

Appendix D - ROM Structure

The information in this appendix is included for the more technically oriented user of the EPROM Programming Kit. If even more technical information is required, contact your GRiD Representative.

The ROM header is always located in the last 256 bytes of the ROM. When the ROM is mapped in, the header is located at segment 9FF0h in memory, regardless of the size of the ROM. A 128 KB ROM starts at memory location 8000h (just above the 512KB of system RAM). A single 64 KB ROM starts at memory location 9000h. A single 32 KB ROM starts at memory location 9800h. A diagram of ROM mapping is given in Figure D-1.

It is possible for one set of files to span multiple ROMs. In the case of the Compass and the GRiDCase external ROM sockets, 64 KB EPROMs are the largest capacity EPROMs that can be used. Two 64 KB EPROMs (or four 32 KB EPROMs) can be mapped into memory as a package, occupying the entire 128 KB of memory available for ROMs. That mapping is shown in Figure D-2.

D-2 Programming EPROMs for GRiDCase and Compass

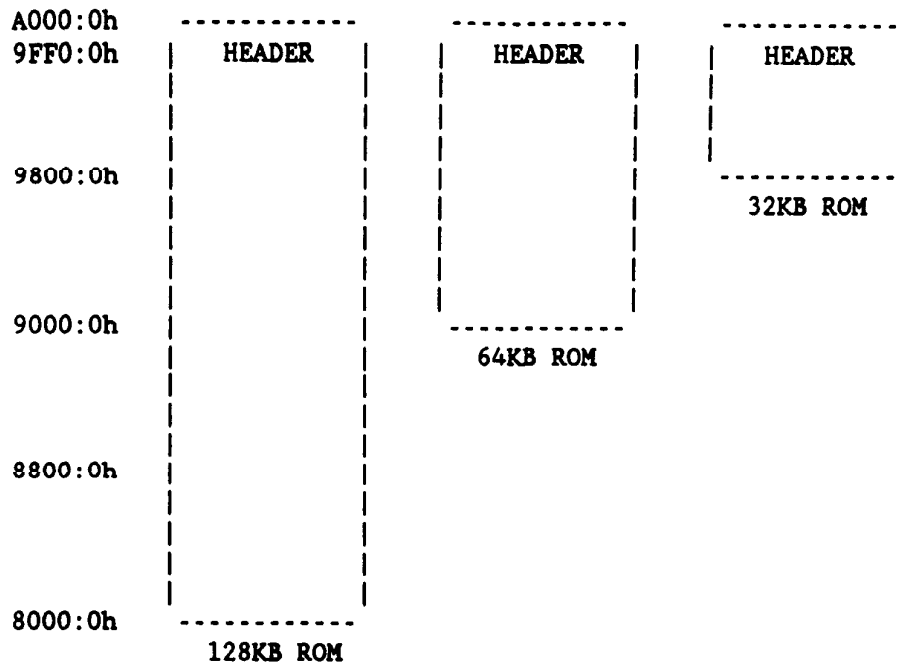


Figure D-1. Memory Mapping of a Single ROM

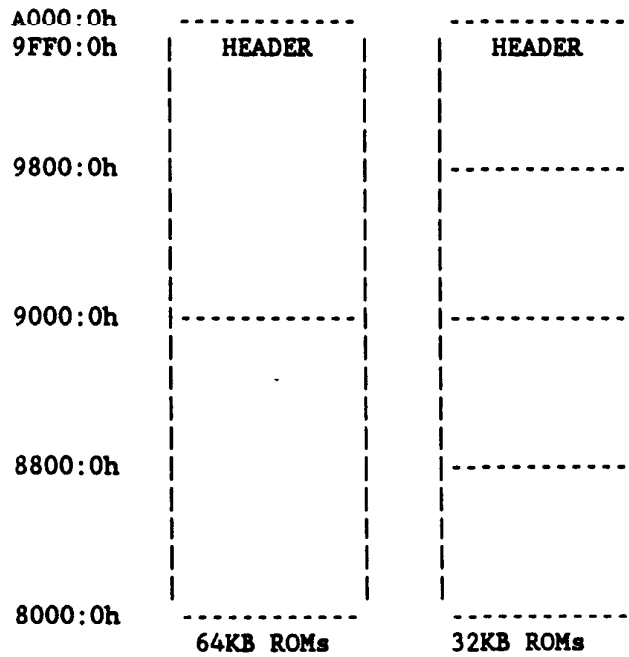


Figure D-2. Memory Mapping of a ROM Package

Note: There is only one ROM header per package.

A single large file or group of files that exceeds 128 KB is stored in multiple ROMs, known as a group. Each ROM in the group has its own

header. Groups are available under MS-DOS only.

