**User’s Manual**

**Thunder Krackens**

**MODULE R6**

Jenn Nguyen

Haofan Zheng

Adam Chandler

Jafar Alkusaimi

4/22//2016

**Welcome to Thunder Kracken’s MPX**

MPX is an operating system created for academic purposes. Each member of the design team has taken careful steps to insure the quality of this operating system. We hope you enjoy using our operating system.

**Introduction**

As you begin to run MPX, you’ll be faced with a terminal. This terminal is the primary interface of MPX. Everything you do will be done through this terminal, so it is best to get familiar with it. MPX’s terminal functions much like other terminals you may be familiar with; to write in the terminal, use the keyboard to input letters. You can edit what you have already wrote in the terminal by navigating with the left and right arrow keys in their respective directions. To undo mistakes, the backspace button removes the character to the left of the cursor and the delete button removes the character that the cursor is currently highlighting. Once you are familiar with the input, you are ready to use MPX. Using MPX requires the ability to input commands to execute functions. These commands are listed below. Note that many commands begin with the subject they deal with. Functions beginning with “mpx” are general commands to be used in mpx. Functions beginning with “pcb” deal with process control blocks. Functions beginning with “mcb” deal with memory control blocks.

**Commands**

This section is comprised of different commands to be used in the terminal. Each command is formatted with its name followed by a colon and then it’s syntax. Note that square brackets indicate a variable clause.

show: mcb show -allocated

Displays the address and size of allocated MCBs.

Exit Status: Always succeeds.

show: mcb show -free

Displays the address and size of allocated MCBs.

Exit Status: Always succeeds.

getdate: mpx getdate

Displays the current date to the user in format month/day/year.

Exit Status: Always succeeds.

gettime: mpx gettime

Displays the current time to the user in format hours:minutes:seconds.

Exit Status: Always succeeds.

help: help [command]

If used alone, displays usage instructions for every command or when used with an individual command, gives a detailed summary of how to use that command.

Arguments:

command The command to be described

Exit Status:

Returns success unless the given command is not a valid command.

LoadR3: mpx loadr3 [priority]

Allows the user to load all r3 processes into memory in the suspended ready state at the given priority.

Arguments:

priority The priority at which to load all of the r3 processes

Exit Status: Succeeds unless the given priority is not in the correct range of values.

setdate: mpx setdate [MM]/[DD]/[YYYY]

Allows the user to set the date to a given date.

Arguments:

MM Specified month must be two digits long.

DD Specified day must be two digits long.

YYYY Specified year must be four digits long.

Exit Status: Return success unless the given month, day, or year is an incorrect value.

settime: mpx settime [HH]:[MM]:[SS]

Allows the user to set the time to a given time.

Arguments:

HH Specified hours ranging from 00 (midnight) to 23 (1AM).

MM Specified minutes.

SS Specified seconds.

Exit Status: Return success unless the given time is invalid values.

shutdown: mpx -shutdown

Shuts down the operating system.

Exit Status: Always succeeds.

version: mpx version

Prints the current version of MPX and completion date.

Exit Status: Always succeeds.

resume: pcb resume [processName]

Puts the given process into the unsuspended state.

Arguments:

processName The name of the process to be unsuspended.

Exit Status: Return Success unless the given name does not apply to any process.

setpriority: pcb setpriority [processName] [processPriority]

Allows the user to change the priority of a process.

Arguments:

processName The name of the process

processPriority The new priority value (Must be a digit 0 – 9)

Exit Status Return success unless there is no process by the given name, priority is not a digit 0-9, or the process already has the given priority.

show -all: pcb show -all

Shows all processes and their attributes.

Exit Status: Always succeeds.

show -blocked: pcb show -blocked

Shows all blocked processes and their attributes.

Exit Status: Always succeeds.

show -ready: pcb show -ready

Shows all ready processes and their attributes.

Exit Status: Always succeeds.

suspend: pcb suspend [processName]

Puts the given process into the suspended state.

Arguments:

processName The name of the process to be suspended.

Exit Status: Return Success unless the given name does not apply to any process

**Welcome to Thunder Kracken’s FAT12 File System**

This system is used to navigate files on using a disk image of a FAT12 storage device such as a floppy disk. This system is used much like the MPX. A command line terminal is used in the same way as mpx, allowing for cursor navigation and backspaces. The functions of this FAT12 File System are executed through the following commands. Note that all function names can be followed by “--help” to receive information on how to use them. Also, note that square brackets indicate a variable clause.

Change Directory: cd [directory]

Changes the current directory to the given directory.

Arguments

directory Name of the directory to become the current directory.

Exit Status: Succeeds unless the given directory does not exist.

Exit: exit

Prompts the user to save changes upon exiting and allows the user to exit the FAT12 System.

Exit Status: Always Succeeds.

List Directory Short: ls

Displays all of the files and directories within the current directory.

Exit Status: Always Succeeds.

List Directory Long: ls -l

Displays all of the files and directories within the current directory along with their respective filename, extension, logical file size, and starting logical cluster.

Exit Status: Always Succeeds.

List Root Directory Information: ls -r

Displays the files and directories inside the root directory.

Exit Status: Always Succeeds.

Move: mv -i [path1] [path2]

import a file from outside of disk image to the disk image.

Arguments:

Path1 The file you want to import.

Path2 The destination directory

Exit Status: Returns success unless the file does not exist or the destination is not valid

Move: mv -m [path1] [path2]

Moves a file within a disk image.

Arguments:

Path1 The file to be moved

Path2 The destination directory

Exit Status: Returns success unless the file does not exist or the destination is not valid

Move: mv -o [path1] [path2]

Extract a file from disk image to outside of disk image.

Arguments:

Path1 The file you want to extract

Path2 the destination file path (should be a valid path to outside of disk image).

Exit Status: Returns success unless the file does not exist or the destination is not valid

Print Boot Sector Information: pb

Displays information from the boot sector.

Exit Status: Always Succeeds.

Rename: rn [file] [name]

Renames the given file to the given name.

Arguments:

file The original name of the file to be renamed

name The new name of the file

Exit Status: Succeeds unless the file cannot be found or another file is already named the new name.

Type: less [file]

Prints the contents of any file ending with “TXT”, “BAT”, or “C” page by page.

Arguments:

file The name of the file that is to be read.

Exit Status: Succeeds unless the file is not the correct file type or does not exist.

Write: write

Prompts the user to save changes onto the disk image.

Exit Status: Always succeeds.