Frequently Asked Questions

What development tools can I use?

The ROM-DOS kit contains a Borland 5.2 C/C++ compiler and the Borland assembler. ROM-DOS building tools and source files are all ported to work with the Borland tools provided. The compiler and assembler also allow development of C, C++ and Assembly program to be run on ROM-DOS. Applications can be remotely debugged, on the actual target hardware, using the Borland TD-Remote debugger. DOS applications generated with other compiler tools can be run on ROM-DOS.

Can I ROM the ROM-DOS Kernel?

Yes – The ROM-DOS Kernel can be loaded into and run directly from ROM. There is a special BUILD application that eases this process.

Can I place Applications in ROM?

Yes – ROM-DOS's ROMDISK utility lets you place applications, configuration files and static data files directly into ROM. ROMDISK.EXE generates either binary image files or Intel Hex files suitable for EPROM programmers and memory download utilities.

Can my Applications run directly from ROM?

Yes – ROM-DOS has extended ROMable EXE program support that allows the applications code to run from ROM in side a ROM disk. The RXE feature has been used with high volume consumer products

Can I retrofit DOS applications with Internet Connectivity?

Yes – The Internet connectivity suite can be used to make current applications remotely managed and connected to the Internet! Applications can be made to speak directly with the SOCKETS TCP/IP stack through the various APIs provided or pre built applications such as the ftp client and server can be added for a fast drop in solution.

Do I need compiler tools to use ROM-DOS?

If you plan to boot from a disk (floppy, hard, SSD...), the bootable files provided with the Development Kit may be exactly what you need. If you plan to customize the ROM-DOS kernel, boot from ROM, or change a piece of provided source code, you will need compiler tools. Please note that your own application program does not have to use the same compiler tools as ROM-DOS's configuration utilities.

For ROM-DOS, you can use either Borland C, Turbo C, or Microsoft C tools. Borland versions 2.0 through 5.x are supported. Microsoft C versions 5 and up are supported, including Visual C. The BUILD.EXE utility used to configure the ROM-DOS kernel, requires only an assembler and linker (TASM and TLINK for Borland or MASM and LINK for Microsoft). Please note that some compiler packages do not contain an assembler so it must be acquired separately.

The development kit does come with source code for various memory disks, miniCMD, and internal device drivers. The use of the C/C++ compiler tools (BCC or CL) will be required to prepare most of this source code should you decide to customize it. Please remember when using a Windows compiler environment, care must be taken to be sure you produce a DOS executable rather than a Windows executable.

Will ROM-DOS work with Pascal, Basic, Quick C ...EXE's?

Yes. A standard DOS .exe file created with any language will work with ROM-DOS and can be placed on a ROMdisk in ROM. With the exception of hardware dependent code (including excessive memory usage), if the application can run from DOS on your desktop, it can run with ROM-DOS from ROM!

How does it compare with MS-DOS?

ROM-DOS is a highly compatible operating system to MS-DOS. ROM-DOS is all original code, designed specifically for the embedded systems market. It is smaller in size than MS-DOS and the per copy royalty is often considerably less, depending on quantity. ROM-DOS's automated BUILD utility makes it easy to put the operating system into ROM.

Is ROM-DOS really compatible: What doesn't run on it?

ROM-DOS is very compatible. It is run daily at Datalight, running applications such as Microsoft Windows, database and spreadsheet software, word processing and publishing software, network applications, mail software and much more. ROM-DOS can be found in hundreds of diverse products around the world! It is also run through a rigorous testing process before each release. Datalight's promise is that if you find an application that runs with MS-DOS and not with ROM-DOS, let us know, and we will make sure it does run in a future release.

Can I get an evaluation kit?

We have a Demo diskette that allows you to boot ROM-DOS on your PC and run applications. Datalight also has prepared evaluation kits for many popular hardware platforms. These kits are offered free of charge to qualified customers. We also offer a 90-day money back guarantee on the full Developer's Kit. We can even send it with an invoice due in 30 days so you can try it completely free and without risk for 30 days, and then you have an additional 60 days to decide if this is the right tool for you.

