USERS GUIDE

SYNCHRONOUS COMMUNICATIONS ACCESS METHOD
DIAL USER APPLICATION GUIDE

SYSTEM TEN COMPUTER BY SINGER

SINGER
FRIDEN DIVISION

USERS GUIDE

SYNCHRONOUS COMMUNICATIONS ACCESS METHOD
DIAL USER APPLICATION GUIDE

SYSTEM TEN* COMPUTER BY SINGER

CONTROL NO. C218AA
DECEMBER 1971

SINGER
FRIDEN DIVISION
2350 WASHINGTON AVE
SAN LEANDRO, CALIF. 94577

*A trademark of the Singer Company

PRINTED IN U.S.A.

PREFACE

This document is a guide to the design and coding of a user application program (UAP) that will operate with an SCA Module in the calling or answering station of a switched-line network. It should be used as a reference, in conjunction with the general SCAM reference manual (SCA Data Communications, Chapter 2), by analysts and programmers responsible for the development of these programs.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

This document is a guide to the design and coding of a user application program that is to operate with an SCA Module in the calling or answering station of a switched-line network.

Section 2 defines switch-line concepts and terminology.

Section 3 is a step-by-step guide to the design of a dial-calling UAP.

Section 4 presents an example of a dial-calling UAP.

Section 5 is a step-by-step guide to the design of a dial-answering UAP.

Section 6 presents an example of a dial-answering UAP.

Section 7 presents supplemental information about the use of SCAM in dial configurations. This section will be deleted when the general SCAM manual (SCA Data Communications Chapter 2) is updated and rereleased.

Appendix A is a listing of the sample dial-calling UAP presented in Section 4.

Appendix B is a listing of the sample dial-answering UAP presented in Section 6.

CHANGE IN TERMINOLOGY

In previous SCA software documents, the term "Synchronous Communications Adapter Module" (SCA Module) has been used to refer to all routines released to support SCA communications. The same term has been used to refer to any program assembled from some selection of these routines. In this and subsequent documents, the term "Synchronous Communications Access Method" (SCAM) will be used to refer to the distributed routines. The term "SCA Module" will still be used to refer to a program assembled from some selection of the SCAM routines.

In previous documentation the term "Dial-Out" has been used to refer to a station, UAP, or SCA Module that initiates calls over a switched line. The term "Dial-In" has been used to refer to a station, UAP, or SCA Module that receives calls over a switched line. In this and subsequent documents, the term "dial-calling" or "calling" will be used in place of "dial-out". The term dial-answering" or "answering" will be used in place of "dial-in".

SWITCHED LINE CONCEPTS AND TERMINOLOGY

A two-way alternate, switched (dial) connection is a temporary connection of two stations over the public telephone network in order to achieve two-way data communications. A connection is established when a dial-calling station successfully calls a dial-answering station, and is terminated when either station executes a disconnect command. Normally both stations issue a disconnect command when a serious error occurs or when either station sends a DLE EØT.

After a connection has been established and a data file and an EØT have been transmitted, it is not necessary to re-dial before beginning a new file transmission. A receiving station may request to become the sending station at the end of any file transmission.

I AM AND WHO ARE YOU IDENTIFICATIONS

Over public telephone lines it is possible for any station to communicate with any other station if their software and hardware are compatible. Therefore it is possible that data may be "stolen" by or received from a station that is not a legitimate part of a given communication network.

To help prevent theft of data or receipt of invalid data, the user may assign identifying codes to each station in his network. These codes will be transmitted and checked for validity each time a connection is made, before data transmission begins. A station's identifying code is called its I AM ID. Each station that is to perform ID checking must have a list of the I AM codes of the stations with which it may communicate. The codes in this list are called the WHO ARE YOU (WRU) IDs. ID checking can be performed by the calling or answering station or by both or neither.

The UAP (calling or answering) defines its station's I AM ID and the list of valid WRU IDs when it initializes the SCA Module. (See Initialization Record in Chapter 2.) The Handshaking Routine (calling or answering) automatically sends an I AM ID if one has been defined and automatically performs ID checking if a list of WRU IDs has been defined. If a station that is to perform ID checking receives an invalid WRU ID or no WRU ID, the station's Handshaking Routine disconnects the line.

The IDs may be defined or redefined whenever initialization occurs. Additional security may be achieved by periodically changing the network's IDs.

NORMAL SEQUENCE OF EVENTS -- CALLING STATION

The Dial-Calling SCA Module waits for the UAP to place the phone number of a station to be dialed in the Call Work Area in Common. It then calls the station, sends an ENQ, and reads an ACK0. If an I AM ID was passed to the SCA Module through initialization, then it sends its I AM ID preceding the ENQ. If the calling SCA Module is to perform ID checking, it expects to receive a WRU ID preceding the ACK0, and it will not continue unless it receives a valid WRU ID.

After the (ID) ENQ has been sent and the (ID) ACK0 has been received, the calling SCA Module sends either data or an EOT, as directed by the UAP. If there is data to send, the SCA Module transmits data until the UAP requests that it send an EOT.

After it has sent an EOT, the calling SCA Module expects to receive an EOT or an ENQ from the answering station. If it receives an ENQ, the calling SCA Module reads data until it receives an EOT. It then checks to see if its UAP has more data to send.

The calling SCA Module continues to alternately send data or an EOT and then receive data or an EOT as long as either station has data to send or until a serious error occurs or it receives a DLE EØT. It assumes that both stations have finished sending data when it receives an EOT instead of an ENQ from the answering station and it finds that its UAP has no data to send. When this happens or when a serious error occurs or a DLE EOT is received, the calling SCA Module sends a DLE EOT, disconnects the line, and waits for another phone number to dial. (Section 3, Write Data step 3c describes a method by which the UAP can force the calling SCA Module to continue to exchange EOTs even after it has detected that neither station has data to send.)

NORMAL SEQUENCE OF EVENTS -- ANSWERING STATION

The Dial-Answering SCA Module waits for a call to come in from a calling station. When a call has come in, it reads an ENQ and responds with an ACKO. If the answering station is to perform ID checking, the SCA Module expects to receive a WRU ID preceding the ENQ, and it will not continue unless a valid WRU ID is received. If an I AM ID was passed to the SCA Module through initialization, the SCA Module sends its I AM ID preceding the ACKO.

After the (ID) ENQ has been received and the (ID) ACKO has been sent, the SCA Module expects to read data. It normally reads data until it receives an EØT. If the calling station has no data to send, it can send an EØT immediately after sending an ENQ and receiving an ACKO.

After an EOT has been received, if the UAP wants to send data, the SCA Module sends an ENQ. If the calling station responds with an ACKO, the SCA Module begins transmitting data supplied by the UAP. It transmits data until the UAP requests that it send an EØT. If the calling station sends an ENQ in response to the EØT, the SCA Module sends an ACKO and again expects to read data.

The dial-answering SCA Module continues to alternately receive data or an EOT and transmit data or an EOT until the UAP requests that it disconnect the line, a serious error occurs, or the calling station sends a DLE EOT. A serious error or the receipt of a DLE EOT causes the SCA Module to post an error to the UAP and disconnect the line. After disconnecting the line, the SCA Module waits for another call to come in.

DESIGN OF A DIAL-CALLING USER APPLICATION PROGRAM

INTRODUCTION

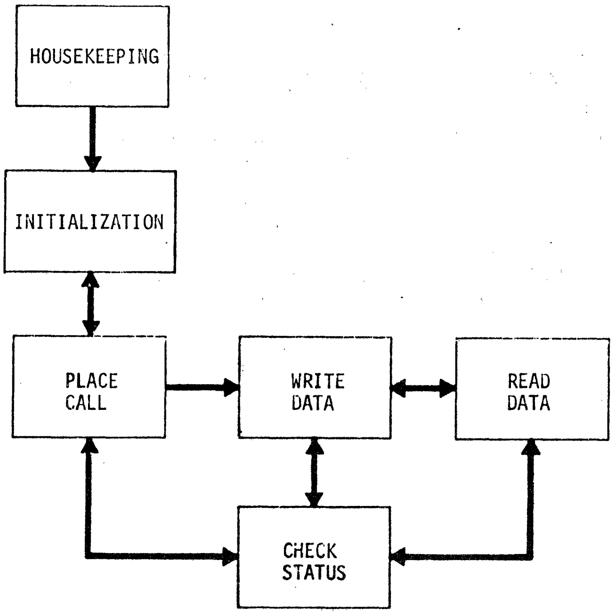
This section is a step-by-step description of the operations to be performed by a user application program (UAP) that is to function with a Dial-Calling SCA Module.

PROGRAM STRUCTURE

In this section the functions of a calling UAP have been divided into six general categories:

- Housekeeping
- Initialization
- Place Call
- Write Data
- Read Data
- Check Status

The following diagram shows the relationship of these functional categories.



Housekeeping is performed once only, immediately after the program is loaded. Initialization is performed after housekeeping. If Version 2 of the Initialization Routine was chosen, the UAP may reinitialize before placing a new call. If Version 3 of the Initialization Routine was chosen, the UAP must reinitialize before placing a new call. The UAP places a call after it first initializes. It places a new call when an error has occurred, when neither station has any more data to send, or when for any reason it disconnects the line. The UAP writes data and an EOT or just an EOT after placing a call. It may write data again after receiving an EOT from the answering station. It reads data after sending an EOT. It checks status whenever it is waiting for the SCA Module to transmit or receive data.

A dial-calling station always writes first after placing a call. A dial-calling UAP must include Write Data steps 1 and possibly 2 or 3, and a Dial-Calling SCA Module must have a Write Routine and at least one Transmit Buffer even for an application in which the calling station never actually transmits data. However, for an application in which the calling station never receives data, the Dial-Calling SCA Module need not have a Read Routine, and the calling UAP need not include Read Data logic. (Read Routine in Section 7 describes programming considerations for a Dial-Calling SCA Module assembled without a Read Routine.)

HOUSEKEEPING

1. Determine that the SCA Module has been loaded and is active. Test ↑ECB+6,7 for a valid SCA partition number. If the SCA Module has not been loaded within some time frame (e.g., 10 cycles through the machine), it is advisable to print an error message and terminate.

2. Initialize the SCA Module's PSB to N (normal).

3. Load initial values into other areas of Common if this was not done by defining constants into them at assembly time.

4. Perform any application-determined housekeeping procedures such as initializing work areas, opening files, etc.

If the SCA's partition number is constant and known at assembly time, then step 1 can test for a specific partition number in ↑ECB+6.7. Step 2 can be eliminated because the correct PSB can be initialized to N at assembly time.

INITIALIZATION

1. Wait for the System Mailbox to become available. Test the Mailbox Status Byte for an A (available). If it is not an A, branch and switch to this test (wait). If it is an A, continue.

   If no other partitions are using the Mailbox, the constant A should be defined into the Mailbox Status Byte at assembly time or this step should be omitted.

2. Move the Initialization Record to the Mailbox, post the Mailbox busy, and address it to the SCA partition.

   a. Move a B to the Mailbox Status Byte.

   b. Move a CO (alpha O) to the To-Address field of the Mailbox.

   c. Move the UAP's partition number or ID to the From-Address field of the mailbox.

   d. Move the Initialization Record to the Mailbox. Normally all or most of the Initialization Record can be defined as a constant in the UAP source deck. However, for some applications it may be necessary for the UAP to perform some workstation or other I/O to get information to complete the Initialization Record.

3. Reset the Inhibit Switch to zero. (This step is optional if this is the first or only initialization and the Inhibit Switch was initialized to zero at assembly time or at Housekeeping step 3.)

4. Analyze the result of the attempt to initialize the SCA Module.

   a. Test for an initialization error. (Test the Inhibit Switch for an E.) If there has not been an initialization error, continue to step 4b.

      Normally an initialization error will require analysis and perhaps reassembly. Therefore the UAP should print an error message requesting a memory dump, reset the Mailbox Status Byte to A (available), and terminate.

      The SCA Module does not reset the System Mailbox to A after posting an initialization error. This allows the UAP to print the invalid Initialization Record in the Mailbox before allowing other partitions to use the mailbox. Note that the SCA Module may have destroyed part of the contents of the Mailbox while trying to process the Initialization Record.

INITIALIZATION, Step 4

b. Test to see if initialization has been successfully completed. After successful initialization the SCA Module moves the UAP's partition number (the From-Address field of the System Mailbox) to the Partition that Last Initialized field (↑ECB+4,5) of the CCB and posts the System Mailbox available (A in the Mailbox Status Byte). The UAP can test either or both of these fields to see if initialization has been successfully completed.

If Initialization has not been successfully completed, return (branch and switch) to step 4a. If initialization has been successfully completed, continue.

Note: the UAP loops through step 4 until initialization has been completed. The UAP should not wait indefinitely for the SCA Module to process the Initialization Record. If after some period (e.g., ten cycles), the SCA Module still has not processed the Initialization Record, the UAP should print an error message, free the Mailbox, and perhaps take a memory dump. It is likely that the SCA partition has not been loaded properly. The user's first attempt to correct the problem should be to reload the SCA partition and try again. If this fails, further problem analysis is required.

PLACE CALL

1. If this is to be the first call after the UAP is loaded, initialization has already taken place. If this is not the first call, and Version 3 of the Initialization Routine was selected, the UAP must reinitialize before continuing. If Version 2 was selected, the UAP may reinitialize. To reinitialize, execute the Initialization logic.

2. If this station has data to transmit after the call is placed, perform Write Data steps 4-7. These steps should be performed before the call is placed so that there is a data block for the SCA Module to transmit immediately after it places the call. If this preparation is not made, one or both stations may time out after the connection is made while the calling SCA Module waits for data to transmit.

3. Prepare the Call Work Area.

   a. Move the length of the phone number to the Length field (↑CWA+18,19) of the Call Work Area.

   b. Move the phone number to the Phone Number field (↑CWA+20).

   c. Move a D (ready) to the Call Work Area Status Byte (↑CWA).

PLACE CALL

4. Test to see if the call has been successfully placed.

   a. The SCA Module posts an A (available) in the Call Work Area Status Byte after it has successfully placed the call. If the status byte contains an A and this station has data to transmit, go to Write Data step 5. If it contains an A and there is no data to transmit, go to Write Data step 1b.

      Note: An A in the Call Work Area Status Byte does not mean that handshaking and the exchange of IDs has been successfully performed. If there is a handshaking or ID error, the UAP will detect it when it checks status from its Write Data or Read Data logic.

   b. The SCA Module posts an E (error) in the Call Work Area Status Byte if there is an error in dialing (for example, a busy signal or no answer or an invalid phone number). If the status byte contains an E, print an error message and terminate or return to step 2.

      Note: The SCA Module tries eight times to dial the number before it posts an E in the Call Work Area Status Byte. It may be several minutes before the E is posted.

   c. If the status byte still contains a D (ready), then the SCA Module is still trying to dial. Branch and switch to the beginning of step 4.

WRITE DATA

The calling station may transmit data after it has placed a call and again after it has received an EOT from the answering station. The application will determine how many files (if any) the calling station has to send, or the UAP may perform some workstation or other I/O to determine if there is data to send.

If the calling station has more than one logical file to transmit, the files may be sent separately (each terminated by an EOT) or they may be sent together as a single transmission. If they are sent as a single transmission, header records (preceded by an SOH; see step 6d) may be used to identify them or some user-defined code in the records may identify them.

WRITE DATA

1. If the call has just been placed, the calling station can write data or it can send an EOT and give the answering station the option to send data.

   a. To write data, go to step 4.

   b. To send an EOT and give the answering station the option to send data, move a Q (EOT request) to the Write Flag and a D (ready) to the status byte of the next Transmit Buffer. Go to Read Data.

2. If an EOT has just been received by this station's Read Routine (see Inhibit Switch code W in Section 7), this station can write data, or (if it has no data to send) it can give the answering station the option to send more data or it can disconnect the line.

   a. To write data, go to step 4.

   b. To give the answering station the option to send more data, reset the Inhibit Switch to zero and go to Read Data. Since this is not the first transmission of the call, the SCA Module checks for a D (write request) in the Operation Code in the CCB. When it finds that the Inhibit Switch has been cleared but a D has not been posted in the Operation Code, it assumes that the UAP has no data to send. It sends an EOT and waits for a response from the answering station.

   c. To disconnect the line, move an E (disconnect request) to the Operation Code in the CCB and reset the Inhibit Switch. Go to Place Call. The SCA Module will send a DLE EOT, disconnect the line, and wait for another phone number to dial.

   Note: If the UAP is going to reinitialize before placing a new call, do not reset the Inhibit Switch. It will be reset at Initialization step 3.

3. If an EOT has just been received by this station's Handshaking Routine (the answering station had no data to send; see Inhibit Switch code X in Section 7), this station can write data, or (if it has no data to send) it can allow the SCA Module to disconnect the line or it can force the line connection to be maintained.

   a. To write data, go to step 4.

WRITE DATA, Step 3

b. To allow the SCA Module to disconnect the line, reset the Inhibit Switch to zero without posting a code in the Operation Code. Go to Place Call. The SCA Module detects that neither station has data to send. it sends a DLE EOT, disconnects the line, and waits for another phone number to dial.

Note: If the UAP is going to reinitialize before placing a new call, do not reset the Inhibit Switch. It will be reset at Initialization step 3.

c. If for some reason the UAP wants to maintain the connection, even when neither station has data to send (if it expects that one of the stations may have data to send later), it should use a Transmit Buffer to request that an EOT be sent. Move a Q (EOT request) to the Write Flag and a D (write request) to the Operation Code in the CCB and then clear the Inhibit Switch to zero. Go to Read Data. The SCA Module will send the EOT and go to its Read Routine.

4. To start the SCA Module transmitting data, the UAP must perform the following steps:

a. Test the Transmit Buffer Status Bytes. All of them should contain A's. (They should have been initialized to A at assembly time or at Housekeeping step 3.) If a Transmit Buffer Status Byte contains some other character, then the buffer contains data or a control character request (Write Flag code) that was prepared for a previous transmission but not transmitted because of an error. The application will determine what the UAP should do with these buffers.

One possibility is to print or save the contents of the buffers and then clear the buffers by resetting all Write Flags to zero and moving an A (available) to the status bytes of all buffers. By posting the buffers available, the UAP allows the contents of the buffers to be destroyed at step 6. Continue to step 4b.

Another possibility is to continue the transmission from where it was left. Reset to A any Transmit Buffer Status Byte that is other then A or D and continue to step 4b. The UAP must somehow insure that it is now communicating with the station that was originally intended to receive the data.

b. Move a D (write request) to the Operation Code in the CCB. (This is optional if this is the first transmission of the call.) Reset the Inhibit Switch to zero.

WRITE DATA

5. Test the status byte of the next Transmit Buffer. The version of the Get-Buffer Routine selected will determine which buffer to test. If Version 2 or 3 was selected, the UAP may test more than one buffer. (See Sequential Buffer in Section 7.)

   a. If the Transmit Buffer Status Byte = A (available), go to step 6.

   b. If the Transmit Buffer Status is not A, check the status of the SCA Module. It is possible that a serious error has occurred or a special control character has been received. The tests for these conditions and the action the UAP should take are discussed under Check Status below.

      If the buffer has not been transmitted, but the status is normal (no serious error, no special control character), then the SCA Module must still be trying to transmit the data block. Branch and switch to the beginning of step 5 to wait for the SCA Module to complete the transmission or for the status to change.

      If a serious error has occurred, the Check Status logic returns to Place Call.

   c. If the Transmit Buffer Status Byte does not contain an A (available), it should contain a D (ready to be transmitted), an E (error), or a V (being transmitted). If it does not contain any of these values (if it has been overlaid by another partition or by a program error), the Write Routine will be waiting for the buffer to be posted ready and will be sending ENQs and IDLE Messages. The UAP will loop between step 5 and Check Status until the SCA Module posts an R in the Inhibit Switch. To detect such errors, the UAP can test the status byte for a D, E, or V if it does not find an A.

6. If the Transmit Buffer Status Byte = A (available), prepare the buffer with data to be transmitted. (The buffer is available when the Write Data logic has just been entered or reentered or when the buffer has just been successfully transmitted.)

   a. If there is no more data to transmit, go to step 9.

   b. Move data to the Data field of the Transmit Buffer.

   c. Move the data length to the buffer's Data Length field. This step may be omitted if the correct length has already been moved there; the SCA Module does not alter the field.

WRITE DATA, Step 6

d. Move the appropriate code to the Write Flag of the buffer if this buffer is to be preceded by an SOH (Start of Header) or if it is to be terminated by an ETX (End of Text) for some application-determined reason.

e. Move a D (ready) to the Buffer Status Byte.

7. If more than one Transmit Buffer is being used, the UAP must have some pointer (perhaps an index register) to point to the correct Transmit Buffer. Update the pointer to point to the next Transmit Buffer. (See Sequential Buffers in Section 7.

8. Return to step 5.

9. When there is no more data to send, request that an EOT be sent:

a. Move a Q (send EOT request) to the Write Flag of the next Transmit Buffer.

b. Move a D (ready) to the Buffer Status Byte.

c. Go to Read Data. If the SCA Module was assembled without a Read Routine, go to Check Status.

Note: If there are two or more Transmit Buffers, an error in the transmission of one of the last data buffers might cause the SCA Module or the UAP' status checking routine (which will be entered when the UAP returns to Read Data to send a DLE EOT. The SCA Module would not detect the send EOT request just posted in the Write Flag until the next time this station is called. This problem is avoided by clearing the Write Flag to zero (at step 4a) before preparing any Transmit Buffers for transmission.

READ DATA

1. Post all Receive Buffers available. Move an A to the status byte of all Receive Buffers.

2. Test a buffer to see if it has been filled with data by the Read Routine. The version of the Get-Buffer Routine selected will determine which buffer to test. If Version 2 or 3 was selected, the UAP may test more than one buffer. (See Sequential Buffers in Section 7.)

   a. If data has not been received (Buffer Status Byte = A), check the status of the SCA Module. It is possible that a serious error has occurred or an EOT has been received. The tests for these conditions and the actions the UAP should take are discussed under Check Status below.

      If data has not been received and the status is normal (no serious error, no EOT), then the SCA Module must still be trying to read a data block. Branch and switch to the beginning of step 2 to wait for the SCA Module to complete its read.

      If an EOT has been received (Inhibit Switch = X or W), the Check Status logic branches to Write Data.

      If a serious error has occured, the UAP's Check Status logic returns to Place Call.

   b. If data has been received (Buffer Status Byte = E), process it and go to step 3. Processing usually means outputting the data area of the buffer to the line printer or some storage device. Relative locations 5-8 of the buffer prefix will contain the number of data characters received. Processing may mean examining the data in the buffer and making decisions based on the contents. If header records are expected, examine relative location 15 of the buffer to see if the data block is preceded by an SOH instead of the standard STX. If data blocks terminated by an ETX are expected, examine relative location N of the buffer (N = 20 plus the data length) to see if the data block is terminated by an ETX instead of the standard ETB.

   c. If the Receive Buffer Status Byte does not contain an E (full), it should contain an A (available) or V (being filled by the Read Routine). If it does not contain any of these values (if it has been overlaid by another partition or by a program error), the Read Routine will be waiting for the buffer to be posted available and will be sending WACKs. The UAP will loop between step 2 and Check Status indefinitely or until the answering station gives up. To detect such errors, the UAP can test for an A or V in the status byte if it does not find an E.

READ DATA

3. Post the Receive Buffer available. Move an A to the status byte of the buffer just processed.

4. If more than one Receive Buffer is being used, the UAP must have some pointer (perhaps an index register) to point to the correct Receive Buffer. See Sequential Buffers in Section 7. Update the pointer to point to the next Receive Buffer.

5. Return to step 2.

CHECK STATUS

The UAP should check the status of the SCA Module whenever it is waiting for the SCA Module to complete some I/O operation. The purpose of status checking is to detect errors and the occurrence of unusual conditions during the transmission or receipt of data or during handshaking or ID exchange.

1. Test for a non-zero code in the Inhibit Switch. If the Inhibit Switch is zero, go to step 4.

2. If the Inhibit Switch setting is X or W (EOT received), go to step 3. If it is other than X or W, then it represents an error. After posting an error code in the Inhibit Switch the SCA Module disconnects the line and waits for another call.

   a. The UAP may take special action depending on the type of code posted. However, for most applications it will be possible to treat all codes the same.

   b. Print an error message. The error message should show the Inhibit Switch setting and the PSB setting.

   c. Reset the Inhibit Switch to zero. Reset the Partition Status Byte to N in case it was also set.

      Note: If the UAP is going to reinitialize before placing the next call, do not reset the Inhibit Switch. It will be reset at Initialization step 3.

   d. If the station has been transmitting data, there may be untransmitted data and unsatisfied control character requests in the Transmit Buffers after the error has been posted. The application will determine what the UAP should do with these buffers. They can be handled at this step or the next time the UAP enters its Write Data logic. Write Data step 4a discusses methods of handling them.

   e. Return to Place Call.

CHECK STATUS

3. If the Inhibit Switch setting is W, then an EOT (normal end of transmission character) was received by this station's Read Routine. Go to Write Data step 2. If the Inhibit Switch setting is X, then the calling station has no data to send. (It sent an EOT instead of data.) Go to Write Data step 3.

4. If there is no Inhibit Switch setting, test for a code other than N (normal) in the Partition Status Byte (PSB).

    a. If the PSB is set to N, then the status is normal. Return to Read Data step 2 or Write Data step 5 to see if the SCA Module has completed the operation the UAP is waiting for.

    b. If there is a PSB code other than N, the UAP may take special action depending on the code set. All of the PSB codes indicate errors except N, T (TI received), and possibly S (WACKs received). The application will determine the action to be taken.

    c. Before the SCA Module posts an error in the PSB, it has tried eight times to perform the I/O operation. The UAP can allow the SCA Module to continue to try to perform the operation or it can request that the line be disconnected. To request that the line be disconnected, go to step 4d.

       To request that the operation be retried when this station is transmitting data (Activity Switch = D), move a D to the Transmit Buffer Status Byte. (The SCA Module posts an E in the status byte after a transmission error.)

       If the answering station is receiving data (Activity Switch = A), the operation will automatically be retried until it is successful or until the UAP posts a control character request in the Read Flag in the CCB.

       Add one to a counter so that the UAP can "give up" after some number of PSB errors.

       Print an error message and the PSB code.

       Reset the PSB to N.

       Return to the Read Data step 2 or Write Data step 5 to wait for the operation to be retried.

CHECK STATUS, Step 4

d. If the UAP does not want the operation to be retried, or if it has been counting PSB errors and is ready to "give up", it should request that a DLE EOT be sent so that the line will be disconnected.

If the station is reading data (Activity Switch = A), move a 2 (send DLE EOT) to the Read Flag in the CCB. If the station is writing data (Activity Switch = D), move a D (send DLE EOT) to the Write Flag in the buffer prefix and a D (ready) to the Buffer Status Byte. If the SCA Module has returned to its Handshaking Routine (Activity Switch = O), there is no need to send a DLE EOT.

Print an error message and the PSB code.

Reset the PSB to N.

See step 2d and Write Data steps 4a and 9 for a discussion of the problem of untransmitted Transmit Buffers.

e. If Version 2 of the Initialization Routine was selected and the UAP is going to reinitialize before placing a new call, move a numeric code to the Inhibit Switch. The code will be reset at Initialization step 3.

f. Go to Place Call.

DIAL-CALLING USER APPLICATION PROGRAM -- EXAMPLE

INTRODUCTION

The sample user application program (UAP) presented in this section may be run with an SCA Module designed for the calling station of a switched network.

The logic of the sample UAP is shown in an overview flowchart in figure 4-1. The same flowchart is broken into segments in figure 4-2 and explained in the narrative accompanying the figure. A listing of the program appears in Appendix A.

