



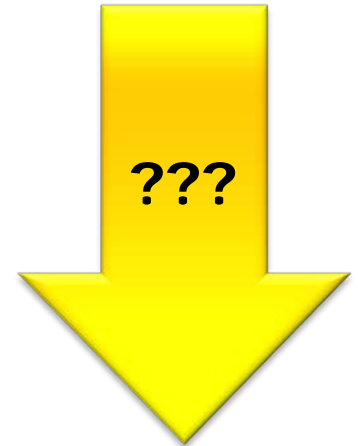
# Build and Debug

*"Compile, link, debug – then repeat..."*

**Prof. Erich Styger**  
[erich.styger@hslu.ch](mailto:erich.styger@hslu.ch)  
+41 41 349 33 01

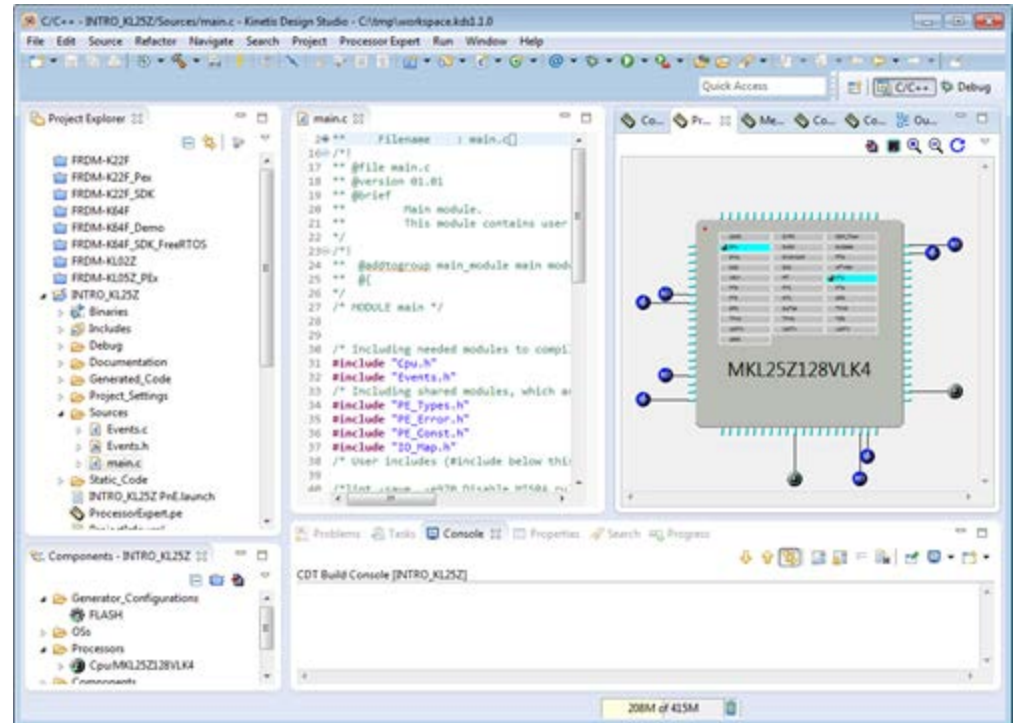
# Learning Goals

- Problem: Need Tools
- Lab Kit & Tools
- Eclipse
- Boards
- Creating/Build the first project
- Download/Debug



