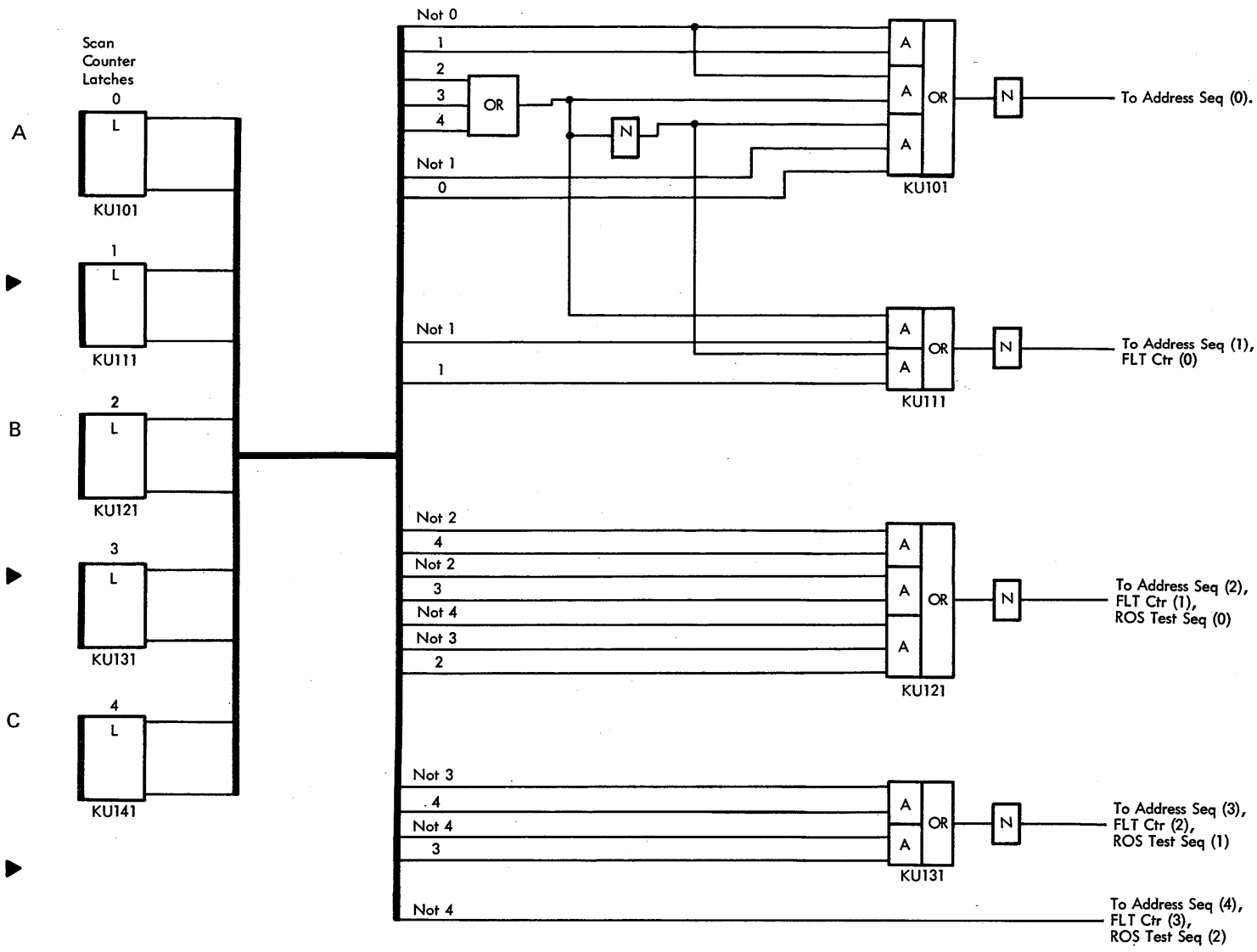
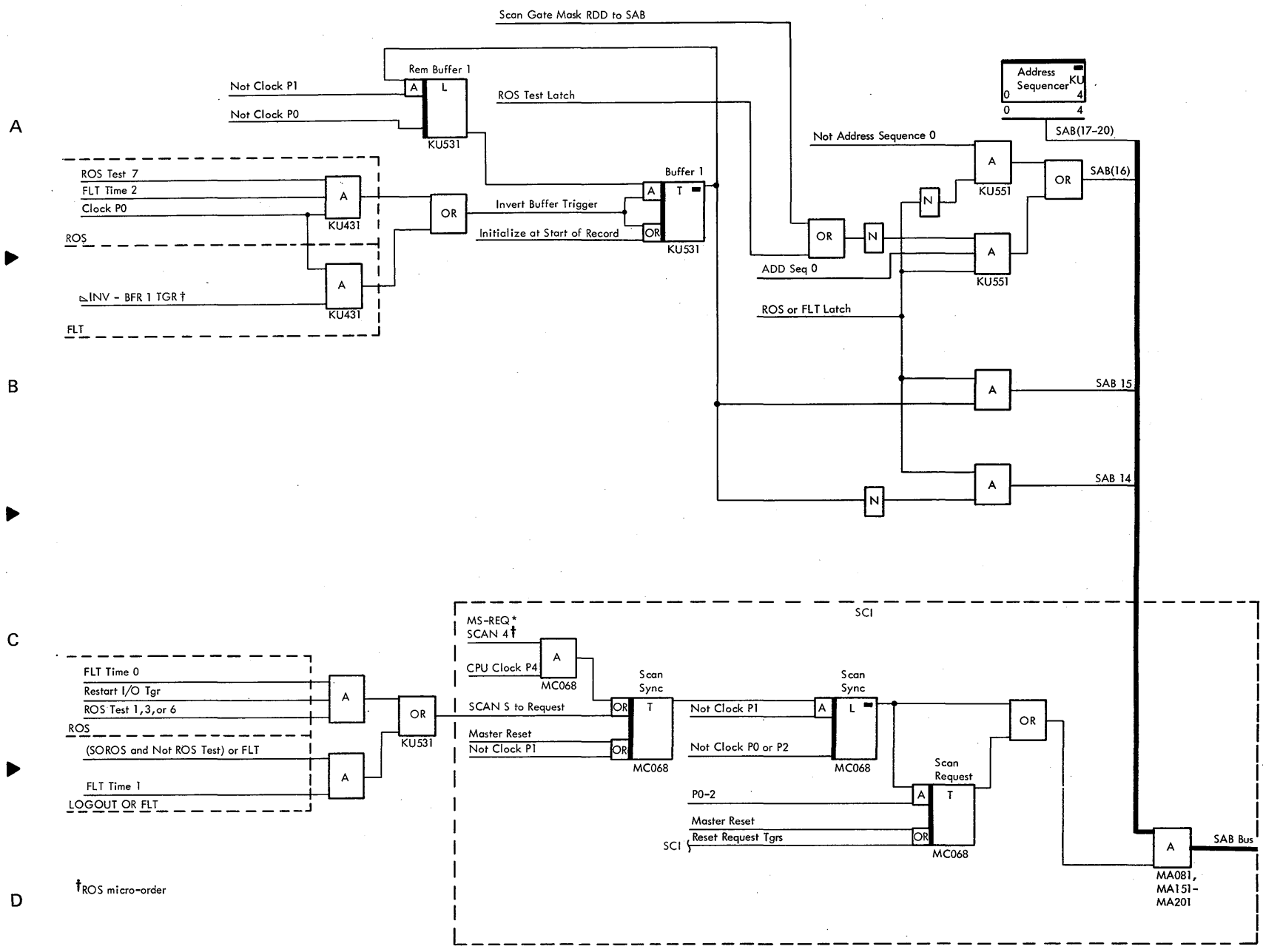


Diagram 6-104. Scan Counter Latches and Decrementer



SAB

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Logout 80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Logout F8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0
Logout 100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Logout 138	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	0	0
Buffer 1:																								
FLT's 100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
FLT's 180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
Buffer 2:																								
FLT's 200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
FLT's 280	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0

Not Address Sequence 0 — Address Sequencer

OFF Buffer 1 Trigger — ON Address Sequencer

Note: Scan-generated main storage addresses start with the highest value required and are decremented to the lowest value required.

Diagram 6-105. Scan Storage Address Generator

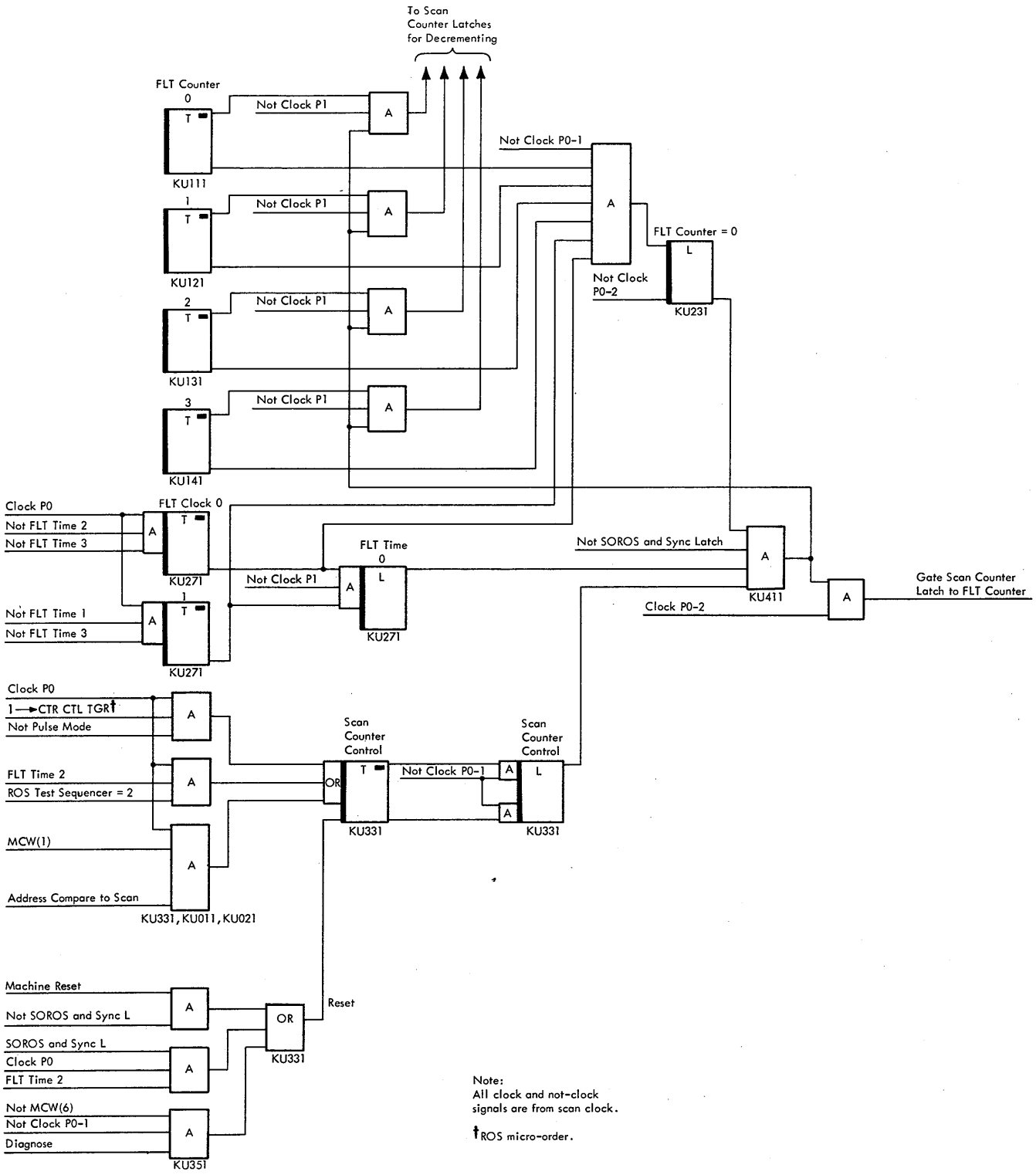
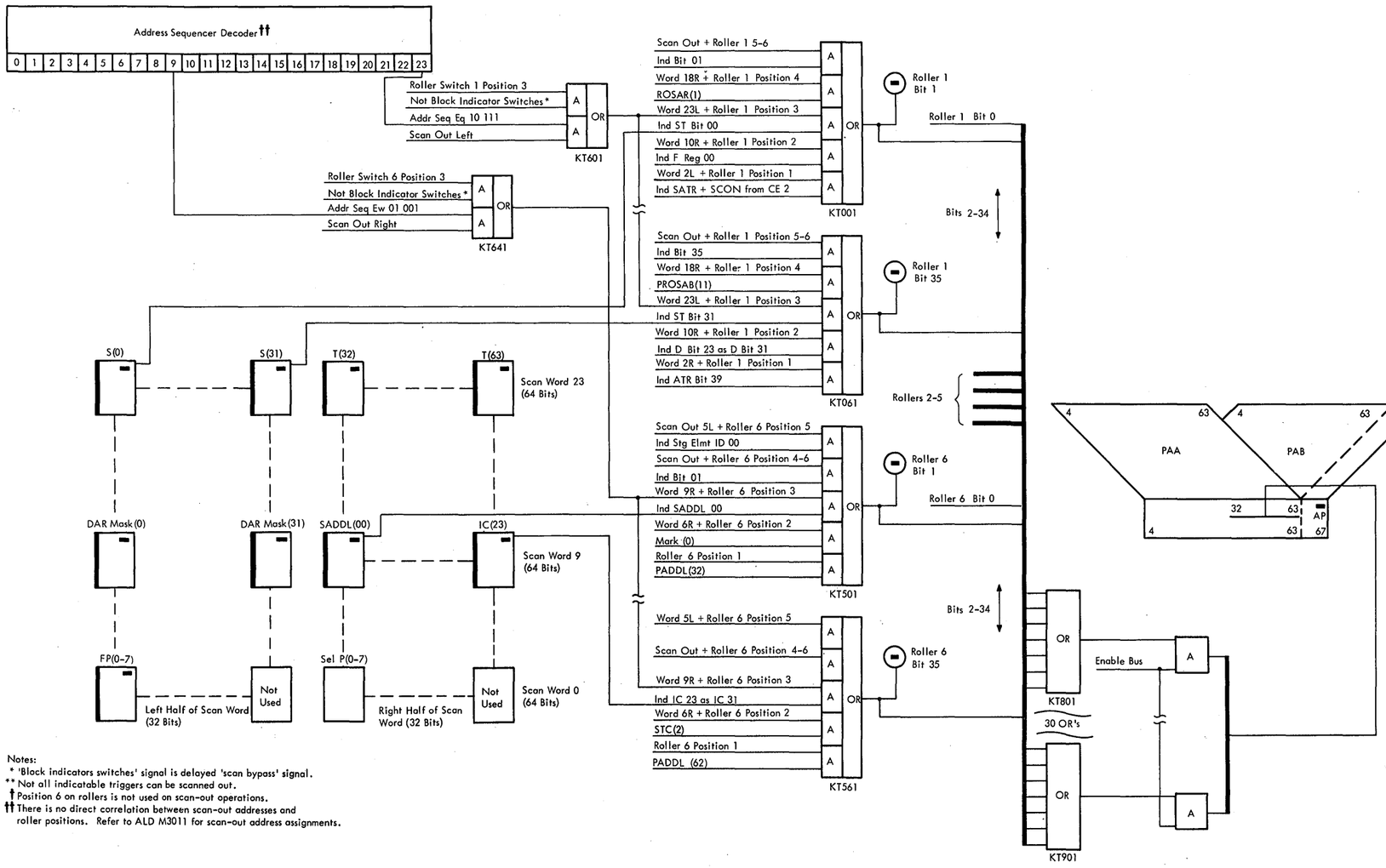


Diagram 6-106. FLT Counter Decrementing



Notes:
* 'Block indicators switches' signal is delayed 'scan bypass' signal.
** Not all indicatable triggers can be scanned out.
† Position 6 on rollers is not used on scan-out operations.
†† There is no direct correlation between scan-out addresses and roller positions. Refer to ALD M3011 for scan-out address assignments.

