

USER MANUAL

For the

ChipWriter

GANG

Gang/Set Memory Device Programmer

DATA I/O

without limitation, loss of use, loss or alteration of data, delays or lost profits or savings, arising from the use of this document, or the product which it accompanies.

No device of this document may be reproduced or transmitted in any form or by means, electronic or mechanical, for any purpose without written permission from Data I/O.

Data I/O Corporation
10525 Willows Road N.E., P.O. Box 97046
Redmond, Washington 98073-9746 USA

Acknowledgements:

Data I/O and ChipWriter are registered trademarks of Data I/O Corporation.

Data I/O Corporation acknowledges the trademarks of other organizations for their respective products or services mentioned in this document.

© 1997 Data I/O Corporation
All rights reserved.

Contents

1.	Introduction	
1.1	About this manual	5
1.2	Conventions	5
1.3	Contents of the package	5
1.4	Low Voltage Features	6
2.	Installing the Software	
2.1	Latest software	
2.2	Making a backup	7
2.3	Running the Install program	7
3.	Connecting the programmer	
3.1	Computer requirements	8
3.2	Connection to the PC	8
3.3	Running the Detect program	8
3.4	Programming non-DIP devices	9
3.5	Adapter usage	9
4.	Getting Started	
4.1	Before you start	10
4.2	Quickstart Tutorials	
	TUTORIAL 1: Programming from a master device	11
	TUTORIAL 2: Programming from a file on disk	11
	TUTORIAL 3: Programming in Stand Alone Mode	11
5.	Using the Programmer Software	
5.1	Basic software configuration	13
5.2	Screen layout	13
5.3	General menu selection	13
5.4	Dialogue boxes	13
5.5	Changing the settings	14
5.6	Using a mouse	
	5.6.1 Left mouse button	14
	5.6.2 Right mouse button	14
	5.6.3 Middle mouse button	14
	5.6.4 Disabling the mouse	14
5.7	Edit and View modes and scroll bars	15
5.8	Running the software under Windows	15
5.9	Using the programmer over a network	15
	5.10 Using EMS software	16
	5.11 Saving software settings	16
6.	Memory Programming	
6.1	Device Selection	16
	6.1.1 To select a device	16
	6.1.2 Changing the device selected	17
	6.1.3 Which device?	17
	6.1.4 Low Voltage devices	17
6.2	Opening files	
	6.2.1 Setting the buffer size	16
	6.2.2 Opening a file	18

6.2.3	File format	18
6.2.4	Handling large files and devices	18
6.2.5	File checksum	19
6.2.6	Wild card loading	19
6.2.7	Saving a file to disk	19
6.3	Using the editing facilities	
6.3.1	Entering Edit mode	20
6.3.2	Viewing without making changes	20
6.3.3	Quickly moving around the buffer	20
6.3.4	Quick buffer editing commands	21
6.3.5	Buffer checksum	21
6.3.6	Buffer display modes	21
6.3.7	Printing the buffer contents	21
6.4	Reading and checking devices	
6.4.1	Reading a device	22
6.4.2	Verifying the contents of a device against a file	21
6.4.3	Verifying the device type	23
6.4.4	Checking if a device is blank	23
6.5	Programming and erasing devices	
6.5.1	Programming a device	23
6.5.2	Single key programming	24
6.5.3	Programming in Gang mode	24
6.5.4	Programming in Set mode	24
6.5.5	Erasing devices	24
6.6	Adding user-defined devices	24
7.	Stand Alone Mode	
7.1	Setting up the internal EEPROM	26
7.2	Setting up a Master configuration EPROM	26
7.3	Operating in stand alone mode	26
7.3.1	Programming in stand alone mode	26
7.3.3	Verifying in stand alone mode	27
7.4	LED Codes	
7.4.1	LED status	27
7.4.2	Power-up codes	27
8.	Batch Software	
8.1	Batch software commands	28
8.2	Batch software utilities	30
8.3	Batch software error levels	31
8.4	Batch software examples	32
APPENDICES		
	Appendix A : Shorthand Keystrokes	33
	Appendix B : Menu Options	34
	Appendix C: Technical Support	37

1. Introduction

Congratulations on choosing a Data I/O programmer. We are sure that you will find the programmer software very easy to use and full of features to support all of your device programming requirements, whether you are using it for development, prototyping, or production. With your new programmer you will be able to

- ✓ Read data from programmed devices
- ✓ Load data in from disk
- ✓ Edit data
- ✓ Program a device
- ✓ Verify the contents of a device

There are many other features of your programmer that will allow you to achieve the results you need. This manual explains these features and how to use them.

1.1 About this manual

This manual provides the instructions for using the ChipWriter Gang programmer from Data I/O. It is organised so that is possible to program a device quickly by following one of the *QuickStart* tutorials. A reference section follows which gives instructions in greater detail.

It is recommended that you read through the relevant sections of this manual before using the programmer, and in case of any difficulties you encounter during use.

1.2 Conventions

Throughout this manual, certain conventions will be followed in typefaces in order to make instructions clear.

e.g. In the line

Type CWGANG <ENTER>

CWGANG : this font is used throughout the manual to indicate something that should be typed into the computer

<ENTER> : Any items encased in <> indicate the name of a key to press, i.e. in this case the ENTER key.

Menu selections are indicated by words separated by forward slashes, e.g. FILE/OPEN indicates the menu option FILE, followed by the sub-menu option OPEN.

Messages displayed by the programmer software are indicated in italics, for example:

DEVICE PROGRAMMED AND VERIFIED

1.3 Contents of the package

When you open the package you will find :

- ChipWriter Gang/Set memory device programmer
- A CD-ROM disk containing the software for the programmer.
- User manual
- A power supply
- A D25 male - male parallel cable

1.4 Low Voltage features

The ChipWriter programmer supports Low-Voltage, 3.3V, devices, as well as standard 5V devices. It is capable of programming and verifying devices down to 3.3 Volts. When using low voltage devices, the programming algorithm in the system will automatically be set to the correct voltages as specified by the manufacturer's programming algorithm. Even if the manufacturer's algorithm specifies verification at 5 volts you may wish to verify a device at 3.3V to ensure that it will work in your low voltage system, with the capabilities of the ChipWriter Gang programmer it is possible to do this. See section 6.6 for full details.

2. Installing the Software

2.1 Latest software

Both device programming algorithms and system operating firmware for this new line of programmers is distributed via the BBS/WEBpage. Since software algorithms are added on a regular basis, it is possible that the software provided with your new system has been superseded by a newer revision.

To insure that you have all the latest device programming algorithms and system software installed on your programmer, it is recommended that you contact the Data-IO BBS/WEBpage.

Contact information is given in Appendix C of this manual.

2.2 Running the Install program

An install program is provided on CD-ROM. This software copies all of the necessary files onto your hard drive. We suggest that you use the following directories to hold the programmer software:

```
C:\DATAIO\CWGANG
```

Where C: is the name of your hard drive

So, if you are installing the software onto the C: drive, to run the installation program place the CD-ROM in the drive and log onto this drive by typing **E:** or **F:** as necessary. Now type:

```
INSTALL C:\DATAIO\CWGANG <ENTER>
```

After installation you should add this directory to your current PATH statement in your AUTOEXEC.BAT file. For example change

```
PATH=C:\DOS;C:\BAT;C:\UTILS
```

to

```
PATH=C:\DOS;C:\BAT;C:\UTILS;C:\DATAIO\CWGANG
```

Then reset the PC.

3. Connecting the Programmer

3.1 Computer requirements

The programmer will operate with any IBM compatible PC (386 and above recommended) with a standard IBM CENTRONICS interface (parallel printer port). This means that it is very easy to move between PCs. Users who prefer to have an internal card or who do not wish to use their existing parallel port can fit an additional parallel card (available from any PC supplier). Alternatively, a manual switch box may be used.

Please note that the programmer is designed to use the Standard Printer Port configuration (SPP). Users with port settings in their BIOS, or on Enhanced parallel port cards, should NOT select the EPP or ECP options.

The software is compatible with MS-DOS v.3.0 or later, and with MS-Windows, v.3.1 or Windows 95.

3.2 Connection to the PC

STEP 1 - Connect one end of the parallel cable to the programmer. The other end of the cable should connect to the PC. The cable will only fit the DB25 Parallel port. Tighten the holding screws thus ensuring that the cable will not cause programming problems.

STEP 2 - Connect the power supply provided to the programmer and plug into any standard electrical outlet.

USE ONLY THE POWER SUPPLY ADAPTOR PROVIDED

Replacements and adapters for different countries (240V, 220V, 100V, 120V) can be supplied by Data I/O or their distributors.

