

类别	内容
关键词	显示文本
摘要	



## 修订历史

版本	日期	原因	编制	审查
V1.0	2019/02/27	创建文档	林青田	



## 销售与服务

### 广州大彩光电科技有限公司

电话：020-82186683

传真：020-82187676

Email: [hmi@gz-dc.com](mailto:hmi@gz-dc.com)（公共服务）

网站： [www.gz-dc.com](http://www.gz-dc.com)

地址：广州高新技术产业开发区玉树工业园富康西街 8 号 C 栋 303 房

官网零售淘宝店： [www.gz-dc.taobao.com](http://www.gz-dc.taobao.com)



## 目录

1. 适合范围.....	1
2. 开发环境版本.....	2
3. 概述.....	3
4. 参考资料.....	4
5. 教程实现.....	5
5.1 准备工程素材.....	5
5.1.1 硬件平台.....	5
5.1.2 UI素材准备.....	5
5.1.3 LUA编辑器.....	6
5.2 API函数说明.....	7
5.3 实现功能.....	8
5.3.1 显示文本.....	8
5.4 下载工程.....	10
5.4.1 下载.....	11
6. 完整程序清单.....	12
7. 免责声明.....	14

## 1. 适合范围

该文档适合所有大彩物联型系列。

## 2. 开发环境版本

1. VisualTFT 软件版本: V3.0.0.944 及以上的版本。

版本查看:

- a) 打开 VisualTFT 软件启动页面如图 2-1 软件版本, 右上角会显示的软件版本号;



图 2-1 软件版本

- b) 打开 VisualTFT, 在软件右下角可以查看软件版本图 2-2 软件版本, 最新版本可登录<http://www.gz-dc.com/>进行下载。

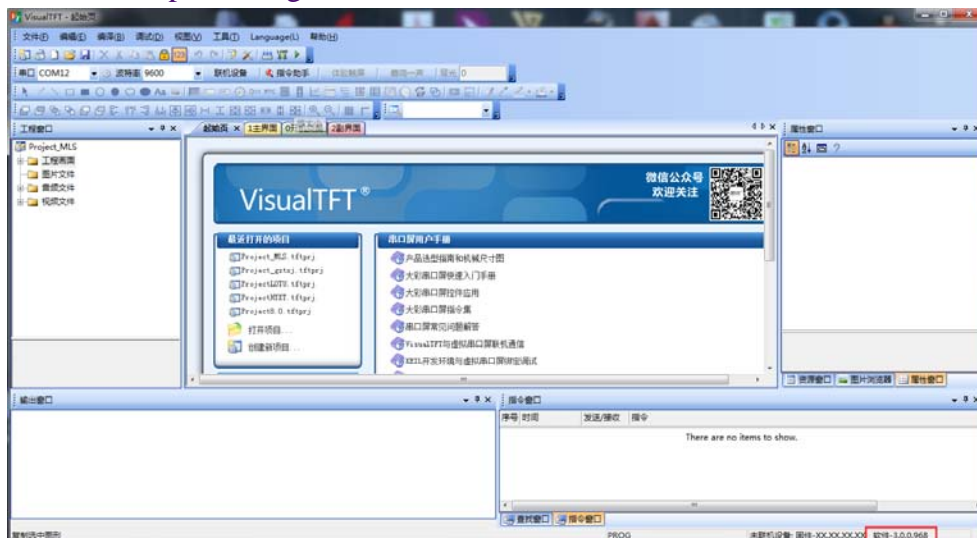


图 2-2 软件版本

2. 串口屏硬件版本: V3.0.301.0 及以上的版本。

版本查看:

- a) 查看屏幕背面版本号贴纸;
- b) VisualTFT 与屏幕联机成功后, 右下角显示的版本号。
3. LUA 语言版本 V5.5。



### 3. 概述

物联型串口屏通过 LUA 脚本配合工程可以完成大部分的内部逻辑处理，可以做到让 MCU 只参与数据处理，不参与屏的逻辑处理。

本文将介绍如何使用 LUA 脚本 API 函数在大彩串口屏中显示字符串。

## 4. 参考资料

1. 《大彩组态串口屏 LUA 脚本 API》 可通过以下链接下载物联型开发包获取:  
<http://www.gz-dc.com/index.php?s=/List/index/cid/19.html>
2. 《LUA 基础学习》 可通过以下链接下载物联型开发包获取:  
<http://www.gz-dc.com/index.php?s=/List/index/cid/19.html>
3. LUA脚本初学者可以通过下面链接进行学习。  
<http://www.runoob.com/lua/lua-arrays.html>



## 5. 教程实现

本章节主要通过 LUA 教程 demo 讲述如何基础运算和字符处理函数的使用以及编写程序的注意事项。本文将分为以下是 3 个阶段讲述教程 DEMO 是如何实现的：

1. 准备工程素材，
2. 实现功能，
3. 下载工程。

### 5.1 准备工程素材

在实现教程 DEMO 前需要作以下三个准备：

1. 硬件平台，
2. UI 素材，
3. LUA 编辑器。

#### 5.1.1 硬件平台

该例程使用大彩物联型 7 寸串口屏 DC80480W070 为验证开发平台。如图 5-1 所示

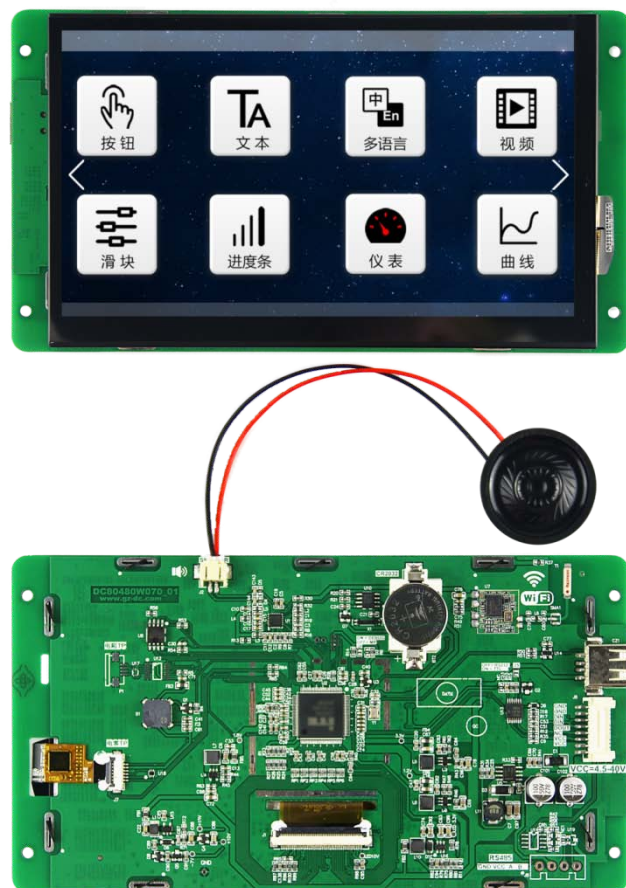


图 5-1 物联型 7 寸串口屏

其他尺寸的物联型串口屏均可借鉴此教程。

#### 5.1.2 UI 素材准备

准备好相应的美工 UI。如图 5-2 所示



图 5-2 素材准备

