



# LogicLoader™ Labs for the DM3730/AM3703 SOM-LV, DM3730/AM3703 Torpedo SOM, and DM3730/AM3703 Torpedo + Wireless SOM

LogicLoader Documentation (LogicLoader version 2.5)

Logic PD // Products  
Published: April 2012  
Last revised: June 2012

## Abstract

These informal LogicLoader™ labs provide an introduction to basic LogicLoader commands and usage with the DM3730/AM3703 SOM-LV, DM3730/AM3703 Torpedo SOM, and DM3730/AM3703 Torpedo + Wireless SOM.

This document contains valuable proprietary and confidential information and the attached file contains source code, ideas, and techniques that are owned by Logic PD, Inc. (collectively "Logic PD's Proprietary Information"). Logic PD's Proprietary Information may not be used by or disclosed to any third party except under written license from Logic PD, Inc.

Logic PD, Inc. makes no representation or warranties of any nature or kind regarding Logic PD's Proprietary Information or any products offered by Logic PD, Inc. Logic PD's Proprietary Information is disclosed herein pursuant and subject to the terms and conditions of a duly executed license or agreement to purchase or lease equipment. The only warranties made by Logic PD, Inc., if any, with respect to any products described in this document are set forth in such license or agreement. Logic PD, Inc. shall have no liability of any kind, express or implied, arising out of the use of the Information in this document, including direct, indirect, special or consequential damages.

Logic PD, Inc. may have patents, patent applications, trademarks, copyrights, trade secrets, or other intellectual property rights pertaining to Logic PD's Proprietary Information and products described in this document (collectively "Logic PD's Intellectual Property"). Except as expressly provided in any written license or agreement from Logic PD, Inc., this document and the information contained therein does not create any license to Logic PD's Intellectual Property.

The Information contained herein is subject to change without notice. Revisions may be issued regarding changes and/or additions.

© Copyright 2012, Logic PD, Inc. All Rights Reserved.

## Revision History

| REV | EDITOR | DESCRIPTION   | LoLo VER. | APPROVAL | DATE     |
|-----|--------|---|-----------|----------|----------|
| A   | SB, GJ | -Initial Release  | 2.5.1     | SB, GJ   | 04/19/12 |
| B   | SO     | -Throughout: Added information for AM3703 configuration of the Torpedo + Wireless SOM | 2.5.2     | SO       | 06/14/12 |

# Table of Contents

|  |           |
|--|-----------|
| <b>1. Introduction</b>   | <b>1</b>  |
| 1.1 Nomenclature, Notices, and Conventions Used in This Document           | 1         |
| <b>2. Lab 1: Establishing Kit Communications</b>                           | <b>2</b>  |
| 2.1 Lab Goals  | 2         |
| 2.2 Prerequisites  | 2         |
| 2.3 Basic Instructions   | 2         |
| 2.4 Tera Term instructions   | 2         |
| 2.5 Connect the Kit to Your Host PC  | 5         |
| <b>3. Lab 2: Learning the LogicLoader Help System</b>                      | <b>6</b>  |
| 3.1 Lab Goals  | 6         |
| 3.2 Prerequisites  | 6         |
| 3.3 LogicLoader Interface  | 6         |
| 3.4 Help for Incorrect Commands  | 7         |
| 3.5 Using the <i>help</i> Command  | 8         |
| <b>4. Lab 3: Using the LogicLoader Command Line Editor</b>                 | <b>13</b> |
| 4.1 Lab Goals  | 13        |
| 4.2 Prerequisites  | 13        |
| 4.3 Instructions   | 13        |
| <b>5. Lab 4: Updating LogicLoader and Recovery</b>                         | <b>17</b> |
| 5.1 Lab Goals  | 17        |
| 5.2 Prerequisites  | 17        |
| 5.3 Connect the Kit to Your Host PC  | 17        |
| 5.4 Download a New Version of LogicLoader                                  | 17        |
| 5.5 Update LogicLoader using a Bootable SD Card                            | 19        |
| 5.5.1 Prepare the SD Card  | 19        |
| 5.5.2 Use the SD Card to Update LogicLoader                                | 19        |
| <b>6. Lab 5: Establishing a Network Connection</b>                         | <b>24</b> |
| 6.1 Lab Goals  | 24        |
| 6.2 Prerequisites  | 24        |
| 6.3 Connect the Kit to Your Host PC  | 24        |
| 6.4 Connect the Kit to Ethernet  | 24        |
| 6.4.1 Network Setup for Ethernet Cross-over Cable and Static IP Addressing | 25        |
| 6.4.2 Network Setup for Standard Ethernet Cable and DHCP Addressing        | 29        |
| 6.4.3 Use the <i>ping</i> Command to Check the Connection                  | 30        |
| 6.4.4 Use the <i>ifmac</i> Command to Display and Change the MAC Address   | 31        |
| <b>7. Lab 6: Utilizing File Systems</b>                                    | <b>33</b> |
| 7.1 Lab Goals  | 33        |
| 7.2 Prerequisites  | 33        |
| 7.3 Fat File System (FATFS)  | 33        |
| 7.3.1 Mount an SD Card   | 33        |
| 7.4 Yet Another Flash File System (YAFFS)                                  | 35        |
| 7.4.1 Develop a Partition Scheme   | 35        |
| 7.4.2 Format YAFFS Partitions  | 35        |
| 7.4.3 Add YAFFS Type Partitions  | 38        |
| 7.4.4 Mount a Partition  | 39        |
| 7.4.5 Copy a File to the Partition   | 40        |
| <b>8. Lab 7: Using Scripts</b>   | <b>41</b> |
| 8.1 Lab Goals  | 41        |
| 8.2 Prerequisites  | 41        |
| 8.3 Connect the Kit to Your Host PC  | 41        |
| 8.4 Basic Scripting Guidelines   | 41        |
| 8.5 Launching Scripts from a Text File                                     | 42        |

8.6 Launching Scripts from an SD Card ..... 45

8.7 Persistent Script Storage ..... 46

**9. Lab 8: Loading an *NK.BIN* File ..... 48**

9.1 Lab Goals ..... 48

9.2 Prerequisites ..... 48

9.3 Connect the Kit to Your Host PC ..... 48

9.4 Load a File from an SD Card ..... 48

9.5 Load a File from YAFFS..... 50

9.6 Execute a Windows CE Image ..... 52

9.7 Load *NK.BIN* and Create a DHCP Network Connection with a Persistent Script ..... 52

# 1. Introduction

These labs provide an informal introduction to the LogicLoader bootloader software. The step-by-step directions act as a means to discover how LogicLoader works with Logic PD System on Modules (SOM).

## 1.1 Nomenclature, Notices, and Conventions Used in This Document

- Within this document DM3730 is used to denote both the DM3730 and AM3703 processors.
- All labs are applicable to the DM3730/AM3703 SOM-LV, DM3730/AM3703 Torpedo SOM, and DM3730/AM3703 Torpedo + Wireless SOM. Use of “DM3730 SOM” suggests text that applies to all three platforms; information specific to one platform will call out the precise name.
- Use of “DM3730 Development Kit” suggests text that applies to both the DM3730 SOM-LV Development Kit and DM3730 Torpedo Development Kit; information specific to one development kit will call out the precise name.

## 2. Lab 1: Establishing Kit Communications

### 2.1 Lab Goals

This lab will demonstrate how to establish a serial connection to your DM3730 Development Kit. A serial connection is the most basic form of communication between a host PC and the development kit.

### 2.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- A terminal emulation program (e.g., Tera Term)

### 2.3 Basic Instructions

If you are familiar with serial communications and a terminal emulation program, open your terminal emulator and set the port settings to:

1. Baud rate: **115200**
2. Data: **8 bit**
3. Parity: **None**
4. Stop: **1 bit**
5. Flow control: **None**

If you are not familiar with serial communications, follow the instructions below for the program you intend to use.

### 2.4 Tera Term instructions

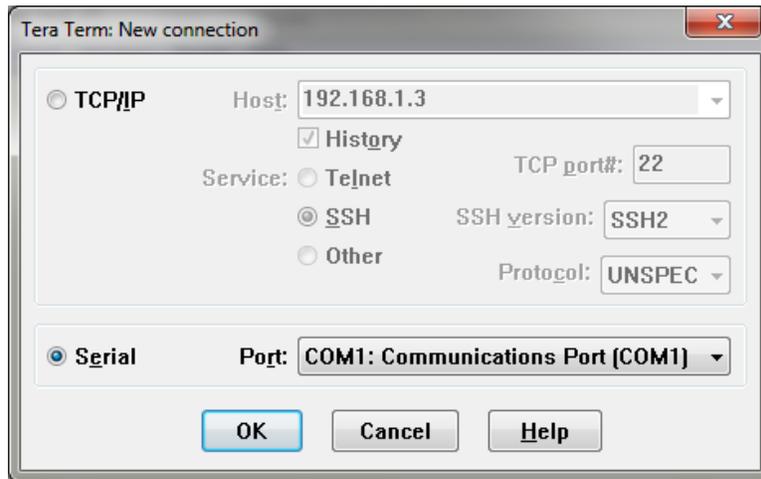
Tera Term is a terminal emulation program that can be downloaded for free as a [ZIP file](#)<sup>1</sup> from Logic PD's website. All Tera Term settings are controlled by an .ini file that you can modify as needed to make your time in Tera Term as efficient as possible (for example you can preset the port settings).

1. Start the Tera Term program.

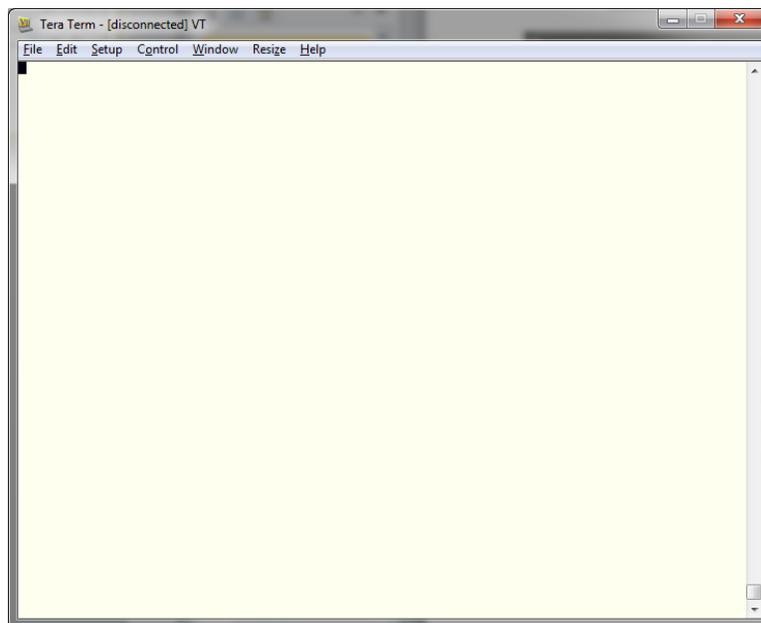
---

<sup>1</sup> <http://support.logicpd.com/downloads/240/>

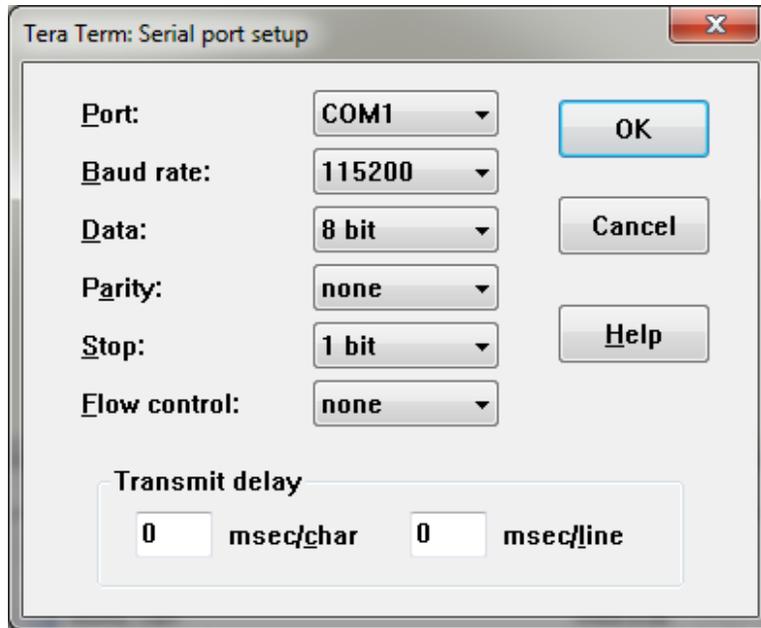
2. In the *Tera Term: New connection* window, make sure the “Serial” option is selected and choose the appropriate COM port to which your development kit is connected. Click OK.



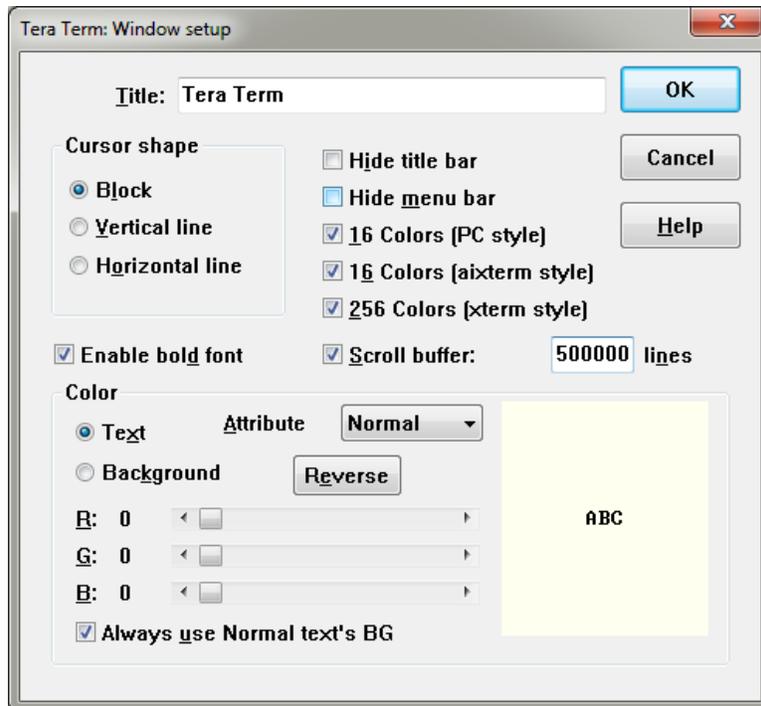
A window similar to the one included below should open.



3. From the menu toolbar, select Setup > Serial port.
4. Select the appropriate COM port for your workstation.
5. Change the port settings to:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**



6. Click OK.
7. Select Setup > Window.
8. Ensure the "Scroll buffer" box is selected and make the amount as big as you like, up to 500000.



9. Click OK.

## 2.5 Connect the Kit to Your Host PC

1. Use the gray serial cable included in the development kit to connect the serial port on the kit to the appropriate port on your host PC.
2. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below.

A screenshot of a terminal window titled "COM5:115200baud - Tera Term VT". The window has a menu bar with "File", "Edit", "Setup", "Control", "Window", "Resize", and "Help". The terminal content is as follows:

```
*****  
                        LogicLoader  
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.  
LoLo Version:      2.5.1  
SOM Model Number:  SOMDM3730-10-2782IFCR-A  
SOM Part Number:   1017318  
SOM Serial Number: 0612M02500  
*****  
losh> █
```

**NOTE:** If you do not see the LogicLoader welcome prompt, please check to make sure the cable is connected correctly. If you still do not see the LogicLoader welcome prompt, the LogicLoader files may not be loaded properly in NAND Flash. Please refer to Section 5 for instructions on how to update and load LogicLoader.

