MicroEJ® ONE

Build Your First Java Platform
On YLCDRX63N (WQVGA)

Getting Started

Reference:	TLT-0680-MAN-PlatformGettingStarted-YLCDRX63N-WQVGA
Revision:	A
Architecture:	Renenas RX
Compiler:	Renenas E2Studio
Board:	YLCDRX63N (WQVGA)

Confidentiality & Intellectual Property

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Intended Audience

The intended audience for this document are developers who wish to develop their first Java application with MicroEJ. Notes:

- This document is for YLCDRX63N (WQVGA) board.
- This document is not a user guide for the C development environment used for the final application link. Please consult the supplier of the C development environment for more information.
- Please visit the website http://www.is2t.com/get-started for more information about MicroEJ® ONE products (platforms, videos, examples, application notes, etc.).

1.2 Scope

This document describes, step by step, how to start your development with MicroEJ:

- Register your product.
- Understand the package.
- Create your own platform.
- Run a Java application on the simulation platform (SimJPF) and on the embedded platform (EmbJPF).

1.3 Prerequisites

- PC with Windows XP or later.
- The MicroEJ® ONE environment must be installed.
- YLCDRX63N (WQVGA) board.
- The Segger J-Link Lite RX utility (normally installed as part of the Renesas E2Studio installation).
- A GNU-C development environment. The examples are packaged ready to run using the Renesas E2Studio C IDE, which this document assumes has been successfully installed. Please visit the Renesas website to obtain a version of the Renesas E2Studio C IDE. Note, however, that developers are free to use a different CDT packaging..

1.4 Terminology

A Java platform (JPF) is a library that, at a minimum, embeds a JVM (the MicroJvm® virtual machine from IS2T), some native libraries (also called firmware, or drivers, or BSP), and some Java libraries that characterize an application domain. Depending on these libraries, some specific tools are provided too.

Every embedded JPF (EmbJPF) has a sibling platform that simulates it on a PC, called the SimJPF. The SimJPF exactly mimics the semantics of the JPF. The SimJPF can be extended with Mocks to simulate the native (C, Ada, asm, ...) code of the JPF, such as drivers, or to connect hardware to the workstation.

Mocks can feed the SimJPF with real external stimuli, so that the Java application can behave as if it was running on the EmbJPF. SimJPF allows the design and functional testing of an embedded Java application in a development environment.

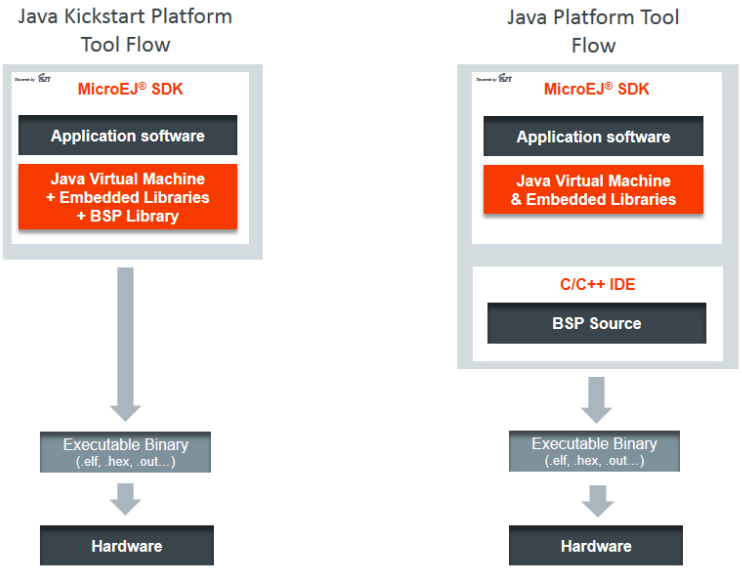

Figure 1.1. Platforms Flow

2 Product Registration

The development version of MicroEJ uses hardware activation keys.

- Open MicroEJ.
- Open welcome page: `Help > Welcome`.

