ELECTRONICS

System 2000

If you don't read the manual......
at least read this sheet!

9 steps to get your programmer going in the fastest time.

1. Unpack programmer, programming module and leads.

2. Plug programming module into programmer.

3. Check voltage on label on back and connect mains.

4. Switch on programmer

5. Ignore message 'Please register'. Press key marked 'I/O'. Display should read

```
Remote control...
                    LOCAL
```

6. Connect the D-type to D-type lead to the programmer socket marked PARALLEL and a PC parallel printer port.

7. Insert the CD and follow the instructions.

8. PROMDRIVER will now automatically register your new Programmer.

9. You are now free to run the programmer in remote mode or to press the LOCAL key and use it in stand-alone mode.

IMPORTANT INFORMATION
Please take time to read Appendix I. This explains important facts about software support, and socket lifetime.

SYSTEM 2000

ELECTRONICS

MQP ELECTRONICS

DEVICE PROGRAMMER SYSTEM 2000

USER MANUAL

MQP Electronics Ltd
Park Road Centre
Malmesbury
Wiltshire
SN16 0BX UK

Tel: +44 (0)1666 825666
Fax: +44 (0)1666 825141

e-mail: sales@mqp.com
W.W.W.   http://www.mqp.com

Issue 1 - 08/92
Issue 2 - 05/94
Issue 3 - 03/95
Issue 4 - 01/96
Issue 5 - 04/96
Issue 6 - 03/97
Issue 7 - 06/01

ELECTRONICS

Table of Contents

1 Introduction

Overview

Thank you for buying this Programmer. We hope that it meets with your requirements and that it will go on serving you for many years to come. We have designed this equipment to be fast and reliable; in fact we believe that it programs 16 EPROMs faster than any other programmer currently on the market.

Programmers of this versatility are the subject of continuous technical review and the manufacturer reserves the right to change details of the specification for the purposes of product improvement.

This manual describes in detail the operation of the programmer system.

The keyboard arrangement is described in Section 2. Sections 3 to 13 explain the operation of the programmer in stand-alone mode.

Section 14 describes in detail how to operate the programmer in remote mode with our PROMDRIVER software package. It also explains how to use a PC to control pre-programmed production tasks, using Batch mode.

The Appendices include error messages hardware details and methods of adjusting the settings of the programmer to suit yourself.

The list of programmable devices can be found in PROMDRIVER's TYPE SELECTION menu.

Setting up your Programmer

Carefully remove the Programmer from its packaging. You will have been supplied with a programmer chassis with keyboard and display. The Ram module, if you have ordered one, will already be fitted (under the keyboard). There will also be one or more programming modules. Fit one of these into the recess in the main unit, gently pressing it until the connectors engage correctly.

Most programmers in the range automatically select the correct mains voltage (110V or 220/240V). However, check the label on the rear of the chassis to confirm that your local voltage meets the requirement. Using the lead provided, connect the programmer to the mains. The on-off switch is on the back of the programmer, at the left hand side.

Please read appendix D of this manual. It lists some DON'Ts when programming.

ELECTRONICS

2. Keyboard Arrangement

The keyboard has been ergonomically designed to allow access to a wide range of functions in the smallest number of key presses. In most cases it should not be necessary to refer to a manual, as the use of the programmer should be intuitive.

The main programming commands are single key presses, and the associated keys are emphasised by being larger than the others, with the all-important PROGRAM key largest of all. This will enable untrained personnel to use the programmer with no difficulty in a production environment. A further aid to production is the ability to lock the settings, to prevent accidental changes to them during a production run.

The keyboard is a rubber keymat type with in-built tactile feedback. In addition an optional audible key-click is provided. This provides a positive feel combined with the ability to do fast and reliable numerical entry in EDIT mode.

All functions which are not single key functions are accessed by pressing the FUNC key, when the display will guide the user to the required function in a very small number of key presses.
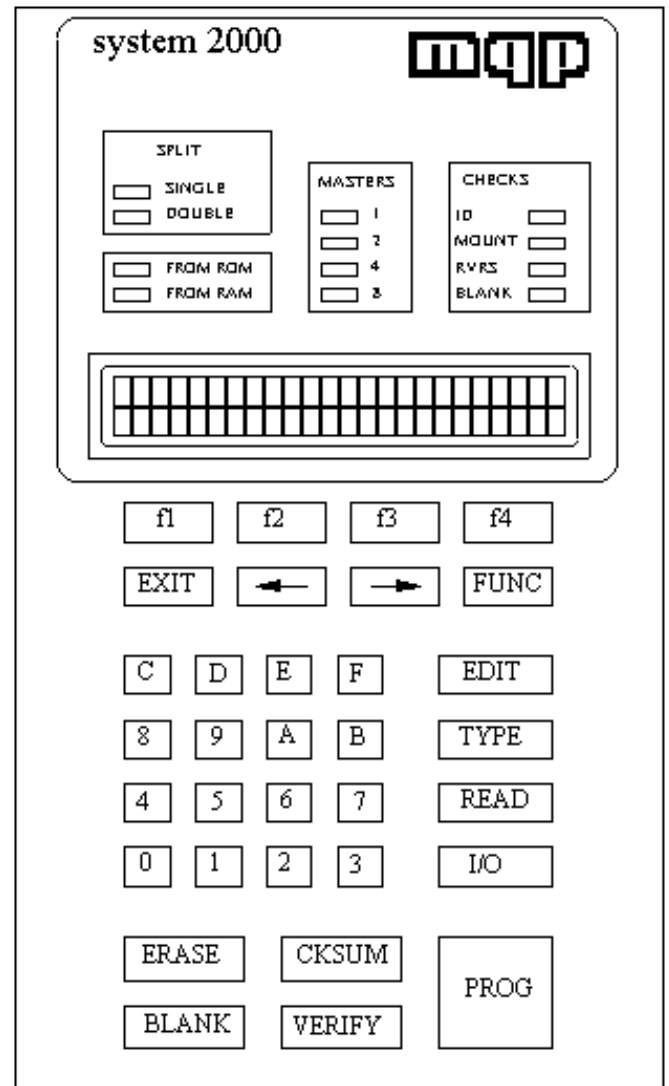
A system of "soft keys" is used to access editing and settings functions; keys f1 to f4 take on the meaning indicated on the second line of the display. In many cases selections can be changed by means of two cursor keys ← and →. The display will indicate when these may be used. The EXIT key may always be used to exit a function without activating it.

ELECTRONICS

Socket Leds

Next to each programming socket is a status LED. This is used to indicate the results of any operation on the devices in that socket. Each LED can represent a number of different states as follows:

OFF

No operation has taken place.

ON WITH SHORT FLASHES OFF

Operation in progress.

ON
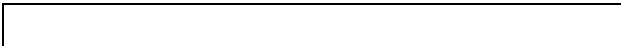
Operation was successful

SLOW FLASH

Operation failed.

FAST FLASH

Device or mount error

SHORT FLASH

Socket empty

DOUBLE FLASH

I D error

Sounder

High frequency

Key press and automatic key repeat after holding down the key for half a second (selective keys only)

Medium frequency

Success tone

Low frequency

Error tone

ELECTRONICS

On switching on

If RAM is fitted it is cleared to the current blank value.

```
Clearing RAM
■■■□□□□□□□□□□□
```

By default a maximum of 8 Megabits is cleared on start up, but this setting can be changed to clear all RAM on start up (see MISC. under the FUNC key).

The currently selected device is displayed.

```
INTEL 27C256
```

After any operation, PROG, BLANK, etc a report is given. You can also see this report later by pressing the REPORT key.

```
INTEL 27C256
REPORT
```

If 'Empty Socket Detect' is enabled (see MISC. under the FUNC key) the word REPORT will not be displayed after all devices have been removed from the sockets.

Certain devices have security and other special functions that can be programmed automatically after normal programming is finished. These are shown on the quiescent display as a number if enabled or as a dash if disabled. See SPECL under the FUNC key for further details.

```
INTEL 87C51        S12--
REPORT
```

This display shows that special functions 1 and 2 are enabled, 3 and 4 are disabled.

On pressing REPORT key

This restores the socket LEDs and display to their state after last operation in case the user needs to refer to the last REPORT again. If 'Empty Socket Detect' is enabled, no report on the last operation will be possible after all devices have been removed from the sockets.

```
REPORT ON LAST OPERATION
                      OK
```

Press OK to acknowledge the message and the last report will be shown.

3. TYPE key

On pressing TYPE key

To cater for the differing requirements of users, four methods of device type selection are provided:

TYPE
- AUTO   Select type by Silicon Signature
- NAME   Select type by Manufacturer and Name
- CODE   Select type by 4 digit Device code
- RECALL Select type by recalling stored type

Any of these methods can be accessed by a single key press, after pressing the TYPE key.

```
           TYPE SELECTION
AUTO    NAME    CODE    RECALL
```

Press the key under the method of type selection required.

On pressing AUTO

If AUTO is pressed, the programmer will attempt to identify the device in the occupied socket nearest A1. After attempting to identify the device, the socket LEDs will indicate the success or failure of the operation on each device.

- If the first device is identified the socket LED will show steady ON.

- Any other sockets containing identical types will also show steady ON.
- Any sockets containing different types that can be detected will show DOUBLE FLASH cadence.

- In the case of any type error, an error tone will sound to alert the operator, and the sockets with unsuccessful devices will be accompanied by a flashing LED.
- LEDs next to empty sockets will show a distinctive short flash.

The display will give a report, as follows. If all devices in the sockets are of identical types, the display will indicate the type:

```
NEC    27C8000
REPORT                    OK
```

On pressing OK (or removing all devices if 'Empty Socket Detect' is enabled) the display will revert to the initial one, and the LEDs are extinguished.

If there was a problem with the identification, an error message is given:

```
DEVICE OR MOUNT ERROR
                          OK
```

```
ID ERROR
                          OK
```

```
REVERSE INSERTION ERROR
                          OK
```

```
MOUNT ERROR
                          OK
```

```
ADDRESS LINE ERROR
                          OK
```

```
TYPE NOT RECOGNISED
                          OK
```

ELECTRONICS

```
ERROR: MIXTURE OF TYPES
                          OK
```

```
ALL SOCKETS EMPTY
                          OK
```

On pressing the key under OK, the display will revert to display the currently selected type and the LEDs will be extinguished.

Please see appendix D for some special warnings concerning AUTO Typesetting.

On Pressing NAME

```
NS    27C010
MANU  TYPE              OK
```

If NAME is pressed the programmer will allow you to specify the device by manufacturer and type number. Pressing MANU or TYPE will allow the cursor keys to change manufacturer or type. Cursor symbols will flash next to the word MANU or TYPE to indicate which will be affected by the cursor keys. Pressing OK will select the currently displayed type.

On pressing CODE

```
ENTER 4 DIGIT CODE:   0000
-----:-------------   OK
```

Each device type has been assigned a Device code, for easy reference. Enter this code to select the device. The device type will be displayed for reference,

or -----:-------- if the code is not recognised. The device is the 4 digit hex code found under the ALG key when the device has been selected by one of the other means.

On pressing RECALL

```
RECALL SETTING FROM 0-F
```

```
KEY 0:    TI    27C240
CLEAR                   OK
```

```
KEY 0: *NO TYPE*
```

No information has been stored under this key.

```
KEY 8: INCORRECT MODULE
```

The information stored under this key was for a device that can only be programmed using a different programming module, so it cannot be loaded with this module. Pressing any of the digit keys 0-F results in that key's stored setting being displayed. Press OK to accept, or another digit key. See STORE under the FUNC key how to save settings.

Device Notes

After a device has been selected, the top level display will show the type. If there is any special information to be mentioned when making this selection then an asterisk will appear next to the device name, and the F4 soft key will be labelled with the function *NOTE.

```
INTEL 27C513 *
REPORT              *NOTE
```

Pressing this key will result in a display of the special note. Some examples follow:

```
OLDER WITH INCORRECT ID
                          OK
```

Sometimes a device has been manufactured with the incorrect silicon signature. Choosing this device selection will prevent checking of the signature, thus allowing it to be read or programmed.

```
OLDER SLOWER ALGORITHM
                     OK
```

The manufacturer has manufactured devices with the same name but different algorithms. Choosing this device selection by name gives access to the older (slower) algorithm on those occasions when it is necessary.

```
RAM MASTER ONLY ALLOWED
                     OK
```

Certain devices may not be programmed from devices in the copy socket but must be read into the RAM buffer from socket A1.

```
WARNING: 21V VPP
                     OK
```

This represents a device with the same name as other manufacturers devices having a lower programming voltage (e.g. 12.5V). It must therefore only be used with the correct device, otherwise device damage may result.

```
PIN CONVERTER REQUIRED
                     OK
```

Some device types are catered for on the programmer by using package converters. These devices are identified by this note.

ELECTRONICS

4. EDIT key

On pressing EDIT key

You will be given the option of display (no changes allowed) or edit, on either the memory buffer (if fitted) or a device in one of the sockets. Editing a device directly is mostly appropriate to EEPROMs, Emulators and similar devices where a change is not irrevocable.

EDIT

   EDIT BUFFER   Edit contents of RAM buffer

          COPY   Copy a block of data

          FILL   Fill a block of data

          NEW   Clear a block of data

          INVERT   Invert the polarity of a block of data

          SEEK   Search for a sequence of bytes

          INSERT   Put into insert mode

          DELETE Put into delete mode

          KILL   Move data down

   DISPLAY BUFFER   Examine contents of RAM buffer (can't be modified)

   EDIT DEVICE   [Reserved for future function]

   DISPLAY DEVICE   Examine contents of device (can't be modified)

```
--BUFFER---    ---DEVICE----
 DISP   EDIT    DISP   EDIT
```

On pressing EDIT BUFFER

A check is made that a RAM module is fitted to the programmer; if not an error message is displayed:

```
No memory available
                      OK
```

Otherwise the normal buffer editing display appears:

```
000000 FF 41 42 FF      .AB.
ADDR DATA             ED fn
```

(ED fn means 'select special edit function')

Four bytes are displayed at a time, along with the ASCII equivalents.

The cursor will indicate either a data or an address character which can currently be changed by over-typing using the digit keys 0-F. Switching between changing address and data can be done by pressing the key under ADDR or DATA.

Moving the cursor beyond the end of the current row of data automatically

---

advances the address to the next row. Holding the cursor key down will result in the address changing at an increasing rate.

Advanced edit functions

More advanced edit functions are accessed by pressing the key under ED fn (edit function). Functions available are displayed four at a time:

```
← SELECT EDIT FUNCTION  →
 COPY   FILL   NEW   INVERT
```

```
← SELECT EDIT FUNCTION  →
 SEEK  INSERT  DELETE  KILL
```

Pressing a cursor key accesses a further 4 functions. Pressing the key under the function, selects it.

