# Wonton软件用户手册



**Version 1.1.2**

2021年9月

## 第一章：软件介绍

Wonton是一个开源多平台的FPGA虚拟外接设备软件。Wonton软件针对复旦大学系列可编程逻辑器件开发，是FDE系统的一部分。软件使用USB协议与FPGA进行数据交互，模拟多种外接器件，当前版本集成了开源波形调试软件GTKWave。

软件的基础界面如下：



## 第二章：软件安装

### 2.1 Wonton软件安装

Windows：

1.双击Wonton-1.1.2-win-setup，运行安装程序即可进行安装。

2.也可以解压Wonton-1.1.2-win-x64.7后进入目录下直接运行Wonton。

Max：

待补充。

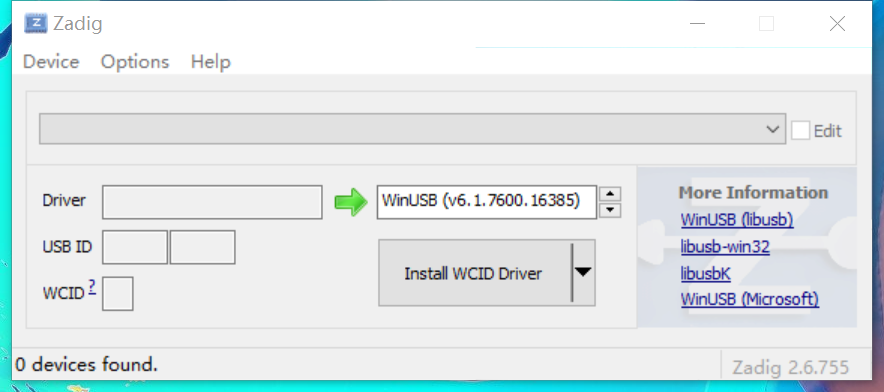
Linux：

待补充。

### 2.2 驱动安装

Windows用户可使用Zadig安装最新驱动。MacOS和Linux用户可直接使用，故跳过此步骤。

将FPGA板的USB口接入电脑后打开Zadig.exe，此时看到的界面如下：



直接点击Install后，打开设备管理器，看到如下设备说明驱动安装成功。



## 第三章：软件使用简介

### 3.1 开始菜单



#### 3.1.1 “新建项目”按钮

按下后会弹出以下页面：



项目名称和地址由用户决定，特别注意名称和路径中尽量不要包含汉字或特殊符号，以避免发生不必要的错误。新建的项目将以.hwproj的格式保存在指定的目录下。引脚约束文件xml在用户得到DC综合出的门级网表文件经由HDLAutoAssign生成。

