

(2) 外部依赖。

本工程界面使用 OpenFX11 写成，外部依赖项包括 org.openjfx 中的
javafx-base-11-win.jar、
javafx-graphics-11-win.jar、
javafx-controls-11-win.jar、
javafx-fxml-11-win.jar。

```

buildscript {
    repositories {
        mavenCentral()
    }
    dependencies {
        classpath 'com.google.gradle:osdetector-gradle-plugin:1.6.0'
    }
}

apply plugin: 'java'
apply plugin: 'application'
apply plugin: 'com.google.osdetector'

ext.platform = osdetector.os == 'osx' ? 'mac' : osdetector.os == 'windows' ? 'win' : osdetector.os

repositories {
    mavenCentral()
}

dependencies {
    compile "org.openjfx:javafx-base:11:$platform"
    compile "org.openjfx:javafx-graphics:11:$platform"
    compile "org.openjfx:javafx-controls:11:$platform"
    compile "org.openjfx:javafx-fxml:11:$platform"
}

compileJava {
    doFirst {
        options.compilerArgs = [
            '--module-path', classpath.asPath,
            '--add-modules', 'javafx.controls,javafx.fxml'
        ]
    }
}

```

(3) 类结构

1. **Controllers** 类：用于控制页面元素与用户的交互

变量 **duration**：液位波动过渡动画的持续时间；

变量 **timelines**：用于保存液体和气体流动动画的 **Timeline**，在动画更新时进行清除；

Controllers 类其余变量的作用已在第一部分介绍；

函数 **initialize()**（源代码见一-2）：由系统调用，对控制类中的变量进行初始化

Controllers 类其余函数的作用已在第一部分介绍。

- ▼ JFX1
 - ▼ src/main/java
 - ▼ HelloFX
 - ▼ Controllers.java
 - ▼ Controllers
 - duration
 - flowNum1
 - flowNum2
 - flowNum3
 - flowNums
 - h1Text
 - h2Text
 - h3Text
 - hMax
 - hTexts
 - lArrow01
 - lArrow02
 - lArrow03
 - lArrow1
 - lArrow2
 - lArrow3
 - liquidRectangle
 - liquidRectangle1
 - liquidRectangle2
 - liquidRectangle3
 - lPath01
 - lPath02
 - lPath03
 - lPath0a
 - lPath0b

- IPath1
- IPath11
- IPath12
- IPath13
- IPath2
- IPath21
- IPath22
- IPath23
- IPath24
- IPath25
- IPath3
- IPath31
- IPath32
- IPath33
- IPath34
- maxHeight1
- maxHeight2
- maxHeight3
- ▲ maxHeight
- outText1
- outText2
- p1Text
- p2Text
- p3Text
- pArrow1
- pArrow2
- pArrow3
- pMax
- pMin
- pPath1

- pPath11
- pPath2
- pPath21
- pPath3
- pPath31
- pPath32
- pTexts
- ▣ timelines
- valveSliderh
- valveSliderp
- ▣ arrowUpdate(Polyline, double, double[][]) : void
- ▣ arrowUpdate(Polyline, double, double[], int) : void
- ▣ bucketMove(int, double, double) : void
- 🌐 initialize(URL, ResourceBundle) : void
- valveValueChange() : void

2.HelloFX 类（主类）：程序的入口点、读取页面设计、加载 JavaFX 界面设计文件、创建窗口进行显示。

```
▼ Main.java
  ▼ Main
    ● main(String[]) : void
    ● start(Stage) : void
```

Main()函数（源代码见一-1）：程序入口点，调用由 javafx.application.Application.launch()函数加载窗口。

Start()函数（源代码见一-1）：由 javafx.application.Application.launch()调用，加载 JavaFX 界面设计文件并创建窗口进行显示。

