DALLAS
SEMICONDUCTOR

1. INTRODUCTION

The Dallas Semiconductor iButton-TMEX is a set of software drivers and utility programs to interface with the Dallas Semiconductor iButton products (Touch Memories). iButtons are high-capacity, general-purpose electronic data carriers, each with a unique registration number acting as general identification number and unique network node address. Communication to iButtons is done over a single data line plus ground reference using the 1-Wire protocol.

The drivers enable TMEX-ready iButton applications to run. The DS0621-SUL iButton-TMEX license, henceforth referred to as TMEX, supports DOS, Microsoft Windows 3.1x, Windows 95 and Windows NT environments on IBM PC-compatible desktop, laptop and handheld DOS computers.

The supported hardware interfaces for these drivers are the COM port using the DS9097(E) Serial Port Adapter as well as the LPT port using the DS1410D or DS1410E Parallel Port Adapter. The modularity of these drivers supports expansion into other hardware media. TMEX versions for platforms other than DOS and Microsoft Windows products are under development.

	Port Type	Overdrive	Strong Pullup	Program Pulse
Parallel Port				
DS1410D	2			
DS1410E	2	X	X	
Serial Port				
DS9097	1			
DS1413	1			
DS9097E	1			X
DS9097U	5	X	X	X
DS9097U-9	5	X	X	
DS1411	5	X	X	

The DS1410D and DS1410E are Parallel Port Adapter that work in the LPT port (Type 2) and do not support programming the EPROM-based iButton products. The DS1410E Adapter supports Overdrive which is a faster communication rate that

some iButton products support.This adapter also supports Strong Pullup which provides power to special purpose iButtons such as the temperature iButton.

The DS9097Xs, DS1413, and the DS1411 are Serial Port Adapters. The DS9097, DS1413 and DS9097E Adapters work in the COM port (Type 1) and do not support Overdrive and Strong Pullup. The DS9097U, DS9097U-9 and DS1411 adapters support Overdrive and Strong pullup. The DS9097E and DS9097U adapters support programming voltage. They are capable of producing a 480 microsecond 12 volt pulse on the MicroLan to program EPROM device such as the DS1982. Caution must be maintained when using programming voltage, however, since any non-EPROM iButtons on the MicroLan may be damaged during a program pulse.

IButton-TMEX can be used
- a) stand alone with the provided utilities
- b) in conjunction with other Dallas Semiconductor TMEX-based kits
- c) with third party applications that require TMEX.

Dallas Semiconductor offers an iButton-TMEX Software Developer's Kit (SDK) DS0621-SDK to develop applications for TMEX. The SDK supports DOS, Windows 3.1x, Windows 95 and Windows NT. It includes multiple source code examples in several languages and a full documentation of the TMEX API.

2. CONTENTS

This document is subdivided into three sections:

a) Windows 95 / Windows NT
b) Windows 3.1x
c) DOS

The Windows 95 / Windows NT section describes the installation of iButton-TMEX under Windows 95 and Windows NT. In addition this section describes the use of the Windows iButton Viewer. This Viewer combines almost all of the features of the DOS utilities in a single program. In addition to this, the Windows iButton Viewer supports a multi-drop environment where multiple iButtons may reside on the 1-Wire bus and are accessed independently of each other. The Windows section describes the installation of iButton-TMEX under Windows 3.1x. The Windows 3.1x version of the iButton Viewer has the same features and is operated in the same way as the Windows 95/NT version and therefore needs no separate documentation. The DOS section gives details on how to install the software correctly on the hard drive of the computer and describes the purpose and use of the iButton-TMEX DOS utilities.

3. WINDOWS 95/NT SECTION

I. Installation
General Information
Except for the installer SETUP.EXE and the two text files README.TXT and LICENSE.TXT, the files on the WINDOWS 95/NT (32-Bit) version of the iButton-TMEX installation disk are compressed. The Windows 95/NT version of the iButton-TMEX 3.00 consists of eleven release files with the release date and time code: IBVIEW32.EXE, SETPRT32.EXE, IBVIEW32.HLP, LICENSE.TXT, README.WRI, IBFS32.DLL, IB10E95.DLL, IB10ENT.DLL, IB97E32.DLL, IB97U32.DLL, DS1410D.SYS and VSAUTHD.VXD. The iButton Viewer IBVIEW32.EXE requires IBFS32.DLL which in turn calls on one of the hardware specific DLLs (IB10E95.DLL, IB10ENT.DLL, IB97E32.DLL or IB97U32.DLL) for the low level 1-Wire communication functions.

If not otherwise specified during the installation, the installer creates a directory \Program Files \Dallas Semiconductor\ on the same drive as Windows with a sub-directory \IBVIEW32\ where the first five of the release files will be copied to. The DLL and VXD files will be copied to the \Windows\System\ directory in Windows 95 and \Windows\System32\ directory in Windows NT. The system file DS1410D.SYS will be copied into the \Windows\System32\Drivers\ directory in Windows NT.

During installation, there are several key entries created in the system Registry.

The uninstall information for the iButton-TMEX 32-Bit installation is kept under key:
\HKEY_LOCAL_MACHINE
                    \Software\Dallas Semiconductor\iButton-TMEX (32-Bit)\3.00

System driver information is kept under key:
\HKEY_LOCAL_MACHINE
                    \System\CurrentControlSet\DS1410D

The driver setup information values are kept under key:
\HKEY_LOCAL_MACHINE
                    \Software\Dallas Semiconductor\iButton-TMEX\3.00

The uninstall information for the iButton Viewer32 is kept under key:
\HKEY_LOCAL_MACHINE
                    \Software\Dallas Semiconductor\iButton Viewer32\3.00

The iButton Viewer32 configuration (window size/position etc.) is kept under key:
\HKEY_CURRENT_USER
                    \Software\Dallas Semiconductor\iButton Viewer32\3.00

The program REGEDIT.EXE in Windows 95 or REGEDT32.EXE in Windows NT can be used to exam registry entries. Changing the iButton-TMEX registry entries is not recommended.

iButton-TMEX (32-Bit) comes with a provided uninstall program. This uninstaller consists of the program UNINST.EXE and a DLL. The program is automatically copied to the \WINDOWS\ directory. The associated DLL and the file DEISL1.ISU which contains control information for the uninstaller are found in the same directory as the iButton Viewer32. The uninstall program can be accessed with the 'Add/Remove Software' utility in Windows 95 (discussed below) or from the uninstall icon in the iButton-TMEX folder in Windows NT 3.51.

Before upgrading to this version of iButton-TMEX always remove any previous version with its uninstaller.   To use the uninstaller in Windows 95:

1. Click on the 'Start' button
2. Click on 'Settings' and then 'Control Panel'
3. Double-click on the 'Add/Remove Programs' icon in the 'Control Panel'
4. Select the 'iButton-TMEX (32-Bit)' line in the 'Add/Remove Programs' window
5. Click OK at the bottom of the 'Add/Remove Programs' window
6. When asked if you want to remove all of iButton-TMEX, click OK.

Welcome Screen

Before installing the Windows 95/NT version of iButton-TMEX, connect a 1-Wire adapter DS9097, DS9097E or DS9097U to the COM port or a DS1410D or DS1410E adapter to the LPT port you will be using and note the actual port number. Close any application program that is currently running.

Insert the iButton-TMEX (32-Bit) installation disk, run the Explorer / File Manager, select your 3½" disk drive, and then double-click on SETUP.EXE. After the setup program is loaded you will get the Welcome screen. During the installation you can move from screen to screen by clicking on the button "Next >" or "< Back" or cancel the installation ("Cancel").

License Agreement

Right after the Welcome Screen the Software License Agreement is displayed. The license agreement can be scrolled through on screen or, after the installation is completed, be printed (LICENSE.TXT). The next screen provides general information on iButton-TMEX and a reminder to uninstall any earlier revision of the program before proceeding with this installation.

User Information
In one of the subsequent screens you are requested to enter your personal name and the name of the company where iButton-TMEX is being installed. Please enter your data correctly and proceed.

Destination Location
In the next step the installer asks for the Destination Location where the program files are to be copied to. The default location is on the Windows drive in the \Program Files \Dallas Semiconductor\IBVIEW32\ directory. You may enter any other destination directory using DOS conventions. If the directory does not yet exist it will be created. You may instead click on "Browse..." and select an existing directory without typing.

Program Folder
Windows allows programs to be grouped arbitrarily into "Program Folders". If you accept the default shown on the Select Program Folder screen, the installer will create a new program folder "iButton-TMEX" with the icons for the iButton Viewer {32 Bit}, iButton-TMEX ReadMe {32-Bit} , the iButton Viewer Help {32-Bit} , the Default Microlan {32-Bit} utility, the Testcom serial port utility and the Testcom ReadMe. Alternatively you may create a new folder by typing its name or select an existing folder by scrolling through the list of folders.

NOTE
When adding iButton-TMEX to an existing folder in Windows NT, please be aware that, depending on other programs that are installed in the same folder, there may already be one or more identical icons for an uninstall program. Although the icons look identical, each is associated with a different program. Therefore, to avoid confusion at a later time, it is recommended the description of the uninstall icons in this folder be changed as soon as the iButton-TMEX installation is completed. The description can be changed one icon at a time by selecting the icon and then clicking on the function "Properties" of the program manager's File menu. The contents of the "Command Line" field indicate what program is associated with the icon. After the program is identified, change the description to "XYZ Uninstall", where XYZ indicates the name of the program to be uninstalled.

MicroLAN Port Selection

After the Program Folder has been selected, one has to specify which of the ports with a 1-Wire adapter connected is to be defined as the TMEX default MicroLAN port. This is done through the Port Selection Screen. It is allowed to have more than one port adapter plugged in at a time. However, only one of them can be selected as the default.

On the Port Selection Screen select the tab of the hardware adapter that is to be used. Next enter the port number where the 1-Wire adapter was attached during the preparation of the installation (see Welcome Screen). After a port with adapter has been selected, the port number, adapter type and driver information will be displayed and the user can proceed to the next screen. If a port with no adapter connected is selected, you will get an error message. In Windows NT the driver for the DS1410E parallel port adapter can not be used until after a re-boot. If a LPT port is selected in Windows NT, it will not be tested.

Setup Complete

When you reach this screen, the iButton-TMEX installation under Windows 95/NT is completed. Before you can run the iButton Viewer32 however, you must restart the computer. If you wish to restart Windows at this time, then click on 'Finish'.
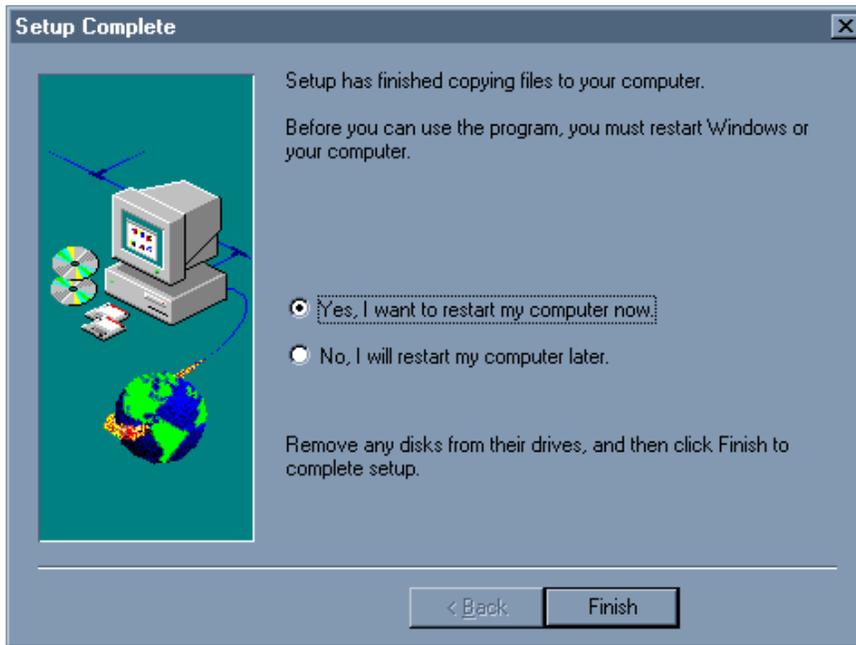
Notes

After iButton-TMEX is installed and Windows restarted, the iButton Viewer and other TMEX programs (not included in iButton-TMEX) can be used without having to run the iButton-TMEX installer again. If you change the 1-Wire port on your computer, however, the application software such as the viewer needs to be reconfigured. The iButton Viewer has a function to do this (see next section for details), but it does not change the TMEX default 1-Wire port in the registry.

If you want to change the default port selection, you can do one of the following:
a) close any TMEX-based program and run the Windows program SETPRT32.EXE by double-clicking the "default microlan" icon under the iButton-TMEX menu
b) uninstall iButton-TMEX and then re-install it with the new port selection
c) edit the registry

Option a) should be preferred since it is easy, quick and does not require any special precaution. Option b) is also a good and safe way to go, but it takes more time. Option c) should only be chosen if one is very experienced in editing the registry. If an error is made under option c) it may become necessary to uninstall iButton-TMEX and then re-install it from scratch

II. iButton Viewer

A. Introduction
The iButton Viewer is a Windows program for communicating with iButtons and other MicroLAN compatible 1-Wire devices. It integrates most of the iButton-TMEX DOS utilities in a single, user-friendly piece of software. In contrast to the iButton-TMEX DOS utilities, the iButton Viewer supports a true network environment. Device-specific viewers allow evaluation of special function devices that are either incompletely or not at all supported by the iButton-TMEX DOS utilities.

The main features of the iButton Viewer are:

- Supports MicroLAN environment with no logical limit for the number of devices on the bus
- Reads and writes Memory iButtons based on SRAM, EPROM and EEPROM technology
- Supports all currently available iButtons and their solder-mount equivalents
- Sophisticated TMEX file viewer for DS1982, DS1985, DS1986, DS1992, DS1993, DS1994, DS1995, DS1996, DS2407 with upload/download from PC to iButton and iButton to clip board
- Time register viewer for DS1994
- Secure viewer for DS1991
- Universal Data Packet viewer for DS1971
- Temperature viewer for DS1920
- Switch viewer for DS2407
- Multiple window, multiple port operation to view several devices simultaneously
- Supports Overdrive communication
- Context sensitive help

The iButton Viewer accepts generic as well as custom ROM devices. For devices currently under development the iButton Viewer will, in most cases, display the part number together with a short description. As soon as these devices become generally available, the iButton Viewer will be revised to include the necessary viewers. Until then a message "This device does not have a viewer." will be displayed. The iButton Viewer is able to read the ROM Registration Numbers of future MicroLAN-compatible 1-Wire devices. However, neither their part number nor a description can be displayed.

