

Recognized as an
American National Standard (ANSI)

IEEE Std 1284.1™-1997
(Stabilized 2009)

IEEE Standard for Information Technology—Transport Independent Printer/System Interface (TIP/SI)

Sponsor

**Microprocessor and Microcomputer Standards Committee
of the
IEEE Computer Society**

Stabilized 19 March 2009

Approved 29 July 1997

IEEE Standards Board

Reaffirmed 19 June 2003

Approved 24 April 1998

American National Standards Institute

Abstract: A protocol and methodology for software developers, computer vendors, and printer manufacturers to facilitate the orderly exchange of information between printers and host computers are defined in this standard. A minimum set of functions that permit meaningful data exchange is provided. Thus a foundation is established upon which compatible applications, computers, and printers can be developed, without compromising an individual organization's desire for design innovation.

Keywords: computers, printer command sets, printer communications, printers

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1997 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1997. Printed in the United States of America.

IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by the Institute of Electrical and Electronics Engineers, Incorporated.

ISBN 1-55937-944-8

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied “**AS IS.**”

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation, or every ten years for stabilization. When a document is more than five years old and has not been reaffirmed, or more than ten years old and has not been stabilized, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal interpretation of the IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Recommendations to change the status of a stabilized standard should include a rationale as to why a revision or withdrawal is required. Comments and recommendations on standards, and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

[This introduction is not part of IEEE Std 1284.1-1997, IEEE Standard for Information Technology—Transport Independent Printer/System Interface (TIP/SI.)]

Local area networks are increasingly becoming the most popular means of interconnecting devices within a corporation. With costs per connection coming down, this trend shows no sign of abating. As networks grow larger, more computers and printers will be interconnected. Any weaknesses in network printing will only be magnified as more devices are made to communicate.

The absence of feedback from existing printers causes many problems in today's network environment. For example, a user could be submitting a job to a remote printer. If that printer is low on toner, most printers today do not have the capability to inform users about this condition. Additionally, if the job is large, the user risks having to wait until the job is finished before finding out that the output is incorrect. The resulting waste of paper, toner, and time could be significant when calculated over a period of time on a large network.

Standardized feedback information from a printer would solve this problem. By the use of this standard, when a printer recognizes a condition that would prevent it from accurately printing a job, it can send a standardized message to a host computer that is monitoring network printing. Upon receipt of this message, the host could then send a message to the user who submitted the job, informing him or her of the error condition. The user could then redirect a job to a more appropriate printer or undertake action to correct the defect at the target printer. When projected over a period of time and a large number of network users, the resulting monetary savings could be substantial.

The example shown above is just one of many error conditions that could occur when printing either on a standalone computer or over a network. By using this standard format for exchanging information between the printer and host, software vendors, network suppliers, and printer manufacturers will now be able to greatly improve the efficiency of network printing.

At the time this standard was completed, the key contributors to the IEEE 1284.1 Working Group were as follows:

Forrest D. Wright, *Chair*
Jay Martin, *Secretary*
Michael Timperman, *Editor*

Chirag Bakshi
Michael Bringmann
Andy Davidson
Neal Fischer
Bob Garcia
Rod Gerhart
Tom Hasting
Tim Hathaway
Bob Herriot
Ken Hilliard
Barry Hills
Terry Jones

Theodore Kearley
Charles Kimber
Rick Landau
Harry Lewis
Rick Lomicka
Ernie Mandese
Stan McConnell
Ron Norton
Roman Orzol
Steve Price
Suzy Price
Paul Reilly
Dave Roach

Avi Schlank
Ron Smith
Gail Songer
Larry Stein
Mike Stilz
Koji Tashiro
Randy Turner
William Wagner
Craig Whittle
Keith Winter
Lloyd Young
Steve Zilles

The following persons were on the balloting committee:

Dharma P. Agrawal
Keith D. Anthony
Charles Brill
George Horansky
Thomas M. Kurihara
Donald C. Loughry
Raymond Lutz

Edward McCreight
Darrell B. McIndoe
James D. Mooney
Roman Orzol
Roy Reed
Larry C. Sollman

Larry Stein
Robert G. Stewart
Michael Timperman
Clarence M. Weaver, Jr.
Gregory Winston
Forrest D. Wright
Oren Yuen

The final conditions for approval of this standard were met on 29 July 1997. This standard was conditionally approved by the IEEE Standards Board on 26 June 1997, with the following membership:

Donald C. Loughry, *Chair*
Richard J. Holleman, *Vice Chair*
Andrew G. Salem, *Secretary*

Clyde R. Camp
Stephen L. Diamond
Harold E. Epstein
Donald C. Fleckenstein
Jay Forster*
Thomas F. Garrity
Donald N. Heirman
Jim Isaak
Ben C. Johnson

Lowell Johnson
Robert Kennelly
E. G. "Al" Kiener
Joseph L. Koepfinger*
Stephen R. Lambert
Lawrence V. McCall
L. Bruce McClung
Marco W. Migliaro

Louis-François Pau
Gerald H. Peterson
John W. Pope
Jose R. Ramos
Ronald H. Reimer
Ingo Rüsçh
John S. Ryan
Chee Kiow Tan
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal
Alan H. Cookson

Kristin M. Dittmann
IEEE Standards Project Editor

Trademarks

The trademarks in the following list are referred to on one or more pages of this standard. This list is provided as a matter of convenience and accommodation. By providing the list, IEEE takes no position on the legal status of any trademarks listed nor of any trademarks omitted from the list. While the information in the list is believed to be correct, it is based on information supplied by those who claim to own the trademarks. Consequently, IEEE disclaims liability for any inaccuracies in the list.

Adobe and Postscript are trademarks of Adobe Systems Incorporated.

Hewlett-Packard is a registered trademark of Hewlett-Packard Company.

IBM and Presentation Manager are registered trademarks of IBM Corporation.

Internetwork Packet Exchange, IPX, NCP, and NetWare Core Protocol are trademarks of Novell, Inc.

Microsoft and Windows are registered trademarks of Microsoft.

Novell and NetWare are registered trademarks of Novell, Inc. in the United States and/or other countries.

TrueType is a registered trademark of Apple Computer, Inc.

CLAUSE	PAGE
1. Overview	1
1.1 Scope	1
1.2 Purpose	1
1.3 Objectives	1
2. Definitions, abbreviations, and acronyms	2
2.1 Definitions	2
2.2 Abbreviations and acronyms	5
3. Software model	6
3.1 IEEE 1284.1 printer model	6
3.2 Printer job data flow	7
3.3 Printer sessions and jobs	9
4. The command/reply structure	11
4.1 General topics	11
4.2 The structure	12
5. Command set	15
5.1 Overview	15
5.2 Request Device Characteristics (RDC)	15
5.3 Request Interpreter Characteristics (RIC)	35
5.4 Job Control (JC)	44
5.5 Request Device Status (RDS)	58
5.6 Printer Configuration Control (PCC)	79
5.7 Request Logical Unit Characteristics (RLUC)	85
5.8 Printer Variable Commands (PVC)	87
5.9 Remote Operator Panel (ROP)	90
6. Alerts	100
6.1 Device Status Alert (DSA)	100
6.2 Interpreter Message Alerts (IMA)	102
6.3 Job Control Alerts (JCA)	104
6.4 Common Printer MIB Alert (CPMA)	107
6.5 Operator Panel Alert (OPA)	108

CLAUSE	PAGE
Annex A (normative) Conformance	109
Annex B (informative) Requirements for a Standard Information Mover Protocol with Link Encapsulation (SIMPLE)	110
Annex C (informative) Internetwork Packet (IP) transport mapping example.....	113
Annex D (informative) Novell® NetWare® transport mapping example.....	119
Annex E (informative) De-multiplexing alerts	122
Annex F (informative) Host ownership of an IEEE 1284.1 printer	123
Annex G (informative) Bibliography.....	124

IEEE Standard for Information Technology—Transport Independent Printer/System Interface (TIP/SI)

1. Overview

1.1 Scope

A standard protocol for the control of printers that is independent of the underlying data stream or page description language (PDL) used to create the printed page is defined in this standard. This protocol is usable by all classes of printers. The scope of this standard is limited to management and control of printers and does not include management or control of a printing system or subsystems.

1.2 Purpose

There is currently no defined, independent standard for controlling printers. Each vendor builds some control into the underlying PDL or data stream. Without an independent, openly defined protocol, applications and operating systems cannot automatically determine the type of printer being addressed. This protocol will provide a minimum implementation subset that will allow automatic identification and configuration of printers and vendor extensibility to provide for growth and product differentiation.

1.3 Objectives

This standard defines a protocol for communications between a host and printer. Its intent is to provide a standard methodology for software developers, computer vendors, and printer manufacturers that facilitates the orderly exchange of information between printers and host computers. A minimum set of functions that permit meaningful data exchange is defined. Thus, this standard establishes a foundation upon which compatible applications, computers, and printers can be developed, without compromising an individual organization's desire for design innovation. The following objectives accompany this standard:

- a) To simplify the printer driver development process by defining a standard set of command/response transactions between the host computer and printer.
- b) To accelerate the development of communicating printers by providing a robust protocol that can be implemented in phases ranging from basic to extended functionality.
- c) To ease customers' printing problems (especially over networks) by accelerating the availability of communicating printers and compatible host software.

- d) To assist software developers in minimizing time to market by establishing a base set of functions that ensure a minimum level of communications between the host and printer.
- e) To facilitate the creation of powerful network print management software by defining transactions that work across a wide range of printers.
- f) To enable the creation of standard control/communications firmware that can be included in many peripheral devices.
- g) To create a standard methodology for host and printer communications that is independent of the transport mechanism used between devices.
- h) To enhance the management of printers in networks by providing a mechanism for printers to readily provide their status and configuration to the host application.
- i) To permit design innovation by providing flexibility within the specification for printer manufacturers to include extensions to the original set of guidelines.
- j) To ensure cross-platform host-to-printer communications by creating an operating system-independent set of guidelines.

The resultant protocol is PDL-independent with the capability of a printer to support multiple PDLs, all active at the same time, if desired.

2. Definitions, abbreviations, and acronyms

2.1 Definitions

This clause provides certain terms and definitions used in this specification that may not be generally familiar or that may be used with a very specific meaning. These definitions are not intended to be absolute but rather to give the sense of the terms as used in the context of this standard.

2.1.1 American National Standard Code for Information Interchange (ASCII): In this standard, a text string of some arbitrary length. The text string may contain nonprintable characters. The length must be stated in another field. The encoding is typically ISO 8859-1, but is specified by the Printer Language field in a “Request Device Characteristics—Request Summary” response. *See also:* IEEE Std 100-1996.

