INTERNATIONAL STANDARD

ISO/IEC 11518-1

First edition
1995-06-01

Information technology — High-Performance Parallel Interface —

Part 1: Mechanical, electrical and signalling protocol specification (HIPPI-PH)

Technologies de l'information — Interface parallèle à haute performance —

Partie 1: Spécification du protocole mécanique, électrique et de signalisation (HIPPI-PH)

INTERNATIONAL ISO/IEC

Reference number
ISO/IEC 11518-1:1995(E)

Contents

Page

Tables

Figures

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 11518-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 25, *Interconnection of information technology equipment*.

ISO/IEC 11518 consists of the following parts, under the general title *Information technology — High-Performance Parallel Interface*:

— *Part 1: Mechanical, electrical and signalling protocol specification (HIPPI-PH)*

— *Part 2: Framing Protocol (HIPPI-FP)*

— *Part 3: Encapsulation of ISO 8802-3 — Logical link control protocol data units (HIPPI-LE)*

— *Part 4: Mapping of HIPPI to IPI device generic command sets (HIPPI-IPI)*

— *Part 5: Memory Interface (HIPPI-MI)*

— *Part 6: Physical Switch Control (HIPPI-SC)*

Annexes A to E of this part of ISO/IEC 11518 are for information only.

Introduction

This part of ISO/IEC 11518 defines the physical layer of an efficient simplex high-performance point-to-point interface operating at speeds of 800 or 1 600 Mbit/s. The -PH abbreviation stands for "physical layer".

Characteristics of this HIPPI physical layer interface include

– point-to-point connections use one or two copper twisted-pair cables for distances of up to 25 m.

– the HIPPI-PH is a simplex interface, capable of transferring data in one direction only. Two HIPPI-PHs may be used to implement a full-duplex interface.

– data transfers are performed and flow controlled in increments of bursts, each burst normally containing 256 words.

– signalling and control sequences are kept simple, and a look-ahead flow control is used, to allow average transfer rates for large file transfers to approach the peak transfer rate, even over distances longer than specified for the HIPPI-PH cables.

– the HIPPI-PH provides support for low-latency, real-time, and variable size packet transfers.

– the HIPPI-PH is designed to facilitate use in a circuit-switched environment. In support of this feature, a limited information field is available for subdevice addressing or other nonspecified control functions during the connection phase of operation. One round-trip cable delay is required to establish or terminate a connection.

– the HIPPI-PH is also designed to transmit multiple packets after a connection has been established. No round-trip cable delays are required between packets.

Figure 1 shows the interrelationship of the different clauses of this part of ISO/IEC 11518. The upper-layer protocols and station management protocols are not covered in this part of ISO/IEC 11518.

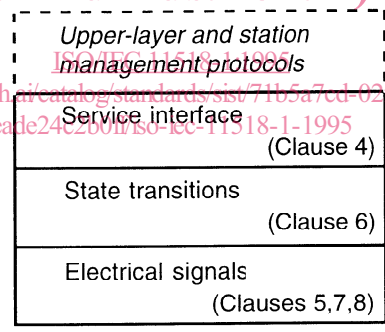

Figure 1 – Control hierarchy

Information technology – High-Performance Parallel Interface –

Part 1: Mechanical, electrical, and signalling protocol specification (HIPPI-PH)

1 Scope

This part of ISO/IEC 11518 provides the mechanical, electrical and signalling protocol specifications for an efficient simplex high-performance point-to-point interface between pieces of data-processing equipment.

The interface described in this part of ISO/IEC 11518 can be operated at peak data rates of 800 or 1 600 Mbit/s, over distances of up to 25 m by means of copper cabling. A distance-independent signalling protocol allows the average data rates to approach the peak data rates, even over distances longer than specified for the HIPPI-PH.

The purpose of this part of ISO/IEC 11518 is to facilitate the development and use of computer systems by providing a common interface at the physical and data framing layers. It provides an efficient interconnection between computers, high-performance display systems, and high-performance, intelligent block-transfer peripherals. The interface is optimized for large block transfers.

2 Definitions, editorial conventions, and abbreviations

2.1 Definitions

For the purposes of this part of ISO/IEC 11518, the following definitions apply.

2.1.1 burst: A group of words, sent during contiguous CLOCK periods. A burst may be sent by the Source for each READY indication received from the Destination. Bursts contain 1 to 256 words. Bursts that contain less than 256 words are called short bursts. A packet contains no more than one short burst. A short burst will be either the first or last burst of a packet.

