The latest version of the DESTEK NIB firmware for MULTIBUS, PC, and S100 is version M400, N400, and S400 respectively. In general this firmware operates as described in the DESTEK user manuals. The following describes changes in the initialization parameters.

Initialization parameters:

byte	default	function
0, 1	none	Board network address, range 0001H - FEFFH
2, 3, 4	undefined	Monitor message address (high, low, sap). Address to which monitor messages are sent
5-8	00000000H	Initial time of day
9	10.	Guaranteed message retry count
10	2 = 0.5 sec.	Guaranteed message acknowledge timer in 250 msec ticks. Amount of time to wait for acknowledge of gauranteed message.
11	64 = 256 bytes	Message size of packet divided by 4
12	0	RESERVED, NOT USED
13	10 = 2.5 sec.	Activity timer in 250 msec ticks. If no activity on guaranteed message channel. for this time, channel will be killed.
14	2 = .5 sec.	Guaranteed message kill time. Channel kill time.
15	0	Monitor message timer in 250 msec ticks.
16	0	Asynch timer message delay in 250 msec ticks.
17	0	Asynch timer sap
18	2	Buffer threshold value

Variable Packet Size

The NIB board sends packets of information across the network. The maximum size of the data packet is specified in byte 11 of the initialization parameter list. the value specified tells the NIB board the maximum amount of data it can send (not including message header). It is specified as the

data size/4

All boards on the network should have the same packet size specified. If a board receives a message with more bytes than it can receive, the message is considered to be illegal and will be discarded.

Message Format and Message Types

The format of network messages is the same as with the older versions of the firmware. The differences are that when a gauranteed message failure occurs all unacked messages in the board are returned to the host with the message type changed to a value of 6.

message types are:

00	- Datagram
01	- Gauranteed message
02	- Statistics
03	- Time of day message
04	- Asynchronous timer message (output to host only)
05	- Error message (output to host only)

Error types:	00	message type error
	01	message size error
	02	network unavailable
	03	no virtual channel available
	04	unused
	05	cannot gaurantee broadcast message (destination address FFFFH)
	06	network transmit error

06	- Gauranteed message error

Statistics and Monitor Message

Statistics are returned only to the host board. When the board receives a message type 2, statistics message request, it will return to the local host a message type 2, with the statistics as the data portion of the message.

USING the NIB-100 NETWORK BOARD

ABSTRACT

This manual describes the DESTEK NIB-100/07 network board. This manual supercedes Revision 00 of USING the NIB-100 NETWORK BOARD. Change bars in the right-hand margin of a Section Heading indicate that part or all of that section has been revised.

PREFACE

Manual Objectives

This manual provides an introduction to the DESTEK NIB-100/07 board and the Desnet network. It describes how to unpack, configure and install the board and provides detailed programming information that you will need in order to communicate with other boards and hosts in the network.

Intended Audience

This manual is intended for OEMs and end-users of DESTEK's NIB-100/07 board; that is, those responsible for installing, configuring and programming the board for effective communication with the network.

Document Structure

The NIB-100/07 Network Board manual consists of the following chapters and appendixes:

Chapter 1 Introduces the product and provides a brief description of the Desnet connection. It tells you what the packing box should contain and offers some hints on unpacking the box.

Chapter 2 Describes how to configure the NIB-100/07 board.

Chapter 3 Tells you how to install the board and connect the network.

Chapter 4 Is the meat of the manual. It defines the programming interface for the NIB-100/07 board.

Appendix A Provides a programming example for a simple device driver for the NIB-100/07 network board.

Appendix B Documents the message format that is transmitted over the network cable.

Appendix C Shows the format of the Network Statistics Message.

Appendix D Shows the format of the Set Configuration information

Appendix E Defines the IBM PC bus signals used on the NIB-100/07 board.

Appendix F Describes the connector (J3) on the NIB-100/07 board. It includes part numbers and pin descriptions.

Table of Contents

CHAPTER 1

Introduction to the NIB-100/07

1. The Desnet Connection

The DESTEK NIB-100/07 plug-in board fits into the IBM PC bus structure and links your host processor to the Desnet network. Once this link is established, you can share the data and peripherals of other Desnet members, providing they have similar software protocols which allow their system to communicate with yours.

Because Desnet is a general-purpose networking system, it works with virtually any kind of system architecture. Using Desnet, you can connect processors of dissimilar architecture, thereby implementing a fully distributed, data processing network.

Desnet uses a bus, or broadcast topology in which all stations share a single high-bandwidth serial channel. Local stations contend for access to the bus. Stations can fail, be added or be removed from the network without disturbing network operations. Each network module has its own self-checking diagnostics.

1.1. The NIB Board

As a user, you can configure a Desnet connection using a mixture of stand-alone and board-level modules. This manual deals exclusively with the NIB series, but you should be aware of the other modules as well:

- o The NIS series modules configure networks of equipment with dissimilar design or architecture. These self-contained modules include all control logic for connection to Desnet. They connect to systems or peripherals through an integral RS-232-C serial port or an integral parallel I/O port, with a Centronics-compatible mode or a full parallel handshaking mode.

- o The NIB series includes the board set for, among others, the S-100 bus and the Multibus. Each NIB board set includes a network controller that provides all control logic and error detection/correction independent of the host system.

Both NIS and NIB modules use baseband coaxial cable as standard interconnect media. You have the option of using other interconnect media such as broadband, fiber optics or a microwave link. Desnet transmission speed is currently 2 megabits per second.

