WINDOWS

WINDOWS 9640

INSTALLATION

Package Components

- Reference Manual
- WINDOWS Diskette

The enclosed diskette contains your personalized copy of WINDOWS 9640 with your name and serial number embedded. Attempting to modify this information under certain conditions will either lock your system up on initial loading, or could cause SEVERE results after having used WINDOWS for any length of time.

- To Install WINDOWS on a Hard Drive System

A batch file has been written (Filename = INSTALL) to support the creation of a sub-directory (WIN) and copies all the necessary files to the hard drive. A batch command file (DIS/VAR 80 format) called WINDOWS will load and run the necessary software, when called from the root directory of the hard drive.

- Getting Started

Loading WINDOWS 9640

WINDOWS 9640 currently supports both the keyboard (discussed later) and the mouse port on the 9640 using an independent mouse driver. Future support is being looked at for the use of joystick, trackball, and Asgard's RS232 based mouse as standalone drivers that will replicate the Myarc Mouse, but no promises or guarantees are being made.

- Using a mouse driver

The mouse driver enclosed is a standalone product developed by Bruce Hellstrom and used in conjunction with WINDOWS and any other WINDOWS developed software.

The mouse driver (Filename = MOUSE), controls all mouse displays when in an active state. For additional information regarding the mouse driver, a programming package will be issued in the future with all relative information.

- To Execute without a Driver

To execute WINDOWS without a driver installed, WINDOWS will prompt a question if you are using a driver, the built in driver, or the keyboard. Selecting the keyboard makes the following keys active, the 4 arrow keys, F8, F9, and F10. The latter three keys will correspond to the Left, Middle, and Right buttons of the mouse in all subsequent discussion and will not be

repeated again.

Selecting D for Default driver at the initial loading screen, will provide the necessary mouse code for WINDOWS for use in Graphics modes. Under certain conditions, a newly loaded program may elect to change graphics/text modes and make the mouse pointer invisible. During this time that the mouse is invisible, it is still active, but difficult to control. The mouse driver solves these problems and is recommended for all usage as it provides support under these circumstances by providing a reference symbol to indicate the position of the mouse. These same conditions also hold true when using the keyboard.

A General Discussion of WINDOWS

WINDOWS 9640 is a program with multiple functions and features. As written, WINDOWS itself requires 64K of user memory, and the driver, if present, requires an additional 16K. With more memory that is available, obtained under the following circumstances, the greater the capabilities of WINDOWS;

1) User has disabled TIMODE
2) User has no internal RAMDISK or SPOOLer
3) User has added the additional 32K, 0 wait state memory.
4) User has modified the Myarc 512K memory expansion card.
5) User has the MEMEX card by Ron Walters / Bud Mills.

Once WINDOWS has loaded, your serial number and the owner of the original copy will be displayed if a driver is not installed.

CAUTION: DO NOT MAKE ANY ATTEMPT TO MODIFY THIS INFORMATION. ATTEMPTING TO DO SO, WILL CAUSE SEVERE PROBLEMS NOT ALWAYS IMMEDIATELY NOTICED!!!!

You will then be prompted for the use of the mouse driver, keyboard, or the built in mouse default code.

Select at the prompt:

Y = Mouse Driver RECOMMENDED, if installed
N = No driver, use keyboard
D = Use default driver

In order to resolve the many various MDOS versions that users have in their possession, numerous tables are polled to obtain your system identification. If a mismatch occurs, the initial prompt after loading WINDOWS will comment that "DOS WINDOS IS NOT INSTALLED". This means the ability to toggle to MDOS from WINDOWS and back is not active. More on this later.

In addition, WINDOWS will read your system screen parameters. Those people that boot in 40 column text mode will have their system screen tables modified to the lower resolution. If you boot in 80 column text mode (or anything other 40 column text), you will be permitted to use the maximum capabilities of WINDOWS. Composite and TV monitors will experience problems using this higher resolution. Use your judgement as conditions vary from

monitor to monitor.

If you have made it this far, then the power you possess is ready to be released. With the initial option screen, you will see the following:

1) Repartition Screen
2) System Information
3) File Options
4) Disk Options
5) Load Task
6) MDOS Window
7) Exit - DeInstall

How to Select an Option

In order to resolve the many different keyboards, mouses, and other future drivers, all options on the menus are selected either thru the keyboard with the arrow keys and F8, F9, F10, or the drivers that exist. The right mouse button (or F10 key) corresponds to the equivalent of an <ENTER> keypress and automatically processes the option that it was on. The left mouse button (or F8 key) backs the option out one menu with it being active most of the time the mouse is being displayed. In addition, any option that prompts for a keypress to initiate a command, the ESC is also active to void that command, along with the equivalent of the left mouse button (or F8 key). The middle mouse button is only used in a few special cases and will be commented later when it is needed.

