技术笔记

Technical Note

教程 6 - 串口屏 RTC 控件的应用 V1.0

工程技术笔记

类别	内容
关键词	时间显示、定时功能
摘要	RTC 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	2019/07/30	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. RTC控件属性.....5
5.1 属性介绍.....5
6. 教程实现.....9
6.1 准备UI素材.....9
6.2 开发工具.....9
6.3 软件版本.....10
6.4 功能实现.....10
6.4.1 建立工程.....10
6.4.2 时间显示.....11
6.4.3 定时功能.....12
6.5 下载工程.....18
6.6 联机调试.....18
6.6.1 硬件连线.....18
6.6.2 联机设备.....18
6.6.3 指令控制.....18
6.6.4 串口屏效果.....19
6.7 部分程序代码.....20
7. 免责声明.....22

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。

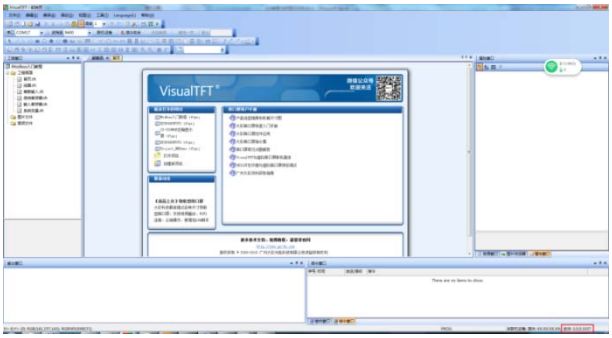


图 2-2 软件版本

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

时间可以通过 RTC 控件来显示。RTC 控件可以通过用户主机修改，也可以通过设置弹出系统键盘，对时间进行输入编辑操作，然后显示在串口屏上。

本例程中介绍文本控件常见的应用，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. 时间显示：显示当前时间。
2. 定时功能：设置一个计时，在屏上显示计时的时间，在达到设定值时通过串口发出通知。

5. RTC 控件属性

5.1 属性介绍

RTC 控件可以显示当前时间和倒计时，如图 5-1 所示。

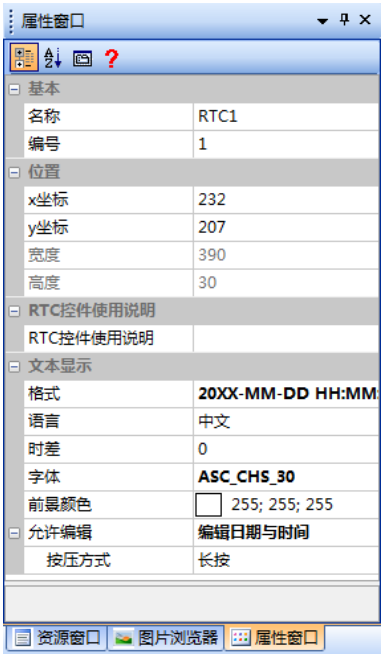


图 5-1 RTC 控件属性

1. 显示时间，显示当前时间。

(1) 格式：时间的显示格式，如图 5-2 所示。



图 5-2 显示格式

a) HH:MM:SS，显示小时、分钟和秒数；
b) 20XX-MM-DD，显示年份、月份和日数；
c) 20XX-MM-DD HH:MM:SS，显示年份、月份、日数、小时、分钟和秒数；
d) 20XX-MM-DD HH:MM:SS 星期几，显示年月日，时间以及星期几；
e) 星期几（中文+英文），显示星期几的中文和英文；
f) 自定义格式。例如“日-月-年,时-分-秒,星期”的格式为：%d-%n-%y,%h:%m:%s,%w(年=%h，月=%d，日=%d，%h，分=%m，秒=%s，星期=%w)，如图 5-3 和图 5-4 所示。