1.2. Unpacking the Box

Each NIB module consists of a Network Interface Unit (NIU) printed circuit board which provides the baseband interface to the network. Other Media Access Unit (MAU) boards are available which provide the broadband coaxial, fiber optic, or microwave interfaces.

Before unpacking the box, inspect it! Are there signs of damage? If you notice something amiss, do not proceed with unpacking. Notify your purchasing agent and contact the shipper's representative.

When unpacking, make sure that all connections remain intact. Remove the board(s) from the impact-deadening material. Be careful not to bump or bend pin connections and printed interface connections. Save the shipping carton -- you may want to transport the module at a later date.

Check the contents of the box against the packing slip. Have you removed everything from the box? If anything is missing, contact your DESTEK representative immediately. Next, inspect the boards for physical damage. If a component appears to be damaged, you should request an inspection by a local agent of the carrier who delivered the package. You must then file a claim with the carrier. To support the claim, you must have a written report of the agent's findings, along with his signature. If a claim is necessary, expect full cooperation from DESTEK.

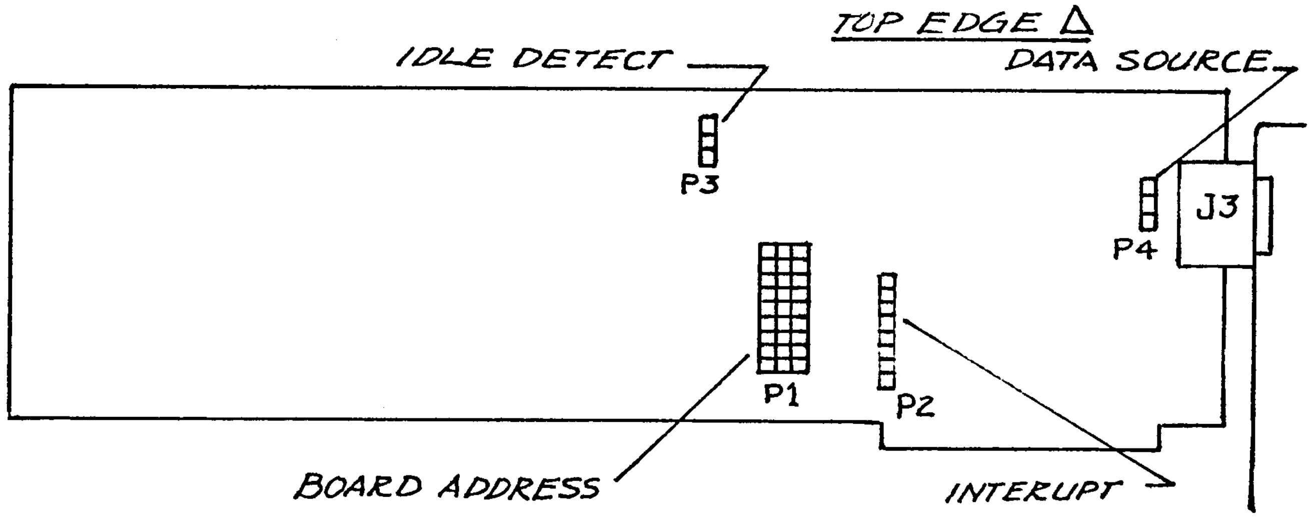
CHAPTER 2

Configuration

2. Configuring the NIB-100/07 Board

Figure 2-1 is a facsimile of the NIB-100/07 board. Because we will refer to it throughout this manual, you should take a few moments to familiarize yourself with the board. It provides a "closeup" of the jumper options and conectors that you will be setting in this section.

FIGURE 2-1 "The NIB-100/07 Board"



Before using the NIB-100/07 board, you must set four jumper options. These four options are:

- The host processor board address (Pl) specifies which I/O address the board will use.
- The interrupt option (P2) specifies the way the NIB board will interrupt the host processor.
- The External carrier/Idle detect Option (P5) specifies whether or not there is an RF modem connected to the board.
- The data source (P4) jumper specifies where the data will come from.

2.1. Setting the Host Processor Board Address

The board address option (P1) defines which I/O address the board will use. To enhance software compatibility, it is a good idea for all Desnet modules in the network to have the same I/O address. Note however, that when two or more boards reside in the same computer, they must have different addresses at P1.

The board address option consists of seven (7) jumpers. These jumpers define the I/O address bits A9 through A1, and are used by the NIB board as the base address. For example, Figure 2-2 shows P1 set to a board address of 200 (hex). The dash (-) indicates the position of the jumper covers.

Figure 2-2 "Setting a Board Address"

```
   P 1

   0     1
   ______
  |o   o--o| A9
  |        |
  |o--o   o| A8
  |        |
  |o--o   o| A7
  |        |
  |o--o   o| A6
  |        |
  |o--o   o| A5
  |        |
  |o--o   o| A4
  |        |
  |o--o   o| A3
  |        |
  |o--o   o| A2
  |        |
  |o--o   o| A1
   ------
```

The final address bit, A0, is used to select one of two registers. Chapter 4 discusses these registers in detail, but briefly, the meaning of A0 is:

0 status register (read-write)

1 data register (read-write)

2.2. Setting the Interrupt Option

The interrupt option (P2) is used to allow the NIB board to indicate the status changes of the board. Note that you must wire this option; you cannot use jumpers. There are two (2) conditions which are monitored, and can cause interrupts:

- o a data byte is available to the host (SA), and

- o host can write a data byte to the NIB (SD)

SA and SD must be hooked to unique lines; never the same line. If SA and SD are connected to the same line, the processor could lose an interrupt from either SA or SD.

The NIB board can generate one of six different interrupts for the IBM PC. These are:

- o interrupt levels IRQ2 to IRQ7

You can specify which interrupt you want to use, or specify none. If no interrupt is to be used then the board must be used in a polled I/O method (see Appendix A; the Programming Example).

Figure 2-3 shows P2 set so that SD is not connected and SA is wired to Vector Interrupt level 2.

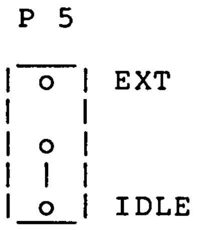
Figure 2-3 "Setting an Interrupt Option"

```
      P 2
     ___
    | o | SD
    |   |
    | o | SA
    | | |
    | o | IRQ2
    |   |
    | o | IRQ7
    |   |
    | o | IRQ6
    |   |
    | o | IRQ5
    |   |
    | o | IRQ4
    |   |
    |_o | IRQ3
```

2.3. Setting the External Carrier/Idle Detect Option

Before a NIB-100/07 board can "talk" on the cable, it must check to see if others are talking. The external carrier/idle detect option (P5) consists of one jumper. Use the idle detect bit (pins 2 and 3) if no carrier signal is available. Use the external carrier bit (pins 1 and 2) when a carrier detect signal is available (as from an RF modem). Figure 2-4 shows P5 set for the idle detect option.

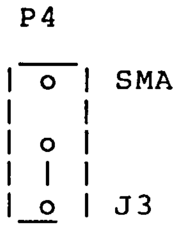
Figure 2-4 "Setting the Idle Detect Option"



2.4. Setting the Data Source Jumper

The data source jumper (P4) consists of three jumpers. Because the data will always come through the 15-pin connector (J3), you should always use the bottom jumper. See Appendix F for a complete description and illustrations of the connector. It includes part numbers and pin descriptions. Figure 2-5 shows P4 set for the 15-pin connector.

Figure 2-5 "Setting the Data Source Option"



CHAPTER 3

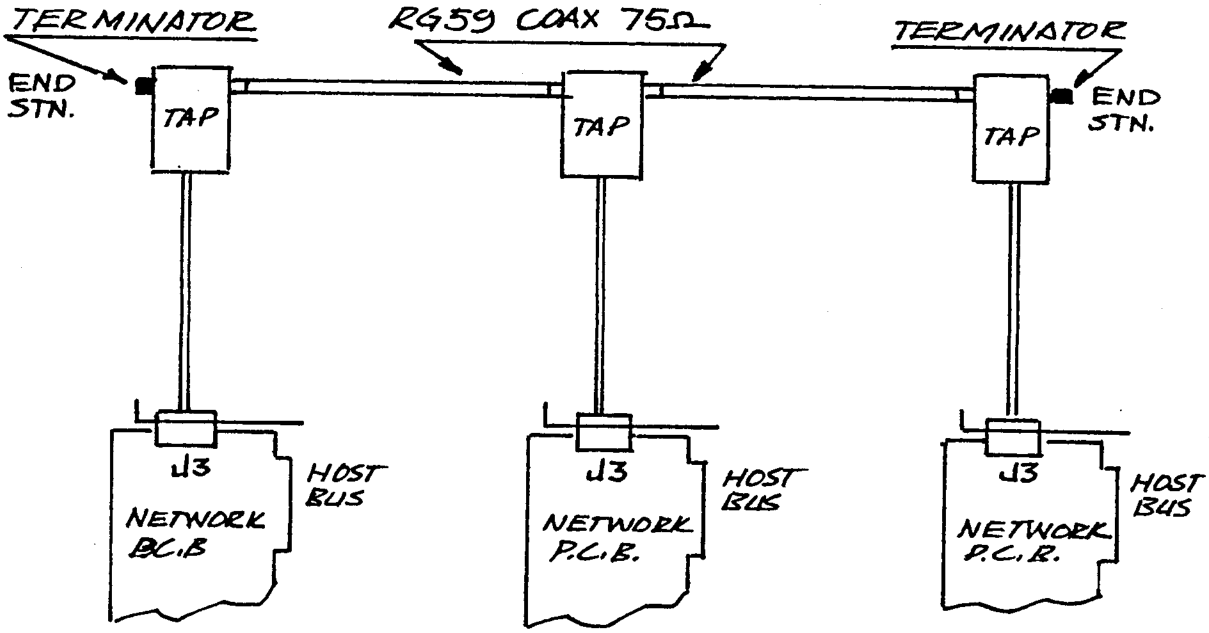
Making the Network Connection

3. Connecting the Network

In this section, you will install the NIB-100/07 board and connect it to the network. After you have set all of the options described in Chapter 2, you can install the board and connect the network. To install the board, simply slide the bottom edge into any slot in the IBM PC bus.

Figure 3-1 illustrates a network tap network connection. This is the way the NIB-100/07 board interfaces to the Desnet network and the host processor.

