File Number S360-30
Order No. GY30-2001-5

Program Logic

IBM System/360 Operating System

Basic Telecommunications Access Method

Program Logic Manual

Program Number 360S-CQ-513

This publication describes the internal logic of the Basic Telecommunications Access Method (BTAM). It is intended for use by persons involved in program maintenance and by system programmers who are altering the program design.

PREFACE

This Program Logic Manual is a guide to the internal structure of the Basic Telecommunications Access Method (BTAM). It is designed to be used with the program listing; program structure at the machine instruction level is not discussed.

Effective use of this manual requires a knowledge of the concepts presented in the following IBM System/360 publications:

- IBM System/360: Principles of Operation, GA22-6821
- IBM System/360 Operating System: Assembler Language, GC28-6514
- IBM System/360 Operating System: Basic Telecommunications Access Method, GC30-2004

In addition, the following publications may be used when information about other elements of the control program is required:

- IBM System/360 Operating System: Input/Output Support (OPEN/CLOSE/EOV), Program Logic Manual, GY28-6609
- IBM System/360 Operating System: System Programmer's Guide, GC28-6550
- IBM System/360 Operating System: Fixed-Task Supervisor, Program Logic Manual, GC28-6612
- IBM System/360 Operating System: Input/Output Supervisor, Program Logic Manual, GY28-6616
- IBM General Information -- Binary Synchronous Communications, GA27-3004

This publication is divided into three sections: a discussion of the general organization of the Basic Telecommunications Access Method (BTAM); a summary of the internal logic of each BTAM element; and a section illustrating internal control blocks and describing control conventions.

Sixth Edition (February 1972)

This edition, GY30-2001-5, is a revision of and renders obsolete GY30-2001-4 and associated Technical Newsletters GY30-2552 and GY30-2564. This edition applies to OS Release 20.6/20.7 and Release 21 combined with the Independent Component Release containing BTAM support for the IBM 3270 Display System.

Minor changes to the text or a small change to an illustration on existing pages is indicated by a vertical line to the left of a change. The following pages are "added": vi.1, viii.1, 14.1, 20.1-20.6, 30.1-30.4, 128.1-128.3, 136.1-136.5, 154.1-154.2, 156.1-156.5, 160.1-160.2, 194.1, 214.1, 224.1-224.34, 244.1, 244.3, 244.5, and 249-253.

Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems or equipment, refer to the latest SRL newsletter for the editions that are applicable and current.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by the IBM Systems Development Division, Publications Center, Department E01, P. O. Box 12275, Research Triangle Park, N. C. 27709. A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be sent to the above address. Comments become the property of IBM.

INTRODUCTION 9
BTAM Macro Instructions 9
BTAM Control Blocks 10
Data Control Block (DCB) 10
Data Extent Block (DEB) 11
Input/Output Block (IOB) 11
Unit Control Block (UCB) 11
Data Event Control Block (DECB) . . 11
Event Control Block (ECB) 11
Interruption Request Block (IRB) . 12
Interruption Queue Element 12
Role of the Control Blocks in Creating the Access Method 12
Evolution of the BTAM Read Command for Remote Stations 13
System Generation 13
Assembly 14
Execution 14
Open Line Groups 14
Read 14.1
System Flow for Local 3270 Operations Using BTAM 20
Read Initial Operation 20
Read and Write Operations 20
CHGNTRY Skip Operation 20.3
CHGNTRY Activate Operation 20.3
RESETPL Operation 20.3
Attention Interruption Handling . . 20.3
Dynamic Buffering 20.6
Read Initial Channel Programs using Auto Poll (Multipoint Only) . . . 21

BTAM ROUTINES AND ASSOCIATED MODULES . 22
BTAM Routines 22
BTAM Open Executor (IGG0193M, IGG0194N, IGG0193Q, and IGG0193S). . . 22
Local 3270 BTAM Open Executor (IGG0194P and IGG0194Q) 23
BTAM Read/Write Routine (IGG019MA). 23
Read/Write Subroutines to Compute Area Address 24
Read/Write Subroutines to Compute Count 26
Channel End and Abnormal End Appendage (IGG019MB) 26
World Trade Telegraph Terminal Channel End and Abnormal End Appendage (IGG019PD) 27
Local 3270 Channel End/Abnormal End Appendage (IGG019PA) 28
PCI (Program Controlled Interrupt) Appendage (IGG019MC) 29
Local 3270 First-Level Attention Routine (IECTATEN) 30
Local 3270 Second-Level Attention Routine (IGG019FG) 30.1
Local 3270 BTAM SVC Routine (IECTSVC) 30.1
Error Recovery Procedures 30.3
Buffer Routine (IGG019MS) 30.4
Change Entry Routine (IECTCHGN) . . 31
Edit Routine (IECTEDIT) 32
Line Error Print Routine (IECTLERP) 32
Line Open Routine (IECTIOPN) . . . 33
Translate Routine (IECTTRNS) . . . 33
On-Line Testing 34
On-Line Testing for Start-Stop Communication Lines 34
On-Line Testing for Binary Synchronous Communication Lines . 34
On-Line Testing for Local 3270 Devices 36
REQBUF/RELBUF (SVC 58) Routine (IGC058) 36
BTAM Close Executor (IGG0203M) . . 36
BTAM Control Information for Channel Program Generation 36
BTAM Channel Command Word 36
Device I/O Modules 36
Read/Write Operation Types 37
List of Device I/O Modules 39
Device I/O Directory 39

BTAM CHANNEL PROGRAMS -- GENERAL 40
BTAM Channel Program Format 40
START-STOP CHANNEL PROGRAMS 43
Channel Programs for AT&T 83B3 Selective Calling Stations 43
Read Initial Channel Program 43
Read Skip Channel Program 43
Write Initial Channel Program . . . 43
Write Break Channel Program 44
Reset Following a Write 44
Channel Programs for Western Union Plan 115A Outstations 45
Read Initial Channel Program 45
Read Skip Channel Program 45
Write Initial Channel Program . . . 46
Write Break Channel Program 46
Reset Following a Write 46
Channel Programs for IBM 1030 Terminals 47
Read Initial Channel Program 47
Read Continue Channel Program . . . 47
Read Repeat Channel Program 47
Read Skip Channel Program 47
Reset Following a Read 48
Write Initial Channel Program . . . 48
Write Continue Channel Program . . . 48
Write Positive Acknowledge Channel Program 48
Write Negative Acknowledge Channel Program 48
Reset Following a Write 48
Channel Programs for IBM 1050 Lines . . 49
Read Initial Program 49
Read Continue Channel Program . . . 49
Read Repeat Channel Program 49
Read Skip Channel Program 49
Reset Following a Read 49
Write Initial Channel Program . . . 50
Write Continue Channel Program . . . 50
Write Continue Conversational Channel Program 50

Write Positive Acknowledge Channel Program 51
Write Negative Acknowledge Channel Program 51
Reset Following a Write 51
Channel Programs for IBM 1050 Dial (Switched Connection Lines) 52
Read Initial Channel Program 52
Read Continue Channel Program . . . 52
Read Conversational Channel Program 52
Read Repeat Channel Program 53
Read Skip Channel Program 53
Reset Following a Read 53
Write Initial Channel Program . . . 53
Write Continue Channel Program . . . 54
Write Conversational Channel Program 54
Write Continue Conversational Channel Program 54
Write Positive Acknowledge and Disconnect Channel Program 54
Write Negative Acknowledge and Disconnect Channel Program 55
Reset Following a Write 55
Channel Programs for IBM 1060 Terminals 56
Read Initial Channel Program 56
Read Continue Channel Program . . . 56
Read Repeat Channel Program 56
Read Skip Channel Program 56
Reset Following a Read 57
Write Initial Channel Program . . . 57
Write Positive Acknowledge Channel Program 57
Write Negative Acknowledge Channel Program 57
Reset Following a Write 57
Channel Programs for WU Models 33 and 35 TWX Lines 58
Read Initial Channel Program 58
Read Conversational Channel Program 58
Read Skip Channel Program 58
Reset Following a Read 58
Write Initial Channel Program . . . 58
Write Conversational Channel Program 59
Write Negative Acknowledge Channel Program 59
Reset Following a Write 59
Channel Programs for IBM 2740 Communications Lines 60
IBM 2740 Basic Channel Programs . . . 60
Read Initial Channel Program 60
Read Skip Channel Program 60
Write Initial Channel Program . . . 60
Reset Following a Write 60
IBM 2740 With Checking 61
Read Initial Channel Program 61
Read Continue Channel Program . . . 61
Read Repeat Channel Program 61
Read Skip Channel Program 61
Reset Following a Read 61
Write Initial Channel Program . . . 61
Write Continue Channel Program . . . 61
Write Conversational Channel Program 62
Write Continue Conversational Channel Program 62
Write Positive Acknowledge Channel Program 62
Write Negative Acknowledge Channel Program 62
Reset Following a Write 62
IBM 2740 With Checking and 2760 Attachment 63
Read Initial Channel Program 63
Read Continue Channel Program . . . 63
Read Continue with Leading Acknowledgment Channel Program . . . 63
Read Repeat Channel Program 63
Read Skip Channel Program 63
Reset Following a Read 64
Write Initial Channel Program . . . 64
Write Continue Channel Program . . . 64
Write Conversational Channel Program 64
Write Initial Optical Channel Program 64
Write Invitational Optical Channel Program 65
Write Positive Acknowledgment Channel Program 65
Write Negative Acknowledgment Channel Program 65
Reset Following a Write 65
IBM 2740 With Dial 66
Read Initial Channel Program 66
Read Conversational Channel Program 66
Read Skip Channel Program 66
Reset Following a Read 66
Write Initial Channel Program . . . 66
Write Conversational Channel Program 66
Write Disconnect Channel Program . . 67
Reset Following a Write 67
IBM 2740 With Dial and Checking . . . 68
Read Initial Channel Program 68
Read Continue Channel Program . . . 68
Read Conversational Channel Program 68
Read Repeat Channel Program 68
Read Skip Channel Program 68
Reset Following a Read 68
Write Initial Channel Program . . . 69
Write Continue Channel Program . . . 69
Reset Following a Write 69
Write Conversational Channel Program 69
Write Continue Conversational Channel Program 69
Write Positive Acknowledge and Disconnect Channel Program 70
Write Negative Acknowledge and Disconnect Channel Program 70
IBM 2740 With Dial, Checking, and 2760 Attachment 70
Read Initial Channel Program 70
Read Continue Channel Program . . . 70
Read Continue with Leading Acknowledgment Channel Program . . . 70
Read Conversational Channel Program 70
Read Repeat Channel Program 71
Read Skip Channel Program 71
Reset Following a Read 71
Write Initial Channel Program . . . 71
Write Continue Channel Program . . . 71

Write Conversational Channel Program 71
Write Initial Optical Channel Program 72
Write Invitational Optical Channel Program 72
Write Conversational Optical Channel Program 72
Write Positive Acknowledgment Channel Program 72
Write Negative Acknowledgment Channel Program 73
IBM 2740 With Dial and Transmit Control 74
Read Initial Channel Program 74
Read Conversational Channel Program 74
Read Skip Channel Program 74
Reset Following a Read 74
Write Initial Channel Program . . . 75
Write Conversational Channel Program 75
Write Disconnect Channel Program . . 75
Reset Following a Write 75
IBM 2740 With Dial, Transmit Control, and Checking 76
Read Initial Channel Program 76
Read Continue Channel Program . . . 76
Read Repeat Channel Program 76
Read Conversational Channel Program 77
Read Skip Channel Program 77
Reset Following a Read 77
Write Initial Channel Program . . . 77
Write Continue Channel Program . . . 78
Write Conversational Channel Program 78
Write Positive Acknowledge and Disconnect Channel Program 78
Write Negative Acknowledge Channel Program 78
Reset Following a Write 78
IBM 2740 With Station Control 79
Read Initial Channel Program 79
Read Skip Channel Program 79
Write Initial Channel Program . . . 79
Reset Following a Write 79
IBM 2740 With Station Control and Checking 80
Read Initial Channel Program 80
Read Continue Channel Program . . . 80
Read Repeat Channel Program 80
Read Skip Channel Program 80
Reset Following a Read 80
Write Initial Channel Program . . . 81
Write Continue Channel Program . . . 81
Write Positive Acknowledge Channel Program 81
Write Negative Acknowledge Channel Program 82
Reset Following a Write 82
Channel Programs for IBM 2741 Communications Terminals 83
Read Initial Channel Program (Nonswitched Line) 83
Read Initial Channel Program (Switched Line) 83
Read Conversational Channel Program 83
Read Skip Channel Program 83
Write Continue Channel Program . . . 83
Write Conversational Channel Program 84
Write Continue Conversational Channel Program 84
Write Disconnect Channel Program (Switched Line Only) 84
Channel Programs for IBM 2848 - 2260 Remote Lines 85
Read Initial Channel Program 85
Read Continue Channel Program . . . 85
Read Repeat Channel Program 85
Read Skip Channel Program 85
Read Buffer Channel Program 86
Reset Following a Read 86
Write Initial Channel Program . . . 86
Write at Line Address Channel Program 86
Write Erase Channel Program 87
Write Continue Channel Program . . . 87
Write Positive Acknowledge Channel Program 87
Write Negative Acknowledge Channel Program 87
Reset Followed by a Write 87
Channel Programs Employing Start-Stop Auto Poll 88
READ Initial (TI) with SSALST . . . 88
READ Initial (TI) with SSAWLST . . . 89
READ Continue (TT) 90
READ Repeat (TP) 90

BINARY SYNCHRONOUS COMMUNICATIONS CHANNEL PROGRAMS 91
Nonswitched Point-to-Point Line 91
Read Channel Programs 91
Read Initial (TI) 91
Read Continue (TT) 92
Read Continue with Leading Graphics (TTL) 92
Read Repeat (TP) 93
Read Repeat with Leading Graphics (TPL) 93
Read Initial Inquiry (TIQ) 94
Read Inquiry (TQ) 94
Read Interrupt (TRV) 94
Write Channel Programs 95
Write Initial Transparent Block (TIX) 96
Write Initial Conversational (TIV) . 97
Write Initial Conversational Transparent (TIVX) 97
Write Continue Transparent Block (TTX) 99
Write Continue Conversational (TTV) 99
Write Continue Conversational Transparent (TTVX) 100
Write Inquiry (TQ) 100
Write Reset (TR) 101
Write Wait-Before-Transmit (TW) . .101
Switched Point-to-Point Line 102
Read Channel Programs 102
Read Initial (TI) (Using Automatic Answering List) 102
Read Initial (TI) (Using Automatic Calling List) 102
Read Initial (TI) (Using Manual Answering List) 103

Read Connect (TC) (Using Automatic Answering List - SWLST) 104
Read Connect with Tone (TCW) (Using Automatic Answering List) . .105
Read Continue (TT) 106
Read Continue with Leading Graphics (TTL) 106
Read Repeat (TP) 107
Read Repeat with Leading Graphics (TPL) 107
Read Inquiry (TQ) 107
Read Interrupt (TRV) 108
Write Channel Programs 108
Write Initial (TI) 109
Write Initial Transparent (TIX) . .109
Write Initial Transparent Block (TIF) 110
Write Initial Conversational (TIV) .110
Write Initial Conversational Transparent (TIVX) 111
Write Connect (TC) 112
Write Continue (TT) 112
Write Continue Transparent (TTX) . .112
Write Continue Transparent Block (TTE) 113
Write Continue Conversational (TTV) 113
Write Continue Conversational Transparent (TTVX) 114
Write Inquiry (TQ) 114
Write Reset (TR) 115
Write Wait-Before-Transmit (TW) . .115
Write Break (TB) 115
Write Disconnect (TD) 116
Nonswitched Multipoint Line 117
Read Channel Programs 117
Read Initial (TI) (Using Open Polling List) 117
Read Initial (TI) (Using Wraparound Polling List) 118
Read Continue (TT) 119
Read Continue with Leading Graphics (TTL) 119
Read Repeat (TP) 119
Read Repeat with Leading Graphics (TPL) 120
Read Inquiry (TQ) 120
Read Interrupt (TRV) 120
Write Channel Programs 121
Write Initial Transparent Block (TIE) 123
Write Initial Conversational (TIV) .123
Write Initial Conversational Transparent (TIVX) 124
Write Continue Transparent Block (TTE) 125
Write Continue Conversational (TTV) 126
Write Continue Conversational Transparent (TTVX) 126
Write Inquiry (TQ) 127
Write Reset (TR) 127
Write Wait-Before-Transmit (TW) . .127

LOCAL CHANNEL PROGRAMS 128.1
Local IBM 3270 Display System 128.1
Read Channel Programs 128.1
Read Initial (TI) 128.1
Read Modified (TM) 128.1
Read Modified from Position (TMP) 128.1
Read Buffer (TB) 128.2
Read Buffer from Position (TBP).128.2
Write Channel Programs 128.2
Write Initial (TI) 128.2
Write Erase (TS) 128.2
Write Unprotected Erase (TUS) .128.3

CHARTS 129

APPENDIX A: OPERATING SYSTEM CONTROL BLOCK LINKAGES 225

APPENDIX B: DATA CONTROL BLOCK 226

APPENDIX C: DATA EXTENT BLOCK 230

APPENDIX D: INPUT/OUTPUT BLOCK 232

APPENDIX E: EVENT CONTROL BLOCK. . . .235

APPENDIX F: DATA EVENT CONTROL BLOCK. .236

APPENDIX G: UNIT CONTROL BLOCK 241

APPENDIX H: UCB DEVICE CODES 243

APPENDIX I: INTERRUPTION REQUEST BLOCK 244.3

APPENDIX J: INTERRUPTION QUEUE ELEMENT 244.5

APPENDIX K: LIST OF BTAM MODULES . . .245
I. Listed Alphabetically by Module Name 245
II. Listed by Routine 246

APPENDIX L: MISCELLANEOUS ROUTINES . .249
Local 3270 ERP Routines (IGE0010E and IGE0110E) 249
TP Recorder (IGEC625F) 249
3270 SCAN (IECTSCAN) 249
SCAN Module Description 250
SCAN Subroutines 251

INDEX 255

FIGURES

Figure 1. BTAM Read Operation . . 16
Figure 2. Control Flow for Read/Write Operation With No Dynamic Buffering 17
Figure 3. Control Flow for Read/Write Operation With Dynamic Buffering .19
Figure 3A. System Flow for Local 3270 Read Initial Operation 20.1
Figure 3B. System Flow for Local 3270 Read and Write Operations (Other than Read Initial) 20.2
Figure 3C. System Flow for Local 3270 CHGNTRY SKIP 20.4
Figure 3D. System Flow for Local 3270 CHGNTRY ACTIVATE 20.4
Figure 3E. System Flow for Local 3270 RESETPL 20.5
Figure 3F. System Flow for Local 3270 Attention Interruption Handling . . 20.5
Figure 4. Logic Flow for Dynamic Buffering 20.6
Figure 5. Dynamic Buffering Read/Write CCWs 30
Figure 6. Buffer Control Block . . 31
Figure 7. Buffer Request Queue . . 31
Figure 8. Format of Line Error Recording Block (LERB) 33
Figure 9. Format of Device I/O Module (showing nonswitched 1050 Offsets) 38
Figure 10. Format of Device I/O Directory 39
Figure 11. Channel Command Codes . 41
Figure 12. TP-Op Codes for BTAM Channel Programs . . 41
Figure 13. Control Block Linkages 225
Figure 14. Data Control Block (DCB) 226
Figure 15. Data Extent Block (DEB) 230
Figure 16. IOB CSW Field 232
Figure 17. Input/Output Control Block (IOB) 233
Figure 18. ECB After WAIT 235
Figure 19. ECB After Post 235
Figure 20. Data Event Control Block (DECB) 236
Figure 21. Unit Control Block (UCB) 241
Figure 22. Interruption Request Block (IRB) 244.3
Figure 23. Interruption Queue Element (IQE) 244.5
Figure 24. Display Alphanumeric Control Area (DACA) 249

TABLES

Table 1. Data Control Block Fields 227
Table 2. BTAM LEB Fields 231
Table 3. BTAM IOB Fields 234
Table 4. BTAM DECB Fields 237
Table 5. BTAM UCB Fields 242
Table 6. BTAM IRB Fields 244.3
Table 7. BTAM IQE Fields 244.5
Table 8. DACA Fields 250

Chart 01. BTAM Open Executor (IGG0193M) (Load 1) (Part 1 of 2) .129
Chart 01A. BTAM Open Executor (IGG0193M) (Load 1) (Part 2 of 2) .130
Chart 01B. BTAM Open Executor (IGG0194N) (Load 2) 131
Chart 02. BTAM Open Executor (IGG0193Q) (Load 3) (Part 1 of 2) .132
Chart 03. BTAM Open Executor (IGG0193Q) (Load 3) (Part 2 of 2) .133
Chart 04. BTAM Open Executor (IGG0193S) (Load 4) (Part 1 of 2) .134
Chart 05. BTAM Open Executor (IGG0193S) (Load 4) (Part 2 of 2) .135
Chart 05A. BTAM Open Executor (Common to all Loads) WTG Table Routine 136
Chart 05B. Local 3270 BTAM Open Executor(IGG0194P) (Part 1 of 3) . . .136.1
Chart 05C. Local 3270 BTAM Open Executor(IGG0194P) (Part 2 of 3) . . .136.2
Chart 05D. Local 3270 BTAM Open Executor(IGG0194P) (Part 3 of 3) . . .136.3
Chart 05E. Local 3270 BTAM Open Executor(IGG0194Q) (Part 1 of 2) . . .136.4
Chart 05F. Local 3270 BTAM Open Executor(IGG0194Q) (Part 2 of 2) . . .136.5
Chart 06. Buffer (IGG019MS) 137
Chart 07. Line Open (IECTLOPN) . . .138
Chart 08. Translate (IECTTRNS) . . .139
Chart 09. REQBUF/RELBUF (IGC058) . .140
Chart 10. BTAM Close Executor (IGG0203M) 141
Chart A1. BTAM Read/Write (IGG019MA) (Part 1 of 4) 142
Chart A2. BTAM Read/Write (IGG019MA) (Part 2 of 4) 143
Chart A3. BTAM Read/Write (IGG019MA) (Part 3 of 4) 144
Chart A4. BTAM Read/Write (IGG019MA) (Part 4 of 4) 145
Chart B1. Channel End Appendage (IGG019MB) (Part 1 of 6) 146
Chart B2. Channel End Appendage (IGG019MB) (Part 2 of 6) 147
Chart B3. Channel End Appendage (IGG019MB) (Part 3 of 6) 148
Chart B4. Channel End Appendage (IGG019MB) (Part 4 of 6) 149
Chart B5. Channel End Appendage (IGG019MB) (Part 5 of 6) 150
Chart B5A. Channel End Appendage (IGG019MB) (Part 6 of 6) 151
Chart B6. WTTA Channel End Appendage (IGG019PD) (Part 1 of 3) .152
Chart B7. WTTA Channel End Appendage (IGG019PD) (Part 2 of 3) .153
Chart B8. WTTA Channel End Appendage (IGG019PD) (Part 3 of 3) .154
Chart B9. Local 3270 Channel End/ Abnormal End Appendage(Part 1 of 2).154.1
Chart B10. Local 3270 Channel End/ Abnormal End Appendage (IGG019PA) (Part 2 of 2) 154.2
Chart C1. Program Control Interrupt Appendage (IGG019MC) 155
Chart C2. Local 3270 First-Level Attention Routine (IECTATEN) 156
Chart C3. Local 3270 Second-Level Attention Routine (IGG019PG) 156.1
Chart C4. Local 3270 BTAM SVC Routine (IECTSVC)(Part 1 of 3) 156.2
Chart C5. Local 3270 BTAM SVC Routine (IECTSVC)(Part 2 of 3) 156.3
Chart C6. Local 3270 BTAM SVC Routine (IECTSVC) (Part 3 of 3) 156.4
Chart D1. On-Line Test Control (IGG019MR)(Part 1 of 5) 156.5
Chart D2. On-Line Test Control (IGG019MR)(Part 2 of 5) 157
Chart D3. On-Line Test Control (IGG019MR)(Part 3 of 5) 158
Chart D4. On-Line Test Control (IGG019MR)(Part 4 of 5) 159
Chart D5. On-Line Test Control (IGG019MR)(Part 5 of 5) 160
Chart D5A. Local On-Test Control (IGG019PI)(Part 1 of 2) 160.1
Chart D5B. Local On-Test Control (IGG019PI)(Part 2 of 2) 160.2
Chart D6. On-Line Test EBCDIC and USASCII/TRANSCODE Message (IGC0706F) (IGC0806F) 161
Chart E0. BTAM Start/Stop ERP Control (IGE0004A) 162
Chart E1. Start/Stop ERP Data Check (IGE0104A) 163
Chart E2. Start/Stop ERP Time Out (IGE0204A) (Part 1 of 2) 164
Chart E2A. Start/Stop ERP Time Out (IGE0204A) (Part 2 of 2) 165
Chart E3. Start/Stop ERP Intervention Required (IGE0304A) (Part 1 of 2) 166
Chart E3A. Start/Stop ERP Intervention Required (IGE0304A) (Part 2 of 2) 167
Chart E4. Start/Stop ERP Lost Data (IGE0404A) 168
Chart E5A. Start/Stop ERP Post (IGE0504A) (Part 1 of 2) 169
Chart E5B. Start/Stop ERP Post (IGE0504A) (Part 2 of 2) 170
Chart E6. Start/Stop ERP Bus Out Error Check (IGE0604A) 171
Chart E7. Start/Stop ERP Read Skip Write Break (IGE0704A) (Load 1). . .172
Chart E8. Start/Stop ERP Status Check (IGE0804A) 173
Chart E9. BTAM Start/Stop ERP Control (IGE0904A) (Load 2) 174

Chart F0. Start/Stop ERP Data Check (IGE0004B) (Load 2) 175
Chart F1. Start/Stop ERP Diagnostic Write/Read (IGE0104B) . . .176
Chart F2. ERP Line Error Recording (Start/Stop and BSC) (IGE0204B) . . .177
Chart F3. Start/Stop ERP Unit Exception (IGE0304B) 178
Chart F4. Start/Stop ERP Read Skip Write Break (IGE0404B) (Load 2) . . .179
Chart F5. Start/Stop ERP Overrun (IGE0504B) 180
Chart F6. ERP Intervention Required Message Writer (Start/Stop & BSC) (IGE0604B) 181
Chart F7. Line Error Print (IECTLERP) 182
Chart F8. ERP Channel Check & Interface Control Check (Start/Stop & BSC) (IGE0804B) 183
Chart F9. ERP Channel Check & Interface Control Check (Start/Stop & BSC) (IGE0804B) (Part 2 of 2) 184
Chart G0. BSC ERP Control (IGE0004C) (Part 1 of 3) 185
Chart G1A. BSC ERP Data Check (IGE0104C) (Part 1 of 2) 186
Chart G1B. BSC ERP Data Check (IGE0104C) (Part 2 of 2) 187
Chart G2A. BSC ERP Time Out (IGE0404C) (Part 1 of 2) 188
Chart G2B. BSC ERP Time Out (IGE0404C) (Part 2 of 2) 189
Chart G3. BSC ERP Intervention Required (IGE0304C) 190
Chart G4. BSC ERP Lost Data (IGE0604C) 191
Chart G5A. BSC ERP Error Post (IGE0204C) (Part 1 of 2) 192
Chart G5B. BSC ERP Error Post (IGE0204C) (Part 2 of 2) 193
Chart G5C. Remote 3270 Error Post Routine (IGE0704B) 194
Chart G6. BSC ERP Control (IGE0004C) (Part 2 of 3) 194.1
Chart G6A. BSC ERP Control (IGE0004C) (Part 3 of 3) 195
Chart G9A. BSC ERP Equipment Check & Command Reject (IGE0804C) (Part 1 of 2) 196
Chart G9B. BSC ERP Equipment Check & Command Reject (IGE0804C) (Part 2 of 2) 197
Chart J0A. BSC ERP Special Return (IGE0504C) (Part 1 of 2) 198
Chart J0B. BSC ERP Special Return (IGE0504C) (Part 2 of 2) 199
Chart J1. BSC ERP Bus Out & Overrun (IGE0704C) 200
Chart J2A. BSC ERP Unit Exception Routine (IGE0904C) (Part 1 of 3) . . .201
Chart J2B. BSC ERP Unit Exception Routine (IGE0904C) (Part 2 of 3) . . .202
Chart J2C. BSC ERP Unit Exception Routine (IGE0904C) (Part 3 of 3) . . .203
Chart K1. Change Entry (IFCTCHGN) (Part 1 of 2) 204
Chart K2. Change Entry (IFCTCHGN) (Part 2 of 2) 205
Chart L1. 2741 Break (IGG019PK) . . .206
Chart M1. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 1 of 5) 207
Chart M2. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 2 of 5) 208
Chart M3. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 3 of 5) 209
Chart M4. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 4 of 5) 210
Chart M5. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 5 of 5) 211
Chart TL. On-Line Test Request (IECTONLT) 212
Chart T0. On-Line Test Control (Start/Stop) (IGC0006F) 213
Chart T01. On-Line Test Control (BSC) (IGC0D06F) 214
Chart T02. On-Line Test Control (Local 3270) (IGC1206F) 214.1
Chart T1. IBM 1030 Terminal Test (IGC0106F) 215
Chart T2. IBM 1050 Terminal Test (IGC0206F) 216
Chart T3. IBM 1060 Terminal Test (IGC0306F) 217
Chart T4. IBM 2740 Terminal Test (IGC0406F) 218
Chart T5. IBM 2848/2260 Terminal Test (IGC0504F) (Load 1) 219
Chart T6A. IBM 2848/2260 Terminal Test (IGC0606F) (Load 2) (Part 1 of 2) 220
Chart T6B. IBM 2848/2260 Terminal Test (IGC0606F) (Load 2) (Part 2 of 2) 221
Chart T7. IBM 2741 Terminal Test (Correspondence Code) (IGC0906F) (PTTC Code) (IGC0A06F) 222
Chart T8. IBM 2760 Terminal Test (IGC0B06F) 223
Chart T9. IBM 2740 Terminal Test (Correspondence Code) (IGC0C06F) . . .224
Chart T10. 3270 EBCDIC Test Module Types 23 and 24) (IGC0E06F) 224.1
Chart T11. 3270 EBCDIC Test Module (Type 25) (IGC1406F) 224.2
Chart T12. 3270 EBCDIC Test Module (Types 26 and 27) (IGC0F06F) 224.3
Chart T13. 3270 EBCDIC Test Module (Type 29) (IGC1306F) 224.4
Chart T14. Remote 3270 ASCII Test Module (Types 29, 30, and 32) (IGC1006F) .224.5
Chart T15. Remote 3270 ASCII Test Module (Types 31, 33, and 34) (IGC1106F) .224.6
Chart X. Local 3270 ERP Routine (IGE0010E) (Part 1 of 6) 224.7
Chart X. Local 3270 ERP Routine (IGE0010E) (Part 2 of 6) 224.8
Chart X. Local 3270 ERP Routine (IGE0010E) (Part 3 of 6) 224.9

Chart X. Iocal 3270 ERP Routine (IGE0010E) (Part 4 of 6) 224.10
Chart X. Local 3270 ERP Routine (IGE0010E) (Part 5 of 6) 224.11
Chart X. Iocal 3270 ERP Routine (IGE0010E) (Part 6 of 6) 224.12
Chart Y. Local 3270 ERP Routine (IGE0110E) (Part 1 of 3) 224.13
Chart Y. Local 3270 ERP Routine (IGE0110E) (Part 2 of 3) 224.14
Chart Y. Local 3270 ERP Routine (IGE0110E) (Part 3 of 3) 224.15
Chart Z. 3270 SCAN (IECTSCAN) (Part 1 of 19) 224.16
Chart Z. 3270 SCAN (IECTSCAN) (Part 2 of 19) 224.17
Chart Z. 3270 SCAN (IECTSCAN) (Part 3 of 19) 224.18
Chart Z. 3270 SCAN (IECTSCAN) (Part 4 cf 19) 224.19
Chart Z. 3270 SCAN (IECTSCAN) (Part 5 of 19) 224.20
Chart Z. 3270 SCAN (IECTSCAN) (Part 6 of 19) 224.21
Chart Z. 3270 SCAN (IECTSCAN) (Part 7 of 19) 224.22
Chart Z. 3270 SCAN (IECTSCAN) (Part 8 of 19) 224.23
Chart Z. 3270 SCAN (IECTSCAN) (Part 9 of 19) 224.24
Chart Z. 3270 SCAN (IECTSCAN) (Part 10 of 19) 224.25
Chart Z. 3270 SCAN (IECTSCAN) (Part 11 of 19) 224.26
Chart Z. 3270 SCAN (IECTSCAN) (Part 12 cf 19) 224.27
Chart Z. 3270 SCAN (IECTSCAN) (Part 13 of 19) 224.28
Chart Z. 3270 SCAN (IECTSCAN) (Part 14 of 19) 224.29
Chart Z. 3270 SCAN (IECSTSCAN) (Part 15 of 19) 224.30
Chart Z. 3270 SCAN (IECTSCAN) (Part 16 of 19) 224.31
Chart Z. 3270 SCAN (IECTSCAN) (Part 17 of 19) 224.32
Chart Z. 3270 SCAN (IECTSCAN) (Part 18 of 19) 224.33
Chart Z. 3270 SCAN (IECTSCAN) (Part 19 cf 19) 224.34

This manual:

- Describes the macro instructions and routines that make up the Basic Telecommunications Access Method;
- Explains the internal logic of optional BTAM facilities, such as Auto Poll and dynamic buffering;
- Shows the formats and contents of the control blocks used in BTAM operation;
- Describes the channel program operation for each of the types of remote or local stations and line configurations supported by BTAM;
- Describes the method by which BTAM Read and Write channel programs are constructed;
- Charts the internal logic of each of the BTAM routines;
- Lists all BTAM modules.

It is assumed that the reader is familiar with the manuals listed in the Preface.

BTAM MACRO INSTRUCTIONS

The BTAM macro instructions provide the user with his means of control over BTAM. They can be divided into ten different categories:

1. Macros used to establish and to modify terminal and line control information.
   - DFTRMLST - creates terminal lists that contain the characters necessary to perform polling, addressing, dialing, answering, automatic polling, or ID checking.
   - CHGNTRY - allows deactivating or activating a polling or addressing entry in a terminal list, or changing control information within an entry, without redefining the list. For a local 3270 display station, CHGNTRY sets or resets the skip flag in the UCB to inhibit or permit Read Initial operations for the device.
   - READ or WRITE (List or Standard form) - creates a data event control block (DECB). Normally, one DECB will be created for each line or local device in the line group. The DECB contains the event control block (ECB) and serves as a parameter list for the READ and WRITE macro instructions of the executable form (MF=E).
2. Macro used to identify a communication line group and to specify any optional features of BTAM that are to be utilized.
   - DCB - creates and initializes a data control block (DCB) for a communication line group. This is the only DCB essential to BTAM.
3. Macro used to activate a line group.
   - OPEN - prepares communication lines or local devices for use, initializes control blocks, and loads BTAM routines.
4. Macro used to correct a line error which occurred during Open.
   - LOPEN - issues SAD (Set Address) or Enable commands to correct a line error. For a local 3270 display station, LOPEN issues an Erase/Write command to unlock the keyboard and reset the modified data tags.
5. Macros used to keep error counts by line and to print error statistics messages.
   - LERB - creates a block to accumulate counts of errors and number of transmissions; prints a message which identifies the line and contains the error counters. Error counters for data check, intervention required, and nontext time-out errors are kept by line on a threshold basis. A message (IFC801I) is printed whenever any of the counters reaches its threshold value before the transmission count is reached, or whenever the transmission count reaches its threshold value.

- LERPRT - prints at the operator's console a message (IFC802I) containing the current values of the cumulative counters that are incremented each time a threshold error count is reached or the transmission count is reached, whichever occurs first.

6. Macros used to obtain and to release buffers.

   - REQBUF - provides one or more buffers if they are available.

   - RELBUF - fulfills an outstanding buffer request if dynamic buffering is specified, or returns a buffer to the buffer pool.

7. Macros used to control the lines and to perform data transmission.

   - READ (Execute [MF=E] form) - updates the DECB parameter list, generates linkage to the BTAM Read/Write routine, and causes BTAM to perform a specified function (such as polling a terminal and reading a message into an input buffer area).

   - WRITE (Execute [MF=E] form) - is the same as READ, except that it normally causes BTAM to perform such functions as addressing a terminal and writing a message to the terminal from an output buffer area.

   - RESETPL - forces a Read operation to terminate by interrupting a repetitive polling operation, or on a switched line, forces a Read or Write operation to terminate by halting an Enable command. For a World Trade telegraph terminal, RESETPL forces a Read Initial operation to end if the command being executed is a Prepare and if no characters are being received from the terminal. For a local 3270 display station, RESETPL cancels a Read Initial operation that was scheduled but not initiated.

8. Macros used to assemble translation tables and to translate message text from one code to another.

   - ASMTRTAB - assembles one or more BTAM translation tables with the problem program.

   - TRNSLATE - translates characters from EBCDIC to transmission code or from a transmission code to EBCDIC.

9. Macro used to relinquish control of the central processing unit.

   - TWAIT - relinquishes control of the CPU when the problem program must wait for the completion of one of a number of events before further processing can be done.

10. Macro used to deactivate a communication line group.

    - CLOSE - stops I/O activity on lines or local devices, frees storage obtained by OPEN, and deletes BTAM routines.

BTAM CONTROL BLOCKS

The BTAM control blocks are used to create the telecommunications environment and to interface with the operating system control program. The control blocks are information areas that are related to a particular facet of the operating system. They contain pointers to other blocks, control information for the system, buffer and device characteristics, etc. This section contains introductory information on the control blocks in preparation for the next section, "Role of the Control Blocks in Creating the Access Method." Detailed control block descriptions and formats are contained in the Appendices.

Data Control Block (DCB)

The data control block (created by a DCB macro instruction) serves four functions:

1. Identifies the access method by a combination of data set organization (DSORG) and macro-reference (MACRF).

2. May specify that optional facilities of the access method are to be used for the given line group (for example, buffer allocation, error recording, error recovery procedures).

3. Contains pointers (after OPEN) to BTAM routines and other control blocks.

4. Contains a table of special characters (BSC only).

The DDNAME parameter of a BTAM DCB uniquely identifies a line group. For a start-stop line group, all lines within that line group must be associated with terminals of the same type. For a BSC line group, different types of remote stations may be connected to the same line. The lines, however, may be associated with different control unit types (2701, 2702, or 2703). At least one DD statement must appear in the job stream for each DCB in the problem program. The ddname on the DD card must be the same as the DDNAME in the DCB for the line group. (Refer to Appendix B for a detailed description and the format of the DCB.)

Data Extent Block (DEB)

One data extent block (DEB) is created in supervisor main storage by Open module 1 (IGG0193M) for each line group. The DEB contains a table of IOS appendage addresses, unit control block addresses for each line or local device, and addresses of other control blocks. The DEB also contains a list of the identifications of all BTAM modules needed to support the devices in the line group. (See Appendix C for detailed description and the format of the DEB.)

Input/Output Block (IOB)

The input/output control block (IOB) provides a means of communication between BTAM and the input/output supervisor (IOS). It is the sole parameter of the IOS execute channel program (EXCP) instruction. One IOB is created for each communication line by Open module 3 (IGG0193Q) or for each local 3270 device by the local 3270 Open executor (IGG0194P).

The basic IOB, 64 bytes in length, contains pointers to the channel program, the event control block, and the terminal lists; it provides areas for storing flags, sense bytes, the channel status word, and the start I/O condition code returned by IOS. Appended to each basic IOB is a variable length area where the channel programs are constructed by the Read/Write routine. (See Appendix D for a detailed description and the format of the IOB.)

Unit Control Block (UCB)

A unit control block (UCB) is built for each line or local device at system generation time and is used by IOS during execution to determine the physical device address and to direct it to the appropriate Start I/O and interrupt subroutines. The primary field of the UCB requiring the attention of the BTAM user is the device type word, which gives details of the stations on the line, that is, control unit, adapter, model, and optional features, or details about the local device. The UCB for a local 3270 device includes a graphic devices segment that contains fields used in handling attention interruptions. (See Appendix G for a detailed description and the format of the UCB.)

Data Event Control Block (DECB)

The data event control block (DECB) is formed in the user program at assembly time by expansion of a READ or WRITE macro instruction of the list format (MF=L) or standard format (MF operand omitted). It provides communication with the BTAM Read/Write module by specifying the operation type, line group, line or local device, and terminal list to be associated with the Read or Write operation. The DECB includes areas for the standard event control block and responses to addressing and longitudinal and vertical redundancy checks (LRCs and VRCs). There is a pointer to the current or only entry in the addressing list, a pointer to the current entry in the polling list, and the address of a terminal list or an entry therein. In addition, there are areas which contain information for error recovery. (See Appendix F for a detailed description and the format of the DECB.)

Event Control Block (ECB)

After initiating an I/O operation, the task (user program) in control can continue processing until it needs the results of that operation. At this point, the task issues a WAIT or TWAIT macro instruction, which signals the supervisor that the task cannot proceed until completion of a specified event (i.e., the I/O operation).

The WAIT macro specifies an event control block (ECB) address, which is the first word of the DECB formed by expansion of a READ or WRITE macro instruction of the list or standard format. The TWAIT macro specifies a list of addresses of ECBs representing events awaiting completion. After a WAIT or TWAIT has been issued, the ECB contains the wait flag (set on), the completion flag (set off), and the address of the program request block (PRB). When the event is posted, the wait flag in the ECB (or in one of the several ECBs, for TWAIT) is turned off, the completion flag is turned on, and a completion code is set in byte 0 of the ECB (the completion code byte includes the wait and completion bits); bytes 1-3 are set to zero. (See Appendix E for a detailed description and the format of the ECB.)

Interruption Request Block (IRB)

The interruption request block (IRB) is used to maintain information about an asynchronously executed routine. (See Appendix I for a detailed description and the format of the IRB.)

Interruption Queue Element

The interruption queue element (IQE) is used to schedule an asynchronously executed routine. (See Appendix J for a detailed description and the format of the IQE.)

ROLE OF THE CONTROL BLOCKS IN CREATING THE ACCESS METHOD

At system generation time, a unit control block (UCB) is established for each device in the system. The UCB for a local 3270 device contains information about the device, together with control information required by IOS and BTAM. Because each communication line represents a device, a UCB is established for each line. The UCB for a line contains the 11-bit hardware address through which the line is ultimately accessed, along with other control information required by IOS and BTAM.

A line group is defined through the interaction of the data control block (DCB) and the data definition (DD) statement. In order to be grouped, the following requirements must be met for all lines in the group:

1. All line connections must be either switched or nonswitched.

2. For start-stop lines, all terminals connected to the line must be of the same type; for BSC lines, any type of BSC station that BTAM supports for that type of line may be connected to it. (For multipoint lines, different types of BSC stations can be connected to the line.)

3. All lines share the same buffer pool, if a pool is specified.

4. Dynamic buffering, if specified, is to be used for all lines within the line group.

In the simplest case, one DD statement and one DCB define a data set and, therefore, a line group. (A DCB always defines a line group, by definition.) The relationship between UCBs and the DD statement is as follows:

1. If the problem programmer defines, at system generation time, a unit list representing a number of communication lines or local devices and subsequently refers to that list in the UNIT operand of the DD statement:

   - The line group will consist of the specified lines or local devices.
   - The relative line number of each line or local devices will correspond to its relative position in the unit list based upon ascending order of addresses.
   - The corresponding UCBs will be associated with that DD statement in the same order.

2. The line group can be defined at the time of data set definition, whether or not unit lists were created at system generation time. If the UNIT operand names a specific line or local devices by its 3-digit hexadecimal address, that line or local devices will constitute the line group.

3. The data set may be concatenated. A concatenated data set is represented by a single DCB (and represents a single line group), but is defined through two or more DD statements, only the first of which has a ddname. The DD statements involved may represent any combination of items 1 or 2 above. The relative line numbers of the lines or local devices involved, and the order in which the corresponding UCB addresses are associated with the data set will form a continuous series. This series is determined by the order of the lines or local devices with respect to the DD statements and the order of the DD statements themselves.

At the time that a job begins execution, the System/360 Operating System job scheduler creates a task I/O table (TIOT). The TIOT contains, in addition to other control information, a DD entry for each data definition associated with the task. Each DD entry, in turn, contains a pointer to the UCB for each device associated with the data definition. A DD entry within the TIOT represents, therefore, the UCB addresses in relative line number order for a line group (or for a segment of a line group in the case of a concatenated data set).

When a data set is opened, the BTAM Open executor constructs a data extent block (DEB) that fully defines the line group in terms of the physical devices involved (see Appendix A). Through a series of pointers (see program listing for details), the Open executor accesses the task I/O table. The supervisor open routine passes to the executor the address of the DCB for the data set being opened. Through a field in that DCB, initialized by the supervisor Open routine, the executor finds within the TIOT the DD entry for the data set being opened. (For a concatenated data set, this entry would be the first of two or more applicable DD entries. Additional entries are contiguous with the entry containing the DDNAME, but have a blank-value name field.)

After determining the number of devices involved, which affects the length required for the DEB, the Open executor issues a GETMAIN to obtain the required storage area and initializes the DEB. This initialization includes moving all applicable UCB addresses from the TIOT into the DEB. When this step is complete, the DEB contains the complete set of UCB pointers for the line group, extending contiguously from a known starting location. This makes it possible to obtain the physical address of any device by converting the extent number to an offset value (relative to the beginning of the table of UCB addresses). The DCB for a line group points to the DEB of the line group; the relative line number for a given line or local devices bears a fixed relation to the UCB address of that line or local devices within the DEB. In general, the DEB provides the information required by other components of the control program for I/O operation management.

After constructing the DEB, the BTAM Open executor performs an extensive analysis of the device type information recorded in the UCB. (Only the first UCB associated with the data set is examined because the device type information field must be identical for all UCBs in any one line group.) The contents of the device type information field of the UCB are described in Appendix H. Specification errors in the UCB (such as a conflict between unit type and adapter type) cause an abnormal end of task.

If no specification errors are detected in the UCB, the Open executor begins input/output block (IOB) construction. As with the DEB, the executor computes the amount of main storage required, issues a GETMAIN, and then performs the required initialization. The number of IOBs to be constructed is equal to the "number of extents" value that was previously stored in the DEB. That value is equal to the number of UCB addresses found in the task I/O table (i.e., the number of lines or local devices in the line group).

EVOLUTION OF THE BTAM READ COMMAND FOR REMOTE STATIONS

A problem program must undergo three job steps before becoming active under control of the Operating System: assembly, linkage edit, and execution. The Operating System itself is the result of a system generation. The steps that effect the generation of a channel program by a BTAM Read operation are:

1. System Generation
2. Assembly
3. Execution
   a. Open Line Groups
   b. Read Operations

Figure 1 illustrates the steps providing information for, or modification of, the channel program necessary for the execution of a BTAM Read operation. The linkage editor job step is not included in the figure because it does not affect the channel program.

System Generation

The IOCONTRL and IODEVICE system generation macro instructions provide the details of the transmission control unit (TCU) and remote station equipment (such as adapter type, device type, and features) that the Open executor needs to determine the appropriate device I/O module to be loaded. The information these macros provide is placed in the unit control block (UCB) for each line group. (See Appendix H for the format of UCBs.)

Other system generation macro instructions directly related to BTAM are TELCMLIB, which causes the telecommunications subroutine library, SYS1.TELCMLIB, to be included in the operating system, and DATAMGT, in which the access method is specified as BTAM.

As a result of system generation, the BTAM routines and macro definitions are included in the system library.

Assembly

All BTAM macros are represented by macro definitions in the system library whether they are strictly BTAM macros (e.g., DFTRMLST and CHGNTRY) or specialized parts of system macros used by BTAM (e.g., READ and WRITE).

When the assembler language problem program that will be using BTAM is assembled, each macro instruction embedded in the program is replaced by its appropriate macro expansion. The expansion may consist of assembler instruction statements (e.g., DC and DS), symbolic machine instruction statements (executable machine instructions), or both.

Execution

A BTAM Read operation, during execution of a user program, is shown in Figure 1 to consist of those functions performed during Open and those functions performed during Read.

The data set (communication line group) is usually opened early in the user program. In any case, the data set must be opened before it is referred to by a READ or a WRITE macro instruction. Closing of the data set (CLOSE macro) is not shown in Figure 1. The CLOSE macro is normally placed in the user program so it will be executed when no more references are to be made to the data set.

Defining the data control block (see Appendix B) with the DCB macro and the terminal list with the DFTRMLST macro is normally done in the definition section of the program (along with DC and DS statements) and does not result in the generation of executable machine instructions.

Open Line Groups

Opening, or preparing, a communication line group for use consists of sequentially proceeding through:

1. OPEN macro-expansion coding
2. System Open routine
3. BTAM Open executor (module 1)
4. BTAM Open executor (module 2)
5. BTAM Open executor (module 3)
6. BTAM Open executor (module 4)
7. System Open routine

The OPEN macro-expansion coding is that system OPEN macro coding that is appropriate for a BTAM data set (communication line group). Execution of this expansion coding provides for identifying the specified DCB and for issuing a supervisor call (SVC) for the system Open routine. Parameters such as INOUT, OUTPUT, and INPUT are ignored by BTAM. (BTAM always opens for both input and output.)

The system Open routine partially initializes the specified DCB and requests that BTAM Open module 1 (IGG0193M) be brought into main storage and executed via an XCTL macro instruction. (BTAM coding consists of a number of modules, or control sections, that are loaded and executed as separate but related units.) The four BTAM Open modules are executed serially in the supervisor transient area. They are re-entrant and operate enabled in supervisor mode.

BTAM Open module 1 reserves storage for, and initializes the DEB (Appendix C). Information found in the UCBs created by system generation is used for this purpose. BTAM Open module 2 (IGG0194N) is loaded and given control via an XCTL macro instruction. This module gets main storage for a buffer pool, if a pool is needed, and builds a chain of buffers. If the line group being opened contains BSC devices, this module places the BSC table of special characters in the DCB. BTAM Open module 3 (IGG0193Q) is then loaded and given control by XCTL. Module 3 gets main storage for, and initializes, the IOBs needed (see Appendix D).

Module 4 of BTAM Open (IGG0193S) is given control via XCTL, initializes IQEs, and, using LOAD macros, loads various BTAM routines and appendages: Read/Write routine (IGG019MA), Channel End/Abnormal End Appendage (IGG019MB), the device I/O modules for the particular type of terminal (start-stop) or line configuration (BSC) involved, and (for dynamic buffering only), the PCI appendage (IGG019MC) and the Buffer routine (IGG019MS). The Buffer routine is loaded in all cases where a buffer pool is defined. All of these modules are only loadable and cannot be given control via XCTL macros.

A directory of I/O modules is created within a section of the Read/Write module and, depending on the particular type of terminal or line, SAD and ENABLE commands are issued to the communication lines via the input/output supervisor (IOS). To complete the Open process, control is passed from BTAM Open module 4 to the last load module of system Open and then to the user program at the point immediately following the OPEN macro expansion. The communication lines are then ready for operation.

Read

The control flow for a Read/Write operation without dynamic buffering is shown in Figure 2, that with dynamic buffering, in Figure 3. This is representative of operating a single communication line in the system. Many lines may be operated concurrently by issuing READ macros (or WRITE macros) for each line before waiting for completion of one or more Read or Write operations.

At the point in the user program where a Read operation is desired, the user must perform the following in sequence:

1. Specify, in register 13, the address of a save area for storing the general register values when control is passed to the Read/Write routine (unless register 13 already contains a save area address).

2. Issue a READ macro instruction.

3. Analyze the codes returned by BTAM in register 15 indicating whether the operation was _initiated_ successfully.

   Note: Return codes may be found in the publication _IBM System/360 Operating System: Basic Telecommunications Access Method_, GC30-2004.

4. Issue a WAIT or TWAIT macro instruction at the point beyond which execution is to proceed only after the Read operation is complete.

5. Analyze the completion code and error flags in the data event control block to determine if the operation was completed successfully.

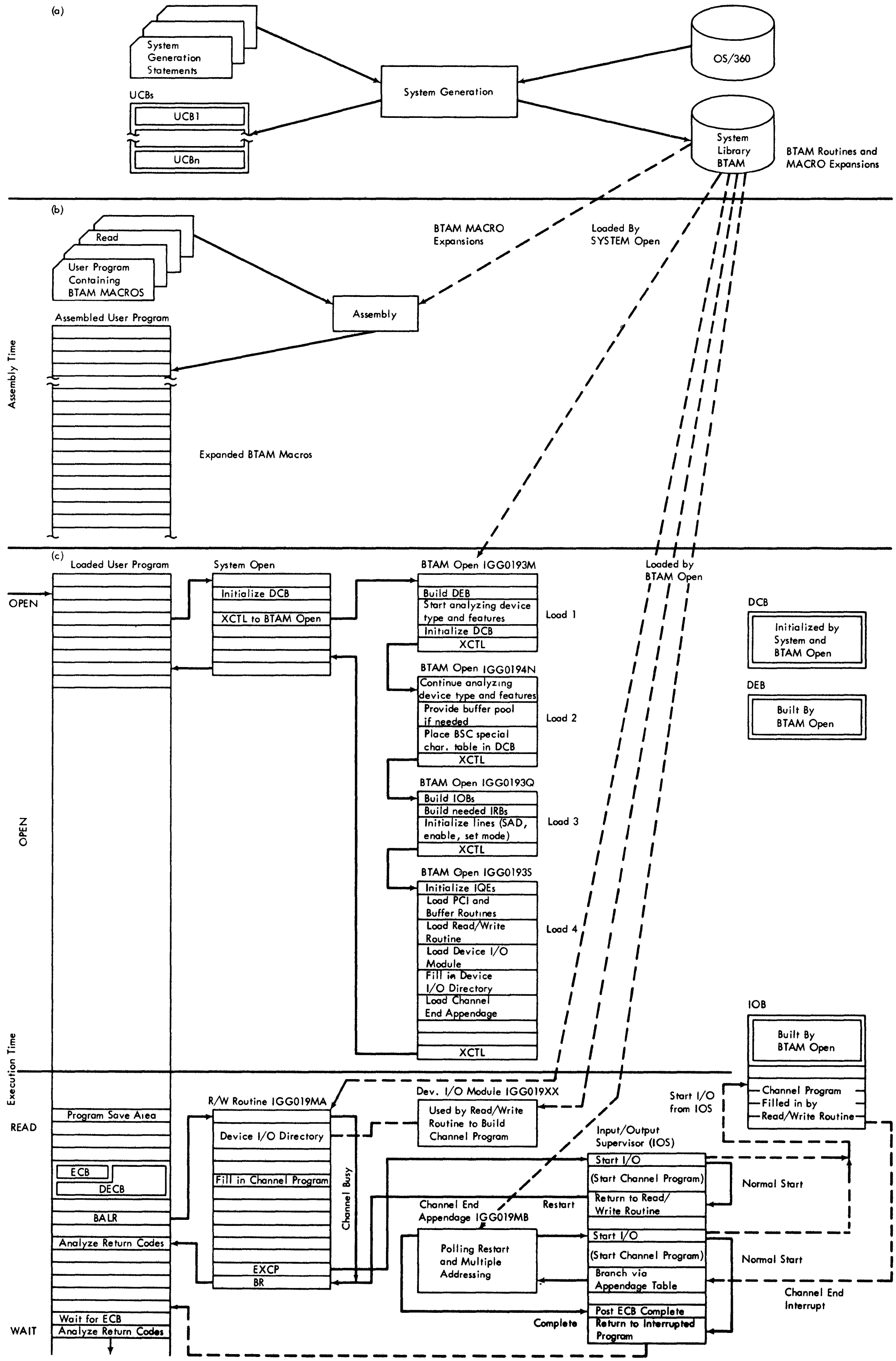


Figure 1. BTAM Read Operation

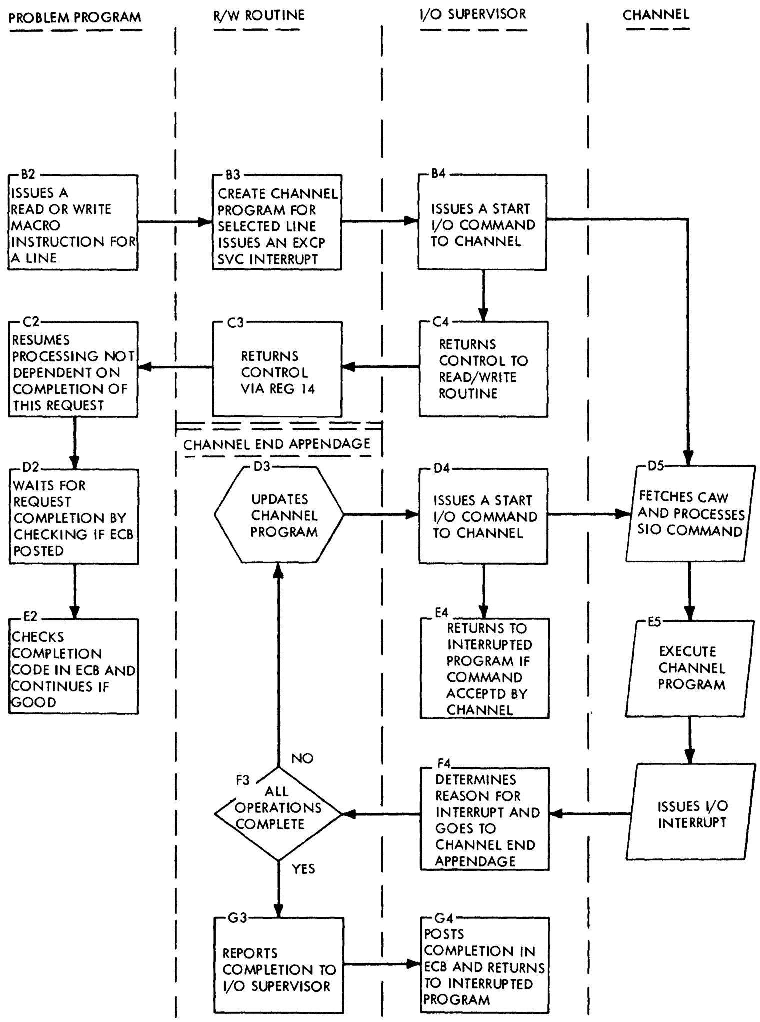


Figure 2. Control Flow for Read/Write Operation With No Dynamic Buffering

The expansion of a READ macro (List or standard form) by the assembler results in initializing a DECB (see Appendix F) with the parameters specified by the READ macro. The DECB is generated or updated by the macro expansion and contains within it an ECB (see Appendix E). The DECB is the only direct communications link between the Basic Telecommunications Access Method (BTAM) and the user program. The ECB within the DECB is the entity upon which a wait is made (i.e., a WAIT macro parameter) and in which completion is posted by IOS when the Read operation is completed.

The principal purpose of the Read/Write routine is to construct the channel program for the particular type of Read or Write desired (i.e., Read Initial, Read Repeat, etc.), making use of the device-dependent information contained in the appropriate device I/O module. Each device I/O module contains, along with special characters and a table of offsets, a number of model channel programs consisting of a sequence of one-byte offsets to the channel command words, one for each type of operation. Each channel program consists of one or more channel command words (CCWs).

If specified in the DCB macro, dynamic buffering (the ability to read variable length records) is achieved by altering the basic channel program. Following the Read Text command are inserted an additional Read Text command and two Read Skip commands. Each Read command uses the PCI appendage to request a buffer through the Buffer routine (IGG019MS), for the next Read command (see Figure 3). If the PCI is queued and the appendage never receives control, the Read command is abnormally completed or completed with error and must be retried. (For a detailed discussion of the dynamic buffering support provided by

BTAM, see the next section, "Dynamic Buffering.")

Just as the DECB is considered the link between the user program and BTAM, the IOB can be considered the link between BTAM and the input/output supervisor (IOS). The channel program built by the Read/Write routine in the IOB field reserved for it has its starting address indicated to IOS in an IOB field.

Once the channel program has been built and passed to IOS for execution, control is returned to the user program, with return codes in register 15 to indicate whether the initiation of the Read was successful. The user program may then proceed with its execution up to a point in the program where a WAIT or TWAIT is specified for completion of the Read operation.

Execution of the channel program proceeds concurrently with that of the user program until a channel end or abnormal end condition is met in the channel program, whereupon IOS regains control and enters the Channel End appendage (IGG019MB). This condition may be caused either by completion of the Read operation or by the need to update and to restart the channel program (e.g., polling the next terminal in the terminal list after a negative polling response).

If the Read operation is completed, IOS posts completion in the ECB for the line, allowing the user program to continue its execution. At this point, the user program should analyze the completion code and flag bits returned by IOS in the ECB (see Appendix E). Further action in the user program should be based on the results of this analysis. This completes the entire Read operation, after which the user program may continue processing or handling of the message obtained from the remote station.

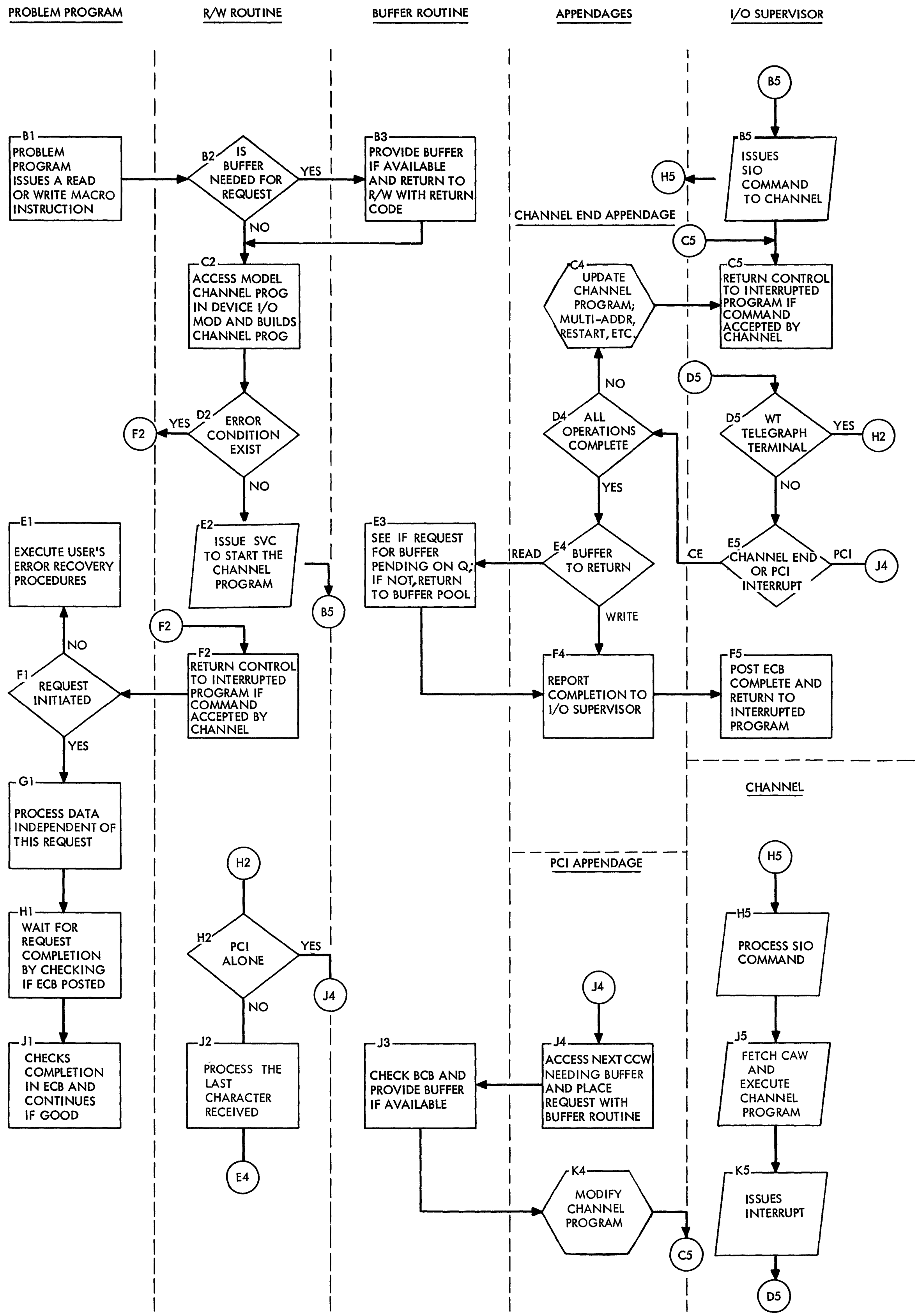


Figure 3. Control Flow for Read/Write Operation With Dynamic Buffering

SYSTEM FLOW FOR LOCAL 3270 OPERATIONS USING BTAM

READ INITIAL OPERATION

Figure 3A shows the system flow for a local 3270 Read Initial operation:

1. The expansion of a READ TI macro instruction issued by the problem program calls the Read/Write routine.
2. The Read/Write routine builds a channel program for the specified device and issues an IECTRDTI macro instruction to test for attentions for the line group. The expansion of IECTRDTI invokes SVC 116 with a routing code of zero, giving control to the SVC FLIH.
3. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at the entry point IECTRDIL.
4. If an attention exists for the line group, the BTAM SVC routine gives a return code of zero; if no attention exists, the Read Initial request is queued, and a return code of four is given. The BTAM SVC routine passes control to the SVC Exit routine.
5. The SVC Exit routine returns control to the Read/Write routine.

6A. If the return code from the BTAM SVC routine was four, the Read/Write routine marks all IOBs for the line group busy and returns control to the problem program.

6B. If the return code was zero, the Read/Write routine initializes the IOB for the device that generated the attention and issues an IECTATNR macro instruction to reset the attention. The expansion of IECTATNR invokes SVC 116 with a routing code of one, giving control to the SVC FLIH.

7. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at entry point IECTATRL.
8. The BTAM SVC routine resets the attention and passes control to the SVC Exit routine.
9. The SVC Exit routine returns control to the Read/Write routine.
10. The Read/Write routine invokes SVC 0 to have the I/O operation scheduled, giving control to the SVC FLIH.
11. The I/O operation is scheduled by the EXCP routine, and the SVC exit routine returns control to the Read/Write routine.
12. The Read/Write routine returns control to the problem program.
13. When the Read Initial operation completes, an I/O interruption without attention status occurs, and control is given to the IOS interruption supervisor through a PSW swap.
14. IOS gives control to the local 3270 channel end/abnormal end appendage to do processing peculiar to the device or access method.
15. The appendage returns control to IOS.
16. When IOS has finished processing, it gives control to the I/O FLIH.
17. The I/O FLIH passes control to the task dispatcher, which dispatches the highest priority ready task in the system.

READ AND WRITE OPERATIONS

Figure 3B shows the system flow for local 3270 Read and Write operations (other than Read Initial):

1. The expansion of a READ or WRITE macro instruction issued by the problem program calls the Read/Write routine.
2. The Read/Write routine builds a channel program and initializes an IOB for the specified device. The Read/Write routine issues an IECTATNR macro instruction to reset the attention. The expansion of IECTATNR invokes SVC 116 with a routine code of one, giving control to the SVC FLIH.
3. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at the entry point IECTATRL.
4. The BTAM SVC routine resets the attention and passes control to the SVC Exit routine.
5. The SVC Exit routine returns control the Read/Write routine.
6. The Read/Write routine invokes SVC 0 to have the I/O operation scheduled, giving control to the SVC FLIH.
7. The I/O operation is scheduled by the EXCP routine, and the SVC Exit routine returns control to the Read/Write routine.
8. The Read/Write routine returns control to the problem program.
9. When the Read or Write operation completes, an I/O interruption without attention status occurs, and control is given to the IOS interruption supervisor through a PSW swap.
10. IOS gives control to the local 3270 channel end/abnormal end appendage to do processing peculiar to the device or access method.
11. The appendage returns control to IOS.

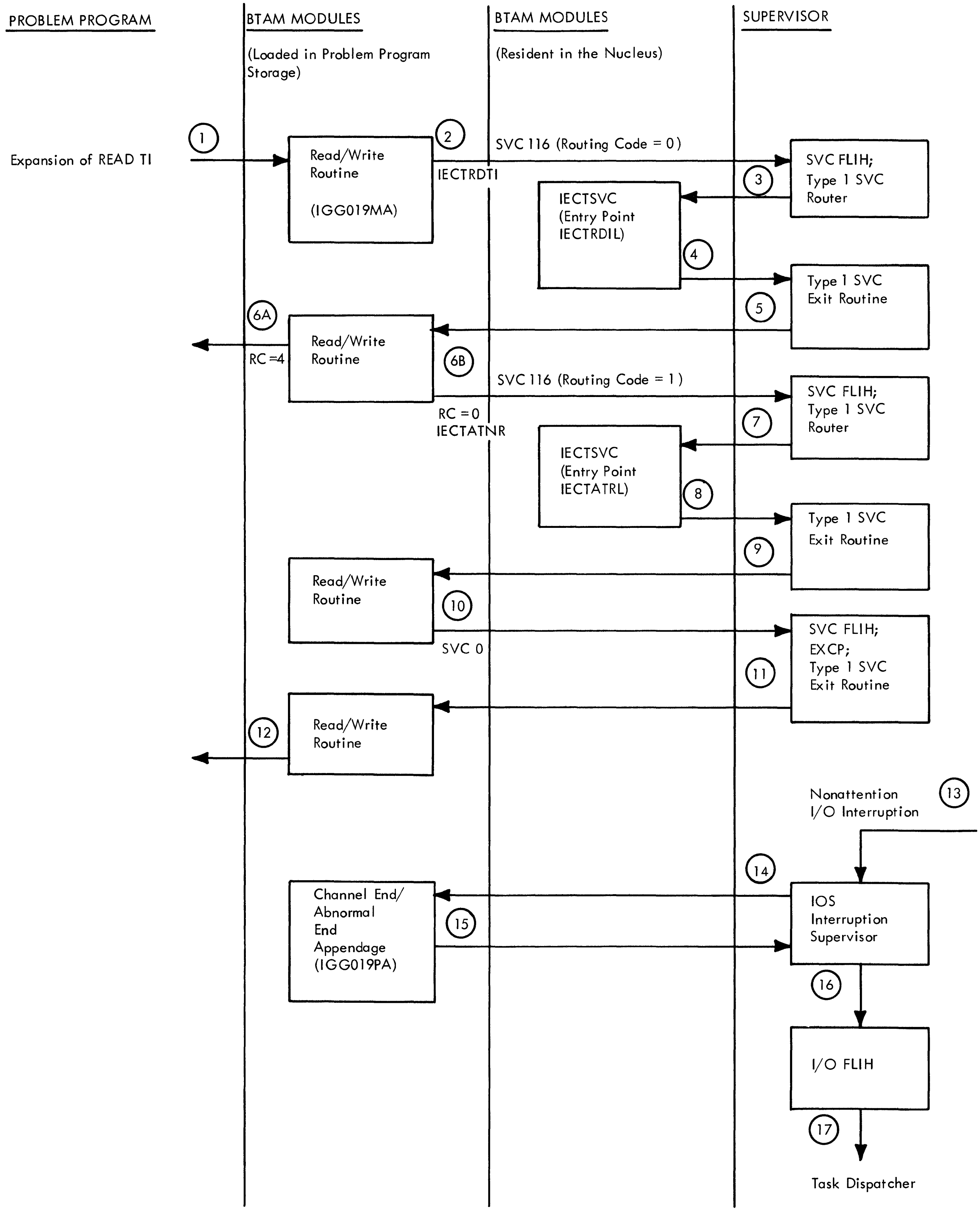


Figure 3A. System Flow for Local 3270 Read Initial Operation

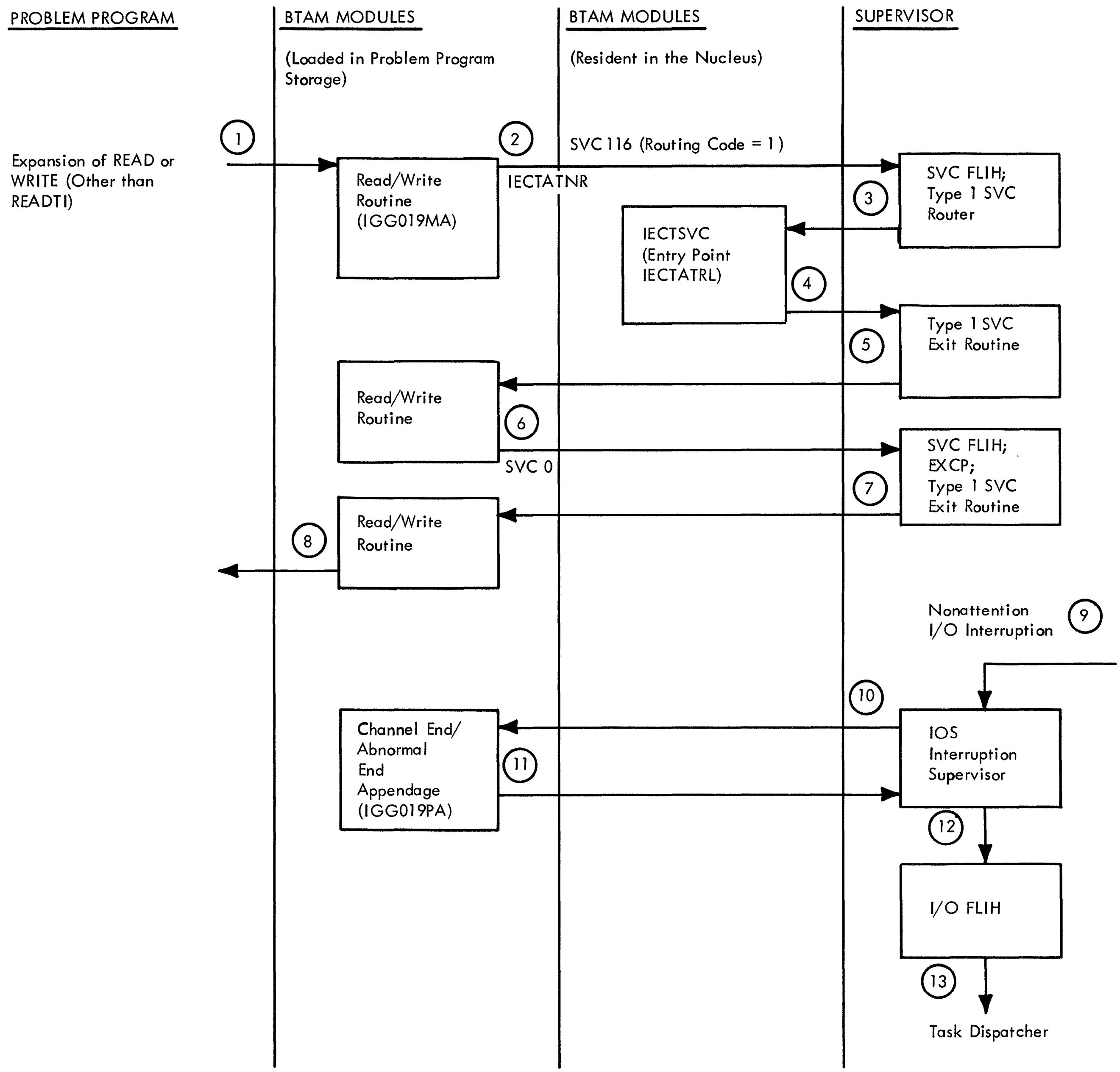


Figure 3B. System Flow for Local 3270 Read and Write Operations (Other than Read Initial)

12. When IOS has finished processing, it gives control to the I/O FLIH.
13. The I/O FLIH passes control to the task dispatcher, which dispatches the highest priority ready task in the system.

CHGNTRY SKIP OPERATION

Figure 3C shows the system flow for a local 3270 CHGNTRY Skip operation:

1. The expansion of a CHGNTRY macro instruction (with SKIP specified) issued by the problem program invokes SVC 116 with a routine code of two, giving control to the SVC FLIH.
2. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at entry point IECTCHSL.
3. The BTAM SVC routine turns on the skip flag for the specified device and passes control to the SVC Exit routine.
4. The SVC Exit routine returns control to the problem program.

CHGNTRY ACTIVATE OPERATION

Figure 3D shows the system flow for a local 3270 CHGNTRY Activate operation:

1. The expansion of a CHGNTRY macro instruction (with ACTIVATE specified) issued by the problem program invokes SVC 116 with a routine code of three, giving control to the SVC FLIH.
2. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at entry point IECTCHAL to turn off the skip flag for the device.
3A. If the activate request satisfies a queued read initial request, the BTAM SVC routine enters the asynchronous exit effector stage 2 (AEE2) to schedule the local 3270 second-level attention routine.
3B. The BTAM SVC routine passes control to the SVC Exit routine.
4A. If AEE2 was not entered, the SVC Exit routine returns control to the problem program.
4B. If AEE2 was entered, the SVC Exit routine gives control to the asynchronous exit effector stage 3 (AEE3).
5. AEE3 calls the second-level attention routine.
6. The second-level attention routine initializes the proper IOB and invokes SVC 0 to have the I/O operation scheduled, giving control to the SVC FLIH.
7. The I/O operation is scheduled by the EXCP routine, and the SVC Exit routine returns control to the second-level attention routine.
8. The second-level attention routine returns control to AEE3.
9. AEE3 returns control to the problem program.

RESETPL OPERATION

Figure 3E shows the system flow for a local 3270 RESETPL operation:

1A. The problem program issues a RESETPL macro instruction. If no Read Initial request is queued, the expansion of RESETPL returns control to the problem program.
1B. The problem program issues a RESETPL macro instruction. If a Read Initial request is queued, the expansion of RESETPL invokes SVC 116 with a routing code of four, giving control to the SVC FLIH.
2. Through the SVC router, control passes to the local 3270 BTAM SVC routine (IECTSVC) at entry point IECTRSTL.
3. The BTAM SVC routine dequeues the Read Initial request and enters the POST routine to post the ECB for the Read Intitial operation.
4. The BTAM SVC routine passes control to the SVC Exit Routine.
5. The SVC Exit routine returns control to the expansion of the RESETPL macro instruction.
6. The expansion of RESETPL returns control to the problem program.

ATTENTION INTERRUPTION HANDLING

Figure 3F shows the system flow for local 3270 attention interruption handling:

1. When an I/O interruption with attention status occurs, control is given to the IOS interruption supervisor through a PSW swap.
2. IOS gives control to the local 3270 first-level attention routine to examine the status of the device that caused the attention interruption.
3. If the attention interruption satisfies a queued Read Initial request, the first-level attention routine enters the asynchronous exit effector stage 2 (AEE2) to schedule the local 3270 second-level attention routine.
4. The first-level attention routine returns control to IOS.

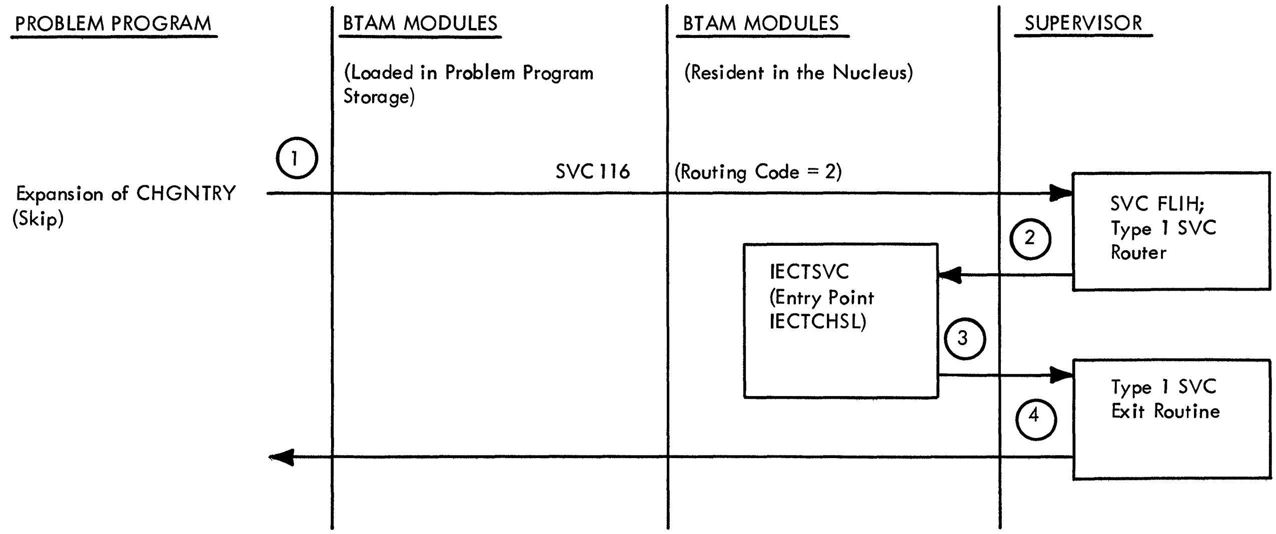


Figure 3C. System Flow for Local 3270 CHGNTRY Skip

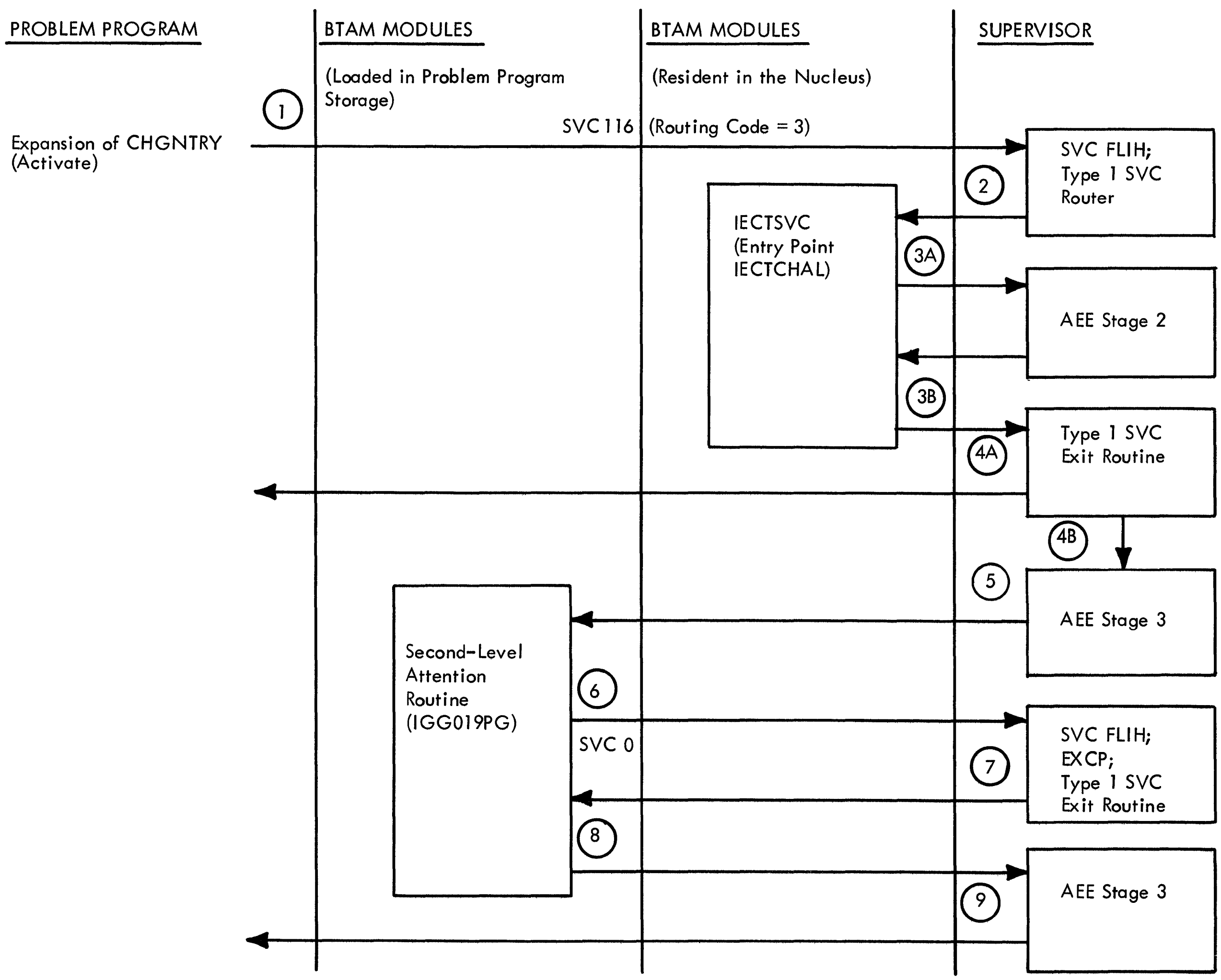


Figure 3D. System Flow for Local 3270 CHGNTRY Activate

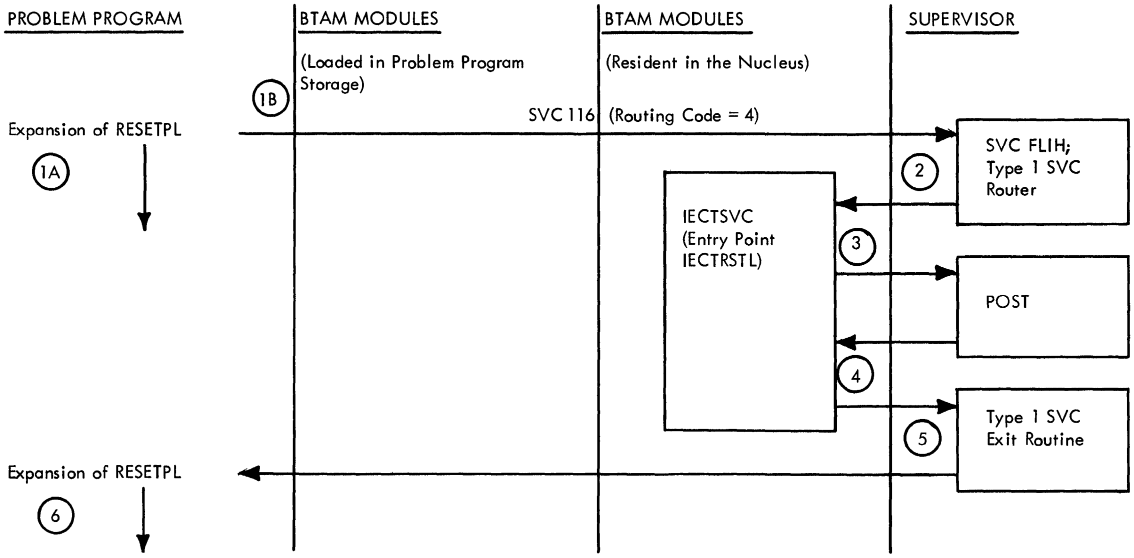


Figure 3E. System Flow for Local 3270 RESETPL

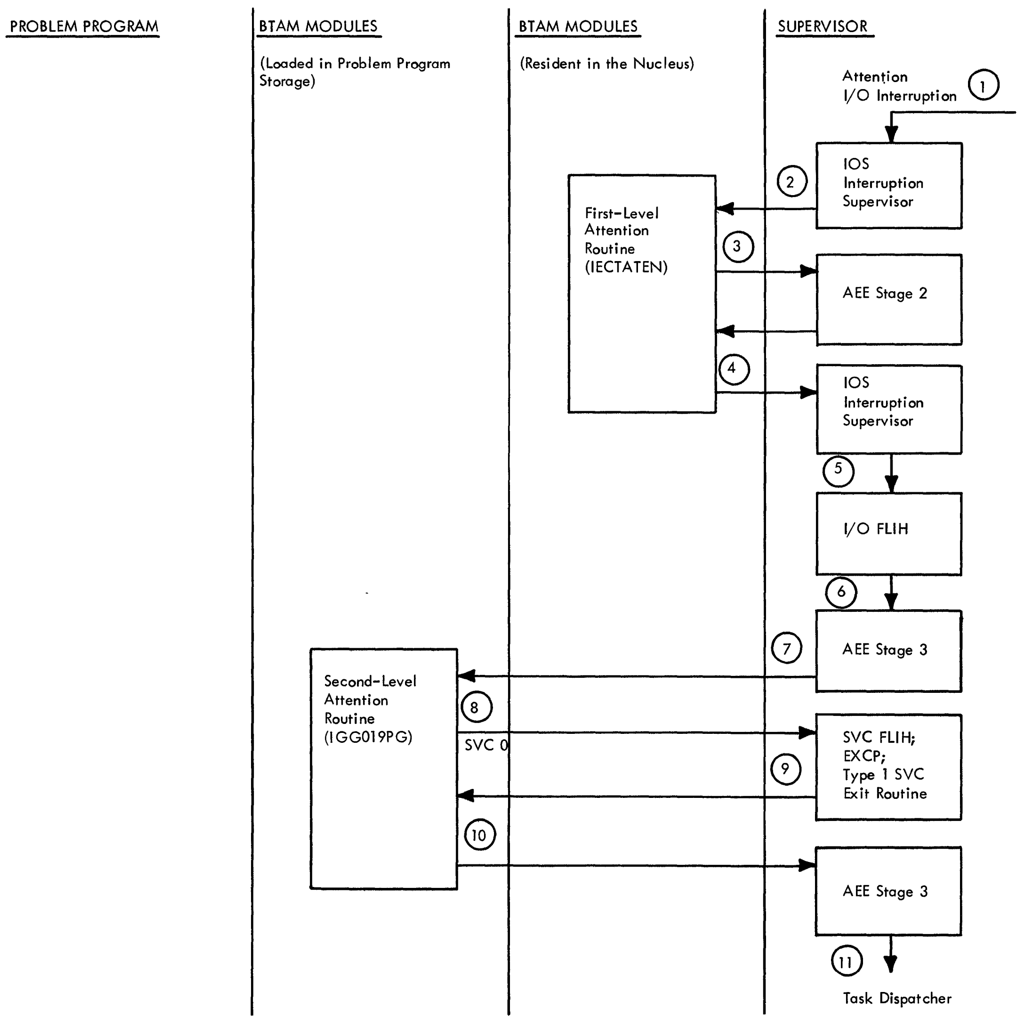


Figure 3F. System Flow for Local 3270 Attention Interruption Handling

5. When IOS has finished processing, it gives control to the I/O FLIH.
6. If AEE2 was entered, the I/O FLIH gives control to the asynchronous exit effector stage 3 (AEE3).
7. AEE3 calls the second-level attention routine.
8. The second-level attention routine initializes the proper IOB and invokes SVC 0 to have the I/O operation scheduled, giving control to the SVC FLIH.
9. The I/O operation is scheduled by the EXCP routine, and the SVC Exit routine returns control to the second-level attention routine.
10. When the second-level attention routine has finished processing, it returns control to AEE3.
11. AEE3 passes control to the task dispatcher, which dispatches the highest priority ready task in the system.

DYNAMIC BUFFERING

The dynamic buffering facility, which provides the ability to receive or transmit a variable amount of data on a Read or Write operation, is provided by BTAM only when BFTEK=D is specified in the DCB macro instruction for the line or line group. When it is specified, BTAM loads two additional modules: the Buffer routine (IGG019MS) and the PCI appendage (IGG0 19MC).

There are two levels of support. In the first, the user requests one or more buffers via the REQBUF macro instruction and returns one or more buffers to the buffer pool via the RFLBUF macro instruction (see Buffer routine discussion in the next chapter). In this instance the user may avoid tying up a full buffer on a Read operation until a positive response to polling is received.

In the second level, the user specifies that BTAM is to supply buffers, including the first one, as needed to accommodate the received data. (This is applicable only for a Read operation.) The user specifies this by coding 'S' as the area operand in the READ macro instruction.

After the user has requested BTAM to supply the first buffer or has supplied the first buffer himself, BTAM's buffer management will continue to provide or return the buffers until data transmission is complete. With dynamic buffering, three additional CCWs are generated following the Read Text CCW (if Read operation) or Write Text CCW (if Write operation). The channel executes this loop of commands until data transmission has been completed.

Fetching of a buffer management CCW by the channel causes a program controlled interrupt (PCI) which, under normal conditions, causes another buffer to be supplied (see Program Controlled Interrupt (PCI) Appendage discussion). Thus, while the channel is reading into (or writing from) buffer N, buffer N+1 is being supplied by BTAM.

If a buffer is not immediately available on a Read operation, a request for a buffer is placed in the buffer request queue. If the request is not filled before data is received for this buffer, the reception of this data into main storage is discontinued by a Read skip command. If the buffer is supplied in time, the PCI appendage changes the Read Skip command to a TIC so that message text enters the next buffer.

For a Write operation, the circumstance of a buffer being unavailable does not occur, because the user has necessarily placed the data in a chain of buffers before initiating the Write operation.

See Figure 4 for the logic flow for establishing a dynamic-buffering channel program.

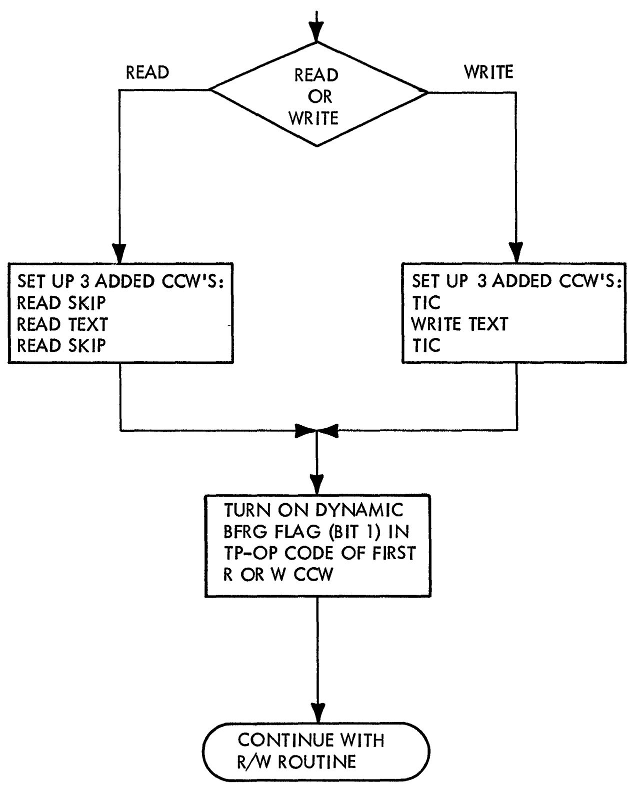


Figure 4. Logic Flow for Dynamic Buffering

READ INITIAL CHANNEL PROGRAMS USING AUTO POLL (MULTIPOINT ONLY)

On a Read Initial operation that uses the Auto Poll facility (whether start-stop or BSC), the Read/Write routine must make a number of decisions in order to set up the appropriate channel program. These decisions are based principally on:

- Whether the READ macro specified the name of a polling list entry, or specified 'S', indicating that BTAM is to poll the last-polled entry;
- Whether the list is of the open or wraparound format;
- Whether BTAM knows which entry was last polled.

In order to set up the Poll commands, the Read/Write routine must have available the address of some entry in a polling list, and must have available, or be able to compute from the address of the entry, the address of the polling list header (this address is three less than the address of the first entry in the list). The DECB fields DECPOLPT and DECENTRY are used in the computation, as follows.

If the user specified the entry operand of the READ macro as 'S', he wishes for BTAM to begin polling with the last-polled entry. The index byte for this entry, and the address of the first entry in the list, are contained in the DECPOLPT field, unless the polling operation being set up is the first polling operation using this list. The Read/Write routine therefore checks DECPOLPT to see if an index and an address are present. If so, the routine saves the index value and subtracts the header length from the address of the first entry to obtain the address of the list header.

If, however, the user specified a particular entry, rather than 'S', (the routine checks the DECTYPE field of the DECB to determine this), or if he specified 'S' but DECPOLPT does not contain an index and address, the Read/Write routine obtains the address of the entry from the DECENTRY field. Then the routine scans forward through the list containing that entry until it encounters a scan stop byte (which by definition contains X'FE'), signifying the end of the list. From the address of this byte and the offset value that follows it, the routine computes the address of the list header.

At this point the routine has the address of the header and the index of some entry in the polling list. Before using these values to set up the Poll commands, the routine determines whether the Read Initial operation is to be executed by the On-Line Test or Error Recovery facilities of BTAM. If not, the routine determines whether all entries in the list are skipped or if the polling list usage count has reached its limit of 15. (This count indicates how many Read operations are currently using the list.) If neither of these conditions prevails, the routine increments the usage count by one and proceeds; if either condition is true, the channel program cannot be executed and control is returned to the user program with a return code of X'10' in register 15.

If the operation is being executed under On-Line Test or Error Recovery procedures, the routine bypasses the step of checking for active entries and usage count, and does not increment the usage count.

Now, the Read/Write routine places in DECPOLPT the address of the first entry and the index of the last-polled or user-specified entry, and proceeds to complete setup of the Read Initial channel program.

If the polling list specified by the READ macro is of the wraparound format (the routine checks the W bit in the list header to determine this), the routine sets up, following the Poll command, two TIC commands and a second Poll command, and places in the second Poll command the address of the first entry (which is contained in DECPOLPT). This first entry is where polling resumes after the first pass through the polling list by the first Poll command.

The routine then places in the first Poll command (or the only Poll command, for a list of the open format), the address of the entry with which polling is to begin. This entry will be the one whose index is contained in DECPOLPT (i.e., the last-polled or user-specified entry), unless that entry is no longer active. The Read/Write routine steps through the polling list from the first entry, inspecting each index until it finds one that equals the index in DECPOLPT, or the first one whose value exceeds the index in DECPOLPT.

The routine then places in the Poll command the address of the entry corresponding to the index found in this manner. If, however, there are no active entries whose indexes equal or exceed the index in DECPOLPT, the routine places the address of the first active entry in the Poll command.

BTAM ROUTINES AND ASSOCIATED MODULES

BTAM consists of 108 load modules that support 21 routines (considering as single routines all error recovery routines and all on-line test routines). Of these modules, 74 are executable and 34 are device I/O modules containing no executable code.

A complete list of BTAM modules is given in Appendix I. For convenience, they are listed both by module name, in alphabetical order, and by name of routine.

This chapter discusses each of the routines and the means by which BTAM channel programs are generated.

BTAM Routines

The BTAM routines (and associated module names) discussed in this section are:

- BTAM Open Executor (IGG0193M, IGG0194N, IGG0193Q, and IGG0193S)
- Local 3270 BTAM Open Executor (IGG0194P and IGG0194Q)
- BTAM Read/Write Routine (IGG019MA)
- Channel End/Abnormal End Appendage (IGG019MB)
- World Trade Telegraph Terminal Channel End/Abnormal End Appendage (IGG019PD)
- Local 3270 Channel End/Abnormal End Appendage (IGG019PA)
- Program Controlled Interrupt (PCI) Appendage (IGG019MC)
- Local 3270 First-Level Attention Routine (IECTATEN)
- Local 3270 Second-Level Attention Routine (IGG109PG)
- Local 3270 BTAM SVC Routine (IECTSVC)
- Error Recovery Procedures (see Appendix I for module names)
- Buffer Routine (IGG019MS)
- Change Entry Routine (IECTCHGN)[1]
- Edit Routine (IECTEDIT)
- Line Error Print Routine (IECTLERP)
- Line Open Routine (IECTLOPN)
- Translate Routine (IECTTRNS)
- On-Line Testing (see Appendix I for module names)
- REQBUF/RELBUF (SVC 58) Routine (IGC058)
- 2741 Break Routine (IGG019PK)
- BTAM Close Executor (IGG0203M)

----

[1]The Change Entry routine is present only if Auto Poll is used.

BTAM Open Executor (IGG0193M, IGG0194N, IGG0193O, and IGG0193S)

Expansion of the OPEN macro instruction in the problem program generates a calling sequence containing a supervisor call (SVC). Execution of this SVC causes the first load of system Open to be loaded into the supervisor transient area.

After performing open functions required by the system, the supervisor Open routine passes control to the BTAM Open executor when the DCB for the data set being opened indicates that BTAM is the access method to be employed.

The Open executor creates the environment under which BTAM operates, based upon information provided through a user-defined data control block (DCB) for a line or line group. (An executor is defined as the set of instructions that is entered by the supervisor Open routine to perform the functions peculiar to an access method.) The routine consists of four load modules. The first load module (IGG0193M) obtains the storage required for the data extent block (DEB), initializes the area in the DEB, and fills in the addresses of the UCBs into the DEB (see Appendix C for the format of the DEB). If the first UCB for the line group is for a graphics device (that is, byte 3 in the UCBTYP field is X'10'), control is passed to the where-to-go subroutine, which transfers control to the first load module (IGG0194P) of the local 3270 BTAM Open executor. Otherwise, the UCBs are checked in connection with the DCB being opened for certain invalid conditions which will terminate the program with system codes. The system code and the invalid conditions checked by load 1 are:

090 A device other than a telecommunications device was specified in the UCB allocated to the DCB being opened.

091 An invalid or unsupported transmission control unit type was specified in the UCB.

092 An invalid or unsupported terminal adapter type was specified in the UCB.

093 An invalid or unsupported terminal type was specified in the UCB.

094 An invalid or unsupported optional feature was specified in the UCB.

095 The lines allocated to the line group were not identical with respect to the terminal type and optional features.

096 There was no buffer pool or no buffering technique defined for the line group.

097 The current Open requires an additional entry in the Device I/O Directory, which is already full; i.e., the maximum number of different device types have already been opened since the last IPL (normal maximum is 16).

098 Dual Communication Interface B or Dual Code Feature B was specified and the transmission control unit was not a 2701 or the Dual Communication Interface or the Dual Code Feature was not specified in the UCB.

Note: For a completion code of 097, additional space in the Device I/O Directory can be provided by changing one statement in the Read/Write routine. The statement following the one labeled IOD must be changed. The duplication factor must be increased to correspond to the number of different device types being used. The new Read/Write routine must be reassembled and link-edited to SVCLIE with RENT,NE,DC as parameters. The IEHIOSUP program must then be executed. See the System Generation manual, GC28-6554, for the procedure in executing this program.

If load 1 has not terminated the task, it transfers control to load 2 or to the local 3270 BTAM Open executor by XCTL. (See Charts 01 and 01A for logic flow of the first load module).

The second load module, IGG0194N, checks the DCB to determine whether the user defined a buffer pool or specified dynamic buffering. If the pool is defined, the buffer length is obtained and placed in the DCB. If the user specified both buffer length and the number of buffers (BUFL and BUFNO), the module obtains the required amount of main storage. If the user has omitted from the DCB both buffer pool information and the BFTEK=D operand, the job ends with a system code of 096. Otherwise, this module gives control to the third load module. (See Chart 01B for the logic flow of IGG0194N.)

The third load module (IGG0193Q) gets main storage for the input/output blocks (IOBs) required for BTAM internal use and for communication with other components of the operating system. It then initializes the lines by executing the appropriate Set Address, Enable, or Set Mode command. (See Chart 02 and 03 for logic flow of the third load module.)

The fourth load module (IGG0193S) completes the initialization process by loading into main storage the Read/Write routine, the channel end/abnormal end appendage, the WT telegraph terminal channel end/abnormal end appendage (if the device is a World Trade telegraph terminal), and a device I/O module required for channel program generation. When establishing the device I/O module, the third load makes the corresponding entry in the device I/O directory. (See Charts 04 and 05 for logic flow of the fourth load module.)

The fourth load module checks for certain options to load other modules, if necessary. One optional function is the creation of a buffer pool that permits the user to obtain and release buffers through the REQBUF and RELBUF macro instructions. The buffer pool is created if the user has provided the appropriate values in the BUFNO and BUFL fields of the data control block. (See IBM System/360 Operating System: Basic Telecommunications Access Method, GC30-2004). If the buffer routine (IGG019MS) is required it is loaded; the program controlled interrupt (PCI) appendage (IGG019MC) is also loaded when dynamic buffering is specified.

Other optional functions include setting up for On-Line Test and Error Recovery Procedures if they are required.

Control flow during Open is governed by a standard transfer-control subroutine, which appears in all load modules of the BTAM Open executor. The subroutine transfers control between the BTAM executors and other Open executors. Once control has been passed to the BTAM modules, all BTAM DCBs in the table constructed by the supervisor Open routine are processed by that module; however, one or more other executors may be in control between the BTAM modules.

Local 3270 BTAM Open Executor (IGG0194P and IGG0194Q)

The local 3270 BTAM Cpen executor receives control from the first load module (IGG1093M) of the BTAM Open executor if the UCE for the first device in the line group is a graphics UCB. The local 3270 Cpen executor creates the environment under which BTAM operates, based on information in the DCB for the line group of local 3270 devices.

The Open executor has twc lcad modules. The first load module (IGG0194P):

- Checks the UCBs of all devices being opened to see whether the devices are local 3270 devices. If one or mcre devices are not local 3270 devices, the prcblem program is terminated with a system code of X'095'.
- Gets main storage for one IOB for each device in the line group and initializes the IOBs.
- Gets main storage for IRB pcinters if cn-line testing is specified.
- Tries to initialize each device in the line group by means of a Write/Erase operation that erases the screen, restores the keyboard, and resets the modified data tags. If this operation fails for a device, the system operator is notified, and he decides whether to continue trying to initialize the device or to try to initialize the other devices in the line group. (If OLTEP is using a device teing opened, the Open executor does not try to initialize the device; instead, it sets a flag in the IOB (bit 0 in byte 0 of IOBINCAM) and proceeds as though a permanent I/O error occurred.)
- Transfers contrcl to the second load mcdule cf the Open executor.

For the logic flow of the first laod module (IGG0194P), see Charts C5B, 05C, and 05D.

The second load module (IGG0194Q):

- Checks the DCB to see whether a buffer pool is defined or requested. If a buffer pool is defined (that is, BUFL and BUFCB were specified), the buffer length is placed into the DCB. If a tuffer pool is requested (that is, BUFL and BUFNO were specified), main storage is obtained, and the buffer pool is constructed. If dynamic buffering is requested, the request is ignored.
- Loads the on-line test ccntrol module (IGG019PI) and builds an IRB and IQE if the module and control blocks are needed.
- Lcads the tuffer routine (IGG019MS) if it is required.
- Loads the local 3270 channel end/abnormal end appendage (IGG019PA) and places its address into the DEBCEA and DEBXCEA fields in the appendage table in the DEB.
- Loads the local 3270 second-level attention routine (IGG109PG) and builds an IRB and IQE.
- Initializes the fields in the graphic devices segment of the UCB.
- Lcads the Read/Write routine (IGG019MA) and places its address into the DCB.
- Searches the device I/C director in the Read/Write routine. If the lccal 3270 I/O module (IGG019PH) has already been loaded, the displacement of its entry in the directory is placed into the DCBDEVTP field in the DCB. If the local 3270 I/O module has not teen loaded and the device I/O directory is full, the problem program is terminated with a system ccde of X'097'. Otherwise, the local 3270 I/O module is loaded, the code frcm the DCBDEVTP field and the address cf the I/O mcdule are placed into an entry in the device I/O directory, and the displacement of the entry is placed into the DCBDEVTP field.
- Clears the entry in the where-to-go (WTG) table to show that Open processing for the DCB is complete.

For the lcgic flow of the second load module (IGG0194Q), see Charts 05E and C5F.

The WTG table routine, a subroutine that appears in all load modules of the Open executors, controls flow during open by transferring control among the BTAM Open executor, the local 3270 BTAM Open executor, and other Open executors.

BTAM Read/Write Routine (IGG019MA)

The Read/Write routine is entered from a problem program using BTAM through the linkage generated by expansion of either the READ or WRITE macro instruction or from Error Recovery Procedure or On-line Test routines. (See Charts A1 through A4 for the lcgic flow of IGG019MA.) The Read/Write routine acts as an intermediary between the calling routine and the input/output supervisor (IOS). The Read/Write routine performs the following functions:

- Selects the ICB for the line specified by the calling routine.
- Accesses the device I/O module; computes, if necessary, the area address and length; sets up, if necessary, additional CCWs (as for Auto Poll and dynamic buffering); and loops on the CCW count until all CCWs have been moved and completed in the channel program area of the IOB for the line.
- For a local 3270 Read Initial request, passes control (by issuing IECTRDTI, and internal macro instruction that invokes SVC 116 with a routing code of zero) to the local 3270 BTAM SVC routine (IECTSVC), which checks for an attention interruption and queues the Read Initial request if none exists.
- For any local 3270 Read or Write request, passes control (by issuing IECTATNR, an internal macro instruction that invokes SVC 116 with a routing code of one) to the local 3270 BTAM SVC routine (IECTSVC), which resets the attention flag and updates the attention count.
- Issues an execute channel program SVC (EXCP), passing control to the I/O supervisor with the address of the IOB as a parameter. Before giving an EXCP macro instruction to start the channel program, BTAM tests whether the program employs a terminal list of the SSALST or SSAWLST form. If so, the usage count field (indication of the total number of polling operations using the terminal list at any one time) of the terminal list is incremented by 1.

The CCWs in the device I/O modules are complete except for the area address and count fields. An index in bytes 1 and 6 of the CCW determines which subroutine is branched to to complete either the area address or the count. If an offset to the normal address is required, this value already exists in the CCW.

Read/Write Subroutines to Compute Area Address

The subroutines for computing the area address are:

Index Value (hex)	Subroutine
00	TESTLNG - If the area address index byte is 00, the Read/Write routine then goes directly to compute the length.
04	DATAREA - The fourth byte of the CCW is added to the address of the area. The area address is the DECAREA field in the DECB for this Read or Write operation. This subroutine computes the area address for a Read or Write text or Read Response CCW. If there is a Read Response CCW, then it will read into the first two bytes of the area, and the Read text CCW which follows will read into the original area address plus two.
08	RESPAREA - The address of the response field in the DECB (DECRESPN) is loaded into the area field of the CCW to read the response to addressing or to text.
0C	SPECCHAR - The address of the special character field is loaded in the area field of the CCW before going to TESTLAST.
10	LIST - The fourth byte of the CCW, which is a one-byte offset, is added to the address of the terminal table entry that contains the dial characters. A one-byte offset is used to bypass the one-byte number which is the number of dial characters. The number of dial characters is moved to the count field. This sets up the field to dial a station on a switched line.
14	PALIST - The DECTYPE field in the DECB is tested to see if the user specified the 'S' operand. If so, the offset in the CCW is picked up to load the polling or addressing pointer, if necessary. The count of characters is added to the station address. The subroutine finds the first nonskipped polling or addressing entry and places its address in the CCW.
18	TWXIDENT - The number of dial characters plus one is added to the list address and then the number of ID characters is moved into the count field. The address of the ID characters is placed in the area address field of the CCW.
1C	PA1050D - The address of the addressing characters in the 1050 Dial list is placed in the area address field.
20	DISABLE - The entry is checked to see if it is in an answering list. If it is, then an Enable CCW is set up instead of a Dial CCW in the channel program area.
24	AUTOPOLL - Barring exceptional conditions, this subroutine sets up the Poll and NOP or TIC commands of the channel programs used with Auto

Poll lists (open or wraparound). It also increments the usage count for the lists.

Exceptional conditions (the appropriate register 15 return code is set):

1. All entries are skipped.
2. The usage count is at its maximum value.

28 BSCCHARS - The fourth byte of the CCW, which is the relative location of the control characters within the DCB, is added to the base address of the control characters. The resultant address which is the location of the control characters, is stored in the area field of the DCB.

2C BSCDIAL - The first byte of the terminal list built by DFTRMLST is tested to see if the list is an answering list. If so, an Enable CCW is moved into the channel program to replace the Dial command and the CCW counter is incremented past the Write Inquiry, Read Response, and Write EOT commands. If the list is a dial list (calling list), the dial count is stored in the count field of the CCW. The fourth byte of the CCW is added to the address of the dial characters and stored in the address field of the CCW.

30 BSCIDENT - The dial character count is added to the list address and used as a base to obtain the number of characters in the remote device's identification sequence. Twice this number plus an offset of 1 is added to the base and used as an address to obtain the number of characters to be transmitted. The number of characters to be transmitted is added to the fourth byte of the CCW and stored in the address field of the CCW.

34 BSCRESP - The DECSNDPT field (1 byte) is added to the address of the DCB and stored in the area field of the CCW. This allows the proper response to be transmitted from that address.

38 BSCWAREA - The fourth byte of the CCW is added to the address of the write area and stored in the address field of the CCW.

3C BSCGRAPH - Places in a Write graphics CCW the address of graphic characters, for BSC Write Leading Graphics operations.

40 BSCADRS - Places in a Read Response CCW the address of the area specified in the DECRESPN field of the DECB and sets the command chain flag in the CCW, if necessary.

44 WTTAWRU - If the operation is a Read Initial with ID Exchange, the address of a WRU byte, if WRU is coded YES in the DCB, is placed in the area address field. If WRU is coded NO or is absent, the address is that of a letters shift character. To construct the Write CPU ID and Read Terminal ID CCWs, the CPU and terminal identification addresses are found as defined by the DFTRMLST macro instruction.

If the operation is a Write Initial or Write Continue, the address of the Pad characters is placed in the area address field, and the count is placed in the count field of the CCW.

48 WTTASNS - When the operation is a Read Initial, the response field byte of the DECB is set to X'FF'.

4C WRITTONE - Places in the CCW the address of a user-specified data tone (for BSC manual calling operation).

50 CNTCHARS - Places in a Write CCW that sends frame change characters to a 2760 the address specified in the DECENTRY field of the DECB. This subroutine is used for Write TIO or Write TCO operations for a 2760 on a nonswitched line, or Write TCO or Write TVO operations for a 2760 on a switched line.

54 FRAMECH - Computes, and places in the Write CCW that sends frame change characters to a 2760, the address of the frame change characters in the terminal list used for the Write operation.

58 SWIDREC - Places the address and length of the terminal list read-in area in the Read ID ENQ CCW. (Terminal list of SWLST form)

5C SWIDSENT - Places the address and length of the ID ACK-0 sequence from the terminal list into the Write ID ACK-0 CCW.

60 SWLSTTIC - Places the address of the Enable CCW in the address field of the TIC CCW.

64 RDFRPOS - Places the address from the DECENTRY field in the DECB into the CCW used to set the buffer address for a local 3270 Read TMP or Read TBP. (If the second byte of the area specified by the entry operand

of the READ macro instruction does not contain an SBA order, control is returned to the problem program with a return code of X'24' in register 15.)

Read/Write Subroutines to Compute Count

The subroutines for computing the count field, if not already computed, are:

00 TESTLAST - The count need not be computed as the required count is in the model CCW; the TESTLAST subroutine determines if the current CCW is the last one in the channel program.

04 DATALNG - Obtains from the DECLNGTH field of the DECB the count for the Read or Write CCW, subtracts from this the value of the CCW count field, and places the result in the CCW count field.

08 BSCWLNG - Obtains from the DECWLNG field the length for the Read or Write CCW, and places it in the CCW count field.

0C CONVLENG - Compares the count in the DECLNGTH field with 20 (decimal); if the count exceeds 20, passes control to TESTLAST, as another CCW must be built to accommodate message text. If the count does not exceed 20, one CCW is sufficient to accommodate the expected data, and the subroutine places the count in the CCW count field, turns on the last-CCW flag (bit 0) of the TP-Cp code field of the CCW, and turns off the chain data flag.

3C BSCGRAPH - Places in a Write graphics CCW the address of graphic characters, for BSC Write Leading Graphics operations.

40 BSCADRS - Places in a Read Response CCW the address of the area specified in the DECRESPN field of the DECB and sets the command chain flag in the CCW, if necessary.

If dynamic buffering is specified, the subroutine DBDATA is entered from DATAREA. This subroutine sets up the additional CCWs that are needed and modifies the appropriate fields (area address, flags, TP-Op code, and count).

Channel End and Abnormal End Appendage (IGG019MB)

The channel end/abnormal end appendage receives control from the I/O supervisor when a channel end interrupt or an abnormal end interrupt occurs in connection with a channel program generated by the Read/Write routine. (An appendage is defined as the set of instructions that is entered by the I/O supervisor to perform the interrupt handling functions peculiar to an access method.) The appendage is entered through a pointer established in the IOS appendage table, which is accessed through a pointer placed in the DEB by the BTAM Open executor.

The primary function of the appendage is to initiate any further channel operations that are to be performed after the basic channel program generated by the Read/Write routine is interrupted by channel end (i.e., when polling, restart, or multiple addressing was specified in the basic channel program). The appendage also posts buffers (via supervisor Post routine) if dynamic buffering is in use, and updates the DECB sense (DECSENSO), command code (DECCMCOD), and CSW status (DECCSWST) fields. The appendage also changes remote 3270 RFT messages to a standard BTAM format and stores information and sets flags in the IOB when a remote 3270 error status message is received. The appendage runs in supervisor mode, disabled. See Charts B1 through B5A for the logic flow of IGG019MB.

Two sets of subroutines can be accessed as outlined in the following list. The second column shows the subroutine accessed at abnormal completion (either unit check, program check, protection check, channel data check, channel control check, interface control check, or chaining check). These subroutines are accessed via an abnormal end appendage entry. The third column shows the subroutine accessed at normal completion.

When an automatic polling operation terminates, either normally or with permanent error, the usage count (an indication of the total number of polling operations using the terminal list at any one time) of the terminal list employed is decremented by 1. If, additionally, the TP code of the last-executed CCW is not less than the code for Read Index (this indicates that an index byte has been read from the channel), BTAM moves the last-read index from byte 0 of the user's I/O area (or from byte 4 of the user's first buffer if dynamic buffer allocation is being used) to byte 36 of the DECB (DECPOLPT). After completion is posted to the DECB, the user may determine from the stored index byte the identity of the responding station and the action he wishes to take. BTAM uses this byte, hence it should not be modified by the user.

TP-Op Code	Abnormal Completion	Normal Completion
00	TP00AB	TP00
01	TPXX	TP01
02	TP02AB	TP02
03	TP03AB	TP03
04	TPXX	TPXX
05	TP05AB	TP05
06	TPXX	TP06
07	TPXX	TP07
08	TPXX	TPXX
09	TP09AB	TP09
0A	TP0AAB	TPXX
0B	TP0BAB	TB0B
0C	TPXX	TP0C
10	TP10AB	TP10
11	TP10AB	TP10
12	TP12AB	TP12
13	TPXX	TPXX
20	TP21AB	TP2A
21	TP21AB	TP2A
22	EXIT	EXIT
23	BRKTST	BRKTST
24	EXIT	EXIT
25	TP25	TP25

See charts B1 through B5A for the functions performed by these subroutines.

World Trade Telegraph Terminal Channel End and Abnormal End Appendage (IGG019PD)

This appendage receives control from the I/O supervisor when a program controlled interrupt, a channel end interrupt, or an abnormal end interrupt occurs in connection with a channel program generated by the Read/Write routine.

The primary function of this appendage, when dynamic buffering is specified, is to give control immediately to the PCI appendage if the PCI bit is alone in the channel and unit status. If the PCI bit is present at the same time as CE and DE, if no characters have been read, and if this is the first PCI, control returns to IOS. If this is not the first PCI, before again giving control to IOS, the appendage must find the CCW preceding the last one used, in order to store its address, incremented by 8, in the channel status word. This address is used to find easily the address of the last buffer when IOS again gives control to the Channel End appendage.

