

Installing TinyOS on Windows Vista

Annie I-An Chen, 2009/1/25

This instruction is for the following hardware/software:

- Operating System: Windows Vista
- Programming Board: Crossbow MIB520
- Mote: Crossbow MPR2400 (MICAz)

The instructions are similar to that given on the following webpage, except for some crucial modifications for Windows Vista and MIB520:

<http://www.tinyos.net/tinyos-2.x/doc/html/install-tinyos.html>

Procedures for installing TinyOS on Windows Vista

1. Install Java SE Development Kit (JDK)
2. Install Cygwin
3. Rebase Cygwin
4. Install TinyOS-compatible Cygwin Package
5. Install compilers, tools and source tree
6. Set Environment Variables
7. Install Graphviz
8. Test it with a simple application!

1. Install Java SE Development Kit (JDK)

<http://java.sun.com/javase/downloads/index.jsp>

The version used by the author is JDK 6 Update 11.

2. Install Cygwin

<http://www.cygwin.com/setup.exe>

Download and install with default settings.

Make sure *python*, *rebaseall*, *rpm*, and your favorite text editor (e.g. *vim*, *emacs*) are selected.

[For Beginners] Some useful Linux commands:

cd: change directory.

For example,

```
$ cd c:/crossbow/cygwin/
```

changes the current directory to *C:\crossbow\cygwin*.

ls: list contents of current directory

For example,

```
$ cd c:/crossbow/cygwin/  
$ ls
```

prints a list of all the files and folders in `C:\crossbow\cygwin\`.

3. Rebase Cygwin

Please refer to `.\cygwin\usr\share\doc\Cygwin\rebase-2.4.4.README` for details.

Due to compatibility issues between Windows Vista and Cygwin, it is important to rebase cygwin before using it.

Execute `.\cygwin\bin\ash.exe` and type in the following command:

```
$ /bin/rebaseall
```

4. Install TinyOS-compatible Cygwin Package

<http://www.tinyos.net/dist-1.2.0/tools/windows/cygwin-1.2a.tgz>

Go to the directory where this file is saved, and unzip the file above by typing the following commands in the Cygwin shell:

```
$ tar zxvf cygwin-1.2a.tgz
```

In Windows Explorer, run `cygwin-installationfiles/setup.exe` under the directory to which the former file was unzipped. Use the following settings while installing:

- Choose “Install from local directory”
- Set root directory to where the current cygwin is installed
- Choose “Install for all users” and “Unix file type”
- Specify local package directory to where the former file was unzipped

After installation, you may need to rebase cygwin again, if errors occur.

5. Install compilers, tools and source tree

Atmel AVR Tools

<http://www.tinyos.net/dist-2.0.0/tools/windows/avr-binutils-2.15tinyos-3.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/avr-gcc-3.4.3-1.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/avr-libc-1.2.3-1.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/avarice-2.4-1.cygwin.i386.rpm>

<http://www.tinyos.net/dist-1.2.0/tools/windows/avr-insight-6.3-1.cygwin.i386.rpm>

TI MSP430 Tools

<http://www.tinyos.net/dist-2.0.0/tools/windows/msp430tools-base-0.1-20050607.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/msp430tools-python-tools-1.0-1.cygwin.noarch.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/msp430tools-binutils-2.16-20050607.cygwin.i386.rpm>

[m](#)

<http://www.tinyos.net/dist-2.0.0/tools/windows/msp430tools-gcc-3.2.3-20050607.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tools/windows/msp430tools-libc-20050308cvs-20050608.cygwin.i386.rpm>

[86.rpm](#)

TinyOS Tools

<http://www.tinyos.net/dist-1.2.0/tinyos/windows/nesc-1.2.8a-1.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tinyos/windows/tinyos-tools-1.2.4-2.cygwin.i386.rpm>

<http://www.tinyos.net/dist-2.0.0/tinyos/windows/tinyos-2.0.2-2.cygwin.noarch.rpm>

Download the files above and run each with the following command:

```
$ rpm -ivh --ignoreos <rpmname>
```

where *<rpmname >* is the file name **.rpm* .

6. Set Environment Variables

Type in the following commands:

```
$ export TOSROOT="<your_directory_cygwin>/opt/tinyos-2.x"
$ export TOSDIR="$TOSROOT/tos"
$ export
CLASSPATH="<your_directory_windows>\opt\tinyos-2.x\support\sdk\java
\tinyos.jar"
$ export MAKERULES="$TOSROOT/support/make/Makerules"
```

where:

- *<your_dircetory_cygwin>* is in forward slashes (cygwin-style), e.g. *c:/crossbow/cygwin*
- *<your_directory_windows>* is in backward slashes (windows-style), e.g. *C:\Crossbow\cygwin*
- The third through fifth line is actually the same line. (Do not change lines in the middle of this command.)

These variables need to be set every time you want to use it to compile nesC/TinyOS applications.

An alternative way is to add the four commands to *.bashrc* so that the variables are set automatically every time the cygwin shell starts up. This could be done using text editors like VIM:

```
$ vim .bashrc
```

[For Beginners] Some useful vim commands:

Please refer to http://www.cs.runet.edu/~mhtay/CPSC120/VIM_Editor_Commands.htm.

7. Install Graphviz

http://www.graphviz.org/Download_windows.php

Download the .msi file and run it with Windows Installer.

8. Test it with a simple application!

In cygwin shell, change directory to *.\cygwin\opt\tinyos-2.x\apps\Blink*. This is a program that

would make the LEDs on MPR2400 twinkle.

```
$ cd $TOSROOT/apps/blink
```

To compile this application, run the following command:

```
make micaz
```

Before installing it on the device, make sure the USB driver is properly installed. Also, check which serial port it is connected to. You can find out in *Control Panel>System>Device Manager>Hardware>Port*. The two virtual serial ports for MIB520 are com_x and $com_{(x+1)}$; com_x is for Mote programming and $com_{(x+1)}$ is for Mote communication.

The following command compiles the application and installs it onto the Mote:

```
make micaz install mib510,/dev/ttyS<x-1>
```

where $\langle x-1 \rangle$ depends on the COM port it is connected to.

Two things are different from that in the MoteWorks Getting Started Guide:

- Although we are using MIB520, here we need to specify it as MIB510 instead, because TinyOS does not support MIB520 yet.
- TinyOS advises us to use $/dev/ttyS\langle x-1 \rangle$ instead of $com\langle x \rangle$. If you use $com\langle x \rangle$, it displays the message “Please use $/dev/ttyS\langle x-1 \rangle$ rather than $com\langle x \rangle$ (the latter often doesn’t work)”.

The following command installs the application onto the Mote without compiling:

```
make micaz reinstall mib510,/dev/ttyS<x-1>
```

While installing, make sure that

- the programming board is plugged in the specified USB port
- the Mote is mounted correctly on the programming board (if so, the LEDs on the board should reflect the LEDs on the Mote)
- the power on the Mote is turned off (otherwise there is a risk of damaging them)

After installing it on the Mote, you should be able to see the LEDs flashing in the following

pattern:

