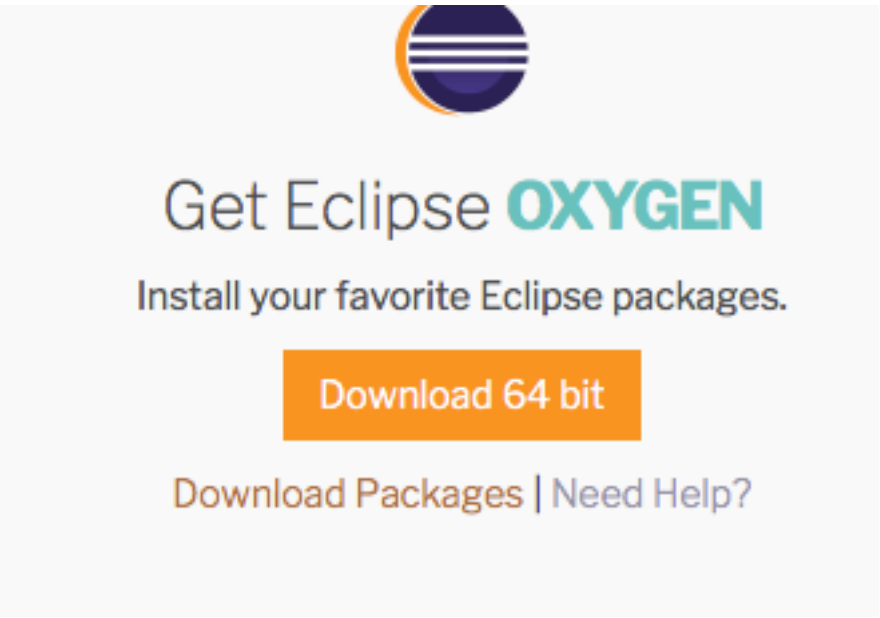
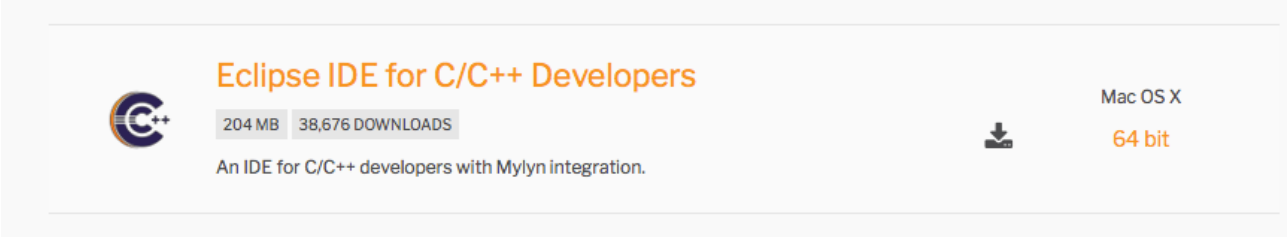
TOOLCHAIN INSTALLATION INSTRUCTIONS FOR MAC OS

For students of the Foundations of ARM course

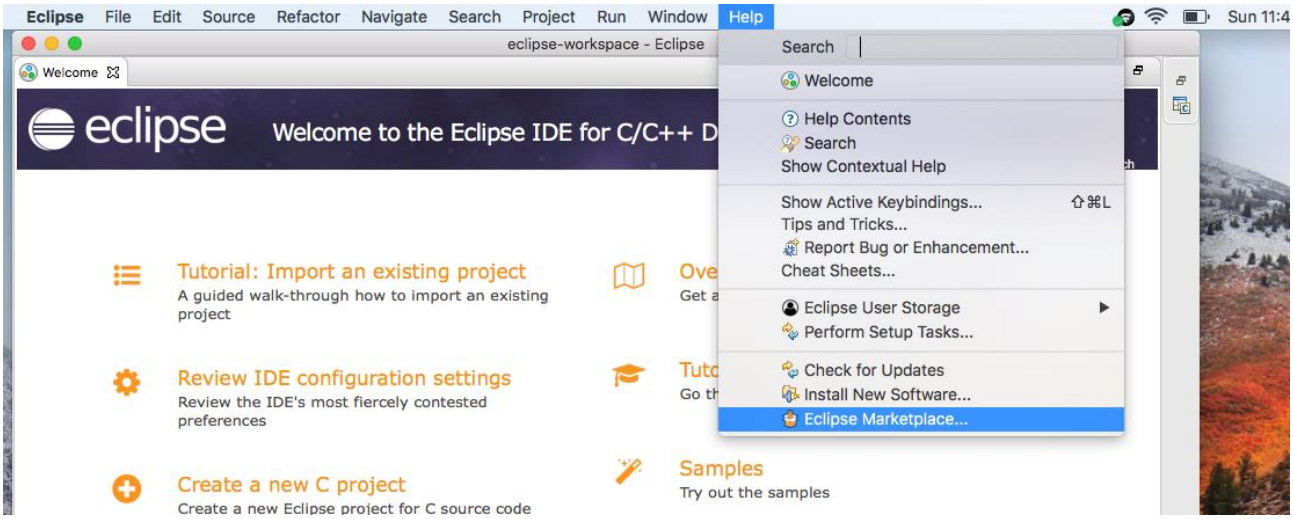
Written by Adrian Forschner
On behalf of MakerMax Inc.

1) Download Eclipse (i prefer to download the CBT version) you can do that under Download Packages

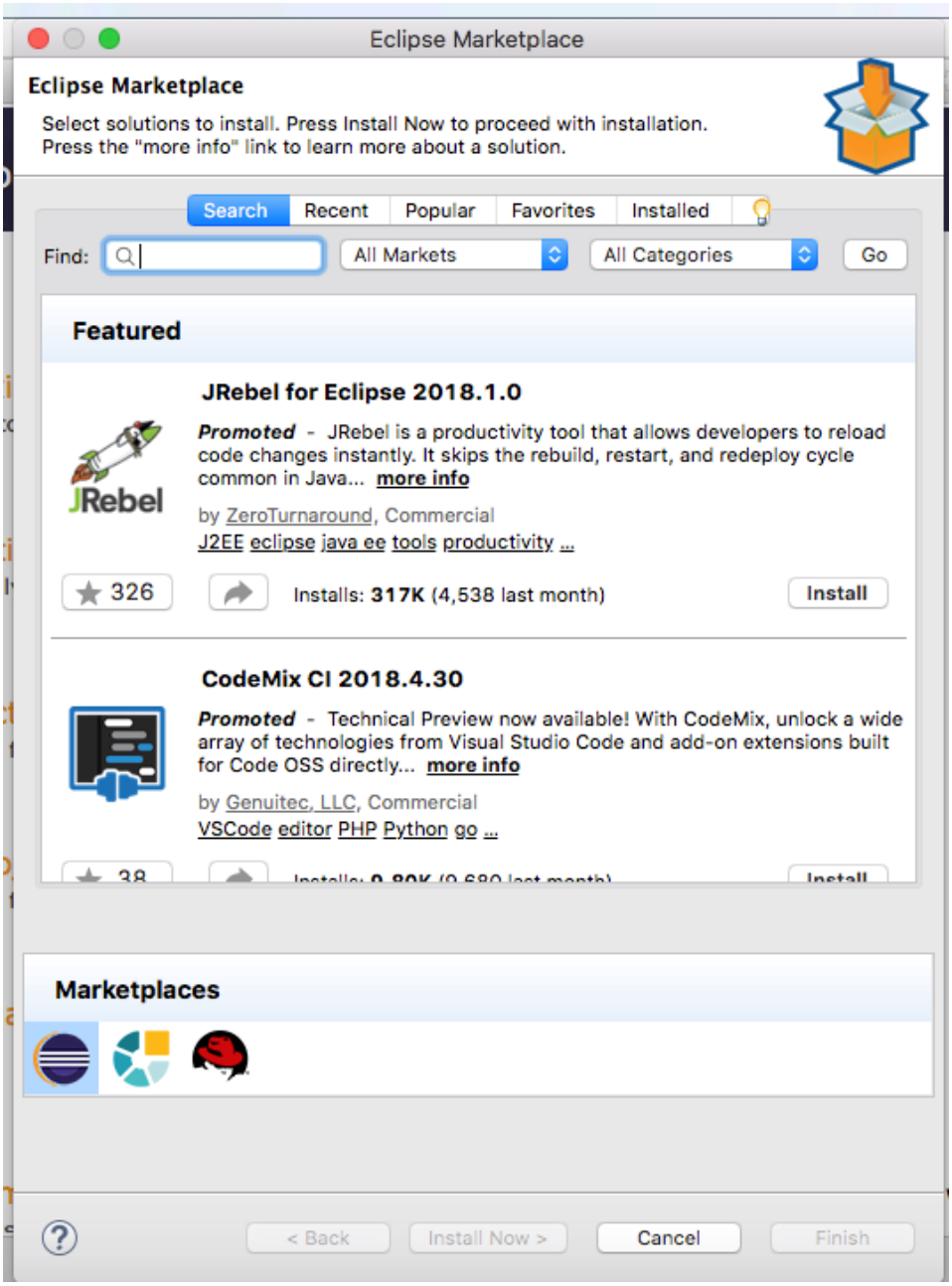




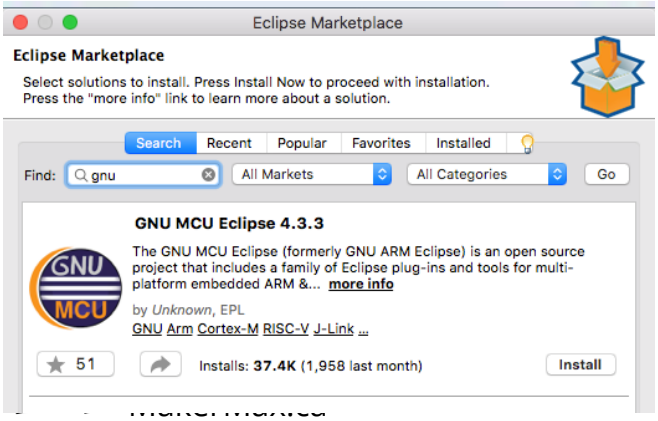
2) Install eclipse and go to Help -> marketplace