If the PCI bit is not present, the appendage analyzes the channel and status bytes. For all commands of World Trade Telegraph operations, channel end, device end, unit exception, and unit check can be considered as normal conditions. All the other conditions are considered as errors and control is given to the channel end/abnormal end appendage (IGG019MB) to take the appropriate action.

TP01: A Prepare command has been halted by a Halt I/O as a result of a RESETPL macro instruction (second operand omitted), or the line has failed.

If only CE, DE, and UE are present and the last command is a Prepare command, the EOT flag is set in the status flags (DECFLAGS) of the DECB, the IOB exception flag is turned OFF, the RESPN byte is reset, and channel end goes to TPXX to give control to the normal channel end appendage. (When a RESETPL is successful there is no transaction [no data has been transferred]. Therefore, the transmission count is decreased by one since the normal appendage [IGG019MB] increases this count by one in any case.)

If only CE and DE or CE, DE, and UC are present and the last command is a Prepare command, a character was entering the data register of the adapter. Channel end executes the next command in order to read the characters that are entering.

HIOREAD: A Read command has been halted by a Halt I/O as a result of a RESETPL macro instruction. The CCW is updated and control goes to IOS in order to reexecute the Read command.

TP07: If the ID from a WT terminal does not match the expected ID, channel end goes to TPXX. If the ID does match, channel end goes to IOS to restart the channel program.

TP11: If the last command is a Read command, a check is made to determine whether at least one character has been read. If no characters have been read, a test is made to determine whether UC is present with CE and DE.

If the time-out bit is present in the sense byte, the EOT flag is set in the status flags (DECFLAGS) of the DECB, the IOB exception flag is turned OFF, and channel end goes to TPXX to give control to the normal channel end appendage.

If the lost data bit is present in the sense byte, it is reset and channel end goes to TPXX. If one or more characters have been read, a test is performed to determine whether the last character is IAM or EOT.

If the last character is IAM, the WRU flag is turned ON in the status flags (DECFLAGS) of the DECB. Then, if the last character is IAM, EOT, or EOM, the residual count in the IOB is incremented by 1; that is, the number of characters read is decremented by 1. In all cases, a check is made to determine whether dynamic buffering was used, and to take the appropriate action.

If CE, DE, and UE are present and the last command is a Write command, or if UC is present and if the data check bit is in the sense byte, the IOB exception flag is turned OFF, the sense byte and the CSW are saved, and a Write Break is executed.

If error counts are specified, the data check counter is updated and is compared with the threshold data check counter; if unequal, channel end goes to TPXX. If equal, the IOB exception flag is turned ON so that the error recovery procedure can print the appropriate message.

TP23: The last CCW is a Write Break, so the sense byte and the CSW are restored, the contention flag is posted in the DECB, and channel end goes to TPXX.

TPXX: Restores the registers and gives control to the normal channel end/abnormal channel end appendage (IGG019MB).

Local 3270 Channel End/Abnormal End Appendage (IGG019PA)

The local 3270 channel end/abnormal end appendage receives control from the I/O supervisor (IOS) when a channel end or abnormal channel end interruption occurs for a local 3270 channel program generated by the Read/Write routine (IGG019MA). IOS uses the address palced into the appendage table of the DEB by the local 3270 BTAM Open executor when it loaded the appendage.

The local 3270 channel end/abnormal end appendage:

- Places information into control block fields.
- Retries channel programs that the local local 3270 ERP could not retry.
- Responds to local 3270 request-for-test (RFT) messages or recognizes that an on-line test is in progress.
- Recognizes that OLTEP is using a device.

The appendage places the residual count from the IOBCSW field in the ICB into the DECCOUNT field in the DECB. It places the status bits from IOBCSW into the DECCSWST field in the DECB. If abnormal channel end occurred because of unit check, the appendage places the sense data from the IOBSENS0 field in the IOB into the DECSENS0 field in the DECB.

If (1) bits 2 and 5 in the IOBFLAG1 field are zero, or (2) bit 5 in IOBFLAG1 is one and bits 5-7 in byte 2 in the IOBERRCT field are 010, 011, 101, or 110, or (3) bit 5 in IOBFLAG1 is one and the IOBECBCC field contains X'44', the appendage sets IOBERRCT to zero, turns off the busy bit (bit 1 in the IOBINCAM field) in every IOB (if the operation was Read Initial) or in the IOB for the operation only (if the operation was not Read Initial and a Read Initial operation is not pending), and returns control to IOS.

If the local 3270 ERP indicated that the appendage should retry the channel program (that is, bit 2 in the IOBFLAG1 field is one and bits 5-7 in byte 2 in the IOBERRCT field are 111), the appendage sets the IOBFLAG1, IOBFLAG2, IOBSENS0, IOBSENS1, IOBFLAG3, and IOBCSW fields to zero, sets the IOBECBCC field to X'7F', sets byte 2 in IOBERRCT to X'05', and returns control to IOS+8.

If a local 3270 RFT message was received (that is, the first three bytes of the input area contain SOH % /), the appendage converts the message to a standard BTAM RFT message, places the address of the message into the IOBERINF field, schedules the on-line test, and returns control to IOS+4. If an on-line test is in progress (that is, bit 7 in byte 1 of the IOBINCAM field is one), the appendage schedules the on-line test, and returns control to IOS+4.

If OLTEP is using a device (that is, bit 1 in the control byte in the UCB is one), the appendage turns on bit 7 in the DECFLAGS field in the DECB, and returns control to IOS.

For the logic flow of the local 3270 channel end/abnormal end appendage (IGG019PA), see Charts B9 and B10.

PCI (Program Controlled Interrupt) Appendage (IGG019MC)

Normally, program controlled interrupts are processed within a short time after the channel command word is fetched by the channel. The PCI appendage is entered from the ICS interrupt handler or from the WT terminal channel end/abnormal end appendage (IGG019PD).)

Dynamic buffering uses the program controlled interrupt feature to provide program control over the assigning of buffers to the Read or Write CCWs. For Read operations the PCI allows BTAM to obtain a buffer from the buffer pool or to set up the next Read command, and if other than the first buffer, to post the previous buffer. For Write operations the PCI indicates that BTAM is to obtain the address of the next buffer from the current buffer, (low-order three bytes of the first word in the buffer), and to set up the next Write command. The PCI appendage will post all buffers; on the first PCI there is no buffer to post since the first PCI fetches the first buffer.

With dynamic buffering, each Read or Write text CCW causes a PCI. The PCI appendage sets up a buffer address and alters the existing CCWs. This Read or Write cycle with PCIs will continue until a channel end interrupt occurs (end of block or end of transmission, or an error condition). (See Chart C1 for logic flow of PCI appendage).

The chain of commands that perform the Read or Write text operations with dynamic buffering has four CCWs. Initially constructed by the Read/Write routine, the four CCWs are actually a loop consisting of alternate Read or Write text commands and Read skip or transfer-in-channel (TIC) commands (see Figure 5). The Read skip or the invalid TIC (TIC with an invalid transfer address) prevents the Read or Write text CCW (other than the first) from being executed until the PCI appendage has updated the next Read or Write CCW with the proper buffer address. After the buffer address has been updated, the Read skip or invalid TIC is altered to a TIC for the next Read or Write text CCW.

Read Operations-Exceptional Conditions: The PCI flag in a CCW fetched on data chaining normally causes the interruption to occur within one character time after the CCW is fetched. The time of interruption, however, depends on the CPU model and the current activity in the system and may be delayed even if the channel is not masked. The unpredictable nature of the interruption time is further complicated by the fact that on the multiplexer channel the interruption may occur only between character times.

As a result of the preceding considerations, precautions must be taken to ensure that if the PCI is delayed so that there is not time to get a buffer, the read operation will terminate in an acceptable manner. Each of the two Read text commands in the loop is followed by a Read command with the PCI and Skip flags on, so that if the PCI on the Read text command fails to occur in time, the Read skip command is executed (with a large count). The Read skip command continues reading data into the channel, but does not place the data into main storage. Thus, termination of the Read operation occurs at the normal time, when EOB or EOT is received, and the line has been cleared of data, so that the problem program may request retransmission of the message. After the Read skip completes, bit 4 of DECFLAGS is turned on.

The PCI flag in the Read skip command serves as an indication to post the previous buffer and to free the unused buffer in the event that one was supplied, but was not obtained in time for the Read operation.

The second exceptional condition occurs when the PCI does occur in time, but no buffer is available. In this case, the IOB for the line is chained to a buffer request queue (two pointers are maintained in the access method section of the DEB for this purpose). The buffer request queue has first priority for any buffers that are released via the buffer (REQBUF/RELBUF) routine. If a buffer becomes available while the request is active, it is assigned to the Read operation by the Buffer routine. If no buffer becomes available, the PCI flag in the Read skip serves as an indication to dequeue the buffer request and to post the previous buffer. The PCI appendage also frees any buffer that became available too late for the Read operation.

Write Operations - Exceptional Conditions: The first condition described in the preceding section may also occur for Write operations, but it is much simpler to handle. If this condition occurs, an invalid TIC command is executed causing a program check, which causes the abnormal end appendage to be entered. This appendage gets the next buffer address and restarts on the next Write command.

The second condition cannot occur for Write operations. When a PCI occurs on a Write text command, the address of the next

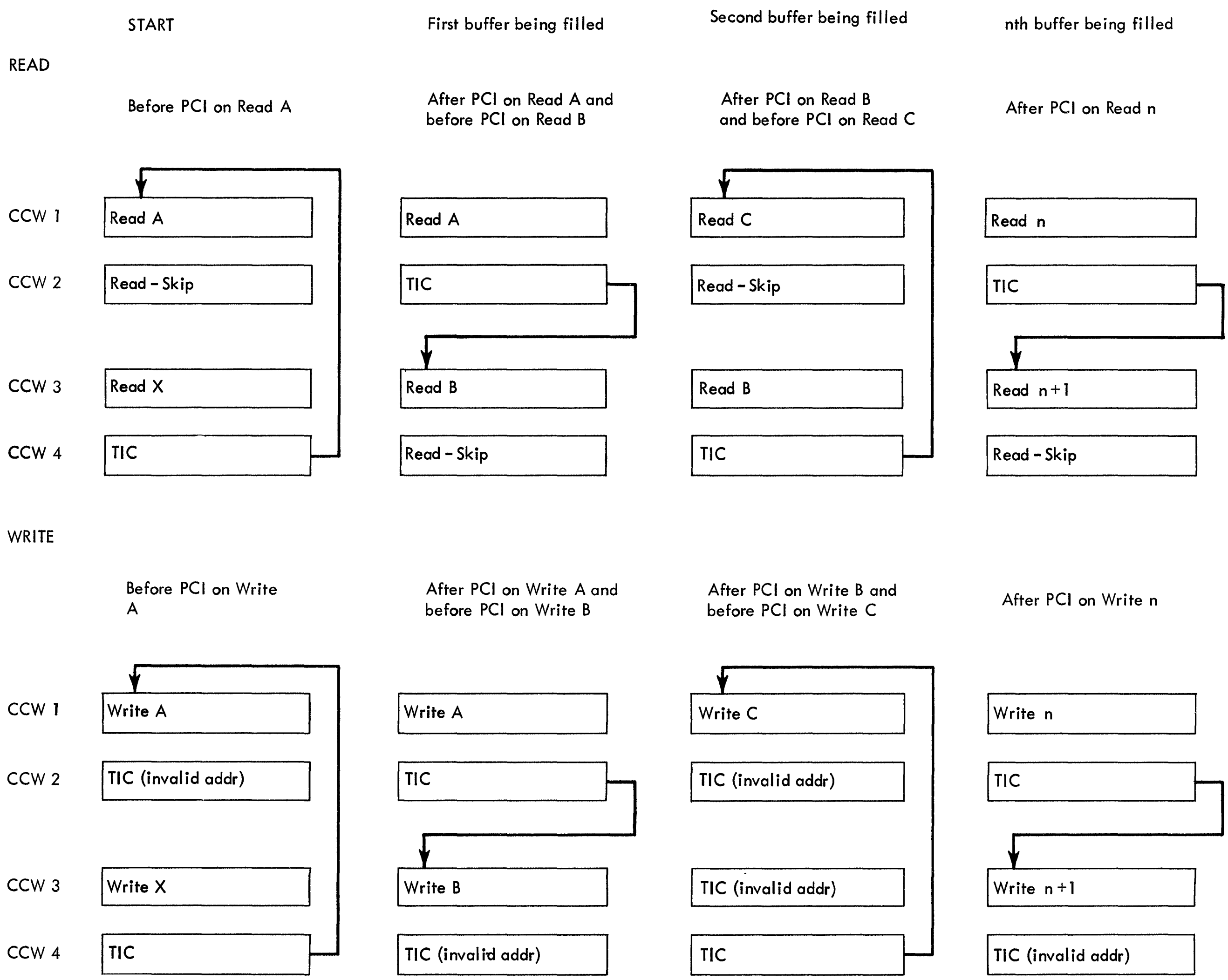


Figure 5. Dynamic Buffering Read/Write CCWs

buffer is obtained from the current buffer, and the next Write command is updated with the buffer address. If the next buffer has a chain address of zeros, the chain data flag is turned off in the next Write command so that the operation terminates on that command. The count for the last Write command is taken from the DECB (DECLNGTH). The TIC command is made valid by updating the address to point to the next Write command.

Local 3270 First-Level Attention Routine (IECTATEN)

During system generation, the address of the local 3270 first-level attention routine (IECTATEN), which is resident in the nucleus, is placed into the attention routine table (ATNTAB), and an index that gives the position of the address in the table is placed into the UCBATI field in the UCBs for local 3270 devices. When the I/O supervisor (IOS) receives control after an I/O interruption from a local 3270 device and finds the attention bit on in the CSW, IOS uses UCBATI and ATNTAB to locate the first-level attention routine and branches to it.

If the local 3270 device is not part of an open line group (that is, the IRB address field in the UCB is zero), the attention routine returns control to IOS. If an error occurred (that is bits other than attention and device end are on in the CSW), the attention routine turns on the intercept flag (bit 3 in the UCBFL1 field) in the UCB and returns control to IOS+4. If the skip flag (bit 6 in the control byte in the UCB) is on, the attention routine turns on the attention flag (bit 7 in the control byte), and returns control to IOS.

If the skip flag is not on and the attention flag is on, the attention routine returns control to IOS. If the attention flag is not on the read initial pending flag (bit 5 in the control byte in the UCB that has device index of one) is not on, the attention routine adds one to the attention count field (in UCB that has a device index of one), sets the attention flag (in the UCB for the device that caused the attention interruption), and returns control to IOS. If the read initial pending flag is on, the attention routine turns it off and schedules the local 3270 second-level attention routine (IGG019PG) by obtaining the IRB address from the UCB and the IQE address from the IRB, placing the address of the UCB for the device that caused the attention interruption into the IQEPARAM field of the IQE, and passing control to the asynchronous exit effector stage 2. When control returns to it, the attention routine returns control to IOS.

For the logic flow of the local 3270 first-level attention routine (IECTATEN), see Chart C2.

Local 3270 Second-Level Attention Routine (IGG019PG)

The local 3270 second-level attention routine (IGG019PG) is an asynchronous routine that is scheduled (through the asynchronous exit effector) by the local 3270 first-level attention routine (IECTATEN) by the local 3270 BTAM SVC routine (IECTSVC) to start a pending Read Initial operation. The second-level attention routine resides in SYS1.SVCLIB; it is loaded into the problem program's partition or region by the local 3270 BTAM Open executor (IGG0194Q).

The attention routine finds the initialized IOB, using the control block link field (in the UCB that has a device index of one), the DEBDCBAD field in the DEB, and the product of the initialized RLN field (in the UCB that has a device index of one) and the DCBEIOBX field in the DCB and the DCBIOBAD field in the DCB. If the RLN of the device that caused the attention interruption (given in the device index field in the UCB for that device) is the same as the RLN of the initialized IOB (given in the initialized RLN field in the UCB that has a device index of one), the attention routine, passing the address of the initialized IOB, invokes EXCP (SVC0) to start the Read Initial operation.

If the RLNs are different, the attention routine finds the IOB for the device that caused the attention interruption, using the control block link field, the DEBDCBAD field, and the product of the device index (in the UCB for the device that caused the attention interruption) and the DCBEIOBX field and the DCBIOBAD field. The attention routine copies the channel program area and the IOBECBPT, IOBFLAG1, and IOBFLAG2 fields from the initialized IOB into the IOB for the device that caused the attention interruption and, passing the address of the newly initalized IOB, invokes EXCP (SVC 0) to start the Read Intitial operation.

When control returns from EXCP, the attention routine zeros the initialized RLN field (in the UCB that has a device index of one), places the RLN of the device that caused the attention interruption into the DECPCIPT field in the DECB, and returns control to the a synchronous exit effector.

For the logic flow of the local 3270 second-level attention routine (IGG019PG), see Chart C3.

Local 3270 BTAM SVC Routine (IECTSVC)

The local 3270 BTAM SVC routine (IECTSVC) checks and sets attention handling flags in the UCBs for local 3270 devices. The SVC routine, which is resident in the nucleus, has five entry points corresponding to five attention handling functions:

- IECTRDIL - Queue a Read Initial request if no outstanding attention interruption exists for the line group.
- IECTATRL - Reset the attention flag for a device and update the attention count for the line group.
- IECTCHSL - Set the skip flag for a device in order to inhibit Read Initial operations.
- IECTCHAL - Reset the skip flag for a device in order to permit Read Initial operations.
- IECTRSTL - Dequeue a Read Initial request.

The SVC routine is invoked by means of SVC 116; a routing code determines which of the five entry points receives control.

IECTRDIL Entry Point: The SVC routine is entered at IECTRDIL from the BTAM Read/Write routine (IGG019MA) during processing of a Read Initial request.

(The Read/Write routine uses IECTRDTI, an internal macro instruction that determines the address of the UCB for the device and invokes SVC 116 with a routing code of zero.) If the address of an invalid UCB is received, the SVC routine returns control to the Read/Write routine with a return code of X'08'. If at least one attention interruption is outstanding (that is, the attention count field in the UCB that has a device index of one is greater than zero), the SVC routine returns control to the Read/Write routine (thorugh the SVC Exit routine) with a return code of X'00'. If no attention interruption is outstanding, the SVC routine queues the Read Initial request by setting the read initial pending flag (bit 5) in the control byte in the UCB that has a device index of one. The SVC routine places the device index from the UCB for the device specified in the Read Initial request into the initialized RLN field in the UCB that has a device index of one. The SVC routine returns control to the Read/Write routine (through the SVC Exit routine) with a return code of X'04'.

IECTATRL Entry Point: The SVC routine is entered at IECTATRL from the BTAM Read/Write routine during processing of any Read or Write request. (The Read/Write routine uses IECTATNR, an internal macro instruction that invokes SVC 116 with a routing code of one.) If the address of a UCB that is invalid or that is not for a local 3270 device is received, the SVC routine returns control to the Read/Write routine with a return code of X'04'. If the attention flag (bit 7 in the control byte) in the UCB for the device specified in the Read or Write request is off, the SVC routine returns control to the Read/Write routine (through the SVC Exit routine) with a normal return code (X'00'). If the attention flag is on, the SVC routine turns it off. If the skip flag (bit 6 in the control byte) in the UCB for the device specified in the Read or Write request is on, the SVC routine returns control with a normal return code. If the skip flag is off, the SVC routine subtracts one from the attention count field in the UCB that has a device index of one and returns control to the Read/Write routine (through the SVC Exit routine) with a normal return code.

IECTCHSL Entry Point: The SVC routine is entered at IECTCHSL from the expansion of the CHGNTRY macro instruction when the SKIP operand is specified. (The expansion invokes SVC 116 with a routing code of two.) If the address of a UCB that is invalid or that is not for a local 3270 device is received, the SVC routine returns control to the macro instruction expansion with a return code of X'08'. If the skip flag in the UCB for the device specified in the skip request is on, the SVC routine returns control to the macro instruction expansion (through the SVC Exit routine) with a normal return code (X'00'). If the skip flag is off, the SVC routine turns it on. If the attention flag in the UCB is off, the SVC routine returns control with a normal return code. If the attention flag is on, the SVC routine subtracts one from the attention count field in the UCB that has a device index of one and returns control to the CHGNTRY macro instruction expansion (through the SVC Exit routine) with a normal return code.

IECTCHAL Entry Point: The SVC routine is entered at IECTCHAL from the expansion of the CHGNTRY macro instruction when the ACTIVATE operand is specified. (The expansion invokes SVC 116 with a routing code of three.) If the address of a UCB that is invalid or that is not for a local 3270 device is received, the SVC routine returns control to the macro instruction expansion with a return code of X'08'. If the skip flag in the UCB for the device specified in the activate request is off, the SVC routine returns control to the macro instruction expansion (through the SVC Exit routine) with a normal return code (X'00'). If the skip flag is on, the SVC routine turns it off. If the attention flag in the UCB is off, the SVC routine returns control with a normal return code. If the attention flag is on and the read initial pending flag in the UCB that has a device index of one is off, the SVC routine adds one to the attention count in the UCB that has a device index of one and returns control with a normal return code. If the read initial pending flag is on, the SVC routine turns it off and schedules the local 3270 second-level attention routine (IGG019PG) by obtaining the IRB address from the UCB and the IQE address from the IRB, placing the address of the UCB for the device specified in the activate request into the IQEPARAM field of the IQE, and passing control to the asynchronous exit effector stage 2. When control returns to it, the SVC routine returns control to the CHGNTRY macro instruction expansion (through the SVC Exit routine) with a normal return code.

IECTRSTL Entry Point: The SVC routine is entered at IECTRSTL from the expansion of the RESETPL macro instruction when a local 3270 device is specified. (The expansion invokes SVC 116 with a routing code of four.) If an invalid address or the address of an invalid UCB is received, the SVC routine returns control to the macro instruction expansion with a return code of X'0C'. If the address of a UCB that is not for a local 3270 device is received, the SVC routine returns control with a return code of X'08'. If the Read Initial operation is in progress (that is, the read initial pending flag in the UCB that has a device index of one is off), the SVC routine returns control to the macro instruction expansion (through the SVC Exit routine) with a return code of X'04'. If the Read Initial request is queued (that is, the read initial pending flag is on), the SVC routine turns the read initial pending flag off and passes control to the Post routine to post (with a completion code of X'48') the ECB associated with the queued Read Initial request. When control returns to it, the SVC routine returns control to the RESETPL macro instruction expansion (through the SVC Exit routine) with a return code of X'00'.

For the logic flow of the local 3270 BTAM SVC routine (IECTSVC), see Charts C4, C5, and C6.

Error Recovery Procedures

The Error Recovery Procedures (ERP) routines are a composite set of routines designed to diagnose and recover, if possible, from all errors occurring during telecommunication operations. The error routines perform the following basic functions:

1. Automatic retry of all errors not involving data transfer.

2. Statistical recording of temporary and permanent unit check errors.

3. Error messages to the operator console for all permanent errors.

In addition, the following optional features can be included and performed by ERP.

1. If, in the DCB macro, the LERB operand specifies the address of the line error recording block (LERB) and the EROPT operand specifies C, (with any of the possible valid combinations), and if the LERB macro instruction is coded, ERP will maintain a count of all data check, timeout, and intervention-required errors, and a count of transmissions on the line.

2. If EROPT=W (with any of the possible valid combinations) is coded in the DCB macro instruction, ERP will retry Write operations for write errors that occurred during text transfer. This parameter, W, is valid only for start-stop terminals; it is ignored for BSC stations as BSC Write operations are always retried. (Write operations are not retried, however, if dynamic buffering is used.)

3. If EROPT=R (with any of the possible valid combinations) is coded in the DCB macro instruction, ERP will retry Read operations for read errors that occurred during text transfer. This parameter, R, is valid for certain start-stop terminals; it is ignored for BSC stations as BSC Read Text operations are always retried. (Read operations are not retried, however, if dynamic buffering is in use.)

The Read and Write retries will <u>not</u> be performed for devices not capable of accepting them. See the DCBERROP field in Table 1 (Appendix B) for the types of stations for which retries can be performed.

Consisting of 27 modules, ERP operates in the nucleus error transient area with a supervisor protection key. The routines are entered from BTAM through IOS, and are always entered through the Control module (IGE0004A or IGE0004C). The other modules are called as needed for error checking and error recovering.

The error recovery control module sets up the linkage for subsequent loads of BTAM error modules. It then determines if entry is due to an initial error (i.e., one that occurred in a user-program Read or Write operation) or as the result of an ERP channel program execution. If it is an initial error, the error flags in IOB are set on to indicate that ERP is in control. By checking the condition code, the sense bytes, and the status bytes in the CSW, ERP determines the type of initial errors encountered and links to the proper error recovery routine.

If ERP was previously in control upon entry to the control module, it checks to determine if an error retry was attempted. If it was, ERP checks to see if the error was recovered. If there is still an error (it may possibly be a different type of error on the same line), the control module determines the type of error and links to the proper routine via the communications vector table (CVT) XCTL routine. If the error retry count has been reached, and statistical data recording is required, ERP exits to the Statistical update routine (part of the resident supervisor) which updates the Outboard Recorder (OBR) before going to the supervisor Post routine to post the operation complete with error (X'41'). Otherwise, if the error was recovered, or the retry count was not reached, ERP checks to see if statistical recording is required. If the error was recovered, the ERP-in-control flags in the IOB are turned off and ERP exits to update the statistic table before going to supervisor Post to post the event complete without error (X'7F'). If no statistical recording is required, ERP simply goes to supervisor Post to post the event complete (X'7F'). If the retry count was not reached and an error occurred, ERP checks for the type of error encountered and then links to the proper ERP module by the CVT XCTL routine.

The remote 3270 Error Post routine (IGE0704B) formats the address and error information from a remote 3270 error status (SOH % R) message into a T-type record. The Error Post routine receives control from the BSC ERP control routine (IGE004C). The Error Post routine passes control to the TP recorder (IGE0625F) to have the T-type record recorded in SYS1.LOGREC.

See Charts E0-E9, F0-F9, G0-G9B, and J0A-J2C for the logic flow of the ERP modules.

For a description of the ERP rouitnes for the local 3270 display system, see Appendix L.

Buffer Routine (IGG019MS)

When the user specifies a buffer pool, he may wish to use the REQBUF and RELBUF macro instruction to obtain from the pool, or restore to the pool, one or more buffers. The Buffer routine performs these functions, when entered from REQBUF or RELBUF via SVC 58. (See Chart 06 for the logic flow of the Buffer routine module.)

When dynamic buffering is being used, the Buffer routine is activated asynchronously by the channel end appendage (IGG019MB) and the PCI appendage (IGG019MC) to provide management of buffers for the channel program already started. The routine is always resident and disabled.

The address of the buffer pool is located in the DCB; the address of the Buffer routine is in the DEB following the table of UCB addresses. The Buffer routine manages a last-in first-out (LIFO) queue of buffers for each buffer pool through use of the buffer control block (BCB). The buffer control block is an 8-byte area in storage whose address is contained in the DCB (DCBBUFCB field). The initialization and location of the BCB is dependent on the choice of buffering techniques. Refer to the publication <u>IBM System/360 Operating System: Basic Telecommunications Access Method</u>, GC30-2004.

The buffer control block contains the address of the first available buffer, the count of available buffers, and the length of each buffer (all are the same length). The length includes the fullword chaining address of the next available buffer. See Figure 6.

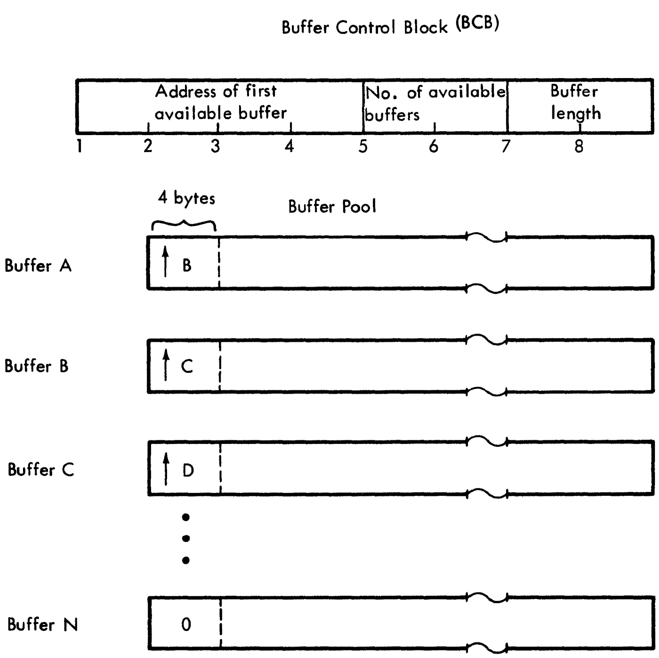


Figure 6. Buffer Control Block

The address of the Buffer routine is placed in the DEB following the table of UCBs. Following the address is a pointer to the request queue and then a pointer to the last entry in the request queue. (These requests are the IOBs for the lines.) These request elements are requests for buffers for Read operations and are placed in the queue only by the PCI appendage. Each element is linked, via the Read CCW requiring the buffer address with a pointer in the data address field, to the next CCW requiring a buffer. (See Figure 7.)

Elements are removed from the queue by the Buffer routine when buffers are returned to the buffer pool.

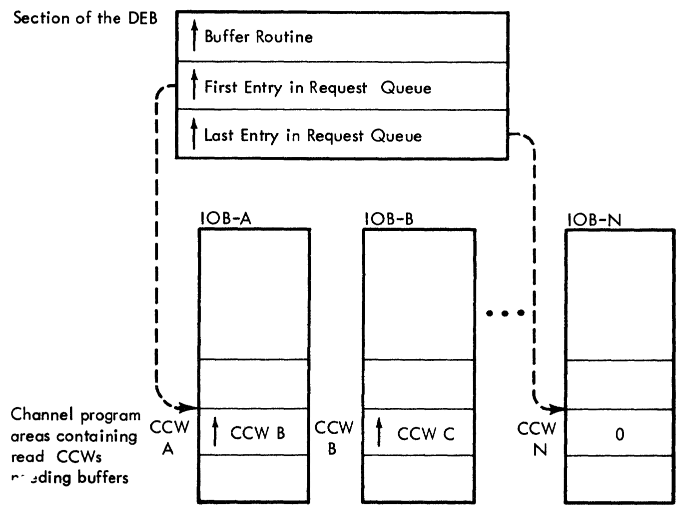


Figure 7. Buffer Request Queue

Change Entry Routine (IECTCHGN)

The Change Entry routine is entered via the expansion of a CHGNTRY macro instruction specifying SSALST, SSAWLST, AUTOLST, or AUTOWLST. The macro expansion includes a V-type address constant that generates an automatic call for the routine when the problem program is link-edited. The routine performs the function of physically restructuring the polling list in order to cause an entry to be skipped or reactivated.

If SKIP is specified as the action operand, the entry corresponding to the index value specified in the position operand will be removed from the list of active entries, the active entries which followed it will be moved up an amount equal to the entry width, the skipped entry will be placed in the position previously occupied by the last active entry, and the number of active entries will be decremented.

If ACTIVATE is specified as the action operand, the entry corresponding to the index value specified in the position operand will be removed from the list of inactive (skipped) entries. All inactive entries which precede it in the list and all active entries of greater index value will be moved down an amount equal to the entry width. The activated entry will be placed in the slot previously occupied by the active entry of next greater index value or by the first inactive entry. The number of active entries will be incremented.

Issuing a CHGNTRY macro causes no action to be performed if the list is currently in use by a Read operation, if the specified entry is already active (for ACTIVATE) or inactive (for SKIP), or if the operand specifying the position of the entry is too large (i.e., no such entry exists).

Return Codes
A return code in register 15 indicates the result of issuing the CHGNTRY macro.

Code	Meaning
X'00'	The requested action was performed or was not needed (for SKIP, the entry was already skipped; for ACTIVATE, the entry was already active). (This code is also returned if the operand specifying the position of the entry is 0.)
X'04'	The requested action was not performed, because the list was in use.
X'08'	The requested action was not performed, because the operand specifying the position of the entry was too large.

(See Charts K1 and K2 for logic flow of IECTCHGN).

Edit Routine (IECTEDIT)

The Edit routine is entered via the expansion of a TPEDIT macro instruction issued in the user program. The purpose of the routine is to edit data received from an IBM 50 Magnetic Data Incriber attached to an IBM 2772 Multipurpose Control Unit. The routine scans the input area into which IBM 50 MDI data is received, for special MDI control codes, such as LZ (left zero fill), LZS (left zero start), DUP (duplicate), and CAN (cancel). Upon encountering one of these codes, the routine performs the appropriate function such as duplicating specified characters from the previous record, deleting cancelled records, and performing left-zero justification. Information in a parameter list passed to the routine from the TPEDIT macro governs the execution of certain of the functions. At user's option, records found to be error will be passed to the user-specified error exit routine; the record so passed is preceded by an error description word that conveys the nature of the error.

Line Error Print Routine (IECTLERP)

The Line Error Print routine is entered via the expansion of a LERPRT macro instruction issued in a BTAM problem program. The macro expansion includes a V-type address constant which generates an automatic call for the routine when the problem program is link-edited. The routine will set up to print line error counts for data check, intervention required, non-text timeout errors, and the number of transmissions for the lines specified in the LERPRT macro instruction or for all lines in the line group. These counts are obtained from the line error recording block (LERB), the format of which is shown in Figure 8.

The routine initializes the pointers before checking for a line group print. If counts for an entire line group are to be printed, the routine determines the number of lines from the data extent block field DEBNMEXT. Since the pointers are initialized to the line prior to the line to be printed, the routine now picks up the line to be printed, updates all the accumulative counters, and places the counts in the generated message before exiting to Write to Operator (WTO) to print the message on the system console. The message area and error counters are cleared. If messages are to be printed for all lines in the group, the routine checks to see if all lines have been printed by looping back to step to the next line in the group. When all messages for lines have been printed, the routine returns to the problem program. (See Chart F7 for logic flow of IECTLERP).

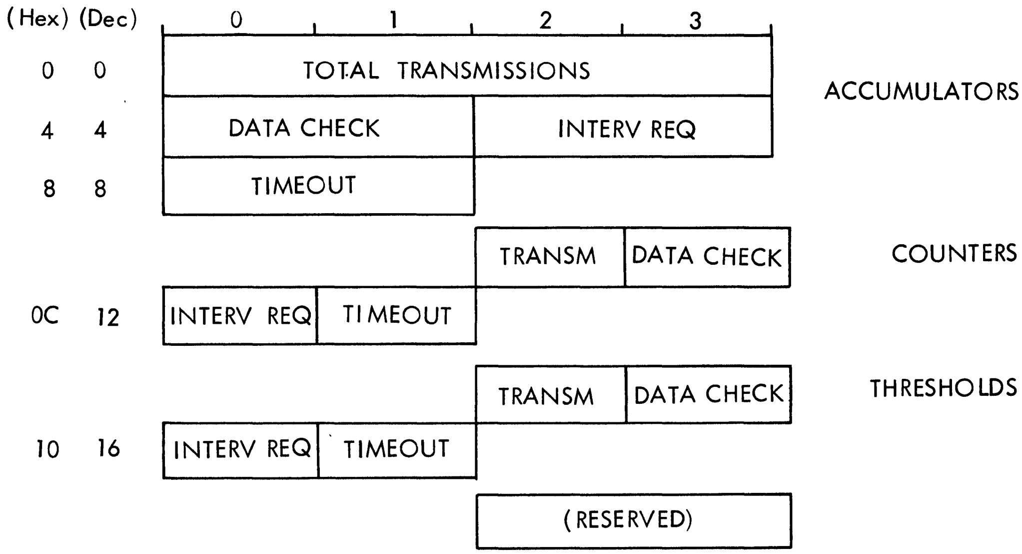


Figure 8. Format of Line Error Recording Block (LERB)

Line Open Routine (IECTLOPN)

The Line Open routine is entered via expansion of the LOPEN macro instruction issued in a BTAM problem program to attempt to correct an error which occurred at Open time or to reestablish data set (modem) synchronism for a line using an IBM 3977 Model 2 modem. The macro expansion includes a V-type address constant which generates an automatic call for the routine when the problem program is link edited. The error is indicated by a return code of X'14' in register 15 after the problem program has issued a READ or WRITE macro instruction.

The routine builds the channel program needed to correct the error condition, then issues an EXCP to execute that program.

The routine first verifies that the line group data set has been opened, the user-specified relative line number is valid, the IOB is not busy, and the transmission control unit is operational. If any of these conditions is not true, the routine returns to the user program with the appropriate return code (X'10', X'0C', X'08', or X'04, respectively) in register 15. If OLTEP is using the device, the routine returns control to the problem program with a return code of X'14' in register 15. Before checking for an operational TCU, the routine checks whether the device specified in the LOPEN macro instruction is a local 3270 device. If it is, the routine turns on the busy flag (bit 1 in IOBINCAM), sets up an Erase/Write command to unlock the keyboard and reset the modified data tags, and starts the channel program by issuing EXCP. If the device is not a local 3270 device, the routine checks whether the TCU is operational.

The routine then sets the busy flag (bit 1 of IOBINCAM) and sets up a Disable command. If the TCU is a 2702, a Set Address (SAD) command is set up following the Disable; if the TCU is a 2701 or 2703, no SAD command is set up. Then, if the TCU is a 2701 or 2703 with a BSC adapter, a Set Mode command is set up following the Disable or SAD command; otherwise, no Set Mode command is set up. The routine then, via EXCP, starts the channel program just built.

If the channel program is not posted complete in the ECB within about four seconds, the routine sets the SAD/Enable error flag in IOBINCAM and returns to the user program with a return code of X'04' in register 15. If the channel program is posted complete, the routine determines if that completion was normal (X'7F' in ECB). If so, the routine turns off the SAD/Enable error and busy flags in IOBINCAM and returns to the user program. If the completion was not normal, the routine turns on the SAD/Enable error flag and returns to the user program with a return code of X'04'.

See Chart 07 for the logic flow of IECTLOPN.

Translate Routine (IECTTRNS)

The Translate routine is entered from the expansion of the TRNSLATE macro instruction issued by a BTAM problem program. The macro expansion includes a V-type address constant that generates an automatic call for the routine when the problem program is link edited. The routine checks first for dynamic buffering. If dynamic buffering was not specified, the routine checks for a count greater than 256 in which case it translates 256 bytes and loops back to check again. When the count is less than 256, IECTTRNS translates the remaining number of bytes and exits.

When dynamic buffering is specified and the count is given, the count is adjusted for the link field. The routine translates 256 bytes at a time until the count is less than 256. After translating the buffer, the routine checks for last buffer. If this is the last buffer, the routine exits; otherwise, it loads the address of the next buffer, loads the count from the DCB, adjusts the count for the link field, and begins the 256 byte translating loop.

If dynamic buffering was specified and the count was not given, the routine gets the count from the DCB and then performs the same as if the count had been specified with dynamic buffering. (See Chart 08 for logic flow of IECTTRNS.)

On-Line Testing

On-line testing is an optional BTAM facility that permits the user to verify proper operation of terminals and of the communication lines that link them to the computer, and to aid in diagnosing line or terminal troubles. On-line testing centers around transmission of predefined standard test messages, whose formats depend on the purpose of the test.

The user requests transmission of test messages during normal BTAM operation, and only the communication lines and terminals he specifies are involved; data transmission proceeds as usual on other lines. Operation of the user's problem program is affected only to the extent of the line time required for test transmissions and of the CPU time required to process requests for tests.

On-line testing is available only if the user codes T among the EROPT options in the DCB macro for the affected line group. This causes the on-line test control module, IGG019MR or IGG109PI, to be loaded at Open time.

On-line testing is implemented somewhat differently for start-stop lines, for binary synchronous lines, and for local 3270 devices, and the three types are therefore discussed separately.

On-Line Testing for Start-Stop Communication Lines

For start-stop communication lines, test requests may be initiated only at remote terminals. The tests requested may involve message switching, comparing of the contents of a test message to a stored pattern in main storage, sending a string of characters to a specified terminal, or checking the IBM SELECTRIC typing element mechanism of a terminal printer.

Test request message formats always begin with a five-byte zoned decimal field containing 9's. Any such message received by a Read text command is recognized as a test request message by the channel end appendage (IGG019MB), which transfers control to the On-Line Test Control Module (IGG019MR) without posting the operation as complete in the DECB for the line. IGG019MR sets up and executes the channel program appropriate to the kind of terminal and line involved, and sends the requested test message. In generating the channel program, the module uses the SVC 66 module (IGC0006F) and the appropriate device-specific test module: IGC0106F (for 1030), IGC0206F (for 1050), IGC0306F (for 1060), IGC0406F (for 2740), IGC0906F and IGC0A06F (for 2741), or IGC0506F and IGC0606F (for 2260/2848), or IGC0B06F (for 2760). These are type 4 SVC routines that operate in the SVC transient area. IGC019MR is entered from the channel end appeandage each time channel end occurs during the on-line test, and upon determining that the test is completed, frees any main storage used by the channel program as a test message area, and restores the problem program Read Initial operation.

On-Line Testing for Binary Synchronous Communication Lines

On-Line tests for BSC lines may be requested by the central computer, by remote computers or terminals, or both, depending upon the type of test and the line and computer/terminal configuration. There are 34 types of tests, not all of which apply to all configurations.

To request a test at a remote computer or terminal, the operator at that location enters a message, called a request-for-test (RFT) message, having a special format. To request a test at the central computer, the programmer codes an ONLTST macro instruction in his problem program at the point at which he wishes a test to be performed. The ONLTST macro generates the proper RFT message and sends it to the remote computer or terminal the user specifies in the ONLTST macro. The format of the RFT message is the same whether it is sent by the central computer or is received by the central computer from a remote computer or terminal.

Transmission of an RFT message is followed by one or more transmissions of test messages. The RFT message contains a field called the X field, by which the user codes the test type, from 00 to 34, to be performed. The test type governs the sequence of I/O operations comprising the test and determines the content of the test message.

Type 00: For this test type the RFT message is accomplished by user-specified text data. For RFT messages received from a 2770 or 2780 terminal the text data is part of the RFT message; for RFT messages received from other types of BSC stations or sent to any type of BSC station, the text data is sent as a separate test message. When the RFT message includes the text data, the text data is received only once, and the receiving computer sends a single acknowledgment in reply. When the text data is contained in a separate test message, the computer or terminal sending the RFT sends the test message the number of times specified in the Y field of the RFT message. The Y field may specify from 1 to 99 transmissions. The receiving computer or terminal sends an acknowledgment after each message.

Type 01: For this test type the RFT message includes user-specified text data. The computer or terminal receiving the RFT message acknowledges it, sets up a test message containing the text data, and sends the test message the number of times specified in the Y field of the RFT message, from 1 to 99. The computer or terminal receiving the test messages sends an acknowledgment each time a message is received. Notice that in type 00 tests the computer or terminal that sends the RFT message also sends the test messages, and receives acknowledgments in reply, while in type 01 tests the computer or terminal that sends the RFT message receives test messages in reply.

Types 02-34: For these types of tests, test messages consist of BTAM-defined, rather than user-specified, text data. Otherwise, they are similar to type 01 tests. When BTAM receives an RFT message it examines the X and Y fields, selects the test message designated by X and sends it the number of times specified by Y (from 1 to 99).

RFT message formats always begin with a two-byte field containing SOH %. Any such message received on a Read Initial or Read Connect operation or on a Read Continue operation after a general poll of a remote 3270 display system or sent via ONLTST macro is recognized as an RFT message by the channel end appendage, IGG019MB, which transfers control to the On-Line Test Control module (IGG019MR) without posting the operation as complete in the DECB for the line. IGG019MR sets up and executes the channel program appropriate to the kind of transmission code (EBCDIC, USASCII, or TRANSCODE) and kind of test requested. IGG019MR uses the SVC 66 module (IGC0006F) and the On-Line Test Control (BSC) module (IGC0D06F) to set up the kind of channel program needed, and (via XCTL from IGC0006F) uses module IGC0706F, IGC0806F, IGC0E06F, IGC1406F, IGC0F06F, IGC1306F, IGC1006F, or IGC1106F if BTAM-defined test messages are to be sent. These modules provide to the channel program the requested kind of test message, and start the channel program. IGG019MR is entered from the channel end appendage each time channel end occurs during the on-line test. Each time it is entered, it determines whether the test has been completed (i.e., all requested test messages have been sent), restarts the channel program as necessary for each test message transmission, performs error checking, and accumulates error counts. When all test messages for a given on-line test have been sent, IGG019MR writes to the operator of the central computer a message containing the error counts for the operation and restores the problem program Read Initial of Read Connect operation.

If the RFT message was received on a Read Continue operation, the operation is posted complete, and an EOT is placed into the first byte of the input buffer.

Note: For the logic flow of on-line test modules for the remote 3270 display system, see Charts D1 through D5, TC and T01, and T10 through T15.

When the user program initiates an on-line test, test mode is not entered until BTAM receives an ACK-0 from the remote station. If after seven retries the ACK-0 has not been received, the test is aborted, and the user-program Read Initial or Read Connect operation is restarted.

During the test, up to 25 consecutive WACK responses will be accepted by BTAM from the remote station before the test is aborted.

On-Line Testing for Local 3270 Devices

On-line tests for local 3270 devices are requested from the local 3270 display station by entering a request-for-test (RFT) message. Transmission of an RFT message is followed by one or more transmissions of a test message. The RFT message contains an X field in which the display station operator codes the test type, from 23 to 28, to be performed. The test type governs the sequence of I/O operations making up the test and determines the content of the test message.

RFT messages received on a Read Initial operation from a local 3270 display station are recognized by the local 3270 channel end/abnormal end appendage (IGG019PA). By means of the asynchronous exit effector, the appendage schedules the local on-line test control (IGG019PI) without posting the operation complete in the DECB. IGG019PI sets up the type of test requested, handles I/O interruptions during the test, and ends the test. By means of SVC 66, IGG019PI passes control to the start-stop on-line test control (IGC0006F), which transfers control to the local 3270 on-line test control (IGC1206F). IGC1206F sets up the channel program for the type of test requested and passes control to a 3270 EBCDIC test module (IGC0E06F, IGC1406F, or IGC1306F) to provide the channel program with the test message for the requested type of test and to start the channel program. IGG019PI is entered from the local 3270 channel end/abnormal end appendage each time channel end occurs during on-line test. Each time it is entered, it determines whether the test has been completed, restarts the channel program as necessary for each test message transmission, and retries (three times) tests with error ending status before terminating test. When all test messages for a given on-line test have been sent, IGG019PI restarts the Read Initial operation.

For the logic flow of on-line test modules for the local 3270 display system, see Charts D5A and D5B, T0 and T02, and T10 through T13.

REQBUF/RELBUF (SVC 58) Routine (IGC058)

SVC 58 provides access to the Buffer routine from either the REQBUF or RELBUF macro. A type I SVC (resident and disabled), the routine is a part of the nucleus of all systems using BTAM. It calculates the address of the buffer routine, releases control to it, and passes information from the Buffer routine to the problem program. (See Chart 09 for the logic flow.)

BTAM Close Executor (IGG0203M)

The Close executor logically detaches a communication line group from the system. (See Chart 10 for the logic flow.) The routine performs the following:

- Halts any outstanding enable commands.
- Stops I/O activity on the lines.
- Disables the lines.
- Frees the storage obtained by open for input/output blocks.
- For a line group of local 3270 devices, frees the storage obtained by the local 3270 Open executor for the interruption request block and interruption queue element and clears the fields in the graphics devices segment of the UCBs.
- Reinitializes DCB fields.
- Frees the storage occupied by the buffer pool, if a buffer pool was obtained by the Open executor.

After the Close executor has been executed for each of the DCBs in the parameter list of the CLOSE macro, it returns control to the system Close routine which completes the processing, deletes the BTAM routines and appendages, frees the storage occupied by the data extent block (DEB), and completes the reinitialization of DCB fields.

BTAM CONTROL INFORMATION FOR CHANNEL PROGRAM GENERATION

This section describes the form and content of the BTAM channel command word (CCW) and the control fields used for channel program generation. A knowledge of these control fields is essential to understanding the channel program discussion in the following chapters.

BTAM Channel Command Word

The format of the BTAM channel command word is as follows:

Command Code	Data Address
0 7	8 31

Flags	000	TP Op Code	Count
32 36	37 39	40 47	48 63

The CCW used within BTAM is identical to that used throughout System/360 except for the addition of a teleprocessing operation code (TP-Op Code) in the sixth byte (bits 40 through 47). This byte, which is unused in other environments, has no effect upon channel operations. Bit 40 is set to one in the last CCW created dynamically for a channel program. Bit 41 is reserved for use with dynamic buffering. The use of bits 42 through 47 is described in the chapter, BTAM Channel Programs General.

Device I/O Modules

A device I/O module contains the control information for the generation of channel programs for a given type of remote or local station or line configuration. There is a spearate device I/O module for each type of start-stop terminal (e.g., IBM 1030, 1050, 2740 with Dial-up, 2740 with Station Control, AT&T 83B3) and each type of BSC line configuration: nonswitched point-to-point, nonswitched multipoint, and switched point-to-point. The module is loaded into main storage by the BTAM Open executor.

There is a separate device I/O module for the local 3270 display system; it is loaded into main storage by the local 3270 BTAM Open executor.

The device I/O modules are listed below and in Appendix I.

Figure 9 shows the format of the device I/O module; the format is the same regardless of the type of device for which the module is used. The device I/O module has four parts:

- A 32-byte table of offsets.
- The channel program offsets for the Read and Write operations.
- The channel command words for the channel programs.
- A table of special characters (start-stop only).

The 32-byte table of offsets is at the beginning of the device I/O module. Each byte contains the binary offset factor used to access the channel program offsets for an I/O operation. If the operation is invalid for the type of device represented by the module, the byte contains an offset value of all ones (hexadecimal FF). If one of these invalid bytes is accessed, control is immediately returned to the calling routine, with register 15 containing return code X'0C' to indicate that the requested operation is not valid for the type of device involved.

Where more than one operation type appears within an offset byte, only one type is valid for the device represented by the device I/O module. In Figure 6, for example, the offset values and operation types are those applicable to the IBM 1050 on a nonswitched line. The shaded areas in the table of offsets indicate the operation types and bytes that do not apply to the IBM 1050 on a nonswitched line.

Byte 18 (hex) contains the offset to the channel command words.

Read/Write Operation Types

An operation type is associated with each byte in the table of offsets; the position of the byte corresponds to the value of the type code in DECTYPE field of the DECB.

Byte (hex)	Operation Type
0	Write Break (TB)
1	Read Initial (TI)
2	Write Initial (TI)
3	Read Continue (TT)
4	Write Continue (TT)
5	Read Conversational (TV)
6	Write Conversational (TV)
7	Read Repeat (TP) Read Continue with ID Exchange (TE)
8	Write Positive Acknowledgment (TA)
9	Read Skip (TS)
A	Write Negative Acknowledgment (TN) Write Reset (TR) Write Disconnect (TN)(TWX)
B	Read Buffer (TB)
C	Write at Line Address (TL) Write Initial Optical (TIO) Write Initial Transparent Block (TIE)
D	Write Initial Conversational (TIV) Read Continue with Leading Acknowledgment (TTA)
E	Write Erase (TS) Write Invitational Optical (TCO) Write Continue Transparent Block (TTE)
F	Write Continue Conversational (TTV)
10	Write Disconnect (TD) (BSC)
11	Read Connect (TC) Read Modified (TM) (Local 3270)
12	Write Initial Transparent (TIX)

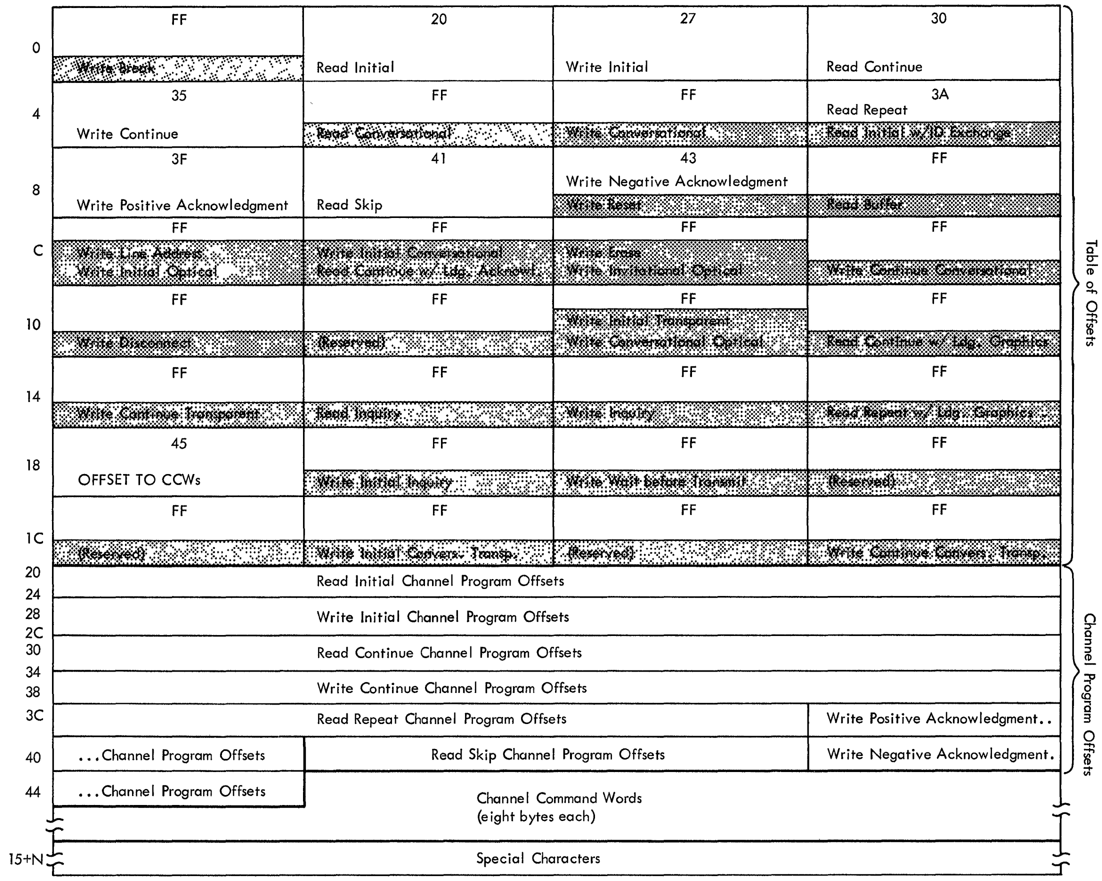


Figure 9. Format of Device I/O Module (showing nonswitched 1050 Offsets)

	Write Conversaticnal Optical (TVO)
	Write Unprotected Erase (TUS) (Local 3270)
13	Read Continue with Leading Graphics (TTL)
	Read Buffer from Position (TBP) (Local 3270)
14	Write Continue Transparent (TTX)
15	Read Inquiry (TQ)
16	Write Inquiry (TQ)
17	Read Repeat with Leading Graphics (TPL)
18	Reserved
19	Read Initial Inquiry (TIQ)
	Read Modified frcm Position (TMP) (Local 3270)
1A	Write Wait Before Transmit (TW)
• 1B	Read Interrupt (TRV)
1C	Write Connect (TC)
1D	Write Initial Conversational Transparent (TIVX)
1E	Read Connect with Tone (TCW)
1F	Write Continue Conversational Transparent (TTVX)

Thus, byte 4 in the table of offsets contains the appropriate offset value for any device for which the Write Continue operation is valid. (Otherwise, byte 4 will contain a hexadecimal FF.)

Note: Although the position of the offset byte for an operation type is the same for all types of stations or lines, the actual offset value contained in that byte is not the same for all types of stations or lines. The offset value is a functionof the number of bytes occupied by preceding channel program offsets, which varies depending on the device involved.

All offset factors are calculated with respect to the first byte of the device I/O module.

Following the table of offsets in the device I/O module are the model channel program offsets for the device; they are contiguous, beginning immediately after byte 32 of the table.

Following the last channel command word in the start-stop device I/O module is the table of special characters for the device (e.g., EOT, N). The field contains the actual hexadecimal representations of the character sequence. For BSC devices, the table of special characters is located in the BSC interface section of the DCB.

List of Device I/O Modules

These are the BTAM device I/O module names; (they are also listed in Appendix I):

Start-Stop Terminals:

IBM 1030	IGG019MJ
IBM 1030 (Auto Poll)	IGG019MK
IBM 1050, Nonswitched	IGG019MD
IBM 1050, Nonswitched (Auto Poll)	IGG019ME
IBM 1050, Switched	IGG019MF
IBM 1060	IGG019MI
IBM 1060 (Auto Poll)	IGG019M4
IBM 2260 (Remote)	IGG019M3
IBM 2740, Basic	IGG019MT
IBM 2740, with Checking	IGG019MO
IBM 2740, with Station Control	IGG019MZ
IBM 2740, with Station Control (Auto Poll)	IGG019M2
IBM 2740, with Station Control and Checking	IGG019MY
IBM 2740, with Station Control and Checking (Auto Poll)	IGG019M1
IBM 2740, with Checking and 2760 Attachment	IGG019PL
IBM 2740, Dial-up	IGG019MU
IBM 2740, Dial-up, with Checking	IGG019MX
IBM 2740, Dial-up, with Transmit Control	IGG019MW
IBM 2740, Dial-up, with Transmit Control and Checking	IGG019MV
IBM 2740, Dial-up, with Checking and 2760 Attachment	IGG019PM
IBM 2741, Nonswitched	IGG019PE
IBM 2741, Switched	IGG019PF
AT&T 83B3	IGG019ML
WU TWX 33/35	IGG019MP
WU 115A	IGG019MN
World Trade Telegraph Terminals	IGG019PB

BSC Stations:

Nonswitched Point-to-Point (BSC1[1])	IGG019M5
Switched Point-to-Point (BSC2[1])	IGG019M6
Nonswitched Multipoint (BSC3[1])	IGG019PC

Local Station:

Local 3270 Display System	IGG019PH

Device I/O Directory

The device I/O directory, contained within module IGG019MA and initialized by the Open executor when the device I/O modules are loaded, contains the address of each device I/O module in main storage.

The format of the device I/O directory is shown as follows:

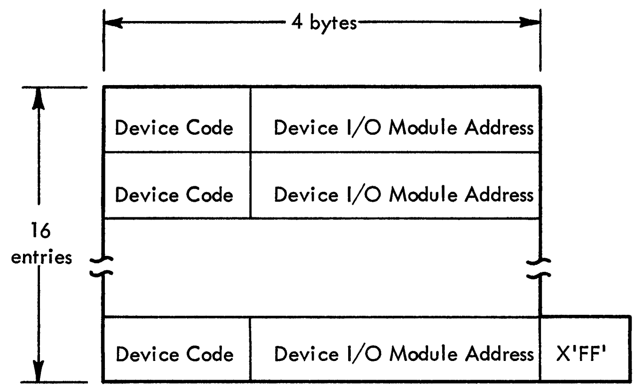


Figure 10. Format of Device I/O Directory

As each DCB is opened, the device I/O module for the corresponding type of station or line configuration is loaded into main storage, unless the required module is already present. The address of the module is placed in the first available directory word, and an index value representing the position of the entry within the directory is placed in field DCBDEVTP of the DCB. The index value for the first directory entry is 0; for the second, 1, etc.

Note: The value contained in DCBDEVTP is not a fixed code related to a particular type of station or line.

Before the index value is placed into DCBDEVTP, the contents of that field are placed into the first byte of the directory entry. This data is a fixed code for each type of station or line, and represents optional features or modes of operation, as well as identifying the type of station or line. This device code is used by the Open routine in determining whether a given device I/O module is in main storage.

----

[1]As specified in the UNIT operand of the IODEVICE macro instruction.

BTAM CHANNEL PROGRAMS -- GENERAL

This chapter shows the command codes and TP-Op codes used in the channel command words composing BTAM channel programs. The following three chapters describe the channel programs generated by the Read/Write routine for each kind of BTAM READ and WRITE macro instruction. Channel programs for start-stop operations are arranged by type of terminal supported by BTAM; channel programs for BSC operations are arranged by kind of line configuration, and apply equally to all types of the BSC stations that may be used in that line configuration. The third chapter describes channel programs for the local 3270 display system.

BTAM Channel Program Format

In the next two chapters, BTAM channel programs are represented graphically in this format:

Operation	Address	Flags	Tp-Op Code	Count

Operation is the type of operation performed by the command, along with the kind of information being sent, for a Write operation, or the kind of information expected, for a Read operation. The operation type is represented in the CCW by the command code (bits 0-7). Upon completion of execution of a command, the command code appears in the DECCMCOD field of the DECB associated with the Read or Write operation. Figure 11 gives the command code for each type of operation (including some not presently used in BTAM).

Address is the data address placed in the CCW (bits 16-31) prior to command execution. In this field,

- Area refers to the data area (buffer) address specified in the DECAREA field of the DECB.
- Table refers to the appropriate location in the table of special characters provided in the DCB (for BSC operations) or the device I/O module (for start-stop operations).
- Response (or Respn) refers to the DECRESPN field of the DECB.
- List refers to the appropriate entry in the terminal list associated with the Read or Write operation.

Flags refer to the flags turned on in the CCW:

- CD Chain Data (bit 32).
- CC Chain Command (bit 33).
- SLI Suppress length indicator (bit 34).
- Skip Suppress data transfer (bit 35).
- PCI Program Controlled Interrupt (bit 36).

TP-Op Code is the code in bits 40-47 of the generated CCW. This code is inspected by the channel end/abnormal end appendage upon the occurrence of channel end and device end.

Bits 42-47 of the code indicate the function preformed by the command; e.g., read response to addressing, write EOT sequence, or read text.

Bits 40 and 41 are set independently of bits 42-47. Bit 40 is set to 1 in the last CCW of the channel program. The Read/Write routine when modifying the basic channel program to use dynamic buffering, sets to 1 bit 41 of each of the CCWs effecting text transfer.

Figure 12 shows the TP-Op codes currently used in BTAM commands. Operations applicable to a specific type or class of remote station are so indicated.

Count is the value placed in CCW bits 48-63 prior to command execution.

- Length refers to the length of the data area (buffer) specified in the DECLNGTH field of the DECB.
- List refers to the length of the appropriate character sequence (polling characters, dial digits, etc.) in the terminal list associated with the Read or Write operation.

Command	Code (Hex)
Test I/O	00
Write	01
Read	02
I/O NOP	03
Sense	04
Diagnostic Write (Auto Wrap)	05
Prepare	06
Poll	09
Inhibit	0A
Break	0D
Search	0E
Diagnostic Read	12
Set Address 0	13
Set Address 1	17
Set Address 2	1B
Address Prepare	1E
Set Address	1F
Set Mode	23
Enable	27
Dial	29
Disable	2F
Write Break	41
Read Clear	42
Release	D4
Reserve	F4

Figure 11. Channel Command Codes

Code	Definition
00	Any command issued by on-line test routine for a local 3270 device.
01	Disable, (when disable is the first command of the channel program). Dial. Enable. Prepare. Write Pad characters. Write Wait-Before-Transmit (WACK). Sense (World Trade telegraph terminals)
02	Write EOA EOT EOT EOT (i.e., DCCC sequence prior to selection. Write EOT (C) sequence prior to polling or addressing. Write response to text. Write EOA (D) and 15 idle characters (Basic 2740). Write EOA PRF o (2740 with 2760).
03	Write polling or addressing characters. Write / (/ is the broadcast addressing character [2740 with Station Control]). Write turnaround sequence (TWX). Write identification sequence (TWX, BSC). Poll, (using SSALST, SSAWLST, AUTOLST, or AUTOWLST form of terminal list. Write Inquiry (ENQ) character.

Figure 12. TP-Op Codes for BTAM Channel Programs (Part 1 of 2)

Code	Definition
04	Write Space character (2740 with Station Control). Write 2848 command (2260 Remote). Write FIGS character (83B3). Write 1 (1030). Write WRU, Write ID, Write Pad characters, Write LTRS characters (World Trade telegraph terminal). Sense (2740).
05	Read response to polling.
06	Read response to addressing.
07	Read ID response (TWX, BSC).
08	Write EOA (D) character following end of addressing (1030, 1050, 1060, 2260 Remote, 2740). Write response to inquiry (ENQ). Write response to text (BSC). Write EOB (B) (2740 with 2760).
09	NOP or TIC following Poll command in a Read channel program using a polling list of the SSALST, SSAWLST, AUTOLST, or AUTOWLST form.
0A	Read index (Auto Poll). Read response to polling (programmed polling).
0B	Read inquiry (ENQ) character (BSC).
0C	Read response to inquiry (ENQ) character (BSC).
10	Write at display line address (2260 Remote).
11	Read text. Write text. Write frame change sequence (2740 with 2760).
12	Read skip or TIC command (dynamic buffering).
13	Write end-of-transparent text (DLE ETX) characters (BSC).
14	(Reserved)
20	Read response to text (start-stop).
21	All reset commands.
22	Read skip.
23	Write break.
24	Any command issued during OPEN, LOPEN, or CLOSE (Set Address, Enable, Disable, and Set Mode commands).
25	Read Response to text (BSC).

Figure 12. TP-Op Codes for BTAM Channel Programs (Part 2 of 2)

CHANNEL PROGRAMS FOR AT&T 83B3 SELECTIVE CALLING STATIONS

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write deselect chars (Figs., H, Ltrs.)	Table	CD	02	3
2.Write polling chars	List	CC,SLI	03	2
3.Read response	Area	CD	05	2
4.Read data	Area +2	SLI	11	Length -2

Initiated by the read/write routine, the read initial channel program places the line in control mode, polls the terminals, and reads the response to polling. If the response is positive, the response will be read into the first byte of the input area. The positive response is followed by the message. Since the read response command specifies a count of 2 (with no suppress length), the positive response followed by the message will reduce the count to zero and data chain to the read which will continue to read the data until the transmission is ended with an EOT. When a negative response is received on the read response, only one byte of data (the negative response) will be read into the message area and channel end/device end will occur (no unit exception). With the wrong length flag on and a nonzero data count, there is no data chaining to the next read command. Instead BTAM's channel end detects the polling TP-Op code and initializes for the next terminal to be polled by returning to IOS for execution with a pointer to the Write polling characters CCW. Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAG2 which is turned on when RESETPL is executed for that line. The bit is turned off by read/write when another operation on that line is initiated. Polling also terminates when the end of list (EOL) bit is detected in an open list (OPENLST).

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read Skip	Zero	SLI, Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on. The skip flag is on and the data is not read into storage.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write deselect chars (Figs., H. Ltrs.)	Table	CD	02	3
2.Write addressing chars	List	CD	03	2
3.Write Shift chars (Ltrs.)	Table	CC,SLI	04	1
4.Read response	Respn		06	1
5.Write data	Area	SLI	11	Length

The write initial channel program places the line in control mode (to allow selection of the terminals) by sending Figs., H., Ltrs., and addresses the terminal by sending two addressing characters. (The 83B3 requires a shift character after the addressing characters.) The response is read. The chaining flags of the read response command depend on the addressing list. If the user wishes to multi-address terminals (83B3), all of the terminals specified must be logically connected to the line before message transmission occurs. If a negative response is received from any terminal, the channel program is terminated and the transmission is suppressed.

If the addressing list entry specified in the WRITE macro-instruction has the EOL bit on (i.e., the entry is the last or only entry in the list), the read/write routine sets the command chaining bit on in the "read response to addressing" CCW; the multiple addressing indication is ignored and the entire channel program executes upon initiation by read/write.

If the specified addressing list entry is not the last, the third command is not altered and an interrupt occurs after the response is read. The channel end routine checks for a negative response. If the response is positive, the routine reinitializes the addressing list pointer in the CCW data address, leaves the command chaining bit off in the "read response to addressing" CCW, and returns control to the input/output supervisor for execution of commands beginning with the one containing a restart TP-Op code. This I/O supervisor-to-channel end loop continues until the end of the addressing list (EOL bit on) has been reached. On the last addressing pass, the "read response to addressing" CCW is command chained to the "write data" command.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Break Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write break	Area	SLI	23	Length

This macro-instruction causes a series of space characters to be placed on the line. The number of space characters transmitted is the number specified in the length parameter. The area address is ignored by the channel.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (Figs., H, 7, Ltrs.,)	Table	SLI	21	9

The reset command, if specified is added to the corresponding channel program (i.e., WRITE TIR). The single CCW, command chained from the last CCW in the base channel program, writes out nine special characters to transmit an end of transmission indication.

CHANNEL PROGRAMS FOR WESTERN UNION PLAN 115A OUTSTATIONS

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (Figs., H, Ltrs.)	Table	CD	02	3
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine places the line in control mode by sending the Figs., H, Ltrs., sequence, polls the terminal with the two polling characters, and reads the response.

The read response command specified a data count of 2, with wrong length indication not suppressed, whereas the length of the response is one byte. When a positive response character, and the first byte of the message are read under control of the "read response" CCW, it reduces the data count to zero and causes data chaining to take place. The rest of the message is read under control of the address and count fields of the "read data" CCW. The execution of the read continues in the channel until an interrupt occurs at the end of transmission. When, on a "read response" CCW, only one byte, a negative response is received, a channel end/device end interrupt occurs. There is no unit exception indication. The data count of 2 for a 1 byte polling response character signals wrong length which suppresses data chaining and allows BTAM to determine that a negative response was received.

The channel end routine detects the polling restart TP-Op code and reinitializes for the next terminal to be polled. Control is returned to IOS for execution of the CCWs beginning with the one containing a "restart" TP-Op code. Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAG2 of the IOB for the line being polled. This bit is set on when the RESETPL macro-instruction is executed and is set off by the read/write routine when another operation on that line is initiated.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, SKIP	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on along with the skip flag so the data is not read into storage.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (Figs., H, Ltrs.)	Table	CD	02	3
2. Write addressing chars	List	CC, SLI	03	2
3. Read response	Respn		06	1
4. Write data	Area	SLI	11	Length

The Write initial channel program, initiated by the read/write routine, places the line in control mode (which allows it to be selected), addresses the terminal, and reads the response to addressing. (The status of the chaining flags for the "read response" depends upon the status of the addressing list.) For multiple addressing, all the specified terminals must be logically connected to the line before message transmission can occur. A negative response from any terminal terminates the channel program and suppresses transmission.

If the addressing list entry specified in the WRITE macro has the end of list (EOL) bit on, the read/write routine will set the command chaining bit on the "read response" CCW. The entire channel program executes upon initiation by read/write with the multiple addressing indication ignored.

If the specified addressing list entry is not the last, the "read response" CCW is not altered and an interrupt occurs after the response is read. Channel end routine checks for a negative response which terminates the program and suppresses the transmission. If the response is positive, channel end reinitializes the addressing list pointer in the CCW data address, leaves the command chaining bit off in the "read response" CCW, and returns control to the input/output supervisor for execution of the commands beginning with the one containing the 03 TP-Op code. This loop continues until the EOL bit on is reached in the addressing list. At this time the read response CCW is command chained to the "write data" CCW.

Write Break Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write break	Area	SLI	23	Length

This macro-instruction causes a series of space characters to be placed on the line. The number of space characters, transmitted is the number specified in the length parameter. The area address is ignored by the channel.

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (Figs., H, 7 Ltrs.)	Table	SLI	21	9

The reset command, if specified is added to the corresponding channel program (i.e., WRITE TIR). The single CCW, command chained from the last CCW in the base channel program, writes out nine special characters to transmit an end of transmission indication.

CHANNEL PROGRAMS FOR IBM 1030 TERMINALS

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling character	List	CC, SLI	03	1
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read initial channel program places the line in control mode by sending three circle C, polls a terminal with one polling character, and reads the response to polling. The read response command has a data count of 2 with no suppressed length. Thus when the response (one byte) is read and it is a positive response, the response will be followed by data. This will reduce the count to zero and cause data chaining to read the rest of the data until an EOB or EOT is received or the count is zero. If the negative response is received, channel end/device end interrupt occurs with unit exception. There was no data chaining because wrong length indication and BTAM reinitializes to poll the next terminal of one was specified in the list. Polling is terminated if the channel end appendage detects a 1 in bit 7 of the IOBFLAG2 in the ICB for that line. This bit is on if a RESETPL has been executed and it is set when the read/write routine initiates another operation on that line. Polling also terminates when the end of list (EOL) bit is detected in an open list (OPENLST).

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write positive response and deselect characters (Circle Y and 3 circle Cs)	Table	CD	02	4
2. Write polling characters	List	CC, SLI	03	1
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read continue channel program sends a positive response to the previous message block, followed by three circle Cs to put the terminal in control mode. The terminal is polled with one polling character and the response is read into the first byte of the message area with a count of two. There is no suppressed length. Thus when the response is read, and it is a positive response (one byte) it will be immediately followed by data. This causes the count to be reduced to zero and data chaining to the next CCW which will read until an EOB or EOT is received or the count is zero. So if only one character is read, a negative response, a channel end/device end interrupt occurs with unit exception.

There is no data chaining due to wrong length and BTAM reinitializes the channel program to poll the next terminal, if there is one in the list. Polling is terminated if channel end detects a 1 in bit 7 of the IOBFLAG2 in the IOB for that line which indicated a RESETP2 macro-instruction was issued for this line. This bit will be turned off when the read/write routine initiates another operation on the line. Polling is terminated when the EOL bit is detected.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response and deselect characters (Circle N and 3 circle Cs)	Table	CD	02	4
2. Write polling characters	List	CC, SLI	03	1
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read repeat channel program sends a negative response N to the previous message block followed by three circle Cs to put the terminal in control mode. The terminal is then polled with one polling character. The channel program then proceeds the same way as the read initial channel program.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read Skip	Zero	SLI, Skip	22	Length

This macro-instruction causes a series of space characters to be placed on the line. The number of space characters transmitted is the number specified in the length parameter. The area address is ignored by the channel.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle Y and 3 circle C s)	Table	SLI	21	4

The reset CCW is chained to the last CCW of the basic channel program if READ TIR, TTR, or TPR is specified. This CCW sends a positive response to the transmission and three circle Cs.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. (Write 3 circle C s, Circle S) Write deselect characters	Table	CD	02	4
2. Write addressing characters	List	CD	03	1
3. Write "1"	Table	CC, SLI	04	1
4. Read addressing response	Respn	CC	06	1
5. Write Circle D	Table	CD	08	
6. Write data	Area+1	CC, SLI	11	Length
7. Read response	Respn +1		20	1

The write initial channel program sends three circle Cs, and a circle S to deselect the 1030 terminals, transmits a single addressing character followed by a 1, and reads the addressing response into the first byte of the DECRESPN in the DECB. Because multiple addressing is not possible with 1030 lines, the read response CCW is command chained to a "write circle D" CCW to send a circle D before the message. The circle D CCW data chained to write the message. This is followed by a CCW to read the response from the terminal.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write Data	Area	CC, SLI	11	Length
2. Read response	Respn +1		20	1

Initiated for the problem program through the read/write routine after a successful write initial, the write continue channel program writes data and then command chains to read the response from the terminal. This response is read into the DECRESPN+1, which is the second byte of a two-byte response field in the DECB.

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle Y and 3 circle C s)	Table	SLI	21	4

The write positive acknowledge channel program sends a Y as a positive response and three circle Cs to deselect the terminal.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

Initiated by the read/write routine, the write negative acknowledge channel program sends three circle Cs to turn the terminal off.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If a reset was specified by WRITE TIR or TTR, the reset function is performed by sending three Cs.

CHANNEL PROGRAMS FOR IBM 1050 LINES

Read Initial Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine places the line in control mode, polls a terminal, and reads the response. (Control mode is that state of the system that allows a terminal to be selected.) The third command (read response character) specifies a data count of 2, with wrong length indication not suppressed, while the length of the response character is one byte. Under the existing configuration of BTAM, the effect of this technique is as follows:

1. Positive Response: The response character and the first byte of the message are read under control of the "read response" CCW. This reduces the data count to zero and causes data chaining to take place. The second and subsequent bytes of the message are read under control of the address and count fields of the "read data" CCW. Execution continues in the channel with an interrupt occurring only at end of transmission.

2. Negative Response: This response causes channel end and device end with unit exception and wrong length record indicated. The channel end routine detects the polling restart TP-Op code, reinitializes for the next terminal to be polled, and returns control to IOS for execution of the CCWs beginning with the one containing a 03 TP-Op code.

Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAG2 of the IOB for the line being polled. This bit is set on when the RESETPL macro-instruction is executed and is set off by the read/write routine when another operation on that line is initiated.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read continue channel program is initiated by the problem program through the read/write routine after a successful read initial operation; the program writes the response character and command chains to read data.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a data check occurs during execution of the 'read data' command of a read initial or read continue operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line, but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and deselect characters (3 circle C s)	Table	SLI	21	4

If the reset function is specified by READ TIR, TTR, or TPR, then the circle D and 3 circle Cs are sent following the basic channel program.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write addressing chars	List	CC, SLI	03	2
3. Read response	Respn		06	1
4. Write circle D	Table	CD	08	1
5. Write data	Area	CC, SLI	11	Length
6. Read response to LRC	Respn +1		20	1

The write initial channel program, initiated by the read/write routine, places the line in control mode, addresses a terminal, and reads the response. The status of the chaining flags for the third command depends upon the status of the addressing list. For multiple component addressing, all specified components must be logically connected to the line before message transmission occurs. A negative response from any component terminates the channel program and suppresses transmission.

If the addressing list entry specified in the WRITE macro-instruction has the EOL bit on (i.e., the entry is the last or only entry in the list), the read/write routine sets the command chaining bit on in the "read response to addressing" CCW; the multiple addressing indication is ignored and the entire channel program executes upon initiation by read/write.

If the specified addressing list entry is not the last, the third command is not altered and an interrupt occurs after the response is read. The channel end routine checks for a negative response. If the response is positive, the routine reinitializes the addressing list pointer in the CCW data address, leaves the commandchaining bit off in the "read response to addressing" CCW, and returns control to the input/output supervisor for execution of commands beginning with the one containing a restart TP-Op code. This I/O supervisor- to-channel end loop continues until the end of the addressing list (EOL bit on) has been reached. On the last addressing pass, the "read response to addressing" CCW is command chained to the "write circle D" command. The "write data" command always chains to the "read response to LRC" command.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write Data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn +1		20	1

The write continue channel program is initiated by the problem program through the read/write routine after a successful write initial operation; the program writes data and command chains to read the response to longitudinal redundancy checking. The response is read into DECRESPN+1, the second byte of the 2-byte response field in the DECB.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to the section "Dynamic Buffering" for an explanation of the additional commands.

Write Continue Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn+1	CC, SLI	20	1
3. Write deselect characters (3 circle Cs)	Table	CD	02	3
4. Write polling characters	List	CC, SLI	03	2
5. Read response	Area	CD	05	2
6. Read data	Area+2	SLI	11	Length-2

The write continue conversational channel program is initiated by the problem program through the read/write routine after a successful write initial operation. The program writes data, command chains to read the response to longitudinal redundancy checking, command chains to place the line in control mode, polls a terminal, reads a response, and reads data on positive response.

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle D and 3 circle Cs)	Table	SLI	21	4

The acknowledge channel program, initiated by the read/write routine, transmits the positive response and three deselect characters to the terminal. The positive response character indicates to the terminal that the previous message was received without transmission errors by the CPU. After writing the circle D, the program writes the deselect characters.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

The negative acknowledge channel program, initiated by the read/write routine, writes the three deselect characters.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If the reset WRITE TIR or TTR is specified, then three circle Cs are sent following the basic channel program if no errors occurred during transmission.

CHANNEL PROGRAMS FOR IBM 1050 DIAL (SWITCHED CONNECTION LINES)

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List	CC, SLI	01	Dial List
Enable	Zero	SLI	01	1
3. Write pad characters	Table	CC, SLI	01	15
4. Write deselect characters (3 circle Cs)	Table	CD	02	3
5. Write polling characters	List	CC, SLI	03	2
6. Read response	Area	CD	05	2
7. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine disables and then either (1) dials a terminal or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the READ Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a READ Initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel-end routine restarts the channel at the Write pad characters command.

The fifteen pad characters are followed by three circle Cs to place the terminal in control mode. The two polling characters are sent. The sixth command (read response character) specifies a data count of 2, with wrong length indication not suppressed, while the length of the response character is one byte. Under BTAM, the effect of this technique is as follows:

1. Positive Response: The response character and the first byte of the message are read under control of the "read response" CCW. This reduces the data count to zero and causes data chaining to take place. The second and subsequent bytes of the message are read under control of the address and count fields of the "read data" CCW. Execution continues in the channel with an interrupt occurring only at end of transmission.

2. Negative Response: This response causes channel end and device end with unit exception and wrong length record indicated. The channel end routine detects the read response to polling TP-Op code, reinitializes for the next terminal to be polled, and returns control to IOS for execution of the CCWs beginning with the one containing a 03 TP-Op code.

Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAG2 of the IOB for the line being polled. This bit is set on when the RESETPL macro instruction is executed and is set off by the read/write routine when another operation on that line is initiated.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read continue channel program is initiated by the problem program through the read/write routine after a successful read initial operation; the program writes the response character and command chains to read data.

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

Initiated by read/write routine after the line connection has been established, the read conversational channel program places the line in control mode, polls a component of the terminal and then reads the response and data. The technique of Read commands is discussed in the 1050 Dial Read Initial Channel Program.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a transmission error occurred during execution of the 'read data' command of a previous read operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI,Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on. The skip flag is on and the data is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect character (Circle D and circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The reset if specified by READ TIR, TTR, TVR, or TPR is added to the end of the basic channel program. The sequence transmits a circle D to reply to the terminal and to halt transmission for it and an EOT character to deselect the remote terminal. It then issues a disable to disconnect the line. If no errors occurred during transmission, the reset function is performed.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List	CC, SLI	01	List
Enable	Zero	SLI	01	1
3. Write pad characters	Table	CD	01	15
4. Write deselect characters (3 circle C s)	Table	CD	02	3
5. Write addressing chars	List	CC, SLI	03	2
6. Read response to address	Respn		06	1
7. Write end-of-addressing	Table	CD	08	1
8. Write data	Area	CC, SLI	11	Length
9. Read response to LRC	Respn+1		20	1

The write initial channel program initiated by the read/write routine disables and, then either: (1) dials a terminal, or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the WRITE Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a WRITE Initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel end routine restarts the channel at the Write pad characters command.

After the pad characters, three circle Cs are sent to place the terminal in control mode. The addressing characters are sent to address the component(s). The status of the chaining flags for the sixth command depends upon the status of the addressing list. For multiple component addressing, all specified components must be logically connected to the line before message transmission occurs. A negative response from any component terminates the channel program and suppresses transmission.

If the addressing list entry specified in the WRITE macro-instruction has the EOL bit on (i.e., the entry is the last or only entry in the list), the read/write routine sets the command chaining bit on in the "read response to addressing" CCW; the multiple addressing indication is ignored and the entire channel program executes upon initiation by read/write.

If the specified addressing list entry is not the last, the third command is not altered and an interrupt occurs after the response is read. The channel end routine checks for a negative response. If the response is positive, the routine reinitializes the addressing list pointer in the CCW data address, leaves the command chaining bit off in the "read response to addressing" CCW, and returns control to the input/output supervisor for execution of commands beginning with the one containing a 03 TP-Op code. This I/O supervisor-to-channel end loop continues until the end of the addressing list (EOL bit on) has been reached. On the last addressing pass, the "read response to addressing" CCW is command chained to the "write circle D" command. The "write data" command always chains to the "read response to LRC" command.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn +1		20	1

After the line connection has previously been established, the write continue channel program is initiated by the problem program through the read/write routine; the program writes data and command chains to read the response to longitudinal redundancy checking. The response is read into DECRESPN+1, the second byte of the 2-byte response field in the DECB.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to the section "Dynamic Buffering" for an explanation of the additional commands.

Write Conversation Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect character (Circle D and 3 circle Cs)	Table	CD	02	4
2. Write addressing chars	List	CC, SLI	03	2
3. Read response	Respn		06	1
4. Write circle D	Table	CD	08	1
5. Write data	Area	CC, SLI	11	Length
6. Read response	Respn+1		20	1

The channel program transmits a circle D and 3 circle Cs with a single CCW. For a discussion of the channel program, see the "Write Initial Channel Program."

Write Continue Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn+1	CC, SLI	20	1
3. Write deselect characters (3 circle Cs)	Table	CD	02	3
4. Write polling characters	List	CC, SLI	03	2
5. Read response	Area	CD	05	2
6. Read data	Area+2	SLI	11	Length-2

The write continue conversational channel program is initiated by the problem program through the read/write routine after a successful write initial operation. The program writes data, command chains to read the response to longitudinal redundancy checking, command chains to place the line in control mode, polls a terminal, reads a response, and reads data on positive response.

Write Positive Acknowledge and Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect character (Circle D and circle C)	Table	CC, SLI	21	2
2. Disable	Zero	SLI	21	1

The write positive acknowledge and disconnect channel program resets the line after sending a positive response. The sequence transmits a circle D to reply to the terminal and to halt transmission for it and an EOT character to deselect the remote terminal. It then issues a disable to disconnect the line.

Write Negative Acknowledge and Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The write negative acknowledge and disconnect channel program sends a circle C to deselect the remote terminal and then issues a disable to disconnect the line.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If a reset was specified by WRITE TIR, TTR, or TVR, the reset function sends a circle C to deselect the terminal and then disables the line.

CHANNEL PROGRAMS FOR IBM 1060 TERMINALS

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read initial channel program places the line in control mode by sending the circle C, polls a terminal with one polling character, and reads the response to polling. The read response command has a data count of 2 with no suppressed length. Thus, when the response (one byte) is read and it is a positive response, the response will be followed by data. This will reduce the count to zero and cause data chaining to read the rest of the data until an ECB or EOT is received or the count is zero. If the negative response is received, channel end/device end interrupt occurs with unit exception. There was no data chaining because of wrong length indication and BTAM reinitializes to poll the next terminal if one was specified in the list. Polling is terminated if the channel end appendage detects a 1 in bit 7 of the IOBFLAG2 in the IOB for that line. This bit is on if a RESETPL has been executed and it is set when the read/write routine initiates another operation on that line. Polling also terminates when the end of list (EOL) bit is detected in an open list (OPENLST).

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle Y and 3 circle C s)	Table	CD	02	4
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read continue channel program sends a positive response to the previous message block, followed by three circle Cs to put the terminal in control mode. The terminal is polled with one polling character and the response is read into the first byte of the message area with a count of two. There is no suppressed length. Thus, when the response is read, and it is a positive response (one byte) it will be immediately followed by data. This causes count to be reduced to zero and data chaining to the next CCW which will read until an ECB or ECT is received or the count is zero. If only one character is read, a negative response, a channel end/device end interrupt occurs with unit exception. There is no data chaining due to wrong length and BTAM reinitializes the channel program to poll the next terminal, if there is one in the list. Polling is terminated if channel end detects a 1 in bit 7 of the ICBFLAG2 in the IOB for that line, which indicated a RESETPL macro-instruction was issued for this line. This bit will be turned off when the read/write routine initiates another operation on the line. Polling is terminated when the ECL bit is detected.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response and deselect characters (Circle N and 3 circle C s)	Table	CD	02	4
2. Write polling characters	List	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read repeat channel program sends a negative response N to the previous message block followed by three circle Cs to put the terminal in control mode. The terminal is then polled with one polling character. The channel program then proceeds the same way as the read initial channel program.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI,Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on along with the skip flag so the data is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write positive response and deselect characters (Circle Y and 3 circle Cs)	Table	SLI	21	4

The reset CCW is chained to the last CCW of the basic channel program if READ TIR, TTR, or TPR is specified. This CCW sends a positive response to the transmission and three circle Cs.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write addressing chars	List	CC, SLI	03	2
3. Read response	Respn		06	1
4. Write circle D	Table	CD	08	1
5. Write data	Area	CC, SLI	11	Length
6. Read response to LRC	Respn +1	SLI	20	1

The write initial channel program, initiated by the read/write routine, places the line in control mode, addresses a terminal, and reads the response. The status of the chaining flags for the third command depends upon the status of the addressing list. For multiple addressing, all specified terminals must be logically connected to the line before message transmission occurs. A negative response from any terminal terminates the channel program and suppresses transmission.

If the addressing list entry specified in the WRITE macro-instruction has the ECL bit on (i.e., the entry is the last or only entry in the list), the read/write routine sets the command chaining bit on in the "read response to addressing" CCW; the multiple addressing indication is ignored and the entire channel program executes upon initiation by read/write.

If the specified addressing list entry is not the last, the third command is not altered and an interrupt occurs after the response is read. The channel end routine checks for a negative response. If the response is positive, the routine reinitializes the addressing list pointer in the CCW data address, leaves the command chaining bit off in the "read response to addressing" CCW, and returns control to the input/output supervisor for execution of a restart TP-Op code. This I/O supervisor-to-channel end loop continues until the end of the addressing list (ECL bit on) has been reached. On the last addressing pass, the "read response to addressing" CCW is command chained to the "write data" command. The "write data" command always chains to the "read response to LRC" command.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle Y and 3 circle C s)	Table	SLI	21	4

The write positive acknowledge channel program sends a Y as a positive response and three circle Cs to deselect the terminal.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

Initiated by the read/write routine, the write negative acknowledge channel program sends three circle Cs to turn the terminal off.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If the reset was specified by WRITE TIR, then the three circle Cs are sent to deselect the terminal.

CHANNEL PROGRAMS FOR TTY MODELS 33 AND 35 TWX LINES

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	SLI	01	1
3. Write pad characters	Table	CD	01	15
4. Write identification	List	CC, SLI	03	List
5. Read data	Area	SLI	11	Length

The read initial channel program, initiated by the read/write routine performs the following:

Disables the line in case not done previously.

Sets the enable latch within the line adapter so that the remote terminal may dial the CPU.

Sends pad characters.

Writes the identification assigned to the CPU by the user's answer list.

Reads the data transmitted by the terminal.

After the enable command has been issued and the terminal has dialed the CPU, a channel end/device end interrupt occurs and the channel end appendage turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of I/O operations that data transmission has begun. If the bit is not turned on in the UCB after a read initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel end appendage restarts the channel at the fourth command (write identification).

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write turnaround char	List	CC, SLI	03	List
2. Read data	Area	SLI	11	Length

The read conversational channel program is initiated by the problem program through the read/write routine after a successful write initial operation.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line, but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If READ TIR or TVR is specified, the reset function is added to the end of basic channel program and is executed only if the basic channel program completed without error. The reset will send a circle C to deselect the terminal and then disable the line.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List	CC, SLI	01	List
3. Read identification	List		07	List
4. Write data	Area	SLI	11	Length

The write initial channel program, initiated by the read/write routine, disables and dials a terminal and, if the identification received was valid, writes the data to the terminal. If the identification was invalid, the channel program is terminated.

After the CPU has read the identification sent from the terminal, an interrupt occurs and the channel end routine compares the identification received with the identification supplied by the user in the dial list. If an unequal compare results, the channel end routine posts the buffer; if dynamic buffering was specified. The event is posted complete. If an equal identification is received, it is assumed the correct terminal has been contacted and the channel is restarted with the write data command.

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	SLI	11	Length

The write conversational channel program is initiated by the problem program throughthe read/write routine after a successful read initial operation.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The write negative acknowledge channel program, initiated by the read/write routine, writes an end-of-transmission character to the terminal and resets, with a disable command, the enable latch within the line adapter.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT	Table	CC, SLI	Reset	1
2. Disable	Area	SLI	Reset	1

If WRITE TIR or TVR is specified, the reset function is added to the end of the basic channel programs. The program writes an end-of-transmission character to the terminal and resets, with a disable command, the enable latch within the line adapter.

CHANNEL PROGRAMS FOR IBM 2740 COMMUNICATIONS LINES

IBM 2740 BASIC CHANNEL PROGRAMS

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (1 circle C)	Table	CC, SLI	21	1
2. Prepare	Zero	CC, SLI	01	1
3. Sense	Respn+1	CC, SLI	04	1
4. Read data	Area	SLI	11	Length

The read initial channel program places the terminal in control mode and sends the prepare command to condition the control unit to receive a message from the terminal. The prepare command removes the circle D from the beginning of the message and the count is reduced to zero, causing command chaining to the sense command and to the read command that reads the message.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on. The skip flag is on and the data is not read into storage.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and 15 idle characters	Table	CD	02	16
2. Write data	Area	SLI	11	Length

The write initial channel program sends a circle D and fifteen idle characters and data chains to the write data to send the message.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If reset was specified by WRITE TIR, the reset function will send the three circle Cs if no errors occurred during the previous transmission.

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (1 circle C)	Table	CC, SLI	02	1
2. Prepare	Zero	CC, SLI	01	1
3. Sense	Respn+1	CC, SLI	04	1
4. Read data	Area	SLI	11	Length

The Prepare command conditions the control unit to receive a message and then command chains to the Read command when a character is received. The circle D sent by the transmitting terminal is deleted by the Prepare command.

Read Continue Channel Program

Operation	Address	Flags	›-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read continue channel program is initiated by the problem program through the read/write routine after a successful read initial operation; the program writes the response character and command chains to read data.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a transmission error occurred during execution of the 'read data' command of a read initial or read continue operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on. The skip flag is on and the data is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and deselect characters (3 circle C s)	Table	SLI	21	4

If the reset is specified by READ TIR, TTR, or TPR, the reset function will send the circle D and three circle Cs if no errors occurred on the data transmissions.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and 15 idle characters	Table	CD	02	16
2. Write data	Area	CC, SLI	11	Length
3. Read response to VRC/LRS	Respn +1		20	1

The write initial channel program sends the circle D to put the terminal in control mode and fifteen idle characters to allow terminal motors to get up to speed. This Write command data chains to the Write data command which sends the message and command chains to the Read response command.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to VRC/LRC	Respn +1		20	1

The write continue channel program will write the data and then command chain to read the response into DECRESPN+1 (VRC/LRC response field in the DECB).

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D	Table	CD	02	1
2. Write data	Area	CC, SLI	11	Length
3. Read response to VRC/LRC	Respn +1		20	1

The write conversational channel program first writes a circle D to put the terminal in receive mode and then data chains to the next write to send the data. When the count is zero, this command chains to read the VRC/LRC response into the response field of the DECB (DECRESPN+1).

Write Continue Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn+1	CC, SLI	20	1
3. Write deselect character (1 circle C)	Table	CC, SLI	02	1
4. Prepare	Zero	CC, SLI	01	1
5. Sense	Zero	CC, SLI	04	1
6. Read data	Area	SLI	11	Length

The write continue conversational channel program is initiated by the problem program through the read/write routine after a successful write initial operation; the program writes data, command chains to read the response to longitudinal redundancy checking, command chains to place the line in control mode, command chains to the prepare command which removes the circle D from the beginning of the message, and command chains to read the message.

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and deselect characters (3 circle C s)	Table	SLI	21	4

The write positive acknowledge channel program sends a circle D as a positive response and three circle Cs to deselect the terminal.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

Initiated by the read/write routine, the write negative acknowledge channel program sends three circle Cs to turn the terminal off.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If the reset is specified by WRITE TIR, TTR or TVR, then the three circle Cs are sent to turn the terminal motors off. The reset will be performed only if no errors occurred during the write for this reset.

IBM 2740 WITH CHECKING AND 2760 ATTACHMENT

In this publication, the term "response" is used instead of "answerback" to signify response to LRC/VRC check, to maintain consistency with the other 2740 channel program explanations. In the OS BTAM SRL publication, the term answerback is used, to maintain consistency with the IBM 2760 Optical Image Unit Component Description publication (GA27-3011). The two terms are synonymous.

Read Initial Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOT EOT EOT	Table	CC,SLI	21	3
2. Prepare	0	CC,SLI	01	1
3. Sense	Respn	CC,SLI	04	1
4. Read text	Area	SLI	11	Length

In the Read Initial channel program, the Write command sets the terminal to standby status; the Prepare command monitors the line for an incoming EOA; when EOA is received, the sense command receives a sense byte into the DECB response field, and the Read text command receives message text.

Read Continue Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write response (Circle Y)	Table	CC,SLI	02	1
2. Read text	Area	SLI	11	Length

The Read Continue channel program sends a positive response and receives another message block.

Read Continue with Leading Acknowledgment Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA EOT EOT EOT	Table	CC,SLI	21	4
2. Prepare	0	CC,SLI	01	1
3. Sense	Respn	CC,SLI	04	1
4. Read Text	Area	SLI	11	Length

The Read Continue with Leading Acknowledgment channel program sends EOA as positive response to text and an EOT sequence to place the terminal in standby status; prepares the TCU data adapter to receive text; receives a sense byte into the DECB response field; and receives message text. (The EOA with which the terminal precedes message text is deleted by the Prepare command.)

Read Repeat Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write response (circle N)	Table	CC,SLI	02	1
2. Read text	Area	SLI	11	Length

The Read Repeat channel orogram sends a negative response and receives another message block (which should be the same text as received by the previous Read operation).

Read Skip Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Read skip	0	SLI,Skip	22	Length

The Read Skip channel program receives a message block but does not place it in main storage.

Reset Following a Read

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA EOT EOT EOT	Table	SLI	21	4

The Reset function, specified by the READ TIR, TTR, or TPR macro, is added to the end of the basic channel program (but is executed only if the text was received correctly). The EOA signifies positive response to text; the EOTs reset the terminal to standby status.

Write Initial Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA and 15 Idle characters	Table	CD	02	16
2. Write text	Area	CC,SLI	11	Length
3. Read response	Respn+1		20	1

The Write Initial channel program sends EOA to place the terminal in receive status and 15 idle characters to allow terminal motors to reach operating speed; sends text, and receives response to text in DECRESPN+1.

Write Continue Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Read response	Respn+1		20	1

The Write Continue Channel program sends a message block and receives response to text in DECRESPN+1.

When dynamic buffering is used, three additional commands are generated and inserted in the channel program following the Write Text command. See the section Dynamic Buffering for an explanation of these commands.

Write Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA	Table	CD	02	1
2. Write text	Area	CC,SLI	11	Length
3. Read response	Respn+1		20	1

The Write Conversational channel program sends EOA to place the terminal in receive status, sends text, and reads response to text.

Write Initial Optical Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA PRE o	Table	CD	02	3
2. Write frame change sequence	List	CD	11	3
3. Write EOB	Table	CC,SLI	08	1
4. Read response	Respn +1	CC,SLI	20	1
5. Write EOT EOT EOT	Table	SLI	21	3

The Write Initial Optical channel program sends EOA PRE o to place terminal in receive status and indicate that the subsequent message text is for the 2760; sends a frame change message and EOB; reads response to text; and resets the terminal to standby status if the response is positive (the EOT sequence also causes the 2760 to execute the frame change function).

Write Invitational Optical Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA PRE o	Table	CD	02	3
2. Write frame change sequence	Entry	CD	11	3
3. Write EOB	Table	CC,SLI	08	1
4. Read response	Respn+1	CC,SLI	20	1
5. Write EOT EOT EOT	Table	CC,SLI	21	3
6. Prepare		CC,SLI	01	1
7. Sense	Respn	CC,SLI	04	1
8. Read text	Area	SLI	11	Length

The Write Invitational Optical channel program sends EOA PRE o to place the terminal in receive status and indicate that the subsequent message text is for the 2760; sends a frame change message and EOB; reads response to text; resets the terminal to standby status if response is positive (the EOT sequence also causes the 2760 to execute the frame change function); prepares the TCU data adapter to receive text; receives a sense byte into the DECB response field; and reads message text. (The EOA with which the terminal precedes message text is deleted by the Prepare command.)

Write Positive Acknowledgment Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA EOT EOT EOT	Table	SLI	21	4

The Write Positive Acknowledgment channel program sends EOA as positive response to text and three EOTs to place the terminal in standby status.

Write Negative Acknowledgment Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOT EOT EOT	Table	SLI	21	3

The Write Negative Acknowledgment channel program sends three EOTs to place the terminal in standby status; the EOTs also signify negative response to text, if the channel program is executed following a Read operation.

Reset Following a Write

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOT EOT EOT	Table	SLI	21	3

The Reset function, specified by the WRITE TIR, TTR, or TVR, is executed at the end of the basic channel program (only if a positive response to text was received by the preceding command). The EOTs reset the terminal to standby status.

IBM 2740 WITH DIAL

Read Initial Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	CC, SLI	01	1
3. Prepare	Zero	CC, SLI	01	1
4. Read data	Area	SLI	11	Length

Initiated by the read/write routine, the read initial channel program disables and then enables the line to receive a call. When a call is received, the Enable command chains to the Prepare command which conditions the control unit to receive a message. When a character is received, the count goes to zero and the Prepare command command chains to read the data. The Prepare command deletes the circle D which is sent by the depression of the BID key at the transmitting terminal.

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Prepare	Zero	CC, SLI	01	1
2. Read	Area	SLI	11	Length

The read conversational channel program sends the Prepare command to condition the control unit to receive a message from a terminal. The Prepare command removes the circle D from the beginning of the message and the count is reduced to zero, causing command chaining to the read which reads the message.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

The read skip channel program reads the data from the terminal under the control of the count field. The suppress length flag is on. The skip flag is on and the data is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If reset is specified by READ TIR or TVR, the reset commands are added to the basic channel program and are executed if no errors occurred during the execution of the basic channel program.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List + 1	CC, SLI	01	List
3. Write pad characters	Table	CC, SLI	01	15
4. Write circle D	Table	CD	02	1
5. Write data	Area	SLI	11	Length

Initiated by the read/write routine, the write initial channel program disables the line and command chains to the Dial command to dial the terminal specified by the DFTRMLST macro-instruction. After dialing, the channel program sends fifteen pad characters before command chaining to a write circle D command which is sent before the data.

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D	Table	CD	02	1
2. Write data	Area	SLI	11	Length

The write conversational channel program sends a circle D and then data chains when the count is zero to a write data command to send the message.

Write Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The write disconnect channel program sends a circle C to put the terminal in control mode and command chains to disable the line.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (CircleC)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If reset is specified by WRITE TIR or TVR, the reset commands are added to the corresponding channel programs. They are executed when the basic channel program completes without error.

IBM 2740 WITH DIAL AND CHECKING

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	CC, SLI	01	1
3. Prepare	Zero	CC, SLI	01	1
4. Read data	Area	SLI	11	Length

Initiated by the read/write routine, the read initial channel program disables and then enables the line to receive a call. When a call is received, the Enable command chains to the Prepare command which conditions the control unit to receive a message. When a character is received, the count goes to zero and the Prepare command command chains to read the data. The Prepare command deletes the circle D which is sent by the depression of the BID key at the transmitting terminal.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read continue channel program is initiated by the problem program through the read/write routine after a successful read initial operation; the program writes the response character and command chains to read data.

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write 3 circle C s	Table	CC, SLI	02	3
2. Prepare	Zero	CC, SLI	01	1
3. Read	Area	SLI	11	Length

Initiated by the read/write routine, the read conversational channel program sends three circle Cs to put the terminal in control mode and then chains to the Prepare command. The Prepare command conditions the control unit to receive a message. When the first character, a circle D caused by the depression of the transmitting terminal's BID key, is received, it is deleted by the Prepare command which chains to read the data.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a transmission error occurred during execution of the 'read data' command of a previous read operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Read skip	Zero	SLI,SKIP	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1.Write response and deselect chars (Circle D and circle C)	Table	CC,SLI	21	2
2.Disable	Zero	SLI	21	1

The reset if specified by READ TIR, TTR, TVR, or TPR is added to the end of the basic channel program. The sequence transmits a circle D to reply to the terminal and to halt transmission for it and an EOT character to deselect the remote terminal. It then issues a disable to disconnect the line. If no errors occured during transmission, the reset function is performed.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List	CC, SLI	01	List
3. Write Pad Characters	Table	CC, SLI	01	15
4. Write circle D	Table	CD	02	1
5. Write data	Area	CC, SLI	11	Length
6. Read response to VRC/LRC	Respn+1		20	1

Initiated by the read/write routine, the write initial channel program disables the line before dialing the terminal specified by the DFTRMLST. The fifteen pad characters are sent to allow the terminal motors to reach the necessary speed before the message is sent to it. Before the data is sent, a circle D is sent to the terminal. After the message is sent the response to VRC/LRC is read into the response field in the DECB (DECRESPN+1).

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn +1		20	1

After the line connection has previously been established, the write continue channel program is initiated by the problem program through the read/write routine; the program writes data and command chains to read the response to longitudinal redundancy checking. The response is read into DECRESPN+1, the second byte of the 2-byte response field in the DECB.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to the section "Dynamic Buffering" for an explanation of the additional commands.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT EOT EOT	Table	CC, SLI	21	3
2. Disable	Zero	SLI	21	1

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D	Table	CD	02	1
2. Write data	Area	CC, SLI	11	Length
3. Read response to VRC/LRC	Respn +1		20	1

The Write conversational channel program sends a circle D after the line has previously been established. The data is sent and the "Write data" command chains to the "read response" CCW.

Write Continue Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn+1	CC, SLI	20	1
3. Write deselect character (1 circle C)	Table	CC, SLI	02	1
4. Prepare	Zero	CC, SLI	01	1
5. Sense	Zero	CC, SLI	04	1
6. Read data	Area	SLI	11	Length

The write continue conversational channel program is initiated by the problem program through the read/write routine after a successful write initial operation; the program writes data, command chains to read the response to longitudinal redundancy checking, command chains to place the line in control mode, command chains to the prepare command which removes the circle D from the beginning of the message, and command chains to read the message.

Write Positive Acknowledge and Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect character (Circle D and circle C)	Table	CC, SLI	21	2
2. Disable	Zero	SLI	21	1

The Write positive acknowledge and disconnect channel program resets the line after sending a positive response. The sequence transmits a circle D to reply to the terminal and to halt transmission for it and an EOT character to deselect the remote terminal. It then issues a disable to disconnect the line.

Write Negative Acknowledge and Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

IBM 2740 WITH DIAL, CHECKING, AND 2760 ATTACHMENT

In this publication, the term "response" is used instead of "answerback" to signify response to LRC/VRC check, to maintain consistency with the other 2740 channel program explanations. In the OS BTAM SRL publication, the term answerback is used to maintain consistency with the IBM 2760 Optical Image Unit Component Description (Form A27-3011). The two terms are synonymous.

Read Initial Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	0	CC,SLI	01	1
2. Enable	0	CC,SLI	01	1
3. Prepare	0	CC,SLI	01	1
4. Sense	Respn	CC,SLI	04	1
5. Read text	Area	SLI	11	Length

The Read Initial channel program disables the line, then enables it to receive a call. When a call is received, the Prepare command deletes the EOA sent by the Bid key at the terminal, the Sense command receives a sense byte into the DECB response field, and the Read text command receives message text.

Read Continue Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write response (circle Y)	Table	CC,SLI	02	1
2. Read text	Area	SLI	11	Length

The Read Continue channel program sends a positive response and receives another message block.

Read Continue with Leading Acknowledgment Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA EOT EOT EOT	Table	CC,SLI	21	4
2. Prepare	0	CC,SLI	01	1
3. Sense	Respn	CC,SLI	04	1
4. Read text	Area	SLI	11	Length

The Read Continue with Leading Acknowledgment channel program sends EOA as positive response to text and an EOT sequence to place the terminal in standby status; prepares the TCU data adapter to receive text; receives a sense byte into the DECB response field; and receives message text. (The EOA with which the terminal precedes message text is deleted by the Prepare command.)

Read Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOT EOT EOT	Table	CC,SLI	02	3
2. Prepare	0	CC,SLI	01	1
3. Sense	Respn	CC,SLI	04	1
4. Read text	Area	SLI	11	Length

The Read Conversational channel program first places the terminal in standby status. The Prepare command deletes the EOA sent by the Bid key at the terminal, the Sense command receives a sense byte into the DECB response field, and the Read Text command receives message text.

Read Repeat Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write response (circle N)	Table	CC,SLI	02	1
2. Read text	Area	SLI	11	Length

The Read Repeat channel program sends a negative response and receives another message block (which should be the same text as received by the previous Read operation).

Read Skip Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Read skip	0	SLI,Skip	22	Length

The Read Skip channel program receives a message block but does not place it in main storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write EOA EOT EOT EOT	Table	CC, SLI	21	4
2. Disable	0	SLI	21	1

The Reset function, specified by the READ TIR, TTR, TVR, or TPR, is executed at the end of the basic channel program (only if the text was received correctly). The first command sends an EOA and 3 EOTs to signify positive response to text and to reset the terminal to standby status. The second command breaks the line connection.

Write Initial Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	0	CC,SLI	01	1
2. Dial	List	CC,SLI	01	List
3. Write pad characters	Table	CC,SLI	01	15
4. Write EOA	Table	CD	02	1
5. Write text	Area	CC,SLI	11	Length
6. Read response	Respn+1		20	1

The Write Initial channel program disables the line, dials a remote terminal, sends 15 pad characters to allow terminal motors to reach operating speed, sends EOA to place terminal in receive status, sends text, and receives response to text in DECRESPN+1.

Write Continue Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Read response	Respn+1		20	1

The Write Continue Channel program sends a message block and receives response to text in DECRESPN+1.

When dynamic buffering is used, three additional commands are generated and inserted in the channel program following the Write Text command. See the section Dynamic Buffering for an explanation of these commands.

Write Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA	Table	CD	02	1
2. Write text	Area	CC,SLI	11	Length
3. Read response	Respn+1		20	1

The Write Conversational channel program sends EOA to place the terminal in receive status, sends text, and reads response to text.

Write Initial Optical Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	0	CC,SLI	01	1
2. Dial	List	CC,SLI	01	List
3. Write pad characters	Table	CC,SLI	01	15
4. Write EOA PRE o	Table	CD	02	3
5. Write frame change sequence	List	CD	11	3
6. Write EOB	Table	CC,SLI	08	1
7. Read response	Respn+1	CC,SLI	20	1
8. Write EOT EOT EOT	Table	SLI	21	3

The Write Initial Optical channel program disables the line; dials a remote terminal; sends 15 pad characters to allow terminal motors to reach operating speed; sends EOA PRE o to place terminal in receive status and indicate that the subsequent message text is for the 2760; sends a frame change message and EOB; reads response to text; and resets the terminal to standby status if the response is positive. (The EOT sequence also causes the 2760 to execute the frame change function.)

Write Invitational Optical Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write EOA PRE o	Table	CD	02	3
2. Write frame change sequence	Entry	CD	11	3
3. Write EOB	Table	CC, SLI	08	1
4. Read Response	Respn+1	CC, SLI	20	1
5. Write EOT EOT EOT	Table	CC, SLI	02	3
6. Prepare	0	CC, SLI	01	1
7. Sense	Respn	CC, SLI	04	1
8. Read text	Area	SLI	11	Length

The Write Invitational Optical channel program sends EOA PRE o to place the terminal in receive status and indicate that the subsequent message text is for the 2760; sends a frame change message and EOB; reads response to text; resets the terminal to standby if response is positive (the EOT sequence also causes the 2760 to execute the frame change function); prepares the TCU data adapter to receive text; receives a sense byte into the DECB response field; and reads message text. (The EOA with which the terminal precedes message text is deleted by the Prepare command.)

Write Conversational Optical Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA PRE o	Table	CD	02	3
2. Write frame change sequence	Entry	CD	11	3
3. Write EOB	Table	CC,SLI	08	1
4. Read response	Respn+1	CC,SLI	20	1
5. Write EOT EOT EOT	Table	SLI	21	3

The Write Conversational Optical channel program sends EOA PRE o to place the terminal in receive status and indicate that the subsequent message text is for the 2760; sends a frame change message and EOB; reads response to text; and resets the terminal to standby status if the response is positive. (The EOT sequence also causes the 2760 to execute the frame change function.)

Write Positive Acknowledgment Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOA EOT EOT EOT	Table	CC,SLI	21	4
2. Disable	0	SLI	21	1

The Write Positive Acknowledgment channel program sends EOA as positive response to text and three EOTs to place the terminal in standby status, then breaks the line connection.

Write Negative Acknowledgment Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT EOT EOT	Table	CC, SLI	21	3
2. Disable	Zero	SLI	21	1

The Write Negative Acknowledgment channel program sends three EOTs to place the terminal in standby status (the EOTs also signify negative response to text, if the channel program is executed following a Read operation), then breaks the line connection.

IBM 2740 WITH DIAL AND TRANSMIT CONTROL

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Dial	List	CC, SLI	01	Dial List
Enable	Zero	SLI	01	1
3. Write pad characters	Table	CC, SLI	01	15
4. Write selection characters	Table	CC, SLI	03	2
5. Read response	Area	CD	05	2
6. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine disables and then either (1) dials a terminal or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the Read Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a read initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel-end routine restarts the channel at the Write pad characters command.

The fifteen pad characters are followed by the selection character (/ space) to select the fifth command (read response character). The read response CCW specifies a data count of 2, with wrong length indication not suppressed, while the length of the response character is one byte. Under BTAM, the effect of this technique is as follows:

1. Positive Response: The response character, a circle D caused by the depression of the transmitting terminals BID key, and the first byte of the message are read under control of the "read response" CCW. This reduces the data count to zero and causes data chaining to take place. The second and subsequent bytes of the message are read under control of the address and count fields of the "read data" CCW. Execution continues in the channel with an interrupt occurring only at the end of the transmission.

2. Negative Response: This response causes channel end and device end with unit exception and wrong length record indicated. There is no polling of component(s) or terminal(s) on the 2740DT; only the sending of the selection characters.

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write selection chars (/ space)	Table	CC, SLI	03	2
2. Read response	Area	CD	05	2
3. Read data	Area+2	SLI	11	Length-2

The read conversational channel program sends the selection characters (1 space) to allow the terminal to send a message. The response is read into the first two bytes of the message area. (See the Read Initial section of the 2740DT for a discussion of the positive or negative response.) If the response was positive the "read response" CCW data chains to read the rest of the message.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If reset is specified by READ TIR or TVR, the reset commands are added to the basic channel program and are executed if no errors occurred during the execution of the basic channel program.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	SLI	01	1
Dial	List	CC, SLI	01	List
3. Write pad characters	Table	CC, SLI	01	15
4. Write circle D	Table	CD	08	1
5. Write data	Area	SLI	11	Length

The write initial channel program initiated by the read/write routine disables and, then either: (1) dials a terminal, or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the WRITE Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a write initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel end routine restarts the channel at the Write pad characters command.

After the pad characters, the channel program data chains to send a circle D before sending the data.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D	Table	CD	08	1
2. Write data	Area	SLI	11	Length

The write conversational channel program sends a circle D and then data chains when the count is to a Write data command to send the message.

Write Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The Write disconnect channel program sends a circle C to put the terminal in control mode and command chains to disable the line.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect character (Circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If reset is specified by WRITE TIR or TVR, the reset commands are added to the corresponding channel programs. They are executed when the basic channel program completes without error.

IBM 2740 WITH DIAL, TRANSMIT CONTROL, AND CHECKING

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	SLI	01	1
Dial	List	CC, SLI	01	List
3. Write pad characters	Table	CC, SLI	01	15
4. Write selection chars (/ space)	Table	CC, SLI	03	2
5. Read response	Area	CD	05	2
6. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine disables and then either (1) dials a terminal or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the Read Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a read initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel-end routine restarts the channel at the Write pad characters command.

The fifteen pad characters are followed by the selection character (/ space) to select the fifth command (read response character) specifies a data count of 2, with wrong length indication not suppressed, while the length of the response character is one byte. Under BTAM, the effect of this technique is as follows:

1. Positive Response: The response character, a circle D caused by the depression of the transmitting terminals BID key, and the first byte of the message are read under control of the "read response" CCW. This reduces the data count to zero and causes data chaining to take place. The second and subsequent bytes of the message are read under control of the address and count fields of the "read data" CCW. Execution continues in the channel with an interrupt occurring only at the end of the transmission.

2. Negative Response: This response causes channel end and device end with exception and wrong length record indicated. There is no polling of component(s) or terminal(s) on the 2740DT; only the sending of the selection characters.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

After the line connection has previously been established, the read continue channel program is initiated by the problem program through the read/write routine; the program writes the response character and command chains to read data.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a data check occurred during execution of the 'read data' command of a read initial or read continue operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write 3 circle C s	Table	CD	02	3
2. Write selection char (/ space)	Table	CC, SLI	03	2
3. Read response	Area	CD	05	2
4. Read data	Area+2	SLI	11	Length-2

The read conversational channel program is initiated by the read/write routine. The channel program sends the three circle Cs to put the terminal in control mode prior to sending the selection characters which tells the terminal it can now send. The response is read into the first two bytes of the message area. If the response is negative, only one byte, an interrupt occurs and the channel end routine recognizes that is is a negative response and the channel program is terminated. If the response is positive, it will be followed by a data. After the first two bytes are read, the count is reduced to zero and this CCW data chains to finish reading the message.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read Skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect character (Circle D and circle C)	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

If READ TIR, TTR, TPR, or TVR is specified, the reset function is added to the end of basic channel program and is executed only if the basic channel program completed without error. The reset will send a circle D and a circle C to deselect the terminal and then disable the line.

If there is dynamic buffering, three additional commands are generated in the channel program following the Write data command. See the section "Dynamic Buffering" for an explanation of the additional commands.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Disable	Zero	CC, SLI	01	1
2. Enable	Zero	SLI	01	1
Dial	List	CC, SLI	01	List
3. Write pad characters	Table	CC, SLI	01	15
4. Write circle D	Table	CD	08	1
5. Write data	Area	SLI	11	Length
6. Read response to VRC/LRC	Respn+1		20	1

The write initial channel program initiated by the read/write routine disables and, then either: (1) dials a terminal, or (2) sets the enable latch within the line adapter so that the remote terminal may dial the CPU. The selection of dial or enable depends upon whether the problem program provides a dial list or an answer list when the WRITE Initial macro-instruction is issued.

If the enable command is issued and the terminal dials the CPU, an interrupt occurs and the channel end routine turns on the arm-seeking bit in the seventh byte of the UCB. This is an indication to any subsequent halting of the I/O operation that data transmission has begun. If the bit is not turned on in the UCB after a write initial macro-instruction is given, the terminal has not dialed the CPU. After turning on the arm-seeking bit, the channel end routine restarts the channel at the Write pad characters command.

After the pad characters, the channel program data chains to send a circle D before sending the data. The response to VRC/LRC is read into the second byte of the field in the DECB (DECRESPN+1).

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to VRC/LRC	Respn +1		20	1

The write continue channel program will write the data and then command chain to read the response into DECRESPN+1 (VRC/LRC response field in the DECB.

Write Conversational Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write Circle D	Table	CD	02	1
2. Write data	Area	CC, SLI	11	Length
3. Read response to VRC/LRC	Respn +1		20	1

The write conversational channel program first writes a circle D to put the terminal in receive mode and then data chains to the next write to send the data. When the count is zero, this command chains to read to VRC/LRC response into the response field of the DECB (DECRESPN+1) .

