iUP-200A/201A UNIVERSAL PROGRAMMER USER'S GUIDE

Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

Order Number: 164852-001

CAUTION

This equipment generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. It has been tested and found to comply with the limits for a Class A Computing Device pursuant to Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

The information in this document is subject to change without notice.

Intel Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this document. Intel Corporation makes no commitment to update nor to keep current the information contained in this document.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication or disclosure is subject to restrictions stated in Intel's software license, or as defined in ASPR 7-104.9(a)(9).

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local sales office to obtain the latest specifications before placing your order.

The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products:

BITBUS	iLBX	iPDS	Plug-A-Bubble
COMMputer	im	iRMX	PROMPT
CREDIT	iMMX	iSBC	Promware
Data Pipeline	Insite	iSBX	QUEX
GENIUS	Intel	iSDM	QUEST
Δ i	intel	iSXM	Ripplemode
i	intelBOS	Library Manager	RMX/80
I^2ICE	Intelevision	MCS	RUPI
ICE	inteligent Identifier	Megachassis	Seamless
iCS	inteligent Programming	MICROMAINFRAME	SOLO
iDBP	Intellec	MULTIBUS	SYSTEM 2000
iDIS	Intellink	MULTICHANNEL	UPI
	iOSP	MULTIMODULE	

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

*MULTIBUS is a patented Intel bus.

REV.	REVISION HISTORY	DATE
-001	Original Issue	3/84

About This Manual

This manual describes the Intel iUP-200A/201A universal programmer and how to use the 201A model in off-line mode to store programs and data into programmable read-only memory (PROM) devices. (The *iPPS PROM Programming Software User's Guide* describes how to use the universal programmer in on-line mode.) It is for engineers and designers who are developing firmware for ROM-, PROM-, EPROM-, or E^2PROM-based systems.

The following paragraphs describe the general contents of each chapter.

Chapter 1	Contains a general description of the universal programmer and a tutorial overview of PROM programming and firmware development.
Chapter 2	Describes how to set up and initialize the universal programmer system.
Chapter 3	Covers the off-line operation of the universal programmer and describes the front panel operation of the iUP-201A.
Chapter 4	Provides a number of off-line programming examples that illustrate using the universal programmer in typical firmware development applications.
Appendix A	Describes error conditions and error messages.
Appendix B	Contains host serial command protocol for connecting the universal programmer to a non-Intel host.
Appendix C	Contains reference tables for numeric conversion and ASCII codes.
Appendix D	Contains a schematic for building an RS-232 to 20ma converter.

Conventions Used In This Book

Throughout this book, the iUP-200A/201A is called the universal programmer.

The following symbol conventions are used in this manual to document notes, cautions, and warnings:

A section of text introduced by the symbol

NOTE

emphasizes comments with special significance.

A section of text introduced by the symbol

CAUTION

gives instructions necessary to avoid damage to equipment or loss of stored information.

A section of text introduced by the symbol

WARNING

gives instructions necessary for personal safety.

Other Pertinent Intel Literature

While this guide is a self-contained document describing the off-line use of the universal programmer, several other Intel documents are related to firmware development and to the development of microprocessor-based systems as follows:

- *iUP-XX Personality Module User's Guide,* order number varies

  Each Intel personality module has its own user's guide describing its installation, applicable devices, applicable hosts, and unique characteristics. Consult the *Microcontroller Handbook*, the *Microprocessor and Peripheral Handbook*, and the *Memory Components Handbook* to determine which personality module to use for specific PROM devices. Intel supplies the appropriate personality module user's guide with each personality module. You can order additional copies from the Intel Literature Department. Refer to page ii for the address.

- *iPPS PROM Programming Software/iUP-200A/201A Universal Programmer Pocket Reference,* order number 164853

  This pocket reference describes the iPPS software and universal programmer off-line operation and command syntax.

- *iPPS PROM Programming Software User's Guide,* order number 164861

  This manual describes how to use the universal programmer in on-line mode.

- *ISIS-XX User's Guide*, order number varies

  This manual is a comprehensive guide to the Intel operating system under which the iPPS runs.

- *Microcontroller Handbook*, order number 210918

  This handbook contains all application notes, article reprints, data sheets, and other user information on the MCS®-48, MCS-51 (8-bit), and the MCS-96 (16-bit) product families.

- *Microprocessor and Peripheral Handbook*, order number 210844

  This handbook contains data sheets on all microprocessors and peripherals.

- *Development Systems Handbook*, order number 210940

This handbook contains data sheets on development systems and supporting software.

- *Memory Components Handbook*, order number 210830

  This handbook contains all application notes, article reprints, data sheets, and other design information on RAMs, DRAMs, EPROMs, E^2PROMs, and bubble memories.

SERVICE INFORMATION

Intel Customer Engineers provide the best service for your Intel products. These trained professionals provide the prompt, efficient on-site installation, preventive maintenance, and corrective maintenance services required to keep your equipment in the best operating condition.

The Intel Customer Engineer provides the service needed through a prepaid service contract or on an hourly charge basis. For further information, contact your local Intel sales office.

United States customers can obtain Intel service and repair assistance by contacting the Intel Product Service Center in their area. Customers outside the United States should contact their sales source (Intel Sales Office or Authorized Distributor) for service information and repair assistance.

Before calling the Product Service Center, have the following information available:

- The date you received the product.
- The complete part number of the product (including dash number). On boards, this number is usually silk-screened onto the board. On other products, it is usually stamped on a label.
- The serial number of the product. On boards, this number is usually stamped on the board. On other products, the serial number is usually stamped on a label mounted on the outside of the chassis.
- The shipping and billing address.
- If the Intel product warranty has expired, a purchase order number for billing.
- Advise the Product Service Center personnel of any applicable extended warranty agreements.

Use the following telephone numbers to contact the Intel Product Service Center:

Western Region:	(602) 869-4951
Midwest Region:	(602) 869-4392
Eastern Region:	(602) 869-4045
International:	(602) 869-4391
TWX:	(910) 952-1330

Always contact the Product Service Center before returning a product to Intel for repair. Product Service Center personnel will give you a repair authorization number, shipping instructions, and other information to ensure fast, efficient service. If you are returning a product damaged during shipment or if the product is out of warranty, you must have a purchase order before Intel can start the repair.

If available, use the original factory packaging material to ship the product to the Intel Product Service Center. If the original packaging material is not available, wrap the product in a cushioning material such as Air Cap SD-240, manufactured by the Sealed Air Corporation, Hawthorne, N.J. Securely enclose it in a heavy-duty corrugated shipping carton, mark it "FRAGILE," and ship it to the address specified by the Intel Product Service Center.

WARNINGS AND CAUTIONS

This section lists the warnings and cautions found in the manual, along with the pages on which they appear.

PAGE

CAUTION — 2-2

To prevent possible damage to the universal programmer, verify that the line voltage select switches are set properly and the power fuse is the right value for the line voltage that is to be used before plugging in the unit and turning it on. Refer to the Setting Line Voltage and Checking and Replacing Power Fuse sections in this chapter.

WARNING — 2-2

Only a qualified technical person should change the line voltage setting.

WARNING — 2-3

Only a qualified technical person should check or replace the power fuse.

CAUTION — 2-7

The universal programmer requires a minimum of three inches of clearance for proper cooling, and its air vents must be clear of any obstructions.

CAUTION — 2-7

To prevent damage to the universal programmer, make sure that the line voltage select switches are set properly and the power fuse is the right value for the line voltage that is to be used before plugging in the unit and turning it on. Refer to the Setting Line Voltage and Checking and Replacing Power Fuse sections in this chapter.

To prevent damage or accidental programming of a PROM device, remove any PROMs from the personality module before powering on the universal programmer.

A device(s) should not be in the PROM socket(s) when the power is turned on or off.

CAUTION 2-9

Do not switch the universal programmer's power on or off when a PROM device is installed in a socket of the personality module. Damage to the PROM device can result.

CAUTION 2-9

If the personality module has more than one socket, install the PROM to be programmed or read only in the selected socket. Do not install a PROM or leave one in an unselected socket. Damage to the PROM can result.

CAUTION 2-10

The orientation mark on one end of the PROM must be toward the top of the socket. If a PROM is not oriented properly, it cannot be programmed and may be damaged. If the PROM is not oriented properly, the following error message may be displayed when the universal programmer attempts to access the PROM:

CHECK PROM INSTALLATION

CAUTION 4-3

Be sure that the type of PROM that you are installing is the same as the type selected with the DEVICE SELECT key. If you do not specify its type correctly, you can damage a PROM when you try to program it or read it.

CONTENTS

FIGURES

TABLES

CHAPTER 1
GENERAL INFORMATION

This chapter contains a general description of the iUP-200A/201A universal programmer, an overview of firmware development, and a description of system configurations.

Overview of the Universal Programer

The Intel universal programmer programs and verifies Intel programmable read only memory (PROM) components and other Intel integrated circuits that contain PROMs. It can be used to program non-erasable bi-polar PROMs, erasable PROMs (EPROMs), and electrically erasable PROMs (E^2PROMs). It can also read ROM (read only memory) in certain ROM-based components.

The two models of the universal programmer are: the iUP-200A universal programmer, which operates on-line as a peripheral of an Intel Intellec® microprocessor development system, and the iUP-201A universal programmer, which can be operated either on-line or off-line.

The universal programmer programs the various PROM devices using personality modules. Personality modules are small units that plug into the front of the universal programmer. They personalize the universal programmer for a specific PROM or family of PROM devices. Figure 1-1 shows the iUP-200A universal programmer with a personality module.

In on-line mode, the Intel PROM programming software (iPPS) controls the universal programmer. The iPPS software is a utility that runs under the ISIS operating system. It controls the operation of the universal programmer through a serial I/O port on the development system. The iPPS software provides a comprehensive set of commands that allow the following operations:

- Reading and writing data to and from disk files.
- Manipulating data in the Intellec memory buffer.
- Mapping data for a particular PROM word size.
- Interleaving data for different addressing schemes.
- Programming a PROM device.
- Reading the contents of the PROM device.
- Verifying the contents of the PROM device.
- Locking selected PROMs from user access.

The *iPPS PROM Programming Software User's Guide* describes on-line operation and the Intel PROM programming software.

You can use the iUP-201A universal programmer as a stand-alone programming system in off-line mode. You can perform four types of off-line operations as follows:

- Duplicating and verifying PROMs.

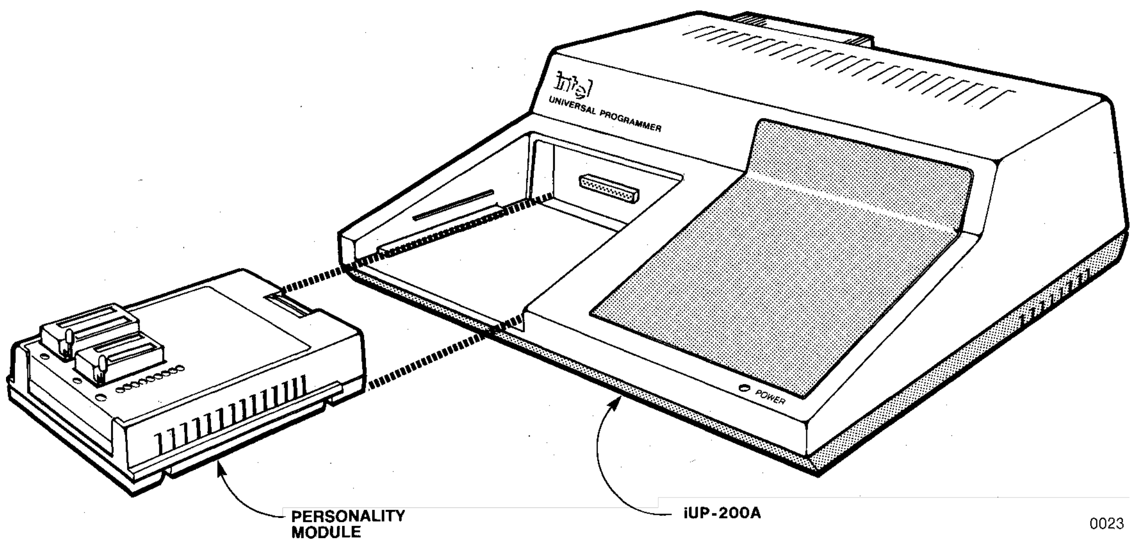


Figure 1-1 iUP-200A Universal Programmer with Personality Module

- Locking selected PROMs from user access.
- Down-loading data from any host system that has an RS-232 interface and programming a PROM with that data.
- Displaying and editing data in local random access memory (RAM).

These off-line operations are performed independently of any host computer system. (Chapters 3 and 4 describe off-line operation.)

You can expand an iUP-200A universal programmer to iUP-201A universal programmer capability by installing the off-line option. You can install this option yourself. The off-line option consists of the following:

- A keyboard with a numeric (hex) keypad and various function keys.
- An alphanumeric display.
- A circuit board that contains control circuitry and 32K bytes of local RAM. Throughout this manual, the local RAM is referred to as URAM (for user RAM).

General Description

The universal programmer is an intelligent peripheral for an Intellec development system. The universal programmer is connected to a host computer by a two-way serial interface. (Refer to the Compatible On-Line Hosts section in Chapter 2 for further information on connecting the universal programmer to a host.) A dedicated Intel 8085 microprocessor with its own RAM and firmware allows the universal programmer to perform many operations independently of the host.

The universal programmer is housed in a low-profile case with an angled front panel. The ON/OFF switch is located on the back panel, and a POWER ON indicator light is located on the front panel.

Air convection through vents at the rear, top, and bottom of the case cool the unit. The left side of the front panel can hold a personality module. Figure 1-1 shows the basic iUP-200A universal programmer with an uninstalled personality module.

The iUP-201A universal programmer is identical to the iUP-200A universal programmer except that it has the off-line option installed. Figure 1-2 shows an iUP-201A universal programmer with a personality module installed.

The hand-sized personality modules plug into the universal programmer. Each personality module has one or more device sockets which hold PROMs for programming. LEDs on the personality module indicate the PROM type selected and the socket to use.

The off-line option of the iUP-201A universal programmer occupies the right half of the front panel and consists of an alphanumeric display and a keyboard.

The alphanumeric display has labeled fields for command, address, and data. The keyboard section contains a numeric (hex) keypad and function keys. The off-line option also adds 32K of user-accessible RAM (URAM).

The back panel of the universal programmer (shown in Figure 1-3) is the same for both the iUP-200A universal programmer and the iUP-201A universal programmer. The main power switch is a rocker switch located in the upper left

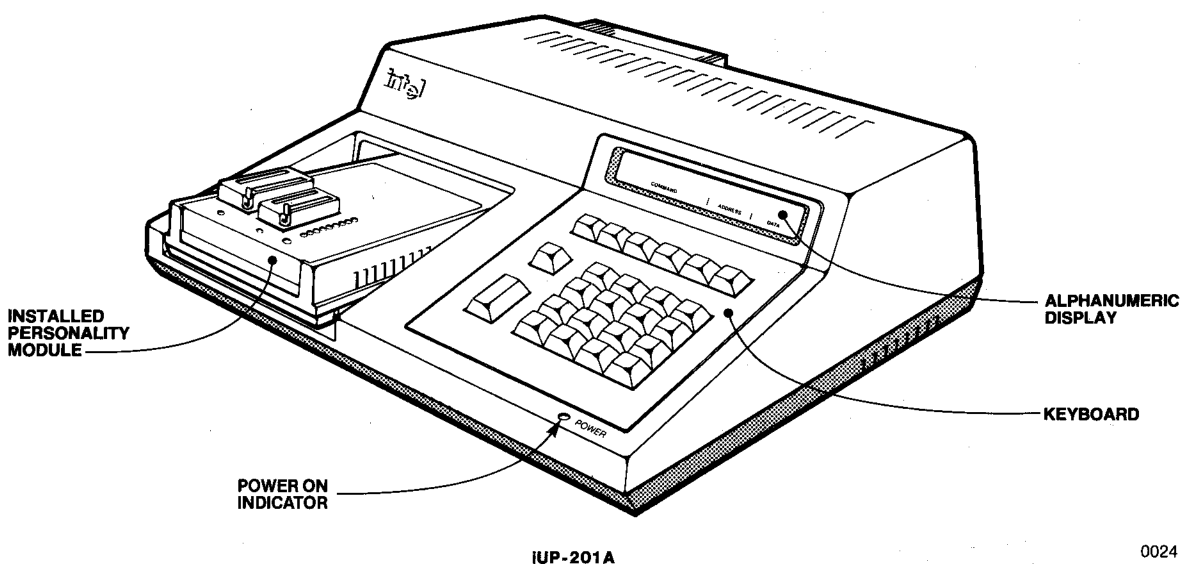


Figure 1-2 iUP-201A Universal Programmer with an Installed Personality Module

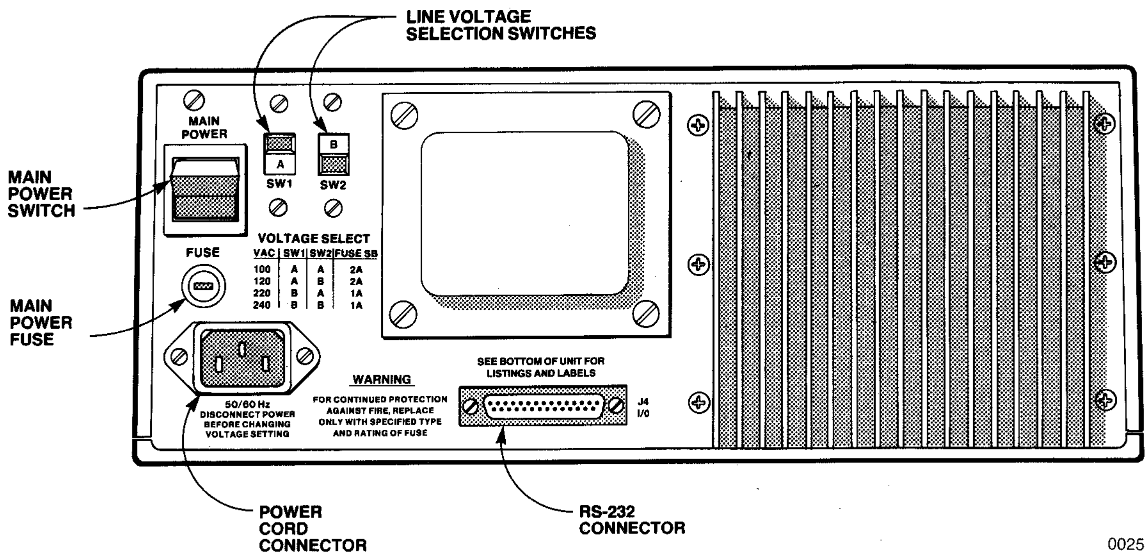


Figure 1-3 Universal Programmer Back Panel

corner when facing the back panel. To its right are two line voltage selection switches. The main power fuse is located below the power switch. The AC power connector is in the lower left corner. An RS-232 connector is at the bottom center.

The various safety listing approvals and the serial number are located on the bottom of the universal programmer.

Physical Specifications

	Depth	Width	Height
Dimensions: iUP-200A/201A:	15″	15″	6″
Personality Module:	7″	5.5″	1.6″

Weight: iUP-200A/201A: 15 pounds maximum
Personality Module: 1 pound maximum

Reading Temperature Range: 10to 40 degrees C.
Programming Temperature Range: 25 ±5 degrees C.
Operating Humidity Range: 10% to 85% RH (non-condensing)

Electrical Specifications

Operating Voltage: Selectable 100, 120, 220, or 240 VAC (each within ±10%) at 47-63 Hz single phase grounded.

Power Consumption: iUP-200A/201A, maximum 120 watts

Fuse Protection:	100 VAC, use 2 ampere slow blow 120 VAC, use 2 ampere slow blow 220 VAC, use 1 ampere slow blow 240 VAC, use 1 ampere slow blow

Firmware Development

Microprocessors and microcomputers are central elements in designing electronic products. The typical development process for microprocessor- and microcomputer-based systems involves integrating hardware and software. Once the software is perfected, it is often installed permanently in a read only memory (ROM) device. Software installed in ROM is referred to as firmware. Firmware is used in systems design because of its relative low cost, high speed, and data non-volatility. Non-volatility means that firmware is retained even when system power is turned off.

Intel manufactures a wide variety of ROM memory components which can be divided into two general types: masked ROMs and electrically programmable ROMs (PROMs).

Intel fabricates firmware into a masked ROM during manufacturing. This is often the most cost effective method for including firmware in a mass-produced product.

Programs and data are electrically programmed into a PROM. To electrically program a PROM, data is presented to a particular address. Voltage is then applied to programming pins to set (burn in) the code in the selected cell.

Bipolar PROMs can be electrically programmed only once; thereafter, they retain their data permanently. The programs or data programmed into an EPROM can be erased by exposing the device to ultraviolet light. An E^2PROM can be electrically erased in a manner similar to that used to program it.

EPROMs and E^2PROMs allow flexibility during firmware development because they can be erased and reprogrammed. Some microcomputer components, such as the Intel 8751, have built-in EPROMs in addition to other logic.

For design convenience, some microcomputer components have pin-compatible counterparts containing masked ROM instead of EPROM. This allows the reprogrammable EPROM version to be used during prototype development and the masked ROM version to be used for the final mass production.

PROM Programming Overview

All PROMs have a characteristic physical word length. This word length is the number of parallel bits that are accessed when a given address is specified. Almost all of Intel's recent PROM components have an 8-bit word; however, many earlier PROMs had 4-bit words.

The difference between a PROM's physical word length and the logical word length used in a system must be considered when programming the PROM. For example, two 8-bit PROM devices must be connected in parallel to provide the 16-bit-wide memory word that the 8086 microprocessor requires. To put an 8086 machine language program (consisting of a series of 16-bit words) into 8-bit PROM devices, the machine code must be formatted so that the 16-bit words are mapped correctly into each pair of 8-bit devices. For example, the first PROM might be programmed with the upper eight bits of each machine code word, while the second PROM might be programmed with the lower eight bits of each machine code word.

The iPPS software provides the software tools to perform automatic interleave mapping of data and machine code into Intel PROMs. (The *iPPS PROM Programming Software User's Guide* describes iPPS operation.)

System Configurations

To program PROMs with the iUP-200A/201A universal programmer in on-line mode, the iPPS software must be running on the Intellec development system. In addition, the universal programmer must be powered on and connected to the Intellec development system through an RS-232 configured cable. If you are using the iUP-201A model, you must select on-line mode with the ON LINE switch (refer to Chapter 3).

