IBM 3274 Control Unit to Distributed Function Device Product Attachment Information

This document is provided for use in conjunction with other relevant IBM publications, and IBM makes no warranty, express or implied, relative to its completeness or accuracy. The information in this document is current as of the date of its publication, but is subject to change by IBM at any time without notice. This document is not intended to be a manufacturing or engineering specification; the user is assumed to understand the interrelationship among any affected systems, machines, programs and media.

IBM may have patents or pending patent applications covering subject matter described in this document, including appearance design patents or applications. The furnishing of this document does not constitute or imply a grant of any license under any patents, patent applications, trademarks, copyrights or other rights of IBM or of any third party, or any right to refer to IBM in any advertising or other promotional or marketing activities. IBM assumes no responsibility for any infringement of patents or other rights that may result from use of this document or from the manufacture, use, lease or sale of apparatus described herein.

Licenses under IBM's utility patents are available on reasonable and non-discriminatory terms and conditions. IBM does not grant licenses under its appearance design patents. Inquiries relative to licensing should be directed in writing to the IBM Director of Commercial Relations, International Business Machines Corporation, Armonk, New York, 10504.

March 1984

CONTENTS

1.0 Introduction 1
1.1 Basic Operation 1
1.2 Interface States and TCA Ownership 2
1.2.1 Interface Disconnected 2
1.2.2 Interface Connected 2
1.2.3 Interface Connected and Idle 2
1.2.4 Interface Connected and Active 3
1.2.5 Reserved Fields 4
1.3 Interface State Diagram 4
1.4 Device States 4
1.4.1 Offline to CU 4
1.4.2 Initialization in Process 4
1.4.3 Online to CU 5
1.4.4 Pending States 5
1.4.5 Online to Host 5
1.4.6 Device State Diagram 5

2.0 Function Split 6
2.1 Common 6
2.2 BSC 6
2.3 Local Channel 6
2.4 SNA 7

3.0 Device Hardware 8
3.1 General Description 8
3.2 Word Formats 8
3.2.1 Command Word to Device 8
3.2.2 Data Word To Device (Bit 2 is most significant) 9
3.2.3 Status Word to Controller 9
3.2.4 Data Word to Controller (Bit 2 is most significant) 9
3.3 Command Codes To Device 10
3.4 READ Command Functions 11
3.4.1 00001-POLL and 10001-POLL/ACK 11
3.4.2 Response To POLL (STATUS WORDS) 11
3.4.3 Other READ Commands 12
3.5 WRITE Command Functions 12
3.6 COAX Transmission Protocol 14
3.6.1 Line Quiesce Pattern 14
3.6.2 Unique Control Code Violation 14
3.6.3 Transmission Termination Sequence (Mini-Code Violation - MCV) . 15
3.6.4 Coax Transmission Waveforms 15
3.7 Transmit Check 18
3.8 Device Buffer 18
3.8.1 Device Buffer Format 19
3.8.2 Device Buffer Control 21
3.9 Controller TCA Support Level 21
3.9.1 Base TCA Support Level 22
3.9.2 Device Initiated Unbind Support Option 22

4.0 Device Control 23
4.1 Initialization 23
4.2 CU Function Request Synchronization 24
4.3 Asynchronous Event Synchronization 25
4.4 Expedited Status (ES) Interface 25
4.4.1 Flow 25
4.4.2 Status Prioritization 26
4.5 Multiple Logical Terminal Routing 26
4.6 RAS Considerations 27
4.6.1 CU Active 27
4.6.2 Device Active 27
4.6.3 Process Timings 28
4.6.4 Error Events 29
4.6.5 Error Event Logging 29
4.7 Coax Error Recovery 29
4.7.1 Retry of Non Start Op Command Queues 29
4.7.2 Retry of Start Op Command Queues 29

4.7.3 Unrecoverable Errors 30
4.7.4 Detection of Synchronization Errors 30

5.0 Defined Function Requests 31
5.1 CHOP 31
5.2 WCUS 31
5.2.1 WCUS(20) Device Identification 32
5.2.2 WCUS(42) - RTM CONTROL 34
5.3 WDAT 35
5.4 WDBD 35
5.5 WLCC 35
5.6 LOCK 36
5.7 RDAT 36
5.8 PDAT 36
5.9 CTCCS 36
5.10 RDBD 37
5.11 RDCOPY 37
5.12 WCTL 37
5.13 RPID 38

6.0 Defined Synchronous Status 39
6.1 FCSE 39
6.2 FC 39
6.3 FCIR - Function Complete Input Required 39
6.4 ERFR 40
6.5 FRA - Function Request Aborted 40
6.6 TABLE SUMMARY 40

7.0 Defined Asynchronous Status Event Values 42
7.1 AEER 42
7.2 AEEP 42
7.3 AEDBA (3290 Only) 43
7.4 AEDBS 43
7.5 AEEB (NON-SNA ONLY) 43
7.6 AEDV 43
7.7 AEFREE 44

8.0 Expedited Status (ES) Requests 45
8.1 Device Interval Timing (X'02') 45
8.2 Response Time Monitor Overview 45
8.2.1 Start Timer For Response Time Monitor (X'04') 46
8.2.2 Stop Timer For Response Time Monitor (X'06') 47

9.0 Local Copy 50
9.1 Data Stream Interface to Subsystem Printers 50
9.2 Local Copy Function Requests and Status 50
9.2.1 RDCOPY 50
9.2.2 WCTL 50
9.2.3 WCUS 50
9.2.4 AEPID 51
9.2.5 RPID 51
9.2.6 AECAN 51
9.2.7 AEFREE (SNA only) 51
9.2.8 AECOPY 52
9.3 OnLine 52
9.4 Print ID Sequence 52
9.5 COPY Sequence 52
9.5.1 Second Request Processing 53
9.6 QUERY 53
9.7 Device Cancel Sequence 55
9.8 Printer Hold (SNA only) 55
9.9 Printer Cleanup 56
9.10 Useable Copy Protocols 56
9.10.1 PRINT ID 56
9.10.1.1 PRINT ID/Printer Number Request Contention (Race Condition) 57
9.10.2 Matrix Change 57
9.10.3 Copy Data > Print Buffer 58
9.10.4 Copy Rejection 59
9.10.4.1 Prior To Service 59

9.10.4.2 Immediate Rejection - Copy Unauthorized 60
9.10.4.3 Printer Error During Data Transfer 61
9.10.5 DEVICE CANCEL 61
9.10.6 DEVICE CANCEL/WCTL Race Condition 61

10.0 Data Base Operations 63
10.1 Down Stream Loading (DSL) - 3290 Only 63
10.1.1 Device Power On DSL Interaction 65
10.2 3274 Configuration Table 65

11.0 Multiple Interactive Screen Support 69
11.1 TCA Fields 69
11.1.1 3274 Initialization 69
11.1.2 Online-to-host 69
11.2 Modifications 69
11.2.1 AEDV 70
11.2.1.1 AEDV (Online) 70
11.2.1.2 AEDV (Offline) 70
11.2.1.3 AEDV (Dump Complete) 70
11.2.2 AEEB (Security Key ON) 71
11.2.3 AEER 71
11.2.4 WCUS (Write Control Unit Status) 71
11.2.5 WCUS:LUSTATUS (SNA only) 71
11.2.6 Race Conditions 71
11.2.7 WDAT 72
11.2.8 Bisync Inbound/Outbound Contention 72

12.0 Attachment Considerations 73
12.1 SNA 73
12.1.1 ACTPU/DACTPU 73
12.1.2 ACTLU/DACTLU 73
12.1.3 Segments Passed To The Device 73
12.1.4 BIND 73
12.1.5 UNBIND 74
12.1.6 Power Off, TCA Disconnect, or Offline Status Change (AEDV) . . 74
12.1.7 Outbound Segmenting 74
12.1.8 Inbound Segmenting 74
12.2 BSC 75
12.2.1 Inbound Operation 75
12.2.2 Inbound Processing 76
12.2.3 Subsequent AID Actions 76
12.2.4 Operator Reset Action 76
12.2.5 Transparency 77
12.2.6 Test Request Key 77
12.2.7 Asynchronous Host Select 77
12.2.8 Outbound Processing 77
12.2.9 Security Key 78
12.3 NON-SNA LOCAL CHANNEL 78
12.3.1 Inbound Operation 78
12.3.2 Subsequent AID actions 79
12.3.3 Operator Reset Action 79
12.3.4 Test Request Key 80
12.3.5 Asynchronous Host Select 80
12.3.6 Host Commands Other Than Read Modified 80
12.3.7 Outbound Processing 80
12.3.8 Security Key 81
12.3.9 Miscellaneous 81

13.0 Communications and Systems Management (C&SM) 82
13.1 Response Time Monitor 82
13.2 Alert Function 82
13.3 Expanded Configuration RECFMS(05) Support (3290 Only) 82

1.0 INTRODUCTION

This document describes the attachment interface between a Distributed Function Device (such as the IBM 3290 Information Panel or an IBM 3270-PC) and a properly customized IBM 3274 Control Unit via a coaxial cable as an extension to the hardware interface described in the Product Attachment Information manual "IBM 3274, 3276 Control Unit to Device". A Distributed Function Device attaches to 3274 Type A adapter ports (except port 0) and uses the Device Cluster Adapter (DCA) transmission protocol. A program in the control unit communicates with a program in the Distributed Function Device through a portion of shared memory in the device which is addressable from the control unit by DCA commands. Distributed Function Devices are solicited by POLLing and are requested to perform functions by means of several DCA commands which cause program interrupts. Host data streams are treated as pass-thru data to the device constituting a function split which makes the control unit largely independent of the device and of functional characteristics of its data streams.

The flexibility of this interface permits the Distributed Function Device to logically configure up to 5 Logical Terminals which can communicate with a host concurrently from different applications independently to different device/LU addresses.

The terms "Distributed Function Device," "Device," and "TCA Device" (Terminal Control Area) are interchangeable.

The following publications are listed for reference and may be useful in understanding this document.

IBM 3270 Information Display System, 3274 Control Unit Description and Programmer's Guide, GA23-0061.

IBM 3290 Information Panel, Description and Reference, GA23-0021.

3270-PC Control Program User's Guide and Reference, (SC23-0103).

1.1 BASIC OPERATION

Communications between the Distributed Function Device and the 3274 control unit (CU) is via a 128 byte Terminal Control Area (TCA) within the device's buffer. Requests are made by the control unit by placing the function request and the necessary parameters in the proper TCA locations and then telling the device to execute the operation via a START OP coax command. On write type data transfer operations, the data must reside in the device's buffer before the request execution is initiated. On read type operations, the device places the data in the specified buffer locations as part of processing the request.

The device then performs the requested function and tells the control unit when it has completed or terminated. A completion code (Synchronous Status) is posted by the device to indicate whether or not processing completed normally. The control unit then reads the completion code and processes it as required.

The device may also make asynchronous requests (Asynchronous Status) to the control unit by placing a request code in the TCA and telling the control unit that the request is present. The control unit then reads the request code and processes it when internal contention conditions allow. Processing by the control unit consists of acknowledging the request and issuing more function requests as required to service the device. Once the request is acknowledged, the device is free to present another asynchronous request.

In addition, devices may also present prioritized Expedited Status in the TCA buffer for functions which must be processed on an immediate basis.

1.2 INTERFACE STATES AND TCA OWNERSHIP

TCA ownership is directly connected with interface states. Violation of ownership rules constitutes an interface synchronization error. If detected, the interface may be forced to disconnect by the offended party.

1.2.1 INTERFACE DISCONNECTED

While in this state, the device owns the entire TCA and data buffer. The interface is considered disconnected when the control unit no longer services device requests or status or the device does not answer POLLs. At any given point in time, it may be impossible for the device to tell what the connection state of the interface is. Excluding a POSITIVE indication to the contrary, the device should always consider the interface as connected until it can determine otherwise beyond a reasonable doubt. This state is exited when the device generates a POR.

The device must have a positive indication that the interface is in this state prior to sending an unsolicited POR to the control unit. Positive indications that the interface is disconnected include:

The device has stopped answering POLLs for a period of time in excess of 10 milliseconds. This causes The control unit to logically disconnect the interface. This condition includes physical power off.

The device receives a TERMINAL RESET Command. The control unit only issues this command when the interface is disconnected.

The CUDSER (Device Specific ERror code) field in the TCA is not 0. The control unit writes a non-zero value in this field when the interface is logically disconnected because a device specific error was detected.

NOTE:
The control unit treats a POR that is received while the interface is not disconnected as an interface synchronization error and force the interface to be disconnected. The next POR is then treated normally.

1.2.2 INTERFACE CONNECTED

This state is entered when the device sends a POR to the control unit. While in this state, the device and the control unit share the TCA and data buffer. At any given point in time, each location is owned exclusively by EITHER the control unit OR the device. When a location is not owned, it may not be altered. Ownership is generally determined by whether or not the interface is idle (no operation in process).

1.2.3 INTERFACE CONNECTED AND IDLE

The interface is considered idle after the device has posted synchronous status for a control unit requested function. It is also considered idle between the time POR is sent to the control unit and the START OP is received for the first function request.

While in this state, the control unit owns locations X'40' to X'7F' in the TCA and all of the data buffer. The device owns locations X'00' to X'3F' with the following exceptions:

- DPASTAT - The device owns this location until it sets a value of X'01' indicating that asynchronous status is present. At that point, ownership belongs to the control unit until it (the CU) sets the location to 0 to acknowledge the asynchronous status. Ownership then returns to the device. The device must set the Asynchronous Status and parameters values in DALTAD thru DAEP4 prior to setting DPASTAT to X'01'.

- DPSSTAT - This field indicates the validity of DSSV thru DSSP3 to the control unit. With SNA overlapped operation, it is necessary to stack Synchronous Status in the TCA and a positive indication of its validity is required. The CU sets this field to 0 when a Function Request is presented to the device and a START OP is issued. The device sets an X'01' in this field when the request completes or terminates and Synchronous Status is available. The device must set the Synchronous Status and Parameter values prior to setting DPSSTAT to X'01'. While this field is X'01', it is owned by the control unit. The device owns the field while its value is 0.

- CUDSER - This field is initialized to 0 by the device prior to sending a POR. The control unit writes a non-zero specific error code (see section 4.6) in this field when the interface is disconnected due to an error. This location is always owned by the control unit while the interface is not in a 'disconnected' state.

- EXFAK - This field indicates the validity of fields EXFLT thru EXFP4 to the control unit. Certain device requests must be serviced on a priority basis whether the device is active or idle. Expedited Status (ES) has service priority over normal asynchronous status. ES can be processed while the device is in an active state. When the request has been serviced the control unit acknowledges ES by resetting the unacknowledged request flag in EXFAK. The device must set the request fields before posting status to X'01' in EXFAK. While this field is X'01' it is owned by the control unit. The device owns the field while its value is 0. The control unit may load response parameters in fields EXFD1 thru EXFD4 before acknowledging ES status. ES status is acknowledged by issuing a READ Terminal ID, causing a TCA interrupt in the device microprocessor. READ Terminal ID is used in this context as an "Alternate Start Operation" command.

1.2.4 INTERFACE CONNECTED AND ACTIVE

This state is considered active from the time that the control unit receives a clean response to a POLL on a START OP command queue until the time when the device posts Synchronous Status in DSSV. While in this state, the control unit owns only the data buffer exclusive of any area specified by request parameters on data transfer type requests.

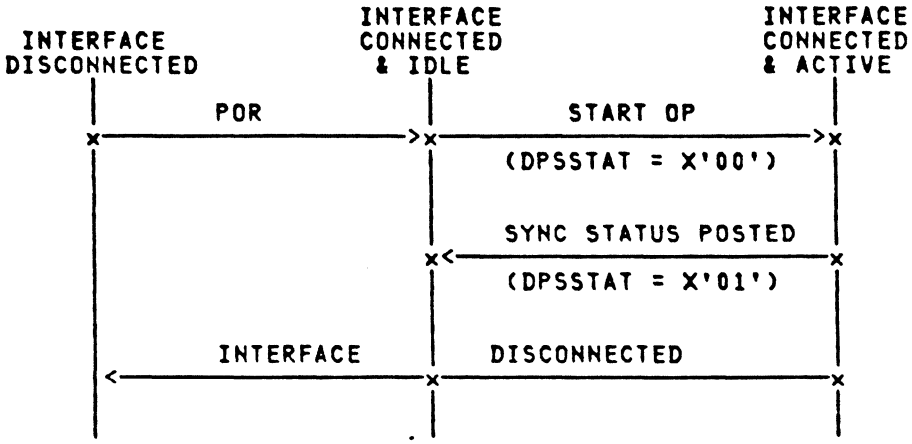
NOTE:
A "command queue" is a series of commands and data that are issued by the control unit to a device without an intervening POLL command. There may or may not be "ending sequences" between the commands and/or data (see section 3.6).

Contention in this area may be caused by a command queue retry by the control unit which may cause locations X'40' to X'47' to be rewritten. Contention is avoided by issuing separate command queue (1) to write to the TCA and (2) to issue the START OP. If the Write command queue fails, rewriting data to the TCA on retries is not noticed by the device since it only examines control unit altered TCA locations when a START OP is received. If the START OP command queue fails, the retries are detected by the device by examining the CUSYN value. If CUSYN has not been toggled, the START OP is a retry and should be ignored. If a START OP is received while one is in process and CUSYN has been toggled, an interface synchronization error has occurred and should be reported to the control unit by the device.

1.2.5 RESERVED FIELDS

Reserved fields in the TCA are excluded from the above discussion. While the interface is connected, reserved fields in the TCA between X'00' and X'3F' belong to the device. Fields between X'40' and X'7F' belong to the control unit. Reserved fields owned by the device must be set to zero. Reserved fields owned by the controller should not be checked by the device.

1.3 INTERFACE STATE DIAGRAM



1.4 DEVICE STATES

Device states reflect the active level of communications between the device and an upstream entity.

1.4.1 OFFLINE TO CU

This state is equivalent to the 'interface disconnected' state described previously. The control unit is no longer servicing requests or status from the device or the device is not responding to POLLs. The device is not recognized by the control unit.

1.4.2 INITIALIZATION IN PROCESS

This state exists between the time that the device sends a POR to the control unit and the time that the control unit posts '3274 READY' via WCUS. While in this state, the control unit performs the initialization necessary to be capable of recognizing the device. Once the TCA is initialized properly, the device is capable of processing Function Requests.

1.4.3 ONLINE TO CU

This state is entered when '3274 READY' is posted to the device via WCUS. While in this state, the device is recognized by the control unit, but may not communicate to the host. The device may request functions from the control unit to utilize locally owned resources, but may not attempt to communicate to the host. The control unit discards any outbound transmissions for the device. This state may also be entered when the device requests that it be taken offline from the host via AEDV.

Device Function Requests that may be utilized by the control unit while in this state are limited to CNOP, WCUS, RDBD, WDBD, RDCOPY, RPID, and WCTL. The device must respond with ERFR Synchronous Status to any other Function Request it receives.

In this state asynchronous status requests of AEEB and AEEP are invalid.

1.4.4 PENDING STATES

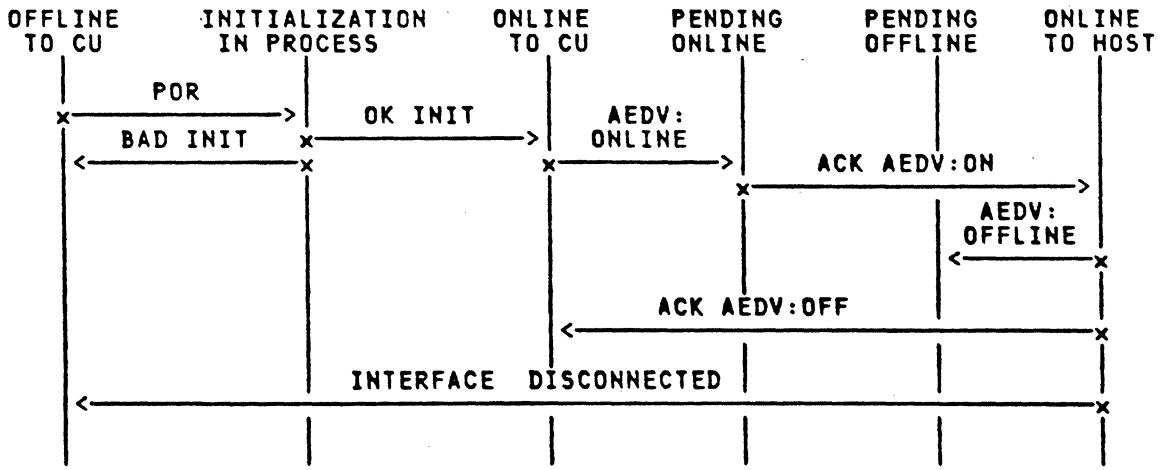
Online/offline to host requests (AEDV) remains pending until the control unit acknowledges asynchronous status.