Write Positive Acknowledge and Disconnect Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and deselect character (Circle C)	Table	CC, SLI	21	2
2. Disable	Zero	SLI	21	1

The Write positive acknowledge channel program sends a circle D as a positive response followed by a circle D to place the terminal in control mode. The line is then disabled.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT	Table	CC, SLI	21	1
2. Disable	Zero	SLI	21	1

The Write negative acknowledge channel program initiated by the read/write routine, writes an end-of-transmission character to the terminal and resets, with a disable command, the enable latch within the line adapter.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT	Table	CC, SLI	Reset	1
2. Disable	Area	SLI	Reset	1

If WRITE TIR, TTR, or TVR is specified, the reset function is added to the end of the basic channel programs. The program writes an end-of-transmission character to the terminal and resets, with a disable command, the enable latch within the line adapter.

IBM 2740 WITH STATION CONTROL

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling character	List	CD	03	1
3. Write space character	Table	CC, SLI	04	1
4. Read response	Area	CD	05	2
5. Read data	Area +2	SLI	11	Length-2

Initiated by the read/write routine, the read initial channel program places the line in control mode, polls the terminals, with one character followed by a space character, and reads the response to polling. If the response is positive, the response will be read into the first byte of the input area. The positive response is followed by the message. Since the read response command specifies a count of 2 (with no suppress length), the positive response followed by the message will reduce the count to zero and data chain to the read will continue to read the data until the transmission is ended with an EOT. When a negative response is received on the read response, only one byte of data (the negative response) will be read into the message area and channel end/device end occurs (no unit exception). With the wrong length flag on and a nonzero data count, there is no data chaining to the next read command. Instead BTAM'S channel end detects the polling TP-Op code and initializes for the next terminal to be polled by returning to IOS for execution with a pointer to the Write polling characters CCW. Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAGS which is turned on when RESETPL is executed for that line. The bit is turned off by read/write when another operation on that line is initiated. Polling also terminates when the end of list (EOL) bit is detected in an open list (OPENLST).

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle Cs and circle S)	Table	CD	02	4
2. Write addressing character	List	CD	03	1
3. Write space character	Table	CC, SLI	04	1
4. Read response	Respn	SLI	06	2
5. Write circle D	Table	CD	04	1
6. Write data	Area	SLI	11	Length

The write initial channel program places the terminal in control mode and sends a circle S to denote that addressing will follow. The terminal is addressed with a one character code followed by a space character. The response to addressing is read into the first byte of the response field in the DECB (DECRESPN). The Write circle D "CCW" data chains to write the data.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	SLI	21	3

If the reset is specified by WRITE TIR, then the three circle Cs are sent to turn the terminal motors off. The reset will be performed only if no errors occurred during the write for this reset.

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle C s)	Table	CD	02	3
2. Write polling character	List	CD	03	1
3. Write space character	Table	CC, SLI	04	1
4. Read response	Area	CD	05	2
5. Read data	Area+2	SLI	11	Length-2

The read initial channel program initiated by the read/write routine places the line in control mode, polls a terminal with one character followed by a space character and reads the response. (Control mode is that state of the system that allows a terminal to be selected.) The third command (read response character) specifies a data count of 2, with wrong length indication not suppressed, while the length of the response character is one byte. Under the existing configuration of BTAM, the effect of this technique is as follows:

1. Positive Response: The response character and the first byte of the message are read under control of the "read response" CCW. This reduces the data count to zero and causes data chaining to take place. The second and subsequent bytes of the message are read under control of the address and count fields of the "read data" CCW. Execution continues in the channel with an interrupt occurring only at the end of transmission.

2. Negative Response: This response causes channel end and device end with unit exception and wrong length record indicated. The channel end routine detects the polling restart TP-Op code, reinitializes for the next terminal to be polled, and returns control to IOS for execution of the CCWs beginning with the one containing a 03 TP-Op code.

Polling is terminated if the channel end routine detects a 1 in bit 7 of IOBFLAG2 of the IOB for the line being polled. This bit is set on when the RESETPL macro-instruction is executed and is set off by the read/write routine when another operation on that line is initiated.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response (Circle Y)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read continue channel program is initiated by the problem program through the read/write routine after a successful read initial operation; the program writes the response character and command chains to read data.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write negative response (Circle N)	Table	CC, SLI	02	1
2. Read data	Area	SLI	11	Length

The read repeat channel program is initiated by the problem program through the read/write routine after a data check occurs during execution of the 'read data' command of a read initial or read continue operation. The program transmits a negative response, and then chains to the second CCW to read data into the main storage area originally specified.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read Skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write circle D and deselect characters (3 circle C s)	Table	SLI	21	4

The reset CCW is chained to the last CCW of the basic channel program if READ TIR, TTR, or TPR is specified. This CCW sends a positive response to the transmission and three circle Cs.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle Cs and circle S)	Table	CD	02	4
2. Write addressing characters	List	CD	03	1
3. Write space characters	Table	CC, SLI	04	1
4. Read response	Respn	SLI	06	2
5. Write circle D	Table	CD	08	1
6. Write data	Area	CC, SLI	11	Length
7. Read response	Respn+1		20	1

The write initial channel program, initiated by the read/write routine, places the line in control mode and informs it that the addressing function will follow by circle S, addresses a terminal with a one character code followed by a space character, and reads the response. The status of the chaining flags for the third command depends upon the status of the addressing list. For multiple component addressing, all specified components must be logically connected to the line before message transmission occurs. A negative response from any component terminates the channel program and suppresses transmission.

If the addressing list entry specified in the WRITE macro-instruction has the ECL bit on (i.e., the entry is the last or only entry in the list), the read/write routine sets the command chaining bit on in the "read response to addressing" CCW: the multiple addressing indication is ignored and the entire channel program executes upon initiation by read/write.

If the specified addressing list entry is not the last, the third command is not altered and an interrupt occurs after the response is read. The channel end routine checks for a negative response. If the response is positive, the routine reinitializes the addressing list pointer in the CCW data address, leaves the command chaining bit off in the "read response to addressing" CCW, and returns control to the input/output supervisor for execution of commands beginning with the one containing a restart TP-Op code. This I/O supervisor-to-channel end loop continues until the end of the addressing list (EOL bit on) has been reached. On the last addressing pass, the "read response to addressing" CCW is command chained to the "write circle D" command. The "write data" command always chains to the "read response to LRC" command.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to section "Dynamic Buffering" for an explanation of the additional commands.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write data	Area	CC, SLI	11	Length
2. Read response to LRC	Respn +1		20	1

The write continue channel program is initiated by the problem program through the read/write routine after a successful write initial operation; the program writes data and command chains to read the response to longitudinal redundancy checking. The response is read into DECRESPN+1, the second byte of the 2-byte response field in the DECB.

When dynamic buffering is encountered in the read/write routine, three additional commands are generated and inserted in the channel program following the write data command. Refer to the section "Dynamic Buffering" for an explanation of the additional commands.

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write response and deselect characters (Circle D and 3 circle C s)	Table	SLI	21	4

The acknowledge channel program, initiated by the read/write routine, transmits the positive response character and three deselect characters to the terminal. The positive response character indicates to the terminal that the previous message was received without transmission errors by the CPU. After writing the circle D, the program writes the deselect characters.

Write Negative Aknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (3 circle Cs)	Table	SLI	21	3

The write negative acknowledge channel program, initiated by the read/write routine, writes the three deselect characters.

Reset Following a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write EOT	Table	CC, SLI	21	1

If reset was specified by WRITE TIR or TTR the reset function is added to the basic channel program. It is performed only if the basic program completed without any errors.

CHANNEL PROGRAMS FOR IBM 2741 COMMUNICATIONS TERMINALS

Read Initial Channel Program (Nonswitched Line)

Operation	Address	Flags	Tp-Op Code	Count
1. Prepare	0	CC,SLI	01	1
2. Inhibit	Area	SLI	11	Length

The prepare command conditions the TCU to read data from the terminal. The Prepare is completed when the first character, EOA, is received from the terminal. The inhibit command reads the message text into the input area.

Read Initial Channel Program (Switched Line)

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	0	CC,SLI	01	1
2. Enable	0	CC,SLI	01	1
3. Prepare	0	CC,SLI	01	1
4. Inhibit	Area	SLI	11	Length

This channel program disables, then enables the line to receive a call from a remote terminal. The Prepare command conditions the TCU to read data from the terminal; the command is completed when the first character, EOA, is received. The Inhibit command reads the message text into the input area.

Read Continue Channel Program
Read Conversational Channel Program

Read Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write EOT	Table	CC,SLI	02	1
2. Prepare	0	CC,SLI	01	1
3. Inhibit	Area	SLI	11	Length

The Write EOT command sets the terminal to transmit state. The Prepare command conditions the TCU to read data from the terminal; the Prepare is completed when the first character, EOA, is received from the terminal. The Inhibit command reads the message text into the input area.

The same sequence of commands is used for both the Read Continue and the Read Conversational channel programs.

Read Skip Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Inhibit	0	SLI,Skip	22	Length

The Inhibit command reads data from the terminal, but because the Skip flag is on, does not place it in main storage.

Write Continue Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1. Write text	Area	SLI	11	Length

The Write text command sends message text to the terminal. This channel program is for use after a Write Conversational channel program has set the terminal to receive state.

Write Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1 Write EOA	Table	CD	02	1
2 Write text	Area	SLI	11	Length

The Write EOA command sets the terminal to receive state, then data chains to write message text. This channel program is for use after a Read operation, to reverse the direction of transmission.

Write Continue Conversational Channel Program

Operation	Address	Flags	Tp-Op Code	Count
1 Write text	Area	CC,SLI	11	Length
2 Write EOT	Table	CC,SLI	02	1
3 Prepare	0	CC,SLI	01	1
4. Inhibit	Area	SLI	11	Length

The Write text command sends message text to the terminal, and the Write EOT command sets the terminal to transmit state. The Prepare command conditions the TCU to read data from the terminal; this command is completed when the first character, EOA, is received from the terminal. The Inhibit command reads the message text into the input area.

This channel program is for use after a Write Continue or Write Conversational operation, to reverse the direction of transmission.

Write Disconnect Channel Program (Switched Line Only)

Operation	Address	Flags	Tp-Op Code	Count
1. Disable	0	SLI	21	1

The Disable command breaks the switched line connection.

CHANNEL PROGRAMS FOR IBM 2848 - 2260 REMOTE LINES

Read Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (1 STX and 15 circle Cs)	Table	CD	02	16
2. Write polling characters	List	CD	03	2
3. Write READ MI code	Table	CC, SLI	04	1
4. Read response	Area	CD	05	2
5. Read data	Area+2	SLI	11	Length-2

Initiated by the read/write routine, the read initial channel program places the line in control mode and polls a terminal with a two character code. For the 2260 devices, the polling characters specify a general poll of the display control, a specific poll of a display station, or a request of a printer status. After the polling characters are sent, the special READ MI code is sent to inform the 2848 that the CPU wants a message.

Specific Poll of a Display Station: On positive response (STX), chains the "read response" to read the message. On negative response (EOT), an interruption occurs. BTAM detects the "polling restart TP-Op Code," initializes the channel program to poll the next entry within the list, and return control to the supervisor.

Request of a Printer Status: If the printer is ready and the buffer is empty, a reservation is set on the printer buffer which prevents transmission of messages from the display stations to the printer buffer. If a message is received indicating these conditions, the "read response" chains to be "read data" CCW. The problem program issues a WRITE Continue to print the message. The next EOT resets the reservation condition.

A negative response is either NAK, which indicates the printer is not ready, or EOT, which indicates the printer is ready but the buffer is not empty. If a NAK response is received, the problem program may send a message to an operator. Both negative responses set the printer request condition which causes the Display Control, upon receipt of a General Poll, to sense if the printer is in a ready condition and if the buffer is empty.

It is advisable for the user to set the End-of-List bit on in the printer. If it is not on, the information EOT or NAK is not posted and BTAM will restart polling and return control to the supervisor.

General Poll of a Display Control: The polling list must specify a general poll, with the second byte a hexadecimal FF. If the printer has a status pending as a result of a previous request (printer status or WRITE Initial), this message will be transmitted and the "read response" CCW will chain to the read data CCW.

If the printer is not ready, the Display Stations are scanned for a message. If a message is pending, it is sent. If there is no message waiting for transmission, a negative response EOT is received. The channel program is interrupted; BTAM detects the "polling restart TP-Op code," updates the channel program, and returns control to the supervisor.

Read Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write ACK	Table	CC,SLI	02	1
2.Read data	Area	SLI	11	Length

The read continue channel program sends a positive response ACK and reads the message. If the previous operation was a specific poll of a display station, an EOT will be returned which ends the operation. If the previous operation was a general poll, a message (if one is sending) will be received; otherwise an EOT is received.

Read Repeat Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write NAK	Table	CC,SLI	02	1
2.Read data	Area	SLI	11	Length

The read repeat channel program sends a negative response NAK and reads the data.

Read Skip Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Read Skip	Zero	SLI, Skip	22	Length

Initiated by the read/write routine, the read skip channel program read the data in to clear the line but it is not read into storage.

Read Buffer Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (1 STX and 15 circle Cs)	Table	CD	02	16
2. Write polling characters	List	CD	03	2
3. Write READ BUFFER code	Table	CC, SLI	04	1
4. Read response	Area	CD	05	2
5. Read data	Area+2	SLI	11	Length-2

The read buffer channel program is used for special applications, primarily diagnostic. It places the line in control mode, sends the polling characters, and sends the READ BUFFER code to indicate the type of operation. The entire display station buffer is then read into the message area. A WRITE Erase macro-instruction should be issued following the READ Buffer. This will erase the CRT and the message(s) from the screen.

Reset Following a Read

Operation	Address	Flags	TP-Op Code	Count
1. Write STX and EOT sequence (3 circle C s)	Table	SLI	21	4

If reset was specified by READ TIR, TTR, TPR, or TBR, the reset command will be added to the end of the corresponding basic channel program. The reset will be performed only if the basic channel program completes without error.

Write Initial Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (1 STX and 15 circle Cs)	Table	CD	02	16
2. Write addressing characters	List	CD	03	2
3. Write WRITE code	Table	CC, SLI	04	1
4. Read response	Respn	CC	06	1
5. Write STX	Table	CD	08	1
6. Write data	Area	CC, SLI	11	Length
7. Read response to Text	Respn+1		20	1

The write initial channel program is for either the printer or the display station. The channel program places the line in control mode, sends the addressing characters and sends the WRITE code. If a printer is addressed, the "read response" CCW reads the addressing sequence response. If either an EOT or a NAK (negative responses) is received, there is an interrupt. The EOT indicates the printer is not ready, and the NAK indicates the printer is ready but that the buffer is not empty. Either of these sets is a printer request.

If the response is positive (ACK) which indicates that the printer is ready and the buffer is empty, the "read response" CCW command chains to send the STX (start of text character) and then send the data. If a transmission error occurs, the operation is stopped and the printer buffer is cleared. (Retransmission may be retried with a WRITE Continue macro-instruction.)

If a display station is addressed, the "read response" CCW reads the addressing sequence response which is normally positive (ACK) chains to read the data. If a transmission error occurs, the user may retry by issuing a WRITE Continue but the erroneous message will not be cleared. The user may issue a WRITE Erase to resend the same message, or a READ BUFFER and Write Erase if several messages were displayed.

Write at Line Address Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (1 STX and 15 circle Cs)	Table	CD	02	16
2. Write addressing characters	List	CD	03	2
3. Write WRITE LINE code	Table	CC, SLI	04	1
4. Read response	Respn	CC	06	1
5. Write STX	Table	CD	08	1
6. Write at line	Area	CC, SLI	10	Length
7. Read response to text	Respn+1		20	1

The write at line address channel program places the line in control mode, addresses a terminal with a two character code, and sends the WRITE LINE code to indicate the operation to the 2848. The response to addressing is read, if it is positive, the read response chains to write the STX character and the data. If the response is negative, the channel program is terminated. The cursor is positioned on a specified line and the characters are displayed from that point. The response is read into the second byte of the response field in the DECB. If a transmission error occurred, the user may retry sending the message with a WRITE Continue. The data will be displayed starting at the same line.

Write Erase Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (1 STX and 15 circle Cs)	Table	CD	02	16
2. Write addressing characters	List	CD	03	2
3. Write ERASE Code	Table	CC, SLI	04	1
4. Read response	Respn	CC	06	1
5. Write STX	Table	CD	08	1
6. Write data	Area	CC, SLI	11	Length
7. Read response to text	Respn+1		20	1

The write erase channel program places the line in control mode, addresses a terminal with the two character code, and sends the special code ERASE. This operation is to erase the CRT and any message on the display screen starting in the upper left-hand corner. The response to addressing is read in the first byte of the response field in the DECB (DECRESPN). If a negative response is received, the channel program is terminated.

If a positive response is received, the "read response" is chained to the write STX character followed by data. The response to text is read into the second byte of the DECB response field. If a transmission error occurred, the user may issue another WRITE Erase macro-instruction to send the same message.

Write Continue Channel Program

Operation	Address	Flags	TP-Op Code	Count
1. Write STX	Table	CD	08	1
2. Write data	Area	CC, SLI	11	Length
3. Read response to text	Respn+1		20	1

The write continue channel program sends a positive response STX followed by the data and reads the response into the second byte of the DECB response field (DECRESPN+1).

Write Positive Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write STX and deselect characters (3 circle C s)	Table	SLI	21	4

The write positive acknowledge channel program sends a positive response STX and three circle Cs to place the line in control mode.

Write Negative Acknowledge Channel Program

Operation	Address	Flags	TP-Op Code	Count
1.Write deselect characters (3 circle C s)	Table	SLI	21	3

The write negative acknowledge channel program turns the terminal motors off. There is no positive response character sent.

Reset Followed by a Write

Operation	Address	Flags	TP-Op Code	Count
1. Write deselect characters (STX and 3 circle Cs)	Table	SLI	21	4

If reset is specified by WRITE TIR, TLR, TSR, or TTR, the reset command is added to the corresponding basic channel program. It sends three circle Cs to turn the terminal motors off.

CHANNEL PROGRAMS EMPLOYING START-STOP AUTO POLL

The only operation types for which Auto Poll requires the building of special channel programs are those Read macro instructions which write polling characters. These are indicated by X in the following table.

Read Operation		Devices				
Name	Code	* 1030	* 1050	* 1060	2740S	* 2740SC
Initial	TI	X	X	X	X	X
Continue	TT	X		X		
Repeat	TP	X		X		
*Reset Option is allowed.						

All other operation types will result in channel programs identical to the current ones.

The form of the channel programs is not device-dependent, hence they are presented by op-type-code only.

READ Initial (TI) with SSALST

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT-Sequence	table	CC,SLI	02	3
2. Poll	any entry	CC,SLI	03	k*n
3. NOP	Poll command	SLI	09	1
4. Read Index	area	CD	0A	2
5. Read Text	area+2	(CC),SLI	11	length-2
6. Write ⒹⒸⒸⒸ	table	SLI	21	4

The line is placed in control mode by sending the EOT-Sequence of three circle Cs. For Poll the count of k*n is figured as follows: k=2 for IBM 1030 and 3 for other devices, n=number of list entries between the starting entry and the first skipped entry or the end of the list. The NOP is the last command executed on a negative response to polling at the end of the list. Upon a positive response to polling, the Read Index command is executed after the Poll, bringing into 'area' the index byte (corresponding to the responding terminal), plus the first byte of data (text). The Read Text command reads the remaining text of the message. The length operand in the READ macro instruction must allow one extra byte for receiving the index byte. The CC flag is set only if command 6 is added.

Command 6, the reset, will be performed only in the event of a successful read operation. Reset command (TIR) is not allowed for IBM 2740 with Station Control, but it is allowed for the other terminals.

READ Initial (TI) with SSAWLST

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT Sequence	table	CC,SLI	02	3
2. Poll	any entry in list	CC,SLI	03	k*n
3. TIC	2nd Poll command	SLI	09	1
4. TIC	Read Index command			
5. Poll	first entry in list	CC,SLI	03	k*n
6. TIC	2nd Poll command	SLI	09	1
7. Read Index	area	CD	0A	2
8. Read Text	area+2	(CC),SLI	11	length-2
9. Write ⒹⒸⒸⒸ	table	SLI	21	4

The line is placed in control mode by sending the EOT-Sequence of three circle Cs. For Poll the count of k*n is figured as follows: k=2 for IBM 1030 and 3 for other devices, n=number of entries between the starting entry and the first skipped entry or the end of the list. On a negative response to polling at the end of the list, command 3 (TIC) is executed to start command 5 (Poll). On a positive response to polling, command 4 (TIC) is executed to start the Read Index command. If either Poll command (2 or 5) terminates with negative response at end of list, command 5 (Poll) restarts polling at the beginning of the list. For command 5 the count is figured as follows: k=2 for the IBM 1030 and 3 for all other devices, n=the total number of active entries in the polling list. On a negative response to polling at the end of the list for command 5, this TIC (command 6) is executed to restart the previous (2nd) Poll command. On a positive response to polling, the Read Index command reads the list entry index byte and the first byte of text into the message area and chains to command 8. The Read Text command causes the remainder of the text to be read into the message area.

Reset command (TIR) is not allowed for IBM 2740 with Station Control, but it is allowed for the other terminals. The reset command will be be performed only in the event of a successful read operation.

The two TIC commands (3 and 6) are changed to NOPs if a RESETPL macro instruction (2nd operand blank or specified as POLLING) is issued. Thus, RESETPL will cause non-productive polling to terminate at the end of the list if Auto Poll is being used, whereas, without Auto Poll, RESETPL causes polling to terminate after the current poll.

READ Continue (TT)

Operation	Address	Flags	TP-OP Code	Count
1. Write Positive Response and EOT-Sequence	table	CC,SLI	02	4
2. See the discussion				

A positive response circle Y and EOT-Sequence of 3 circle Cs is sent to the terminal. The remainder of the channel program is identical to READ TI (either form), beginning with the second command.

READ Repeat (TP)

Operation	Address	Flags	TP-OP Code	Count
1. Write Negative Response and EOT-Sequence	table	CC,SLI	02	4
2. See the discussion				

A negative response circle N and EOT-Sequence of 3 circle Cs is sent to the terminal. The remainder of the channel program is identical to READ TI (either form), beginning with the second command.

This chapter gives the channel programs corresponding to the Read and Write operations that may be used for communications lines to which are connected stations employing binary synchronous communications (BSC) techniques. Channel programs are grouped by type of line configuration. The three types of configuration, the name of the device I/O module containing the model channel commands, and the value coded in the UNIT operand of the IODEVICE macro instruction are as follows:

Configuration	IODEVICE Module	IODEVICE UNIT Operand
Nonswitched point-to-point	IGG019M5	BSC1
Switched point-to-point	IGG019M6	BSC2
Nonswitched Multipoint	IGG019PC	BSC3

For any given line configuration, the channel programs may be used for any of the types of remote BSC stations that can be connected to that line configuration.

NONSWITCHED POINT-TO-POINT LINE

READ CHANNEL PROGRAMS

Eight options are available for the READ macro instruction. The operation codes and corresponding options are:

TI - Read Initial
TT - Read Continue
TTL - Read Continue with Leading Graphics
TP - Read Repeat
TPL - Read Repeat with Leading Graphics
TIQ - Read Initial Inquiry
TQ - Read Inquiry
TRV - Read Interrupt

The channel programs for these options are as follows.

Read Initial (TI)

Read TI monitors the line for an incoming character, which should always be an ENQ, sends a positive response, and receives a message block from a remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Prepare	0	CC, SLI	01	1
2. Read ENQ	Response	SLI	0B	2
3. Write response	Table	CC, SLI	08	2
4. Read text	Area	SLI	11	Length

1. This command causes the data adapter to monitor the line for an incoming character.

2. An ENQ character is expected from the remote station, indicating that it is ready to transmit.

3. An ACK-0 sequence is sent to the remote station, to indicate that the central computer is ready to receive.

4. A message block is read from the remote station.

Read Continue (TT)

Read TT is for use following successful receipt of a message block from a remote station. It sends a positive response and receives another message block from the same station.

Operation	Address	Flags	TP-OP Code	Count
1. Write ACK-0 or ACK-1	Table	CC, SLI	08	2
2. Read text	Area	SLI	11	Length

1. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.

2. A message block is read from the remote station.

Read Continue with Leading Graphics (TTL)

Read TTL is an extension of the Read Continue channel program and is used similarly. It is for use where the positive response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write ACK-0 or ACK-1	Table	CC,SLI	08	2
3. Read Text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.
2. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.
3. A message block is read from the remote station.

Read Repeat (TP)

Read TP is for use following unsuccessful receipt of a message block from a remote station. It sends a negative response and reads a message block from the same station. The station should react to the negative response by resending the same message block that elicited the negative response.

Operation	Address	Flags	TP-OP Code	Count
1. Write NAK	Table	CC,SLI	08	1
2. Read text	Area	SLI	11	Length

1. A NAK character is sent to the remote station.
2. The previous message block is reread from the remote station.

Read Repeat with Leading Graphics (TPL)

Read TPL is an extension of the Read Repeat channel program and is used similarly. It is for use where the negative response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write NAK	Table	CC,SLI	08	1
3. Read text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.
2. A NAK character is sent to the remote station.

3. The previous message block is reread from the remote station.

Read Initial Inquiry (TIQ)

Read TIQ monitors the line for an incoming character, which should be an ENQ, and receives it in the DECRESPN field. This channel program is for use where character phase has not already been established (if it has already been established when the channel program begins execution, the Prepare command ends immediately).

Operation	Address	Flags	TP-OP Code	Count
1. Prepare	0	CC,SLI	01	1
2. Read ENQ	Response	SLI	0B	2

1. This command causes the data adapter to monitor the line for an incoming character.

2. An ENQ character is expected from the remote station, indicating that it is ready to transmit.

Read Inquiry (TQ)

Read TQ receives an incoming character in the DECRESPN field. This channel program is for use when character phase is already established and an ENQ is expected.

Operation	Address	Flags	TP-OP Code	Count
1. Read ENQ	Response	SLI	0B	2

1. An ENQ character is expected from the remote station.

Read Interrupt (TRV)

Read TRV sends an RVI (Reverse interrupt) sequence and receives text from the remote station. This channel program is for use where the preceding message block from the station was received without error, but the central computer wishes temporarily to stop receiving message text. The remote station may continue sending message text, or it may send EOT, signifying that it has finished sending.

Operation	Address	Flags	TP-OP Code	Count
1. Write RVI sequence	Table	CC,SLI	08	2
2. Read Text	Area	SLI	11	Length

1. A reverse interrupt (RVI) sequence is sent to the remote station,

which treats it as though it were the correct alternating positive acknowledgment (ACK-0 or ACK-1).

2. A message block or EOT is received from the remote station.

WRITE CHANNEL PROGRAMS

Seventeen options are available for the WRITE macro instruction. The operation codes and corresponding options are:

TI	- Write Initial
TIR	- Write Initial and Reset
TIX	- Write Initial Transparent
TIXR	- Write Initial Transparent and Reset
TIE	- Write Initial Transparent Block
TIV	- Write Initial Conversational
TIVX	- Write Initial Conversational Transparent
TT	- Write Continue
TTR	- Write Continue and Reset
TTX	- Write Continue Transparent
TTXR	- Write Continue Transparent and Reset
TTE	- Write Continue Transparent Block
TTV	- Write Continue Conversational
TTVX	- Write Continue Conversational Transparent
TQ	- Write Inquiry
TR	- Write Reset
TW	- Write Wait-Before-Transmit

Write Initial (TI)
Write Initial and Reset (TIR)

Write TI (or TIR) bids for use of the line by sending an ENQ to the remote station; then upon receiving a positive response, sends a message block and receives a response to text. Write TIR sends an EOT to relinquish use of the line if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC,SLI	03	1
2. Read response	Response	SLI	0C	2
3. Write text	Area	CC,SLI	11	Length
4. Read response	Response	SLI	25	2
5. Write EOT (TIR only)	Table	SLI	21	1

1. An ENQ character is sent to the remote station to bid for use of the line, i.e., the central computer wishes to transmit.

2. A response to the ENQ is read from the remote station. If the response is ACK-0, the channel program is restarted at the next command. If the response is not ACK-0, the operation is posted complete, with or without error.

3. A message block is sent to the remote station.

4. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

5. The EOT character informs the remote station that the central computer is relinquishing control of the line.

Write Initial Transparent (TIX)
Write Initial Transparent and Reset (TIXR)

Write TIX (or TIXR) bids for use of the line by sending an ENQ to the remote station; then, upon receiving a positive response, sends a transparent message block and receives a response to text. Write TIXR sends an EOT to relinquish use of the line if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC,SLI	03	1
2. Read response	Response	SLI	0C	2
3. Write text	Area	CC,SLI	11	Length
4. Write DLE ETX	Table	CC,SLI	13	2
5. Read response	Response	SLI	25	2
6. Write EOT (TIXR only)	Table	SLI	21	1

1. An ENQ character is sent to the remote station to bid for use of the line, i.e., the central computer wishes to transmit.

2. A response to the ENQ is read from the remote station. If the response is ACK-0, the channel program is restarted at the next command. If the response is not ACK-0, the operation is posted complete, with or without error.

3. A message block is sent to the remote station. (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

4. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

5. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

6. The EOT character informs the remote station that the central computer is relinquishing control of the line.

Write Initial Transparent Block (TIE)

Write TIE functions identically to Write Initial Transparent (TIX) except that the fourth command sends a DLE ETB sequence instead of a DLE ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Initial Conversational (TIV)

Write TIV bids for use of the line by sending an ENQ to the remote station; then, upon receiving a positive response, sends a message block and receives a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC,SLI	03	1
2. Read ACK-0	Response	SLI	0C	2
3. Write text	Area	CC,SLI	11	Length
4. Read response	Area	CD,SLI	25	20
5. Read text	Area+20	SLI	11	Length -20

1. An ENQ character is sent to the remote station to bid for use of the line, i.e., the central computer wishes to transmit.
2. A response to the ENQ is read from the remote station. If the response is ACK-0, the channel program is restarted at the next command. If the response is not ACK-0, the operation is posted complete, with or without error.
3. A message block is sent to the remote station.
4. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.
5. The remainder of the message block is read from the remote station.

Write Initial Conversational Transparent (TIVX)

Write TIVX bids for use of the line by sending an ENQ to the remote station; then, upon receiving a positive response, sends a transparent message block and receives from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC,SLI	03	1
2. Read response	Response	SLI	0C	2
3. Write text	Area	CC,SLI	11	Length
4. Write DLE ETX	Table	CC,SLI	13	2
5. Read response	Area	CD,SLI	25	20
6. Read text	Area+20	SLI	11	Length -20

1. An ENQ character is sent to the remote station to bid for use of the line, i.e., the central computer wishes to transmit.

2. A response to the ENQ is read from the remote station. If the response is ACK-0, the channel program is restarted at the next command. If the response is not ACK-0, the operation is posted complete, with or without error.

3. A message block is sent to the remote station. (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

4. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

5. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

6. The remainder of the message block is read from the remote station.

Write Continue (TT)
Write Continue and Reset (TTR)

Write TT and TTR are for use after initial transmission has occurred, to send another message block and receive a response from the remote station. Write TTR sends an EOT to relinquish use of the line if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Read response	Response	SLI	25	2
3. Write EOT (TTR only)	Table	SLI	21	1

1. A message block is sent to the remote station.

2. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

3. The EOT character informs the remote station that the central computer is relinquishing control of the line.

Write Continue Transparent (TTX)
Write Continue Transparent and Reset (TTXR)

Write TTX and TTXR are for use after initial transmission has occurred, to send a transparent messge block and receive a response to the text from the remote station. Write TTXR sends an EOT to relinquish use of the line if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Write DLE ETX	Table	CC,SLI	13	2
3. Read response	Response	SLI	25	2
4. Write EOT (TTXR only)	Table	SLI	21	1

1. A message block is sent to the remote station. (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

3. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

4. The EOT character informs the remote station that the central computer is relinquishing control of the line.

Write Continue Transparent Block (TTE)

Write TTE functions identically to Write Continue Transparent (TTX) except that the second command sends a DLE ETB sequence instead of a DLE ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Continue Conversational (TTV)

Write TTV is for use after initial transmission has occurred, to send another message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Read response	Area	CD,SLI	25	20
3. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station.

2. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

3. The remainder of the message block is read from the remote station.

Write Continue Conversational Transparent (TTVX)

Write TTVX is for use after initial transmission has occurred, to send a transparent message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Write DLE ETX	Table	CC,SLI	13	2
3. Read response	Area	CD,SLI	25	20
4. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station. (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

3. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

4. The remainder of the message block is read from the remote station.

Write Inquiry (TQ)

Write TQ sends an ENQ and receives a response. This channel program is used to solicit a response from a remote station when the previous response was invalid or an expected response was not received.

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC,SLI	03	1
2. Read response	Area	SLI,CD	25	20
3. Read text	Area+20	SLI	11	Length-20

1. The ENQ character requests the remote station to resend its last response.

2. A response to the ENQ is read from the remote station. The response is read into the user-supplied input area or the DECRESPN area if count is equal to or less than 2. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

3. The remainder of the message block is read from the remote station.

Write Reset (TR)

Write TR is for use when the central computer wishes to relinquish use of the line.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	SLI	21	1

1. The EOT character informs the remote station that the central computer is relinquishing control of the line.

Write Wait-Before-Transmit (TW)

Write TW is for use in place of a Read channel program that begins by sending a positive response to text (e.g. Read TT). Write TW sends a WACK sequence, which the remote station treats as if it were the usual alternating acknowledgment (ACK-0 or ACK-1); the remote station responds to the WACK by sending an ENQ instead of another message block. This channel program may be reexecuted for as long as necessary for the central computer to delay receiving text.

Operation	Address	Flags	TP-OP Code	Count
1. Write WACK	Table	CC,SLI	01	2
2. Prepare	Zero	CC,SLI	01	1
3. Read ENQ	Response	SLI	0B	2

1. The WACK sequence informs the remote station that the central computer is temporarily not ready to continue receiving.

2. This command causes the data adapter to monitor the line for an incoming character.

3. An ENQ character is expected from the remote station.

SWITCHED POINT-TO-POINT LINE

READ CHANNEL PROGRAMS

Nine options are available for the READ macro instruction. The operation codes and corresponding options are:

TI - Read Initial
TC - Read Connect
TCW - Read Connect with Tone
TT - Read Continue
TTL - Read Continue with Leading Graphics
TP - Read Repeat
TPL - Read Repeat with Leading Graphics
TQ - Read Inquiry
TRV - Read Interrupt

The channel programs for these options are as follows.

Read Initial (TI) (Using Automatic Answering List)

Read TI, when used with an automatic answering list of the BSCLST or DIALST format, causes a call from a remote station to be answered, identification sequences exchanged, and a message block received. Use of either or both identification sequences is optional, but the ENQ character must be received and the ACK-0 sequence must be sent.

Operation	Address	Flags	TP-OP Code	Count
1. Enable	0	CC,SLI	01	1
2. Read ID ENQ	List	SLI	07 (0B)*	List
3. Write ID ACK-0	List	CC,SLI	03	List
4. Read text	Area	SLI	11	Length

*TP-Op Code for Dial List

1. The Enable command conditions the data adapter to answer calls from a remote station.
2. The identification sequence received from the remote station is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ENQ must always be present.)
3. The identification sequence of the central computer, followed by ACK-0, is sent to the remote station. (The ID sequence is optional; the ACK-0 must always be present.)
4. A message block is read from the remote station.

Read Initial (TI) (Using Automatic Calling List)

Read TI, when used with an automatic calling list of the BSCLST or DIALST format, dials the remote station whose telephone number appears in the list, exchanges identification sequences with it, and receives a message block.

Operation	Address	Flags	TP-OP Code	Count
1. Dial	List	CC, SLI	01	List
2. Write ID ENQ	List	CC, SLI	03	List
3. Read ID ACK-0	List	SLI	07 (0C)*	List
4. Write EOT	Table	CC, SLI	08	1
5. Read ENQ	Response	SLI	0B	2
6. Write ACK-0	Table	CC, SLI	08	2
7. Read text	Area	SLI	11	Length

*TP-Op Code for Dial List

1. The telephone number of the remote station is dialed.

2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)

3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)

4. The EOT character informs the remote station that the central computer is relinquishing control of the line.

5. An ENQ character is expected from the remote station, indicating that it is ready to transmit.

6. An ACK-0 sequence is sent to the remote station, to indicate that the central computer is ready to receive.

7. A message block is read from the remote station.

Read Initial (TI) (Using Manual Answering List)

Read TI, when used with a manual answering list of the WTLIST or BSCLST format, causes a call from a remote station to be answered, a data tone to be sent identification sequences exchanged, and a message block received. Use of either or both identification sequences is optional, but the ENQ character must be received and the ACK-0 sequence must be sent.

This Read Initial option is for use when the transmission control unit over which calls are received is not equipped with an automatic answering unit.

If the READ TI macro refers to a terminal list of the WTLIST format, the channel program is:

Operation	Address	Flags	TP-OP Code	Count
1. Enable	0	CC, SLI	01	1
2. Write tone	List	CC, SLI	01	List
3. Read ID ENQ	List	SLI	07	List
4. Write ID ACK-0	List	CC, SLI	03	List
5. Read Text	Area	SLI	11	Length

If the READ TI macro refers to a terminal list of the BSCLST format, the channel program is the same, except that the second command is omitted.

1. The Enable command conditions the data adapter to answer calls from a remote station.

2. A data tone is sent to inform the calling station (remote station) that the answering station (central computer) is in data mode.

3. The identification sequence received from the remote station is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ENQ must always be present.)

4. The identification sequence of the central computer, followed by ACK-0, is sent to the remote station. (The ID sequence is optional; the ACK-0 must always be present.)

5. A message block is read from the remote station.

Read Connect (TC) (Using Automatic Answering List - SWLST)

Read TC establishes contact with a remote station and performs specific actions based on the ID sequence, if any, received from the station. This channel program is for use when the expanded ID verification facility is to be employed, and requires a terminal list of the SWLST form. The possible actions that may be performed include reading a message block, disconnecting the line, and immediately returning control to the user program.

After the sequence is received from the remote station, BTAM analyzes it. If the sequence matches one of authorized sequences in the answering list, BTAM places the address of the entry containing the matching ID-ENQ sequence (or ENQ alone) in the first fullword of the list, then examines the control byte of that list entry to determine what action to take.

If the control byte value is 0, BTAM restarts the channel program at the third command to send the ID ACK-0 sequence (or ACK-0 alone) given in the list, and then reads a message block, if any. If the control byte value is 1, BTAM restarts the channel program at the fifth command to break the line connection, and then, via the TIC command, restarts the channel program at the Enable command. If the control byte value is 2, BTAM immediately posts normal completion (X'7F').

If the received sequence does not match any of the authorized ID ENQ sequences (or ENQ alone), BTAM determines whether ENQ alone, an invalid sequence, or DLE EOT was received. If ENQ alone was received, BTAM posts normal completion (X'7F'). If an invalid sequence was received, BTAM retries the Read ID ENQ command up to seven times. If all retries are unsuccessful, BTAM disconnects the line, turns on bit 3 of DECFLAGS, and posts normal completion. If a timeout occurs on the Read ID ENQ command, BTAM disconnects the line and restarts the channel program with the Enable command.

If DLE EOT was received, BTAM turns on bit 1 of DECFLAGS and posts normal completion.

Operation	Address	Flags	TP-OP Code	Count
1. Enable	0	CC, SLI	01	1
2. Read ID ENQ	List	SLI	07	List
3. Write ID ACK-0	List	CC, SLI	03	List
4. Read text	Area	SLI	11	Length
5. Write DLE EOT	Table	CC, SLI	21	2
6. Disable	0	CC, SLI	21	1
7. TIC		SLI	09	1

1. The Enable command conditions the data adapter to answer calls from a remote station.

2. The ID ENQ sequence (or ENQ alone) is received from the remote station and is analyzed.

3. The ID ACK-0 sequence (or ACK-0 alone) contained in the answering list is sent to the remote station.

4. A message block is read from the remote station.

5. The DLE EOT sequence informs the remote station that the central computer is going to break the line connection.

6. The Disable command breaks the line connection.

7. This command transfers in channel to the Enable command.

Read Connect with Tone (TCW) (Using Automatic Answering List)

Read TCW, when used with an automatic answering list of the SWLST format, functions the same as Read Connect (TC), as described above, except that the channel program contains an added command, Write Data Tone characters. This operation is for use on a line equipped with an automatic answering unit that does not automatically send a data tone upon receiving a call. Upon completion of the Enable command, which occurs when a call is received, the channel program sends a user-specified character sequence that the operator at the calling station hears as an audible tone.

Operation	Address	Flags	TP-OP Code	Count
1. Enable	0	CC, SLI	01	1
2. Write tone	User area	CC, SLI	01	Length
3. Read ID ENQ	List	SLI	07	List
4. Write ID ACK-0	List	CC, SLI	03	List
5. Read text	Area	SLI	11	Length
6. Write DLE EOT	Table	CC, SLI	21	2
7. Disable	0	CC, SLI	21	1
8. TIC		SLI	09	1

Read Continue (TT)

Read TT is for use following successful receipt of a message block from a remote station. It sends a positive response and receives another message block from the same station.

Operation	Address	Flags	TP-OP Code	Count
1. Write ACK-0 or ACK-1	Table	CC, SLI	08	2
2. Read text	Area	SLI	11	Length

1. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.

2. A message block is read from the remote station.

Read Continue with Leading Graphics (TTL)

Read TTL is an extension of the Read Continue channel program and is used similarly. It is for use where the positive response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write ACK-0 or ACK-1	Table	CC, SLI	08	2
3. Read text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.

2. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.

3. A message block is read from the remote station.

Read Repeat (TP)

Read TP is for use following unsuccessful receipt of a message block from a remote station. It sends a negative response and reads a message block from the same station. The station should react to the negative response by resending the same message block that elicited the negative response.

Operation	Address	Flags	TP-OP Code	Count
1. Write NAK	Table	CC,SLI	08	1
2. Read text	Area	SLI	11	Length

1. A NAK character is sent to the remote station.

2. The previous message block is reread from the remote station.

Read Repeat with Leading Graphics (TPL)

Read TPL is an extension of the Read Repeat channel program and is used similarly. It is for use where the negative response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write NAK	Table	CC,SLI	08	1
3. Read text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.

2. A NAK character is sent to the remote station.

3. The previous message block is reread from the remote station.

Read Inquiry (TQ)

Read TQ receives an incoming character in the DECRESPN field. This channel program is for use when an ENQ is expected from the remote station, after the line connection (and character phase) have already been established.

Operation	Address	Flags	TP-OP Code	Count
1. Read ENQ	Response	SLI	0B	2

1. An ENQ character is expected from the remote station.

Read Interrupt (TRV)

Read TRV sends an RVI (Reverse Interrupt) sequence and receives text from the remote station. This channel program is for use where the preceding message block from the station was received without error, but the central computer wishes temporarily to stop receiving message text. The remote station may continue ssending message text, or it may send EOT, signifying that it has finished sending.

Operation	Address	Flags	TP-OP Code	Count
1. Write RVI sequence	Table	CC,SLI	08	2
2. Read Text	Area	SLI	11	Length

1. A reverse interrupt (RVI) sequence is sent to the remote station, which treats it as though it were the correct alternating positive acknowledgment (ACK-0 or ACK-1).

2. A message block or EOT is received from the remote station.

WRITE CHANNEL PROGRAMS

Sixteen options are available for the WRITE macro instruction. The operation codes and the corresponding options are:

- TI - Write Initial
- TIX - Write Initial Transparent
- TIE - Write Initial Transparent Block
- TIV - Write Initial Conversational
- TIVX - Write Initial Conversational Transparent
- TC - Write Connect
- TT - Write Continue
- TTX - Write Continue Transparent
- TTE - Write Continue Transparent Block
- TTV - Write Continue Conversational
- TTVX - Write Continue Conversational Transparent
- TQ - Write Inquiry
- TR - Write Reset
- TW - Write Wait-Before-Transmit
- TB - Write Break
- TD - Write Disconnect

The channel programs for these options are as follows.

Write Initial (TI)

Write TI dials the remote station whose telephone number appears in the addressing list, exchanges identification sequences with it, sends a message block, and receives a response to text.

Operation	Address	Flags	TP-OP Code	Count
1. Dial	List	CC, SLI	01	List
2. Write ID ENQ	List	CC, SLI	03	List
3. Read ID ACK-0	List	SLI	07	List
4. Write text	Area	CC, SLI	11	Length
5. Read response	Response	SLI	25	2

1. The telephone number of the remote station is dialed.
2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)
3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)
4. A message block is sent to the remote station.
5. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), that fact is indicated in the DECB.

Write Initial Transparent (TIX)

Write TIX dials the remote station whose telephone number appears in the addressing list, exchanges identification sequences with it, sends a transparent message block, and receives a response to text.

Operation	Address	Flags	TP-OP Code	Count
1. Dial	List	CC, SLI	01	List
2. Write ID ENQ	List	CC, SLI	03	List
3. Read ID ACK-0	List	SLI	07	List
4. Write text	Area	CC, SLI	11	Length
5. Write DLE ETX	Table	CC, SLI	13	2
6. Read response	Response	SLI	25	2

1. The telephone number of the remote station is dialed.

2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)

3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)

4. A message block is sent to the remote station. (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

5. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

6. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), that fact is indicated in the DECB.

Write Initial Transparent Block (TIE)

Write TIE functions identically to Write Initial Transparent (TIX) except that the fifth command sends a DLE ETB sequence instead of a DLE ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Initial Conversational (TIV)

Write TIV dials the remote station whose telephone number appears in the addressing list, exchanges identification sequences with it, sends a message block, and receives a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Dial	List	CC,SLI	01	List
2. Write ID ENQ	List	CC,SLI	03	List
3. Read ID ACK-0	List	SLI	07	List
4. Write text	Area	CC,SLI	11	Length
5. Read response	Area	CD,SLI	25	20
6. Read text	Area+20	SLI	11	Length -20

1. The telephone number of the remote station is dialed.

2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)

3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained

in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)

4. A message block is sent to the remote station.

5. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

6. The remainder of the message block is read from the remote station.

Write Initial Conversational Transparent (TIVX)

Write TIVX dials the remote station whose telephone number appears in the addressing list, exchanges identification sequences with it, sends a transparent message block and receives from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Dial	List	CC, SLI	01	List
2. Write ID ENQ	List	CC, SLI	03	List
3. Read ID ACK-0	List	SLI	07	List
4. Write text	Area	CC, SLI	11	Length
5. Write DLE ETX	Table	CC, SLI	13	2
6. Read response	Area	CD, SLI	25	20
7. Read text	Area+20	SLI	11	Length -20

1. The telephone number of the remote station is dialed.

2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)

3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)

4. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

5. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

6. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

7. The remainder of the message block is read from the remote station.

Write Connect (TC)

Write TC is for use when calls to remote stations must be initiated manually by the console operator rather than by program control.

Operation	Address	Flags	TP-OP Code	Count
1. Enable	0	CC,SLI	01	1
2. Write ID ENQ	List	CC,SLI	03	List
3. Read ID ACK-0	List	SLI	07	List

1. The Enable command conditions the data adapter so that the console operator may manually dial a remote station. This command is terminated when the operator places the data set (modem) in data mode.

2. The identification sequence of the central computer, followed by ENQ, is sent to the remote station. (The ID sequence is optional; the ENQ must always be present.)

3. The identification sequence (followed by ACK-0) is received from the remote station and is compared with the expected sequence contained in the terminal list. If the two sequences do not match, the operation is posted complete-with-error. (The ID sequence is optional; the ACK-0 must always be present.)

Write Continue (TT)

Write TT is for use after initial transmission has occurred, to send another message block and receive a response from the remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC,SLI	11	Length
2. Read response	Response	SLI	25	2

1. A message block is sent to the remote station.

2. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), that fact is indicated in the DECB.

Write Continue Transparent (TTX)

Write TTX is for use after initial transmission has occurred, to send a transparent message block and receive a response from the remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Write DLE ETX	Table	CC, SLI	13	2
3. Read response	Response	SLI	25	2

1. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

3. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), that fact is indicated in the DECB.

Write Continue Transparent Block (TTE)

Write TTE functions identically to Write Continue Transparent (TTX) except that the second command sends a DLE ETB sequence instead of a DLE ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Continue Conversational (TTV)

Write TTV is for use after initial transmission has occurred, to send another message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Read response	Area	CD, SLI	25	20
3. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station.
2. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.
3. The remainder of the message block is read from the remote station.

Write Continue Conversational Transparent (TTVX)

Write TTVX is for use after initial transmission has occurred, to send a transparent message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Write DLE ETX	Table	CC, SLI	13	2
3. Read response	Area	CD, SLI	25	20
4. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

3. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

4. The remainder of the message block is read from the remote station.

Write Inquiry (TQ)

Write TQ sends an ENQ and receives a response. This channel program is used (1) to solicit a response from a remote station when the previous response was invalid or an expected response was not received, or (2) to indicate to the remote station that the central computer wishes to transmit (used after the line connection has been established).

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC, SLI	03	1
2. Read response	Area	SLI, CD	25	20
3. Read text	Area+20	SLI	11	Length-20

1. The ENQ character is either a request to the remote station to resend its last response or a bid for use of the line.

2. A response to the ENQ is read from the remote station. The response is read into the user-supplied input area or the DECRESPN area if count is equal to or less than 2. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

3. The remainder of the message block is read from the remote station.

Write Reset (TR)

Write TR is for use when the central computer has finished sending message blocks.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CC, SLI	08	1
2. Read response	Response	SLI	0B	2

1. The EOT character informs the remote station that the central computer is relinquishing control of the line.

2. A response to the EOT is read from the remote station. Valid responses are EOT (the remote station does not wish either to transmit or to break the line connection); ENQ (the remote station wishes to transmit); and DLE EOT (the remote station is breaking the line connection).

Write Wait-Before-Transmit (TW)

Write TW is for use in place of a Read channel program that begins by sending a positive response to text (e.g. Read TT). Write TW sends a WACK sequence, which the remote station treats as if it were the usual alternating acknowledgment (ACK-0 or ACK-1); the remote station responds to the WACK by sending an ENQ instead of another message block. This channel program may be reexecuted for as long as necessary for the central computer to delay receiving text.

Operation	Address	Flags	TP-OP Code	Count
1. Write WACK	Table	CC, SLI	01	2
2. Read ENQ	Response	SLI	0B	2

1. The WACK sequence informs the remote station that the central computer is temporarily not ready to continue receiving.

2. An ENQ character is expected from the remote station.

Write Break (TB)

Write TB is used to break the line connection without first notifying the remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Disable	0	SLI	21	1

1. The Disable command breaks the line connection.

Write Disconnect (TD)

Write TD is used to break the line connection after first notifying the remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Write DLE EOT	Table	CC,SLI	21	2
2. Disable	0	SLI	21	1

1. The DLE EOT sequence informs the remote station that the central computer is going to break the line connection.

2. The Disable command breaks the line connection.

NONSWITCHED MULTIPOINT LINE

READ CHANNEL PROGRAMS

Seven options are available for the READ macro instruction. The operation codes and corresponding options are:

- TI - Read Initial
- TT - Read Continue
- TTL - Read Continue with Leading Graphics
- TP - Read Repeat
- TPL - Read Repeat with Leading Graphics
- TQ - Read Inquiry
- TRV - Read Interrupt

The channel programs for these options are as follows.

Read Initial (TI) (Using Open Polling List)

Read TI, when used with an open polling list, causes one pass to be made through the list, successively polling each of the remote stations represented by entries in the list. Upon receiving a positive response from any station in the list, a message block is received from that station. If all stations return negative responses, or if any station fails to respond at all, the channel program ends.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT (Poll)	Table	CC, SLI	02	1
2. Poll	List	CC, SLI	03	List
3. I/O No-op	0	SLI	09	1
4. Read index	Area	CD	0A	2
5. Read text	Area+2	SLI	11	Length-2

1. The EOT character places all remote stations on the line in control mode.
2. This command initiates the Auto Poll function. If a positive response is received for any remote station in the polling list, the channel program is restarted at command 4. If all stations return negative responses, or if any station fails to return a response, command (3) is executed.
3. Execution of this command ends the channel program; the operation is posted complete.
4. The index byte and the first character (if any) are read.
5. The remainder of the message block is read from the remote station.

Read Initial (TI) (Using Wraparound Polling List)

Read TI, when used with a wraparound polling list, causes continuous cycling through the polling list, successively polling each of the remote stations represented by entries in the list. Upon receiving a positive response from any station in the list, a message block is received from that station. If all stations return negative responses, polling is restarted from the beginning of the list. Polling is thus performed continuously until some station responds with a message block or the channel program is terminated, as by a RESETPL macro instruction.

If any station fails to respond at all, timeout is indicated in DECSENS0 and a completion code of X'41' is posted.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CD, SLI	08	1
2. Poll	List	CC, SLI	03	List
3. TIC		SLI	09	1
4. TIC		SLI	09	1
5. Poll	List	CC, SLI	03	List
6. TIC		SLI	09	1
7. Read index	Area	CD	0A	2
8. Read text	Area+2	SLI	11	Length-2

1. The EOT character places all remote stations on the line in control mode.

2. This command initiates the Auto Poll function, starting with the remote station whose polling list entry is specified in the macro. If that station or any station whose list entry follows that station's entry returns a positive response, the status modifier bit is turned on in the CSW, which causes the next command to be skipped. If all stations return negative responses, or if any station fails to return a response, the next command is executed.

3. The TIC command restarts the channel program at command 5.

4. The TIC command restarts the channel program at command 7.

5. This command restarts the Auto Poll function, starting with the remote station whose polling list entry appears first in the list. If that station or any other station appearing in the list returns a positive response, the status modifier bit is turned on in the CSW, which causes the next command to be skipped. If all stations return negative responses, the next command is executed. If any station fails to respond at all, timeout is indicated in DECSENS0 and a completion code of X'41' is posted.

6. The TIC command restarts the channel program at command 5.

7. The index byte and the first text character (if any) are read.

8. The remainder of the message block is read from the remote station.

Read Continue (TT)

Read TT is for use following successful receipt of a message block from a remote station. It sends a positive response and receives another message block from the same station.

Operation	Address	Flags	TP-OP Code	Count
1. Write ACK-0 or ACK-1	Table	CC,SLI	08	2
2. Read text	Area	SLI	11	Length

1. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.

2. A message block is read from the remote station.

Read Continue with Leading Graphics (TTL)

Read TTL is an extension of the Read Continue channel program and is used similarly. It is for use where the positive response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write ACK-0 or ACK-1	Table	CC,SLI	08	2
3. Read text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.

2. A positive response to text (ACK-0 or ACK-1) is sent to the remote station.

3. A message block is read from the remote station.

Read Repeat (TP)

Read TP is for use following unsuccessful receipt of a message block from a remote station. It sends a negative response and reads a message block from the same station. The station should react to the negative response by resending the same message block that elicited the negative response.

Operation	Address	Flags	TP-OP Code	Count
1. Write NAK	Table	CC,SLI	08	1
2. Read text	Area	SLI	11	Length

1. A NAK character is sent to the remote station.

2. The previous message block is reread from the remote station.

Read Repeat with Leading Graphics (TPL)

Read TPL is an extension of the Read Repeat channel program and is used similarly. It is for use where the negative response is to be preceded by user-specified graphic characters.

Operation	Address	Flags	TP-OP Code	Count
1. Write graphics	Area	CD	01	Length
2. Write NAK	Table	CC, SLI	08	1
3. Read text	Area	SLI	11	Length

1. One to seven graphics are sent to the remote station.

2. A NAK character is sent to the remote station.

3. The previous message block is reread from the remote station.

Read Inquiry (TQ)

Read TQ receives an incoming character in the DECRESPN field. This channel program is for use when an ENQ is expected from the remote station.

Operation	Address	Flags	TP-OP Code	Count
1. Read ENQ	Response	SLI	0B	2

1. An ENQ character is expected from the remote station.

Read Interrupt (TRV)

Read TRV sends an RVI (Reverse Interrupt) sequence and receives text from the remote station. This channel program is for use where the preceding message block from the station was received without error, but the central computer wishes temporarily to stop receiving message text. The remote station may continue sending message text, or it may send EOT, signifying that it has finished sending.

Operation	Address	Flags	TP-OP Code	Count
1. Write RVI sequence	Table	CC, SLI	08	2
2. Read text	Area	SLI	11	Length

1. A reverse interrupt (RVI) sequence is sent to the remote station, which treats it as though it were the correct alternating positive acknowledgment (ACK-0 or ACK-1).

2. A message block or EOT is received from the remote station.

WRITE CHANNEL PROGRAMS

Seventeen options are available for the WRITE macro instruction. The operation codes and corresponding options are:

TI	- Write Initial
TIR	- Write Initial and Reset
TIX	- Write Initial Transparent
TIXR	- Write Initial Transparent and Reset
TIE	- Write Initial Transparent Block
TIV	- Write Initial Conversational
TIVX	- Write Initial Conversational Transparent
TT	- Write Continue
TTR	- Write Continue and Reset
TTX	- Write Continue Transparent
TTXR	- Write Continue Transparent and Reset
TTE	- Write Continue Transparent Block
TTV	- Write Continue Conversational
TTVX	- Write Continue Conversational Transparent
TQ	- Write Inquiry
TR	- Write Reset
TW	- Write Wait-before-Transmit

The channel programs for these options are as follows.

Write Initial (TI)
Write Initial and Reset (TIR)

Write TI and TIR reset to control mode all remote stations on the line, address a specific remote station, read a response, and if the response is positive, send a message block and receive a response to text. Write TIR sends an EOT to reset the stations to control mode if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CC, SLI	02	1
2. Write addressing characters	List	CC, SLI	03	List
3. Read response	Response	SLI	06	2
4. Write text	Area	CC, SLI	11	Length
5. Read response	Response	SLI	25	2
6. Write EOT (TIR only)	Table	SLI	21	1

1. The EOT character places all remote stations on the line in control mode.

2. This command transmits the addressing sequence contained in the addressing list specified in the macro.

3. The response to addressing is read from the addressed station. If the response is ACK-0, the channel program is restarted at the next command. Otherwise, it is posted complete.

4. A message block is sent to the remote station.

5. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

6. The EOT character places the remote stations on the line in control mode.

<u>Write Initial Transparent (TIX)</u>
<u>Write Initial Transparent and Reset (TIXR)</u>

Write TIX and TIXR reset to control mode all remote stations on the line, address a specific remote station, read a response, and if the response is positive, send a transparent message block and receive a response to text. Write TIXR sends an EOT to reset the stations to control mode if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CC, SLI	02	1
2. Write addressing characters	List	CC, SLI	03	List
3. Read response	Response	SLI	06	2
4. Write text	Area	CC, SLI	11	Length
5. Write DLE ETX	Table	CC, SLI	13	2
6. Read response	Response	SLI	25	2
7. Write EOT (TIXR only)	Table	SLI	21	1

1. The ECT character places all remote stations on the line in control mode.

2. This command transmits the addressing sequence contained in the addressing list specified in the macro.

3. The response to addressing is read from the addressed station. If the response is ACK-0, the channel program is restarted at the next command. Otherwise, it is posted complete.

4. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

5. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

6. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).

7. The EOT character places the remote stations on the line in control mode.

Write Initial Transparent Block (TIE)

Write TIE functions identically to Write Initial Transparent (TIX) except that the fifth command sends a DLE ETB sequence instead of a DLE ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Initial Conversational (TIV)

Write TIV resets to control mode all remote stations on the line, addresses a specific remote station, reads a response, and if the response is positive, sends a message block and receives a response to text, followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CC,SLI	02	1
2. Write addressing characters	List	CC,SLI	03	List
3. Read response	Response	SLI	06	2
4. Write Text	Area	CC,SLI	11	Length
5. Read Response	Area	CD,SLI	25	20
6. Read Text	Area+20	SLI	11	Length -20

1. The EOT character places all remote station on the line in control mode.

2. This command transmits the addressing sequence contained in the addressing list specified in the macro.

3. The response to addressing is read from the addressed station. If the response is ACK-0, the channel program is restarted at the next command. Otherwise, it is posted complete.

4. A message block is sent to the remote station.

5. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

6. The remainder of the message block is read from the remote station.

Write Initial Conversational Transparent (TIVX)

Write TIVX resets to control mode all remote stations on the line, addresses a specific remote station, reads a response, and if the response is positive, sends a transparent message block and receives a response to text, followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	CC, SLI	02	1
2. Write addressing characters	List	CC, SLI	03	List
3. Read response	Response	SLI	06	2
4. Write Text	Area	CC, SLI	11	Length
5. Write DLE ETX	Table	CC, SLI	13	2
6. Read Response	Area	CD, SLI	25	20
7. Read Text	Area+20	SLI	11	Length -20

1. The EOT character places all remote stations on the line in control mode.

2. This command transmits the addressing sequence contained in the addressing list specified in the macro.

3. The response to addressing is read from the addressed station. If the response is ACK-0, the channel program is restarted at the next command. Otherwise, it is posted complete.

4. A message block is sent to the remote station.

5. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the station.

6. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

7. The remainder of the message block is read from the remote station.

Write Continue (TT)
Write Continue and Reset (TTR)

Write TT and TTR are for use after initial transmission has occurred, to send another message block and receive a response from the remote station. Write TTR sends an EOT to reset the stations to control mode if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Read response	Response	SLI	25	2
3. Write EOT (TTR only)	Table	SLI	21	1

1. A message block is sent to the remote station.
2. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).
3. The EOT character places the remote stations on the line in control mode.

Write Continue Transparent (TTX)
Write Continue Transparent and Reset (TTXR)

Write TTX and TTXR are for use after intitial transmission has occurred, to send a transparent message block and receive a response to text from the remote station. Write TTXR sends an EOT to reset all stations to control mode, if a positive response to text is received.

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Write DLE ETX	Table	CC, SLI	13	2
3. Read response	Response	SLI	25	2
4. Write EOT (TTXR only)	Table	SLI	21	1

1. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)
2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.
3. A response to text is read from the remote station; if it is not the expected response (ACK-0 or ACK-1), the condition is indicated in the DECB and the operation is posted complete (regardless of whether the Reset option was specified).
4. The EOT character places the remote stations on the line in control mode.

Write Continue Transparent Block (TTE)

Write TTE functions identically to Write Continue Transparent (TTX) except that the second command sends a DLE ETB sequence instead of a DLE

ETX sequence; only the ETB appears in the input buffer of the remote station.

Write Continue Conversational (TTV)

Write TTV is for use after intial transmission has occurred, to send another message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Read response	Area	CD, SLI	25	20
3. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station.

2. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

3. The remainder of the message block is read from the remote station.

Write Continue Conversational Transparent (TTVX)

Write TTVX if for use after initial transmission has occurred, to send a transparent message block and receive from the remote station a response to text followed by a message block (if any).

Operation	Address	Flags	TP-OP Code	Count
1. Write text	Area	CC, SLI	11	Length
2. Write DLE ETX	Table	CC, SLI	13	2
3. Read response	Area	CD, SLI	25	20
4. Read text	Area+20	SLI	11	Length -20

1. A message block is sent to the remote station.

   (Note: The length specified should not include the ETX that might have been placed in the buffer during a previous operation.)

2. The two-character sequence DLE ETX is sent to the remote station; only the ETX appears in the input buffer of the remote station.

3. A response to text is read from the remote station. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

4. The remainder of the message block is read from the remote station.

Write Inquiry (TQ)

Write TQ sends an ENQ and receives a response. This channel program is used to solicit a response from a remote station when the previous response was invalid or an expected response was not received.

Operation	Address	Flags	TP-OP Code	Count
1. Write ENQ	Table	CC, SLI	03	1
2. Read response	Area	SLI,CD	25	20
3. Read text	Area+20	SLI	11	Length-20

1. The ENQ character requests the remote station to resend its last response.

2. A response to the ENQ is read from the remote station. The response is read into the user-supplied input area or the DECRESPN area if count is equal to or less than 2. If the response is message text exceeding 20 characters, this command is data-chained to the next command. Otherwise, the operation is posted complete, with or without error.

3. The remainder of the message block is read from the remote station.

Write Reset (TR)

Write TR is for use when the central computer wishes to reset to control mode all remote stations on the line.

Operation	Address	Flags	TP-OP Code	Count
1. Write EOT	Table	SLI	21	1

1. The EOT character places the remote stations on the line in control mode.

Write Wait-Before-Transmit (TW)

Write TW is for use in place of a Read channel program that begins by sending a positive response to text (e.g. Read TT). Write TW sends a WACK sequence, which the remote station treats as if it were the usual alternating acknowledgment (ACK-0 or ACK-1); the remote station responds to the WACK by sending an ENQ instead of another message block. This channel program may be reexecuted for as long as necessary for the central computer to delay receiving text.

Operation	Address	Flags	TP-OP Code	Count
1. Write WACK	Table	CC,SLI	01	2
2. Read ENQ	Response	SLI	0B	2

1. The WACK sequence informs the remote station that the central computer is temporarily not ready to continue receiving.

2. An ENQ character is expected from the remote station.

LOCAL CHANNEL PROGRAMS

LOCAL IBM 3270 DISPLAY SYSTEM

READ CHANNEL PROGRAMS

Five options are available for the READ macro instruction. The operation codes and corresponding options are:

- TI - Read Initial
- TM - Read Modified
- TMP - Read Modified from Position
- TB - Read Buffer
- TBP - Read Buffer from Position

The channel programs for these options follow.

Read Initial (TI)

Read TI reads modified fields from a local 3270 display station after the display station operator causes an attention interruption.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Read Modified	area	SLI	00	length

Read Modified (TM)

Read TM reads modified fields from a local 3270 device independently of action by the display station operator.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Read Modified	area	SLI	00	length

Read Modified from Position (TMP)

Read TMP reads modified fields from a local 3270 device beginning at a specified location in the buffer.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Write	area	CC, SLI	00	4
3. Read Modified	area	SLI	00	length

Read Buffer (TB)

Read TB reads the entire buffer of a local 3270 device.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Read buffer	area	SLI	00	length

Read Buffer from Position (TBP)

Read TBP reads the entire buffer of a local 3270 device beginning at a specified location.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Write	area	CC, SLI	00	4
3. Read buffer	area	SLI	00	length

WRITE CHANNEL PROGRAMS

Three options are available for the WRITE macro instruction. The operation codes and corresponding options are:

TI - Write Initial
TS - Write Erase
TUS - Write Unconditional Erase

The channel programs for these options follow.

Write Initial (TI)

Write TI writes a message to a local 3270 device.

Operation	Address	Flags	TP-Op Code	Count
1. Select	area	CC	00	length
2. Write	area	SLI	00	length

Write Erase (TS)

Write TS clears the buffer of a local 3270 device to nulls (binary zeros) and writes a message to the device.

Operation	Address	Flags	TP-Op Code	Count
1. Erase/Write	area	SLI	00	length

Write Unprotected Erase (TUS)

Write TUS clears all unprotected fields in the buffer of a local 3270 device to nulls (binary zeros).

Operation	Address	Flags	TP-Op Code	Count
1. Erase all unprotected	area	CC, SLI	00	1
2. NOP				

Chart 01. BTAM Open Executor (IGG0193M) (Load 1) (Part 1 of 2)

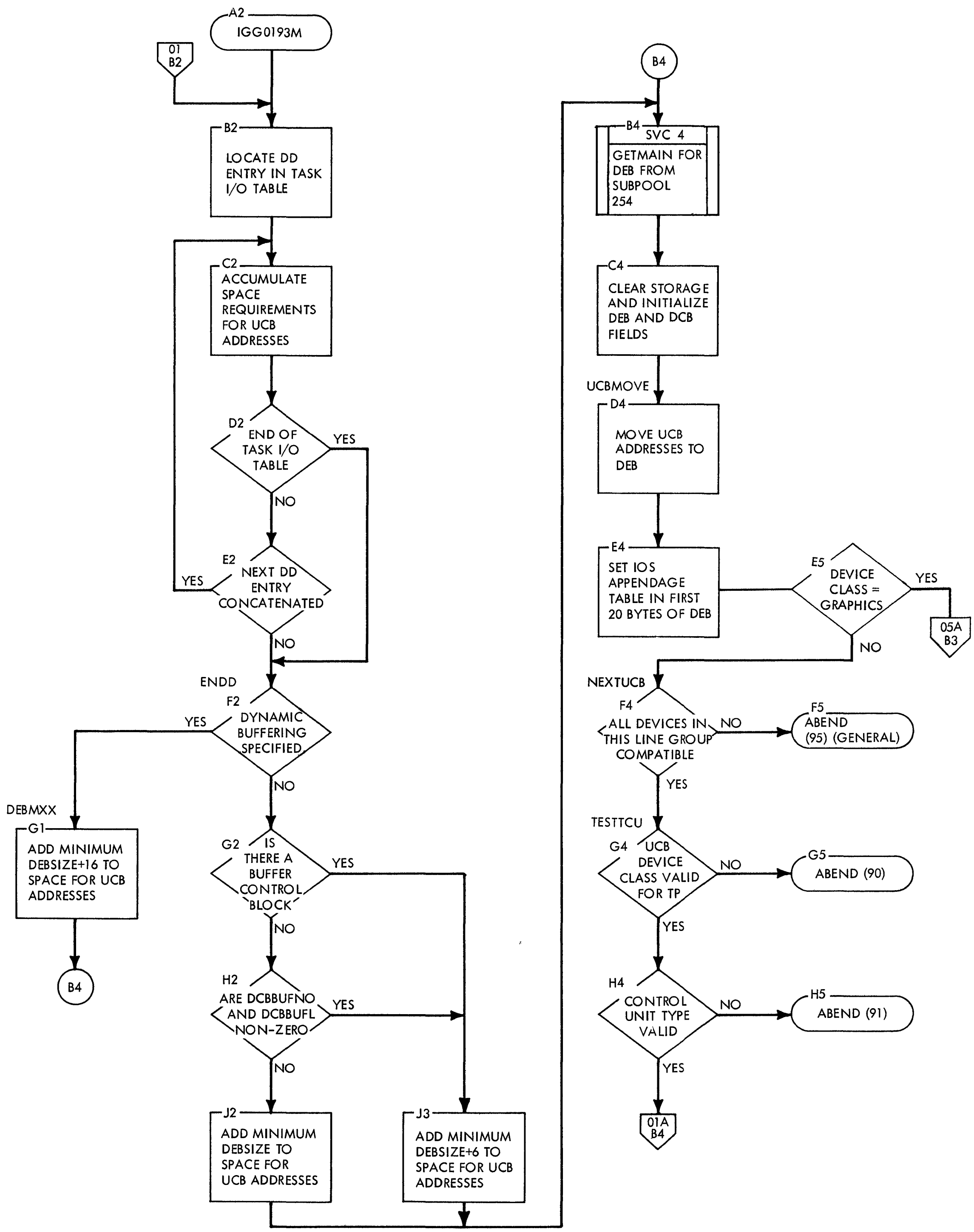


Chart 01A. BTAM Open Executor (IGG0193M) (Load 1) (Part 2 of 2)

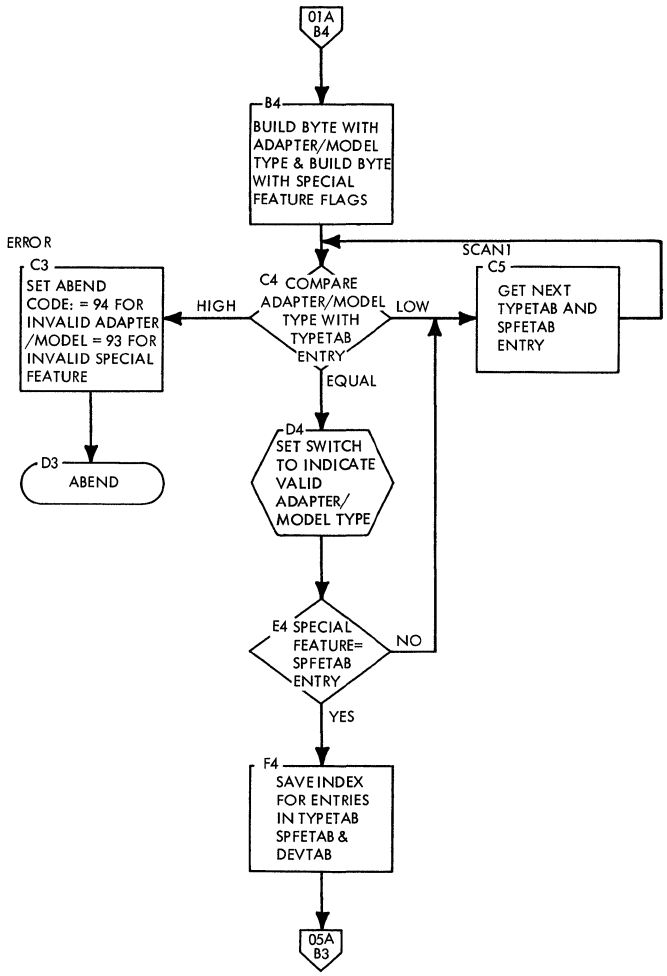


Chart 01B. BTAM Open Executor (IGG0194N) (Load 2)

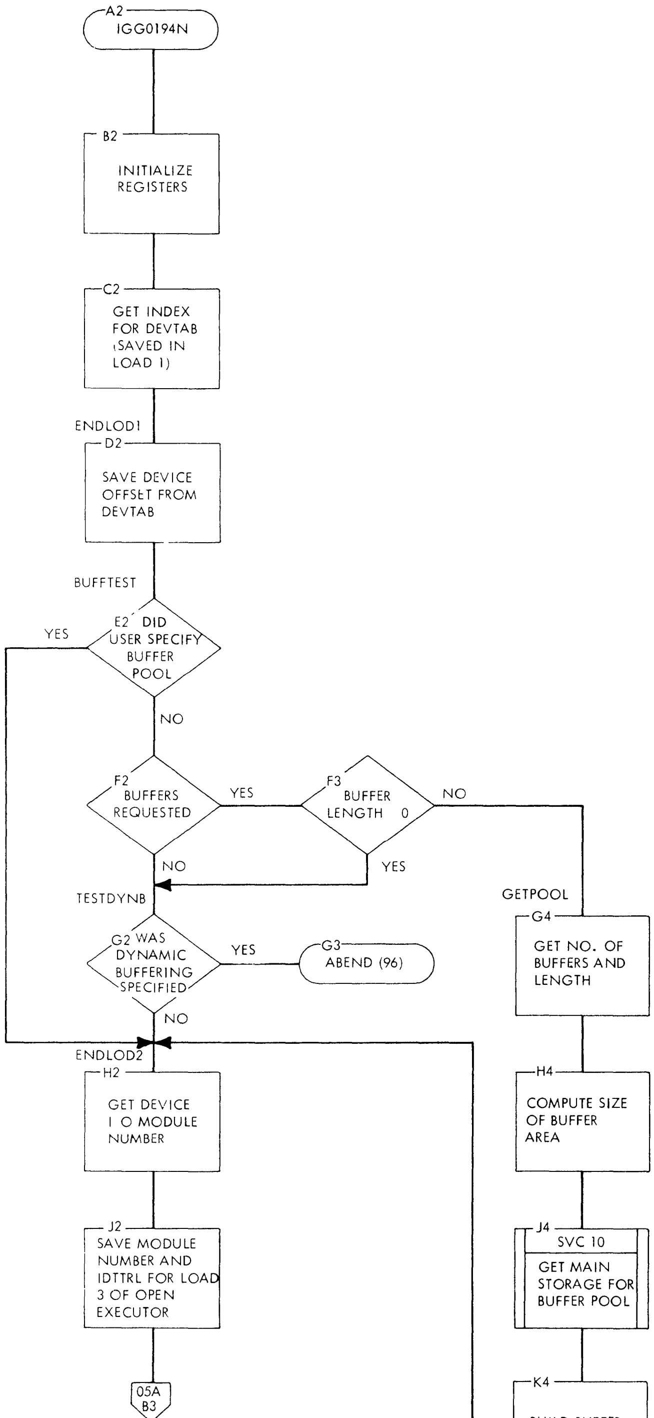


Chart 02. BTAM Open Executor (IGG0193Q) (Load 3) (Part 1 of 2)

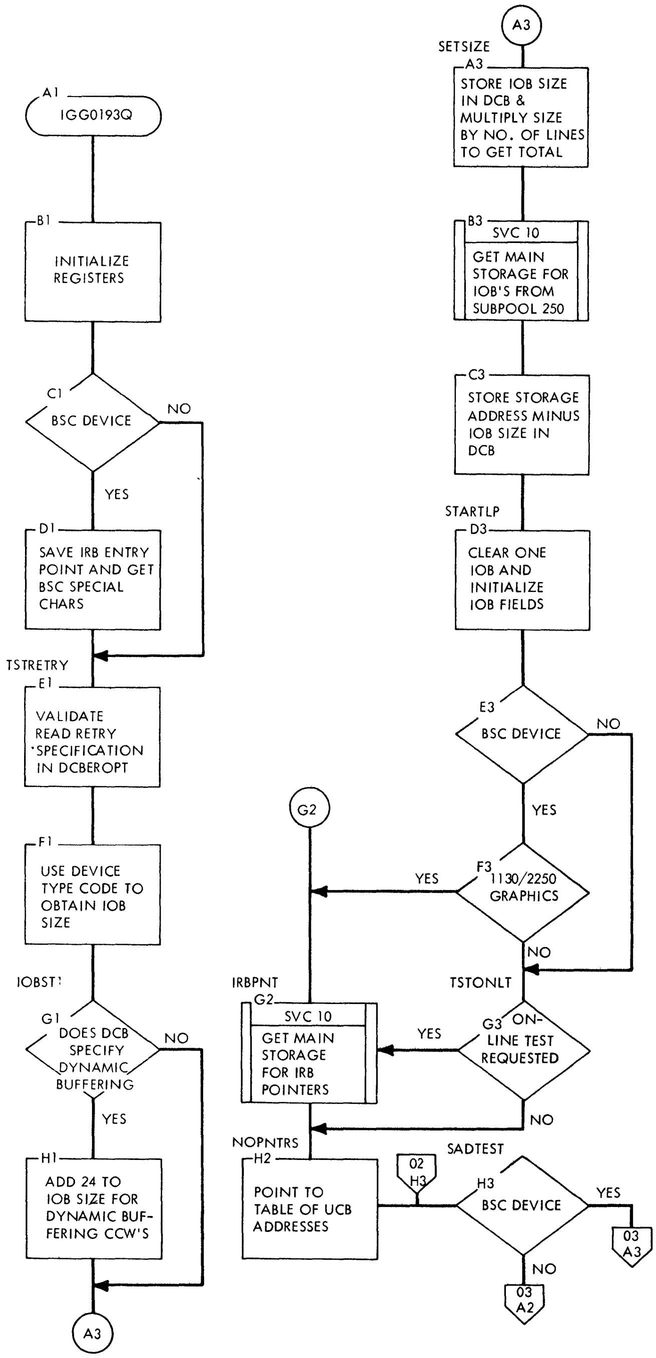


Chart 03. BTAM Open Executor (IGG0193Q) (Load 3) (Part 2 of 2)

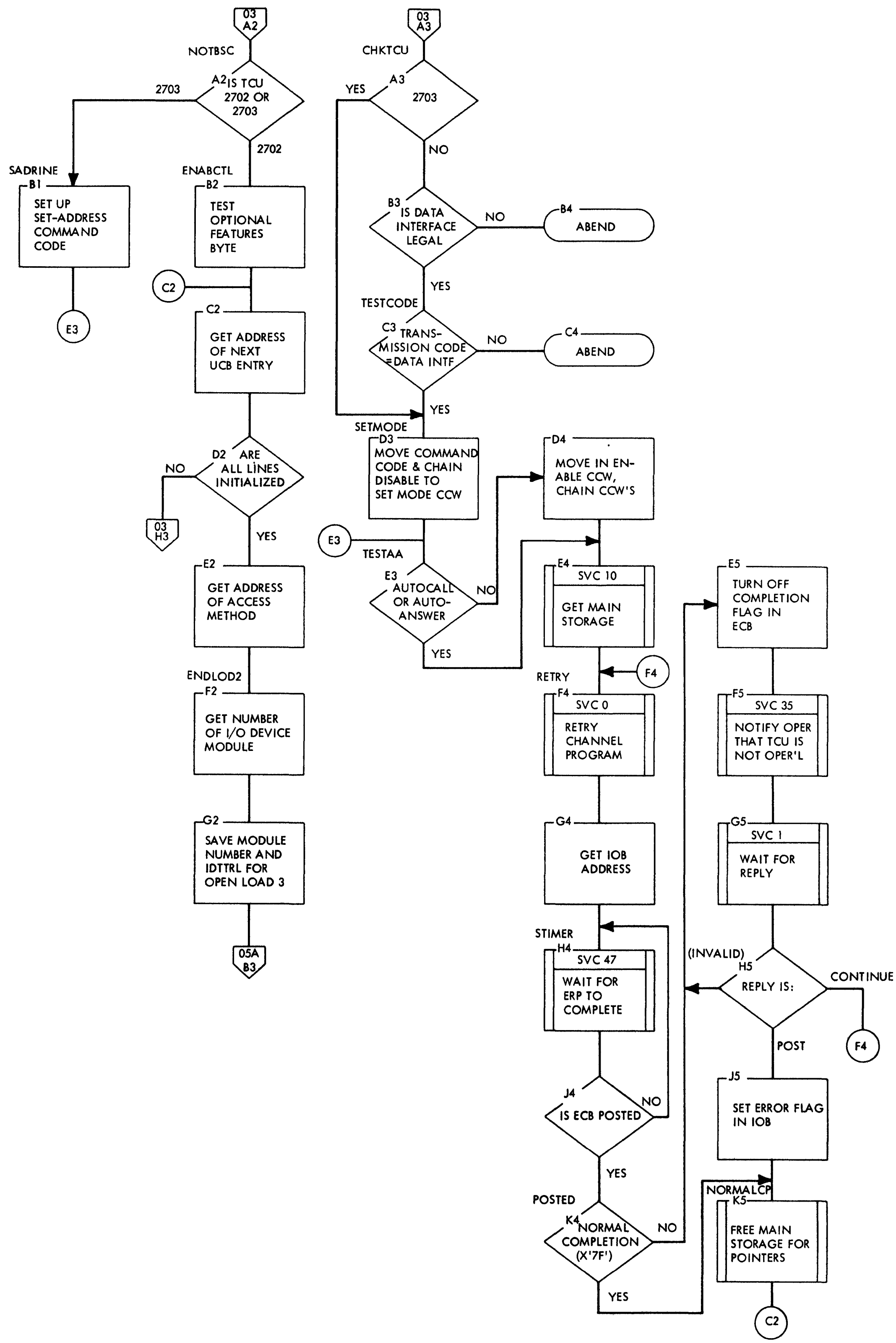


Chart 04. BTAM Open Executor (IGG0193S) (Load 4) (Part 1 of 2)

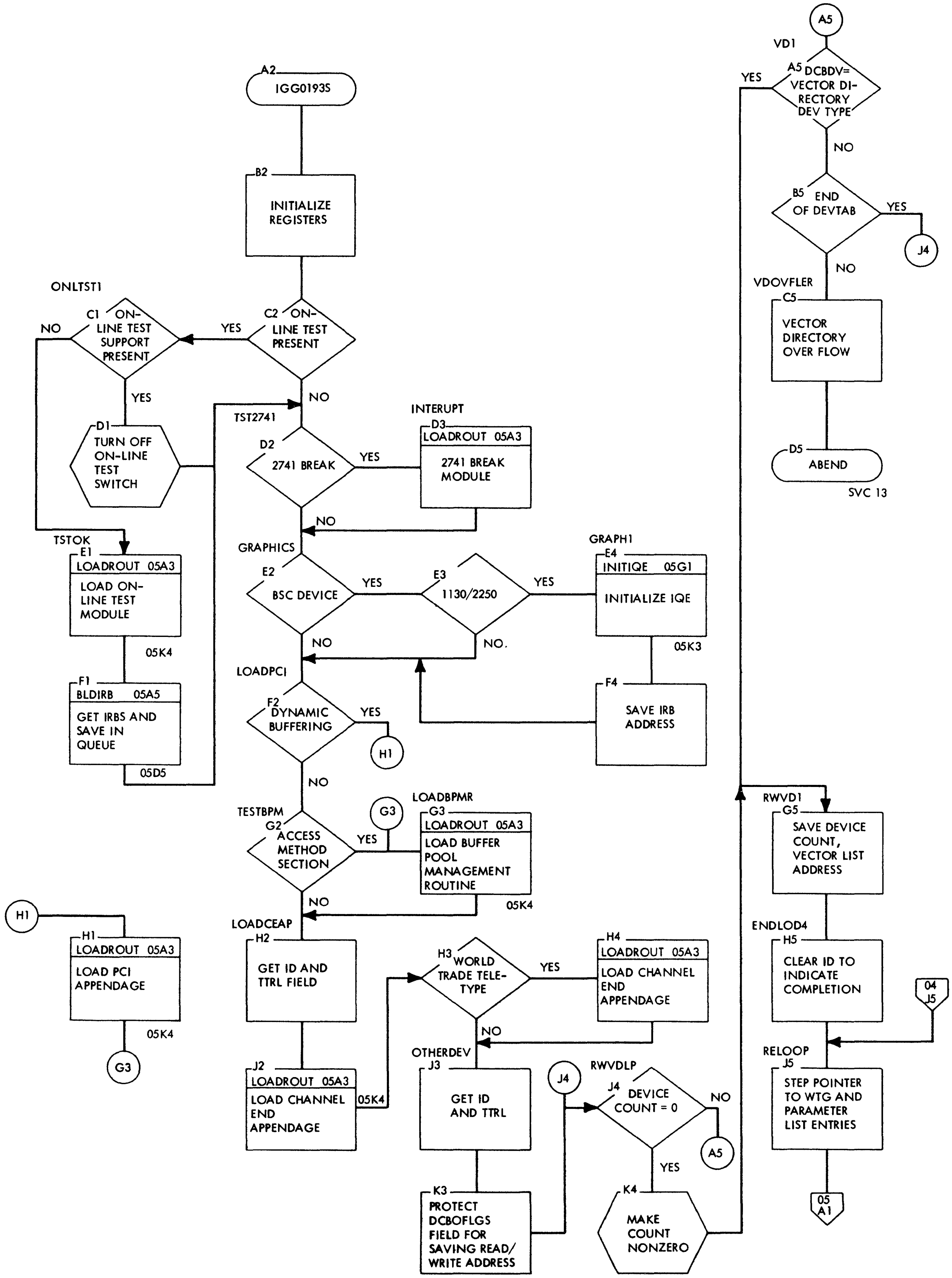


Chart 05. BTAM Open Executor (IGG0193S) (Load 4) (Part 2 of 2)

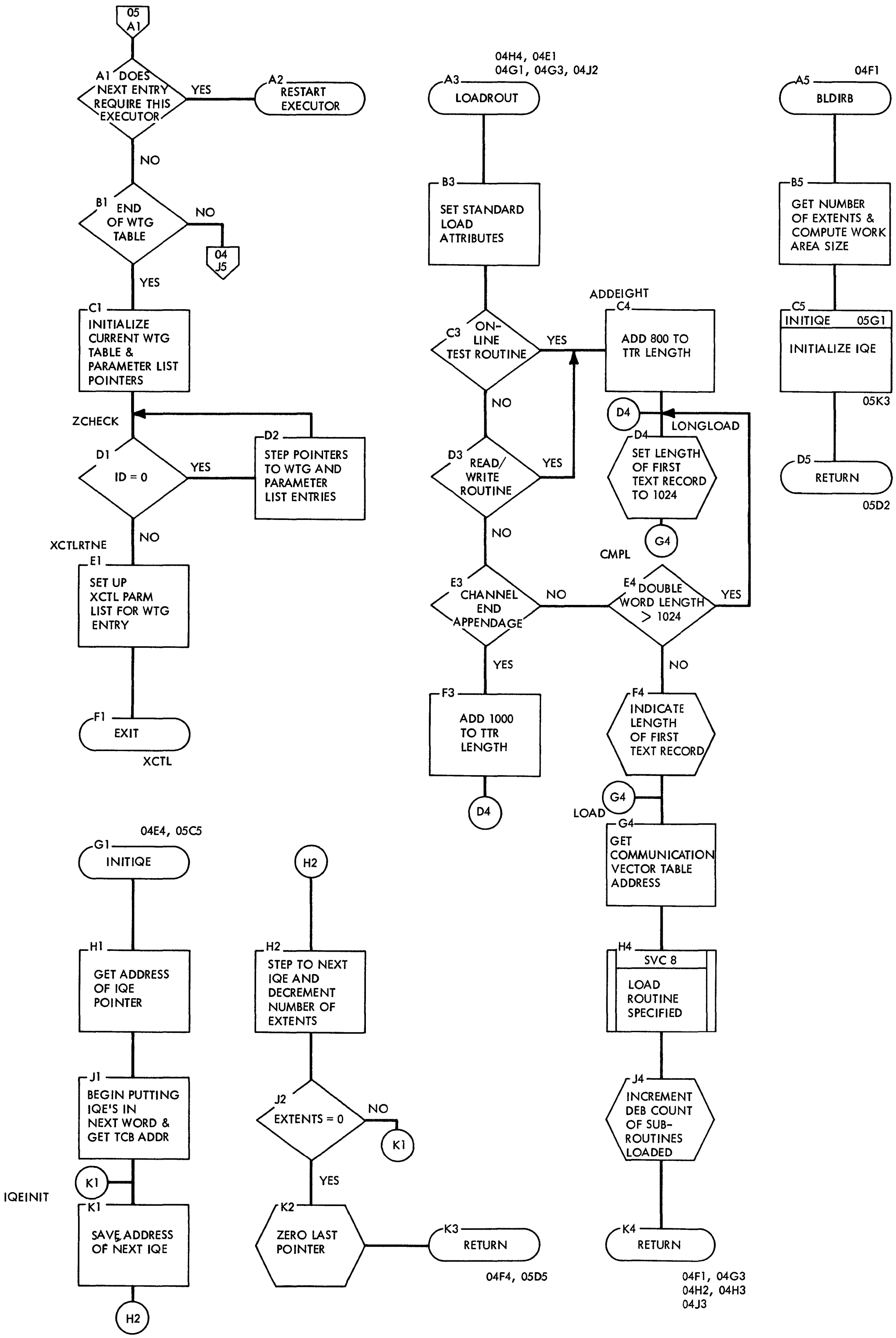
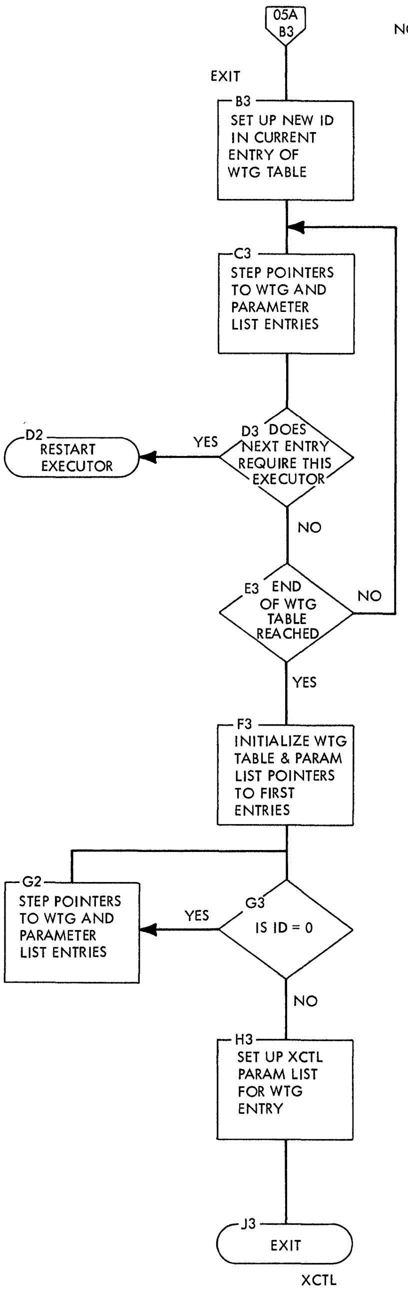


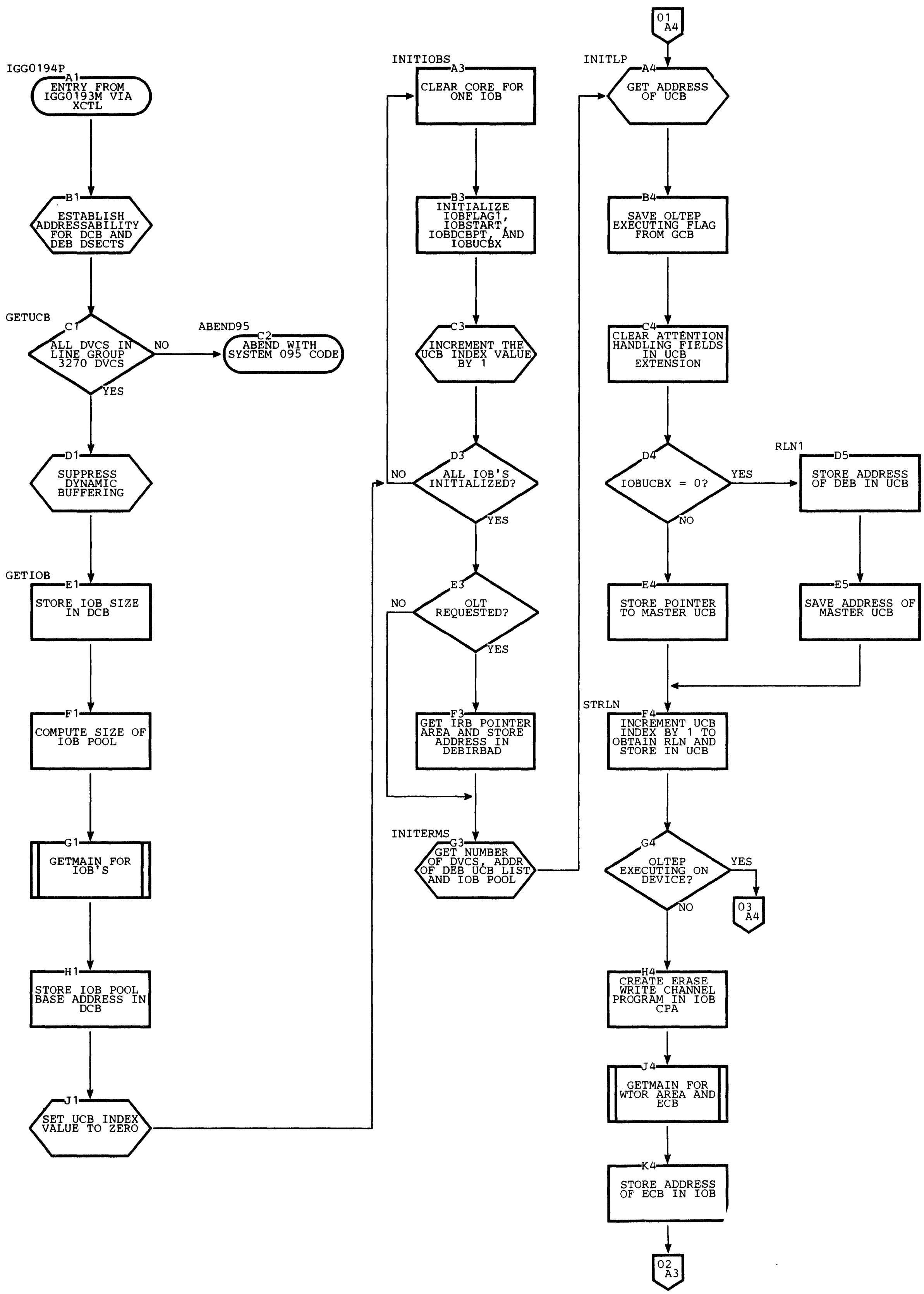
Chart 05A. BTAM Open Executor and Local 3270 BTAM Open Executor (Common to all Loads) WTG Table Routine

THIS CODE OCCURS IN ALL BTAM OPEN EXECUTOR MODULES (IGG0193M, IGG0194N, IGG0193Q, IGG0193S, AND IGG0194P):



NOTE: LAST MODULES (IGG0193S AND IGG0194Q) SET NEW ID TO ZERO; OTHER MODULES SET ID OF NEXT MODULE:
IGG0193M SETS 4P (FOR DEVICE CLASS = GRAPHICS)
IGG0193M SETS 4N (FOR DEVICE CLASS = COMMUNICATION EQUIPMENT)
IGG0194N SETS 3Q
IGG0193Q SETS 3S
IGG0194P SETS 4Q

NOTE: WTG = WHERE-TO-GO TABLE


IGG0194P
A1
ENTRY FROM IGG0193M VIA XCTL
B1
ESTABLISH ADDRESSABILITY FOR DCB AND DEB DSECTS
GETUCB
C1
ALL DVCS IN LINE GROUP 3270 DVCS
NO
YES
ABEND95
C2
ABEND WITH SYSTEM 095 CODE
D1
SUPPRESS DYNAMIC BUFFERING
GETIOB
E1
STORE IOB SIZE IN DCB
F1
COMPUTE SIZE OF IOB POOL
G1
GETMAIN FOR IOB'S
H1
STORE IOB POOL BASE ADDRESS IN DCB
J1
SET UCB INDEX VALUE TO ZERO
INITIOBS
A3
CLEAR CORE FOR ONE IOB
B3
INITIALIZE IOBFLAG1, IOBSTART, IOBDCBPT, AND IOBUCBX
C3
INCREMENT THE UCB INDEX VALUE BY 1
D3
ALL IOB'S INITIALIZED?
NO
YES
E3
OLT REQUESTED?
NO
YES
F3
GET IRB POINTER AREA AND STORE ADDRESS IN DEBIRBAD
INITERMS
G3
GET NUMBER OF DVCS, ADDR OF DEB UCB LIST AND IOB POOL
01
A4
INITLP
A4
GET ADDRESS OF UCB
B4
SAVE OLTEP EXECUTING FLAG FROM GCB
C4
CLEAR ATTENTION HANDLING FIELDS IN UCB EXTENSION
D4
IOBUCBX = 0?
YES
NO
RLN1
D5
STORE ADDRESS OF DEB IN UCB
E4
STORE POINTER TO MASTER UCB
E5
SAVE ADDRESS OF MASTER UCB
STRLN
F4
INCREMENT UCB INDEX BY 1 TO OBTAIN RLN AND STORE IN UCB
G4
OLTEP EXECUTING ON DEVICE?
YES
NO
03
A4
H4
CREATE ERASE WRITE CHANNEL PROGRAM IN IOB CPA
J4
GETMAIN FOR WTOR AREA AND ECB
K4
STORE ADDRESS OF ECB IN IOB
02
A3

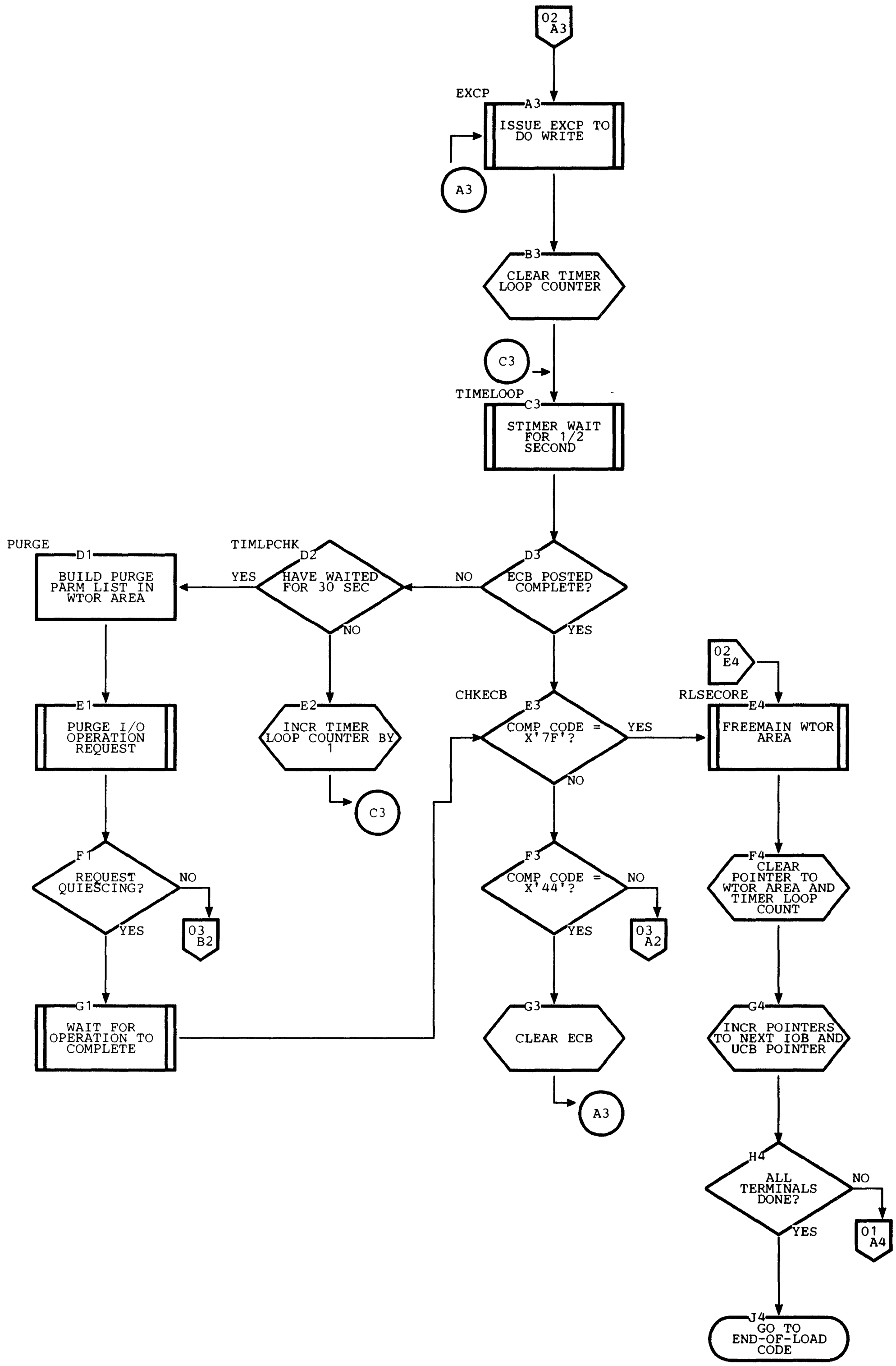

02 A3
EXCP
A3
ISSUE EXCP TO DO WRITE
A3
B3
CLEAR TIMER LOOP COUNTER
C3
TIMELOOP
C3
STIMER WAIT FOR 1/2 SECOND
PURGE
D1
BUILD PURGE PARM LIST IN WTOR AREA
TIMLPCHK
D2
HAVE WAITED FOR 30 SEC
YES
NO
D3
ECB POSTED COMPLETE?
NO
YES
E1
PURGE I/O OPERATION REQUEST
E2
INCR TIMER LOOP COUNTER BY 1
C3
CHKECB
E3
COMP CODE = X'7F'?
YES
NO
02 E4
RLSECORE
E4
FREEMAIN WTOR AREA
F1
REQUEST QUIESCING?
NO
YES
03 B2
F3
COMP CODE = X'44'?
NO
YES
03 A2
F4
CLEAR POINTER TO WTOR AREA AND TIMER LOOP COUNT
G1
WAIT FOR OPERATION TO COMPLETE
G3
CLEAR ECB
A3
G4
INCR POINTERS TO NEXT IOB AND UCB POINTER
H4
ALL TERMINALS DONE?
NO
YES
01 A4
J4
GO TO END-OF-LOAD CODE

Chart C5D. Local 3270 BTAM Open Executor (IGG0194P) (Part 3 of 3)

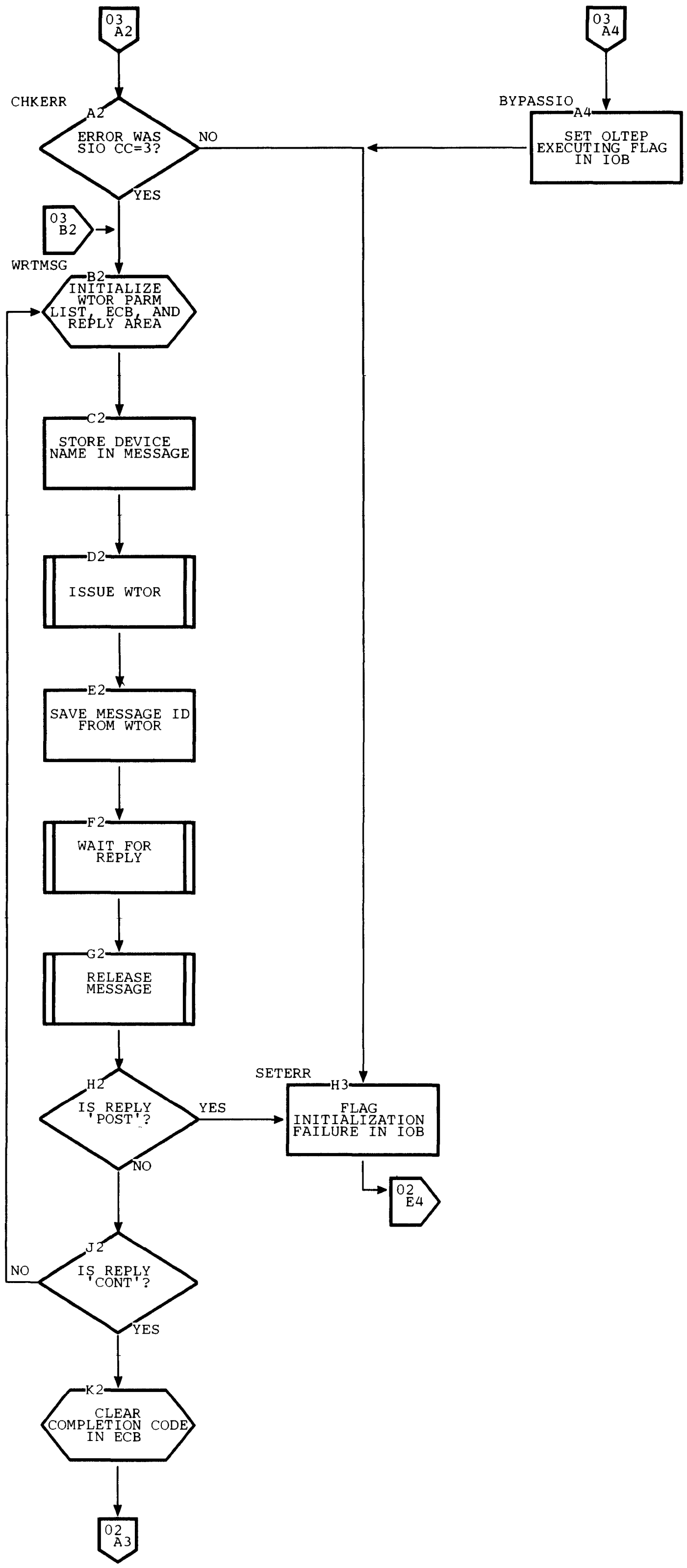


Chart 05E. Local 3270 BTAM Open Executor (IGG0194Q) (Part 1 of 2)

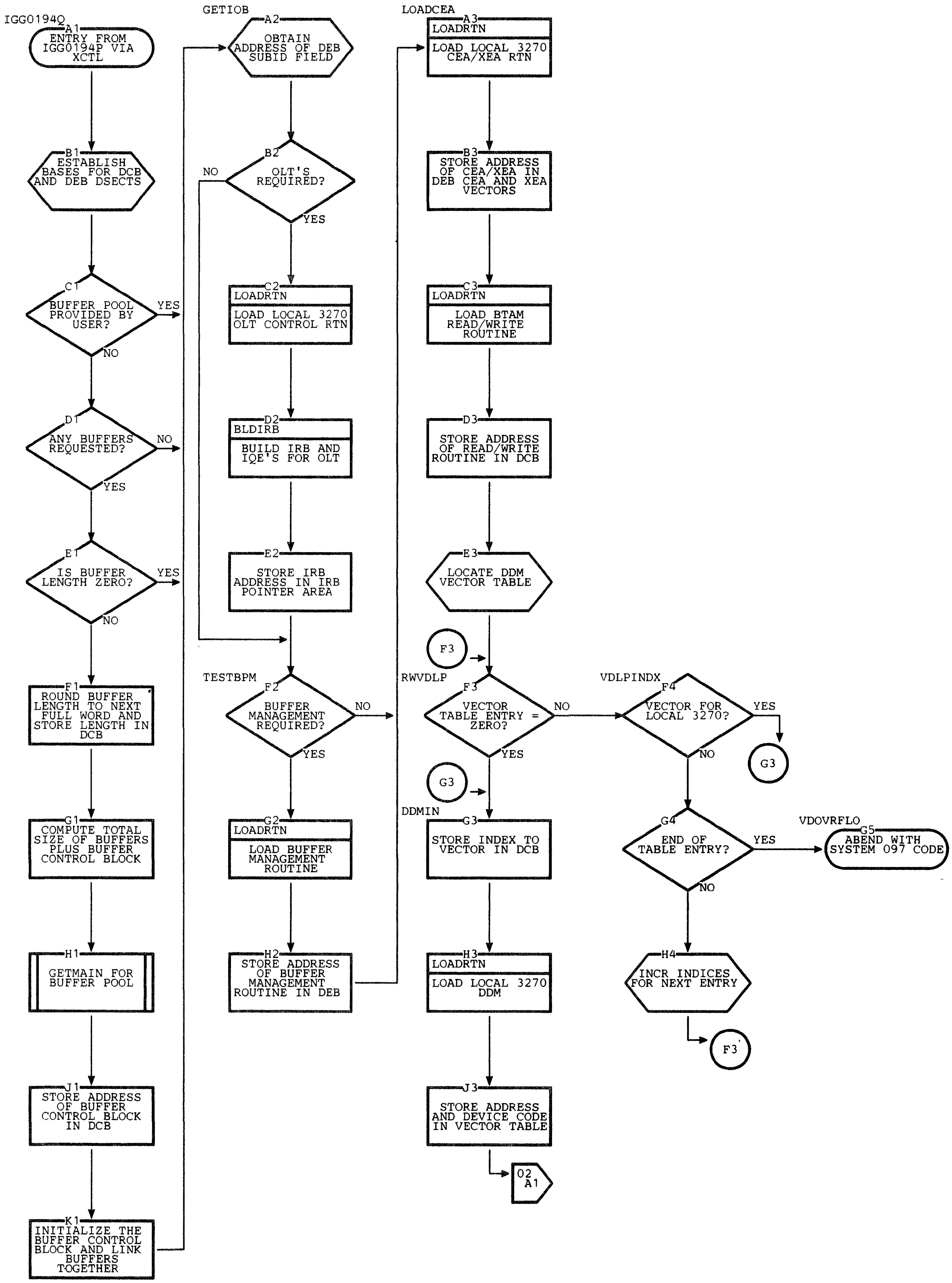


Chart 05F. Local 3270 BTAM Open Executor (IGG0194Q) (Part 2 of 2)

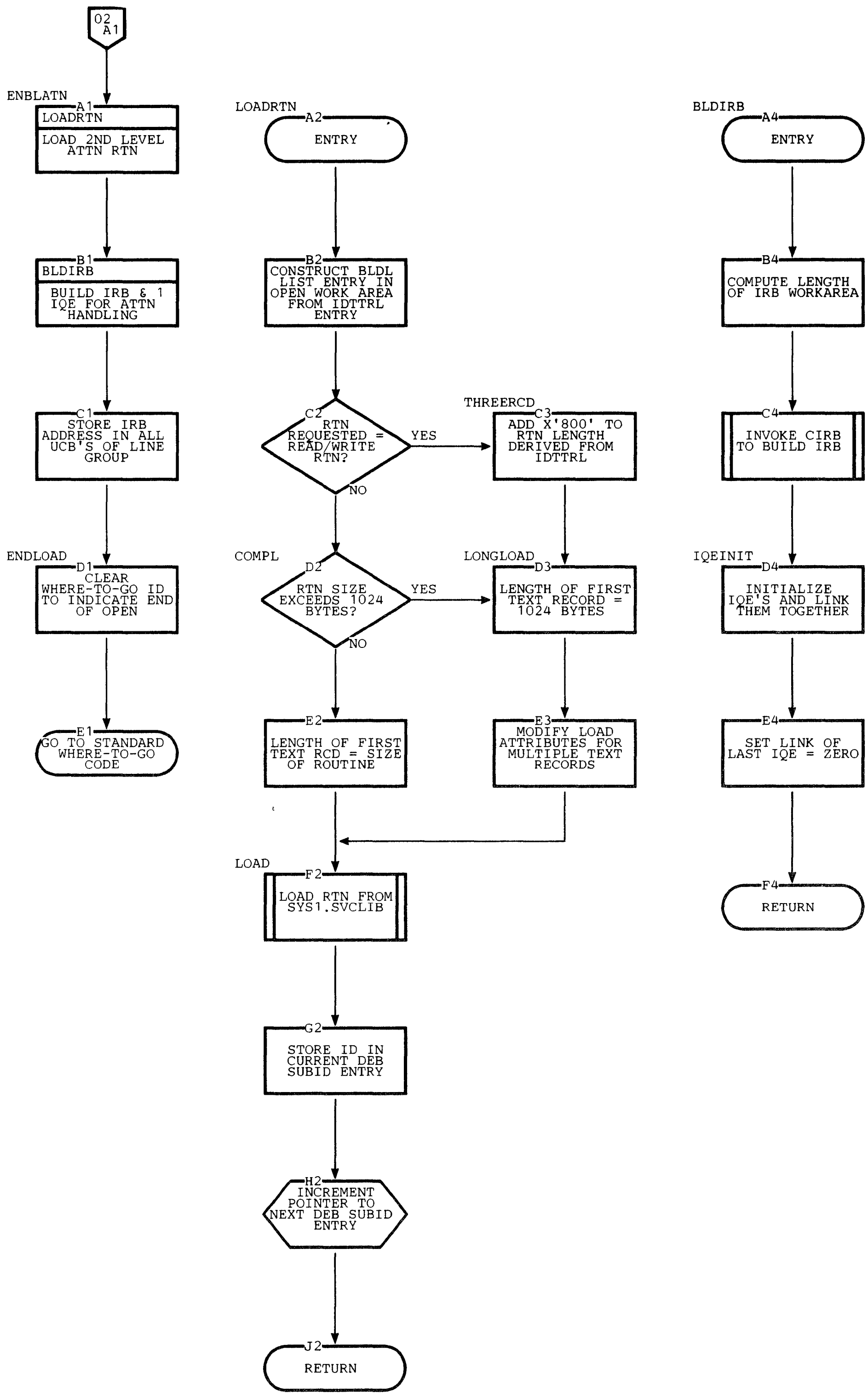


Chart 06. Buffer (IGG019MS)

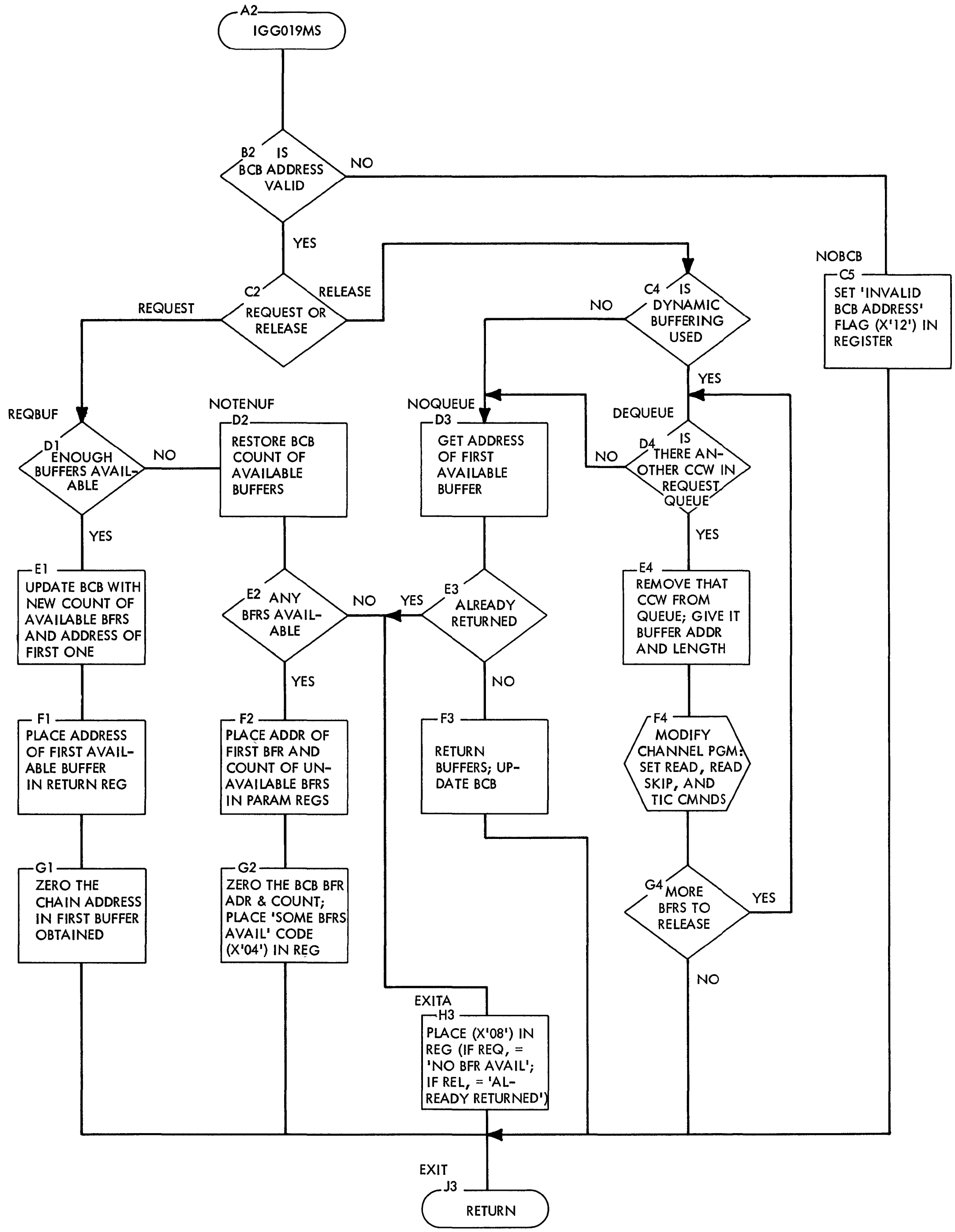


Chart 07. Line Open (IECTLOPN)