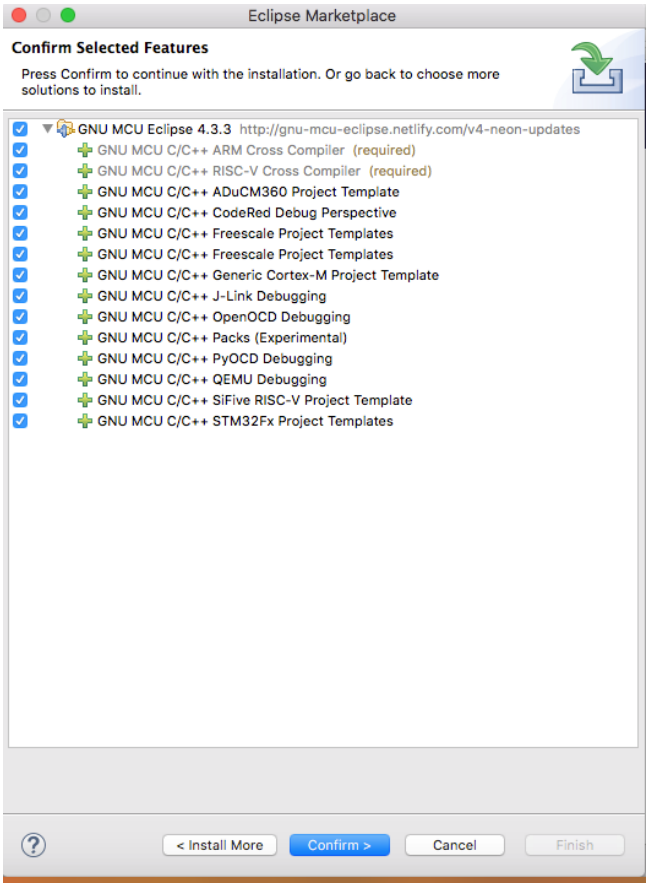
3) In marketplace search for GNU MCU



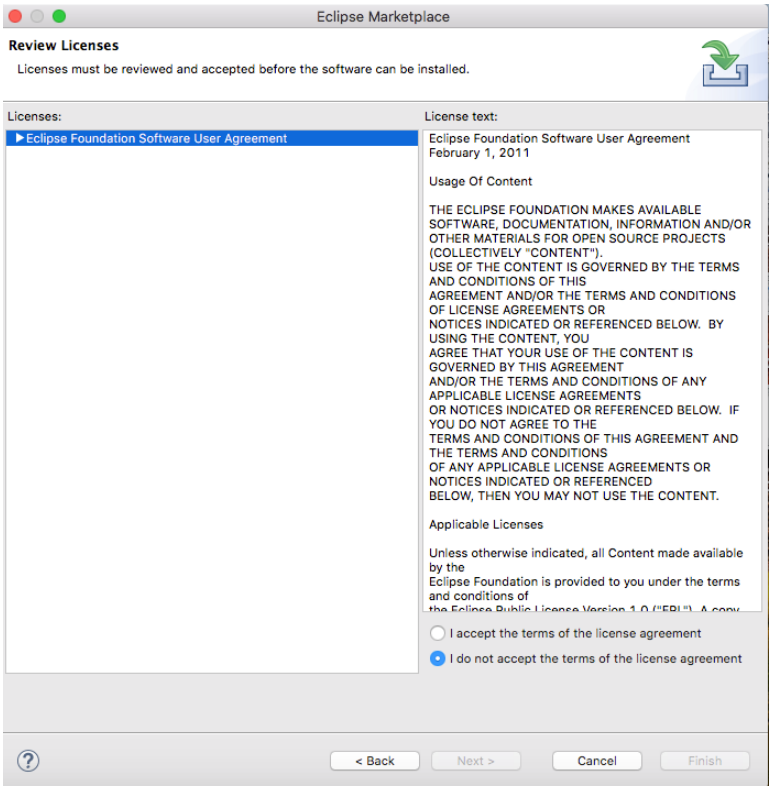
This will install the GNU MCU plug in

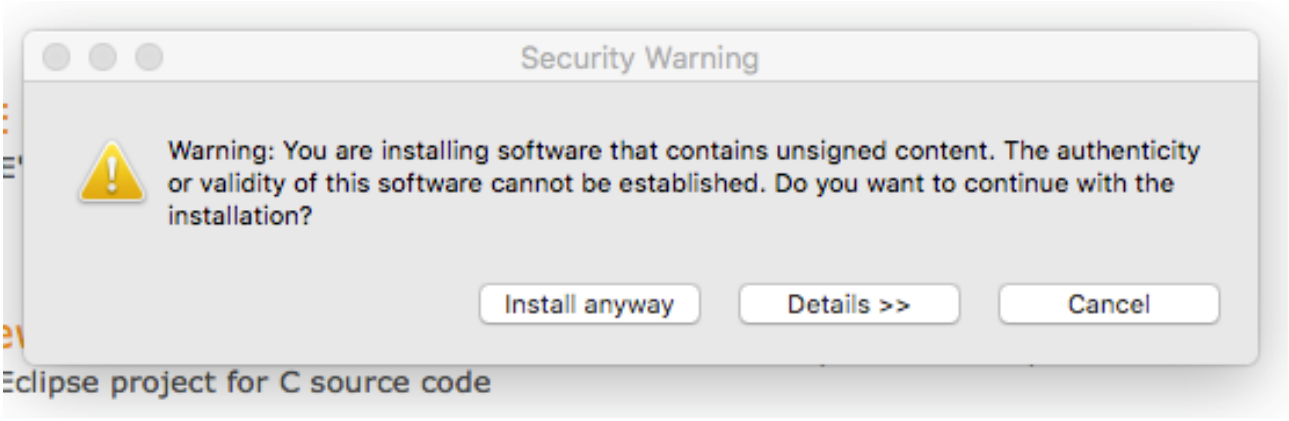


4) Check all the plugins you like to install:

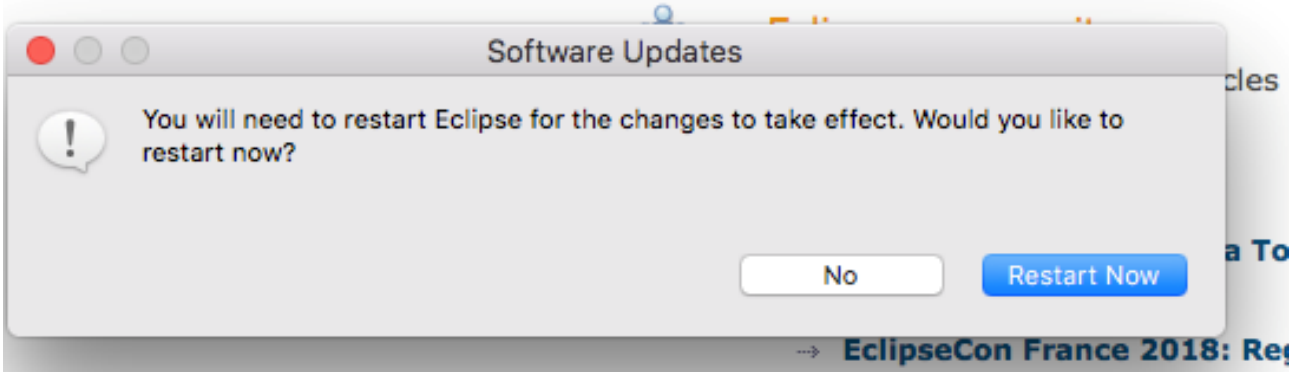


5) Accept the license terms





6) Choose install anyway and then restart eclipse



3) Go to the Github page of the gnu eclipse mcc plugins (https://github.com/gnu-mcu-eclipse)