（4）构建结构：

本工程除编译生成.class 文件外，还使用 Gradle 的打包功能将源程序连同 openjfx 库一同编译打包至 libs/JFX1.jar 文件中。

Gradle 编译及打包脚本：

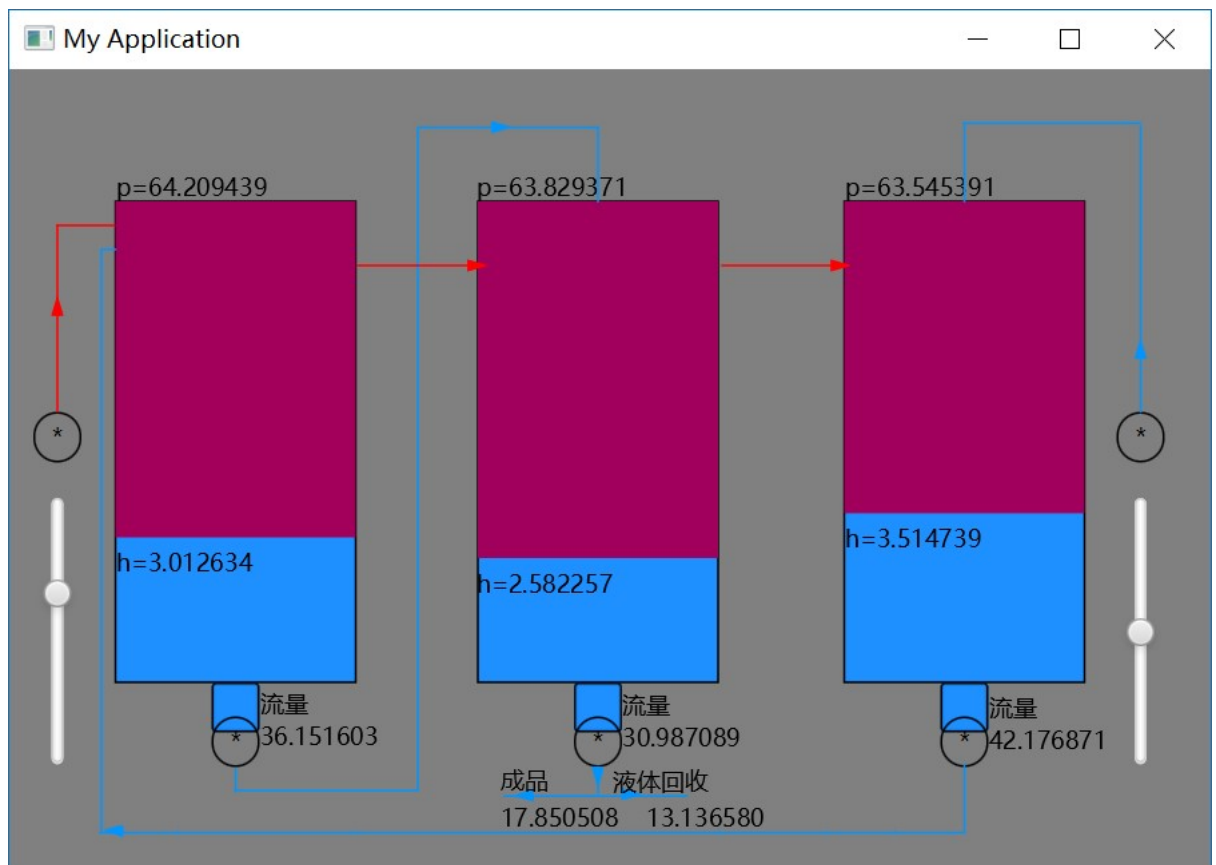
```
27 compileJava {
28     doFirst {
29         options.compilerArgs = [
30             '--module-path', classpath.asPath,
31             '--add-modules', 'javafx.controls,javafx.fxml'
32         ]
33     }
34 }
35
36 run {
37     doFirst {
38         jvmArgs = [
39             '--module-path', classpath.asPath,
40             '--add-modules', 'javafx.controls,javafx.fxml'
41         ]
42     }
43 }
44
45 jar {
46     from {
47         configurations.compile.collect { it.isDirectory() ? it : zipTree(it) }
48     }
49     manifest {
50         attributes 'Main-Class': 'HelloFX.Main'
51     }
52 }
53
54 mainClassName = 'HelloFX.Main'
```

