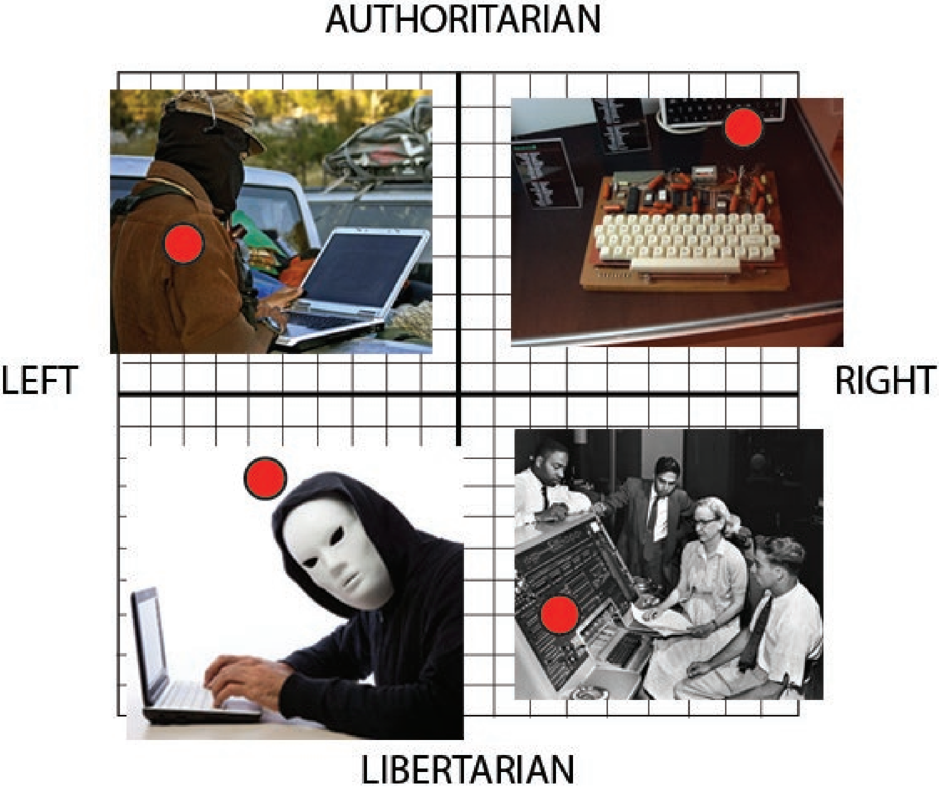
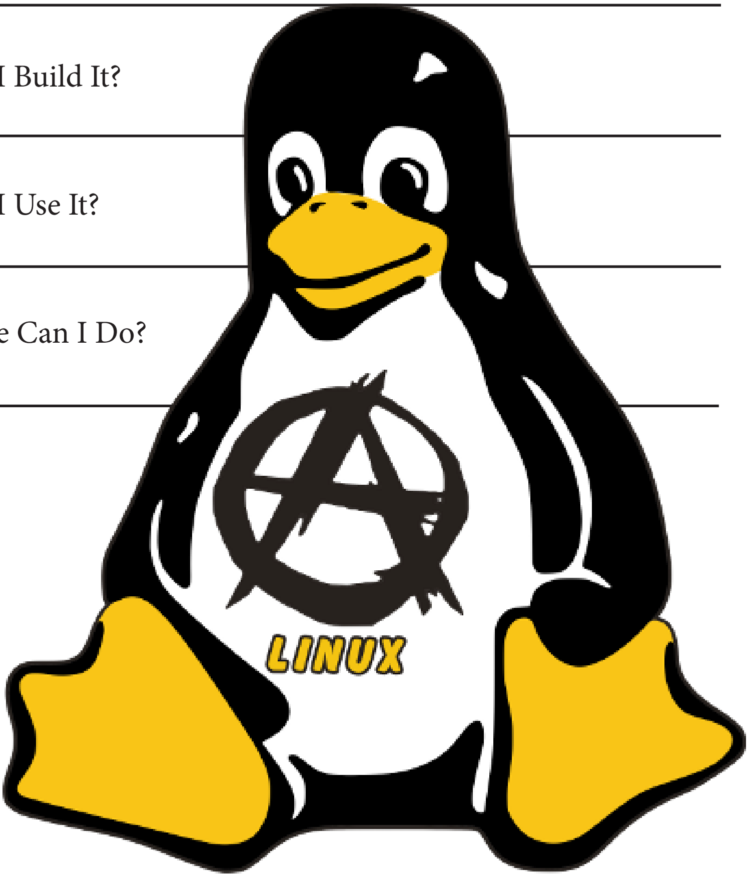