1.4.5 ONLINE TO HOST

The CU puts the device in this state when specifically requested to do so via AEDV. The device remains in this state until (1) it is returned to an online to control unit state via AEDV or (2) the interface is disconnected. While in this state, the device communicates with the control unit as described for the online to control unit state. In addition, upstream communications with the host are now allowed and outbound transmissions are forwarded to the device.

In this state asynchronous status requests of AEDBA and AEDBS are invalid.

1.4.6 DEVICE STATE DIAGRAM



2.0 FUNCTION SPLIT

This interface must operate in three distinct environments:

1. A 3270 protocol BSC control unit.
2. A 3270 protocol local channel attachment control unit.
3. A FID2 SNA control unit.

NOTE:
The CU serves primarily as a multiplexor converting host link protocols to coax protocols and vice versa.

The function split for each of these is as follows:

2.1 COMMON

Control Unit	Device
Error Logging	I/O Event Initiation
Indicator event status	Device error reporting
Power on/off	Device RAS and testing
Hung device detection	Local Function
Communication area management	Operator indicators
Limited CU file access	Hung CU detection

2.2 BSC

Control Unit	Device
BSC protocols including:	Data Stream from STX to EOT/EOB
Transparency	Read Command Detection
Select	
Specific POLL	
General POLL	
Inbound Blocking	
Line Control	
Line hit recovery	
Test header creation	

2.3 LOCAL CHANNEL

Control Unit	Device
Local Channel protocols including CE, DE	Channel Command processing
	Data Stream processing
	Test header creation

2.4 SNA

Control Unit	Device
PU Services	All other SNA functions
ACTLU/DACTLU processing Session termination on power off (single session) Outbound Routing	ACTLU parameter processing

3.0 DEVICE HARDWARE

The CU is the owner of the coax and controls all flow across the coax interface. In general, it POLLs the device for status changes, writes to the device, or reads from the device. The device is required to have an addressable buffer which is accessed by coax Read and Write commands. Higher level functions are communicated via data placed in the buffer. The CU initiates a function by writing data into the device buffer and then telling the device to interpret the data. The device initiates a function by placing data in the buffer and responding to a POLL with status requesting the CU to interpret the data.

3.1 GENERAL DESCRIPTION

Data to be transmitted from a controller to a device or device to controller is carried on a single coax line per device. The coax type is RG62AU with a maximum length of 1.5 kilometers. Data is transmitted in a serial by bit fashion using a binary dipulse technique. (See paragraph 3.6 for coax transmission protocol.)

Data to be transmitted over the coax has a bit rate of 2.3587 MHz. Communication is as follows:

Twelve (12) bits are assembled to form one (1) twelve (12) bit word for transmission in either direction over the coax. The first bit of the twelve (12) bit word is used to delimit successive words from the controller and is always a "one (1)" bit and are referred to as the "Sync bit". The last bit of each twelve (12) bit word is the parity bit that maintains even parity when added to the preceding eleven (11) bits. Word groups of twelve (12) bits each may be contiguous. In this case, the sync bit of the next word must directly follow the parity bit of the preceding word with no intervening pad bits. A word from the controller to the device (display or printer) is a command or data word. Each Write type command causes a Transmission Turnaround / Auto Response (TT/AR) following the last word of each group of contiguous words sent from the controller, and the device responds with clean status (bits 1 and 12) if the word(s) was (were) received without a Transmit Check. A word from a device in response to a Read type command is either data or a status word. The device must begin response (data, status or TT/AR) within 5.5 microseconds after receiving the ending sequence from the controller (both read and write type commands.) The 5.5 usec. is measured from the end of the last bit time of the received ending sequence to the beginning of the first bit time of the transmitted starting sequence.

The 12 bit command word from the controller to a device contains address bits (all zero) and a command code. The address portion of the command word is three bits in length (Bits 2,3,4) which provides five bits for command codes (Bits 5,6,7,8 and 9).

Reserved bits in all commands and responses are zero.

3.2 WORD FORMATS

3.2.1 COMMAND WORD TO DEVICE

1	234	56789	10	11	12
SYNC BIT	YYY ADDR.	XXXXX CMND	0	1 CMND.	X Parity

3.2.2 DATA WORD TO DEVICE (BIT 2 IS MOST SIGNIFICANT)

1	2345	6789	10	11	12
SYNC BIT	XXXX DATA	XXXX WORD	X *	0 Data	0 Parity

* Bit 10 is a parity bit (odd) for the preceding eight bits.

Data Words of less than 8 significant bits are right justified (by the controller) and the high order bits set to zero.

3.2.3 STATUS WORD TO CONTROLLER

1	2345	6	7	8	9	10	11	12
SYNC BIT	0000	X	0	0	X	0	0	X
		(		STATUS	BITS		)	PARITY

OR:

1	2345	6789	10	11	12
SYNC BIT	SPECIAL STATUS		1	0	X PARITY

A status word is always sent (in response to a POLL command) from a device that has power on and has completed its POR sequence. (Prior to receiving POR Response from a device, the controller holds the device 'deactivated.' The controller POLLs the device but ignores any response except POR Response.) A response of all zeros except for bits 1 and 12 indicates that there are no error conditions to be reported up line and no operator activity requiring service. This response is referred to as TT/AR or a "clean" response.

3.2.4 DATA WORD TO CONTROLLER (BIT 2 IS MOST SIGNIFICANT)

1	2345	6789	10	11	12
SYNC BIT	XXXX DATA	XXXX WORD	P *	0	P Parity

*Bit 10 = Parity bit (odd) for the eight bit (2 thru 9) data word for Read Data and Read Multiple commands.

Data Words of less than 8 significant bits must be right justified (by the device) and the high order bits set to zero.

3.3 COMMAND CODES TO DEVICE

The following commands apply to the base address '000' (bits 2 - 4). Commands addressed to a non-existent feature must be treated exactly like reserved commands (see note below).

READ COMMANDS (XXXX1) 56789		XXX11: Response Parity Checked XXX01: " Not " "
00001	POLL	
00011	READ DATA	
00101	Reserved	
10101	Reserved	
01001	READ TERMINAL I.D.	
10001	POLL/ACK	
10011	Reserved	
01101	Reserved	
11001	Reserved	
01011	READ MULTIPLE	
10111	Reserved	
01111	Reserved	
00111	Reserved	
11011	"	
11101	"	
11111	"	

WRITE COMMANDS (XXXX0) 56789	
00000	Reserved
00010	RESET
00110	Reserved
01100	WRITE DATA
01010	Reserved
00100	LOAD ADDRESS COUNTER HIGH
10100	LOAD ADDRESS COUNTER LOW
01000	START OPERATION
11010	Reserved
11100	DIAGNOSTIC RESET
01110	Reserved
11000	"
10000	"
10010	"
10110	"
11110	"

NOTES:
In response to the reserved read commands, the device must return an all zero data word with bad parity (bits 2 thru 10 all zero) regardless of bit 8 in the read command.

The reserved write commands reset the previous command. If no other command or data word directly follows the reserved command, TT/AR takes place.

3.4 READ COMMAND FUNCTIONS

3.4.1 00001-POLL AND 10001-POLL/ACK

The POLL command (Hex 1) does not use the address portion of the command word for address. Bits in the address portion are unused.

Bits 2,3,4 = insignificant
Bit 5 = ACKnowledge last input message to controller.

1	2	3	4	5	6	7	8	9	10	11	12
SYNC BIT	-	-	-	X	0	0	0	1	0	1	P
					(POLL Command)						

The response word to a POLL is a one word status response. If a non-zero status word is sent to the controller, the device should anticipate receiving a POLL/ACK to acknowledge the acceptance of the first status word, cause the device to respond with "clean" status and reset the previously returned status bits. Upon receipt of the clean status response the controller may issue another POLL, without the ACK bit, and the device must respond with the second status word. If the second POLL does not have the ACK bit on, the device must respond with the first status word again. Repetitive POLLing and POLL ACKing of the device may continue until an all zero status response to a POLL is received at the controller or the controller reaches an error threshold.

The priority of POLL response is:

1 POR complete Special status code.
2 Base Status (Bits 6,9) *

* Multiple bits of base status may be returned in a POLL response. If a Status Bit is returned and not ACK'd, the same bit must be returned in the next POLL response. The other status bit may be added to a previously returned status bit if a POLL is received prior to receipt of a POLL/ACK.

If there is no status to send, an all zero POLL response is sent indicating that service is not required at the device and the controller is released to POLL the next device.

3.4.2 RESPONSE TO POLL (STATUS WORDS)

The status response word from the device is:

1	2345	6	7	8	9	10	11	12
SYNC BIT	0000 ADDR	STATUS TRANS	RES 0	RES 0	RES 0	SPECIAL STATUS	RESERVED 0	PARITY BIT

Bit 1 = Sync Bit

Bits 2, 3, 4, 5 = 0 (address)

Bit 6 = 1 Status (synchronous or asynchronous) has been set in the Status Register. See sections 6.0 and 7.0

Bit 7 = Reserved

Bit 8 = Reserved

Bit 9 = Reserved

Bit 10 = 1 Redefines bits 2 thru 9 as being Special status.

Special status codes are:

2345 6789

0000 0010 Device has powered on since last POLL. This code is sent only in response to a POLL received after a power on (or Reset Command) sequence is complete. See Reset Command.

Bit 11 = Reserved

Bit 12 = Parity Bit - maintains even parity of the preceding eleven (11) bits.

3.4.3 OTHER READ COMMANDS

Each of these commands causes the device to return one or more Data Words. The ending sequence must follow the 12th (P) bit of the last Response word.

00011 READ DATA
The read data command causes the addressed device to respond with one data word from storage at the current address counter value. The address counter steps up once at the completion of the command.

01011 READ MULTIPLE
This command causes the device to respond with one or more data words from storage beginning at the current I/O address counter value. The read terminates (with ending sequence) when the two low order bits of the I/O address counter step to 00. A maximum of four bytes is returned.

01001 READ TERMINAL ID
This command causes the device to respond with one data word. In addition, the command interrupts the device processor. The control unit acknowledges Expedited Status using Read Terminal ID as an 'alternate Start Op' mechanism.

The format of the response data word is as follows:

1	2 3 4 5	6 7 8	9	10	11	12
SYNC BIT	0 0 0 0	0 0 0	1	0	0	P
	(Distributed	Function	Device	)		

3.5 WRITE COMMAND FUNCTIONS

Many of the Write Commands are defined as being followed by one (or more) bytes of data. The device executes the command following receipt of the data byte. If

a second command is received instead of the data byte for the first command, the first command is lost and the second command sequence started. Write type commands remain active until reset by the next command (including POLL). Data sent while no command is stored is lost with TT/AR being returned.

00010 RESET

In a TCA device, the RESET command resets any pending status in the coax adapter and interrupts the microprocessor. The microprocessor then terminates any operation in process and causes the adapter to respond to a POLL with the POR complete status code. The Adapter is then able to accept and execute any valid command. The message buffer must be cleared, and the controller output area is cleared. The following portion of the Device Output Area is initialized (reference section 3.8.1):

Bytes 0 thru B:	All zero.
Bytes C thru 11:	Terminal ID bytes initialized.
Bytes 12 thru 7F:	All zero.
Bytes 80 thru 9F:	Device Information (section 4.1)

NOTE:
POR Complete must not be returned if the reset (either Command, Power On, or operator initiated) 'failed', that is, if the device is broken.

This command is only issued during error recovery and the Controller IML sequence. The device must be capable of accepting two or more successive Reset commands (without intervening POLL commands) and respond with a single POR Response to a subsequent POLL. Prior to returning POR Response the device may terminate communication with the controller.

01100 WRITE DATA
The WRITE DATA command causes the device to accept all following data words for storage in the buffer until another command is received. The data to be stored in the buffer must be loaded at the location indicated by the address counter. The address counter must step up once for each data word received.

NOTE:
The controller is responsible for preventing address overflow while writing (or reading) the device buffer.

10100 LOAD ADDRESS COUNTER LOW
This command, followed by one data word, loads the 8 bits of the data word into the 8 low order bits of the address counter.

00100 LOAD ADDRESS COUNTER HIGH
This command, when followed by one data word, loads the data word into the high order bits of the address counter.

01000 START OPERATION
This command is used to invoke processing of a function request. The coax adapter interrupts the microprocessor. Upon completion of the operation the device processor stores synchronous status in the TCA buffer and sets "status available" in the POLL response. To prevent controller microcode timeout, the device must complete the operation within a specified time (see section 4.6.3).

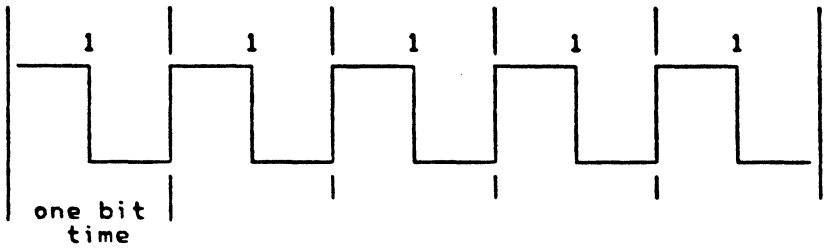
11100 DIAGNOSTIC RESET
This command is similar to the RESET command discussed above and is intended for service only.

3.6 COAX TRANSMISSION PROTOCOL

The dipulse technique is controlled by the driver receiver logic that guarantees a voltage transition of the coax at mid-bit time. Prior to valid data being transmitted, the coax must be conditioned to ensure that bit and byte synchronization can be achieved. This requires the transmission of a line quiesce and code violation pattern which is generated by the coax driver logic.

3.6.1 LINE QUIESCE PATTERN

It is necessary to establish an equilibrium switching condition on the line after the null condition of line turn around before valid data can be properly detected at the receiver. Each data sequence from either controller or device after line turn around is therefore preceded by the following 5 bit biphase encoded data.

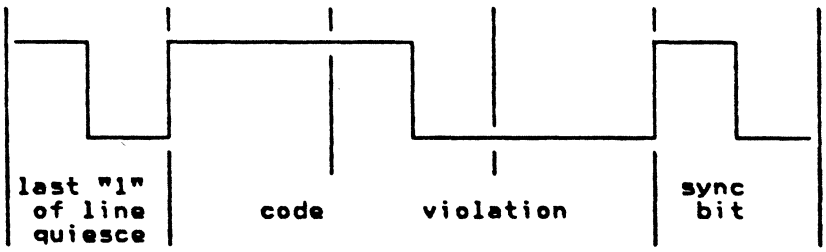


The bit polarity is shown at the logic to Driver/Receiver interface. Polarity on the coax is inverted.

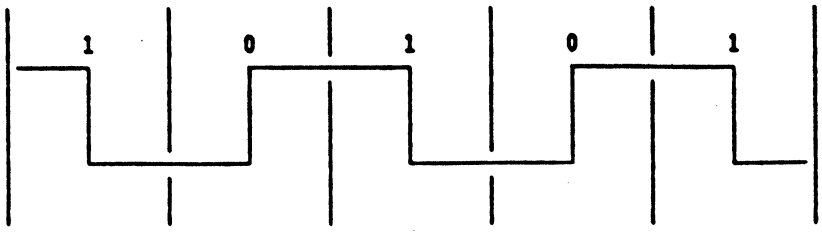
3.6.2 UNIQUE CONTROL CODE VIOLATION

A code violation follows the line quiesce pattern to differentiate between the quiesce pattern and the start of the valid data following the code violation. This is necessary because, due to varying line lengths, it is not possible to predict where the received data becomes valid. However, the code violation is received properly and provides a clean reference mark for start of transmission.

A unique balanced code violation sequence containing leading and trailing buffer bits to eliminate history dependence on adjacent data appears as follows:



The trailing buffer bit is actually the sync bit of the following data byte. This code violation is unique in that it contains pulse widths (1 1/2 bit pulse widths) not present in normal biphase data (1/2 or 1 bit pulse widths) shown here for comparison.

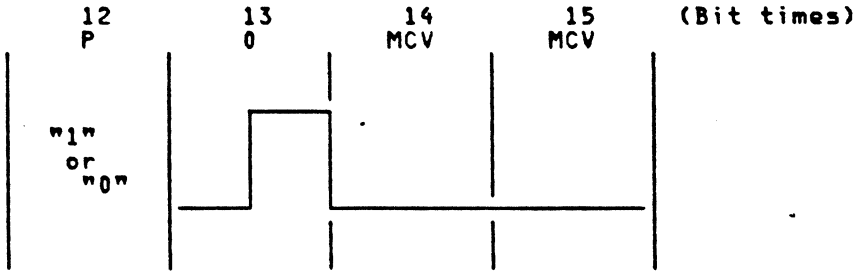


Each bit has mid-bit transition. Thus, once decoded, this code violation provides, in addition to a reference mark for start of transmission, an unequivocal definition of bit boundaries.

A means is provided for re-establishing line synchronization with the device by using the receipt of a legitimate code violation to reset the device's SERDES.

3.6.3 TRANSMISSION TERMINATION SEQUENCE (MINI-CODE VIOLATION - MCV)

In order that the receiver demodulation logic is reset at the end of a transmission, so that a subsequent transmission may be properly demodulated, a special termination sequence is used:



Ending Sequence

The last byte of data transmitted has 15 bits. The first 12 bits are as previously defined (starting with sync and ending with a parity bit). The thirteenth bit is a zero followed by two bit times without a mid-bit transition. (These are referred to as mini-code violations.) The first mini-code violation is always used to reset the receiver logic. The second merely guarantees that the line does not discharge and generate a spurious clock pulse while the logic is detecting the first MCV. The zero in the thirteenth bit position allows for discriminating a transmit check condition, generated as a result of illegally padded zero bits between bytes, from a normal ending sequence.

3.6.4 COAX TRANSMISSION WAVEFORMS

Bits on the coax appear as positive and negative going pulses. Binary data is phase encoded such that a 212 nanosecond (ns) up-level followed by a 212 ns down-level represents a binary 0. Similarly, a 212 ns down-level followed by a 212 ns up-level represents a binary 1. A predistorted pulse is generated for every transition from an up-level to a down-level or vice versa. (See waveforms in Figures A and B.)

The waveforms shown in Figures A and B are the signals measured across the coax at the transmitting unit (either control unit or device). The waveforms shown in Figures C and D show the signal across the coax at the receiving end of 1.5 kilometers of coax.