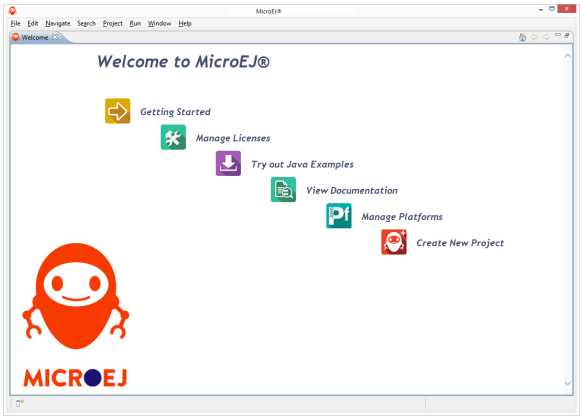

Figure 2.1. MicroEJ Welcome Page

- Insert your USB dongle.
- Press the Manage Licenses button. Hardware dongles are automatically detected when the MicroEJ preferences main page is shown.
- Click on Refresh when a new hardware dongle is plugged in.

Note

The MicroEJ preferences page is also accessible from the menu `Window > Preferences > MicroEJ`.

3 Overview

3.1 Platforms

Several Java platforms (JPF) are pre-installed in MicroEJ.

- Open MicroEJ.
- Open the welcome page: `Help > Welcome`.
- Press the `Manage Platforms` button.

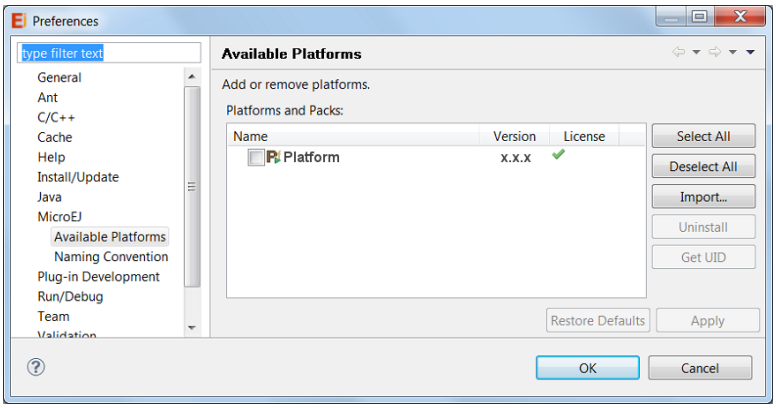

Figure 3.1. Available Platforms

This Available Platforms page shows all platforms installed in MicroEJ. Each platform has its own characteristics and specific behavior.

Note

- The Available Platforms page is automatically populated by the work-in-progress JPFs.
- The Available Platforms page is also accessible from the menu `Window > Preferences > MicroEJ > Available Platforms`.

3.2 Renesas RX GCC

This Java platform architecture has been designed to be extended to run on any board with a Renesas RX CPU. The Section 4, “Create and Use Your First Java Platform” uses this platform to run your first Java application on your first extended platform.

This platform requires the Renesas E2Studio C compiler to create the final binary file for a Java application.

4 Create and Use Your First Java Platform

The aim of this chapter is to create a platform from a Java platform architecture. The platform will then be used to run a Java application in subsequent chapters.

Although it is possible to use MicroEJ to create every aspect of a platform in accordance with specific requirements, in this chapter we will use a pre-packaged example of a platform that is already configured for the YLCDRX63N (WQVGA).

- Open MicroEJ.
- Open the Java Platform Example wizard: `File > New > Java Platform`.
- Select the Java platform architectue `Renesas RX GCC` from the combo box. A list of examples is available:

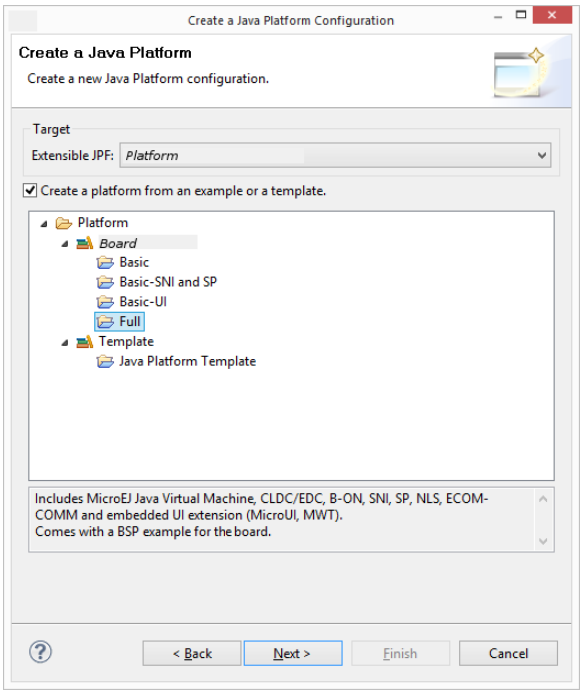

Figure 4.1. Java Platform Configuration (page 1)

- Select the example configuration `Full` in menu YLCDRX63N (WQVGA).
- Click on Next. Give a name which be used as prefix for all platform projects. For instance: `MyPlatform`.

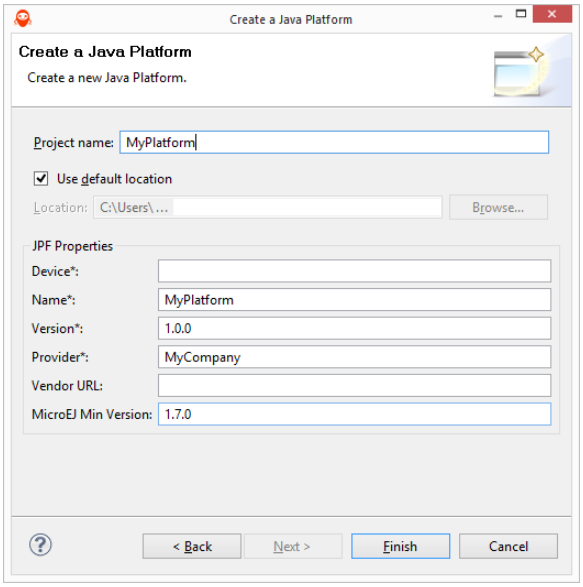

Figure 4.2. Java Platform Configuration (page 2)

- Click on Finish. The selected example is imported as several projects prefixed by the given name:
  - `[example]-bsp` contains a ready-to-use package for Renesas E2Studio on YLCDRX63N (WQVGA) board.
  - `[example]-configuration` contains some files to configure the platform automatically.
  - `[example]-microuiGeneratorExtension` contains some files to extend the MicroUI Image Generator tool.

A README file and a platform configuration file are automatically opened. The README file explains the aim of the projects and how to launch an example. We recommend you follow the steps described in this document, and refer to the README file for more detail and latest changes.

The platform needs to be built with several additional modules (Java and natives libraries, scripts etc.).

- From the platform configuration file, click on the link `Build Platform`

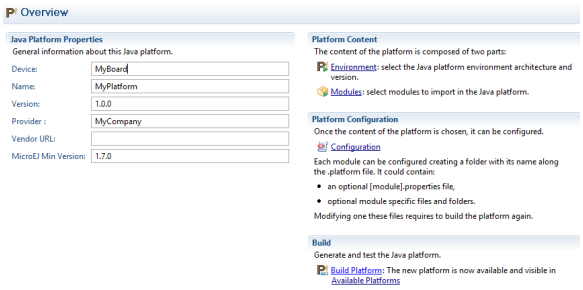

Figure 4.3. Platform Configuration File

The build starts. This step can take several minutes. You can see the progress of the build steps in the Eclipse console. Please wait for the final message `BUILD SUCCESSFUL`.

At the end of the execution the platform is fully built for the YLCDRX63N (WQVGA) board and is ready to be linked into the Renesas E2Studio project. The name of this platform is YLCDRX63N-WQVGA-MyPlatform-RXx_GCC.

