UM1864
User manual

Getting Started with Games in Java demonstration

Introduction

Games in Java is a graphical Java demonstration, built for STM32F429 Discovery kit and developed with STM32Java (STM32-JAVA-GAMES).

This demonstration includes:

- A Java application layer developed with STM32Java
- A complete Board Support Package Layer for STM32F429 Discovery kit with an MDK-ARM project based on STM32CubeF4

The demonstration is composed of three games:

- Puzzle: a simple sliding puzzle game
- 2048: the famous game originally developed by the web developer Gabriele Cirulli
- Labyrinth: is the classic game where the player controls a steel ball by tilting the board

Contents

List of figures

1 Getting Started

1.1 Scope

The purpose of this user manual is to describe the “Games in Java” demonstration, developed with STM32-Java, built for the board STM32F429 Discovery kit.

The document gives an overview of the demonstration and describes, step by step, how to:

- Download the demonstration on STM32F429 Discovery kit.
- Run the demonstration on STM32Java Simulator.
- Build and download the demonstration on board.

1.2 Intended audience

The intended audience for this document is developers, who wish to develop a Java application with STM32-Java:

- This document is for STM32F429 Discovery kit.
- For more information, visit www.stm32java.com

1.3 Prerequisites

- PC with Windows XP or later
- Hardware requirement:
  - STM32F429 Discovery kit
- Software requirement:
  - STM32-Java 3.0.1 (or higher)
  - MDK-ARM Keil™ uVision4 (v4.7x).MDK-ARM v5 is not yet supported (scheduled in Q2 2015).
  - STM32 ST-LINK Utility (refer to URL *www.st.com*)

To run, edit or compile this demonstration the user has to download the demonstration archive which contains:

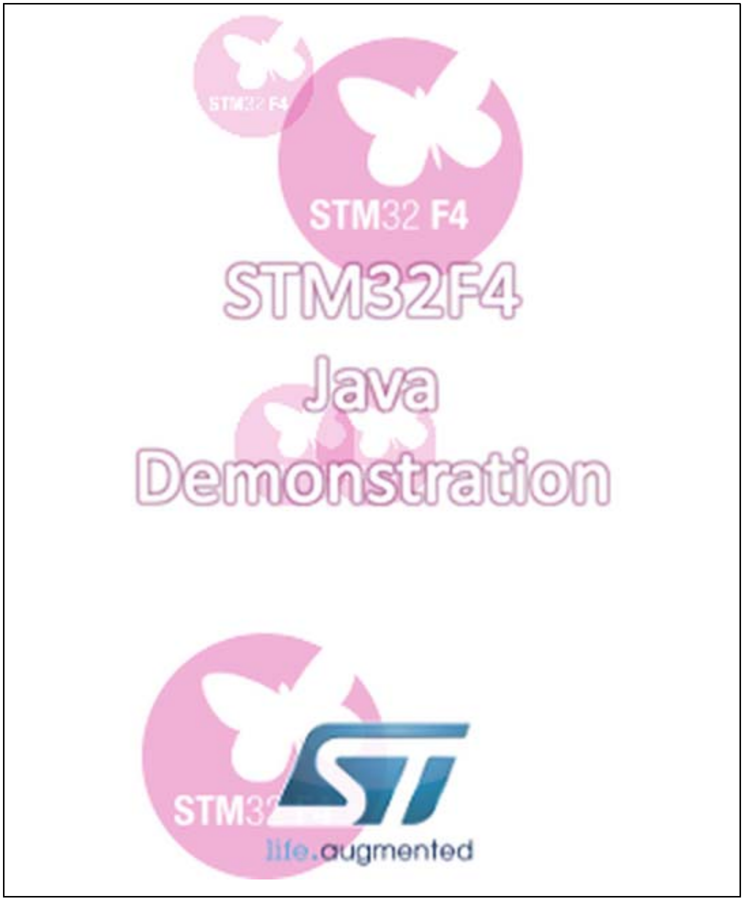
- **Getting Started with Games in Java.pdf**
  The current document
- **GamesInJava.hex**
  The binary format of the demonstration
- **GamesInJava-src.zip**
  The demonstration java source code
- **STM32F429I-DISCO-Cube-CM4_ARMCC-1.0.0.jpf**
  The compiled Java Platform for STM32F429 Discovery kit
- **STM32F429I-DISCO-bsp.zip**
  The board support package (BSP) project containing the source code of Low-Level APIs, ensuring the communication between the Java application and the hardware.