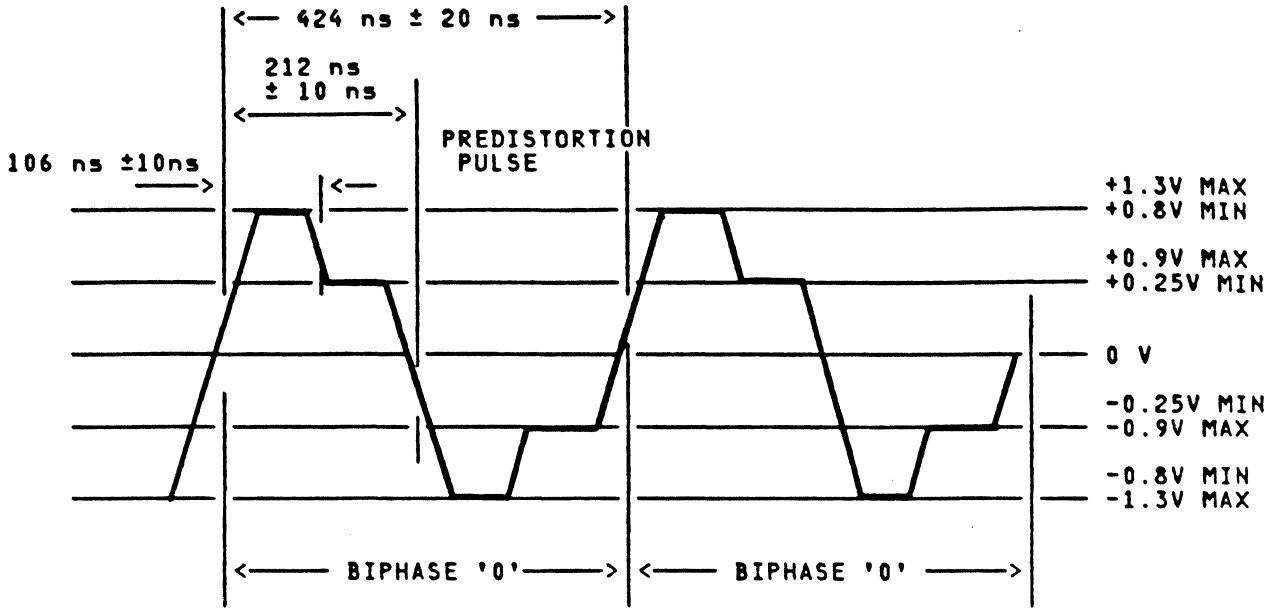


Figure A (Waveform at Transmitting Unit)

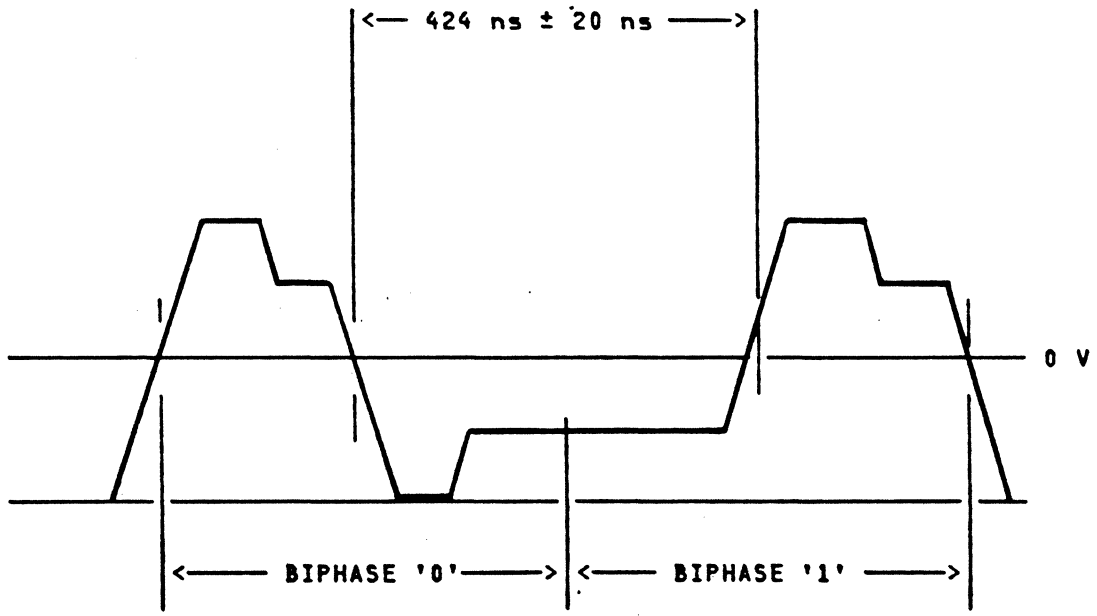


Figure B (Waveform at Transmitting Unit)

All Rise and Fall Times 30 ns MAX.
Rise and Fall Times are Exaggerated For Clarity.

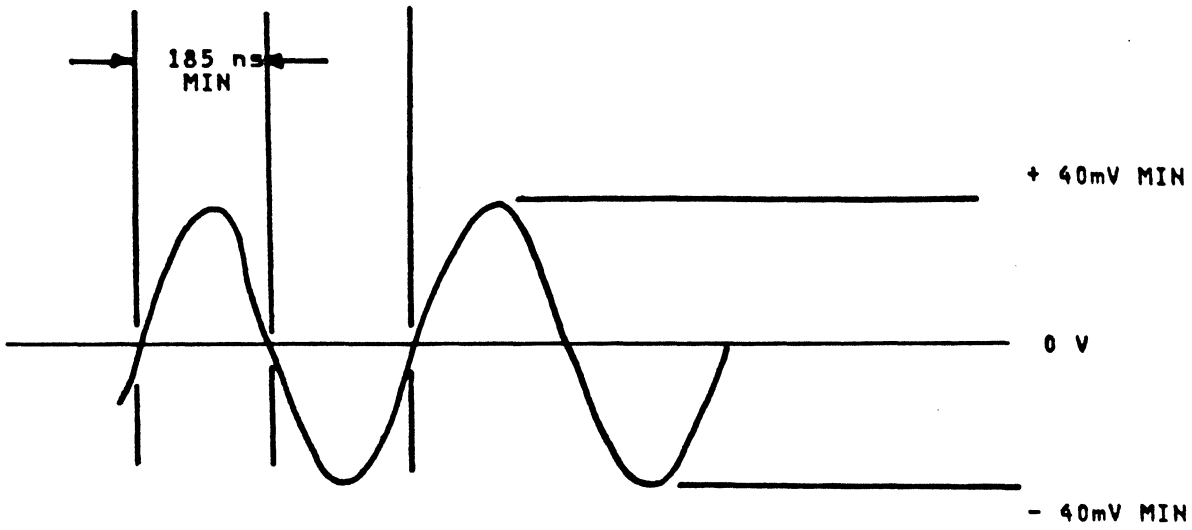


Figure C (Waveform at Receiving End of 1.5 Kilometers)

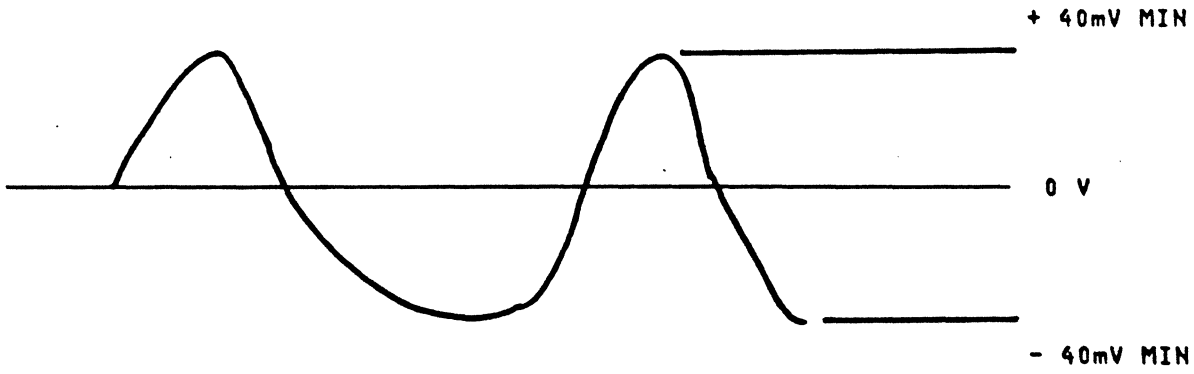


Figure D (Waveform at Receiving End of 1.5 Kilometers)

3.7 TRANSMIT CHECK

A Transmit check is defined as follows:

1. A 0 in the sync bit location not followed by the mini-code violation.
2. The loss of mid-bit transition detected at other than normal ending sequence time.
3. A transmission parity error (bit 12 not being even.)

When a transmit check is sensed in the device, the device must cease accepting all data and all commands, and must suppress the TT/AR. The stored command, if any, must not be reset. Normal operations must resume upon receipt of the next Line Quiesce/Code Violation.

The controller also tests the same three conditions to provide for error recovery.

3.8 DEVICE BUFFER

The Device buffer must be at least 4096 for SNA attachments and 8192 bytes for non-SNA attachments to offload control unit buffers and achieve concurrent line and control unit utilization with device processing. In SNA environments, outbound transmission to different logical units may be interleaved in the queued TCA buffer. The outbound pacing parameters of the aggregate active Logical Units must be considered for desirable subsystem performance since the 4096 buffer is shared by all active Logical Units. In the non-SNA environment data must be buffered in the TCA to free the transmission line and make the control unit available to other devices on the cluster while the device processes the data asynchronously. The device buffer must be byte addressable by SACH and SACL from 0 to the buffer size - 1. The buffer is split into two logical sections. Locations X'00' to X'7F' are fixed format and the remainder is allocated at the discretion of the CU.

NOTE:
The 3270 PC currently implements a 4096 byte TCA buffer for non-SNA attachment.

3.8.1 DEVICE BUFFER FORMAT

NAME	ADDRESS	DESCRIPTION
DPASTAT	X'00'	Asynchronous status present flag X'01' indicates last asynchronous status is unacknowledged. X'00' indicates last asynchronous status is acknowledged.
DPSSTAT	X'01'	Synchronous status present flag X'01' indicates last synchronous status is unacknowledged. X'00' indicates last synchronous status is acknowledged.
DSSV	X'02'	Synchronous status value.
DSSP	X'03'	Synchronous status parameter.
DSSP2	X'04'	Synchronous status parameter.
DSSP3	X'05'	Synchronous status parameter.
DALTAD	X'06'	Logical Terminal Address (to host)
DAEV	X'07'	Asynchronous status event value.
DAEP	X'08'	Asynchronous status event parameter #1
DAEP2	X'09'	" " " " #2
DAEP3	X'0A'	" " " " #3
DAEP4	X'0B'	" " " " #4
DTID1	X'0C'	Terminal ID (see Note 1) B'xxxxxx10' Distributed Function Device with a TCA buffer.
DTID2	X'0D'	Reserved - Terminal ID extension. - Set to X'00'
DTID3	X'0E'	Reserved
DTID4	X'0F'	Reserved
DBUF	X'10'-X'11'	Device buffer size in bytes (valid after power on, AEDV:online & AEDV:offline)
	X'12'-X'1F'	Reserved-must be 0.
EXFLT	X'20'	LT Address (or X'FF' if physical device)
EXFRQ	X'21'	Expedited Status value
EXFP1	X'22'	ES status Parameter 1 (if needed)
EXFP2	X'23'	ES status Parameter 2 (if needed)
EXFP3	X'24'	ES status Parameter 3 (if needed)
EXFP4	X'25'	ES status Parameter 4 (if needed)
EXFAK	X'26'	Post/Acknowledgment flag byte X'01' indicates last Expedited Status is unacknowledged. X'00' indicates last Expedited Status is acknowledged.

NAME	ADDRESS	DESCRIPTION
	X'27'-X'3F'	Reserved-must be 0.
CUDP	X'40'-X'41'	Data address within the device buffer. (Must be aligned on HW boundary).
CULTAD	X'42'	Logical Terminal Address (from host)
--	X'43'	Reserved
CUFRV	X'44'	Synchronous function request value.
CUSYN	X'45'	Request synchronization switch (toggle).
CUFRP	X'46'-X'49'	Synchronous function request parameters.
CUFRP1	X'46'	Parm 1
CUFRP2	X'47'	Parm 2
CUFRP3	X'48'	Parm 3
CUFRP4	X'49'	Parm 4
--	X'4A'-X'4F'	Reserved parameter area.
CUDPORT	X'50'	Device port number (1 - 31)
CUAT	X'51'	Control unit host attachment protocol (see Note 2)
CUDSER	X'52'-X'53'	Error code value for last-ditch-command-queue.
CULTA1-5	X'54'-X'58'	MIS Addresses. See section 11.1.1.
	X'59'-X'5B'	Reserved.
EXFD1	X'5C'	ES Response parameter 1 (if needed)
EXFD2	X'5D'	ES Response parameter 2 (if needed)
EXFD3	X'5E'	ES Response parameter 3 (if needed)
EXFD4	X'5F'	ES Response parameter 4 (if needed)
EXTIME	X'60'	Host transaction timing X'01' indicates Host Timing mode is in effect. X'00' indicates Device Timing mode is in effect.
	X'61' -X'7D'	Reserved.
CUSLVL	X'7E'-X'7F'	Controller TCA Support Level (See Note 3 and section 3.9)
CUDATA	X'80'-max	Data Area (see section 4.1).

NOTE 1:

DTID1 aligns on address with PCIA printer ID. Bit 7 must be off for TCA devices. Bit 6/7 (xxxxxx10) indicates that the device is a Distributed Function Device. The TCA buffer size must be 4K for SNA, 8K for Non-SNA. Bit 0 has no function. It may be set to 0 or 1. Bit 1 may only be set in response to a Diagnostic Reset command indicating that the device is "Ready to Dump." Bits 2 - 5 are reserved. DTID2-DTID4 are reserved.

NOTE 2:

CUAT (attachment protocol) has following flags defined:

Bit 0
- = 0 TP attached
- = 1 local attached

Bit 1
- = 0 SNA protocol
- = 1 non-SNA protocol

(All other bits are reserved)

NOTE 3:

CUSLVL is a two-byte field that indicates the optional TCA functions supported by the controller. It has the following flags defined (see also section 3.9.

X'0000' - Base TCA Function Support Level (see section 3.9.1)
Bits 0-B - Reserved
Bit C - 1 = Device Initiated UNBIND Option Support (see section 3.9.2)
Bits D-F - Reserved

3.8.2 DEVICE BUFFER CONTROL

Bytes X'00' - X'3F' are owned and set by the device. They are not altered by the CU except DPASTAT, DPSSTAT and EXFAK are reset by the CU to acknowledge receipt of Asynchronous/Synchronous/Expedited status respectively. For details see section 1.2.3.

Bytes X'40' - X'7F' are owned and set by the CU. They are not altered by the Device except for being initialized to zero by power on or the RESET command.

Bytes X'80' - n are controlled and allocated by the CU and are altered by the device only on request from the CU. The data in this area is preceded by a four byte Message Header (except when a device has posted a POR).

Length-2 bytes. Length of the data plus flag and length fields.

Flags-2 bytes:		VALID OUT BOUND (from device)	VALID IN BOUND (to device)
Bit 0 = 1	first of message	x	x
= 0	not first of message	x	x
Bit 1 = 1	last of message	x	x
= 0	not last of message	x	x
Bit 2 = 0	except for WCTL (see section 5.12)		x
Bit 3 = 1	BSC transparency required	x	x
= 0	BSC transparency not required	x	x
Bit 4 = 1	inbound BSC test request function CU generates actual %/ header on message		x
= 0	not a test request function.		x
Bit 5 = 1	SNA channel buffer overrun (RU size exceeds Bind max. specification 1536)	x	
= 0	SNA complete record	x	
Bit 6 = 1	Data wraps to location X'80' (SNA only)	x	
= 0	No wrap		
Bit 7	Reserved		
Bit 8 = 1	FMH present (Local Copy) (see section 9.6)	x	
= 0	No FMHP present		
Bit 9 = 1	Query Reply Expected (Local Copy) (see section 9.6)	x	
= 0	Query Reply not Expected		
Bit 10-15	Reserved		

For SNA attach, outbound data requested by WDAT does not use bits 0 and 1. Segmented data is indicated by the transmission header.

3.9 CONTROLLER TCA SUPPORT LEVEL

CUSLVL indicates to the device any optional TCA functions which are supported by the controller. A device may connect to a controller that supports options that it (the device) does not support since all such options are downward compatible. However, if a device attempts to use a particular option that is not supported by the controller it is attached to, that device may cause a synchronization error and be disconnected from the controller with a 240 machine check.

3.9.1 BASE TCA SUPPORT LEVEL

A value of CUSLVL equal to zero indicates that the controller does <u>not</u> support any options.

3.9.2 DEVICE INITIATED UNBIND SUPPORT OPTION

CUSLVL Bit 'C' set to '1' indicates that the controller supports receipt of a device initiated UNBIND.

4.0 DEVICE CONTROL

Device control is composed of interchanges between the CU and the Device in a particular sequence to achieve a desired state. This section defines the rules governing those interchanges.

4.1 INITIALIZATION

Synchronization of "power on" is a CU responsibility. If a device is powered on before the CU, then the CU must issue a Reset command to that device to force power on initialization. At the time a "power on" special status is sent to the CU, the device must have locations X'00' - X'7F' of its buffer set to zero except DTID1 (Terminal ID field) and DBUF which are set to their appropriate values. The CU identifies the device via the Read Terminal ID command and by reading the contents of DTID1 in the device communications area. This byte must remain unaltered while power is on in the Device. DBUF value may be altered after POR, online, or offline.

In addition locations X'80'-'84' must contain the appropriate device information, as outlined below, when the device returns a POR. Note however, that if the power on response is the result of a "Diagnostic Reset" command and the device has set DTID1 bit 1=1 indicating that it is ready to dump, the TCA buffer starting at location X'80' should <u>not</u> be altered with the device information. Locations X'85'-'A3' are optional, however, if these fields are not used, they <u>must</u> be = 0.

This information is read back and stored in the controller. It is intended for enhanced network management and device problem determination purposes. The controller may disconnect any device not supplying the mandatory fields (1 and 2) as discussed below.

The following information is mandatory:

Information	Length	Format
1. Device Type (Device Number, Not Device Emulated) (Start at X'80')	4 Bytes	EBCDIC (Numeric) For Non-IBM Products, this field must be right-justified padded with X'F0' if necessary.
2. Customer Programmable and IBM/Non-IBM Identifiers (Start at X'84')	1 Byte	Bits 0-3: X'1' = Hardware or Microcode X'E' = Customer Programmable Machine Bits 4-7: X'1' = IBM Product X'9' = Non-IBM Product Other Values Reserved

The following information is optional:

Information	Length	Format
3. Model Number (Start at X'85')	3 bytes	EBCDIC AE Chars, right-justified and padded with X'40' chars. X'000000' if unknown or n/a.
4. Plant of Manufacture or Origin (Start at X'88')	2 bytes	EBCDIC Characters designating manufacturing location. X'0000' if unknown or n/a.
5. Serial Number (Start at X'8A')	7 bytes	EBCDIC AE Chars, right-justified and padded with X'F0' chars. X'00..00' if unknown or n/a.
6. Software Release Level (Start at X'91')	3 bytes	EBCDIC AE Chars, right-justified and padded with F0 characters X'000000' if unknown or n/a.
7. Device Specific Information (Start at X'94')	16 bytes	EBCDIC AE Chars, user defined padding and justification. X'00..00' if unknown or n/a.

NOTES: AE Characters are EBCDIC 0-9, A-Z, $, #, @, period, null.
n/a = not applicable.
Device Specific Information may be Release or EC Levels or any other data a product may wish to supply to identify its characteristics.

The CU, after identifying the device, initializes bytes X'50' - X'51' (CUDPORT,CUAT), and bytes X'54' to X'58' (CULTA) prior to issuing any Start Op command. CUAT is not modified subsequently.

4.2 CU FUNCTION REQUEST SYNCHRONIZATION

The CU prepares a function request by writing the following into the device buffer:

1. Synchronous status must be acknowledged (DPSSTAT). Optionally, asynchronous status may be acknowledged.
2. The data address, if any, into CUDP
3. The function request value into CUFRV
4. Toggling the request synchronization flag CUSYN
5. Any associated parameters into CUFRP
6. The data, if any, into CUDATA (optional)

If the above sequence is successful, then, by separate command queue, the request is initiated with a Start Operation.

NOTE:
Command queue retry does not cause the request synchronization (CUSYN), asynchronous acknowledgement (DPASTAT), or synchronous acknowledgement (DPSSTAT) fields to exhibit multiple transitions. Further, the device does not examine these fields until a Start Operation is received.

The Start Op causes the Device to interpret and process the function request. When the processing is complete, the device posts request completion (DPSSTAT),

stores completion code in DSSV with any associated parameters. Then, status available (bit 6) is set on and made available as a POLL response.

Once Start Operation is issued, the CU looks for function completion status. The CU does this by POLL. When the status is available, the CU reads DPSSTAT, DSSV and its parameters, and processes that status.

The Device hardware must reset the status available bit as a POLL response on receipt of POLL ACK.

The CU does not issue a second function request until the function completion status of the first request has been read and acknowledged by writing DPSSTAT to X'00'. The CU may write CUDSER and issue a Start Operation command anytime, but only if the intention is to disconnect the interface by sending out the "last-ditch-command-queue".

4.3 ASYNCHRONOUS EVENT SYNCHRONIZATION

The Device reports an asynchronous event to the CU by placing the event identification in DAEV with any associated parameters, X'01' in DPASTAT and by subsequently setting Status Available (bit 6) in response to a POLL.

Following receipt of an Status Available POLL response, the CU reads DPASTAT, DAEV and DAEP(s) and processes them as appropriate. The CU acknowledges receipt of the status by writing X'00' to DPASTAT and making a (any) function request to the Device.

