乐升半导体
LEVETOP

LEVETOP

串口屏方案

Serial Uart TFT Panel Solution

应 用 手 册

Application Note

V1.0

www.levetop.cn
Levetop Semiconductor Co., Ltd.

版 本 记 录

版 别	发 布 日 期	改 版 说 明
V1.0	2022/02/09	● 应用手册整合版本初版。

© Levetop Semiconductor Co., Ltd.

目　　录

目　　录 3
图 目 录 6
表 目 录 12
1. 概述 13
1.1 TFT 串口屏的软硬件架构 13
1.2 TFT 串口屏的演示板（Demo Kit) 16
1.3 串口屏开发流程 18
1.4 TFT 串口屏的升级 21
1.4.1 使用 LT_VCOM_GUI 软件更新 22
1.4.2 使用 LT_Uart_GUI 软件更新 25
1.4.3 用 SD 卡更新 UartTFT_Flash.bin 30
1.4.4 用 U 盘更新 31
1.4.5 LT7689 的升级与更新 34
2. 串口指令 36
2.1 串口屏指令集 36
2.2 主控端与 TFT 串口屏协议表 39
2.3 各芯片支持的指令 44
2.4 RS-232(UART) 通讯协议 47
2.5 主控端发送指令的范例 49
3. 图文 UI 编辑器（UI_Editor.exe） 52
3.1 UI_Editor 界面的简介 52
3.1.1 使用 UI_Editor 的设计流程 57
3.1.2 使用 UI_Editor 的注意事项 59
3.1.3 屏幕初始化和背景图的设定 61
3.1.4 显示图的设定 64
3.1.5 界面的编辑与调试 67
3.2 显示图片的设定 72
3.2.1 设定显示多张图片 72
3.2.2 设定循环显示重叠图片 73

© Levetop Semiconductor Co., Ltd.

3.2.3 设定卷动出现图片 74
3.2.4 设定循环卷动图片 75
3.2.5 设定显示 GIF 动画图片 76
3.2.6 设定显示二维码图片 77
3.2.7 设定显示控件功能的图片 78
3.2.8 设定显示虚拟按键 79
3.2.9 设定开机画面 80
3.2.10 设定触摸进度条图 83
3.2.11 设定环形触摸进度条 86
3.2.12 设定指针图 87
3.2.13 设定进度条图 90
3.2.14 设定带透明度的单张图片 93
3.2.15 控件滑动及显示 94
3.2.16 设定显示数字键盘 95
3.2.17 实时曲线的设定 98
3.2.18 设定环形指标图 99
3.2.19 固定图片显示指令 100
3.2.20 摄像头显示指令 101
3.3 显示文字的设定 103
3.3.1 图像形式显示文字 103
3.3.2 图片式的数字显示 104
3.3.3 真彩图片式数字显示 106
3.3.4 使用字库显示文字 107
3.3.5 图片式的 ASCII 显示 108
3.4 几何绘图功能 110
3.4.1 基本绘图功能 110
3.4.2 表格制作 114
3.5 其他功能 115
3.5.1 全屏触摸滑动设定及显示指令 115
3.5.2 背光控制指令 116
3.5.3 起始控制指令 117
3.5.4 图片缓冲指令 117
3.5.5 声音控制指令 118
3.5.6 串口屏侦测指令 119
3.5.7 User 手动输入指令 121
3.5.8 RTC 数字时钟指令 122
3.5.9 RTC 模拟时钟指令 123

© Levetop Semiconductor Co., Ltd.

3.5.10 显示星期指令 125
3.5.11 寄存器操作 126
3.5.12 关闭触控功能指令 127
3.5.13 全键盘指令 128
3.5.14 滑动手势指令 129
3.5.15 电阻屏控制指令 130
3.5.16 时钟控制指令 131
3.5.17 串口屏复位 132
3.5.18 设定音量命令 132
3.5.19 影音播放控制命令 133
3.5.20 影音播放命令 134
3.5.21 使用矢量字库的文字设定及显示指令 136
3.6 WAV 音频、字库文件的说明 137
3.6.1 WAV 音频文件说明 137
3.6.2 字库文件的说明 137
3.6.3 真彩数字图片文件说明 138
3.7 BIN 文件的生成和串口发送指令 139
3.8 UI_Editor 范例下载 145
3.9 支持压缩图片的指令 147
4. 制作字库与 Wav 档的 Bin 文件 148
4.1 制作字库的 Bin 文件 148
4.1.1 全字库制作 148
4.1.2 自定义字库制作 151
4.2 制作 Wav 檔的 Bin 文件 154
4.2.1 音频文件转 WAV 154
4.2.2 制作 WAV 档 Bin 文件 159
4.2.3 典型音频驱动电路 162
5. 版权说明 163

© Levetop Semiconductor Co., Ltd.