Not implemented in the version 3.00 of the iButton Viewer are the equivalents of the iButton-TMEX DOS utilities TMEMCOPY, TOPT and TTREE. Since the Windows and the DOS version of iButton-TMEX are independent of each other, both can be installed on the same computer without interfering with each other. This allows running the iButton Viewer for most of the time and supplementing it with the DOS utilities to copy the entire contents of an iButton (TMEMCOPY), to defragment the data structure (TOPT) or to write the outline of a device's file structure to a disk file

(TTREE). For details on the installation of the iButton-TMEX DOS utilities and their use please refer to the DOS SECTION of this document.

B. Main Screen
The iButton Viewer is launched like other Windows program by double-clicking on its icon. The main screen of this program (see marked sample below) consists of several sections, each with its special purpose. These sections are: Network Listbox (left), Part Number drop-down Combobox (top), Viewer Launching Button (lower right), Menu Bar (top) and the Status Bar (bottom).
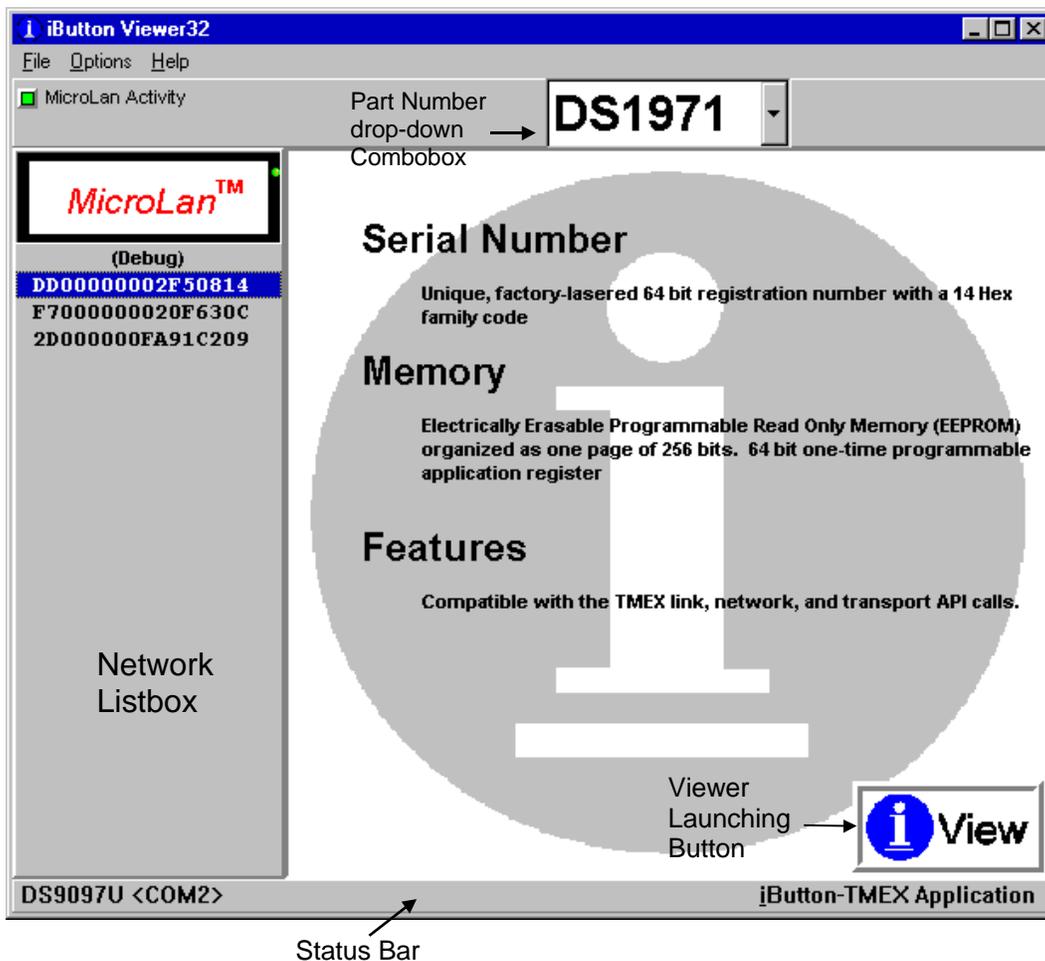
The **Network Listbox** is the primary switchboard of the iButton Viewer. It lists the ROM codes (= network node addresses) of all devices currently connected to the 1-Wire bus. One can scroll through this list with the cursor keys or click on a network node address to select a device. Depending on the number of devices connected,

the Network Listbox may have a scroll bar to allow reading all addresses and selecting a device. If no device is connected the Network Listbox will be empty.

After a device has been selected in the Network Listbox, its node address is highlighted and its part number appears in the **Part Number Combobox**. The area below the Part Number Combobox will then display some of the device's characteristics. The Part Number Combobox also may be used independently of the population of the 1-Wire bus. Clicking on the down arrow at the right of the Combobox will display the list of devices that are recognized by the program. A scroll bar allows reading the entire list. Clicking on a part number will close the box and display the device description. If there is a device with the selected part number on the 1-Wire bus, it will automatically be selected.

The next step after selecting a device is launching a device viewer. There are currently six device viewers implemented in the program. The family code of the device (i. e., the two least significant digits of the node address as displayed in the Network Listbox) determines which viewer applies. Usually the iButton Viewer selects the device viewer automatically. For some devices (such as the DS1994) there may be more than one viewer. In such a case the program will display the names of the applicable device viewers and ask the user to select one when trying to launch the device viewer.

There are several ways to launch a device viewer: a) clicking on the **Viewer Launching Button**, b) double-clicking on a network node address shown in the Network Listbox or c) pressing the right mouse button. Method a) is the direct one. Method b) combines device selection with launching the device viewer. Method c) displays a three item Listbox next to the cursor where one then has to click on "Viewer". Regardless of the method, a viewer window will pop up that is associated only to the individual device that has selected. The individual device viewers are described in detail on the subsequent pages.

The iButton Viewer Main Screen has two more sections: Menu Bar and Status Bar. The **Menu Bar** has three entries: File, Options and Help. File opens a path to exit the iButton Viewer program, which is equivalent to closing the iButton Viewer's window. Help launches the well known Windows Help Utility to access the program-specific help file.

Options opens a list with five items: a function "MicroLAN Port" to dynamically change the 1-Wire Port that is monitored by the iButton Viewer and four individual flags "Auto Viewer Start", "Auto Viewer Close", "Auto Save Size/Position" and "File Auto Overdrive". The function "MicroLAN Port" is similar to the port selection scheme used during the iButton-TMEX installation. However, the port selected with "MicroLAN Port" is valid for the iButton Viewer only. It does not alter the default selection made during the iButton-TMEX installation. The **Status Bar** shows the currently valid port name at its left side for verification.

The "Auto Save Size/Position" flag, if checked, will force the iButton Viewer to pop up always in the same window size and position it had when it was last used. The port selection function as well as the exit path are also accessible through the right mouse button. More details on port selection and a full discussion of the four option flags are found in the section "Viewer Customization/Advanced Features" later in this document.

Just below the entry "File" in the menu bar there is a blinking spot labeled **MicroLan Activity**. Whenever this spot is green it indicates that MicroLan communication such as ROM Search (see device data sheets) is in progress. An activity indicator is also found on each device viewer window. This indicator is green whenever the associated device is accessed.

C. File Viewer

The file viewer is the common tool to access the memory of any 1-Wire device with at least 1K bit of memory. It is capable of reading and writing devices manufactured in SRAM and EPROM technology. The file viewer supports the TMEX file format. Valid file names are maximum four characters long and have a numeric extension in the range of 0 to 99 (decimal). With EPROM devices it is possible to create so-called "append files". Append files can be modified by appending data at the end of existing data. This provides an economic way of updating information without re-writing the whole file, as it is normally done in file oriented systems. A file is created as an "append file" if one specifies an extension of 100.

As any advanced file system, TMEX allows organizing files into several levels of sub-directories. As a consequence, accessing a file requires two steps: first, selecting the file in the directory tree; second, displaying the file's data. If the file is new, one first has to select the directory where the file is to be created and then create it. For this reason the file viewer operates through two windows: the **Directory Window** to select/create a file and the **File Editor Window** to read/edit its contents.

Understanding The Directory Window

The Directory Window consists of three major sections:

file listbox	(left) to display file and directory names,
information area	(center) to display information on the selected file, directory and device
button area	(right) with buttons for the four most often used file functions.

File Viewer, Directory Window
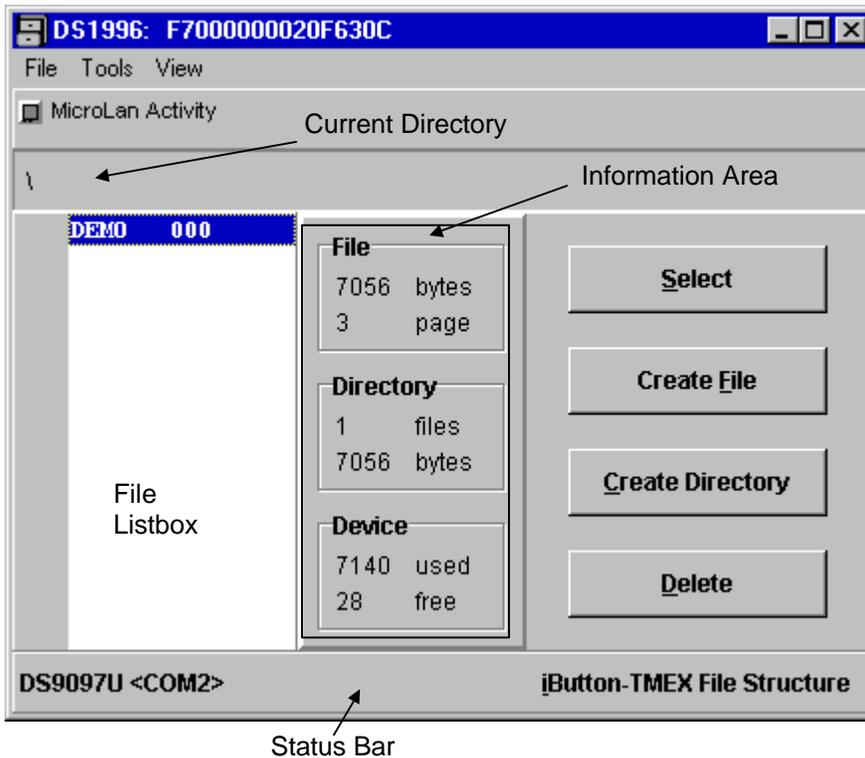
In addition to this the Directory Window has a **Status Bar** as well as a **Menu Bar**. The Status Bar indicates the type and number of the port through which the selected device is accessed. After one has begun a write session to an EPROM device the message "Program Buffer Enabled" appears in the center of the Status Bar. This is to remind that all changes are collected in a buffer to be written all at once to minimize the consumption of memory space. Before actually writing the program checks the population of the 1-Wire bus. The message "*Non program devices detected on the MicroLan ! Programming voltage may damage these non-program devices. Remove these devices and then click OK.*" will be displayed if non-EPROM devices have been found and writing will not take place unless these devices have been removed.

File and device functions that are needed less often are accessible through the Menu Bar. The menus and their functions are:

File   **Rename** to change the name (not the path) of a file
          Attribute to toggle the attribute of a file between normal and "read-only" or to toggle the attribute of a directory between normal and "hidden",
          Program Now to copy the buffer contents to the EPROM device
          Exit to exit the file viewer
Tools   **Format** to prepare a memory device for the storage of data files
View   **Refresh** to re-read the directory information and update the File Listbox
          Show Hidden to include hidden directories in the File Listbox in order to select one and reverse its hidden attribute

Managing Files And Directories
The File Listbox presents information in a similar way as DOS. Revision 3.00 of the iButton Viewer does not allow sorting files. Therefore, files are listed in the same sequence as their entries are stored in the device directory. As more files or directories are created or deleted, this sequence may change.

A file or directory is selected in the File Listbox by clicking on it once. This will display the file size in a granularity of 28 bytes and the file's physical starting page address under the headline "File" in the information area. A directory is not considered to have a size; therefore only its starting page is displayed. Double-clicking on a file name or clicking on the Select Button opens the File Editor Window. Double-clicking on a directory name changes from the current directory to the sub-directory. The path to the sub-directory will be displayed just above the File Listbox.

In the information area under the headline "Directory" the total number of files and directories in the current directory level is displayed together with their combined size in a granularity of 28 bytes. The total balance of used and unused bytes within a device is shown under the headline "Device". A 32 byte memory page can only store 28 bytes since four bytes per page are reserved for formatting and CRC protection.

If one has entered a sub-directory the File Listbox begins with two dummy files marked as ". <DIR>" and ".. <DIR>". The first one is identical to the current directory. The only effect from clicking on it is getting the physical page address of the sub-directory displayed under "File". The second one is the reference to the next higher directory level. Clicking on it once displays the physical page address of the higher directory. Double-clicking it changes from the current level to the next higher one. These dummy files are counted when determining the total number of files in a sub-directory.

Formatting A Device

Like a floppy disk, a fresh memory iButton needs to be formatted before data files can be written to it. If the user has selected a device and starts the file viewer, the program first tries to read the device directory. Since an unformatted device does not have a directory this reading will fail. The user will get the error message: "*Could not read the directory ! Format the device now ?*" Simply click on "OK" and the device will be formatted. Clicking on "CANCEL" will not format the device. However, upon attempting to create a file, the message "*Error reading the directory, operation aborted !*" will be displayed. After the formatting is completed the confirmation message "*Device successfully Formatted !*" will be displayed.

Formatting is required in any case, regardless of SRAM or EPROM technology. When formatting an EPROM device, the program will give several notifications to alert the user of the special conditions that apply. The first message is: "*WARNING, a buffer has been created to log changes to this EPROM device. Do NOT change the memory of this device in any other window !*" This is to remind the user, that although it is possible the device should not be written to through more than one File Viewer window. With SRAM based devices this is not a problem since data is written immediately without buffering it. A further warning "*The Program Buffer will be written when this window is exited or from the file menu. Aborting the EPROM write will negate any changes made.*" is then displayed. This confirms that any physical writing will be delayed until the file viewer to the device is exited. All data changes until then will be collected in a buffer.

A device may be re-formatted at any time. With an SRAM-based device this will erase the device directory and release the memory space to store new data. As with the "Quick Format" function the memory is not physically cleared. Although most of the data remains in the device, the links to it are lost. Re-formatting an EPROM-based device is also possible. However, one will not be able to re-gain memory space for new data. The only effect of re-formatting is clearing the device directory so that the existing files will no longer be found by the File Viewer. As explained in the data sheets of EPROM-based iButtons, the current directory is just re-directed to an unused page. This leaves the links to old files intact and makes them still read-accessible with customized software.

