10. Using the PROM Programmer

The PROM Programmer is a hardware and software package that facilitates user programming of RCA and other industry-standard PROM's. Specifically, the RCA CDP18U42, the Intel 2704, 2708, 2758, and 2716 UV-erasable PROM's, or equivalent PROM's can be programmed with this system. In addition, Intel 1702-type PROM's can be read (but not programmed) and copied into lower-power CDP-18U42 CMOS PROM's or combined into larger-size PROM's. The PROM Programmer is included with the CDP18S694V1 and V3 systems and may be added to the CDP18S693V1 and V3 systems by means of the PROM Programmer package CDP18S680.

The PROM Programmer package includes a plug-in Microboard module and a magnetic tape cassette containing the operating software. The software is designed for flexibility so that, in addition to the numerous basic operations provided, more sophisticated procedures can be derived from these building blocks. For example, a PROM can be copied from another PROM or programmed from a tape-resident file. The user can also combine two smaller PROM's into a larger one (two 2708's copied into one 2716). PROM's are automatically verified after programming. The user, however, can manually verify the erasure of a PROM, verify one PROM against another, or verify a PROM against a tape-resident file. These operations can be done with either positive (non-inverted) or negative (inverted) data.

An additional feature allows PROM's to be saved on a tape file (in a reloadable format) that can be used to generate masked ROM's. For PROM production purposes, a software feature facilitates rapid programming of many PROM's from the same source.

Installation

The PROM Programmer Module, as provided with the CDP18S694V1 and V3, is installed in the bottom slot of the five-card chassis. When the module is to be used for programming PROM's, the user should remove the top cover and exchange the PROM Programmer module for the Microboard module in the top slot. By having the PROM programmer module in the top slot, access is provided to the three PROM programming sockets.

An external power supply is required as a programming voltage to be connected to jack J3, labelled PROG PWR. The voltage required is given in Table XVI. The supply must be well regulated and capable of supplying peak currents of 100 milliamperes. A -9-volt supply connected to J2 is also necessary when the programmer is used with a 1702 PROM. J1 is a ground connection. In addition to the external power supplies (Programming Power and -9 volts), voltages of +12 volts and –5 volts (both ±5%) must be supplied to the Microboard Universal Backplane pins 20 and 11, respectively, when 1702, 2704, or 2708 devices are to be operated on. Detailed specifications for the power supply requirements are given in Appendix I.

The PROM programmer module is assigned to Group Select 4. The module generates a signal on flag line EF1 of the CPU indicating whether the 1702/1842 switch is in the correct position.

Table XVI—Programming Voltages Required at Jack J3

PROM Type No.	Programming Voltage
2716/2758	+25 V±0.1 V
2704/2708	+26 V±0.1 V
CDP18U42	+20 V±0.1 V

Memory Organization and Use

The memory map for the PROM Programmer software is given in Fig. 17. The program loads starting at address 0000 and occupies the lowest two kilobytes of RAM. The hexadecimal addresses from 0800 to 0FFF comprise a data buffer area used to hold data moved to and from the PROM being operated on. The buffer memory is treated in pages, where a page is defined as 256 bytes. Thus, the first page of buffer memory occupies the hexadecimal addresses 0800 through 08FF and is called page 8; the second page, 0900 through 09FF, is called page 9; etc. The data of a 1702 or CDP18U42 occupy one page, a 2704 two pages, a 2708 or 2758 four pages, and a 2716 eight pages.

The program asks for the starting page in the data buffer against which operations are to be performed.

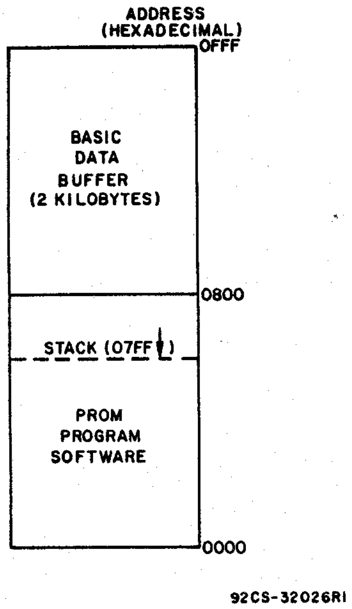


Fig. 17—Memory map for PROM Programmer software.

The response must be a one- or two-digit hexadecimal number equal to or greater than eight. Because a leading zero is assumed, only a single digit has to be entered for pages 8 through F. A number less than eight is not accepted by the program, thereby preventing accidental overwrite of some areas. Additional RAM can be used for additional page buffer areas.

The user can copy a PROM into any page (⩾ 8), program a PROM from any page, or verify a PROM against any page. He does not necessarily have to start at page 8 and can take advantage of additional RAM if the MCDS is equipped with more than 4 kilobytes. However, when an object file is loaded into the buffer area, loading starts at address 0800. Likewise, when the buffer area is saved on a disk, the SAVEing begins at address 0800. Refer to the Section Operations and Commands for details on the operation of the object file reader and SAVE programs.

It is relatively easy to copy two smaller PROM's into one larger one. For example, to copy two 2704's into one 2708, the following steps would be followed.

(1) Copy the first 2704 starting at page 8. It will occupy two pages, addresses 0800 through 09FF.

(2) Copy the second 2704 starting at page A. It will occupy addresses 0A00 through 0BFF.

(3) Then, program a 2708 starting at page 8.

Reversing the procedure would permit programming several smaller PROM's from a larger one. Some care must be exercised in keeping track of where data is loaded in buffer memory for these combined operations. The software will not prevent overwriting of data in the buffer area because that may be the desired opeation.

When programming takes place, data is taken from the starting page specified for the appropriate number of bytes. Thus, the data to be programmed must be contiguous in the buffer area.

Software Considerations

General Procedure

The PROM Programmer software is designed to prompt each step of a procedure so that operation of the system is self-explanatory and, it is hoped, relatively error-proof. For each type of operation (except SAVE) the user is asked to identify the type of PROM being used and whether positive (noninverted) or negative (inverted) data is to be used. When the program is started, and after completion of each operation, the following message is typed, asking what operation is next.

F, V, P, C, S, J, U?

The single letters refer to the following operations. The user should enter the appropriate letter.

F—Fill selected pages of the RAM buffer with 1's (positive logic) or 0's (negative logic)

V—Verify a PROM against the RAM buffer or a file

P—Program a PROM from the RAM buffer or a file

C—Copy a PROM into the RAM buffer

S—Save the RAM buffer on a file

J—Rewind the tape

U—Exit to the monitor

For all except the SAVE operation, the PROM Programmer asks for the PROM type by the message

TYPE#?

The user should enter one of the following:

1702, 1842, 2704, 2708, 2758, or 2716

followed by a carriage return (CR).

The next question asked is whether the operation is to be performed with positive logic (non-inverted data) or negative logic (inverted data).

LOGIC =P, N?

The user replies by typing a P for positive logic or an N for negative logic.

These three prompts are common for the V, C, P, and F commands. All operations also have in common the manipulation of data in the RAM buffer area previously described. Data to be programmed goes to a PROM from the buffer area. Likewise, data copied from a PROM goes into the buffer, etc. Succeeding questions ferret out the area of RAM to be used.

When an operation is completed, the program types

DONE

In the case of a verify (V) or programming (D) sequence, it asks if the user wants to repeat the operation with a question such as

REPROGRAM 1842?

The user responds with a "Y" for yes or an "N" for no. Because a "yes" response starts another programming cycle immediately, the new PROM should be in place before an affirmative response is given. A "no" response restarts the program.

The Section **Operations and Commands** gives details and examples of each operation.

Error Messages

Whenever the program receives an inappropriate response to a question, it types a "?" and repeats the question. It continues this reaction to each incorrect response until an acceptable one is received. Then, it will proceed.

The "?" is also generated if a number less than 8 is entered in response to the PAGE? question or if the user asks to program a 1702 (a non-existent function).

