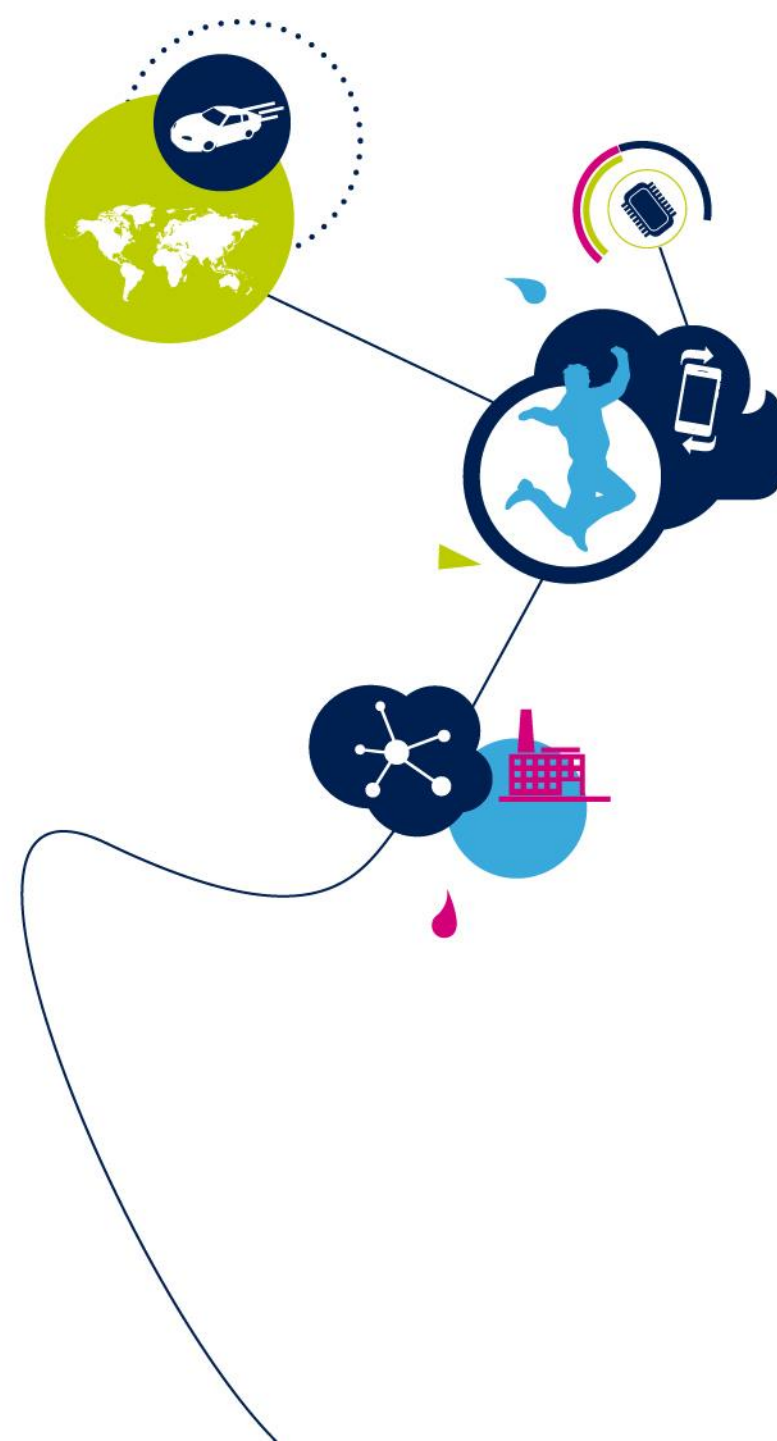


动手实验3

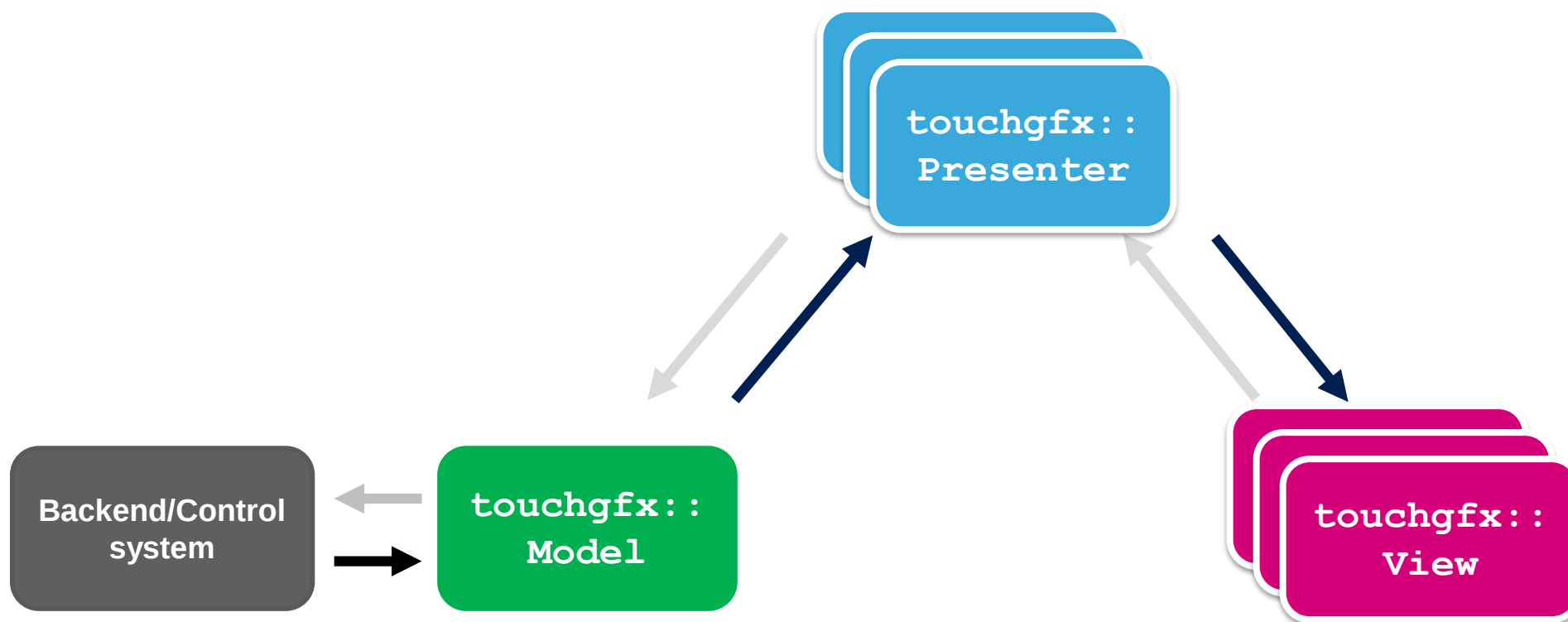
使用TouchGFX MVP 来实现UI和硬件的双向交互

-基于FreeRTOS



实验目标

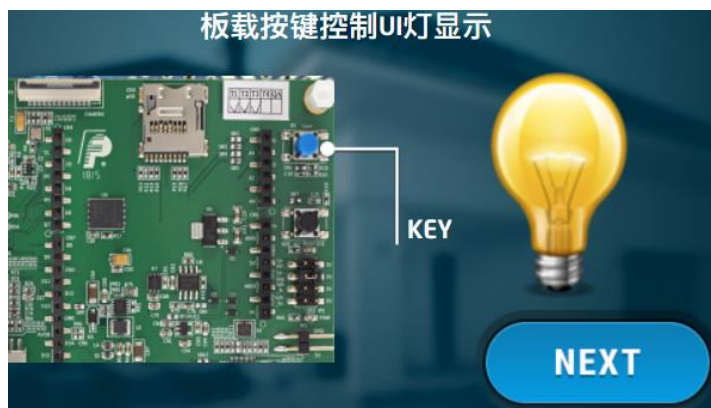
通过动手实验理解TouchGFX MVP+RTOS使用方法



功能

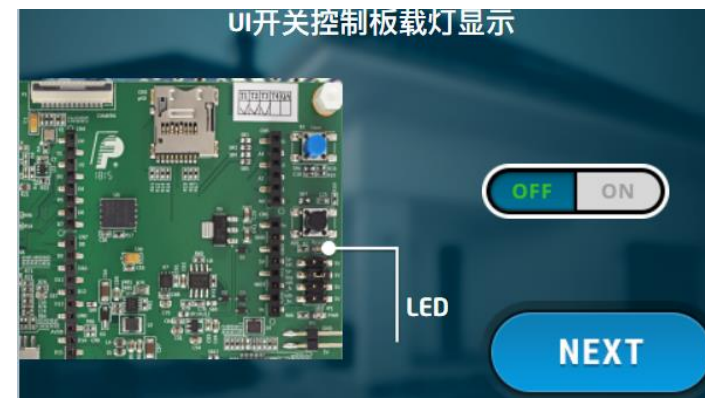
- 界面1

- 板载按键控制UI灯显示
- 板上KEY GPIO为PI1



- 界面2

- UI开关控制板载灯显示
- 板上LED GPIO为PI1



- 图片和字库资源

- HandOn3_res.7z



HandsOn3_res.7z

- 宏定义

- #define UBP_LIGHT_ON 0x01
- #define UBP_LIGHT_OFF 0x02
- #define UBP_KEY 0x03

-可使用TouchGFX Designer的模板

- Application Template
 - STM32F746G Discovery Kit
- UI Template
 - Blank UI

TouchGFX DESIGNER

MY APPLICATIONS ONLINE APPLICATIONS

Create New Application

APPLICATION NAME

MyCustomizedWidgetDemo

APPLICATION DIRECTORY

C:\TouchGFXProjects

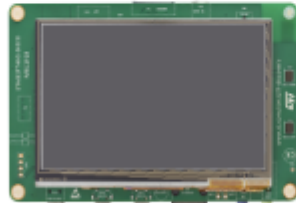
APPLICATION TEMPLATE

STM32F746G Discovery Kit
by STMicroelectronics (v2.0.0)

Board Name
STM32F746G-...

Operating System
FreeRTOS

Resolution
480 x 272



UI TEMPLATE

Blank UI
by STMicroelectronics (v1.1.0)