Radical DIY Computing

a workshop reader by

LEE TUSMAN & ECHO THEOHAR

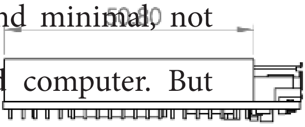

LINUX

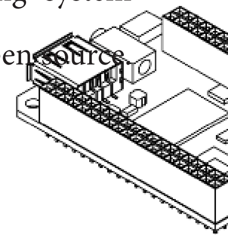
INTRODUCTION

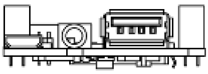
This is a DIY zine about building your own Linux computer with a combination of gathered, found and recycled parts. We'll walk through the process of building a cheapo Linux computer with as many scavenged parts as you can find. Not to overstress it too much, but doing this will be easy and fun. In the old days, you'd have to source a hard drive, motherboard, hard drive, monitor, keyboard, mouse, speakers. You still have to do that, but with a key difference. With the CHIP computer as a base and pre-configured Linux operating systems available, getting disparate parts to work together out-of-the-box is relatively straightforward. It's not quite as simple as turning on a Mac but it's not much harder, you'll learn how the different parts work together, and your computer will be custom built and serve as a good base for more exploratory and radical computing.

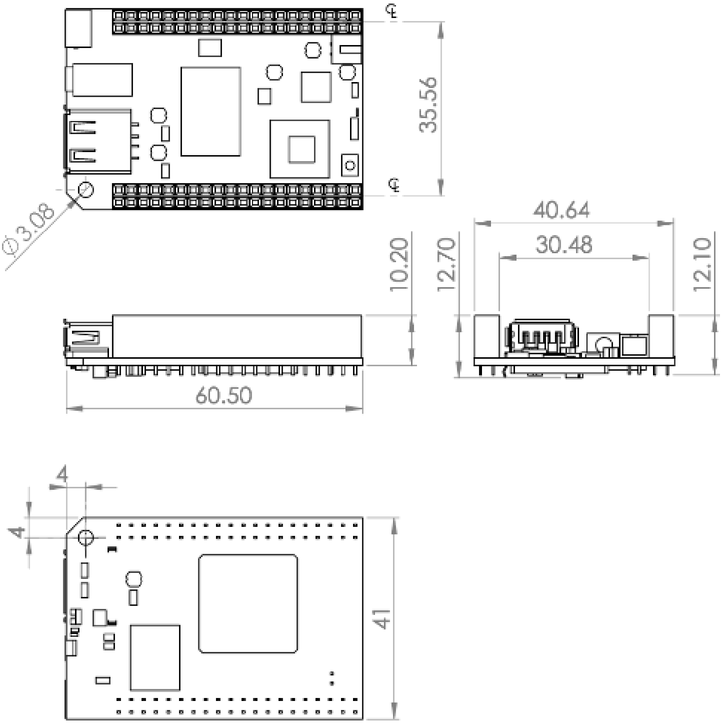
WHAT IS CHIP?

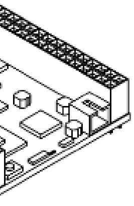
This zine is designed to help you build a computer based around using CHIP. CHIP is manufactured by Next Thing Co., based in Oakland, California, released for the first time in late 2016. CHIP is a low-cost development board with a version of Linux installed but absolutely zero hardware, so you need to add your own components like a case, keyboard, mouse, monitor. At the time of writing this in winter 2017 it costs $9. CHIP was released in late 2016 and is considered an alternative to the minimal Raspberry Pi series of computers. The CHIP is cost-effective and minimal, not the fastest, most streamlined computer. But it's extremely versatile. Out of the box, CHIP includes a working operating system with applications, bluetooth and wifi. The CHIP board and its customized operating system based on GNU Linux/Debian are open-source.