# Default Lab Kits & Tools

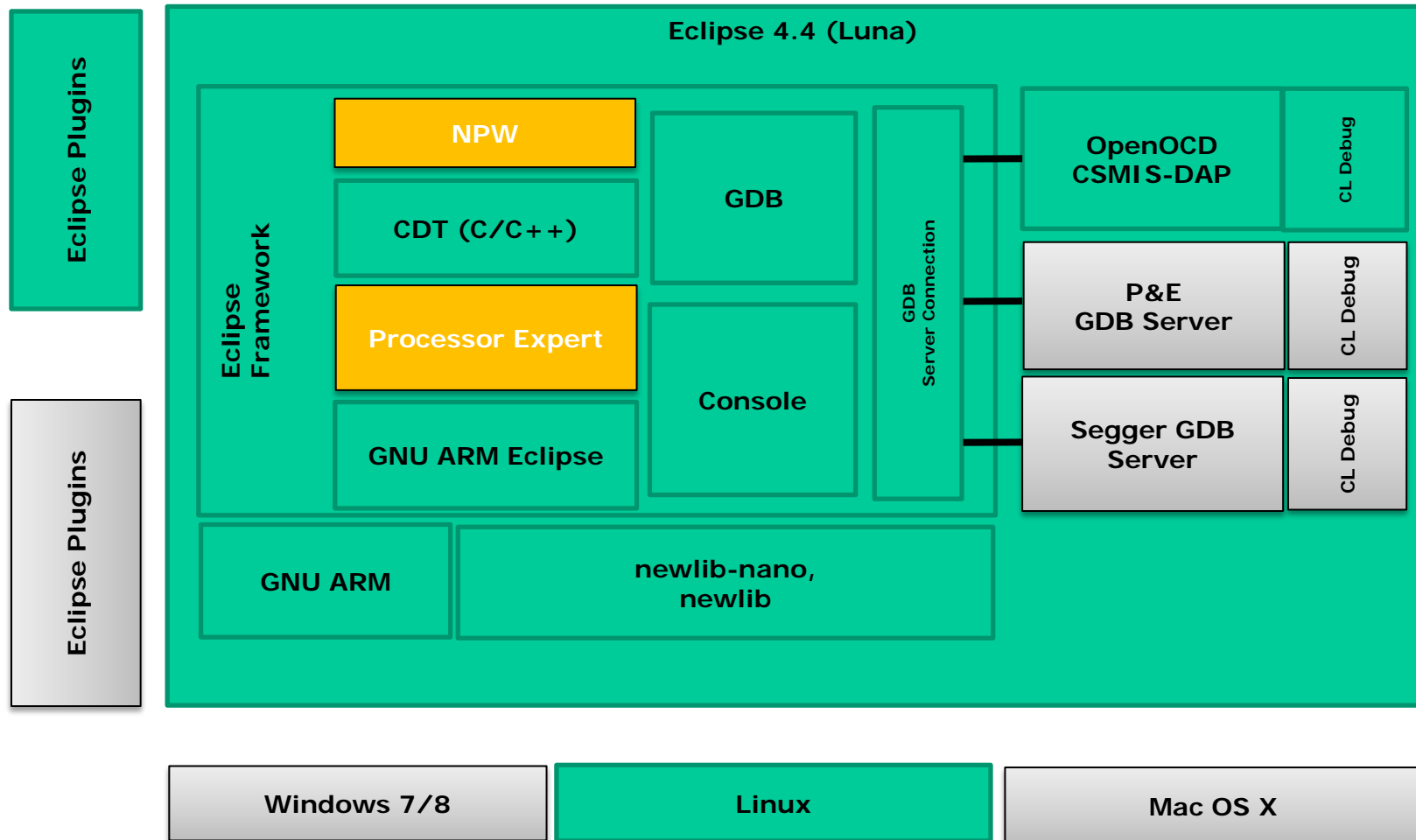
- Kinetis Design Studio
  - Processor Expert
- GIT VCS (Git)
- Remote Board
- Robot Board



# Lab Philosophy

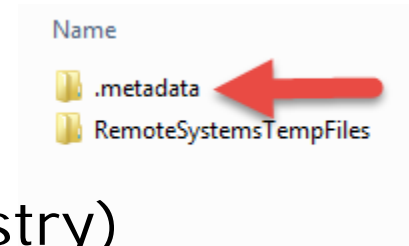
- You need to maintain your project!
- Multiple projects: Remote & Robo
  - Remote: **K20DX128** (ARM Cortex-M4)
  - Robo: **K22FX512** (ARM Cortex-M4F)
- Processor Expert Components
  - Low level functionality
  - Common Utilities
- Your own project
  - Start with new/empty project
  - Add your sources
  - Add/copy/enable instructor files and components

# Kinetis Design Studio



# Eclipse Workspace

- Where Eclipse stores the 'meta data'
  - 'global' options across projects
  - NEVER move/share meta data (is like registry)
- Never share a wsp with other Eclipse versions/products
  - Use separate workspaces
- Wsp *could* contain project folders, but
  - Recommendation: keep wsp and projects separate
  - Easier for VCS