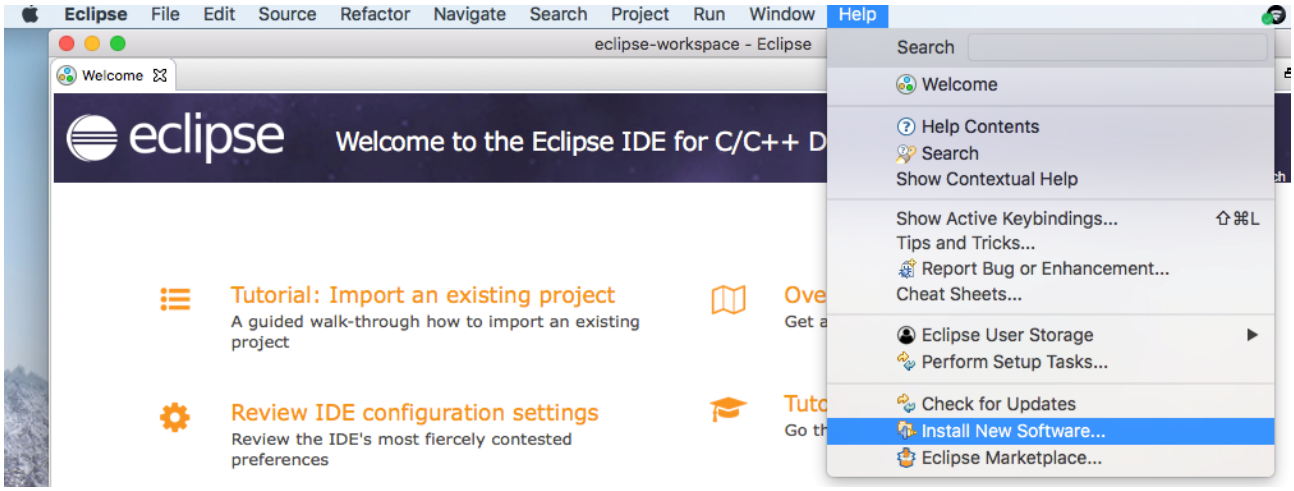
from here you need to download opened and the toolchain
the toolchain is at https://github.com/gnu-mcu-eclipse/arm-none-eabi-gcc
and you need to choose
gnu-mcu-eclipse-arm-none-eabi-gcc-7.2.1-1.1-20180401-0515-osx.tgz

for opened you go to https://github.com/gnu-mcu-eclipse/openocd/releases
gnu-mcu-eclipse/openocd

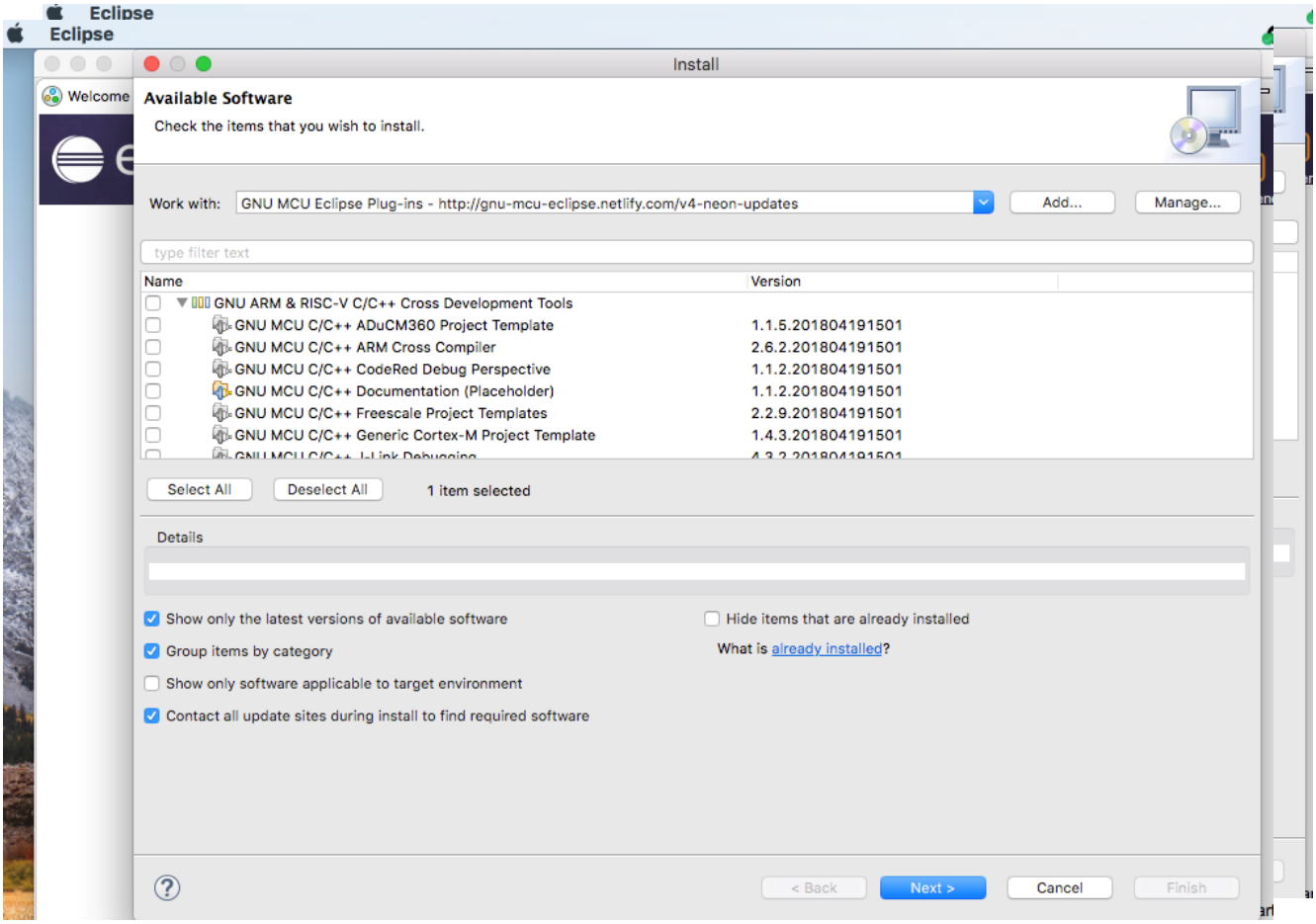
and you choose:
gnu-mcu-eclipse-openocd-0.10.0-7-20180123-1217-osx.tgz

7) Unpack the downloaded archives in a folder of your choosing.

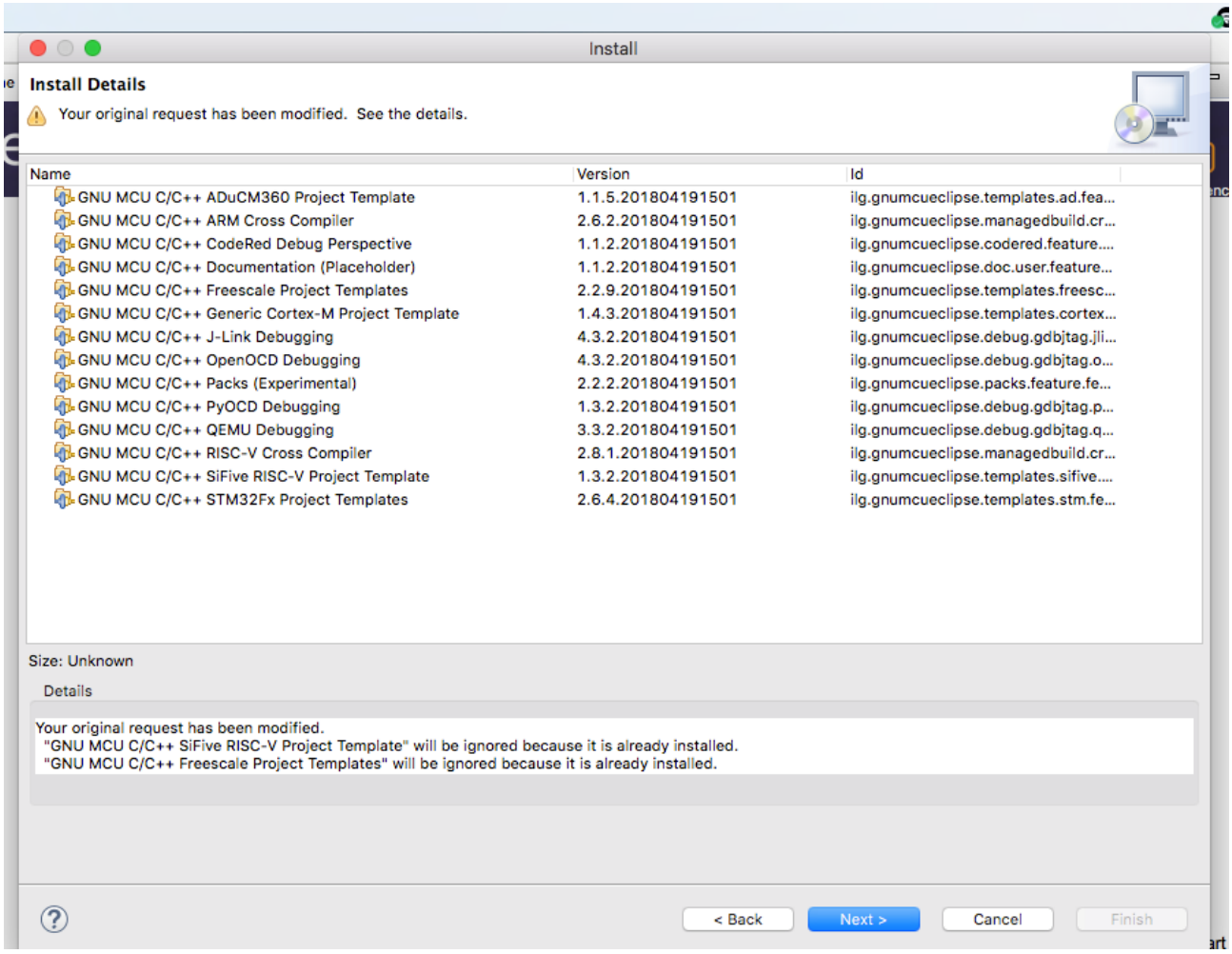
8) Reopen eclipse and proceed to Help -> Install New Software



