TABLE OF CONTENTS

Appendices

1. Introduction

The Multi-Format-BIOS (MFB) is an extension of the normal Gemini BIOS, and as such supports all the usual enhanced features such as disk error trapping, screen edit, screen dump etc. However the MFB will support a variety of CP/M disk formats and disk sizes simultaneously, and provides the ideal way of copying CP/M disk files between dissimilar formats.

The BIOS is table-driven, there being a table entry for each logical drive on the system. These tables define the physical characteristics of each drive (size, tracks, step rate etc), together with the format to be used. To reconfigure the BIOS for alternative formats only these tables need to be changed.

The BIOS will support most current soft-sectored formats, but cannot handle those formats that depend on unique hardware features. (See Appendix A).

To support the MFB five programs (HARDWARE, SETUP, WHIG, ANALYSE, and UPDATE) and a data file (DRVFMT.DAT) are supplied:

DRVFMT.DAT holds information on the system's hardware configuration (what disk drives are connected, and their physical addresses), the characteristics of the disk drives - size, tracks, number of sides, step rate etc, and finally details of the disk formats - reserved tracks, number of directory entries, disk block allocation size etc. The three main utility programs all pick up their data from this file.

HARDWARE is used to list/enter data to DRVFMT.DAT on the system's hardware configuration, and the physical characteristics of the drives. **You will only need this program if you change any of the drives fitted to your system, add an additional drive or wish to change the displayed drive names.**

SETUP is the main utility program and is used to add new formats to DRVFMT.DAT, to format disks, and to set up a different configuration of formats within the BIOS. This is the program that directly patches the drive and format tables within the BIOS, making system reconfiguration only a few seconds work.

ANALYSE is provided to help determine the physical characteristics of an unknown format. When presented with a disk of unknown format, it will determine the recording density used, the number of tracks-per-inch, the number of sectors per track, the sector size, and a few other parameters relating to the physical format.

WHIG is a simple utility that displays the current configuration of the BIOS.

UPDATE is a utility that will merge two Data files together, and can also be used to sort an existing data file into alphabetical order.

2. System Hardware

The BIOS is based on the Gemini Multiboard range of 80-BUS compatible computer boards, and in particular requires the GM829 disk controller card.

Currently the BIOS will support up to six logical drives. These are made up from one Winchester drive, four floppy disk drives, and one "memory" drive. Logical drive A is fixed in type and cannot be redefined. Two alternatives are available, one with a Gemini format 96tpi Micropolis drive as drive A, the other with a Rodime R0201 5Mbyte Winchester as drive A.

A system could for example consist of:-

Drive A: Rodime R0201 5Mbyte Winchester.
Drive B: A 5.25" 96tpi double-sided drive.
Drive C: A 5.25" 48tpi double-sided drive.
Drive D: A 5.25" 100tpi double-sided drive.
Drive E: An 8" double-sided drive.
Drive M: Optional Memory drive.

3. BIOS customisation

The standard Gemini program CONFIG can be used to set up the usual features of printer support, baud rates and the like, and the details of operation can be found in the normal Gemini documentation. The enhanced features of the MFB are handled by the program SETUP and are described below.

4. Using an MFB system

The system appears as a normal Gemini CP/M system, with the usual enhanced features such as on-screen editing and screen dump available. All CP/M programs can be used as usual, and the only difference is that the size of the TPA is slightly reduced due to the extra features incorporated into the BIOS.

The BIOS configuration,(i.e. what logical drive is what format), can be changed quickly at any time by running the program SETUP and selecting option 4 (see section 5). As the system is so flexible and can be changed so easily, the utility WHIG (What Have I Got?) is provided, so that you can see the current assignment of the logical drives.

IMPORTANT

One point to remember when using SETUP is that it performs two different functions. One is to maintain the format information in the data file DRVFMT.DAT. The other is to use this information to patch the BIOS to support those formats. So if you have created a format BLOGGS, and set up the BIOS so that Drive C is BLOGGS format, and subsequently discover an error in your data entry for the format, you have to do two things. The first to use option 1 in SETUP to correct the BLOGGS entry in the data file, and the second is to use option 4 to set up the BIOS again, this time using the correct data for the BLOGGS format. CHANGING THE DATA FILE DOES NOT ALTER ANY INFORMATION IN THE BIOS. It is only when you use option 4 in SETUP that any changes occur.

5. SETUP

The program SETUP is used to enter the details of formats and drives to be handled by MFB. This data is stored in a disk file (DRVFMT.DAT) which should be present on drive A whenever SETUP is run. The data file is restricted to drive A, because if SETUP is used to construct a new BIOS configuration, all the pointers associated with the other drives may change, resulting in the BDOS flagging them as Read/Only and thus preventing the save of an updated version of the file.

5.1. General

While SETUP is running, the data file is held in memory, and is only written to disk if the Format-a-disk option is selected, or when finally exiting to CP/M. Thus if the program is exited prematurely (by typing ^C for example), any new data entered since the last save will not be written to disk.

5.1.1. Data input and defaults

Most command input to SETUP is obtained via a buffered line input routine.i.e. SETUP does not act on your reply until the RETURN key is presssed. In some instances (shown below) SETUP will display on the current line the default input value that it will assume if nothing is typed in except RETURN. Note that even if you only wish to change one character of the value displayed, the correct value must be TYPED IN FULL as this method of input does not support on-screen editing. For confirmation SETUP redisplays the value it has accepted when RETURN is pressed.

5.1.2. Input checking

In general SETUP checks data on entry, and will provide error messages and/or warning messages if incorrect or inconsistent values are entered. The checking is comprehensive but not exhaustive, and is intended to catch the small mistake rather than the gross error. For example SETUP will not let you define a sector translation table with the wrong number of sectors, or where a sector number is entered more than once. But it will let you define a format with 50 1024-byte sectors to a track on a single density 5" disk.

5.1.3. Error messages

In the event of an error occuring, either in the response you have entered (e.g. illegal character), or as a result of that entry or an earlier entry, an error message will flash briefly at the bottom of the screen before the program continues. It may request you to re-enter the erroneous value(s), or may return to the main menu depending on the circumstances surrounding the error.

5.1.4. Aborting a Command

You may abort the current operation at any time by typing ESC (the ESCAPE key) while replying to the prompt for input by SETUP. e.g. If you have selected option 2, (List current Formats), you may terminate the listing at any time by pressing "ESC" in response to the prompt "Type <RETURN> to continue".

5.1.5. Print-out of display

You may print out the contents of the current screen at any time by typing ^P (pressing the 'P' key whilst holding down the CONTROL key) while replying to the prompt for input by SETUP. e.g. If you have selected option 2, (List current Formats), you may obtain a printer copy of the display by pressing ^P in response to the prompt "Type <RETURN> to continue".

5.2. Main Menu

On running SETUP it will display the initial menu:

```
================================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201     (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800

          Return to CP/M.............0
          Enter a new format.........1
          List current Formats.......2
          Format a disk..............3
          Construct a BIOS...........4

          Enter required option :

================================================================================
```

The top three lines of the display will remain in place throughout the execution of the various options within SETUP, and show the current configuration of the MFB.

The top line of the display gives the version number of SETUP. The second line shows the formats supported by each drive, and the third line shows the drive physical address (in brackets) followed by the name of the supported drive. If a drive is undefined its entry remains blank.

If the running system does not incorporate the MFB a message to that effect will be displayed in place of the format/drive information on the second and third lines. SETUP will still run, but you will not be allowed to use option 4 (see below).

Enter a number between 0 and 4 followed by RETURN to perform the operation you require.

5.3. Enter a new Format

Typing "1 <RETURN>" in the main menu will select this option and produce the screen display:

```
================================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201     (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800
             Format Definition

       Format name.........:
       Format description...:

Density (S/D).....:      Invted data (Y/N).:      Reserved tracks....:
Drive type (8/5/W):      Index mark (Y/N)..:      Logical secs/track.:
Tracks-per-inch...:      First side........:      First sector no....:
Tracks-per-disk...:      Side change (T/S).:      Phys. sector skew..:
Sides-per-disk....:      Block size (k)....:      Gap 1 length.......:
Sectors-per-track.:      Extent mask.......:      Gap 2 length(34/43):
Bytes-per-sector..:      Directory entries.:      Gap 3 length(>34)..:

Physical sector numbers:


Sector translate table :



================================================================================
```

The cursor will be positioned ready to accept a Format Id (see below). If you have just selected option "1", then the string typed in here will be compared against those already stored in the data file. If a match is found then that entry will be copied to the screen, and you will pass immediately to the "Save, Delete, Edit, Ignore? " prompt (see below). If the string did not match an existing entry in the data file, then the remaining fields have to be filled in.

The answers required for each entry are shown below:-

Format name: Enter an eight-character string. This is the string that will be used to identify this particular format entry. The input string is forced to upper case irrespective of whether upper or lower case is used in typing it in.