NEXT/THING COMPANY	SHEET 1 OF 1	MATERIAL:	
	UNITS: mm	FINISH:	
	TOLERANCE: +/- 0.1 mm	COLOR:	
TITLE: C.H.I.P. PCBA		CONTACT:	PHONE NO:
PART NO: 100088	DATE: 6/9/2016	REVISION: r1.0	EMAIL: ahoyahoy@nextthing.co

WHAT IS OPEN SOURCE AND FREE CULTURE?

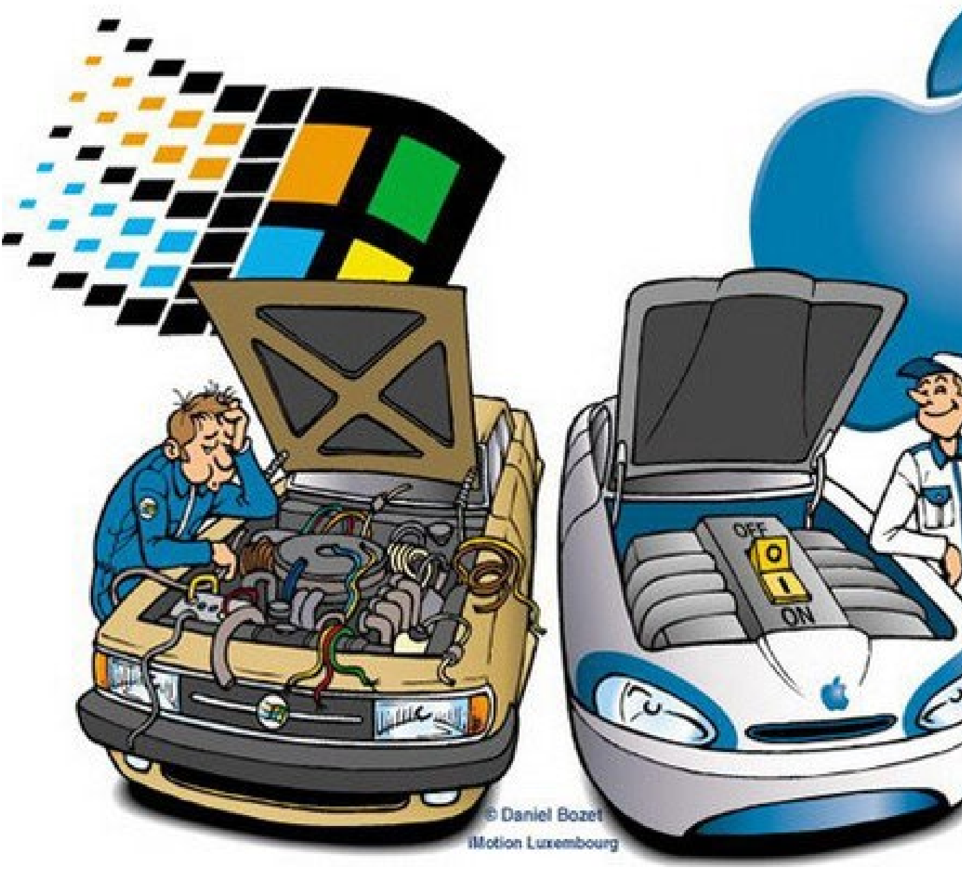
In the early days of computing in the 1950s - 70s computers were complex machines used by academic, government and corporate teams. Computers were distributed with their underlying source code so that they could be fixed or extended. As universities were some of the first users of computers they developed a culture of sharing and modification. Programs were distributed with their source code as well or printed in books and magazines. As computers and operating systems became standardized and home computing began to be a market companies began locking down the underlying code of their systems and software they sold. AT&T distributed their Unix Operating System without the source code available, and in the 1980s began charging for "patches" to upgrade systems. In 1976 Bill Gates wrote an open letter directed at the Homebrew Computer Club and various computer hobbyist zines.

