



Press Your Button for Raspberry Pi

Created by Roberto Marquez



<https://learn.adafruit.com/press-your-button-for-raspberry-pi>

Last updated on 2022-12-01 02:39:12 PM EST

Table of Contents

Demo Video & Overview	3
Assembly	3
Software Prerequisites	6
<ul style="list-style-type: none">• Java 8 and the Latest JavaFX APIs• OpenJDK Overlay of JavaFX APIs	
Application Installation	7
Usage	8
<ul style="list-style-type: none">• Background• Playing a Game• End of Run	
Going Further	9
Resources	9
<ul style="list-style-type: none">• Source Code• Build Server• JavaFX Screen Buffer on Raspberry Pi	

Demo Video & Overview

Remember the game show "Press Your Luck"? You can now play a version at home with this DIY Raspberry Pi project. The project uses Raspberry Pi GPIO to detect button presses for game of chance and display the game board on a TV

JavaFX API were used to present the user interface on the Raspberry Pi's frame buffer.

Java Standard Edition for Embedded Devices was used as the implementation platform.

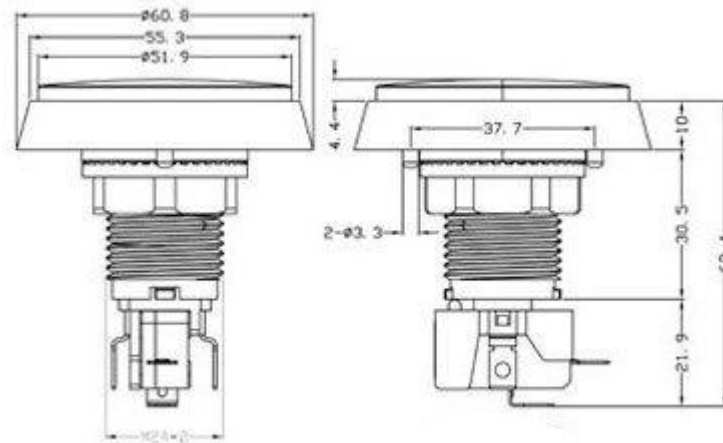
Assembly

This project uses the following hardware:

- [Raspberry Pi Model B \(http://adafru.it/998\)](http://adafru.it/998) (or better)
- [Arcade Button \(http://adafru.it/1194\)](http://adafru.it/1194)



Here the hardware diagram of the arcade button.



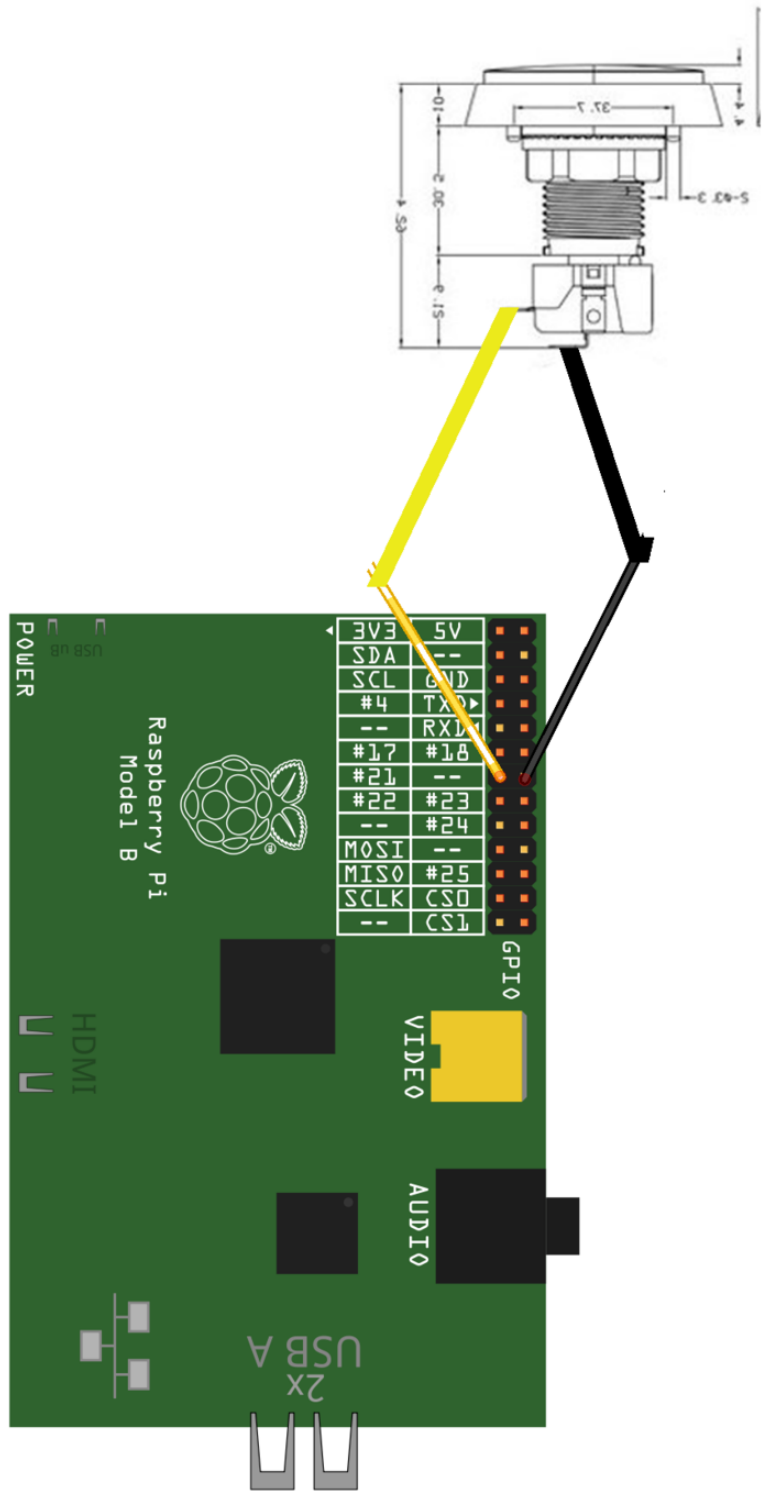
Wiring the arcade button for input into the Raspberry Pi is as follows:

The two connections on either of the arcade button's long sides are for powering the built-in LED. These connections are seen on the bottom left, in the diagram above. Connect these to the Raspberry Pi's power and ground sources if you want the LED lit.

The connection on the very bottom is wired to Ground on the Raspberry Pi. This connection is seen on the bottom-right side of the diagram above.

The button connection that sticks out of the housing, in the same direction as the Ground connection, is connected to pin 13, on the Raspberry Pi model B Revision 2, pin header. So if looking down at the Raspberry Pi with the header in the top right corner, then the input pin connection is to 7th pin down on the left side.

Here is the wiring diagram for the project:



Made with  Fritzing.org

Software Prerequisites

Java 8 and the Latest JavaFX APIs

This project requires Java 8 and specifically the JavaFX API version introduced with Java SE build 1.8.0u40. The latest version of Raspbian comes with Java 8, but unfortunately it does not include the latest JavaFX APIs.

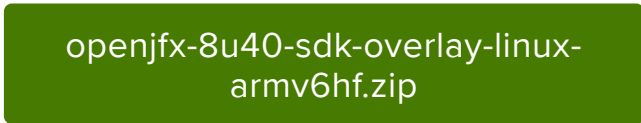
The good news is that an overlay is available. The overlay updates the built-in JavaFX APIs to the latest.

OpenJDK Overlay of JavaFX APIs

A good old chap, by the name of [Chris Newland \(\)](#), has provided the latest OpenJDK JavaFX APIs as an overlay for the Java SE Runtime Environment that ships with Raspbian.

If you want the nightly build, then head over to <http://108.61.191.178/> () and select a 'nightly' build from the options under Download OpenJFX builds for Linux ARMv6 hard float (Raspberry Pi etc.).

Otherwise, download this local copy of the 'OpenJFX 8u40 stable for armv6hf' overlay:



openjfx-8u40-sdk-overlay-linux-
armv6hf.zip

When the download for the overlay is complete, copy it (the whole zip file) to the Raspberry Pi.

Log into the Raspberry Pi, as the pi user.

At this point, you are logged into the Raspberry Pi. Now we need to find the destination for the overlay; that is the directory of the Java Runtime on the filesystem.

