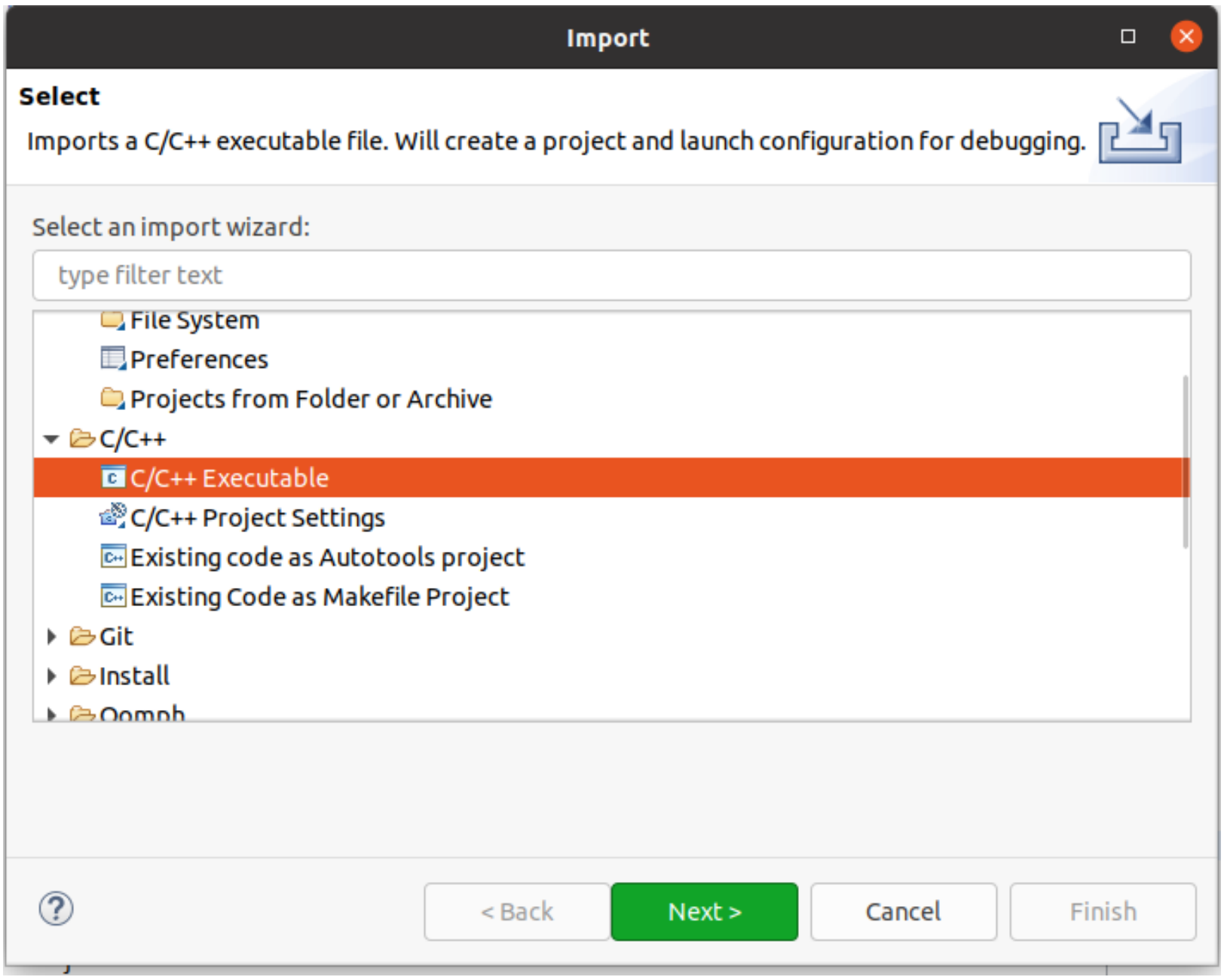