SUMMARY OF FUNCTIONS

The sample UAP gets phone numbers to dial from CALLNO cards. The format of the CALLNO cards is shown under Card Input below. The UAP moves the call number information to the Call Work Area and posts the work area ready so that the SCA Module will place the call. After successfully placing a call, the SCA Module sends its ID "CALLER" and an ENQ to the answering station. It expects the answering station to respond with the ID "HELLO" and an ACKO. The IDs "CALLER" and "HELLO" are passed to the SCA Module via the Initialization Record.

The CALLNO card indicates to the UAP whether it is to write and then read data or read data only. If it is to write first, it reads data cards and moves the card data to the Transmit Buffers to be transmitted. When it reads a unit separator card, it requests that an EOT be sent. It then expects to receive data followed by an EOT or to receive just an EOT from the answering station. It prints the data it receives on the line printer.

If the CALLNO card indicates that the calling station is to read only, the UAP first requests that an EOT be sent and then enters its read data logic.

In either case (write and then read, or read only), after receiving an EOT from the answering station, the UAP requests that a DLE EOT be sent and the line be disconnected, and then it looks for another CALLNO card in the card reader.

CARD INPUT

This UAP reads phone numbers to be dialed and data to be transmitted from cards. The card input conventions described here apply to this UAP only. Another UAP might get phone numbers from the workstation or tape or disc or they might be assembled into the UAP. Data to be transmitted may be read from tape or disc for many applications.

CALLNO Cards

The CALLNO cards read by the sample UAP have the following format:

Columns	Field	Contents	Meaning
1-6	Card ID	CALLNO	CALLNO card identifier.
7	Function Code	R	This station should read data only.
		W	This station should write data first and then read data.
18-19	Length	2 numeric digits	The length of the phone number to be dialed, 01 to 24.
20-43	Phone Number	numeric digits	The phone number to be dialed; maximum 24 digits.

Transmission Deck

The "transmission deck" input to this sample program consists of data cards to be transmitted to the calling station and a unit-separator card. A unit-separator card must be the last card of every transmission deck.

A CALLNO card with function code W (write first) must be followed by a transmission deck. A CALLNO card with function code R (read only) may not be followed by a transmission deck.

DIAL-CALLING USER APPLICATION PROGRAM -- EXAMPLE

SELECTION OF SCAM ROUTINES

This UAP was designed to run with an SCA Module made up of the following routines:

- Handshaking Routine -- Version 6 : Dial-Calling
- Initialization Routine -- Version 1: One-Time Initialization
- Get-Buffer Routine -- Version 1: Sequential
- Error Routine -- Version 2
- Read Routine
- Write Routine

DEFINING COMMON

The following areas in Common will be referenced by both the SCA Module and the UAP. The user can determine their location and initial values by making changes to the distributed Common deck. The Common deck must be included during assembly of the SCA Module and of the UAP. The initial values of Common must be loaded before beginning execution of the SCA Module and the UAP.

- Communications Control Block -- 18 characters beginning at 1000C

  For this example the CCB is defined at location 1000C in the Common deck. It is initialized to all zeros.

- System Mailbox -- 420 characters beginning at 0580C

  The System Mailbox is defined at location 0580C in the distributed Common deck. No changes need be made to the Common deck.

- SCA's Partition Status Byte -- 1 character between 0560C and 0579C

  Both the SCA Module and the sample UAP calculate the address of the SCA's PSB in their housekeeping routines. The UAP initializes it to N (normal) at execution time. No changes need be made to the Common deck.

- Transmit Buffers -- Two 100-character areas, one beginning at location 1500C and one beginning at 1600C; Receive Buffers -- Two 102-character areas, one beginning at location 1700C and one beginning at 1900C

  The Transmit and Receive Buffer definitions may be included in the UAP or added to the Common deck. The SCA Module saves them when it finds the Initialization Record in the System Mailbox.

  Note: A 102-character Receive Buffer may contain up to 80 data characters. A 100-character Transmit Buffer may contain up to 80 data characters. See Transmit Buffer Length and Receive Buffer Length in Chapter 2.

- IDLE Message Buffer -- 44 blanks beginning at location 1050C

  For this example the IDLE Message Buffer is located at location 1050C in the Common deck. It is initialized to all blanks.

- Call Work Area -- 44 positions beginning at location 1050C

  For this example the Call Work Area overlaps the IDLE Message Buffer at location 1050C.

INITIALIZATION

The following information is moved to the System Mailbox by the UAP in order to initialize the SCA Module. The first five characters follow the standard conventions for use of the System Mailbox. The remaining information is the Initialization Record.

System Mailbox Core Location	Number of Bytes	Contents	Meaning
580	1	B	Mailbox busy.
581-582	2	CØ	Mailbox To-Address Field; ID of SCA partition.

System Mailbox Core Location	Number of Bytes	Contents	Meaning
583-584	2	nn	Mailbox From-Address field; UAP partition number.
585-588	4	1500	Address in Common of first Transmit Buffer.
589-592	4	1600	Address in Common of second Transmit Buffer.
593	1	Z	End of list of Transmit Buffer addresses.
594-601	8	unused	Only two Transmit Buffers are being defined.
602-605	4	1700	Address in Common of first Receive Buffer.
606-609	4	1900	Address in Common of second Receive Buffer.
610	1	Z	End of list of Receive Buffer addresses.
611-618	8	unused	Only two Receive Buffers are being defined.
619-622	4	0082	Maximum number of characters to be read. (80 data characters plus one ETB or ETX plus one overflow position.)
623-642	20	unused	Area not used for dial configurations.
643	1	6	Length of this station's ID.
644	1	5	Length of answering stations' IDs.

System Mailbox Core Location	Number of Bytes	Contents	Meaning
645-650	6	CALLER	This station's I AM ID.
651-655	5	HELLO	This station's WRU list. (The ID expected from the answering stations.)
656	1	Z	End of WRU list.

STATUS

The user application main program branches and links to the Status Subroutine

- In its read logic when it finds the Receive Buffer has not been filled by the SCA Module's Read Routine, and
- In its write logic when it finds that a Transmit Buffer that it has posted ready has not been transmitted by the SCA Module's Write Routine.

The UAP's Status Subroutine checks the Partition Status Byte (for a code other than N) and the Inhibit Switch (for a non-zero value) to see if the SCA Module has posted an error or status code. The Status Subroutine posts normal or restart status and then returns to the main program.

Normal Status

The Status Subroutine posts normal status when no codes have been posted in the Inhibit Switch or the PSB. The main program continues to wait for its Receive Buffer to be filled or its Transmit Buffer to be transmitted.

Restart Status

The Status Subroutine posts restart status when an error has occurred or an EOT has been received. When the main program detects restart status, it looks for another CALLNO card in the card reader. When it finds another CALLNO card, it requests that the new phone number be dialed.

PSB Errors

Before the SCA Module posts an error in the PSB, it has tried eight times to perform the I/O instruction. PSB errors are considered temporary, because they can be retried. (See Read Errors and Write Errors under Partition Status Byte in Chapter 2.) In this example the UAP does not allow the SCA Module to retry the operation. It requests that a DLE EOT be sent, it reports the PSB error on the workstation, and it posts restart status.

Inhibit Switch Errors and EOT

The Dial-Calling SCA Module always returns to the Handshaking Routine after posting a non-zero code in the Inhibit Switch. The Inhibit Switch code may indicate a serious error or the receipt of an EOT. Inhibit Switch errors cannot be retried. The Status Subroutine reports the error or the EOT on the work station and posts restart status.

Error Codes

In this example all errors are reported on the work station. A UAP may be designed to take special action depending on the type of error posted in the PSB or Inhibit Switch. See Section 7 of this document and Chapter 2 for the meaning of the codes that may be posted.

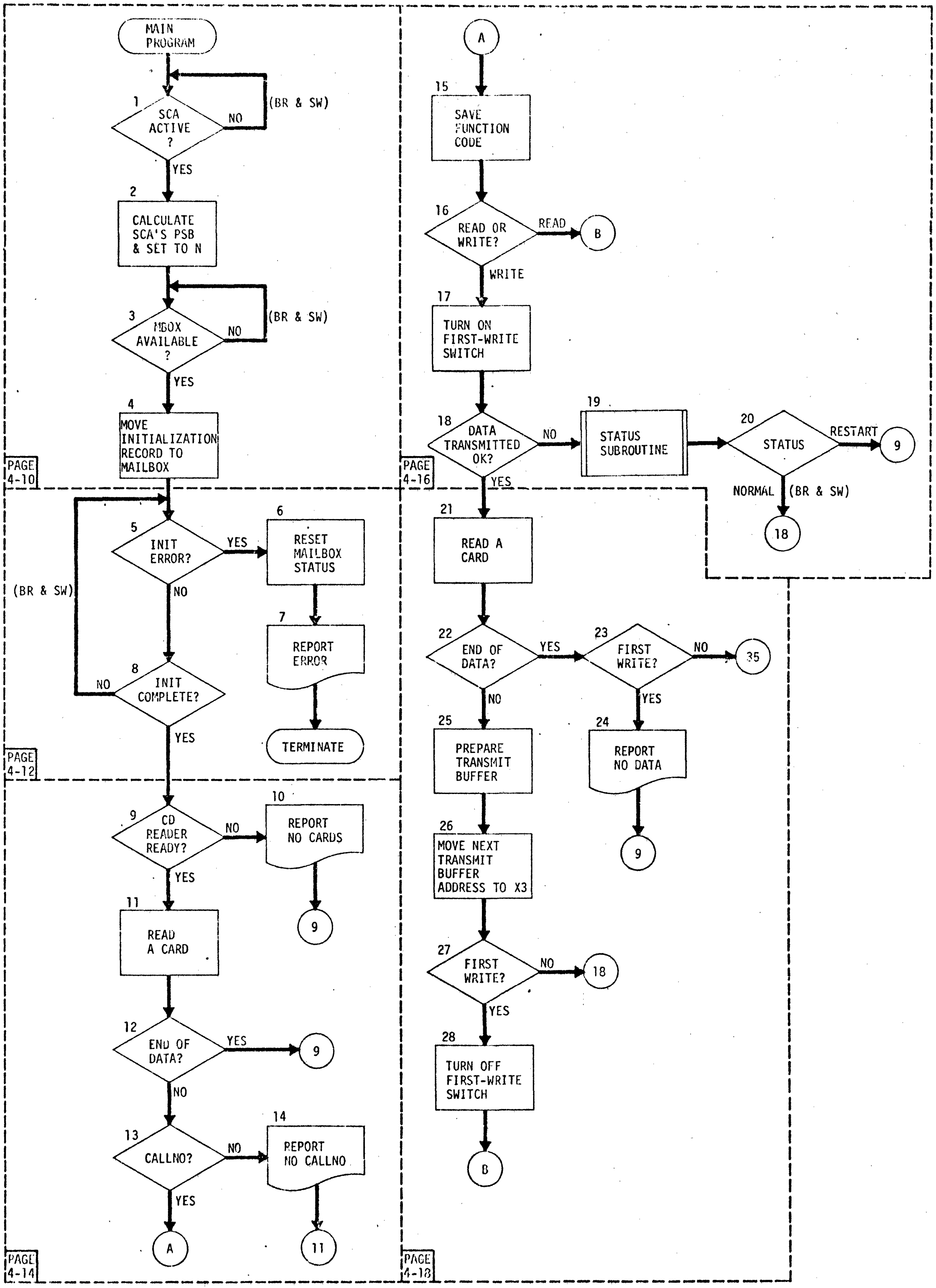


Figure 4-1. Overview of Dial-Calling UAP Example

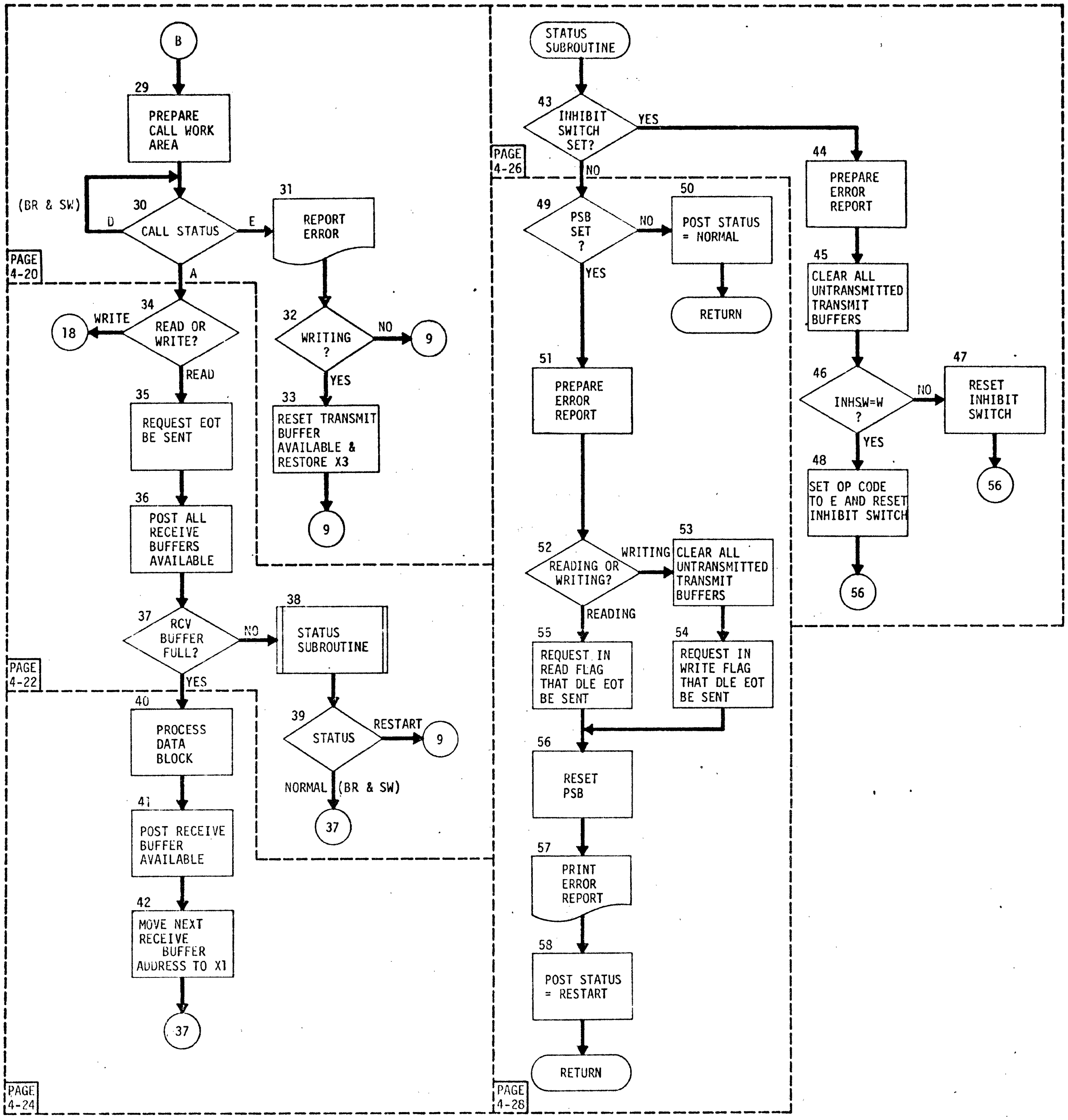


Figure 4-1. Overview of Dial-Calling UAP Example (continued)

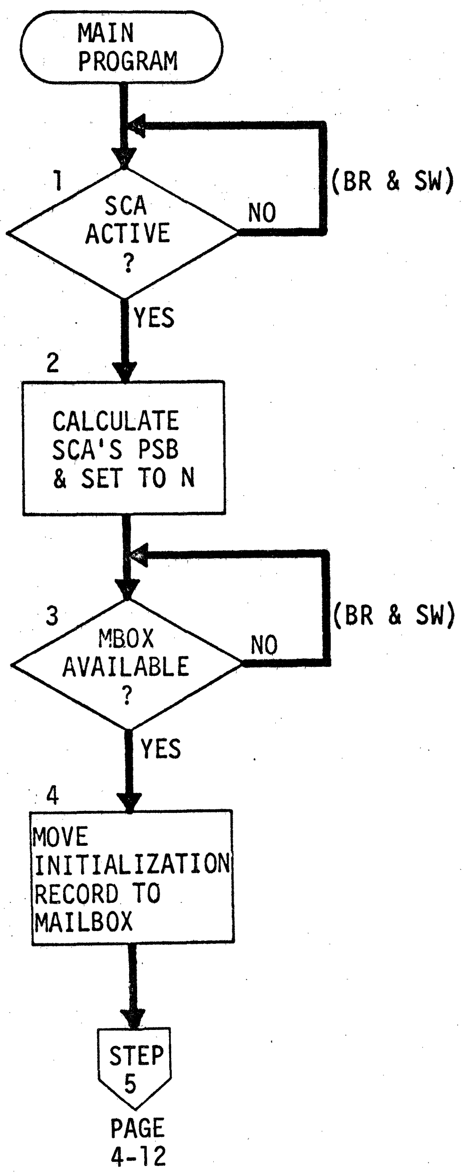


Figure 4-2. Dial-Calling UAP Example (STEPS 1-4)

MAIN PROGRAM

- START

1 IS THE SCA MODULE ACTIVE?

Is ♦ECB+6,7 = a valid partition number for the SCA? When the SCA Module begins execution it stores its partition number in relative locations 6 and 7 of the CCB (♦ECB+6,7). The UAP examines these two positions to see if the SCA Module is loaded and executing.

If YES, go to step 2.
If NO, branch and switch to this test to wait for the SCA Module to become active. A UAP might count the number of times the test has been made so that it can print an error message and terminate after some number.

2 CALCULATE THE SCA's PARTITION STATUS BYTE ADDRESS AND SET IT TO N

Add the SCA's partition number (♦ECB+6,7) to 0560C to get the address of the SCA Module's Partition Status Byte for error checking. Move an N (normal) to the PSB.

3 MAILBOX AVAILABLE?

Mailbox Status Byte = A?
Is the System Mailbox available for use? An A in the status byte (relative position 0) indicates the System Mailbox is available. B,C, or X in the status byte indicates the System Mailbox is being used. If no other partitions are using the mailbox, this test can be omitted.

If YES, go to step 4.
If NO, branch and switch to this test to wait for the Mailbox to become available. A UAP might count the number of times the test has been made so that it can print an error message and terminate after some number.

4 MOVE THE INITIALIZATION RECORD TO THE SYSTEM MAILBOX

Move the Initialization Record into the System Mailbox and post the Mailbox busy with information from the UAP partition for the SCA partition. The information moved to the System Mailbox is shown under Initialization in this section.

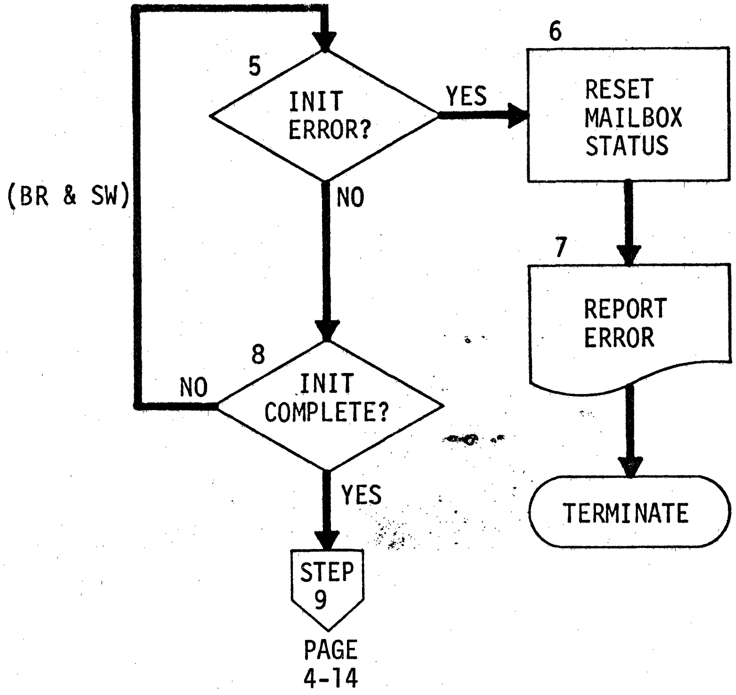


Figure 4-2. Dial-Calling UAP Example (STEPS 5-8)

5 INITIALIZATION ERROR?

Inhibit Switch = E?
Was there an error during initialization?
The SCA Module posts an E in the Inhibit Switch (↑ECB) if there is an error in the Initialization Record.

If NO, go to step 8.
If YES, go to step 6.

6 RESET MAILBOX STATUS

Reset the Mailbox Status Byte to A.

7 REPORT ERROR

Print a message on the workstation indicating that the program is terminating due to an initialization error and then stop processing.

8 INITIALIZATION COMPLETE?

Mailbox Status Byte = A?
Has initialization been successfully completed?
After successful initialization the SCA Module posts the System Mailbox available (A in the Mailbox Status Byte).

If NO, branch and switch to step 5.
If YES, go to step 9.

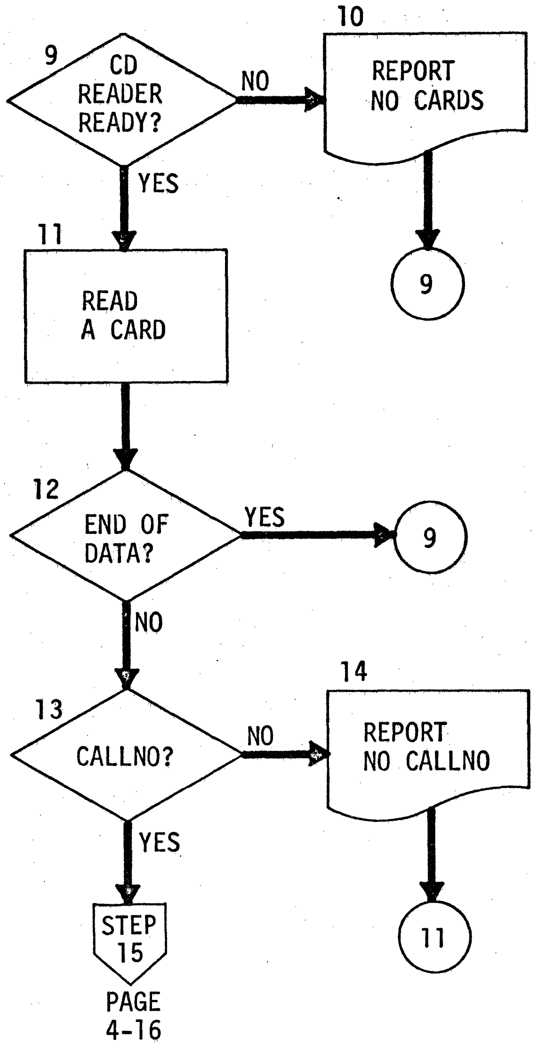


Figure 4-2. Dial-Calling UAP Example (STEPS 9-14)

9 CARD READER READY?

Query the card reader to see if it is ready.

If YES, go to step 11.
If NO, go to step 10.

10 REPORT NO CARDS

Print a message on the workstation indicating that the program is waiting for cards. Return to step 9 to see if the card reader has been readied. The sample UAP prints this message only once each time it enters the loop between steps 9 and 10. It uses a switch to make sure that the message is printed only once.

11 READ A CARD

Read a card. The UAP is looking for a CALLNO card.

12 END OF DATA?

Did the attempt to read data (step 11) result in an end of file condition?

If YES, go to step 9.
If NO, go to step 13.

13 CALL NUMBER?

Was the card just read at step 11 a correctly formatted CALLNO card? See Card Input in this section for the format of the CALLNO cards expected by this sample UAP.

If YES, go to step 15.
If NO, go to step 14.

14 REPORT NO CALLNO CARD

Print a message on the console indicating that the UAP just read a data card or an incorrect CALLNO card when it is expecting a CALLNO card.

Return to step 11 to read another card. The program will loop through steps 11-14 until a correct CALLNO card or a unit-separator card is read. If it empties the card reader without finding a correct CALLNO card or a unit separator, it will wait at step 11 for the card reader to be readied.

The sample program uses a switch to insure that this message is printed only once each time the loop is entered.

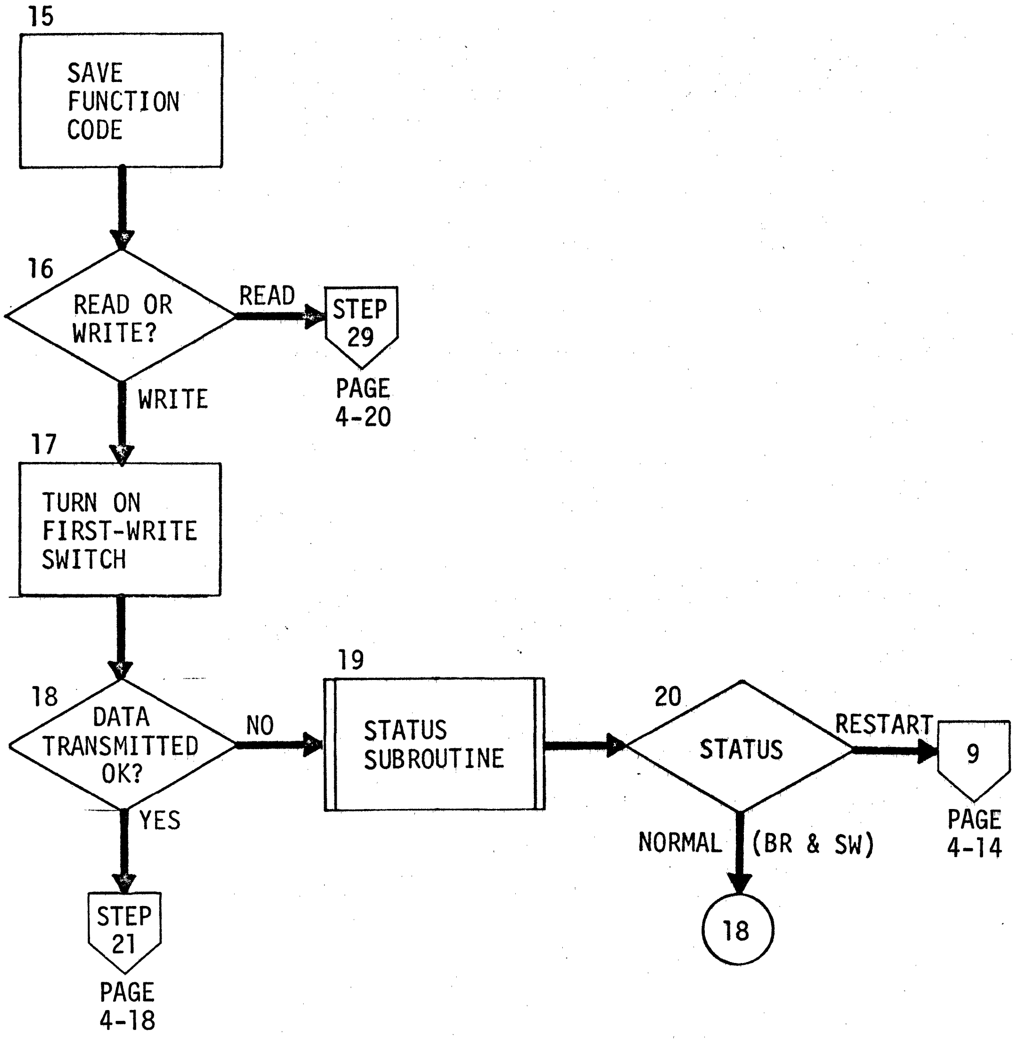


Figure 4-2. Dial-Calling UAP Example (STEPS 15-20)

15 SAVE FUNCTION CODE (W OR R)

The UAP has just read a CALLNO card. Column 7 of the card contains a function code, W or R, to indicate if this station is to write and then read or just read data. This step saves the function code in a work area in the UAP so that it can be tested by step 34.

16 READ OR WRITE?

Function code = R or W?
Is this station to read data only or write and then read data? The function code can be tested in the card input area or in the UAP work area to which it has been moved (step 15).

If this station is to read only, go to step 29.
If it is to write data first, go to step 17.

17 TURN ON FIRST-WRITE SWITCH

Turn on a switch in the UAP. The switch will be tested by steps 23 and 27. When it is on, it indicates that the UAP is preparing the first data block to be transmitted but the phone number has not yet been dialed.

18 DATA TRANSMITTED OK?

Buffer Status Byte = A? Has the buffer been successfuly transmitted? X3 points to the correct Transmit Buffer. (See step 26.)

If YES, go to step 21.
If NO, then the buffer has been readied by the UAP, but it has not yet been successfully transmitted by the SCA Module. Go to step 19.

19 STATUS SUBROUTINE

Branch and link to the Status Subroutine.

20 STATUS

Test the status posted by the Status Subroutine.

Status = normal means that the Write Routine is still trying to transmit the data block. Branch and switch to step 18.
Status = restart means that a serious error has occurred and control has returned to the Handshaking Routine. Go to step 9.

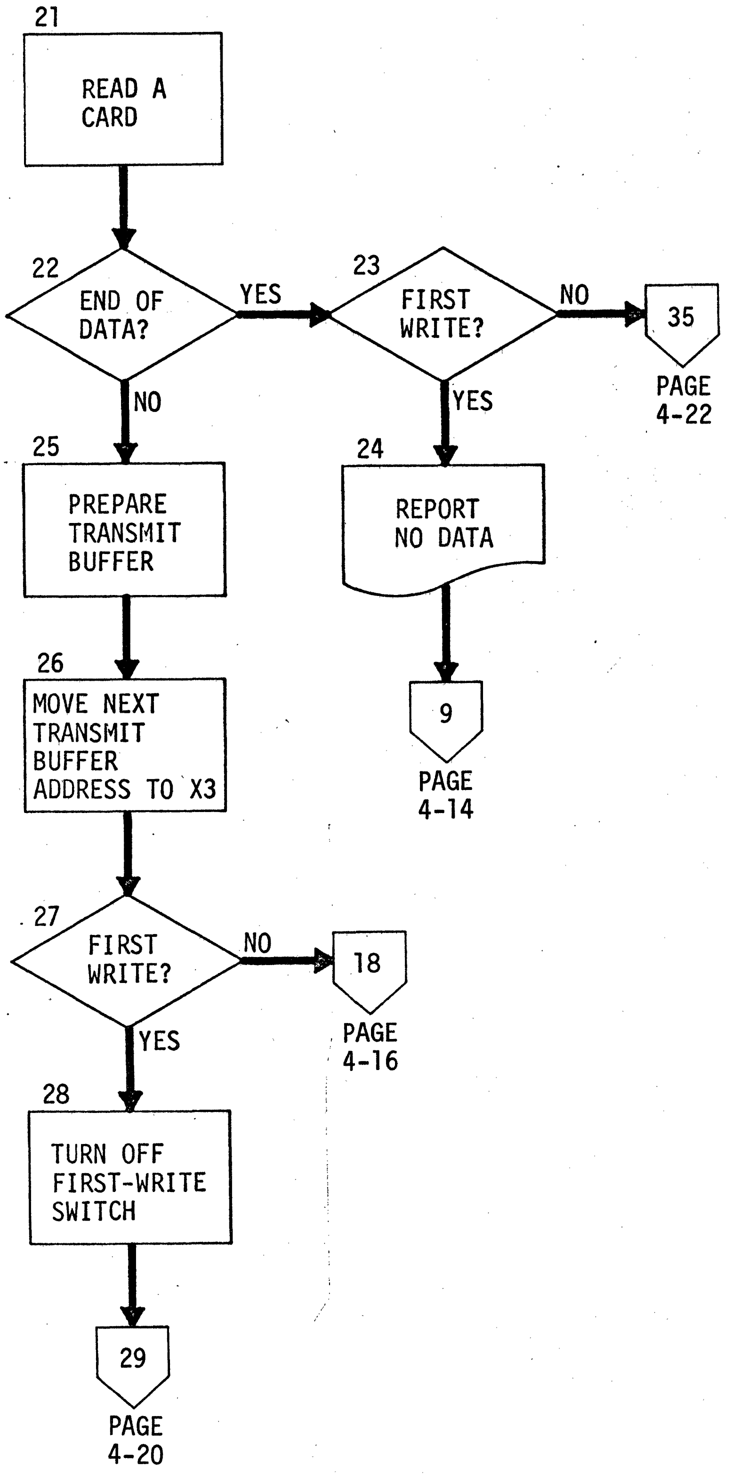


Figure 4-2. Dial-Calling UAP Example (STEPS 21-28)

21 READ A CARD

Read a data card.

22 END OF DATA?

Did the attempt to read data (step 21) result in an end of file condition?

If YES, then there is no more data to transmit. Go to step 23.
If NO, go to step 25.

23 FIRST WRITE?

Is the UAP first-write switch on? (See step 17.)

If YES, go to step 24.
If NO, then the unit-separator card just read at step 22 represents a normal end-of-file for this transmission. Go to step 35.

24 REPORT NO DATA

There has been an error in the card input sequence. A CALLNO card with a function code W (write first) was followed immediately by a unit-separator card instead of data. Report the error. Go to step 9 to look for the next CALLNO card.

25 PREPARE A TRANSMIT BUFFER

Move data (read at step 21) to the Data Field of the Transmit Buffer. Move the data length to the buffer's Data Length field. Move a D (ready) to the Buffer Status Byte. X3 points to the correct Transmit Buffer (See step 26.)

26 MOVE NEXT TRANSMIT BUFFER ADDRESS TO X3

Exchange the Transmit Buffer address work area and X3. Steps 18, 25, 35 and 54 use index register 3 (X3) to address fields in the correct Transmit Buffer. When the UAP is loaded, the address of the first Transmit Buffer is already in X3; the address of the second Transmit Buffer is in a work area in the UAP. The two addresses are exchanged after a Transmit Buffer has been filled.

27 FIRST WRITE?

Is the UAP first-write switch on?

If NO, then the phone number for this transmission has already been dialed. Go to step 18.
If YES, then the phone number has not yet been dialed. Go to step 28.

28 TURN OFF FIRST-WRITE SWITCH

Turn off the UAP first-write switch. Go to step 29 to request that the phone number be dialed.

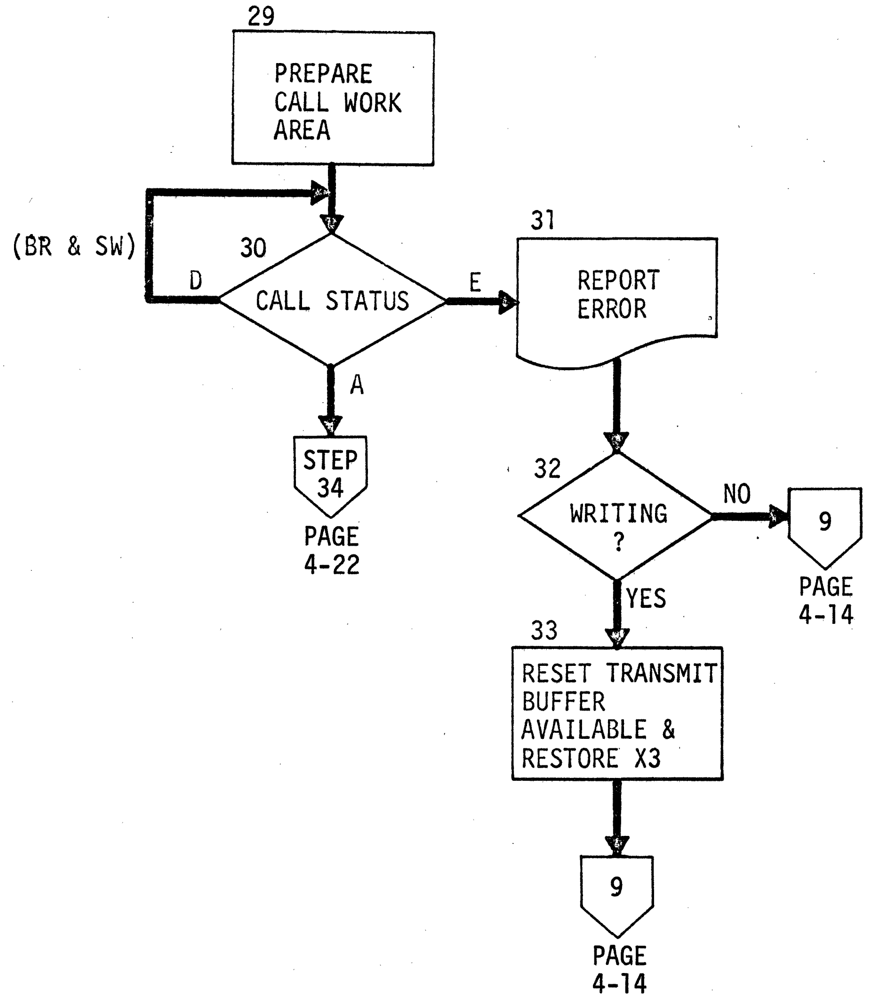


Figure 4-2. Dial-Calling UAP Example (STEPS 29-33)

29 PREPARE CALL WORK AREA

Move the phone number length from the CALLNO card to the Length Field (relative locations 18 and 19) of the Call Work Area. Move the phone number to the Call Work Area beginning in relative location 20. Move a D (ready) to the status byte (relative location 0).

30 CALL STATUS

Test the Call Work Area Status Byte to see the status of the SCA Module's attempt to dial the phone number.

Status byte = A means the phone numner has been successfully dialed. Go to step 34. Note: If there is an error in handshaking or in the exchange of IDs after the call is placed, the error will be detected by the UAP when it enters its Status Subroutine at step 19 or 38.
Status Byte = E means there was an error in the attempt to dial. Go to step 31.
Status Byte = D means that the SCA Module is still trying to place the call. Branch and switch to this step to wait for the SCA Module to complete the attempt.

Note: Program logic errors can easily cause the UAP to stay in this loop. (The status byte remains D because the SCA Module is not dialing; it is waiting for some other action by the UAP.) To help in identifying program logic errors, the UAP can branch and switch to the Status Subroutine before returning to this test. The Status Routine will print diagnostic information that will help the user to debug his program.

31 REPORT ERROR

Print a message on the workstation indicating that there was an error in dialing.

32 WRITING?

Activity Switch (↑ECB+3) = D? Is this station writing data?

If YES, go to step 33.
If NO, go to step 9.

33 RESET TRANSMIT BUFFER AVAILABLE AND RESTORE X3

Because of the dial error, the Transmit Buffer prepared at step 25 will not be sent. Reset the buffer available. Step 26 updated X3 after the first Transmit Buffer was prepared. Perform the operation described by step 26 again so that next time the UAP is preparing to transmit data it will use the same Transmit Buffer.

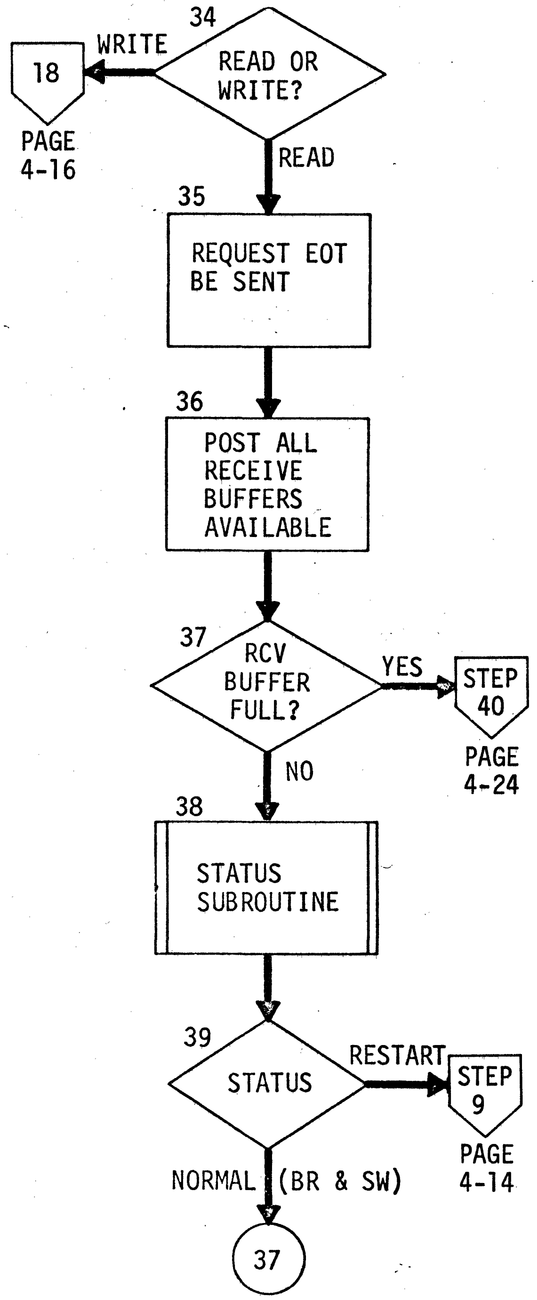


Figure 4-2. Dial-Calling UAP Example (STEPS 34-39)

34 READ OR WRITE?

The call has been successfully made and ID checking (if performed) was successful. Test the function code from the CALLNO card to see if the station is to write before reading or read only. The function code was saved in a work area in the UAP by step 15.

If this station is to read only, go to step 35.
If it is to write first, go to step 18.

35 REQUEST THAT AN EOT BE SENT

The Dial Calling SCA Module expects to write data and an EOT or just an EOT before reading data. To request that an EOT be sent, move a Q (EOT request) to the Write Flag and a D (ready) to the status byte of the next Transmit Buffer. X3 points to the correct Transmit Buffer. (See step 26.)

36 POST ALL RECEIVE BUFFERS AVAILABLE

Move an A (available) to the status byte of both Receive Buffers.

37 RECEIVE BUFFER FULL?

Buffer Status Byte = E? Has the Receive Buffer been filled with data? The Read Routine posts an E in the Buffer Status Byte when the buffer has been filled. X1 points to the correct Receive Buffer. (See step 42).

If NO, go to step 38.
If YES, go to step 40.

38 STATUS SUBROUTINE

Branch and link to the Status Subroutine.

39 STATUS

Test the status posted by the Status Subroutine

Status = normal means the SCA Module is still trying to read the data block. Branch and switch to step 37.

Status = restart means an error has occurred or an EOT has been received and the line has been disconnected. Go to step 9 to look for the next CALLNO card.

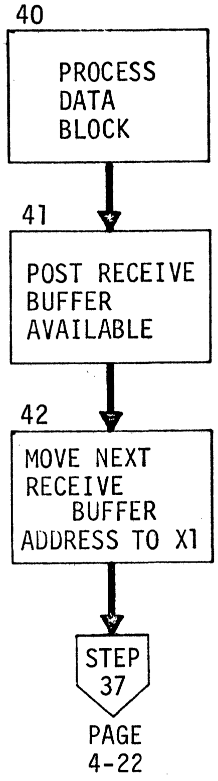

40
PROCESS DATA BLOCK
41
POST RECEIVE BUFFER AVAILABLE
42
MOVE NEXT RECEIVE BUFFER ADDRESS TO X1
STEP 37
PAGE 4-22

Figure 4-2. Dial-Calling UAP Example (STEPS 40-42)

40 PROCESS DATA BLOCK

Process the data block just received. X1 points to the correct Receive Buffer (See step 42.) The Data Length field (relative locations 5-8) of the Receive Buffer Prefix contains the number of data characters received. The sample UAP prints the data on the line printer.

41 POST RECEIVE BUFFER AVAILABLE

Post an A (available) in the Buffer Status Byte (relative position 0) of the Receive Buffer. X1 points to the correct Receive Buffer.

42 MOVE NEXT RECEIVE BUFFER ADDRESS TO X1

Exchange the Receive Buffer address work area and X1. Steps 37, 40 and 41 use index register 1 (X1) to address fields in the correct Receive Buffer. When the UAP is loaded, the address of the first Receive Buffer is already in X1; the address of the second Receive Buffer is in a work area in the UAP. The two addresses are exchanged by this step after a Receive Buffer has been posted available. Return to step 37.

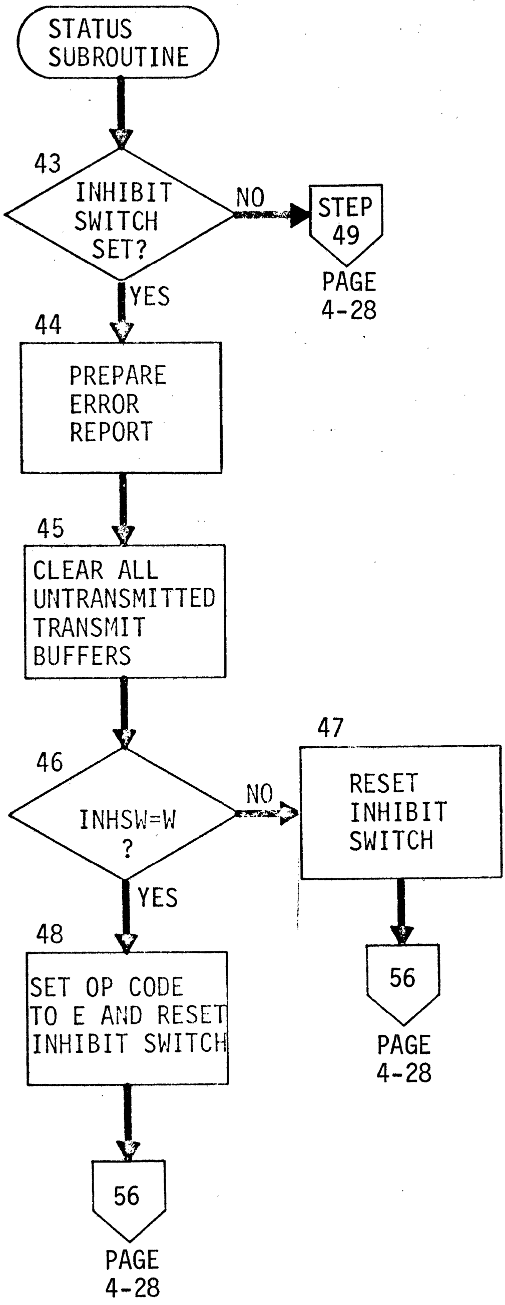


Figure 4-2. Dial-Calling UAP Example (STEPS 43-48)

STATUS SUBROUTINE

43 INHIBIT SWITCH SET?

Inhibit Switch = some code other than zero? Has the SCA Module posted an error or EOT code in the Inhibit Switch?

If YES, go to step 44.
If NO, go to step 49.

44 PREPARE ERROR REPORT

Save the Inhibit Switch setting and the PSB setting so that they can be printed at step 57. They are not printed at this step, because a timeout may occur if the UAP delays in resetting the Inhibit Switch if it was X or W.

45 CLEAR ALL UNTRANSMITTED TRANSMIT BUFFERS

Test the status bytes of both Transmit Buffers. If the status Byte is other than A (available), then the buffer contains data or a control character request prepared by the UAP for this transmission but not transmitted by the SCA Module.

If there is an untransmitted Transmit Buffer, save the contents of the buffer so it can be printed at step 57. Move a zero to the untransmitted buffer's Write Flag and reset its status byte to A so that the buffer will be ready for use the next time the UAP has data to transmit. The printout will be a record of data that was "lost" (not transmitted).

46 INHIBIT SWITCH = W?

Has an EOT just been received by the SCA Module's Read Routine?

If YES, then the UAP must request that the line be disconnected. Go to step 48.
If NO, then the Inhibit Switch setting indicates an error and the line has already been disconnected or it indicates that an EOT was received by the Handshaking Routine and the line will be disconnected when the UAP resets the Inhibit Switch. Go to step 47.

47 RESET INHIBIT SWITCH

Reset the Inhibit Switch to zero. Go to step 56.

48 SET OPERATION CODE TO E AND RESET THE INHIBIT SWITCH

Move an E (disconnect request) to the Operation Code (↑ECB+2) and a 0 to the Inhibit Switch. Go to step 56.

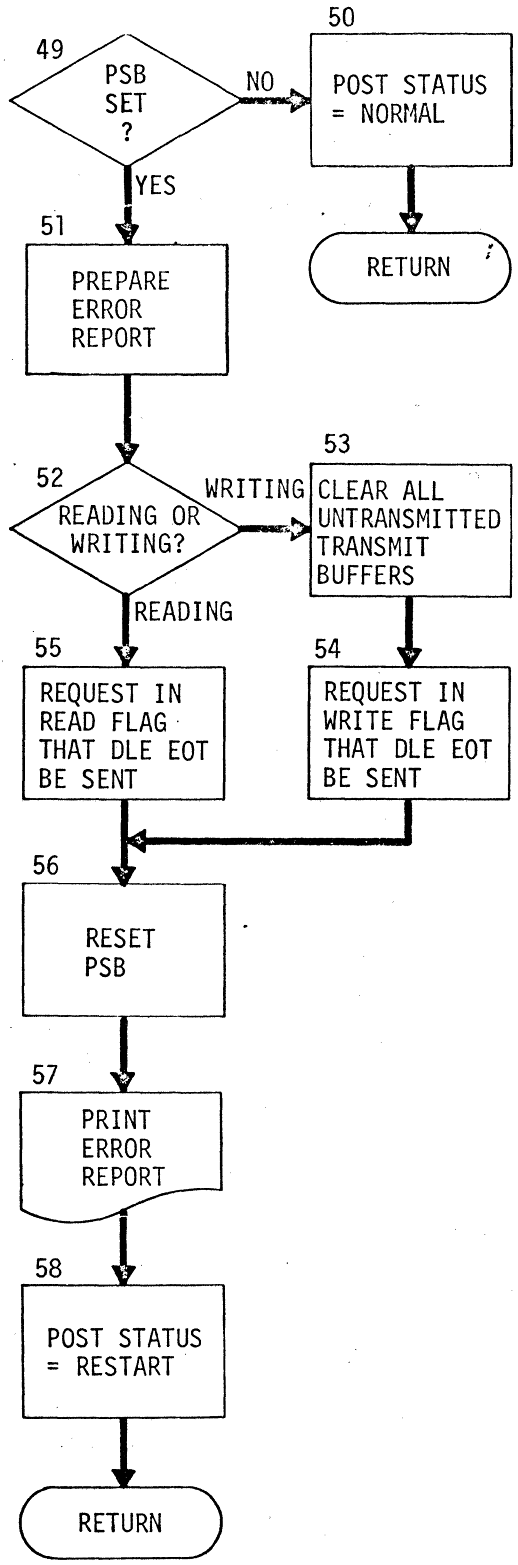


Figure 4-2. Dial-Calling UAP Example (STEPS 49-58)

49 PARTITION STATUS BYTE SET?

PSB = some character other than N? Has The SCA Module posted an error code in the PSB?

If NO, go to step 50.
If YES, go to step 51.

50 POST STATUS = NORMAL

Return to the main program with normal status posted.

51 PREPARE ERROR REPORT

Save the PSB code and the Inhibit Switch code (zero) so that they can be printed at step 57.

52 READING OR WRITING

Test the Activity Switch (↑ECB+3) to see if this station is reading or writing data.

If reading (Activity Switch = A), go to step 55.
If writing (Activity Switch = D), go to step 53.

53 CLEAR ALL UNTRANSMITTED TRANSMIT BUFFERS

Test the status bytes of both Transmit Buffers. If there is an untransmitted Transmit Buffer (status byte = other than A), perform the operations described under step 45.

54 REQUEST IN THE WRITE FLAG THAT A DLE EOT BE SENT

Move a D (DLE EOT request) to the Write Flag of the next Transmit Buffer and post the buffer available. X3 points to the correct Transmit Buffer. (See step 26.) Go to step 56.

55 REQUEST IN THE READ FLAG THAT A DLE EOT BE SENT

Move a 2 (DLE EOT request) to the Read Flag in the CCB (↑ECB+1).

56 RESET PARTITION STATUS BYTE

Reset the PSB to N (normal).

57 PRINT ERROR REPORT

Print the Inhibit Switch Code and the PSB code saved at step 44 or 51. Clear the error code work areas in which the codes were saved. Print any untransmitted Transmit Buffers that were saved at step 45 or 53 and clear those work areas.

58 POST STATUS = RESTART

Return to the main program with restart status posted.

DESIGN OF A DIAL-ANSWERING USER APPLICATION PROGRAM

INTRODUCTION

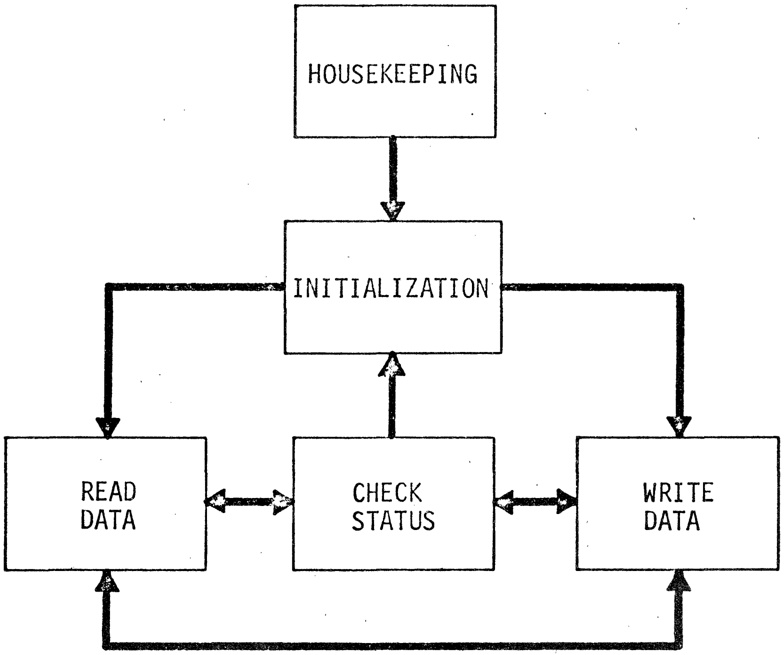
This section is a step-by-step description of the operations to be performed by a user application program (UAP) that is to function with a Dial-Answering SCA Module.

PROGRAM STRUCTURE

In this section the functions of an answering UAP have been divided into five general categories:

- Housekeeping
- Initialization
- Read Data
- Write Data
- Check Status

The following diagram shows the relationship of these functional categories.



Housekeeping is performed once only, immediately after the program is loaded. Initialization is performed after housekeeping. If Version 2 of the Initialization Routine was chosen, the UAP may reinitialize before writing data and when the line has been disconnected. If Version 3 of the Initialization Routine was chosen, the UAP must reinitialize before writing data and when the line has been been disconnected. The UAP enters its Read Data logic whenever it is waiting for a call to come in and after sending an EOT to the calling station. The UAP writes data after it has read data and an EOT or just an EOT from the calling station. It checks status when it is waiting for the SCA Module to transmit or receive data.

The UAP waits for a call to come in by entering its read data logic. The receipt of data or an EOT is the UAP's first indication that a call has come in. A dial-answering UAP must include Read Data steps 1 and 2, and a Dial-Answering SCA Module must contain a Read Routine and at least one Receive Buffer, even for an application in which the answering station never actually receives data from the calling station. However, for an application in which the answering station never sends data to the calling station, the Dial-Answering SCA Module need not contain a Write Routine. The answering UAP can send an EOT instead of data by coding Write Data step 1a. (Write Routine in Section 7 describes programming considerations for a Dial-Answering SCA Module assembled without a Write Routine.)