### 5.1.3 LUA 编辑器

上位机 VisualTFT 内部已集成了 LUA 开发编译环境，点击菜单栏工具，选择 LUA 编辑器，如图 5-3 所示。

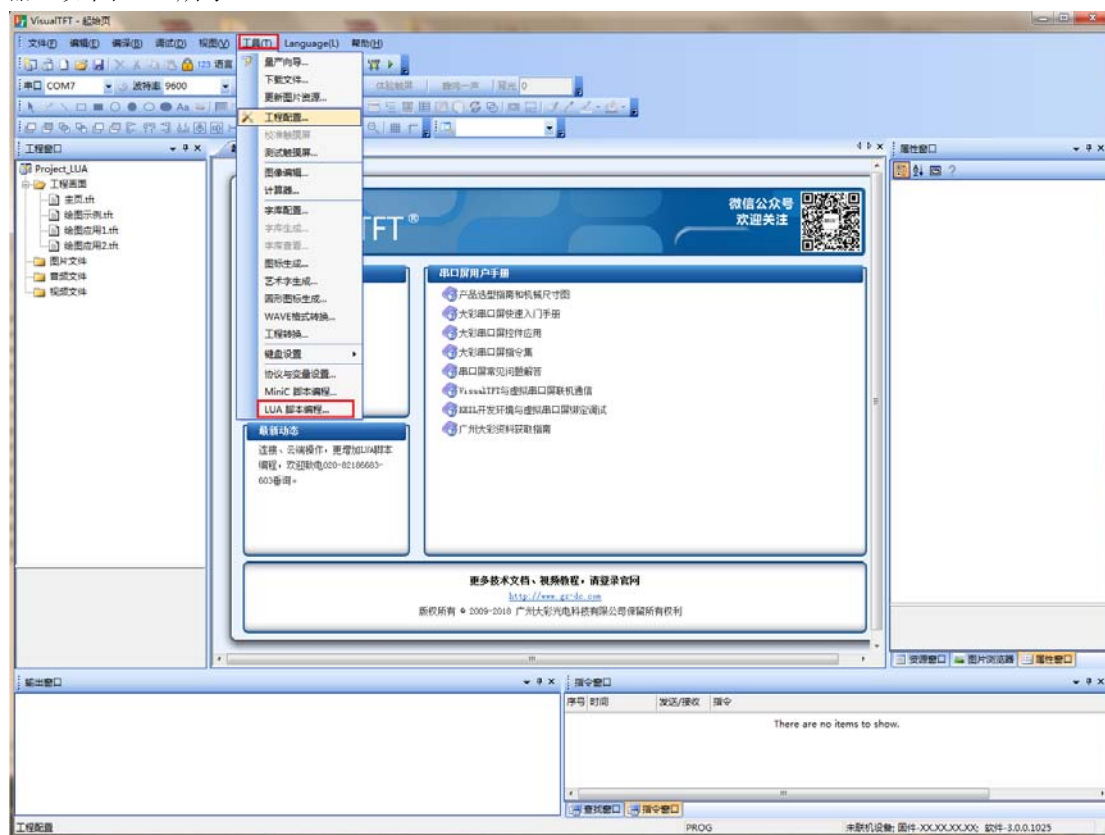


图 5-3 打开 LUA 编辑器

打开编辑器后，Visual TFT 画面如图 5-4 所示；

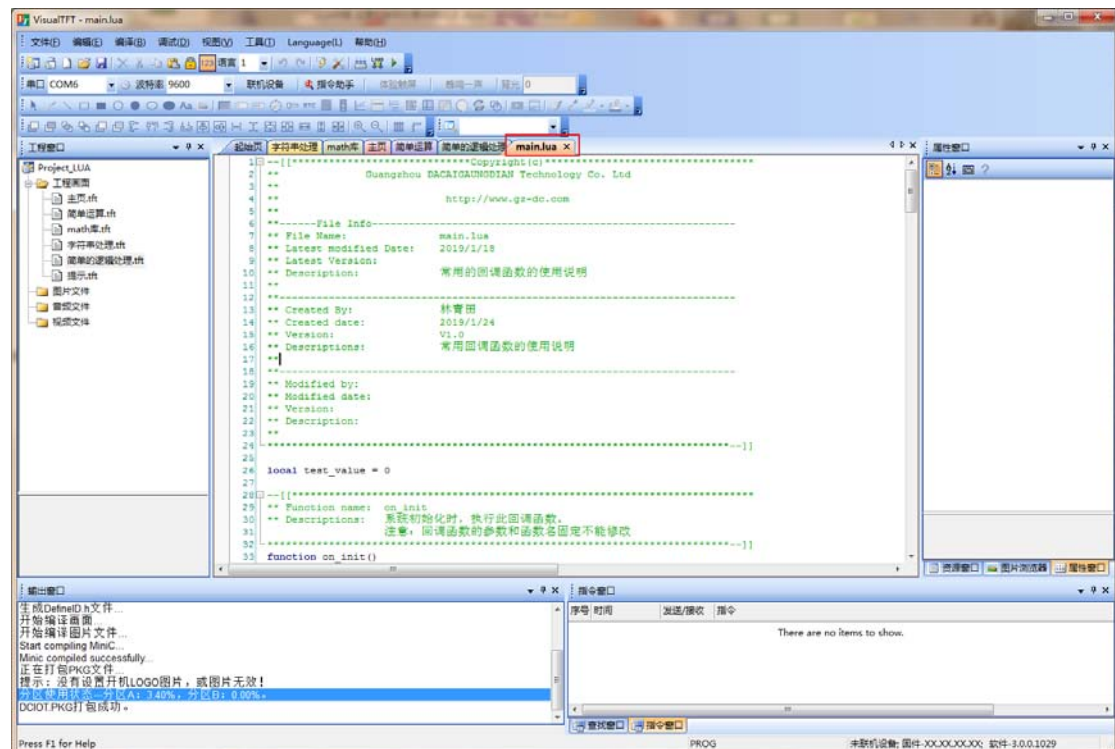


图 5-4 LUA 编辑器

## 5.2 API 函数说明

大彩科技针对 LUA 脚本提供了丰富的 API 接口函数，具体函数可以查阅文档《物联型 LUA 脚本 API 》，如图 5-5 所示。

3. API 接口函数 .....	8
3.1 控件属性类.....	8
3.1.1 change_screen(screen) .....	8
3.1.2 set_value(screen,control,value) .....	8
3.1.3 get_value(screen,control) .....	8
3.1.4 set_visiable(screen,control,visiable) .....	8
3.1.5 set_enable(screen,control,enable) .....	8
3.1.6 set_fore_color(screen,control,color) .....	8
3.1.7 set_back_color(screen,control,color) .....	8
3.1.8 set_text(screen,control,text) .....	8
3.1.9 get_text(screen,control) .....	8
3.2 常用回调函数.....	8
3.2.1 on_init() .....	8
3.2.2 on_systick() .....	8
3.2.3 on_control_notify(screen,control,value) .....	8
3.2.4 on_screen_change(screen).....	9
3.2.5 on_press(state,x,y).....	9