2 Demonstration overview

This demonstration runs on STM32F429 Discovery kit board and benefits from different components, such as the gyroscope and Chrom-ART Accelerator™ for enhanced graphic content creation (DMA2D).

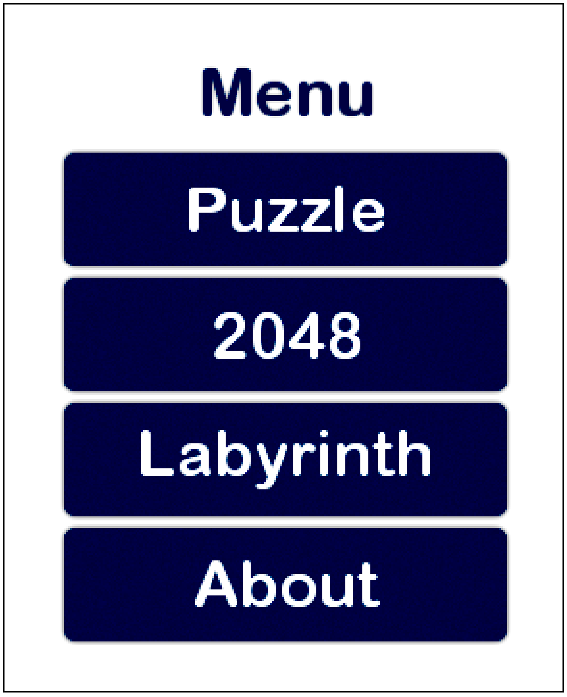
As showed in *Figure 1*, the demonstration starts with a simple splash-screen, containing bouncing logos showing the z-index and transparency capabilities of Chrom-ART Accelerator™:

Figure 1. Games in Java: Splash Screen



A menu can be displayed by touching the screen. This menu allows the user to launch the demonstration sub-applications (*Figure 2*). The navigation through the demonstration menus and games is ensured via touch screen.

Figure 2. Games in Java: Menu



2.1 Puzzle Game

This game is a sliding puzzle (like fifteen-puzzle) that consists of a frame of square tiles in random order with one tile missing.

The objective of the puzzle is to place the tiles in order, by making sliding moves that use the empty space.

The puzzle game can be played in two modes: picture mode (*Figure 3*) or numbered mode (*Figure 4*) with three levels of difficulties (3x3, 4x4 and 5x5) for each mode.

Figure 3. Games in Java: Puzzle 1

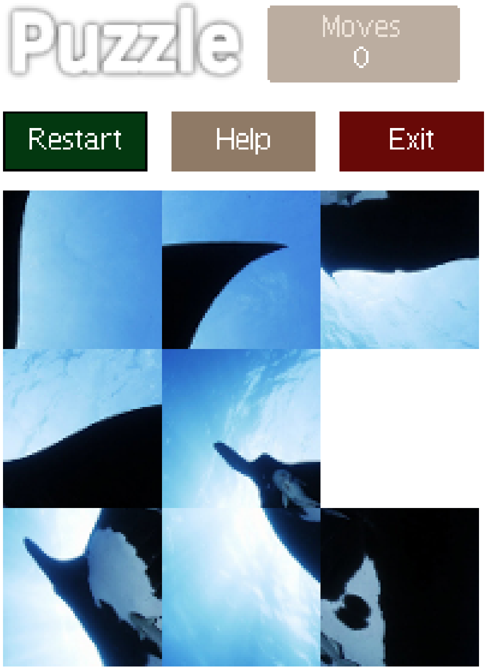
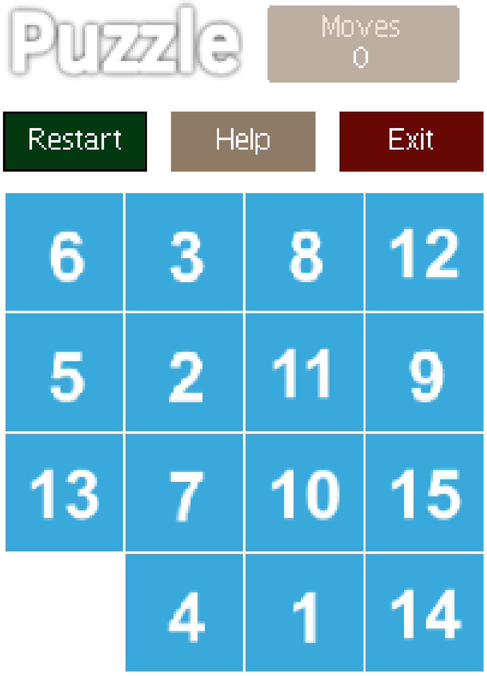