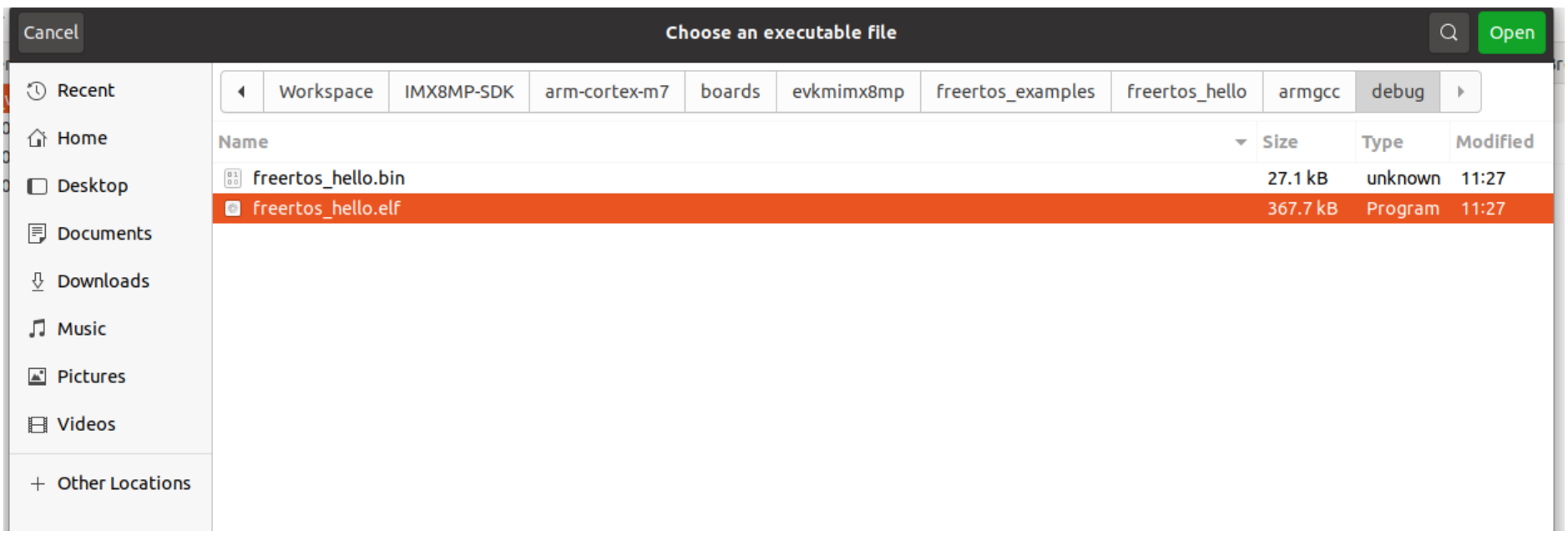