The following error messages may also be generated when a tape-read operation is being performed:

NO OBJECT CODE

or

FORMAT ERROR

or

LOAD ABORTED

The last message follows a non-recoverable error message.

Aborting Operations

When a PROM is being verified, a list of errors is generated showing any mismatches between the PROM data and what is in the RAM buffer. A long typeout can be halted by pressing the BREAK key. Because the BREAK key is sampled at the end of each typed line, it must be held down long enough to be "seen" by the program. The message

*****ABORTED*****

is typed and, when the BREAK key is released, the program restarts from the beginning.

A programming operation can be aborted by pressing and holding the BREAK key. Again, the message is typed and the program restarts. Response to this key may take a few seconds.

Correcting Typing Errors

All numbers entered in response to a question must be followed by a carriage return (CR). The program looks only at the last two or four digits, as appropriate, that have been entered before the CR. Any preceding digits are ignored. This feature provides a convenient means of correcting typing errors. The user needs only to continue typing until the correct digits have been entered before pushing CR. For example, typing

27142704(CR) is equivalent to

2704(CR)

Or, in response to the PAGE# = question, typing

41008(CR) is equivalent to

08(CR)

Inappropriate responses result in a "?" appearing and the question reasked as described in the subsection "Error Messages". These kinds of errors are corrected by simply entering a valid response on the next try.

Errors of the third kind, in which an incorrect though valid response is made, must be corrected by pressing the RESET button on the MCDS and restarting the program with the monitor program's P command. For example, if a "V" command is inadvertently entered instead of the "P" command, the program should be restarted to correct the situation.

Precautions

A PROM should never be inserted or removed with power on. The user should always make sure that the power ON/OFF switch S2 on the PROM programmer module is in the OFF position before inserting or removing a PROM.

To prevent any potential problems, the external power supplies (programming power and—9 volts), when present, should be sequenced on simultaneously or after the MCDS is turned on and shut off before or at the same time the MCDS is shut off.

There is another switch (S1) on the PROM Programmer board that selects between a 1702 and CDP18U42 configuration for socket 1.

NOTE: This switch should never be in the 1702 position if a CDP18U42 is in the socket and the —9-volt supply active. If such were the case, —9 volts would be connected to an input pin of the CDP18U42, and a destructive latch-up condition could be triggered in the device. If the program generates the warning message

SET SWITCH!

it detects the wrong position after the program has started and waits before proceeding. This warning, however, should not be relied on to prevent damage to the device. **Before inserting a CDP18U42, be sure switch S1 is in the correct position.**

The user should be sure to insert each PROM into its correct socket and be sure that it is properly oriented. Pin 1 is the pin adjacent to the handle of the zero-insertion force sockets. **Never have more than one PROM in the module at any time.** Because the PROM sockets share common address and data buses, contention or loading problems could be caused if more than one device is inserted. Additionally, extraneous signals to a socket which should be empty might cause harm to a device left in it.

The PROM programmer software derives its timing from the 2.0-MHz clock frequency of the crystal on the CDP18S601 Microboard Computer Module. If this frequency is changed, the results are unpredictable.

Also, note the power requirements for the PROM Programmer module as given in Appendix I. If additional modules are installed in the MCDS that could tax the internal power supplies to their limits, these modules should be removed before attempting to run the programmer.

Before starting a programming operation, be sure that the external programming power supply (connected to J3) is at the proper voltage as given in Table XVI. Excessive voltage can cause permanent damage to a PROM.

It is recommended that the programmer module not be run on an extender card in the MCDS system. An extender card will exaggerate bus-loading effects and may lead to intermittent operation of the programmer, particularly in systems already loaded with extra I/O or memory modules. If anomalous operation is observed, a first step should be to remove any extender cards and extraneous modules from the system.

Operations and Commands

Following is a discussion of each command and operating sequence.

Programming Operations—The "P" Command

The system is capable of programming the RCA CDP18U42 CMOS PROM and Intel 2704, 2708, 2758, and 2716 type PROM's. Data to be programmed comes from the RAM Buffer area, pages 8 and up in the MCDS RAM. The data must first be loaded into the appropriate page(s) by COPYing another PROM into them or by loading the appropriate section of a tape file into them. The file loading, when requested, leads automatically into a programming cycle. Refer to the section, **File Reader Subroutines** for a detailed discussion of the file loaders. Programming begins immediately from the specified page.

Procedure

The program always begins by printing

MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?

After the user types the letter 'P", the program asks

TYPE #?

The user should enter one of the following:

1842, 2704, 2708, 2758, or 2716

followed by a (CR).

The program next asks if operation is to be with true or inverted data by typing

LOGIC=P, N?

A "P" response causes the operation to occur with true (non-inverted) data; an "N" response causes the operation to proceed using an inverse of the data. For example, to copy one PROM into another, inverting data, one could copy the master PROM using positive logic and program the copy PROM using negative logic or vice-versa. Negative logic would not be used for both the copy and program step because that would cause a double inversion.

The program next asks

(T)APE OR (R)AM?

The user should enter the letter "R" if the data to be programmed is already in the buffer RAM.

The next prompt is

PAGE# =

asking for the first page on which data is located. Data is taken from that page and subsequent pages as required for the PROM being programmed.

If the user enters the letter "T", indicating data to be programmed is on a tape file, the program will ask for the unit by typing

TAPE#

The user should respond with either a "0" or a "1".

Next, the program asks for the address range of the PROM by typing

LOWEST PROM ADDRESS (XX00)?

The user should respond with a four-digit number followed by a (CR). This number should be the lowest address of data to be programmed into the PROM. For instance, if a 2704 PROM is to be programmed for data addresses 0200 through 03FF, then 0200 would be typed as a response. A default value of 0000 is entered if the user hits the carriage return without typing in a number. If a number is entered, **all four digits must be entered.** The file reader searches the object file and loads only the data falling between the lowest address specified and up to 2 kilobytes above that address. The appropriate data is copied into RAM starting at page 8 and then the actual programming begins automatically.

Note that programming will be initiated when the 2-kilobyte RAM area is filled or an end-of-file marker is read from tape. Thus, if the user enters an improper start address, the PROM will be programmed with invalid data.

Any programming operation is automatically followed by a VERIFY sequence. If the verify is successful, the program types

DONE

If programming is not successful, a list of errors is typed. This list can be stopped by depressing the BREAK key until the

*****ABORTED*****

message is typed.

If programming has been done successfully, a message such as

REPROGRAM 1842?

appears, identifying the PROM type just programmed. If the user does not want to program another device of the same type with the same information, he should type "N" for NO. If the programming of another PROM of the same type is required, the user should take the following steps:

—Shut off power switch S2
—Remove the PROM just programmed
—Insert the new PROM to be programmed
—Turn on switch S2
—Type the letter "Y" for yes on the terminal.

Another programming sequence will begin immediately.

During a programming operation, the LED indicator labeled PROG on the programmer board is ON, providing a visual indication of operation. The programming operation can be halted at any time by pressing and holding the BREAK key until the message

ABORTED

is typed.

The programming time required for each PROM type is given in Table XVII.

Table XVII Programming Times—Typical Values.

Type No.	Time
CDP18U42	3 seconds
2704	1 minute 25 seconds
2708	2 minutes 45 seconds
2716	1 minute 45 seconds
2758	50 seconds

Examples

In the following examples, system-generated output is underlined.