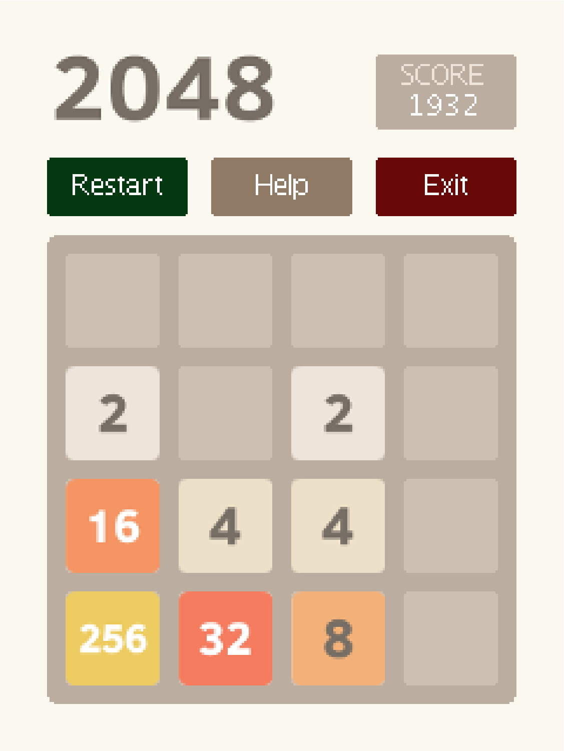
Figure 4. Games in Java: Puzzle 2



2.2 2048 Game

2048 is the famous game originally developed by Gabriele Cirulli, which objective is to slide numbered tiles on a grid, to combine them and create a tile with the number 2048 (*Figure 5*).