Format description: You can enter a string of up to twenty-one characters to expand on the "name" field.

Density (S/D): Enter S for single density, or D for double density. This field is used by the BIOS for configuring the disk controller when accessing a disk of this particular format. It is also used by the format option when formatting a disk.

Drive type (8/5): This is a single letter that identifies the type of drive required for this particular format. SETUP cross checks this entry against that of the specified drive entry when constructing a BIOS (see below).

Tracks-per-inch: Enter the appropriate figure for the drive (e.g. 48 or 96). SETUP cross-checks this field with a Drive description when setting up a BIOS.

Tracks-per-disk: Enter the number of tracks per side expected by the Format. SETUP cross-checks this field with that in a Drive description when setting up a BIOS. It also uses this field in calculating the capacity of a disk.

Sides-per-disk: Enter the number of sides to the disk required by this Format.(For floppies this will be 1 or 2). SETUP cross-checks this field with that in a Drive description when setting up a BIOS. It also uses this field in calculating the capacity of a disk.

Sectors-per-track: Enter the number of physical sectors per track per side of the disk. SETUP uses this figure to determine the physical sector numbers (shown below) and also in accessing the disk. The number must lie between 1 and 52.

Bytes-per-sector: Enter the number of bytes per sector. This should be a power of 2, and lie between 128 and 1024. SETUP uses this figure in determining whether Blocking/Deblocking is to be performed, and also in calculating the capacity of a disk. The Format-disk command also uses this field.

Inverted data (Y/N): The reply to this question is normally N for No. However some formats (e.g. Superbrain) write the complement of the data onto the disk. If this field is set to "Y" the BIOS will complement the data during all disk I/O.

Index mark (Y/N): This field is used by the Format-disk section. If a 'Y' is entered the format program will write an Index mark at the start of every track on the disk before writing the data sectors. With the Western Digital controller on GM829 and soft-sectored formats, the Index mark is superfluous and is never used. It must be left off for the Gemini formats (to ensure that there is sufficient space for the data blocks), but can be included for those formats that allow for it.

First side: This field is currently not used. It is included for 8" double density formats which usually split the two sides of a double-sided disk into two logical drives. Currently the BIOS does not support this feature, but may be extended at a later date to do so.

Side change (T/S): For formats that use both sides of a double-sided disk this field indicates whether the BIOS should access both sides of the disk on a given track (option T), before stepping to the next track, or use one side completely before using the second side (option S).

Block size (kbytes): This is the block allocation size used by CP/M for this particular format. SETUP uses this field when it constructs the disk parameter block for a particular drive while setting up a BIOS.

Extent mask: Normally you should enter an '*' for this field. SETUP takes this as an instruction to use the correct value for this parameter when constructing the disk parameter block. However some formats have been encountered, (e.g. Televideo), where this value has been set incorrectly. In this case the value that SETUP is to use should be entered here. (See appendix B).

Directory entries: This is the number of directory entries supported by this particular format.

Reserved tracks: Enter the number of tracks on the disk reserved for the system tracks.

Logical secs/track: Normally a logical track is equivalent to a physical track and SETUP will use a figure calculated automatically based on the Bytes-per-sector/Sectors-per-track/Side change fields. As with the 'Extent Mask' field this automatic option is selected by entering an '*' (or 0). However in the case of the supplied RAIRBB and M3300 QD formats you fill find that the field contains a different value to that which you would expect. This is because, taking the M3300 QD format as an example, the system is configured so that a physical sector is a logical track. (Hence the logical secs/track is 4). For formats like this the BIOS has to do additional work to translate a logical track and sector combination to the correct physical track and sector numbers on every disk Read/Write. In the unlikely event of your encountering a similar format, the appropriate logical track size should be entered here.

NOTE: The last three items above, plus the block size, can easily be determined by doing a "STAT DSK:" command on a system running the desired format.

First sector no: Enter the number of the first sector of a track. This is usually either 0 or 1.

Physical sector skew: This field is used for those formats, such as the Gemini formats, that achieve sector skewing by physically skewing the sectors on a disk during the formatting operation. If no skew is to be used then a 0 should be entered. This field is used only by the Format-disk option.

Gap 1: This field is used by the Format-disk section, and is the number of bytes that are inserted following the Index mark, (if present), and before the first sector header.

Gap 2: This field is used by the Format-disk section, and is the number of bytes that are inserted between a sector header and its associated data block.

Gap 3: This field is used by the Format-disk section, and is the number of bytes that are inserted between the end of a sector and the start of the next sector header.

Physical sector numbers: This field shows the order in which sectors will be written onto the disk by the Format-disk option. It is automatically filled in by SETUP using the values from the First sector no/Physical sector skew/Sectors-per-track fields.

Sector translate table: If sector skewing is performed by a table, (such as in the standard 8" single density format), then the table must be entered here. There must be as many entries as there are physical sectors. If the physical sector size is 128 then the translation is done via SECTRAN in the BIOS in the usual way, but if the sector size is > 128 then the translation is performed within the disk read/write routines of the BIOS. SETUP will not let you duplicate sector numbers in the table, and can help you with the entry if you require.

You may either enter all the sector numbers in one go on a single line, or press <RETURN> after entering each number. In the latter case SETUP will assist you in constructing the table. It will determine the sector skew you are using by examining the first two sector numbers entered. Thereafter every time <RETURN> is pressed it will redisplay the values entered so far, and will add onto the end what it thinks should be the next sector number. If <RETURN> on its own is pressed this will be taken as the default value for the next sector, and it will update the display.

NOTE SETUP does not automatically "slip" the next sector number if the skew does not exactly fit in with the sectors per track. However if you fail to notice that a sector number is about to be repeated SETUP will provide an error message when <RETURN> is pressed, and will not accept the entry. You can then manually alter the number and continue.

If you have previously entered a sector translation table and wish to delete it, this can be done by entering a single '-' at the start of the table.

Once all the fields have been completed the prompt:

Save, Delete, Edit, Ignore? (S/D/E/I):

will appear at the bottom of the screen. As indicated four options are open:

S Will save the displayed entry to the data file, overwriting any existing entry with the same "Format name"

D Will delete any entry with a matching "Format name" from the data file.

E Will move the cursor back to the start of the entry, allowing any erroneous values to be changed. Note that this time altering the "Format name" field to match an existing entry will not result in that entry overwriting the currently displayed values.

It is also possible to quickly enter a new format that is very similar to one that already exists. By typing the ID of the existing format in the first instance, the appropriate values are copied into the display. Then the "E" option can be used to alter the ID field, and any of the other values that need to be changed.

I Will return you to the main menu without altering the data file.

5.4. List Current Formats

Typing "2 <RETURN>" in the main menu will result in the data on all the current Formats being displayed on the screen. In the "full" display mode the entries are shown one at a time in the form illustrated below. The RETURN key is used to step through from one entry to the next, and the ESCape key can be used to return to the main menu.

```
================================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201       B = GEMQDDS     C = GEMDDDS     D = M2200 QD    E = IBM3740
(W)  R0201      (1)  1015F-VI   (2)  FD250      (1)  1015F-VI   (3)  SA800
             Format Definition

       Format name.........: IBM3740
       Format description...: Standard SD 8"

Density (S/D).....: S     Invted data (Y/N).: N     Reserved tracks....: 2
Drive type (8/5/W): 8     Index mark (Y/N)..: Y     Logical secs/track.: *
Tracks-per-inch...: 48    First side........: 0     First sector no....: 1
Tracks-per-disk...: 77    Side change (T/S).: S     Phys. sector skew..: 0
Sides-per-disk....: 1     Block size (k)....: 1     Gap 1 length.......: 32
Sectors-per-track.: 26    Extent mask.......: *     Gap 2 length(34/43): 17
Bytes-per-sector..: 128   Directory entries.: 64    Gap 3 length(>34)..: 33

Physical sector numbers: 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26

Sector translate table : 01 07 13 19 25 05 11 17 23 03 09 15 21 02 08
14 20 26 06 12 18 24 04 10 16 22

                    Type <RETURN> to continue

================================================================================
```