> "As the majority of hobbyists must be aware, most of you steal your software. Hardware must be paid for, but software is something to share. Who cares if the people who worked on it get paid?"

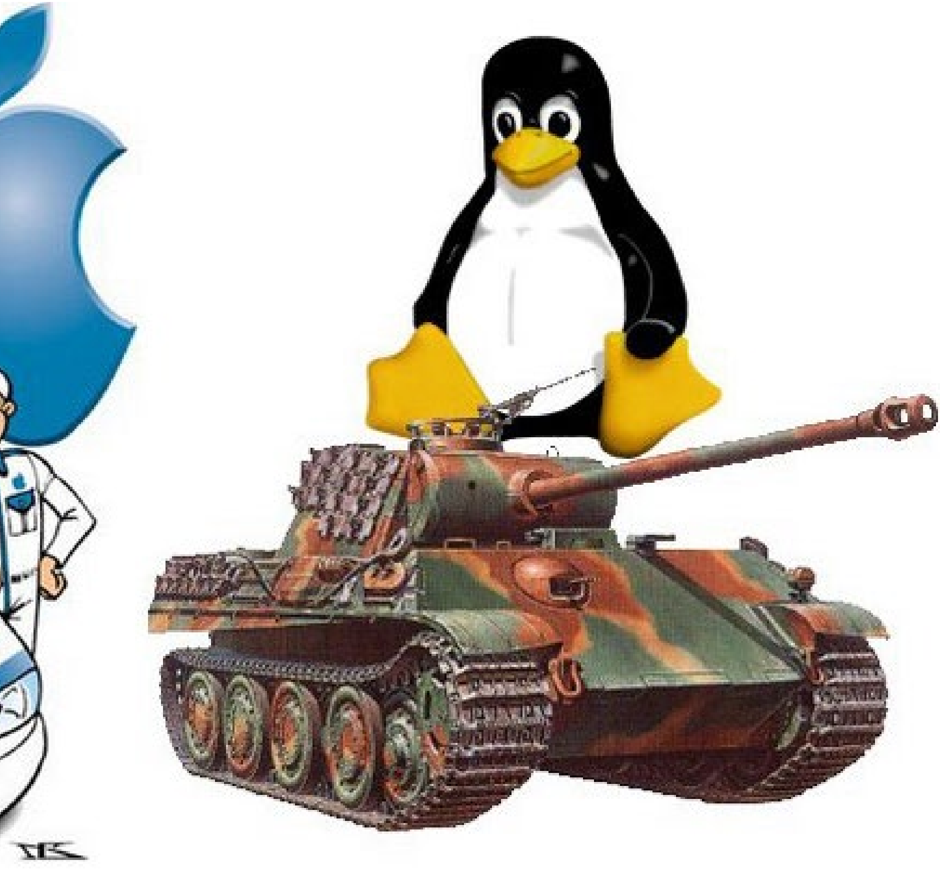
Gates' letter complained that computer hobbyists using his BASIC software and programming language without paying for it provoked strong reaction and disagreement. Even Apple responded with an ad in a computer magazine stating "our philosophy is to provide software for our machines free or at minimal cost". But the writing was on the wall and most companies charged for their computer operating systems and software. In 1980 copyright law was extended to computer programs.

In response, programmers, hackers and other hobbyists continued to write their own software and share their code. In 1983 Richard Stallman, a hacker at the MIT Artificial Intelligence Lab, announced the GNU project to develop a Unix-compatible free operating system. This community was interested in supporting “free” software. In free software culture, the “free” means free as in “liberty” or software that is free to modify and share. This also kickstarted the concept of copyleft, the practice of offering people the right to freely distribute copies and modified versions of a work as long as the new work contains the same license. The term Open Source was widely disseminated in 1998 when a team of programmers from various organizations and companies met with businesses and encouraged them to use the term for licesnced software in which the copyright holder provides the rights to study, change, and distribute the software to anyone for any purpose. Richard Stallman says that “Open source is a development methodology; free software is a social movement”, and there are internet fights over the differences. We’ll move on from that for now.