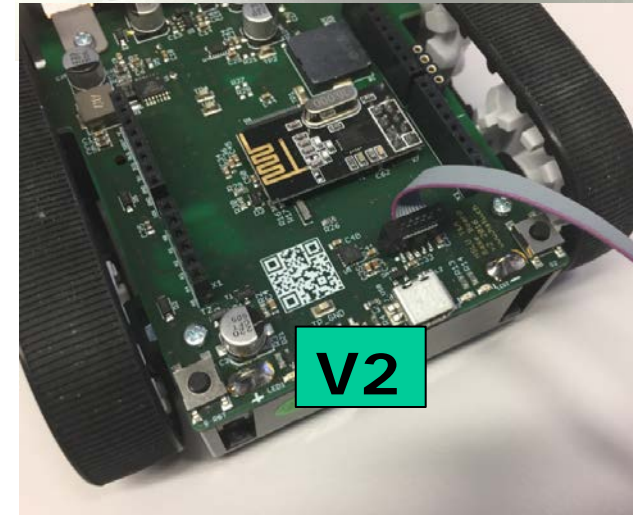
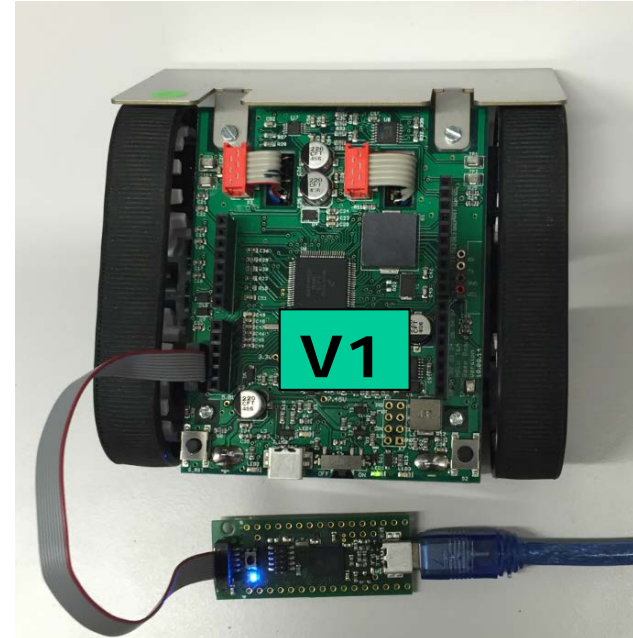


C:\user\dev\wsp\_kds  
.metadata

C:\user\projects  
Project1  
Project2  
...

