# SmartMesh IP Easy Start Guide for VManager

# Table of Contents

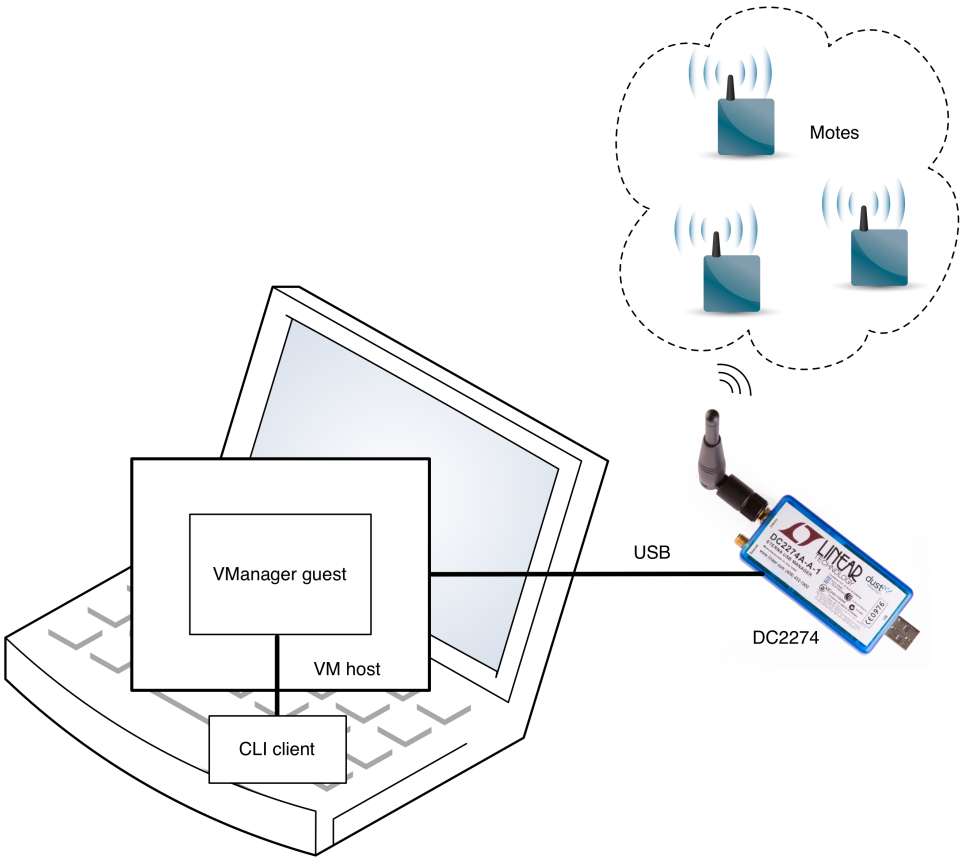

1 Introduction 3
1.1 Revision History 4
2 Basic Steps 5
2.1 Steps 1 & 2: VManager Installation and AP Bridge Configuration 5
2.2 VManager Download 5
2.3 VManager Installation - VirtualBox 5
2.4 VManager Installation - VMware 6
2.5 AP Mote Programming & Configuration 6
2.6 AP Mote Installation - On Local Host 7
2.7 Step 3: Form a Wireless Mesh Network 8
2.8 Step 4: Bring Up the Manager CLI 9
2.9 Ping a Mote 10
2.10 That's It! 11
2.11 Next Steps - Software and Hardware Design 12
2.11.1 Software Design 12
2.11.2 Hardware Design 12

# 1 Introduction

The purpose of this document is to provide a quick and easy out-of-the-box experience with the SmartMesh IP VManager.

