

## Using Guiliani RZ/A SDK with TES eGML with e2Studio

Product:	Guiliani-SDK for Renesas RZ/A1
Release version:	2.6
Release date:	July 31, 2024

### **Table of Contents**

1	Introduction .....	3
2	Assumed Knowledge.....	3
3	e <sup>2</sup> studio Workspace .....	3
3.1	Board Support Package (BSP).....	4
3.1.1	Directory Structure .....	4
3.1.2	Build Configurations .....	4
3.2	BSP_Test .....	5
3.2.1	Directory Structure .....	5
3.2.2	Build Configurations .....	5
3.3	SR_GuilianiDemo / StreamRuntime .....	6
3.3.1	Directory Structure .....	6
3.3.2	Build Configurations .....	7
4	Debug Configurations .....	8
5	Annex .....	10
5.1.1	Startup Sequence of Guiliani Demo Application .....	10

## **List of Figures**

Fig. 1 e2Studio Workspace of SDK Project.....	4
Fig. 2 Debug Configurations.....	8
Fig. 3 Timeout settings.....	9
Fig. 4 Startup Sequence of Guiliani Demo Application.....	10

## **List of Tables**

Table 1 Directory Structure of BSP Project.....	4
Table 2 Directory Structure of BSP_Test Project .....	5
Table 3 Directory Structure of SR_GuilianiDemo.....	6
Table 4 Files in “Common” Directory .....	6
Table 5 Files in “Include” and “Source” Directories .....	6
Table 6 Files in “Share” Directory .....	6

## 1 Introduction

The SDK for Renesas RZ/A boards contains an e2Studio project, which can be used for editing and debugging the Demo. e2Studio is an eclipse based Integrated Development Environment (IDE). This document describes the different projects, their directory structure and the build configurations included in the e2Studio project workspace of the demo.

This guide does not explain how to create e2Studio project and configure the settings. It rather explains an e2Studio workspace, which is already created and included in SDK so that user can quickly test Guiliani demo and do the changes as per the requirements.

## 2 Assumed Knowledge

- Basic to advanced knowledge of C and C++
- General understanding and hands-on experience of e2Studio or eclipse (If you are not familiar with any of these tools, we recommend you to read “User’s Manual: Getting Started Guide” of e2Studio, available on Renesas website)

## 3 e<sup>2</sup> studio Workspace

e2Studio projects are available in the SDK within the Renesas folder. Launch e2Studio IDE and import the projects into your workspace.

The folder includes four projects (Fig. 1):

- BSP: Renesas Board Support Package (BSP) files for RSK-RZA1H (DisplayIt) board
- BSP\_Test: A test project to quickly test BSP without Guiliani
- SR\_GuilianiDemo: The Guiliani demo
- StreamRuntime: Base StreamRuntime-project