2.1.2 ASCIIz: In this standard, an ASCII string concatenated with a NULL character.

2.1.3 big endian: A term used to imply that bytes in a word and words in a double word are transmitted most significant byte or word first in a serial stream of bytes.

2.1.4 bit: A single binary integer. A set bit represents a binary “1.” A cleared bit represents a binary “0.”

2.1.5 bit-encoded byte: A byte with a definition for each bit.

2.1.6 bit-encoded word: A word with a definition for each bit.

2.1.7 byte: An entity composed of 8 bits, used to define a unit element of memory or transmitted data. It is capable of describing integers in the decimal range –128 to 127.

2.1.8 command: A message from the host directed to the printer that may or may not include print data.

2.1.9 connection: A parallel interface state that is outside the scope of this standard and is not defined herein. This state is indicative of the state of the physical or logical connection between a host and the printer. Only in this state can data be transferred between a host and the printer.

2.1.10 console language: The human language in which information is to be displayed on local or remote consoles.

2.1.11 deprecated: Supported by predecessors to this standard but no longer used.

2.1.12 desktop management interface (DMI): A facility, normally host resident, for handling and translating defined interfaces for component information, event information, stored management information format (MIF) structures and management information. In this context, a component is an integral device or product, such as a printer; and event is an asynchronous alert or trap. The management information is used by an application such as a user management

program, or may be converted for use by a network management facility. The desktop management interface is controlled by the Desktop Management Task Force.

2.1.13 Desktop Management Task Force (DMTF): An association of software developers, host computer, and peripheral device manufacturers that promulgate a platform-independent interface standard for the management of desktop computers and the peripheral devices attached to such computers. In addition to defining the interfaces between components, the host-based service layer and management applications, the association supports the development of consistent management information format (MIF) structures that define the significant manageable and status attributes of various components incorporated into desktop computers.

2.1.14 document: An encoded, electronically transmittable image, set of images, or image-related information, which is handled by the printer interface control unit.

2.1.15 double word: A field composed of two words. In a message, the most significant word is transmitted/received first. It is capable of describing integers in the decimal range $-2\,147\,483\,648$ to $2\,147\,483\,647$.

2.1.16 duplex: For purposes of this standard, a type of printing that involves the process of creating images or impressions on both sides of the printing media.

2.1.17 facsimile: A process by which textual or pictorial images are communicated, typically but not exclusively, over telephone lines. The images may be coded in raster or compressed raster format (such as CCITT group 3) or in a page description language such as Adobe™ PostScript™. Facsimile typically operates down to the physical link level and includes protocols providing control and addressing mechanisms specific to the media being used. This is distinguished from the process of communicating similarly encoded images over local or wide area networks. However, both may be considered implementations of remote printing.

2.1.18 feed direction: On most printers, the direction that the medium is moved through the marking engine. For a printer in which the medium is not moved, the feed direction may be considered as along the Y axis. The across feed direction is the direction orthogonal to the feed direction; it is also called the crossfeed or scan direction on some printers.

2.1.19 finishing: An operation or group of operations performed on the printed media after it emerges from the printer output mechanism. Finishing includes operations such as stapling (stitching), punching, binding, folding, cutting, etc., which may or may not be considered part of the printing process. Note that operations of collating and sorting are normally considered printer output functions rather than finishing.

2.1.20 host: Whatever is driving (i.e., providing) commands or data to the printer (e.g., a workstation, a print server, or spooler).

2.1.21 impression: The process of marking the media. A single-sided, one-color printer requiring one pass per sheet would produce one impression per sheet. A similar printer printing duplex would produce two impressions per sheet. A two-pass printer providing a base color and a highlight color would produce two impressions per side, etc.

2.1.22 interpreter: A functional entity that translates one or more printer control or page description languages into a form suitable for the marking engine. Since printers sometimes emulate original implementations of these languages, interpreters are sometimes called emulations.

2.1.23 interpreter language: The printer machine language, or page description language, by which information to be imaged or to be used in imaging is coded.

2.1.24 job: That entity originated or initiated by a user which is handled by the printer interface control unit. A job need not result in the imaging of information on media.

2.1.25 kilobyte (kB): Equivalent to 1024 bytes.

NOTE — As used in this document, the term kilobyte (kB) means 2^{10} or 1024 bytes, megabyte (MB) means 1024 kilobytes, and gigabyte (GB) means 1024 megabytes.

There is a plan to deprecate this base-2 usage because of the potential for confusion with the base-10 definition. An alternative notation for base-2 is under development. Because the above mentioned base-2 usage is widespread in some areas of application, however, it may be employed in IEEE publications for a limited time.

2.1.26 legacy application or device: A printing application or device in existence prior to the existence of this standard. Hence, such device is unaware of the protocol defined herein.

2.1.27 link: The physical or logical connection between a host and a printer.

2.1.28 logical unit (LU): An addressable, functional group. In the case of a printer, scanner, or facsimile it is a functional group concerned with the storage, acquisition and/or processing of a textual and/or pictorial image. A printer may have one or more interpreters. The design of a particular printer determines if these interpreters are capable of concurrent operation.

2.1.29 management information base (MIB): A simple network management protocol (SNMP) compatible data structure that defines the functional groups and management objects of a unit or system. This standard includes a mechanism whereby the object values can be obtained from the printer by a device acting as the SNMP agent for the printer.

2.1.30 management information format (MIF): A desktop management interface (DMI) compatible data structure that defines the functional groups and management objects of a unit or system.

2.1.31 marking engine: A set of electrical and mechanical components that moves the print media and marks that media. In some implementations, a facsimile transmission function is considered to be a marking engine.

2.1.32 message: A logical grouping of one or more packets sent either from host to printer (a command message) or from printer to host (a response message).

2.1.33 message length: Although messages can be of any length up to 65 539 bytes, the packet size should be selected for effective transmissions over the physical link without requiring disassembly and reassembly. For connections through a network, the packet size of that network would generally be the most efficient.

2.1.34 multiple-packet error rejection: Error handling and rejection notification occurs on a message-by-message basis. This standard assumes the existence of a reliable transport layer protocol. Error detection and packet sequencing is a transport layer function and is beyond the scope of this standard.

2.1.35 network printing alliance protocol (NPAP): A transport independent printing protocol from which this standard has been derived.

2.1.36 NULL: A byte with all bits set to zero.

2.1.37 object identifier (OID): In general, a unique representation (name) of a manageable object defined in a management information base (MIB).

2.1.38 octet: An eight-bit data entity (byte).

2.1.39 packet: The basic message element used by this standard; a structured field, having a start byte, a two-byte length field (the first two bytes), a flag byte, a command byte, followed by the subcommand and/or data fields.

2.1.40 page: A logical representation of a single unit of printing media. It is a function of the document formatting rather than the printing process. There are one or more pages per impression. A “four-up” single-color printing will typically have four pages per impression.

2.1.41 page description language (PDL): A formal printer machine language, consisting of commands and data (or equivalently, operators and operands) used to specify and control the content and format of printed pages. A data stream encoded in a page description language is rendered into the printed page image by an interpreter.

2.1.42 print engine: That set of electrical and mechanical mechanisms that move the print media or paper and marks that paper.

2.1.43 printer: An intelligent device that includes, as a primary function, the ability to convert an electrically transmitted or stored image into a physical image formed by colorant on some medium (such as paper).

2.1.44 printer interface control unit (PICU): The set of electronics that interfaces external communications ports, common peripheral interfaces (such as font cards or disk drives), the logical units, and the marking engine. It is the function of the printer interface control unit to coordinate and sequence all the functions and operations of the printer.

2.1.45 printer language: The human language used for the American National Standard Code for Information Interchange (ASCII) strings within all command and response messages, other than those to be printed or those to be displayed on the local or remote consoles (e.g., English, French, German).

2.1.46 printing: That set of operations implemented by the printer that results in an image rendered as marks on the selected media.

2.1.47 print media: That consumable upon which the marking engine marks so as to form a text and/or pictorial image, typically paper.

2.1.48 reply: Messages from the printer to the host. Syn.: response.

2.1.49 session: A printer state that allows the logical grouping of one or more jobs into a sequential, referenceable collection.

2.1.50 sheet: A cut piece of print media, such as a sheet of paper.

2.1.51 simple network management protocol (SNMP): A protocol used for the management of network nodes and devices, used extensively on internet and other networks. The protocol provides for the communication of status and setup information between a management console and a managed device using values of objects defined in the management information base (MIB) for the managed object.

2.1.52 space character: A byte (hex 20) used in text strings that represents a space.

2.1.53 start of packet byte: A single byte (hex A5, decimal 165) that is used by both the printer device and the host to quickly determine whether or not they are synchronized.

2.1.54 transport or transport layer: The middle layer in the ISO seven-layer open system communications reference model, and the boundary between the communication subnet layers (physical, data link, and network) and the host process layers (session, presentation, and application).

2.1.55 unsigned byte: A byte that represents positive integers in the decimal range 0–255.

2.1.56 unsigned dword: A dword that represents positive integers in the decimal range 0–4 294 967 295.

2.1.57 unsigned word: A word that represents positive integers in the decimal range 0–65 535.

2.1.58 word: A field composed of two eight-bit bytes. In a byte serial message, the most significant byte is transmitted/received first, (big endian). It is capable of describing integers in the decimal range –32 768 to 32 767.

2.2 Abbreviations and acronyms

ASCII	American National Standard Code for Information Interchange
bbyte	bit-encoded byte
bword	bit-encoded word
CPMA	common printer MIB/MIF alerts
DMI	desktop management interface
DMTF	Desktop Management Task Force
DSA	device status alerts
dword	double word
IETF	Internet Engineering Task Force
IMA	interpreter message alerts
JCA	job control alerts
kB	kilobyte

NOTE — See NOTE under “kilobyte” in definitions.

LAN	local area network
LU	logical unit
MIB	management information base
MIF	management information format
NPAP	network printing alliance protocol
OID	object identifier
PDL	page description language
PICU	printer interface control unit
RDC	request device characteristics
RFC	Request For Comments (as defined by the Internet Engineering Task Force)
TCP/IP	Transmission Control Protocol/Internet Protocol
TIP/SI	transport independent printer system interface
udword	unsigned dword
uword	unsigned word

3. Software model

3.1 IEEE 1284.1 printer model

A printer can be thought of as a minimum of three logical entities: The print engine, the printer interface control unit (PICU), and one or more logical units (LUs) (one of which is an interpreter). Figure 1 shows a logical model of the IEEE 1284.1 printer.

It is important to note that while an interpreter is an LU, the converse is not always true. An LU can be another logically separate device in the printer such as a facsimile engine or document scanner.