# Robo: Power Debug Connection

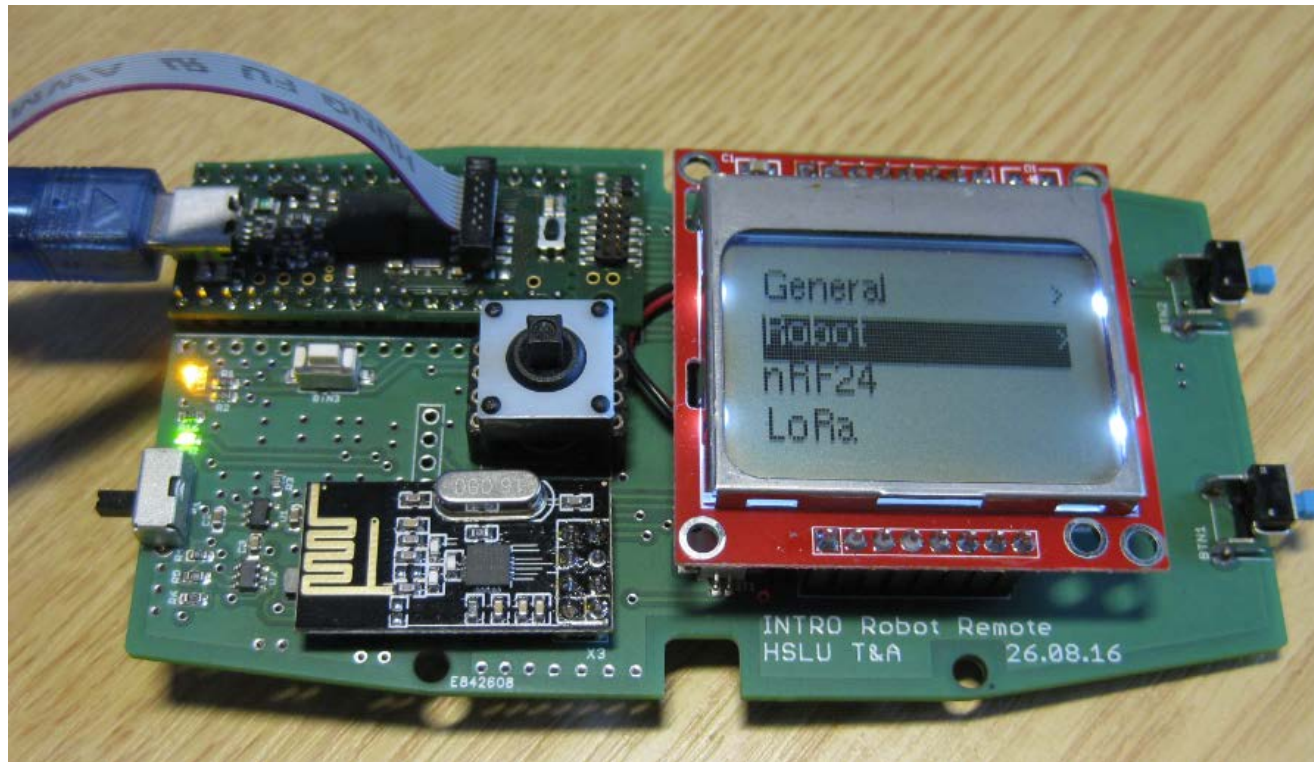
- Microcontroller Power
  - USB
  - Batteries
- Motor Power
  - Batteries
- Debug (SWD/JTAG)
  - **tinyK20**
  - Red stripe: Pin 1!
  - Optional:
    - P&E Multilink
    - Segger J-Link





# INTRO Robot Remote

- Mikrocontroller: tinyK20
- USB: Power and battery charging
- Debug: SWD (e.g. tinyK20)





# Bootloader/Windows 10 Problem

- Windows 10 writes to MSD device
- Can cause bootloader corruption ☹️
- Solution (see article): Change Group Policy order
  - **DAPLink** bootloader
- Problem if target is resetting
  - Exchange tinyK20 in that case
- [https://mcuoneclipse.com/2016/08/01/bricking\\_and\\_recovering\\_opensda\\_boards\\_in\\_windows\\_8\\_and\\_10/](https://mcuoneclipse.com/2016/08/01/bricking_and_recovering_opensda_boards_in_windows_8_and_10/)

