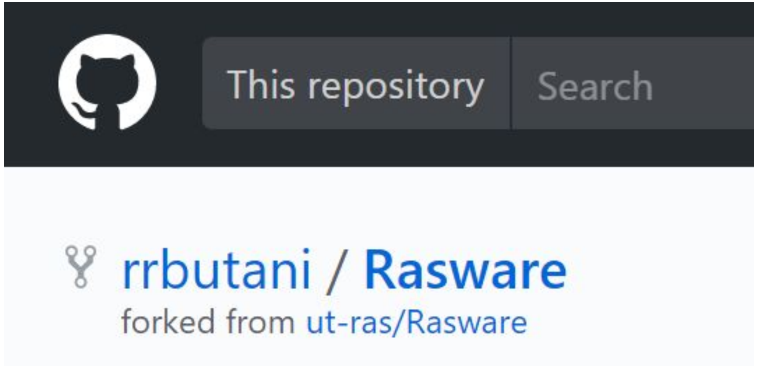
LEARN ME SOME GIT
and
RASLib

What You Should Already Have

- A computer with RASware installed on it
- A shared git fork



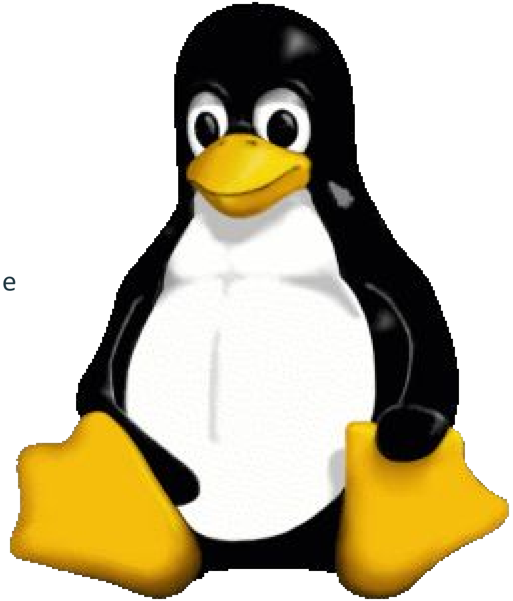
rrbutani / Rasware
forked from ut-ras/Rasware

What is Terminal and why should I use it?

Terminal - The terminal is an interface in which you can type and execute text based commands. It can be much faster to complete some tasks using a Terminal than with graphical applications and menus. Another benefit is allowing access to many more commands and scripts. Bash is a scripting language/interpreter that lets you interface with the terminal.

How To Use Bash

- Super simple git guide. - http://rogerdudler.github.io/git-guide/
- Most useful commands
    - ls - lists all files in your current directory
    - cd - changes to home directory
    - cd “path” or cd “name” - changes to a specified directory (folder)
    - mkdir - makes a new directory (folder)
    - touch “name” - create a new file
    - apt search “name” - searches through things to install with a specified name
    - apt install “name” - installs specified thing
    - rm “name” - removes specified thing
    - rmdir “name” - removes specified file
    - mv - move or rename
    - If in doubt, add info before a command to find out how it’s used
    - telnet towel.blinkenlights.nl - StarWars - A new hope
    - Tab to autocomplete
    - xdg-open . - open current directory in file explorer
    - xdg-open “file name” - open file using the default application
- LMGTFY



Version Control: Git

- Some useful Git Guides:
  - https://agripongit.vincenttunru.com/ (http://bit.ly/2jQoY8h)
  - http://rogerdudler.github.io/git-guide/ (http://bit.ly/MgjmhR)
- Why is Version Control Important?
- How does Git work?
- How do I use Git?
- What’s a GitHub?

Some Useful Git Commands:

- `git status`
- `git fetch // git pull`
- `git add <file/files>`
- `git diff <file>`
- `git checkout <file/branch>`
- `git commit -m <message>`
- `git branch`
- `git push`

Git/Bash Demo

1. Go to your RASWare folder
2. Create a file called HelloWorld.c
3. Open the file in a text editor (atom/subl/nano)
4. Put this in your the file and save it:

```
#include<stdio.h>
int main(void) {
    printf(“Hello World!\n”);
}
```

5. Compile it using - “gcc -o HelloWorld HelloWorld.c”
6. Test it using - “./HelloWorld”
7. Git add the file
8. Git commit
9. Git push

git config --global user.name “<your name>”

git config --global user.email “<your email>”

:wq to quit vi
^x and y to quit nano

How to Read Libraries?

RASLIB

- Library
  - A library is collection of code that you can use so you don't have to write your robots code from scratch
- Header File (a .h file such as motor.h)
  - A Header file specifies what commands a library contains and how to properly use each one
  - Example of what's in a header
  - `* Sets a motor speed`
  - `* @param mtr Pointer to an initialized tMotor, returned by InitializeMotor`
  - `* @param speed Float on range [-1, 1] where -1 means maximum backward speed and 1 means maximum forward speed`
  - `void SetMotor(tMotor *mtr, float speed);`
  - This tells you how to call the function setMotor(MotorName,speed)
- Check the RASWare Wiki for (nicely formatted!) versions of the RASLib header files