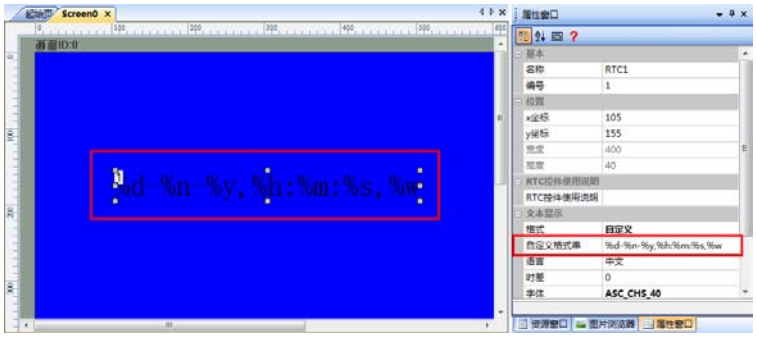


图 5-3 设置自定义格式串

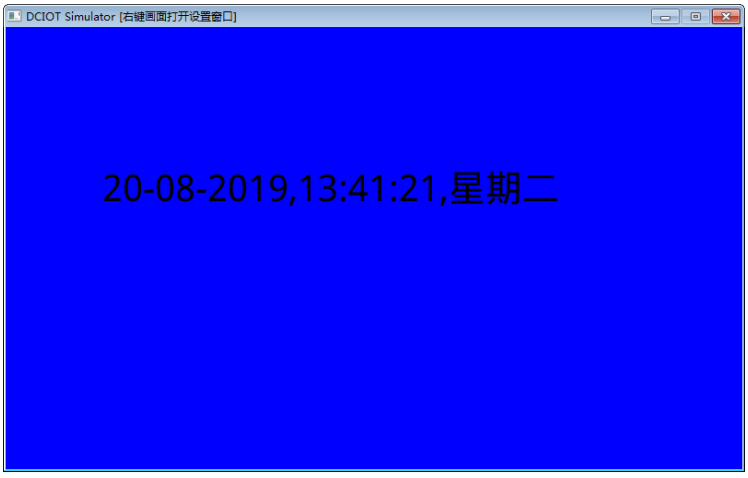


图 5-4 自定义格式的显示

(2) 语言：中文、英文。
(3) 时差：与当前系统时间的差值，范围[-12~+12]。
(4) 字体：字体大小和字体样式。
(5) 前景颜色：字体颜色。

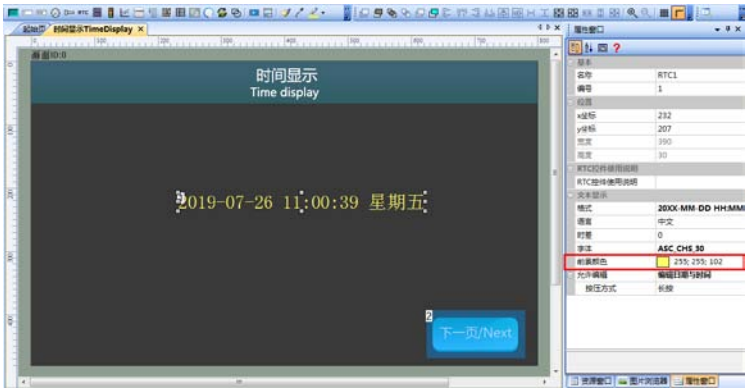


图 5-5　改变字体颜色

(6)　允许编辑：点击 RTC 显示区域，弹出日期/时间设置界面，如图 5-6 所示。

a)　按压方式：长按/点击进入编辑。

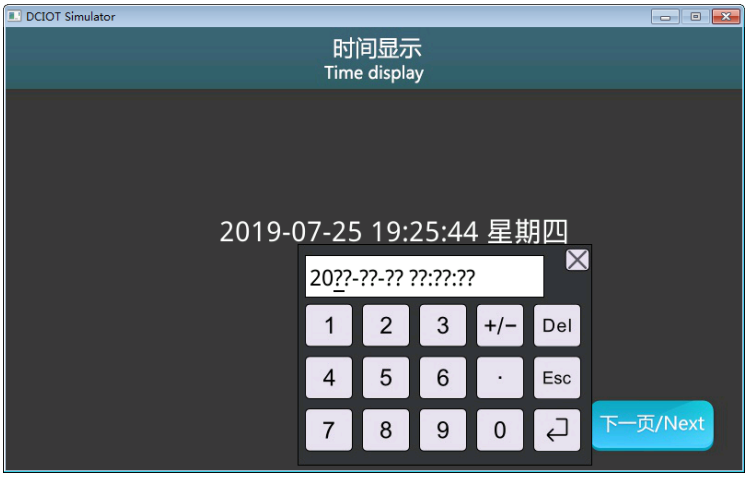


图 5-6　长按编辑时间

2.　计时

(1)　格式：计时时间的显示格式，如图 5-7 所示。

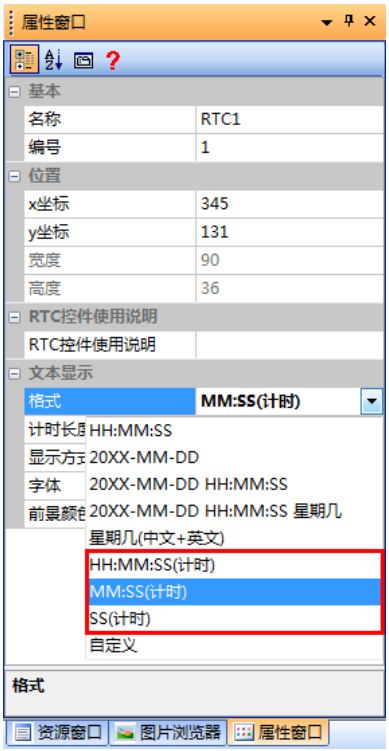


图 5-7　计时时间的格式

(2)　计时长度：计时时间长度（秒）。

(3)　显示方式：倒计时、顺计时。

(4)　字体：字体大小。

(5)　前景颜色：字体颜色。

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。