## Bricking and Recovering OpenSDA Boards in Windows 8 and 10

★★★★★ 10 Votes

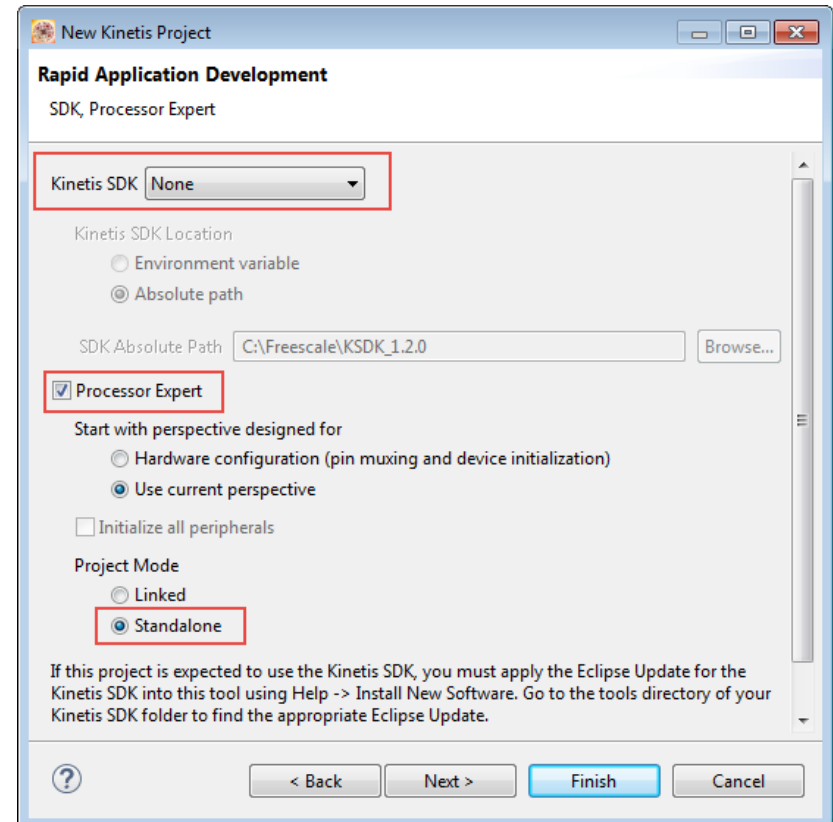
Getting a board from a distributor like Farnell/Element14/Mouser (add your own distributor) means that chances are high that the default firmware on it is written years from now because the inventory has not been updated, or because boards are still produced with that original firmware (because of testing?). So what happens if I use board with a firmware developed pre-Windows 8/10 area?



— Freshly Unboxed NXP FRDM-KL25Z Board

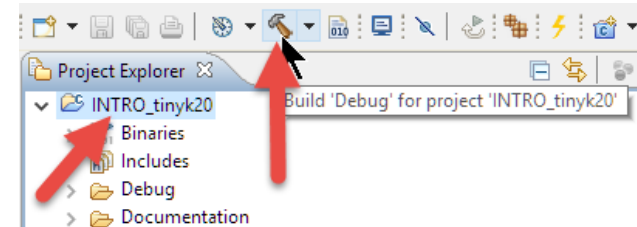
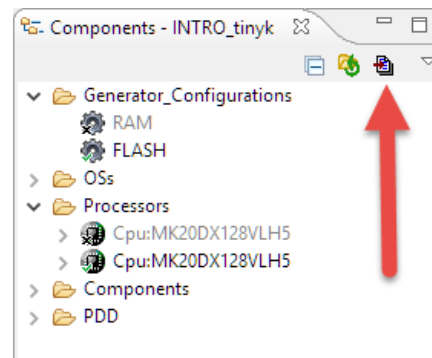
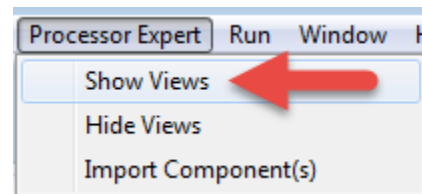
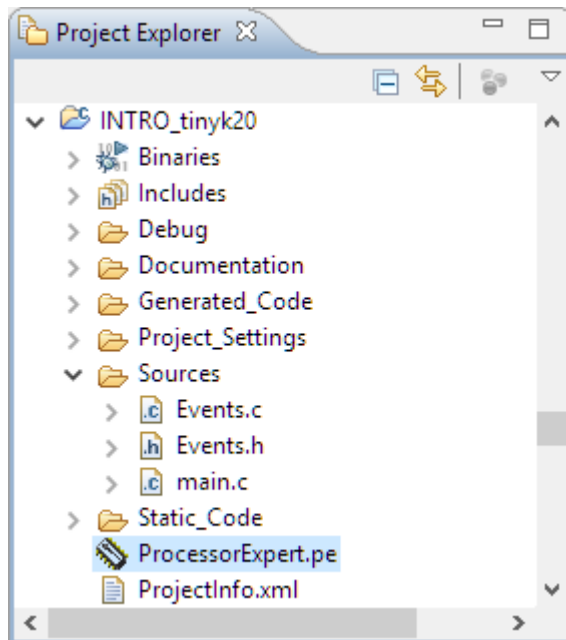
# Creating the Project(s) (tinyK20)

- Menu *File > New > Kinetis Project*
- Provide Project Name
- Device: Kinetis K, MK20, ➔ **MK20DX128xxx5**
- No SDK, Processor Expert
- GNU C Compiler



