guiliani.de

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

Product:	SDK Guiliani for Renesas RZ/A
Release version:	2.4
Release date:	June 18, 2021

Table of Contents

1 Introduction 3
2 Assumed Knowledge 3
3 Prerequisites 3
4 e²Studio Workspace 3
4.1 Board Support Package (BSP) 4
4.1.1 Directory structure 4
4.1.2 Build configurations 4
4.2 BSP_Test 5
4.2.1 Directory structure 5
4.2.2 Build configurations 5
4.3 SR_GuilianiDemo 5
4.3.1 Directory structure 5
4.3.2 Build configurations 6
4.4 StreamRuntime 6
4.4.1 Directory structure 6
4.4.2 Build configurations 7

5 Debug Configurations 8
6 Annex 9
6.1.1 Startup sequence of Guiliani Demo application 9
6.1.2 Guiliani input wrapper 9

List of Figures

Fig. 1 e2Studio Workspace of SDK Project 4
Fig. 2 Debug Configurations 8
Fig. 5 Startup Sequence of Guiliani Demo Application 9

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 7
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 Prerequisites

- Installed Guiliani SDK for Renesas GR-Mango
- Installed e2Studio v2021-04
- Installed GNU ARM Toolchain 6-2017-q2-update
- Installed Libgen Update for GNU ARM Embedded Toolchains (at least Version 1.2018.02)
- GNU MCU Eclipse Windows Build Tools (2.9-20170629-1013)

