



Ultra low power IEEE 802.15.4 compliant wireless sensor module

Humidity, Light, and Temperature sensors with USB

Quick Start Guide

This document describes how to establish a development environment for Tmote on a Windows PC using the Tmote Tools CD. It describes how to install the Tmote development tools, how to install the USB driver to access Tmote as a virtual COM port, how to compile and install a TinyOS application, and how to run basic Java tools such as SerialForwarder to read and write data to and from the wireless network.



This document is not a comprehensive guide for developing for Tmote; instead the basic steps for getting started with TinyOS and Tmote are presented in this guide. Other useful documents for development include the Tmote Datasheet and the Tmote Application Notes found at <http://www.moteiv.com>



Contents

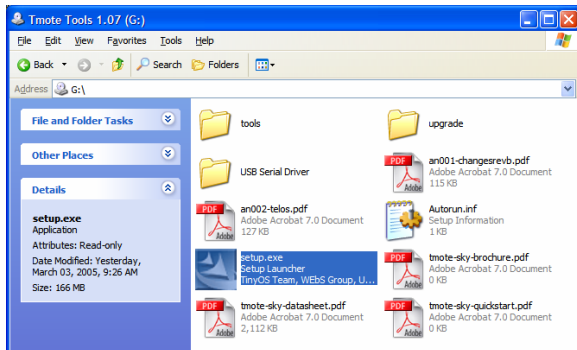
| | |
|---|----|
| Quick Start Guide..... | 1 |
| Contents..... | 2 |
| Windows Quick-Start | 3 |
| Tools Install..... | 3 |
| Tools Upgrade..... | 5 |
| USB Serial COM Driver Install | 6 |
| USB Serial COM Driver Install | 6 |
| Application Compile and Reprogram | 8 |
| Running Java Tools | 10 |
| Bringing It All Together..... | 11 |
| Tmote/TinyOS Manual Install for Microsoft Windows | 12 |
| Cygwin | 12 |
| Initial Environment..... | 12 |
| Java..... | 12 |
| JavaComm..... | 13 |
| FTDI VirtualComm USB Drivers..... | 13 |
| TinyOS CVS Repository..... | 13 |
| MSPGCC | 14 |
| Patched MSP430 Assembler | 14 |
| MSP430-BSL for Tmote | 15 |
| MoteList..... | 15 |
| TinyOS Scripts | 15 |
| Environment..... | 16 |
| nesC..... | 16 |
| TinyOS Java Tools..... | 16 |
| General Information | 17 |
| Document History..... | 17 |
| Disclaimer | 17 |
| Address Information | 18 |
| Headquarters | 18 |

Windows Quick-Start

This section guides you through the process of installing the Tmote tools and COM drivers, how to compile and install applications, and how to run Java tools like SerialForwarder.

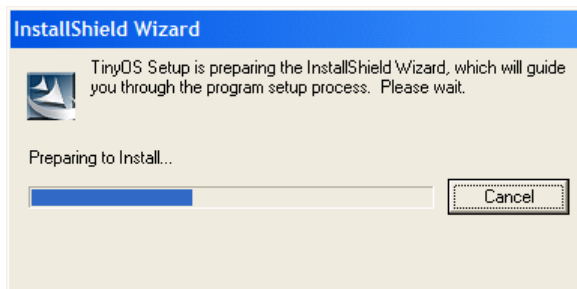
Tools Install

A complete install includes Cygwin, Java, the MSPGCC MSP430 compiler, TinyOS sources, and other supporting programs.

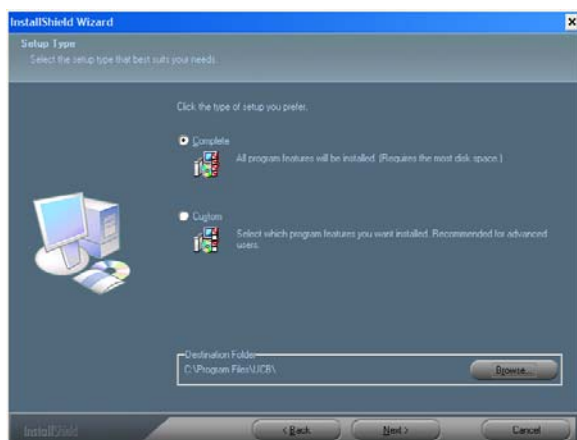


Run **setup.exe** from the Tmote Tools CD.