To re-format a device, click on "Tools" in the Menu Bar and click on option "Format". This will display a message "*Format this device now ?*". Click on "OK" and the device will be formatted. The end of formatting will be confirmed in the same way as with a fresh device.

Creating And Writing A File

To create a file, click on the "Create File" button. This will open a child-window where the name of the file is to be entered. The file name consists of up to four characters and a file extension in the range of 0 to 99 (decimal). With EPROM devices the extension 100 is also permissible if one intends to create an "append file". The extension is separated from the file name by a decimal point. If no extension is entered, the value 000 will be assigned by default. After the file name and extension are entered, click on "OK". The program returns back to the Directory Window where the newly created file name will show up in the File Listbox. Now double-click on the file name or click once on the name and once on the select button. This will open the File Editor Window where the empty file is to be filled with data.

Creating A Directory

To create a directory, click on the "Create Directory" button. This will open a child-window where the name of the directory is to be entered. The maximum length of a directory name is four characters. After the directory name is entered, click on "OK". This transfers back to the Directory Window where the newly created directory will show up in the File Listbox. To enter the directory, double-click on the directory name. Now the File Listbox will contain only the two dummy files ". <DIR>" and ".. <DIR>". Files can be created in the directory as described in the previous paragraph.

Deleting A File Or Directory

To delete a file, select the file in the File Listbox by clicking on it once and then click on the "Delete" button. The program will ask for a confirmation. Click on "OK" to delete the file. To delete a directory, select the directory in the File Listbox by clicking on it once and then click on the "Delete" button. As with a file, the program will ask for a confirmation. Click on "OK" to delete the directory. The directory will only be deleted if it is empty. Otherwise an error message "Directory is not empty, operation aborted!" will be displayed. This message needs to be acknowledged by clicking on "OK". After all the files in the directory are deleted individually, the deletion of the directory will be successful.

Renaming A File

To rename a file select the item to be renamed in the File Listbox by clicking on it. Then click on "File" in the Menu Bar and click on option "Rename". This will open a child-window with a field for entering the new file or directory name. After the name and extension are entered, click on "OK". This will update the name and close the child-window. **Note:** It is not possible to convert a file to an "append file" or vice versa by renaming it. Renaming is only possible as long as the new extension does not indicate different file characteristics.

Moving A File

There is no direct way to move a file from one directory or device to another. If moving a file within a device, open the file first in the File Editor Window and copy its contents to the clipboard (see File Editor description, Copy All). Exit the file editor and delete the file. Now go to the directory the file is supposed to be moved to and create the file name. Next open the file and paste in the contents of the clipboard (see File Editor description, Paste All) and exit the file editor.

Moving or copying a file from one device to another is done in a similar way. First the contents of the source file needs to be copied to the clipboard. Then the file name is created in another device and the contents is then pasted into the opened file and written to the target device. This procedure works independent of the device technology. It is possible to copy the contents of an "append file" to an SRAM device. However, the extension of the copy must be different from 100.

Attributes

TMEX is able to set attributes for files and directories. Possible attributes are "read-only" (R) for files and "hidden" (H) for directories. To make a file read-only, select the file in the File Listbox, open the "File" menu in the menu bar and click on option "Attribute". This will open a box with the text " Read Only". Now click on the box and the attribute of the file will change. The character "R" will now appear to the right of the file name in the File Listbox. To reverse the "read-only" attribute, follow the same steps again. Instead of " Read Only" the contents of the box will be "√ Read Only". Now click on the box and the attribute of the file will be reversed. Clicking anywhere else will not change the attribute.

To hide a directory, select the directory in the File Listbox, open the "File" menu in the menu bar and click on option "Attribute". This will open a box with the text " Hidden". Now click on the box and the directory will no longer be found in the File Listbox. To make a hidden directory visible again, open the "View" menu in the menu bar and click on Show Hidden. This will list the hidden directory (if any) in the File Listbox with the character "H" to the right of the name as attribute. To make the hidden directory permanently visible, select it in the File Listbox, open the "File" menu in the menu bar and click on option "Attribute". This will open a box with the text "√ Hidden". Now click on the box and from now on the directory will be listed again. Clicking anywhere else will not change the attribute.

Show Hidden

This option of the "View" menu is implemented to find hidden directories, select them and make them visible again. For details on the use of this option, see the description under "Attributes".

Refresh

Since it is possible to access a single device through more than one directory window it may be possible that one window shows files that have already been deleted through another window or does not show items that just have been created, for example. To avoid uncertainty caused by contradicting information, the refresh option has been implemented. To refresh the contents of the Directory Window, open the "View" menu in the menu bar and click on "Refresh". This will automatically re-read the device directory and display the updated information.

Program Now (EPROM only)

If one intends to write to an EPROM device before closing the Directory Window, one can do this through the option "Program Now" listed under "File" in the menu bar. After having selected this option, one will get the message: "*Program the EPROM device now ? CANCEL will leave the device unchanged.*" To actually write the new information to the device, click "OK". Clicking on "Cancel" will not write but keep all changes in the buffer. If one does not use the "Program Now" option, one will get the message "*Program the EPROM ..."* on exiting the Directory Window. If one now clicks on "Cancel", however, the buffer will be cleared all changes are lost.

File Editor Window

The purpose of the File Editor Window is to examine and edit an iButton file that has previously been selected in the Directory Window of the File Viewer. Files can be viewed in three different ways: as ASCII **Text**, **Binary** Data in hexadecimal format and **BMP** Picture. These different views are reflected in the three tabs that fill the center of the File Editor Window (see below).

The name of the selected file and its path are listed above the display area. The mode of viewing is determined by the file contents. The file extension, except for values of 100 and higher, does not indicate the file type. If a file is recognized as BMP formatted picture it will be displayed as such. If a file is not a picture and does not contain Null bytes (code 00 hex), it is assumed to be ASCII text. Otherwise its contents will be displayed as binary data. An empty file will be treated as text. Text can be edited in text mode as well as in binary mode. Binary files can be edited in the binary mode only. Although pictures can be displayed with the file viewer, they have to be edited using special graphics programs.

Regardless of the file type, one can change the viewing mode by clicking on the label of the tab. However, if the representation is not adequate to the file contents, one may get error messages (when trying to view a text or binary file as BMP picture or viewing a picture as text, for example) and editing may not be possible.

File Viewer, File Editor Window, Text Mode
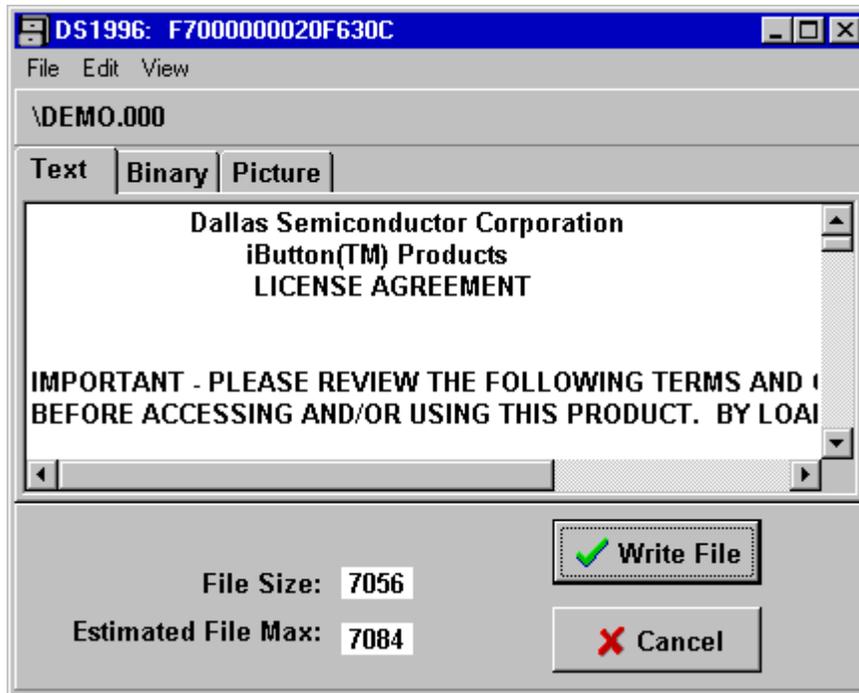
The **File Size** of the file being viewed or edited and its maximum possible size are indicated beneath the viewing tabs. In contrast to the file size in the Directory Window, which is rounded upwards to the next multiple of 28 bytes, the file size shown in the Editor Window is always exact. When editing text, the number will increase or decrease as characters are added or deleted. In Binary mode the file size is updated whenever a group of bytes has been revised (see "Editing Binary Files", below).

The **Estimated File Max**{imum} Size is calculated from the number of unused pages within the device plus the number of pages already used. This estimate usually is very accurate. Under certain conditions, however, the file may finally be up to 28 bytes longer than the estimate displayed. Extensive use of sub-directories reduces the memory space available for data storage. Therefore, after several sessions of creating/editing/deleting files and directories one may be able to reclaim one or more memory pages by running the iButton-TMEX DOS utility TOPT, which defragments files and directories.

To the right of File Size and Estimated File Max there are two buttons, "*Write File*" and "*Cancel*". The Write File button has to be clicked on if the changes made during an editing session are to be written to the device. With an SRAM device, data will be written immediately. With an EPROM device one will get the message that a buffer

is created and no writing will take place before leaving the file viewer or the Program Now function (Directory Window, File Menu) has been invoked. Clicking on Cancel will close the File Editor Window and transfer back to the Directory Window ignoring all edits made during the session. If one has modified a file and tries to close the editor window, one will be reminded of this by the message "File had changed. Update Now ?" Click on "OK" to save the changes or on "Cancel" to exit without updating.

The Menu Bar of the File Editor Window contains elementary functions to exchange data between the iButton and the PC and the Windows Clipboard, respectively. These menus and their functions are:

File	**Open File (PC)** to select an file on the PC to replace the entire contents of the iButton file,	
	Save File (PC) to copy the entire iButton File to the PC	
	Exit to exit the file editor window	
Edit	**Copy All** to copy the entire contents of the iButton file to the clipboard,	
	Paste All to replace the entire contents of the iButton file with data from the clipboard	
	Clear	
View	**Proportional** to temporarily change to a non-proportional font (text tab only)	

Transfers to the iButton will only be successful if the data does not exceed the storage capacity available in the device. Otherwise one will get an error message and the remainder of the file data or clipboard contents will be cut off.

Editing Text Files
If a newly created file is selected, the cursor will be in the beginning of the empty text tab. Now one can type, erase and edit text similar to the Windows Notepad utility. All standard Windows keystroke editing commands apply. The most useful of these are:

Cut	= Shift + Del		
Copy	= Ctrl + Ins	or	Ctrl + C
Paste	= Shift + Ins	or	Ctrl + V
Clear	= Ctrl + Del		

In contrast to the menu functions, the keystroke commands affect the selected text section only, not the entire iButton file. When editing tables it may be useful to change the character font to a non-proportional one. To do this, click on the View menu. This will open a box with the text "√ Proportional". Click on the box to change to the non-proportional font. To change back to a proportional font, repeat these steps. The check mark " √ "will now be missing.

To copy a file **from the PC to the iButton** file, use the function Open File (PC) from the File Menu. The program will then display all "TXT" files found in the directory where the iButton Viewer Program is installed. The file selection can be changed to list "BMP" only or to include all files. However, the file size will not be visible. One can change the directory following the Windows convention to access any file stored

in the computer. After the file in the PC is found, double-click it or click it once and then click OK to transfer its contents to the file editor, replacing previous data. Depending on the data transferred, the editor may change automatically to the binary or BMP display mode.

To copy a **segment of a file to the iButton**, access the file with a text editor or word processor and copy the segment to the Windows clipboard. Then return to the File Editor window and use the function Paste All from the Edit Menu. This will replace the previous contents of the iButton file with data from the clipboard.

Data can be copied **from the iButton to the PC** in two different ways: into a file of its own or to the clipboard to be incorporated into an existing file. To copy **to an independent file**, use the function Save File (PC) from the File Menu. By default, the program will create the file in the directory where the iButton Viewer Program is installed. The file name on the PC will be the same as in the iButton. For text files the extension will change to TXT and to BMP if one is copying a picture. Binary files will keep their original extension. However, name, extension and path of the file may be changed, if desired. To copy from the iButton **to the clipboard**, use the function Copy All from the Edit Menu. ASCII text or a picture can now be pasted into a word processor document and be re-used. Copy All does not work with binary files.

Editing Binary Files
Any file can be displayed and edited in the binary format. The picture below shows a sample text file in hexadecimal presentation. Each line starts with the byte address inside the file (not the device) followed by 8 bytes of data, each separated by a space. With an empty file one will only see the address information [000].

To enter or edit data, double-click on the line where the editing is to begin. This will select all eight bytes in that line and copy them into a child-window, as in the example below, where the editing takes place. Click anywhere in this line to set the cursor. Entering new data without setting the cursor will replace the whole line.

Data can be inserted or overwritten as in any typical Windows Application and is not limited to the 8 bytes of the line. Even the keystroke commands apply, but their use is rather limited. Input data may be hexadecimal or ASCII. Groups of *two* characters in the range of 0 to 9, A to F (not case sensitive) separated by a space are understood as hexadecimal (see picture). *Single* characters separated by a space are understood as ASCII and are translated into their Hex equivalent. Hexadecimal and ASCII data may be mixed in the same line during editing. Spaces themselves are delimiters and therefore cannot be entered as character, only as code 20 hex.

After the new data is entered, click on OK to transfer the edits to the main window where it will be inserted from the selected line onwards. Any changes in length will be reflected in the File Size below the display section.