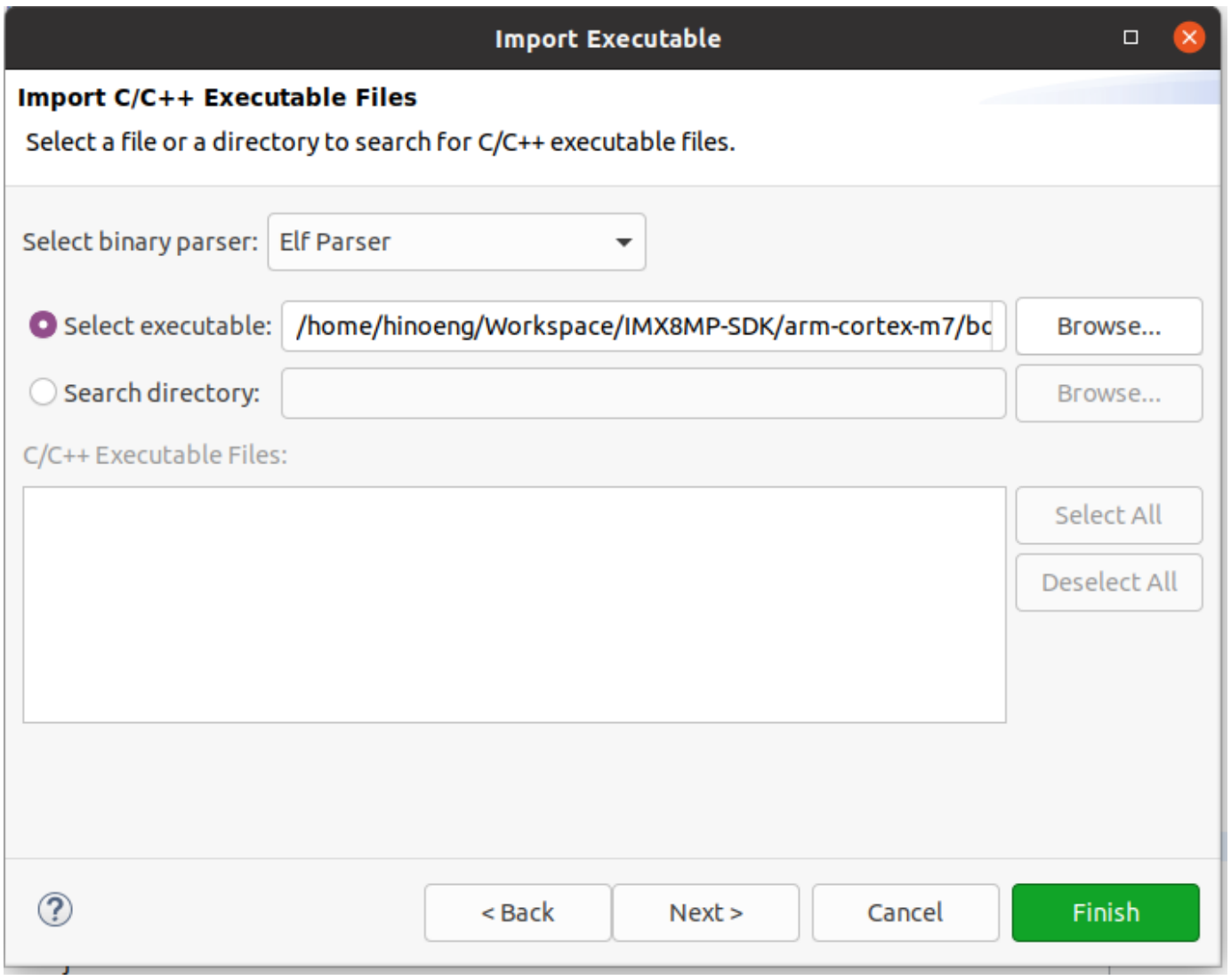