COPY

```
COPY:   000000-000000
  TO:   000000-000000   OK
```

This editing function allows a block of data to be copied from a range of addresses to another range. The cursor keys allow all the addresses concerned to be modified. Note that the 'from' and 'to' ranges will automatically be kept the same size by the programmer.

FILL

```
FILL: 000000-000000
WITH: 00              OK
```

This editing function fills a range of memory addresses with a constant value. The cursor keys allow all the parameters involved to be changed.

NEW

```
NEW: ALL DATA WILL BE
     DELETED          OK
```

This editing function allows all data in the buffer to be set to the currently defined blank value.

INVERT

```
INVERT:   000000-000000
                       OK
```

The polarity of a block of data can be inverted by this editing function.

SEEK

This edit function allows a sequence of bytes to be located in the buffer. You may specify the range over which the search should take place and up to 4 bytes to be searched for. It is possible to specify the search string in ASCII by pressing the → key.

As the keyboard does not support the complete ASCII character set, the ASCII values are set using the DECR and INCR (f2 & f3) keys After specifying the string press → to return to the hexadecimal display. Press OK to start the search. You will be given the address at which the string was found, or an error tone. Press OK again to search for the next occurrence of the string.

INSERT

Inserting involves moving data in all locations from the cursor upwards to an address one higher than before. This operation is done in the normal editing display, but as a safety

precaution, only after it has been enabled by selecting the INSERT mode to be ON.

```
INSERT: ON
CHANGE                    OK
```

```
INSERT: OFF
CHANGE                    OK
```

On returning to the main edit display the f3 key is now defined as INS (insert).

```
000000   FF FF FF FF
ADDR   DATA   INS    ED fn
```

Pressing this key will cause all data in the buffer to move up, starting at the cursor address. This operation will take a second or so.

DELETE

```
DELETE: ON
CHANGE                    OK
```

This is the opposite of insert and accessed in a similar way. The currently defined blank value is written into the top location in memory.

```
000000   FF FF FF FF
ADDR   DATA   DEL    ED fn
```

KILL

```
KILL: 000000-000000
                          OK
```

This is a similar operation to delete except that it operates on a block of data bytes. All data above the block is moved down to take its place. The currently defined blank value is written into the top locations in memory.

On pressing DISP BUFFER

Display buffer is similar to Edit buffer except that no changes can be made the buffer contents.

On pressing EDIT DEVICE

On pressing DISP DEVICE

Edit or display device are similar to edit or display buffer except that you must select the device in question.

```
    SELECT THE SOCKET
  A1      A2      A3      A4
```

Possible error messages are:

```
No type selected
                          OK
```

or:

```
DEVICE/SOCKET FAULTY
                          OK
```

---

5. I/O key

On pressing I/O key

The I/O key is used to allow input and output through the parallel and serial ports located on the back of the programmer.

If communications have been set up to HOST IS A PC (see COMMS, HOST under the FUNC key) then control will pass to the PROMDRIVER program running on a PC, and the following is displayed:

```
Remote control
                      LOCAL
```

When it is possible to return control to the programmer, this can be achieved by pressing the LOCAL key.

If communications have been set up to HOST IS A NON-PC, you will be offered the choice of inputting or outputting data via the selected port.

```
I/O
├──INPUT
│     ├── FORMAT Select Input file format
│     ├── PARAMS      Select first RAM and file
│                     address
├──OUTPUT
      ├── FORMAT       Select Output file format
      ├── PARAMS       Select RAM range and first
                       file address
```

It is assumed that you have connected the appropriate port via a suitable lead to the communicating device, and set up the correct parameters for the transfer (under FUNC, COMMS). See Appendix H for connections details. The safest procedure is to enable the receiving end first and then the sending end, although if using hardware handshaking it should be possible to start either end first.

```
   SELECT I/O FUNCTION
INPUT OUTPUT
```

This display allows you to choose to input or output a file.

On pressing INPUT

```
INPUT A FILE
FORMAT PARAMS        START
```

will be displayed. You may set up the file format and address range before starting.

Pressing FORMAT allows you to set the file format to one of:

INTHEX   Intel hexadecimal
INTHX∗   Special Intel hexadecimal
MOTHEX   Motorola Hexadecimal

The special Intel Hex format includes a stream of nulls at the beginning and end of the file. Press OK when you have chosen the appropriate setting.

Pressing PARAMS allows you to choose the start RAM and file addresses.

```
 ADDRS   000000     00000000
CLEAR   RAM      FILE      OK
```

The File Address defines the default for the first address in the file that will be stored in RAM. The RAM Address is the first address at which this is stored. These addresses can be changed by using the hexadecimal keypad.

Pressing OK will return you to the INPUT A FILE screen. Press START to commence input.

On pressing OUTPUT

```
OUTPUT A FILE
FORMAT PARAMS       START
```

The FORMAT choices are the same as for INPUT above.

Selecting PARAMS allows you choose the RAM range and the first File Address.

```
000000    000000    00000000
 FROM       TO    F.ADDR   OK
```

Pressing OK will return you to the OUTPUT A FILE screen. Press START to commence outputting the file. If the special Intel Hex format has been selected 50 nulls are added before and after the file.

While a file is being transferred, a changing number will indicate the current file address every 256 bytes.

6. Socket Checks

When any key is pressed which starts a read or write operation on the sockets, the sockets are first subjected to a series of checks to validate the devices present.

```
NO DEVICE TYPE SELECTED
                      OK
```

```
COPY SOCKET EMPTY
                      OK
```

```
RAM MASTER ONLY ALLOWED
                      OK
```

```
LAST OPERATION ABORTED
                      OK
```

```
INSUFFICIENT RAM
                      OK
```

```
CANNOT READ - ROM MASTER
                      OK
```

```
ADDRESS LINE ERROR
                      OK
```

```
ALL SOCKETS EMPTY
                      OK
```

```
PIN CONVERTER REQUIRED
                      OK
```

Various checks are normally performed before each operation. Most of these checks can be disabled by the user, where found necessary.

The checks are done in the following order.

Verify pin count

If incorrect, the LEDs on all sockets indicate FAST FLASH, the ERROR tone sounds and the display shows:

```
DEVICE OR MOUNT ERROR
                      OK
```

and the operation is aborted. The LEDs go out when OK is pressed.

Verify ID all sockets (if not disabled)

If incorrect the LEDs on the sockets with wrong ID indicate DOUBLE FLASH, all others are OFF, the ERROR tone sounds and the display shows:

```
ID ERROR
                      OK
```

and the operation is aborted. The LEDs go out when OK is pressed.

Reverse insertion check (if not disabled)

Checks for devices that have been inserted incorrectly, e.g. in the wrong part of the socket or upside down. If incorrect, the LEDs on all sockets indicate FAST FLASH, the ERROR tone sounds and the display shows:

```
REVERSE INSERTION ERROR
                      OK
```

and the operation is aborted. The LEDs go out when OK is pressed.

Mount check (if not disabled)

Checks that the devices have been inserted correctly in the sockets. If incorrect the LEDs on sockets with errors indicate FAST FLASH, all others are OFF, the ERROR tone sounds and the display shows:

```
MOUNT ERROR
                          OK
```

and the operation is aborted. The LEDs go out when OK is pressed.

Caution

During any operation all socket LEDs will be ON with short flashes OFF. Devices should not be inserted or removed at this time or damage may result.

Notes On Disabling Checks

Reverse Insertion Check

The reverse check makes use of the property of the memory chip pins to detect reverse insertion without damaging the device. Some manufacturers' devices may give a false indication owing to their internal construction. If it is known that good devices falsely give reverse indication the detection should be disabled to allow normal operation to take place.

ID check

Devices that do not have IDs will automatically ID check disabled. Don't do Auto Type if there is no ID in the part.

Mount check

Disabling mount check is primarily available for servicing checks and should not normally be done.

---

7. READ key

On pressing READ key

Starting reading is a single key operation. The PROM or PROMs will be read and a visual indication of progress will be given on the display in the form of a thermometer-like bar. Reading is automatically preceded by the usual pre-operation checks.

```
READING MASTER...
■■■■□□□□□□□□□□
```
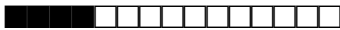
```
A1 READ
cksum:FC00          OK
```

Displaying the checksum has been included as a helpful indication that the correct device has been read. It is only displayed when no splits and only one master have been selected. The word 'ALERT' and the 🔑 icon will be displayed if master checksum monitoring is currently enabled.

```
READING MASTER...
■■■■□□□□□□□□□□
```

```
A1/2/3/4/5/6/7/8 READ
                    OK
```

```
NO DEVICES READ
                    OK
```

After reading, the socket LEDs show which sockets were read. LEDs next to sockets not required to be read will be turned off. Sockets not read (because they were not masters) will show a slow flash, and successfully read devices will show a steady ON

LED. Sockets not read because they were empty will show a short flash.
If not enough RAM is fitted, all LEDs will FLASH, an error tone will sound and the display will show:

```
INSUFFICIENT RAM
                    OK
```

The display and LEDs will be cleared when OK is pressed.

8. PROG key

On pressing PROG key

```
PROGRAMMING...
■■■■■□□□□□□□□□□
```

Starting programming is a single key operation. The PROM or PROMs will be programmed and a visual indication of progress will be given on the display in the form of a thermometer-like bar. Programming is automatically preceded by the usual pre-operation checks, and a blank check, and followed by a verification (see below).

If master checksumming is enabled, this is checked against the checksum of the current master, either RAM or ROM. If different, the following error message is displayed:

```
MASTER CHECKSUM WRONG
                      OK
```

and programming is aborted. Otherwise a blank check is performed (if not disabled) and the display will show:

```
BLANKCHECK...
■■■■□□□□□□□□□□□
```

Any failures will be remembered and these sockets will not be programmed but programming on other sockets will proceed.

After programming and verifying, the socket LEDs will indicate the success or failure of the operation on each device. In the case of failure, an error tone will sound to alert the operator, and the sockets with unsuccessful devices will be accompanied by a flashing LED. LEDs next to empty sockets will show a distinctive short flash, and successfully programmed devices will show a steady ON LED.

The display will give a report, as follows. If all devices in the sockets have been successfully programmed the display will show:

```
ALL PARTS PROGRAMMED OK
cksum:FCC0            OK
```

On pressing the key under OK, the display will revert to display the currently selected type and the LEDs will be extinguished.

Displaying the checksum has been included as a helpful indication that the correct device has been selected when copying from the copy socket, or that RAM has not been changed when programming from there. It is only displayed when no splits and only one master have been selected. The 🔑 icon will be displayed against the checksum if master checksum monitoring is enabled.

If there have been any failures a full report on each socket is given. Some typical displays are shown below. Notice the flashing cursor keys indicating that the cursor keys are active and allow backwards or forwards movement through the reports.

Pressing the NEXT key puts the next socket report on the display.

```
A1: VER+ FAIL @ 000000
                    NEXT
```

Indicates a verification failure on a raised Vcc level.

```
A1: VER- FAIL @ 000000
                    NEXT
```

Indicates a verification failure on a lowered Vcc level.

```
A1: VER  FAIL @ 000000
                    NEXT
```

Indicates a verification failure when just one Vcc level has been selected.

On the last report, pressing the OK key clears the LEDs and the display reports.

At any time, the report sequence can be terminated by pressing EXIT. This causes the LEDs and display to assume the quiescent state. Additionally the single key functions (such as PROG) will still work immediately when the display is showing a report, so single key press programming is always possible.

ELECTRONICS

9. VERIFY key

On pressing VERIFY key

```
VERIFYING AT  4.50V...
■■■■■□□□□□□□□□□
```

If master checksumming is enabled, this is checked against the checksum of the current master, either RAM or ROM. If different, the following error message is displayed:

```
MASTER CHECKSUM WRONG
                        OK
```

and verification is aborted. Assuming no errors are discovered, the usual socket checks are performed and. verification proceeds according to previously set VCC limits, with a bar indicating progress. On most devices, the verification process is extremely fast owing to the special circuitry incorporated into the programmer.

After programming and verifying, the socket LEDs will indicate the success or failure of the operation on each device. In the case of failure, an error tone will sound to alert the operator, and the sockets with unsuccessful devices will be indicated by a flashing LED. LEDs next to empty sockets will show a distinctive short flash, and successfully programmed devices will show a steady ON LED.

The display will give a report, as follows. If all devices in the sockets have been successfully verified the display will show:

```
ALL PARTS VERIFIED OK
Cksum:FCC0              OK
```

The 🔑 icon will be displayed against the checksum if master checksum monitoring is currently enabled.

On pressing the key under OK, the display will revert to display the currently selected type and the LEDs will be extinguished.

If there have been any failures a full report on each socket is given. Some typical displays are shown below. Flashing cursor keys indicate that the cursor keys are active and allow backwards or forwards movement through the reports.

```
A1: VERIFIED OK
                      NEXT
```

Pressing the NEXT key puts the next socket report on the display.

```
A1: VER+ FAIL @ 000000
                      NEXT
```

Indicates a verification failure on a raised Vcc level

```
A1: VER- FAIL @ 000000
                      NEXT
```

Indicates a verification failure on a lowered Vcc level

```
A1: VER  FAIL @ 000000
                      NEXT
```

Indicates a verification failure when just one Vcc level has been selected.

On the last report, pressing the OK key clears the LEDs and the display reports.

At any time, the report sequence can be terminated by pressing EXIT. This causes the LEDs and display to assume the quiescent state. Additionally the single key functions (such as PROG) will still work immediately when the display is showing a report, so single key press programming is always possible.

10. BLANK key

On pressing BLANK key

```
BLANKCHECK...
■■■■■□□□□□□□□□
```
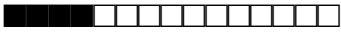
Performing a BLANK check is a single key operation. This check will normally be done as part of the programming operation (unless disabled).

Before the operation the usual socket checks are performed. Assuming no errors are discovered, blank checking proceeds according to previously set verify VCC limits, with a bar indicating progress. On most devices, the blankcheck process is an extremely fast one owing to the special circuitry incorporated into the programmer.

After blank checking, the socket LEDs will indicate the success or failure of the operation on each device. In the case of failure (a non-blank device) an error tone will sound to alert the operator, and the sockets with non-blank devices will be accompanied by a flashing LED. LEDs next to empty sockets will show a distinctive short flash, and successfully programmed devices will show a steady ON LED.

The display will give a report, as follows. If all devices in the sockets have been successfully verified the display will show:

```
ALL PARTS ARE BLANK
                   OK
```

On pressing the key under OK, the display will revert to display the currently selected type and the LEDs will be extinguished.

If there have been any failures a full report on each socket is given. Some typical displays are shown below. Notice the flashing cursor keys indicating that the cursor keys are active and allow backwards or forwards movement through the reports.

```
A1: BLANK
                 NEXT
```

```
A1: B/C FAIL @ 000000
                 NEXT
```

Pressing the NEXT key puts the next socket report on the display.