Fig. Full Format Display

If only a brief summary is required a 'Q' (for Quick) should be appended to the "2". (i.e. type "2q"). In this case only the format name and format description fields are printed, one per line, without any pause until the list is complete. If there is insufficient space available in the display area to show all the formats, then the cursor-up and cursor-down keys may be used to scroll the list of formats up and down through the display area.

```
================================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201       B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201      (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800

                          Formats

ALTOS DJ  Altos mini                    ALTOS5-5  Altos mini
ALTOSDD   Altos 8" DD                   ALTOSNS   Altos non-standard
BBC       BBC - non CP/M                COMPUGRA  MCS Typeset NOT CP/M
CROMSDSS  Cromemco SD/SS                DEC       DEC Rainbow & Prof.
DECVT180  DEC VT180 format              DIGICO    Prince
EPSON/QX  Epson QX10                    FTS SG    Single sided (Spec)
GEMDDDS   Gemini 48tpi DD               GEMQDDS   Gemini Galaxy DS
GEMQDSS   Gemini Galaxy SS              GEMSDDS   Gemini 48tpi SD
IBM3740   Standard SD 8"                IBMPC     IBM PC CP/M86
IBMS34    8" Double density             ICL/8801  ICL Vers CP/M 86
IDS 2009  from Holland                  ITT3030   From Atebo
KAYPRO    from Holland                  M2200 DD  BT MERLIN Doub. dens
M2200 QD  BT MERLIN Quad Dens           MCCOMBO   5" Booting disc
MCS-TYPE  MCS Typeset Not CP/M          MEMDDDS   Memory 4 but EXM=0!
MIMI803   MIMI 803                      NASCOMDS  Nascom QD/DS
NASCOMSS  NAscom QD/SS                  NCRDM5    NCR Decision mate V
ORION DS  Orion CP/M 86                 ORION SS  Orion CP/M 86
          Use Cursor keys to Scroll - RETURN to exit
================================================================================
```

Fig. Abbreviated Format Display

5.5. Format a Disk

Typing "3 <RETURN>" in the main menu will select this option. If SETUP is running under the Multi-Format-BIOS you will see the prompt:

Which Drive? ([A-E] or ESC)

You may respond with either a Logical drive number (A-E), or press the ESCape key. If a valid logical drive letter is entered then SETUP will use the drive and format pair (currently displayed at the top of the screen) which are associated with that logical drive. If ESCape is pressed SETUP will display the non-MFB prompt of:

```
==========================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W) R0201      (1) 1015F-VI  (2) FD250     (1) 1015F-VI  (3) SA800
                        Format a Disk option

ALTOS DJ  Altos mini                    ALTOS5-5  Altos mini
ALTOSDD   Altos 8" DD                   ALTOSNS   Altos non-standard
BBC       BBC - non CP/M                COMPUGRA  MCS Typeset NOT CP/M
CROMSDSS  Cromemco SD/SS                DEC       DEC Rainbow & Prof.
DECVT180  DEC VT180 format              DIGICO    Prince
EPSON/QX  Epson QX10                    FTS SG    Single sided (Spec)
GEMDDDS   Gemini 48tpi DD               GEMQDDS   Gemini Galaxy DS
GEMQDSS   Gemini Galaxy SS              GEMSDDS   Gemini 48tpi SD
IBM3740   Standard SD 8"                IBMPC     IBM PC CP/M86
IBMS34    8" Double density             ICL/8801  ICL Vers CP/M 86
IDS 2009  from Holland                  ITT3030   From Atebo
KAYPRO    from Holland                  M2200 DD  BT MERLIN Doub. dens
M2200 QD  BT MERLIN Quad Dens           MCCOMBO   5" Booting disc
MCS-TYPE  MCS Typeset Not CP/M          MEMDDDS   Memory 4 but EXM=0!
MIMI803   MIMI 803                      NASCOMDS  Nascom QD/DS
NASCOMSS  NAscom QD/SS                  NCRDM5    NCR Decision mate V
ORION DS  Orion CP/M 86                 ORION SS  Orion CP/M 86
                    Enter Format code:
==========================================================================
```

In reply to this enter the Format that you require. SETUP will locate and use the drive appropriate to that format. If the number of available formats is too large to be displayed in the space available, then cursor-up and cursor-down keys can be used to scroll the display up and down within the available display window.

NOTE: The reply is forced to upper case.

Once a valid format has been selected the display will change to:

```
================================================================================
        Gemini Multi-Format-BIOS Setup program      Version 2.5
 A = R0201       B = GEMQDDS     C = GEMDDDS     D = M2200 QD    E = IBM3740
(W)  R0201      (1)  1015F-VI   (2)  FD250      (1)  1015F-VI   (3)  SA800
                      Format a Disk option


                      Drive....: SA800
                      Format...: IBM3740



                      Insert Disk in drive E



                      Type <RETURN> to continue

================================================================================
```

There will be a short delay before the last two messages appear while SETUP constructs the appropriate track image in memory that the format program will use. If you started the option by specifying a format, then the drive type will appear in the "Insert Disk.." message, rather than a logical address as is illustrated above.

Upon pressing <RETURN> SETUP will format the disk, displaying the number of the track currently being formatted as it proceeds. Once the format operation is complete SETUP proceeds to read the entire disk to verify that all the sectors can be read. During the verification phase it will report all retries that occur, along with the reason why the read failed. It is only after eight retries that a sector is abandoned, the message <<< Bad <<< being appended to the display to indicate this.

Any disk that produces more than one or two retries should be regarded with suspicion, and any one that results in <<< Bad <<< messages should be discarded if they re-occur if the disk is re-formatted.

The more usual error messages during the verification read are:

RNF - Record not found. The controller was unable to locate either the sector header, or the following data block.

CRC - CRC error. A CRC error was detected, either in the header field or the following data block.

Other possibilities are:-

DNR - Drive Not Ready.

WP - Disk Write protected. (Should not appear).

LD - Lost Data. (Should not appear).

??? - Undefined error status. (Should not appear).

5.6. Construct a BIOS

Typing "4 <RETURN>" in the main menu will select this option. As this option directly patches the running BIOS it will not run unless the current BIOS is an MFB.

As mentioned above this option directly patches the running BIOS, thus the current configuration will be disturbed. The patching starts with drive B, and proceeds through C,D,E etc, and ends when either Drive E has been defined, or the ESCAPE key is pressed.

On completion you are given the option of copying the changed BIOS to the system tracks of the disk in drive A in order to record the changes permanently. NOTE This is NOT equivalent to a "SYSGEN". Only the changed parts of the BIOS are written to the disk, not the entire CCP/BDOS/BIOS.

The first prompt for a Format Code is shown below. Note that a series of ^^^^^^^^ on the fourth row of the display indicate which logical drive is being defined.

```
================================================================================
        Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201       B =            C =            D =            E =
(W)  R0201
configuring->    ^^^^^^^^^^^

                                      Formats
ALTOS DJ  Altos mini                    ALTOS5-5  Altos mini
ALTOSDD   Altos 8" DD                   ALTOSNS   Altos non-standard
BBC       BBC - non CP/M                COMPUGRA  MCS Typeset NOT CP/M
CROMSDSS  Cromemco SD/SS                DEC       DEC Rainbow & Prof.
DECVT180  DEC VT180 format              DIGICO    Prince
EPSON/QX  Epson QX10                    FTS SG    Single sided (Spec)
GEMDDDS   Gemini 48tpi DD               GEMQDDS   Gemini Galaxy DS
GEMQDSS   Gemini Galaxy SS              GEMSDDS   Gemini 48tpi SD
IBM3740   Standard SD 8"                IBMPC     IBM PC CP/M86
IBMS34    8" Double density             ICL/8801  ICL Vers CP/M 86
IDS 2009  from Holland                  ITT3030   From Atebo
KAYPRO    from Holland                  M2200 DD  BT MERLIN Doub. dens
M2200 QD  BT MERLIN Quad Dens           MCCOMBO   5" Booting disc
MCS-TYPE  MCS Typeset Not CP/M          MEMDDDS   Memory 4 but EXM=0!
MIMI803   MIMI 803                      NASCOMDS  Nascom QD/DS
NASCOMSS  NAscom QD/SS                  NCRDM5    NCR Decision mate V
ORION DS  Orion CP/M 86                 ORION SS  Orion CP/M 86
                    Enter Format code:

================================================================================
```

A default Format code will be shown if the logical drive has been defined in the previous configurations. In response to the prompt a format code should be selected from those displayed on the screen and entered. (Or RETURN pressed to take the default). If the number of available formats is too large to be displayed in the space available, then cursor-up and cursor-down keys can be used to scroll the display up and down within the available display window.

(The cursor keys will only be recognised while the cursor is positioned at the start of the data field). One other option is open, and that is to press the ESCape key instead. Doing this will terminate the "drive definition" phase.

Once RETURN is pressed SETUP checks that the selected format exists. If it does not the error message ("Format Code is undefined") will flash briefly at the bottom of the screen, and you will be prompted for a new format.

Once the format has been located in the data file, SETUP checks the characteristics of all the drives in your system (as defined under HARDWARE), in order to locate the drive that can support it. (Right type of drive, correct tpi, sufficient number of tracks/sides, etc). If no suitable drive can be found, the error message "No drive supports the format" will be displayed and you will be requested to re-enter the format.

NOTE 1: It is permissable to enter several formats that will use the same physical drive. For example on a system with a Pertec FD250 type drive:

Logical drive B could be GEMDDDS format on the FD250,
Logical drive C could be GEMSDDS format on the FD250,
Logical drive D could be SBRAIN format on the FD250.