Figure 1-4 illustrates the system data flow in on-line mode. The four major system devices (PROM, buffer, file, and URAM) are shown with arrows indicating the directions of data flow between these devices.

The host development system contains the buffer and file devices, while the universal programmer contains the PROM and URAM devices. The URAM device is only available in the iUP-201A universal programmer. The PROM device is shown as part of a personality module.

The PROM device is the PROM component that is plugged into a socket on the personality module.

The buffer device provides a temporary area where the iPPS software stores and manipulates data.

The file device is an ISIS file that contains programs or data.

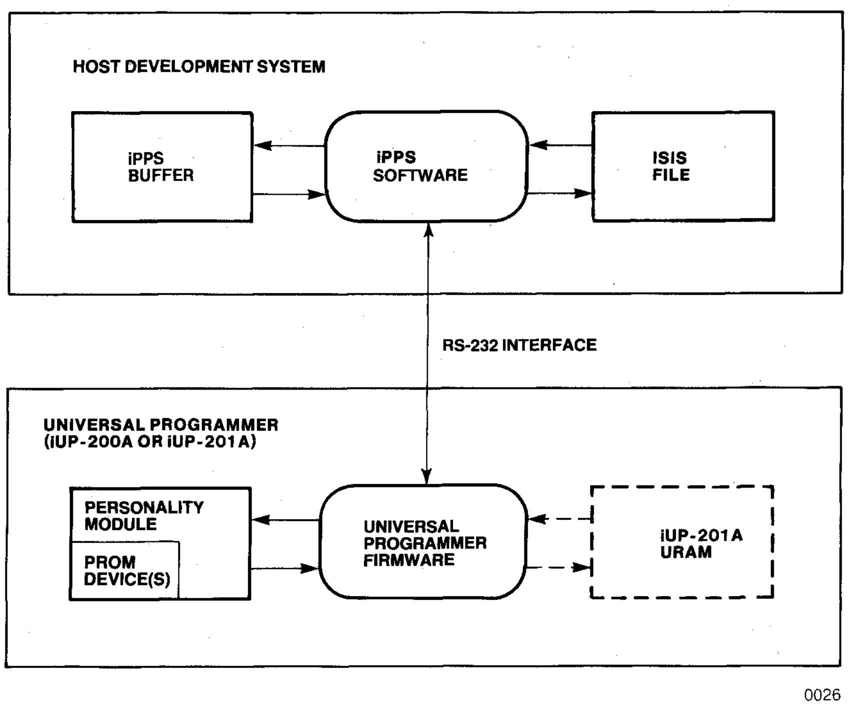


Figure 1-4 On-line System Data Flow

The URAM device is user RAM that is physically located in the iUP-201A universal programmer. This RAM is generally used for off-line data manipulation but is accessible for certain on-line operations.

The iPPS software and the iUP-200A/201A firmware (which run on the host development system and the universal programmer, respectively) handle all data transfers between the four logic devices. The two programs communicate by a serial interface command protocol. Refer to Appendix B for more information on this protocol.

Figure 1-5 illustrates the system data flow for the universal programmer in off-line configuration. In this configuration, only the PROM and URAM devices are available.

Through front panel controls (on the iUP-201A universal programmer), you direct the universal programmer to modify the contents of the local URAM, to program URAM data into the PROM device, to load the URAM with the data in the PROM device, or to lock the PROM device from access. You can also download data in Intel 8080 hexadecimal format into the URAM from any host system that has an RS-232 interface. In off-line configuration, the universal programmer firmware does not recognize any iPPS commands.

Summary of On-Line Commands

In the on-line configuration shown in Figure 1-4, iPPS commands control the modification and transfer of data between the four iPPS storage devices. The commands that the iPPS software recognizes can be divided into six groups: the program control group, the utility group, the buffer group, the formatting group, the copy group, and the security group. The *iPPS PROM Programming Software User's Guide* describes each command.

Program Control Group

The following program control commands control iPPS program execution in various ways.

EXIT Exits the iPPS software and returns control to the ISIS operating system.

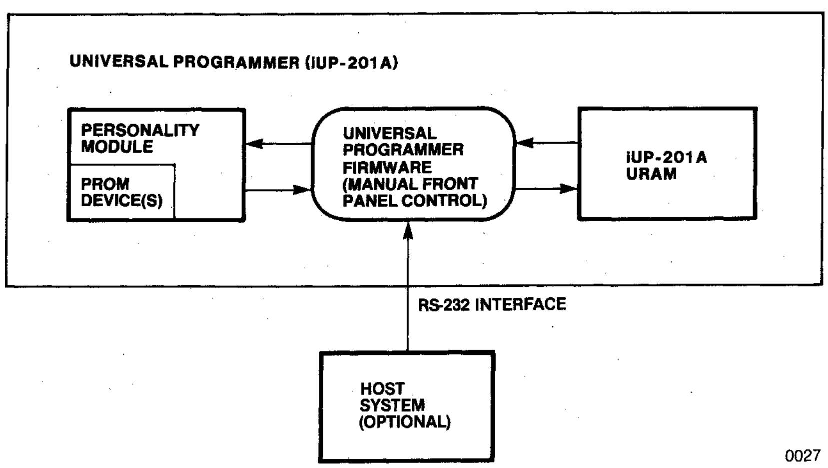


Figure 1-5 Off-line System Data Flow

<ESC>	Terminates the current command.
REPEAT	Re-executes the previous command.
ALTER	Allows edit and re-execution of the previous command.

Utility Group

The following utility commands display data, help, and status information and set default values.

DISPLAY	Displays PROM, buffer, or file data on the terminal.
PRINT	Prints PROM, buffer, or file data on the local printer.
QUEUE	Prints PROM, buffer, or file data on the network spooled printer.
HELP	Displays help information.
MAP	Displays buffer structure and status.
BLANKCHECK	Checks for unprogrammed PROMs.
OVERLAY	Checks if non-blank PROM devices can be programmed.
TYPE	Selects PROM type.
INITIALIZE	Initializes default number base and default file type.
WORKFILES	Specifies the drive device for temporary work files.

Buffer Group

The following buffer commands edit, modify, and verify data in the buffer.

SUBSTITUTE	Examines and modifies the buffer data.
LOADDATA	Loads a section of the buffer with a constant.
VERIFY	Verifies data in a PROM with buffer data.

Formatting Group

The following data formatting command formats and rearranges data from the PROM, buffer, or file devices.

FORMAT	Interactively formats buffer, PROM, or file data and places the result in a work file.

Copy Group

The copy group contains variations of the general-purpose COPY command. The following commands let you copy data between the buffer, PROM, or file devices.

COPY (file to PROM)	Programs a PROM with data from a disk file.
COPY (PROM to file)	Stores PROM data in a disk file.
COPY (buffer to PROM)	Programs a PROM device with buffer data.
COPY (PROM to buffer)	Loads the buffer with data in a PROM.
COPY (buffer to file)	Stores buffer data in a disk file.
COPY (file to buffer)	Loads the buffer with data in a disk file.

The following iPPS commands transfer data to and from the iUP-201A local RAM (URAM) only during on-line operation.

COPY (file to URAM)	Loads disk file data into the local URAM.
COPY (URAM to file)	Stores local URAM data into a disk file.
COPY (buffer to URAM)	Loads buffer data into the local URAM.
COPY (URAM to buffer)	Loads local URAM data into the buffer.

Security Group

The following security group command programs selected devices to prevent unauthorized access.

KEYLOCK	Interactively prompts for any related parameters and locks the device from access.

Summary of Off-Line Functions

The iUP-201A function keys manually control the manipulation and transfer of data between the URAM and PROM devices. These keys perform the following functions:

ON-LINE	Selects on-line or off-line operation.
ROM TO RAM	Loads PROM data into the URAM.
VER	Verifies PROM data against URAM data.
PROG	Programs the PROM with URAM data.
DEVICE SELECT	Selects the type of PROM device.
CLEAR	Clears entry or terminates the current off-line function.
ENTER	Enters addresses, data, and baud rates.
SHIFT ADDR-0	Selects data display function.
SHIFT DATA-1	Selects modify data function.
SHIFT FILL-2	Fills a selected address range in URAM with a constant.
SHIFT LOAD-3	Loads URAM with data in Intel 8080 hexadecimal file format, down-loaded from a host system through an RS-232 interface.
SHIFT LOCK-4	Locks the EPROM device from unauthorized access.

Chapter 3 describes the iUP-201A function keys.

CHAPTER 2
PREPARATION FOR USE

This chapter describes the universal programmer's hardware specifications and installation procedures and how to prepare and initialize the system.

Compatible On-line Hosts

You can connect the universal programmer to the following Intel development systems (the *Development Systems Handbook* contains information on other compatible Intel hosts):

- Intellec 800
- Intellec Series II
- Intellec Series III
- Intellec Series IV

These hosts must have a single- or double-density flexible disk drive or a hard disk. You also need a separate CRT terminal for the user interface if you use the Intellec 800. The Intellec 800 also requires a serial converter (refer to the RS-232 Cable Installation section in this chapter). Communication takes place over serial channel 1 or serial channel 2, as specified in the iPPS invocation line (described in the *iPPS PROM Programming Software User's Guide*). The universal programmer automatically adjusts itself during initialization to the host computer's serial baud rate (110 to 9600 baud).

All hosts use the iPPS software which runs as a utility under the ISIS operating system. Refer to the appropriate development system documentation for details on booting the ISIS operating system.

You need at least 64K of memory for the iPPS software. The ISIS operating system (version V3.4 and later for Intellec 800 and Series II and III and V1.0 and later for Series IV) must be in memory when the iPPS software is executing because ISIS input/output (I/O) routines are used.

The iPPS software runs only on Intel development systems. Because the universal programmer uses the industry standard serial RS-232 interface, it can also communicate with non-Intellec development systems or computers. This communication can be handled in the following two ways:

- User-written RS-232 control drivers
- iUP-201A universal programmer's download feature

RS-232 control drivers that have been written to operate the universal programmer in its on-line mode must follow a strictly defined command protocol. Refer to Appendix B for further information on this command protocol.

The iUP-201A universal programmer's serial down-load feature allows data in Intel 8080 hexadecimal format to be loaded into the iUP-201A URAM from any host system that has an RS-232 interface. In this mode, no special software drivers need to be written. Refer to the discussion of SHIFT-LOAD 3 in Chapter 3 for detailed instructions on using this feature.

EPROM Erasure

You can erase EPROMs by exposing the integrated circuit to ultraviolet light through a window on the chip package. Erasure occurs when the exposure light has a wavelength shorter than approximately 4000 Angstroms.

Sunlight and some florescent lamps have wavelengths in the 3000-4000 Angstrom range. Constant exposure to room-level florescent light erases the typical EPROM in approximately three years, while exposing the EPROM to direct sunlight erases it in about one week.

If the EPROM device is exposed to these lighting conditions for extended periods of time, you should cover the device window with opaque labels (available from Intel) to prevent unintentional erasure.

The optimum light for erasing EPROMs has a wavelength of 2537 Angstroms. The integrated dose (UV intensity X exposure time) for erasure should be a minimum of 15 W-sec/cm^2. The erasure time is approximately 15 to 20 minutes using an ultraviolet lamp with 12,000 uW/cm^2 power rating. The EPROM should be within one inch of the lamp tubes during erasure.

Do not power up the EPROM during erasure. If you power up the EPROM device during erasure, the internal current paths will cancel the energy that the ultraviolet light provides, and the device will not be erased.

Consult the *Microcontroller Handbook*, the *Microprocessor and Peripheral Handbook*, and the *Memory Components Handbook* for further information on erasing EPROM components. Individual device specifications also contain erasure information.

Installation Procedures

This section describes installation procedures for the universal programmer.

CAUTION

To prevent possible damage to the universal programmer, verify that the line voltage select switches are set properly and the power fuse is the right value for the line voltage that is to be used before plugging in the unit and turning it on. Refer to the Setting Line Voltage and Checking and Replacing Power Fuse sections in this chapter.

Setting Line Voltage

WARNING

Only a qualified technical person should change the line voltage setting.

The universal programmer accepts one of four different AC line voltages. Before plugging in the universal programmer and turning it on, you should set it to the appropriate available line voltage as follows:

1. Verify that the universal programmer is unplugged from its power source.

2. Note the four line voltage settings summarized in a table printed on the back panel of the universal programmer. (Figure 2-1 shows the switches set for 120 VAC.)

   Set SW1 and SW2 for the line voltage to be used to power the unit (*verify that the letters are visible on the switches*) as follows:

   - 100 VAC: set both SW1 and SW2 to position A.
   - 120 VAC: set SW1 to position A and SW2 to position B.
   - 220 VAC: set SW1 to position B and SW2 to position A.
   - 240 VAC: set both SW1 and SW2 to position B.

3. If the correct power fuse is installed, plug the universal programmer into a power source of the selected voltage and turn it on. (Checking and replacing the power fuse is described in the next section.)

Checking and Replacing Power Fuse

WARNING

Only a qualified technical person should check or replace the power fuse.

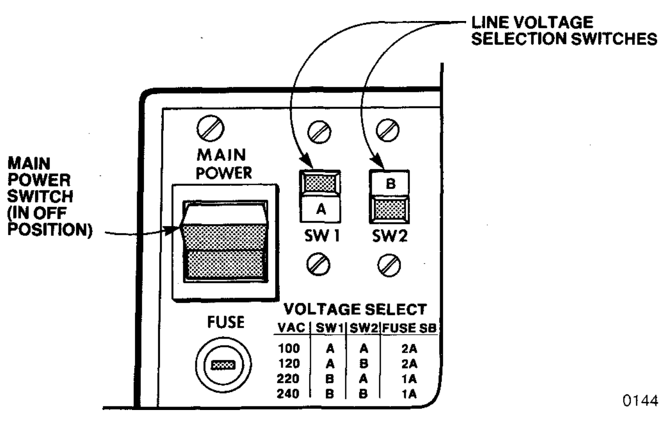


Figure 2-1 Setting Line Voltage

The fuse is located directly beneath the MAIN POWER switch on the back panel. (Figure 2-2 shows the fuse being removed.)

To replace the fuse:

1. Unplug the main power plug from its power source.
2. Using a flat blade screwdriver or a small coin, turn the fuse holder counter-clockwise as shown in Figure 2-2.
3. Once loosened, slide the fuseholder and fuse out as shown in Figure 2-2.
4. Check the fuse for continuity and verify that it is the proper size and amperage rating.
5. If the fuse is damaged or the wrong size or rating, replace it with one that is the correct amperage rating and size for the line voltage at which the unit is to be operated. The correct amperage ratings are as follows:

   100 VAC: 2 ampere slow blow
   120 VAC: 2 ampere slow blow
   220 VAC: 1 ampere slow blow
   240 VAC: 1 ampere slow blow

6. Slide the fuse and fuseholder back into the fuse receptacle and tighten by turning the fuseholder clockwise.

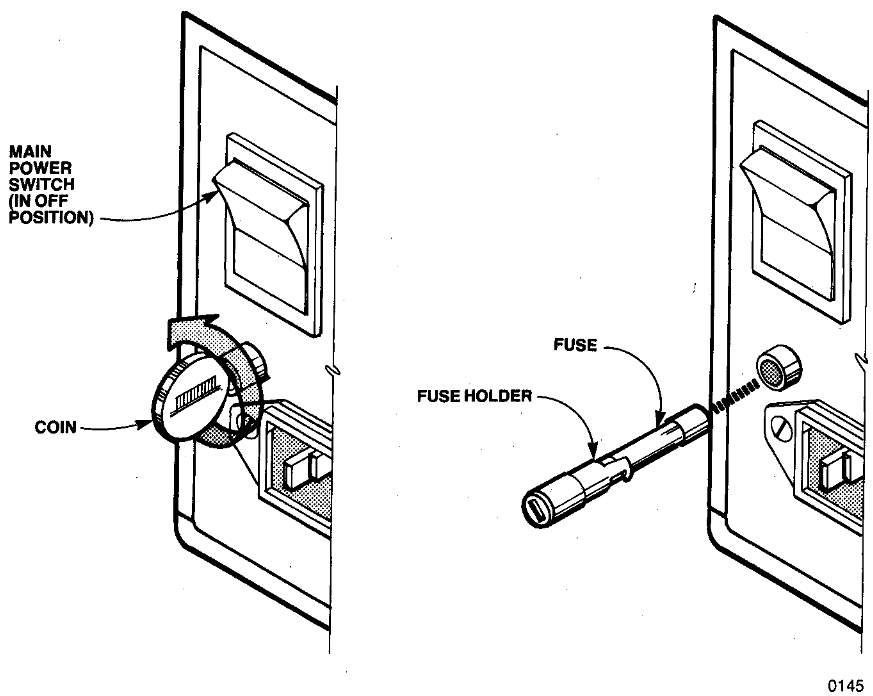


Figure 2-2 Replacing Power Fuse

7. Verify that the line voltage selector is set properly and then plug the main power plug back into the power source and turn on the universal programmer.

RS-232 Cable Installation

The RS-232 cable connects the host development system and the universal programmer. Perform the following steps to connect this cable:

1. Plug either end of the RS-232 cable into the RS-232 connector on the back panel of the universal programmer (see Figure 1-3). Main power can be either on or off. Connect the external ground wire to an available heat sink screw.

2. Plug the other end of the RS-232 cable into the appropriate serial connector on the back panel of the host development system. (This is the serial channel number you will specify when invoking the iPPS software. In most cases, this will be serial channel 1.) The serial connectors for Intellec systems are listed below.

   - Intellec 800. To use the Intellec 800 as a host system, use the connector labeled "TTY" (serial channel 1) or "CRT" (serial channel 2). To use TTY, you need an RS-232 to 20ma converter. Appendix D contains a schematic for making a converter.

   - Intellec Series II and Series III. To use either of these development systems as a host, use the connector labeled "SERIAL CH1/TTY" (J2) for serial channel 1 and "SERIAL CH2" (J3) for serial channel 2 on the development system back panel. When using channel 1, verify that the IEU board of the Series II/III has jumpers W5-A to W5-C, W6-A to W6-D, and W6-C to W6-B connected.

     Use one of the following configurations for channel 2:

     – Use the IEU board in its standard configuration (verify that it is jumpered W7-A to W7-B and W7-D to W7-C) and modify the iUP cable on the hose end by swapping pins 2 (transmit) and 3 (receive) and adding a jumper between pins 4 (RTS) and 5 (CTS).

     – Use the standard cable and reconfigure the IEU board by jumpering W7-A to W7-C, W7-D to W7-B, and W1-A to W1-C.

   - Intellec Series IV. To use the Series IV as a host, use the connector labeled "SERIAL CHANNEL 2" (whether you specify serial channel 1 or serial channel 2 when invoking the iPPS software). Verify that the IEU board is jumpered E28 to E29, E22 to E23, E25 to E26, and E16 to E17. If your IEU board is rev.-001 or -002, verify that it is jumpered E40 to E42 and E41 to E43; if your IEU board is rev.-003 or greater, E40 to E43 and E41 to E42.

     In addition, if you use channel 1, verify that the IEU board is jumpered E11 to E13 and E31 to E33.

     If you use channel 2, verify that the IEU board is jumpered E12 to E13 and E31 to E32.

Personality Module Installation

Personality modules adapt the universal programmer to a particular family of PROM devices. You cannot use a personality module alone; it must be installed in the universal programmer. The personality module plugs into the front panel connector of the universal programmer (see Figure 2-3). No further connection is required. During insertion and removal of the personality module, the main power switch can be either on or off.

System Initialization

This section describes system behavior during power on and initialization. Before powering on the universal programmer, the required equipment must be in place and properly connected.

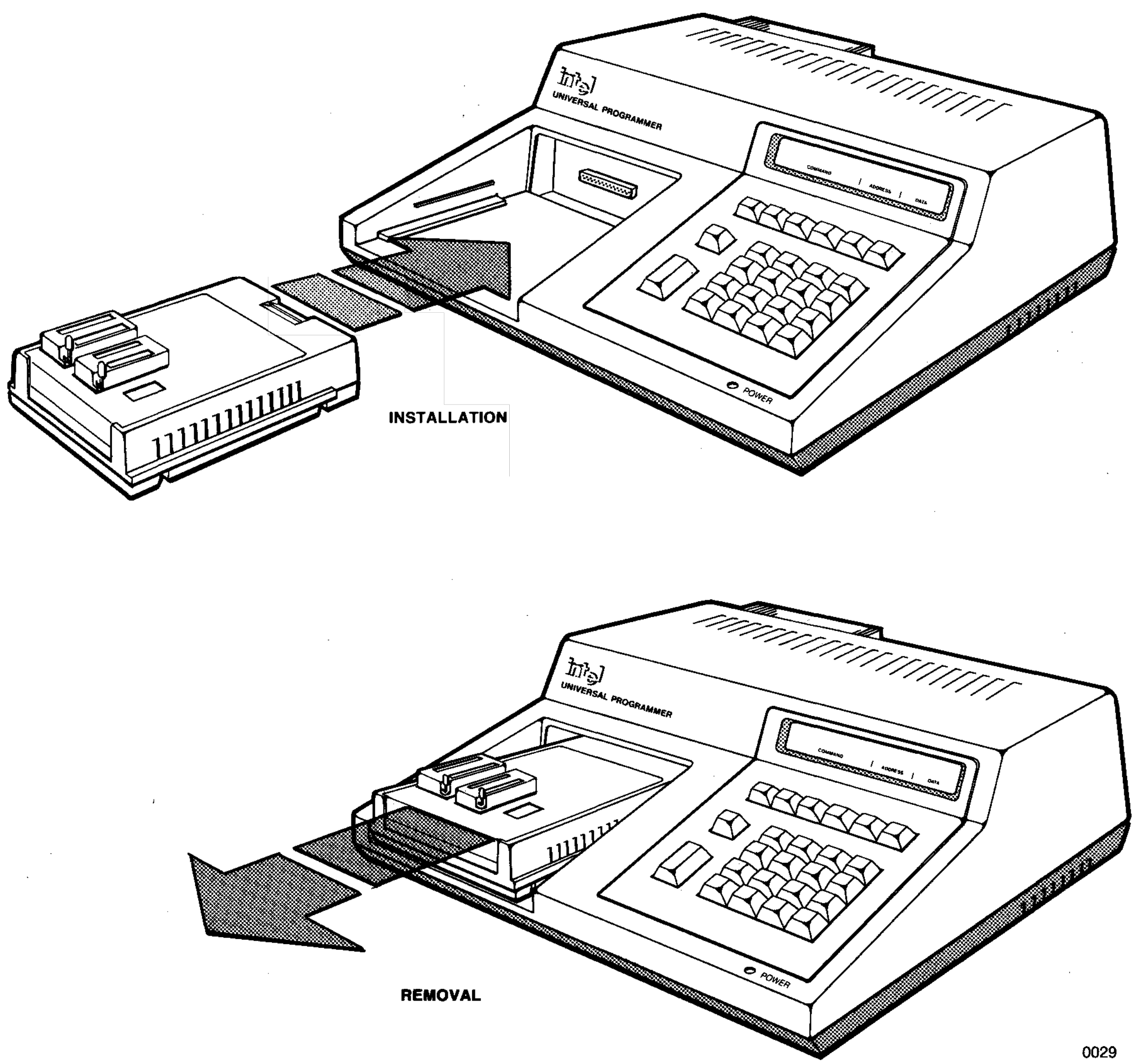


Figure 2-3 Personality Module Installation

CAUTION

The universal programmer requires a minimum of three inches of clearance for proper cooling, and its air vents must be clear of any obstructions.

