HEWLETT-PACKARD CO.

NOTE: This page provides a running history of changes for a multi-page drawing which cannot conveniently be re-issued completely after each change. When making a change, list for each page all before-and-after numbers (within reason; use judgement, and use "extensive" revision note if loss of past history is tolerable, or retype complete page) and associate with each a symbol made up of the change letter and a serial subscript to appear here and on the page involved (there enclosed in a circle, triangle, or other attention-getting outline).

Ltr	REVISIONS	DATE	INITIALS
B	RETYPED. PCO 40-12534. JS	8-3-82	B. Hayton
C	REPLACED PAGE 26 PER PCO 40-13419	2-2-83	

Model No. 98629 ERS

Stock No.

Description

By ALAN NELSON

Supersedes

Date 8-02-82

Sheet No. 1 of 29

Drawing No. A-98629-90303-1

98629-90303-1: Firmware External Reference Specification

PURPOSE

The purpose of this document is to describe what the firmware in the 98629A Resource Management Interface does.

SCOPE

This document is not meant to completely describe the firmware in the 98629A. The firmware IRS and source code are necessary to completely understand and support this code. However, this document will describe in some detail what the operation of the firmware is.

TABLE OF CONTENTS

98629A ERS -- INTRODUCTION

I. Introduction

This document is being written to describe the THEODORE interface from an external point of view. First of all, what is THEODORE? THEODORE is the interface card used to connect both the 9826 desktop computer and the Shared Resource Manager to the 98028 Resource Management Multiplexer via special (up to 60 metre long) cables. It is a 98628 interface card with a different ROM and an extra RAM chip installed, and different line receiver terminator circuitry. This document is not a hardware ERS; the hardware for THEODORE is described in detail in the 98628/98629 hardware ERS, available from Manufacturing Specs, under A-5955-6582-1.

Other related documents:

1818-1739-1	ROM maintenance/generation
A-98629-90301-1	98629A Firmware Reproduction Documentation
A-98629-90302-1	98629A Firmware IRS
C-5957-4326-1	GANGLINK Link Access Procedure Drawing

98629A ERS -- DESIGN OBJECTIVES

II. Design Objectives

The must objectives for the firmware in THEODORE are these:

A. The firmware must be produced in a timely manner.

B. The firmware must support both the 9816/26/36 connection to the Shared Resource Manager, and the Shared Resource Manager's connection to desktop computers.

C. The firmware in the interface must conform to the standards required by the "intelligent" interface drivers.

D. The firmware for the interface must fit into 8K bytes of code.

E. There must be a self-test feature in the firmware.

F. There must be a Link Access Protocol which will recover from lost packets, CRC errors, Rx overruns/overflows, and missing clocks.

The "high want" objectives are these:

A. High performance is important; consequently, every effort will be made to streamline the code in time-critical areas.

B. A provision for downloading code to the interface is a high want.

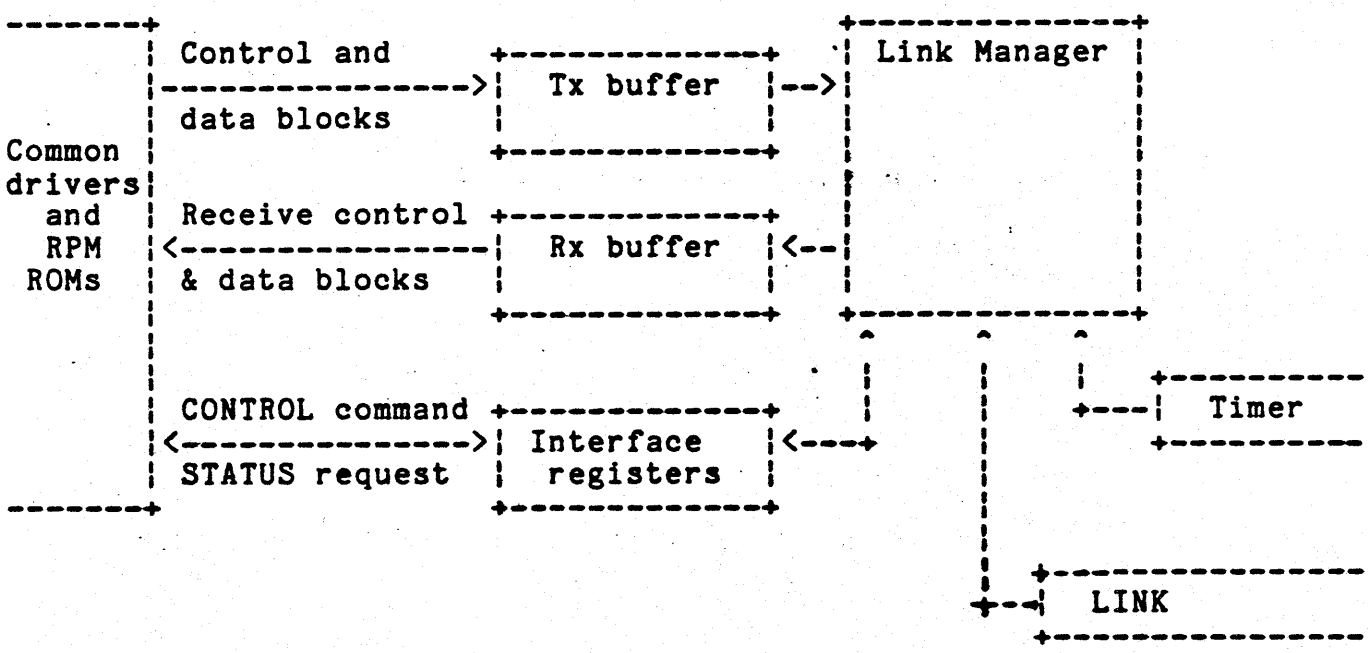
C. Detection of fault conditions and possible problems is desirable.