2.1.2 connection: Condition of the HIPPI-PH when data transfers from Source to Destination are possible.

2.1.3 Destination: The equipment that receives the data.

2.1.4 I-Field: A 32-bit information field sent as part of the sequence of operations establishing a connection from a Source to a Destination.

NOTE – The contents of the I-Field are defined by an upper-layer protocol and are not defined in this part of ISO/IEC 11518.

2.1.5 length/longitudinal redundancy check-word (LLRC): A single word that is sent on the DATA BUS from Source to Destination after each burst.

2.1.6 optional: Features that are not required by this part of ISO/IEC 11518. However, if any optional feature defined by this part of ISO/IEC 11518 is implemented, it shall be implemented according to this part of ISO/IEC 11518.

2.1.7 packet: A data set sent from Source to Destination. A packet is composed of one or more bursts. The HIPPI-PH specification does not limit the maximum packet size, but a maximum size may be imposed by a given HIPPI-PH implementation, or by an upper-layer protocol.

2.1.8 service interface (SI): The means by which the HIPPI-PH provides services to upper-layer protocols.

2.1.9 Source: The equipment that transmits the data.

2.1.10 state: The current condition of the HIPPI-PH, excluding transitions, as indicated by the control signals.

2.1.11 station management (SMT): The supervisory entity that monitors and controls the HIPPI-PH.

2.1.12 upper-layer protocol (ULP): The protocols above the service interface. These could be done in hardware or software, or they could be distributed between the two.

2.1.13 wait: Wait is every CLOCK period when there is no valid information on the DATA BUS. Some wait times are required by the HIPPI-PH signalling protocol; others may be the result of flow control or upper-layer protocol operations.

2.1.14 word: A unit of information, consisting of 32 or 64 bits, matching the DATA BUS width, and transferred from the Source to the Destination during a CLOCK period.

2.2 Editorial conventions

In this part of ISO/IEC 11518, certain terms that are proper names of signals or similar terms are printed in upper case to avoid possible confusion with other uses of the same words (e.g., REQUEST, CONNECT, BURST). Any lower case uses of these words have the normal technical English meaning.

A number of conditions, sequences parameters, events, states, or similar terms are printed with the first letter of each word in upper case and the rest lower case (e.g., Source, Destination). Any lower case uses of these words have the normal technical English meaning.

2.3 Abbreviations

CCI	Connection Control Information
ECL	Emitter Coupled Logic
HIPPI-PH	**H**igh-**P**erformance **P**arallel **I**nterface - Mechanical, electrical, and signalling protocol specification - **P h**ysical (Layer)
LLRC	Lengh/Longitudinal Redundancy Check
PHSM	Physical (layer) Station Management
SI	Service Interface
SMT	Station Management
ULP	Upper-Layer Protocol

3 HIPPI structure

3.1 Configuration characteristics

The HIPPI-PH has been designed in a modular fashion to support different peak bandwidth requirements.

3.1.1 800 Mbit/s

An HIPPI-PH with a DATA BUS width of 32-bit words provides 800 Mbit/s data transfer rates.

3.1.2 1 600 Mbit/s

An HIPPI-PH with a DATA BUS width of 64-bit words provides 1 600 Mbit/s data transfer rates.

3.2 Logical framing hierarchy

Figure 2 shows the basic organization of the information on the HIPPI-PH.

Once a connection is established a packet (or multiple packets) can be sent from the Source to the Destination. Each packet shall contain one or more bursts. Bursts shall contain 1 to 256 words. Bursts that contain less than 256 words are called short bursts. A packet shall contain no more than one short burst. A short burst shall be either the first or last burst of a multi-burst packet.

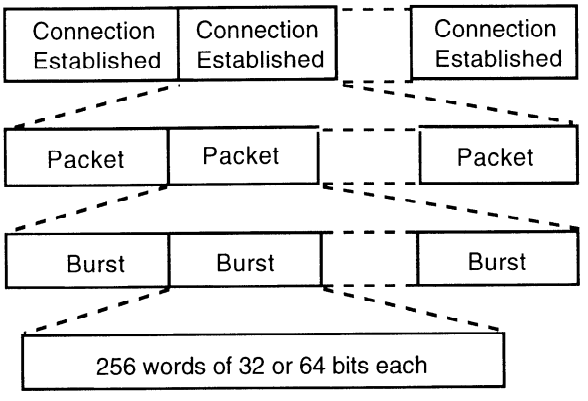

Figure 2 – Logical framing hierarchy

4 Service interface

This clause specifies the services provided by HIPPI-PH. The intent is to allow ULPs to operate correctly with this HIPPI-PH. How many of the services described herein are chosen for a given implementation is up to that implementor; however, a set of HIPPI-PH services must be supplied sufficient to satisfy the ULP(s) being used. The services as defined herein do not imply any particular implementation, or any interface.

