1. To Create Local Repository
   1. Create directory for repository called <RepoName>
   2. Place all files and directory for that repository in <RepoName>
   3. Initialize the git repository origin to be named “main”
      1. git init -b main
   4. Add all files to prepare to commit that are in the current directory
      1. git add .
   5. Commit Files
      1. git commit -m "Intial Commit on ##/##/##"’
2. Push files from local to remote repo
   1. Assure the repo that the git files go to correct Repo called <RepoName>
      1. git remote add origin [https://github.com/<topLevelRepoUserName>/<RepoName>.git](https://github.com/%3ctopLevelRepoUserName%3e/%3cRepoName%3e.git)
         1. i.e. <topLevelRepoUserName> = neurio
         2. i.e. <RepoName> = PwrMicro\_PSIM\_Model
   2. Check the remote repo again
      1. git remote -v
   3. git push -u origin master
      1. NOTE on the “-u” :
         1. -u ensures For every branch that is up to date or successfully pushed, add upstream (tracking) reference
3. Create a Branch from main branch on a local repo, make changes then merge
   1. git checkout -b <BRANCHNAME>
   2. make changes to files
   3. git add .
   4. git commit -m “This is a commit”
   5. git checkout main
   6. git merge <BRANCHNAME>
4. Create branch
   1. git branch <branchName>
5. Rename Branch
   1. git branch -m <newname>
6. List branches in local repo
   1. git branch
7. Get remote branches
   1. git branch –remotes
8. Initialize a local branch after git clone
   1. git branch -t my\_local\_branch origin/remote\_branch
      1. After cloning, this will create local branch the tracks remote branch
9. Get local and remote branches
   1. git branch--all
10. Create a new local branch
    1. git checkout -b|-B <new-branch>
11. Delete Branch
    1. git branch --delete <branchName>
12. List diff in branch for local repo
    1. git diff <BRANCH1> <BRANCH1>
13. Show what will happen when you fetch from remote repo
    1. git fetch –dry-run
14. Fetch from remote repo
    1. git fetch [https://github.com/<topLevelRepoName>/<RepoName>.git](https://github.com/%3ctopLevelRepoName%3e/%3cRepoName%3e.git)
       1. i.e. <topLevelRepoUserName> = neurio
       2. i.e. <RepoName> = PwrMicro\_PSIM\_Model
15. Clone a Remote Repo into a local directory
    1. git clone [https://github.com/<topLevelRepoName>/<RepoName>.git](https://github.com/%3ctopLevelRepoName%3e/%3cRepoName%3e.git)
       1. i.e. <topLevelRepoUserName> = neurio
       2. i.e. <RepoName> = PwrMicro\_PSIM\_Model
16. Get Help on git command
    1. git help <COMMAND>
17. Get status on current git repo
    1. git status
18. Get log of changes made to git repo
    1. git log --graph --decorate
19. Start local repo. Create remote repo using github.com. If you commit a readme file then do the following:
    1. git fetch
    2. git pull origin master --allow-unrelated-histories
       1. This is necessary because you will receive an error message that the remote and local have unrelated histories.
20. Push all branches from a local repo to remote repo
    1. git push --all [https://github.com/<topLevelRepoName>/<RepoName>.git](https://github.com/%3ctopLevelRepoName%3e/%3cRepoName%3e.git)
21. Restore a files from a previous commit
    1. git restore --source=<commit hash> file\_name
22. Show commits for a file
    1. git show <commit hash>:filename
23. Overview Schematic
24. 