

# Using Guiliani RZ/A SDK with GR-Mango Evaluation Board Kit within e2Studio

Product:	SDK Guiliani for Renesas RZ/A
Release version:	2.6
Release date:	August 20, 2024

## **Table of Contents**

1	Introduction .....	3
2	Assumed Knowledge.....	3
3	e2Studio 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 .....	5
3.3.1	Directory structure.....	5
3.3.2	Build configurations.....	6
3.4	StreamRuntime .....	6
3.4.1	Directory structure.....	6
3.4.2	Build configurations.....	7
4	Debug Configurations .....	8

5	Annex .....	10
5.1.1	Startup sequence of Guiliani Demo application.....	10
5.1.2	Guiliani input wrapper .....	10

## **List of Figures**

Fig. 1	e2Studio Workspace of SDK Project.....	4
Fig. 2	Debug Configurations.....	8
Fig. 3	Flasher Selection.....	9
Fig. 4	Error Message for Timeout.....	9
Fig. 5	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 <SDK>\SR_GuilianiDemo .....	5
Table 4	Files in <SDK>\SR_GuilianiDemo\Common Directory .....	6
Table 5	Files in <SDK>\SR_GuilianiDemo\Include and <SDK>\SR_GuilianiDemo\Source Directory.....	6
Table 6	Linker Script in <SDK>\Renesas\BSP\generate\.....	6
Table 7	Files in <SDK>\GSE\Share Directory .....	6
Table 8	Directory Structure of <SDK>\StreamRuntime.....	6
Table 9	Files in <SDK>\StreamRuntime\Common Directory .....	7
Table 10	Files in <SDK>\StreamRuntime\Include and <SDK>\StreamRuntime\Source Directory.....	7
Table 11	Linker Script in <SDK>\Renesas\BSP\generate\.....	7
Table 12	Files in <SDK>\GSE\Share Directory .....	7

## 1 Introduction

The SDK for Renesas RZ/A boards contains an e2Studio project, which can be used for editing and debugging the Guiliani 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 an e2Studio project and configure the settings. It rather explains an e2Studio workspace, which is already created and included in the SDK so that the user can quickly test the Guiliani demo and do the changes according to his 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 e2Studio Workspace

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

The following four projects are available (Fig. 1):

- BSP: Renesas Board Support Package (BSP) files for GR-Mango Evaluation Board Kit
- BSP\_Test: A test project to quickly test BSP without Guiliani
- SR\_GuilianiDemo: The Guiliani demo
- StreamRuntime: The StreamRuntime demo