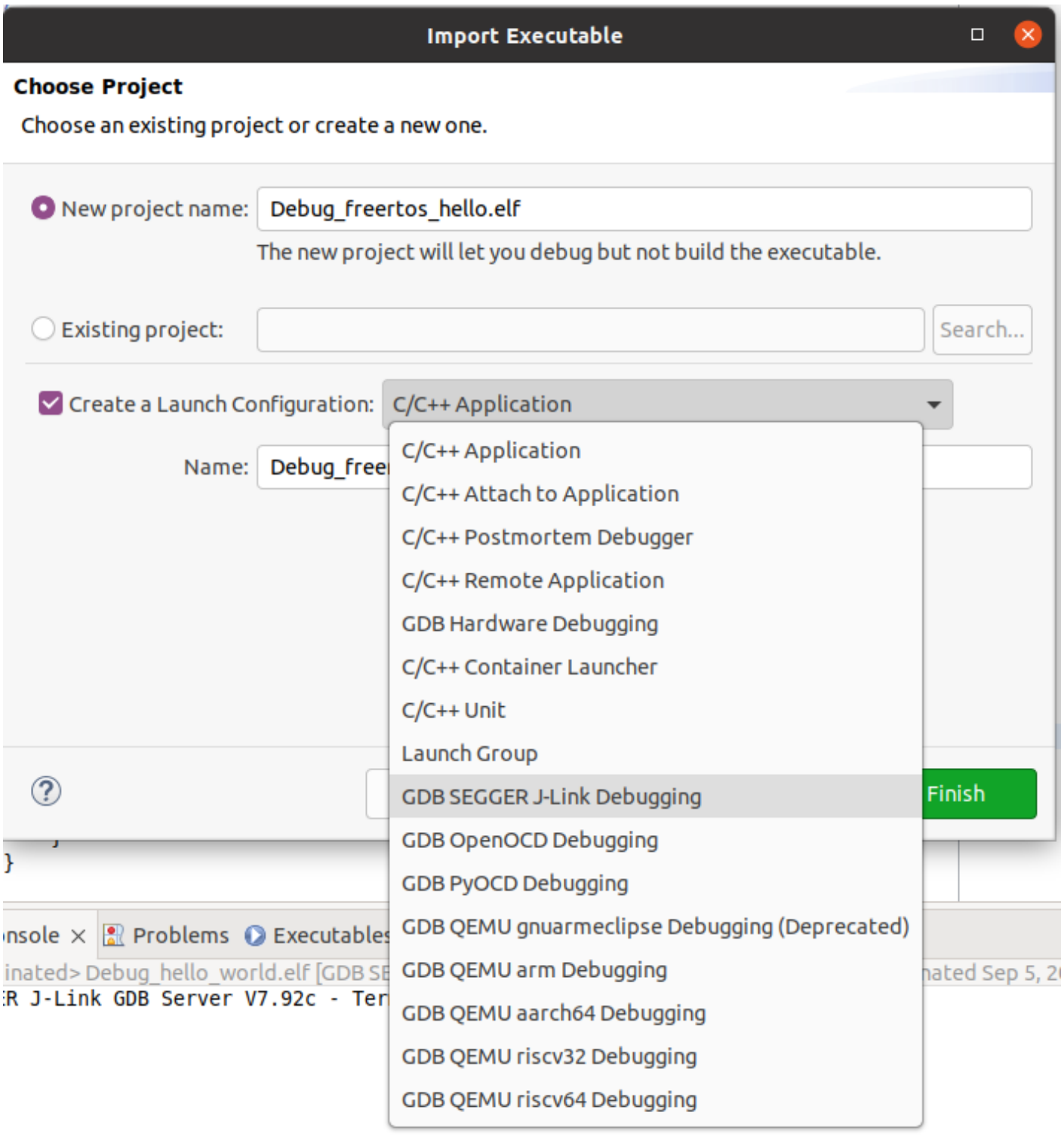