1. Program a 2704 from RAM pages 8, 9

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 2704
LOGIC=P, N? P
(T)APE OR (R)AM? R      PAGE# = 8
DONE
REPROGRAM 2704? N
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

2. Program two CDP1842's with the same data from a tape on drive 1. Program for address 0100 to 01FF.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 1842
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 1
LOWEST PROM ADDRESS (XX00)? 0100
LOADING COMPLETED      ..FILE LOADED
                       ..PROGRAMMING BEGINS
                       ..PROGRAMMING AND
                       ..VERIFY DONE
REPROGRAM 1842? Y      ..INSERT NEXT 1842,
                       ..THEN TYPE Y.
DONE                   ..PROGRAMMING AND
                       ..VERIFY DONE
REPROGRAM 1842? N      ..RESTART PROGRAM
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? U ..RETURN TO MONITOR
*
```

Verify Operations— The "V" Command

Verify operations compare a selected area of the RAM buffer against the contents of a PROM. As with the programming operation, the buffer is loaded by either COPYing a PROM into it or loading data from a file. The user can select the page(s) against which a PROM is to be verified, or, if he chooses to use a file, the appropriate data is loaded starting at page 8 and automatically verified from there. See subsection "File Reader Subroutines" later in this section for details on the file reader operation.

Procedure

A verify operation is automatically done after a PROGRAM or COPY operation. If there are no errors, the DONE message appears. Any mismatch between a PROM address and the corresponding buffer address results in an error listing such as

ERR;M(0800)=FF, PROM=00

This error listing says that location 0800 in the RAM buffer contains an FF, but the corresponding location of the PROM (which would be its first address in this case) contains a 00. A long error listing can be stopped by pressing and holding the BREAK key until the message

ABORTED

is typed. The program then restarts from the beginning when the BREAK key is released.

The program begins by asking for the type of operation to be performed, for the PROM type number, and whether the operation is with positive or negative logic. The verify command has exactly the same command format and requirements as the "P" command.

At the end of a verify operation, the program will ask if another PROM of the same type is to be verified against the same RAM data. If the answer is no, the user should type an "N". If the answer is yes, the following steps should be taken.

—Shut off power switch S2
—Remove the PROM just verified
—Insert the new PROM to be verified
—Turn on switch S2
—Type the letter Y for yes on the terminal

Examples

1. Verify a 2708 against RAM pages 9 through C.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? V
TYPE #? 2716
LOGIC=P, N? P
(T)APE OR (R)AM? R      PAGE# =9
DONE
REVERIFY 2708? N
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

2. Verify a 2716 against a tape file. Verify against addresses 0500 through 0CFF of the file.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? V
TYPE #? 2716
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 1
```

```
LOWEST PROM ADDRESS (XX00)? 0500
LOADING COMPLETED
ERR:M(0800)=18, PROM=FF ..ERROR LSTG STARTS
ERR:M(0801)=C0, PROM=FF
ERR:M(0802)=04, PROM=FF ..BREAK KEY PRESSED
***ABORTED***
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

Copy Operations—The "C" Command

Procedure

The copy operation reads the contents of a PROM into the selected page(s) of the buffer RAM. Make sure there is sufficient free RAM to hold all the data to be copied. For example, copying a 2716 requires two kilobytes of RAM buffer and the copy must begin on page 8 if there are only four kilobytes of RAM in a MCDS system. The program does not test for the presence of RAM at the assigned page(s). A copy operation is automatically followed by a verify. Upon completion of a copying operation, the program responds with a "DONE" message or a list of errors if the copy did not verify. Errors could be due to a hardware problem or attempting to copy into non-existent RAM.

A direct copy of the PROM is made if positive logic is chosen; the data is inverted if negative logic is chosen. A COPY followed by a PROGRAM operation is the sequence used to reproduce PROM's. Refer to the next Section for examples of compound operations.

Example

Copy a 2704 into RAM pages 8 and 9.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? C
TYPE #? 2704
LOGIC=P, N? P      PAGE# =8
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

The Fill Command

Procedure

This command is used to fill selected page(s) of the buffer RAM with all 1's for positive logic, or all 0's for negative logic. The user should respond to the PAGE#= prompt with the starting page, as usual. The number of pages that will be filled is determined by the PROM type number entered—1 page for 1702 or CDP18U42, 2 pages for a 2704, etc.

This command is used in verifying that a PROM has been completely erased. The erased PROM is VERIFY'ed against a buffer filled with 1's or 0's, as appropriate.

Example

Fill RAM pages B through E with 0's (4 pages).

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? F
TYPE #? 2708      ..2708 IS 4-PG DEVICE
LOGIC=P, N? N      PAGE# =B
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

The "U" Command

Procedure

The "U" command returns control to the monitor program which responds by printing an asterisk (*) prompt.

Example

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? U
*
```

The "J" Command

Procedure

The "J" command allows the user to rewind tapes.

Example

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? J
REWIND TAPES, THEN TYPE ANY KEY X
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

The SAVE Operation—The "S" Command

The SAVE routine allows the user to copy any number of pages of the buffer RAM onto a file in reloadable format. The file thus generated will be in ASCII-Hex and can be used to program other PROM's. Also, it can be loaded into RAM for a ROM emulator function (a Micromonitor feature) or it can serve as the data for generation of masked ROM's such as the CDP1831/33.

Procedure

The prompts for this command are different from the previous ones. The user is not asked for the type of PROM or the starting page. Saving always starts on page 8 and is done in positive logic. The number of pages to be saved is designated in hexadecimal—from 1 to F7. The S command, in conjunction with the tape file reader and other commands, can be used to merge or truncate files and combine several smaller PROM's for the generation of one larger ROM. See the next section for examples of compound command usage.

When the SAVE operation is started, it asks for the number of pages of data to be saved on tape by typing

# PAGES?

The SAVE operation then asks for the tape number the data is to be written to by typing

TAPE#?

The user should respond with either a "0" or a "1". The response to the prompt

LOWEST PROM ADDRESS (XX00)?

is used to generate the starting address on the file. For example, the response "0100" causes data to be written on the file as follows:

```
I
0100 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX;
0110 XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX;
```

Be sure the correct starting address is assigned.

SAVE'd files on tape cassette will have a leader and trailer of nulls appended automatically.

Example

Save RAM pages 08 through 12 (hex) under the file on a tape in drive #1. The data is from a PROM with starting address 0200.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? S
# PAGES? A ..TEN PAGES WILL BE SAVED
 TAPE#? 1
 LOWEST PROM ADDRESS (XX00)? 0200
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U?
```

File Reader Subroutines

The File Reader Subroutine is a progam called by either the PROGRAM or VERIFY routines. It loads data from a file into the buffer RAM starting at page 8. It is an intelligent loader that will search through a file and load data in the address range of interest (+2 kilobytes from the initial address) ignoring extraneous data. It will handle either hex or listing files created by the COSMAC assemblers and files created by the monitor program (W command). When the user replies to the prompt

LOWEST PROM ADDRESS (XX00)?

the program takes that address as the first address in the range and adds 2K to get the last address. It will thus load up to 2 kilobytes of data, filling the basic buffer (but not writing into any additional CDS RAM) if there is sufficient data. Of course, if the file contains less than 2 kilobytes in the address range, some of the buffer will not be used. Once a 2-kilobyte file is loaded, small PROM's can be programmed from successive pages of the buffer. Refer to the next section for an example.

If no data is found, the program types

NO OBJECT CODE

In the tape cassette system, the program will restart from the beginning.

Format errors detected on the cassette will cause the program to type the following messages and restart from the beginning.

FORMAT ERROR
LOAD ABORTED

When loading starts, the message

LOADING

is typed. At the completion of a successful load, the word COMPLETED is added to the sentence. The program then goes automatically into the program or verify operation.

NOTE: In response to the prompt

LOWEST PROM ADDRESS (XX00)?

a full four-digit number followed by a carriage return (CR) should be entered. If the user hits CR without entering any numbers, the default address of 0000 is used. Typing mistakes can be corrected by continuing to type until the correct four digits have been entered before hitting CR. Only the last four digits before the CR are recognized and only the first two of the group are meaningful—the last two are always taken as 00. Below are some examples.

Response	Yields
0200(CR)	0200
0251(CR)	0200
2(CR)	0000
030244(CR)	0200
03029(CR)	3000
(CR)	0000

Application Hints

It is good practice to verify that a PROM is erased before attempting to program it. A suitable technique is to use the "F" command to fill the buffer with 1's or 0's and then the "V" command to verify the PROM against it. When a PROM is being programmed to work in an RCA 1800-series microprocessor-based system, it is advisable to first fill the buffer with all 0's before loading in the data to be programmed. This step will set data in the unused addresses of the PROM to 00, which is an IDLE instruction in the RCA 1800-series instruction set. Thus, an erratic program fetching from an erroneous address will pick up the idle instruction and stop. A PROM thus programmed must use the same procedure when it is verified. That is, the buffer is first filled with 0's, the data loaded, and the verify operation is made. This technique assures that no "errors" will appear when the programmed "00" locations are verified against random data left in the buffer.

An application note (ICAN-6847) describing a hardware adjunct that can be used to program Intel 2732 PROM's with this system is available on request.

Examples of Compound Operations

This section gives five examples of compound operations.

Example 1

Program a CDP18U42 from a 1702.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? C     ..COPY THE 1702 INTO RAM
TYPE #? 1702
LOGIC=P, N,? P      PAGE#=8
DONE     ..REMOVE 1702, SET SWITCH,
         ..INSERT 1842
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 1842
LOGIC=P, N? P
(T)APE OR (R)AM? R      PAGE#=8
DONE
REPROGRAM 1842? N     ..RESTART PROGRAM
```