This would enable transfers to be carried out between the various formats with only one Pertec drive in the system, but obviously all transfers would have to be via drive A or drive M (if fitted). To attempt to "PIP" files directly between B,C and D will only result in disaster! If the transfer of the files is to be done via drive A it will probably be more convenient to do them via a CP/M user area other than USER 0.

NOTE 2: If required, SETUP can select a 96tpi drive to support a format that normally uses a 48tpi drive. In this case SETUP arranges for the BIOS to double-step the drive so that it appears to the system as a 48tpi drive. It is also possible to influence SETUPs choice of drive when requesting a format. This is done by following the format ID by a physical drive address in square brackets []. e.g. If the drive with a physical address of 0 is a 96tpi drive, then entering GEMDDDS[0] will result in it being selected to support the format (and double-stepped as a result), in place of the 48tpi drive which would have been the natural choice.

Once an acceptable format code has been entered, SETUP modifies the BIOS so that it is supported, and moves on to configure the next drive. Once all drives have been defined, (or the ESCape key is pressed), SETUP prompts with:

Modify the system track to match?

Responding with a 'Y' results in the system track being updated to match the current BIOS. 'N' leaves it alone, in which case the current configuration will only remain valid until the RESET button is pressed. (Remember typing 'Y' is NOT equivalent to doing a "SYSGEN" of the system track. Only the changed parts of the BIOS are written to the disk in response to this, not the entire CCP/BDOS/BIOS).

During the construction of a new BIOS the message:

Workspace Overflow - requires a smaller system

or Sector translation table overflow

may occur. The former occurs because as new drives are added to the BIOS SETUP allocates workspace at the end of the BIOS for the appropriate space allocation bit-map and directory check area. The larger the capacity of the drive added, the more space required. Each time SETUP allocates workspace it checks its pointer to the top of the BIOS to ensure that it has not wrapped round to 0. In the event of it doing so the above message appears and the BIOS configuration ceases. All drives defined upto (but not including) the current drive remain defined.

The latter message occurs when the area of memory reserved in the BIOS for the sector translation table overflows. Currently SETUP creates an individual table for each drive (when required), and so a system with two 8" drives would use up 52 bytes of the table area. A total of 168 bytes is reserved for the table.

6. WHIG

WHIG, (What Have I Got?), is a simple program that just displays the current system configuration at the top of the screen in a format similar to that of SETUP. If you are working with a variety of different formats you may forget exactly what configuration you are currently running. Typing "WHIG" answers this question with a display

```
                    **** Current BIOS configuration ****
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201     (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800
```

at the top of the screen.

7. ANALYSE

ANALYSE is a program that has been written to assist in determining the values required by SETUP for the physical characteristics of a disk format. ANALYSE will attempt to determine:

* Density
* No. of sides in use
* Disk tracks-per-inch (tpi)
* Sector size
* First sector number (0 or 1)
* Sectors/track
* Index mark presence
* Gap1/Gap2/Gap3 sizes
* The order of the physical sector nos.

ANALYSE will do its best to determine the above items, but by the very nature of what it is attempting to do it may make some mistakes in interpreting the data. It is advisable to run the program several times to check that the results it gives are consistent. With 5.25" disks, it has sometimes been found helpful to try the disk in both the 96tpi and the 48tpi drives.

To get the best out of ANALYSE the disk under analysis should be a freshly formatted one, or one with only a limited amount of data on it that has not been heavily used. With a disk on which the sectors of a track have been re-written several times, the Gap1/Gap2/Gap3 figures may vary between runs of the program.

In the following sections the operation of ANALYSE will be described, with the possible pitfalls at each stage being highlighted.

7.1. Running ANALYSE

ANALYSE uses the same data file as SETUP (DRVFMT.DAT) and expects to find it on drive A. Without the data file ANALYSE will be unable to proceed.

When ANALYSE is run it first prompts for a logical drive address in a similar way to the Format-a-Disk option of SETUP. If the drive you wish to use is not currently defined as a logical drive on the system, then pressing <ESC> will result in the alternative prompt of a non-MFB system. If ANALYSE is not running under an MFB system, it will prompt for a drive type. If you use this option it will take the physical drive address and drive data from the information record stored in DRVFMT.DAT by HARDWARE. Once the drive to be used has been determined, ANALYSE asks for the disk to be analysed to be placed in the selected drive, and the drive door to be closed. Pressing <RETURN> will then start the analysis procedure.

7.2. Density

ANALYSE starts by homing the head on the drive, and then steps the head in to track 4. It sets the density to single and attempts to perform a "Read Address" command on side 0. In response to this command the controller will read (and pass back to the program) the six bytes of the next sector header

that passes under the drive head. If the controller fails to find a header, ANALYSE flips the density to double and tries again. If this read fails ANALYSE flips the density back to single and repeats. After four retries without success it will abandon all further attempts to analyse the disk.

A successful completion indicates that the correct density has been selected.

7.3. Side

If the drive is double-sided ANALYSE then attempts to do a "Read Address" on side 1 using the previously determined density. If this succeeds then ANALYSE deems the disk to be double sided. If the "side" field of the sector header is not set to "1", (as should be the case), then this fact is reported as it means that the controller will be unable to read that side of the disk. (The early GEMINI GM805 format program did not set this field correctly - a fact that didn't matter with the 1771 controller. With the 1797 the "side select compare" that occurs on a Read or Write cannot be disabled).

BEWARE: The fact that ANALYSE has found data on the other side of the disk does not necessarily mean that the format is a double-sided one. The disk may be an old disk from another computer that has been reformatted on a new system. e.g. An early Gemini double-sided disk that has been re-used on a Galaxy 2.

7.4. Tracks-Per-Inch

Next ANALYSE checks the track number field of the sector header. If the disk tpi matches that of the drive it should find a "4" there. (It started by doing a seek to track 4). If the drive TPI is 96 and the disk was formatted on an 48 tpi drive, then it will find it is positioned over track 2. Similarly a 48 tpi drive and a 96 tpi disk will result in it finding track 8. If the track number does not match any of these combinations, then ANALYSE will print a series of " ????".

7.5. First sector number

ANALYSE now attempts to read sector 0 of the track to find out if it exists. If it cannot locate sector 0 after several retries, it then attempts to read sector 1. One of the two reads should succeed, the first successful one being the first sector number of the track. In the unlikely event of ANALYSE failing to find either sector 0 or sector 1 it reports the fact and returns.

7.6. Sector size

When the read routine successfully reads the first sector, it returns a count of the number of bytes it read. This is taken to be the sector size.
CAUTION: The sector size is coded in one byte of the sector header. The controller chip can interpret this byte in one of two ways depending on the polarity of one bit in the Read command issued to it. In the BIOS this is

fixed, (set), so the size field is interpreted as either 128/256/512/1024 bytes per sector. The alternative setting gives 256/512/1024/2048 bytes per sector. It is unlikely that any format should adopt this second approach, and there is no provision for its support in the MFB.

7.7. Sectors-per-track

ANALYSE now continues to increment the sector number and re-read the disk until it cannot locate a sector. From this it can determine the figure for sectors-per-track.

7.8. Index mark, Gaps, etc

Now comes the section of the analysis that is most prone to error. ANALYSE steps the head in to an inner track in the hope of finding a virgin track that has not been rewritten since the disk was formatted. It then executes a "Read Track" command.

This command reads in an entire track from the disk to the buffer. It reads in every byte that is actually written on the track. This includes all the gaps, sector markers, sector headers, CRC check bytes, etc. In an ideal world this command would execute perfectly, but in practice this is not always the case. The data is actually written on the disk as a serial bit stream, and the controller has to synchronise to the byte boundaries within this bit stream* when reading the data back. Occasionally it does lose sync, (especially if a sector has been rewritten which introduces a discontinuity into the bit stream), but will re-synchronise to the next marker it finds.

In the following sections ANALYSE is looking for specific bytes (the markers) in the data stored in the buffer. Based on known facts and information it has already derived it looks in specific areas for them. If it fails to find the byte it is looking for, it will display the part of the buffer it is examining, tell you what it is looking for, and ask you to locate it. To ease the problem of counting the bytes between the start of the display and the wanted marker, ANALYSE numbers the rows and columns of the display in inverse video. An example is illustrated on the next page:

---

* When the disk is formatted, or data is written to it, the controller writes certain markers to the disk with code violations - it misses out certain clock transitions that should be present. When reading the data back it can get byte-synchronisation by finding these markers. The "missing clocks" code violation prevents it synchronising to any data elsewhere that might imitate the marker's bit-pattern.

```
Row numbers
!! .. .. .. ..C o l u m n   n u m b e r s .. .. .. .. .. ..
================================================================================

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
00 E4 E4 E4 E4 E4 E4 E0 00 00 00 00 00 00 00 00 00 00 00 14 A1 A1 BE 19 00
24 00 02 11 8E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 00
48 00 00 00 00 00 00 00 00 00 00 00 14 A1 A1 FB E5 E5 E5 E5 E5 E5 E5 E5 E5
72 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5
96 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5
<0 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5
>0 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5 E5

        I can't locate the sector header (FE). Enter byte offset:

================================================================================
```

Here the sector header FE has been misread as BE. You should respond with 22, the number of bytes it is away from the start of the display.