9) In the Work With dropdown menu, select GNU MCU Eclipse Plug-ins

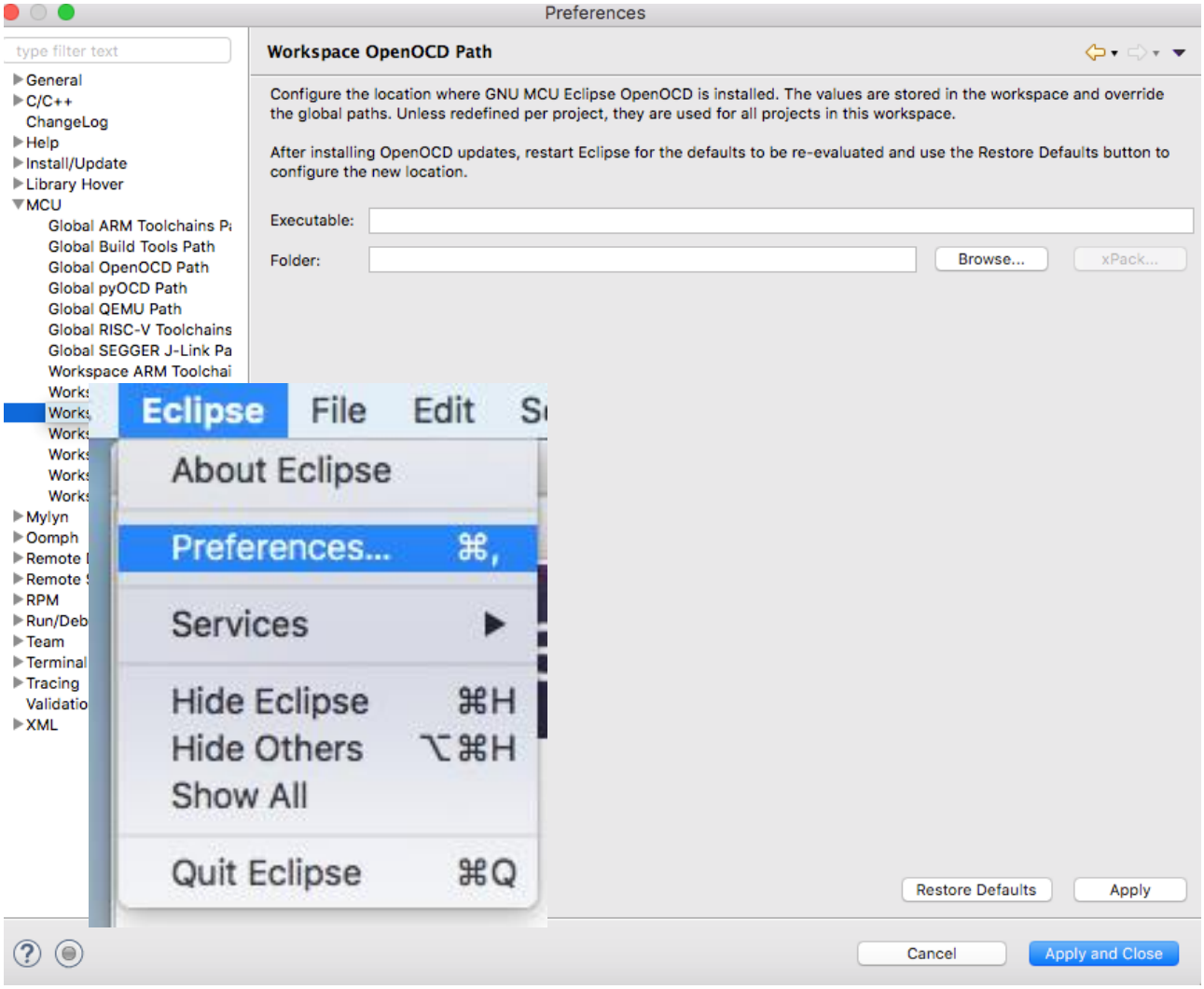

Eclipse
Eclipse
Install
Welcome
Available Software
Check the items that you wish to install.
Work with:
GNU MCU Eclipse Plug-ins - http://gnu-mcu-eclipse.netlify.com/v4-neon-updates
Add...
Manage...
type filter text
Name
Version
GNU ARM & RISC-V C/C++ Cross Development Tools
GNU MCU C/C++ ADuCM360 Project Template
1.1.5.201804191501
GNU MCU C/C++ ARM Cross Compiler
2.6.2.201804191501
GNU MCU C/C++ CodeRed Debug Perspective
1.1.2.201804191501
GNU MCU C/C++ Documentation (Placeholder)
1.1.2.201804191501
GNU MCU C/C++ Freescale Project Templates
2.2.9.201804191501
GNU MCU C/C++ Generic Cortex-M Project Template
1.4.3.201804191501
Select All
Deselect All
1 item selected
Details
Show only the latest versions of available software
Hide items that are already installed
Group items by category
What is already installed?
Show only software applicable to target environment
Contact all update sites during install to find required software
< Back
Next >
Cancel
Finish