Example 2

Verify that a 2704 has been erased, then program it from a file. The PROM will occupy addresses from 400-5FF.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? F
TYPE #? 2704
LOGIC=P, N? P      PAGE#=8
    ..FILL PGS. 8 & 9 WITH FF'S
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? V
TYPE #? 2704
LOGIC=P, N? P
(T)APE OR (R)AM? R     PAGE#=8
DONE     ..ERASURE VERIFIED—NOW PROGRAM
REVERIFY 2704? N
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 2704
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 1
LOWEST PROM ADDRESS (XX00)? 0400
LOADING COMPLETED
DONE     ..PROGRAMMED AND VERIFIED
REPROGRAM 2704? N     ..RESTART PROGRAM
```

Example 3

Combine and save two CDP18U42's on tape to make a CDP1831 ROM with address range 0100-02FF.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? C   ..COPY 1ST PROM INTO PG 8
TYPE #? 1842
LOGIC=P, N? P      PAGE# =8
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? C   ..COPY 2ND PROM INTO PG 9
TYPE #? 1842
LOGIC=P, N? P      PAGE# =9
DONE
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? S
# PAGES? 2
TAPE#? 1
LOWEST PROM ADDRESS (XX00)? 0100
DONE
```

Example 4

Program two 2708's from a file. One occupies addresses 0000-03FF and the other 0400-07FF.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 2708
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 0
LOWEST PROM ADDRESS (XX00)? (CR)
LOADING COMPLETED
DONE     ..1ST PROM PROGRAMMED
         ..FOR 0000-03FF
REPROGRAM 2708? N
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE#? 2708
LOGIC=P, N? P
(T)APE OR (R)AM? R      PAGE#= C
DONE       ..FOR 0400-07FF
REPROGRAM 2708? N
```

Example 5

Program two 2716's from a file. One occupies addresses 0000-07FF and the other 0800-0FFF.

```
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 2716
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 0      ..LOAD 1ST ADDRESS RANGE
LOWEST PROM ADDRESS (XX00)? (CR)
LOADING COMPLETED
DONE     ..1ST PROM PROGRAMMED
REPROGRAM 2716? N
MCDS PROM PROGRAMMER VER X
F, V, P, C, S, J, U? P
TYPE #? 2716
LOGIC=P, N? P
(T)APE OR (R)AM? T
TAPE#? 0      ..LOAD 2ND ADDRESS RANGE
LOWEST PROM ADDRESS (XX00)? 0800
LOADING COMPLETED
DONE     ..2ND PROM PROGRAMMED
REPROGRAM 2716? N
```

ICAN-6847

Programming 2732 PROM's with the CDP18S480 PROM Programmer

by D. Block

The CDP18S480 PROM Programmer was designed to program a variety of industry-standard PROM's, including the Intel 2704, 2708, 2758, and 2716's and equivalent products from other suppliers. With a simple hardware addition to the PROM Programmer, and without any software changes, the CDP18S480 can also be used to program Intel 2732 PROM's. This addition to the PROM family is organized as 4K × 8 (4 kilobytes x 8 bits) and operates from a single +5-volt supply. All inputs are T^2L compatible except for pin 20 ($\overline{OE}/V_{PP}$), which must be pulsed to 25 volts during programming.

Many electrical and mechanical characteristics have been maintained between the 2716 (organized as 2K × 8) and the 2732 for upwards compatibility. For example, only the definitions of pins 20 and 21 have changed between the two versions, as shown in Fig. 1. In the 2732, pin 20 continues to represent an output enable function, but with V_{PP} (the programming voltage) also impressed on it; pin 21 becomes the A_{11} address pin.

Electrically, the programming voltage (+25 volts) and programming pulse width (50 milliseconds nominal) are the same for both the 2716 and 2732. The technique for 2732 programming described here takes advantage of this fact. The 2732 will be treated as two 2716's for all operations, and the logic circuitry described below will handle necessary logic-level conversions and voltage switching. The hardware is designed to plug into the 2716 socket of the PROM programmer.

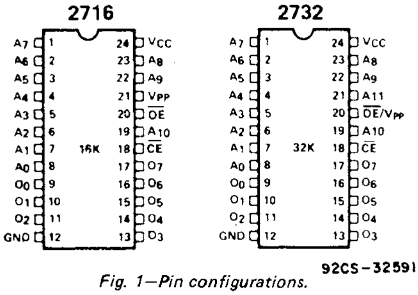


Fig. 1—Pin configurations.

Theory of Operation

Fig. 2 shows the programming waveforms for the 2716 and 2732. Note that the voltages on pin 21 of the 2716 are almost those required for pin 20 of the

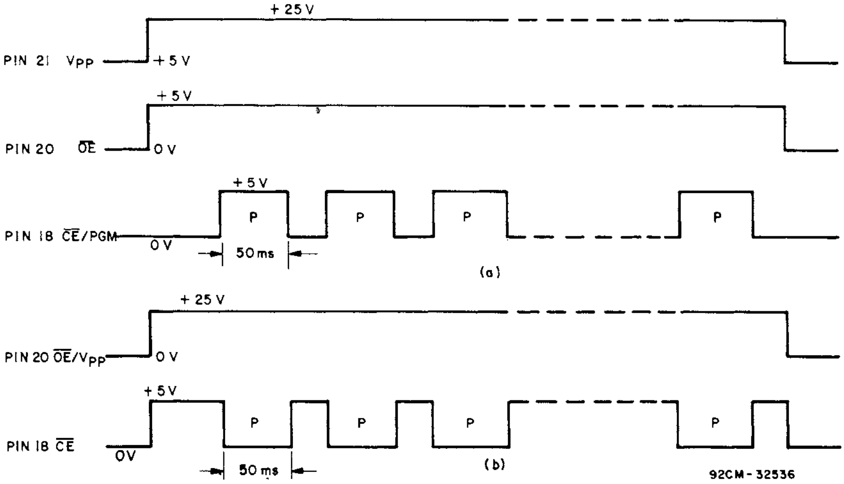


Fig. 2—Programming waveforms (not to scale): (a) 2716, (b) 2732.

2732. However, pin 20 must go to ground in the Read mode. The transistor circuitry in Fig. 3 is designed so that when pin 20 of the 2716 is low, the input to pin 20 of the 2732 will also be low. When pin 20 of the 2716 is high, pin 20 of the 2732 is connected to the programming voltage (if a programming operation is in effect) or to +5 volts, disabling the chip.

The programming pulses applied to pin 18 of the two devices differ in polarity. Inversion of the pin 18 signal is accomplished through pin 20 of the 2716 by means of an Exclusive-OR gate. The conditions for reading and programming are met by this arrangement.

As mentioned above, pin 21 on the 2732 is the A_{11} address pin. Fig. 3 shows this pin connected to a toggle switch. The 2732 will be exercised twice for each operation: once with A_{11} low, and then with A_{11} high.

Hardware Construction

The circuitry of Fig. 3 can be constructed on a small vectorboard that plugs directly into the 2716 socket of the PROM Programmer. A ZIF, zero-insertion-force, socket is required for the 2732. It is recommended that solder connections not be made directly to this socket, since the solder may wick up the leads and interfere with operation of the pin clamps. Instead, the ZIF socket should be plugged into a second socket, and connections made to that socket.

Only a CD4049 inverter should be used to drive pin 18 of the 2732. Other CMOS inverters may not have sufficient drive to handle the T^2L input. The use of capacitor C1 is recommended by Intel to prevent transients from exceeding the +26-volt maximum rating of the device.

Operation

The 2732 will be programmed as two 2716's.[1] When programming from a disk file, sequential sections of the file must be loaded in, since the disk-file reader loads only 2 kilobytes at a time into the buffer. Similarly, when copying one 2732 into another, several steps are necessary. The examples below should help clarify the requirements. CDS output is underlined. The CDOS software version is assumed.

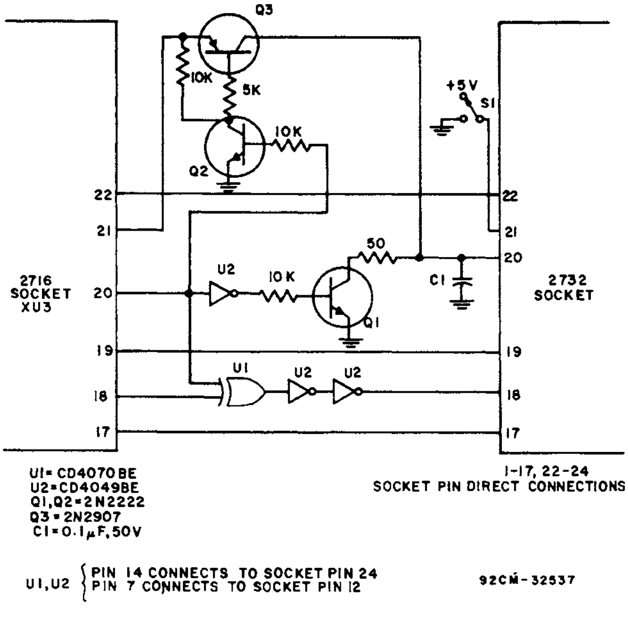


Fig. 3—Circuit diagram.

Example 1:

Program a 2732 from a disk file called FILE.OBJ for the address range 0000-0FFF. First, set S1 so that pin 21 is low.