Figure 3-1 "Network Tap Topology"

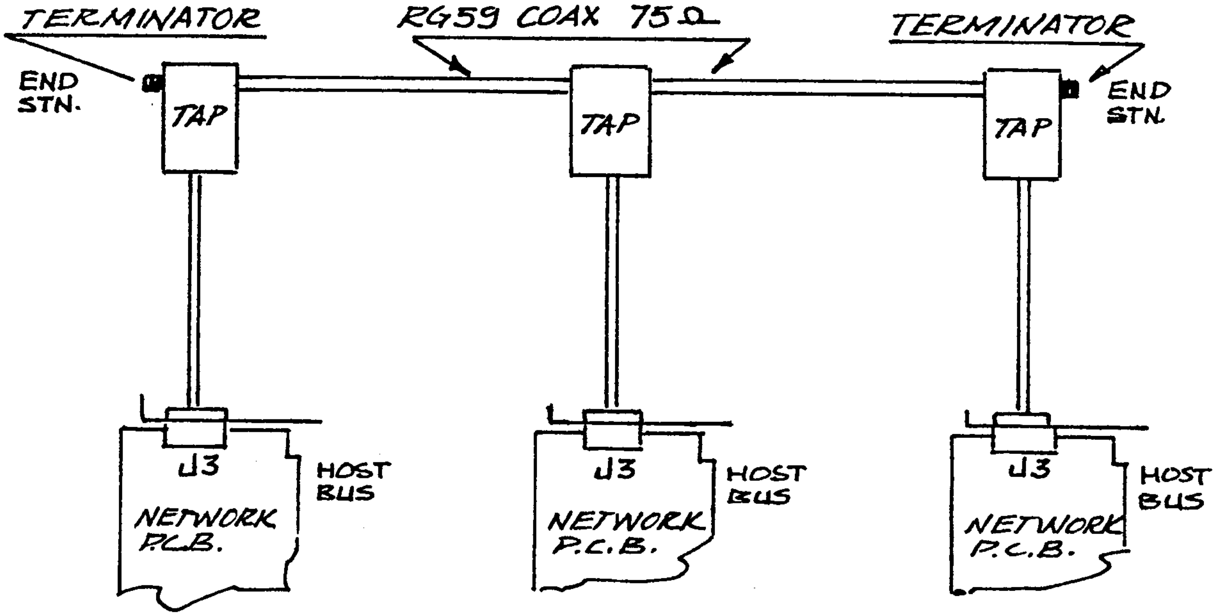


3.1. Connecting via the Network Tap

To connect via the network tap, install the cable before connecting any of the nodes to the network. Next, connect the taps to the coaxial cable and place terminators in the first and last taps.

To connect a node to a tap, set the Data Source Option jumper (P4) so that the data comes in through the 15-pin connector (J3). Finally, attach a control cable between the tap and J3 on the board. Figure 3-2 illustrates the proper way to connect with network taps.

Figure 3-2 "Connecting with Network Taps"



CHAPTER 4

NIB-100/07 Programming Interface

4. Programming the NIB-100/07 Board

In this section you will learn how to start and reset the NIB-100/07 board. You will also learn how to use the registers, the buffer memory and the controller to send and receive messages and acknowledgements over the Desnet network.

To send messages over the network, you should be aware of the following components:

- Status and Data registers for transferring data from the host to the board and from the board to the host. These registers control all communication with the board.

- Buffer memory which holds outgoing messages to the network and incoming messages from the network.

- The Network Interface Unit (NIU) which controls the transfer of data from the buffer to the host and the host to the buffer, and the sending of acknowledgements when messages are received.

4.1. Starting and Resetting the NIB-100/07 Board

Before using the NIB-100/07 board you must read the diagnostic code returned in the data register. This happens in two instances:

- After the power is initially applied
- After a software reset (writing to the status register)

The value returned to the data register indicates whether or not the board is operational. If the board is operational, the value in the data register will be a zero. If the board is not operational, the non-zero value will indicate which part failed. If there is no value at all in the data register, the board is non-operational. Naturally, if the board fails you should not use it because there is no guarantee that messages will be sent or delivered.

The values returned to the data register are defined as follows. All values are in hexadecimal:

00 - board is operational

1x - ROM failure

2x - RAM failure

3x - CTC failure

4x - DMA failure

5x - network interface failure

6x - miscellaneous failure

The "x" isolates, where possible, the exact cause of the failure.

After resetting the NIB-100/07 board and receiving verification that it is operational, you must write certain control information to configure the board. Appendix D provides complete details about this configuration information, but briefly, the information consists of the following:

- The board's network address
- Where to send optional statistics messages
- The current time of day
- The transmit retry count
- The statistics message delay value
- The asynchronous timer delay value
- The asynchronous timer Service Access Point
- The buffer threshold value

After sending these parameters to the board, you will receive a result code in the data register. If the result code is 0, the board has been properly initialized. If the result code is not 0, the value indicates which parameter was in error, starting with 1, the board's network address.

4.2. The Status Register

The status register is a bi-directional (read/write), 1-byte register that contains the input/output status of the data register. You should examine the status register before trying to write to or read from the NIB board. This will guarantee proper communication.

If you write to the status register, it will automatically result in a software reset of the board and you will have to check the diagnostic code as described in Section 4.1. In addition, all messages currently in the internal buffers, as well as any transmission or reception in progress will be lost. The actual data you write to the status register has no significance -- it is ignored.

The status register is also plays an integral part in the NIB's interrupt logic. When you read the status register, the interrupt lines to the host are automatically cleared. Therefore, you should always read the status register in your interrupt handling routine; otherwise, the hardware will continually force interrupts into your system if you have set the interrupt jumpers.

4.2.1. Contents of Status Register

The status register contains the following bits:

```
RWIx xxxx
```

You can ignore the "x" positions of the register. They have no meaning here. Instead, let's concentrate on the "R," "W" and " I" bits now.

The R bit indicates whether or not there is data for the host processor to read. If set to 0, there is no data to read. If set to 1, there is data and the host processor may read it.

The W bit indicates whether or not the host processor can write to the NIB-100 board. If set to 1, data can be written. If set to 0, the host processor cannot write to the board.

The I bit indicates that there is an interrupt pending to the host processor. If set to 0, no interrupt service is required. If set to 1, an interrupt is pending.