On the last report, pressing the OK key clears the LEDs and the display reports.

At any time, the report sequence can be terminated by pressing EXIT. This causes the LEDs and display to assume the quiescent state. Additionally the single key functions (such as BLANK CHECK) work immediately when the display is showing a report, so single key press operation is always possible.

Note on VCC used for blankcheck:

Blankcheck is performed at a VCC which is most likely to reveal inadequate erasure. Typical EPROMs will be checked at -5% or the current negative verify level whichever is lower.

ELECTRONICS

11. ERASE key

On pressing ERASE key

```
ERASING...
████□□□□□□□□□□
```
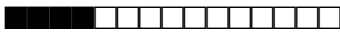
Only applies to electrically erasable devices.

Erasing a device is a single key operation. The PROM or PROMs will be erased and a visual indication of progress will be given on the display in the form of a thermometer-like bar. Erasure will automatically be followed by a verification to check success.

After erasing and verifying, the socket LEDs will indicate the success or failure of the operation on each device, in the usual way.

```
ALL PARTS ERASED
                           OK
```

```
A1: ERASE FAILURE
                           OK
```

ELECTRONICS

12. CKSUM key

On pressing CHECKSUM key

```
READING CHECKSUM...
■■■■□□□□□□□□□□
```
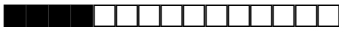
This operation displays the checksum of the master ROM or RAM, and of each of the devices in the programming sockets.

Two checksum options are offered: 4 digit, and 8 digit checksums. 4 digit checksums are traditional but are not adequate checks of successful programming on larger devices. Their main use is for fast identification of programmed devices; the 4 digit checksum is often used on a label on the device.

8 digit checksums lose no bits to overflow during the calculation and are more useful as a programming check. However, please note that the verification of every byte in a device is as fast on this programmer as calculating a checksum, and has the additional benefit of being done at high and low Vcc levels.

Before the operation the usual socket checks are performed. Assuming no errors are discovered, checksum calculation proceeds, with a bar indicating progress. On most devices, checksum calculation is extremely fast owing to the special circuitry incorporated into the programmer.

After programming and verifying, the socket LEDs will indicate which sockets contain devices. LEDs next to empty sockets will show a

distinctive short flash, and detected devices will show a steady ON LED. The display will give a report, as follows.

8 Digit Checksums

A full report on each socket is given. Some typical displays are shown below. Notice the flashing cursor keys indicating that the cursor keys are active and allow backwards or forwards movement through the reports.

```
RAM M1: C/S=01234567
                 NEXT
```

```
SKT M1: C/S=01234567
                 NEXT
```

Pressing the NEXT key puts the next report on the display.

On the last report, pressing the OK key clears the LEDs and the display reports.

At any time, the report sequence can be terminated by pressing EXIT. This causes the LEDs and display to assume the quiescent state. Additionally the single key functions (such as CKSUM) will still work immediately when the display is showing a report, so single key press operation is always possible.

The sequence for the display of Master and Socket checksums is as follows (empty sockets will be omitted):

1 master (RAM) sequence:
M1 A1 A2 A3 A4 A5 A6 A7 A8 B1 B2 B3 B4 B5 B6 B7 B8

2 master (RAM) sequence:
M1 A1 A2 A3 A4 B1 B2 B3 B4
M2 A5 A6 A7 A8 B5 B6 B7 B8

4 master (RAM) sequence:
M1 A1 A2 B1 B2
M2 A3 A4 B3 B4
M3 A5 A6 B5 B6
M4 A7 A8 B7 B8

8 master (RAM) sequence:
M1 A1 B1
M2 A2 B2
M3 A3 B3
M4 A4 B4
M5 A5 B5
M6 A6 B6
M7 A7 B7
M8 A8 B8

ROM master sequence:
C1 A1 A2 A3 A4 A5 A6 A7 A8

4 Digit Checksums

In this mode up to 5 checksums are displayed simultaneously. This is a typical display:

```
 M1      A1      A2      A3      A4
5678    5678    5678    -----   5678
```

In this exceptional case the function key f4 is not labelled. However, it still works as a NEXT/OK key as usual. The cursor and EXIT keys also have their normal functions.

The correct checksum for the devices is always shown at the left of the display. Empty sockets are indicated with the symbol ----.

The sequence for the display of Master and Socket checksums is as follows:

1 master (RAM) sequence:
M1 A1 A2 A3 A4
M1 A5 A6 A7 A8
M1 B1 B2 B3 B4
M1 B5 B6 B7 B8

2 master (RAM) sequence
M1 A1 A2 A3 A4
M1 B1 B2 B3 B4
M2 A5 A6 A7 A8
M2 B5 B6 B7 B8

4 master (RAM) sequence:
M1 A1 A2 B1 B2
M2 A3 A4 B3 B4
M3 A5 A6 B5 B6
M4 A7 A8 B7 B8

8 master (RAM) sequence:
M1 A1 B1
M2 A2 B2
M3 A3 B3
M4 A4 B4
M5 A5 B5
M6 A6 B6
M7 A7 B7
M8 A8 B8

ROM master:
C1 A1 A2 A3 A4
C1 A5 A6 A7 A8

---

13. FUNC key

These functions are explained on the following pages.

```
FUNC
  ├─STORE              Saves settings under a numeric key
  ├─RECALL             Recalls settings under a numeric key
  ├─RAM
  │    ├─ SPLIT                Settings for 8 bit devices on 16 or 32 bit busses
  │    ├─ SETS                 Settings for sets of different PROM patterns
  │    ├─ CKSUM
  │    │    ├─ BITS                   Set no. of digits displayed in checksums
  │    │    ├─ DIGITS                 Set size of data summed
  │    │    └─ MASTER                 Set and enable master checksum
  │    ├─ CLR                  Clear all RAM
  │    ├─ DISP                 Set size of data displayed
  │    ├─ EMPTY                Set RAM empty fill value
  │    └─ PATT(n)              Store either of two test patterns in all RAM
  ├─DEVICE
  │    ├─ ID                   Enable/disable id check
  │    ├─ MOUNT                Enable/disable mount check
  │    ├─ REVERSE              Enable/disable reverse check
  │    ├─ BLANK                Enable/disable blank check and set blank value
  │    ├─ VERIFY               Set verification Vcc limits
  │    ├─ SPECL
  │    │    ├─ AUTO                   Enable lock functions etc. of devices
  │    │    └─ MANUAL                 Perform lock functions etc. of devices
  │    ├─ CKSUM
  │    │    ├─ BITS                   Set no. of digits displayed in checksums
  │    │    ├─ DIGITS                 Set size of data summed
  │    │    └─ MASTER                 Set and enable master checksum
  │    ├─ Tacc                 NORMAL or SLOW device access time
  │    └─ ADDR                 Set reduced address space
  ├─COMMS
  │    ├─ HOST                 PC or non-PC host
  │    ├─ PORT                 Serial port parameters (set by host in PC mode)
  │    ├─ PRINTER USED?        Use parallel port for comms or printer
  │    └─ PRINTER PARAM        Set up printer (if enabled)
  │         ├─ INIT                   Set up initialisation string
  │         ├─ TERM                   Set up termination string
  │         └─ PAGEL                  Set up page length
  ├─MASTER
  │    ├─ ROM                  Set ROM (COPY skt.) to program and verify from
  │    └─ RAM                  Set RAM to program and verify from
  ├─INFO               Various useful programmer information displays
  ├─SOUND
  │    ├─ CLICKS               Enable/disable
  │    └─ TONES                Enable/disable
  ├─EETEST             Performs goods inwards test on re-programmable devs.
  ├─MISC.
  │    ├─ KEYBOARD LOCK        Locks/unlocks settings
  │    ├─ EMPTY SKT DETECT     Enable/disable
  │    ├─ START UP TESTS       Enable/disable
  │    └─ START: CLR ALL RAM   Enable/disable
  ├─ALG.               Prog. algorithm and info. of selected device
  ├─PRINT              Print device list
  ├─DEUTSCH | ENGLISH  Change display to other language
  └─STAT               Enable/disable programming statistics
```

ELECTRONICS

Functions accessed from the FUNC key

All functions not having a dedicated key are accessed by pressing the FUNC key. On pressing FUNC the main function categories are displayed. The cursor keys show other functions, four at a time.

```
SELECT FUNCTION
STORE   RECALL   RAM    DEVICE
```

```
SELECT FUNCTION
COMMS   MASTER   INFO   SOUND
```

```
SELECT FUNCTION
EETEST   MISC.    ALG.    PRINT
```

```
SELECT FUNCTION
DEUTSCH      STAT
```

Press the key under the required function. The organisation of functions can be seen from this overview chart.

STORE

```
SAVE SETTINGS UNDER 0-F
```

Current settings will be saved under the hexadecimal digit key required. These include the current type, split mode and number of masters, the RAM and DEVICE settings (including any master checksum) and the current language. On selecting a key, a check is made that this is where you want to store the current settings.

```
SAVE UNDER KEY 8?
                           OK
```

Press OK to accept, or another digit key. Press EXIT to abort.

RECALL

```
RECALL SETTING FROM 0-F

```

Pressing any of the digit keys 0-F results in that key's stored setting being displayed. On selecting a key, the settings that were previously stored under that key are displayed as well as the appropriate LED settings. Press OK to accept, or another digit key.

```
KEY 8:   INTEL 27C256
CLEAR                      OK
```

If these are the required settings, press OK. Alternatively, any of the single keys (e.g. PROG) can be used immediately and the new programmer settings will be used.

If these are not the required settings, other numeric keys can be pressed until the correct settings are selected.

On pressing CLEAR, a check will be made to confirm that you want to delete the current key settings.

```
KEY 8:   INTEL 27C256
  delete?                  OK
```

Press OK to clear the settings, EXIT to abort.

RAM

This function controls the Buffer RAM settings

SPLIT MODE   8/16/32 bits

SETS   8/16 sets of 1
4/8 sets of 2
2/4 sets of 4
1/2 sets of 8

CLEAR   all RAM

DATA DISP8, 16 or 32 bits

EMPTY FILL VALUE   00, FF or device default

CHECKSUM 8 bit or 16 bit, set master checksum

PATT(n) store either of two test patterns in all RAM

```
BUFFER RAM SETTINGS
SPLIT    SETS    CKSUM    CLR
```

```
BUFFER RAM SETTINGS
DISP    BLANK   PATT1    PATT2
```

Press the function key under the sub-function required.

SPLIT MODE Display

```
32 BIT (DOUBLE SPLIT)
CHANGE                      OK
```

```
16 BIT (SINGLE SPLIT)
CHANGE                      OK
```

```
8 BIT (NO SPLIT)
CHANGE                      OK
```

Press the key under CHANGE to switch data width between 8, 16 or 32 bit data. The number of sets will automatically be changed if necessary, as indicated on the display LEDs. Press EXIT to restore settings as on entry.

N.B. The split settings can only be changed in systems with modules MP2 or MP4.

SET MODE Display

```
8 SETS OF 1 MASTER
CHANGE                      OK
```

```
1 SET  OF 8 MASTERS
CHANGE                      OK
```

For module MP4, press the key under CHANGE to switch between 2 sets of 8 devices, 4 sets of 4 devices, 8 sets of 2 devices, or 16 sets of 1 device. If the SPLIT mode requires 2 or 4 devices then only valid choices will be offered.

For module MP2, press the key under CHANGE to switch between 1 set of 8 devices, 2 sets of 4 devices, 4 sets of 2 devices, or 8 sets of 1 device. If the SPLIT mode requires 2 or 4 devices then only valid choices will be offered.

Press EXIT to restore settings as on entry.

N.B. The number of sets can only be changed in systems with modules MP2 or MP4.

CLEAR all RAM

```
ALL RAM WILL BE ERASED
                            OK
```

Press OK to set all RAM to the currently selected erase value, EXIT to abort.

DATA DISPLAY

```
DISPLAY: AS DEVICE (8)
CHANGE                     OK
```

```
DISPLAY: 8
CHANGE                     OK
```

```
DISPLAY: 16
CHANGE                     OK
```

```
DISPLAY: 32
CHANGE                     OK
```

This setting selects how the data will be shown when a device or the RAM buffer is displayed or edited.
Press the key under CHANGE to modify the selection. Press EXIT to restore setting as on entry.
EMPTY FILL Value Setting

```
FILL: AS DEVICE (FF)
CHANGE                     OK
```

```
FILL: FF
CHANGE                     OK
```

```
FILL: 00
CHANGE                     OK
```

This setting determines the value put into the RAM buffer on start up or when the CLR or NEW function is used. It may be set to 00, FF, or the default for the device currently selected.
Press the key under CHANGE to modify the selection. Press EXIT to restore setting as on entry.

CHECKSUM calculation

```
SELECT CHECKSUM VALUES
BITS   DIGITS  MASTER   OK
```

This allows changes to made to three different settings. Press BITS to change the CHECKSUM width, press DIGITS to change the length of the calculated CHECKSUM or MASTER to input a MASTER CHECKSUM.

CHECKSUM WIDTH

```
CHECKSUM WIDTH  8 BITS
CHANGE                     OK
```

This setting determines the checksum should be based on 8 or 16 bit calculation. It cannot be changed for 8 bit devices.
Press the key under CHANGE to modify the selection.

CHECKSUM LENGTH

```
CHECKSUM LENGTH 4 DIGITS
CHANGE                     OK
```

This setting determines how the checksum will be displayed (see under the CKSUM single function key). Press the key under CHANGE to modify the selection. Press EXIT to restore setting as on entry.

MASTER CHECKSUM MONITORING

Enabling Master Checksum Monitoring, makes programming extremely secure. Once enabled, every time the PROG or VERIFY key is pressed, the checksum of the master RAM or ROM is checked against the master checksum and if different, an error message is displayed and the programmer will not allow programming or verifying to continue.

Checksum Monitoring is only permitted when a single master and no splits are set.

```
MASTER CKSUM:    OFF
OFF  AUTO  ENTER        OK
```

```
MASTER CKSUM:     1245
OFF     AUTO     ENTER   OK
```

```
MASTER CKSUM:   D6AB4E21
OFF     AUTO     ENTER   OK
```

Press OFF to disable master checksum monitoring. Pressing AUTO will calculate the checksum of the current master, either RAM or ROM, store it as the master checksum and enable monitoring. Press ENTER to manually enter the master checksum and enable monitoring. Press OK when finished.

PATTERN - PATT1 or PATT2

```
ALL RAM WILL BE FILLED
                    OK
```

Press OK to fill all RAM with a test pattern, EXIT to abort.

The first pattern is essentially an incrementing pattern with numbers from 0 to FF hex stored on each page (256 bytes) of RAM, so that for example location xxxxE3 contains E3 hex where xxxx is any number. However, the first two bytes of each page contain the page number stored least significant byte first (little Endian) and the middle bytes of each page contain FF hex.

