# Setup

We will be using the following tools for these labs:

* Git revision control
* UNIX make utility
* Vivado 2014.2

Following are instructions to install and set up Git and make, and to configure Vivado to run from the Windows command shell.

## Install GitHub for Windows

Git will be used as the revision control tool for these labs. It is free and one of the most widely-used tools and GitHub also serves as the revision control platform for the Vivado Tcl Store.

### GitHub Info and Installation

See the Windows GitHub introduction page here:

[https://windows.GitHub.com](https://windows.github.com)

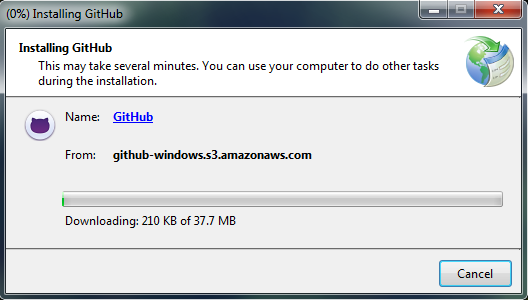
From there, click on

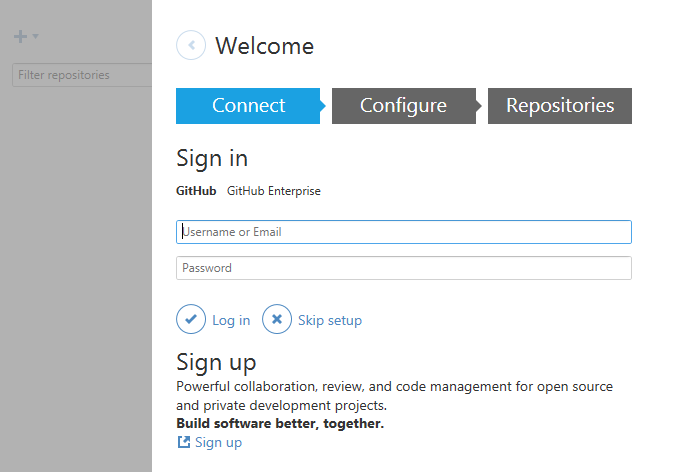


to download the setup program, or go to this URL:

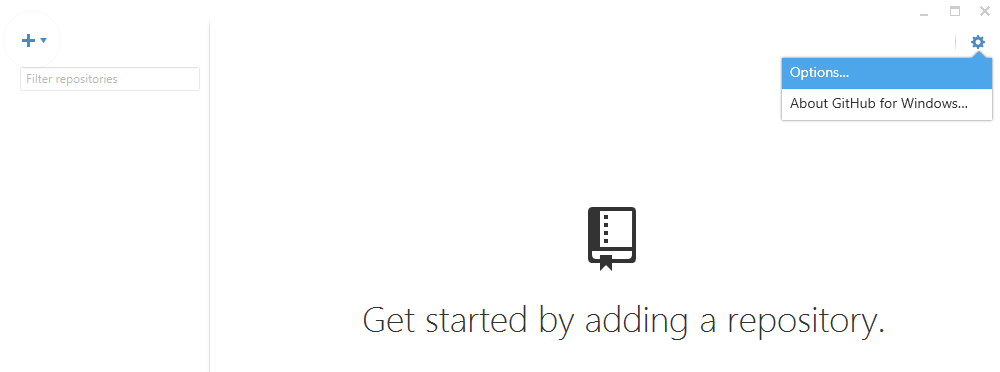
[https://GitHub-windows.s3.amazonaws.com/GitHubSetup.exe](https://github-windows.s3.amazonaws.com/GitHubSetup.exe)

Once GitHubSetup.exe is downloaded, run it to install GitHub.



Once the installation is complete you will see the Welcome page. Here you can **Log in** if you already have a GitHub account, or click the **Sign up** link at the bottom to create a new account. Or you may just continue without logging in by clicking on **Skip setup**. 

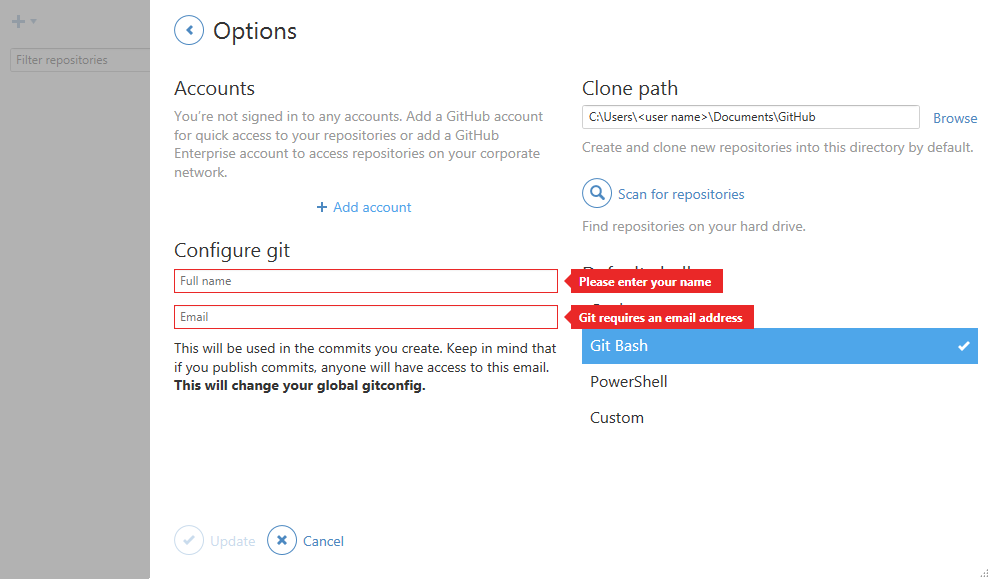
The next page shows a start page where no repositories have been set up, which should be typical for most. We will skip adding repositories for now and instead right-click on the blue Settings icon in the upper right corner and choose **Options…**



Here we will configure the following:

1. In the lower right, choose Git Bash as the default shell instead of PowerShell.
2. Enter your name and email which Git uses to track changes.