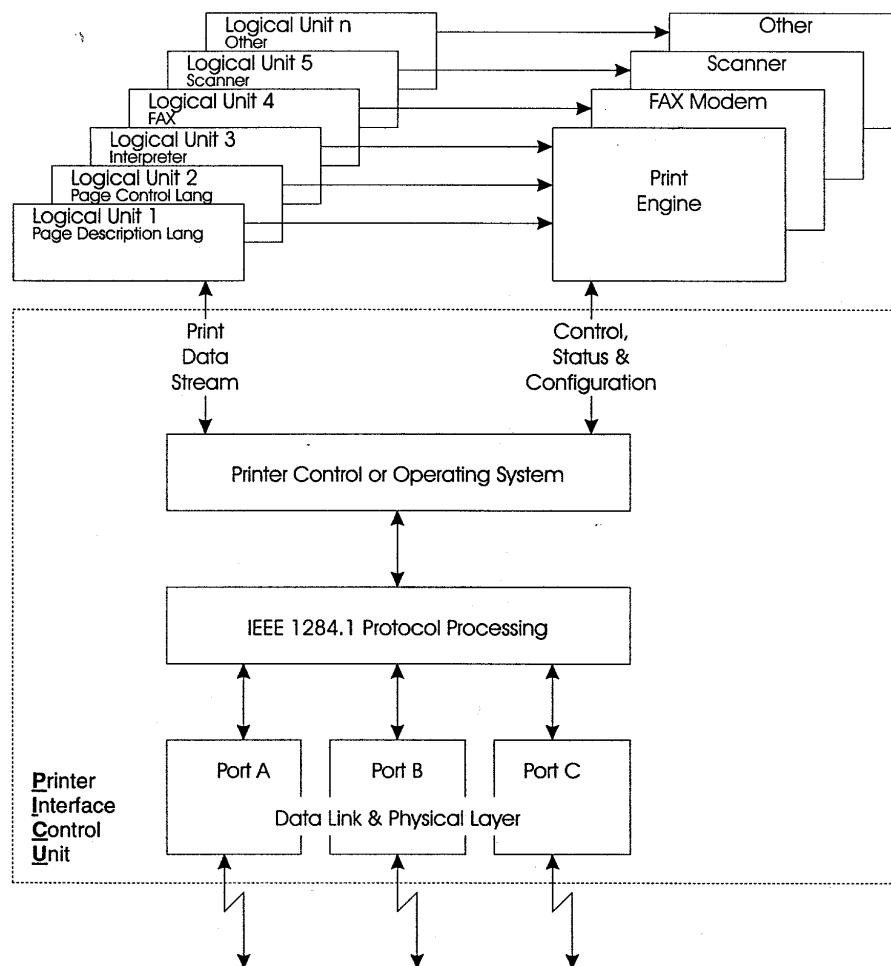


Figure 1— Printer model

The management of producing a printed document is complex. The task can be divided into two overlapping pieces, the management of printing and the management of the printer. Printing encompasses the entire process of producing a printed document: generation of the file to be printed, selection of a printer, choosing printing properties, routing, queuing, resource management, scheduling, and final printing including notification of the user. This standard enables communications of printer status and attributes that, when used by applications, greatly enhance both the management of printing and of the printer.

In this standard, commands have been defined that allow the manipulation of manageable objects that are contained in a printer and are capable of being referenced using an OID such as those print objects defined in the IETF Printer MIB, RFC1759[B6].¹ Job delivery is outside the scope of RFC1759, as it has chosen only to cover the management of the printer itself. In doing so, RFC1759's printer model differs from the one presented in Figure 1. The RFC1759 model is basically an exploded view of the print engine and control blocks used by this standard.

The function of this standard's protocol is to provide a means of returning configuration and status information in a manner that is independent of the printer's physical connection, imaging technology, or embodied LU(s). In this model, the printer is intended to be managed by some intermediate system element external to the printer. The protocol provides a method for

- Retrieving traditionally static information, such as the number and type of paper input and output trays, imaging resolution and speed, interpreter capabilities and their relationship to the print mechanism, etc.
- Returning real-time status information related to the print mechanism that is independent of the LUs, such as paper and printing supplies levels.
- A separate, logical, out-of-band communication channel for exchanging commands and responses. This out-of-band channel is key to the effective management of the printer while page processing is being done.
- Selection or activation of an LU via host control.
- Job separation.
- Reporting job statistics independent of the LU.

The protocol has been designed to be link independent. The only requirement is that the link must be capable of bidirectional data transmission. Numerous methods and media exist today to facilitate this transport of information. Included are various LAN protocols (ISO/IEC 8802-2: 1994, TCP/IP, etc.), asynchronous serial communications, and, more recently, the bidirectional parallel port defined by IEEE Std 1284-1994 [B3].

3.2 Printer job data flow

The printer job data flow model is used to visualize the sequence of events for printing and for the identification of the position of a particular job in the printing process. A generalized model of the data flow for a job is shown in Figure 2. Data is received by the protocol processing layer from the data link and physical layer. The model assumes that the printer is a simple buffered device and handles print data via a simple, single-threaded queue.

The first action taken by the printer is to determine if the received data is directed to an LU or is a command. This decision is facilitated by examining the flag byte in the packet header. If the data packet contains a command, it is placed in a command buffer. With the exception of the Start Session, Start Job, End Job, and End Session commands, all commands are acted on immediately. The size of the command buffer is dependent on the specific printer implementation.

¹The numbers in brackets preceded by the letter B correspond to those of the bibliography in Annex G.

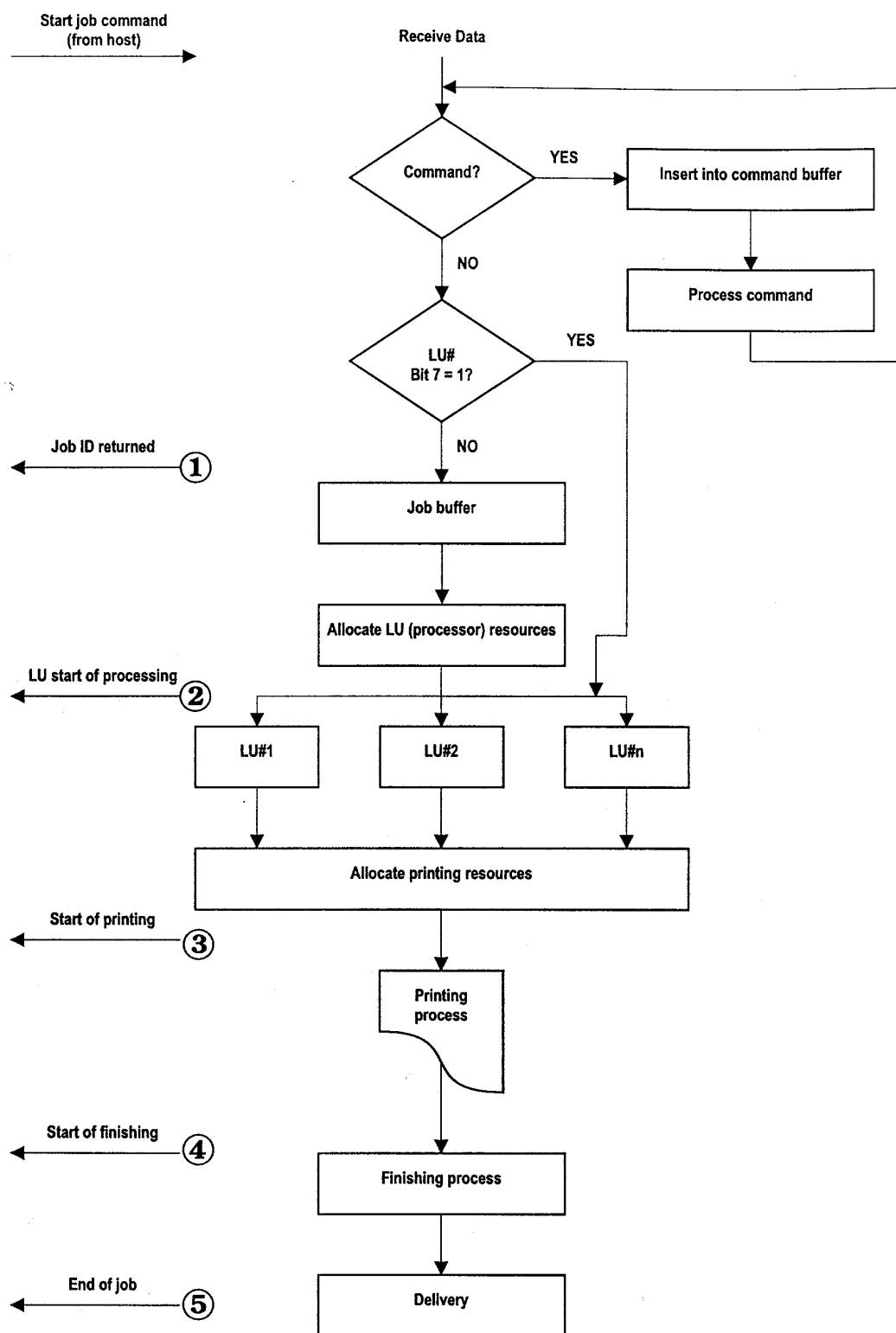


Figure 2— Job data flow model

When the data is LU destined, the higher order bit (bit 7) of the LU address is used to determine if the data is to be put into the job buffer or sent immediately to the active LU, bypassing the buffer.

As data is removed from the job buffer, the printer activates the appropriate LU. It is important to note that one or more LUs may be active at any point depending on printer implementation. When the LU finishes processing or reaches an appropriate point, the printer control system then allocates the print engine to the LU. For this discussion, it is assumed that a printer has only one print engine available for use, even if there are multiple LUs processing data simultaneously. Once the document has been printed, any available and requested finishing operations are performed. Upon completion of any document finishing, the document is marked as delivered and the end of job signaled to the host if requested.

The preceding explanation is a highly simplified and sequential model of the printing process. In an actual implementation most of the steps of starting an LU, processing data, printing data and finishing can be overlapped. The start of LU processing may imply that printing and or finishing resources are allocated simultaneously. It is not required that one step in the printing process be completed before the next is started.

3.2.1 Job data flow model

The IEEE 1284.1 protocol provides the ability for the host system to coordinate and report on the status of the various steps in the printing process so that effective printer and job management can be implemented. The model is provided as an aid to the visualization of the printing process and does not dictate any particular implementation or control system for a printer. The circled numbers on the job data flow model, Figure 2, show status points with the following brief description as to their meaning.

