技术笔记

Technical Note

# 教程 7 - 串口屏动画控件的应用 V1.0

工程技术笔记

| 类别 | 内容 |
| --- | --- |
| 关键词 | 开机动画、制作 ICON 文件、播放 ICON 文件 |
| 摘要 | 动画控件的使用方法 |

广州大彩光电科技有限公司

## 修订历史

| 版本 | 日期 | 原因 | 编制 | 审查 |
|---|---|---|---|---|
| V1.0 | 2019/07/31 | 创建文档 | 林绍佳 | 刘启鑫 |

# 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（公共服务）

网站：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

# 目录

1. 适合范围.....1
2. 开发环境版本.....2
3. 参考文档.....3
4. 概述.....4
5. 文本控件属性.....5
5.1 属性介绍.....5
6. 教程实现.....7
6.1 准备UI素材.....7
6.2 开发工具.....7
6.3 软件版本.....8
6.4 功能实现.....8
6.4.1 建立工程.....8
6.4.2 开机动画.....9
6.4.3 ICON文件播放.....11
6.5 下载工程.....14
6.6 联机调试.....14
6.6.1 硬件连线.....14
6.6.2 联机设备.....15
6.6.3 指令控制.....15
6.6.4 串口屏效果.....15
7. 部分程序代码.....16
8. 免责声明.....19

# 1. 适合范围

文档适合基本型、经济型、商业型、F 型、物联型、86 盒、485 系列等所有大彩协议串口屏产品。

# 2. 开发环境版本

1. VisualTFT 软件版本：V3.0.0.1037 及以上的版本，版本查看方式：
   (1) 打开 VisualTFT 软件启动页面如图 2-1 所示，右上角显示的软件版本号；




图 2-1 软件版本

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

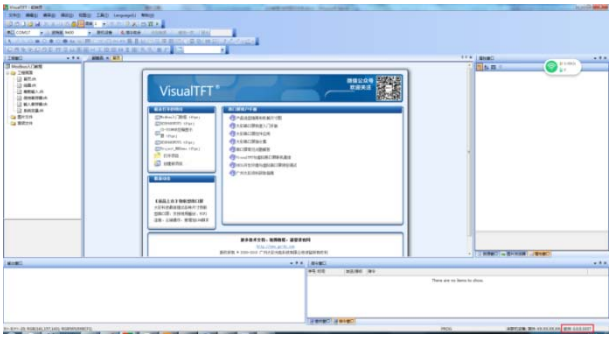


图 2-2 软件版本

2. 硬件固件版本查看方式：
   (1) 查看屏幕背面版本号贴纸，如图 2-3 所示。




图 2-3 背面贴纸

   (2) 画面中创建一个文本控件，属性【输入方式】选择“系统变量”，【系统变量】选择“固件版本号”，编译工程下载到屏中查看。

# 3. 参考文档

本例程中涉及所需的文档如下：

1. 《大彩串口屏指令集 V5.0 .pdf》 ：用于查看指令解析；

# 4. 概述

GIF 动画、ICON 图标文件可以通过动画控件来显示。用户可以设置动画控件自动播放，也可以设置触摸。

本例程中介绍动画控件常见的应用，结合工程中的画面介绍每一个应用的配置，常见的应用如下所示。

1. 开机动画：开机后，动画自动播放，并在播放完成后执行指定的动作。
2. ICON 文件播放：使用 VisualTFT 软件制作 ICON 文件，ICON 图标循环播放。

# 5. 动画控件属性

## 5.1 属性介绍

本章节重点介绍动画控件的属性，属性窗口如图 5-1 所示。




图 5-1 动画控件

1. 初始化显示：进入该画面时控件是否初始化显示。
   (1) 选择“否”，控件初始化不显示。关闭之后用户如果想要把动画显示出来,可以用指令助手的显示隐藏控件发送指令进行初始化显示。
   (2) 选择“是”，开启初始化显示，控件初始化显示。
2. 动画文件：选择播放文件。
3. 透明处理：是否显示图标的透明色。
   (1) 使用透明色：图标中的透明色直接透明显示。
   (2) 与背景混合：图标中的透明色显示当前画面的相同坐标的背景色或背景图片。
4. 自动播放：进入该画面时动画是否自动播放。
   (1) 否：不自动播放。
   (2) 是：自动播放，如图 5-2 所示。
      a) 间隔时间：自动播放间隔时间（单位为毫秒）
      b) 循环次数：循环播放次数（0 表示不断循环播放）
      c) 播放结束时：播放结束时的处理（不处理、发送通知、切换到下一画面）




