GA27-3043-0

Systems

IBM 3735 Programmable Buffered Terminal Concept and Application

IBM

GA27-3043-0

IBM 3735 Programmable Buffered Terminal Concept and Application

Systems

IBM

Preface

The purpose of this manual is two-fold: First, it introduces the IBM 3735 Programmable Buffered Terminal; and, second, it provides planning information for those who may be involved in installing a 3735. Because of this dual purpose, not all sections of the manual may be of interest to each reader. Those interested in an overview and in the highlights of the 3735 will find the Introduction of most value; those who need more detailed information on the terminal's operation and functional capabilities will find the sections on Operating Characteristics and Implementation Considerations of most interest; systems and application programmers will find information on Programming Considerations for the 3735 in Appendix B.

Generally, the prerequisites for this manual are an understanding of the concept and application of the IBM System/360 and the IBM System/370 in a teleprocessing environment. The section on Programming Considerations requires more extensive knowledge of the applicable operating system (OS or DOS).

Although not intended as a terminal operator's guide or as a programmer's guide, the manual presents general information concerning these subjects.

Abbreviations

ALU	Arithmetic and Logic Unit
bps	bits per second
BSC	Binary Synchronous Communications
BTAM	Basic Telecommunications Access Method
CPU	central processing unit
DLM	delimiter
DOS	Disk Operating System
FDM	form description macro
FDP	Form Description Program
FDU	Form Description Utility
I/O	input/output
IRS	Interrecord Separator
OS	Operating System
RS	Record Separator
TCAM	Telecommunications Access Method
TCP	Terminal Control Program
TRF	Trouble Record Form
UPSI	User Program Switch Indicators

First Edition (January 1971)

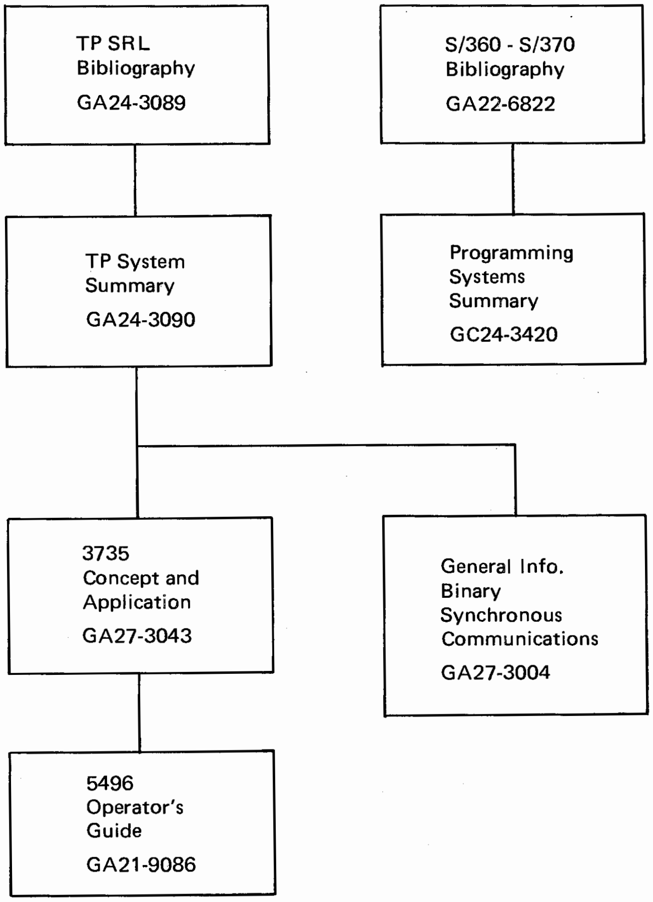
Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems or equipment, refer to the latest SRL Newsletter for the editions that are applicable and current.

Request for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This manual has been prepared by the IBM Systems Development Division, Publications Center, Department E01, P. O. Box 12275, Research Triangle Park, North Carolina 27709. A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be sent to the above address.

IBM 3735 SRL Publications Availability Guide

Use this guide to determine what available publications will best fulfill your individual requirements.


TP SRL Bibliography
GA24-3089
S/360 - S/370 Bibliography
GA22-6822
TP System Summary
GA24-3090
Programming Systems Summary
GC24-3420
3735 Concept and Application
GA27-3043
General Info. Binary Synchronous Communications
GA27-3004
5496 Operator's Guide
GA21-9086

Contents

Introduction 7
Form Description Program Concept 9
Configuration 10
Applications 11
Source Recording 11
Batch Transmission 11
Inquiry 11
Combined Operation 11
Program Support 11
Data Security 12
Problem Determination 12

Operating Characteristics 13
Control Unit 13
Magnetic Disk 13
Dynamic Buffers 13
Arithmetic and Logic Unit 13
Binary Synchronous Communications Adapter 13
Selectric I/O II 13
Printer 13
Keyboard 14
Operating Procedures 14
Local Mode 14
Selecting an FDP 14
Creating a Document 15
Error Correct Mode 15
Playback Mode 15
Request Mode 15
Communicate Mode 15
Erase Data From CPU 15
Erase Operator-Created Data 17
Listing of Resident FDPs 17
Listing of Resident Data Records 17
Diagnostic Tests 17
Special Features 17
Multipoint Communications 17
Synchronous Clock 18
Keylock 18
5496 Adapter 18
Pinfeed Platens 18
Forms Stacker 18
Communications Facilities 18

Implementation Considerations 19
Forms Design 19
Forms Description 19
Form 19
Page 20
Line 20
Field 20
Source 20
Operation 21
Sink 21
Control 22
System Considerations 22
Skipped Fields 22
Form Records 24
Transmittal Blocks 24

Appendix A. Code Charts 27

Appendix B. Programming Considerations 29
Form Description Macros 29
Assembly 29
Utility 29
Transmission 30

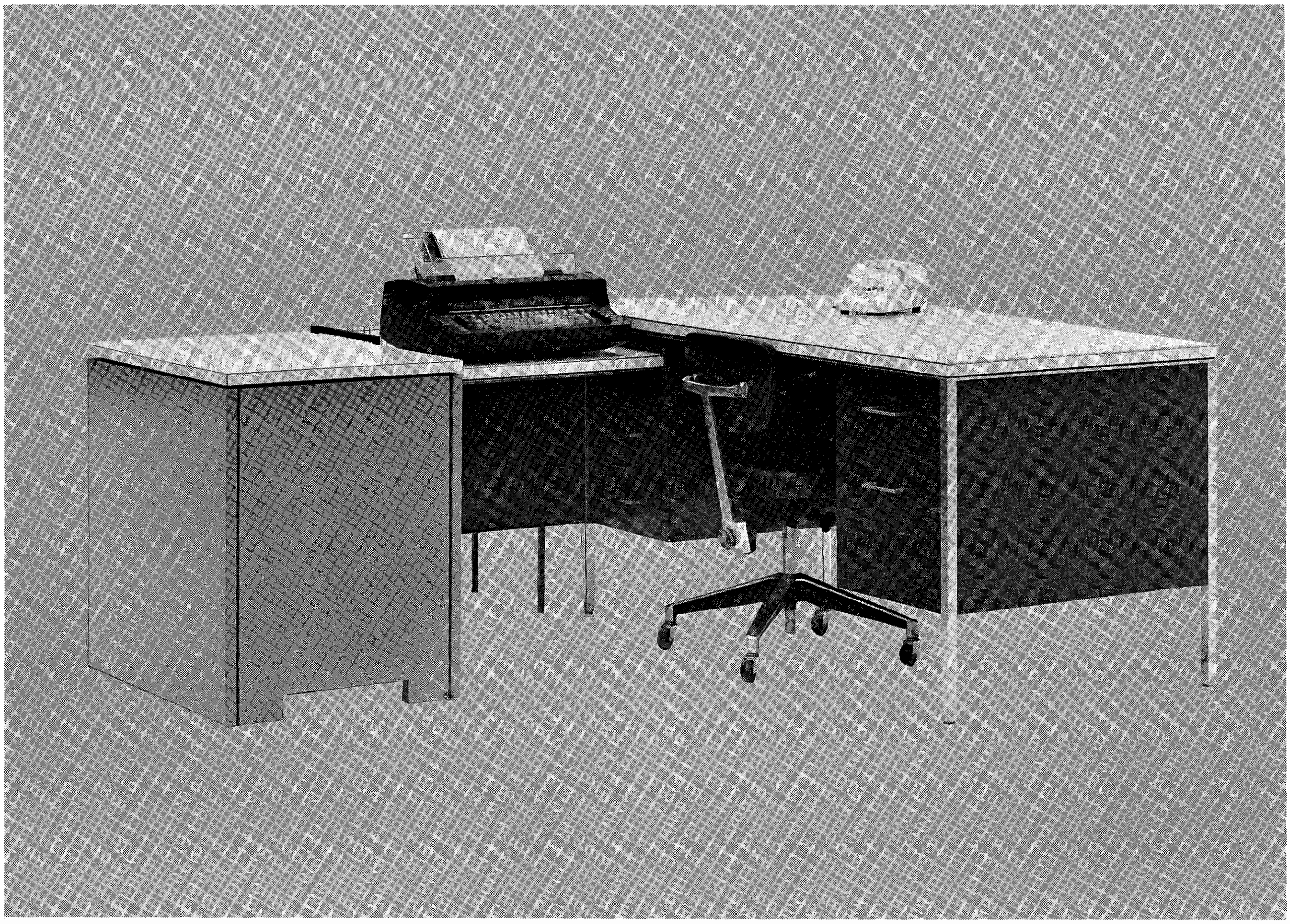
Appendix C. Form Description Example 31
Sample Application 31
Describing the Form 31

Glossary 43

Index 45

Illustrations

Figure	*Title*	*Page*
	IBM 3735 Programmable Buffered Terminal, Typical Configuration	Frontispiece
1	IBM 3735 Operating Environment	7
2	Sample Form	8
3	FDP Generation	9
4	Basic Configuration	10
5	Feature List	10
6	3735/CPU Configurations	10
7	Diagnostic Tests	12
8	Control Unit Data Flow	14
9	Keyboard	15
10	Operating Keys	16
11	Operating Indicators	17
12	Data Source	21
13	Content Checks	21
14	Arithmetic and Compare Operations	22
15	Data Sink	22
16	Edit Functions	23
17	Examples of Zero Suppression	23
18	Examples of Drifting Characters	23
19	Examples of Insertion Characters	24
20	Transmission Formats	25
A-1	EBCDIC	27
A-2	USASCII	28
C-1	Application Example	31
C-2	Sales Invoice Form	33
C-3	Locating Lines and Fields on the Form	34
C-4	Annotated Form and Master Card	35
C-5	Form Description (5 Parts)	36
C-6	Completed Form	41
C-7	Form Record Transmission Format	42



IBM 3735 Programmable Buffered Terminal, Typical Configuration (Telephone, Desk, and Chair Not Furnished by IBM)

Preparing source documents–invoices, waybills, personnel forms, contracts–and capturing data go hand in hand. Most of the data required for day-to-day business operation is generated during preparation of the many forms used in the business. Accounts receivable are accumulated during preparation of order-entry forms; accounts payable are generated during preparation of purchase orders; inventory accounts are described by shipping and receiving invoices. Thus, much of the information required for data processing originates during preparation of these "source documents." Preparing forms and capturing selected data quickly and accurately is essential to efficient operation of the modern data processing system.

Many business firms using centralized data processing have turned to teleprocessing to satisfy these requirements. Interactive terminals bring the power and flexibility of the central computer to the operator of a typewriter-like terminal; thus, the CPU program can automatically provide operator guidance, error detection, formatting, editing, and many other functions. Data is readily captured by programmed selection from the input data. High transmission costs and programming overhead have limited the application of interactive terminals.

Buffered terminals, which allow local data storage for periodic batch transmission to the CPU, have, by more efficient use of the transmission facilities and by reducing CPU program requirements, lowered the cost of centralized processing. But, buffered terminals sacrifice many of the automated procedures for operator guidance and assistance in preparing the source documents.

The IBM 3735 Programmable Buffered Terminal combines the power and flexibility of the interactive terminal with the efficiency of batch transmission to provide a new dimension in source-document preparation and data capture. The 3735 is a programmed terminal consisting of the popular IBM Selectric® keyboard printer and a highly sophisticated desk-side control unit. Designed primarily for those applications using preprinted (fixed-format) forms and batch processing, the 3735 can, via its own internal program, be tailored to fit the requirements of a wide variety of data processing environments.

The 3735 stores, under program control, information generated during document preparation for later transmission to the CPU. Typically, a full day's operator output can be stored for unattended transmission to the CPU, and the CPU can return processed data for use in the next day's operation (Figure 1).

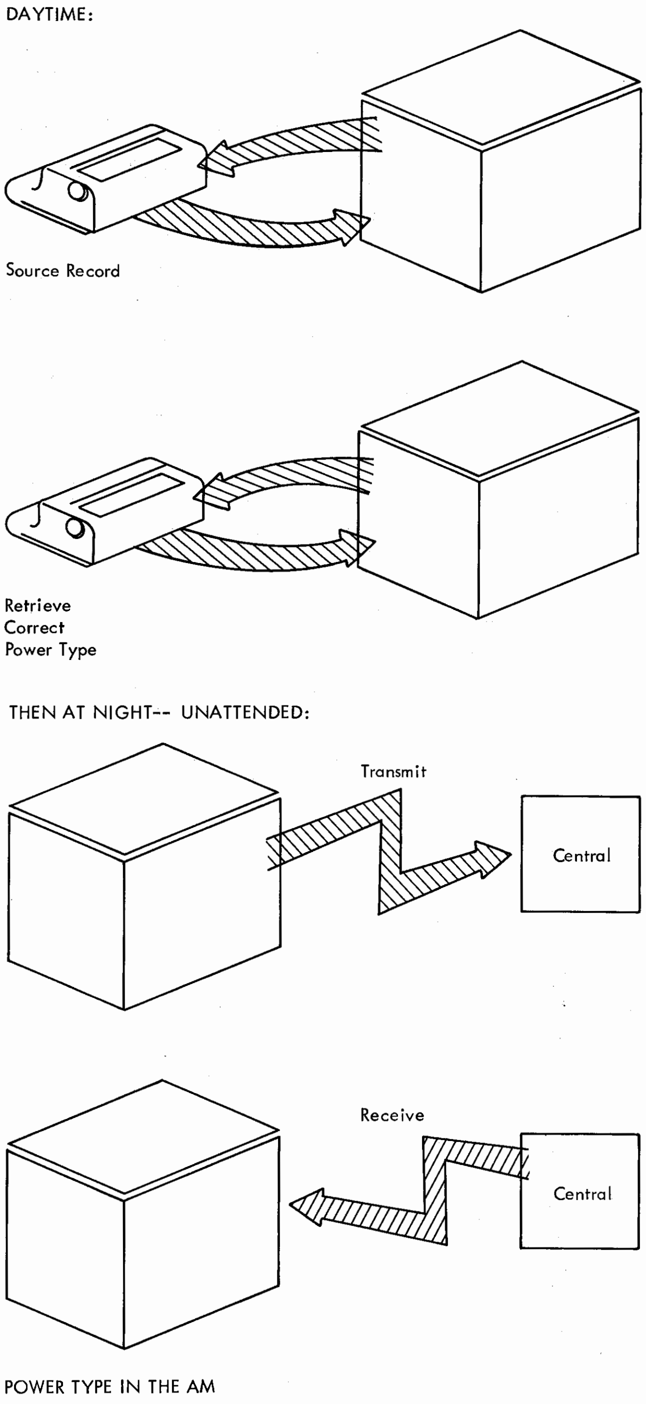
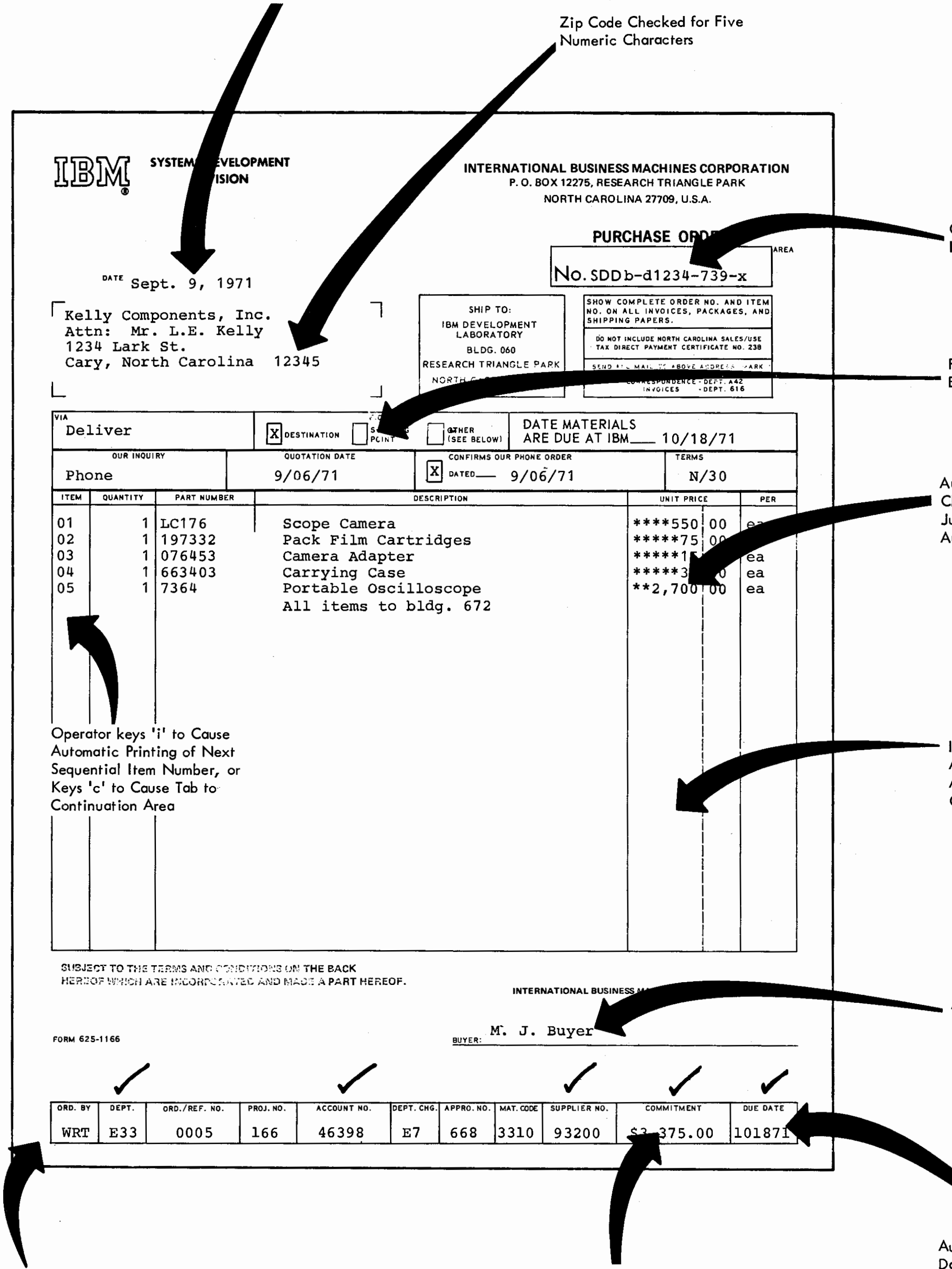


Figure 1. IBM 3735 Operating Environment

FORM/PRINT-ELEMENT AUTOMATICALLY POSITIONED FOR ENTRY OF EACH FIELD



Date Automatically Emitted from Special Storage Area

Zip Code Checked for Five Numeric Characters

IBM® SYSTEM EVELOPMENT ISION

INTERNATIONAL BUSINESS MACHINES CORPORATION
P. O. BOX 12275, RESEARCH TRIANGLE PARK
NORTH CAROLINA 27709, U.S.A.

