Tektronix®
COMMITTED TO EXCELLENCE

PLEASE CHECK FOR CHANGE INFORMATION AT THE REAR OF THIS MANUAL.

4041R04 UTILITY ROMPACK ACCESSORIES KIT (020-0102-00)

INSTRUCTION MANUAL

Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077

Serial Number _____

070-4765-00
Product Group 76

First Printing JUNE 1983

TABLE OF CONTENTS

SECTION 1
INTRODUCTION

The 4041R04 Utility Rompack Accessories Kit provides the user with the capability to transfer files from DC-100 tapes to proms (Programmable ROMs). Doing this requires a (user-provided) prom programmer and the PRMBLD and RMXFER programs, which are contained on a tape provided with the 4041R04 utility rompack accessories kit.

The PRMBLD program builds an image of a DC-100 tape file in memory and stores the image back onto a DC-100 tape (either the same tape as the source file's, or a different tape).

The RMXFER program outputs the prom image produced by PRMBLD to a prom programmer in any of six popular microprocessor formats.

This manual gives complete details on how to use the PRMBLD and RMXFER programs. This information is also included in files PBHELP and RXHELP, provided on the Accessories Kit tape.

SECTION 2
REPLACEABLE MECHANICAL PARTS

Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
	020-0102-00			1		COMPONENT KIT:OPT ACCESSORY	80009	020-0102-00
	020-0101-00			1		.COMPONENT KIT:ROMPACK	80009	020-0101-00
	119-0859-00			2		..EPROM CARRIER:W/BLANK 156-1610-00	80009	119-0859-00
	062-6786-00			1		.SOFTWARE PKG:BLANK TAPE	80009	062-6786-00
	070-4765-00			1		.MANUAL,TECH:INSTR,4041R04	80009	070-4756-00
	670-8255-00			1		.CKT BOARD ASSY:ADAPTER,PROM PROGRAMMER	80009	670-8255-00
	131-2745-00			1		..CONN,RCPT,ELEC:MICROCKT,2 x 12,0.100 CTR	27264	8878-NS-24
	136-0676-00			24		..SOCKET,PIN TERM:FOR 0.016-0.022 DIA PIN	00779	51965-1
	352-0536-00			1		..HOLDER,CONTACT:40 PIN,NYLON	80009	352-0536-00

CROSS INDEX—MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip
00779	AMP, INC.	P.O. BOX 3608	HARRISBURG, PA 17105
27264	MOLEX PRODUCTS CO.	5224 KATRINE, AVE.	DOWNERS, IL 60515
80009	TEKTRONIX, INC.	P O. BOX 500	BEAVERTON, OR 97077

SECTION 3
THE PRMBLD PROGRAM

The PRMBLD program is used to create Prom image files from files on a DC-100 magnetic tape. Once created, the Prom image files are stored on DC-100 magnetic tapes (either the same tape as the source file, or a different tape) and then transferred to Proms by running PRMBLD's companion program, RMXFER.

Requirements for Using the PRMBLD Program

A 4041R04 Utility Rompack must be installed in order to run the PRMBLD program. In addition, the 4041 must have 64 K or more of random-access memory.

How To Use the PRMBLD Program

1) Insert the Prom-builder Utility tape into the tape drive, load the PRMBLD program, and run it.

   If the 4041 does not have a 4041R04 Utility Rompack installed, or does not contain enough memory to hold the Prom image files that will be built, an appropriate message appears on the system console device.

2) The 4041 asks whether the rom images are to be stored on a different tape from the tape the source file(s) are on.

   If you type "Y", the 4041 reads the source file(s), then prompts for a new tape. It then checks the volume label of the new tape to make sure it is different from the source tape, and prompts you to insert a different tape if the new tape is the same as the source.

   If you type "N", the 4041 writes the image files on the same tape as the source files.

3) The 4041 prompts for each file on the source tape that is to be stored in a Prom. Respond with each file's name, one file at a time. When all the files to be stored as Proms have been entered, respond to the file name query with a <cr>.

4) PRMBLD stores Prom images in files with the names ROM1, ROM2, etc. If a file named ROM1, ROM2, etc., already exists on your tape, PRMBLD informs you of the fact and asks you for a new name for the file. It then renames the existing file to the new name, then writes the Prom image into file(s) ROM1, ROM2, etc. If you want the existing file to be overwritten, respond to the file name query with a <cr>.

SECTION 4
THE RMXFER PROGRAM

The RMXFER program is used to transmit Prom image files to a Prom programmer.

Requirements for Using the RMXFER Program

To run the RMXFER program, the 4041 must contain at least 64K bytes of random access memory. The program can run with or without an Option 1 interface; running the program without an Option 01 interface is more cumbersome, however, because a connecting cable must be switched from the system console to the Prom programmer before the Prom images can be transmitted.

In addition, the Prom programmer being used must be capable of programming Motorola MCM68764C35 or MCM68766C35 proms.

How to Use the RMXFER Program

RMXFER prompts the user for various inputs to control the transfer of rom image files to a prom programmer. The sequence of prompts is as follows:

1. "Enter one of the following codes to specify the translation format type:"

Code	Format
1	Tektronix Hexadecimal
2	ASCII-Hex(space)
3	Intel Intellec 8/MDS
4	Motorola Exorciser
5	MOS Technology
6	Fairchild Fairbug

   Code:

   This program supports the six translation formats listed. The user should determine which format applies to the type of prom programmer being used, and enter that code. Examples of the actual formats are given in Appendix A.

2. "Enter baud rate for transmission to prom programmer:"

   Determine which baud rate your prom programmer can handle and enter that rate. Any rate that is permissible for the 4041 COMM driver's BAU parameter may be entered (50, 110, 150, 300, 600, 1200, 2400, 4800, or 9600 baud). If an error is detected, the user is re-prompted for another entry.

3. "Enter the rom file name to be transmitted: "

   Enter the file name of the file that contains a rom image. If the file is not present or does not contain the correct information, the user is informed and re-prompted for another file name.

4. "Set up the prom programmer system to receive the rom image from file xxxxxx and enter <cr> when ready: "

   The file image has been read into the 4041 and is ready for transmission to the prom programmer. Before entering the <cr> be certain that the prom programmer is ready to receive a prom image in the translation format previously specified. (See your prom programmer manual for instructions on how to do this.)

   If the 4041 System Controller being used does not contain Option 1, the following message is displayed on the system console:

   "The system console is being switched over to the front panel. Connect the RS-232 cable to the prom programmer and press "PROCEED""

   At this point the message "Press Proceed" is displayed on the front panel. Once the cable has been connected to the prom programmer, press the "PROCEED" button on the front panel. The messages in steps 5 and 6 will be displayed on the front panel and thermal printer, and not at the terminal.

5. "EVEN prom ready for programming. Enter <cr> when ready for the ODD prom: "

   The even prom image has been transmitted to the prom programmer. Check to make sure that no errors were detected by the prom programmer.

   If there were none, proceed with the programming of the EVEN prom part. When the programming is complete, set up the prom programmer to receive the ODD part for the specified translation format and enter <cr> (from the front panel, press PROCEED).

   If there were errors detected, set up the prom programmer to receive the EVEN prom image once more, then enter any character, followed by <cr> (from the front panel, press PROCEED). The EVEN prom image will be transmitted again.

6. "ODD prom ready for programming."

   The odd prom image has been transmitted to the prom programmer. Check to make sure that no errors were detected by the prom programmer.

   If there were none, proceed with the programming of the ODD prom part.

   If there were errors detected, set up the prom programmer to receive the ODD prom image once more, then enter any character, followed by <cr> (from the front panel, press PROCEED). The ODD prom image will be transmitted again.

APPENDIX A
MICROPROCESSOR FORMATS

This appendix describes the six microprocessor formats into which RMXFER converts PRMBLD output. Information on these formats was obtained from Section 2 of "Translation-Format Package", publication #056-1901, published by the Data I/O Corporation, 1980.