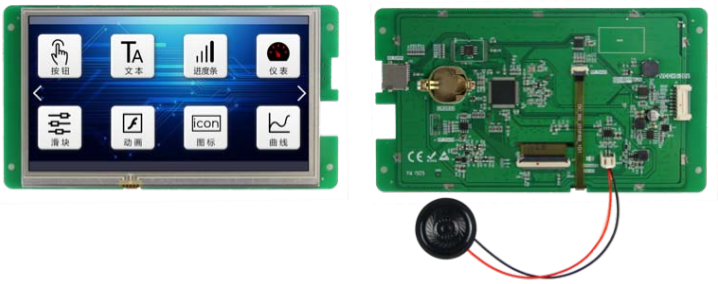


图 6-1 F 型 7 寸电阻屏

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，如表 6-1 所示，软件开发工具如表 6-2 所示。

表 6-1 工具清单

配件	作用
电源适配器 5V/9V/12V 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）
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。

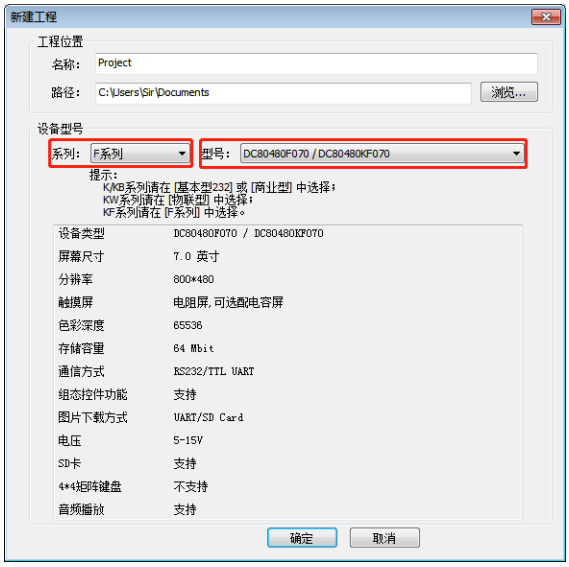


图 6-4 新建工程

6.4.2 时间显示

【时间显示】画面，介绍设置显示当前时间的方法。

1. 画面配置

在【时间显示】画面的“背景图片”导入相应的美工图片。在该画面中添加 RTC 控件，如图 6-5 所示。

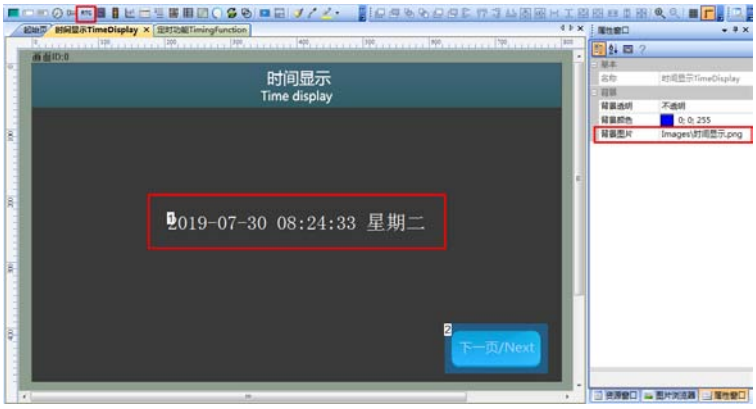


图 6-5 画面配置

2. 属性配置

以控件 ID 为 1 的 RTC 控件为例，设置【格式】为“20XX-MM-DD HH:MM:SS 星期几”、【字体】为“ASC_CHS_30”、【前景颜色】为白色，配置如图 6-6 所示。