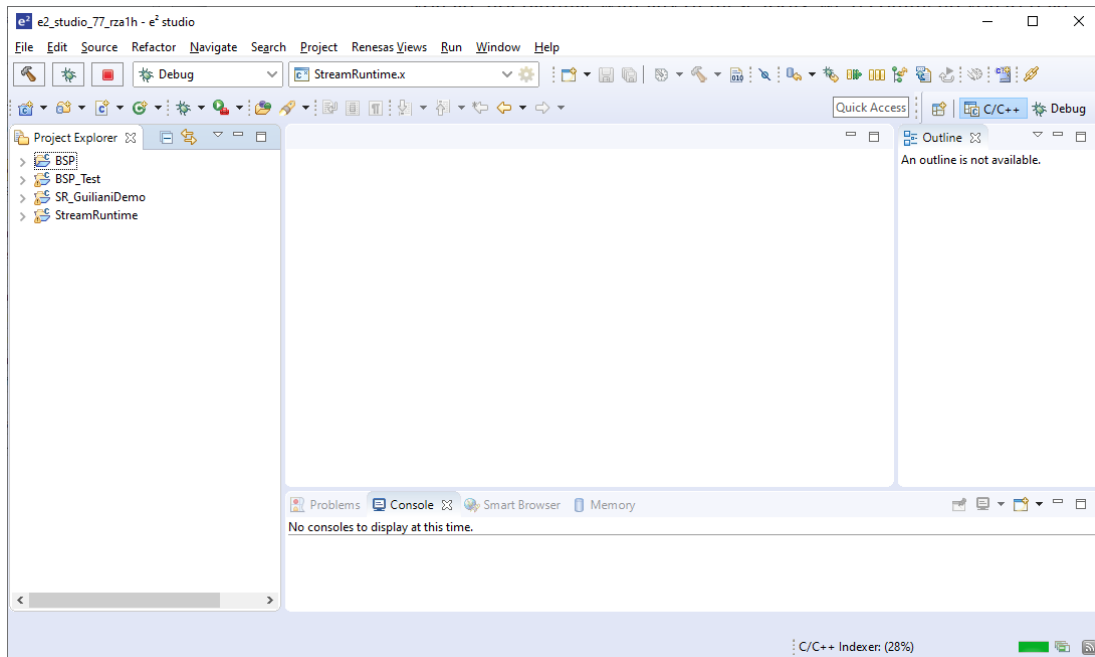


Fig. 1 e2Studio Workspace of SDK Project

### 3.1 Board Support Package (BSP)

This includes BSP for DisplayIt. The BSP contains initialization code for clocks, RAM, caches, peripherals which are specific to the boards. It also includes driver files and FreeRTOS port for the boards.

#### 3.1.1 Directory Structure

Directory	Description
src/renesas	Includes source code for drivers, middleware, startup, low level initialization and cache operations
src/freertos	Port for FreeRTOS operating system

Table 1 Directory Structure of BSP Project

#### 3.1.2 Build Configurations

- Debug/Release: It builds board support package for DisplayIt board. When the project is built, it creates a library libBSP.a for DisplayIt board, which can be used by SR\_GuilianiDemo and BSP\_Test projects.

## 3.2 BSP\_Test

This project allows a user to quickly test BSP of DisplayIt boards without Guiliani application. The test program can be flashed on the board and can be debugged.

### 3.2.1 Directory Structure

Directory	Description
linker_script	Linker scripts for DisplayIt boards.
src	Application source code

Table 2 Directory Structure of BSP\_Test Project

### 3.2.2 Build Configurations

- Debug: debug the BSP\_Test program from QSPI flash of DisplayIt board.

## 3.3 SR\_GuilianiDemo / StreamRuntime

These projects contain sample projects to run Guiliani on the board. SR\_GuilianiDemo is the Guiliani technical showcase and StreamRuntime is a minimal application as a good starting point for a new application.

### 3.3.1 Directory Structure

Directory	Description
Common	Common files over different Guiliani applications
Include	Project specific includes
Share	shared files between GSE and Guiliani application
Source	Project specific sources

Table 3 Directory Structure of SR\_GuilianiDemo

File	Description
[Include Source]/Platform/FreeRTOS/StreamRuntimeStartup.[cpp h]	Target specific initialization of Wrappers and configurations
[Include Source]/Platform/rza1/*.cpp[h]	RZA1 specific files for display and touch device
Source/Platform/FreeRTOS/StreamRuntime.cpp	Program entry points (main-function)
[Include Source]/Platform/win/StreamRuntimeStartup.[cpp h]”	Windows specific initialization of Wrappers and configurations
[Include Source]/StreamRuntimeBase.[h cpp]	Common functionality for all StreamRuntime projects
[Include Source]/StreamRuntimeConfig.[h cpp]	Loads project configuration
[Include Source]/StreamRuntimeGUI.[h cpp]	Loads GUI

Table 4 Files in “Common” Directory

File	Description
CustomExtension	Custom extensions.
GUIConfig/User*Resource.h	Resource IDs generated by GSE
MyGUI_SR.[cpp h]	GUI entry point

Table 5 Files in “Include” and “Source” Directories

File	Description
GUIConfig.cpp	This contains constants which hold the count of number of global properties, image resources, font resources, text resources, etc.

Table 6 Files in “Share” Directory

## 3.3.2 Build Configurations

There are two configurations available for SR\_GuilianiDemo project:

1. **DISPLAYIT\_QSPI\_Debug**: Debug configuration for DisplayIt. The demo application runs from QSPI flash. The project is built with debug information. Choose this configuration to debug the application.
2. **DISPLAYIT\_QSPI\_Release**: Release configuration for DisplayIt. The application runs from QSPI flash. The project is built with very little to no debug information and is optimized for speed or performance. Choose this configuration to test the performance.

