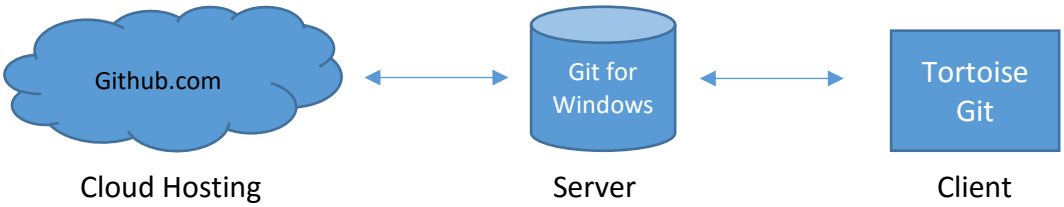
Getting Started with Github, Tortoise Git and LabVIEW

Getting started with Git and Installing all components:



Open a Github account and explore Github with very basic text files.

Go to Github.com and select the free option for public files. (private repositories are not paid only)

Add a few text files to your repository.

(Optional) Complete this short tutuorial: https://guides.github.com/activities/hello-world/

(Optional) Open a second Github account, or work with a buddy to experiment with multiple branches and merging Pull Requests with text files.

Install Git for Windows and TortoiseGit

Tutorials:

https://help.github.com/articles/git-and-github-learning-resources/

https://www.inboundnow.com/getting-started-tortoisegit-github/

Software downloads:

Git for Windows is here:

http://www.git-scm.com/

TortoiseGit is here:

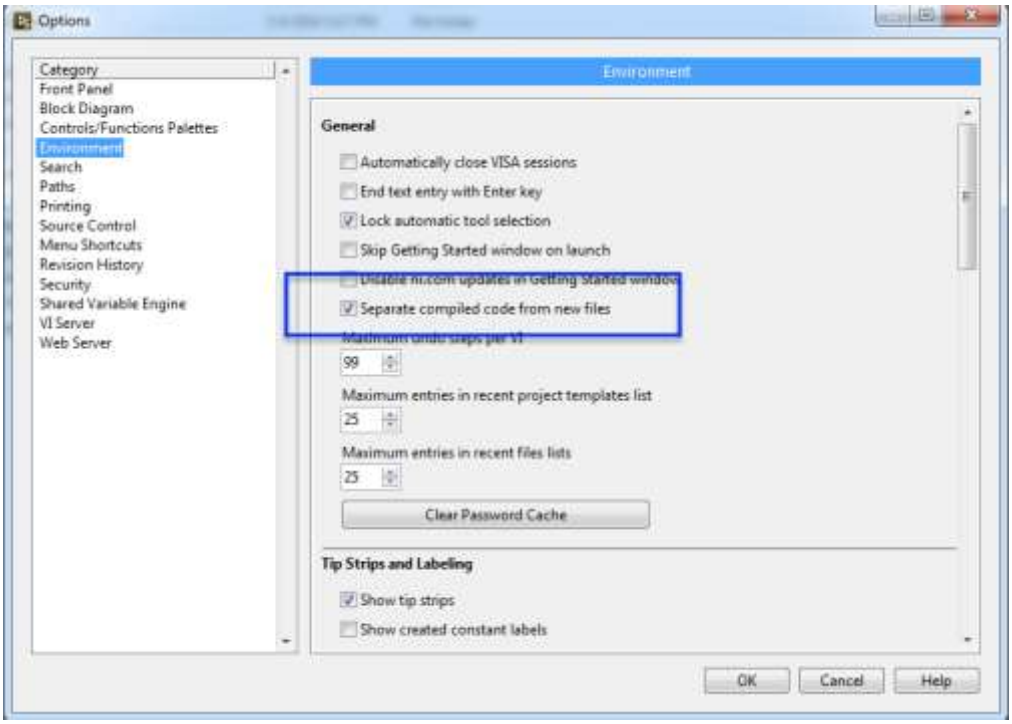
https://tortoisegit.org

Follow these instructions for setting SSH keys:

https://help.github.com/articles/generating-an-ssh-key/

(Note that when creating an ssh key, Windows interprets the “public” extension of .pub as the Microsoft Publisher extension)