3.2.6 on_usb_inserted(driver) .....	9
3.2.7 on_usb_removed().....	9
3.3 绘图函数.....	9
3.3.1 on_draw(screen).....	9
3.3.2 redraw() .....	9
3.3.3 set_pen_color(color).....	9
3.3.4 draw_line(x0,y0,x1,y1,width) .....	9
3.3.5 draw_rect(x0,y0,x1,y1,fill).....	9
3.3.6 draw_circle(x,y,r,fill).....	9
3.3.7 draw_ellipse(x0,y0,x1,y1,fill) .....	10
3.3.8 draw_image(image_id,frame_id,dstx,dsty,width,height,srcx,srcy) .....	10
3.3.9 draw_image_file(filename,dstx,dsty,width,height,srcx,srcy) .....	10
3.3.10 load_surface (filename).....	10
3.3.11 destroy_surface (surface) .....	10
3.3.12 draw_surface (surface,dstx,dsty,width,height,srcx,srcy) .....	10
3.3.13 draw_text(text,x,y,w,h,font,color,align).....	11
3.4 寄存器访问.....	11

图 5-5 API 函数文档

本教程文档中所涉及到的部分的相关接口函数

1. 函数 set\_text (screen, control, str)  
注释：指定某个画面中的文本控件或者二维码控件显示指定内容：
2. 函数 on\_screen\_change(screen)  
注释：当画面切换至目标画面 ID 时，执行此回调函数  
参数：screen 表示目标画面 ID