1) Repartition Screen

1) Set Window Mode
   - Resets WINDOWS display to your default parameters
   - Used when another program modifies the screen tables
2) Set 80 Columns
   - Sets screen to 80 column text mode
3) Set 40 Columns
   - Sets screen to 40 column text mode
4) Toggle Tasks
   - This option allows one to toggle any program that has been previously loaded with the LOAD TASK option (initial menu, opt # 5) into the run state (multitasking with any other program)
   - multiple programs can be selected from this screen, if they are already loaded.

NOTE: This option (Toggle Tasks) can be tricky for various reasons as described below.

a) To exit from this WINDOW, select either WINDOWS or the left mouse button. If you select WINDOWS, this disables all other tasks. If you select the left mouse button, you will then disable the majority of WINDOWS routines and activate those programs selected, until you re-activate it again. This is done by holding the LEFT ALT key and then pressing the LEFT SHIFT key with the LEFT ALT in the down position. Pressing this key sequence disables all programs except for WINDOWS and turns control of the computer back to WINDOWS.

5) Park Hard drives
- this feature pulls up an additional window of various hard drives (both the Wincheseter personality card and the HFDC by Myarc) that effectively forces the system to read the very last accessible sector on the hard drive and then positions the hard drive head over that last block of information. It the drive were to then crash, the only information damaged would be that block that is empty, unless you have a 99% packed hard disk.

6) Resize Window
- this option will only work for those programs that have been written to use WINDOWS. This option will not control the video displays and keyboard inputs associated with programs that do not access WINDOWS special code.
- to use this option, the middle button on the mouse selects your upper left corner of the window, and right button selects the lower right corner, then pressing the left button exits this resize option.

7) Toggle Screen
- toggles to video page 2 with right mouse button.
- Use middle mouse button to toggle back to the normal page for Windows (this option used to look at graphics that have been used by other programs and left button to return to a display menu.

8) Exit - DEINSTALL WINDOWS
- exits back to MDOS, or to any current running task.

2) System Information

Selection of this option displays the current version, the legal owner's name, serial number, and the author of WINDOWS.
- press any key or mouse button to exit.

3) File Options
1) Directory
- Used to catalog any floppy device
2) Unprotect File
- type in any valid MDOS Filename and path and the file will be unprotected
3) Delete File
- type in any valid MDOS Filename and path and the file will be deleted.
4) Protect File
- type in any valid MDOS Filename and path and the file will be protected.
5) View File
- type in any valid MDOS Filename and path and the screen will be cleared and toggled to 80 colum

text mode and the file displayed, 20 lines at a time.
6) Rename File
- type in any valid MDOS Filename and path for the file to be renamed, and then ONLY the new Filename WITHOUT the path for the new name.

4) Disk Operations
1) Format Floppy
- selects from menu's the drive you wish to format and then the floppy specifications (SS/SD, DS/SD, DS/DD, and DS/QD). At the final prompt, the LEFT mouse button, or the ESC key will exit the options without implementing the routine.
2) Clone Floppy
- selects from menu's the source and target drives you wish to clone. This option requires that the target drive have enough sectors available for the copy (Do not copy a DS/SD disk to a SS/SD disk, or to the same drive).
3) Sweep Floppy
- erases all files on any standard floppy disk format if it is SS/SD, DS/SD, DS/DD, or DS/QD. Other formats are not supported as they may contain 'special' information.
4) Create Dir
- creates a directory with the specified name entered on any device.

ex, to create a Directory called FORTRAN on DSK1, the following formats are legal

A:FORTRAN
DSK1.FORTRAN

5) Remove Dir
- removes an empty directory from any device

ex, to remove a Directory called FORTRAN on DSK1, the following formats are legal

A:FORTRAN
DSK1.FORTRAN

6) Rename Disk
- displays the volume name of any disk pointed to and prompts for a new disk name. If nothing is entered, the name is not modified

7) Rename Dir
- renames a directory on any device or pathname

ex, to rename a directory called TELCO on your hard drive system with the following pathname, F:COMM.TELCO, you would enter at the first prompt any of the

following formats:

F:COMM.TELCO
HDS1.COMM.TELCO

and then specify the new directory name excluding the pathname of the file, such as XMODEM (10 chars max)

5) Load Task

This option is one of the most powerful features and will sometimes be very confusing unless events are planned. Described below is what the LOAD Task can and can not do.

CAN DO

Any program that DOES NOT depend on command line arguments may be loaded thru this option. Any program currently written that resets the screen mode will have some difficulty. Many programs reset the mode, or modify it, but WINDOWS has tried to resolve these problems in it's coding whenever it can.

WINDOWS is designed to run as what may be referenced as the MDos Shell of multiple programs. Programs are under development and will be released in 9640 News, a disk magazine by the author of this program that will utilize the functions and capabilities of WINDOWS.