```
PROM PROGRAMMER, VERSION XX
 ENTER V,C,P,F,S,U: P
 TYPE#: 2716 (CR)
 LOGIC=P,N: P
 (D)ISK OR (R)AM: D
 INPUT FILENAME: FILE.OBJ
 ENTER LOWEST PROM ADDRESS (XXOO):  0000(CR)
 LOADING COMPLETED
 DONE!                   . . The first half of the PROM is done
 REPROGRAM 2716?: N      . . Set S1 in the opposite position
 PROM PROGRAMMER, VERSION XX
 ENTER V,C,P,F,S,U: P
 TYPE#: 2716(CR)
 LOGIC=P,N?: P
 (D)ISK OR (R)AM?: D
 INPUT FILENAME: FILE.OBJ
 ENTER LOWEST PROM ADDRESS (XXOO): 0800(CR)   . . Second half of
                                                   address range.
 LOADING COMPLETED
 DONE!                   . . Entire 2732 is now programmed
 REPROGRAM 2716? N
```

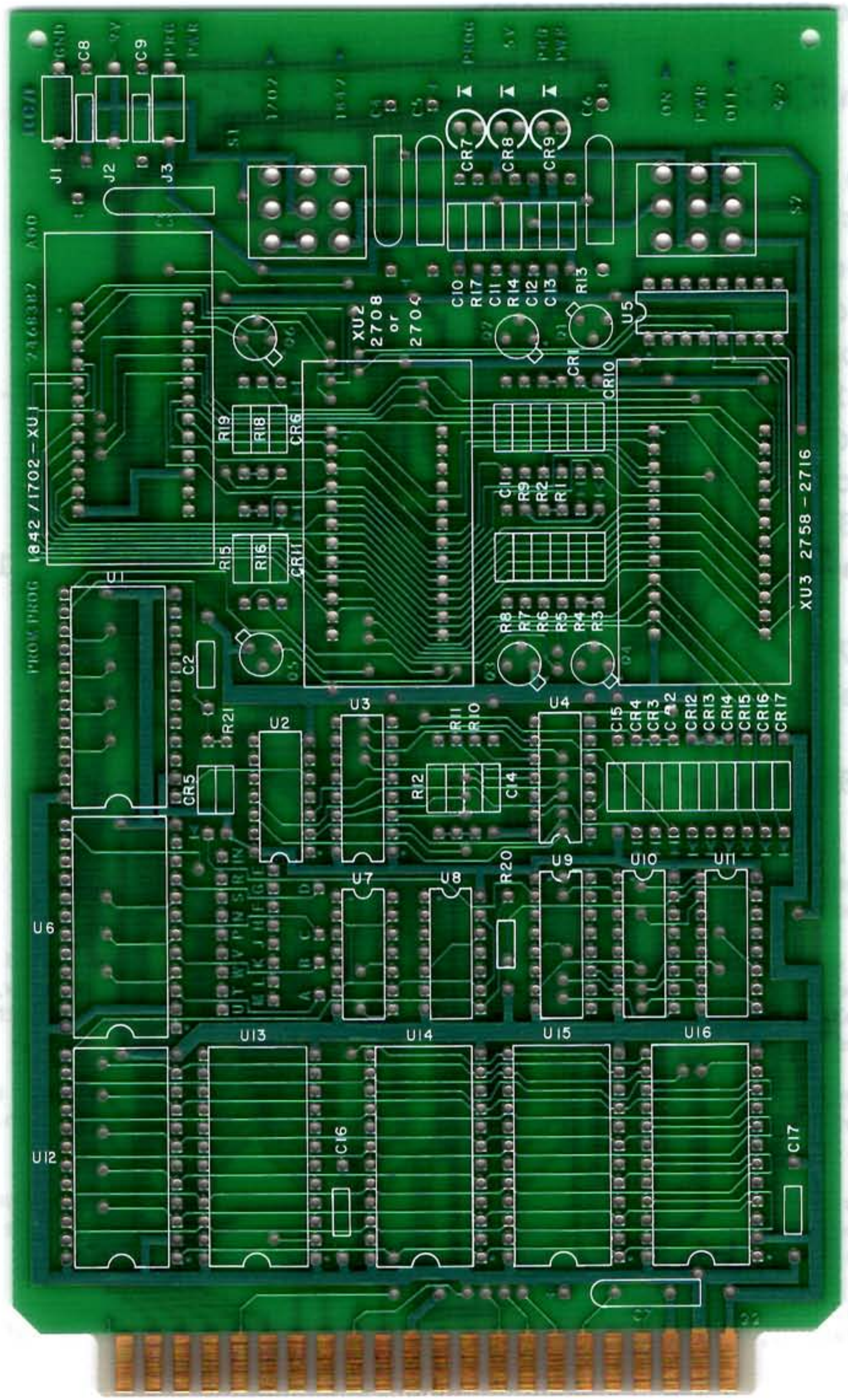
Example 2:

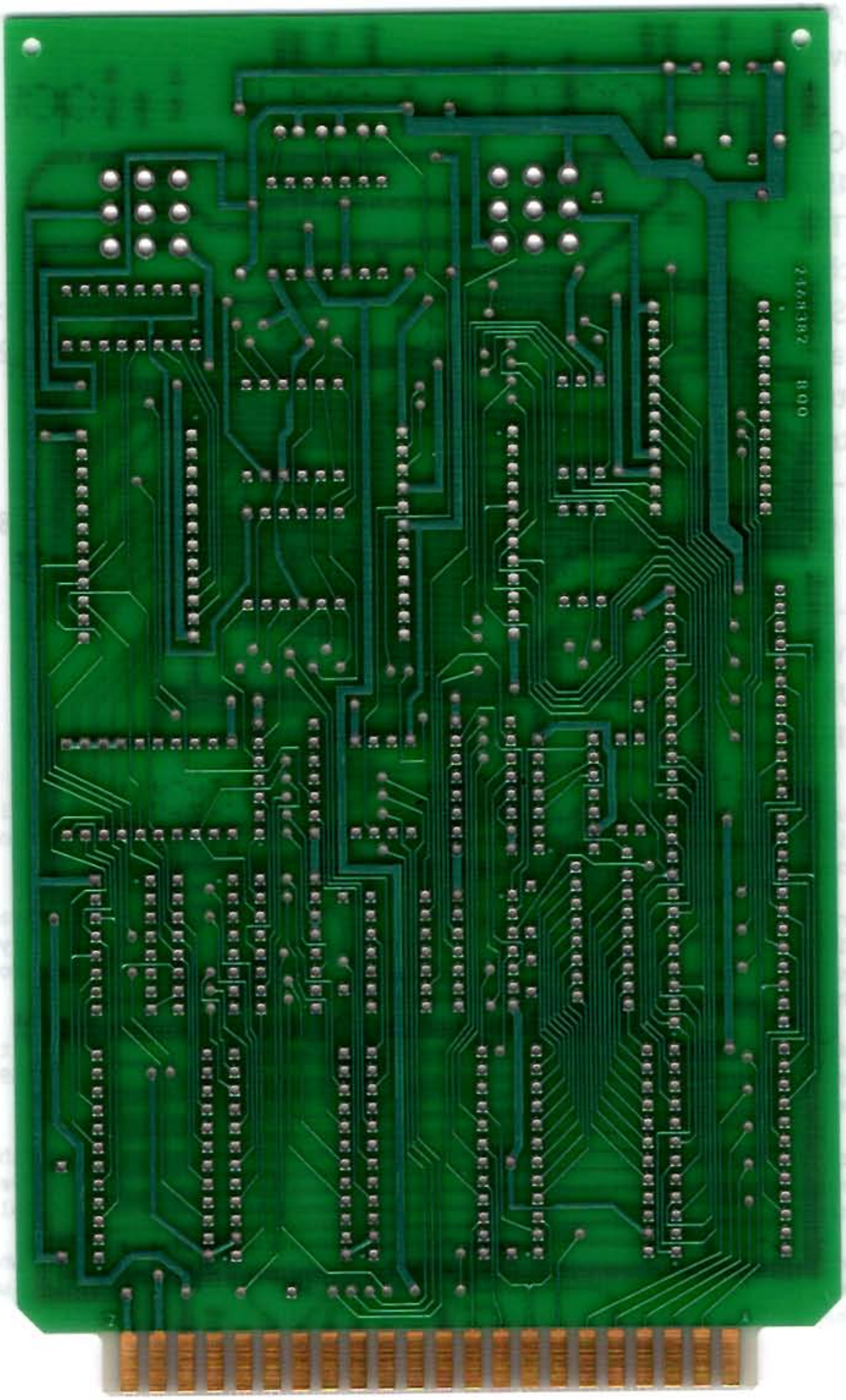
Copy two 2716's into one 2732. Insert the lower address 2716 first.