- 1) Job ID returned.
This is the first point of synchronization where the printer returns a printer assigned job identification number to the host after a Start of Job command has been received from the host. The job ID allows the host to track a particular job.
- 2) LU start of processing.
This is a synchronization point where the printer has allocated the resources necessary to run a job on a particular LU and is about to begin processing the job. The host has the option of suspending a print job at this point and interacting with the printer to determine if the appropriate resources are available to complete the particular job that is about to be processed by the LU. Checking for a particular font would be an example of an action that could be taken at this point.
- 3) Start of printing.
This is a synchronization point where the printer is about to start physically printing the first page of the job. The host has the option of suspending a print job at this point to interact with the printer to determine if the appropriate resources are available to complete the printing, such as specific media installed in the printer.
- 4) Start of finishing.
This is a synchronization point where the host has the option of suspending a print job to interact with the printer to provide detailed instructions on final finishing of the document.
- 5) End of job.
This is the final synchronization point where the print job is complete and job statistics are returned to the host regarding the print job.

3.3 Printer sessions and jobs

3.3.1 Session and job concepts

Within the IEEE 1284.1 framework the basic logical grouping of data is the job. Jobs are delineated in the data stream by the Start Job and End Job commands. The Start Job command resets accounting statistics and may cause the printer to initialize the graphic engine. The End Job command instructs the printer to log the accounting statistics, and release any nonpersistent job-related resources.

A session allows the logical grouping of one or more jobs into a sequential, referenceable collection. Empty jobs and empty sessions are valid. Sessions are delineated in the data stream by the Start Session and End Session commands. The Start Session command is an indicator to the printer that a collection of jobs follows. The printer shall use this command to lock port rotation in a multiple link and/or multiple protocol environment. The End Session command marks the end of the collected data, causes the printer to print and eject any remaining sheets, and allows link rotation.

Because the session commands directly affect link rotation, grouping unrelated jobs in the same session could lead to an unfair allocation of printing resources. Therefore, it is recommended that a session not contain a collection of unrelated jobs. This recommendation is not an attempt to limit the number of jobs in a session. It is a recommendation that the host should refrain from concatenating unrelated jobs together in a session to circumvent link rotation.

3.3.2 Session Data Flow Model

Once a physical or logical connection is established between the host and printer, a start session command prompts the printer to enter an “in-session” state and return an identifying number for the session. While the printer is in this “in-session” state, the Start Job and End Job commands are used to delineate related print data. Job statistics are accumulated for this delineated data and are provided to one or more hosts. (See Figure 3.)

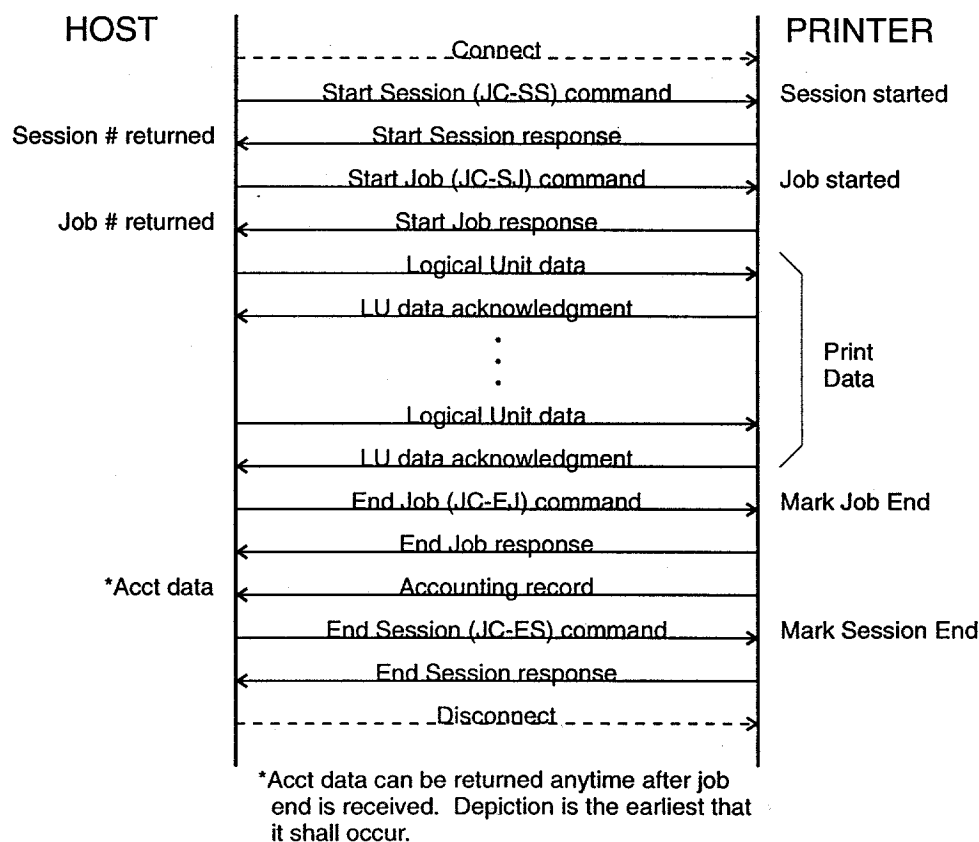


Figure 3— Session and job chronological data flow

4. The command/reply structure

4.1 General topics

4.1.1 Commands

Commands are host to printer messages instructing the printer to perform some action.

4.1.2 Responses

Responses are printer to host messages sent as the direct result of a command from the host. Every message sent to the printer with the reply required flag (flag bit 4) set shall result in a response from the printer. In the case where the printer has no data to return to the host, a short acknowledgment consisting of the original command with the flags appropriately updated shall be returned.

4.1.3 Alerts

An unsolicited printer to host message is called an alert. These messages are initiated by exception conditions detected in the printer. These events can occur either asynchronously to the command stream (e.g., out of toner/paper/supplies, cover open, etc.) or synchronously to the command stream (e.g., erroneous command). In either case, an alert is issued.

The various types of alerts are as follows:

- a) Device status alerts (DSA), resulting from
 - 1) Events that occur inside the printer asynchronous to the data stream (i.e., out of toner/paper/supplies, cover open, job completed, etc.)
 - 2) Message errors (i.e., “invalid logical unit,” etc.), when the host did not expect a response (flag bit 4 cleared in message from host to printer)
- b) Interpreter message alerts (IMA), resulting from interpreter responses to data stream queries
- c) Job control alerts (JCA), resulting from
 - 1) A print job completed
 - 2) A print job starting logical unit processing
 - 3) A sheet from a print job being stacked in the output device
- d) Common printer MIB/MIF alerts (CPMA), resulting from a critical alert entry (as defined by RFC1759 [B6]) being added to the printer’s alert table.

4.1.4 General flow

In general, the host is the master and initiates all exchanges with the printer. With the exception of defined alerts, the printer shall not send an unsolicited response to the host. Solicited responses require that the reply required flag is set in the received message.

It is recommended that the host communicate with the printer with the reply required flag set in all commands sent to the printer. In this mode the host sends each message (reply required flag set) to the printer and waits for a positive acknowledgment (printer response) before sending the next message.

When a command with the reply required flag set is received, indicating that the message contains data to be delivered to a LU, the printer must send a simple response acknowledging that the data was received. Any printer response action required as a result of the LU processing data in the message is defined to be unsolicited. The decision to return this unsolicited message to the host is governed by a prior command that enabled interpreter message alerts. If these alerts are enabled, the requested response will be returned to the host as an alert (flag bit 4 cleared) from the interpreter (flag bit 6 cleared, command field = interpreter id) directly following an Interpreter Message Alert message.

4.1.5 Guidelines for manufacturer-unique commands

Vendors may define and implement their own unique commands, or “extensions,” to IEEE Std 1284.1-1997. The vendors who implement their own commands can still be classified as compliant to the standard, but only if they have implemented the IEEE 1284.1 commands in accordance with this standard. Support for the unique commands is the responsibility of the defining vendor.

Compliant devices shall reject properly structured commands that are not implemented in the device by setting the flag bits to indicate a data error in the response message sent back to the host. A vendor defining new manufacturer-specific commands shall comply with the architecture of commands/subcommands, packet structure, field definitions, general flow and all other protocols defined in this standard. Vendor compliance shall allow existing compliant devices to parse and properly reject the extended commands that they do not implement.

4.2 The structure

4.2.1 Basic packet structure

The basic structure for all packets (host to printer and printer to host) is of the form shown in Table 1.

Table 1— Basic packet structure

Packet header				Data field
Start of Packet Byte	Packet length	Flag	Command or LU #	Data...

4.2.2 Definition of command fields

4.2.2.1 Start of Packet Byte (byte 1)

This byte (hex A5) is used by the host to indicate a start of packet. When the printer is in a mode to receive a packet from the host, the first byte received must be a Start of Packet Byte. If this byte is not the first byte received, the printer will know immediately that it is out of sync with the host or is not receiving commands or data compliant with this standard.

4.2.2.2 Packet length (bytes 2 and 3)

Packet length is the number of bytes in this packet not including this length field or the Start of Packet Byte (16-bit unsigned word). The default “maximum packet size” for commands is 64 bytes. This value shall be used until the host determines the printer’s exact capabilities via a Request Device Status.

4.2.2.3 Flag (byte 4)

The purpose of the flag is to provide a single byte that the printer can examine to obtain control information. The flag bits are defined in Table 2.

Table 2— Command flag bits

Bit 7	No Operation (NOP). If this bit is set, it indicates that the host requires the printer to disregard (i.e., not process) this message regardless of its contents. If the Reply Required bit is set, the printer is still required to acknowledge the receipt of this packet (see bit 4). If this bit is cleared, it indicates that the printer is to process the command.
Bit 6	Destination (IEEE 1284.1 Component or LU #). If set, this bit indicates that the message is for the bidirectional software component in the printer and the fifth byte in the message contains the command op code. If cleared, it indicates the message contains data for an LU with the fifth byte in the message being the logical unit number that is to receive the data.
Bit 5	Continue. If set, this bit indicates that the next packet is a continuation of this message. If cleared, it indicates that this message is “complete.”
Bit 4	Reply Required. The host sets this bit when it requires a response from the printer. If this bit is set, all commands sent from the host to the printer will be acknowledged by the IEEE 1284.1 component within the printer. If the command is data for an LU, the printer’s IEEE 1284.1 component will acknowledge the data as being received. The LU will not respond to the Reply Required flag. If the NOP bit is set, the printer’s IEEE 1284.1 component will acknowledge the data as being received and discarded (byte #5 of the printer response will have the same value as byte #5 of the host to printer command). If this bit is cleared, the IEEE 1284.1 component will only respond when errors are detected. The responses will be sent as unsolicited messages to the host.
Bit 3	Undefined (reserved)
Bit 2	Undefined (reserved)
Bit 1	Undefined (reserved)
Bit 0	Undefined (reserved)

4.2.2.4 Command byte (byte 5)

The meaning of the command byte is dependent on the state of bit 6 in the flag byte.