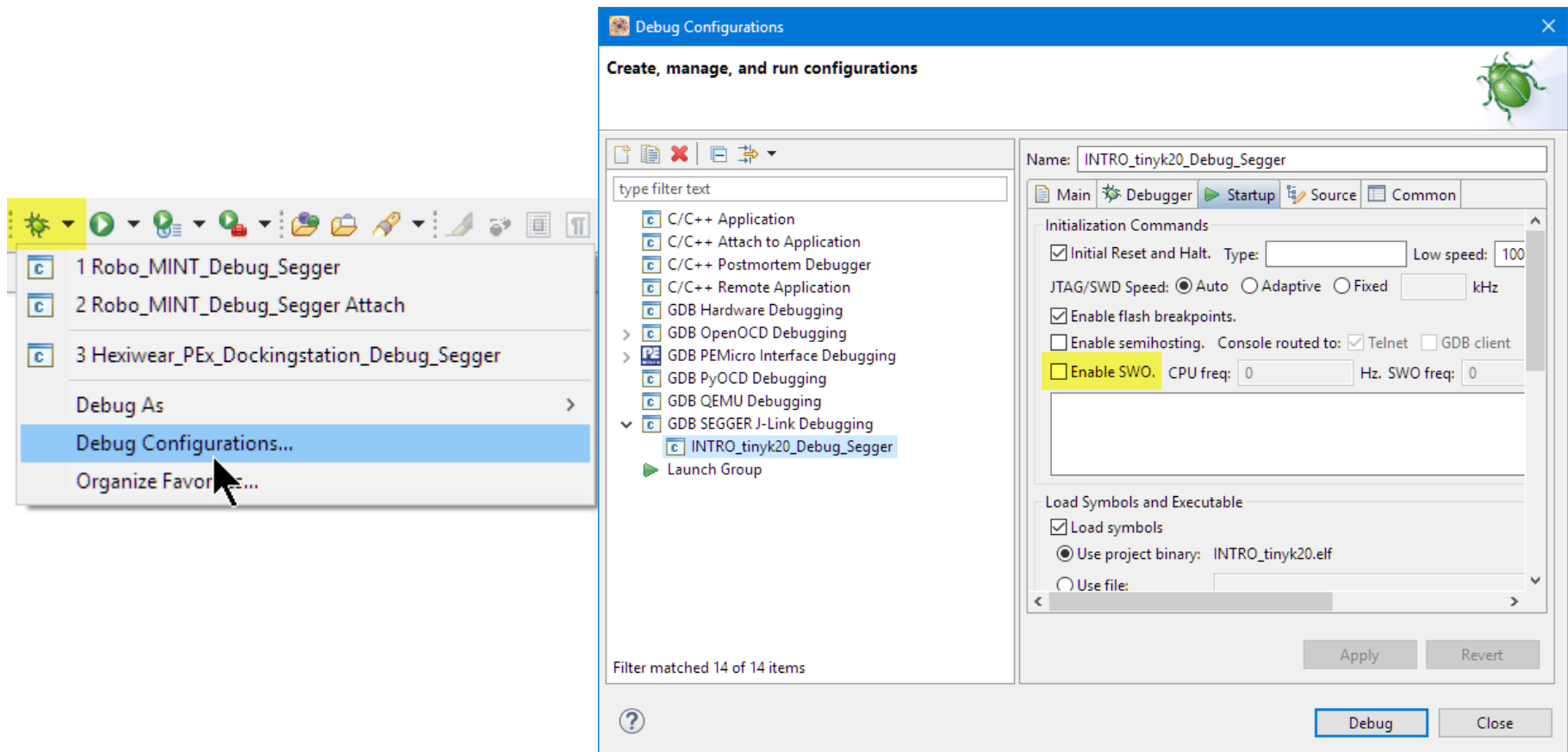
# Generating and Building

- ProcessorExpert.pe XML file
- *Processor Expert > Show Views*
- 'Generate code' button
- **Generated\_Code** folder
- Select project and build