COLOR DEPTH

24 bit

WIDTH

480

HEIGHT

272

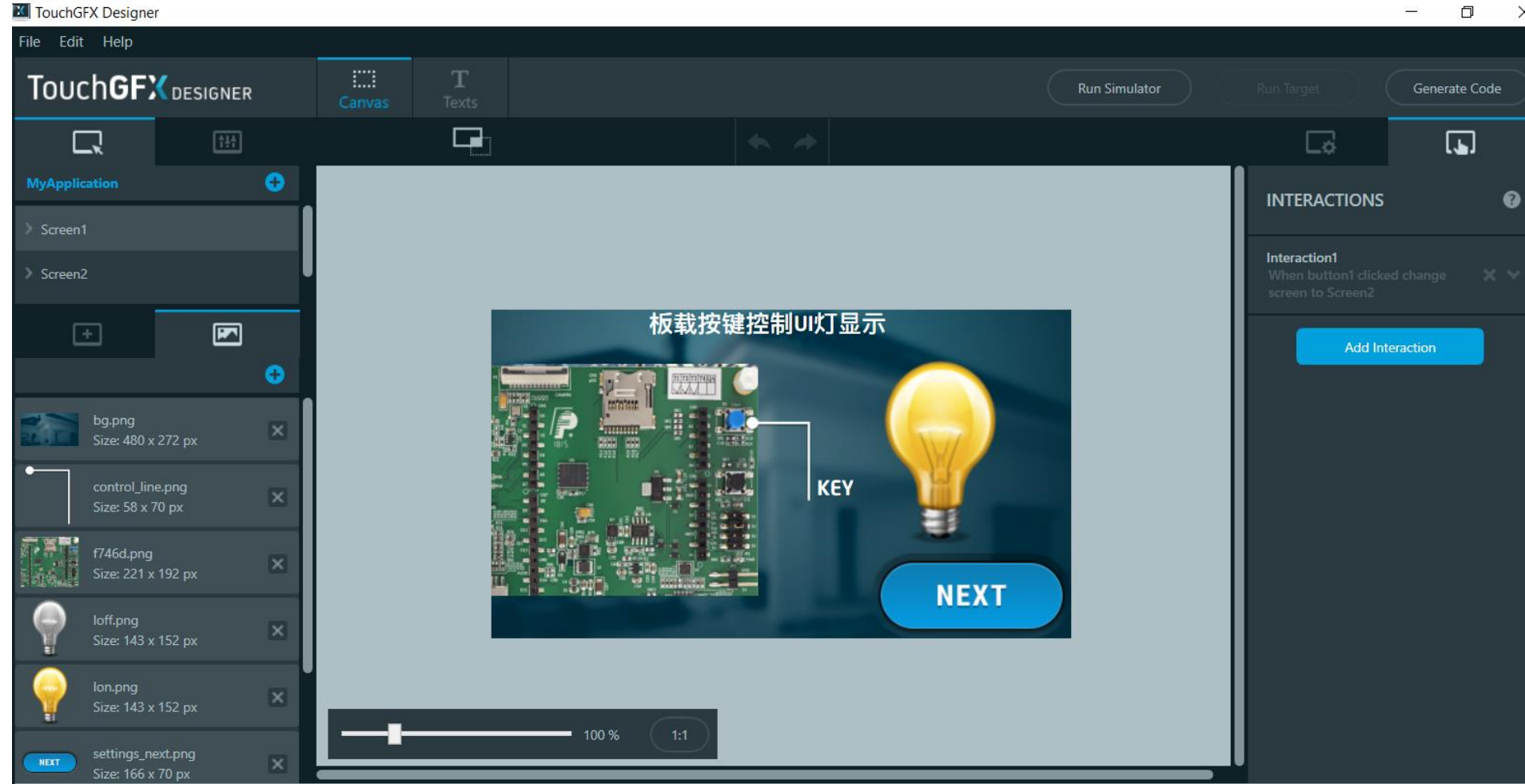
CREATE

TouchGFX Designer开发UI

5

TouchGFX Designer

- 界面1

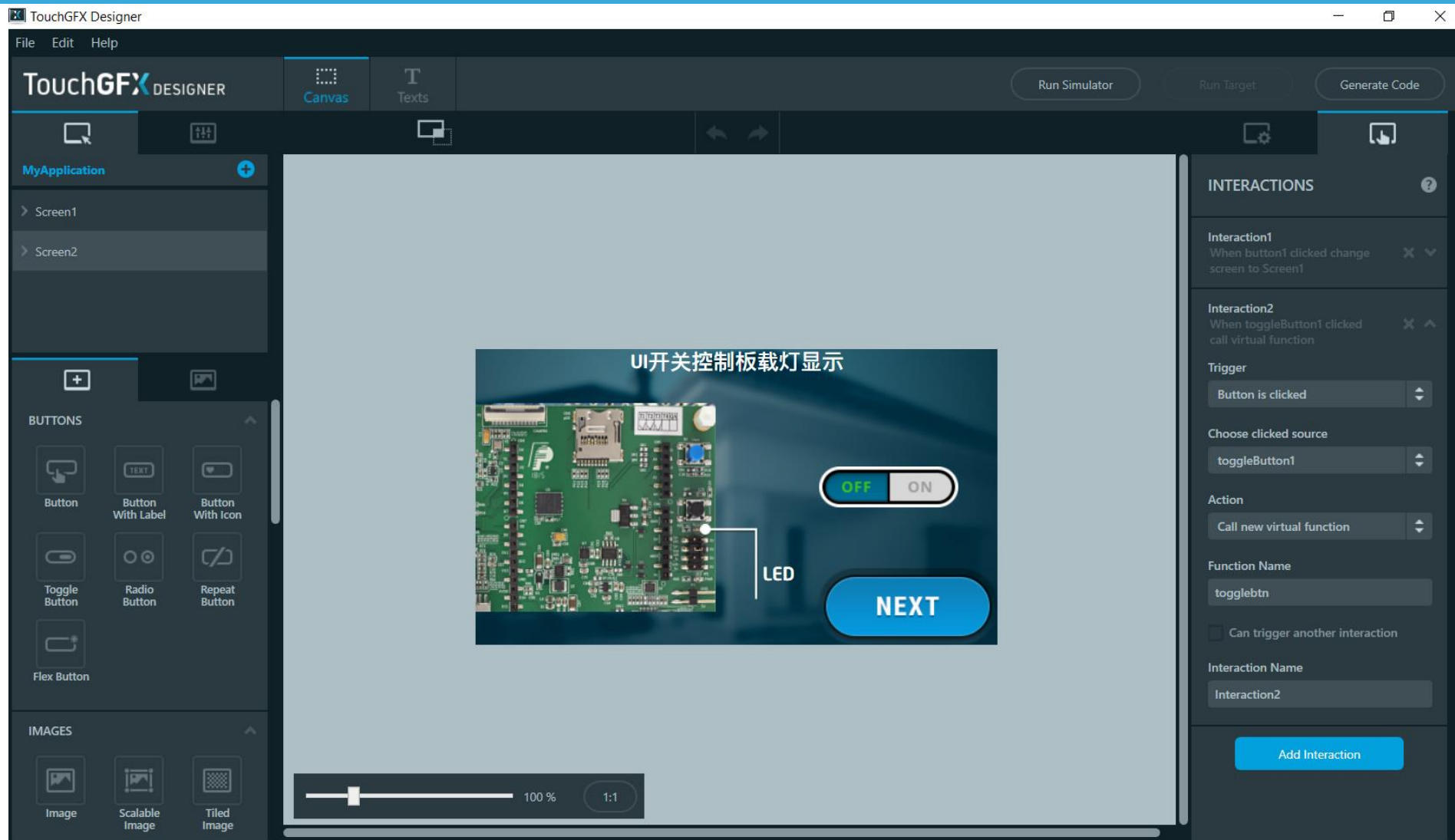


TouchGFX Designer开发UI

6

TouchGFX Designer

- 界面2

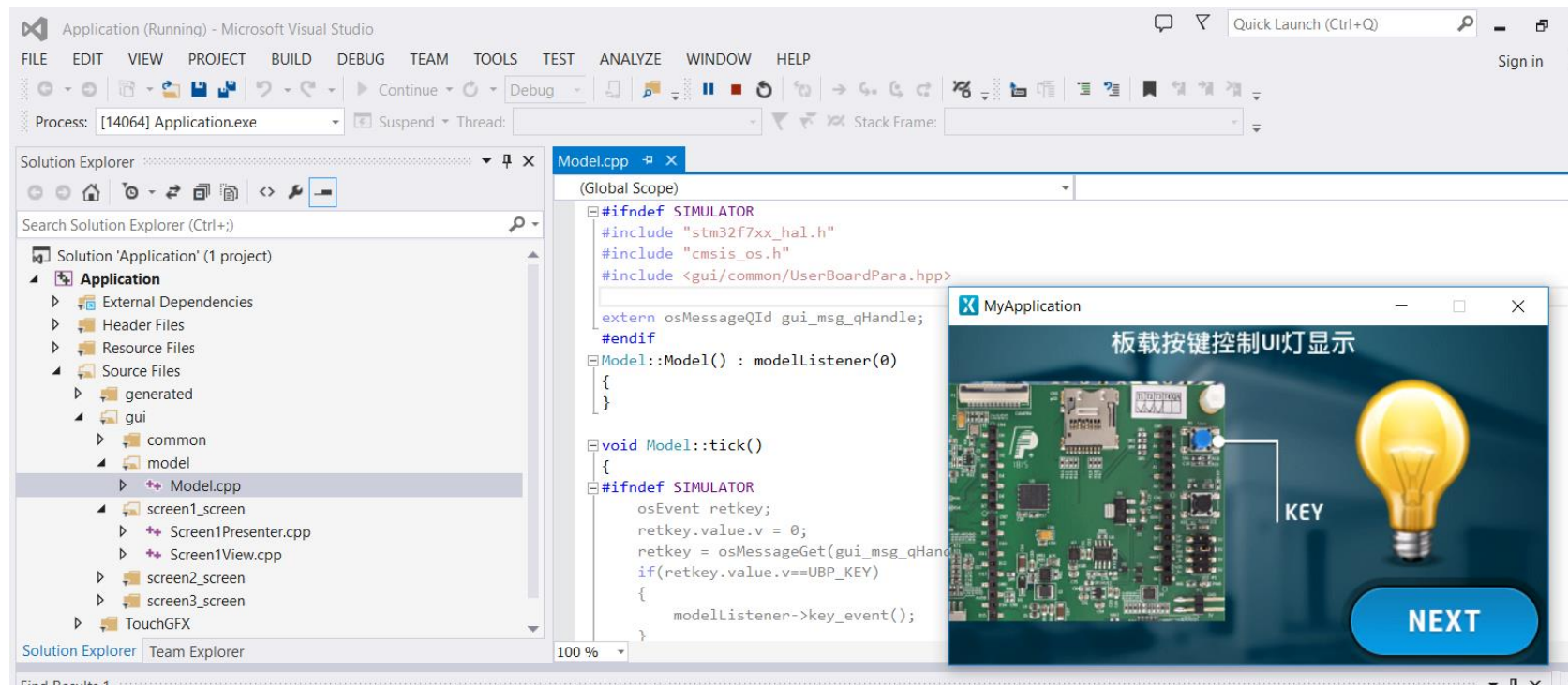


在Visual Studio中仿真UI

7

- 页面效果和界面切换可以在VS中进行仿真，然后再在IAR/KEIL中加入控制硬件的程序
- 在IAR中控制MCU的程序可以和UI部分在Model层中用宏定义分离

```
#ifndef SIMULATOR  
//翻转GPIO  
#endif
```



界面1 → Model & Presenter

ModelListener.hpp

```
class ModelListener