Powering On

This section describes the behavior of the universal programmer immediately after it is powered on. This section assumes a properly connected system.

CAUTION

To prevent damage to the universal programmer, make sure that the line voltage select switches are set properly and the power fuse is the right value for the line voltage that is to be used before plugging in the unit and turning it on. Refer to the Setting Line Voltage and Checking and Replacing Power Fuse sections in this chapter.

To prevent damage or accidental programming of a PROM device, remove any PROMs from the personality module before powering on the universal programmer.

A device(s) should not be in the PROM socket(s) when the power is turned on or off.

After verifying that the line voltage selector and fuse are properly set, plug the universal programmer into a power source and set the MAIN POWER switch (rear panel) to ON (see Figure 2-1).

Self-test diagnostics are automatically run when the universal programmer is powered on. If any of these tests fail, error messages are displayed.

If the universal programmer is model iUP-200A, error messages are displayed on the terminal only after you enter the TYPE command (described in the *iPPS PROM Programming Software User's Guide*). If the iPPS software is not running when the iUP-200A universal programmer is turned on, no error messages are displayed on the terminal. Therefore, Intel recommends that the iPPS software be running before the universal programmer is powered on.

If your universal programmer is model iUP-201A, error messages are displayed only on the unit's alphanumeric display because the iUP-201A universal programmer initializes in off-line mode. While diagnostics are being performed, the following message is displayed:

DIAGNOSTICS IN PROGRESS

The following self-test diagnostics are performed:

1. A power supply test verifies that the universal programmer's internal voltages are within tolerance. If this test fails, then the following error message is displayed:

   POWER SUPPLY FAILURE

2. The following mother board tests are performed (for both the iUP-200A universal programmer and iUP-201A universal programmer):

   - CPU instruction test of the 8085 processor
   - Check-sum test on the internal firmware ROM
   - Read/write memory test on the 8085's RAM
   - Internal timer test

   If any of these tests fail, the following error message is displayed:

   MOTHERBOARD FAILURE

3. An interface test is performed (iUP-201A universal programmer only) with the iUP-201A universal programmer's KEY/RAM board. If this test fails, the following error message is displayed:

   KEY/RAM BOARD FAILURE

   If it passes all the self-diagnostic tests, the iUP-201A universal programmer displays the following message in the command, address, and data fields of the alphanumeric display:

   IUP READY 000000 55

   You can then use the iUP-201A universal programmer in off-line mode or switch to on-line mode for control by the iPPS software on the Intellec development system.

On-line Initialization

To operate in on-line mode, you must connect the universal programmer to an Intellec development system running under the ISIS operating system. Refer to the *iPPS PROM Programming Software User's Guide* for details on on-line operation.

Off-line Initialization

After checking the line voltage selector switches and power fuse, plug the model iUP-201A universal programmer into a power source and power it on. When powered on, the iUP-201A universal programmer always initializes in off-line mode.

Off-line operations do not require that the universal programmer be connected to a host development system. However, off-line functions are often performed in conjunction with on-line operations. For example, data can be copied to the URAM during on-line operation and then copied to a PROM off-line. Data can also be copied from a host system using the off-line SHIFT-LOAD 3 function (refer to the discussion of SHIFT-LOAD 3 in Chapter 3).

PROM Device Installation

PROMs are programmed in the personality module. Perform the following steps to insert the PROM to be programmed or read into the personality module.

1. Verify that the appropriate personality module for the device that is to be programmed is installed (refer to the Personality Module Installation section in this chapter).

CAUTION

Do not switch the universal programmer's power on or off when a PROM device is installed in a socket of the personality module. Damage to the PROM device can result.

2. If the universal programmer is not already powered on, switch on the power switch and wait for the initialization procedure to complete.

   Do not power on the system until you have read the Powering On section of this chapter.

3. Set the universal programmer for the PROM type to be programmed or read, using either the TYPE command (during on-line operation) or the DEVICE SELECT key (during off-line operation). LEDs next to the PROM designations on the personality module indicate the PROM type selected. If the personality module has more than one socket, a socket indicator light indicates the appropriate socket for the selected PROM type. Refer to the appropriate personality module user's guide if your personality module has a master socket and a program socket.

4. Raise the socket locking arm on the selected socket (see Figure 2-4).

CAUTION

If the personality module has more than one socket, install the PROM to be programmed or read only in the selected socket. Do not install a PROM or leave one in an unselected socket. Damage to the PROM can result.

5. Insert the PROM to be programmed into the socket with pin 1 of the device going into the socket pin hole at the upper left corner of the socket.

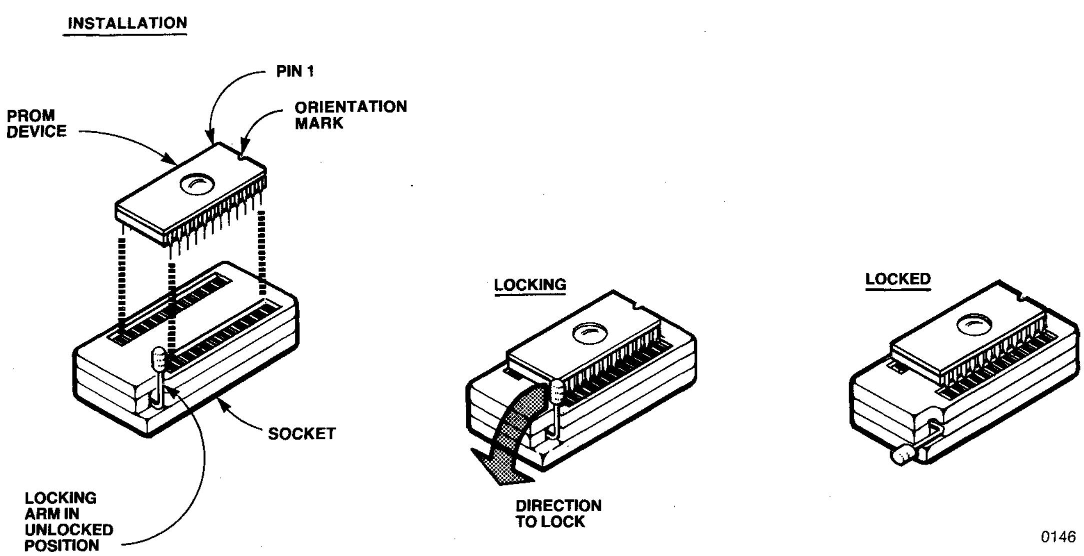


Figure 2-4 PROM Device Installation

CAUTION

The orientation mark on one end of the PROM must be toward the top of the socket. If a PROM is not oriented properly, it cannot be programmed and may be damaged. If the PROM is not oriented properly, the following error message may be displayed when the universal programmer attempts to access the PROM:

CHECK PROM INSTALLATION

6. Secure the PROM in the socket by moving the locking arm forward and down until it is parallel with the top of the personality module.

The PROM device is now ready for reading, programming, or verifying.

CHAPTER 3
OFF-LINE OPERATION

This chapter describes universal programmer off-line operation and the iUP-201A function keys. Off-line operation is available only on the iUP-201A model.

General Off-line Functioning

The iUP-201A universal programmer has a keyboard and an alphanumeric display for off-line operation. Internally, the iUP-201A universal programmer contains 32K of RAM. This RAM buffer is the URAM logical device that the iPPS software communicates with during on-line operation. Note that for PROMs greater than 32K bytes, URAM is used for off-line editing and cannot upload programs to the iPPS software.

In addition, the iUP-201A local firmware lets the universal programmer down-load data in Intel 8080 hexadecimal format into the RAM buffer (using an RS-232 interface) from any host system. This down-load operation does not use the iPPS software.

Only two storage devices are accessible during off-line operation: PROM and RAM. Off-line operations with these devices often consist of copying the data from a master PROM into the off-line RAM device, then programming the data back into a blank PROM. For example, the following off-line steps illustrate how to copy a master PROM that has less than 32K bytes.

1. Switch the iUP-201A universal programmer to off-line mode.
2. Insert the PROM to be copied.
3. Select the appropriate PROM type if required.
4. Press the ROM TO RAM switch to copy the PROM data into the RAM.
5. Remove the PROM that was copied.
6. Insert a blank PROM of the proper type.
7. Press the PROG switch to program the RAM data into the new PROM.

You can make additional copies of the original PROM by repeating steps 6 and 7.

The iUP-201A universal programmer also lets you modify data in the RAM on a byte-by-byte basis or fill a selected address range with a constant.

The off-line operation of the iUP-201A universal programmer requires all data transfers between the PROM device and the RAM to include the whole range of PROM device addresses. You can partially program a PROM in off-line mode by filling the addresses in RAM that do not contain valid data with a constant equal to the blank state of the PROM.

Internal Memory

This chapter refers to the 32K of universal programmer memory as RAM. This memory is a buffer that holds data to be manipulated off-line and programmed into the PROM. It is referred to as URAM when regarded as a logical device during on-line operations. (The *iPPS PROM Programming Software User's Guide* discusses on-line operation.)

The selected PROM type determines the address range of the RAM. Addresses are entered at the universal programmer hexadecimal keypad whenever access to RAM is required during off-line operations.

Keyboard/Display Overview

The keyboard illustrated in Figure 3-1 has a set of function keys plus a 16-key numeric keypad for entering addresses and data, in hexadecimal.

The alphanumeric display, a 24-character LED-type display, is divided into three fields: the command field, the address field, and the data field.

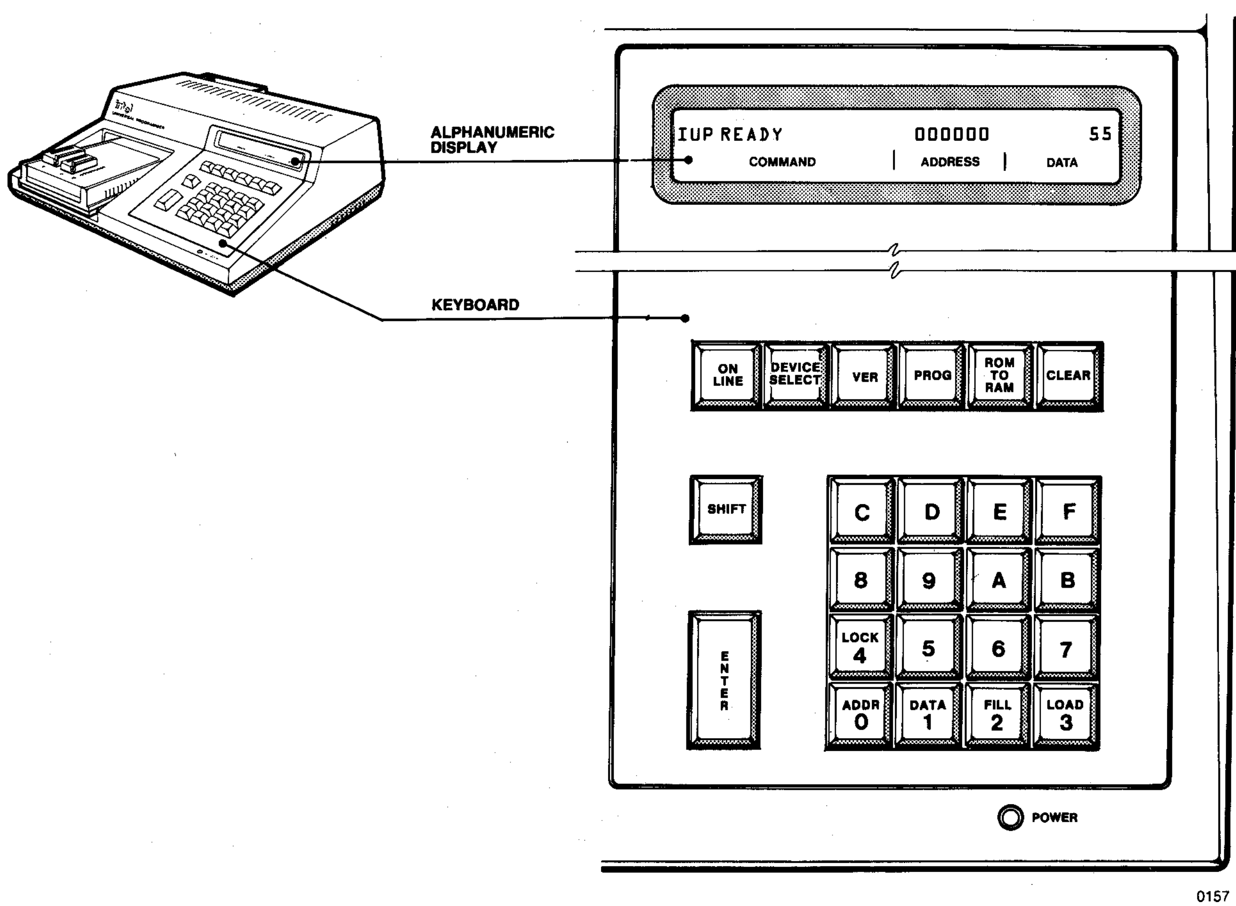
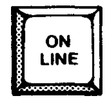


Figure 3-1 iUP-201A Keyboard and Display

The function keys let you select the operation you desire. Some operations require additional parameters such as addresses and data. These parameters are entered at the 16-key hexadecimal keypad.

Function Key Descriptions

The remainder of this chapter describes each off-line function key.



Selects on-line or off-line operation

Function

The ON LINE key selects either the on-line or the off-line mode of operation for the universal programmer. (Each time the ON LINE key is pressed, the universal programmer switches from one mode to the other.) After pressing the ON LINE key to enter on-line mode, the following message is displayed in the command field of the iUP-201A alphanumeric display:

ON LINE

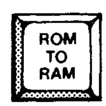
When in on-line mode, all iUP-201A keys are disabled (with the exception of the ON LINE key).

Pressing the ON LINE key again switches the universal programmer to off-line mode and causes the following message to be displayed:

IUP READY 000000 55

NOTE

The remaining keys described in this chapter operate only when the universal programmer is in off-line mode.



Loads PROM data into RAM

Function

For PROMs less than 32K bytes, the ROM TO RAM key loads the data from the PROM device installed in the personality module IC socket into the RAM. While loading is in progress, the following message is displayed in the command field of the alphanumeric display:

LOADING

When loading is complete, the following message is displayed:

END LOAD CKSUM = dddd

where dddd is the hexadecimal value of the check-sum obtained. The check-sum is the 2's complement of the 16-bit sum of all the bytes loaded into the RAM.

For PROMs greater than 32K you will get the message LOADING, although loading does not occur. You will get a valid check-sum, though. The ROM TO RAM key clears any editing done on PROMs greater than 32K. If you do not need a check-sum, use the faster DEVICE SELECT key to clear editing changes on PROMs greater than 32K.

If the PROM device is blank, the following message is displayed after loading is complete:

PROM BLANK   CKSUM = dddd

Thus the ROM TO RAM key is a convenient off-line way to determine if a PROM device is blank.

Verifies PROM against RAM data

Function

The VER key compares the PROM device data against the RAM data. While verification is in progress, the following message is displayed in the command field of the alphanumeric display:

VERIFY

If all locations compare, the VERIFY message in the command field is blanked out, and the following message is displayed at the end of the verification:

END VERIFY   CKSUM = dddd

where dddd is the hexadecimal value of the check-sum obtained. The check-sum is the 2's complement of the 16-bit sum of all the bytes verified.

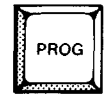
For PROMs greater than 32K you will get the message VERIFY, although verification does not occur. You will get a valid check-sum, though.

If a mismatch occurs, the following message is displayed:

VER ERR @   aaaaaa   dd

The address field contains the address (aaaaaa) at which the mismatch occurred, and the data field (dd) contains the exclusive OR of the expected and actual data.

After a mismatch, pressing the VER key again continues verification, starting at the next address after the mismatch address. A subsequent mismatch causes the events described previously. Pressing the CLEAR key aborts the verify function if a mismatch occurs.



Programs PROM with RAM data

Function

The PROG key programs the PROM device after first verifying that the PROM is blank. If the PROM device passes the blank PROM check, programming begins. If the PROM fails the blank PROM check, the following message is displayed:

BLANKCK ERR aaaaaa dd

The address at which the error was detected is displayed in the address (aaaaaa) field. The hex value of the expected blank state is XORed with the actual value found at that address of the PROM device and displayed (in hexadecimal) in the data (dd) field. A binary one in the data (dd) field indicates an unerased bit(s). For example, the data field contains 02. Converting 02 to binary (using Table C-2) gives you 00000010. The bit containing 1 was not erased.

If the PROM is already programmed, you can use the PROG key to perform a stuck bit check. The stuck bit check compares all PROM device bits that are preprogrammed with corresponding bits in the RAM. If all the preprogrammed (stuck) bits in the PROM are the same as their corresponding bits in the RAM, the universal programmer begins programming the PROM. Otherwise, the following message is displayed:

DEVICE WILL NOT PROGRAM

If this occurs, press the CLEAR key and try another PROM device.

The following message is displayed whenever a blank PROM check is in progress:

BLANKCK

The following message is displayed whenever the stuck bit check is in progress:

STUCKBIT CHK

The BLANKCK and STUCKBIT checks usually occur so quickly that you will not notice them.

Whenever programming is in progress, the following message blinks on the alphanumeric display:

PROGRAMMING

If an error occurs during programming and a location on the PROM device cannot be programmed to match the RAM data, the following message is displayed:

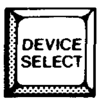
PROGRAM ERR aaaaaa dd

The address at which the error was detected is displayed in the address field (aaaaaa). The result of an exclusive OR of the expected and the actual (RAM) data is displayed in the data field. If this error occurs, the PROM is probably faulty. Press the CLEAR key and try another PROM.

A verification is automatically performed after the PROM device has been programmed. Refer to the VER function key description in this chapter for more details. If all locations are programmed without error the following message is displayed:

END VERIFY CKSUM = dddd

where dddd is the hexadecimal value of the check-sum obtained. The check-sum is the 2's complement of the 16-bit sum of all the bytes programmed into the PROM device.



Selects type
of PROM device

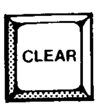
Function

For PROMs less than 32K bytes, the DEVICE SELECT key selects the PROM type when you use a personality module capable of programming more than one type of PROM.

Each time the DEVICE SELECT key is pressed, the next valid PROM type for the installed personality module is indicated by a lighted indicator next to the PROM type printed on the personality module front panel. When the key is pressed on the last PROM type, the first PROM type is again indicated. Press the key until the desired PROM type is indicated.

The DEVICE SELECT key is inoperative if the installed personality module can only program one type of PROM. DEVICE SELECT is also inactive during programming, verifying, or ROM to RAM loading.

The DEVICE SELECT key clears any editing changes in PROMs greater than 32K bytes.



Terminates current off-line
function or clears entry

Function

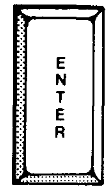
The CLEAR key terminates any ROM TO RAM, VER, PROG, or LOCK function in progress and restores the display to the following:

IUP READY 000000 dd

where dd is the data value in RAM at address 000000.

Pressing CLEAR will also restore the display after some error conditions.

When entering addresses or data in display, modify, fill, lock, or load modes, pressing the CLEAR key returns the field to the value it was at before the entry of the new address or data. If no entry has been made, pressing CLEAR exits the function and displays the IUP READY prompt.

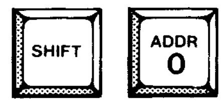


Enters addresses, data, and baud rates

Function

The ENTER key lets you enter addresses, data, and baud rates in display, modify, fill, lock, and load modes. After entering an address or data from the hexadecimal keyboard, press the ENTER key to tell the firmware to accept the entry.

Refer to the SHIFT-ADDR 0, SHIFT-DATA 1, SHIFT-FILL 2, SHIFT-LOAD 3, and SHIFT-LOCK 4 sections in this chapter for further information on the ENTER key.



Selects display data function

Function

The SHIFT key used with the ADDR 0 key lets you select an address for display. When you press the SHIFT and ADDR 0 keys simultaneously, the following appears on the alphanumeric display:

EDIT ADDR 000000 dd

The least significant digit of the address field blinks to indicate the current field of entry. Enter the most significant digit first. After entering a digit, the previously entered digits are shifted left. The least significant digit stops blinking once the address field is filled. Press the ENTER key to display the data at the selected address in RAM. If you press the CLEAR key instead of the ENTER key, the address field returns to the original address.

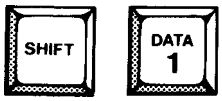
Any valid address entered on the hex keypad followed by the ENTER key becomes the current address value; the data at that location in RAM is displayed. If you press the ENTER key again without entering an address, the next sequential location is displayed. If you hold down the ENTER key for more than 1-½ seconds, sequential addresses are displayed at ½-second intervals until you release the key.

If you enter an address greater than the current valid size for the selected PROM type, the following message is flashed on the display for one second:

ADDRESS OUT OF RANGE

Then, all fields (command, address, and data) are displayed as they were prior to entering the invalid address. Address field editing restarts at the address that was in the address field before the erroneous entry.

Note that if no personality module is installed, valid addresses cover the entire range of the RAM (32K). Otherwise, the currently selected PROM type selects the valid range.



Selects modify data function

Function

The SHIFT key used with the DATA 1 key lets you edit the data at a specified address. When you press the SHIFT and DATA 1 keys simultaneously, the following appears on the alphanumeric display:

EDIT DATA aaaaaa dd

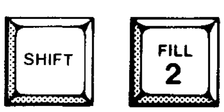
The data field displays the hexadecimal value for the byte of data located at the RAM address currently in the address field. The least significant digit of the data field blinks to indicate the current field of entry. Enter the most significant digit first. Upon entering a digit, the previously entered digit is shifted left. The least significant digit stops blinking once the data field is filled. Press the ENTER key to modify the data at the selected address in RAM. If you press the CLEAR key instead of the ENTER key, the data field returns to its original value.