Note: If you have an existing system with **Cygwin** installed, we *highly recommend* that you uninstall it first to ensure the Tmote Tools installer can create a consistent and complete installation.



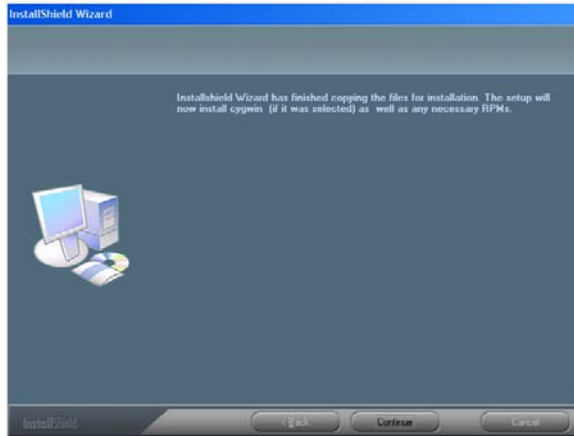
The installer may spend some time unpacking data in an “InstallShield Wizard” window.



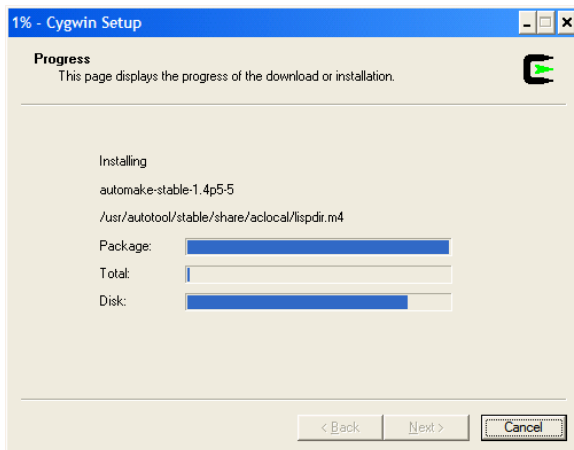
Once the main InstallShield Wizard window opens, if this is a clean system, select “Complete” install, which installs Cygwin, Java, the MSPGCC MSP430 compiler, TinyOS sources, USB serial drivers, and other programs. If you already have some of these programs installed, do a “Custom” install for only the new things you need.

Note: If you have an existing system with **Cygwin** installed, we *highly recommend* that you uninstall it first to ensure the Tmote Tools installer can create a consistent and complete installation.

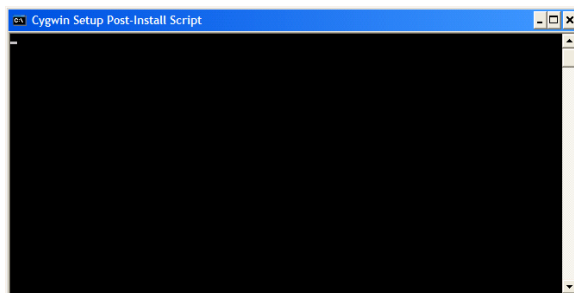
The installer then goes through the formalities such as asking you to read the Java licenses before it installs those packages and verifying your install options before it continues.



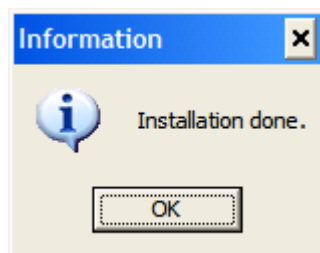
After completing the initial install phase, the installer continues on to install Cygwin and its related packages.



Cygwin continues the installation in a "Cygwin Setup" window.



Then, Cygwin performs some additional installation operations in a series of "Cygwin Setup Post-Install Script" windows.



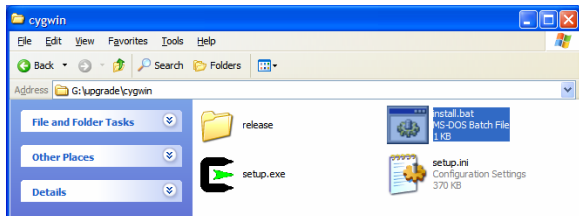
Everything is done installing when this "Information" window appears.

You are now ready to start programming for Tmote!

Tools Upgrade

NOTE: This section is ONLY for users who have previously installed TinyOS. If you followed the steps to install TinyOS on the previous pages, please skip this section.