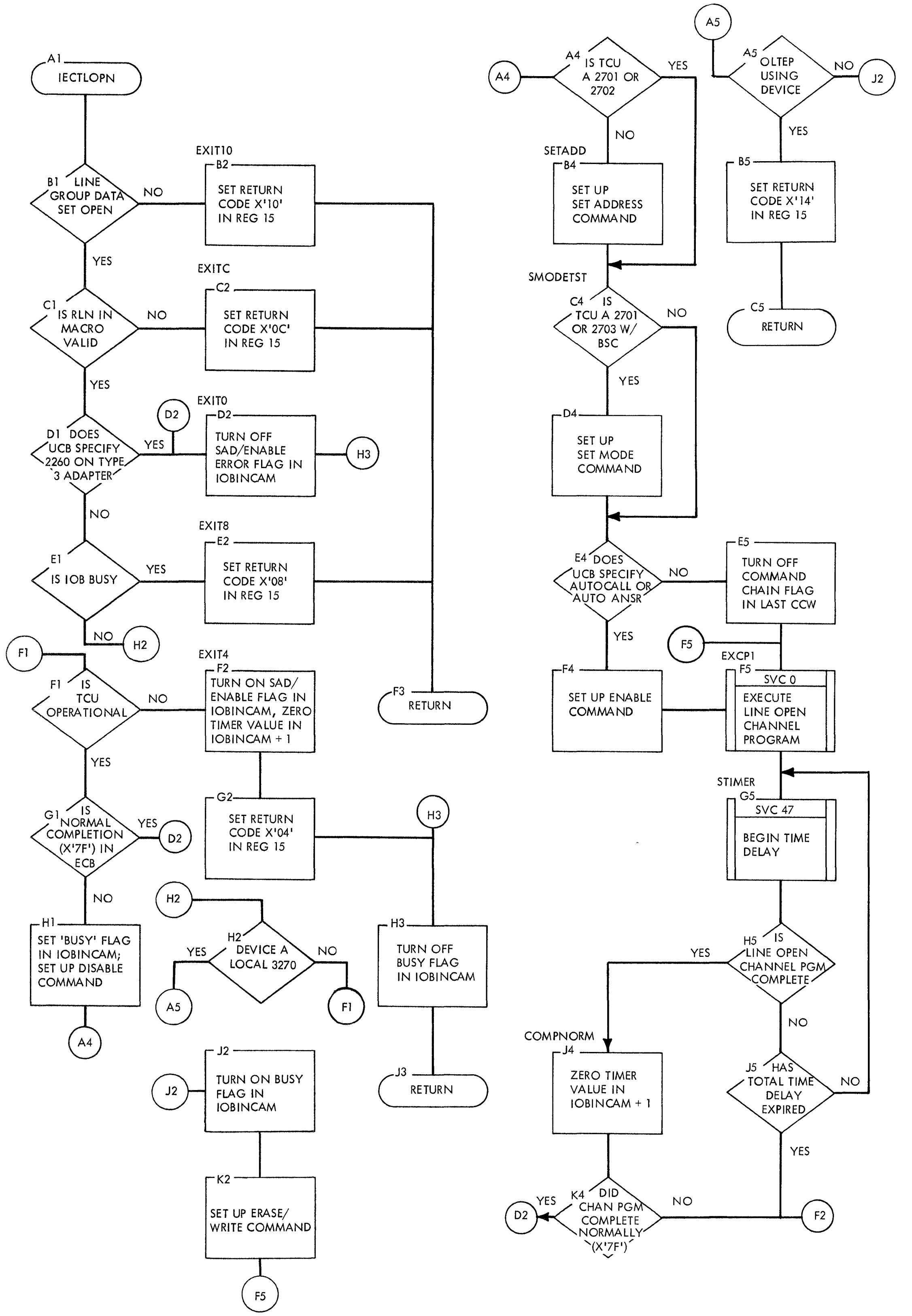


Chart 08. Translate (IECTTRNS)

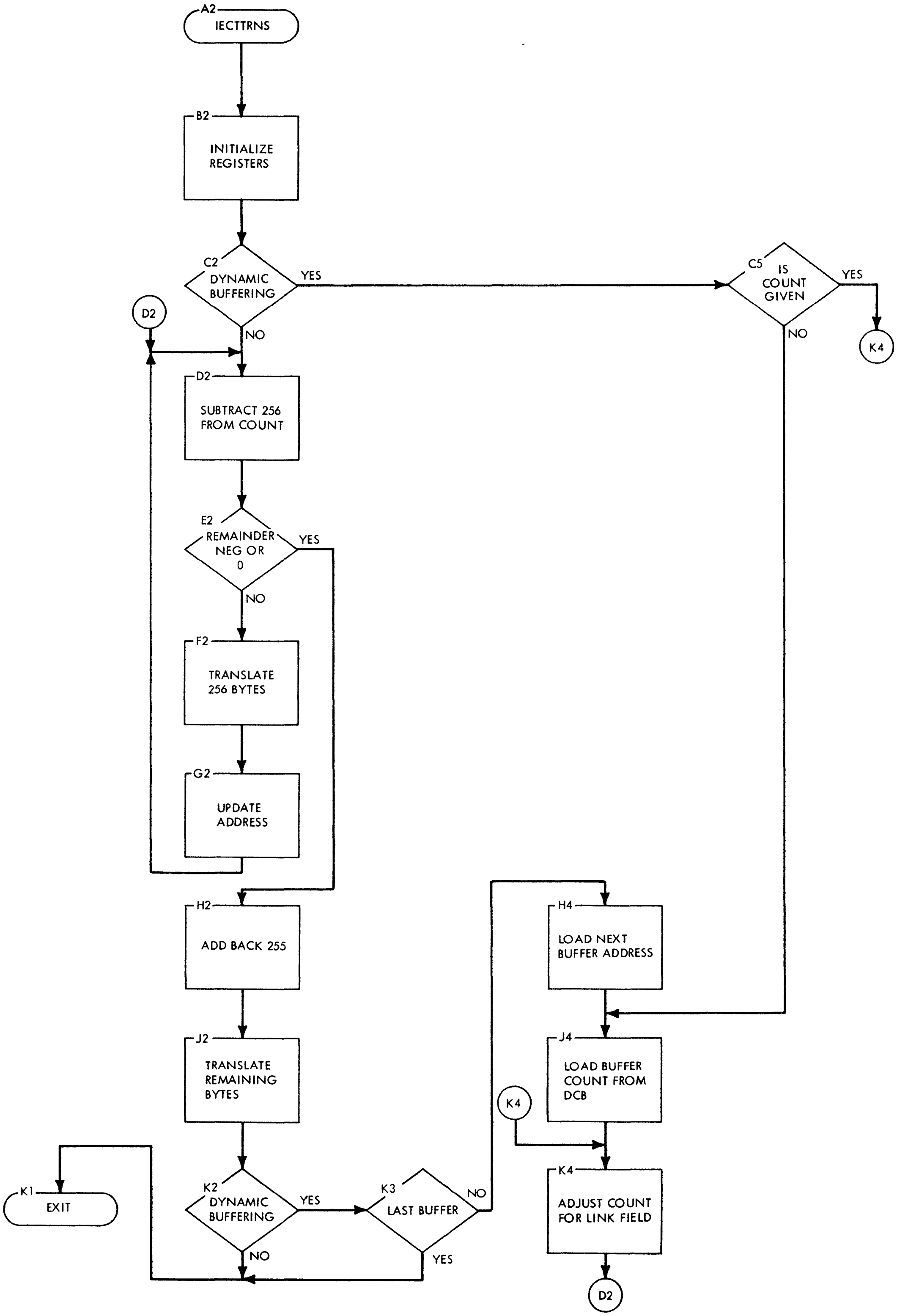


Chart 09. REQBUF/RELBUF (IGC058)

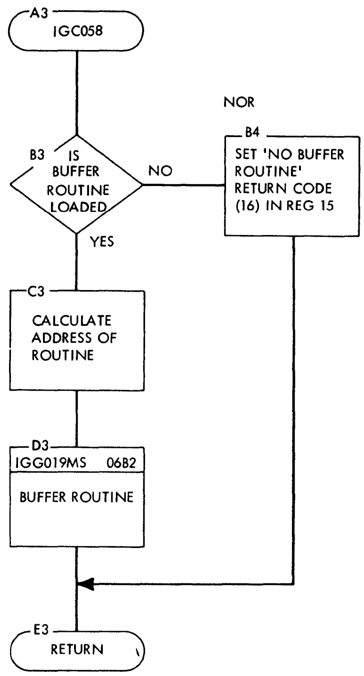


Chart 10. BTAM Close Executor (IGG0203M)

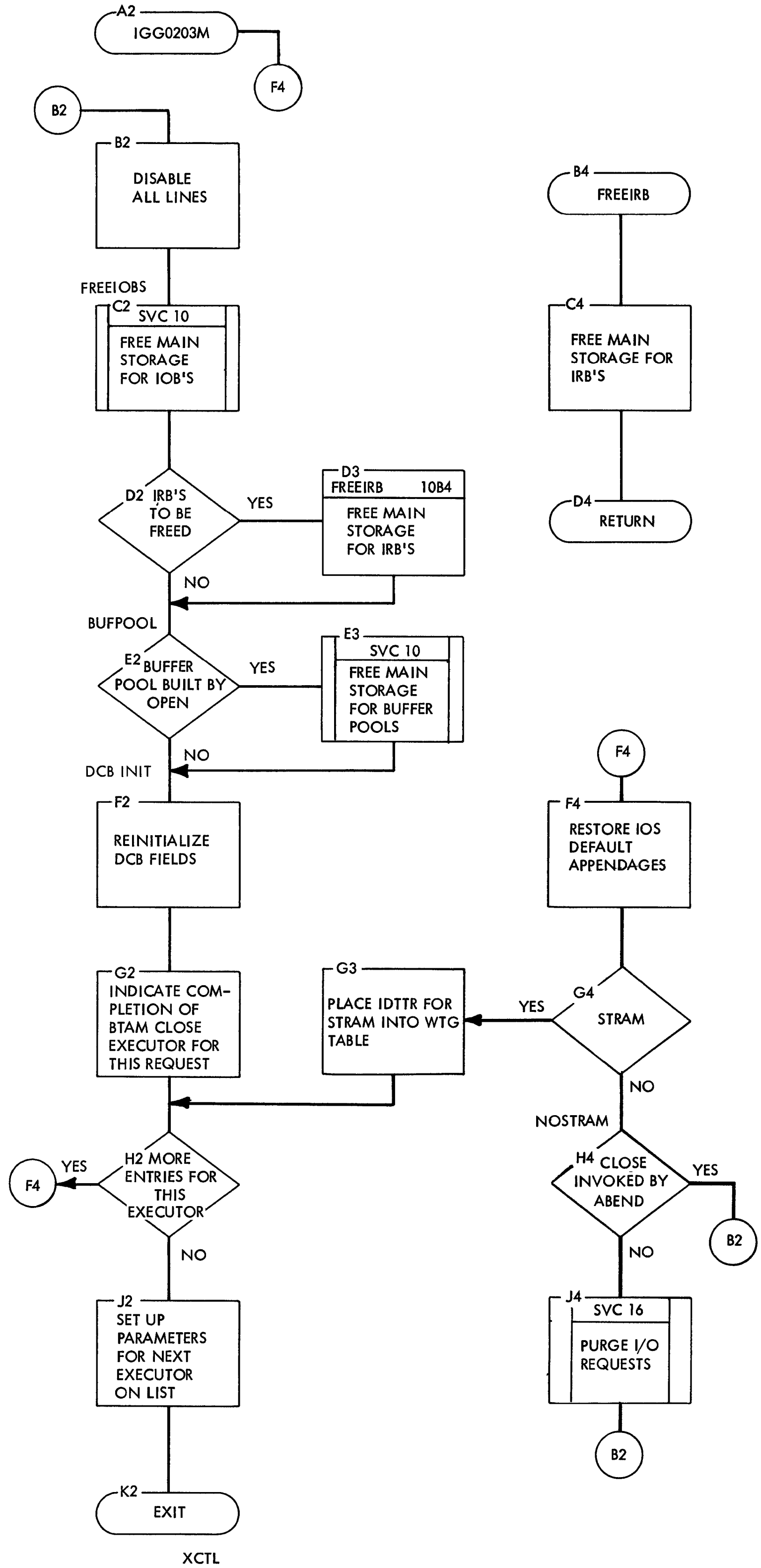


Chart A1. BTAM Read/Write (IGG019MA) (Part 1 of 4)

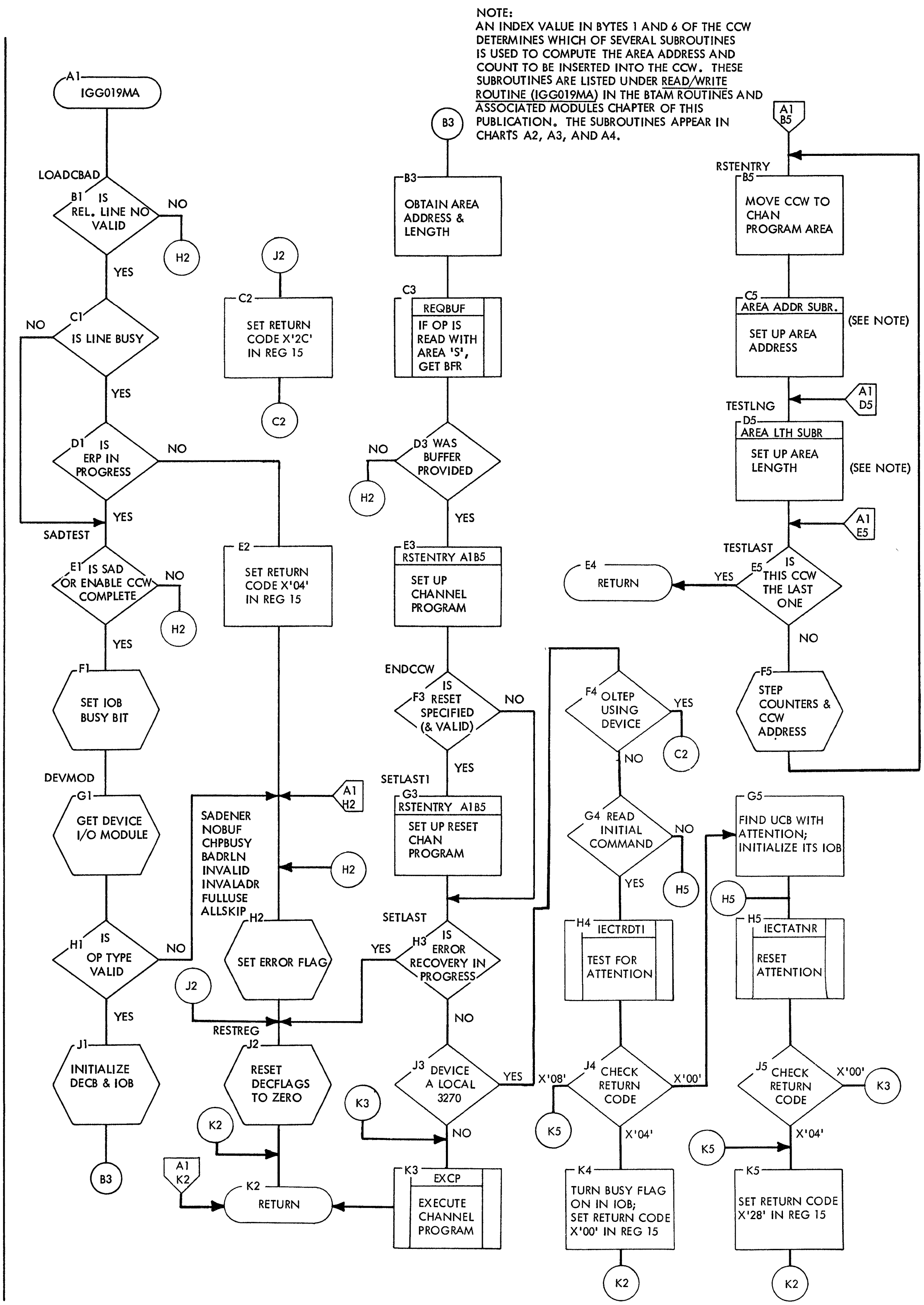


Chart A2. BTAM Read/Write (IGG019MA) (Part 2 of 4)

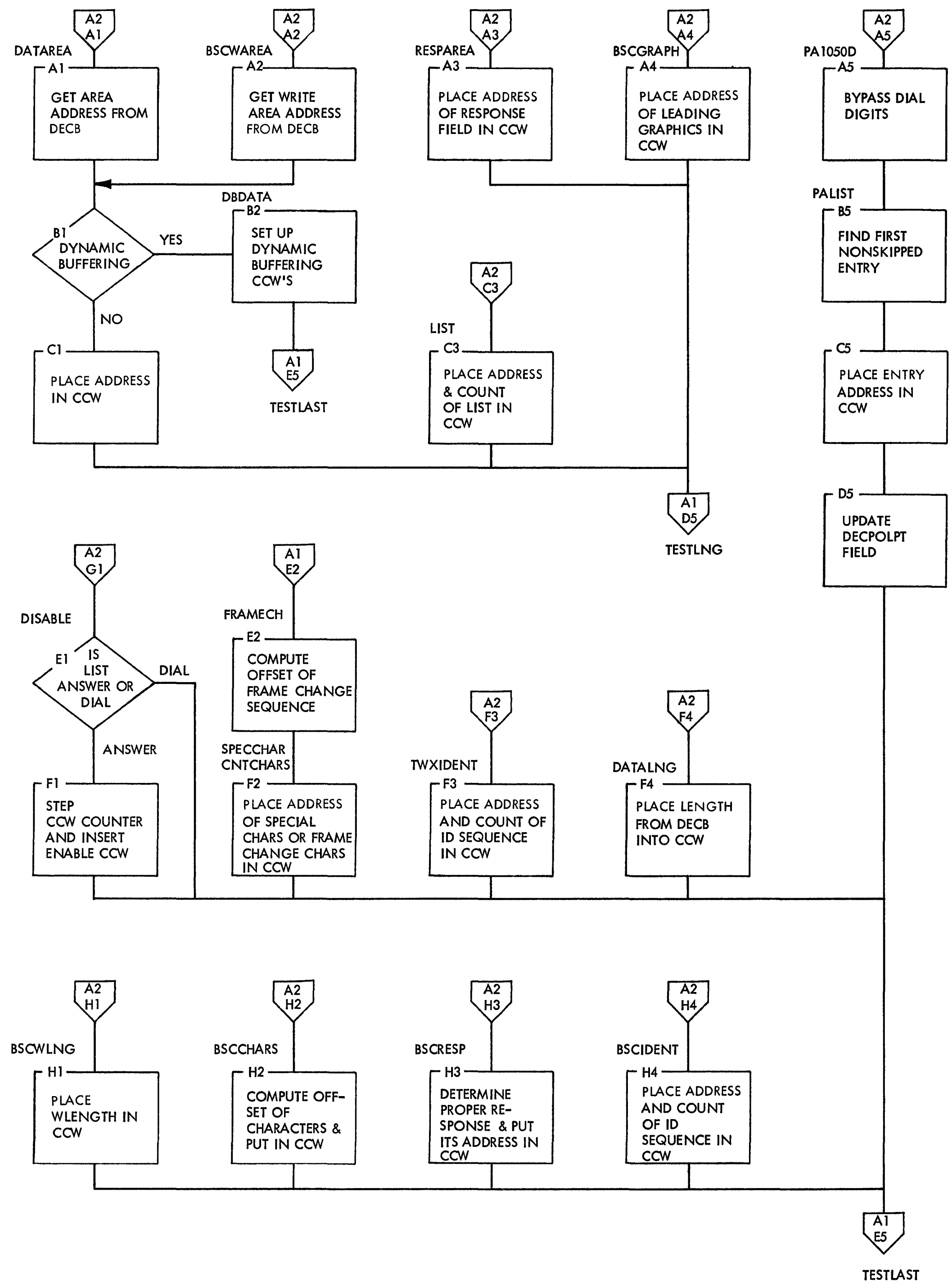
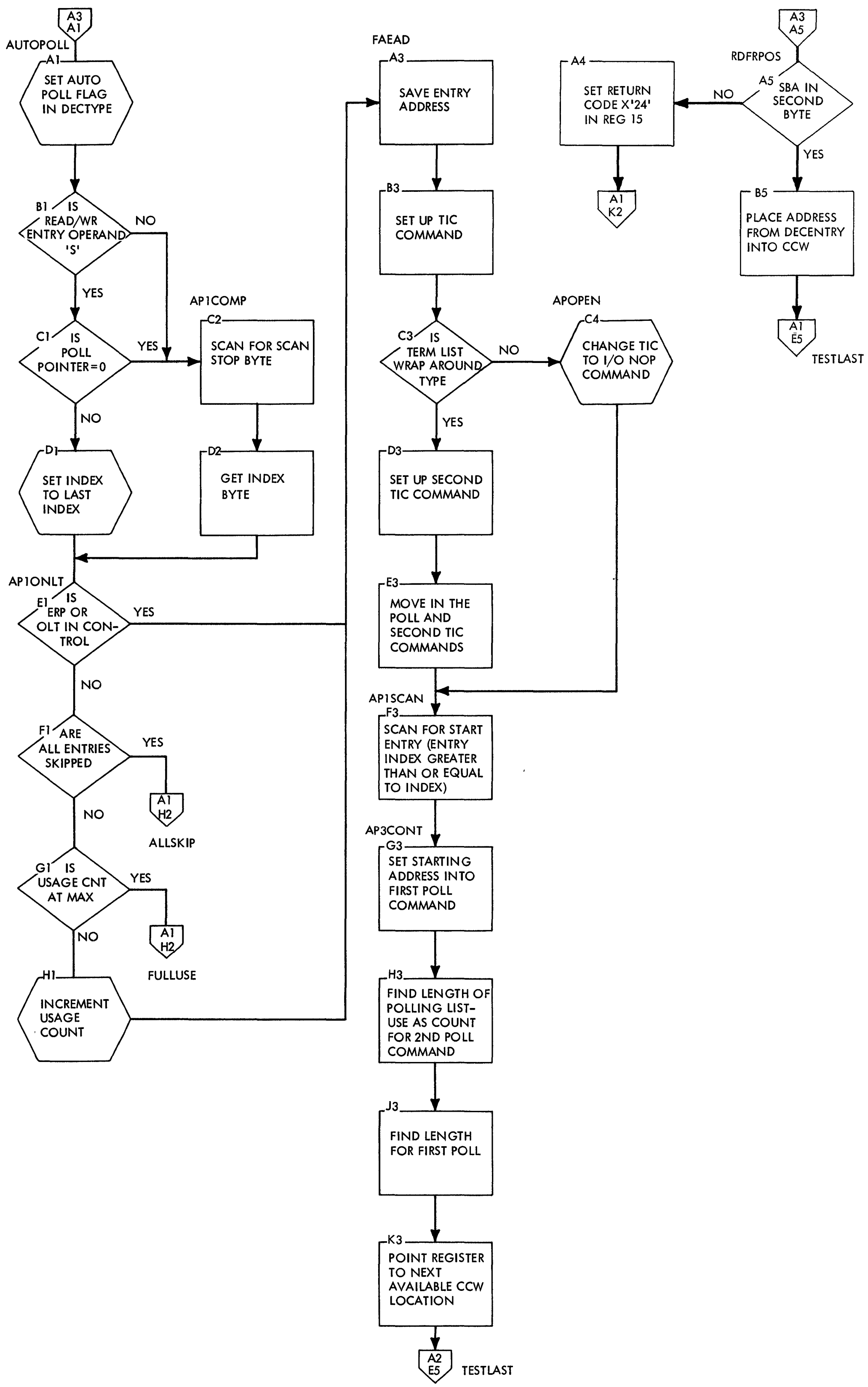


Chart A3. BTAM Read/Write (IGG019MA) (Part 3 of 4)



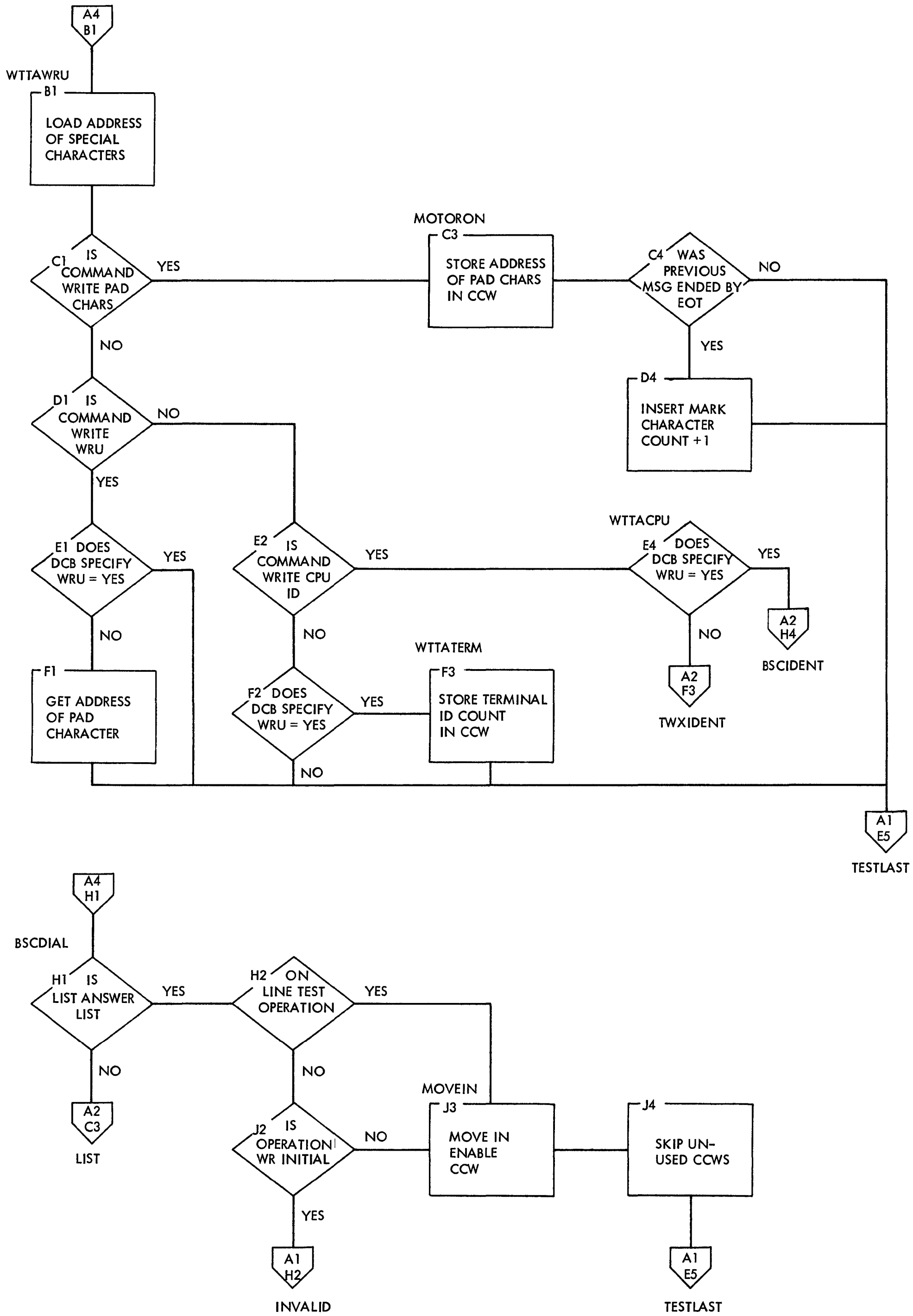

A4 B1
WTTAWRU
B1
LOAD ADDRESS OF SPECIAL CHARACTERS
C1 IS COMMAND WRITE PAD CHARS
YES
NO
MOTORON
C3
STORE ADDRESS OF PAD CHARS IN CCW
C4 WAS PREVIOUS MSG ENDED BY EOT
NO
YES
D4
INSERT MARK CHARACTER COUNT +1
D1 IS COMMAND WRITE WRU
NO
YES
E1 DOES DCB SPECIFY WRU = YES
YES
NO
E2 IS COMMAND WRITE CPU ID
YES
NO
WTTACPU
E4 DOES DCB SPECIFY WRU = YES
YES
NO
A2 H4
BSCIDENT
A2 F3
TWXIDENT
F1
GET ADDRESS OF PAD CHARACTER
F2 DOES DCB SPECIFY WRU = YES
YES
NO
WTTATERM
F3
STORE TERMINAL ID COUNT IN CCW
A1 E5
TESTLAST
A4 H1
BSCDIAL
H1 IS LIST ANSWER LIST
YES
NO
A2 C3
LIST
H2 ON LINE TEST OPERATION
YES
NO
J2 IS OPERATION WR INITIAL
NO
YES
A1 H2
INVALID
MOVEIN
J3
MOVE IN ENABLE CCW
J4
SKIP UN-USED CCWS
A1 E5
TESTLAST

Chart B1. Channel End Appendage (IGG019MB) (Part 1 of 6)

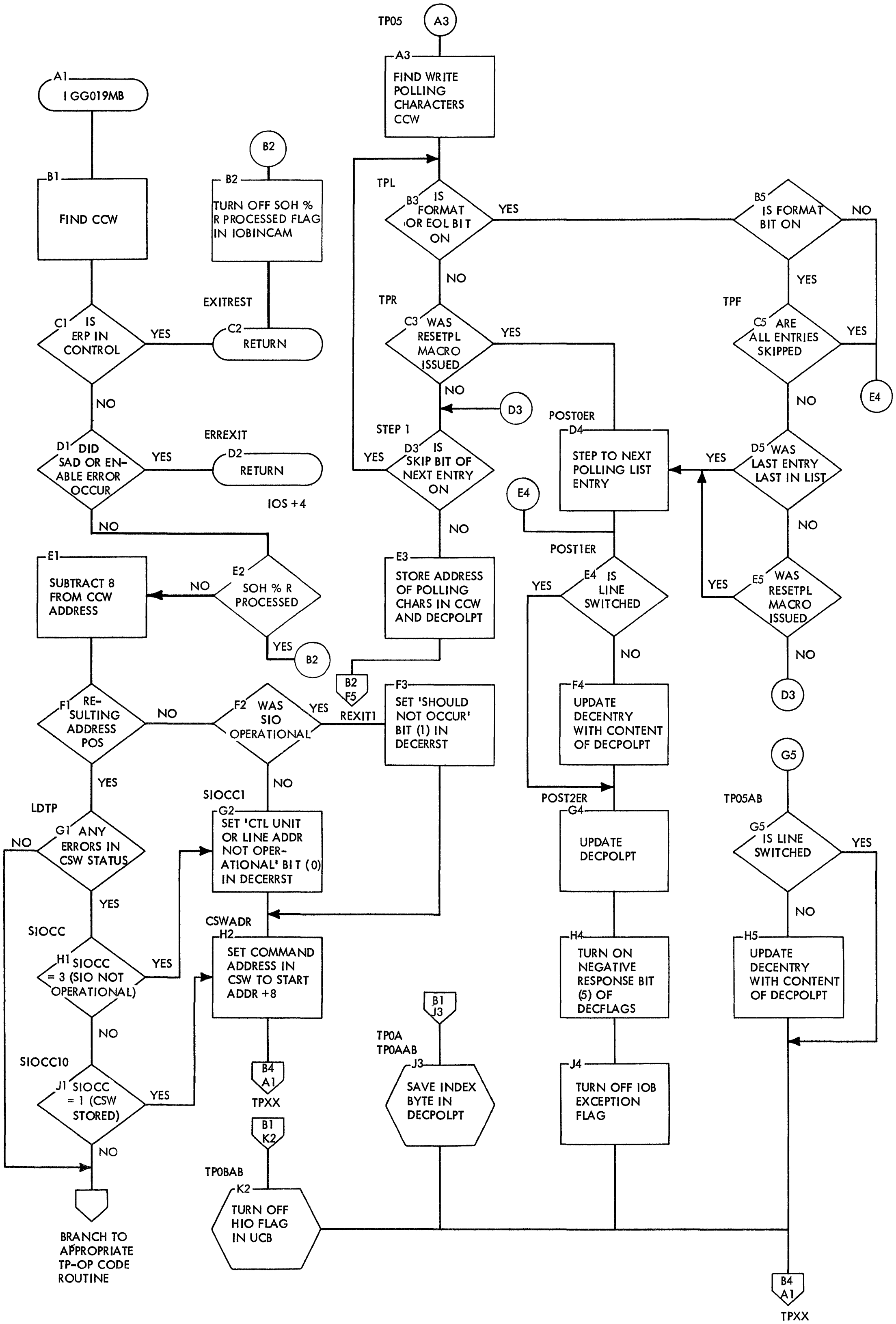


Chart B2. Channel End Appendage (IGG019MB) (Part 2 of 6)

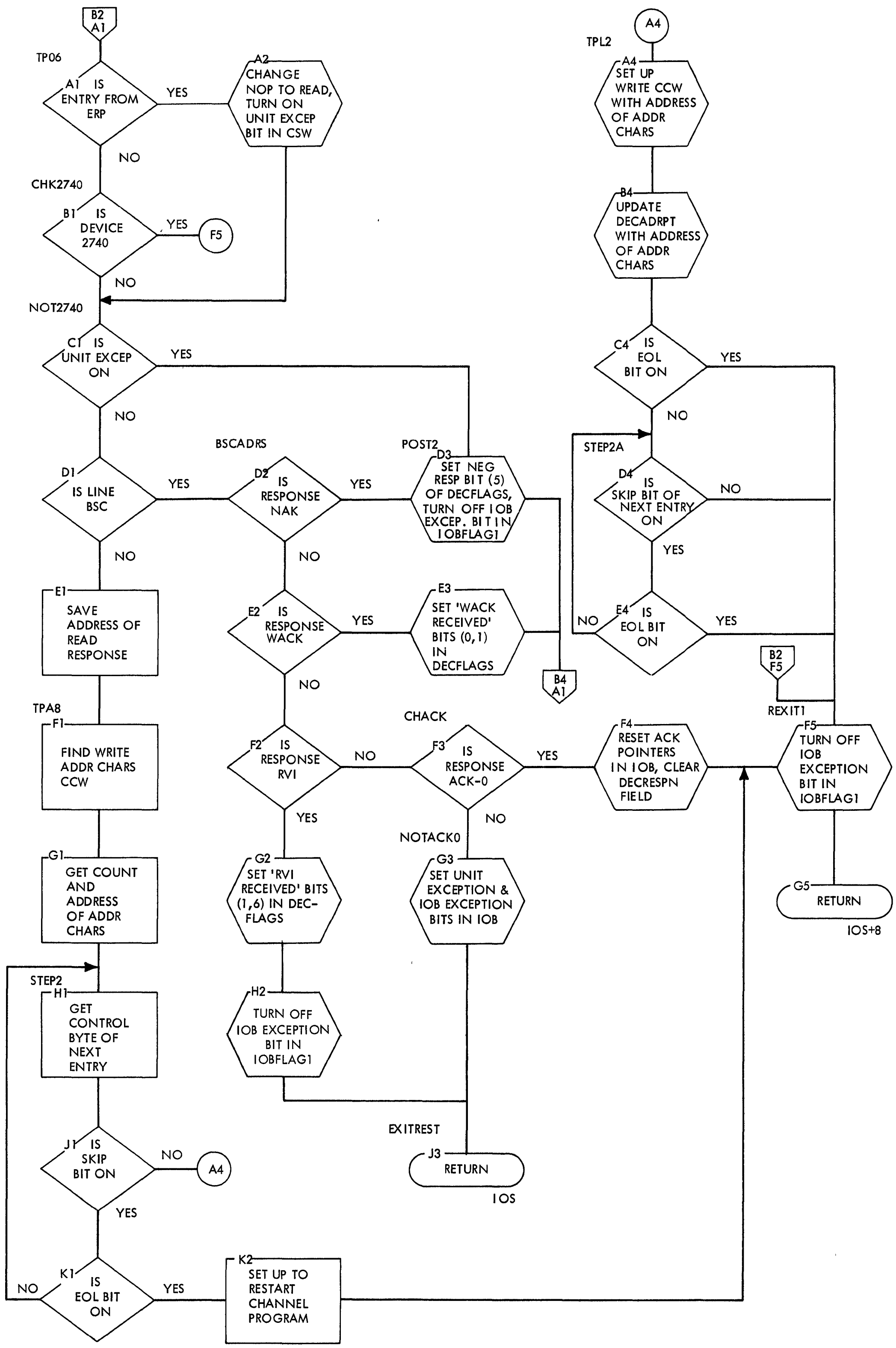


Chart B3. Channel End Appendage (IGG019MB) (Part 3 of 6)

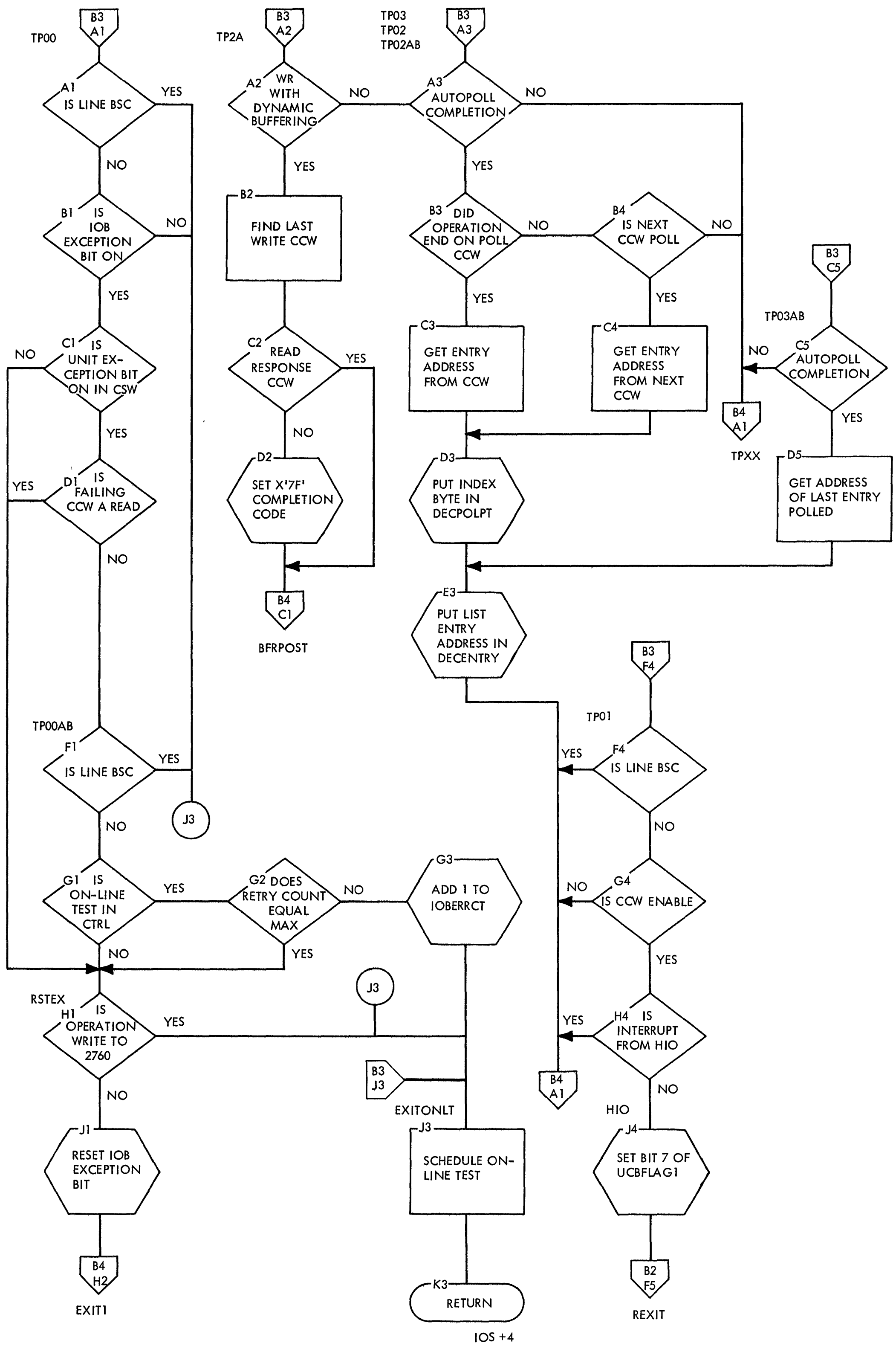


Chart B4. Channel End Appendage (IGG019MB) (Part 4 of 6)

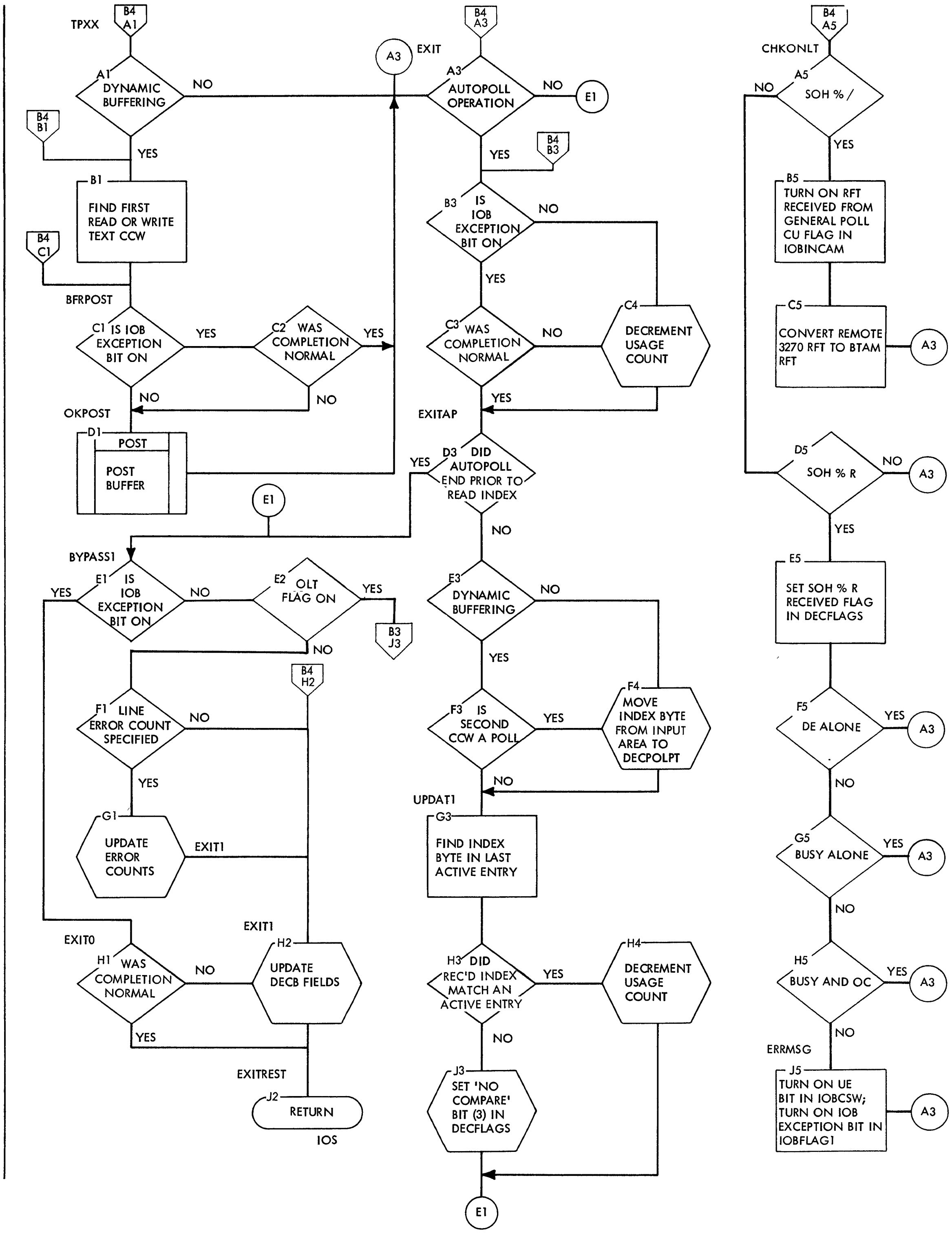


Chart B5. Channel End Appendage (IGG019MB) (Part 5 of 6)

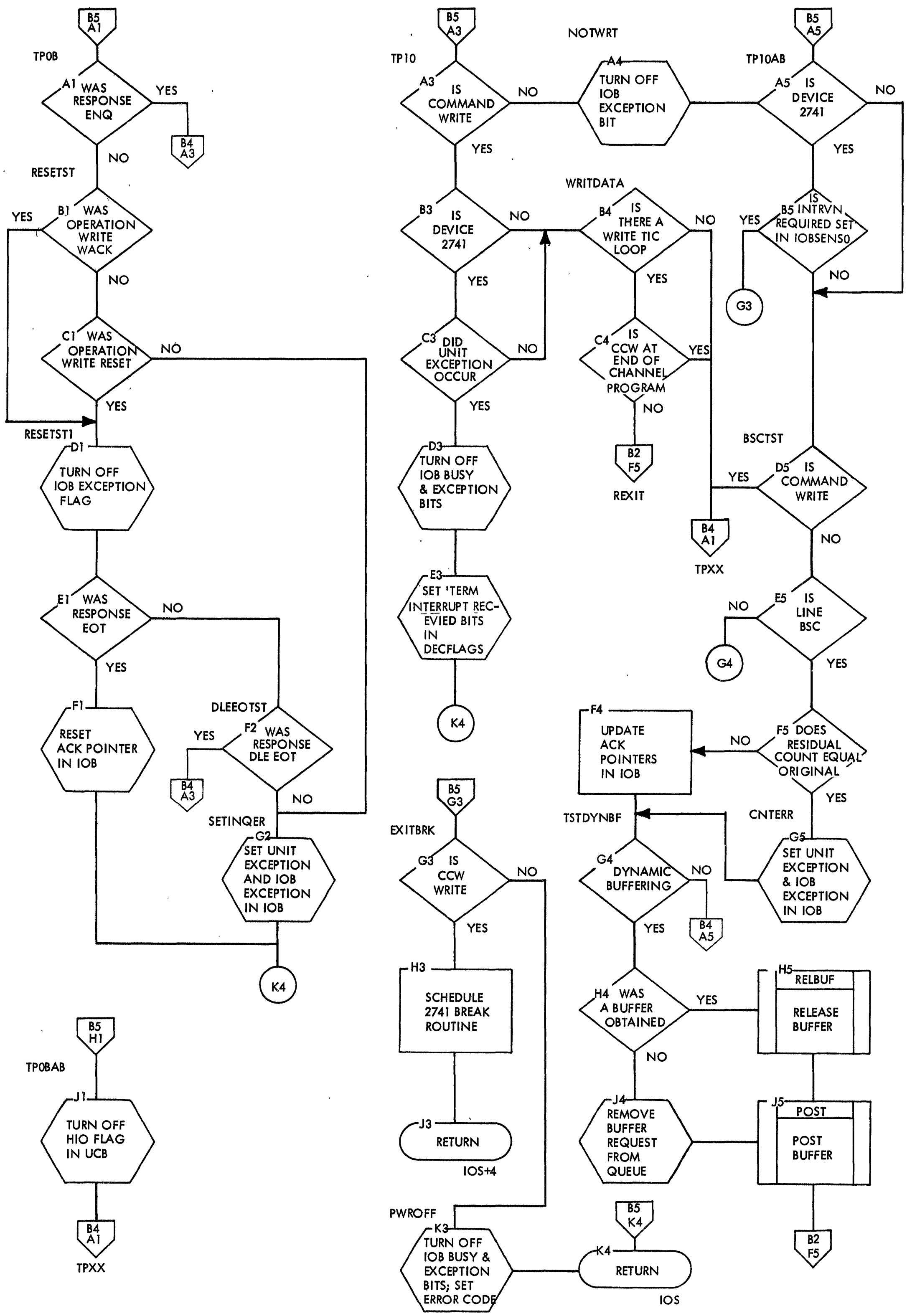
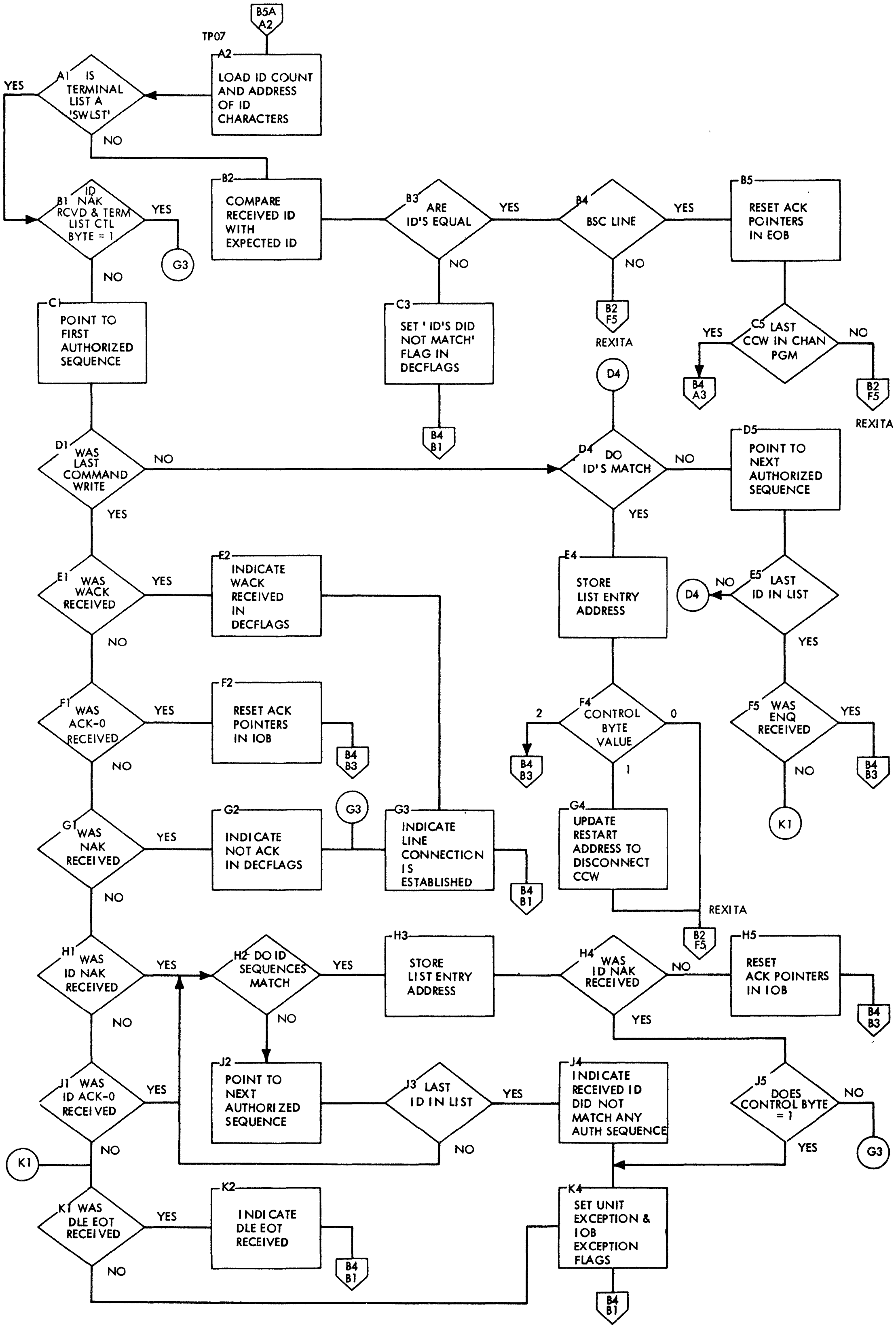
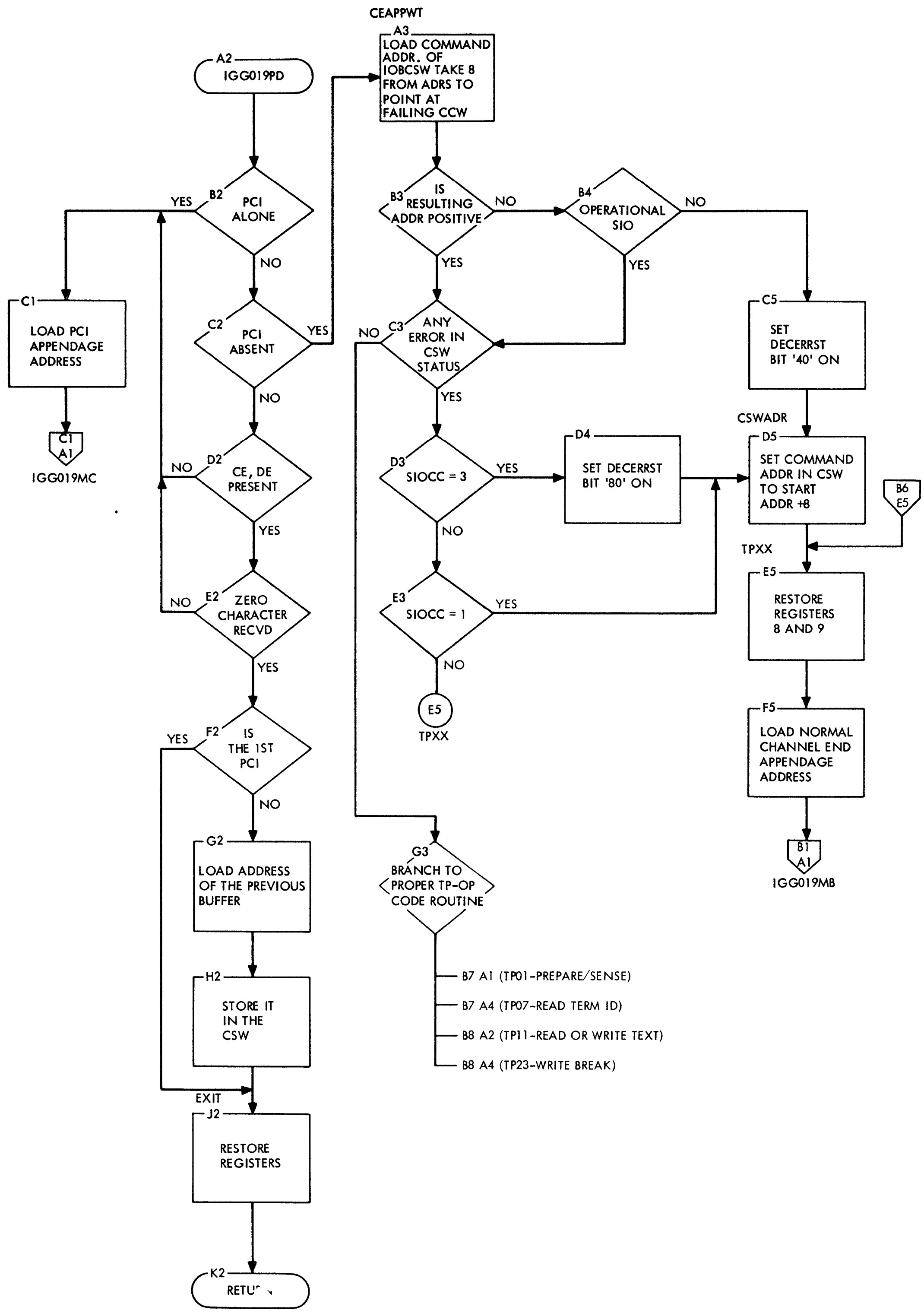


Chart B5A. Channel End Appendage (IGG019MB) (Part 6 of 6)




CEAPPWT
A3
LOAD COMMAND ADDR. OF IOBCSW TAKE 8 FROM ADRS TO POINT AT FAILING CCW
A2
IGG019PD
B2
PCI ALONE
YES
NO
B3
IS RESULTING ADDR POSITIVE
NO
YES
B4
OPERATIONAL SIO
NO
YES
C1
LOAD PCI APPENDAGE ADDRESS
C1 A1
IGG019MC
C2
PCI ABSENT
YES
NO
C3
ANY ERROR IN CSW STATUS
NO
YES
C5
SET DECERRST BIT '40' ON
CSWADR
D2
CE, DE PRESENT
NO
YES
D3
SIOCC = 3
YES
NO
D4
SET DECERRST BIT '80' ON
D5
SET COMMAND ADDR IN CSW TO START ADDR +8
B6 E5
TPXX
E2
ZERO CHARACTER RECVD
NO
YES
E3
SIOCC = 1
YES
NO
E5
RESTORE REGISTERS 8 AND 9
E5
TPXX
F2
IS THE 1ST PCI
YES
NO
F5
LOAD NORMAL CHANNEL END APPENDAGE ADDRESS
B1 A1
IGG019MB
G2
LOAD ADDRESS OF THE PREVIOUS BUFFER
G3
BRANCH TO PROPER TP-OP CODE ROUTINE
B7 A1 (TP01-PREPARE/SENSE)
B7 A4 (TP07-READ TERM ID)
B8 A2 (TP11-READ OR WRITE TEXT)
B8 A4 (TP23-WRITE BREAK)
H2
STORE IT IN THE CSW
EXIT
J2
RESTORE REGISTERS
K2

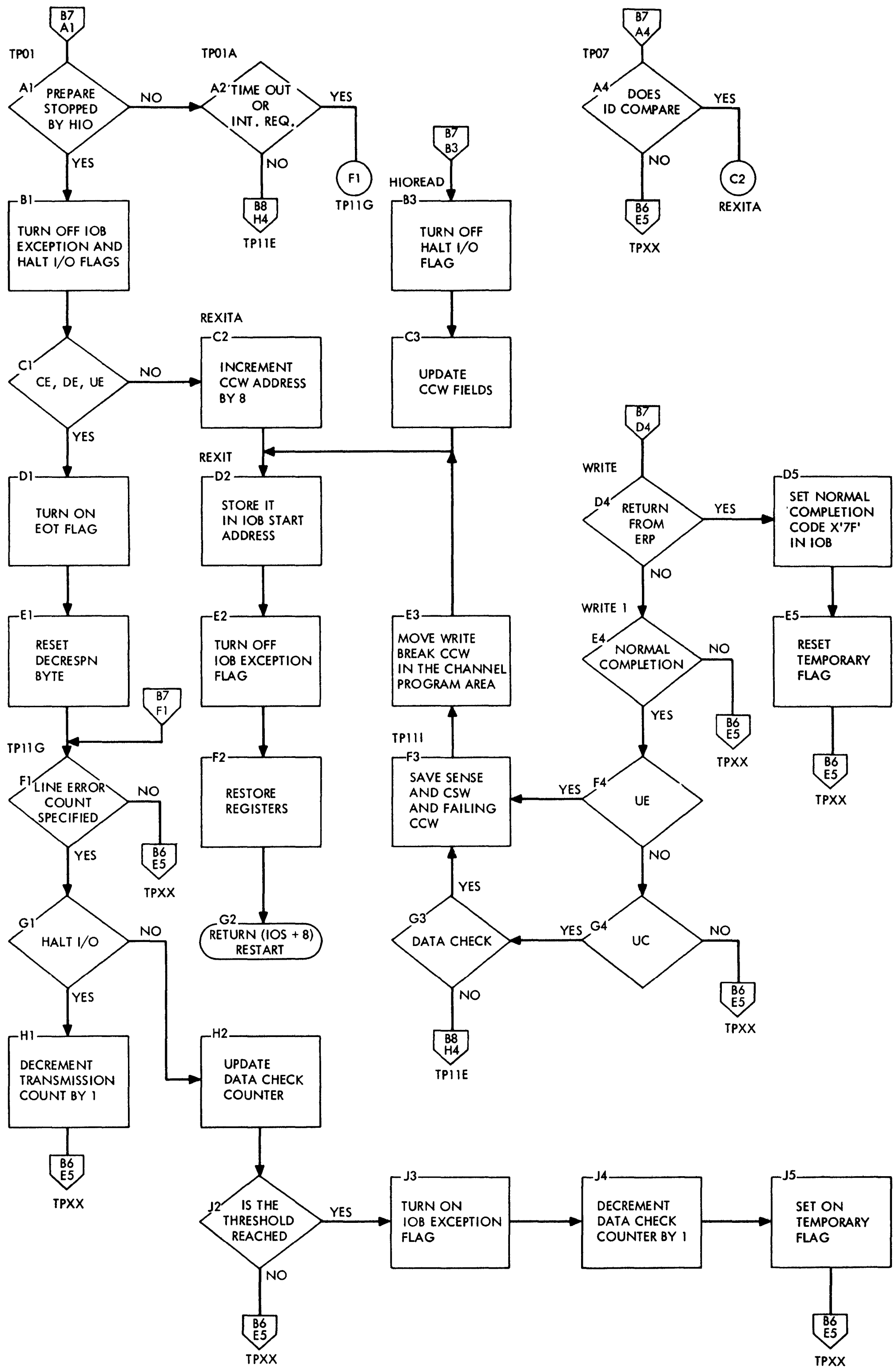

B7 A1
TP01
A1 PREPARE STOPPED BY HIO
NO
YES
TP01A
A2 TIME OUT OR INT. REQ.
YES
NO
F1
TP11G
B8 H4
TP11E
B1 TURN OFF IOB EXCEPTION AND HALT I/O FLAGS
B7 B3
HIOREAD
B3 TURN OFF HALT I/O FLAG
TP07
B7 A4
A4 DOES ID COMPARE
YES
NO
C2
REXITA
B6 E5
TPXX
C1 CE, DE, UE
NO
YES
REXITA
C2 INCREMENT CCW ADDRESS BY 8
C3 UPDATE CCW FIELDS
REXIT
D1 TURN ON EOT FLAG
D2 STORE IT IN IOB START ADDRESS
B7 D4
WRITE
D4 RETURN FROM ERP
YES
NO
D5 SET NORMAL COMPLETION CODE X'7F' IN IOB
E1 RESET DECRESPN BYTE
E2 TURN OFF IOB EXCEPTION FLAG
E3 MOVE WRITE BREAK CCW IN THE CHANNEL PROGRAM AREA
WRITE 1
E4 NORMAL COMPLETION
NO
YES
B6 E5
TPXX
E5 RESET TEMPORARY FLAG
B6 E5
TPXX
B7 F1
TP11G
F1 LINE ERROR COUNT SPECIFIED
NO
YES
B6 E5
TPXX
F2 RESTORE REGISTERS
TP11I
F3 SAVE SENSE AND CSW AND FAILING CCW
YES
F4 UE
NO
G1 HALT I/O
NO
YES
G2 RETURN (IOS + 8) RESTART
G3 DATA CHECK
YES
NO
B8 H4
TP11E
G4 UC
YES
NO
B6 E5
TPXX
H1 DECREMENT TRANSMISSION COUNT BY 1
B6 E5
TPXX
H2 UPDATE DATA CHECK COUNTER
J2 IS THE THRESHOLD REACHED
YES
NO
B6 E5
TPXX
J3 TURN ON IOB EXCEPTION FLAG
J4 DECREMENT DATA CHECK COUNTER BY 1
J5 SET ON TEMPORARY FLAG
B6 E5
TPXX

Chart B8. WTTA Channel End Appendage (IGG019PD) (Part 3 of 3)

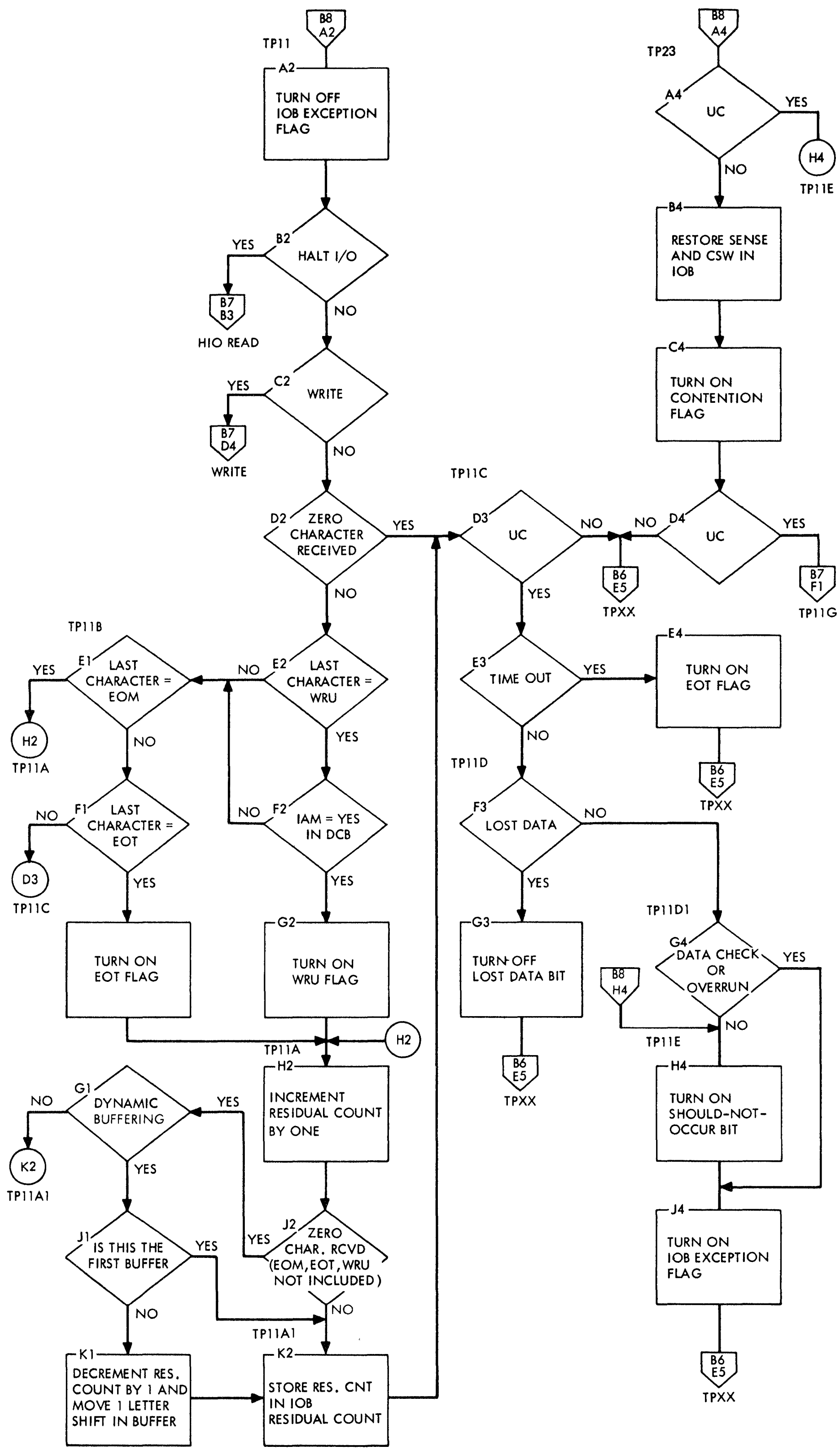


Chart B9. Local 3270 Channel End/Abnormal End Appendage (IGG019PA) (Part 1 of 2)

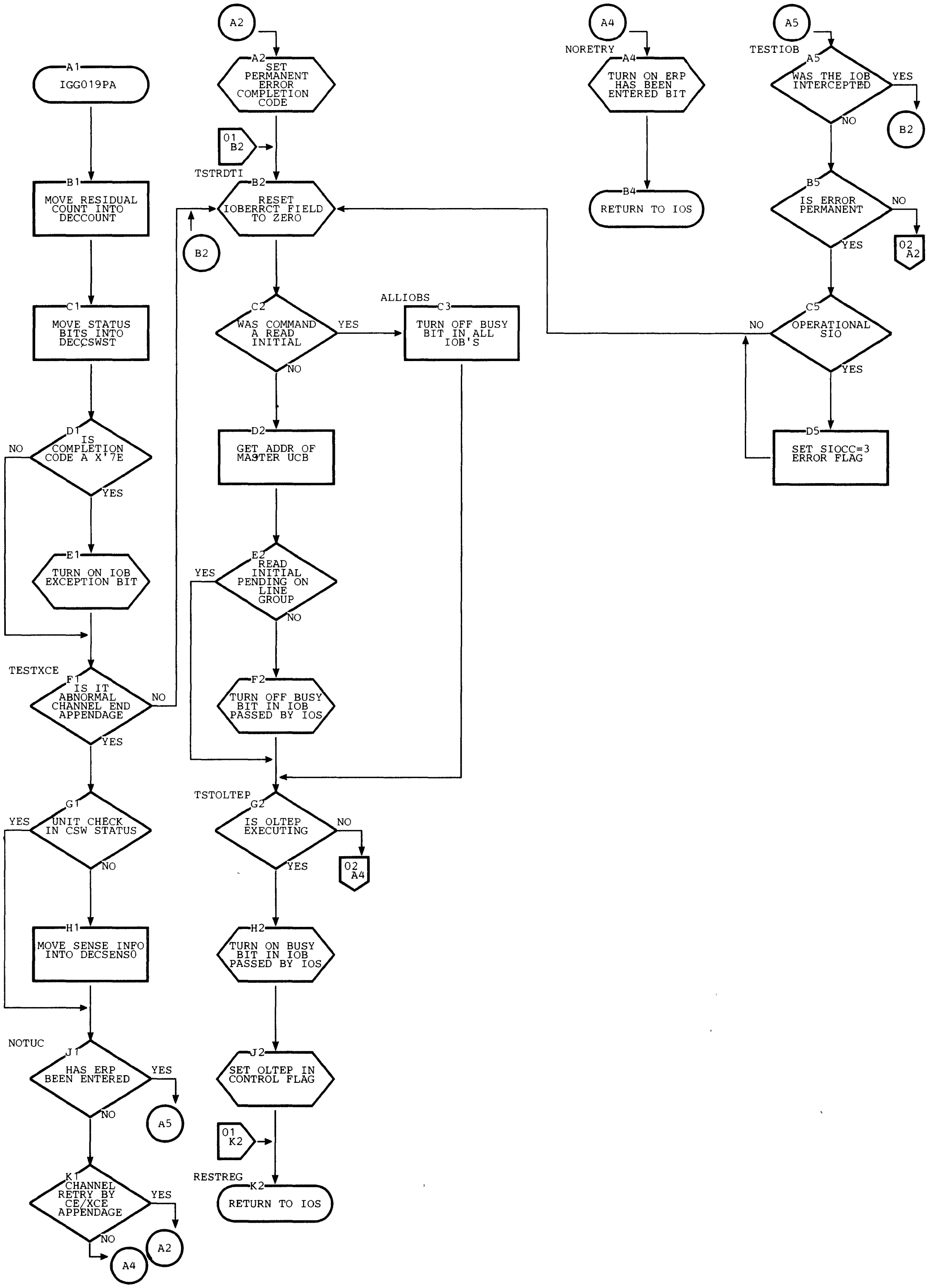


Chart B10. Local 3270 Channel End/Abnormal End Appendage (IGG019PA) (Part 2 of 2)

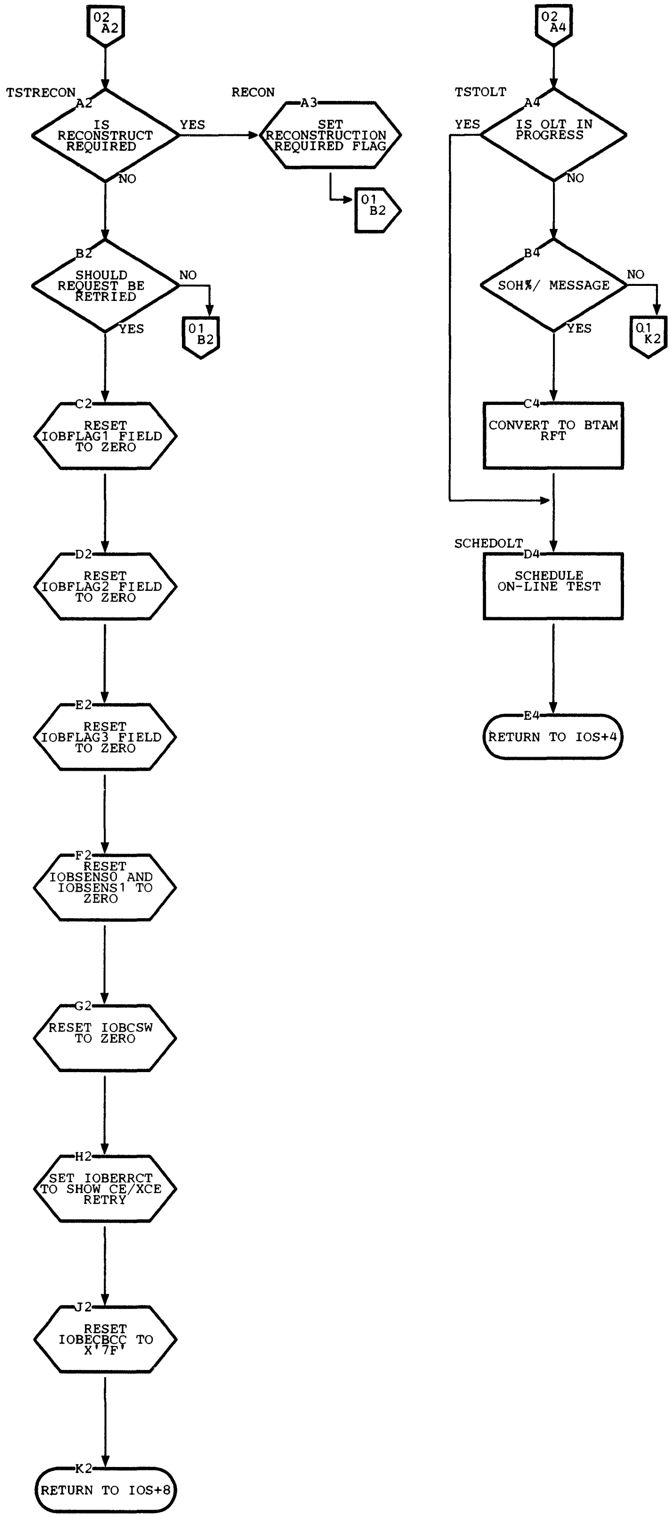


Chart C1. Program Control Interrupt Appendage (IGG019MC)

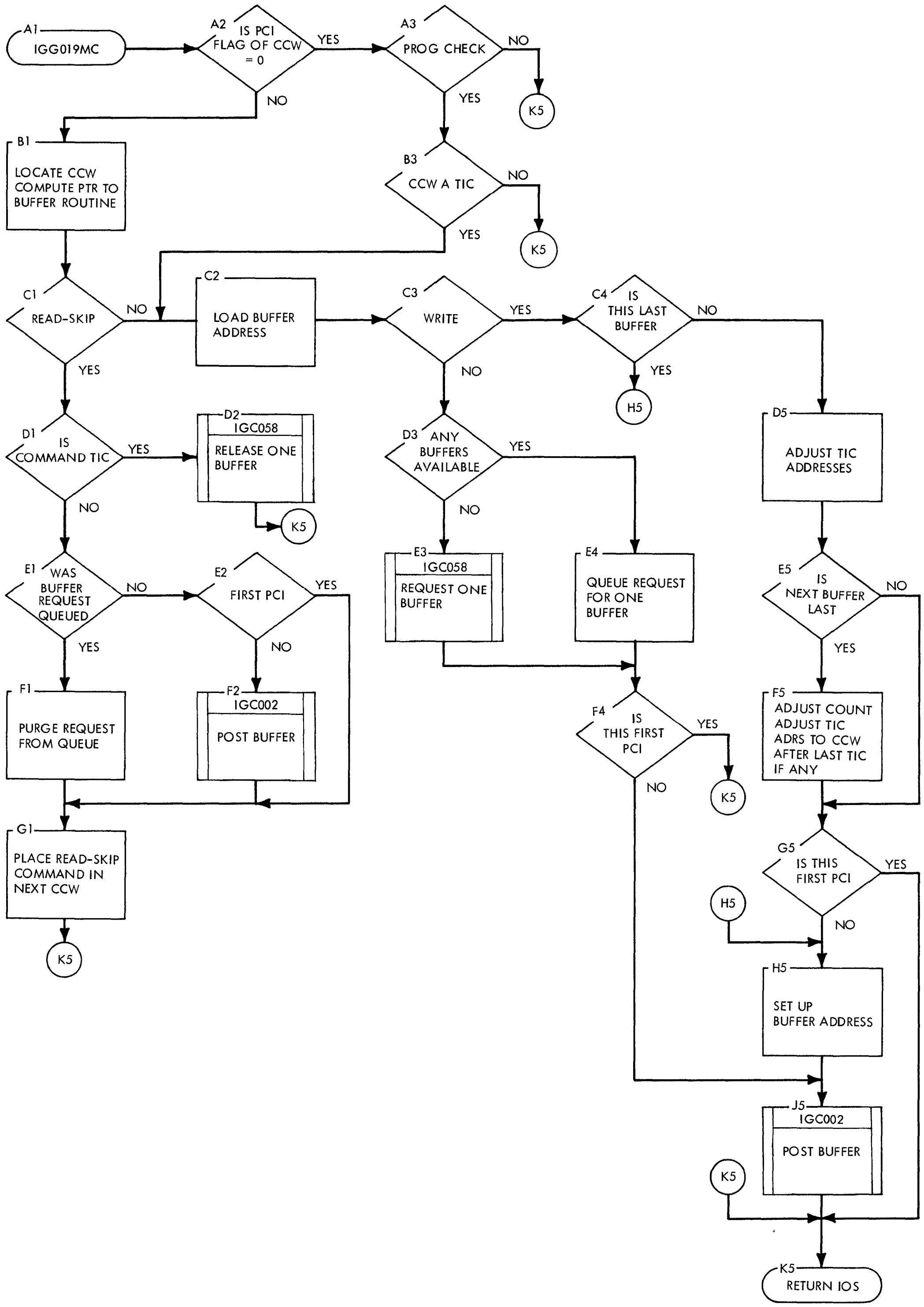


Chart C2. Local 3270 First-Level Attention Routine (IECTATEN)

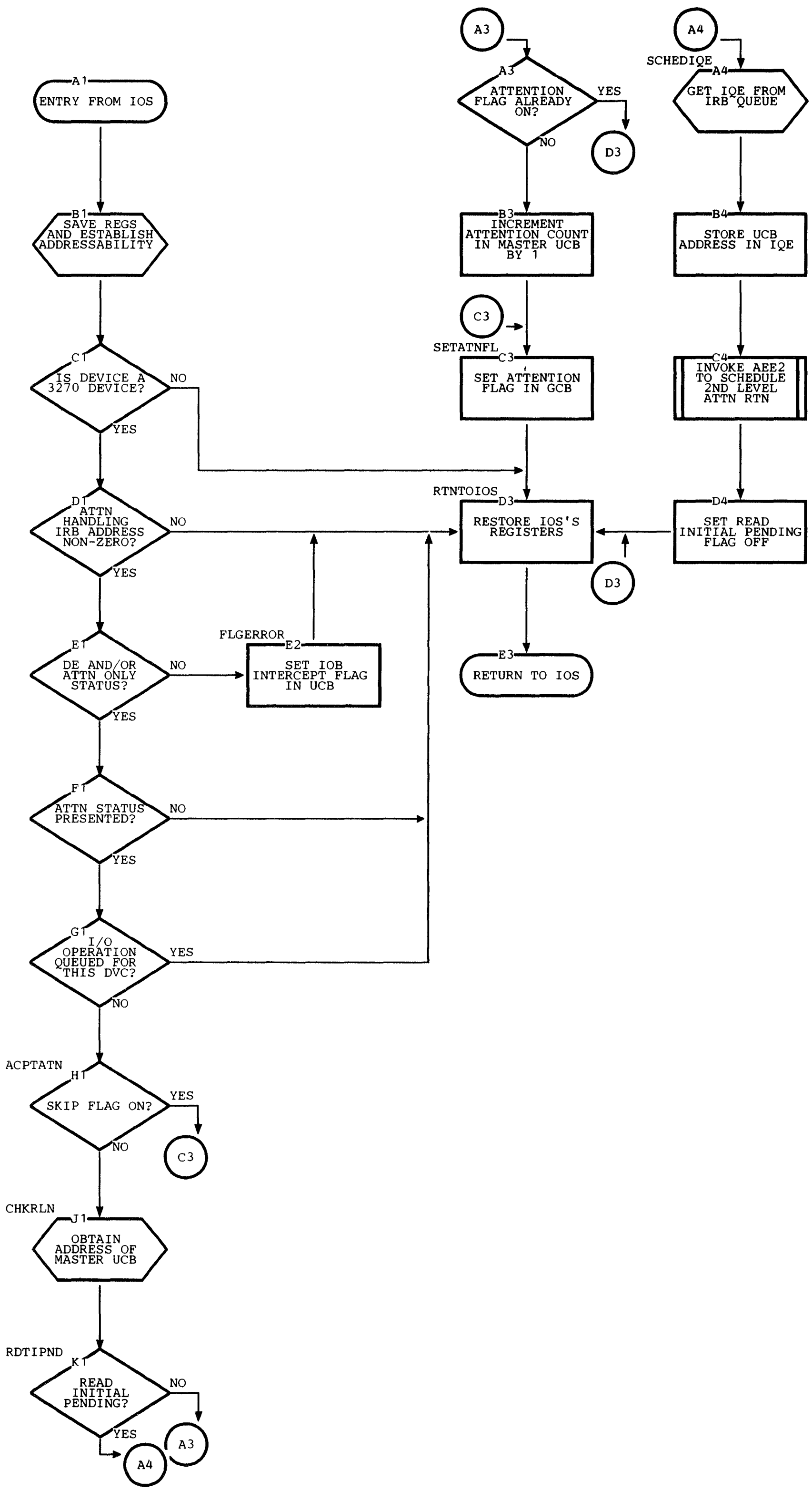


Chart C3. Local 3270 Second-Level Attention Routine (IGG019PG)

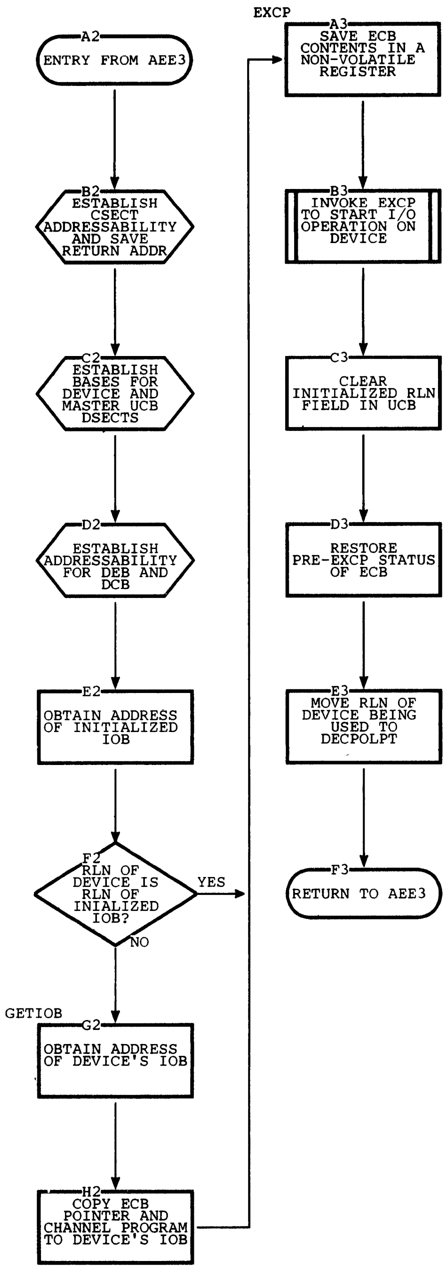
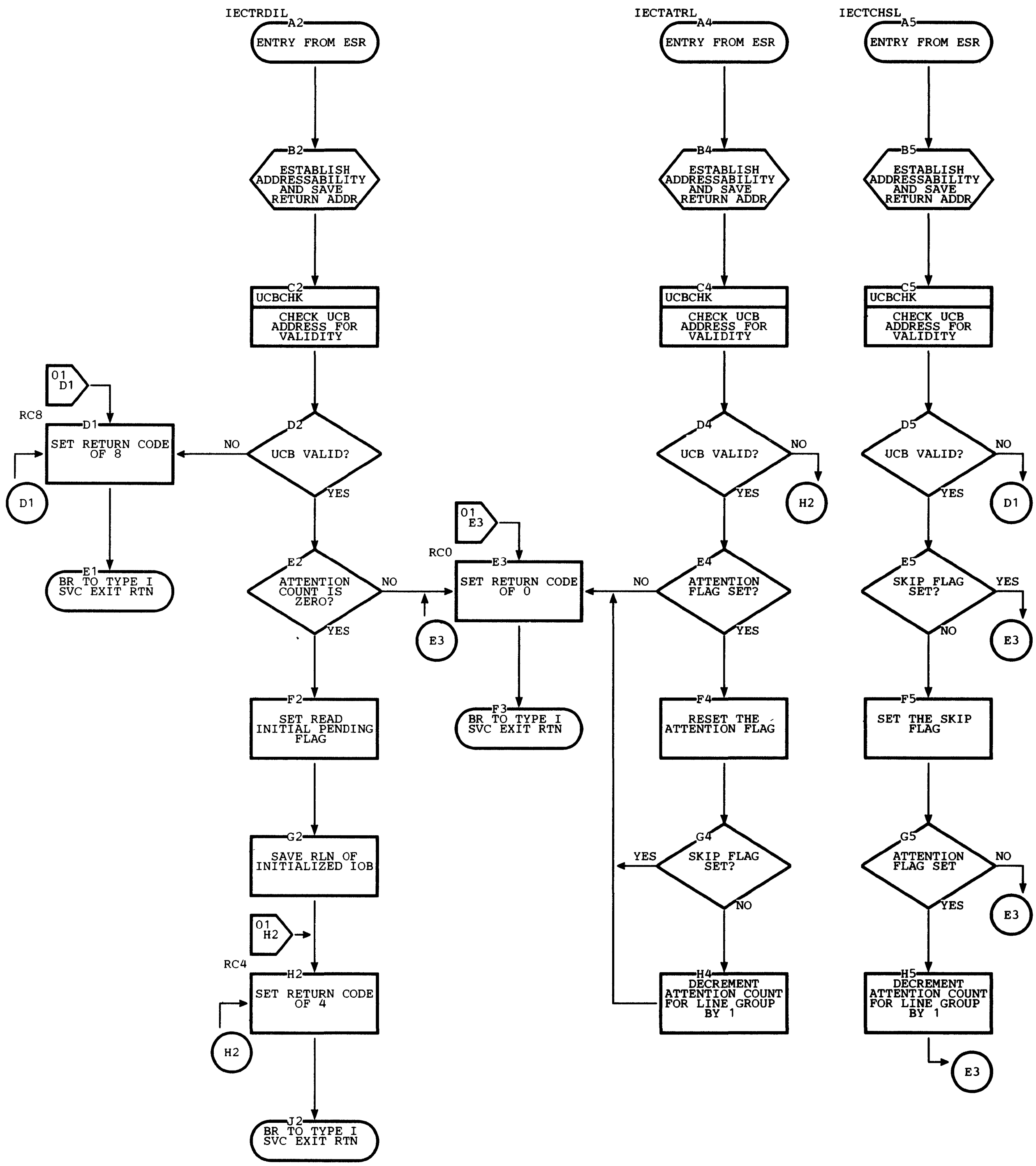


Chart C4. Local 3270 BTAM SVC Routine (IECTSVC) (Part 1 of 3)



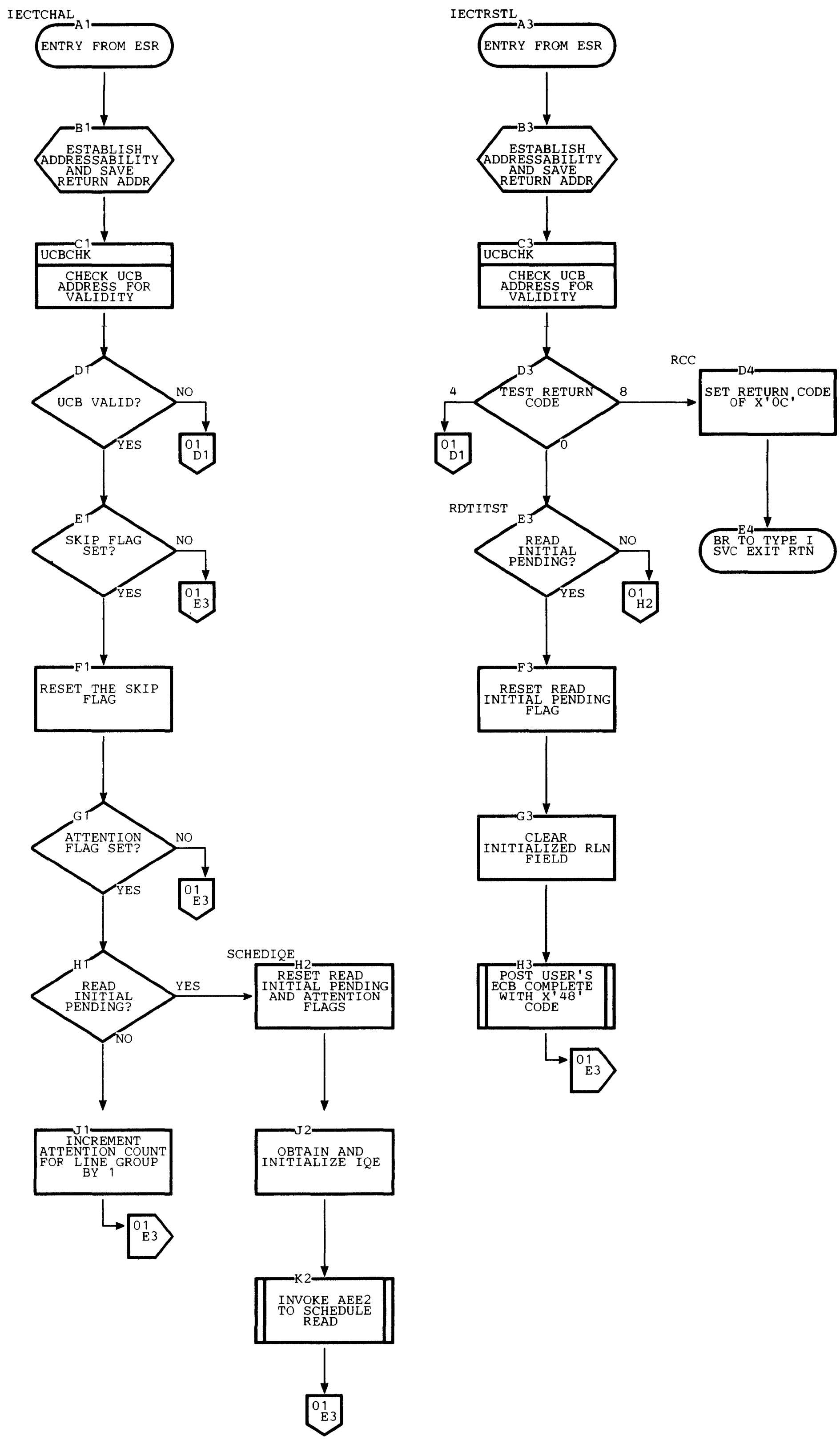
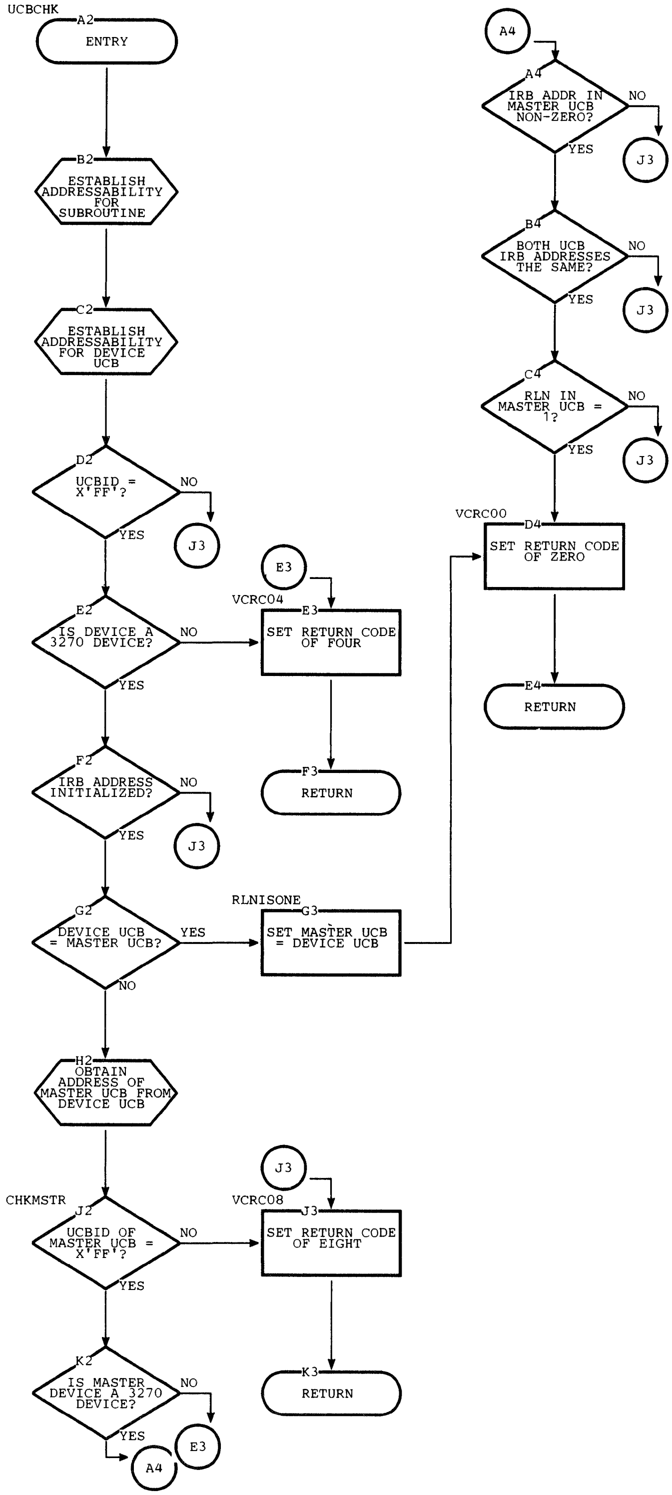

IECTCHAL
A1
ENTRY FROM ESR
B1
ESTABLISH ADDRESSABILITY AND SAVE RETURN ADDR
C1
UCBCHK
CHECK UCB ADDRESS FOR VALIDITY
D1
UCB VALID?
NO
01 D1
YES
E1
SKIP FLAG SET?
NO
01 E3
YES
F1
RESET THE SKIP FLAG
G1
ATTENTION FLAG SET?
NO
01 E3
YES
H1
READ INITIAL PENDING?
YES
NO
J1
INCREMENT ATTENTION COUNT FOR LINE GROUP BY 1
01 E3
SCHEDIQE
H2
RESET READ INITIAL PENDING AND ATTENTION FLAGS
J2
OBTAIN AND INITIALIZE IQE
K2
INVOKE AEE2 TO SCHEDULE READ
01 E3
IECTRSTL
A3
ENTRY FROM ESR
B3
ESTABLISH ADDRESSABILITY AND SAVE RETURN ADDR
C3
UCBCHK
CHECK UCB ADDRESS FOR VALIDITY
D3
TEST RETURN CODE
4
01 D1
8
0
RCC
D4
SET RETURN CODE OF X'0C'
E4
BR TO TYPE I SVC EXIT RTN
RDTITST
E3
READ INITIAL PENDING?
NO
01 H2
YES
F3
RESET READ INITIAL PENDING FLAG
G3
CLEAR INITIALIZED RLN FIELD
H3
POST USER'S ECB COMPLETE WITH X'48' CODE
01 E3

Chart C6. Local 3270 BTAM SVC Routine (IECTSVC) (Part 3 of 3)



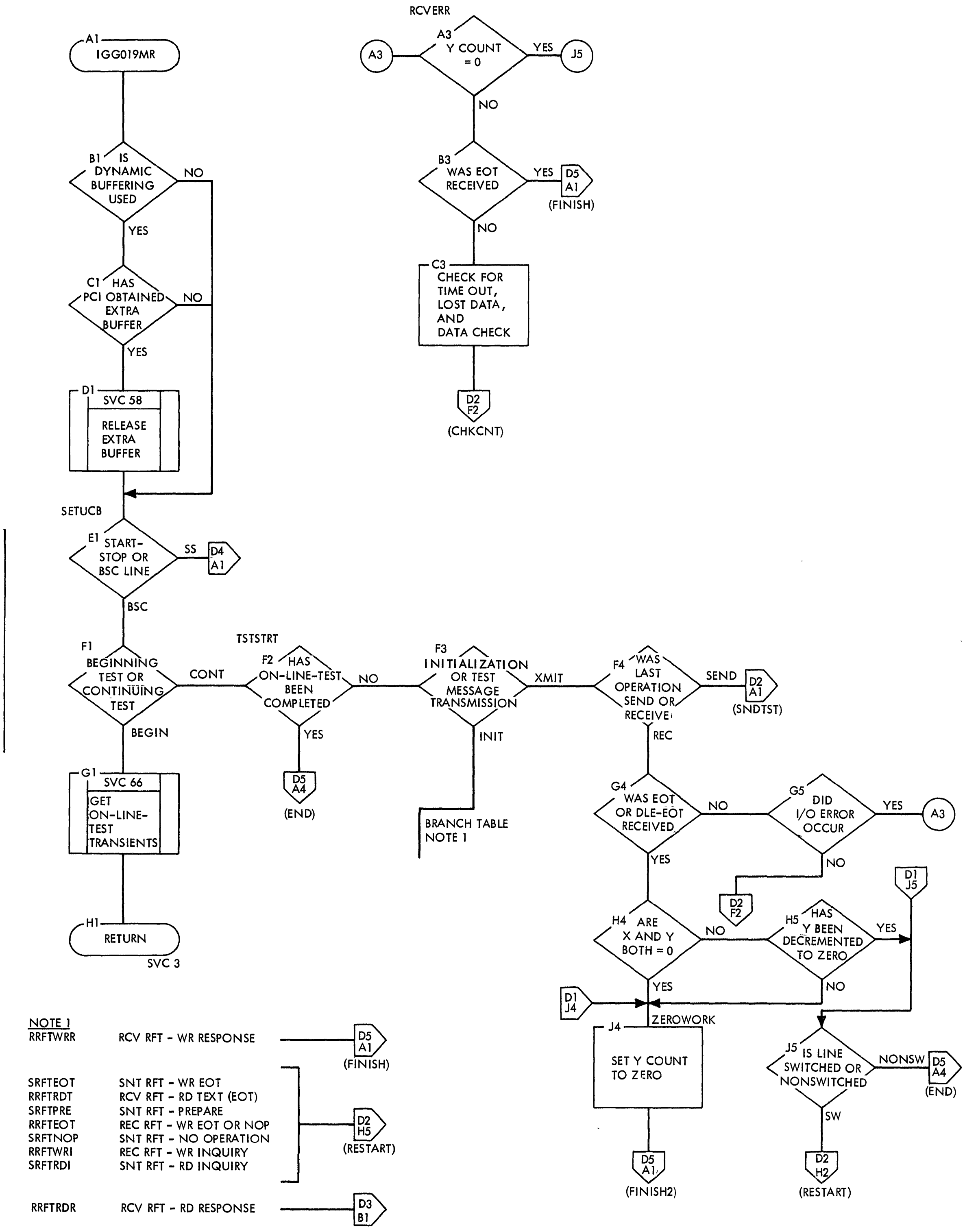

A1
IGG019MR
B1 IS DYNAMIC BUFFERING USED
NO
YES
C1 HAS PCI OBTAINED EXTRA BUFFER
NO
YES
D1
SVC 58
RELEASE EXTRA BUFFER
SETUCB
E1 START-STOP OR BSC LINE
SS
D4 A1
BSC
F1 BEGINNING TEST OR CONTINUING TEST
CONT
BEGIN
G1
SVC 66
GET ON-LINE-TEST TRANSIENTS
H1
RETURN
SVC 3
RCVERR
A3
A3 Y COUNT = 0
YES
J5
NO
B3 WAS EOT RECEIVED
YES
D5 A1
(FINISH)
NO
C3
CHECK FOR TIME OUT, LOST DATA, AND DATA CHECK
D2 F2
(CHKCNT)
TSTSTRT
F2 HAS ON-LINE-TEST BEEN COMPLETED
NO
YES
D5 A4
(END)
F3 INITIALIZATION OR TEST MESSAGE TRANSMISSION
XMIT
INIT
BRANCH TABLE NOTE 1
F4 WAS LAST OPERATION SEND OR RECEIVE
SEND
D2 A1
(SNDTST)
REC
G4 WAS EOT OR DLE-EOT RECEIVED
NO
G5 DID I/O ERROR OCCUR
YES
A3
NO
D2 F2
YES
H4 ARE X AND Y BOTH = 0
NO
H5 HAS Y BEEN DECREMENTED TO ZERO
YES
D1 J5
NO
YES
D1 J4
ZEROWORK
J4
SET Y COUNT TO ZERO
D5 A1
(FINISH2)
J5 IS LINE SWITCHED OR NONSWITCHED
NONSW
D5 A4
(END)
SW
D2 H2
(RESTART)
NOTE 1
RRFTWRR RCV RFT - WR RESPONSE
D5 A1
(FINISH)
SRFTEOT SNT RFT - WR EOT
RRFTRDT RCV RFT - RD TEXT (EOT)
SRFTPRE SNT RFT - PREPARE
RRFTEOT REC RFT - WR EOT OR NOP
SRFTNOP SNT RFT - NO OPERATION
RRFTWRI REC RFT - WR INQUIRY
SRFTRDI SNT RFT - RD INQUIRY
D2 H5
(RESTART)
RRFTRDR RCV RFT - RD RESPONSE
D3 B1

Chart D2. On-Line Test Control (IGG019MR) (Part 2 of 5)

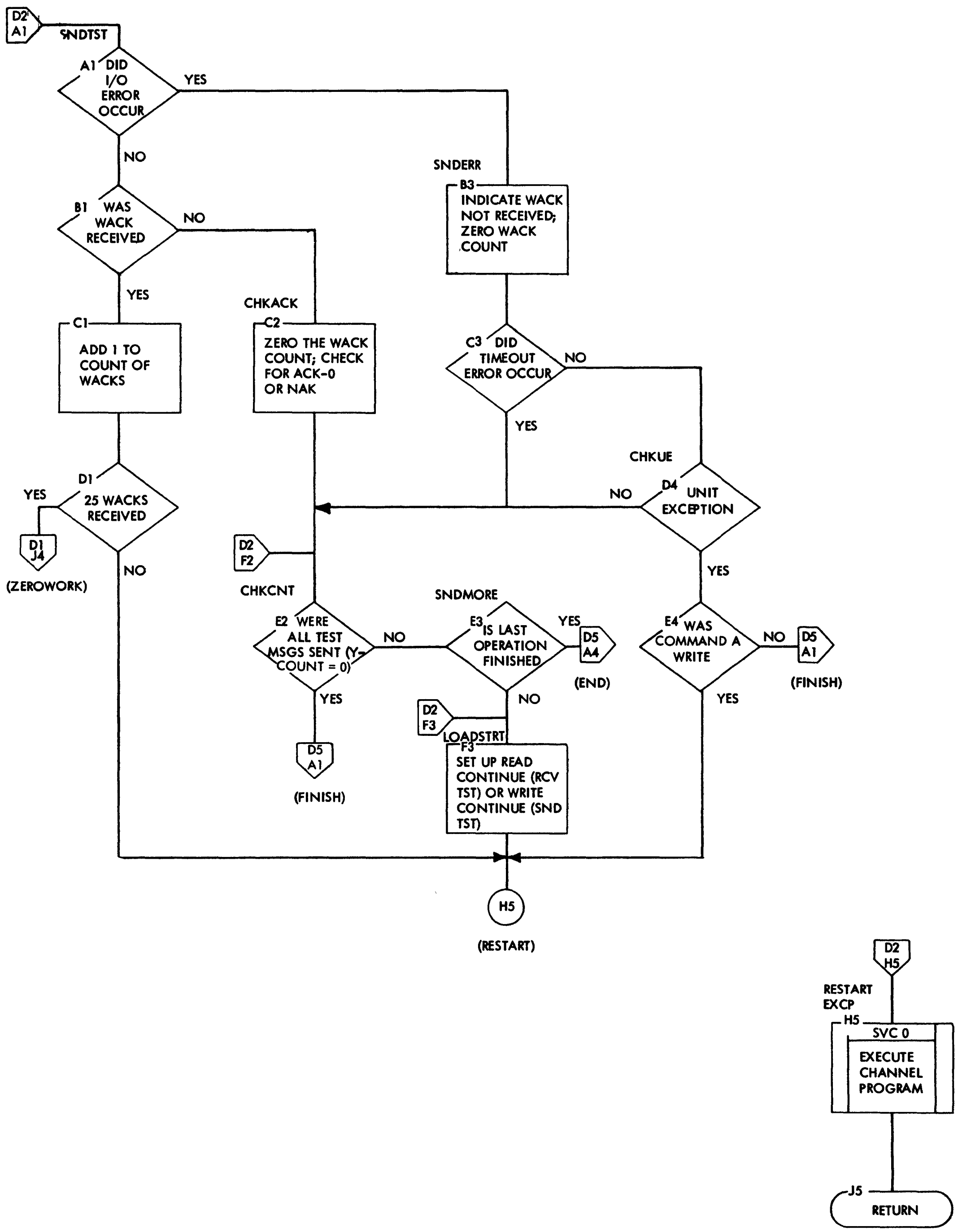


Chart D3. On-Line Test Control (IGG019MR) (Part 3 of 5)

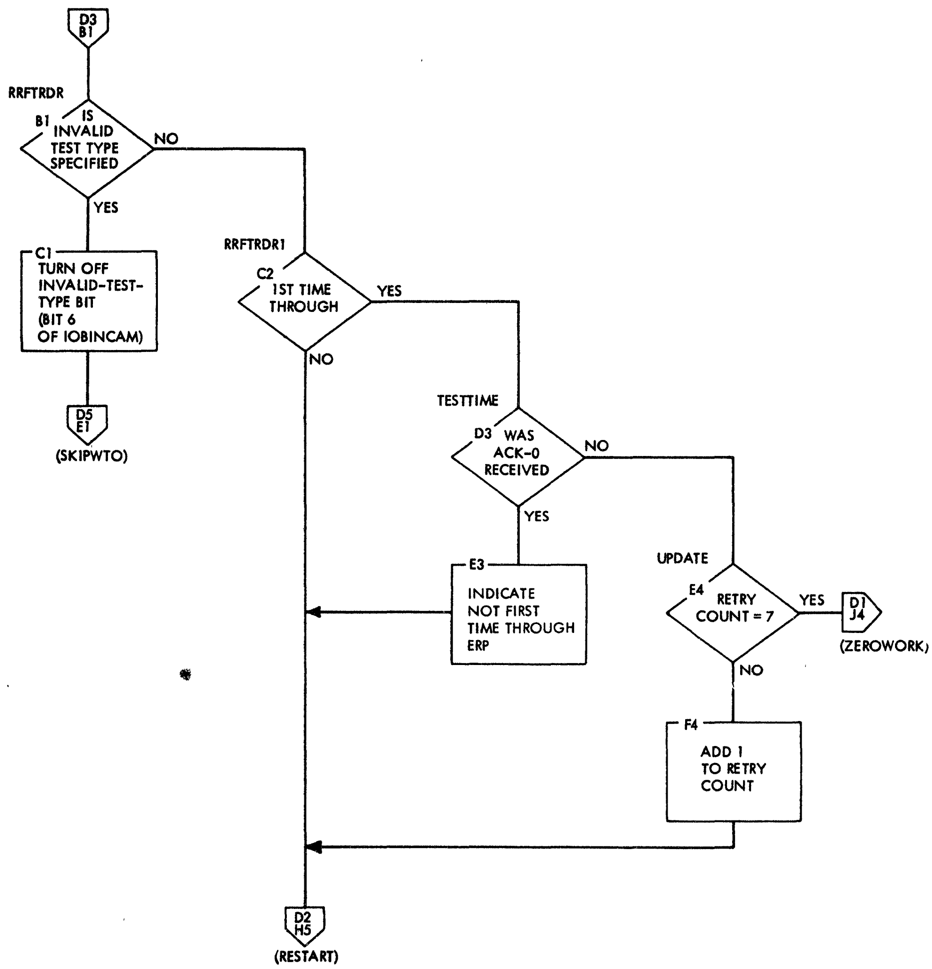


Chart D4. On-Line Test Control (IGG019MR) (Part 4 of 5)

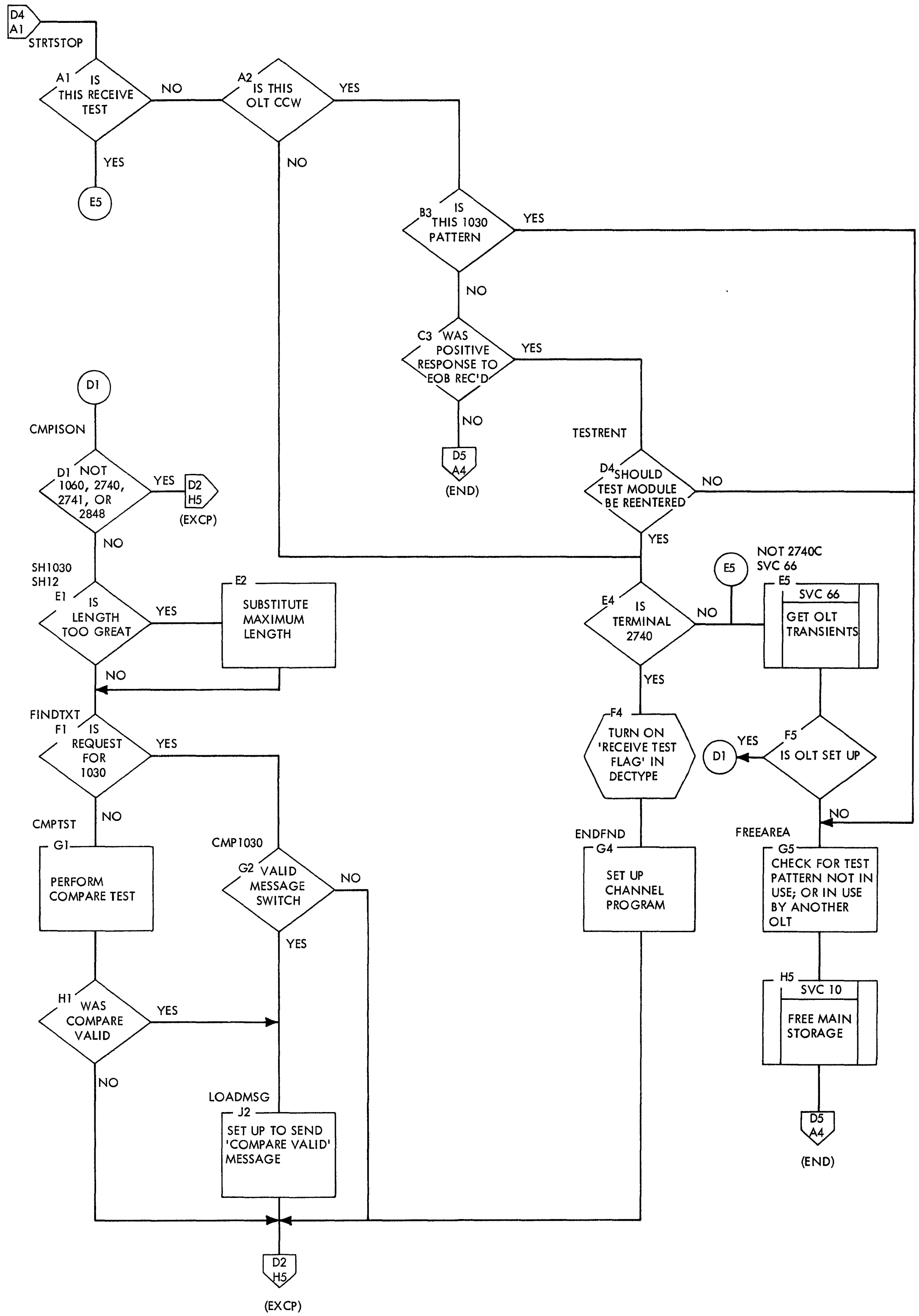
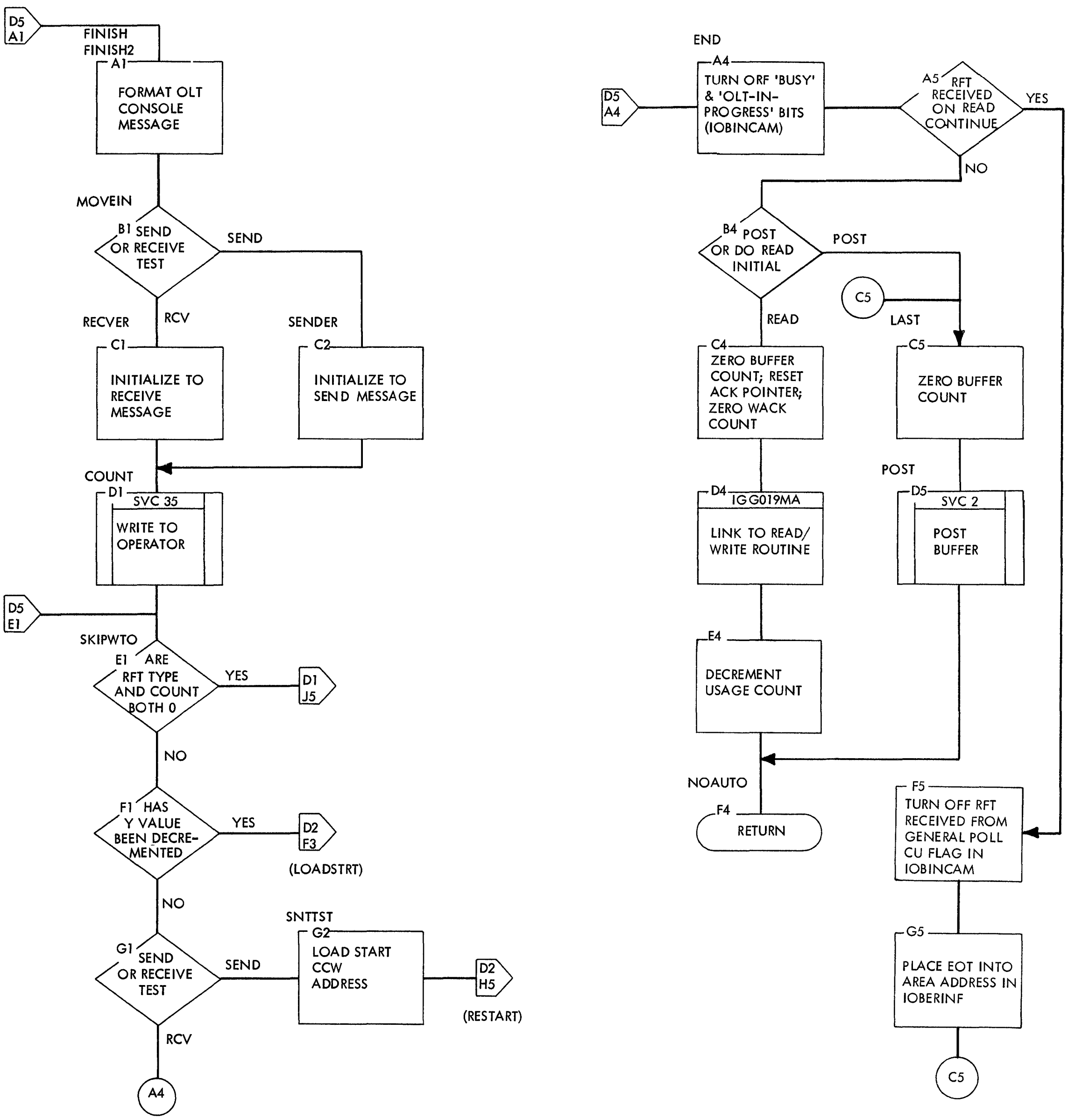


Chart D5. On-Line Test Control (IGG019MR) (Part 5 of 5)



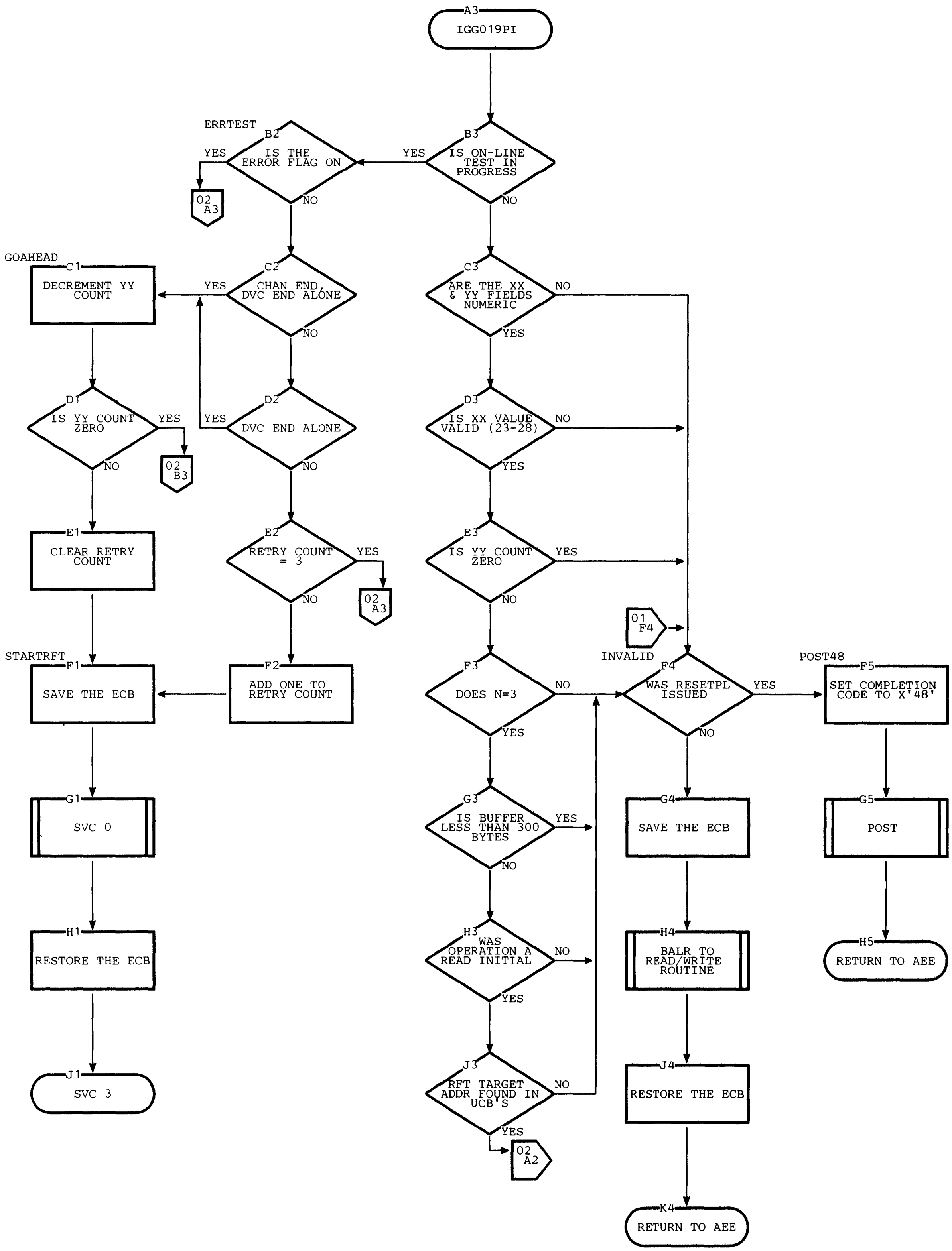

IGG019PI
ERRTEST
IS THE ERROR FLAG ON
IS ON-LINE TEST IN PROGRESS
GOAHEAD
DECREMENT YY COUNT
CHAN END, DVC END ALONE
ARE THE XX & YY FIELDS NUMERIC
IS YY COUNT ZERO
DVC END ALONE
IS XX VALUE VALID (23-28)
CLEAR RETRY COUNT
RETRY COUNT = 3
IS YY COUNT ZERO
STARTRFT
SAVE THE ECB
ADD ONE TO RETRY COUNT
DOES N=3
INVALID
WAS RESETPL ISSUED
POST48
SET COMPLETION CODE TO X'48'
SVC 0
IS BUFFER LESS THAN 300 BYTES
SAVE THE ECB
POST
RESTORE THE ECB
WAS OPERATION A READ INITIAL
BALR TO READ/WRITE ROUTINE
RETURN TO AEE
SVC 3
RFT TARGET ADDR FOUND IN UCB'S
RESTORE THE ECB
RETURN TO AEE