The largest
and most famous
open source project
in the world
is Linux.



© Daniel Bozet
iMotion Luxembourg



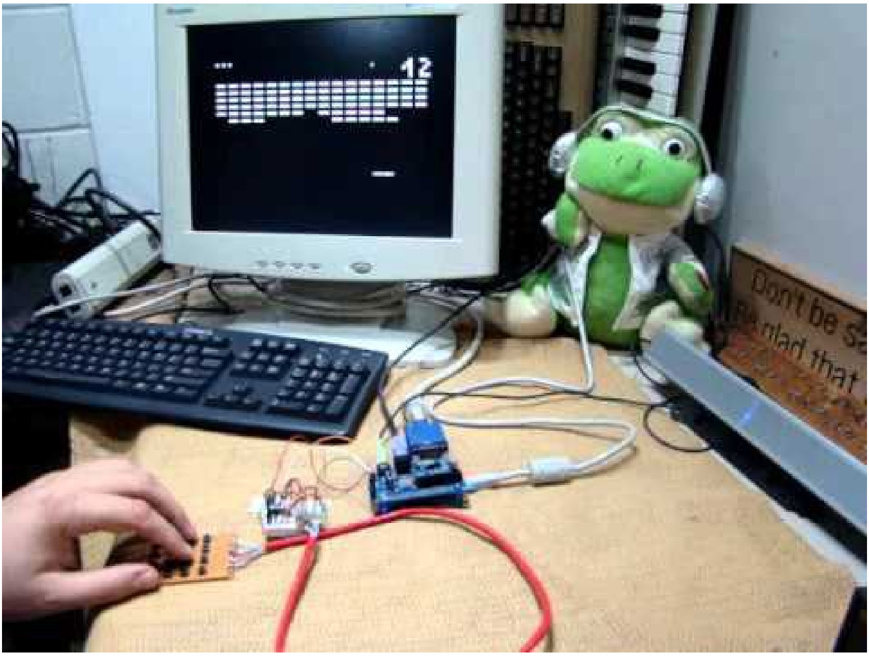
WHY LINUX?

The Unix Philosphy originated by famous programmer/computer scientist Ken Thompson is that programs are small, modular, reusable, and elegant. Unix was a loved, rock-solid operating system. But it wasn't free. A number of people attempted to make alternate free and open source alternative versions of the operating system. Communities of coders worked in email lists and online groups, sharing code and ideas. In 1991, Linus Torvalds released Linux. It is one of the most well-known open source projects of all time and one of the most successful. Linux was originally developed for desktop computers but first became widely used on web servers.

Today it used as the base system on all Android phones, Mac OS X, and in dozens of various alternative Linux distributions, called *flavors* or *distros* that package the operating system with utilities and applications. One of the more popular desktop versions of Linux is called Ubuntu. The CHIP development board comes packaged with a modified version of the well-known Linux Debian operating system, the first version of which was released in 1993 but is continually updated with new major releases every couple years.

WHAT DO I NEED?

Computer parts can be gathered from many places: your old computer parts kicking around in a basement or garage; your cousin's garage; university dumpsters; curbside trash; e-waste sites; thrift stores; stoop and garage sales. You are looking for cheap parts that just might work. If you have to fork over a few dollars, consider whether you have a way of checking them in store (if it's a monitor for example), or just keep your budget low so that you won't be out of money if it doesn't work at home. Honestly, you can spend good money and buy nice parts and build a nice computer. And if that's your goal, by all means, go ahead. It will still be way cheaper than almost any other computer. But if you like the idea of assembling a computer from scavenged parts, the CHIP is a low-risk, easy-to-assemble affair. You're looking for just a few parts (and you have lots of options). It's a tiny unit that takes a variety of standard hardware, with USB input, wifi and bluetooth. Out of the box you can output to a television with component cables, or with a DIP adaptor you can connect to a VGA or HDMI monitor. You could even use a projector instead. You can use a wireless keyboard connected via bluetooth, a wired USB keyboard or a very old keyboard with a S/2 connection if you find or buy a $2 USB-S/2 converter adapter. You can make a computer case out of a box, a toy, legos, or no case at all. It's all up to you. I've lucked out in thrift stores by finding cheap usb keyboards and mice. I've also found USB phone chargers with the right power output and micro-USB adapter cables that will plug in. So the whole process can be rather cheap. And it feels like success to build a computer from cheap or scavenged parts.