Diagram 6-107. Scan-Out Bus Data Flow

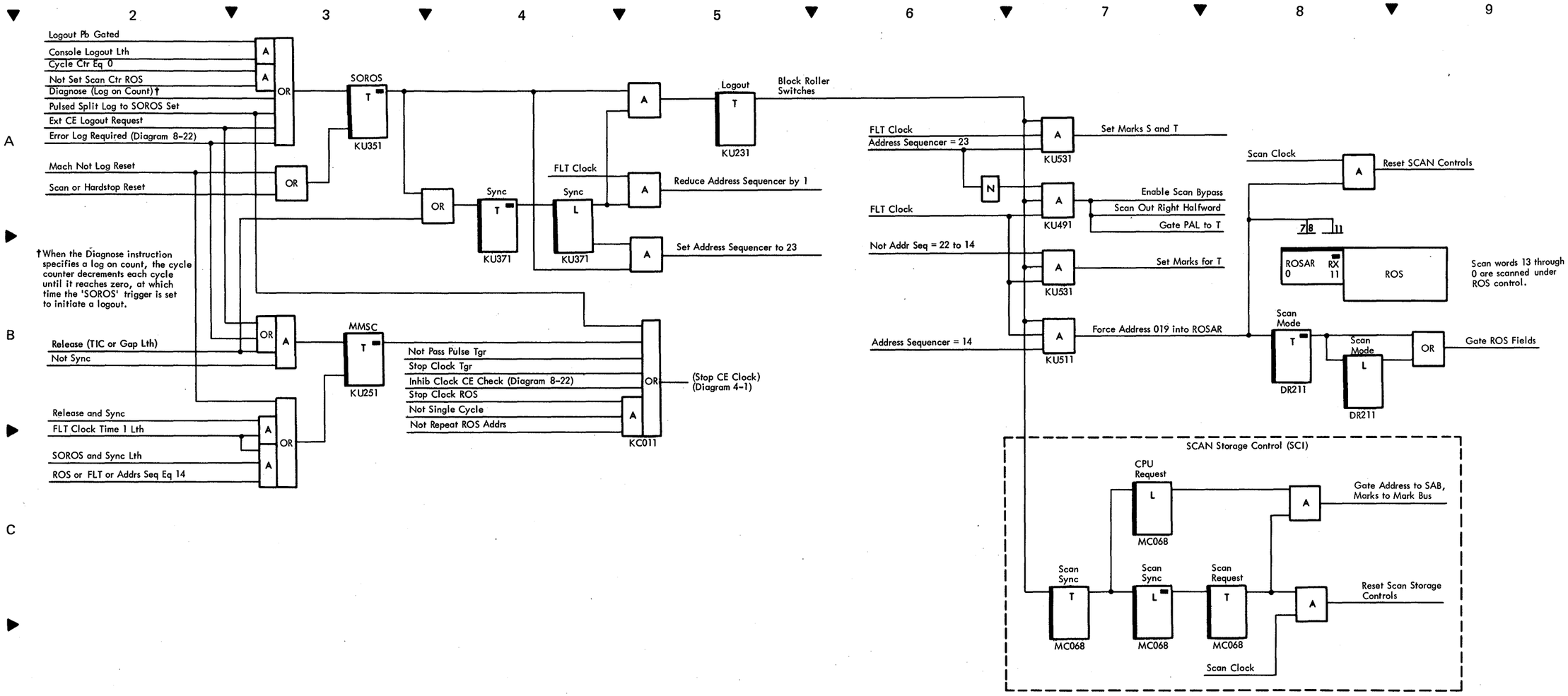


Diagram 6-108. Logout Control Logic

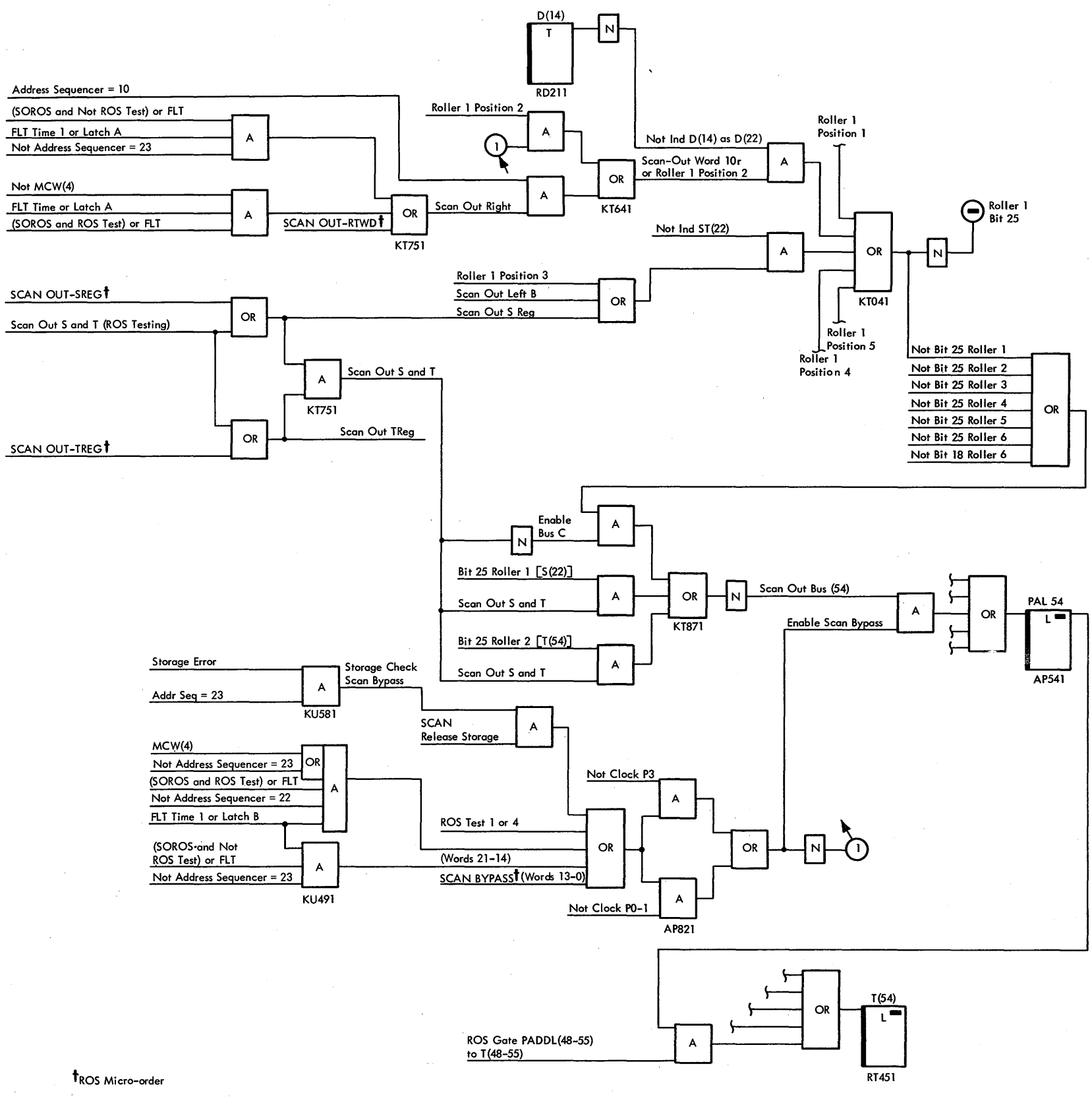


Diagram 6-109. Scan-Out Path For One Bit

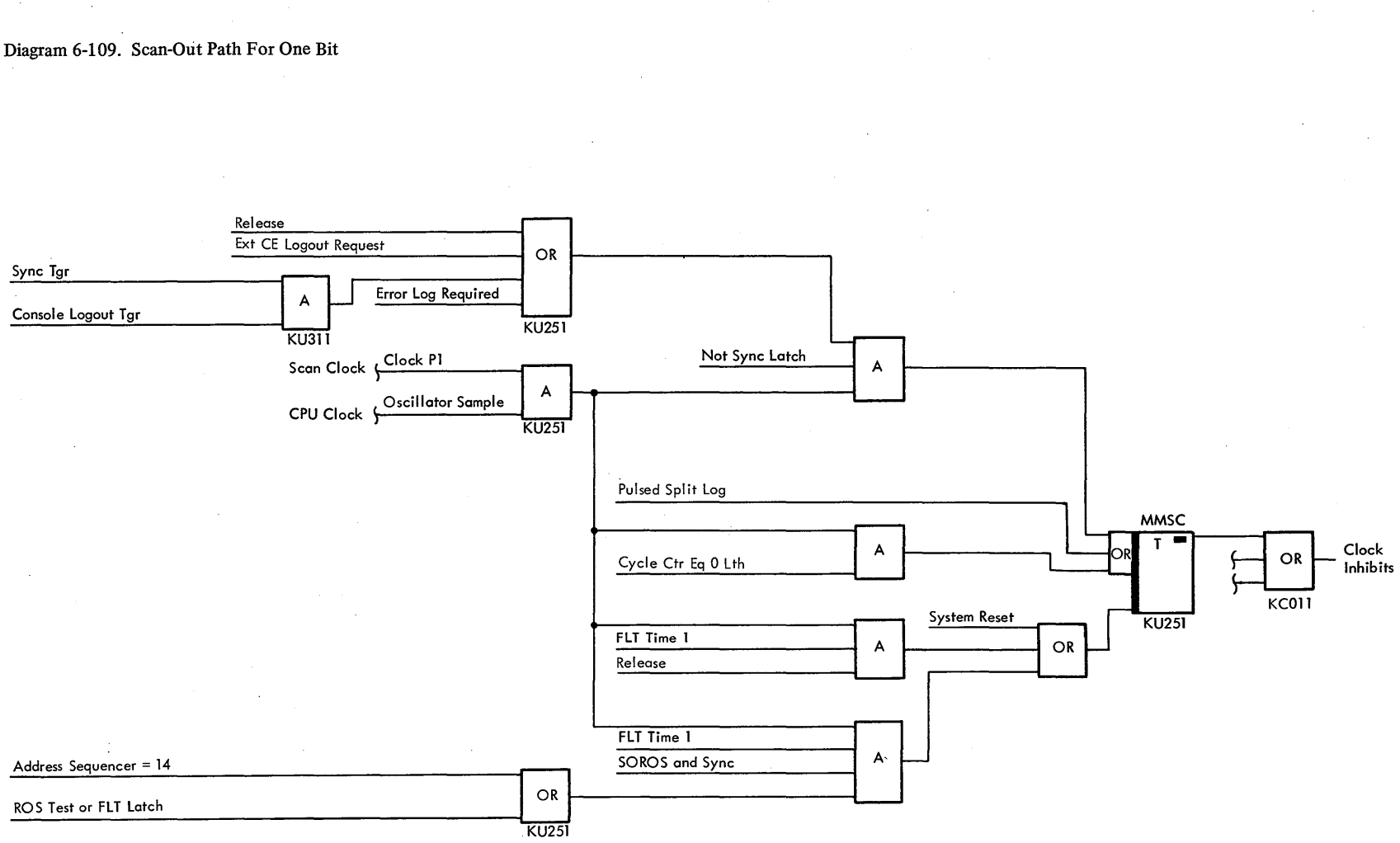


Diagram 6-110. Maintenance Mode Stop Clock Logic

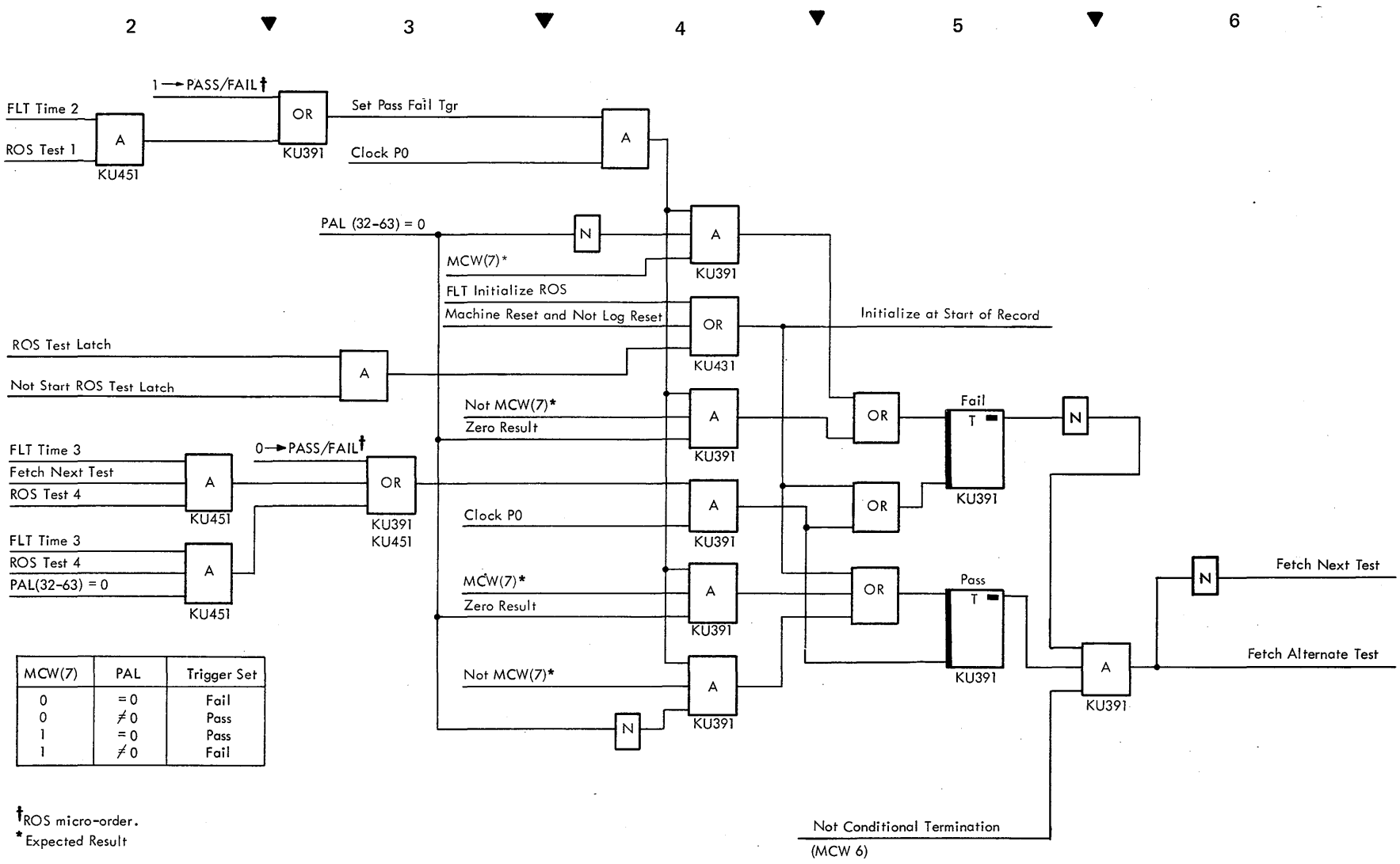


Diagram 6-111. Scan Control Triggers

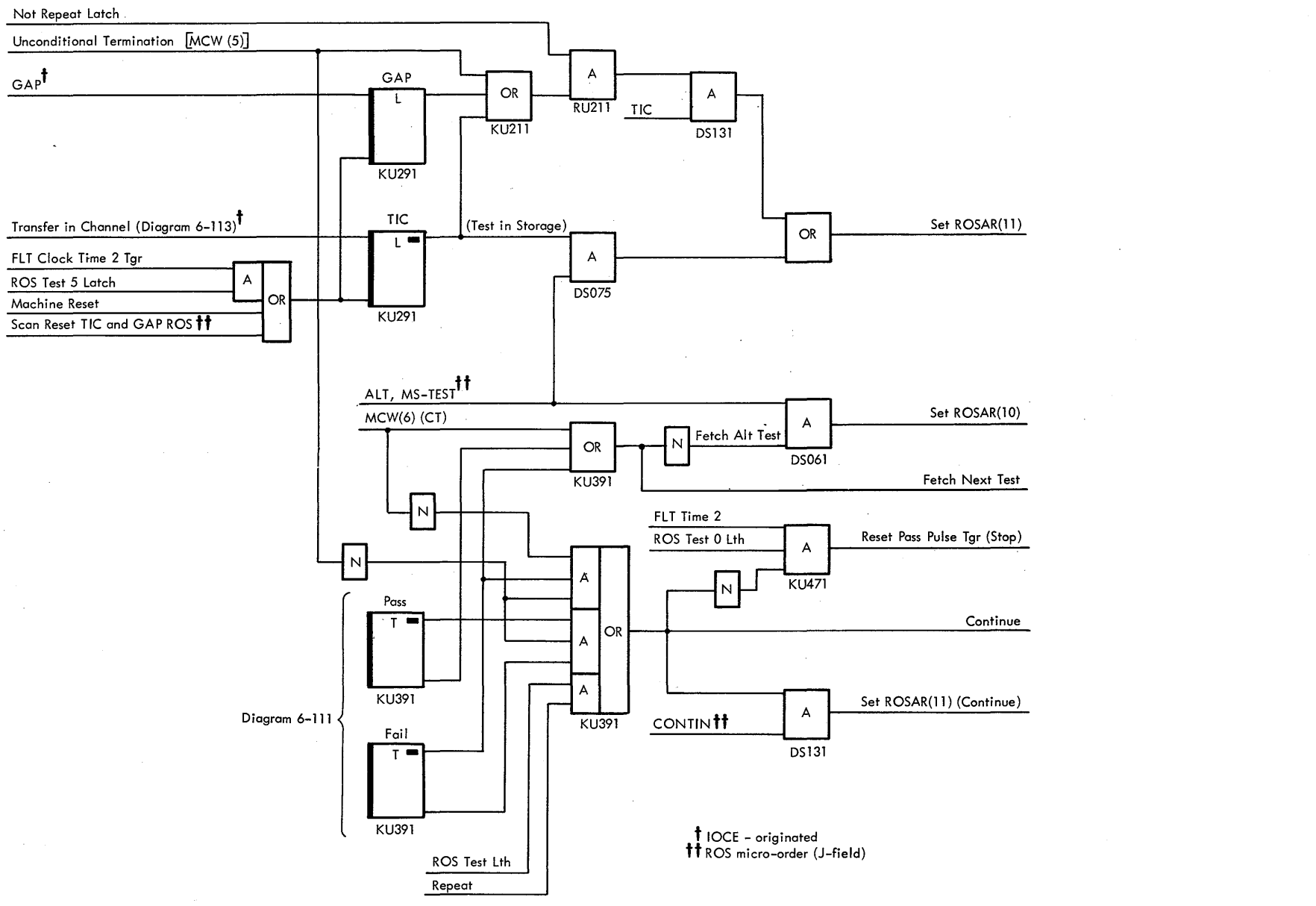


Diagram 6-112. Scan Control of ROS Microbranching

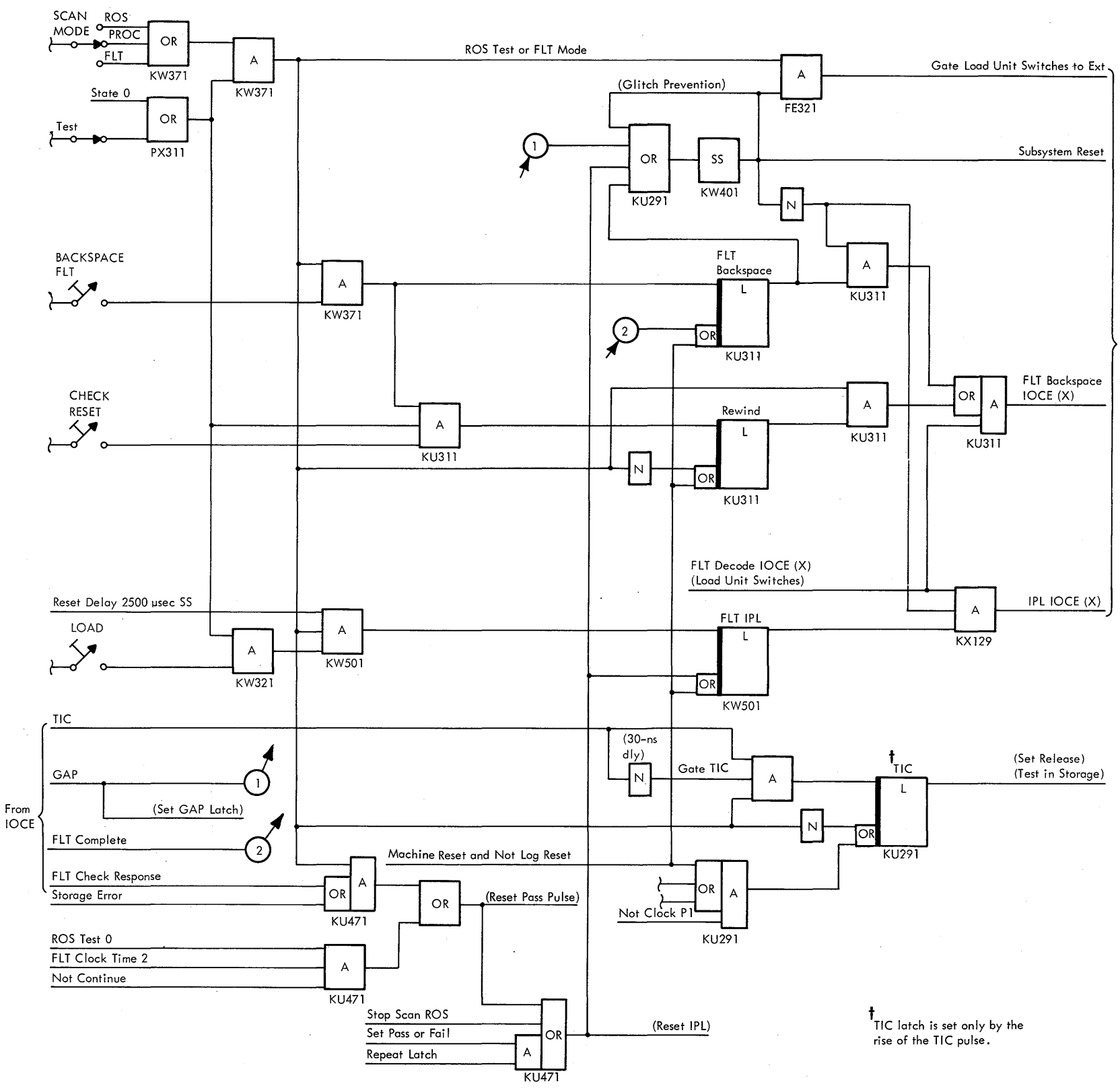


Diagram 6-113. CE Scan/IOCE Interface

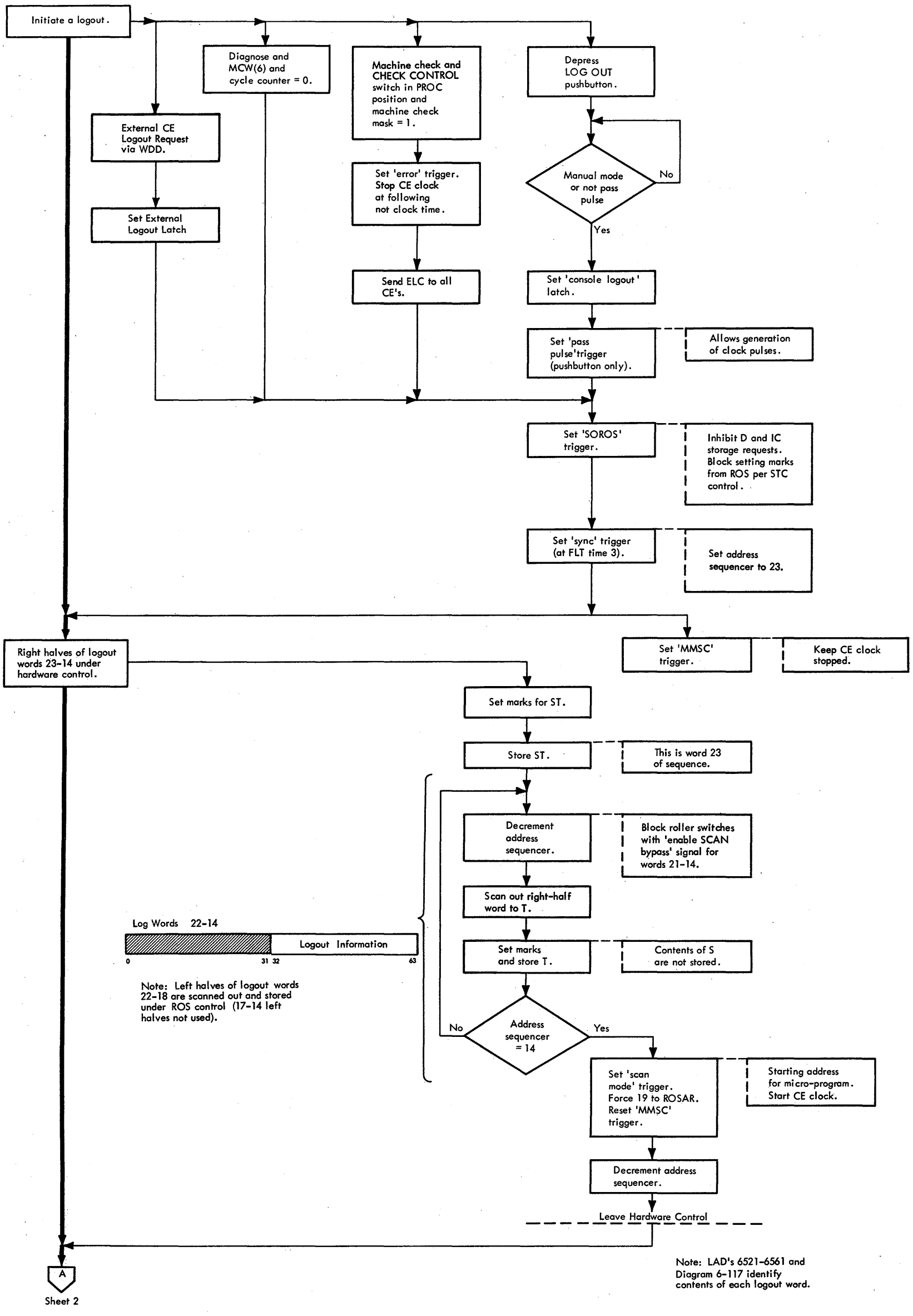


Diagram 6-114. Logout Sequence (Sheet 1 of 2)