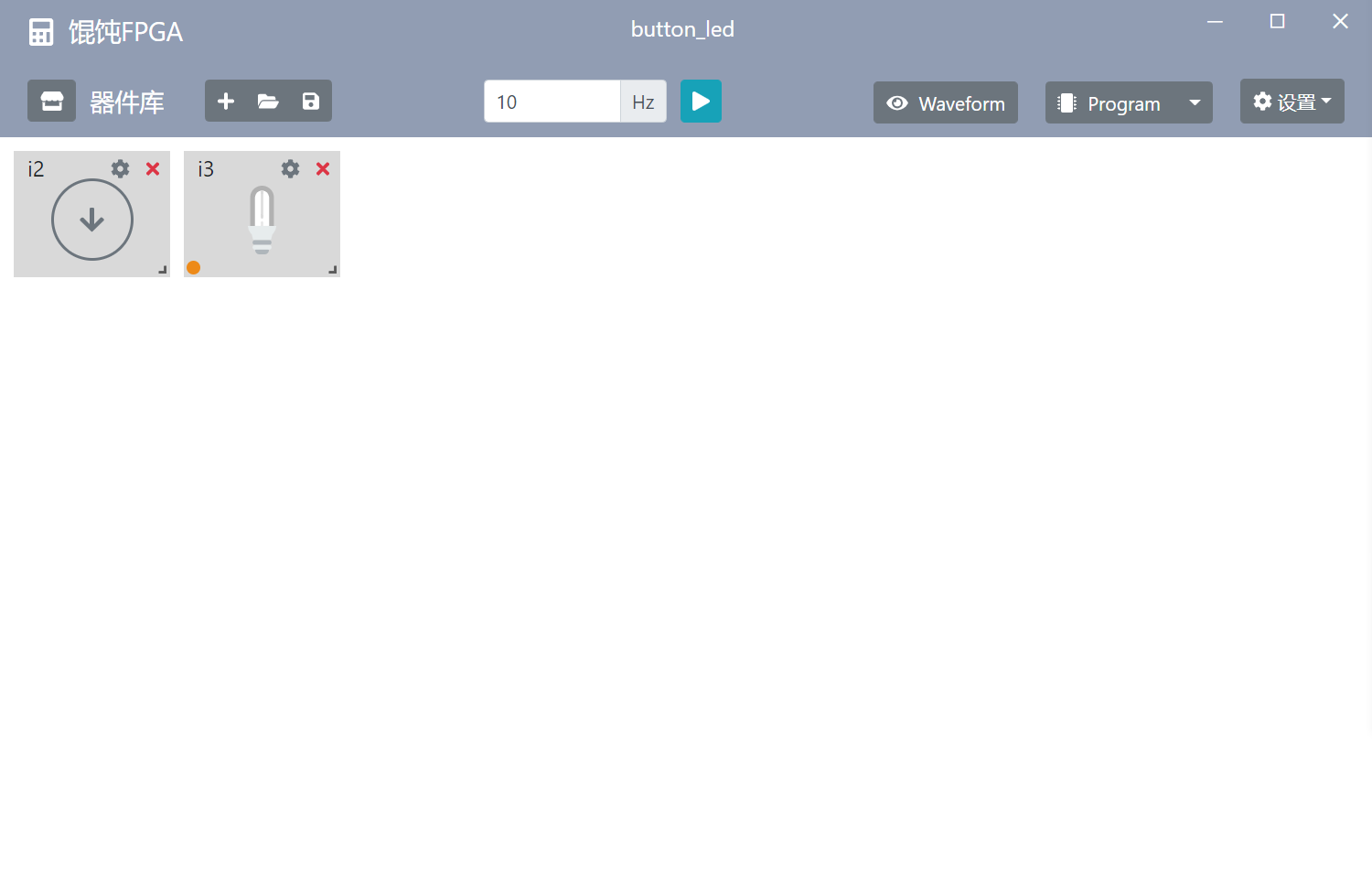
#### 3.1.2 “打开项目”按钮

打开已有的.hwproj项目。.hwproj文件中存储了器件位置，引脚接线和位流文件位置信息。在移动了项目中文件的位置后特别注意是否需要重新指定位流文件位置。

#### 3.1.2 打开最近项目

开始菜单右侧记录了最近打开过的项目的位置。

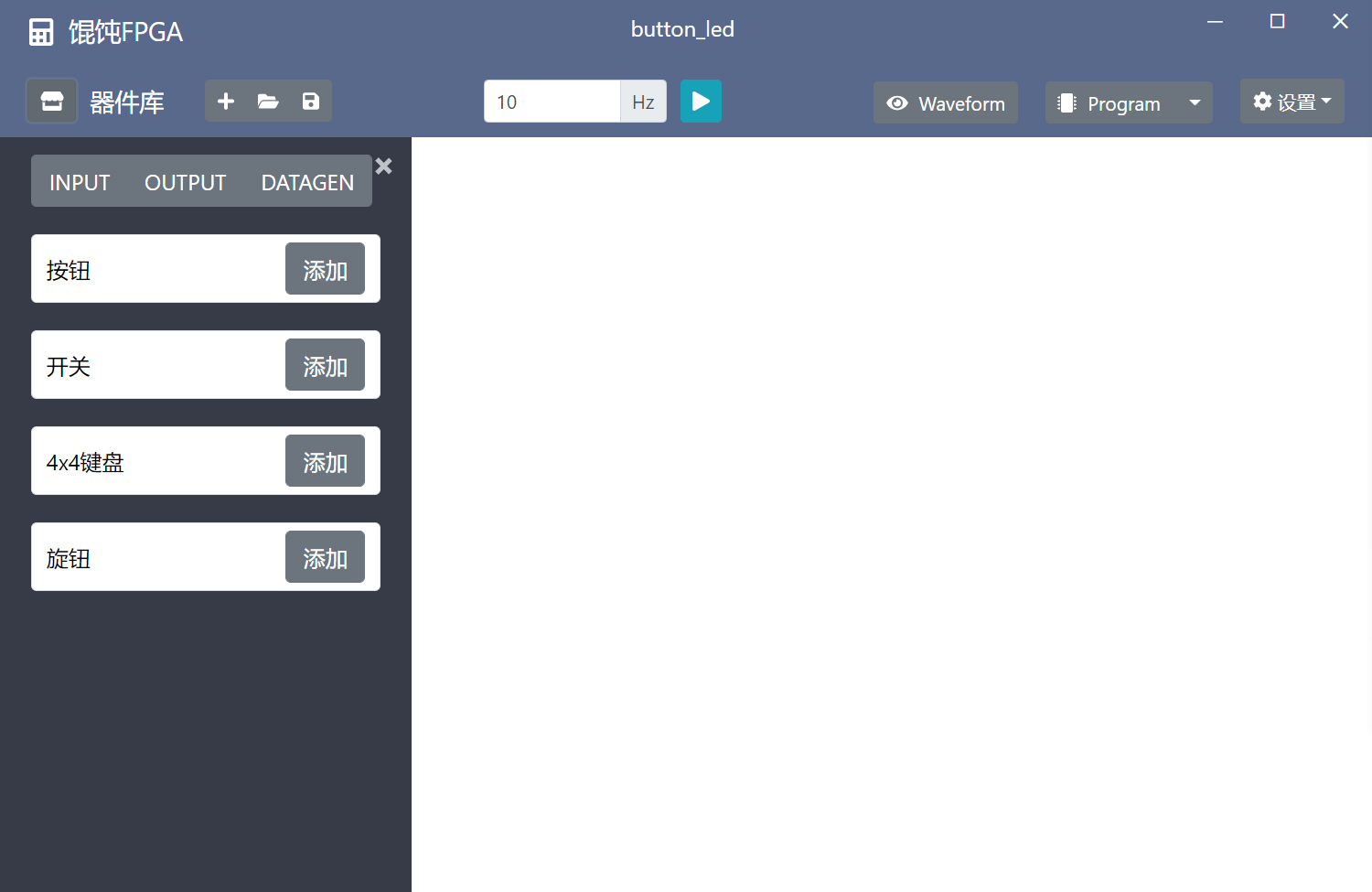
### 3.2 主界面介绍



软件顶端居中位置是项目名称，如本例中项目名称为button\_led。

#### 3.2.1 器件库

按下顶栏左侧第一个按钮调出器件库：

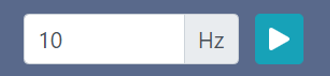


按下INPUT，OUTPUT和DATAGEN切换器件种类。按下添加按钮对应器件将自动添加到主界面的白色区域。

#### 3.2.2 文件管理

三个按钮分别对应“新建项目”，“打开项目”和“保存项目”。

#### 3.2.3 运行项目

 输入数字可以调整计算机生成的时钟的频率，计算机生成的时钟会自动接在clk端口上。点击频率栏右侧按钮可以开始或终止运行。

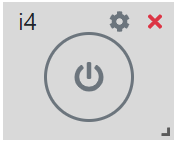
#### 3.2.4 Waveform

点击后自动调出波形调试软件GTKWave（详见第四章），请注意要先运行项目，不然不能生成波形文件。

#### 3.2.5 Program

点击该按钮的下拉菜单，点击“选择文件”选择由DC生成的对应的位流文件，如果之前保存过项目下拉栏中会显示之前保存的位流文件位置。点击Program将位流文件写入FPGA板中。