### 3. Lab 2: Learning the LogicLoader Help System

#### 3.1 Lab Goals

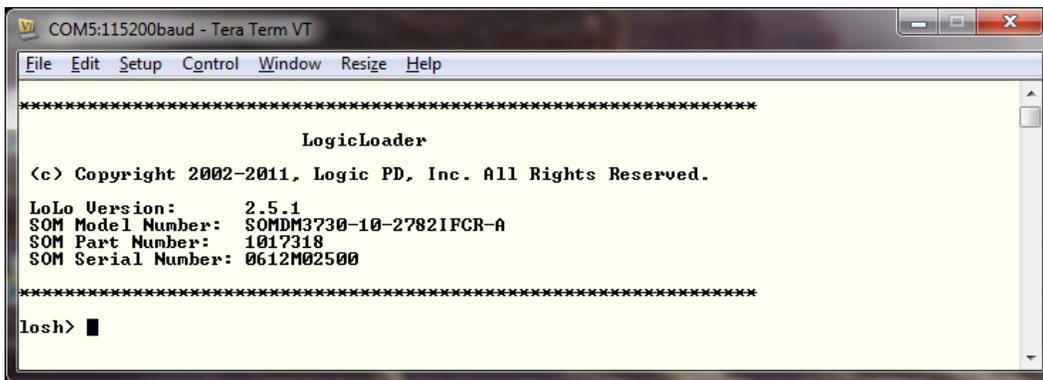
This lab will teach you the basics of working with the LogicLoader built-in help system.

#### 3.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- A serial port communications connection between the development kit and host PC

#### 3.3 LogicLoader Interface

1. Start your terminal emulation program (e.g., Tera Term).
2. Ensure the null-modem serial cable provided with the development kit is connected to both the development kit and the host PC.
3. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below.

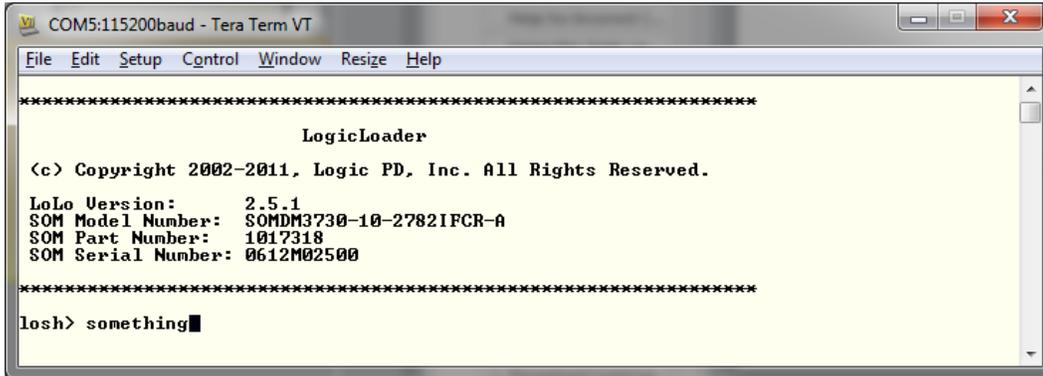


Note the string `LoLo Version: 2.5.1` in the image above. This tells us that the development kit is running version 2.5.1 of the LogicLoader software. This version of LogicLoader also provides the SOM model number, part number and serial number. This allows you to send one screenshot to Technical Support and have all of the information in one place.

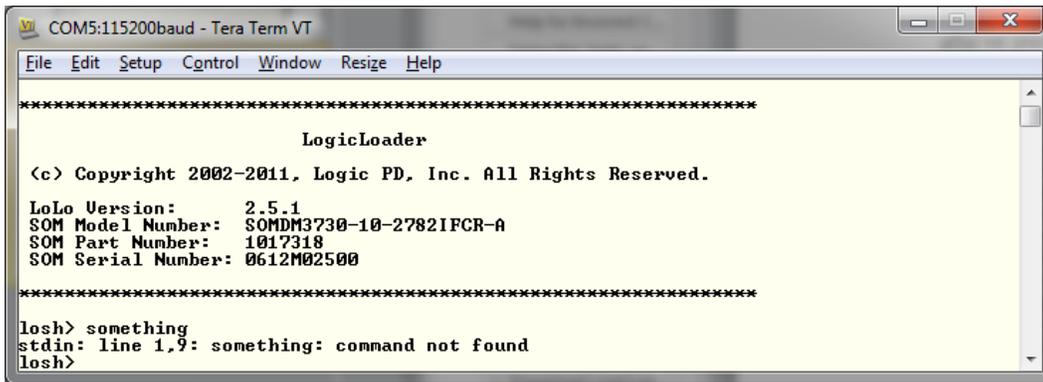
The string `losh>` is the LogicLoader command shell waiting for you to type in a command.

### 3.4 Help for Incorrect Commands

Type the word **something** at the `losh>` prompt and press **Enter**.



You should receive output similar to that included below.



That was a disappointment. Of course, this is because there is no actual command named *something* in the LogicLoader program. However, let's take a quick look at what was printed out. After we pressed Enter, LogicLoader displayed the following string:

```
stdin: line 1,9: something: command not found
```

There are actually four pieces of information contained in this string, each separated by a colon (:).

The first piece of information is `stdin`. This is an abbreviation for the words *standard input*. Because LogicLoader is a program that you interact with, it needs a way to receive information from you so it can accomplish the tasks you want it to do. The manner in which a program receives input from users has traditionally been called standard input or *stdin*. On a desktop PC, you are probably accustomed to using a keyboard and a mouse to interact with programs. The operating system (e.g., Microsoft Windows) collects the characters you type and information about where you moved and clicked your mouse and passes them on to the program you are using. With LogicLoader, there isn't a mouse or a keyboard. Instead, LogicLoader collects the characters you send over your serial port via the terminal emulation program and treats it as standard input.

With this in mind, let's look again at the string of information returned by LogicLoader after we entered the *something* command:

```
stdin: line 1,9: something: command not found
```

The first string is telling you that the error encountered by LogicLoader occurred when it tried to receive something from the standard input, which is the development kit's serial port.

**NOTE:** As a quick aside, standard input has a partner in crime named *standard output* or *stdout*. On your host PC, standard output is usually the display or monitor attached to the PC. On a DM3730 Development Kit, standard output is also the kit's serial port. So when LogicLoader has anything to say to you, it talks to you on its standard output. We will come back to the concept of standard output or *stdout* a bit later because it can be useful to tell LogicLoader to send its information someplace other than *stdout*.

The next piece of information in the string is *line 1,9*. This is LogicLoader telling you that the error it encountered occurred on the first line of data received on *stdin* between the first and ninth characters. If we look back at what we typed, we'll notice that the word *something* has nine letters; therefore, we can assume that LogicLoader had a problem with that particular word.

The next piece of information is the actual word *something*. This is LogicLoader printing out the input that caused the problem.

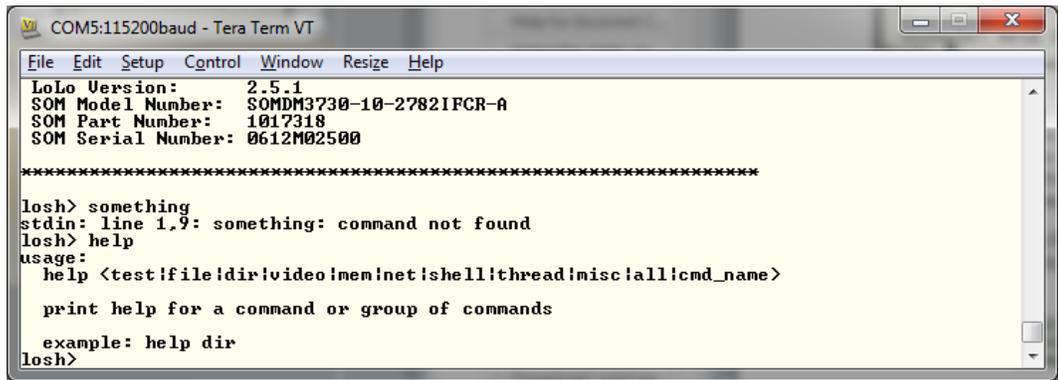
The final thing that the LogicLoader tells us is *command not found*. This informs us that LogicLoader couldn't find a command named *something* to execute.

In general, if you make a mistake, LogicLoader will try to tell you what you did wrong. Don't worry about messing up or trying to type every command perfectly the first time. LogicLoader is a stable and forgiving program and you won't hurt your board by playing around with it.

Now that we have established that there isn't a command named *something*, let's find out what commands are included in LogicLoader.

### 3.5 Using the *help* Command

1. Type `help` at the `losh>` prompt and press **Enter**.



Was that helpful? Maybe or maybe not, depending on how you expect a help system to work. Let's take a look at LogicLoader's help system. The first thing you should know is that LogicLoader interprets the word *help* as a command. The *help* command actually needs to be typed in conjunction with another word (its argument) to be a valid LogicLoader command.

So, by only typing *help*, we typed an invalid LogicLoader command. We haven't done much right yet, have we?

It's okay that we made a mistake though, because LogicLoader recognized it and is trying to teach us how to use the *help* command. Let's look at what LogicLoader printed:

```
usage:
  help <test|file|dir|video|mem|net|shell|thread|misc|all|cmd_name>
  print help for a command or group of commands
  example: help dir
```

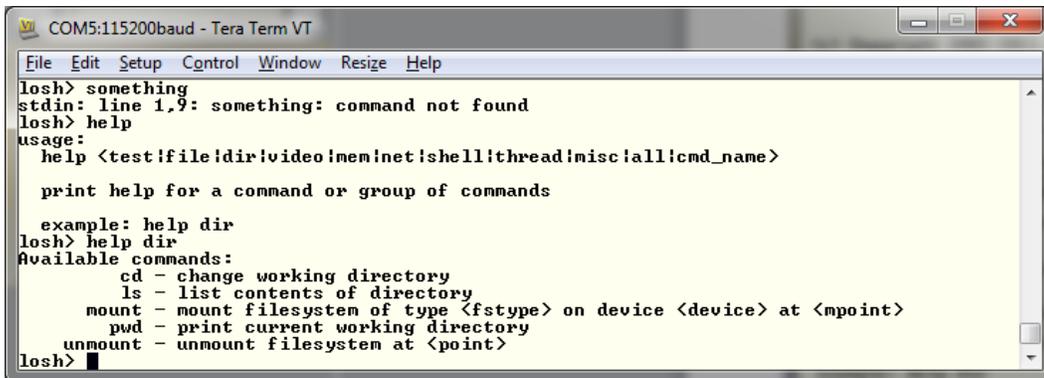
The output above is known as a command's help topic. If you type a valid LogicLoader command, but don't use it in a valid manner, LogicLoader will print out the command's help topic to remind you of the command's proper format and use.

The first string of output given by LogicLoader, included again below, is the command's usage string.

```
usage:
  help <test|file|dir|video|mem|net|shell|thread|misc|all|cmd_name>
```

It tells us how the command should be typed. Some commands need to be supplied with arguments to work correctly. If this is the case, the necessary arguments will be printed between angle bracket symbols (< >). If more than one option exists for an argument, the possible arguments will be listed and separated by the vertical bar symbol (|). The above usage string is telling us that the *help* command must be given an argument and that argument can be the word *test*, or *file*, or *dir*, etc. The rest of the output simply tries to explain what a command does and provide an example of how to use the command. Let's follow LogicLoader's example and see what happens.

2. Type `help dir` at the `losh>` prompt and press **Enter**.



```
COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> something
stdin: line 1,9: something: command not found
losh> help
losh> help
usage:
  help <test|file|dir|video|mem|net|shell|thread|misc|all|cmd_name>
  print help for a command or group of commands
  example: help dir
losh> help dir
Available commands:
  cd - change working directory
  ls - list contents of directory
  mount - mount filesystem of type <fstype> on device <device> at <mpoint>
  pwd - print current working directory
  unmount - unmount filesystem at <point>
losh>
```

Now the *help* command is working for us. It printed a list of LogicLoader commands relating to directories, along with a short summary of what each command does.

3. Type `help ls` at the `losh>` prompt and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
example: help dir
losh> help dir
Available commands:
    cd - change working directory
    ls - list contents of directory
    mount - mount filesystem of type <fstype> on device <device> at <mpoint>
    pwd - print current working directory
    unmount - unmount filesystem at <point>
losh> help ls
usage:
ls [dir]

list contents of directory

example: ls
         ls /dev
losh>

```

Again, by using `ls` as the `help` command's argument, we've asked the `help` command to print the help topic for a specific LogicLoader command.

We've just learned three important things about LogicLoader's `help` command:

- First, if you have completely forgotten how to use LogicLoader, just type `help` at the `losh>` prompt. This will print the help topic for the actual `help` command.
- Second, the `help` command can print a summary group of commands relating to a specific topic. If you use `help dir`, you will get a brief listing of all LogicLoader commands relating to directories. If you use `help file`, you will get a listing of LogicLoader commands used when working with files.
- Third, the `help` command will display the help topic for a single command. If you know a command's name but forgot how to use it, just use the `help` command followed by the name of the command and you will see the help topic for that command.

Before we continue, let's take a closer look at the `ls` command's help topic. It looks a little different than what we saw before. Notice the command's usage string, reprinted below.

```
usage:
    ls [dir]
```

There aren't any angle bracket symbols (`<` `>`) on that line, but there is a pair of square brackets (`[]`) with the word `dir` in between them. Arguments that *must* be supplied with a given command are listed in angle brackets. Arguments that *may optionally* be used with a command are listed in square brackets. So the `ls` command's usage string tells you that you can either type the command alone or optionally type the command with the name of the directory whose contents you want to list.

Try both of the examples as listed by the `ls` command's help topic.

4. Type `ls` at the `losh>` prompt and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
  mount - mount filesystem of type <fstype> on device <device> at <mpoint>
  pwd - print current working directory
  umount - unmount filesystem at <point>
losh> help ls
usage:
ls [dir]

list contents of directory

example: ls
         ls /dev
losh> ls
D :                               tftp      0
D :                               lboot    0
D :                               etc       0
D :                               dev       0
losh>
    
```

The first invocation of the `ls` command simply listed the contents of the current directory.

5. Type `ls /dev` at the `losh>` prompt and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> ls /dev
S :                               id0      0
S :                               pm0      0
S :                               i2c0     0
S :                               i2c1     0
S :                               ata0d     0
S :                               ata0c     0
S :                               ata0b     0
S :                               ata0a     0
S :                               ata0      0
S :                               flash0d   0
S :                               flash0c   0
S :                               flash0b   0
S :                               flash0a   0
S :                               flash0    8388608
S :                               null      0
S :                               uart0     0
S :                               sdmmc0d   0
S :                               sdmmc0c   0
S :                               sdmmc0b   0
S :                               sdmmc0a   0
S :                               sdmmc0    0
S :                               nand0d    0
S :                               nand0c    0
S :                               nand0b    0
S :                               nand0a    0
S :                               nand0     0
losh>
    
```

The second invocation of the `ls` command listed the contents of the `/dev` directory because we supplied the command with an optional argument.

6. Before we leave the topic of LogicLoader's help facility completely, let's take a look at all of the available commands. Type `help all` at the `losh>` prompt and press **Enter**.