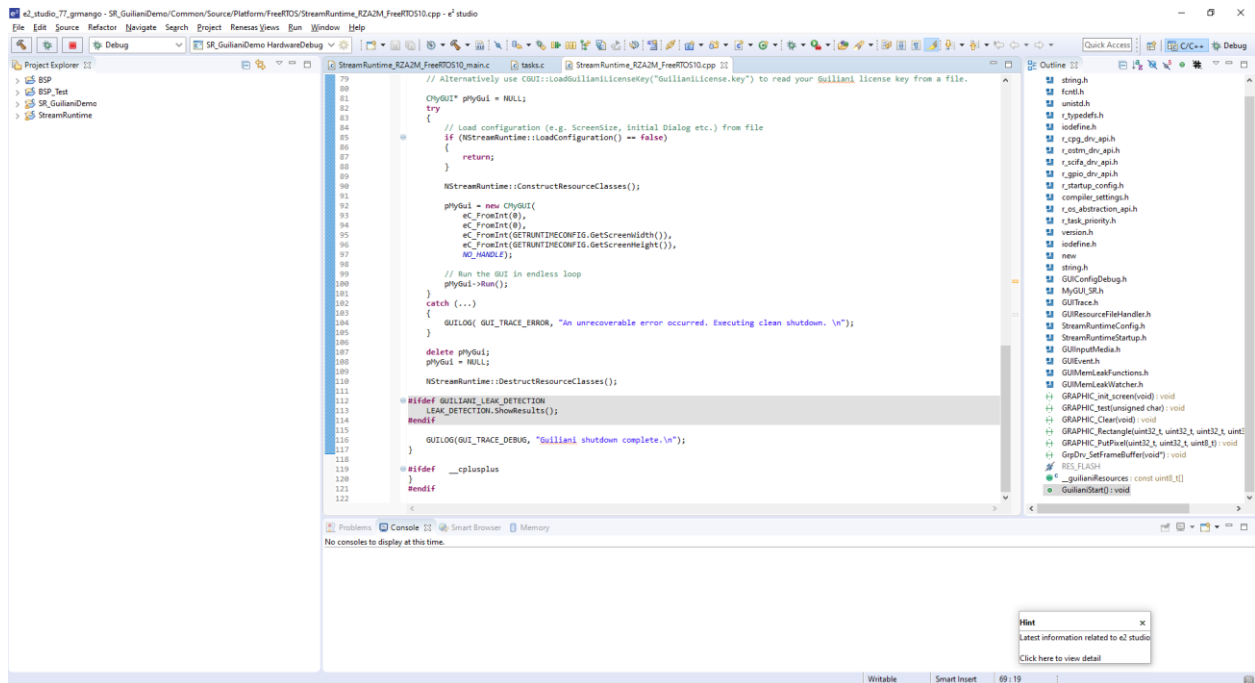


Fig. 1 e2Studio Workspace of SDK Project

## 3.1 Board Support Package (BSP)

This SDK includes the BSP for the GR-Mango Evaluation Board Kit. The BSP contains initialization code for clocks, RAM, caches and peripherals which are specific to the boards. It also includes driver files and a FreeRTOS port for the evaluation board kit.

### 3.1.1 Directory structure

Directory	Description
generate	Includes source code for drivers, startup, low level initialization and cache operations
src/config_files	FreeRTOS-configuration-header
src/freertos	Port for FreeRTOS operating system
src/renesas	Includes middleware and display and camera specific port settings

Table 1 Directory Structure of BSP Project

### 3.1.2 Build configurations

- **Debug:** It builds board support package for GR-Mango Evaluation Board Kit in debug mode. When the project is built, it creates a library libBSP.a in a subfolder Debug, which can be used by SR\_GuilianiDemo, StreamRuntime and BSP\_Test projects.

- Release: It builds board support package for GR-Mango Evaluation Board Kit in release mode. When the project is built, it creates a library libBSP.a in a subfolder Release, which can be used by SR\_GuilianiDemo, StreamRuntime and BSP\_Test projects.

## 3.2 BSP\_Test

This project allows a user to quickly test BSP of GR-Mango Evaluation Board Kit without the need of the Guiliani application. The test program can be flashed on the board and can be debugged. It is a simple blinking application.

### 3.2.1 Directory structure

Directory	Description
src	Application source code

Table 2 Directory Structure of BSP\_Test Project

### 3.2.2 Build configurations

- HardwareDebug: BSP\_Test program from flash of GR-Mango Evaluation Board Kit in Debug mode.
- Release: BSP\_Test program from flash of GR-Mango Evaluation Board Kit in Release mode.

## 3.3 SR\_GuilianiDemo

This project contains the files required for porting Guiliani on the Renesas board.

### 3.3.1 Directory structure

Directory	Description
Common	Common files over different Guiliani applications
GuilianiDemo	Contains GSE projects with different resolutions
Include	Project specific includes
Source	Project specific sources

Table 3 Directory Structure of <SDK>\SR\_GuilianiDemo

File	Description
Platform/*/StreamRuntime*.[c cpp h]	Program entry points (main function) for different platforms
[Include Source]/Platform/FreeRTOS/StreamRuntimeStartup.[h cpp]	Target specific initialization of wrappers and configurations
[Include Source]/Platform/win/pc/StreamRuntimeStartup.[h cpp]	Windows specific initialization of wrappers and configurations

[Include Source]/StreamRuntimeConfig.[h cpp]	Loads project configuration
[Include Source]/StreamRuntimeGUI.[h cpp]	Loads GUI