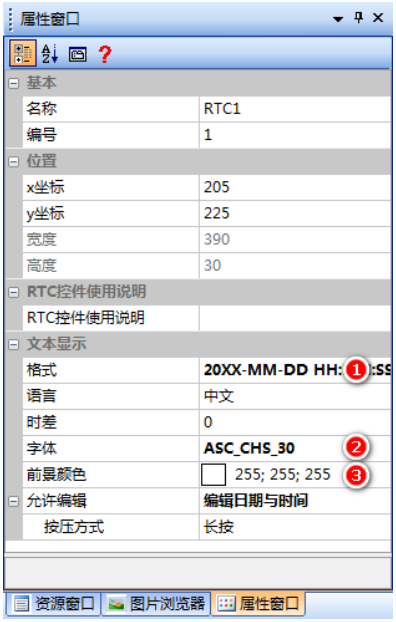


图 6-6 RTC 控件属性栏窗

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【RTC 时钟控件】，在“RTC 时钟设置”勾选“系统当前时间”。如例程中，设置画面（画面 ID：0）中的 RTC 控件（控件 ID：1），勾选“系统当前时间”，点击“设置”。则虚拟屏中该 RTC 控件显示当前系统时间，运行效果如图 6-7 所示。

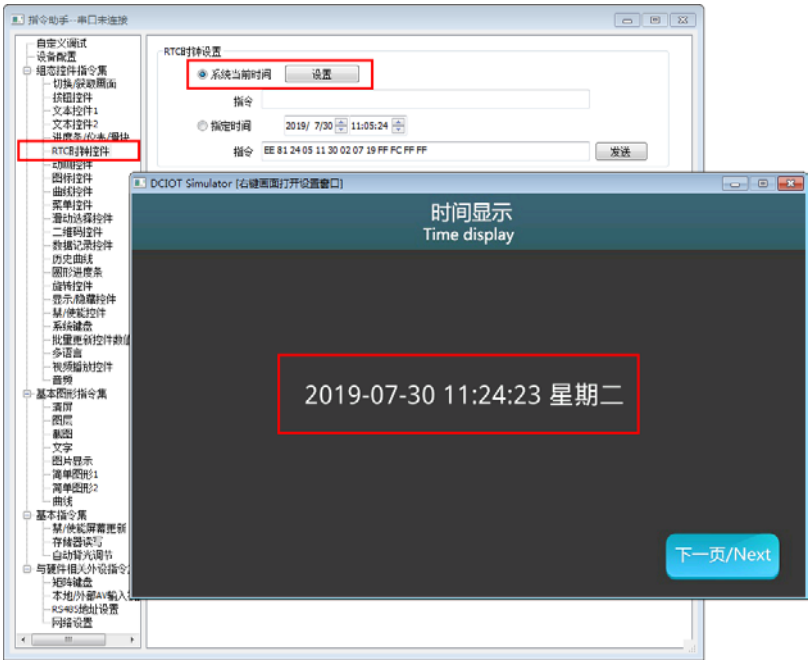


图 6-7 RTC 控件

4. MCU 获取 RTC 系统时间

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。

例如用户需获取串口屏的系统时间，则直接调用获取 RTC 函数 `ReadRTC()`即可，代码如程序清单 1 所示。

程序清单 1 读取时间

```
/*********************************************************************
** Function name:       void ReadRTC(void);
** Descriptions:        获取串口屏中的系统时间
** input parameters:    无
** output parameters:   无
*********************************************************************/
{
    ……
    ReadRTC();
    ……
}
```

6.4.3 定时功能

【定时功能】画面，介绍按下按钮后启动、暂停计时器的功能。

1. 画面配置

在【定时功能】画面的“背景图片”导入相应的美工图片。在该画面中添加 1 个 RTC 控件（控件 ID：1），2 个按钮控件（控件 ID：3 和 4），如图 6-8 所示。