{
public:
    ModelListener() : model(0) {}

    virtual ~ModelListener() {}
    virtual void key_event() {}
    void bind(Model* m)
    {
        model = m;
    }
protected:
    Model* model;
};
```

Screen1Presenter.hpp

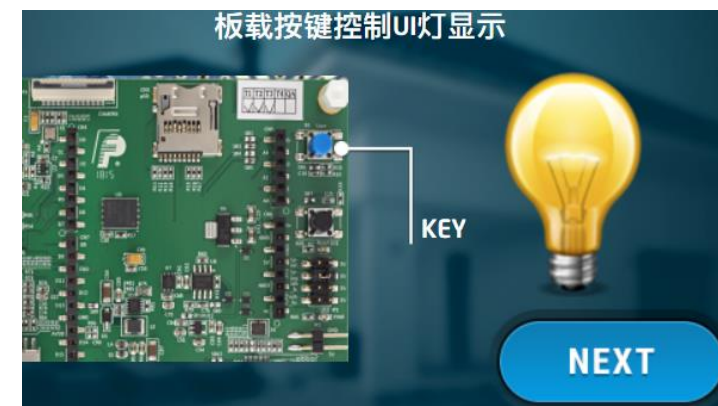
```
class Screen1Presenter : public Presenter,
public ModelListener
{
public:
    void key_event();
};
```

Model.cpp

```
void Model::tick()
{
#ifdef SIMULATOR
    osEvent retkey;
    retkey.value.v = 0;
    retkey = osMessageGet(gui_msg_qHandle,1);