There is a choice of two patterns, the second one is the complement of the first.

DEVICE

This function controls the device function settings:
ID CHECK
Silicon Signature Check
MOUNT CHECK
Check for devices in sockets
REV. CHECK
Check for devices upside down
BLANK CHECK
Blank check performed before programming, and actual blank value
VERIFY RANGE
Sets VCC high and low levels for verification
SPECIALS
AUTO   sets lock bits etc
MANUAL   manually blow lock bits
CHECKSUM
8 bit or 16 bit, set master checksum
Tacc
Set normal/slow access time
ADDR
Change the address range of the device

```
DEVICE SETTINGS
  ID   MOUNT  REVERSE BLANK
```

```
DEVICE SETTINGS
VERIFY SPECL CKSUM     Tacc
```

```
DEVICE SETTINGS
 ADDR
```

Press the key under the sub-function required.

ID-CHECK DISABLE

```
ID-CHECK   ENABLED
CHANGE                    OK
```

```
ID-CHECK  DISABLED
CHANGE                    OK
```

If enabled this function will check the silicon signature of every device operated on. If there is a mismatch, a warning will be given.

If the device is an older one without an ID, this message will be displayed:

```
NO ID IN DEVICE
                          OK
```

MOUNT CHECK DISABLE

```
MOUNT CHECK   ENABLED
CHANGE                    OK
```

```
MOUNT CHECK  DISABLED
CHANGE                    OK
```

This function does not stop a check for the presence of a device, but allows programming to be attempted even if a device is not detected as present.

REVERSE CHECK DISABLE

```
REVERSE CHECK   ENABLED
CHANGE                    OK
```

```
REVERSE CHECK  DISABLED
CHANGE                    OK
```

Reverse check checks for the orientation of a device before VCC and VPP are applied. Some device types may trigger this precaution even when correctly inserted, so this check may be disabled. The

presence of an incorrectly inserted device will still be detected, but not before an attempt has been made to program it.

BLANK CHECK SETTINGS

```
BLANK CHECK   ENABLED
CHANGE VALUE              OK
```

```
BLANK CHECK  DISABLED
CHANGE VALUE              OK
```

This function checks that all devices to be programmed are blank before programming. Disable if not required, e.g. for EPROM emulators. The blank value will be assumed the normal value for the current device. To change this press VALUE.

```
BLANK: AS DEVICE (FF)
CHANGE                    OK
```

Press CHANGE to set value to 00 or FF. The device manufacturer is shown as USER! if this setting is changed. Note that this change only lasts until the type is next set.

VERIFY RANGE

```
VERIFY Vcc 4.50 TO 5.50
PERCENT   DEFAULT         OK
```

Verification of programming may be performed at any two VCC levels between 2.75V and 6.75V or, by pressing PERCENT, at VCC +/- 0%, 5%, 10% and 20%. Press EXIT to restore setting as on entry.

SECURITY / SPECIAL FUNCTIONS DISPLAY

```
SPECIAL FUNCTIONS
 AUTO   MANUAL
```

Some special functions apply to certain devices, such as the security locking of the programmed memory. The AUTO setting allows a predetermined selection of security operations to be applied automatically to certain device types after programming and verification. This would save much time in production, by cutting out a whole set of operations on a device.

```
AUTO SPECIALS        S12--
LOCK1  LOCK2  LOCK3  ENCRY
```

Special functions that will be automatically performed after programming are shown by a number against the S. a dash - indicates that the device has this function but that it is currently disabled. In the example shown above, the device has four special functions but only lock 1 and lock 2 will be programmed when the PROG key is pressed.

When the programmer is used manually in development the same operations may be required, after testing the device. The MANUAL selection allows the special function to be applied manually to the device. These operations vary depending on the device but a typical example involving locking microprocessor is shown below.

```
MANUAL SPECIALS
LOCK1  LOCK2  LOCK3  ENCRY
```

Press the key under the function you want to perform now.

```
SPECIAL FUNC. PERFORMED
                      OK
```

```
A1: SPECIAL FUNC. FAILED
                      OK
```

The module LEDs flash in the same way as they would for programming, see under the PROG single function key.

Devices that have special functions are described in more detail in appendix B.

CHECKSUM CALCULATION

This procedure is exactly the same as for CHECKSUM under the RAM key.

TACC SETTING

```
ACCESS TIME - NORMAL
CHANGE                OK
```

When programming certain devices, it is necessary to slow down the access time in order to ensure correct programming. Press the key under CHANGE to enable this when told to do so by your supplier. It is often necessary to enable this function when a large and complicated package converter is being used.

ADDR SETTING

```
ADDRESS RANGE
00000  -  00FFFF      OK
```

This allows different start and end addresses to be set for operations on the currently selected device. These are used to program microprocessors which almost program like EPROMs, but do not

---

have the complete address range fitted.

Use the f1 key to select the low address and the f3 key for the high address. Pressing the ← key decrements the address by 2K and pressing → key increments the address by 2K. Note, you cannot directly type these addresses using the hex keypad.

This change lasts until the type is next set. Until then the manufacturer is shown as USER!

COMMS

```
               ---PRINTER---
HOST   PORT    USED?  PARAM
```

Press the key under the sub-function required.

HOST

```
HOST IS A PC
CHANGE                    OK
```

```
HOST IS A NON-PC
CHANGE                    OK
```

HOST - PC

If the PC setting is chosen it is assumed that the PROMDRIVER software is running on the PC. Uploads and downloads will be controlled from the PC, allowing advantage to be taken of the high speed transfer capabilities in PROMDRIVER. This is much faster than transferring, for example, Intel Hex files using a serial port. The parallel port selection will make use of the special hardware of the PC to provide (in conjunction with

PROMDRIVER) a high speed two way communications channel.

When the I/O single function key is pressed, the programmer goes automatically into remote control mode and it assumes that the HOST is a PC is being controlled by PROMDRIVER software.

HOST - NON-PC

This setting should be chosen to allow simple file transfers to and from the programmer. The serial port allows transfers in both directions while the parallel port is configured as a standard Centronics one-way printer interface (data flow towards the programmer).

When the I/O single function key is pressed the options of data input or output will be offered.

It should be apparent from the above that the PC HOST selection is the preferred method for high performance.

PORT

```
SERIAL:38400b/s 7D+2S
TYPE   SPEED   MODE    OK
```

This screen controls the selection of the non-PC protocol parameters. Pressing TYPE selects between SERIAL and PARALLEL. SPEED allows you to select the Baud rate if serial comms have been chosen. Similarly MODE allows the selection of the correct number of data bits and start/stop bits.

PRINTER USED?

```
PARALLEL DRIVES PRINTER
CHANGE                    OK
```

ELECTRONICS

```
PARALLEL USED FOR COMMS
CHANGE                    OK
```

This function allows switching the use of the parallel port between communications with the host PC or with a printer for printing the device list.

PRINTER Parameters

```
    PRINTER FUNCTIONS
INIT   TERM   PAGEL      OK
```

These functions allow the setting of the printer initialisation and termination strings as well as the printer page length.

Note: these functions are not yet implemented and cannot be changed.

MASTER

MASTER is ROM
MASTER is RAM buffer

```
MASTER: FROM ROM
 RAM    ROM               OK
```

This function determines whether devices are programmed and verified from the RAM buffer or from a device in the ROM socket. An error message will be given if ROM master is selected with no copy socket in the programming module.

```
No copy socket on this
module                    OK
```

It is permitted to select RAM master when the no RAM is fitted. An error warning will be given if an attempt is made to use the RAM when there is insufficient or none, see Socket Checks.

INFO

This function gives information about the hardware, firmware and installed options. The various items of information are accessed by pressing the MORE key.

INFO Module firmware revision

Memory size

Programming module fitted

Mother board hardware revision

Module hardware revision

Unit serial number

Main board serial numbers

Module serial numbers

Socket usage

Module service date

Software support date

```
PROMWARE VERSION V03.75
                        MORE
```

```
BUFFER SIZE: 2048KByte
                        MORE
```

```
BUFFER SIZE: None fitted
                        MORE
```

```
MODULE FITTED: MP4
                        MORE
```

```
MAIN BOARD REVISION
0002                    MORE
```

```
MODULE REVISION
0001                    MORE
```

```
UNIT SERIAL NUMBER
00000001                MORE
```

```
MAIN BOARD SERIAL NOs.
0001 0001 0001          MORE
```

ELECTRONICS

```
┌─────────────────────────────────┐
│ MODULE SERIAL NOs.              │
│ 0001 0001 0001          MORE   │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ SOCKET CYCLES:                  │
│ 879                     MORE   │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ MODULE SERVICED:                │
│ 08/05/01                MORE   │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ SUPPORT TILL:                   │
│ 28Mar2002               MORE   │
└─────────────────────────────────┘
```

The programmer will stay in INFO mode until the EXIT key is pressed.

SOUND

This function controls the audible sounds emitted by the programmer. These comprise key-clicks and operation complete (success and failure) tones.

```
┌─────────────────────────────────┐
│ AUDIBLE FEEDBACK                │
│ CLICKS TONES            OK     │
└─────────────────────────────────┘
```

Press CLICKS to change key clicks

```
┌─────────────────────────────────┐
│ KEY CLICKS ARE ON               │
│ CHANGE                  OK     │
└─────────────────────────────────┘
```

or TONES to change operation complete tones.

```
┌─────────────────────────────────┐
│ OPERATION TONES ARE OFF         │
│ CHANGE                  OK     │
└─────────────────────────────────┘
```

EEPROM TEST

This function allows the testing of EEPROMs and certain other reprogrammable devices using a special test pattern which thoroughly tests both the data storage and the address line integrity. The devices will be left in the unprogrammed state after testing. The programmer will stay in the EEPROM TEST mode until the EXIT key is pressed.

```
┌─────────────────────────────────┐
│ EEPROM TEST                     │
│                         START  │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ TESTING...                      │
│ ███▉□□□□□□□□□□                  │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ Not applicable to this          │
│ Device                  OK     │
└─────────────────────────────────┘
```

MISC.

All the miscellaneous functions can be enabled under this key. They are displayed one after another - pressing the OK key moves on to the next function. The programmer will stay in MISC. mode until the EXIT key is pressed.
MISC.

> Lock
> Enable empty socket detect
> Start up tests
> Clear all RAM on start up

LOCK

```
┌─────────────────────────────────┐
│ KEYBOARD IS UNLOCKED            │
│ CHANGE                  OK     │
└─────────────────────────────────┘
```

```
┌─────────────────────────────────┐
│ KEYBOARD IS LOCKED!             │
│ CHANGE                  OK     │
└─────────────────────────────────┘
```

This function allows the settings to be locked. After locking, no setting may be changed, and no editing command is permitted to modify data, until the settings are unlocked again.

An audible tone will sound and a message will be displayed to indicate an illegal attempt to change a setting.

Enable Empty Socket Detect

```
EMPTY SKT. DETECT ON
CHANGE                    OK
```

```
EMPTY SKT. DETECT OFF
CHANGE                    OK
```

If this setting is set to on and reverse check is on, the report of the last operation will be cleared from the display when the last device is taken out of its socket. The default setting is off. This feature provides a useful check for the operator to determine whether or not the devices currently in the sockets have been programmed.

Start Up Tests

```
START UP TESTS: ON
CHANGE                    OK
```

```
START UP TESTS: OFF
CHANGE                    OK
```

Clearing RAM on start up

By default, on start up only a maximum of 8Mbits of RAM is cleared. However if a larger RAM module is fitted it must all be cleared use this function.

```
START:CLR ALL RAM: OFF
CHANGE                    OK
```

```
START:CLR ALL RAM: ON
CHANGE                    OK
```

ALG.

This function displays information about the currently selected device and its programming algorithm. The various items of information are accessed by pressing the MORE key.

```
ALGORITHM     CODE
25/1000/3000N C013   MORE
```

The three items separated by the / symbol are the maximum number of pulses, the length of each pulse in microseconds and the length of any overprogram pulse in microseconds. If an N is displayed with the third term it indicates that the over-programming is multiplicative, otherwise it is additive.

The second term is the Device code for the current device.

```
Vpp   Vcp   SIZE
12.75 6.25 020000    MORE
```

The programmer will stay in ALG. mode until the EXIT key is pressed.

PRINT

Press this key to get a complete device list for the currently installed programming module.

```
Printing...
```

```
No printer installed
                          OK
```

DEUTSCH | ENGLISH

This function selects the language used on the programmer display.

Please enquire whether the language of your choice is available.

ELECTRONICS

STAT

This function enables programming statistics to be gathered and reset.

```
STATISTICS (COUNT 12000)
OFF   RESET              OK
```

```
STATISTICS (OFF)
OFF   RESET              OK
```

Press ON to enable statistics collection, OFF to suspend it or RESET to zero the count of programmed devices and enable collection.

When statistics are enabled, the top level display permits the display of accumulated statistics:

```
INTEL 27C202C
REPORT   STAT   ALL
```

By pressing STAT, the accumulated statistics are displayed, showing the total number of PROMs programmed, how many of them were successful, how many failed and the failure rate.

```
COUNT  GOOD    BAD   (1.0%)
100    99        1     OK
```

14. Remote Operation With Promdriver

The programmer can be driven remotely using the supplied PROMDRIVER software package. The I/O key must be pressed and the screen will show REMOTE
Either the serial or parallel port of your PC can be connected to the programmer with the lead provided. If using a serial port with a 9 pin connector you will need to use the extension 'tail' that is provided. (Note: Windows Promdriver only supports parallel communications.)
The communications ports on the programmer are located on the back and are labelled.
The same cable is used for either parallel or serial communications. It is turned round to use it in the opposite mode. Parallel communications will always be the faster method.

Promdriver Installation (DOS)

Firstly, install the files on the floppy disc provided on to your hard disc drive. Put the disc into your floppy drive and log on to that drive. Type:

INSTALL

The program will then ask you to enter a directory name on your hard disc to which it can copy the required files. It will not write anything to your disc without asking your permission first. We do not use any form of copy protection, create any hidden files or modify your disc in any way.

Please feel free to circulate this disc to your colleagues and friends as a demonstration of our software. If you have any comments about our products or software we would like you hear from you.

Menu Mode

Type PD2 on your computer to execute PROMDRIVER software. The first time of operation our Initialisation program will be run automatically for you to select the method of communications and printer parameters (see information during INSTALL).
If you have added a PATH=\promdriver_directory to your AUTOEXEC.BAT FILE (see our INSTALL instructions and your MS-DOS manual) you will be able to execute PD from any directory on your disc.
To change the communications port or printer parameters at any time, type:

PD2 /I

and the initialisation program will be run again.
Normal mode of operation uses colour and pop up windows for all functions. Context sensitive HELP is available by pressing F1 and all the options available are indicated on the screen.

Promdriver Installation (Windows)

Follow the instructions supplied with the CD.