You will use components in the VManager Setup figure:

- A SmartMesh IP starter kit including a DC2274A-A or DC2274A-B and one or more motes
- A VM host program running the VManager guest image
- A terminal application to communicate with the manager's command line interface (CLI)




**VManager Setup**

## 1.1 Revision History

| Revision | Date | Description |
|---|---|---|
| 1 | 12/17/2015 | Initial Release |
| 2 | 08/19/2016 | Phase I Production |
| 3 | 10/31/2016 | Added DC9021B kit information |

# 2 Basic Steps

There are four steps to demonstrate the capabilities of the SmartMesh IP VManager:

1. Install the VM host and configure the VManager guest image
2. The DC2274A-B in the latest DC9021B evaluation kit comes pre-programmed as an APM. Alternatively, the DC2274A-A in other kits may be re-program and configured to make it an Access Point Mote (APM)
3. Form a wireless mesh network - just power the motes on and a network will form
4. Bring up the manager command line interface (CLI) - this will allow you to see the network on the manager

Once you've done these four steps you'll perform one exercise - namely ping a mote in the network to see its temperature and battery voltage.



The CLI is intended for human interaction with a mote or manager, e.g. during development, or for interactive troubleshooting. The API is intended for machine-to-machine communications, e.g. a host program talking to the manager or a sensor processor talking to the mote.

## 2.1 Steps 1 & 2: VManager Installation and AP Bridge Configuration



For the purposes of this evaluation, the AP Bridge Software is configured to use the default AP Mote clock source (Auto). AP Motes with a GPS time source should not be used.

## 2.2 VManager Download

The VManager can be found in two separate files as follows:

- VManager_xxxx.ova: This is a full Virtual Machine image containing the VManager
- SmartMesh IP.zip: This file contains all IC based software that is required for the AP

Both files are available for download through your MyLinear account. Contact your local sales representative to gain access through your myLinear document locker.

## 2.3 VManager Installation - VirtualBox

1. Download **VirtualBox version >= 5.x** at https://www.virtualbox.org/wiki/Downloads
2. Download the **VirtualBox Extension Pack** from the same site

3. Install both VirtualBox and the Extension Pack on your machine as instructed by the installation software, a reboot will be required
4. Download the VManager package from the myLinear document locker as described above
5. Import the VManager instance into VirtualBox
    - Under **File** --> **Import Appliance**... select the file `VManager_xxxx.ova`
    - Make NO changes to the settings and click the **Import** button
6. **Start** the virtual machine now installed called "VManager"

## 2.4 VManager Installation - VMware

1. Download VMware Workstation Player for Windows at https://my.vmware.com/en/web/vmware/free#desktop_end_user_computing/vmware_workstation_player/12_0
    - Note: It is possible to run as a free player, but a license should be purchased for commercial use
2. Install VMware Workstation Player on your machine as instructed by the installation software
3. Download the VManager package from the myLinear document locker as described above
4. Open the VManager instance from VMware Workstation Player
    - Under Player -> File -> Open, select the file VManager_xxxx.ova
    - Click the Import button
    - If you see a pop warning "*The import failed because ... did not pass OVF .... Click Retry to relax OVF ...*"
        - click Retry
5. Select the virtual machine
    - Click "*Edit virtual machine settings*"
    - Under the "Hardware" tab -> "Network Adapter", Select "*Bridged: Connected directly to the physical network*"
    - Click OK
6. Start the virtual machine called "VManager" by selecting it and clicking the "Play" button

## 2.5 AP Mote Programming & Configuration

If you are using the latest DC9021B evaluation kit, it includes a DC2274A-B which is already programmed and configured as an AP Mote. This step can be skipped.

If you are using a DC2274A-A reference board or customer built AP Mote board can also be used. A standard DC90xx reference device cannot be used since it is missing some connections.

The AP Mote hardware is the LTC58xx IC, LTM58xx micromodule, or LTP59xx module programmed with the AP software binary. Unlike the DC90xx reference devices that are available in the SmartMesh IP kits, the AP Mote PCB requires that two additional signals be made available to the AP Bridge software process, namely the TIMEn pin for a PPS (Pulse-Per-Second) from an optional GPS device, and the reset pin. The DC2274A-A and DC2274A-B reference boards contain these additions and support the AP Bridge software.

Programming a DC2274A-A as an AP Mote:

- Install the Eterna Serial Programmer Utility (ESP). It is available in the `SmartMesh Tools.zip` file in your MyLinear document locker. You can extract its contents anywhere convenient.
- Connect your DC2274A-A to a USB port - if this is the first connection, Windows should launch the FTDI driver installer. If not, see the troubleshooting section of the SmartMesh IP VManager User's Guide.
- Copy the `prog_APM_DC2274A-A.bat` file and supporting image components from the SmartMesh IP.zip file ( ../Eterna/AP Mote ) into your ESP folder.
- Run the `.bat` file to re-program your DC2274A-A into an Access Point Mote.

For default operation without using GPS as a time source, the AP Mote can be used "as-is" in its default state. If however the network is configured with a GPS time source, the AP Mote(s) will require configuration, the clock sources must be assigned as described in the "Access Points" section of the SmartMesh IP VManager User's Guide. The configuration instructions for the AP Mote are in the "Configuring the Gateway" section of the "VManager AP Bridge User's Guide".

## 2.6 AP Mote Installation - On Local Host

The AP Bridge software is already installed on the VManager VM and will automatically be used if an Access Point (like a DC2274A-B) is connected directly to that host computer. If an AP Mote is to be used locally, i.e. installed on the same computer as the VManager instance, then follow these steps:

- Enable port forwarding in the virtual machine. The VManager virtual machine must not be running.
  - In VirtualBox --> go into the **Settings** menu window and select **USB**, then enable the USB ports (select USB 3.0)
  - In VMware --> go into "Edit virtual machine settings" and select "USB Controller" under the Hardware tab, choose USB 3.0 for USB compatibility.

- Plug one or more AP Mote(s), such as the DC2274A-B, into a USB port
- Start the virtual machine

- Attach the USB DC2274A AP Mote(s) to the VM
  - In VirtualBox --> In the **Settings**->**USB** menu, click the "**+**" sign on the right and select the the evaluation board(s) that appear
    - For example --> LTC DC2274A WITH MEMORY 60xxx [0800]
  - In VMware --> From the virtual machine screen's top right corner, right click the USB stick icon, select "*Connect (Disconnect from host)*"

- Open an SSH window to the VManager machine with Putty (or similar application)
  - In VirtualBox --> Connect to localhost:2222 (from the host running the VM)
  - In VMware --> Connect to <Host IP address>:22

- Login using these credentials
  - User = `dust`
  - Password = `dust`

- Execute the following commands to configure the AP Mote(s) installed by typing the input after the prompt (your $ prompt line may differ). Note that the following shows an example where two AP Motes are connected to the system.

```
$ update-apc-config auto
   --> You will be prompted for the dust user password ... "dust"
installed udev rules, APC configuration will be created automatically when an AP mote is connected
Create APC configuration for LTC_DC2274A_WITHOUT_MEMORY_603528
 supervisor conf file created
 APC conf file for apc-603528 created
apc-603528: available
apc-603528: added process group
Create APC configuration for LTC_DC2274A_WITHOUT_MEMORY_603983
 supervisor conf file created
 APC conf file for apc-603983 created
apc-603983: available
apc-603983: added process group
No config updates to processes
```

- To verify that each AP Bridge instance (APC) has been correctly installed, run `update-apc-config` which should yield the following results:

```
$ update-apc-config
Name         Host              Port    api-device       reset-device
----------   ---------------   -----   --------------   --------------
apc-603528   localhost         9100    /dev/serial/by+  /dev/serial/by+
apc-603983   localhost         9100    /dev/serial/by+  /dev/serial/by+
```



The AP Bridge software is configured by default to use the *Auto* clock source for all AP Motes in the system. No changes are needed unless GPS timing is used at the AP Motes.

## 2.7 Step 3: Form a Wireless Mesh Network

Even though much of the effort associated with this guide involves using the PC, it is important to point out that your PC has nothing to do with forming a wireless mesh network. Once the VManager is running on a server (which could be anywhere in the world), your client application is completely independent of the server.

Power on one mote using the slide switch. Watch the two green LEDs. One will blink slowly, indicating the mote is searching for the network. When that green LED changes from blinking to solid, that means the mote has sent a join request. When the second green LED turns on, the mote has joined.