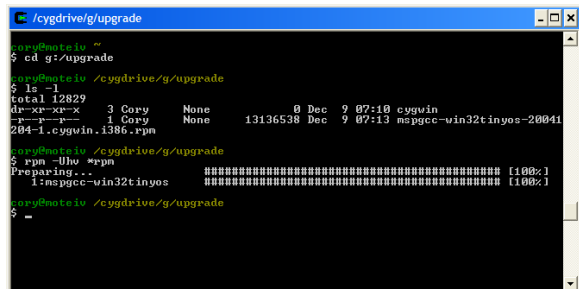
If you already have a previous version of Tmote Tools or Telos Tools installed, you can follow these steps to upgrade your system for Tmote.



First we need to update your Cygwin installation. Go to the folder upgrade/cygwin/ on the Tmote Tools CD and run "install.bat".

This batch file assumes you installed the previous Tmote Tools or Telos Tools into the recommended location C:\Program Files\UCB\.

If you elected to install to a different location, you can open a Command Window ("cmd.exe") and enter and run an appropriately modified command in the upgrade/cygwin folder similar to the one in install.bat.



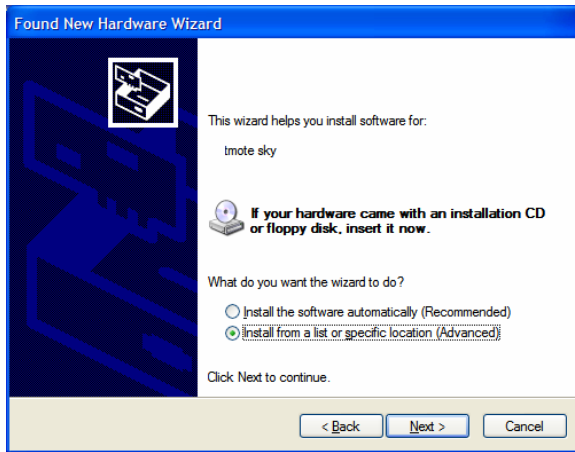
Then open a Cygwin shell, go to the upgrade directory on the Tmote Tools CD, and install all the RPM packages in there with the command:

```
rpm -Uhv *rpm
```

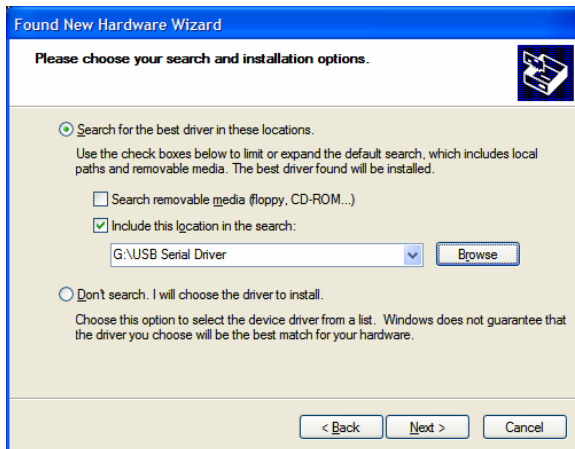
USB Serial COM Driver Install

Microsoft Windows now automatically installs the necessary drivers if connected to the Internet. This section describes the process to install those drivers for Tmote if they are not already installed and an Internet connection is not available. The first time a Tmote module is plugged into a system, if the drivers are not found, Windows opens the “Found New Hardware Wizard”.

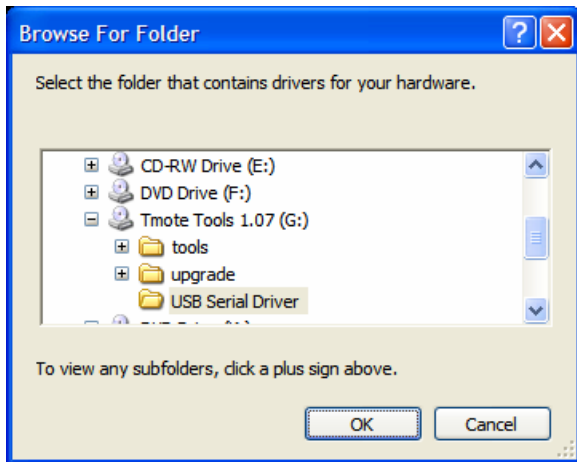
Note: If you experience any problems with your drivers, we recommend you uninstall all other “USB Serial” device drivers from your system.