ROM Header

The ROM header contains three kinds of information: identification fields, bootstrap fields and directory fields. The organization of the data in the ROM header is given in Figure D-3. The "ID", "BOOT", and "DIR" columns indicate what kind of information is contained in the field. A more complete description of these field types is given in the three sections following this table.

ADDRESS	OFFSET	FIELD NAME	LENGTH	ID	BOOT	DIR
9FF0:0h	00h	romHereFlag	2	*		
9FF0:2h	02h	sysType	1	*		
9FF0:3h	03h	systemRom	1		*	
9FF0:4h	04h	bootSector	2		*	
9FF0:6h	06h	romAddr	1	*		
9FF0:7h	07h	romSize	1	*		
9FF0:8h	08h	waitStates	1	*		
9FF0:9h	09h	dirSelector	2			*
9FF0:Bh	0Bh	dirLength	2			*
9FF0:Dh	0Dh	numFiles	2			*
9FF0:Fh	0Fh	pageZeroSelector	2			*
9FF1:1h	11h	totNumPages	2			*
9FF1:3h	13h	romId	2	*		
9FF1:5h	15h	copyRight	45	*		
9FF4:2h	42h	time	11	*		
9FF4:Dh	4Dh	partNumber	9	*		
9FF5:6h	56h	partName	15	*		
9FF6:5h	65h	bootLength	1		*	
9FF6:6h	66h	bootMessage	30		*	
9FF8:4h	84h	bootId	1		*	
9FF8:5h	85h	numRomsInPkg	1	*		
9FF8:6h	86h	romSumsArray (4)	8	*		
9FF8:Eh	8Eh	romIdText	50	*		
9FF8:Fh	8Fh	unused	60			
9FFF:Ch	FCh	numRomsInGroup	1	*		
9FFF:Dh	FDh	groupSequenceId	1	*		
9FFF:Eh	FEh	patchCodeFlag	1	*		
9FFF:Fh	FFh	checksum	1	*		

Figure D-3. ROM Header

D-4 Programming EPROMs for GRiDCase and Compass

ROM Identification Fields

romHereFlag	Set to BB66h to indicate presence of ROM header.
sysType	Identifies the operating system the ROM is designed for. A value of 0 indicates a GRiD-OS ROM, and a value of 1 indicates an MS-DOS ROM. Values up to 80h are reserved for GRiD use.
romAddr	Set to zero (0).
romSize	Size of ROM in 1 KB units (32, 64, 128).
waitStates	Set to three (3).
romId	A 2-byte value that uniquely identifies this ROM. Values 0 through 5000h are reserved for GRiD use.
copyRight	A 45 character ASCII field for copyright information.
time	An 11 byte field that contains a time/date stamp identifying when the ROM was made. The bytes are defined as follows: 0,1 year 2 month 3 day 4 hour 5 minutes 6 seconds 7 hundredths of seconds 8 day of week (not used) 9,10 day of year (not used)
partNumber	A 9 byte ASCII field that contains the part number. GRiD part numbers are formatted as 'nnnnnn-nn'.
partName	A 15 byte ASCII field that contains " <- Part Number".
numRomsInPkg	If the total ROM size is less than 128 KB and the hex file spans multiple ROMs, then the ROMs are known as a package. This is the number of ROMs in the package.
romSumsArray	A four word array that contains the individual checksums of the ROMs in the package.
romIdText	A 50 byte string identifying this package.
numRomsInGroup	If the total ROM size is greater than 128 KB, then the ROMs are known as a group. This is the number of ROMs in the group. (MS-DOS only.)

groupSequenceId	For a group, each ROM has its own header. This field indicates the sequence number of this ROM in the group. (MS-DOS only.)
patchCodeFlag	Always false.
checksum	A single byte checksum of all of the ROMs represented by this header (a package).

ROM Bootstrap Fields

systemRom	A value of 80h identifies a bootable ROM.
bootSector	A word value to add to 8000h to create a segment address of the boot sector.
bootLength	Length of the boot message.
bootMessage	A 30 byte ASCII message. This message will only be displayed if there is more than one bootable ROM available. In that case the user is prompted with the boot messages.
bootId	Integer number the user would enter to select between bootable ROMs. For the convenience of the user the boot message should indicate what this number is.

ROM Directory Fields

dirSelector	A word value to add to 8000h to create the segment address of the first directory sector.
dirLength	Number of paragraphs in directory.
numFiles	Number of directory entries.
pageZeroSelector	A word value to add to 8000h to create the segment address of the first data sector.
totNumPages	Number of data sectors in ROM.