    if(retkey.value.v==UBP_KEY)
    {
        modelListener->key_event();
    }
#endif
}
```

Screen1Presenter.cpp

```
void Screen1Presenter::key_event()
{
    view.handleKey(UBP_KEY);
}
```



界面1 → View

Screen1View.hpp

```
class Screen1View : public Screen1ViewBase
{
public:
    Screen1View();
    virtual ~Screen1View() {}
    virtual void handleKey(int key);
    virtual void togglelight();
protected:
    bool light_status_off = true;
};
```

Screen1View.cpp

```
void Screen1View::handleKey(int key)
{
    if(key==UBP_KEY)
    {
        togglelight();
    }
}
void Screen1View::togglelight()
{
    if(light_status_off==true)
    {
        lon1.setBitmap(Bitmap(BITMAP_LOFF_ID));
        light_status_off=false;
    }else{
        lon1.setBitmap(Bitmap(BITMAP_LON_ID));
        light_status_off=true;
    }
    lon1.invalidate();
}
```



界面2 → Model & Presenter

Model.hpp

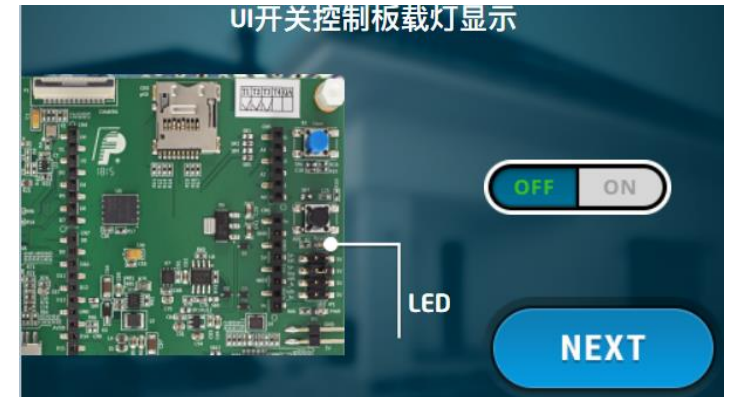
```
class Model
{
    virtual void set_light_onoff(uint8_t data);
};
```