HOUSEKEEPING

1. Determine that the SCA Module has been loaded and is active.

   Test ↑ECB+6,7 for a valid SCA partition number. If the SCA Module has not been loaded within some time frame (e.g., 10 cycles through the machine), it is advisable to print an error message and terminate.

2. Initialize the SCA Module's PSB to N (normal).

3. Load initial values into other areas of Common if this was not done by defining constants into them at assembly time.

4. Perform any application-determined housekeeping procedures such as initializing work areas, opening files, etc.

In the sample program it is assumed that the SCA's partition number is variable. If the SCA's partition number is constant and known at assembly time, then sample program step 1 can test for a specific partition number in ↑ECB+6,7. Sample program step 3 can be eliminated because the correct PSB can be initialized to N at assembly time.

INITIALIZATION

1. Wait for the System Mailbox to become available. Test the Mailbox Status Byte for an A (available). If it is not an A, branch and switch to this test (wait). If it is an A, continue.

   If no other partitions are using the Mailbox, the constant A should be defined into the Mailbox Status Byte at assembly time or this step should be omitted.

2. Move the Initialization Record to the Mailbox, post the Mailbox busy, and address it to the SCA partition.

   a. Move a B to the Mailbox Status Byte.

   b. Move a CO (alpha O) to the To-Address field of the Mailbox.

   c. Move the UAP's partition number or ID to the From-Address field of the mailbox.

   d. Move the Initialization Record to the Mailbox. Normally all or most of the Initialization Record can be defined as a constant in the UAP source deck. However, for some applications it may be necessary for the UAP to perform some workstation or other I/O to get information to complete the Initialization Record.

3. Reset the Inhibit Switch to zero. (This step is optional if this is the first or only initialization and the Inhibit Switch was initialized to zero at assembly time or at Housekeeping step 3.)

4. Analyze the result of the attempt to initialize the SCA Module.

   a. Test for an initialization error. (Test the Inhibit Switch for an E.) If there has not been an initialization error, continue to step 4b.

      Normally an initialization error will require analysis and perhaps reassembly. Therefore the UAP should print an error message requesting a memory dump, reset the Mailbox Status Byte to A (available), and terminate.

      The SCA Module does not reset the System Mailbox to A after posting an initialization error. This allows the UAP to print the invalid Initialization Record in the Mailbox before allowing other partitions to use the mailbox. Note that the SCA Module may have destroyed part of the contents of the Mailbox while trying to process the Initialization Record.

INITIALIZATION, Step 4

b. Test to see if initialization has been successfully completed.

After successful initialization the SCA Module moves the UAP's partition number (the From-Address field of the System Mailbox) to the Partition that Last Initialized field (↑ECB+4,5) of the CCB and posts the System Mailbox available. The UAP can test either or both of these fields to see if initialization has been successfully completed.

If Initialization has not been successfully completed, return (branch and switch) to step 4a. If initialization has been successfully completed, continue.

Note: the UAP loops through step 4 until initialization has been completed. The UAP should not wait indefinitely for the SCA Module to process the Initialization Record. If after some period (e.g., ten cycles), the SCA Module still has not processed the Initialization Record, the UAP should print an error message, free the Mailbox, and perhaps take a memory dump. It is likely that the SCA partition has not been loaded properly. The user's first attempt to correct the problem should be to reload the SCA partition and try again. If this fails, further problem analysis is required.

READ DATA

The receipt of data or an EOT is the answering UAP's first indication that a call has come in. The UAP enters its Read Data logic when it is waiting for a call to come in and after it has performed its Write Data logic when it is waiting for more data from the same station.

1. Post all Receive Buffers available. Move an A to the status byte of all Receive Buffers.

2. Test a buffer to see if it has been filled with data by the Read Routine. The version of the Get-Buffer Routine selected will determine which buffer to test. If Version 2 or 3 was selected, the UAP may test more than one buffer. (See Sequential Buffers in Section 7.)

   a. If data has not been received (Buffer Status Byte = A), check the status of the SCA Module. It is possible that a serious error has occurred or an EOT or DLE EOT has been received. The tests for these conditions and the actions the UAP should take are discussed under Check Status below.

READ DATA, Step 2a

If data has not been received and the status is normal (no serious error, no EOT, no DLE EOT), then the SCA Module must still be trying to read a data block. Branch and switch to the beginning of step 2 to wait for the SCA Module to complete its read.

If an EOT has been received (Inhibit Switch = X or W), the Check Status logic branches to Write Data.

If a DLE EOT has been received or a serious error has occurred (Inhibit Switch = other than X, W, or O), the SCA Module has disconnected the line. The Check Status logic branches to Initialization or Read Data.

b. If data has been received (Buffer Status Byte = E), process it. Processing usually means outputting the data area of the buffer to the line printer or some storage device. Relative locations 5-8 of the buffer prefix will contain the number of data characters received. Processing may mean examining the data in the buffer and making decisions based on the contents. If header records are expected, examine relative location 15 of the buffer to see if the data block is preceded by an SOH instead of the standard STX. If data blocks terminated by an ETX are expected, examine relative location N of the buffer (N = 20 plus the data length) to see if the data block is terminated by an ETX instead of the standard ETB.

c. If the Receive Buffer Status Byte does not contain an E (full), it should contain an A (available) or V; (being filled by the Read Routine). If it does not contain any of these values (if it has been overlaid by another partition or by a program error), the Read Routine will be waiting for the buffer to be posted available and will be sending WACKs. The UAP will loop between step 3 and Check Status indefinitely or until the calling station gives up. To detect such errors, the UAP can test for an A or V in the status byte if it does not find an E.

3. Post the Receive Buffer available. Move an A to the status byte of the buffer just processed.

4. If more than one Receive Buffer is being used, the UAP must have some pointer (perhaps an index register) to point to the correct Receive Buffer. (See Sequential Buffers in Section 7.) Update the pointer to point to the next Receive Buffer.

5. Return to step 2.

WRITE DATA

The answering station transmits data and an EOT or just an EOT after it has received an EOT from the calling station.

The application will determine how many files (if any) the answering station has to send, or the UAP may perform some workstation or other I/O to determine if there is data to send.

If the answering station has more than one logical file to transmit, the files may be sent separately (each terminated by an EOT) or they may be sent together as a single transmission. If they are sent as a single transmission, header records (preceded by an SOH; see step 4d) may be used to identify them or some user-defined code in the records may identify them.

Note: If the Dial-Answering SCA Module was assembled without a Write Routine or if the calling station is a System Ten with a Dial-Calling SCA Module assembled without a Read Routine, step 1 is the only Write Data step that can be executed.

1. If there is no data to send, the UAP can request that an EOT be sent so that the calling station has the option to send more data, or the UAP can request that a DLE EOT be sent and the line be disconnected.

    a. To send an EOT, reset the Inhibit Switch to zero without posting a D (write request) in the Operation Code. The SCA Module will send an EOT and read a response from the calling station.

       The calling station may respond with an ENQ and data or an EOT (no data to send) or it may disconnect the line or send an invalid response which causes the answering SCA Module to disconnect the line.

       If Version 1 of the Initialization Routine was selected, the UAP can go to Read Data without waiting to see how the calling station responded. Any response other than data will be detected when the Read Data logic checks status.

       If the UAP is to reinitialize when the line has been disconnected or if for some application-determined reason the UAP needs to know if the line has been disconnected, wait for the SCA Module to post a non-zero code in the Inhibit Switch or the Activity Switch.

       If the SCA Module posts an A in the Activity Switch, then the calling station is sending data. Go to Read Data.

       If the SCA Module posts an X in the Inhibit Switch, then the calling station sent an EOT instead of data. Go to Write Data .

WRITE DATA, Step 1a

If the SCA Module posts any other code in the Inhibit Switch, then the line has been disconnected. Print an error message. Go to Initialization if reinitialization is required. Otherwise clear the Inhibit Switch and go to Read Data.

b. To send a DLE EOT and disconnect the line, move an E (disconnect request) to the Operation Code. Go to Initialization if reinitialization is required. Otherwise go to Read Data.

2. To start the SCA Module transmitting data, the UAP must perform the following steps:

   a. Test the Transmit Buffer Status Bytes. All of them should contain As. (They should be initialized to A at assembly time or at Housekeeping step 3.) If a Transmit Buffer Status Byte contains some other character, then the buffer contains data or a control character request that was prepared but not transmitted the last time the station was called. The application will determine what the UAP should do with these buffers.

      One possibility is to print or save the contents of the buffers and then clear the buffers by resetting all Write Flags to zero and moving an A (available) to the status bytes of all buffers. By posting the buffers available, the UAP allows the contents of the buffers to be destroyed at step 4. Continue to step 2b.

      Another possibility is to continue the transmission from where it was left. Reset to A any Transmit Buffer Status Byte that is other than A or D and continue to step 2b. The UAP must somehow insure that it is now communicating with the station that was originally intended to receive the data.

   b. Move a D (write request) to the Operation Code in the CCB.

   c. If reinitialization is required, perform Initialization.

   d. Reset the Inhibit Switch to zero. (This step is unnecessary if the UAP has just reinitialized at step 2c. The Initialization logic resets the Inhibit Switch.)

WRITE DATA

3. Test the status byte of the next Transmit Buffer. The version of the Get-Buffer Routine selected will determine which buffer to test. If Version 2 or 3 was selected, the UAP may test more than one buffer. (See Sequential Buffers in Section 7.)

   a. If the Transmit Buffer Status Byte = A (available), go to step 4.

   b. If the Transmit Buffer Status Byte is not A, check the status of the SCA Module. It is possible that a serious error has occurred or a special control character has been received. The test for these conditions and the action the UAP should take are discussed under Check Status below.

      If the buffer has not been transmitted, but the status is normal (no serious error, no special control character), then the SCA Module must still be trying to transmit the data block. Branch and switch to the beginning of step 3 to wait for the SCA Module to complete the transmission or for the status to change.

      If a serious error has occurrred, the Check Status logic returns to Read Data to wait for a new call to come in.

   c. If the Transmit Buffer Status Byte does not contain an A (available), it should contain a D (ready to be transmitted), an E (error), or a V (being transmitted). If it does not contain any of these values (if it has been overlaid by another partition or by a program error), the Write Routine will be waiting for the buffer to be posted ready and will be sending ENQs and IDLE Messages. The UAP will loop between step 3 and Check Status until the SCA Module posts an R in the Inhibit Switch. To detect such errors, the UAP can test the status byte for a D, E, or V if it does not find an A.

4. If the Transmit Buffer Status Byte = A (available), prepare the buffer with data to be transmitted. (The buffer is available when the Write Data logic has just been entered or reentered or when the buffer has just been successfully transmitted.)

   a. If there is no more data to transmit, go to step 6.

   b. Move data to the Data field of the Transmit Buffer.

   c. Move the data length to the buffer's Data Length field. This step may be omitted if the correct length has already been moved there; the SCA Module does not alter the field.

WRITE DATA, Step 4

d. Move the appropriate code to the Write Flag of the buffer if this buffer is to be preceded by an SOH (Start of Header) or if it is to be terminated by an ETX (End of Text) for some application-determined reason.

e. Move a D (ready) to the Buffer Status Byte.

5. If more than one Transmit Buffer is being used, the UAP must have some pointer (perhaps an index register) to point to the correct Transmit Buffer. Update the pointer to point to the next Transmit Buffer. (See Sequential Buffers in Section 7.) Return to step 3.

6. When there is no more data to send, request that an EOT be sent:

   a. Move a Q (send EOT request) to the Write Flag of the next Transmit Buffer.

   b. Move a D (ready) to the Buffer Status Byte.

The SCA Module will send an EOT and read a response from the calling station. The calling station may respond with an ENQ and data or an EOT (no data to send) or it may disconnect the line or send an invalid response which causes the answering SCA Module to disconnect the line.

If Version 1 of the Initialization Routine was selected, the UAP can go to Read Data without waiting to see how the calling station responded. Any response other than data will be detected when the Read Data logic checks status.

If the UAP is to reinitialize when the line has been disconnected or if for some application-determined reason the UAP needs to know if the line has been disconnected, wait for the SCA Module to post a non-zero code in the Inhibit Switch or the Activity switch.

If the SCA Module posts an A in the Activity Switch, then the calling station is sending data. Go to Read Data.

If the SCA Module posts an X in the Inhibit Switch, then the calling station sent an EOT instead of data. Return to Write Data step 1.

If the SCA Module posts any other code in the Inhibit Switch, then the line has been disconnected. Print an error message. Go to Initialization if reinitialization is required. Otherwise clear the Inhibit Switch and go to Read Data.

WRITE DATA, Step 6

Note: If there are two or more Transmit Buffers, an error in the transmission of one of the last data buffers might cause the SCA Module or the UAP's status checking routine (which will be entered when the UAP returns to Read Data) to send a DLE EOT. The SCA Module would not detect the send EOT request just posted in the Write Flag until the next time this station is called. This problem is avoided by clearing the Write Flag to zero (at step 2a) before preparing any Transmit Buffers for transmission.

CHECK STATUS

The UAP should check the status of the SCA Module whenever it is waiting for the SCA Module to complete some I/O operation. The purpose of status checking is to detect errors and the occurrence of unusual conditions during the transmission or receipt of data or during handshaking or ID exchange.

1. Test for a non-zero code in the Inhibit Switch. If the Inhibit Switch is zero, go to step 4.

2. If the Inhibit Switch setting is X or W (EOT received), go to step 3. If it is other than X or W, then it represents an error. After posting an error code in the Inhibit Switch the SCA Module disconnects the line and waits for another call.

   a. The UAP may take special action depending on the type of code posted. However, for most applications it will be possible to treat all codes the same.

   b. Print an error message. The error message should show the Inhibit Switch setting and the PSB setting.

   c. Reset the Inhibit Switch to zero. Reset the Partition Status Byte to N in case it was also set.

      Note: If the UAP is going to reinitialize before placing the next call, do not reset the Inhibit Switch. It will be reset at Initialization step 3.

   d. If the station has been transmitting data, there may be untransmitted data and unsatisfied control character requests in the Transmit Buffers after the error has been posted. The application will determine what the UAP should do with these buffers. They can be handled at this step or after the station has been called again. Write Data step 2a discusses methods of handling them.

   e. Return to Read Data to wait for another call to come in.

CHECK STATUS

3. If the Inhibit Switch setting is X, then the calling station has no data to send. (It sent an EOT instead of data.) If the Inhibit Switch setting is W, then an EOT (normal end of transmission character) was received while reading. In either case, go to Write Data to see if this station has data to send.

4. If there is no Inhibit Switch setting, test for an error code (any code other than N) in the Partition Status Byte (PSB).

   a. If the PSB is set to N, then the status is normal. Return to Read Data step 2 or Write Data step 3 to see if the SCA Module has completed the operation the UAP is waiting for.

   b. If there is a PSB code other than N, the UAP may take special action depending on the code set. All of the PSB codes indicate errors except N, T (TI received), and possibly S (WACKs received). The application will determine the action to be taken.

   c. Before the SCA Module posts an error in the PSB, it has tried eight times to perform the I/O operation. The UAP can allow the SCA Module to continue to try to perform the operation or it can request that the line be disconnected. To request that the line be disconnected, go to step 4d.

      To request that the operation be retried when this station is transmitting data (Activity Switch = D), move a D to the Transmit Buffer Status Byte. (The SCA Module posts an E in the status byte after a transmission error.)

      If the answering station is receiving data (Activity Switch = A) the operation will automatically be retried until it is successful or until the UAP posts a control charater request in the Read Flag in the CCB.

      Add one to a counter so that the UAP can "give up" after some number of PSB errors.

      Print an error message and the PSB code.

      Reset the PSB to N.

      Return to the Read Data step 2 or Write Data step 3 to wait for the operation to be retried.

CHECK STATUS, Step 4

d. If the UAP does not want the operation to be retried, or if it has been counting PSB errors and is ready to "give up", it should request that a DLE EOT be sent so that the line will be disconnected.

If the station is reading data (Activity Switch = A), move a 2 (send DLE EOT) to the Read Flag in the CCB. If the station is writing data (Activity Switch = D), move a D (send DLE EOT) to the Write Flag in the buffer prefix and a D (ready) to the Buffer Status Byte. If the SCA Module has returned to its Handshaking Routine (Activity Switch = 0), there is no need to send a DLE EOT.

Print an error message and the PSB code.

Reset the PSB to N.

See step 2d and Write Data steps 2a and 6 for a discussion of the problem of untransmitted Transmit Buffers.

e. If Version 2 of the Initialization Routine was selected and the UAP is going to reinitialize before receiving a new call, move a numeric code to the Inhibit Switch. The code will be reset at Initialization step 3.

f. If reinitialization is necessary, go to Initialization. Otherwise go to Read Data to wait for a new call to come in.

DIAL-ANSWERING USER APPLICATION PROGRAM -- EXAMPLE

INTRODUCTION

The sample user application program (UAP) presented in this section may be run with an SCA Module designed for the answering station of a switched network.

The logic of the sample UAP is shown in an overview flowchart in figure 6-1. The same flowchart is broken into segments in figure 6-2 and is explained in the narrative accompanying the figure. A listing of the program appears in Appendix B.

SUMMARY OF FUNCTIONS

The sample UAP expects to receive calls from one station only. It expects the calling station to send the ID "CALLER" immediately after dialing. The SCA Module will then send this station's ID, "HELLØ". It expects to receive a data file followed by an EOT from the calling station and then to transmit a data file followed by an EOT. It prints the data it receives. It gets from cards data to transmit. If either station has nothing to send, it can simply send an EØT.

When this station has sent an EØT or has detected an error, the UAP returns to the beginning of its read logic, and waits for a data block or an EØT from the other station. If the line has been disconnected, a new call must come in before the UAP will find that a data block or an EØT has been received. When a new call comes in, the SCA Module automatically performs handshaking and exchanges IDs -- no special UAP action is required. In this example the UAP doesn't care if a new call has come in or if it is continuing transmission via a previous connection.

SELECTION OF SCAM ROUTINES

This UAP was designed to run with an SCA Module made up of the following routines:

- Handshaking Routine -- Version 5: Dial-Answering
- Initialization Routine -- Version 1: One-Time Initialization
- Get-Buffer Routine -- Version 1: Sequential
- Error Routine -- Version 2
- Read Routine
- Write Routine

DEFINING COMMON

The following areas in Common will be referenced by both the SCA Module and the UAP. The user can determine their location and initial values by making changes to the distributed Common deck. The Common deck must be included during assembly of the SCA Module and of the UAP. The initial values of Common must be loaded before beginning execution of the SCA Module and the UAP.

- Communications Control Block -- 18 bytes beginning at 1000C

  For this example the CCB is defined at location 1000C in the Common deck. It is initialized to all zeros.

- System Mailbox -- 420 bytes beginning at 0580C

  The System Mailbox is automatically located at 0580C in the Common deck. In this example the Mailbox Status Byte (location 0580C) is initialized to A (available). (In this example the SCA Module and its UAP are the only programs in core that use the System Mailbox.)

- SCA's Partition Status Byte -- 1 byte between 0560C and 0579C

  Both the SCA Module and the sample UAP calculate the address of the SCA's PSB in their housekeeping routines. The UAP initializes it to N (normal) at execution time. No changes need be made to the Common deck.

- Transmit Buffer -- 100 bytes beginning at location 1500C and Receive Buffer -- 102 bytes beginning at location 1600C

  The Transmit and Receive Buffer definitions may be included in the UAP or added to the Common deck. The SCA Module defines them when it reads the Initialization Record.

  Note: A 102-character Receive Buffer may contain up to 80 data characters. A 100-character Transmit Buffer may contain up to 80 data characters. See Transmit Buffer Length and Receive Buffer Length in Chapter 2.

- IDLE Message Buffer -- 44 blanks beginning at location 1018C

  For this example the IDLE Message Buffer is located at location 1018C in the Common deck. It is initialized to all blanks.

DIAL-ANSWERING USER APPLICATION PROGRAM -- EXAMPLE

INITIALIZATION

The following information is moved to the System Mailbox by the UAP in order to initialize the SCA Module. The first five characters follow the standard conventions for use of the System Mailbox. The remaining information is the Initialization Record.

System Mailbox Core Location	Number of Bytes	Contents	Meaning
580	1	B	Mailbox busy.
581 - 582	2	CØ	Mailbox To-Address field. ID of SCA partition.
583 - 584	2	nn	Mailbox From-Address field. UAP partition number.
585 - 588	4	1500	Address in Common of Transmit Buffer.
589	1	Z	End of list of Transmit Buffer Addresses.
590 - 601	12	unused	Only one Transmit Buffer is being defined.
602 - 605	4	1600	Address in Common of the Receive Buffer.
606	1	Z	End of list of Receive Buffer addresses.
607 - 618	12	unused	Only one Receive Buffer is being defined.
619 - 622	4	0082	Maximum number of characters to be read. (80 data characters plus one ETB or ETX plus one overflow character.)
623 - 642	20	unused	Area not used for dial configurations.
643	1	5	Length of this station's ID.
644	1	6	Length of calling station's ID.

System Mailbox Core Location	Number of Bytes	Contents	Meaning
645 - 649	5	HELLØ	This station's I AM ID.
650	1	unused	The I AM ID selected does not use all of the six positions available.
651 - 656 x	6	CALLER	This station's WRU list. (The ID expected from the calling station.)
657	1	Z	End of WRU list. This station will accept a call only from a station that sends "CALLER" as its ID after calling.

STATUS

The user application main program branches and links to the Status Subroutine

- In its read logic when it finds the Receive Buffer has not been filled by the SCA Module's Read Routine,
- And in its write logic when it finds that a Transmit Buffer that it has posted ready has not been transmitted by the SCA Module's Write Routine.

The UAP's Status Subroutine checks the Partition Status Byte (for a code other than N) and the Inhibit Switch (for a non-zero value) to see if the SCA Module has posted an error condition or the receipt of an EØT. The Status Subroutine posts normal, error, or EØT status and then returns to the main program. The main program branches to a location determined by the status.

Normal Status

The Status Subroutine posts normal status when no codes have been posted in the Inhibit Switch or the PSB. The main program continues to wait for its Receive Buffer to be filled or its Transmit Buffer to be transmitted.

Error Status

The Status Subroutine posts error status if it finds an error code in the PSB or the Inhibit Switch.

PSB Errors

Before the SCA Module posts an error in the PSB, it has tried eight times to perform the I/O instruction. PSB errors are considered temporary, because they can be retried. (See Read Errors and Write Errors under Partition Status Byte in Chapter 2.) However, in this example the UAP does not retry. When the Status Subroutine finds an error code in the PSB, it requests that a DLE EØT be sent, it reports the error on the work station, and it posts error status. The main program returns to the beginning of its read logic.

Inhibit Switch Errors

The SCA Module always disconnects the line after posting an error in the Inhibit Switch. (Note: The receipt of an EØT while reading or during handshaking is posted in the Inhibit Switch but is not considered an error and does not cause the line to be disconnected.) Inhibit Switch errors cannot be retried. When the Status Subroutine finds an Inhibit Switch error it posts error status and it reports the error on the work station. The main program returns to the beginning of its read logic to wait for a new call to come in.

Error Codes

In this example all errors are reported on the work station. A UAP may be designed to take special action depending on the type of error posted in the PSB or Inhibit Switch. See Chapter 2 for the meaning of the codes that may be posted.

EØT Status

The receipt of an EØT while the SCA Module is performing handshaking or reading data means that the calling station is ready to receive from this station. The SCA Module posts a W (EØT while reading) or X (EØT during handshaking) in the Inhibit Switch. When the Status Subroutine finds one of these codes it posts EØT status and the main program branches to its write logic. Note: If the SCA Module receives an EØT while writing, it posts a Q in the Inhibit Switch, which is considered an error by the Status Subroutine.

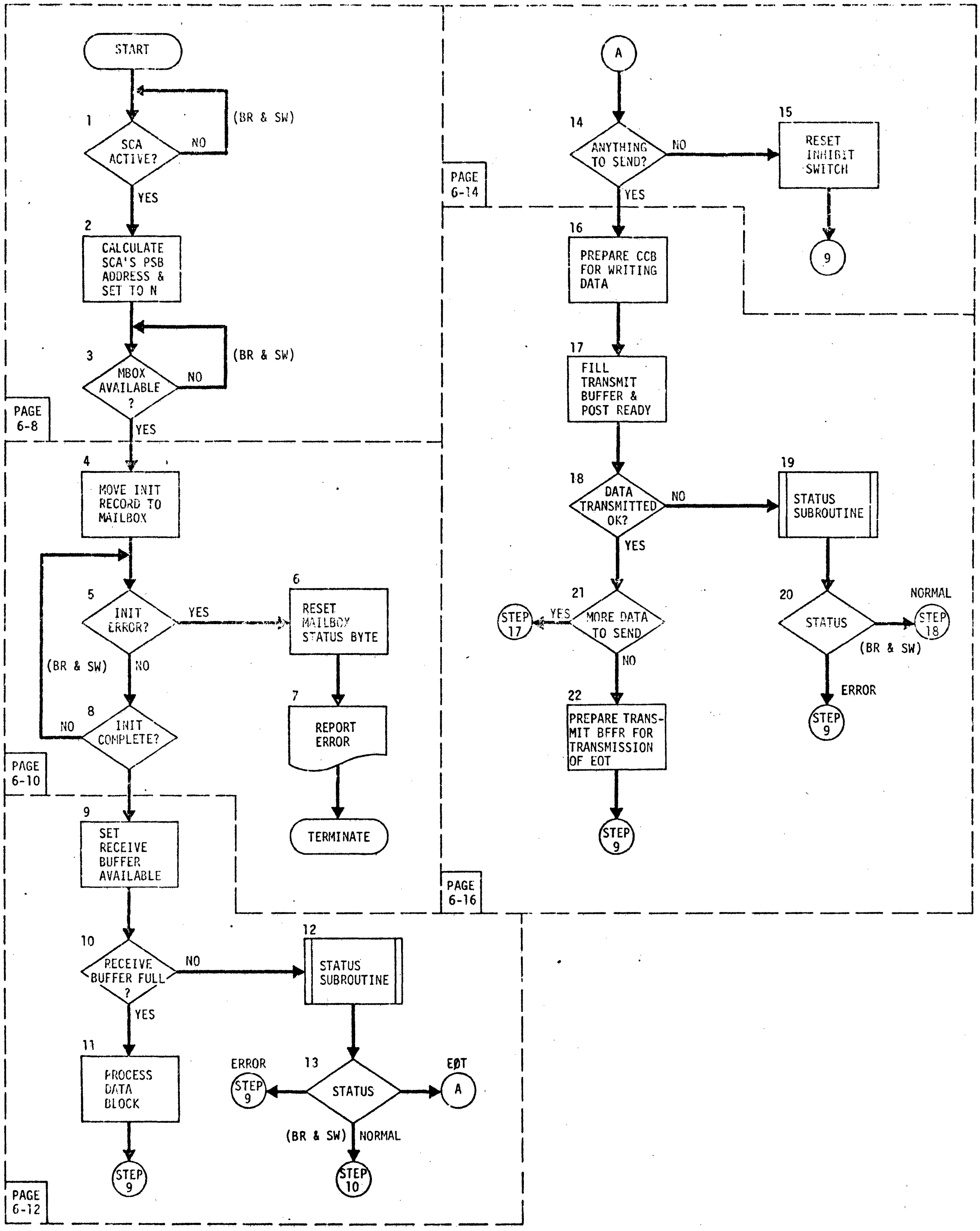


Figure 6-1. UAP Main Program

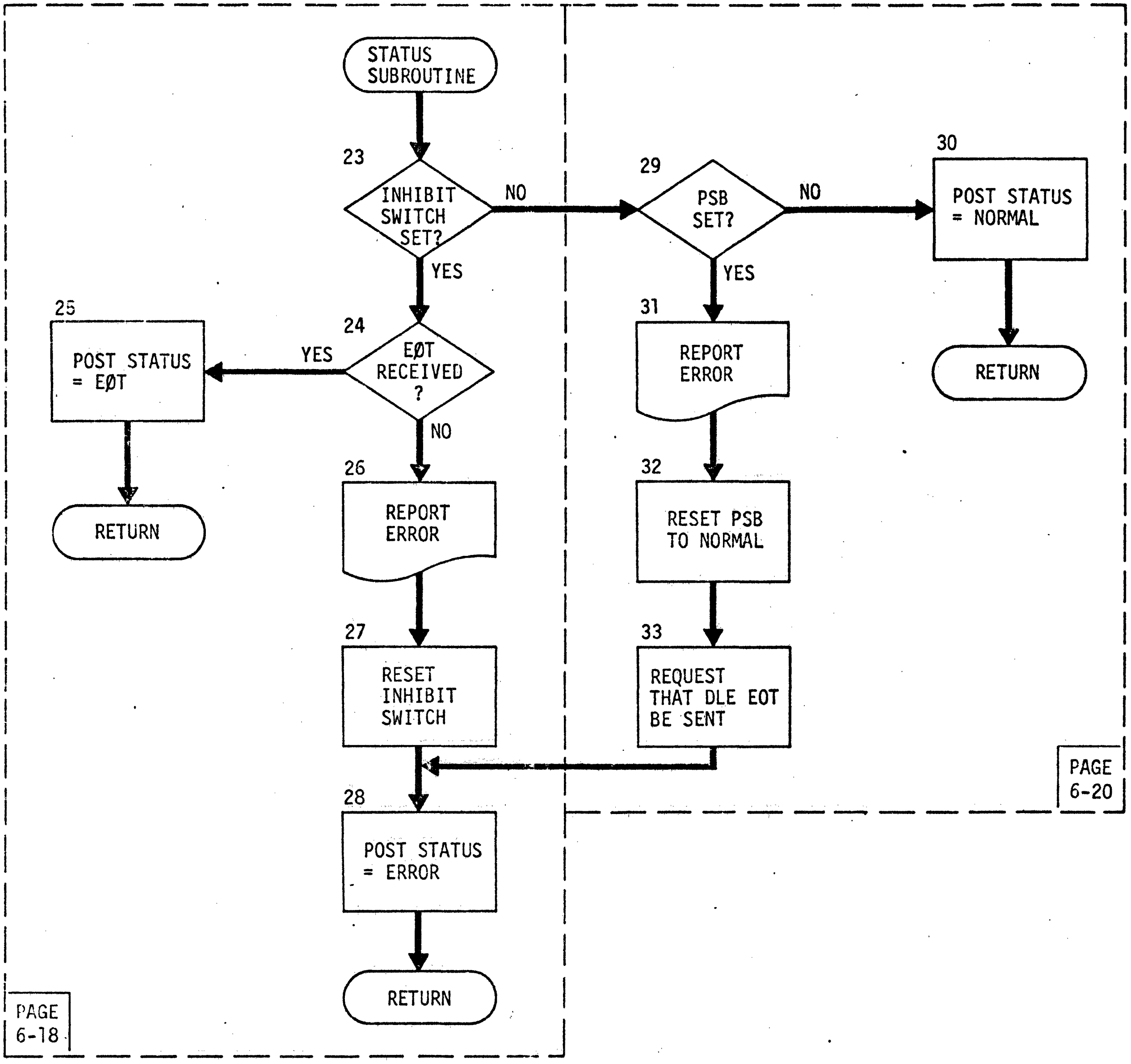


Figure 6-1. UAP Main Program (continued)