If bit 6 is set, this field contains a command for the IEEE 1284.1 command processor. Commands 0–127 (hex 0 to 7F) and 240–255 (hex F0 to FF) are reserved for IEEE 1284.1 commands and future extensions. Commands 128–239 (hex 80 to EF) are available for manufacturer- or device-specific assignment. Assignment of value usage in this range is not managed by the IEEE.

If bit 6 is cleared, this field contains the logical unit number destination for this message. A value of 0 is used to select the default LU. The default LU is defined as the LU compatible with today's machine definition. For example, it may be the power on default interpreter that can be overridden by the front panel or the data stream (as in emulation switching).

4.2.2.5 Data bytes (bytes 6, 7,...)

The meaning of the data bytes are dependent on the state of bit 6 in the flag byte.

If bit 6 is set, this field contains, depending on the command, additional command information and/or data for the IEEE 1284.1 command processor.

If bit 6 is cleared, this field contains data for the logical unit.

4.2.3 General printer to host protocol considerations

4.2.3.1 Command error

If a command is received by the printer that is not understood (i.e., undefined), the printer shall set the error flag (bit 7), set bit 3, and clear bit 2 in the packet header of the response. The printer shall perform no other action on that command other than to flush it from internal buffers. If the host thinks that the printer should have accepted the command, it shall retry the command again.

4.2.3.2 Rejecting a command

Both the host and the printer shall always be able to accept at least one full packet at any time. This packet does not necessarily need to be acted upon. A message sent from the host to the printer can be rejected by the printer. To indicate that a given packet has been rejected, the printer shall set the error flag (bit 7) and clear both error type flags (bits 2 and 3) in the packet header for the reply message. At this point, the host can determine why the printer could not execute the command by issuing an Request Device Status command, which is *always* honored. The host can also simply keep sending the command until it is accepted and operated upon.

4.2.3.3 Data error

If a command is received by the printer that has data that is not understood (i.e., undefined), the printer shall set the error flag (bit 7), set bit 2, and clear bit 3 in the response sent back to the host. The printer shall perform no other action on that command other than to flush it from internal buffers. If the host thinks that the printer should have accepted the command, it shall retry the command again.

4.2.4 Definition of response fields

4.2.4.1 Start of Packet Byte (byte 1)

This byte is used by the printer to indicate a start of packet. When the host is in a mode to receive a packet from the printer, the first byte received shall be a Start of Packet Byte. If this byte is not the first byte received, the host immediately know that it is out of sync with the printer.

4.2.4.2 Packet length (bytes 2 and 3)

Packet length is the number of bytes in this packet not including the length field or the Start of Packet Byte (16 bit, unsigned word). The default "maximum packet size" for responses is 64 bytes. This value shall be used until the host tells the printer otherwise using the Printer Configuration Control command.

4.2.4.3 Flag (byte 4)

The purpose of the flag is to provide a single byte that the host can examine to obtain control information for the message and a quick view of the printer's status. The flag bits for a printer response are defined in Table 3.

Table 3— Response flag bits

Bit 7	Error. If set, this bit indicates that this response contains error information. This error flag is used to indicate a command or data error for the command sent to the printer.
Bit 6	Source (IEEE 1284.1 Component or LU #). If set, this bit indicates that the message is from the IEEE 1284.1 command processor in the printer and the command field contains the command op code to which the printer is responding. If cleared, it indicates the message contains data from an interpreter and the command field contains the logical unit number of the specific interpreter that generated the data.
Bit 5	Continue. If set, this bit indicates that the next packet is a continuation of this message. If cleared, it indicates that this message is “complete.”
Bit 4	Reply Type. If this bit is set, it means that this reply was required by the host (i.e., solicited via bit 4 in the message from the host). When cleared, this bit indicates that this message was not solicited by the host but was caused by an “error” event at the printer. The error condition is asynchronous and can occur at any time.
Bit 3..2	Command/data check type, encoded as follows: NOTE—These bits are valid only when the Error flag (bit 7) is set. Bits 7..3..2 Definition when contained in a message from the printer. 0xx No errors were detected in the command or data fields. 100 No errors were detected in the command or data fields, but the command was rejected. 101 Data Error 110 Command Error 111 Reserved
Bit 1..0	Printer Status. These two bits are set to the state of the printer at the time of the reply, independent of whether or not they were selected (i.e., “armed”) by the PCC command. The bits are defined as follows: Bits 1..0 Definition when contained in a message from the printer. 00 Printer is fine; no alert condition exists. 01 Printer is still OK, but a condition exists that the operator may want to know about (a warning—low paper, job end, etc.). 10 A condition exists that will prevent further printing, but the operator can fix it (operator intervention required—out of paper, output bin full, toner low, etc.). 11 A condition exists that will prevent further printing and a service/repair call is required to fix it (laser failure, fuser died, system board error, etc.).

4.2.4.4 Command byte (byte 5)

In a reply, the original op code or interpreter ID of the message that caused the reply shall be returned. If the message was unsolicited from the IEEE 1284.1 command processor (i.e., bit 4 is cleared and bit 6 is set), this field shall be set to a value in the range 240–255 (hex F0 to FF). Currently 255 (hex FF) is used to identify Device Status Alerts, 254 (hex FE) is used to identify Interpreter Message Alerts, 253 (hex FD) is used to identify Job Control Alerts, 252 (hex FC) is used to identify Common Printer MIB Alerts, 251 (hex FB) is used to identify Operator Panel Alerts, and 240 (hex F0) is reserved for manufacturer-specific Device Status Alerts.

4.2.4.5 Response data bytes (bytes 6, 7,...)

The data bytes shall be interpreted as defined in Table 4.

Table 4— Interpretation of response data bytes 6, 7,...

Flag Bits 6..4 (Byte #4)	Command (Byte #5)	Data (Bytes #6–n)
00 Message from an Interpreter	LU #	Message from the interpreter identified in byte #5
01 Required acknowledgment from IEEE 1284.1 command processor that data for an interpreter was received and buffered	LU #	N/A
10 Alert	240 Manufacturer-Specific Alert 251 Operator Panel Alert 252 Common Printer MIB/MIF Alert 253 Job Control Alert 254 Interpreter Message Alert 255 Device Status Alert	Alert data
11 Required response from bidirectional component	IEEE 1284.1 command	Command Response as defined in command details for the IEEE 1284.1 command indicated by byte #5

5. Command set

5.1 Overview

The defined commands and their hexadecimal equivalents are shown in Table 5.

Table 5— Commands

Command	Hexadecimal equivalent	Status
Request Device Characteristics	01	current
Request Interpreter Characteristics	02	current
Printer Configuration Control	03	current
Request Device Status	04	current
Job Control	05	current
Request Logical Unit Characteristics	06	current
Control Packet Rejection	07	deprecated
Common Printer Variable	08	current
Remote Op Panel	09	current

5.2 Request Device Characteristics (RDC)

5.2.1 Overview

The RDC command provides the host system with a method of obtaining information concerning the capabilities and characteristics of the printer. The host system would use this information to communicate to the application such data as paper size and output capabilities such as sorting. Additionally, the host could use this information for automatic scheduling and routing applications, such as that provided by print servers on a local area network.

Using the RDC command, there are various categories of characteristics information that can be obtained. Table 6 shows the various categories.

Table 6— RDC categories

Category	Description
Summary	Provides for an overview of the printer and its capabilities.
Input	Describes the printing media, paper, input capabilities, and features of the printer.
Output	Describes the printing media, paper, output capabilities, and features of the printer.
Options	Description of the various options contained completely within the printer such as expanded memory, mass storage or communications functions, etc.

The host queries for the desired information by sending a RDC command with a subcommand in the data field that specifies the category of information desired. The subcommand is a single unsigned byte. The subcommand values for the various categories of characteristics are shown in Table 7.

Table 7— RDC subcommands

Subcommand	Characteristic description
00	Summary information
01	Reserved
02	Input characteristics
03	Output characteristics
04	Option characteristics
05 to FF	Reserved

5.2.2 Request Summary

The RDC Request Summary subcommand (RDC-RS) provides the host system with an overview of the capabilities of the printer. Using the data returned by the printer, the host knows how to enquire about the details of other features, such as input and output capabilities. Table 8 shows the format of the command sent to the printer.

Table 8— RDC-RS command

Bytes	Value (hex)	Description	Notes
5		Request device characteristics	Packet header
1	00	Request summary	Data field

After the printer receives the command, it will return a summary data packet back to the host as shown in Table 9.

Table 9— RDC-RS response

Bytes	Value (hex)	Description	Notes
5		Request device characteristic	Packet header

Bytes	Value (hex)	Description	Notes
1	00	Request summary	Data field
1	ubyte	Major revision level of standard supported	
1	ubyte	Minor revision level of standard supported	
1	ubyte	Printer specific extension revision level supported	
1	ubyte	Marking technology	
1	ubyte	Color capabilities	
2	uword	Number of monochrome/color levels supported	
1	bbyte	Duplex capabilities	
2	uword	Maximum number of entries in query job(s) completed queue	
1	ubyte	Speed units definition	
1	ubyte	Speed of printer in speed units	
1	ubyte	Length units definition	
2	uword	Horizontal logical units per length unit	
2	uword	Vertical logical units per length unit	
1	ubyte	Counter units	
1	ubyte	Reserved	
4	udword	Total installed memory (bytes)	
2	uword	Maximum receive packet size	
2	uword	Maximum number of outstanding commands	
1	00	Deprecated, must be zero	
1	ubyte	Number of logical units supported	
1	ubyte	Number of inputs supported	
1	ubyte	Number of outputs supported	
1	ubyte	Number of options supported	
1	ubyte	Printer language and character encoding	
1	ubyte	Length of product name (not including this byte)	
n	ASCII	Product name	
1	ubyte	Length of product revision (not including this byte)	
n	ASCII	Product revision	
1	ubyte	Length of serial number (not including this byte)	
n	ASCII	Serial number	
2	uword	Maximum receive command packet size	

5.2.2.1 Major/minor revision level of standard supported

The combination of these two unsigned bytes specify the revision level of the standard supported by this printer. For this standard, the major revision level is 2 and the minor revision level is 0.

5.2.2.2 Printer-Specific extension revision level supported

This byte is used by the printer to indicate the presence of additional capabilities that are extensions to the standard and are unique to this device. If the value of this byte is 0, then there are no extensions to the standard. A value greater than zero indicates the presence of extensions that are specific to that manufacturer of the printer. The significance of the value of this byte, other than 0, is dependent on the manufacturer of the printer.

5.2.2.3 Marking technology

This unsigned byte defines the type of marking technology used by the printer. The host system can use this field for scheduling and printer type selection. The values for this field are defined in Table 10.

