

## Running GGI\_TLM on Windows 10.

GGI\_TLM may be run on windows 10 machines by using cygwin. Cygwin creates a unix type environment in which GGI\_TLM can be compiled and run. This document describes the installation of paraview (used for result visualisation), cygwin and the download, setup and compilation of GGI\_TLM. This document should be sufficient to get the system running and give a brief check that it is working correctly. The user is referred to the documentation which comes along with the installation (user guide, theory manual, test case documentation) for further information.

### ***Paraview installation***

Install paraview for windows from the website:

[www.paraview.org](http://www.paraview.org)

navigate to the downloads page and select a suitable version (e.g. paraview-4.0.1-Windows-64bit.exe which works fine for me on a Windows 10 machine... Some later versions have problems on my machine)  
run the .exe file and follow the installation instructions.

### ***Cygwin installation***

Go to the website [www.cygwin.com](http://www.cygwin.com)

Under 'Installing Cygwin', click setup-x86\_64.exe

Download the cygwin setup for your machine (32 or 64 bit as appropriate) i.e. click setup-x86.exe or setup-x86\_64.exe as appropriate (the following notes are based on 64bit installation but I guess it shouldn't be too different for 32 bit.)

Save the file

Double click on the file to run it. (i.e. run the cygwin setup executable)

Click next at the copyright page

Choose a download source:

(e.g. Install from Internet)

Then click next

Choose Root directory:

(e.g. C:\cygwin64)

Select Install for all users

Then click next

Select Local Package Directory

(e.g. C:\Users\eezcjs\Downloads)

Then click next

Select Your Internet Connection as appropriate

(e.g. Direct Connection)

Then click next

Choose a download Site

(e.g. cygwin.mirror.uk.sargasso.net works fine in the UK)

Then click next

Select Packages (i.e. what to install) Default is a minimal install and we need some extra stuff so add the following (by clicking where it says 'default' to change to 'install'):

Devel

X11

In the following click the + sign to expand the top level list (e.g. Editors) , find the package in the list (e.g. nedit), then click where it says 'skip' this then changes to the version number to be installed (I have added the versions used to create a working version in yellow below).

Editors->nedit 5.5-6

Graphics->gnuplot 5.2.7-1

Libs->libopenmpi-devel 3.1.1-2

Libs->libopenmpi-hwloc 1.10.4-1

Interpreters->Lua

Interpreters->Lua5.1

Math->ngspice (note for ngspice click the src box to get the source code)

Then click next

The Cygwin setup proceeds – this takes quite a long time...

Click Finish

An icon 'Cygwin64 Terminal' appears on the desktop

Run Xlaunch from the programs menu

(Programs->Cygwin-X->Xlaunch)

Select Display Settings:

Select 'Multiple windows'

Click next

Select how to start clients

Click 'start no client'

Click next

Extra settings

Tick clipboard and Native OpenGL

Click next

Configuration complete

Click finish.

At this stage running gnuplot gives the error 'C:/cygwin64/bin/gnuplot.exe: error while loading shared libraries: cyglua-5.2.dll: cannot open shared object file: No such file or directory' In order to resolve this, re-run the setup-x86\_64.exe from the Cygwin webpage. Go through to the select packages stage and pick

Start a cygwin terminal by clicking the icon on desktop.

Find the name of your cygwin home directory with the command

**pwd**

(The result will be something like /home/user)

## Ngspice:

The Cygwin installation should have included the Ngspice source code. This may be found in the directory **/usr/src**

**cd /usr/src/ngspice-29-1.src**

**gunzip ngspice-29.tar.gz**

**tar -xvf ngspice-29.tar**

**cd ngspice-29**

**mkdir ngspice\_GGI\_TLM**

**cd ngspice\_GGI\_TLM**

**../configure --with-ngshared --enable-xspice -enable-cider**

**make install**

## **GGI\_TLM**

Goto the website

[www.github.com/ggiemr/GGI\\_TLM](http://www.github.com/ggiemr/GGI_TLM)

click 'download ZIP' button on the right hand side.