Once you press the ENTER key, the address is incremented automatically and the data from the next sequential location is displayed in the data field for editing.

To exit the modify data mode, press the CLEAR key; the IUP READY prompt then returns. To modify data at another address or range of addresses, press the SHIFT and ADDR 0 keys to select the address, then press the SHIFT and DATA 1 keys to enter data modify mode.

You can make as many changes as you want if your PROM is less than 32K bytes. For PROMs greater than 32K, the universal programmer allocates thirty-one 1K blocks starting on 1K boundaries. You can make all the changes you want within each 1K block, but you can make only 31 changes if each address is in a different 1K block. The following error message is displayed if you try to make changes in more than thirty-one 1K blocks:

EDIT BUFFER FULL



Interactively fills RAM with a constant

Function

The SHIFT key used with the FILL 2 key selects the fill function. The fill function lets you load a contiguous section of RAM locations with a constant.

When you select the FILL function, you are prompted for parameters. During entry of these prompted values, the least significant digit of the input field blinks to indicate the current digit of entry. Enter the most significant digit first; upon entering a digit, the previously entered digits are shifted left. Once the prompted field is filled, the least significant digit stops blinking. Enter the entry by pressing the ENTER key or a function key.

The following message is displayed when you press the SHIFT and the FILL 2 keys simultaneously:

FILL FROM 000000

The address field displays 000000 with the least significant digit blinking. You can then enter a starting address using the hex keypad. The data field is blank.

After entering the start address, the following message is displayed:

FILL TO aaaaaa

The address field contains the previously entered start address as a default ending address. You can then enter an ending address using the hex keypad. The data field is blank. If the ending address you entered is less than the starting address or greater than the highest PROM address, the following message flashes on the display for one second:

ADDRESS OUT OF RANGE

Then the FILL TO address is requested again.

When you enter an acceptable address, the following message is displayed:

FILL WITH dd

The address field is blank. The dd is the blank state of the current PROM device selected. You can then enter a data constant using the hex keypad. After pressing ENTER, the selected address range in RAM is filled with the selected constant value.

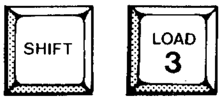
When the operation is complete, the universal programmer displays the following:

EDIT ADDR aaaaaa dd

Press CLEAR to exit the function and return to the IUP READY prompt.

You can fill as many address ranges with a constant as you want if your PROM is less than 32K bytes. For PROMs greater than 32K bytes, the universal programmer allocates thirty-one 1K blocks starting on 1K boundaries. You can fill as many address ranges with a constant as you want within each 1K block, but you can fill only 31 address ranges if each address range is in a different 1K block. The following error messages is displayed if you try to fill addresses in more than thirty-one 1K blocks:

RANGE EXCEEDS BUFFER



Downloads data from
host system off-line

Function

The SHIFT key used with the LOAD 3 key selects the load data off-line (HEX LOAD) function. This function lets you down-load data into the RAM buffer (using an RS-232 interface) in off-line mode from a host system. The data loaded must be in the Intel 8080 hexadecimal file format. (Refer to the *iPPS PROM Programming Software User's Guide* for information on file formats.) You can select any baud rate from 110 to 9600: 110, 150, 300, 600, 1200, 2400, 4800, or 9600.

NOTE

For PROMs greater than 32K, only the first 32K bytes of data are loaded. The rest of the data is truncated.

When you press the SHIFT and LOAD 3 keys simultaneously, the following appears on the alphanumeric display:

BAUD RATE = 2400

The least significant digit blinks. If 2400 baud is acceptable, press ENTER. To change the baud rate, enter the desired baud rate from the hexadecimal keypad. Press ENTER after you enter the baud rate. If you enter an unacceptable baud rate, the following message is flashed on the display for approximately two seconds:

ILLEGAL BAUD RATE

The BAUD RATE request is then displayed once more. Enter the baud rate again.

When the baud rate has been accepted, the following request is displayed:

START ADDRESS 0000

The least significant digit blinks. Enter the address in the file that is to be loaded into location 0000H of the RAM buffer and press ENTER. The data from the file address will be mapped to location 0000H in the RAM buffer. The iUP-201A firmware recognizes addresses from 0000H to FFFFH (64K range). If the START ADDRESS is greater than 8000H, the firmware automatically wraps around from memory location FFFFH to 0000H in mapping the RAM memory (see the example in Figure 3-2).

After you enter the start address, the following message is displayed:

HEX LOAD MODE

This message indicates that the iUP-201A universal programmer is ready to receive data from the RS-232 communications link.

As the data is being read, a check-sum is calculated at the end of each record. If an error in the check-sum occurs, the following error message is displayed:

CHECK SUM ERROR

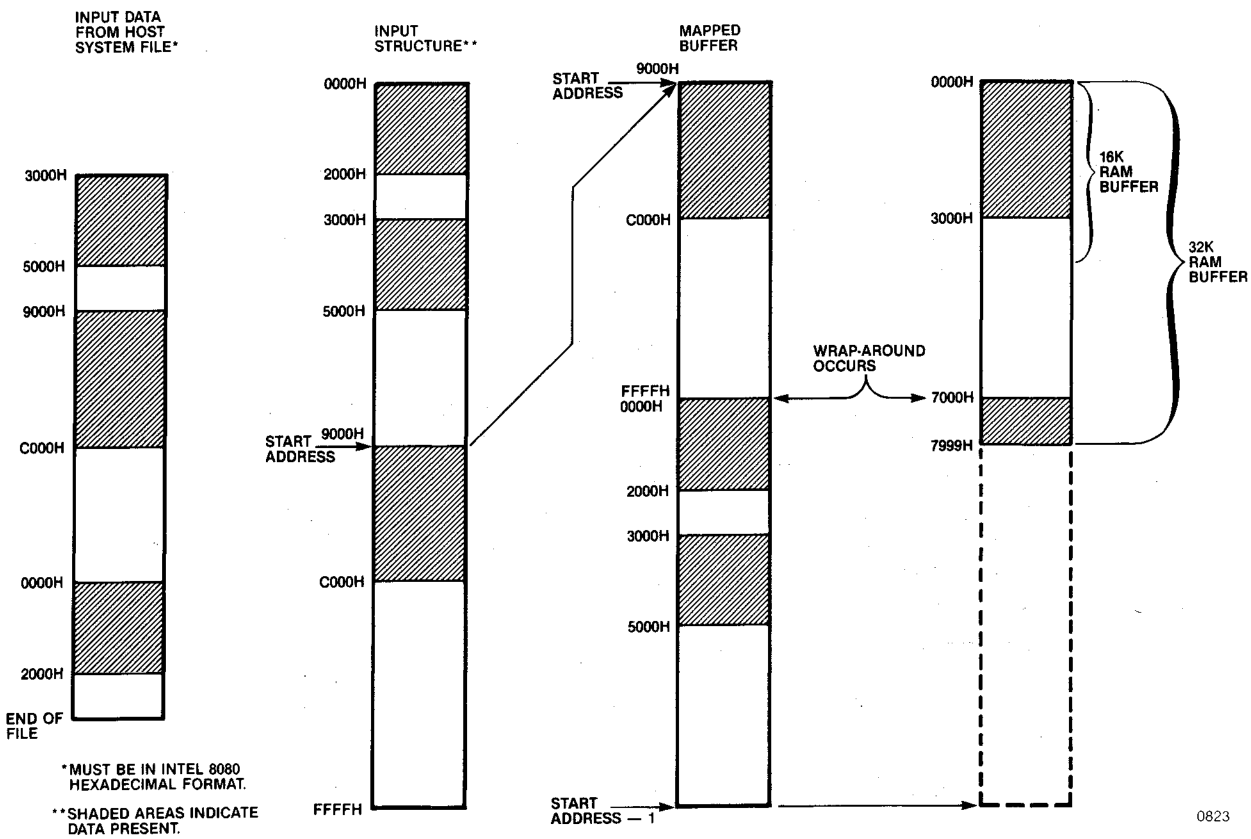


Figure 3-2 SHIFT-LOAD 3 Function Data Manipulation

The load operation is then terminated. Try downloading the data again. If you get the CHECK SUM ERROR message a second time, recreate the data.

If the data being read does not match the Intel 8080 hexadecimal file format, the following error message is displayed:

ILLEGAL FILE TYPE

The load operation is then terminated.

When the end of file is reached, the following message is displayed:

END OF FILE

The message indicates that the operation is completed. Press the CLEAR key to return to the IUP READY prompt.

SHIFT | LOCK 4

Locks the EPROM device

Function

The SHIFT key used with the LOCK 4 key locks the EPROM device from unauthorized access (on EPROMs on which this feature is supported). Two types of EPROMs can be locked: those with a security bit and those that are authenticated.

NOTE

The exact wording of the prompts and error messages may vary among personality modules. Refer to the user's guide of the personality module you are using for legal parameter values.

EPROM devices with a security bit can be locked from external access but cannot be unlocked without erasing the device. When you press the SHIFT and LOCK 4 keys simultaneously, the following appears in the command field of the alphanumeric display:

EXECUTE? - - Y=F/N=0

If you enter F, the EPROM is locked and the following message is displayed in the command field:

END LOCK

If the EPROM cannot be locked, the following message is displayed in the command field:

LOCK FAILED

If the EPROM is blank or already locked, the following message is displayed in the command field when you enter SHIFT-LOCK 4:

ALREADY LOCKED

You can lock and unlock authenticated EPROMs using an 8-byte identity code. The universal programmer needs several values to lock an EPROM. It asks you for this information by sending you prompts and blinking the data field. Enter the values you want to change and press ENTER. If you do not want to change a value in the data field, press only ENTER to call up the next prompt.

When you press the SHIFT and LOCK 4 keys simultaneously, the following appears in the command field of the alphanumeric display:

DELAY CNT dd

The delay count lets you specify the speed at which the authentication handshake takes place.

The key number identifies the PROM to the key manager. The universal programmer prompts for the first byte of the key number as follows:

KEY NUM-1

The following prompt is for the second byte of the key number:

KEY NUM-2

The following prompt is for the 8-byte identity code, which you enter one byte at a time.

KEY 1 dd

Then, one at a time KEY 2 through KEY 8 prompt you to enter values.

After you enter the 8-byte identity code, the following message appears in the command field and the data field blinks, prompting you to enter a command mask:

CMD MASK dd

The command mask value is necessary for locking the EPROM.

After entering all the parameters, the following prompt appears in the command field:

EXECUTE? - - Y=F/N=0

If you enter F (for yes), all parameters entered are written and the EPROM is locked. Note that you must enter a valid value for the command mask before the EPROM will lock.

If you enter 0 (for no), the DELAY CNT prompt appears. This lets you go through all the parameters again, either changing or retaining values.

If you enter an invalid parameter, the following message is displayed in the command field:

ILLEGAL PARAMETER VALUE

You are then prompted to enter another value for that parameter.

If the lock failed because a parameter cannot be programmed, you receive the following message:

FAILED @ XXXXXXXXXXXX dd

NOTE

Press the CLEAR key to clear the last entry; press the CLEAR key again to abort the LOCK command. If the LOCK command is aborted, no parameters are programmed into the EPROM.

CHAPTER 4
OFF-LINE PROM PROGRAMMING EXAMPLES

Introduction

The iUP-200A/201A universal programmer offers a wide variety of PROM programming, data display, and data editing capabilities. Not only can you quickly duplicate PROMs or program a file from a development system into a PROM, but you can also display and modify data before it is programmed into a PROM. The following examples will give you hands-on experience in using these features.

Before you go through these examples, read Chapter 3 of this manual to become familiar with the off-line function keys.

Table Of Contents Of Examples

The following table of contents is a quick reference to the examples in this chapter. It lists the programming or editing functions illustrated and the iUP-201A function keys demonstrated in each example.

EXAMPLE **PAGE**

Off-line Examples

The iUP-201A keyboard and display module let you perform many of the PROM-to-PROM programming functions available on-line in off-line mode. These functions include the following:

- Copying a PROM to a PROM.
- Copying an Intel 8080 hexadecimal file from a host system to a PROM.
- Modifying the data in the iUP-201A RAM before programming a PROM.

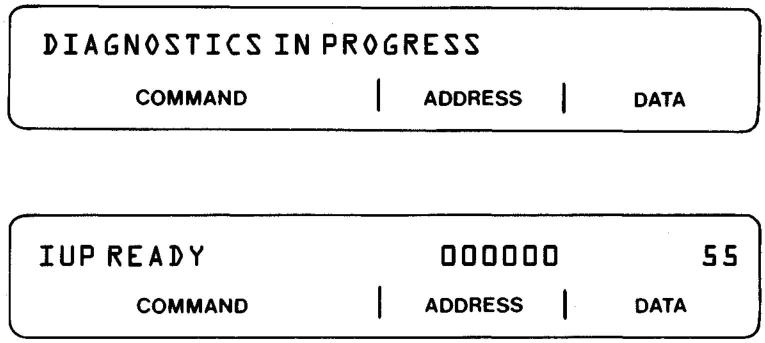
The following examples illustrate these functions. In these examples, the key-in sequence for a particular operation is on the left of the page and the resulting display is in the alphanumeric display right of the key-in sequence. Any numbers given in a key-in sequence are entered on the hexadecimal keypad. You must press the <ENTER> key after entering the number to complete the entry.

Off-line iUP-201A Initialization

You must initialize the iUP-201A firmware before performing any off-line functions.

Key-in Sequence

Power switch turned on



0870

When the rear panel power switch is switched on, a diagnostics routine is run. When the diagnostics are complete, the IUP READY prompt is displayed. This is the off-line prompt. The iUP-201A universal programmer must be in off-line mode to perform the following examples. Pressing the <ON LINE> key causes the ON LINE prompt to be displayed.

Key-in Sequence

0871

Before you begin off-line programming, you must set the iUP-201A universal programmer for the type of PROM to be duplicated or programmed. Each time you press the <DEVICE SELECT> key, the personality module is set to the next device type. The LED indicators on the personality module indicate the current PROM type selected and the socket in which it should be installed.

A 2716 EPROM that contains sample code is used in all the following examples except locking EPROMs. The data and check-sums shown were obtained from this code.

Duplicating a PROM

The first step in copying a PROM to a PROM is to read the contents of the master PROM into the iUP-201A universal programmer's RAM memory.

NOTE

The off-line RAM and the on-line iPPS URAM are the same.

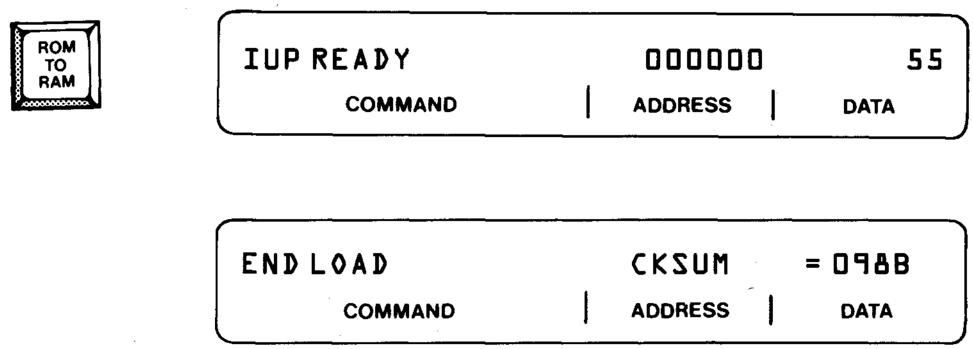
To read the contents of the master PROM into the iUP-201A universal programmer's RAM memory, place the master PROM in the PROM socket of the personality module. (Refer to Chapter 2 for PROM device installation.)

CAUTION

Be sure that the type of PROM that you are installing is the same as the type selected with the DEVICE SELECT key. If you do not specify its type correctly, you can damage a PROM when you try to program it or read it.

To copy the contents of the master PROM into the iUP-201A RAM, press the ROM TO RAM key.

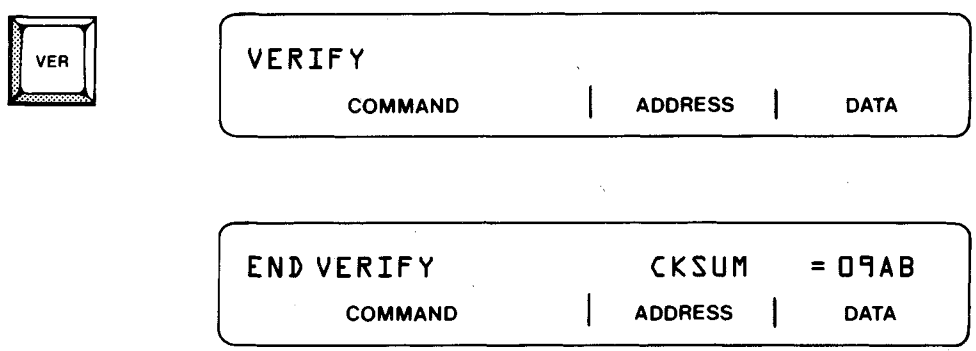
Key-in Sequence



0872

The check-sum is the 2's complement of the 16-bit sum of each byte read. The verify function lets you check the contents of the RAM against the contents of the master PROM.

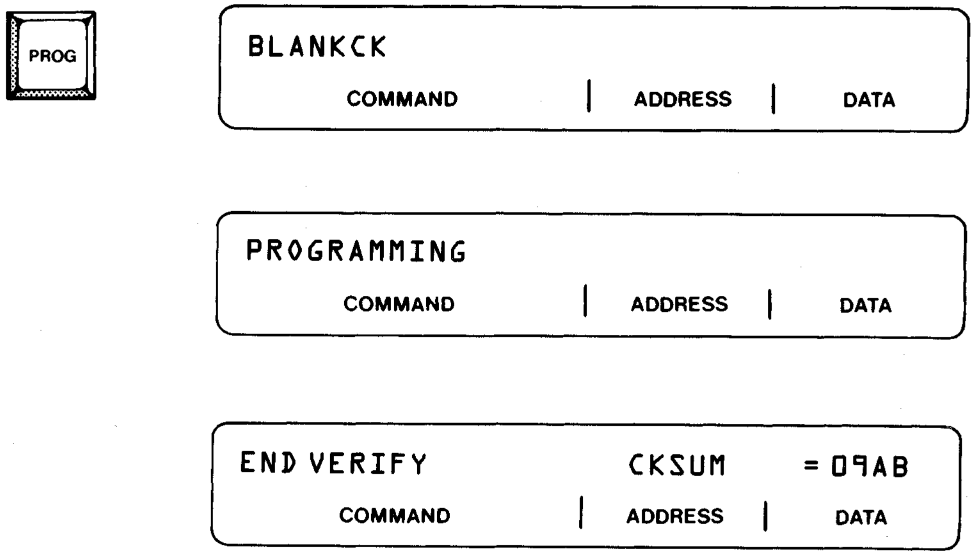
Key-in Sequence



0873

With the RAM loaded and verified, you can program the new PROM. Remove the master PROM from the personality module and replace it with a blank PROM. Copy the contents of the RAM to the blank PROM.

Key-in Sequence

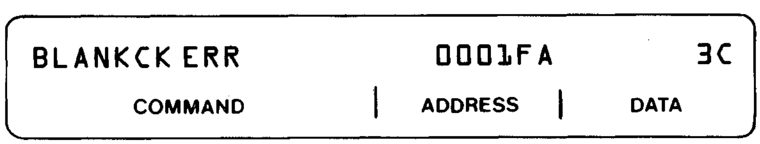


0874

As part of the programming routine, the iUP-201A universal programmer performs a blankcheck of the new PROM to verify that it can be programmed. When programming is complete, it performs the verify function to check that the programming was done correctly.

Note that the blankcheck routine generally happens so fast that the message "BLANKCK" is flashed momentarily. Also, the message "PROGRAMMING" blinks to indicate that programming is in progress. PROM programming can last several minutes.

If the iUP-201A universal programmer discovers that the PROM to be programmed is not blank, it displays a blankcheck error message such as the following:

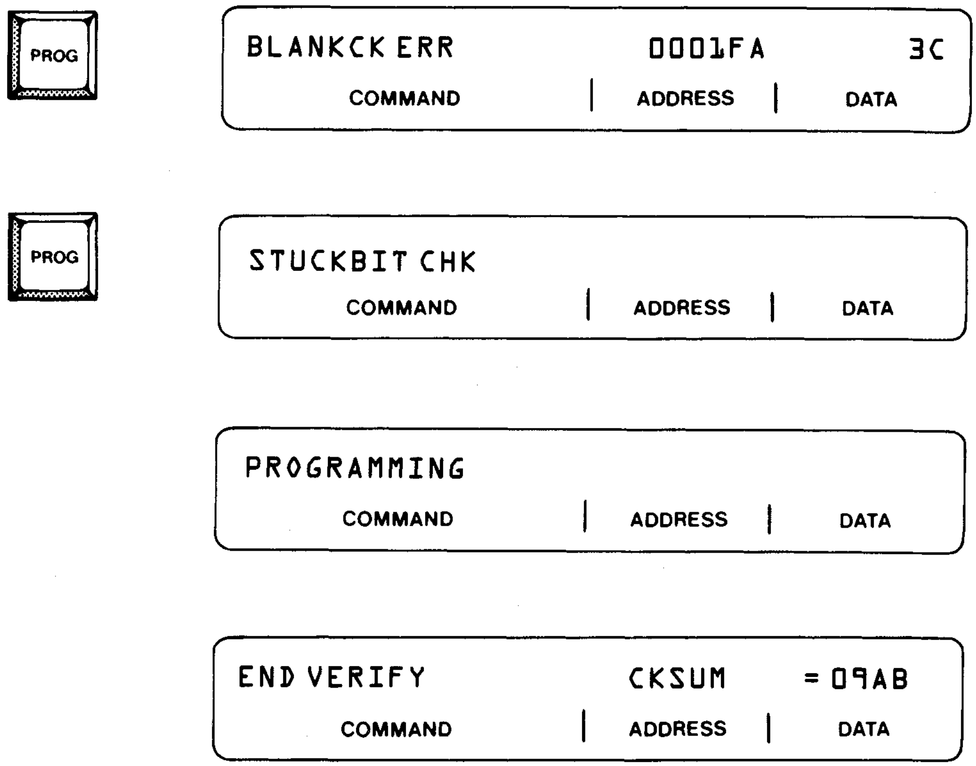


0875

The address of the byte that is not blank is displayed.