D. SA testing of the interface is desirable.

98629A ERS -- BLOCK DIAGRAM

III. Block Diagram

This interface card is a serial, synchronous bit oriented protocol card. It interfaces the 9826, 9836, or 9816 (referred from here on as the mainframe) to the Shared Resource Manager, and vice versa.



			MODEL	STK NO	
			98629 ERS		
			BY Alan Nelson	DATE 7-21-82	
	SEE PAGE 1 FOR REV.		APPD B. Hankton	SHEET NO 5	OF 29
LTR	PC NO	APPROVED DATE	SUPERSEDES	DWG NO A-98629-90303-1	
	REVISIONS				

98629A ERS -- TERMINOLOGY

IV. Terminology

Some terms used in this ERS or related documents are:

Inbound — This is used to describe information (data or control) which is being sent from the I/O card to the mainframe.

Outbound — This is used to describe information (data or control) which is being sent from the mainframe to the I/O card.

Buffer or Queue — A structure containing data and control information which is sent from one process to another. In this case, the two processes are the mainframe and I/O card.

Producer — The process which accesses a buffer and puts information into it.

Consumer — The process which accesses a buffer and takes information from it.

Circular — A particular type of buffer which has these attributes:

- The producer and consumer can access it at the same time.
- The data in the buffer can wrap around from the end to the front, so the buffer never "runs out" of space unless the producer attempts to put more data in the buffer than the buffer size.
- Once the data has been consumed, it is assumed to be useless and may be written over by the producer.

Control block — A structure stored inside a queue which holds

control information. An inbound control block is called a receive control block in this document.

Block mode	A mode of communication whereby only blocks of data are transferred on the data comm line. When running in this mode, the interface will usually consume and produce large chunks of data at one time.
Packet	A chunk of data sent over the network medium. This contains the information that one interface card sends to another, encapsulated in and transferred by the proper link protocol (level 2).
Node	One unit on the network which can be distinctively addressed.
Level 1	Electrical standard of connections on the network medium.
Level 2	Link control protocol. This is used to pass packets from one node to another on the same network. Level 2 consists of the mechanism used to transfer data packets according to a Link Access Protocol, which attempts to ensure that data is transferred correctly.
I packet	Information packet. Data being transferred from one specific interface to another specific interface is transferred in information packets.
UI packet	Unnumbered Information packet. Data being broadcast from one interface to all others on the network is sent in an unnumbered information packet.
RR response	Receiver Ready response. This special packet is sent to the transmitter of an I packet by the receiver of that I packet as an indicator that the I packet has been correctly received.

98629A ERS -- TERMINOLOGY

SABM packet	Set Asynchronous Balanced Mode packet. This special packet requests that a connection be made between the generator of this packet and the receiver of the SABM.
UA response	Unnumbered Acknowledge response. This is the response to the SABM so that the sender knows that the receiver has received his SABM and is currently connected.
RC request	Roll Call request. This packet is actually generated by the mainframe but the response is expected to be generated by the receiving interface card. It is used for system diagnosis and configuration.
RCR response	Roll Call Response. This special packet is generated as a response to a Roll Call request received by an interface.