图 5-2 自动播放

5. 触摸设置：手指在控件上触摸或滑动时，是否通知或换帧。
   (1) 禁止触摸：不能触摸。
   (2) 允许触摸：触摸控件时发送通知。
   (3) 水平/垂直滑动时换帧：水平/垂直滑动时，切换帧并发送通知。
      a) 步长（像素）：滑动切换一帧的单位，切换帧数=滑动距离/步长，如图 5-3 所示。




图 5-3 滑动换帧

# 6. 教程实现

该例程使用大彩 F 型 7 寸电阻串口屏 DC80480F070 为验证开发平台。如图 6-1 所示。

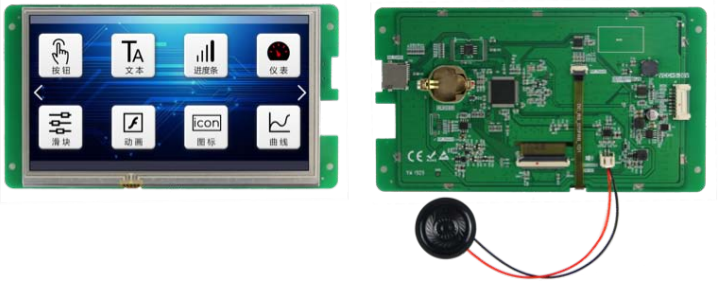


图 6-1　F 型 7 寸电阻屏

## 6.1　准备 UI 素材

准备好相应的美工 UI。如图 6-2 所示。




图 6-2　UI 准备

## 6.2　开发工具

准备以下开发工具，硬件开发工具如表 6-1 所示，软件开发工具如表 6-2 所示。

表 6-1　硬件开发工具

| 硬件开发工具 | 功能 |
|---|---|
| 电源适配器 5V 1A | 用于给串口屏供电 |
| 调试转接板 | 用于与 PC 调试（仅开发时用到） |
| 8P-PH2.0 线缆 | 用于串口屏和用户主机连接 |
| SD 卡 | 用于脱机量产下载工程 |
| RS-232 串口线 | 用于 PC 连接串口屏调试 |

表 6-2　软件开发工具

| 软件开发工具 | 功能 |
|---|---|
| Configure Virtual Serial Port Driver | 用于创建虚拟一对串口 |
| VisualTFT | 用于工程配置 |

调试连接示意图，如图 6-3 所示。




图 6-3　调试连接图

## 6.3　软件版本

本例程所实现的功能 VisualTFT 版本和固件版本如下表 6-3 所示。

表 6-3　版本清单

| 版本 | 版本号 |
|---|---|
| VisualTFT 版本 | V3.0.0.1037 |
| 基本型（经济型、商业型）固件 | V2.22.912.259 及以上的版本支持 |
| F 型固件 | V4.1.66.0 及以上的版本支持 |
| 物联型固件 | V3.0.301.0 及以上的版本支持 |