On receipt of a Start Op while DPASTAT is zero, the Device may report a queued asynchronous event. Only one asynchronous event may be reported and unacknowledged at any time. It is the responsibility of the Device to queue any asynchronous events while waiting for an acknowledgement.

If the CU cannot process an asynchronous event (because, for example, it is processing a synchronous function for that device), the CU queues that event (acknowledged or unacknowledged,) until it can be processed. If the event is unacknowledged, the CU need only remember that an event occurred.

4.4 EXPEDITED STATUS (ES) INTERFACE

ES provides a means to service device requests on an immediate basis independent of other states of the device or its logical terminals. ES communication utilizes unique TCA fields. ES requests from the device are given priority over synchronous status and asynchronous device requests. ES requests may be used to communicate status of intermediate steps or totally asynchronous events during concurrent execution of a function request which has not been completed. The device prepares status to be returned on a bit 6 POLL response. Processing of ES does not allow a function request to be issued. The controller signals the device when processing has completed by issuing a command which causes a TCA interrupt. This command, in this context, is called the "Alternate Start Operation", but is identical in format and performs the same coax function as Read Terminal ID. The Read Terminal ID command is an alternate means of interrupting the device processor.

4.4.1 FLOW

The controller - device sequence for ES is, as follows:

```
DESCRIPTION            CONTROLLER                 DEVICE

Device Request                                    Store ES Request and
                                                  Post 'ACK' Flag
Post Coax Status.
                       POLL------------------->
                       <-----------bit 6 status

Read Status            Read ES TCA Area------->
                       <-------------data------

Process Request Code

        .
        .
        .

Write 'ACK' Field      Write Data------------->

Interrupt Device       Alternate Start Op----->
                       <-----------Data Response

Device Interrupt                 Start Operation------------->

Device Post-Processing
```

Successful receipt of the "Alternate Start Op" ends the ES communication. The device does not examine the TCA ES area until receipt of this command. As with function request synchronization, the Alternate Start Operation command must be issued separately from writing request acknowledgement in EXFAK. Should the device receive multiple TCA interrupts resulting from error recovery, receipt of additional interrupts must be ignored (as determined by the state of the ACK flag in EXFAK). The device must recognize the difference between an initial interrupt and retry.

4.4.2 STATUS PRIORITIZATION

The base status response to POLL can potentially represent device requests to service any or all of the three device status paths: Expedited, Synchronous, and Asynchronous. To process the status response to POLL, the control unit reads EXFAK, DPSSTAT, and DPASTAT. Status is serviced in the order of Expedited, Synchronous, and Asynchronous (high to low priority). Asynchronous status processing may be deferred (remain unacknowledged) as long as the control unit and device are involved in a logical transaction.

4.5 MULTIPLE LOGICAL TERMINAL ROUTING

Not all operations have a Logical Terminal routing requirement. CU file access and local copy, for instance, are considered physical device level operations. Logical Terminal applications are those involving system transmissions. If the function is LT specific, requests from the control unit carry a one byte LT address in CULTAD. WDAT in SNA is an exception. The LT (Logical Unit) address is transferred in the SNA Transmission Header of the message instead of CULTAD. LT specific asynchronous device requests carry the same one byte address in DALTAD. Only the device understands the mapping of device resources to a specific address. When a function is non-LT specific the routing value in CULTAD or DALTAD is loaded with X'FF' (which cannot be configured as a valid LT address). LT routed functions are:

CONTROL UNIT	DEVICE
WDAT	AEEP
PDAT	AEEB
RDAT	AEER (Program Check)
LOCK	AEDV (Online/Offline)
WLCC	
CTCCS	
WCUS (in some cases)	

4.6 RAS CONSIDERATIONS

A portion of device control is involved with verification that the interface, the device, and the CU are functional. These considerations are discussed in the following sections.

If a device or part of the control unit is not functional or becomes not functional, the controller attempts to write a WCUS containing containing the appropriate error code to each TCA device affected. These codes fall into several categories as follows:

2NN Error Codes - 3274 Detected Device or DCA Errors

3NN Error Codes - 3274 Errors Detected by the 3274

4NN Error Codes - Application Program Checks Detected by the 3274

5NN Error Codes - Communication Line or Channel Errors Detected by the 3274

6NN Error Codes - TCA Device Detected Hardware Errors

7NN Error Codes - TCA Device Detected Application Program Checks

More information about these error codes may be found in the documentation referenced in the Introduction (section 1.0).

4.6.1 CU ACTIVE

While the CU is active, it periodically POLLs the device. If the device detects an absence of polling or other coax activity for a sustained period of time, it may assume the CU is inactive. The maximum time between POLLs for an active CU does not normally exceed 1 second. If/when the DCA hangs, POLLs may cease for about 20 seconds.

A CU could be polling, but not otherwise functional. The Device can detect this by periodically presenting Expedited Status. If the ES request is not acknowledged within 10 seconds, the CU is not functional. The Device sets an indicator to display this condition. For performance considerations the device should not present Expedited Status for the purpose of soliciting a CU response more frequently than once every 30 seconds.

4.6.2 DEVICE ACTIVE

If the device is idle and cannot honor any function requests, then it must not answer or acknowledge POLLs. The CU treats such a device as powered off during this period of time, and may notify the host of the condition. The device must exit the powered off state by responding to POLL with the POR response. Due to

performance considerations power off/power on transitions should not occur more frequently than once every 5 seconds.

If the CU issues a function request to the device and the device does not report back with either function completion status or Expedited Status (timer interrupt, see section 8.1) within 1 second, the CU assumes the device is malfunctioning and terminates all further communication with the device (except polling) until a "power on special status" is received. During the disconnect sequence, the 3274 attempts to write a 243 machine check into the field CUDSER to notify the TCA device of what has happened. Expedited Status maintains the active (busy) state while the synchronous status indicates function completion.

4.6.3 PROCESS TIMINGS

There are two types of timeout requirements for 3274/Distributed Function Device communications:

- Host Transaction timing and
- Device Response timing.

Both types of timing are done by the Distributed Function Device and are reported to the 3274 via Expedited Status (ES).

A field in the TCA (EXTIME) indicates which type of timing should be done. The OFF/NORMAL/DEFAULT state of this field is Device Response timing. Host Transaction timing occurs only when the field (flag) is set.

In Device Response timing mode, the device must report ES every 0.375 to 0.75 seconds from the issuance of a Function Request until the device reports synchronous status.

During Host Transaction timing, the device must report ES every 0.375 to 0.75 seconds until the "Host Transaction timing" flag is reset, independent of function requests and device states. The control unit initiates this type of timing on a Function Request by setting the flag in EXTIME. Host Transaction timing ends when the flag is reset. The flag is set/reset anytime by the control unit without initiating a device interrupt.

For non-SNA (both BSC and local channel), a maximum of 9 ES TIMER interrupts may be presented by the device. Host Transaction timing is not defined for SNA attachments.

For Device Response timing, 49 to 63 ES TIMER interrupts are allowed (at the discretion of the control unit).

If the device times out, the interface is disconnected with a 243 machine check and appear powered off to the host. The control unit does not require Device Response timing before the first device ONLINE request (AEDV), i.e., during IML and down stream loading of the device.

The time intervals allowed (in seconds) are as follows:

ES TIMER INTERVAL	NON-SNA BSC and LOCAL CHANNEL 9 Intervals	DEVICE RESPONSE TIMING 49 to 63 Intervals
MIN 0.375	3.375	18.375 to 23.625
MAX 0.75	6.75	36.75 to 47.25

4.6.4 ERROR EVENTS

The Device must report errors as asynchronous events. The frequency of reporting hardware errors must be limited to avoid overrunning the CU. If an error or error sequence is occurring repeatedly, it is not reported on each occurrence unless manual intervention (such as an operator depressing a RESET key), a new host transmission, or a delay of at least 1 second occurs. Sections 6.1 and 7.1 describe the mechanism for reporting such errors.

4.6.5 ERROR EVENT LOGGING

The Device may report each error to the control unit for maintenance statistics when the error occurs (under constraint in preceding paragraph). The CU maintains report summary counters for host program errors, transient hardware errors, and permanent hardware errors regardless of whether the Device is powered on.

Devices are not required to report 630, 632, 633, 635 or 636 errors to the controller. If they are sent by the device, the controller logs them but does not generate an Alert because these codes are generated either as the result of a 3NN code sent to the device from the controller or are considered a non-error condition such as "disk not ready" (see section 10.1).

4.7 COAX ERROR RECOVERY

A transmission error from the CU to the device is detected by the device which must then ignore that coax transmission and inhibit the TT/AR response. When the 3274 does not get the TT/AR it stops immediately and goes into error recovery state.

If the error occurs on the transmission from the device to the CU, TT/AR error or parity error, the CU stops and goes into error recovery state.

4.7.1 RETRY OF NON START OP COMMAND QUEUES

When a string of COAX commands and data fails, it normally causes the 3274 to attempt recovery by retrying the operation. This is possible due to the design of the device adapter and 3274 which prevents any invalid commands or data from being processed or stored into the device's memory.

4.7.2 RETRY OF START OP COMMAND QUEUES

The recovery from a command queue containing a Start Op command is similar to the above but has several implications which must be understood. There are four different error states which are possible as the result of a Start Op command failure.

1. The Start Op command was lost on the transmission from the CU to the device. In this case when the command is reissued by the CU the device processes it normally.

2. The Start Op was received by the device but the TT/AR response was lost on the device to CU transmission. In this case the device is presented with an interrupt and the CU attempts recovery by resending the Start Op command.

a. The device may not yet have started to process the Start Op command. In this case the device must NO-OP the second Start Op and no problems arise.

b. The device has started to process the function request specified by the Start Op but has not completed processing the request. The device-CU hand-shaking protocol allows the device to recognize the second Start Op as a duplicate and ignore the (second) Start Op.

c. The device has completed processing the function request, has posted the completion (synchronous) status but the CU has not yet processed the completion status. Since the synchronous status has not been acknowledged by the CU, the device must also recognize and ignore the second Start Op.

4.7.3 UNRECOVERABLE ERRORS

In the event that the CU receives invalid device status, the CU initiates the device disconnect action of posting an error code (which, if non-zero, is the value of the machine check indicator to be displayed) in CUDSER and issuing a Start Operation command. This is the "last ditch command queue." Retry and synchronization states described below are NOT applicable. The device is only re-connected if it returns a POR response. The device should test CUDSER for non-zero with the receipt of each Start Op command. Certain device errors cause device disconnect without the "last ditch command queue."

4.7.4 DETECTION OF SYNCHRONIZATION ERRORS

It is essential that the CU and the device be able to maintain synchronization over the COAX interface at all times. This interface is controlled via COAX commands and status bytes in the device buffer.

Every time a new function request is made the byte CUSYN must be toggled between X'01' and X'00'. The device must interpret the toggled value of CUSYN when a new function request is made as the acknowledgement of the previous synchronous status.

Command queue retry may cause a Start Op to be issued multiple times, but CUSYN is not changed. This allows the device to ignore the additional Start Op(s) if the first one was actually received.

If CUSYN is toggled but the device is still processing a function request when a second Start Op is issued, the device must report a fatal synchronization error (ERFR) to the CU, set the DSSP to X'04' (Synchronization Error), and the device indicators should be updated to reflect the error. The CU then puts the device into power off state. The device records the synchronization error in the log which may be examined using test mode.

At power-on time the device must set CUSYN to X'00'. The device should expect the first function request from the 3274 to set CUSYN to X'01'. The 3274 maintains the value of CUSYN to avoid reading it from the device buffer prior to each Start Op.

5.0 DEFINED FUNCTION REQUESTS

The function requests that may be issued by the CU to the Device as described in section 4.2 are:

Value	Name	Function
X'01'	CNOP	Cause interrupt on Device
X'02'	WCUS	Write Control Unit Status
X'03'	WDAT	Write Data from Host
X'04'	WDBD	Write Data Base Data
X'05'	RDCOPY	Read block of SCS data for local copy
X'06'	WLCC	Write Local Channel Command
X'07'	LOCK	Non-SNA host selection, device ready request
X'08'	RDAT	Generate inbound data for host
X'09'	WCTL	Write printer characteristics for local copy
X'0A'	PDAT	Prepare read data prior to host notification
X'0B'	CTCCS	Terminate chained command sequence
X'0C'	RDBD	Generate request for data base data
X'0D'	RPID	Read printer assignment

5.1 CNOP

CNOP has no function other than to allow a Start Op to be issued without a specific function being performed. It may be used to acknowledge asynchronous events. CNOP has no parameters.

5.2 WCUS

WCUS is used to report CU state changes and CU events that are detected by the CU and normally communicated to the operator. The 3274 does not manage device indicators or indicator areas. (The device does not communicate indicator reset to the 3274.)

WCUS is classified as either:

1. An anticipated event, contextually valid to a given multi-step sequence, such as 'copy request queued' or 'printer printing'. Multiple events are prioritized in a contextual logic progression. Events are grouped by function.

2. Unscheduled status is reported per external conditions asynchronously, such as the Reminders and LUSTATUS groups. Unscheduled status is allowed to interrupt normal protocol sequences and take priority over a scheduled event. Unscheduled status may be delayed due to processing algorithms in the CU or the device.

   If the device is not "online" to host (AEDV) the following unscheduled status is deferred until the device is "online":

   -Comm Check Reminder
   -LUSTATUS Group
   -Printer Assignment

WCUS function requests for LU/PU status are only sent once per status change, or following the device's being placed ONLINE to the host; i.e. LU active WCUS only issued once from ACTLU or online to DACTLU, ACTLU, DACTPU, ACTPU or offline.

When a read or write file function request completes successfully, a CNOP function request is sent to the device to acknowledge the asynchronous status.

If a read or write file function request completes with error, an appropriate WCUS is sent to the device (see also section 5.10).

The following is a list of the conditions (detected by the CU) by area in priority (high to low) order:

Group Name	Condition Name	Event/ Status	Parm 1	Parm 2-4
Input Inhibit	Machine Check	Event	01	NNN
	Communication Check	"	02	"
	Program Check	"	03	"
Readiness Group	Ready, DSL allowed	Event	10	000000
	Ready, no DSL allowed	"	10	020000
Identity	Device Identification	Status	20	000000
Reminders	Communications Check Reminder	Status	30	NNN
	No Reminder	"	31	000000
	Disk Not Ready (cover open)	"	60	000000
	Ready (cover closed)	"	61	000000
LT Status	LU Active	"	40	ACTLU Parm2: = RU byte 1
	LU not Active	Status	41	000000
Disk Completion	Fatal Hardware Error	Status	70	000000
	Disk Media Error	"	70	040000
	Disk Overrun	"	70	060000
	Disk not Ready	"	70	0A0000
	Wrong Disk	"	70	140000
	File not Found	"	71	020000
	File not Writable	"	71	080000
	File not Readable	"	71	100000
	File Locked (contention)	"	71	0C0000
	File not Locked	"	71	120000
	File Overflow	"	71	0E0000

NOTE:
NNN numbers are the 3270 error codes which are packed decimal and right-justified in bytes 2,3.

NOTE:
Any WCUS condition not recognized by the device must be acknowledged with normal function complete status.

See section 9.2.3 for WCUS values for LOCAL COPY.

5.2.1 WCUS(20) DEVICE IDENTIFICATION

When a device reports AEDV(Online), the controller may issue a WCUS(20). When the Start Op is issued to the device, the controller has placed the information as described below in the TCA buffer beginning at location X'80' to identify itself to the device.

When the device responds Function Complete to this command, it may first place its corresponding information in the TCA buffer as described below, or may simply choose to return the FC without updating the buffer with its data (not support the command).

Upon receipt of the function complete from the device, the controller reads the appropriate area of the TCA buffer. If the device has provided its own information, the controller updates the information received at device POR time (see section 4.1) and then checks that the "device type" and "flag byte" are valid. As noted in section 4.1, other device information is optional.

If the device has failed to provide the required information either as a POR or WCUS(20) response, the controller disconnects the interface with a 240 machine check.

The device information is stored in the controller and is intended for enhanced network management and device problem determination purposes. It is also suggested that the device retain the information provided by the controller in the WCUS(20) command, and make this information available to the device operator upon request.

Included in both the controller request and the device response are two bytes (X'82-83') that must be set by the originator to X'0000'.

Upon receipt of WCUS(20) Start Op at the device, the controller has placed the following information in the device's TCA buffer:

Location	Data	Comment
X'80'	X'--'	Length of Data
X'81'	X'00'	Data Format Identifier
X'82-83'	X'0000'	Reserved. Must be set to zero by the controller and not checked by the device
X'84-87'	FxFxFxFx	Device Type of Controller in EBCDIC.(Numeric)
X'88'	Bits 0-3:	X'1'=Hardware or Microcode X'E'=Customer Programmable Machine
	Bits 4-7:	X'1'=IBM Machine X'9'=Non-IBM Machine Other Values Reserved For This Byte

Some or all of the following information may also be included, depending on the length byte above:

X'89-8B'	X'------'	Model Number in EBCDIC AE (X'000000' if unknown)
X'8C-8D'	X'----'	Plant of Manufacture: EBCDIC per IBM Standard CB-0-2021-000 (X'0000' if unknown)
X'8E-94'	X'.....'	Seven Digit Serial Number in EBCDIC AE right justified and padded with F0 (X'00...00' if unknown)
X'95-99'	X'---'	Release Level of Program in EBCDIC AE, right justified and padded with F0 (X'000' if unknown)
X'98-A7'	X'.....'	Maximum 16 Digits of User Infromation in EBCDIC AE, User Defined Padding and Justification

Note: AE Characters are EBCDIC 0-9, A-Z, $, #, @, period, null.

When a device that does not support the WCUS(20) command returns function complete to the controller, the ID byte (X'81') remains as set by the controller. If the

device chooses to update its POR parameters, the TCA buffer should appear as follows:

Location	Data	Comment
X'80'	X'--'	Length of Data
X'81'	X'01'	Data Format Identifier
X'82-83'	X'----'	Options Supported by Distributed Function Device (Flags)
	Bits 0-B	Reserved
	Bit C	Device supports Device Initiated UNBIND option. May only be set if CUSLVL Option Bit 'C' is supported by controller.
	Bits D-F	Reserved

Note: Reserved bits in this field must be set to zero by the device and not checked by the controller.

X'84-87'	FxFxFxFx	Device Type in EBCDIC (Numeric)
X'88'	Bits 0-3:	X'1'=Hardware or Microcode X'E'=Customer Programmable Machine
	Bits 4-7:	X'1'=IBM Machine X'9'=Non-IBM Machine
	Other	Values Reserved For This Byte

Some or all of the following information may also be included, depending on the length byte above:

NOTE:
Information previously given by the device in the POR response must also be given here, i.e., zero data from WCUS(20) response overrides information given in the POR response.

X'89-8B'	X'------'	Model Number in EBCDIC AE (X'000000' if unknown)
X'8C-8D'	X'----'	Plant of Manufacture: EBCDIC per IBM Standard CB-0-2021-000 (X'0000' if unknown)
X'8E-94'	X'.....'	Seven Digit Serial Number in EBCDIC AE, right justified and padded with F0 (X'00...00' if unknown)
X'95-97'	X'---'	Release Level of Program in EBCDIC AE, right justified and padded with F0 (X'000' if unknown or not applicable)
X'98-A7'	X'.....'	Maximum 16 Digits of User Information in EBCDIC AE, User Defined Padding and Justification

5.2.2 WCUS(42) - RTM CONTROL

The RTM interface and Last Transaction Time display can be enabled or disabled at any time via a WCUS(42) from the CU. If a WCUS(42) is not received from the CU, the RTM interface defaults to the disabled state.

WCUS	Parameter	Value	Description

PARM 1	42	RTM WCUS
PARM 2	00	RTM Disabled for LT in CULTAD
	non-00	RTM Enabled for LT
PARM 3	00	LTT Not Authorized
	01	LTT Authorized
PARM 4	00	Reserved

Refer to section 8.2 for a complete description of the RTM interface.

5.3 WDAT

Write Data is used to pass data received from the host to the Device. The CU allocates a portion of the data area of the Device buffer, places the length, flags, and data in the allocated area, and issues the function request with the starting address of the allocated area specified in CUDP.

CUFRP1 and 2 contain a TCA buffer address if (and only if) Flag bit 6 of the message header is set indicating data wrap. CUFRP 1 and 2 contain the address of the last byte of valid data, the high address, before wrap. The data always wraps to location X'80'. The first 40 bytes of a message is not wrapped.