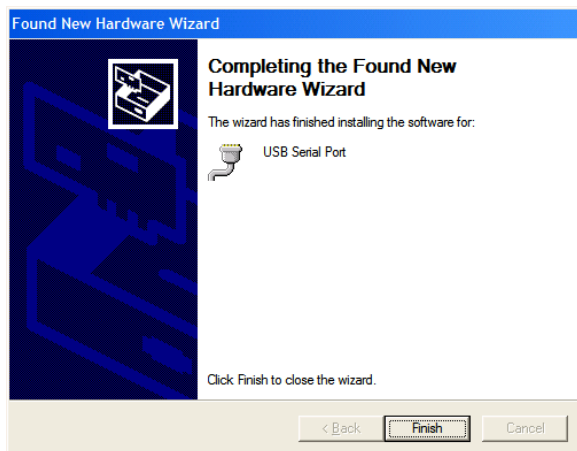
Select “Install from a list or specific location (Advanced)” then “Next”.



When prompted, select only “Include this location in the search” then click “Browse”.



Select the “USB Serial Driver” folder from the Tmote Tools CD, click “OK”, then “Next”.



Windows installs the driver files, completing the installation for Tmote as a “USB Serial Port”.



After installation, a second New Hardware window may appear for a “USB Serial Port”. At this point it is okay to select “Install the software automatically”.

Application Compile and Reprogram

Cygwin is used to compile and reprogram applications for Tmote.



To open a shell, invoke the Cygwin shortcut that the installer placed on your desktop.

```

Cygwin@tmoteiv ~
$

```

This is an initial Cygwin bash shell prompt. The install shield installs the TinyOS sources into /opt/tinyos-1.x/. Now we'll compile CntToLedsAndRfm, a program that increments a counter, displaying it to the Tmote LED's while sending a message of the count over the radio.

```

/opt/tinyos-1.x/apps/CntToLedsAndRfm
$ cd /opt/tinyos-1.x/apps/CntToLedsAndRfm/
Cygwin@tmoteiv /opt/tinyos-1.x/apps/CntToLedsAndRfm
$ make tmote
make -o build/telesh
compiling CntToLedsAndRfm to a telesh binary
gcc -o build/telesh/main.exe -Os -I../lib/Counters -ndisable-hwml -I/opt/tinyos-1.x/tos/lib/CC2420Radio -I/opt/tinyos-1.x/tos/./meta/ST232P -Wall -Wshadow -DBOARD=EP_IPS_ARM_GROUP=40x7d -Unesc=all -target=telesh -fnesc-cfile=build/telesh/app.c -board= -DIDENT_PROGRAM_NAME="CntToLedsAndRfm" -DIDENT_PROGRAM_NAME_BYTES="67,110,116,84,111,76,181,100,115,65,110,100,82,102,109,9" -DIDENT_USER_ID="Copy" -DIDENT_USER_ID_BYTES="67,111,114,121,0" -DIDENT_HOSTNAME="PRIMAL" -DIDENT_HOSTNAME_BYTES="80,82,73,72,65,76,0" -DIDENT_USER_HASH=0x080b87e4L -DIDENT_UNIX_TIME=0x42286bdl CntToLedsAndRfm.o -lm
C:/PROGRAM1/UCB/cygwin/opt/tinyos-1.x/tos/lib/CC2420Radio/CC2420RadioM.nc:115: warning: 'Send_sendDone' called asynchronously from 'sendFailed'
compiled CntToLedsAndRfm to build/telesh/main.exe
11666 bytes in ROM
375 bytes in RAM
msp430-objcopy --output-target=ihex build/telesh/main.exe build/telesh/main.ihex
writing IOS image
Cygwin@tmoteiv /opt/tinyos-1.x/apps/CntToLedsAndRfm
$

```

Compiling CntToLedsAndRfm is as easy as changing to its directory and running "make tmote".

```

cd /opt/tinyos-1.x/apps/CntToLedsAndRfm
make tmote