4.3. The Data Register

The data register is a bi-directional 1-byte register that transfers information either to or from the NIB-100 board. If the host processor executes an output instruction, the data register receives output data from the host processor and transfers it to the NIB board for transmission over the network. If the host processor executes an input instruction, the data register receives input data from the NIB board and transfers it to the host processor. See Appendix A (Programming Example) for an example of how to program the board interface.

4.3.1. Output Data (To NIB)

The host sends a message to the board one byte at a time, through the data register. To prevent data loss, the host cannot send a byte to the data register until the status register indicates that the data register is empty.

When bit 6 (the "W" bit) of the status register equals 0, the data register is full with the previous byte of the transmission. Each time bit 6 changes to 1, the NIB sends an optional interrupt to the host, indicating that the data register can receive another byte.

4.3.2. Input Data (from NIB)

When a byte of data from the board fills the data register, bit 7 (the "R" bit) of the status register equals 1. The NIB board then interrupts the host to receive the data. After the host reads the byte of data, bit 7 is set back to 0 and the data register can receive the next byte from the board. This process continues until the host has read the entire message.

4.4. Sending and Receiving Messages

When programming the NIB-100 board, you should remember that it is an asynchronous device; that is, it operates independently from the host processor. Because the NIB does not synchronize its work to the host, it can be collecting messages from the network, passing results to the host, and transmitting network bound messages.

The NIB-100/07 board can perform four commands:

- Send a datagram message
- Send a guaranteed message
- Get board statistics
- Get the time of day

You may issue these commands in any order. You need not wait for one command to finish before issuing another. However, to take advantage of this overlap, you will need to write a more sophisticated driver using a software mechanism called the Service Access Point (see Section 4.4.2) and possibly using the interrupt option. Also, remember that it is possible for a later request to finish before an earlier one; i.e., completion requests do not follow in any temporal order. Finally, be aware that it might take a significant amount of time -- perhaps seconds -- for a request to complete.

Along with the commands, the NIB-100/07 board can generate two message types: the asynchronous timer interrupt and the error response message. The board sends the asynchronous timer interrupt, unsolicited, if the proper value is set in the configuration parameters (see Appendix D). The board sends the error response message when it cannot satisfy a request.

4.4.1. Specifying a Station Destination Address

To send a message from one station to another station in the network, you must specify a station destination address. This address consists of two parts: the network station address and the service access point. Appendix D describes how to set the network station address. Briefly, if you set the address to FFFF (hex), all stations in the network will receive the message. If you specify an address of 0, the message will be returned to the host. The 0 address is simply a synonym for the current board address. Any other address specifies a particular network station to recieve this message. If the address you specify matches that of your station then you will receive the message, just as if you had supplied an an address of 0.

4.4.2. The Service Access Point (SAP)

The service access point is a software mechanism that provides more detail about where a message comes from (the Source Service Access Point or SSAP) and where the message should be delivered (the Destination Service Access Point or DSAP). The values of the SAPs are optional, but must be supplied as part of the message; you can use them to help divide station address space. In fact, the SAP field is ignored by the NIB and is for host use only. However, you will need to use SAPs if you wish to take advantage of overlapping commands, or if you wish to send a message to a particular channel on a remote node.

4.5. Message Formats

To send or receive a message, the NIB board must be presented a sequence of bytes. All multi-byte fields are transmitted high order byte first; low order byte last. The datagram message and other messages have the same structure, except for the TYPE byte. Section 4.5.6 outlines the differences for the Type byte. Figure 4-1 illustrates the message structure.

Figure 4-1 "Message Structures"

Message Size (7+n)	[2]
Destination Address	[2]
Destination SAP	[1]
Source Address	[2]
Source SAP	[1]
Message Type	[1]
Message	[n]

4.5.1. Size Bytes

The first two bytes of the message must indicate the number of bytes in the remainder of the message. This number can range from 7 to 263.

4.5.2. Destination Address Bytes

The next two bytes specify the destination address; i.e.; where the message should be delivered. You only need to provide the destination address when passing a message to the board.

4.5.3. Destination SAP Byte

Following the destination address is the Destination Service Access Point (DSAP). It provides more information (perhaps a channel number) about where to send the message in a particular station. The DSAP is ignored by the NIB-100 and is for host use only. It can contain any value the user's system requires, except for those values reserved by DESTEK (see Appendix B).

4.5.4. Source Address Bytes

The next two bytes specify the source address for the message. When you are sending a message to the NIB-100 board it will automatically insert correct values for this field. When you recieve a message from the NIB-100 board this field will have a value which indicates the sender of the message. If you where going to send a reply message you would then use this field as the destination address for the message.

4.5.5. Source SAP Byte

The Source Service Access Point provides more detail about where a message comes from. For example, it could tell which port or channel number on the remote address originated the message. An incoming message will include the SSAP, but the receiver can ignore it.

4.5.6. The Type Byte

Once again, the Type byte is where the datagram message and the guaranteed message part ways. The Type byte has the following meaning:

- 00 - Datagram Message
- 01 - Guaranteed Message
- 02 - Statistics Information
- 03 - Get Time of Day Message
- 04 - Asynchronous Timer Message
- 05 - Error Message

4.5.7. The Data Bytes

You need only supply the data portion of the message when you are sending a Datagram or Guaranteed message to the NIB-100. For all other message types this field is ignored if it is sent as part of a message.

4.6. The Datagram Message

The NIB board transmits and receives datagram messages across the network. The datagram message does not require a response from the receiving NIB or the host program. Therefore, a datagram will never tell the sending host whether or not the message was delivered. The sending NIB will transmit a datagram message only once, and will not verify that the message arrived.