5.4 WDBD

Write Data Base Data is used to pass data to the Device as retrieved from a file as a result of an asynchronous request for data base access. The CU performs the same actions as for WDAT if the action is successful plus setting CUFRP1 to one byte file identifier requested asynchronously by the device (AEDBA), and setting CUFRP2 (flag byte). CUFRP2 flags are:

Bit 0	= 0	File retrieved from disk.
	= 1	File retrieved from 3274 memory.
Bits 1-7		Reserved-must be zero

Multiple WDBD request may be required to transfer a data file. Device buffer control flags (FOM/LOM) are set accordingly. If the device returns FRA status, the the CU terminates the request if it is not LOM. If the data base access is unsuccessful, the CU issues WCUS (Machine-Check, Data-Base-Error, or Not Ready). A Disk Reminder may or may not have been previously issued. See section 10.1 for specific error conditions.

5.5 WLCC

Write Local Channel Command is used for a non-SNA channel attached 3274 and is used to pass a local channel command to the Device.

The CU places the channel command byte in CUFRP1. CUFRP2 is set to '01' if inbound data was sent to the host.

CUFRP2 is set to '04' if the current command was chained from a previous command. These values are bit significant (both conditions could exist).

Valid commands include:

Erase All Unprotected	6F
Erase/Write	F5
Erase Write Alternate	7E
Read Buffer	F2
Read Modified	F6
Write	F1
Write Structured Field	F3

5.6 LOCK

In Non-SNA attachments, LOCK is used to synchronize device and control unit to a ready state at host selection time. No parameters are required. Normal responses are Function Complete or Function Complete Synchronous Error (Busy or IR). LOCK is not a valid request in SNA attachments.

5.7 RDAT

Read Data is used when the CU is ready to send data from the Device to the host. The CU allocates space in CUDATA, and places the address of the allocated area in CUDP, then issues the function request. The Device places the actual length of the data + flags + length, the flags, and the data in the Device buffer at the location specified by CUDP. CUFRP1 and CUFRP2 (halfword) represent the maximum number of data segments which the controller processes out to the host link. The target length of each segment is passed in CUFRP3/CUFRP4 (halfword). In SNA, one and only one complete RU must be constructed using these segmenting parameters. RU size is derived from these parameters. If the BIND specifies a smaller RU size, then the BIND takes precedence. See Attachment Considerations (section 12.0). The data length may not exceed the target length.

NOTE:
CUFRP1/2 is set for SNA attachment only. In Non-SNA, this parameter is not present and the device must assume a value of 1.

5.8 PDAT

PDAT, or prepare inbound data before notifying host, causes the same function to be performed as RDAT with the same parameter values used. This improves line/channel utilization and controller throughput.

5.9 CTCCS

Terminate Chained Command Sequence is used in non-SNA only to indicate the end of a selection sequence. This corresponds to EOT in BSC and no more command chaining in a CU non-SNA channel attachment.

The following positional flags may be used for CTCCS:

CUFRP2	'00'	Normal termination. Note: This value used for WSF chaining error. WCUS program check is written after CTCCS.
CUFRP2	'01'	Read data accepted by host
CUFRP2	'02'	Sequence terminated due to invalid select command processing by controller.

(Program Check WCUS is written following CTCCS)

CUFRP2 '08' Sequence terminated due to communications check. For BSC, this means that a 501 comm. check has occurred. The device must determine whether the state is to be reset (to state 1) or the request is to be reissued.

For SLHA, this flag tells the device to reset to state 1. This parameter is set when a comm. check has occurred as a result of a Channel Systems Reset (505) having been received from the host. This parameter may also be set if a Selective Reset has been received after the device has been selected (locked.)

5.10 RDBD

In response to asynchronous Data Base Store requests (AEDBS) the CU requests the device to load the file in the device buffer in the same manner as data is requested for RDAT. CUFRP1 contains the one byte file parameter requested by AEDBS. CUFRP3,4 specifies the maximum length of data accepted by the CU. Data is prepared in t e buffer with a 4 byte header (length and flags). The file may require multiple RDBDs to complete (as indicated by segment flags).

Successfully updating the file is indicated only by the acknowledgment of asynchronous status. Request failures are indicated by disk completion status (WCUS).

NOTE:
File access is protected with Read/Write Lock to other devices during the transaction.

If a READ is attempted of a locked file larger than a cache buffer, a "file locked" return code is passed to the requesting device. This is to prevent downstream loading a partially updated file.

If the CU is unable to respond to either AEDBS or AEDBA with a RDBD then the CU issues WCUS. See section 10.1 for details of WCUS parameters.

5.11 RDCOPY

Tells the device to send a block of the local copy data stream (SCS format) in the CUDATA area pointed to by CUDP. The maximum length of the block is specified in CUFRP3 and CUFRP4.

NOTE:
CUFRP1/2 are not used.

5.12 WCTL

Provides the characteristics of the printer for local copy operation in the CUDATA area, including APL support, Extended SCS support for Set Attribute, PS loaded Alias names and flags, and printer switch settings. The CU ensures that selection criteria for the copy device includes the SCS print feature. The format of these characteristics is:

BYTE NO.	DATA
0,1	Length of data beginning at CUDATA
2,3	Flags. Bit 0=1 First of Message Bit 1=1 Last of Message Bit 2=0 PCIA Data Format (Base) =1 FMH + Query Reply SF Format (Extended) Bits 3-15 Reserved (must be zero)
4-17	Device Characteristics Byte 4: Reserved (must be zero) Byte 5 = Printer ID byte X'01' Byte 6-17 = Printer ID bytes X'04' thru X'0F'
18-29	Alias Table - dynamic state of PS RAMs, if present. Each entry consists of 1 byte Alias and 1 byte PS variable flags corresponding to byte 8 of LPS Structured Field. (If the "alias" = X'FF' the flags are ignored).

NOTE:
CUFRP1-4 are not used.

In order to use Graphic Escape in the data stream the printer must indicate support of APL in its terminal ID.

In order to use a Set Attribute:

a. Highlight - The printer must indicate support of underline highlighting in its terminal ID. The controller "corrects" the terminal ID for printer type 0001.

b. Color - The printer must indicate support of color in its terminal ID.

c. Character set - The printer must indicate the presence of the PS feature, and both the device and the printer must have matching aliases and CB bit equal to "compare."

5.13 RPID

RPID requests the device to respond with:

a. printer port address, class number, or assignment request in DSSP or

b. FRA to terminate the sequence. See section 9.4.

6.0 DEFINED SYNCHRONOUS STATUS

When the Device has completed processing of a function request, it posts DPSSTAT, places completion status in DSSV and any associated parameter in DSSP. If the request involved generation of data, the data is placed in the indicated portion of the buffer with data length and flags. The device then sets Status Available as the POLL response. The following values are defined for DSSV:

Name	Value	Description
FCSE	X'02'	Function Complete with Synchronous Error
FC	X'04'	Function Complete
FCIR	X'06'	Function Complete with Input Required
ERFR	X'08'	Error in Function Request
FRA	X'0A'	Function Request Aborted

Undefined values cause interface disconnect.

6.1 FCSE

Function Complete with Synchronous Error indicates request processing was terminated due to one of the following conditions placed in DSSP. The controller also logs an 'NN' value in the Device Control Block as indicated by DSSP2 (packed decimal):

DSSP:	DSSP2:	Reason:	DSSP3:
X'01'	n/a	Device Busy	
X'02'	6NN	Device Error	XXXX 0000
X'03'	7NN	Command Reject	XXXX 0000
X'04'	n/a	Intervention Required (security key off or local copy busy)	
X'05'	6NN	Data Check	XXXX 0000
X'06'	7NN	Op Check	XXXX 0000

This status is not used in SNA because the Device (rather than the CU) is responsible for reporting synchronous errors to the host. Asynchronous status is employed.

The device displays internally detected machine and program checks as 6NN and 7NN numbers respectively. The numbers 601-699 and 701-799 have been reserved for Distributed Function Device use. In addition, the four high order bits of the next byte (DSSP3) are used to bump four 3274 device RAS counters. These counters can be displayed by a "/1" test in TEST mode. The interpretation of these counters is device dependent and is described in the device documentation referenced in the Introduction (section 1.0). The four low order bits of this byte must be zero.

6.2 FC

Function Complete is used to report normal function completion.

6.3 FCIR - FUNCTION COMPLETE INPUT REQUIRED

FCIR is used to request an RDAT by the CU. The device is prepared to accept an RDAT function request.

In non-SNA attachments, it is used to report normal completion of a WDAT or WLCC function request which contained a READ command in the data. If the WLCC command is a Read then the parameter, DSSP, indicates whether an RDAT request is required, or data is already available in the buffer.

```
DSSP =  X'00' :RDAT must be issued to prepare data
     =  X'01' :Data is available in TCA buffer
```

In SNA, this status may only be used to signal the CU that a response consisting of a single segment RU is pending. The CU issues an RDAT to read it before issuing any other function request.

6.4 ERFR

Error in Function Request indicates the interface is broken. The 3274 logs a 241 machine check error code. The CU then disconnects the device (see section 4.7.3).

The device must place a value in DSSP as follows:

```
X'00'          Unsupported CUAT
X'01'          Unsupported CUFRV
X'02'          Unsupported CUFRP
X'03'          Unsupported CUDP
X'04'          Synchronization (CUSYN) Error
X'05' - X'0F'   Reserved
X'10' - X'FF' Device specific errors
```

6.5 FRA - FUNCTION REQUEST ABORTED

This completion code is used by the device as a mechanism to cope with contention situations. Generally, it is a means of cancelling an asynchronous request because another event of greater significance has occurred. For example, the device generates an AEEP request but responds FRA to the PDAT or RDAT to service it because the operator pressed the RESET key. See table 6.6.

6.6 TABLE SUMMARY

Summary of request codes and their applicable parameters:

F.R.	CODE	CUFRP1	CUFRP2	CUFRP3	CUFRP4	CUDP
CNOP	01	—	—	—	—	—
WCUS	02	PP	PP	PP	PP	—
WDAT	03	<—WA——>		—	—	DA
WDBD	04	FN	FF	—	—	DA
RDCOPY	05	—	—	<——LS——>		DA
WLCC	06	CC	FF	—	—	—
LOCK	07	—	—	—	—	—
RDAT	08	<#segments->		<— LS ——>		DA
WCTL	09	—	—	—	—	DA
PDAT	0A	<#segments->		<— LS ——>		DA
CTCCS	0B	—	FF	—	—	—
RDBD	0C	FN	—	<— LS ——>		DA
RPID	0D	—	—	—	—	—

Where:

— = does not apply or is not used by 3274, ignore parameter - contents may be unpredictable
PP = parameter data for WCUS.
DA = data buffer address.
CC = channel command.
FF = function dependent flags.
LS = maximum length of segment.
FN = Data Base item name - 1 byte identifier.
WA = wrap address (SNA only)

Correlation of request codes and synchronous status completion:

F.R.	FC	FCSE: BSY	DE	CR	IR	DC	OC	FCIR	ERFR	FRA
CNOP @	N	—	ID	—	—	—	—	—	ID	—
WCUS @	N	—	ID	—	—	—	—	—	ID	—
WDAT	N	—	ID	UP	—	UP	UP	UP	ID	—
WDBD @	N	—	ID	—	—	—	—	—	ID	UP
RDCOPY@	N	—	ID	—	—	—	—	—	ID	—
WLCC *	N	—	ID	UP	—	—	—	UP	ID	—
LOCK *	N	UP	ID	—	UP	—	—	—	ID	—
RDAT	N	—	ID	—	—	UP	—	—	ID	UP
WCTL @	N	—	ID	—	—	—	—	—	ID	UP
PDAT	N	—	ID	—	—	UP	—	—	ID	UP
CTCCS *	N	—	ID	—	—	—	—	—	ID	—
RDBD @	N	—	ID	—	—	—	—	—	ID	UP
RPID @	N	—	ID	—	—	—	—	—	ID	UP

Where:

* = Non-SNA only commands/response
N = Normal completion
UP = Valid response to be processed
ID = Valid response, disconnects interface - device broken
— = Invalid response, disconnects interface
@ = Only these request codes may be used when the device is in initialized state but not online to host

7.0 DEFINED ASYNCHRONOUS STATUS EVENT VALUES

Asynchronous Events are initiated by the Device and detected at the CU via a POLL response with status available set. The Device must have already placed a value in DAEV and any associated parameter in DAEP, DAEP2, DAEP3.

The following values are defined for DAEV:

Name	Value	Description
AEER	X'20'	Asynchronous Error
AEEP	X'22'	Inbound Event Pending
AEDBA	X'24'	Data Base Access Needed
AEEB	X'26'	End IR
AEDV	X'28'	Device-CU Local Status
AEFREE	X'2A'	Release Printer
AEPID	X'2C'	Request Printer Assignment
AECOPY	X'2E'	Copy Request
AECAN	X'30'	Cancel Copy Request
AEDBS	X'32'	Request Data Base Store

7.1 AEER

The device displays internally detected machine and program checks as 6NN and 7NN numbers, respectively. The numbers 601-699 and 701-799 have been reserved for Distributed Function Device use. These must be reported to the CU as AEER status. In addition, the four high order bits of the next byte are used bump four 3274 device RAS counters. Bits 4 through 7 of this byte must be zero. Any non-zero value in DAEP4 is stored by the control unit as the most recent error qualifier for potential FRU isolation. The RAS counters and the Error Qualifier may be displayed by a "/1" test in TEST Mode.

In addition, type 04 status indicates a "log only" function (no error code or qualifier).

When the device is attached to an SNA controller, AEER status is used to generate Alerts for errors detected by the device. Alert is a C&SM function that flows on the SSCP-PU session. AEER status generates an NMVT formatted record for 6NN and 7NN errors only. Alert requires unique controller and host support. The Logical Unit (LT) address must be identified for program checks since they are PLU-SLU session related. Therefore, DALTAD must contain a valid Logical Unit address for 7NN errors.

AEER	TYPE	1	2	3
Temp Error	01	6NN	XXXX 0000	Qualifier
Perm Error	02	6NN	XXXX 0000	"
Prog Check	03	7NN	XXXX 0000	"
Log Only	04	-	XXXX 0000	-

NOTES:

1. XXXX is used to bump the four (bit significant) counters.
2. Permanent Device Error - interface disconnected.
3. The Qualifier is stored by the CU if non-zero.

7.2 AEEP

Inbound Event Pending is raised when a message is to be sent to the host. This may be used for actions such as an operator pressing the ENTER key.

For SNA AEEP is also used to generate data from an outbound READ command.

7.3 AEDBA (3290 ONLY)

Data Base Access Needed is used to load the 3290. There are parameters. The CU is expected to access the IML (Initial Microcode Load) data base performing the request as indicated by the parameters to AEDBA.

Value	Description
DAEP:	Subsequent WDBD results in the data transfer of the named data entity (Non-zero value used to identify Data Base file).

DAEP2, a flag byte parameter, informs the control unit of the type of transaction to be performed:

Bit 0	= 0	Read with requirement to Write (update) file on disk. Once initiated, the transaction must be completed. The CU puts a Read/Write Lock on the disk file to other devices until the transaction is completed.
	= 1	Normal access (Read only) transaction
Bits 1-7		Reserved; must be zero.

7.4 AEDBS

Data base store is requested. The one byte parameter (DAEP) specifies the file name to be modified. This asynchronous status is followed by a RDBD request from the CU.

7.5 AEEB (NON-SNA ONLY)

A value of X'01' is used to report END IR if IR had been previously reported on an FCSE completion. This request is considered global (multi-Logical Terminals). The device is responsible for determining which LTs owe the host Device End status. DAEP2 must contain a bit map representing the online LTs which owe the host a Device End based on FCSE:IR being returned to LOCK for the respective LTs. All zeros in DAEP2 is invalid.

7.6 AEDV

This status is sent to the CU to determine control unit status and control the device's availability to the host. Parameter values indicate the functions to be performed:

DAEP	DAEP2	DAEP3	DESCRIPTION
'01'	LT Bit Map	'00'	Put device LTs online to the host (If an LT is already online to the host, this request is invalid for that LT. DAEP2=00 is invalid.)
'02'	'00'	'00'	Takes all devices offline from the host. The devices are returned to an initialized state and may use its local resources, but may not communicate with the host. (If a device is already offline, this request is invalid for that device. All LTs are taken offline only if they are online. DAEP2=00 is invalid if all LTs are already offline.
'03'	'00'	'00'	Used for service in conjunction with the Diagnostic Reset command. DUMP Complete.

7.7 AEFREE

See section 9.0.

8.0 EXPEDITED STATUS (ES) REQUESTS

The expedited status communication area of the TCA contains several contiguous fields. Expedited Status (ES) provides an accurate device timer interval interrupt to the control unit measuring elapsed time for a Start Operation (Function Request) in process. ES requests also service the RTM function.

Expedited Status is initiated by the device. The control unit responds to ES by writing the ES response parameters into the appropriate fields and setting the "ES present" flag (EXFAK) to ACK. The CU's acknowledgment is completed by a Read Terminal ID command. The CU must not issue a Function Request as part of the response to ES.

ES request codes are:

- 02 - device busy timer interval
- 04 - start RTM timer
- 06 - stop RTM timer
- all others reserved

8.1 DEVICE INTERVAL TIMING (X'02')

All function requests are timed by the controller for hung device detection. If the controller does not receive ES status from the device within .75 seconds, the device may be disconnected with a machine check error code. The device must report busy status or synchronous completion status within this interval. See process timings in section 4.6.3. Device timing begins at receipt of the function request. EXLTA is set to X'FF' to indicate a physical device level function.

In the idle state the device may use this request as a no-op to detect whether or not the control unit is active.

The control unit uses this request to set and reset "host timing mode" and TCA field EXTIME, (see section 4.6.3).

8.2 RESPONSE TIME MONITOR OVERVIEW

The 3274 Response Time Monitor is a mechanism whereby end-to-end user response time can be measured depending on a definition dictated by the controller customizing process or, in certain cases, an application in the host. Response times for each logical terminal (LT) are measured and maintained in the controller. However, since each device processes its own data stream, it must also implement some of the RTM function.

Response time is measured on an LT basis. Default parameters are established during the controller customizing process for all LTs and may be updated through the host interface for each LT. Upon receipt of an AEDV(01) ONLINE request (assuming the controller supports RTM) the CU sends a WCUS(42) for each active LT. This WCUS notifies the device whether the RTM interface is enabled or disabled (see section 5.2.2). WCUS(42) also notifies the device whether or not the operator is authorized to view the last transaction time indicator. For those devices that support the controller RAS tests, this authority also pertains to the operator's ability to view the RTM logs for all devices on the cluster.

Response time is defined as the time interval from the beginning of an Attention ID (AID) host operation (e.g., ENTER key) to receipt of a resulting data stream (processed by the device) that satisfies the RTM STOP definition. However, an abnormal condition could cause an ABORT of the RTM transaction timing. (An accurate description of the RTM STOP definitions and ABORT conditions may be found in the RTM Final Functional Specification.) The CU (or the host) provides

the device with the appropriate RTM definition, the device notifies the CU when to START and STOP (or ABORT) the RTM transaction timing, and the CU passes the resulting time interval back to the device when appropriate.

NOTE:
The fields carrying the LT addresses (EXFLT for ES and CULTAD for a WCUS) must be valid.

The RTM function can be enabled or disabled (on an LT basis) by the WCUS(42). The device may initiate an RTM transaction via the START RTM Expedited Status command ONLY when the RTM interface is enabled for that LT. The proper (current) RTM definition is supplied by the CU in response to START RTM ES, unless the CU's response is "RTM Disabled." This RTM definition remains in effect throughout the RTM transaction. The RTM transaction ends when:

a. The appropriate STOP condition is met (detected by the device),

b. An ABORT condition is detected and reported by the device,

c. An ABORT condition is detected and reported by the CU,

d. The LT goes offline,

e. The device interface is disconnected (e.g., the device powers off)

Additionally, an RTM parameter supplied with both the WCUS(42) and the START RTM ES response, specifies whether or not the device is authorized to display the "Last Transaction Time" (LTT) indicator. The device must react to the most recent LTT display authorization regardless of whether it flows on the WCUS(42) or on the START RTM ES response.

NOTE:
This is permission to view the indicator only. The operator must provide a specific request to subsequently view the indicator.