Chart D5B. Local On-Line Test Control (IGG019PI) (Part 2 of 2)

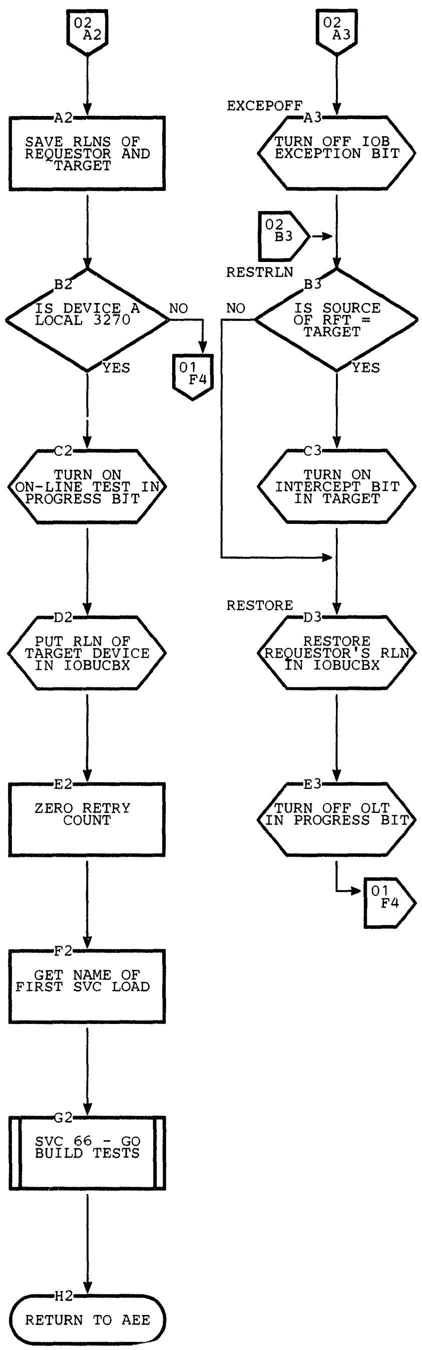


Chart D6. On-Line Test EBCDIC and USASCII/TRANSCODE Message (IGC0706F) (IGC0806F)

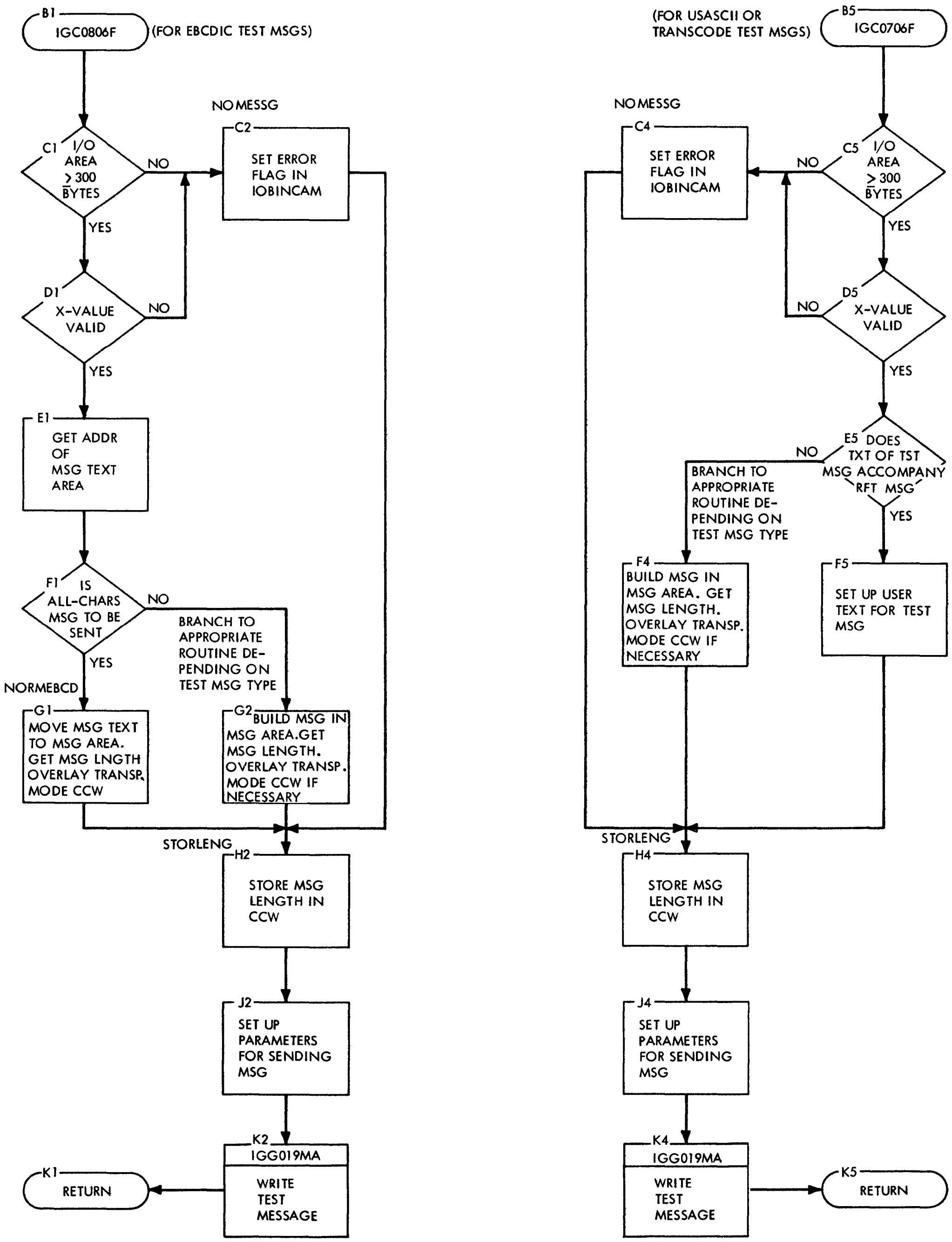


Chart E0. BTAM Start/Stop ERP Control (IGE0004A)

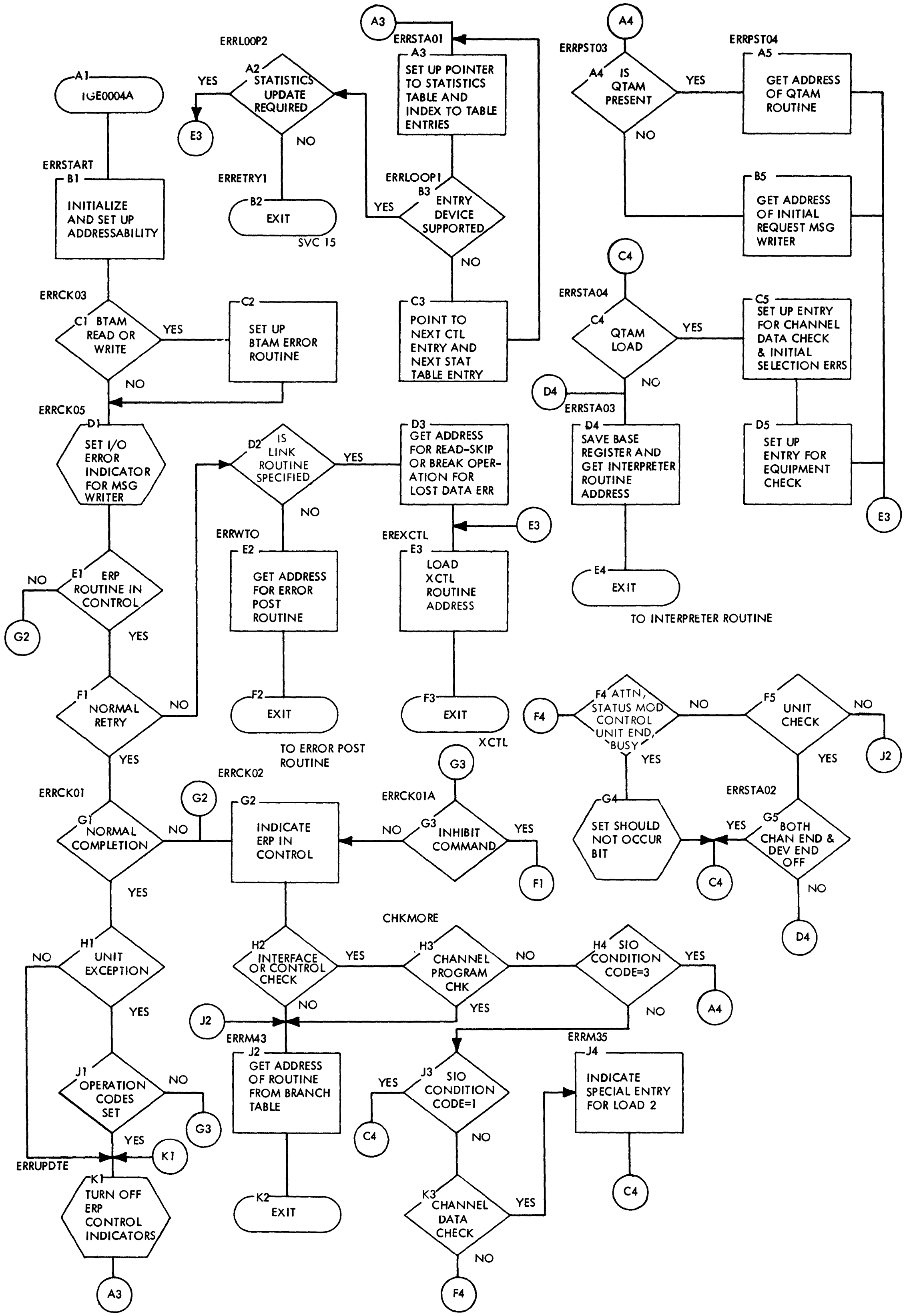


Chart E1. Start/Stop ERP Data Check (IGE0104A)

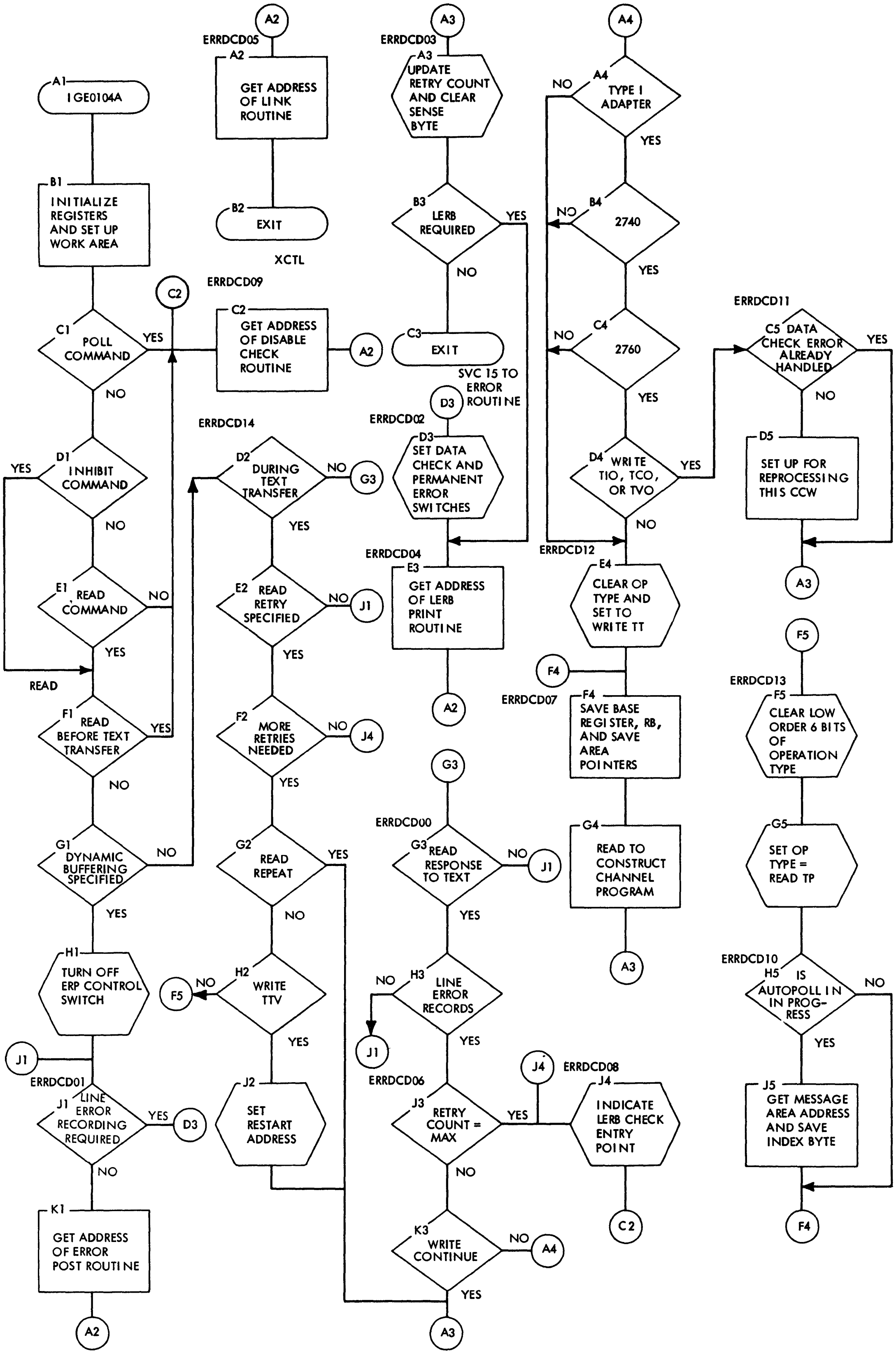


Chart E2. Start/Stop ERP Time Out (IGE0204A) (Part 1 of 2)

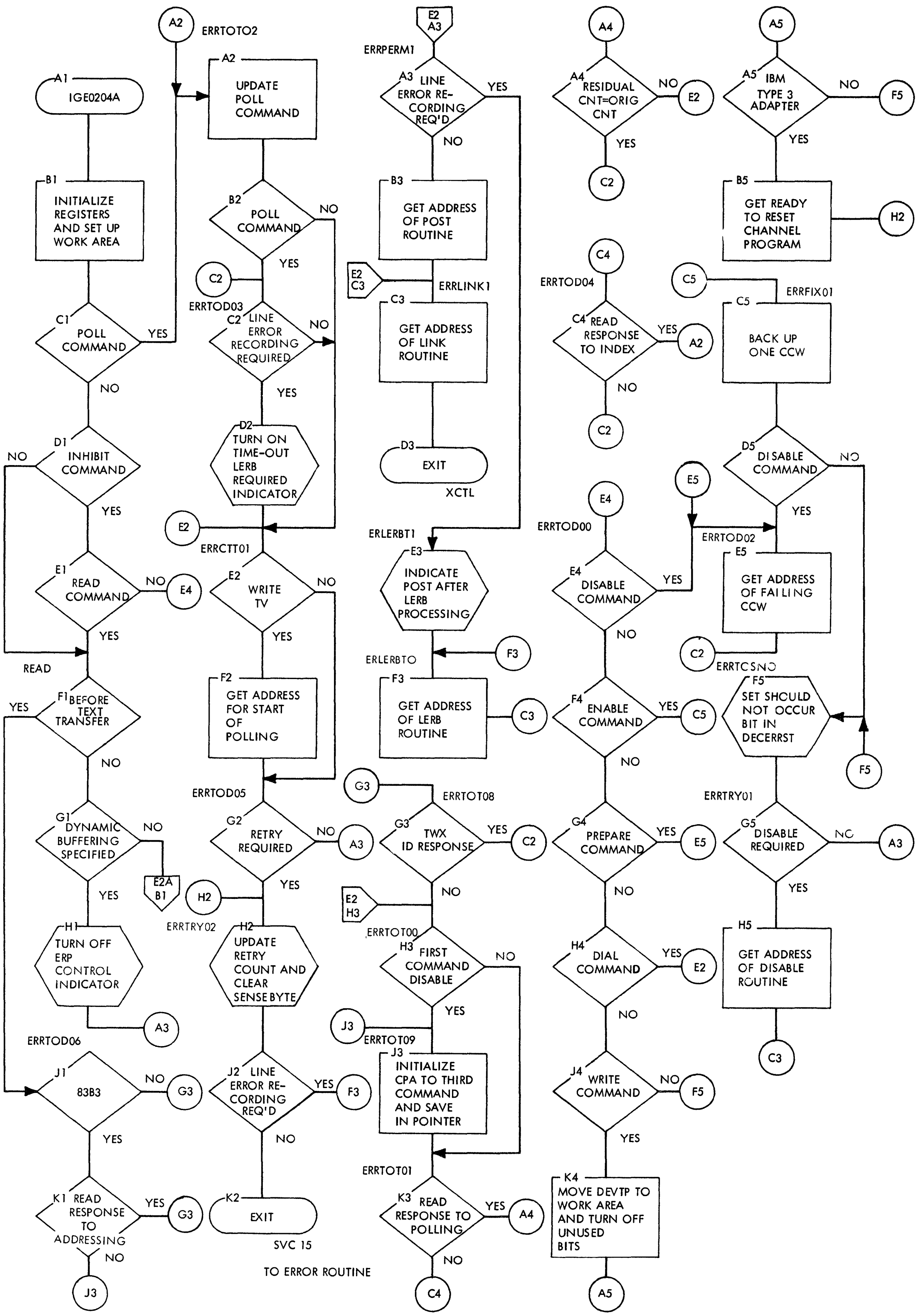


Chart E2A. Start/Stop ERP Time Out (IGE0204A) (Part 2 of 2)

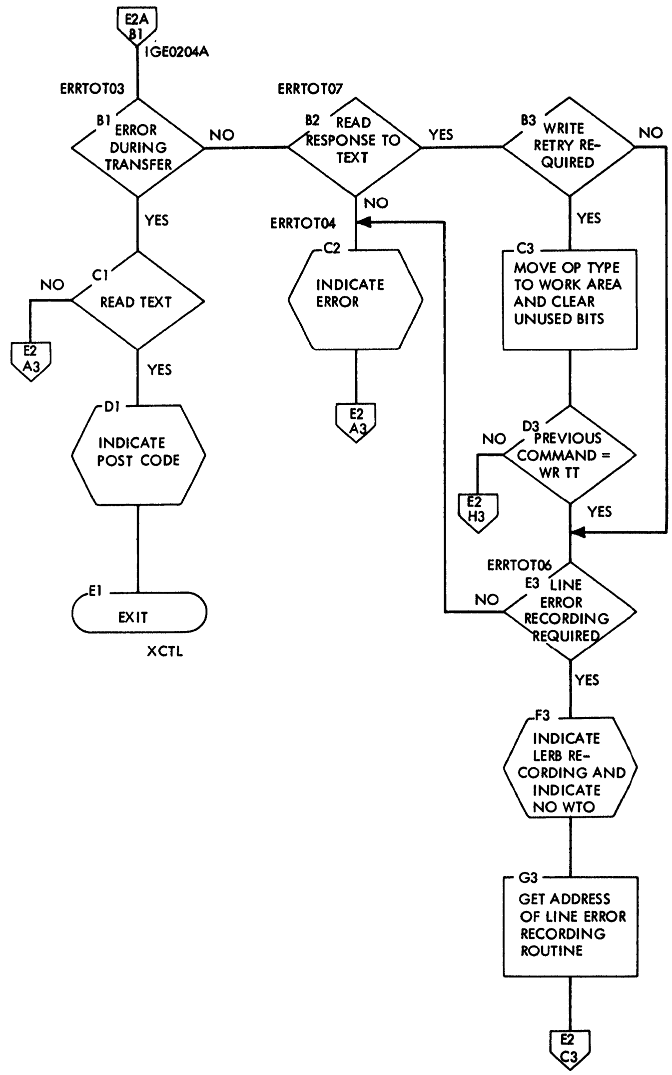


Chart E3. Start/Stop ERP Intervention Required (IGE0304A) (Part 1 of 2)

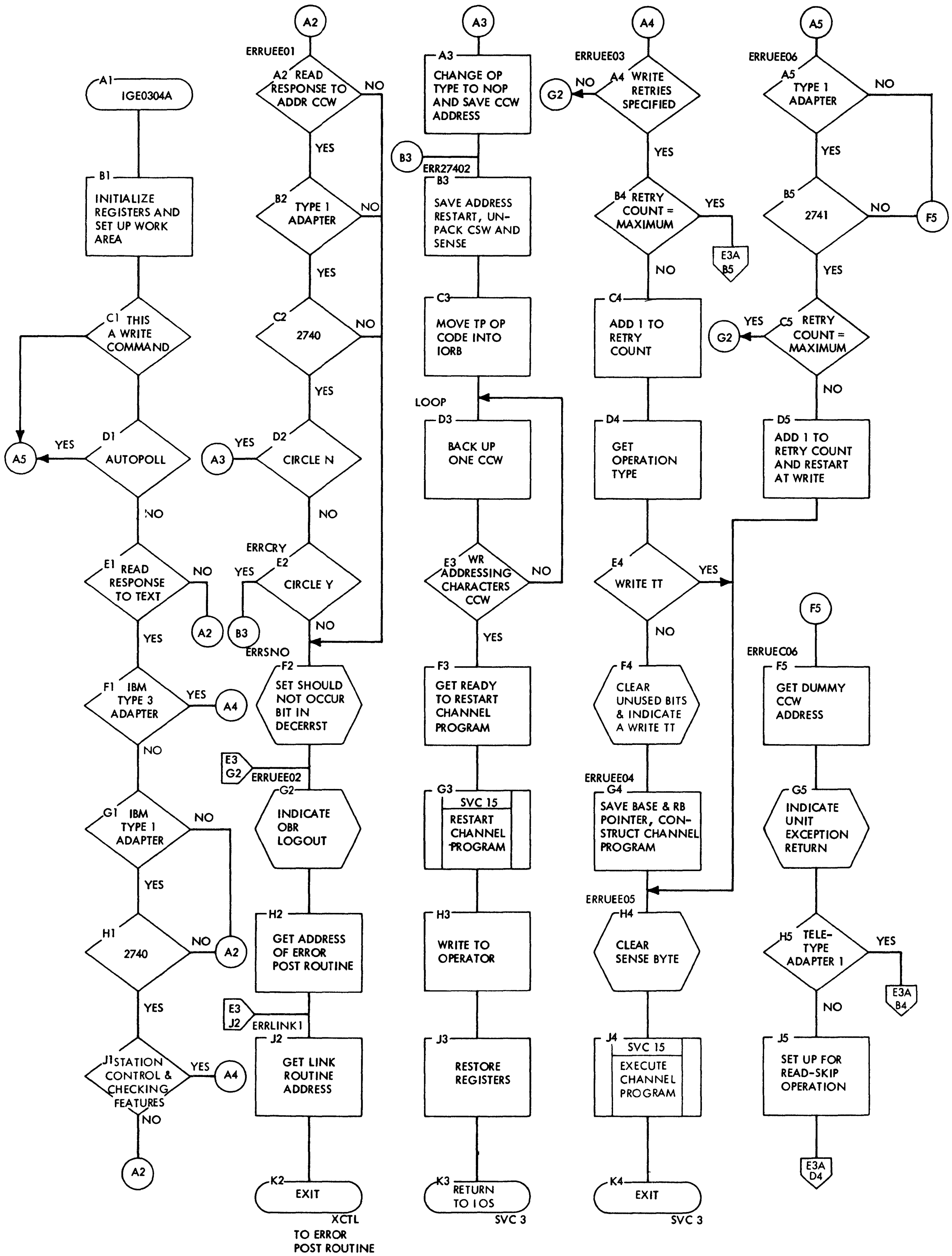


Chart E3A. Start/Stop ERP Intervention Required (IGE0304A) (Part 2 of 2)

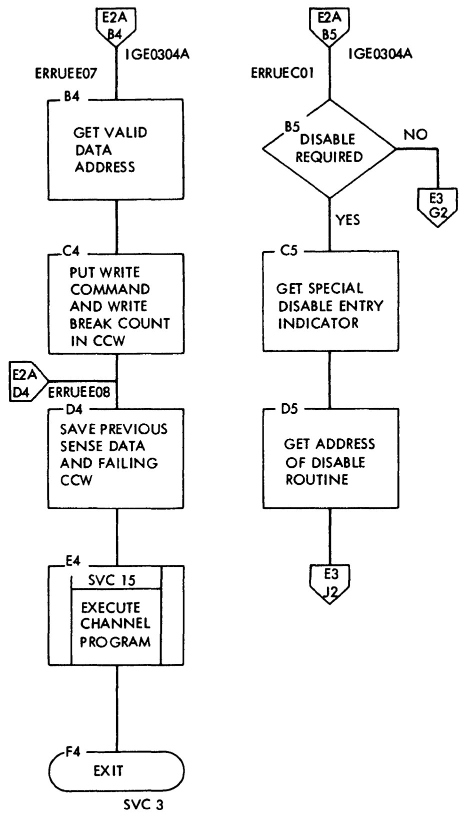


Chart E4. Start/Stop ERP Lost Data (IGE0404A)

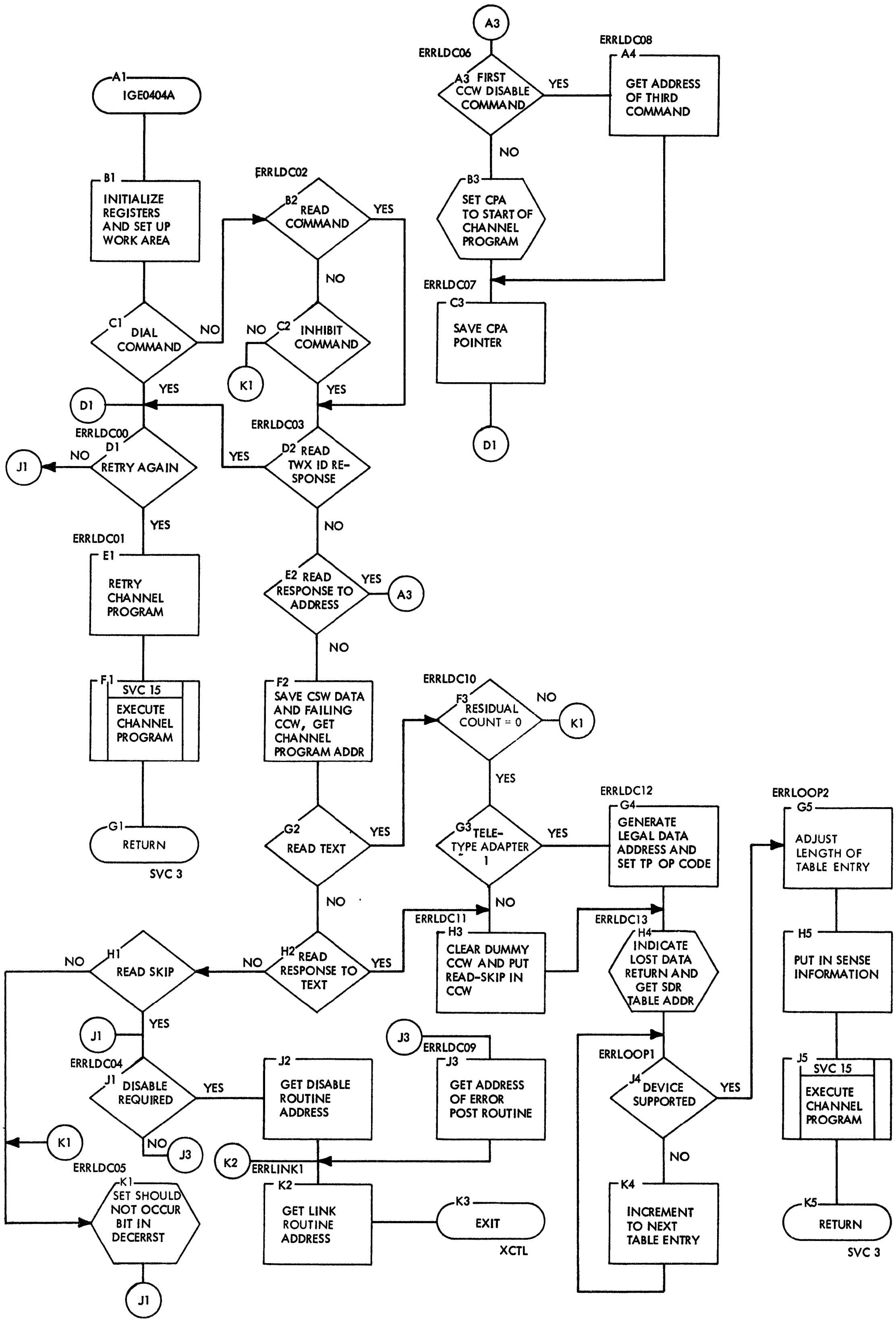
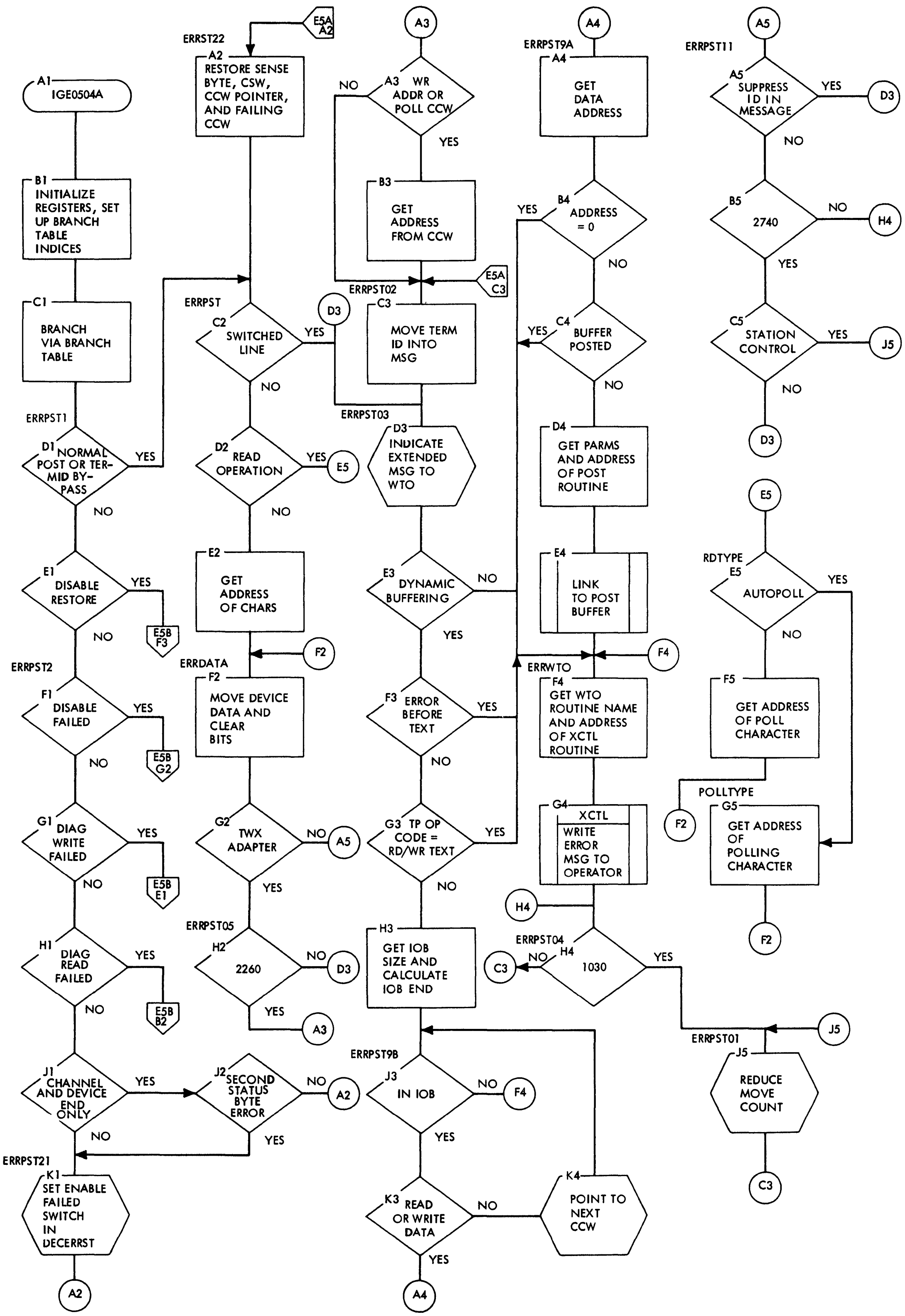


Chart E5A. Start/Stop ERP Post (IGE0504A) (Part 1 of 2)



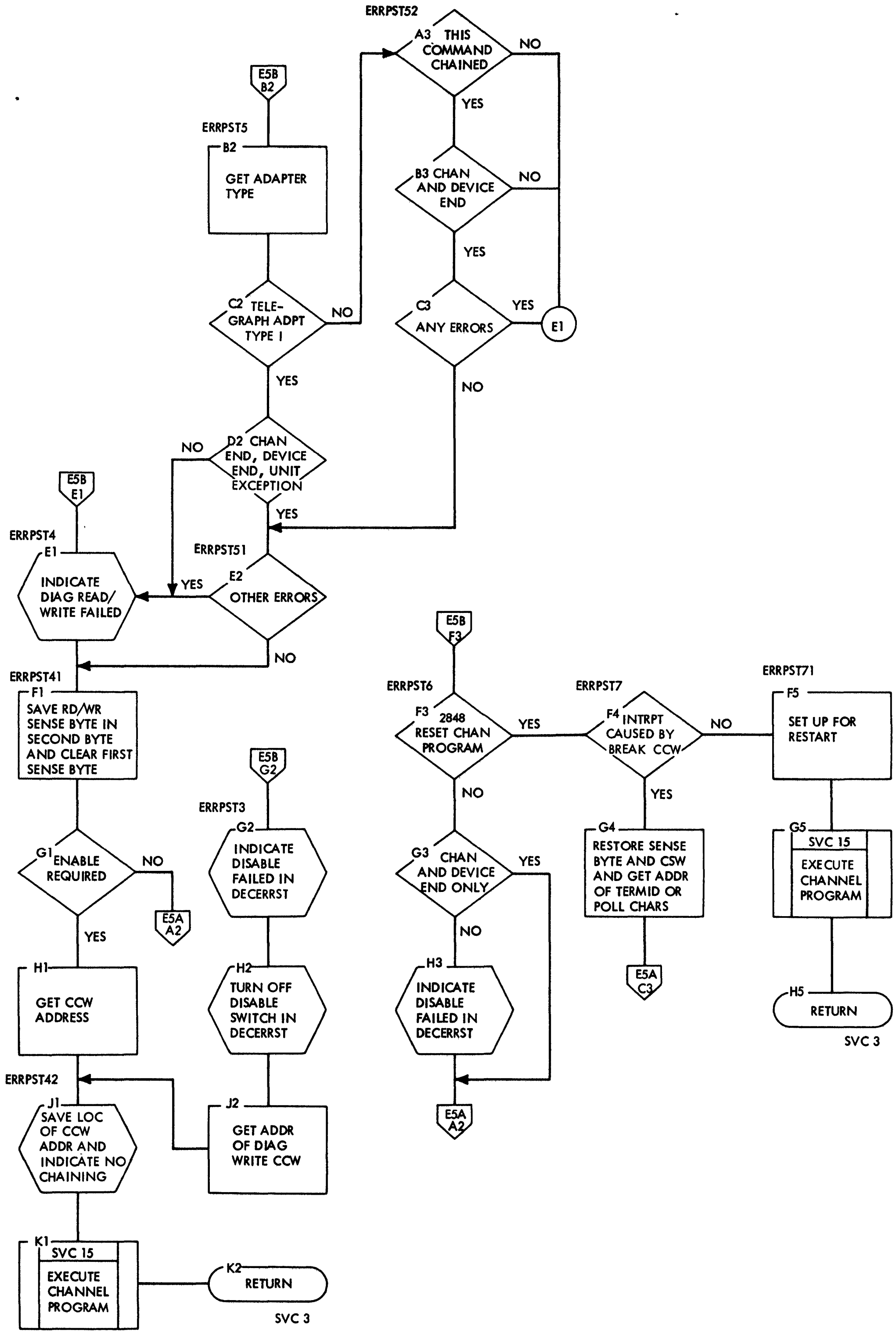

ERRPST52
A3 THIS COMMAND CHAINED
NO
YES
E5B B2
ERRPST5
B2
GET ADAPTER TYPE
B3 CHAN AND DEVICE END
C2 TELE-GRAPH ADPT TYPE I
C3 ANY ERRORS
E1
D2 CHAN END, DEVICE END, UNIT EXCEPTION
E5B E1
ERRPST4
E1
INDICATE DIAG READ/ WRITE FAILED
ERRPST51
E2 OTHER ERRORS
ERRPST41
F1
SAVE RD/WR SENSE BYTE IN SECOND BYTE AND CLEAR FIRST SENSE BYTE
G1 ENABLE REQUIRED
E5A A2
H1
GET CCW ADDRESS
ERRPST42
J1
SAVE LOC OF CCW ADDR AND INDICATE NO CHAINING
K1
SVC 15
EXECUTE CHANNEL PROGRAM
K2
RETURN
SVC 3
E5B G2
ERRPST3
G2
INDICATE DISABLE FAILED IN DECERRST
H2
TURN OFF DISABLE SWITCH IN DECERRST
J2
GET ADDR OF DIAG WRITE CCW
E5B F3
ERRPST6
F3 2848 RESET CHAN PROGRAM
G3 CHAN AND DEVICE END ONLY
H3
INDICATE DISABLE FAILED IN DECERRST
E5A A2
ERRPST7
F4 INTRPT CAUSED BY BREAK CCW
G4
RESTORE SENSE BYTE AND CSW AND GET ADDR OF TERMID OR POLL CHARS
E5A C3
ERRPST71
F5
SET UP FOR RESTART
G5
SVC 15
EXECUTE CHANNEL PROGRAM
H5
RETURN
SVC 3

Chart E6. Start/Stop ERP Bus Out Error Check (IGE0604A)

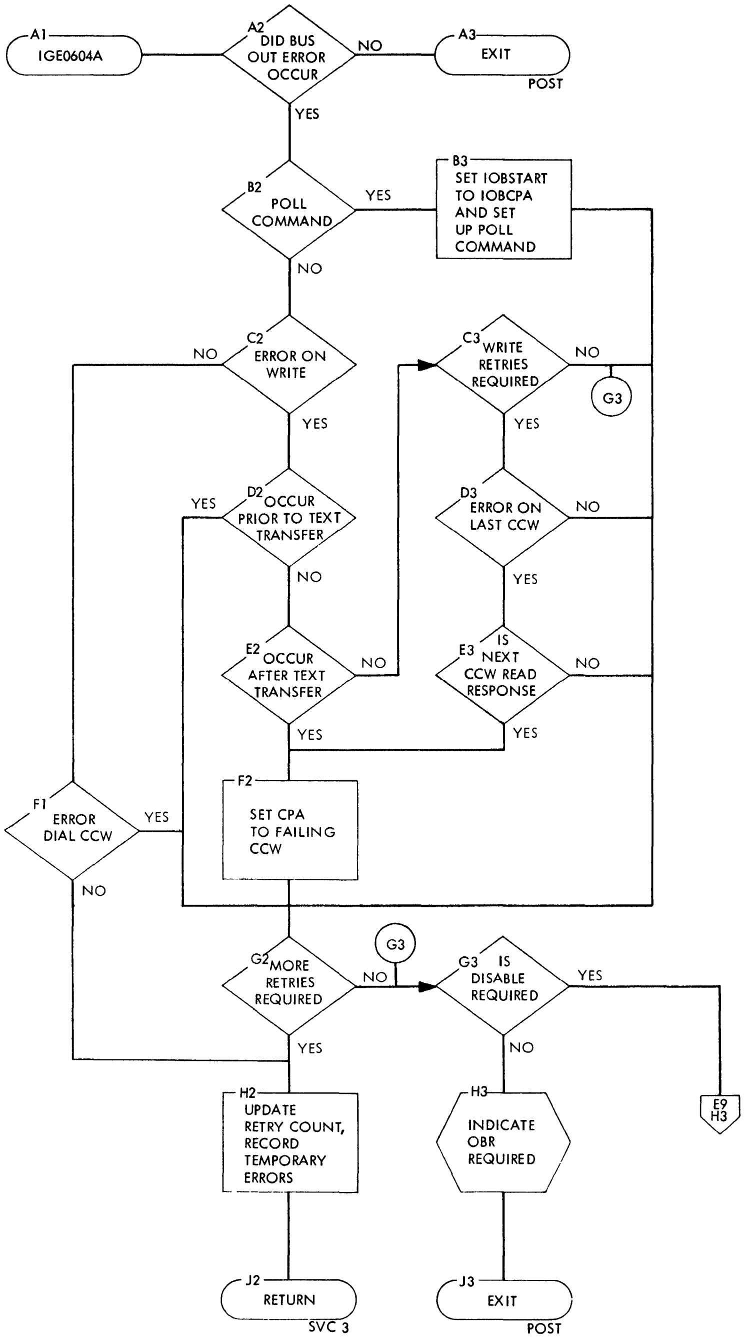


Chart E7. Start/Stop ERP Read Skip Write Break (IGE0704A) (Load 1)

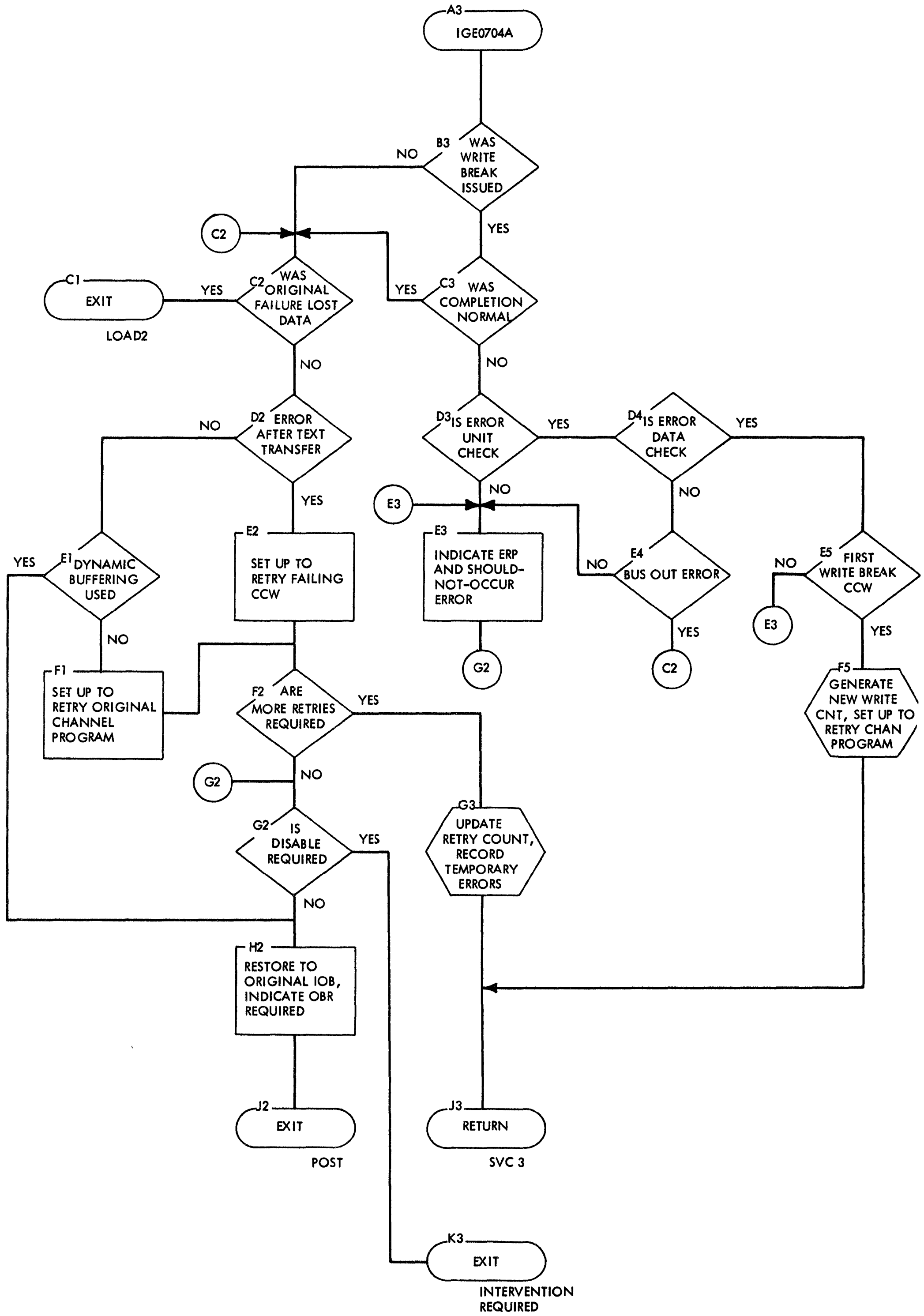


Chart E8. Start/Stop ERP Status Check (IGE0804A)

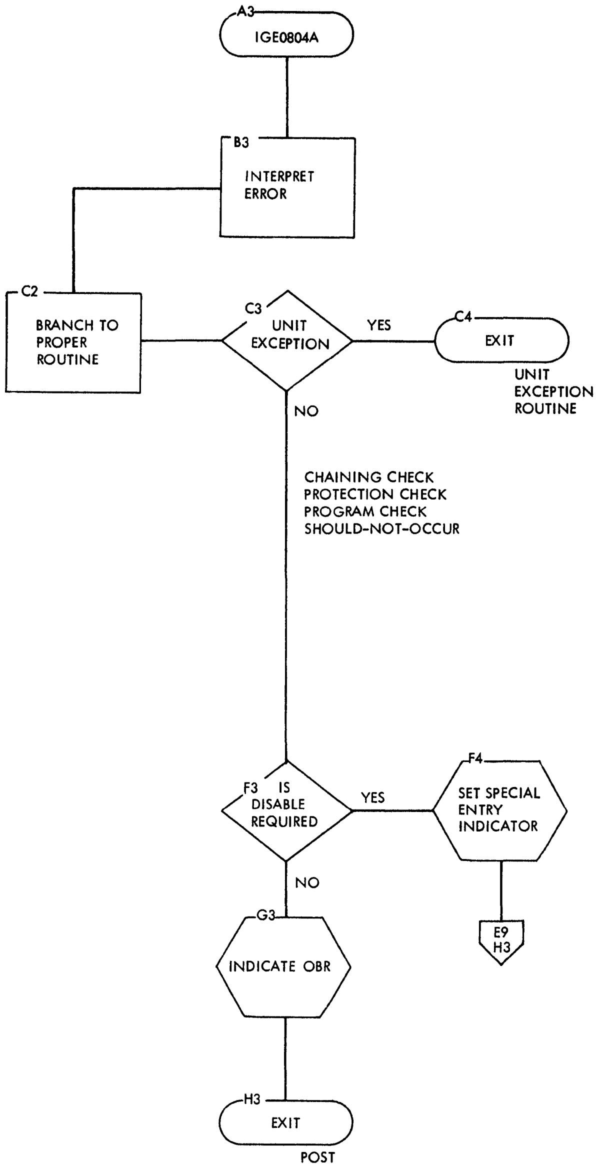


Chart E9. BTAM Start/Stop ERP Control (IGE0904A) (Load 2)

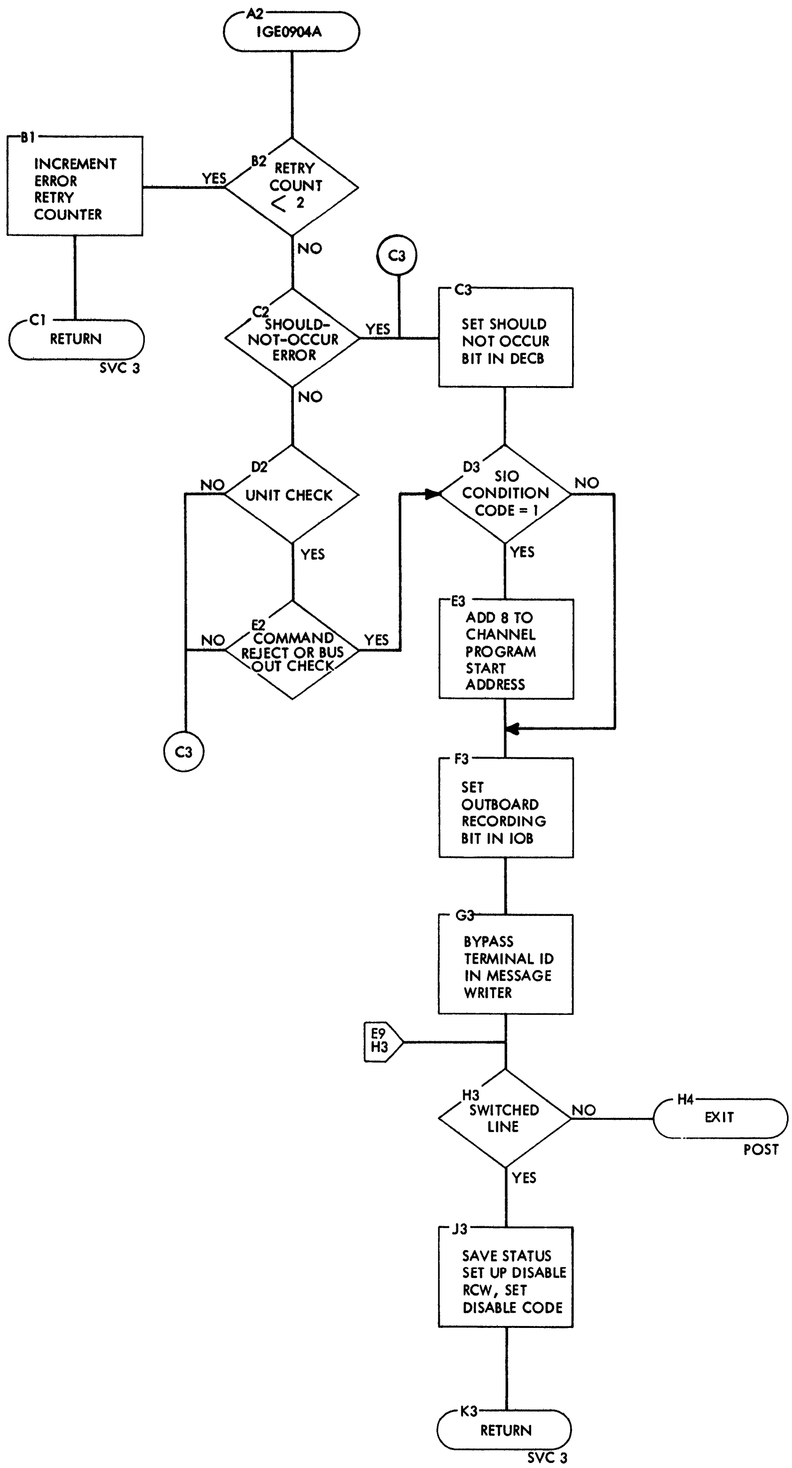


Chart F0. Start/Stop ERP Data Check (IGE0004B) (Load 2)

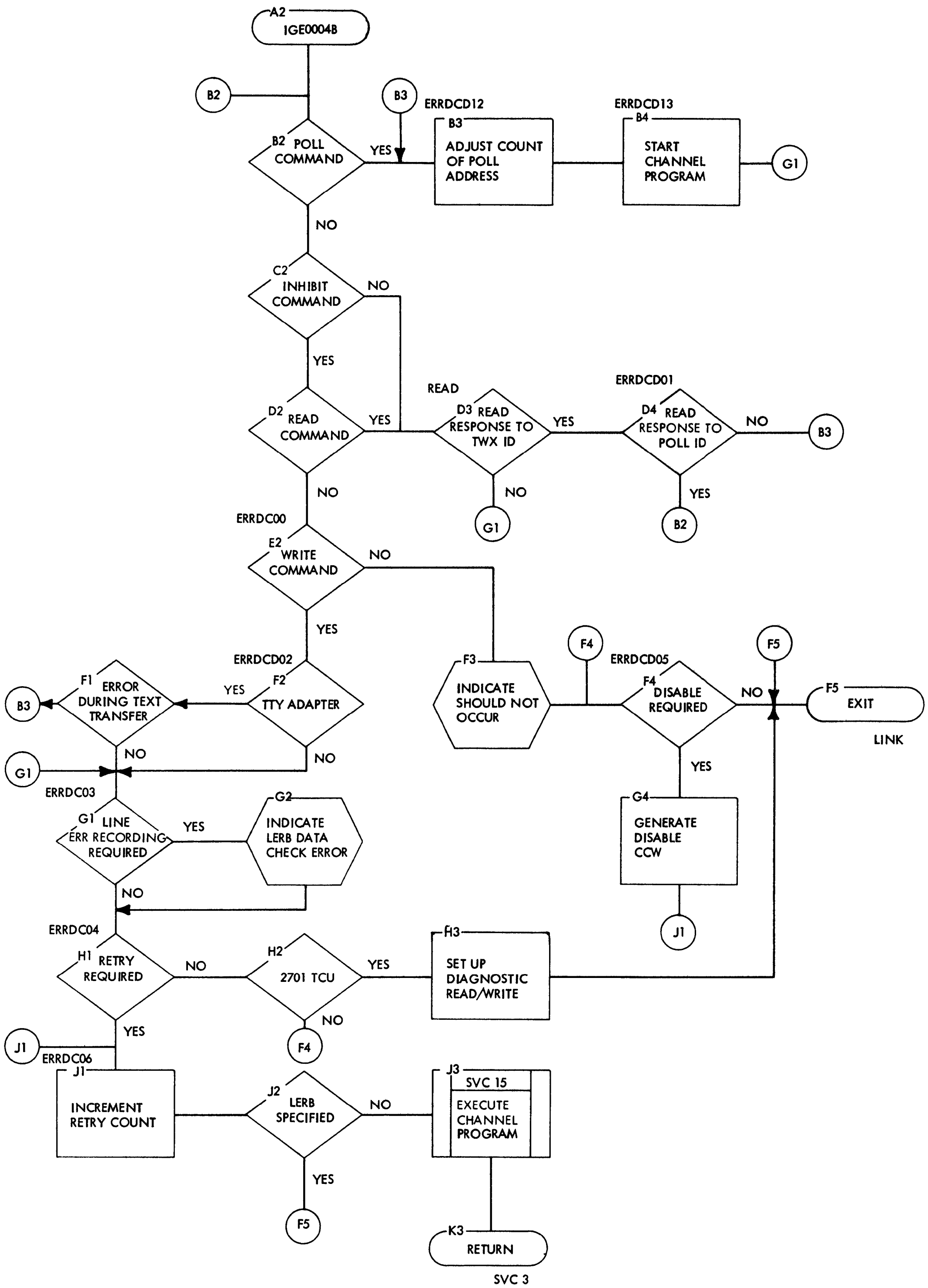


Chart F1. Start/Stop ERP Diagnostic Write/Read (IGE0104B)

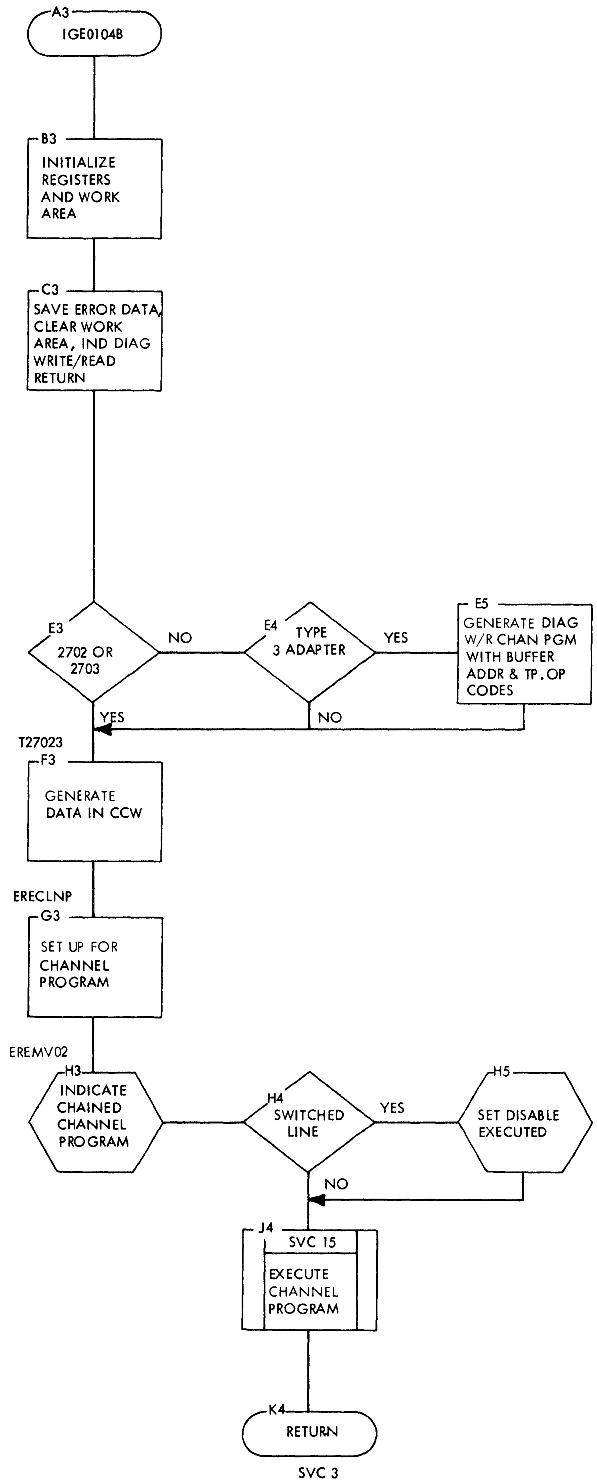


Chart F2. ERP Line Error Recording (Start/Stop and BSC) (IGE0204B)

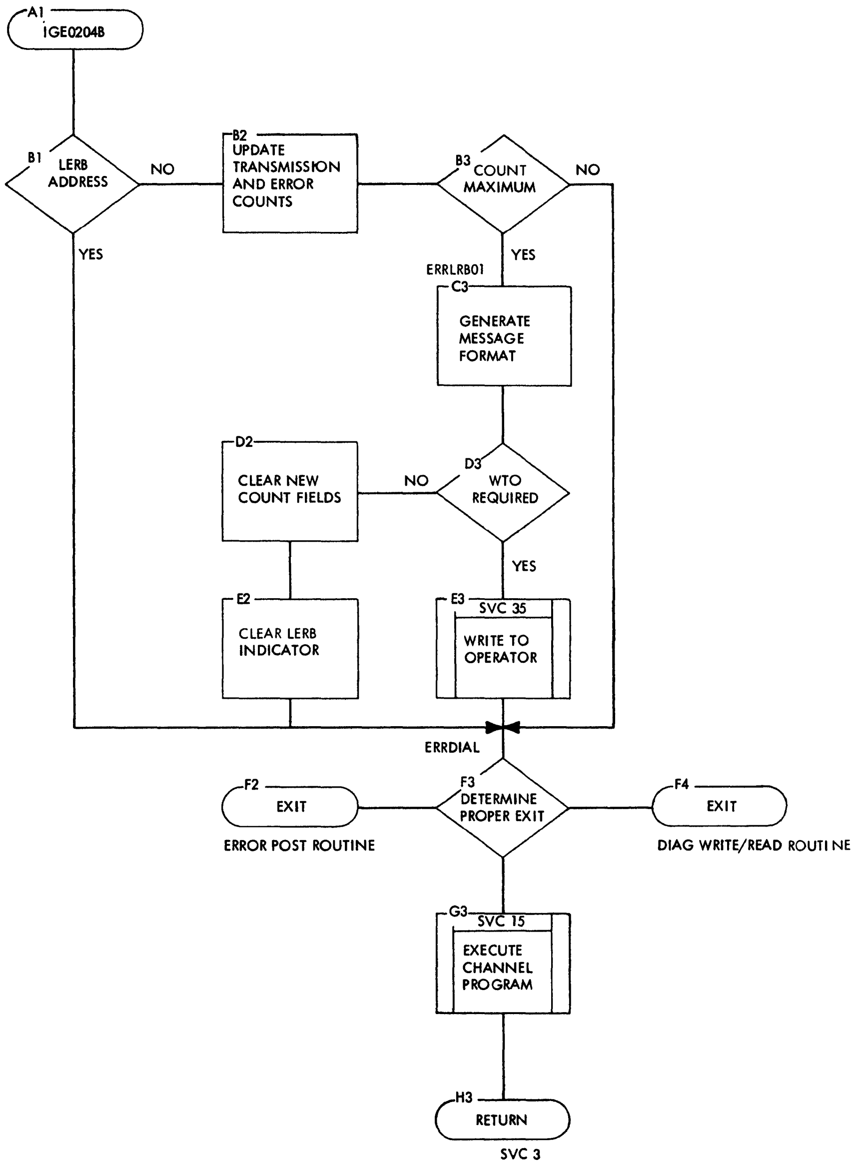


Chart F3. Start/Stop ERP Unit Exception (IGE0304B)

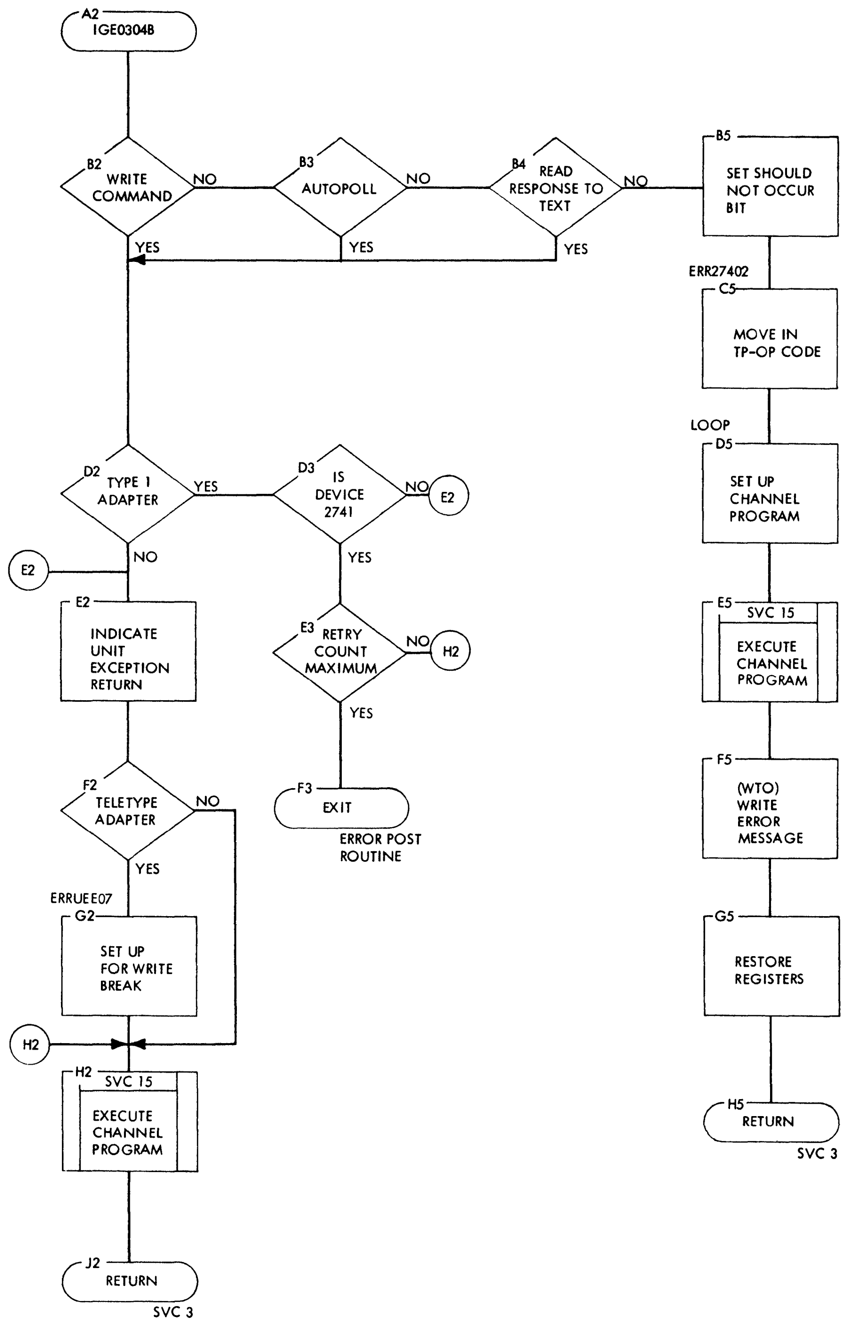


Chart F4. Start/Stop ERP Read Skip Write Break (IGE0404B) (Load 2)

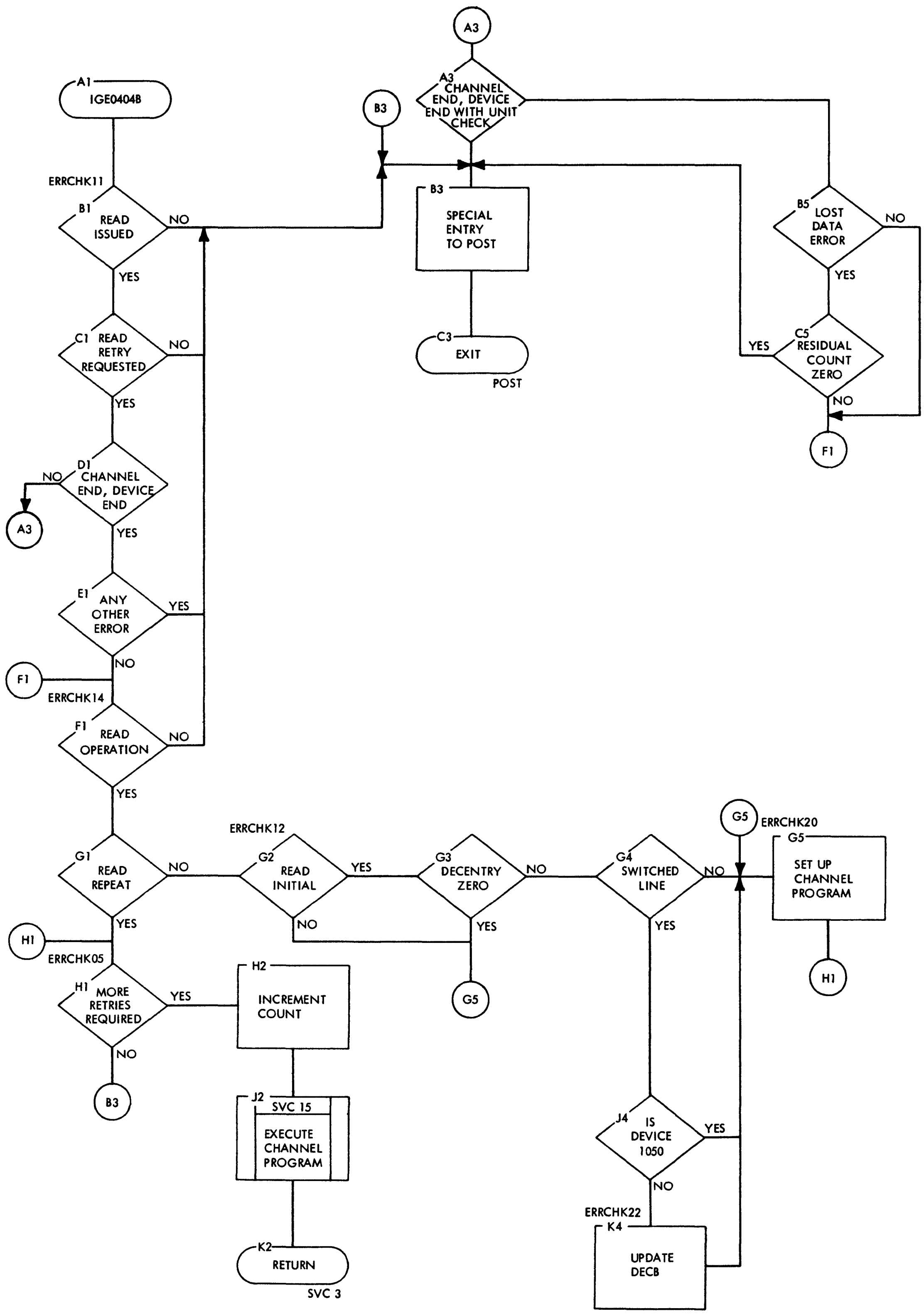


Chart F5. Start/Stop ERP Overrun (IGE0504B)

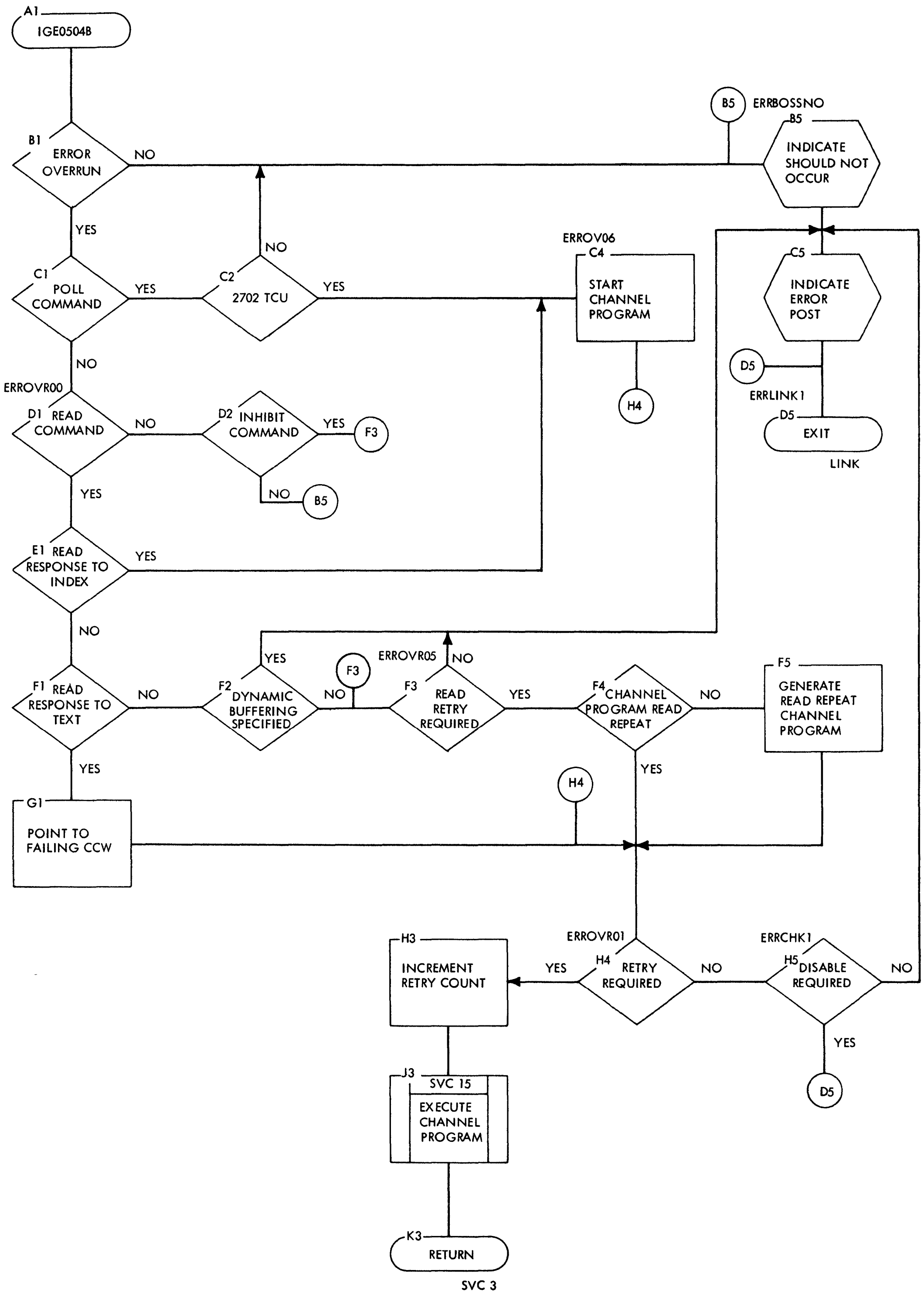


Chart F6. ERP Intervention Required Message Writer (Start/Stop & BSC) (IGE0604B)

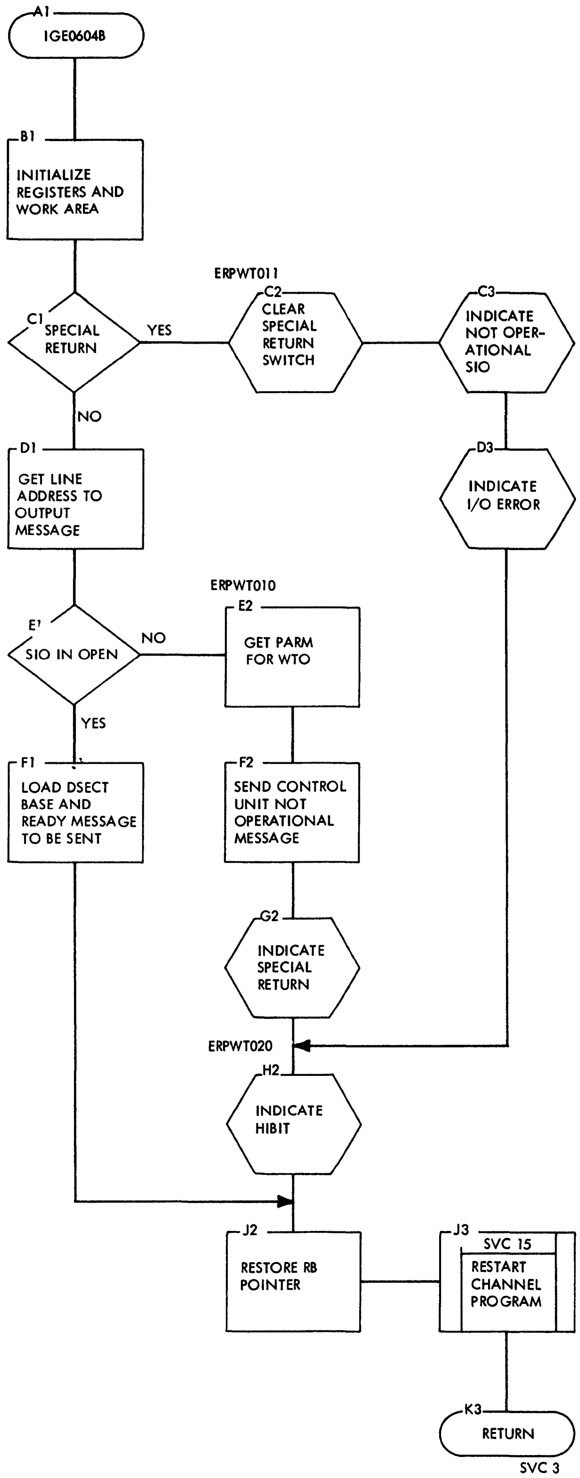


Chart F7. Line Error Print (IECTLERP)

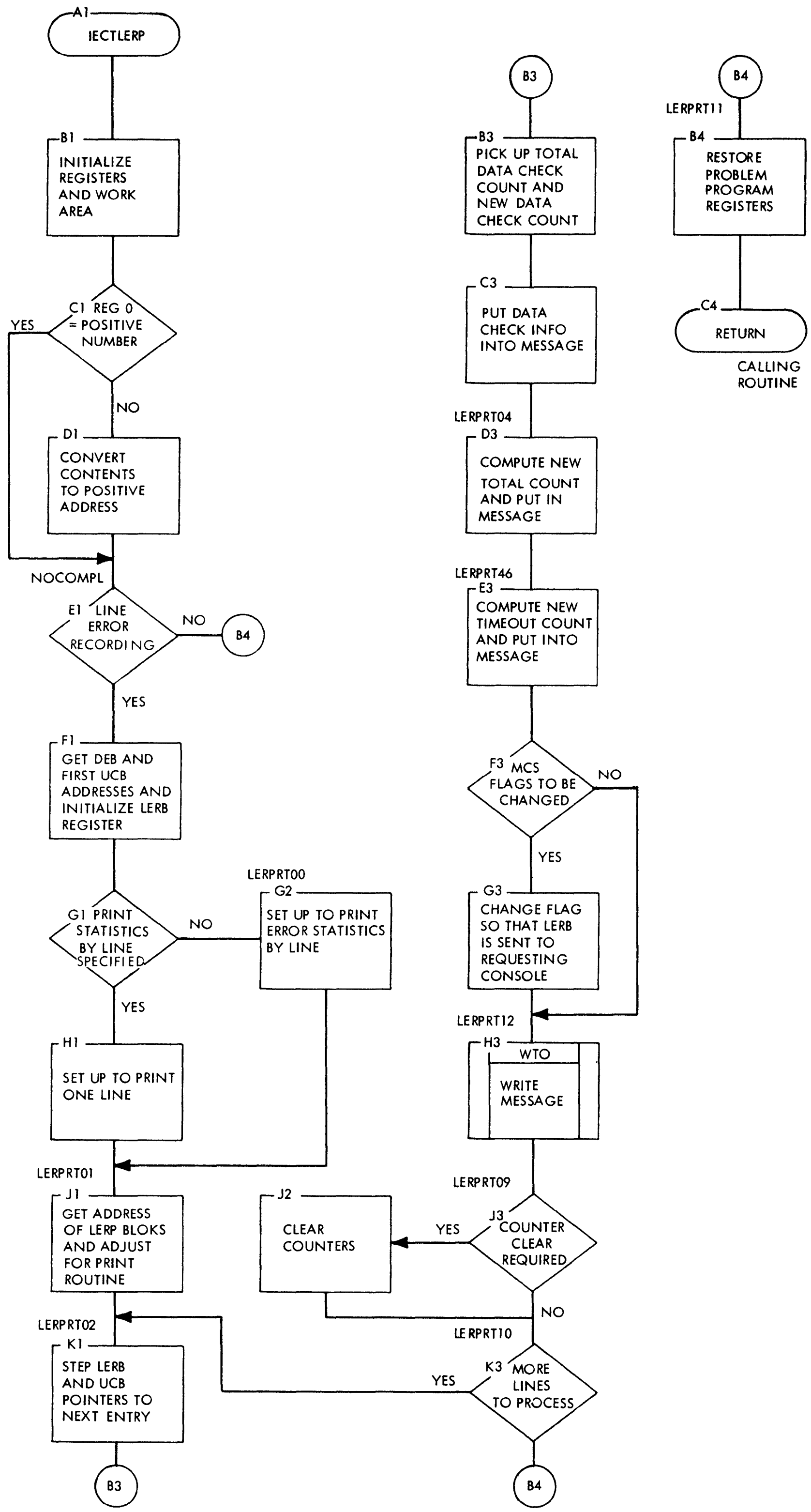


Chart F8. ERP Channel Check & Interface Control Check (Start/Stop & BSC) (IGE0804B) (Part 1 of 2)

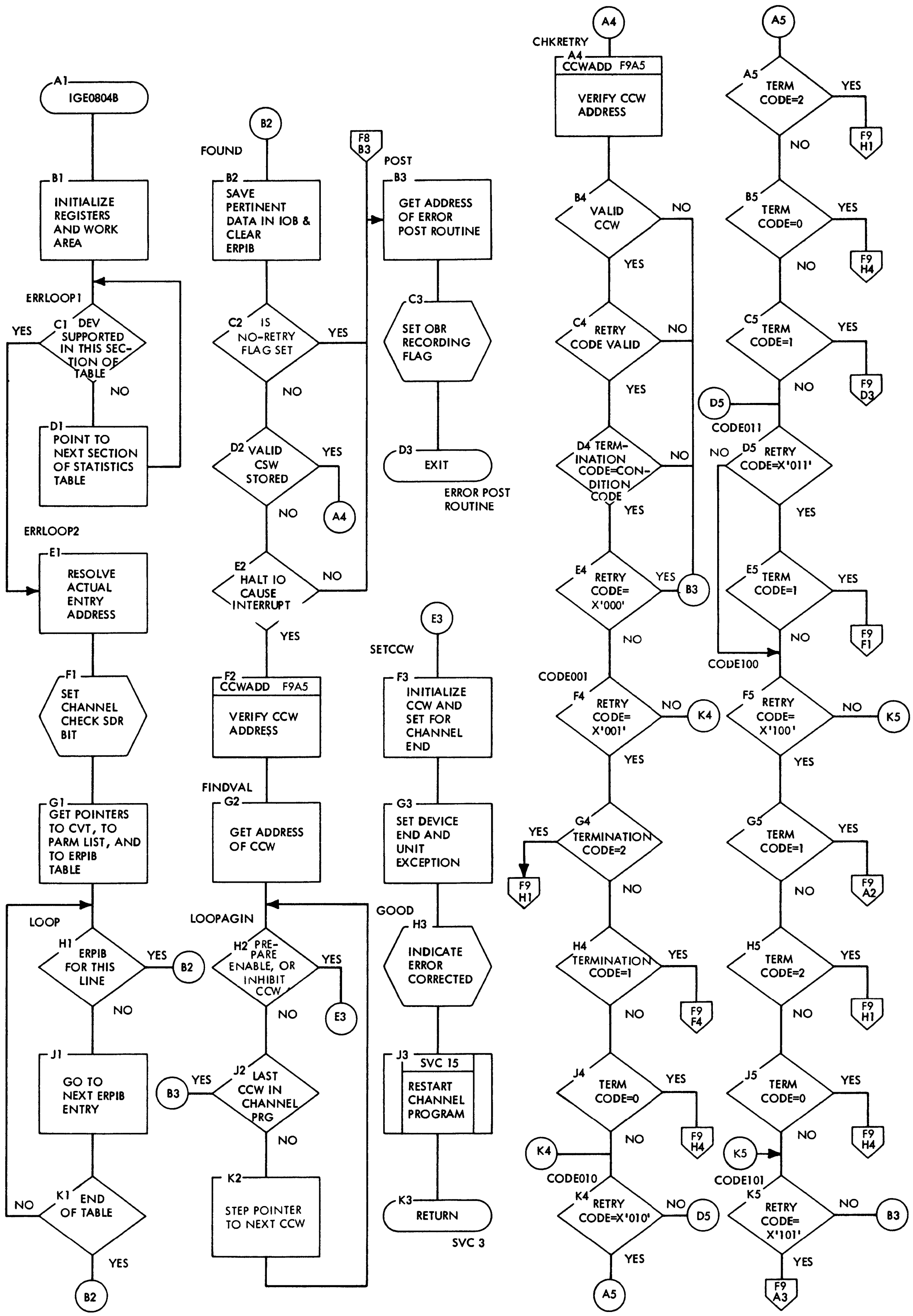


Chart F9. ERP Channel Check & Interface Control Check (Start/Stop & BSC) (IGE0804B) (Part 2 of 2)

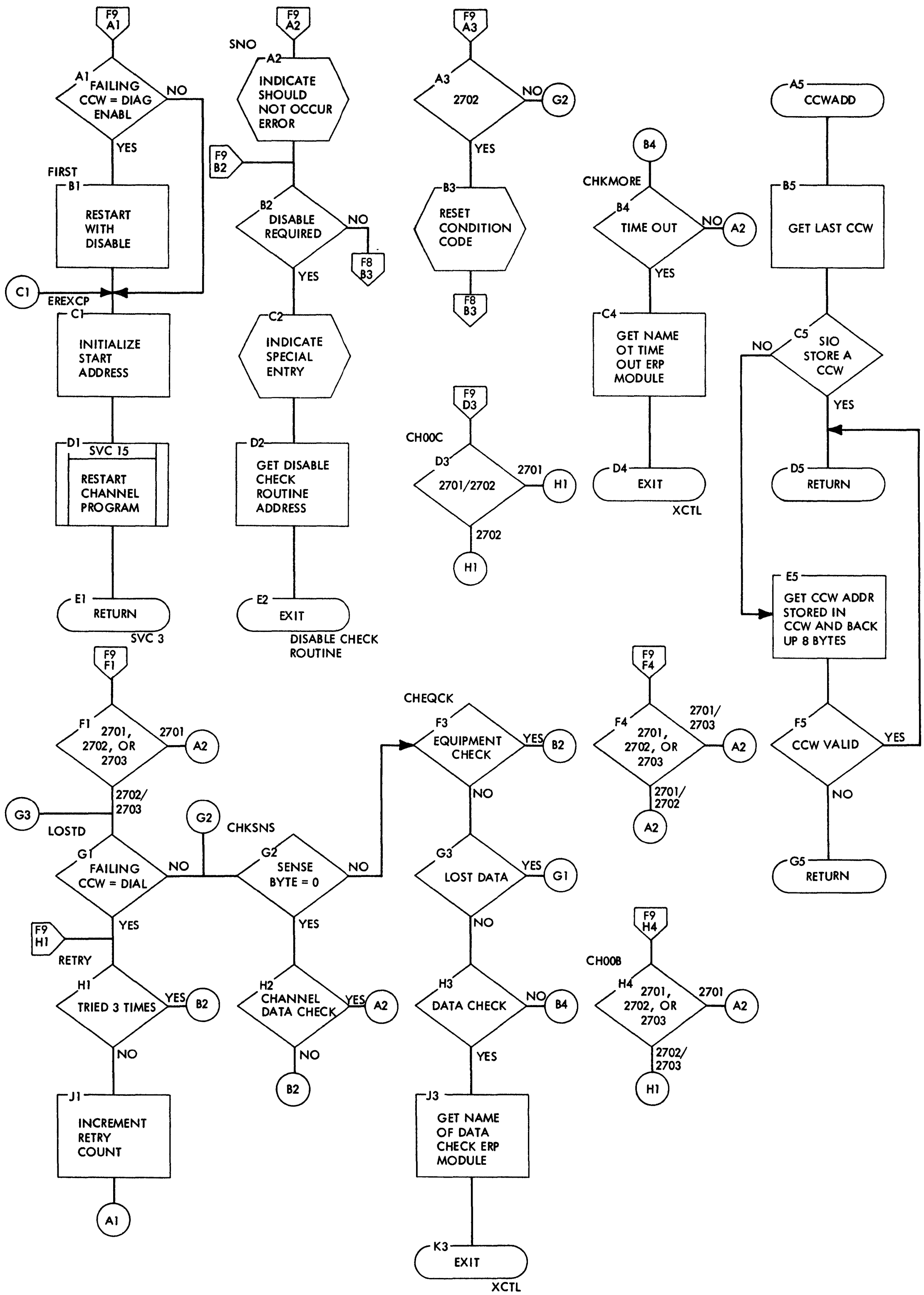


Chart G0. BSC ERP Control (IGE0004C)

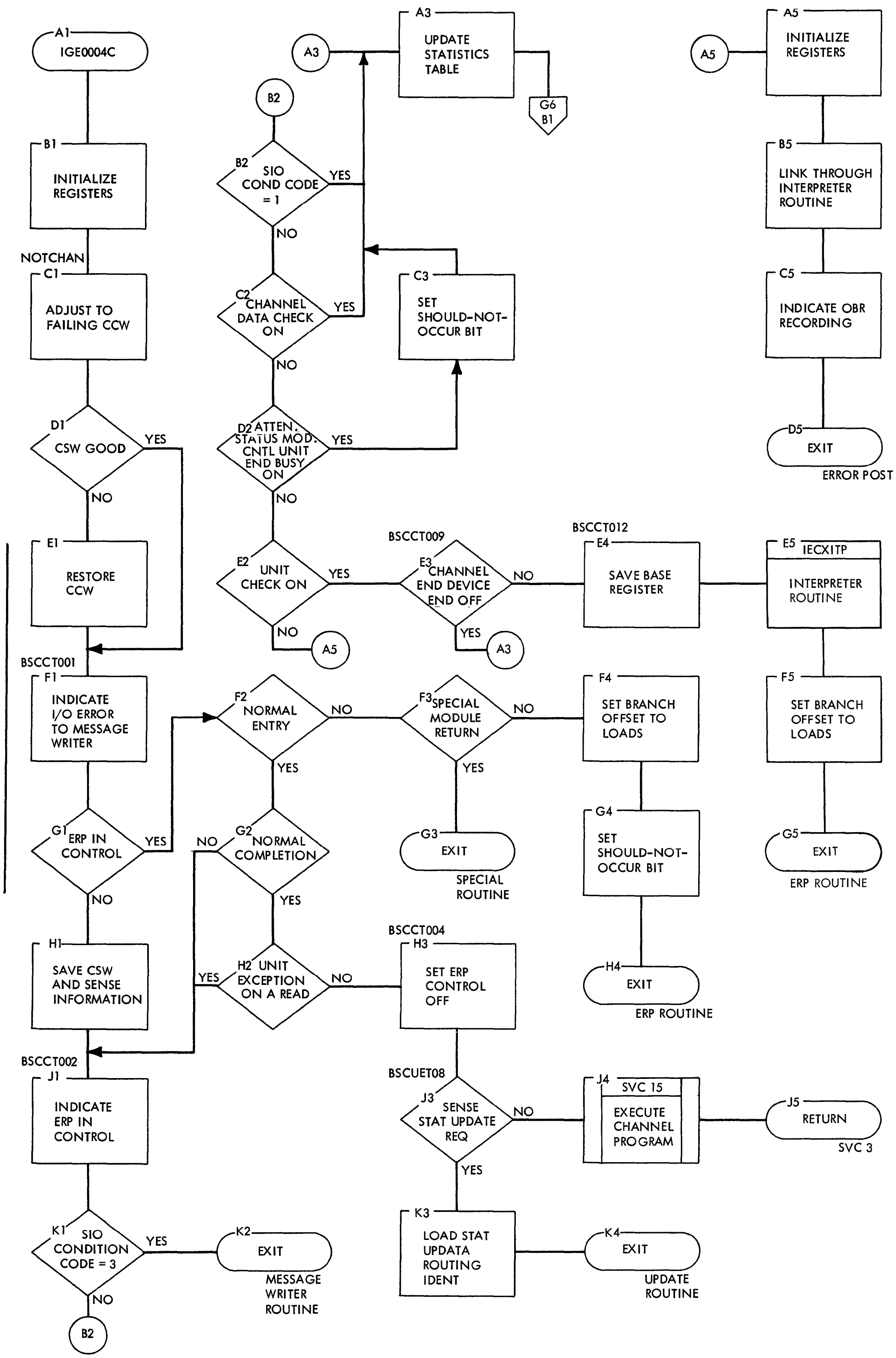


Chart G1A. BSC ERP Data Check (IGE0104C) (Part 1 of 2)

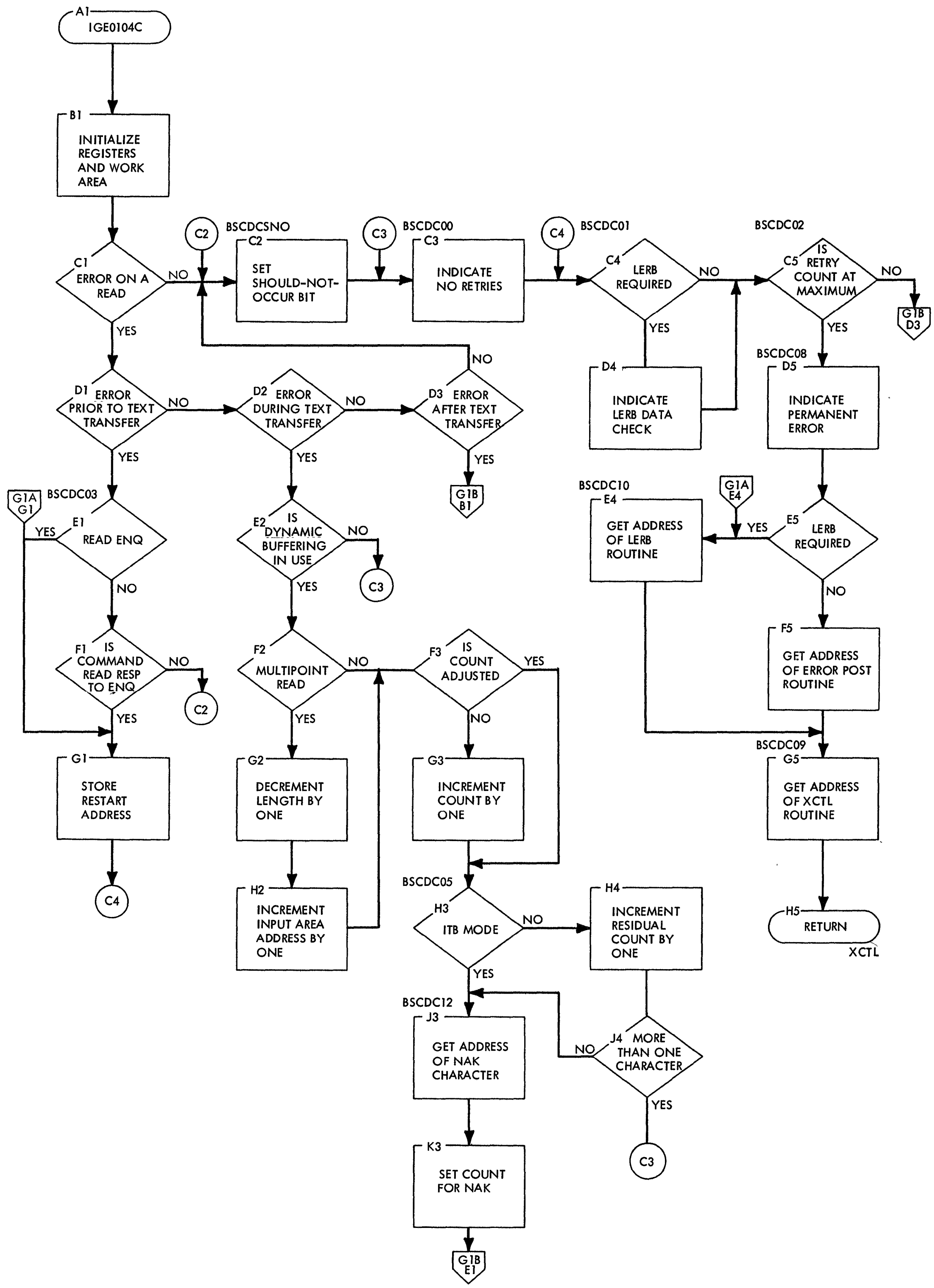


Chart G1B. BSC ERP Data Check (IGE0104C) (Part 2 of 2)

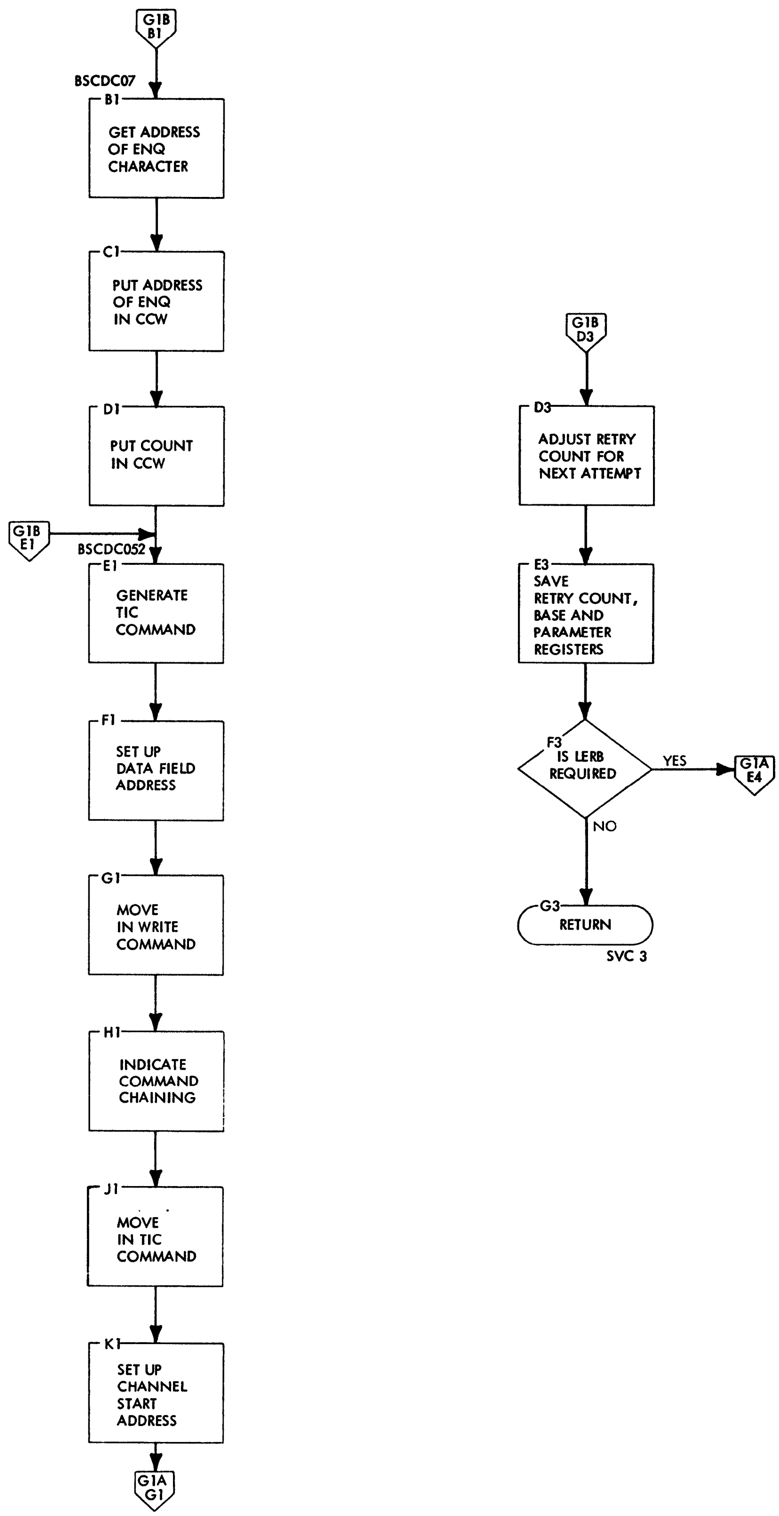


Chart G2A. BSC ERP Time Out (IGE0404C) (Part 1 of 2)

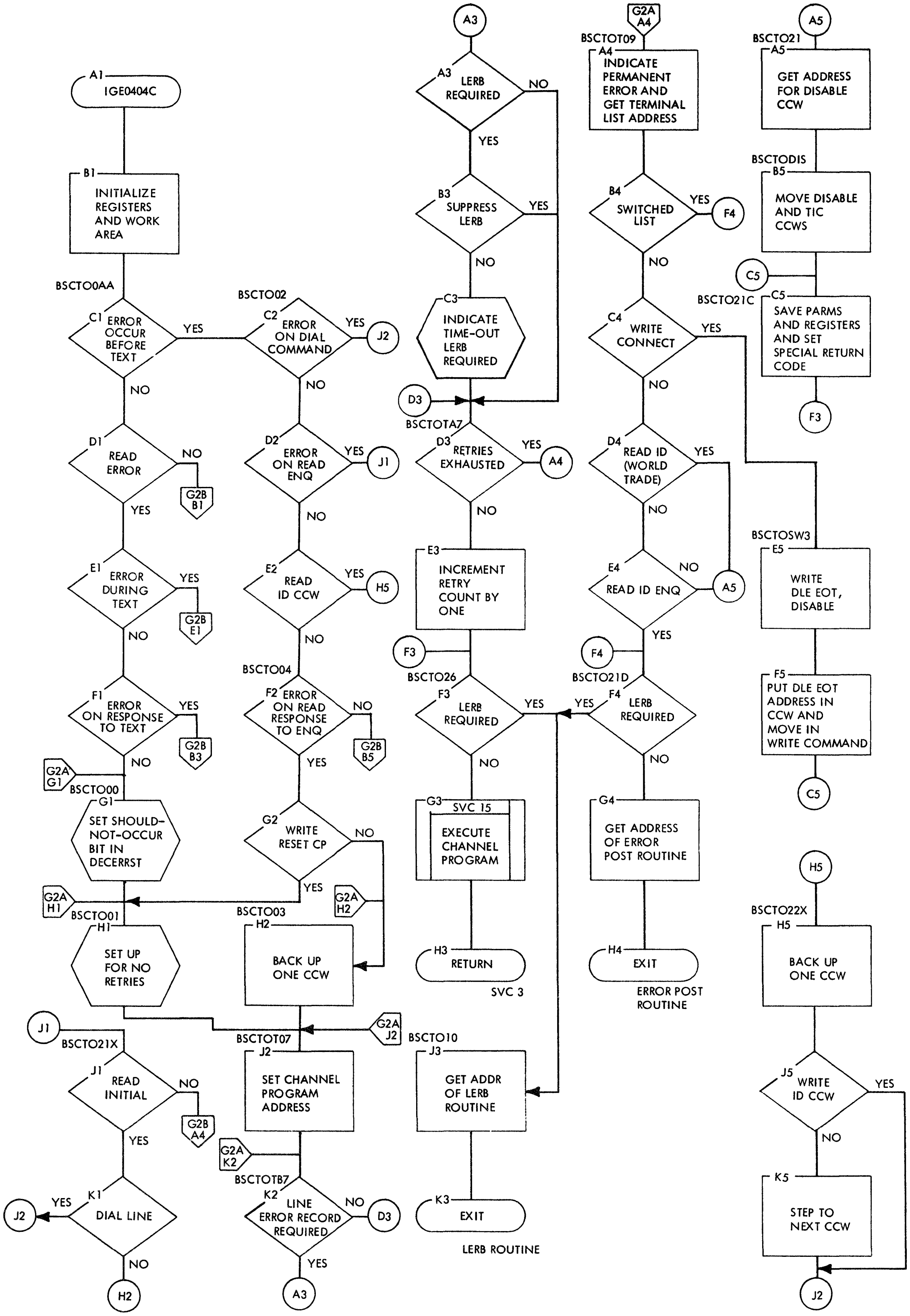


Chart G2B. BSC ERP Time Out (IGE0404C) (Part 2 of 2)

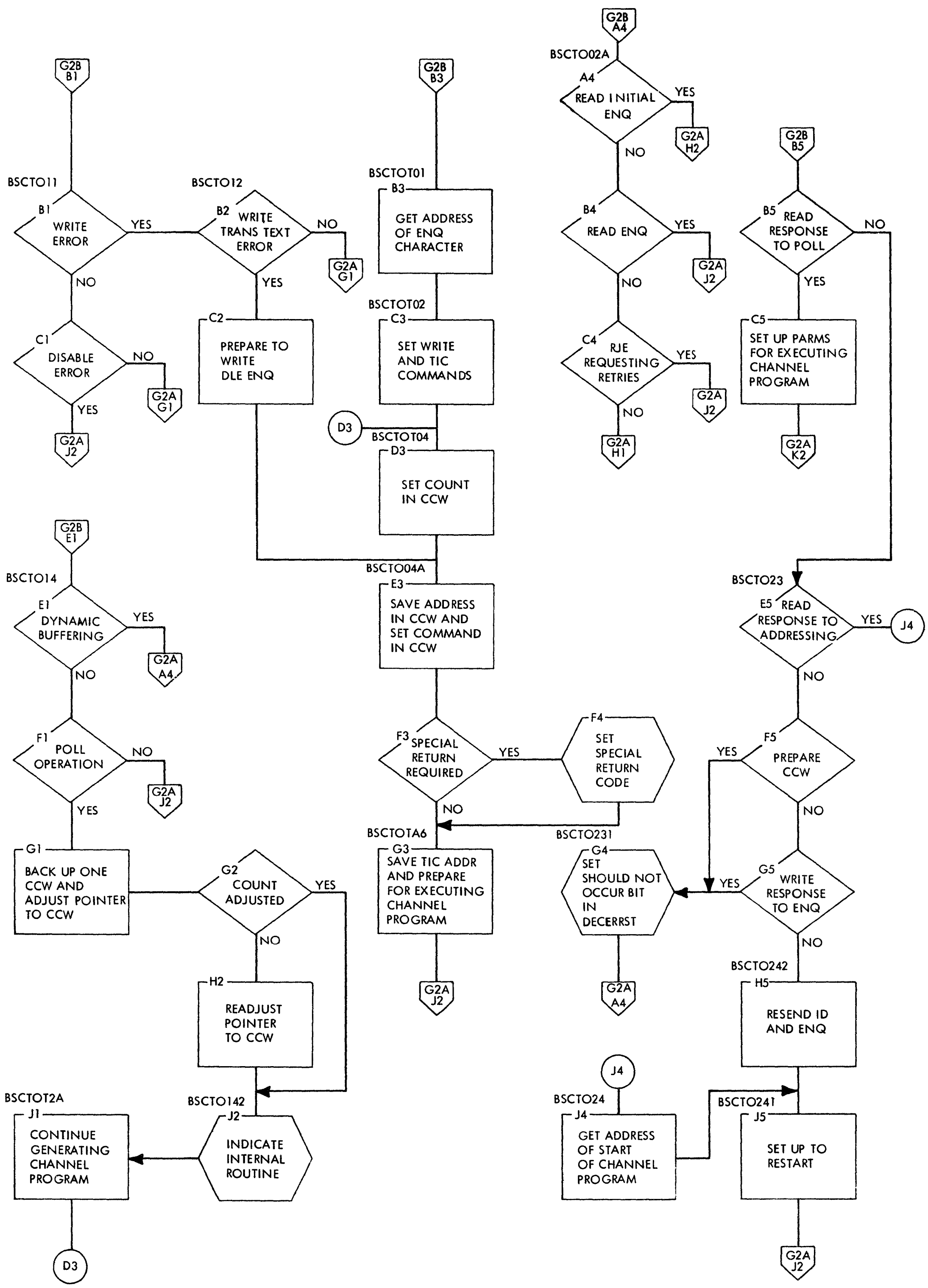


Chart G3. BSC ERP Intervention Required (IGE0304C)

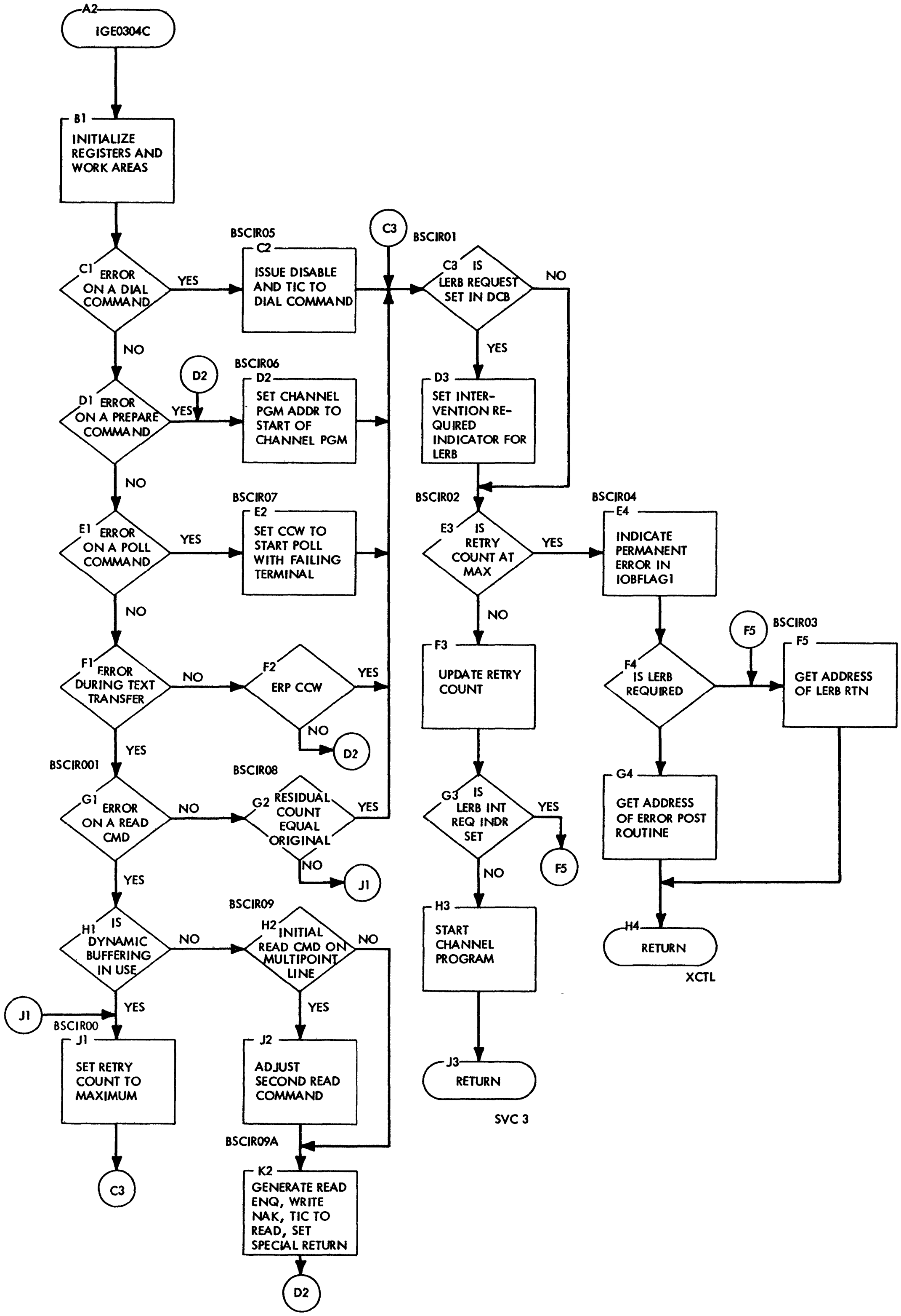


Chart G4. BSC ERP Lost Data (IGE0604C)

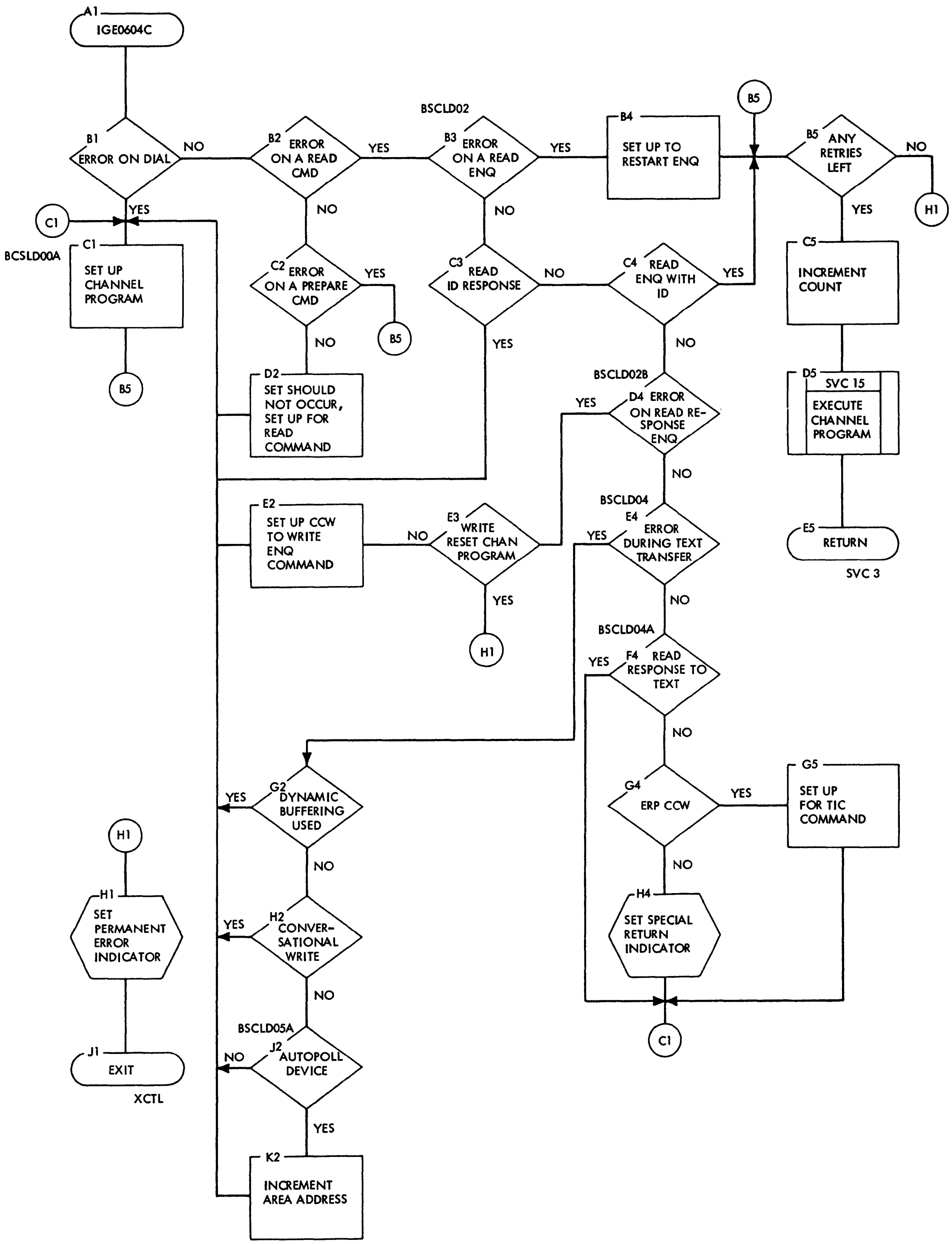


Chart G5A. BSC ERP Error Post (IGE0204C) (Part 1 of 2)

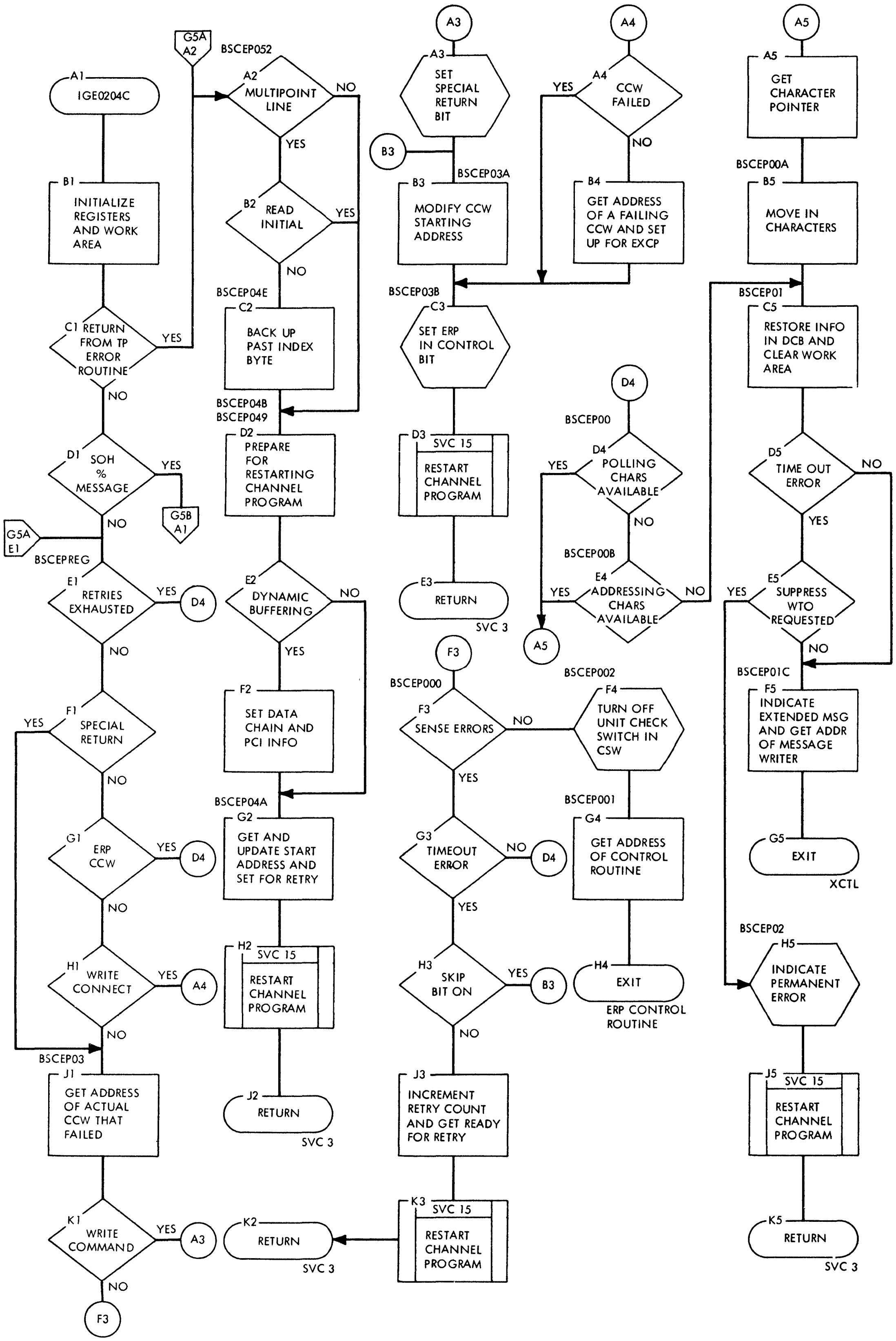


Chart G5B. BSC ERP Error Post (IGE0204C) (Part 2 of 2)