Table 4 Files in <SDK>\SR\_GuilianiDemo\Common Directory

File	Description
CustomExtension	Custom extensions.
GUIConfig/User*Resource.h	Resource IDs generated by GSE
GUIConfigCustom/*	Custom IDs for use in Guiliani application.
Demo_*.h cpp	Specific code for the different demo parts
MyGUI_SR.h cpp	GUI entry point

Table 5 Files in <SDK>\SR\_GuilianiDemo\Include and <SDK>\SR\_GuilianiDemo\Source Directory

File	Description
linker_settings.ld	Linker script for GR-Mango Evaluation Board Kit

Table 6 Linker Script in <SDK>\Renesas\BSP\generate\

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

Table 7 Files in <SDK>\GSE\Share Directory

### 3.3.2 Build configurations

There are two configurations available for SR\_GuilianiDemo project.

1. HardwareDebug: Debug configuration for GR-Mango Evaluation Board Kit. The demo application runs from QSPI flash. Choose this configuration to debug the application.
2. Release: Release configuration for GR-Mango Evaluation Board Kit. The application runs from QSPI flash. Choose this configuration to test the performance.

## 3.4 StreamRuntime

This project contains the files required for porting Guiliani on the Renesas board.

### 3.4.1 Directory structure

Directory	Description
Common	Common files over different Guiliani applications
GuilianiDemo	Contains GSE projects with different resolutions
Include	Project specific includes
Source	Project specific sources

Table 8 Directory Structure of <SDK>\StreamRuntime

File	Description
[Include Source]&Platform/*/StreamRuntime*.h cpp]	Program entry points (main function) for different platforms
[Include Source]/Platform/FreeRTOS/StreamRuntimeStartup.h cpp]	Target specific initialization of wrappers and configurations
[Include Source]/Platform/win/pc/StreamRuntimeStartup.h cpp]	Windows specific initialization of wrappers and configurations
[Include Source]/StreamRuntimeConfig.h cpp]	Loads project configuration
[Include Source]/StreamRuntimeGUI.h cpp]	Loads GUI

Table 9 Files in <SDK>\StreamRuntime\Common Directory

File	Description
CustomExtension	Custom extensions.
GUIConfig/User*Resource.h	Resource IDs generated by GSE
GUIConfigCustom/*	Custom IDs for use in Guiliani application.
MyGUI_SR.h cpp]	GUI-application entry point

Table 10 Files in <SDK>\StreamRuntime\Include and <SDK>\StreamRuntime\Source Directory

File	Description
linker_settings.ld	Linker script for GR-Mango Evaluation Board Kit

Table 11 Linker Script in <SDK>\Renesas\BSP\generate\

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

Table 12 Files in <SDK>\GSE\Share Directory

### 3.4.2 Build configurations

There are two configurations available for StreamRuntime project.

1. HardwareDebug: Debug configuration for GR-Mango Evaluation Board Kit. The demo application runs from QSPI flash. Choose this configuration to debug the application.
2. Release: Release configuration for GR-Mango Evaluation Board Kit. The application runs from QSPI flash. 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 each debug configuration is a combination of the project name and its build configuration. For example *SR\_GuilianiDemo HardwareDebug* configuration is for project *SR\_GuilianiDemo* with *HardwareDebug* configuration.