The device must not issue either an RTM Start Timer or Stop Timer request if the controller has not sent a WCUS(42) indicating that RTM is supported on the controller. This is interpreted by the controller as an interface error and results in the device being disconnected from the interface.

8.2.1 START TIMER FOR RESPONSE TIME MONITOR (X'04')

The Start Timer request is associated with initiating an inbound host event (AID). When an online LT begins an inbound operation (e.g.,pressing an ENTER key) and that LT has RTM enabled as indicated by the controller, the device should indicate this "start" condition to the controller via expedited status:

```
EXFLT = LT address
EXFRQ = 04 - Start timer
```

The controller acknowledges the expedited status with the following parameters:

```
EXFD1 = RTM definition
        00 - RTM Interface Disabled for this LT
        01 - Time to First Character
        02 - Time to Keyboard Available
        03 - Time to Change Direction/End Bracket (valid for
             SNA attachment only)

EXFD2 = Local display of LTT Authorized
        00 - Last Transaction Time NOT Authorized
        01 - Last Transaction Time Authorized
```

The device may issue START RTM ES only when the RTM interface is enabled. The device must also remember the most recent state of the RTM interface for each (online) LT.

The RTM transaction is considered active if the CU responds to START RTM ES with a non-zero RTM definition. Conversely, the RTM transaction is considered not active or not started if the CU's response is "RTM disabled."

The device must not issue a START RTM ES for an LT with an outstanding active RTM transaction. This is a violation of the RTM interface.

If the CU responds "RTM disabled" to the START RTM ES, the device must not issue either START RTM or STOP RTM ES until notified that RTM is enabled via WCUS(42) (see section 5.2.2).

If the RTM hardware becomes non-functional, the controller reports this to the device via a WCUS(01) 382 machine check. The device should not issue a START RTM for any LT. If a START RTM is received, the LT is notified that RTM is disabled. A subsequent START RTM from that same LT (with RTM disabled) violates this interface and may cause the CU to disconnect.

8.2.2 STOP TIMER FOR RESPONSE TIME MONITOR (X'06')

When the device detects that an appropriate RTM STOP or ABORT condition for an active RTM transaction has occurred, it notifies the CU with the STOP RTM ES:

```
EXFLT = LT address
EXFRQ = 06 - Stop timer
EXFP1 = 00 - Stop timer for this LT and record time.
        01 - Abort timer for this LT. Do not record
             the time in the RTM log.
```

The controller acknowledges the expedited status as follows:

```
EXFD1,2 = Last transaction time in 25 millisecond
          increments (16 bit unsigned value)
EXFD3   = 00 - Above time valid
          01 - Above time invalid (for example, Abort
               or RTM hardware problem in controller)
```

The RTM transaction ends (or becomes non-active) with the CU's acknowledgement of the STOP RTM ES.

The device must end each active RTM transaction with STOP RTM ES (for the proper LT) unless the LT goes offline or the device is powered off. In this case, the controller will terminate the outstanding RTM transaction(s).

The device is responsible for detecting and reporting all ABORT conditions to the controller, except when the LT goes offline or the RTM hardware becomes non-functional.

The device must not report STOP RTM ES unless an RTM transaction is active for that LT. The device must only return one STOP RTM for each START RTM and must not issue multiple STARTS or STOPS for a given LT. Any violation of this rule will cause the controller to disconnect the interface.

The device is responsible for "rounding" the parameters passed to it in EXFD1,2 (Last Transaction Time) to the nearest one-tenth of a second when displaying the last transaction time indicator. This can be easily accomplished by adding X'0002' (50 msec) to EXFD1,2 and shifting right two places.

<u>NOTE:</u>
The last transaction time is included in the controller response regardless of whether or not the operator is authorized to view it.

If the device is notified that the RTM hardware is non-functional (via WCUS(01) 382 machine check), the RTM transaction is considered aborted by the CU. If the device issues a STOP RTM ES, the CU responds with an invalid Last Transaction Time (denoted by EXFD3 = 01).

Examples of the RTM flow follow.

```
                3274                          DISTRIBUTED FUNCTION
                                                     DISPLAY
                  <------------------------------------------- ONLINE
              ACK ------------------------------------------->
                                  .
                                  .
                                  .
         WCUS(42) ------------------------------------------->
           RTM enabled/LTT display enabled
                  <------------------------------------------- ACK
                                  .
                                  .
                                  .
                  <------------------------------------------- START RTM
RTM Definition/LTT
display enabled   ------------------------------------------->
                                  .
                                  .
                                  .
         WCUS(42) ------------------------------------------->
           RTM enabled/LTT display disabled
                  <------------------------------------------- ACK
                                  .
                                  .
                                  .
                  <------------------------------------------- STOP
 LAST TRANSACTION
  TIME (time valid)------------------------------------------->
```

Note that the last transaction time is not displayed by the device in this case, since the last notification was that LTT display was disabled.

The following example assumes that the RTM interface is initially enabled:

```
                3274                        DISTRIBUTED FUNCTION
                                                  DISPLAY
                  <------------------------------------------ START RTM
RTM Definition/LTT
display enabled   ------------------------------------------>
                            .
                            .
                            .
        WCUS(42)  ------------------------------------------>
    RTM disabled
                  <------------------------------------------ ACK
                            .
                            .
                            .
                  <------------------------------------------ STOP or ABORT
  LTT not valid   ------------------------------------------>
```

The asynchronous nature of the interface is demonstrated above. The ES flows on a higher priority than the WCUS. Therefore, the CU must respond to either the STOP or ABORT after the RTM interface is disabled in this manner.

9.0 LOCAL COPY

This section describes the interface to support a device local copy capability, including printer assignment, host (SNA ONLY) or operator (PCM/SNA) initiated copy requests, data stream transfer, control unit events and status significant to the device, and a set of useable protocols.

9.1 DATA STREAM INTERFACE TO SUBSYSTEM PRINTERS

The device must generate a pass-thru SCS data stream using a set of device controls including NL, CR, FF, SHF, SVF, SLD, HT, VT, Set Attribute (if the printer supports extended data stream functions), and Graphic Escape (if the printer has APL ROS installed). The printer selected must support SCS. Device characteristics of the printer are given to the device for evaluation. The device must send only valid SCS characters and control codes to the printer. If the screen or partition copy exceeds the physical size of the print buffer then multiple data transfers with intervening print operations must be performed in order to accommodate the large presentation space. The printer remains allocated to the device until the copy has been completed. SCS chaining is used to emulate a logical unit of work. To the printer the request must appear as a host SCS print chain.

Function Management Headers are not supported on this interface.

For Multiple Logical Unit considerations the device port is allowed only a single queued local copy request and the control unit processes the copy requests serially.

Attentions received from the printer while performing the device Local Copy is stacked by the CU if the printer is bound LU1, and discarded by the CU if the printer is not bound LU1.

The device may split orders (such as SA, GE) across chained data. The controller sets First-In-Segment-First-In-Chain (FSFIC) on first-of-message data and Last-In-Segment-Last-In-Chain (LISLIC) on last-of-message data sent to the printer.

9.2 LOCAL COPY FUNCTION REQUESTS AND STATUS

9.2.1 RDCOPY

See description in section 5.11.

9.2.2 WCTL

See description in section 5.12.

9.2.3 WCUS

Write Control Unit Status (WCUS, value X'02') is the vehicle used to shuttle status to the device resulting from CU state changes and and other events. These are reported directly to the device and indirectly to the device operator. Status is on a priority basis. Separate WCUS values exist for Local Copy:

PARAM 1	PARAM 2	PARAM 3	STATUS
51			Request Queued
52			Long Term Busy
53	nn		Printer Busted
		01	IR
		02	Equipment Check
		03	Data Check (including data stream errors)
54	nn		Invalid Printer Number
55	nn		Assignment Not Allowed
56	nn		Printer Assigned
57			Printer Available
58	nn		Printing Started
59			Request Dequeued
5A			Local Copy Not Configured
5B	nn		Print Complete
5C	nn		Printer Operational (IR Clr'd)

Where nn = xx Printer port address or class number
'FE' Printer selection possible following matrix change
'FF' No assignment

9.2.4 AEPID

AEPID (value X'2C') is the device's request for printer assignment (see section 9.4). Out of sequence AEPID is responded to with WCUS(5A) if local copy not configured.

9.2.5 RPID

RPID (value X'0D') is CU reply to AEPID, and requests the device to respond with either printer port address or class number (see section 9.4).

9.2.6 AECAN

AECAN (value X'30') is the device's request to cancel the queued copy request (see section 9.8). Out of sequence AECAN is responded to with WCUS(5A) if local copy not configured.

9.2.7 AEFREE (SNA ONLY)

AEFREE (value X'2A') is a device request for printer release from Local Copy Hold. This is allowed only when a printer previously held for Local Copy is to be released without requesting AECOPY (see section 9.8).

Printer Release (without AEFREE request) is forced by ACTLU, DACTLU, and UNBIND if the printer is in Hold state.

For MDF implementation, AEFREE is sent only when all LUs are finished with the copy hold.

Out of sequence AEFREE is responded to with WCUS(5A) if local copy not configured.

9.2.8 AECOPY

AECOPY (value X'2E') is a device request for Local Copy. The copy request is classified by the value of DAEP as to operator initiated and host initiated requests:

00 = operator initiated
01 = host initiated (see section 9.5)

Out of sequence AECOPY is responded to with WCUS(5A) if local copy not configured.

9.3 ONLINE

At the device online time, the 3274 issues a WCUS with the default printer assignment if the 3274 has been configured for local copy and the Print Authorization Matrix (PAM) allows local copy from this port.

9.4 PRINT ID SEQUENCE

The device may request printer assignment by sending an AEPID asynchronous status. The CU responds with an RPID, to which the device must return Function Complete (FC), printer port address or class number, or 'what printer have you got?' (X'FE'). These digits are checked for numeric validity before they are passed to the 3274. The 3274 responds with either a Printer Assigned (including X'FF' - no assignment), Invalid Printer number, or Assignment Not Allowed WCUS function request.

FRA response to the RPID is also valid and terminates the sequence. This occurs when an outbound message (WDAT), WCUS(56) due to a PAM matrix change, or a lock request is received from the 3274 at the same time an AEPID request is sent by the device. An AEPID received during a copy that is currently printing causes the interface to the 3274 to be disconnected. The device must not to send AEPID during buffer transfer.

9.5 COPY SEQUENCE

The device must check that the pressing of the PRINT key or the host requested write is allowed, i.e., the current host state allows the print request, and a copy request is not currently queued. (AECOPY while queued causes the interface to be disconnected.)

The device may send asynchronous status, AECOPY, to the 3274. It is not necessary to send the printer or class number as the 3274 already has this information. The CU responds (WCUS):

Request Queued (51) (may be sent twice)
or Local Copy not configured (5A)

Following a WCUS(51), if the printer is not available, the 3274 returns additional status via WCUS indicating one of the following:

Long Term Busy (52)
Printer Broken (53)
IR (01)
EC (02)
Assignment not allowed (55)

When the printer becomes available (or immediately if the printer is free), the function request WCTL is issued. The 3274 has placed the printer characteristics in the TCA buffer at the address specified by CUDP. The actual format of this data has been defined previously (see section 5.12). If the characteristics are acceptable, synchronous status of FC is returned.

If the characteristics are unacceptable, synchronous status of ERFR (if no SCS) or FRA (otherwise, such as DEV CAN / WCTL race condition) is returned and the COPY request is terminated.

If the device returned status of FC, the 3274 issues the function request RDCOPY. The device must place the copy data in the TCA buffer at the address specified by CUDP. CUFRP 3/4 must contain the target length for the data.

The entire buffer transfer may be accomplished with a series of RDCOPY function requests. Printing starts after one full buffer of data is loaded into the printer. The smaller of the TCA and printer buffer is used to determine the actual amount of data that is printed at one time. The device is notified via WCUS(58) that printing has started. Following the entire transfer, the controller writes WCUS(59), request dequeued.

The print phase of the copy sequence is terminated by the 3274 returning one of the following via WCUS:

Printer broken (53) (failure during printing)
Print complete (5B) (good completion)

9.5.1 SECOND REQUEST PROCESSING

During the final segment print of the copy data, after the previous request has been dequeued, a second request may be queued. The copy sequence operates as described above, unless the printer fails while printing the last segment of the previous request. 'Second Request Abort' processing is defined to be the rejection of a subsequent copy request when the printer fails on the first. The controller does not send any additional status to the device when this situation occurs. The request is simply dequeued. A race condition occurs when WCUS(53) for Printer Busted is sent at the same time that the second copy request (AECOPY) is issued. The controller services the copy request normally up to the point WCTL is issued. At this point, the device sends FRA to the WCTL, thus ending the sequence.

9.6 QUERY

In order to provide local copy to advanced printers with variable pitch and potentially other functions which can affect the print format, the Distributed Function Device is able to send Query requests to and receive Query replies from the assigned printer before generating a local copy data stream. New printers indicate support of architecture for Query List in their "Terminal ID" (PCIA). This information is conveyed to the display by the "Extended" WCTL request. Consistent with the local copy interface, the display must not pass any data stream to the printer which would produce error status or unknown results.

A local Save/Restore function is performed by the control unit to an advanced printer which supports Save/Restore Structured Field architecture. This operation is transparent to the Distributed Function Display, except to point out that the display should <u>not</u> assume responsibility for initiating the function. This capability allows the display to change pitch and MPP without adversely affecting output formats set by a host application sharing the printer. The control unit initiates a Save SF to the printer at the beginning of the Distributed Function Display local copy transaction and later initiates a Restore SF at the conclusion of the transaction.

In order to allow Query information to flow between the display and the printer, Function Management Headers and Structured Fields are supported as pass thru data over the local copy interface.

The local copy protocols are unchanged with printers which do not support Function Management Headers or in instances when the use of Query is unnecessary, i.e., Distributed Function Device local copy to base printers (vs. advanced printers). The existing interface continues to be supported without modification.

Initial printer characteristics are supplied to the Distributed Function Display via a WCTL function request. This includes Printer ID (PCIA) and the current Alias Table. If the display requires dynamic format information from the printer the "extended" WCTL function is requested. The "extended" WCTL contains Query Reply data read directly from the printer, whereas, a "base" WCTL conveys the information present in the printer PCIA area. The Query function is handled as a conversational element of the Load Transfer phase, initiated by the first RDCOPY (Read Copy) request and completed in an extended WCTL request. The Load/Print Phase of local copy is initiated subsequent to the RDCOPY Query/ extended WCTL reply.

NOTE:
Local copy always results in multiple transfers. WCUS (58) and WCUS (59) status changes are not issued during RDCOPY Query/WCTL processing.

RDCOPY Query:

The data stream constructed by the device must conform to architecture commencing with a Function Management Header (FMH). The message must not exceed 256 bytes and set FOM/LOM flags in Device Buffer Control Flags (see section 3.8.2). Two new flags are defined:

FMH present (local copy only)

Query Reply expected (local copy only)

See section 5.12 for the WCTL bit 2 flag definitions.

The controller accepts a valid Query request on the first RDCOPY request with buffer control flags having the value X'C0C0' followed by the FMH/SF data.

The controller transfers the query message to the printer and issues a start print. Upon completion, the controller transfers the Query Reply data from the printer to the display and issues an extended WCTL function request. The maximum amount of data transferred is 512 bytes. The WCTL flag byte is set to X'E000' (FOM/LOM/QR). The display responds with Function Complete.

The controller issues another RDCOPY to receive printable data. Normal copy sequences are resumed.

DIAGRAM: (Note: Detail of device responses omitted)

```
PRINTER                 3274                    DISTRIBUTED FUNCTION
                                                        DEVICE

Printer assignment_____________________________________________

                        <--------------------Prt Rqst AECOPY
                        WCUS(51) (request queued)---------->
                        .
     (printer available) .
                        .
                        Base WCTL------------------------------>
<----------------Save SF
                        .
                        .
                        .
Load transfer__________________________________________________

                        RDCOPY---------------------------------->
                        <-------------Copy Data:Query(FMHP/QREX)
<--------------------Query
                        .
          (Query reply) .
                        .
                        Extended WCTL---------------------------->
                        RDCOPY---------------------------------->
                        <--------------------------copy data

Print__________________________________________________________
       (printing started)
                        WCUS (58)------------------------------>
                        .
                 (normal sequence)
                        .
                        .
                        WCUS (completion status)---------------->
<-----------Restore SF
```

9.7 DEVICE CANCEL SEQUENCE

To cancel an operator initiated copy request that is queued, the device sends asynchronous status, AECAN, to the 3274. See section 9.10.5.

The device may also cancel the copy sequence by returning FRA response to the WCTL. See section 9.10.6 for cancel race description.

To cancel the copy sequence after the buffer transfer has started but before the buffer transfer is complete, the device should respond to a RDCOPY request with FC, NL and the Last Of Message flag set to terminate the print. This is necessary as a full partition copy could potentially require a long time to print.

AECAN must not be sent during a host initiated copy sequence. If sent, the controller disconnects the interface.

9.8 PRINTER HOLD (SNA ONLY)

If a host copy request had been rejected recently (due to local copy), then when a printer which may be allocated to the device becomes available, the 3274 holds

the printer (reject a new begin bracket). If the printer is still needed for host copy, a copy request (AECOPY (01)) is returned. If the device doesn't need the printer for host copy no more, it must send asynchronous status AEFREE.

```
PRINTER                 3274                DISTRIBUTED FUNCTION
                                                  DEVICE
                            <---------------- AECOPY(01)PRT RQST
                            ----------------> WCUS (51)
                            .
                            .
                            ----------------> WCUS (52)
                            .                   (or WCUS (53,01)
                            .
(Printer becomes available)
                            ----------------> WCUS (57) PRT AVAIL

                If printer is still needed by device:

                            <---------------- AECOPY(01)PRT RQST
                            ----------------> WCUS (51)
                            .
                            <---------------- AEFREE
                            .
                            .

                If printer is no longer needed by device:

                            <---------------- AEFREE
```

If the printer which is being held has a component failure, the device is notified with WCUS(57) (Printer available). The device must notify the host, allowing a retransmission of the copy request. When the device retries the copy, it receives printer broken status.

9.9 PRINTER CLEANUP

If the 'from' device powers down during multiple data transfers, printer cleanup of new line (NL) is required. When the 'from' device fails, NL (with end of message (EOM) set) is sent to the printer to prevent a subsequent print from overprinting the last printed line. If the printer fails with a temporary error such as data check, parameter error, or IR (non-power off), the NL sequence is sent to preserve subsequent print integrity.

9.10 USEABLE COPY PROTOCOLS

A set of diagrams depicting the flow of certain Local Copy sequences for the device are shown below. These are examples and are not intended to be all-inclusive.

9.10.1 PRINT ID

To change printers authorized for copy, the device operator presses the PRINT ID key. The device is responsible for tracking its own current printer assignment state by making the proper request. This is how it is processed by the 3274:

```
3274                                DISTRIBUTED FUNCTION
                                          DEVICE

   <------------------------------- PRINT ID AEPID
   ------------------------------->  RPID (RQST PTR ADDR)
   <------------------------------- FUNC COMPT (mm)**
   ------------------------------->  WCUS (COMPLT STAT) nn*
```

COMPLETION STATUS	CU STATUS
1. Good Completion	WCUS(56)
2. Invalid Printer ID	WCUS(54)
3. Unauthorized Printer ID	WCUS(55)

*List Of nn Status

```
nn =  xx   Printer port address or print class number
   = 'FF'  No assignment
```

**List Of mm Status

```
mm =  xx   Printer port address or print class number
   = 'FE'  Matrix changed but valid assignment possible
```

9.10.1.1 PRINT ID/Printer Number Request Contention (Race Condition)

If an AEPID is sent to the 3274 at the same time the 3274 is sending a WDAT, Lock Request, or WCUS(56) to the device, a FRA terminates the sequence as follows:

```
3274                                DISTRIBUTED FUNCTION
                                          DEVICE

   <--------------/                  PRINT ID AEPID
                  /-------------->  WDAT, LOCK, or WCUS(56)
   ------------------------------->  RPID (RQST PTR ADDR)
   <------------------------------- FRA (ABORT AEPID SEQ)
            (EXIT PRINT ID MODE)

      (the device executes the WDAT, LOCK, or WCUS(56)
       and returns FC.)
```

9.10.2 MATRIX CHANGE

The PAM is changed via host application combined with an operator keyboard request. This must be conducted from port 0 (a non TCA Device). As a result of a matrix change the appropriate indicator is broadcast to all devices, including Distributed Function Devices (via a WCUS).