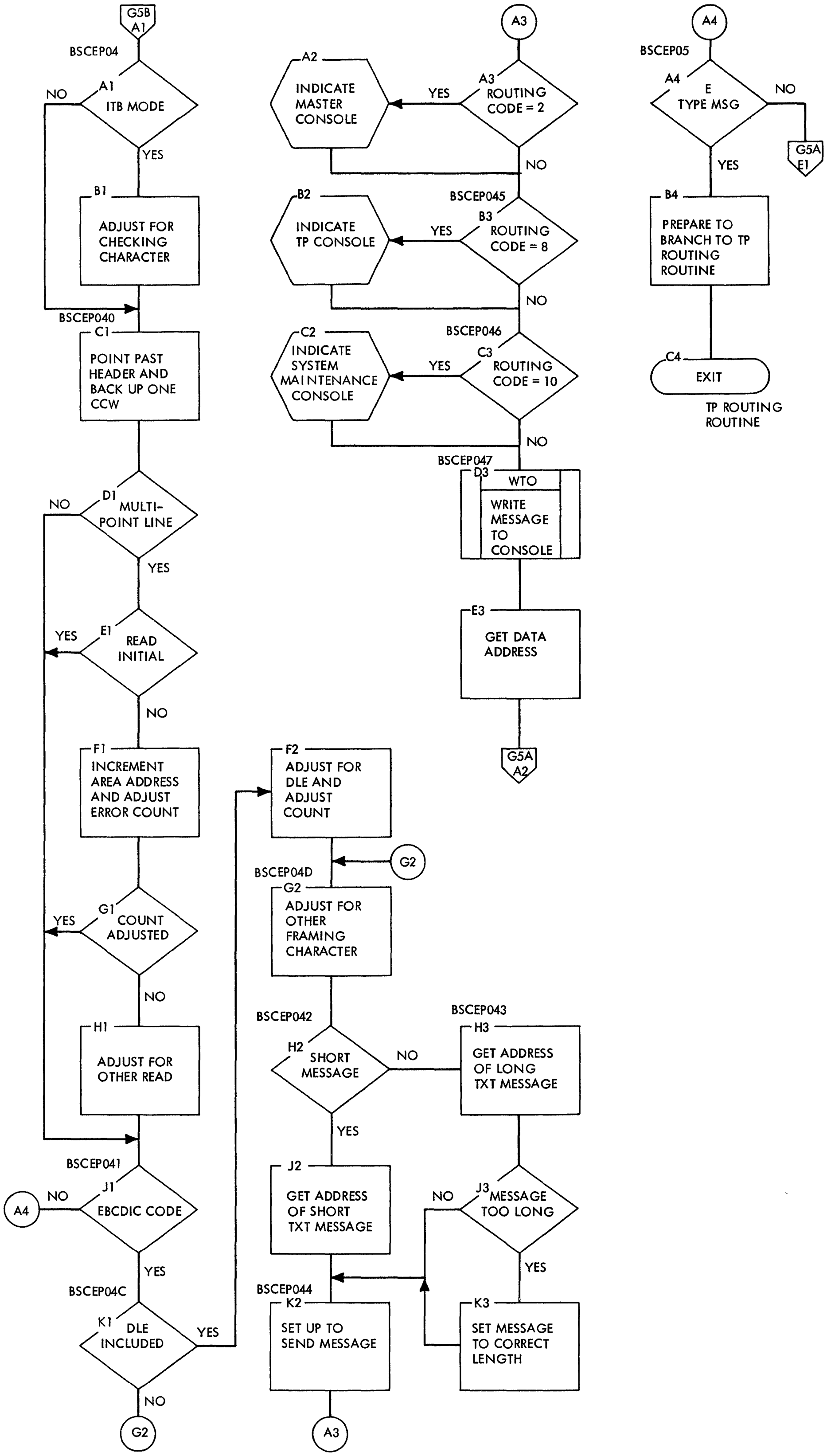


Chart G5C. Remote 3270 Error Post Routine (IGE0704B)

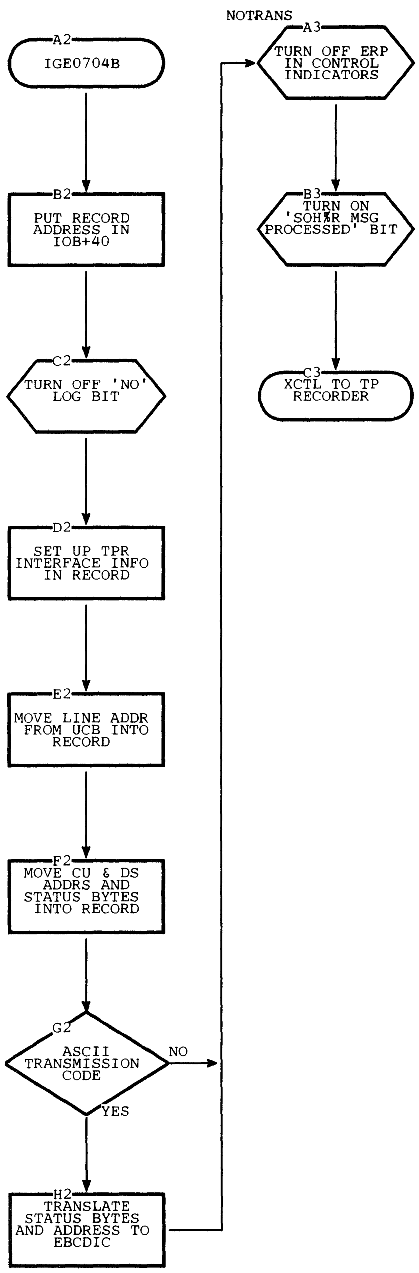
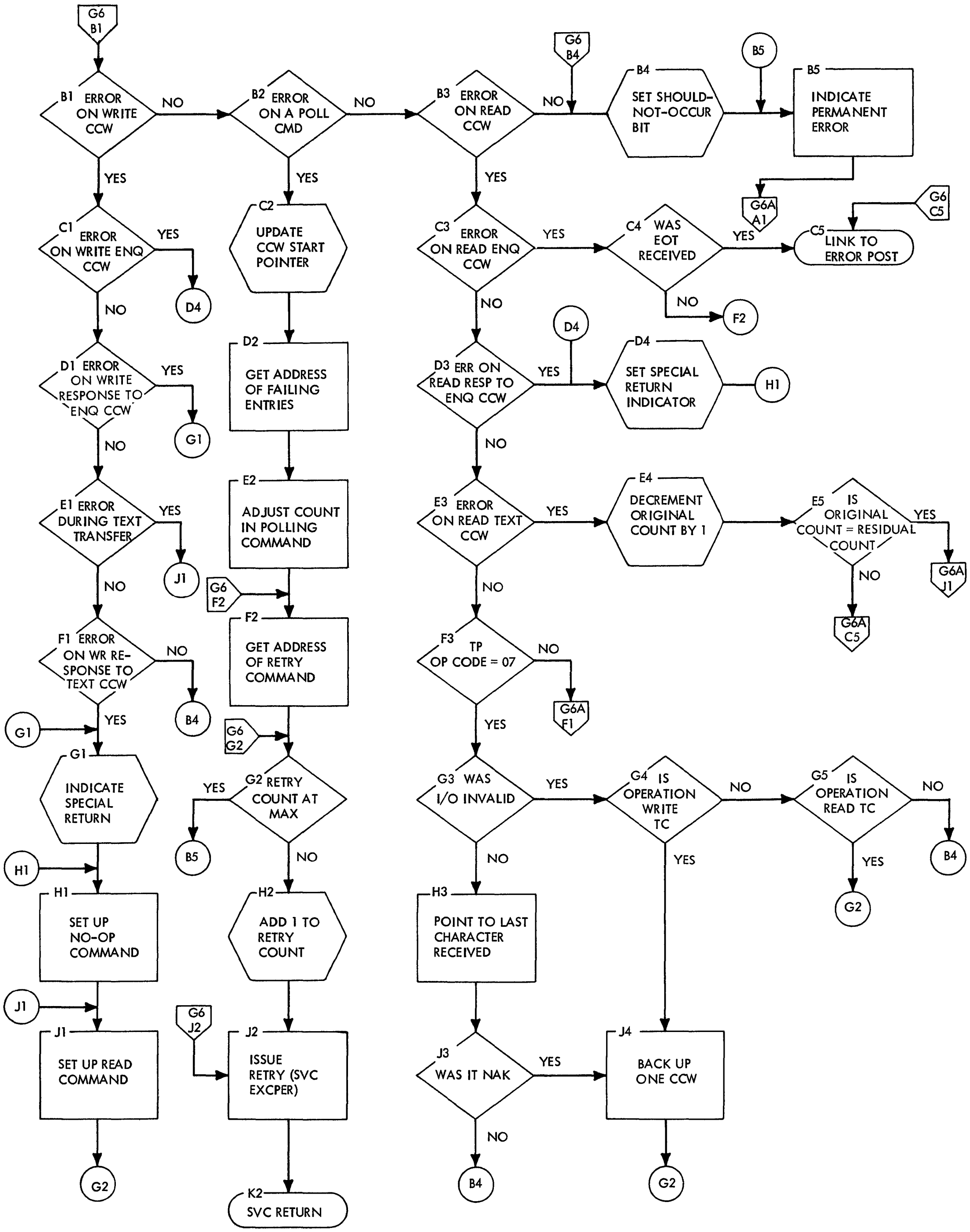


Chart G6. BSC ERP Control (IGE0004C) (Part 2 of 3)



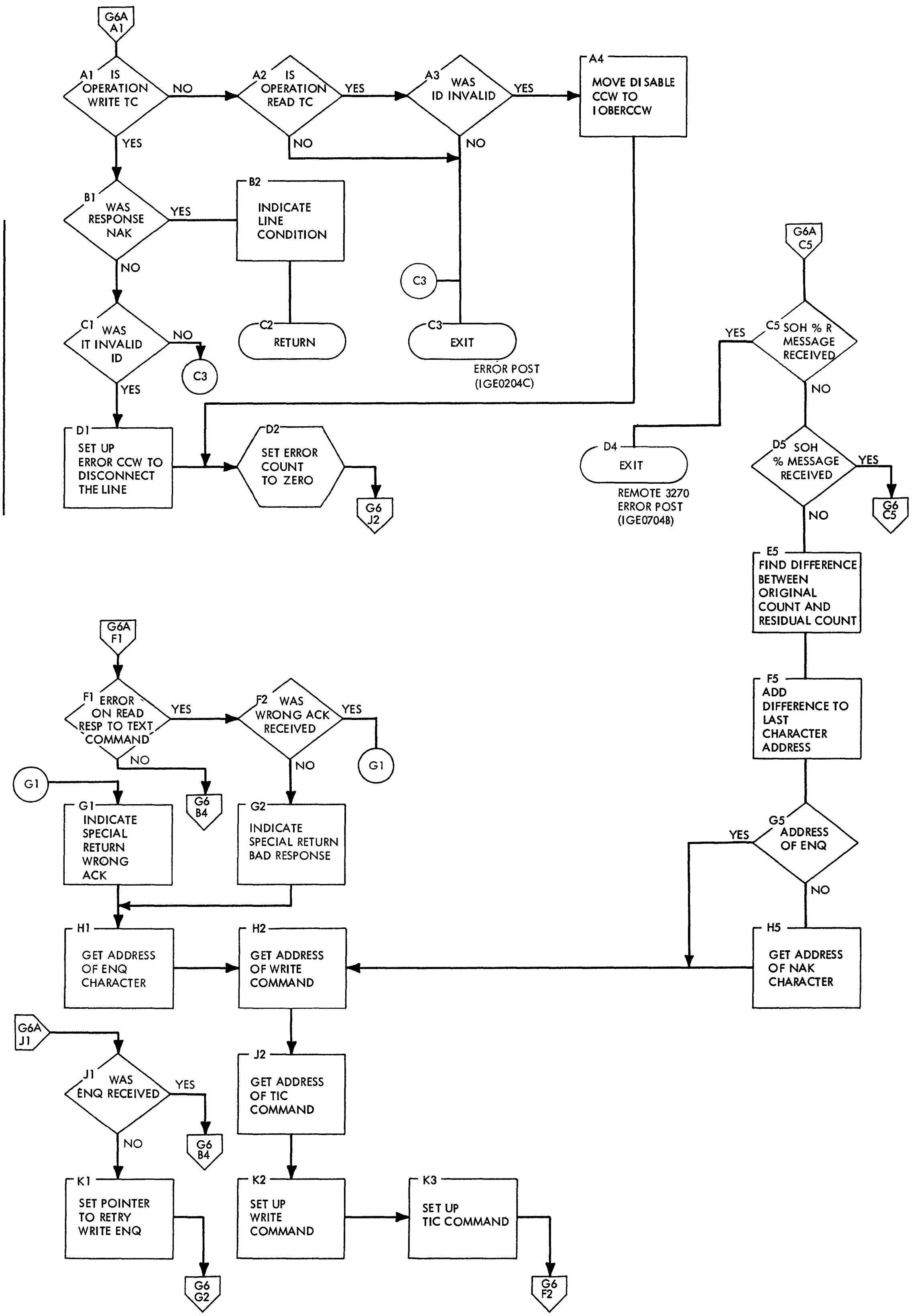

G6A A1
A1 IS OPERATION WRITE TC
NO
YES
A2 IS OPERATION READ TC
YES
NO
A3 WAS ID INVALID
YES
NO
A4 MOVE DISABLE CCW TO IOBERCCW
B1 WAS RESPONSE NAK
YES
NO
B2 INDICATE LINE CONDITION
C3
C1 WAS IT INVALID ID
NO
YES
C3
C2 RETURN
C3 EXIT
ERROR POST (IGE0204C)
G6A C5
C5 SOH % R MESSAGE RECEIVED
YES
NO
D1 SET UP ERROR CCW TO DISCONNECT THE LINE
D2 SET ERROR COUNT TO ZERO
G6 J2
D4 EXIT
REMOTE 3270 ERROR POST (IGE0704B)
D5 SOH % MESSAGE RECEIVED
YES
NO
G6 C5
E5 FIND DIFFERENCE BETWEEN ORIGINAL COUNT AND RESIDUAL COUNT
G6A F1
F1 ERROR ON READ RESP TO TEXT COMMAND
YES
NO
F2 WAS WRONG ACK RECEIVED
YES
NO
G1
F5 ADD DIFFERENCE TO LAST CHARACTER ADDRESS
G1
G6 B4
G1 INDICATE SPECIAL RETURN WRONG ACK
G2 INDICATE SPECIAL RETURN BAD RESPONSE
G5 ADDRESS OF ENQ
YES
NO
H1 GET ADDRESS OF ENQ CHARACTER
H2 GET ADDRESS OF WRITE COMMAND
H5 GET ADDRESS OF NAK CHARACTER
G6A J1
J1 WAS ENQ RECEIVED
YES
NO
G6 B4
J2 GET ADDRESS OF TIC COMMAND
K1 SET POINTER TO RETRY WRITE ENQ
K2 SET UP WRITE COMMAND
K3 SET UP TIC COMMAND
G6 G2
G6 F2

Chart G9A. BSC ERP Equipment Check & Command Reject (IGE0804C) (Part 1 of 2)

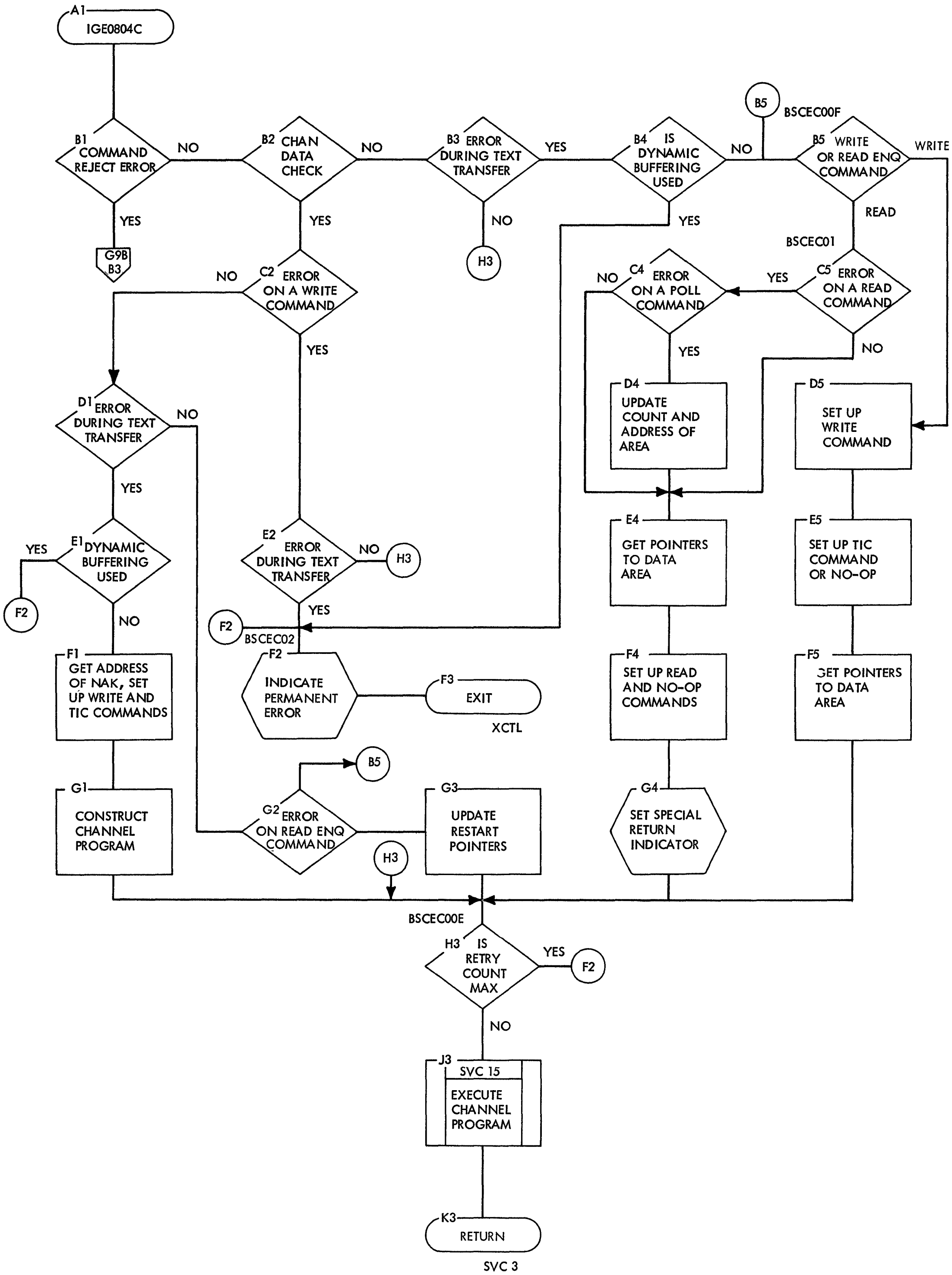


Chart G9B. BSC ERP Equipment Check & Command Reject (IGE0804C) (Part 2 of 2)

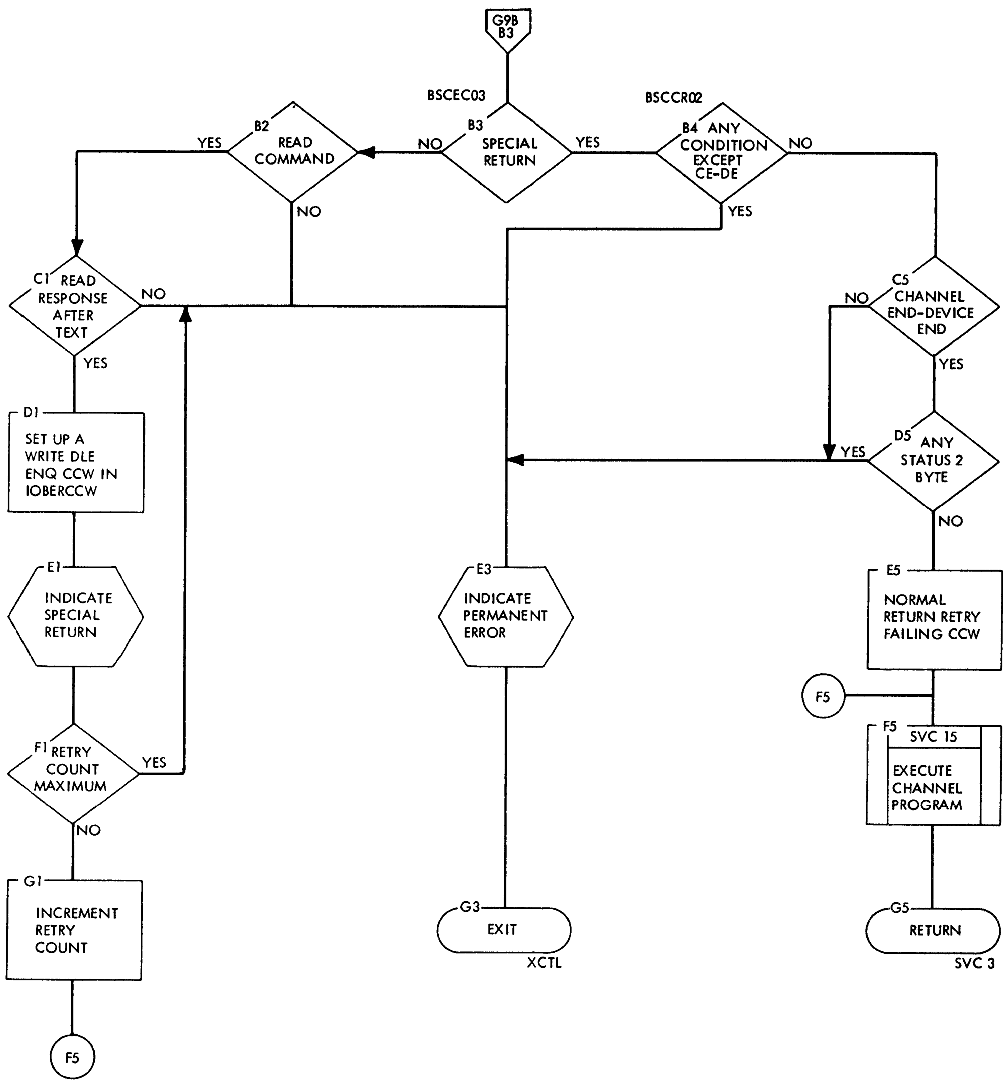
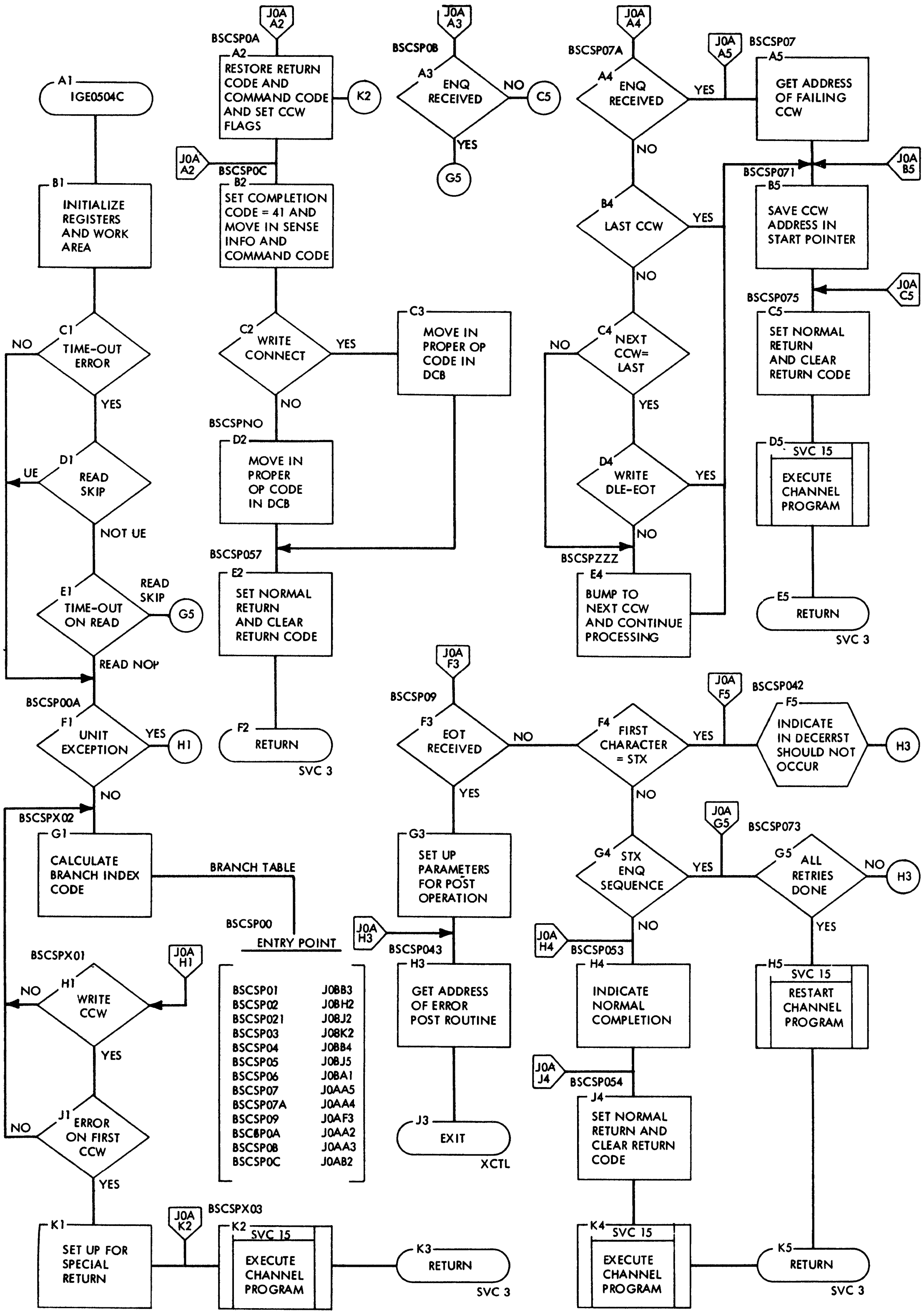
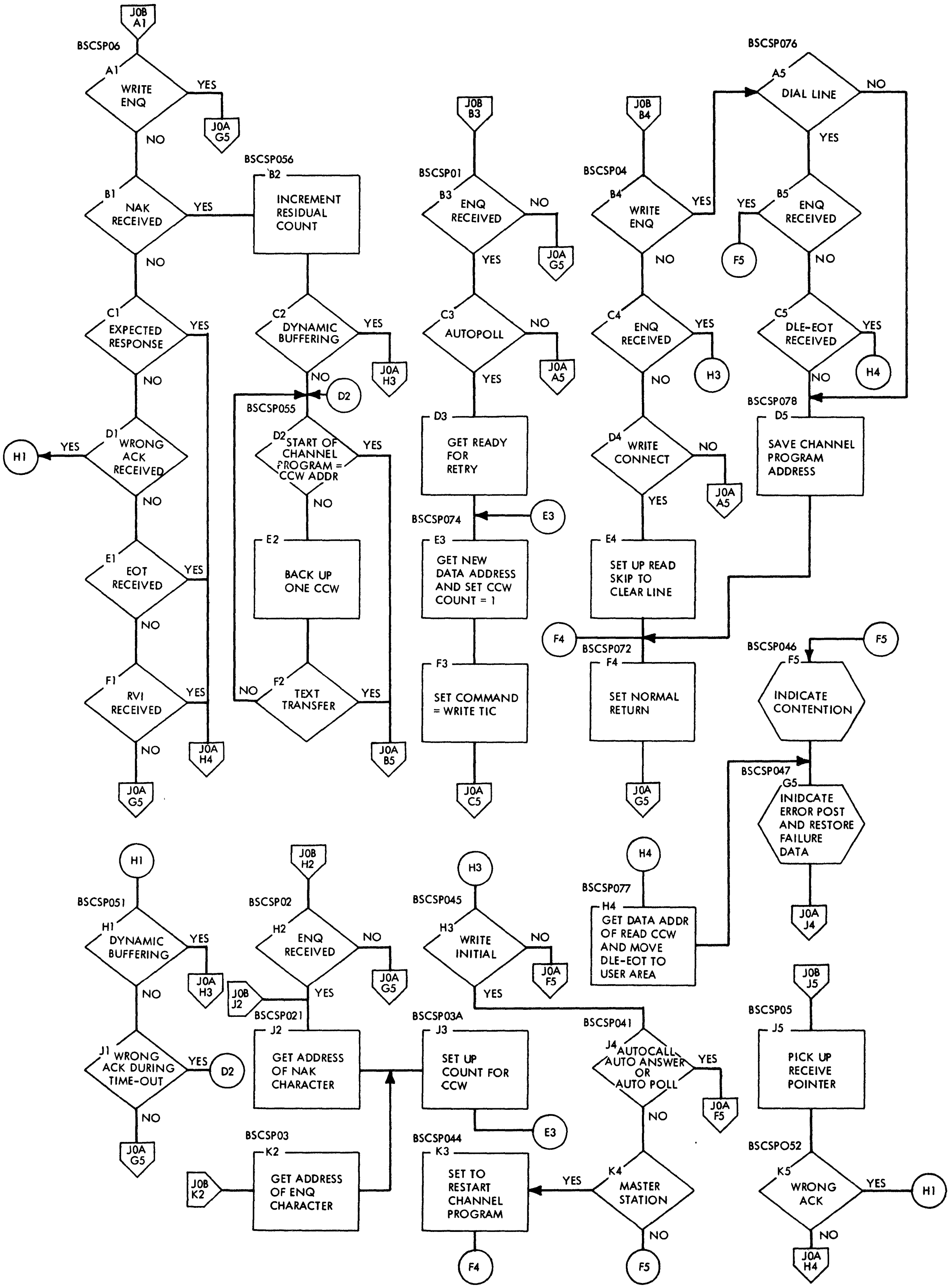


Chart J0A. BSC ERP Special Return (IGE0504C) (Part 1 of 2)





J0B A1
BSCSP06
A1 WRITE ENQ
YES
J0A G5
NO
B1 NAK RECEIVED
YES
BSCSP056
B2 INCREMENT RESIDUAL COUNT
NO
C1 EXPECTED RESPONSE
YES
NO
D1 WRONG ACK RECEIVED
YES
H1
NO
E1 EOT RECEIVED
YES
NO
F1 RVI RECEIVED
YES
NO
J0A G5
J0A H4
C2 DYNAMIC BUFFERING
YES
J0A H3
NO
D2
BSCSP055
D2 START OF CHANNEL PROGRAM = CCW ADDR
YES
NO
E2 BACK UP ONE CCW
F2 TEXT TRANSFER
NO
YES
J0A B5
J0B B3
BSCSP01
B3 ENQ RECEIVED
NO
J0A G5
YES
C3 AUTOPOLL
NO
J0A A5
YES
D3 GET READY FOR RETRY
E3
BSCSP074
E3 GET NEW DATA ADDRESS AND SET CCW COUNT = 1
F3 SET COMMAND = WRITE TIC
J0A C5
J0B B4
BSCSP04
B4 WRITE ENQ
YES
NO
C4 ENQ RECEIVED
YES
H3
NO
D4 WRITE CONNECT
NO
J0A A5
YES
E4 SET UP READ SKIP TO CLEAR LINE
F4
BSCSP072
F4 SET NORMAL RETURN
J0A G5
BSCSP076
A5 DIAL LINE
NO
YES
B5 ENQ RECEIVED
YES
F5
NO
C5 DLE-EOT RECEIVED
YES
H4
NO
BSCSP078
D5 SAVE CHANNEL PROGRAM ADDRESS
BSCSP046
F5
F5 INDICATE CONTENTION
BSCSP047
G5 INIDCATE ERROR POST AND RESTORE FAILURE DATA
J0A J4
H1
BSCSP051
H1 DYNAMIC BUFFERING
YES
J0A H3
NO
J1 WRONG ACK DURING TIME-OUT
YES
D2
NO
J0A G5
J0B H2
BSCSP02
H2 ENQ RECEIVED
NO
J0A G5
YES
J0B J2
BSCSP021
J2 GET ADDRESS OF NAK CHARACTER
BSCSP03
K2 GET ADDRESS OF ENQ CHARACTER
J0B K2
H3
BSCSP045
H3 WRITE INITIAL
NO
J0A F5
YES
BSCSP03A
J3 SET UP COUNT FOR CCW
E3
BSCSP044
K3 SET TO RESTART CHANNEL PROGRAM
F4
H4
BSCSP077
H4 GET DATA ADDR OF READ CCW AND MOVE DLE-EOT TO USER AREA
BSCSP041
J4 AUTOCALL AUTO ANSWER OR AUTO POLL
YES
J0A F5
NO
K4 MASTER STATION
YES
NO
F5
J0B J5
BSCSP05
J5 PICK UP RECEIVE POINTER
BSCSP052
K5 WRONG ACK
YES
H1
NO
J0A H4

Chart J1. BSC ERP Bus Out & Overrun (IGE0704C)

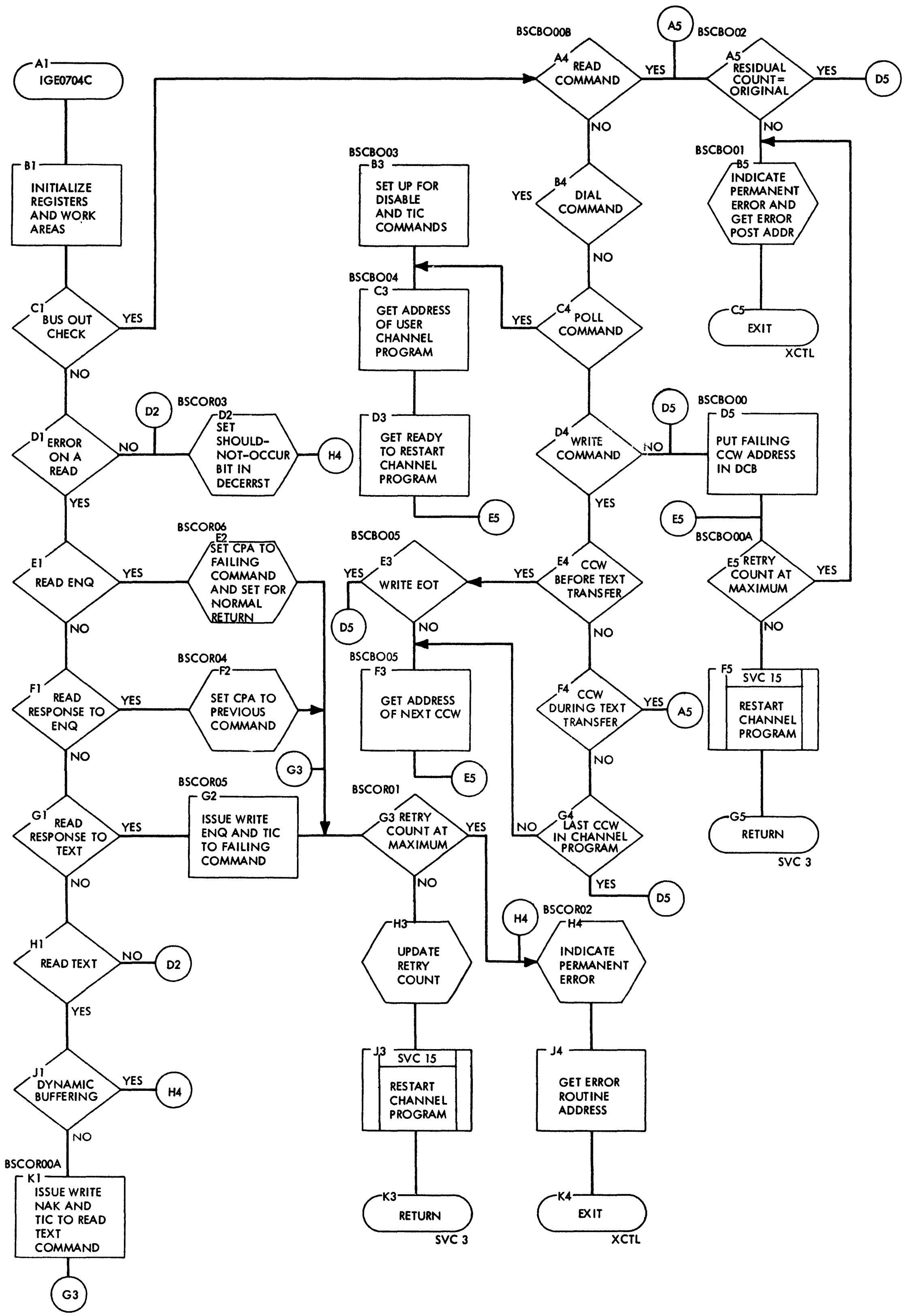


Chart J2A. BSC ERP Unit Exception Routine (IGE0904C) (Part 1 of 3)

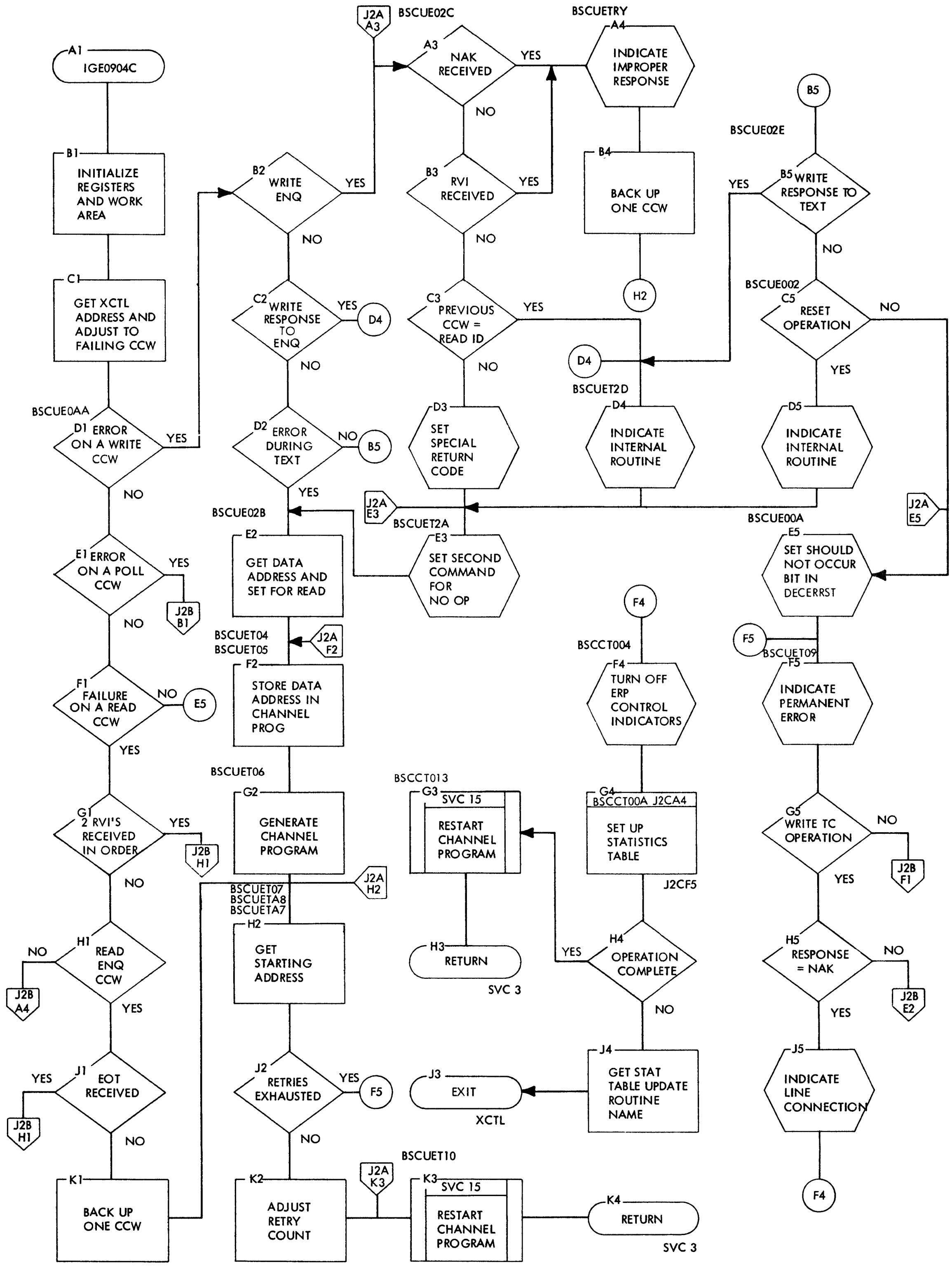
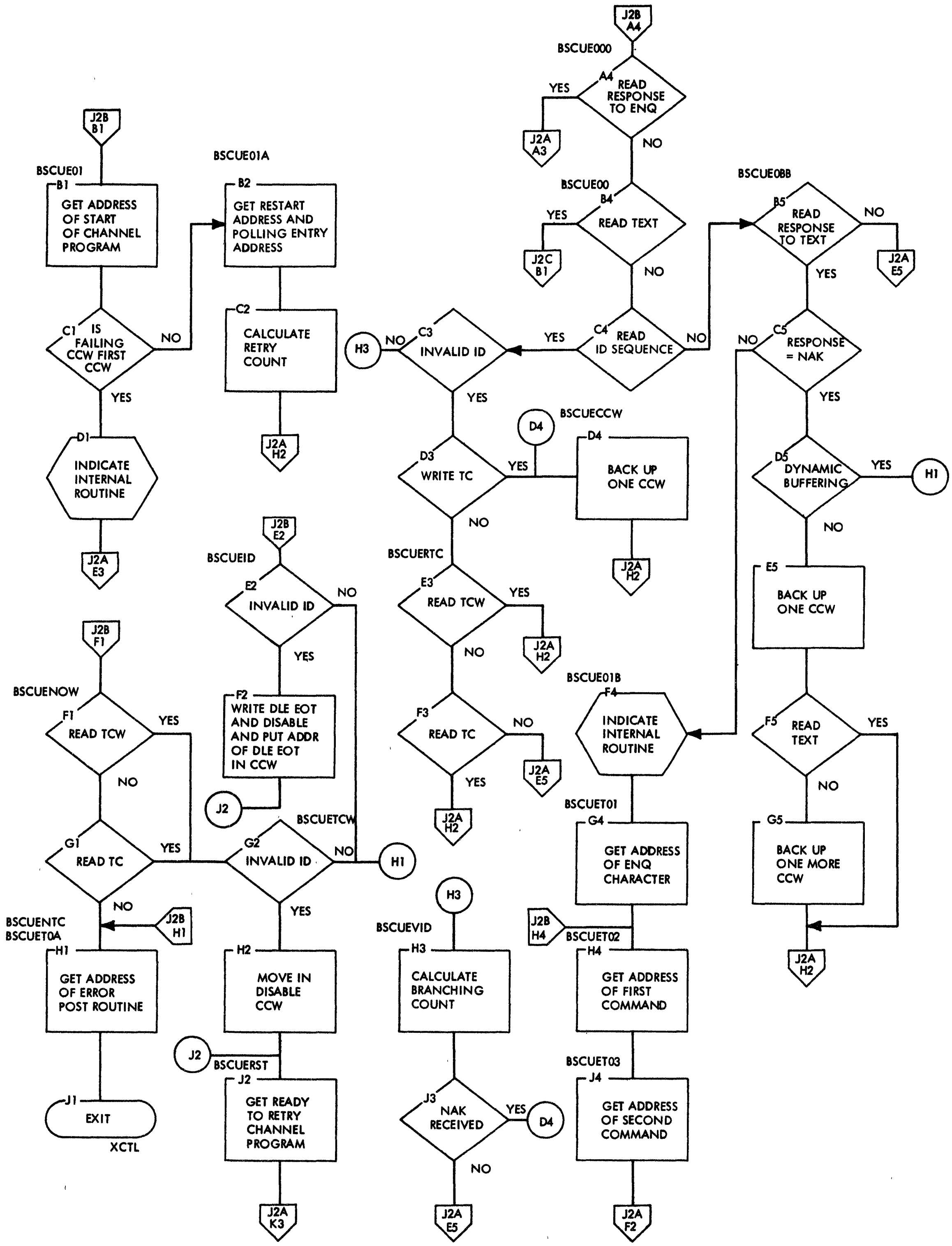


Chart J2B. BSC ERP Unit Exception Routine (IGE0904C) (Part 2 of 3)



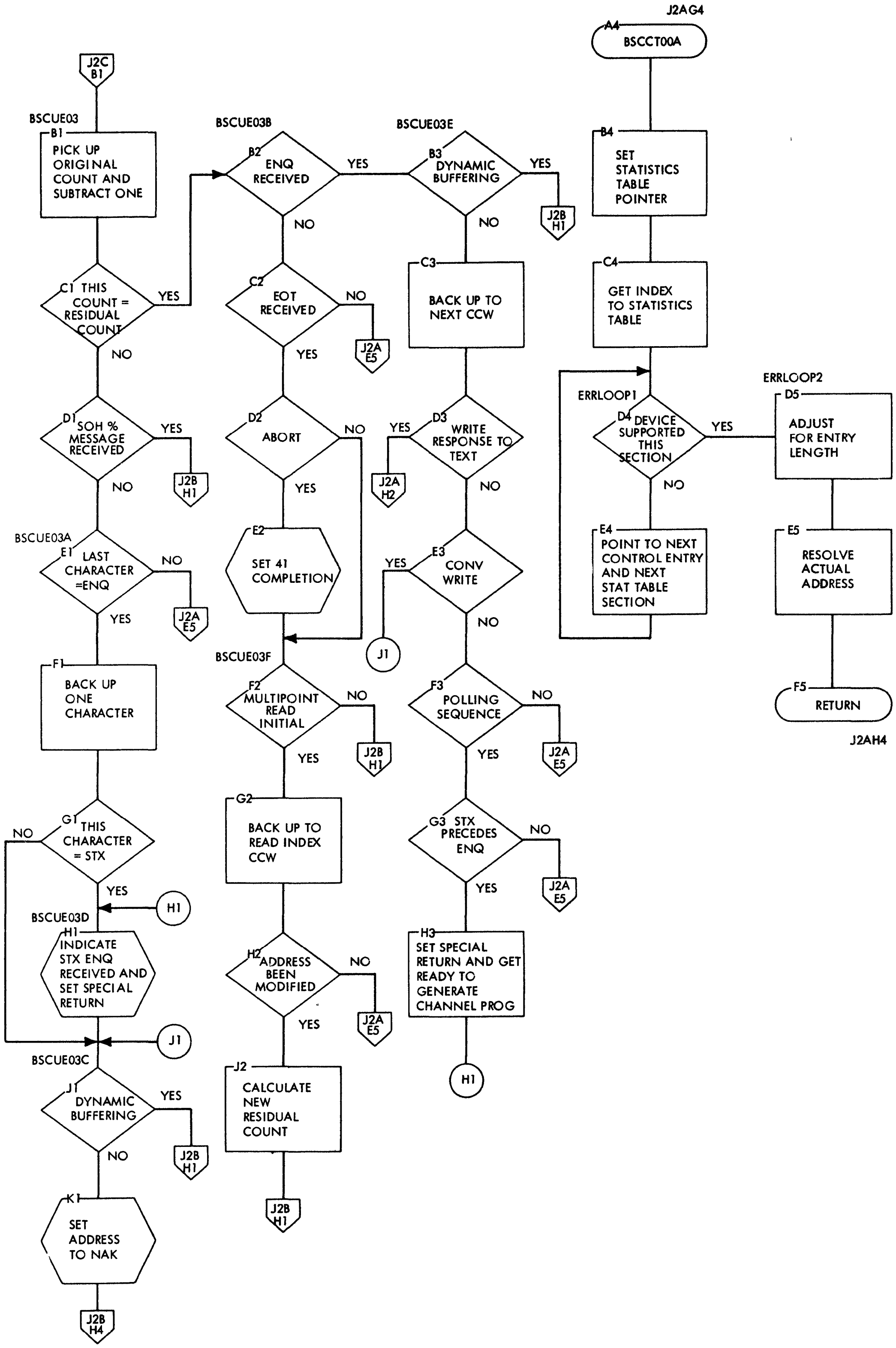

J2C B1
BSCUE03
B1 PICK UP ORIGINAL COUNT AND SUBTRACT ONE
C1 THIS COUNT = RESIDUAL COUNT
YES
NO
D1 SOH % MESSAGE RECEIVED
J2B H1
BSCUE03A
E1 LAST CHARACTER =ENQ
J2A E5
F1 BACK UP ONE CHARACTER
G1 THIS CHARACTER = STX
BSCUE03D
H1 INDICATE STX ENQ RECEIVED AND SET SPECIAL RETURN
BSCUE03C
J1 DYNAMIC BUFFERING
K1 SET ADDRESS TO NAK
J2B H4
BSCUE03B
B2 ENQ RECEIVED
C2 EOT RECEIVED
D2 ABORT
E2 SET 41 COMPLETION
BSCUE03F
F2 MULTIPOINT READ INITIAL
G2 BACK UP TO READ INDEX CCW
H2 ADDRESS BEEN MODIFIED
J2 CALCULATE NEW RESIDUAL COUNT
BSCUE03E
B3 DYNAMIC BUFFERING
C3 BACK UP TO NEXT CCW
D3 WRITE RESPONSE TO TEXT
J2A H2
E3 CONV WRITE
J1
F3 POLLING SEQUENCE
G3 STX PRECEDES ENQ
H3 SET SPECIAL RETURN AND GET READY TO GENERATE CHANNEL PROG
H1
J2AG4
A4 BSCCT00A
B4 SET STATISTICS TABLE POINTER
C4 GET INDEX TO STATISTICS TABLE
ERRLOOP1
D4 DEVICE SUPPORTED THIS SECTION
E4 POINT TO NEXT CONTROL ENTRY AND NEXT STAT TABLE SECTION
ERRLOOP2
D5 ADJUST FOR ENTRY LENGTH
E5 RESOLVE ACTUAL ADDRESS
F5 RETURN
J2AH4

Chart K1. Change Entry (IECTCHGN) (Part 1 of 2)

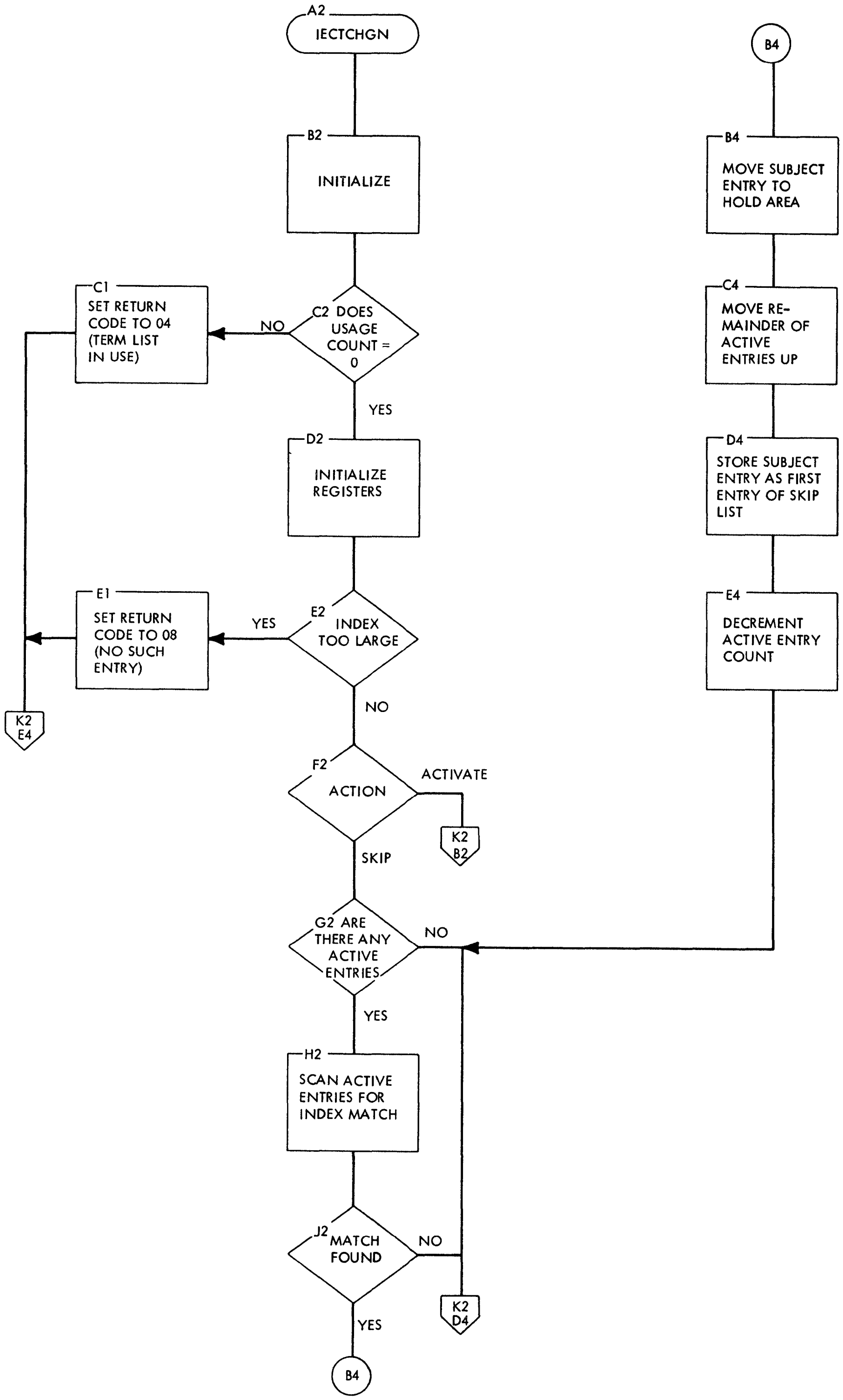


Chart K2. Change Entry (IECTCHGN) (Part 2 of 2)

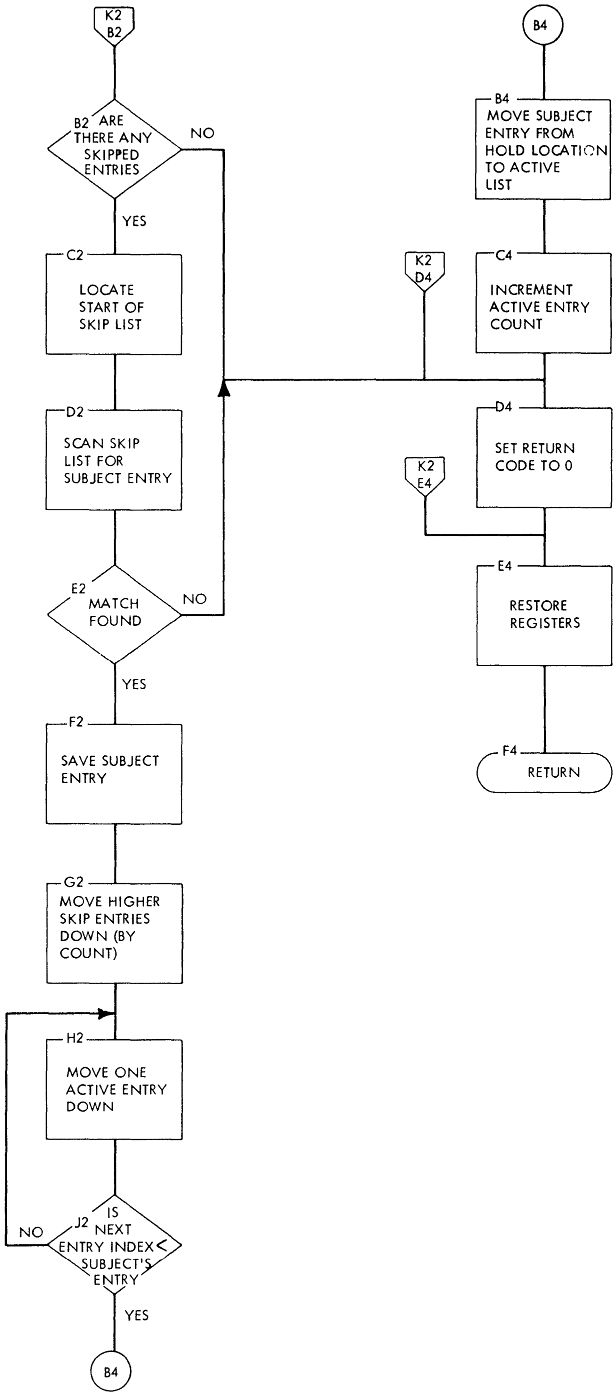
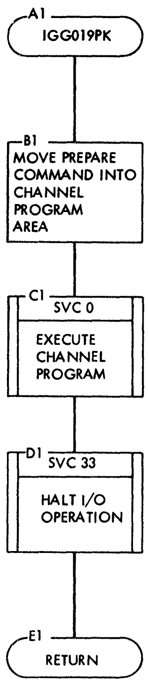
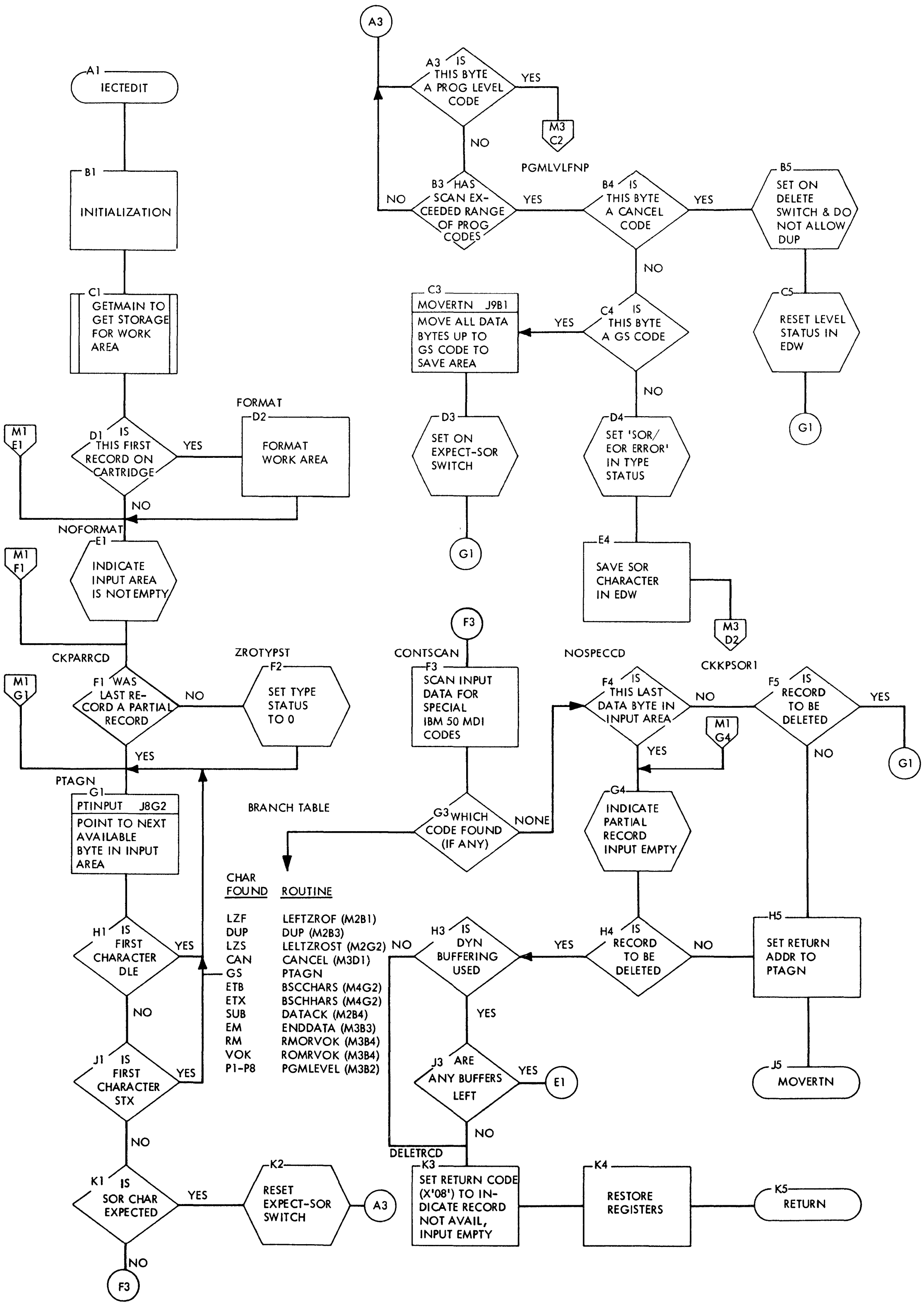


Chart L1. 2741 Break (IGG019PK)





A1 IECTEDIT
B1 INITIALIZATION
C1 GETMAIN TO GET STORAGE FOR WORK AREA
D1 IS THIS FIRST RECORD ON CARTRIDGE
YES
NO
FORMAT
D2 FORMAT WORK AREA
M1 E1
NOFORMAT
E1 INDICATE INPUT AREA IS NOT EMPTY
M1 F1
CKPARRCD
F1 WAS LAST RECORD A PARTIAL RECORD
ZROTYPST
F2 SET TYPE STATUS TO 0
M1 G1
PTAGN
G1 PTINPUT J8G2
POINT TO NEXT AVAILABLE BYTE IN INPUT AREA
H1 IS FIRST CHARACTER DLE
J1 IS FIRST CHARACTER STX
K1 IS SOR CHAR EXPECTED
K2 RESET EXPECT-SOR SWITCH
A3
F3
A3 IS THIS BYTE A PROG LEVEL CODE
M3 C2
PGMLVLFNP
B3 HAS SCAN EXCEEDED RANGE OF PROG CODES
B4 IS THIS BYTE A CANCEL CODE
B5 SET ON DELETE SWITCH & DO NOT ALLOW DUP
C5 RESET LEVEL STATUS IN EDW
G1
C3 MOVERTN J9B1
MOVE ALL DATA BYTES UP TO GS CODE TO SAVE AREA
C4 IS THIS BYTE A GS CODE
D3 SET ON EXPECT-SOR SWITCH
D4 SET 'SOR/EOR ERROR' IN TYPE STATUS
E4 SAVE SOR CHARACTER IN EDW
M3 D2
CKKPSOR1
CONTSCAN
F3 SCAN INPUT DATA FOR SPECIAL IBM 50 MDI CODES
NOSPECCD
F4 IS THIS LAST DATA BYTE IN INPUT AREA
F5 IS RECORD TO BE DELETED
M1 G4
G4 INDICATE PARTIAL RECORD INPUT EMPTY
BRANCH TABLE
G3 WHICH CODE FOUND (IF ANY)
NONE
CHAR FOUND
ROUTINE
LZF LEFTZROF (M2B1)
DUP DUP (M2B3)
LZS LELTZROST (M2G2)
CAN CANCEL (M3D1)
GS PTAGN
ETB BSCCHARS (M4G2)
ETX BSCHHARS (M4G2)
SUB DATACK (M2B4)
EM ENDDATA (M3B3)
RM RMORVOK (M3B4)
VOK ROMRVOK (M3B4)
P1-P8 PGMLEVEL (M3B2)
H3 IS DYN BUFFERING USED
H4 IS RECORD TO BE DELETED
H5 SET RETURN ADDR TO PTAGN
J3 ARE ANY BUFFERS LEFT
E1
J5 MOVERTN
DELETRCD
K3 SET RETURN CODE (X'08') TO INDICATE RECORD NOT AVAIL, INPUT EMPTY
K4 RESTORE REGISTERS
K5 RETURN

Chart M2. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 2 of 5)

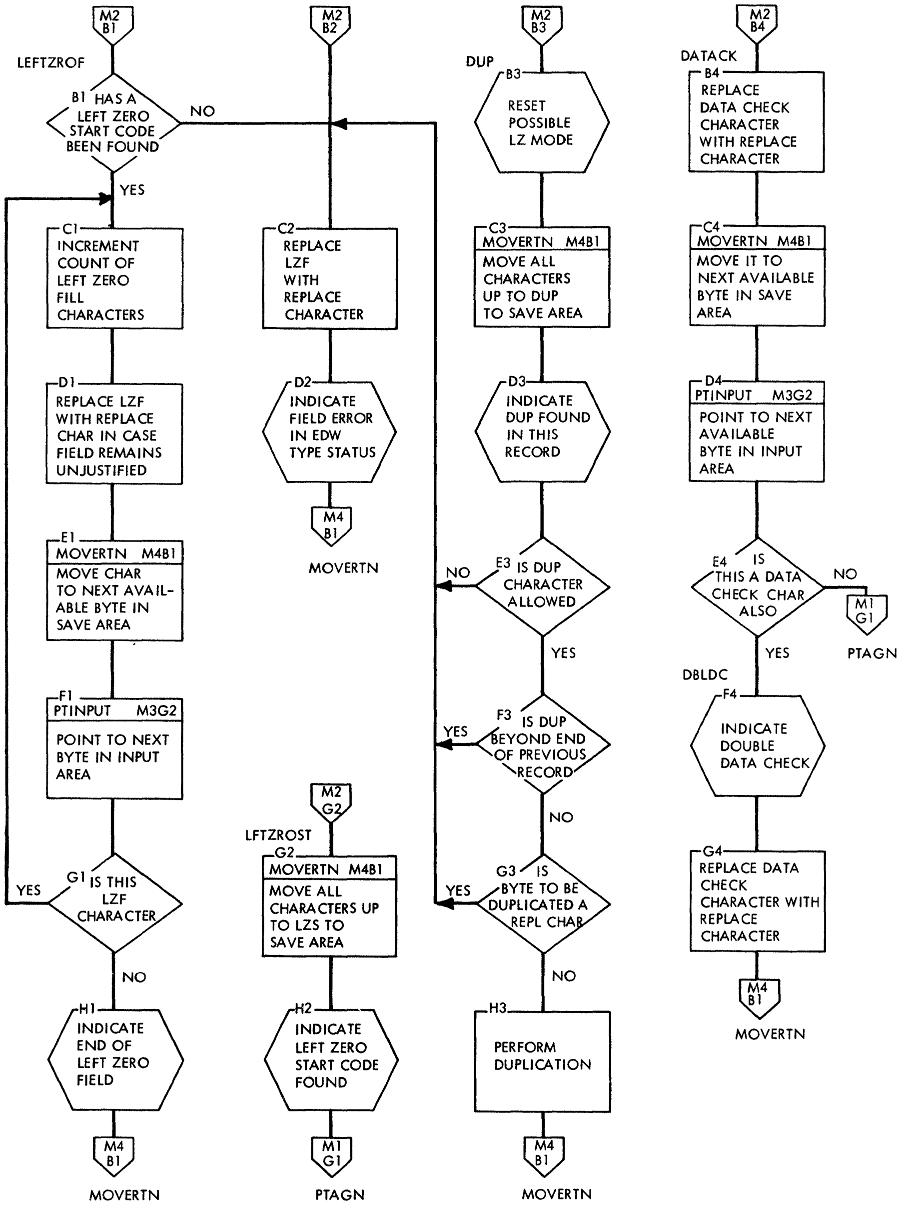


Chart M3. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 3 of 5)

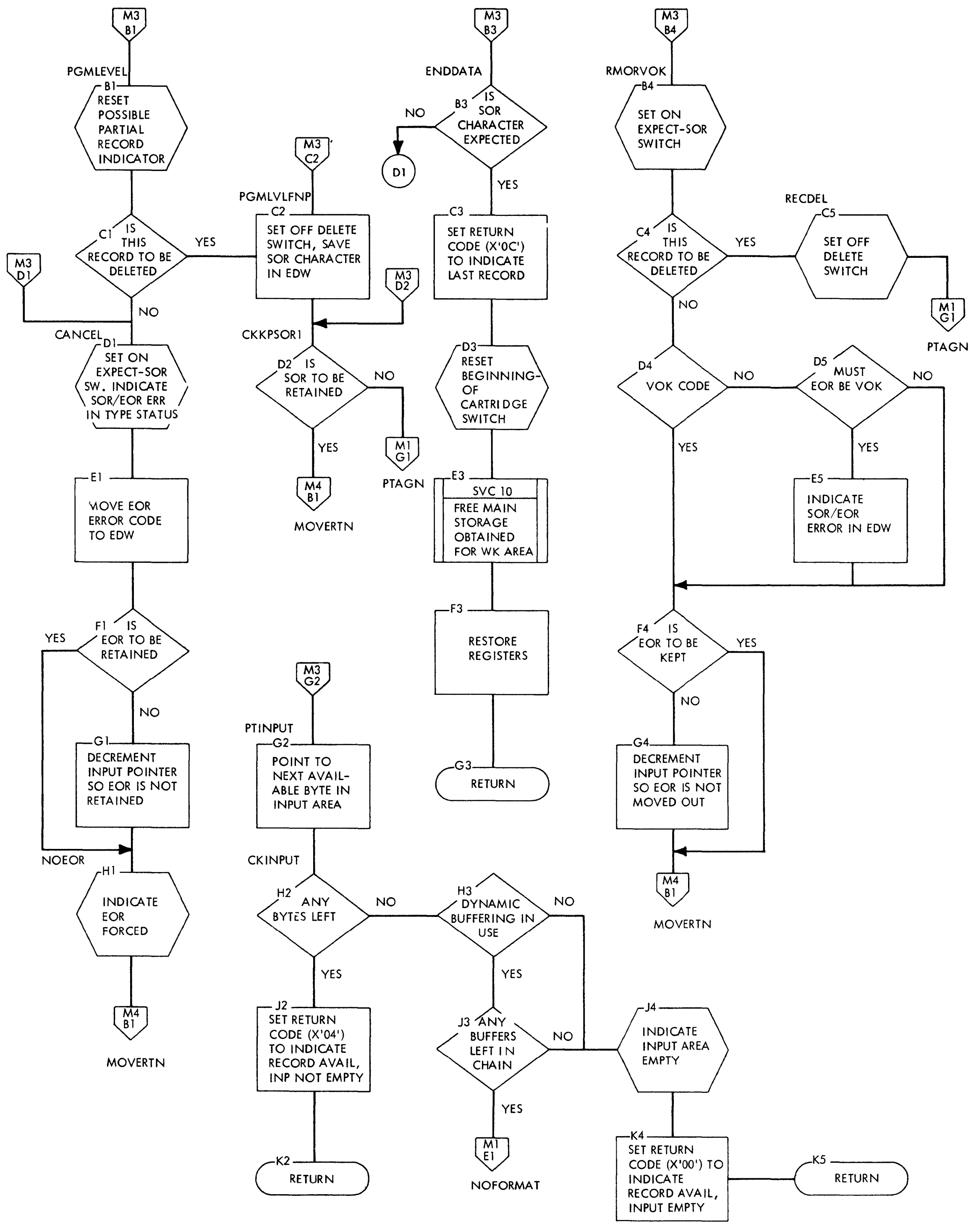


Chart M4. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 4 of 5)

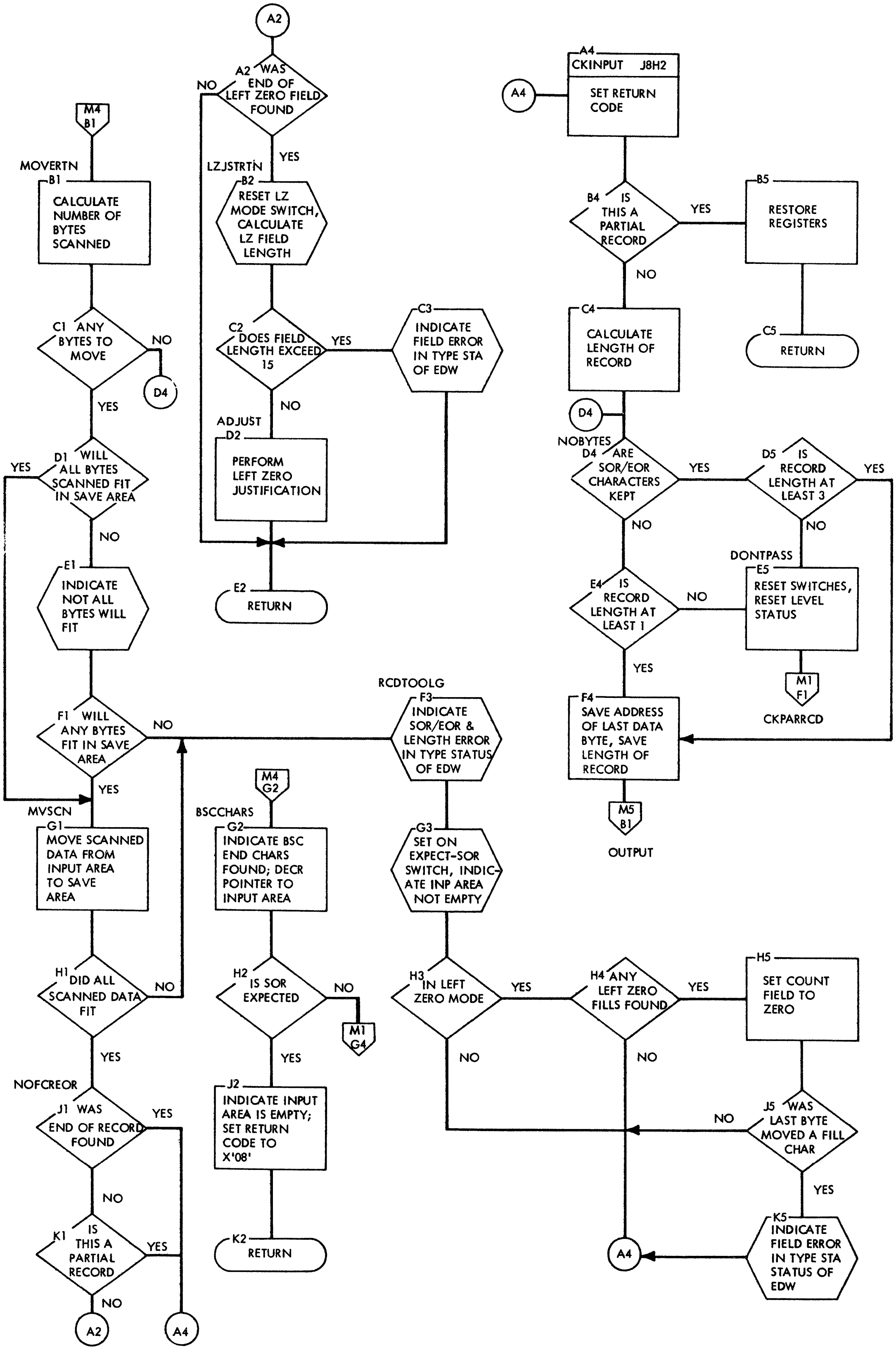


Chart M5. Edit (IBM 50 Magnetic Data Inscriber) (IECTEDIT) (Part 5 of 5)

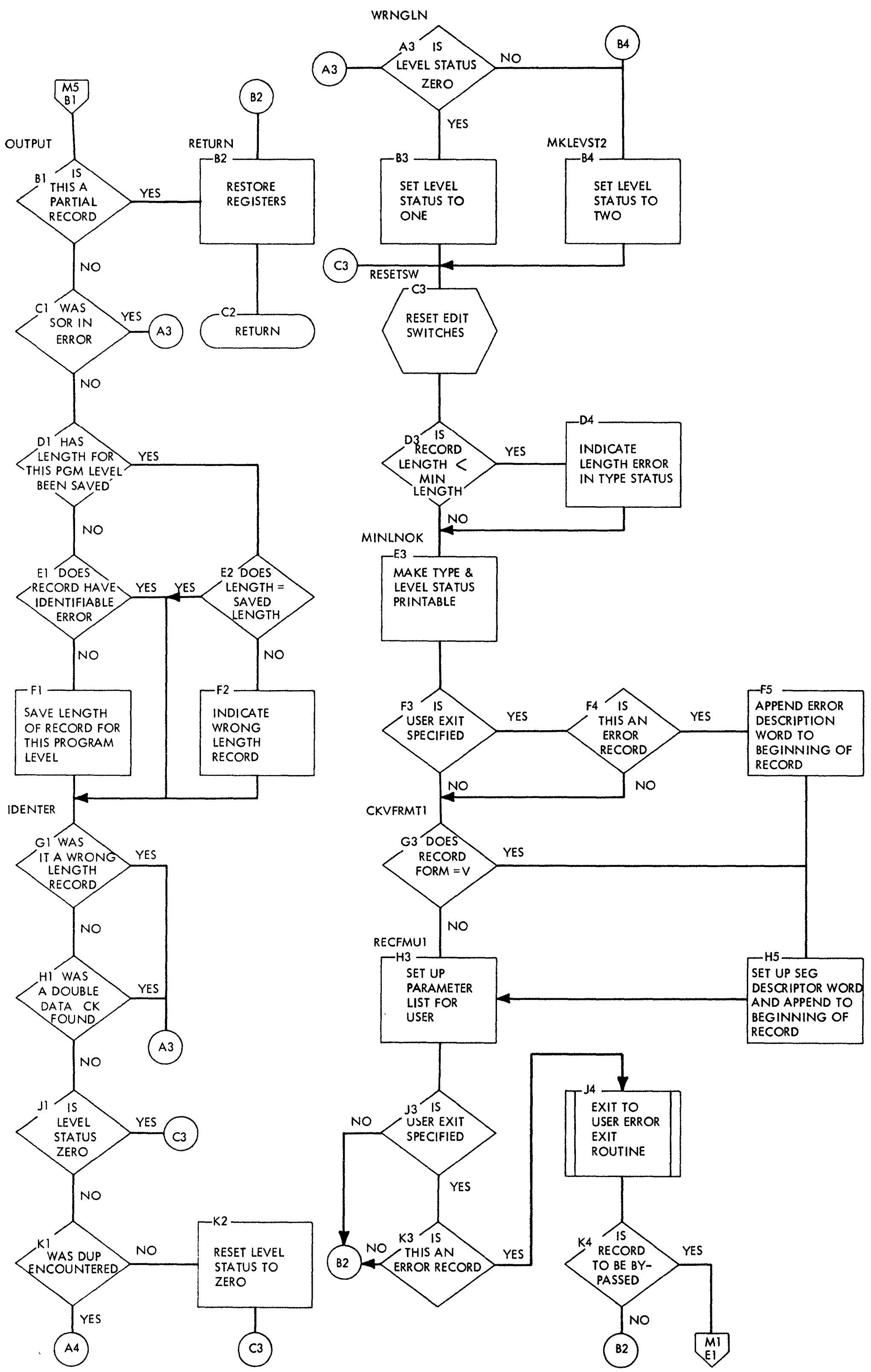


Chart TL. On-Line Test Request (IECTONLT)

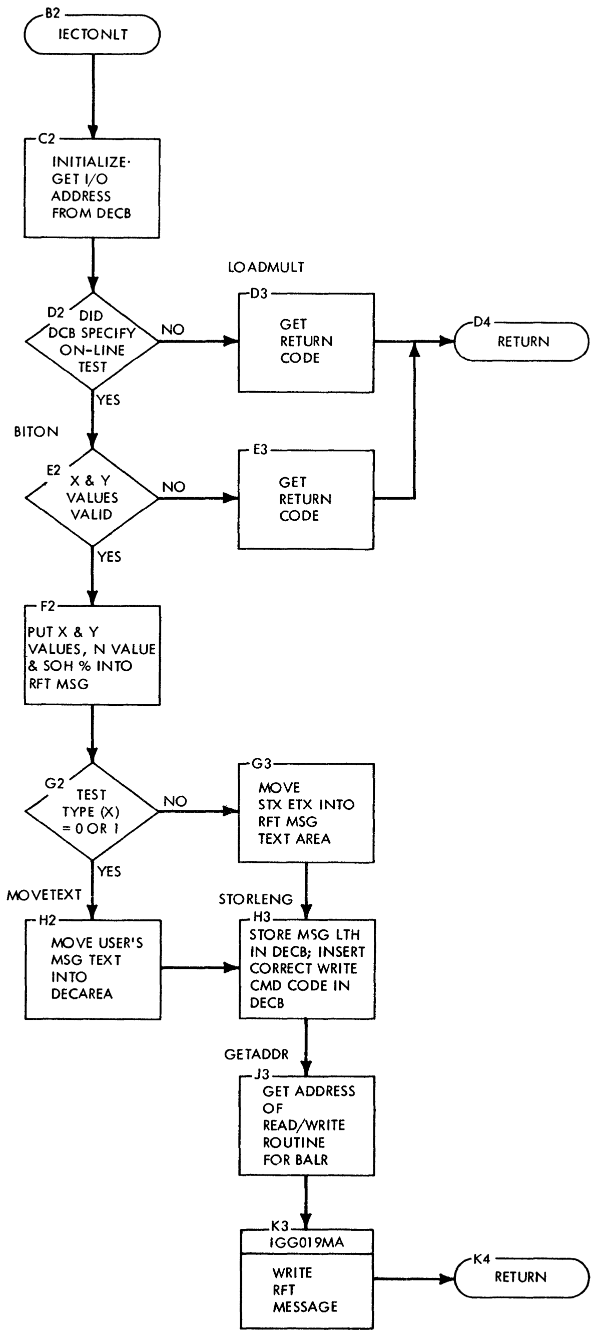


Chart T0. On-Line Test Control (Start/Stop) (IGC0006F)

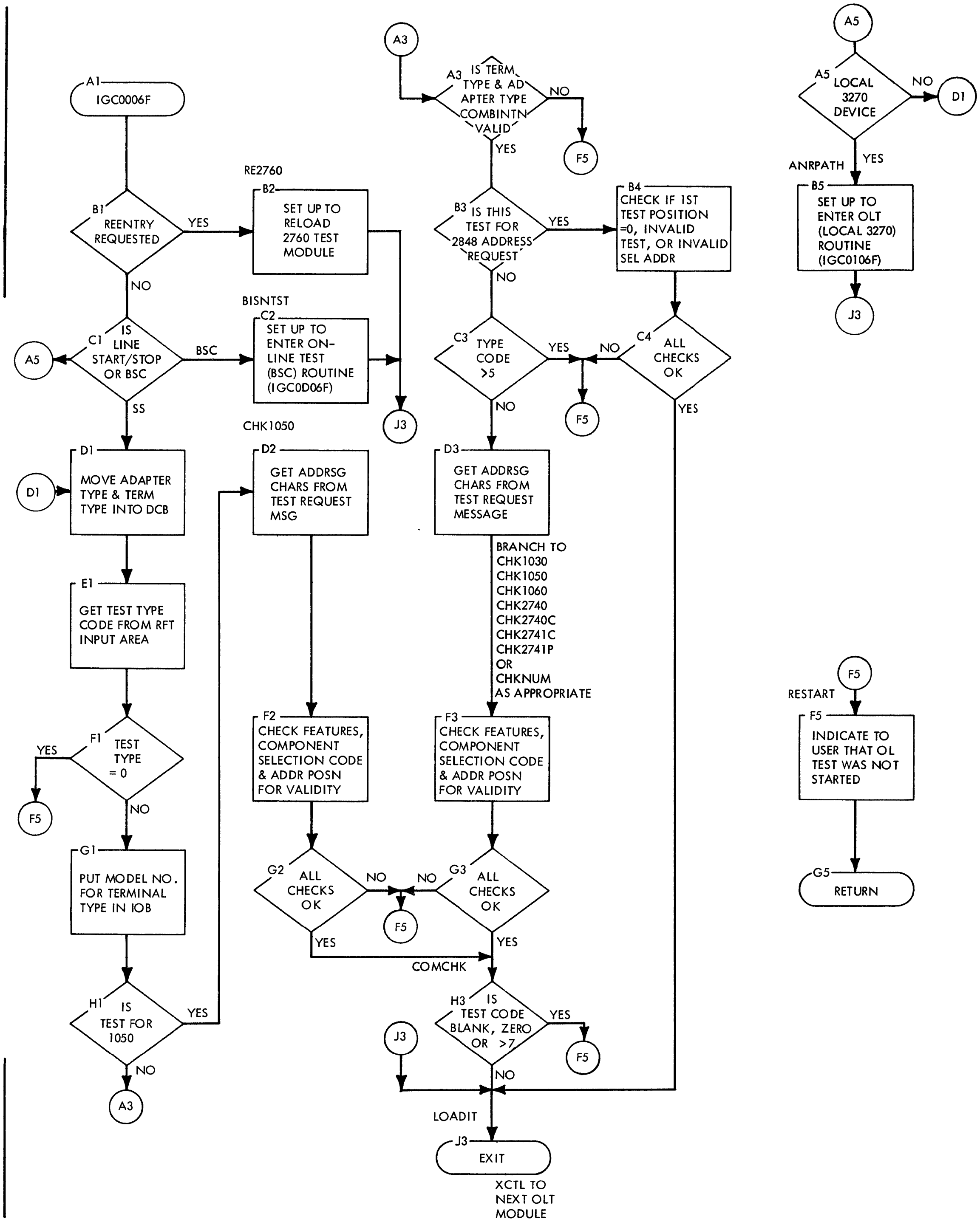


Chart T01. On-Line Test Control (BSC) (IGC0D06F)

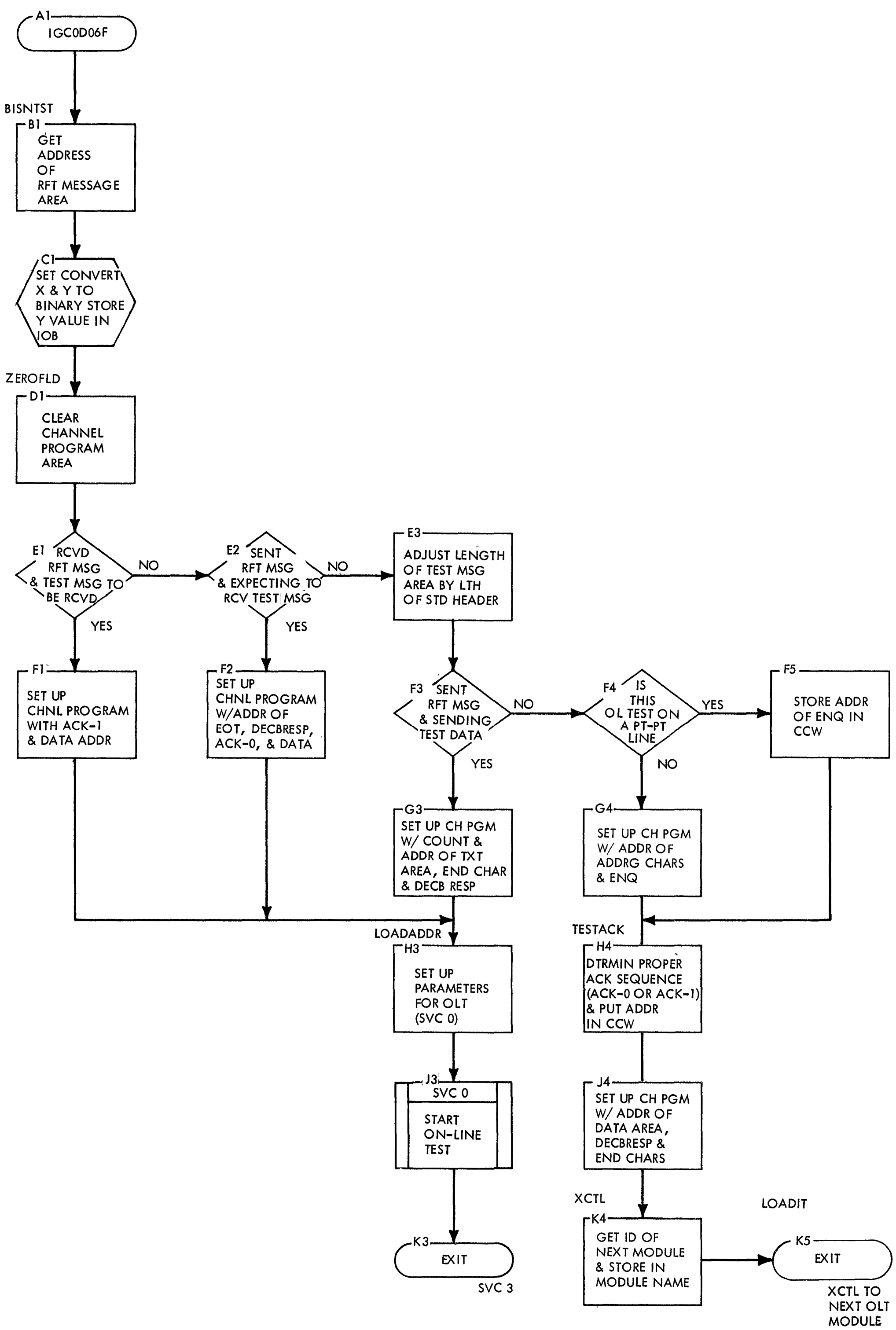


Chart T02. On-Line Test Contrcl (Local 3270) (IGC1206F)

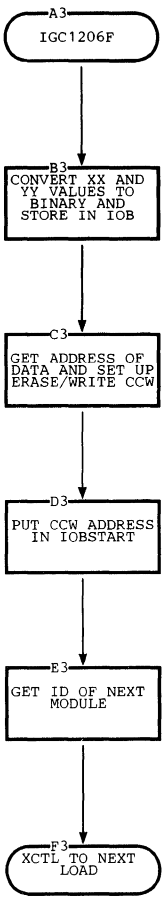
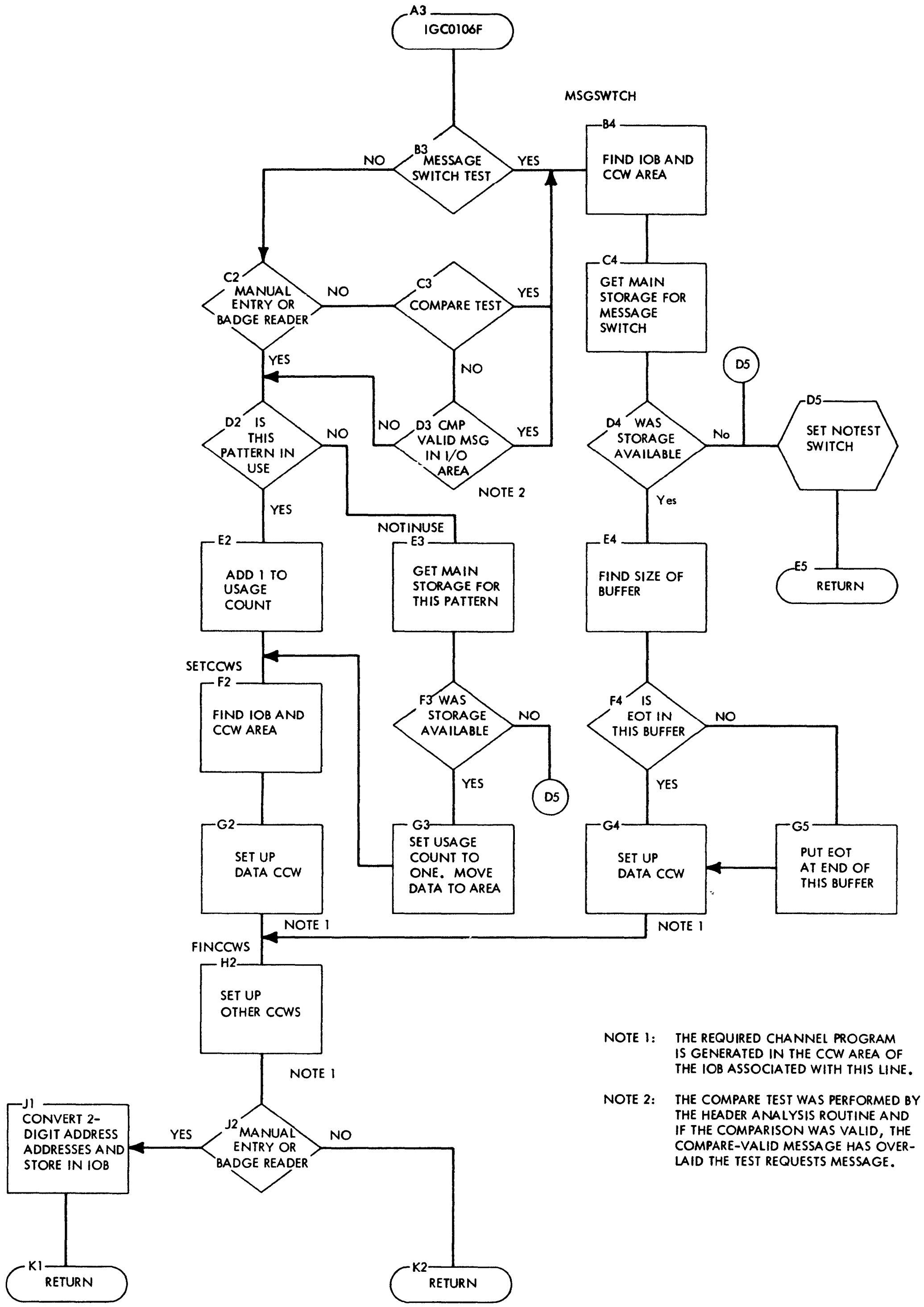


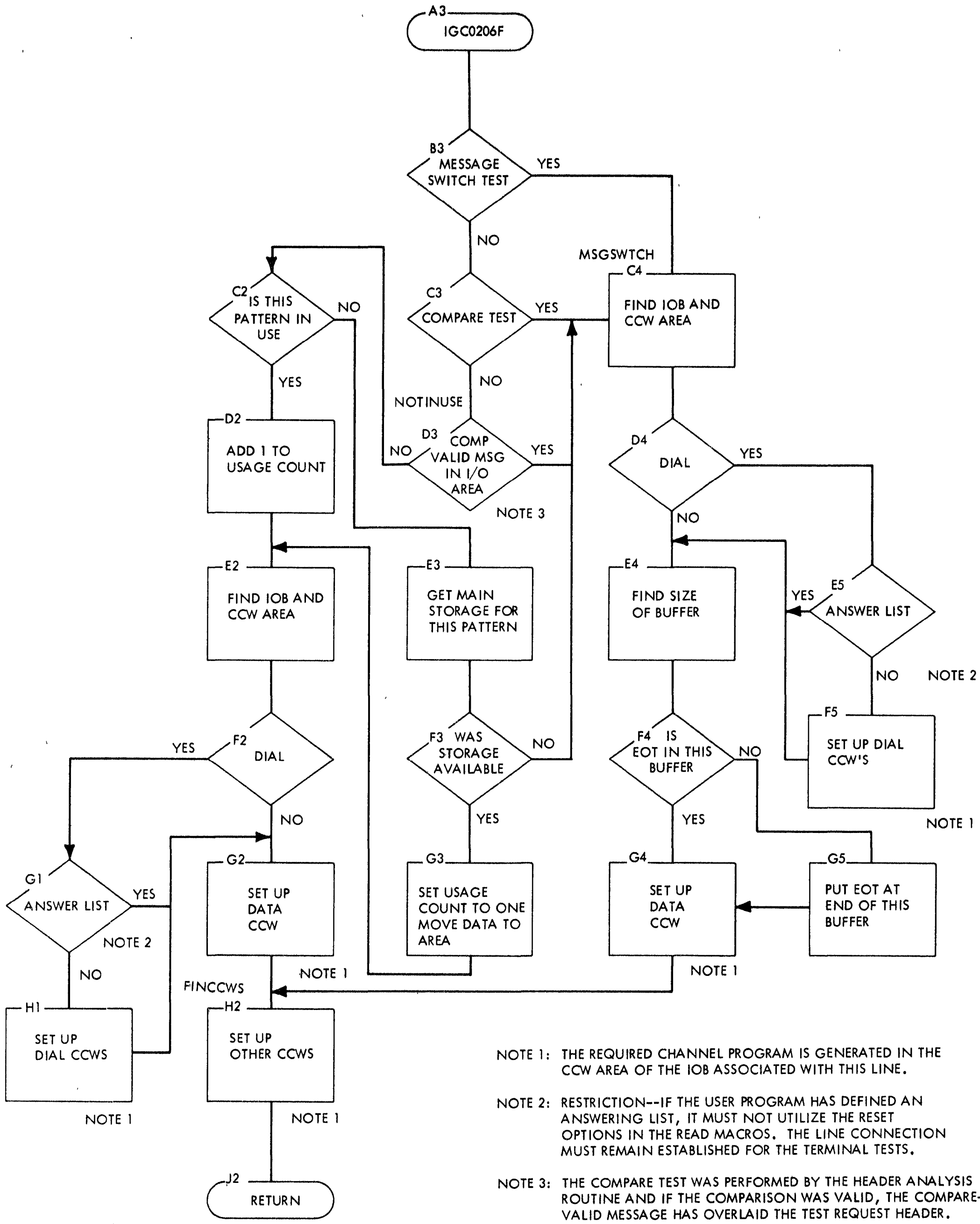
Chart T1. IBM 1030 Terminal Test (IGC0106F)



NOTE 1: THE REQUIRED CHANNEL PROGRAM IS GENERATED IN THE CCW AREA OF THE IOB ASSOCIATED WITH THIS LINE.

NOTE 2: THE COMPARE TEST WAS PERFORMED BY THE HEADER ANALYSIS ROUTINE AND IF THE COMPARISON WAS VALID, THE COMPARE-VALID MESSAGE HAS OVERLAID THE TEST REQUESTS MESSAGE.

Chart T2. IBM 1050 Terminal Test (IGC0206F)

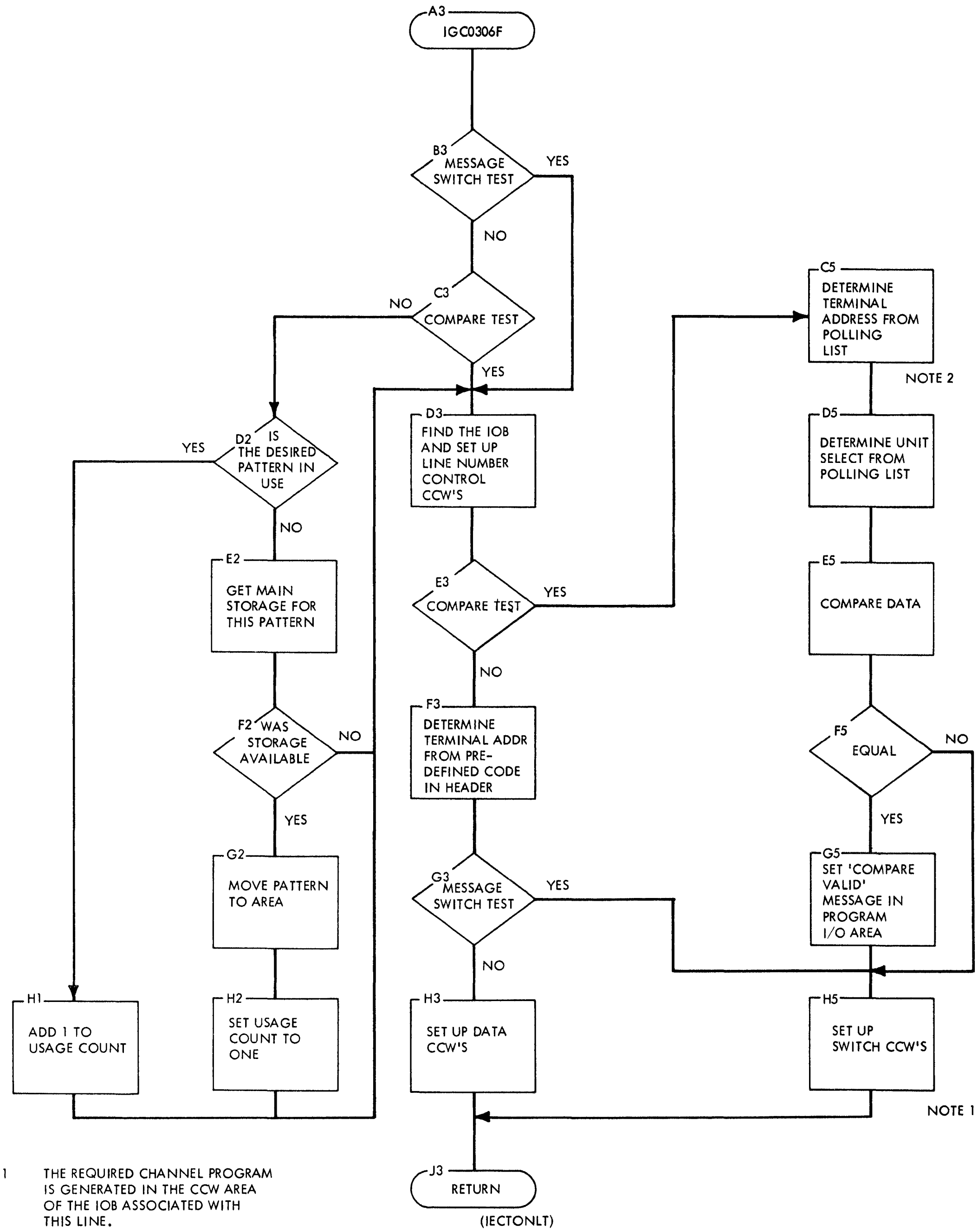


NOTE 1: THE REQUIRED CHANNEL PROGRAM IS GENERATED IN THE CCW AREA OF THE IOB ASSOCIATED WITH THIS LINE.

NOTE 2: RESTRICTION--IF THE USER PROGRAM HAS DEFINED AN ANSWERING LIST, IT MUST NOT UTILIZE THE RESET OPTIONS IN THE READ MACROS. THE LINE CONNECTION MUST REMAIN ESTABLISHED FOR THE TERMINAL TESTS.

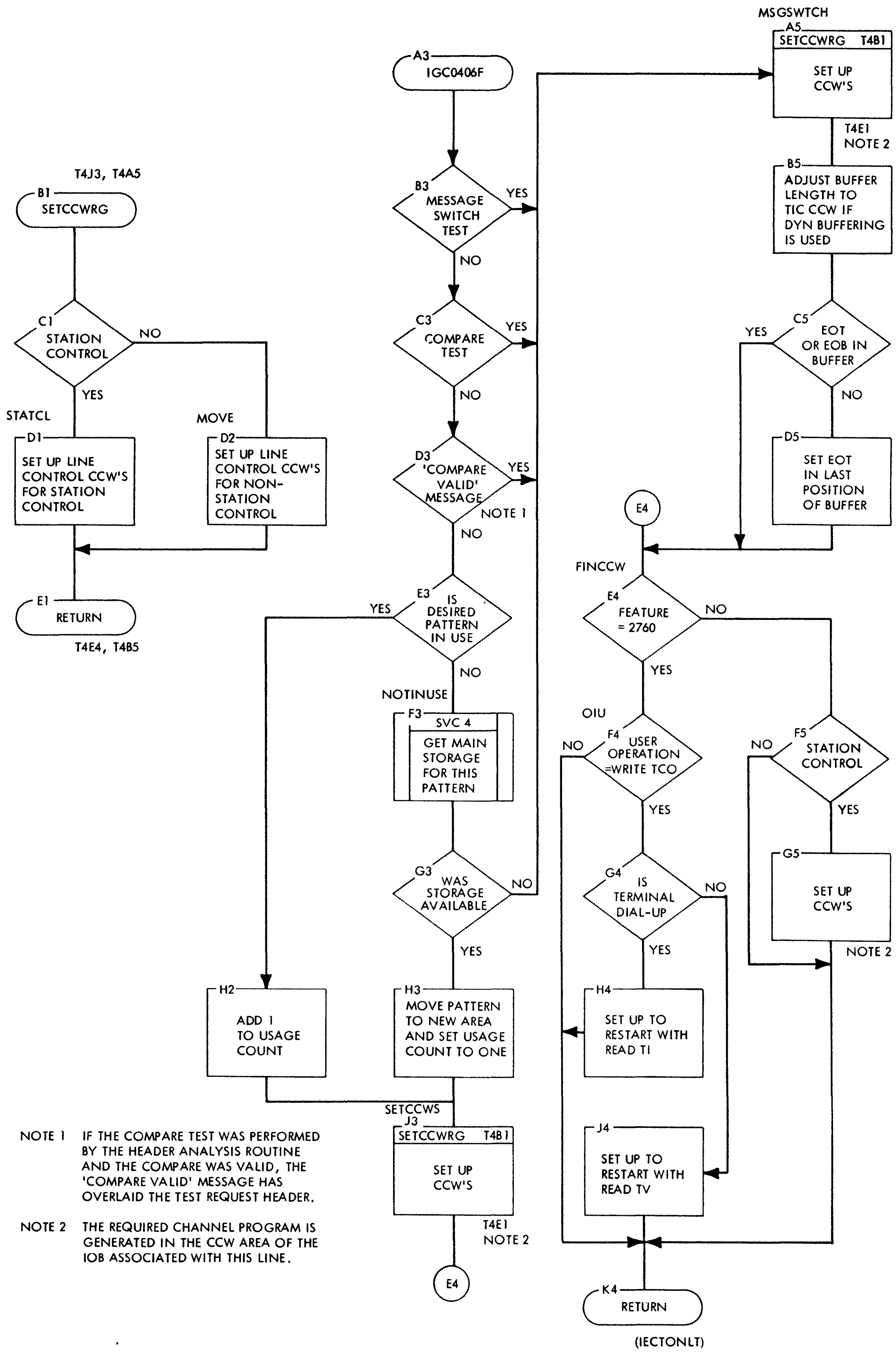
NOTE 3: THE COMPARE TEST WAS PERFORMED BY THE HEADER ANALYSIS ROUTINE AND IF THE COMPARISON WAS VALID, THE COMPARE-VALID MESSAGE HAS OVERLAID THE TEST REQUEST HEADER.

Chart T3. IBM 1060 Terminal Test (IGC0306F)



NOTE 1 THE REQUIRED CHANNEL PROGRAM IS GENERATED IN THE CCW AREA OF THE IOB ASSOCIATED WITH THIS LINE.

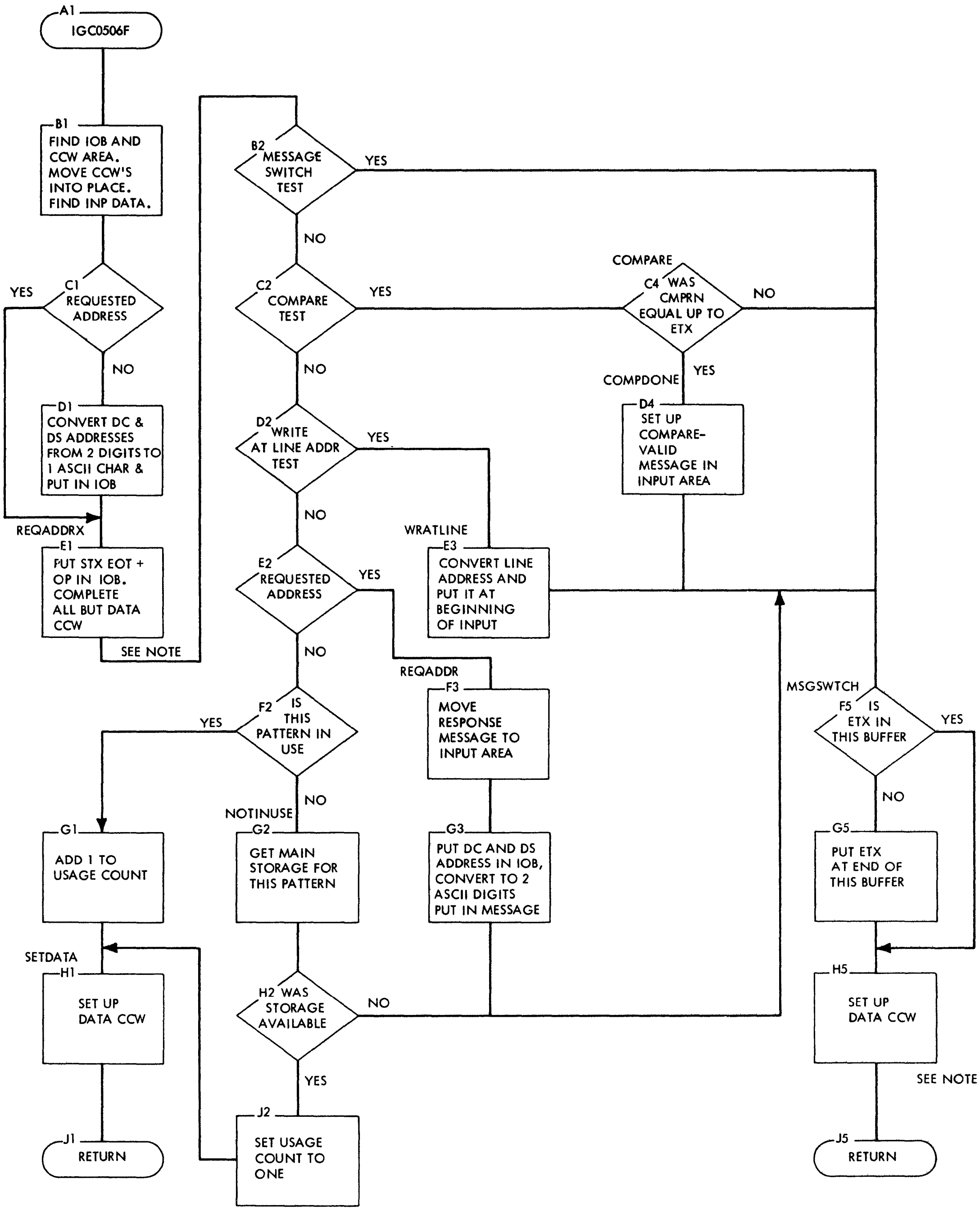
NOTE 2 COMPARE TEST SENDS RESPONSES TO ONLY THE REQUESTING TERMINAL.



NOTE 1 IF THE COMPARE TEST WAS PERFORMED BY THE HEADER ANALYSIS ROUTINE AND THE COMPARE WAS VALID, THE 'COMPARE VALID' MESSAGE HAS OVERLAID THE TEST REQUEST HEADER.

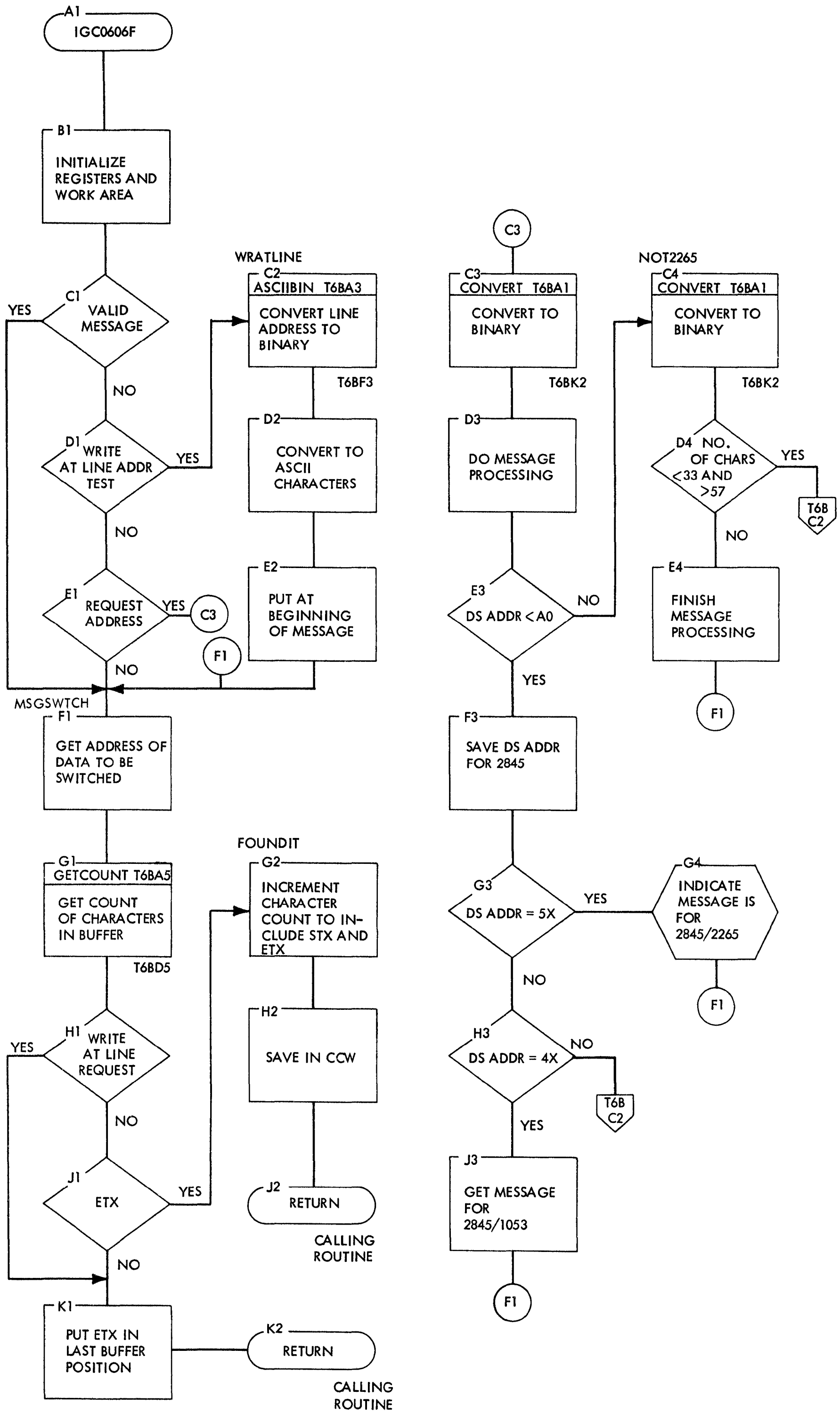
NOTE 2 THE REQUIRED CHANNEL PROGRAM IS GENERATED IN THE CCW AREA OF THE IOB ASSOCIATED WITH THIS LINE.

Chart T5. IBM 2848/2260 Terminal Test (IGC0504F) (Load 1)



NOTE: THE CHANNEL PROGRAM IS GENERATED IN THE CCW AREA OF THE IOB ASSOCIATED WITH THIS LINE.