Figure 3 shows the relationship of the HIPPI-PH interfaces.

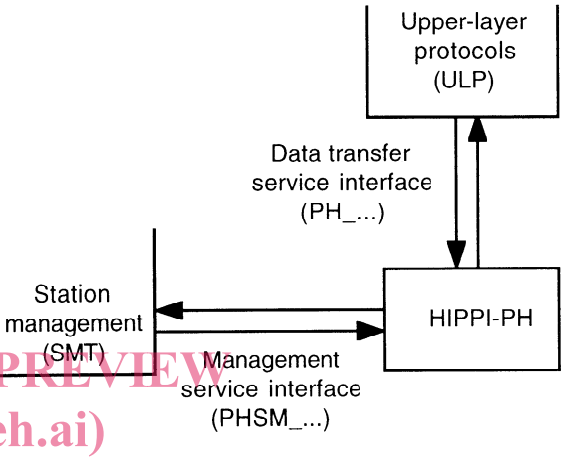

Figure 3 – HIPPI-PH service interface

4.1 Service primitives

The primitives, in the context of the state transitions in clause 5, are declared required or optional. Additionally, parameters are either required, conditional, or optional. All of the primitives and parameters are considered as required except where explicitly stated otherwise.

HIPPI-PH service primitives are of four types.

– *Request primitives* are issued by a service user to initiate a service provided by the HIPPI-PH. In this part of ISO/IEC 11518, a second Request primitive of the same name shall not be issued until the Confirm for the first request is received.

– *Confirm primitives* are issued by the HIPPI-PH to acknowledge a Request.

– *Indicate primitives* are issued by the HIPPI-PH to notify the service user of a local event. This primitive is similar in nature to an unsolicited interrupt. Note that the local event may have been caused by a service Request. In this part of ISO/IEC 11518, a second Indicate primitive of the same name shall not be issued until the Response for the first Indicate is received.

– *Response primitives* are issued by a service user to acknowledge an Indicate.

4.2 Sequences of primitives

The order of execution of service primitives is not arbitrary. Logical and time sequence relationships exist for all described service primitives. Time sequence diagrams are used to illustrate a valid sequence. Other valid sequences may exist. The sequence of events between peer users across the user/provider interface is illustrated. In the time sequence diagrams the HIPPI-PH users are depicted on either side of the vertical bars while the HIPPI-PH acts as the service provider.

4.3 Service primitives summary

Initiate a Connection
- PH_RING.Request (CCI)
- PH_RING.Confirm
- PH_RING.Indicate (CCI)
- PH_RING.Response

Complete the Connection
- PH_ANSWER.Request (Accept/Reject)
- PH_ANSWER.Confirm
- PH_ANSWER.Indicate (Accept/Reject)
- PH_ANSWER.Response

Flow Control
- PH_FLOW.Request
- PH_FLOW.Confirm
- PH_FLOW.Indicate (Enabled)
- PH_FLOW.Response

Packet Control
- PH_PACKET.Request (Begin/End)
- PH_PACKET.Confirm (Accept/Reject)
- PH_PACKET.Indicate (Begin/End,Status)
- PH_PACKET.Response

Burst Transfer
- PH_TRANSFER.Request (Length,Burst)
- PH_TRANSFER.Confirm (Accept/Reject)
- PH_TRANSFER.Indicate (Status,Length,Burst)
- PH_TRANSFER.Response

Terminate the Connection
- PH_HANGUP.Request
- PH_HANGUP.Confirm
- PH_HANGUP.Indicate
- PH_HANGUP.Response

Control Interface
- PHSM_CONTROL.Request (Parameter_List)
- PHSM_CONTROL.Confirm (Status,Status_List)

Interface Status
- PHSM_STATUS.Request
- PHSM_STATUS.Confirm (Status)
- PHSM_STATUS.Indicate
- PHSM_STATUS.Response

4.4 Operational sequences

Primitives issued by the ULP shall be serviced by the HIPPI-PH in the sequences defined in the state transitions of clause 6. An implementation may present the HIPPI-PH with multiple requests for services, but the HIPPI-PH shall service the requests one at a time.