The platform is now ready for use and available in the Platforms list of your MicroEJ repository (`Windows > Preferences > MicroEJ`)

5 Run an Example on the Simulated Java Platform (SimJPF)

The aim of this chapter is to create a simple Java application from a built-in example. This example will initially be run on the simulator (SimJPF) of the platform created in the previous chapter. Then, in the next chapter, this application will be compiled and deployed on the YLCDRX63N (WQVGA) board using the EmbJPF.

5.1 Create Example

- Open MicroEJ.
- Open the welcome page: `Help > Welcome`.
- Press the Try out Java Examples button to open the New Java Example Project wizard.
- Select the Java platform YLCDRX63N-WQVGA-MyPlatform-RXx_GCC from the combo box.
- Select the example `Examples > MicroUI > MVC`.

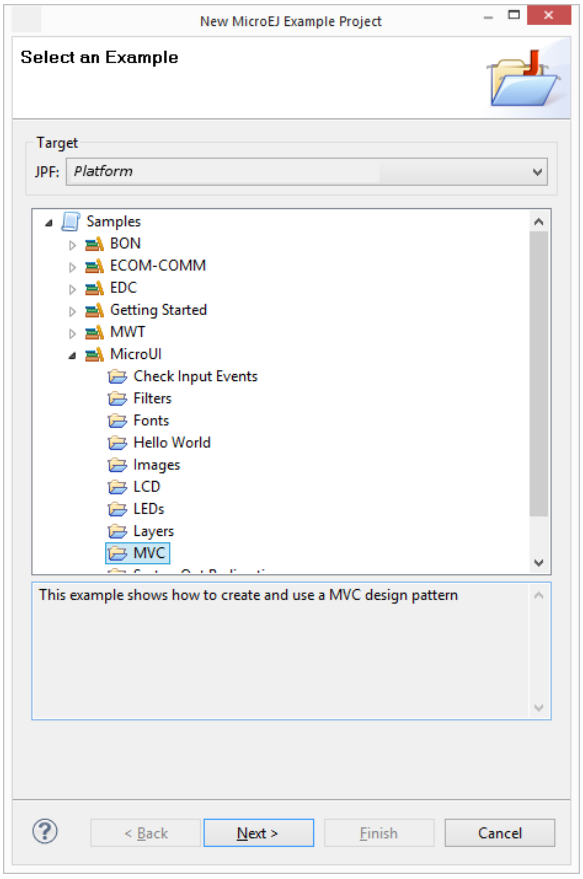

Figure 5.1. New MicroEJ Example Project (page 1)

- Click on Next. The next page suggests a name for the new project.

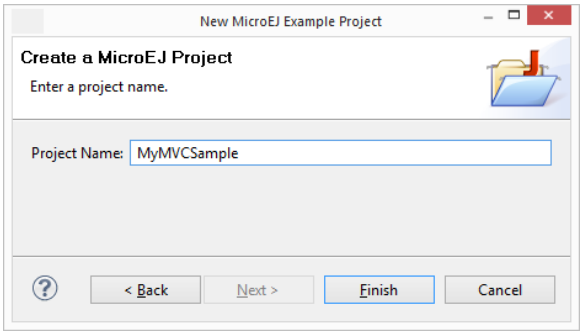

Figure 5.2. New MicroEJ Example Project (page 2)

- Click on Finish. The selected example is imported into a project with the given name. The main class (the Java class which contains the main() method, here: MVCDemo.java) is automatically opened.

Note

Note: The New Java Example Project wizard is also accessible from the menu `File > New > Java Example`.

5.2 Run Example

- Open the run dialog (`Run > Run configurations...`) and select the MicroEJ launcher MyMVCSample SimJPF.
- Click on `Run`: the application starts. It is executed on the simulator (SimJPF) of the selected JPF (YLCDRX63N-WQVGA-MyPlatform-RXx_GCC). A picture of the board appears. The result of the execution is shown on the display of the board. The touchscreen can be used to adjust the division of the screen areas.