Power on all the remaining motes, and within one or two minutes you will have a complete wireless mesh network running.

## 2.8 Step 4: Bring Up the Manager CLI

The CLI allows you to type commands from a PC to interact with the manager. Interacting with the CLI requires an SSH client such as TeraTerm or PuTTY, or a Linux console.

Either use the SSH connection from step 2, or open a new connection. By default, the VirtualBox VManager configuration forwards port 2222 to the SSH port of the guest, so connect to <Host IP address>:2222. The default username and password are both `dust`. So if the host address is 192.168.1.100, run:

```
$ ssh dust@192.168.1.100 -p 2222
dust@192.168.1.100's password:

Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-71-generic i686)

 * Documentation:  https://help.ubuntu.com/

  System information as of Wed Dec 16 12:00:40 PST 2015

  System load:  0.0               Processes:           82
  Usage of /:   4.7% of 39.34GB   Users logged in:     0
  Memory usage: 16%               IP address for eth0: 10.0.2.15
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

16 packages can be updated.
13 updates are security updates.

Last login: Wed Dec 16 12:00:43 2015 from 10.0.2.2
dust@vmanager-vm:~$
```

Launch the CLI application by calling the ***console*** command. Again, the default username and password are both `dust`. Note that while the login has the same default values, the API/CLI login is separate from the VManager guest OS Linux login.

```
dust@vmanager-vm:~$ console

Welcome to the VManager CLI Console on Linux
Version 1.0.1.10 (built 2016/06/03 11:18:19)
Enter your username: dust
dust's password:
$>
```

Type 'help' to see a list of commands. We are interested in two commands: `ping`, and `sm`

```
$> help
For more information on a specific command, type HELP command-name
clear           Clear screen
config          Stored configuration commands
exec            Execute a network command
exit            Exit console
help            Show help for a command
logout          Exit console
ping            Ping a mote or AP
quit            Exit console
reset           Reset a mote, AP mote, or the network.
show            Show system and/or network information
sm              Show motes
trace           Control console traces
version         Show version
```

Use the `sm` command (this is short for "show motes") to obtain the list of connected motes:

```
$> sm
AP MAC                  Id  Clk State  State time  Age Jn  Nbrs Links
----------------------- --- --- ----- ------------ --- --- ---- -----
00-17-0D-00-00-60-3A-DC   1 Int  Oper   1-23:20:55 1122   1   12    66

Mote MAC                Id  State  State time  Age Jn  Nbrs Links
----------------------- --- ----- ------------ --- --- ---- -----
00-17-0D-00-00-12-34-56   2  Oper   1-23:18:16  18   1   12    15
00-17-0D-00-00-38-00-F5   3  Oper   1-23:19:37   3   1   12    17
00-17-0D-00-00-30-47-3A   4  Oper   1-23:19:37   7   1   12    17
00-17-0D-00-00-30-43-2A   5  Oper   1-23:19:47   2   1   12    17

APs: 1, Motes: 4, 5 live, 0 joining
```

## 2.9 Ping a Mote

You can interact with your motes via the manager CLI. Type the following command to send a "ping" to a mote. The mote will respond with battery voltage (in mV) and temperature (in °C) and the manager will add the round-trip time when printing the response.

```
$> help ping
Usage: ping <macAddr|moteId>

Send a ping to a specific mote or AP.
$> ping 00-17-0D-00-00-12-34-56
Done. Callback id: 1

$>
PING 2015-12-17 11:03:18.149 Reply from Mote #2, mac: 00-17-0d-00-00-12-34-56
  CallbackId: 1, Latency: 6508ms (1 hops), Data: 3237mV, 24C
```

Note that this is an internal mote command, not an ICMPv6 Echo request.

## 2.10 That's It!

You've successfully set up a network and interacted with the manager and a mote. There are many other things you can do with the manager CLI. See the SmartMesh IP VManager CLI Guide for details. Many of the SmartMesh IP Application Notes involve interacting with the manager CLI.

# 2.11 Next Steps - Software and Hardware Design

With the starter kit, hardware design and software design may be done separately.