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

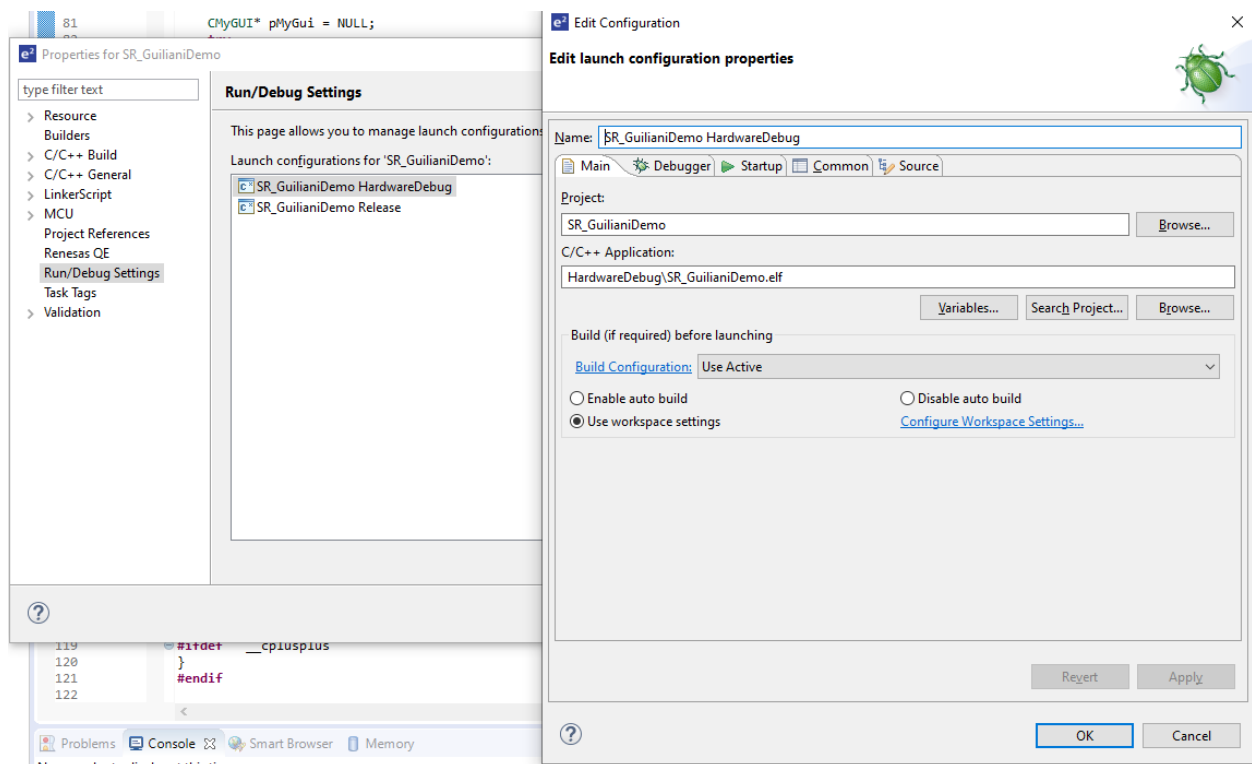
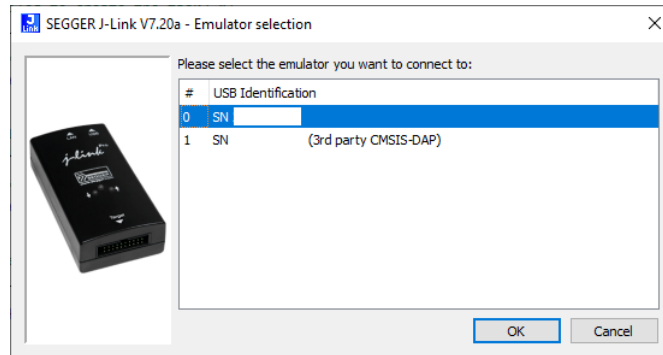


Fig. 2 Debug Configurations



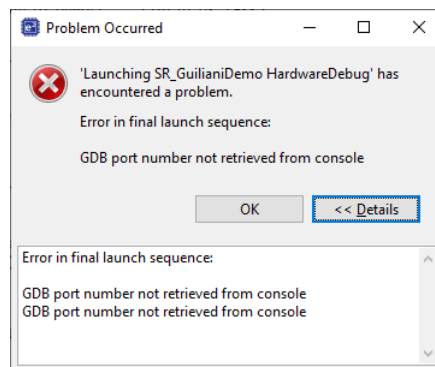
**Note:** when the GR-Mango is connected via Micro-USB with the PC, starting the debug-session will bring up another window to select the JLink, which should be used for flashing.



**Fig. 3 Flasher Selection**

Please select the entry in the list which does **not** have the text (3<sup>rd</sup> party CMSIS-DAP) and click on “OK”. After that the flashing will start.

If you spent too long selecting the JLink a timeout will occur and the debug-session will be aborted. The window below will appear for notification.