#### 3.2.6 器件



左上角显示器件编号，右上角分别是设置按钮和移除器件按钮，每个器件都可以拖动，按住右下角可调整器件占用面积。

点击器件右上角的设置按钮会弹出以下界面：



用户可以在此将每个管脚连接到对应的端口上。

#### 3.2.7 设置

设置下拉栏中有三个功能。黑暗模式（暂未优化完全），开发者工具，“关于”按钮（点击进入当前维护Wonton软件的开发人员Github主页）。

### 3.3 器件介绍

#### 3.3.1 INPUT器件

##### 按钮：



位宽：1

功能：默认输出0，按下鼠标输出1，松开输出0

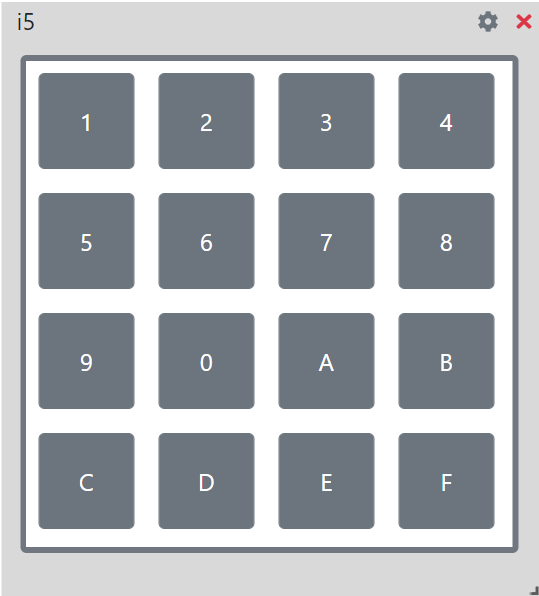
##### 开关：



位宽：1

功能：默认输出0，点击后变化开关状态

##### 4x4键盘

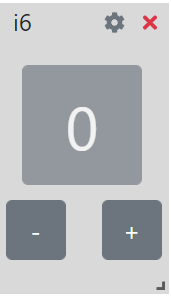


位宽：8

功能：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 按下 | 行[0] | 行[1] | 行[2] | 行[3] | 列[0] | 列[1] | 列[2] | 列[3] |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 5 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 9 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| A | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| B | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| C | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| D | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| E | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| F | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

##### 旋钮



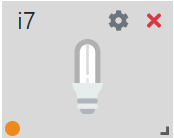
位宽：11

功能：点击+ -按钮调整器件目前状态

|  |  |
| --- | --- |
| 状态 | RPB1,2,3,4,5,6,7,8,9,A,B |
| 0 | 00000000000 |
| 1 | 10000000000 |
| 2 | 01000000000 |
| 3 | 00100000000 |
| 4 | 00010000000 |
| 5 | 00001000000 |
| 6 | 00000100000 |
| 7 | 00000010000 |
| 8 | 00000001000 |
| 9 | 00000000100 |
| 10 | 00000000010 |
| 11 | 00000000001 |