In addition, other notable programmers have already requested information on using Window's routines.

CAN'T DO

WINDOWS can not 'multi-task' with programs that have set the Geneve's interrupts to an OFF state. An example of such a program is the GPL Interpreter for the 9640. GPL can be loaded and executed thru WINDOWS, and upon exit of GPL, Windows will be active, but not when GPL is still running. One option of WINDOWS, called RESIZE Window, will not function with any previously written MDOS program, or any program that does not call and request WINDOW's special code. Those programs that do request the necessary code will have no problems, with the exception of WINDOWS itself which takes care of itself.

Programs that write directly to the Video Display Processor that do not use the recommended MDOS display routines may have problems, when screen modes are changed.

With the memory constraints of WINDOWS and attempting to still permit users to have enough CPU available to run two additional programs simultaneously, support for saving the screen images of each program was not provided. Future support is being looked into as the amount of available CPU increases for Memory Expanded systems, and for users that have opted to expand with one of several (?) video expansion cards that may be developed.

6) MDOS Window

Permits toggling to the MDOS Command Line Interpreter (CLI) and performing any needed activity. Loading any program while in this MDOS Window, will permit the CLI to multi-task with the newly loaded program until which time you press the LEFT ALT key and remain holding and pressing the LEFT SHIFT key. Windows and the newly loaded program (if you loaded one), would then multi-task with one another. Using this option to load a program removes control from WINDOWS of controlling that newly program.

7) Exit - Deinstall

Exits to MDOS. Use the & (AMPERSAND) key or an AUTOEXEC file to recover additional memory that WINDOWS or a driver will no longer be using.

APPENDIX A

Listed below is the available information being made with the purchase of WINDOWS 9640 on using the Assembly language support code. WINDOWS support code can be accessed using the MDOS XOP call in the following format:

```
ELEVEN DATA 11
START  .       (Your paramaters and code)
       XOP  @ELEVEN,0
```

Programs that utilize any WINDOWS supported code should not use any of the following memory space of any program they write.

>0000 to >03FF
>E000 to >E9FF
>F040 to >F214

The rest of available memory is available. These memory chunks being reserved represent buffers, internal workspaces, and other tables that are necessary to control a program in the multi-tasking state. Writing any information into these areas could be harmful to any screen display.

Due to MDOS's internal structure and requirement and the additional constraints needed for WINDOWS, every task must have an available page of memory mapped into the address range of >E000 to >FFFF with a workspace located in one of the available locations of that page of memory.

Detailed below is information regarding the MDOS assembly language support code.

```
*****************************************************
* Opcode >00   Acquire Multi-tasking page for Windows 9640*
*      R1 = MUST BE CLEARED FOR FUTURE EXPANSION         *
*      R2 = MUST BE CLEARED FOR FUTURE EXPANSION         *
*                                                        *
*****************************************************
```

- Sets up the necessary screen tables and buffers necessary for WINDOWS to multi-task. This should be the first opcode called of any access to this library from each running program.

```
*******************************************************
* OPCODE >01        TEXT DISPLAY                      *
* TEXT AT >E400 is automatically displayed, null terminated
*                                                     *
* CLR R1 FOR EXPANSION                                *
* CLR R2 FOR EXPANSION                                *
*                                                     *
*******************************************************
```

- text should be located at >E400, null terminated (it's o.k. to write in this space (254 bytes max) for any displays to the appropriate window.

```
*******************************************************
* KEYIN ROUTINE FOR ACCEPTING/PRINTING TEXT           *
* R0 =>02                                             *
* R1 = STRING LENGTH                                  *
* CLR R2 FOR EXPANSION                                *
* CLR R3 FOR EXPANSION                                *
*                                                     *
* RETURNED                                            *
* Beginning at >E100 is the string entered, null terminated
*******************************************************
```

- this opcode invokes a routine that will accept and display keyboard input (and store it at >E100) with delete, backspace, and other multiple functions commonly associated with normal keyboard entry.

- string is null terminated (254 bytes max)

```
*******************************************************
* Opcode >03   Forces window to your defined positions *
* R0 = >03                                            *
* R1 = top row                                        *
* R2 = left column                                    *
* R3 = bottom row                                     *
* R4 = right column                                   *
*******************************************************
```

- this opcode redefines your WINDOW to the new parameters that you define without regard to any other program running.

```
*******************************************************
* Opcode >04      no parameters                       *
* Shut multi-tasking off                              *
*******************************************************
```

- turn multi-tasking capability off in your program.

```
*******************************************************
* Opcode >05      no parameters                       *
* Turn multi-tasking on                               *
*******************************************************
```



```
*******************************************************
```