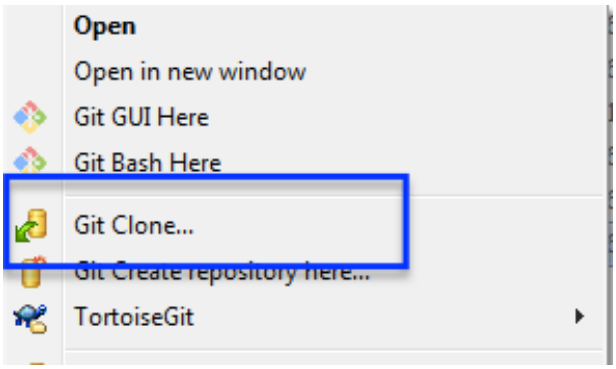
!!! Make sure you have “source only” enabled



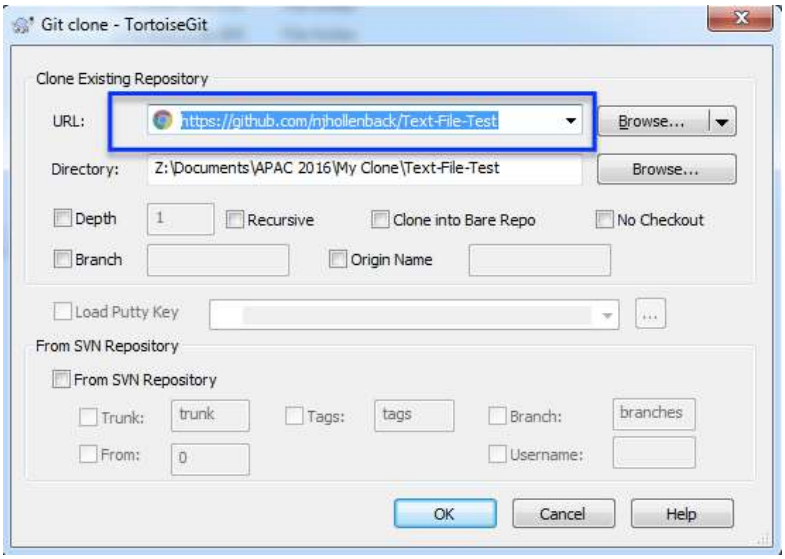
Setting up your clone of the repository that you created on Github:

Clone the master.

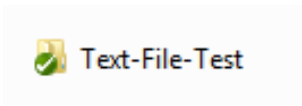
From your browser, navigate to your repo on Github
Go to windows explorer and create a new folder.
Right click on the folder and select “Git Clone”:



You should see the url for your Github repository. If not, copy it here. The select “OK”



You should now see a folder with your repo:

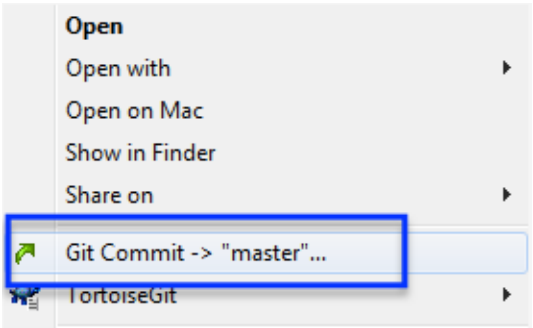


You will need to reboot in order to see the icon overlays. If you already have Tortoise SVN installed (or other Windows file explore extension with icon overlays such as Dropbox), you may not see the icon overlays for everything in the folder after you reboot.

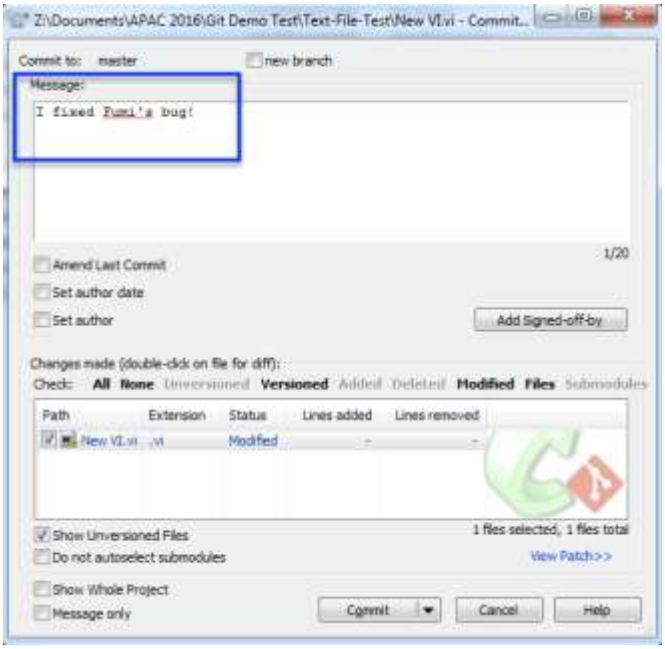
Make modifications and commit to your local clone of the repository

Open a file and make changes. Save changes.