File Viewer, File Editor Window, Binary Mode
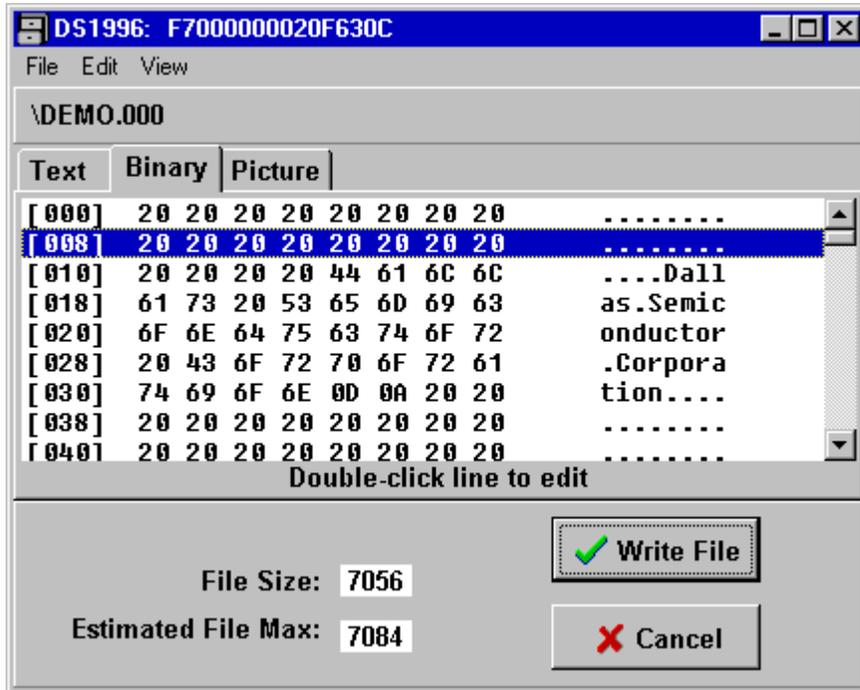
File Viewer, File Editor Window, Binary Mode, Hex-Editor Child-Window
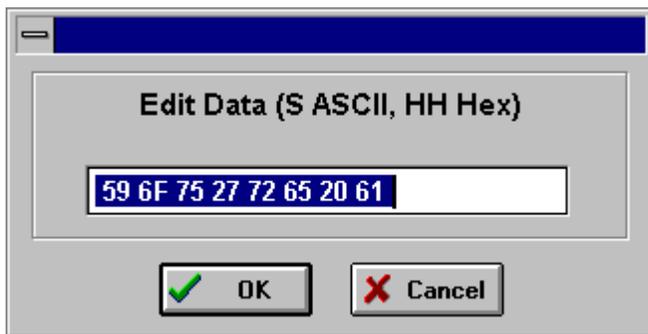
Handling BMP Picture Files

The BMP mode has been implemented to support viewing bitmap files stored in iButtons and exchanging them between the iButton and PC. Bitmap files (DOS file extension BMP) are created using programs such as Paintbrush (PBRUSH.EXE) and other graphic programs. They are pasted into a previously created and opened iButton file using the "Open File (PC)" function of the File Menu or "Paste All" of the Edit Menu. Data transfer of BMP pictures to the PC is also possible. Use the File Menu function "Save File (PC)" to create a disk file or the Edit Menu function "Copy All" to copy to the clipboard from where the picture can be pasted into a word processor document, for example.

Although bitmap files can be displayed and edited in the binary mode, it is not recommended to do so unless the exact format of BMP files is known in great detail. Changing or adding bytes arbitrarily is likely to corrupt the file and may lead to unpredictable results. The file may not even be recognized as a bitmap.

File Viewer, File Editor Window, BMP Mode
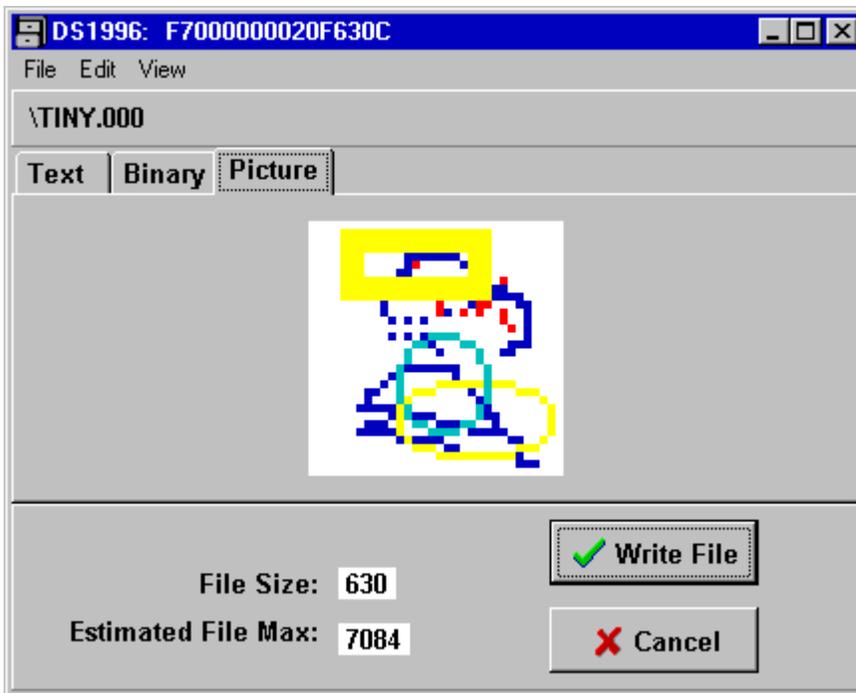
Editing EPROM Append Files

Append files may contain text or binary data. They are viewed in the same way as any other data file. However, there are some notable differences. After one has selected an append file in the Directory Window the following message will be displayed: "*The file selected is an AddFile. The data entered will be appended to the end of the file. Note that the current contents of the file is displayed.*" After clicking on OK the message will disappear and one can read the full file contents, either in the text or binary mode.

As soon as one clicks anywhere in the display area of the tab, the display area will be cleared and one can enter the data to be appended. In binary mode the first line will show the address of the next available byte of the file. After data entry is finished one proceeds in the same way as with other text or binary files that are to be written to EPROM devices.

There are no restrictions on the data within an append file. The program determines the end of data by loading the append file into a buffer and then scanning it backwards (i. e., from the highest to the lowest address). The address of the first byte found that is different from FF hex is taken as "end of file marker". The starting address for new data will be "end of file marker" + 1. Since unprogrammed EPROM memory cannot be distinguished from a byte that deliberately has been written FF hex, the data of an append file **must not end** with FF hex. Otherwise it will be overwritten with new data in the next session. This special precaution does not apply to text files since FF hex is not a valid character code.

D. Time Viewer

The Time Viewer is an easy to use tool to read and set all of the counters and registers in the register page of the DS1994 iButton device. In addition, one has full access to the control and status register, which makes it possible to use all of the device's special features. The Time Viewer Window with sample settings is shown below.
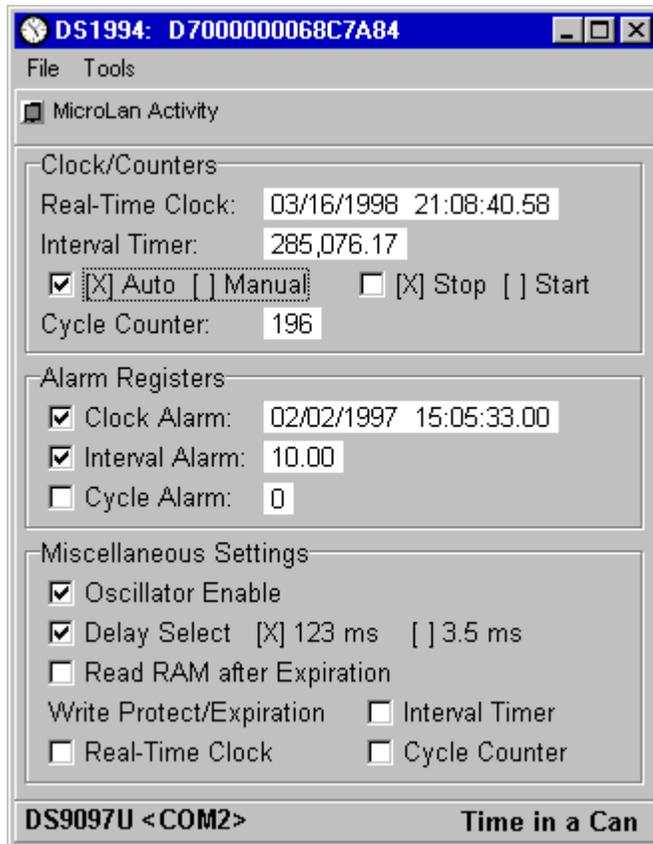
The window consists of three major sections labeled "Clock/Counters", "Alarm Registers", and "Miscellaneous Settings". In the "Clock/Counters" section the viewer displays and periodically updates the readings of the three counters of the device. These counters are called Real Time Clock (binary format, UNIX-compatible), Interval Timer and Cycle Counter. Associated with each counter is an Alarm Register of the same size. The contents of these registers is shown in the section "Alarm Register". Bits and flags that control the device operation are found either with the counters (interval timer control "Auto, Manual"; "Stop, Start"), the Alarm Registers (alarm enable, one for each counter) or in a separate section "Miscellaneous Settings" (Oscillator Enable, Delay Select, Read RAM after Expiration and Write Protect/Expiration). As with the other windows, there is also a Status Bar that indicates the

communication port in use and a short Menu Bar with the entries File (to exit the Time Viewer) and Tools (to set the clock).

Changing Settings
Any setting in the white fields of the display and control sections can be changed by simply clicking on it. For control functions, the contents of the field will toggle between "×" as a check mark and plain white. For registers a child window will pop up with the present contents of the register. To replace the contents completely, simply type the new value. Otherwise position the cursor, select the portion to be replaced and start typing, just as with Notepad, for example. After the new value is entered, click on "OK" to transfer the value to a buffer or click on "Cancel" to ignore the editing. The new value will be displayed in purple and two temporary buttons "Write" and "Abort" will appear below the Menu Bar. One may write each change individually or collect the changes of several controls and registers in the buffer and write them all at once. When write-protecting one or more counters, all the updates must be collected and written all at once. For the change(s) to become effective one has to click on the "Write" button. "Abort" will return to the previous settings without writing the buffer. Regardless if data is written the temporary buttons will now be gone. Closing or exiting the Timer Viewer while the temporary buttons are visible will ignore the updates collected in the buffer.

Operating The Clock
The real-time clock of the DS1994 is a 40-bit binary counter that increments 256 times per second. The 8 least significant bits count the fractional seconds. The upper 32 bits represent the number of seconds passed since January 1st, 1970, 0:00:00 hours (= 12:00:00 a. m.) as reference point. The 40 bits of the real-time clock can accumulate 136 years before rolling over.

With a DS1994 as shipped from the factory the real-time clock is not running. To **set the time**, click on the white field right of the text "Real-Time Clock" and enter the actual date AND time in the format MM/DD/YYYY hh:mm:ss (24-hour format). Alternatively one can use the function "Set Real-Time Clock From PC Clock" of the Tools Menu. In addition to setting the clock, the crystal oscillator needs to be enabled to get the clock running. To do this, click on the small square field "**Oscillator Enable**" in the Miscellaneous Settings section until the "×" appears and click on the temporary "Write" button below the Menu Bar. Now the real-time clock should be running and one should see how the date and time are constantly updated.

The crystal oscillator can be turned off by clicking on the "Oscillator Enable" control field until the "×" disappears. This will conserve energy if the device is not in use for a long time. **However, for the interval timer and cycle counter to function the** crystal **oscillator** that controls the real-time clock **must be enabled.**

Operating The Interval Timer
The interval timer is a 40-bit binary counter that, when enabled, counts at the same rate as the real-time clock. Its contents are displayed in seconds as a decimal

number with fractions. The interval timer is either in the Automatic or in the Manual operation mode. The Automatic Mode is selected if the white square "[×] Auto [ ] Manual" in the Clock/Counter section of the Time Viewer Window contains the "×" check mark. Otherwise the timer is in the manual mode.

In the **Automatic mode**, the interval timer will begin counting after the data line has been high for a period of time determined by the **Delay Select** bit (see description below). The interval timer will stop counting after the data line has been low for the same time period. The automatic mode can be useful when monitoring the operating time of equipment. Simply provide a DC voltage in the range of 2.8 V to 6.0 V to the data contact of the DS1994 as long as the equipment is powered up. Removing the device and reading the interval timer will tell exactly how many seconds the equipment was running.

In the **Manual mode**, the interval timer is operated under software control. It is either running or stopped, regardless of the voltage on the data line. One may toggle between start and stop by clicking on white control field "[×] Stop [ ] Start". The interval timer is stopped, if the control field contains the "×" check mark. Otherwise the timer is running and one will see that its contents are periodically updated. Although one may change its setting, the Stop/Start control field has no function if the device is in the Automatic mode. The Manual mode can be useful when monitoring how long computer-controlled equipment is really productive (not just powered up). Whenever the equipment changes from idle to active, its controlling computer should start the interval timer and stop it as soon as the equipment returns to the idle state. In addition to this counting service the DS1994 also provides time and date information for event logging purposes and non-volatile memory for setup data.

To **preset or clear** the interval timer, click on the white field right of the text "Interval Timer" and enter the new value as a decimal number (integer or with decimal point, no comma). See section "Change Settings" above for details. The maximum value the interval timer can be preset to is approximately 4294 million (decimal) with a granularity of about 1/100 of a second.

Operating The Cycle Counter
The cycle counter is a 32-bit binary counter. It increments after the logical level at the data line changes from high to low and then stays low for a period of time determined by the Delay Select bit (Miscellaneous Settings section).

The cycle counter can be useful when monitoring the number of power up/down cycles of equipment or just counting pulses that represent revolutions of an object, for example. For cycle counting, connect a DC voltage in the range of 2.8 V to 6.0 V to a switch and connect the switch to the data contact of the DS1994. Every time the switch closes and opens the cycle counter will increment. In the example given for the Automatic mode with the interval timer the cycle counter will automatically count the power up/down cycles, thereby supplementing the run time information with stress information.

To **preset or clear** the cycle counter, click on the white field right of the text "Cycle Counter" and enter the new value as decimal integer number. See section "Change Settings" above for details. The maximum value the cycle counter can be preset to is approximately 4294 million (decimal).

Delay Select

The cycle counter and interval timer of the DS1994 are designed to work with digital signals as well as mechanical switches. A mechanical contact closure, however, is not a clean transition from an off to an on state. At a closer look one can see that a mechanical switch generates a series of uncontrolled short pulses ("bounces") until it finally reaches the steady on or off state. Therefore the DS1994 has an on-chip debouncing logic that filters the transitions and provide clean signals that finally control the counters.

One can set the time delay of the debouncing logic to either 123 ± 2 ms or 3.5 ± 0.5 ms. If possible, the longer delay should be chosen. This gives a good filtering and also allows communication on the data line without incrementing the cycle counter or starting the interval timer (if the communication occurs in short data packets with an idle low status). The shorter delay allows higher counting speeds, but under certain conditions may lead to incrementing the cycle counter by wave forms that the device generates itself (interrupts).

The **Delay Select** bit is found in the Miscellaneous Settings section of the Time Viewer Window. A check mark "[×]" in the control field left of "Delay Select" indicates the long delay, a missing mark represents the short delay. To toggle between the two delay values click on the control field and proceed as explained under "Changing Settings".