The following sequence of service primitives is an example of normal operation of the HIPPI-PH.

4.4.1 Enable the interface

The interface may be enabled using whatever implementation dependent enabling method, if any, is specified.

4.4.2 Initiate a connection

PH_RING primitives shall be used to initiate a connection from the Source to the Destination. The Connection Control Information (CCI) may be used for non-specified control functions.

4.4.3 Complete the connection

PH_ANSWER primitives shall be used to establish or reject the connection.

4.4.4 Enable Destination reception

When it is ready, the Destination ULP may use the PH_FLOW.Request primitive to indicate that it is willing to accept bursts from the HIPPI-PH.

4.4.5 Start a packet

PH_PACKET (Begin) primitives shall be used to indicate the start of a packet.

4.4.6 Send burst

PH_TRANSFER primitives shall be used to transfer one burst of a packet.

4.4.7 Send more bursts

More bursts may be sent for this packet by returning to 4.4.6.

4.4.8 Terminate the packet

PH_PACKET (End) primitives shall be used to indicate the end of a packet.

4.4.9 Send more packets

More packets may be sent by returning to 4.4.5.

4.4.10 Terminate the connection

The Source or Destination may terminate the connection by using the PH_HANGUP primitives. Note that dedicated point-to-point HIPPI-PHs may never need to terminate the connection once it is established, or only terminate the connection as an error recovery process. See the cautionary note in 5.3.6.

4.4.11 Initiate another connection

Return to 4.4.2 to initiate another connection.

4.5 Initiate connection service primitives

Figure 4 is a diagram of these primitives. They shall be used to request that a connection be established between the Source and Destination.

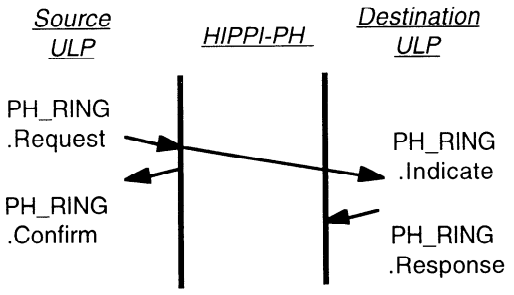

Figure 4 – Initiate the connection service primitives

4.5.1 PH_RING.Request

Issued by the Source ULP to request establishment of a connection from the Source to the Destination. In the case where a previous connection attempt did not complete with a PH_ANSWER.Indicate, and was aborted with a Source ULP issued PH_HANGUP.Request, the Source must wait a time T1 before initiating another connection. T1 shall be a round-trip propagation delay plus appropriate action time. No assumptions are made as to where or how the timer is implemented. It may be done in the ULP or the HIPPI-PH. The timer may be implemented with hardware, software, or a mixture of the two. The default value of T1 shall be approximately 2 ms.

Semantics – PH_RING.Request (CCI)

The CCI parameter is a 32-bit field that may be used for non-specified control operations for the connection establishment.

Issued – The Source ULP issues this primitive to the HIPPI-PH when a connection to a Destination ULP is required.

Effect – The HIPPI-PH shall initiate a connection.

4.5.2 PH_RING.Confirm

This primitive acknowledges the PH_RING.Request from the Source ULP.

Semantics – PH_RING.Confirm

Issued – The HIPPI-PH shall issue this primitive to the Source ULP to acknowledge a PH_RING.Request.

Effect – Unspecified

4.5.3 PH_RING.Indicate

This primitive indicates to the Destination ULP an attempt by a Source ULP to establish a connection.

Semantics – PH_RING.Indicate (CCI)

The CCI parameter is a 32-bit field that may be used for non-specified control operations for connection establishment. This CCI may be different from the CCI supplied by the PH_RING.Request primitive due to the action of intermediate devices, such as switches, between the Source and Destination.

Issued – The HIPPI-PH shall issue this primitive to the Destination ULP when it has received a connection request.

Effect – The Destination ULP should accept or reject the connection request.

4.5.4 PH_RING.Response

This primitive acknowledges the PH_RING.Indicate from the HIPPI-PH.

Semantics – PH_RING.Response

Issued – The Destination ULP issues this primitive to acknowledge receipt of the PH_RING.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_RING.Indicate.

4.6 Complete the connection service primitives

Figure 5 is a diagram of these primitives. To respond to a connection request, the PH_ANSWER primitives shall be used.