```

```

/opt/tinyos-1.x/apps/CntToLedsAndRfm
compiling CntToLedsAndRfm to a telesh binary
gcc -o build/telesh/main.exe -Os -I../lib/Counters -ndisable-hwml -I/opt/tinyos-1.x/tos/lib/CC2420Radio -I/opt/tinyos-1.x/tos/./meta/ST232P -Wall -Wshadow -DBOARD=EP_IPS_ARM_GROUP=40x7d -Unesc=all -target=telesh -fnesc-cfile=build/telesh/app.c -board= -DIDENT_PROGRAM_NAME="CntToLedsAndRfm" -DIDENT_PROGRAM_NAME_BYTES="67,110,116,84,111,76,181,100,115,65,110,100,82,102,109,9" -DIDENT_USER_ID="Copy" -DIDENT_USER_ID_BYTES="67,111,114,121,0" -DIDENT_HOSTNAME="PRIMAL" -DIDENT_HOSTNAME_BYTES="80,82,73,72,65,76,0" -DIDENT_USER_HASH=0x080b87e4L -DIDENT_UNIX_TIME=0x42286bdl CntToLedsAndRfm.o -lm
C:/PROGRAM1/UCB/cygwin/opt/tinyos-1.x/tos/lib/CC2420Radio/CC2420RadioM.nc:115: warning: 'Send_sendDone' called asynchronously from 'sendFailed'
compiled CntToLedsAndRfm to build/telesh/main.exe
11666 bytes in ROM
375 bytes in RAM
msp430-objcopy --output-target=ihex build/telesh/main.exe build/telesh/main.ihex
writing IOS image
Cygwin@tmoteiv /opt/tinyos-1.x/apps/CntToLedsAndRfm
$ motelist
Reference  ConnPort  Description
-----
M49V086   COM3         tmote sky
Cygwin@tmoteiv /opt/tinyos-1.x/apps/CntToLedsAndRfm
$

```

Use the motelist utility to show which COM ports actively have Tmote modules. Just type "motelist" at the command prompt:

```

motelist

```

It may take a few moments for a Tmote module to appear in the list the first time it is plugged into a system, because Windows must automatically install the drivers and allocate a COM port.



Low Power Wireless Sensor Module

```
~/opt/tinyos-1.x/apps/CntToLedsAndRfm
Reference  ConnPort  Description
M49UD86G  COM3       tmote sky

$ make tmote reinstall,1
~/opt/tinyos-1.x/tools/make/msp/set-mote-id --objcopy msp430-objcopy --objdump msp430-objdump --target ihex build/telesh/main.ihex build/telesh/main.ihex.out-1
Found note on COM3 (using bsl.auto)
installing telesh binary using bsl
msp430-bsp1 --telesh -c 2 -r -e -l -p build/telesh/main.ihex.out-1
msp430 Bootstrap Loader Version: 1.39-telesh-?
Mass Erase...
Transmit default password ...
Invoking BSL...
Transmit default password ...
Current bootstrap loader version: 1.61 (Device ID: f16c)
Changing baudrate to 38400 ...
Program ...
11698 bytes programmed.
Reset device ...
mv -f build/telesh/main.exe.out-1 build/telesh/main.ihex.out-1
~/opt/tinyos-1.x/apps/CntToLedsAndRfm
$
```

To install the application onto the Tmote module if only one Tmote is connected, enter the following command:

make tmote reinstall, 1

“reinstall,1” means “program the module with the already compiled binary image, set the network address of the module to 1”.

```
~/opt/tinyos-1.x/apps/CntToLedsAndRfm
$ motelist
Reference  ConnPort  Description
M49UD86G  COM3       tmote sky
M49UC0RD  COM4       tmote sky

~/opt/tinyos-1.x/apps/CntToLedsAndRfm
$ make tmote reinstall,1 bsl,3
~/opt/tinyos-1.x/tools/make/msp/set-mote-id --objcopy msp430-objcopy --objdump msp430-objdump --target ihex build/telesh/main.ihex build/telesh/main.ihex.out-1
installing telesh binary using bsl
msp430-bsp1 --telesh -c 3 -r -e -l -p build/telesh/main.ihex.out-1
msp430 Bootstrap Loader Version: 1.39-telesh-?
Mass Erase...
Transmit default password ...
Invoking BSL...
Transmit default password ...
Current bootstrap loader version: 1.61 (Device ID: f16c)
Changing baudrate to 38400 ...
Program ...
11698 bytes programmed.
Reset device ...
mv -f build/telesh/main.exe.out-1 build/telesh/main.ihex.out-1
~/opt/tinyos-1.x/apps/CntToLedsAndRfm
$
```

To install the application onto a Tmote module if more than one Tmote is connected, you must specify the serial port to install onto using the following command:

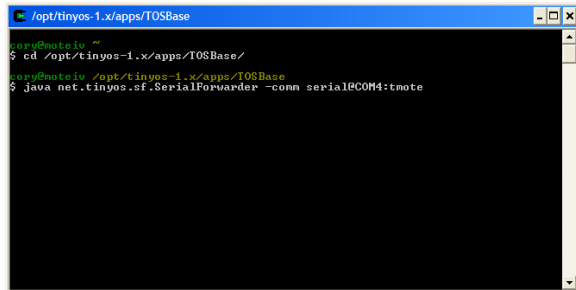
make tmote reinstall,1 bsl,3

“bsp,3” means “send the program using the Bootstrap Loader on COM4”. Note that the bsl programmer uses Linux-style indexing for the COM ports, so COM4 is specified as “3” for bsl.