Using Alarms

To generate an alarm two things need to come together: the value of the counter must be equal to the value of the alarm register AND the alarm must be enabled. Setting the registers and enabling the alarms is done through the Alarm Registers section of the Time Viewer window. Alarm values are entered in the same way as counter values. To **enable an alarm**, click on the control field left of the name of the alarm type until the check mark "[×]" appears. This arms the device to generate interrupt signals as the alarm goes off. To disable the alarm, click again on the control field. The check mark will disappear. After the value is entered and the alarm is enabled, proceed as explained under "Changing Settings".

The iButton Viewer will perform a Conditional Search for all DS1994s being viewed and read the Status Register as soon as an alarm has gone off. The type of alarm (time, interval or cycle) is then determined from the settings of the alarm flags inside the status register and a message like the sample below is displayed.
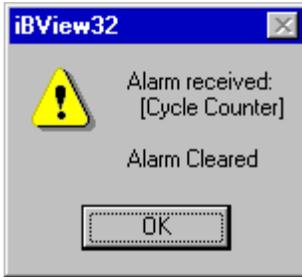
Reading the status register automatically clears the alarm flags. Once an alarm is enabled, it can be armed for the next event by writing a higher value into the alarm register or resetting the counter.

Compared to having the bus master permanently reading the counters and checking against a limit, using alarms in conjunction with interrupts is very efficient. It reduces traffic on the MicroLAN bus and makes sure that the master is informed the fastest possible way. Even if, due to poor electrical contacts during reading, the status of the alarm flags are lost, their status can be reconstructed by comparing counter registers and alarm registers and checking the interrupt enable bits.

Write Protect / Programmable Expiration
The DS1994 can be used as a programmable expiration controller, for example, to force a service call if a machine has been used for a predefined number of operating hours, a certain number of power on/off cycles, or if a certain date is reached. This mode is invoked by write protecting a counter/alarm register pair after it has been loaded with the desired values. The programmable expiration takes place when one or more registers have been write-protected and the first alarm condition occurs.

The behavior of the DS1994 after expiration is determined by the "**Read RAM after Expiration**" flag, located next to the Write Protect flags in the "Miscellaneous Settings" section of the Time Viewer Window. If this flag is set (check mark "[×]" is visible) one will still be able to read the memory, but any writing will be impossible after expiration. If this flag is not set (no check mark), only the ROM contents can be accessed after expiration. Before expiration the status of this flag has no impact on device operation.

Write-protecting counter/register pairs protects not only the register pair from modification but also all other write protect bits and the "Read RAM after Expiration" flag. The oscillator can still be started, but not stopped. If the cycle counter is write protected, the Delay Select setting can no longer be changed. Write protecting the interval timer also freezes the operation mode of the interval timer. If the timer is in manual mode, it will run continuously as soon as the write protection becomes effective.

The expiration controller function works even if the alarm for the particular register is not enabled. If the alarm is enabled, the expiration will be signaled by interrupt

pulses. In this case the "Read RAM after Expiration" flag should be set. Otherwise it will not be possible to stop the interrupt signals that go along with the expiration and slow down the communication on the MicroLAN bus.

To **invoke the expiration controller mode**, first set all counters to their starting values and the associated alarm registers to the value that marks the expiration. Enable the alarms (optional) and set the "Read RAM after Expiration" flag (unless the memory must not be accessible after expiration). All these settings are not critical and can be repeated. In the final step, set the write protect/Expiration bit for the counter(s) that shall determine the expiration. If the expiration shall be triggered by the counter that reaches its limit first, then write-protect all counters that apply and then write the buffered changes to the device. Once a counter has been write-protected one cannot protect another one.

The Write Protect / Expiration Controller function is not reversible. The chance to inadvertently invoke this mode is low. In contrast to the other settings, the program will pop up an extra window (see below) and ask for confirmation.
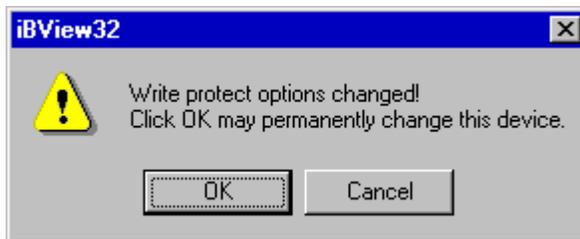
Only if one clicks on "OK" is the Write Protect / Expiration Controller function really activated.

E. Secure Viewer

The Secure Viewer is used to read and write the DS1991 MultiKey iButton. In contrast to Memory iButtons, this device uses a file system that is implemented in hardware. The device is designed to store three files of 48 bytes each that are called Subkeys. Instead of 4 bytes with a numeric extension, the file name consists of 8 bytes and is called ID Field. In addition to the ID Field, each file has an 8-byte write-only password to protect its contents from unauthorized read access.

As with the other Memory iButtons, this device also includes a scratchpad. The scratchpad may be used to write data and read it back for verification before it is copied to its final destination in one of the Subkeys. With its 64 bytes, the scratchpad of the DS1991 can hold the ID Field, Password and all 48 data bytes of a Subkey. The usage of the scratchpad for intermediate storage is highly recommended for an environment where dwell time of the device at the 1-Wire network is short. Assuming that the dwell time is long enough, the iButton Viewer does not use the scratchpad and writes directly to the Subkey. As a consequence, the scratchpad becomes available as an additional publicly accessible memory space.

All of the memory areas of the DS1991 MultiKey iButton with the ID fields and password-entry boxes of the three Subkeys are laid-out in the window of the Secure Viewer in a straight-forward way, as shown below.

The three Subkeys combined into a block "Secure Subkeys" fill the lower three quarters of the window. To the left, above each Subkey's data field there is a box labeled "ID" to display the contents of the key's ID Field. Above the center of each Subkey's data field is another box labeled "Password". When reading a Subkey, one has to type the password into this box to give the program access to the secure data. Above the block of Subkeys there is a smaller block called "Scratchpad". The scratchpad is represented as one line that displays the contents of the scratchpad in character form. As usual, the window is completed by a Status Bar that indicates the communication port in use and a short Menu Bar with the entries File (to exit the Secure Viewer) and View (to refresh the screen, in case a single device is viewed through multiple windows). This sample picture was generated using a device in the state as it is shipped from the factory. The "U"s are the ASCII representation of the character code 55 hex (= 85 decimal), a pattern that typically is written to the device before it leaves the factory.
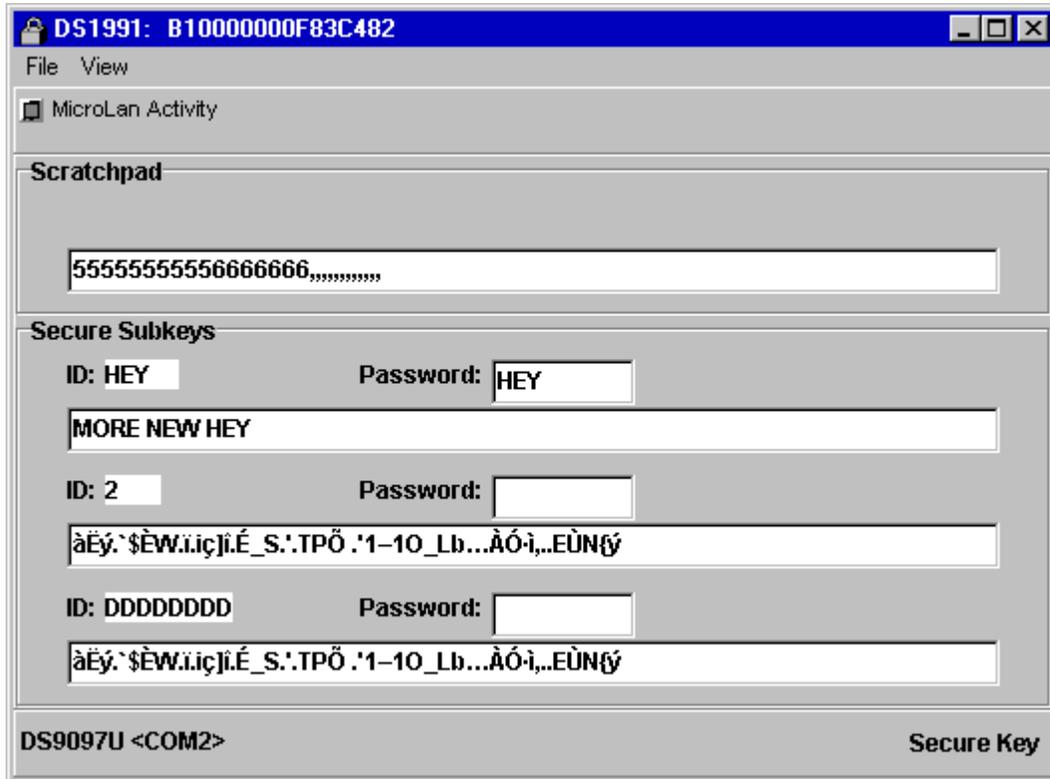
The Secure Viewer does not know or remember passwords. To meet the requirements of the communication protocol the viewer uses 8 spaces (code 20 hex or 32 decimal) as password when trying to read the Subkeys. Since this is usually different from the correct password, the device responds with "garbage" data, a pseudo-random pattern based on the provided password (not the stored one). The "garbage" is then checked for its printability. All non-printable codes, i. e., codes below 20 hex (32 decimal), are replaced by a "." before data is displayed in character form as Subkey contents. Depending on national settings, the characters actually seen may be different from those shown in the sample window.

Writing Secure Data

Before one can write secure data to a Subkey for the first time, one has to give the Subkey a name (optional) and specify a password (mandatory). To change the name of a Subkey, click on the box that displays the current contents of the key's ID Field. This will open a child-window with the current name already selected for re-placement, as shown below.
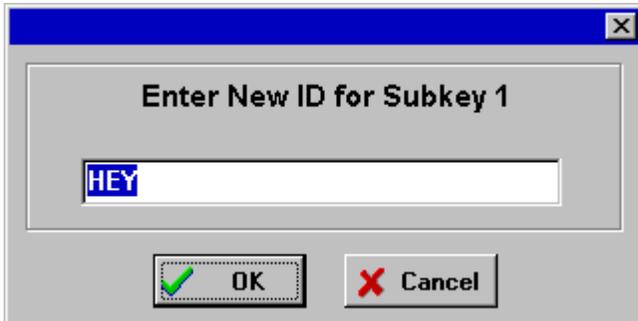
Enter the new name and click OK. This will bring up a similar window for setting the password.
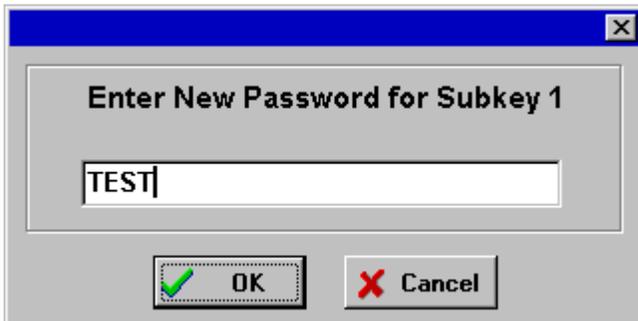
The default password of the Secure Viewer is 8 spaces. All but the first space are removed from the editing area of this child-window. The remaining single space is selected for replacement. Simply fill in the new password and click on OK. Clicking on "Cancel" will ignore the new name as well as the password. For both, the ID Field and the password, the program allows up to 8 characters, including embedded spaces, to be entered. Shorter names are padded with spaces to make the 8 characters necessary to meet the hardware requirements of the device.

After file name and password have been entered, the new names will be visible for the Subkey selected. Since writing a password erases the contents of the Subkey, the Subkey's data field will now be filled with 48 dots, representing the non-printable code 00 hex. Now double-click on the dots to select them for replacement and type the new contents of the Subkey. With the first character entered, a "Write" button will appear above the right end of the Subkey's data field. When finished typing, click on

this temporary button to transfer the text to the secure Subkey. The text entered will be appended by spaces to fill the 48 bytes of the secure storage space of the key. To cancel the transfer without writing, exit the window of the Secure Viewer or click on "Refresh" in the view menu. After the data is written to the device, the "Write" button will disappear, but the file name, password and contents will remain visible. To make the password and the true contents of the Subkey invisible, overtype the password.

Reading Secure Data

To read secure data, click on the password-entry box of the key to be read and type the password. When the typing ends, the program assumes that the password is complete and will access the device. If the password is correct, the secure data will be displayed in its true length, with trailing spaces removed. Otherwise the display will show random data, based on the password entered. Note that the password is case-sensitive and must be entered exactly as it is stored in the device. Although the program fills short passwords with trailing spaces to make up the 8 bytes required by the DS1991, these trailing spaces need not be entered.

Modifying Secure Data

After secure data is read from one of the Subkeys of the DS1991 it can easily be modified and written back to the device. First access the Subkey as described in section "Reading Secure Data". Now double-click on the word to be replaced and enter new text. To replace a selection, select the text to be replace with the mouse and type the new text. To replace the whole contents, select the full line. Alternatively, one can position the cursor anywhere in the data field and start erasing or typing, whatever is necessary. The program counts the characters and stops accepting more data as soon as the data field contains 48 characters. As described in section "Writing Secure Data", a "Write" button will appear that needs to be clicked to finally write to the device.

Changing The ID Field And/Or Password

The Secure Viewer allows the user to change the ID Field of a Subkey and its password while keeping the Subkey's contents. However, this is a multi-step procedure involving the Windows clipboard and keystroke commands. The procedure begins with displaying the contents of the subkey who's ID or password is to be changed. See section "Reading Secure Data" for details. Next select all the contents of the Subkey's data field with the mouse and copy it to the clipboard using the keystroke command Control + C.

Now click on the display area of the Subkey's ID Field. This will pop up the child-window for entering the file name with the current name filled-in and selected. Make changes as desired or click on OK. This will transfer to the password child-window with the current valid password filled-in and selected. Make the desired changes and click on OK. This will write the new ID and password and erase the Subkey's contents. Now double-click on the Subkey's data field that is now filled with dots and paste in the contents of the clipboard using the keystroke command Control + V. Now click on the "Write" button to write the data to the device.

Trying to change the password by simply typing new text in the password field is not possible. The programs would assume that one is going to try another password for reading the Subkey's contents. Changing the data field of the Subkey first and then the password before writing to the device will give the error message "Error writing the sub-key, password may be incorrect !". This comes from the fact that the program tries to write to the device using the password shown on the screen. In the read-back cycle after writing the program will then see that the data does not match and therefore display the error message.