4 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

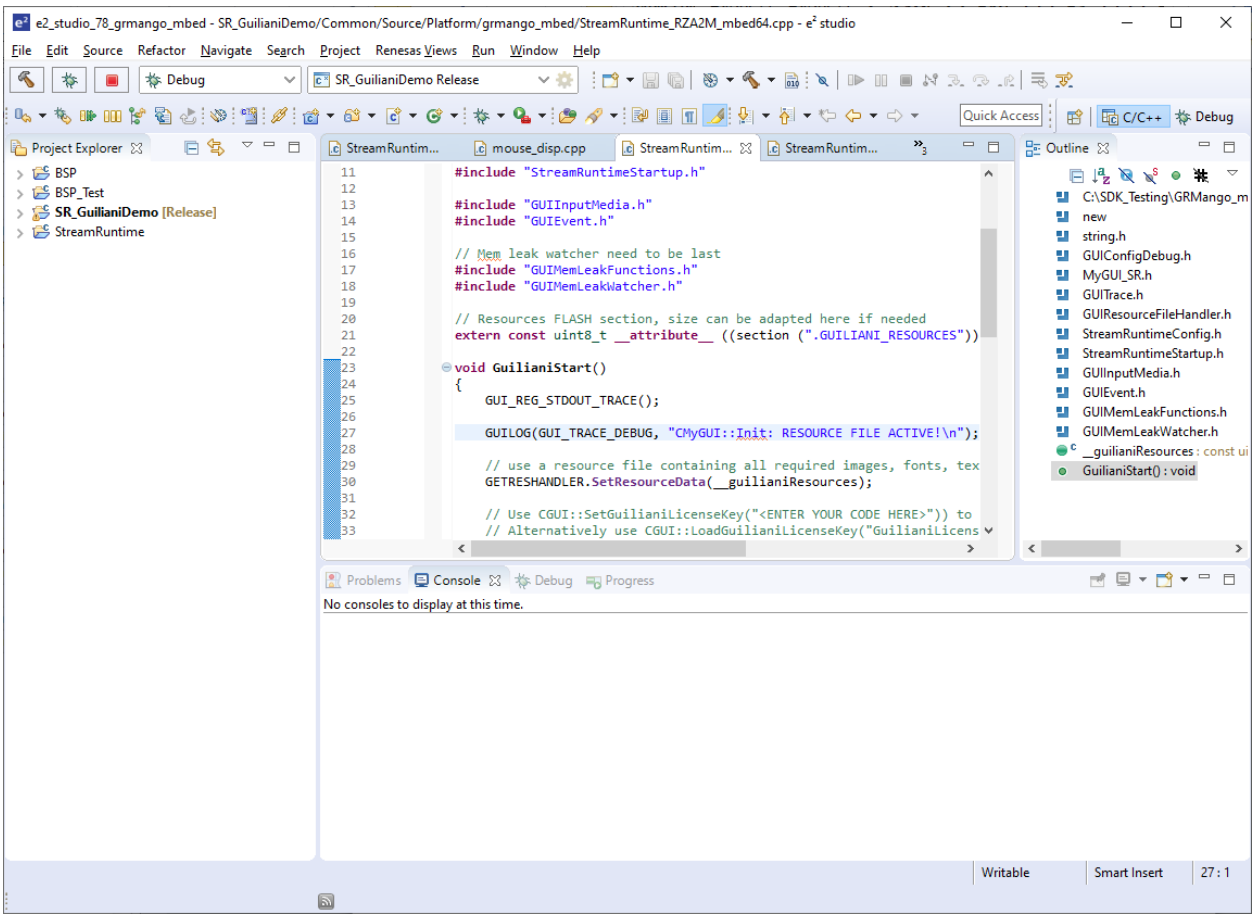


Fig. 1 e2Studio Workspace of SDK Project

4.1 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 mbed OS port for the evaluation board kit.

4.1.1 Directory structure

Directory	Description
mbed-gr-libs	Includes source code for drivers, startup, low level initialization and cache operations
mbed-os	mbed abstraction-level

Table 1 Directory Structure of BSP Project

4.1.2 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.

4.2 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 displaying a grid and the mouse-cursor on the screen. The current mouse-location is output via serial console.

4.2.1 Directory structure

Directory	Description
mouse_proc	Additional code for initializing mouse

Table 2 Directory Structure of BSP_Test Project

4.2.2 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.

4.3 SR_GuilianiDemo

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

4.3.1 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

Common/[Include|Source]/Platform/grmango_mbed/StreamRuntimeStartup.[cpp|h]	Target specific initialization of wrappers and configurations
Common/[Include|Source]/Platform/win/pc/StreamRuntimeStartup.cpp"	Windows specific initialization of wrappers and configurations
Common/[Include|Source]/StreamRuntimeConfig.[h|cpp]	Loads project configuration
Common/[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_*.[cpp|h]	Specific code for the different demo parts
MyGUI_SR.[cpp|h]	GUI entry point

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

File	Description
RZA2M.ld	Linker script for GR-Mango Evaluation Board Kit. This file will be generated prior to build.

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

4.3.2 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.

4.4 StreamRuntime

This project contains the minimum files required for creating a new Guiliani-application.

4.4.1 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
Common/[Include|Source]/Platform/*/StreamRuntime*.[cpp|h]	Program entry points (main function) for different platforms
Common/[Include|Source]/Platform/grmango_mbed/StreamRuntimeStartup.[cpp|h]	Target specific initialization of wrappers and configurations
Common/[Include|Source]/Platform/win/pc/StreamRuntimeStartup.cpp"	Windows specific initialization of wrappers and configurations
Common/[Include|Source]/StreamRuntimeConfig.[h|cpp]	Loads project configuration
Common/[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.
Demo_*.[cpp|h]	Specific code for the different demo parts
MyGUI_SR.[cpp|h]	GUI-application entry point

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

File	Description
RZA2M.ld	Linker script for GR-Mango Evaluation Board Kit. This file will be generated prior to build.

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

4.4.2 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.

5 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.

