

Recommended Development Env.

Docker

1. Install docker from <https://www.docker.com/> based on your operating system.
2. Get familiar with Docker development environment.
3. Pull tinyOS image from [docker hub](#) – https://hub.docker.com/r/ucmercedandeslab/tinyos_debian/
 - a. To pull the image
 - i. **docker pull ucmercedandeslab/tinyos_debian**
 - ii. To list the available images
 1. **Docker images**
 - iii. To run the image in this case – tinyos
 1. **docker run -ti <Image_Name>**
 2. **-ti** : to open an interactive bash console(refer to the docker docs)
 - iv. To test the installation
 1. **cd /opt/tinyos-main** #TinyOS root folder
 2. **cd /apps/Blink/**
 3. **make micaz sim** #compile first tinyOS program
 - v. micaz is the type of board that we are simulation and the sim flag makes the simulation binaries instead of the full install. If it is able to compile, then you have successfully installed TinyOS. If not, then look at the output and see if you can find the issue. If you are unable to solve the problem, ask the TA for help.

Some useful links to get started with Docker:

1. <https://docs.docker.com/get-started/>
2. <https://medium.freecodecamp.org/a-beginner-friendly-introduction-to-containers-vms-and-docker-79a9e3e119b>
3. <https://medium.com/@kangzeroo/docker-easy-as-build-run-done-e174cc452599>

Important points:

4. Docker has a higher learning curve initially but it's worth it and will help you guys in collaboration and future projects.
5. If you are on windows 10 home edition you'll have to install docker toolbox or upgrade your windows version to install docker.
6. Virtual Machines like Virtual Box and VMware don't jell well with docker on windows 10 Pro.
 - a. **bcdedit /set hypervisorlaunchtype auto** – to use docker
 - b. **bcdedit /set hypervisorlaunchtype off** – to use virtual machines

Testing TOSSIM

We first want to see if TinyOS was properly installed so go into the TinyOS directory.

```
cd $TOSROOT/apps/Blink/
```

Now we are going to compile your first TinyOS program.

```
make micaz sim
```

micaz is the type of board that we are simulation and the sim flag makes the simulation binaries instead of the full install.

If it is able to compile, then you have successfully installed TinyOS. If not, then look at the output and see if you can find the issue. If you are unable to solve the problem, ask the TA for help.

Once you are done you can checkout the skeleton code which is a foundation for you to build the future projects. Explore the contents and try to get an understanding of how the module and component system works.

Go to where you will store your project and check out the skeleton code.

```
git clone https://github.com/UCMAndesLab/CSE160-Project-Skeleton-Code
```

For Native/Virtual Machine Installation:

<https://catcourses.ucmerced.edu> under files ->

Projects -> Project 0 – Getting starter.pdf