CHIP BASIC ITEMS CHECKLIST

✓ CHIP computer

✓ Power supply for the CHIP (Battery or USB charger)

✓ Keyboard

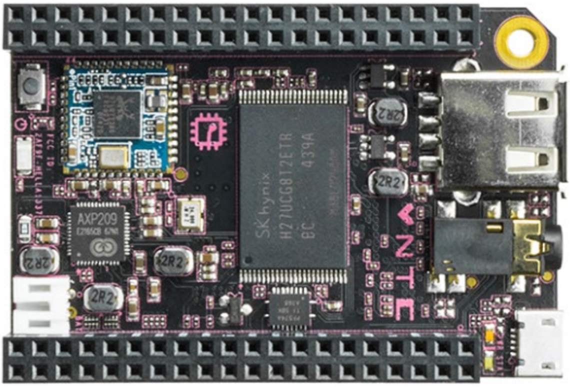
✓ Mouse

✓ Monitor

✓ Composite RCA cable that connects to an oldschool TV or HDMI cable with HDMI DIP adaptor, or VGA cable with VGA DIP adaptor

✓ Speaker(s) and 1/8” cable to connect to the CHIP or headphones (optional)

✓ A case (optional: purchased, made out of legos, etc)



DETAILED BREAKDOWN

Note: the chip already has wireless internet connection + 4 GB of storage built in, so no need for SD cards!

Power Supply ▼

As mentioned above, purchase or find the right size power supply. The CHIP uses a 5V (5 Volts, 2 Amps) USB wall charger power supply. The cable itself is a standard USB to microUSB phone charger type cable. The CHIP can also be powered via battery. For more info check the CHIP's manual on your computer or the getchip.com website.

Monitor ▼

Tons of 5-10 year-old monitors in varying degrees of quality can be found at thrift stores, basements and curbside trash days. You can even use a television. You'll need the right adaptor cable for your CHIP and monitor. The simplest thing is to have a monitor with HDMI input connected with an HDMI cable + adaptor to your CHIP. If you have a VGA monitor you'll need to get a dongle with female HDMI on one side and male VGA on the other (probably! Again, check your actual equipment to be sure) + an adaptor.

Input ▼

Any keyboard you like will do with USB or bluetooth wireless. Any USB mouse or trackball should work as well.

Speaker(s) ▼

This is optional. You can connect headphones, crappy computer speakers, a buzzer, a cable to your stereo, or to your television input. If you're using a TV as your monitor and it has audio input, you might as well use that.

Case ▼

This is completely optional. You can buy a custom case, 3D print or lasercut your own, build something from legos, or go without. Feel free to do an image search online for CHIP case for ideas.

HOW DO I BUILD IT?

After collecting all the parts, the assembly is trivial!

1. Plug in your keyboard and mouse to your CHIP. Plug in a USB hub first and then plug into that. If you want to plug in both you may need a powered hub, or else use a wired mouse with bluetooth keyboard. For bluetooth support you can refer to the manual on your CHIP when you turn it on or online in the manual found at getchip.com.

3. Plug into a monitor. If you have a TV monitor, you'll use a Composite cable (a RCA-type cable), which are easy to find in tech closets, basements, thrift stores, etc. If you have an HDMI monitor or VGA monitor, you'll need the right DIP adaptor in addition to your cable. At first, this may look scary. But it's actually quite easy. Align the adaptor's pins exactly over the Chip board. The USB connector should be vertically on the opposite end from the HDMI or VGA connector. Gently press the two boards together so they snap into place. You can now attach a HDMI or VGA monitor.