PURCHASE OR AREA

No. SDDb-d1234-739-x

DATE Sept. 9, 1971

Kelly Components, Inc.
Attn: Mr. L.E. Kelly
1234 Lark St.
Cary, North Carolina 12345

SHIP TO:
IBM DEVELOPMENT LABORATORY
BLDG. 060
RESEARCH TRIANGLE PARK
NORTH

SHOW COMPLETE ORDER NO. AND ITEM NO. ON ALL INVOICES, PACKAGES, AND SHIPPING PAPERS.

DO NOT INCLUDE NORTH CAROLINA SALES/USE TAX DIRECT PAYMENT CERTIFICATE NO. 238

SEND ALL MAIL TO ABOVE ADDRESS ARK
CORRESPONDENCE - DEPT. A42
INVOICES - DEPT. 616

Order Number Automatically Incremented and Printed

Fields Automatically Skipped Because 'Destination' Indicated

VIA Deliver	☒ DESTINATION	□ S POINT	□ OTHER (SEE BELOW)	DATE MATERIALS ARE DUE AT IBM___ 10/18/71

OUR INQUIRY	QUOTATION DATE	CONFIRMS OUR PHONE ORDER	TERMS
Phone	9/06/71	☒ DATED___ 9/06/71	N/30

ITEM	QUANTITY	PART NUMBER	DESCRIPTION	UNIT PRICE		PER
01	1	LC176	Scope Camera	****550	00	e
02	1	197332	Pack Film Cartridges	*****75	0	
03	1	076453	Camera Adapter	*****1		ea
04	1	663403	Carrying Case	*****3	0	ea
05	1	7364	Portable Oscilloscope	**2,700	00	ea
			All items to bldg. 672			

Automatic Fill of Unused Positions; Checked for Numeric Only; Right Justified with Space for Decimal; Automatic Comma Insertion.

Operator keys 'i' to Cause Automatic Printing of Next Sequential Item Number, or Keys 'c' to Cause Tab to Continuation Area

If Line Entries Exceed the 22 Allowed on this Form an Automatic Skip to the Footing Occurs

SUBJECT TO THE TERMS AND CONDITIONS ON THE BACK HEREOF WHICH ARE INCORPORATED AND MADE A PART HEREOF.

INTERNATIONAL BUSINESS

FORM 625-1166

BUYER: M. J. Buyer

Automatic Print from Table in Special Storage

	✓			✓				✓	✓	✓
ORD. BY	DEPT.	ORD./REF. NO.	PROJ. NO.	ACCOUNT NO.	DEPT. CHG.	APPRO. NO.	MAT. CODE	SUPPLIER NO.	COMMITMENT	DUE DATE
WRT	E33	0005	166	46398	E7	668	3310	93200	$3 375.00	101871

Fields Checked for Mandatory Entry of Alphabetic or Numeric Characters as Specified; Entries Automatically Centered Within Field.

Total Calculated and Checked Against Maximum Allowable Amount; Right Justified with Decimal and Comma Insertion; Batch Totals Accumulated as Required.

Automatic Duplication of Due Date from Above

AUTOMATIC SKIP TO NEXT FORM (CONTINOUS FORMS) AFTER LAST FIELD COMPLETED

✓ ITEMS MARKED FOR TRANSMISSION TO CPU

Figure 2. Sample Form

During document preparation, the 3735 can provide:

- Operator guidance: setup instructions, exception messages, indication of keying or procedural errors.
- Programmed forms control: automatically positions the form for printing each line.
- Automatic print-element positioning: automatically positions data within the predefined fields.
- Format and edit operations: center, left or right justify, underline, character fill, decimal-comma insertion.
- Logical decisions: conditional field skipping and entry requirements.
- Arithmetic operations: add, subtract, multiply, divide, and divide and round.
- Power typing: automatic printing of information previously entered or internally generated.

Figure 2 shows a sample form and indicates several functions that can be implemented within the terminal program.

These functions coupled with efficient forms design can:

$ Reduce source-document preparation costs.
$ Reduce rework expense.
$ Reduce errors in the data base.
$ Reduce line costs.

FORM DESCRIPTION PROGRAM (FDP) CONCEPT

Two levels of program control are used on the 3735. Form Description Programs (FDPs), generated by the customer, specify the functions desired for processing a specific form type. A Terminal Control Program (TCP), recorded in the control unit during manufacture, interprets the FDPs and provides detailed terminal control. This two-level approach relieves the customer of much of the detailed programming required to provide the many functions available on the 3735; a minimum FDP may exercise thousands of TCP instructions.

A unique set of macro instructions used for generation of the FDPs further simplifies terminal installation. Forms are described using macro statements assembled by the CPU's operating system, formatted by a utility program for later transmission, and transmitted to the 3735 via the customer's application program and teleprocessing access method (Figure 3). The FDPs reside in the terminal until replaced by the CPU.

This method of FDP generation has several advantages:

- Minimum CPU programming is required to generate the terminal FDPs.
- Terminal programs can be changed easily to keep up with forms changes, changing data requirements, validity checking rules, and changing procedures.
- Programs can be varied from terminal to terminal without creating an excessive programming burden.

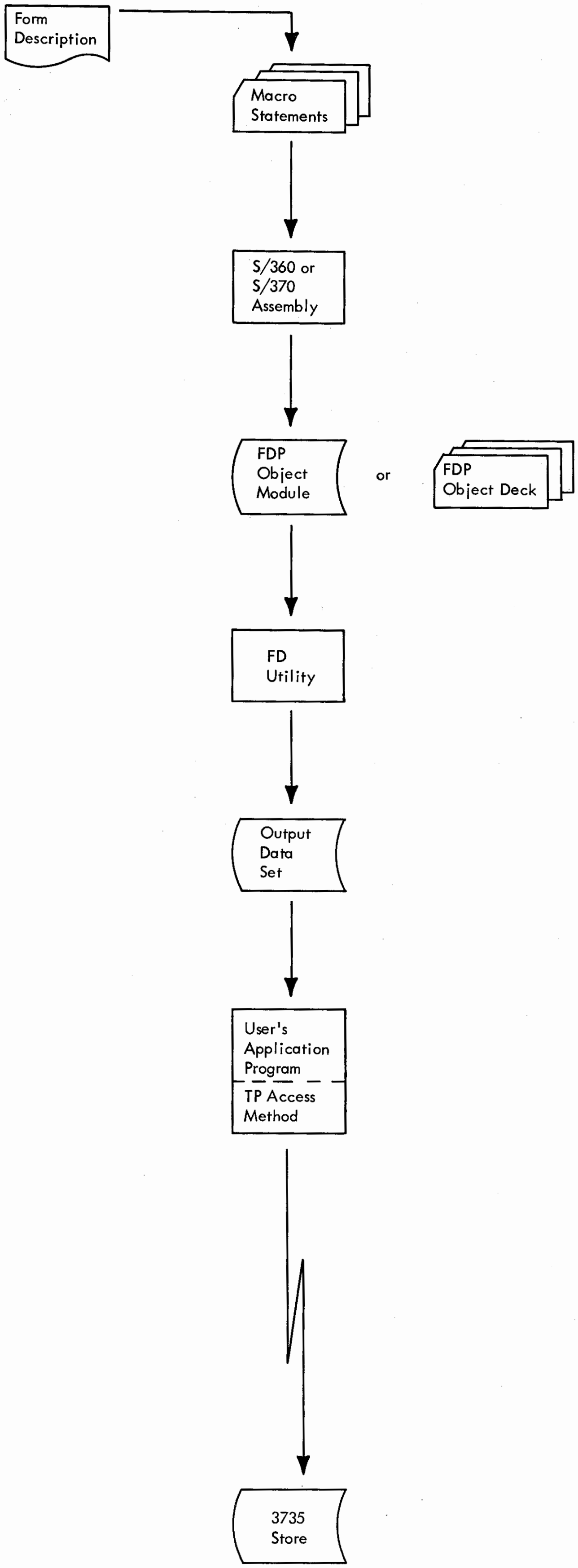


Figure 3. FDP Generation

- The staff at central has complete control over the unique program complement of each terminal, and can schedule the introduction of new record formats to coincide with new CPU programs to handle them.

CONFIGURATION

The basic configuration of the 3735 consists of an IBM Selectric I/O II keyboard printer and a desk-side control unit (Figure 4). A magnetic disk storage device within the control unit contains the Terminal Control Program, the FDPs, and the user data storage. The number of Form Description Programs, each representing a different form type, that can be stored depends on the length of the programs; however, a typical FDP requires about 1000 bytes of terminal storage. Basic storage capacity for customer FDPs and data storage is about 62K bytes. This basic capacity is expandable in increments of 42K bytes to a total capacity of approximately 145K bytes.

The 3735 uses the binary synchronous method of communications line control; thus, it is compatible with most systems and programs using this method of line control.

The Arithmetic Logic Unit (ALU) controls all data flow to and from the disk storage and the I/O devices.

Figure 5 lists the features of the 3735. These are available in one of three ways:

1. Basic: present on all machines.
2. Specify: available at no cost, on a specify basis.
3. Special: available at additional cost.

Figure 6 shows the System/360 and System/370 configurations possible with the 3735.

The IBM 5496 Data Recorder attachment feature allows attaching the 5496 to the 3735. The 5496 is a buffered, operator-oriented, key-entry unit used to create and read the 96-column punched card.

UNIT	BASIC	SPECIFY	SPECIAL
Control	62K byte data and FDP store 45.2K byte control program store EBCDIC transmission code BSC data-link control VRC on internal transfers LRC on disk data Point-to-Point switched FDP control Auto answer	ASCII transmission code Transmission speed: 600 (World Trade), 1200, 2000, 2400 bps Data set attachment	Additional data/FDP store in increments 41.8K bytes to total capacity of 145K bytes Synchronous clock (600 or 1200 bps) Multipoint on leased lines Keylock 5496 Data Recorder adapter
Keyboard-Printer	Keyboard Interlock Acoustic slot filter Operator guidance indicators Ten-digit numeric pad 15.5 character/second print speed	Keyboard nomenclature and print element for World Trade	Pin-feed platen Forms stacker

Figure 5. Feature List

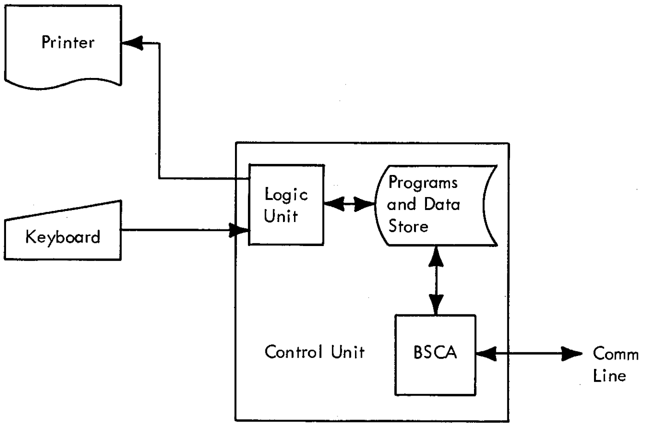
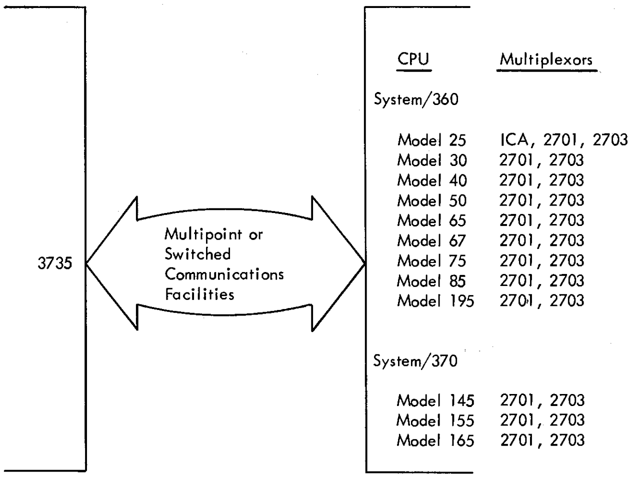


Figure 4. Basic Configuration



CPU	Multiplexors
System/360	
Model 25	ICA, 2701, 2703
Model 30	2701, 2703
Model 40	2701, 2703
Model 50	2701, 2703
Model 65	2701, 2703
Model 67	2701, 2703
Model 75	2701, 2703
Model 85	2701, 2703
Model 195	2701, 2703
System/370	
Model 145	2701, 2703
Model 155	2701, 2703
Model 165	2701, 2703

Figure 6. 3735/CPU Configurations

APPLICATIONS

The 3735 can be used in a number of applications that are common to many industries.

Source Recording

This application uses the keyboard printer to create a document while data is being captured on the disk buffer. Extensive validity checking and intermediate totals ensure that each transaction is complete and, to a large degree, correct. Errors detected at the terminal can be corrected while the source information is still available to the terminal operator. Audit trails via batch totals are a byproduct of terminal operation. Automated procedural and data checking, incorporated within the FDPs, ensure that fewer errors enter the computer system.

Cards with repetitive data—such as stock numbers, descriptions, and unit prices of frequently ordered items—can be kept at the terminal in a tub file and entered via the card reader for faster, more accurate operation. Price extensions can be made automatically following entry of the ordered quantity.

Savings accrue through more accurate and efficient terminal operation, fewer errors entering the system, and a minimum line time and CPU overhead requirement.

The following listing suggests a few specific source-recording applications:

Insurance	Policy Writing Endorsements Claim Notices and Checks
Manufacturing	Contract Writing Remote Plant Accounting Dealer Accounting Financial Reporting
Transportation	Waybills Manifests Freight Accounting
Distribution	Order Entry Billing Inventory Control

Batch Transmission

The large magnetic disk is an ideal transmission buffer. In addition, the logic and formatting capability of the 3735 allow only selected data from each input record to be transmitted. The terminal can develop batch totals for better control of files being transmitted. Records returned to the terminal can be directed to output devices under terminal (FDP) control with data selection and editing as required.

Inquiry

An FDP can provide simple inquiry consisting of a request transaction sent to central, followed by a completely formatted reply transmitted back. Or the FDP can allow keying of abbreviated inquiries with the terminal adding transaction codes and other system information to complete the transaction. The reply can contain just the variable data to be printed out, with appropriate headings and editing added by the FDP. This approach makes the inquiry operation simpler, less prone to error, and less demanding on line time.

Combined Operation

The on-line use of the 3735 for an order-entry application illustrates the advantages of off-line source recording and on-line inquiry.

In this application, the 3735 is programmed for order entry. Input data formats and validity checks are programmed to guide the operator through the order procedure. When an operator enters an item number and quantity, the terminal can automatically send an inquiry to see if sufficient quantity of this item is in stock to meet the order. If it is, the central CPU can return the price to the terminal for a price-times-quantity extension at the terminal. If not, the operator can be notified to select an alternate quantity or item.

The advantage of this approach is that it has most of the functions of on-line order entry without the additional CPU, file, multiplexer, and line costs of the on-line approach. A single leased line with a simple inquiry program to an abbreviated inventory file is all that is required to support multiple terminals. With a minimum commitment, a customer can enter an environment that would have required a major step into on-line processing with a non-programmed terminal.

PROGRAM SUPPORT

System/360 and System/370 support the 3735 under OS and DOS. This support provides for transmitting data between the computer and the terminal, for generating Form Description Programs, and for preparing the FDPs for transmission to the terminal.

Since the 3735 is a binary synchronous terminal, the BSC support existing in OS TCAM and BTAM and in DOS BTAM is sufficient to handle the transmission of data between the computer and the terminal.

The FDP generation support is System/360 and System/370 OS and DOS Assembler Language macro instructions. The form description (FD) macro instructions provide error checking for the user.

A Form Description Utility (FDU) is provided to prepare the FDPs for transmission to the 3735. The FDPs are placed in an output data set for selection by the user's application program. When the terminal receives these programs, they reside on the 3735 disk and the operator uses them to control terminal functions.

The FDU operates under OS and DOS. It operates independently of the user's TP program; however, it is dependent upon the assembly of the FD macro instructions. The utility is scheduled through the job input stream.

The minimum system that can use the form description macros and the Form Description Utility is:

OS–System/360 Model 40; 128K
DOS–System/360 Model 25; 16K

DOS requires 32K bytes of storage to support teleprocessing.

DATA SECURITY

Features that assist in protecting data and preventing unauthorized use of the 3735 are:

Terminal ID: This basic feature provides each 3735 with a unique identifier (ID). The terminal transmits the ID to the CPU before beginning data transmissions. The identification characters are IBM-assigned and are not duplicated between terminals.

CPU Security ID: This is a feature of the TP access method that allows it to send up to 15 CPU identification characters to the terminal. An ID list containing IDs for up to 14 CPUs can be stored in the 3735 to allow it to recognize any of several CPUs.

Keylock: This 3735 special feature provides a keylock which, when locked, disables all operator activity related to input, output, or control of data at the 3735.

In addition, the customer can implement sophisticated data-protection and user-identification techniques within the terminal's program. For example, passwords can be associated with specific data fields, or within inquiry routines, and the operator can be required to enter the password before operation continues.

PROBLEM DETERMINATION

To ensure maximum availability of the 3735, its basic design incorporates several problem-determination aids. Figure 7 lists diagnostic tests that are included within the terminal's control program. In addition to the diagnostic tests, a trouble record form allows the terminal operator to test functions associated with the FDPs.

The trouble record form (TRF) is a fixed-format document that the terminal operator processes in much the same way as a normal working document. Completion of the TRF exercises each of the terminal functions related to processing forms, and assures the operator that the terminal can perform the functions.