Promdriver Operation

This manual does not cover PROMDRIVER operation. Context sensitive help screens will provide all the information you need to make use of PROMDRIVER extremely easy.

Batch (Script) Mode

Introduction

PROMDRIVER has been enhanced to provide a comprehensive batch mode (DOS) or script mode (Windows) to streamline the use of the programmer in a production department.

This allows the user to write a script to automate programming jobs. For example, the file can be downloaded and the correct device type selected automatically. The operator can then be prompted to insert devices as required. A running total of Programmed devices, Failures and Quantity still required may be displayed. Other features include user defined messages, user defined menus, and remote setting of programmer functions.

This chapter defines the Batch Programming Language. Certain features have not yet been implemented and have been marked with a * symbol.

Screen elements of Batch mode presentation (DOS)

Heading

By default the programmer manufacturer's logo will appear at the top of the screen, occupying 6 lines (4 available for text). By using the HEADING command the user can customise or remove this heading.

Operator Instruction Area

This is a mandatory area of the screen capable of 3 lines of text display (this may be extended if necessary). It will be coloured in a manner to invite attention.

Menus or messages

The area occupied by the operator instruction window is also used for messages or menus. The message or menu will be another window opened over the top of the instruction area. It will have a minimum size of 3 lines of text, and a maximum length of 12 lines (i.e. down to the bottom of the screen).

'Failures', 'Programmed' and 'Required' boxes.

These boxes each have their own standard location at the bottom of the screen. On colour screens they each have an appropriate colour.

Failure - RED

Programmed - GREEN

Required - BLUE

Promdriver Enhanced Batch Mode Specification

Like most programming languages the PROMDRIVER Programming Language uses English words only, even on other language versions of the software. The user interface will still be in the required language, but the script language itself always uses English keywords.

All scripts will be held in non-document (plain ASCII) files generated by the user using his or her favourite word processor; e.g. SCRIPT.SCR. There is no special restriction of the filename (no extension is required).

In this case the batch file mode would be invoked by typing:

PD2 @SCRIPT.SCR

This tells PROMDRIVER to open the file SCRIPT.SCR and execute the instructions in it.

Syntax (DOS and Windows)

Commands normally appear in the file one per line. Some special commands spread over several lines. Comments are permitted; introduced by a semi-colon, and finishing at the end of the line.

Any keyword may include the underscore character (_) at any point in the word. This is ignored by the parser but may improve readability.

e.g. BYTESELECT and BYTE_SELECT will be interpreted the same.

The help key (F1) from the main screen in menu mode will give further information about all valid batch instructions.

Control Flow – Labels

A label is always preceded by a colon in the first column, and is the only item on the line (apart from comments, if any). References to labels are not preceded by a colon. Labels are of any length with 16 significant characters. Upper and lower case are equivalent. Labels may contain alphanumerics and the underscore '_'.

Example:

START_AGAIN_12

GOTO START_AGAIN_12

(actually an example of an infinite loop)

Control Flow Commands

GOTO <label>

This instruction is self-explanatory but some restrictions are placed on its use to avoid the usual pit-falls of programming with GOTO's.

Conditionals

There are 2 structures for 'if' type conditions, one for statements including GOTO's, and one for other statements.

IF <condition> GOTO <label>

The GOTO on a condition must be on the same line as the IF as shown. No ENDIF or ELSE are required or allowed.

IF <condition>
<commands>
[ELSE
<commands>]
ENDIF

GOTO is not allowed within a conditional section.
Typical conditions are:
IF MENU_RESULT = <n> *
IF YES
IF NO
IF ESC
IF OTHER_REPLY
IF MORE_NEEDED
IF ERRORLEVEL = <n>
where <n> is a decimal integer.

Error handling

Each function having the possibility of an error (e.g. Programming, Append, Read etc) will set a (16 bit) variable called ERRORLEVEL. 0 will signify success, and other values will have fixed predetermined meanings (e.g. DEVICE NOT PRESENT, DEVICE NOT BLANK, USER ABORTED ETC). ERRORLEVEL can be tested after each operation using an IF construction (see above).

If a syntactical error is detected in a batch command, it is reported on the screen (with the erroneous line number) and processing is immediately aborted.

Command Description

TYPE <device_code>
TYPE AUTO
TYPE ?

TYPE AUTO is the same as AUTOKEY (see below).

ERRORLEVEL codes set

0 = type selected ok
1 = user aborted selection
2 = type not recognised

TYPENAME "<text_string>"

Once a device type has been selected, (see above), its name can be changed to whatever the user wants.
This can be useful when programming micro-processors that program like EPROMs. First the TYPE is set using the EPROM's Device code, then using TYPENAME, the text string in the status area can be corrected to display the micro-processor's actual name.
For other special commands for micro-processors that program like EPROMs see appendix F.

FORMAT <file_format>
where <file_format> is:

BINARY
INTHEX
MOTHEX
INTOBJ80
INTOBJ51
INTOBJ86
INTOBJ286
FREEHEX
FREEDEC
TEXTAG

ERRORLEVEL codes set
0 = set ok

FILEADDRESS <fileaddress>
where address is a hex number between 0 and FFFFFFh

ERRORLEVEL codes set
0 = set ok

POLARITY <polarity>
where <polarity> is TRUE or INVERTED
ERRORLEVEL codes set

0 = set ok

BYTESELECT <byteselect>

where <byte_select> is:
NORMAL
REVERSED
ODD
EVEN

ERRORLEVEL codes set

0 = set ok

NOEXIT
NOEXIT ON *
NOEXIT OFF *
This command is available to prevent accidental exit from batch mode Some actions cause an immediate exit from batch programming e.g. user presses ESC. This may not be desirable. In the case of an infinitely looping program. ALT/F10 will allow exit.

EXIT
Terminates batch file execution.

WAIT n
Delays for n seconds.

SPECIAL MANUAL <command>
[<hex_value> [<hex_value>]]
SPECIAL AUTO <command> ON
[<hex_value> [<hex_value>]
SPECIAL AUTO <command> OFF

Special functions such as 'Lock PROM' can be found for some devices in the DEVICE SPECIALS selection in menu mode. Most of these functions are also available in batch mode.
There are two ways of activating each special function, similar to menu mode and stand-alone mode. MANUAL activates the special function at the time it is encountered in the script file. An AUTO command activates the special function every time programming occurs.

The appropriate <command> is the item number to the left of the menu and it will depend on the device concerned. For example, with an 8751 the only function available is 1, which blows the security bit.

To find out which functions are available look at the menu mode SPECIALS with the appropriate device selected. The help screen (F1) will give further information.

Available parameters are indicated in square brackets [ ]. These are either a hex value, e.g. in the case of a PIC micro-controller's ID, or the item number in a subsequent menu, e.g. for PIC oscillator setting of XT, the <hex_value> should be set to 3.

ERRORLEVEL codes set

0 = Set ok

1 = Function programming failure

MONITOR_CKSUM <setting>
where <setting> is:
  OFF
  ON
  AUTO
  MANUAL <checksum>

This allows the setting and disabling of the master checksum. If set, this will be used before programming and verification operations to ensure that the master (ROM or RAM) is still valid. If invalid the operation will be aborted.

AGAIN?
AGAIN? GOTO <label>
Displays message asking if a repeat is required. If reply is YES or ENTER, starts executing batch mode instructions from the start, or from the label.

MENU
 "<menuline_1>" [GOTO <label> | TYPE <device_code> | NOOPTION]
 "<menuline_2>" [GOTO <label> | TYPE <device_code> | NOOPTION]
 …
 "<menuline_n>" [GOTO <label> | TYPE <device_code> | NOOPTION]
MENUEND

Displays a menu with movable highlight bar. On selection of a line with an associated GOTO, goes to that line. On selection of any other line goes to next instruction in file after MENUEND.

If line text is followed by TYPE <device_code> then this is processed as a normal TYPE setting command.

If a line is followed by NOOPTION it is only displayed but not offered as an option, so the cursor cannot be moved to such lines. This is a useful command for headers and other explanatory lines.

GOTO and TYPE may both appear on the same line but NOOPTION may not be accompanied by either. Window size depends on number of lines and length of lines in menu.

DELETE <filename>
Deletes the file specified without asking the user. No action if file does not exist.

RENAME   <old_filename> <new_filename>

Renames the file specified without asking the user. No action if file does not exist.

%REQUIRED
%PROGRAMMED
%FAILURES
%TYPENAME
%FORMAT
%FILEADDRESS
%POLARITY
%BYTESELECT
%SERNO
%FILECKSM *
String variables which substitute the value of the variable into a text string, i.e. inside a pair of quotes ("), of a heading, message or menu.

MASTERSPLIT <setting>
where <setting> is:

```
        ROM
        1_NONE
        2_NONE
        2_SINGLE
        4_NONE
        4_SINGLE
        4_DOUBLE
        8_NONE
```

```
        8_SINGLE
        8_DOUBLE
```

The number is the number of masters and NONE, SINGLE and DOUBLE indicate the split setting. The default is 1_NONE if the programmer has RAM fitted otherwise it is ROM.

BLANKCHECK ON
BLANKCHECK OFF
Defaults to ON

IDCHECK ON
IDCHECK OFF
Defaults to ON

MOUNTCHECK ON
MOUNTCHECK OFF
Defaults to ON

REVCHECK ON
REVCHECK OFF
Defaults to ON

STATUS ON
STATUS OFF
Enables, or disables, the display of current settings like file name, file type, byte select, polarity, file address, device type, adapter required, etc. Default is OFF.

HEADING
"<heading_line_1>"
["<heading_line_2>"
 ["<heading_line_3>"
 ["<heading_line_4>"]]]
HEADINGEND
If HEADING is not specified the batch window will show the default heading information block. Using

---

HEADING replaces this with user defined information, which may be null information, if there are no lines between HEADING and HEADINGEND. Any message lines after line 4 will be ignored. HEADING must be the first actual command in the response file, if the default heading is to be completely disabled. The heading may be changed, at any time in the script.

```
MESSAGE [GET_YES_NO]
"<message_line_1>"
["<<message_line_2>"
["<message_line_3>"
["<message_line_4>"] ] ]
MESSAGE_END
```

Messages will appear in the message area. This area will be enlarged (by opening a window over the top) if the message height requires it. The maximum size is 12 lines, after which message lines will be truncated. The message is displayed until any key is pressed.

If GET_YES_NO is included on the MESSAGE line, the relevant parameters (YES, NO, ESC or OTHER_REPLY) are set when the key is pressed and these parameters can be later tested by the IFconstruction.

RUN <program_name> ["<argument>" [..."<argument>"]]
RUN_STDIO <program_name> ["<argument>" [..."<argument>"]]

Runs a user program with optional parameters. RUN will assume that the user program either does no screen I/O, or that it deals with restoring the screen itself. RUN_STDIO will open up a window (full screen) for user I/O, and then clear it away afterwards.
The user program may return a code which will be placed into ERRORLEVEL.

FAILURES 0
FAILURES OFF
FAILURES ON
This causes a failures box to be displayed (or removed) or resets the number of failures to the number shown if the box is already displayed. The number of failures is automatically incremented by any 'Failure to program' messages in Program, Append, Serial Number Append, or Special Program operations, but not those caused by 'device not present'.

PROGRAMMED 0
PROGRAMMED +
PROGRAMMED OFF
PROGRAMMED ON
This causes a programming success quantity box to be displayed (or removed) or resets the number of successes to the number shown if the box is already displayed.
The number of successes is incremented by a PROGRAMMED command, e.g.

PROGKEY
IF ERRORLEVEL = 0
        PROGRAMMED +
        REQUIRED –
ENDIF

REQUIRED 20
REQUIRED -
REQUIRED ?
REQUIRED OFF
REQUIRED ON
This causes a devices required quantity box to be displayed (or removed) or resets the number required to the number shown if the box is already displayed.
The number required is decremented by a REQUIRED - command. Note: the REQUIRED number can go negative if more devices are programmed than were requested.

VERIFYLEVEL <setting>
where <setting> is:
+/-0%
+/-5%
+/-10%
+/-20%
Verification of programming may be performed at VCC +/- 0%, 5%, 10% and 20%.

VERIFY_LOW <setting>
VERIFY_HIGH <setting>
where <setting> is a valid VCC value.

EMPTYFILL <setting>
where <setting> is:
> 0
> FF
> DEFAULT

This defines the hexadecimal value that will be used to fill parts of the buffer that are not defined in the file to be downloaded, displayed or printed. It will normally be set to the blank value for the currently selected device (DEFAULT), but you may want to force it to another value so that, for instance, the checksum is consistent with another manufacturer's programmer.

DOWNLOAD <buf_start_addr> <buf_end addr> <file_name><file_start_addr>
Downloads defined part of file to defined part of buffer. If <file_name> is ? the operator will be prompted to enter the filename.
ERRORLEVEL codes set
0 = type selected ok
1 = no type set

UPLOAD <file_name>
Uploads buffer to file. If <file_name> is ? the operator will be prompted to enter the filename.
ERRORLEVEL codes set
0 = type selected ok
1 = no type set

ACCESSTIME <setting>
where <setting> is:
NORMAL

SLOW

By setting the access time to SLOW, this function allows you to increase the access time for reading, verifying and checksumming the selected device. This is necessary for certain devices especially micro-processors that program like EPROMs but in different packages. If set to SLOW, the setting will be displayed in the status area.
The default setting is NORMAL.

ADDR_RANGE <low_address> <high_address>
This function permits the programming of a microprocessor that programs like an EPROM but is smaller than the EPROM.
The programmer will adjust the input addresses to the nearest 2K byte boundary and the setting will be displayed in the status area.

SERNO FILE <file_name>
SERNO AUTO
SERNO OFF
SERNO CHANGE
These functions allow the serial numbering of programmable devices to be carried out automatically each time that they are programmed. SERNO FILE <file_name> must be processed in the script file before any of the other SERNO instructions can appear. <file_name> refers to the serial number filename (suffix .SNO) that has been set up in menu mode by typing N from the main menu.
SERNO AUTO and SERNO OFF are equivalent to auto and manual in menu mode. SERNO CHANGE allows changes to the next serial number to be programmed from

within batch mode.

CLEAR_RAM
Clears all RAM to the current erase value

NEW
Puts all settings as they were when batch mode first started.

PROGKEY
Same as pressing the PROG key on the standalone programmer.

READKEY
Same as pressing the READ key on the standalone programmer.

ERASEKEY
Same as pressing the ERASE key on the standalone programmer.

VERIFYKEY
Same as pressing the VERIFY key on the standalone programmer.

CKSUMKEY
CKSUMKEY <nnnn>
CKSUMKEY <nnnnnnnn>
Same as pressing the CKSUM key on the standalone programmer.
If a 4 or 8 digit hexadecimal number is included in the instruction, an error message will be displayed against any socket where the device check-sum does not equal this number.