## 5.3 实现功能

教程中每个功能的实现步骤如下：

1. 配置工程属性，
2. 编写程序以及编译，
3. 运行程序。

### 5.3.1 显示文本

1. 在教程 DEMO 的画面中，实现点击按钮在文本框中显示“HELLO LUA”，运行虚拟屏后工程画面如图 5-6 所示

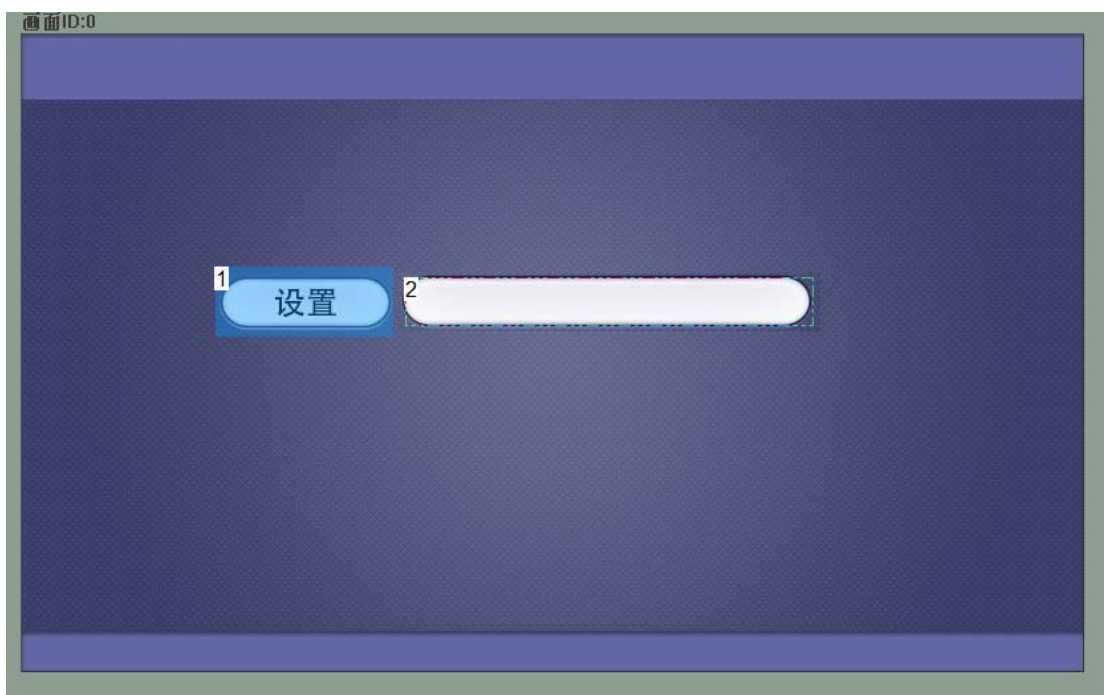


图 5-6 工程画面

2. LUA 程序解释: 按下按钮后在文本框中显示“HELLO LUA”, 程序如程序清单 1 所示;

程序清单 1 获取绘图类型

```
--[[*****]]
** Function name:  on_control_notify
** Descriptions:   用户通过触摸修改控件后, 执行此回调函数。
                   点击按钮控件, 修改文本控件、修改滑动条都会触发此事件。
                   注意: 回调函数的参数和函数名固定不能修改

** Input value :   screen 画面 ID
                   control 控件 ID
                   value   控件值(包括文本控件输入的值)

*****--]]

function on_control_notify(screen,control,value)
    if screen == 0 and control == 1 and value == 1 then
        set_text(0,2,"HELLO LUA")
    end
end
```

编写完功能模块后将工程和脚本一起进行编译, 编译成功后可以使用软件中的虚拟屏查看程序是否实现功能, 点击工具栏中编译工程按钮, 可将工程和脚本的一起编译, 操作如图 5-7 所示