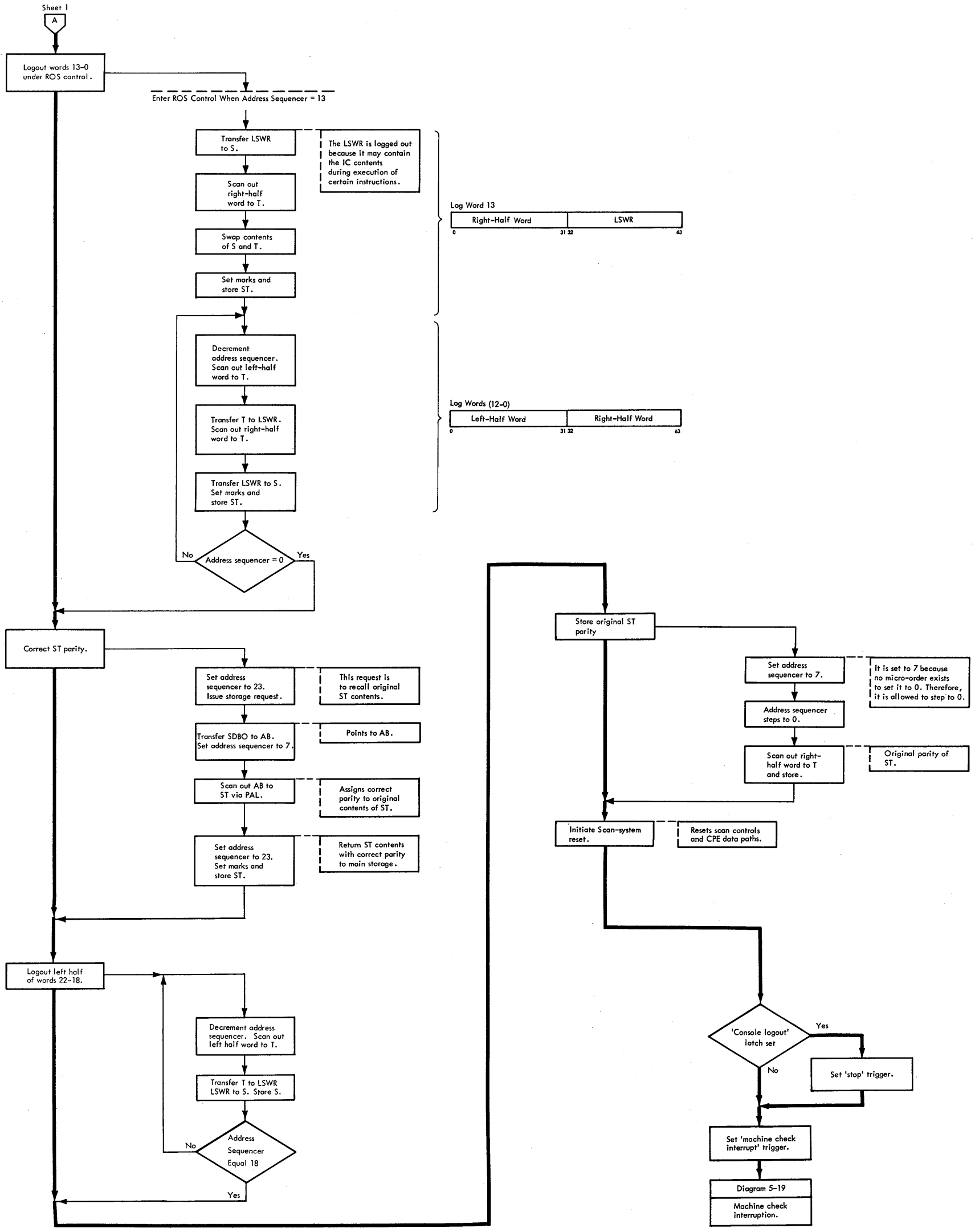


Diagram 6-114. Logout Sequence (Sheet 2 of 2)

2 3 4 5 6

Note: This is a summary flow chart. For detailed ROS test sequence operation refer to sheets 2 through 5.

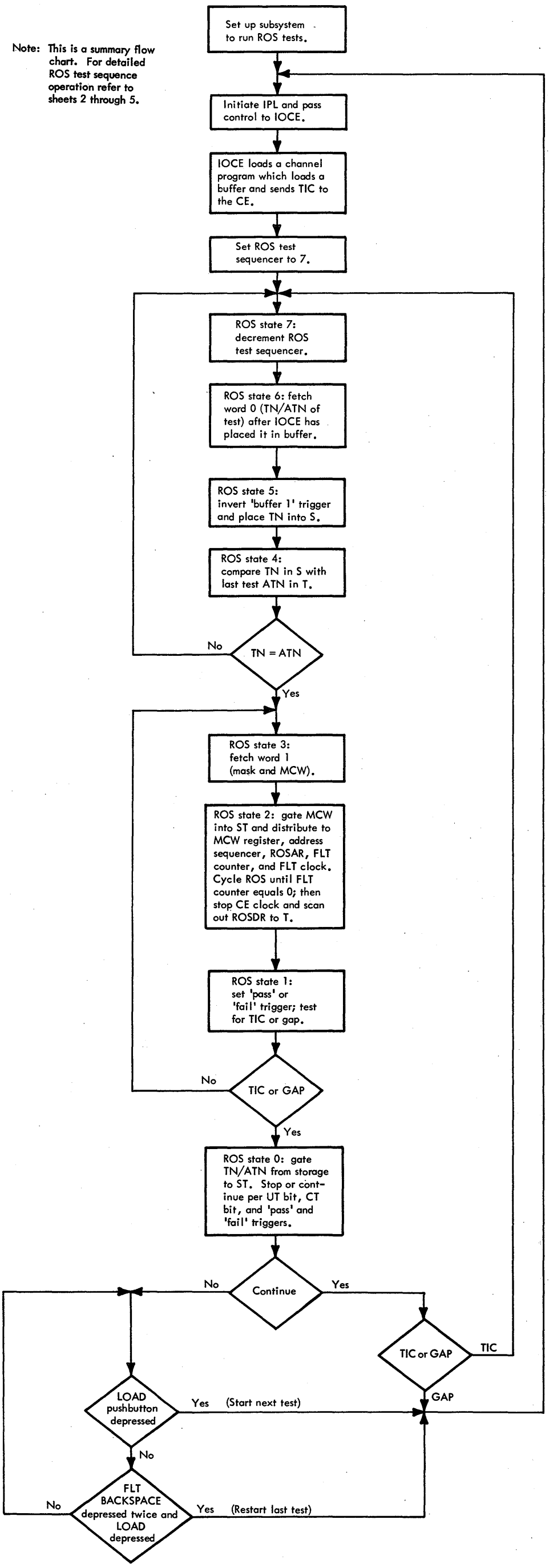


Diagram 6-115. ROS Test Sequence (Sheet 1 of 5)

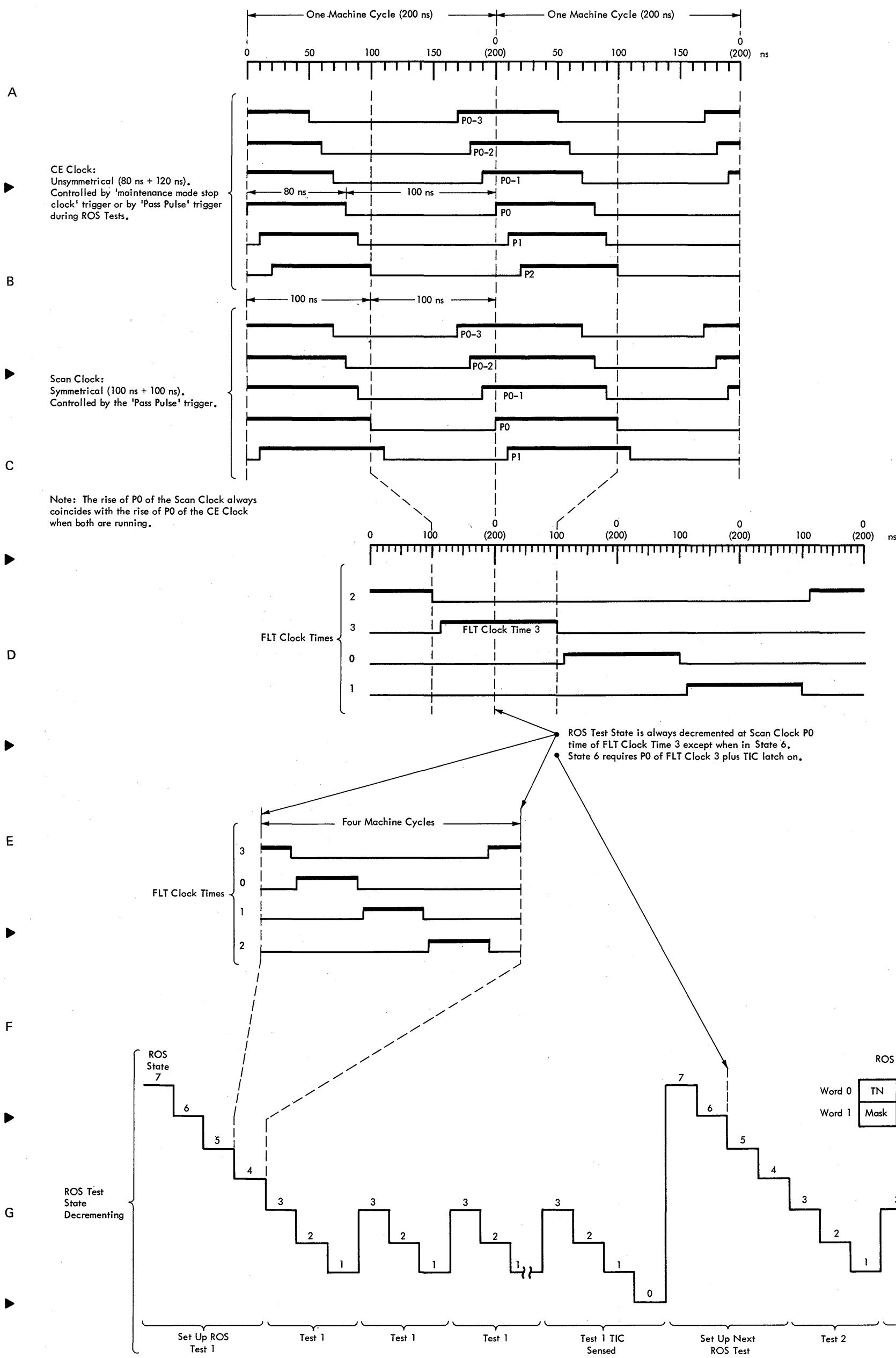


Diagram 6-115. ROS Test Sequence (Sheet 2 of 5)

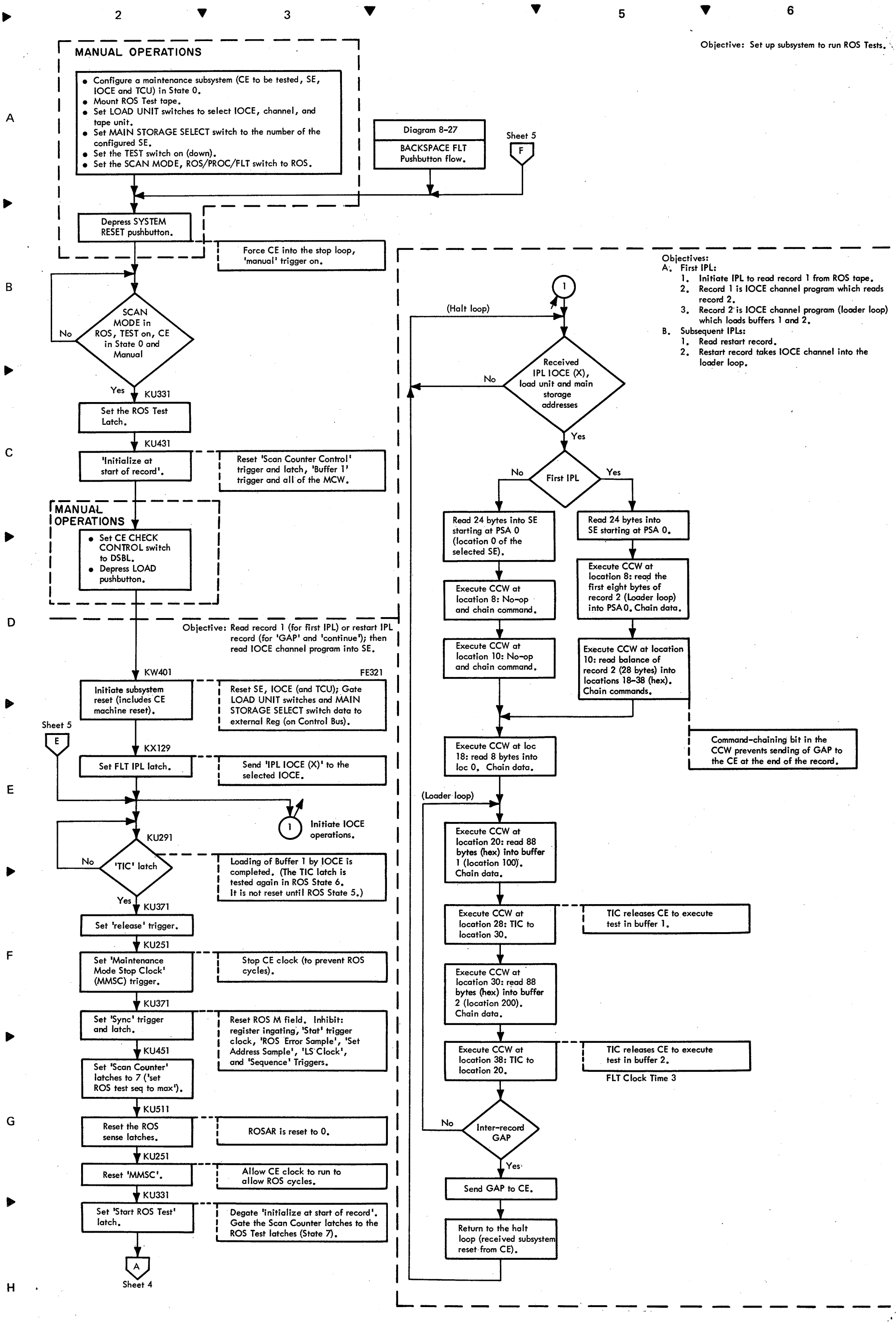


Diagram 6-115. ROS Test Sequence (Sheet 3 of 5)

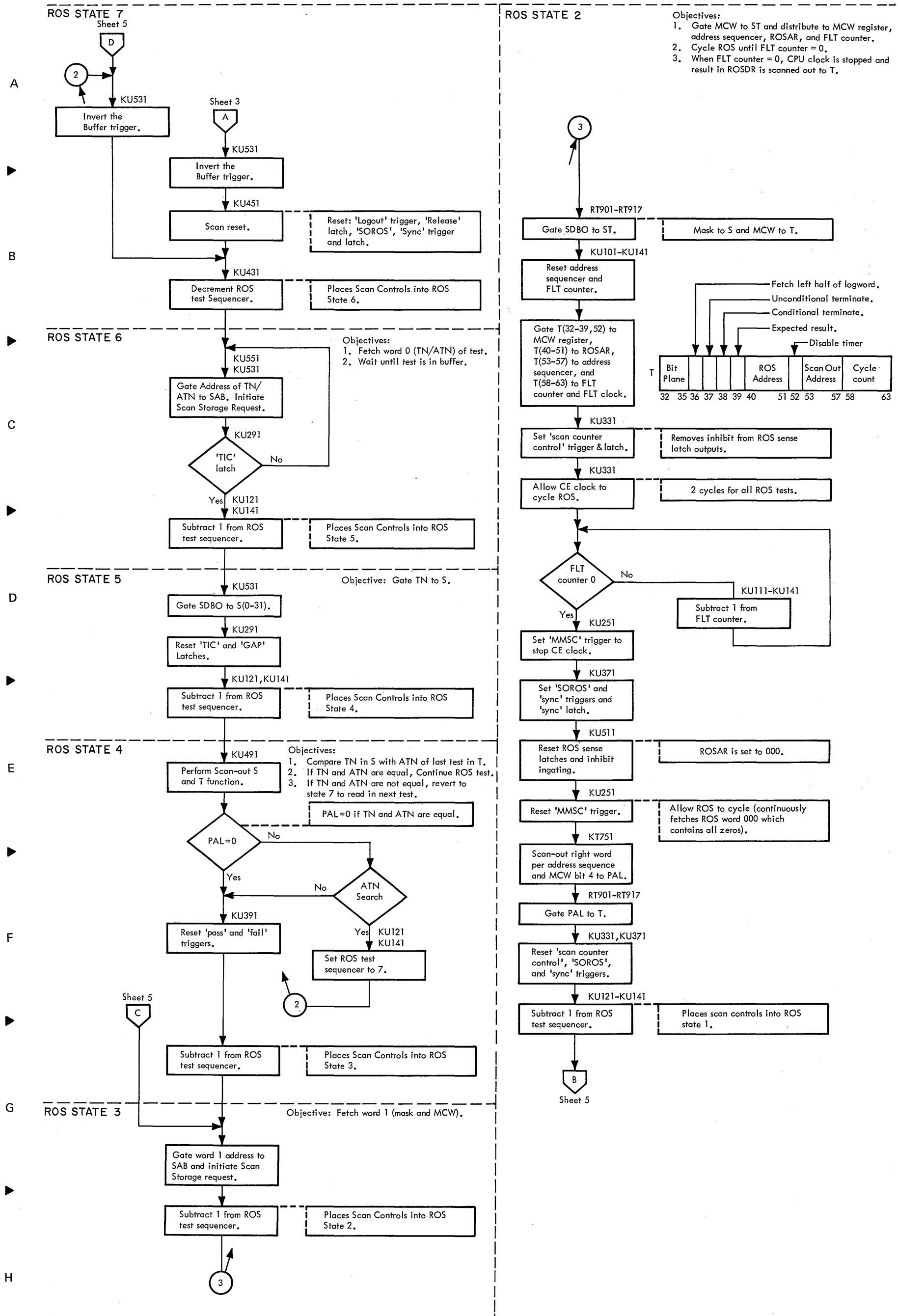


Diagram 6-115. ROS Test Sequence (Sheet 4 of 5)