The fact that the PROM is not blank does not necessarily mean it cannot be programmed. Pressing the PROG key again causes the iUP-201A universal programmer to perform a stuck bit routine to check if the non-blank bits correspond with bits in the RAM. If the routine indicates that the PROM can be programmed successfully, the programming routine is initiated automatically. Here is an example of programming a non-blank PROM.

Key-in Sequence

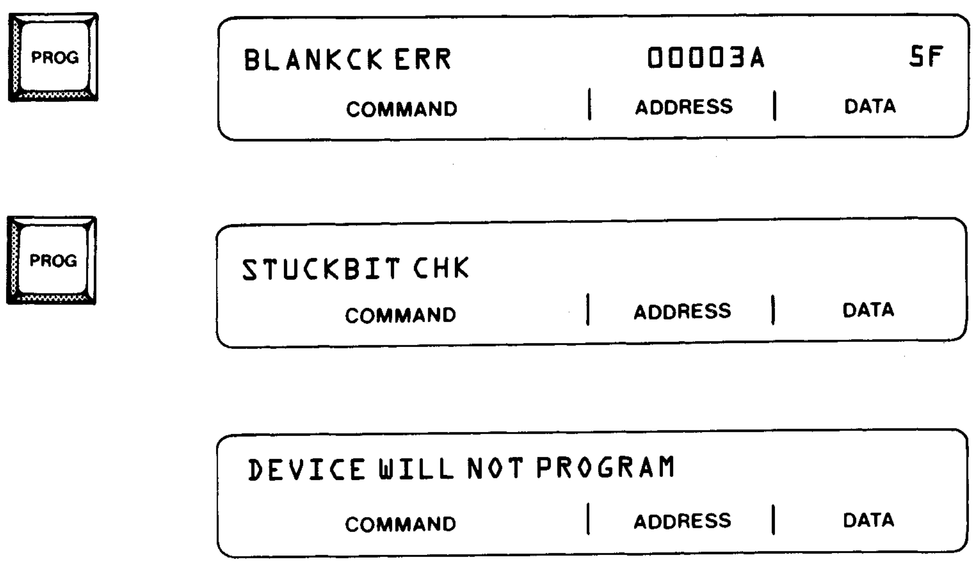


0876

The stuck bit routine usually occurs so fast that the STUCKBIT CHK message is flashed momentarily.

In the following example, the stuck bit routine determines that the PROM cannot be programmed.

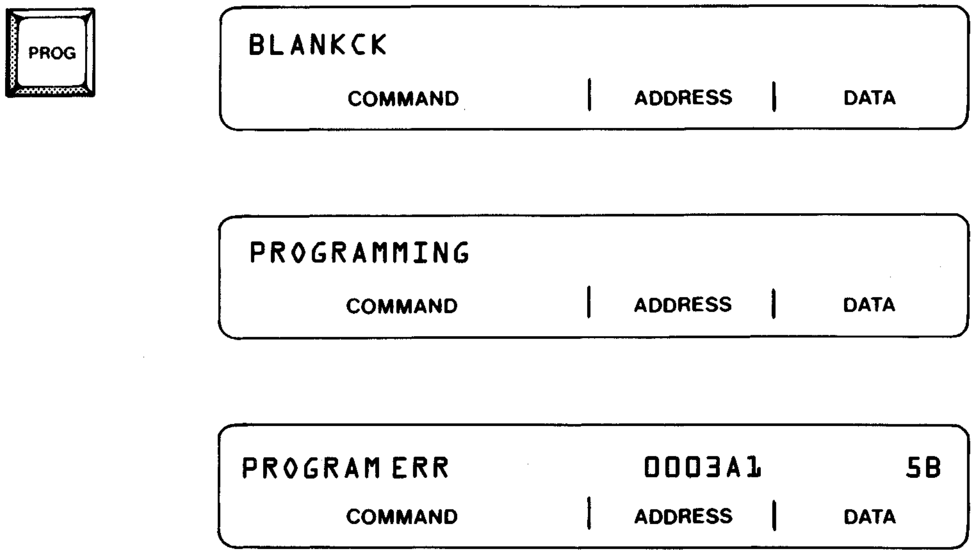
Key-in Sequence



0877

If an error occurs during programming or if the PROM is defective and one or more of the bits cannot be programmed to match the RAM data, the following message is displayed:

Key-in Sequence



0878

The address at which the error was detected is displayed in the address field. A program error is most often caused by a faulty PROM. Pressing the CLEAR key lets you program another blank PROM with the same information.

Copying a Development System File to a PROM

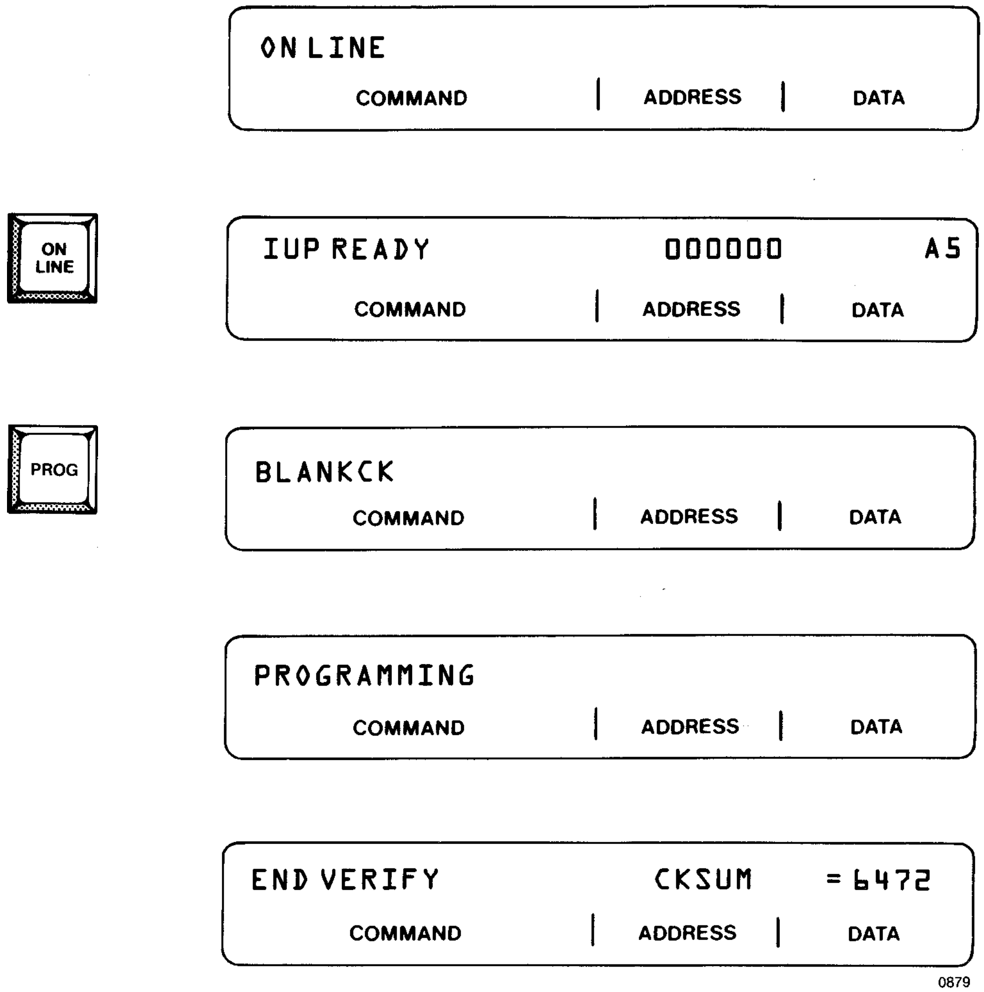
The iUP-201A universal programmer can copy a file from a development system into its RAM buffer either on-line with the iPPS software or off-line. Once the file is in the RAM, you can copy it into a PROM off-line.

In on-line mode, use the COPY FILE TO URAM command. (Note that in on-line mode, the term URAM refers to the RAM buffer in the iUP-201A universal programmer.) In the following example, the data from file CODE4.IUP on drive 1 is copied (starting at address 380H in the file) to the RAM (starting at URAM address 00H).

```
PPS>COPY :F1:CODE4.IUP(380,7FF) TO URAM
  CHECK SUM = 6472
PPS>
```

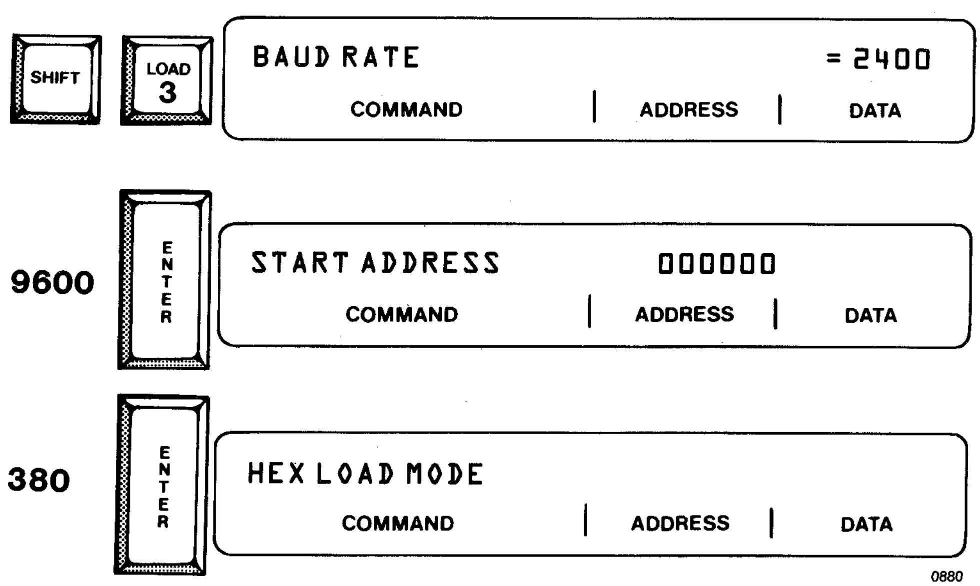
You can then program a PROM from the RAM off-line, as in the following example.

Key-in Sequence



You can perform this same function off-line (without invoking the iPPS software) provided that the file is in Intel 8080 hexadecimal format and the PROM is no larger than 32K bytes. Start writing the file into the iUP-201A RAM buffer off-line as shown in this example.

Key-in Sequence



When you start the SHIFT-LOAD 3 function, you are prompted for a baud rate. You can select any standard baud rate from 110 to 9600. In this example, 9600 baud is entered. You are then prompted for a starting address in the file to be down-loaded. This file starting address is loaded into location 00H in the RAM buffer. In this case, a file starting address of 380H is selected. After you enter the starting address, the message "HEX LOAD MODE" is displayed. You can now initiate the down-load from the host system.

When using an Intel development system, enter ISIS commands to tell the development system to down-load the file to the teletype port as follows:

>**SPEED 9600**
>**COPY :F1:CODE4.IUP TO :TO:**
>COPIED :F1:CODE4.IUP TO :TO:
>

After the file is read, the message "END OF FILE" is displayed on the iUP-201A alphanumeric display. Press the <CLEAR> key to return to the IUP READY prompt. If the file is not in Intel 8080 hexadecimal format, the message "ILLEGAL FILE TYPE" is displayed.

In this example, the file is loaded into the RAM buffer, with address 380H of the file stored at address 00H of the RAM buffer (see Figure 4-1).

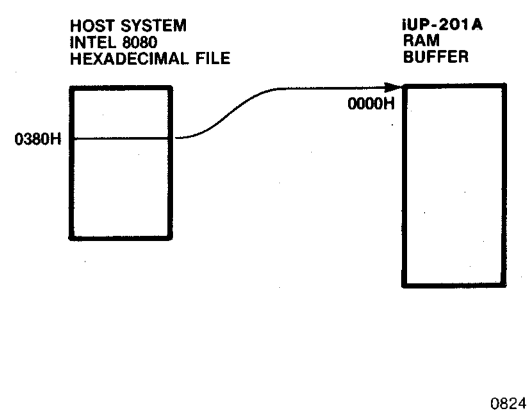
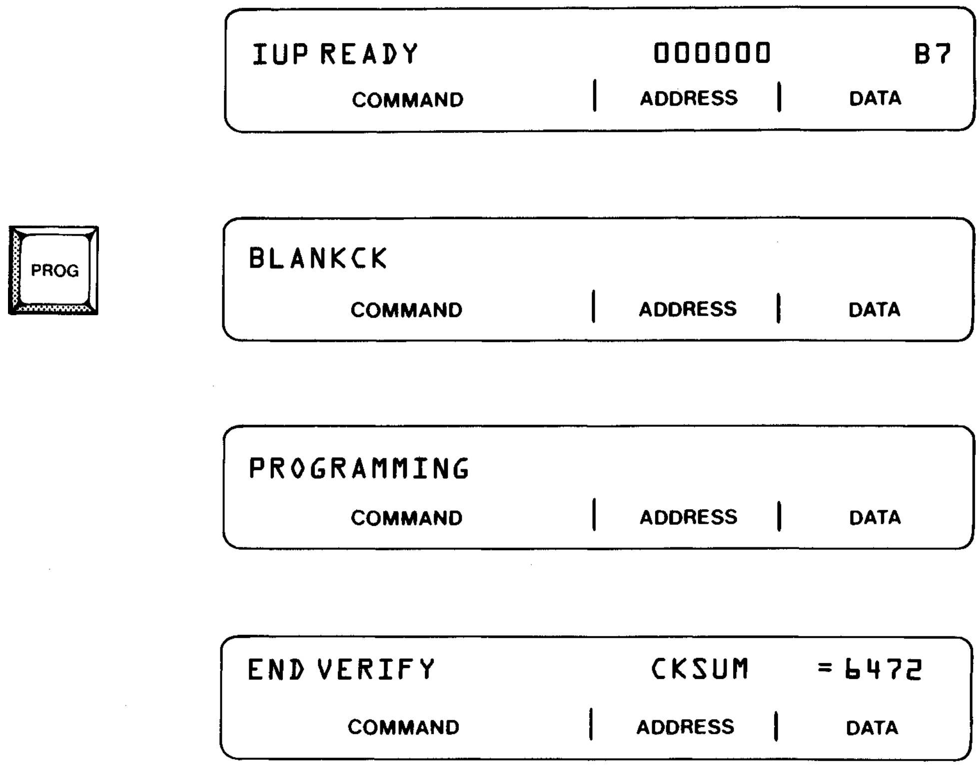


Figure 4-1 Example Addresses of SHIFT-LOAD 3 Function

Once the file is loaded into RAM, you can copy it to a PROM as shown in the following example:

Key-in Sequence

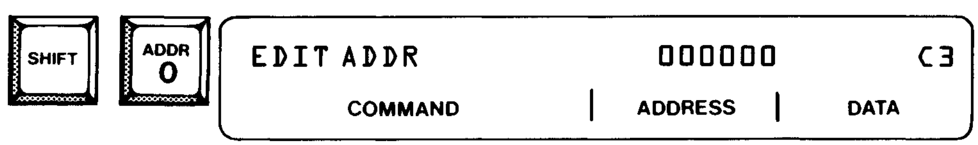


0881

Modifying Data in the iUP-201A RAM

The iUP-201A universal programmer lets you display data in its RAM buffer and modify selected bytes of data. Use the SHIFT-ADDR 0 function to display data.

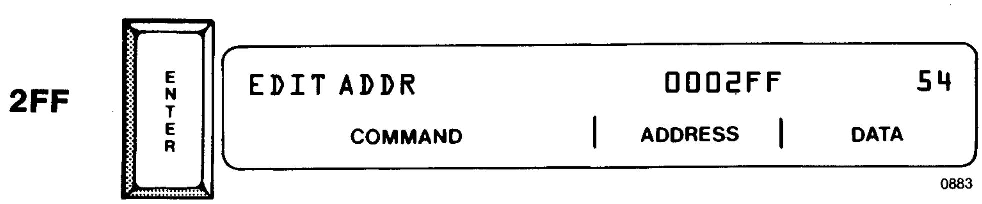
Key-in Sequence



0882

The least significant digit of the address field blinks to indicate that addresses are to be entered. To look at the contents of a specific byte, enter its address as shown in the following example.

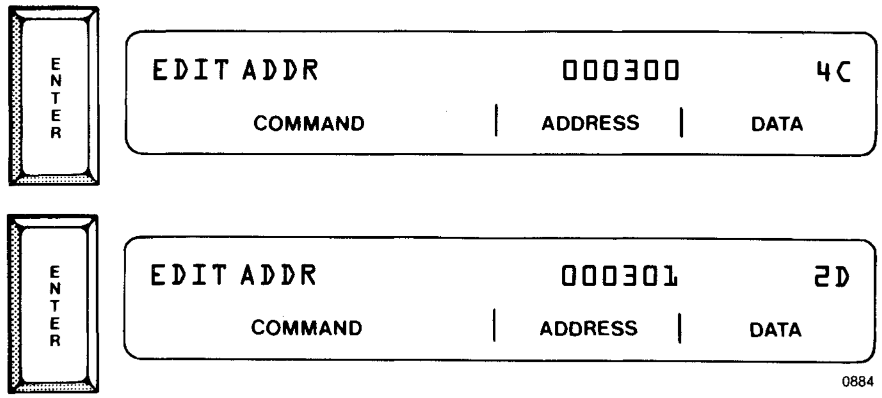
Key-in Sequence



0883

Pressing the ENTER key alone automatically increments the address by one.

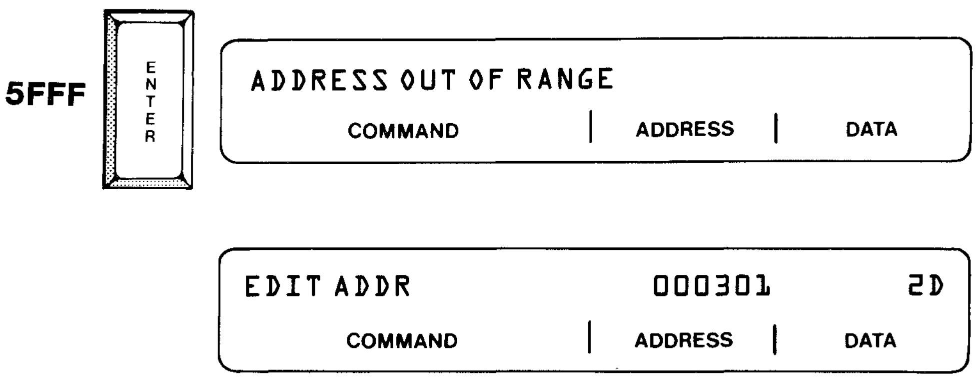
Key-in Sequence



0884

If you enter an address greater than the highest allowable PROM address, the following message is displayed.

Key-in Sequence

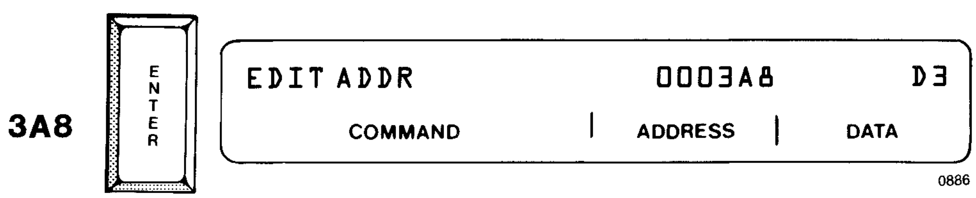


0885

Approximately one second after the message "ADDRESS OUT OF RANGE" is displayed, the previous address and data are displayed again.

To modify the data in a byte, first select the address of the byte that you wish to modify.

Key-in Sequence



0886

Then enter the data edit mode (SHIFT-DATA 1).

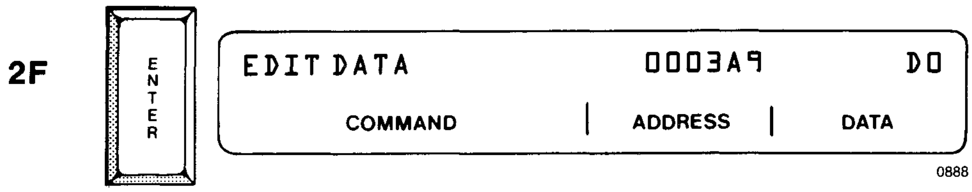
Key-in Sequence

0887

The least significant field of the data field blinks, indicating that you are in data modify mode.

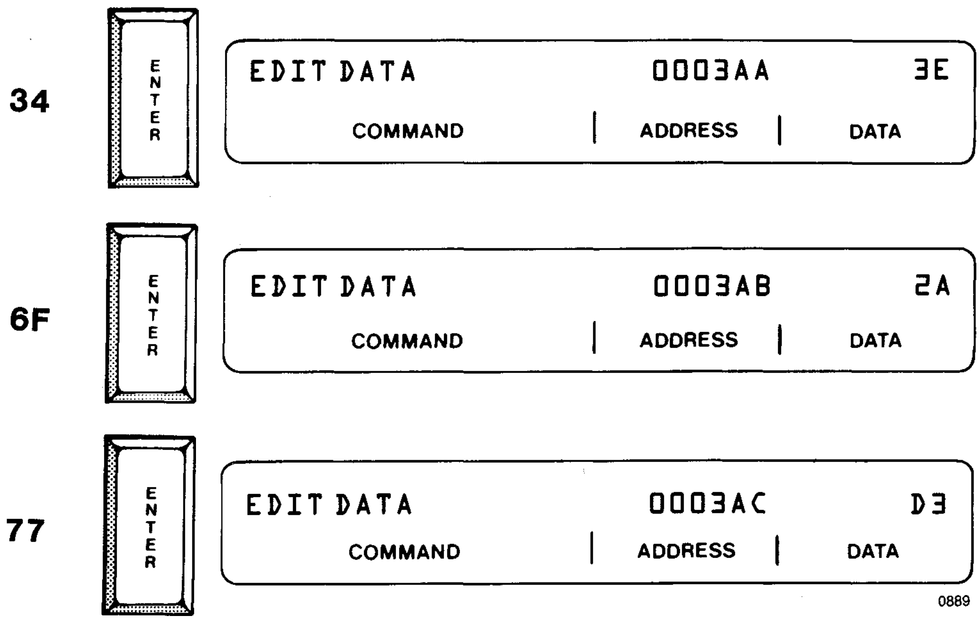
Enter the new data to be stored at the selected address.