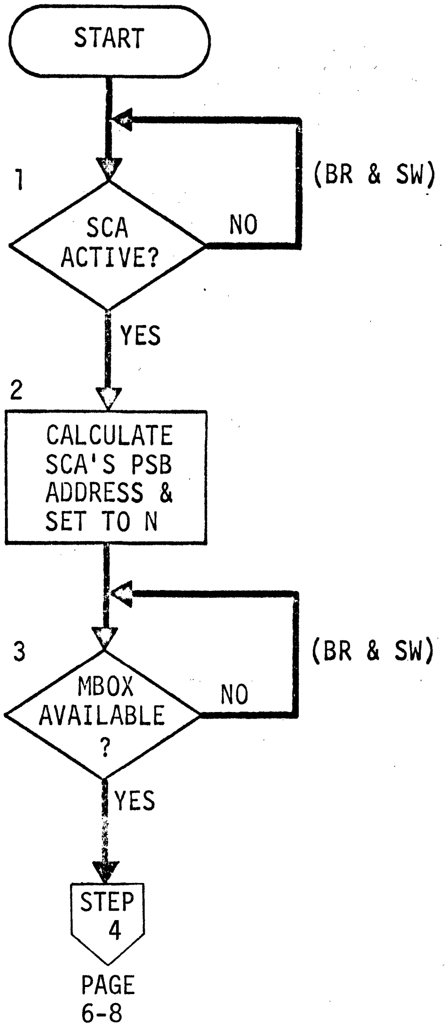


Figure 6-2. UAP Main Program (STEPS 1-3)

DIAL-ANSWERING USER APPLICATION PROGRAM -- EXAMPLE

MAIN PROGRAM

. START

1 IS THE SCA MODULE ACTIVE?

Is ↑ECB+6,7 = a valid partition number for the SCA? When the SCA Module begins execution it stores its partition number in relative locations 6 and 7 of the CCB (↑ECB+6,7). The UAP examines these two positions to see if the SCA Module is loaded and executing.

If YES, go to step 2.
If NO, branch and switch to this test to wait for the SCA Module to become active. A UAP might count the number of times the test has been made so that it can print an error message and terminate after some number.

2 CALCULATE THE SCA'S PARTITION STATUS BYTE ADDRESS AND SET IT TO N

Add the SCA's partition number (↑ECB+6,7) to 0560C to get the address of the SCA Module's Partition Status Byte for error checking. Move an N (normal) to the PSB.

3 MAILBOX AVAILABLE?

Mailbox Status Byte = A? Is the System Mailbox available for use? An A in the status byte (relative position 0) indicates the System Mailbox is available. B,C, or X in the status byte indicates the System Mailbox is being used. If no other partitions are using the mailbox, this test can be omitted.

If YES, go to step 4.
If NO, branch and switch to this test to wait for the Mailbox to become available. A UAP might count the number of times the test has been made so that it can print an error message and terminate after some number.

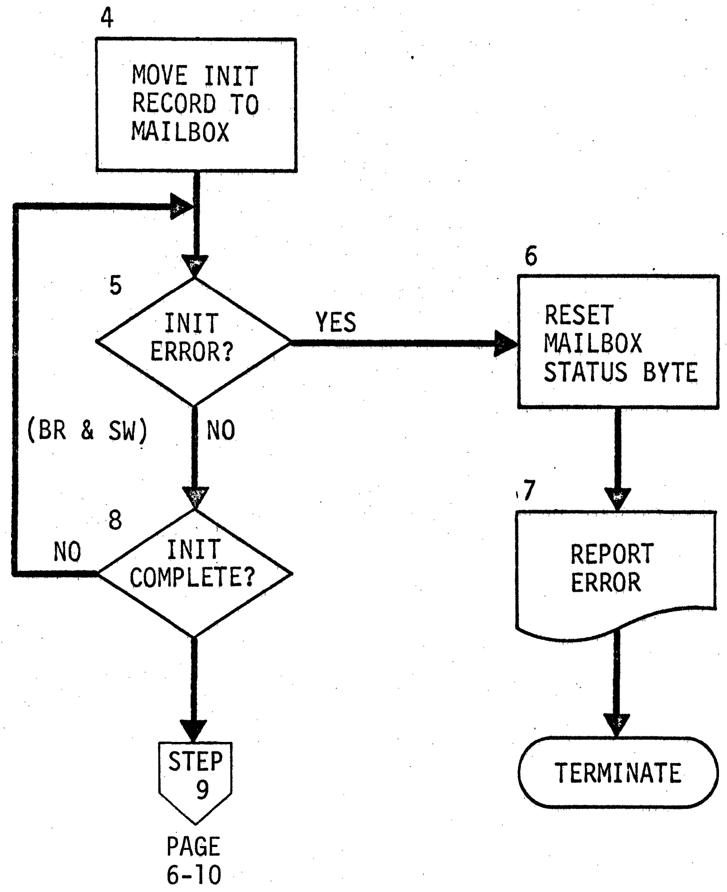


Figure 6-2. UAP Main Program (STEPS 4-8)

4 MOVE THE INITIALIZATION RECORD TO THE SYSTEM MAILBOX

Move the Initialization Record into the System Mailbox and post the Mailbox busy with information from the UAP partition for the SCA partition. The information moved to the System Mailbox is shown under Initialization in this section.

5 INITIALIZATION ERROR?

Inhibit Switch = E?
Was there an error during initialization?
The SCA Module posts an E in the Inhibit Switch (↑ECB) if there is an error in the Initialization Record.

If NO, go to step 8.
If YES, go to step 6.

6 RESET MAILBOX STATUS

Reset the Mailbox Status Byte to A.

7 REPORT ERROR

Print a message on the workstation indicating that the program is ending due to an initialization error and then stop processing.

8 INITIALIZATION COMPLETE

Mailbox Status Byte = A?
Has initialization been successfully completed?
After successful initialization the SCA Module posts the System Mailbox available (A in the Mailbox Status Byte).

If NO, branch and switch to step 5.
If YES, go to step 9.

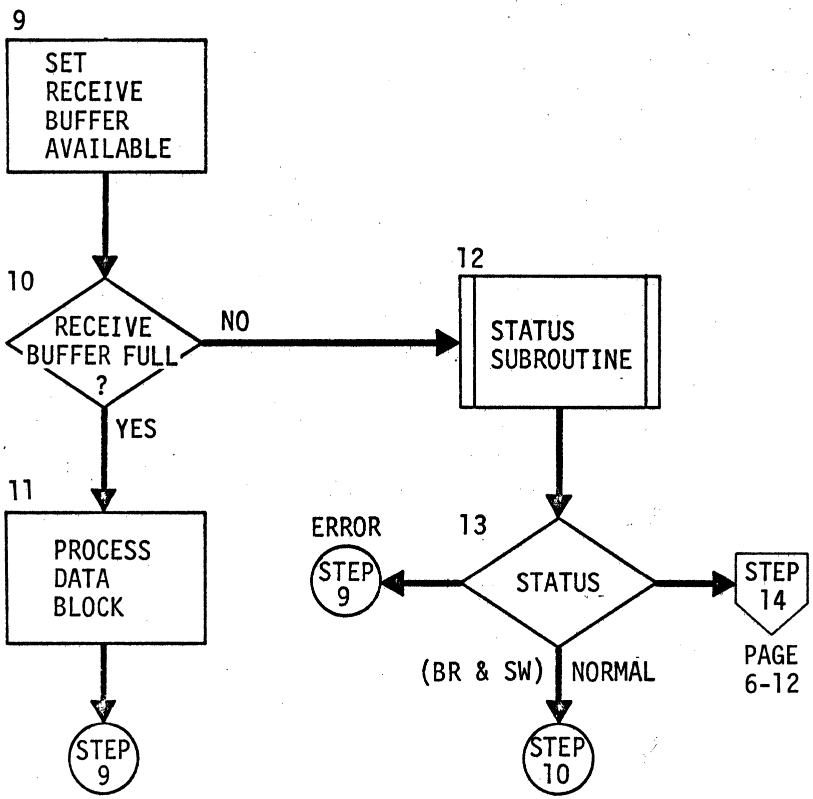


Figure 6-2. UAP Main Program (STEPS 9-13)

9 SET RECEIVE BUFFER AVAILABLE

Post an A (available) in the Buffer Status Byte (relative position 0) of the Receive Buffer.

10 RECEIVE BUFFER FULL?

Buffer Status Byte = E? Has the Receive Buffer been filled with data? The Read Routine posts an E in the Buffer Status Byte when the buffer has been filled.

If YES, go to step 11.
If NO, go to step 12.

Note: A full Receive Buffer or EØT status is the UAP's first indication that a call has come in. Until the first call comes in and the first data block is read or an EØT is received, the UAP loops from this test to the Status Subroutine, to step 13 (which will find the status normal), and back to this test.

11 PROCESS DATA BLOCK

Process the data block just received. In this example the data block is printed on the line printer. Return to step 9.

12 STATUS SUBROUTINE

Branch and link to the Status Subroutine.

13 STATUS

Test the status posted by the Status Subroutine.

Status = normal means that the SCA Module is waiting for a call to come in or for the next data block or an EØT to be received from the calling station. Branch and switch to step 10.

Status = error means that an error has occurred. Return to the beginning of the read logic at step 9.

Status = EØT means an EØT has been received from the calling station. The calling station has no more data to send. Go to step 14 to see if this station has any data to send.

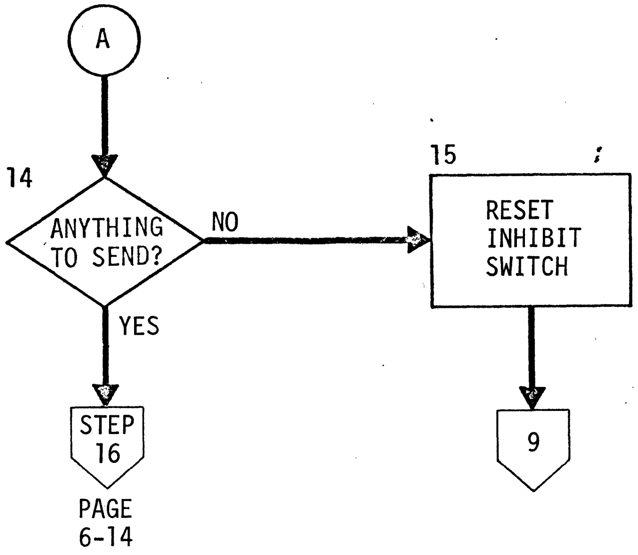


Figure 6-2. UAP Main Program (STEPS 14-15)

14 ANYTHING TO SEND?

Does this station have anything to send? In this example the UAP looks for data in the card reader.

If NO, go to step 15.
If YES, go to step 16.

15 RESET THE INHIBIT SWITCH

Reset the Inhibit Switch to zero. When the SCA Module finds that the UAP has reset the Inhibit Switch without posting a D (write request) in the Operation Code, it sends an EOT.

Go to step 9.

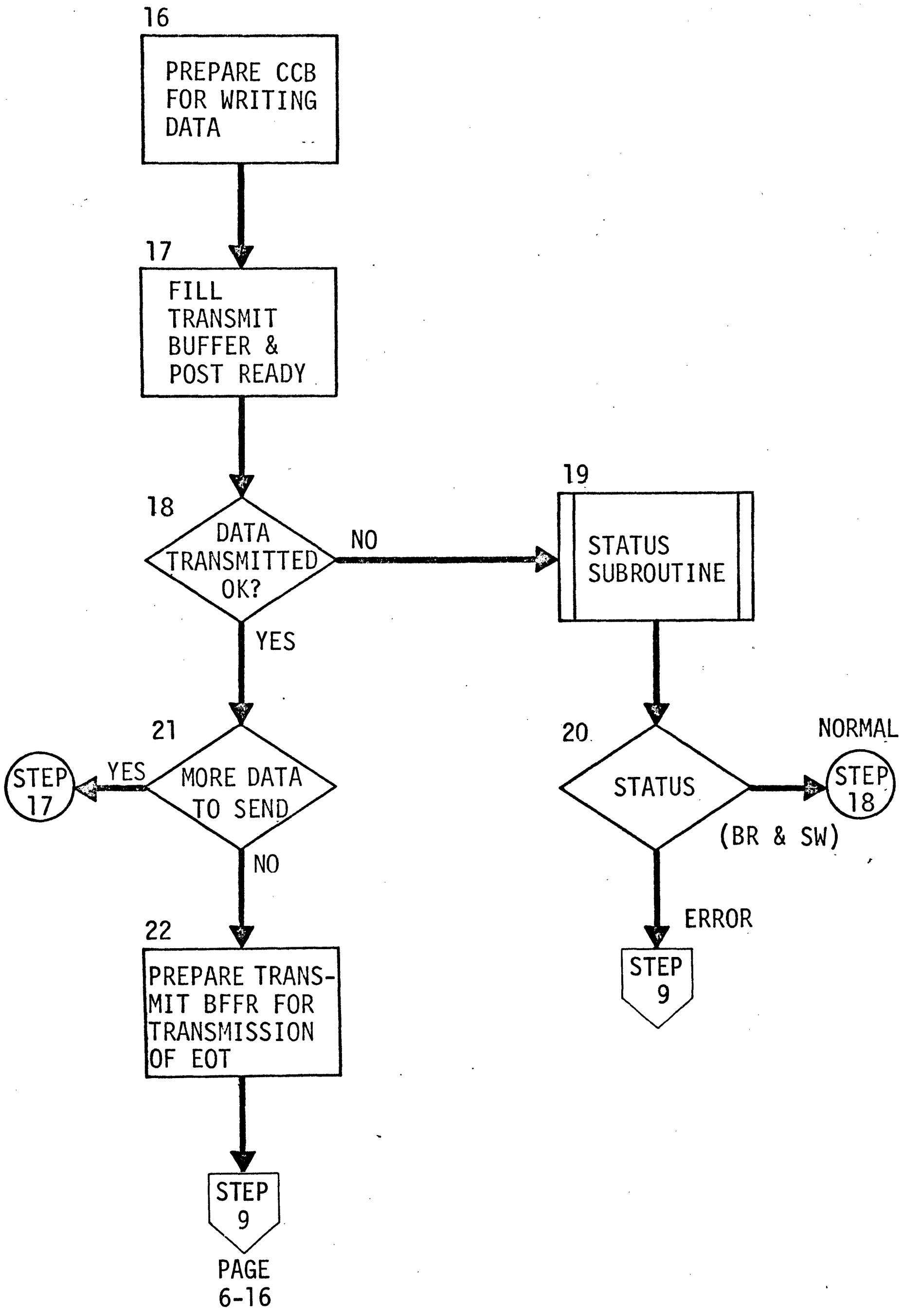


Figure 6-2. UAP Main Program (STEPS 16-22)

16 PREPARE CCB FOR WRITING DATA

Move a D to the Operation Code (↑ECB+2) and reset the Inhibit Switch (↑ECB) to zero. An EØT has just been received and the appropriate code has been posted in the Inhibit Switch. The SCA Module will not continue processing until the UAP has reset the Inhibit Switch to zero. Control will not enter the Write Routine until the Operation Code is set to D by the UAP.

17 FILL THE TRANSMIT BUFFER AND POST IT READY

Move data to the Data field of the Transmit Buffer. Move the data length to the buffer's Data Length field. Move a D (ready) to the Buffer Status Byte.

18 DATA TRANSMITTED OK?

Buffer Status Byte = A?
Has the data block been transmitted successfully? The Write Routine posts an A in the Buffer Status Byte after successfully transmitting the data block.

If YES, go to step 21.
If NO, go to step 19.

19 STATUS SUBROUTINE

Branch and link to the Status Subroutine.

20 STATUS

Test the status posted by the Status Subroutine.

Status = normal means that the Write Routine is still trying to transmit the data block. Branch and switch to step 18.

Status = error means that an error has occurred or an EØT or DLE EØT has been received. (Note: Receipt of an EØT while writing is considered an error and causes error status, not EØT status, to be posted.) Return to the beginning of the read logic at step 9.

21 MORE DATA TO SEND?

Does this station have more data to send? In this example the UAP looks for more data in the card reader.

If YES, go to step 17.
If NO, go to step 22.

22 PREPARE THE TRANSMIT BUFFER FOR TRANSMISSION OF AN EØT

Move a Q to the Write Flag and a D to the Buffer Status Byte in the Transmit Buffer. A Q in the Write Flag indicates that the Write Routine should send an EØT instead of data. The Write Routine will not look at the Write Flag unless the buffer is posted ready (D in the Status Byte). Return to step 9, the beginning of the read logic, to wait for more data or an EØT from the calling station or for a new call to come in.

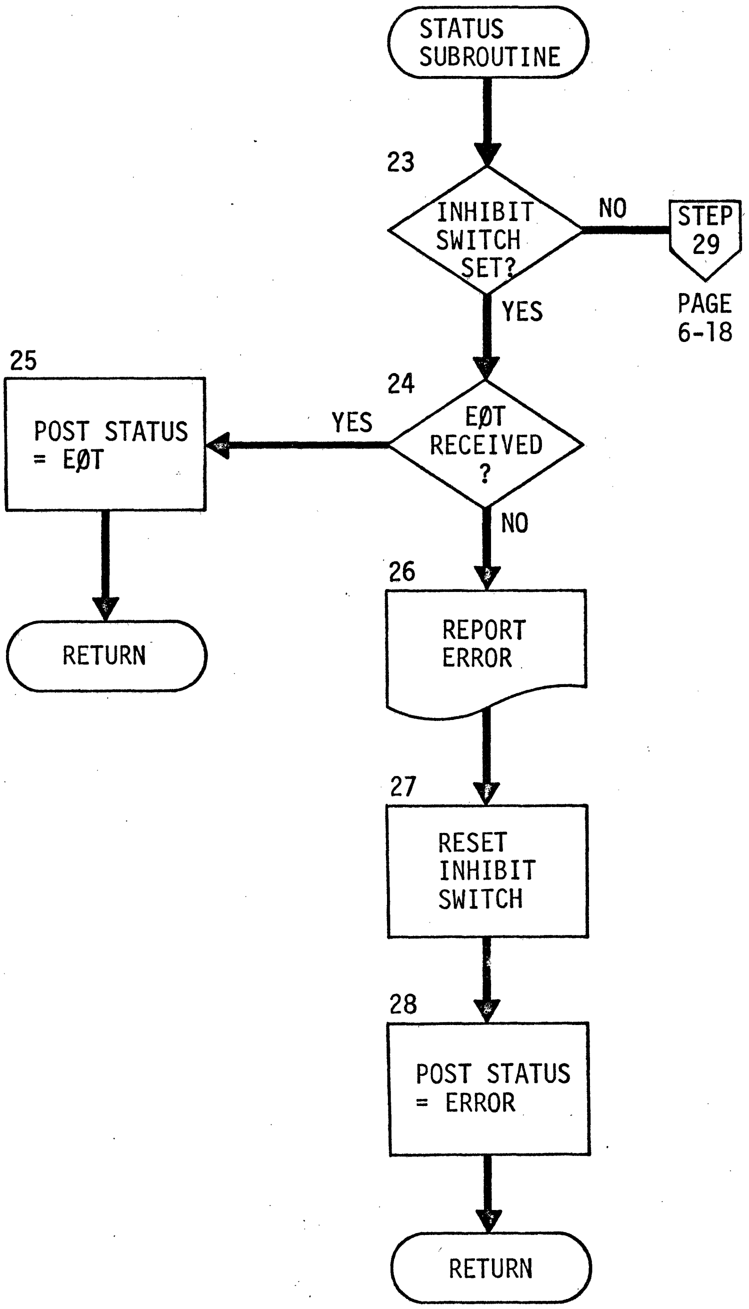


Figure 6-2. UAP Main Program (STEPS 23-28)

STATUS SUBROUTINE

23 INHIBIT SWITCH SET?

Inhibit Switch = some code other than zero?
Has the SCA Module posted an error or EØT code in the Inhibit Switch?

If YES, go to step 24.
If NO, go to step 29.

24 EØT RECEIVED?

Inhibit Switch = W or X?
Was an EØT received while reading or during handshaking? The SCA Module posts a W in the Inhibit Switch if an EØT is received while reading data. It posts an X if an EØT is received during handshaking. Either code means that the calling station is ready to receive data from this station.

If YES, go to step 25.
If NO, then the Inhibit Switch setting indicates an error. Go to step 26.

25 POST STATUS = EØT

Return to the main program with EØT status posted.

26 REPORT ERROR

Print a message on the workstation indicating that an Inhibit Switch error has occurred and the line is being disconnected. Print the error code.

27 RESET INHIBIT SWITCH

Reset the Inhibit Switch to zero.

28 POST STATUS = ERROR

Return to the main program with error status posted.

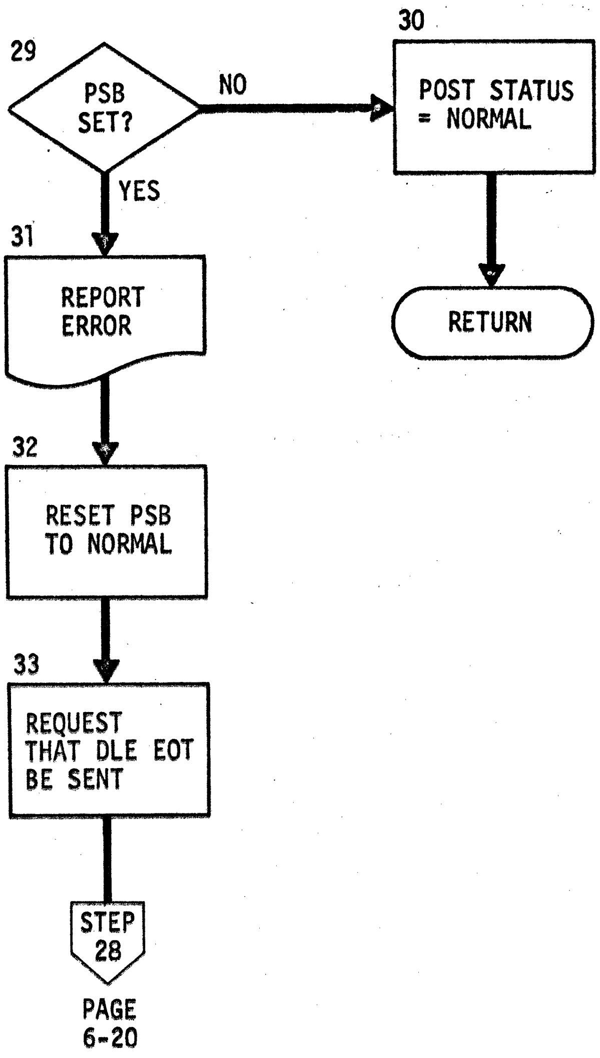


Fig [illegible]-2. UAP Main Program (STEPS 29-33)

29 PARTITION STATUS BYTE SET?

PSB = some character other than N?
Has the SCA Module posted an error code in the PSB?

If NO, go to step 30.
If YES, go to step 31.

30 POST STATUS = NORMAL

Return to the main program with normal status posted.

31 REPORT ERROR

Print a message on the workstation indicating that a PSB error has occurred. Print the error code.

32 RESET PSB TO NORMAL

Move an N (normal) to the Partition Status Byte.

33 REQUEST THAT DLE EØT BE SENT

Test the Activity Switch (↑ECB+3 in the CCB) to see if this station is reading data (Activity Switch = A) or writing data (Activity Switch = D) or if the line has been disconnected (Activity Switch = 0).

To send a DLE EØT while reading data, move a 2 (send DLE EØT) to the Read Flag in the CCB.

To send a DLE EØT while writing data, move a D (send DLE EØT) to the Write Flag in the Transmit Buffer and a D (ready) to the Buffer Status Byte.

If the line has been disconnected, do not send a DLE EØT.

Go to step 28.

DIAL SCAM USAGE -- SUPPLEMENTAL INFORMATION

INTRODUCTION

This section provides supplemental information about the SCAM for dial configurations. It should be used as reference in conjunction with the general SCAM manual, SCA Data Communications Chapter 2. This section contains changes and additions to the general SCAM manual. It will be deleted when the general manual is updated and rereleased.

SCAM ROUTINES

Handshaking Routine

Select Version 5 (Dial-Answering) or Version 6 (Dial-Calling) of the Handshaking Routine.

Initialization Routine

Select any version of the Initialization Routine. Version 1 (One-Time Initialization) is the easiest to use and should be adequate for most applications.