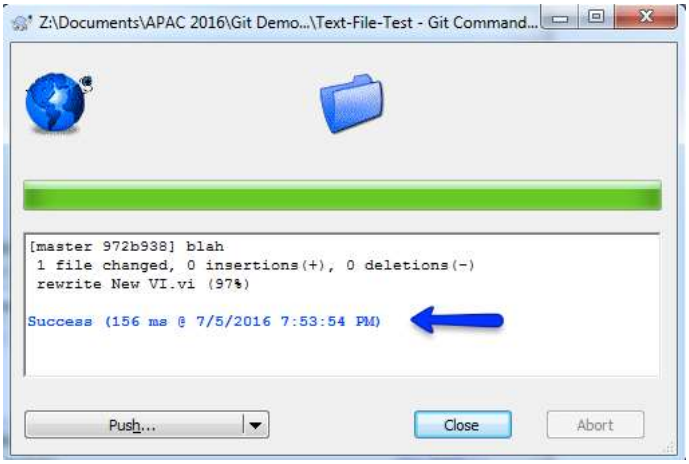
Then “Commit” to your local clone of the repository by right-clicking on the file or the main folder containing the repository



Enter your meaningful comments. Then select “Commit”



You will see the “flipping” turtle and then you should see “success”!
DO NOT select “Push” after the commit at this time.
Select “Close”



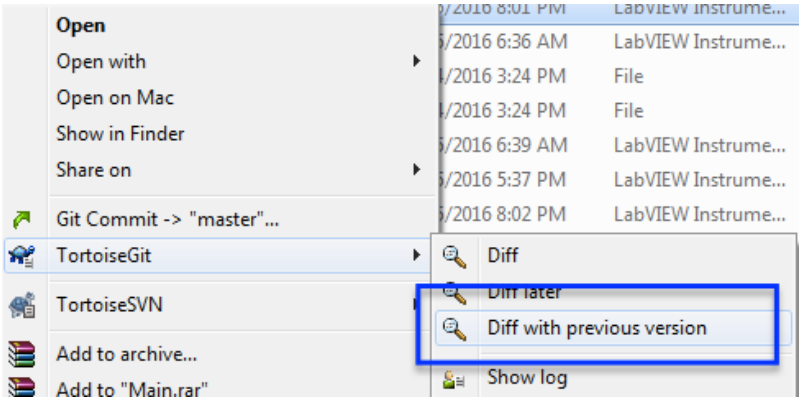
For those of you who have used TortoiseSVN, this will feel exactly like the work flow you are accustomed to. At this point we have not pushed your changes on your laptop to the cloud on Github. Next you will install the LabVIEW Graphical Diffing tool

Set up Tortoise Git for Graphical Diff and do a “Diff”

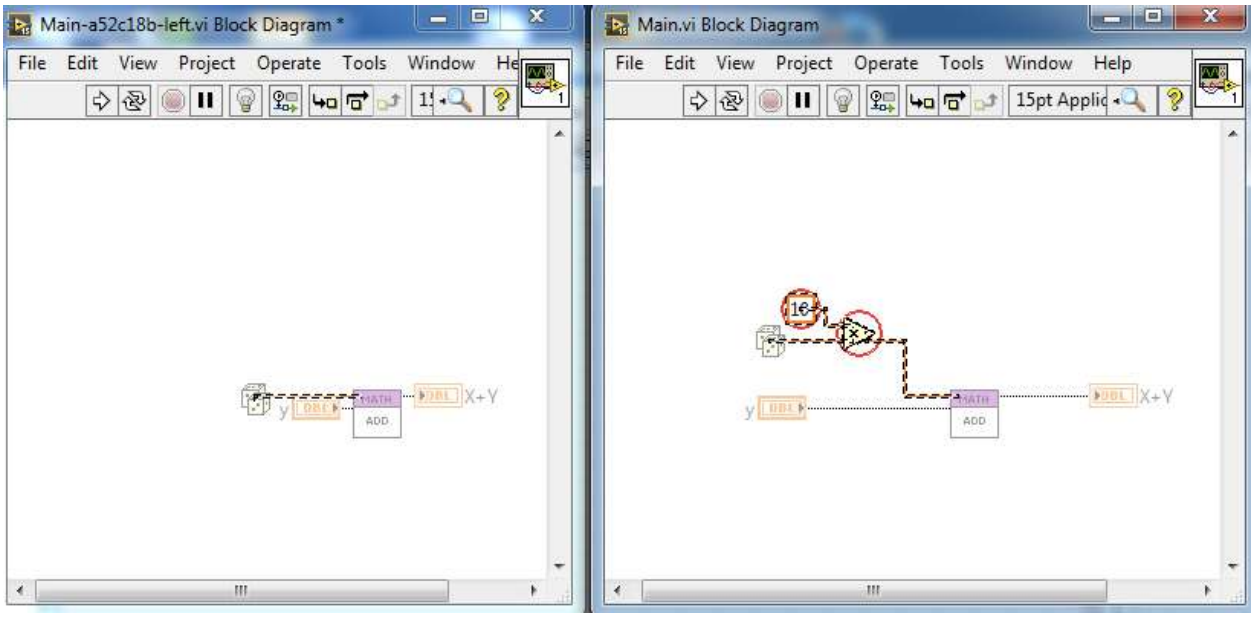
Follow the instructions here:

https://decibel.ni.com/content/docs/DOC-2936

Make a change to a VI and save.
Right click on the original VI and select “Diff with Previous”

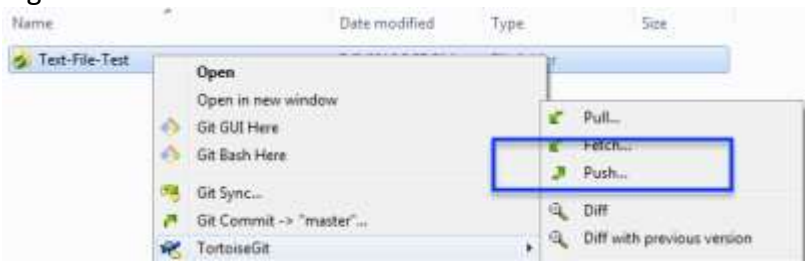


Observe the results in the LabVIEW Diff Tool:

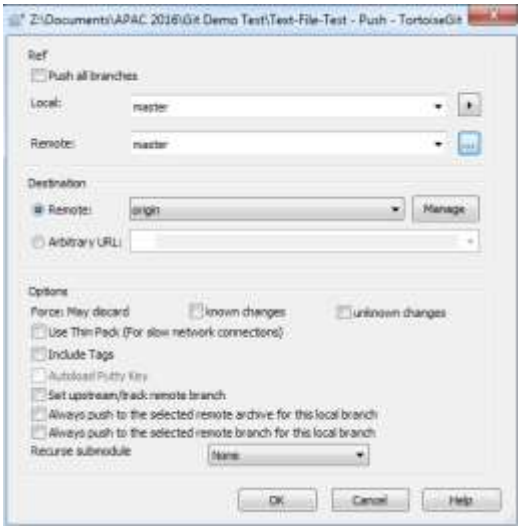


Push Your Changes to Github

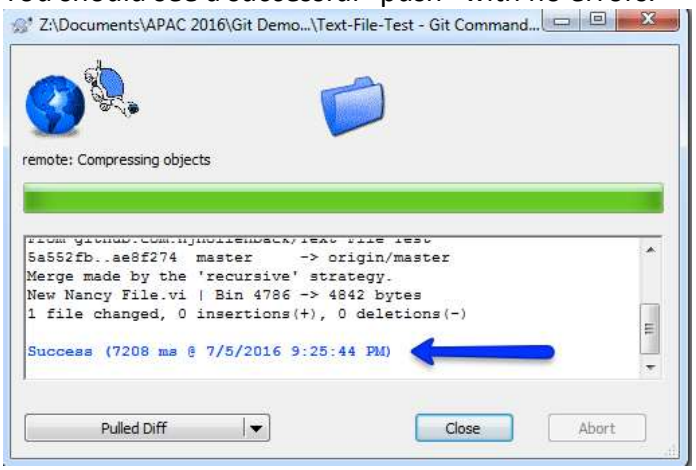
Right click on the folder and select “TortoiseGit >> Push”



(Right now we are just working on of the “master” repository on the remote Github cloud service. In the future, you may push to a branch.)

