Environment Setup Guide

Major Steps

1. Download and Install Eclipse
2. Quickstart guide for iOBC
   1. JLink Drivers
   2. FTDI Drivers
   3. iOBC package sdk instal – sans eclipse
   4. Putty Settings
   5. JLink hardware connection
3. Clone the Repo – import to eclipse
4. Eclipse Settings and build DEBUG and RELEASE

Contents

[Download and Install Eclipse – Windows 10 Guide (64 bit) 3](#_Toc89352533)

[Install iOBC SDK elements 4](#_Toc89352534)

[Clone the Repo “Calacorm/MAAE4997B\_CDH” 4](#_Toc89352535)

[Setup Eclipse and import projects 4](#_Toc89352536)

# Download and Install Eclipse – Windows 10 Guide (64 bit)

1. Goto: <https://www.eclipse.org/downloads/>

Graphical user interface, application

Description automatically generated

Download the installer “Download x86\_64”, file: “eclipse-inst-jre-win64.exe”.

1. The File:



1. Install Eclipse IDE for Embedded C/C++ Developers to a location of your choosing and do not modify the VM settings.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

1. Once eclipse is installed don’t open it yet, we need to install the iOBC SDK elements.

# Clone the Repo

1. Clone “[owners\_username\_here]/MAAE4997B\_CDH” to a location of your choosing where you can find it again for project import in Eclipse.
2. I placed the cloned repo into the ISIS folder to keep everything tidy and in one spot.

# Install JLink from SEGGER – Adapted from ISIS QuickStart 3.1

1. Find in the repository, the folder “..\MAAE4997B\_CDH\Supplementary\_Content\Setup\_Files\JLinkSetup”
2. You may use the file to install JLink from SEGGER V7.54d for Windows 10 64-bit.
3. You may go to <https://www.segger.com/jlink-software.html> and download a newer version or different type. Please Document any changes if you get setup with a different version.
4. **NOTE: the version in the ISIS-OBC guide does not exist anymore (Lowest is V5.00), so we had to use new version and fortunately, it worked.**
5. **IMPORTANT: Note the path of installation. You will need to know the path to this: “..\SEGGER\JLink\JLinkGDBServerCL.exe”**

# Install FTDI Driver - Adapted from ISIS QuickStart 3.2

1. Find in the repository the folder “..\MAAE4997B\_CDH\Supplementary\_Content\Setup\_Files\FTDI\_Driver”
2. The versions will not change very much so the included file should be ok to use.
3. **File is for Windows (Desktop) – includes support for many versions of windows.**
4. Run the setup file “CDM212364\_Setup” to install driver for our debug USB adapter.
5. [https://ftdichip.com/drivers/vcp-drivers/](https://ftdichip.com/drivers/vcp-drivers/%20) Is the website, we used the ‘setup executable’ link on the right to get the setup wizard. Please select a different one if needed; document any changes.

# Install iOBC SDK elements - Adapted from ISIS QuickStart 3.3

1. Please follow the steps in this guide and not steps 1,2,3 in the ISIS-OBC QuickStart Guide. We have adapted the steps from the ISIS guide to modernize them and we use newer versions of drivers.
2. Acquire the “ISIS-OBC SDK.exe” file and install the elements included, **do not install Eclipse from here, it is very old, uncheck that as an install option**.
   1. This file is over half a gig and can not be stored on the repo and is not on the SVN. ***Consult Bruce - Project Lead, for a copy***.
3. This will install to C:/ISIS/ by default and setup the path environmental variables to point to toolchain elements in this folder.

# Setup Eclipse and import projects

1. Start up eclipse.
2. Create a workspace in a spot of your choosing, **don’t use the default workspace**, pick a name that makes sense. I chose “eclipse-workspace-iOBC-2021” and placed it in the ISIS folder to keep everything tidy and in one spot. You shouldn’t have to directly access workspace files so it could be placed wherever is convenient.
3. Close the welcome panel and select import from the left hand project explorer pane.
   1. *You may find import under File menu as well.*
4. From the list select General - Existing Projects into Workspace.
5. Browse for the root directory and locate the repo folder we cloned earlier.
6. The projects associated with the repo will be listed select what you want.
   1. At time of writing there is only one project: “hal-demo”
7. All options should be unchecked and just finish to import the project.
8. Expand the project and includes folder in the project explorer.

Graphical user interface, text, application

Description automatically generated

Figure 1 - hal-demo project shown for example

Validate that the /ISIS/application/toochain/ paths are there as well as the hal includes. More may be there for future projects and other libraries. Expect the toolchains includes to stem from the ISIS folder (from SDK install), and the others are stemming from where the project repo is located. **See troubleshooting if the includes are not there. Please update image and documentation if this changes.**

1. Under the project folder find and open the file “Jlink.launch”.
2. Change the following line (line 7) to point to the **JLinkGDBServerCL** file:

<stringAttribute key="org.eclipse.ui.externaltools.ATTR\_LOCATION" value="C:\Program Files\SEGGER\JLink\JLinkGDBServerCL.exe"/>

**NOTE: your path may differ from the one shown in the above example!**

1. Under the project folder find and right click on “Jlink.launch” and select “Run As” then select “1 JLink”. A window will pop up since we don’t have the hardware connected, just click cancel. This should be done for each project (if you have other iOBC projects) you would like to run via the debug interface; a Jlink.launch file will be required.

Text

Description automatically generated

Figure 2 – hal-demo

1. The purpose of this is to add the JLink launch configuration for easy access later on this button.



Figure 3 – eclipse external tools button and drop down menu

1. Attempt to build the project with the Hammer icon, beside the hammer icon is a drop down that lets you select the build configuration: debug or release. Try doing both to validate setup has worked.

## Running the code on iOBC Hardware

**The following steps require iOBC hardware, JLink Usb Adapter, mounting stand, 3.3 V Ac adapter. If this is your first lab with the hardware, please ensure to have Project Lead or Team Lead support you for proper setup and ESD safety procedures.**

1. Follow the **“ISIS-OBC QuickStart Guide v2.2”** [**Step 4**] to hookup the JLink debug link to your computer.
2. Start Putty in the C:/ISIS/applications/PuTTy/ Folder. If you put the ISIS files someplace else, please find the applications folder to locate putty, you may attempt to use a more modern version of putty if you wish. *Document the use of a modern version of PuTTy if you do.*
3. **Follow section 7.1.2 of the iOBC quick start guide** to setup the putty COM port then start putty up so the terminal window is open.
4. In Eclipse, launch JLink from the button mentioned in step 12.
5. Verify no errors in the console output, the JLink is now armed and waiting.
6. To run debug mode, right click “hal-demo-debug.launch” and Debug As and select the option there. This will add the debug launch configuration and can be accessed via the shortcut from here on. Debug will start and the code will run on the iOBC, PuTTY console will display console output. Eclipse will ask to switch to debug mode where breakpoints may be used.