Key-in Sequence



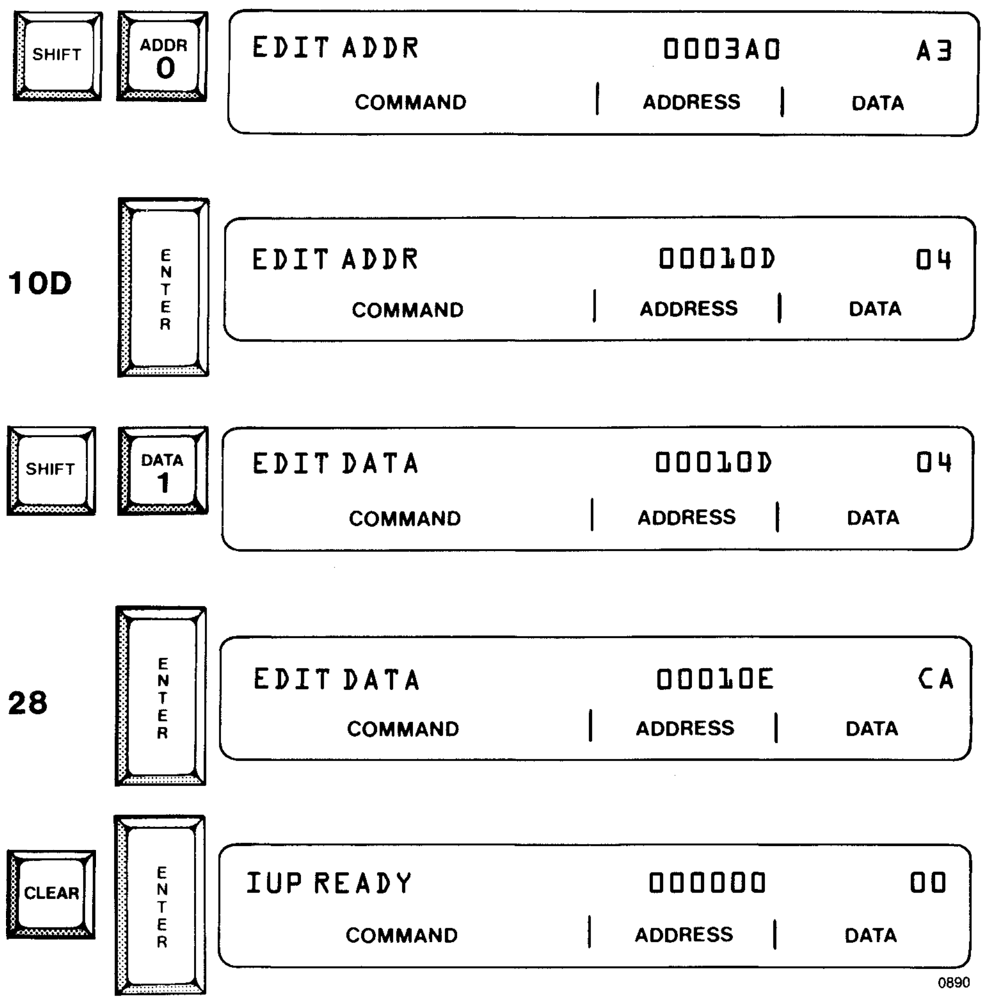
Once the data is modified, the address is automatically incremented and the least significant field on the data field blinks, indicating that you can modify the next byte. You can continue to modify bytes in sequential addresses in this manner.

Key-in Sequence



To modify a byte that is not in sequence, enter display data mode (SHIFT-ADDR 0) to establish the address to be modified and then enter edit data mode again.

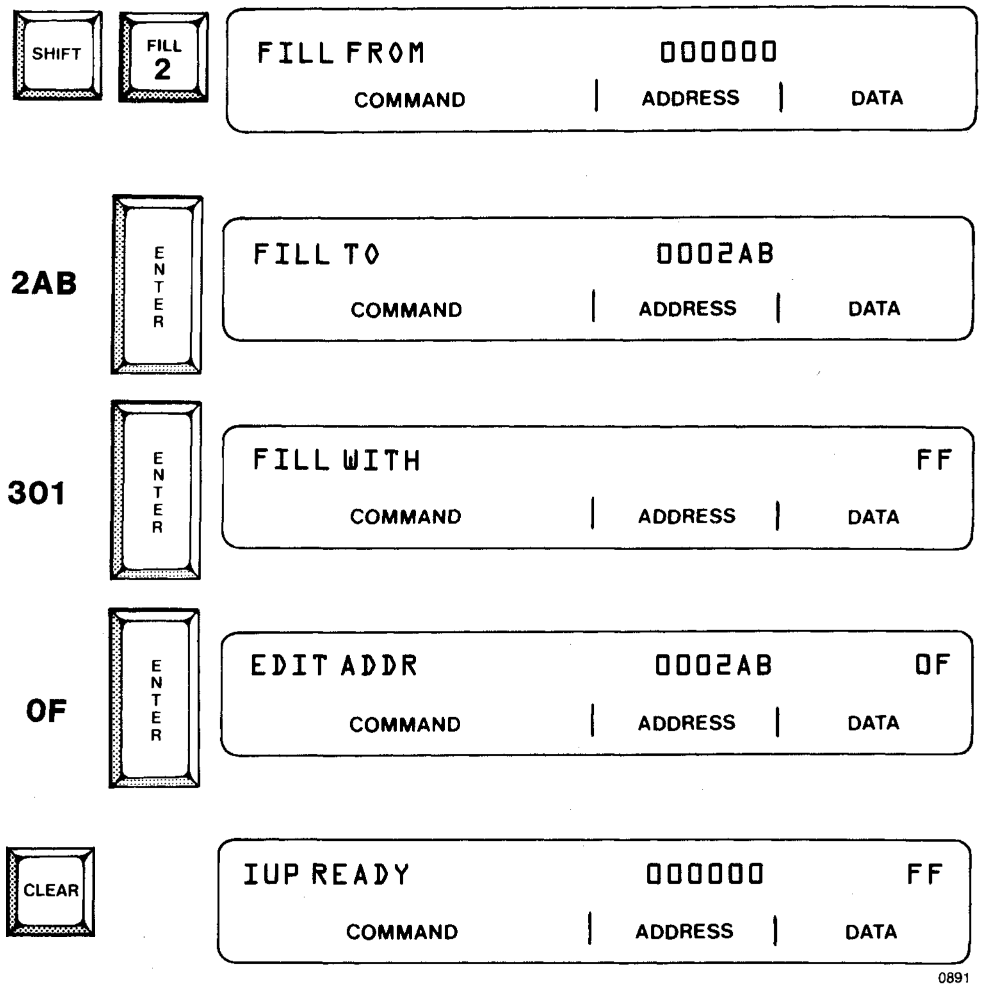
Key-in Sequence



Pressing the CLEAR key returns program control to the IUP READY state.

The SHIFT-FILL 2 function is an alternate method of modifying the data in the iUP-201A RAM buffer. This function lets you set all the bytes in a selected address range to a specific constant. In the following example, the bytes from addresses 2ABH to 301H are filled with the constant 0FH.

Key-in Sequence



Once the addresses are filled with the constant, the iUP-201A universal programmer goes into display address mode and the data in the first address to be filled is displayed. Press the <CLEAR> key to return to the prompt "IUP READY."

Note that the maximum address that you can enter from a 2716 EPROM is 7FFH. If you enter an address greater than 7FFH, the message "ADDRESS OUT OF RANGE" is displayed.

Locking an EPROM

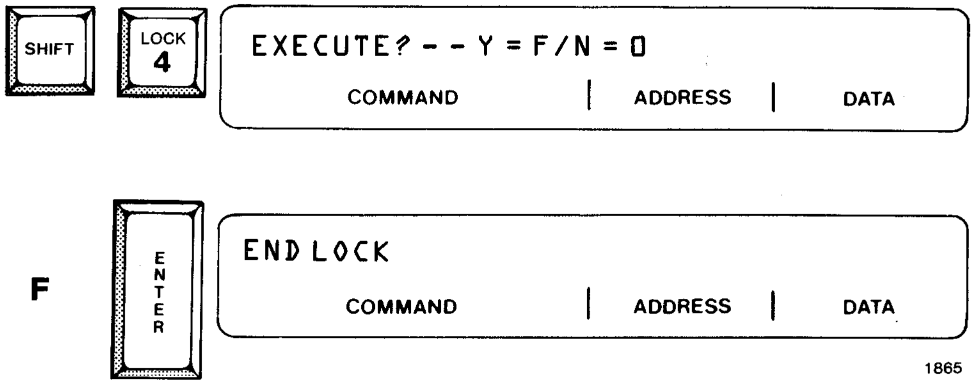
After programming an EPROM, you can protect it from unauthorized access by locking it. You can lock two types of EPROMs: those with a security bit and those that are authenticated.

NOTE

Prompts and error messages may vary among personality modules.

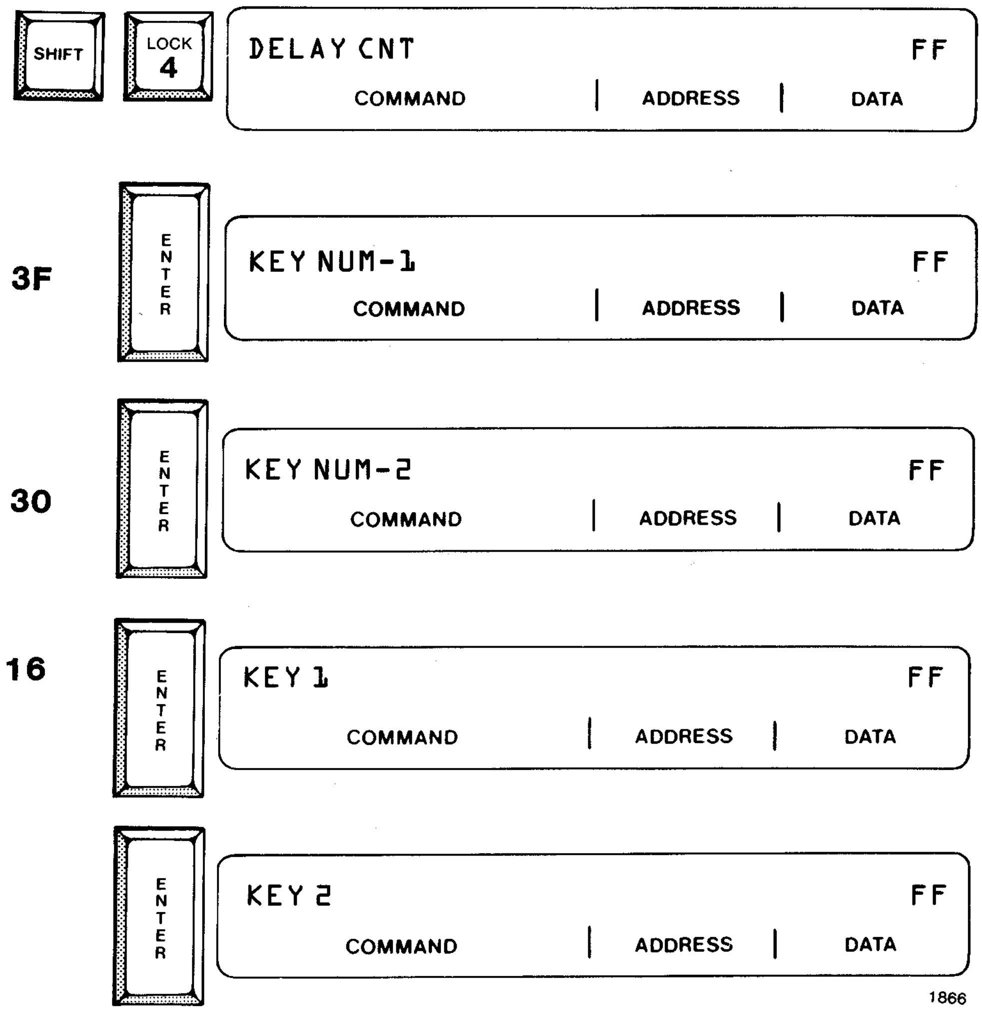
The following example illustrates locking an 8751H microcontroller which has a security bit.

Key-in Sequence



1865

The following examples illustrate locking an 87128 authenticated EPROM.



1866

		COMMAND / ADDRESS / DATA
01	ENTER	KEY 3 FF
11	ENTER	KEY 4 FF
FF	ENTER	KEY 5 FF
10	ENTER	KEY 6 FF
41	ENTER	KEY 7 FF

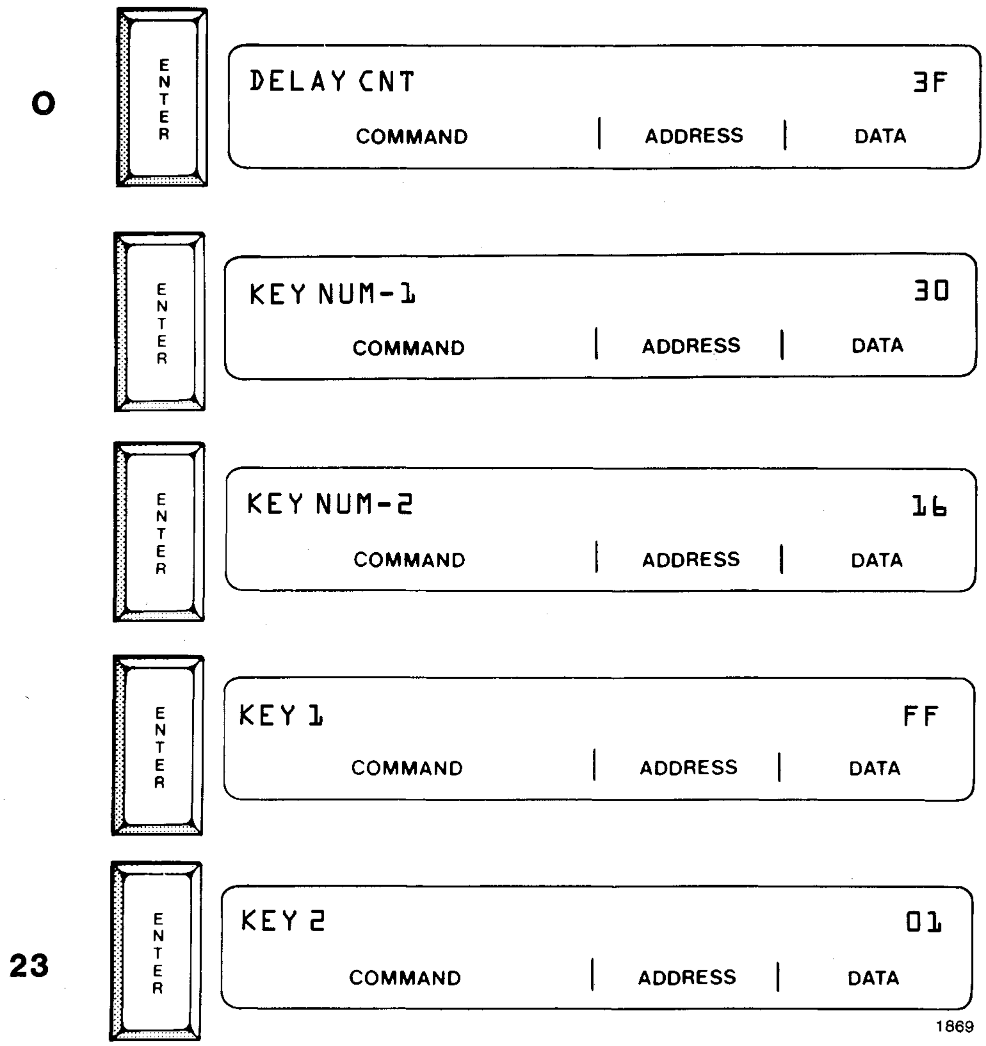
1867

		COMMAND / ADDRESS / DATA
	ENTER	KEY 8 FF
3F	ENTER	CMD MASK FF
E2	ENTER	EXECUTE? - - Y = F/N = 0

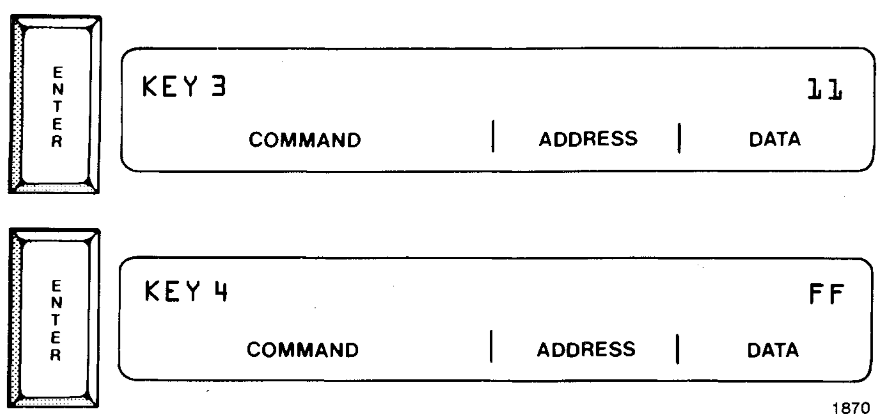
1868

By entering 0 (no) to the EXECUTE? prompt, you are able to go through the command sequence again and verify your entries.

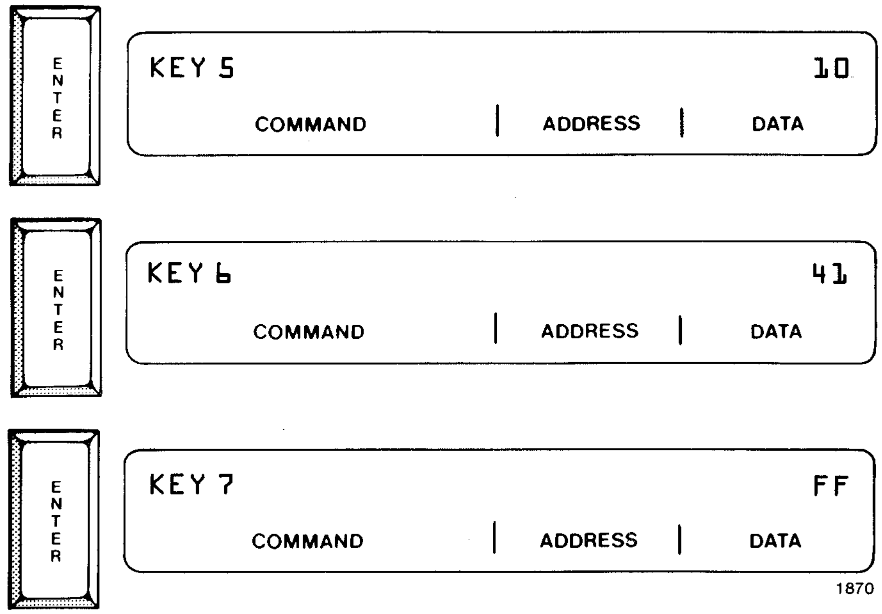
Key-in Sequence



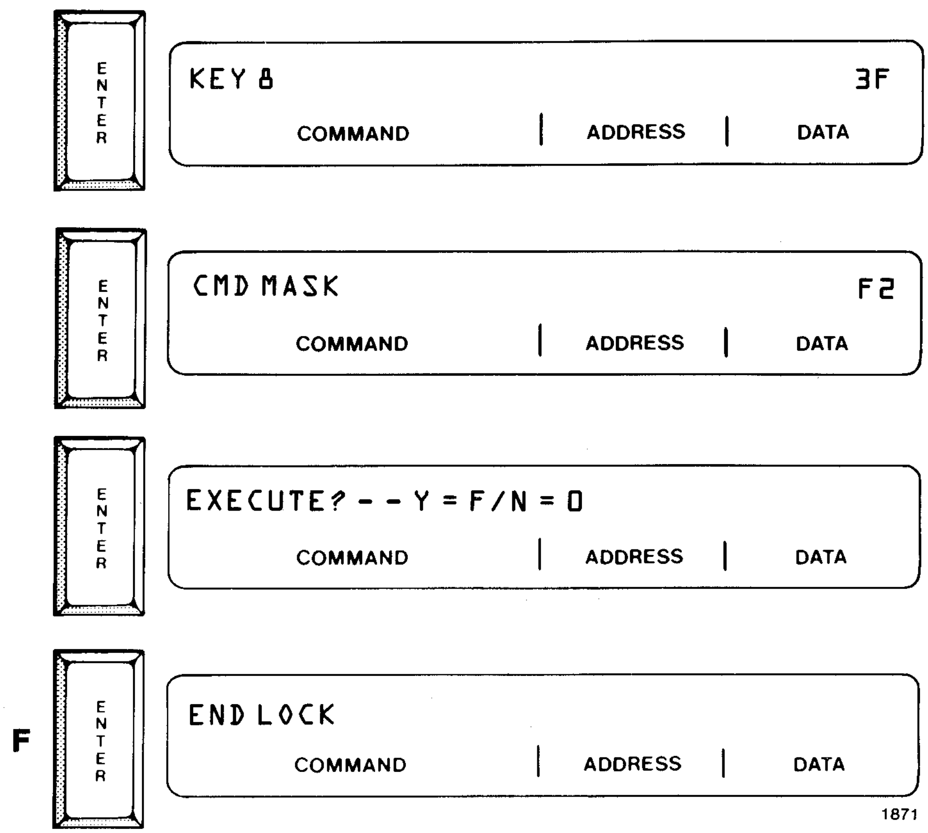
1869



1870



1870



1871

The EPROM is now locked and cannot be unlocked without the identity code.

APPENDIX A
ERROR MESSAGES AND CONDITIONS

Self-Diagnostic Errors

Self-test diagnostics are run when the universal programmer is powered on. Errors indicate hardware failures in the universal programmer. Refer to the Service Information section in the Preface for information about service for hardware failures. For any failing tests, the corresponding error messages are displayed on the iUP-201A alphanumeric display.

POWER SUPPLY FAILURE
The internal voltages in the iUP-200A/201A are not within correct tolerances.

MOTHER BOARD FAILURE
One of the components of the iUP-200A/201A mother board did not pass the diagnostics. The bad component could be the 8085 CPU, the ROM firmware, the RAM, or the timer.

KEY/RAM BOARD FAILURE
The circuit board containing the iUP-201A keyboard and URAM did not pass the diagnostics.