图 目 录

图 1-1：主控端 MCU 的 Uart 与串口屏的 Uart 连接示意图 13
图 1-2：使用上位机软件开发的示意图 14
图 1-3：主控板与 TFT 串口屏连接 14
图 1-4：串口屏的指令协议范例一 15
图 1-5：串口屏的指令协议范例二 15
图 1-6：LT7689 串口屏的开发演示板（Demo Kit） 16
图 1-7：LT268C 串口屏的开发演示板 - 1 16
图 1-8：LT268C 串口屏的开发演示板 - 2 17
图 1-9：开发流程（1） 18
图 1-10：开发流程（2） 19
图 1-11：开发流程（3） 20
图 1-12：用 LT_VCOM_GUI 更新 22
图 1-13：开启软件 LT_VCOM_GUI_Vxx.exe 22
图 1-14：选择更新项目 23
图 1-15：更新完成后进行重置和运行程序 24
图 1-16：LT268C Uart 升级接线 25
图 1-17：打开 LT_Uart_GUI 软件 25
图 1-18：MCU_Code.bin 更新成功 26
图 1-19：更新 Flash 程序配置 27
图 1-20：Flash 程序更新完成 28
图 1-21：软件文件 28
图 1-22：添加 Flash ID 29
图 1-23：进入更新倒计时画面 30
图 1-24：UartTFT_Flash.bin 数据更新中 30
图 1-25：更新 U 盘 Bootloader 程序 31
图 1-26：更新的程序名称及对应的文件夹 31
图 1-27：以 SD Card 进行更新中 U 盘更新硬件配置 32
图 1-28：U 盘更新程序界面 32
图 1-29：U 盘/SD 卡格式化 33
图 1-30：SD Card 插入 SD 卡槽内 34
图 1-31：以 SD Card 进行更新中 35
图 3-1：UI_Editor 主画面 52
图 3-2：UI_Editor 工具同级文件目录 54
图 3-3：PROJECT 文件夹下级的工程文件目录 55
图 3-4：UI_Editor 重装载工程文件 56
图 3-5：使用 UI_Editor 的设计流程 57

© Levetop Semiconductor Co., Ltd.

图 3-6：不同的工程建立不同的项目目录.....59
图 3-7：点击“SAVE”保存当前制作的工程.....60
图 3-8：点击编译“Build All”保存当前制作的工程.....60
图 3-9：可以自行选择删除不需要的工程文件夹.....60
图 3-10：打开的 UI_Editor.....61
图 3-11：新建工程.....61
图 3-12：设置图层分辨率.....62
图 3-13：拖动屏幕框到任意位置.....62
图 3-14：取消了屏幕框选择功能.....63
图 3-15：增加图层.....64
图 3-16：新建图层与主图层.....64
图 3-17：新建独立的图层.....65
图 3-18：变更图层背景.....65
图 3-19：变更主图层背景.....66
图 3-20：删除或移动图层.....66
图 3-21：删除操作方式一.....67
图 3-22：删除操作方式二.....67
图 3-23：异位置克隆操作.....68
图 3-24：同位置克隆操作.....68
图 3-25：复制功能一.....69
图 3-26：复制功能二.....69
图 3-27：复制功能三.....70
图 3-28：编辑状态.....70
图 3-29：调试状态.....71
图 3-30：设定显示多张图片.....72
图 3-31：设定循环显示重叠图片.....73
图 3-32：设定卷动出现图片.....74
图 3-33：设定循环卷动图片.....75
图 3-34：设定显示 GIF 动画图片.....76
图 3-35：设定显示二维码.....77
图 3-36：设定显示控件功能的图片.....78
图 3-37：设定显示虚拟控件.....79
图 3-38：点选左上角“mainLay”.....80
图 3-39：选择每一组 Command 的指令组合.....81
图 3-40：开机画面所选的指令.....81
图 3-41：以 9A 指令来设定执行多组指令.....82
图 3-42：以 9A、01 指令来执行多组指令.....82
图 3-43：设定滑动条及前景颜色与背景颜色.....83

© Levetop Semiconductor Co., Ltd.