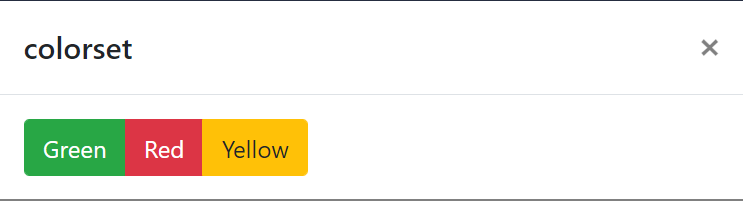
#### 3.3.2 OUTPUT器件

##### LED



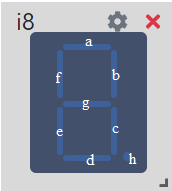
位宽：1

功能：输入为1时点亮，为0时不亮。左下角显示LED颜色，点击该按钮弹出窗口：



可改变LED灯亮起时的颜色。

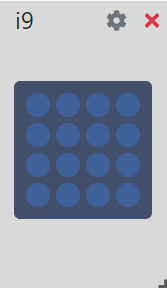
##### 七段数码管



位宽：8

功能：对应的位置输入为1时点亮，为0时不亮。

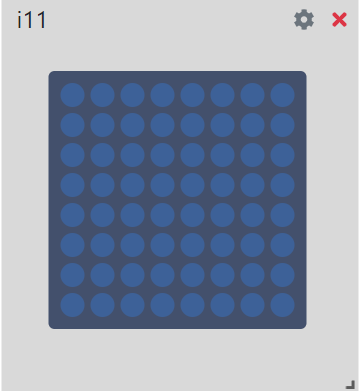
##### LED阵列4x4



位宽：8

功能：对应的行信号和列信号均为1时对应位置的LED点亮

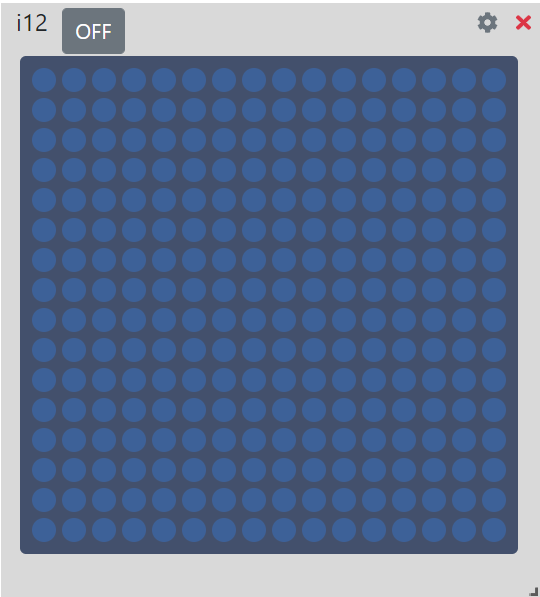
##### LED阵列8x8



位宽：16

功能：对应的行信号和列信号均为1时对应位置的LED点亮

##### LED阵列16x16

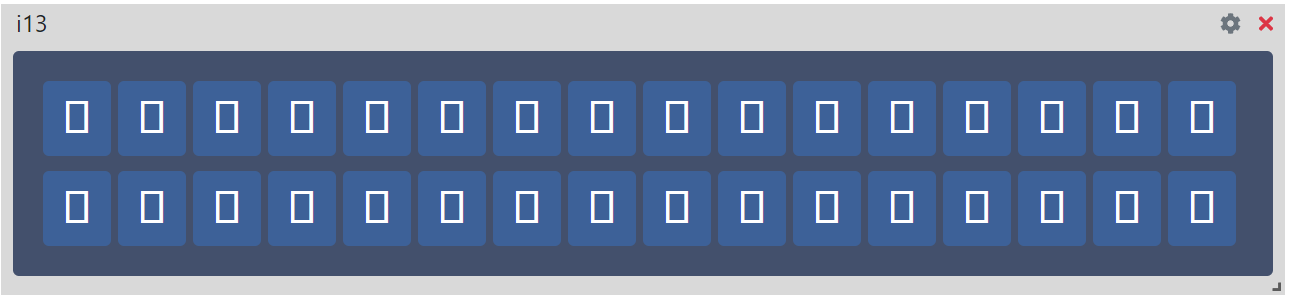


位宽：32

功能：对应的行信号和列信号均为1时对应位置的LED点亮。

该器件支持模拟视觉残留现象，可用于扫描显示。点击左上角的按钮可选择模拟视觉残留现象功能的开关，设定的模拟视觉残留时间是500ms。

##### LED文字显示

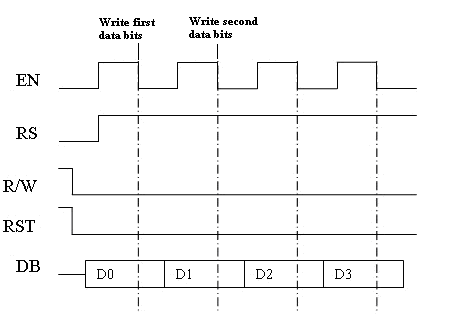


位宽：12

功能：

|  |  |
| --- | --- |
| 端口 | 功能 |
| DB0~DB7 | 输入的数据 |
| RS | Register select  为0时选择指令寄存器  为1时选择数据寄存器 |
| RW | Read/Write  为0时进行写操作  为1时不进行操作 |
| EN | EN下降沿传入数据 |
| RST | RESET  为1时清楚所有显示数据并将光标移动到初始  位置（左上角） |