TEST NAME	PURPOSE	WHEN USED	RESULTS
Bring-up	Test control unit electronics	Runs automatically whenever terminal power is turned on, or when hardware error is detected.	Test results are indicated in operator guidance lights.
Disk surface	Test all storage areas.	Selected by terminal operator.	Test results print on Selectric I/O.
Bi-sync	Test functions of communications adapter	Selected by terminal operator.	Test results are indicated in operator guidance lights.
Selectric I/O II	Test functions of keyboard/ printer	Selected by terminal operator.	Test results are indicated in operator guidance lights.
Reader/punch	Test functions of reader/punch	Selected by terminal operator.	Test results are indicated in operator guidance lights.
On-line	Test functions associated with communications link.	Selected by terminal operator.	Test results print on Selectric I/O.

Figure 7. Diagnostic Tests

CONTROL UNIT

Except for turning power on, the 3735 control unit requires no operator attention, and familiarity with its internal operation is unnecessary for operation of the 3735; however, some systems requirements can be more readily understood through a knowledge of the concepts used in the control unit's design. Figure 8 shows the internal organization; the following paragraphs describe each of the units shown.

Magnetic Disk

This is the main storage of the 3735. It contains the Terminal Control Program and the customer's data and FDPs. Information is stored in eight-bit bytes (seven data bits and a parity bit), each byte representing one character, organized into 480 byte sectors. In data-storage areas, four bytes of each sector are used for identification information; thus 476 bytes remain for working storage. In FDP storage areas, 234 bytes of each sector store FDPs; the balance of the sector is not used.

Storage Areas

In addition to the Terminal Control Program, the FDPs, and the customer data storage, the magnetic disk provides the following storage areas:

Counters: 21 ten-digit counters. Used for temporary arithmetic storage during calculations.

Special Storage: 236 bytes of byte-addressable storage. Can be structured and used as desired by the customer.

Inquiry Buffer: 236 bytes of byte-addressable storage. Used for the assembly of inquiry messages to be sent to the CPU, and for the receipt of responses from the CPU.

Reader/Punch Buffers: Two 96-byte byte-addressable storage areas. Used to buffer information going to and from the 5496 Data Recorder.

Data Directory: Size-dependent on quantity of FDPs and form records presently stored. Provides the data management function for the 3735; records the identification and storage location of each FDP and each form record presently stored in the terminal.

Dynamic Buffers

Three dynamic buffers buffer data and programs moving to and from the disk. Each buffer stores one sector of information; during program execution, data is collected in a dynamic buffer and is then written in the desired disk location.

The dynamic buffers also store the logic indicators: 105 bits of bit-addressable storage. Indicators are set on or off (bit or no-bit) or inverted, depending on the results of an FDP-specified test. The FDP can then test the indicators to control branching or other functions.

Information received from the communications line enters one of the dynamic buffers and remains there until the complete block is received and checked. If the block is correctly received, the data is stored on the magnetic disk as a sector of information. Because each sector contains 476 bytes of usable storage, message blocks from the communications line must not contain more than 476 characters of data (not including data link control characters).

Arithmetic and Logic Unit

This is the calculator and decision maker of the 3735. The ALU is the tool used by the Terminal Control Program to interpret the customer's FDP, and selects the appropriate routines for performance of the desired functions.

Binary Synchronous Communications Adapter

This adapter, activated by the TCP, transfers information to and from the communications channel. Message exchanges between the 3735 and the CPU conform to the rules of the binary synchronous method of data link control as described in the Systems Reference Library publication, *General Information–Binary Synchronous Communications,* GA27-3004.

SELECTRIC ® I/O II

Printer

The Selectric ® I/O II printer has the desirable features of other Selectrics:

- Single globe-shaped print element.
- Two-stroke storage system that reduces typing errors by storing the second of two rapidly keyed characters until the first is printed.
- No platen carriage, thus eliminating vibration caused by horizontal movement of a heavy platen.
- Dual-velocity printing to improve print quality on such characters as the period and comma.

Character spacing is ten characters per inch; line spacing is six lines per inch. Certain optional 10-pitch print elements may be used; some differences between keytop/print-element nomenclature may result. A single-color fabric ribbon packaged in a convenient cartridge is provided.

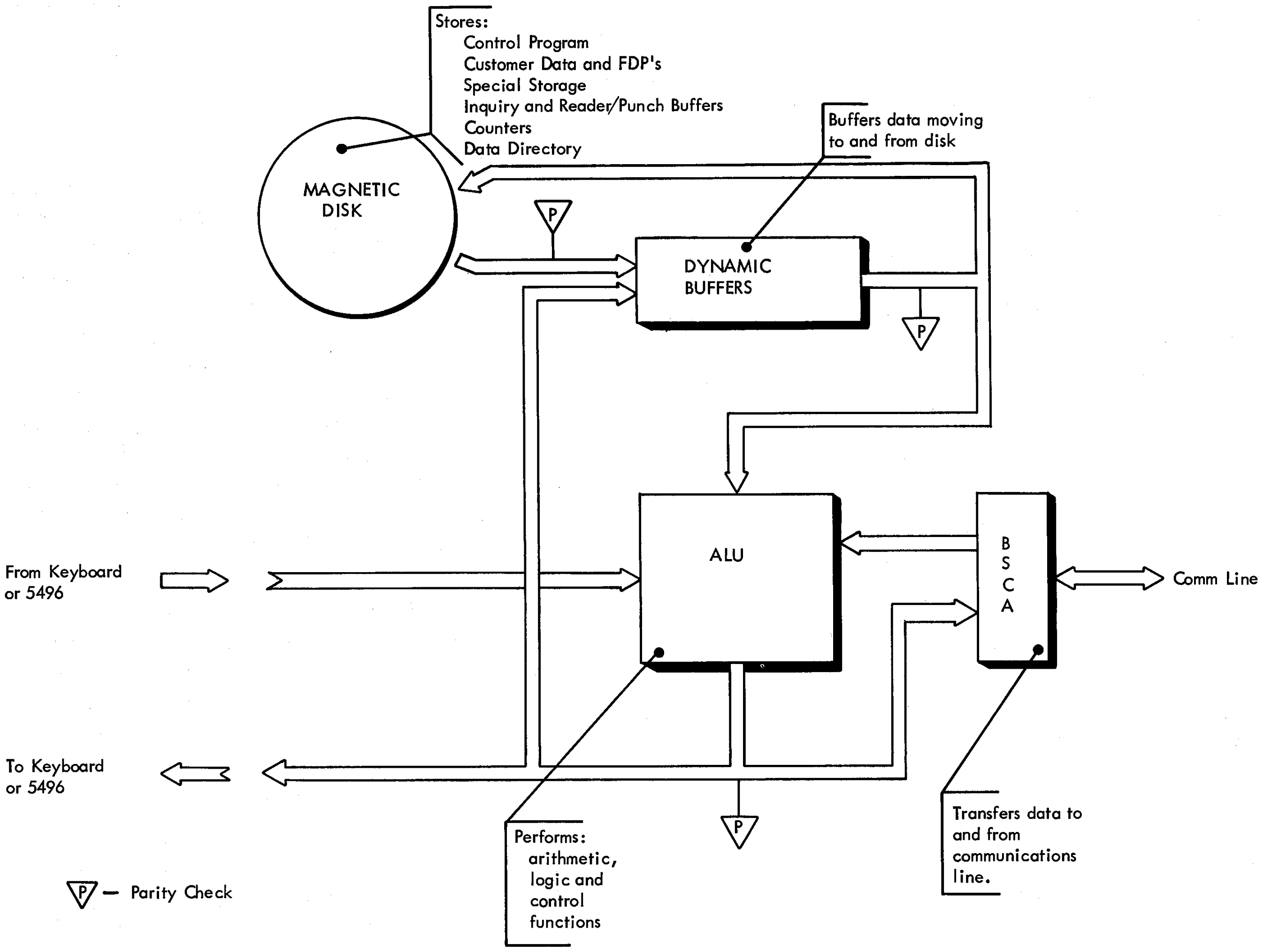


Figure 8. Control Unit Data Flow

Keyboard

The Selectric I/O keyboard (Figure 9) is the operator's interface to the 3735. The keyboard contains all operating controls, and operator guidance indicators, used during normal processing. As shown by the shaded area of Figure 9, the keyboard has a numeric insert in an adding machine format. This set of keys function as numeric-input keys under control of the FDP.

In addition to the keys found on the standard office Selectric typewriter, the Selectric I/O II has the controls needed to interface with the Form Description Programs. Figure 10 lists these keys and their functions; Figure 11 lists the operator guidance indicators.

OPERATING PROCEDURES

Detailed operating procedures will be provided in an operator's guide. The following paragraphs present an overview of typical procedures.

When power is turned on in the 3735, the terminal performs a series of diagnostic tests to validate functional operation. An indicator-light pattern alerts the operator when the tests are complete. To proceed, the operator places the terminal in local mode.

Local Mode

When in local mode, the terminal can be used much as any typewriter. Information keyed is printed without program action; the operator controls terminal functions. In addition to allowing typewriter-like operation, local mode allows selection of other operating modes. A typical operating sequence would be to select an FDP and begin the day's processing.

Selecting an FDP

Its associated identifier (a three-digit number) selects an FDP. After selection, the operator may elect to initiate printout of an operator message associated with the pro-

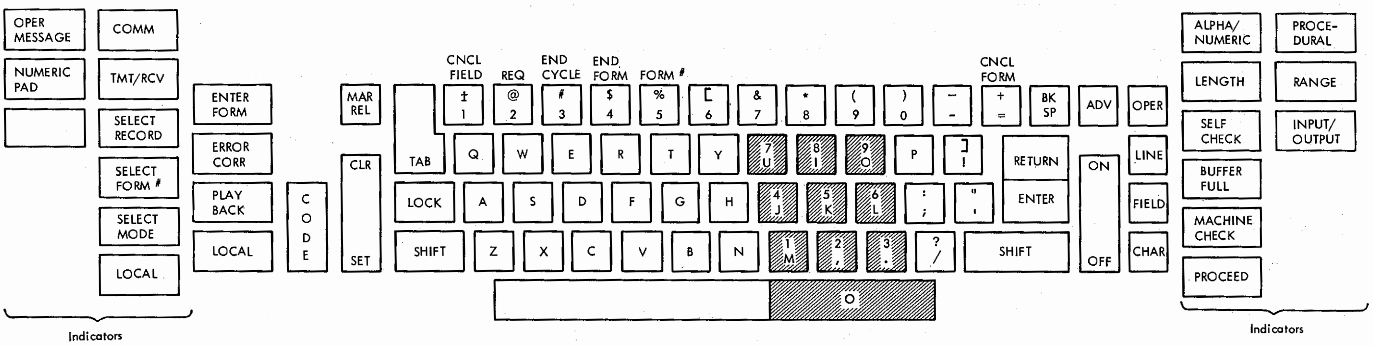


Figure 9. Keyboard

gram, to proceed with a tab-set routine (under direction of the FDP), or to begin processing documents under program control. Normal document creating is done in enter-form mode.

Creating a Document

Placing the terminal in enter-form mode initiates processing under control of the selected FDP. Each field of the form is processed through operator/terminal interaction as specified by the program. When the form has been completed, the terminal automatically prepares to process the next form (fixed-length forms), or the operator signals 'End Form' (variable-length forms). Processing under control of this FDP can then continue on the next form, or another program or mode of operation can be selected.

In case of error during processing of a form, the operator action depends on the type of error and on its location. The operator can correct errors detected in the field being processed by backspacing and retyping. To correct errors detected on the line being processed but in a field preceding the one presently under process, the operator must either enter error-correct mode or complete the document and print it out for correction.

Error-Correct Mode

Placing the terminal in error-correct mode causes the print element and the FDP to return to the beginning of the line being processed. Counters and indicators are set to the value existing when the line was first entered. The operator can then move the print element and program across the line, a field or a character at a time, correcting errors as they are reached. After correcting all errors on the line, the operator places the terminal back in enter-form mode and continues processing the form.

With the terminal in error-correct mode, processing cannot continue beyond the point obtained during the previous enter-form operation. The operator cannot use error-correct mode to correct errors detected in lines preceding the line being processed, but can quickly correct them by placing the terminal in playback mode.

Playback Mode

Playback mode allows previously processed documents to be called out for power typing under FDP control. The information can be corrected or updated, or only printed to obtain a clean copy. Operator control of the typeout is completely flexible; the operator can allow printing of a line at a time, a field at a time, or a character at a time. In this way corrections or updates can be incorporated by allowing the terminal to power-type to the point of change, and keying in the change.

Request Mode

Request mode allows the operator to select one of several auxiliary operations. To select the desired operation, the operator places the terminal in request mode and keys in an identifier associated with the operation. This two-step procedure prevents inadvertent selection of an operating mode. The following paragraphs describe the operations that can be selected.

Communicate Mode

Communicate mode is used when communication with the CPU is desired. Either the operator or the FDP can place the terminal in communicate mode.

When inquiry is used on multipoint lines, the FDP establishes communicate mode without operator action. When dial-up communications are used, the operator must dial the CPU.

Generally, the operator selects communicate mode after the day's processing is complete. This leaves the terminal in condition to communicate with the CPU at the CPU's request. In this case, those fields marked for transmission are sent to the CPU.

Erase Data from CPU

This operation allows form records received from the CPU to be erased from the terminal's storage. The records are erased after they have been printed out and the terminal is being prepared to accept operator-created records.

KEYTOP	NAME	FUNCTION
ENTER FORM*	Enter Form	Places the terminal in normal entry mode under control of a previously selected FDP.
ERROR CORR*	Error Correct	Places the terminal in error correct mode to allow correction of an operator-detected error in a field preceding the field being processed but on the same line.
PLAYBACK*	Playback	Places the terminal in playback mode.
LOCAL*	Local	Places the terminal in local mode and under operator (as opposed to FDP) control.
CODE	Code	Allows use of the alternate function of the 1, 2, 3, 4, 5, and = keys.
CNCL FIELD	Cancel Field	Cancels data which has been entered into the field presently being processed, and positions the print element at the beginning of the field.
REQ	Request	Places the terminal in request mode to allow selection of one of the following: Communicate mode Erase data from CPU Erase all current operator created data Listing of resident FDPs Listing of resident data records On-line test (either of two) Diagnostic Test (one of four)
END CYCLE	End Cycle	Causes the FDP to end cycling (repetition of control for a series of fields) and skip to the field to be processed next.
END FORM	End Form	Signals FDP that a complete form record has been entered.
FORM #	Form Number	Allows selection of an FDP.
CNCL FORM	Cancel Form	Cancels a selected form record.

Mode Sensitive Keys

KEYTOP	NAME	MODE	FUNCTION
RETURN ENTER	Carrier Return and Enter	Local	Causes a carrier return and a line feed; if held at its second level, causes repetitive line feeds.
		Form #	Indicates that the FDP identifier has been entered.
		Enter Form	Indicates that a field has been entered.
ADV	Advance	Error Correct	Causes print element to advance one character without printing and without destroying information advanced over.
OPER	Operator	Enter Form	Unlocks keyboard after a program detected error.
		Playback	Initiates and halts power typing of the form record.
LINE	Line	Playback	Causes a line of data to print.
FIELD	Field	Error	Causes print element to advance one field without printing.
		Playback	Causes printing of one field.
CHAR	Character	Error	Causes print element to advance one character without printing.
		Playback	Causes one character to print.

* These keys are interlocked such that pressing one of the keys resets the others. The mode selected does not become active until OPER is pressed.

Figure 10. Operating Keys

Erase Operator-Created Data

This allows all operator-created data records residing in the terminal storage unit to be erased.

Listing of Resident FDPs

This provides a listing of identifiers and operator messages associated with each of the FDPs presently stored in the terminal.

Listing of Resident Data Records

This provides a listing of the identifiers associated with all data records presently stored in the terminal.

Diagnostic Tests

This allows selection of one of the terminal diagnostics or one of two on-line tests.

SPECIAL FEATURES

Multipoint Communications

This feature allows the 3735 to be multidropped with other BSC terminals on a single communications channel. The feature can be used on leased or privately owned communication facilities. Although the 3735 does not feature transparency, it can be intermixed on the same multipoint line with terminals that do have transparency.

INDICATOR	NAME	INDICATES
OPER MESSAGE	Operator Message	The TCP has a message for the terminal operator.
NUMERIC PAD	Numeric Pad	The keyboard is in numeric keypad mode.
COMM	Communicate	The terminal is in communicate mode and cannot be used for FDP controlled operations.
TMT/RCV	Transmit/Receive	Connection is established with the CPU and active communications are in process, or the CPU has attempted to select or to dial the terminal.
SELECT RECORD	Select Record	Operator can enter identifier of a record for playback.
SELECT FORM #	Select Form Number	Operator can enter identifier for desired FDP.
SELECT MODE	Select Mode	Operator can select the desired operating mode.
LOCAL	Local	The terminal is in local mode.
ALPHA/NUMERIC	Alpha/Numeric Entry Error	An attempt has been made to enter numeric information in a field specified by the FDP as an alphabetic field, or an attempt to enter alphabetic data in a field specified as numeric, or an attempt to enter non-alphameric in a field specified as alphameric.
LENGTH	Field Overrun or Incomplete	An attempt has been made to skip a mandatory field, to enter less than the specified minimum character length, or to exceed the specified maximum.
SELF CHECK	Self Check Digit Error	Detection of a self-checking digit error.
BUFFER FULL	Buffer Full	The terminal is presently (when the light comes on) storing in a sector of the disk three sectors (about 950 characters) from the end.
MACHINE CHECK	Machine Check	A machine-detected error.
PROCEED	Proceed	No error conditions exist and operation can proceed.
PROCEDURAL	Procedural	Operator has violated procedural sequences, such as attempting to enter playback mode prior to completion of a form record being created.
RANGE	Range	Field content entered did not lie between FDP specified minimum and maximum, or did not equal one or more specified constants.
INPUT/OUTPUT	Input/Output	The selected I/O device is not operational.

Figure 11. Operating Indicators

Synchronous Clock

This feature provides clocking for operation at 600 or 1200 bps with data sets not providing their own clocks.

Keylock

This feature provides a key-operated lock. When locked, it inhibits all operator activity related to input, output, or control of data.

5496 Adapter

This feature supplies the necessary controls for attaching the IBM 5496 Data Recorder to the 3735. The 5496 is a free-standing, buffered, operator-oriented key entry unit, used to create and to read the 96-column punched card. When attached to the 3735 and in auto mode, the 5496 can be used as a data source or as a data sink by the 3735. The data recorder cannot be used as a stand-alone keypunch when operating under control of the 3735.

Pinfeed Platens

Pinfeed platens allow the convenient use of continuous forms. These platens are available in the following sizes (sizes are in inches):

Overall Form Width	*Hole-to-Hole Width*	*Writing Line*
5 3/4	5 1/4	4 5/8
6 1/2	6	5 3/8
8	7 1/2	6 7/8
8 1/2	8	7 3/8
9 1/2	9	8 3/8
9 7/8	9 3/8	8 3/4
10 3/8	9 7/8	9 1/4
10 1/2	10	9 3/8
10 5/8	10 1/8	9 1/2
11 3/4	11 1/4	10 5/8
12	11 1/2	10 7/8
13	12 1/2	11 7/8
13 5/8	13 1/8	12 1/2

Each 3735 is shipped with the standard friction-feed platen, which the customer can replace with a pinfeed platen.