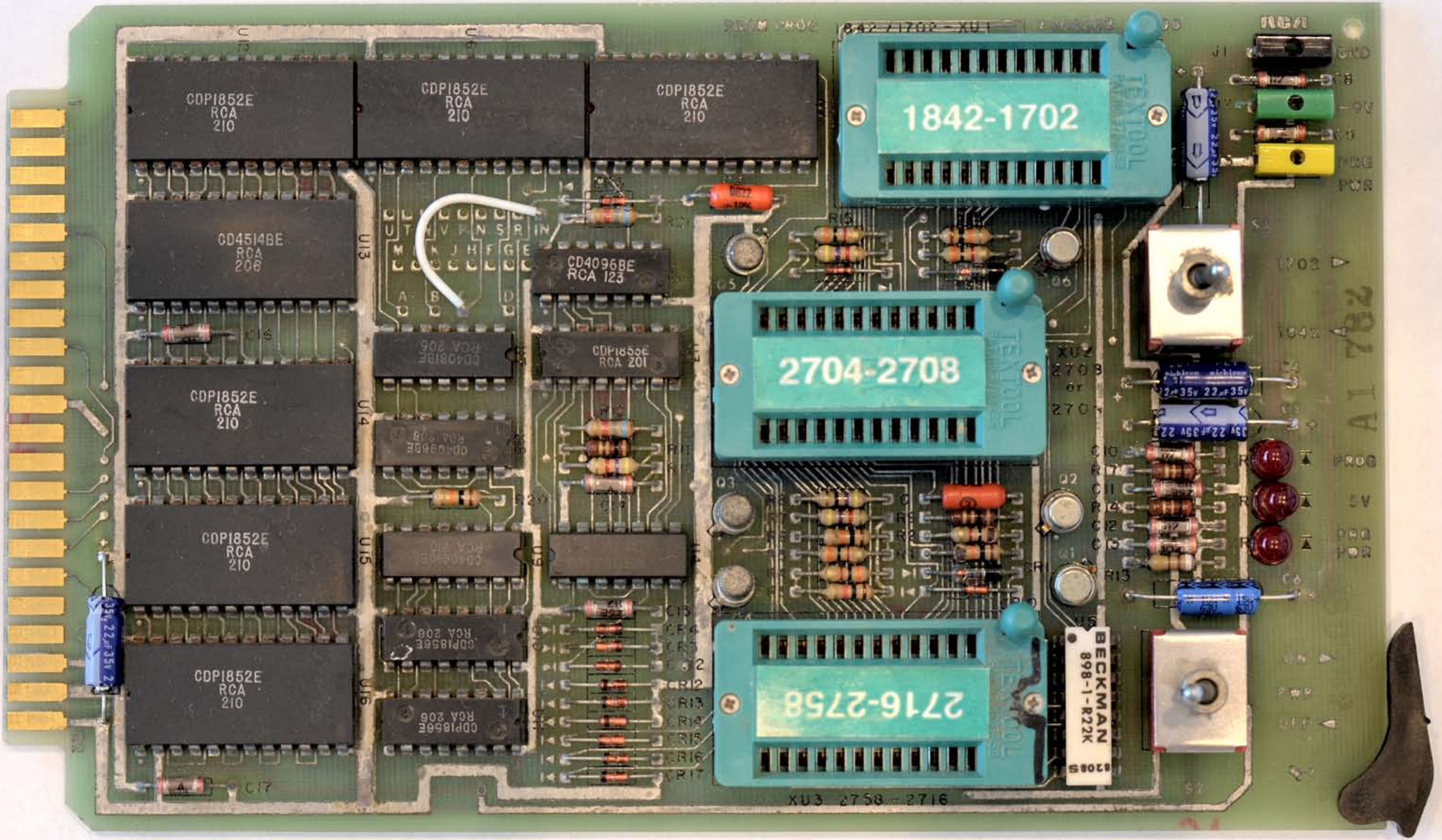
```
PROM PROGRAMMER, VERSION XX
ENTER V,C,P,F,S,U: C
TYPE#: 2716(CR)
LOGIC=P,N?: P      PAGE#=08(CR)    . . First half=pg 8-F
DONE!
PROM PROGRAMMER, VERSION XX        . . Insert higher address 2716
ENTER V,C,P,F,S,U: C
TYPE#: 2716(CR)
LOGIC=P,N?: P      PAGE#=10(CR)    . . Second half=pg 10-17
DONE!
PROM PROGRAMMER, VERSION XX        . . Remove 2716, insert 2732, set S1 low
ENTER V,C,P,F,S,U: P
TYPE#: 2716(CR)
LOGIC=P,N?: P
(D)ISK OR (R)AM? R     PAGE?: 08(CR)
DONE!
REPROGRAM 2716: N                  . . Set S1 high
PROM PROGRAMMER, VERSION XX
ENTER V,C,P,F,S,U: P
TYPE#: 2716(CR)
LOGIC=P,N?: P
(D)ISK OR (R)AM? R    PAGE#=10(CR) . . Second half of data
DONE!
REPROGRAM 2716? N                  . . Entire 2732 complete
```

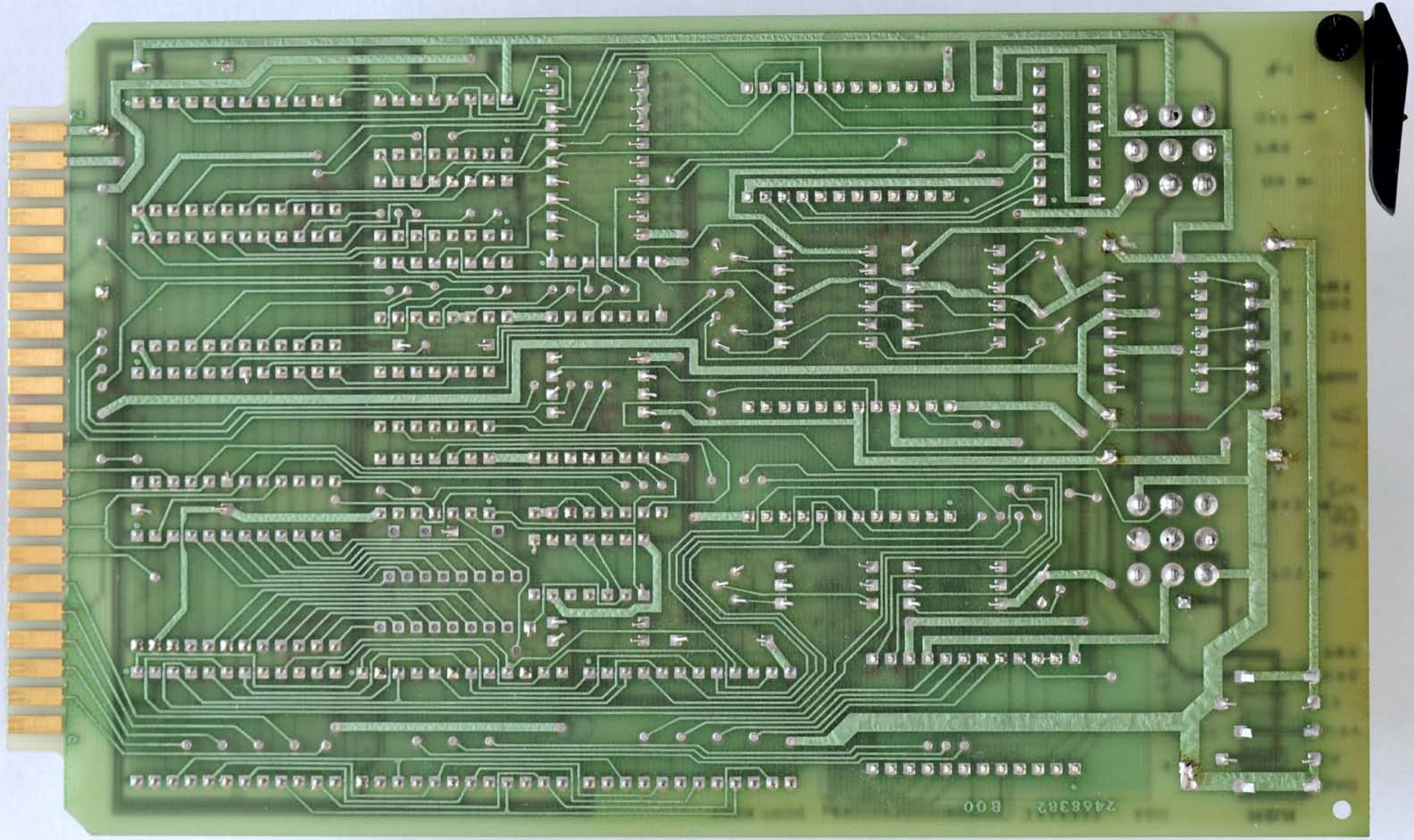
Reference:

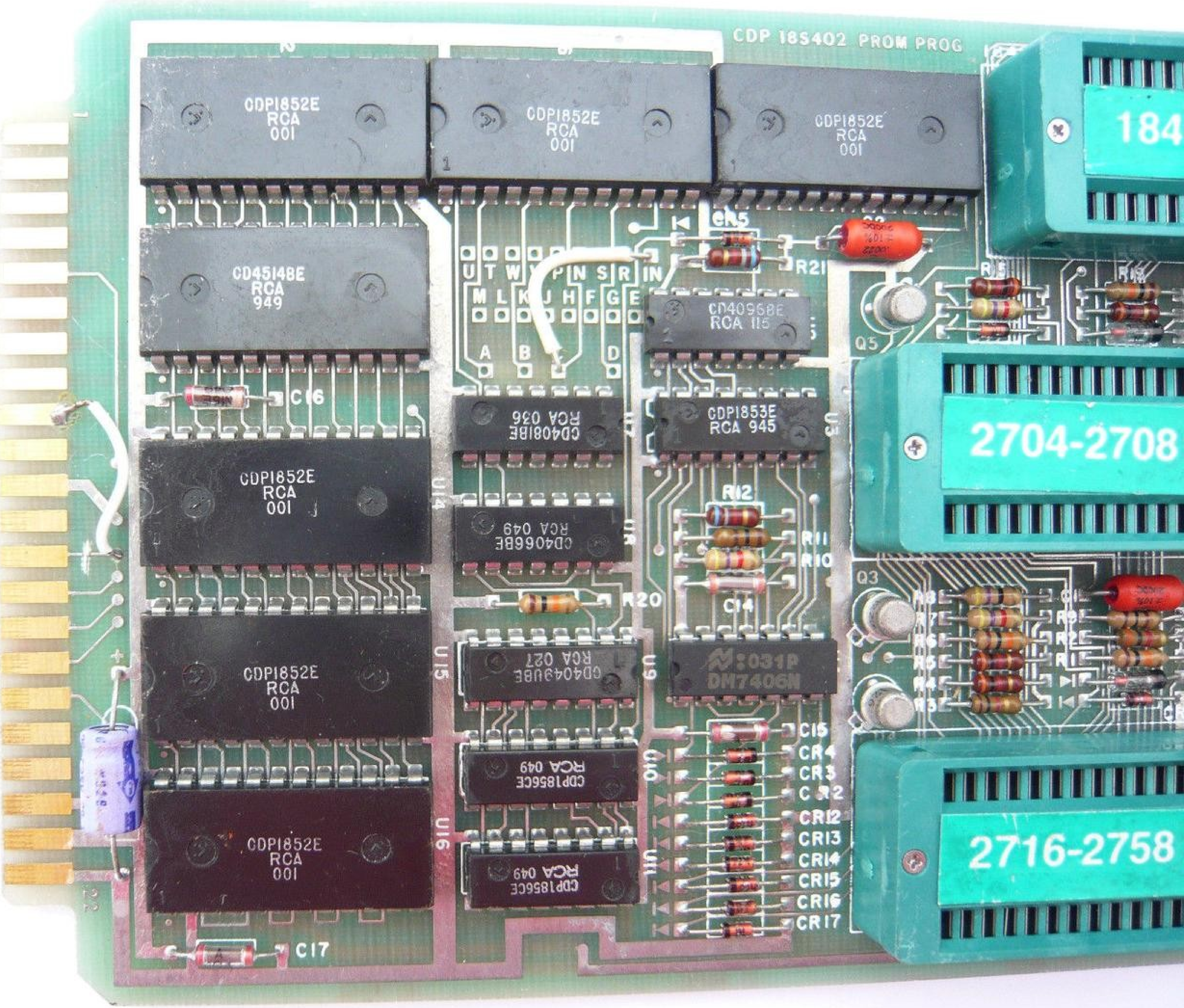