- turn multi-tasking capability on in your program if it was previously turned off. The default loading parameters are always on.

```
*******************************************************
* Opcode >06    40/80 screen Toggle    WINDOWS call only *
*                                                     *
*******************************************************
```

- this call, usable only within WINDOWS, toggles the buffers within the XOP to be mapped for the appropriate screen positions. Not usable in any other program.

```
*******************************************************
* Opcode >07    Drop Task Windows and parameters      *
*  Removes WINDOWS's task from XOP code               *
*  No   parameters required                           *
*******************************************************
```

- removes all buffers from Windows internal code when it was called with opcode >00

```
*******************************************************
* OP >08    BOXIT WITH PARAMETERS PASSED FOR LOCATIONS *
* Used to frame in Graphics mode a box for later text *
* R0 = >08                                   printing *
* R1 = TLCX    top left corner X                      *
* R2 = TLCY    top left corner y                      *
* R3 = BRCX    bottom right corner x                  *
* R4 = BRCY    bottom right corner y                  *
* R5 = COLORS  foreground/background colors  >ffbb    *
* CLR R6 FOR EXPANSION                                *
*******************************************************
```

- this option displays a framed WINDOW that is seen in WINDOWS frequently.
- use OP >02 to print text into a newly defined box

```
*******************************************************
* Opcode >09            Drop Last called Window       *
* Erase WINDOW display with last background color     *
*      R0 = >09                                       *
*******************************************************
```

- use this opcode to erase the Window created with opcode >08

```
*******************************************************
* Opcode >0A                                          *
* Activate WINDOWS 9640 from Sleeping state           *
*******************************************************
```

- take WINDOWS into multi-tasking state

```
*******************************************************
* Opcode >0B                                          *
* Deactivate WINDOWS 9640 to sleeping state           *
*******************************************************
```

- take WINDOWS out of multi-tasking state and only allow your program to run.

```
##############################################################
# Opcode >0D    Task 0 pages                                 #
# Returns at >E610 list of all tasks 0 pages                 #
##############################################################
```

- returns a list at >E610 of all task header pages called from Opcode >00, 8 bytes in length

```
##############################################################
# Opcode >0E    Task 0 pages                                 #
# Returns at >E618 list of all tasks 7 pages                 #
##############################################################
```

- returns a list at >E618 of all WINDOW pages with page 7 (>E000 to >FFFF) for WINDOWS internal support of various mapping across programs

Notes from the Author

WINDOWS 9640 has expanded further than it was ever dreamed about for several reasons. The majority of those reasons have been from YOU asking for options, utilities, features to further enhance this program.

In it's original concept, WINDOWS, was being developed to provide supporting software for multi-tasking on the Geneve without each program playing havoc with the screen, keyboard, cursor position, and numerous other 'problems'. Those problems have been solved if you are using WINDOWS code. Controlling the entire Geneve system has been extremely difficult, with the limited information that has been made public.

I would like to thank the following people for their contributions that have been invaluable to producing WINDOWS. They are:

Paul Charlton, whom I first consulted to see if what I was thinking about developing was even 'do-a-ble' on the Geneve, and some necessary information that made it possible for the MDOS Windows toggle to be MDOS version independent.

Bruce Hellstrom, the author of the first interrupt driven, mouse driver for the Geneve that is providing the driver for inclusion in WINDOWS, and some supporting information learned about the keyboard and direct access of the keyboard.

Mike Dodd, a talented young individual that provided some additional insights into the keyboard and direct access of the keyboard.

Al Beard, the author of 9640 Fortran, that provided beta testing, new ideas, and concepts to be used in WINDOWS and the author of a MDOS debugger that helped progress WINDOWS thru the initial stages.

Tom Freeman, the author of DISkASSEMBLER, whom I am indebted with much gratitude for developing a fine program that permitted the exploration of the Geneve in great detail when a lot of information was still missing.

Michelle Miller, my wife that has allowed me spend many hours programming which I am deeply indebted too. Thanks Sheli.....

And to anyone else that I may have left out, thank you too.

As I sit here putting the final touches to WINDOWS, many things can still be done, and most likely will be. When software updates occur, to either correct bugs that may be found (I hope there aren't any), to include new suggestions, or to add additional supporting multi-tasking code, you will be notified with a postcard and a minimal update fee if required. And to close this version, I thank you for your purchase of WINDOWS 9640.

########## Updates and Warranty ##########

Warranty - Any defective diskette or booklet will be repaired or replaced if returned to 9640 News within 90 days. 9640 News will make the judgement as to whether the diskette or software is defective and will either repair or replace it at our option. PLEASE MAKE AND USE A BACK UP COPY of your diskette for your own personal use. 9640 News is not responsible for any damage incurred by the use of WINDOWS 9640. 9640 News makes no warrants of the usefulness of this program with any person's particular needs.