图 3-44：设定滑动条背景图片 83
图 3-45：新增滑动条的移动图标头 84
图 3-46：背景图片与滑动图标头各自对齐 84
图 3-47：设定滑动条背景图片与移动图标模式 85
图 3-48：背景图片 Command 选择窗口 85
图 3-49：滑动图标头 Command 选择窗口 85
图 3-50：选择完毕后点击确认 85
图 3-51：添加环形触控进度条底图和触控图标 86
图 3-52：设定环形进度条 86
图 3-53：设定指针对左右两侧的背景颜色 87
图 3-54：添加指针背景图片 88
图 3-55：调整指针背景图片显示位置 88
图 3-56：选择指针背景图片 Command 89
图 3-57：点选指针背景图片模式第一组数据 89
图 3-58：为指针添加背景图片指令 89
图 3-59：设定进度条背景与进度条移动时的颜色 90
图 3-60：设定背景图片 90
图 3-61：调整进度条背景图片显示位置 91
图 3-62：选择进度条背景图片 Command 91
图 3-63：点选进度条背景图片模式数据 91
图 3-64：选择进度图的背景图片 92
图 3-65：设定带透明度的单张图片 93
图 3-66：控件滑动及显示 94
图 3-67：添加数字键盘底图 95
图 3-68：添加数字键盘底图 96
图 3-69：数字键盘设定 96
图 3-70：数字键盘设定 97
图 3-71：数字键盘设定 97
图 3-72：曲线指令设定 98
图 3-73：曲线参数设定 98
图 3-74：环状指标图参数设定 99
图 3-75：摄像头显示操作 102
图 3-76：图片形式显示文字 103
图 3-77：图片式的数字显示的设置窗口 104
图 3-78：调整图片式数字显示的字体框 105
图 3-79：调整真彩图片式数字显示的字体框 106
图 3-80：字库显示文字设置 107
图 3-81：图片式的 ASCII 显示设置 108

© Levetop Semiconductor Co., Ltd.

图 3-82：图片式的 ASCII 显示设置 108
图 3-83：图片式的 ASCII 显示设置 109
图 3-84：画圆设置 110
图 3-85：画四边形设置 111
图 3-86：画方柱体设置 112
图 3-87：画直线设置 112
图 3-88：画圆环设置 113
图 3-89：表格设置 114
图 3-90：触控滑动设定及显示指令 115
图 3-91：Command 发送 B4 指令 115
图 3-92：背光控制指令 116
图 3-93：背光控制指令 116
图 3-94：起始控制指令 117
图 3-95：图片缓存指令 117
图 3-96：UI_Editor 里的 WAV 音频文件 118
图 3-97：声音控制指令 118
图 3-98：声音控制指令 118
图 3-99：User 手动输入指令 121
图 3-100：数字时钟指令 92h 设置 122
图 3-101：模拟时钟指令 93h 设置 123
图 3-102：模拟时钟指令 93h 设置 123
图 3-103：模拟时钟指令 93h 设置 124
图 3-104：模拟时钟指令 93h 设置 124
图 3-105：显示星期指令 9Dh 设置 125
图 3-106：显示星期指令 9Dh 设置 125
图 3-107：CA-CF 操作举例说明 126
图 3-108：9Eh 操作举例说明 127
图 3-109：全键盘相关文件 128
图 3-110：设置 A6 指令 128
图 3-111：设置 C9 指令 129
图 3-112：电阻屏的 4 个角落校验 130
图 3-113：设定音量命令 132
图 3-114：设定音量命令 132
图 3-115：F7 指令设置 132
图 3-116：影音播放控制命令 133
图 3-117：影音播放控制命令 133
图 3-118：影音播放命令 134
图 3-119：影音播放命令 134

© Levetop Semiconductor Co., Ltd.

图 3-120：MP4 视频播放 135
图 3-121：矢量字库设定 136
图 3-122：Wav BIN 文件 137
图 3-123：FONT 文件 137
图 3-124：真彩数字图片文件 138
图 3-125：INFO 内配置参数 139
图 3-126：工程编译 139
图 3-127：工程编译成功 140
图 3-128：Command 发送指令给 TFT 串口屏 140
图 3-129：指令表导出 PDF 格式 141
图 3-130：解压缩及打开压缩包 145
图 3-131：范例目录拷贝到 UI_Editor 的 PROJECT 文件夹内 145
图 3-132：打开 UI_Editor 点击 Load 146
图 3-133：打开 UI_Editor 演示范例 146
图 4-1：制作中文字库 148
图 4-2：选择字体 149
图 4-3：设置字库 149
图 4-4：保存字库 150
图 4-5：字库制作完成 150
图 4-6：导出的字库 Bin 文件 151
图 4-7：选择 txt 文档 151
图 4-8：存放文字的 txt 文档 152
图 4-9：保存字库 152
图 4-10：字库制作完成 153
图 4-11：导出的字库 Bin 文件 153
图 4-12：选择功能 154
图 4-13：wav 功能界面 155
图 4-14：添加 wav 文件 155
图 4-15：输出配置 156
图 4-16：进入剪辑功能 156
图 4-17：剪辑界面 157
图 4-18：选择输出文件夹 157
图 4-19：开始转换 158
图 4-20：导出的 wav 文件 158
图 4-21：WAV Bin 文件制作界面 159
图 4-22：添加 wav 文件 159
图 4-23：添加成功 160
图 4-24：WAV Bin 文件制作界面 160

© Levetop Semiconductor Co., Ltd.