```

losh> help all
Available commands:
  bitmap - draws a bitmap on the screen
  bootme - starts a 'BOOTME' transfer with Platform Builder
  burn - burns the loaded image into flash device [device]
  cache-off - stop the processor's cache
  cache-on - start the processor's cache
  cat - print contents of a file to stdout
  cd - change working directory
  cp - copies file <src> to new file <dest>
  cpu-freq - Sets the core CPU frequency, and core voltage level
  date - display the number of seconds since boot
  dd - Performs a raw data dump of a device to a file, or writes a raw
      data file to a device
  draw-test - draws framed red, green, blue and stipple test patterns
  echo - echo a string to stdout or write to [file]
  erase - erase a section of non-volatile memory
  exec - disable cache & ints, then jump to a loaded OS, or to [addr]
  help - print help for a command or group of commands
  hd - print a file to stdout in hex format
  ifconfig - configure a network interface
  ifmac - program/display the MAC address for a network interface
  info - print information from chosen category
  jump - jump to a loaded image, or to [addr]
  kill - stop thread <thread_id> from executing
  load - download a binary image
  load-buf - sets or displays the load buffer address
  load-var - load shell variables from file
  ls - list contents of directory
  make-var - make-var is used to create a shell variable
  md5sum - calculate an MD5 hash on a file or memory chunk
  mem-cmp - compare memory from <a> to <b> for <len>
  mem-copy - copy memory <src> to <dst> for <count> in sizes of [bhw]
  mem-fill - fill memory at <addr> for <count> with <value> in sizes of [bhw]
  mount - mount filesystem of type <fstype> on device <device> at <mpoint>
  part-add - create partition <a-d> on <device> from <start_blk> with <length>
  part-rem - Removes the partition <entry> <device>
  pin - redefine, or inhibit (dnu- do not use) a CPU pin
  ping - ping remote host at <ip-address> [reps] times
  ps - display the list of currently executing threads
  pwd - print current working directory
  reset - resets the processor
  save-var - save shell variables to a file
  set - print shell variables or set shell options
  rm - removes a file or files
  sleep - sleep for <ms> number of milliseconds
  slide-show - displays a slide show of bitmaps on the screen
  source - execute a series of losh commands from <file>
  tsleep - thread sleep for <ms> number of milliseconds
  unmount - unmount filesystem at <point>
  unset - destroys a variable
  update - load and install an update image for this Logic product
  video-add - create a new video display configuration based on current settings
  video-clear - clear the default video screen
  video-close - turns off and disconnects the default video device
  video-fb - sets or displays the video frame buffer address
  video-init - connect default video device but do not enable the controller
  video-off - turns off an initialized display
  video-on - turns on an initialized display
  video-open - open the default video device
  w - write memory [of specified width, w is default] at addr
  x - examine memory with [width][format] at an addr for a [len]

losh>

```

Supplying the `help` command with the `all` argument lists the name of each command built into the version of LogicLoader running on your development kit, along with a brief description of what the command does. For additional information, see Logic PD's [LogicLoader v2.5 User Guide](http://support.logicpd.com/downloads/1428/)<sup>2</sup> and [LogicLoader v2.5 Command Description Manual](http://support.logicpd.com/downloads/1440/).<sup>3</sup> A SOM-specific addendum to the user guide is also available on the [downloads page](http://support.logicpd.com/auth/)<sup>4</sup> for your development kit under the *LogicLoader Bootloader/Monitor* heading. You should download and review these documents now.

<sup>2</sup> <http://support.logicpd.com/downloads/1428/>

<sup>3</sup> <http://support.logicpd.com/downloads/1440/>

<sup>4</sup> <http://support.logicpd.com/auth/>

## 4. Lab 3: Using the LogicLoader Command Line Editor

### 4.1 Lab Goals

This lab will teach you how to work more efficiently with LogicLoader by learning its command line editing features.

### 4.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- A serial port communications connection between the development kit and host PC

### 4.3 Instructions

1. At the `losh>` prompt, use the `help` command.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
*****
                        LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> help
usage:
  help <test|file|dir|video|mem|net|shell|thread|misc|all|cmd_name>
  print help for a command or group of commands
example: help dir
losh>
    
```

As discussed in Section 3, the `help` command requires an argument. Therefore, just typing `help` prints out the commands usage string to remind us how the command should look.

Now that we've been reminded that the `help` command needs an argument, let's pretend that we wanted to use the `help video` command because we wanted to see all of the commands that relate to a display. We could just type `help video` at the prompt and press Enter. But, there is an easier way.

- At the `losh>` prompt, press the up-arrow key on your keyboard.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
*****
                        LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> help
usage:
  help <test!file!dir!video!mem!net!shell!thread!misc!all!cmd_name>
  print help for a command or group of commands
  example: help dir
losh> help
    
```

Notice how the `help` command reappears. The up-arrow cycles through the last fifteen commands that you have typed. Type a few more commands and then use the up-arrow and Enter key to find and repeat them.

The down-arrow works in the opposite way of the up-arrow. Using the arrows can save you quite a bit of typing if you need to repeat a command often. But what if you typed the command wrong to begin with?

- At the `losh>` prompt, type `help video-open` and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> help video-open
usage:
  video-open <display # or name> <bpp>
  open the default video device
  example: video-open 5 16
           video-open MyScreen 16
  See 'info video [name:number]' for a list of supported displays
  Supported depths: 8, 16, 24
losh> █
    
```

The `video-open` command is used to initialize LogicLoader's display. We can see that a display depth of 8, 16, or 24 bits per pixel is supported.

Display (or color) depth refers to the number of bits per pixel in which color is represented on a display. This information is specific to each display. For more information on Logic PD displays, please refer to the specific display's quick start guide and any associated application notes. For any other display, please refer to the appropriate documentation accompanying the display.

4. At the `losh>` prompt type `info video` and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> info video
Supported Video Devices
Number  Type      Name
-----  -
002     Pre Defined LQ121S1DG31
003     Pre Defined LQ036Q1DA01
005     Pre Defined LQ64D343
007     Pre Defined LQ10D368
015     Pre Defined LK043T1DG01
016     Pre Defined LW700AT8309
040     Pre Defined DUI_720p
041     Pre Defined DUI_SUGA
042     Pre Defined DUI_SXGA
043     Pre Defined DUI_UXGA
044     Pre Defined DUI_UGA
045     Pre Defined DUI_UXGAplus
046     Pre Defined DUI_UXGA
047     Pre Defined DUI_XGA
losh>
    
```

We won't go into much detail concerning the specifics of that command during this lab. However, by reading the command's help topic we can see that it takes two mandatory arguments: a display number and depth. By reading the output further, we can see that the version used during the writing of this lab accepted display numbers 2, 3, 5, 7, 15, 16, 40, 41, 42, 43, 44, 45, 46, and 47.

5. At the `losh>` prompt type `video-open 0 16` and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
Number  Type      Name
-----  -
002     Pre Defined LQ121S1DG31
003     Pre Defined LQ036Q1DA01
005     Pre Defined LQ64D343
007     Pre Defined LQ10D368
015     Pre Defined LK043T1DG01
016     Pre Defined LW700AT8309
040     Pre Defined DUI_720p
041     Pre Defined DUI_SUGA
042     Pre Defined DUI_SXGA
043     Pre Defined DUI_UXGA
044     Pre Defined DUI_UGA
045     Pre Defined DUI_UXGAplus
046     Pre Defined DUI_UXGA
047     Pre Defined DUI_XGA
losh> video-open 0 16
error: video-open: found no suitable display driver
error: video-open: found no suitable display driver
losh>
    
```

Notice that we typed the command wrong and LogicLoader prompted us. The error was a result of trying to initialize screen number zero, which our help topic told us was not supported.

Instead of retyping the entire command line, let's just fix the command we sent to LogicLoader. To do so:

- a. Press the up-arrow to bring back the command previously entered.
- b. Use the left-arrow to bring the cursor to the right of the `0` character.
- c. Press **Backspace** to delete the errant zero.
- d. Type the number of a supported display (such as 15) in place of the zero.
- e. Press **Enter**.

**NOTE:** The LK043T1DG01 4.3" LCD (number 015) is the default display that is shipped with the DM3730 Development Kits.

The new command should look like this: `video-open 15 16`

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
002 Pre Defined LQ121S1DG31
003 Pre Defined LQ036Q1DA01
005 Pre Defined LQ64D343
007 Pre Defined LQ10D368
015 Pre Defined LK043T1DG01
016 Pre Defined LW700A18309
040 Pre Defined DUI_720p
041 Pre Defined DUI_SUGA
042 Pre Defined DUI_SXGA
043 Pre Defined DUI_UXGA
044 Pre Defined DUI_UGA
045 Pre Defined DUI_UXGAplus
046 Pre Defined DUI_UXGA
047 Pre Defined DUI_XGA

losh> video-open 0 16
error: video-open: found no suitable display driver
error: video-open: found no suitable display driver
losh> video-open 15 16
video-open display: width: 480 height: 272 bpp: 16 disp: 15
losh>

```

Notice how the command is correct now? Editing the command line instead of retyping the entire line cut our key presses in half.

If the `video-open` command worked correctly, the screen should now be on. If the screen does not display something, please make sure the display is connected properly and the correct display and color depth have been selected.

For those of you familiar with traditional terminal emulators or for those using a program that doesn't properly send Backspace and arrow key presses, see the series of key combinations included below that provide another option to help move you around a command line.

- Ctrl+A** - Move to the beginning of the line
- Ctrl+E** - Move to the end of the line
- Ctrl+B** - Move backward (to the left) one character
- Ctrl+F** - Move forward (to the right) one character
- Ctrl+D** - Delete the character under the cursor

Go ahead and take some time to familiarize yourself with the above key combinations and command line editing.

## 5. Lab 4: Updating LogicLoader and Recovery

### 5.1 Lab Goals

This lab will teach you how to download a new version of LogicLoader to your DM3730 Development Kit using the *update* command. In case the development kit's memory gets corrupted, please follow these instructions for the recovery procedure as well.

### 5.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- An SD card
- A terminal emulation program (e.g., Tera Term) with the following port settings:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**

### 5.3 Connect the Kit to Your Host PC

1. Use the gray serial cable included in the development kit to connect the serial port on the kit to the appropriate port on your host PC.
2. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below. Take note of what version number appears in this welcome prompt, as that piece of information will be required later on.



```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
*****
                        LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****
losh> █
  
```

**NOTE:** If you do not see the LogicLoader welcome prompt, please check to make sure the cable is connected correctly. If you still do not see the LogicLoader welcome prompt, the LogicLoader files may not be loaded properly in NAND Flash. Please follow the procedure in this lab to update the software.

### 5.4 Download a New Version of LogicLoader

1. Log into Logic PD's [downloads page](http://support.logicpd.com/auth/)<sup>5</sup> using your username and password.

<sup>5</sup> <http://support.logicpd.com/auth/>

2. Click on the **All Downloads** link under the *DM3730 Development Kit* heading for your kit.
3. Scroll down until you find the section header *LogicLoader Bootloader/Monitor*.
4. Click on the + icon next to the **LogicLoader for DM3730 / AM3703 SOMs** link; this expands the list of all available LogicLoader versions. If the current version is newer than what is on your DM3730 SOM, you should update.
5. Click on the version number marked (*current*) and save the file to your host PC. This ZIP file contains all the files necessary to install and use LogicLoader. **NOTE:** Most of the file names will begin with a part number; the instructions below will omit specific part numbers for brevity.

A brief description of the included file formats and files is listed below.

#### File format descriptions:

- .elf files - These are used by the LogicLoader *load elf* command. Use these files if your SOM already has a working version of LogicLoader running on it.
- .upd files - These are used by the LogicLoader *update* command. Again, use these files if your SOM already has a working version of LogicLoader running on it.
- .raw files - These are stripped, binary versions of the software that may be used by some programmers. LogicLoader can load files of this type.

#### File descriptions:

- *boot* - These files are destined for internal SRAM. They are used with a JTAG emulator to restore LogicLoader to your module. They are loaded into internal SRAM and are run to initialize DRAM.
- *lolo\_RAM*. - These files are destined for DRAM. They are used with a JTAG emulator to restore LogicLoader to your module. They are loaded into DRAM after it has been initialized with the *boot* file. Once run, they produce a `l0sh>` prompt from which you can restore LogicLoader to flash on your module.
- *lolo\_NAND* - These files are destined for flash. They are used to update your SOM to the latest version of LogicLoader from the `l0sh>` prompt.
- *lolo\_SD* - These files are destined for an SD Card. They are used to update your SOM to the latest version of LogicLoader from the `l0sh>` prompt.
- *MLO* – This file is a NoLo image with a header appended to the beginning of the file. This header is required by the DM3730 boot ROM for bootable files. The *MLO* file is intended to be placed on an SD card for booting from SD.
- *lboot.elf* - This file is exactly the same image as the *lolo\_SD.elf* file. The reason it is included again with this name is for convenience. *lboot.elf* is intended to be placed on an SD card for booting from SD. Both *MLO* and *lboot.elf* must be on the SD card to boot into LogicLoader from SD.
- *lboot.sup* - This file is used at boot time to configure how LogicLoader operates. This file is optional and can be read/modified with a text editor. To use this file, it must be placed in the boot location (i.e., the same directory as the *lboot.elf* file).
- *lboot.var* - This file is used at boot time to load variables into the shell. This file is optional and can be read/modified with a text editor. To use this file, it must be placed in the boot location (i.e., the same directory as the *lboot.elf* file).
- *lboot.lol* - This file is used at boot time to perform one or more actions that would otherwise be typed into the shell. This file is optional and can be read/modified with a text

editor. To use this file, it must be placed in the boot location (i.e., the same directory as the *lboot.elf* file).

The LogicLoader ZIP file also contains files with “nolo” in the title that are not mentioned above. NoLo is a pre-loader for LogicLoader and is covered in more detail in the *LogicLoader v2.5 User Guide*.

Now that you have downloaded the latest version of LogicLoader to your computer, you can update LogicLoader on your development kit using a bootable SD card.

## 5.5 Update LogicLoader using a Bootable SD Card

### 5.5.1 Prepare the SD Card

1. The first step is to prepare the SD card to boot. A requirement of the processor is that the SD card must be formatted without an extended partition to boot properly. The built-in Windows formatting tool is not always able to do this. Logic PD has found that the HP USB Formatting Tool version 2.0.6 appears to be the best tool for formatting an SD card so it can be used to boot a processor based on OMAP3 technologies (e.g., DM3730, AM3703).

The most recent version of the tool available on HP's site doesn't seem to work as well as version 2.0.6. As such, the tool needs to be downloaded from any number of third-party websites. To find these websites, perform a Google search for: **HP Formatting Tool v2.0.6**. **NOTE:** As with downloading anything from the Internet, use caution in what sites you visit and what you download. Logic PD has no association with any of these websites and makes no guarantees to their validity or the usability of downloads from their sites.

2. After the HP USB Formatting Tool version 2.0.6 has been installed on your host PC, insert an SD card into your USB card reader, connect the card reader to your host PC, and open the formatting tool. **IMPORTANT NOTE:** Remember that formatting an SD card will erase its entire contents; back-up the SD card's contents before continuing or select a card that doesn't contain vital information.
3. Select the SD card device “FAT32” from the *File System* drop-down menu. Click “Start”.
4. You will receive a warning that the contents of the card will be erased. Assuming you have backed up the contents of the SD card, click “Yes”.
5. When formatting completes, you will receive a confirmation screen with the new file system type and the space allocation on the card. You are now ready to add the LogicLoader files to the SD card and boot from SD.

### 5.5.2 Use the SD Card to Update LogicLoader

1. Locate the LogicLoader files that you downloaded to your computer in Section 5.4.
2. Copy the *MLO* file from that folder to the SD card. **IMPORTANT NOTE: This file must be the first file added to the SD card for the development kit to boot properly.**
3. Next, copy the *lboot.elf* file to the SD card.
4. Remove the SD card from your USB card reader and insert it into the bootable SD card slot on your DM3730 Development Kit baseboard.

- Slide the power switch to the ON position; you should see a screen similar to that shown below. **HINT:** You can hold down the **v** key on the keyboard during boot-up to verify the *Boot Device* is SDMMC indicating the kit booted from the SD card.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
NoLo Version : 2.5.1
NoLo Build   : EPS-LNX-EDN-01 Wed Dec 14 16:35:20 CST 2011
NoLo Compiler: gcc version 4.3.3 (Sourcery G++ Lite 2009q1-203)
ID version   : 3
Platform    : DM/AM 37x SOM-LU
Image type  : Elf
Boot Device  : SDMMC

*****

                        LogicLoader

(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.

LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500

*****

losh>
    
```

Note that the first section of the banner says the *Image Type* is Elf and the *Boot Device* is SDMMC.

- Because NoLo and LogicLoader reside in NAND, it is necessary to erase the old LogicLoader before updating to a new one. Type `erase /dev/nand0 B0 B2048` and press **Enter**.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
NoLo Version : 2.5.1
NoLo Build   : EPS-LNX-EDN-01 Wed Dec 14 16:35:20 CST 2011
NoLo Compiler: gcc version 4.3.3 (Sourcery G++ Lite 2009q1-203)
ID version   : 3
Platform    : DM/AM 37x SOM-LU
Image type  : Elf
Boot Device  : SDMMC

*****

                        LogicLoader

(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.

LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500

*****

losh> erase /dev/nand0 B0 B2048
erasing nand: 100%
erased /dev/nand0' start=0x0: len=0x800 bytes/blocks skipped 0
losh>
    
```

Because you booted the kit from SD, you will not be warned that you are erasing locked areas in NAND. You now have a blank flash device to update to.

- Type `update` and press **Enter**. Your screen should look similar to that included below.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

NoLo Version : 2.5.1
NoLo Build   : EPS-LNX-EDN-01 Wed Dec 14 16:35:20 CST 2011
NoLo Compiler: gcc version 4.3.3 (Sourcery G++ Lite 2009q1-203)
ID version   : 3
Platform    : DM/AM 37x SOM-LU
Image type  : Elf
Boot Device : SDMMC

*****

                LogicLoader

<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.

LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-27821FCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500

*****

losh> erase /dev/nand0 B0 B2048
erasing nand: 100%
erased /dev/nand0' start=0x0: len=0x800 bytes/blocks skipped 0
losh> update
send update file:
    
```

LogicLoader is waiting for you to send the new version over the serial port.

- From the menu toolbar, select **File > Send file...** to send the new LogicLoader version to your kit.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
File Edit Setup Control Window Resize Help
New connection... Alt+N
Duplicate session Alt+D
Cygwin connection Alt+G
Log...
Comment to Log...
View Log
Show Log dialog...
Send file...
Transfer
SSH SCP...
Change directory...
Replay Log...
TTY Record
TTY Replay
Print... Alt+P
Disconnect Alt+I
Exit Alt+Q

1 Wed Dec 14 16:35:20 CST 2011
.3.3 (Sourcery G++ Lite 2009q1-203)
-LU

*****

icLoader

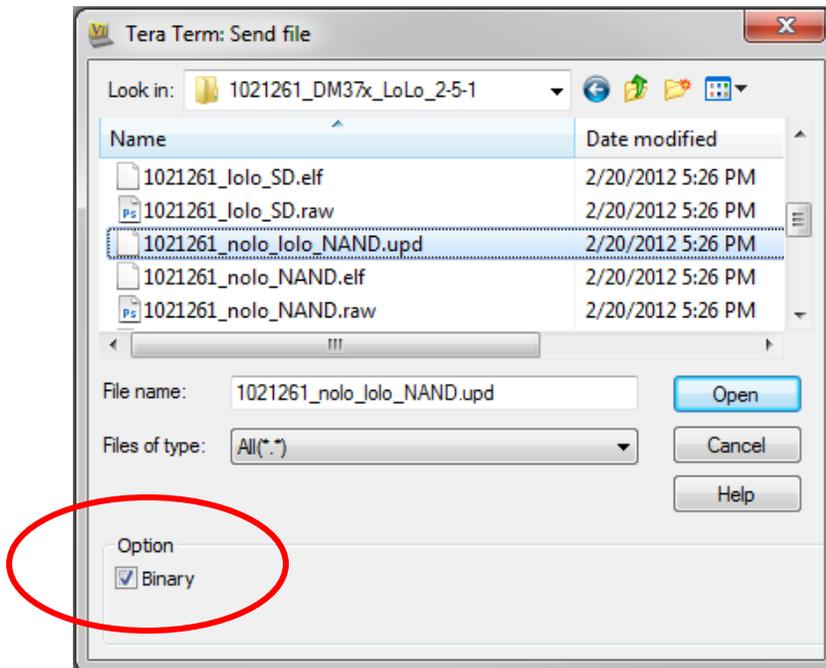
gic PD, Inc. All Rights Reserved.

0-10-27821FCR-A
00
*****
048
0: len=0x800 bytes/blocks skipped 0
    
```

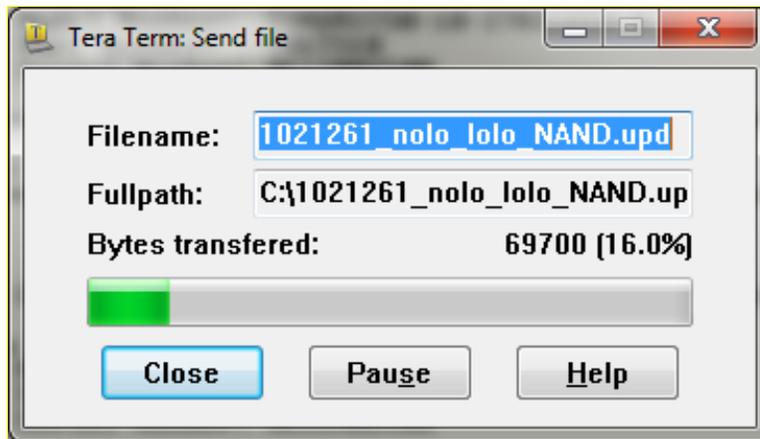
- Navigate to the location where you extracted the LogicLoader files and select the file that includes `nolo_lolo_NAND.upd`. The `.upd` files are used by the LogicLoader `update` command. This file contains both NoLo and LogicLoader components necessary for updating LogicLoader. If you do not see a file that includes `nolo_lolo_NAND.upd`, please refer to the

*Release\_Notes.txt* file included in the LogicLoader version you downloaded for special instructions.

10. Verify that the “Binary” checkbox is selected. Click Open.



11. The terminal emulation program will begin sending the file to the DM3730 SOM.



- Verify that the download was successful by waiting for an update done message to appear.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
SOM Part Number: 1017318
SOM Serial Number: 0612M02500
*****
losh>
*****
                LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****
losh> erase /dev/nand0 B0 B2048
erasing nand: 100%
erased '/dev/nand0' start=0x0: len=0x800 bytes/blocks skipped 0
losh> update
send update file:
.....
YAFFS core: yaffs: Mounting temp
YAFFS core: save exit: isCheckpointed 1
update done.
losh>
    
```

- Remove the SD card and reset your development kit.
- If the new version number is displayed, you now have the latest and greatest version of LogicLoader installed on your kit.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
NoLo Version : 2.5.1
NoLo Build   : EPS-LNX-EDN-01 Wed Dec 14 16:15:25 CST 2011
NoLo Compiler: gcc version 4.3.3 (Sourcery G++ Lite 2009q1-203)
ID version   : 3
Platform    : DM/AM 37x SOM-LV
Image type  : Elf
Boot Device : NAND
*****
                LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****
losh>
    
```

Note that if you held down the v button while resetting the kit, the first section of the banner says the *Image Type* is Elf and the *Boot Device* is NAND.

## 6. Lab 5: Establishing a Network Connection

### 6.1 Lab Goals

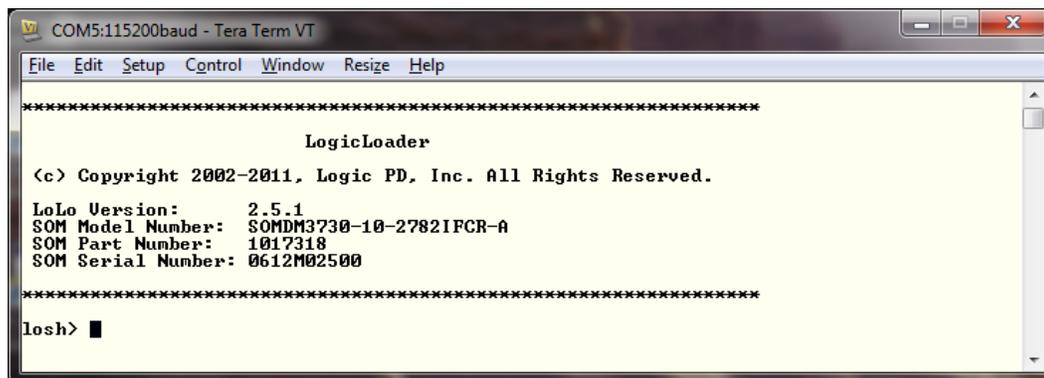
This lab will teach you how to connect your host PC to the DM3730 Development Kit via Ethernet.

### 6.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- A terminal emulation program (e.g., Tera Term) with the following port settings:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**

### 6.3 Connect the Kit to Your Host PC

1. Use the gray serial cable included in the development kit to connect the serial port on the kit to the appropriate port on your host PC.
2. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below.



```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
*****
                        LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> █
  
```

**NOTE:** If you do not see the LogicLoader welcome prompt, please check to make sure the cable is connected correctly. If you still do not see the LogicLoader welcome prompt, the LogicLoader files may not be loaded properly in NAND Flash. Please refer to Section 5 for instructions on how to update and load LogicLoader.

### 6.4 Connect the Kit to Ethernet

There are two types of addressing that can be used to connect to your development kit via Ethernet:

- Static IP addressing allows you to manually set the Ethernet addresses on your host PC and development kit.
- Dynamic IP addressing uses a Dynamic Host Configuration Protocol (DHCP) to obtain an Ethernet address. DHCP requires a DHCP server on the network, which is usually built into

modern Ethernet switches. For more information on DHCP, please see the network switch's user manual.

There are also two different ways you can connect your host PC to the development kit via Ethernet:

- A cross-over cable that connects directly between the kit and host PC via the RJ-45 connectors. This option is commonly used with static IP addressing.
- A standard Ethernet cable that requires the host PC and kit to be connected to a network switch. This option is commonly used with either dynamic or static IP addressing.

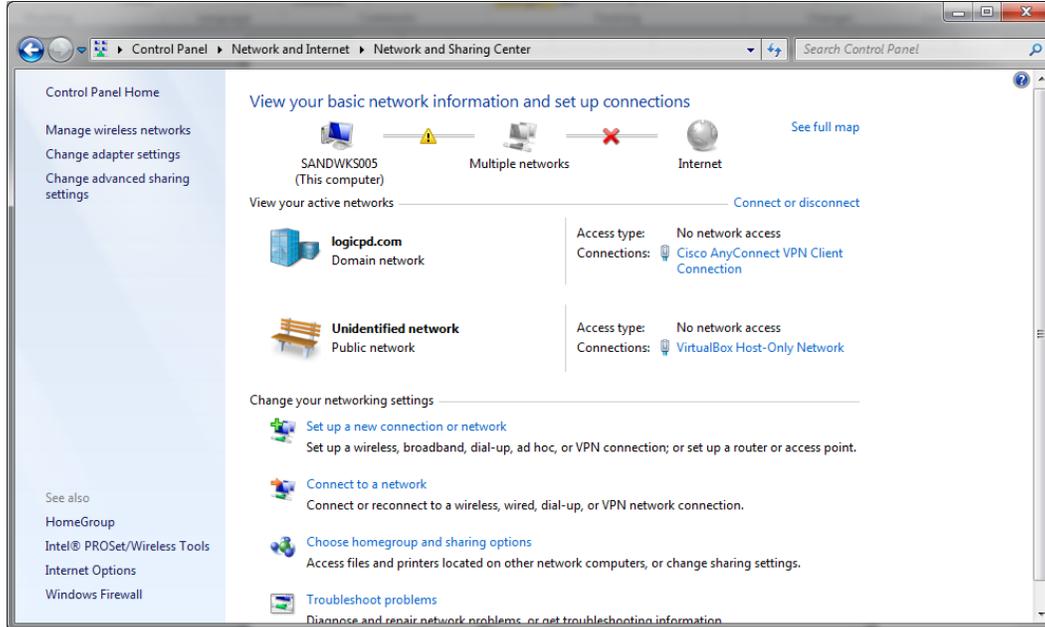
Adding Ethernet connectivity to your kit will make several procedures easier and is essential for connecting to Platform Builder for Windows CE. We will discuss both connecting directly using a cross-over Ethernet cable with static IP addressing and connecting through a network switch with dynamic IP addressing.

#### 6.4.1 Network Setup for Ethernet Cross-over Cable and Static IP Addressing

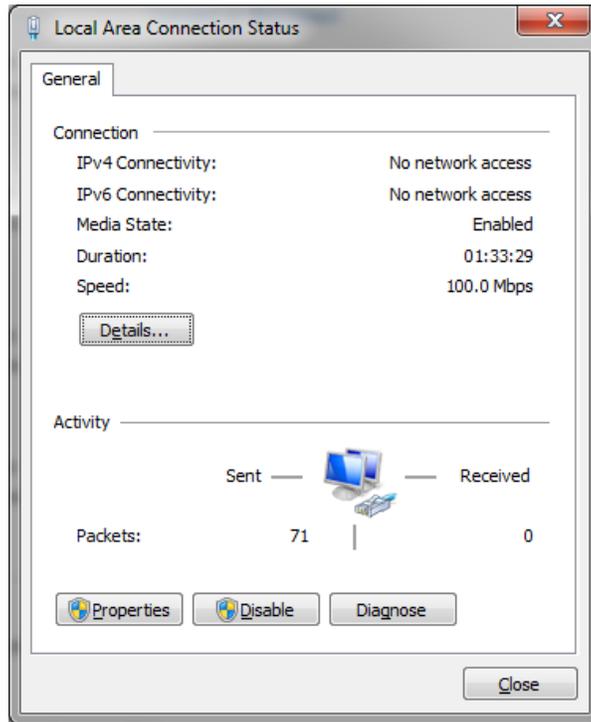
If you are using the Ethernet cross-over cable, you will need to set up the kit address using the full *ifconfig* command. If you don't know the IP address of your host PC, select Start > Run and type `cmd`. Then type `ipconfig /all` and locate the IP address for the Wired LAN connection that you are using.

The IP address of the kit must match the IP address of your host PC for as many 255 number sets as there are in the subnet mask. For example, if your subnet mask is 255.255.255.0 then the first three number sets of the IP address must be the same on both devices.

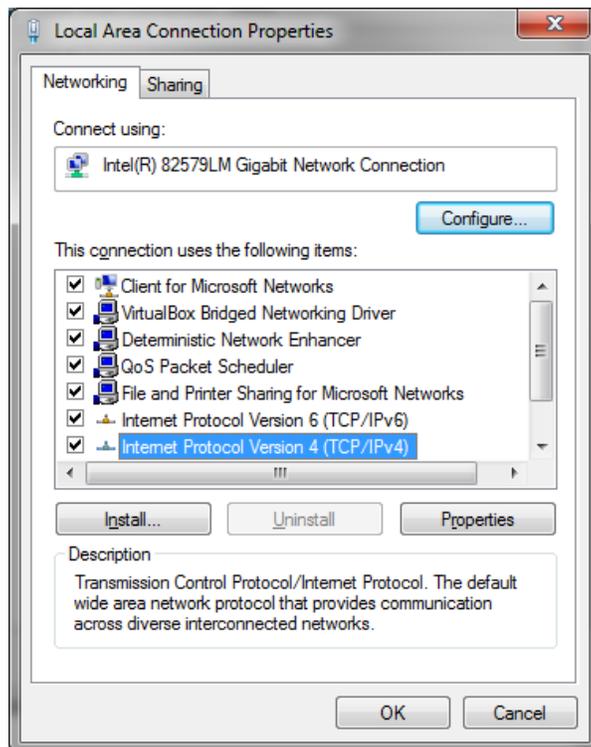
1. If you would like to set your IP address, select Start > Control Panel > Network and Sharing Center.



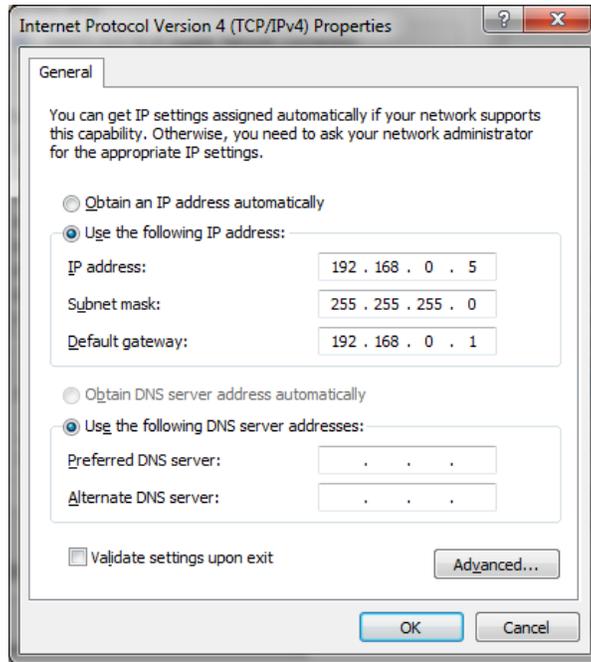
2. Double-click on the icon for your wired connection. In the *Local Area Connection Status* window, click on Properties.



3. Scroll down until you see *Internet Protocol Version 4 (TCP/IPv4)*. Click on the name to select it and then click Properties.



4. Set your IP address and subnet mask as desired. Click OK.



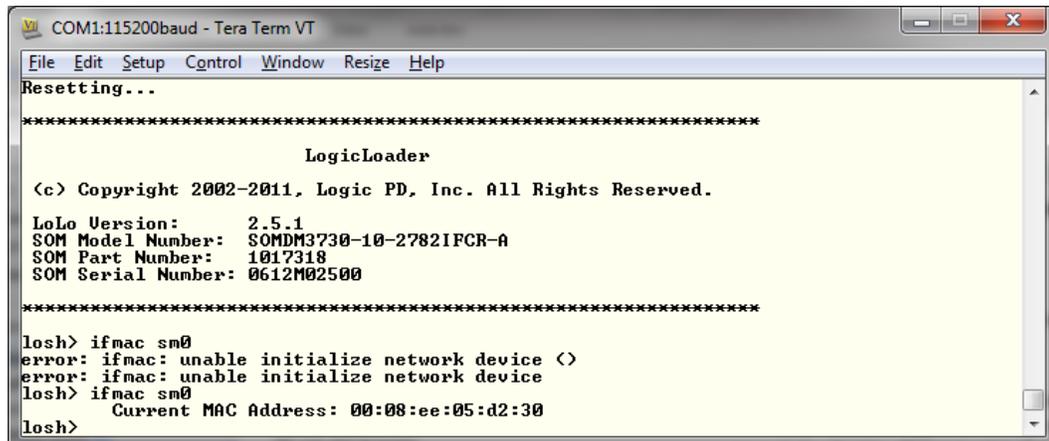
5. Once you have the IP address and subnet mask of your host PC, return to your Tera Term window. Once there, you need to set the networking parameters on the development kit.
  - a. First verify that you have a MAC address associated with the kit. In order to obtain the MAC address, use the *ifmac* command on the *sm0* (Ethernet) interface:

```
ifmac sm0
```

You may see the following error message the first time you use the *ifmac* command:

```
error: ifmac: unable initialize network device
```

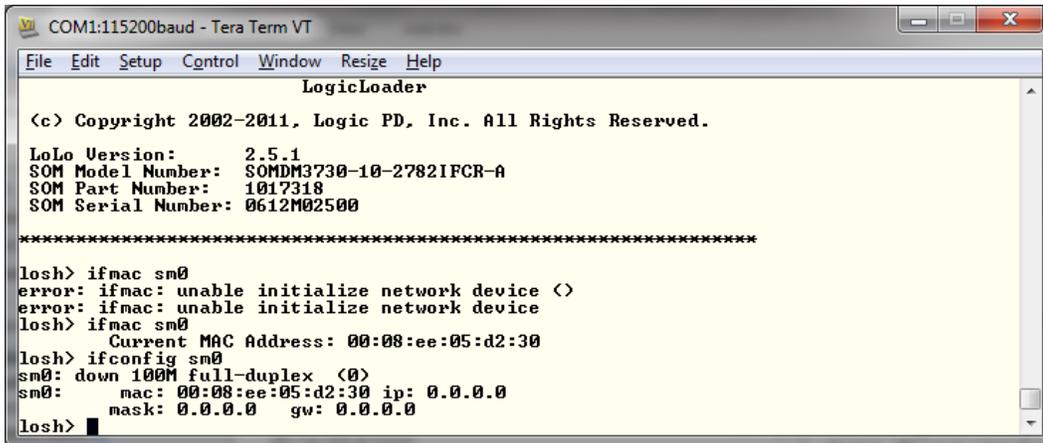
Repeat the *ifmac sm0* command and you should see the kit's MAC address.



- b. To check the IP configuration, use the command below.

```
ifconfig sm0
```

The output should be similar to that included below.



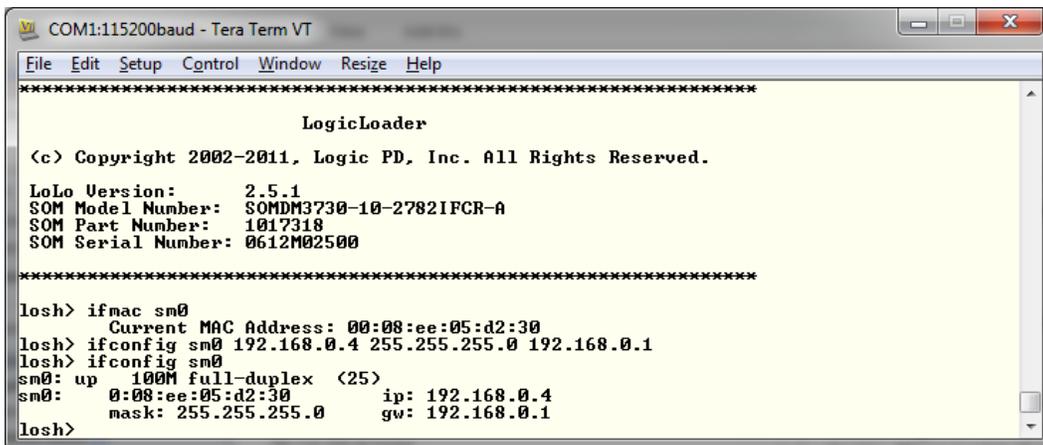
```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> ifmac sm0
error: ifmac: unable initialize network device (<)
error: ifmac: unable initialize network device
losh> ifmac sm0
Current MAC Address: 00:08:ee:05:d2:30
losh> ifconfig sm0
sm0: down 100M full-duplex (<0>)
sm0:   mac: 00:08:ee:05:d2:30 ip: 0.0.0.0
      mask: 0.0.0.0   gw: 0.0.0.0
losh>
    
```

- c. Next, use the command below to set a static IP address, substituting the names within the angle brackets with the values specific to your system.

```
ifconfig sm0 <IP address of kit> <subnet mask> <IP address of gateway >
```

Since the kit is connected via a cross-over cable, the gateway address does not really matter; however, it should be configured anyway. The figure below shows an example of the full *ifconfig* command. Remember that your numbers will be different.



```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> ifmac sm0
Current MAC Address: 00:08:ee:05:d2:30
losh> ifconfig sm0 192.168.0.4 255.255.255.0 192.168.0.1
losh> ifconfig sm0
sm0: up 100M full-duplex (<25>)
sm0:   00:08:ee:05:d2:30 ip: 192.168.0.4
      mask: 255.255.255.0 gw: 192.168.0.1
losh>
    
```

You now have set up your networking parameters and can skip to Section 6.4.3.

### 6.4.2 Network Setup for Standard Ethernet Cable and DHCP Addressing

If you have a network connection with DHCP, you can use it to quickly set your network parameters. Please make sure your host PC is also configured properly on the same subnet/LAN.

1. Connect both the development kit and host PC to the network switch using standard Ethernet cables.
2. From the `losh>` prompt, use the `ifconfig` command.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500

*****

losh> ifmac sm0
      Current MAC Address: 00:08:ee:05:d2:30
losh> ifconfig sm0 192.168.0.4 255.255.255.0 192.168.0.1
losh> ifconfig sm0
sm0: up 100M full-duplex <25>
sm0:   0:08:ee:05:d2:30   ip: 192.168.0.4
      mask: 255.255.255.0   gw: 192.168.0.1
losh> ifconfig

sm0: up 100M full-duplex <25>
sm0:   mac: 00:08:ee:05:d2:30 ip: 192.168.0.4
      mask: 255.255.255.0   gw: 192.168.0.1
losh>
    
```

In the output, you can see the interface information for `sm0`. The MAC address is `00:08:ee:05:d2:30`. The IP address, subnet mask and gateway have been set.

3. Now, use the `ifconfig sm0 dhcp` command. LogicLoader will request the IP address, subnet mask and gateway parameters from the DHCP server.
4. To display the new values, use the `ifconfig` command again, as shown below. It may take a few moments to get the DHCP information from the server.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

losh> ifmac sm0
      Current MAC Address: 00:08:ee:05:d2:30
losh> ifconfig sm0 192.168.0.4 255.255.255.0 192.168.0.1
losh> ifconfig sm0
sm0: up 100M full-duplex <25>
sm0:   0:08:ee:05:d2:30   ip: 192.168.0.4
      mask: 255.255.255.0   gw: 192.168.0.1
losh> ifconfig

sm0: up 100M full-duplex <25>
sm0:   mac: 00:08:ee:05:d2:30 ip: 192.168.0.4
      mask: 255.255.255.0   gw: 192.168.0.1
losh> ifconfig sm0 dhcp
Starting DHCP on sm0 ..
losh> ifconfig

sm0: up 100M full-duplex <25>
sm: 00:08:ee:05:d2:30   ip: 192.168.0.101
      mask: 255.255.255.0   gw: 192.168.0.1
losh>
    
```

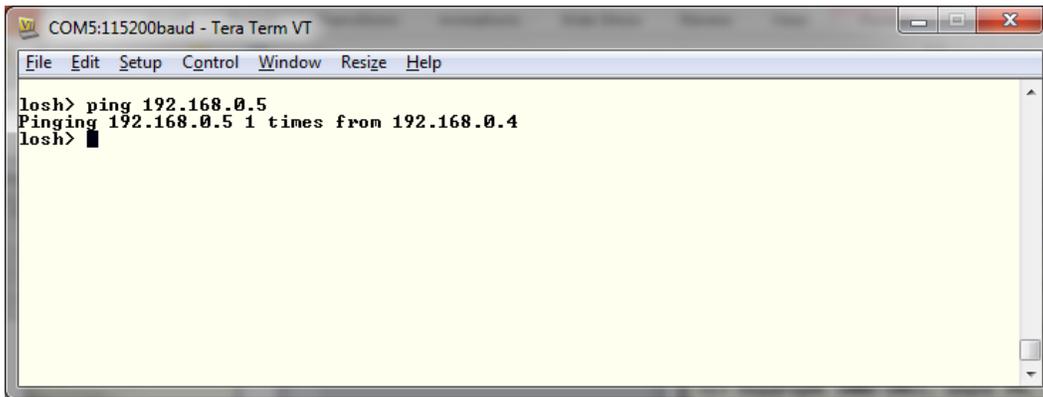
You can also use static addressing with a network switch using the instructions in Section 6.4.1; connect the kit via standard Ethernet cables to the switch. In this scenario, make sure the gateway field in the `ifconfig` command is set to the address of the switch/gateway.

### 6.4.3 Use the *ping* Command to Check the Connection

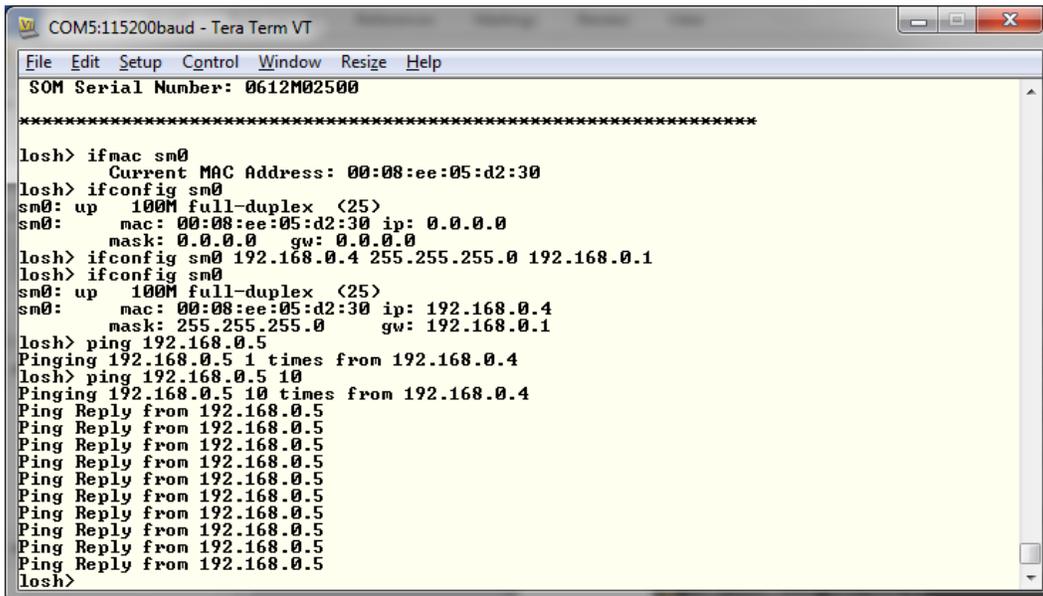
In the previous sections, you used the *ifconfig* command to initialize your Ethernet connection. Now you are going to use the *ping* command to check the connection to a specific workstation. You will need to have the development kit connected via Ethernet and know the IP address of the target host PC. If you don't know the IP address of the target, select Start > Run and type *cmd*. Type *ipconfig /all* and locate the IP address for the connection that you are using.

The *ping* command can be used with just the IP address of the target host PC or with a number of different pings specified.

To ping your host PC one time, type *ping xxx.xxx.xxx.xxx* at the *losh>* prompt, where *xxx.xxx.xxx.xxx* is the IP address of your host PC.



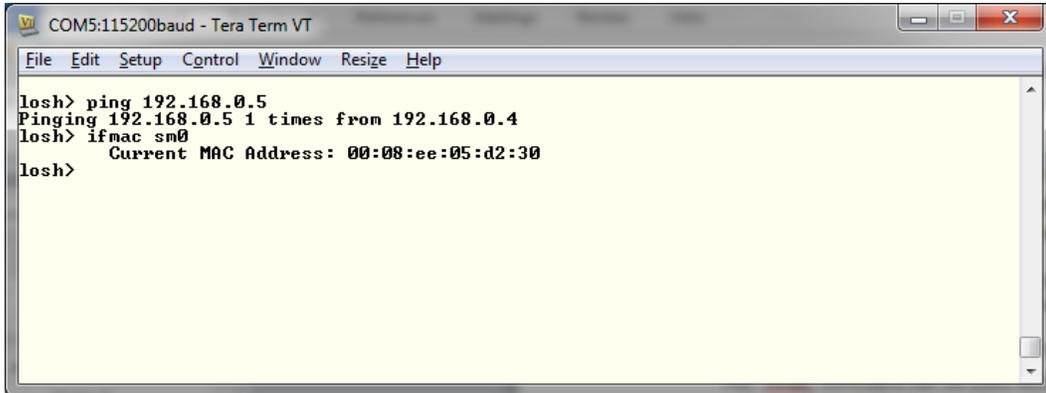
If you type *ping xxx.xxx.xxx.xxx 10* you will ping the target 10 times. Remember you can just hit the up-arrow to repeat the last command and then add the 10 at the end. An example is shown below. **NOTE:** Remember your IP address will be different than that in the example below.



You have confirmed the connection. This process can be used any time you need to set up a network connection between a kit and a host PC.

#### 6.4.4 Use the *ifmac* Command to Display and Change the MAC Address

The *ifmac* command can be used with just the interface name to display the current MAC address of your development kit. To do so, type `ifmac sm0`. It will take LogicLoader a few seconds to get the MAC address the first time you use the command. After the current MAC address is displayed, the *ifmac* command will display immediately.



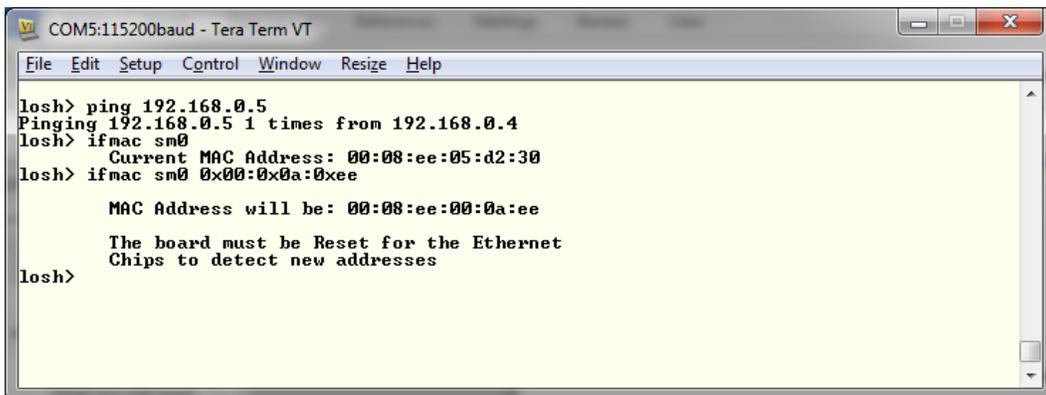
```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> ping 192.168.0.5
Pinging 192.168.0.5 1 times from 192.168.0.4
losh> ifmac sm0
Current MAC Address: 00:08:ee:05:d2:30
losh>

```

If you need to set your MAC address to something different than the current address, you can do so by adding the new address to the previous command. To change the address, you can send bytes 4, 5, and 6. The first three bytes will default to Logic PD's 0x00:0x08:0xEE.

1. Type `ifmac sm0 0x00:0x0a:0xee` at the `losh>` prompt. LogicLoader will display the following message to confirm the change has taken place.