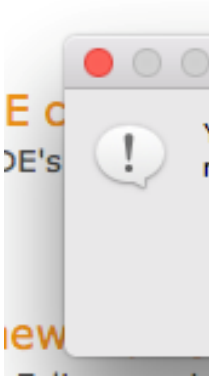
9) Click Next



10) Click Install Anyway

11) Click Restart Now

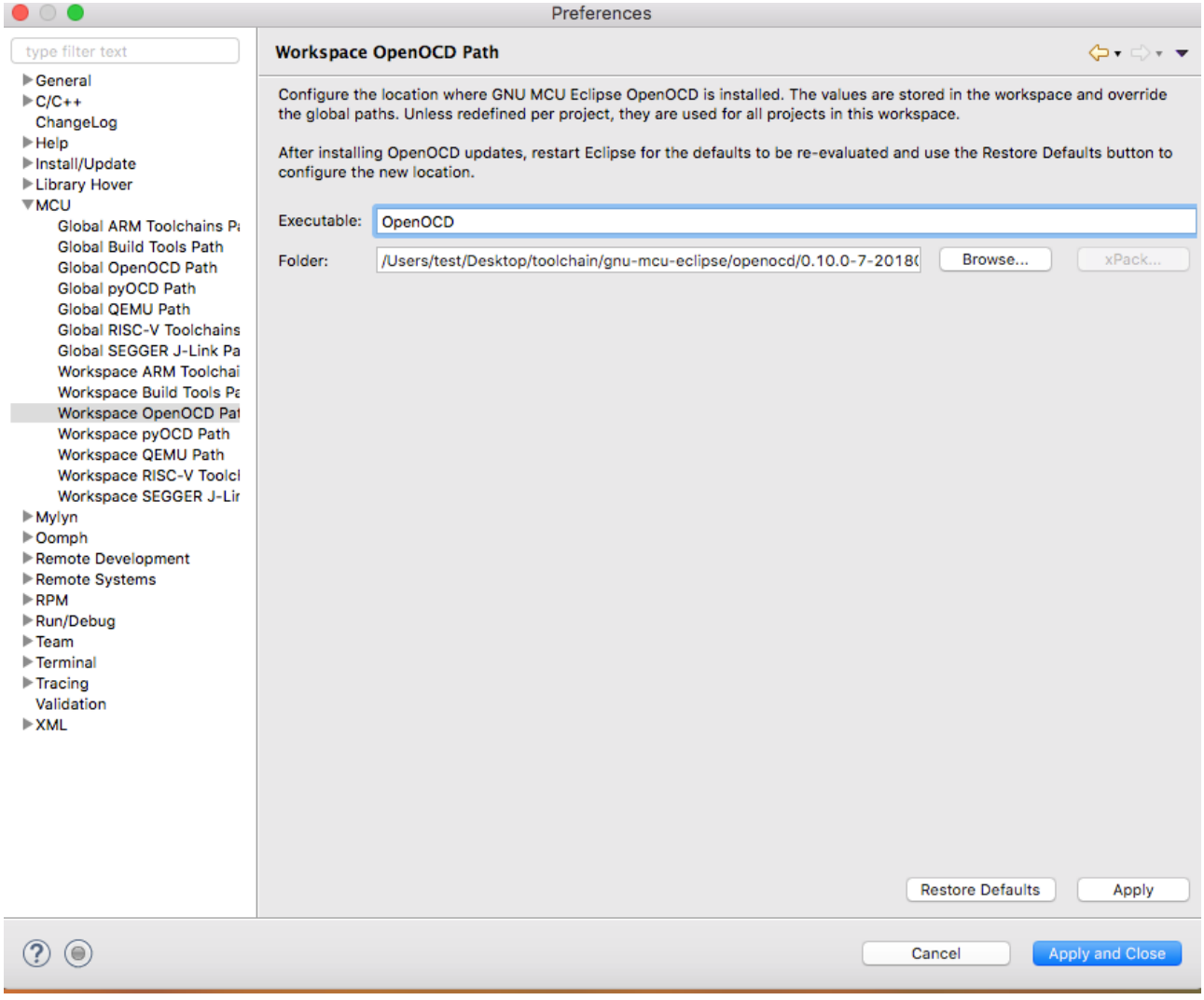




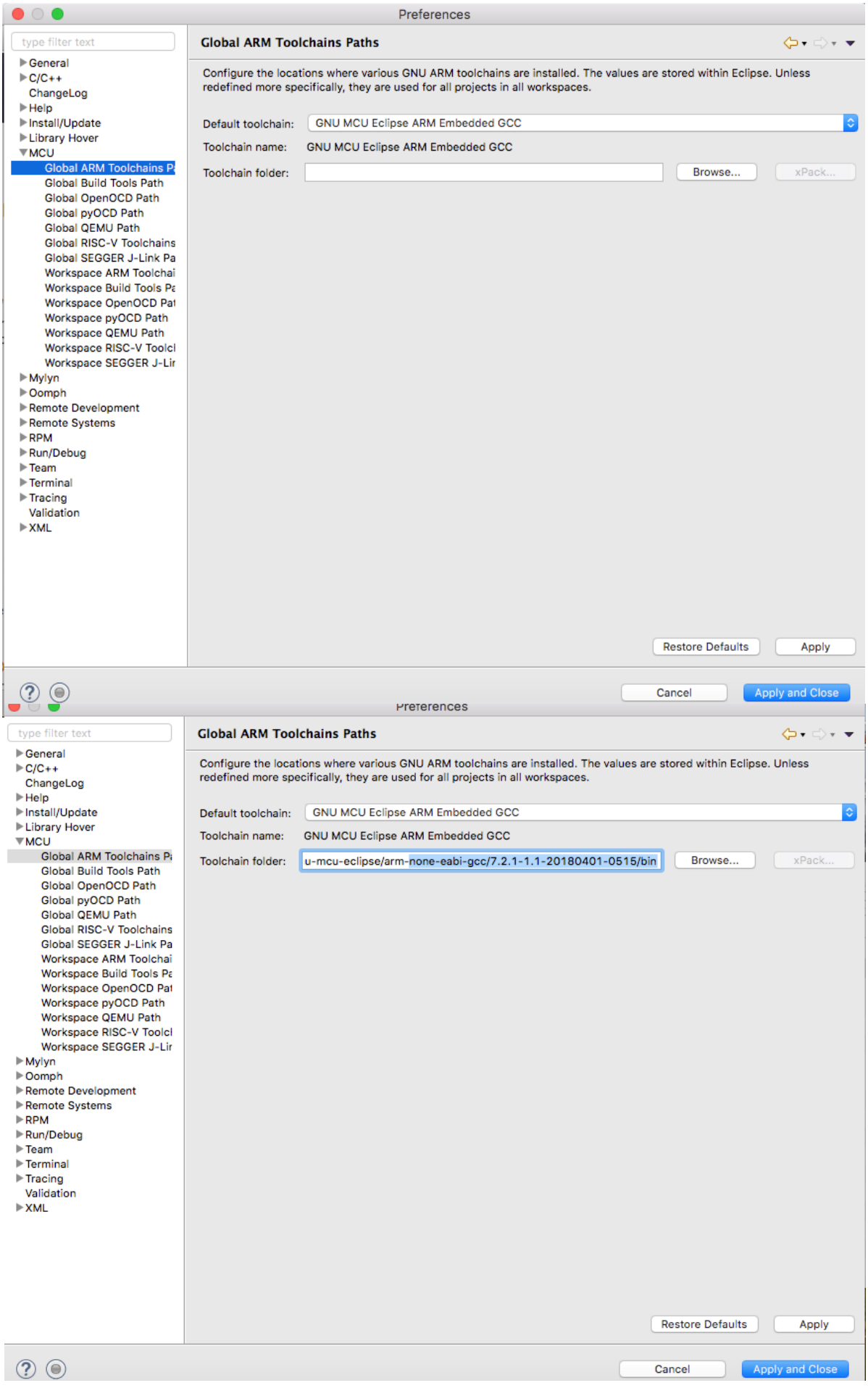
12)

After Eclipse Restarts, go to the Eclipse -> Preferences

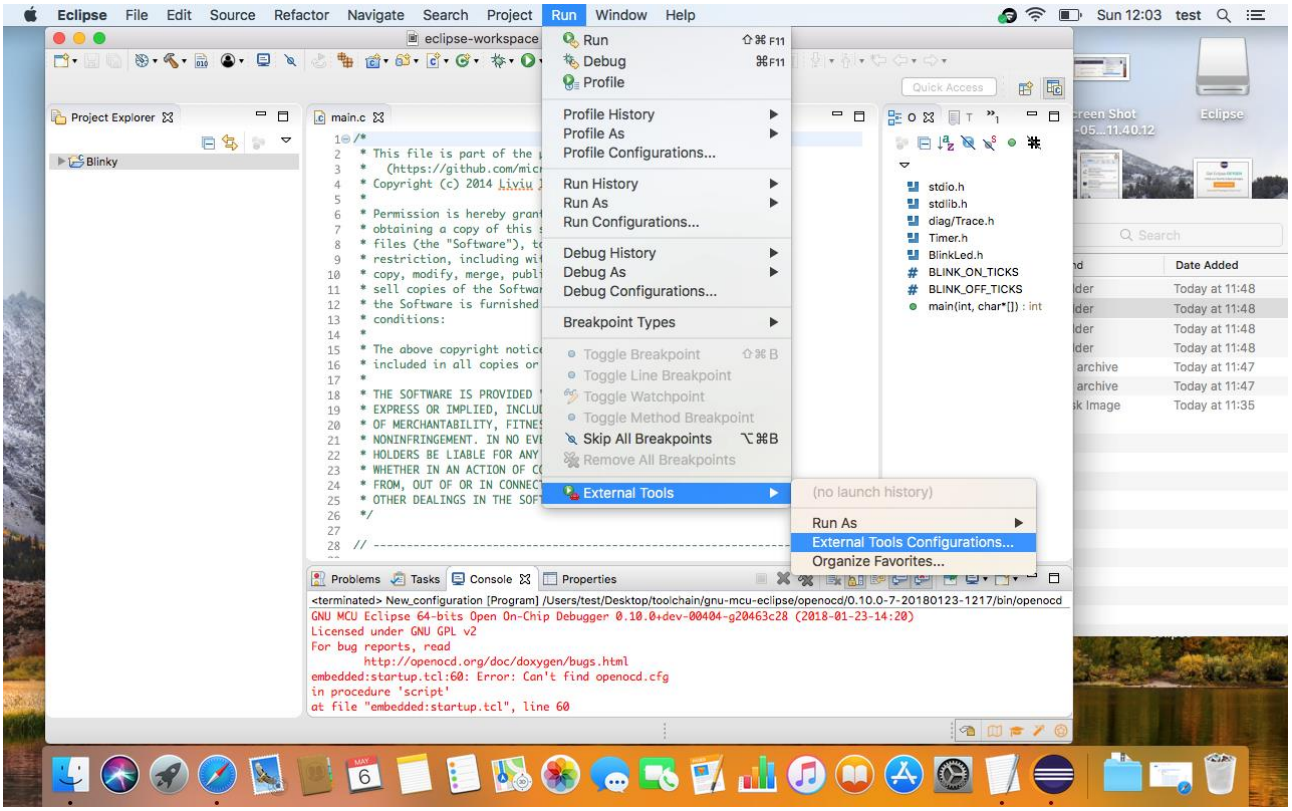
13) Navigate to Workspace OpenOCD Path under MCU and enter the directory where you copied the OpenOCD executable


Preferences
type filter text
General
C/C++
ChangeLog
Help
Install/Update
Library Hover
MCU
Global ARM Toolchains Pa
Global Build Tools Path
Global OpenOCD Path
Global pyOCD Path
Global QEMU Path
Global RISC-V Toolchains
Global SEGGER J-Link Pa
Workspace ARM Toolchai
Workspace Build Tools Pa
Workspace OpenOCD Pat
Workspace pyOCD Path
Workspace QEMU Path
Workspace RISC-V Toolcl
Workspace SEGGER J-Lir
Mylyn
Oomph
Remote Development
Remote Systems
RPM
Run/Debug
Team
Terminal
Tracing
Validation
XML
Workspace OpenOCD Path
Configure the location where GNU MCU Eclipse OpenOCD is installed. The values are stored in the workspace and override the global paths. Unless redefined per project, they are used for all projects in this workspace.
After installing OpenOCD updates, restart Eclipse for the defaults to be re-evaluated and use the Restore Defaults button to configure the new location.
Executable:
OpenOCD
Folder:
/Users/test/Desktop/toolchain/gnu-mcu-eclipse/openocd/0.10.0-7-2018(
Browse...
xPack...
Restore Defaults
Apply
Cancel
Apply and Close