When the NIB board receives a datagram from the network and buffer space is available, it queues the message for delivery to the host processor. The NIB delivers the message, one byte at a time, by writing to the data register. As each byte is written to the data register, the status register changes and an optional interrupt is generated. Once it delivers the entire message, the NIB removes it from the queue and passes the next available message to the host.

4.7. The Guaranteed Message

There are times when it might be necessary to make certain that a message is delivered to another network station. At these times it will not be sufficient to use Datagram messages, but to use Guarantee message service. The NIB-100 boards, in a co-operating manner, will try to deliver your message (or messages) to the intended destination. Should this not be possible after repeated attempts you will be informed by an error message.

When the NIB board receives a guaranteed message from the network it checks if buffer space is available. If there is no space, the receiving NIB tells the sending NIB that it could not receive the message. The sending NIB will then attempt to re-transmit until it it receives a good acknowledgement from the receiving NIB.

Should the addressed NIB-100 board not receive the message during an attempt to send it, the sending NIB-10 board will requeue for retransmission after some time has elapsed. It will attempt this retransmission until it has either delivered it, or determined that it could not because the transmission retry count has expired.

Upon successful reception, the NIB board queues the message for delivery to the host processor. The NIB delivers the message, one byte at a time, by writing an input instruction and the data to the data register. As each byte is written to the data register, the status register changes and an optional interrupt is generated. Once it delivers the entire message, the board removes it from the queue, and passes the next available message to the host.

There may be times when you exceed the capacity of the NIB board to transmit guaranteed messages. If you exceed the capacity of the board, your message will not be transmitted and you will receive an error message. At these times you should just retry the transmission of the message. Also, you should be aware that you cannot issue a guaranteed message for global transmissions; i.e., those messages going to more than one receiver. See Appendix D, "Configuration Parameters" for any other constraints.

4.8. The Statistics Message

The NIB board can also report statistics and other information. If the TYPE byte is a 02, the board will return board-level statistics to the host. See Appendix C (The Network Statistics Message) for details about the contents of the message. The NIB-100 can transmit these statistics automatically to any station in the network. You might want to do this in order to provide error information, billing, or network usage information. Figure 4-2 illustrates the format of the statistics message.

Figure 4-2 "Receiving Statistics"

23	[2]
Destination Address	[2]
Destination SAP	[1]
Source Address	[2]
Source SAP	[1]
02	[1]
Statistics	[16]

4.9. The Time of Day Message

The NIB-100 board is capable of reporting the time of day. If the TYPE byte is 03, this is a time of day message. The four-byte data field (sent highest order bit first; lowest order bit last) is the number of 1 millisecond pulses since the last reset of the board or set configuration command. Figure 4-3 illustrates the format of the time of day message.

Figure 4-3 "Time of Day Message"

11	[2]
Destination Address	[2]
Destination SAP	[1]
Source Address	[2]
Source SAP	[1]
03	[1]
Time of Day	[4]

4.10. Asynchronous Timer Message

If the host receives a message from the board with a 04 in the TYPE byte, it is an asynchronous timer message. These messages are useful for host systems that require periodic interrupts, but do not have hardware clocks available. A system might need these periodic interrupts for forced scheduling, additional message support (host-to-host acknowledgements), etc. The asynchronous timer interrupt is also useful for guaranteed message delivery protocols. The board will send an asynchronous timer message, unsolicited, if the proper value is set in the configuration parameters at board startup or reset time (see Appendix D). Figure 4-4 illustrates the format of the asynchronous timer message.

Figure 4-4 "Asynchronous Timer Message"

11	[2]
Destination Address	[2]
Destination SAP	[1]
Source Address	[2]
Source SAP	[1]
04	
Time of Day	[4]

4.11. Error Messages

Finally, if the host receives a message from the NIB board with a 05 in the TYPE byte, it is an error message. When the NIB board cannot satisfy a request, it returns the error message. The two-byte data field consists of the old TYPE value and the reason for the error. The defined reasons are:

- 0 - invalid TYPE field
- 1 - invalid SIZE field
- 2 - no access to network
- 3 - no available guaranteed message ports
- 4 - guaranteed message port killed; most likely can't access network cable (busy)
- 5 - tried to issue broadcast guaranteed message

An invalid TYPE (type 0) error occurs when a message is sent to the NIB board that it cannot recognize. An type 1 error occurs if there are too few or too many bytes in the message. The type 2 error occurs if the NIB board cannot get access to the network after trying for approximately one minute. The type 3 error occurs if you try to send too many guaranteed messages too close together. You should wait a few seconds and then try again. Error type 4 occurs when a sending and receiving NIB get out of sequence by more than one message and the two NIBs cannot synchronize. Finally, error type 5 occurs when you try to send a guaranteed message with an address of FFFF (hex).

The original destination, DSAP, source and SSAP fields are returned in the error message. Figure 4-5 illustrates the format of the error message returned from the NIB-100 board.