```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> ping 192.168.0.5
Pinging 192.168.0.5 1 times from 192.168.0.4
losh> ifmac sm0
Current MAC Address: 00:08:ee:05:d2:30
losh> ifmac sm0 0x00:0x0a:0xee
MAC Address will be: 00:08:ee:00:0a:ee
The board must be Reset for the Ethernet
Chips to detect new addresses
losh>

```

2. Reset your development kit and use the `ifmac sm0` command to verify the new MAC address.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
Chips to detect new addresses
losh>
*****
                LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> ifmac sm0
        Current MAC Address: 00:08:ee:00:0a:ee
losh>
    
```

3. If you prefer to set the whole address, you can send all 6 bytes by typing `ifmac sm0 0x01:0x02:0x03:0x04:0x05:0x06`.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
                LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> ifmac sm0
        Current MAC Address: 00:08:ee:05:d2:30
losh> ifmac sm0 0x01:0x02:0x03:0x04:0x05:0x06
        MAC Address will be: 01:02:03:04:05:06
        The board must be Reset for the Ethernet
        Chips to detect new addresses
losh>
    
```

4. Reset your kit and use the `ifmac sm0` command to verify the new MAC address.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
Chips to detect new addresses
losh>
*****
                LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> ifmac sm0
        Current MAC Address: 01:02:03:04:05:06
losh>
    
```

## 7. Lab 6: Utilizing File Systems

### 7.1 Lab Goals

This lab will teach you how to utilize file systems on the DM3730 Development Kit.

### 7.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- An SD card
- A terminal emulation program (e.g., Tera Term) with the following port settings:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**

### 7.3 Fat File System (FATFS)

#### 7.3.1 Mount an SD Card

The first method of loading a file onto your development kit that we will cover in this lab is to use an SD card. This section describes how to mount an SD card in LogicLoader.

1. Store a file on an SD card.
2. Insert the SD card into the development kit's SD card slot.
3. Start a terminal emulation program on your host PC and connect the gray serial cable included in the development kit to the serial port on the kit and the appropriate port on your host PC.
4. Power on the development kit and verify the LogicLoader prompt appears in the terminal emulator window.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
*****
                LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****
losh> █
  
```

- Next, let's look at the help topic output for the `mount` command. Type `help mount`.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh>
losh>
losh>
losh> help mount
usage:
mount <fstype> <device> <mpoint> [args]
fstype can be fatfs, yaffs, emu
args can be -ro -rw or nothing then the default is ronly

mount filesystem of type <fstype> on device <device> at <mpoint>

example: mount fatfs /dev/ata0a /cf
mount fatfs /dev/sdmmc0a /sd
mount yaffs /dev/nand0a /yp
mount emu /dev/flash0a /e1
mount yaffs /e1 /yp
losh>
losh>
    
```

Remember that variables in the `<>` are required and variables in the `[]` are optional. Help usually gives the most frequently used options in the examples it provides.

- Type `mount fatfs /dev/sdmmc0a /sd`. Or, if you are working in TeraTerm, you can highlight the sample line, then right click by the `losh>` prompt and press **Enter**. The screen displayed should be similar to that included below.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh>
losh>
losh> help mount
usage:
mount <fstype> <device> <mpoint> [args]
fstype can be fatfs, yaffs, emu
args can be -ro -rw or nothing then the default is ronly

mount filesystem of type <fstype> on device <device> at <mpoint>

example: mount fatfs /dev/ata0a /cf
mount fatfs /dev/sdmmc0a /sd
mount yaffs /dev/nand0a /yp
mount emu /dev/flash0a /e1
mount yaffs /e1 /yp
losh>
losh> mount fatfs /dev/sdmmc0a /sd
losh>
    
```

- Use the `cd /sd` command to change directories to the SD card.
- Then use the `ls` command to confirm that your card has been mounted and your file is on the card.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
args can be -ro -rw or nothing then the default is ronly

mount filesystem of type <fstype> on device <device> at <mpoint>

example: mount fatfs /dev/ata0a /cf
mount fatfs /dev/sdmmc0a /sd
mount yaffs /dev/nand0a /yp
mount emu /dev/flash0a /e1
mount yaffs /e1 /yp
losh>
losh> mount fatfs /dev/sdmmc0a /sd
losh> cd /sd
losh> ls
:
: MLO 48420
: LBOOT.SUP 1532
: LBOOT.UAR 1787
: LBOOT.ELF 385152
losh>
    
```

You now have access to the files on the SD card.

## 7.4 Yet Another Flash File System (YAFFS)

Yet Another Flash Filing System (YAFFS) was developed by a company named Aleph One Limited and was then incorporated by Logic PD into the LogicLoader software. The partition entries for YAFFS partitions are not persistent—they must be restored on each boot. However, the partitions and data remain persistent.

This section will describe several ways to work with YAFFS in LogicLoader.

### 7.4.1 Develop a Partition Scheme

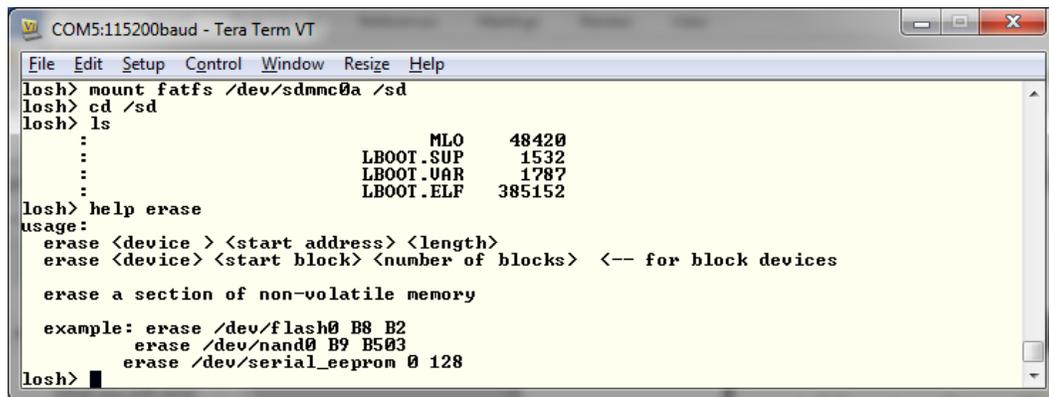
You can use LogicLoader to mount up to four YAFFS partitions at a time. The following limits are imposed on partitions:

- Each partition must have a unique name.
- Each partition must exist on local flash accessible from LogicLoader's `/dev/flashx` device file (where `x` is an instance index). For example: `/dev/flash0` or `/dev/flash1`.
- Each partition must span at least four physical flash blocks.
- A partition **must not** overlap the flash blocks that contain LogicLoader. For information about the location of these items, see the *LogicLoader User Guide Addendum* for your hardware.

### 7.4.2 Format YAFFS Partitions

All file systems need to be formatted before they can be mounted. Because YAFFS was designed from the ground up to work with embedded flash technologies, it understands an erased flash device to be both formatted and empty.

To prepare your partition for mounting, you should first use LogicLoader's `erase` command to erase the section of flash you are going to use. Several different parameters are available when using the `erase` command; to see what our options are, type `help erase`.



```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> mount fatfs /dev/sdmmc0a /sd
losh> cd /sd
losh> ls
:
:                               MLO      48420
:                               LBOOT.SUP  1532
:                               LBOOT.UAR  1787
:                               LBOOT.ELF  385152
losh> help erase
usage:
erase <device> <start address> <length>
erase <device> <start block> <number of blocks>  <-- for block devices

erase a section of non-volatile memory

example: erase /dev/flash0 B8 B2
         erase /dev/nand0 B9 B503
         erase /dev/serial_eeprom 0 128
losh>

```

The `help` command gives us three examples. The first is a NOR example and the second is a NAND example. The third example is for the serial EEPROM, which is only available on legacy SOMs.

Now that you know what the options are, it is a good idea to know what you are erasing before you do it. To find out what is in your kit's memory, use the *info mem* command.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
info mem
Configuration of /dev/nand0:
  Total Blocks:      4096
  Bytes Per Block:  135168
  Has Chunks:       yes
  Total Chunks:     262144
  Chunks Per Block: 64
  Bytes Per Chunk:  2112
  Bytes Per Spare:  64
  NoLo Partition:   Block 0
  /lboot Partition: Blocks 1 -> 17

Configuration of /dev/flash0:
  Base Address:     0x10000000
  Total Blocks:     64
  Bytes Per Block:  131072
  Has Chunks:       no

Configuration of SDRAM:
  Base:             0x80000000
  Size:             268435456

Configuration of SRAM_0:
  Base:             0x40200000
  Size:             65536

LoLo RAM Usage:
  .text:           0x80000000 -> 0x8005c514
  .bss:            0x8005c53c -> 0x800648a4
  Stack:          0x800648a4 -> 0x800678a4
  Translation Table: 0x80068000 -> 0x80070000
  Heap:           0x80070000 -> 0x80400000
  Download buffer: 0x80800000

Loaded Image:      None

Heap Usage:
  Available:       3735296
  Used:           126320
  Free:           3608976
  Maximum:        127076
  Errors:         0

Boot Loader Files:
  LoLo:           /lboot/lboot.elf

losh>
    
```

#### 7.4.2.1 A NAND Example:

NAND can be erased using blocks. If there are bad blocks marked by the manufacturer, you might need to adjust your partition location. Since LogicLoader now resides in NAND, let's look at where it is located so we can erase the rest.

1. At the `losh>` prompt, type `info part /dev/nand0`.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

losh> info part /dev/nand0
Partition Mounted Type ECC Method Name Start Length
-----
a:      Y      6c      bch4      a          1      17
b:      N      00      -----      Empty -----
c:      N      00      -----      Empty -----
d:      N      00      -----      Empty -----

losh>
    
```

On this kit, LogicLoader resides in the first partition, but all the other partitions are empty.

2. To erase the rest of NAND for our partition, skipping the area where LogicLoader resides from block 1 to 17, type `erase /dev/nand0 B18 B2030`.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

losh> info part /dev/nand0
Partition Mounted Type ECC Method Name Start Length
-----
a:      Y      6c      bch4      a          1      17
b:      N      00      -----      Empty -----
c:      N      00      -----      Empty -----
d:      N      00      -----      Empty -----

losh> erase /dev/nand0 B18 B2030
erasing nand: 100%
erased '/dev/nand0' start=0x12: len=0x7ee bytes/blocks skipped 0
losh>
    
```

If you set your erase parameters incorrectly and erase part of LogicLoader, your kit will not boot. LogicLoader will warn you, so you have the option to say no. If you do erase part of LogicLoader, you can recover by booting from an SD card and updating LogicLoader as described in Section 5.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

c:      N      00      -----      Empty -----
d:      N      00      -----      Empty -----

losh> erase /dev/nand0 B18 B2030
erasing nand: 100%
erased '/dev/nand0' start=0x12: len=0x7ee bytes/blocks skipped 0
losh> erase /dev/nand0 B16 B2030
dev nand erase: The area 0x210000 - 0x107be000 is protected.

***** WARNING!! *****

You are about to erase one or more protected areas. This area
may be protected because it contains data used by LoLo, or it may
be a partitioned space. If you erase this location you may loose
data in a partition, or LoLo may not be able to boot. Are you
sure you want to do this?

***** WARNING!! *****

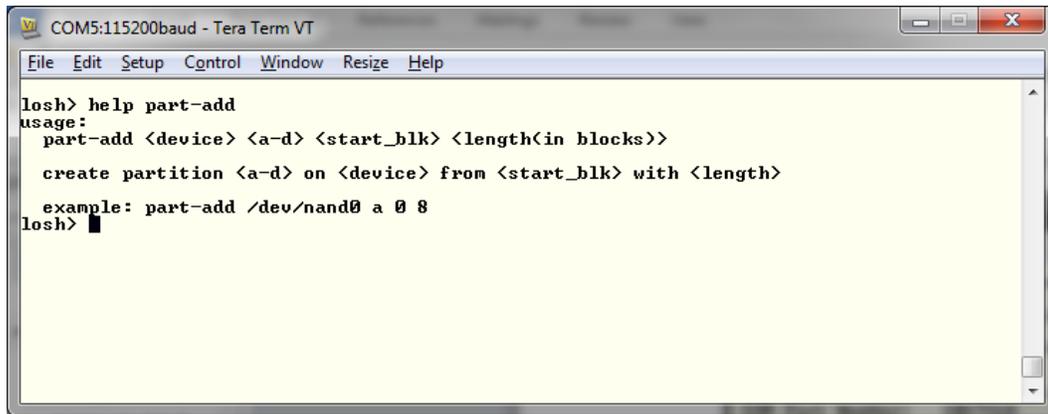
Do you wish to continue? <y/n>
    
```

### 7.4.3 Add YAFFS Type Partitions

LogicLoader maintains a partition table in RAM. Before a YAFFS partition can be mounted, it must be added to the partition table using the *part-add* command. The *part-add* command takes the following arguments:

- <device> - The type of flash device on which the partition resides.
- <partition> - The partition being added (a-d).
- <start> - The physical starting address of the partition.
- <length> - The length (in blocks) of the partition.

To begin, type `help part-add` to get additional information about the *part-add* command.

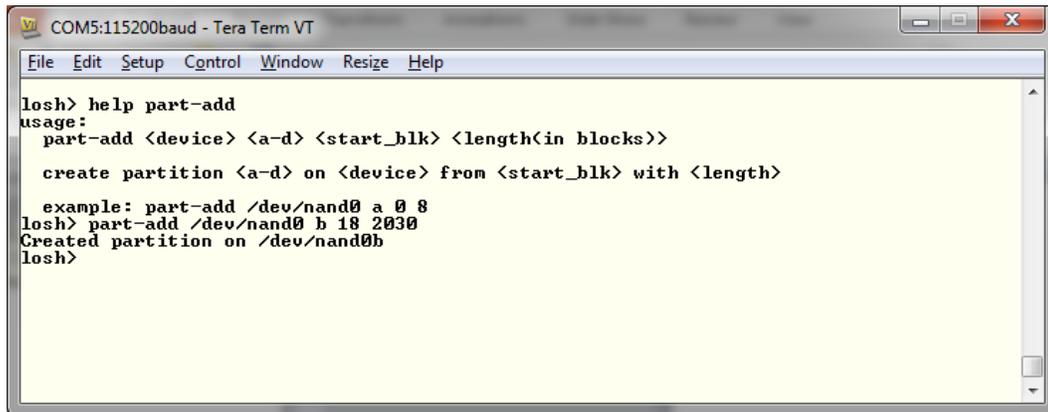


```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> help part-add
usage:
part-add <device> <a-d> <start_blk> <length(in blocks)>
create partition <a-d> on <device> from <start_blk> with <length>
example: part-add /dev/nand0 a 0 8
losh>

```

The device we're adding the partition to is *nand0*. Since LogicLoader resides in the first partition, this is the second partition and it will be partition b. The partition will start at block 18 and go to block 2048. So, type `part-add /dev/nand0 b 18 2030`.



```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> help part-add
usage:
part-add <device> <a-d> <start_blk> <length(in blocks)>
create partition <a-d> on <device> from <start_blk> with <length>
example: part-add /dev/nand0 a 0 8
losh> part-add /dev/nand0 b 18 2030
Created partition on /dev/nand0b
losh>

```

Why 2030 in the command? Remember that the second parameter is the length, not the end of the partition.