图 6-8　画面配置

2.　属性配置

以设置控件 ID 为 1 的 RTC 控件作顺计时为例，设置【格式】为“MM:SS(计时)”、【计时长度】为“30”、【显示方式】为“顺计时”、【字体】为“ASC_CHS_36”，配 置如图 6-9 所示。

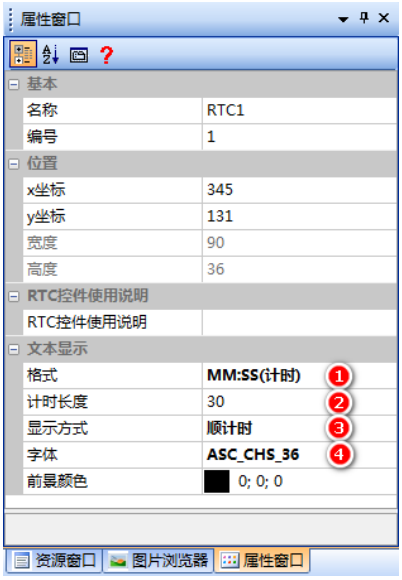


图 6-9　RTC 控件顺计时属性栏窗

设置控件 ID 为 1 的 RTC 控件作倒计时，原理同上，只需把【显示方式】为“倒计时”。

设置按钮开启（控件 ID：3）的属性，设置【触摸虚框】为“否”、【按下时的图片】选择设置好的图片、勾选【裁剪】，配置如图 6-10 所示。

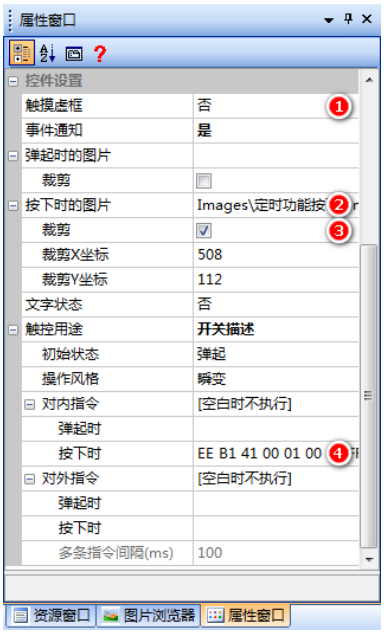


图 6-10　按钮控件属性栏窗

选择按钮控件属性栏中的【对内指令】→“按下时”，弹出指令填写框，选择“指令助手”，如图 6-11 所示。

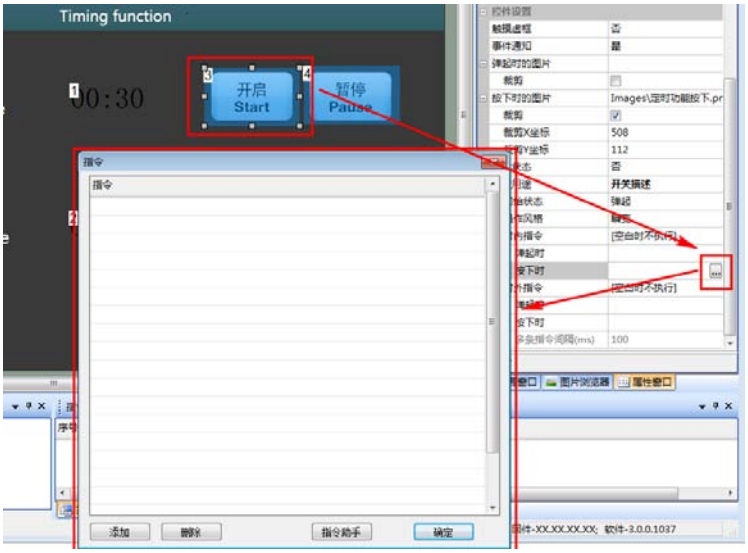


图 6-11　对内指令

在指令助手界面，左侧导航栏选择【RTC 时钟控件】，设置“设置定时器”区域的指令参数，填写对应的参数。如例程中，控制画面（画面 ID：1）中的图标控件（控件 ID：1），【操作类型】为“开启定时器”，如图 6-12 所示。