If your universal programmer is model iUP-201A, then these messages are displayed on the off-line alphanumeric display because the iUP-201A initializes in off-line mode. If your universal programmer is model iUP-200A, these messages appear on the host-development terminal only after you enter the TYPE command. The iPPS must be running before you turn on the iUP-200A.

If your universal programmer passes all the self-diagnostic tests, the iUP-201A displays the following in the command, address, and data fields of the alphanumeric display:

IUP READY   000000   55

Off-line Errors

With the model iUP-201A, the following error messages can appear on the LED display to indicate an error during off-line functions:

MODULE CHECKSUM ERROR
A check-sum error was detected. The load function (ROM TO RAM key), verify function (VER key), and program PROM function (PROG key) all compute a check-sum on the data stored in the personality module's firmware ROM.

MODULE NOT INSTALLED
A personality module is not installed.

CHECK PROM INSTALLATION
A PROM device is not installed, or the PROM is installed improperly. Refer to Chapter 2 for instructions on installing the PROM.

USE MASTER SOCKET
The PROM device is not installed in the master socket, or the PROM device is installed improperly in the master socket.

VER ERR @ aaaaaa dd

The verify function (VER key) displays this error message if it detects a mismatch of data between the RAM and PROM. Refer to the VER function key description in Chapter 3.

BLANKCK ERR aaaaaa dd

The program function (PROG key) displays this error message if the PROM fails the blank PROM test before programming begins. Refer to the PROG function key description in Chapter 3.

DEVICE WILL NOT PROGRAM

The program function (PROG key) displays this error message if the pre-programmed bits in the PROM do not match the corresponding RAM bits and the PROM cannot be programmed. Refer to the PROG function key description in Chapter 3.

PROGRAM ERR aaaaaa dd

The program function (PROG key) displays this error message if an error occurs during PROM programming and the PROM cannot be programmed to match the data in the RAM. Refer to the PROG function key description in Chapter 3.

ADDRESS OUT OF RANGE

The address keyed into the address field is greater than the largest valid address for the currently selected PROM device type. This error is also displayed during the FILL function when the ending address is less than the starting address or is out of range.

RANGE EXCEEDS BUFFER

For PROMs greater than 32K bytes, either the address range entered generates more than thirty-one 1K blocks or the address range exceeds the PROM size.

EDIT BUFFER FULL

Only thirty-one 1K blocks can be edited in a PROM larger than 32K.

ILLEGAL BAUD RATE

The baud rate keyed in is not a standard baud rate within the range of 110 to 9600. Enter the correct baud rate: 110, 150, 300, 600, 1200, 2400, 4800, or 9600.

CHECK SUM ERROR

The check-sum calculated for a record being read from a file does not match the check-sum in the file.

ILLEGAL FILE TYPE

The data being downloaded from a file in off-line mode does not match Intel 8080 hexadecimal file requirements.

ILLEGAL OPERATION

The operation entered is not supported on this PROM.

ILLEGAL PARAMETER VALUE

The value entered is not a valid parameter value.

ALREADY LOCKED

The EPROM is blank or already locked.

LOCK FAILED

The EPROM cannot be locked.

FAILED @ XXXXXX dd

The lock failed at location XXXXXX.

APPENDIX B
HOST SERIAL COMMAND PROTOCOL

This appendix describes the command protocols for connecting the universal programmer to a host system other than an Intel development system.

Universal Programmer to Host System Interface

The universal programmer communicates with the host system using a serial interface. The processor in the universal programmer automatically sets the receive and transmit baud rates to those of the host system. The automatic baud rate adjustment selects one of the following rates between 110 and 9600 baud: 110, 150, 300, 600, 1200, 2400, 4800, and 9600.

Communication Protocol

The universal programmer unit conforms to the E.I.A. Standard RS-232-C for data communication equipment. The RS-232 interface signals implemented in the hardware are shown in Table B-1.

Transmit Formats

The types of information transferred using the serial interface are control commands and data.

The control commands are sent from the host development system to the universal programmer to control PROM programming, resetting, etc. These commands must be generated by host software such as the iPPS software that runs on an Intellec development system. A control command summary is shown in Table B-2.

Data is also transmitted with some of the controls commands.

Most of the commands are followed by a command status code response that is sent back to the host development system by the universal programmer.

Table B-1 RS-232 Interface Signals

Pin	Signal (from host)	Description
1	CGND	Chassis ground
2	XTD	Transmitter data
3	RCD	Receiver data
7	SGND	Signal ground

Table B-2 ASCII Command Characters

ASCII Character	Description
A	ACKNOWLEDGE. The A command tells the universal programmer that the software running on the host system is ready to accept data. The A command is used for handshaking.
B	UNPROGRAMMED PROM TEST. The B command causes the universal programmer to perform a test that determines if the PROM is unprogrammed (blank) and reflects the outcome of the test in the command status code returned to the host.
C	CLEAR. The C command resets any operation in progress and clears all universal programmer flags. This command resets the personality module and the device select pointer. The C command must be the first command issued to the universal programmer because it calculates the baud rate.
D	DOWNLOAD. The D command sends data from the host system to the local RAM in the universal programmer. The user RAM begins at absolute location 8000H.
E	EXECUTE. The E command causes the universal programmer to transfer control to the personality module specific code at the processor's local RAM address 7020H.
F	FINISH. The F command terminates read (R) or program (P) commands. It clears the personality module control bits and turns off the personality module power supplies. You can use this command to re-establish the baud rate.
L	LOCK. The L command transfers user-entered parameters to the universal programmer (when needed) and locks the EPROM device from unauthorized access. A sequence of L commands are executed for each KEYLOCK command. The reply requests more data, until all parameters are transferred.
P	PROGRAM. The P command programs a specified PROM location with data supplied by the host.
R	READ PROM. The R command requests the universal programmer to transmit the contents of specified PROM locations.
S	SELECT PROM TYPE. The S command selects the next PROM type that is valid for a particular personality module. Some personality modules can program more than one type of PROM. Each time this command is issued by the host development system, the universal programmer steps once through a list of valid PROMs for the personality module installed.
U	UPLOAD. The U command requests that the universal programmer transmit data from the local RAM to the host system buffer. The user RAM begins at absolute location 8000H.
V	VERIFY. The V command requests that the universal programmer transmit the currently selected PROM type and PROM parameters to the host system.

Two different command formats are used to transmit information from the host to the universal programmer: commands with parameters and commands without parameters.

The following are commands with parameters.

Download (D) command
Lock (L) command
Program (P) command
Read (R) command
Upload (U) command

The P and D commands require up to 67 additional bytes, three bytes of address and up to 64 bytes of data. The R and U commands require only four additional bytes, three bytes of address and a byte that indicates the number of data bytes (which can be up to 64) to be read (0 bytes is illegal). The L command requires two additional bytes, one for sequence number and one for the user-entered data requested by the previous lock acknowledge message. The first L command contains a sequence number of 0; successive commands echo the sequence number of the previous reply.

Both command formats (with or without parameters) require a 20H start byte followed by a byte count prefix of the number of bytes to follow (i.e., ASCII CMD CHAR through CHECK SUM MSB). The start byte and byte count prefix precede the control command character (the ASCII character for the command shown in Table B-2).

Both command formats (with or without parameters) are followed by a 2-byte check-sum suffix. The check-sum is the 2's complement of the binary sum of the previous bytes in the record being transmitted. This check-sum does not include the start byte or the check-sum itself.

Figure B-1 shows both host transmit formats.

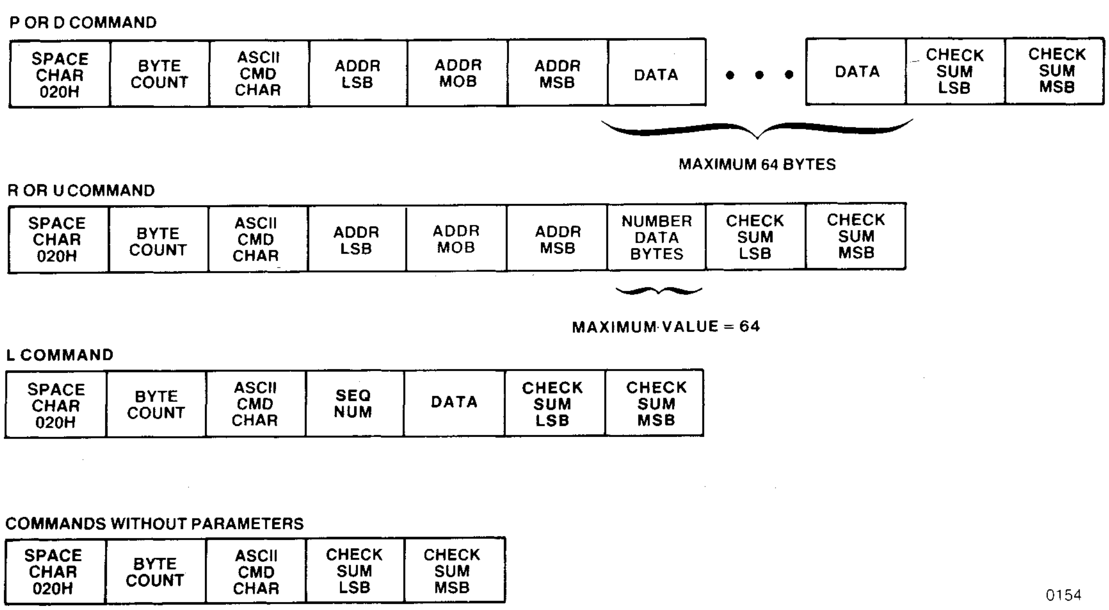


Figure B-1 Host Transmit Formats

Acknowledge Formats

The universal programmer acknowledges a host transmission (except for the host acknowledge command and the host execute command) by sending a block of information (acknowledge block) back to the host. The host software must be able to interpret the acknowledge block correctly. For example, if a message is sent back (status code 21H indicates that a message follows), the host software must receive the message data and display the message on the terminal.

The information contained in the acknowledge block varies depending on the command requested by the host and whether or not any check-sum errors were detected.

Every acknowledge block sent to the host contains a byte count prefix (i.e., number of bytes in STATUS CODE through CHECK SUM MSB), a status byte, an echo of the command character, and a check-sum suffix.

The echo of the command character is a repeat of the ASCII character for the command being acknowledged.

The check-sum is the 2's complement of the binary sum of the previous bytes in the record (including the byte count).

Only the first byte of the block is initially sent to the host; the remainder is transmitted after the host responds with an acknowledge command. If the host responds with any command besides acknowledge, the entire block awaiting transmission in the universal programmer is lost, and the universal programmer responds to the new command issued by the host.

The status word is coded as shown in Table B-3. The codes are in hexadecimal.

The entries in Table B-4 are logically ORed with the appropriate value in Table B-3 (00H or C0H) to produce the status byte.

Table B-3 Status Word Codes

Code	Description
00	OPERATION COMPLETED, NO ERRORS. The 00H status byte indicates that transmission completed correctly for the iUP-200A universal programmer.
80	OPERATION COMPLETED, NO ERRORS. The 80H status byte indicates that transmission completed correctly for the iUP-201A universal programmer with the 16K URAM installed.
C0	OPERATION COMPLETED, NO ERRORS. The C0H status byte indicates that transmission completed correctly for the iUP-201A universal programmer with the 32K URAM installed.

Table B-4 Status Byte Codes

Code	Description
01	PROGRAMMING ERROR. The 01H status byte occurs in response to the host program (P) command if a location cannot be programmed with the specified data.
02	CHECK SUM ERROR. The 02H status byte indicates that the command record received by the universal programmer contains an error.
03	MODULE NOT INSTALLED. The 03H status byte indicates that an operation was attempted without a personality module installed.
04	DEVICE INSTALLED IMPROPERLY. The 04H status byte indicates that the device to be programmed or read is either installed improperly or is not present at all. Note that not all personality modules can detect whether a device is improperly installed.
05	UNPROGRAMMED ERROR. The 05H status byte is sent by the universal programmer in response to the UNPROGRAMMED PROM TEST (B) command if the PROM in question has been programmed.
06	MODULE UPLOAD ERROR. The 06H status byte indicates that the personality module specific code cannot be correctly loaded into address 7020H of universal programmer memory.
07	ERROR. The 07H status byte indicates that a system error occurred and the command should be aborted.
08	FORMAT ERROR. The 08H status byte indicates that the last transmission to the universal programmer did not meet the host-to-universal programmer command protocol.
09	COMMAND NOT SUPPORTED BY DEVICE. The 09H status byte indicates that the KEYLOCK command does not support the device type selected; the command will be aborted.
0A	DEVICE IS LOCKED. The 0A status byte indicates that the user tried to lock a device that was already locked or was blank and appeared locked; the command will be aborted.
0B	LOCK FAILED. The 0B status byte indicates that an error occurred while attempting to program the security bit to lock the device; the command will be aborted.
0C	LOCK FAILED AT <parameter>. The 0C status byte indicates that an error occurred while attempting to program lock-related parameters into an authenticated EPROM.
0D	ILLEGAL PARAMETER VALUE. The 0D status byte indicates that a specified parameter value is illegal.
0E & 0F	UNDEFINED.
10 - 1F	SELF-TEST ERROR. One of these status bytes can be sent in response to the clear (C) command if a universal programmer hardware failure is detected. The assignment for these error codes is as follows: 10H MOTHERBOARD FAILURE 11H KEY/EXPANSION RAM FAILURE 12H POWER SUPPLY FAILURE 13H - 1FH UNDEFINED

Table B-4 Status Byte Codes (continued)

Code	Description
20	REPEAT OPERATION. The 20H status byte requests that the host software transmit the full data buffer again. You can use this feature to program certain PROMs.
21	MESSAGE FOLLOWS. The 21H status byte indicates that a message follows which can be displayed on the host terminal by the host software. Following this message, another status message will be sent in response to the command.
22	RESET DEAD-MAN TIMER. The 22H status byte indicates that the host system timer should be reset to prevent the host system from timing out.
23 - FF	Not used.

The amount of information the universal programmer sends back to the host depends on the command. The following host commands need only the status byte and check-sum:

CLEAR (C)
DOWNLOAD (D)
FINISH (F)
LOCK (L) with error condition requiring no additional data
PROGRAM (P) without error condition
SELECT PROM TYPE (S)
UNPROGRAMMED PROM TEST (B)

The following commands need additional information:

LOCK (L)
PROGRAM (P) with error condition
READ (R)
UPLOAD (U)
VERIFY PROM TYPE (V)

Figure B-2 shows all universal programmer acknowledge formats.

The ADDRESS RANGE bytes shown in Figure B-2 indicate (with a 24-bit number) the total available address space for a selected PROM type.

The DATA INFO byte in the V command acknowledge is defined as follows:

BIT 0 - BIT 5 These bits indicate the word length of the selected PROM type (e.g., 01000 = 8 bits long).

BIT 6 This bit indicates whether the selected PROM type is programmed high or low. A zero in this bit position indicates that the PROM type can be programmed only with lows (zeros). A one in this bit position indicates that only highs (ones) can be programmed into the PROM type.

BIT 7 Not used.

The REQUEST FLAG byte in the L command acknowledge is defined as follows:

BIT 0	Not used.
BIT 1	Requests the iPPS software to prompt for a byte value.
BIT 2	Requests the iPPS software to query the user for a yes/no decision.
BITS 3-7	Not used.

The DATA byte in the L command acknowledge contains the current value associated with the parameter described by the ASCII characters.

The SEQUENCE number in the L command acknowledge is the code assigned by the universal programmer by which it identifies the parameter value being requested. A sequence number of 0 signals the end of the LOCK command to the iPPS software.

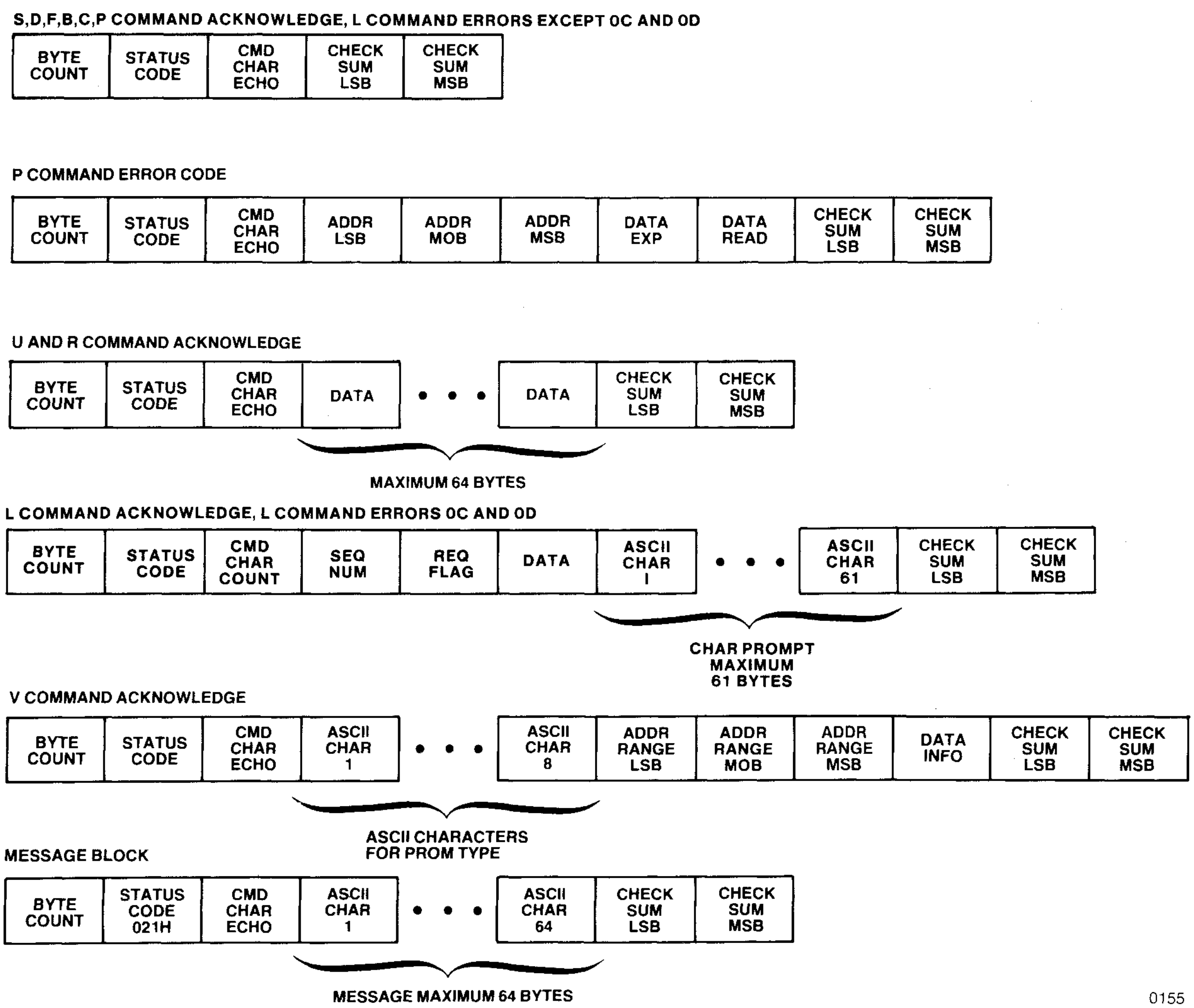


Figure B-2 Universal Programmer Acknowledge Formats

Data Link Format

All records passed between the host and the universal programmer are sent a byte at a time. Each byte received by the universal programmer must be prefixed by a start bit and suffixed with at least one stop bit. All bytes transmitted by the universal programmer are prefixed with one start bit and suffixed with one stop bit. Figure B-3 shows the typical format for passing bytes over the serial data link.

On power-up or reset or when the serial data link baud rate is changed, the host system must transmit a Clear (C) command. The C command is used by the universal programmer processor to automatically set the baud rate. If the baud rate is changed after it is initially identified, three sequential check-sum errors occur before the new rate is identified. This means that four clear commands must be sent before the new rate is identified.

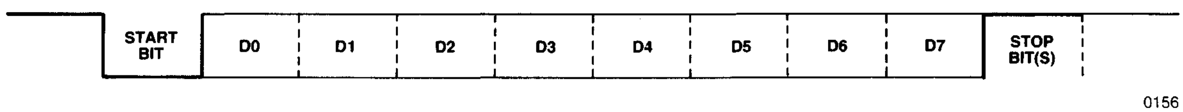


Figure B-3 Serial Byte Format

APPENDIX C
REFERENCE TABLES

Table C-1 shows hexadecimal to decimal and decimal to hexadecimal conversion. To find the decimal equivalent of a hexadecimal number, locate the hexadecimal number in the correct position and note the decimal equivalent. Add the decimal numbers.

To find the hexadecimal equivalent of a decimal number, perform the following steps:

1. Find the largest decimal number that does not exceed your starting number. Record its hexadecimal equivalent, including place holders (zeros).
2. Subtract the decimal number you found from the starting decimal number. The difference is your new starting number. If the new starting number is greater than zero, repeat steps one and two.
3. Add the hexadecimal equivalents.

Table C-2 contains base conversions for decimal, binary, hexadecimal, and octal.

Table C-3 contains powers of two.

Table C-4 shows conversions between powers of 2 and 16.

Table C-5 contains powers of 16.

Table C-6 lists ASCII code.

Table C-7 defines ASCII control codes.

Table C-8 translates ASCII code to binary.