STEP 3 - Switch on the PC.

STEP 4 - Run the DETECT program.

3.3 Running the DETECT program

The ChipWriter has been supplied with selftesting software which can check that the programmer has been installed correctly. Before using the programmer for the first time, or whenever moving it to a new PC, run the DETECT program. This has two purposes :

- It confirms which parallel port is being used by the programmer.
- It checks the programmer hardware.

WARNING :

Always ensure that there are no devices in the programmer before running Detect, as leaving a device in the socket could damage the chip and give false results

To run the program, connect the programmer to the PC and mains supply, remove any devices and type:

DETECT <ENTER>

at the DOS prompt.

The DETECT will report the programmer's firmware version number and run a communications test. It will

then run through a series of memory checks and will report any faults found. The program will report which port the programmer is connected to. This information can then be used to set up the parallel port in the main CWGANG menu software.

If any error is reported, then you will need to contact Data I/O Technical Support or your distributor. In order to assist with your problem, it is possible to obtain a printout of the Detect results by using:

```
DETECT >PRN
```

where PRN is the name of the resident printer device (i.e.. LPT1).

```
or          DETECT >ERROR.LST
```

```
followed by PRINT ERROR.LST
```

See Appendix C for Technical Support information

3.4 Programming non-DIP devices

Devices in packaging other than DIP require socket adapters. A range of adapters is available from Data I/O for most common devices. If you are unsure as to which adapter to use, please contact Data I/O or your distributor for advice.

Adapters from other sources which are simple pin conversions can usually be used successfully. Where devices require non-standard adapters to be supported, these are available from Data I/O.

All adapters available from Data I/O are standard across the ChipWriter range.

3.5 Adapter usage

Insert the adapter into the ZIF socket of the programmer with the bottom end of the adapter in the bottom pins of the socket. The text on the adapter boards should be the right way up. Lock the ZIF socket handle in place. For PLCC adapters, the notched

corner of the adapter socket will usually be in the top left hand corner. Devices are placed in the socket in the 'live-bug' position, i.e. with legs downwards for all sockets 28-pins and above. 20-pin sockets are "dead-bug" loading. Push the device down to lock in place, then push down the socket sides to release.

See figure 3.1 below:

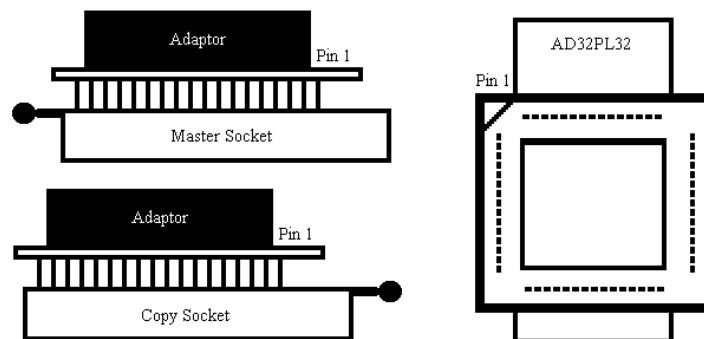


Figure 3.1 - Adapter Usage

4. Getting Started

We have provided some quickstart tutorials to help you start using the programmer as quickly as possible. If you have not used a programmer before, these may help you to begin. However, these should not be taken as a comprehensive guide to the programmer's capabilities. The following tutorials are provided:

TUTORIAL 1. Programming from a master device.

TUTORIAL 2. Programming from a file on disk.

TUTORIAL 3. Programming in Stand-Alone mode.

These tutorials are provided to get you using the programmer quickly for simple tasks. However, we recommend that you read the reference sections of the manual if you have any difficulty and before attempting any more complex operations.

The quickstart tutorials use the menu driven software. For information on how to use the batch (DOS command) software, see section 8.

4.1 Before you start

If you are not sure how to select menu items please see section 5.3 for more information .

Figure 4.1 below illustrates the orientation of devices in the programmer: The chip inserted in the MASTER socket is located at the bottom of the ZIF socket with the bottom pins next to the ZIF socket handle. Devices inserted in any of the eight COPY sockets should be located at the bottom of the ZIF socket with the bottom pins furthest away from the ZIF socket handle.

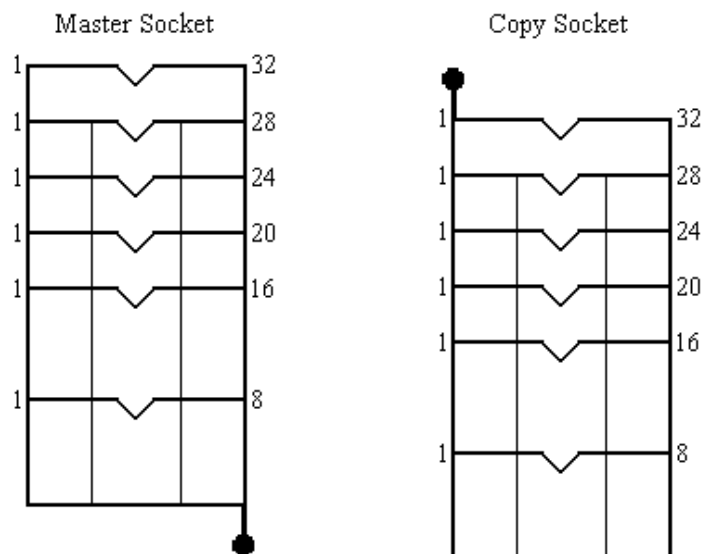


Figure 4.1 - Device Location

If you are using a socket adapter, please see the instructions with the adapter for how to place the device in the socket.

4.2 Quickstart tutorials

TUTORIAL 1:

Programming an EPROM from a master device.

1. Type CWGANG <ENTER>
2. Select the device you are using as a master, using DEVICE/MANUFACTURER. Select the manufacturer and press <ENTER>. You will then be given a list of different types of devices. Select the type and press <ENTER>. You will then be prompted for the device name. Choose the correct device name from the menu you are given and press <ENTER> to accept. The details of the device you have chosen will now be displayed at the bottom of the screen.
3. Ensure that the data memory buffer size is large enough to hold the data to be read from the master device. You can change the buffer size using DATA/BUFFER-SIZE. See section 6.2.1 for more information.
4. Insert the master device into any of the sockets of the programmer.
5. Select PROCESS/READ to read the contents of the chip into the buffer. Press <F10> to accept the default addresses.
6. Select the device you are copying to if it is different (see 2.)
7. Place the copy device(s) into the eight copy sockets only. The MASTER socket is Read-only
8. Select PROCESS/PROGRAM. Press <F10> to accept the default addresses.
9. Appropriate messages will be displayed at the end of programming, showing which devices have been programmed correctly, which sockets are empty etc.

TUTORIAL 2:

Programming an EPROM from a file on disk

1. Type CWGANG <ENTER>
2. Select the device you are wanting to program using DEVICE/MANUFACTURER. Select the manufacturer and press <ENTER>. You will then be given a list of different types of devices. Select the type and press <ENTER>. You will then be prompted for the device name. Choose the correct device name from the menu you are given and press <ENTER> to accept. The details of the device you have chosen will now be displayed at the bottom of the screen.
3. Ensure that the data memory buffer size is large enough to hold the data to be read from the master device. You can change the buffer size using DATA/BUFFER-SIZE. See section 6.2.1 for more information.
4. Select FILE/OPEN
5. Specify the path and filename where your data is located.
6. Select the file format used (HEX auto recognition will detect any of the HEX formats available)
7. Press <F10> to accept the default settings and load the data. A checksum will be displayed on the screen.
8. Place the copy device(s) into the eight copy sockets only. The MASTER socket is Read-only
9. Select PROCESS/PROGRAM. Press <F10> to accept the default addresses.
10. Appropriate messages will be displayed at the end of programming, showing which devices have been programmed correctly, which sockets are empty etc.

TUTORIAL 3:

Programming in Stand Alone Mode

The programmer can be configured to work in stand alone mode in two ways:

Using the internal EEPROM OR Using Configuration EPROM(s)

Both methods require a DATA Master EPROM programmed with the required copy data. See the above tutorials.

Setting up the Internal EEPROM

1. Type CWGANG <ENTER>
2. Select the device you are wanting to program using DEVICE/MANUFACTURER. Select the manufacturer and press <ENTER>. You will then be given a list of different types of devices. Select the type and press <ENTER>. You will then be prompted for the device name. Choose the correct device name from the menu you are given and press <ENTER> to accept. The details of the device you have chosen will now be displayed at the bottom of the screen.
3. From the MODE menu, select the option: SET UP INTERNAL EEPROM. The programmer is now configured for the selected device. A power on code is displayed (see section 7.6 for full information).