Notice that the partition was added on */dev/nand0b*. When we go to mount the partition, we'll need to use that in the *mount* command.

### 7.4.4 Mount a Partition

To mount a partition in the partition table, the *mount* command is used. This command takes the following arguments:

- <fstype> - The type of file system being mounted.
- <point> - The name of the YAFFS partition.

To use the *mount* command, type `mount yaffs /dev/nand0b /YaffsPart1`.

```

COM5:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

losh> help part-add
usage:
part-add <device> <a-d> <start_blk> <length(in blocks)>

create partition <a-d> on <device> from <start_blk> with <length>

example: part-add /dev/nand0 a 0 8
losh> part-add /dev/nand0 b 18 2030
Created partition on /dev/nand0b
losh> mount yaffs /dev/nand0b /YaffsPart1
YAFFS core: yaffs: Mounting YaffsPart1
losh>
    
```

Now change directories and see the contents of your new partition.

You can now store a file in nonvolatile memory so you don't have to download it to the kit every time you boot. Unfortunately, you still have to set up the partition and mount the partition after every power cycle.

LogicLoader also has a shortcut that will allow you to add the rest of flash to a partition and mount it all in one command; however, it only works if the entire NAND flash is blank. To use this shortcut, type `mount yaffs /dev/nand0 /YaffsPart1`.

```

COM3:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help

*****
                        LogicLoader
(c) Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****

losh> mount yaffs /dev/nand0 /YaffsPart1
YAFFS core: yaffs: Mounting YaffsPart1
losh>
losh>
    
```

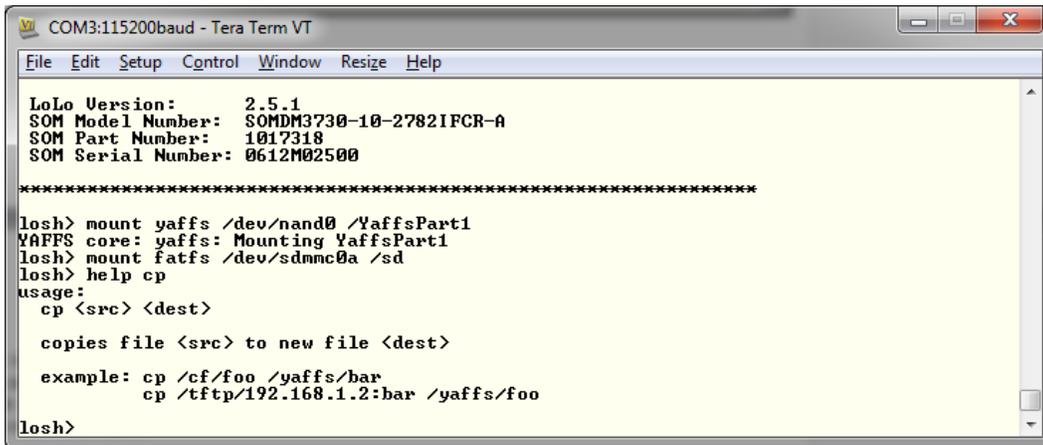
This is equivalent to using the following command sequence:

```

part-add /dev/nand0 b 18 2030
mount yaffs /dev/nand0b /YaffsPart1
    
```

### 7.4.5 Copy a File to the Partition

Now that you have the partition, let's put a file in it. Type `help cp` to get additional information about the `cp` command.



```

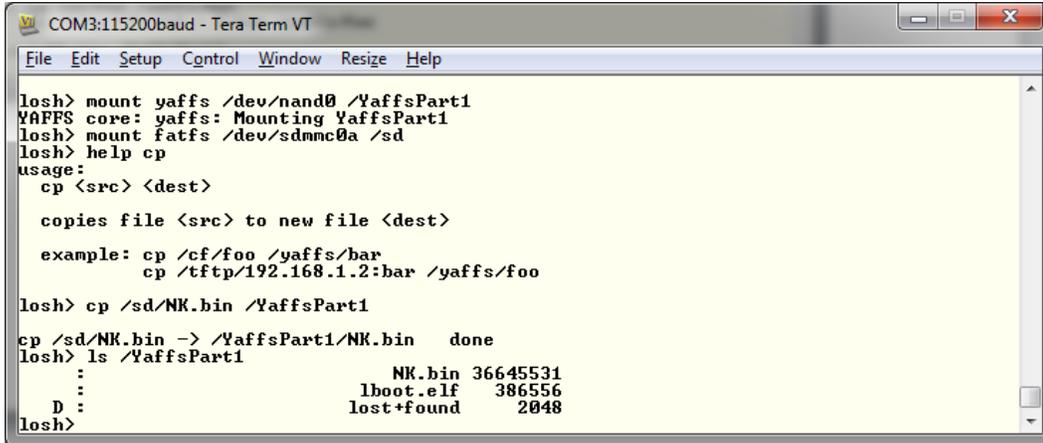
COM3:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:  1017318
SOM Serial Number: 0612M02500
*****
losh> mount yaffs /dev/nand0 /YaffsPart1
YAFFS core: yaffs: Mounting YaffsPart1
losh> mount fatfs /dev/sdnmc0a /sd
losh> help cp
usage:
cp <src> <dest>

copies file <src> to new file <dest>

example: cp /cf/foo /yaffs/bar
         cp /tftp/192.168.1.2:bar /yaffs/foo
losh>
    
```

The two examples are to copy from an SD card to a YAFFS partition or to copy from a TFTP to a YAFFS partition. If you are going to use TFTP, you will need to have the IP address of your TFTP server and an Ethernet connection to the server. Review Section 6 if you need a refresher on how to do this.

Using the example from the `help` window, we'll copy an `NK.BIN` file from an SD card to our YAFFS partition by typing `cp /sd/NK.bin /YaffsPart1`.



```

COM3:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> mount yaffs /dev/nand0 /YaffsPart1
YAFFS core: yaffs: Mounting YaffsPart1
losh> mount fatfs /dev/sdnmc0a /sd
losh> help cp
usage:
cp <src> <dest>

copies file <src> to new file <dest>

example: cp /cf/foo /yaffs/bar
         cp /tftp/192.168.1.2:bar /yaffs/foo
losh> cp /sd/NK.bin /YaffsPart1
cp /sd/NK.bin -> /YaffsPart1/NK.bin done
losh> ls /YaffsPart1
:
:          NK.bin 36645531
:          lboot.elf 386556
D :          lost+found 2048
losh>
    
```

In the next lab you will learn how to store scripts that can automate all of these processes and boot an image automatically.

## 8. Lab 7: Using Scripts

### 8.1 Lab Goals

This lab will teach you how to use scripting to optimize your application on the DM3730 Development Kit.

### 8.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- An image to load onto your development kit and boot with a script
- A terminal emulation program (e.g., Tera Term) with the following port settings:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**.

### 8.3 Connect the Kit to Your Host PC

1. Use the gray serial cable included in the development kit to connect the serial port on the kit to the appropriate port on your host PC.
2. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below.

The screenshot shows a terminal window titled 'COM5:115200baud - Tera Term VT'. The window contains the following text:

```

*****
                        LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.

LoLo Version:      2.5.1
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****

losh> █
    
```

**NOTE:** If you do not see the LogicLoader welcome prompt, please check to make sure the cable is connected correctly. If you still do not see the LogicLoader welcome prompt, the LogicLoader files may not be loaded properly in NAND Flash. Please refer to Section 5 for instructions on how to update and load LogicLoader.

### 8.4 Basic Scripting Guidelines

Scripts can be used to automate any command or command sequence that can be entered on the command line. Scripts can exist as a single text file containing the commands you want to execute separated by semicolons (;) or a text file with each command on a separate line.

You can enter scripts using the same syntax you use at the `losh>` prompt. In fact, that is exactly what you are doing. You are setting up a list of LogicLoader commands to be executed one after

the other. By separating those commands by a semicolon or a new line, you are telling LogicLoader how to handle the script.

## 8.5 Launching Scripts from a Text File

If your script is saved as a text file, you can run a script from Tera Term by using the *Send | File* option when LogicLoader is waiting at the `l0sh>` prompt. Let's make a very simple script called *video\_test* to open your display, clear the display, and run the draw-test.

Below is an example of what the text file should look like if the commands are separated by semicolons.

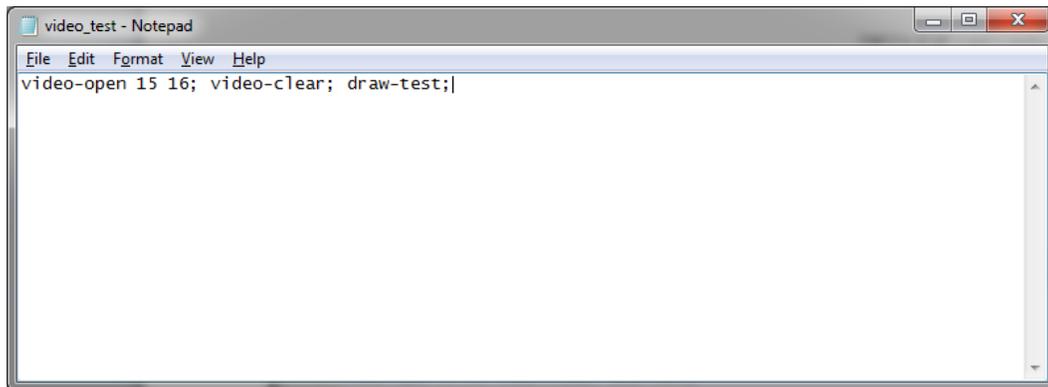
```
video-open 15 16; video-clear; draw-test;
```

Below is an example of what the text file should look like if each command has a separate line.

```
video-open 15 16  
video-clear  
draw-test
```

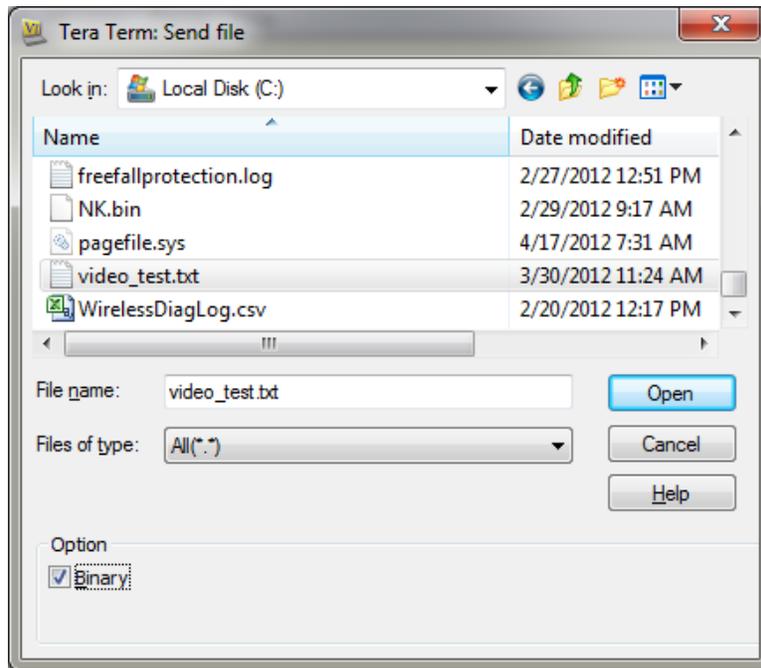
Be sure to place a carriage return after the last command to execute it; otherwise, you'll have to manually press Enter to see the test pattern.

1. Open a text editor program and create a text file that includes the commands above.

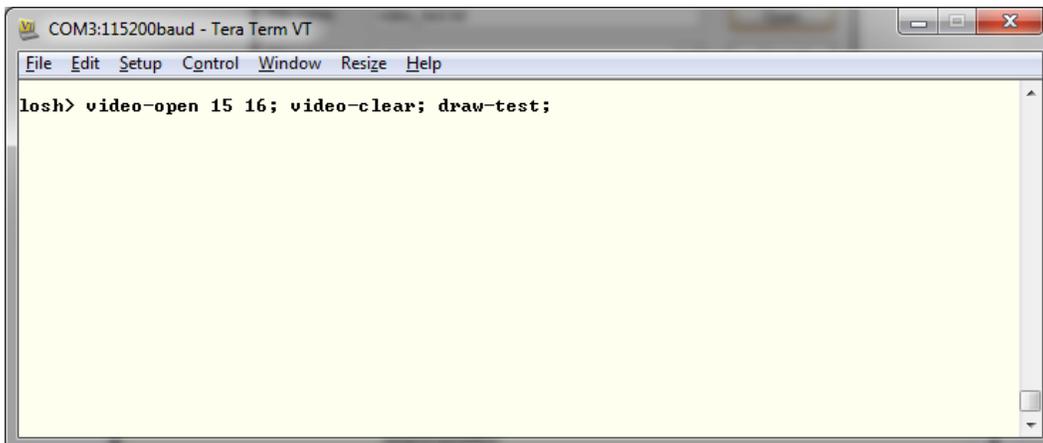


2. Save the file with the name *video\_test.txt*.

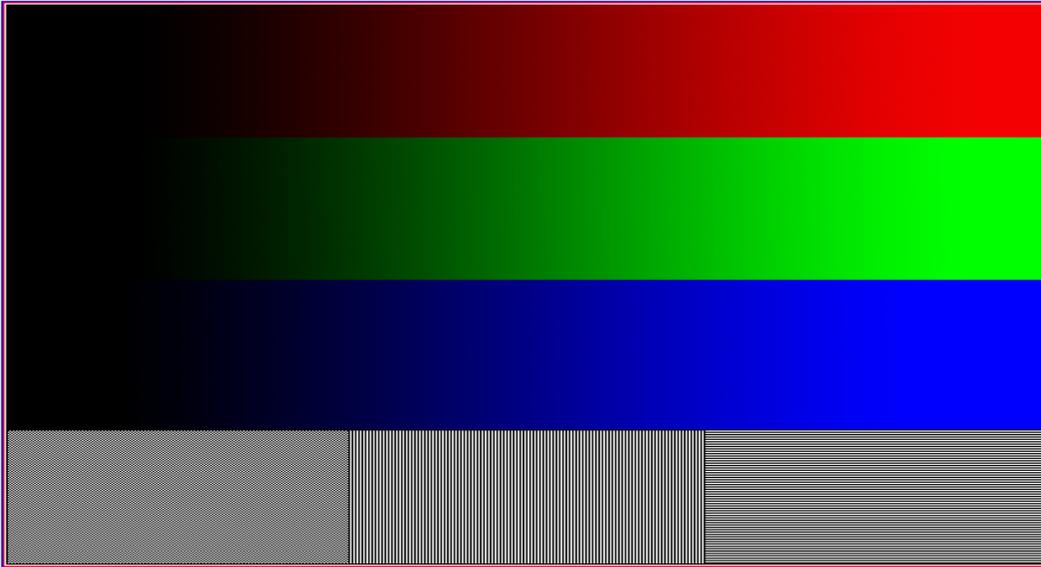
3. In Tera Term, select File > Send file....
4. Then locate the *video\_test.txt* file you just created. Click Open to send the script to your development kit.