## 6.4　功能实现

### 6.4.1 建立工程

根据串口屏的实际型号，在 VisualTFT 软件菜单栏选择【文件】→“新建工程”，选择对应的系列和型号，如图 6-4 所示。

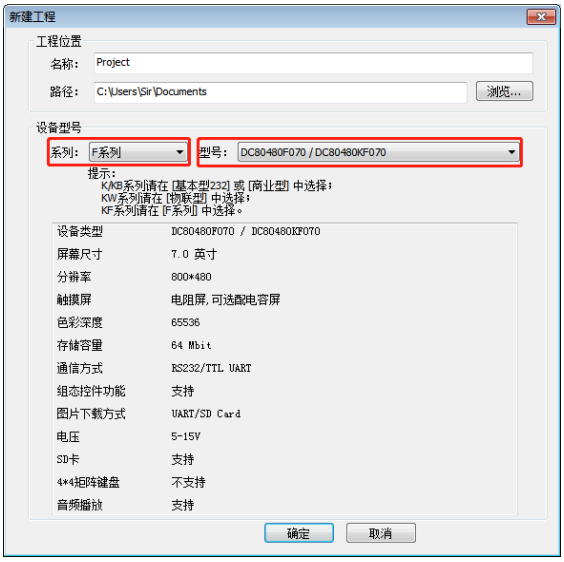



图 6-4　新建工程

### 6.4.2 开机动画

【开机动画】画面，介绍开机动画设置，可以设置开机动画自动播放，并在播放结束后自动跳到下一页。

1. 画面配置

在【开机动画】画面的“背景图片”导入相应的美工图片。在该画面中添加动画控件，如图 6-5 所示。

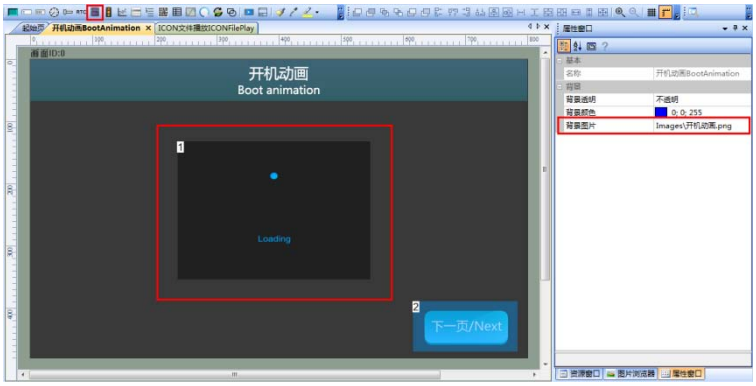



图 6-5 画面配置

2. 属性配置

设置动画控件（控件 ID：1）的属性，【动画文件】选择 GIF 文件、设置【间隔时间】为“35”、【播放结束时】为“切换到下一帧”，配置如图 6-6 所示。

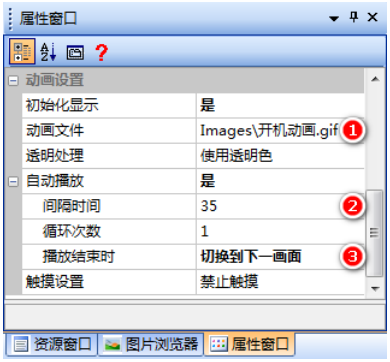



图 6-6 动画控件属性栏窗

3. 运行预览

运行虚拟屏，VisualTFT 软件和虚拟屏用“虚拟串口”建立联机。打开【指令助手】，选择左侧导航栏【动画控件】，设置“动画控件显示”区域的指令参数。对画面（画面 ID：0）中的动画控件（控件 ID：1）暂停播放。则在虚拟屏中，该动画控件暂停，运行效果如图 6-7 所示。

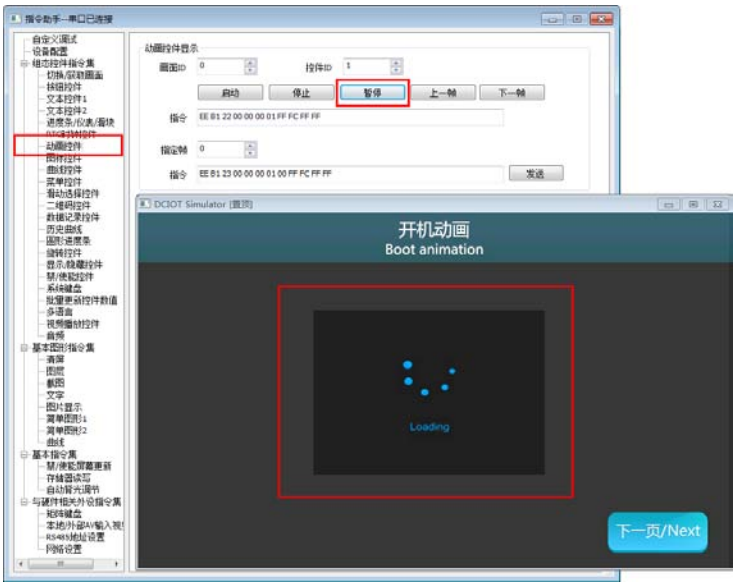



图 6-7 动画暂停播放

4. 外部 MCU 控制

参考开发包 keil 程序中的 him.dever.h 文件函数声明以及 him.dever.c 文件中的定义。

(1) MCU 控制开启播放动画

例如用户需开始播放动画，则直接调用播放动画函数 AnimationStart ( )即可，代码如程序清单 1 所示。

程序清单 1 播放动画

```
/*********************************************************************
** Function name:       void AnimationStart(uint16 screen_id,
**                                          uint16 control_id);
** Descriptions:        开始播放动画
** input parameters:    screen_id:     画面 ID
**                      control_id:    控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationStart(0, 1); //画面 ID0，控件 ID1,开始播放动画
    ……
}
```