Build 文件夹结构：

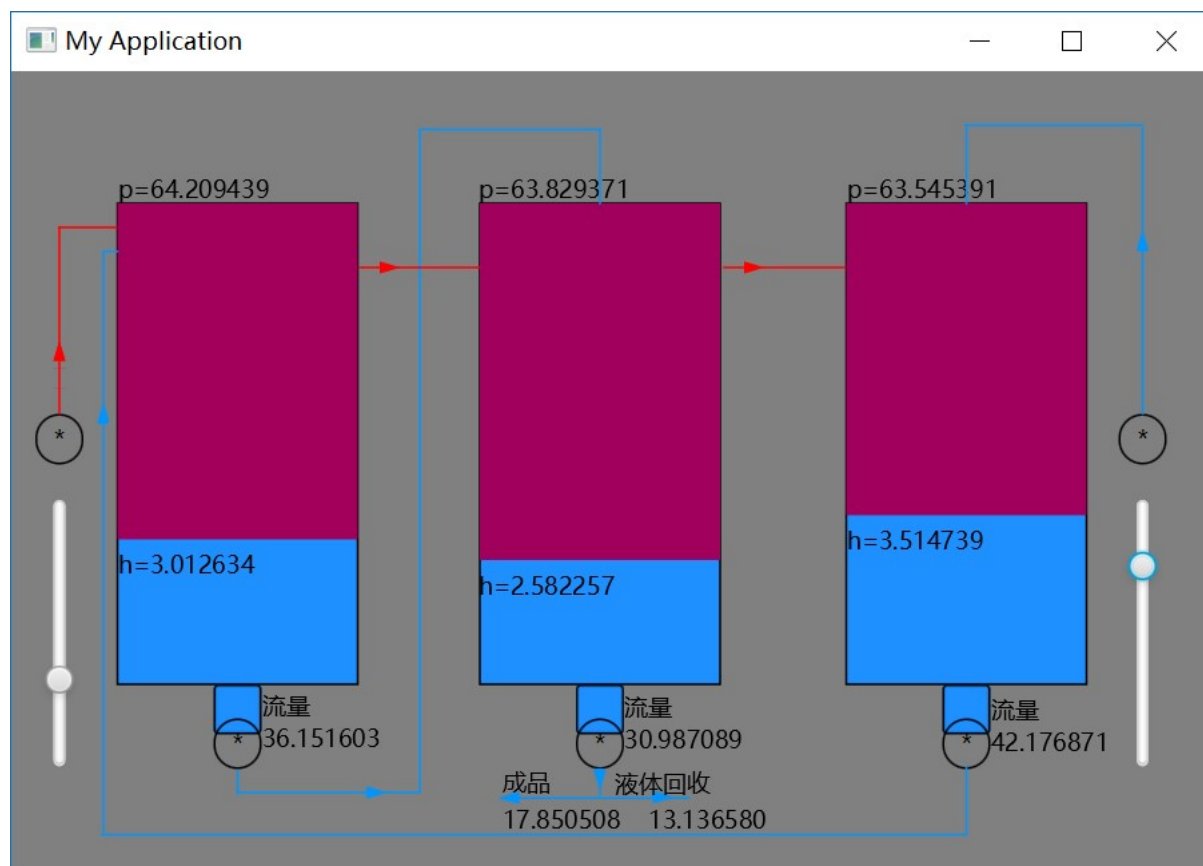
- ▼ build
 - ▼ classes
 - ▼ java
 - ▼ main
 - ▼ HelloFX
 - Controllers.class
 - Main.class
- ▼ distributions
 - JFX1.tar
 - JFX1.zip
- ▼ libs
 - JFX1.jar
- ▼ resources
 - ▼ main
 - MainPage.css
 - MainPage.fxml
- ▼ scripts
 - JFX1
 - JFX1.bat
- ▼ tmp
 - compileJava
- ▼ jar
 - MANIFEST.MF