Chart T6A. IBM 2848/2260 Terminal Test (IGC0606F) (Load 2) (Part 1 of 2)



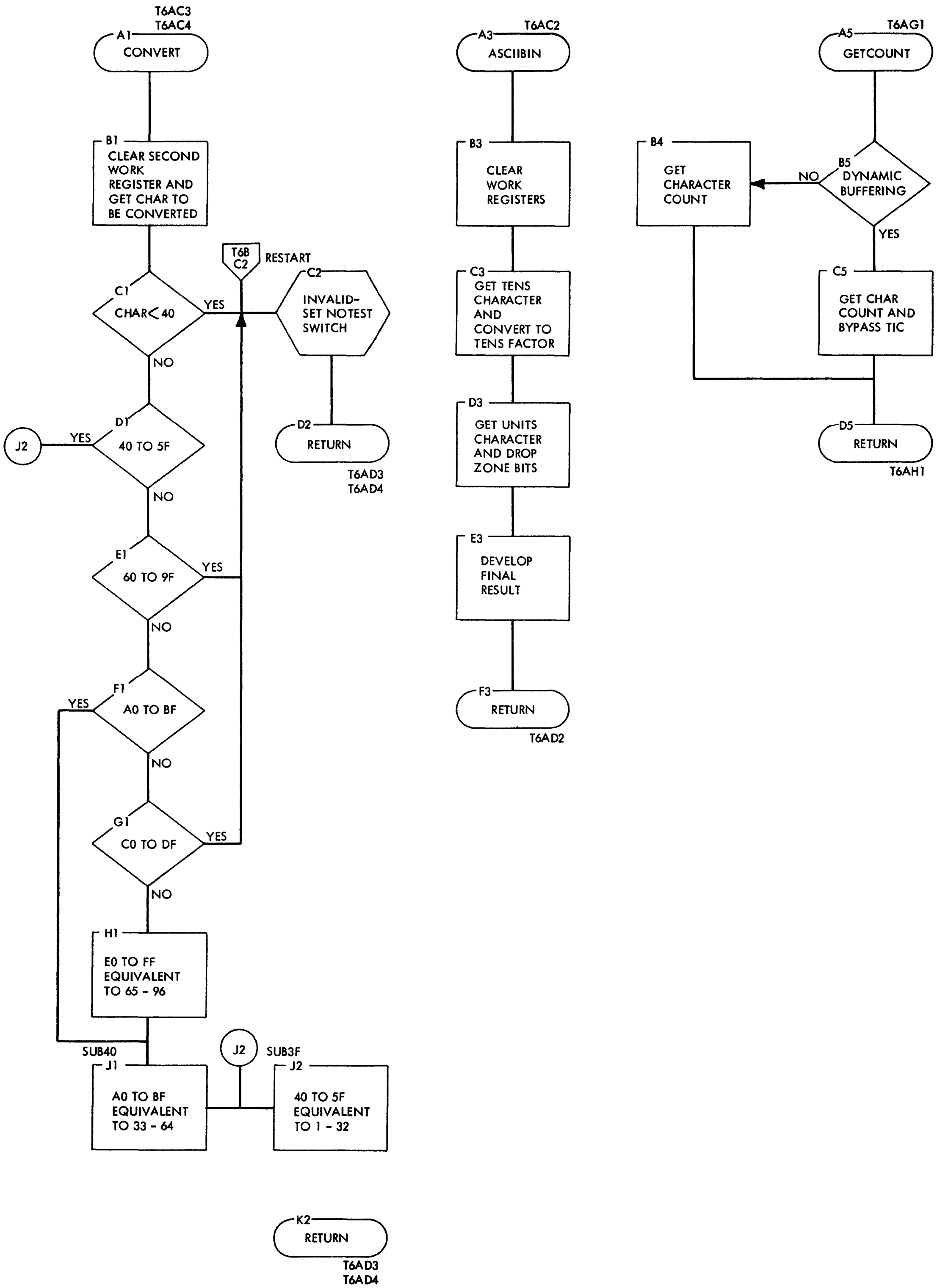

T6AC3
T6AC4
A1
CONVERT
B1
CLEAR SECOND WORK REGISTER AND GET CHAR TO BE CONVERTED
C1
CHAR< 40
YES
NO
T6B C2
RESTART
C2
INVALID-SET NOTEST SWITCH
D2
RETURN
T6AD3
T6AD4
J2
YES
D1
40 TO 5F
NO
E1
60 TO 9F
YES
NO
F1
A0 TO BF
YES
NO
G1
C0 TO DF
YES
NO
H1
E0 TO FF EQUIVALENT TO 65 - 96
SUB40
J1
A0 TO BF EQUIVALENT TO 33 - 64
J2
SUB3F
J2
40 TO 5F EQUIVALENT TO 1 - 32
K2
RETURN
T6AD3
T6AD4
T6AC2
A3
ASCIIBIN
B3
CLEAR WORK REGISTERS
C3
GET TENS CHARACTER AND CONVERT TO TENS FACTOR
D3
GET UNITS CHARACTER AND DROP ZONE BITS
E3
DEVELOP FINAL RESULT
F3
RETURN
T6AD2
T6AG1
A5
GETCOUNT
B4
GET CHARACTER COUNT
NO
B5
DYNAMIC BUFFERING
YES
C5
GET CHAR COUNT AND BYPASS TIC
D5
RETURN
T6AH1

Chart T7. IBM 2741 Terminal Test (Correspondence Code) (IGC0906F) (PTTC Code) (IGC0A06F)

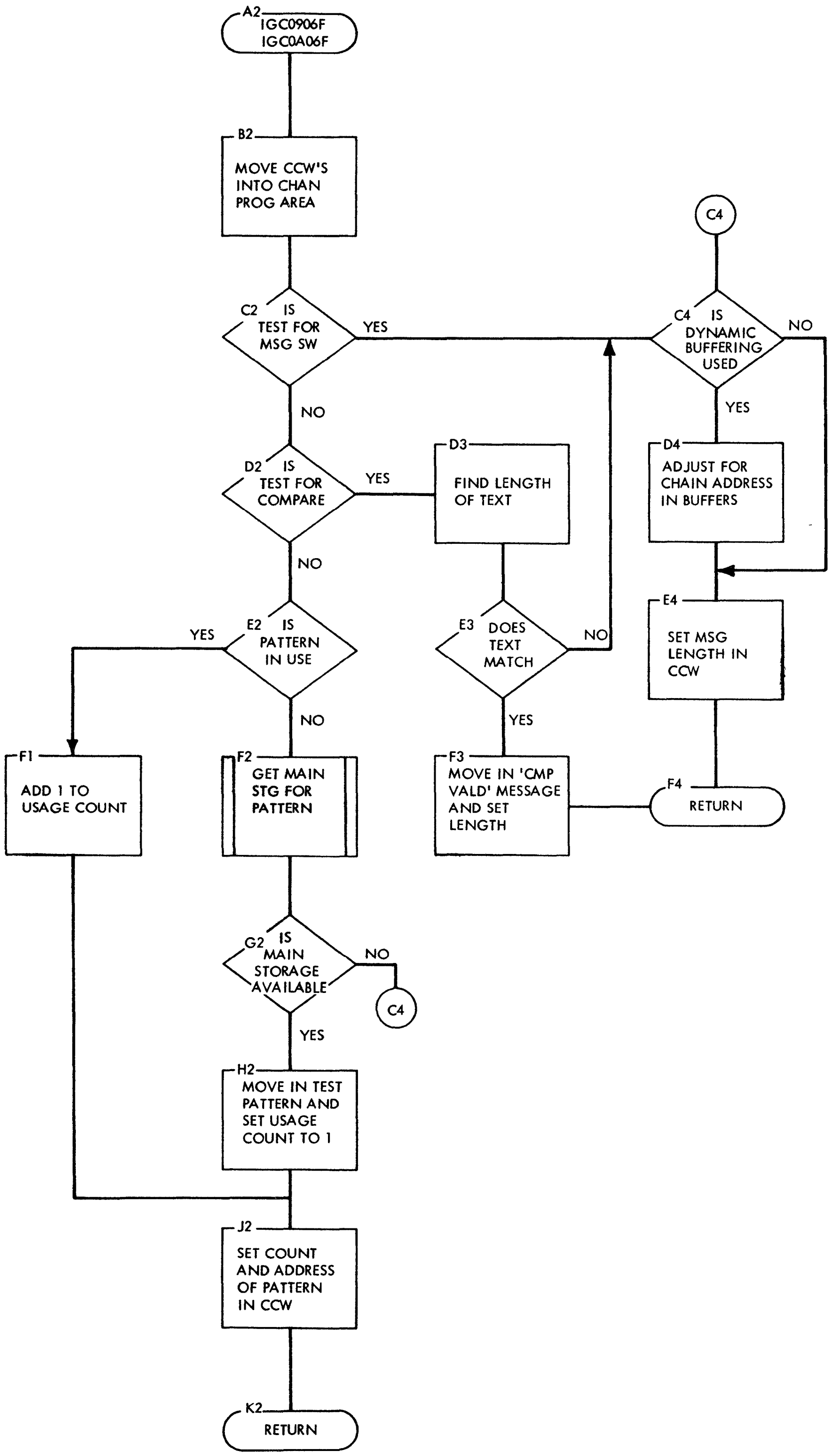


Chart T8. IBM 2760 Terminal Test (IGC0B06F)

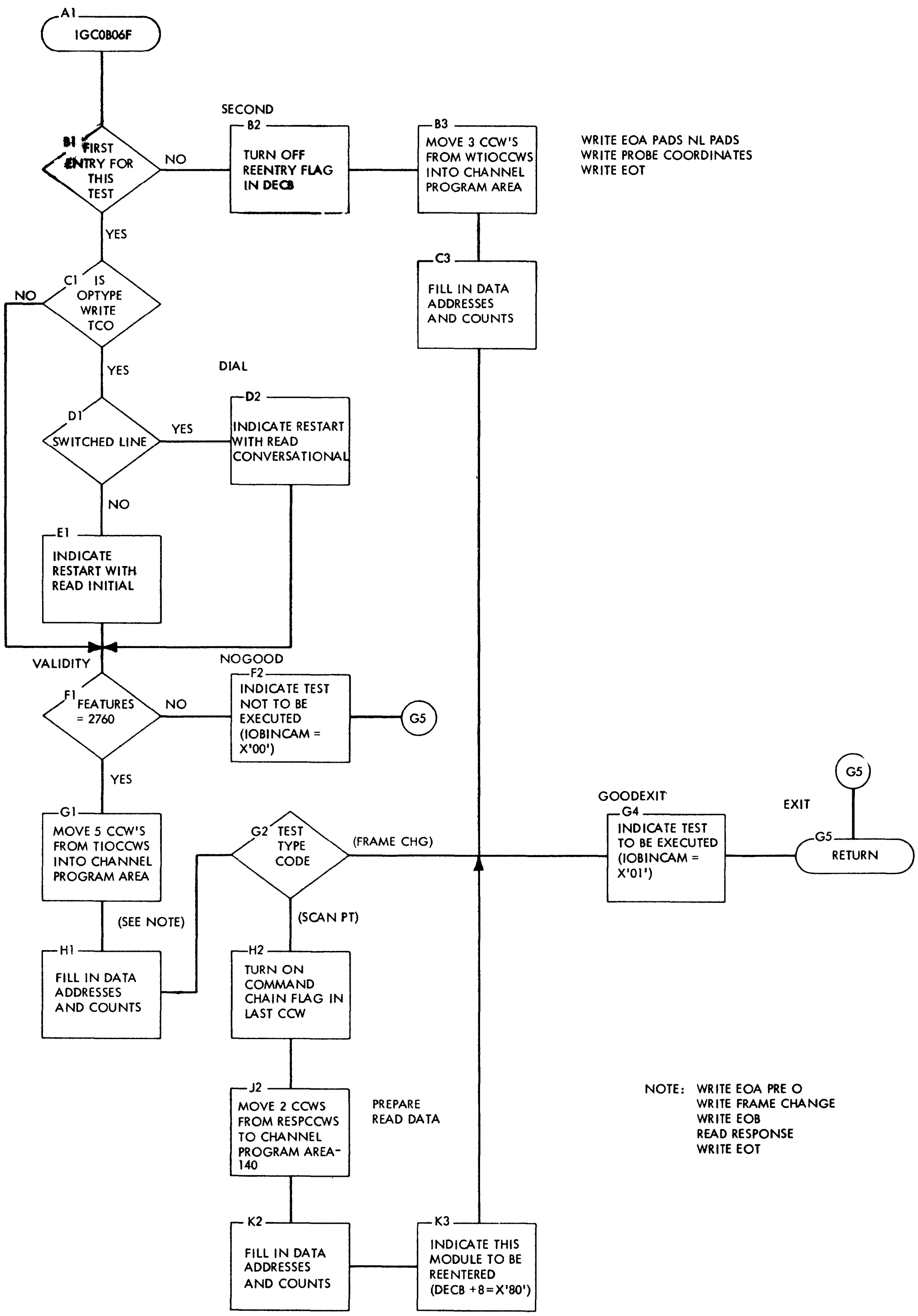


Chart T9. IBM 2740 Terminal Test (Correspondence Code) (IGC0C06F)

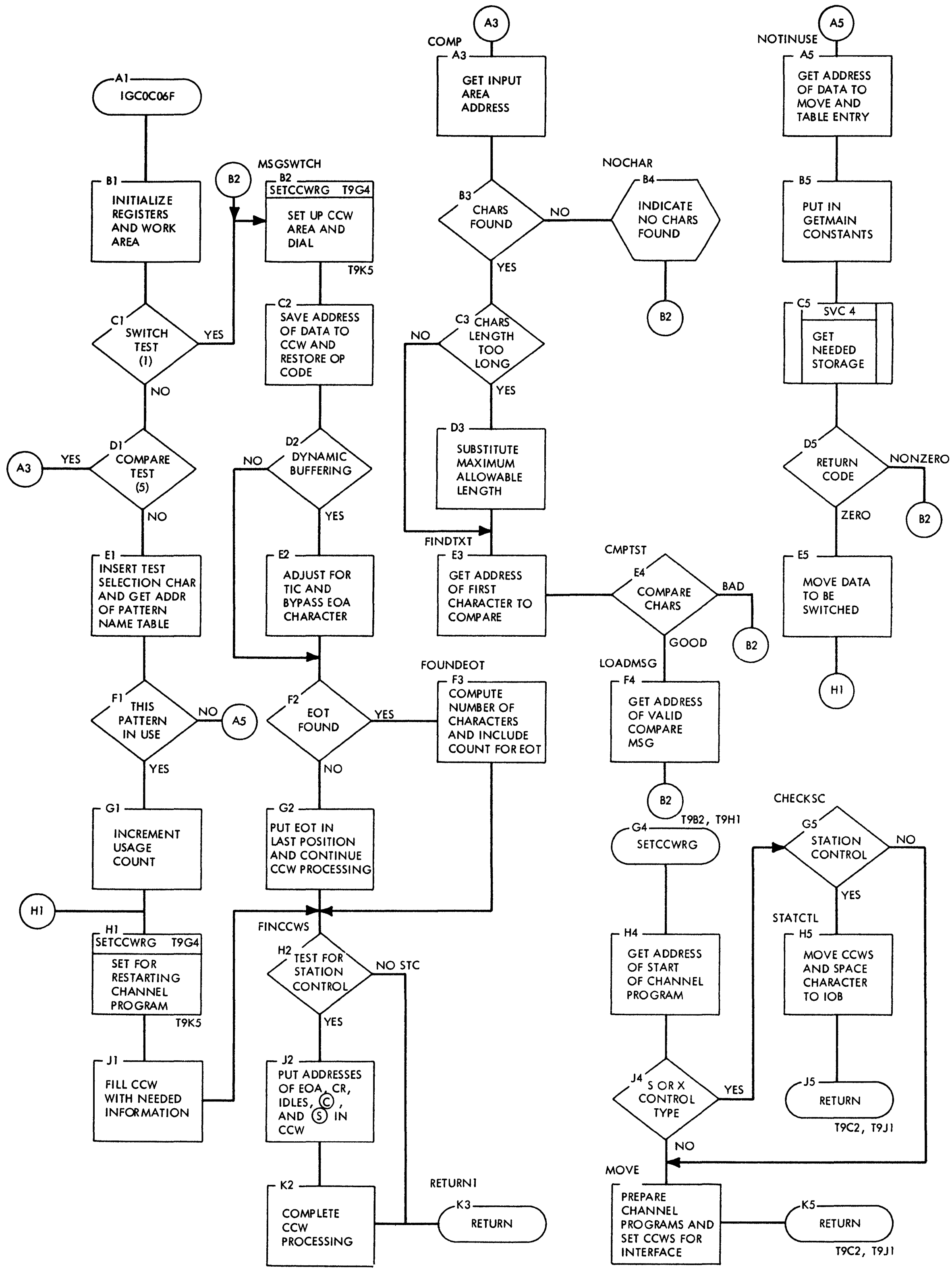


Chart T10. 3270 EBCDIC Test Module (Types 23 and 24) (IGC0E06F)

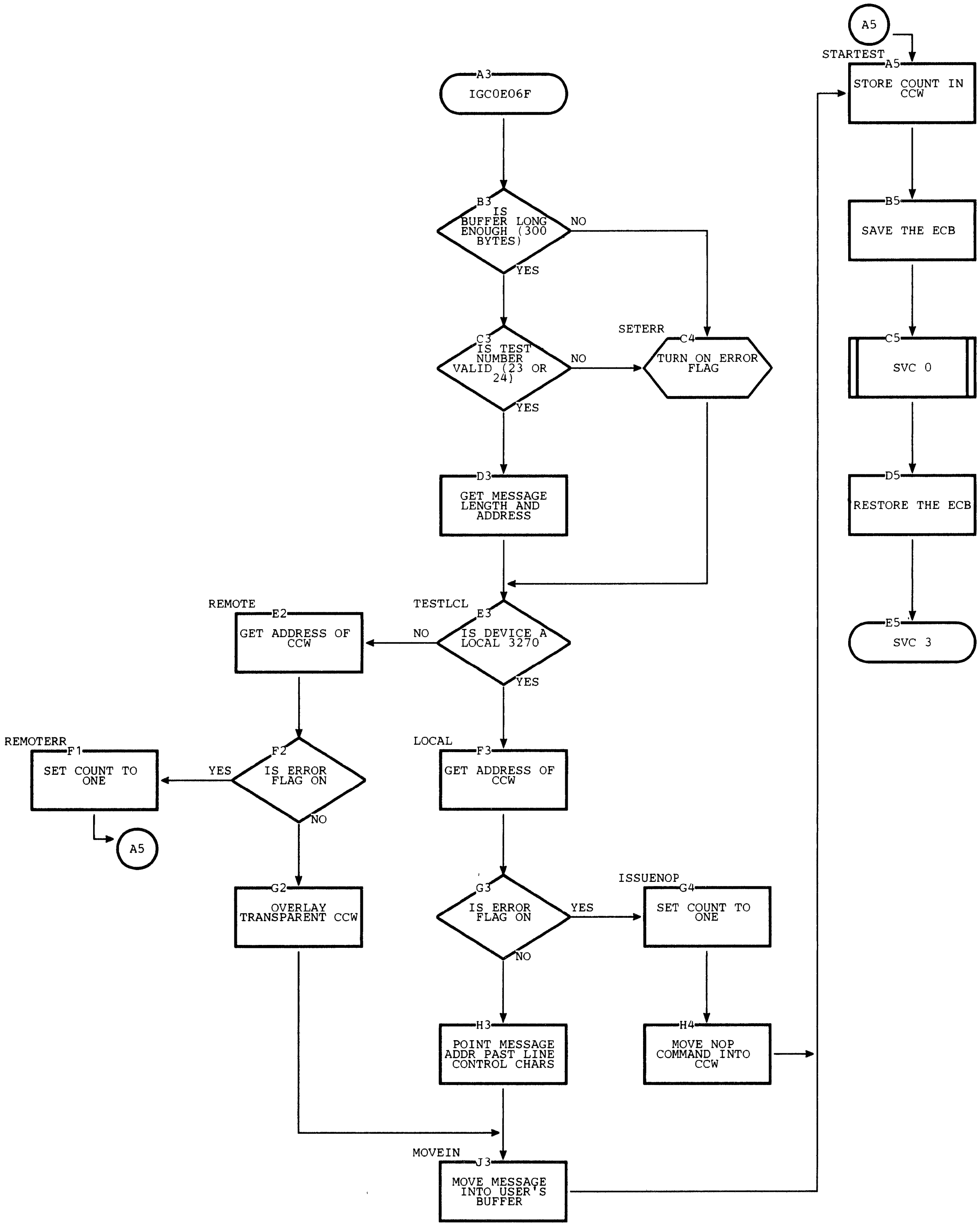


Chart T11. 3270 EBCDIC Test Module (Type 25) (IGC1406F)

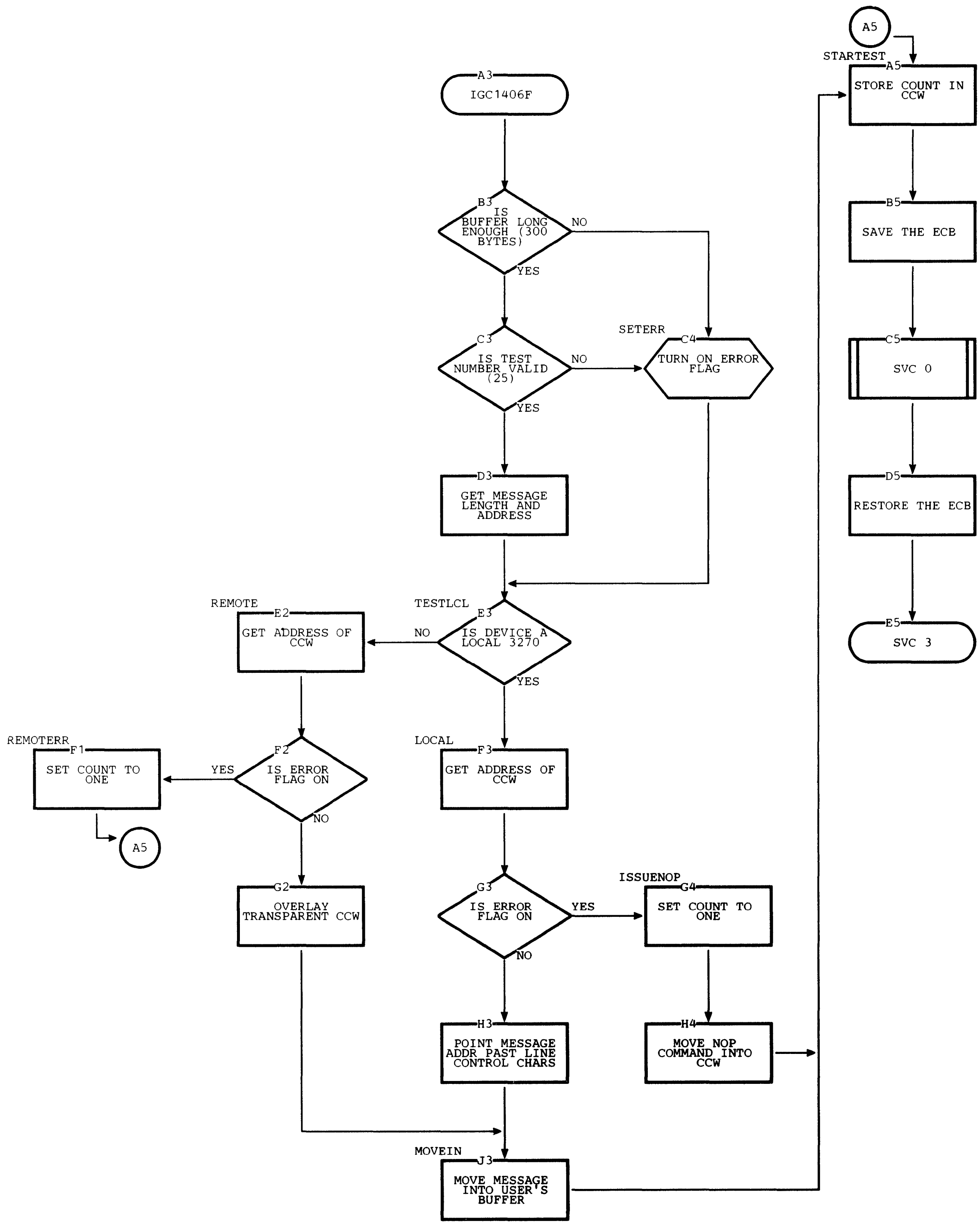


Chart T12. 3270 EBCDIC Test Module (Types 26 and 27) (IGC0F06F)

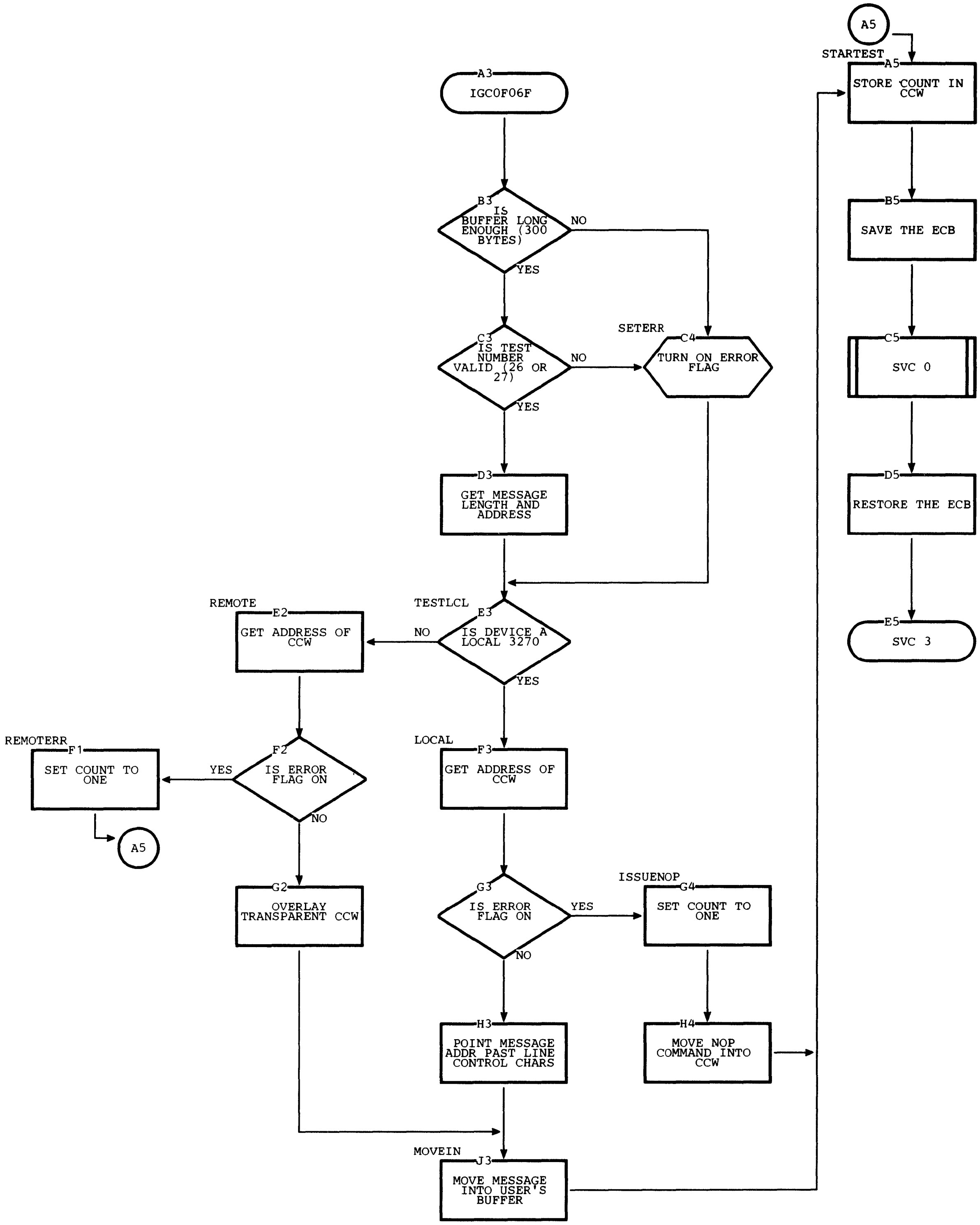


Chart T13. 3270 EBCDIC Test Module (Type 28) (IGC1306F)

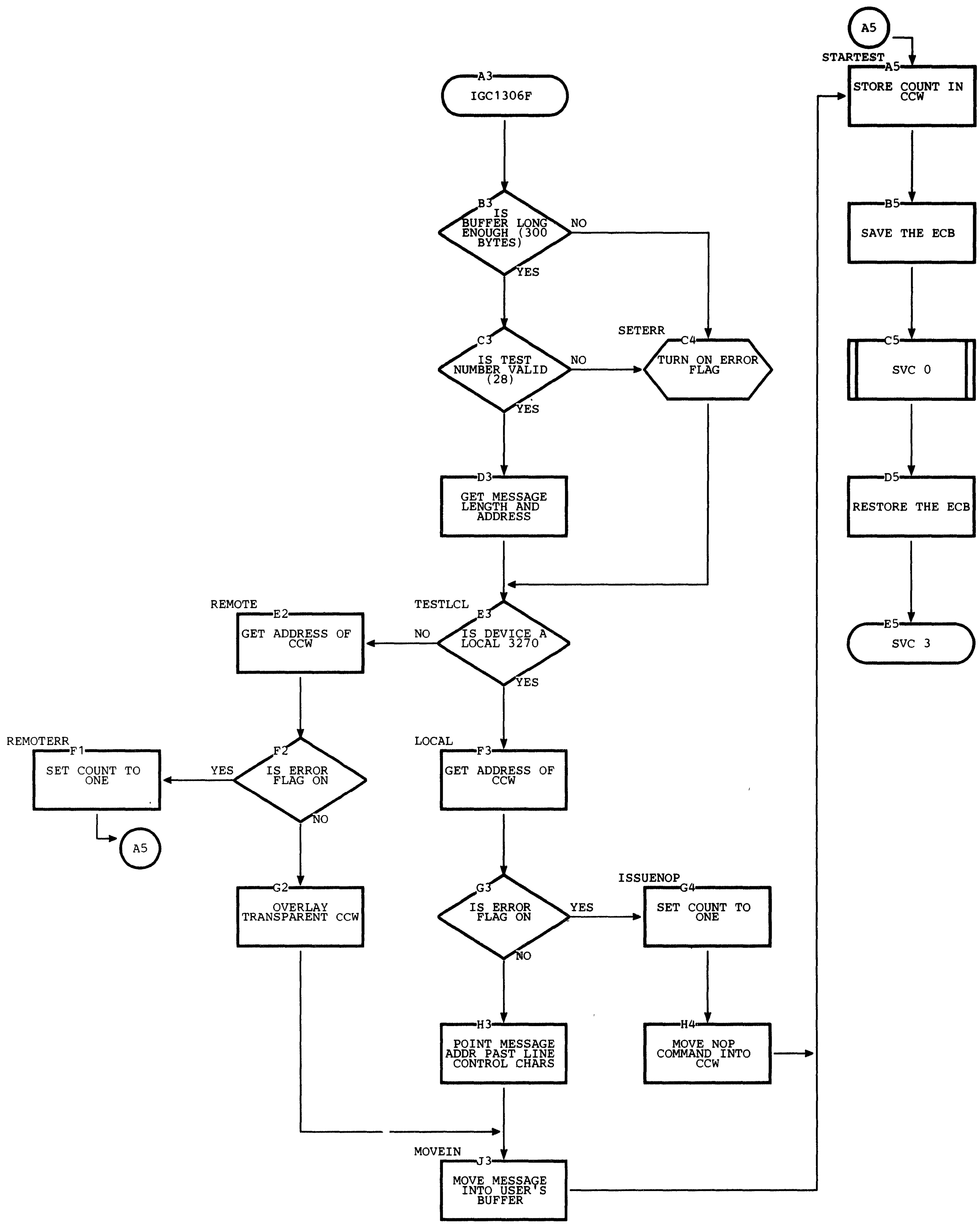


Chart T14. Remote 3270 ASCII Test Module (Types 29, 30, and 32) (IGC1006F)

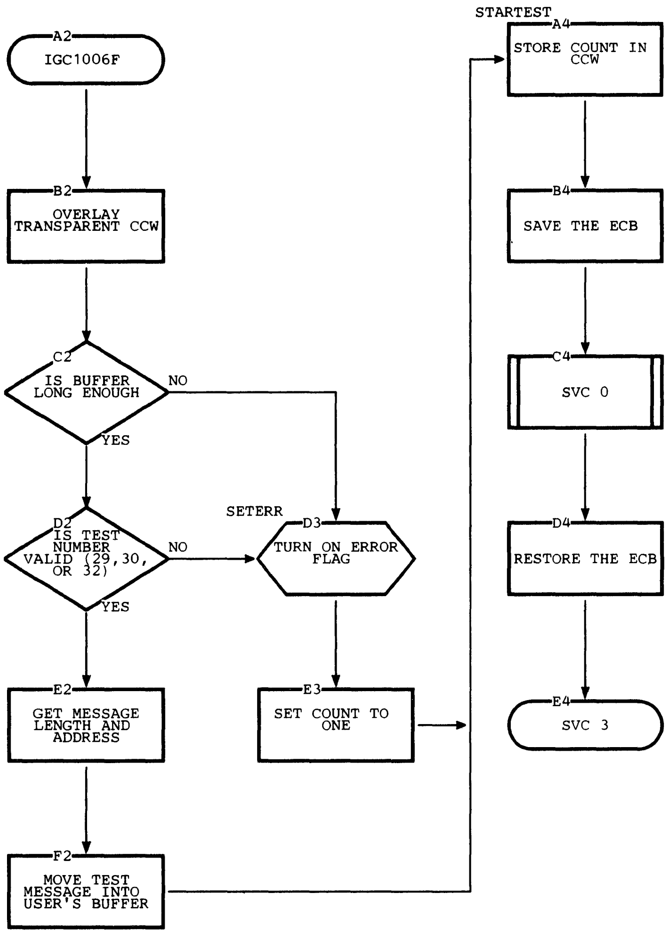


Chart T15. Remote 3270 ASCII Test Module (Types 31, 33, and 34) (IGC1106F)

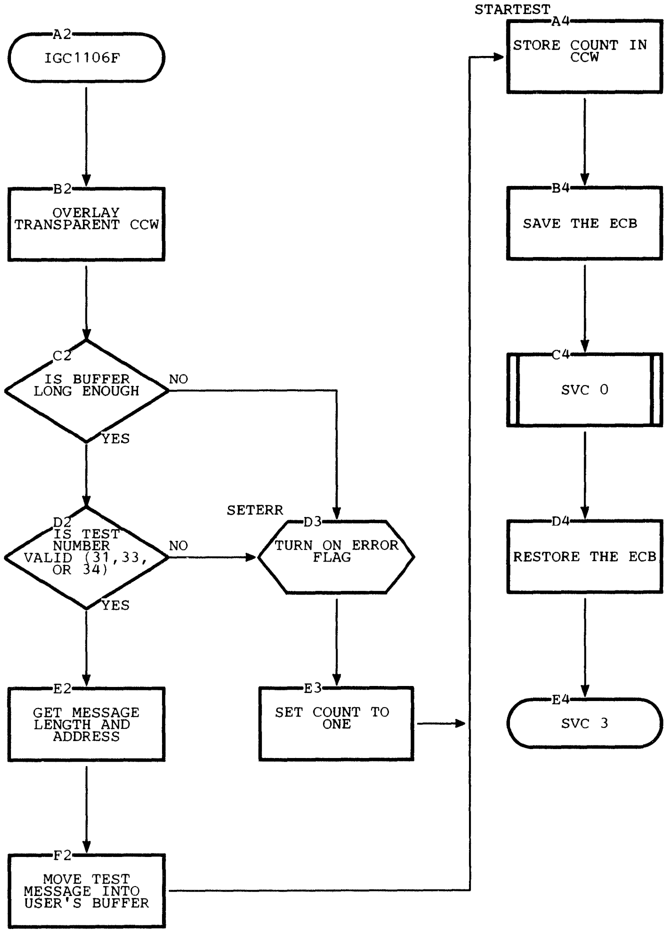
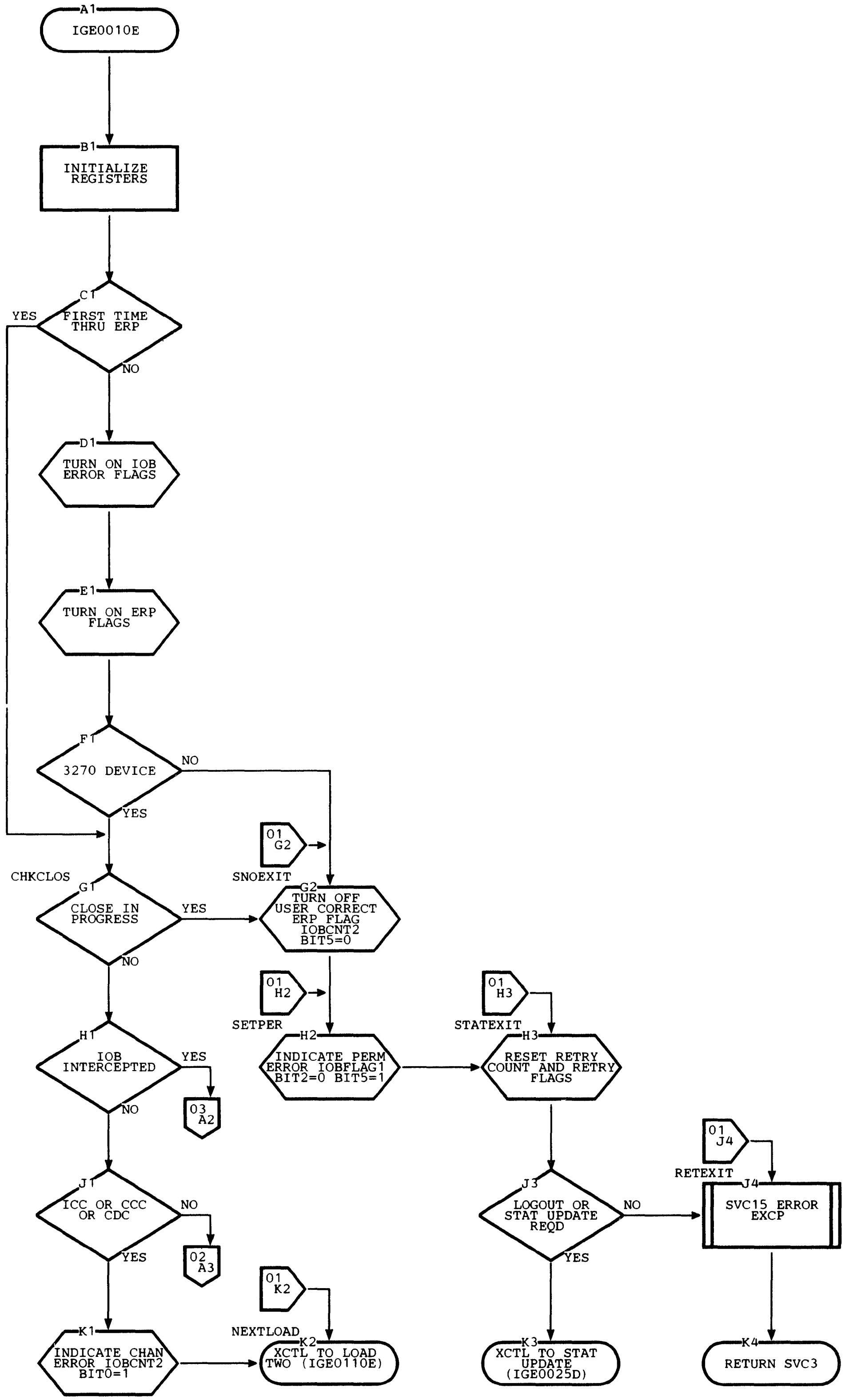


Chart X. Local 3270 ERP Routine (IGE0010E) (Part 1 of 6)



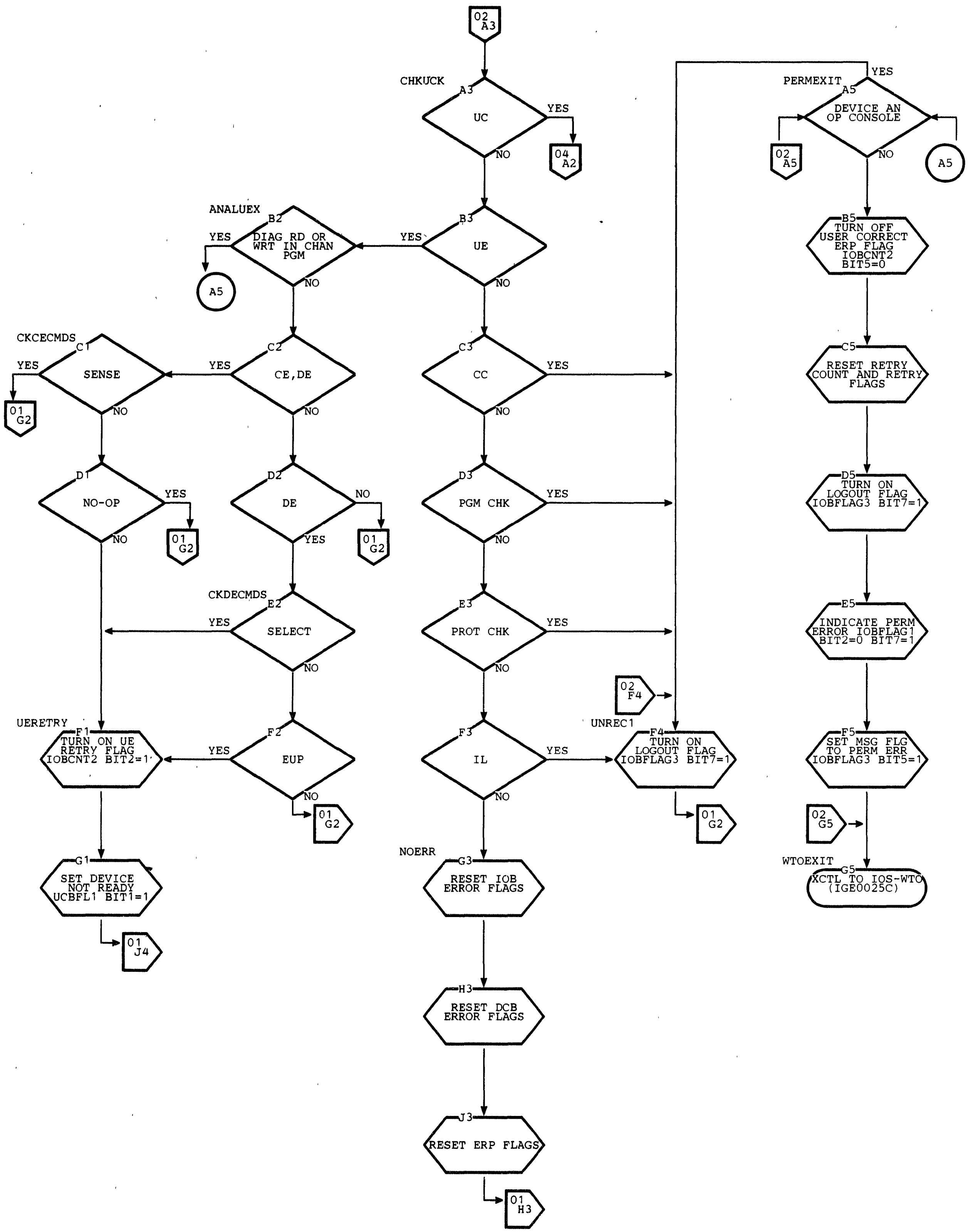

02 A3
CHKUCK
A3
UC
YES
04 A2
NO
PERMEXIT
A5
DEVICE AN OP CONSOLE
YES
02 A5
A5
NO
ANALUEX
B2
DIAG RD OR WRT IN CHAN PGM
YES
A5
NO
B3
UE
YES
NO
B5
TURN OFF USER CORRECT ERP FLAG IOBCNT2 BIT5=0
CKCECMDS
C1
SENSE
YES
01 G2
NO
C2
CE,DE
YES
NO
C3
CC
YES
NO
C5
RESET RETRY COUNT AND RETRY FLAGS
D1
NO-OP
YES
01 G2
NO
D2
DE
NO
01 G2
YES
D3
PGM CHK
YES
NO
D5
TURN ON LOGOUT FLAG IOBFLAG3 BIT7=1
CKDECMDS
E2
SELECT
YES
NO
E3
PROT CHK
YES
NO
E5
INDICATE PERM ERROR IOBFLAG1 BIT2=0 BIT7=1
02 F4
UERETRY
F1
TURN ON UE RETRY FLAG IOBCNT2 BIT2=1
F2
EUP
YES
NO
01 G2
F3
IL
YES
NO
UNREC1
F4
TURN ON LOGOUT FLAG IOBFLAG3 BIT7=1
01 G2
F5
SET MSG FLG TO PERM ERR IOBFLAG3 BIT5=1
02 G5
G1
SET DEVICE NOT READY UCBFL1 BIT1=1
01 J4
NOERR
G3
RESET IOB ERROR FLAGS
WTOEXIT
G5
XCTL TO IOS-WTO (IGE0025C)
H3
RESET DCB ERROR FLAGS
J3
RESET ERP FLAGS
01 H3

Chart X. Local 3270 ERP Routine (IGE0010E) (Part 3 of 6)

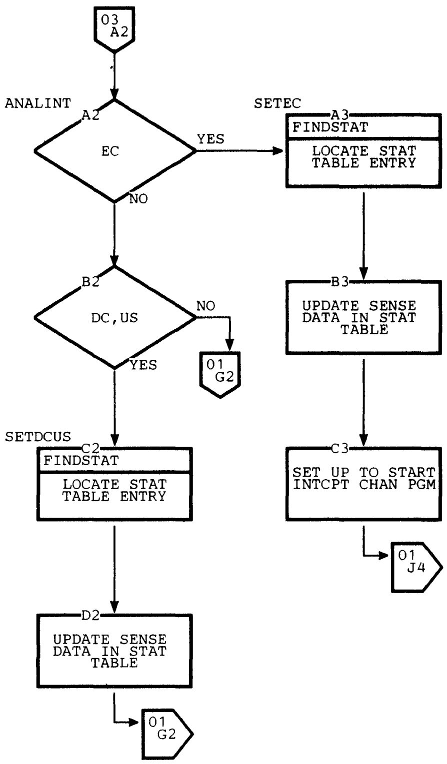
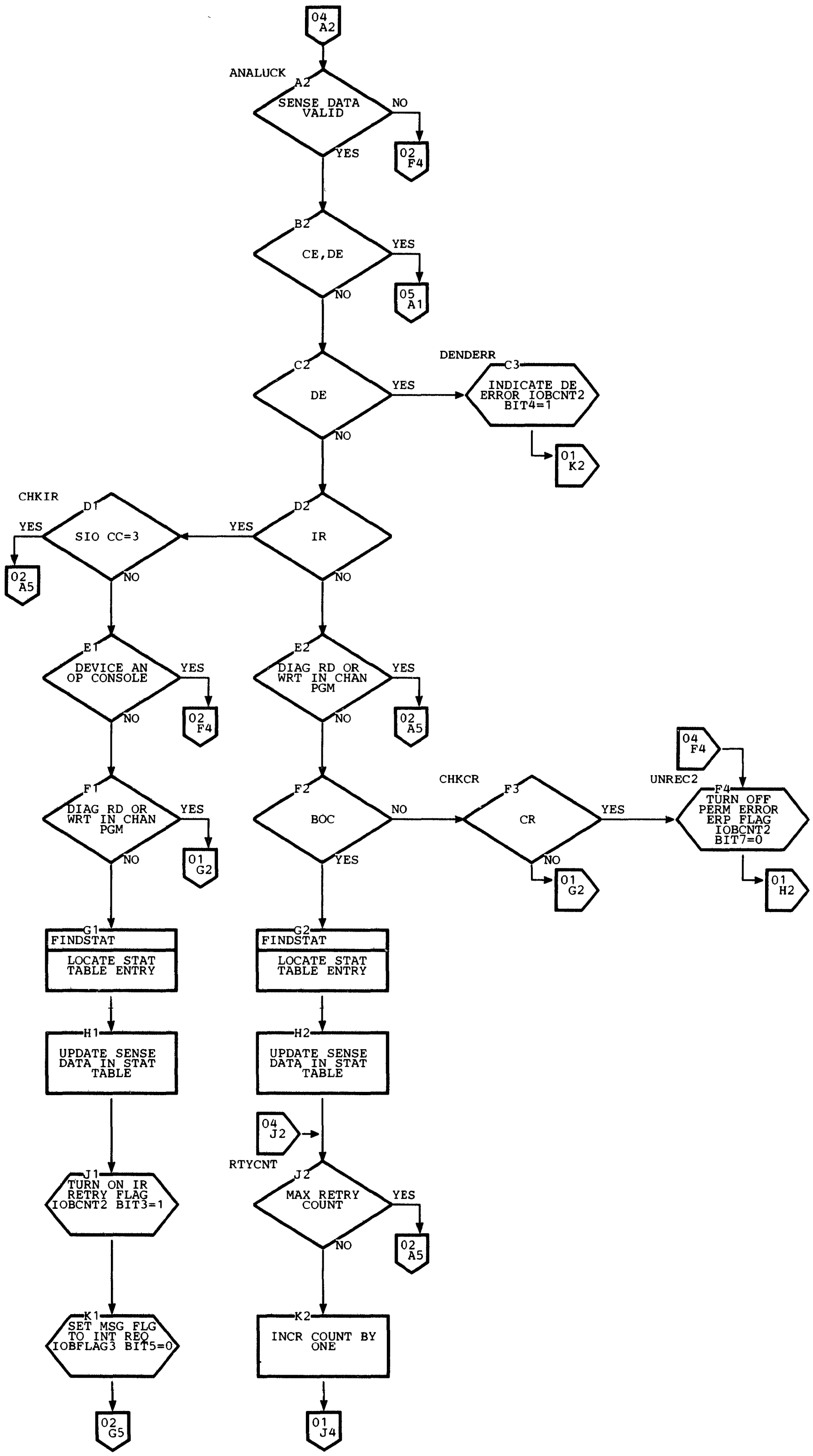


Chart X. Local 3270 ERP Routine (IGE0010E) (Part 4 of 6)



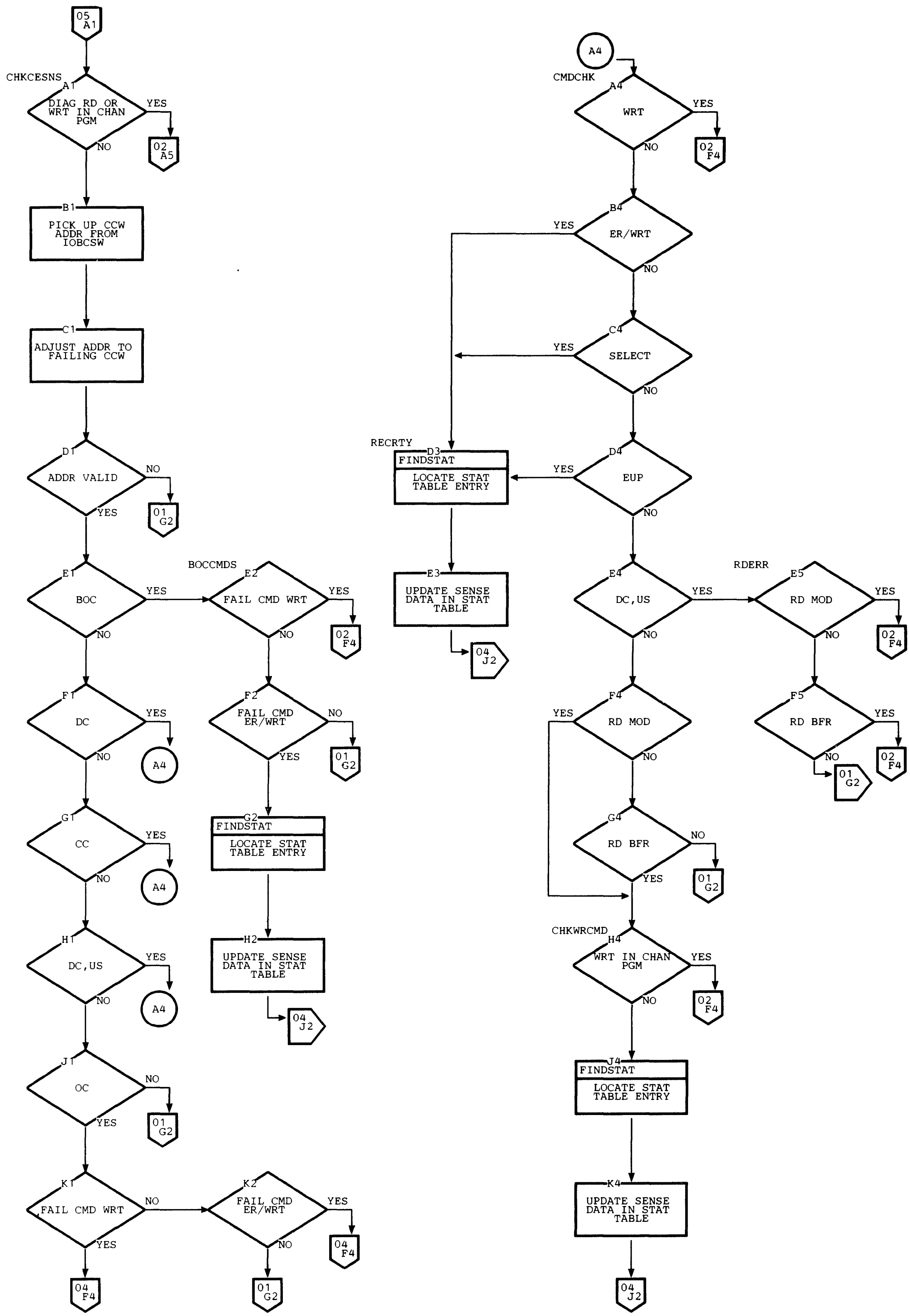

05 A1
CHKCESNS
A1 DIAG RD OR WRT IN CHAN PGM
YES
02 A5
NO
B1 PICK UP CCW ADDR FROM IOBCSW
C1 ADJUST ADDR TO FAILING CCW
D1 ADDR VALID
NO
01 G2
YES
E1 BOC
YES
BOCCMDS
E2 FAIL CMD WRT
YES
02 F4
NO
F1 DC
YES
A4
NO
F2 FAIL CMD ER/WRT
NO
01 G2
YES
G1 CC
YES
A4
NO
G2 FINDSTAT LOCATE STAT TABLE ENTRY
H1 DC,US
YES
A4
NO
H2 UPDATE SENSE DATA IN STAT TABLE
04 J2
J1 OC
NO
01 G2
YES
K1 FAIL CMD WRT
NO
K2 FAIL CMD ER/WRT
YES
04 F4
NO
01 G2
YES
04 F4
A4
CMDCHK
A4 WRT
YES
02 F4
NO
B4 ER/WRT
YES
NO
C4 SELECT
YES
NO
RECRTY
D3 FINDSTAT LOCATE STAT TABLE ENTRY
D4 EUP
YES
NO
E3 UPDATE SENSE DATA IN STAT TABLE
04 J2
E4 DC,US
YES
RDERR
E5 RD MOD
YES
02 F4
NO
F4 RD MOD
YES
NO
F5 RD BFR
YES
02 F4
NO
01 G2
G4 RD BFR
NO
01 G2
YES
CHKWRCMD
H4 WRT IN CHAN PGM
YES
02 F4
NO
J4 FINDSTAT LOCATE STAT TABLE ENTRY
K4 UPDATE SENSE DATA IN STAT TABLE
04 J2

Chart X. Local 3270 ERP Routine (IGE0010E) (Part 6 of 6)

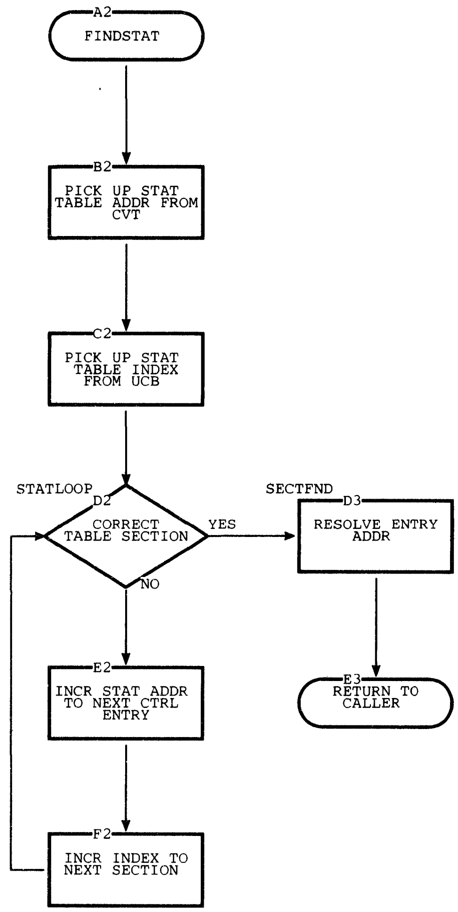


Chart Y. Local 3270 ERP Routine (IGE0110E) (Part 1 of 3)

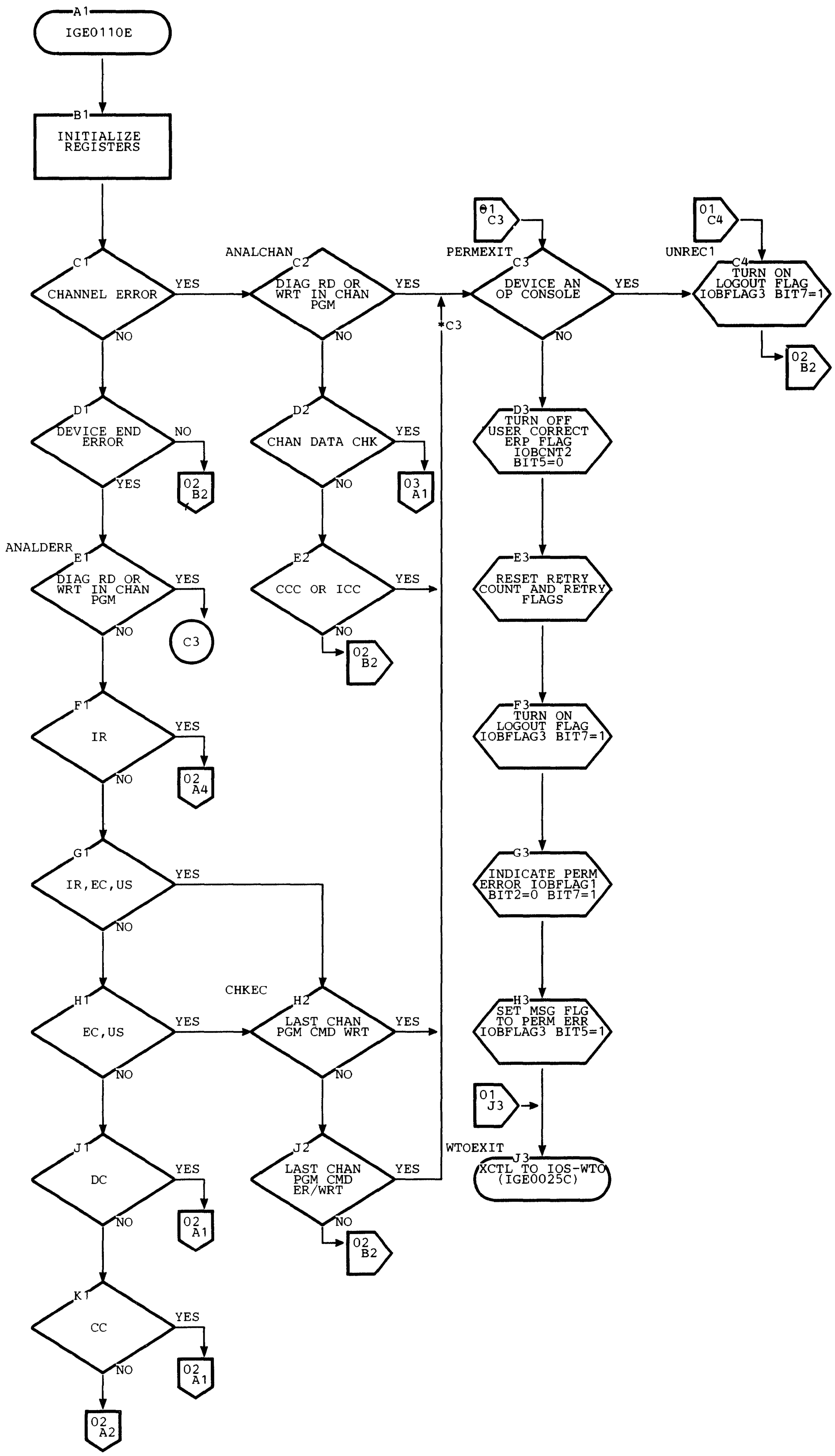
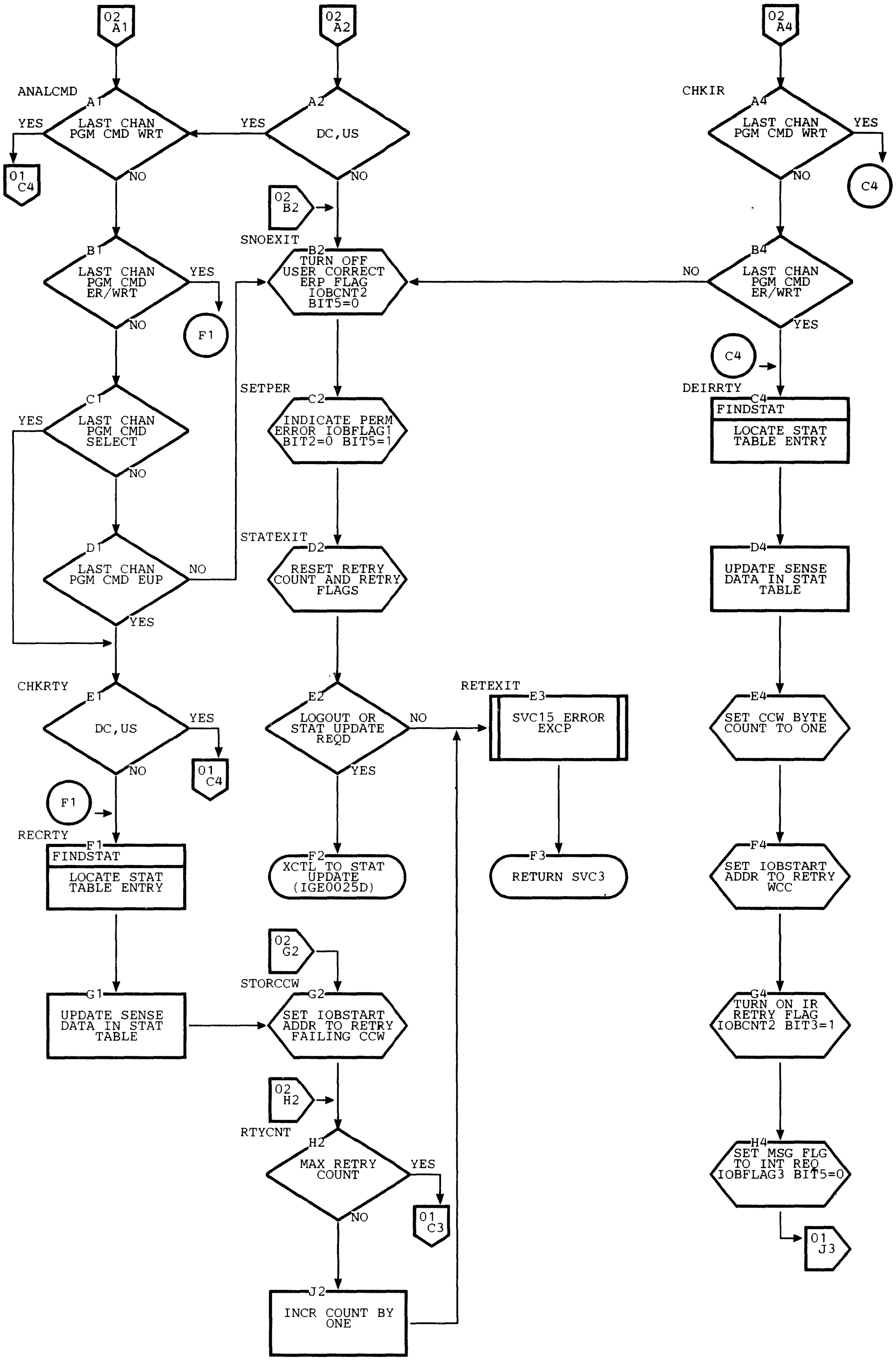
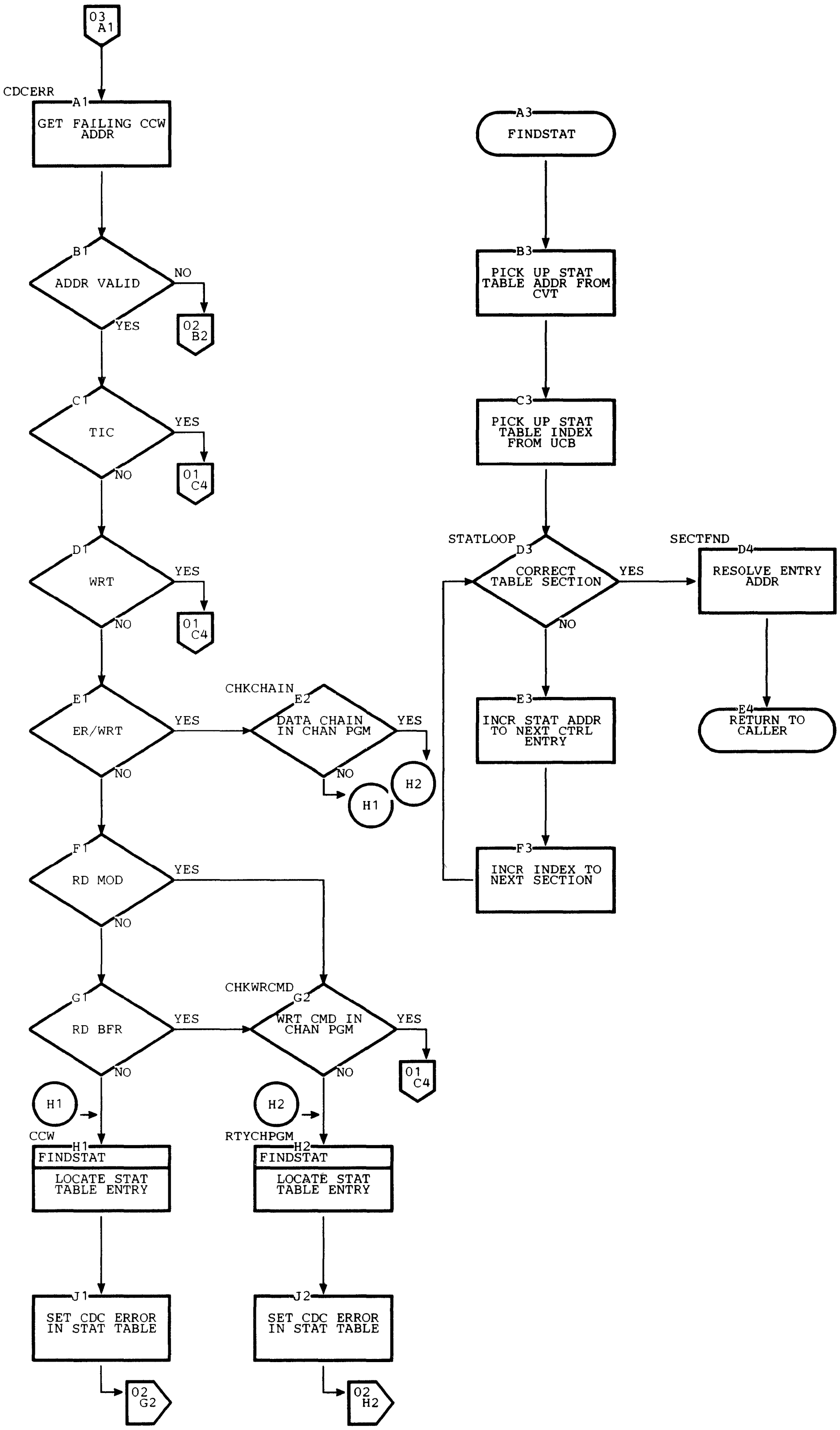
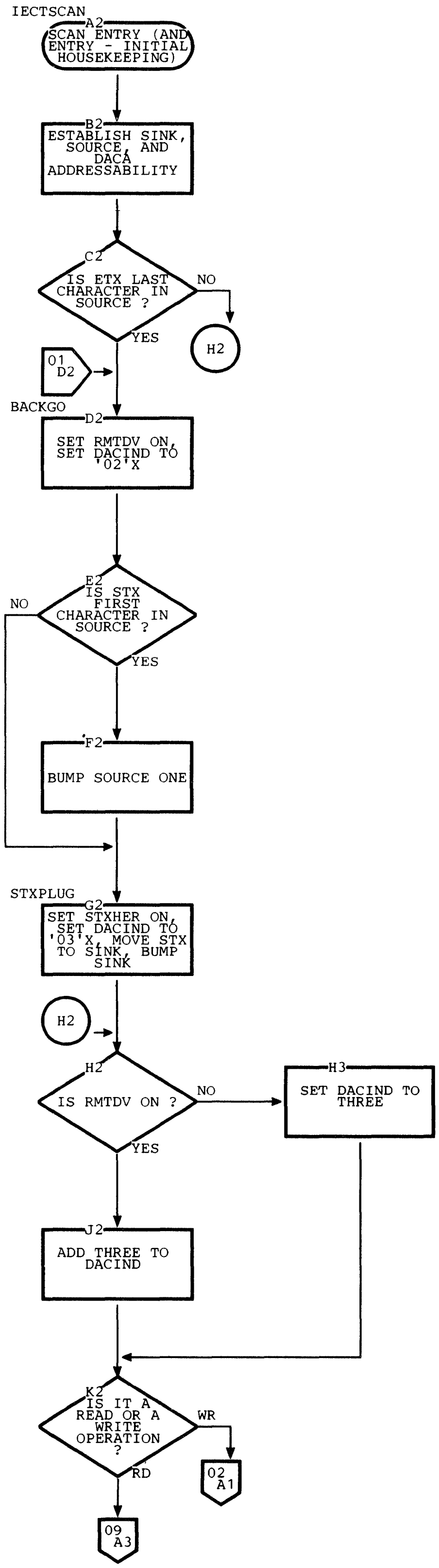
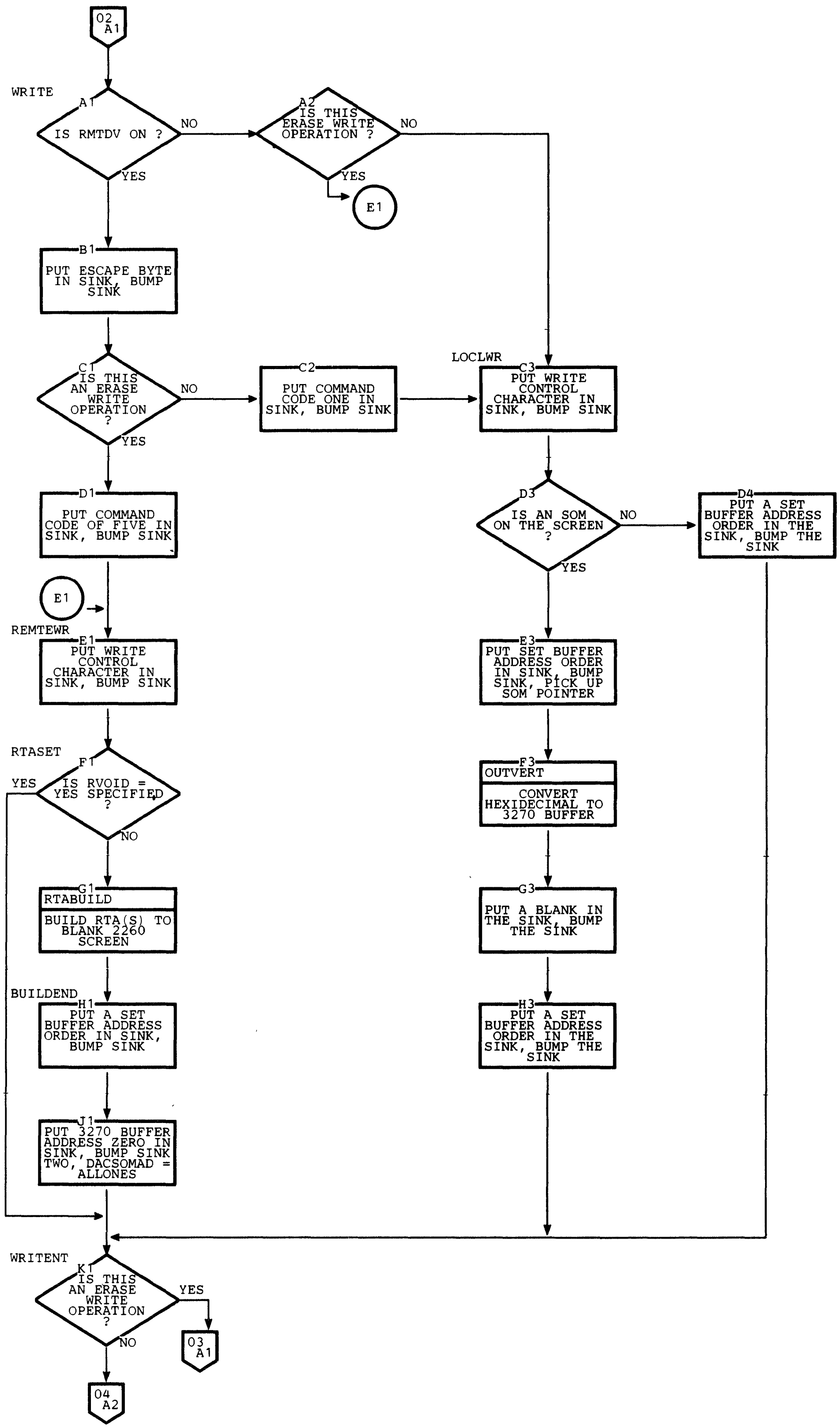


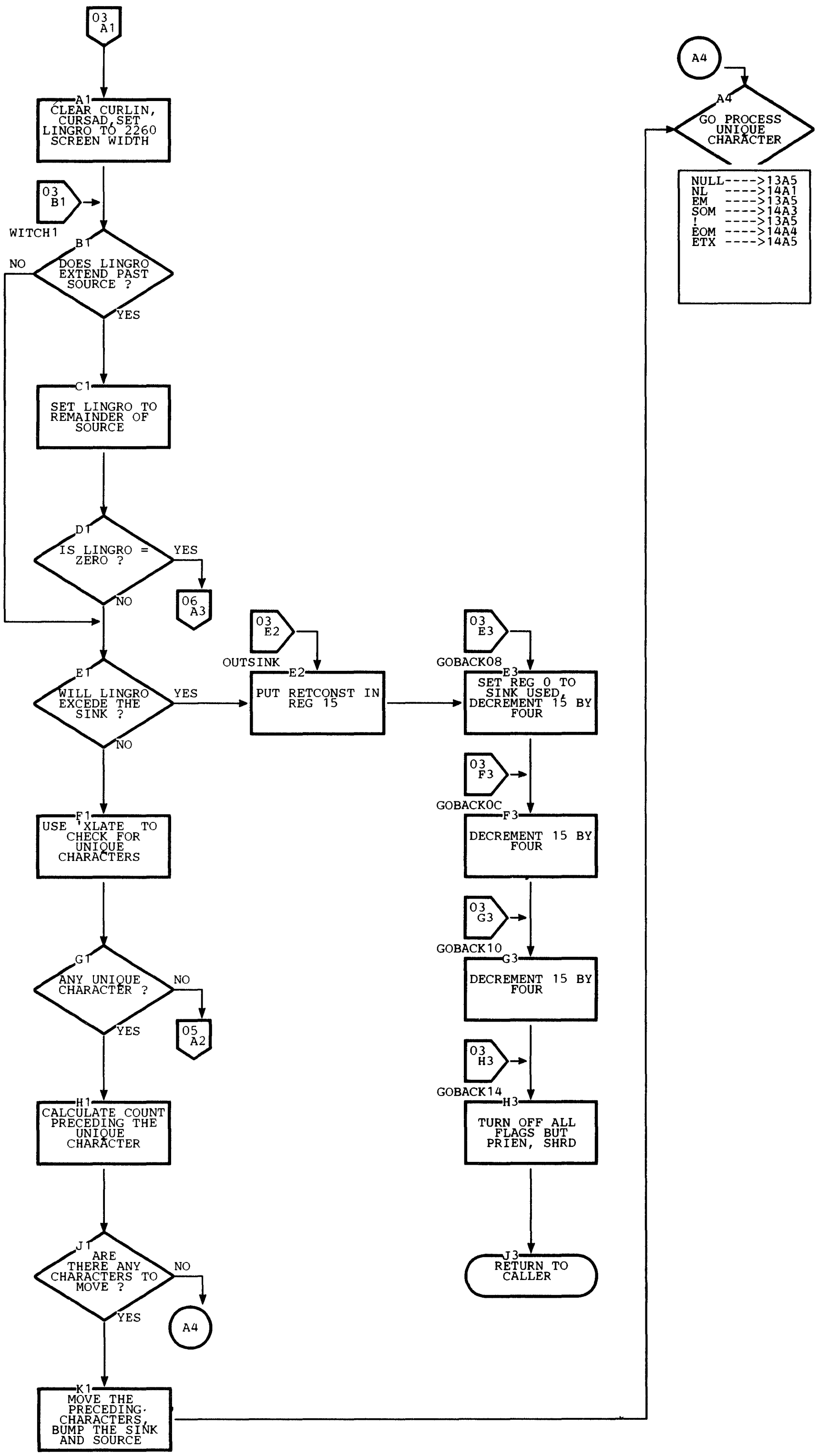
Chart Y. Local 3270 ERP Routine (IGE0110E) (Part 2 of 3)

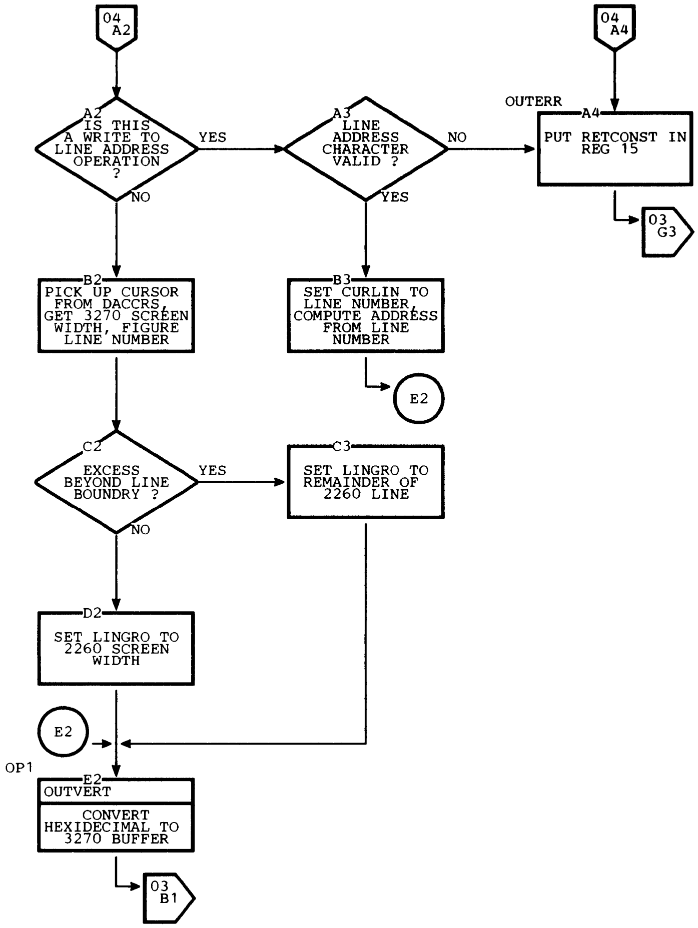



03 A1
CDCERR
A1
GET FAILING CCW ADDR
B1
ADDR VALID
NO
02 B2
YES
C1
TIC
YES
01 C4
NO
D1
WRT
YES
01 C4
NO
E1
ER/WRT
YES
CHKCHAIN
E2
DATA CHAIN IN CHAN PGM
YES
H2
NO
H1
NO
F1
RD MOD
YES
NO
G1
RD BFR
YES
CHKWRCMD
G2
WRT CMD IN CHAN PGM
YES
01 C4
NO
H1
CCW
H1
FINDSTAT
LOCATE STAT TABLE ENTRY
H2
RTYCHPGM
H2
FINDSTAT
LOCATE STAT TABLE ENTRY
J1
SET CDC ERROR IN STAT TABLE
02 G2
J2
SET CDC ERROR IN STAT TABLE
02 H2
A3
FINDSTAT
B3
PICK UP STAT TABLE ADDR FROM CVT
C3
PICK UP STAT TABLE INDEX FROM UCB
STATLOOP
D3
CORRECT TABLE SECTION
YES
NO
E3
INCR STAT ADDR TO NEXT CTRL ENTRY
F3
INCR INDEX TO NEXT SECTION
SECTFND
D4
RESOLVE ENTRY ADDR
E4
RETURN TO CALLER


IECTSCAN
A2
SCAN ENTRY (AND ENTRY - INITIAL HOUSEKEEPING)
B2
ESTABLISH SINK, SOURCE, AND DACA ADDRESSABILITY
C2
IS ETX LAST CHARACTER IN SOURCE ?
NO
H2
YES
01 D2
BACKGO
D2
SET RMTDV ON, SET DACIND TO '02'X
E2
IS STX FIRST CHARACTER IN SOURCE ?
NO
YES
F2
BUMP SOURCE ONE
STXPLUG
G2
SET STXHER ON, SET DACIND TO '03'X, MOVE STX TO SINK, BUMP SINK
H2
H2
IS RMTDV ON ?
NO
H3
SET DACIND TO THREE
YES
J2
ADD THREE TO DACIND
K2
IS IT A READ OR A WRITE OPERATION ?
WR
02 A1
RD
09 A3


02 A1
WRITE
A1 IS RMTDV ON ?
NO
YES
A2 IS THIS ERASE WRITE OPERATION ?
NO
YES
E1
B1 PUT ESCAPE BYTE IN SINK, BUMP SINK
C1 IS THIS AN ERASE WRITE OPERATION ?
NO
YES
C2 PUT COMMAND CODE ONE IN SINK, BUMP SINK
LOCLWR
C3 PUT WRITE CONTROL CHARACTER IN SINK, BUMP SINK
D1 PUT COMMAND CODE OF FIVE IN SINK, BUMP SINK
D3 IS AN SOM ON THE SCREEN ?
NO
YES
D4 PUT A SET BUFFER ADDRESS ORDER IN THE SINK, BUMP THE SINK
E1
REMTEWR
E1 PUT WRITE CONTROL CHARACTER IN SINK, BUMP SINK
E3 PUT SET BUFFER ADDRESS ORDER IN SINK, BUMP SINK, PICK UP SOM POINTER
RTASET
F1 IS RVOID = YES SPECIFIED ?
YES
NO
F3 OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
G1 RTABUILD
BUILD RTA(S) TO BLANK 2260 SCREEN
G3 PUT A BLANK IN THE SINK, BUMP THE SINK
BUILDEND
H1 PUT A SET BUFFER ADDRESS ORDER IN SINK, BUMP SINK
H3 PUT A SET BUFFER ADDRESS ORDER IN THE SINK, BUMP THE SINK
J1 PUT 3270 BUFFER ADDRESS ZERO IN SINK, BUMP SINK TWO, DACSOMAD = ALLONES
WRITENT
K1 IS THIS AN ERASE WRITE OPERATION ?
YES
NO
03 A1
04 A2


03 A1
A1 CLEAR CURLIN, CURSAD, SET LINGRO TO 2260 SCREEN WIDTH
03 B1
WITCH1
B1 DOES LINGRO EXTEND PAST SOURCE ?
NO
YES
C1 SET LINGRO TO REMAINDER OF SOURCE
D1 IS LINGRO = ZERO ?
YES
06 A3
NO
E1 WILL LINGRO EXCEDE THE SINK ?
YES
NO
F1 USE 'XLATE TO CHECK FOR UNIQUE CHARACTERS
G1 ANY UNIQUE CHARACTER ?
NO
05 A2
YES
H1 CALCULATE COUNT PRECEDING THE UNIQUE CHARACTER
J1 ARE THERE ANY CHARACTERS TO MOVE ?
NO
A4
YES
K1 MOVE THE PRECEDING CHARACTERS, BUMP THE SINK AND SOURCE
03 E2
OUTSINK
E2 PUT RETCONST IN REG 15
03 E3
GOBACK08
E3 SET REG 0 TO SINK USED, DECREMENT 15 BY FOUR
03 F3
GOBACK0C
F3 DECREMENT 15 BY FOUR
03 G3
GOBACK10
G3 DECREMENT 15 BY FOUR
03 H3
GOBACK14
H3 TURN OFF ALL FLAGS BUT PRIEN, SHRD
J3 RETURN TO CALLER
A4
A4 GO PROCESS UNIQUE CHARACTER
NULL---->13A5
NL ---->14A1
EM ---->13A5
SOM ---->14A3
! ---->13A5
EOM ---->14A4
ETX ---->14A5


04 A2
A2 IS THIS A WRITE TO LINE ADDRESS OPERATION ?
YES
NO
A3 LINE ADDRESS CHARACTER VALID ?
NO
YES
04 A4
OUTERR
A4 PUT RETCONST IN REG 15
03 G3
B2 PICK UP CURSOR FROM DACCRS, GET 3270 SCREEN WIDTH, FIGURE LINE NUMBER
B3 SET CURLIN TO LINE NUMBER, COMPUTE ADDRESS FROM LINE NUMBER
E2
C2 EXCESS BEYOND LINE BOUNDRY ?
YES
NO
C3 SET LINGRO TO REMAINDER OF 2260 LINE
D2 SET LINGRO TO 2260 SCREEN WIDTH
E2
OP1
E2 OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
03 B1

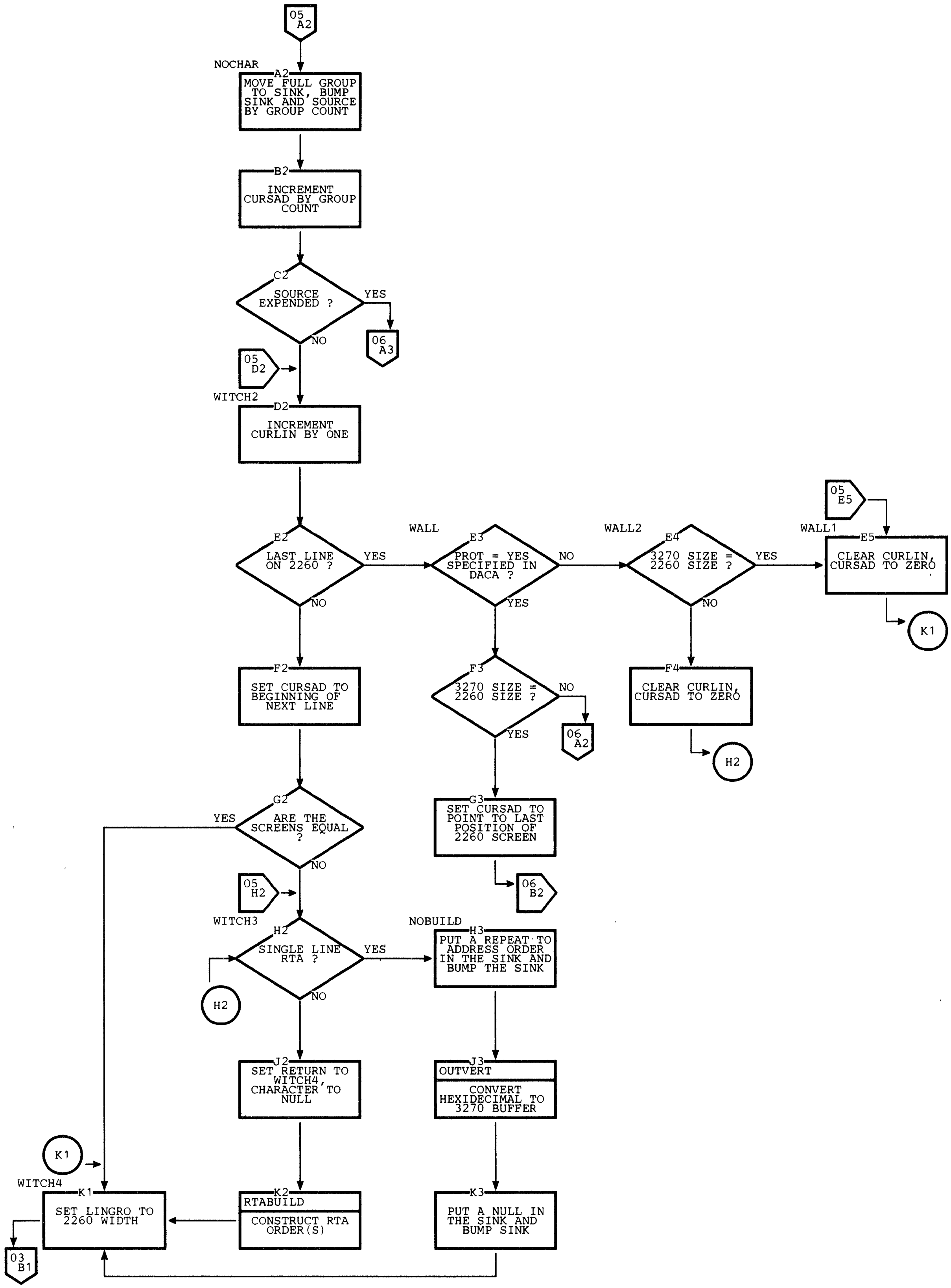
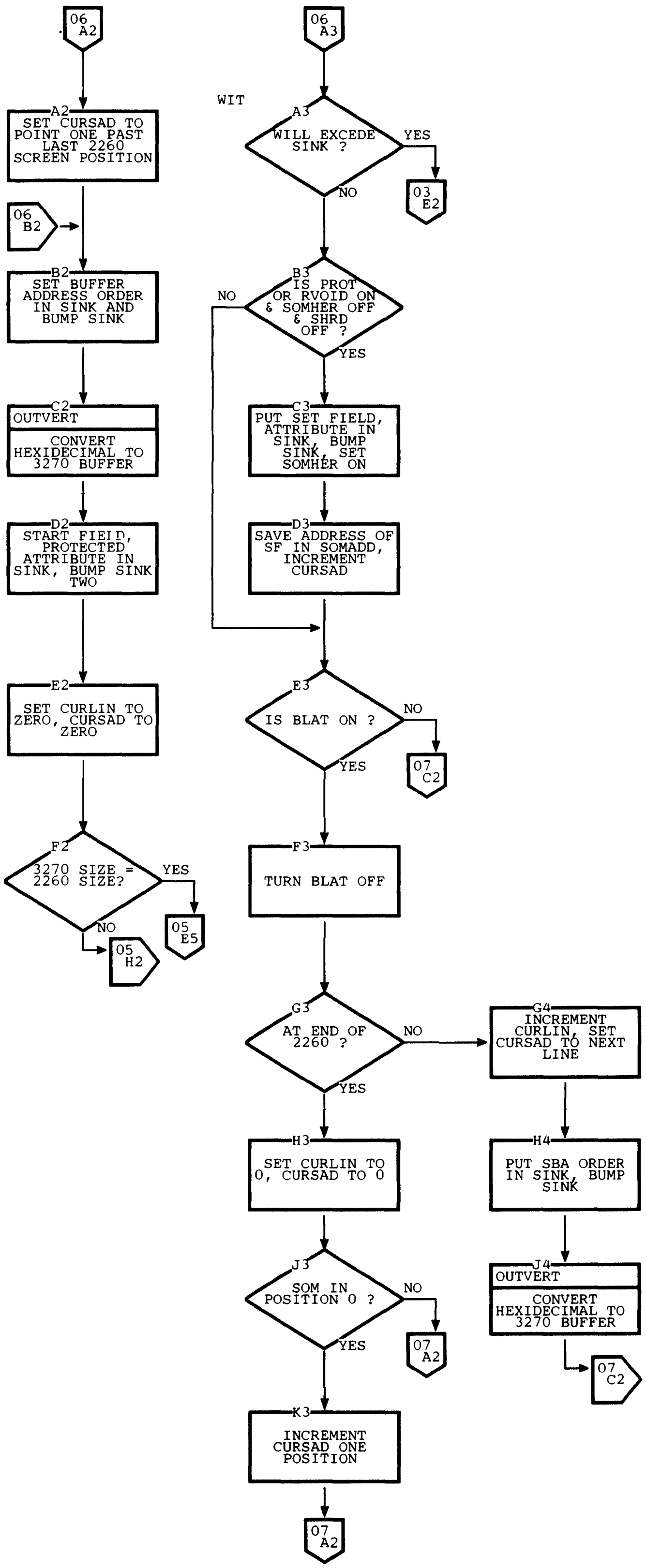
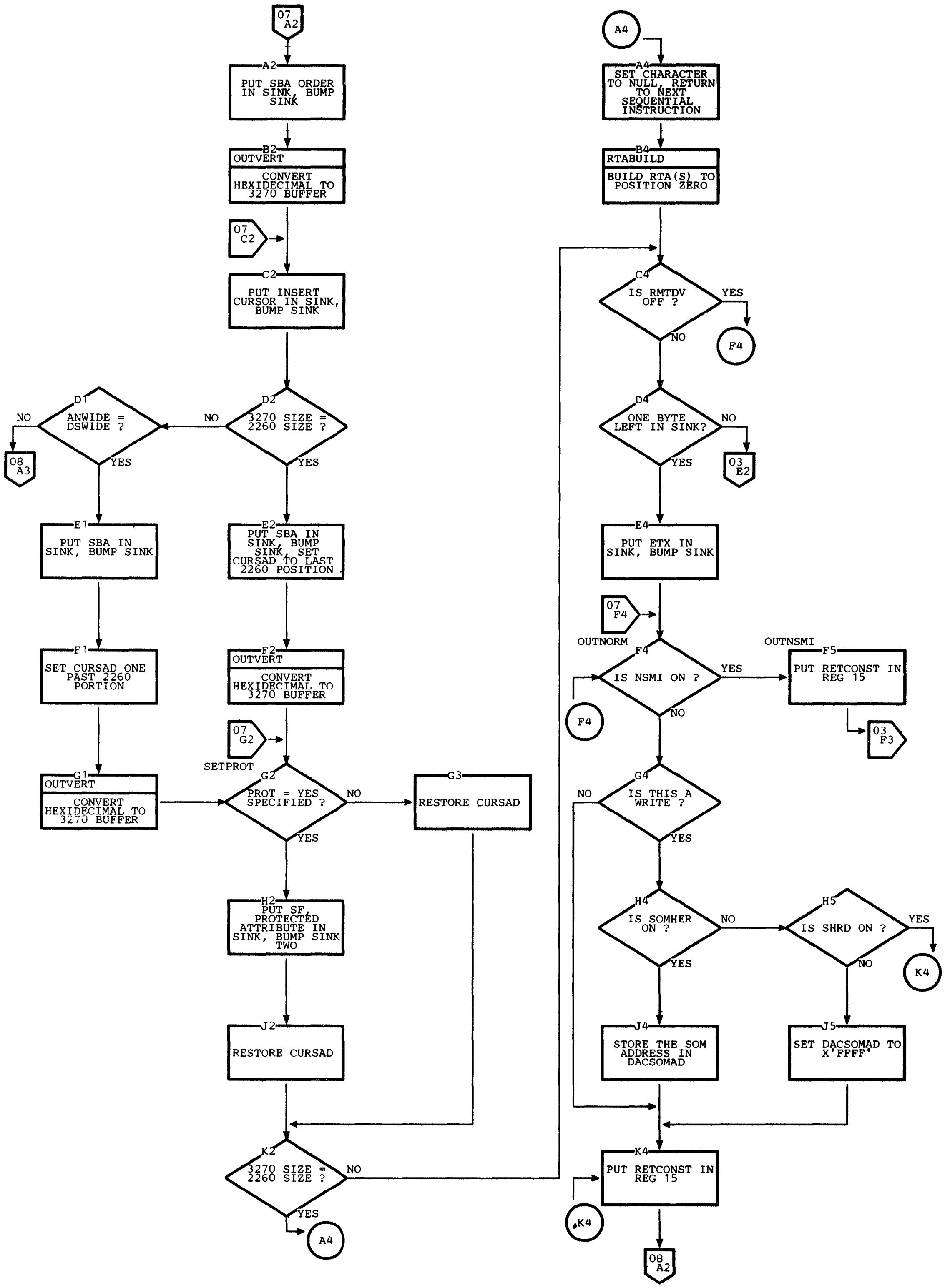
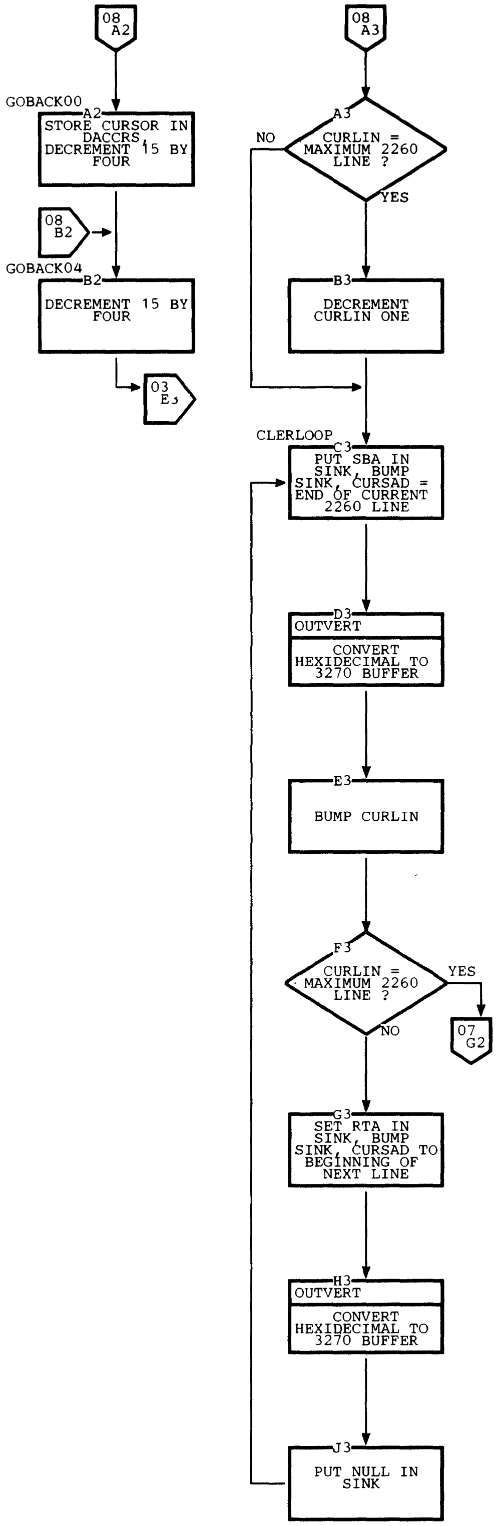
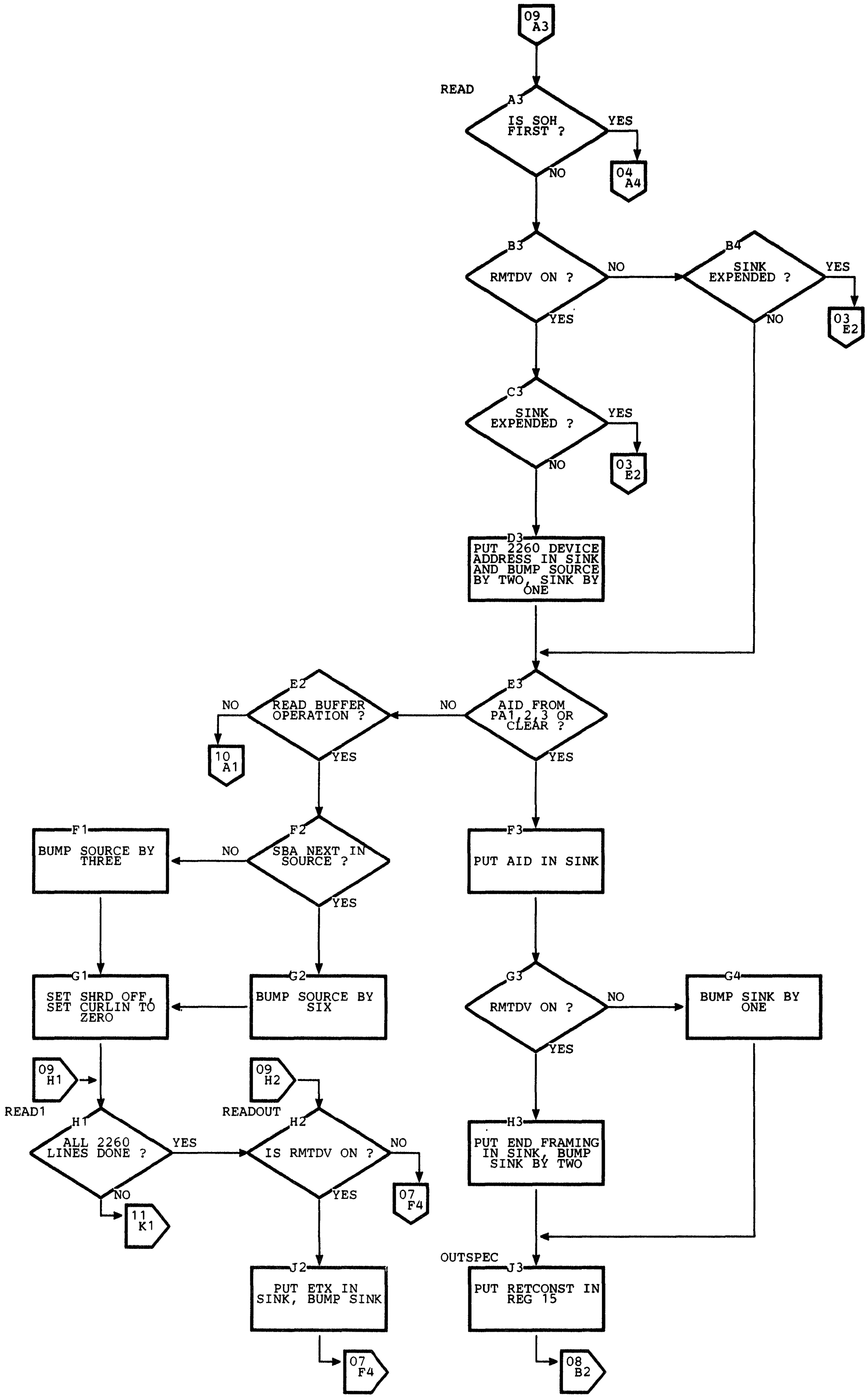

05 A2
NOCHAR
A2
MOVE FULL GROUP TO SINK, BUMP SINK AND SOURCE BY GROUP COUNT
B2
INCREMENT CURSAD BY GROUP COUNT
C2
SOURCE EXPENDED ?
YES
06 A3
NO
05 D2
WITCH2
D2
INCREMENT CURLIN BY ONE
E2
LAST LINE ON 2260 ?
YES
WALL
E3
PROT = YES SPECIFIED IN DACA ?
NO
WALL2
E4
3270 SIZE = 2260 SIZE ?
YES
05 E5
WALL1
E5
CLEAR CURLIN, CURSAD TO ZERO
K1
NO
F2
SET CURSAD TO BEGINNING OF NEXT LINE
F3
3270 SIZE = 2260 SIZE ?
NO
06 A2
YES
G3
SET CURSAD TO POINT TO LAST POSITION OF 2260 SCREEN
06 B2
F4
CLEAR CURLIN, CURSAD TO ZERO
H2
G2
ARE THE SCREENS EQUAL ?
YES
NO
05 H2
WITCH3
H2
SINGLE LINE RTA ?
YES
NOBUILD
H3
PUT A REPEAT TO ADDRESS ORDER IN THE SINK AND BUMP THE SINK
H2
NO
J2
SET RETURN TO WITCH4, CHARACTER TO NULL
J3
OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
K1
WITCH4
K1
SET LINGRO TO 2260 WIDTH
K2
RTABUILD
CONSTRUCT RTA ORDER(S)
K3
PUT A NULL IN THE SINK AND BUMP SINK
03 B1

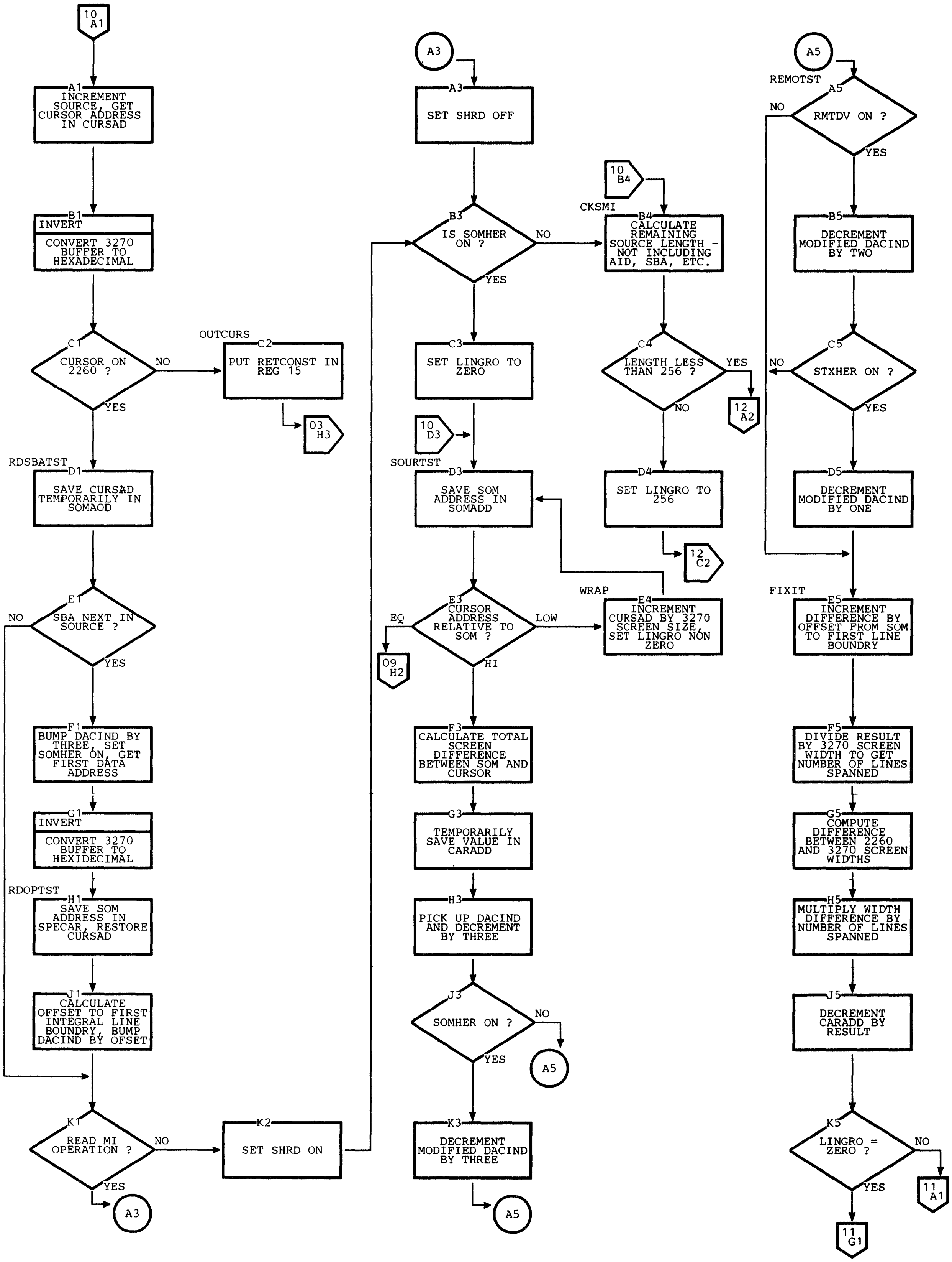
Chart Z. 3270 SCAN (IECTSCAN) (Part 6 of 19)

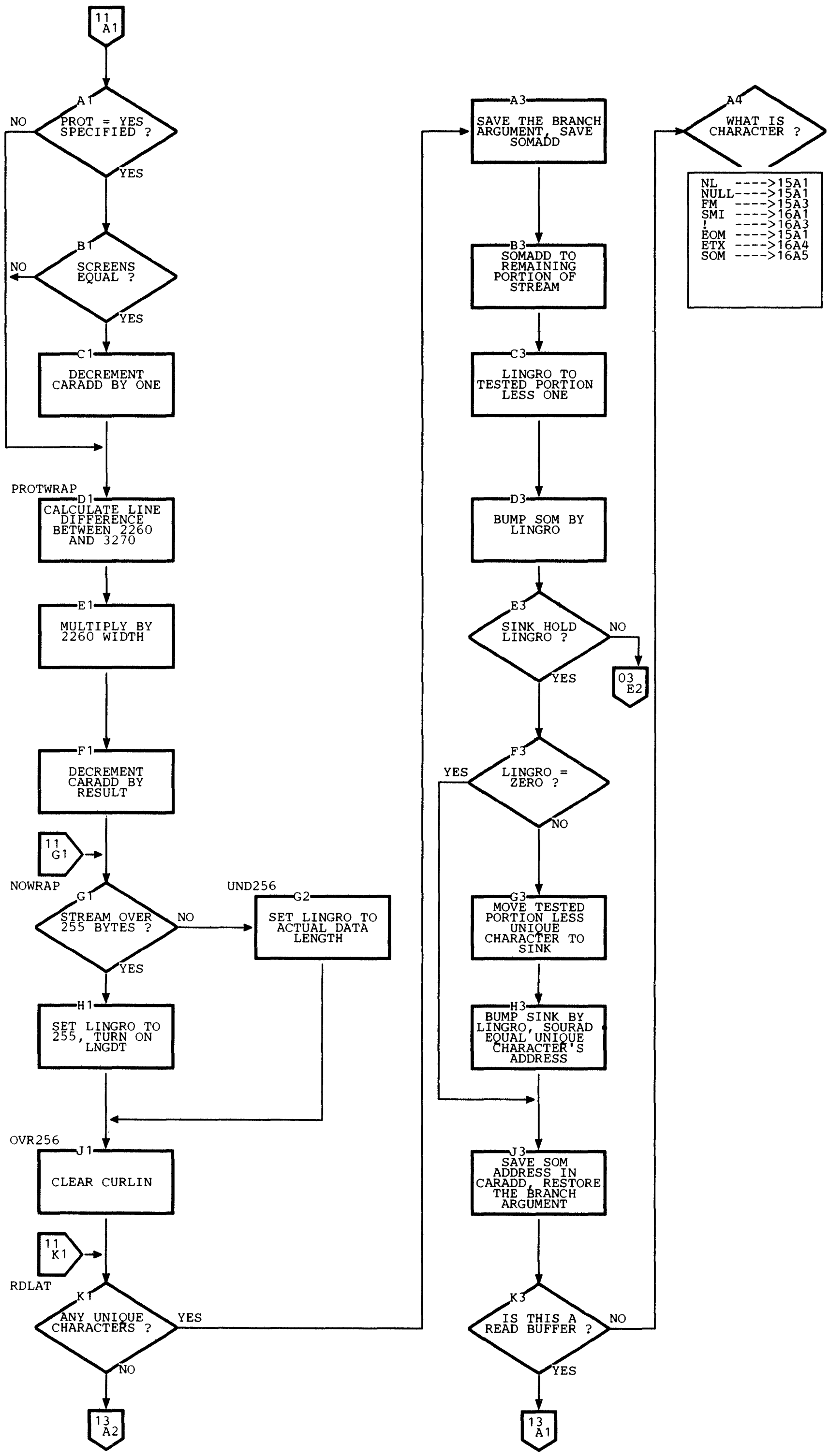



07 A2
A2 PUT SBA ORDER IN SINK, BUMP SINK
B2 OUTVERT CONVERT HEXIDECIMAL TO 3270 BUFFER
07 C2
C2 PUT INSERT CURSOR IN SINK, BUMP SINK
D2 3270 SIZE = 2260 SIZE ?
NO
YES
D1 ANWIDE = DSWIDE ?
NO
YES
08 A3
E1 PUT SBA IN SINK, BUMP SINK
E2 PUT SBA IN SINK, BUMP SINK, SET CURSAD TO LAST 2260 POSITION
F1 SET CURSAD ONE PAST 2260 PORTION
F2 OUTVERT CONVERT HEXIDECIMAL TO 3270 BUFFER
G1 OUTVERT CONVERT HEXIDECIMAL TO 3270 BUFFER
07 G2
SETPROT
G2 PROT = YES SPECIFIED ?
NO
YES
G3 RESTORE CURSAD
H2 PUT SF, PROTECTED ATTRIBUTE IN SINK, BUMP SINK TWO
J2 RESTORE CURSAD
K2 3270 SIZE = 2260 SIZE ?
NO
YES
A4
A4
A4 SET CHARACTER TO NULL, RETURN TO NEXT SEQUENTIAL INSTRUCTION
B4 RTABUILD BUILD RTA(S) TO POSITION ZERO
C4 IS RMTDV OFF ?
YES
NO
F4
D4 ONE BYTE LEFT IN SINK?
NO
YES
03 E2
E4 PUT ETX IN SINK, BUMP SINK
07 F4
OUTNORM
F4 IS NSMI ON ?
YES
NO
F4
OUTNSMI
F5 PUT RETCONST IN REG 15
03 F3
G4 IS THIS A WRITE ?
NO
YES
H4 IS SOMHER ON ?
NO
YES
H5 IS SHRD ON ?
YES
NO
K4
J4 STORE THE SOM ADDRESS IN DACSOMAD
J5 SET DACSOMAD TO X'FFFF'
K4 PUT RETCONST IN REG 15
K4
08 A2

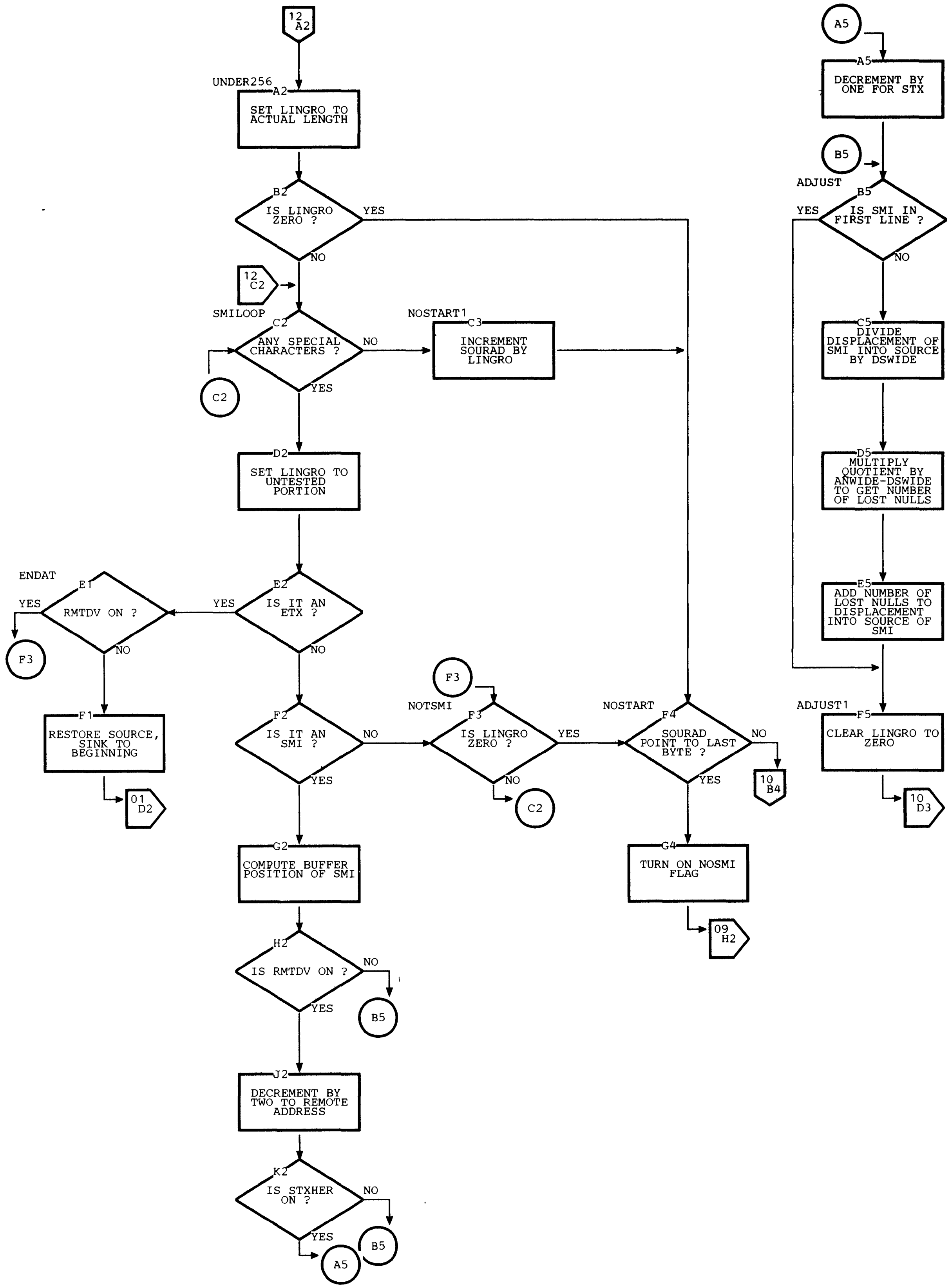

08 A2
GOBACK00
A2
STORE CURSOR IN DACCRS, DECREMENT 15 BY FOUR
08 B2
GOBACK04
B2
DECREMENT 15 BY FOUR
03 E3
08 A3
A3
CURLIN = MAXIMUM 2260 LINE ?
NO
YES
B3
DECREMENT CURLIN ONE
CLERLOOP
C3
PUT SBA IN SINK, BUMP SINK, CURSAD = END OF CURRENT 2260 LINE
D3
OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
E3
BUMP CURLIN
F3
CURLIN = MAXIMUM 2260 LINE ?
YES
NO
07 G2
G3
SET RTA IN SINK, BUMP SINK, CURSAD TO BEGINNING OF NEXT LINE
H3
OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
J3
PUT NULL IN SINK

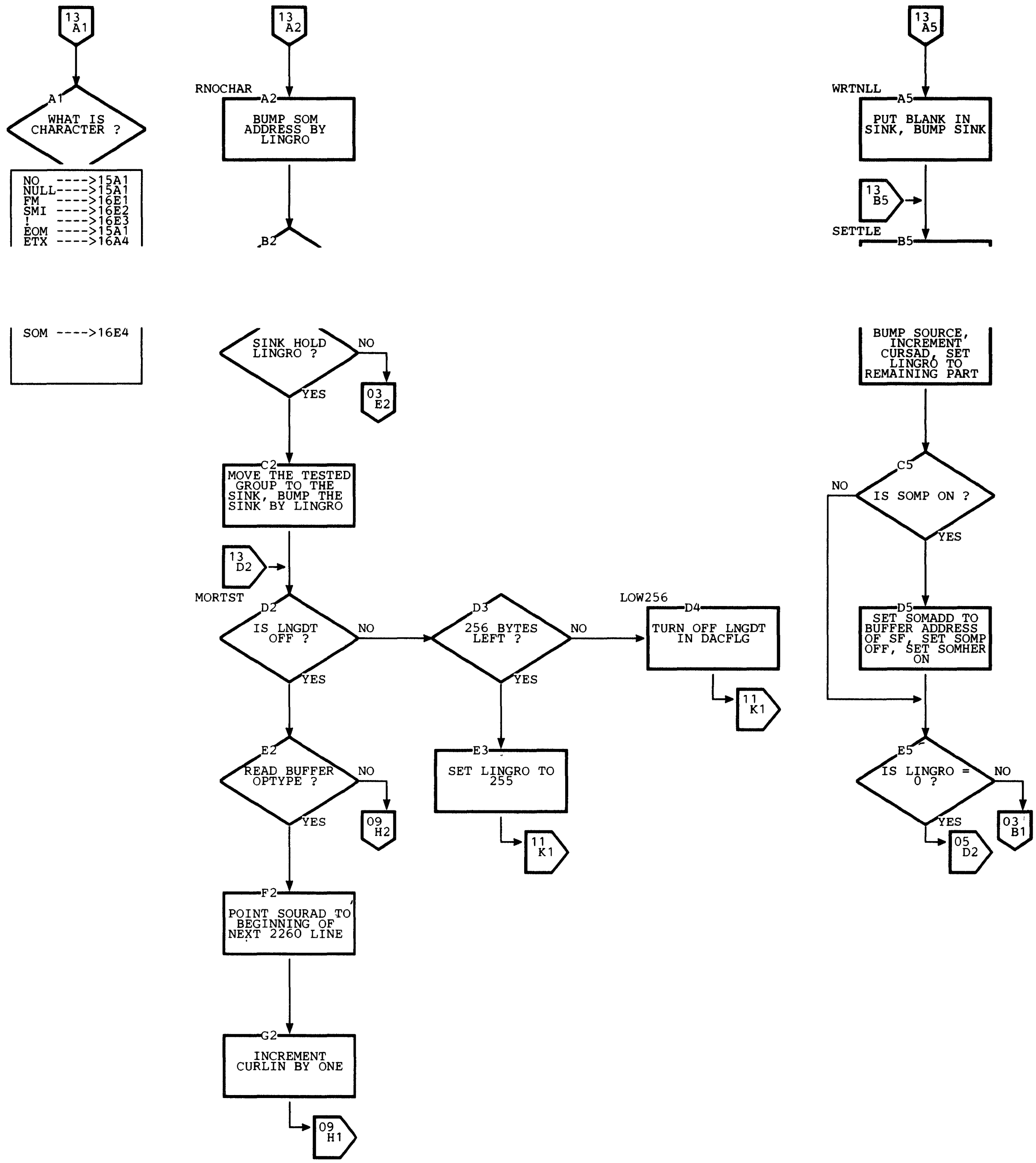

09 A3
READ
A3
IS SOH FIRST ?
YES
NO
04 A4
B3
RMTDV ON ?
NO
YES
B4
SINK EXPENDED ?
YES
NO
03 E2
C3
SINK EXPENDED ?
YES
NO
03 E2
D3
PUT 2260 DEVICE ADDRESS IN SINK AND BUMP SOURCE BY TWO, SINK BY ONE
E2
READ BUFFER OPERATION ?
NO
YES
10 A1
E3
AID FROM PA1,2,3 OR CLEAR ?
NO
YES
F1
BUMP SOURCE BY THREE
F2
SBA NEXT IN SOURCE ?
NO
YES
F3
PUT AID IN SINK
G1
SET SHRD OFF, SET CURLIN TO ZERO
G2
BUMP SOURCE BY SIX
G3
RMTDV ON ?
NO
YES
G4
BUMP SINK BY ONE
09 H1
READ1
H1
ALL 2260 LINES DONE ?
YES
NO
11 K1
09 H2
READOUT
H2
IS RMTDV ON ?
NO
YES
07 F4
H3
PUT END FRAMING IN SINK, BUMP SINK BY TWO
OUTSPEC
J2
PUT ETX IN SINK, BUMP SINK
07 F4
J3
PUT RETCONST IN REG 15
08 B2