时序的工作模式：



用户也可根据第五章的lab5加深对该器件的理解。

### 3.3.3 DATAGEN器件

##### ROM



位宽：32

功能：点击setup，弹出窗口：



选择txt文件向ROM的各个引脚写入数据。Setup菜单中会显示当前的输出。

支持的格式为二进制数，十进制数和十六进制数。

txt文本内容应如下格式：

‘b11110000111100001010010100001111

‘b1111 1111 1111

‘d128

‘h02ff

### 3.4 使用注意事项

Wonton是一个正在开发中的项目。

项目的发起人：<https://github.com/WangyuHello/Wonton>

项目当前的维护人：<https://github.com/Hi2129/Wonton_master>

如果遇到bug和功能错误，请到项目当前维护人的主页上提交bug，也欢迎同学前来进行更深一步的学习。

**未来的优化方向：**

**·**.hwproj新增保存器件的配置，如LED的颜色，ROM读取的txt文件的位置

·优化黑夜模式

·考虑添加图像显示LCD器件

**已知的BUG：**

·文件的路径和名称中不能有“+”号，建议项目名称和路径不要带有特殊符号，后续会在输入栏提醒

·打开已有项目，再新增器件时可能出现器件重叠现象，如果出现请删掉器件重新添加，后续会优化

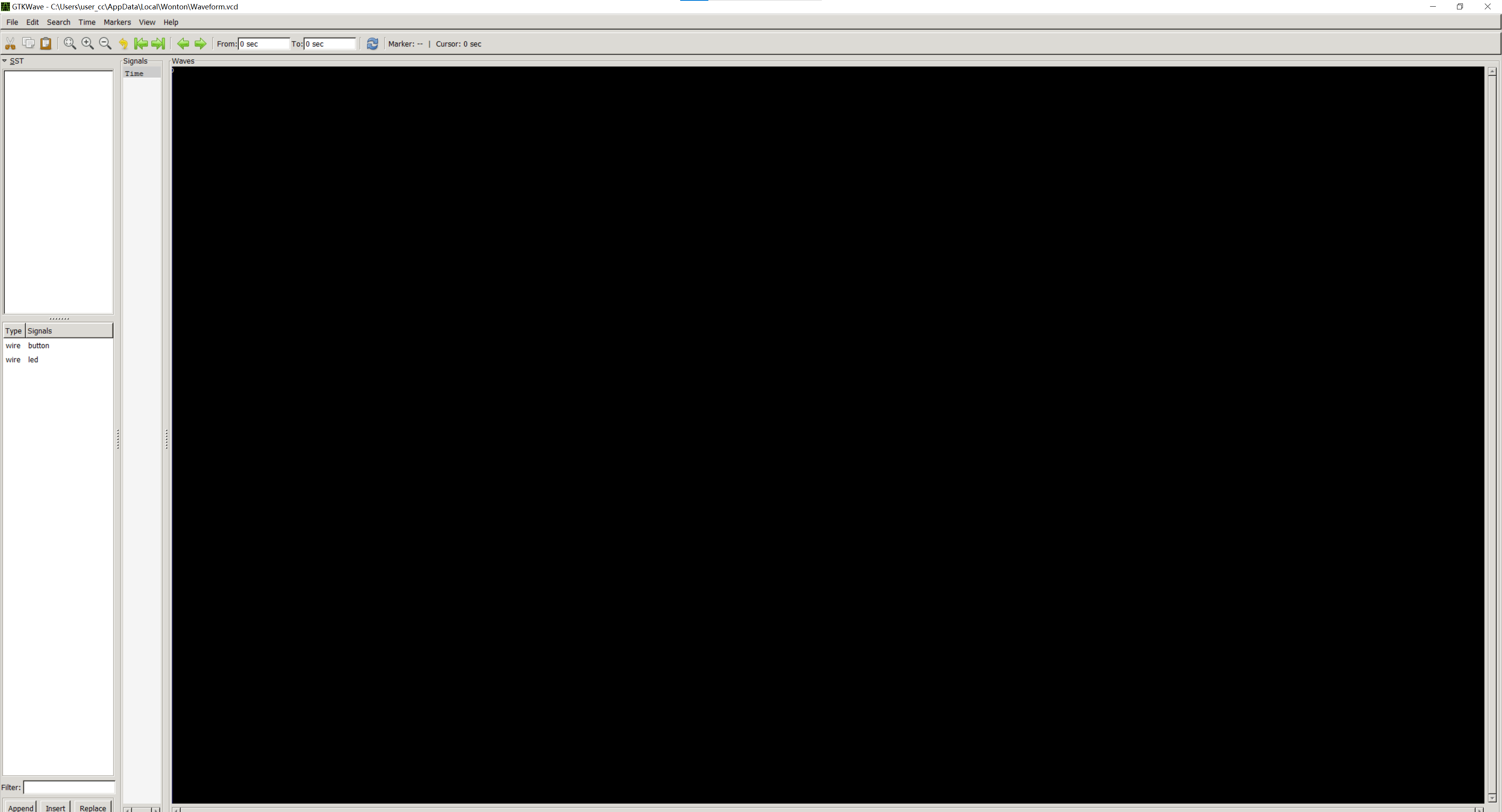
## 第四章：GTKWave使用简介

### 4.1GTKWave简介

GTKWave 是为了在芯片上的大型系统上执行调试任务而开发的，并且被用作第三方调试工具的替代品。对于 Verilog，GTKWave 能宏观显示波形，也能通过注释信号来调试 RTL 时序。