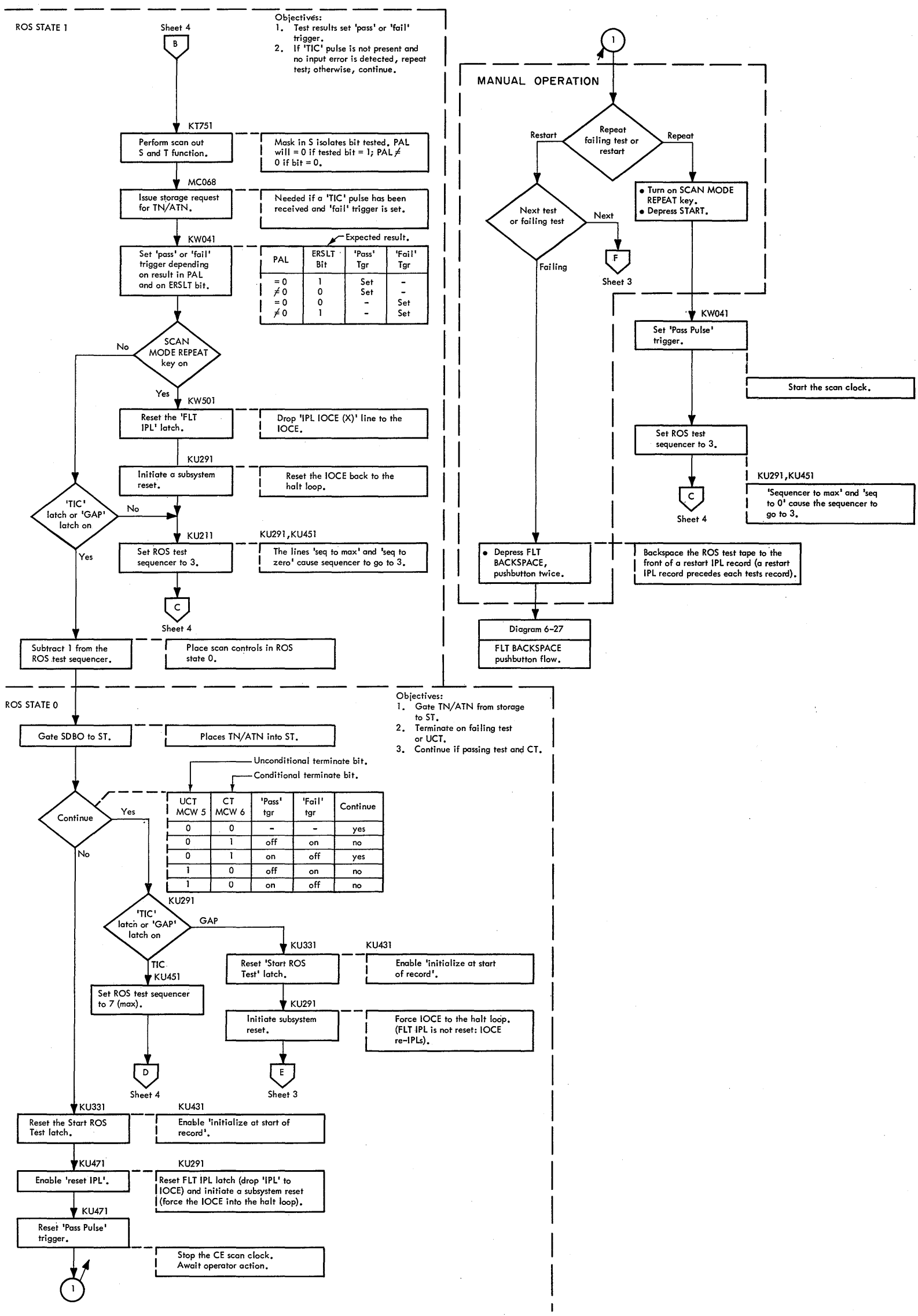
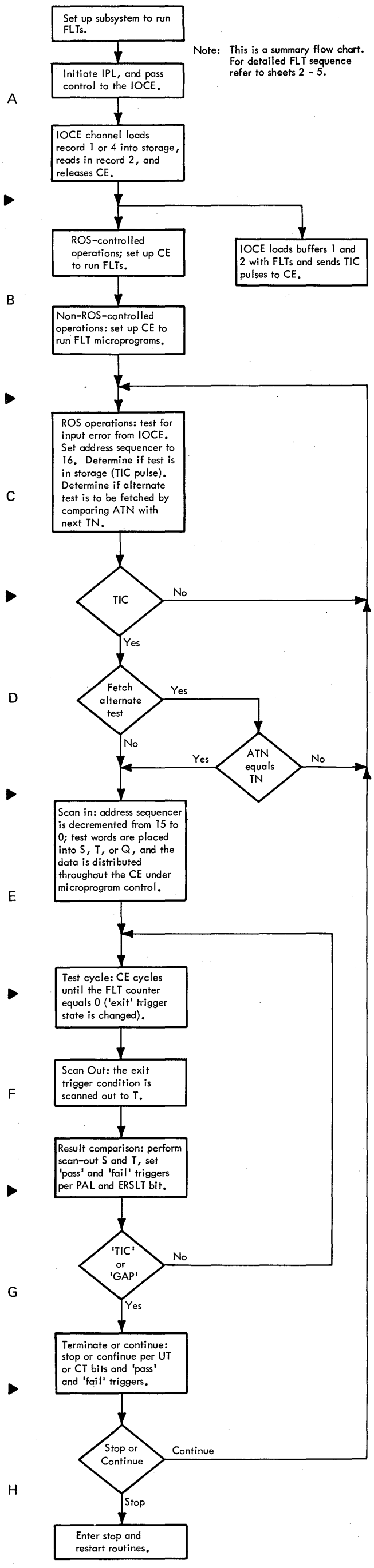


Diagram 6-115. ROS Test Sequence (Sheet 5 of 5)



Note: This is a summary flow chart. For detailed FLT sequence refer to sheets 2 - 5.

Diagram 6-116. FLT Sequence (Sheet 1 of 5)

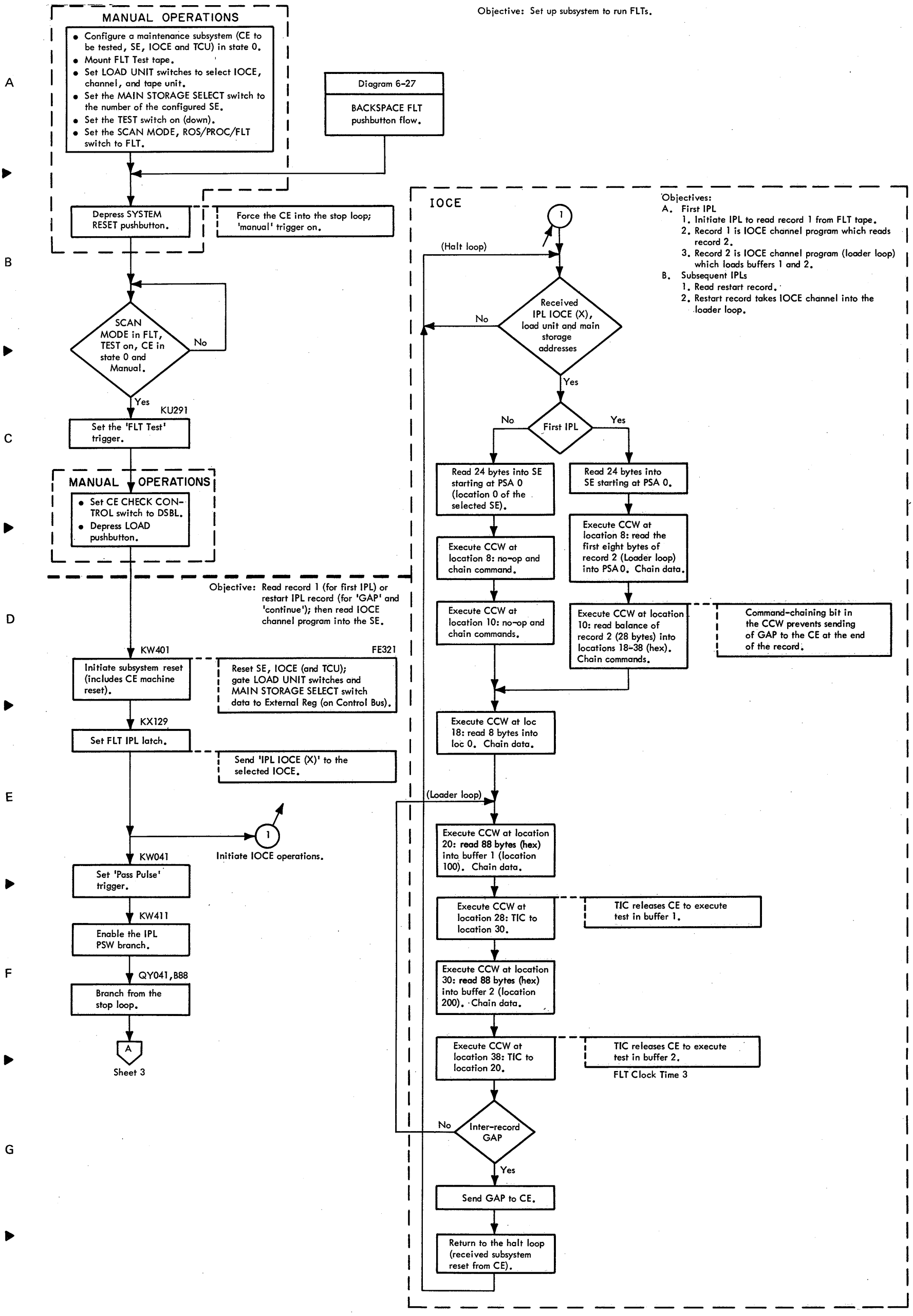


Diagram 6-116. FLT Sequence (Sheet 2 of 5)

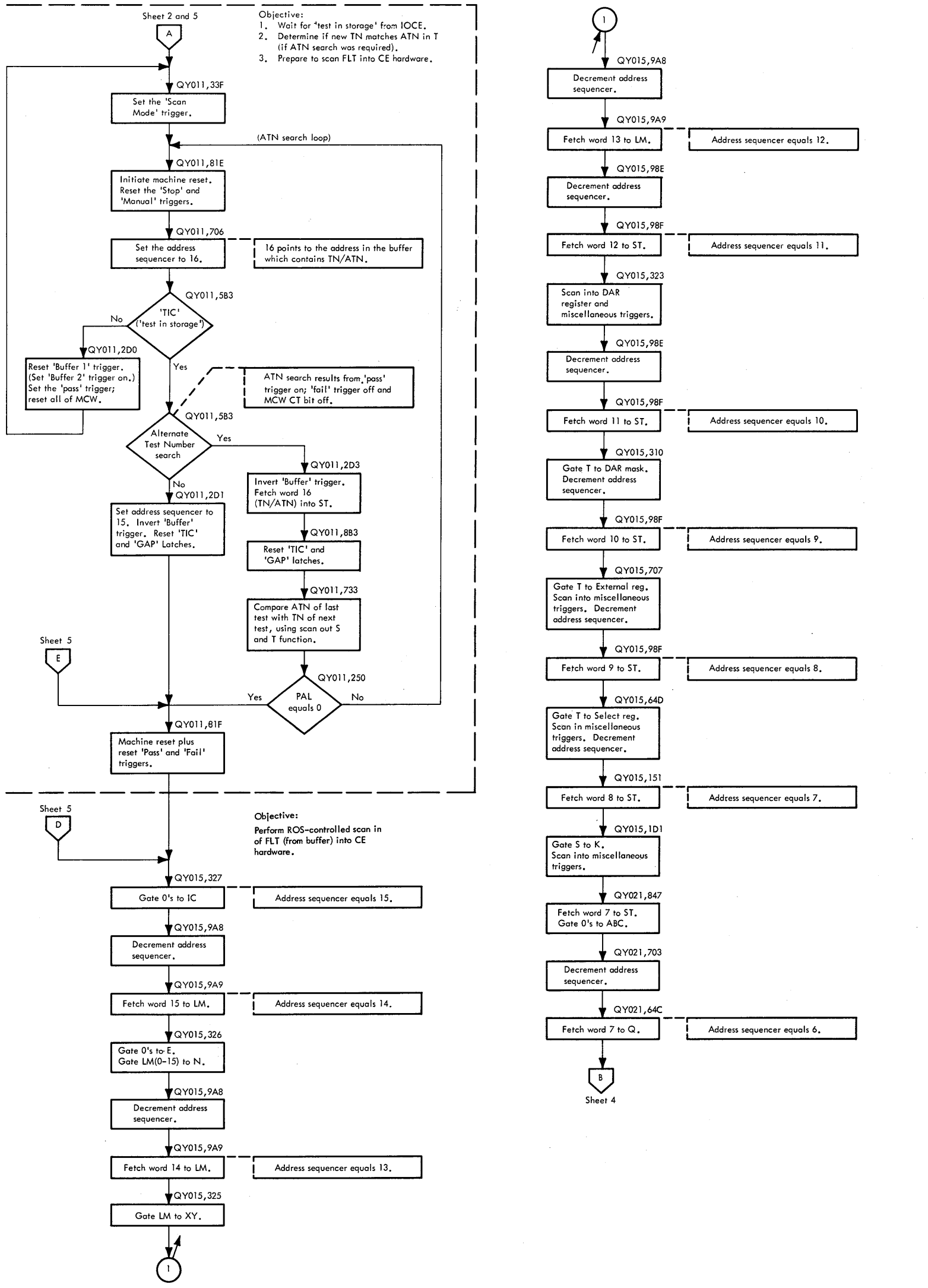


Diagram 6-116. FLT Sequence (Sheet 3 of 5)

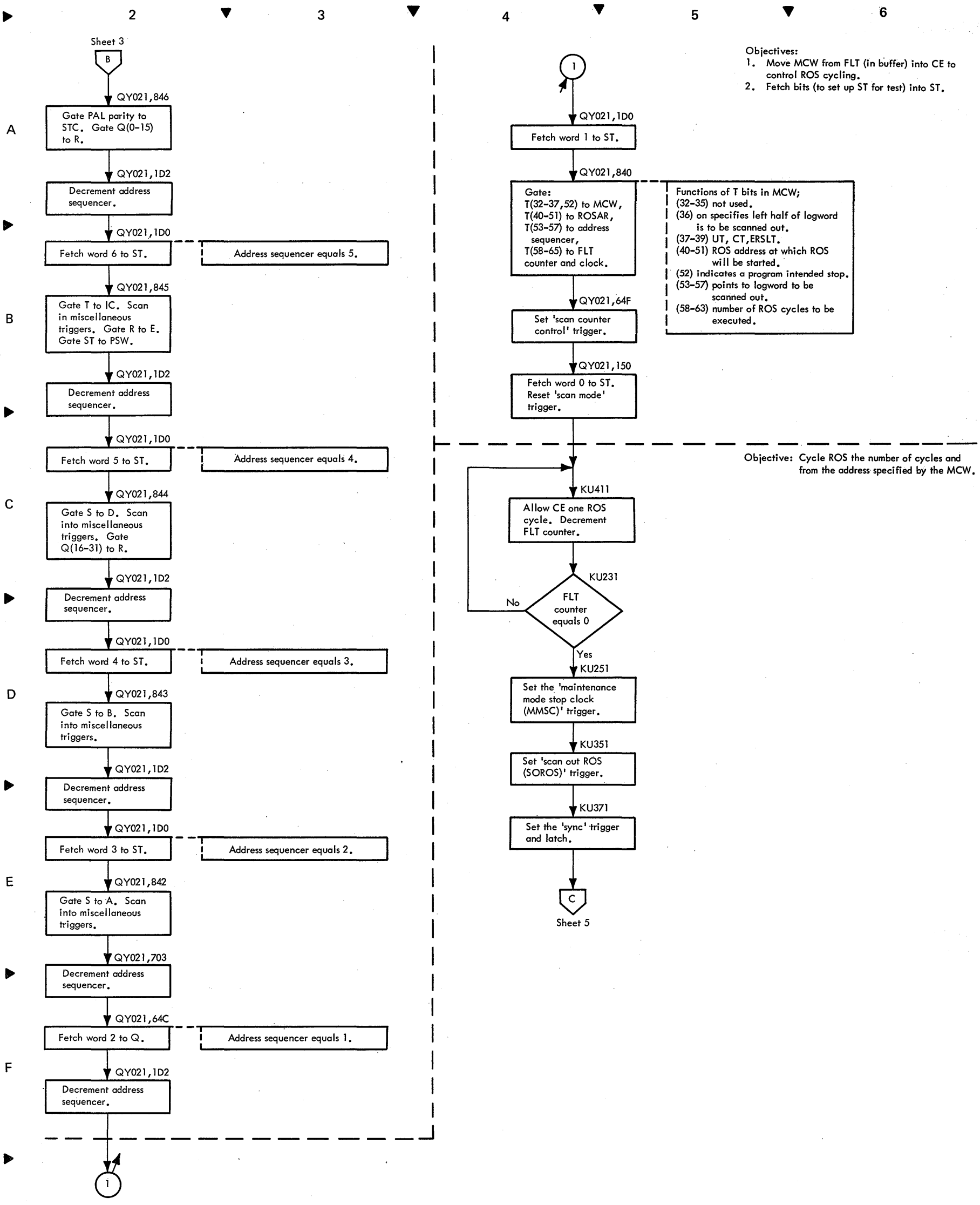
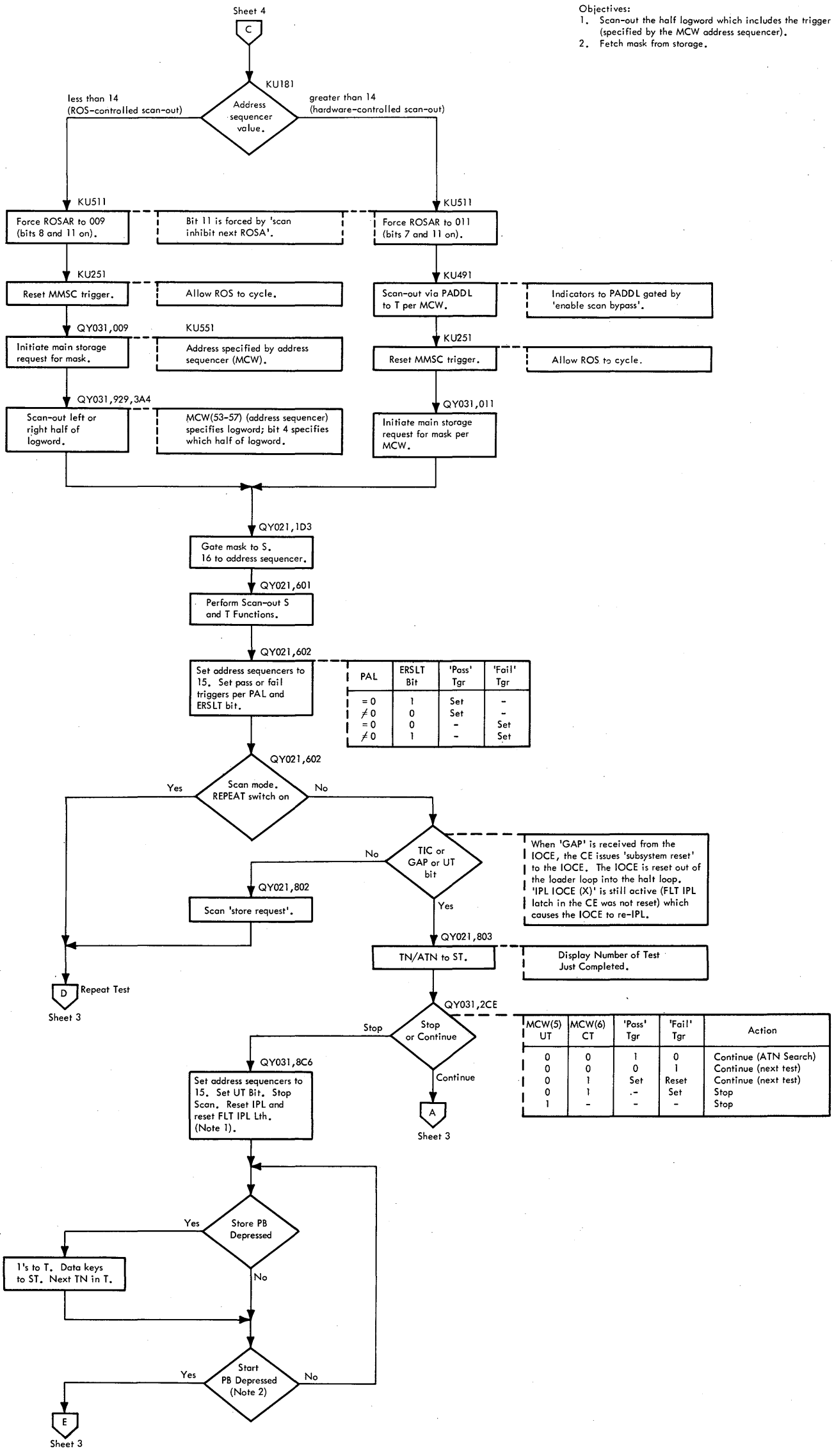


Diagram 6-116. FLT Sequence (Sheet 4 of 5)



Objectives:
1. Scan-out the half logword which includes the trigger being tested (specified by the MCW address sequencer).
2. Fetch mask from storage.