7.8.1. Index mark

First ANALYSE looks for the Index mark (FC). This may well be absent, but ANALYSE will check with you first before deciding that there is no Index mark. During this check ANALYSE will home the head, seek back to the inner track, and re-read several times if it cannot locate the mark.

NOTE: If the data is garbled at the start of the buffer ANALYSE may find what it thinks is an Index mark. It is advisable to repeat this analysis several times to see if it is consistent.

7.8.2. Gap 1

Having found (or not found) the Index mark, ANALYSE then looks for the first sector header (FE). The distance between this and the start of the buffer (or the Index mark if it is present) is taken as Gap 1.

7.8.3. Gap 2

Next ANALYSE skips the sector header and its CRC bytes (6 bytes) and starts looking for the start of the associated data block (FB). This distance gives Gap 2.

7.8.4. Gap 3

ANALYSE then skips the data block and its CRC (Sector size + 2 bytes), and looks for the start of the next sector header (FC). This gives it Gap 3.

7.8.5. Physical sector numbers

Finally ANALYSE goes back to the start of the buffer and steps through looking for the sector headers (FE), and printing out the sector numbers it finds in the headers in the order that it finds them on the track.

This completes the operation of ANALYSE. You will now be offered the option of repeating the analysis (either on the same disk or a new disk in the same drive), or returning to the CP/M command level.

8. UPDATE

The program UPDATE is a simple maintenance program for the data file DRVFMT.DAT. It can be used to sort an existing data file into alphabetical order, and can also be used to merge two data files together.

8.1. Running UPDATE

When UPDATE is run it expects to find the data file DRVFMT.DAT on the currently logged in drive. Once the data file has been located and loaded into memory the main menu will be displayed:-

```
================================================================================
      Gemini Multi-Format-BIOS Maintenance program      Version 1.0

                    Return to CP/M.....0
                    List Formats.....1
                    Sort Data File.....2
                    Merge two data files....3


                    Enter required option:

================================================================================
```

If options "2" or "3" are used, the action takes place upon the memory image of DRVFMT.DAT, and the disk file is not actually updated until option "0" is taken.

8.2. Listing the current Formats

Typing "1 <RETURN>" in the main menu will result in the data on all the current Formats being displayed on the screen. In the "full" display mode the entries are shown one at a time in the form illustrated below. The RETURN key is used to step through from one entry to the next, and the ESCape key can be used to return to the main menu.

```
================================================================================
      Gemini Multi-Format-BIOS Maintenance program     Version 1.0

                              Formats

        Format name.....: IBM3740
        Format description...: Standard SD 8"

Density (S/D).....: S     Invted data (Y/N).: N     Reserved tracks....: 2
Drive type (8/5/W): 8     Index mark (Y/N)..: Y     Logical secs/track.: *
Tracks-per-inch...: 48    First side........: 0     First sector no....: 1
Tracks-per-disk...: 77    Side change (T/S).: S     Phys. sector skew..: 0
Sides-per-disk....: 1     Block size (k)....: 1     Gap 1 length.......: 32
Sectors-per-track.: 26    Extent mask.......: *     Gap 2 length(34/43): 17
Bytes-per-sector..: 128   Directory entries.: 64    Gap 3 length(>34)..: 33

Physical sector numbers: 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26

Sector translate table : 01 07 13 19 25 05 11 17 23 03 09 15 21 02 08
14 20 26 06 12 18 24 04 10 16 22

                        Type <RETURN> to continue

================================================================================
```

If only a brief summary is required a 'Q' (for Quick) should be appended to the "1". (i.e. type "1q"). In this case only the format name and format description fields are printed, one per line, without any pause until the list is complete. If there is insufficient space available in the display area to show all the formats, then the cursor-up and cursor-down keys may be used to scroll the list of formats up and down through the display area.

```
==========================================================================
     Gemini Multi-Format-BIOS Maintenance program     Version 1.0

                              Formats

ALTOS DJ   Altos mini                  ALTOS5-5   Altos mini
ALTOSDD    Altos 8" DD                 ALTOSNS    Altos non-standard
BBC        BBC - non CP/M              COMPUGRA   MCS Typeset NOT CP/M
CROMSDSS   Cromemco SD/SS              DEC        DEC Rainbow & Prof.
DECVT180   DEC VT180 format            DIGICO     Prince
EPSON/QX   Epson QX10                  FTS SG     Single sided (Spec)
GEMDDDS    Gemini 48tpi DD             GEMQDDS    Gemini Galaxy DS
GEMQDSS    Gemini Galaxy SS            GEMSDDS    Gemini 48tpi SD
IBM3740    Standard SD 8"              IBMPC      IBM PC CP/M86
IBMS34     8" Double density           ICL/8801   ICL Vers CP/M 86
IDS 2009   from Holland                ITT3030    From Atebo
KAYPRO     from Holland                M2200 DD   BT MERLIN Doub. dens
M2200 QD   BT MERLIN Quad Dens         MCCOMBO    5" Booting disc
MCS-TYPE   MCS Typeset Not CP/M        MEMDDDS    Memory 4 but EXM=0!
MIMI803    MIMI 803                    NASCOMDS   Nascom QD/DS
NASCOMSS   NAscom QD/SS                NCRDM5     NCR Decision mate V
ORION DS   Orion CP/M 86               ORION SS   Orion CP/M 86
                Use Cursor keys to Scroll - RETURN to exit
==========================================================================
```

Fig. Abbreviated Format Display

8.3. Sorting a Data file

Typing "2" in the main menu will select this option, and results in a an alphabetic sort being performed on the contents of the data file. If only a few changes have been to the file, then the sort will be rapid. If the data file is totally disorganised, then the sort can take several seconds to complete.

8.4. Merging two files

Typing "3" in the main menu will select this option, and will result in a prompt for the name of the data file to be merged with the existing file. This should be entered in the conventional manner:- e.g. B:DRVFMT.DAT or NEWDATA.DAT. UPDATE will open the file and read it into a memory buffer.

In the merge operation that follows, only those records in the second file that hold information on Formats are used. The Configuration record, and any Drive records are ignored. UPDATE takes the format records from the second file, and compares them one-by-one with the existing data file. There are three courses of action taken by UPDATE during this comparison:-

i) If an identical entry already exists in the data file, then the entry from the new file is ignored.

ii) If no entry already exists in the data file with the same "Format Identifier", then the record is appended to the file.

iii) If an entry does have a matching "Format Identifier", but differences are found in the remaining fields, then the merge stops, and UPDATE will alternately display the entry from the main data file, and the entry from the additional file. By looking for those data fields that are changing on the display, the areas of disagreement can be found, and a decision can be made on which version to accept.

The alternating display will continue until the ESCape key is pressed, at which point the display will "freeze", and the prompt "Press RETURN to continue" will appear at the bottom of the screen. Pressing RETURN will then select that version of the format for entry in the data file, and the merging process will continue. (i.e. pressing the ESCape key at the appropriate time will select a particular version of the conflicting entries. This is then displayed as confirmation, and the RETURN key allows the program to continue.)

When the merge is complete, the "sort" option is automatically invoked to tidy up the new version of the data file, following which UPDATE returns to the main menu.

9. HARDWARE

The program HARDWARE is used to enter details of the disk drives connected to the system. This data is stored in a disk file (DRVFMT.DAT) which should be present on drive A whenever HARDWARE is run.

9.1. General

You should **NOT** normally need HARDWARE. It is only required if you should ever change the drives physically connected to your system. SETUP uses the entries created by HARDWARE to find out what is available to it, and how the drives must be handled. (e.g. the stepping rate to use). If a format only requires a single sided drive, then SETUP, although it may select a double-sided drive from those connected to the system, will treat it as a single-sided drive so far as that format is concerned. Similarly the 40-track 48tpi drive may be treated as a 35-track drive. (e.g. in the SUPERBRAIN formats).

9.1.1. Data input and defaults

Most command input to HARDWARE is obtained via a buffered line input routine. i.e. HARDWARE does not act on your reply until the RETURN key is presssed. In some instances (shown below) HARDWARE will display on the current line the default input value that it will assume if nothing is typed in except RETURN. Note that even if you only wish to change one character of the value displayed, the correct value must be TYPED IN FULL as this method of input does not support on-screen editing. For confirmation HARDWARE redisplays the value it has accepted when RETURN is pressed.

9.1.2. Error messages

In the event of an error occuring, either in the response you have entered (e.g. illegal character), or as a result of that entry or an earlier entry, an error message will flash briefly at the bottom of the screen before the program continues. It may request you to re-enter the erroneous value(s), or may return to the main menu depending on the circumstances surrounding the error.