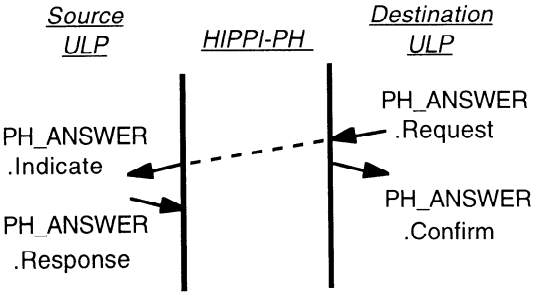

Figure 5 – Complete the connection service primitives

4.6.1 PH_ANSWER.Request

This primitive is issued by the Destination ULP in response to the PH_RING primitives to instruct the HIPPI-PH to reject or establish a connection.

Semantics – PH_ANSWER.Request (Accept/Reject)

The Accept/Reject parameter instructs the HIPPI-PH to complete the connection or to reject it.

Issued – The Destination ULP should issue this primitive to the HIPPI-PH in response to a PH_RING.Indicate.

Effect – The HIPPI-PH shall complete the connection if the Accept parameter is used. If the Reject parameter is used, the HIPPI-PH may either do nothing (in which case the Source ULP shall not receive a PH_ANSWER .Indicate), or perform a short connection sequence (in which case the Source ULP shall receive a PH_ANSWER .Indicate with the Reject parameter).

4.6.2 PH_ANSWER.Confirm

This primitive acknowledges the PH_ANSWER.Request from the Destination ULP.

Semantics – PH_ANSWER.Confirm

Issued – The HIPPI-PH shall issue this primitive to the Destination ULP to acknowledge a PH_ANSWER .Request.

Effect – Unspecified

4.6.3 PH_ANSWER.Indicate

This primitive indicates to the Source ULP that the connection has been accepted or rejected.

Semantics – PH_ANSWER.Indicate (Accept/Reject)

If the Accept parameter is used, the connection has been accepted and is available for data transfer. If the Reject parameter is used, the connection has been rejected and the interface is available for another PH_RING.Request.

Issued – The HIPPI-PH shall issue this primitive to the Source ULP when it has determined the status of the connection request.

Effect – The connection sequence is complete.

4.6.4 PH_ANSWER.Response

This primitive acknowledges the PH_ANSWER.Indicate from the HIPPI-PH.

Semantics – PH_ANSWER.Response

Issued – The Source ULP issues this primitive to acknowledge receipt of the PH_ANSWER.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_ANSWER.Indicate.

4.7 Flow control service primitives (optional)

Figure 6 is a diagram of these primitives. These primitives are optional, and may be used to pass flow control information between the HIPPI-PH and the Source and Destination ULPs.

The Flow Control service primitives are used by the Destination ULP to inform the HIPPI-PH that it is willing to accept a burst in the form of the PH_TRANSFER.Indicate.

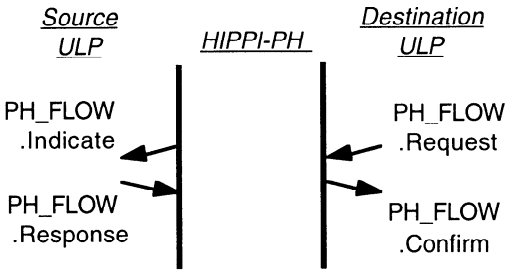

Figure 6 – Flow control service primitives

4.7.1 PH_FLOW.Request

This primitive is issued by the Destination ULP to tell the HIPPI-PH that it is ready to accept a burst from the HIPPI-PH.

Semantics – PH_FLOW.Request

Issued – The Destination ULP issues this primitive to the HIPPI-PH when it is ready to receive bursts.

Effect – The HIPPI-PH shall be enabled to send another burst to the Destination ULP. Each PH_FLOW.Request enables an additional PH_TRANSFER.Indicate.

4.7.2 PH_FLOW.Confirm

This primitive acknowledges the PH_FLOW.Request from the Destination ULP.

Semantics – PH_FLOW.Confirm

Issued – The HIPPI-PH shall issue this primitive to the Destination ULP to acknowledge a PH_FLOW.Request.

Effect – Unspecified

4.7.3 PH_FLOW.Indicate

This primitive notifies the Source ULP that it can send a burst. It shall indicate to the Source ULP the current number of available bursts that the HIPPI-PH is willing to accept.

Semantics – PH_FLOW.Indicate (Enabled)

The Enabled parameter shall be the current number of bursts that the HIPPI-PH is willing to accept. The range is implementation dependent.