**Fig. 4 Error Message for Timeout**

## 5 Annex

### 5.1.1 Startup sequence of Guiliani Demo application

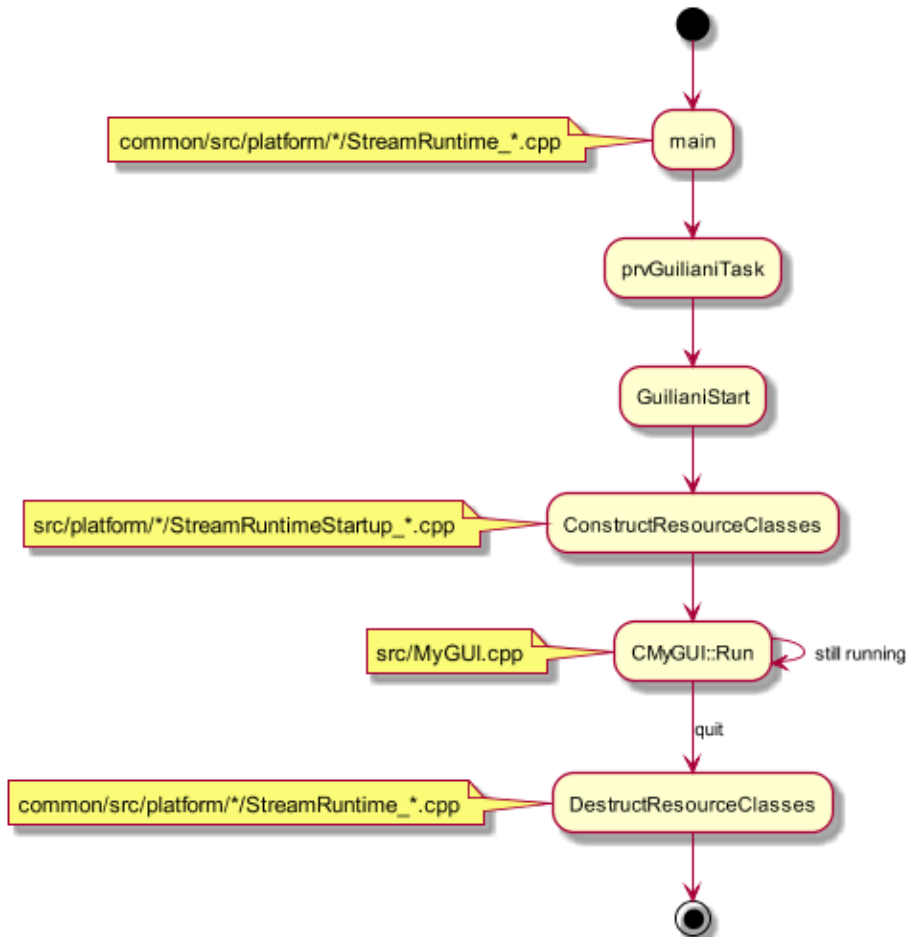


Fig. 5 Startup Sequence of Guiliani Demo Application

### 5.1.2 Guiliani input wrapper

Guiliani input wrapper wraps the functionality to get the touch events on the display. Input wrappers for the GR-Mango Evaluation Board Kit are already included in Guiliani library (`libGuiliani.a`) available in this SDK. An additional file `GUIInputExample.cpp` (available under `SDK/InputWrapper` directory) is provided as an example of an input wrapper so as to illustrate the implementation of the input wrapper. Then the user can modify this file to implement his own input wrapper according to his requirements. For debugging purpose, this file can be included in e2Studio project by using the import option. An instance of the input wrapper has to be created in `StreamRuntimeStartup.cpp`.

The main method of the Guiliani input wrapper is `GetEvent`. `GetEvent` shall last at maximum `uiIdleTime` (in milliseconds). The method should return as soon as possible after `StopIdle` is

called. In this case the `GetEvent` method waits until either `uiIdleTime` is exceeded or an (asynchronous) event from the touch screen occurs or `StopIdle` is called. When an event occurs the touch controller is read out and a `MouseEvent` is generated.

In general `GetEvent` can return either a `MouseEvent` or a `KeyboardEvent` or `NULL`.