You’ve now successfully compiled and installed an application onto Tmote!

Running Java Tools

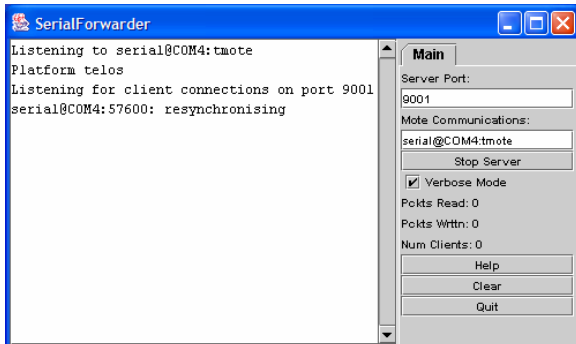
In addition to reprogramming, USB is used to receive data from Tmote into a PC. When a module is plugged into a PC, it can essentially act as a “Tmote module adapter”, bridging the PC and the Tmote networks. SerialForwarder is a Java tool that listens for TinyOS packets on a *serial* port and *forwards* them over a local TCP network socket. This allows more than one application to send and receive packets to the attached module. First plug in a Tmote module to your PC’s USB port.



Run SerialForwarder by entering the following command on a single line:

```
java net/tinyos.sf.SerialForwarder -comm  
serial@COM4:tmote
```

The full Java package specification for SerialForwarder is “net/tinyos.sf.SerialForwarder”. “-comm serial@COM4:tmote” tells SerialForwarder to communication using the serial protocol over COM4 with an attached a Tmote module.



This is the Java application window created by SerialForwarder. It allows you to stop and start forwarding packets, specify a new communication device, and displays useful statistics.



Low Power Wireless Sensor Module

Bringing It All Together

You now have the knowledge and experience to use all these tools together. To gather data from the Tmote network, connect a second Tmote module to a USB port. Change to the TOSBase directory in TinyOS (in the directory `/opt/tinyos-1.x/apps/TOSBase`). Compile with "make tmote" and install a TOSBase application on the second Tmote module ("make tmote reinstall"). The TOSBase application acts as a gateway, relaying Tmote packets to the PC and vice versa. Keep your first Tmote module running `CntToLedsAndRfm`, attach the Tmote module running TOSBase to a USB port, and run `SerialForwarder` on that port. Now run "java `net/tinyos.tools.Listen`", configure `SerialForwarder` for the virtual COM port assigned to the TOSBase Tmote, and see the raw data from the first mote arrive through the TOSBase mote and onto your PC. If you don't know the port Tmote is using, run `motelist` from any Cygwin command window. The "Packets Read" counter should now be increasing each time the first Tmote module increments its count on the LEDs. You have successfully set up a wireless network with Tmote and can move on to the TinyOS tutorial in `/opt/tinyos-1.x/doc/tutorial/` or start developing your own applications.

If you need help using the Tmote module, contact Moteiv Support at support@moteiv.com. You can also find help for programming in TinyOS on the TinyOS Help mailing list at tinuos-help@millennium.berkeley.edu.



Tmote/TinyOS Manual Install for Microsoft Windows

This document describes how to download and install the software and tools necessary to develop for Tmote using TinyOS. This document is also available on the web at <http://moteiv.com/support/install-windows.html>.

Last revised on March 4, 2005.

Cygwin

[Cygwin](#) [1] is a Linux-like environment in Windows. It provides the basic development environment for TinyOS and Tmote. Cygwin's [setup program](#) [2] downloads and installs necessary packages. It doesn't hurt to install every package, it just takes a little more space and time to download.

[1] <http://cygwin.com/>

[2] <http://cygwin.com/setup.exe>

Initial Environment

This install guide uses certain environment variables which are listed here with their defaults. Double check their values before using them in this guide. They must be defined using unix-style paths and not as windows paths like "c:/" or "c:\".

```
export JDKROOT=/cygdrive/c/j2sdk1.4.2_07
export MSPGCCROOT=/cygdrive/c/msp gcc
export TOSROOT=$HOME/tinyos-1.x
```

Java

[Java](#) [1] is used for the tools that interface between a mote and your PC. The TinyOS tools have been tested on the [Java 2 Platform, Standard Edition v 1.4.2 \(J2SE\)](#) [2].