5. Congratulations, you have a Linux computer.

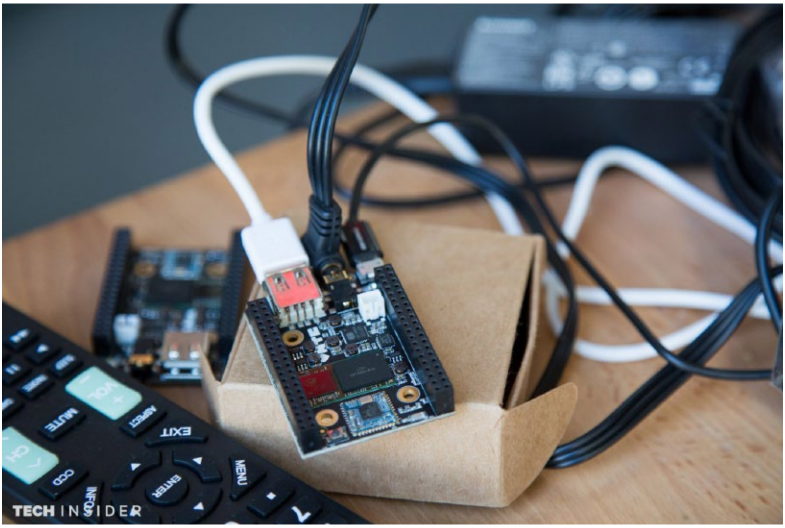
HOW DO I USE IT?

Wifi ▼

Turn it all on by plugging into power. You'll boot into the sytem after some moments and see a desktop. It's different than Windows or Mac but should still look somewhat familiar. In the top right click on the Network icon and choose a Wifi network to connect to.

Bluetooth ▼

If you are using a bluetooth device, click on the bluetooth icon in the top right. Choose Setup New Device. You'll be guided through steps to pair your device and possibly prompted for a code. In the future this information will be saved so you don't have to re-enter this.



Applications ▼

CHIP comes with a variety of basic applications such as text editors, calendar, games, screenshot software, music and video players, web browser and more. These can be accessed from the **Computer Things!** menu. Check out:

- Display - In the Settings section. You may want to adjust your monitor's resolution.
- AbiWord - Word Processor software
- IceWeasel - a clone of FireFox
- MPlayer - a video player
- Leafpag - a note-taking app
- Synaptic Package Manager - not the most attractive app, but it is what lets you search for new programs, download and up date the programs on your computer
- Terminal - the commandline for text-based interaction with your operating system

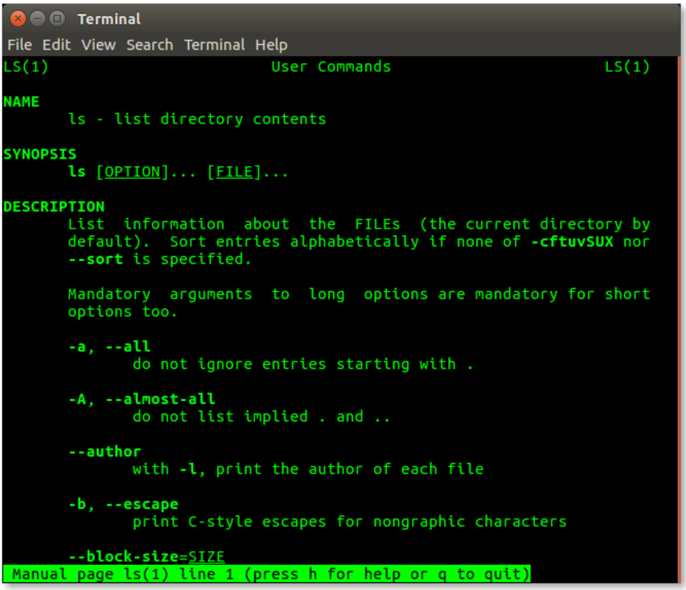
WHY USE THE TERMINAL?