Reading/Writing The Scratchpad
With the Secure Viewer the scratchpad is treated as a Subkey without ID Field and password. The current contents therefore are displayed automatically. For editing and writing the scratchpad the same rules apply as for the data field of a secure Subkey. See section "Modifying Secure Data" for details. With the scratchpad, however, the program accepts up to 64 bytes.

F. UDP Viewer

The UDP Viewer provides a simple means to read and write universal data packet. UDP stands for **U**niversal **D**ata **P**acket, a simple format that protects a serial data stream against transmission errors by cutting it into packets that each are preceded by a length information (e. g., one byte) and followed by an inverted CRC16 over the length byte and the data of the packet. With its 128 bytes of memory capacity and the formatting overhead of three bytes the DS1992 can finally store 125 bytes of user data.

The window of the UDP viewer (see below) has some similarity with the File Editor Window of the File Viewer in text mode. The center of the window provides a large area for reading/editing text. The lower part of the window displays the character count of the data packet and its maximum size. The window is completed by a Status Bar that indicates the communication port is in use and a short Menu Bar with the entries File (to exit the UDP Viewer) and View (to refresh the screen in order to cancel edits that are not yet saved).
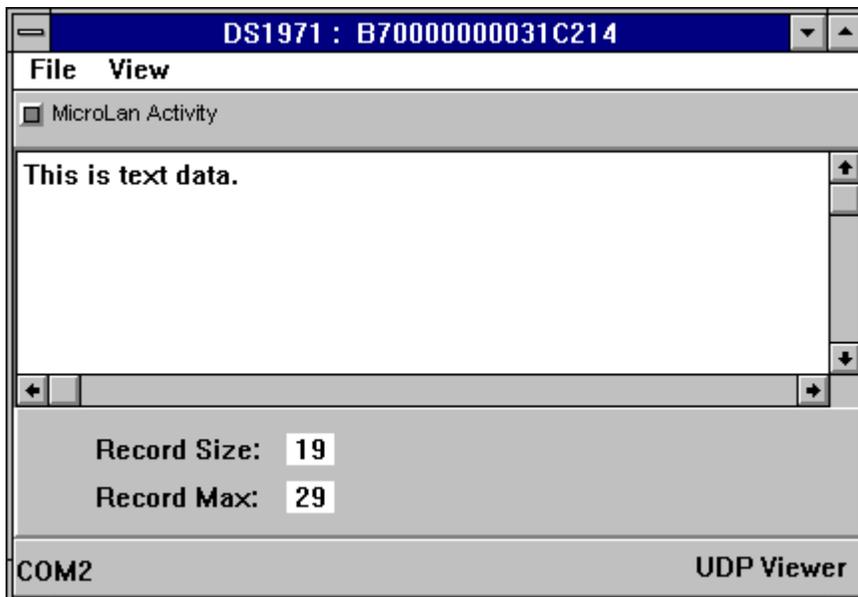
Reading Text

As the UPD Viewer is activated it automatically reads the main memory of the selected device, checks the data format and displays the data content as ASCII text. This works well if the device has been programmed in a previous session of the UDP viewer. Since a fresh device contains a constant pattern (typically 55 hex = 85 decimal) in every location, the format test will fail and the error message "*Could not read a valid UDP on this device !*" will be displayed in a separate window. In this case simply acknowledge the message by clicking "OK". If a device has been formatted by the File Viewer it will contain a valid data packet, but no text data.

Since the UDP viewer does not support editing binary data it will display the message "*WARNING, this record contains binary data. It may not display correctly in this editor.*" Simply acknowledge the warning. The text area will show a small raised "a" as its only contents and the Record Size will be 1.

Editing And Writing Text
Once the text stored in the device is displayed, one can start typing new text if the device is empty or modify existing text. To replace a word, double-click it and enter new text. To replace a selection, select the text to be replace with the mouse and type the new text. Alternatively, one can position the cursor anywhere in the data field and start erasing or typing, whatever is necessary. The program counts the characters and stops accepting more data as soon as the data field contains 29 characters. The current number of characters is displayed as "Record Size" below the text reading/editing area.

As soon as the first modification is made, a button "Write UDP" will appear in the lower right of the UDP Viewer window. To write the new text to the device click this button. To ignore the edits, exit the window of the UDP viewer or click on "Refresh" in the view menu.

G. Temperature Viewer

The Temperature Viewer was created to demonstrate the features of the DS1920 Temperature iButton. The DS1920 is a single-chip direct-to-digital thermometer with 1-Wire interface and parasitic power supply. It directly converts the temperature at the chip into a 9-bit digital reading in ° Celsius with a 0.5 °C resolution over a working temperature range from -55 to +100 °C (-67 to +212 °F). Access to counter registers of the temperature conversion logic allows the user to increase the resolution of the temperature reading to 1/100 °C approximately, depending on the actual temperature. To watch for extreme heat or cold the chip includes two non-volatile trip point registers that store an upper and lower temperature limit and are linked to the temperature converter by a comparison logic. If the actual temperature gets outside the predefined range, this logic will make the device respond to the conditional search command to signal an alarm.
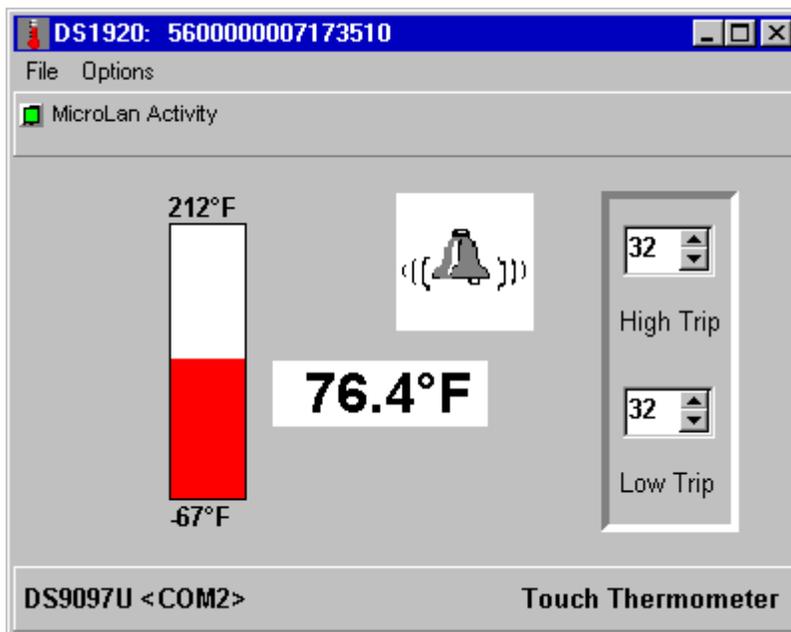
The resources of the DS1920 are laid-out in the window of the Temperature Viewer as shown above. The temperature reading pictured as an analog thermometer and as a digital display occupy the majority of the window. The high and low trip points are displayed at the right side of the window. If the measured temperature is outside the interval set by the two trip points, an alarm bell will be visible above the digital temperature display. The window of the Temperature Viewer is completed by a Status Bar that indicates the communication port in use and a short Menu Bar with the entries File (to exit the viewer) and Options (to change the temperature scale).

Reading The Temperature

The temperature display is updated continuously. With every new temperature conversion the color of the number in the digital display toggles between red and black to make the new value distinguishable from the previous one. The temperature viewer uses the contents of the counter registers of the conversion logic to interpolate the reading to a resolution of 0.1 °C/F.

When a brand new DS1920 is viewed, both trip points are likely to be at 32 °F (= 0 °C) and the alarm bell will be visible. For information on how to set the trip pints, please see the next paragraph. Although the DS1920 internally operates in the °C scale, the Temperature Viewer uses the °F scale to display the temperature reading and trip points whenever it sees a device for the first time. For convenience, the temperature scale can easily be changed to °C and back to °F. The program will remember the last setting when the same device is accessed later. To change between scales, open the Options menu and click on the preferred scale. Changing the scale automatically recalculates the actual temperature reading and the trip points.

Changing The Trip Points

The trip point temperatures are stored in the device with a resolution of 1 °C. By hardware design, changing the value of one trip point automatically re-writes the other. To change the value of a trip point, click on the up or down arrow to the right of the trip point display. A single click will increment or decrement the value by 1 °C. Keeping the mouse button pressed will change the number repeatedly in the direction of the arrow. Alternatively one can click on the value of the trip point and use the cursor up/down keys to change the setting. The upper and lower limits for the trip point setting are +100 °C (+212 °F) and -55°C (-67 °F) respectively. A few seconds after the last change new values will be written to the device. The numbers become black to indicate that they are identical to those stored in the device.

When using the °F scale the trip point value as selected on screen will be converted and rounded to the nearest °C value before it is written to the device. When read back, another conversion and rounding is performed. Since ± 1 °F is a smaller step than the ± 1 °C that the device can handle, the value read back may actually be ± 1 °F off from the desired value.

The Temperature Viewer does not check if the values of the trip points define a true interval. For proper functioning of the alarm signaling scheme the high trip point must have a higher value than the low trip point. To identify alarms, the program performs a Conditional Search on the DS1920 being viewed. If the device responds, the alarm bell will be displayed.

H. Switch Viewer

The Switch Viewer was created to demonstrate the basic features of the DS2407 Dual-Channel Addressable Switch Plus Memory. The DS2407 combines several functions in a single chip:

- electronic on/off switch
- activity sensor
- level sensor
- 1K-bit EPROM

The switching and sensing functions are reflected in the window of the Switch Viewer as shown below. The memory is accessed by means of the File Viewer. It has the same size as with the DS1982 and is typically used to store application specific data.
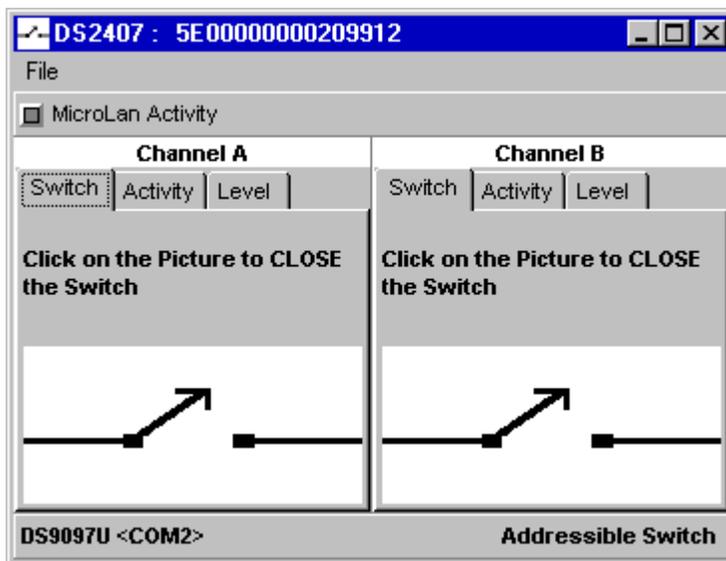
The Switch Viewer window has two identical working areas A and B, one for each channel. Each area has three tabs: Switch, Activity and Level. The window is completed by a Status Bar that indicates the communication port in use and a short Menu Bar with File (to exit the viewer) as its only entry. As the Switch Viewer is started, the Switch tab is opened for both channels. The symbol of the switch in the lower half of the tab shows the actual status of the switches. On power-up both switches are in the open (off) state. Although the device can be programmed to power-up with one or both switches on, this function is not supported in this version of the Switch Viewer.

To **close the switch**, click anywhere on the switch symbol. The picture will immediately change to a closed (on) switch. If a load (LED via resistor to 5 Volt, for example) is actually connected to the switch, it will now be powered up (the LED will emit light). To **open the switch** again, click once more on the symbol of the switch. With each click the symbol will toggle from closed to open or vice versa and the load will be switched off or on. The switch remains in its current state if the Switch Viewer is closed. If the data line is disconnected, all DS2407 devices will return to their power-on default setting as soon as communication on the 1-Wire bus resumes.

The input (sensing) capabilities of the DS2407 are accessed through the tabs Activity and Level. To change to these modes, click on the label of the tab. The picture below shows an example where channel A monitors activity and channel B monitors the voltage level. **For any sensing to function, the switch of the sensing channel must be open**. Otherwise it will short the input to ground and thereby prevent the voltage from rising.

Activity is defined as the transition of voltage from low to high or high to low. The activity mode is most useful to monitor piezo sensors. Since the sensing inputs of the DS2407 have an internal pull-down resistor of approximately 10 MΩ, one has to apply external voltage to generate activity. This can be done by a mechanical switch from the I/O channel of the DS2407 to a positive voltage of 5 V, for example. In series with this switch should be a resistor of a few kΩ or more to prevent damage to the DS2407 in case the switch is not open as it must be for sensing. The activity tab has two option settings, "Auto Clear" and "Sound on Activity". Auto Clear, if checked, clears the activity sensor so that it can capture the next transition. If checked, Sound on Activity allows a short beep to be generated with every transition as long as the activity is viewed. Whenever activity is detected, the lower half of the Activity tab shows an alarm bell to optically signal activity. With Auto Clear set, the bell disappears shortly after the activity was detected. If the bell remains visible, the activity sensor is most likely stimulated repeatedly or the Auto Clear is not checked (no "×" in the box left to Auto Clear). Since the activity sensor cannot distinguish between internally and externally induced transitions, activity will be signaled as one changes from switch mode to activity mode if there is a load connected and the load has been switched before.
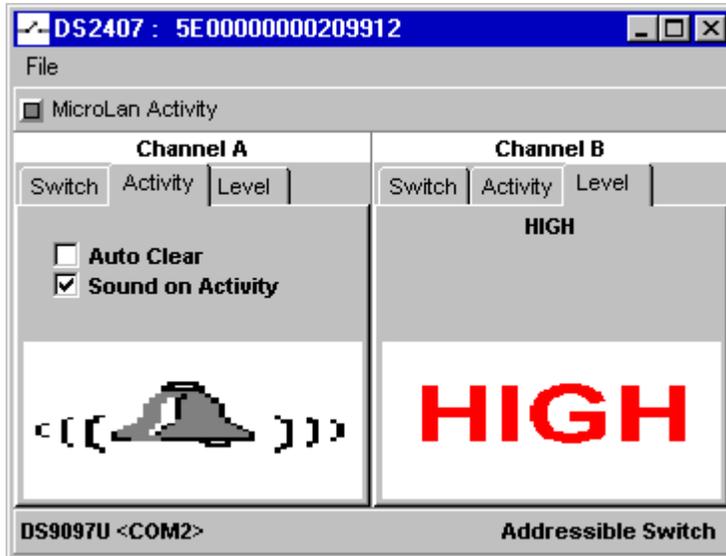
In the **Level** mode, a new sample of the logical level at the I/O channel is taken whenever the DS2407 is accessed and displayed as HIGH or LOW in the lower half of the Level tab. In contrast to Activity, which is sensitive to transitions, Level does not catch instances where the voltage level deviates from its quiescent level for a short moment only. In case one needs to see both, one may connect both channels outside of the device and view the activity with one and the level with the other.