(2) MCU 控制停止播放动画

例如用户需停止播放动画，则直接调用停止播放动画函数 AnimationStop ( )即可，代码如程序清单 3 所示。

程序清单 2 停止播放动画

```
/*********************************************************************
** Function name:       void AnimationStop(uint16 screen_id,
```

```
**                                              uint16 control_id);
** Descriptions:          停止播放动画
** input parameters:     screen_id:      画面 ID
**                       control_id:     控件 ID
** output parameters:    无
** Returned value:       无
*********************************************************************/
{
    ……
    AnimationStop(0, 1); //画面 ID0，控件 ID1,停止播放动画
    ……
}
```

### 6.4.3 ICON 文件播放

【ICON 文件播放】画面，介绍制作 ICON 文件、播放 ICON 文件设置。

1. 画面配置

在【ICON 文件播放】画面的“背景图片”导入相应的美工图片。在该画面中添加动画控件，如图 6-8 所示。




图 6-8 画面配置

2. 制作 ICON 文件

在 VisualTFT 软件中，打开【菜单】-【工具】-【图标生成】，如图 6-9 所示。

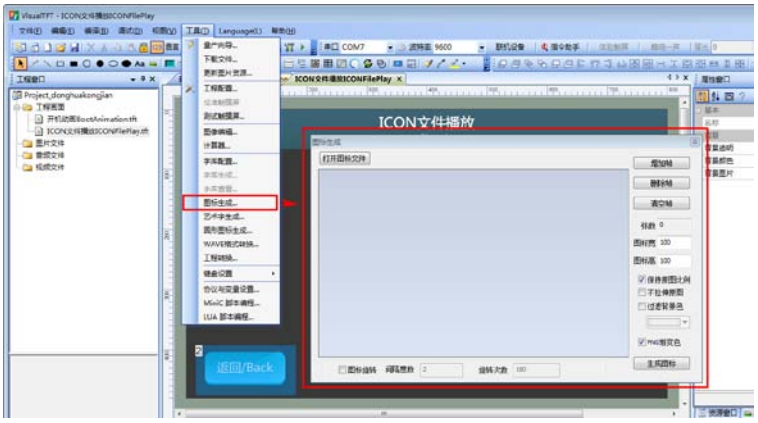



图 6-9 图标生成工具

点击【添加帧】，添加图片，如图 6-10 所示。

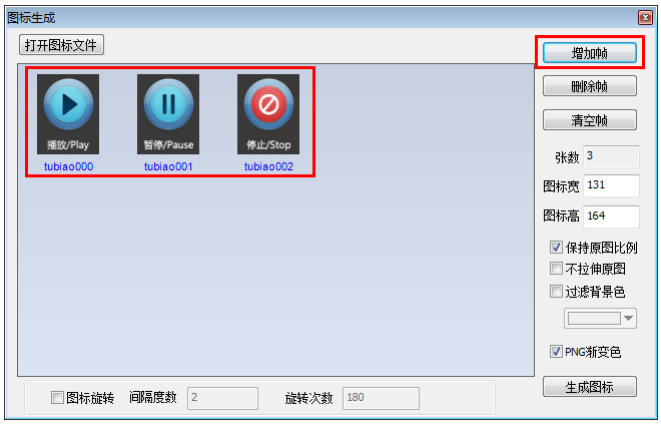



图 6-10 添加帧

帧添加完成之后，我们通过拖动可以调整帧与帧之间的排列，帧的排列决定了最后生成图标的帧数，如图6-11所示，“播放/Play”帧为第0帧；“暂停/Pause”帧为第1帧；“停止/Stop”为第2帧。帧数添加排列完毕后点击生成图标。