98629A ERS -- FUNCTIONAL SPECIFICATION

V. Functional Specification

A. Features

THEODORE will be able to connect the 9826 to the multiplexer over a distance of up to 60 metres. The interface will use synchronous I/O over twisted pair cables at a data rate of 700 Kbits/second (87.5 Kbytes/second). Other features of the interface are:

-- shared memory with the mainframe, using the same scheme as the 98628 for sharing the memory

-- half-duplex communications to other mainframes

-- self-test procedure at power-up and reset

-- can interrupt the mainframe when data is received

-- has a receive data buffer of 2047 bytes and a transmit buffer of 1391 bytes

-- has an SA mode for servicing by signature analysis methods

-- has timeouts of

1) loss of Rx or Tx clock during data transfers

2) link access

3) proper response to transmitted packets

-- responds to received data for interface to interface communication reliability

-- requires responses to transmitted data packets and will retry up to 20 times in an attempt to get the correct response from the intended data receiver

98629A ERS -- FUNCTIONAL SPECIFICATION

B. Modes of Operation

There will be switch settings on the interface card to select:

1) link address

2) select code

3) interrupt level

Furthermore, there will be control registers that the interface implements which can change the operation and mode of the interface. These will be covered in the section on control and status registers.

C. Performance

As mentioned before, the interface will communicate with other mainframes at a rate of 700 Kbits/second, and at a distance of up to 120 metres. Furthermore, the interface will:

-- accept received message packets at any time that it is not transmitting data (if there is space in the Rx buffer)

-- be ready to receive data within 600 us microseconds after a prior reception

-- begin transmitting 200 us microseconds after being given the command to transmit (if no receptions occur first)

Limitations:

-- the interface can only communicate in half-duplex mode; i.e., it can only transmit or receive at one time, not simultaneously

-- when the receive buffer is full, no more message packets will be received until enough room for another minimum sized message packet is available; likewise transmissions will

hang until there is sufficient room in the transmit buffer for a message packet.

D. Applicable standards

The format of the shared memory and the operation of the interface will adhere to the standards for intelligent interface cards for the 9826.

E. Error handling

Errors may occur for a number of reasons (enumerated later). These are reported to the mainframe via an error code which has meaning to the mainframe drivers. The mainframe will be interrupted when the error occurs if the interface is enabled to interrupt. The causes for errors are as follows:

304. Too many characters without terminator. This error will occur when an OUTPUT statement is executed which completely fills the transmit buffer on the interface card, but does not write a terminator block to the interface. Data in the buffer will be lost.

306. Self-test failure. This error indicates that the interface failed its self test and that there is a hardware failure on the interface card.

315. Missing clock. If the Tx clock (supplied by the multiplexer pod) is lost for 20 consecutive data transmissions, this error will occur. Its occurrence signifies:

a) a break in the cable

b) failure of the multiplexer

c) failure in the interface card itself.

316. CTS false too long. When the interface is requested to transmit data, it must be given a clear to send from the multiplexer within 1 second or this error will occur. This error signifies:

a) a break in the cable between the interface and the multiplexer

b) a failure in the interface card itself

c) a failure in the multiplexer

d) a failure in another interface connected to the multiplexer

326. Register address out of range. This error will occur if an illegal operation is attempted on the interface, such as writing to a control register which is not implemented.

327. Register value out of range. If a CONTROL statement is executed containing an illegal value for the control register being written, this error will occur.

353. Data link failure. If the interface retries to transmit a packet 20 times without receiving the proper response from the receiver, it will generate this error.

98629A ERS -- INTERFACE SPECIFICATION

VI. Interface Specification

A. There are a number of hardware registers on the interface card; these registers and their usage are defined in the A-5955-6582-1 ERS, so they will not be repeated here. They include reset/ID, interrupt level, the semaphore, etc. as well as a command register, data register, and buffer pointers. The exact usage of these registers and memory locations are described in the A-5955-6582-1 Hardware ERS.

The operation of the interface can be divided into four categories:

-- Powerup/reset
-- Interface/mainframe communication
-- Outbound data transfers
-- Inbound data transfers

1) Powerup and resets

The powerup reset is done by hardware. The interface card initializes all peripherals and data structures, then indicates the outcome of self-test to the mainframe by interrupting the mainframe with an error code (a code of 0 is reserved for the case of no error).