Two situations are under consideration here:

Matrix load

1. IML (new matrix)
2. ALT/ERASE EOF(Port 0)

```
3274                                DISTRIBUTED FUNCTION
                                          DEVICE

                                 ----------------------> WCUS (COMPLT STAT) nn
                                                            (asynchronous event)
```

PRINTER ASSIGNMENT WCUS(56) RESPONSE

1. Current assignment no longer valid	WCUS (nn = 'FF' no assgn)
2. Current assignment no longer valid but new assignment available	WCUS (nn = 'FE' WHAT PRINTER)
3. Current assignment still valid	WCUS (nn = same nn PRINTER ASSIGNMENT)
4. No current assignment and a new one exists	WCUS (nn = new nn PRINTER ASSIGNMENT)

NOTE: If not customized for copy, there is no initial WCUS (nn). Thus, the device must assume no assignment initially (at POR).

9.10.3 COPY DATA > PRINT BUFFER

A normal print sequence occurs when the PRINT key is presses even though the copy data is greater than the printer buffer.

```
PRINTER                 3274              DISTRIBUTED FUNCTION
                                                  DEVICE

PRINTER ASSIGNMENT-------------------------------------------------------
                                <---------------- PRT RQST AECOPY
                                ----------------> WCUS (51)
                                                    (REQUEST QUEUED)
                                .
                                .
        (Printer available)
                                ----------------> WCTL
                                                    (WRT PRT PROFILE)

LOAD TRANSFER------------------------------------------------------------
                                ----------------> RDCOPY (1)
                                <---------------- COPY DATA
 (Data written to printer buffer)
                                .
                                .
                                .
                                .
                     (printer buffer full)

PRINT--------------------------------------------------------------------
 (Printing started)
                                ----------------> WCUS (58)(nn)
                                .                     (PRINTING STARTED)
                                .
 (Print completed)

CONTINUATION OF LOAD TRANSFER--------------------------------------------
                                ----------------> RDCOPY (2)
                                <---------------- COPY DATA
 (Data written to printer buffer)
                                .
                     (Printer buffer full)

PRINT--------------------------------------------------------------------
```

At this point the PRINT phase is repeated, and the LOAD TRANSFER phase is repeated until LDM is received from the device, as follows:

```
      PRINTER                 3274                DISTRIBUTED FUNCTION
                                                        DEVICE

CONTINUATION OF LOAD TRANSFER------------------------------------------
                              ----------------> RDCOPY (n)
                              <---------------- COPY DATA (LDM)
 (Data written to printer buffer)
                          (END COPY DATA)

PRINT------------------------------------------------------------------
 (Printing started)
                              .
                              .
                              ----------------> WCUS (59)
                              .                   (REQUEST DEQUEUED)
                              .
 (Print completed)
                          (COPY COMPLETE)
                              ----------------> WCUS (COMPLT STAT)(nn)
```

COPY COMPLETION STATUS	CU STATUS
1. Good Completion	WCUS(5B)
2. IR	WCUS(53)(01)
3. Equipment Check	WCUS(53)(02)
4. Data Check or parameter error (nn as shown in 9.10.1)	WCUS(53)(03)

9.10.4 COPY REJECTION

9.10.4.1 Prior To Service

PRINTER UNAVAILABLE (WHILE REQUEST QUEUED)

If the local copy terminates due to some error or suspension of processing, the copy is rejected in the following manner:

```
      PRINTER                 3274                DISTRIBUTED FUNCTION
                                                        DEVICE

                              <---------------- PRT RQST AECOPY
                              ----------------> WCUS (51)
                              .                   (REQUEST QUEUED)
                              .
                      ERROR/SUSPENSION ENCOUNTERED
                              ----------------> WCUS (COMPLT STAT)(nn)
```

COPY COMPLETION STATUS	CU STATUS
1. IR	WCUS (53)(01)
2. Equipment Check	WCUS (53)(02)
3. Busy With Host (Host initiated only)	WCUS (52)
4. Unauthorized due to matrix change	WCUS (55)
5. TO device not a printer	WCUS (55)
6. Printer w/o SCS	WCUS (55)
7. Printer in SYSTEM mode	WCUS (55)
8. No printer assigned (nn as shown in 9.10.1)	WCUS (55)

LONG TERM BUSY (PRINT KEY RQST ONLY)

```
PRINTER                 3274                DISTRIBUTED FUNCTION
                                                  DEVICE

                          <------------------ PRT RQST AECOPY
                          ------------------> WCUS (51)
                                                (REQUEST QUEUED)
                 (PRT ALLOC TO HOST)
                          ------------------> WCUS (52)
                                                (LONG TERM BUSY)
                          .
                          .
          (Printer freed up by host)
                          ------------------> WCTL(nn) (WRT PRT PROF)
                                               (implied request q'd)

     (PRT OPERATIONAL - REFER TO 9.10.3 AT RDCOPY)
  or (PRT NOT FUNCTIONAL - after data transfer started)
                          ------------------> WCUS (COMPLT STAT)(nn)
.
```

COPY COMPLETION STATUS	CU STATUS
1. IR	WCUS (53) (01)
2. Equipment Check (nn as shown in 9.10.1)	WCUS (53) (02)

9.10.4.2 Immediate Rejection - Copy Unauthorized

Copy rejection can also occur on this sequence:

```
PRINTER                 3274                DISTRIBUTED FUNCTION
                                                  DEVICE

                          <------------------ PRT RQST AECOPY
                          ------------------> WCUS (COMPLT STAT)(nn)
```

COPY COMPLETION STATUS	CU STATUS
1. No Copy Configured	WCUS (5A)
2. Second request race condition (Printer Busted)	WCUS (53)

9.10.4.3 Printer Error During Data Transfer

If an error on the printer is encountered during the Load Transfer phase, no printing takes place in sequence below:

```
        PRINTER              3274              DISTRIBUTED FUNCTION
                                                     DEVICE

                             <---------------- PRT RQST AECOPY
                             ----------------> WCUS (51)
                                                (REQUEST QUEUED)
                             .
                             .
   (Printer available)
                             ----------------> WCTL (WRT PRT PROF)
                             ----------------> RDCOPY
                             <---------------- COPY DATA
 (Data written to printer buffer)
                             .
 --Printer error--           .
    Before printing started:
           (IF OTHER PRT AVAIL, REFER TO 9.10.3 at WCTL)
            Otherwise, and
   After printing started:
                             ----------------> WCUS (COMPLT STAT)(nn)
```

COPY ERROR COMPLETION STATUS	CU STATUS
1. IR (Includes power off)	WCUS (53) (01)
2. Permanent error (nn as shown in 9.10.1)	WCUS (53) (02)

9.10.5 DEVICE CANCEL

AECAN dequeues the request only if the request is queued but not actually being serviced yet.

```
     PRINTER                 3274              DISTRIBUTED FUNCTION
                                                     DEVICE

                             <---------------- DEV CAN RQST AECAN
                             ----------------> WCUS (COMPLT STAT)(nn)
```

COMPLETION STATUS	CU STATUS
1. Good Completion (Queued and not being serviced or, not queued)	WCUS (56)

9.10.6 DEVICE CANCEL/WCTL RACE CONDITION

If an asynchronous AECAN request is sent to the 3274 by the device at the same time the 3274 is sending a WCTL function request to the the device after an operator initiated copy request has been queued, the following sequence occurs:

```
PRINTER                 3274                DISTRIBUTED FUNCTION
                                                  DEVICE

                     <---------------- AECOPY (00)
                     ----------------> WCUS(51) (RQST QUEUED)
                     <--------/AECAN (DEV CAN RQST)
                            /-------> WCTL nn (WRT PTR PROFILE)
                     <---------------- FRA (ABORT SEQUENCE)
                     ----------------> WCUS(nn) (COMPLT STAT)*

COMPLETION STATUS                     CU STATUS

      Good Completion                   WCUS (56)
                                     (previous ptr assignment)
```

10.0 DATA BASE OPERATIONS

Data Base operations are initiated by the device with either AEDBA (Fetch) or AEDBS (Store) asynchronous status. The CU accesses or stores data base data and communicates with the device using WDBD, RDBD, CNOP, and WCUS function requests.

10.1 DOWN STREAM LOADING (DSL) - 3290 ONLY

After a 3290 powers on and the 3274 has successfully written WCUS:READY to the device, the 3290 may request a Down Stream Load (DSL). Before responding with a POR and while in the "offline" state, the 3290 guarantees a 4096 byte TCA buffer is available to the 3274 by setting this value (4096) in DBUF. When the DSL operation is complete, the 3290 may respond with a request to go "online."

All requests used in these processes use a one byte file identifier.

NOTE:
These DSL files represent microcode and data entities required by the 3290. The 3290 Load Disk contains a disk directory with entries for each file (including shared tables in the 3274). DSL file are normally accessed in ascending disk sequence to minimize 3274 response time. Each file must begin and end on a record boundary.

Requests which cannot be satisfied cause WCUS status to be written to the inhibit area of the requesting device. In addition, a WCUS inhibit or reminder status is broadcast to all devices with a data-base-access capability (except data-base errors). Cover open/Cover closed are broadcast as Disk Reminders. Access errors are the following:

- Machine Check (not ready):

  386 unrecoverable disk overrun error
  387 disk media initialization error
  388 disk media error
  389 disk hardware error

- Data Base Error: file not found (includes controller RAM tables)
  (The file ID requested by AEDBA was not found in the device System Disk DSL file directory.)

- Mechanism Not Ready : Cover open/disk not inserted.
  (Disk reminder is broadcast if mechanism becomes readied.)

The 3290 is required to display the following error numbers for disk access errors in the 3274.

3290 Number	WCUS Value	3274 MC Error	Probable Cause:Result
630*	7000	389	Fatal hardware error: disk adapter interface is disabled.
631	7102	---	File not found: possible 3290 logic error.
632*	7004	388	Disk media error - defective diskette: replace.
633*	7006	386	Disk overrun - bad record: replace diskette.
634	7108	---	Attempted to write a non-writeable file: possible 3290 logic error.
635*	700A	---	Disk not ready: cover may be open/no disk.
636*	710C	---	Disk file locked (terminal contention): retry.
637	710E	---	Disk file overflow attempting to write too many records: possible 3290 logic error.
638	7110	---	Disk file not readable, protocol error.
639	7112	---	Disk file not locked (attempt to Write a file not locked) - protocol error.
640	7014	---	Wrong disk in 3274: replace diskette and retry.

* The 3290 is not required to report 630, 632, 633, 635 or 636 errors to the 3274 (AEER Status). If they are sent, the 3274 logs them, but does not generate an ALERT either because the 3274 already generated a 3NN alert (e.g., disk hardware error) or because the condition is not considered an error (e.g., "disk not ready").

10.1.1 DEVICE POWER ON DSL INTERACTION

Example:

```
 Comments    3290                        3274      Comments

Device POR
Run BATS      —POR SPECIAL STATUS——>                Perform basic
                                                       device initialization
              <——WCUS:READY————————                 - Ready for DSL
              ————————FC————————————>
Request Config
Data          ——AEDBA:CONFIG————————>
              <——WDBD:CONFIG————————               Fetch and write config data
              ————————FC————————————>
Request Diag.——AEDBA:DIAG————————>
              <——WDBD:DIAG————————                 Fetch and write diagnostics
Run diagnostics———FC————————————>
                      ETC.
Request Base Code        .
Request Keyboard         .
      Table              .
Request EBCDIC/Internal  .
      Translation Table  .
Request Character        .
      Generators         .
Request Attachment       .
      Code               .
Request Feature          .
      Code (Optional)    .

Finish init. ———AEDV:ONLINE————————>              Notify host
```

NOTE: The order of transaction, and actual files, are device dependent.

10.2 3274 CONFIGURATION TABLE

Customization data for the Distributed Function Device feature/function is stored in the 3274 Configuration Table. Per controller customizing, these bits are defined as set by the customer and are not validity checked by the controller. They are defined as follows:

Displ	Length	Value	Description
01	1		Feature Disk Level Identifier (Packed Decimal)
02	1		System Disk Level Identifier (Packed Decimal)
03-1D	27		Reserved
1E	1		Language Code (3274 Customizing Question 121)
		X'00'	Reserved
		X'01'	Domestic EDCDIC
		X'02'	ASCII (USA/English)
		X'03'	Austrian/German
		X'04'	Belgian
		X'05'	Brazilian
		X'06'	Canadian (French)
		X'07'	Danish

Displ	Length	Value	Description
		X'08'	Danish (Alternate)
		X'09'	Finnish/Swedish
		X'0A'	Finnish/Swedish (Alternate)
		X'0B'	French (Qwerty)
		X'0C'	French (Azerty)
		X'0D'	Austrian/German (Alternate)
		X'0E'	International
		X'0F'	Italian
		X'10'	Japanese (English)
		X'11'	Japanese (Katakana)
		X'12'	Portuguese
		X'13'	Spanish
		X'14'	Spanish (Alternate)
		X'15'	Spanish Speaking
		X'16'	UK English
		X'17'	Norwegian
		X'18'	Finnish/Swedish
		X'19'	English (WT)
		X'1A'	Norwegian (Alternate)
		X'1B'	Finnish/Swedish (Alternate)
		X'1C'	Portuguese (Alternate)
		X'1D'	Canadian Bilingual
		X'1E'	French Azerty 105
		X'1F'	Swiss German
		X'20'	Swiss French
		X'21-FF'	Reserved
1F-25	7		Reserved
26	1		Controller Options (Flags)
		X'00'	Reserved
		X'01'	3270 Extended Datastream Present
		X'02-FF'	Reserved
27-41	27		Reserved
42	1		Controller/Device PATCH ID #1 (Packed Decimal) Last 2 digits of PTR or '00'
43	1		Controller/Device PATCH ID #2 (Packed Decimal) Last 2 digits of PTR or '00'
44	1		Controller/Device PATCH ID #3 (Packed Decimal) Last 2 digits of PTR or '00'
45	1		Controller/Device PATCH ID #4 (Packed Decimal) Last 2 digits of PTR or '00'
46	1		Controller/Device PATCH ID #5 (Packed Decimal) Last 2 digits of PTR or '00'
47	1		Controller/Device PATCH ID #6 (Packed Decimal) Last 2 digits of PTR or '00'
48	1		Controller/Device PATCH ID #7 (Packed Decimal) Last 2 digits of PTR or '00'
49	1		Controller/Device PATCH ID #8 (Packed Decimal) Last 2 digits of PTR or '00'
4A	1		Controller/Device PATCH ID #9 (Packed Decimal) Last 2 digits of PTR or '00'
4B	1		Controller/Device PATCH ID #10 (Packed Decimal) Last 2 digits of PTR or '00'

Displ	Length	Value	Description
4C	1		Controller/Device PATCH ID #11 (Packed Decimal) Last 2 digits of PTR or '00'
4D	1		Controller/Device PATCH ID #12 (Packed Decimal) Last 2 digits of PTR or '00'
4E	1		Controller/Device PATCH ID #13 (Packed Decimal) Last 2 digits of PTR or '00'
4F	1		Controller/Device PATCH ID #14 (Packed Decimal) Last 2 digits of PTR or '00'
50	1		Controller/Device PATCH ID #15 (Packed Decimal) Last 2 digits of PTR or '00'
51	1		Controller/Device PATCH ID #16 (Packed Decimal) Last 2 digits of PTR or '00'
52	1		Number of Microcode RPQ's
		X'00'	None
		X'01'	1
		X'02'	2
		X'03'	3
		X'04-FF'	Reserved
53	1		Reserved
54	5		RPQ #1 (Packed Decimal; Must = 0 if not used)
54-55			Last 4 Digits of RPQ Number
56-58			EC Level of RPQ (6 digits)
59	5		RPQ #2 (Packed Decimal; Must = 0 if not used)
59-5A			Last 4 Digits of RPQ Number
5B-5D			EC Level of RPQ (6 digits)
5E	5		RPQ #3 (Packed Decimal; Must = 0 if not used)
5E-5F			Last 4 Digits of RPQ Number
60-62			EC Level of RPQ (6 digits)
63-8F	45		Reserved
90	1		3290 Load Disk EC Number (Packed Decimal)
91	1		3290 Load Disk Suffix Level (Packed Decimal)
92-96	5		3290 RPQ ID Number (Packed Decimal) Last 5 Digits of RPQ Number (Must = 0 if not used)
97-9B	5		Reserved
9C-9D	2		Distributed Function Device RPQ Parameters (Free Form; Device Dependent)
9E	1		Reserved

Displ	Length	Value	Description
9F	1		Distributed Function Device Utility Field (3274 Customizing Question 173)
		Bit 0 = 0	The Distributed Function Device will not send form controls (SCS) to a printer in performing local copy
		= 1	The Distributed Function Device will send form controls (SCS) to a printer in performing local copy.
		Bit 1 = 0	No form feed before copy
		= 1	Form feed before copy
		Bit 2 = 0	No form feed after copy
		= 1	Form feed after copy
		Bits 3-7	Reserved (must = 0)
A0	1		Distributed Function Device Utility Field (3274 Customizing Question 174)
		Bits 0-7	Reserved (must = 0)
A1-A3	3		Reserved
A4	1		3274 RPQ Imbeds
		Bit 0	Clear Key
		Bit 1	Reserved
		Bit 2	Clicker Option
		Bits 3-7	Reverved
A5-CF	43		Reserved
D0	1		Response Time Monitor Configuration
		X'00'	Not Configured
		X'01'	RTM Configured. No Host. Port 0 Display
		X'02'	RTM Configured. No Host. All Ports Display
		X'03'	RTM Configured. Host Support. No Display
		X'04'	RTM Configured. Host Support. Port 0 Display
		X'05'	RTM Configured. Host Support. All Ports Display
		X'06-FF'	Reserved
D1-FC	44		Reserved
FD-FF	3		3290 Password (3274 Customizing Question 175) (Packed Decimal; '000000' to '999999')

11.0 MULTIPLE INTERACTIVE SCREEN SUPPORT

MIS enables the user to execute multiple independent sessions concurrently to different Logical Terminals, each with its own resources and characteristics on the same physical device through a single coax port. Each Logical Terminal can be uniquely addressed by the host. Definition and management of the device resources is a device responsibility. As a function of customization the Control Unit provides a static valid device address table for mapping a host device address to the correct port. With MIS, there is no longer a direct relationship between the device address to the host and the 3274 device port (coax) address. When the device requests to be put online to the host it includes the Logical Terminal Set (LTS). This set may be all or a subset of the Logical Terminals configured at the port address to which it is attached. A maximum of 5 logical terminals is supported on each device (128 maximum per 3274 on an SNA attachment, 32 maximum on a non-SNA attachment).

The following interface extensions are included to support MIS.

11.1 TCA FIELDS

11.1.1 3274 INITIALIZATION

CULTA fields, in addition to CUDPORT and CUAT, are initialized at power on by the control unit, reflecting customization data. The device addresses are ordered from lowest address to highest address. In non-SNA, device addresses range from 0 to 31 (decimal). In SNA, device addresses for Secondary Logical Units range from 02 to 129 (decimal)..

If only CULTA1 has been initialized (zero is valid), then the TCA Device on the attached port has not been customized with MIS capability.

MIS may be customized for two, three, four, or five terminals on a given port. If only CULTA1 and CULTA2 are initialized, the CU supports only two logical terminals. If all addresses have been initialized then the CU supports up to a maximum of 5 logical terminals on the attached port.

11.1.2 ONLINE-TO-HOST

Communications with Logical Terminal addressing is only meaningful and valid if the device is currently in the online-to-host state. The device buffer format includes an LU Address parameter for asynchronous status (DALTAD) and control unit function requests (CULTAD).

11.2 MODIFICATIONS

Communication between the control unit and the device requires an explicit Logical Terminal address. Those functions which are LTA specific are: WDAT, RDAT, PDAT, LOCK, CTCSS, WCUS (for Program Check, ACTLU/DACTLU), and AEEP. All other function requests are physical device level functions. DALTAD/CULTAD for physical device level functions must be set to value of X'FF' (which cannot be configured as a valid LT address). The following asynchronous status requests and control unit function requests are modified to support MIS:

11.2.1 AEDV

This status is used by the device to request that one or more of its logical terminals (LTs) be put ONLINE or that all of its LTs be taken OFFLINE from the host. It is also used to signal that a DUMP initiated by a Diagnostic Reset command has completed.