Model.cpp

```
void Model::set_light_onoff(uint8_t data)
{
    #ifndef SIMULATOR
        osMessagePut(gui_msg_qHandle,data,200);
    #endif
}
```

Screen2Presenter.hpp

```
class Screen2Presenter : public Presenter, public ModelListener
{
    Screen2Presenter(Screen2View& v);
    void set_light_off(uint8_t data)
    {
        model->set_light_onoff(data);
    }
};
```



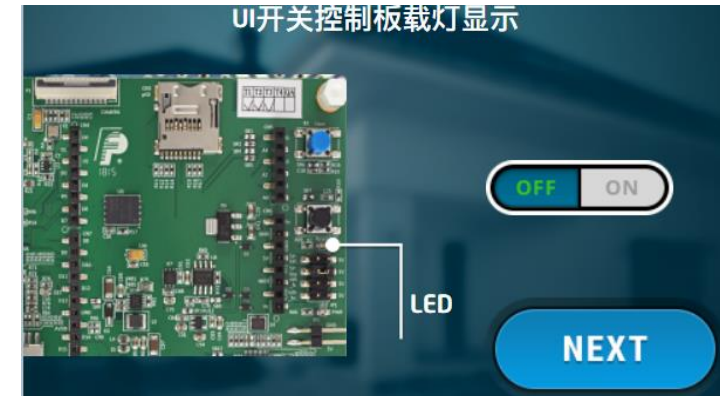
界面2 → View

Screen2View.hpp

```
class Screen2View : public Screen2ViewBase
{
public:
    Screen2View();
    virtual ~Screen2View() {}
    virtual void setupScreen();
    virtual void tearDownScreen();
    virtual void togglebtn();
protected:
    bool togglebtn_status_off ;
};
```

Screen2View.cpp

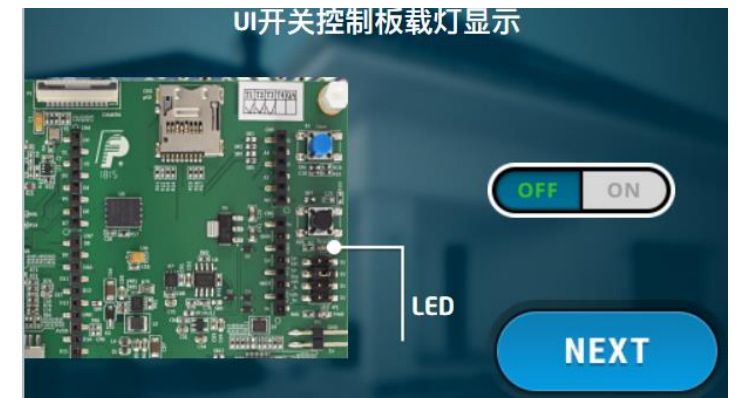
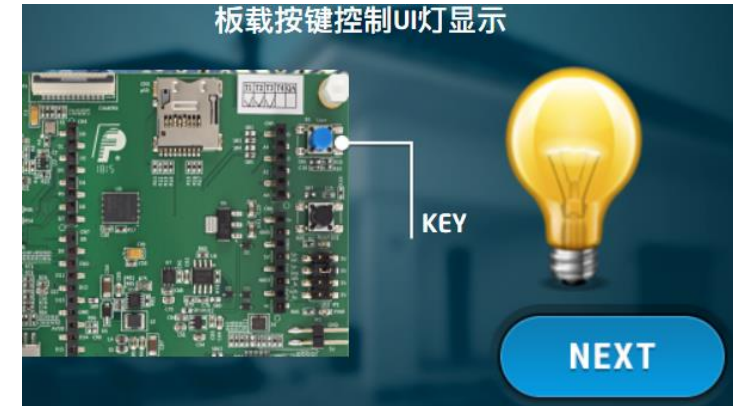
```
void Screen2View::togglebtn()
{
    if(toggleButton1.getState()==true)
    {
        presenter->set_light_off(UBP_LIGHT_OFF);
    }else{
        presenter->set_light_off(UBP_LIGHT_ON);
    }
}
```