The DS2407 is available as Dual Channel and as Single Channel version. For the single channel version the window of the Switch Viewer includes channel A only and therefore is only half the size. Everything else works the same.

I. Viewer Customization/Advanced Features

Re-sizing Windows
As any common Windows program, the iButton Viewer allows several device viewer windows to be open at the same time. These windows are opened with a default size and position on screen. Both the size and position can be modified to provide more space for reading/editing files with the file viewer or UDP viewer, for example. To store modified window size/or position, the option "Auto Save Size/Position" must be checked.

MicroLan Port (Options Menu)
This function allows the user to dynamically change the 1-Wire Port that is monitored by the iButton Viewer to open additional sessions on a second or third port. It does not alter the default selection made during the iButton-TMEX installation. When activated, MicroLAN Port will display a window that provides a list of the possible ports similar to the port selection scheme used during the iButton-TMEX installation. A brief test is performed on the selected port to verify that communication is possible. If one changes the port the list of devices in the Network Listbox is cleared. If "Auto Viewer Start" is enabled another viewer for each device discovered will be launched.

Auto Viewer Start (Options Menu)
If this option is checked the iButton Viewer will automatically launch the primary viewer for each device discovered on the MicroLAN. For devices that have been viewed before, the latest window size and position will be used if "Auto Save Size/Position" is enabled. Otherwise the default window settings are applied. Auto Viewer Start is convenient when preparing demonstrations. If "Auto Viewer Close" is also checked, the primary viewer of arriving devices will automatically be opened in a predefined position and size and disappear as the device departs from the MicroLAN.

Auto Viewer Close (Options Menu)
If this option is checked the iButton Viewer will automatically close subsequently started device viewers when the device has departed from the MicroLAN. This option is normally used in conjunction with "Auto Viewer Start" to automatically start and close the primary device viewer when communicating with a transient device.

Auto Save Size/Position (Options Menu)
This option is checked automatically. If this option is checked the size and position of the windows of the iButton Viewer and any subsequently started viewers together with the ROM code of every device viewed will be saved automatically in the IBVIEW.INI file (16-bit) or the registry (32-bit) and retrieved when the program is started again. This feature is convenient if the pool of devices viewed is limited and one cannot or does not want to use the default window sizes or positions. For each device the windows configuration is saved, so that if, many buttons are tested with this option enabled the IBVIEW.INI file (or registry) may grow substantially over time. To reduce the file size if this option was enabled accidentally or if one is concerned

about decreasing system performance and/or disk space one may simply delete the IBVIEW.INI file or the applicable registry keys. The next time the iButton Viewer is started, it will generate a new and short initialization file or new registry key without reporting an error message.

File Auto Overdrive (Options Menu)
If this option is checked the file viewer will attempt to communicate at a faster communication mode called Overdrive if the selected communication port and iButton device supports it. Overdrive-capable iButtons are the DS1995, DS1996 and DS1986. This option should be disabled if the length of the MicroLAN cable exceeds 10 meters. Refer to the introduction to review which adapters are capable of overdrive.

Notes
If the iButton Viewer is minimized, all open device viewer windows will disappear. To keep device viewers visible as icons, reduce the window of the iButton Viewer as far as possible but do not use the Windows minimize function.

On the iButton Viewer and on all device viewers there is a green "MicroLAN Activity" indicator. On the device viewers the activity indicator blinks while the particular device is accessed. If the indicator does not blink at all, then some other application has not released the MicroLAN. TMEX arbitrates the MicroLAN access by different applications and thus prevents collision. If a program is not acting correctly this error condition will persist.

4. WINDOWS 3.1 SECTION

I. Installation

General Information

Except for the installer SETUP.EXE, the files on the WINDOWS 3.1 version of the iButton-TMEX installation disk are compressed. The Windows 3.1 version of the iButton-TMEX 3.00 consists of nine release files with the release date and time code: IBVIEW.EXE, SETPORT.EXE, IBVIEW.HLP, LICENSE.TXT, README.WRI, IBFS.DLL, IB10E.DLL, IB97E.DLL , IB97U and IBTSR.DLL. The iButton Viewer IBVIEW.EXE requires IBFS.DLL which in turn calls on one of the hardware specific DLLs (IB10E.DLL or IB97E.DLL or IB97U.DLL or IBTSR.DLL) for the low level 1-Wire communication functions.

If not otherwise specified during the installation, the installer creates a directory \DALLAS\ on drive C:\ with a sub-directory \IBVIEW\ to where the first five of the re-lease files will be copied. The DLL files will be copied to the \WINDOWS\SYSTEM\ directory.

During installation, the setup-files IBUTTON.INI (always) and IBVIEW.INI (only if the iButton Viewer is launched) will be created in the \WINDOWS\ directory. The file IBUTTON.INI contains the TMEX default information for the type of the communica-tion port and the port number as well as the link to the DLL with the 1-Wire drivers. This file is revised automatically when re-installing the Windows 3.1 version of iButton-TMEX. The file IBVIEW.INI stores the actual detail setup information for the iButton Viewer IBVIEW.EXE. This file is maintained by the iButton Viewer and updated frequently. During the installation of iButton-TMEX the Windows initialization file WIN.INI is also updated. The added section can be identified by its headline "[iButton-TMEX]".

The Windows 3.1 version of the iButton-TMEX 3.00 comes with an uninstall pro-gram. This uninstaller consists of the program UNINST16.EXE and a DLL. The pro-gram is automatically copied to the \WINDOWS\ directory. The associated DLL and the file DEISL1.ISU which contains control information for the uninstaller are found in the same directory as the iButton Viewer. The installer and uninstaller maintain a file named PROTOCOL.INI which is found in the \WINDOWS\ directory.

Before upgrading to this version of iButton-TMEX always remove any previous version with its uninstaller. To use the uninstaller in Windows 3.1:

1. Double-click on the uninstaller icon
2. Confirm file deletion by clicking on "Yes'"

Welcome Screen
Before installing the Windows 3.1 version of iButton-TMEX, connect a 1-Wire adapter DS9097 or DS9097E to the COM port or a DS1410D or DS1410E adapter to the LPT port you will be using and note the actual port number. Close any application program that is currently running.

Insert the iButton-TMEX (16-Bit) Windows 3.1 installation disk, run the file manager, select your 3½" disk drive, and then double-click on SETUP.EXE. After the setup program is loaded you will get the Welcome screen. During the installation you can move from screen to screen by clicking on the button "Next >" or "< Back" or cancel the installation ("Cancel").

License Agreement
Right after the Welcome Screen the Software License Agreement is displayed. The license agreement can be scrolled through on screen or, after the installation is completed, be printed (LICENSE.TXT). The next screen provides general information on iButton-TMEX and a reminder to uninstall any earlier revision of the program before proceeding with this installation.

User Information
In one of the subsequent screens you are requested to enter your personal name and the name of the company where iButton-TMEX is being installed. Please enter your data correctly and proceed.

Destination Location
In the next step the installer asks for the Destination Location to where the program files should be copied. The default location is C:\DALLAS\IBVIEW\. You may enter any other destination directory using DOS conventions. If the directory does not yet exist it will be created. You may also click on "Browse..." and select an existing directory.

Program Folder
Windows allows the user to group programs arbitrarily into "Program Folders". If you accept the default shown on the Select Program Folder screen, the installer will create a new program folder "iButton-TMEX" with the icons for the iButton Viewer {32 Bit}, iButton-TMEX ReadMe {32-Bit} , the iButton Viewer Help {32-Bit} , the Default Microlan {32-Bit} utility, the Testcom serial port utility and the Testcom ReadMe. Alternatively you may create a new folder by typing its name or select an existing folder by scrolling through the list of folders.

NOTE
When adding iButton-TMEX to an existing folder, please be aware that, depending on other programs that are installed in the same folder, there may already be one or more identical icons for an uninstall program. Although the icons look identical, each is associated with a different program. Therefore, to avoid confusion at a later time, it is recommended to change the description of the uninstall icons in this folder as soon as the iButton-TMEX installation is completed. The description can be changed one icon at a time by selecting the icon and then clicking on the function "Properties" of the program manager's File menu. The contents of the "Command Line" field indicates what program is associated with the icon. After the program is identified, change the description to "XYZ Uninstall", where XYZ indicates the name of the program to be uninstalled.

Example:
If the command line is C:\WINDOWS\UNINST16.EXE -fC:\Dallas\iBView\DeIsL1.isu then the uninstaller with the name UNINST16.EXE located in C:\WINDOWS\ will remove \DALLAS\IBVIEW\ using DEISL1.ISU in \DALLAS\IBVIEW\ as control file.

MicroLAN Port Selection

After the Program Folder has been selected, one has to specify which of the ports with a 1-Wire adapter connected is to be defined as the TMEX default MicroLAN port. This is done through the Port Selection Screen. It is allowed to have more than one port adapter plugged in at a time. However, only one of them can be selected as the default.

On the Port Selection Screen select the tab of the hardware adapter that is to be used. Next enter the port number where the 1-Wire adapter was attached during the preparation of the installation (see Welcome Screen). After a port with adapter has been selected, the port number, adapter type and driver information will be displayed and the user can proceed to the next screen. If a port with no adapter connected is selected, you will get an error message.

Setup Complete

When you reach this screen, the iButton-TMEX installation under Windows 3.1 is completed. Before you click on "Finish" to close the installer you may click on one or both of the white boxes to access the README.WRI file and/or launch the iButton Viewer as the next application. If you click on both boxes the iButton Viewer will be started first with the README.WRI being displayed.
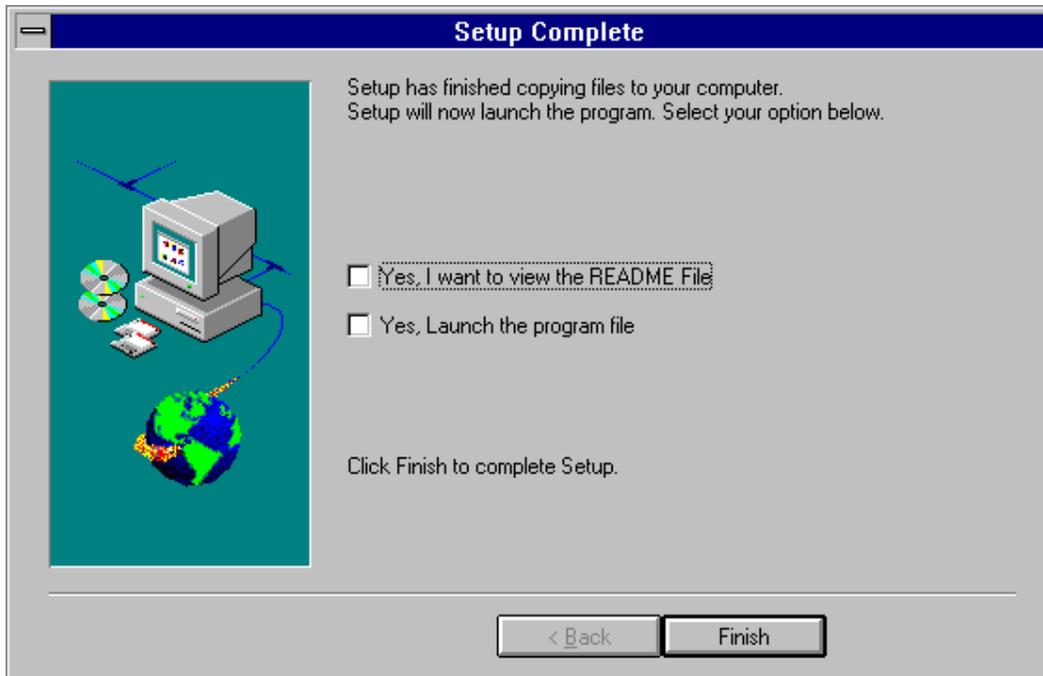
Notes

After iButton-TMEX is installed, the iButton Viewer and other TMEX programs (not included in this iButton-TMEX) can be used without having to run the iButton-TMEX installer again. If you change the 1-Wire port on your computer, however, the application software such as the viewer needs to be reconfigured. The iButton Viewer has a function to do this (see next section for details), but it does not change the TMEX default 1-Wire port in IBUTTON.INI.

If you want to change the default port selection, you can do one of the following:
a) close any TMEX-based program and run the Windows program SETPORT.EXE through the file manager or Program Manager
b) uninstall iButton-TMEX and then re-install it with the new port selection
c) edit the IBUTTON.INI file

Option a) should be preferred since it is easy, quick and does not require any special precaution. Option b) is also a good and safe way to go, but it takes more time. Option c) should only be chosen if one is very experienced in editing control files. If an error is made under option c) it may become necessary to uninstall iButton-TMEX and then re-install it from scratch.

.

II. iButton Viewer Windows 3.1x

The Windows 3.1x version of the iButton Viewer has the same user interface and features as the Windows 95/NT version of the program. Please refer to section 4.II of this document for the detailed program description.

5. DOS SECTION

I. Installation
General Information
The DOS installation disk is uncompressed. If not otherwise specified, the installer will create a directory \DALLAS\ on drive C:\ with three sub-directories \BIN\, \DOC\ and \DRIVERS\. In the sub-directory \DOC\ another sub-directory \TMEX\ will be created. The sub-directory \BIN\ will contain all the executables, \DRIVERS\ will contain the TMEX DOS TSR drivers and the documentation will be found in \DOCS\TMEX\.

After the directories are created, the files of the installation disk will be copied to their respective locations in these directories. Next, the device type and port number are chosen. In a final step the current AUTOEXEC.BAT file is copied to a backup file and then updated. A log-file in subdirectory \DOCS\TMEX\ lists all the operations that have taken place during installation. This log file also contains information on how to manually un-install the DOS version of iButton-TMEX.

Text messages appearing on the screen during installation are shown in the `courier` font. The **`underlined`** texts are default choices. Text entered by the user is shown in *`courier italics`*.

Welcome Screen
To install iButton-TMEX, exit any application program and go to a DOS prompt. Although the installer will work correctly if executed in the DOS Window of Windows 3.1 it is recommended to exit Windows during the installation.

Insert the iButton-TMEX DOS disk and at the DOS prompt type the name of your 3½" disk drive, e.g. *A:* and then press <enter>. If your 3½" floppy drive is any other letter, please substitute.