Forms used with pinfeed platens should be edge-stapled, glued, or stitched. The number of copies that can be satisfactorily printed depends on paper and carbon thickness, and on the height from which the forms are fed. A maximum of an original and four copies can be printed (using 12-pound paper and lightweight carbons).

Forms Stacker

This feature permits placing continuous forms on a stand above floor level, and provides for continuous-forms stacking after printing.

COMMUNICATIONS FACILITIES

The 3735 operates in half-duplex mode via appropriate data sets over the following facilities:

- Common-Carrier Switched Telecommunications Network at 1200 or 2000 bps.
- Common-Carrier Voice-Grade Private-Line Data Channels: Type 3002 Channel with C1 conditioning at 1200 bps; Type 3002 Channel with C2 conditioning at 2000 or 2400 bps.

  Note: Conditioning shown is required for common-carrier-provided data sets. Other data sets may have different conditioning requirements.

The private-line channels may be half-duplex or duplex, point-to-point* or multipoint. Although the 3735 operates in half-duplex mode, duplex private-line facilities are recommended to minimize line-turnaround times. Refer to the IBM Systems Reference Library publication, *General Information–Binary Synchronous Communications,* GA27-3004, for additional information and restrictions pertaining to communications facilities.

The data set (modem) required by the 3735 may be supplied by IBM, the common carrier, or the customer. The data-set interface must conform to the electrical characteristics as specified by EIA Standard RS-232C and be operationally compatible with the 3735 Data Set Attachment interface. Attachment is made via a data-set interface cable that is terminated by a 25-pin data-set connector plug. The following circuits described in RS-232C are utilized by the 3735:

Connector Pin	*RS-232C Designation*	*Circuit Description*
1	AA	Protective Ground
2	BA	Transmitted Data
3	BB	Received Data
4	CA	Request to Send
5	CB	Clear to Send
6	CC	Data Set Ready
7	AB	Signal Ground (Common Return)
8	CF	Received Line Signal Detector
15**	DB	Transmit Signal Element Timing
17**	DD	Receiver Signal Element Timing
20	CD	Data Terminal Ready
22	DE	Ring Indicator

Additional data-set-interface information is available from your IBM teleprocessing representative.

*The 3735 uses multipoint line control when used on point-to-point facilities.

**Utilized only at 2000 or 2400 bps when operating with synchronous data sets.

This section presents information useful in planning for installation of the 3735. The information is in three major catagories: "Forms Design", "Forms Description", and "System Considerations". Although segmented for descriptive purposes, the information is very much interrelated and should be considered as a single subject. For example, "Forms Description" outlines functions that can be specified for processing a given form (checking, editing, etc.), but the functions chosen may suggest changes in the form design. "System Considerations" discusses the formatting of data transmitted to and received from the CPU. Each of these activities may suggest changes in forms design and operating procedures. Thus, close interaction among the people planning the installation is a must.

The information presented here is for planning purposes. Detailed how-to information on macro coding and terminal operation will be provided in programmer's and in operator's guides.

FORMS DESIGN

The 3735 places few restrictions on forms design. The standard friction-feed platen accommodates forms up to 15 inches wide with up to 130 characters per line. Printing is at ten characters per inch; vertical spacing is at six lines per inch. An original and up to four carbon copies can be printed (using 12-pound paper and lightweight carbon). Several sizes of pinfeed platens are available (see "Special Features") which allow the use of continuous forms.

Forms are normally processed sequentially, left-to-right and top-to-bottom, during preparation. Therefore, the forms design should require development of information in this sequence. For example, totals should appear on the form after all items required for the total have been entered.

FORMS DESCRIPTION

Each form type processed on a 3735 must have a corresponding Form Description Program. The FDPs are developed from descriptions of the form. These descriptions define pages, lines, and fields of the form, and the functions desired for each.

In addition to specifying the way in which a document is processed, the FDP also specifies which data must be stored for transmission to the CPU. Consideration must be given not only to the data to be transmitted but also to its message format. Transmission-line time and CPU programming requirements can be reduced by limiting message content to include only information required at the CPU, and by utilizing to the maximum the local-processing capabilities of the 3735.

The following paragraphs describe functions that can be specified for the FDP. All or only a few may be used on any specific form; however, each function should be considered with the aim of (1) reducing operator keying requirements, (2) utilizing to the maximum the checking facilities available, and (3) making the most efficient use of CPU/terminal interaction.

Form descriptions should normally be developed sequentially from left to right, top to bottom, and page by page. A printer-spacing chart, form GX20-1776, is a convenient aid in establishing field and line positions on the form.

The descriptions are presented in the following sequence:

Form: Descriptions that apply to the overall form.

Page: Descriptions that apply to a single page of the form.

Line: Descriptions that apply to a single line of the form.

Field: Descriptions that apply to a single field of the form.

Control: Descriptions that are not directly related to a field.

Although this organization is similar to that used when coding the macros for FDP generation, the intent is not to teach coding techniques but rather to assist communications between the form designer and the person who does the coding. A complete description of the functions desired, presented in an organized manner, will ensure an efficient FDP.

Counters and logic indicators are mentioned in the descriptions to indicate 3735 capabilities. Manipulation of these units is best left to the person actually coding the form description macros.

The descriptions emphasize functions that are used for document creation. However, the same principles apply when preparing FDPs that are not associated with the creation of a form—for example, FDPs that allow the terminal operator to initiate inquiries to the CPU; FDPs for transmitting a file of cards from the 5496, or receiving a file of cards from the CPU, or for generating batch totals accumulated during the day's operation.

Form

The Form description must specify a three-digit identifier. The terminal operator uses this identifier to call out the program for processing a document.

An operator message can be specified for the form; this message is included within the body of the FDP and can be requested by the terminal operator before beginning processing under FDP control. The message may include instructions for document preparation, machine setup, application name, etc.

Tab settings to be used during processing of the form should be indicated in the Form description. Each stop is indicated by listing the character position at which the stop is to be set. The terminal operator can perform a tab set routine, under FDP control, before starting to process forms under this FDP.

One of three formats can be specified for fields that are to be transmitted to the CPU. The formats are:

> *Transmit Blanks:* Unused character positions of the field are filled with blanks for transmission to the CPU. Thus, a ten-character field in which only three characters are entered, during document creation, is transmitted as three characters followed by seven blanks. No characters are transmitted for fields that are skipped or bypassed.
>
> *Transmit Delimiter:* The field is transmitted with a delimiter following the last character entered. Thus, a ten-character field in which only three characters are entered is transmitted as three characters followed by the delimiter. If all characters of the field are entered, ten characters followed by the delimiter are transmitted.
>
> *No Delimiter:* The field is transmitted without inserted blanks or delimiter.

Page

The Page description provides identification and size information about the page. The size information is used during generation of the FDP to ensure that the fields and lines of the form will fit the page size intended. If the form consists of more than one page, each page must be described. If a single page is used, the Page description can be included with the Form description. Line and Field descriptions for the page follow each Page description.

Each Page description must identify the page being described. Identification is by number and optionally by name. Page numbers are normally in ascending sequence but need not be consecutive.

The overall height of the page is described by indicating the total number of usable line positions on the page; for example, a page 11 inches long has a height of 66 (at six lines per inch).

Vertical margins indicate the first line and the last line on which a field may appear. Vertical margins are indicated by line number, counting from the top of the page; for example, if the first field on a form may occur on the fourth line from the top, and the last field may occur on the 64th line, the vertical margins are 4 and 64.

Line

Line descriptions describe each line to be processed within a page. A Line description, identified by a line number, should be provided for each line to be processed by the 3735.

The Line description defines the line's horizontal space requirements. Each Line description should contain:

1. Line Identification: Express as a line number. Line numbers are normally in ascending sequence; however, they need not be consecutive.
2. Width: Express as the total number of character spaces available on the line (a page 8.5 inches wide has a width of 85).
3. Horizontal Margin: Express in character positions representing the first and last positions used.

If the width and horizontal margins of all lines on a page are the same, these descriptions can be included with the Page description.

If FDP control for a series of lines is identical in all fields, the series can be designated for cycling. It is then unnecessary to repeat Line and Field descriptions for each line; the first line or group of lines is described, and the number of repetitions is indicated. The line and field to be processed following the cycle must also be indicated. This cycle control is desirable for two reasons: it shortens the FDP, thus saving storage space on the 3735 disk; and it allows the operator to terminate cyclic operation if all lines are not required on a specific transaction.

Field

Field descriptions follow each Line description and detail the control desired for each field of the line. Many commonly used functions—such as centering data within a field, comma and decimal insertion, and self-check number verification—have been implemented within the terminal's control program. These functions are described in the following paragraphs. Other functions not specifically implemented—such as operator identification routines, inquiry routines, and batch totals—can be implemented using combinations of the functions described.

Each Field description is identified by the field's horizontal position on the line, expressed as the first (left) and the last (right) character positions of the field or, optionally, as the first character and the length. Maximum field size is 127 characters.

Field descriptions tell where the data comes from (source), what is done with it (operation), and where it is to go (sink).

Source

The keyboard is assumed to be the data source; if no other source is indicated, the operator must key in the data. Other sources that can be indicated are shown in Figure 12. A variety of checking functions can be specified for the input data; these are shown in Figure 13.

SOURCE	RESULT
Keyboard	Data originates from the Selectric I/O keyboard. Specifying Keypad activates the 10 digit numeric keypad. Specifying No Enter allows automatic advance when field entry is complete (operator need not press ENTER). Note: If no source is specified, the keyboard is assumed to be the data source.
Card Reader	Data originates from the 5496 card reader (from a card image buffer). Specify the first and last column of the desired field. Negative amounts must be indicated by an over-punch in the units position, thus: 0 = } 5 = N 1 = J 6 = O 2 = K 7 = P 3 = L 8 = Q 4 = M 9 = R Leading blanks of numeric fields are converted to zeros. Blanks following least-significant digit in left-justified fields are ignored.
Inquiry	Data originates from the inquiry buffer. Specify the first and last character positions of the desired field.
Storage	Data originates from the 236-byte special storage. Specify the first and last character positions of the desired field.
FDP ID	The three-digit identifier of the active FDP is used as data.
Record Number	The three-digit sequence number of the record in process is used as data. This number is automatically updated for each record created.
Counter	Data originates from a counter (10 digits and an optional sign).
Emitted	Data originates from the FDP. List the actual characters to be used, maximum = 125. If the characters represent a constant which is to be used in arithmetic operations, a maximum of 10 characters can be specified.

Figure 12. Data Source

Operation

Arithmetic and compare functions allow the source data to be processed as required by the application.

Arithmetic and compare operations available are shown in Figure 14.

KIND	RESULT
Alphameric	Alphabetic, numeric and space are accepted as valid data. Special characters are not accepted.
Alpha	Only alphabetic and space are accepted as valid data.
Numeric	Only digits 0 through 9 and a sign are accepted as valid data. If input is from the keyboard a sign character is accepted in the first character position. If input is not from the keyboard a minus sign over-punch is accepted in the units position; thus: 0 = } 5 = N 1 = J 6 = O 2 = K 7 = P 3 = L 8 = Q 4 = M 9 = R
Maximum	Checks that not more than the specified number of characters are entered.
Minimum	Checks that at least the number of characters specified are entered.
Fixed	Checks that exact number of characters specified are entered.
Value	Checks that the value of the data entered is greater than, equal to, or less than one or more specified values.
Self Check	Checks the validity of a self-checking number. Either modulus 10 or 11 can be specified.
Optional	Suppresses self check, value, and fixed or minimum checks if the operator makes no entries in the field during document creation (presses ENTER).
Batch Number	Flags the input as belonging to a batch indicated by the specified batch number. Batch numbers from 0 through 127 can be specified. This provides the ability under FDP control to automatically accumulate batch totals for up to and including 128 different batches.

Figure 13. Content Checks

Sink

The Selectric® I/O II printer is assumed to be the destination of data from the specified source. Other destinations that can be selected are shown in Figure 15; choosing one or more of these does not prevent printing the data as well. Unless specified otherwise, each field is printed after entry is complete. This allows checking the data before printing it on the document. If print-while-entering is specified, each character prints as it is keyed; this suppresses all edit and justify functions. If a field is not to be printed, no-print must be specified.

OPERATION	RESULT
Clear	Specified counter is set to + zero.
Add	Source data is added to specified counter.
Subtract	Source data is subtracted from specified counter.
Multiply	Source data is multiplied by specified counter.
Divide	Source data is divided into specified counter, remainder is ignored.
Divide and Round	Source data is divided into specified counter, remainder is used to half-adjust units digit of quotient.
Compare	Source data is compared to one or more specified values, the result of the comparison is used to set, reset, or invert a specified indicator.

Results of arithmetic operations appear in the specified counter, and cannot exceed 10 digits. The number of operations (for example, add to counter 1 and to counter 2) which can be specified is not limited.

Figure 14. Arithmetic and Compare Operations

The way in which the data is "pictured" for output is indicated by edit specifications. Figure 16 shows editing functions that can be described for each position of the output field. Figures 17-19 show examples of how the edit functions are used to picture the output field.

Control

This description provides for testing of logical conditions (indicators), modifying the content of counters, and nonsequential processing. Use of these logical and arithmetic functions can implement such functions as: table lookup; field skipping; and password identification. The forms designer should define the function desired and the data available; this allows the person doing the macro coding to implement an FDP that provides exactly the functions desired.

Card reading and punching instructions are also included in the Control description. A Read Card instruction signals the 5496 to read a card and the data enters a card-image buffer where it is available for use by the FDP. Similarly, a Punch Card instruction signals the 5496 to accept data from a card-image buffer, and punch the data into a card.

SYSTEM CONSIDERATIONS

The preceding sections have described functions which can be specified for creation of forms on the 3735. The installation planners must also consider how the 3735 will be incorporated into the total system. This section provides information on how the 3735 formats data for transmission, and how data from the CPU must be formatted for proper handling by the 3735.

SINK	RESULT
Print:	
While	Output prints while it is being keyed.
After	Output prints after entry and checking is complete.
Center	Output is centered with leading and trailing blanks.
Zero Fill	Output is right-justified with leading zeros.
Blank Fill	Output is right-justified with leading blanks.
Underline	Output is underlined.
Edit	Output prints with specified editing.
Card Punch	Output is stored into a card image buffer starting at the character position specified. Right justify with leading blanks or zeros can be specified. Negative amounts are over-punched in the units position, thus: 0 = } 5 = N 1 = J 6 = O 2 = K 7 = P 3 = L 8 = Q 4 = M 9 = R
Inquiry	Output is stored in the inquiry buffer starting at the character position specified. Right justify with leading blanks or zeros can be specified. Negative amounts are stored with an alphabetic character in the units position; see 'Punch' above.
Storage	Output is stored in the special storage area starting at the character position specified. Right justify with leading blanks or zeros can be specified. Negative amounts are stored with an alphabetic character in the units position; see 'Punch' above.
The following additional functions can be specified for the output data.	
No Print	Field is not printed.
No Transmit	Source data is not transmitted.
Transmit Edit	Data generated by the justify or edit functions is transmitted.

Figure 15. Data Sink

Skipped Fields

During document creation, some fields of a form may be skipped. Information previously entered on the form may cause the skip; for example, the FDP may specify that a credit entry causes skipping of a debit field. The terminal operator may initiate the skip when all fields within a group of fields designated for cycling are not used, or the operator may elect to pass an optional field because the information is not required for the transaction being processed.

EDIT	RESULT
9	A decimal digit is accepted for output in the associated position.
Z	The associated position is made blank if it contains a leading zero.
B	A blank is inserted in the associated position.
Y	The associated position is made blank if it contains a zero.
*	An asterisk is placed in the associated position if the position contains a leading zero.
,	A comma is inserted in the associated position if zeros are not being suppressed.
/	A slash is inserted in the associated position if zeros are not being suppressed.
.	A period is inserted in the associated position if zeros are not being suppressed.
V	Stops zero suppression.
$	A dollar sign is inserted in the associated position; if more than one $ is indicated, the dollar sign is inserted to the left of the most significant digit.
-	A minus sign is placed in the associated position if the field is less than 0; if more than one - is indicated, the minus sign is inserted to the left of the most significant digit.
+	A plus sign is inserted in the associated position if the field is equal to or greater than zero; if more than one + is indicated, the plus sign is inserted to the left of the most significant digit.
S	A minus sign is placed in the associated position if the field is less than zero; a plus sign is placed in the associated position if the field is equal to or greater than zero. If more than one S is indicated, the sign is inserted to the left of the most significant digit.
CR	A CR is inserted in the associated positions if the field is less than zero; otherwise, the associated positions are blank.
DB	A DB is inserted in the associated positions if the field is equal to or greater than zero; otherwise, the associated positions are blank.

Figure 16. Edit Functions

Fields in which the operator does not enter data are not considered skipped fields. These fields are transmitted with or without blanks or delimiter as specified by the FDP.

No automatic indication of fields skipped is provided for transmission to the CPU. Thus, the installation planner must consider how the fields are to be identified by the CPU. The FDP may be written such that all fields are filled, either with input data or with blanks; thus, the CPU can identify each field by its position within the record.

Transmittal message length can be reduced by identifying skips by a unique character string inserted after the skip. This allows the CPU program to recognize that fields have been skipped and to identify the data which follows.

SOURCE DATA	PICTURE SPECIFICATION	RESULTANT OUTPUT
12345	ZZZ99	12345
00100	ZZZ99	bb100
00000	ZZZ99	bbb00
00100	ZZZZZ	bb100
00000	ZZZZZ	bbbbb
00100	*****	**100
00000	*****	*****
00100	YYYYY	bb1bb
10203	9Y9Y9	1b2b3

Figure 17. Examples of Zero Suppression

SOURCE DATA	PICTURE SPECIFICATION	RESULTANT OUTPUT
12345	$999.99	$123.45
00123	$ZZZ.99	$bb1.23
00000	$ZZZ.ZZ	$bbbbbb
12345	$$$9.99	$123.45
00123	$$$9.99	bb$1.23
12	99$	12$
12	$$$,999	bbb$012
1234	$$$,999	b$1,234
12345	S999.99	+123.45
-12345	S999.99	-123.45
-12345	+999.99	b123.45
12345	-999.99	b123.45
12345	999.99S	123.45+
00123	++B+9.99	bbb+1.23
00123	---9.99	bbb1.23
-00123	SSS9.99	bb-1.23

Figure 18. Examples of Drifting Characters

Lines and fields skipped when an end-cycle function is performed are treated in the same way as described above; no automatic indication is provided. Thus, inserted identifiers may be required.

Form Records

An end-of-record indication (RS for ASCII machines, IRS for EBCDIC machines) is inserted automatically at the end of each form record transmitted from the 3735. Each record received by the 3735 must also end with the end-of-record indication.

Records received by the 3735 must also have the identifier of the associated FDP as the first three characters of the record. This is required to indicate which FDP must be used during output of the record.