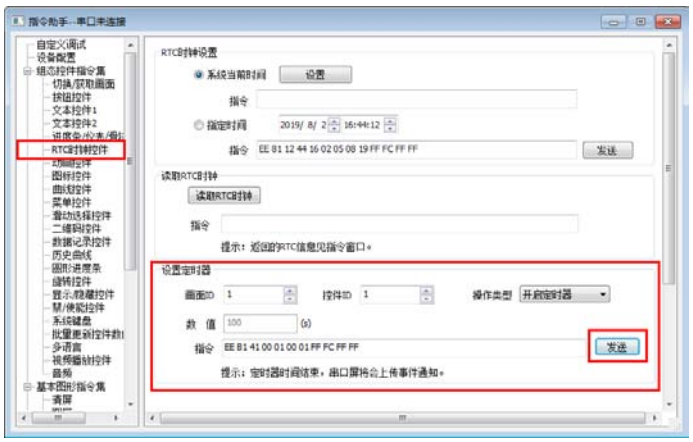


图 6-12 指令助手

复制此指令，关闭指令助手，将指令粘贴到指令框中，如图 6-13 所示。

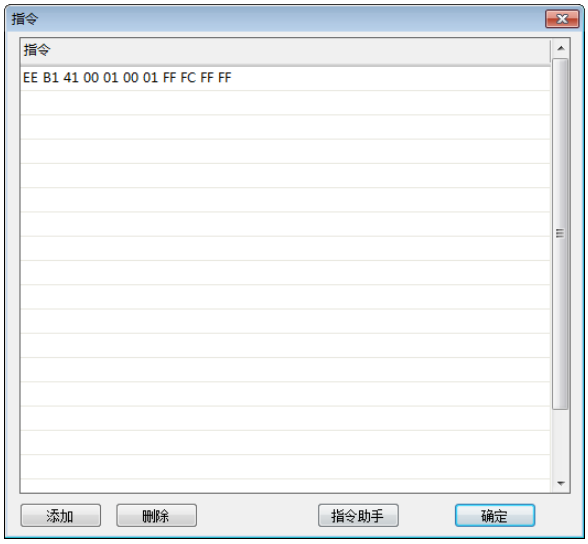


图 6-13 填写指令

按钮暂停（控件 ID：4）的指令框中设置画面 ID：1，控件 ID：1 的顺计时停止。
按钮开启（控件 ID：5）的指令框中设置画面 ID：1，控件 ID：2 的倒计时开启。
按钮暂停（控件 ID：6）的指令框中设置画面 ID：1，控件 ID：2 的倒计时停止。

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【RTC 时钟控件】，在“设定定时器”的操作类型中选择“设置定时器”。如例程中，控制画面（画面 ID：1）中的 RTC 控件（控件 ID：2），填写【数值】为“55”，点击“发送”。则虚拟屏该 RTC 控件显示“00:55”，运行效果如图 6-14 所示。



图 6-14 RTC 控件

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 设置计时时间

例如用户需设置串口屏的计时时间，则直接调用设置 RTC 计时的函数 SeTimer() 即可，函数定义如程序清单 2 所示。

程序清单 2 设置计时值

```
/*************************************************************
** Function name:       void SeTimer(uint16 screen_id,
**                                   uint16 control_id,
**                                   uint32 timeout);
** Descriptions:        设置计时控件
** input parameters:    screen_id   画面 ID
**                      control_id 控件 ID
**                      timeout     倒计时(秒)
** output parameters:   无
*************************************************************/
{
    ……
    SeTimer(1,1,50);   //设置画面 1 的 RTC 控件 1 的值为 50S
    ……
}
```

(2) MCU 开启计时

例如用户需开启串口屏的计时时间，则直接调用开启 RTC 计时的函数

StartTimer()即可，函数定义如程序清单 3 所示。

程序清单 3 开启计时

```
/*********************************************************************
** Function name:       void StartTimer(uint16 screen_id,
**                                      uint16 control_id);
** Descriptions:        开启计时控件
** input parameters:    screen_id   画面 ID
**                      control_id 控件 ID
** output parameters:   无
*********************************************************************/
{
   ……
   StartTimer(1,1);   //开启画面 1 的 RTC 计时控件 1
   ……
}
```