Import
Select
Imports a C/C++ executable file. Will create a project and launch configuration for debugging.
Select an import wizard:
type filter text
File System
Preferences
Projects from Folder or Archive
C/C++
C/C++ Executable
C/C++ Project Settings
Existing code as Autotools project
Existing Code as Makefile Project
Git
Install
Oomph
< Back
Next >
Cancel
Finish


Cancel
Choose an executable file
Open
Recent
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
Other Locations
Workspace
IMX8MP-SDK
arm-cortex-m7
boards
evkmimx8mp
freertos_examples
freertos_hello
armgcc
debug
Name
Size
Type
Modified
freertos_hello.bin
27.1 kB
unknown
11:27
freertos_hello.elf
367.7 kB
Program
11:27


Import Executable
Import C/C++ Executable Files
Select a file or a directory to search for C/C++ executable files.
Select binary parser:
Elf Parser
Select executable:
/home/hinoeng/Workspace/IMX8MP-SDK/arm-cortex-m7/bc
Browse...
Search directory:
Browse...
C/C++ Executable Files:
Select All
Deselect All
< Back
Next >
Cancel
Finish


Import Executable
Choose Project
Choose an existing project or create a new one.
New project name:
Debug_freertos_hello.elf
The new project will let you debug but not build the executable.
Existing project:
Search...
Create a Launch Configuration:
C/C++ Application
Name:
Debug_free
C/C++ Application
C/C++ Attach to Application
C/C++ Postmortem Debugger
C/C++ Remote Application
GDB Hardware Debugging
C/C++ Container Launcher
C/C++ Unit
Launch Group
GDB SEGGER J-Link Debugging
GDB OpenOCD Debugging
GDB PyOCD Debugging
GDB QEMU gnuarmeclipse Debugging (Deprecated)
GDB QEMU arm Debugging
GDB QEMU aarch64 Debugging
GDB QEMU riscv32 Debugging
GDB QEMU riscv64 Debugging
Finish
}
nsole
Problems
Executables
inated> Debug_hello_world.elf [GDB SE
nated Sep 5, 2
R J-Link GDB Server V7.92c - Ter

Eclipse - /home/hinoeng/Workspace/IMX8MP-SDK/arm-cortex-m7/boards/evkmimx8mp/freertos_example

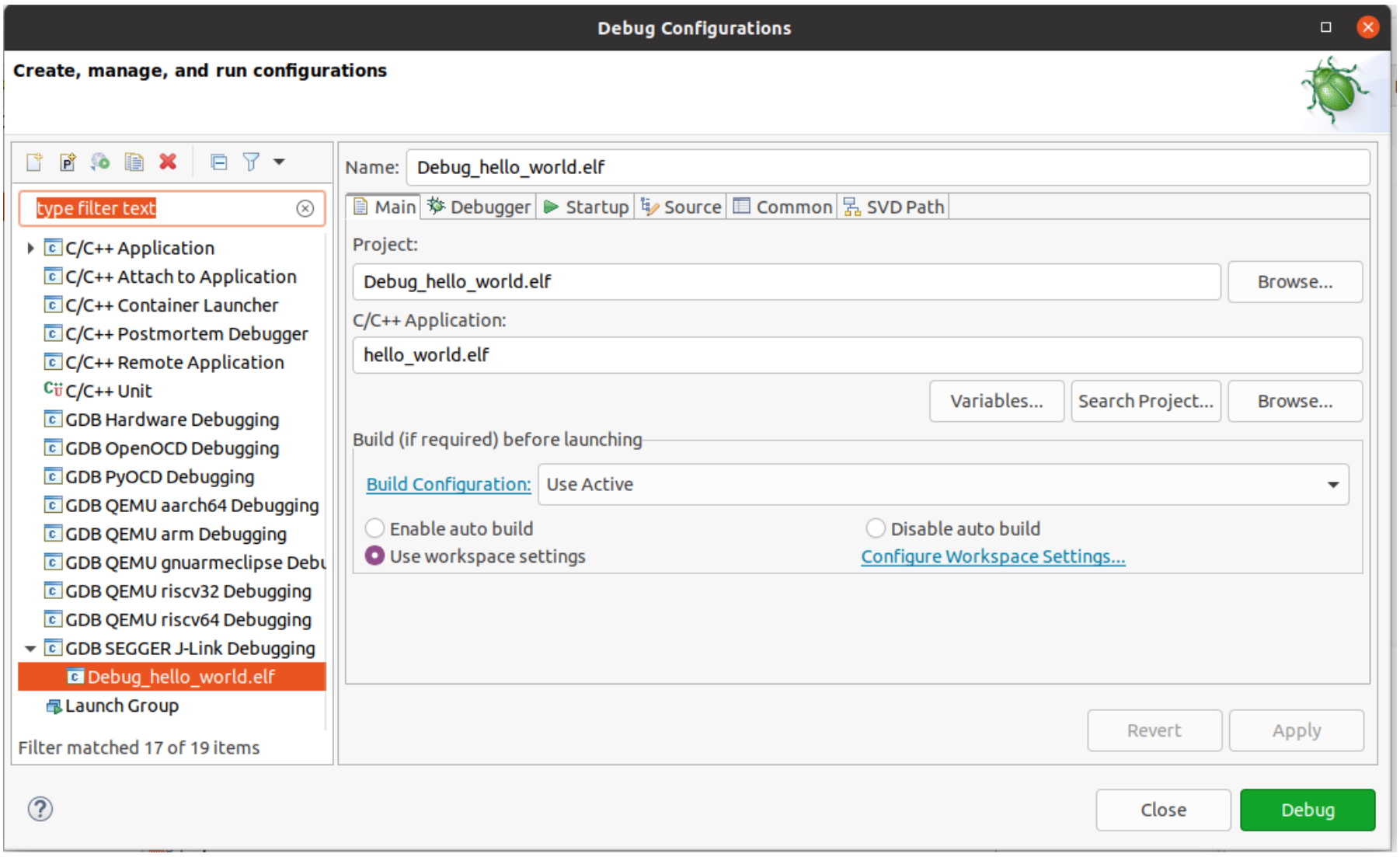
File Edit Source Refactor Navigate Search Project Run Window Help