Setting up a Configuration EPROM (The configuration EPROM needs to be a standard 27C512 device)

1. Type CWGANG <ENTER>
2. Select the device you will be programming in Stand Alone Mode, using DEVICE/MANUFACTURER. Select the manufacturer and press <ENTER>. You will then be given a list of different types of devices. Select the type and press <ENTER>. You will then be prompted for the device name. Choose the correct device name from the menu you are given and press <ENTER> to accept. The details of the device you have chosen will now be displayed at the bottom of the screen.
3. Use DATA/FILL to ensure the buffer is filled with FF.
4. From the MODE menu, select the option: SET UP MASTER EPROM.
5. The on screen memory area now contains the required programming information for the selected device type.
6. Using the same method as in (2.), select the 27C512 device you will be using as the CONFIGURATION master.
7. Place the device into one of the 8 copy sockets of the programmer.
8. Select PROCESS/PROGRAM. Press <F10> to accept the default addresses.
9. Appropriate messages will be displayed at the end of programming, showing that the device has been programmed correctly, and which sockets are empty etc.

Now that the configuration is complete, switch off the programmer and remove the parallel cable. If you are using a Configuration EPROM, place the device in the MASTER socket. If you are using the Internal EEPROM, ensure that the Master socket is empty. When the GLV-32 is powered up, it waits to see if there is a message from the PC. If no message is received, it looks for a 27C512 CONFIGURATION EPROM in the Master socket. If no device is present, it looks in the internal EEPROM. If no data is found, or if there is a problem locating the Configuration EPROM, the Master LED will flash three times in succession and will continue this error message until the machine is reset.

Once the machine has found the correct device parameters, LED's 0-7 will display a code byte indicating which type of device it is configured for. See section 7.6 for more information. Once the code has been checked, clear the display by pressing reset.

To program devices:

Place the DATA Master EPROM in the master socket, then place the blank devices in sockets 0-7. Press START to begin programming. While programming is taking place, the indicator LED's will light for each socket in which there is a device; the Master LED will flash. During the VERIFY stage, the master LED will remain on permanently. If programming is successful, a "walking" sequence will appear on the LED's. If unsuccessful, the LED's will flash to indicate which sockets have failed.

5. Using the Programmer Software

5.1 Basic Software Configuration

The operating software contains the following program:

CWGANG.EXE : Used to program EPROM's, EPROM's and Flash PROMS.

These are menu driven programs which contain all the features required to program and test devices. The actual programming information for individual devices is held in library files, which allows easy updating of the software to accommodate new devices and algorithm updates.

Command line Batch software is also included which allows the programmer to be operated from DOS batch files. Details of the commands available are shown in section 8.

On first installing the software and programmer on a PC, remember to alter the parallel port selection within CWGANG . EXE by selecting OPTIONS/PARALLEL PORT NO .

5.2 Screen layout

Each program displays information as follows :

Menu headings are displayed across the top of the screen.

Device programming details are displayed in the bottom window.

Data memory contents are displayed in the main window.

The software and library version numbers are displayed at the top of the screen, just below the Menu bar. These are useful to note should you need to contact Technical Support or to check for new version numbers on the BBS or Internet.

5.3 General menu selection

The software has been designed to make the programmer as simple as possible to use. All functions are contained within the menu system. The menus are obtained by pressing the <ESCAPE> key. The user can then move around the menus by pressing the cursor keys. To select an option press <ENTER>.

Each menu function is available by pressing the first letter of the function. For instance, the DEVICE pull-down menu is available by pressing the 'D' key. Some options can be selected by pressing **ALT+letter**. These are indicated by a triangle next to a letter on the menu display. For example, Program can be selected by pressing ALT+P. A full list of shorthand keystrokes is given in Appendix 1.

5.4 Dialogue boxes

The menu system makes extensive use of dialogue boxes. Within these boxes the following prompt types may appear :

Prompt	:	requires you to type in some information
	>	will display a series of options when you press <ENTER>.

Inside the dialogue boxes use the cursor keys to move between options; use <ENTER> to select the contents of the dialogue box. Use <F10> to accept all chosen options within the window. If you wish to abort from the window, press <ESCAPE>.

5.5 Changing the settings

Before starting, you may wish to change any of the following. They are found in the OPTIONS menu. This will have to be done for both EPROM and PAL :

OPTIONS/PARALLEL PORT NO

Selects parallel port to be used. This is detected by the DETECT utility which should be run when the unit has been connected to a PC for the first time. Once the parallel port has been chosen it will be remembered when next using the programmer.

OPTIONS/COLOR MODE

To choose whether the display is MONO or COLOR

OPTIONS/SNOW PRECAUTIONS

Usually set to OFF but set it to ON if you have problems with snow on your monitor. (affects older PC's with CGA)

OPTIONS/HI-RES MODE

Select ON for 40/50 line mode. Effective only with EGA and VGA monitors.

5.6 Using a mouse

The software can be mouse driven. However your mouse driver software must previously have been installed. This is normally done automatically in your CONFIG.SYS file (by loading a device driver such as MOUSE.SYS) or in your AUTOEXEC.BAT file (by invoking a mouse driver program such as MOUSE.COM). If you are unsure how to do this please refer to your system manual or the manual supplied with your mouse. CWGANG.EXE automatically enables the mouse if one is installed (see later if you wish to disable it).

5.6.1 Left mouse button (LMB)

The LMB is used to select items in a list, such as a menu selection or a device name from the device list. Simply move the pointer to the required item and press the LMB. If the item is a menu heading, such as FILE, PROCESS etc., then this will open up that particular menu. If the item is a sub-menu or part of a list this will highlight the chosen item. To select the highlighted item Double Click on the LMB. Double Clicking performs the same action as pressing <ENTER>.

5.6.2 Right mouse button (RMB)

Pressing the RMB is exactly the same as pressing <ESC>.

5.6.3 Middle mouse button (MMB)

Pressing the MMB is exactly the same as pressing <F10>. When this key is a possible input, the window being displayed will have "F10" in the bottom border. Pointing at the text "F10" and pressing the LMB will also produce the same result (for two button mouse users).

5.6.4 Disabling the mouse

To disable the use of the mouse invoke each of the programs with the /NM option. e.g.

```
CWGANG /NM
```

5.7 Edit and view modes and scroll Bars

In order to observe data in the data buffer area without risking modification a VIEW mode is available in the DATA menu. This will only allow scrolling, paging etc. Mouse users can enter this mode directly by pointing at the buffer window and pressing the LMB. In order to make moving around the buffer much easier, Scroll Bars have been added to the window borders.

5.8 Running the software under Windows

The CWGANG software can be run in a DOS window under Windows 3.1x or Windows 95. Example .PIF and .ICO files are included. To create an ICON for the ChipWriter - CWGANG.EXE, for example on Windows 3.1, do the following:

- Select NEW PROGRAM ITEM in the Program Manager FILES/NEW menu.
- Change Description to CWGANG
- Change Command Line to:
C:\DATAIO\CWGANG\CWGANG.PIF
- Change Working Directory to C:\DATAIO\CWGANG
- Select CHANGE ICON and change the file in the CHANGE ICON prompt to
C:\DATAIO\CWGANG\EPROM.ICO
or C:\DATAIO\CWGANG\EPROM1.ICO
- Select OK

Use Optional Parameters if required. The mouse functions should work in a DOS window in 386 enhanced mode in Windows 3.1x or Windows 95, but if you have trouble see section 10 of the Windows readme file.

5.9 Using the programmer software over a network

It is possible to run the programmer software from a network hard drive with the programmer itself connected to the local computer.

Some networks do not pass the full execution path to the .EXE file. The CWGANG program therefore has difficulty in locating the library and device files. To circumvent this problem use the following environment variables to set the full network path of the executable;

For ChipWriter Gang:

```
SET CWGANG=[PATHNAME]
```

where PATHNAME is the full network path where CWGANG.EXE is located.

For example, if the programmer software is held in

```
Z:\CWGANG
```

Then the statement

```
set CWGANG=Z:\CWGANG
```

should be added to the AUTOEXEC.BAT file of the computer where the programmer is connected.

5.10 Using EMS software

The programmer is supplied with EMS software which will enable expanded memory to be accessed if it is available. This requires an expanded memory manager compatible with EMS 3.2 standard. It allows up to 8 Mb of memory and will run on any PC with an expanded memory board or a 286 or 386 with an expanded memory manager. The software will not give the EMS option if a driver such as EMM386.EXE is not installed correctly.