在安装Wonton时会自动安装GTKWave。Windows系统中GTKWave会自动安装在Wonton安装目录下的\resources\bin\gtkwave下，Mac系统下请手动将GTKWave安装在Desktop上。如果不能直接调起的话请手动打开GTKWave。Windows系统下Wonton生成的波形文件在LocalAppData/Wonton文件夹下，Mac/Linux系统下Wonton生成的波形文件在home/.wonton隐藏文件夹下。

### 4.2基本使用



如上图，在Wonton中运行程序后，点击Wavaform进入GTKWave，将左下角信号拖动到Signal栏即可查看此段时间内的运行波形。

更详细的使用指南见附录：GTKWave使用文档\_中文，或GTKWave官网：<http://gtkwave.sourceforge.net/>

## 第五章：使用示例

**Lab0:**

简单的一个 开关/按钮 加一个LED灯 的lab。

**Lab1：**

4x4键盘 的测试lab，可用于熟悉键盘的输出。

**Lab2：**

16x16 LED阵列的测试lab。

**Lab3：**

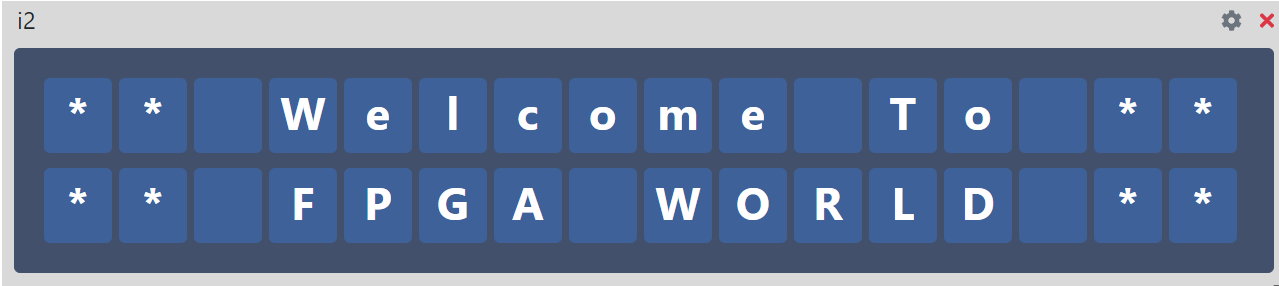
ROM的测试lab，用于熟悉ROM的工作机制。

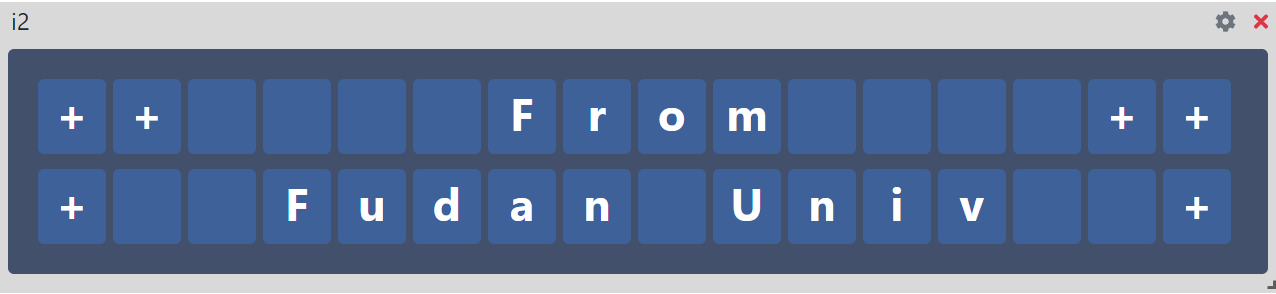
**Lab4：**

旋钮的测试lab。

**Lab5：**

文字显示器件的lab，用于熟悉其工作机制。将扫描显示以下内容：





**Lab6：**

旋钮和七段数码管的测试lab。

**Lab7：**

16x16LED阵列的测试lab，开启模拟视觉残留效应后显示如下。