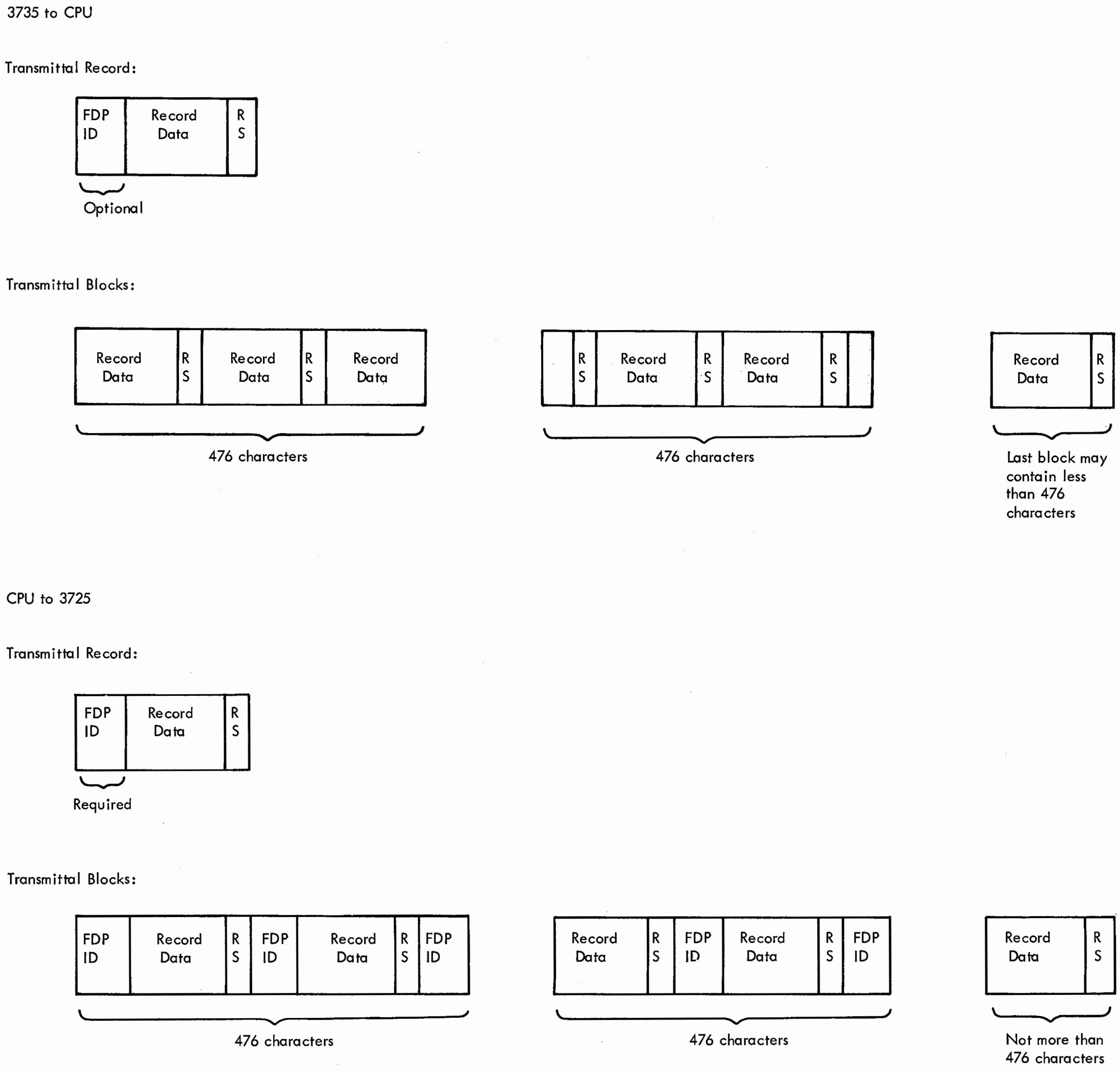
The FDP identifier is not supplied automatically on records transmitted from the 3735, but can be inserted by the FDP. Figure 20 summarizes the record formats.

Transmittal Blocks

Message blocks transmitted to the 3735 must not exceed 476 characters. Block length may be less than 476 characters; however, during reception each block is stored as a sector of information on the magnetic disk. Therefore blocks less than 476 characters reduce the overall message length that can be accommodated by the 3735 storage. Following reception of a message, the Terminal Control Program searches the received records and packs them into contiguous blocks; thus, the storage reduction is applicable only during reception of the short blocks.

SOURCE DATA	PICTURE SPECIFICATION	OUTPUT FIELD
1234	9,999	1,234
123456	9,999.99	1,234.56
1234	ZZ.ZZ	12.34
1234	ZZV.99	12.34
0003	ZZ.ZZ	bbbb3
0003	ZZV.99	bb.03
0000	ZZ.ZZ	bbbbb
0000	ZZV.99	bb.00
123456789	9,999,999.99	1,234,567.89
1234567	**,999.99	12,345.67
0012345	**,999.99	***123.45
123456789	9.999.999,99	1.234.567,89
123456	99/99/99	12/34/56
123456	99.9/99.9	12.3/45.6
001234	ZZ/ZZ/ZZ	bbb12/34
000012	ZZ/ZZ/ZZ	bbbbbb12
000000	ZZ/ZZ/ZZ	bbbbbbbb
000000	**/**/**	********
123456	99B99B99	12b34b56
123	9BB9BB9	1bb2bb3
12	9BB/9BB	1bb/2bb

Figure 19. Examples of Insertion Characters



Notes: IRS replaces RS when EBCDIC is used.

Data -link control characters and sequences not shown.

Figure 20. Transmission Formats

Figures A-1 and A-2 present the EBCDIC and USASCII code charts.

CHAR.	EBCDIC CODE								HEX
NUL									00
SOH								7	01
STX							6		02
ETX							6	7	03
PF						5			04
HT						5		7	05
LC						5	6		06
DEL						5	6	7	07
					4				08
RLF					4			7	09
SMM					4		6		0A
VT					4		6	7	0B
FF					4	5			0C
CR					4	5		7	0D
SO					4	5	6		0E
SI					4	5	6	7	0F
DLE				3					10
DC1				3				7	11
DC2				3			6		12
DC3(TM)				3			6	7	13
RES				3		5			14
NL				3		5		7	15
BS				3		5	6		16
IL				3		5	6	7	17
CAN				3	4				18
EM				3	4			7	19
CC				3	4		6		1A
CU1				3	4		6	7	1B
IFS				3	4	5			1C
IGS				3	4	5		7	1D
IRS				3	4	5	6		1E
IUS				3	4	5	6	7	1F
DS			2						20
SOS			2					7	21
FS			2				6		22
			2				6	7	23
BYP			2			5			24
LF			2			5		7	25
ETB(EOB)			2			5	6		26
ESC(PRE)			2			5	6	7	27
			2		4				28
			2		4			7	29
SM			2		4		6		2A
CU2			2		4		6	7	2B
			2		4	5			2C
ENQ			2		4	5		7	2D
ACK			2		4	5	6		2E
BEL			2		4	5	6	7	2F
			2	3					30
			2	3				7	31
SYN			2	3			6		32
			2	3			6	7	33
PN			2	3		5			34
RS			2	3		5		7	35
UC			2	3		5	6		36
EOT			2	3		5	6	7	37
			2	3	4				38
			2	3	4			7	39
			2	3	4		6		3A
CU3			2	3	4		6	7	3B
DC4			2	3	4	5			3C
NAK			2	3	4	5		7	3D
			2	3	4	5	6		3E
SUB			2	3	4	5	6	7	3F

CHAR.	EBCDIC CODE								HEX
SPACE		1							40
		1						7	41
		1					6		42
		1					6	7	43
		1				5			44
		1				5		7	45
		1				5	6		46
		1				5	6	7	47
		1			4				48
		1			4			7	49
¢		1			4		6		4A
.		1			4		6	7	4B
<		1			4	5			4C
(		1			4	5		7	4D
+		1			4	5	6		4E
|		1			4	5	6	7	4F
&		1		3					50
		1		3				7	51
		1		3			6		52
		1		3			6	7	53
		1		3		5			54
		1		3		5		7	55
		1		3		5	6		56
		1		3		5	6	7	57
		1		3	4				58
		1		3	4			7	59
!		1		3	4		6		5A
$		1		3	4		6	7	5B
*		1		3	4	5			5C
)		1		3	4	5		7	5D
;		1		3	4	5	6		5E
¬		1		3	4	5	6	7	5F
-		1	2						60
/		1	2					7	61
		1	2				6		62
		1	2				6	7	63
		1	2			5			64
		1	2			5		7	65
		1	2			5	6		66
		1	2			5	6	7	67
		1	2		4				68
		1	2		4			7	69
		1	2		4		6		6A
,		1	2		4		6	7	6B
%		1	2		4	5			6C
—		1	2		4	5		7	6D
>		1	2		4	5	6		6E
?		1	2		4	5	6	7	6F
		1	2	3					70
		1	2	3				7	71
		1	2	3			6		72
		1	2	3			6	7	73
		1	2	3		5			74
		1	2	3		5		7	75
		1	2	3		5	6		76
		1	2	3		5	6	7	77
		1	2	3	4				78
		1	2	3	4			7	79
:		1	2	3	4		6		7A
#		1	2	3	4		6	7	7B
@		1	2	3	4	5			7C
'		1	2	3	4	5		7	7D
=		1	2	3	4	5	6		7E
"		1	2	3	4	5	6	7	7F

CHAR.	EBCDIC CODE								HEX
	0								80
a	0							7	81
b	0						6		82
c	0						6	7	83
d	0					5			84
e	0					5		7	85
f	0					5	6		86
g	0					5	6	7	87
h	0				4				88
i	0				4			7	89
	0				4		6		8A
	0				4		6	7	8B
	0				4	5			8C
	0				4	5		7	8D
	0				4	5	6		8E
	0				4	5	6	7	8F
	0			3					90
j	0			3				7	91
k	0			3			6		92
l	0			3			6	7	93
m	0			3		5			94
n	0			3		5		7	95
o	0			3		5	6		96
p	0			3		5	6	7	97
q	0			3	4				98
r	0			3	4			7	99
	0			3	4		6		9A
	0			3	4		6	7	9B
	0			3	4	5			9C
	0			3	4	5		7	9D
	0			3	4	5	6		9E
	0			3	4	5	6	7	9F
	0		2						A0
	0		2					7	A1
s	0		2				6		A2
t	0		2				6	7	A3
u	0		2			5			A4
v	0		2			5		7	A5
w	0		2			5	6		A6
x	0		2			5	6	7	A7
y	0		2		4				A8
z	0		2		4			7	A9
	0		2		4		6		AA
	0		2		4		6	7	AB
	0		2		4	5			AC
	0		2		4	5		7	AD
	0		2		4	5	6		AE
	0		2		4	5	6	7	AF
	0		2	3					B0
	0		2	3				7	B1
	0		2	3			6		B2
	0		2	3			6	7	B3
	0		2	3		5			B4
	0		2	3		5		7	B5
	0		2	3		5	6		B6
	0		2	3		5	6	7	B7
	0		2	3	4				B8
	0		2	3	4			7	B9
	0		2	3	4		6		BA
	0		2	3	4		6	7	BB
	0		2	3	4	5			BC
	0		2	3	4	5		7	BD
	0		2	3	4	5	6		BE
	0		2	3	4	5	6	7	BF

CHAR.	EBCDIC CODE								HEX
{	0	1							C0
A	0	1						7	C1
B	0	1					6		C2
C	0	1					6	7	C3
D	0	1				5			C4
E	0	1				5		7	C5
F	0	1				5	6		C6
G	0	1				5	6	7	C7
H	0	1			4				C8
I	0	1			4			7	C9
	0	1			4		6		CA
	0	1			4		6	7	CB
⑀	0	1			4	5			CC
	0	1			4	5		7	CD
⑂	0	1			4	5	6		CE
	0	1			4	5	6	7	CF
}	0	1		3					D0
J	0	1		3				7	D1
K	0	1		3			6		D2
L	0	1		3			6	7	D3
M	0	1		3		5			D4
N	0	1		3		5		7	D5
O	0	1		3		5	6		D6
P	0	1		3		5	6	7	D7
Q	0	1		3	4				D8
R	0	1		3	4			7	D9
	0	1		3	4		6		DA
	0	1		3	4		6	7	DB
	0	1		3	4	5			DC
	0	1		3	4	5		7	DD
	0	1		3	4	5	6		DE
	0	1		3	4	5	6	7	DF
\	0	1	2						E0
	0	1	2					7	E1
S	0	1	2				6		E2
T	0	1	2				6	7	E3
U	0	1	2			5			E4
V	0	1	2			5		7	E5
W	0	1	2			5	6		E6
X	0	1	2			5	6	7	E7
Y	0	1	2		4				E8
Z	0	1	2		4			7	E9
	0	1	2		4		6		EA
	0	1	2		4		6	7	EB
⑁	0	1	2		4	5			EC
	0	1	2		4	5		7	ED
	0	1	2		4	5	6		EE
	0	1	2		4	5	6	7	EF
0	0	1	2	3					F0
1	0	1	2	3				7	F1
2	0	1	2	3			6		F2
3	0	1	2	3			6	7	F3
4	0	1	2	3		5			F4
5	0	1	2	3		5		7	F5
6	0	1	2	3		5	6		F6
7	0	1	2	3		5	6	7	F7
8	0	1	2	3	4				F8
9	0	1	2	3	4			7	F9
	0	1	2	3	4		6		FA
	0	1	2	3	4		6	7	FB
	0	1	2	3	4	5			FC
	0	1	2	3	4	5		7	FD
	0	1	2	3	4	5	6		FE
	0	1	2	3	4	5	6	7	FF

Figure A-1. EBCDIC

CHAR.	USASCII CODE								HEX
NUL									00
SOH	1								01
STX		2							02
ETX	1	2						P	03
EOT			3						04
ENQ	1		3					P	05
ACK		2	3					P	06
BEL	1	2	3						07
BS				4					08
HT	1			4				P	09
LF		2		4				P	0A
VT	1	2		4					0B
FF			3	4				P	0C
CR	1		3	4					0D
SO		2	3	4					0E
SI	1	2	3	4				P	0F
DLE					5				10
DC1	1				5			P	11
DC2		2			5			P	12
DC3	1	2			5				13
DC4			3		5			P	14
NAK	1		3		5				15
SYN		2	3		5				16
ETB	1	2	3		5			P	17
CAN				4	5			P	18
EM	1			4	5				19
SUB		2		4	5				1A
ESC	1	2		4	5			P	1B
FS			3	4	5				1C
GS	1		3	4	5			P	1D
RS		2	3	4	5			P	1E
US	1	2	3	4	5				1F
SPACE						6			20
!	1					6		P	21
"		2				6		P	22
#	1	2				6			23
$			3			6		P	24
%	1		3			6			25
&		2	3			6			26
'	1	2	3			6		P	27
(				4		6		P	28
)	1			4		6			29
*		2		4		6			2A
+	1	2		4		6		P	2B
,			3	4		6			2C
-	1		3	4		6		P	2D
.		2	3	4		6		P	2E
/	1	2	3	4		6			2F
0					5	6		P	30
1	1				5	6			31
2		2			5	6			32
3	1	2			5	6		P	33
4			3		5	6			34
5	1		3		5	6		P	35
6		2	3		5	6		P	36
7	1	2	3		5	6			37
8				4	5	6			38
9	1			4	5	6		P	39
:		2		4	5	6		P	3A
;	1	2		4	5	6			3B
<			3	4	5	6		P	3C
=	1		3	4	5	6			3D
>		2	3	4	5	6			3E
?	1	2	3	4	5	6		P	3F

CHAR.	USASCII CODE								HEX
@							7		40
A	1						7	P	41
B		2					7	P	42
C	1	2					7		43
D			3				7	P	44
E	1		3				7		45
F		2	3				7		46
G	1	2	3				7	P	47
H				4			7	P	48
I	1			4			7		49
J		2		4			7		4A
K	1	2		4			7	P	4B
L			3	4			7		4C
M	1		3	4			7	P	4D
N		2	3	4			7	P	4E
O	1	2	3	4			7		4F
P					5		7	P	50
Q	1				5		7		51
R		2			5		7		52
S	1	2			5		7	P	53
T			3		5		7		54
U	1		3		5		7	P	55
V		2	3		5		7	P	56
W	1	2	3		5		7		57
X				4	5		7		58
Y	1			4	5		7	P	59
Z		2		4	5		7	P	5A
[	1	2		4	5		7		5B
\			3	4	5		7	P	5C
]	1		3	4	5		7		5D
Λ		2	3	4	5		7		5E
—	1	2	3	4	5		7	P	5F
`						6	7	P	60
a	1					6	7		61
b		2				6	7		62
c	1	2				6	7	P	63
d			3			6	7		64
e	1		3			6	7	P	65
f		2	3			6	7	P	66
g	1	2	3			6	7		67
h				4		6	7		68
i	1			4		6	7	P	69
j		2		4		6	7	P	6A
k	1	2		4		6	7		6B
l			3	4		6	7	P	6C
m	1		3	4		6	7		6D
n		2	3	4		6	7		6E
o	1	2	3	4		6	7	P	6F
p					5	6	7		70
q	1				5	6	7	P	71
r		2			5	6	7	P	72
s	1	2			5	6	7		73
t			3		5	6	7	P	74
u	1		3		5	6	7		75
v		2	3		5	6	7		76
w	1	2	3		5	6	7	P	77
x				4	5	6	7	P	78
y	1			4	5	6	7		79
z		2		4	5	6	7		7A
{	1	2		4	5	6	7	P	7B
¦			3	4	5	6	7		7C
}	1		3	4	5	6	7	P	7D
~		2	3	4	5	6	7	P	7E
DEL	1	2	3	4	5	6	7		7F

Note: HEX shown corresponds to bit order on System/360 channel;
P-bit not included in HEX.

Figure A-2. USASCII

FORM DESCRIPTION MACROS

Assembly

The System/360 and System/370 OS and DOS Assembler Language macro instructions provide the user with a means of describing his forms. The output of the system assembly is an object module suitable for input to the FD utility. Also produced is an assembly listing that includes diagnostic and error information concerning the macro instructions. The assembly provides the following functions:

- Check input statements for completeness and accuracy.
- Provide meaningful comments to aid in debugging the program.
- Calculate number of spaces, carrier returns, etc., for proper print-element positioning.
- Generate FD program sequences to satisfy the requirements of the 3735.
- Sequence-number output records (object program).
- Set flags in the output if the input statements contain errors that invalidate the output.

The output of the system is object modules whose data portions are FD-unpacked programs. These object modules can be punched in cards or written in a data set as card images. Modules are then used as the input to the FD utility.