If the EMS function is working correctly, it will be possible to increase the data memory buffer size to 1024K without problem. If this is not possible, and the maximum it can be set to is 384K for instance, then the EMM386 driver has not been correctly installed. Check the CONFIG.SYS file in the root directory on your hard drive to check that the following line exists:

```
DEVICE=C:\DOS\EMM386.EXE
```

Also ensure that the **NOEMS** option has **not** been included on this line. DOS maybe replaced by WINDOWS if you are running under Windows 95. It is also possible to run MEMMAKER and select YES when prompted for EMS memory requirements.

5.11 Saving software settings

When exiting the programmer software, settings such as parallel port, the device chosen, etc. are automatically saved to avoid having to repeatedly set up the chosen options.

6. Memory Programming

This section covers the all the memory type devices which are programmed using CWGANG.EXE.

To use this program, at the DOS prompt, type

```
CWGANG <CR>
```

6.1 Device selection

In order to tell the programmer software which programming algorithm to use, it is necessary first to select the device which is being used.

6.1.1 To select a device

To do this, enter CWGANG.EXE and select

```
DEVICE/MANUFACTURER
```

A list of supported manufacturers will be displayed on the screen. Use the up and down cursor keys to find the manufacturer of the device you are using. You can also press the initial letter of the manufacturer you need, e.g. pressing <M> will take the cursor down to MACRONIX. When the cursor is highlighting the required manufacturer, press <ENTER>.

You will then be given a choice of device types to select. Highlight the type required or press the first letter, and press <ENTER>.

The software then displays a list of all the supported devices of the manufacturer and type selected. Highlight the required device name and press <ENTER> again.

The details of the selected device will then be displayed in the bottom device of the screen.

6.1.2 Changing the device selected

Once a device has been selected, to select another device of the same manufacturer and type, press

DEVICE/DEVICE NAME

To select another device type within the same manufacturer, select

DEVICE/TYPE

6.1.3 Which device?

Generally it is necessary to select the device name which is as close as possible to that stamped on the chip you are using. The software uses the correct nomenclature according to the manufacturer's data sheet. Speed and package indicators are only required when they affect programming algorithms.

6.1.4 Low Voltage devices

Low voltage devices are selected exactly as other devices. In the device information in the bottom window, you will notice that the Vcc voltages during program and verify (VccP & VccV respectively) will be a combination of 5V and/or 3.3V. As most programmers are still unable to provide 3.3V circuits, manufacturers still allow the devices to be programmed at 5V. The ChipWriter programmers can not only verify a low voltage device at its nominal operating voltage, they can also program with voltages down as low as 3.3V as well, giving much greater flexibility.

6.2 Opening data files

Files can be loaded from disk into the data buffer and saved to disk after editing using the OPEN and SAVE options in the FILE menu. See section 6.2.8 for more details on saving files.

6.2.1 Setting the data memory buffer size

It is necessary to ensure that the data memory area has been set to a size large enough to accept the contents of the file to be loaded. Do this using

DATA/BUFFER SIZE

Enter the required size in Kbytes. The maximum buffer size available is determined by the available memory of the PC. The programmer software uses approximately 150K plus the chosen buffer size. The EMS software will use expanded memory for the buffer before using conventional DOS memory. If the buffer cannot be increased sufficiently see section 5.10 for information on installing EMS.

6.2.2 Opening A file

Select

FILE/OPEN

This then displays a number of different options:

FILE NAME : Enter the required filename

FILE TYPE : Select the required file format (see section 6.2.3)

FILE WINDOW LOW ADDRESS : Enter the required file start address. Default = 0

FILE WINDOW HIGH ADDRESS : Enter the required file end address. Default = FFFFFFFF

FILE INCREMENT : Select from Every Byte, Every 2nd Byte etc. up to Every 8th Byte

BUFFER START ADDRESS : Enter the required address. Default = 0

Pressing <F10> accepts the parameters and opens the selected file, displaying the checksum when loaded (See 6.2.5).

6.2.3 File format

The file type can be selected by pressing <ENTER> to see the options available. Choose from:

```
RAW/BINARY
ASCII-SPACE-HEX
ASCII-OCTAL
HEX-AUTO RECOGNITION
HEX-MOTOROLA S1, S2 OR S3 RECORDS
HEX-INTEL MCS86
HEX-TECTRONIX
HEX-EXTENDED TECTRONIX
HEX-TI TAG
```

Choosing HEX-AUTO RECOGNITION is generally the easiest way of loading a HEX formatted file. However some linkers put additional headers at the start of the HEX file which can confuse the loader when using this option. Hence the other options.

ASCII-SPACE-HEX embodies the formats of ASCII space HEX, ASCII ' HEX etc. The software assumes that the file consists of HEX digits separated by non HEX characters. Checksum information is not looked for and no separators are needed.

e.g.. If a file consisted of A BCD EF this would appear as

Address 0	1	2	3
Data 0A	BC	0D	EF

6.2.4 Handling large files and devices

In order to facilitate the programming of memory devices and the loading of files whose size is greater than the maximum buffer size the FILE/OPEN command (ALT-O) has the following parameters:

```
FILE WINDOW LOW ADDRESS (FWLA)
```

FILE WINDOW HIGH ADDRESS (FWHA)
 FILE INCREMENT (FINC)
 BUFFER START ADDRESS (BADD)

A byte is ONLY loaded into the buffer if its FILE ADDRESS is inside the window limits defined by the above (inclusive) range. Data at FWLA is loaded into the buffer at BUFFER START ADDRESS. The FILE INCREMENT works within this window. See Figure 6.1 below.

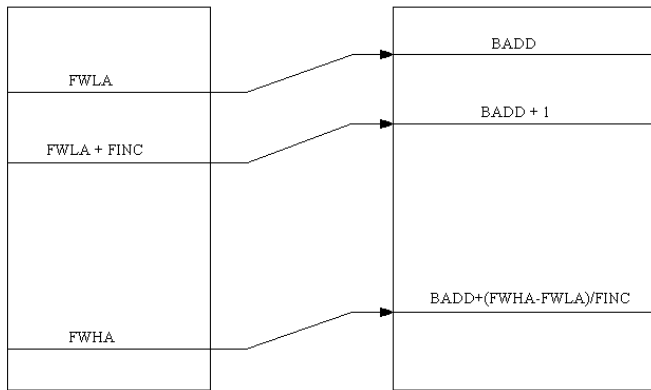


Figure 6.1 - Handling Large Files

If you have file loading problems, check the above parameters. Normally FWLA=0 and FWHA=FFFFFFFF (i.e. the whole file is to be loaded). One point to note is that these parameters are stored each time the software is quit. If in a previous session FWLA was changed to 10000 (HEX), then files with information in the range 0-FFFF will appear empty. Therefore, it is important to check all the load parameters each time a file is opened.

6.2.5 File checksum

When a file has been opened and loaded into the data buffer, a checksum is calculated on the contents of the file and displayed on the screen.

This is displayed as three figures:

- The sum of unsigned bytes with the carry being added to the MSB of the checksum.
- The complement of (a)
- The arithmetic negative of (a)

NOTE: The checksum displayed on FILE/OPEN is the checksum of the FILE only, and does not necessarily include the whole address range of the selected device. A checksum of the device, or of a particular address range, can be calculated using DATA/CHECKSUM as explained in 6.3.5.

6.2.6 Wild card loading

Wild cards may be used in the filename to perform drive/directory searches. Wild cards consist of * and ?.

- * matches to any number of characters
- ? matches to any one character

A list of matching filenames is displayed alphabetically (Sub-Directories first) and the user may select using the cursor keys or the first letter of the name.

6.2.7 Saving a file to disk

Data edited in the buffer area or read in from a device can be saved to disk in a number of different formats. It is recommended that frequently used files are saved in binary mode to save disk space and reduce loading time. Use:

FILE/SAVE AS

This saves a user defined address range from the data memory area, in any of the following formats:

RAW/BINARY
ASCII-SPACE-HEX
ASCII-OCTAL
HEX-MOTOROLA (S2 format)
HEX-INTEL

To save any modifications to the currently loaded file, simply select:

FILE/SAVE

6.3. Using the editing facilities

As well as loading from disk or reading a pre-programmed device, data to be programmed can be entered directly into the buffer from the keyboard. In addition, files can be amended using the DATA menu options.