A reset is done by a write into the RESET register. The interface card sets the semaphore, executes the reset, and resets the semaphore as a signal to the mainframe that the reset is complete.

2) Interface/mainframe communication

The mainframe and the interface communicate to each other via a set of registers and two sets of control block/data block buffers. These registers are used to pass status and control information between the card and the mainframe. The buffers are used to pass data and control information between the card and the mainframe.

The mainframe uses the COMMAND_REG to pass STATUS requests to the interface. The requested status register number is encoded within the command. The interface card responds to a STATUS request by placing the contents of the STATUS register into the DATA_REG and

clearing the COMMAND_REG.

CONTROL statements may cause a command to be written into COMMAND_REG, or may generate a control block to be enqueued, depending upon the type of CONTROL statement. If a command is written into the COMMAND_REG, the command is executed and then completion is signaled to the mainframe by clearing the COMMAND_REG as explained before. However, if a control block is enqueued with data, its actions will not be taken until the interface processes pending control or data blocks.

Errors in STATUS or CONTROL commands will be flagged as soon as the error is detected.

The control statement will cause the operation of the interface card to be modified, based upon what the control statement is. Any registers that are not defined will produce a "register number" error. Illegal values for defined registers will generate a "register range" error. The implemented control registers are listed here:

98629A ERS -- INTERFACE SPECIFICATION

Control registers:

Register number	Data req'd	Meaning
0	yes	Equivalent to WRITEIO Sc,0; Data. Resets the card if the MSB is set.
5	yes	Attached to ",END" specifier in OUTPUT list. Specifies for interface to transmit data in transmit buffer.
12	yes	Sets the number of retries which the interface will attempt when transmitting data. Default is 20.
101	no	Done immediately by drivers at CLEAR statement execution time. Causes the interface to clear receive and transmit data buffers.
111	no	Enqueued by drivers immediately before the end-of-line characters are sent. This action is ignored.
120	yes	Causes interface processor to jump to RAM portion whose LSB = value given. Is used for a debugging aid.
121	yes	Set mask for INT_COND interrupting conditions (reasons why the card interrupts; serviced by the drivers) Bit 0: errors Bit 1: read available (received data block) The poweron/reset state is bit 0 set.

123	yes	Sets up address used by CONTROL 124 (magic poke) and STATUS 124 (magic peek)
124	yes	Magic poke. Writes the value given into the address specified by a previous CONTROL 123.

98629A ERS -- INTERFACE SPECIFICATION

STATUS allows the user to read certain status registers on the interface card. These registers are given here:

Reg. #	Meaning
0	Returns the hardware ID field (52 or 180) of this interface. If the remote control bit (marked "R") is set, the ID will be 180; if not set, it will be 52. This register returns the same value for a 98628 or 98629 interface card.
1	Returns 1 if interrupts from the interface are enabled, 0 if not.
2	Returns <>0 if this interface is busy. This register should contain a 0 value (not busy).
3	Returns firmware ID number for this interface (3). This register is used to determine that this interface is a 98629 instead of a 98628 (which never returns 3)
5	Returns two bits specifying information about data available: 00 -- Receive buffer empty 01 -- Receive data available, no control blocks buffered 10 -- Receive control blocks available, no data buffered 11 -- Both control blocks and data available Note: only case "00" and "11" should ever occur
6	Returns the node address of this interface card.
7	Returns total number of CRC errors received since card reset/poweron.