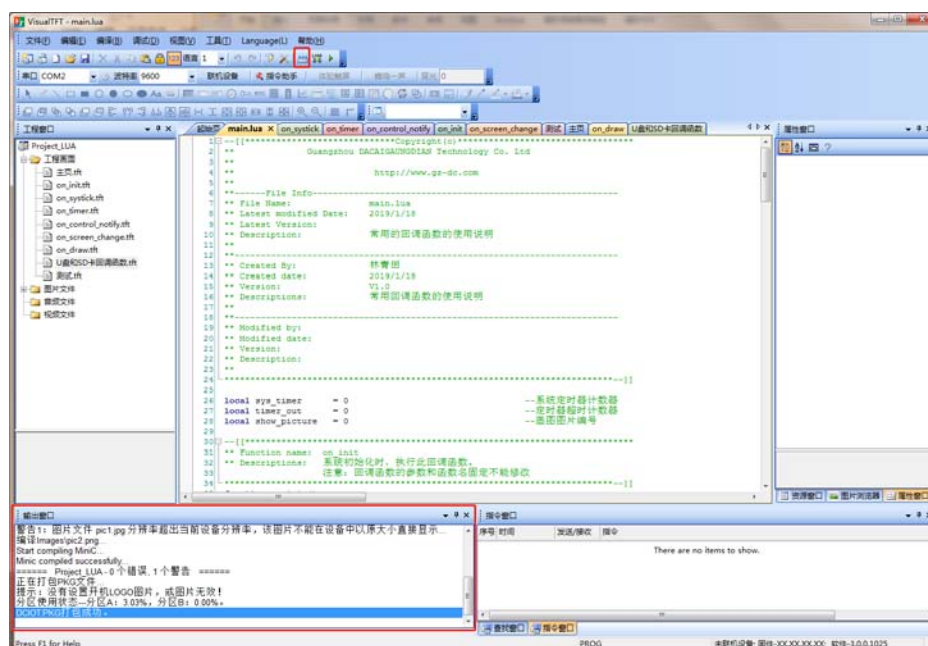


图 5-7 编译工程

注意：目前在软件 Visual TFT 集成的 LUA 脚本编译器无法进行语法检测，所以编写 LUA 程序是尽量分模块编写，已达到节省调试的时间。

3. 点击工具栏中的运行虚拟串口屏，软件先会将当前工程进行编译，编译成功后工程会在虚拟屏上运行起来，如图 5-8 所示



图 5-8 基本绘图示例

## 5.4 下载工程

在我司的上层软件 Visual TFT 中集成了 LUA 程序的编译器，可以实现在编译工程的同时将 LUA 脚本程序一起编译，并且将编译后的图片和程序集合在一个名为 DCIOT.PKG 的

文件中。编译后只需要把 DCIOT.PKG 文件拷贝到 U 盘中，接上串口屏并重新上电即可将图片和程序下载到屏中。

### 5.4.1 下载

工程编译成功后在输出窗口会提示编译成功，如所示。编译成功后打开工程目录，找到 output 文件夹，将文件夹中的 DCIOT.PKG 文件拷贝到 U 盘中，如图 5-9 和图 5-10 所示；接上串口屏重新上电，等到提示烧录工程成功后，拔掉 U 盘重新上电即可。