After installing the JDK, ensure that JDKROOT is set for your installation, such as:

```
export JDKROOT=/cygdrive/c/j2sdk1.4.2_07
```

[1] <http://java.sun.com/>

[2] <http://java.sun.com/j2se/1.4.2/download.html>



JavaComm

[JavaComm](#) [1] is an additional package for Java that provides access to serial ports, required for tools like SerialForwarder. [JavaComm for Windows](#) [2] is available directly from Sun; [JavaComm for Linux](#) [3] requires more effort. Both require some command-line effort to correctly install, feel free to read the instructions, though these commands work nicely to get JavaComm installed for Windows:

```
unzip javacomm20-win32.zip
cp commapi/win32com.dll $JDKROOT/jre/bin/
cp commapi/comm.jar $JDKROOT/jre/lib/ext/
cp commapi/javax.comm.properties $JDKROOT/jre/lib/
```

- [1] <http://java.sun.com/products/javacomm/>
- [2] <http://java.sun.com/products/javacomm/downloads/index.html>
- [3] http://wass.homelinux.net/howtos/Comm_How-To.shtml

FTDI VirtualComm USB Drivers

[FTDI](#) [1] provides [Windows drivers](#) [2] to communicate with their USB-to-RS232 chips as a standard COM port; Linux drivers have been available as part of the kernel since 2.4.0. Use the drivers [Microsoft WHQL certified version currently dated 20 Apr 2004](#).

- [1] <http://www.ftdichip.com/>
- [2] <http://www.ftdichip.com/Drivers/FT232-FT245Drivers.htm>
- [3] <http://www.ftdichip.com/Drivers/FT232-FT245/VCP/Win/R9052154.zip>

TinyOS CVS Repository

The [TinyOS](#) [1] CVS repository is hosted by the [TinyOS SourceForge project](#) [2]. To get the latest, bleeding edge version of the code, enter the following commands at a shell prompt; just press Enter when prompted for a password:

```
cvs -d:pserver:anonymous@cvs.sf.net:/cvsroot/tinyos login
cvs -z3 -d:pserver:anonymous@cvs.sf.net:/cvsroot/tinyos co tinyos-1.x
```

After downloading the TinyOS 1.x source code, ensure that TOSROOT is set for your installation, such as:

```
export TOSROOT=$HOME/tinyos-1.x
```

- [1] <http://www.tinyos.net/>
- [2] <http://sourceforge.net/projects/tinyos/>



MSPGCC

[MSPGCC](#) [1] is the compiler used by TinyOS for the TI MSP430 platform. A patched version of this compiler its supporting tools is necessary for Tmote Rev B.

The easiest way to install MSPGCC for Cygwin is to use the MSPGCC RPM provided on the Tmote Tools CD found in the “upgrade” folder. This RPM installs the tools into /usr/local/mspgcc and requires a Cygwin installed or updated on or after December 4, 2004.

As an alternative, both Cygwin and Linux users can use the build-mspgcc script found in the tools/mspgcc folder on the Tmote Tools CD. The build script by default installs into /opt/msp430, though this can be easily changed by setting an environment variable before building. To use that script, copy the entire contents of the tools/mspgcc directory and its subdirectories containing source archives to a local working directory. Then as root run:

```
bash build-mspgcc install
```

If you don't have the Tmote Tools CD available, it can be found in the TinyOS CVS in the directory tinynos-1.x/tools/src/mspgcc/. You can run it directly inside that directory and it will download, compile, and install the patched tools for you.

If the build-mspgcc script doesn't seem to work, you can also follow the manual compilation instructions described in the ToolChain README in the same directory as build-mspgcc.

After installing MSPGCC, if you're following along this guide, set the MSPGCCROOT for your installation, such as:

```
export MSPGCCROOT=/opt/msp430
```

[1] <http://mspgcc.sourceforge.net/>

Patched MSP430 Assembler

The MSP430 assembler patched for TinyOS and nesC is now built as a part of the build-mspgcc script, instructions above.



MSP430-BSL for Tmote

msp430-bsl is the programmer, or "Boot Strap Loader", for the MSP430 developed by the MSPGCC project. Tmote requires a special version distributed with TinyOS to support its unique hardware design. Go to \$TOSROOT/tools/src/mspgcc-pybsl/ and follow the instructions detailed in README.msp430-bsl-tmote. For Tmote sky, msp430-bsl version 1.39-tmote-7 or later is required.