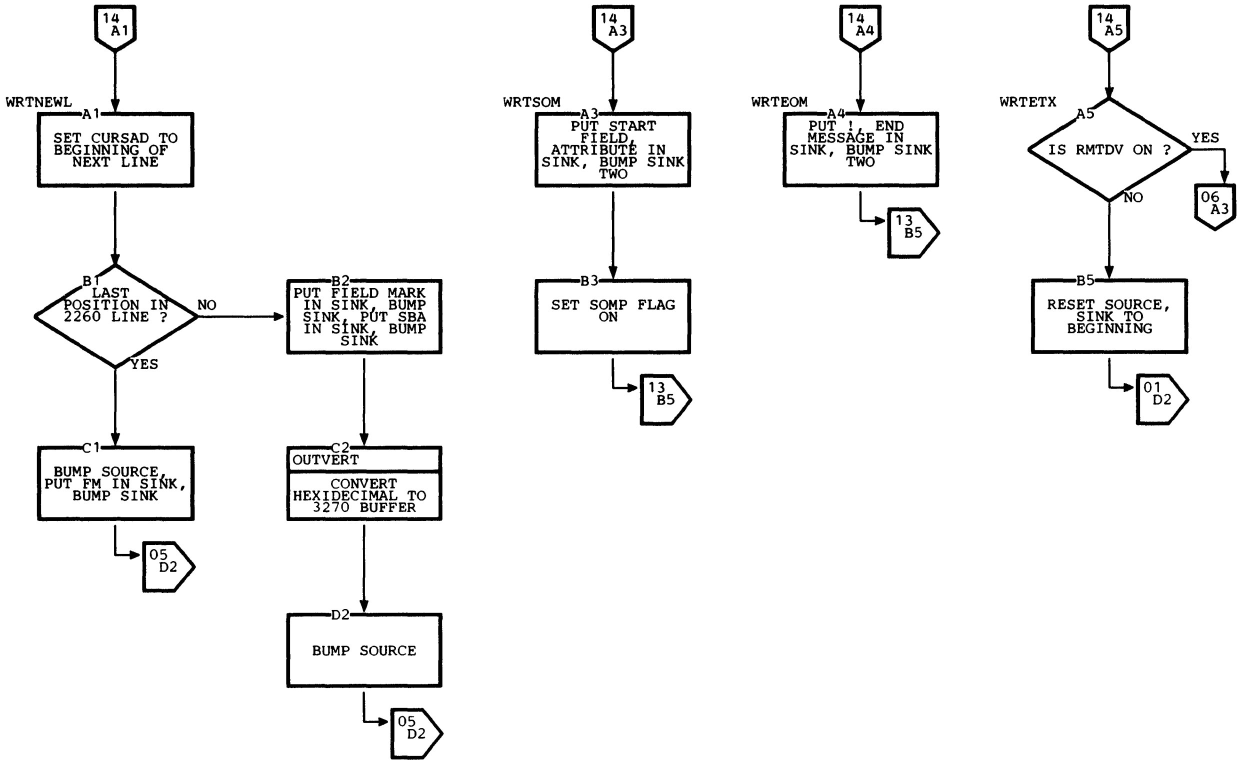

10 A1
A1 INCREMENT SOURCE, GET CURSOR ADDRESS IN CURSAD
B1 INVERT
CONVERT 3270 BUFFER TO HEXADECIMAL
C1 CURSOR ON 2260 ?
NO
YES
OUTCURS
C2 PUT RETCONST IN REG 15
03 H3
RDSBATST
D1 SAVE CURSAD TEMPORARILY IN SOMAOD
E1 SBA NEXT IN SOURCE ?
NO
YES
F1 BUMP DACIND BY THREE, SET SOMHER ON, GET FIRST DATA ADDRESS
G1 INVERT
CONVERT 3270 BUFFER TO HEXIDECIMAL
RDOPTST
H1 SAVE SOM ADDRESS IN SPECAR, RESTORE CURSAD
J1 CALCULATE OFFSET TO FIRST INTEGRAL LINE BOUNDRY, BUMP DACIND BY OFSET
K1 READ MI OPERATION ?
NO
YES
A3
K2 SET SHRD ON
A3
A3 SET SHRD OFF
B3 IS SOMHER ON ?
NO
YES
C3 SET LINGRO TO ZERO
10 D3
SOURTST
D3 SAVE SOM ADDRESS IN SOMADD
E3 CURSOR ADDRESS RELATIVE TO SOM ?
EQ
LOW
HI
09 H2
F3 CALCULATE TOTAL SCREEN DIFFERENCE BETWEEN SOM AND CURSOR
G3 TEMPORARILY SAVE VALUE IN CARADD
H3 PICK UP DACIND AND DECREMENT BY THREE
J3 SOMHER ON ?
NO
YES
A5
K3 DECREMENT MODIFIED DACIND BY THREE
A5
10 B4
CKSMI
B4 CALCULATE REMAINING SOURCE LENGTH - NOT INCLUDING AID, SBA, ETC.
C4 LENGTH LESS THAN 256 ?
YES
NO
12 A2
D4 SET LINGRO TO 256
12 C2
WRAP
E4 INCREMENT CURSAD BY 3270 SCREEN SIZE, SET LINGRO NON ZERO
A5
REMOTST
A5 RMTDV ON ?
NO
YES
B5 DECREMENT MODIFIED DACIND BY TWO
C5 STXHER ON ?
NO
YES
D5 DECREMENT MODIFIED DACIND BY ONE
FIXIT
E5 INCREMENT DIFFERENCE BY OFFSET FROM SOM TO FIRST LINE BOUNDRY
F5 DIVIDE RESULT BY 3270 SCREEN WIDTH TO GET NUMBER OF LINES SPANNED
G5 COMPUTE DIFFERENCE BETWEEN 2260 AND 3270 SCREEN WIDTHS
H5 MULTIPLY WIDTH DIFFERENCE BY NUMBER OF LINES SPANNED
J5 DECREMENT CARADD BY RESULT
K5 LINGRO = ZERO ?
NO
YES
11 A1
11 G1



11 A1
A1 PROT = YES SPECIFIED ?
NO
YES
B1 SCREENS EQUAL ?
NO
YES
C1 DECREMENT CARADD BY ONE
PROTWRAP
D1 CALCULATE LINE DIFFERENCE BETWEEN 2260 AND 3270
E1 MULTIPLY BY 2260 WIDTH
F1 DECREMENT CARADD BY RESULT
11 G1
NOWRAP
G1 STREAM OVER 255 BYTES ?
NO
YES
UND256
G2 SET LINGRO TO ACTUAL DATA LENGTH
H1 SET LINGRO TO 255, TURN ON LNGDT
OVR256
J1 CLEAR CURLIN
11 K1
RDLAT
K1 ANY UNIQUE CHARACTERS ?
YES
NO
13 A2
A3 SAVE THE BRANCH ARGUMENT, SAVE SOMADD
B3 SOMADD TO REMAINING PORTION OF STREAM
C3 LINGRO TO TESTED PORTION LESS ONE
D3 BUMP SOM BY LINGRO
E3 SINK HOLD LINGRO ?
NO
03 E2
YES
F3 LINGRO = ZERO ?
YES
NO
G3 MOVE TESTED PORTION LESS UNIQUE CHARACTER TO SINK
H3 BUMP SINK BY LINGRO, SOURAD EQUAL UNIQUE CHARACTER'S ADDRESS
J3 SAVE SOM ADDRESS IN CARADD, RESTORE THE BRANCH ARGUMENT
K3 IS THIS A READ BUFFER ?
NO
YES
13 A1
A4 WHAT IS CHARACTER ?
NL ---->15A1
NULL---->15A1
FM ---->15A3
SMI ---->16A1
! ---->16A3
EOM ---->15A1
ETX ---->16A4
SOM ---->16A5


12 A2
UNDER256
A2
SET LINGRO TO ACTUAL LENGTH
B2
IS LINGRO ZERO ?
YES
NO
12 C2
SMILOOP
C2
ANY SPECIAL CHARACTERS ?
NO
YES
C2
NOSTART1
C3
INCREMENT SOURAD BY LINGRO
D2
SET LINGRO TO UNTESTED PORTION
ENDAT
E1
RMTDV ON ?
YES
NO
F3
E2
IS IT AN ETX ?
YES
NO
F1
RESTORE SOURCE, SINK TO BEGINNING
01 D2
F2
IS IT AN SMI ?
NO
YES
F3
NOTSMI
F3
IS LINGRO ZERO ?
YES
NO
C2
NOSTART
F4
SOURAD POINT TO LAST BYTE ?
NO
YES
10 B4
G2
COMPUTE BUFFER POSITION OF SMI
G4
TURN ON NOSMI FLAG
09 H2
H2
IS RMTDV ON ?
NO
YES
B5
J2
DECREMENT BY TWO TO REMOTE ADDRESS
K2
IS STXHER ON ?
NO
YES
B5
A5
A5
A5
DECREMENT BY ONE FOR STX
B5
ADJUST
B5
IS SMI IN FIRST LINE ?
YES
NO
C5
DIVIDE DISPLACEMENT OF SMI INTO SOURCE BY DSWIDE
D5
MULTIPLY QUOTIENT BY ANWIDE-DSWIDE TO GET NUMBER OF LOST NULLS
E5
ADD NUMBER OF LOST NULLS TO DISPLACEMENT INTO SOURCE OF SMI
ADJUST1
F5
CLEAR LINGRO TO ZERO
10 D3


13 A1
A1
WHAT IS CHARACTER ?
NO ---->15A1
NULL---->15A1
FM ---->16E1
SMI ---->16E2
! ---->16E3
EOM ---->15A1
ETX ---->16A4
SOM ---->16E4
13 A2
RNOCHAR
A2
BUMP SOM ADDRESS BY LINGRO
B2
SINK HOLD LINGRO ?
NO
03 E2
YES
C2
MOVE THE TESTED GROUP TO THE SINK, BUMP THE SINK BY LINGRO
13 D2
MORTST
D2
IS LNGDT OFF ?
NO
YES
D3
256 BYTES LEFT ?
NO
YES
LOW256
D4
TURN OFF LNGDT IN DACFLG
11 K1
E2
READ BUFFER OPTYPE ?
NO
09 H2
YES
E3
SET LINGRO TO 255
11 K1
F2
POINT SOURAD TO BEGINNING OF NEXT 2260 LINE
G2
INCREMENT CURLIN BY ONE
09 H1
13 A5
WRTNLL
A5
PUT BLANK IN SINK, BUMP SINK
13 B5
SETTLE
B5
BUMP SOURCE, INCREMENT CURSAD, SET LINGRO TO REMAINING PART
C5
IS SOMP ON ?
NO
YES
D5
SET SOMADD TO BUFFER ADDRESS OF SF, SET SOMP OFF, SET SOMHER ON
E5
IS LINGRO = 0 ?
NO
03 B1
YES
05 D2


14 A1
WRTNEWL
A1
SET CURSAD TO BEGINNING OF NEXT LINE
B1
LAST POSITION IN 2260 LINE ?
NO
YES
B2
PUT FIELD MARK IN SINK, BUMP SINK, PUT SBA IN SINK, BUMP SINK
C1
BUMP SOURCE, PUT FM IN SINK, BUMP SINK
05 D2
C2
OUTVERT
CONVERT HEXIDECIMAL TO 3270 BUFFER
D2
BUMP SOURCE
05 D2
14 A3
WRTSOM
A3
PUT START FIELD, ATTRIBUTE IN SINK, BUMP SINK TWO
B3
SET SOMP FLAG ON
13 B5
14 A4
WRTEOM
A4
PUT !, END MESSAGE IN SINK, BUMP SINK TWO
13 B5
14 A5
WRTETX
A5
IS RMTDV ON ?
YES
06 A3
NO
B5
RESET SOURCE, SINK TO BEGINNING
01 D2

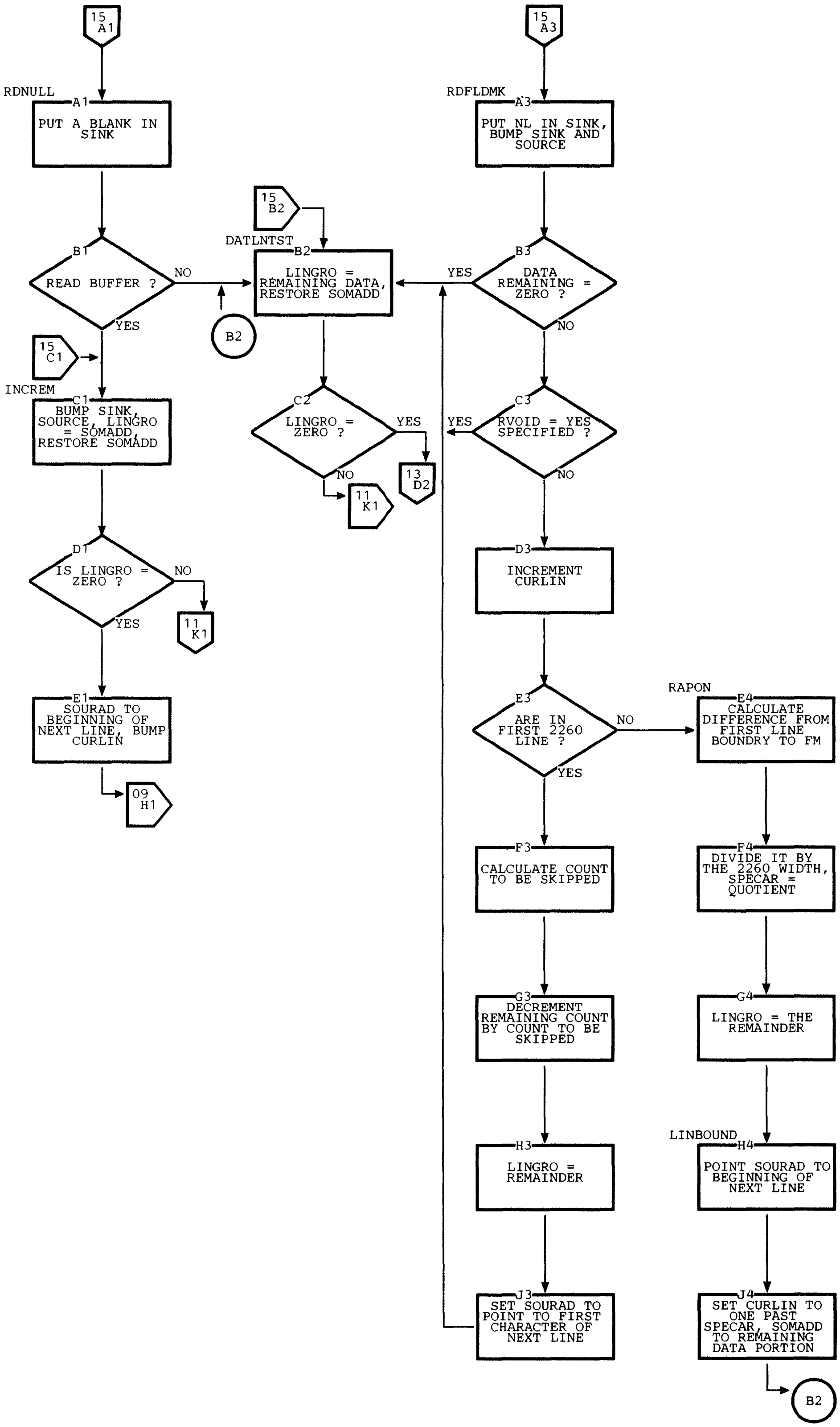
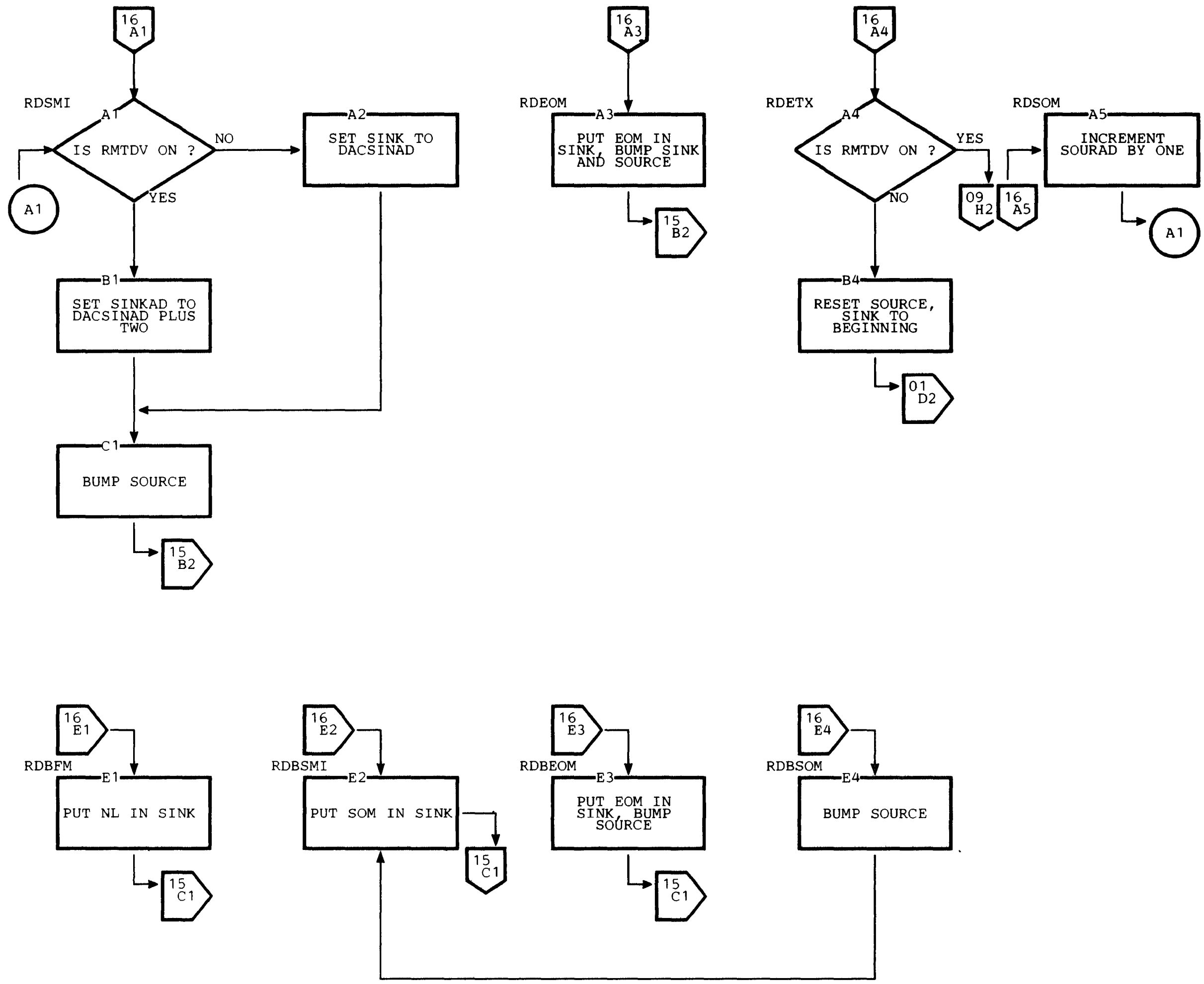
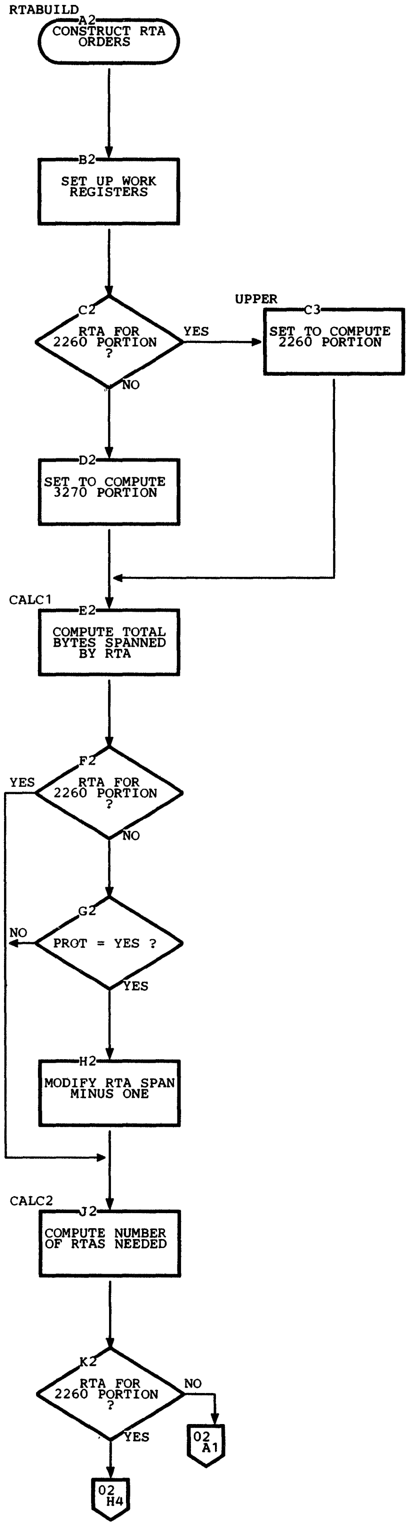
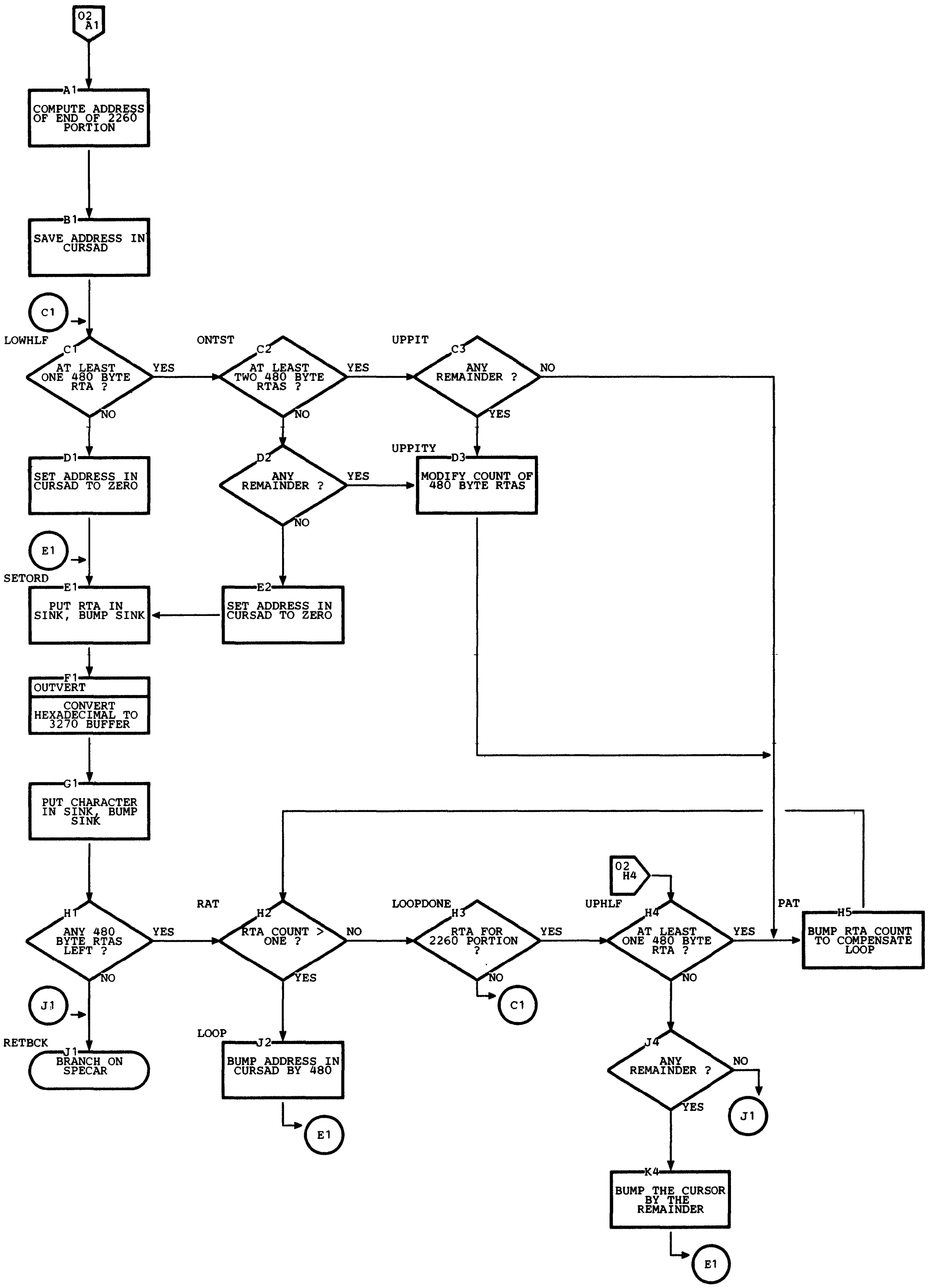

15 A1
RDNULL
A1
PUT A BLANK IN SINK
B1
READ BUFFER ?
NO
YES
15 C1
INCREM
C1
BUMP SINK, SOURCE, LINGRO = SOMADD, RESTORE SOMADD
D1
IS LINGRO = ZERO ?
NO
11 K1
YES
E1
SOURAD TO BEGINNING OF NEXT LINE, BUMP CURLIN
09 H1
15 B2
DATLNTST
B2
LINGRO = REMAINING DATA, RESTORE SOMADD
B2
C2
LINGRO = ZERO ?
YES
13 D2
NO
11 K1
15 A3
RDFLDMK
A3
PUT NL IN SINK, BUMP SINK AND SOURCE
B3
DATA REMAINING = ZERO ?
YES
NO
C3
RVOID = YES SPECIFIED ?
YES
NO
D3
INCREMENT CURLIN
E3
ARE IN FIRST 2260 LINE ?
NO
YES
F3
CALCULATE COUNT TO BE SKIPPED
G3
DECREMENT REMAINING COUNT BY COUNT TO BE SKIPPED
H3
LINGRO = REMAINDER
J3
SET SOURAD TO POINT TO FIRST CHARACTER OF NEXT LINE
RAPON
E4
CALCULATE DIFFERENCE FROM FIRST LINE BOUNDRY TO FM
F4
DIVIDE IT BY THE 2260 WIDTH, SPECAR = QUOTIENT
G4
LINGRO = THE REMAINDER
LINBOUND
H4
POINT SOURAD TO BEGINNING OF NEXT LINE
J4
SET CURLIN TO ONE PAST SPECAR, SOMADD TO REMAINING DATA PORTION
B2

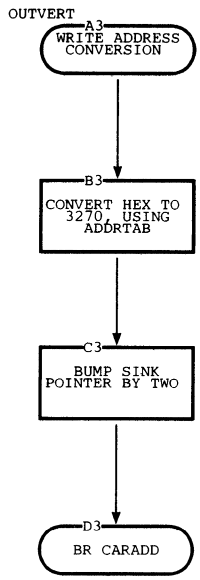
Chart Z. 3270 SCAN (IECTSCAN) (Part 16 of 19)

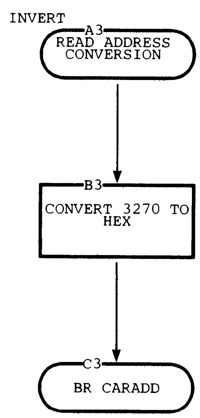



RTABUILD
A2
CONSTRUCT RTA ORDERS
B2
SET UP WORK REGISTERS
C2
RTA FOR 2260 PORTION ?
YES
NO
UPPER
C3
SET TO COMPUTE 2260 PORTION
D2
SET TO COMPUTE 3270 PORTION
CALC1
E2
COMPUTE TOTAL BYTES SPANNED BY RTA
F2
RTA FOR 2260 PORTION ?
YES
NO
G2
PROT = YES ?
NO
YES
H2
MODIFY RTA SPAN MINUS ONE
CALC2
J2
COMPUTE NUMBER OF RTAS NEEDED
K2
RTA FOR 2260 PORTION ?
NO
YES
02 A1
02 H4


02 A1
A1
COMPUTE ADDRESS OF END OF 2260 PORTION
B1
SAVE ADDRESS IN CURSAD
C1
LOWHLF
C1
AT LEAST ONE 480 BYTE RTA ?
YES
NO
ONTST
C2
AT LEAST TWO 480 BYTE RTAS ?
YES
NO
UPPIT
C3
ANY REMAINDER ?
NO
YES
D1
SET ADDRESS IN CURSAD TO ZERO
D2
ANY REMAINDER ?
YES
NO
UPPITY
D3
MODIFY COUNT OF 480 BYTE RTAS
E1
SETORD
E1
PUT RTA IN SINK, BUMP SINK
E2
SET ADDRESS IN CURSAD TO ZERO
F1
OUTVERT
CONVERT HEXADECIMAL TO 3270 BUFFER
G1
PUT CHARACTER IN SINK, BUMP SINK
H1
ANY 480 BYTE RTAS LEFT ?
YES
NO
RAT
H2
RTA COUNT > ONE ?
NO
YES
LOOPDONE
H3
RTA FOR 2260 PORTION ?
YES
NO
C1
02 H4
UPHLF
H4
AT LEAST ONE 480 BYTE RTA ?
YES
NO
PAT
H5
BUMP RTA COUNT TO COMPENSATE LOOP
J1
RETBCK
J1
BRANCH ON SPECAR
LOOP
J2
BUMP ADDRESS IN CURSAD BY 480
E1
J4
ANY REMAINDER ?
NO
YES
J1
K4
BUMP THE CURSOR BY THE REMAINDER
E1

SCAN - CONVERSION OF OUTGOING ADDRESSES TO 3270 BUFFER





The System/360 Operating System provides interfaces among programs by means of control blocks and tables. These blocks have standardized formats. They contain numerous fields of information and references by the program. Some of these fields are pointers to other blocks. Figure 13 shows the various blocks and the linkages pertinent to BTAM.

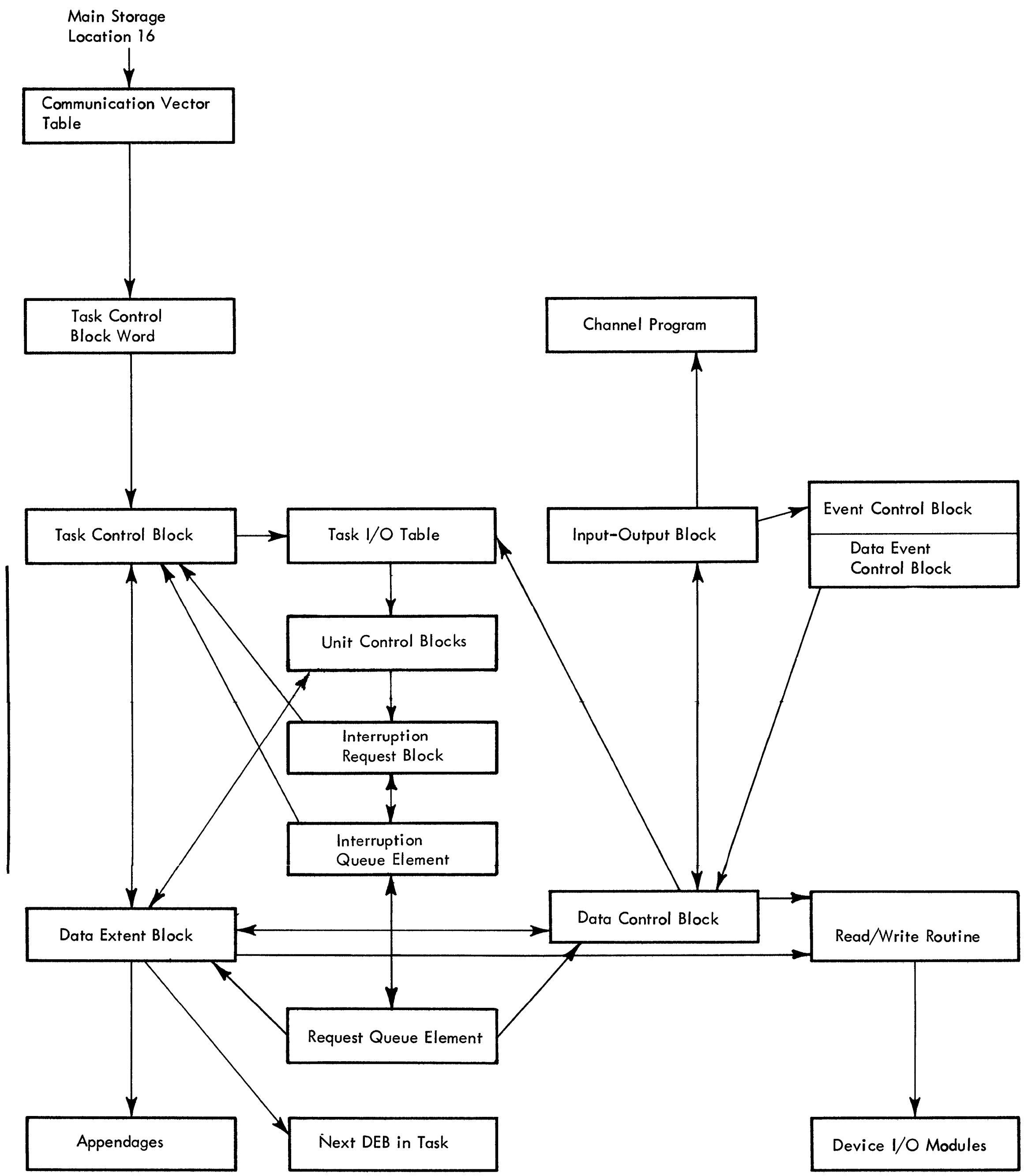


Figure 13. Control Block Linkages

The data control block (DCB) provides information about its associated data set (in this case, a communication line group). Each such data set has a DCB. Expansion of the DCB macro-instruction at assembly time reserves storage for the DCB and initializes the DCB with parameters from the macro defining the DD name, the data set organization, the macros used for input/output, and the exit list address.

Other parameters can be defined by the DD cards in the job stream, or dynamically by the program modules at any time before the data set of the DCB is opened. (There must be at least one DD card for each DCB in the program.)

Figure 14 is a diagram of the DCB. Table 1 shows the contents and possible sources of each field.

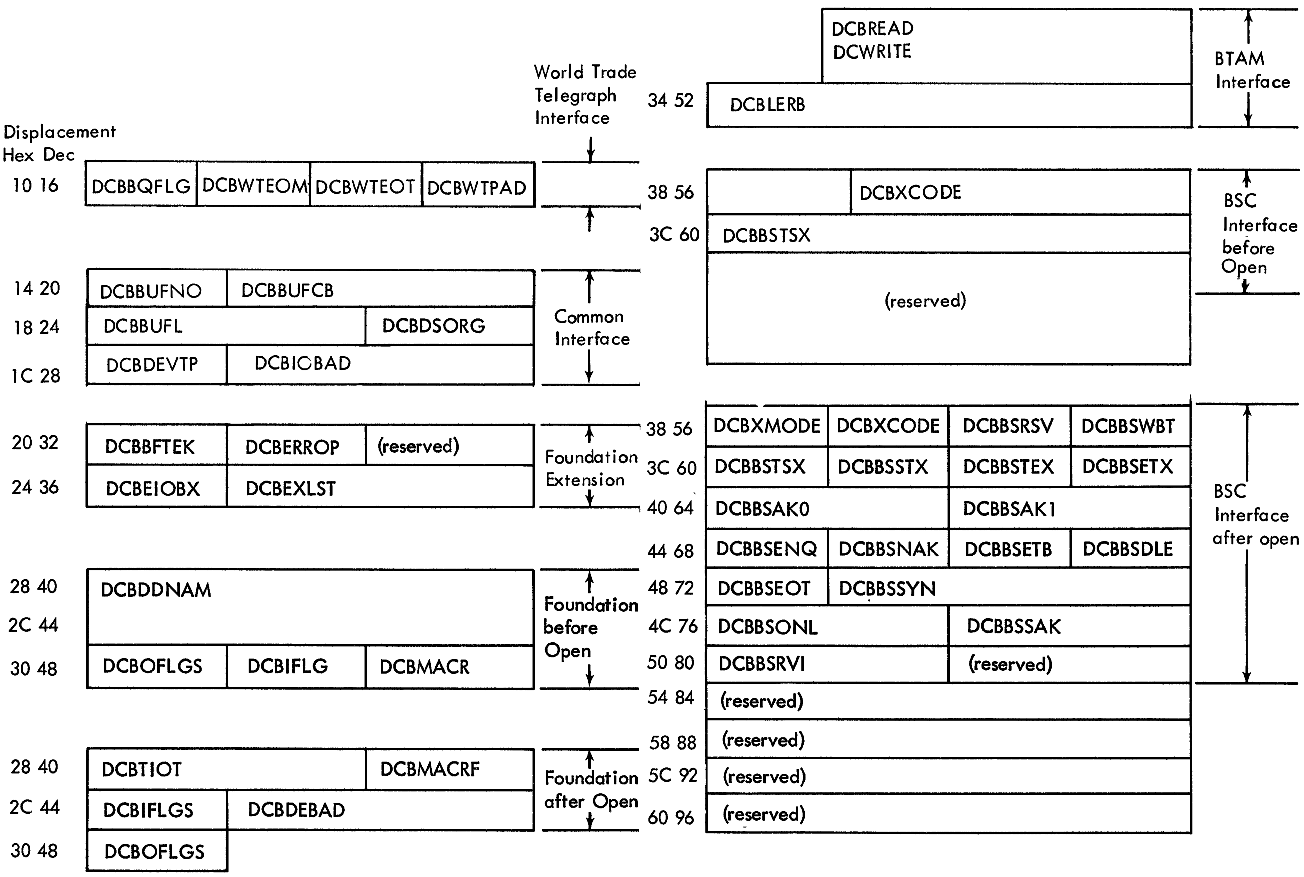


Figure 14. Data Control Block (DCB)

Table 1. Data Control Block Fields (Part 1 of 3)

Section	Relative Loc IHADCB+ (DEC)	Length (in bytes)	Name DCB+	Contents
World Trade Telegraph Interface	0	16		(Reserved)
	16	1	BQFLG	World Trade telegraph flag byte
	17	1	WTEOM	The EOM character (WT telegraph terminals)
	18	1	WTEOT	The EOT character (WT telegraph terminals)
	19	1	WTTPAD	Number of pad (LTRS) characters required for Motor-On delay (WT telegraph terminals)
Common Interface	20	1	BUFNO	Number of buffers to be obtained by Open (0-255).
	21	3	BUFCB	Address of buffer control block.
	24	2	BUFL	Buffer length (the length of buffers to be obtained by open for a buffer pool, and/or the length to be used if the length parameter of a READ or WRITE macro is coded as 'S') 0-32760.
	26	2	DSORG	Data set organization (a communication line-group specification of CX sets bit 3)
	28	1	DEVTP	Index to device entry in device I/O directory.
	29	3	IOBAD	IOB Address (the address of the first IOB, the IOB for line 1, minus the length of an IOB (DCBEIOBX). The IOB address for any line associated with this DCB is equal to DCBIOBAD plus the product of the line number and DCBEIOBX value.)
Foundation Extension	32	1	BFTEK	Buffering technique (Bit 4-dynamic buffering).

Table 1. Data Control Block Fields (Part 2 of 3)

Section	Relative Loc IHADCB +	Length (in bytes)	Name DCB+	Contents
Foundation Extenstion (Continued)	33	1	ERROP	Error recovery procedures. The EROPT parameter inthe DCB will cause one or more of the following bit codes to be set. Bit / Bit Value / Code in ERROPT and Meaning 3 / 1 / T Specifies that the on-line test facility is to be used for the line group. This option is valid for all IBM stations, with or without error recovery procedures. 4 / 1 / C Specifies that threshold error counts and cumulative error counts are to be maintained in the line error recording block (LERB) for data check, intervention required, and non-text timeout errors. 5 / 1 / W Specifies that text-write errors are to be retried in addition to the basic error recovery procedures. This option is valid for all start-stop terminals except World Trade telegraph terminals. It is invalid for BSC stations. It results in an additional copy of the message for each retry (except for the 2260 with the line address feature and the 1050 card punch and paper tape punch with the line correction feature). This parameter is ignored for BSC stations and WT telegraph terminals. 6 / 1 / R Specifies that text-read errors are to be retried in addition to the basic error recovery procedures. This option is valid only for the following terminals: 1050 terminals (valid for the card reader and paper tape reader only if the line correction feature is installed), 2740 terminals with the checking feature, and 2260 terminals. 7 / 1 / N Specifies that no error recovery procedures are to be provided for the line group. This parameter and E,R,W, and C are mutually exclusive. This parameter is invalid for BSC stations; if coded, it is ignored. 7 / 0 / E Specifies that the basic error recovery procedures (ERP) are to be provided for the line group. If EROPT is omitted, E is assumed.
	34	1	BUFCT	Maximum number of buffers to be obtained by BTAM for a Read operation (dynamic buffering).
	35	1		RJE requesting no retries.
	36	1	EIOBX	Extended IOB index (size of IOB's associated with this DCB).
	37	3	EXLST	Exit list (the address of a user-provided list that may contain an entry, control code and address, for a DCB exit).
Foundation before Open	40	8	DDNAM	Data definition name as used in DD statement (used by open to locate the appropriate DD entry in the task I/O table).
	48	1	OFLGS	Flags used by OPEN. Bit 3 - OPEN: This bit set on when an OPEN has been successfully completed. Bit 6 - LOCK: This bit is set on by an I/O support function if the DCB is to be processed by that function.
	49	1	IFLG	Used by IOS in communicating error conditions and in determining error procedures.
	50	2	MACR	Macro instruction reference (specifies major macros and various options associated with them. Used by Open to determine access method. Not used by BTAM). Bit 2 - Read } Bit 10 - Write } BTAM
Foundation after Open	40	2	TIOT	Points to the DD entry in the task I/O table for this DCB. It is the offset of the DD entry relative to the beginning of the task I/O table.
	42	2	MACRF	Same as MACR.
	44	1	IFLGS	Same as IFLG.
	45	3	DEBAD	Address of the associated DEB.
	48	1	OFLGS	Same as OFLGS above.
BTAM interface	49	3	READ	} Address of Read-Write module.
	49	3	WRITE	}
	52	4	LERB	Address of line error recording block. The LERB address is specified by the DCB parameter LERB. This field is present if C or T is specified in the DCB parameter EROPT.

Table 1. Data Control Block Fields (Part 3 of 3)

Section	Relative Loc IHADCB +	Length (in bytes)	Name DCB+	Contents
BSC Interface (after Open)*	56		XMODE	Mode of transmission for binary synchronous communication (BSC). Bit / Bit Value / Meaning 0 — Reserved. 1 / 1 — Intermediate block checking is to be performed. The transmission control unit will recognize the EIB character and perform block checking without turning the line around. 2 / 1 — Transmission is through a 2701 Data Adapter Unit Dual Communication Interface B. 3 — Reserved. 4 / 1 — Transmission is in code B for a 2701 Data Adapter Unit Dual Code Feature. 5-7 — Reserved.
	57	1	XCODE	BSC control station flag and transmission code. Bit & State 0123 4567 0... .... This is the control station. 1... .... This is the remote station. .... 00.. EBCDIC transmission code is used. ..1. 1... 6-bit TRANSCODE is used. ...1 .1.. USASCII is used. .1.. ..XX A synchronous exit is to be taken to processor-to-processor (PTOP) FORTRAN data processing routine (before open only)
	58	1	BSRSV	DLE control character.
	59	1	BSWBT	(reserved)
	60	1	BSTSX	DLE control character (before Open, BSTSX is a 4-byte field containing address of PTOP interface resolution routine, if PTOP is specified at system generation).
	61	1	BSSTX	STX control character.
	62	1	BSTEX	DLE control character.
	63	1	BSETX	ETX control character.
	64	2	BSAK0	ACK-0 control character.
	66	2	BSAK1	ACK-1 control character.
	68	1	BSENQ	ENQ control character.
	69	1	BSNAK	NAK control character.
	70	1	BSETB	ETB control character.
	71	1	BSDLE	DLE control character.
	72	1	BSEOT	EOT control character.
	73	1	BSSYN	SYN control character.
	74	1	BSTBE	DLE control character.
	75	1	BSTEB	ETB control character.
	76	2	BSONL	SOH% control sequence.
	78	2	BSSAK	WACK control sequence (X'106B'-EBCDIC, X'103B'-USASCII)
	80	2	BSRVI	DLE @ control sequence.
	82	18	---	(reserved)

*Before Open, BSC interface contains DCBXCODE and DCBBSTSX fields only.

One data extent block (DEB) is created in protected storage by open module 1 for each line group (data set). The DEB contains a table of IOS appendage addresses, unit control block addresses for each line or local 3270 device, and addresses of other control blocks. The DEB also includes a list of the identifications of all BTAM modules needed to support the devices in the line group. Figure 15 is a diagram of the DEB. Table 2 shows the contents of each field, giving source codes where known.

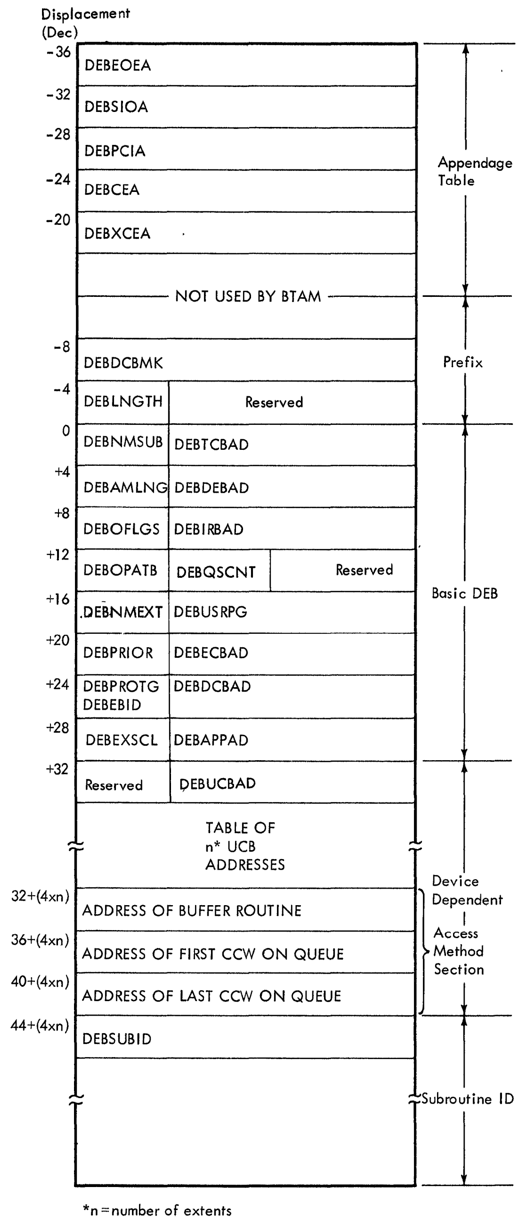


Figure 15. Data Extent Block (DEB)

Table 2. BTAM DEB Fields

Section	Source	Location IECTDEB+	Size (in bytes)	Name DEB+	Contents
Appendage Table	F	-36	4	EOEA	Address of end of extent appendage branched to by IOS (not used by BTAM).
	F	-32	4	SIOA	Address of start I/O appendage branched to by IOS (not used by BTAM).
	G	-28	4	PCIA	Address of program controlled interrupt appendage branched to by IOS.
	G	-24	4	CEA	Address of channel end appendage branched to by IOS.
	G	-20	4	XCEA	Address of abnormal end appendage branched to by IOS.
Prefix	D	-8	4	DCBMK	DCB modification mask used by I/O Support.
	F	-4	1	LNGTH	Length of DEB in doublewords.
Basic	G	0	1	NMSUB	Number of subroutines loaded by open executor.
	F	1	3	TCBAD	TCB address for this DEB.
	F	4	1	AMLNG	Number of bytes in Access Method Section.
	F	5	3	DEBAD	Address of next DEB in the same task.
		8	1	OFLGS	Data set status flag (not used by BTAM).
		9	3	IRBAD	IRB address for error exit (not used by BTAM).
		12	1	OPATB	Indicates file type (not used by BTAM).
		13	3	QSCNT	Number of devices executing user channel programs.
	F	16	1	NMEXT	Number of extents (number of lines in the line group).
		17	3	USRPG	Address of first IOB in user purge chain.
		20	1	PRIOR	Priority of the task.
		21	3	ECBAD	Address of a parameter list used to locate the purge ECB for an SVC purge request.
	D	24	1/2	PROTG	Protection tag assigned to this task.
	F	24 1/2	1/2	DEBID	Hex "F" identifies this block as a DEB.
	F	25	3	DCBAD	Address of DCB associated with this DEB.
	F	28	1	EXSCL	Extent scale = 2 for communication devices indicating 4 bytes per extent (used to determine size of device dependent section).
	F	29	3	APPAD	Address of I/O appendage table preceding DEB.
Device Dependent	F	32	(Note 1)	UCBAD	Table of addresses of UCBs for each of n* lines or local 3270 devices.
Access Method Dependent		32+(4xn)	4		Address of buffer routine (loaded when dynamic buffering or buffer pool specified).
		36+(4xn)	4		Address of first CCW on queue (optional; required only when dynamic buffering is specified).
		40+(4xn)	4		Address of last CCW on queue (optional; required only when dynamic buffering is specified).
Subroutine ID	G	44+(4xn)	(Note 2)	SUBID	2-character subroutine IDs (last two characters of 8-byte name).

*n = number of extents.
Note 1: Size = NMEXT shifted left logical EXSCL bits (in this case, 4 bytes/extent).
Note 2: Size = 2 X NMSUB bytes.

APPENDIX D: INPUT/OUTPUT BLOCK

The input/output control block (IOB) provides communication between BTAM and IOS. It is the sole parameter of the IOS execute channel program (EXCP) instruction. One IOB is created for each communication line by open module 1.

The basic IOB, 64 bytes in length, contains pointers to the channel program, the event control block, and the terminal lists; it provides areas for storing flags, sense bytes, the channel status word, and the start I/O condition code returned by IOS. Appended to each basic IOB is a variable length area where the channel programs are constructed by the read-write routine.

Figure 16 shows details of the channel status word (CSW) field and Figure 17 is a diagram of the IOB. Table 3 contains descriptions and sources of the contents of the IOB fields.

0 – 7	8 – 31	
IOS Flags	CCW ADDRESS	
Control Unit and Device Status	Channel Status	Count
32 – 39	40 – 47	48 – 63

	Bits	
	8-31	Address of Last CCW+8
Control Unit and Device Status	32	Attention
	33	Status Modifier
	34	Control Unit End
	35	Busy
	36	Channel End
	37	Device End
	38	Unit Check
	39	Unit Exception
Channel Status	40	PCI
	41	Incorrect Length
	42	Program check
	43	Protection check
	44	Channel data check
	45	Channel control check
	46	Interface control check
	47	Chaining check
Count	48-63	Residual count difference between number of characters sent or received and length specified.

Figure 16. IOB CSW Field

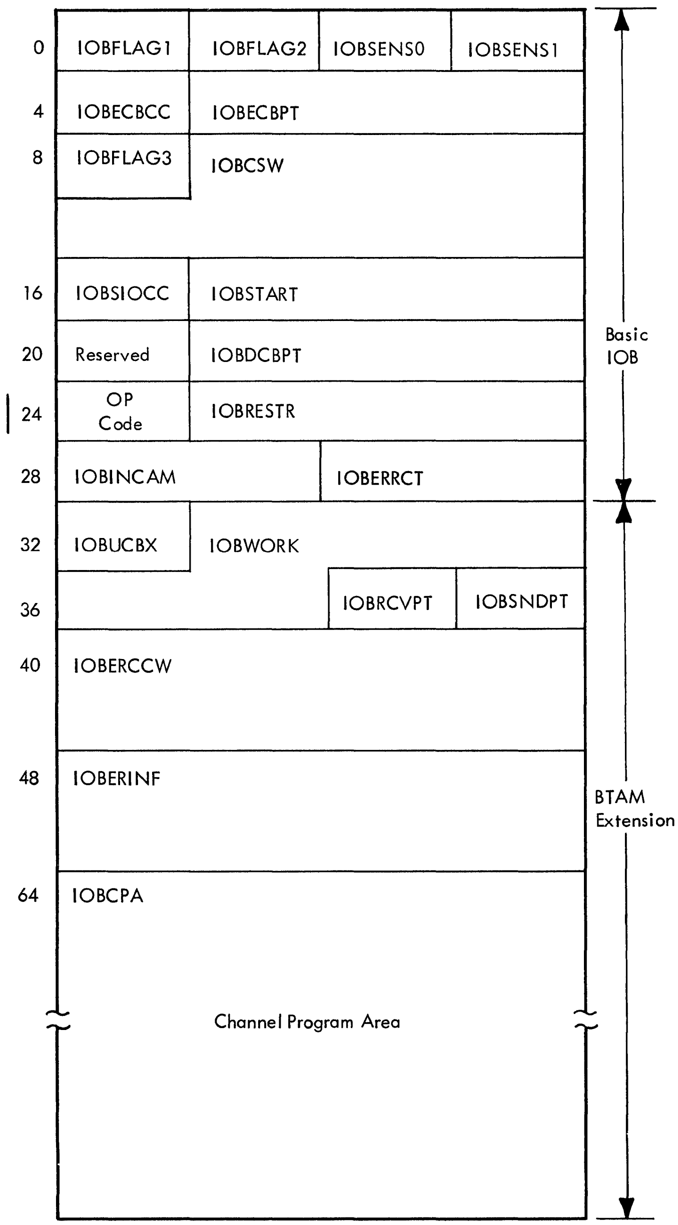


Figure 17. Input/Output Control Block (IOB)

Table 3. BTAM IOB Fields

Relative Location	Length (in bytes)	Name IOB+	Contents
0	1	FLAG1	Flag byte 1 Bit 0 - Data chaining in channel program 1 - Command chaining in channel program. 2 - Error recovery procedure is in control of current operation. 3 - (not used by BTAM) 4 - (not used by BTAM) 5 - Exceptional condition. After ERP routine returns and this bit is on, the error is considered permanent. 6 - Unrelated (I/O requests need not be scheduled FIFO). (This bit is always on for BTAM.) 7 - Start/restart (IOS is to use channel program address IOBSTART (0) or IOBRESTR (1) (This bit is always 0 for BTAM, unless the IOB is for a local 3270 device.)
1	1	FLAG2	Flag byte 2 Bit Meaning 0, 1, 2 Not used by BTAM. 3 Extended message flag (indicator for IOS message writer routine) 4-6 Not used by BTAM. 7 RESETPL macro instruction was used.
2	1	SENS0	One byte of sense data stored here when a unit check error occurs. All zeros means successful completion. Bit 0 - Command reject 1 - Intervention required 2 - Bus out check 3 - Equipment check 4 - Data check 5 - Overrun 6 - Lost data 7 - Time-out
3	1	SENS1	Second sense byte (local 3270 display system only). If the sense data for IOBSENS0 could not be obtained for a local 3270 device, this byte contains X'FE'.
4	1	ECBCC	Completion code for an I/O event as it will appear in the first byte of the ECB.
5	3	ECBPT	Address of event control block (ECB) associated with this I/O request.
8	1	FLAG3	IOS error routine flags.
9	7	CSW	The channel status word is stored here at channel end time (low-order 7 bytes).
16	1	SIOCC	The condition code from execution of start I/O is stored here.
17	3	START	Address of the first CCW at which to start I/O for normal conditions.
20	1		Reserved
21	3	DCBPT	Address of the DCB associated with this IOB.
25	3	RESTR	Address of CCW at which to start I/O for restart operations.
28	2	INCAM	Bit Meaning Byte 0: 0 SAD or ENABLE issued by OPEN resulted in a permanent I/O error. If the IOB is for a local 3270 device, this bit indicates that OPEN did not initialize the device, because it was being used by OLTEP. 1 This IOB is currently in use by an I/O operation (IOB busy). 2 RVI was received. Turned on when RVI sequence is received. Turned off when the next proper positive acknowledgment (ACK-0 or ACK-1) is received. 3 Turned on, after a remote 3270 RFT message has been received, to indicate that the message cam from a control unit capable of a general poll. 4 Reserved. 5 Turned on after a remote 3270 error status message has been processed. (The error status message begins with SOH % R and provides status information about a remote 3270 device.) 6 Reserved. 7 Line is currently under on-line test operation. Byte 1: Used by BTAM for timer value (OPEN and LOPEN).
30	2	ERRCT	Error counter (local 3270 display system only). Byte 0: The local 3270 ERP uses this byte to keep a count of retry attempts. Byte 1: Bits 0-4 - The local 3270 ERP uses these bits as flags during retry attempts. Bits 5-7 State Meaning 000 If bit 5 of IOBFLAG1 is off, the local 3270 ERP has recovered from the error. If bit 5 of IOBFLAG1 is on, the local 3270 ERP has not processed the error. 010 The local 3270 ERP could not recover from the error. The problem program must reconstruct the buffer image. 011 The local 3270 ERP determined that the error is permanent and nonrecoverable. 110 The local 3270 ERP determined that the channel program or data stream builder of the problem program can recover from the error. 111 The local 3270 ERP determined that the local 3270 channel end/abnormal end appendage should perform retry. 101 The local 3270 channel end/abonormal end appendage is attempting to recover from the error by retrying the channel program.
32	1	UCBX	UCB index (line number-used as index to appropriate UCB address in the DEB).
33	5	WORK	Work area used by BTAM Error routines and On-Line Test Routines.
38	1	RCVPT	Received ACK (ACK-0 or ACK-1).
39	1	SNDPT	Sent ACK (ACK-0 or ACK-1).
40	8	ERCCW	CCW area used by BTAM error recoverv routines. } The contents of these fields vary unpredictably
48	16	ERINF	Error information field used by the BTAM error recovery routines.
64		CPA	Channel program area (length depends on type of stations or line configuration).

After initiating an I/O operation, the task (user program) in control can continue processing until it needs the results of that operation. At this point, the task issues a WAIT macro instruction, which signals the supervisor that the task cannot proceed until completion of a specified event (i.e., the I/O operation). The WAIT specifies an event control block (ECB) address, which is the first word of the data event control block (DECB) formed by expansion of a READ or WRITE macro instruction. Figure 18 shows the ECB after the WAIT was issued. Bit 0 is the wait flag, and bit 1 is set on completion. Bits 8 through 31 specify the address of the program request block.

0	1	8 –	– 31
1	0		PRB Address

Figure 18. ECB After WAIT

The supervisor posts completion of the event by setting the wait flag off and the completion flag on. It inserts a completion code in bits 2 through 31 (Figure 19).

0	1	2 – 7	8 – 31
0	1	Completion Code	0

Figure 19. ECB After Post

COMPLETION CODES: Upon completion of an I/O operation, bits 2 through 7 of the event control block (ECB) in the DECB contain a completion code specified by the input/output supervisor. Bit 0 (the wait flag) is zero, bit 1 (the completion flag) is one, and bits 8 through 31 are zero. The first byte of the ECB (bits 0 through 7) contain one of the following "completion" codes in hexadecimal notation:

Code	Meaning
7F	Normal Completion. Channel end/device end status; may include either unit exception, incorrect length, or both.
41	I/O Error. All channel end/device end/unit check conditions, plus program check, protection check, or chaining check conditions. This completion code also includes unit exception and incorrect length status if they occur in combination with one of the preceding error conditions. Status is present in the sense byte (IOBSENS0) only if a unit check condition has occurred.
44	I/O Request Rejected. A device error was detected after the last I/O operation on the device was posted complete.
48	Enable Command Halted or I/O Operation Purged. Indicates one of the following:

- An enable command (automatic answering function for a switched-connection line) was terminated by halt I/O as a result of a RESETPL macro instruction (second operand omitted or specified as ANSRING).
- A Read Initial operation for a World Trade telegraph terminal was terminated by a Halt I/O as a result of a RESETPL macro instruction (second operand omitted).
- A Read Initial operation for a local 3270 display station was terminated as a result of a RESETPL macro instruction (second operand omitted or specified as ATTENT).
- An enable command was terminated as a result of closing (CLOSE macro instruction) a line group with enable commands outstanding.
- An I/O operation was purged at channel end interrupt time as a result of closing with I/O operations still in progress.

The data event control block (DECB) is formed in the user program at assembly time by expansion of a READ or WRITE macro instruction of the list or standard form. It provides communication with the BTAM read-write module, specifying operation type, line group, line or local 3270 device, and terminal list. Areas for the standard ECB and responses to addressing and LRC checks are also included in the DECB. The format of the block is shown in Figure 20. The contents of its fields are shown in Table 4.

0	DECSDECB			
4	DECTYPE		DECLNGTH	
8	DECONLTT DECDCBAD			
12	DECAREA			
16	DECSENS0	DECSENS1	DECCOUNT	
20	DECCMCOD	DECENTRY		
24	DECFLAGS	DECRLN	DECRESPN	
28	DECTPCOD	DECERRST	DECCSWST	
32	DECADRPT			
36	DECPOLPT			
40	RESERVED		DECWLNG	PRESENT ONLY IN DECBS FOR BSC LINES
44	DECWAREA			

Figure 20. Data Event Control Block (DECB)

Table 4. BTAM DECB Fields (Part 1 of 4)

Location IECTDECB+	Length (in bytes)	Name DEC +	Contents
0	4	SDECB	Event control block (ECB) for this I/O request (see Appendix E).
4	2	TYPE	Operation types:
			1st byte: bit 0 – Current operation is a read operation using Auto Poll
			bit 1 – if 1, RJE requested WTO time-out message suppression
			bit 4 – if 1, Write Inquiry macro was issued — in any combination
			bit 5 – Entry coded as 'S'
			bit 6 – Area coded as 'S'
			bit 7 – Length coded as 'S'
			2nd byte. bits 3,4,5,6,7 — Value / Operation Type / Operation Type Code
			00 Write Break TB
			01 Read Initial TI
			02 Write Initial TI
			03 Read Continue TT
			04 Write Continue TT
			05 Read Conversational TV
			06 Write Conversational TV
			07 Read Repeat TP
			Read Continue with Identification Exchange TE
			08 Write Positive Acknowledgment TA
			09 Read Skip TS
			0A Write Neg Acknowledgment TN
			Write Disconnect (TWX) TN
			Write Reset (BSC) TR
			0B Read Buffer TB
			0C Write at Line Address TI
			Write Initial Optical (2740 w/Checking & 2760, or w/Checking, Dial-up & 2760) TIO
			Write Initial Transparent Block (BSC)
			0D Write Initial Conversational TIV
			Read Continue with Leading Acknowledgment TTA
			0E Write Erase TS
			Write Initial Optical (2740 w/Checking & 2760, or w/Checking, Dial-up & 2760) TCO
			Write Continue Transparent Block (BSC) TTE
			0F Write Continue Conversational TTV
			10 Write Disconnect TD
			11 Read Connect TC
			Read Modified (Local 3270) TM
			12 Write Initial Transparent TIX
			Write Conversational Optical (2740 w/Checking, Dial-up & 2760) TVO
			Write Unprotected Erase (Local 3270) TUS
			13 Read Continue with Leading Graphics TTL
			Read Buffer from Position (Local 3270) TBP
			14 Write Continue Transparent TTX
			15 Read Inquiry TQ
			16 Write Inquiry TQ
			17 Read Repeat with Leading Graphics TPL
			18 (Reserved)
			19 Read Initial Inquiry TIQ
			Read Modified from Position (Local 3270) TMP
			1A Write Wait Before Transmitting TW
			1B Read Interrupt TRV
			1C Write Connect TC
			1D Write Initial Conversational Transparent TIVX
			1E Read Connect with Tone TCW
			1F Write Continue Conversational Transparent TTVX
			bit 0 – Specifies reset for type code TIR, TTR, TVR, TLR, TSR, TIXR TPR, TTXR.
			Bits 1 and 2 are reserved.
6	2	LNGTH	Buffer length or message area length
8	1	ONLTT	When BSC on-line test is in control, the 0 and 1 bits have the following meanings;
			bit 0 – if 0, on-line test was requested by RFT message, if 1, on-line test was initiated by ONLTST macro
			bit 1 – if 0, test messages are sent by BTAM if 1, test messages are received by BTAM
9	3	DCBAD	Address of the associated DCB.
12	4	AREA	Address of message area or first buffer.
16	1	SENS0	Sense information, as set by the control unit, when the CSW status (DECCSWST) indicates a Unit Check.
			Bit — Meaning
			0 Command reject
			1 Intervention required
			2 Bus out check
			3 Equipment check
			4 Data check
			5 Overrun
			6 Lost data
			7 Timeout
17	1	SENS1	(Reserved)
18	2	COUNT	Residual count from the CSW for the last CCW that was executed.
20	1	CMCOD	Command code, identifies the type of command upon which the error occurred.
21	3	ENTRY	Address of a terminal list.

Table 4. BTAM DECB Fields (Part 2 of 4)

Location IECTDECB+	Length (in bytes)	Name DEC+	Contents
24	1	FLAGS	Status flags which may be set regardless of whether there was an I/O error (i.e., the completion code in the DECSDECB may be either 7F or 41). Bit 0: For start-stop operations, this bit is reserved. For BSC operations, it indicates that a WACK (wait-before-transmit) was received, if bit 1 is also on. If bit 1 is not on, bit 0 indicates that an error status message was received. (An error status message begins with SOH % S and provides status information about a remote station.) Bit 1: For start-stop operations, this bit is reserved. For BSC operations, it indicates that some response other than ACK-0 or ACK-1 was received into the DECRESPN field. Examination of the response will determine which action should be taken to reestablish proper communication. (This bit is set when WACK (see also bit 0) is received or when RVI (see also bit 6) is received. Bit 2: For start-stop operations, this bit is reserved. For BSC operations, it means that an incorrect alternating acknowledgment was received: ACK-1 received when ACK-0 was expected, or vice versa. If this bit is on and the completion code for the operation is 7F (i.e., no line transmission error occurred), a complete message has been lost. Bit 3: The ID received from a TWX 33/35, or a BSC station did not equal the expected ID as defined in the terminal list specified in the WRITE TI, WRITE TC, or READ TC macro instruction, or the index received as a result of an Auto Poll operation did not match the index byte in any of the active entries in the polling list. For BSC (nonswitched line) this bit on indicates that contention has occurred and this is not the control station. For WT telegraph, this bit on indicates that contention occurred, or that a received ID differs from expected ID in terminal list specified by READ TE macro. Bit 4: No buffer was available upon completion of a dynamic buffering read command. The last buffer is posted complete and the remainder of the message is read from the communications line (under control of a dynamic buffering read skip command), but the data is not placed into storage. Bit 5: Indicates one of the following: 1. A negative response to polling has been received from the terminal represented by the last active (nonskip) entry in an open polling list (OPENLST). 2. A negative response to polling has been received following a RESETPL macro instruction of the POLLING type (second operand omitted or specified as POLLING). 3. All of the entries in a wrap around polling list (WRAPLST) are inactive (all the skip bits are on). Note: Condition 3 can occur only as a result of one or more skip bits being turned on after initiation of a programmed polling operation, with a wrap around polling list. If all skip bits were on at the time that the READ macro instruction was executed, no I/O operation would be initiated. 4. Negative response to addressing has been received. 5. The last message sent from a WT telegraph terminal was ended by EOT or a timeout. 6. 2741 power off or other Intervention Required condition. Bit 6: 2741: Write operation was ended by terminal interrupt. WT telegraph: Message ended with WRU signal. BSC stations: A reverse Interrupt (RVI) sequence was received (see also bit 1). Remote 3270: If bit 6 is on, but bit 1 is not on, an error status message was received. (The error status message begins with SOH % R and provides status information about a remote 3270 device.) Bit 7: WT telegraph: The last message sent from a WT telegraph terminal ended with WRU. BSC stations: The STX ENQ sequence was received. Local 3270: OLTEP is using the device to run diagnostics.
25	1	RLN	Relative Line number.
25	2	RESPN	1st byte, response to terminal addressing; 2nd byte, response to LRC/VRC check.

Table 4. BTAM DECB Fields (Part 3 of 4)

Locations IECTDECB+	Length (in bytes)	Name DEC+	Contents
28	1	TPCOD	TP-Op code. Bits 2-7 of these codes identify types of commands that are not identifiable by the command code alone. Bits 0 and 1 are used in conjunction with, but independently of, bits 2-7, as described below.

Hex value	Meaning
00	Any command issued by on-line test routine or for a local 3270 device.
01	Disable, when disable is the first command of a channel program Dial Enable Prepare Write EOA EOT EOT EOT Write graphics Write wait before transmitting Sense (WT telegraph)
02	Write EOA EOT EOT EOT prior to selection Write EOT sequence prior to polling or addressing Write response to text Write EOA and 15 idle characters (Basic 2740) Write EOA PRE o
03	Write polling or addressing characters or /character (/ is the broadcast addressing character) (2740 w/Station Control) Poll Write turnaround sequence (TWX) Write CPU-ID sequence (TWX and BSC) Write ENQ (BSC)
04	Write WRU Write ID Write Pad characters Write letters shift characters (World Trade telegraph terminals). Write space (2740 w/Station Control) Write 2848 command (2260R) Write FIGS shift (83B3) Write 1 (1030) Sense (2740)
05	Read response to polling
06	Read response to addressing
07	Read ID response (TWX, BSC, WT Telegraph)
08	Write end of addressing character following addressing (1030, 1050, 1060, 2260R, 2740) Write response to inquiry (BSC) Write response to text (BSC) Write EOB (2760)
09	NOP or TIC following Poll in READ with SSALST, SSAWLST, AUTOLST, or AUTOWLST

Table 4. BTAM DECB Fields (Part 4 of 4)

Location IECTDECB+	Length (in bytes)	Name DEC+	Contents
28	1	TPCOD	(Continued) Hex Value / Meaning 0A Read Index (Auto Poll) Read response to polling (programmed polling) 0B Read inquiry (BSC) 0C Read response to inquiry (BSC) 10 Write at line address (2260R) 11 Read or write text; Write frame change sequence (2760) 12 Read skip or TIC for dynamic buffering 13 Write end of transparent text characters (DLE ETX) (BSC) 20 Read response to text (Start-Stop) 21 All reset commands 22 Read skip 23 Write break 24 Any command issued during OPEN, LOPEN, or CLOSE (Set Address, Enable, Disable, and Set Mode commands) 25 Read response to text (BSC) In addition to the TP-Op codes, bits 0 and 1 may be set to zero or one. A 1 in the indicated bit has the following meaning: bit 0 the command just executed was the last command in the channel program. (For on-line tests, the command just executed was the last of the initialization commands [i.e., prior to Read Text or Write Text commands]). bit 1 the command just executed was the first Read or Write Text command executed in the channel program using Dynamic Buffering. (For BSC on-line test, the command just executed was an initialization command [i.e., not a Read Text or Write Text command] .)
29	1	ERRST	Error status flags which may be set if an I/O error has occurred (i.e., a completion code of 41 in the DECSDECB). Bit 0: The START I/O instruction resulted in a condition code of 3, indicating that the control unit or the specified line address is not operational. Bit 1: An error condition which should not occur (is undefined for the particular command or device) has occurred. Bit 2: An error condition occurred on an I/O operation initiated by the error recovery routines as an intermediate recover procedure, as part of diagnostic write/read procedure (2701 only), or as part of a disconnect procedure for a switched line. Bit 3: A diagnostic write/read operation terminated in error, indicating a control unit failure (2701 only). An error occurred that makes the integrity of the device regeneration buffer doubtful (local 3270 only). Bit 4: A disable command was issued to a switched line by the error recovery routines after detection of a permanent error on that line. Bits 5-7: (Reserved)
30	2	CSWST	The status bits from the CSW for the last CCW that was executed.
32	4	ADRPT	Pointer to the addressing list entry used in the previous operation.
36	4	POLPT	For programmed polling, pointer to the current entry in the polling list which will point to the terminal to be polled. For Auto Poll, the first byte contains an index identifying the current entry in the polling list and the last three bytes contain a pointer to the beginning of the polling list. For BSC on-line test, pointer to the user-specified text data provided in ONLTST macro [for test types 00 and 01].)
40	2		(Reserved.)
42	2	WLNG	Length in bytes of the data area (in leading-graphics, conversational, or READ TCW operations). (BSC only)
44	4	WAREA	Address of the data area (in leading graphics, conversational, or READ TCW operations). (BSC only)

A unit control block (UCB) is built for each line or local 3270 device at system generation time and is used by IOS during execution to determine physical locations. The only field requiring the attention of the BTAM user is the device type word, which gives details of the terminals on the line or local 3270 device: control unit, adapter, model, and optional features. This word is fully described in Appendix H. Figure 21 shows the format of the UCB. Table 5 shows the contents of the UCB.

Displacement (DEC)				
0	SRTEJBNR Internal Job No.	SRTECHAN Allocation Channel Mask	UCBID Identifier	SRTESTAT status Byte A
4	UCBCHA Channel Address	UCBUA Unit Address	UCBFL1 Flag Byte 1	UCBDTI Device Table Index
8	UCBETI Error Routine Key	UCBSTI Statistics Table Index	UCBLCI Channel Table Index	UCBATI Attention Table Index
12	UCBWGT Flags and Mask	UCBNAME Unit Name		
16	UCBTYP Communications Equipment or Graphics Device Type			
20	UCBLTS Last Request Element		UCBSNS Sense Information	
24	Additonal Optional Features		Attention Count	Control Byte
28	IRB Address			
32	Initialized RLN	(reserved)		
36	Device Index	Control Block Link		

Graphic Devices Segment

Figure 21. Unit Control Block (UCB)

Table 5. BTAM UCB Fields

Relative Location (DEC)	Size (in bytes)	Name	Contents
0	1	SRTEJBNR	Internal job identification.
1	1	SRTECHAN	Allocation channel mask.
2	1	UCBID	UCB identification - hex FF.
3	1	SRTESTAT	Status byte A.
4	1	UCBCHA	Channel address. Bit 0 - Halt I/O has been issued. Bit 1 - Status modifier. Bit 2, Bit 3, Bit 4 } reserved. Bit 5, Bit 6, Bit 7 } channel address (binary).
5	1	UCBUA	Unit address.
6	1	UCBFL1	Flags: Bit 0 - UCB busy (set at SIO, reset at DE). 1 - UCB not ready (awaiting operator intervention). 2 - Post flag (waiting DE or error at DE will be passed at next SIO). 3 - Job intercept 4 - Control unit busy 5 - When bits 5 & 6 are 01, indicates HIO command is inhibited because channel program is already in progress. 6 - Enable command has completed. 7 - I/O error routine is in control of the line.
7	1	UCBDTI	Index to Device Table (one entry/device type).
8	1	UCBETI	Index used to get error routine for this device.
9	1	UCBSTI	Index to Statistics Table.
10	1	UCBLCI	Index to Logical Channel Table.
11	1	UCBATI	Index to Attention Table.
12	1	UCBWGT	Flags & channel mask.
13	3	UCBNAME	Unit Name (EBCDIC)
16	4	UCBTYP	Device type. (see Appendix H).
20	2	UCBLTS	Last request element.
22	2	UCBSNS	Sense information.
24	2		An extension of the optional features byte in the UCBTYP field (local 3270 display system only). Byte 0: Bit 0 - magnetic card reader adapter. Bit 1 - selector pen. Bit 2 - numeric lock. Bits 3-8 - reserved. Byte 1: Bits 0-8 - reserved.
26	1		If the device index is one, this byte contains the number of attentions not serviced in the line group; otherwise, this byte is reserved (local 3270 display system only).
27	1		Attention handling flags (local 3270 display system only). Bit 0 - OLTEP executing flag. If this flag is one, the device is being used by OLTEP. Bits 1-4 - reserved. Bit 5 - read initial pending flag. Device index is one: If this flag is one, a Read Initial is outstanding and waiting for an attention interruption. If this flag is zero, and the initialized RLN field is nonzero, the second-level attention routine was scheduled to start a read operation. Device index is not one: This bit is reserved. Bit 6 - skip flag. If this flag is one, a Read Initial for the device is ignored. Bit 7 - attention flag. If this flag is one, an attention interruption was received from the device.
28	4		Address of the interruption request block (IRB) that points to the second-level attention routine (local 3270 display system only).
32	1		Initialized relative line number (local 3270 display system only). Device index is one: If this byte is zero, no Read Initial is pending for the line group. If this byte is one through 255, it is the relative line number of the IOB initialized by the BTAM Read/Write routine for a pending Read Initial. Device index is not one: This byte is reserved.
36	1		The device's relative line number, which is an index to the table of UCB addresses in the DEB (local 3270 display system only).
37	3		Control block link (local 3270 display system only). Device index is one: This field contains the address of the DEB for the line group. Device index is not one: This field contains the address of the UCB with a device index of one.

The device type word (UCBTYP) field located at relative location on hex 16 in the unit control block (UCB) has the format and content shown below.

UCBTYP				
Byte 1		Byte 2	Byte 3	Byte 4
IOS Flags	Model Code	Optional Features	Device Class	Unit Type
UCB + 16 (10)		17 (11)	18 (12)	19 (13)

Offset Dec	Offset Hex	Bytes and Alignment	Bit and State	Hex. Dig.	Field Description, Contents, Meaning
16	(10)	1 Byte 1	xxxx		I/O Supervisor flags
			x...		(Reserved bit)
			.1..		Overrunable device.
			..1.		Burst mode.
			..0.		Byte mode.
			...1		Data chaining.
			 xxxx		Model Code The value in this field and the value in the adapter type field (byte 4, bits 0-3) together identify the model. Adapter Type — Model
			 0001	-1	1- 1050 2- 1030 3- 1050 4- 83B3 5- TWX 6- WTTA 8- 2260
			 0010	-2	1- 1060 4- 115A
			 0011	-3	
			 0100	-4	1- 2740
			 0101	-5	1- 2741C (Correspondence code) 9- BSC1 (Nonswitched point-to-point)
			 0110	-6	1- 2741P (PTTC/BCD or EBCD code) 9- BSC2 (Switched point-to-point)
			 0111	-7	9- BSC3 (Nonswitched multipoint)
			xxxx xxxx		Model Code (local 3270 display system)
			0001 0001	11	Model 1
			0001 0010	12	Model 2

Offset Dec	Offset Hex	Bytes and Alignment	Bit and State	Hex. Dig.	Field description, Contents, Meaning
17	(11)	. 1 Byte 2			Optional features.
			1...		Automatic calling.
			.1..		Automatic polling.
			..1.		Checking (2740 only)
			..1.		Dual communication interface (2701 SDA-II)
			...1		Automatic answering.
			.... .1..		Dual code (2701 SDA-II)
			.... 10..		Station control (2740 only)
			.... 01..		Transmit control (2740 only)
			.... 11..		2760 Attachment.
			xx		Binary Value: 0 SADZER; 1 SADONE; 2 SADTWO; 3 SADTHREE
			xxx.		Keyboard Type (local 3277 display station)
			000.		No keyboard
			001.		66-key EBCDIC typewriter keyboard
			010.		78-key EBCDIC typewriter keyboard
			011.		66-key data entry keyboard
			100.		78-key operator console keyboard
			101.		66-key ASCII typewriter keyboard
			110.		78-key ASCII typewriter keyboard
			...1		Audible Alarm (local 3277 display station)
			.... xxx.		Character Generator Type (local 3277 display station)
			.... 000.		Domestic character generator
			.... 001.		ASCII A character generator
			.... 010.		ASCII B character generator
			.... 011.		United Kingdom character generator
			.... 100.		French character generator
			.... 101.		German character generator
			x		Character Generator Case (local 3277 display station)
			0		Monocase character generator
					Note: See byte 24 of the UCB for additional optional features (local 3277 display station)
			0000 0000		No Optional Features (local 3277 display statio

Offset Dec	Offset Hex	Bytes and Alignment	Bit and State	Hex. Dig.	Field description, Contents, Meaning
18	(12)	. . 1			Device Class.
		Byte 3		40	Communication equipment.
				10	Graphics.
19	(13)	. . . 1			Adapter Type.
		Byte 4		1-	IBM Terminal Adapter, Type I.
				2-	IBM Terminal Adapter,Type II.
				3-	IBM Telegraph Adapter.
				4-	Telegraph Adapter, Type I.
				5-	Telegraph Adapter, Type II.
				6-	World Trade Telegraph Adapter.
				7-	Synchronous Adapter, Type I.
				8-	IBM Terminal Adapter, Type III.
				9-	Synchronous Adapter, Type II.
					Control Unit
				-1	2702
				-2	2701
				-3	2703
					Unit Type (local 3270 display system)
				09	3277 display station
				0A	3284 printer
				0B	3286 printer

The interruption request block (IRB) is used to maintain information about an asynchronously executed routine.

If on-line testing is required (that is, T was coded among the EROPT options in the DCB macro instruction), an IRB that points to the on-line test control module is created and initialized for each line group during open. (If the line group contains local 3270 devices, this IRB is created by IGG0194Q and points to IGG019PI ; otherwise, it is created by IGG0193S and points to IGG019MR.)

If the line group contains local 3270 devices, the BTAM Open executor (module IGG0194Q) also creates and initializes an IRB that points to the local 3270 second-level attention routine (module IGG019PG).

Figure 22 shows the format of the IRB. Table 6 shows the contents of the IRB.

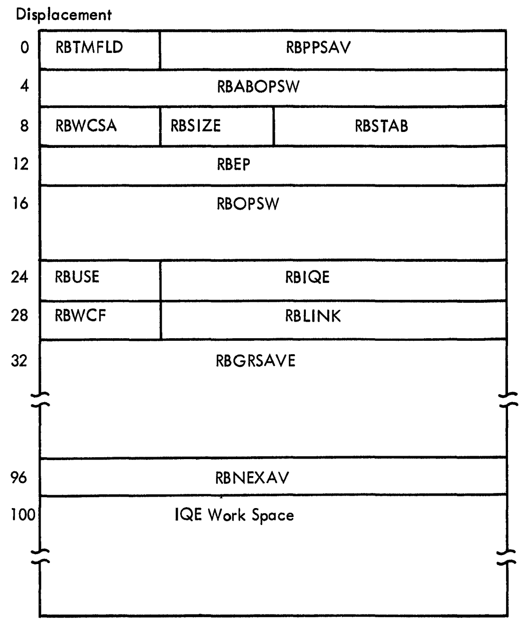


Figure 22. Interruption Request Block (IRB)

Relative Location	Length (Bytes)	Name RB+	Contents
0	1	TMFLD	Flags for the timer routines.
1	3	PPSAV	Address of the problem program's register save area.
4	4	ABOPSW	After execution of the ABTERM routine, contains bits 32-63 of the user's old PSW.
8	1	WCSA	Wait count save area.
9	1	SIZE	Size of this IRB in doublewords.
10	2	STAB	Status and attribute bits.
12	4	EP	Entry-point address of the asynchronously executed routine.
16	8	OPSW	User's old PSW.
24	1	USE	Use count used by ATTACH.
25	3	IQE	List origin for interruption queue elements (IQEs).
28	1	WCF	Number of requests waiting (wait count).
29	3	LINK	Address of the previous IRB on the queue; or the address of the TCB if this IRB is first on the queue.
32	64	GRSAVE	General register (0-15) save area.
96	4	NEXAV	Address of the next available IQE. For local 3270 attention handling, RBNEXAV contains IRB+100 if the IQE is available or 0 if it is not.
100			IQE work space. For local 3270 attention handling, contains one IQE.

Table 6. BTAM IRB Fields

The interruption queue element (IQE) is used to schedule an asynchronously executed routine.

If on-line testing is required (that is, T was coded among the EROPT options in the DCB macro instruction), IQEs are created and initialized during open. (If the line group contains local 3270 devices, these IQEs are created by IGG0194Q; otherwise, that are created by IGG0193S.)

If the line group contains local 3270 devices, the BTAM Open executor (module IGG0194Q) also creates and initializes an IQE in the IQE work area of the IRB that points to the local 3270 second-level attention routine.

Figure 23 shows the format of the IQE. Table 7 shows the contents of the IQE.

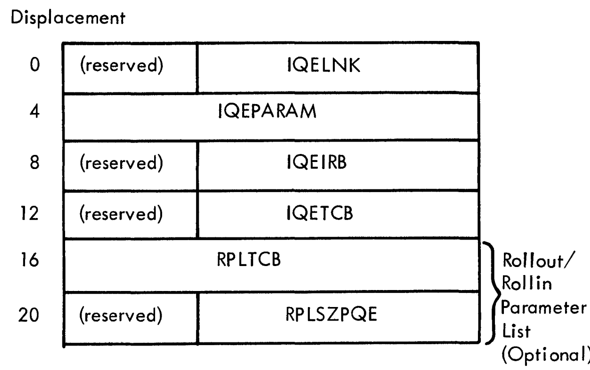


Figure 23. Interruption Queue Element (IQE)

Relative Location	Length (Bytes)	Name	Contents
1	3	IQELNK	Address of the next IQE on the queue.
4	4	IQEPARAM	The parameter that is to be passed to the asynchronously executed routine.
9	3	IQEIRB	Address of the IRB that points to the asynchronously executed routine.
13	3	IQETCB	Address of the TCB for the task executing the asynchronous routine.
16	4	RPLTCB	Address of the TCB for the task requiring or releasing an extension to a region.
21	3	RPLSZPQE	Size of the region requested (rollout) or address of the PQE describing the area (rollin).

Table 7. BTAM IQE Fields

This appendix lists all of the routines and modules composing BTAM; they are listed by module name (alphabetically) and by name of routine.

All modules whose names begin with IECT reside in the SYS1.TELCMLIB library; all others reside in the SYS1.SVCLIB library.

I. Listed Alphabetically by Module Name

(* Following a name indicates a device I/O module rather than an executable routine)

Module Name	Name of Routine
IECATEN	Local 3270 First-Level Attention Routine
IECTCHGN	Change Entry
IECTEDIT	Edit (IBM 50 Magnetic Data Inscriber)
IECTLERP	Line Error Print
IECTLOPN	Line Open
IECTONLT	On-Line Test Request
IECTSVC	**Local 3270 BTAM SVC Routine**
IECTTRNS	Translate
IGC058	REQBUF/RELBUF (SVC 58)
IGC0006F	On-Line Test Control (Start-Stop)
IGC0106F	IBM 1030 Terminal Test
IGC0206F	IBM 1050 Terminal Test
IGC0306F	IBM 1060 Terminal Test
IGC0406F	IBM 2740 Terminal Test
IGC0506F	IBM 2848/2260 Terminal Test (Load 1)
IGC0606F	IBM 2848/2260 Terminal Test (Load 2)
IGC0706F	USASCII and Transcode Test Message
IGC0806F	EBCDIC Test Message
IGC0906F	IBM 2741 Terminal Test (Correspondence Code)
IGC0A06F	IBM 2741 Terminal Test (PTTC Code)
IGC0B06F	IBM 2760 Terminal Test
IGC0C06F	IBM 2740 On-Line Test (Correspondence Code)
IGC0D06F	On-Line Test Control (BSC)
IGC0E06F	3270 EBCDIC Test Module (Types 23, 24)
IGC0F06F	3270 EBCDIC Test Module (Types 26, 27)
IGC1006F	Remote 3270 ASCII Test Module (Types 29, 30, 32)
IGC1106F	Remote 3270 ASCII Test Module (Types 31, 33, 34)
IGC1206F	On-Line Test Control (Local 3270)
IGC1306F	3270 EBCDIC Test Module (Type 25)
IGC1406F	3270 EBCDIC Test Module (Type 28)
IGE0004A	BTAM Start-Stop ERP Control (Load 1)
IGE0004B	Start-Stop ERP Data Check (Load 2)
IGE0004C	BTAM BSC ERP Control
IGE0104A	Start-stop ERP Data Check (Load 1)
IGE0104B	Start-Stop ERP Diagnostic Write/Read
IGE0104C	BSC ERP Data Check
IGE0204A	Start-Stop ERP Time Out
IGE0204B	ERP Line Error Recording (Start-Stop & BSC)
IGE0204C	BSC ERP Error Post
IGE0304A	Start-Stop ERP Intervention Required
IGE0304B	Start-Stop ERP Unit Exception
IGE0304C	BSC ERP Intervention Required
IGE0404A	Start-Stop ERP Lost Data
IGE0404B	Start-Stop ERP Read Skip/Write Break Check (Load 2)
IGE0404C	BSC ERP Time Out
IGE0504A	Start-Stop ERP Error Post
IGE0504B	Start-Stop ERP Overrun
IGE0504C	BSC ERP Special Return
IGE0604A	Start-Stop ERP Bus Out
IGE0604B	ERP Intervention Required Message Writer (Start-Stop and BSC)
IGE0604C	BSC ERP Lost Data
IGE0704A	Start-Stop ERP Read Skip/Write Break Check (Load 1)
IGE0704B	Remote 3270 Error Post Routine
IGE0704C	BSC ERP Bus Out and Overrun
IGE0804A	Start-Stop ERP Status Check
IGE0804B	ERP Channel Check and Interface Control Check (Start-Stop and BSC)
IGE0804C	BSC ERP Equipment Check and Command Reject
IGE0904A	BTAM Start-Stop ERP Control (Load 2)
IGE0904C	BSC ERP Unit Exception
IGG0193M	BTAM Open Executor (Load 1)
IGG0193O	BTAM Open Executor (Load 3)
IGG0193S	BTAM Open Executor (Load 4)
IGG0194N	BTAM Open Executor (Load 2)
IGG0194P	Local 3270 BTAM Open Executor
IGG0194Q	Local 3270 BTAM Open Executor

IGG019MA	BTAM Read/Write
IGG019MB	Channel End/Abnormal End Appendage
IGG019MC	Program Controlled Interrupt (PCI) Appendage
IGG019MD	IBM 1050, Nonswitched*
IGG019ME	IBM 1050, Nonswitched (Auto Poll)*
IGG019MF	IBM 1050, Switched*
IGG019MI	IBM 1060*
IGG019MJ	IBM 1030*
IGG019MK	IBM 1030 (Auto Poll)*
IGG019ML	AT&T 83B3*
IGG019MN	WU 115A*
IGG019MP	WU TWX 33/35*
IGG019MR	On-Line Test Control
IGG019MS	Buffer
IGG019MT	IBM 2740, Basic*
IGG019MU	IBM 2740, Dial-up*
IGG019MV	IBM 2740, Dial-up, with Transmit Control and Checking*
IGG019MW	IBM 2740, Dial-up, with Transmit Control*
IGG019MX	IBM 2740, Dial-up, with Checking*
IGG019MY	IBM 2740, with Station Control and Checking*
IGG019MZ	IBM 2740, with Station Control*
IGG019M0	IBM 2740, with Checking*
IGG019M1	IBM 2740, with Station Control and Checking (Auto Poll)*
IGG019M2	IBM 2740, with Station Control (Auto Poll)*
IGG019M3	IBM 2260 (Remote)*
IGG019M4	IBM 1060 (Auto Poll)*
IGG019M5	Nonswitched Point-to-Point* (BSC1)[1]
IGG019M6	Switched Point-to-Point* (BSC2)[1]
IGG019PA	Local 3270 Channel End/ Abnormal End Appendage
IGG019PB	World Trade Telegraph Terminals*
IGG019PC	Nonswitched Multipoint* (BSC3)[1]
IGG019PD	World Trade Telegraph Terminal Channel End/Abnormal End Appendage
IGG019PE	IBM 2741, Nonswitched*
IGG019PF	IBM 2741, Switched*
IGG019PG	Local 3270 Second-Level Attention Routine
IGG019PH	Local IBM 3270*
IGG019PI	Local On-Line Test Control
IGG019PK	2741 Break
IGG019PL	IBM 2740, with Checking and 2760 Attachment*
IGG019PM	IBM 2740, Dial-up, with Checking and 2760 Attachment*
IGG019PN	IBM 1050, Nonswitched*
IGG019PO	IBM 1050, Switched*
IGG019PP	IBM 2740, with Checking*
IGG019PQ	IBM 2740, Dial-up with Checking*
IGG0203M	BTAM Close Executor

---

[1]As specified in the UNIT operand of the IODEVICE macro instruction.

II. Listed by Routine

Name of Routine	Module Name
BTAM Open Executor (Load 1)	IGG0193M
BTAM Open Executor (Load 2)	IGG0194N
BTAM Open Executor (Load 3)	IGG0193Q
BTAM Open Executor (Load 4)	IGG0193S
Local 3270 BTAM Open Executor	IGG0194P
Local 3270 BTAM Open Executcr	IGG0194Q
BTAM Read/Write	IGG019MA
Channel End/Abnormal End Appendage	IGG019MB
World Trade Telegraph Terminal Channel End/Abnormal End Appendage	IGG019PD
Local 3270 Channel End/Abnormal End Appendage	IGG019PA
Program Controlled Interrupt (PCI) Appendage	IGG019MC
Local 3270 First-Level Attention Routine	IECTATEN
Local 3270 Second-Level Attention Routine	IGG019PG
Local 3270 BTAM SVC Routine	IECTSVC

2741 Break	IGG019PK
Buffer	IGG019MS
REQBUF/RELBUF (SVC 58)	IGC058
Change Entry	IECTCHGN
Edit (IBM 50 Magnetic Data Inscriber)	IECTEDIT
Line Error Print	IECTLERP
Line Open	IECTLOPN
Translate	IECTTRNS
BTAM Close Executor	IGG0203M
On-Line Test Routines:	
On-Line Test Request	IECTONLT
On-Line Test Control	IGG019MR
Local On-Line Test Control	IGG019PI
On-Line Test Control (Local 3270)	IGC1206F
On-Line Test Control (Start-Stop)	IGC0006F
On-Line Test Control (BSC)	IGC0D06F
IBM 1030 Terminal Test	IGC0106F
IBM 1050 Terminal Test	IGC0206F
IBM 1060 Terminal Test	IGC0306F
IBM 2740 Terminal Test	IGC0406F
IBM 2740 Terminal Test	IGC0C06F
IBM 2741 Terminal Test (Correspondence Code)	IGC0906F
IBM 2741 Terminal Test (PTTC Code)	IGC0A06F
IBM 2760 Terminal Test	IGC0B06F
IBM 2848/2260 Terminal Test (Load 1)	IGC0506F
IBM 2848/2260 Terminal Test (Load 2)	IGC0606F
USASCII and Transcode Test Message	IGC0706F
EBCDIC Test Message	IGC0806F
3270 EBCDIC Test Module (Types 23, 24)	IGC0E06F
3270 EBCDIC Test Module (Types 26, 27)	IGC0F06F
Remote 3270 ASCII Test Module (Types 29, 30, 32)	IGC1006F
Remote 3270 ASCII Test Module (Types 31, 33, 34)	IGC1106F
3270 EBCDIC Test Module (Type 25)	IGC1306F
3270 EBCDIC Test Module (Type 28)	IGC1406F

BTAM Error Recovery Procedures (ERP):

Common (Start-Stop & BSC) Routines:	
Line Error Recording	IGE0204B
Intervention Required Message Writer	IGE0604B
Channel Check & Interface Control Check	IGE0804B
Start-Stop ERP Routines:	
BTAM Start-Stop ERP Control (Load 1)	IGE0004A
BTAM Start-Stop ERP Control (Load 2)	IGE0904A
Data Check (Load 1)	IGE0104A
Data Check (Load 2)	IGE0004B
Diagnostic Write/Read	IGE0104B
Time Out	IGE0204A
Error Post	IGE0504A
Intervention Required	IGE0304A
Unit Exception	IGE0304B
Lost Data	IGE0404A
Read Skip/Write Break Check (Load 1)	IGE0704A
Read Skip/Write Break Check (Load 2)	IGE0404B
Overrun	IGE0504B
Bus Out	IGE0604A
Status Check	IGE0804A

BSC ERP Routines:	
BTAM BSC ERP Control	IGE0004C
Data Check	IGE0104C
Time Out	IGE0404C
Error Post	IGE0204C
Intervention Required	IGE0304C
Special Return	IGE0504C
Lost Data	IGE0604C
Remote 3270 Error Post Routine	IGE0704B
Bus Out & Overrun	IGE0704C
Equipment Check & Command Reject	IGE0804C
Unit Exception	IGE0904C

Device I/O Modules (Nonexecutable):

Start-Stop Terminals:	
IBM 1030	IGG019MJ
IBM 1030 (Auto Poll)	IGG019MK
IBM 1050, Nonswitched	IGG019MD
IBM 1050, Nonswitched (Auto Poll)	IGG019ME
IBM 1050, Switched	IGG019MF
IBM 1050, Nonswitched	IGG019PN
IBM 1050, Switched	IGG019PO
IBM 1060	IGG019MI
IBM 1060 (Auto Poll)	IGG019M4
IBM 2260 (Remote)	IGG019M3
IBM 2740, Basic	IGG019MT
IBM 2740, with Checking	IGG019M0
IBM 2740, with Station Control	IGG019MZ
IBM 2740, with Station Control (Auto Poll)	IGG019M2
IBM 2740, with Station Control & Checking	IGG019MY
IBM 2740, with Station Control & Checking (Auto Poll)	IGG019M1
IBM 2740, with Checking & 2760 Attachment	IGG019PL
IBM 2740, Dial-up	IGG019MU
IBM 2740, Dial-up, with Checking	IGG019MX
IBM 2740, Dial-up, with Transmit Control	IGG019MW
IBM 2740, Dial-up, with Transmit Control & Checking	IGG019MV
IBM 2740, Dial-up, with Checking & 2760 Attachment	IGG019PM
IBM 2740, with Checking	IGG019PP
IBM 2740, Dial-up with Checking	IGG019PQ
IBM 2741, Nonswitched	IGG019PE
IBM 2741, Switched	IGG019PF
AT&T 83B3	IGG019ML
AT&T TWX 33/35	IGG019MP
WU 115A	IGG019MN
World Trade Telegraph Terminals	IGG019PB
BSC Stations:	
Nonswitched Point-to-Point (BSC1[2])	IGG019M5
Switched Point-to-Point (BSC2[2])	IGG019M6
Nonswitched Multipoint (BSC3[2])	IGG019PC
Local Station:	
IBM 3270	IGG019PH

----

[2]As specified in the UNIT operand of the IODEVICE macro instruction.

LOCAL 3270 ERP ROUTINES (IGE0010E AND IGE0110E)

The local 3270 ERP routines (IGE0010E and IGE0110E) handle error recovery and error recording for the local 3270 display station and the local 3284 or 3286 printer. (The UCB for a local 3270 device has a X'69' in the UCBETI field to indicate the proper ERP.) The ERP routines reside in SYS1.SVCLIB and are executed in the IOS transient area.

The local 3270 ERP receives control from IOS after an I/O error involving a local 3270 device has been detected by the device, the control unit, or the channel. While the error condition is being processed, all I/O interruptions from the device are passed to the ERP by IOS. When the ERP has recovered from the error condition or has determined that it cannot correct the condition, the ERP returns control to IOS.

For the logic flow of the local 3270 ERP routines (IGGE0010E and IGE0110E), see Charts X (5 parts) and Y (3 parts).

TP RECORDER (IGE0625F)

The TP recorder (IGE0625F) records in SYS1.LOGREC T-type records containing information from remote 3270 error status (SCH % R) messages. These records have the format:

Buffer Size	Not Used	XCTL Code	Record ID	CUA	CU	DS	STAT1	STAT2
2	1	2	1	2	1	1	1	1

Length (in Bytes)

where:
- Buffer Size is X'000B'
- Not Used is X'00'
- XCTL Code is X'0000' indicating that control is to be passed to IOS)
- Record ID is X'03'
- CUA is channel/unit address
- CU/DS/STAT1/STAT2 are information from the error status message (translated into EBCDIC if it was received in ASCII)

The TP recorder receives control from the remote 3270 Error Post routine (IGE0704B) and passes control to IOS.

3270 SCAN (IECTSCAN)

The 3270 subroutine to change alphanumeric (SCAN) converts data streams produced by 2260 application programs into data streams that can be sent to 3270 devices and converts data streams received from 3270 devices into data streams that can be processed by 2260 application programs.

SCAN receives control from the expansion of the SCANREQ macro instruction, which constructs the Display Alphanumeric Control Area (DACA) and places into it:

- The addresses and lengths of the source and sink areas
- The device address of the 2260
- Descriptions of the 2260 and 3270 screens
- Switches indicating the type of processing (RVOID or PROT or both) and the type of 2260 operation performed with the source data stream

SCAN consists of a write subroutine that deals with output data streams, a read subroutine that deals with input data streams, and routines that maintain a logical 2260 screen image on a logical 3270 screen. The read and write subroutines communicate through the DACCRS, DACSOMAD, DACFLG, and DACIND fields in the DACA. SCAN receives as input the DACA, a source data stream, and a sink output area. SCAN produces as output a converted data stream in the sink area, the length of that data stream in register 0, and a return code in register 15.

Figure 24 shows the format of the DACA. Table 8 shows the contents of the DACA. For the logic flow of SCAN (IECTSCAN), see Chart Z (19 parts).

Displacement				
0	DACCRS		DACSOMAD	
4		DACSRCAD		
8		DACSINAD		
12	DACSRCLN		DACSINLN	
16	DACDSSCR		DACANSCR	
20	DACDSAD	DACOPTYP	DACFLG	DACIND

Figure 24. Display Alphanumeric Control Area (DACA)

Relative Location	Bytes and Alignment	Name	Contents
0	2	DACCRS	Binary address of the cursor in the 3270 buffer.
2	. . 2	DACSOMAD	Binary address of the last set field (SOM) order in the 3270 buffer.
4	1		Temporary save byte: used during conversion from 3270 six-bit to binary.
5	. 3	DACSRCAD	Storage address of the source.
8	1		Temporary save byte: used during conversion from 3270 six-bit to binary.
9	. 3	DACSINAD	Storage address of the sink.
12	2	DACSRCLN	Binary length of the source.
14	. . 2	DACSINLN	Binary length of the sink.
16	2	DACDSSCR	2260 screen descriptor:
16	1	DSLINE	Binary number of lines on the 2260 screen.
17	. 1	DSWIDE	Binary character width of the 2260 screen.
18	. . 2	DACANSCR	3270 screen descriptor:
18	. . 1	ANLINE	Binary number of lines on the 3270 screen.
19	. . . 1	ANWIDE	Binary character width of the 3270 screen.
20	1	DACDSAD	2260 device address.
21	. 1	DACOPTYP	Special processing indicators and BTAM operation used with 2260:
		1	RVOID = YES.
		. 1	PROT = YES.
		. . 1	Write to cursor.
		. . . 1	Write to line address.
		. . . . 1 . . .	Write erase.
		. . . . . 1 . .	Read manual input (MI).
		. . . . . . 1 .	Short read manual input (MI).
		 1	Read buffer.
22	. . 1	DACFLG	Flags used by SCAN:
		1	Cursor positioning switch (BLAT).
		. 1	Last read was a short read MI.
		. . 1	SOM present in data stream (output).
		. . . 1	Data stream more than 256 bytes.
		. . . . 1 . . .	SOM present in data stream (input or output).
		. . . . . 1 . .	No SMI present in data stream (input).
		. . . . . . 1 .	STX present in data stream.
		 1	Remote device.
23	. . . 1	DACIND	Contains offset into the source of the first character that is on an integral boundary.

Table 8. DACA Fields

SCAN Module Description

Entry Point: The only entry point is at the first executable instruction labeled IECTSCAN.

Input: Register 1 contains the address of the DACA, which contains information supplied by means of the SCANREQ macro instruction.

Output: The sink area contains the converted data stream. Register 0 contains the length of that data stream. Register 15 contains one of the fcllcwing return codes:

0 Normal.

4 The source area contained only an AID character; the sink area contains the unaltered AID character. (The CLEAR, PA1, PA2, or PA3 key on the 3270 keyboard was pressed, or the control unit could not determine the cause of the interruption.)

8 The sink area is not long enough to contain the entire converted data stream. (SCANREQ should be reissued with a larger sink area provided.)

C No start of message symbol (SOM or SMI) is present in the input data stream just processed; the operation was not a read buffer operation.

10 An SOH was the first character in the data stream (cther than an STX). (On a read operation, either a sense/status message was received or the operator entered a request for test. On a write to line address operaticn, the line address character did not fall within the valid line addresses for a 2260.)

14 On a read operation, the cursor location did not fall within the boundaries of the 2260 screen.

External References: SCAN references the DACA through DACAD, the source area through DACSRCAD, and the sink area through DACSINAD.

Table/Work Areas: SCAN uses the DACA provided by the application program. SCAN uses translate tables to locate certain characters and to convert hexadecimal addresses.

Exits: SCAN exits by means of a branch return on register 14 with a return code in register 15.

Attributes: Reentrant.

SCAN Subroutines

Initial Housekeeping: (See part 1 of Chart Z.) Addressability is set up for the source area and the sink area. If the device is remote, bit 7 of DACFLG is set to one. If the source contains an STX, bit 6 of DACFLG is set to one, and the STX is placed into the sink. Depending on the operation type, control is passed to a read or write subroutine.

WRITE: (See part 2 of Chart Z.) Housekeeping related to output data streams is done. The opening orders and control needed for the 3270 are placed into the sink, based on bit 7 of DACFLG and the write operation type. Depending on bit 1 of DACFLG and the contents of DACSOMAD, an order may be inserted to remove an existing SOM from the 3270 screen. If RVOID=NO for a write erase operation, the screen is set to blanks. Control is passed to WRITENT.

WRITENT: (See part 2 of Chart Z.) The current 2260 line and current 2260 cursor position are established, and the group of characters to be dealt with initially from the source area is computed for the write operation type. For write to cursor and write to line address, the 3270 buffer position of the first character is placed into the sink. For write erase, the 3270 buffer position is already zero. Control is passed to WITCH1.

WITCH: (See part 3 of Chart Z.) Initialization, termination, and data stream processing (not involving special characters) specifically related to the data are done.

- WITCH1 - If the sink is too small, control is passed to OUTSINK. If the group of characters is zero, control is passed to WIT. Characters that do not require special processing are moved into the sink area. If a character that requires special processing is found (by means of a translate table), a branch (determined by the function byte selected from the table) is made into WRSPCHAR. If no special processing is needed, control is passed to NOCHAR.
- NOCHAR - The entire data group is moved into the sink. If all data has been processed, control is passed to WIT; if data remains, control is passed to WITCH2.
- WITCH2 - If screen wrap exists, control is passed to WALL. Otherwise, the cursor is adjusted to the next 3270 line. If the 2260 and 3270 screens are not equal, control is passed to WITCH3. If they are equal, control is passed to WITCH4.
- WITCH3 - An RTA order is inserted to place nulls in the excess 3270 screen. Control is passed to WITCH4.
- WITCH4 - The grouping is set to equal the 2260 width. Control is passed to WITCH1.

WALL: (See part 5 of Chart Z.) The 3270 screen is formatted to maintain the 2260 image. If PROT=YES, the protected attribute is inserted. The current line and the cursor position are set to zero. If the 2260 and 3270 screens are not equal, control is passed to WITCH3. Otherwise, control is passed to WITCH4.

WRSPCHAR; (See part 13 of Chart Z.) A character requiring special processing is dealt with by the appropriate subroutine.

- WRTNEWL - Processes a new line character. Control is passed to WITCH2.
- WRTSOM - Processes an SOM symbol. Control is passes to SETTLE.
- WRTEOM - Processes an EOM symbol. Control is passed to SETTLE.
- WRTETX - Processes and ETX. If the device is known to be remote, control passes to WIT. Otherwise, control returns to the housekeeping subroutine.
- WRTNULL - Processes a null. Control is passed to SETTLE.
- SETTLE - Housekeeping is done for other subroutines. Control is passed to WITCH1.

WIT: (See part 6 of Chart Z.) Processing for output data is terminated. If PROT=YES, an SOM is checked for on the screen. An insert cursor order is set following the last data, and, if it is a remote data stream, an ETX is placed at the end. Control is passed to OUTNORM.

READ: (See part 9 of Chart Z.) If an SOH is present, control is passed to CUTERR. If the device is remote, the 3270 device address is replaced by the contents of DACDSAD. If the AID character indicates that the CLEAR, PA1, PA2, or PA3 key was pressed, the AID character is placed into the sink (followed by an end framing character if needed), and control is passed to OUTSPEC. For a read buffer operation, processing begins at READ1, and control passes to RDLAT and back to READ1 after each line has been processed. When the last 2260 line has been processed, control is passed to READOUT. For other read operations, the cursor position is related to a storage address. If an SBA sequence is present, the field position is related to a storage address, and control is passed to SOURTST. If no SBA sequence is present, control is passed to CKSMI.

SOURTST: (See part 10 of Chart Z.) The boundaries of data conversion are established, and a branch is taken to RDLAT.

- CKSMI - An SMI (X'4A') is searched for in the data stream. If one is found, its position becomes the field position, and control is passed to SOURTST. If no SMI is found, bit 5 in DACFLG is set to one, and control is passed to READOUT.
- READOUT - An ETX is inserted if needed, and a branch is taken to OUTNORM.

RDLAT: (See part 11 of Chart Z.) Characters that do not require special processing are moved into the sink area. If a character that requires special processing is found (by means of a translate table), a branch (determined by the function byte selected from the table) is made into KERF. If no special processing is needed, control is passed to RDMOV.

- RDMOV - The entire group is moved into the sink. Control is passed to MORTST.
- MORTST - If nothing remains in the source area to process, control is passed to READOUT. Otherwise, for a read buffer operation, the current line is incremented, the source pointer is set at the beginning of the line, and control is passed to READ1. For other read operations, the length is set to 255 or all remaining data, and control is passed to RDLAT.

KERF: (See part 16 of Chart Z.) A character requiring special processing is dealt with by the appropriate subroutine.

- RDSCM - Processes a start field. Control is passed to RDSMI.
- RDSMI - Processes an operator-entered SOM (X'4A'). Control is passed to DATLNTST.
- RDECM - Processes an ECM. Control is passed to DATLNTST.
- RDFLDMK - Processes a 3270 field mark. Control is passed to DATLNTST.
- RDETX - Processes an ETX. If the device is remote, control is passed to READOUT. Otherwise, control returns to the housekeeping subroutine.
- RDBSOM - Processes a start field. Control is passed to RDBSMI.
- RDBSMI - Processes an operator-entered SOM (X'4A'). Control is passed to INCREM.
- RDBEOM - Processes an EOM. Control is passed to INCREM.
- RDBFM - Processes a field mark. Control is passed to INCREM.
- RDNULL - Processes a null or invalid character. For read buffer operation, control passes to INCREM. Otherwise, control passes to DATLNTST.
- DATLNTST - Handles return for a normal read. If the remaining group is zero, control passes to MORTST. Otherwise, control passes to RDLAT.
- INCREM - Handles return for a read buffer. If the remaining group is zero, the source is pointed to the next line, the line is incremented, and control is passed to READ1. Otherwise, control is passed to RDLAT.

INVERT: (See part 19 of Chart Z.) Incoming 3270 addresses are converted to hexadecimal addresses.

OUTVERT: (See part 19 of Chart Z.) Outgoing 3270 addresses are converted to 3270 addresses.

OUTPROC: (See part 7 of Chart Z.) A special function is performed by the appropriate condition return processor, register 15 is loaded with a 16 (X'10'), and control is passed to a point that leaves the required return code in register 15.

- OUTNORM - If bit 5 of DACFLG is set to one, control is passed to OUTNSMI. If bit 4 of DACFLG is set to one and the operation is a write, the SOM address is saved in DACSOMAD. If bit 4 of DACFLG and bit 6 of DACCPTYP are set to zero, a halfword of ones is placed into DACSOMAD. Control is passed to GOBACK00.

- OUTSPEC - Control is passed to GOBACK04.
- OUTSINK - Control is passed to GOBACK08.
- OUTNSMI - Control is passed to GOBACK0C.
- CUTERR - Control is passed to GOBACK10.

abnormal end appendage
  description 26-28
  logic flow 146-154
adapter type codes 243-244
ASMTRTAB macro instruction 10
attention interruption handling (local 3270) 20.3, 20.5
Auto Poll 20-21, 88-90

BTAM control blocks 10-11
  linkages 225
  role in creating access method 11-13
BTAM routines 22-248
buffer
  allocation 20
  control block (BCB) 31
  request queue 20,31
  routine (IGG019MS)
    description 30-31
    logic flow 137
  unavailable 20
buffering, dynamic (see dynamic buffering)

Change Entry routine (IECTCHGN) 31-32
  logic flow 204-205
channel command codes 40, 41
channel command word (CCW) 17, 36
channel end/abnormal end appendages (IGG019MB, IGG019PD) (see also local 3270)
  completions, normal and abnormal 26-27
  description 26-28
  function 26
  logic flow 146-154
channel programs
  BSC 91-128
  (see Contents for specific line types)
  command code 40,41
  control information for generating 36-39
  count 40
  flags 40
  format in this manual 40
  general information 40-42
  local 3270 128.1
  model 17
  start-stop 43-90
  (see Contents for specific station types)
  TP-Op codes 40, 41-42
CHGNTRY macro instruction 9
CHGNTRY operation (local 3270) 20.3, 20.4
CHGNTRY routine (IECTCHGN) 31-32
  logic flow 204-205
CLOSE macro instruction 10-13
Close executor (IGG0203M)
  description 36
  logic flow 141
command codes 40, 41
communication lines, requirements for grouping 11
completion codes 235

control blocks, BTAM
  linkages 225
  role in creating access method 11-13
control information for channel program generation 36-39
  channel command word (CCW) 17, 36
  device I/O directory 23, 38-39
  device I/O module 36-37, 245-248

data control block (DCB)
  creation 10
  description 226
  fields 227-229
  format 226
  function 10
data definition (DD) statement 10, 11
data event control block (DECB)
  creation 11
  descriptions 236
  fields 11, 237-240
  format 236
  function 9, 11, 17
data extent block (DEB)
  creation 10
  description 230
  fields 231
  format 230
  function 10, 12
data set, line group
  concatenation 12
  definition 11-13
  opening 13
DATAMGT macro instruction 13
DCB (see data control block)
DCB macro instruction 9
DD (data definition) statement 11
DDNAME parameter for DCB 10
DEB (see data extent block)
DECB (see data event control block)
device codes 243-244
device I/O directory 23, 38-39
  format 39
  providing additional space in 23
device I/O module 36-37, 38
  example 37
  format 36-37
  lists of 38, 245-248
DFTRMLST macro instruction 9, 13
dynamic buffering
  description 20, 28
  logic flow 19, 20
  Read/Write CCWs 29

ECB (see event control block)
Edit routine (IECTEDIT) 32
  logic flow 207-211
error counters 9
error recovery procedures (ERP) (see also local 3270) 30

error statistics 9
   message IEC801I 9
   message IEC802I 9
event control block (ECB)
   creation 11
   description 11, 235
   format 235
   function 11

features, optional 243-244

Input/Output block (IOB)
   creation 11
CSW field 232
   description 232
   fields 234
   format 232, 233
   function 11
interruption queue element (IQE)
   description 244.5
   fields 244.5
   format 244.5
   function 12
interruption request block (IRB)
   description 244.3
   fields 244.3
   format 244.3
   function 12
IOCONTRL macro instruction 13
IODEVICE macro instruction 13

LERB macro instruction 9
LERPRT macro instruction 9, 32
Line Error Print (IECTLERP) routine
   description 32
   logic flow 182
line group
   defining 11-13
   opening 14
line group data set (see data set, line group)
Line Open (IECTLOPN) routine 33
   logic flow 138
local 3270 BTAM Open executor (IGG0194P, IGG0194Q)
   description 23
   logic flow 136.1, 136.4
local 3270 BTAM SVC routine (IECTSVC)
   description 30.1
   logic flow 156.2
local 3270 channel end/abnormal end appendage (IGG019PA)
   description 28
   logic flow 154.1
local 3270 ERP routines (IGE0010E, IGE0110E)
   description 249
   logic flow 224.7
local 3270 first-level attention routine (IECTATEN)
   description 30
   logic flow 156
local 3270 second-level attention routine (IGG019PG)
   description 30.1
   logic flow 156.1
LOPEN macro instruction 9, 33

macro instruction
   assembly 13
   categories 9

ONLTST macro instruction 34
on-line test
   description 34-35
   list of modules 34
   logic flow 156-161, 212-224
   RFT messages 34-35
   types, BSC 35
Open executor (see also local 3270) 12, 14, 22, 23
   description 14, 22,23
   logic flow 129-136
opening line groups 14
OPEN macro instruction 9
optional feature codes 243-244

Poll command (Auto Poll) 21
Program controlled interrupt (PCI) appendage (IGG019MC)
   description 28-29
   logic flow 155
   with dynamic buffering 17, 28
program request block (PRB) 11

Read Initial operation (local 3270) 20, 20.1
READ macro instruction 9, 10
Read operation
   completion 15, 17-18
   evolution of 13-19
   execution 13
   local 3270 20, 20.2
   logic flow 16
   types 37-38
Read/Write routine (IGG019MA)
   area address subroutines 24-25
   count subroutines 25-26
   description 23-26
   functions 24
   logic flow 142-145
   purpose 17
relative line numbers 12
RELBUF macro instruction 9, 20, 35
remote 3270 Error Post routine (IGE0704B)
   description 30.4
   logic flow 194
REQBUF macro instruction 9, 20, 35
REQBUF/RELBUF (IGC058) routine 35
RESETPL macro instruction 10
RESETPL operation (local 3270) 20.3, 20.5
RFT (request-for-test) message 34-35

SVC 58 (IGC058) routine
description 35-36
logic flow 140
SYS1.TELCMLIB 13

table of offsets 17, 36-37
table of special characters 38
TELCMLIB macro instruction 13
TP recorder (IGE0625F), description 249
TPEDIT macro instruction 32
Tp-Op codes
definition 36
invalid 36
location 36
summary 41-42
Translate (IECTTRNS) routine
description 33-34
logic flow 139
TRNSLATE macro instruction 10, 33
TWAIT macro instruction 10,11

unit control block (UCB)
creation 11
description 241
device codes 243-244
device type word 243-244
fields 242
format 241
function 11
graphic devices segment 241, 242
system generation 11

WAIT macro 11
World Trade telegraph channel end appendage 152-154
WRITE macro instruction 9, 10
Write operation
local 3270 20, 20.2
types 37-38
2741 Break routine, logic flow 206
3270 SCAN (IECTSCAN)
description 249
logic flow 224.16

GY30-2001-5

IBM System/360 OS BTAM PLM (S360-30) Printed in U.S.A. GY30-2001-5

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

READER'S COMMENT FORM

IBM System/360 Operating System
Basic Telecommunications Access Method
Program Logic Manual

Order No. GY30-2001-5

- How did you use this publication?

  As a reference source ☐
  As a classroom text ☐
  As .... ☐

- Based on your own experience, rate this publication . . .

As a reference source:	..... Very Good	.... Good	..... Fair	..... Poor	..... Very Poor
As a text:	..... Very Good	.... Good	..... Fair	..... Poor	..... Very Poor

- What is your occupation? ....

- We would appreciate your other comments; please give specific page and line references where appropriate. If you wish a reply, be sure to include your name and address.

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

YOUR COMMENTS, PLEASE . . .

Your answers to the questions on the back of this form, together with your comments, help us produce better publications for your use. Each reply is carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in using your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

Cut Along Line

Fold **Fold**

FIRST CLASS
PERMIT NO. 569
RESEARCH TRIANGLE PARK
NORTH CAROLINA

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY . . .

IBM Corporation
P. O. Box 12275
Research Triangle Park
North Carolina 27709

Attention: Publications Center, Dept. E01

Fold **Fold**

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)