6.3.1 Entering EDIT mode

To edit the contents of the data buffer, select

DATA/EDIT

This allows you to type in HEX or OCTAL code, or ASCII characters. Once EDIT is selected, the cursor will move to the HEX display in the buffer window. Use the <TAB> key to switch between HEX and ASCII. In HEX mode, only keys 0-9 and A-F are allowed. In OCTAL mode only keys 0 to 7 are allowed. The cursor keys can be used to move around the buffer.

6.3.2 Viewing without making changes

If you wish to view the buffer contents beyond the first screen without danger of making unintentional changes to the data, use

DATA/VIEW

This allows you to view the data contents as HEX or OCTAL code or ASCII characters. Once VIEW is selected, the cursor will move to the HEX display in the buffer window. Use the <TAB> key to switch between HEX and ASCII.

6.3.3 Quickly moving around the data

The <HOME> and <END> keys allow you to move more quickly. Press once to move to the beginning (end) of the line, twice to move to the beginning (end) of the page or three times to move to the beginning (end) of the file.

When you have finished editing or viewing the buffer, press <ESC> to return to the main menu.

6.3.4 Quick buffer editing commands

DATA/FILL

Allows you to fill a defined area of the data area with information in byte, word or long word format. Enter the buffer start and end addresses and the information to be inserted. Use less than 3 characters for a byte, 3 or 4 for a word, more than 4 for a long word. Press <F10> to accept.

DATA/COPY

Copies a defined block of buffer memory to a new location.

DATA/SEARCH

Searches for data in HEX, OCTAL or ASCII format. Use the Up/Down arrow in the dialogue box to move between HEX, OCTAL and ASCII data entry.

DATA/JUMP TO

Jumps to a chosen address in the data area

DATA/SWAP BYTES

Swaps odd and even bytes within a defined range.

6.3.5 Buffer checksum

The command:

DATA/CHECKSUM

performs a checksum on the contents of a user defined range in the buffer. Select

BUFFER START ADDRESS : default to 0

BUFFER END ADDRESS : default to end of buffer

The checksum is displayed as three figures:

- a) The sum of unsigned bytes in the memory range with the carry being added to the MSB of the checksum.
- b) The complement of (a)
- c) The arithmetic negative of (a)

NOTE: The memory area can be larger than the size of the opened data file. If there is a discrepancy between the expected and calculated checksums, recalculate specifying the start and end address of the file.

6.3.6 Buffer display modes

The contents of the buffer can be displayed in 8-bit HEX, 8-bit OCTAL, 12-bit (16-bit storage, LSB = low address) or 16-bit mode (LSB = low address). To change the display, select:

OPTIONS/DISPLAY MODE

and select the display mode required.

6.3.7 Printing the data contents

A print-out of all or part of the data memory contents can be obtained by selecting:

DATA/PRINT

Specify the start and end addresses of the block to be printed, and the name of the output device (e.g. LPT1).

6.4 Reading and checking devices

This section explains how to read the contents of a programmed device, verify the contents against known data, check the device signature and identify a device from its signature, and perform illegal bit checks and blank checks on devices.

6.4.1 Reading a device

To read the contents of a device, select the correct device name from the `DEVICE` menu as described in section 6.1 and place the device to be read in the programmer socket.

Please note that some devices CANNOT be read, either because the chip type itself is designed not to be read for security purposes, or because the on-chip security features have been enabled when programming.

Before reading EPROM's, ensure that the buffer size is set large enough to contain the data (see section 6.2.1 on setting the buffer size). **NOTE:**

To read the contents, or part of the contents, of a device, select

`PROCESS/READ`

The programmer software then quickly accesses the library file for the programming algorithm required. You are then required to select the socket you wish to read from, 0 - 7, or the Master socket. A dialogue box then appears requesting Device Start and End address, and Buffer start address:

`DEVICE START ADDRESS :`

Enter the first address from which you wish to start reading data from the device. Default = 0.

`DEVICE END ADDRESS :`

Enter the last address from which you wish to read data from the device. Defaults to the highest address available for the selected device.

`BUFFER START BYTE ADDRESS :`

Enter the buffer address where you wish to begin reading the data into. Default = 0.

`BUFFER INCREMENT :`

Default = Every byte.

Press <F10> to accept the options displayed and the device will then be read. The contents of the device will then be visible in the main data buffer, and a checksum will be displayed.

Once the contents of a device have been read, they can be edited using the `DATA/EDIT` option, and then saved to disk, or programmed into another device. See section 6.3 for more detail on editing data.

6.4.2 Verifying the contents of a device against a file

To verify the contents of the selected device(s) against known data, first ensure that the buffer size is large enough to handle the file. Then open the file against which you need to verify the data. Place the device(s) to be verified in the programmer and select:

`PROCESS/VERIFY`

If the data matches, the message `VERIFY OK` will be displayed. If there is a mismatch, the message will indicate the first address which has failed to verify.

Note: The `VERIFY` operation is only performed on the `COPY` sockets. It is NOT possible to verify a device in the Master socket.

6.4.3 Verifying the device type

Many newer devices contain Electronic Signature codes which enable you to electronically identify the device. It is possible to verify a known device type against this ID code. The ID code is also used by your programmer to check that the correct device type has been selected before proceeding with any process that may damage a chip. However, not all devices contain this ID code, and the checks for electronic signature may themselves damage a device without a signature, or one of a different number of pins from that expected. This means that these checks cannot be foolproof and need to be used with some caution.

To verify the signature of a device, select

PROCESS/VERIFY ELECTRONIC ID

If the database contains a manufacturer's code and device code they can be verified against that of the device.

6.4.4 Checking if a device is blank

In order to program a device, it must either be fully blank, or be capable of containing the information to be programmed into it in addition to the information it already contains.

In order to test whether a device is completely unprogrammed, select

PROCESS/BLANK CHECK

This option checks if the device is blank (all 1's or all 0's depending on the actual part).

NOTE: It is possible for a device to appear blank even though it is not fully erased. If problems are encountered in programming a device, try erasing it for a longer period and then reprogramming.

6.5 Programming and erasing devices

IMPORTANT NOTE

Data I/O use only the specified programming algorithms as provided by the semiconductor vendors. While every effort is made to ensure that programming algorithms are correct, the said algorithms are not guaranteed by the manufacturers and therefore cannot be guaranteed by Data I/O.

Users are advised to ensure that devices work correctly in their systems before programming large quantities.

6.5.1 Programming devices

To program the selected device, use

PROCESS/PROGRAM

This performs a blank check, chip erase (for electrically erasable devices), program and verify. After selecting the PROGRAM option, a dialogue box is displayed which enables the required Programming Parameters to be selected.

The most basic configuration will offer the following options:

DEVICE START ADDRESS : Enter the address where you wish the programming operation to start. Defaults to 0.

DEVICE END ADDRESS : Enter the last address of the device you need programming. Defaults to the device end address.

BUFFER START ADDRESS : Enter the address of the first byte in the buffer from where you want to start programming. Default is 0.

BUFFER INCREMENT : Allows you to program every byte, every second byte, every 4th byte or every 8th

byte. Default is Every Byte.

To accept the address selections or defaults, press <F10>.

The programmer will first check the device selection and position, then perform a blank check, and then proceed with programming.

While a device is being programmed, a message will be displayed in the centre of the screen showing the address which has been reached. Following programming, an automatic verify will be conducted once completed, a programming status report will display the status of each socket.

6.5.2 Single key programming

If you need to program more than one device in exactly the same way, simply press <F10> to repeat the process. You are given the option to change any of the address settings before proceeding again by pressing <F10>.

6.5.3 Gang programming

The ChipWriter Gang is designed to program up to 8 devices at a time. From the menu bar, select MODE/GANG MODE. This allows the user to program 1 to 8 devices placed in sockets 0 to 7 with the same information. It is not necessary to specify how many sockets are being used or which sockets they are. All of the functions in the PROCESS menu, other than READ, can be performed simultaneously on multiple devices once the gang mode has been selected. The PC displays the status of each socket during each function.

6.5.4 Set programming

The ChipWriter Gang can program in 16-bit or 32-bit Set modes.

For 16-bit mode, select MODE/SETS OF 2. The programmer will send information into 4 sets of 2 devices. Sockets 0 to 3 will contain the LSB or the first address specified from the buffer, and then every second byte in the buffer. This maybe ODD or EVEN addresses, depending on the given Start address. Sockets 4 to 7 will contain the MSB or the second specified address, and then every second byte in the buffer.

For 32-bit mode, select MODE/SETS OF 4. The programmer will send information into 2 sets of 4 devices. Sockets 0 and 1 will contain the LSB or the first byte specified from the buffer, and then every fourth byte. This maybe ODD or EVEN addresses, depending on the given Start address. Sockets 2 and 3 will contain the second byte and every fourth byte from there, sockets 4 and 5 will contain the third byte, and every fourth byte from there. Sockets 6 and 7 will contain the MSB or the fourth address, and then every fourth byte in the buffer.