What does the demo disk do?

It is a bootable version of ROM-DOS so you can boot it on your PC and try out applications. It also explains some of the features of ROM-DOS and Datalight's other products.

It's smaller? What's missing?

ROM-DOS's code was designed to be smaller. You will find all the standard DOS desktop utilities needed for an embedded system. Some utilities have been refined to provide only the basics to optimize space. However, ROM-DOS is not a static product. The development team continues to add new features and support as industry requirements and technology change.

Are you licensing DOS from Microsoft?

NO! ROM-DOS has been written exclusively by Datalight. All of the code is ours. We have no association with Microsoft.

Do you have source code?

We provide source code for all our device drivers and miniCMD in our Software Developers Kit. We also can provide complete code for the ROM-DOS kernel & command processor (full source code license) for $10,000.

Does it come on a ROM or do we have to program it ourselves?

We provide all of the software and manuals, but not the ROM's. There are so many choices available for sizes and types of ROMs and Flash memory that we can't provide ready-made ROM-DOS ROMs. Our Development Kit allows you to configure ROM-DOS and ROM disks exactly how you want them. The output from our BUILD and ROMDISK utilities are binary image or Intel Hex format files, ready to use with EPROM programmers or Flash update utilities. The BUILD program can also output files ready to be SYSed onto a floppy disk or hard drive. Although we can't provide the ROM, our tools make it easy to get the job done yourself.

Does ROM-DOS work from a disk?

Yes, ROM-DOS can be booted from floppy disks, hard-drives, CD ROM drives, flash and solid state disk drives.

How does ROM-DOS boot from ROM?

ROM-DOS installs itself via a BIOS extension. ROM-DOS's BIOS extension code points Int 19h to the start of the ROM-DOS code. After the BIOS initializes and is ready to boot, it will issue an Int 19h which then jumps directly to ROM-DOS in ROM, instead of booting from the floppy or hard drive.

How do I put ROM-DOS on my system?

There are many ways to put ROM-DOS onto a system. You can SYS a floppy, hard-disk or memory card drive, or program ROM-DOS into ROM or Flash using our utilities to prepare image or hex files. You can also boot from flash or solid state disk drives by following the instructions provided with the disk or flash file manager software. Our BUILD utility allows you to configure ROM-DOS for your needs. The BUILD helps you create the type of file needed for your system.

Where do I put ROM-DOS in ROM on my system?

Because ROM-DOS boots via a BIOS extension, you must place ROM-DOS within the BIOS scan area. A standard BIOS will begin scanning for BIOS extensions starting at C000:0h and continue on 2k boundaries up to F000:0h or even as high as F800:0h. The BIOS itself is located in the 64k ROM area between F000:0h and FFFF:0h. Therefore, ROM-DOS is usually placed somewhere in the address range of C000:0h to F000:0h. Note: in some systems, part or all of the 64k memory area starting at C000:0h is reserved for video usage.

Where does the ROM disk go?

Typically a ROM disk is placed in the same address range as ROM-DOS itself. ROM-DOS's built-in ROM disk device driver searches between C000:0h and FFFF:0h for it's own ROM disk identification signature. However, because ROM-DOS is designed to be flexible, the search address range can be expanded to include lower addresses such as A000:0. ROM-DOS can also access ROM disks in extended memory and paged memory by using one of the customizable memory disk drivers provided with the full Development Kit.

I only have one ROM socket-how do I get that additional ROM space?

Much depends on the size of the ROM socket and size of BIOS. If the boot ROM is large enough to hold the BIOS, ROM-DOS and a ROM disk (if needed), you can place everything into the single ROM. If the ROM is only large enough to hold the BIOS, you

---

may need to use an add-on card. Several manufacturers provide this type of device. Another possibility is the use of paged flash memory. If this is available, ROM-DOS can be booted right off the flash drive using Datalight's FlashFX product. Only a small 15-17k FlashFX BIOS extension piece needs to be located in the ROM area.

My system won't have any physical disks, where do I put my application?