Get-Buffer Routine

Select any version of the Get-Buffer Routine. Version 1 (Sequential) should be selected for most standard file transmissions (for transmissions in which the records are to be kept in sequence).

Number of Buffers

The UAP indicates how many Transmit Buffers and how many Receive Buffers it is using by placing the addresses of the buffers in the Initialization Record. The design of the UAP is simplified if only one Transmit Buffer and one Receive Buffer are defined. (The sample dial-answering UAP presented in Section 6 uses only one Transmit and one Receive Buffer.) However, transmission will be faster if more buffers are defined. For line speeds up to 2400 bps it is recommended that two Transmit Buffers and two Receive Buffers be defined. (The sample dial-calling UAP presented in Section 4 uses two Transmit and two Receive Buffers.)

Sequential Buffers

If Version 1 (Sequential) Get-Buffer is selected, the SCA Module uses the buffers one at a time, in sequence, beginning with the first. The UAP must use the buffers in the same sequence. The SCA Module uses index registers to point to the "current" Transmit and Receive Buffers. The following rules, showing the conditions under which the sequential Get-Buffer updates the Transmit Buffer pointer, should help the user to use the buffers correctly.

1. If the current Transmit Buffer is not ready (status byte not = D), the SCA Module waits for that buffer to be posted ready. (The Write routine sends ENQs and IDLE messages while it waits.)

2. If the buffer is ready (status byte = D) but the UAP has requested in the Write Flag of the buffer that an EOT or DLE EOT be sent instead of data, it sends the EOT or DLE EOT (and it resets the Write Flag to zero and the buffer status byte to A), but it does not update the Transmit Buffer pointer. The next time the Write Routine is entered, it waits for the same Transmit Buffer to be posted ready.

3. If the buffer is ready and contains data to be transmitted (no EOT or DLE EOT request), the SCA Module tries to transmit the data block. If there is an error in transmission, it posts an E (error) in the buffer status byte and an error code in the Inhibit Switch or the PSB or both, but it does not update its Transmit Buffer pointer. The next time the Write Routine is entered, it waits for the same buffer to be posted ready.

4. When the Write Routine has successfully transmitted data from a given Transmit Buffer, it resets the status byte to A (available) and the Write Flag to zero and then it continues to the next Transmit Buffer.

Sequential Receive Buffers are handled in approximately the same way.

1. If the current Receive Buffer is not available (status byte not = A), the SCA Module waits for that buffer to be posted available. (The Read Routine sends WACKs while it waits.)

2. If the current Receive Buffer is available, the SCA Module tries to receive data from the other station. If it receives an EOT or DLE EOT instead of data, or if an error occurs, it posts the appropriate code in the PSB or the Inhibit Switch or both, but it does not update its Receive Buffer pointer. The next time the Read Routine is entered it will try to fill the same Receive Buffer.

3. When the Read Routine has successfully received data into a given Receive Buffer, it posts an E (filled) in the buffer status byte and then it tries to fill the next buffer.

Version 3 of the Get-Buffer Routine (Priority-Sequential) treats all Receive Buffers as sequential buffers (as described above). It treats the first Transmit Buffer as a priority buffer. It always checks the priority Transmit Buffer first. If that buffer is not ready, it checks one of the remaining (sequential) buffers. (See Get-Buffer Routine: Version 3 on page 2-10 in Chapter 2.)

The use of priority buffers is described under Get-Buffer Routine: Version 2 on page 2-10 in Chapter 2.

Error Routine

Select either version of the Error Routine. Version 2 (Expanded Error Counts) is recommended unless there is not enough memory available for its use.

Read Routine

The Read Routine must be included in the assembly of a Dial-Answering SCA Module. If the answering station never actually receives data, only one Receive Buffer is needed, and it can be only 22 positions (the length of the buffer prefix plus two).

A Dial-calling SCA Module need not contain a Read Routine if the calling station never receives data. It is not necessary to define a Receive Buffer if the Read Routine is omitted. For error-free assembly of the calling SCA Module, the following instructions must be inserted in place of the Read Routine:

```
↑READ    BC   ↑RDEX(5)

↑RDEX    BC   0(5)

↑RD8     DM   5C
```

Write Routine

The Write Routine must be included in the assembly of a Dial-Calling SCA Module. If the calling station never actually transmits data, only one Transmit Buffer is needed, and it can be only 20 positions (the length of the buffer prefix).

A Dial-Answering SCA Module need not contain a Write Routine if the answering station never transmits data. It is not necessary to define a Transmit Buffer if the Write Routine is omitted. For error-free assembly of the answering SCA Module, the following instructions must be inserted in place of the Write Routine:

```
↑WRITE  BC  ↑WRTEX(5)

↑WRTEX  BC  0(5)
```

INTERPARTITION INTERFACE

Communications Control Block

Inhibit Switch (↑ECB)

Both the calling and answering SCA Modules disconnect the line after posting any code other than zero, X, or W in the Inhibit Switch. The meaning of code X is discussed below. The meaning of code W is discussed under Inhibit Switch in Chapter 2.

All of the Inhibit Switch codes discussed in Chapter 2 may be posted by a dial SCA Module. In addition, the following codes may be posted by a dial Handshaking Routine.

Code Meaning

P An error in handshaking immediately after the call was established. The errors described under PSB codes A, C, D and S in Chapter 2 or the receipt of an invalid WRU ID will cause this code to be posted.

O (alpha) DLE EOT or EOT received during handshaking immediately after the call was established.

When this code is posted by a calling SCA Module it usually menas that the answering station did not accept the calling station's I AM ID.

Z Error in handshaking after the first transmission.

The errors described under PSB codes A, C, D, and S in Chapter 2 will cause this code to be posted.

X EOT received by the Handshaking Routine.

The other station sent an EOT instead of an ENQ when it had the option to send data. This normally means the other station has no data to send and is ready to receive data from this station.

Read Flag (↑ECB+1)

The dial SCA Module's use of the Read Flag is described on page 3-6 of Chapter 2.

Operation Code (↑ECB+2)

Code — Meaning

A — Read request.

The dial SCA Module does not check the Operation Code for an A before entering its Read Routine. There is no need for a dial UAP to post an A in the Operation Code.

D — Write Request.

After a dial SCA Module has received an EOT from the other station, it waits for its UAP to reset the Inhibit Switch to zero and then it checks the Operation Code. It enters its Write Routine only if the UAP has posted a D in the Operation Code. This means that the Dial-Answering SCA Module will never enter its Write Routine unless the UAP has posted a D in the Operation Code.

The Dial-Calling SCA Module enters its Write Routine automatically after a call has been placed and initial handshaking (and ID exchange, if requested) has taken place, regardless of the Operation Code setting. It will not reenter its Write Routine unless the UAP has posted a D in the Operation Code.

0 (zero) — Off.

The SCA Module moves a zero to the Operation Code whenever it exits its Read Routine or Write Routine.

After receiving an EOT from the other station, the dial SCA Module waits for the UAP to reset the Inhibit Switch and then checks the Operation Code for a D (see code D above). If it finds a zero instead of a D, it assumes that the UAP has no data to send.

When a Dial-Answering SCA Module Handshaking Routine finds that its UAP has no data to send, it sends an EOT to the calling station. If the calling station is a System Ten with an SCA Module, this EOT will cause an X (EOT received by the Handshaking Routine) to be posted in the Inhibit Switch of the calling station. (See Inhibit Switch code X above.)

When a Dial-Calling SCA Module Handshaking Routine finds that its UAP has no data to send, its action depends on the timing of the EOT just received. If the EOT was received by the Read Routine (see Inhibit Switch code W in Chapter 2), it sends an EOT and waits for a response. If the EOT was received by the Handshaking Routine (see Inhibit Switch code X above), then neither station has data to send: it sends a DLE EOT and disconnects the line.

Code	Meaning
E	Disconnect request. After receiving an EOT from the other station, the dial SCA Module waits for the UAP to reset the Inhibit Switch and then checks the Operation Code for a D (see code D above). If it finds an E instead of a D it sends a DLE EOT and disconnects the line.

Activity Switch (↑ECB+3)

The SCA Module posts a code in the Activity Switch to indicate whether it is in its Read Routine, its Write Routine, or its Handshaking Routine. The UAP can test the switch but it should not alter it.

Code	Meaning
A	The SCA Module is in its Read Routine.
D	The SCA Module is in its Write Routine.
zero	The SCA Module is in its Handshaking Routine. The Activity Switch is zero from the time the SCA Module is loaded until it establishes the first connection and completes initial handshaking and ID exchange. It is reset to zero when the line is disconnected and remains zero until a new connection is established and initial handshaking and ID exchange is again completed.

Partition That Last Initialized (↑ECB+4,5) and SCA's Partition Number (↑ECB+6,7)

The dial SCA Module's use of these fields is described on page 3-7 of Chapter 2.

Partition Status Byte

The dial SCA Module's use of the PSB is described on pages 3-8 to 3-12 of Chapter 2.

System Mailbox and Initialization Record

The dial SCA Module's use of the System Mailbox for initialization is described on pages 3-13 to 3-17 of Chapter 2.

Transmit and Receive Buffers

The dial SCA Module's use of Transmit and Receive Buffers is described on pages 3-18 to 3-28 of Chapter 2 and under Get-Buffer Routine in this section.

IDLE Message Buffer

The IDLE Message Buffer does not have to be used by the UAP to pass phone numbers to the Dial-Calling SCA Module. A new area in Common, the Call Work Area is used instead. (See Call Work Area below.) The Call Work Area may overlap the IDLE Message Buffer.

The dial SCA Module does use the IDLE Message Buffer for the transmission of IDLE Messages as described on pages 3-29 to 3-31 of Chapter 2.

Call Work Area

The Call Work Area is a 44-character area in Common used by a calling UAP to pass phone numbers to the Dial-Calling SCA Module. The Call Work Area may overlap the IDLE Message Buffer.

Format of the Call Work Area

STATUS BYTE	NOT USED				LENGTH		PHONE NUMBER			
↑CWA	+1			+17	+18	+19	+20			+43

Call Work Area Fields

Relative Location	Length	Meaning
0	1	Status Byte. D -- ready. The UAP moves a D to the status byte to indicate to the SCA Module that it has prepared the Call Work Area with a phone number to be dialed. A -- available. The SCA Module moves an A to the Status byte to indicate to the UAP that the phone number has been successfully dialed. E -- error. The SCA Module moves an E to the status byte to indicate to the UAP that there was an error in dialing (for example, a busy signal or no answer or an invalid phone number). The SCA Module tries eight times to place the call before it posts an E in the status byte. This may take several minutes.
1-17	17	Unused.
18-19	2	Length. The UAP moves the two-digit length of the phone number to be dialed to this field.
20-43	24	Phone Number. The UAP moves the phone number to be dialed (up to 24 digits) to this field.