98629A ERS -- INTERFACE SPECIFICATION

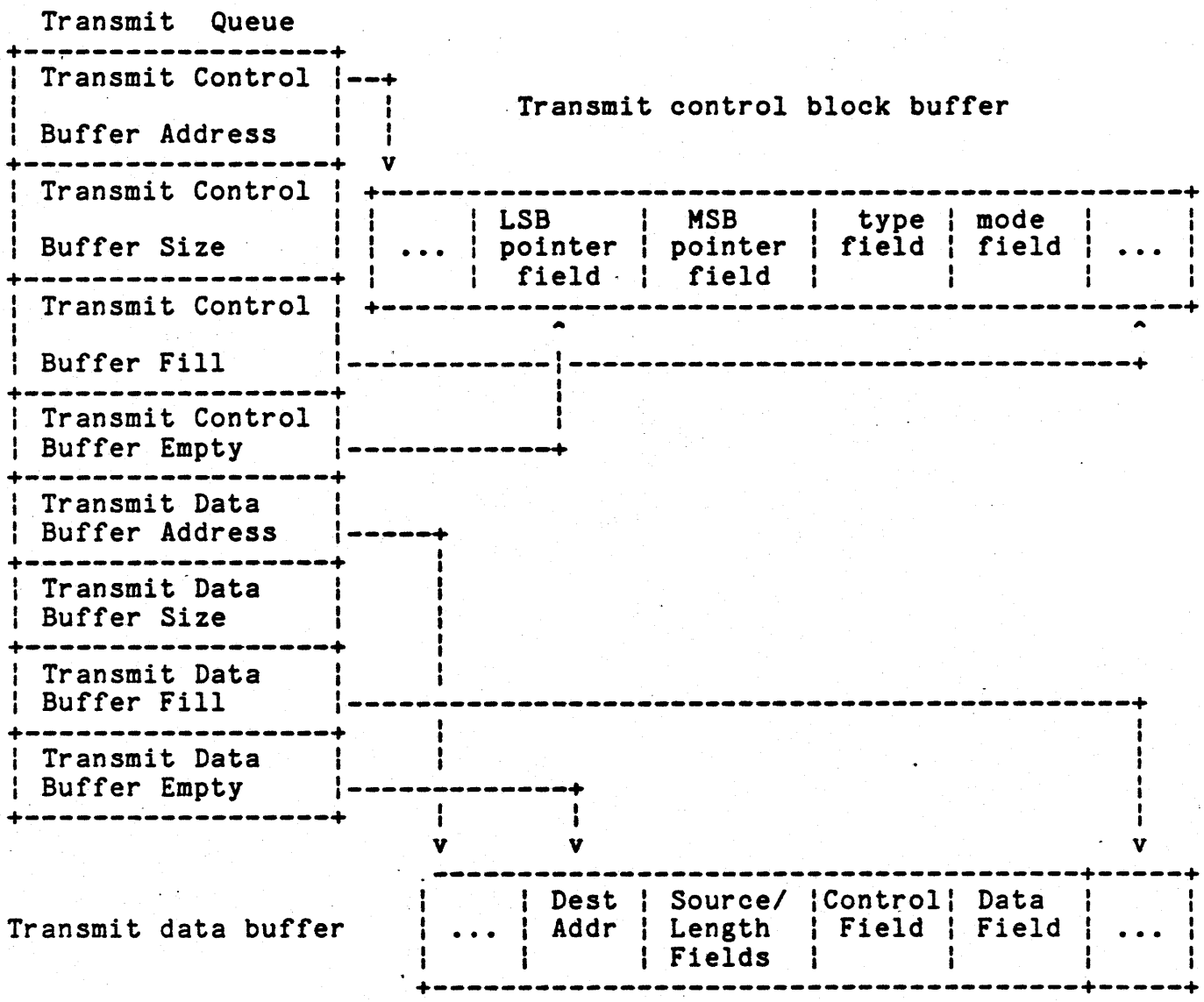
8	Returns total number of times the receive buffer has overflowed since reset/poweron.
9	Returns the type field of a control block, if the last ENTER terminated on a control block. This register should yield a 5 if an ENTER has terminated normally since poweron/reset.
10	Returns the mode field of a control block, if the last ENTER terminated on a control block. This register should yield a 1 if an ENTER has terminated normally since poweron/reset.
11	Returns the amount of space available in the transmit data buffer.
12	Returns the number of transmission retries performed by the interface since poweron/reset.
121	Returns mask for INT_COND interrupt conditions: Bit 0: don't care Bit 1: read available
124	Magic peek. Returns a byte of data from the address specified by a previous CONTROL 123.