1. The operating procedure for the 2716 is given in: "Operator's Manual for PROM Programmer CDP18S480," RCA Solid State publication MPM-222A.

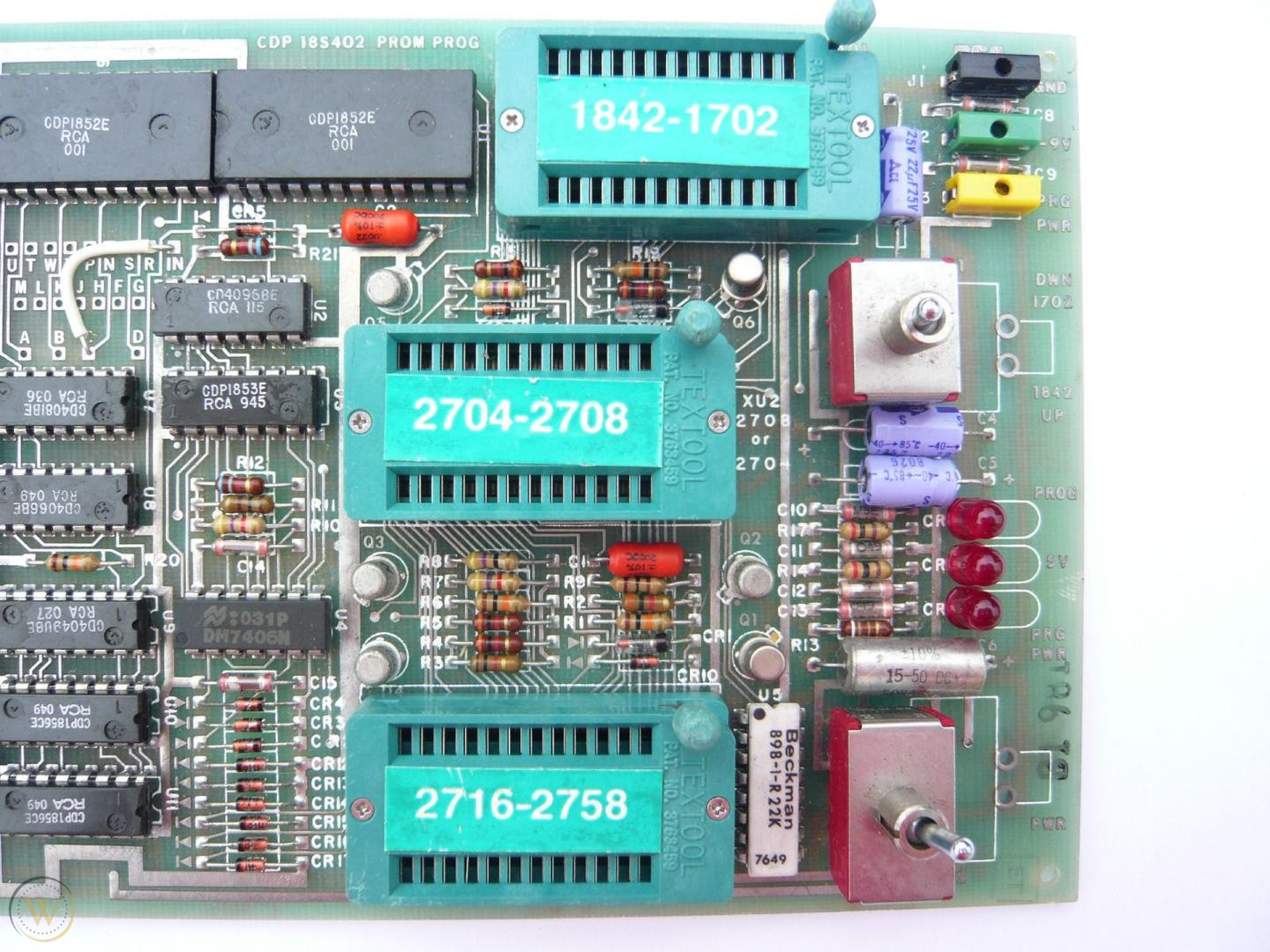

GND
9V
PRG PWR
C8
C9
J1
J2
J3
S1
S2
PROG
5V
PRG PWR
CR7
CR8
CR9
C10
R17
C11
R14
C12
C13
R13
XU2
2708
or
2704
CR1
CR10
U5
1842 / 1702 - XU1
R19
R18
CR6
R15
R16
CR11
C1
R9
R2
R1
R8
R7
R6
R5
R4
R3
XU3 2758 - 2716
PROM PROG
U1
C2
CR21
CR5
U2
U3
R11
R10
R12
C14
U4
C15
CR4
CR3
C32
CR12
CR13
CR14
CR15
CR16
CR17
U6
U7
U8
R20
U9
U10
U11
U12
U13
C16
U14
U15
U16
C17


2168382 B00


RCA
PROM PROG
1842-1702
2704-2708
2716-2758
XU2 2708 or 2704
XU3 2758 - 2716
CDP1852E RCA 210
CD4514BE RCA 206
CD4096BE RCA 123
CDP1853E RCA 201
CDP1856E RCA 206
BECKMAN 898-1-R22K
GND
-9V
PRG
PROG
5V
PRG POR
1702
1842
ON
PWR
OFF
A1702




CDP 18S402 PROM PROG
CDP1852E RCA 001
CDP1852E RCA 001
CDP1852E RCA 001