6.5.5 Erasing devices

For Flash memory, and some parallel EEPROM parts which support a device erase feature, use

PROCESS/ERASE

This electrically erases the data held on the chip.

To erase all other EPROM's, as well as NVRAM's, fill the buffer with #FF (Hx) and program the whole device.

6.6. Adding user defined devices

If you need to program a device that is not on the list, please contact Data I/O or your distributor to find out if it is available on the latest software. For EPROM/EEPROM devices that have not yet been implemented, there is

the facility to add user-defined devices.

The user database can also be used to keep frequently used devices from different manufacturers in one place, rather than having to select these devices via the menu system every time. To do this, select the device required, then select `DEVICE/ADD TO USER DATABASE`. The device name could be changed to a project name if required using `DEVICE/MODIFY PARAMETERS` before adding to the user database.

Note: This feature is for the convenience of advanced users only. Data I/O Corp. cannot be held responsible for incorrect programming using the user database.

The information needed to add your own devices will be found in the relevant data sheet. Please contact the device manufacturer for up to date information.

Please Note

- User defined programming information may not work for Flash PROM's since not all of these have standard programming algorithms.
- In order to protect the manufacturer recommended algorithms, it is not possible for the user to change the manufacturers' database.

The procedure for adding user defined devices is as follows:

1. Select a device which is similar to that to be added from the devices menu. This will minimise the number of changes which it is necessary to make, and will ensure that the relevant options are displayed. Have ready the device data sheet.
2. Select `DEVICE/MODIFY PARAMETERS`. This will access the User database, thus protecting the manufacturer database from unintentional changes.
3. Modify the required parameters:

Note: Always modify parameters in a top to bottom order

`DEVICE NAME` : can be up to 16 characters
`DEVICE CODE` : Electronic signature on chip
`MANUFACTURER CODE` : Electronic signature on chip
(If the ID codes are left blank, then the Electronic Signature check will be disabled)
`VccP` : Vcc during programming. Enter the desired voltage
`Vpp` : Programming voltage. Enter the desired voltage
`Vcc(During Verify)` : Select from given options
`ORGANISATION` : Select from given options
`LAYOUT` : Select from given options
`PULSE TYPE` : Select from given options
`PULSE WIDTH(µS)` : Typical values are 100 for the latest EPROM's, 1000 for older 2764 to 27010s. 50000 for 2716 and 2732.
`MAX NUM OF TRIES` : Typical values are 1, 25.
`OVER PROGRAM` : Enter over-program pulse width.
`OverProg Pulse` : select variable or fixed. Generally N/A or variable.

4. Once device parameters have been modified for the user database, select `DEVICE/ADD TO USER DATABASE` to add them to the database on disk.

7. Stand Alone Mode

The ChipWriter Gang can be configured to be used away from a PC. This allows the programming specifications to be completely controlled away from the production area, thus reducing the margin for operator error.

The Stand-Alone mode can work in one of two ways. The programmer contains a serial EEPROM which can be set up with the specifications for the device to be programmed. The data to be copied can then be programmed into a master device. Exact copies can then be made from the Master socket into any of the Copy sockets 0 to 7. This would be the normal method for copying EPROM's or EPROM's. The device settings remain stored when the power is turned off.

The second method is to download the device parameters into any 27C512 EPROM. The programmer uses this to store the programming configuration data and algorithms. This method would typically be used if it was desired to set up the programmer for more than one type of EPROM without having to go back to the PC. In this case two EPROM's are needed. A 27C512 for configuration information, and a DATA MASTER EPROM for holding the user data. The DATA MASTER will be the same type as that to be programmed.

7.1 Setting up the internal EEPROM

Select the device you are wanting to program from the `DEVICE` menu as described in section 6.1. From the `MODE` menu, select the option: `SET UP INTERNAL EEPROM`. The programmer is now configured for the selected device. A power on code is displayed (see section 7.4).

7.2 Setting up a Master configuration EPROM

Ensure that the device to be used as the configuration EPROM is a standard 27C512 device. Select the device you are wanting to program from the `DEVICE` menu as described in section 6.1. Use `DATA/FILL` to ensure the buffer is filled with `FF`. From the `MODE` menu, select the option: `SET UP MASTER EPROM`. The on screen memory area now contains the required programming information for the selected device type. Now select the 27C512 device being used for the `CONFIGURATION` master and program in the same way as described in 6.5. Only sockets 0 to 7 can be used for programming as the Master socket is read-only.

7.3 Operating in stand-alone mode

Once the configuration is complete, switch off the programmer. If you are using a Configuration EPROM, place the device in the Master socket. If you are using the Internal EEPROM, ensure that the Master socket is empty. When the programmer is powered up, it waits to see if there is a message from the PC. If no message is received, it looks for a 27C512 EPROM in the Master socket. If no device is present, it looks in the internal EEPROM. If no data is found, or if there is a problem locating the Configuration EPROM, the Master LED will flash three times in succession and will continue this error message until the machine is reset.

Once the machine has found the correct device parameters, LED'S 0-7 will display a code byte indicating which type of device it has been configured for. See section 7.6 for more information. Once the code has been checked, clear the display by pressing `RESET`.

7.3.1 Programming in stand-alone mode

Place the DATA Master EPROM in the master socket, then place the blank devices in sockets 0-7. Press `START` to begin programming. While programming is taking place, the indicator LED's will light for each socket in which there is a device; the Master LED will flash. During the `VERIFY` stage, the Master LED will remain on permanently. If programming is successful, a "walking" sequence will appear on the LED's (LED's 0,4 1,5 2,6 3,7). If unsuccessful, the LED's will flash to indicate which sockets have failed.

NOTE: Flash memory devices cannot be programmed in stand-alone mode. Flash devices require a sequence of bytes to be written to them in order to initialise the Read operation. As the Master socket is READ only, in order to eliminate any possibility of the Data Master device being corrupted, it is not possible to write any initialisation codes to the Master socket.

7.3.2 Verifying in stand-alone mode

Devices are automatically verified after programming. If it is wished to verify them separately, press and hold RESET, then press START while still keeping RESET held down; then release START, then release RESET. For any device which fails to verify, the associated LED will flash. Press RESET to clear the error.

7.4 LED Codes

The eight Copy socket LED's and the Master socket LED represent a different programmer status at different times.

7.4.1 LED status

Flashing Master LED repeated three times

Bad Master device or internal EEPROM not set up correctly.

Flashing Master LED and any other flashing LED

Misaligned or faulty device. If all eight socket LED's flash, this implies that all sockets are faulty or empty.

Flashing Master LED and any other non-flashing LED.

Programming is taking place.

Master LED on and other LED's off

Verifying Master EPROM.

Master LED off and any other flashing LED

Verify error

Master LED off and any other non-flashing LED.

Power up code for device

7.4.2 Power-up codes

When the ChipWriter Gang is configured for stand-alone mode it will inform the user of the power-up code. These are typically (in HEX):

01	2716	02	2732	03	2764
04	27128	05	27256	06	27512
07	27010	08	27020	09	27040
0A	27080	etc.			
11	2816	etc.			

If any other power up codes or error codes are reported, contact Data I/O or your distributor.

8. Batch Software

Software for batch processing on the ChipWriter Gang is provided free of charge. Contact Data I/O or download from the BBS or Web site.