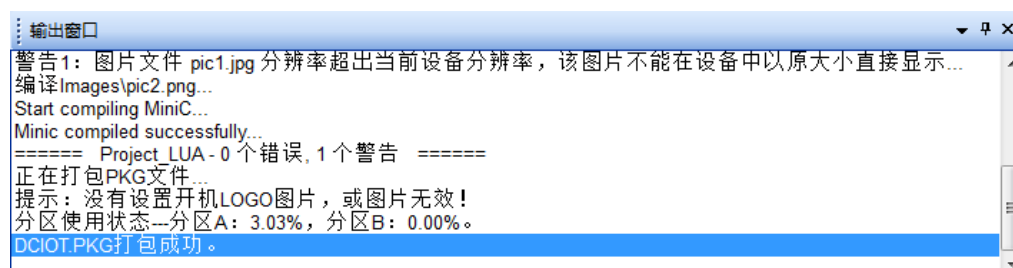


图 5-9 编译成功

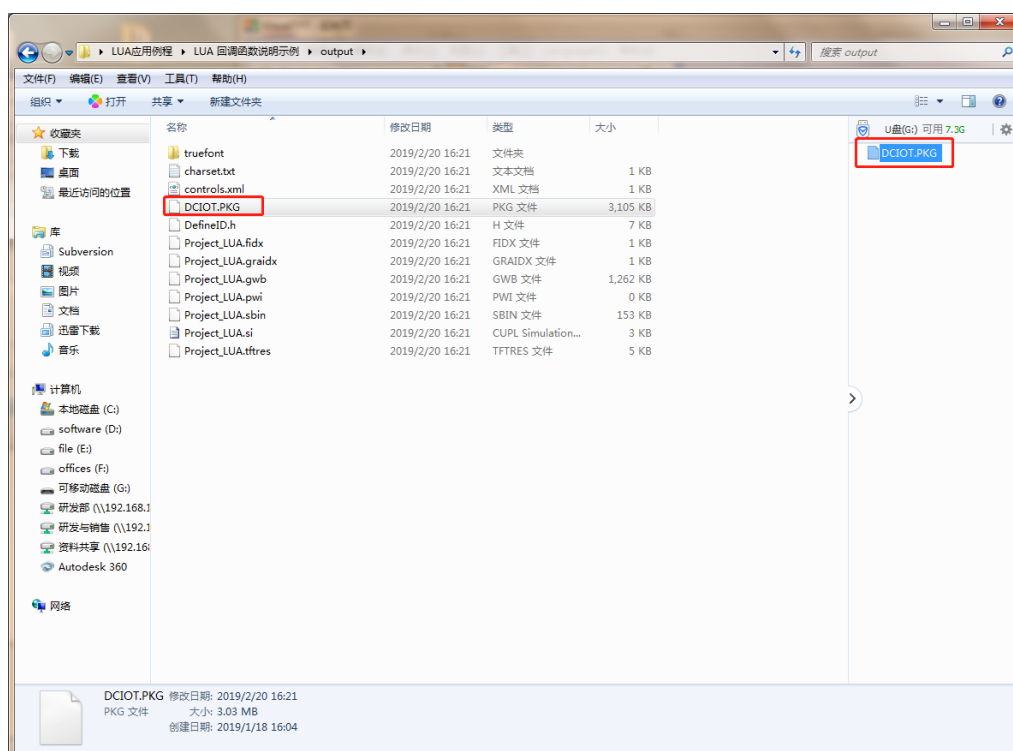


图 5-10 拷贝

## 6. 完整程序清单

LUA 教程-显示文本的完整程序，如程序清单 2 所示，查看程序时请结合教程 demo 理解：

程序清单 2 完整程序

```
[[*****Copyright(c)***** **
**
**          Guangzhou DaCai
**
**          http://www.gz-dc.com
**
**-----File Info-----
** File Name:          main.lua
** Latest modified Date: 2019/2/22
** Latest Version:
** Description:        显示文字
**
**-----
** Created By:          林青田
** Created date:        2019/2/22
** Version:             V1.0
** Descriptions:        显示文字
**
**-----
** Modified by:
** Modified date:
** Version:
** Description:
**
*****--]]

--[*****
** Function name:  on_control_notify
** Descriptions:  用户通过触摸修改控件后，执行此回调函数。
                  点击按钮控件，修改文本控件、修改滑动条都会触发此事件。
                  注意：回调函数的参数和函数名固定不能修改
** Input value :   screen 画面 ID
                  control 控件 ID
                  value   控件值(包括文本控件输入的值)
*****--]]

function on_control_notify(screen,control,value)
if screen == 0 and control == 1 and value == 1 then
    set_text(0,2,"HELLO LUA")
end
```





```
end
```

```
--[[*****  
END FILE  
*****-]]
```

## 7. 免责声明

广州大彩光电科技有限公司所提供的所有服务内容旨在协助客户加速产品的研发进度，在服务过程中或者其他渠道所提供的任何例程程序、技术文档、CAD 图等资料和信息，都仅供参考，客户有权不使用或自行参考修改，本公司不提供任何的完整性、可靠性等保证，若是客户使用过程中因任何原因造成的特别的、偶然的或间接的损失，本公司不承担任何责任。