save (or copy) the file (GGI\_TLM-master.zip) to the cygwin home directory (From windows it will look something like C:\cygwin64\home\user)

Start a cygwin terminal (icon on desktop)

Look at the files in your home directory with the command

**ls**

This should include the file GGI\_TLM-master.zip

Expand the zip file with the command

**unzip GGI\_TLM-master.zip**

You should now have the following directory in you home directory (use **ls**)  
GGI\_TLM-master

In your home directory, edit the .bash\_profile file with the command:

**nedit .bash\_profile &**

(note: following a command with & runs the command in the background and returns control to the terminal. If you get an error: 'Error: Can't open display' then use the following command: **export DISPLAY=:0** and try nedit again.)

add the following lines at the end of the .bash\_profile file:

**PATH="\$HOME}/GGI\_TLM-master/GGI\_TLM/bin:.\$PATH"**

**export DISPLAY=:0**

Then save the file and exit nedit.

(this only needs to be done once when you set up the system, once you have edited the file, exit the cygwin terminal and restart it)

The next stage is to set things up to make GGI\_TLM. First go into the main GGI\_TLM directory

**cd GGI\_TLM-master/GGI\_TLM**

edit the file make\_configuration\_data with

**nedit make\_configuration\_data &**

and set the name of the fortran compiler correctly for cygwin

**FC=mpif90**

Also set the MPI\_INCLUDE directory correctly for cygwin

**MPI\_INCLUDE=/usr/include**

The compilation flags should not need to be changed

edit the file make\_run\_configuration\_data with

**nedit run\_configuration\_data &**

and set the MPI\_RUN\_COMMAND for cygwin

**MPI\_RUN\_COMMAND="mpirun -np "**

The SEQ\_RUN\_COMMAND should be "" but this shouldn't require changing.

Go into the source code directory

**cd SRC**

and make both sequential and mpi versions of GGI\_TLM with the commands:

**make clean**

## **make all**

alternatively if compilation with Ngspice is required, before compiling GGI\_TLM with Ngspice you will need to edit the file **SRC/Makefile\_GGI\_TLM** to enable Cygwin to find the ngspice library. The relevant part of the file is shown below with the editing completed for Cygwin compilation.

**# the following are required for the ngspice link**

```
ifeq ($(NGSPICE),TRUE)
# for linux use the following two lines:
#     export LD_LIBRARY_PATH=/lib:/usr/lib:/usr/local/lib
#     NGSPICE_LINK_LIBS= -lngspice -lpthread -ldl
# for cygwin, un-comment the following line and comment out the above two lines:
#     NGSPICE_LINK_LIBS= -L/usr/local/lib/ -lngspice -lpthread -ldl
#
#     MAKE_NGSPICE_LINK=MAKE_ngspice_link
#     NGSPICE_MODE=INCLUDE_NGSPICE
else
#     NGSPICE_MODE=NO_NGSPICE
endif
```

Then make GGI\_TLM with the command:

**make NGSPICE=TRUE**

This process may take a few minutes...

Once the system is made correctly you can try to run some of the test cases.

Go into the TEST\_DATA directory

**cd GGI\_TLM-master/GGI\_TLM/TEST\_DATA**

run the DIPOLE test case for example with

**run\_automatic\_test run\_seq DIPOLE**

plot the results of the simulations against the reference results to check everything is OK with

**run\_automatic\_test plot\_ref DIPOLE**

A gnuplot window should appear with plots (hit enter in the terminal window to step through the sequence) of time domain electric field, time domain current at the dipole feed point and dipole impedance as a function of frequency.

This test case produces some results which can be viewed with paraview. Start paraview and open files in the DIPOLE directory (the path will be something like  
C:\cygwin64\home\user\GGI\_TLM-master\GGI\_TLM\TEST\_DATA\DIPOLE)

For example the file `E_field_13MHz.frame...vtk` shows an animation of the vector  $E$  field at 13MHz on a plane close to the dipole antenna. Other files are available with the dipole geometry, mesh, outer boundary etc. These are described in the documentation.