1) TEKTRONIX HEXADECIMAL

Each record begins with a slash ("/"), four characters specifying the address of the first data byte, a two-character byte count, and a two-character checksum (the eight-bit sum of the four-bit hexadecimal values of the six digits that make up the address and byte count).

This is followed by a sequence of data bytes. Each data byte is transmitted as two ASCII hex characters.

Each record terminates with a two-byte checksum (the eight-bit sum, modulo 256, of the four-bit hexadecimal values of the digits that make up the data bytes).

The End-of-File record consists of a slash, a four-character address, a two-character byte count (always equal to 0), and a checksum (the eight-bit sum of the four-bit hexadecimal values of the six digits that make up the transfer address and the byte count).

2) ASCII-HEX (SPACE)

ASCII-HEX format may be used to express either 4-bit or 8-bit data; two ASCII hex characters are used for each item of 8-bit data, one ASCII hex character for each item of 4-bit data.

Successive data items are separated by spaces. Data bytes are addressed sequentially unless preceded by an address.

Addresses begin with the two characters "$A", followed by two to four hex characters, followed by a comma. The Prom programmer skips to the specified address to store the next data byte; succeeding bytes are again stored sequentially.

The start code is a nonprintable STX character (CTRL-B). The end code is a nonprintable ETX character (CTRL-C).

Immediately after the end code has been transmitted, the characters "$S" are transmitted, followed by a four-character checksum (computed as the binary summation of all the data transferred).

3) INTEL INTELLEC 8/MDS

Each data record begins with a colon (":"), a two-character byte count, four characters specifying the address of the first data byte, and a two-character record type (set to 00).

This is followed by a sequence of data bytes, where each data byte is represented by two ASCII hex characters.

Each record terminates with a two-character checksum, computed as the two's complement of the binary sum of the preceding bytes in the record, including the byte count, address, and data bytes.

The end-of-file record contains a colon, a two-character byte count (set to 00), a four-character address, and a two-character record type (set to 01).

4) MOTOROLA EXORCISER

Motorola data records begin with the characters "S1".

The next two characters represent the byte count, which specifies the number of data, address, and checksum bytes in the record.

The next four characters specify the address of the first data byte in the record.

This is followed by the data bytes, where each data byte is represented by two ASCII hex characters.

Each data record terminates with a two-character checksum (given by the one's complement of the binary sum of the preceding bytes in the record, including the byte count, address, and data bytes).

The end-of-file record starts with the characters "S9", followed by a two-character byte count (set to 03), followed by a four-character address (set to 0000), followed by a two-character checksum.

5) MOS TECHNOLOGY

Each data record begins with a semicolon (";"), followed by a two-character byte count, followed by four characters specifying the address of the first data byte in the record, followed by the data bytes (each represented by two ASCII hex characters).

Each data record terminates with a four-character checksum, which represents the sixteen-bit binary sum of all hexadecimal bytes in the record, including the address and byte count. The carry is dropped from the most significant bit.

The end-of-file record contains a semicolon, followed by a two-character byte count (set to 00), followed by a four-character record count (set to the number of records transmitted), followed by a four-character checksum, computed as above.

6) FAIRCHILD FAIRBUG

In Fairchild Fairbug format, data are grouped into files, consisting of eight-byte records plus control characters.

Each file is preceded by a five-character prefix and ends with a one-character suffix.

The start-of-file character (first character of the prefix) is an "S", followed by four characters specifying the address of the first data byte.

The end-of-file character (suffix) is an asterisk ("*"), which indicates the end of data transmission.

Each data record is preceded by an "X", followed by eight data bytes (where each data byte is represented as two ASCII hex characters). A one-character hexadecimal checksum follows the data in each data record. The checksum represents, in hexadecimal notation, the sum of the binary equivalents of the 16 digits in the record; the half carry from the fourth bit is ignored.