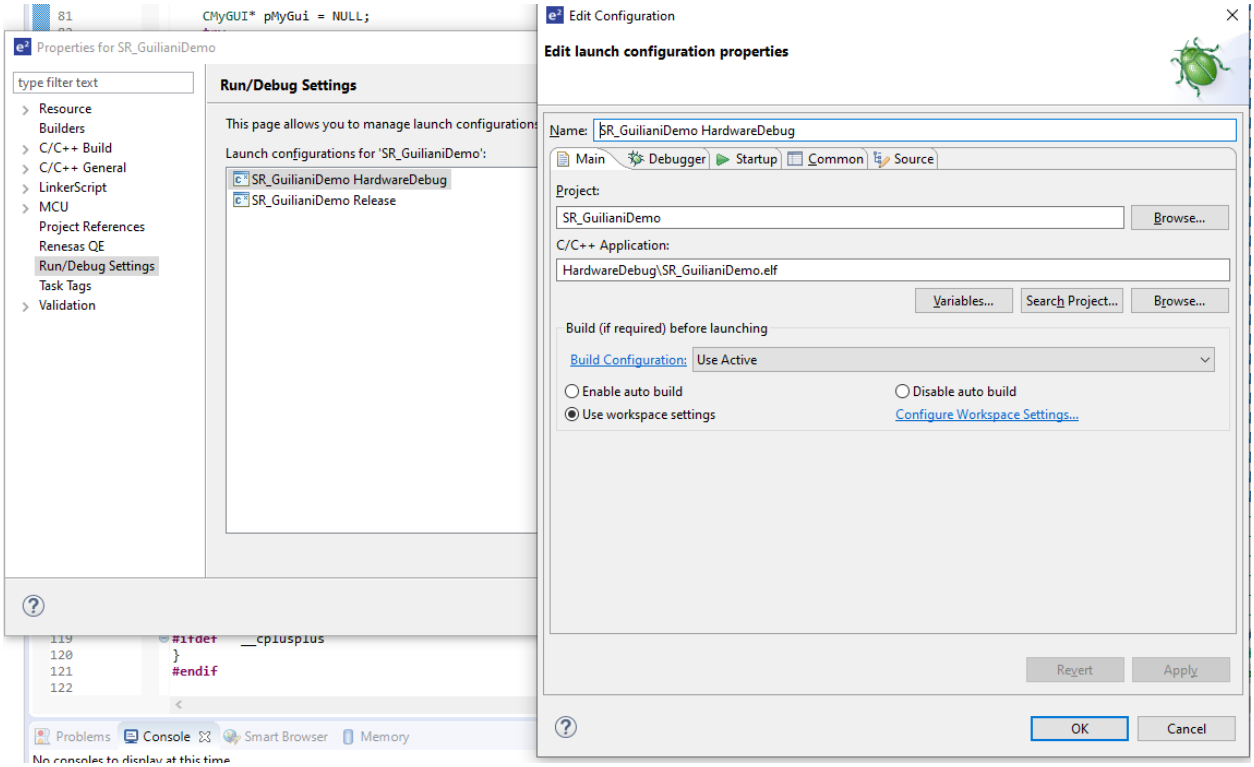


Fig. 2 Debug Configurations

6 Annex

6.1.1 Startup sequence of Guiliani Demo application

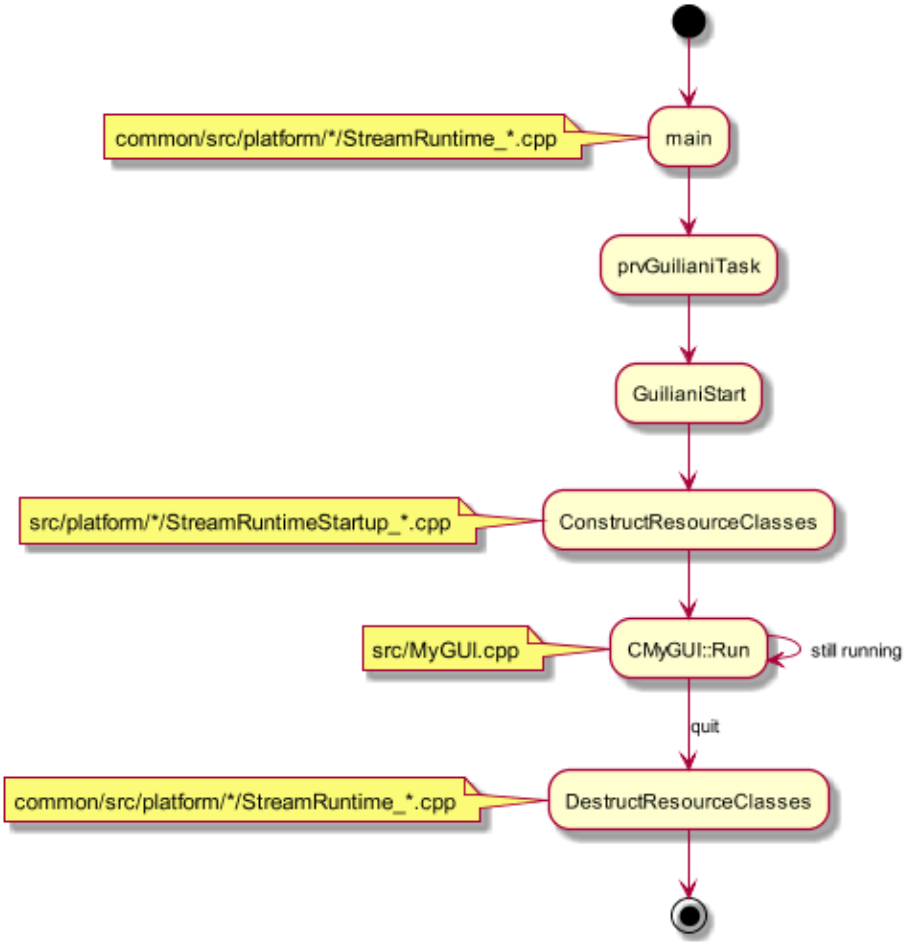


Fig. 3 Startup Sequence of Guiliani Demo Application

6.1.2 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 occurrs the touch controller is read out and a MouseEvent is generated.

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