98629A ERS -- INTERFACE SPECIFICATION

3) Outbound data transfers

Outbound data is written into the transmit buffers by the mainframe in the form of data blocks. The mainframe will generate a data block which begins with the destination node address. Space for the source address, length field, and control field must be reserved by the mainframe (whatever is written in these locations will be overwritten by the interface). Finally, the mainframe writes the information field. After generating all these fields, the mainframe moves the Tx data buffer fill pointer to point to the byte in the Tx data buffer which follows the last byte in the information field. It then generates a control block which has a pointer field containing the same value as the Tx data buffer fill pointer which it has just written. Finally, it generates a term value of 5 (indicating a transmit command) and a mode of 1 in the control block. Data transmission by the interface is triggered by the updating of the Tx control buffer fill pointer by the mainframe. When this process is complete, the transmit data and control buffers look like this:

98629A ERS -- INTERFACE SPECIFICATION



			MODEL	STK NO	
			98629 ERS		
			BY Alan Nelson	DATE 7-21-82	
	SEE PAGE 1 FOR REV.		APPD B. Thayton	SHEET NO 20 OF 29	
LTR	PC NO	APPROVED DATE	SUPERSEDES	DWG NO A-98629-90303-1	
	REVISIONS				

7820-006-1

If the transmit data buffer becomes full, but the mainframe has not generated the control block, the interface card will generate an error #304.

When the mainframe generates the final control block, and updates the transmit data buffer fill pointer (Tx FP), the data block is assumed to be complete. The interface fills in the source address (which it has read from its node address switches) and the packet length. The interface then enables the SIO transmitter, which causes it to send flags. The interface then begins to monitor CS from the pod for permission to begin sending data. When CS goes true, the interface transfers data from where the transmit data empty pointer (Tx EP) points to the last byte in the data block (pointed to by Tx FP -1). If the interface does not receive CS true within 1 second, it will generate a "CTS false too long " (#316) error.

After transmitting the data, the interface begins waiting for an appropriate response from the intended receiver of the packet just sent. If this response does not appear after 100 milliseconds, the interface will assume that the transmitted packet was lost and re-transmit it. It then waits again for a response. This process is repeated until the appropriate response is received or the interface has retried 20 times, at which time it gives up and reports a "Data link failure" (#353) error.

4) Inbound data transfers

Data is received from the link whenever another interface on the link sends a message packet with a destination address which matches the node address of this interface or the broadcast address, which is an address of all 1's (11111111). When this occurs, the interface is interrupted and data is transferred into the receive data buffer. If the CRC is correct on the received packet, the interface updates the Rx data buffer fill pointer, which is set to point at the next available byte in the Rx data buffer.