9.1.3. Aborting a Command

You may abort the current operation at any time by typing ESC (the ESCAPE key) while replying to the prompt for input by HARDWARE. e.g. If you have selected option 3, (List current drive types), you may terminate the listing at any time by pressing "ESC" in response to the prompt "Type <RETURN> to continue".

9.2. Main Menu

On running HARDWARE it will display the initial menu:

```
=========================================================================
        Gemini MFB - Hardware data entry program        Version 1.0
 A = R0201       B = GEMQDDS     C = GEMDDDS     D = M2200 QD   E = IBM3740
(W)  R0201      (1)  1015F-VI   (2)  FD250      (1)  1015F-VI  (3)  SA800

          Return to CP/M..............0
          Enter a new drive type.......1
          Enter a new configuration....2
          List current drive types.....3
          List current configuration...4

          Enter required option :


=========================================================================
```

The top three lines of the display will remain in place throughout the execution of the various options within HARDWARE, and show the current configuration of the MFB.

The top line of the display gives the version number of HARDWARE. The second line shows the formats supported by each drive, and the third line shows the drive physical address (in brackets) followed by the name of the supported drive. If a drive is undefined its entry remains blank.

If the running system does not incorporate the MFB a message to that effect will be displayed in place of the format/drive information on the second and third lines.

Enter a number between 0 and 4 followed by RETURN to perform the operation you require.

9.3. Entering a new drive type

Typing "1 <RETURN>" in the main menu will select this option, and will result in the following screen display:

```
=========================================================================
        Gemini MFB - Hardware data entry program          Version 1.0
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W) R0201      (1) 1015F-VI  (2) FD250      (1) 1015F-VI  (3) SA800

     Drive type definition


          Drive Id.................:
          Drive type (8/5/W).......:
          Drive description........:

          Tracks-per-inch..........:
          Tracks-per-side..........:
          Sides-per-drive..........:
          Stepping rate (ms).......:
          Settling time (ms).......:
          Head Load time (ms)......:


=========================================================================
```

The cursor will be positioned ready to accept a Drive Id (see below). If this is the first entry immediately following the selection of option "1", then the string typed in here will be compared against those already stored in the data file. If a match is found then that entry will be copied to the screen, and the displayed values will form the defaults for all the subsequent questions.

The answers required for each entry are shown below:-

Drive Id : This is an eight-character string that is used as the identifier for this particular entry in the data file. The examples supplied in the data file use the manufacturers type number for the drive. This string is forced to upper-case characters irrespective of whether upper or lower case is used on input.

Drive type (8/5/W): This is a single letter that defines the type of drive being described. Currently only 8 (for 8" drives) and 5 (for 5.25" drives) are accepted. This information is used by the BIOS to configure the disk controller appropriately.

Drive description: This is a twenty-five character string that may be used as a comment field (e.g. standard or slimline drive).

Tracks-per-inch: Enter the appropriate figure for the drive (e.g. 48 or 96). This field is used for cross-checking with a Format description.

Tracks-per-side: Enter the number of tracks per side supported by the drive. This figure is used for cross-checking with a Format description.

Sides-per-drive: Enter the number of sides available on the drive. (For floppies this will be 1 or 2).

Stepping rate (ms): Enter the track-to-track stepping rate for the drive as given in the drive manual. The BIOS will use this value when it moves the head on the drive.

Settling time (ms): Enter the time required for the head to settle following a "step" or a "seek".

Head Load time (ms): Enter the time required for the drive head to stabilise following the operation of the head load solenoid.

Once all the entries have been completed the prompt:

Save, Delete, Edit, Ignore? (S/D/E/I):

will appear at the bottom of the screen. As indicated four options are open:

S Will save the displayed entry to the data file, overwriting any existing entry with the same "Drive Id".

D Will delete any entry with a matching "Drive Id" from the data file.

E Will move the cursor back to the start of the entry, allowing any erroneous values to be changed. Note that this time altering the "Drive Id" field to match an existing entry will not result in that entry overwriting the currently displayed values.

I Will return you to the main menu without altering the data file.

9.4. Entering a new configuration

Typing "2<RETURN>" in the main menu will result in the data entry for the current hardware configuration being displayed on the screen in the form illustrated below, and the cursor will be positioned under the first entry on the display.

```
==========================================================================
       Gemini MFB - Hardware data entry program     Version 1.0
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201     (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800

                Drive type : Address

                1015F-VI       0
                FD250          1
                SA800          3
```

```
==========================================================================
```

You should now type in a new entry, or hit return to take the default value (currently displayed under the cursor). HARDWARE will not let you define two drives with the same physical address. The 'Drive type' field should match with an entry in the 'Drive type's section. (See following section). If it does not, an error message "Drive type is undefined" will appear whenever SETUP uses the configuration table to try and find a drive to match a particular format.

When the table is complete the ESCape key should be pressed to terminate the data entry phase. This will produce the message:

Save/Edit/Ignore?

at the bottom of the screen. As indicated three options are open:

S Will save the displayed entry to the data file.

E Will move the cursor back to the start of the entry, allowing any erroneous values to be changed.

I Will return you to the main menu without altering the data file.

9.5. List current drive types

Typing "3 <RETURN>" in the main menu will result in the data on all the current drive types being displayed on the screen. In the "full" display mode the entries are shown one at a time in the form illustrated below. The RETURN key is used to step through from one entry to the next, and the ESCape key to return to the main menu.

If only a brief summary is required a 'Q' (for Quick) should be appended to the "3". (i.e. type "3q <RETURN>"). In this case only the drive name and drive description fields are printed, one per line, without any pause until the list is complete.

```
=======================================================================
        Gemini MFB - Hardware data entry program     Version 1.0
 A = R0201       B = GEMQDDS     C = GEMDDDS     D = M2200 QD   E = IBM3740
(W)  R0201      (1)  1015F-VI   (2)  FD250      (1)  1015F-VI  (3)  SA800

     Drive type definition


          Drive Id.................: SA800
          Drive type (8/5/W).......: 8
          Drive description........: Standard Shugart

          Tracks-per-inch..........: 48
          Tracks-per-side..........: 77
          Sides-per-drive..........: 1
          Stepping rate (ms).......: 8
          Settling time (ms).......: 15
          Head Load time (ms)......: 30



                    Type <RETURN> to continue

=======================================================================
```

9.6. List the current configuration

Typing "4 <RETURN>" in the main menu will result in the data entry for the current hardware configuration being displayed on the screen in the form illustrated below.

```
================================================================================
        Gemini MFB - Hardware data entry program      Version 1.0
 A = R0201       B = GEMQDDS     C = GEMDDDS     D = M2200 QD    E = IBM3740
(W)  R0201      (1)  1015F-VI   (2)  FD250      (1)  1015F-VI  (3)  SA800

                 Drive type : Address

                 1015F-VI       0
                 FD250          1
                 SA800          3


                        Type <RETURN> to continue

================================================================================
```

There should be one entry for each physical drive on the system (excluding the winchester). Each entry is of the form:-

<Drive ID> <Physical address>

i.e. In the example above the SA800, (an 8" single sided drive), is connected to the system and is physically configured so that it responds to address 3 on the drive interface cable. Similarly at address 0 is a Micropolis 1015F-VI, (a 5" double-sided 96tpi drive), and at address 1 is a Pertec FD250, (a 5" double-sided 48tpi drive).

10. Entering New Formats to DRVFMT.DAT

The following section describes, in a general way, the steps to be followed in adding a new format to the BIOS. Further specific information will be found in the section describing the use of SETUP. Some understanding of CP/M disk structure and directory structure is necessary in order to follow some of the descriptions below.

To enter a new format on the system there are a variety of parameters that have to be determined. These fall into two categories: Physical and Logical. The physical parameters are those that the system needs to be able to read and write data to the disk. These are things such as the recording density, sector size, number of sectors per track etc. The Logical information is the information that the BIOS (and CP/M) need in order to read and write data to the correct areas of the disk. The latter are most easily determined from a running target system, but can be deduced with a bit of detective work given a disk from that target system.

10.1. Determining the Physical characteristics

Before the BIOS can read or write data to the disk it needs to know the following information:

Disk track density (48/96 tpi) - so the correct drive can be used.
Recording density (single/double).
Physical sector size (128/256/512/1024 bytes per sector).
First sector number. (0 or 1)
Number of sectors per track.
Number of tracks per side.
Number of sides in use (1 or 2).

and if you want to format disks as well as read and write them the following is also required:

Is an Index mark required?
What are the sizes of the inter-record gaps?
Are the sectors physically skewed on the disk?

If the format option of SETUP is not going to be used, then arbitary figures may be entered for the Index/Gap/Skew parameters.

In the absence of any information on the disk format the program ANALYSE can be run. (See section 7). Unless you are extremely unlucky this should provide you with the essential information required.