8.1 Batch software commands

GPCONFIG

usage: GPCONFIG -m manf.typ -n devicename [options]

This command needs to be run first in order to configure the programmer for Batch mode operation

where: manf is the manufacturer file name and
typ = epr|eep|ucs
(see EPROMPAR directory for a complete list)
devicename is an exact devicename, as shown in menu software

options:

-eq error quiet mode, i.e. error returned only via ERRORLEVEL
-e displays error message on next line of screen (default)
-e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
-p 1|2|3 parallel port (default = 1)
-h gives this help screen

GPBCHK

usage: GPBCHK [options]

options:

-ds ADDR device start address in HEX (default=MIN ADDRESS)
-de ADDR device end address in HEX (default=MAX ADDRESS)
-eq error quiet mode, i.e. error returned only via ERRORLEVEL
-e displays error message on next line of screen (default)
-e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
-h gives this help screen

GPCHKSUM

usage: GPCHKSUM [options]

options:

-ds ADDR device start address in HEX (default=MIN ADDRESS)
-de ADDR device end address in HEX (default=MAX ADDRESS)
-eq error quiet mode, i.e. error returned only via ERRORLEVEL
-e displays error message on next line of screen (default)
-e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
-a displays current device address on next line of screen (default)
-a R,C displays current device address at screen row=R, col=C (requires ANSI.SYS)
-h gives this help screen

GPProg

usage: GPProg file [options]

where: file is filename containing data to be programmed.

options:

- ds ADDR device start address in HEX (default=MIN ADDRESS)
- de ADDR device end address in HEX (default=MAX ADDRESS)
- fl ADDR file window low address (default=0)
- ft B|H file type is Binary|HEX (default=HEX)
- fi 1|2|4 file increment, 1=every byte/word, 2=every other byte/word etc.
- eq error quiet mode, i.e. error returned only via ERRORLEVEL
- e displays error message on next line of screen (default)
- e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
- a displays current device address on next line of screen (default)
- a R,C displays current device address at screen row=R, col=C (requires ANSI.SYS)
- b NUM buffer size in Kbytes. Min=64, default=device size (to system max)
- r NUM number of times to repeat. Each repeat is prompted at R,C. The R,C parameters of the -e command must be specified
- i ignore electronic signature error and continue to program
- h gives this help screen

GPRead

usage: GPRead file [options]

where: file is filename for output

Options:

- ds ADDR device start address in HEX (default=MIN ADDRESS)
- de ADDR device end address in HEX (default=MAX ADDRESS)
- eq error quiet mode, i.e. error returned only via ERRORLEVEL
- e displays error message on next line of screen (default)
- e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
- a displays current device address on next line of screen (default)
- a R,C displays current device address at screen row=R, col=C (requires ANSI.SYS)
- h gives this help screen

GPVerify

usage: GPVerify file [options]

where: file is filename to verify against

options:

- ds ADDR device start address in HEX (default=MIN ADDRESS)
- de ADDR device end address in HEX (default=MAX ADDRESS)
- fl ADDR file window low address (default=0)
- ft B|H file type is Binary|HEX (default=HEX)
- fi 1|2|4 file increment, 1=every byte/word, 2=every other byte/word etc.
- eq error quiet mode, i.e. error returned only via ERRORLEVEL
- e displays error message on next line of screen (default)
- e R,C displays error message at screen row=R, col=C (requires ANSI.SYS)
- b NUM buffer size in Kbytes. Min=64, default=device size (to system max)
- r NUM number of times to repeat. Each repeat is prompted at R,C (the R,C parameters of the -e command must be specified)
- h gives this help screen

N.B. the environment variable CWGANG must point to the CWGANG directory

8.2 Batch software utilities

TEXT

Usage: text message row col [fg bg]
message is the text to display (possibly enclosed in double quotes)
row, col is the position on the screen
fg is the foreground colour of the text
bg is the background colour of the text

BOX

Usage: box col [trow lcol brow rcol]
Default box is the whole screen
col is the colour of the box
trow, lcol is the position of the top left corner of the box on the screen (0,0 to max of 22,77)
brow, rcol is the position of the bottom right corner of the box on the screen (trow+2, lcol+2 to max. of 24,79)

ANSIESC

Usage: ansiesc message [message...]
where: message is ordinary text with the special escape code sequences :
\$e = esc code (0x1b)
\$\$ = '\$'
\$f = formfeed char (for use with redirection to printer)
\$s = space
\$n = carriage ENTER and newline
\$d = current directory
\$xHH = char HH (HEX)

For example:

```
ansiesc $e[0m "Current Directory is " $e[1;5;7;33;45m$d$e[0m
```

See DOS Manual for a list of ANSI ESC CODES (including key redefinition's)

BEEP

Usage: beep ifreq1 duration1 [ifreq2 duration2 ...]
Makes a beep. Frequency is inversely proportional to ifreq

For example:

```
beep 400 8 500 8 400 8 500 8 400 8 500 8
```

WINDOW

Usage: Window trow lcol brow rcol type[fgcol][bgcol][title]
trow, lcol is the position of the top left corner of window on screen (0,0 to max of 22,77)
brow, rcol is the position of the bottom right corner of window on screen (trow+2, lcol+2 to max. of 24,79)
type is the type of border (0=double, 1=single, 2= blank, 3= horizontal double)
fgcol = foreground colour
bgcol = background colour
title is the optional title for the window (If title is specified then fgcol and bgcol must also be specified)

WAITKEY

Usage: waitkey keylist
keylist is a list of chars to wait for. All others are ignored. Possible keys are: A-Z, 0-9, F1-F10, UP, DOWN, LEFT, RIGHT, SPACE, ENTER, PAGEDOWN, PAGEUP, HOME, END, ESC or DEFAULT.
Returns when a key in keylist is pressed and sets the ERRORLEVEL equal to the position of the key within the keylist (leftmost argument = 1, etc.)

If DEFAULT is specified any key not in the keylist returns ERRORLEVEL = 0

For example:

```
waitkey CR space F10 a b c DEFAULT
IF ERRORLEVEL 6 goto Keyc
IF ERRORLEVEL 5 goto Keyb
IF ERRORLEVEL 4 goto Keya
IF ERRORLEVEL 3 goto F10KEY
IF ERRORLEVEL 2 goto SPACE
IF ERRORLEVEL 1 goto CR
:default
```

Note: Since, in DOS, ERRORLEVEL n statement implies ERRORLEVEL >=n then a reverse order is important here.

WARN

Usage : Warn message [fgcol] [bgcol] [bfgcol] [bbgcol][type]

fgcol = foreground colour

bgcol = background colour

bfgcol , bbgcol
refers to border surround

type is the type of border (0=double, 1=single, 2=blank, 3=horizontal double) default is white on black, type 1 border

8.3 Batch software error levels

Message Returned	Error Level
No Error	0
Bad Command Line Parameters	1
No Environment Variable found	2
Manufacturers File Not Found or Invalid	3
Bad Devicename Specified	4
Bad Device Driver Specified	5
Programmer is not a ChipWriter Gang	6
Bad Library File	8
Bad communication with Programmer	9
Checksum Error	10
Memory Error	11
Bad file Read	12
Bad Device Specified	13
File was not found or could not be opened	14
Bad definitions in Command Line Parameters	15
Bad address range has been specified	16
Device is not Blank	17
Verify Error	18
Device or Parameter is not Programmable	20
Bad Electronic Signature	21
Device is not Secure	22
Device not Erased	24

Any other values are internal errors and you should contact Data I/O before continuing.

8.4 Batch software examples

Example 1:

This batch file shows the simplest way of using the batch programs. No error checking is done and all information is echoed to the screen.

```
REM Setting up the environment variable to point to the ChipWriter software
REM This only needs to be done once and may be done in the AUTOEXEC.BAT file.
SET CWGANG=C:\DATAIO\CWGANG
REM Configuring the programmer to program Signetics 27C64A
GPCONFIG - M SIGNETIC.EPR -N 27C64A
REM Programming and verifying the device
GPPROG TESTFILE.HEX
```

Example 2:

This batch file has basic error checking throughout. It reports a message at the end informing the user of the success or failure of the attempt to program. For a complete list of errorlevels see section 8.3

Remember In DOS, when errorlevels are being tested, a test of `errorlevel = 1` will jump to 'fail' if the errorlevel is greater than or equal to 1.

```
@ECHO OFF
REM Setting up the environment variable to point to the ChipWriter software
REM This only needs to be done once and may be done in the AUTOEXEC.BAT file.
SET CWGANG=C:\DATAIO\CWGANG
REM Configuring the programmer to program Signetics 27C64A
GPCONFIG - M SIGNETIC.EPR -N 27C64A
If errorlevel = 1 goto fail
REM Programming and verifying the device
GPPROG TESTFILE.HEX
If errorlevel = 1 goto fail
echo Device programmed OK
goto end
:fail
echo Programming error
:end
```


Appendix A: Shorthand Keystrokes

For quick operation of Data I/O Programmers the following shorthand keystrokes are available :-

CWGANG

Open	Alt+O
Save	Alt+S
Exit	Alt+X
Data Edit	Alt+E
Checksum	Alt+C
Fill Buffer	Alt+F
Goto Address	Alt+G
Buffer Size	Alt+Z
Read	Alt+R
Verify	Alt+V
Blank Check	Alt+B
Program	Alt+P
Manufacturer	Alt+M
Type	Alt+T
Devicename	Alt+N
DOS Command	Alt+D
Display Mode	Alt+Y
Gang Mode	Alt+1
Sets of 2	Alt+2
Sets of 4	Alt+4

Appendix B : Menu Options

This section gives a brief overview of each of the menu options available in CWGANG.EXE and tells you in which section of the manual you will find more detailed information.