BLANKKEY

Same as pressing the BLANK key on the standalone programmer.

AUTOKEY
Auto type select.

Example Script

This is a printout of an example of a user generated script file called:
SAMPLE1.BAT
Typically it would be invoked using an MS-DOS batch file such as the following (ACME.BAT):
rem run pd2 in batch mode with script to program 27010's
PD2 @SAMPLE1.SCR
As may be seen the effect of the user typing ACME is to load PD2 and cause it to run the script file SAMPLE1.SCR.
The actual script follows, and contains several comments to explain the use of commands. The total effect of the script is to produce a number of EPROMs for a particular job, allowing for a choice of EPROM type, either from a predetermined list or by auto-selection. The correct file will be automatically downloaded, and a count is maintained on screen of programmed devices and failures. The user is also asked to say how many he is required to blow, and the number still required is displayed. We suggest that you use a copy of this file to modify for your own purposes until you become familiar with the programming language.

Sample script 'SAMPLE.SCR'
; sample batch program to blow some 27C010s
; programming commands are shown in UPPER CASE for clarity although they
; may equally well be in lower case.

    ; if first command is HEADING then the default logo will not appear

```
HEADING
"      ACME MANUFACTURING CORP."
" "

"      Program to blow some 27C010s for the Mega-Cyclotron job"
HEADINGEND
```

    ; allow operator to pick one of the available parts
    ; lines with no_option cannot be selected

```
MENU
" As Acme Co. stores may provide different EPROM types please select the " NO_OPTION
"                type available from the following menu" NO_OPTION
" " NO_OPTION            ; example of a line which can't be chosen
"      AMD 27C010" TYPE 99A9         ; example of a menu type setting
"      Intel  27C010" TYPE EDE8
"      ST   27C010" TYPE 8889
"      NS   27C010" TYPE C013
"      OTHER" GOTO AUTOSELECT             ; example of a menu goto
"           (use Cursor Up and Down, then press enter)" NO_OPTION
MENU_END
```

    ; wait command causes delay for a number of seconds
```
WAIT 1
GOTO DOJOB          ; if we're here we must have selected a type

:AUTOSELECT          ; label to goto if type wasn't in list
                     ; tell operator to put in apart to identify
MESSAGE
" Place one of the devices in any socket of the programmer"
" in order to automatically determine its type "
"       ....when ready press any key"
MESSAGE_END
```
    ; now do an auto select operation
```
TYPE AUTO
```

```
                    ; demonstrate that we know the type now
MESSAGE
" The type of device you wish to program is %type_name "
"        ….press any key "
MESSAGE_END
                    ; in all cases we get here to blow the devices
:DOJOB
                    ; first we download the file required
FORMAT BINARY
DOWNLOAD 0 1F000 SAMPLE1.SCR
                    ; turn on the box showing how many successfully programmed
                    ; devices we get
PROGRAMMED 0
                    ; and a box showing failures
FAILURES 0
                    ; ask the operator how many he needs to do
                    ; and open up a box with this number
REQUIRED ?
                    ; this label is where we'll loop back to,
                    ; to program more parts
:LABEL
                    ; tell operator to put some blank parts in
MESSAGE
""""

"        Put some blank %type_name parts in sockets and press any key"
""""

"                = (Press ESC to exit when done) ="
MESSAGE_END
                    ; do some programming
PROG_KEY
                    ; add successes to programmed message box
PROGRAMMED +
                    ; decrement number required by appropriate quantity
REQUIRED -
                    ; …and loop
GOTO LABEL
        ; in this file operator decides when he's finished and exits by
        ; pressing ESC
```

Alphabetical Command List

ACCESSTIME	Slow down the read access time
ADDRRANGE	Change the address range of a device
AGAIN?	Repeat script from start if user answers "yes"
AGAIN? GOTO	Repeat script from label
AUTOKEY	Do remote auto type set
BLANKCHECK	Turn automatic blank check on or off
BLANKKEY	Do a remote blank check
BYTESELECT	Set which bytes are selected during download
CKSUMKEY	Do a remote checksum operation
CLEAR_RAM	Clear out all of RAM
DELETE	Delete an MS-DOS file
DOWNLOAD	Load a file into the programmer RAM
EMPTYFILL	Over-rides blank value of device if required
ERASEKEY	Do a remote erase operation
EXIT	Exit script
FAILURES	Turn on/off 'Failures' box
FILEADDRESS	Set default start address in files
FORMAT	Set file format
GOTO	Start executing from a label
HEADING,HEADINGEND	Define a screen heading
IDCHECK	Turn automatic id check on or off
IF,ELSE,ENDIF	Condition construction
MANUAL	Allow remote operation of certain keys
MASTERSPLIT	Define master split setting
MENU,MENUEND	Display a menu and get response
MESSAGE, MESSAGE_END	
	Display a message (optionally get a response)
MONITORCKSUM	Validates the checksum before programming
MOUNTCHECK	Turn automatic mount check on or off
NEW	Resets all parameters
NOEXIT	Prevent ESC key from exiting script
POLARITY	Set data polarity for file downloads
PROGKEY	Do a remote program operation
PROGRAMMED	Turn on/off 'Programmed' box
READKEY	Do a remote READ operation
RENAME	Renames the specified file
REQUIRED	Turn on/off 'Required' box
REVCHECK	Turn automatic reverse check on/off
RUN	Run a user program under MS-DOS

RUN_STDIO	Run a user program requiring use of screen and keyboard under MS-DOS
SER_NO	Serial numbering
SPECIAL	Manually or automatically program specials
STATUS	Turn on/off on-screen status information
TYPE	Set device type
TYPENAME	Change device type name
UPLOAD	Read programmer buffer into a file
VERIFYKEY	Do a remote verify check
VERIFYLEVEL	)
VERIFY_LOW	) Change the verify voltage range
VERIFY_HIGH	)
WAIT	Do nothing for a defined number of seconds

String variables

%BYTESELECT	expands to ODD, EVEN, ALL etc.
%FAILURES	expands to number of failures
%FILEADDRESS	expands to default file address
%FORMAT	expands to description of file format
%POLARITY	expands to TRUE or INVERTED
%PROGRAMMED	expands to number of successes
%REQUIRED	expands to number required
%SER_NO	expands to next serial number
%TYPENAME	expands to description of type set

Truth variables

ERRORLEVEL=<n>	ESC true if ESC key was pressed
MORE_NEEDED	true if required variable > 0
NO	true if reply was 'no'
OTHER_REPLY	true if reply was not 'yes', 'no', or ESC
YES	true if reply was 'yes'

ELECTRONICS

Appendix A

A full device list can be provided by running the MQP Electronics program DEVSUP (Device Support) and pressing the Output S2000 list.

ELECTRONICS

Appendix B - Special Functions

Special Functions on MP3

The programming module MP3 programs 16 bit devices (word wide) and other devices that are bigger than 8 bits wide. A function has been added to local mode to permit the programming of alternate words into the EPROMs when required.

The soft key f3 is used for this function. When in the quiescent state, the display will show the currently selected type and which words will be programmed:

```
INTEL 27C210

REPORT        ALL
```

The word ALL above f3 indicates that all words will be programmed, the normal mode. Pressing f3 will cause the programmer to cycle round the valid settings of ODD, EVEN and ALL. By using this key, you can select ODD a blow a set of PROMs, then select EVEN and blow a set. Therefore if you have a file on a PC for a target system with a 32 bit wide bus, downloading it using PROMDRIVER and programming two sets of PROMs by the above method in local mode results in 32 bit wide programming.

PIC16C5X

When programming PICs on the MP3 module, in conjunction with PIC pin converters, the special functions required by PICs are enabled.

The programmer display shows how many functions are available while it is displaying the type selected. It does this in the top right of the display with a typical format:

```
MCHIP 16C54 *          S----
REPORT                 *NOTE
```

The 'S' stands for 'specials' and the following - haracters indicate four functions which are currently switched off. The four functions for a Microchip 16C54 are:

1.   Watchdog Disable

2.   Code Protect

3.   Oscillator Type

4.   User ID

For example if watchdog disable and code protect were currently turned on for automatic activation after normal programming the display would show:

```
MCHIP 16C54 *          S12--
REPORT                 *NOTE
```

Setting an automatic special function

Press the FUNC key, then the DEVICE key, then the cursor right key, then SPECL and the display will show:

```
SPECIAL FUNCTIONS
AUTO MANUAL
```

Pressing AUTO will result in a display for setting up automatic programming functions:

```
AUTO SPECIALS          S----
WDOFF   LOCK   OSC      ID
```

Use the 4 function keys to set up the desired automatic functions and

---

then press EXIT until you reach the top level display again.

When pressing OSC a further menu is offered:

```
OSC TYPE: XT
CHANGE                    OK
```

Pressing CHANGE allows the oscillator type to be altered or programming it to be disabled.

When pressing ID a further menu is offered:

```
ID: 1234 (OFF)
  ON                      OK
```

Use the keys 0-F and the cursor keys to enter the user ID required.

Please note that the settings can be copied by reading a 'master'
pic device. Only the code protect function will be unaffected by this,
as a protected PIC cannot of course be used as a master device.
The code protect setting should be set as required, as described above.

Performing a manual special function
Press the FUNC key, then the DEVICE key, then the cursor right key, then SPECL and the display will show:

```
SPECIAL FUNCTIONS
  AUTO   MANUAL
```

Pressing MANUAL will result in a display for performing manual programming functions:

```
MANUAL SPECIALS
  WDOFF  LOCK    OSC    ID
```

Pressing one of the 4 function keys marked will immediately perform the labelled function on all the socketed devices (in the case of OSC and ID you will also be asked to specify a

value). Please refer to the manufacturers data sheet for the device concerned for any restrictions on the order in which special function may be used.

PIC16C6x/7x

```
MCHIP 16C61 *          S12-
REPORT               *NOTE
```

This is similar to the PIC16C5x except that the special functions have been modified. ID setting and Lock are the same but the other functions have all been arranged under one key called CONFIG for both MANUAL and AUTO specials.

Setting an automatic special function
Pressing AUTO will result in a display for performing programming functions:

```
AUTO SPECIALS          S12-
CONFIG    ID    LOCK
```

On selecting CONFIG, a further menu is offered:

```
CONFIG IS OFF
CHANGE         OPTIONS  OK
```

showing the current setting of the AUTO CONFIG option and offering the chance to change the current CONFIG state.

```
       CONFIG OPTIONS
  WDTE   PWRTE  BODEN   FOSC
```

The meaning of these options is described in the Microchip data books.

Performing a manual special function
Pressing MANUAL will result in a warning:

```
Warning: RAM must have
```

```
correct data          OK
```

This reminds you that the correct locations in memory must be loaded before operating this function. These are described in full in the Microchip data books.

On pressing OK, the following display for performing manual programming functions will appear:

```
    MANUAL SPECIALS
CONFIG  ID  LOCK
```

Press the relevant F key for the function that you want to program into all devices currently in the sockets.

AMD 29F400T & 29F400B

These devices have 2 special functions, unlock all sectors and lock selected sectors:

```
   AUTO SPECIALS        S--
UNLOCK   LOCK
```

Pressing UNLOCK toggles the setting of special 1, pressing LOCK displays a further menu allowing you to select or deselect locking of each sector in turn.

```
SECTOR 00 @ 000000:LOCK
CHANGE  NEXT        OK
```

Pressing CHANGE toggles the setting (displayed on the top line) of the currently displayed sector, NEXT moves you on to the next sector. Note: the addresses of the sectors are different in the 29F400T and 29F400B, these features are described in further detail in the AMD data books.

Motorola 68HC705C8

This device has one special function - lock.

```
MOT 68HC705C8          S-
```

```
AUTO SPECIALS          S-
 LOCK
```

Special Functions on MP5

Using the MP5 module it is possible to program the special lock and encryption functions either manually, or automatically during programming. The following notes describe how to achieve this.

The display shows how many functions are available while it is displaying the type selected. It does this in the top right of the display with a typical format:

```
SIG 87C51             S---
REPORT
```

The 'S' stands for 'specials' and the following - characters indicate three functions which are currently switched off.

The three functions for a Signetics 87C51 are:
1.   Lock Bit 1
2.   Lock Bit 2
3.   Encryption Table

For example if lock bit 1 and the encryption table were currently turned on for automatic activation after normal programming the display would show:

```
SIG 87C51              S1-3
REPORT
```

Setting an automatic special function
Press the FUNC key, then the DEVICE key, then the cursor right key, then SPECL and the display will show:

```
SPECIAL FUNCTIONS
 AUTO MANUAL
```

Pressing AUTO will result in a display for setting up automatic programming functions:

```
     AUTO SPECIALS      S---
 LOCK1 LOCK2 ENCRY
```

Use the 3 function keys to set up the desired automatic functions and then press EXIT until you reach the top level display again.

Performing a manual special function
Press the FUNC key, then the DEVICE key, then the cursor right key, then SPECL and the display will show:

```
SPECIAL FUNCTIONS
 AUTO MANUAL
```

Pressing MANUAL will result in a display for performing manual programming functions:

```
MANUAL SPECIALS
 LOCK1 LOCK2 ENCRY
```

Pressing one of the 3 function keys marked will immediately perform the labelled function on all the socketed devices. Please refer to the manufacturers data sheet for the device concerned for any restrictions on the order in which special function may be used.

Special note on the encryption table in 87C51 family devices

The encryption table is a table of data up to 64 bytes long used to provide security on programmed devices. The programmer takes the data for this table from its RAM buffer at an address immediately following the last programmed code address. This data can be loaded into the RAM by manually editing it, or by downloading it from a PC using PROMDRIVER. The latter is the preferred method as it prevents errors. The size of file download will automatically default to a size which includes the size of the encryption table, and it is left to the system user to ensure that the encryption table is present and at the correct object file address.

For example in an 87C51 the encryption table must start at address 1000 hex and finish at address 101F hex.

Special note on the Signetics / Philips implementation of security functions

After lock bit 2 is programmed on such devices the silicon signature is no longer functional. This is not a fault of the programmer, rather a characteristic of the devices. Trying to read the device will result in an 'ID error'.

Special Functions on MP2/4

Atmel 29C040

Using the MP2 or MP4 module it is possible to protect parts of Flash devices etc. where appropriate.

```
ATMEL 29C040           S---


```

The Atmel AT29C040 has 3 lock functions - data protection enable (LOCK) and lock out of the lower and upper boot blocks.

```
   AUTO SPECIALS       S---
 LOCK   LK-LOW  LK-UP
```

Note: that data protection enable must be done at the same time as programming, so no equivalent manual function is available. To disable data protection, program the part with the correct data with the auto special function number 1 switched off.

Atmel 29C512 & 29C010

Using the MP2 or MP4 module it is possible to lock these Flash devices.

```
ATMEL 29C512           S-


```

```
   AUTO SPECIALS       S-
   LOCK

```

Note: that data protection enable must be done at the same time as programming, so no equivalent manual function is available. To disable data protection, program the part with the correct data with the auto special function switched off.