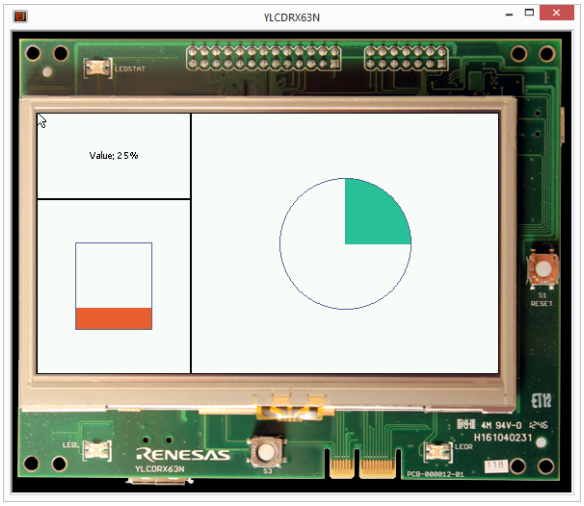

Figure 5.3. MVC Example on the Simulator

6 Run the Example on the YLCDRX63N (WQVGA) Board (Emb-JPF)

6.1 Compile Java Application

- Open the run dialog (`Run > Run configurations...`) and select the MicroEJ launcher `MyMVCSample EmbJPF`.
- Click Run: the application is compiled, and the compilation result (an ELF file) is copied into a well-known location in the workspace. The Renesas E2Studio BSP project will look for it there when it performs the final link.

6.2 Link and Deploy Java Application

The aim of the final step is to:

- Compile the BSP project (such as drivers).
- Link the BSP and the others libraries (MicroJvm Virtual Machine, native stacks, Java application).
- Deploy the full application on the YLCDRX63N (WQVGA) board.

Note

This final step uses a C IDE – this document assumes the use of the Renesas E2Studio.

- In MicroEJ, expand the project `[example]-bsp` and the folder MicroEJ/e2studio. A Ant file (`e2studio.xml`) is available.

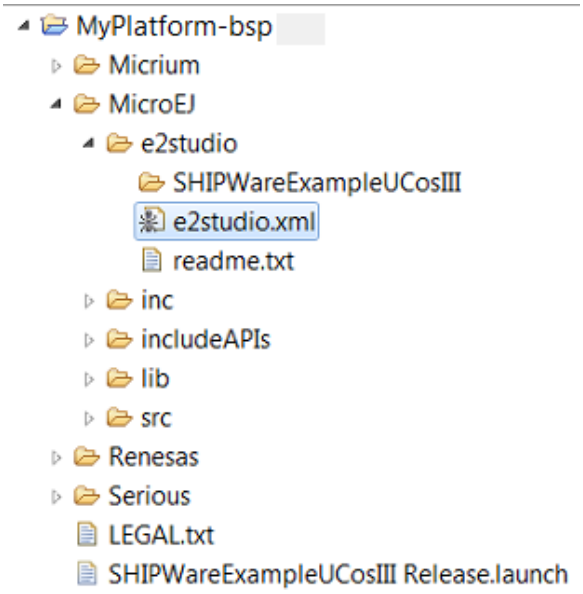

Figure 6.1. Renesas E2Studio Project File

- Right-click on this file and select `Run As > Ant Build...` to open the C IDE (Renesas E2Studio).

Note

If an error occurred on Renesas E2Studio path location, please update the Ant property `e2studio` in the `e2studio.xml` file with the right Renesas E2Studio path location.

The remaining steps are performed within Renesas E2Studio.

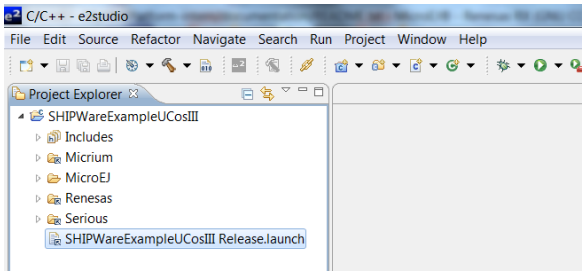

Figure 6.2. C IDE

- Build the Renesas E2Studio project by right-clicking on the project in the Project Explorer and selecting menu `Build project`. The project is compiled and linked.
- Ensure that:
  - The YLCDRX63N (WQVGA) board is connected via JTAG cable to the Segger J-Link RX.
  - The YLCDRX63N (WQVGA) board is connected via a USB cable to the PC.
  - The Segger J-Link RX is connected via USB cable to the PC.
  - Deploy and run the application. Right click on the project launcher (*.launch file in your project). `Run > Debug Configurations... > SHIPWareExampleUCosIII Release`

The application downloads and starts (this takes approximately 15-20 seconds). The result of the execution is shown on the display of the board.