(3) MCU 停止计时

例如用户需停止串口屏的计时时间，则直接调用停止 RTC 计时的函数 StopTimer ()即可，函数定义如程序清单 4 所示。

程序清单 4 停止计时

```
/*********************************************************************
** Function name:       void StopTimer(uint16 screen_id,
**                                     uint16 control_id);
** Descriptions:        停止计时控件
** input parameters:    screen_id   画面 ID
**                      control_id  控件 ID
** output parameters:   无
*********************************************************************/
{
   ……
   StopTimer(1,1);   //停止画面 1 的 RTC 计时控件 1
   ……
}
```

(4) MCU 暂停计时

例如用户需暂停串口屏的计时时间，则直接调用暂停 RTC 计时的函数 PauseTimer ()即可，函数定义如程序清单 5 所示。

程序清单 5 暂停计时

```
/*********************************************************************
** Function name:       void PauseTimer(uint16 screen_id,
**                                      uint16 control_id);
** Descriptions:        暂停计时控件
** input parameters:    screen_id   画面 ID
```

```
**                          control_id 控件 ID
** output parameters:   无
*********************************************************************/
{
    ……
    PauseTimer(1,1);   //暂停画面 1 的 RTC 计时控件 1
    ……
}
```

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。

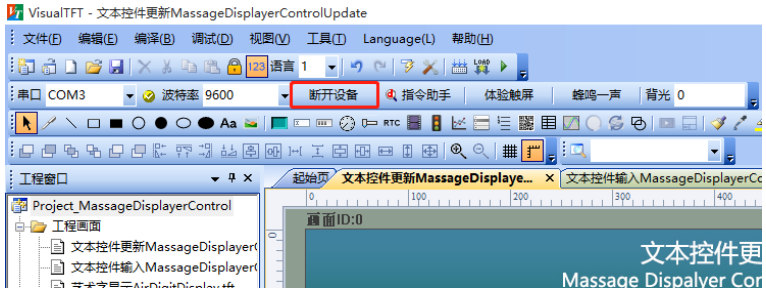


图 6-16 联机设备

6.6.3 指令控制

打开【指令助手】，选择左侧导航栏【RTC 时钟控件】，在“RTC 时钟设置”区域，勾选指定时间，在右边选项栏中设置时间，点击发送。如设置画面 0 的 RTC 控件 1 的显示设置的时间，如图 6-17 所示。



图 6-17　设置 RTC 时间

6.6.4 串口屏效果

串口屏接收到指令后，RTC 控件 1 显示“2019-07-30 11:05:24 星期二”。

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 RTC 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 6 所示。

程序清单 6　RTC 控件 keil 程序

```
/*********************************************************************
** Function name:       void ReadRTC(void);
** Descriptions:        获取串口屏中的系统时间
** input parameters:    无
** output parameters:   无
*********************************************************************/
{
  ……
  ReadRTC();
  ……
}

/*********************************************************************
** Function name:       void SeTimer(uint16 screen_id,
**                                   uint16 control_id,
**                                   uint32 timeout);
** Descriptions:        设置计时控件
** input parameters:    screen_id   画面 ID
**                      control_id 控件 ID
**                      timeout     倒计时(秒)
** output parameters:   无
*********************************************************************/
{
  ……
  SeTimer(1,1,50);   //设置画面 1 的 RTC 控件 1 的值为 50S
  ……
}

/*********************************************************************
** Function name:       void StartTimer(uint16 screen_id,
**                                      uint16 control_id);
** Descriptions:        开启计时控件
** input parameters:    screen_id   画面 ID
**                      control_id 控件 ID
** output parameters:   无
*********************************************************************/
```

```
{
  ……
  StartTimer(1,1);   //开启画面 1 的 RTC 计时控件 1
  ……
}

/*************************************************************************
** Function name:       void StopTimer(uint16 screen_id,
**                                     uint16 control_id);
** Descriptions:        停止计时控件
** input parameters:    screen_id   画面 ID
**                      control_id  控件 ID
** output parameters:   无
*************************************************************************/
{
   ……
   StopTimer(1,1);   //停止画面 1 的 RTC 计时控件 1
   ……
}

/*************************************************************************
** Function name:       void PauseTimer(uint16 screen_id,
**                                      uint16 control_id);
** Descriptions:        暂停计时控件
** input parameters:    screen_id   画面 ID
**                      control_id 控件 ID
** output parameters:   无
*************************************************************************/
{
  ……
   PauseTimer(1,1);   //暂停画面 1 的 RTC 计时控件 1
  ……
}
```

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。