Notes:
1. To restart after a stop, depress FLT BACKSPACE twice, then the LOAD Pushbutton once. This will bring in the next test and run it (Diagram 6-27).
2. To loop on failing test, operator must place SCAN MODE REPEAT Switch down before depressing START pushbutton.

Diagram 6-116. FLT Sequence (Sheet 5 of 5)

LOG WORD NO.	PSA LOCATION	PSA WORD NO.	BIT POSITIONS IN MAIN STORAGE 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	COMMENTS
0	128 (80)	32	F REG P 0-7	D REG P 8-15	D REG P 16-23	D REG P 24-31					Q REGISTER P 0-7	Q REGISTER P 8-15	Q REGISTER P 16-23	Q REGISTER P 24-31	Q REGISTER P 32-39	Q REGISTER P 40-47	Q REGISTER P 48-55	Q REGISTER P 56-63	DAR MASK P 0-7	DAR MASK P 8-15	DAR MASK P 16-23	DAR MASK P 24-31													PARITY FOR F, D, Q AND DAR MASK
A	132	33	SELECT REG P 0-7	SELECT REG P 8-15	SELECT REG P 16-23	SELECT REG P 24-31					ST REGISTER P 0-7	ST REGISTER P 8-15	ST REGISTER P 16-23	ST REGISTER P 24-31	ST REGISTER P 32-39	ST REGISTER P 40-47	ST REGISTER P 48-55	ST REGISTER P 56-63	CCR P 0-7	CCR P 8-15	CCR P 16-23	CCR P 24-31													PARITY FOR SR, K, ST AND CCR
1	136 (88)	34	CE SEL SATR 1		G REG P 0-7	ATR P 32-39	ATR P 0-7	ATR P 8-15	ATR P 16-23	ATR P 24-31	IC REQ	RPT INSTR ADJ	INSTR STEP TGR	DISABLE INTER-LEAVE	STAT TGR A	I FETCH 1 TGR	IL NOT AVAIL	EXT INTRPT CE4 WDD 28	MCW RTP 0	ADDR SEQ 0	FLT CLOCK 0		MARK P 0-7	ST REG P 32-39	AB REG P 64-67	PADDL P 64-67									
▶	140	35	EXTERNAL TRIGGERS PERMIT IOCE 2	PERMIT IOCE 3	LOGOUT IOCE 1	LOGOUT IOCE 2	LOGOUT IOCE 3				AB REGISTER P 0-7	AB REGISTER P 8-15	AB REGISTER P 16-23	AB REGISTER P 24-31	AB REGISTER P 32-39	AB REGISTER P 40-47	AB REGISTER P 48-55	AB REGISTER P 56-63	R REG P 0-7	R REG P 8-15	E REG P 0-7	E REG P 8-15													
2	144 (90)	36	CE SEL SATR 2	CE SEL SATR 3	CE SEL SATR 4	TIME CLOCK STEP		SELECT IOCE 1	SELECT IOCE 2	SELECT IOCE 3	IOCE MC REQ	INTRPT GATE TGR			EXTERNAL REG P 0-7	EXTERNAL REG P 8-15	EXTERNAL REG P 16-23	EXTERNAL REG P 24-31	G REG (WRITE DIRECT) 0	1	2	3	4	5	6	7	ATR-2 32	33	34	35	36	37	38	39	G REG, ATR-2
B	148	37	PAM/RCU 1/1	2/2	3/	SE/DE 1	2	3	6	7	1/	2/	3/	4/	5/	6/1	7/2	8/3	9/4	10/5	18	19	CE 1	2	3	4	IOCE 24	25	26	27	28	1	2	3	SELECT REG
3	152 (98)	38	ADDRESS TRANSLATION REGISTER (ATR-1) 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	ATR-1
▶	156	39	STAT TRIGGERS B	C	D	E	F	G	H	BLOCK I FETCH	I FETCH 2 TGR	I FETCH 3 TGR	EXEC IN PROG TGR		PROG STORE COMP	I FETCH REQ TGR	BR INV ADDR TGR	I FETCH INV ADDR TGR	CE1 WDD 20	CE1 RDD 21	CE2 WDD 22	CE2 RDD 23	EXTERNAL INTERRUPT REGISTER TIMER 24	PB 25	CE3 WDD 26	CE3 RDD 27	CE4 RDD 29	PIR 30	DAR 31	PIR 1	2	3		TIMING GT TGR	
4	160 (A0)	40	PROGRAM STATUS WORD SYSTEM MASK 0	1	2	3	4	5	6	7	KEY 8	9	10	11	A 12	M 13	W 14	P 15	IC IN LSWR	COND CODE 34	35	PROGRAM STATUS WORD PROGRAM MASK 36	37	38	39	MC INTRPT			PROGRAM INTERRUPT CODE 8	4	2	1	INTERRUPT PRIORITY 1	2	PSW
C	164	41	D REQ	SCAN REQ	SET KEY	INSERT KEY	3 CYCLE REQ	PAGE 1 LTH			RPT INSTR INIT	PULSE MODE ADJ	PULSE MODE INIT	STOP TGR	BLOCK TGR	PASS PULSE TGR	BLOCK INTRPT	FORCE ADDR TGR	SINGLE CYCLE TGR	CHECK SUM	CLOCK INHIBIT	RPT TEST	FLT TEST	ROS TEST TGR	SCC TGR	SYNC TGR	DISABLE INTER-LEAVE REV STOR ADDR	FLT BKSP	CONSOLE LOG OUT	SOROS	DIAG TGR	FLT REL	TIC	GAP	
5	168 (A8)	42	STORAGE CHECK ADDRESS REGISTER 0	1	2	3	360 MODE	TEST MODE	SE STPD IN 360 MODE		SAB CHK	SDBI CHK	SE/DE T/O	SE/DE ADDR CHK	CHECK REGISTER 2 SE/DE DATA CHK	FETCH CHK	LOS TO SE/DE		IOCE BUS CHK	LOCAL STORE BUS CHK	CCR PTY CHK	ATR PTY CHK	PSBAR PTY CHK	PSBAR NOT CONF	PSBAR ALT CHK	SPLIT LOG OUT	LOG ROS CHK	LOG ADDR CHK	CE LOG REQ	RDD T/O					CR2
▶	172	43	MCW REGISTER LACMP 1	RSA FSP 2	RMP 3	RSARP 4	5	LOC 6	7	DT 20	SCAN ADDRESS SEQUENCER 1	2	3	4	SCAN CYCLE COUNTER 0	1	2	3	FLT CLOCK 1		ROS TEST SEQUENCER 0	1	2					FLT PASS	FLT FAIL		FLT STG ERR		MAINT MODE STOP CLOCK	SCAN BUFFER 1	
6	176 (B0)	44	EXTERNAL REGISTER 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	EXTERNAL REG
D	180	45	MARKS 0	1	2	3	4	5	6	7								TX TGR	AB REG 64	65	66	67		ABC 0	1	2	PADDL 64	65	66	67	INSERT LOCAL STORE SIGN	STC 0	1	2	
7	184 (B8)	46	AB REGISTER 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	AB REG (0-31)
▶	188	47	AB REGISTER 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	AB REG (32-63)

Diagram 6-117. CE Logword Formats (Sheet 1 of 3)

LOG WORD NO.	PSA LOC-ATION	PSA WORD NO.	BIT POSITIONS IN MAIN STORAGE																														COMMENTS
			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	
8	192 (C0)	48	PHYSICAL PSBAR 9	10	11	12	P 9-12	PSBAR COUNTER 9	10	11	P 9-15	LOGICAL PSBAR 9	10	11	12	13	14	15	16	17	18	19	ALT PSBAR				PROGRAM STATUS WORD D SYSTEM MASK 16	17	18	19	SE STPD 24	PSA LOCKOUT 1	2	3	PSBAR
	196	49	STATE S_0	S_1	SCON 1	2	3	4	ILOS 6		CONFIGURATION CONTROL REGISTER (CCR) 1/	2/	3/	4/	5/	SE/DE 6/1	7/2	8/3	9/4	10/5	18	19	CE 1	2	3	4	24	25	26	27	28	IOCE 1	2	3	CCR
	200 (C8)	50	DIAGNOSE ACCESSIBLE REGISTER MASK IOCE 1	SPARE	2	SPARE	3	SPARE	1/	2/	3/	4/	SE/DE 5/	6/1	7/2	8/3	9/4	10/5	SPARE 16	SPARE 17	PAM/RCU 1/1	2/2	3/	TCU 1	2	3	CE OWN OTC	OBS	SPARE 26	CE 1	2	3	4	SPARE 31	DAR MASK
	204	51	SERIAL ADDER LATCHES (SADDL) 0	1	2	3	4	5	6	7	INSTRUCTION COUNTER (IC) 8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	SADDL, IC
10	208 (D0)	52	R REGISTER 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	E REGISTER 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	R REG, E REG
	212	53	F REGISTER (READ DIRECT) 0	1	2	3	4	5	6	7	D REGISTER 8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	D REG, F REG
11	216 (D8)	54	Q REGISTER 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	Q REG
	220	55	Q REGISTER 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	Q REG
12	224 (E0)	56	DIAGNOSE ACCESSIBLE REGISTER (DAR) IOCE1 A	B	IOCE2 A	B	IOCE3 A	B	1/	2/	3/	4/	SE/DE ELC 5/	6/1	7/2	8/3	9/4	10/5	SPARE 16	SPARE 17	PAM/RCU ELC 1/1	2/2	3/	TCU ELC 1	2	3	CE OWN OTC	OBS	SPARE 26	CE ELC 1	2	3	4	SPARE 31	DAR
	228	57	CHECK REGISTER 1 PARALLEL ADDER FULL SUM-CHECK 4-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63									PARALLEL ADDER HALF-SUM CHECK 4-7	8-15	16-23	24-31	32-39	40-47	48-55	56-67			ROS PARITY CHECK 2-42	43-68	69-99	SADD HALF SUM CHK	SADD FULL SUM CHK	CR1
13	232 (E8)	58	LSWR P 0-7	P 8-15	P 16-23	P 24-31	E REG PTY 0-7	CHECK REG 1 PADD FULL SUM 64-67	MPR DECODE PTY CHK	PADD HALF SUM CHK	EXT TGRS PERMIT IOCE 1	IOCE INTRPT REQ	N REG P 0-7	P 8-15	PSW REG P 0-7	P 8-15	NO RETRY	SVC INTRPT	PHYS PSBAR PTY 9-12	PSBAR CNTR 12	LOGICAL PSBAR PTY 16-19		SADDL P 0-7	IC REG P 8-15	P 16-23	P 24-31								
	236	59	LOCAL STORAGE WORKING REGISTER (LSWR) 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	LSWR
14	240 (F0)	60																																	
	244	61	T32-63 TO PADDA 32-63	32-63 COMP	31-62 L1	31-62 C-L1	T 32-47 TO PADDA 48-63	T 48-63 TO PADDA 48-63	K 0-31 TO PADDA 32-63	8-31	D8-31 TO PADDA 7-30 L1	7-30 C-L1	40-63	40-63 COMP	FMTO	FMTM	FMTW	IC 8-31 TO PADDB 40-63	AB 4-7 TO PADDB 4-7	AB 8-31 TO PADDB 8-31	AB 32-63 TO PADDB 32-63	AB 64-67 TO PADDB 64-67	AB 6-31 TO PADDB 4-29L2	AB 32-67 TO PADDB 30-65L2	AB 64-67 TO PADDB 28-31	EX 6 TO PADDB 28-63	HOT ONE TO PADDB 60	E 8-11 TO PADDB 56-59	E 12-15 TO PADDB 60-63	E 8-11 TO PADDB 60-63	Q 4-15 TO PADDB 52-63	Q 20-31 TO PADDB 52-63	Q 36-47 TO PADDB 52-63	Q 52-63 TO PADDB 52-63	
15	248 (F8)	62																																	
	252	63	READ ONLY STORAGE DATA REGISTER (ROSDR) 70	71	72	73	74	75	76	77	79	80	81	82	83	84	P 43-68	86	88	89	90	P 69-99	92	93	94	95	97	98	99		RIGHT DIGIT TGR	S TGR	LEAVE TGR	STEP ABC TGR	ROSDR

Diagram 6-117. CE Logword Formats (Sheet 2 of 3)

LOG WORD NO.	PSA LOCATION	PSA WORD NO.	BIT POSITIONS IN MAIN STORAGE	COMMENTS
16	256 (100)	64		
	260	65	READ ONLY STORAGE DATA REGISTER (ROSDR): 37 38 39 40 41 42 43 44 46 47 48 49 50 51 52 53 55 56 57 58 59 60 61 62 64 65 66 67 68 INHIBIT LS WRITE PROSA A LAST FLT MODE	ROSDR
17	264 (108)	66		
	268	67	READ ONLY STORAGE DATA REGISTER (ROSDR): 2 3 4 5 6 7 8 10 11 12 13 14 15 16 17 19 P 2-42 21 22 23 24 25 26 28 29 30 31 32 33 34 35	ROSDR
18	272 (110)	68	L REG: P 0-7, P 8-15, P 16-23, P 24-31; K REG: P 0-7, P 8-15, P 16-23, P 24-31	
	276	69	READ ONLY STORAGE ADDRESS REGISTER (ROSAR): 1 2 3 4 5 6 7 8 10 11; PREVIOUS ROS ADDRESS (A): 0 1 2 3 4 5; PREVIOUS ROS ADDRESS (B): 7 8 9 10 11 0 1 2 4 5 6 7 8 9 10 11	ROSAR, PROSA, PROSB
19	280 (118)	70	M REG: P 32-39, P 40-47, P 48-55, P 56-63; X REG: P 0-7, P 8-15, P 16-23, P 24-31; Y REG: P 32-39, P 40-47, P 48-55, P 56-63; N REGISTER: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	N REG
	284	71	ROSAR: 0 9; PROSA: A6 B3; READ ONLY STORAGE DATA REGISTER (ROSDR): 0 9 18 27 36 45 54 63 69 78 87 96; S 0-31 TO PADDA 32-63; D 8-31 TO PADDA 8-31 C; AB 0-31 TO PADDB 32-63; F 4-7 TO PADDB 60-63; LOCAL STORAGE ADDRESS REG: 0 1 2 3 4; WR LOCAL STORE	ROSAR, LSAR
20	288 (120)	72	K REGISTER: 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	K REG
	292	73	MAINTENANCE CONTROL WORD (MCW): DE/SE 32 33; SELECT 34 35; DG SELECT 36 37 38 39; CVG SELECT 40 41 42 43; 44 45; RNOP 46; FDGR 47; DSE 1 48; RERP 49; RC 50; WRAP DE 51	MCW
21	296 (128)	74	LM REGISTER: 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	LM REG
	300	75	LM REGISTER: 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	LM REG
22	304 (130)	76	XY REGISTER: 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	XY REG
	308	77	XY REGISTER: 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	XY REG
23	312 (138)	78	ST REGISTER: 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	ST REG
	316	79	ST REGISTER: 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	ST REG

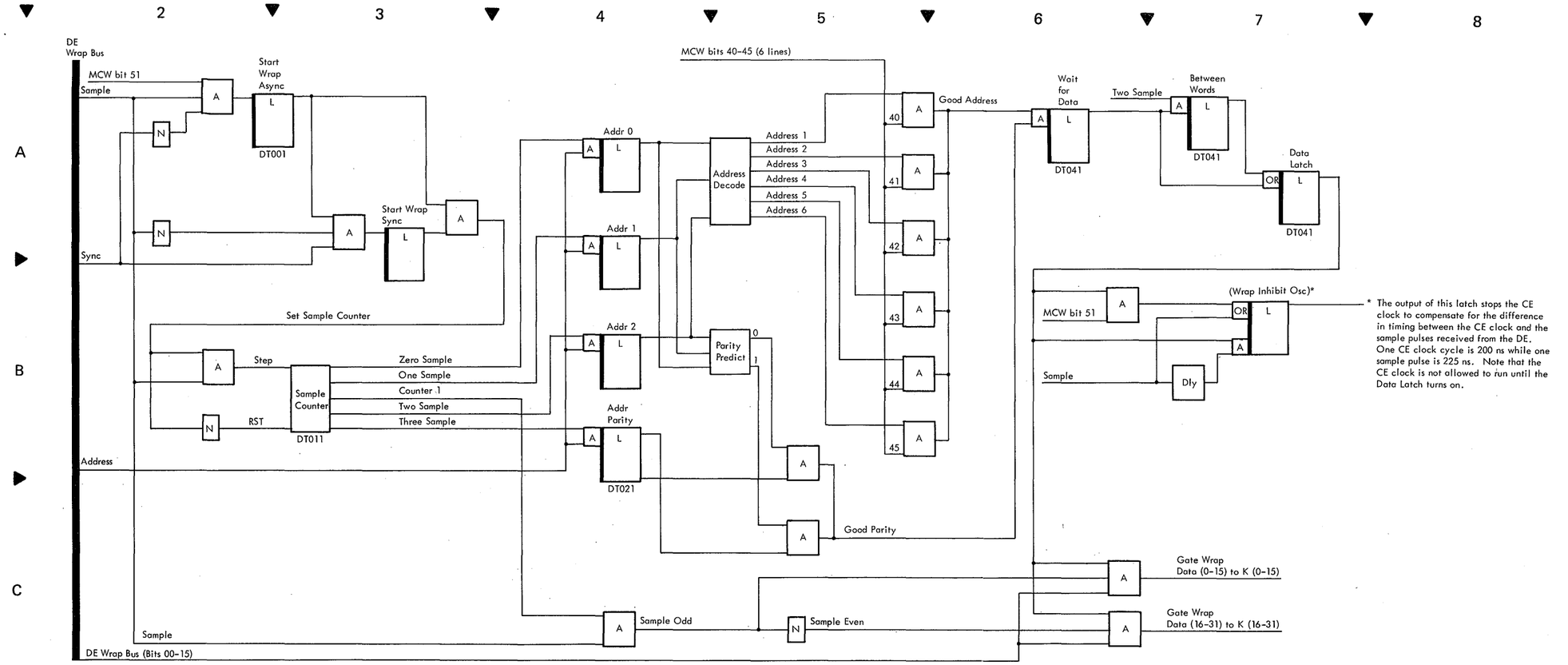
Diagram 6-117. CE Logword Formats (Sheet 3 of 3)