Although it is true that a DOS system needs a disk of some sort, ROM-DOS does not require a physical floppy or hard-drive. ROM-DOS can use a ROM disk to load application programs, the command processor, and other boot files. ROM-DOS has a built-in ROM disk device driver that is ready to use. The Development Kit also comes with source code examples of many special purpose memory disks such as a paged-memory disk, extended memory address disk, RAM disk, and fixed-address disk.

How do I get my application into ROM?

Standard DOS EXE files that can run on your desktop PC can easily be placed on a ROMDISK without any modification. We provide all of the utilities necessary to place ROM-DOS and your application in ROM. The ROMDISK building utility allows you to place your application onto a ROM disk. You simply create a directory, place all of the files for the disk into it (applications, command.com, autoexec.bat, config.sys, device drivers, etc.), and run the ROMDISK builder with the directory name as an argument. The output from both the BUILD utility and the ROMDISK builder can be either a binary image or Intel Hex file ready to be programmed into ROM or FLASH.

Do you have an XMS or EMS driver?

Yes, we have a HIMEM.SYS and an EMM386.EXE driver.

What are the advantages/disadvantages to RXE (ROMable executables)?

An RXE is a standard application that runs the code out of ROM rather than copying it to RAM at runtime. Its main advantage is that it saves the RAM space required for code. The downside is that ROM can be slower than RAM. Your application also must follow some basic rules such as not containing any self modifying code. You must also have the source code available to convert an EXE to an RXE. For this reason, most third-party code is not RXEable.

Is it multi-tasking?

No. ROM-DOS is compatible with MS-DOS. The standard multi-tasking kernels that run on top of DOS will work fine.

How do I debug with ROM-DOS?

The best debugging method is to use familiar tools to debug your application right on our desktop system. For target system debugging, you can use Borland Turbo Debugger in Remote mode.

Do we have International Support?

Yes, 26 countries. ROM-DOS also supports double byte characters for Asian languages through third party drivers which write the Asian characters to the screen. Contact Datalight if you need help locating these third party drivers.

Can I use the device drivers that came with my hardware?   (How do I use them? config, built-in, TSR)

Yes. You can load them as devices in config.sys or as TSR's. Datalight's memory disk devices drivers can also be built right into the ROM-DOS kernel.

Why does the price list contain options for "Sockets licensed with ROM-DOS", "Sockets licensed without ROM-DOS" and "ROM-DOS/Sockets"?

There are many possible ways that a partner or customer can obtain Sockets. It has been Datalight's policy to slightly discount the license fees when multiple Datalight products are being used together. The difference between "Sockets licensed with ROM-DOS" and "ROM-DOS/Sockets" is that "ROM-DOS/Sockets" implies that the licenses are being purchased together. A customer who already has a ROM-DOS license may wish to add Sockets. Datalight wishes to extend a discounted price to the Sockets license when it is being used with the ROM-DOS product. Thus, the three categories equate to Sockets licensed with ROM-DOS is Sockets being added to an existing ROM-DOS license, Sockets licensed without ROM-DOS is Sockets being used on a different OS, and ROM-DOS/Sockets is a combination licensed when the products are purchased simultaneously.

What is the size and footprint of the ROM-DOS kernel?

The configurable nature of the ROM-DOS kernel will make the exact footprint size dependent upon your build selections. The following char contains some common sizes based upon fairly standard build selections and default configurations using ROM-DOS version 4.11.1497:

Kernel Type	Hard Drive Space	Exact bytes	Memory Footprint	Exact bytes
7.1 with LFN	76K	77760	66K	68208
6.22 with LFN	70K	71392	60K	61872
6.22 no LFN	57K	57472	46K	48016
6.22 ROM	44K	44944	46K	48016

Further size reductions are possible by replacing the complete command interpreter (command.com) with minicmd that is configurable to the exact DOS kernel commands you wish to support. Minicmd can provide space savings up to 40K.

For more information on customizing the size and footprint of the ROM-DOS kernel please review the Datalight ROM-DOS White Papers or refer to the build information in the Datalight ROM-DOS Developer's Guide.