图 6-11 调整帧的顺序

最后保存好生成的图标，如图6-12所示。

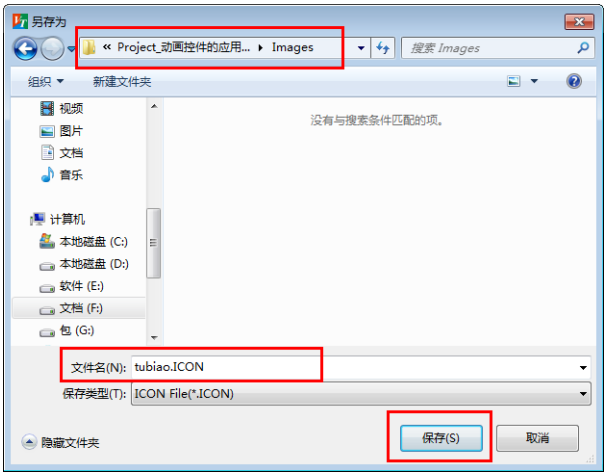



图 6-12 保持为 ICON 文件

3. 属性配置

设置动画控件（控件 ID：1）的属性，【动画文件】选择制作好的 ICON 文件、设

置【间隔时间】为“550”，配置如图 6-13 所示。

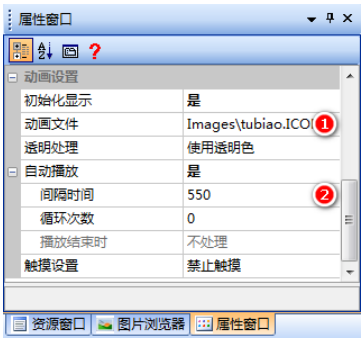



图 6-13　动画控件属性栏窗

4. 运行预览

运行虚拟屏，VisualTFT 软件和虚拟屏用“虚拟串口”建立联机。打开【指令助手】，选择左侧导航栏【动画控件】，设置“动画控件显示”区域的指令参数。对画面（画面 ID：1）中的动画控件（控件 ID：1）暂停播放，然后手动显示下一帧。则在虚拟屏中，该动画控件暂停并显示下一帧，运行效果如图 6-14 所示。




图 6-14　动画显示下一帧

5. 外部 MCU 控制

参考开发包 keil 程序中的 him.dever.h 文件函数声明以及 him.dever.c 文件中的定义。

(1) MCU 控制停止播放动画

例如用户需暂停动画，则直接调用停止动画函数 AnimationStop ( )即可，代码如程序清单 3 所示。

程序清单 3　暂停动画

```
/*********************************************************************
** Function name:       void AnimationStop(uint16 screen_id,
**                                         uint16 control_id);
** Descriptions:        停止播放动画
** input parameters:    screen_id:     画面 ID
```

```
**                      control_id:   控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationStop(0, 1); //画面 ID0，控件 ID1,停止播放动画
    ……
}
```

(2) MCU 控制显示下一帧

例如用户需显示动画的下一帧，则直接调用显示动画下一帧函数 AnimationPlayNext ( )即可，代码如程序清单 4 所示。

程序清单 4 显示下一帧

```
/*********************************************************************
** Function name:       void AnimationPlayNext(uint16 screen_id,
**                                             uint16 control_id);
** Descriptions:        播放下一帧
** input parameters:    screen_id:      画面 ID
**                      control_id:     控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationPlayNext(0, 1); //画面 ID0，控件 ID1,播放下一帧
    ……
}
```