Atmel & Xicor 28C010, SST 28EE011 & 29EE010

These devices can be locked and unlocked on the MP2 or MP4 module.

```
ATMEL 28C010           S--


```

```
   AUTO SPECIALS       S12
UNLOCK LOCK
```

When the AUTO settings are enabled, unlocking occurs before programming and locking occurs after programming. It is necessary to make the above selection if locked devices are being re-programmed as it is not possible to program a device when locked .

Special Functions on MP6

Microchip 24C65

```
MCHIP 24C65            S--


```

The 24C65 has the capabilities of selecting a block to be high endurance (ENDUR), and of write-protecting a range of consecutive blocks (LOCK).

```
   AUTO SPECIALS       S--
ENDUR LOCK
```

Selecting ENDUR will allow you to select the number of the high endurance block. (Please note that this will NOT affect data already programmed into the device.)

```
High Endurance Block ON
   15            OFF   OK
```

Selecting LOCK will allow you to set the number of the first block to be protected and the quantity of blocks.

ELECTRONICS

```
Start    Qty  : ON
   15    00   OFF       OK
```

READing a master device will set the special function parameters automatically.

Xilinx 17000 Family

```
XILIN 17128D            S-
```

The Xilinx range of serial EPROMs include parts which can have their RESET polarity programmed to an active low state.

```
   AUTO SPECIALS        S-
RST-LOW
```

Reading a master device will set the special function automatically.

ELECTRONICS

Appendix C

Modules

The various modules for the programmer system that have been developed to date are:

TYPE	DEVICES	SOCKETS	MEMORY REQUIRED
MP2	BYTE WIDE EPROMS	8 + 1 DIL	OPTIONAL
MP3	WORD-WIDE DEVICES	8 OFF DIL	8/16 MBIT
MP4	BYTE WIDE EPROMS	16 OFF DIL	8/16 MBIT
MP5	8751/8748 FAMILIES	8 OFF DIL	8 MBIT
MP6	SERIAL E(E)PROMS	8 OFF DIL	8 MBIT
MP9	SIMMs, PCMCIA (see below)	see below	see below
SOCKET CARDS FOR MP9			
MP9-01	80 pin SIMMs	8	typ.128 MBIT (MM70)
MP9-02	72 pin SIMMs	8	typ.128 MBIT (MM70)
MP9-03	PCMCIA cards	8	typ.128 MBIT (MM70)
MP9-03 EJECT	PCMCIA cards	8 Eject Sockets	typ 128 MBIT (MM70)
MP9-04	72 pin DIMMs	8	typ 128 MBIT (MM70)

As the PIC family of microprocessors from Microchip Technology Inc. have 12, 13 or 14 bit busses they are programmed with the module MP3 with the appropriate MQP package converters.

Switch off the programmer before changing programming modules. Carefully lift out the installed module, and put the new module into the connectors in the module recess without bending pins.

Note: the current settings will be lost except for those under the RECALL key. Therefore certain hex keys can be used to STORE device types and associated settings for each programming module. This makes it easier to set up a repeat job whenever the module is changed. (See STORE & RECALL keys under FUNC and TYPE).

The Display

The viewing angle of the display can be adjusted if it is difficult to read in the lighting conditions. Firstly switch off the programmer and switch on again with the hex digit 1 pressed down. The viewing angle can be changed using the two cursor keys ←and→. When the ideal setting has been selected press OK and the new value will be stored and used until it is changed again.

Volume Control

The key clicks and tones generated by the programmer can be controlled in two ways. Firstly they can be silenced completely by using the SOUND key under FUNC.

Secondly there is a volume control fitted under the programming module. To adjust this, turn the programmer off and carefully remove the module. Turn it back on (you will get a warning message but you can ignore this by pressing OK). You will see a small hole in the bottom left hand corner of the recess where the module usually sits. Under this hole there is a control that can be adjusted by turning clockwise (louder) or anti-clockwise (softer) until key presses and error bleeps are at the required volume. Switch off the programmer again before re-installing the module.

Software and Firmware Upgrades

The manufacturer is regularly upgrading the firmware of its products to keep pace with developments in the field. See Appendix I for further information about software support. Please contact your supplier.

Switch off the programmer before removing the programming module. Carefully lift out the module. You will see one or two EPROMs in the module recess. Carefully remove these with a PROM extractor tool. Put the new EPROM(s) in the socket(s) making sure that the notch indicating pin 1 is at that the top and that all the pins go into the sockets.

Note if two EPROMs have been supplied, EPROM number 1 must go into

the left hand socket. The programming module can then be carefully re-inserted.

Unfortunately changing the firmware version erases all settings that have been stored in the programmer, except for the viewing angle (see above).

Changing the RAM Module

A new RAM module may be required because the current one is not big enough for the larger devices or different split and set parameters that you want to use.

New or replacement modules can be obtained from your supplier. Switch off the programmer before removing the RAM module. Undo the screw at the top of the keyboard/display module. Slide this module up slightly and carefully lift up. Note that it is attached to the main unit with a length of ribbon cable. DO NOT PULL THIS TIGHT. The RAM module can then be gently removed and the new one fitted the same way round.

Re-fit the keyboard/display module by sliding it down until the tongue

at the bottom fits inside the main unit. Replace the screw at the top and gently tighten it while pressing down on the back end of the keyboard/display module (DO NOT OVERTIGHTEN).

The size of the new RAM module can be displayed using the INFO key under FUNC (see chapter 13).

Physical Description

Length: 38 cm

Width: 23 cm

Height: 15 cm (MP2, MP4 at highest point)

Weight: 10 kg (including typical programming module)

Technical Specification

Power consumption: 70W max.

Voltage settings: 110 or 240V (self selecting)

Appendix D

WARNINGS

DO NOT remove devices from sockets while the socket led is showing operation in progress (LED on with short flashes off).

DO make sure that the programmer is switched off before removing or inserting a programming module or ram module.

It is not good practice to switch on or off the programmer with device(s) in the sockets. This can damage some devices.

Warning on AUTO Typesetting:

Automatic type setting depends on the device in question having been provided by the manufacturer with a so-called Silicon Signature. Not all devices can be identified in this way. Indeed some devices can even be damaged by the special voltages involved. In particular some microprocessors which are claimed by their manufacturer to program like a particular EPROM using a pin converter will probably never contain the silicon signature. NEVER use auto type selection if you know that the device does not contain a silicon signature.

If you ever need to transport the unit, please observe the following precautions:

- Switch off and disconnect from the mains
- Unplug the programming module
- Pack the programming module and the chassis separately. If you cannot find the original purpose made packaging, please ensure that each item is well padded.
- The mains lead should be packed separately to avoid damage to the equipment from plugs.

ELECTRONICS

Appendix E

Error Messages

NO DEVICE TYPE SELECTED	
	OK

A device type must be selected before this operation (see TYPE single function key).

COPY SOCKET EMPTY	
	OK

ROM master has been chosen but the copy socket is empty or contains a faulty device.

RAM MASTER ONLY ALLOWED	
	OK

For certain devices it is not possible to program from the copy socket. You must program from RAM.

INSUFFICIENT RAM	
	OK

A larger RAM module is required to carry out this operation.

CANNOT READ - ROM MASTER	
	OK

It is not possible to READ from the copy socket, it must be done from socket A1 (see READ single function key).

ALL SOCKETS EMPTY	
	OK

A1: NO DEVICE	
	NEXT

Device absent from this socket.

A1: PROGRAMMED OK	
	NEXT

Device in this socket has been programmed and verified correctly.

A1: ID ERROR	
	NEXT

Device in this socket has the wrong silicon signature, i.e. it is the wrong type or faulty.

A1: MOUNT ERROR	
	NEXT

The device in this socket is inserted wrongly or is faulty.

A1: PROG FAIL @ 000000	
	NEXT

The device in this socket failed to programmed, probably due to a faulty device.

ADDRESS LINE ERROR	
	OK

A fault has been detected in a device.

RAM HAS NOT BEEN LOADED	
CONTINUE	ABORT

An attempt has been made to program when the master RAM is all blank, i.e. after a restart or CLR operation. Press continue if you want to program all the devices with the blank value.

This feature not yet	
implemented	OK

Please contact your distributor to see when this feature will be implemented.

Not applicable to this	
device	OK

An attempt has been made to perform a function that is not allowed with the currently selected device, e.g. trying to ERASE a UV erasable PROM.

NO PROGRAMMING MODULE	
	OK

ELECTRONICS

The programmer has been switched on without a programming module. No useful operations can be performed except adjusting the volume control (see appendix C).

```
SOFTWARE VERSION ERROR -
REQUIRES AT LEAST VXX.XX
```

The programmer has been switched on with the wrong firmware version for this programming module. Please contact your supplier.

```
PIN CONVERTER REQUIRED
                           OK
```

```
WRONG ADAPTER
                           OK
```

Certain devices need our adapters in the sockets to be able to program, verify or read them. Please contact your supplier if you are not sure which pin converter to use.

Appendix F

Common Questions and Answers.

Q. After programming (or reading, blankchecking or checksumming) a set of EPROMs the report does not stay on the display. Why is this?

A. You have the detect empty socket turned on and the devices you are using are not suitable for use with this function, as they cannot be detected by the method the programmer uses.
The solution is to turn off the detect empty socket function when programming this device type (press FUNC then MISC to locate this setting).

Q. Although devices blank check OK, they always fail to program, usually with a failure at a low address.

A. The devices are damaged or especially for devices like the 87C51 one or more of the lock bits have been blown.

Q. I want to program a micro-processor or micro-controller that the manufacturer says programs exactly like an EPROM. How do I do program it?

A. First, set the device type to the EPROM, note, you will not be able to use auto type select to do this. Then you may need to do any of the following: if the address space of the micro is smaller than the EPROM use ADDR in the FUNC, DEVICE menu to change the lower and upper addresses. If large and complicated package converters are being used, you may have to reduce the access time for reading and checksumming the device. This can be found under the Tacc key in FUNC, DEVICE. Also you will need to turn ID Check off, also see under FUNC, DEVICE.

Q. I get a different checksum with this programmer than another manufacturer's programmer.

A. This programmer calculates the checksum by adding the contents of all bytes in the device. The bytes that are not defined in the downloaded file are filled with the blank value for the device, usually FF hex. Some other programmer manufacturers fill the unused parts of their devices, especially 8751s, with 00. To make the checksums the same, we have added a special feature to allow you to select 00 as the blank value when filling unused parts of devices. It can be found under the FUNC, RAM, EMPTY keys.

Q. All the text is being displayed in German or another language.

A. The language selected has been changed. Change it back to English by selecting ENGLISH in the FUNC menu; press FUNC, ← and f1. Note, the selected language is

ELECTRONICS

stored along with device information under the numeric RECALL keys, so if the language keeps changing, check these, see RECALL in the FUNC menu.

Q. *When I put some devices into the socket, the warning buzz sounds continuously.*

A. Either you have put the device into the socket incorrectly, upside down or not in the bottom part of the socket, or you are using an older device that fails the reverse check. Old Intel 8751H (21V) cause the reverse detect to operate all the time. The way to stop this is to disable reverse check, see in the FUNC menu under DEVICE.

Q. *When I switch the programmer on, the display briefly shows the message:*

```
UNSUPPORTED
```

Why?

A This programmer has never been registered, and is by default supported up to 1st February 1996. So all devices added to the menu before that date are automatically supported. For support for devices added after that date registration is required (see appendix I).

Q. *When I use auto device selection of a certain device why do I see the message:*

```
CHECK SUPPORT CONTRACT
                      OK
```

A. This device was added to the menu after the expiry of software support for the programmer, so is not available without a new registration code (see appendix I).

Q. *How do I know which sectors are locked on AMD 29F400B or 29F400T devices?*

A. Select the correct device type and read the device into memory. The 11 word locations following the end address of the device (40000 hex onwards) will contain in ascending order the setting for each sector. If the least significant bit is 0 the sector is locked, if set to 1 the sector is not locked.

Q. *How can I find out which special features are enabled on Microchip PIC devices?*

A. Select the correct device type and read the device into memory. The special feature (configuration word, ID value and locked status are stored in memory at the addresses recommended by Microchip, Inc (see the relevant data books).

Q. After certain operations (READ, VERIFY etc.) a 🔑 *icon appears (and ALERT flashes) on the screen.*

A. This symbol appears against the checksum if master checksum monitoring is currently enabled (see under FUNC, DEVICE, CKSUM, MASTER).

ELECTRONICS

Appendix G

Tree Structures of programmer functions

TYPE Key
TYPE

— AUTO   Select type by Silicon Signature

— NAME   Select type by Manufacturer and Name

— CODE   Select type by 4 digit Device code

— RECALL Select type by recalling stored type

EDIT Key
EDIT

— EDIT BUFFER   Edit contents of RAM buffer

       — COPY   Copy a block of data
       — FILL   Fill a block of data
       — NEW   Clear a block of data
       — INVERT   Invert the polarity of a block of data
       — SEEK   Search for a sequence of bytes
       — INSERT   Put into insert mode
       — DELETE   Put into delete mode
       — KILL   Move data down

— DISPLAY BUFFER   Examine contents of RAM buffer (can't be modified)
— EDIT DEVICE   [Reserved for future function]
— DISPLAY DEVICE   Examine contents of device (can't be modified)

FUNC Key

FUNC
```
├─STORE                    Saves settings under a numeric key
├─RECALL                   Recalls settings under a numeric key
├─RAM
│    ├─ SPLIT                   Settings for 8 bit devices on 16 or 32 bit busses
│    ├─ SETS                    Settings for sets of different PROM patterns
│    ├─ CKSUM
│    │    ├─BITS                     Set no. of digits displayed in checksums
│    │    ├─DIGITS                   Set size of data summed
│    │    └─MASTER                   Set and enable master checksum
│    ├─ CLR                     Clear all RAM
│    ├─ DISP                    Set size of data displayed
│    ├─ EMPTY                   Set RAM empty fill value
│    ├─ PATT(n)                 Store either of two test patterns in all RAM
├─DEVICE
│    ├─ ID                      Enable/disable id check
│    ├─ MOUNT                   Enable/disable mount check
│    ├─ REVERSE                 Enable/disable reverse check
│    ├─ BLANK                   Enable/disable blank check and set blank value
│    ├─ VERIFY                  Set verification Vcc limits
│    ├─ SPECL
│    │    ├─AUTO                     Enable lock functions etc. of devices
│    │    └─MANUAL                   Perform lock functions etc. of devices
│    ├─ CKSUM
│    │    ├─BITS                     Set no. of digits displayed in checksums
│    │    ├─DIGITS                   Set size of data summed
│    │    └─MASTER                   Set and enable master checksum
│    ├─ Tacc                    NORMAL or SLOW device access time
│    ├─ ADDR                    Set reduced address space
├─COMMS
│    ├─ HOST                    PC or non-PC host
│    ├─ PORT                    Serial port parameters (set by host in PC mode)
│    ├─ PRINTER USED?           Use parallel port for comms or printer
│    └─ PRINTER PARAM           Set up printer (if enabled)
│         ├─INIT                     Set up initialisation string
│         ├─TERM                     Set up termination string
│         └─PAGEL                    Set up page length
├─MASTER
│    ├─ ROM                          Set ROM (COPY skt.) to program and verify from
│    └─ RAM                          Set RAM to program and verify from
├─INFO                     Various useful programmer information displays
├─SOUND
│    ├─ CLICKS                       Enable/disable
│    └─ TONES                        Enable/disable
├─EETEST                   Performs goods inwards test on re-programmable devs.
├─MISC.
│    ├─ KEYBOARD LOCK                Locks/unlocks settings
│    ├─ EMPTY SKT DETECT             Enable/disable
│    ├─ START UP TESTS          Enable/disable
│    └─ START: CLR ALL RAM           Enable/disable
├─ALG.                     Prog. algorithm and info. of selected device
├─PRINT                    Print device list
├─DEUTSCH | ENGLISH        Change display to other language
├─STAT                     Enable/disable programming statistics
│
```

Appendix H

Serial and Parallel Port Connections (PC Host Mode)

When using a PC compatible in 'PC Host' mode (in other words using PROMDRIVER software), it is only necessary to use the 25 way D-type lead provided, with the possible addition of the 25 pin to 9 pin converter.