CD4514BE RCA 949
CD4096BE RCA 115
U T W V P N S R IN
M L K J H F G E
A B C D
CR5
R21
R15
Q5
184
2704-2708
2716-2758
C16
CD4081BE RCA 036
CDP1853E RCA 945
U14
CDP1852E RCA 001
CD4066BE RCA 049
R12
R11
R10
C14
R20
Q3
R8
R7
R6
R5
R4
R3
U15
CDP1852E RCA 001
CD4049UBE RCA 027
DM7406N 031P
C15
CR4
CR3
CR2
CR12
CR13
CR14
CR15
CR16
CR17
CDP1856CE RCA 049
CDP1856CE RCA 049
U16
CDP1852E RCA 001
C17


CDP 18S402 PROM PROG
1842-1702
2704-2708
2716-2758
TEXTOOL
PAT. NO. 3763459
CDP1852E
RCA
001
CD40968E
RCA 115
CDP1853E
RCA 945
CD4081BE
RCA 036
CD4066BE
RCA 049
CD4043UBE
RCA 027
DM7406N
CDP1856CE
RCA 049
Beckman
898-1-R 22K
7649
XU2
2708
or
2704
GND
+9V
PRG
PWR
DWN
1702
1842
UP
PROG
5V
PRG
PWR
PWR
15-50 DC
Q1
Q2
Q3
Q5
Q6
U2
U3
U4
U5
J1
CR1
CR5
CR10
C1
C4
C5
C6
C8
C9
C10
C11
C12
C13
C14
C15
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R17
R19
R20
R21