Figure 4-5 "Error Message Returned from NIB-100"

09	[2]
Original Dest. Address	[2]
Original Dest. SAP	[1]
Original Src. Address	[2]
Original Src. SAP	[1]
05	[1]
Original Msg. Type	[1]
Error Code	[1]

Appendix A

NIB-100/01 Programming Example

This Appendix provides a programming example for a simple device driver for the NIB-100/07 network board. It assumes that interrupts are not available, and as a result it is necessary to poll the status register. The only necessary comment is to state that the symbol "NIB" is defined somewhere to be the base address of the board. There are three routines provided - NIBIN, NIBOUT and NIBRST which get a data byte, output a data byte to the NIB board, and reset the NIB board, respectively.

```
NIBSTAT equu     NIB                      ;status port address
NIBDATA equ      NIB+1                    ;data port address

OUTFULL equ      040h                     ;output buffer full - can't write
INFULL  equ      080h                     ;input data available - must read

;       NIBIN - get a data byte
;
;                input - no arguments
;                output - AL register has data byte
;
NIBIN:  in       al,(NIBSTAT)             ;get status register
        and      al,INFULL                ;is data available
        jz       NIBIN                    ;no, so wait
        in       al,(NIBDATA)             ;yes, get the data
        ret                               ;back to caller

;       NIBOUT - output a byte
;
;                input - AH register has data byte
;                output - none
;
NIBOUT: in       al,(NIBSTAT)             ;get status register
        and      al,OUTFULL               ;data output register still full?
        jz       NIBOUT                   ;yes, so wait
        out      (NIBDATA),ah             ;output the byte
        ret                               ;back to caller
```

```
;       NIBRST - reset the NIB
;
;
;
NIBRST: out     (NIBSTAT),al            ;force NIB to reset
        mov     bx,100
NIBR:   dec     bx                      ;wait for reset to work
        jnz     NIBR
        call    NIBIN                   ;get diagnostic code
        or      ah,ah <= AL, AL   SEE NIBIN
        jz      NIBR00                  ;it's okay
        halt                            ;it's bad, so halt
NIBR00: mov     bx,0                    ;start of parm area
        mov     cl,CEND-CONS            ;size of area
NIBR01: mov     al,CONS[bx]             ;get a parm byte
        call    NIBOUT                  ;and give it to the NIB
        inc     bx                      ;point to next parm byte
        dec     cl
        jnz     NIBR01                  ;go for more...
        call    NIBIN                   ;get the result code
        or      ah,ah
        rz                              ;zero is okay
        halt                            ;all others in error
CONS:   db      x,y                     ;network address - fill in
        db      0,0,0                   ;statistics msg address
        dw      0,0                     ;current time of day
        db      0                       ;use default retry count
        db      0                       ;use default send delay value
        db      0                       ;use default rcv delay value
        db      0                       ;use default rr delay value
        db      0                       ;use default rnr delay value
        db      0                       ;use default kill delay value
        db      0                       ;no generation of auto stat msg
        db      0,0                     ;no asynch timer msg
        db      0                       ;use default buffer threshold
CEND:
```

Appendix B

Network Message Format

This appendix describes the message format that is transmitted over the network cable. It is included for informational use only and does not effect the user's interface to the NIB-100/07 board.

0111 1110	[1]
Destination Address	[2]
Destination SAP	[1]
Source Address	[2]
Source SAP	[1]
Control	[1]
Data	[1..256]
Fcs	[2]
0111 1110	[1]

The fields have the following meaning:

Destination Addr - 1 to FEFF (hex) user assignable
FF00 to FFFE (hex) Destek reserve
FFFF (hex) global message

Destination SAP -	00 to 7F (hex) user assignable 80 to FF (hex) Destek reserved
Source Address -	see Destination Address
Source SAP -	see Destination SAP
Control -	00 datagram message 01 - guaranteed message
Data -	Data field
Fcs -	frame check sequence, a 16-bit CRC

Appendix C

Network Statistics Message

Every Destek network board has the ability to automatically send board level statistics to a network monitor or to the host. Additionally, a host program can specifically request these statistics. Board level statistics can be used to locate problems in the network.

The Network Statistics Message has the following format:

no. of host input msgs	[2]
no. of host data msgs	[2]
no. of host guar. msgs	[2]
no. of host rejected	[2]
no. of rcv'd msgs	[2]
no. of rcv'd data msgs	[2]
no. of rcv'd guar msgs	[2]
no. of rcv'd rejected	[2]

The fields in this message have the following meaning:

no. of host input msgs - total number of messages given to the NIB board by the host

no. of host data msgs - total number of datagram messages given to the NIB board by the host no. of host guar. msgs - total number of guaranteed messages given to the NIB board by the host

no. of host rejected - total number of illegal messages given to the NIB board by the host

no. of rcv'd msgs - total number of network messages received by the NIB board in response to network transmissions

no. of rcv'd data msgs - total number of datagram messages that the board received over the network

no. of rcv'd guar msgs - total number of guaranteed messages that the board received over the network

no. of rcv'd rejected - number of messages received over the the network and not intended for this board

Appendix D

The Configuration Parameters

The configuration parameters provides a software method that allows you you to assign board configuration values. The 19-byte section contains the following information:

Station Address	[2]
Statistics Address	[3]
Initial Time of Day	[4]
Retry Count	[1]
Send Time	[1]
Rcv Time	[1]
RR Time	[1]
RNR Time	[1]
Kill Time	[1]
Stat Msg Delay	[1]
Async Timer Delay	[1]
Async Msg SAP	[1]
Threshold Count	[1]

The fields here have the following meaning:

Station Address - This address will become the network station address of this board. Each board in the system must have a unique sixteen bit address in the range 1 to FEFF (hex). The addresses FF00 through FFFE are reserved by DESTEK. Finally, the address FFFF (hex) is used for global messages within the network. When setting or changing this address, you should vary the high order byte first; then the low order byte. A unique high order byte reduces the overhead on NIB board processing because the hardware checks the high order byte whereas the software checks the low order byte. This is the only value which must be nonzero.

Statistics Address - If statistics messages are to be generated automatically (see Stat Msg Delay parameter), this is the network address that will receive statistics from the board. If the first two bytes are 0, this message will be sent to the host cpu currently connected to this NIB. This address includes the SAP as well as the station address.