The 25 way D-type lead is a standard fully connected male to female type.

Use of the parallel port is preferred for reasons of speed, and immunity from interaction with other software.

To transfer files to or from a device other than a PC compatible it is necessary to use 'Non-PC Host' mode, and to set up a suitable communication link between the programmer and the other device.

Two methods are possible; serial (DOS only) and parallel.

Parallel (Non-PC Host)

The parallel port is one way only (into the programmer) and emulates exactly the port on a Centronics printer. This allows file transfers from any computer using a command to 'copy a file to a printer'. It has the advantage of speed and allows the use of standard existing operating system software.

The programmer parallel connector is a 25 pin plug (male) with the following pin connections:

Pin	Direction	Name	Description
1	INPUT	\STROBE	Mandatory.
2	INPUT	D0	Mandatory.
3	INPUT	D1	Mandatory.
4	INPUT	D2	Mandatory.
5	INPUT	D3	Mandatory.
6	INPUT	D4	Mandatory.
7	INPUT	D5	Mandatory.
8	INPUT	D6	Mandatory.
9	INPUT	D7	Mandatory.
10	OUTPUT	\ACK	Mandatory.
11	OUTPUT	BUSY	Mandatory.
12	OUTPUT	PE	Mandatory.
13	OUTPUT	SLCT	Mandatory.
14	INPUT	\AUTO FD XT	Optional.
15	OUTPUT	\ERROR	Mandatory.
16	INPUT	\INIT	Optional.
17	INPUT	\SLCT IN	Optional.
18-25	-	GND	Mandatory.

Such a lead should ideally not exceed 2m in length.

Serial (RS232C) (Non-PC Host - DOS only)

The Programmer is equipped with a female 25 way D type connector. Only pins 2, 3, 5, 6, 7, 8, and 20 are connected. The pinout is designed so that connection to an IBM PC, XT or compatible may be achieved using an off-the-shelf male to female RS232C cable which connects at least pins 2, 3, 5, 6, 7, 8 and 20.

PC Compatible Connection Chart.

IBM PC 25 pin male		IBM AT 9 pin male		PROGRAMMER. 25 pin female	
1	GROUND				
2	TXD O/P	3 TXD	O/P	2	TXD I/P
3	RXD I/P	2 RXD	I/P	3	RXD O/P
4	RTS O/P	7 RTS	O/P		
5	CTS I/P*	8 CTS	I/P*	5	CTS O/P*
6	DSR I/P	6 DSR	I/P		
7	GROUND	5 GROUND		7	GROUND
8	DCD I/P	1 DCD	I/P	8	DCD O/P
20	DTR O/P*	4 DTR	O/P*	20	DTR I/P*
22	(RI) I/P	9 (RI)	I/P		

*flow control

If using the programmer with other host computers a lead may be made up quite simply using the following table of pin connection requirements. The use of pins and

---

connector by the Programmer is that of a DCE (Data Communications Equipment) or Modem.
WARNING: When used with PROMDRIVER under MS-DOS all connections are mandatory. The optional requirement referred to in this section refer to 'Non-PC Host' mode operation only.

Programmer Pin Descriptions for use with Devices other than PC Compatibles

Pin	Direction	Function	Name	Description
2	INPUT	TXD	Data to Programmer	Mandatory
3	OUTPUT	RXD	Data from Programmer	Mandatory
5	OUTPUT	CTS	Data Flow Control by Programmer	Optional output. Programmer turns this signal OFF if it is not prepared to accept a character at that time.
6	OUTPUT	DSR	Programmer On.	Optional output. This output is ON whenever the Programmer is powered.
7	-	GND	Signal Ground.	Mandatory.
8	OUTPUT	DCD	Programmer On.	Optional Output. This output is always ON when the Programmer is powered.
20	INPUT	DTR	Data Flow Control by Host.	If not used connect to pin 8 (IMPORTANT). If DTR is taken to OFF by the Host data transmission on pin 3 will cease after the character currently being transmitted, and will resume when DTR is taken to ON.

RS232 specifies that ON is a voltage greater than +6 and OFF is a voltage more negative than -6.

Flow Control
Three types of data flow control (or lack of it) are used by the Programmer.
RS232 control line.
The Programmer will stop transmitting if the RS232 line DTR is turned OFF, and will continue if it is turned ON. It will turn CTS to OFF if it wants the other end of the link to stop transmitting.
XON / XOFF.
The Programmer will stop transmitting if it receives an XOFF character (Control/S). It will resume if it receives an XON (Control/Q). It will send XOFF if turning CTS off did not stop the flow, and will send XON when ready for further characters.
None.
no flow control is implemented then successful communication depends on the devices in communication being able to process all input and output communication without character loss. This is unlikely at any usefully high data rate.
'PC-Host' mode protocol uses hardware handshaking, using DTR and CTS. If using 'Non-PC Host' mode, it is essential to provide a form of positive flow control when transmitting files to the Programmer. Hardware handshaking is strongly recommended.

Speed
It is possible that the host may not be capable of handling the full 38400 bits/second rate without losing characters. If you have problems on a particular computer

with data loss as evidenced by 'Format Error' messages or lock-ups then try running the serial link at a lower rate.

Interface Parameters

If using 'Non-PC Host' mode the serial port of the host computer should be set up with the same parameters as the programmer.

Mode:	Asynchronous.
Speed:	300, 600, 1200, 2400, 4800, 9600, 19200 or 38400 b/s. The higher the rate the faster the data transfers will take place.
Bits:	8 data + no parity bit + 1 stop bit. This is the recommended setting but any setting with at least 7 data bits will be OK for Intel and Motorola Hex data formats, provided both ends of the link are set the same.
Receive Parity:	Ignore.

ELECTRONICS

Appendix I

Support Arrangements

- Software Support
- Socket Lifetime
- Repair Service Outside Warranty

Software Support

Software support includes support of new device types and the addition of new software features. The purchase of every system includes one year's free support. After this year the user may purchase support for a further year at a time, or take out a continuous support contract for which he will be invoiced at a reduced price at yearly intervals.

During a year of support the user will have access to updates, via our Web-page, at any time. Any user requiring extra upgrades and not having access will be asked to pay a handling charge.

The updates are available to any user at any time. Devices added after the registered support date cannot be selected, but they will be shown in the menus for information. On the standalone programmer display they will flash to show that they are not available. On the PROMDRIVER type selection screen they will be displayed in red (DOS) or have a red cross against them (Windows).

Registration of a new Programmer

A new programmer must be registered by the user. Without registration only limited device support is available. If the programmer is used with a PC registeration is automatic. If no PC is available, a code from the supplier must be typed into the programmer keyboard. On request the distributor may pre-register the unit before sending it out.

Activation of further year's support.

To purchase a year's extra support the user must supply the serial number of the programmer. He will receive a new registration number in return. This number may be entered directly into the programmer, or into the PROMDRIVER registration screen. Full details of this operation are given below.

Support for new devices.

MQP will add support for any new device, provided that it is technically feasible, at the request of any customer purchasing a new system, or with current software support. This assumes that the device is not obsolete or obsolescent, and is subject to availability of data and samples.

This will be free of charge provided no hardware changes or additions are necessary. If further hardware is required an additional Development Charge may be required.

Web Page

See front page of this manual for the address.

WHAT TO DO

Automatic Support Registration Using Promdriver

The easiest way to register your new programmer is to connect it to a PC and run it in remote mode. The programmer will automatically be registered for one year's free software support.

To use this method of registration, follow the PROMDRIVER installation instructions. After the installation screen, you will see a window informing you that registration has completed. During registration a 12 character registration code is assigned to your programmer. This code is stored, for your record, along with the programmer serial number, in a file called REGIST.TXT. This is stored in the same directory as the PROMDRIVER application software. Further registrations of the same or other programmers will be added to the file.

Please do not edit or delete this file. It is your record of programmer registration information. The programmer itself stores the registration information, so any computer may be used in future to remotely run the programmer

Manual Support Registration

If the programmer is new then please try to use the automatic registration method described above, as it will save much time.

However there are a number of situations where it will be necessary to register the programmer in stand-alone mode. Examples are:

- The programmer is new but there is no PC available.
- The programmer is not new and needs to be registered or re-registered to allow it to work with the latest software version.

Manual support registration in stand-alone mode

The procedure is as follows:

Contact your supplier and obtain a 12 digit registration code. The supplier will need to know the programmer serial number (from the label on the back of the unit). The registration code will be provided free of charge provided that the programmer chassis, or a module for that chassis is still in warranty. Otherwise he will have to make a charge for software support.

- Switch on the programmer while holding down the '9' key on the programmer keypad. You will see the registration status.

If the unit is new and has never been registered the message will read:

```
Please register
 (see manual)           OK
```

If you do not register, only a few sample devices will be available. Other device type names will flash and cannot be selected. Press OK and you will see the Registration

---

Screen (see below)

If the unit is not new it will show the date until which it is currently supported. Older units which have never been registered will display:

```
Registered till 01Feb96
REGISTER                  OK
```

Units which have been registered previously will show a later date than this.

Press register to proceed to the Registration Screen.

Registration Screen

```
S/No  REGISTRATION
351126 000000000000    OK
```

First check that the serial number shown is the one for which you obtained the registration code. IMPORTANT: If they do not match, registration will fail. (Note: The first two serial number digits may be shown as 00 on older units. You may correct this using the left arrow key. This is not mandatory, it is just for your own convenience.)

Type in the registration code exactly as shown. Then press OK. The new registration status will be shown. e.g.

```
Registered till 03Jul02
REGISTER                  OK
```

Press OK and the programmer will start normally.

Manual support registration using PROMDRIVER

This is similar in principle to stand-alone registration, but is accessed from the PROMDRIVER main menu. Press the F4 key to start the registration process. You will need the registration code from your supplier.

Information about support status

In stand-alone mode press the FUNC - INFO keys and look for the item:

SUPPORT TILL:

This shows the date to which software support is currently provided. Devices added after this date cannot be selected.

In remote PROMDRIVER mode press the F3 key from the main menu. The Info screen will give the unit serial number and the support date (from PROMDRIVER V8.94 onwards).

Socket Lifetime

One of the most critical components of a programmer is the ZIF socket. Although the sockets used on our programmers are hard-wearing, they still have a limited lifetime. Even under ideal conditions this will be around 25,000 cycles. When exposed to a non-clean environment this can be further reduced. For

example touching the pins of the devices being programmed can transfer contamination to the ZIF socket. Eating food in the vicinity of the programmer, leaving the programmer uncovered when not in use, or using in a dusty environment can all lead to socket contamination. The first signs of contamination will be hard to identify and are often just assumed to be caused by a programmer fault, as the number of programming failures rises.

Another source of socket failure is simply rough handling. The more modern device packages in particularl require more delicate socket mechanisms, and many of these can easily be damaged by mishandling. For surface mount devices in particular, it is essential to use a proper vacuum pen handling tool.

As a result of these potential causes of socket damage, we have to exclude the sockets from our initial warranty and from any service contract, except where a failure is due to faulty manufacture of the socket. If sockets need cleaning or replacing during the guarantee period, we will have to charge accordingly.

Cycle Counter

All our programming modules have a cycle counter, which counts programming cycles. This count may be displayed at any time under INFO. When the maximum socket lifetime is reached, a message will appear on the programmer display when it is first switched on, saying

```
SOCKET LIFETIME EXCEEDED
```

Pressing OK will remove the message and operation may continue as normal.

To maintain reliability of programming it is important not to use the programmer beyond the lifetime of its sockets. On reaching the socket count limit we recommend that you arrange with your supplier to return the module for refurbishment. A charge will be made for parts and labour involved.

It is not cost effective to ignore this advice. For example the sockets used on MP4 can, if used in clean conditions, program 25000 x 16, or 400,000 devices before socket lifetime is exceeded. This represents several thousand pounds worth of devices. The loss, due to faulty programming caused by worn or dirty sockets, of even a small proportion of the next 400,000 parts will cost very much more than a timely refurbishment.

For a production department needing continuity of operation a

ELECTRONICS

cost efficient solution is to purchase a second module to cover the time when a module is not available due to refurbishment.

Repair Service Outside Warranty

Repairs are carried out on a 'return to base' basis, normally within 2-3 days. The price will be available on request.

If a programmer returned for repair is found to have unreliable sockets, caused by contamination, or if the socket lifetime is exceeded, we will normally replace all sockets, making a charge for this, and reset the cycle counter.

Hardware Service Contract

An annual service contract is available. The contract covers repair of faulty units which have not reached their socket cycle limit.

Not covered are:

- faults caused by misuse
- replacement of sockets
- adapters
- carriage charges in either direction
- software upgrade support (see SoftwareSupport)

If a unit has reached its socket cycle limit, or has contaminated or damaged sockets, the normal charges for refurbishment will apply before the unit can be repaired under the service contract.

Transporting the Programmer

- Please observe the following advice if you need to transport your programmer:
- If possible use the original purpose-made packaging.
- Always unplug the programming module and pack it separately.
- Pack any leads or other items returned separately.
- Remember that the programmer components are heavy and will undoubtedly receive rough handling in transit. Apply sufficient padding accordingly.
- Always obtain a returns number first if returning a unit to us.
- Always send the programmer prepaid, or it will not be accepted.

Manual S2000 range -Rev 07   Copyright © 1992-2001 MQP Electronics Ltd