11.2.1.1 AEDV (Online)

This form of AEDV asynchronous status is used by the device to request that one or more of its LTs be put ONLINE to the host. When the controller issues a WCUS(10) (Readiness Group), it is required to set CULTA1-5 with the assigned non-zero logical terminal addresses if MIS is supported (these fields are X'FF' if the controller does not support MIS). When the device responds ONLINE to a controller that supports MIS, it must include an additional parameter (DAEP2) containing an ordered bit map corresponding to CULTA1-5 reflecting the LTs that are requesting online status:

```
Bit 0......1.......2.......3........4......5......6......7
    |       |       |       |        |      |      |      |
  CULTA1  CULTA2  CULTA3  CULTA4   CULTA5    0      0      0
```

A bit "on" represents a request to the controller to put that logical terminal online to the host. That logical terminal is requesting "power on" to the host. The controller must be capable of processing multiple logical terminal "power on" requests in the same AEDV status.

TCA support requires that an ONLINE request be issued by the device only when all logical terminals are OFFLINE. Logical terminals whose bits are off in the above byte remains OFFLINE.

The device cannot request a logical terminal to be put online if the field CULTAx has not been initialized with a valid address. Any attempt by the host to communicate with a valid logical terminal address which has not requested "online" status is treated by the control unit as communication to a powered off device.

Bit 0 in DAEP2, corresponding to CULTA1, must always be set if the port address has not been customized for MIS capability.

11.2.1.2 AEDV (Offline)

This form of AEDV asynchronous status is used by the device to request that all of its online LTs be taken OFFLINE from the host. The DAEP2 parameter must equal zero. At least one LT must be online for this request to be valid. This is equivalent to the physical device being powered off to the host.

11.2.1.3 AEDV (Dump Complete)

This form of AEDV status is used by the device to indicate that it has completed the transmission of a DUMP to the controller. It is only valid when used to terminate a dump operation that was initiated by the controller via a "Diagnostic Reset" command. The device must continue to respond to controller POLLs until it receives an acknowledgement from the controller.

11.2.2 AEEB (SECURITY KEY ON)

The 'End IR' condition on the non-SNA interface is considered global (multi-Logical Terminal). Since End-IR can potentially cause the same type of Device end/busy thrashing as power on from multiple LTs, End-IR cannot be scheduled serially from device code to host code. The device is responsible for determining which LTs owe the host a Device End. DAEP2 must contain a bit map (bits 0 thru 4 are valid, 5 thru 7 must be zero) representing online LTs which owe the host a device end, based on FCSE:IR having been returned to 'Lock' for the respective LTs. All zeros in DAEP2 is invalid.

11.2.3 AEER

Asynchronous error logging is performed on a device basis only (except Program Checks for SNA). DALTAD must be loaded appropriately.

11.2.4 WCUS (WRITE CONTROL UNIT STATUS)

Control Unit status which is LU specific, such a program check, must carry a valid Logical Terminal Address (CULTAD). If the status is non-LU specific, then the parameter must contain a X'FF'.

11.2.5 WCUS:LUSTATUS (SNA ONLY)

The following modification should be reflected in the base (5.2) section, whether the 3274 MIS capability is, or is not, utilized. The LU address parameter (CULTAD) replaces the same in CUFRP3. For ACTLU and DACTLU the control unit loads CULTAD with a valid SLU address.

ACTLU is accepted for a powered off device. However, WCUS is not issued to the device unless/until the LU in question enters the "online" state. Given this situation, the 3274 rejects Bind, if a Bind follows an ACTLU.

Note: an LU that is "offline" is a logically powered off device. Powered off devices are assumed to be 3278s.

Following AEDV:ONLINE status from the device, LU status (WCUS) is re-issued by the control unit to each logical unit in the online-to-host state.

For handling a Physical Unit reset (ACTPU-Cold or DACTPU) all Logical Units are deactivated. WCUS:LUSTATUS is broadcast to each device station. For this situation the LU address parameter contains X'FF', indicating that the entire LT set is inactive. A TCA device with MIS is responsible for deactivating each logical unit as an outboard function.

11.2.6 RACE CONDITIONS

Controller function requests must be processed by the TCA Device with normal responses unless asynchronous device status has been acknowledged. A specific example of this is when a host tries to write a message to a logical terminal at the same time device is going to host offline state (AEDV: offline). Until the control unit acknowledges this asynchronous status, the device must process the outbound request because the asynchronous status is still considered to be in a pending state.

11.2.7 WDAT

In SNA the logical terminal address on WDAT requests is contained in the 'DAF' field of the data, rather than in CULTAD. The specified LT in the DAF is verified by the control unit to be both valid and online. Although the current LT address must be obtained in the Transmission Header (DAF), CULTAD contains a valid, online LT address. CULTAD may be ignored for WDAT on SNA attachments.

11.2.8 BISYNC INBOUND/OUTBOUND CONTENTION

Inbound data (AEEP) remains in a pending read state until RDAT is issued by the control unit. If an outbound selection to a different logical terminal (LOCK request) is issued, the pending read state must be preserved by the device. The device must reschedule AEEP for the original logical terminal after receiving end sequence (CTCCS) to the outbound operation.

If the LOCK request is to the same LT that has the inbound data pending, the pending read state must be reset by the device and the inbound data request (AEEP) must not be rescheduled.

12.0 ATTACHMENT CONSIDERATIONS

A significant portion of the interface is attachment dependent. This section describes those dependencies.

12.1 SNA

The CU handles all PU services and link connection status. The Device is informed of the status of the link and PU via WCUS. The CU also processes ACTLU and DACTLU, as this can only be done by the CU for a powered off Device. Session status for a single session is also monitored to allow termination of the session at power off. Multiple sessions are not considered.

12.1.1 ACTPU/DACTPU

Any change in the PU status causes the following actions in the CU:

> When the device becomes "online" to host WCUS are issued to each Device for the Reminder Group and the LU Status. If the PU is not active, a Communications Check Reminder may be issued. The CU may delay issuing these function requests to the Device because of asynchronous processing of PU requests.

WCUS function requests for LU/PU status are only sent once per status change, or following device being placed online to host; i.e. LU active WCUS only issued once from ACTLU or online to DACTLU, ACTLU, DACTPU, ACTPU or offline.

12.1.2 ACTLU/DACTLU

If the device is "online" when the CU processes a valid ACTLU/DACTLU, ACTLU (ERP), a WCUS is issued for the LU Status. The CU returns a positive response to the host for a powered off (or "offline") device.

12.1.3 SEGMENTS PASSED TO THE DEVICE

Segments sent to the device via WDAT have been checked for length ≥ 6 (≥ 9 if FSFIC), FID=2, DAF'=device LU address. No segments failing these tests are sent to the device.

12.1.4 BIND

The 3274 detects RUs containing a BIND command. If the BIND is for a powered off Device, the 3274 returns a negative response X'080A' or X'0845' depending upon SSCP support of NOTIFY. If the device is powered on, a 'BIND PENDING' Finite State Machine (FSM) is set for that LU and the BIND is transferred to the device. When the BIND response is found in a subsequent inbound transmission, it is checked. If the response is positive, the BOUND FSM is set. If the response is negative, it is simply transferred inbound to the host without any FSM changes.

The device must reject Binds in the bound state which carry a different OAF than expected. No FSMs in the CU are modified.

12.1.5 UNBIND

When the 3274 detects an UNBIND for an LU with a valid OAF, it sends it on to the device. If the response is positive, all the device FSMs (except ACTLU) are reset. If the UNBIND has an invalid OAF, the 3274 rejects it.

12.1.6 POWER OFF, TCA DISCONNECT, OR OFFLINE STATUS CHANGE (AEDV)

While Bound:

If in this state, UNBIND and NOTIFY (if supported) must be sent inbound. The OAF/DAF combination saved at BIND time is used. The BIND FSM is reset.

While BIND PENDING:

A power-off response is returned by the 3274 to the host.

While UNBIND PENDING:

The BIND and UNBIND PEND FSMs are reset. An UNBIND is sent to the host by the 3274.

In all the above cases, any subsequent FMDATA, or in process chains or segments sent by the host, is rejected as outlined above.

12.1.7 OUTBOUND SEGMENTING

Segment checking is the responsibility of the 3274. The FOM and LOM flags associated with the data area are not used for the WDAT function request. A segment as pointed to by CUDP is loaded on a halfword boundary. A segment is sent to the Device with one WDAT function request. The CU detects segmenting errors. This data is not passed on to the TCA device.

12.1.8 INBOUND SEGMENTING

Segment gathering must be done by the device. The flags associated with the data area for the PDAT and RDAT function requests refer to a single RU as a message. When the 3274 issues a PDAT, it sets CUFRP1 and CUFRP2 to the maximum number of segments it (the CU) can accept and sets CUFRP3 and CUFRP4 to the maximum segment size (data,headers,flags,length) which it can accept. On receipt of PDAT the device must generate an entire RU of one or more segments at the buffer address pointed to by CUDP. The segments must contiguous in storage on halfword boundaries each containing a length field and a flags field. The length must be the true length of th segment, i.e., the length of length field + flag field + headers + data. This could be an odd number. The flags field must have FOM set in the first segment and LOM set in the last. FOM and LOM must be off in all other segments.

An FCIR response to a WDAT is used for response handling. When the CU issues an RDAT it sets CUFRP3 and CUFRP4 as above. The device must confine the RU generated to a single segment with FOM and LOM set. CUDP for the RDAT request points to an area of the buffer reserved for response and data base access (not overlapping with the area used by WDAT for outbound data).

12.2 BSC

The CU handles BSC protocols including transparency, inbound blocking, specific POLL, general POLL, selection, and line control. Checkpoint/restore is not supported. The device does not receive a WDAT request on a block of data until a valid BCC character has been received by the CU. However, the data may be written to the TCA buffer as it is received. The 3274 issues a Start Op is only after the full block is verified.

12.2.1 INBOUND OPERATION

The device must send status AEEP to the 3274 as a result either an operator AID generating action such as ENTER or by a host READ PARTITION structured field. The 3274 may improve its throughput by issuing a PDAT request prior to queueing the device on an inbound service list. The device must prepare the appropriate inbound data for the PDAT function request in the same manner as for the RDAT function request. Synchronous status FC must be returned when either all the data to be sent has been prepared or the TCA inbound buffer is full.

When the 3274 is ready to send the data to the host, it issues an RDAT function request to the device using the same parameters as the PDAT request. The device must recognize that the appropriate data has already be generated and immediately return synchronous status FC to the 3274. The RDAT parameters must be ignored if a PDAT was previously issued followed by a normal Function Complete.

A conversational reply (ETX only) appears as a normal inbound operation followed by an outbound operation.

All subsequent RDAT function requests from the 3274 require that the data stream be prepared synchronously.

The PDAT and RDAT parameters are as follows:

CUFRP1 - Unused

CUFRP2 - Unused

CUFRP3/4- length of block to be prepared
(257, except 255, 1st block)

CUDP - buffer address to place data.

DEVICE	3274	Comments
AEEP	——>	Request for inbound operation
PDAT	<——	Prepare data request
FC	——>	Data ready in TCA buffer, sched. host request to wait for POLL
RDAT	<——	Read data after POLL
FC	——>	Immediate response from the device
RDAT	<——	Generate additional inbound data
FC	——>	Data read, response delayed by the device processing time.
CTCCS	<——	Final ACK0/ACK1 received from host
FC	——>	OK

12.2.2 INBOUND PROCESSING

An inbound message is defined (for Beginning/End of Message flags) as the data to be sent from a terminal in response to a single POLL or data stream read command. The device must send read data blocked up to the maximum size specified by the parameter CUFRP34 as set by the CU.

12.2.3 SUBSEQUENT AID ACTIONS

If the operator attempts any AID action on the inbound partition following the initial one and before System Lock is reset, the action must be rejected with the retry indicator.

If the AID action is attempted on another partition (XSYSTEM not displayed), the minus function indicator must be displayed.

If the operator resets System Lock prior to RDAT and attempts a new AID action, the original data is invalidated. New data must be prepared in the buffer when the RDAT is received.

12.2.4 OPERATOR RESET ACTION

If the operator resets System Lock prior to PDAT, status FRA must be returned to the PDAT request.

If the operator resets System Lock prior to RDAT, status FRA must be returned to the RDAT request.

12.2.5 TRANSPARENCY

When sending data inbound in transparency mode, the length of the data prepared must be one less than the maximum length specified in CUFRP3/4 of PDAT. This is to allow for the insertion of a DLE character (by the CU).

12.2.6 TEST REQUEST KEY

If a Test Request key action is to be performed, the Test Request flag in the data header must be set and the appropriate inbound data placed in the buffer. The length of the inbound data prepared must be 2 bytes less than the maximum allowed (one byte test request header + one for transparency).

The CU generates the actual SOH%/STX header on the message to the host.

12.2.7 ASYNCHRONOUS HOST SELECT

If a Lock request is made following the recognition by the device of the inbound operation but prior to the receipt of PDAT, the lock must be rejected by the device with status FCSE:BUSY.

If a Lock request is made following the PDAT request but prior to receipt of RDAT, the Lock must be accepted by the device and the pending data must be discarded.

12.2.8 OUTBOUND PROCESSING

An outbound message is defined (for Beginning/End of message flags) as the data contained between STX,ESC and ETX/ETB. The CU places the outbound data in the data portion of the Device Buffer but does not issue a WDAT request until a good BCC is received. The amount of data which may be transferred to the device at one time is only limited by the size of the Device Buffer.

If a READ command is encountered, the device must return status FCIR to the CU and wait for RDAT requests from the CU.

When selection is terminated, CTCCS is issued.

DEVICE	3274	Comments
	<——— LOCK	Lock request when select received
	FC ———>	Lock performed
	<——— WDAT	Indicate outbound data in TCA buffer
	FC ———>	Datastream processed
	<——— CTCCS	End Sequence
	FC ———>	OK

12.2.9 SECURITY KEY

If the Security key is in the lock position and a lock request is received, the device must return FCSE:IR. When the Security key is subsequently turned off, the device must return asynchronous status AEEB:IR to the 3274.

The 3274 then sends DE to the host.

12.3 NON-SNA LOCAL CHANNEL

The 3274 processes the local channel protocols not requiring device interactions as well as those requiring an immediate response to enhance channel performance. The device is responsible for processing commands and data stream.

The device must have an 8K (8192) byte TCA buffer.

12.3.1 INBOUND OPERATION

The device sends status AEEP to the 3274 as a result of either an operator AID generating action such as ENTER or by a host READ PARTITION structured field. The 3274 then sets attention status on the channel. The device must then wait for a READ type command from the host.

If the device has accepted a Lock Request, the device is considered to entered "receive" state and must prevent the operator from generating an inbound operation.

The 3274 may improve its throughput by issuing a PDAT request prior to sending ATTN to the host. The PDAT must be processed prior to sending ATTN. The device must prepare the appropriate inbound data for the PDAT function request in the same manner as for the RDAT function request. Synchronous status FC must be returned when either all the data to be sent has been prepared or the TCA block limit has been reached.

When the 3274 receives a Read or Select Read type command for the device, the appropriate Read command is passed as parameter CUFRP1 of the Write Local Channel Command (WLCC) function request. If the command was Read Modified and a PDAT request had previously been processed, status FCIR:DA must be returned. The 3274 may then read the data from the TCA buffer without making an RDAT request. If more data remains to be sent, the 3274 issues RDAT requests which cause the datastream to be prepared synchronously.

If the command was Read Buffer or PDAT had not been processed, status FCIR:READ must be returned. The 3274 normally follows with RDAT function requests until all of the read data has been transmitted. The device must set the last of message flag in the header for the last block sent to the 3274.

Parameter CUFRP2 of the next WLCC or CTCCS function request is set to X'01' if the host read operation was successful.

The PDAT and RDAT parameters are as follows:

```
CUFRP1  - Unused
CUFRP2  - Unused
CUFRP34 - Maximum length of block;  X'0E0E'.
CUDP    - Buffer address to place data.
```

DEVICE	3274	Comments
AEEP	——>	Request for inbound operation
PDAT	<——	Prepare data request
FC	——>	Data ready in TCA buffer, queue to host to send ATTN
LOCK	<——	Lock request when select received
FC	——>	Lock performed
WLCC:RM	<——	Send Read Modified command
FCIR:Data avail	——>	Device ready to send read data
RDAT	<——	Generate additional inbound data
FC	——>	Data ready, response delayed by the device processing time.
CTCCS	<——	End Sequence
FC	——>	OK

12.3.2 SUBSEQUENT AID ACTIONS

If the operator attempts any AID action on the inbound partition following the initial one but before System Lock is reset, the action must be rejected with the retry indicator. If the AID action is attempted on another partition (XSYSTEM not displayed), the minus function indicator must be displayed.

If the operator resets System Lock prior to WLCC:RM and attempts a new AID action, the original data is invalidated and the new appropriate data must be generated when RDAT is received.

12.3.3 OPERATOR RESET ACTION

If the operator resets System Lock prior to PDAT, status FRA must be returned to the PDAT request. The 3274 may then forget that AEEP was received.

If the operator resets System Lock prior to WLCC:RM, the AID value, cursor address, and modified data from partition 0 must be placed in the TCA buffer upon receipt of RDAT. If partition 0 does not exist, only the reset AID and cursor address are returned.

:ST REQUEST KEY

Request reads, the device must place SOHX/STX in the data area but must
ne Test Request flag in the data header.

SYNCHRONOUS HOST SELECT

request is made between the operator AID action but before it is
d by the 3274, the Lock must be rejected by the device with status
. The receipt of AEEP by the 3274 indicates that the busy condition no
ists.

HOST COMMANDS OTHER THAN READ MODIFIED

) request other than WLCC:RM is received, the device must invalidate the
eady prepared and return synchronous status FCIR:READ. When the RDAT
is received, the new data must be generated as appropriate for the

of a write type command invalidates the prepared data and the device
urn synchronous status FC.

OUTBOUND PROCESSING

ype (and Erase All Unprotected) commands are passed to the device as
er C[illegible]1 of the WLCC function request. The actual data (none for EAU
) is [illegible]nt to the device via WDAT function requests.

astream or chaining errors detected by the device must cause synchronous
FCSE to be returned with a parameter specifying Op Check.

3274	Comments
DCK	Lock request when select received
FC →	Lock performed
C:command	Send Write type command
FC →	The device ready to accept data
WDAT	Indicate outbound data in TCA buffer
FC →	Datastream processed
CTCCS	End Sequence
FC →	OK

13.0 COMMUNICATIONS AND SYSTEMS MANAGEMENT (C&SM)

Response Time Monitor (RTM) and Alert are C&SM functions that use the SNA SSCP-PU (controller) session to transmit data. In addition, RTM is supported on non-SNA attachments as a Subsystem only function (no host interface).

13.1 RESPONSE TIME MONITOR

The 3274 Response Time Monitor is a mechanism whereby end-to-end user response time can be measured depending on a definition dictated by the controller customizing process or, in certain cases, an application in the host. Response times for each logical terminal are measured and maintained in the controller. See Section 8.2 for a more detailed definition and description of the RTM.

13.2 ALERT FUNCTION

The alert function is a mechanism whereby the controller can pass error conditions relating to problems within the subsystem to a host via the SNA SSCP-PU session. If supported by the controller, such errors are passed up to the host via the Network Management Vector Transport (NMVT) and do not require any additional support from the attached device. The Alert function does require NPDA programming support in the host that supports the 3274 controller but this program does not require any additional support for 3290 displays. This C&SM function is intended for enhanced customer network problem determination and management. This function is not available on non-SNA attachments.

In addition to those errors detected by the controller on behalf of the devices connected to it, if the 3290 detects an error and reports it to the controller via AEER status reflecting a 6NN or 7NN error code (see section 7.1), this error also causes an alert record to be sent up to the host. Certain 6NN errors do not cause alerts to be generated by the controller as these are the result of controller detected problems or are considered status rather error conditions.

13.3 EXPANDED CONFIGURATION RECFMS(05) SUPPORT (3290 ONLY)

Some configurations of the controller microcode support an enhanced RECFMS(05) record when attached via a SNA protocol. When the controller receives an REQMS(05) from the host on the SSCP-PU session, the controller responds with the RECFMS header and the first 241 bytes of its configuration table. Since the 3290 display, which downstream loads (DSL) from the controller, also maintains some of its configuration data in this table, it also becomes available for network management.

This enhancement does not require any additional support from the 3290.

******* END OF DOCUMENT *******