Table C-1 Hexadecimal to Decimal Conversion

Most Significant Byte				Least Significant Byte			
Digit 4		Digit 3		Digit 2		Digit 1	
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC
0	0	0	0	0	0	0	0
1	4 096	1	256	1	16	1	1
2	8 192	2	512	2	32	2	2
3	12 288	3	768	3	48	3	3
4	16 384	4	1 024	4	64	4	4
5	20 480	5	1 280	5	80	5	5
6	24 576	6	1 536	6	96	6	6
7	28 672	7	1 792	7	112	7	7
8	32 768	8	2 048	8	128	8	8
9	36 864	9	2 304	9	144	9	9
A	40 960	A	2 560	A	160	A	10
B	45 056	B	2 816	B	176	B	11
C	49 152	C	3 072	C	192	C	12
D	53 248	D	3 328	D	208	D	13
E	57 344	E	3 548	E	224	E	14
F	61 440	F	3 840	F	240	F	15
7654		3210		7654		3210	
Byte				Byte			

Table C-2 Base Conversions

DEC	BIN	HEX	OCT	DEC	BIN	HEX	OCT
0	0000 0000	00	000	51	0011 0011	33	063
1	0000 0001	01	001	52	0011 0100	34	064
2	0000 0010	02	002	53	0011 0101	35	065
3	0000 0011	03	003	54	0011 0110	36	066
4	0000 0100	04	004	55	0011 0111	37	067
5	0000 0101	05	005	56	0011 1000	38	070
6	0000 0110	06	006	57	0011 1001	39	071
7	0000 0111	07	007	58	0011 1010	3A	072
8	0000 1000	08	010	59	0011 1011	3B	073
9	0000 1001	09	011	60	0011 1100	3C	074
10	0000 1010	0A	012	61	0011 1101	3D	075
11	0000 1011	0B	013	62	0011 1110	3E	076
12	0000 1100	0C	014	63	0011 1111	3F	077
13	0000 1101	0D	015	64	0100 0000	40	100
14	0000 1110	0E	016	65	0100 0001	41	101
15	0000 1111	0F	017	66	0100 0010	42	102
16	0001 0000	10	020	67	0100 0011	43	103
17	0001 0001	11	021	68	0100 0100	44	104
18	0001 0010	12	022	69	0100 0101	45	105
19	0001 0011	13	023	70	0100 0110	46	106
20	0001 0100	14	024	71	0100 0111	47	107
21	0001 0101	15	025	72	0100 1000	48	110
22	0001 0110	16	026	73	0100 1001	49	111
23	0001 0111	17	027	74	0100 1010	4A	112
24	0001 1000	18	030	75	0100 1011	4B	113
25	0001 1001	19	031	76	0100 1100	4C	114
26	0001 1010	1A	032	77	0100 1101	4D	115
27	0001 1011	1B	033	78	0100 1110	4E	116
28	0001 1100	1C	034	79	0100 1111	4F	117
29	0001 1101	1D	035	80	0101 0000	50	120
30	0001 1110	1E	036	81	0101 0001	51	121
31	0001 1111	1F	037	82	0101 0010	52	122
32	0010 0000	20	040	83	0101 0011	53	123
33	0010 0001	21	041	84	0101 0100	54	124
34	0010 0010	22	042	85	0101 0101	55	125
35	0010 0011	23	043	86	0101 0110	56	126
36	0010 0100	24	044	87	0101 0111	57	127
37	0010 0101	25	045	88	0101 1000	58	130
38	0010 0110	26	046	89	0101 1001	59	131
39	0010 0111	27	047	90	0101 1010	5A	132
40	0010 1000	28	050	91	0101 1011	5B	133
41	0010 1001	29	051	92	0101 1100	5C	134
42	0010 1010	2A	052	93	0101 1101	5D	135
43	0010 1011	2B	053	94	0101 1110	5E	136
44	0010 1100	2C	054	95	0101 1111	5F	137
45	0010 1101	2D	055	96	0110 0000	60	140
46	0010 1110	2E	056	97	0110 0001	61	141
47	0010 1111	2F	057	98	0110 0010	62	142
48	0011 0000	30	060	99	0110 0011	63	143
49	0011 0001	31	061	100	0110 0100	64	144
50	0011 0010	32	062	101	0110 0101	65	145

Table C-2 Base Conversions (continued)

DEC	BIN	HEX	OCT	DEC	BIN	HEX	OCT
102	0110 0110	66	146	153	1001 1001	99	231
103	0110 0111	67	147	154	1001 1010	9A	232
104	0110 1000	68	150	155	1001 1011	9B	233
105	0110 1001	69	151	156	1001 1100	9C	234
106	0110 1010	6A	152	157	1001 1101	9D	235
107	0110 1011	6B	153	158	1001 1110	9E	236
108	0110 1100	6C	154	159	1001 1111	9F	237
109	0110 1101	6D	155	160	1010 0000	A0	240
110	0110 1110	6E	156	161	1010 0001	A1	241
111	0110 1111	6F	157	162	1010 0010	A2	242
112	0111 0000	70	160	163	1010 0011	A3	243
113	0111 0001	71	161	164	1010 0100	A4	244
114	0111 0010	72	162	165	1010 0101	A5	245
115	0111 0011	73	163	166	1010 0110	A6	246
116	0111 0100	74	164	167	1010 0111	A7	247
117	0111 0101	75	165	168	1010 1000	A8	250
118	0111 0110	76	166	169	1010 1001	A9	251
119	0111 0111	77	167	170	1010 1010	AA	252
120	0111 1000	78	170	171	1010 1011	AB	253
121	0111 1001	79	171	172	1010 1100	AC	254
122	0111 1010	7A	172	173	1010 1101	AD	255
123	0111 1011	7B	173	174	1010 1110	AE	256
124	0111 1100	7C	174	175	1010 1111	AF	257
125	0111 1101	7D	175	176	1011 0000	B0	260
126	0111 1110	7E	176	177	1011 0001	B1	261
127	0111 1111	7F	177	178	1011 0010	B2	262
128	1000 0000	80	200	179	1011 0011	B3	263
129	1000 0001	81	201	180	1011 0100	B4	264
130	1000 0010	82	202	181	1011 0101	B5	265
131	1000 0011	83	203	182	1011 0110	B6	266
132	1000 0100	84	204	183	1011 0111	B7	267
133	1000 0101	85	205	184	1011 1000	B8	270
134	1000 0110	86	206	185	1011 1001	B9	271
135	1000 0111	87	207	186	1011 1010	BA	272
136	1000 1000	88	210	187	1011 1011	BB	273
137	1000 1001	89	211	188	1011 1100	BC	274
138	1000 1010	8A	212	189	1011 1101	BD	275
139	1000 1011	8B	213	190	1011 1110	BE	276
140	1000 1100	8C	214	191	1011 1111	BF	277
141	1000 1101	8D	215	192	1100 0000	B0	300
142	1000 1110	8E	216	193	1100 0001	C1	301
143	1000 1111	8F	217	194	1100 0010	C2	302
144	1001 0000	90	220	195	1100 0011	C3	303
145	1001 0001	91	221	196	1100 0100	C4	304
146	1001 0010	92	222	197	1100 0101	C5	305
147	1001 0011	93	223	198	1100 0110	C6	306
148	1001 0100	94	224	199	1100 0111	C7	307
149	1001 0101	95	225	200	1100 1000	C8	310
150	1001 0110	96	226	201	1100 1001	C9	311
151	1001 0111	97	227	202	1100 1010	CA	312
152	1001 1000	98	230	203	1100 1011	CB	313

Table C-2 Base Conversions (continued)

DEC	BIN	HEX	OCT	DEC	BIN	HEX	OCT
204	1100 1100	CC	314	230	1110 0110	E6	346
205	1100 1101	CD	315	231	1110 0111	E7	347
206	1100 1110	CE	316	232	1110 1000	E8	350
207	1100 1111	CF	317	233	1110 1001	E9	351
208	1101 0000	D0	320	234	1110 1010	EA	352
209	1101 0001	D1	321	235	1110 1011	EB	353
210	1101 0010	D2	322	236	1110 1100	EC	354
211	1101 0011	D3	323	237	1110 1101	ED	355
212	1101 0100	D4	324	238	1110 1110	EE	356
213	1101 0101	D5	325	239	1110 1111	EF	357
214	1101 0110	D6	326	240	1111 0000	F0	360
215	1101 0111	D7	327	241	1111 0001	F1	361
216	1101 1000	D8	330	242	1111 0010	F2	362
217	1101 1001	D9	331	243	1111 0011	F3	363
218	1101 1011	DA	332	244	1111 0100	F4	364
219	1101 1011	DB	333	245	1111 0101	F5	365
220	1101 1100	DC	334	246	1111 0110	F6	366
221	1101 1101	DD	335	247	1111 0111	F7	367
222	1101 1110	DE	336	248	1111 1000	F8	370
223	1101 1111	DF	337	249	1111 1001	F9	371
224	1110 0000	E0	340	250	1111 1010	FA	372
225	1110 0001	E1	341	251	1111 1011	FB	373
226	1110 0010	E2	342	252	1111 1100	FC	374
227	1110 0011	E3	343	253	1111 1101	FD	375
228	1110 0100	E4	344	254	1111 1110	FE	376
229	1110 0101	E5	345	255	1111 1111	FF	377

Table C-3 Powers of Two

2^n	n
256	8
512	9
1 024	10
2 048	11
4 096	12
8 192	13
16 384	14
32 768	15
65 536	16
131 072	17
262 144	18
524 288	19
1 048 576	20
2 097 152	21
4 194 304	22
8 388 608	23
16 777 216	24

Table C-4 Conversion Between Powers of Two and Sixteen

2^m	=	16^n
2^0	=	16^0
2^4	=	16^1
2^8	=	16^2
2^{12}	=	16^3
2^{16}	=	16^4
2^{20}	=	16^5
2^{24}	=	16^6
2^{28}	=	16^7
2^{32}	=	16^8
2^{36}	=	16^9
2^{40}	=	16^{10}
2^{44}	=	16^{11}
2^{48}	=	16^{12}
2^{52}	=	16^{13}
2^{56}	=	16^{14}
2^{60}	=	16^{15}
2^{64}	=	16^{16}

Table C-5 Powers of Sixteen

16^n	n
1	0
16	1
256	2
4 096	3
65 536	4
1 048 576	5
16 777 216	6
268 435 456	7
4 294 967 296	8
68 719 476 736	9
1 099 511 627 776	10
17 592 186 044 416	11
281 474 976 710 656	12
4 503 599 627 370 496	13
72 057 594 037 927 936	14
1 152 921 504 606 846 976	15

Table C-6 ASCII Code List

Decimal	Octal	Hexadecimal	Character	Decimal	Octal	Hexadecimal	Character
0	000	00	NUL	32	040	20	SP
1	001	01	SOH	33	041	21	!
2	002	02	STX	34	042	22	"
3	003	03	ETX	35	043	23	#
4	004	04	EOT	36	044	24	$
5	005	05	ENQ	37	045	25	%
6	006	06	ACK	38	046	26	&
7	007	07	BEL	39	047	27	'
8	010	08	BS	40	050	28	(
9	011	09	HT	41	050	29	)
10	012	0A	LF	42	052	2A	*
11	013	0B	VT	43	053	2B	+
12	014	0C	FF	44	054	2C	,
13	015	0D	CR	45	055	2D	-
14	016	0E	SO	46	056	2E	.
15	017	0F	SI	47	057	2F	/
16	020	10	DLE	48	060	30	0
17	021	11	DC1	49	061	31	1
18	022	12	DC2	50	062	32	2
19	023	13	DC3	51	063	33	3
20	024	14	DC4	52	064	34	4
21	025	15	NAK	53	065	35	5
22	026	16	SYN	54	066	36	6
23	027	17	ETB	55	067	37	7
24	030	18	CAN	56	070	38	8
25	031	19	EM	57	071	39	9
26	032	1A	SUB	58	072	3A	:
27	033	1B	ESC	59	073	3B	;
28	034	1C	FS	60	074	3C	<
29	035	1D	GS	61	075	3D	=
30	036	1E	RS	62	076	3E	>
31	037	1F	US	63	077	3F	?

Table C-6 ASCII Code List (continued)

<table>
<tr><th>Decimal</th><th>Octal</th><th>Hexadecimal</th><th>Character</th><th>Decimal</th><th>Octal</th><th>Hexadecimal</th><th>Character</th></tr>
<tr><td>64</td><td>100</td><td>40</td><td>@</td><td>96</td><td>140</td><td>60</td><td>`</td></tr>
<tr><td>65</td><td>101</td><td>41</td><td>A</td><td>97</td><td>141</td><td>61</td><td>a</td></tr>
<tr><td>66</td><td>102</td><td>42</td><td>B</td><td>98</td><td>142</td><td>62</td><td>b</td></tr>
<tr><td>67</td><td>103</td><td>43</td><td>C</td><td>99</td><td>143</td><td>63</td><td>c</td></tr>
<tr><td>68</td><td>104</td><td>44</td><td>D</td><td>100</td><td>144</td><td>64</td><td>d</td></tr>
<tr><td>9</td><td>105</td><td>45</td><td>E</td><td>101</td><td>145</td><td>65</td><td>e</td></tr>
<tr><td>70</td><td>106</td><td>46</td><td>F</td><td>102</td><td>146</td><td>66</td><td>f</td></tr>
<tr><td>71</td><td>107</td><td>47</td><td>G</td><td>103</td><td>147</td><td>67</td><td>g</td></tr>
<tr><td>72</td><td>110</td><td>48</td><td>H</td><td>104</td><td>150</td><td>68</td><td>h</td></tr>
<tr><td>73</td><td>111</td><td>49</td><td>I</td><td>105</td><td>151</td><td>69</td><td>i</td></tr>
<tr><td>74</td><td>112</td><td>4A</td><td>J</td><td>106</td><td>152</td><td>6A</td><td>j</td></tr>
<tr><td>75</td><td>113</td><td>4B</td><td>K</td><td>107</td><td>153</td><td>6B</td><td>k</td></tr>
<tr><td>76</td><td>114</td><td>4C</td><td>L</td><td>108</td><td>154</td><td>6C</td><td>l</td></tr>
<tr><td>77</td><td>115</td><td>4D</td><td>M</td><td>109</td><td>155</td><td>6D</td><td>m</td></tr>
<tr><td>78</td><td>116</td><td>4E</td><td>N</td><td>110</td><td>156</td><td>6E</td><td>n</td></tr>
<tr><td>79</td><td>117</td><td>4F</td><td>O</td><td>111</td><td>157</td><td>6F</td><td>o</td></tr>
<tr><td>80</td><td>120</td><td>50</td><td>P</td><td>112</td><td>160</td><td>70</td><td>p</td></tr>
<tr><td>81</td><td>121</td><td>51</td><td>Q</td><td>113</td><td>161</td><td>71</td><td>q</td></tr>
<tr><td>82</td><td>122</td><td>52</td><td>R</td><td>114</td><td>162</td><td>72</td><td>r</td></tr>
<tr><td>83</td><td>123</td><td>53</td><td>S</td><td>115</td><td>163</td><td>73</td><td>s</td></tr>
<tr><td>84</td><td>124</td><td>54</td><td>T</td><td>116</td><td>164</td><td>74</td><td>t</td></tr>
<tr><td>85</td><td>125</td><td>55</td><td>U</td><td>117</td><td>165</td><td>75</td><td>u</td></tr>
<tr><td>86</td><td>126</td><td>56</td><td>V</td><td>118</td><td>166</td><td>76</td><td>v</td></tr>
<tr><td>87</td><td>127</td><td>57</td><td>W</td><td>119</td><td>167</td><td>77</td><td>w</td></tr>
<tr><td>88</td><td>130</td><td>58</td><td>X</td><td>120</td><td>170</td><td>78</td><td>x</td></tr>
<tr><td>89</td><td>131</td><td>59</td><td>Y</td><td>121</td><td>171</td><td>79</td><td>y</td></tr>
<tr><td>90</td><td>132</td><td>5A</td><td>Z</td><td>122</td><td>172</td><td>7A</td><td>z</td></tr>
<tr><td>91</td><td>133</td><td>5B</td><td>[</td><td>123</td><td>173</td><td>7B</td><td>{</td></tr>
<tr><td>92</td><td>134</td><td>5C</td><td>\</td><td>124</td><td>174</td><td>7C</td><td>|</td></tr>
<tr><td>93</td><td>135</td><td>5D</td><td>]</td><td>125</td><td>175</td><td>7D</td><td>}</td></tr>
<tr><td>94</td><td>136</td><td>5E</td><td>^</td><td>126</td><td>176</td><td>7E</td><td>~</td></tr>
<tr><td>95</td><td>137</td><td>5F</td><td>_</td><td>127</td><td>177</td><td>7F</td><td>DEL</td></tr>
</table>

Table C-7 ASCII Control Code Definition

Abbreviation	Meaning	Decimal Code
NUL	NULL Character	0
SOH	Start of Heading	1
STX	Start of Text	2
ETX	End of Text	3
EOT	End of Transmission	4
ENQ	Inquiry	5
ACK	Acknowledge	6
BEL	Bell	7
BS	Backspace	8
HT	Horizontal Tabulation	9
LF	Line Feed	10
VT	Vertical Tabulation	11
FF	Form Feed	12
CR	Carriage Return	13
SO	Shift Out	14
SI	Shift In	15
DLE	Data Link Escape	16
DC1	Device Control 1	17
DC2	Device Control 2	18
DC3	Device Control 3	19
DC4	Device Control 4	20
NAK	Negative Acknowledge	21
SYN	Synchronous Idle	22
ETB	End of Transmission Block	23
CAN	Cancel	24
EM	End of Medium	25
SUB	Substitute	26
ESC	Escape	27
FS	File Separator	28
GS	Group Separator	29
RS	Record Separator	30
US	Unit Separator	31
SP	Space	32
DEL	Delete	127

Table C-8 ASCII Code in Binary

MSB / LSB	0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
0 0000	NUL	DLE	SP	0	@	P	`	p
1 0001	SOH	DC1	!	1	A	Q	a	q
2 0010	STX	DC2	"	2	B	R	b	r
3 0011	ETX	DC3	#	3	C	S	c	s
4 0100	EOT	DC4	$	4	D	T	d	t
5 0101	ENQ	NAK	%	5	E	U	e	u
6 0110	ACK	SYN	&	6	F	V	f	v
7 0111	BEL	ETB	'	7	G	W	g	w
8 1000	BS	CAN	(	8	H	X	h	x
9 1001	HT	EM	)	9	I	Y	i	y
A 1010	LF	SUB	*	:	J	Z	j	z
B 1011	VT	ESC	+	;	K	[	k	{
C 1100	FF	FS	,	<	L	\	l	|
D 1101	CR	GS	-	=	M	]	m	}
E 1110	SO	RS	.	>	N	^	n	~
F 1111	SI	VS	/	?	O	_	o	DEL

RS-232 TO 20 ma CONVERTER SCHEMATIC

Refer to Figure D-1 to build your own RS-232 to 20ma converter. Keep the connections between MDS 800 and the converter as short as possible because the power supply is derived from the Intellec 800 through the TTY connector.

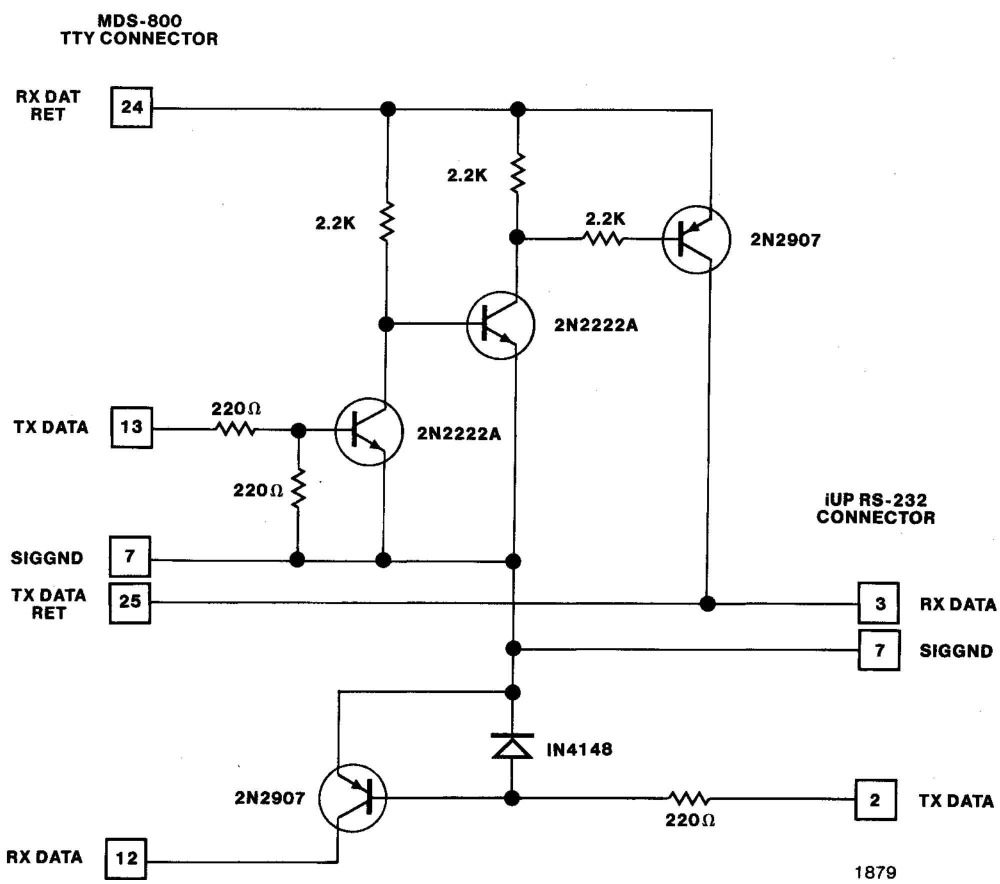


Figure D-1 RS-232 to 20ma Converter Schematic

INDEX

REQUEST FOR READER'S COMMENTS

Intel's Technical Publications Departments attempt to provide publications that meet the needs of all Intel product users. This form lets you participate directly in the publication process. Your comments will help us correct and improve our publications. Please take a few minutes to respond.

Please restrict your comments to the usability, accuracy, readability, organization, and completeness of this publication. If you have any comments on the product that this publication describes, please contact your Intel representative. If you wish to order publications, contact the Intel Literature Department (see page ii of this manual).

1. Please describe any errors you found in this publication (include page number).

2. Does the publication cover the information you expected or required? Please make suggestions for improvement.

3. Is this the right type of publication for your needs? Is it at the right level? What other types of publications are needed?

4. Did you have any difficulty understanding descriptions or wording? Where?

5. Please rate this publication on a scale of 1 to 5 (5 being the best rating). ______

NAME ______ DATE ______

TITLE ______

COMPANY NAME/DEPARTMENT ______

ADDRESS ______

CITY ______ STATE (COUNTRY) ______ ZIP CODE ______

Please check here if you require a written reply. ☐

WE'D LIKE YOUR COMMENTS ...

This document is one of a series describing Intel products. Your comments on the back of this form will help us produce better manuals. Each reply will be carefully reviewed by the responsible person. All comments and suggestions become the property of Intel Corporation.

NO POSTAGE
NECESSARY
IF MAILED
IN U.S.A.

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 79 BEAVERTON, OR

POSTAGE WILL BE PAID BY ADDRESSEE

Intel Corporation
5200 N.E. Elam Young Parkway.
Hillsboro, Oregon 97123

DSHO Technical Publications

INTEL CORPORATION, 3065 Bowers Avenue, Santa Clara, California 95051 (408) **987-8080**

Printed in U.S.A.

DS-103/3K/0484/WCP
INSTRUMENTATION