

# macOS ARM xv6-riscV Install

- **Prerequisite**

- Xcode
- Command Line Tools part of Xcode
- Homebrew (<https://brew.sh/>)

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

- **Installing xv6-riscV**

Open the macOS terminal of your choice.

Note: Everything with \$ just paste in your terminal do not include the \$

Confirm home-brew is update:

```
$ brew update  
$ brew upgrade
```

Install rosetta (probably don't need but idk)

```
$ softwareupdate --install-rosetta
```

Install qemu

```
$ brew install qemu
```

Get the riscv-toolchain (<https://github.com/riscv-software-src/homebrew-riscv>):

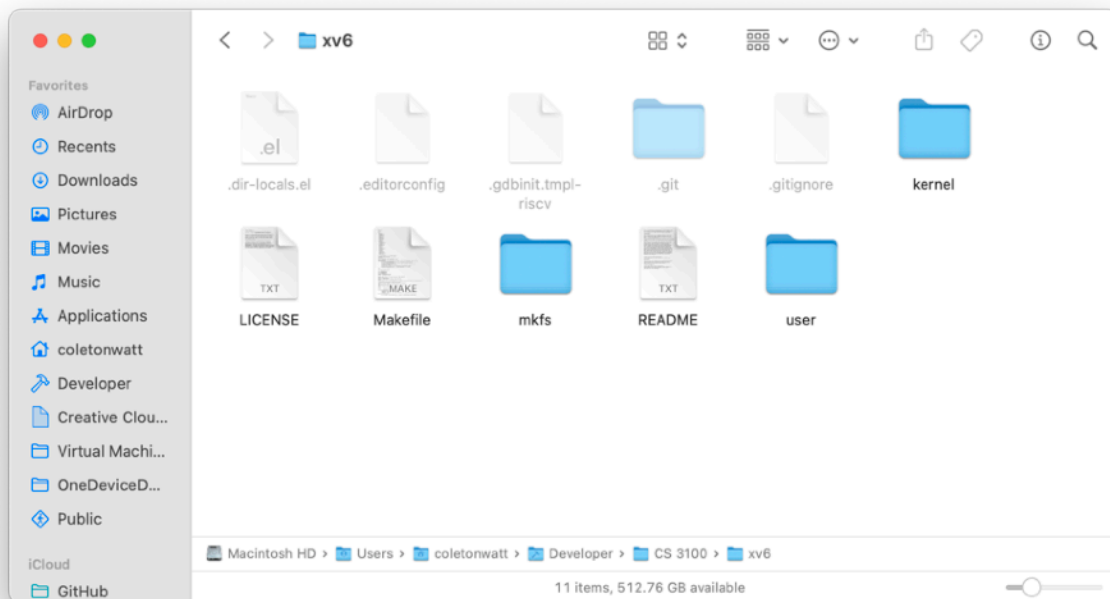
```
$ brew tap riscv-software-src/riscv  
$ brew install riscv-tools
```

This install does take forever specially the make command (mine took like 20ish minutes)

```
$ brew install riscv-openocd
```

Test the install

```
$ brew test riscv-tools
```



Download the xv6-riscv file  
and where you put is up to you I used GitHub Desktop

how

<https://github.com/mit-pdos/xv6-riscv>

Open the file location in terminal (if you have the bottom path bar you can right click the file and select Open in Terminal)

Run xv6

\$ make qemu

```
xv6 — qemu-system-riscv64 ◀ make qemu — 80x24

Last login: Thu Jan 11 10:50:23 on ttys009
[coletonwatt@coletons-mbp xv6 % make qemu
mkfs/mkfs fs.img README user/_cat user/_echo user/_forktest user/_grep user/_init user/_kill user/_ln user/_ls user/_mkdir user/_rm user/_sh user/_stressfs user/_usertests user/_grind user/_wc user/_zombie
nmeta 46 (boot, super, log blocks 30 inode blocks 13, bitmap blocks 1) blocks 1954 total 2000
ballocc: first 759 blocks have been allocated
ballocc: write bitmap block at sector 45
qemu-system-riscv64 -machine virt -bios none -kernel kernel/kernel -m 128M -smp 3 -nographic -global virtio-mmio.force-legacy=false -drive file=fs.img,if=none,format=raw,id=x0 -device virtio-blk-device,drive=x0,bus=virtio-mmio-bus.0

xv6 kernel is booting

hart 2 starting
hart 1 starting
init: starting sh
$ █
```

**You are now in xv6 congrats to exit press control+A,  
then X**

**Control+A, then C will open Qemu  
\$ make qemu clean will recompile all files of xv6**

If you need help you can reach out to me in class or through Discord (eleninja102)

- **Optional**

Orbstack (<https://orbstack.dev/>) allows you to run linux commands in a linux machine similar to WSL for windows. Also is great if you are working with docker or Kubernetes

<https://pdos.csail.mit.edu/6.S081/2020/xv6/book-riscv-rev1.pdf>

Vargrant can emulate x86 and allow for xv6-x86, however, it is pretty unstable and laggy.