Next run the installer by typing *SETUP* and then pressing <enter>

You will get the Welcome screen

```
   Welcome to the DOS iButton-TMEX installation utility.
   This will install the Dallas Semiconductor iButton
   TMEX drivers for DOS.  These drivers will enable TMEX-
   ready iButton applications to run. Also included are
   multiple DOS iButton utilities.  This installation
   program can be run again with different options
   selected.

                   (Press any key to continue)
```

After pressing <enter> the next screen will appear.

Choose the TMEX directory on the hard drive.
The installer will attempt to put TMEX in the same directory as other TMEX products. The default directory for these products is C:\DALLAS\. However, any other hard drive and directory may be specified at the prompt. If the directory does not yet exist it will be created automatically.

```
  Please enter the drive and directory for the iButton-TMEX
  files.   This will be the same as with other iButton-TMEX kits.

                  (Press ENTER when complete)
 C:\DALLAS
```

To accept the default, press <enter>. To change the name of the subdirectory and/or the hard drive, erase the default name by using the backspace key ($\Leftarrow$, erase left) and then type the new name(s) according to the DOS conventions and press <enter>. If the input is incorrect or if any problem is found during the creation of the (sub-) directories, an appropriate error message will be displayed.

Get permission to overwrite all older files found

After the directories are created successfully, the installer asks for permission to override existing files with the same names. To avoid a mixed installation of new and old versions of programs and/or drivers, it is recommended to allow the installer to overwrite all older files found.

```
     Overwrite all older files?

     Choosing YES is recommended.
     Choosing NO will prompt for each overwrite.

              (Press Y or N to continue)
```

Since there is no default, either **Y** or **N** has to be entered. After this the installer will begin copying one file after another from the installation disk to the hard drive. Each file name will be displayed on screen as the file is copied.

```
 Copying file TDIR.EXE to:
          C:\DALLAS\BIN\TDIR.EXE
```

Select Default 1-Wire Port Type

After all files are copied, the installer asks for the type of port to be used, since it determines the type of TSR driver to be loaded. The selection is made through the following screen.

```
 Which iButton-TMEX port type is to be used as the default?
 The standard choices are COM port with the DS9097(E) adapter,
 the LPT port with the DS1410E adapter or
 the COM port with the DS9097U adapter.

 The fourth choice is for third party hardware TSR drivers.

                   1) COM    DS9097(E)
                   2) LPT    DS1410E
                   3) COMU   DS9097U
                   4) Other  (Third party)

           (Choose 1,2,3 or 4 and then press ENTER)
```

The default choice is the serial COM port. Currently only the COM port supports writing to EPROM based iButtons. Enter a number or use the ↑ or ↓ key to select and then press <enter>.

Select Default 1-Wire Port Number

After the port type is selected, the port number needs to be specified. This port number will be the default TMEX port number for the DOS utilities described later in this document. The port number selection screen may look like this.

```
What will be the default MicroLan port number?
For example a port number of 1 using the LPT
drivers would be LPT1.

          (Type the number from 1 to 3 and press ENTER)

                              1
```

Enter a valid port number and then press <enter>. When using the LPT port, it does not matter if a printer is connected, if the printer is powered up or online. When using the COM port, select a port number that is not occupied by the mouse or a modem. The DS9097(E) requires a true ± 12V RS232C port for operation.

After the port number is specified, the installer will copy the AUTOEXEC.BAT file to a backup file and then update the AUTOEXEC.BAT file according to the data entered during the installation. The name of the backup file is shown on the screen.

```
          Changes have been made to (C:\AUTOEXEC.BAT)
          The original was saved as (C:\AUTOEXEC.BK1)

                  (Press any key to continue)
```

The installer will maintain up to 10 versions of the backup file before older versions are overwritten.

To finish the installation, press <enter> and another message will appear, indicating the location of the installation log file.

```
iButton-TMEX Version 3.10 has been installed.

The following file contains the installation log and details
to allow uninstallation:
               C:\DALLAS\DOC\TMEX\INSTALL.LOG

If you would like to change any of the options then re-run
this installation utility.

After ending this program, quit Windows if you are in it and
re-boot the computer.

               (Press any key to END setup)
```

Re-booting

After the installer is finally exited by pressing a key, the computer must be re-booted to make the installation effective. When booting into DOS and the TMEX TSRs are installed successfully, one will get a confirmation message that may look like this.

```
Copyright (C) 1992-1996 Dallas Semiconductor Corporation
IBFS97E.EXE - iButton-TMEX DOS driver using the DS9097E on the
              COM port
            - Version 3.10
DOWF0305 V3.00 mm/dd/yy installed on interrupt 62H.
TMEXF04 V3.00 mm/dd/yy installed on interrupt 63H.
C:\>
```

The additions made to AUTOEXEC.BAT can easily be discovered. This is an example for an installation where the COM port number 2 was chosen.

```
REM **************** Begin TMEX Section ****************
PATH=C:\DALLAS\BIN;%PATH%
SET OneWirePort=2
SET TMEXDIR=C:\DALLAS
C:\DALLAS\DRIVERS\IBFS97E.EXE
REM **************** End   TMEX Section ****************
```

II. iButton-TMEX DOS Utility Programs

A. General Information
The programs described in the following pages provide file and directory functions similar to DOS operating system utilities. Before running any of these programs, an iButton-TMEX Version 3.00 DOS-TSR-driver must be installed for the port-type to be used. Note that **writing** to EPROM-based iButtons requires the use of the DS9097E COM Port Adapter.

Almost all of the programs allow the specification of the 1-Wire port number on the command line. However, since the port number is set during installation, usually there is no need to specify the port number on the command line. To verify the common default 1-Wire port number look for the parameter `OneWirePort` in the TMEX section of the AUTOEXEC.BAT file. Although the port number and port type can be changed by editing the AUTOEXEC.BAT file, this is not recommended. The safest way to change port type and port number is to re-run the installation program.

Each of the utility programs provides detailed syntax instructions when run with `?` as the first command line parameter. Optional parameters are denoted with `<>`.

The results of all of the utilities can be redirected into a file such as:
```
        TDIR   >RESULTS.TXT
```

With TMEX the format of a file name is 'NAME.XXX' where 'NAME' is 1 to 4 alpha-numeric characters. Valid characters for file names and directory names are:
> A B C D E F G H I J K L M N O P Q R S T U V W
> X Y Z 0 1 2 3 4 5 6 7 8 9 ! # $ % & ' - @ ^ _ ` { } ~

'XXX' is a decimal number extension in the range 0-99 for regular files and 100 for Addfiles (see TAPPEND). If no extension is specified, the value 0 is assumed.

B. Program Descriptions

TAPPEND DOS Equivalent: none

To create a special Addfile with extension 100 in a EPROM iButton and to append data at the end of the file without re-writing already existing data. This utility automatically formats a new iButton thus saving one page of memory.

Syntax
```
tappend source-file destination-file <OneWirePort>
```

Parameters

`source-file`	selects the DOS file to be appended to the destination file in the iButton.
`destination-file`	selects the Addfile in the iButton that will get the contents of the souce-file appended; the name must have the format NAME.xxx, where 'NAME' is up to a 4 digit alphanumeric name and XXX is a number extension of 100.
`OneWirePort`	selects the 1-Wire port for communication

TATTRIB DOS Equivalent: ATTRIB

To change the attribute of a file or sub-directory in the current directory on an iButton device. For iButton-TMEX Version 3.00, supported attributes are read-only for files (R) and hidden for sub-directories (H).

Syntax
```
tattrib (+R | -R) file-name <OneWirePort>
tattrib (+H | -H) directory-name <OneWirePort>
```

Parameters

`file-name`	selects the file to be modified; the name must have the format NAME.XXX, where 'NAME' is up to a 4 digit alphanumeric name and XXX is a number extension in the range 0-99.
`directory-name`	selects the directory to be modified; a directory name is up to 4 alphanumeric digits
`OneWirePort`	selects the 1-Wire port for communication

Options

`+R`	sets the attribute "read-only" for the selected file
`-R`	erases the attribute "read-only" for the selected file
`+H`	sets the attribute "hidden" for the selected directory
`-H`	erases the attribute "hidden" for the selected directory

TCD   **DOS Equivalent: CD**
To read or set the current directory in an iButton device. Running TCD with no ar-
gument displays the the default 1-Wire port and the current directory.

Syntax
```
tcd <<OneWirePort:>path>
```

Parameters
`OneWirePort:`   selects the 1-Wire port for communication
`path`   selects path for the directory; the path follows stan-
                        dard DOS path conventions. Note that port number
                        and path are separated by a colon if both parame-
                        ters are specified.

TCHK   **DOS Equivalent: none**
To check the file structure of an iButton device for inconsistencies and fragmenta-
tion. Each file is checked and its status displayed. Files can be de-fragmented and
errors be fixed with the utility TOPT, in most cases.

Syntax
```
tchk <OneWirePort>
```

Parameters
`OneWirePort:`   selects the 1-Wire port for communication

TCOPY   **DOS Equivalent: COPY**
To copy a DOS file to an iButton's current directory (only in this direction). For the
other direction use TTYPE and redirect the information to a DOS file.

Syntax
```
tcopy source-file destination-file <OneWirePort>
```

Parameters
`source-file`   selects the DOS file to be copied to the destination
                        file in the iButton.
`destination-`   selects the destination file in the iButton that will
`file`   hold the contents of the souce-file; the name must
                        have the format NAME.xxx, where 'NAME' is up to
                        a 4 digit alphanumeric name and XXX is a number
                        extension in the range of 0-99.
`OneWirePort:`   selects the 1-Wire port for communication

TDEL   **DOS Equivalent: DEL**

To delete a file from the current directory on the current iButton device. Note that this program can delete only a single file at a time. Use TFORMAT to delete all file entries at once.

Syntax
```
tdel filename <OneWirePort>
```

Parameters

`filename`   selects the file to be deleted from the current direc-
                          tory on the current iButton; the name must have the
                          format NAME.xxx, where 'NAME' is up to a 4 digit
                          alphanumeric name and XXX is a number exten-
                          sion in the range of 0-99.

`OneWirePort:`   selects the 1-Wire port for communication

TDIR   **DOS Equivalent: DIR**

To display the names of files in the current directory of an iButton device. This pro-gram also shows the length, starting location, and attributes of each file and sub-di-rectory.

Syntax
```
tdir <OneWirePort>
```

Parameters

`OneWirePort:`   selects the 1-Wire port for communication

TFORMAT   **DOS Equivalent: FORMAT**

To (quick) format an iButton device and prepare it for for storing files. This program deletes all file entries and directories regardless of their attributes. Formatting is also required for EPROM devices before they can accept data. However, they cannot be re-formatted once they are programmed.

Syntax
```
tformat <OneWirePort>
```

Parameters

`OneWirePort:`   selects the 1-Wire port for communication

TMD   **DOS Equivalent: MD or MKDIR**

To create a new sub-directory in the current directory.

Syntax
    `tmd directory-name <OneWirePort>`

Parameters
    `directory-name`   specifies the name of the directory to be created; a
                            directory name is 1 to 4 characters long and must
                            be unique for the current directory.
    `OneWirePort:`   selects the 1-Wire port for communication

TMEMCOPY   **DOS Equivalent: DISKCOPY**

To copy the file structure tree from one iButton device to another one. The destination iButton can be of a different type and memory size as long as it is large enough to store all of the files/directories of the source device. This program works in two steps: the first device probed will be regarded as source device and supply the data; the second device probed will be regarded as destination device and will be programmed.

Syntax
    `tmemcopy <OneWirePort>`

Parameters
    `OneWirePort:`   selects the 1-Wire port for communication

TOPT   **DOS Equivalent: none**

To check the file structure in an iButton device in the same way as in TCHK and to automatically correct errors and file fragmentations. Unfragmented files have their data is adjacent pages and therefore can be read in a single access. TOPT works only for NV-RAM-based iButton devices.

Syntax
    `topt <OneWirePort>`

Parameters
    `OneWirePort:`   selects the 1-Wire port for communication

TPEEK   **DOS Equivalent: none**

To report any TM_iButton EX TSR drivers loaded and the interrupts they are installed on.

Syntax
    `tpeek`

Parameters
    `(none)`   (N/A)

TRD   **DOS Equivalent: RD or RMDIR**

To remove a sub-directory from the current directory. The directory must be empty before it can be removed.

Syntax
    `trd directory-name <OneWirePort>`

Parameters
    `directory-name`   specifies the name of the directory to be removed.
    `OneWirePort:`   selects the 1-Wire port for communication

TREN   **DOS Equivalent: REN**

To rename a file in the current directory. The program can only rename a single file at a time.

Syntax
    `tren oldfilename newfilename <OneWirePort>`

Parameters
    `oldfilename`   selects the file to be renamed.
    `newfilename`   specifies the new name for the selected file; the
                        name must have the format NAME.xxx, where
                        'NAME' is up to a 4 digit alphanumeric name and
                        XXX is a number extension in the range of 0-99.
    `OneWirePort:`   selects the 1-Wire port for communication

TTEMP

This program reads the temperature from a DS1920/DS1820 iButton

Syntax ttemp <loop> <OneWirePort>

Parameters

`loop`	sets the program to run continuously until the user exits the program by pressing any key
`OneWirePort`	selects the 1-Wire port for communication

TTIME
This program reads/sets the time and date stamp from/to a DS1994 iButton.

Syntax
  ttime <OneWirePort> <dXX/XX/XX> <tXX:XX:XX.XX>

Parameters

`OneWirePort`	selects the 1-Wire port for communication
`dXX/XX/XX`	specifies the date
`tXX:XX:XX.XX`	specifies the time

Note: Date and time must both be provided or omitted.

TTREE   DOS Equivalent: TREE
To display a list of all files and sub-directories in an iButton device.

Syntax
  ttree <OneWirePort>

Parameters

`OneWirePort:`	selects the 1-Wire port for communication

TTYPE   DOS Equivalent: TYPE
To copy the contents of the selected file in the current directory to the standard output device (computer screen). Together with output redirection this program will copy a file from a iButton device to a DOS file.

Syntax
  ttype filename <OneWirePort>

Parameters

`filename`	selects the file to be copied or displayed
`OneWirePort:`	selects the 1-Wire port for communication

TVIEW   DOS Equivalent: (DOSSHELL, to some extent)

To display the names, directories and device information of an iButton device and to select and display the file contents. Files are selected by using the arrow keys ($\uparrow \downarrow$) and then pressing the ENTER key. Selecting a sub-directory will result in the current directory changing to that sub-directory and the new file names being displayed. This version of TVIEW has a limit of only displaying 13 files/directories at a time.

Syntax
```
tview <OneWirePort> <bw>
```

Parameters

`OneWirePort:`	selects the 1-Wire port for communication
`bw`	changes the display mode into a black and white mode for monochrome displays such as LCDs.