CWGANG.EXE

<u>Menu Option</u>	<u>Section</u>
FILE MENU	
Open Open a binary or HEX formatted from disk and load it into buffer memory area.	6.2
Save As Save an area of the buffer to a binary or formatted file on disk.	6.2.7
Save Save the currently loaded file	6.2.7
Exit Exit the software and save the settings.	5.11
DATA MENU	
Edit Edit the contents of the data buffer.	6.3.1
View View the contents of the data buffer without making any changes.	6.3.2
Fill Fill an area of the buffer with a specified value.	6.3.4
Copy Copy an area of the buffer to another area.	6.3.4
Search Search an area of the buffer for a specified value or string.	6.3.3
Goto Goto a user specified area of the buffer.	6.3.3
Swap Bytes Swaps odd and even bytes. Useful for data into more than 1 EPROM.	6.3.4
Buffer Size Changes to size of the buffer.	6.2.1

Print 6.3.7
Prints out an area of the buffer

Checksum 6.3.5
Calculates a checksum over a given area.

PROCESS MENU

Read 6.4.1
Read the device in the ZIF socket and store the data in the buffer.

Verify 6.4.2
Verify the data in the device against the data in the buffer.

Verify Electronic ID 6.4.3
Checks the electronic signature of the device.

Blank Check 6.4.4
Checks to see if a device is blank.

Program 6.5.2
Programs data in the buffer into the device.(inc. Verify Sig & Verify)

Erase 6.5.4
Erases Electronically erasable devices.

DEVICE MENU

Manufacturer 6.1.1
Selects the manufacturer of the device to be programmed.

Type 6.1
Selects the type of device to be programmed.

DeviceName 6.1
Selects the devicename of the device to be programmed.

Modify Parameters 6.6
Allows users to add their own devices.

Add to User Database 6.6
Stores new devices in the user library.

OPTIONS MENU

Parallel Port 5.5

Tells the software which parallel port the programmer is connected.

Screen Colour 5.5
Selects either mono or colour displays.

Snow Precautions 5.5
Overcomes snow problems on old CGA displays.

Hi-Res Mode 5.5
Changes number of lines on EGA and VGA displays.

Display Mode 5.5
Switch between 8, 12, 16 bit display.

DOS Command 5.5
Enables a DOS command to be run from the software.

APPENDIX C: Technical Assistance, Warranty, and Repair

This appendix includes details about contacting Data I/O for technical assistance, repair and warranty services. The appendix also explains the Bulletin Board Service.

If you require technical support for any reason, please ensure that you have the following information on hand:

1. The serial number of your programmer.
2. The firmware version : To find out which firmware version you are using run DETECT. The firmware version number is displayed at the end of the test.
3. The software version and library version for the device you are programming: To find out which version of the library you are using connect the programmer and run CWGANG. Select the manufacturer and device name of the device being programmed. The library version number is then displayed on the third line of the screen.
4. The results of the DETECT program: To run the program, ensure that you are in the directory containing the programmer software (e.g. C:\CWGANG) and type :

DETECT

at the DOS prompt. The program will then report on which port the programmer is connected to. This information can then be used to set up the parallel port in the CWGANG.EXE software. If an error is reported then obtain a printout by using

DETECT >PRN

or

DETECT >ERROR.LST followed by PRINT ERROR.LST

5. The exact device name of the device you are programming.
6. An exact note of any error messages you are getting.

It will assist our engineers if you are next to the programmer when calling.

Data I/O Customer Support

United States

For technical assistance, contact:

Data I/O Customer Resource Center

Telephone: 800-247-5700

Fax: 425-869-2821

For repair or warranty service, contact:

Data I/O Central Dispatch

Telephone: 800-735-6060

Fax: 425-881-0561

Canada

For technical assistance, contact:

Data I/O Customer Resource Center

Telephone: 800-247-5700

Fax: 425-869-2821

For repair or warranty service, contact:

6725 Airport Road, Suite 302
Mississauga, Ontario, L4V 1V2
Telephone: 905-678-0761
Fax: 905-678-7306

Japan

For technical assistance, repair, warranty service, contact:

Data I/O Japan

Osaki CN Building 2F
5-10-10, Osaki Shinagawa-Ku
Tokyo 141
Telephone: 3-3779-2207 (Operations)
3779-2203 (Other)

Germany

For technical assistance, repair, warranty service, contact:

Data I/O GmbH

Lochhamer Schlag 5a
82166 Gräfeling
Telephone: 089-858580
Fax: 089-8585810

Other European Countries

For technical assistance, repair or warranty service, contact your local Data I/O representative.

Other Countries World-wide

For technical assistance, repair or warranty service, contact the office below and ask for the number of your local Data I/O representative.

Data I/O International
10525 Willows Road N.E.
P.O. Box 97046
Redmond, WA USA 98073-9746
Telephone: 425-881-6444
Fax: 425-882-1043
Telex: 4740166

Technical Assistance

You may contact Data I/O for technical assistance by calling, sending a fax or electronic mail (e-mail), or using the Bulletin Board Service (BBS). To help us give you quick and accurate assistance, please provide the following information:

- Product version number

- Product serial number (if available)
- Detailed description of the problem you are experiencing
- Error messages (if any)
- Device manufacturer and part number (if device-related)

Calling

Calling the appropriate Data I/O Customer Support number listed at the front of this Appendix. When you call, please be at your programmer or computer, have the product manual nearby, and be ready to provide the information listed above.

Sending a fax

Fax the information listed above with your name, phone number, and address to the appropriate Data I/O Customer Support fax number listed at the front of this Appendix.

Sending E-mail

To reach Data I/O via e-mail, send a message including your name, phone number, e-mail address, and the information listed above to the following address:

techhelp@data-io.com

Logging on to the BBS

The Data I/O Bulletin Board System (BBS) enables you to:

- Obtain a wide range of information on Data I/O products, including current product descriptions, new revision information, technical support information, application notes, and other miscellaneous information
- Access device support information
- Request support for a particular device
- Leave messages for the BBS system operator, Customer Support personnel, or other customers
- Download many DOS and Windows utilities

Online help files provide more information about the BBS and its capabilities.

Before you can log on to the BBS, you must have your telecommunications software and modem installed and operating. Refer to your telecommunications software manual and your modem manual for setup procedures.

Have the following information available prior to contacting the BBS:

Name and Password

File transfer protocol (used by your telecommunication software)

Use the following procedure to log on to the BBS:

- To contact the BBS, call one of the following numbers:

(Modem)

United States

+1-206-881-3465

Germany	+49-89-858-5880
	+49-89-858-5833
Japan	+81-3-3779-2233

(Internet: Using, telnet, go to `bbs.data-io.com`)

- If this is the first time you have called, when the logon screen appears, type `new` at the prompt to create a new account and provide the requested information.

Note: Be sure to write down your user name and password for future reference. This is your personal BBS account and should not be shared.

- Select the Library Files menu
- Select the ChipWriter Library (CHIPWRIT)
- Search for the ChipWriter Gang files.

Using the Data I/O Web Page

The Data I/O Home Page on the World Wide Web (WWW) includes links to online information about technical products, general information about Data I/O, a list of sales offices, and technical user information such as application notes and device lists.

To access the WWW, you will need an Internet account with Web access, and a Web browser such as Netscape or Mosaic. The address of the data I/O Home Page is:

`http://www.data-io.com`

You may also access the data files from an FTP at the following address location:

`ftp.data-io.com/dataio/chipwriter`

Click the (Programmer Device Support) image or go to Technical Information and then ChipWriter Gang software.

- Once you have reached the ChipWriter Gang Library, select the correct file and download the ChipWriter Gang compressed file.
- The ChipWriter Gang file comes in a self-extracting file format. Place the file on a floppy disk or temporary hard drive directory and then expand the file by running it. The following files should be created:

<u>File Name</u>	<u>Description</u>
DIOGANG.EXE	
INSTALL.BAT	

Note: It is recommended that the disk is labelled "ChipWriter Gang" and add the version number to avoid mismatched software version numbers if it becomes necessary to use the disk again.

- At the DOS prompt, run the INSTALL.BAT file and follow the instructions for installing in to the correct directory.

Warranty Conditions

Data I/O Corporation warrants this product against defects in materials and workmanship at the time of delivery and thereafter for a period of one (1) year.

The foregoing warranty and the manufacturers' warranties, if any, are in lieu of all other warranties, expressed, implied or arising under law, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Data I/O maintains customer service offices throughout the world, each staffed with factory trained technicians to provide prompt, quality service. For warranty service contact your distributor or local sales office.