Once finished, click on the Update button at the bottom to save changes.



Now you may exit the GitHub GUI. We will be using the Git Shell for revision control.



To test Git, double-click the Git Shell icon to launch the Git Shell. It should open a command shell with a bash prompt ($). Run git status:

$ git status

# On branch master

#

# Initial commit

#

nothing to commit (create/copy files and use "git add" to track)

If you encounter errors or other dubious output please ask your instructor for assistance.

## Install make for Windows

We will also use the **make** utility to generate Vivado projects based on source files under revision control. The make utility reads Makefiles which contain sets of rules (commands) to build a specific target (output file). If you are unfamiliar with make and Makefiles, the following link serves as an introduction and reference:

<https://www.gnu.org/software/make/manual/html_node/Makefiles.html>

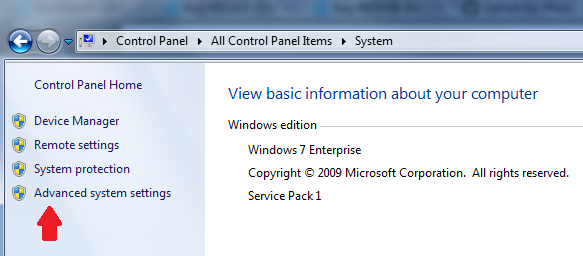
If you do not have the make utility installed on your system, you can download the GNU make for Windows at:

<http://gnuwin32.sourceforge.net/packages/make.htm>

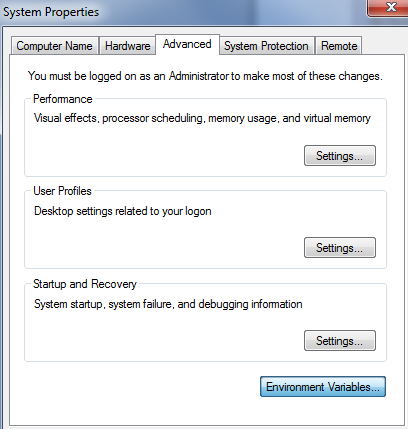
Under downloads, choose the “Complete package, except sources” The setup program installs make in the directory:

C:\Program Files (x86)\GnuWin32\bin

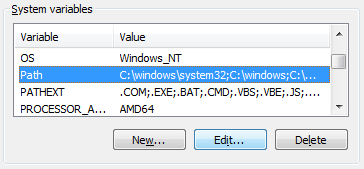
Add this path to your Windows path variable so that you can run make without specifying the full path. In the Control Panel, choose System and Advanced system settings:



Then click **Environment Variables** at the bottom:

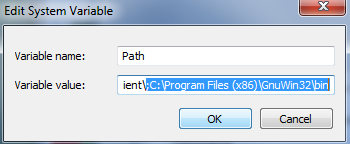


Select the Path System variable and click Edit:

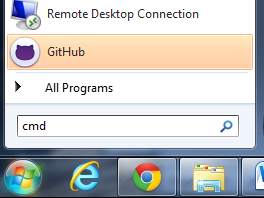


Copy and paste the following at the end of the Path value (note the leading semicolon to separate the path value from the existing Path:

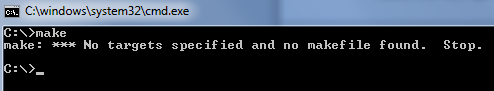
;C:\Program Files (x86)\GnuWin32\bin



Click OK to save changes. To test if make is installed successfully, open a Windows command shell. From the start button, enter **cmd** in the search bar and hit Enter.



In the command shell, run **make**. It should fail with the following message:



If you get a different message or experience some other difficulty, ask your instructor for assistance.

## Lab Files

## Setting up Vivado

If vivado.exe version 2014.2 is not in your Path, you can configure Vivado when you open a Windows

### Open Windows command shell and configure env variables

CALL c:\Xilinx\Vivado\2014.2\settings64.bat

TODO – set up PATH variable for make to msys in vivado\_hls – below only works in bash

export PATH=/c/Xilinx/Vivado\_HLS/$VER/msys/bin:$PATH

### Make sure it’s working

Git config

make

vivado -help

vivado\_hls -h

xsdk.bat

## Directory Structure

## Source Files

### HDL

### Scripts

### XDC Constraints

### Simulation Testbenches

### CPP

# RTL Projects

# IP Instantiated with RTL

# IPI Integrator

# Packaged Custom RTL IP with IPI

# System Generator Projects

# HLS

# Other - TBD

## DCP

## EDK/XPS

# Appendix A: Solutions