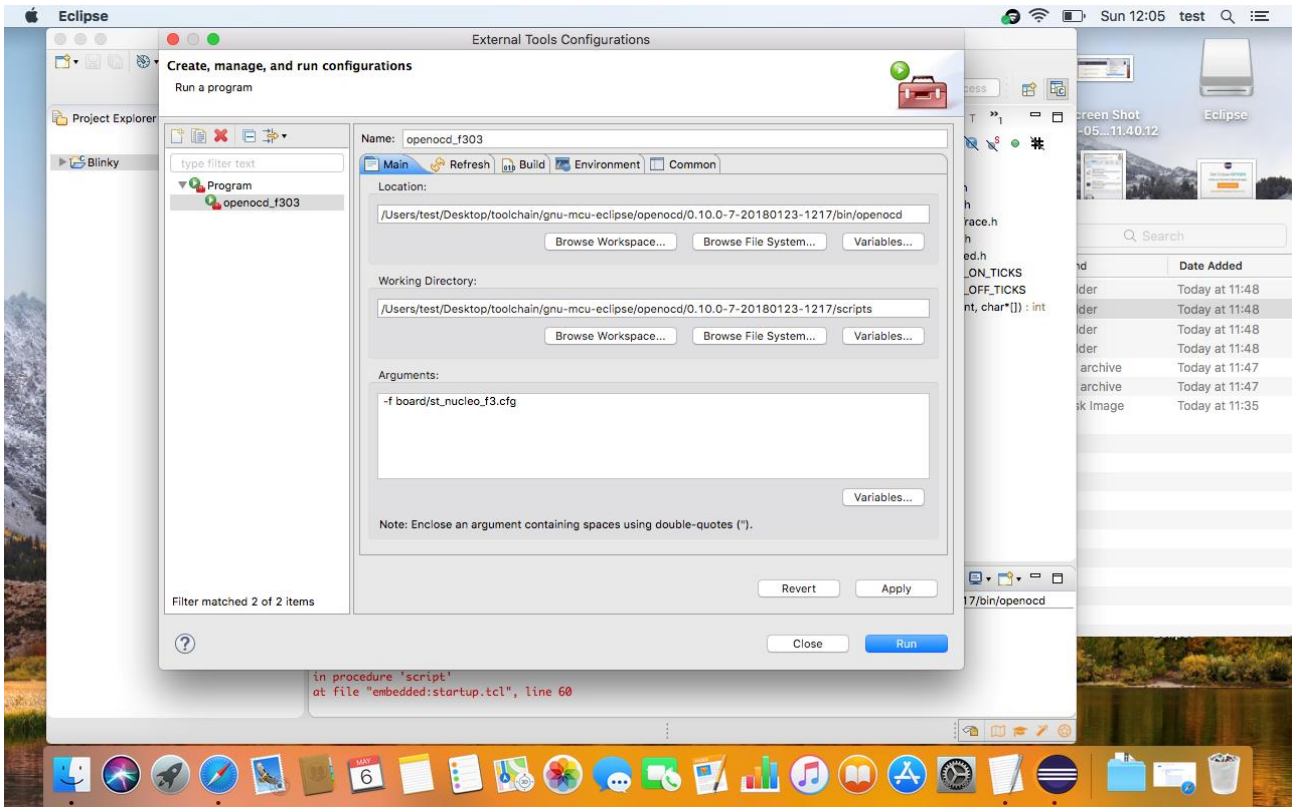
14) Navigate to Global ARM Toolchain Paths under MCU and enter the location of the GNU MCU Eclipse installation as shown below



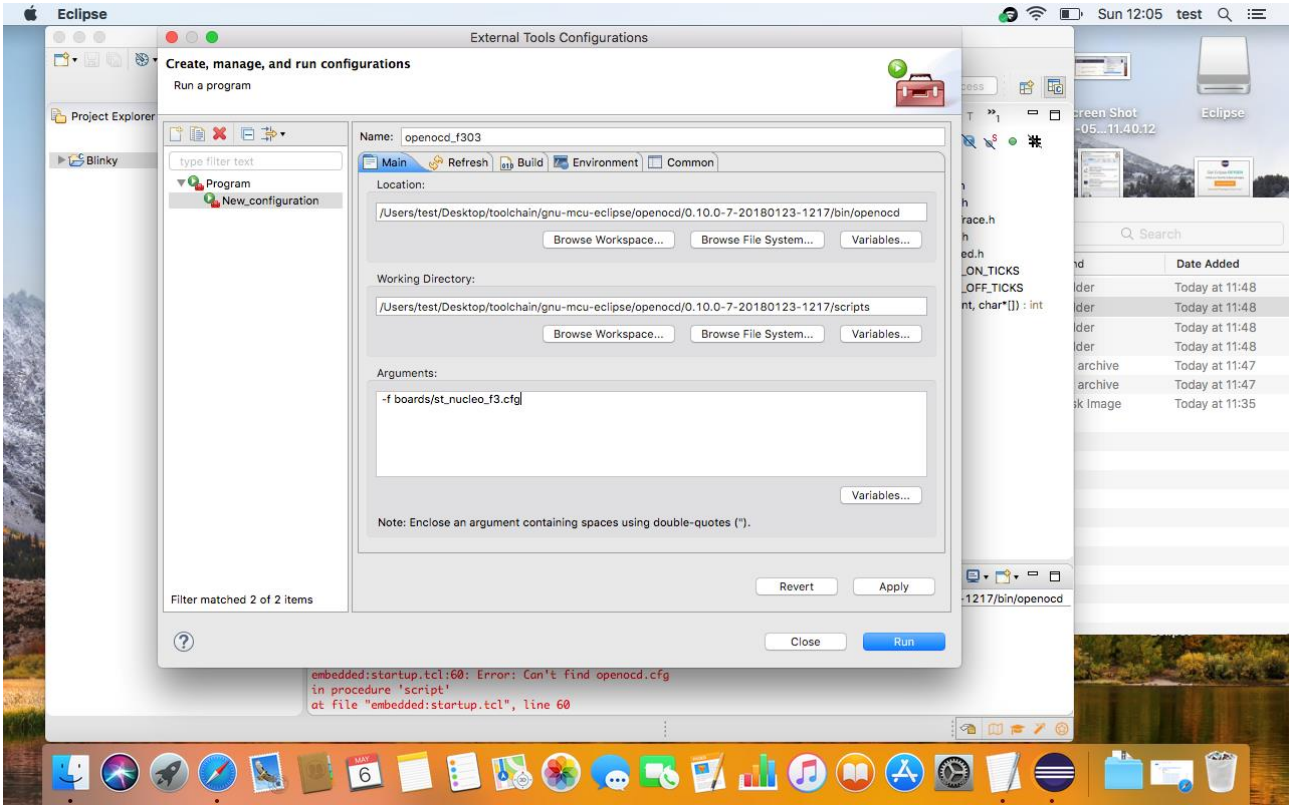
15) Navigate to Run -> External Tools -> External Tool Configurations