5. The *Tera Term* window will display output information similar to that included below.



6. Your LCD should show the draw-test pattern included below.



7. If you watched your screen when you sent the script, it should have flashed something and then gone to the draw-test pattern.
8. Now, let's put some time in between the commands so we can see what is going on. Type **help sleep** to get additional information about the *sleep* command.

```

COM3:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> video-open 15 16; video-clear; draw-test;
video-open display: width: 480 height: 272 bpp: 16 disp: 15
losh> help sleep
usage:
  sleep <ms>

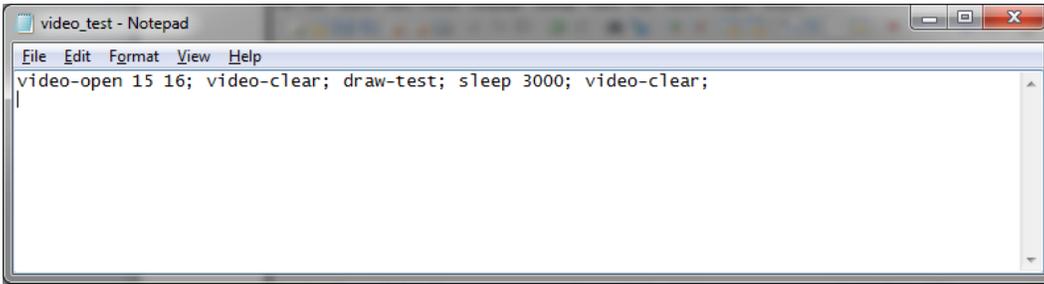
  sleep for <ms> number of milliseconds

  example: sleep 1
losh>

```

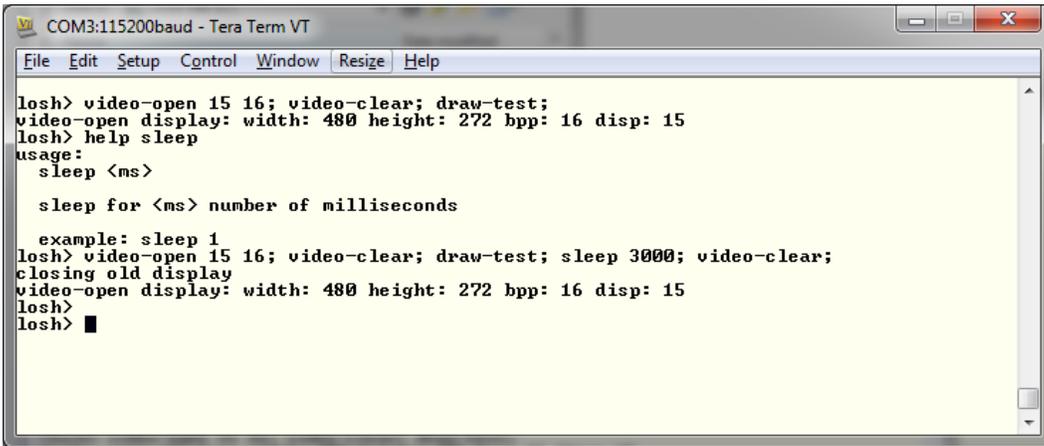
9. The *help* file lets us know we can make LogicLoader take a nap by adding the *sleep* command between commands in our script. After the *draw-test* command, let's add *sleep 3000*, which will add a three-second pause after the *draw-test* command. Let's also put another *video-clear* command at the end of the script so we can see the test pattern disappear.

With these changes, your *video\_test.txt* file should look similar to the example below.



```
video-open 15 16; video-clear; draw-test; sleep 3000; video-clear;
```

10. Save the text file and once again use *Send | File* to send the script. Now you can actually see what the script is doing with each command.



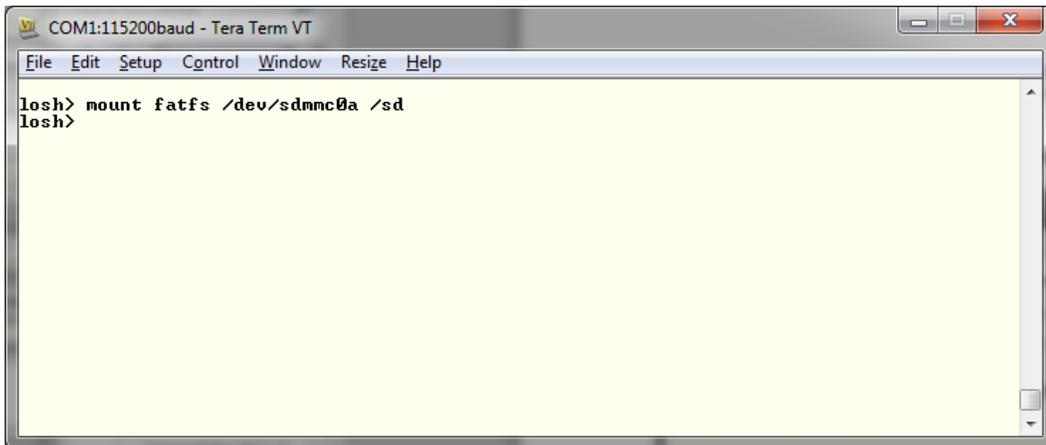
```
losh> video-open 15 16; video-clear; draw-test;
video-open display: width: 480 height: 272 bpp: 16 disp: 15
losh> help sleep
usage:
  sleep <ms>
  sleep for <ms> number of milliseconds
  example: sleep 1
losh> video-open 15 16; video-clear; draw-test; sleep 3000; video-clear;
closing old display
video-open display: width: 480 height: 272 bpp: 16 disp: 15
losh>
losh>
```

## 8.6 Launching Scripts from an SD Card

If you have a script saved as a text file on an SD card or on your development kit, you can use the *source* command to run the script. Let's add the necessary commands to mount the SD card and then run the script from the SD card. We will be using the commands you learned in Section 7 to mount the SD card.

1. Using your host PC and a USB card reader, save the *video\_test.txt* file that you created in Section 8.5 to an SD card. Remove the SD card from your USB card reader and insert the SD card into the SD card slot on your DM3730 Development Kit.

- At the `losh>` prompt, type `mount fatfs /dev/sdmmc0a /sd`.



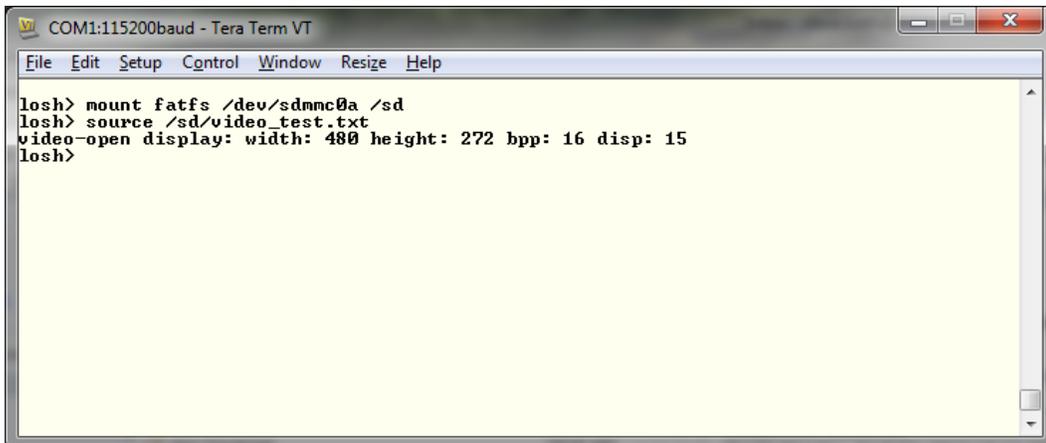
```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> mount fatfs /dev/sdmmc0a /sd
losh>

```

- Next, type `source /sd/video_test.txt`.

You should see the LCD on your development kit flash white, then show the test screen pattern, and then return to white. The *Tera Term* window will display output information similar to that included below.



```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
losh> mount fatfs /dev/sdmmc0a /sd
losh> source /sd/video_test.txt
video-open display: width: 480 height: 272 bpp: 16 disp: 15
losh>

```

Note that the commands from the script on the SD card did not appear on the screen, but the test pattern did appear for three seconds and then was cleared. What happens when you reboot your kit? Nothing; you have to go through the commands all over again. To overcome this, we'll look at how to store a script so it runs every time you power on your kit.

## 8.7 Persistent Script Storage

In order for a script to persist across power cycles, the script must be stored to a local, non-volatile memory device on the system. The two script storage mechanisms supported by LogicLoader on the DM3730 SOMs are the SD card and the resident flash array. LogicLoader has three different files that a user can modify to fit their specific requirements for boot-up configuration. The following is a brief description of each file. It is recommended that you review the *Release\_Notes.txt* file included with the LogicLoader version you downloaded and the *LogicLoader v2.5 User Guide* for more detailed information.

- The *lboot.sup* file is the first file read by the kit when it is powered up. This file is where you would add LogicLoader setup information. Since this file is loaded very early in the boot process, it can contain special configuration information as well. Such information may include setting processor pins that LogicLoader should avoid using.
- The *lboot.var* file is the second file read after power up. This file contains a list of variables that are created in the LogicLoader shell. These variables can be static values or pointers to addresses in memory.
- The *lboot.lol* file is the last file read after power up. This file contains LogicLoader shell commands necessary to load and launch the user's OS and any setup processes that could otherwise be done at the LogicLoader prompt.

For this example, we will be updating the *lboot.lol* script to include our *video\_test.txt* script upon startup.

1. Find the *lboot.lol* file that accompanied the *lboot.elf* file and open it with a text editor. Upon inspection of the file, you will see a general description for each file type, in addition to a brief explanation on how they all fit together.
2. Add the lines below to the end of the file.

```
mount fatfs /dev/sdmmc0a /sd
source /sd/video_test.txt
```

These commands should look familiar, as they were used in the previous section.

3. Save the file as *lboot.lol* in the same directory that the *lboot.elf* and *MLO* files are in. This will be the root directory (*/*) on the SD card if you are booting from the SD card. If you uploaded to NAND flash, it should be in the same partition and folder as the *lboot.elf* and *MLO* files.
4. Once the save is complete, cycle the power on the kit. You should see the results of the *video\_test.txt* file on the screen upon reboot.

If you store a boot script that doesn't return to LogicLoader, like a Windows CE image boot script, you won't be able to get back to the `l0sh>` prompt by rebooting. Instead, the kit will just run the script again. To get out of this situation, you can hold down the **q** key when you boot. LogicLoader checks the keyboard first and will not run the script if the **q** key is detected. When you return to the `l0sh>` prompt, you can disable your script if you need to.

Again, please see the *LogicLoader v2.5 User Guide* for more flexible/powerful programming options.

## 9. Lab 8: Loading an *NK.BIN* File

### 9.1 Lab Goals

This lab will teach you how to use scripting to load an *NK.BIN* file and boot it on the DM3730 Development Kit.

### 9.2 Prerequisites

- A DM3730 Development Kit
- A host PC or laptop
- An available serial port or USB-to-serial adapter
- An SD Card
- An Ethernet connection between the development kit and host PC
- A terminal emulation program (e.g., Tera Term) with the following port settings:
  - a. Baud rate: **115200**
  - b. Data: **8 bit**
  - c. Parity: **None**
  - d. Stop: **1 bit**
  - e. Flow control: **None**.

### 9.3 Connect the Kit to Your Host PC

1. Use the gray serial cable included in the development kit to connect the serial port on the kit to the appropriate port on your host PC.
2. Power on the development kit. You should see a LogicLoader welcome prompt similar to that included below.

The screenshot shows a terminal window titled "COM5:115200baud - Tera Term VT". The window contains the following text:

```

*****
                        LogicLoader
<c> Copyright 2002-2011, Logic PD, Inc. All Rights Reserved.
LoLo Version:      2.5.1
SOM Model Number:  SOMDM3730-10-2782IFCR-A
SOM Part Number:   1017318
SOM Serial Number: 0612M02500
*****
losh> █
    
```

**NOTE:** If you do not see the LogicLoader welcome prompt, please check to make sure the cable is connected correctly. If you still do not see the LogicLoader welcome prompt, the LogicLoader files may not be loaded properly in NAND Flash. Please refer to Section 5 for instructions on how to update and load LogicLoader.

### 9.4 Load a File from an SD Card

The DM3730 processors have the ability to boot directly from an SD card. This ability is what allows Logic PD to ship SD cards that boot directly to Android, Linux, and Windows CE OSs with the kit. If for some reason you would like to boot from NAND and then execute an OS image, you

can do that by following the process explained here. The process for YAFFS is very similar and will be covered below in Section 9.5.

In previous labs, you learned how to mount an SD card. We will now go a step further to load an *NK.BIN* Windows CE image from the SD card onto the development kit.

1. First, type `help load` to get additional information on the `load` command.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
SOM Model Number: SOMDM3730-10-2782IFCR-A
SOM Part Number: 1017318
SOM Serial Number: 0612M02500
*****
losh> help load
usage:
load <type> [source]
      <type> /tftp/<server:filename>
      raw <dest addr> <length> [source]
      <type> -dhcp

      <type> can be 'elf', 'bin' (wince img), 'raw' or 'srec'
      when no [source] is given, stdin is read
      -dhcp looks in the boot packet for a filename

download a binary image

example: load elf
losh>
    
```

You can see from the output that there are several ways to use the `load` command. We are interested in loading a Windows CE image, so we will use the `load bin` command.

2. Use the commands you learned in Section 7 to mount the SD card.
3. Now type `ls /sd` to view the contents of the SD card.

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Resize Help
load <type> [source]
      <type> /tftp/<server:filename>
      raw <dest addr> <length> [source]
      <type> -dhcp

      <type> can be 'elf', 'bin' (wince img), 'raw' or 'srec'
      when no [source] is given, stdin is read
      -dhcp looks in the boot packet for a filename

download a binary image

example: load elf
losh> ls /sd
:
:           MLO      48420
:           LBOOT.SUP 1532
:           LBOOT.UAR 1787
:           LBOOT.ELF 385152
:           NK.BIN 36645531
:           video_test.txt 68
:           LBOOT.LOL 2430
losh>
    
```

You can see a file called *NK.BIN*; that is the Windows CE image we are going to load.







```

lboot - Notepad
File Edit Format View Help
# at the time the file is loaded.  whereas, the last file loaded
# has all the resources available, and can leverage LoLo's command
# line parser to simplify the readability of the file.  Users
# should consider this fact when laying out a new design.  Consider
# future maintenance of the files, and realize the file(s) loaded
# last in the boot process will generally be more flexible than the
# early loading counterparts.

echo "\n\nHello I am lboot.lol, running from $SYS_BOOT_DEVICE\n\n"

echo "SYS_INTEGRITY_PASS: $SYS_INTEGRITY_PASS\n"
echo "SYS_INTEGRITY_FAIL: $SYS_INTEGRITY_FAIL\n"

info device /dev/pm0
info var example_var

ifmac sm0
ifconfig sm0 dhcp
mount fatfs /dev/sdmmc0a /sd
load bin /sd/NK.BIN
exec
exit

```

For YAFFS on NAND:

```

ifmac sm0
ifconfig sm0 dhcp
mount yaffs /dev/nand0 /YaffsPart1
load bin /YaffsPart1/NK.BIN
exec
exit

```

3. Save the file as *lboot.lol* to the same directory that the *lboot.elf* and *MLO* files are in. This will be the root directory (/) on the SD card if you are booting from the SD card. If you uploaded the file to NAND flash, it should be in the same partition and folder as the *lboot.elf* and *MLO* files.
4. Once the save is complete, cycle the power on the kit. You should see the Windows CE desktop on the kit's display after a few moments.

If you store a boot script that doesn't return to LogicLoader, like a Windows CE image boot script, you won't be able to get back to the `l0sh>` prompt by rebooting. Instead, the kit will just run the script again. To get out of this situation, you can hold down the **q** key when you boot. LogicLoader checks the keyboard first and will not run the script if the **q** key is detected. When you return to the `l0sh>` prompt, you can disable your script if you need to.

Again, please see the *LogicLoader v2.5 User Guide* for more flexible/powerful programming options.