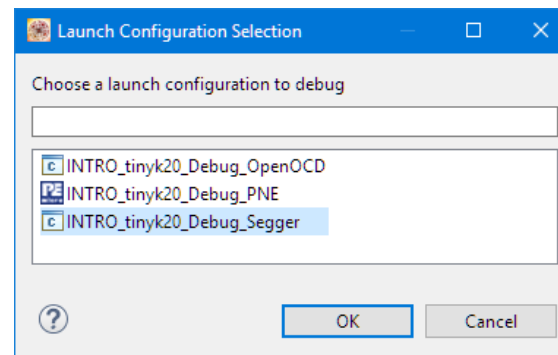
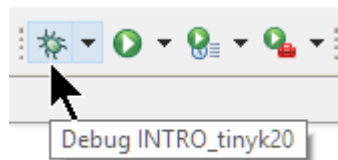
# Segger J-Link Debugger

- Disable SWO
- Glitch in UI handling

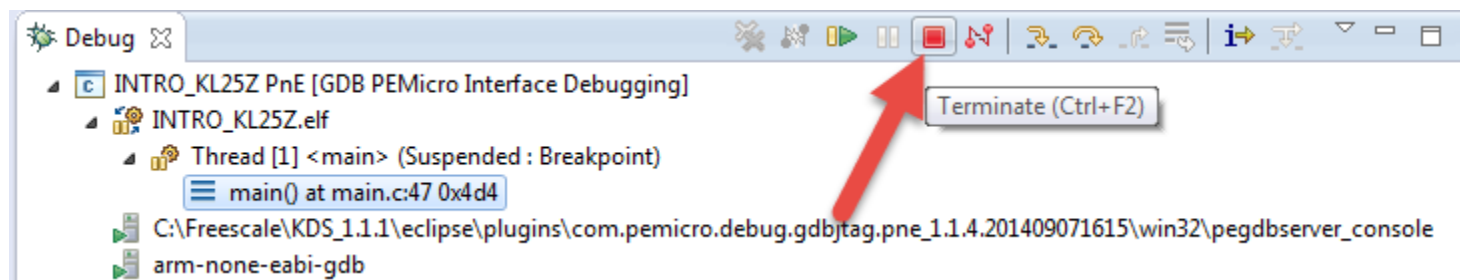


# GDB Debugging

- Select Project, then use drop down entry
  - or: verify correct project shown in tooltip



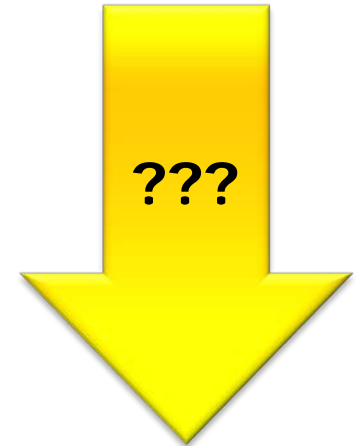
- Always terminate debug session before starting new one



# Summary

- *Problem: Need development tools*

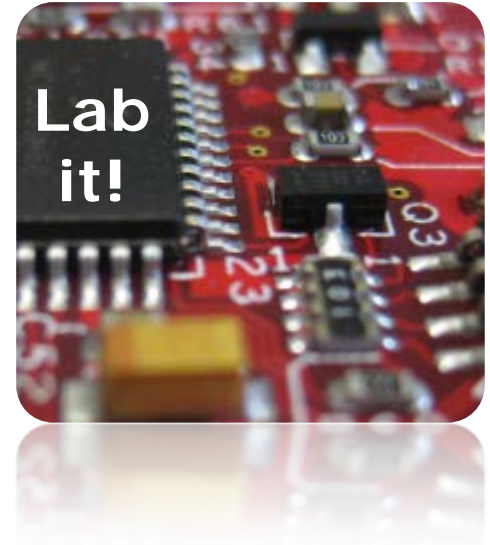
- Eclipse
  - Build
  - Debug
- Hardware/Boards
  - Robo
  - Remote
- General System Setup





## Lab 4: First Projects

- (Installation of Kinetis Design Studio)
- Create new project with wizard
  - Processor Expert project
  - Robo: K22FX512
  - Remote: MK20DX128
- Build, download, debug
- Tips
  - Parallel Build
  - SWO



Lab #4