16) Add a new configuration using the icon on the top left

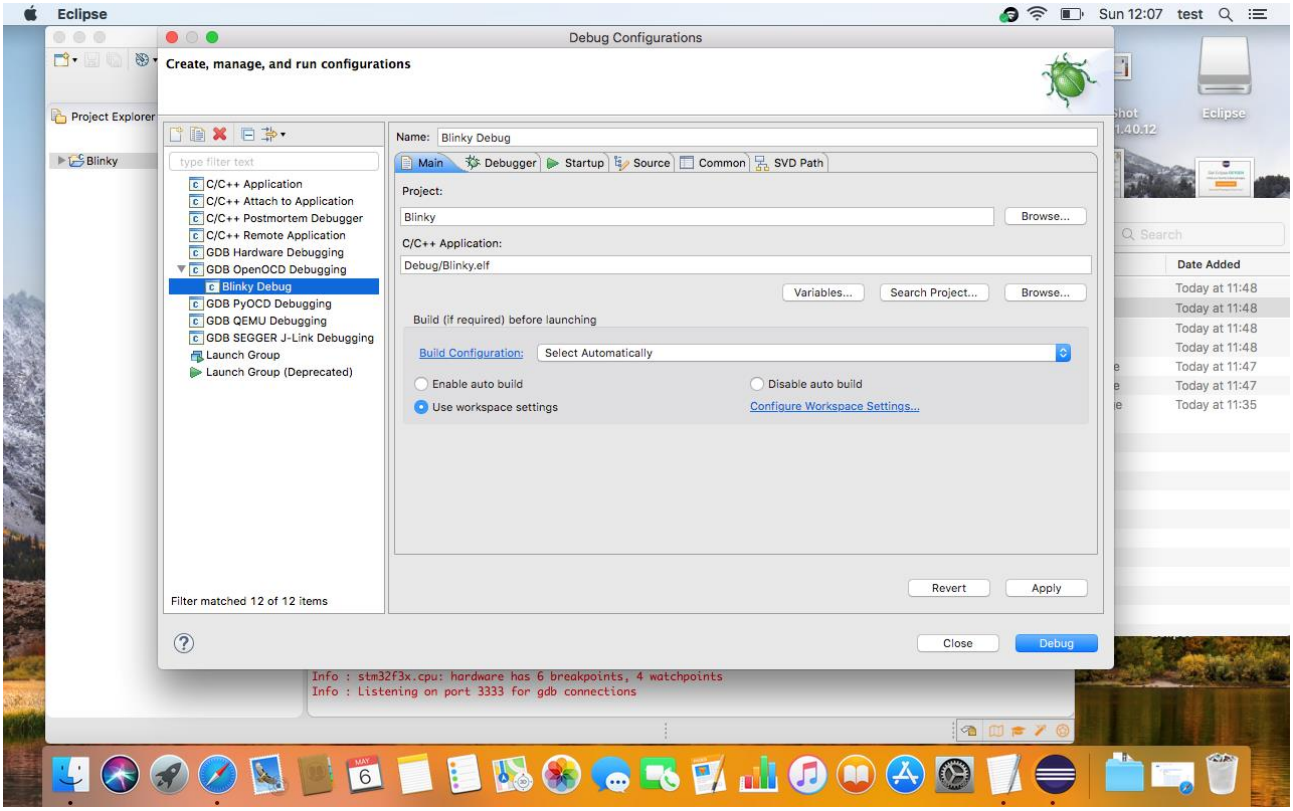


17) Enter the settings as shown below and hit Apply, then Close

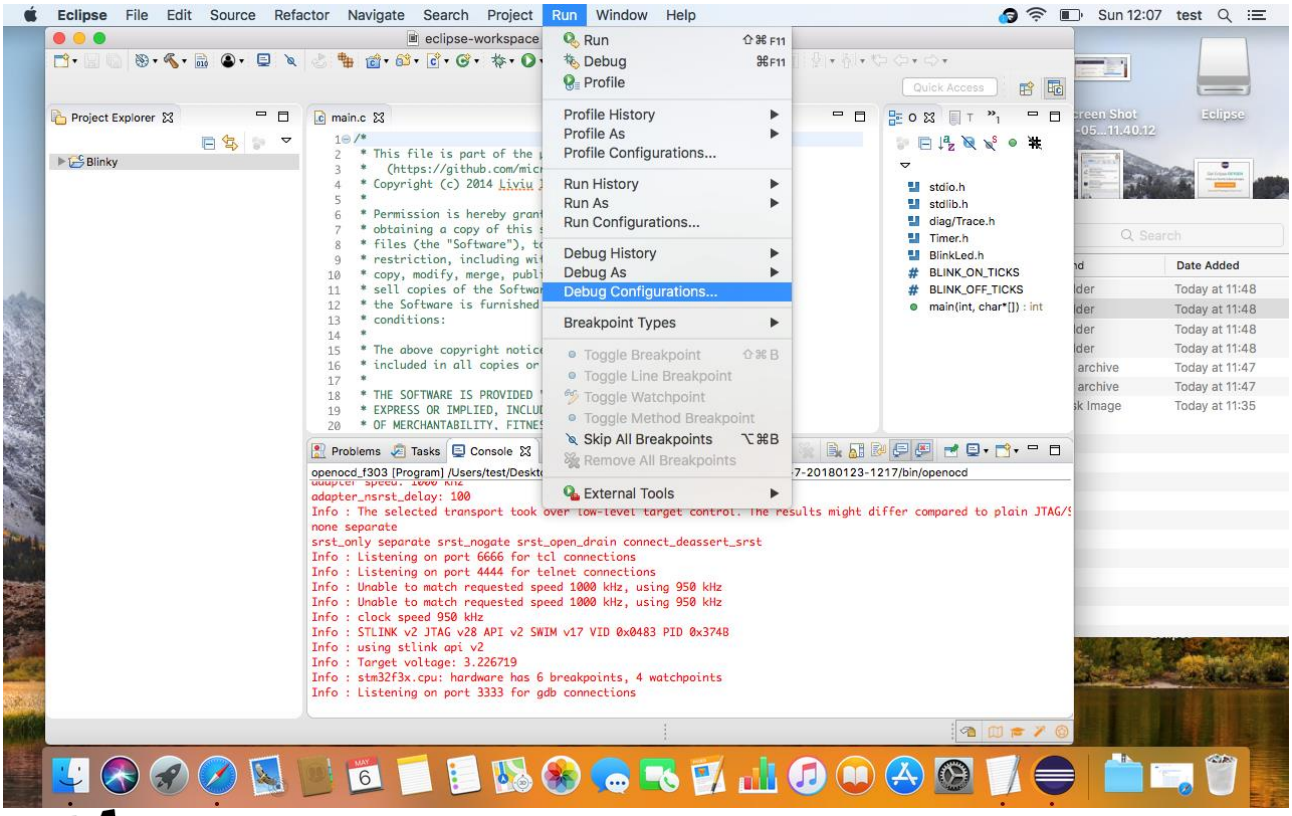


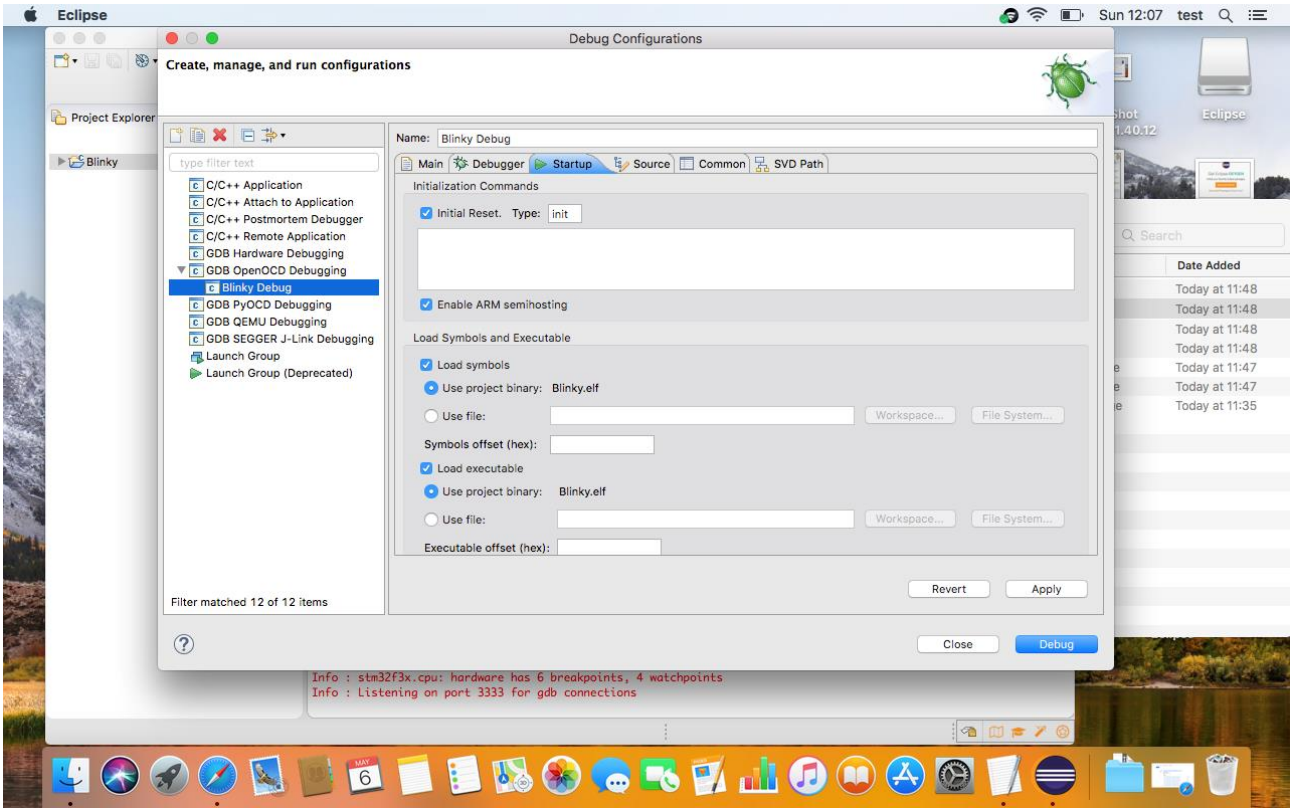
18) Click on Run -> Debug Configurations