WORD NO.	BIT POSITIONS IN MAIN STORAGE 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	COMMENTS
0	SESDR (EVEN) (or IOCE SBO)																																SESDR (EVEN)
	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	
1	SESDR (EVEN)																																SESDR (EVEN)
	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	
2	MARKS (EVEN)									IOCE SAB (EVEN)																							MARKS, SESAR (EVEN)
	0-7	8-15	16-23	24-31	32-39	40-47	48-53	54-63		TAG (EVEN)				HI/LO			SESAR 1-14 (EVEN)												IOCE SAB				
										1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
3	CCR(0-7)								REQUEST (EVEN)								RESPONSE (EVEN)								CCR (8-15)								CCR, REQUEST AND RESPONSE LATCHES (EVEN)
	CE					IOCE			CE				IOCE				CE				IOCE				STATE		SCON						
	1	2	3	4		1	2	3	1	2	3	4	1	2	3		1	2	3	4	1	2	3		S_0	S_1	1	2	3	4	14	15	
4										SE ID							OPERATION		OPERATION (EVEN)				BLK SEL LATCH		SCON BACK UP LATCHES				MARK PARITY (EVEN)	ADDR PARITY			TYPE OF OP (EVEN)
										1	2	3	4				ISK	SSK	T & S	STORE	FETCH	SUPP LOG CHK	EVEN	ODD	1	2	3	4		1-5 0-7	6-12 8-15	13-19 16-23	
5	CHECK CONDITION (EVEN)								CHECK CONDITIONS (COMMON)								OBS	OTC THERMAL					CCR PARITY		SESDR PARITY (EVEN)								CHECKS (EVEN AND COMMON), SESDR (EVEN)
	MARK PARITY CHK	ADDR PARITY CHK	DATA PARITY CHK	SP PARITY CHK	KEY	NORMAL OP CHK			CCR	TAG PARITY CHK	TAG	SAB (IOCE)	SBO (IOCE)										0-7	8-15	SBO (IOCE) 0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	
6	SESDR (ODD)																																SESDR (ODD)
	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	
7	SESDR (ODD)																																SESDR (ODD)
	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	
8	MARKS (ODD)								IOCE SAB (ODD)																								MARKS, SESAR (ODD)
	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	TAG (ODD)				HI/LO		SESAR 1-14 (ODD)																			
									1	2	3	4	5	6	7		8	9	10	11	12	13	14	15	16	17	18	19					
9									REQUEST (ODD)								RESPONSE (ODD)																REQUEST AND RESPONSE LATCHES (ODD)
									CE				IOCE				CE				IOCE												
									1	2	3	4	1	2	3		1	2	3	4	1	2	3											
10	SPIKR								SPOKR										OPERATION (ODD)										MARK PARITY (ODD)	ADDR PARITY			SPIKR, SPOKR, TYPE OF OP (ODD)
	P 0-4	0	1	2	3	4			P 0-4	0	1	2	3	4					T & S	STORE	FETCH	SUPP LOG CHK								1-5 0-7	6-12 8-15	13-19 16-23	
11	CHECK CONDITION (ODD)						SPAR																		SESDR PARITY (ODD)								CHECKS (ODD), SPAR, SESDR (ODD)
	MARK PARITY CHK	ADDR PARITY CHK	DATA PARITY CHK	SP PARITY CHK	KEY	NORMAL OP CHK	P	X	Y	2	3	4	5	6	7										0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	

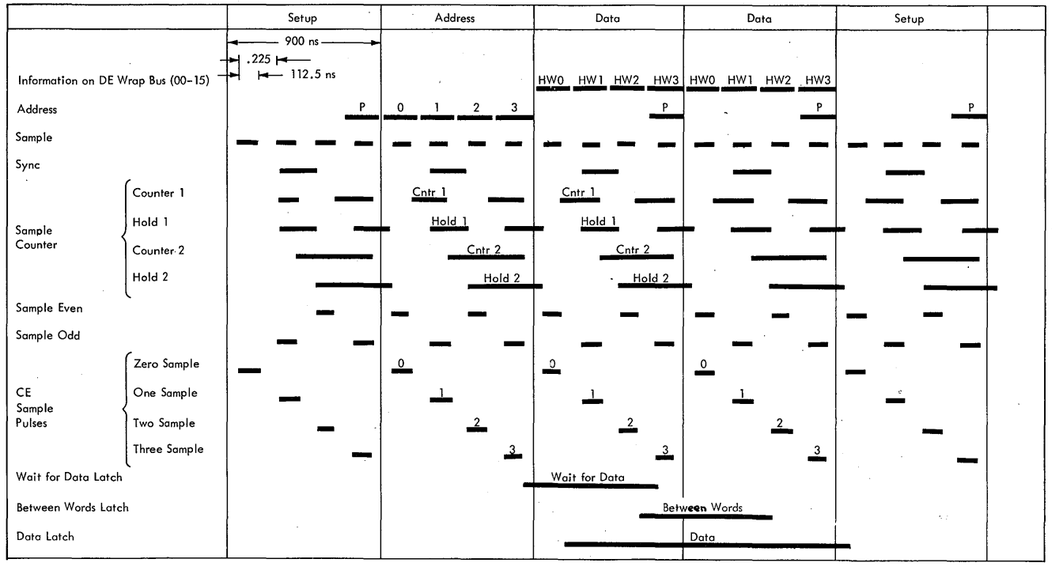
Diagram 6-118. SE Logword Formats

WORD NO.	BIT POSITIONS IN MAIN STORAGE																																COMMENTS
	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	
0	DESDR (EVEN)																																DESDR (EVEN)
	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	
1	DESDR (EVEN)																																DESDR (EVEN)
	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	
2	MARKS (EVEN)								DESDR PARITY (EVEN)								DESAR 1-14 (EVEN)																MARKS DESDR, DESAR (EVEN)
	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	P 6-12	6	7	8	9	10	11	12	P 13-19	13	14	15	16	17	18	19	
3											SPIKR						SPOKR						OPERATION		CHECK CONDITION (COMMON)								CHECKS (EVEN AND COMMON), SPIKE, SPOKE
	MARK PARITY	T & S	STORE	FETCH	DATA CHK	ADDR CHK	MARK PARITY CHK	CANCEL LATCH			P 0-4	0	1	2	3	4	P 0-4	0	1	2	3	4	ISK	SK	SP ADDR CHK	SP KEY CHK	KEY MIS-MATCH	OBS	OTC				
4	DESDR (ODD)																																DESDR (ODD)
	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	
5	DESDR (ODD)																																DESDR (EVEN)
	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	
6	MARKS (ODD)								DESDR PARITY (ODD)								DESAR 1-14 (ODD)																MARKS DESDR, DESAR (ODD)
	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63	P 6-12	6	7	8	9	10	11	12	P 13-19	13	14	15	16	17	18	19	
7																																	CHECKS (ODD)
	MARK PARITY	T & S	STORE	FETCH	DATA CHK	ADDR CHK	MARK PARITY CHK	CANCEL LATCH																									
8	CCR																																CCR
	DG																	STATE		SCON						CE							
	P 0-7	1A	1B	2B	2C	3C	3D	4D	4A	P 8-15	5A	5B	6B	6C	7C	7D	8D	8A	P 16-23	S_0	S_1	1	2	3	4	IDES		1	2	3	4	CCR INVALID	
9	RESPONSE																CVG																RESPONSE LATCHES, DG ADDR REG ID
	CE				A				B						C					D								ADDRESS REG					
	1	2	3	4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	A	B	C	D		
10	DG REGISTER CHECK								CHECK CONDITIONS (COMMON)										DG ADDRESS REGISTER														CHECKS DG ADDR REG
	1	2	3	4	5	6	7	8	TAG PARITY CHK	TAG	CCR PARITY CHK	MULTI ACCEPT	LS PARITY CHECK	NORMAL OP	REFRESH PARITY CHECK		P 1-7	1	2	3	4	5	6	7	P 8-14	8	9	10	11	12	13	14	
11	TAG						DE ID				HALFWORD COUNTER				DG DATA REGISTER																		DG DATA REG
	P 1-5	1	2	3	4	5	1	2	3	4	8	4	2	1	P 0-7	0	1	2	3	4	5	6	7	P 8-15	8	9	10	11	12	13	14	15	

Diagram 6-119. DE Logword Formats



* The output of this latch stops the CE clock to compensate for the difference in timing between the CE clock and the sample pulses received from the DE. One CE clock cycle is 200 ns while one sample pulse is 225 ns. Note that the CE clock is not allowed to run until the Data Latch turns on.



Note:
The chart at the left shows the timing for transferring one quadword of data to the CE via the DE Wrap Bus. The Diagnose instruction controls the total number transferred by setting the quadword count in the MCW.

A-Register (see AB-Register)
AB-Register:
  Data Flow 2-1
  Usage:
    Add, Subtract, and Compare Decimal 5-302 (Sheet 1)
    Branching Instructions 3-4
    Decimal Instructions 3-3
    Divide, Decimal 5-306 (Sheet 1)
    Divide, Fixed-Point 5-110 (Sheet 4)
    Divide, Floating-Point 5-213
    Fixed-Point Instructions 3-1
    Floating-Point Instructions 3-2
    I/O Instructions 3-6
    Logical Instructions 3-3
    Multiply, Decimal 5-305 (Sheet 1)
    Multiply, Fixed-Point 5-109 (Sheet 3)
    Multiply, Floating-Point 5-210
    RR I-Fetch, One-Cycle 5-7
    RX I-Fetch, One-Cycle 5-7
    RX I-Fetch, Two-Cycle Indexed 5-11
    Status Switching Instructions 3-5
AB-Register Byte Counter (see ABC)
ABC:
  Data Flow 2-1
  Incrementer Latches 4-207
  Positive Logic Diagram 4-207
  Triggers 4-207
  Usage:
    Decimal Instructions 3-3
    Divide, Decimal 5-306 (Sheet 1)
    Divide, Floating-Point 5-213
    Floating-Point Instructions 3-2
    Logical Instructions 3-3
    Multiply, Floating-Point 5-210
    Status Switching Instructions 3-5
Accept Trigger 4-604 (Sheet 1)
Add, A (5A); Fix Pt, RX 5-108
Add, AP (FA); Dec, SS:
  Complement Add Sequence 5-303
  GIS 5-301
  True Add Sequence 5-302
Add, AR (1A); Fix Pt, RR 5-108
Add Halfword, AH (4A); Fix Pt, RX 5-108
Add Logical, AL (5E); Fix Pt, RX 5-108
Add Logical, ALR (1E); Fix Pt, RR 5-108
Add Normalized, AD (6A); Fl Pt, RX (Long) 5-207
Add Normalized, ADR (2A); Fl Pt, RR (Long) 5-207
Add Normalized, AE (7A); Fl Pt, RX (Short) 5-206
Add Normalized, AER (3A); Fl Pt, RR (Short) 5-206
Add Unnormalized, AU (7E); Fl Pt, RX (Short) 5-206
Add Unnormalized, AUR (3E); Fl Pt, RR (Short) 5-206
Add Unnormalized, AW (6E); Fl Pt, RX (Long) 5-207
Add Unnormalized, AWR (2E); Fl Pt, RR (Long) 5-207
Address Compare Trigger 6-6
Address Decode Logic, SCI 4-602
Address Gating Logic, SCI 4-602
Address Gating, SCI 4-602
Address Generator, Storage; Scan 6-105
Addressing, ROS 4-105 (Sheet 1)
Adjust Parity Flip-Latch 4-205
And, N (54); Logic, RX 5-404
And, NC (D4); Logic, SS 5-404
And, NI (94); Logic, SI 5-404
And, NR (14); Logic, RR 5-404
AND Function, SAL(0) 4-406
Any Storage Error Trigger 4-606
Array Drivers, ROS 4-106
ATR:
  Data Flow 2-1
  Usage:
    Insert ATR 5-802
    Load PSBAR 5-805
    Set ATR (In issuing CE) 5-808
    Set ATR (In receiving CE) 5-809

B-Field Transfer to LAL 4-301
B-Register (see AB-Register)
Bit Position Logic, Bit 47; Parallel Adder 4-410
Block I-Fetch Trigger 5-16
Block Trigger 4-1, 6-12, 6-102
Branch and Link, BAL (45); Br, RX 5-503
Branch and Link, BALR (05); Br, RR 5-502
Branch Invalid Address Trigger 5-29 (Sheet 1)
Branch on Condition, BC (47); Br, RX 5-501
Branch on Condition, BCR (07); Br, RR 5-501
Branch on Count, BCT (46); Br, RX 5-504
Branch on Count, BCTR (06); Br, RR 5-504
Branch on Index High, BXH (86); Br, RS 5-505
Branch on Index Low or Equal, BXLE (87); Br, RS 5-505
Branch Requests 5-4
Branching Instructions:
  Branch and Link, BAL (45); RX 5-503
  Branch and Link, BALR (05); RR 5-502
  Branch on Condition, BC (47); RX 5-501
  Branch on Condition, BCR (07); RR 5-501
  Branch on Count, BCT (46); RX 5-504
  Branch on Count, BCTR (06); RR 5-504
  Branch on Index High, BXH (86); RS 5-505
  Branch on Index Low or Equal, BXLE (87); RS 5-505
  Data Flow 3-4
  Execute, EX (44); RX 5-506
Buffer 1 Trigger and Latch 6-105

Carry Lookahead Logic:
  Parallel Adder 4-411
  SAL(0-3) 4-402
CC Logic, Parallel Adder 4-416
CCR:
  Data Flow 2-1
  Usage:
    CE Control 4-210
    Psuedo-SCON (IPL or PSW Restart) 6-8A
    SCON 5-810
CE:
  CHECK Switch Logic 6-22
  Control Panel 6-1
  Data Flow 2-1
  Force Address 6-7
  IPL 6-8
  MACHINE RESET & FORCE ADDRESS 6-7
  PSW RESTART 6-8
  Request:
    Sensing Logic, SCI 4-601
    Timing, Typical 4-611
  Roller Sw Indicators 6-2
  TEST Switch 4-3
Characteristic Data Path:
  Floating-Point Divide 5-213
  Floating-Point Multiply 5-210
Check Registers:
  Inputs 6-22
  Roller Display 6-2
Check Reset 6-7
Clock:
  Control Logic 4-3
  Control Logic Functional Sequence, SCI 4-603
  FLT 6-103
  Scan 6-102
  Signal Generator 4-3
Compare C (59); Fix Pt, RX 5-108
Compare, CD (69); Fl Pt, RX (Long) 5-207
Compare, CDR (29); Fl Pt, RR (Long) 5-207
Compare, CE (79); Fl Pt, RX (Short) 5-206
Compare, CER (39); Fl Pt, RR (Short) 5-206
Compare, CL (55); Logic, RX 5-403
Compare, CLC (D5); Logic, SS 5-403
Compare, CLI (95); Logic, SI 5-403
Compare, CLR (15); Logic, RR 5-403
Compare CP (F9); Dec, SS:
  Complement Add Sequence 5-303
  GIS 5-301
  True Add Sequence 5-302
Compare, CR (19); Fix Pt, RR 5-108
Compare, Halfword, CH (49); Fix Pt, RX 5-108
Console Logout Latch 6-25
Console Signal Trigger 5-24
Convert to Binary, CVB (4F); Fix Pt, RX 5-111
Convert to Decimal, CVD (4E); Fix Pt, RX 5-112
CPU Clock Go Trigger 6-102, 4-603
CPU 5 Trigger and Latch 4-603
CPU 4 Trigger and Latch 4-603
CPU Request Latch 6-108
CPU Sequencers 4-603
CPU Store in Progress Exceptional Condition 5-18
CPU Store Trigger 5-18
CPU 3 Trigger and Latch 4-603
CPU 2 Trigger and Latch 4-603

D-Register:
  Address Gating, BCU 4-602
  Data Flow 2-1
  Usage:
    Add, Subtract, and Compare, Decimal 5-302 (Sheet 1)
    B-Field Transfer to LAL 4-201
    Branching Instructions 3-4
    Decimal Instructions 3-3
    Divide, Decimal 5-306 (Sheet 1)
    Divide, Floating-Point 5-213
    Fixed-Point Instructions 3-1
    Floating-Point Instructions 3-2
    Instruction Requests During End Op 5-2
    I/O Instructions 3-6
    Logical Instructions 3-3
    Multiply, Decimal 5-305 (Sheet 1)
    Multiply, Floating-Point 5-210
    Operand Prefetching During End Op 5-1
    RR I-Fetch, One-Cycle 5-7
    RS I-Fetch, One-Cycle 5-10
    RX I-Fetch, One-Cycle 5-10
    RX I-Fetch, Two-Cycle Indexed 5-11
    Selection of I-Fetch Sequence 5-5
    SI I-Fetch, One-Cycle 5-10
    Status Switching Instructions 3-5
D Request Trigger 4-601
D Sync Trigger and Latch 4-601

DAR:
Data Flow 2-1
Usage (Diagnose) 5-609
DAR MASK:
Data Flow 2-1
Usage (Diagnose) 5-609
Data Flow:
Branching Instructions 3-4
CE 2-1
Decimal Instructions 3-3
Direct Control Operation 5-607 (Sheet 2)
Divide, Fixed-Point 5-110 (Sheet 4)
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
I/O Instructions 3-6
Logical Instructions 3-3
Multiply, Fixed-Point 5-109 (Sheet 3)
Multiply, Floating-Point 5-210
ROS 4-105 (Sheet 2)
Scan 6-101
Status Switching Instructions 3-5
DE Wrap:
Bus Controls 6-201
Data Flow 2-1
Decimal Add 6 Logic, Serial Adder 4-403
Decimal Correction Logic, SAL(0–3) 4-404
Decimal Instructions:
Add, AP (FA); SS:
Complement Add Sequence 5-303
GIS 5-301
True Add Sequence 5-302
Compare, CP (F9); SS:
Complement Add Sequence 5-303
GIS 5-301
True Add Sequence 5-302
Data Flow 3-3
Divide, DP (FD); SS 5-306
Move With Offset, MVO (F1); SS:
GIS 5-307
Not Word Overlap Sequence 5-312
Word Overlap Sequence 5-313
Multiply, MP (FC); SS 5-305
Pack, PACK (F2); SS:
GIS 5-307
Not Word Overlap Sequence 5-308
Word Overlap Sequence 5-309
Subtract, SP (FB); SS:
Complement Add Sequence 5-303
GIS 5-301
True Add Sequence 5-302
Unpack, UNPK (F3); SS:
GIS 5-307
Not Word Overlap Sequence 5-310
Word Overlap Sequence 5-311
Zero and Add, ZAP (F8); SS 5-304
DEFEAT INTERLEAVING Switch Gating 6-11
Delayed Block I-Fetch Trigger 5-29 (Sheet 1)
Delayed I-Fetch Protection Gate Trigger 5-29 (Sheet 1)
Delayed I-Fetch Storage Request Trigger 5-29 (Sheet 1)
Diagnose (83); Stat Sw, SI 5-609
Direct Control Operation:
Data Flow 5-607
Direct Control Interface 3-5
Read Direct, RDD (85); Stat Sw, SI 5-608
Write Direct, WRD (84); Stat Sw, SI 5-607
Disable Interleaving and Reverse Storage Address Trigger 6-11
Disable Interleaving Trigger 6-11
Disable Interval Timer Logic 6-21
Display Instructions:
Convert and Start Symbols, CSS (02); DSPY, RR 5-903
Convert Weather Lines, CVWL (03); DSPY, RR 5-905
Load Chain, LC (51); DSPY, RX 5-906
Repack Symbols, RPSB (0F); DSPY, RR 5-902
Divide, D (5D); Fix Pt, RX 5-110 (Sheet 2)
Divide, DD (6D); Fl Pt, RX (Long) 5-215
Divide, DDR (2D); Fl Pt, RR (Long) 5-215
Divide, DE (7D); Fl Pt, RR (Long) 5-215
Divide, DER (3D); Fl Pt, RR (Short) 5-214
Divide, DP (FD); Dec, SS 5-306
Divide, DR (1D); Fix Pt, RX 5-110 (Sheet 1)
Divide, Fixed-Point:
Algorithm 5-110 (Sheet 3)
Data Flow 5-110 (Sheet 4)
Initialization 5-110 (Sheets 1 and 2)
Termination 5-110 (Sheets 5 and 6)
Divide, Floating-Point; Data Flow 5-213
Dividend Bits, Transfer of Low-Order; Fixed-Point 5-110 (Sheet 4)
Divisor Multiple, Derivation of; Fixed-Point 5-110 (Sheet 4)
Drive Line Decode, ROS 4-105 (Sheet 1)