As an alternative to manually compiling msp430-bsl.exe, Moteiv provides a [precompiled version](#) [1] of msp430-bsl that you may download and install:

```
wget http://moteiv.com/support/msp430-bsl.zip
unzip -o -d $MSPGCCROOT/bin msp430-bsl.zip msp430-bsl.exe python23.dll \
lib/\*
```

[1] <http://moteiv.com/support/msp430-bsl.zip>

Motelist

Motelist is a command-line utility that displays which serial ports have Tmote motes attached. Run it with "motelist"; try "motelist -h" for additional options. To compile and install, enter the following command:

```
cd $TOSROOT/tools/src/motelist/ && make && make install
```

As an alternative to manually compiling motelist.exe, Moteiv provides a [precompiled version](#) [1] that you may download and install:

```
wget http://moteiv.com/support/motelist.zip
unzip -o -d /usr/local/bin motelist.zip motelist.exe
```

[1] <http://moteiv.com/support/motelist.zip>

TinyOS Scripts

Some TinyOS scripts, specifically set-mote-id, are necessary to the build process. Install these scripts by entering the following command:

```
cd $TOSROOT/tools/scripts/ && make install prefix=/usr/local
```



Environment

Now is a good time to fully prepare your environment for Tmote and TinyOS. Add these commands like these to your ~/.bashrc script. Note that the xxxROOT variables must match your installation and must be defined using unix-style paths.

```
# /usr/local/bin
export PATH="/usr/local/bin:$PATH"

# Java
export JDKROOT=/cygdrive/c/j2sdk1.4.2_05
export PATH="$JDKROOT/bin:$PATH"
# MSPGCC
export MSPGCCROOT=/opt/msp430
export PATH="$MSPGCCROOT/bin:$PATH"

# TinyOS
export TOSROOT=$HOME/tinyos-1.x
export TOSDIR=$TOSROOT/tos
export CLASSPATH="`$TOSROOT/tools/java/javapath`"
export MAKERULES=$TOSROOT/tools/make/Makerules
```

Before continuing to the final steps of this guide, ensure that these commands have been executed, possibly by logging back into your bash shell. Compiling in TinyOS requires TOSDIR be set, using the TinyOS Java tools requires that CLASSPATH be properly set, and building for Tmote requires that MAKERULES be set.

nesC

[nesC](#) [1] is the programming language used for TinyOS, and it requires its own front-end compiler. Download the [nesC 1.1.3 source archive](#) [2] from SourceForge

Then, to compile and install nesC, enter the follow command at the shell prompt:

```
./configure && make && make install
```

[1] <http://nesc.sourceforge.net/>

[2] <http://prdownloads.sourceforge.net/nesc/nesc-1.1.3.tar.gz>

TinyOS Java Tools

The TinyOS Java tools require the MIG utility provided by nesC to generate certain source files for processing messages. Compile the Java sources with the following commands:

```
cd $TOSROOT/tools/java; make; make
```




General Information

Document History

| Revision | Date | Notes |
|----------|------------|-----------------|
| 1.0 | 2005/03/04 | Initial Release |

Disclaimer

Moteiv Corporation believes the information contained herein is correct and accurate at the time of this printing. However, Moteiv Corporation reserves the right to make changes to this product without notice. Moteiv Corporation does not assume any responsibility for the use of the described product; neither does it convey any license under its patent rights, or the rights of others. This product is not designed for use in life support devices or any other system where malfunction can reasonably be expected to result in significant personal injury to the user. This product is not designed for critical systems where failure of the product to perform affects safety or effectiveness. Moteiv Corporation customers using or selling products for use in such applications do so at their own risk and agree to fully indemnify Moteiv Corporation for any damages resulting from improper use or sale.

As far as possible, major changes of product specifications and functionality, will be stated in product specific errata notes published at the Moteiv website. The latest updates are available from the Moteiv website at www.moteiv.com or by contacting Moteiv directly.



Low Power Wireless Sensor Module

Address Information

Web site: <http://www.moteiv.com>
E-mail: info@moteiv.com
Technical Support E-mail: support@moteiv.com
Phone Number: +1.510.965.1312
Fax Number: +1.510.295.2411

Headquarters

Moteiv Corporation
7224 View Ave
El Cerrito, CA 94530

© 2005 Moteiv Corporation