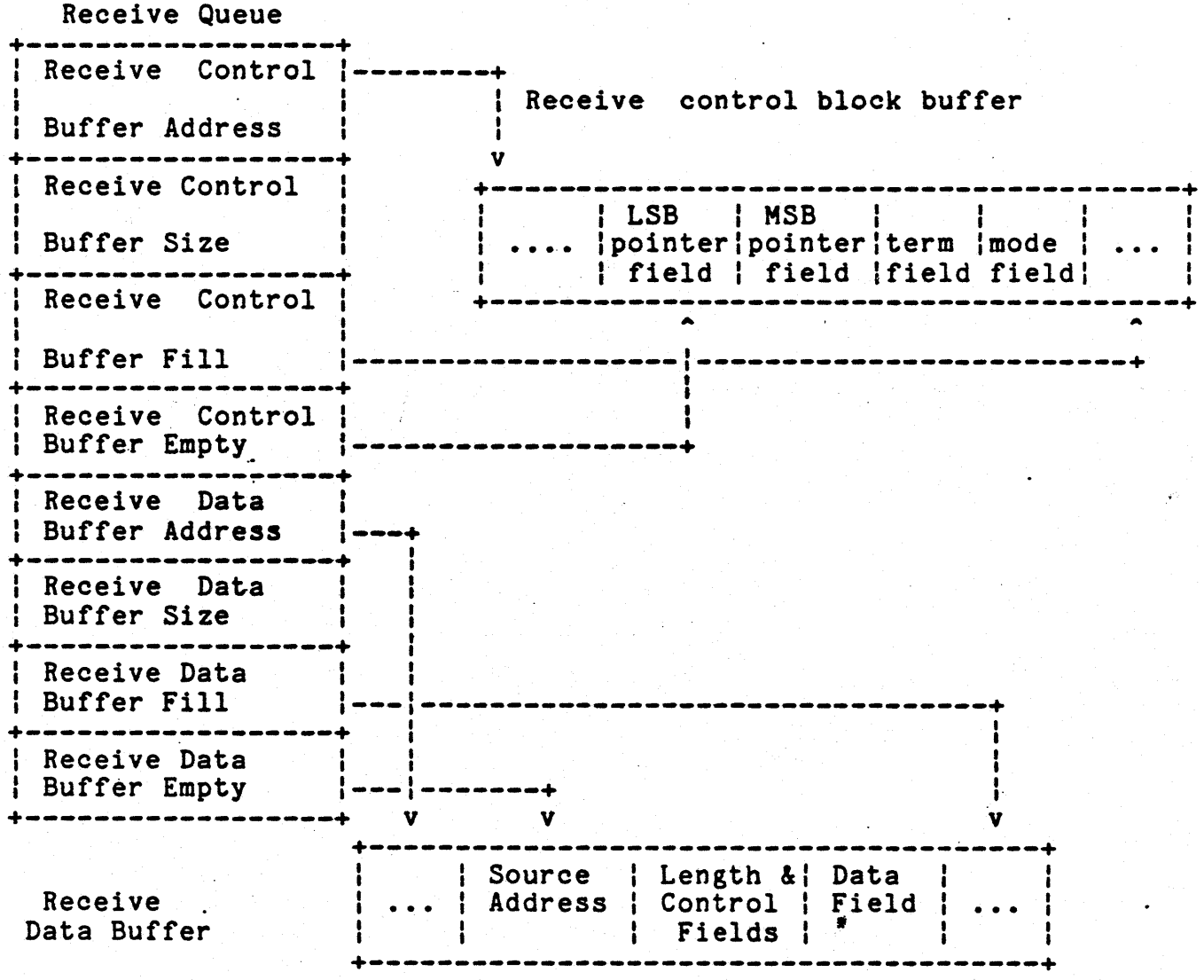
98629A ERS -- INTERFACE SPECIFICATION

If the received data is to be given to the mainframe, the control field is zeroed, and an appropriate response is queued up to be sent to the transmitter of the packet. The interface then generates a control block which points to the same byte pointed to by the Rx data buffer fill pointer. This control block also contains a term value of 5 (indicating Rx data) and a mode of 1. If the interface is enabled, it interrupts the mainframe, indicating an interrupt with Rx data available. At this point, the mainframe may read the data in the data buffer.

If the CRC on a message packet is not correct, the interface does not generate control blocks or move the Rx FP. If the interface runs out of room in the receive data buffer, or the SIO's receive FIFO, an error counter is incremented and the data is ignored (no control blocks are generated, etc.).

98629A ERS -- INTERFACE SPECIFICATION

After a correctly received message packet, the buffers will look like this:



				MODEL	STK NO
				98629 ERS	
				BY Alan Nelson	DATE 7-21-82
	SEE PAGE	1 FOR REV.		APPD B. Keyton	SHEET NO 23 OF 29
LTR	PC NO	APPROVED	DATE	SUPERSEDES	DWG NO A-98629-90303-1
	REVISIONS				

98629A ERS -- LINK INTERFACE

VII. Link Interface

The Resource Management Link consists of up to four desktops which are connected to the Shared Resource Manager through a Resource Management Multiplexer (the 98028A pod). When an interface is directed to send a message packet by its mainframe to the Shared Resource Manager, it executes the following steps:

1) It fills in the source address and packet length in the packet

2) The interface enables its transmitter to send flags

3) When the pod sends the interface a transmit clock, it will detect flags being transmitted from this interface and will indicate CS true to the interface.

4) When the interface detects CS as being true, it disables its receiver and proceeds to transmit the message packet.

5) At the end of the transmission, the interface enables its receiver and disables the transmitter.

6) The pod negates CS and stops sending a transmit clock when it detects no more data being sent.

7) The interface returns to its background job.

Note that the interface can receive message packets at any time except during the time that it is actually transmitting data. When a packet is transmitted by another node on the link which has a destination address matching that of this node, the USRT will interrupt the processor on the interface and indicate that data is being received. The interface performs these functions:

1) Immediately following the interrupt, the interface saves status.

2) A timer is enabled which will make sure that the packet is received within the proper time allotted (if the timer expires, it

means that the receive clock stopped in the middle of a reception).

3) The data is moved from the USRT into the receive data buffer.

4) The trailing CRC is checked for correctness; if correct, the interface generates the appropriate control blocks and updates the Rx FP.

5) Finally, the interface restores CPU status and returns to its background routine.