10.2. Determining the Logical parameters

Having found out how data is recorded on the disk surface, we now have to determine exactly where it is placed. Some of this information is required by the BIOS, some the BIOS passes across to CP/M for use by the BDOS. What has to be determined now is:

If the disk is double-sided, does the system use side 0 of the disk in its entirity before starting on side 1? Or does it use track n/side 0, track

n/side 1, track n+1/side 0, track n+1 side 1,....etc? The latter gives better system performance, but the former can provide compatibilty between single and double-sided formats if a hardware supplier offers systems with either single or double-sided drives.

How many tracks are reserved at the start of the disk for a CP/M system?

How many directory entries does the format allow for?

What is the "Block size" used by the system? (CP/M allocates space on the disk in fixed units or blocks. The minimum size supported is 1K, and the most likely values to be found are 1K, 2K or 4K, depending on the capacity of the disk. If the disk capacity exceeds 256k, then the minimum block size must be 2K.)

Is there a logical skew applied to the sectors, and if so what? If ANALYSE has shown that a Physical sector skew is in use, then it is probable that no logical skew is used. Conversely if no physical skew exists, (ANALYSE listed the sector numbers in correct numerical order), then a logical skew is likely to be in force.

If you have access to a system which supports the target format, then most of these parameters can be determined by doing a STAT DSK:. This results in a print out like the following: (actually the GEMINI QDDS format).

```
   B: Drive Characteristics
6304: 128 Byte Record Capacity
 788: Kilobyte Drive  Capacity
 128: 32  Byte Directory Entries
 128: Checked  Directory Entries
 512: Records/ Extent
  32: Records/ Block
  40: Sectors/ Track
   2: Reserved Tracks
```

From this we can see that:
There are 128 Directory entries.
The Block size is 32 records = 32 * 128 bytes = 4k.
There are 40 sectors per track. We already know that there are 10 512-byte **physical** sectors/track/side, which is equivalent to 40 **logical** 128-byte sectors. This implies that the format is likely to use one side of the disk completely, before starting on the second. (N.B. Note that it only implies that this is so, there exist odd formats - such as one from RAIR - where the logical to physical translation goes:

```
Track 0 == track 0 side 0
track 1 == track 0 side 1
track 2 == track 1 side 0
track 3 == track 1 side 1
.........................
.........................
```

To cater for formats where a logical track does not correspond to a physical track, the "logical tracks" field of the "Format" should be set appropriately - see section 5).

Another advantage of the "STAT DSK:" listing is that it gives the total capacity of the disk. From this figure, together with the sectors/track and the reserved tracks figures, the total number of tracks on the disk can be calculated. From this you can determine, for example, whether a double-sided 48tpi disk comes from a drive with 35 tracks per side, or 40 tracks per side. (The latter is the more modern option).

If the format is unknown some arbitary values can be entered temporarily for the parameters. e.g.

128 directory entries
2k block size
'T' toggle track - use both sides
1 Reserved track

Also, importantly at this stage, no skew table should be entered.

Once the format has been entered option 4 of SETUP should be used to add it into the current BIOS (say as drive B). Next SETUP should be exited, and a disk patching utility such as DU* should be run. This program allows us to directly examine the sectors of the disk, and using it we can search for the directory, determine the number of reserved tracks, the number of directory entries, the logical skew factor in use, and finally, the system block size.

Ideally for this work the disk should hold several files, one of which should contain a reasonable amount of ASCII data. An ASCII file considerably simplifies the task of determining the sector ordering if any logical skew has been applied.

10.3. SUPERBRAIN format example

The following section will refer to the use of the disk patching utility DU.COM to examine a Superbrain QD format disk.

Initially you need a Superbrain QD disk containing some data, and failing such a disk to hand you can use your MFB system to produce it. Start off by using SETUP to configure your MFB system so that the Superbrain QD format is supported (using the existing data table entry) as one of your logical drives. Next insert a disk into the drive and use option 3 of SETUP to format it. Finally 'PIP' a series of files across onto the disk. We want about sixteen files on the disk, including a text file.

Now let us assume we have just been presented with the disk, but have been given absolutely no information about it - the worst possible situation.

---

*DU.COM - available (with DU-V77.ASM) from the CP/M User group on VOL78
Also Sig/M (Amateur Computer Group of New Jersey) Volume 44

1. Place the disk in one of the 5.25" drives and run ANALYSE. (See section 7). This should provide you with the following information on the disk's physical characteristics:

   Double density.
   48tpi
   Double-sided.
   First sector is 1.
   10 sectors per track
   No Index mark
   Gaps of 44/38/35 (Gemini table)
   No physical sector skew

   One still unknown item is whether the Superbrain has 35-track drives, or 40-track drives.

2. Run SETUP and use option 1 to define a format - say TRIAL. Enter the data supplied by ANALYSE on the disk format. For the various other parameters enter some arbitary values, but DO NOT enter a sector skew table.

```
==========================================================================
       Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201      B = GEMQDDS    C = GEMDDDS    D = M2200 QD   E = IBM3740
(W)  R0201      (1)  1015F-VI  (2)  FD250     (1)  1015F-VI  (3)  SA800
            Format Definition

       Format name.........: TRIAL
       Format description...: Attempt at Superbrain

Density (S/D).....: D    Invted data (Y/N).: N    Reserved tracks....: 1
Drive type (8/5/W): 5    Index mark (Y/N)..: N    Logical secs/track.: *
Tracks-per-inch...: 48   First side........: 0    First sector no....: 1
Tracks-per-disk...: 40   Side change (T/S).: T    Phys. sector skew..: 0
Sides-per-disk....: 2    Block size (k).....: 2    Gap 1 length.......: 44
Sectors-per-track.: 10   Extent mask........: *    Gap 2 length(34/43): 38
Bytes-per-sector..: 512  Directory entries.: 128  Gap 3 length(>34)..: 35

Physical sector numbers:  1  2  3  4  5  6  7  8  9 10


Sector translate table : None


==========================================================================
```

3. Save the entry and then use option 4 to set up a new BIOS and define drive B as TRIAL format.

4. Exit SETUP.

Before running DU it is advisable to draw up a logical-to-physical sector translation table as an aid in locating exactly which physical sector DU is currently displaying on the screen. DU works in terms of CP/M logical sectors of 128 bytes each, and starts each track at sector 1. So for the Superbrain QD format the table would look like:

Logical:Physical	Logical:Physical	Logical:Physical
1 : 1	15 : 4	29 : 8
2 : 1	16 : 4	30 : 8
3 : 1	17 : 5	31 : 8
4 : 1	18 : 5	32 : 8
5 : 2	19 : 5	33 : 9
6 : 2	20 : 5	34 : 9
7 : 2	21 : 6	35 : 9
8 : 2	22 : 6	36 : 9
9 : 3	23 : 6	37 : 10
10 : 3	24 : 6	38 : 10
11 : 3	25 : 7	39 : 10
12 : 3	26 : 7	40 : 10
13 : 4	27 : 7	
14 : 4	28 : 7	

with sectors 41-80 being the corresponding physical sector numbers, but on the other side of the disk.

5. Run DU.

6. Log in Drive B. (LB)

7. Move to the start of the disk. (T0;S1)

8. Start displaying successive sectors until the first directory sector is found. (D;+;/)
(^S can be used to pause the listing, ^C to stop the search loop).
The display doesn't make any sense! No reasonable ASCII (like file names) ever turns up in the right hand side of the display, and the one important point to notice is that unwritten sectors display as a series of 1A bytes, rather than usual CP/M E5s. In fact 1A is the logical complement of E5, and so the data must be recorded in an inverted form on the disk. So..

7. Exit from DU (X) and run SETUP again.

8. Select option 1 and type in TRIAL to reload the trial format. Change the "Invted data (Y/N)" field from N to Y. Save the changed format. Use option 4 again to write the corrected information into the BIOS. Exit from SETUP.

9. Run DU again (as in step 6). This time the start of the directorysector should be found. Note down the track and sector number of it. (It should be Track 2, sector 1). This tells us that there are two reserved tracks.

At this point it may be possible to determine the sector skew by examining the directory area. The first track (ignoring the system tracks), should contain the directory, and probably the start of the first file on the disk. There are four directory entries per logical sector, so with a physical sector size of 512 bytes we need 17 directory entries to ensure that at least two sectors of the directory are in use. Because there are four logical sectors to one physical sector we know that, for example, if logical sector 1 holds directory information, then so do logical sectors 2,3 and 4 as they are part of the same physical sector. So...