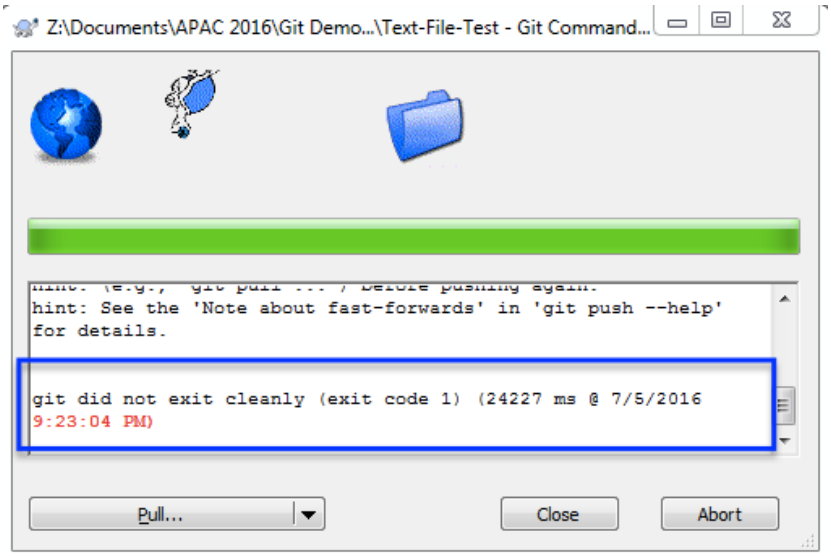


You should see a successful “push” with no errors.



Confirm by creating a new folder and a new clone from the repository and observer that your changes are in the new clone!!!

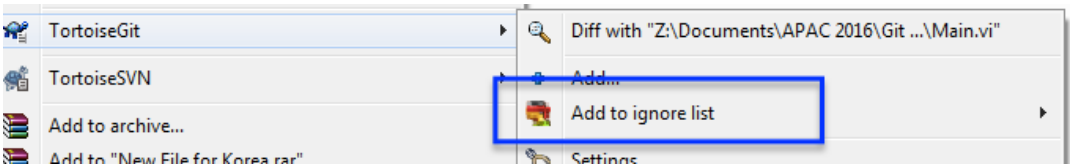
If you see the following, you may need to do a pull before the push



Other tasks to explore

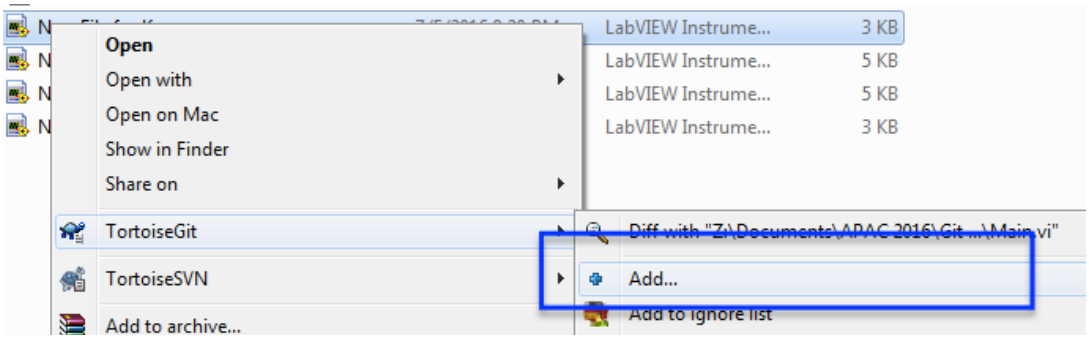
Remember to exclude the *.aliases and the *.lvlps.

Go to “TortoiseGit >> Add to Ignore List



Add a file to the repository and do a commit and push. Observe the results in Github.

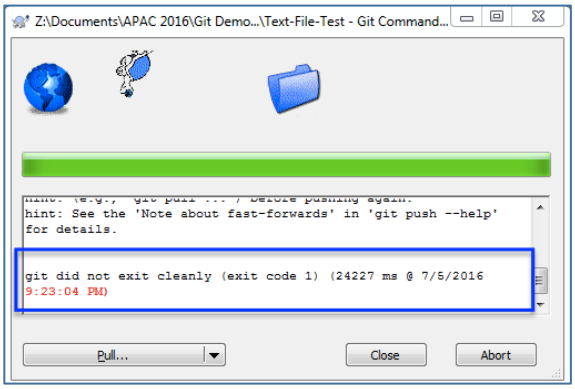
Remember to Add the file by right clicking and selecting “TortoiseGit >> Add”



Then commit and push and observe the new file in Github

Team with someone else. Each person will clone the same repo, make changes commit, and push.

Note: When you get the error on the push, do a pull and then a push.



Explore the log file by selecting “TortoiseGit >> Show log”