## 6.5 下载工程

在我司的上层软件 Visual TFT 把工程编译成功后，可以通过以下两种方法下载到大彩串口屏设备中：

1. 串口下载；
2. SD 卡下载。

注：本例程中以 F 型为例，不同平台下载方式略有不同，具体参考不同平台的下载文档。

## 6.6 联机调试

### 6.6.1 硬件连线

硬件连接接线如图 6-15 所示。




图 6-15　硬件连线

### 6.6.2 联机设备

将工程下载到串口屏后，在 VisualTTF 的菜单栏，点击【联机设备】，如图 6-16 所示。

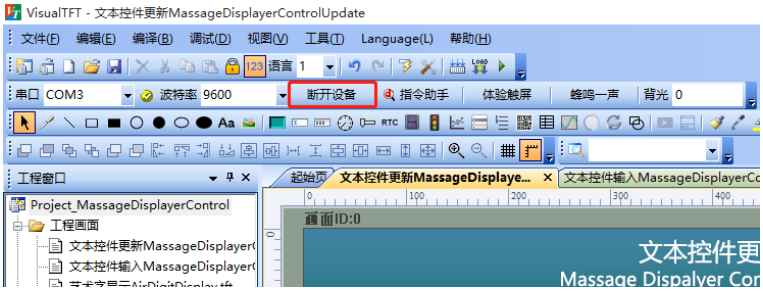



图 6-16　联机设备

### 6.6.3 指令控制

打开【指令助手】，选择左侧导航栏【动画控件】，设置“动画控件显示”区域的指令参数。如设置画面 1 的动画控件 1 为第 2 帧，如图 6-17 所示。




图 6-17　文本更新显示

### 6.6.4 串口屏效果

串口屏接收到指令后，图标切换到指定帧。

# 7. 部分程序代码

我司提供用户单片机控制程序，可前往下载开发包，里面提供 API 函数接口，用户可移植相应的程序；本例程涉及的用户单片机控制动画控件的程序，参考开发包 keil 程序中的 him.dever.h 文件函数声明以及 him.dever.c 文件中的定义。

keil 程序中的函数定声明如程序清单 5 所示。

程序清单 5　动画控件 keil 程序

```
/*********************************************************************
** Function name:       void AnimationStart(uint16 screen_id,
**                                          uint16 control_id);
** Descriptions:        开始播放动画
** input parameters:    screen_id:     画面 ID
**                      control_id:    控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationStart(0, 1); //画面 ID0，控件 ID1,开始播放动画
    ……
}

/*********************************************************************
** Function name:       void AnimationStop(uint16 screen_id,
**                                         uint16 control_id);
** Descriptions:        停止播放动画
** input parameters:    screen_id:     画面 ID
**                      control_id:    控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationStop(0, 1); //画面 ID0，控件 ID1,停止播放动画
    ……
}

/*********************************************************************
** Function name:       void AnimationPause(uint16 screen_id,
**                                          uint16 control_id);
** Descriptions:        暂停播放动画
** input parameters:    screen_id:     画面 ID
**                      control_id:    控件 ID
```

```
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationPause(0, 1); //画面 ID0，控件 ID1,暂停播放动画
    ……
}

/*********************************************************************
** Function name:       void AnimationPlayFrame(uint16 screen_id,
**                                              uint16 control_id,
**                                              uint8 frame_id);
** Descriptions:        播放动画指定帧
** input parameters:    screen_id:     画面 ID
**                      control_id:  控件 ID
**                      frame_id:      帧 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationPlayFrame(0, 1, 2); //画面 ID0，控件 ID1,播放第二帧
   ……
}

/*********************************************************************
** Function name:       void AnimationPlayPrev(uint16 screen_id,
**                                             uint16 control_id);
** Descriptions:        播放上一帧
** input parameters:    screen_id:     画面 ID
**                      control_id:    控件 ID
** output parameters:   无
** Returned value:      无
*********************************************************************/
{
   ……
   AnimationPlayPrev(0, 1); //画面 ID0，控件 ID1,播放上一帧
   ……
}

/*********************************************************************
** Function name:       void AnimationPlayNext(uint16 screen_id,
**                                             uint16 control_id);
```

```
** Descriptions:        播放下一帧
** input parameters:   screen_id:     画面 ID
**                      control_id:    控件 ID
** output parameters:  无
** Returned value:      无
*********************************************************************/
{
    ……
    AnimationPlayNext(0, 1); //画面 ID0，控件 ID1,播放下一帧
    ……
}
```

# 8. 免责声明

本文档提供有关广州大彩光电科技有限公司（以下简称：大彩科技）产品的信息，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，若在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。大彩科技产品不能在用于军事、医疗、救生或维生等用途中作为唯一控制设备。

本文档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除大彩科技在其产品的销售条款和条件中声明的责任之外，大彩科技概不承担任何其它责任。并且，大彩科技对大彩科技产品的销售和/或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。大彩科技可能随时对产品规格及产品描述做出修改，恕不另行通知。