98629A ERS - Communications Protocol, Level 2

VIII. Communications Protocol -- Level II

A. Packet format

A message packet supported by level 2 will have the following appearance:

Order of sending:

First ----------> Last

Leading Flags	Dest. Addr.	Source Addr.	Length field	Level field	Control field	Data field	CRC MSB	CRC LSB	Trailing Flags

Leading and trailing flags:

A flag is an eight-bit pattern which is unique in that it has six consecutive ones. This indicates that it is not normal data because normal data has a zero inserted after every five consecutive ones. The bit pattern for a flag is 01111110. These flags delimit the beginning and the end of a message packet. There are at least 10 leading flags and 70 trailing flags before and after the packet.

Destination address:

This byte is the node address of the interface which is to receive this message packet. Its value can be in the range of 0-63, and 255. If the destination address is a 255 (11111111), it is regarded as a broadcast address and will be received by all interfaces.

Source address:

This byte is the node address of the interface which is sending the message packet. It is used by the receiver node for replies to the packet. It has the same value range as the destination address.

98629A ERS -- Communications Protocol, Level 2

Packet length:

These 2 bytes are filled in by the interface. This field (LSB first) represents the total length of the packet from destination address to CRC bytes, inclusive.

Control field:

This byte indicates the packet type. See the next section on Link Access Protocol for the types and their meaning.

Level field:

This byte is an indicator of which level in the communications hierarchy should handle the packet. If the level is 2, the packet is a "Roll Call" request which should be handled by the interface card.

Data field:

This field consists of from 16 to 700 bytes. It is the information which the upper levels are transferring between themselves.

CRC field:

This 2 byte field contains a cyclic redundancy checksum of all bits from the first bit in the destination address to the last bit of the data field. This field is used by the receiver of the packet to verify that there have been no errors in transmission.

B. Link Access Protocol

The Link Access Protocol is a system of communication which attempts to ensure data transfers. When an interface transmits data, it expects to receive a proper response from the receiver interface (except broadcast data). If it doesn't receive this response within 100 milliseconds, it re-transmits the data. It repeats this process 20 times, until it gives up and generates an error 353, "Data Link Failure".

Of course, the receiver must generate an appropriate response to a received packet (unless it was broadcast). It does this by enqueueing the proper response so that the background process will transmit it. Sequence numbers are used in the Information packets (which transfer most data); the receiving interface increments the sequence number sent and returns it in the response (RR packet). If the transmitter does not receive a response (nor one with the expected sequence number, which is the incremented value), it re-transmits.

Interfaces get "in sequence" (initialize their sequence numbers) by a connection sequence. If an interface is requested to transmit to a node that it is not currently connected with, it will transmit a connection request (SABM) to that node and expect a response (UA). When it receives this UA, it sets its sequence numbers to zero and proceeds to transmit the requested data packet. Likewise, if an interface receives a packet from a node to which it is not connected, it will ignore the packet but will reply with an SABM. The transmitter is then responsible to give the receiver the UA that it expects, followed by a re-transmission of the original packet.

Finally, a special request exists named "Roll Call" (RC). This is a special request which contains the Level field set to 2. When the receiving interface recognizes that it has received such a request, it is responsible to respond with a "Roll Call Response" (RCR). The formats of each of these packets is given in the GANGLINK ERS, # A-5957-4306-1, available from manufacturing specs.

The exact Link Access Protocol definition is graphically illustrated on drawing # C-5957-4326-1, available from manufacturing specs.

Control field format:

The control field contains a value which indicates the type of packet being transmitted. These types and formats are:

```
          +---------+
I:        |XXX1XXX0|  Information packet; type is always even
          +---------+

            Nr  Ns    Nr= receiving seq. number, Ns= sending seq. number
```

				MODEL	STK NO
				98629 ERS	
				BY Alan Nelson	DATE 7-21-82
	SEE PAGE 1 FOR REV.			APPD B. Keaton	SHEET NO 28 OF 29
LTR	PC NO	APPROVED	DATE	SUPERSEDES	DWG NO A-98629-90303-1
	REVISIONS				

98629A ERS -- Communications Protocol, Level 2

RC:	00011011	Roll Call request; = 1BH
RCR:	01011011	Roll Call Response; = 5BH
RR:	XXX10001	Receiver Ready (Information packet response); =X1H
	Nr	Nr= Ns of received I packet + 1 (if I packet is in sequence)
SABM:	00111111	Set Asynchronous Balanced Mode (connection request); = 3FH
UA:	01110011	Unnumbered Acknowledge (response to an SABM); type = 73H
UI:	00010011	Unnumbered Information packet (broadcast); =13H