Table 10— RDC-RS Marking technology codes

Value	Description
00	Electrophotographic (LED, laser, other)
01	Impact—moving head—dot matrix (9 wire, 24 wire)
02	Impact—moving head—formed
03	Impact—band
04	Ink jet—aqueous
05	Ink jet—solid
06	Pen
07	Thermal transfer
08	Thermal sensitive
09	Thermal diffusion (sublimation)
0A	Electrostatic
0B	Photographic (microfiche, imagesetter, other)
0C	Ion deposition
0D	E-beam
0E	Typesetter
0F	Electroerosion
10	Ink jet—other
11	Thermal—other
12 to FD	Reserved
FE	Other
FF	Unknown

5.2.2.4 Color capabilities

This unsigned byte defines the color capabilities of the printer. The host system can use this field for scheduling and printer type selection. This field has encoded meanings as defined by Table 11.

Table 11— RDC-RS color capabilities codes

Value	Description
00	Monochrome
01	Spot color
02	3-primary (CMY) color
03	4-primary (CMYK) color
04 to FD	Reserved
FE	Other
FF	Unknown

5.2.2.5 Monochrome/color levels supported

This unsigned word defines the number of color or monochrome levels that are supported by the printer.

For monochrome printers, this field contains the number of distinct gray scale levels supported at the printer's maximum resolution, not including "white" or "unmarked." For example, a black/white printer with no gray-scale capability should report 1 in this field.

For spot-color printers, this field contains the number of spot colors supported, not including "white" or "unmarked." For example, a black/white/red printer should report 2 in this field.

For 3-primary and 4-primary color printers, this field contains the maximum number of distinct levels, not including "white" or "unmarked," of any one primary at the printer's maximum resolution. Note that the number of distinct

colors achievable by combining primaries may be significantly less than the product of the number of levels in each primary.

5.2.2.6 Duplex capabilities

This bit-encoded byte defines the duplex capabilities of the printer. The host system can use this field to determine whether or not the printer supports duplex printing. This field has encoded meanings as defined by Table 12. The absence of any set bits indicates that the printer is not capable of supporting two-sided printing.

Table 12— RDC-RS duplex capability encoding

Bit	Description
0	Short edge binding duplex capable (zero=No, one=Yes)
1	Long edge binding duplex capable (zero=No, one=Yes)
2–7	Reserved

5.2.2.7 Maximum number of entries in query job(s) completed queue

This unsigned word defines the maximum number of entries that the printer will retain in its Query Job(s) Completed Queue. (See 5.4 for further information.)

5.2.2.8 Speed units definition

This unsigned byte defines the units that a printer uses when reporting various printer speed capabilities. This field is used together with the value of the byte that returns the printer speed. Table 13 shows the encoded meanings for this field.

Table 13— RDC-RS speed unit codes

Value	Description
00	Impressions or sheets per minute
01	Inches per second
02	Feet per second
03	Yards per second
04	Millimeters per second
05	Centimeters per second
06	Decimeters per second
07	Meters per second
08	Characters per second
09	Tens of characters per second
0A	Hundreds of characters per second
0B	Thousands of characters per second
0C	Lines per minute
0D	Tens of lines per minute
0E	Hundreds of lines per minute
0F	Thousands of lines per minute
10 to FD	Reserved
FE	Other
FF	Unknown

5.2.2.9 Speed of printer in speed units

This unsigned byte defines the maximum speed or how fast the printer can print. The printer speed is expressed in speed units as defined in Table 13. An example of the speed of a printer would be 26 impressions per minute.

5.2.2.10 Length units definition

This is an unsigned byte that defines the units used by the printer when reporting values where a length is involved with the definition. Examples of this would be resolution, 300 dots per inch (dpi), or paper width of 11 in. Table 14 defines the values associated with the various units of length.

Table 14— RDC-RS length unit codes

Value	Description
00	0.001 in
01	0.01 in
02	0.1 in
03	Inches
04	0.01 mm
05	0.1 mm
06	Millimeters
07	Centimeters
08 to FD	Reserved
FE	Other
FF	Unknown

5.2.2.11 Horizontal logical units per length unit

This unsigned word defines the logical units per length unit in the horizontal direction. Horizontal direction is defined to be orthogonal (90 degrees) to the feed or motion of the printing media. Logical units have no physical dimension.

5.2.2.12 Vertical logical units per length unit

This unsigned word defines the logical units per length unit in the vertical direction. Vertical direction is defined to be in-line with the feed or motion of the printing media. Logical units have no physical dimension.

5.2.2.13 Counter units

This unsigned byte defines the unit of measurement used when reporting printer statistics. Table 15 shows the encoding for this field.

Table 15— RDC-RS counter unit codes

Value	Description
00	Pages
01	Inches
02	Feet
03	Yards
04	Millimeters
05	Centimeters
06	Decimeters
07	Meters
08	Characters
09	Tens of characters
0A	Hundreds of characters
0B	Thousands of characters
0C	Lines
0D	Tens of lines
0E	Hundreds of lines
0F	Thousands of lines
10 to FD	Reserved
FE	Other
FF	Unknown

5.2.2.14 Total installed memory

This unsigned dword specifies the total number of bytes of random access memory (RAM) installed in the printer. This value reflects the total amount of RAM and in no way implies how much of that memory is available for various functions within the printer such as communications buffers, font storage, form storage, etc. (The amount of memory available to the user using a given interpreter is available to the host through the Request Interpreter Characteristics command.) If the printer is unable to provide this information, the hex value FFFFFFFF will be returned.

5.2.2.15 Maximum receive packet size

This unsigned word defines the maximum size (in bytes) of a data packet that can be received by the printer. If the printer receives a packet that is larger than the value specified, the printer shall reject the packet. A value less than hex 40 is not supported.

5.2.2.16 Maximum number of outstanding commands

This unsigned word defines the number of commands that the printer can receive and be working on at any one time. An outstanding command means any command from the host where the host expects a response from the printer and for which the printer has not yet responded. If the host sends more commands than defined by this value, the printer shall reject the command and the host will have to re-send the command. A value of zero received in this field is defined to be equivalent to a value of one.

The printer's responses to multiple commands from the host should be in the same order in which the commands were received.

5.2.2.17 Number of logical units supported

This field contains the number, or count, of logical units (LUs) supported by the printer. The host can determine the LU type (e.g., page interpreter) by using the Request Logical Unit Characteristics command.

A value of 255 (hex FF) for this field indicates that the printer does not know how many interpreters are supported.

5.2.2.18 Number of inputs supported

This unsigned integer contains the number, or count, of printing media (paper) inputs that the printer currently supports. This number may change as optional input devices are added and removed from the printer. The host can inquire about the details of each input by using the Request Input Characteristic subcommand of the RDC command.

A nonzero value for this parameter indicates that the printer shall support the Request Input Characteristics subcommand of the RDC command.

A value of 255 (hex FF) for this field indicates that the printer does not know how many inputs are supported.

5.2.2.19 Number of outputs supported

This unsigned byte contains the number, or count, of output destinations for the printing media (paper) that the printer currently supports. This number may change as optional output devices are added and removed from the printer. The host can enquire about the details of each output by using the Request Output Characteristic subcommand of the RDC command.

A nonzero value for this parameter indicates that the printer shall support the Request Output Characteristics subcommand of the RDC command.

A value of 255 (hex FF) for this field indicates that the printer does not know how many outputs are supported.

5.2.2.20 Number of options supported

This unsigned byte contains the number, or count, of options currently installed in the printer. An option may be an optional font, memory card, mass storage device, etc. This number may change as options are added and removed from the printer. The host can inquire about the details of each option by using the Request Options Characteristics subcommand of the RDC command.

A nonzero value for this parameter indicates that the printer shall support the Request Options Characteristics subcommand of the RDC command.

A value of 255 (hex FF) for this field indicates that the printer does not know how many options are supported.

5.2.2.21 Printer language and character encoding

This unsigned byte defines the language and encoding that will be used for all messages sent to the host by the printer. Table 16 defines the values associated with the various languages used.

Table 16— RDC-RS printer language and character encoding codes

Value	Description	Character Set
00	English	ISO 8859-1 (Latin 1)
01	German	ISO 8859-1 (Latin 1)
02	French	ISO 8859-1 (Latin 1)
03	Canadian French	ISO 8859-1 (Latin 1)
04	Spanish	ISO 8859-1 (Latin 1)
05	Italian	ISO 8859-1 (Latin 1)
06	Danish	ISO 8859-1 (Latin 1)
07	Norwegian	ISO 8859-1 (Latin 1)
08	Dutch	ISO 8859-1 (Latin 1)
09	Japanese	JIS X0208;1990 (16 bit) Kanji, JIS X0201;1976 (7&8 bit) Katakana
0A	Korean	KS C 5601;1987 (16 bit)
0B	Traditional Chinese	BIG-5 (16 bit) Taiwan Industry Standard
0C	Simplified Chinese	GB2312;1980 (16 bit)
0D	Swedish	ISO 8859-1 (Latin 1)
0E	Japanese	Shift-JIS
0F to FD	Reserved	
FE	Other	
FF	Unknown	

5.2.2.22 Length of product name

This is an unsigned byte that defines the length of the following ASCII string that specifies the Product Name.

5.2.2.23 Product name

This is an ASCII string that defines the model and manufacturer of the printer. This is a human-readable string with the intention that an operator can understand the field without translation from the host. It is recommended that the field not include commas because within some operating systems commas are used as delimiters. This will allow a self-configuring driver to use this printer descriptive string in its user dialog and store it.

The purpose of this field is to indicate printer capabilities as opposed to selection of a printer. This field shall not be used to enable drivers to selectively support certain printers. Therefore, a driver shall not assume that if the printer answers that it is a “XYZ printer,” that all “XYZ printer” features are supported. Therefore, driver software shall not infer the command set based upon this one inquiry from the host.

The suggested format is

Manufacturer’s name (as on the logo):Product Name:Model Number

For example:

ABC Printer Company:XYZ Printer:12345

5.2.2.24 Length of product revision

This is an unsigned byte that specifies the length of the following ASCII string that contains the product revision.

5.2.2.25 Product revision

This is an ASCII string that is defined by the manufacturer of the printer to provide the revision level of the printer and whatever other data the manufacturer deems important. This field may or may not be human readable depending on manufacturer of the printer.

5.2.2.26 Length of serial number

This is an unsigned byte that specifies the length of the following ASCII string that contains the serial number.

5.2.2.27 Serial number

This is an ASCII string that can be used by the printer manufacturer to return the serial number of the printer.

5.2.2.28 Maximum receive command packet size

This unsigned word defines the maximum size (in bytes) of a command packet that can be received by the printer. If the printer receives a command packet that is larger than the value specified, the printer shall reject the packet. A value less than hex 40 shall not be used.