Figure 5. Games in Java: 2048

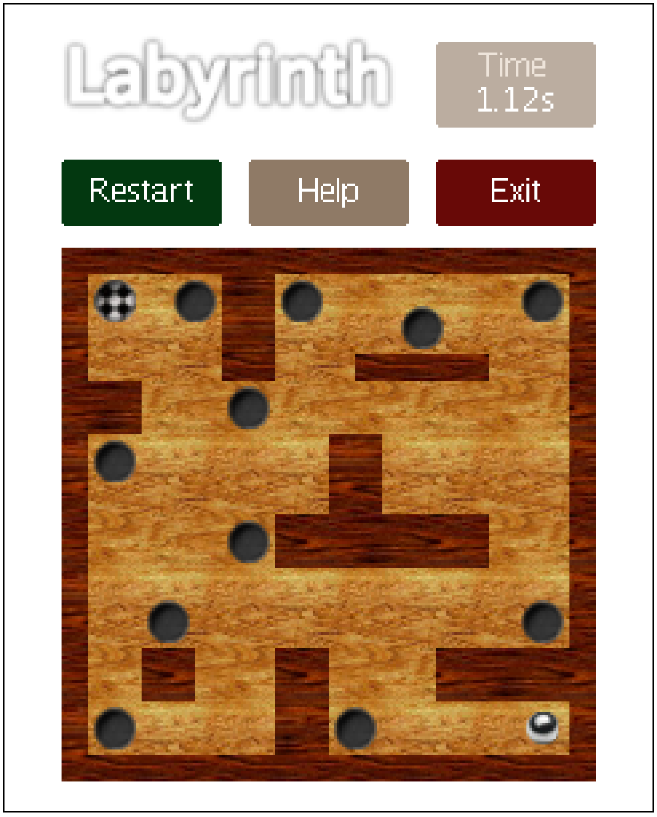


2.3 Labyrinth Game

Labyrinth (*Figure 6*) is the classic game where the player controls a steel ball by tilting a wooden labyrinth. To win the player should move the ball to the target without falling in holes. This game is based on gyroscope sensor available on STM32F429 Discovery kit, for this reason it is not available on the simulator of STM32-Java environment.

Labyrinth game contains three difficulties levels.

Figure 6. Games in Java: Labyrinth



3 Download the demonstration on STM32F429 Discovery kit

To program the demonstration on STM32F429 Discovery kit, the user should:

1. **Power up the board by connecting it to a PC with an USB cable.**
2. **Open the "GamesInJava.hex" within the package with STM32 ST-LINK Utility:**
   1. a) Open STM32 ST-LINK Utility
   2. b) File > Open file (browse for "GamesInJava.hex")
3. **Download the Binary to the board**
   1. a) Target > Program
4. **Reset the board to run the demonstration**
   There are three ways to reset the board:
   1. a) Push the reset button mounted on the Demo board.
   2. b) Unplug then plug the USB cable.
   3. c) The STM32 can also be reset by debuggers.

4 Run the demonstration on simulator

This section provides the needed steps to run the demonstration on STM32-Java Simulator **SimJPF**.

Warning: Important: the Labyrinth game is not available on simulator, since it uses the gyroscope sensor available on the STM32F429 Discovery kit.

1. Open STM32-Java environment
2. Import “STM32F429I-DISCO-Cube-CM4_ARMCC-1.0.0.jpf”
   a) Window > Preferences >
   b) MicroEJ > Available Platforms >
   c) Import > Select file > Browse for “STM32F429I-DISCO-Cube-CM4_ARMCC-1.0.0.jpf”within the package
3. Import the demonstration Java source
   a) File > Import
   b) General > Existing Projects into Workspace
   c) Select archive file > Browse for “GamesInJava-src.zip” within the package
   d) Click “Finish”
4. Run the demonstration
   a) Run > Run Configurations
   b) MicroEJ Application > Select **GamesInJava SimJPF**
   c) Click Run
   – The demonstration runs now on Simulator

5 Build and download the demonstration on board

This section guides the user to build then execute the demonstration on the STM32F429 Discovery kit.

1. Execute Steps from 1 to 3 described in *Section 4: Run the demonstration on simulator*.
2. Import the demonstration Board Support Package - BSP
   a) File > Import
   b) General > Existing Projects into Workspace
   c) Select archive file > Browse for “STM32F429I-DISCO-bsp.zip”
   d) Click Finish
3. Build the Java demonstration
   a) Run > Run Configurations
   b) MicroEJ Application > Select **GamesInJava EmbJPF** > Run
4. Connect the board
5. Build MDK-ARM Project and download the demonstration
   a) From the Package Explorer, open STM32F429I-DISCO-bsp/buildAndDeploy.xml
   b) Check Keil UV4.exe path at line 4 and update the Keil path if needed
   <property name="UV4" location="C:/Keil/UV4/UV4.exe"/>
   c) Run STM32F429I-DISCO-bsp/buildAndDeploy.xml as Ant Build
   Right click on STM32F429I-DISCO-bsp/buildAndDeploy.xml > Run As > 1 Ant Build
   This Ant Script will call the MDK-ARM executable to build the board support package and link it with the demonstration output file “javaapp.o” then the Ant Script call MDK-ARM to load the binary image to the STM32F429 Discovery kit
   d) Congratulations the demonstration runs on the board!
   e) Enjoy the demonstration!

6 Revision history

Table 1. Document revision history

Date	Revision	Changes
05-Mar-2015	1	Initial release.
05-May-2015	2	Updated *Section : Introduction*