Debug × Project Explorer

- Debug_freertos_hello.elf [GDB SEGGER J-Link Debugging]
  - freertos_hello.elf
    - Thread #1 57005 (Suspended : Breakpoint)
      - main() at freertos_hello.c:45 0x136a
  - JLinkGDBServer
  - arm-none-eabi-gdb
  - Semihosting and SWV

hello_world.c | fsl_uart.c | board.c | freertos_hello.c ×

```
28 /*******************************************************************************
29  * Prototypes
30  ******************************************************************************/
31 static void hello_task(void *pvParameters);
32
33 /*******************************************************************************
34  * Code
35  ******************************************************************************/
36 /*!
37  * @brief Application entry point.
38  */
39 int main(void)
40 {
41     /* Init board hardware. */
42     /* M7 has its local cache and enabled by default,
43      * need to set smart subsystems (0x28000000 ~ 0x3FFFFFFF)
44      * non-cacheable before accessing this address region */
45     BOARD_InitMemory();
46
47     /* Board specific RDC settings */
48     BOARD_RdcInit();
49
50     BOARD_InitBootPins();
51     BOARD_BootClockRUN();
52     BOARD_InitDebugConsole();
53     if (xTaskCreate(hello_task, "Hello_task", configMINIMAL_STACK_SIZE + 100
54         pdPASS)
55     {
56         PRINTF("Task creation failed!.\r\n");
57         while (1)
58             ;
59     }
60     vTaskStartScheduler();
61     for (;;)
62         ;
63 }
```


Debug Configurations
Create, manage, and run configurations
type filter text
C/C++ Application
C/C++ Attach to Application
C/C++ Container Launcher
C/C++ Postmortem Debugger
C/C++ Remote Application
C/C++ Unit
GDB Hardware Debugging
GDB OpenOCD Debugging
GDB PyOCD Debugging
GDB QEMU aarch64 Debugging
GDB QEMU arm Debugging
GDB QEMU gnuarmeclipse Debu
GDB QEMU riscv32 Debugging
GDB QEMU riscv64 Debugging
GDB SEGGER J-Link Debugging
Debug_hello_world.elf
Launch Group
Filter matched 17 of 19 items
Name: Debug_hello_world.elf
Main
Debugger
Startup
Source
Common
SVD Path
Project:
Debug_hello_world.elf
Browse...
C/C++ Application:
hello_world.elf
Variables...
Search Project...
Browse...
Build (if required) before launching
Build Configuration:
Use Active
Enable auto build
Disable auto build
Use workspace settings
Configure Workspace Settings...
Revert
Apply
Close
Debug