## 2.11.1 Software Design

- The application note "Data Publishing for SmartMesh IP" walks through the steps needed to use the mote APIs to join a network and send data. The mote needs to be placed into **slave** mode via its CLI in order to activate the API - evaluation kit motes ship in **master** demo mode. See the " Master vs. Slave" section of the SmartMesh IP User's Guide for details on changing the mode.
- The SmartMesh IP User's Guide defines basic network terms and concepts, and discusses the use of APIs at a high level.
- At a minimum, a mote application needs to:
    - Configure any parameters needed prior to join (such as *joindutycyle*)
    - Use the *join* API to cause a mote to begin searching for a network
    - Monitor the mote state to see when it is ready to accept data
    - Open a socket and bind a UDP port to it
    - Request services in order to publish data
    - The SmartMesh IP Mote API Guide covers other commands to configure the mote.
    - The SmartMesh IP Mote CLI Guide covers using the human interface to observe mote activity.
- At a minimum, a host application connected to the manager needs to:
    - Configure any parameters needed prior to join (such as *networkID*)
    - Subscribe to notifications to observe mote status and collect data
    - The SmartMesh IP VManager API Guide covers other commands to configure the manager, e.g. configure security (use of ACL), or collect detailed statistics from Health Report notifications.
    - The SmartMesh IP VManager CLI Guide covers using the human interface to observe manager activity (including traces of mote state or data).
- Some of the Advanced software topics covered in the SmartMesh IP User's Guide, and SmartMesh IP Application Notes include:
    - Over-the-Air-Programming
    - Using the TestRadio API commands for top-level assembly testing

## 2.11.2 Hardware Design

- Select a hardware platform - modularly certified or chip level?
- The hardware integration guides (see the SmartMesh IP webpage for links to specific guides) cover the important considerations for robust hardware development.

### Trademarks

Eterna, Mote-on-Chip, and SmartMesh IP, are trademarks of Dust Networks, Inc. The Dust Networks logo, Dust, Dust Networks, and SmartMesh are registered trademarks of Dust Networks, Inc. LT, LTC, LTM and  are registered trademarks of Linear Technology Corp. All third-party brand and product names are the trademarks of their respective owners and are used solely for informational purposes.

### Copyright

This documentation is protected by United States and international copyright and other intellectual and industrial property laws. It is solely owned by Linear Technology and its licensors and is distributed under a restrictive license. This product, or any portion thereof, may not be used, copied, modified, reverse assembled, reverse compiled, reverse engineered, distributed, or redistributed in any form by any means without the prior written authorization of Linear Technology.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015 (b)(6/95) and DFAR 227.7202-3(a), and any and all similar and successor legislation and regulation.

### Disclaimer

This documentation is provided “as is” without warranty of any kind, either expressed or implied, including but not limited to, the implied warranties of merchantability or fitness for a particular purpose.

This documentation might include technical inaccuracies or other errors. Corrections and improvements might be incorporated in new versions of the documentation.

Linear Technology does not assume any liability arising out of the application or use of any products or services and specifically disclaims any and all liability, including without limitation consequential or incidental damages.

Linear Technology products are not designed for use in life support appliances, devices, or other systems where malfunction can reasonably be expected to result in significant personal injury to the user, or as a critical component in any life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness. Linear Technology customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify and hold Linear Technology and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Linear Technology was negligent regarding the design or manufacture of its products.

Linear Technology reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products or services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to Dust Network's terms and conditions of sale supplied at the time of order acknowledgment or sale.

Linear Technology does not warrant or represent that any license, either express or implied, is granted under any Linear Technology patent right, copyright, mask work right, or other Linear Technology intellectual property right relating to any combination, machine, or process in which Linear Technology products or services are used. Information published by Linear Technology regarding third-party products or services does not constitute a license from Linear Technology to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from Linear Technology under the patents or other intellectual property of Linear Technology.

Dust Networks, Inc is a wholly owned subsidiary of Linear Technology Corporation.

© Linear Technology Corp. 2012-2016 All Rights Reserved.