Congratulations, you have built the Java application on your own platform!

7 Appendix

7.1 YLCDRX63N Technical Specifications

The following table illustrates some board features used by the Java platform. It doesn't list all board features (such as all available memories on the board).

MCU architecture	RX (RX63N)
MCU Clock speed	100MHz (165 DMIPS)
Internal Flash	2 MBytes
Internal RAM	128 KBytes
External RAM	16 MBytes (SDRAM)

Table 7.1. YLCDRX63N Technical Specifications

7.2 Board Setup

This section explains how to configure and connect the YLCDRX63N (WQVGA) board.

7.2.1 Jumpers and Switches

Set all jumpers and switches to their default settings: please refer to the YLCDRX63N (WQVGA) documentation (Renesas documentation: YLCDRX63N (WQVGA) Embedded GUI Solution Kit User Manual).

7.2.2 Power Supply

Select your power supply mode: please refer to the YLCDRX63N (WQVGA) documentation (YLCDRX63N (WQVGA) Embedded GUI Solution Kit User Manual).

7.2.3 PC-Board Connection

Plug a serial cable between the PC and the board serial communication interface channel 6 (SCI6). This channel is available on J4 connector (pins 17 and 18). Note the PC's COM port, it will be used later in the application launcher. This allows all traces to be received from the board and printed on the console.

Use a simple serial cable with only data lines (crossed) and signal ground.

Plug a JTAG cable between the Segger J-Link Lite RX and the YLCDRX63N (WQVGA). Plug a USB cable between the PC and YLCDRX63N (WQVGA). Plug a USB cable between the PC and the Segger J-Link Lite RX connector. This allows you to program the board using the Segger J-Link Lite RX .

7.2.4 Summary

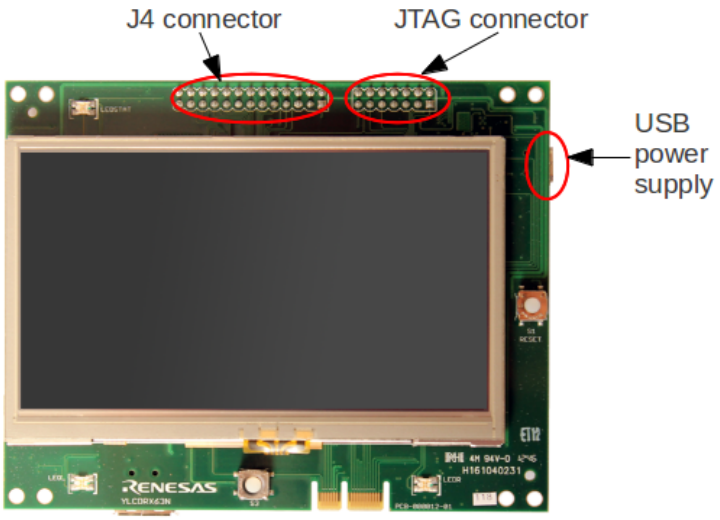

Figure 7.1. Board configuration

7.3 Demos Installation

This chapter explains how to install extra demos and application notes. They have to be imported as a standard Eclipse project archive file:

- Go to the website http://www.is2t.com/download.
- Download zip files containing one or more demos.
- Open MicroEJ.
- Open the wizard Import: `File > Import`.
- Select the item `General > Existing Projects into Workspace`.
- Click on Next.
- Select the downloaded zip file.

8 Document History

Date	Revision	Description
May 22th 2014	A	First release