E-Register:
Data Flow 2-1
Incrementer, Bits 14 and 15 4-203
Parity Prediction After Incrementing 4-204
Usage:
Branch Requests 5-4
Branching Instructions 3-4
Decimal Instructions 3-3
Direct Control Operation 5-607
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
I/O Instructions 3-7
Logical Instructions 3-3
Multiply, Floating-Point 5-210
RR I-Fetch, One-Cycle 5-7
RR I-Fetch, Two-Cycle 5-8
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle 5-12
RX I-Fetch, Two-Cycle Indexed 5-11
RX I-Fetch, Two-Cycle Nonindexed 5-12
SI I-Fetch, One-Cycle 5-10
SI I-Fetch, Two-Cycle 5-12
Status Switching Instructions 3-5
E-Register Incrementer:
Bits 14 and 15, Positive Logic Diagram 4-203
Parity Prediction After Incrementing 4-204
Early End Op, Instruction Requests 5-3
Edit, ED (DE); Logic, SS 5-411
Edit and Mark, EDMK (DF); Logic, SS 5-411
EEOP Request Trigger 5-30
E(8–15) Parity Trigger 4-204
End Op:
Instruction Requests 5-2
Operand Prefetching 5-1
Error Controls 6-22
Error Trigger 4-606, 4-413, 4-414, 6-22, 6-108
Exceptional Conditions:
CPU Store In Progress 5-18
Invalid Instruction Address Test 5-29
Manual Control 5-27
Program Store Compare 5-28
Q-Register Refill 5-30
Timer 5-17
Excess-6 Logic, Parallel Adder 4-415
Exclusive-OR Function, SAL(0) 4-406
Exclusive-OR, X (57); Logic, RX 5-406
Exclusive-OR, XC (D7); Logic, SS 5-406
Exclusive-OR, XI (97); Logic, SI 5-406
Exclusive-OR, XR (17); Logic, RR 5-406
Execute, EX(44); Br, RX 5-506
External Interruptions· 5-24
External Register:
Data Flow 2-1
Usage:
SATR 5-808, 809
SCON 5-810
SIOP 5-807
External Signal Flip-Latches 5-607
External Signal Trigger 5-24

F-Register:
Data Flow 2-1
Usage:
Add, Subtract, and Compare, Decimal 5-302 (Sheet 1)
Direct Control Operation 5-607 (Sheet 2)
Divide, Decimal 5-306 (Sheet 1)
Divide, Fixed-Point 5-110 (Sheet 4)
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
Multiply, Decimal 5-305 (Sheet 1)
Multiply, Fixed-Point 5-109 (Sheet 3)
Multiply, Floating-Point 5-210
Status Switching Instructions 3-5
Fail Trigger 6-111, 6-112
Fault Locating Test (see FLT)
Final Error Latch 4-413, 4-414
Fixed-Point Instructions:
Add, A (5A); RX 5-108
Add, AR (1A); RR 5-108
Add Halfword, AH (4A); RX 5-108
Add Logical, AL (5E); RX 5-108
Add Logical, ALR (1E); RR 5-108
Compare, C (59); RX 5-108
Compare, CR (19); RR 5-108
Compare Halfword, CH (49); RX 5-108
Convert to Binary, CVB (4F); RX 5-111
Convert to Decimal, CVD (4E); RX 5-112
Data Flow 3-1
Divide:
Algorithm 5-110 (Sheet 3)
Data Flow 5-110 (Sheet 4)
Initialization 5-110 (Sheets 1 and 2)
Termination 5-110 (Sheet 2)
Divide, D (5D); RX 5-110 (Sheet 2)
Divide, DR (1D); RR 5-110 (Sheet 1)
Load, L (58); RX 5-101
Load, LR (18); RR 5-101
Load and Test, LTR (12); RR 5-103
Load Complement, LCR (13); RR 5-104
Load Halfword, LH (48); RX 5-102
Load Multiple, LM (98); RS 5-107
Load Negative, LNR (11); RR 5-106
Load Positive, LPR (10); RR 5-105
Multiply:
Algorithm 5-109 (Sheet 2)
Data Flow 5-109 (Sheet 3)
Initialization 5-109 (Sheet 1)
Multiply, M (5C); RX 5-109
Multiply, MR (1C); RR 5-109
Multiply Halfword, MH (4C); RX 5-109
Shift Left Double, SLDA (8F); RS 5-117
Shift Left Single, SLA (8B); RS 5-116
Shift Right Double, SRDA (8E); RS 5-119

Shift Right Single, SRA (8A); RS 5-118
Store Halfword, STH (40); RX 5-114
Store Multiple, STM (90); RS 5-115
Store, ST (50); RX 5-113
Subtract Halfword, SH (4B); RX 5-108
Subtract Logical, SL (5F); RX 5-108
Subtract Logical, SLR (1F); RR 5-108
Subtract, S (5B); RX 5-108
Subtract, SR (1B); RR 5-108
Floating-Point Instructions:
Add Normalized, AD (6A); RX (Long) 5-207
Add Normalized, ADR (2A); RR (Long) 5-207
Add Normalized, AE (7A); RX (Short) 5-206
Add Normalized, AER (3A); RR (Short) 5-206
Add Unnormalized, AU (7E); RX (Short) 5-206
Add Unnormalized, AUR (3E); RR (Short) 5-206
Add Unnormalized, AW (6E); RX (Long) 5-207
Add Unnormalized, AWR (2E); RR (Long) 5-207
CC Setting 5-201
Compare, CD (69); RX (Long) 5-207
Compare, CDR (29); RR (Long) 5-207
Compare, CE (79); RX (Short) 5-206
Compare, CER (39); RR (Short) 5-206
Data Flow 2-1
Divide, Data Flow 5-213
Divide, DD (6D); RX (Long) 5-215
Divide, DDR (2D); RR (Long) 5-215
Divide, DE (7D); RX (Short) 5-214
Divide, DER (3D); RR (Short) 5-214
Halve, HDR (24); RR (Long) 5-209
Halve, HER (34); RR (Short) 5-208
Load and Test, LTDR (22); RR (Long) 5-205
Load and Test, LTER (32); RR (Short) 5-204
Load Complement, LCDR (23); RR (Long) 5-205
Load Complement, LCER (33); RR (Short) 5-204
Load, LD (68); RX (Long) 5-203
Load, LDR (28); RR (Long) 5-202
Load, LE (78); RX (Short) 5-203
Load, LER (38); RR (Short) 5-202
Load Negative, LNDR (21); RR (Long) 5-205
Load Negative, LNER (31); RR (Short) 5-204
Load Positive, LPDR (20); RR (Long) 5-205
Load Positive, LPER (30); RR (Short) 5-204
Multiply, Data Flow 5-210
Multiply, MD (6C); RX (Long) 5-212
Multiply, MDR (2C); RR (Long) 5-212
Multiply, ME (7C); RX (Short) 5-211
Multiply, MER (3C); RR (Short) 5-211
Save Signs and Insert Sign Function 5-201
Store, STD (60); RX (Long) 5-216
Store, STE (70); RX (Short) 5-216
Subtract Normalized, SD (6B); RX (Long) 5-207
Subtract Normalized, SE (7B); RX (Short) 5-206
Subtract Normalized, SER (3B); RR (Short) 5-206
Subtract Unnormalized, SU (7F); RX (Short) 5-206
Subtract Unnormalized, SUR (3F); RR (Short) 5-206
Subtract Unnormalized, SW (6F); RX (Long) 5-207
Subtract Unnormalized, SWR (2F); RR (Long) 5-207
FLT:
Backspace Pb 6-27
Clock 6-103
Counter, Decrementing 6-106
Sequence 6-116
FLT Counter = 0 Latch 6-106
FLT Test Trigger 6-26, 6-111
Force Address Latch 4-1, 6-7
Fraction Data Path:
Floating-Point Divide 5-213
Floating-Point Multiply 5-210
FREQUENCY ALTERATION Switch, Clock Signal Generator 4-3
Full-Sum Checking Logic, PA(48–55) 4-514
Full Sum Error (48–55) Trigger 4-514
Functional Units (see Specific Unit)

G-Register:
Data Flow 2-1
Usage:
Direct Control Operation 5-607
Status Switching Instructions 3-5
GAP Latch 6-113
Gate A(8–15) to SBA Trigger 4-401
Gate A(0–7) to SBA Trigger 4-401
Gate B(56–63) to SBA Trigger 4-401
Gate Control Triggers for 'B + T' Micro-order 4-409
Gate F to SBA Trigger 4-401
Gate I-Fetch Invalid Address Trigger 5-29 (Sheet 1)
Gate IL Not Available Trigger 5-16
General Initialization Sequence (see GIS)
GIS:
Add, AP (FA); Dec, SS 5-301
Compare, CP (F9); Dec, SS 5-301
Logical Instructions 5-401
Move With Offset, MVO (F1); Dec, SS 5-307
Pack, PACK (F2); Dec, SS 5-307
Subtract, SP (FB); Dec, SS 5-301
Unpack, UNPK (F3); Dec, SS 5-307

Half-Sum Checking Logic, PA(48–55) 4-413
Half-Sum Error Trigger 4-413
Half-Sum Precheck Error (48–55) Trigger 4-413
Halt I/O, HIO (9E); I/O, SI 5-703
Halve, HDR (24); Fl Pt, RR (Long) 5-209
Halve, HER (34); Fl Pt, RR (Short) 5-208
Hold In Flip-Latch 5-607 (Sheet 2)
Hold MC IPL Latch 6-9

I-Fetch:
Block I-Fetch Trigger 5-16
Branch Requests 5-4
Common Interruption Routine 5-26
CPU Store in Progress Exceptional Condition 5-18
External Interruptions 5-24
Instruction Requests During Early End Op 5-3
Instruction Requests During End Op 5-2
Interrupt Code Triggers:
Nonbranch Setting of 5-20
Program Interruption Code 5-22
Spec Y-Branch Setting of 5-21
Invalid Instruction Address Test Exceptional Condition 5-29
I/O Interruption 5-25
Machine Check Interruption 5-19
Manual Control Exceptional Conditions 5-27
Operand Prefetching During End Op 5-1
Program Interruptions 5-22
Program Store Compare Exceptional Condition 5-28
Q-Register Refill Exceptional Condition 5-30
RR, Flowchart 5-6
RR, One-Cycle 5-7
RR, Two-Cycle 5-8
RS, Flowchart 5-13
RS, One-Cycle 5-10
RS, Two-Cycle 5-12
RX, Flowchart 5-9
RX, One-Cycle 5-10
RX, Two-Cycle Indexed 5-11
RX, Two-Cycle Nonindexed 5-12
Sequence, Selection of 5-5
Sequencers 5-15
SI, Flowchart 5-13
SI, One-Cycle 5-10
SI, Two-Cycle 5-12
SS, Flowchart 5-14
Supervisor Call Interruption 5-23
Timer Exceptional Condition 5-17
I-Fetch Invalid Address Trigger 5-29 (Sheet 1)
I-Fetch Request Trigger 5-29 (Sheet 1)
IC:
Address Gating, BCU 4-602
Data Flow 2-1
Parity Adjustment 4-205
Usage:
Add, Subtract, and Compare, Decimal 5-302 (Sheet 1)
B-Field Transfer to LAL 4-201
Branching Instructions 3-4
Decimal Instructions 3-3
Divide, Decimal 5-306 (Sheet 1)
Fixed-Point Instructions 3-1
I-Fetch Sequencer Control 5-15
Instruction Requests During Early End Op 5-3
Instruction Requests During End Op 5-2
Logical Instructions 3-3
Multiply, Decimal 5-305 (Sheet 1)
Operand Prefetching During End Op 5-1
RR I-Fetch, One-Cycle 5-7
RS I-Fetch, One-Cycle 5-10
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
Selection of I-Fetch Sequence 5-5
SI I-Fetch, One-Cycle 5-10
Status Switching Instructions 3-5
IC P(16–23) Flip-Latch 4-205
IC Request Trigger 4-601
IC Sync Latch 4-601, 5-15 (Sheet 1)
IC Sync Trigger 4-601
Inhibit LS Write Trigger 5-21
Initial Program Load (See IPL)
Input/Output (see I/O)
Insert Character, IC(43); Logic, RX 5-408
Insert Key Trigger 4-601
Insert Storage Key, ISK (09); Stat Sw, RR 5-606
Instruction:
Requests:
During Early End Op 5-3
During End Op 5-2
Step Routine 6-13
Instruction Counter (see IC)
Instruction Fetching (see I-Fetch)
Instruction Length Not Available Trigger 5-20
Instruction Step Latch 6-12
Instructions (see Specific Instruction or Class)
Interrupt Code Triggers 5-20, 5-21, 5-22
Interruptions:
Common Routine 5-26
External 5-24
Gating, Block 6-6
Interrupt Code Triggers 5-20, 5-21, 5-22
I/O 5-25
Machine Check 5-19
Program 5-22
Supervisor Call 5-23
Interrupts Latch 6-6
Invalid Address Logic, SCI 4-604
Invalid Digit Logic, Serial Adder 4-405
Invalid Instruction Address Test Exceptional Condition 5-29
Invalid Instruction Address Trigger 5-29 (Sheet 1)

I/O:
Instructions:
Data Flow 3-6
Halt I/O, HIO (9E); SI 5-703
Start I/O, SIO (9C); SI 5-701
Test Channel, TCH (9F); SI 5-704
Test I/O, TIO (9D); SI 5-702
Interruption 5-25
System Data Flow 2-1
IPL:
Gating 6-9
Operation 6-8
IPL Status Latch 6-9

K-Register:
Data Flow 2-1
Usage:
Convert and Sort Symbols 5-903
Convert Weather Lines 5-905
Delay 5-803
DE Wrap 6-201
Mode Word 5-806
Repack Symbols 5-902

Late BCU Cleanup Latch 4-208
LM-Register:
Data Flow 2-1
Usage:
Convert Weather Lines 5-905
LM to XY Formatting 4-211
Repack Symbols 5-902
Load Address, LA (41); Logic, RX 5-409
Load and Test, LTDR (22); Fl Pt, RR (Long) 5-205
Load and Test, LTER (32); Fl Pt, RR (Short) 5-204
Load and Test, LTR (12); Fix Pt, RR 5-103
Load Complement, LCDR (23); Fl Pt, RR (Long) 5-205
Load Complement, LCER (33); Fl Pt, RR (Short) 5-204
Load Complement, LCR (13); Fix Pt, RR 5-104
Load Halfword, LH (48); Fix Pt, RX 5-102
Load, L (58); Fix Pt, RX 5-101
Load, LD (68); Fl Pt, RX (Long) 5-203
Load, LDR (28); Fl Pt, RR (Long) 5-202
Load, LE (78); Fl Pt, RX (Short) 5-203
Load, LER (38); Fl Pt, RR (Short) 5-202
Load, LR (18); Fix Pt, RR 5-101
Load Multiple, LM (98); Fix Pt, RS 5-107
Load Negative, LNDR (21); Fl Pt, RR (Long) 5-205
Load Negative, LNER (31); Fl Pt, RR (Short) 5-204
Load Negative, LNR (11); Fix Pt, RR 5-106
Load Positive, LPDR (20); Fl Pt, RR (Long) 5-205
Load Positive, LPER (30); Fl Pt, RR (Short) 5-204
Load Positive, LPR (10); Fix Pt, RR 5-105
Load PSW, LPSW (82); Stat Sw, SI 5-601
Local Storage:
Data Flow 2-1
Read/Write Controls 4-301
Usage:
B-Field Transfer to LAL 4-201
Branching Instructions 3-4
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
Operand Prefetching During End Op 5-1
R(8–11) or R(12–15) Transfer to LAL 4-202
RR I-Fetch, One-Cycle 5-7
RS I-Fetch, One-Cycle 5-10
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
Scan 6-101
SI I-Fetch, One-Cycle 5-10
Status Switching Instructions 3-5
LOG OUT Pushbutton Logic 6-25
Logical Function, SAL(0) 4-406
Logical Instructions:
AND, N (54); RX 5-404
AND, NC (D4); SS 5-404
AND, NI (94); SI 5-404
AND, NR (14); RR 5-404
Compare, SL (55); RX 5-403
Compare, CLC (D5); SS 5-403
Compare, CLI (95); SI 5-403
Compare, CLR (15); RR 5-403
Data Flow 3-3
Edit, ED (DE); SS 5-411
Edit and Mark, EDMK (DF); SS 5-411
Exclusive-OR, X (57); RX 5-406
Exclusive-OR, XC (D7); SS 5-406
Exclusive-OR, XI (97); SI 5-406
Exclusive-OR, XR (17); RR 5-406
GIS 5-401
Insert Character, IC (43); RX 5-408
Load Address, LA (41); RX 5-409
Move, MVC (D2); SS 5-402
Move, MVI (92); SI 5-402
Move Numerics, MVN (D1); SS 5-402
Move Zones, MVZ (D3); SS 5-402
OR, O(56); RX 5-405
OR, OC(D6); SS 5-405
OR, OI (96); SI 5-405
OR, OR (16); RR 5-405
Shift Left Double, SLDL (8D); RS 5-412
Shift Left Single, SLL (89); RS 5-412
Shift Right Double, SRDL (8C); RS 5-412
Shift Right Single, SRL (88); RS 5-412
Store Character, STC (42); RX 5-408
Test Under Mask, TM (91); SI 5-407
Translate and Test, TRT (DD); SS 5-410
Translate, TR (DC); SS 5-410
Logout:
Control Logic 6-108
Sequence 6-114
Logout Trigger 6-108
LS (see Local Storage)