5.2.3 Request Input Characteristics

The RDC Request Input Characteristics subcommand (RDC-RIC) provides the host system with the details of all print media input sources of the printer. The host system can request detailed information concerning each input individually or collectively.

The host requests information for all inputs or a particular input using the byte in the data field immediately following the subcommand. The format of the RDC-RIC command is shown in Table 17.

Table 17— RDC-RIC command

Bytes	Value (hex)	Description	Notes
5		Request Device Characteristics	Packet header
1	02	Subcommand: Request Input Characteristic	Data field
1	ubyte	Input ID 00 = Return Data About all Available Inputs xx = Return Data About a specific Input	

The identification numbers for the various input devices of the printer are assigned by the printer. The host system can obtain all the identification numbers initially by requesting data for all input features.

Table 18 shows the format of the data packet returned by the printer in response to the RDC-RIC command.

Table 18— RDC-RIC response

Bytes	Value (Hex)	Description	Notes
1		Request device characteristic	Packet header
1	02	Subcommand:Request input characteristic	Data field
1	ubyte	Number of inputs to follow	Input #1
1	ubyte	Input identification number	
4	udword	Maximum input capacity—capacity units	
1	bbyte	Input features Byte #1	
1	bbyte	Input features Byte #2	
1	ubyte	Input feed type	
1	ubyte	Medium characteristic	
1	ubyte	Encoded medium size	
2	uword	Minimum size of medium across feed direction in horizontal logical units	
2	uword	Minimum size of medium in feed direction in vertical logical units	
2	uword	Maximum size of medium across feed direction in horizontal logical units	
2	uword	Maximum size of medium in feed direction in vertical logical units	
2	uword	Current size of medium across feed direction in horizontal logical units	
2	uword	Current size of medium in feed direction in vertical logical units	
2	uword	Top margin in vertical logical units	
2	uword	Extent of printable area in feed direction in vertical logical units	
2	uword	Left margin in horizontal logical units	
2	uword	Extent of printable area across feed direction in horizontal logical units	
1	ubyte	Length of input description (not including this byte)	
n	ASCII	Input description string	
1	ubyte	Length of medium description (not including this byte)	
n	ASCII	Medium description string	
...	...	...	Input n

5.2.3.1 Number of inputs to follow

This unsigned byte defines how many input descriptions that the printer is returning in this message. If the host requested the input characteristics for a specific input, then this number will be a value of one. If data for all inputs is requested, then this number will be the total number of inputs currently installed on the printer.

5.2.3.2 Input identification number

This unsigned byte is the logical identification number that the printer has assigned to a particular input. The identification number for a particular input is assigned by the printer and printer manufacturer. The identification numbers for functionally equivalent input devices may be different from printer to printer, depending on the manufacturer. For example, a manual input may have an identification number of 1 for one manufacturer and an identification number of 3 for another manufacturer or different printer model from the same manufacturer.

The printer shall assign identification numbers sequentially each time the printer is initialized or powered on. This means that if the number of input options is changed, a particular input may not have the same identification number as before the change in printer configuration. However, if the printer's configuration has not changed, the assigned identification number shall be the same following each initialization.

The number of input devices that the printer returns in the RDC-RS response indicates the total number of input devices installed on the printer at that time and included in the response to an RDC-RIC.

5.2.3.3 Maximum input capacity—capacity units

This unsigned double word specifies the maximum capacity of this particular input feature. Units for this value are defined by the Counter units parameter, which is returned in the RDC-RS response. Hex FFFFFFFF indicates that the printer is unable to provide this value.

5.2.3.4 Input features byte #1 and #2

These bit-encoded bytes enumerate features associated with a particular input source. The bit assignments for the defined features are shown in Table 19 and Table 20.

Table 19— RDC-RIC Input Features Byte #1 encoding

Bit	Description	Notes
0	Security available zero=No, one = Yes	
1–7	Reserved; Must be zero	

Table 20— RDC-RIC Input Features Byte #2 encoding

Bit	Description	Notes
0–7	Reserved; must be zero	

5.2.3.5 Input feed type

This unsigned byte specifies the general feed type for the input feature. The values that corresponds to various input feed types are shown in Table 21.

Table 21— RDC-RIC input feed type codes

Value	Description	Notes
00	Reserved	
01	Sheet feed auto	
02	Sheet feed manual	
03	Continuous roll	
04	Continuous fan fold	
05 to FD	Reserved	
FE	Other	
FF	Unknown	

5.2.3.6 Medium characteristic

This unsigned byte specifies the general characteristic of the medium that is currently installed in the input feature. Table 22 specifies the defined values and their meaning.

Table 22— RDC-RIC input medium codes

Value	Description	Notes
00	Reserved	
01	Paper	
02	Envelope	
03	Transparency	
04 to FD	Reserved	
FE	Other	
FF	Unknown	

5.2.3.7 Encoded medium size

This unsigned byte specifies the size of the medium that is installed in a particular input source. The size of the medium is an encoded value used to specify a standard ISO size. The values and the corresponding ISO medium sizes are a subset of ISO/IEC 10175 [B4] and are shown in Table . If the size of the medium is unknown, the printer will return the hex value FF.

Table 23— RDC-RIC input size codes

Value (hex)	DescriptionMillimetersApprox. inches							Notes
00	Reserved							
01	A	x			8 1/2	x	11	US sizes
02	B	x			11	x	17	
03	C	x			17	x	22	
04	D	x			22	x	34	
05	E	x			34	x	44	
06	F	x			44	x	68	
07	Legal	x			8 1/2	x	14	
08	1/2 Letter	x			8 1/2	x	5 1/2	
09	Executive	x			7 1/4	x	10 1/2	
10	4A0	1682	x	2378	66 1/4	x	93 3/8	ISO A series
11	2A0	1189	x	1682	46 3/4	x	66 1/4	
12	A0	841	x	1189	33 1/8	x	46 3/4	
13	A1	594	x	841	23 3/8	x	33 1/8	
14	A2	420	x	594	16 1/2	x	23 3/8	
15	A3	297	x	420	11 3/4	x	16 1/2	
16	A4	210	x	297	8 1/4	x	11 3/4	
17	A5	148	x	210	5 7/8	x	8 1/4	
18	A6	105	x	148	4 1/8	x	5 7/8	
19	A7	74	x	105	2 7/8	x	4 1/8	
1A	A8	52	x	74	2	x	2 7/8	
1B	A9	37	x	52	1 1/2	x	2	
1C	A10	26	x	37	1	1x	1 1/2	
1D to 1F	Reserved							
20	RA0	860	x	1220	33 7/8	x	48 1/8	
21	RA1	610	x	860	24 1/8	x	33 7/8	
22	RA2	430	x	610	17	x	24 1/8	
23	SRA0	900	x	1280	35 1/2	x	50 3/8	
24	SRA1	640	x	900	25 1/4	x	35 1/2	
25	SRA2	450	x	640	17 7/8	x	25 1/4	
26 to 2F	Reserved							

Table 23— RDC-RIC input size codes (Continued)

Value (hex)	DescriptionMillimetersApprox. inches							Notes
30	B0	1000	x	1414	39 3/8	x	55 5/8	ISO B series
31	B1	707	x	1000	27 7/8	x	39 3/8	
32	B2	500	x	707	19 5/8	x	27 7/8	
33	B3	353	x	500	12 7/8	x	19 5/8	
34	B4	250	x	353	9 7/8	x	12 7/8	
35	B5	176	x	250	7	x	9 7/8	
36	B6	125	x	176	5	x	7	
37	B7	88	x	125	3 1/2	x	5	
38	B8	62	x	88	2 1/2	x	3 1/2	
39	B9	44	x	62	1 3/4	x	2 1/2	
3A	B10	31	x	44	1 1/4	x	1 3/4	ISO B series
3B to 3F	Reserved							
40	C0	917	x	1297	36 1/8	x	51	ISO C series
41	C1	648	x	917	25 1/2	x	36 1/8	
42	C2	458	x	648	18	x	25 1/2	
43	C3	324	x	458	12 3/4	x	18	
44	C4	229	x	324	9	x	12 3/4	
45	C5	162	x	229	6 3/8	x	9	
46	C6	114	x	162	4 1/2	x	6 3/8	
47	C7	81	x	114	3 1/4	x	4 1/2	
48	C8	57	x	81	2 1/4	x	3 1/4	
49 to 4F	Reserved							
50	Envelope C3	324	x	458		x		ISO envelopes
51	Envelope B4	250	x	353		x		
52	Envelope C4	229	x	324		x		
53	Envelope B5	176	x	250		x		
54	Envelope C5	162	x	229		x		
55	Envelope B6/C4	125	x	324		x		
56	Envelope B6	125	x	176		x		
57	Envelope C6	114	x	162		x		
58	Envelope DL	110	x	220		x		
59	Envelope C7/6	81	x	162		x		
5A	Envelope C7	81	x	114		x		
5B to 5F	Reserved							

Table 23— RDC-RIC input size codes (Continued)

Value (hex)	DescriptionMillimetersApprox. inches							Notes
60	Post card	100	x	148				Japanese
61 to 6F	Reserved							
70	B0	1030	x	1456	40.55	x	57.32	JIS B series
71	B1	728	x	1030	28.66	x	40.55	
72	B2	515	x	728	20.28	x	28.66	
73	B3	364	x	515	14.33	x	20.28	
74	B4	257	x	364	10.12	x	14.33	
75	B5	182	x	257	7.17	x	10.12	
76	B6	128	x	182	5.04	x	7.17	JIS B series
77	B7	91	x	128	3.58	x	5.04	
78	B8	64	x	91	2.52	x	3.58	
79	B9	45	x	64	1.77	x	2.52	
7A	B10	32	x	45	1.26	x	1.77	
7B to 7F	Reserved							
80	Envelope 7-3/4				3 7/8	x	7 1/2	U.S. envelopes
81	Envelope 9				3 7/8	x	8 7/8	
82	Envelope 10				4 1/4	x	9 1/2	
83 to FD	Reserved							
FE	Other							
FF	Unknown							

5.2.3.8 Minimum size of medium across feed direction

This unsigned word specifies the minimum size of the print medium measured orthogonal to (across) the feed direction of the medium for a particular input feature. Units for this value are specified by the printer in Horizontal Logical Units. This unit value is returned in the RDC-RS response. A Horizontal Logical Unit is the unit of measure used to define the Horizontal Print Resolution also returned in the RDC-RS response. (For example, a Horizontal Logical Unit could be dots or pels for a laser or ink jet printer and might be expressed as characters for a fully formed character printer such as a daisy wheel printer.) If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.9 Minimum size of medium in feed direction

This unsigned word specifies the minimum size of the print medium measured in the feed or motion direction of the medium for a particular input feature. Units for this value are specified by the printer in Vertical Logical Units. This unit value is returned in the RDC-RS response. A Vertical Logical Unit is the unit of measure used to define the Vertical Print Resolution also returned in the RDC-RS response. (For example, a Vertical Logical Unit could be dots or pels for a laser or ink jet printer and might be expressed as lines for a fully formed character printer such as a daisy wheel printer.) If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.10 Maximum size of medium across feed direction

This unsigned word specifies the maximum size of the print medium measured orthogonal to (across) the feed direction of the medium for a particular input feature. Units for this value are specified by the printer in Horizontal Logical Units. If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.11 Maximum size of medium in feed direction

This unsigned word specifies the maximum size of the print medium measured in the feed or motion direction of the medium for a particular input feature. Units for this value are specified by the printer in Vertical Logical Units. If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.12 Current size of medium across feed direction

This unsigned word specifies the current size of the print medium measured orthogonal to (across) the feed direction of the medium for a particular input feature. Units for this value are specified by the printer in Horizontal Logical Units. If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.13 Current size of medium in feed direction

This unsigned word specifies the current size of the print medium measured in the feed or motion direction of the medium for a particular input feature. Units for this value are specified by the printer in Vertical Logical Units. If the size is unknown, the printer returns a hex value of FFFF.