三、程序运行截图

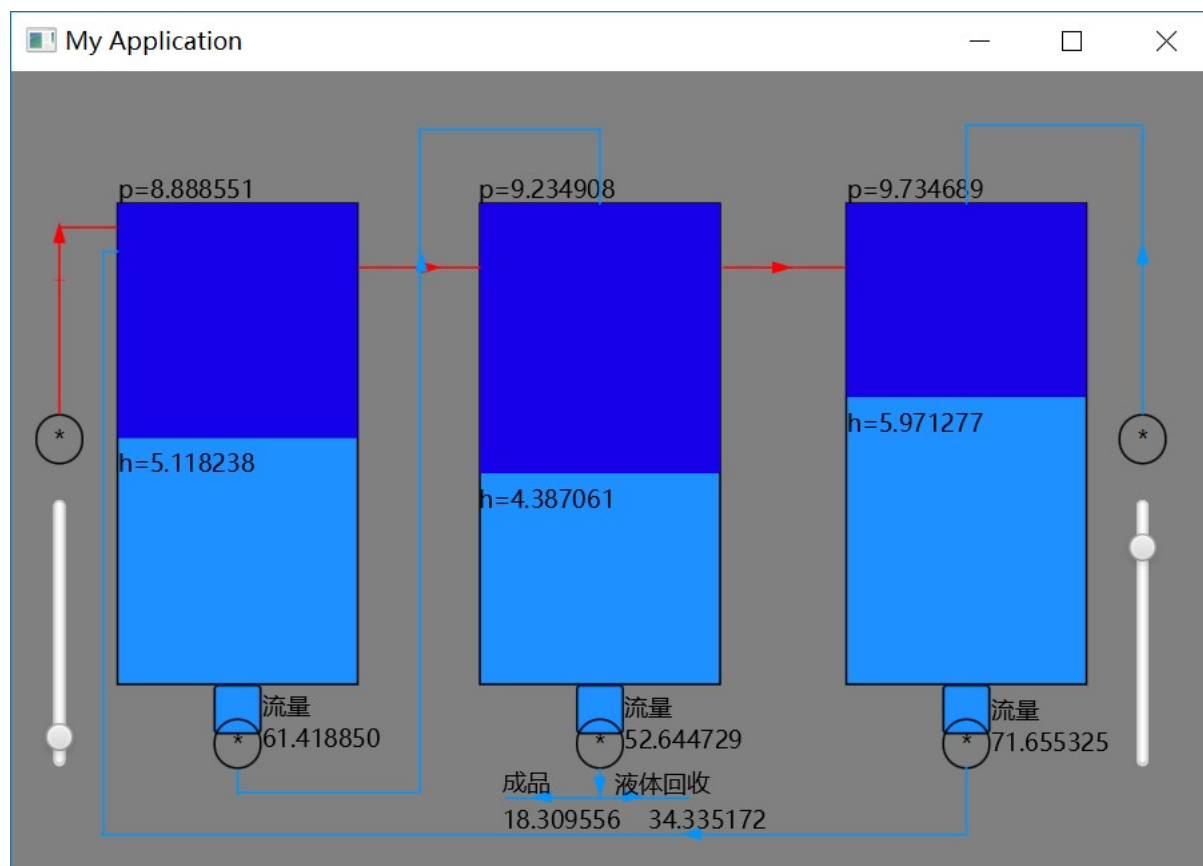
正常运行：



正常运行:



液体过量:



气体过量