On the Pi used for this project, you can see below that java really points to `/usr/lib/jvm/jdk-8-oracle-arm-vfp-hflt/jre/bin/java`

```
pi@raspberrypi:~ $ which java
/usr/bin/java

pi@raspberrypi:~ $ ls -la /usr/bin/java
lrwxrwxrwx 1 root root 22 Nov 21 20:54 /usr/bin/java -> /etc/alternatives/java

pi@raspberrypi:~ $ ls -la /etc/alternatives/java
lrwxrwxrwx 1 root root 51 Nov 21 20:54 /etc/alternatives/java -> /usr/lib/jvm/
jdk-8-oracle-arm-vfp-hflt/jre/bin/java
```

From the information above, the following is used for the destination of the OpenJDK JavaFX APIs overlay.

- /usr/lib/jvm/jdk-8-oracle-arm-vfp-hflt/

Next, change directory to where the overlay ZIP file was copied.

Issue this command to perform the overlay:

```
sudo unzip openjfx-8u40-sdk-overlay-linux-armv6hf.zip -d /usr/lib/jvm/jdk-8-oracle-
arm-vfp-hflt/
```

Inflating messages are printed next. Answer 'A' to confirm replacement of existing files.

If no error messages are shown, then the overlay is complete :)

Application Installation

Download the application's executable JAR:

press-your-button-fx-0.1-
SNAPSHOT.jar

Copy the JAR file to the Raspberry Pi.

Run the application with the following command:

```
sudo JAVAFX_DEBUG=1 java -jar press-your-button-fx-0.1-SNAPSHOT.jar
```

If everything went well, then the following is printed to the console:

```
*Provisioning GPIO.  
*GPIO provisioned :)
```

Usage

Background

This game is similar to "Press Your Luck", but differs in several ways.

- single or multiplayer (1-3 players)
- winner is determined by the first player to reach a pre-determined score
- Whammys only remove half the current player's score, and not the whole thing

Playing a Game

Pressing the 'New Game' button brings up a configuration dialog for the next game.

NOTE: If you see the following error message when you click the 'New Game' button, then ensure the instructions in the 'OpenJDK Overlay' section were performed correctly.

```
Caused by: java.lang.NoClassDefFoundError: javafx/scene/control/Alert  
    at  
    org.onebeartoe.games.press.your.button.fx.FXMLController.handleNewGameButton(FXMLController.java:249)  
    ... 61 more  
Caused by: java.lang.ClassNotFoundException: javafx.scene.control.Alert
```

One to three players are supported.

The target score determines how high the winning score is.

To stop the score board, press either the Stop button on the application's GUI or the arcade button attached to the Raspberry Pi.

Once a player has stopped the score board, the current player's selected panel is shown in RED lettering. If a point value (number) is shown in RED, then that amount is added to the current player's score. If a 'W' is shown in RED, then the current player loses half his score. Whammy!

After a player has taken their turn, press the 'Next Player' button to continue the game.

The first player to reach the target score wins!

End of Run

To quit the game altogether, hit Ctrl+C on the keyboard.

Going Further

This project is open source. You can modify it to meet your preferences.

Some ideas to extend the application include:

- change the game sounds
 - support for more than 3 players
 - add an animation when a Whammy occurs
 - use LEDs as indicators for winner/whammy
-

Resources

Source Code

- <https://github.com/onebeartoe/press-your-button/> ()

Build Server

- <https://onebeartoe.ci.cloudbees.com/job/press-your-button/> ()

JavaFX Screen Buffer on Raspberry Pi

- - Basic Example - <http://javafx.steveonjava.com/javafx-on-raspberry-pi-3-easy-steps/> ()
 - Clock Example - <http://www.oracle.com/webfolder/technetwork/tutorials/obe/java/RaspberryPiFX/raspberrify.html> ()

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [Adafruit Accessories](#) category:

Click to view products by [Adafruit](#) manufacturer:

Other Similar products are found below :

[3209](#) [2974](#) [3562](#) [2503](#) [3011](#) [3048](#) [2973](#) [2868](#) [2958](#) [2836](#) [2968](#) [3610](#) [3568](#) [2858](#) [3585](#) [3584](#) [3484](#) [2865](#) [3556](#) [3262](#) [3348](#) [2499](#) [2878](#)
[2963](#) [3219](#) [298](#) [2194](#) [862](#) [460](#) [905](#) [02-LDR1](#) [02-LDR12](#) [02-LDR13](#) [02-LDR14](#) [02-LDR20](#) [02-LDR21](#) [02-LDR22](#) [02-LDR23](#) [02-LDR4](#)
[1008](#) [1020](#) [1031](#) [1054](#) [1056](#) [1065](#) [1071](#) [1077](#) [1088](#) [1094](#) [1095](#)