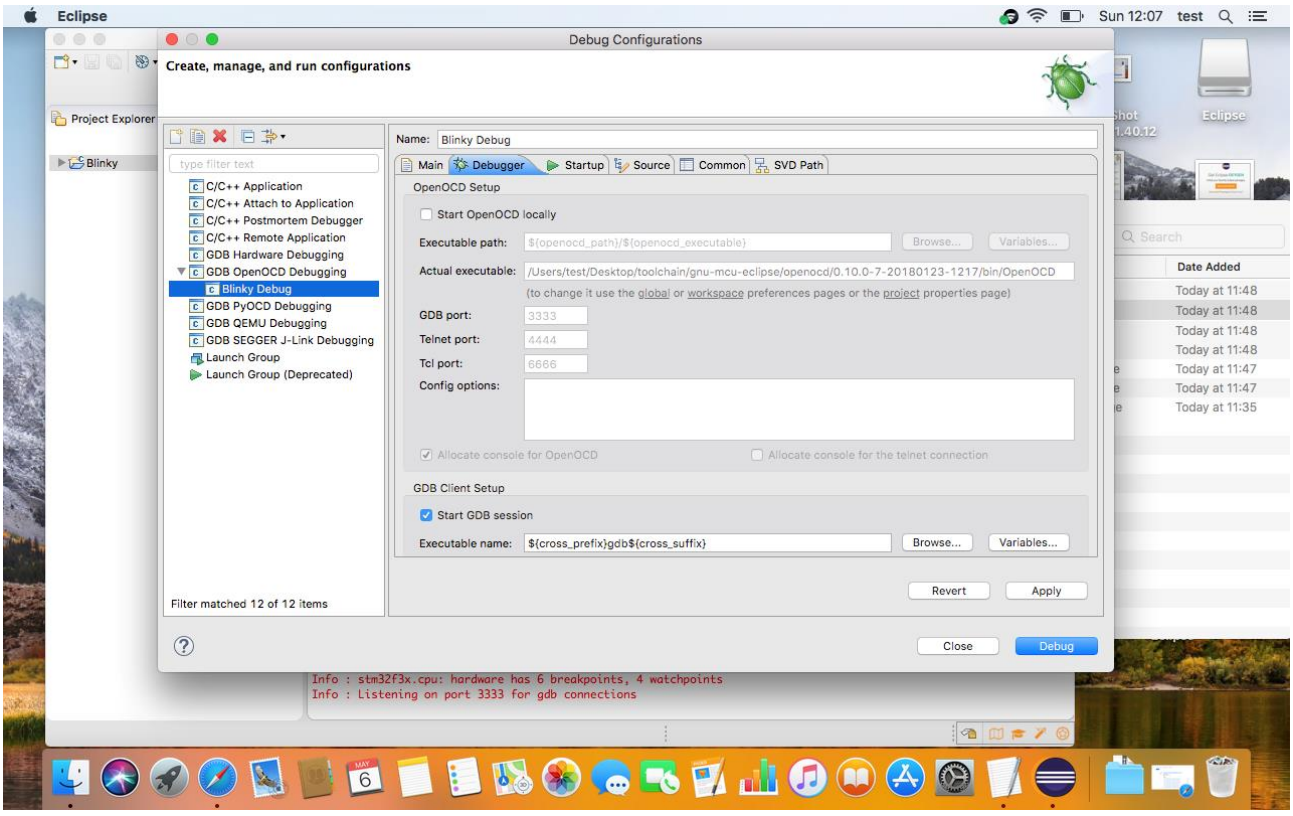
19) Click on GDB OpenOCD Debugging and Add a new configuration using the icon on the top left. Follow



the settings as shown below.

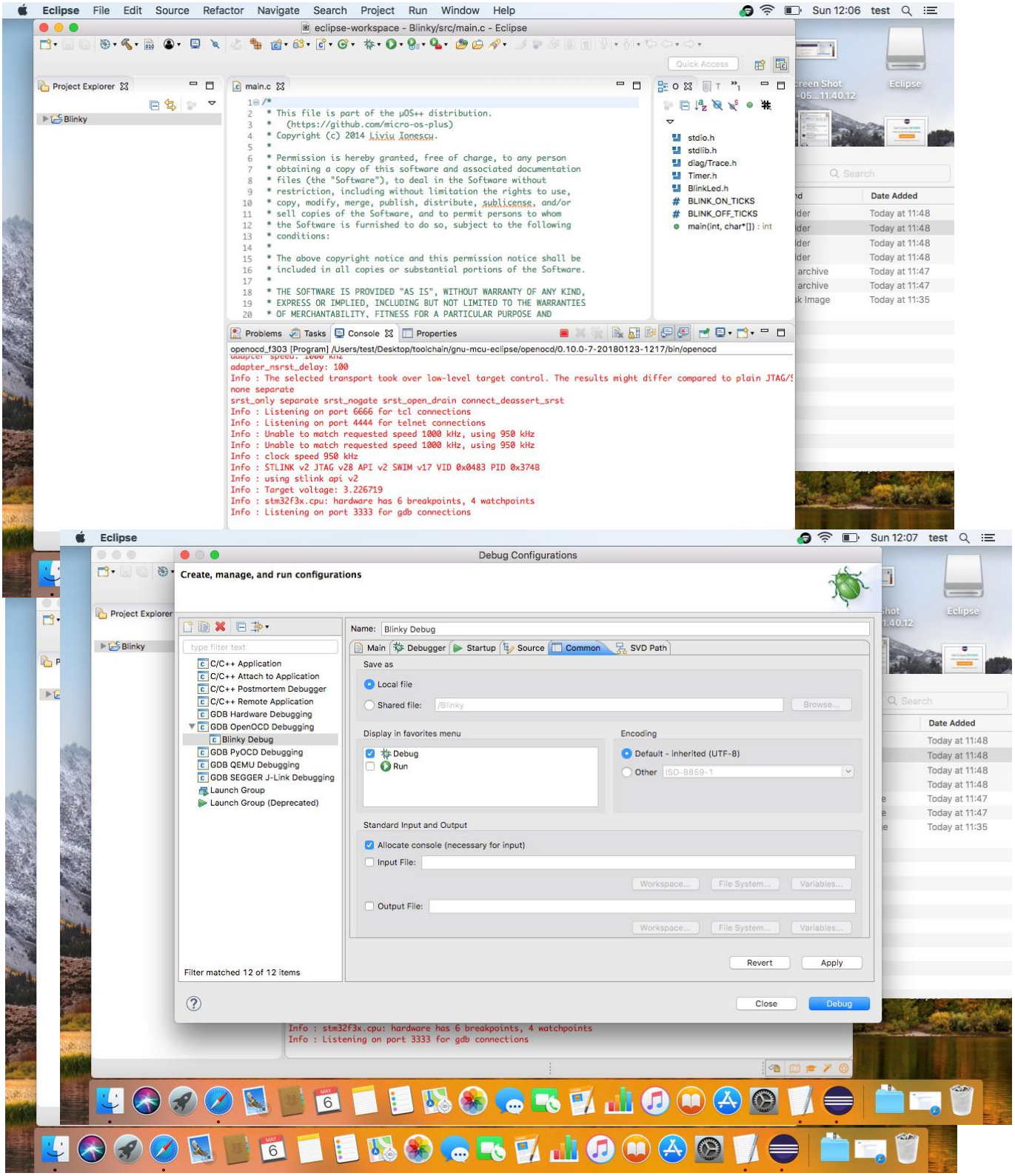


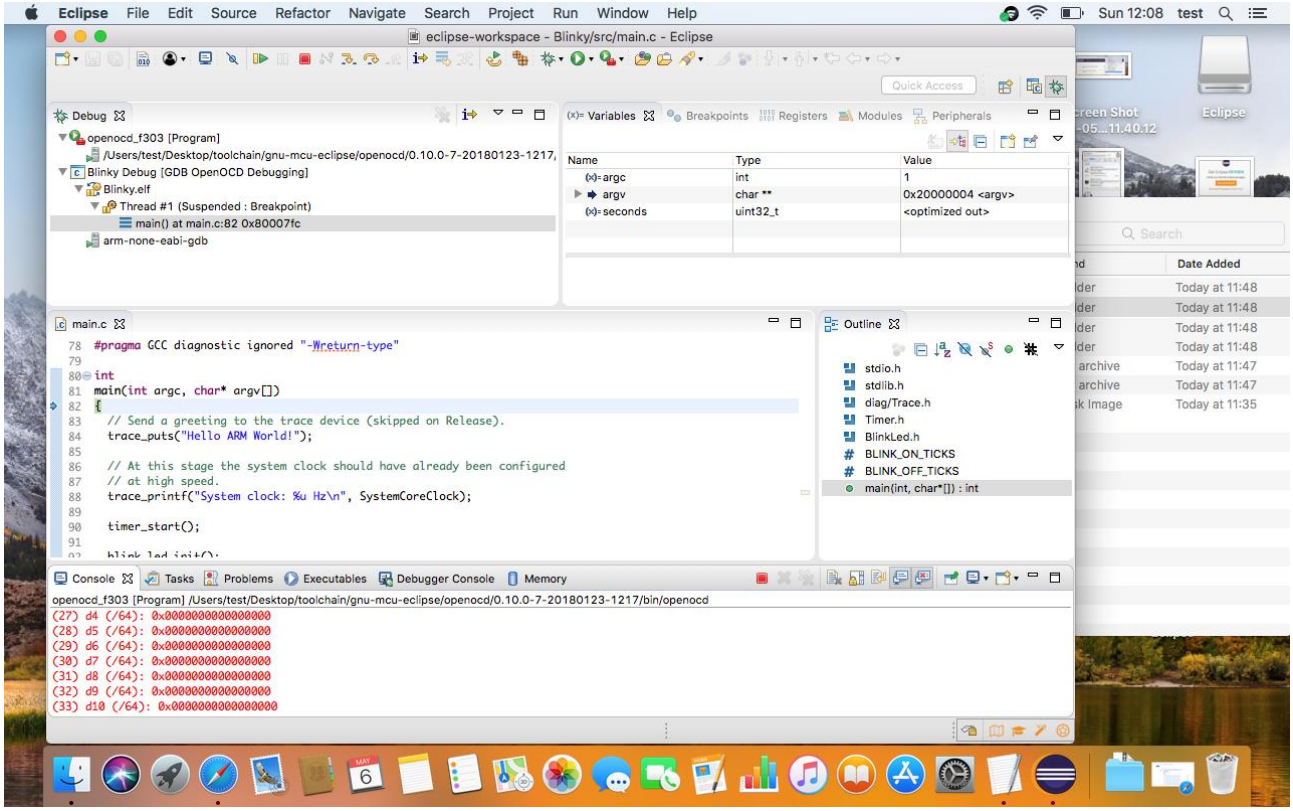




20) Hit Apply and Close

21) Start OpenOCD from Run -> External Tools -> OpenOCD as explained in course videos





22) Hit Allow

23) Start the Blinky Debug as shown in course videos. Your project should now be running in debug mode!