Issued – The HIPPI-PH shall attempt to issue this primitive to the Source ULP every time it receives an indication that another burst can be transmitted.

Effect – Unspecified

4.7.4 PH_FLOW.Response

This primitive acknowledges the PH_FLOW.Indicate from the HIPPI-PH.

Semantics – PH_FLOW.Response

Issued – The Source ULP issues this primitive to acknowledge receipt of the PH_FLOW.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_FLOW.Indicate.

4.8 Packet service primitives

Figure 7 is a diagram of these primitives. They shall be used to delimit one or more bursts into entities called packets.

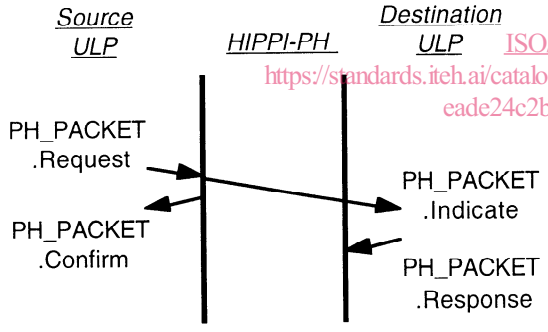

Figure 7 – Packet service primitives

4.8.1 PH_PACKET.Request

This primitive is issued by the Source ULP to delimit one or more bursts into a packet entity.

Semantics – PH_PACKET.Request (Begin/End)

The parameter marks either the beginning or the end of the packet.

Issued – The Source ULP issues this primitive to the HIPPI-PH to delimit one or more bursts into a packet.

Effect – Upon receipt of a Begin parameter, the HIPPI-PH shall mark the start of a packet. Upon receipt of an End parameter, the HIPPI-PH shall mark the end of a packet.

4.8.2 PH_PACKET.Confirm

This primitive acknowledges the PH_PACKET.Request from the Source ULP.

Semantics – PH_PACKET.Confirm (Accept/Reject)

The parameter denotes whether or not the primitive was accepted by the HIPPI-PH. The PH_PACKET.Request primitive may have been issued by the ULP along with other primitives, e.g., RING.Request. If the connection sequence fails, then the packet delimiter would not be inserted and this PH_PACKET.Confirm primitive would indicate Reject.

Issued – The HIPPI-PH shall issue this primitive to the Source ULP to acknowledge a PH_PACKET.Request.

Effect – Unspecified

4.8.3 PH_PACKET.Indicate

This primitive indicates to the Destination ULP that a packet delimiter has been received from the Source ULP.

Semantics – PH_PACKET.Indicate (Begin/End, Status)

The Begin/End parameter indicates either the beginning or the end of the packet.

The Status parameter may include, but is not limited to, an error when a burst with an illegal size and correct LLRC was received as part of this packet.

Issued – The HIPPI-PH shall issue this primitive to the Destination ULP when it receives a packet boundary indication.

Effect – Unspecified

4.8.4 PH_PACKET.Response

This primitive acknowledges the PH_PACKET.Indicate from the HIPPI-PH.

Semantics – PH_PACKET.Response

Issued – The Destination ULP issues this primitive to acknowledge receipt of the PH_PACKET.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_PACKET.Indicate.

4.9 Data transfer service primitives

Figure 8 is a diagram of these primitives. They shall be used to transfer a burst from the Source ULP to the Destination ULP.

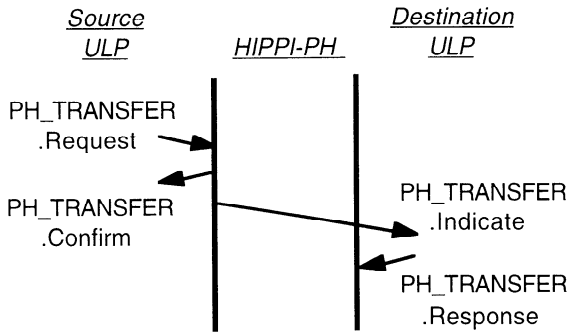

Figure 8 – Data transfer service primitives

4.9.1 PH_TRANSFER.Request

This primitive is issued by the Source ULP to request the transfer of a burst.

Semantics – PH_TRANSFER.Request (Length,Burst)

The length parameter may be specified in any units, but an integral number of words shall be transferred.

Issued – The Source ULP issues this primitive to the HIPPI-PH to request the transfer of a burst to the Destination ULP.