10. Starting at track 2 sector 1, display the contents of all sectors on the track, (T2;S1; then D;+;/80), and note down the contents of each physical sector (block of four logical sectors). Use the ^S (Control/S) key to pause and to restart the screen output while doing this. Your list should look like this:

	Physical Sector	Contents
	1	Directory
	2	Part of a file
	3	End of current directory
	4	Part of a file
	5	Unused (E5s)
	6	Part of a file
	7	Unused (E5s)
	8	Part of a file
	9	Part of a file
	10	Part of a file
	11	Unused (E5s)
	12	Unused (E5s)
	13	Unused (E5s)
	14	Unused (E5s)
	15	Unused (E5s)
	16	Unused (E5s)
	17	Unused (E5s)
	18	Unused (E5s)
	19	Unused (E5s)
	20	Unused (E5s)
Track 3	1	Part of a file

From this we can deduce the following:
By following the directory entries from the first to the third physical sector the sector skew must start 1,3,5,7,9,2..... The data file starts at sector 9, so with the skew implied above, the directory occupies 4 sectors, giving a total of 4 * 512 / 32 directory entries (=64). Also note that sectors 11-20, which lie on the second side of the disk, are unused. This implies that the format uses all of side 0, before starting on side 1. (If the disk holds a lot of files, then data will be found on the second side. In this case an ASCII file should be located, and it is a matter of determining whether the text appears amongst the data on the both sides of the disk, or only on several tracks of one side of the disk.)

11. Locate the ASCII file on the disk, and follow the text from one physical sector to the next to confirm that you have deduced the correct skew from the directory.

12. Go back and display the first directory sector (T2;S1;D). The display should look something like:

```
G=05:00, T=2, S=1, PS=0
00  00504950 20202020  20434F4D 0000003A  *.PIP     COM...:*
10  01020304 00000000  00000000 00000000  *................*
20  00535441 54202020  20434F4D 00000029  *.STAT    COM...)*
30  05060700 00000000  00000000 00000000  *................*
40  00442020 20202020  20434F4D 00000017  *.D       COM....*
50  08090000 00000000  00000000 00000000  *................*
60  00534554 55502020  20444F4D 01000080  *.SETUP   DOC....*
70  0A0B0C0D 0E0F1011  12131415 16171819  *................*
```

From this you can see that the first file starts at block number 1. (See Appendix C for a description of the CP/M directory format). In step 10 we determined that the directory consisted of four physical sectors, so as the directory only occupies one block, (block 0), the block size must be 512 * 4 = 2k.
NOTE: If the disk had been in use for a while, the file that used the first available block number might not be the first entry in the directory. The "M" (map) command of DU can be a help here, as it displays the block numbers in order and shows which files occupy which blocks. However this command will only work properly if the data on the format that was used to set up the BIOS was correct, (or nearly correct), so at this stage this may or may not produce a useful listing.

13. Finally try to read a sector from a track >34. (e.g. T37;S1). This read should fail, indicating that only 35 tracks exist on the disk. (N.B. strictly this depends on the past history of the disk. At some time in the past it may have been used with a different format that was similar, but used 40 track drives. It is to answer questions like this, that the STAT DSK: information is useful.)

14. Now exit from DU (X) and go back into SETUP. Modify the data entry for TRIAL so that it now lines up with the information you have discovered. (See next page.)

15. Use option 4 to set up the BIOS again, this time using the correct parameters.

16. Exit SETUP and try DIR B:. If all is well you should see a correct directory listing without any strange occurrences (like blank files, multiple entries for the same file, or rubbish).

17. Finally TYPE the text file to the screen as a check, and try out (with care!) some of the COM files on the disk.

```
================================================================================
       Gemini Multi-Format-BIOS Setup program     Version 2.5
 A = R0201      B = TRIAL      C =            D =           E =
(0)  R0201     (2)  FD250
            Format Definition

       Format name..........: TRIAL
       Format description...: Attempt at Superbrain

Density (S/D).....: D    Invted data (Y/N).: Y    Reserved tracks....: 2
Drive type (8/5/W): 5    Index mark (Y/N)..: N    Logical secs/track.: *
Tracks-per-inch...: 48   First side........: 0    First sector no....: 1
Tracks-per-disk...: 40   Side change (T/S).: S    Phys. sector skew..: 0
Sides-per-disk....: 2    Block size (k)....: 2    Gap 1 length.......: 44
Sectors-per-track.: 10   Extent mask.......: *    Gap 2 length(34/43): 38
Bytes-per-sector..: 512  Directory entries.: 64   Gap 3 length(>34)..: 35

Physical sector numbers:  1  2  3  4  5  6  7  8  9 10


Sector translate table :  1  3  5  7  9  2  4  6  8 10


================================================================================
```

Fig. Final version of Superbrain Format

NOTE. DU was used initially with NO logical skew set in the data file. This was done as a logical skew can be confusing at that stage. If it had been set correctly then everthing would have been fine, but if a mistake had been made it can be quite difficult to work out what physical sector you are actually looking at, and how the table should be modified to correct the error.

APPENDIX A

A. Incompatible formats

The Gemini GM829 disk controller card, (the basis of the MFB system), uses the Western Digital WD1797 floppy-disk-controller. The 1797 will handle most of the CP/M formats currently in use, but there are exceptions, which are detailed below.

A.1. Hard sectored disks

The controller cannot read or write hard sectored disks (e.g. North Star Horizon).

A.2. Non-standard recording technique

The controller cannot read or write disks that do not use FM or MFM encoding, together with the appropriate standard for header records and CRC polynomials. (e.g. APPLE disks cannot be read/written).

A.3. Unique hardware

The MFB does not support specialist hardware requirements. (e.g. The SIRIUS records disks at a constant bit density, and achieves this by linking the rotational speed of the drive to the track selected. By doing so it fits more sectors onto the outer tracks of the disk, but they can't be read in conventional constant speed drives.)

A.4. CP/M "compatible" formats

Some systems run CP/M compatible operating systems. The MFB will only be able to support these systems if they use the same directory structure as CP/M. For example the TORCH operating system, though billed as 'CP/M compatible', uses a totally different disk structure, and so cannot be supported.

A.5. Asymmetric track or sector ordering

The MFB does not currently support any format where the method of determining the next logical sector varies from track to track. So far two examples have been found:

Alphatronic Double-sided where the logical progression (expressed as track/side) is 0/0 1/0 0/1 1/1 2/0 2/1 3/0 3/1 ..... Here they have used an "S" type access on the first two tracks, followed by a "T" type access for the remainder of the disk.

Cipher where the order of the sector numbers in the sector skew table changes from track to track. (By doing this Cipher get a slight increase in performance by making an allowance for the track-to-track stepping time of the disk drive, but from a software point of view it would have been simpler if this feature had been incorporated into their Format program, and excluded from the BIOS).

APPENDIX B

B. Incorrect CP/M formats

Several formats have been encountered where the disk parameter block has been set up incorrectly. (The error lies in the extent mask.) The net effect of this is to under utilise the directory entries for large files. Provision has been made in the format data tables used by SETUP to ensure that, if necessary, SETUP can construct an identical parameter block to that used by target system.

In the examples encountered so far, the extent mask has been set to 0 when a figure of 1 should have been used. Some visible symptoms of this error are:

Disk from target system (EXM=0) running in MFB system with EXM=1	Disk written by MFB with EXM = 1 running in target system with EXM=0
In response to DIR command: File appears twice if size>16k.	In response to DIR command: File does not appear if size>16k

.....Files over 16k will not be copied correctly.....

APPENDIX C

C. CP/M directory structure

Each CP/M directory entry consists of 32 bytes. These are used as follows:

byte

0 Set to E5 if there is no valid entry, otherwise holds the User number (00-1Fh) that the file was created under.

1-8 Hold the first part of the file name in ASCII.

9-11 Hold the extension of the file name (e.g. COM). If the $SYS or $R/O attributes are set, then these are reflected in the most significant bit of bytes 9 and 10

12 Holds the largest extent number of the entry in bits 4-0.

13 Not used.

14 Holds the 4 high order bits of the extent number in bits 3-0.

15 Holds the number of records (sectors) in this extent. (0-128)

16-31 Holds the numbers of the disk blocks occupied by the file. If the total disk size is >255 blocks, then a maximum of eight 16-bit block numbers can be held. If the total disk size is <256 blocks, then a maximum of sixteen 8-bit block numbers can be held.

e.g. Entry from a system with a 2k block size and <256 blocks.

```
00 57 53 20 20 20 20 20  20 43 CF 4D 01 00 00 0A  *.WS      COM....*
02 03 04 05 06 07 08 09  0A 00 00 00 00 00 00 00
```

File WS.COM, in User Area 0, $SYS attribute set, extents 0 & 1 open, occupying blocks 2->0Ah. Total size is 128+10 sectors. (128 records in extent 0, 10 records in extent 1).

Same file on a system with 4k block size and >256 blocks.

```
00 57 53 20 20 20 20 20  20 43 CF 4D 01 00 00 0A  *.WS      COM....*
10 00 11 00 12 00 13 00  14 00 00 00 00 00 00 00
```

details as above except blocks utilised are 10H->14H.