Utility

The FD utility is IBM's support for the preparation of the FD-unpacked programs for transmission to the 3735 terminals. The utility operates basically the same under OS and DOS. It reads the output of the assembly from a card reader or equivalent sequential input device. It does a certain amount of checking for program integrity. It formats the FD-unpacked programs into blocks of 476 bytes. Then the utility writes the formatted FD-unpacked program blocks into a user-specified data set that is available to the user's application program.

The main storage required by the FD utility is that required for the minimum OS or DOS Linkage Editor available to the system. The main storage required by the utility exclusive of the Linkage Editor step is no more than 10K bytes. The following three I/O devices are required: a card reader or equivalent device; a printer; and a direct-access storage device. The utility uses no more than 10 tracks of 2311 storage, or the equivalent, for program residence. The user's secondary storage requirements depend on the number and size of the forms being described.

The FD utility is scheduled through the job input stream, encompassing three sequential job steps: Control, Linkage Editor, and Storage. There is one optional feature: in OS, the REPLACE parameter; in DOS, the UPSI statement. These will be discussed later.

To include the FD utility in his operating system, the user must copy it from the component library on which it is distributed. This may require allocating additional space. Prior to the execution of the FD utility, the system programmer must ensure that suitable input and output data sets are designated through correct coding of the job control statements.

If an uncorrectable error is encountered, the FD utility takes the appropriate action, depending on which step of the three-step sequence is being executed. This is either the termination of the job or the deletion of a partially created sequence of blocks from the user's data set. A diagnostic listing is produced at the end of each of the three steps; the listing produced by the last step names each FD-unpacked program that was added to, deleted from, or not added to the user's data set, and the space required on the 3735 disk.

The response to environmental errors arising from improper input, such as a card missing or out of sequence, is to write a message in the diagnostic listing, produced in the control step, and to exit. Input following the erroneous card is not processed.

The response to errors such as insufficient allocation of space in a data set is to terminate processing and to note the condition in the listing.

The response to implementation errors, such as errors in system control blocks, is those error recovery and termination options provided by OS and DOS.

In OS, if a form is a duplicate (one with the same form name) of a member in the user's library and the REPLACE option has not been specified, the form is stored as a new member under a temporary name: for example, TEMP01, TEMP02, . . . , TEMP09. If these temporary names have already been exhausted, the new member is not stored. The form is replaced if the REPLACE option has specified that all duplicates are to be replaced, or has specified, by name, the forms to be replaced. The action taken is noted in the listing.

In DOS, if a form is a duplicate and no UPSI statement was present in the job stream, the records are stored with a temporary key in the user's indexed sequential data set, if possible. If the UPSI option is used, it can be coded in two ways. It can be coded with all 1's (// UPSI 11111111), which means that all duplicates are to be replaced. It can also be coded with a mixture of 1's and 0's or all 0's; for example, // UPSI 01010101. Forms 1-8 in the input sequence replaces a previous version according to whether a 0 or a 1 is coded in the respective place in the UPSI byte. Forms 9, 10, ---- have the default of 0; that is, temporary key is used if a duplicate exists. As in OS, the action taken is noted in the listing.

The diagnostic listing produced from the execution of each of the three steps contains information of interest to the programmer adding, creating, modifying, or extending a library of FD-unpacked programs. It includes a heading identifying the step and then a list of messages.

Following are some examples of the types of diagnostics noted in the listing:

Successful completion
Last valid record (card) number
Erroneous record (card) number
Invalid sequence number
Invalid deck ID
Premature end-of-file
Invalid card type
Name of FD program stored
Duplicate name, if it was stored and if so, under
 what name
Insufficient space

The messages that are listed vary slightly between OS and DOS. Response to any errors indicated is to correct the error and re-execute the job.

Transmission

The output of the FD utility is 476-byte blocks containing macro-generated bit strings forming the data portion of FD messages. These FD programs reside in a user's data set. The user must create a teleprocessing application program to select and transmit the FD programs, just as he must create more complex programs to process the data captured and transmitted by the 3735 terminals

In transmitting the FD programs to the individual terminals, not all FD programs in the user's library must be transmitted to all 3735 terminals. The user can be selective, sending only certain FD programs from the library to certain terminals. However, when transmitting the FD programs from the user's data set to the terminal, all the FD programs that are to reside at a terminal must be transmitted at the same time. Additional FD programs cannot be added at the terminal. The FD programs already there must be retransmitted along with any additional ones, to get the total number of FD programs desired at the terminal.

The FD programs must be transmitted according to the following scheme:

1. The 3735 terminal must transmit its data to the CPU first.
2. The CPU must transmit a user-prepared block which informs the 3735 that the following transmission blocks are FD programs.
3. The CPU transmits the FD programs in 476-byte unpacked blocks.
4. The CPU then transmits other data, if any.

Appendix C. Form Description Example

The following paragraphs present a sample form, and show how this form can be described to develop an FDP for processing it. This example illustrates how the lines and fields of the form, and the processing desired, are indicated. The completed description would be converted into macro statements for assembly by the CPU's operating system and development of the FDP. The macro statements would closely follow the description shown; however, they would reflect the rules of syntax and structure required by the operating system macro language.

SAMPLE APPLICATION

The form to be described is a Sales Invoice used by the local warehouse of a centralized distributing company. In this application, salesmen submit their orders each day. The orders are batched, and processed the next morning.

Figure C-1 shows how source information for processing the Sales Invoice is developed: the Order Sheets would typically be item listings on which the salesman indicates quantity ordered, quoted price, and customer number. These are sent to the Order Department, where shipped and backordered quantities and method of shipping are added. Customer master cards, from a tub file, complete the source information required to create the Sales Invoice.

Using this source information, the 3735 operator, with guidance from the FDP, creates the invoice. Completed Sales Invoices are used as ship orders, for customer verification, and as a local record of the transaction. Captured data is sent to the CPU for use in billing and inventory control.

DESCRIBING THE FORM

Figure C-2 shows the form to be described. The first step in describing the form is identifying its lines and fields. Figure C-3 shows how the printer spacing chart is used to locate line and character positions corresponding to the fields of the preprinted form. Each line is indicated by its line number, each field by its first and last character position. Figure C-4 shows the form after all lines and fields have been recorded. Note that not all lines for the body of the form have been identified; this is allowed because each of these lines have identical control and need not be described separately.

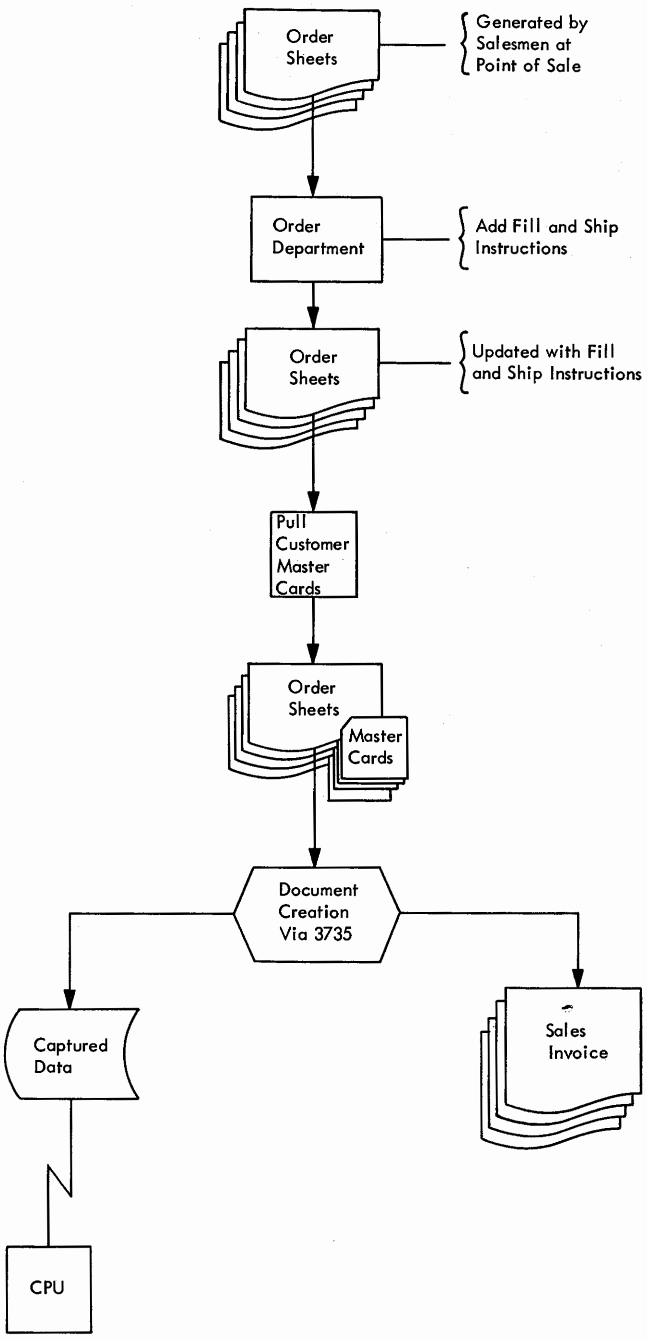
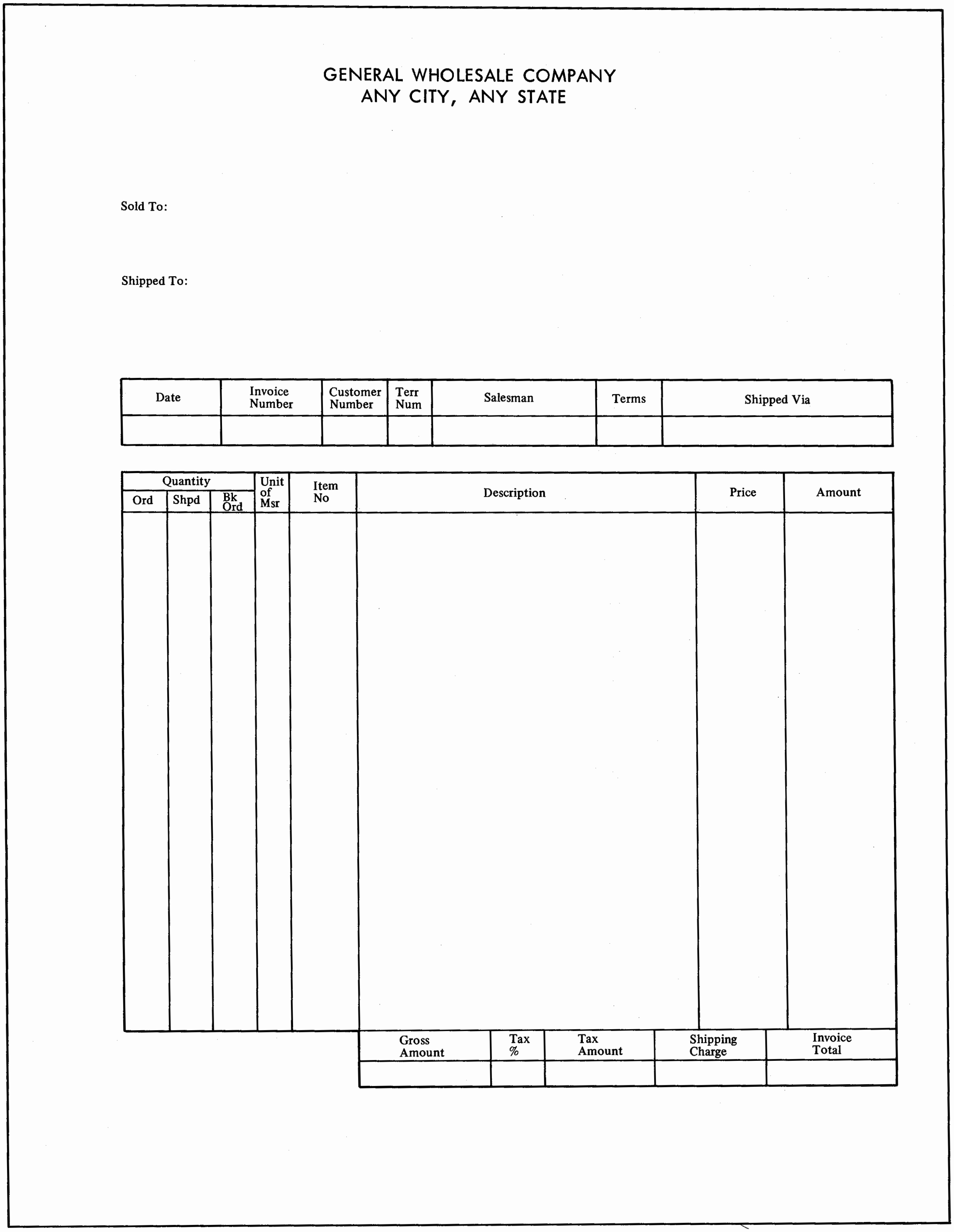

Order Sheets
Generated by Salesmen at Point of Sale
Order Department
Add Fill and Ship Instructions
Order Sheets
Updated with Fill and Ship Instructions
Pull Customer Master Cards
Order Sheets
Master Cards
Document Creation Via 3735
Captured Data
Sales Invoice
CPU

Figure C-1. Application Example

A customer master card is marked with the fields that are used on the Sales Invoice.

The form description is now written by reference to the annotated form and master card. Functions that can be specified have been described in the "Implementation Considerations" section of this manual. Figure C-5 shows the completed description. Although this description shows counter and indicator manipulation, this level of detail would not be required; if a description of the results desired and the data to be used were provided, these functions could be implemented by the person doing the macro coding.

Figure C-6 shows a completed form processed under control of the FDP resulting from the illustrated description. Figure C-7 shows the record format transmitted to the CPU for this completed form.



GENERAL WHOLESALE COMPANY
ANY CITY, ANY STATE

Sold To:

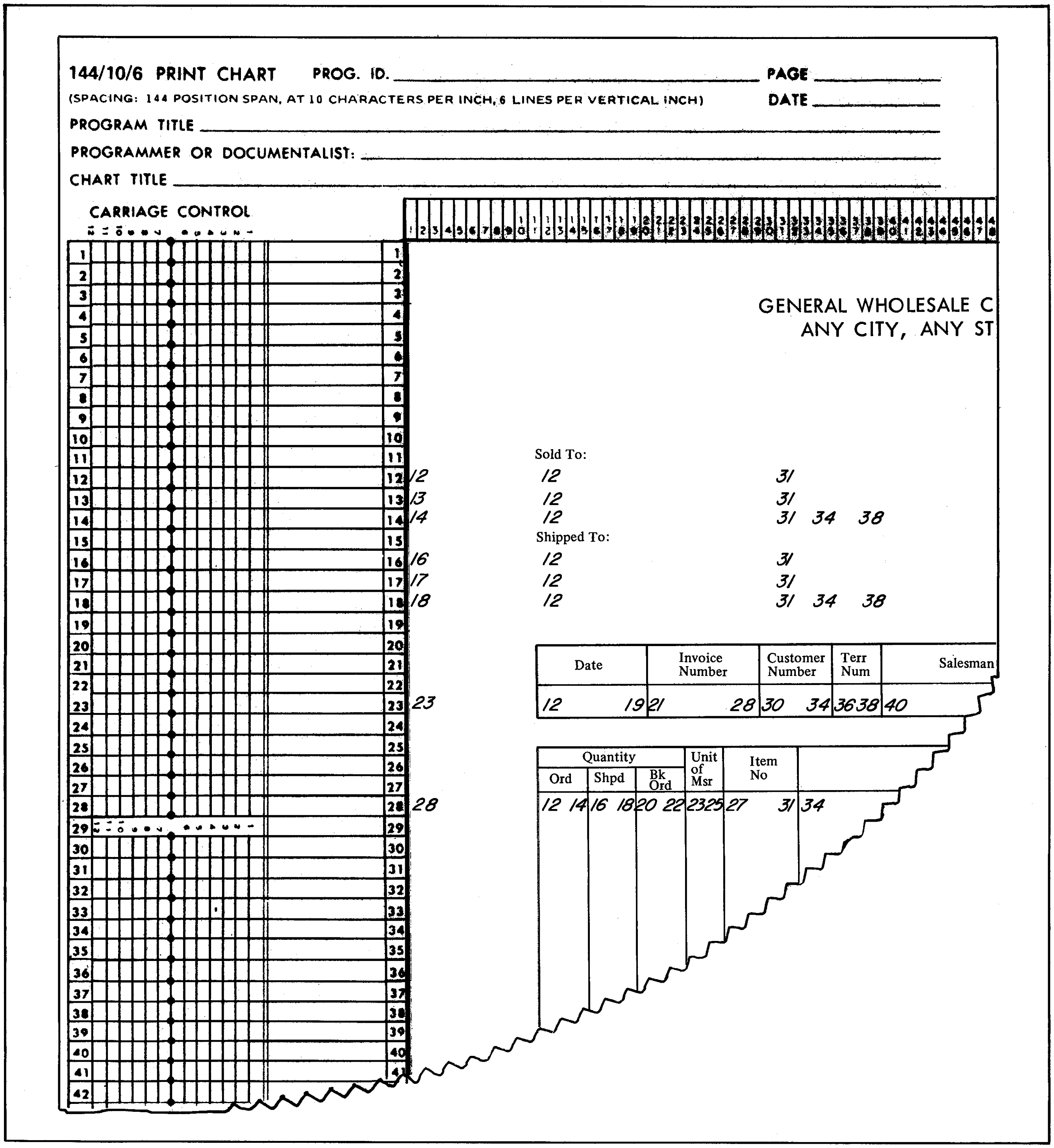
Shipped To:

Date	Invoice Number	Customer Number	Terr Num	Salesman	Terms	Shipped Via

Quantity			Unit of Msr	Item No	Description	Price	Amount
Ord	Shpd	Bk Ord					

Gross Amount	Tax %	Tax Amount	Shipping Charge	Invoice Total

Figure C-2. Sales Invoice Form



144/10/6 PRINT CHART PROG. ID. ______ PAGE ______

(SPACING: 144 POSITION SPAN, AT 10 CHARACTERS PER INCH, 6 LINES PER VERTICAL INCH) DATE ______

PROGRAM TITLE ______

PROGRAMMER OR DOCUMENTALIST: ______

CHART TITLE ______

CARRIAGE CONTROL

GENERAL WHOLESALE C
ANY CITY, ANY ST

Sold To:

12	12	31		
13	12	31		
14	12	31	34	38

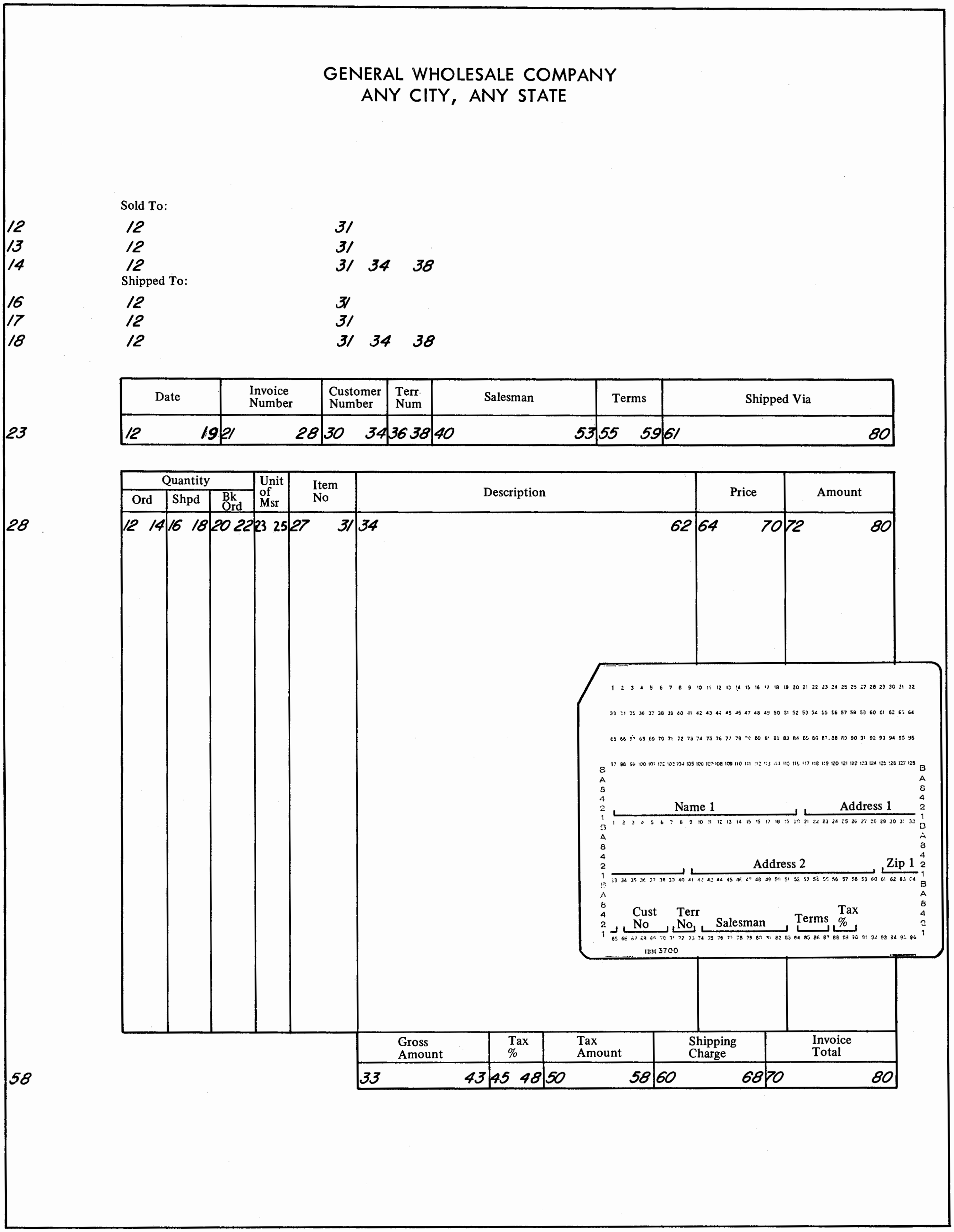
Shipped To:

16	12	31		
17	12	31		
18	12	31	34	38

	Date	Invoice Number	Customer Number	Terr Num	Salesman
23	12 19	21 28	30 34	36 38	40

	Quantity			Unit of Msr	Item No	
	Ord	Shpd	Bk Ord			
28	12 14	16 18	20 22	23 25	27 31	34

Figure C-3. Locating Lines and Fields on the Form



GENERAL WHOLESALE COMPANY
ANY CITY, ANY STATE

Sold To:

12 12 31
13 12 31
14 12 31 34 38

Shipped To:

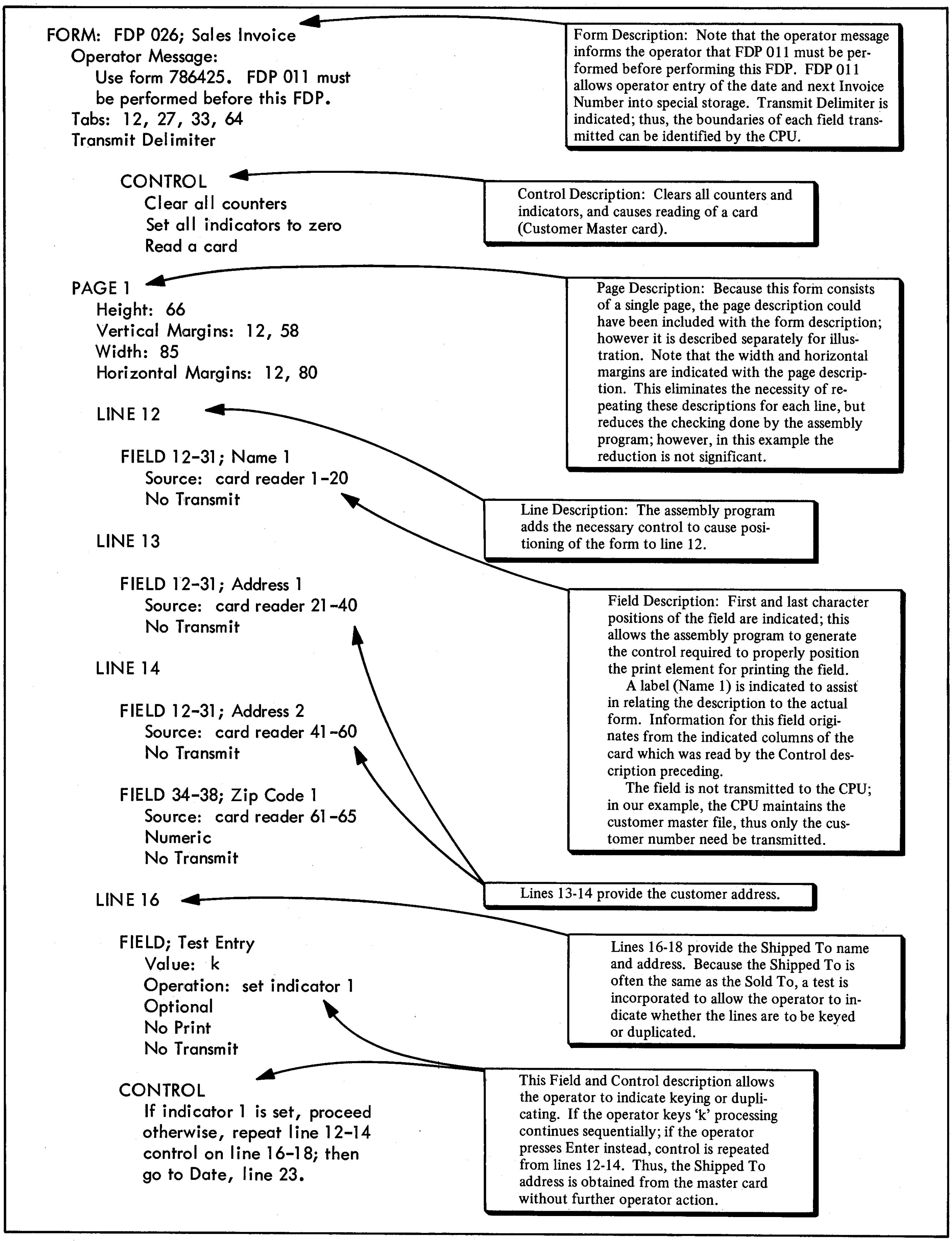
16 12 31
17 12 31
18 12 31 34 38

	Date	Invoice Number	Customer Number	Terr Num	Salesman	Terms	Shipped Via
23	12 19	21 28	30 34	36 38	40 53	55 59	61 80

	Quantity			Unit of Msr	Item No	Description	Price	Amount
	Ord	Shpd	Bk Ord					
28	12 14	16 18	20 22	23 25	27 31	34 62	64 70	72 80

	Gross Amount	Tax %	Tax Amount	Shipping Charge	Invoice Total
58	33 43	45 48	50 58	60 68	70 80

Figure C-4. Annotated Form and Master Card



```
FORM: FDP 026; Sales Invoice
   Operator Message:
      Use form 786425.  FDP 011 must
      be performed before this FDP.
   Tabs: 12, 27, 33, 64
   Transmit Delimiter
```

Form Description: Note that the operator message informs the operator that FDP 011 must be performed before performing this FDP. FDP 011 allows operator entry of the date and next Invoice Number into special storage. Transmit Delimiter is indicated; thus, the boundaries of each field transmitted can be identified by the CPU.

```
         CONTROL
            Clear all counters
            Set all indicators to zero
            Read a card
```

Control Description: Clears all counters and indicators, and causes reading of a card (Customer Master card).

```
   PAGE 1
      Height: 66
      Vertical Margins: 12, 58
      Width: 85
      Horizontal Margins: 12, 80
```

Page Description: Because this form consists of a single page, the page description could have been included with the form description; however it is described separately for illustration. Note that the width and horizontal margins are indicated with the page description. This eliminates the necessity of repeating these descriptions for each line, but reduces the checking done by the assembly program; however, in this example the reduction is not significant.

```
      LINE 12
```

Line Description: The assembly program adds the necessary control to cause positioning of the form to line 12.

```
         FIELD 12-31; Name 1
            Source: card reader 1-20
            No Transmit
```

Field Description: First and last character positions of the field are indicated; this allows the assembly program to generate the control required to properly position the print element for printing the field.

A label (Name 1) is indicated to assist in relating the description to the actual form. Information for this field originates from the indicated columns of the card which was read by the Control description preceding.

The field is not transmitted to the CPU; in our example, the CPU maintains the customer master file, thus only the customer number need be transmitted.

```
      LINE 13

         FIELD 12-31; Address 1
            Source: card reader 21-40
            No Transmit

      LINE 14

         FIELD 12-31; Address 2
            Source: card reader 41-60
            No Transmit

         FIELD 34-38; Zip Code 1
            Source: card reader 61-65
            Numeric
            No Transmit
```

Lines 13-14 provide the customer address.

```
      LINE 16
```

Lines 16-18 provide the Shipped To name and address. Because the Shipped To is often the same as the Sold To, a test is incorporated to allow the operator to indicate whether the lines are to be keyed or duplicated.

```
         FIELD; Test Entry
            Value: k
            Operation: set indicator 1
            Optional
            No Print
            No Transmit

         CONTROL
            If indicator 1 is set, proceed
            otherwise, repeat line 12-14
            control on line 16-18; then
            go to Date, line 23.
```

This Field and Control description allows the operator to indicate keying or duplicating. If the operator keys 'k' processing continues sequentially; if the operator presses Enter instead, control is repeated from lines 12-14. Thus, the Shipped To address is obtained from the master card without further operator action.

Figure C-5. Form Description (Part 1 of 5)

```
  FIELD 12-31; Name 2
     Maximum 20
     No Transmit

LINE 17

  FIELD 12-31; Address 3
     Maximum 20
     No Transmit

LINE 18

  FIELD 12-31; Address 4
     Maximum 20
     No Transmit

  FIELD 34-38; Zip Code 2
     Fixed 5
     Numeric
     Optional
     No Enter
     No Transmit
```

This field shows checking for Numeric Only and Fixed number of characters. If, during document creation, the operator mistakenly keys an alphabetic character, a check-indicator lights, and the character can be rekeyed.

Printing of the field is done after entry and checking are complete, thus all checks are satisfied before the field is printed. Because a zip code may not be available, Optional is indicated. This allows the operator to press Enter and pass the field.

```
LINE 23

  FIELD 12-19; Date
     Source:  storage 1-6
     Numeric
     Edit:  Y9/99/99
     No Transmit

  FIELD 21-28; Invoice Number
     Source:  storage 7-13
     Operation:  add to Invoice counter
     Edit:  99B99999
     Transmit Edit
```

Source data originates from the special storage area. The information was entered by the operator during performance of FDP 011. Output editing provides a space following the first two characters.

The field is transmitted with edit; the CPU can recognize the 'Sales Invoice' transaction by the structure of the invoice number. Other transactions would have identifying numbers structured differently to allow CPU recognition.

```
  FIELD; Increment Invoice Number
     Source:  Emit 1
     Operation:  add to Invoice counter
     No Print
     No Transmit

  FIELD; Store Invoice Number
     Source:  Invoice counter
     Sink:  storage 7-13
     No Print
     No Transmit
```

These descriptions increment the Invoice Number by one, and replace the old number in the special storage area; thus, the Invoice Number is updated for processing the next form.

```
  FIELD 30-34; Customer Number
     Source:  card reader 66-70
     Numeric
```

Figure C-5. Form Description (Part 2 of 5)

```
FIELD 36-38; Territory Number
   Source:  card reader 71-73
   Numeric
   No Transmit

FIELD 40-53; Salesman
   Source:  card reader 74-83
   Alpha
   No Transmit

FIELD 55-59; Terms
   Source:  card reader 84-87
   Numeric
   Edit:  99/99
   No Transmit

FIELD; Test Entry
   Fixed 1
   Value:  a or d or p or r or t
   Operation:
      It a set indicator 2
      If d set indicator 3
      If p set indicator 4
      If r set indicator 5
      If t set indicator 6
   No Print
```

This is another example of operator control of the FDP sequence. By keying the appropriate letter the operator causes branching to a field description which emits the desired Shipped Via identification. The keyed character is transmitted to the CPU to identify the method of shipment.

```
CONTROL; Test Indicators
   If 2 go to Print Air Freight
   If 3 go to Print Deliver
   If 4 go to Print Pick Up
   If 5 go to Print R R Exp
   If 6 go to Print Truck

FIELD 61-80; Print Air Freight
   Source:  Emit Air Freight
   Center
   No Transmit

CONTROL; Go To
   Go to Line 28

FIELD 61-80; Print Deliver
```

Descriptions for print Deliver, print Pick Up, etc., would be identical to that shown for print Air Freight, except that different emit characters would be specified.

```
LINE 28:  Cycle through Amount Field
   28 times; next field Gross Amount.
```

This illustrates cyclic operation. Each line in the body of the form has identical control functions; thus cycling is indicated for the total number of lines in the body of the form. The field to be processed after cycling is also indicated.

During document creation, the operator is guided through the cycled lines until all lines required by the transaction have been completed. The operator then indicates 'End Cycle' and the program skips to the Gross Amount field.

Figure C-5. Form Description (Part 3 of 5)

```
FIELD 12-14; Order Quantity
   Minimum 1, Maximum 3
   Numeric
   Operation: clear Quantity
      Ordered and Backordered
      counters, and add to both
   Right Justify with Blank Fill
```

Keyboard entry of the Order Quantity is checked for Minimum, Maximum and Numeric Only. The entries are added to counters in preparation for calculating Quantity Backordered and Amount. Because this field occurs in a cyclic group, the counters must be cleared to discard information accumulated on the previous line.

```
FIELD 16-18; Shipped Quantity
   Maximum 3
   Numeric
   Operation: subtract from
      Backordered counter
   Optional
   Right Justify with Blank Fill

FIELD 20-22; Backordered
   Source: Backordered counter
   Numeric
   Edit: ZZZ
```

Backordered amount prints from a counter. If the amount is zero (Ordered=Shipped) the edit description suppresses printing of the zeros and the field remains blank.

```
FIELD 23-25; Unit of Measure
   Fixed 2
   Value: ea or dz or gr or lb
   No Enter
   No Transmit

FIELD 27-31; Item Number
   Fixed 5
   Numeric
   Self Check: modulus 10
   No Enter
```

This illustrates a self checking number. If the number is correctly keyed, it prints, and the program/print element advance automatically to the next field (because No Enter is specified).

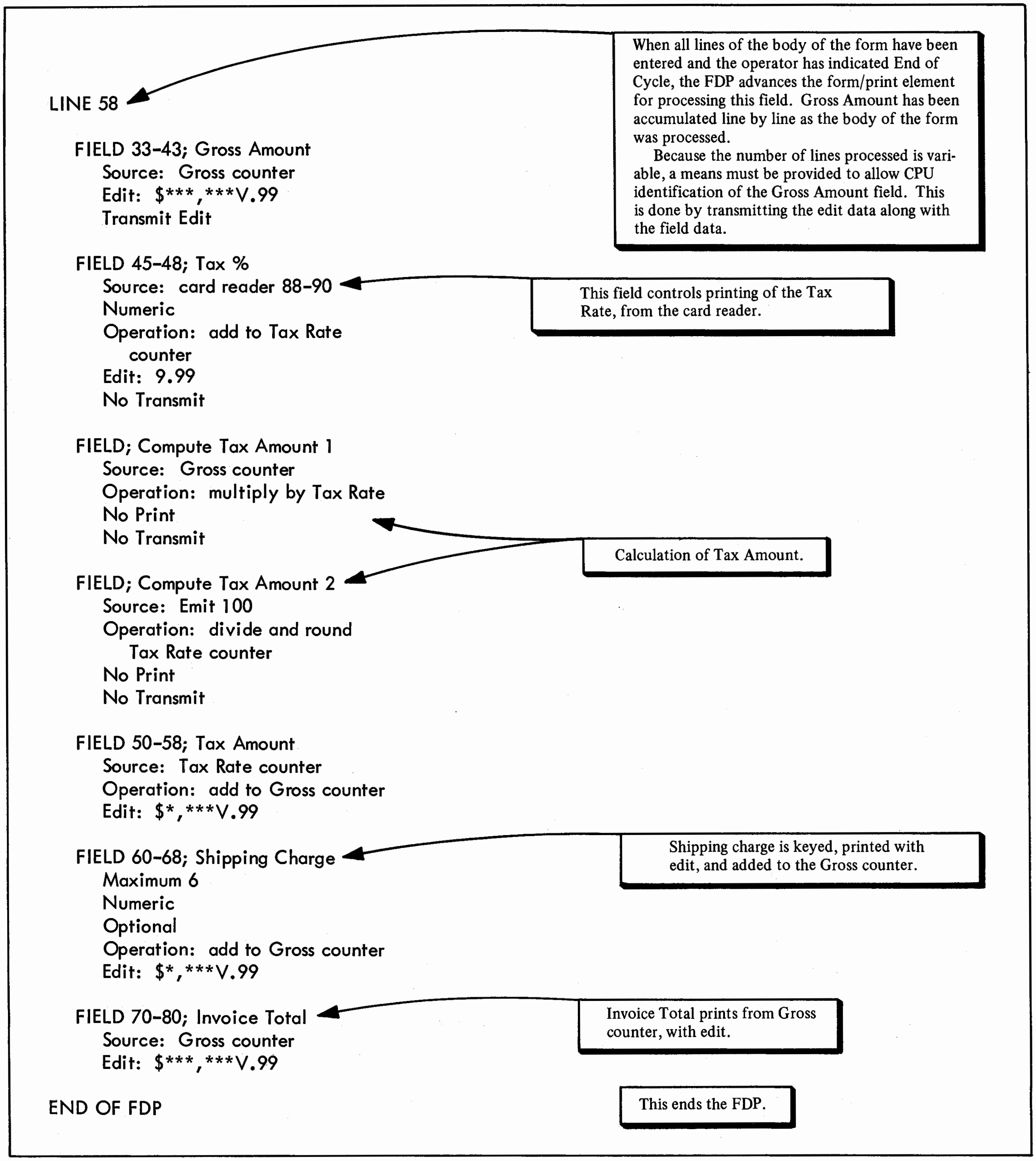
```
FIELD 34-62; Description
   Maximum 29, Minimum 2
   No Transmit

FIELD 64-70; Price
   Maximum 5, Minimum 2
   Numeric
   Operation: multiply by Quantity
      Ordered counter
   Edit: $***V.99
```

This description causes printing of the edited Price (from the keyboard) and multiplying by the Quantity Ordered to obtain Amount.

```
FIELD 72-80; Amount
   Source: Quantity Ordered counter
   Operation: add to Gross counter
   Edit: $*,***V.99
```

Amount is printed from the Quantity Ordered counter, with editing, and added to the Gross counter.