Initial Time of Day - If the NIB will keep the current time of day, this is the start value, in 1 millisecond increments. It can be set to 0 in which case the count is just the time from the last reset of the NIB.

Retry Count - If set to 0, do not retry message transmissions. If set from 1 to 255, retry indicated number of times.

Send Time - A guaranteed message timer. This is the amount of time, in 250 millisecond increments, to keep the communications channel open between host send requests. (Default is 12.)

Rcv Time - A guaranteed message timer. This is the amount of time, in 250 millisecond increments, to keep the communications channel open between network-available messages. (Default is 8.)

RR Timer - A guaranteed message timer. This is the amount of time to wait,in 250 millisecond increments, for an acknowledgement before trying to retransmit. (Default is 2.)

RNR Timer - A guaranteed message timer. After a negative acknowledgement, this is the amount of time to wait ,in 250 millisecond increments, before trying to retransmit. (Default is 10.)

Kill Timer - A guaranteed message timer. This is the amount of time, in increments of 250 millisecond, that must elapse while a commucications channel is being killed. During this time, the NIB cannot accept any messages any messages over this communication channel. (Default is 8.)

Stat Msg Delay - The amount of time to wait, in minutes, between statistics messages. If this is set to 0, statistics messages are kept, but not automatically sent.

Async Timer Delay - The amount of time to wait, in 250 millisecond increments, between async timer messages. If set to 0, no asynchronous timer messages will be generated.

Async Msg SAP - This is the intended receiver of the async timer message. (See Async Timer Delay.)

Threshold Count - This is the minimum number of free buffers in the NIB to hold in reserve. There are a total of 12 buffers. (Default is 2.)

Appendix E

IBM PC Bus Specifications for the NIB-100/07 Board

This section describes the IBM PC bus signals used on the NIB-100/07 board. Only the signal lines used are documented.

Pin Number	Mnemonic	Meaning
A 2	BD7	Data bit 7
A 3	BD6	Data bit 6
A 4	BD5	Data bit 5
A 5	BD4	Data bit 4
A 6	BD3	Data bit 3
A 7	BD2	Data bit 2
A 8	BD1	Data bit 1
A 9	BD0	Data bit 0
A 11	AEN	Address Enable
A 22	BA9	Address bit 9
A 23	BA8	Address bit 8
A 24	BA7	Address bit 7
A 25	BA6	Address bit 6
A 26	BA5	Address bit 5
A 27	BA4	Address bit 4
A 28	BA3	Address bit 3
A 29	BA2	Address bit 2
A 30	BA1	Address bit 1
A 31	BA0	Address bit 0
B 1	GND	Ground
B 2	RESDRV	Reset
B 3	+5	5 volts
B 10	GND	Ground
B 13	-IOR	I/O read request
B 14	-IOW	I/O write request
B 29	+5	5 volts
B 31	GND	Ground

Optional Signals

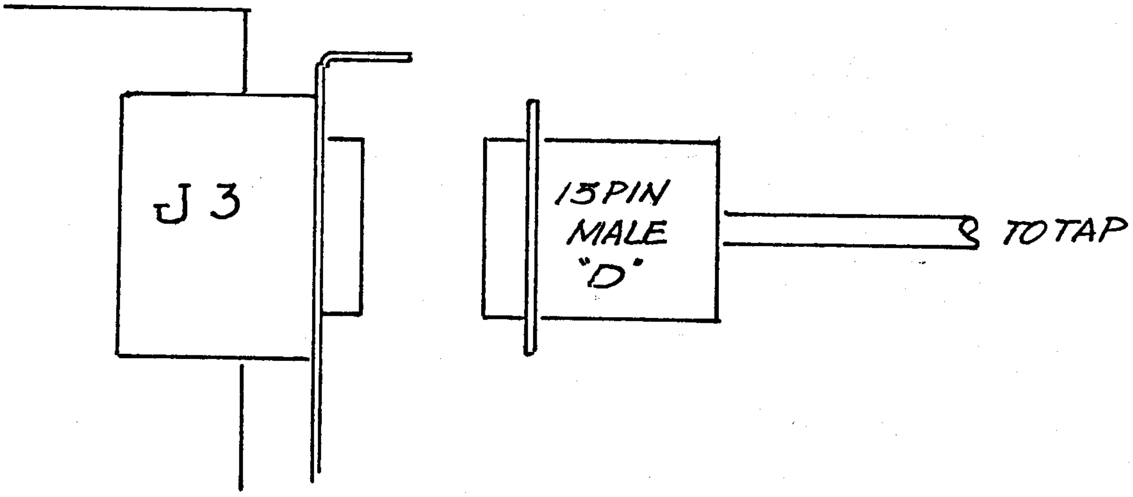
B 4	IRQ2	Interrupt request 2
B 21	IRQ7	Interrupt request 7
B 22	IRQ6	Interrupt request 6
B 23	IRQ5	Interrupt request 5
B 24	IRQ4	Interrupt request 4
B 25	IRQ3	Interrupt request 3

Appendix F

NIB-100/07 Connector

This appendix describes the connector associated with J3. Figure F-1 shows the location of the connector and lists the associated part numbers.

Figure F-1 "NIB-100 Connector"



The pin descriptions for J3, the 15 pin D connector female are:

15 pin D female

1	Isolated Gnd
2	Gnd
3	Gnd
4	Gnd
5	Gnd
6	Gnd
7	Gnd
8	N.C.
9	Isolated +5V
10	+5V
11	XR-Data
12	T-Data
13	RTS-L
14	XCarrier
15	N.C