Effect – The HIPPI-PH shall accept the burst for transmission.

4.9.2 PH_TRANSFER.Confirm

This primitive acknowledges the PH_TRANSFER.Request from the Source ULP.

Semantics – PH_TRANSFER.Confirm (Accept/Reject)

The parameter denotes whether or not the primitive was accepted by the HIPPI-PH.

Issued – The HIPPI-PH shall issue this primitive to the Source ULP to acknowledge a PH_TRANSFER.Request.

Effect – Unspecified

4.9.3 PH_TRANSFER.Indicate

This primitive indicates to the Destination ULP that a burst has been received from the Source ULP.

Semantics – PH_TRANSFER.Indicate (Status,Length, Burst)

The Status parameter reports any parity or LLRC errors detected during the data transfer if error checking is supported. Illegal length bursts, with correct LLRC, shall be indicated in the PH_PACKET.Indicate primitive if error checking is supported.

Issued – The HIPPI-PH shall issue this primitive to the Destination ULP when a burst has been received.

Effect – Unspecified

4.9.4 PH_TRANSFER.Response

This primitive acknowledges the PH_TRANSFER.Indicate from the HIPPI-PH.

Semantics – PH_TRANSFER.Response

Issued – The Destination ULP issues this primitive to acknowledge receipt of the PH_TRANSFER.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_TRANSFER.Indicate.

4.10 Hangup service primitives

Figure 9 is a diagram of these primitives. They shall be used to terminate a connection between the Source and Destination. Note that a Hangup can be initiated from either the Source or Destination and will usually affect both the local and opposite ends of the interface.

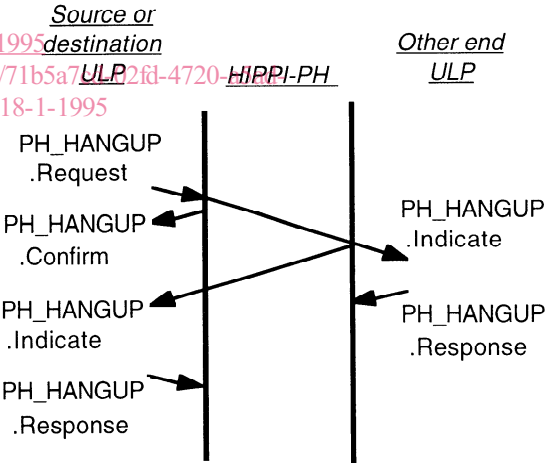

Figure 9 – Hangup service primitives

4.10.1 PH_HANGUP.Request

This primitive is issued by either the Source ULP or Destination ULP to request that the connection be terminated.

Semantics – PH_HANGUP.Request

Issued – The Source ULP or Destination ULP issues this primitive to the HIPPI-PH to terminate a connection.

Effect – The HIPPI-PH shall terminate the connection.

4.10.2 PH_HANGUP.Confirm

This primitive acknowledges the PH_HANGUP.Request to the issuing ULP.

Semantics – PH_HANGUP.Confirm

Issued – The HIPPI-PH shall issue this primitive to the ULP to acknowledge a PH_HANGUP.Request.

Effect – Unspecified

4.10.3 PH_HANGUP.Indicate

This primitive indicates to the ULP that the connection has been terminated. If a connection exists, the ULP that issued the PH_HANGUP.Request shall receive a PH_HANGUP.Confirm when the hangup sequence starts and a PH_HANGUP.Indicate when the sequence completes. The other end ULP shall receive only a PH_HANGUP.Indicate.

Semantics – PH_HANGUP.Indicate

Issued – The HIPPI-PH shall issue this primitive to the ULP when the connection has been terminated.

Effect – The ULP may no longer use the connection for data transfers, and all pending .Requests are cancelled.

4.10.4 PH_HANGUP.Response

This primitive acknowledges the PH_HANGUP.Indicate from the HIPPI-PH.

Semantics – PH_HANGUP.Response

Issued – The ULP issues this primitive to acknowledge receipt of the PH_HANGUP.Indicate.

Effect – The HIPPI-PH is enabled to issue another PH_HANGUP.Indicate.

4.11 Control service primitives

Figure 10 shows the SMT Control service primitives. They shall be used to set parameters and control the interface. Note that a Control primitive can be initiated from either the Source or Destination.