Figure C-5. Form Description (Part 4 of 5)



```
LINE 58

   FIELD 33-43; Gross Amount
      Source:  Gross counter
      Edit:  $***,***V.99
      Transmit Edit
```

When all lines of the body of the form have been entered and the operator has indicated End of Cycle, the FDP advances the form/print element for processing this field. Gross Amount has been accumulated line by line as the body of the form was processed.

Because the number of lines processed is variable, a means must be provided to allow CPU identification of the Gross Amount field. This is done by transmitting the edit data along with the field data.

```
   FIELD 45-48; Tax %
      Source:  card reader 88-90
      Numeric
      Operation:  add to Tax Rate
         counter
      Edit:  9.99
      No Transmit
```

This field controls printing of the Tax Rate, from the card reader.

```
   FIELD; Compute Tax Amount 1
      Source:  Gross counter
      Operation:  multiply by Tax Rate
      No Print
      No Transmit

   FIELD; Compute Tax Amount 2
      Source:  Emit 100
      Operation:  divide and round
         Tax Rate counter
      No Print
      No Transmit
```

Calculation of Tax Amount.

```
   FIELD 50-58; Tax Amount
      Source:  Tax Rate counter
      Operation:  add to Gross counter
      Edit:  $*,***V.99

   FIELD 60-68; Shipping Charge
      Maximum 6
      Numeric
      Optional
      Operation:  add to Gross counter
      Edit:  $*,***V.99
```

Shipping charge is keyed, printed with edit, and added to the Gross counter.

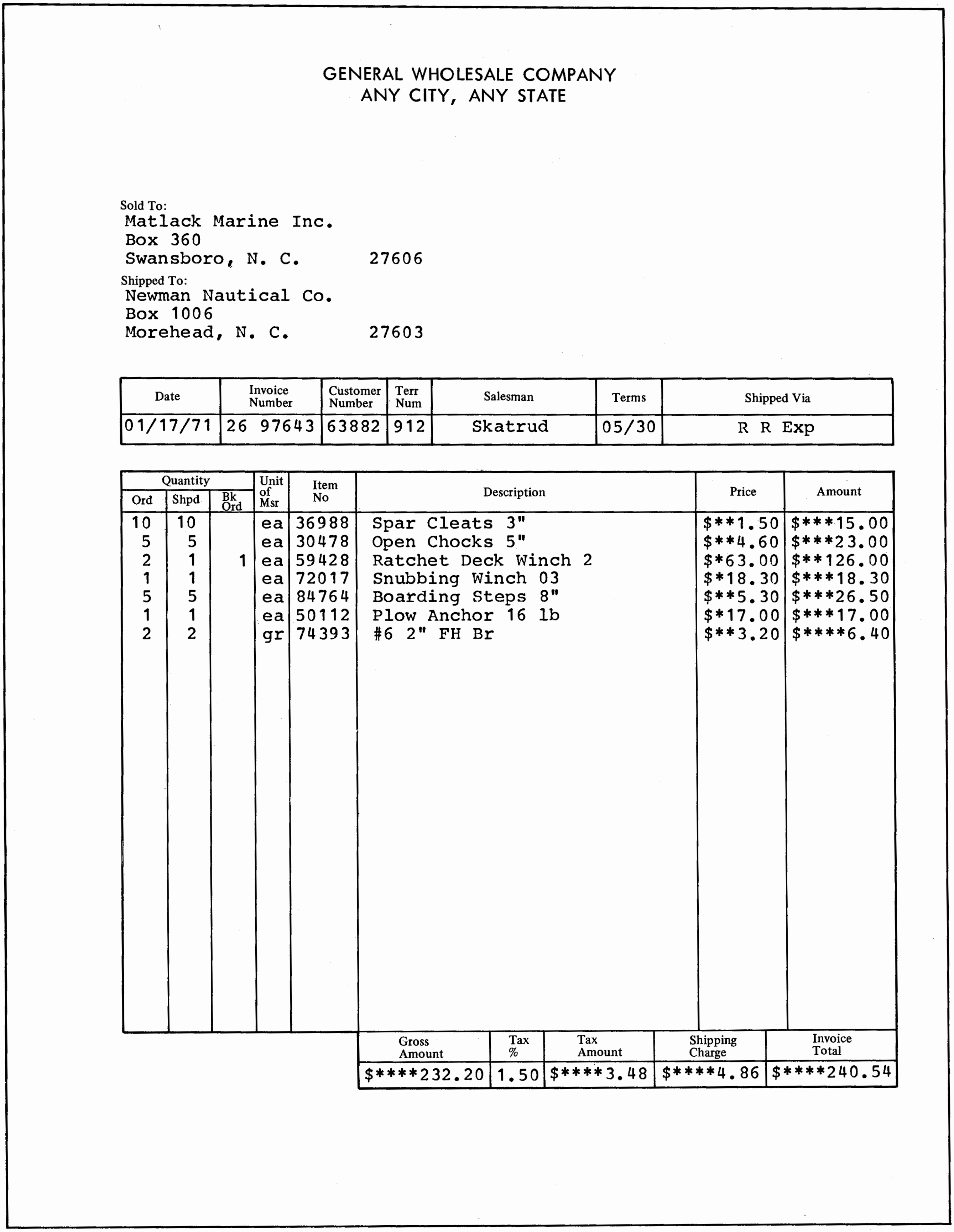
```
   FIELD 70-80; Invoice Total
      Source:  Gross counter
      Edit:  $***,***V.99
```

Invoice Total prints from Gross counter, with edit.

```
END OF FDP
```

This ends the FDP.

Figure C-5. Form Description (Part 5 of 5)



GENERAL WHOLESALE COMPANY
ANY CITY, ANY STATE

Sold To:
Matlack Marine Inc.
Box 360
Swansboro, N. C.   27606

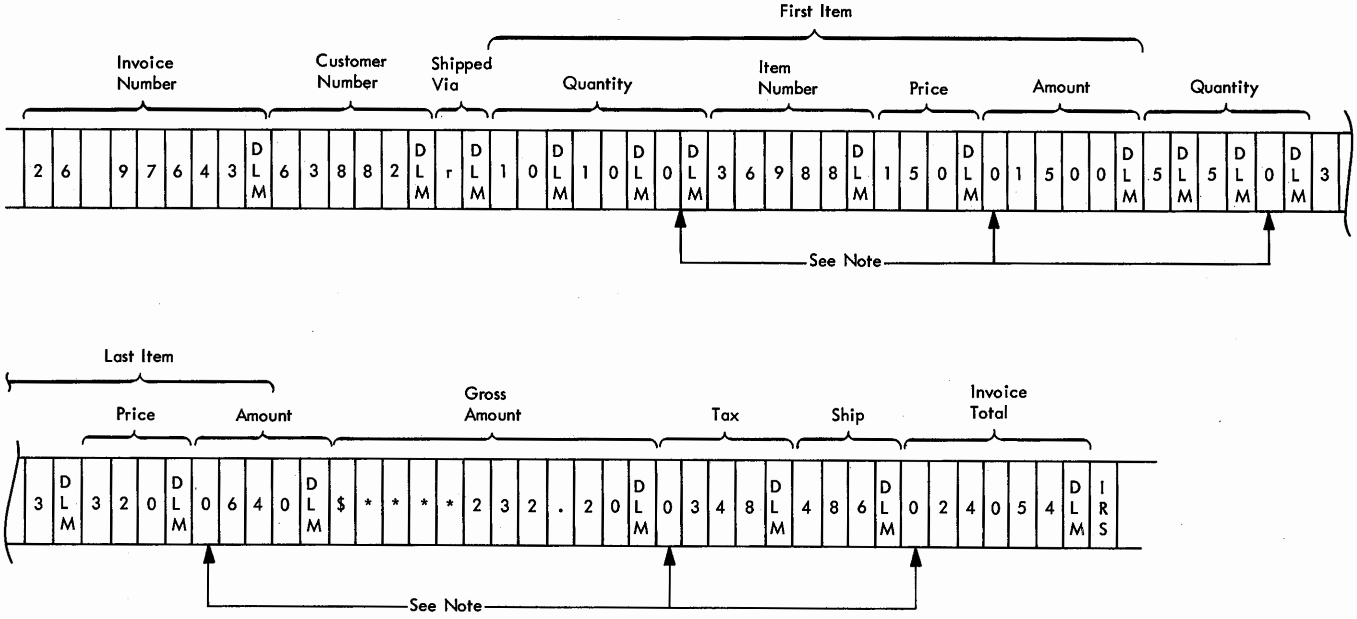
Shipped To:
Newman Nautical Co.
Box 1006
Morehead, N. C.   27603

Date	Invoice Number	Customer Number	Terr Num	Salesman	Terms	Shipped Via
01/17/71	26 97643	63882	912	Skatrud	05/30	R R Exp

Quantity			Unit of Msr	Item No	Description	Price	Amount
Ord	Shpd	Bk Ord					
10	10		ea	36988	Spar Cleats 3"	$**1.50	$***15.00
5	5		ea	30478	Open Chocks 5"	$**4.60	$***23.00
2	1	1	ea	59428	Ratchet Deck Winch 2	$*63.00	$**126.00
1	1		ea	72017	Snubbing Winch 03	$*18.30	$***18.30
5	5		ea	84764	Boarding Steps 8"	$**5.30	$***26.50
1	1		ea	50112	Plow Anchor 16 lb	$*17.00	$***17.00
2	2		gr	74393	#6 2" FH Br	$**3.20	$****6.40

Gross Amount	Tax %	Tax Amount	Shipping Charge	Invoice Total
$****232.20	1.50	$****3.48	$****4.86	$****240.54

Figure C-6. Completed Form


First Item
Invoice Number
Customer Number
Shipped Via
Quantity
Item Number
Price
Amount
Quantity
2 6 9 7 6 4 3 DLM 6 3 8 8 2 DLM r DLM 1 0 DLM 1 0 DLM 0 DLM 3 6 9 8 8 DLM 1 5 0 DLM 0 1 5 0 0 DLM 5 DLM 5 DLM 0 DLM 3
See Note
Last Item
Price
Amount
Gross Amount
Tax
Ship
Invoice Total
3 DLM 3 2 0 DLM 0 6 4 0 DLM $ * * * * 2 3 2 . 2 0 DLM 0 3 4 8 DLM 4 8 6 DLM 0 2 4 0 5 4 DLM IRS
See Note

Note:

All transmitted fields originating from a counter which are not specified for Transmit Edit are transmitted as 10 digit fields; thus, the fields indicated would be filled with left zeros to increase their size to 10 digits. For the sake of brevity, these left zeros have not been shown.

Figure C-7. Form Record Transmission Format

Glossary

Form Description Program (FDP): A computer-generated program which, when resident in the 3735 and selected by the terminal operator, activates subroutines of the terminal control program to perform processing of a specific form type or application.

Form Description Macros (FDM): Macro statements which, when assembled by a computer operating system, produce a Form Description Program.

Form Description Utility (FDU): A program which prepares the output of the form description macro assembly for selection by the customer application program and transmission to the 3735.

Terminal Control Program (TCP): A terminal-resident program which controls terminal operation. Subroutines of this program are selected by analysis of the FDP, or from the terminal keyboard, and control terminal operation during performance of the associated function. Subroutines of the TCP also provide communications line control and data management functions required for terminal operation.

Index

Note: An italicized page reference indicates that the item referred to appears in a Figure on that page.

applications
  batch transmission 11
  combined operations 11
  example 31
  inquiry 11
  source recording 11
Arithmetic and Logic Unit 13
arithmetic operations 21, *22*
assembly of FDMs 9, 29

batch
  totals 11, *21*
  transmission 11
binary synchronous communications
  adapter 13
  compatibility 10
buffers
  dynamic 13
  inquiry 13
  reader/punch 13

capacity, storage
  minimum, CPU 12
  3735 10
character spacing 13, 19
checking functions
  document creation *21*
  examples *36-40*
  FDM assembly 29
  utility 29
combined operations 11
communicate mode 15
communications data-link control 10, 13
communications facilities 18
compare functions *22, 36-40*
conditioning, data channels 18
configuration 10
controls, keyboard *16*
control
  cyclic 20
  program 9, 13
  unit 13, *14*
  3735 9
counters 13
CPU ID 12
CPU/3735 configurations *10*
creating a document 15
cyclic control 20

data
  capture 7
  channel conditioning 18
  directory 13
  erase 15, 17
  protection 12
  reception 13
  security 12
data set interface 18
design, form 19
diagnostic tests *12,* 14
diagnostic test selection 17
diagnostics, utility 29
dial-up communications 15
disk 10, 13
document creation 7, 15
dynamic buffers 13

editing functions *23, 37-40*
EIA standard 18
enter form mode 15
erase data
  from CPU 15
  operator created 17
error correct mode 15
error recovery, FDM implementation errors 29
example, form description 31

features
  basic 10
  special 10
  specify 10
field description
  examples *36-40*
  operation 21
  sink 21
  source 20
fixed-length forms 15
format
  keyboard 14, *15*
  output data 22
  transmission 22, 23
form description
  control 22
  development of 19
  example *36*
  form 19
  field 20
  line 20
  page 20
form description macros
  assembly 11, 29
  purpose 9
Form Description Program
  concept 9
  identification 14, 19
  generation 11
  selection 14
Form Description Utility 11, 29
form design 19
forms stacker 18
functions
  arithmetic 21, *22*
  checking 20, *21*
  compare 21, *22*
  edit 22, *23*
  justify 21, *22*
  self-check *22*

horizontal margins 20

identification
  CPU 12
  example in form description *36-40*
  FDP 14, 19
  line 20
  page 20
  terminal 12
implementation considerations 19
indicators
  keyboard 17
  logic 13
inquiry

  - application 11
  - buffer 13
  - operating procedure 15
- interface, data set 18

- justify 21, *22*

- keyboard 14, *15*
- keylock 12, 18

- line identification 20, *36*
- listings
  - resident FDPs 17
  - resident data records 17
  - utility diagnostic 29
- local mode 14
- logic indicators 13
- logic unit 13

- macros, form description
  - assembly 11, 29
  - purpose 9
- magnetic disk 10, 13
- minimum system (CPU) 12
- modems 18
- modules, object 29
- multipoint communications 18

- object modules 29
- operations 20-21
- operating
  - characteristics 13
  - procedures 14
- operating modes
  - communicate 15
  - enter form 15
  - local mode 14
  - error correct 15
  - playback 15
  - request 15
- operator
  - guidance 9, 14
  - message 19

- page description 20, *36*
- page identification 20
- pinfeed platens 18
- playback mode 15
- printer 13
- printer ribbon 13
- problem determination 12
- program
  - application 9, 29
  - FDU 11, 29
  - FDM 11, 29
  - support 11
  - TCP 9, 13
- programming considerations 29
- power type 15
- punch buffer 7, 15

- restrictions
  - communications facilities 18
  - FDP loading 30
  - forms 19
  - message blocks 24, *25*
  - system capacity 12, 29
  - transmission sequence 30
- reader/punch buffer 13

- REPLACE option 29
- request mode 15
- ribbon, printer 13

- selecting an FDP 14
- Selectric I/O II 13
- self-check *17, 21*
- sink 21, *36-40*
- source
  - data 20
  - documents 7
  - examples *36-40*
- source recording
  - application 11
  - example 31
  - operating procedure 15
- spacing, character 13, 19
- special features
  - Forms Stacker 18
  - Keylock 18
  - Multipoint Communications 17
  - Pinfeed Platens 18
  - Synchronous Clock 18
  - 5496 Adapter 18
- special storage 13
- specify features 10
- speed
  - printing *10*
  - transmission *10,* 18
- stacker, forms 18
- standard, EIA 18
- storage capacity
  - CPU 12
  - 3735 10
- switched network 18
- synchronous clock 18
- synchronous data sets 18

- tab set
  - description 20, *36*
  - routine 15
- Terminal Control Program
  - purpose 9
  - where stored 13
- terminal ID 12
- tests, diagnostic
  - listing *12*
  - selection *12,* 17
  - utility 29
- transmission formats
  - blocks 24, *25*
  - example record *42*
  - fields 20, 22
  - records 24
- transmission of FDPs 30
- trouble record form 12

- UPSI statement 29
- utility program 11, 29

- variable-length forms 15
- vertical margins 20

- width, page 20

- 5496
  - adapter 18
  - application 11
  - example 31

READER'S COMMENT FORM

IBM 3735 Programmable Buffered Terminal
Concept and Application

Order No. GA27-0343-0

- How did you use this publication?

As a reference source	□
As a classroom text	□
As a self-study text	□

- Based on your own experience, rate this publication . . .

As a reference source:	Very Good	Good	Fair	Poor	Very Poor
As a text:	Very Good	Good	Fair	Poor	Very Poor

- What is your occupation?

- We would appreciate your other comments; please give specific page and line references where appropriate. If you wish a reply, be sure to include your name and address.

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

YOUR COMMENTS, PLEASE . . .

This publication is one of a series that serves as a reference source for systems analysts, programmers, and operators of IBM systems. Your answers to the questions on the back of this form, together with your comments, help us produce better publications for your use. Each reply is carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Please note: Requests for copies of publications and for assistance in using your IBM system should be directed to your IBM representative or to the IBM sales office serving your locality.

Cut Along Line

Fold Fold

FIRST CLASS
PERMIT NO. 569
RESEARCH TRIANGLE PARK
NORTH CAROLINA

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

POSTAGE WILL BE PAID BY . . .

IBM Corporation
P. O. Box 12275
Research Triangle Park
North Carolina 27709

Attention: Publications Center, Dept. E01

Fold Fold

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

3735

3735 (Basic)	$330.—	
2 Storage Exp @	30.—	ea.
multi-pt.	15.—	if using on leased line.
Printer (3286) adapter	15	
5496 Adapter	25.—	
3286 Mod. 3	165.—	
5496	155.—	
3735 attachment for 5496	45.—	

145 K, multi-pt., max system comes out to $810.—

Big selling points may be error correction & Logic capability

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)