5.2.3.14 Top margin

This unsigned word defines the top unprintable area for this input as shown in Figure 4. The units for this value are specified by the printer in Vertical Logical Units.

A hex value of FFFF indicates that the printer does not have or cannot determine this value.

Note that this value specifies the limitation of the printing mechanism and that particular interpreters may place additional limitations upon the margins. The margins supported by the various interpreters can be obtained by using the Request Interpreter Characteristic command.

5.2.3.15 Extent of printable area in the feed direction

This unsigned word defines the printable area in the feed direction for this input as shown in Figure 4. The units for this value are specified by the printer in Vertical Logical Units.

A hex value of FFFF indicates that the printer does not have or cannot determine this value.

Note that this value specifies the limitation of the printing mechanism and that particular page interpreters may place additional limitations upon the margins. The margins supported by the various interpreters can be obtained by using the Request Interpreter Characteristic command.

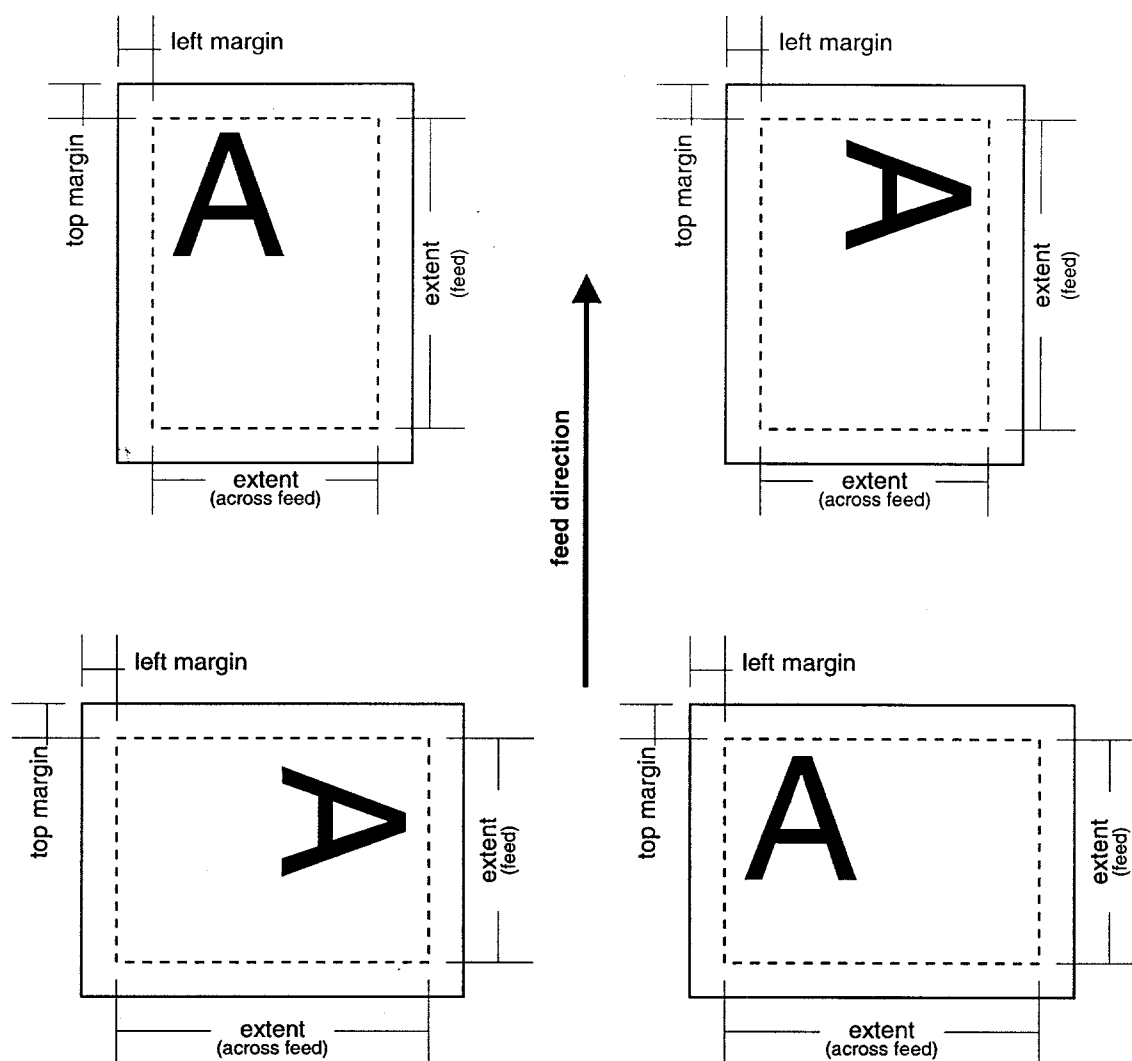


Figure 4— Media margins and printable areas

5.2.3.16 Left margin

This unsigned word defines the left unprintable area for this input as shown in Figure 4. The units for this value are specified by the printer in Horizontal Logical Units.

A hex value of FFFF indicates that the printer does not have or cannot determine this value.

Note that this value specifies the limitation of the printing mechanism and that particular page interpreters may place additional limitations upon the margins. The margins supported by the various interpreters can be obtained by using the Request Interpreter Characteristic command.

5.2.3.17 Extent of the printable area across feed direction

This unsigned word defines the printable area across the feed direction for this input as shown in Figure 4. The units for this value are specified by the printer in Horizontal Logical Units.

A hex value of FFFF indicates that the printer does not have or cannot determine this value.

Note that this value specifies the limitation of the printing mechanism and that particular page interpreters may place additional limitations upon the margins. The margins supported by the various interpreters can be obtained by using the Request Interpreter Characteristic command.

5.2.3.18 Length of input description

This unsigned byte defines the length of the following Input Description ASCII string. If the value of this byte is zero, then there will be no Input Description String.

5.2.3.19 Input description string

This is an ASCII string that can be used as a description of input feature (device) on the printer. The content of this field is printer-implementation dependent. This field is intended to provide a means of providing a human-readable description of the input feature. An example would be “Automatic Envelope Feeder” or “Standard Input Tray.”

5.2.3.20 Length of medium description

This unsigned byte defines the length of the following Medium Description ASCII string. If the value of this byte is zero, then there will be no Medium Description String.

5.2.3.21 Medium description string

This is an ASCII string that can be used as a description of input medium. The content of this field is printer-implementation dependent. This field is intended to provide a means of providing a human-readable description of the input feature. An example of the use of this description would be to describe the current medium that an operator has installed in the input, such as “IRS 1040—Long.”

5.2.4 Request Output Characteristic

The RDC Request Output Characteristic subcommand (RDC-ROC) provides the host system with the details of all print media output features of the printer. The host system can request detailed information concerning each output individually or collectively.

The host requests information for all outputs or a particular output using the byte in the data field immediately following the subcommand. The format of the RDC-ROC command is shown in Table 24.

Table 24— RDC-ROC command

Bytes	Value (hex)	Description	Notes
5		Request device characteristic	Packet header
1	03	Subcommand: Request output characteristic	Data field
1	ubyte	Output ID 00 = Return data about all available outputs xx = Return data about a specific output	

The identification numbers for the various output devices of the printer are assigned by the printer. The host system can obtain all the identification numbers initially by requesting data for all output features.

Table 25 shows the format of the data packet returned by the printer in response to the RDC-ROC command.

Table 25— RDC-ROC response

Bytes	Value (hex)	Description	Notes
5		Request device characteristic	Packet header
1	03	Subcommand: Request output characteristic	Data field
1	ubyte	Number of output features to follow	
1	ubyte	Output identification number	
1	ubyte	Number of output positions	
4	udword	Maximum capacity per output position	
1	bbyte	Standard output features supported Byte 1	
1	bbyte	Standard output features supported Byte 2	
1	ubyte	Length of output description	
n	ASCII	Output description string	
...	...	...	Output n

5.2.4.1 Number of output features to follow

This unsigned byte defines how many output descriptions that the printer is returning in this message. If the host requested the output characteristics for a specific output position, then this number will be one. If data for all output positions is requested, then this number will be the total number of outputs currently installed on the printer.

5.2.4.2 Output identification number

This unsigned byte is the logical identification number that the printer has assigned to a particular output. The identification number for a particular output is assigned by the printer and printer manufacturer. The identification numbers for functionally equivalent output devices may be different from printer to printer, depending on the manufacturer. For example, an output collator may have an identification number of 1 for one manufacturer and an identification number of 4 for another manufacturer or different printer model from the same manufacturer.

The printer shall assign identification numbers sequentially each time the printer is initialized or powered on. This means that if the number of output options is changed, a particular output may not have the same identification number as before the change in printer configuration. However, if the printer's configuration has not changed, the assigned identification number shall be the same following each initialization.

The number of output devices that the printer returns in the RDC-RS response indicates the total number of output devices installed on the printer at that time and included in the response to an RDC-ROC.

5.2.4.3 Number of output positions

This unsigned byte specifies the number of output positions for this particular output device. A ten-bin collator is an example of a multiple position output device and would have a hex value of A in this field.

5.2.4.4 Maximum capacity per output position

This unsigned double word specifies the maximum capacity for each output position of this particular output feature. Units for this value are defined by the Counter units parameter that is returned in the RDC-RS response. The hex value FFFFFFFF indicates that the printer is unable to provide this value.

5.2.4.5 Standard output features supported Bytes 1 and 2

These bit-encoded bytes define the availability of certain standard functions or features on the output device. Table 26 and Table 27 define the meaning of the individual bits in these bytes.