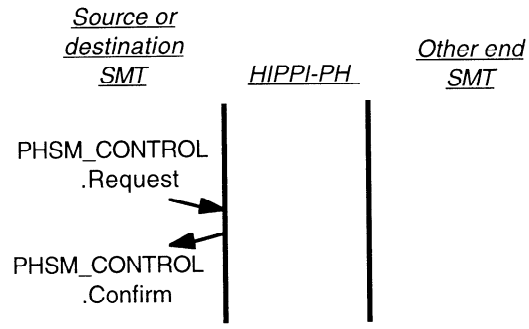

Figure 10 – Control service primitives

4.11.1 PHSM_CONTROL.Request

This primitive is issued by either the Source SMT or Destination SMT to set parameters within the interface, start diagnostics, or otherwise control the interface. One function is specified and others are left to specific implementations.

Semantics – PHSM_CONTROL.Request (Parameter_List)

The Parameter_List specifies what is to be set, started, etc. The Parameter list includes but is not limited to:

Reset

Issued – The Source or Destination SMT issues this primitive to perform some control function over the interface as a whole.

Effect – The HIPPI-PH shall perform the function specified. The Reset parameter shall take the HIPPI-PH to the Disabled state and cancel all pending .Requests.

4.11.2 PHSM_CONTROL.Confirm

This primitive acknowledges the PHSM_CONTROL.Request to the issuing SMT.

Semantics – PHSM_CONTROL.Confirm (Status, Status_ List)

Status reports the success or failure of the PHSM_CONTROL.Request operation. There should be one status value in Status_List for each event initiated with the PHSM_CONTROL.Request.

Issued – The HIPPI-PH shall issue this primitive to the SMT when all of the operations specified in the PHSM_CONTROL.Request have been accepted or completed.

Effect – Unspecified

4.12 Status service primitives

Figure 11 shows the SMT Status service primitives. They shall be used to obtain local status information from the HIPPI-PH. Note that a Status primitive can be initiated from either the Source or Destination and shall only affect the local end of the interface.

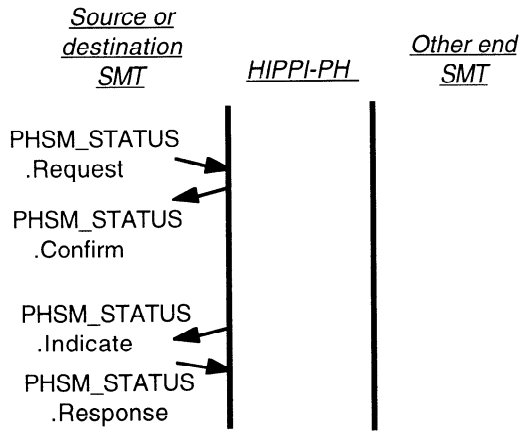

Figure 11 – Status service primitives

4.12.1 PHSM_STATUS.Request

This primitive is issued by either the Source SMT or Destination SMT to request a status report from the HIPPI-PH.

Semantics – PHSM_STATUS.Request

Issued – The SMT issues this primitive to the HIPPI-PH when it wishes to obtain the status of the interface.

Effect – The HIPPI-PH shall respond with a PHSM_STATUS.Confirm.

4.12.2 PHSM_STATUS.Confirm

This primitive replies to the previous PHSM_STATUS.Request with status information.

Semantics – PHSM_STATUS.Confirm (Status)

Status shall contain, but is not limited to:
INTERCONNECT state
Errors

Issued – The HIPPI-PH shall issue this primitive to the SMT to acknowledge a PHSM_STATUS.Request.

Effect – Unspecified

4.12.3 PHSM_STATUS.Indicate

This primitive informs the SMT entity that a major event has occurred that affects the operation of the HIPPI-PH.

Semantics – PHSM_STATUS.Indicate

Issued – The HIPPI-PH shall issue this primitive to the SMT whenever a major event is detected. Major events include but are not limited to a change in the INTERCONNECT signal.

NOTE – If a PHSM_CONTROL.Request was accepted successfully but not completed, then an PHSM_STATUS.Indicate could be used to indicate completion.

Effect – Upon receipt of this primitive, the local SMT entity should issue a PHSM_STATUS.Request to read the status of the HIPPI-PH and determine which event occurred.

4.12.4 PHSM_STATUS.Response

This primitive acknowledges the PHSM_STATUS.Indicate from the local SMT entity.

Semantics – PHSM_STATUS.Response

Issued – The SMT issues this primitive to acknowledge receipt of the PHSM_STATUS.Indicate.

Effect – The HIPPI-PH is enabled to issue another PHSM_STATUS.Indicate.