Machine Check Interrupt Trigger 5-19
Machine Check Interruption 5-19
Main Storage, Data Flow:
I/O Instructions 3-6
System 2-1
Maintenance Control Word (see MCW)
Maintenance Features:
Diagnose (83); Stat Sw 5-609
MCW 5-609 (Sheet 1)
Maintenance Mode Stop Clock (see MMSC)
Manual Control Exceptional Conditions 5-27
Manual Trigger 6-6
Mark Triggers:
Data Flow 2-1
Positive Logic Diagram 4-209
Usage, Scan 6-101
MCW 5-609 (Sheet 1)
MCW-Register:
Data Flow 2-1
Usage:
Scan 6-101
Status Switching Instructions 3-5
Mixer:
Data Flow 2-1
Usage:
Convert Weather Lines 5-905
LM to XY Reformatting 4-211
Repack Symbols 5-902
XY Parity Prediction 4-212
MMSC Logic 6-110
MMSC Trigger 6-108, 6-110
Move, MVC (D2); Logic, SS 5-402
Move, MVI (92); Logic, SI 5-402
Move Numerics, MVN (D1); Logic, SS 5-402
Move With Offset, MVO (F1); Dec, SS:
GIS 5-307
Not Word Overlap Sequence 5-312
Word Overlap Sequence 5-313
Move Zones, MVZ (D3); Logic, SS 5-402
Multiple Computing Element Instructions:
Delay, DLY(0B); Mple, RR 5-803
Insert ATR, IATR (0E); Mple, RR 5-802
Load ID, LI(0C); Mple, RR 5-801
Load PS Base Address (Preferential Storage), LPSB (A1); Mple, SI 5-805
Move Word, MVW (D8); Mple, SS 5-806
Set Address Translator, SATR (0D); Mple, RR Receiving 5-808
Set Configuration SCON (01); Mple, RR 5-810
Start I/O Processor, SIOP (9A); Mple, SI 5-807
Store PS Base Address (Preferential Storage), SPSB (A0); Mple, SI 5-804
Test & Set, TS (93) Mple, SI 5-811
Multiple, Derivation of; Fixed-Point Multiply 5-109 (Sheet 3)
Multiply, Fixed-Point:
Algorithm 5-109 (Sheet 2)
Data Flow 5-109 (Sheet 3)
Initialization 5-109 (Sheet 1)
Multiply Floating Point, Data Flow 5-210
Multiply Halfword, MH (4C); Fix Pt, RX 5-109
Multiply, M (5C); Fix Pt, RX 5-109
Multiply, MD (6C); Fl Pt, RX (Long) 5-212
Multiply, MDR (2C); Fl Pt, RR (Long) 5-212
Multiply, ME (7C); Fl Pt, RX (Short) 5-211
Multiply, MER (3C); Fl Pt, RR (Short) 5-211
Multiply, MP (FC); Dec, SS 5-305
Multiply, MR (1C); Fix Pt, RR 5-109

N-Register:
Data Flow 2-1
Usage (Repack Symbols) 5-902

Operand Prefetching During End Op 5-1
OR Function, SAL(0) 4-406
OR, O (56); Logic, RX 5-405
OR, OC (D6); Logic, SS 5-405
OR, OI (96); Logic, SI 5-405
OR, OR (16); Logic, RR 5-405

Pack, PACK (F2); Dec, SS:
GIS 5-307
Not Word Overlap Sequence 5-308
Word Overlap Sequence 5-309
PAL(48–55) Parity Latch 4-412
Parallel Adder:
Address Gating, SCI 4-602
Bit-Position Logic, Bit 47 4-510
Carry Lookahead Logic 4-511
Data Flow 2-1
Excess-6 Logic 4-415
Full-Sum Checking Logic, PA(48–55) 4-414
Gate Control Triggers for 'B + T' Micro-order 4-409
Half-Sum Checking Logic, PA(48–55) 4-413
Parity Generation, PAL(48–55) 4-412
Set CC Logic 4-416

Usage:
Branching Instructions 3-4
Decimal Instructions 3-3
Divide, Decimal 5-306 (Sheet 1)
Divide, Fixed-Point 5-110 (Sheet 4)
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
I-Fetch Sequencer Control 5-15
Logical Instructions 3-3
Multiply, Decimal 5-305 (Sheet 1)
Multiply, Fixed-Point 5-109 (Sheet 3)
Multiply, Floating-Point 5-210
RR I-Fetch, One-Cycle 5-7
RS I-Fetch, One-Cycle 5-10
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
Scan 6-101
Scan-Out Path 6-107
SI I-Fetch, One-Cycle 5-10
Status Switching Instructions 3-5
Parity:
Generation, PAL(48–55) 4-412
IC, Adjustment for Stepping 4-205
Predict Logic, Serial Adder 4-407
Prediction, after Incrementing E-Register Incrementer 4-204
Partial Product Bit, Derivation of; Fixed-Point Multiply 5-109 (Sheet 3)
Partial Product Byte, Derivation of; Fixed-Point Multiply 5-109 (Sheet 3)
Pass Pulse Trigger 4-1, 6-12, 6-102
Pass Trigger 6-112, 6-113
PIR:
Data Flow 2-1
Usage (Diagnose) 5-609
Power-on Reset 6-7
Product/Quotient Bit Logic, Serial Adder 4-408
Program Interrupt Flip-Latch 5-22
Program Interruptions 5-22
Program Status Word (see PSW)
Program Store Compare Exceptional Condition 5-28
PROSAR A, Data Flow 4-105 (Sheet 2)
PROSAR B, Data Flow 4-105 (Sheet 2)
Protected Branch Address Trigger 5-29 (Sheet 1)
Protection Check to CPU Latch 5-20
PSA Access Error 4-608
PSBAR (Preferential Storage Base Add Reg):
Counter 4-608
Logical:
Data Flow 2-1
Usage:
Address Decode & Gating Logic 4-602
Load PSBAR 5-805
Store PSBAR 5-804
Physical:
Data Flow 2-1
Usage:
Load PSBAR 5-805
Store PSBAR 5-804
PSW Register:
Data Flow 2-1
Usage:
Branching Instructions 3-4
Status Switching Instructions 3-5
Pulse Mode:
Controls 6-23
Operation 6-24
Pulse Mode Adjust Trigger 6-23
Pulse Mode Initialization Trigger 6-23
Pulse Mode Setup Latch 6-23
Pushbutton:
Gating, Stop Loop 6-5
Signal Generation 6-3

Q-Register:
Data Flow 2-1
Usage:
B-Field Transfer to LAL 4-201
Branching Instructions 3-4
I-Fetch Sequencer Control 5-15
Instruction Requests During Early End Op 5-3
Instruction Requests During End Op 5-2
Operand Prefetching during End Op 5-1
RR I-Fetch, One-Cycle 5-7
RR I-Fetch, Two-Cycle 5-8
RS I-Fetch, One-Cycle 5-10
RS I-Fetch, Two-Cycle 5-12
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
RX I-Fetch, Two-Cycle Nonindexed 5-12
Selection of I-Fetch Sequence 5-5
SI I-Fetch, One-Cycle 5-10
SI I-Fetch, Two-Cycle 5-12
Status Switching Instructions 3-5
Q-Register Refill Exceptional Condition 5-30
Quotient Bits, Derivation of; Fixed-Point Divide 5-110 (Sheet 4)

R-Register:
Data Flow 2-1
Usage:
B-Field Transfer to LAL 4-201
Branch Requests 5-4
Branching Instructions 3-4
I-Fetch Sequencer Control 5-15
Instruction Requests During Early End Op 5-3
Instruction Requests During End Op 5-2
Operand Prefetching During End Op 5-1
R(8–11) or R(12–15) Transfer to LAL 4-202
RR I-Fetch, One-Cycle 5-7
RR I-Fetch, Two-Cycle 5-8
RS I-Fetch, One-Cycle 5-10
RS I-Fetch, Two-Cycle 5-12
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
RX I-Fetch, Two-Cycle Nonindexed 5-12
Selection of I-Fetch Sequence 5-5
SI I-Fetch, One-Cycle 5-10
SI I-Fetch, Two-Cycle 5-12
Status Switching Instructions 3-7
RATE Switch Logic 6-12
Read Direct, RDD (85) 5-607
Read-Only Storage (see ROS)
Read-Only Storage Address Register (see ROSAR)
Read-Only Storage Backup Register (see ROSBR)
Read-Only Storage Data Register (see ROSDR)
Read-Only Storage Previous Address Register A and B (see PROSAR A and PROSAR B)
Read/Write Controls, Local Storage 4-301
R(8–11) or R(12–15) Transfer to LAL 4-202
REPEAT INSN Switch:
Logic 6-15
Routine 6-16
Repeat Instruction Adjust Trigger 6-15
Repeat Instruction Initialization Latch 6-15
Repeat ROS Address Flip-Latch 4-101
REPEAT ROS ADDRESS Switch Gating 6-17
Request Sensing Logic 4-601
Reset:
Machine 6-7
RESTART FLT I/O Pushbutton Logic 6-27
Ripple Storage, Loop Routine 6-18
Roller Switches 6-2
ROS:
Addressing:
Drive Line Decode 4-105 (Sheet 1)
Select Line Decode 4-105 (Sheet 1)
Sense Amplifiers 4-105 (Sheet 1)
Sense Latches 4-105 (Sheet 1)
Sense Line and Driver Distribution 4-105 (Sheet 1)
Array Drivers 4-106
Data Flow 4-105 (Sheet 2)
ROSAR(0–5) Logic 4-101
ROSAR(6–9) Logic 4-102
ROSAR(10) Logic 4-103
ROSAR(11) Logic 4-104
Timing 4-105 (Sheet 1)
ROS Microbranching, Scan Control 6-112
ROS Test Initiate Trigger 6-26
ROS Test Latch 6-26
ROS Test Sequence 6-15
ROS Test Trigger 6-112
ROS Transfer and Repeat Address Trigger and Latch 6-17
ROS TRANSFER Switch Gating 6-17
ROSAR:
Data Flow 4-105 (Sheet 2)
ROSAR(0–5) Logic 4-101
ROSAR(6–9) Logic 4-102
ROSAR(10) Logic 4-103
ROSAR(11) Logic 4-104
Selection of I-Fetch Sequence 5-5
Usage:
I-Fetch Sequencer Control 5-15
Scan 6-101
ROSAR Latches, Data Flow 4-105 (Sheet 2)
ROSBR, Data Flow 4-105 (Sheet 2)
ROSDR:
Data Flow 4-105 (Sheet 2)
Logic 4-107
ROSDR Latches, Data Flow 4-105 (Sheet 2)
RR I-Fetch:
Flowchart 5-6
One-Cycle 5-7
Two-Cycle 5-8
RS I-Fetch:
Flowchart 5-13
One-Cycle 5-10
Two-Cycle 5-12
RX I-Fetch:
Flowchart 5-9
One-Cycle 5-10
Two-Cycle Indexed 5-11
Two-Cycle Nonindexed 5-12

S-Register (see ST-Register)
SAB Parity Conversion Logic, SCI 6-10
SADDL Parity (0–7) Latch 4-407
Sample Pulse Trigger 5-17
Scan Clock 6-102
Scan Control Triggers 6-111
Scan Counter Control Trigger and Latch 6-106
Scan Counter Latches and Decrementer 6-104
Scan/IOCE Interface 6-113
Scan Logic:
Control Triggers 6-111
Data Flow 6-101
FLT Clock 6-103
FLT Counter, Decrementing 6-106
Logout Control Logic 6-108

MMSC Logic 6-110
ROS Microbranching, Control 6-112
Scan Clock 6-102
Scan Counter Latches and Decrementer 6-104
Scan/IOCE Interface 6-113
Scan-Out Bus, Data Flow 6-107
Scan-Out Path 6-109
Scan Storage Address Generator 6-105
Scan MODE ROS/PROC/FLT Switch Logic 6-26
Scan Mode Trigger and Latch 6-108
Scan-Out Bus, Data Flow 6-107
Scan-Out Path 6-109
Scan-Out ROS (see SOROS)
Scan Request Trigger 4-601, 6-105, 6-108
Scan Storage Address Generator 6-105
Scan Sync Trigger and Latch 4-601, 6-105, 6-108
SCI:
Address Decode Logic, Basic System 4-602
Address Gating Logic 4-602
CE Request Timing, Typical 4-601
Clock Control Logic 4-603
CPU Sequencers 4-603
Data Flow 2-1
Invalid Address Logic 4-604
Request Sensing Logic 4-601
SAB Parity Conversion Logic 4-610
Single-Cycle Mode, Servicing of Requests 4-612
Storage Error Logic 4-606
Segmented Clock Inhibit Latch 4-2
Segmented Clock Storage Selected Latch 4-2
Select Line Decode, ROS 4-105 (Sheet 1)
Select Register:
Data Flow 2-1
Usage:
SATR 5-808, 809
SCON 5-810
Sense Amplifier, ROS 4-105 (Sheet 1)
Sense Latches, ROS:
Addressing 4-105 (Sheet 1)
Data Flow 4-105 (Sheet 2)
Sense Line and Driver Distribution, ROS 4-105 (Sheet 1)
Serial Adder:
AND Function, SLA(0) 4-406
Carry Lookahead Logic, SAL(0–3) 4-402
Data Flow 2-1
Decimal Add 6 Logic 4-403
Decimal Correction Logic, SAL(0–3) 4-403
Exclusive-OR Function, SAL(0) 4-406
Input Bus Logic 4-401
Invalid Digit Logic 4-405
Logical Function, SAL(0) 4-406
OR Function, SAL(0) 4-406
Parity Predict Logic 4-407
Product/Quotient Bit Logic 4-408
Usage:
Add, Subtract, and Compare, Decimal 5-302 (Sheet 1)
Decimal Instructions 2-1
Direct Control Operation 5-607 (Sheet 2)
Divide, Decimal 5-306 (Sheet 1)
Divide, Fixed-Point 5-110 (Sheet 4)
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
Logical Instructions 3-3
Multiply, Decimal 5-305 (Sheet 1)
Multiply, Fixed-Point 5-109 (Sheet 3)
Multiply, Floating-Point 5-210
Status Switching Instructions 3-5
Set Program Mask, SPM (04); Stat Sw, RR 5-602
Set Storage Key, SSK (08); Stat Sw, RR 5-605
Set System Mask, SSM (80); Stat Sw, SI 5-603
Shift Left Double, SLDA (8F); Fix Pt, RS 5-117
Shift Left Double, SLDL (8D); Logic, RS 5-412
Shift Left Single, SLA (8B); Fix Pt, RS 5-116
Shift Left Single, SLL(89); Logic, RS 5-412
Shift Right Double, SRDA (8E); Fix Pt, RS 5-119
Shift Right Double, SRDL (8C); Logic, RS 5-412
Shift Right Single, SRA (8A); Fix Pt, RX 5-118
Shift Right Single, SRL (88); Logic, RS 5-412
SI I-Fetch:
Flowchart 5-13
One-Cycle 5-10
Two-Cycle 5-12
Sign Data Path:
Floating-Point Divide 5-213
Floating-Point Multiply 5-210
Single-Cycle Inhibit Routine 6-14
Single Cycle Latch 6-12
Single-Cycle Mode, Servicing of Requests by SCI 4-612
Single-Cycle Routine 6-14
SOROS Trigger 6-25, 6-108
SS I-Fetch, Flowchart 5-14
ST Register:
Bits 15 and 16 4-306
Data Flow 2-1
Usage:
Add, Subtract, and Compare, Decimal 5-302 (Sheet 1)
Branching Instructions 3-4
Decimal Instructions 3-3
Direct Control Operation 5-607 (Sheet 2)
Divide, Decimal 5-306 (Sheet 1)
Divide, Fixed-Point 5-110 (Sheet 4)
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
I/O Instructions 3-7
Logical Instructions 3-3
Multiply, Decimal 5-305 (Sheet 1)
Multiply, Fixed-Point 5-109 (Sheet 3)
Multiply, Floating-Point 5-210
RS I-Fetch, One-Cycle 5-10
RX I-Fetch, One-Cycle 5-10
RX I-Fetch, Two-Cycle Indexed 5-11
Scan 6-101
SI I-Fetch, One-Cycle 5-10
Status Switching Instructions 3-5
ST-Register Byte Counter (see STC)
Start I/O, SIO (9C); I/O, SI 5-701
STAT A, Set CC; Parallel Adder 4-416
STAT B, Positive Logic Diagram 4-501
STAT C, Save Signs and Insert Sign; Floating-Point 5-201
STAT D, Save Signs and Insert Sign; Floating-Point 5-201
STAT F, Save Signs and Insert Sign; Floating-Point 5-201
Status Switching Instructions:
Data Flow 3-5
Diagnose (83); SI 5-609
Insert Storage Key, ISK (09); RR 5-606
Load PSW, LPSW (82); SI 5-601
Read Direct, RDD (85); SI 5-607
Set Program Mask, SPM (04); RR 5-602
Set Storage Key, SSK (08); RR 5-605
Set System Mask, SSM (80); SI 5-603
Supervisor Call, SVC (0A); RR 5-604
Test and Set, TS (93); SI 5-609
Write Direct, WRD (84); SI 5-607
STC:
Bipolar Latches 4-208
Data Flow 2-1
Incrementer Latches 4-208
Positive Logic Diagram 4-208
Triggers 4-208
Usage:
Decimal Instructions 3-3
Direct Control Operation 5-607 (Sheet 2)
Divide, Decimal 5-306 (Sheet 1)
Divide, Floating-Point 5-213
Fixed-Point Instructions 3-1
Floating-Point Instructions 3-2
Logical Instructions 3-3
Multiply, Floating-Point 5-210
Status Switching Instructions 3-5
Stop Clock Trigger 4-1
Stop Loop:
Pushbutton Gating 6-5
Routine 6-4
Stop Trigger 6-6
Storage Error Logic, SCI 4-606
Storage Ripple Loop Routine 6-18
STORAGE SELECT Switch Gating 6-10
Store Character, STC (42); Logic, RX 5-408
Store Halfword, STH (40); Fix Pt, RX 5-114
Store Latch 4-209
Store Multiple, STM (90); Fix Pt, RS 5-115
Store, ST (50); Fix Pt, RX 5-113
Store, STD (60); Fl Pt, RX (Long) 5-216
Store, STE (70); Fl Pt, RX (Short) 5-216
Store to LCS Flip-Latch 5-18
Store to Main Storage Flip-Latch 5-18
Subtract Halfword, SH (4B); Fix Pt, RX 5-108
Subtract Logical, SL (5F); Fix Pt, RX 5-108
Subtract Logical, SLR (1F); Fix Pt, RR 5-108
Subtract Normalized, SD (6B); Fl Pt, RX (Long) 5-207
Subtract Normalized, SDR (2B); Fl Pt, RR (Long) 5-207
Subtract Normalized, SE (7B); Fl Pt, RX (Short) 5-206
Subtract Normalized, SER (3B); Fl Pt, RR (Short) 5-206
Subtract, S (5B); Fix Pt, RX 5-108
Subtract, SP (FB); Dec, SS:
Complement Add Sequence 5-303
GIS 5-301
True Add Sequence 5-302
Subtract, SR (1B); Fix Pt, RR 5-108
Subtract Unnormalized, SU (7F); Fl Pt, RX (Short) 5-206
Subtract Unnormalized, SUR (3F); Fl Pt, RR (Short) 5-206
Subtract Unnormalized, SW (6F); Fl Pt, RX (Long) 5-207
Subtract Unnormalized, SWR (2F); Fl Pt, RR (Long) 5-207
Supervisor Call, SVC (0A); Stat Sw, RR 5-604
Supervisor Call Interruption 5-23
Supervisor Call Trigger 5-23
Sync Latch 6-108, 6-111
Sync Trigger 6-27, 6-108, 6-111

T-Register (see ST-Register)
1052 Adapter (9020E Only):
Block Diagram 6-28
Data Transfer-Read 6-31
Data Transfer-Write 6-30
Ending Sequence 6-32
Initial Selection-Read, Write, Sense 6-29
Sense and Status Bytes 6-33
Test and Set, TS (93); Stat Sw, SI 5-811
Test Under Mask, TM (91); Logic, SI 5-407
TIC Latch 6-112, 6-113
Time Clock at Limit Trigger 5-24
Time Clock Step Trigger 5-17
Timer Exceptional Condition 5-17
Timing Gate Trigger 3-6, 5-607 (Sheet 2)
Transfer In Channel (see TIC)
Translate, TR (DC); Logic, SS 5-410

Unpack, UNPK (F3); Dec, SS:
GIS 5-307
Not Word Overlap Sequence 5-310
Word Overlap Sequence 5-311

Wait State Gating and Microprogram 6-20
Wait State Trigger 6-20
Word Overlap Sequence:
Move With Offset, MVO (F1); Dec, SS 5-313
Pack, PACK (F2); Dec, SS 5-309
Unpack, UNPK (F3); Dec, SS 5-311
Write Direct, WRD (84); Stat Sw, SI 5-607
Write Local Store Trigger 4-301

XY-Register:
Data Flow 2-1
Usage:
Convert Weather Lines 5-905
LM to XY Formatting 4-211
Repack Symbols 5-902
XY Parity Prediction 4-212

Zero and Add, ZAP (F8); Dec, SS 5-304