## 4 Debug Configurations

Under *Run* → *Debug Configurations* → *Renesas GDB Hardware Debugging* menu of e2Studio, debug configurations are created for each build configuration present in e2Studio workspace (Fig. 2). The name of the each debug configuration is combination of a project name and its build configuration. For example, *SR\_GuilianiDemo RSK\_QSPI\_Debug* configuration is for project *SR\_GuilianiDemo* with *RSK\_QSPI\_Debug* configuration.

After a project is built, its debug configuration can be launched by clicking on Debug button. This will flash the binary file on the board and start debugging.

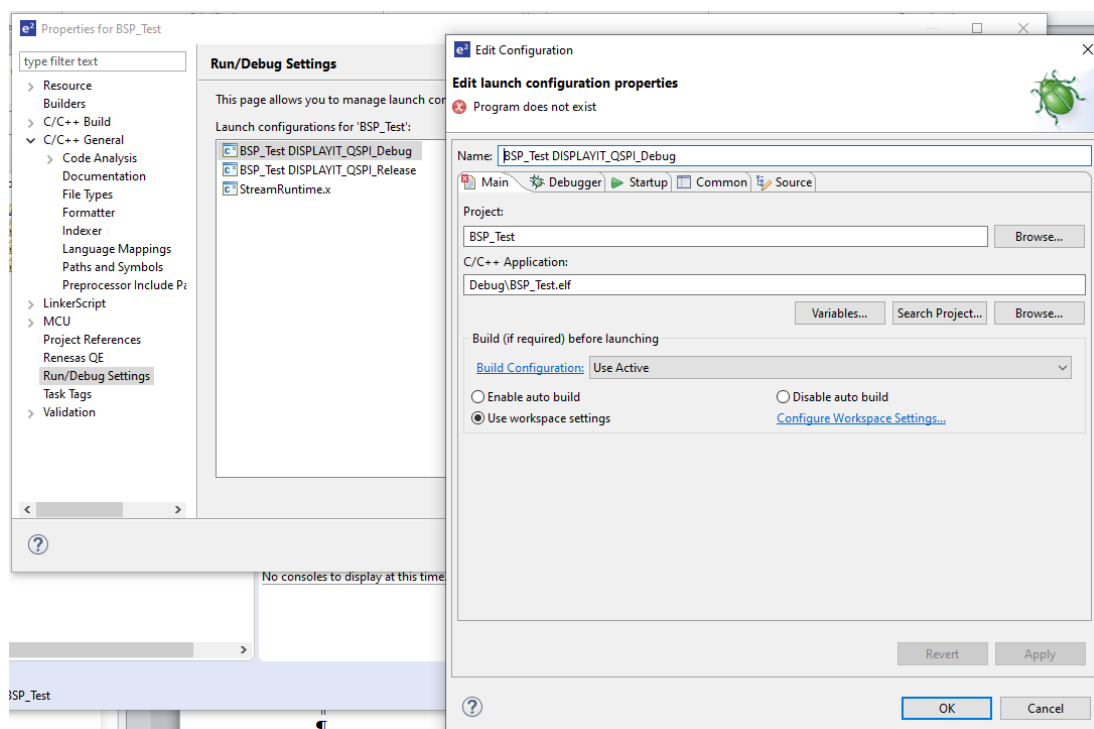
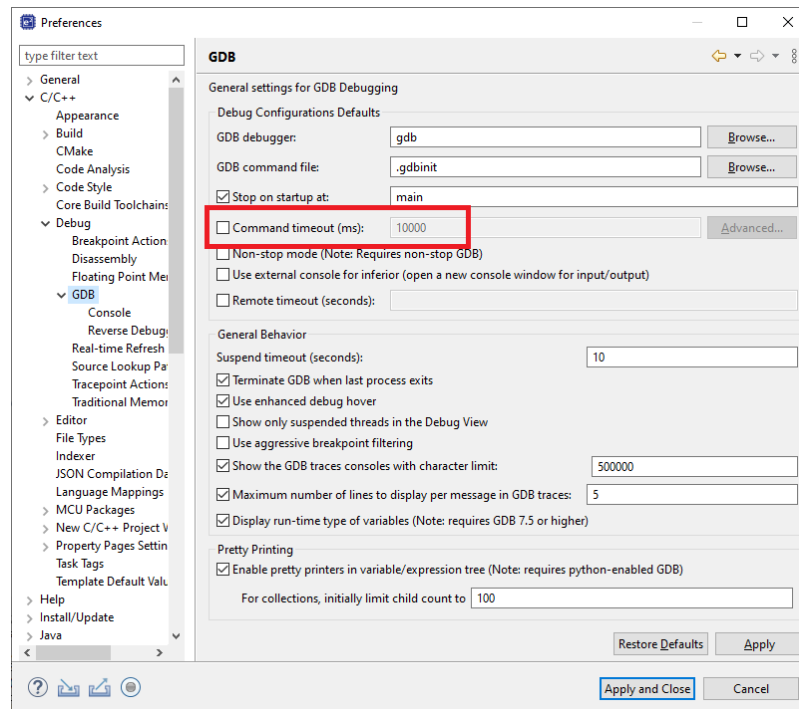