You have a graphical operating system. Why would you use the terminal?

The terminal is the place where you type commands to manipulate files on your computer or launch programs that perform tasks. In some ways the command prompt is the *simplest* kind of computer interface possible. It is likely that you are more familiar with interfaces that have windows and buttons, but the command prompt is an interface entirely driven by text input.

You enter a command, execute it, and the command is performed. Then you can enter another command that builds off this, or does something entirely different. Writing a program for the command prompt is easier than writing a program with a graphical interface (for example windows and buttons). It is so much easier, in fact, that the majority of programs written today are written for the command line. Many of the programs written this way are customized, personal programs. This is why learning to use the command prompt will open up new worlds for you. You will have access to a wide variety of programs and technologies, many that may have been off-limit or hidden before.

It is simple and minimalist. It is ubiquitous. Nearly every computer from OS X, CHIP, to web servers, internet devices, routers and robots can be controlled through the command line, and even more devices are being created that can be interacted with that are online (through the so-called “Internet Of Things”, which is a nebulous term but is driven on the back of the command line). The commandline is present on devices that don’t even have a screen, mouse, internet, et cetera. Some tasks are lightyears faster (or instant) when calling the command from the command line instead of using a specific application or having to find a specific program to download from the internet. If you’re working on an old or underpowered computer, this could be a gamechanger that allows you to keep and use your computer rather than letting it become garbage.



```
LS(1)                            User Commands                           LS(1)

NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about  the  FILEs  (the current directory by
       default).  Sort entries alphabetically if none of -cftuvSUX nor
       --sort is specified.

       Mandatory  arguments  to  long  options are mandatory for short
       options too.

       -a, --all
              do not ignore entries starting with .

       -A, --almost-all
              do not list implied . and ..

       --author
              with -l, print the author of each file

       -b, --escape
              print C-style escapes for nongraphic characters

       --block-size=SIZE
 Manual page ls(1) line 1 (press h for help or q to quit)
```

SHORT TERMINAL GLOSSARY FOR BEGINNERS

▷ **cd** change directory. open a folder.

▷ **mkdir** make directory. create a folder.

▷ **ls** list files in the current directory so you know what is in it.

▷ **mv** move a file from one directory to another, or to give it a new name.

▷ **rm** remove a file. delete it, and beware! Use the **-r** to make it recursive to delete a directory.

▷ **sudo** “super user do”. Many commands need administrator-like privileges, otherwise they won’t work. You’ll see sudo as the first word in a lot of commands - all it is doing is giving the command the necessary access. You’ll be asked for a password the first time you use sudo. **The default password and user is “chip”.**

▷ **apt-get** the command used for installing, removing, and finding software for Debian Linux systems, such as the CHIP Operating System.

▷ **pwd** “present working directory”. In case you forget where you are. Not much to it: **pwd** will output the directory name, such as **/Users/home/chip/Pictures/Vacation/**

▷ **grep** a tool used for searching through files. It’s quite deep and can be complicated, but if you see the word grep in some command, you know it’s searching for a match.

WHAT ELSE CAN I DO?

- ▷ Build a Web Radio
- ▷ Build an encrypted web browser and communication tool
- ▷ Make a custom case for your computer
- ▷ Browse the community forum bulletin board at bbs.nextthing.co and look for new projects and ideas
- ▷ Download a Nintendo or Doom emulator
- ▷ Make a Chippy Ruxpin
- ▷ Build a Piratebox Anonymous file sharing and messaging system
- ▷ Teach yourself to program in Python or Javascript
- ▷ Build your own alternative homemade smartphone
- ▷ Make games with Pico8
- ▷ Learn Bash, the language of the commandline

Note: More terminal commands (with examples) may be found at https://docs.getchip.com/chip.html

ADD’L RESOURCES

* bbs.nextthing.co - Chip online bulletin board and forums

* In The Beginning Was The Command Line by Neil Stephenson

- famous cyberpunk author on the importance of free culture and

the command line versus proprietary operating systems

* edX free course Introduction to Linux

* any book or website on Linux!