主程序

Main.cpp

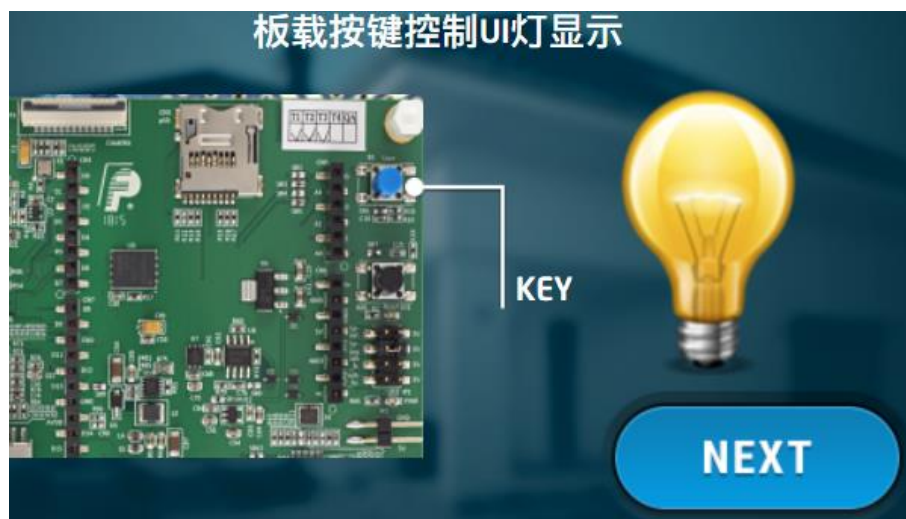
```
void Startprototask(void const * argument)
{
    /* USER CODE BEGIN Startprototask */
    /* Infinite loop */
    osEvent retkey;
    for(;;)
    {
        if(HAL_GPIO_ReadPin(GPIOI, GPIO_PIN_11 )==GPIO_PIN_SET)
        {
            osMessagePut(gui_msg_qHandle,UBP_KEY,200);
            osDelay(200);
        }
        retkey.value.v = 0;
        retkey = osMessageGet(gui_msg_qHandle,10);
        if(retkey.value.v==UBP_LIGHT_OFF||retkey.value.v==UBP_LIGHT_ON)
        {
            HAL_GPIO_TogglePin(GPIOI,GPIO_PIN_1);
        }
    }
    /* USER CODE END Startprototask */
}
```



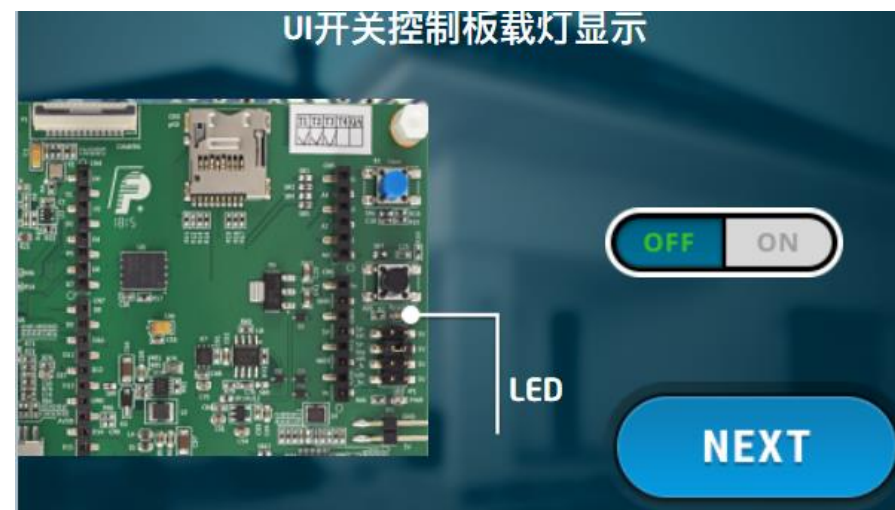
期望实现结果

实验完成期望的运行结果如下所示：

- 1.在Visual Studio 上仿真
- 2.在STM32F746Discovery板上运行



相互切换





Thank You !