Fig. 2 Debug Configurations



**Notice:** When you are experiencing timeouts during start of a debug-session you might want to change some settings in e2studio.

In the menu “Window” select “Preferences” and then go to item “C/C++ / Debug / GDB” and either increase the time at “Command Timeout [ms]” or deactivate the whole item.



**Fig. 3 Timeout settings**

## 5 Annex

### 5.1.1 Startup Sequence of Guiliani Demo Application

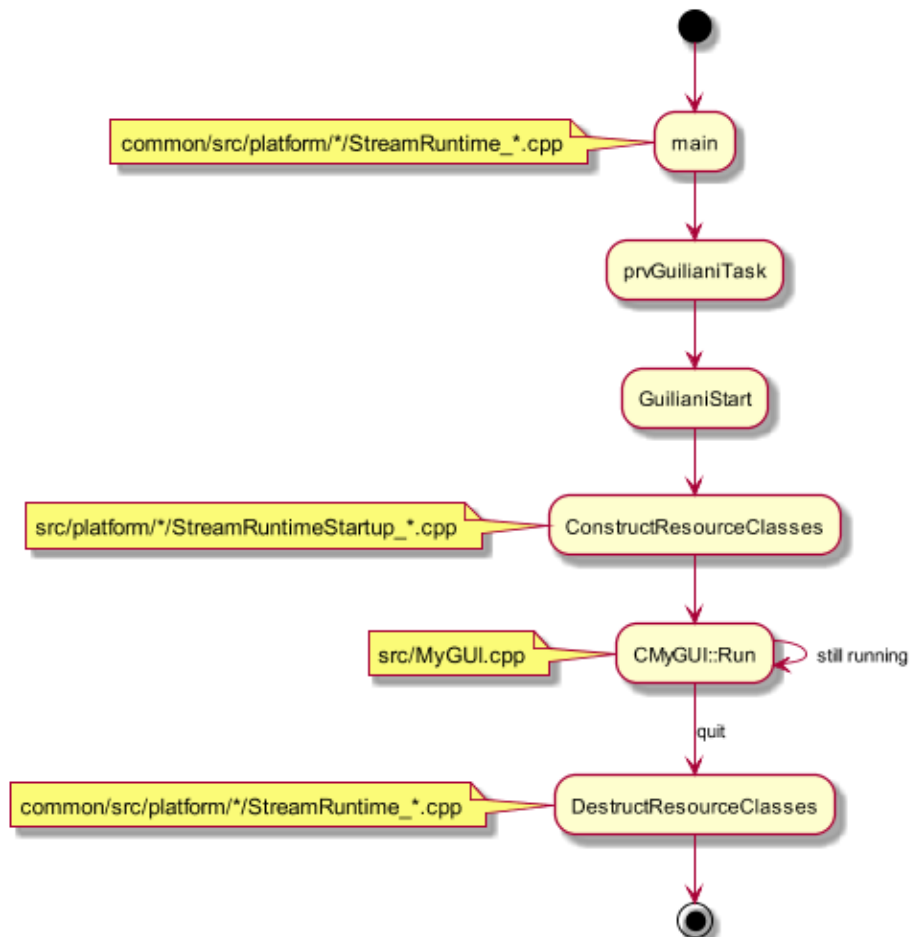


Fig. 4 Startup Sequence of Guiliani Demo Application