APPENDIX A

LISTING OF SAMPLE DIAL-CALLING UAP PRESENTED IN SECTION 4

```
PAGE 0001  11/24/71  (4-02)      LABEL TABLE FOR USER APPLICATION PROGRAM FOR DIAL-CALLING STATION

 LABEL     LOC   LNTH  INDX         MESSAGES

 A        1700    01    0
 BCLEAR   1490    10    0
 BCLREX   1690    10    0
 BF2ADR   1719    04    0
 BF4ADR   1723    04    0
 BUFF1    1500C   20    0
 BUFF2    1600C   20    0
 BUFF3    1700C   20    0
 BUFF4    1900C   20    0
 BUFS     2061    04    0        UNUSED
 CALLNO   1728    06    0
 D        1701    01    0
 DIAL00   0370    10    0
 DIAL05   0430    10    0
 DIAL10   0450    10    0
 DIAL15   0470    10    0
 DIAL20   0490    10    0
 DIAL25   0540    10    0
 DIAL30   0580    10    0
 DIAL35   0630    10    0
 DIAL37   0670    10    0
 DIAL40   0710    10    0
 DIAL45   0780    10    0
 DIAL55   0850    10    0
 DIAL60   0860    10    0
 DIAL65   0970    10    0
 DIAL70   0990    10    0
 DIAL75   1010    10    0
 DIAL80   1050    10    0
 DIAL85   1090    10    0
 E        1702    01    0
 ERR1     1738    20    0
 ERR2     1758    21    0
 ERR3     1779    14    0
 ERR4     1793    10    0
 ERR6     1803    15    0
 ERR6A    1818    06    0
 ERR6B    1824    01    0
 ERR7     1825    20    0
 ERR7A    1845    80    0
 ERR8     1925    20    0
 ERR8A    1945    80    0
 ERR9     2025    30    0
 FRSTWT   1727    01    0
 FUNC     1737    01    0
 IDEN     2119    40    0        UNUSED
 K2       1713    01    0
 K5       1714    01    0
 KON80    1715    04    0
 M        1703    01    0
 MAIL     2056    05    0
 N        1704    01    0
 OPCON    1734    03    0
 Q        1705    01    0
 QM       1709    01    0
 SPACE    2055    01    0
```

```
PAGE 0002  11/24/71  (4-02)      LABEL TABLE FOR USER APPLICATION PROGRAM FOR DIAL-CALLING STATION

 LABEL     LOC  LNTH INDX          MESSAGES

 STAT15   1250   10   0
 STAT20   1270   10   0
 STAT30   1340   10   0         UNUSED
 STAT35   1370   10   0
 STAT40   1380   10   0
 STAT45   1390   10   0
 STAT50   1420   10   0
 STAT55   1450   10   0
 STAT60   1550   10   0
 STAT65   1610   10   0
 STAT70   1680   10   0
 STATEX   1480   10   0
 STATUS   1160   10   0
 V        1706   01   0
 W        1707   01   0
 X        1708   01   0         UNUSED
 X1       0011   04   0
 X2       0021   04   0
 X3       0031   04   0
 ZEROS    1710   03   0
 ↑B1      0001C  02   0         UNUSED
 ↑B10     0010C  01   0
 ↑B5      0005C  04   0
 ↑B9      0009C  01   0
 ↑BASE    0000C  01   0
 ↑CWA     1050C  44   0
 ↑ECB     1000C  18   0
 ↑IDEN    0643C  99   0         UNUSED
 ↑IDLEB   1050C  80   0         UNUSED
 ↑IMSG    0560C  20   0         UNUSED
 ↑K101    0101C  04   0         UNUSED
 ↑MAIL    0580C  05   0
 ↑SEND    0585C  17   0         UNUSED
 ↑SINK    0602C  21   0         UNUSED
 ↑STAT    0560C  20   0
 91 LABELS
```

```
PAGE 0001  11/24/71  (4-02) PROGRAM LISTING FOR USER APPLICATION PROGRAM FOR DIAL-CALLING STATION

SEQ. LOCN  INSTR/DATA OP  A/R   L I  B/S   L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0020 0000                                      0002  *
0030 0000                                      0003  *
0040 0000                                      0004  * SAMPLE USER APPLICATION PROGRAM FOR DIAL-CALLING STATION
0050 0000                                      0005  *
0060 0000                                      0006  *


0080 0000                                      0008         NORMAL
0090 0000                 0011                 0009         ORG    11
0100 0011  170P           0001       0004      0010  X1     DM     A'BUFF3'               ADDRESS OF RECEIVE BUFFER
0110 0015                 0021                 0011         ORG    21
0120 0021  056P           0001       0004      0012  X2     DM     A'↑STAT'               ADDRESS OF PSB'S
0130 0025                 0031                 0013         ORG    31
0140 0031  150P           0001       0004      0014  X3     DM     A'BUFF1'               ADDRESS OF TRANSMIT BUFFER
0150 0035                 0300                 0015         ORG    300
0160 0300                                      0016  *
0170 0300                                      0017  * STEP 1
0180 0300                                      0018  *

0200 0300  PQPOV21710 14  1006C 0 0  1710  2 0 0020         C      ↑ECB+6(2),ZEROS        IS SCA ACTIVE?
0210 0310  SOSR080300 11  0320  3 0  0300  8 0 0021         BC     *+10(3),*-10(8)        IF NOT WAIT FOR IT
0220 0320                                      0022  *
0230 0320                                      0023  * STEP 2
0240 0320                                      0024  *

0260 0320  2QPPV40021 07  1006C 2 0  0021  4 0 0026         S      ↑ECB+6(2),X2           COMPUTE SCA'S PSB ADDR. SAVE IN X2
0270 0330  P170410POP 08  1704  0 0  0000C 1 2 0027         MC     N,↑BASE(,2)            SET SCA'S PSB TO NORMAL
0280 0340                                      0028  *
0290 0340                                      0029  * STEP 3
0300 0340                                      0030  *

0320 0340  PPU8P11700 14  0580C 0 0  1700  1 0 0032         C      ↑MAIL(1),A             IS MAILBOX AVAILABLE?
0330 0350  ROSV080340 11  0360  2 0  0340  8 0 0033         BC     *+10(2),*-10(8)        IF NOT WAIT FOR IT
0340 0360                                      0034  *
0350 0360                                      0035  * STEP 4
0360 0360                                      0036  *

0380 0360  P20560058P 08  2056  0 0  0580C 0 0 0038         MC     MAIL(100),↑MAIL        MOVE INIT RECORD TO MAIL-BOX
0390 0370                                      0039  *
0400 0370                                      0040  * STEP 5
0410 0370                                      0041  *

0430 0370  PQPOP11702 14  1000C 0 0  1702  1 0 0043  DIAL00 C      ↑ECB(1),E              WAS THERE INIT ERROR?
0440 0380  ROSY050430 11  0390  2 0  0430  5 0 0044         BC     *+10(2),DIAL05(5)      IF NOT BRANCH
0450 0390                                      0045  *
0460 0390                                      0046  * STEP 6
0470 0390                                      0047  *

0490 0390  P17001058P 08  1700  0 0  0580C 1 0 0049         MC     A,↑MAIL                RESET MAIL-BOX AVAILABLE
0510 0400                                      0050  *
0520 0400                                      0051  * STEP 7
0530 0400                                      0052  *

0550 0400  017P330001 01  1703  0 0  0001  3 0 0054         W      M(0),1(3)              CARRIAGE RETURN TO WORK STATION
0560 0410  017S810020 01  1738  0 0  0020  1 0 0055         W      ERR1(0),20(1)          REPORT ERROR
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0570 0420  U9YY000000 11 9990  5 0 0000  0 0 0056         BC     9990(5)              BLOW UP
0580 0430                                    0057  *
0590 0430                                    0058  * STEP 8
0600 0430                                    0059  *

0620 0430  PPU8P11700 14 0580C 0 0 1700  1 0 0061  DIAL05 C      ↑MAIL(1),A           IS INIT COMPLETE?
0630 0440  R0TU080370 11 0450  2 0 0370  8 0 0062         BC     *+10(2),DIAL00(8)    YES 60 ON,ELSE WAIT FOR INIT
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0650 0450                                    0064  *
0660 0450                                    0065  * STEP 9 INITIALISATION IS COMPLETE,FETCH NO TO CALL.
0670 0450                                    0066  *

0690 0450  P17Q010490 09 1710  0 0 0490  1 0 0068  DIAL10 MN     ZEROS(1),DIAL20       SET REPORT SWITCHES
0700 0460  P17Q010580 09 1710  0 0 0580  1 0 0069         MN     ZEROS(1),DIAL30
0710 0470  117P930001 01 1709  1 0 0001  3 0 0070  DIAL15 W      QM(1),1(3)            IS CARD READER READY?
0720 0480  TOTY050540 11 0490  4 0 0540  5 0 0071         BC     *+10(4),DIAL25(5)     IF SO BRANCH
0730 0490                                    0072  *
0740 0490                                    0073  * STEP 10
0750 0490                                    0074  *

0770 0490  POTW000000 11 0470  0 0 0000  0 0 0076  DIAL20 BC     DIAL15(0)             BRANCH BACK UNLESS REPORT SWITCH
0780 0500  P17Q410490 09 1714  0 0 0490  1 0 0077         MN     K5,DIAL20             RESET REPORT SWITCH
0790 0510  017P330001 01 1703  0 0 0001  3 0 0078         W      M(0),1(3)             CARRIAGE RETURN TO WORK STATION
0800 0520  017U810021 01 1758  0 0 0021  1 0 0079         W      ERR2(0),21(1)         REPORT CARD READER NOT READY
0810 0530  UOTW000000 11 0470  5 0 0000  0 0 0080         BC     DIAL15(5)             BRANCH BACK
0820 0540                                    0081  *
0830 0540                                    0082  * STEP 11
0840 0540                                    0083  *

0860 0540  1105P10044 00 1050C 1 0 0044  1 0 0085  DIAL25 R      ↑CWA(1),44(1)         READ A CARD INTO CALL WORK AREA
0870 0550                                    0086  *
0880 0550                                    0087  * STEP 12
0890 0550                                    0088  *

0910 0550  SOTU000000 11 0450  3 0 0000  0 0 0090         BC     DIAL10(3)             GO BACK IF UNIT SEPARATOR
0920 0560                                    0091  *
0930 0560                                    0092  * STEP 13
0940 0560                                    0093  *
0950 0560  PQP5P61728 14 1050C 0 0 1728  6 0 0094         C      ↑CWA(6),CALLNO        DOES THIS CARD CONTAIN 'CALLNO'
0960 0570  ROVS000000 11 0630  2 0 0000  0 0 0095         BC     DIAL35(2)             YES BRANCH
0970 0580                                    0096  *
0980 0580                                    0097  * STEP 14
0990 0580                                    0098  *

1010 0580  POUT000000 11 0540  0 0 0000  0 0 0100  DIAL30 BC     DIAL25(0)             BRANCH BACK UNLESS REPORT SWITCH
1020 0590  P17Q410580 09 1714  0 0 0580  1 0 0101         MN     K5,DIAL30             RESET REPORT SWITCH
1030 0600  017P330001 01 1703  0 0 0001  3 0 0102         W      M(0),1(3)             CARRIAGE RETURN TO WORK STATION
1040 0610  017W910014 01 1779  0 0 0014  1 0 0103         W      ERR3(0),14(1)         REPORT NO CALLNO
1050 0620  UOUT000000 11 0540  5 0 0000  0 0 0104         BC     DIAL25(5)             BRANCH BACK TO FIND CALLNO
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

1070 0630                                    0106  *
1080 0630                                    0107  * STEP 15     THIS SECTION TRANSMITS DATA
1090 0630                                    0108  *

1110 0630  P105V11737 08 1056C 0 0 1737  1 0 0110  DIAL35 MC     ↑CWA+6(1),FUNC       SAVE 7TH CHAR OF CARD JUST READ
1120 0640                                    0111  *
1130 0640                                    0112  * STEP 16
1140 0640                                    0113  *

1160 0640  PQW3711707 14 1737  0 0 1707  1 0 0115         C      FUNC,W               DOES IT SAY WRITE
1170 0650  ROVV050850 11 0660  2 0 0850  5 0 0116         BC     *+10(2),DIAL55(5)    IF NOT BRANCH TO READ
1180 0660                                    0117  *
1190 0660                                    0118  * STEP 17
1200 0660                                    0119  *

1220 0660  P171411727 08 1714  0 0 1727  1 0 0121         MC     K5,FRSTWT            TURN ON FIRST WRITE SWITCH
1230 0670                                    0122  *
1240 0670                                    0123  * STEP 18
1250 0670                                    0124  *

1270 0670  PPPOPQQ700 14 0000C 0 3 1700  1 0 0126  DIAL37 C      ↑BASE(1,3),A         IS XMIT BUFFER AVAILABLE?
1280 0680  ROWQ000000 11 0710  2 0 0000  0 0 0127         BC     DIAL40(2)            IF YES GO TO READ A NEW CARD
1290 0690                                    0128  *
1300 0690                                    0129  * STEP 19
1310 0690                                    0130  *

1330 0690  V1TX151160 11 1481  6 0 1160  5 0 0132         BC     STATEX+1(6),STATUS(5) ELSE GO TO STATUS ROUTINE
1340 0700                                    0133  *
1350 0700                                    0134  * STEP 20
1360 0700                                    0135  *

1380 0700  Q0TU080670 11 0450  1 0 0670  8 0 0137         BC     DIAL10(1),DIAL37(8)  RESTART OR NORMAL
1390 0710                                    0138  *
1400 0710                                    0139  * STEP 21
1410 0710                                    0140  *

1430 0710  1002PQP080 00 0020C 1 3 0080  1 0 0142  DIAL40 R      ↑B10+10(1,3),80(1)   READ NEXT CARD INTO XMIT BUFFER
1440 0720                                    0143  *
1450 0720                                    0144  * STEP 22
1460 0720                                    0145  *

1480 0720  S0WS050780 11 0730  3 0 0780  5 0 0147         BC     *+10(3),DIAL45(5)    BRANCH IF OK
1490 0730                                    0148  *
1500 0730                                    0149  * STEP 23
1510 0730                                    0150  *

1530 0730  PQW2711710 14 1727  0 0 1710  1 0 0152         C      FRSTWT(1),ZEROS      IF FLAG CHECK IST TIME SWITCH
1540 0740  R1PQ000000 11 1010  2 0 0000  0 0 0153         BC     DIAL75(2)            IF SET WE HAVE NOT DIALED YET ELSE
1550 0750                                    0154  *                                GO TO RECEIVE
1560 0750                                    0155  * STEP 24
1570 0750                                    0156  *

1590 0750  017P330001 01 1703  0 0 0001  3 0 0158         W      M(0),1(3)            REPORT NO DATA TO SEND
1600 0760  020R510030 01 2025  0 0 0030  1 0 0159         W      ERR9(0),30(1)
1610 0770  U0TU000000 11 0450  5 0 0000  0 0 0160         BC     DIAL10(5)            BRANCH BACK
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S   L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

1620 0780                                     0161   *
1630 0780                                     0162   * STEP 25
1640 0780                                     0163   *

1660 0780  P171540PPU 08 1715  0 0 0005C 4 3 0165  DIAL45 MC      KON80,↑B5(,3)        SET LENGTH=80
1670 0790  P170110PPP 08 1701  0 0 0000C 1 3 0166         MC      D,↑BASE(,3)          SET BUFFER READY TO TRANSMIT
1680 0800                                     0167   *
1690 0800                                     0168   * STEP 26
1700 0800                                     0169   *

1720 0800  PQWQ940031 15 1719  0 0 0031   4 0 0171         X       BF2ADR,X3            XCHANGE BUFFER POINTERS
1730 0810                                     0172   *
1740 0810                                     0173   * STEP 27
1750 0810                                     0174   *

1770 0810  PQW2711710 14 1727  0 0 1710   1 0 0176         C       FRSTWT(1),ZEROS      IS FIRST TIME SWITCH SET?
1780 0820  ROVW000000 11 0670  2 0 0000   0 0 0177         BC      DIAL37(2)            NO GO ON TO SEND
1790 0830                                     0178   *
1800 0830                                     0179   * STEP 28
1810 0830                                     0180   *

1830 0830  P171011727 08 1710  0 0 1727   1 0 0182         MC      ZEROS(1),FRSTWT      YES RESET IT AND WAIT FOR DIAL
1840 0840  UOXU000000 11 0850  5 0 0000   0 0 0183         BC      DIAL55(5)
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

1860 0850                                   0185   *
1870 0850                                   0186   * STEP 29  THIS SECTION DIALS THE NO.
1880 0850                                   0187   *

1900 0850  P17011105P 08 1701  0 0 1050C 1 0 0189  DIAL55 MC     D,↑CWA                   MARK CALL WORK AREA READY. CALL NO
1910 0860                                   0190   *                                      AND LENGTH READ IN DIRECTLY FROM
1920 0860                                   0191   * STEP 30                              CARD READER
1930 0860                                   0192   *

1950 0860  PQP0S11701 14 1003C 0 0 1701  1 0 0194  DIAL60 C      ↑ECB+3(1),D              HAS A DATA-LINK BEEN ESTABLISHED?
1960 0870  R0YY000000 11 0990  2 0 0000  0 0 0195         BC     DIAL70(2)                YES
1970 0880  PQP5P11702 14 1050C 0 0 1702  1 0 0196         C      ↑CWA(1),E                NO SO WAS THERE DIAL-ERROR
1980 0890  R0YP050970 11 0900  2 0 0970  5 0 0197         BC     *+10(2),DIAL65(5)        IF NOT BRANCH
1990 0900                                   0198   *
2000 0900                                   0199   * STEP 31
2010 0900                                   0200   *

2030 0900  017P330001 01 1703  0 0 0001  3 0 0202         W      M(0),1(3)                CARRIAGE RETURN TO WORK-STATION
2040 0910  017Y310010 01 1793  0 0 0010  1 0 0203         W      ERR4(0),10(1)            REPORT DIAL ERROR
2050 0920                                   0204   *
2060 0920                                   0205   * STEP 32
2070 0920                                   0206   *

2090 0920  PQW371170/ 14 1737  0 0 1707  1 0 0208         C      FUNC(1),W                WERE WE ABAOUT TO WRITE?
2100 0930  R0YT050450 11 0940  2 0 0450  5 0 0209         BC     *+10(2),DIAL10(5)        IF NOT RESTART
2110 0940                                   0210   *
2120 0940                                   0211   * STEP 33
2130 0940                                   0212   *

2150 0940  P170010PPP 08 1700  0 0 0000C 1 3 0214         MC     A,↑BASE(,3)              RESET FIRST XMIT BUFFER AVAILABLE
2160 0950  PQWQ940031 15 1719  0 0 0031  4 0 0215         X      BF2ADR,X3                AND RESTORE XMIT BUFF POINTERS
2170 0960  U0TU000000 11 0450  5 0 0000  0 0 0216         BC     DIAL10(5)
2180 0970  V1TX151160 11 1481  6 0 1160  5 0 0217  DIAL65 BC     STATEX+1(6),STATUS(5) BRANCH TO STATUS SUBROUTINE
2190 0980  Q0TU080860 11 0450  1 0 0860  8 0 0218         BC     DIAL10(1),DIAL60(8)  RESTART OR NORMAL
2200 0990                                   0219   *
2210 0990                                   0220   * STEP 34
2220 0990                                   0221   *

2240 0990  PQW371170/ 14 1737  0 0 1707  1 0 0223  DIAL70 C      FUNC,W                   TO WRITE OR READ?
2250 1000  R0VW000000 11 0670  2 0 0000  0 0 0224         BC     DIAL37(2)                BRANCH TO WRITE
```

```
SEQ. LOCN  INSTR/DATA OP  A/R   L I  B/S   L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

2270 1010                                    0226  *
2280 1010                                    0227  * STEP 35
2290 1010                                    0228  *

2310 1010  P170510PPY 08 1705  0 0 0009C 1 3 0230  DIAL75 MC     Q,↑B9(,3)            SET EOT IN WRITE FLAG
2320 1020  P170110PPP 08 1701  0 0 0000C 1 3 0231         MC     D,↑BASE(,3)          SET XMIT BUFF READY TO GO
2330 1030                                    0232  *
2340 1030                                    0233  * STEP 36  THIS SECTION RECEIVES DATA
2350 1030                                    0234  *

2370 1030  P17001170P 08 1700  0 0 1700C 1 0 0236         MC     A,BUFF3              SET RECEIVE BUFFERS AVAILABLE
2380 1040  P17001190P 08 1700  0 0 1900C 1 0 0237         MC     A,BUFF4
2390 1050                                    0238  *
2400 1050                                    0239  * STEP 37
2410 1050                                    0240  *

2430 1050  PPPOP1Q702 14 0000C 0 1 1702  1 0 0242  DIAL80 C      ↑BASE(1,1),E         IS RECEIVE BUFFER FULL?
2440 1060  R1PY000000 11 1090  2 0 0000  0 0 0243         BC     DIAL85(2)            YES
2450 1070                                    0244  *
2460 1070                                    0245  * STEP 38
2470 1070                                    0246  *

2490 1070  V1TX151160 11 1481  6 0 1160  5 0 0248         BC     STATEX+1(6),STATUS(5) BRANCH TO STATUS SUBROUTINE
2500 1080                                    0249  *
2510 1080                                    0250  * STEP 39
2520 1080                                    0251  *

2540 1080  Q0TU081050 11 0450  1 0 1050  8 0 0253         BC     DIAL10(1),DIAL80(8)  RESTART OR NORMAL
2550 1090                                    0254  *
2560 1090                                    0255  * STEP 40
2570 1090                                    0256  *

2590 1090  P00PU4Q106 09 0005C 0 1 1106  4 0 0258  DIAL85 MN     ↑B5(4,1),*+16        SET LENGTH FOR PRINT INSTRUCTION
2600 1100  200RP1P100 01 0020C 2 1 0100  1 0 0259         W      ↑B10+10(2,1),100(1)  PRINT RECEIVE BUFFER
2610 1110  P2055100RP 08 2055  0 0 0020C 1 1 0260         MC     SPACE,↑B10+10(,1)    CLEAR FIRST 80 CHARS OF RECV BUFF
2620 1120  W002P9P0RQ 08 0020C 7 1 0021C 9 1 0261         MC     ↑B10+10(79,1),↑B10+11(,1)
2630 1130                                    0262  *
2640 1130                                    0263  * STEP 41
2650 1130                                    0264  *

2670 1130  P1700100PP 08 1700  0 0 0000C 1 1 0266         MC     A,↑BASE(,1)          SET RECV BUFFER AVAILABLE
2680 1140                                    0267  *
2690 1140                                    0268  * STEP 42
2700 1140                                    0269  *

2720 1140  PPPQ141723 15 0011  0 0 1723  4 0 0271         X      X1,BF4ADR            XCHANGE RECV BUFFER POINTERS
2730 1150  U1PU000000 11 1050  5 0 0000  0 0 0272         BC     DIAL80(5)            BRANCH BACK FOR NEXT ONE
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

2750 1160                                     0274  *
2760 1160                                     0275  * STEP 43    STATUS ROUTINE
2770 1160                                     0276  *

2790 1160  PQPOP11710 14 1000C 0 0 1710  1 0 0278  STATUS C      ↑ECB(1),ZEROS        IS INHIBIT SWITCH SET?
2800 1170  R1RW000000 11 1270  2 0 0000  0 0 0279         BC     STAT20(2)            NO
2810 1180                                     0280  *
2820 1180                                     0281  * STEP 44
2830 1180                                     0282  *

2850 1180  P100P11818 08 1000C 0 0 1818  1 0 0284         MC     ↑ECB(1),ERR6A        ELSE PREPARE REPORT WITH INH SW
2860 1190  P000PW1824 08 0000C 0 2 1824  1 0 0285         MC     ↑BASE(1,2),ERR6B     AND PSB CODES
2870 1200                                     0286  *
2880 1200                                     0287  * STEP 45
2890 1200                                     0288  *

2910 1200  V1VY151490 11 1691  6 0 1490  5 0 0290         BC     BCLREX+1(6),BCLEAR(5) BRANCH TO SUBR TO CLEAR XMIT BFS
2920 1210                                     0291  *
2930 1210                                     0292  * STEP 46
2940 1210                                     0293  *

2960 1210  PQPOP11707 14 1000C 0 0 1707  1 0 0295         C      ↑ECB(1),W            END OF DATA?
2970 1220  R1RU000000 11 1250  2 0 0000  0 0 0296         BC     STAT15(2)            YES
2980 1230                                     0297  *
2990 1230                                     0298  * STEP 47
3000 1230                                     0299  *

3020 1230  P17103100P 08 1710  0 0 1000C 3 0 0301         MC     ZEROS,↑ECB           NO RESET INH SWITCH & BRANCH TO
3030 1240  U1SX000000 11 1380  5 0 0000  0 0 0302         BC     STAT40(5)            RESET PSB
3030 1250                                     0303  *
3040 1250                                     0304  * STEP 48
3050 1250                                     0305  *

3070 1250  P17343100P 08 1734  0 0 1000C 3 0 0307  STAT15 MC     OPCON,↑ECB           RESET INHIBIT SWITCH & SET OP-CODE
3080 1260  U1SX000000 11 1380  5 0 0000  0 0 0308         BC     STAT40(5)            BRANCH TO RESET PSB TO NORMAL
3090 1270                                     0309  *
3100 1270                                     0310  * STEP 49
3110 1270                                     0311  *

3130 1270  PPPOPW1704 14 0000C 0 2 1704  1 0 0313  STAT20 C      ↑BASE(1,2),N         IS PSB SET?
3140 1280                                     0314  *
3150 1280                                     0315  * STEP 50
3160 1280                                     0316  *

3180 1280  R1TX000000 11 1480  2 0 0000  0 0 0318         BC     STATEX(2)            STATUS NORMAL IS CC=2
3190 1290                                     0319  *
3200 1290                                     0320  * STEP 51
3210 1290                                     0321  *

3230 1290  P100P11818 08 1000C 0 0 1818  1 0 0323         MC     ↑ECB(1),ERR6A        ELSE PREPARE REPORT WITH INH SW
3240 1300  P000PW1824 08 0000C 0 2 1824  1 0 0324         MC     ↑BASE(1,2),ERR6B     AND PSB CODES
3250 1310                                     0325  *
3260 1310                                     0326  * STEP 52
3270 1310                                     0327  *
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

3290 1310  PQPOS11700 14 1003C 0 0 1700  1 0 0329          C      ↑ECB+3(1),A           ARE WE READING OR WRITING
3300 1320  R1SW000000 11 1370  2 0 0000  0 0 0330          BC     STAT35(2)             READING
3310 1330                                    0331  *
3320 1330                                    0332  * STEP 53
3330 1330                                    0333  *

3350 1330  V1VY151490 11 1691  6 0 1490  5 0 0335          BC     BCLREX+1(6),BCLEAR(5)  BRANCH TO CLEAR XMIT BFS SUBR
3360 1340                                    0336  *
3370 1340                                    0337  * STEP 54
3380 1340                                    0338  *

3400 1340  P170110PPY 08 1701  0 0 0009C 1 3 0340  STAT30 MC     D,↑B9(,3)             SET DLE-EOT CODE IN WRITE FLAG
3410 1350  P170110PPP 08 1701  0 0 0000C 1 3 0341         MC     D,↑BASE(,3)           SET XMIT BUFFER READY TO GO
3420 1360  U1SX000000 11 1380  5 0 0000  0 0 0342         BC     STAT40(5)             BRANCH TO RESET PSB
3430 1370                                    0343  *
3440 1370                                    0344  * STEP 55
3450 1370                                    0345  *

3470 1370  P17131100Q 08 1713  0 0 1001C 1 0 0347  STAT35 MC     K2,↑ECB+1             SET DLE-EOT CODE IN READ FLAG
3480 1380                                    0348  *
3490 1380                                    0349  * STEP 56
3500 1380                                    0350  *

3520 1380  P170410POP 08 1704  0 0 0000C 1 2 0352  STAT40 MC     N,↑BASE(,2)           RESET PSB TO NORMAL
3530 1390                                    0353  *
3540 1390                                    0354  * STEP 57
3550 1390                                    0355  *

3570 1390  U1TR000000 11 1420  5 0 0000  0 0 0357  STAT45 BC     STAT50(5)             BRANCH UNLESS REPORT SWITCH(VAR 0)
3580 1400  218R510100 01 1825  2 0 0100  1 0 0358         W      ERR7(2),100(1)        REPORT FIRST XMIT BUFFER
3590 1410  P17Q411390 09 1714  0 0 1390  1 0 0359         MN     K5,STAT45             RESET REPORT SWITCH
3600 1420  U1TU000000 11 1450  5 0 0000  0 0 0360  STAT50 BC     STAT55(5)             BRANCH UNLESS REPORT SWITCH(VAR=0)
3610 1430  219R510100 01 1925  2 0 0100  1 0 0361         W      ERR8(2),100(1)        REPORT 2ND XMIT BUFFER
3620 1440  P17Q411420 09 1714  0 0 1420  1 0 0362         MN     K5,STAT50             RESET REPORT SWITCH
3630 1450  017P330001 01 1703  0 0 0001  3 0 0363  STAT55 W      M(0),1(3)             CARRIAGE RETURN TO WORK STATION
3640 1460  018P310022 01 1803  0 0 0022  1 0 0364         W      ERR6(0),22(1)         REPORT INH SW AND PSB SETTINGS
3650 1470                                    0365  *
3660 1470                                    0366  * STEP 58
3670 1470                                    0367  *

3690 1470  PQW1011713 14 1710  0 0 1713  1 0 0369         C      ZEROS(1),K2           SET CC=1 FOR STATUS RESTART
3700 1480  U0PP000000 11 0000  5 0 0000  0 0 0370  STATEX BC     0(5)                  EXIT TO RETURN ADDRESS


3730 1490  PPPOPQQ702 14 0000C 0 3 1702  1 0 0373  BCLEAR C      ↑BASE(1,3),E          IS THE ERROR IN THE 1ST XMIT BUFF
3740 1500  R1UU000000 11 1550  2 0 0000  0 0 0374         BC     STAT60(2)             YES SO POINTER IS CORRECT
3750 1510  PQWQ940031 15 1719  0 0 0031  4 0 0375         X      BF2ADR,X3             ELSE LOOK AT THE OTHER BUFFER
3760 1520  PPPOPQQ702 14 0000C 0 3 1702  1 0 0376         C      ↑BASE(1,3),E          IS ERROR IN 2ND XMIT BUFFER
3770 1530  R1UU000000 11 1550  2 0 0000  0 0 0377         BC     STAT60(2)             YES SO POINTER NOW IS CORRECT
3780 1540  PQWQ940031 15 1719  0 0 0031  4 0 0378         X      BF2ADR,X3             ELSE RESTORE POINTER TO ORIG POSN
3790 1550  PPPOPQQ700 14 0000C 0 3 1700  1 0 0379  STAT60 C      ↑BASE(1,3),A          IS 1ST XMIT BUFF AVAILABLE?
3800 1560  U1VQ000000 11 1610  5 0 0000  0 0 0380         BC     STAT65(5)             YES
3810 1570  X002PPQ845 08 0020C 8 3 1845  0 0 0381         MC     ↑B10+10(80,3),ERR7A   SET UP REPORT
3820 1580  P17Q011390 09 1710  0 0 1390  1 0 0382         MN     ZEROS(1),STAT45       AND SET UP REPORT SWITCH
3830 1590  P170010PPP 08 1700  0 0 0000C 1 3 0383         MC     A,↑BASE(,3)           SET XMIT BUFF AVAIL AGAIN
3840 1600  P170610PPY 08 1706  0 0 0009C 1 3 0384         MC     V,↑B9(,3)             WRITE FLAG=NORMAL
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

3850 1610  PPPS141719 15 0031  0 0 1719  4 0 0385  STAT65 X      X3,BF2ADR            XCHANGE XMIT BUFF POINTERS
3860 1620  PPPOPQQ700 14 0000C 0 3 1700  1 0 0386         C      ↑BASE(1,3),A         IS 2ND XMIT BUFFER AVAILABLE
3870 1630  R1VX000000 11 1680  2 0 0000  0 0 0387         BC     STAT70(2)            YES
3880 1640  X002PPQ945 08 0020C 8 3 1945  0 0 0388         MC     ↑B10+10(80,3),ERR8A  SET UP REPORT
3890 1650  P17Q011420 09 1710  0 0 1420  1 0 0389         MN     ZEROS(1),STAT50      SET UP REPORT  SWITCH
3900 1660  P170610PPY 08 1706  0 0 0009C 1 3 0390         MC     V,↑B9(,3)            WRITE FLAG=NORMAL
3910 1670  P170010PPP 08 1700  0 0 0000C 1 3 0391         MC     A,↑BASE(,3)          BSB=AVAILABLE
3920 1680  PQWQ940031 15 1719  0 0 0031  4 0 0392  STAT70 X      BF2ADR,X3            XCHANGE BUFF POINTERS BACK AGAIN
3930 1690  UOPP000000 11 0000  5 0 0000  0 0 0393  BCLREX BC     0(5)                 EXIT FROM BUFF CLEAR SUBR
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

3950 1700                                    0395  *
3960 1700                                    0396  * LABELS USED BY UAP
3970 1700                                    0397  *



3990 1700  A               0001      0001    0399  A      DM     C1'A'
4000 1701  D               0001      0001    0400  D      DM     C1'D'
4010 1702  E               0001      0001    0401  E      DM     C1'E'
4020 1703  M               0001      0001    0402  M      DM     C1'M'
4030 1704  N               0001      0001    0403  N      DM     C1'N'
4040 1705  Q               0001      0001    0404  Q      DM     C1'Q'
4050 1706  V               0001      0001    0405  V      DM     C1'V'
4060 1707  W               0001      0001    0406  W      DM     C1'W'
4070 1708  X               0001      0001    0407  X      DM     C1'X'
4080 1709  ?               0001      0001    0408  QM     DM     C1'?'
4090 1710  000             0001      0003    0409  ZEROS  DM     C3'000'
4100 1713  2               0001      0001    0410  K2     DM     C1'2'
4110 1714  5               0001      0001    0411  K5     DM     C1'5'
4120 1715  0080            0001      0004    0412  KON80  DM     C4'0080'

4140 1719  160P            0001      0004    0414  BF2ADR DM     A'BUFF2'
4150 1723  190P            0001      0004    0415  BF4ADR DM     A'BUFF4'
4160 1727                  0001      0001    0416  FRSTWT DM     C1' '                 FIRST WRITE SWITCH
4170 1728  CALLNO          0001      0006    0417  CALLNO DM     C6'CALLNO'
4180 1734  OOE             0001      0003    0418  OPCON  DM     C3'OOE'
4190 1737                  0001      0001    0419  FUNC   DM     C1' '

4210 1738  INITIALISA      0001      0020    0421  ERR1   DM     C20'INITIALISATION ERROR'
4220 1758  CARD READE      0001      0021    0422  ERR2   DM     C21'CARD READER NOT READY'
4230 1779  NO CALL NO      0001      0014    0423  ERR3   DM     C14'NO CALL NO'
4240 1793  DIAL ERROR      0001      0010    0424  ERR4   DM     C10'DIAL ERROR'
4250 1803  INHIBIT SW      0001      0015    0425  ERR6   DM     C15'INHIBIT SWITCH '
4260 1818    PSB           0001      0006    0426  ERR6A  DM     C6'  PSB '
4270 1824                  0001      0001    0427  ERR6B  DM     C1' '
4280 1825  XMIT BUFF       0001      0020    0428  ERR7   DM     C20'XMIT BUFF OVERWRITE '
4290 1845                  0001      0080    0429  ERR7A  DM     C80' '
4300 1925  XMIT BUFF       0001      0020    0430  ERR8   DM     C20'XMIT BUFF OVERWRITE '
4310 1945                  0001      0080    0431  ERR8A  DM     C80' '
4320 2025  NO DATA FO      0001      0030    0432  ERR9   DM     C30'NO DATA FOLLOWING CALLNOW CARD'
4330 2055                  0001      0001    0433  SPACE  DM     C1' '

4350 2056  BCOD1           0001      0005    0435  MAIL   DM     C5'BCOD1'
4360 2061  150P            0001      0004    0436  BUFS   DM     A'BUFF1'
4370 2065  160P            0001      0004    0437         DM     A'BUFF2'
4380 2069  Z               0001      0009    0438         DM     C9'Z'
4390 2078  170P            0001      0004    0439         DM     A'BUFF3'
4400 2082  190P            0001      0004    0440         DM     A'BUFF4'
4410 2086  Z               0001      0009    0441         DM     C9'Z'
4420 2095  0082            0001      0004    0442         DM     C4'0082'
4430 2099                  0001      0020    0443         DM     C20' '
4440 2119  65CALLERHE      0001      0040    0444  IDEN   DM     C40'65CALLERHELLOZ'
4450 2159                  0001      0040    0445         DM     C40' '
```

```
SEQ. LOCN  INSTR/DATA UP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

4470 2199                                    0447  *
4480 2199                                    0448  *
4490 2199                                    0449  * LABELS IN COMMON REQUIRED FOR SCA ROUTINES
4500 2199                                    0450  *
4510 2199                                    0451  *


4530 0000C                                   0453         COMMON
4540 0000C               000P                0454         ORG    0
4550 0000C &&&---000     0007      0009      0455         DM     7C9'&&&---000'       DUMMY TO SHOW UP IN OBJECT DECK
4560 0063C               000P                0456         ORG    0
4570 0000C               0001      0001      0457  ↑BASE  DM     C1
4580 0001C               0001      0002      0458  ↑B1    DM     C2
4590 0003C               0001      0002      0459         DM     C2
4600 0005C               0001      0004      0460  ↑B5    DM     C4
4610 0009C               0001      0001      0461  ↑B9    DM     C1
4620 0010C               0001      0001      0462  ↑B10   DM     C1

4640 0011C               010Q                0464         ORG    101
4650 0101C               0000      0004      0465  ↑K101  DM     0C4

4670 0101C               056P                0467         ORG    560
4680 0560C               0000      0020      0468  ↑IMSG  DM     0C20                 IDLE MESSAGE
4690 0560C               0001      0020      0469  ↑STAT  DM     C20' '               PARTITION STATUS BYTES
4700 0580C A             0001      0005      0470  ↑MAIL  DM     C5'A'                MAIL-BOX=AVAILABLE AT LOAD TIME
4710 0585C               0001      0017      0471  ↑SEND  DM     C17                  +5 -TRANSMIT BUFFERS
4720 0602C               0001      0021      0472  ↑SINK  DM     C21                  +22-RECEIVE BUFFERS
4730 0623C               0001      0020      0473         DM     C20                  +43-DIAL DATA (NOT USED HERE)
4740 0643C               0001      0099      0474  ↑IDEN  DM     C99                  +63 ID DATA


4760 0742C                                   0476  * VARIABLE LOCATION REFERENCES

4780 0742C               100P                0478         ORG    1000
4790 1000C 0000000000    0001      0018      0479  ↑ECB   DM     C18'000000000000000000'  COMMUNICATIONS CONTROL BLOCK
4800 1018C               105P                0480         ORG    1050
4810 1050C               0000      0044      0481  ↑CWA   DM     0C44
4820 1050C A             0001      0080      0482  ↑IDLEB DM     C80'A'


4840 1130C               150P                0484         ORG    1500
4850 1500C A             0001      0020      0485  BUFF1  DM     C20'A'
4860 1520C               0001      0080      0486         DM     C80
4870 1600C A             0001      0020      0487  BUFF2  DM     C20'A'
4880 1620C               0001      0080      0488         DM     C80
4890 1700C A             0001      0020      0489  BUFF3  DM     C20'A'
4900 1720C               0001      0082      0490         DM     C82
4910 1802C               190P                0491         ORG    1900
4920 1900C A             0001      0020      0492  BUFF4  DM     C20'A'
4930 1920C               0001      0082      0493         DM     C82
```

```
4950 2002C                                   0495         END
```

```
PAGE 0001  11/24/71 CROSS REFERENCE LISTING FOR USER APPLICATION PROGRAM FOR DIAL-CALLING STATION

DEF.    SYMBOL   ..........................REFERENCES........................

0399    A        0032  0049  0061  0126  0214  0236  0237  0266  0329  0379
                 0383  0386  0391
0373    BCLEAR   0290  0335
0393    BCLREX   0290  0335
0414    BF2ADR   0171  0215  0375  0378  0385  0392
0415    BF4ADR   0271
0485    BUFF1    0014  0436
0487    BUFF2    0414  0437
0489    BUFF3    0010  0236  0439
0492    BUFF4    0237  0415  0440
0436    BUFS
0417    CALLNO   0094
0400    D        0166  0189  0194  0231  0340  0341
0043    DIAL00   0062
0061    DIAL05   0044
0068    DIAL10   0090  0137  0160  0209  0216  0218  0253
0070    DIAL15   0076  0080
0076    DIAL20   0068  0077
0085    DIAL25   0071  0100  0104
0100    DIAL30   0069  0101
0110    DIAL35   0095
0126    DIAL37   0137  0177  0224
0142    DIAL40   0127
0165    DIAL45   0147
0189    DIAL55   0116  0183
0194    DIAL60   0218
0217    DIAL65   0197
0223    DIAL70   0195
0230    DIAL75   0153
0242    DIAL80   0253  0272
0258    DIAL85   0243
0401    E        0043  0196  0242  0373  0376
0421    ERR1     0055
0422    ERR2     0079
0423    ERR3     0103
0424    ERR4     0203
0425    ERR6     0364
0426    ERR6A    0284  0323
0427    ERR6B    0285  0324
0428    ERR7     0358
0429    ERR7A    0381
0430    ERR8     0361
0431    ERR8A    0388
0432    ERR9     0159
0416    FRSTWT   0121  0152  0176  0182
0419    FUNC     0110  0115  0208  0223
0444    IDEN
0410    K2       0347  0369
0411    K5       0077  0101  0121  0359  0362
0412    KON80    0165
0402    M        0054  0078  0102  0158  0202  0363
0435    MAIL     0038
0403    N        0027  0313  0352
0418    OPCON    0307
0404    Q        0230
0408    QM       0070
```

```
PAGE 0002  11/24/71 CRUSS RFFERENCE LISTING FOR USER APPLICATION PROGRAM FOR DIAL-CALLING STATION

DEF.    SYMBOL  ........................REFERENCES........................

0433    SPACE   0260
0307    STAT15  0296
0313    STAT20  0279
0340    STAT30
0347    STAT35  0330
0352    STAT40  0302  0308  0342
0357    STAT45  0359  0382
0360    STAT50  0357  0362  0389
0363    STAT55  0360
0379    STAT60  0374  0377
0385    STAT65  0380
0392    STAT70  0387
0370    STATEX  0132  0217  0248  0318
0278    STATUS  0132  0217  0248
0405    V       0384  0390
0406    W       0115  0208  0223  0295
0407    X
0010    X1      0271
0012    X2      0026
0014    X3      0171  0215  0375  0378  0385  0392
0409    ZEROS   0020  0068  0069  0152  0176  0182  0278  0301  0369  0382
                0389
0458    ↑B1
0462    ↑B10    0142  0259  0260  0261  0261  0381  0388
0460    ↑B5     0165  0258
0461    ↑B9     0230  0340  0384  0390
0457    ↑BASE   0027  0126  0166  0214  0231  0242  0266  0285  0313  0324
                0341  0352  0373  0376  0379  0383  0386  0391
0481    ↑CWA    0085  0094  0110  0189  0196
0479    ↑ECB    0020  0026  0043  0194  0278  0284  0295  0301  0307  0323
                0329  0347
0474    ↑IDEN
0482    ↑IDLEB
0468    ↑IMSG
0465    ↑K101
0470    ↑MAIL   0032  0038  0049  0061
0471    ↑SEND
0472    ↑SINK
0469    ↑STAT   0012
```