Debug Configurations

Create, manage, and run configurations

type filter text

- C/C++ Application
- C/C++ Attach to Application
- C/C++ Container Launcher
- C/C++ Postmortem Debugger
- C/C++ Remote Application
- C/C++ Unit
- GDB Hardware Debugging
- GDB OpenOCD Debugging
- GDB PyOCD Debugging
- GDB QEMU aarch64 Debugging
- GDB QEMU arm Debugging
- GDB QEMU gnuarmeclipse Debugging (Deprecated)
- GDB QEMU riscv32 Debugging
- GDB QEMU riscv64 Debugging
- GDB SEGGER J-Link Debugging
  - Debug_hello_world.elf
- Launch Group

Filter matched 17 of 19 items

Name: Debug_hello_world.elf

Main | Debugger | Startup | Source | Common | SVD Path

J-Link GDB Server Setup

- [x] Start the J-Link GDB server locally
- [ ] Connect to running target

Executable path: /opt/JLink/JLink_Linux_V792c_x86_64/JLinkGDBServer — Browse... Variables...

Actual executable: /opt/JLink/JLink_Linux_V792c_x86_64/JLinkGDBServer

(to change it use the global or workspace preferences pages or the project properties page)

Device name: MIMX8ML8_M7 — Supported device names

Endianness: (•) Little ( ) Big

Connection: (•) USB ( ) IP — (USB serial or IP name/address)

Interface: ( ) SWD (•) JTAG

Initial speed: ( ) Auto ( ) Adaptive (•) Fixed 4000 kHz

GDB port: 2331

SWO port: 2332 — [ ] Verify downloads [ ] Initialize registers on start

Telnet port: 2333 — [ ] Local host only [ ] Silent

Log file: — Browse...

Other options:

- [x] Allocate console for the GDB server
- [x] Allocate console for semihosting and SWO

GDB Client Setup

Executable name: /opt/GCC/arm-gnu-toolchain-12.2-arm-none-eabi/bin/arm-none-eabi-gdb — Browse... Variables...

Actual executable: /opt/GCC/arm-gnu-toolchain-12.2-arm-none-eabi/bin/arm-none-eabi-gdb

Other options:

Commands:

Revert Apply

Close Debug

Debug Configurations

Create, manage, and run configurations

type filter text

- C/C++ Application
- C/C++ Attach to Application
- C/C++ Container Launcher
- C/C++ Postmortem Debugger
- C/C++ Remote Application
- C/C++ Unit
- GDB Hardware Debugging
- GDB OpenOCD Debugging
- GDB PyOCD Debugging
- GDB QEMU aarch64 Debugging
- GDB QEMU arm Debugging
- GDB QEMU gnuarmeclipse Debugging (Deprecated)
- GDB QEMU riscv32 Debugging
- GDB QEMU riscv64 Debugging
- GDB SEGGER J-Link Debugging
  - Debug_hello_world.elf
- Launch Group

Filter matched 17 of 19 items

Name: Debug_hello_world.elf

Main | Debugger | Startup | Source | Common | SVD Path

Initialization Commands

- [x] Initial Reset and Halt Type: — Low speed: 1000 kHz

JTAG/SWD Speed: ( ) Auto ( ) Adaptive (•) Fixed 4000 kHz

- [x] Enable flash breakpoints
- [x] Enable semihosting Console routed to: [x] Telnet [ ] GDB client
- [x] Enable SWO CPU freq: 0 Hz. SWO freq: 0 Hz. Port mask: 0x1

Load Symbols and Executable

- [x] Load symbols
  - (•) Use project binary: hello_world.elf
  - ( ) Use file: — Workspace... File System...

Symbols offset (hex):

- [x] Load executable
  - (•) Use project binary: hello_world.elf
  - ( ) Use file: — Workspace... File System...

Executable offset (hex):

Runtime Options

- [ ] RAM application (reload after each reset/restart)

Run/Restart Commands

- [x] Pre-run/Restart reset — Type: — (always executed at Restart)

Revert Apply

Close Debug