APPENDIX B

LISTING OF SAMPLE DIAL-CALLING UAP PRESENTED IN SECTION 6

```
PAGE 0001  11/24/71  (4-02)      LABEL TABLE FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

 LABEL     LOC  LNTH INDX            MESSAGES

 A        1142   01    0
 BUFF1    1500C  20    0
 BUFF2    1600C  20    0
 CARRET   1712C  01    0
 D        1143   01    0
 DIAL00   0370   10    0
 DIAL05   0440   10    0
 E        1144   01    0
 ERR1     1159   20    0
 ERR3     1179   17    0
 ERR3A    1196   01    0
 ERR4     1197   17    0
 ERR4A    1214   01    0
 IDEN     1093   40    0          UNUSED
 KON80    1155   04    0
 M        1712C  01    0
 MAIL     1030   05    0
 N        1145   01    0
 ONE      1153   01    0
 Q        1146   01    0
 QM       1141   01    0
 READ     0460   10    0
 READ05   0470   10    0
 READ10   0500   10    0
 READ15   0540   10    0
 SPACE    1140   01    0
 STAT     1133   04    0          UNUSED
 STAT05   0800   10    0
 STAT10   0820   10    0
 STAT15   0860   10    0
 STAT20   0880   10    0
 STAT25   0990   10    0
 STATEX   0870   10    0
 STATUS   0740   10    0
 TCON     1137   03    0
 TWO      1154   01    0
 V        1147   01    0
 W        1148   01    0
 WRITE    0570   10    0
 WRTE05   0610   10    0
 WRTE10   0620   10    0
 WRTE15   0670   10    0
 WRTE20   0710   10    0
 X        1149   01    0
 X1       0011   04    0          UNUSED
 X2       0021   04    0
 X3       0031   04    0          UNUSED
 ZEROS    1150   03    0
 ↑B1      0001C  02    0          UNUSED
 ↑B10     0010C  01    0
 ↑B5      0005C  04    0
 ↑B9      0009C  01    0
 ↑BASE    0000C  01    0
 ↑ECB     1000C  18    0
 ↑IDEN    0643C  99    0          UNUSED
 ↑IDLEB   1018C  44    0          UNUSED
```

```
PAGE 0002  11/24/71  (4-02)      LABEL TABLE FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

 LABEL     LOC  LNTH INDX         MESSAGES

 ↑IMSG    1062C  20    0       UNUSED
 ↑K101    0101C  04    0       UNUSED
 ↑MAIL    0580C  05    0
 ↑SEND    0585C  17    0       UNUSED
 ↑SINK    0602C  21    0       UNUSED
 ↑STAT    0560C  20    0
 62 LABELS
```

```
PAGE 0001  11/24/71  (4=02) PROGRAM LISTING FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0020 0000                                    0002  *
0030 0000                                    0003  *
0040 0000                                    0004  * SAMPLE USER APPLICATION
0050 0000                                    0005  *
0060 0000                                    0006  *
0070 0000                                    0007         NORMAL
0080 0000                 0011               0008         ORG    0011
0090 0011  150P           0001      0004     0009  X1     DM     A'BUFF1'
0100 0015                 0021               0010         ORG    0021
0110 0021  056P           0001      0004     0011  X2     DM     A'↑STAT'              ADDRESS OF PSB'S
0120 0025                 0031               0012         ORG    0031
0130 0031  160P           0001      0004     0013  X3     DM     A'BUFF2'


0150 0035                 0300               0015         ORG    0300
0160 0300                                    0016  *
0170 0300                                    0017  * STEP 1
0180 0300                                    0018  *

0200 0300  PQPOV21150 14 1006C 0 0 1150  2 0 0020         C      ↑ECB+6(2),ZEROS       IS SCA ACTIVE?
0210 0310  SOSR080300 11 0320  3 0 0300  8 0 0021         BC     *+10(3),*=10(8)       IF NOT WAIT FOR IT
0220 0320                                    0022  *
0230 0320                                    0023  * STEP 2
0240 0320                                    0024  *

0260 0320  2QPPV40021 07 1006C 2 0 0021  4 0 0026         S      ↑ECB+6(2),X2          COMPUTE SCA'S PSB AND SET IN X2
0270 0330  P114510POP 08 1145  0 0 0000C 1, 2 0027        MC     N,↑BASE(,2)           SET SCA'S PSB TO NORMAL
0280 0340                                    0028  *
0290 0340                                    0029  * STEP 3
0300 0340                                    0030  *

0320 0340  PPU8P11142 14 0580C 0 0 1142  1 0 0032         C      ↑MAIL(1),A            IS MAILBOX AVAILABLE?
0330 0350  ROSV080340 11 0360  2 0 0340  8 0 0033         BC     *+10(2),*=10(8)       IF NOT WAIT FOR IT
0340 0360                                    0034  *
0350 0360                                    0035  * STEP 4
0360 0360                                    0036  *

0380 0360  P10300058P 08 1030  0 0 0580C 0 0 0038         MC     MAIL(100),↑MAIL       MOVE INIT RECORD TO MAIL=BOX
0390 0370                                    0039  *
0400 0370                                    0040  * STEP 5
0410 0370                                    0041  *

0430 0370  PQPOP11144 14 1000C 0 0 1144  1 0 0043  DIAL00 C      ↑ECB(1),E             WAS THERE INIT ERROR?
0440 0380  ROSY050440 11 0390  2 0 0440  5 0 0044         BC     *+10(2),DIAL05(5)     IF NOT BRANCH
0450 0390                                    0045  *
0460 0390                                    0046  * STEP 6
0470 0390                                    0047  *

0490 0390  P11421058P 08 1142  0 0 0580C 1 0 0049         MC     A,↑MAIL               RESET MAIL=BOX AVAILABLE
0500 0400  P11501100P 08 1150  0 0 1000C 1 0 0050         MC     ZEROS(1),↑ECB         AND INHIBIT SWITCH
0510 0410                                    0051  *
0520 0410                                    0052  * STEP 7
0530 0410                                    0053  *

0550 0410  017QR30001 01 1712C 0 0 0001  3 0 0055         W      M(0),1(3)             CARRIAGE RETURN TO WORK STATION
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0560 0420  011U910020 01 1159  0 0 0020  1 0 0056         W      ERR1(0),20(1)        REPORT ERROR
0570 0430  U9YY000000 11 9990  5 0 0000  0 0 0057         BC     9990(5)              BLOW UP
0580 0440                                    0058  *
0590 0440                                    0059  * STEP 8
0600 0440                                    0060  *

0620 0440  PPU8P11142 14 0580C 0 0 1142  1 0 0062  DIAL05 C      ↑MAIL(1),A           IS INIT COMPLETE?
0630 0450  ROTV080370 11 0460  2 0 0370  8 0 0063         BC     *+10(2),DIAL00(8)    YES 60 ON,ELSE WAIT FOR INIT
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0650 0460                                    0065  * INITIALISATION HAS BEEN SUCCESSFULLY COMPLETED
0660 0460                                    0066  *
0670 0460                                    0067  * STEP 9
0680 0460                                    0068  *
0690 0460  P114210PPP 08 1142  0 0 0000C 1 3 0069  READ   MC     A,↑BASE(,3)          SET READ BUFFER AVAILABLE
0700 0470                                    0070  *
0710 0470                                    0071  * STEP 10
0720 0470                                    0072  *
0730 0470  PPPOPQQ144 14 0000C 0 3 1144  1 0 0073  READ05 C      ↑BASE(1,3),E         IS READ BUFFER FULL
0740 0480  ROTY050540 11 0490  2 0 0540  5 0 0074         BC     *+10(2),READ15(5)    IF NOT BRANCH
0750 0490                                    0075  *
0760 0490                                    0076  * STEP 11
0770 0490                                    0077  *
0780 0490  POOPUTP506 09 0005C 0 3 0506  4 0 0078         MN     ↑B5(4,3),READ10+6    SET BUFF LENGTH IN PRINT INSTRN.
0790 0500  200RPQP100 01 0020C 2 3 0100  1 0 0079  READ10 W      ↑B10+10(2,3),100(1)  PRINT READ BUFFER
0800 0510  P114010PRP 08 1140  0 0 0020C 1 3 0080         MC     SPACE,↑B10+10(,3)          CLEAR OUT READ BUFFER
0810 0520  Y002PYPPRQ 08 0020C 9 3 0021C 9 3 0081         MC     ↑B10+10(99,3),↑B10+11(,3)
0820 0530  UOTV000000 11 0460  5 0 0000  0 0 0082         BC     READ(5)
0830 0540                                    0083  *
0840 0540                                    0084  * STEP 12
0850 0540                                    0085  *
0860 0540  VOXW150740 11 0871  6 0 0740  5 0 0086  READ15 BC     STATEX+1(6),STATUS(5) BRANCH & LINK TO STATUS ROUTINE
0870 0550                                    0087  *
0880 0550                                    0088  * STEP 13
0890 0550                                    0089  *
0900 0550  QOTV030570 11 0460  1 0 0570  3 0 0090         BC     READ(1),WRITE(3)     ERROR OR EOT
0910 0560  XOTW000000 11 0470  8 0 0000  0 0 0091         BC     READ05(8)            NORMAL
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

0930 0570                                    0093  * EOT IS RECOGNISED INVITING THIS COMPUTER TO SEND
0940 0570                                    0094  *
0950 0570                                    0095  * STEP 14
0960 0570                                    0096  *
0970 0570  111T130001 01 1141  1 0 0001  3 0 0097  WRITE  W      QM(1),1(3)          IS CARD READER READY IE ANYTHING
0980 0580                                    0098  *                                 TO SEND
0990 0580  TOUY050610 11 0590  4 0 0610  5 0 0099         BC     *+10(4),WRTE05(5)   IF SO BRANCH TO STEP 16
1000 0590                                    0100  *
1010 0590                                    0101  * STEP 15
1020 0590                                    0102  *                                 ELSE SET INH SWITCH ZERO, READ
1030 0590  P1150310OP 08 1137  0 0 1000C 3 0 0103         MC     ZERO,4ECB           FLAG ZERO & OP CODE D
1040 0600  UOTV000000 11 0710  5 0 0000  0 0 0104         BC     READ(5)             GO TO STEP 22
1050 0610                                    0105  * THERE IS SOMETHING TO SEND
1060 0610                                    0106  *
1070 0610                                    0107  * STEP 16
1080 0610                                    0108  *
1090 0610  P113731OOP 08 1137  0 0 1000C 3 0 0109  WRTE05 MC     TCON,†ECB
1100 0620                                    0110  *    RESETS INH SW& RD FG, OP CD=D
1110 0620                                    0111  *
1120 0620                                    0112  * STEP 17
1130 0620                                    0113  *
1140 0620  1002P1P080 00 0020C 1 1 0080  1 0 0114  WRTE10 R      †B10+10(1,1),80(1)  READ CARD FROM CD-READER TO XMITBF
1150 0630  SOWQ010620 11 0710  3 0 0620  1 0 0115         BC     WRTE20(3),*-10(1)   FLAG INDICATES NO MORE DATA-GO TO
1160 0640                                    0116  *                                 STEP 22. TRY AGAIN ON ERROR
1170 0640  P114710OPY 08 1147  0 0 0009C 1 1 0117         MC     V,†B9(,1)           WRITE FLAG = NORMAL
1180 0650  P115540OPU 08 1155  0 0 0005C 4 1 0118         MC     KON80,†B5(,1)       CHAR COUNT IN BUFF PREF 5-8 = 80
1190 0660  P114310OPP 08 1143  0 0 0000C 1 1 0119         MC     D,†BASE(,1)         D IN BSB TO POST BUFFER FULL
1200 0670                                    0120  *
1210 0670                                    0121  * STEP 18
1220 0670                                    0122  *
1230 0670  PPPOP1Q142 14 0000C 0 1 1142  1 0 0123  WRTE15 C      †BASE(1,1),A        HAS TRANSMIT BUFFER GONE O.K.
1240 0680  ROVR000000 11 0620  2 0 0000  0 0 0124         BC     WRTE10(2)           IF SO GO BACK FOR MORE DATA AT
1250 0690                                    0125  *                                 STEP 17. NOTE.READ COMMENTS AT
1260 0690                                    0126  *                                 STEP 21
1270 0690                                    0127  * STEP 19                         CARD READER
1280 0690                                    0128  *
1290 0690  VOXW150740 11 0871  6 0 0740  5 0 0129         BC     STATEX+1(6),STATUS(5) BRANCH & LINK TO STATUS SUBROUTIN
1300 0700                                    0130  *
1310 0700                                    0131  * STEP 20
1320 0700                                    0132  *
1330 0700  QOTV080670 11 0460  1 0 0670  8 0 0133         BC     READ(1),WRTE15(8)   ERROR OR NORMAL
1340 0710                                    0134  *
1350 0710                                    0135  * STEP 21                         NOT APPLICABLE HERE AS FLAG FOR NO
1360 0710                                    0136  *                                 MORE DATA TO SEND RECEIVED AT STEP
1370 0710                                    0137  * NO MORE DATA TO SEND            17 FROM CARD READER
1380 0710                                    0138  * STEP 22
1390 0710                                    0139  *
1400 0710  P114610OPY 08 1146  0 0 0009C 1 1 0140  WRTE20 MC     Q,†B9(,1)           EOT IN WRITE FLAG
1410 0720  P114310OPP 08 1143  0 0 0000C 1 1 0141         MC     D,†BASE(,1)         D IN BSB
1420 0730  UOTV000000 11 0460  5 0 0000  0 0 0142         BC     READ(5)             GO BACK TO STEP 9
```

```
SEQ. LOCN  INSTR/DATA UP  A/R   L I  B/S   L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

1440 0740                                      0144  *
1450 0740                                      0145  * STATUS SUBROUTINE PERFORMED :    1)WHILST WAITING TO BE CALLED
1460 0740                                      0146  *                                 2)WHILST READING DATA
1470 0740                                      0147  *                                 3)WHILST WRITING DATA
1480 0740                                      0148  *
1490 0740                                      0149  * THREE POSSIBILITIES FOR EXIT:    1)NORMAL-CONDITION CODE =2
1500 0740                                      0150  *                                 2)EOT RECEIVED WHILST READING
1510 0740                                      0151  *                                  CONDITION CODE =3
1520 0740                                      0152  *                                 3)ERROR RECEIVED,CONDITION CODE =1


1540 0740                                      0154  *
1550 0740                                      0155  * STEP 23
1560 0740                                      0156  *

1580 0740  PQPOP11150 14 1000C 0 0 1150   1 0 0158  STATUS C      ↑ECB(1),ZEROS         IS INHIBIT SWITCH SET?
1590 0750  ROXX000000 11 0880  2 0 0000   0 0 0159         BC     STAT20(2)             IF NOT BRANCH
1600 0760                                      0160  *
1610 0760                                      0161  * STEP 24
1620 0760                                      0162  *

1640 0760  PQPOP11148 14 1000C 0 0 1148   1 0 0164         C      ↑ECB(1),W             YES SO IS IT EOT RECEIVED?(X OR W)
1650 0770  ROXP000000 11 0800  2 0 0000   0 0 0165         BC     STAT05(2)
1660 0780  PQPOP11149 14 1000C 0 0 1149   1 0 0166         C      ↑ECB(1),X
1670 0790  ROXP050820 11 0800  2 0 0820   5 0 0167         BC     *+10(2),STAT10(5)     IF NOT IT'S AN ERROR SO BRANCH
1680 0800                                      0168  *
1690 0800                                      0169  * STEP 25
1700 0800                                      0170  *

1720 0800  PQQ5311150 14 1153  0 0 1150   1 0 0172  STAT05 C      ONE,ZEROS             POST STATUS=EOT RECVD,CC=3
1730 0810  UOXW000000 11 0870  5 0 0000   0 0 0173         BC     STATEX(5)             EXIT TO CALLING ADDRESS
1740 0820                                      0174  *
1750 0820                                      0175  * STEP 26
1760 0820                                      0176  *

1780 0820  P100P11214 08 1000C 0 0 1214   1 0 0178  STAT10 MC     ↑ECB(1),ERR4A         MOVE INH SW CODE TO ERROR REPORT
1790 0830  017QR30001 01 1712C 0 0 0001   3 0 0179         W      CARRET(0),1(3)        CARRIAGE RETURN TO LINE PRINTER
1800 0840  011Y710018 01 1197  0 0 0018   1 0 0180         W      ERR4(0),18(1)         REPORT ERROR ON WORK STATION
1810 0850                                      0181  *
1820 0850                                      0182  * STEP 27
1830 0850                                      0183  *

1850 0850  P11503100P 08 1150  0 0 1000C  3 0 0185         MC     ZEROS,↑ECB            RESET INH SWITCH
1860 0860                                      0186  *
1870 0860                                      0187  * STEP 28
1880 0860                                      0188  *

1900 0860  PQQ5011153 14 1150  0 0 1153   1 0 0190  STAT15 C      ZEROS(1),ONE          POST STATUS=ERROR CC=1
1910 0870  UOPP000000 11 0000  5 0 0000   0 0 0191  STATEX BC     0(5)                  EXIT TO CALLING ROUTINE
1920 0880                                      0192  *
1930 0880                                      0193  * STEP 29
1940 0880                                      0194  *

1960 0880  PPPOPQ1145 14 0000C 0 2 1145   1 0 0196  STAT20 C      ↑BASE(1,2),N          IS PSB SET TO OTHER THAN NORMAL?
1970 0890                                      0197  *
```

```
PAGE 0006  11/24/71  (4-02) PROGRAM LISTING FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

1980 0890                                   0198  * STEP 30
1990 0890                                   0199  *

2010 0890  ROXW000000 11 0870  2 0 0000  0 0 0201         BC     STATEX(2)            IF NOT EXIT STATUS=NORMAL CC=2
2020 0900                                   0202  *
2030 0900                                   0203  * STEP 31
2040 0900                                   0204  *

2060 0900  P000PQ1196 08 0000C 0 2 1196  1 0 0206         MC     ↑BASE(1,2),ERR3A     MOVE PSB TO ERROR REPORT
2070 0910  017QR30001 01 1712C 0 0 0001  3 0 0207         W      CARRET(0),1(3)       CARRIAGE RETURN TO WORK STATION
2080 0920  011W910018 01 1179  0 0 0018  1 0 0208         W      ERR3(0),18(1)        REPORT ERROR
2090 0930                                   0209  *
2100 0930                                   0210  * STEP 32
2110 0930                                   0211  *

2130 0930  P114510POP 08 1145  0 0 0000C 1 2 0213         MC     N,↑BASE(,2)          RESET PSB TO NORMAL
2140 0940                                   0214  *
2150 0940                                   0215  * STEP 33
2160 0940                                   0216  *                                 STATUS EOT

2180 0940  PQPOR11143 14 1002C 0 0 1143  1 0 0218         C      ↑ECB+2(1),D          ARE WE WRITING
2190 0950  ROYV050990 11 0960  2 0 0990  5 0 0219         BC     *+10(2),STAT25(5)    IF NOT BRANCH
2200 0960  P1143100PY 08 1143  0 0 0009C 1 1 0220         MC     D,↑B9(,1)            DLE-EOT IN WRITE FLAG FOR HANG UP
2210 0970  P1143100PP 08 1143  0 0 0000C 1 1 0221         MC     D,↑BASE(,1)          POST TRANSMIT BUFFER FULL
2220 0980  U0XV000000 11 0860  5 0 0000  0 0 0222         BC     STAT15(5)            BRANCH TO SET ERROR STATUS
2230 0990  PQP0S11142 14 1003C 0 0 1142  1 0 0223  STAT25 C      ↑ECB+3(1),A          ARE WE READING?
2240 1000  R1PQ050860 11 1010  2 0 0860  5 0 0224         BC     *+10(2),STAT15(5)    IF NOT,SCA NEVER LEFT HAND-SHAKING
2250 1010                                   0225  *                                 SO NO NEED TO SEND DLE-EOT
2260 1010  P11541100Q 08 1154  0 0 1001C 1 0 0226         MC     TWO,↑ECB+1           SEND DLE-EOT IN READ FLAG
2270 1020  U0XV000000 11 0860  5 0 0000  0 0 0227         BC     STAT15(5)            BRANCH TO SET ERROR STATUS
```

```
SEQ. LOCN  INSTR/DATA OP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

2290 1030                                    0229  *
2300 1030                                    0230  * CONSTANTS FOR DRIVER
2310 1030                                    0231  *                                       MAIL IS INITIALISATION RECORD
2320 1030  BCOD1           0001       0005   0232  MAIL   DM      C5'BCOD1'
2330 1035  150P            0001       0004   0233         DM      A'BUFF1'                 TRANSMIT BUFFER ADDRESS
2340 1039  Z               0001       0013   0234         DM      C13'Z'
2350 1052  160P            0001       0004   0235         DM      A'BUFF2'                 RECEIVE BUFFER ADDRESS
2360 1056  Z               0001       0013   0236         DM      C13'Z'
2370 1069  0082            0001       0004   0237         DM      C4'0082'                 MAX NO OF CHARS TO BE READ
2380 1073  076353651       0001       0020   0238         DM      C20'076353651'           DIAL LENGTH=/DIGITS & NO ITSELF
2390 1093  56HELLO CA      0001       0040   0239  IDEN   DM      C40'56HELLO CALLERZ'

2410 1133  056P            0001       0004   0241  STAT   DM      A'↑STAT'                 PARTITION STATUS BYTE
2420 1137  00D             0001       0003   0242  TCON   DM      C3'00D'                  CONSTANT FOR INH SW,RD FG & OP=CD
2430 1140                  0001       0001   0243  SPACE  DM      C1' '
2440 1141  ?               0001       0001   0244  QM     DM      C1'?'
2450 1142  A               0001       0001   0245  A      DM      C1'A'
2460 1143  D               0001       0001   0246  D      DM      C1'D'
2470 1144  E               0001       0001   0247  E      DM      C1'E'
2480 1145  N               0001       0001   0248  N      DM      C1'N'
2490 1146  Q               0001       0001   0249  Q      DM      C1'Q'
2500 1147  V               0001       0001   0250  V      DM      C1'V'
2510 1148  W               0001       0001   0251  W      DM      C1'W'
2520 1149  X               0001       0001   0252  X      DM      C1'X'
2530 1150  000             0001       0003   0253  ZEROS  DM      C3'000'
2540 1153  1               0001       0001   0254  ONE    DM      C1'1'
2550 1154  2               0001       0001   0255  TWO    DM      C1'2'
2560 1155  0080            0001       0004   0256  KON80  DM      C4'0080'

2580 1159  INITIALISA      0001       0020   0258  ERR1   DM      C20'INITIALISATION ERROR'
2590 1179  TEMPORARY       0001       0017   0259  ERR3   DM      C17'TEMPORARY ERROR '
2600 1196                  0001       0001   0260  ERR3A  DM      C1' '
2610 1197  PERMANENT       0001       0017   0261  ERR4   DM      C17'PERMANENT ERROR '
2620 1214                  0001       0001   0262  ERR4A  DM      C1' '
```

```
SEQ. LOCN  INSTR/DATA UP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

2640 1215                                    0264  *
2650 1215                                    0265  *
2660 1215                                    0266  * LABELS IN COMMON REQUIRED FOR SCA ROUTINES
2670 1215                                    0267  *
2680 1215                                    0268  *


2700 1215                                    0270  * FIXED LOCATION REFERENCES

2720 0000C                                   0272         COMMON
2730 0000C                000P               0273         ORG    0
2740 0000C &              0063      0001     0274         DM     63C1'&'             DUMMY FOR FINDING SCA IN OBJECT DK
2750 0063C                000P               0275         ORG    0                   INDEXED ADDRESSES
2760 0000C                0001      0001     0276  †BASE  DM     C1                  BASE - STATUS
2770 0001C                0001      0002     0277  †B1    DM     C2                  +1 - TO/FROM
2780 0003C                0001      0002     0278         DM     C2
2790 0005C                0001      0004     0279  †B5    DM     C4                  +5 - LENGTH
2800 0009C                0001      0001     0280  †B9    DM     C1                  +9 - TRANSMIT CODE
2810 0010C                0001      0001     0281  †B10   DM     C1

2830 0011C                010Q               0283         ORG    101
2840 0101C                0000      0004     0284  †K101  DM     0C4                 B-REGISTER CONSTANT


2860 0101C                                   0286  * SYSTEM MAILBOX - LABELS USED IN INITIALISATION

2880 0101C                058P               0288         ORG    580
2890 0580C A0000          0001      0005     0289  †MAIL  DM     C5'A0000'           BASE - STATUS + TO/FROM
2900 0585C                0001      0017     0290  †SEND  DM     C17                 +5  - TRANSMIT BUFFERS
2910 0602C                0001      0021     0291  †SINK  DM     C21                 +22 - RECEIVE BUFFERS
2920 0623C                0001      0020     0292         DM     C20                 +43 - DIAL DATA -NOT USED HERE
2930 0643C                0001      0099     0293  †IDEN  DM     C99                 +63 - IDENTIFICATION DATA


2950 0742C                                   0295  * VARIABLE LOCATION REFERENCES

2970 0742C                056P               0297         ORG    560
2980 0560C                0001      0020     0298  †STAT  DM     C20' '              PARTITION STATUS BYTES

3000 0580C                100P               0300         ORG    1000
3010 1000C 0000000000     0001      0018     0301  †ECB   DM     C18'000000000000000000' COMMUNICATIONS CONTROL BLOCK

3030 1018C                101X               0303         ORG    1018
3040 1018C                0001      0044     0304  †IDLEB DM     C44                 IDLE MESSAGE BUFFER
3050 1062C                0000      0020     0305  †IMSG  DM     0C20                FIELD FOR IDLE MESSAGE

3070 1062C                150P               0307         ORG    1500
3080 1500C A              0001      0020     0308  BUFF1  DM     C20'A'
3090 1520C                0001      0080     0309         DM     C80
3100 1600C A              0001      0020     0310  BUFF2  DM     C20'A'
3110 1620C                0001      0082     0311         DM     C82
3120 1702C                171R               0312         ORG    *+10
```

```
SEQ. LOCN  INSTR/DATA UP  A/R  L I  B/S  L I LINE  LABEL  OPCODE OPERAND(S) AND/OR COMMENTS

     1712C               0000       0001      0313  M      DM     0C
3130 1712C M             0001       0001      0314  CARRET DM     C1'M'             CARRIAGE RETURN CONTROL CHAR
3140 1713C                                    0315         END
```

```
PAGE 0001  11/24/71 CROSS REFERENCE LISTING FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

DEF.    SYMBOL  ........................REFERENCES......................

0245    A       0032  0049  0062  0069  0123  0223
0308    BUFF1   0009  0233
0310    BUFF2   0013  0235
0314    CARRET  0179  0207
0246    D       0119  0141  0218  0220  0221
0043    DIAL00  0063
0062    DIAL05  0044
0247    E       0043  0073
0258    ERR1    0056
0259    ERR3    0208
0260    ERR3A   0206
0261    ERR4    0180
0262    ERR4A   0178
0239    IDEN
0256    KON80   0118
0313    M       0055
0232    MAIL    0038
0248    N       0027  0196  0213
0254    ONE     0172  0190
0249    Q       0140
0244    QM      0097
0069    READ    0082  0090  0133  0142
0073    READ05  0091
0079    READ10  0078
0086    READ15  0074
0243    SPACE   0080
0241    STAT
0172    STAT05  0165
0178    STAT10  0167
0190    STAT15  0222  0224  0227
0196    STAT20  0159
0223    STAT25  0219
0191    STATEX  0086  0129  0173  0201
0158    STATUS  0086  0129
0242    TCON    0103  0109
0255    TWO     0226
0250    V       0117
0251    W       0164
0097    WRITE   0090
0109    WRTE05  0099
0114    WRTE10  0124
0123    WRTE15  0133
0140    WRTE20  0104  0115
0252    X       0166
0009    X1
0011    X2      0026
0013    X3
0253    ZEROS   0020  0050  0158  0172  0185  0190
0277    ↑B1
0281    ↑B10    0079  0080  0081  0081  0114
0279    ↑B5     0078  0118
0280    ↑B9     0117  0140  0220
0276    ↑BASE   0027  0069  0073  0119  0123  0141  0196  0206  0213  0221
0301    ↑ECB    0020  0026  0043  0050  0103  0109  0158  0164  0166  0178
                0185  0218  0223  0226
0293    ↑IDEN
```

```
PAGE 0002  11/24/71 CROSS REFERENCE LISTING FOR USER APPLICATION PROGRAM FOR DIAL ANSWER

DEF.    SYMBOL   .........................REFERENCES.......................

0304    ↑IDLEB
0305    ↑IMSG
0284    ↑K101
0289    ↑MAIL    0032  0038  0049  0062
0290    ↑SEND
0291    ↑SINK
0298    ↑STAT    0011  0241
```

SINGER
FRIDEN DIVISION
CONTROL NO. C218AA