图 4-25：导出 bin 文件 161
图 4-26：错误提示 161
图 4-27：点击 OK 导出 161
图 4-28：导出成功 162
图 4-29：生成的 bin 文件 162
图 4-30：用 DAC_OUT 输出及小功率放大器的音频驱动电路 162

© Levetop Semiconductor Co., Ltd.

表 目 录

表 1-1：各芯片升级方式.....21
表 2-1：串口屏指令集.....36
表 2-2：主控端与 TFT 串口屏协议表.....39
表 2-3：各芯片支持的指令.....44
表 2-4：串口屏接收的指令信息.....47
表 2-5：串口屏反馈的信息.....48
表 3-1：固定图片显示指令.....100
表 3-2：摄像头显示指令.....101
表 3-3：几何绘图功能.....110
表 3-4：起始控制指令.....119
表 3-5：串口屏版本信息.....119
表 3-6：电阻式触控屏控制指令.....130
表 3-7：RTC 时钟控制指令.....131
表 3-8：FAh 影音播放设定指令.....135
表 3-9：使用矢量字库的文字设定指令 - 2.....136
表 3-10：指令总表.....142
表 3-11：各芯片支持压缩图片的指令.....147

© Levetop Semiconductor Co., Ltd.

1. 概述

本应用手册主要在说明 Levetop TFT 图形加速控制器在 TFT 串口屏上的应用，及介绍串口屏开发工具 - 图文 UI 编辑器（UI_Editor.exe）的使用方式，让 TFT 屏厂的制造业者或是系统端的客户能够依据功能需求快速地规划及完成其产品在 TFT 屏的显示应用，避免为了处理 TFT 显示画面进行冗长的程序开发。

1.1 TFT 串口屏的软硬件架构

所谓的 TFT 串口屏就是在 TFT 显示模块上加上 MCU 及 TFT 控制器，该 MCU 负责接收主控端主板送来的串口（Uart）指令，然后依据这些定义好的指令去显示出图片或是动画，主控端主板上的 MCU 不需要为了繁琐的图片显示去编写复杂的程序，因此 TFT 串口屏实际上就是一种指令屏的架构。

TFT 串口屏对主控端主要是透过 Uart (SCI) 接口来通讯，如下图以 LT7689 为例的示意图。果要达到较远距离的通讯效果，通常需要加上 RS232 或是 RS485 的专用驱动芯片，可参考下一章。

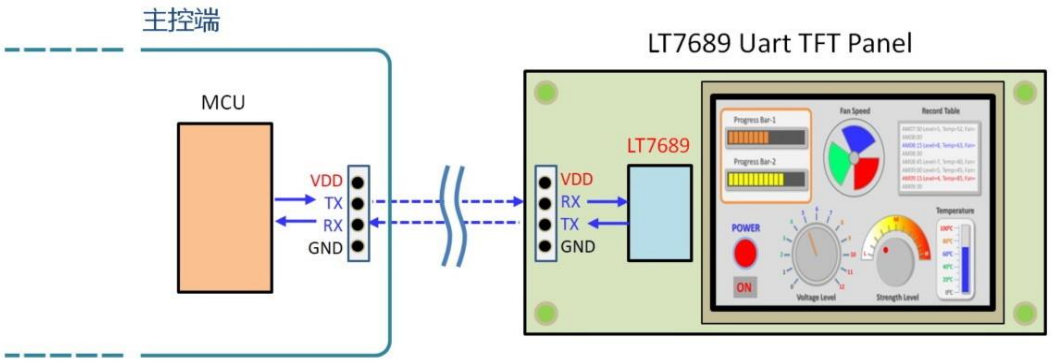


图 1-1：主控端 MCU 的 Uart 与串口屏的 Uart 连接示意图

在使用 Levetop 串口屏方案之前必须要用上位机软件做开发，乐升半导体提供了**图文 UI 编辑器（UI_Editor.exe）**上位机软件，可以单独对 Levetop 的 TFT 串口屏进行设置及显示功能的开发，上位机软件开发时会将使用到的图片、文字、动画等信息产生 Bin 檔，开发者可以透过 SD 卡、USB 碟，或是专用的 SPI Flash 烧录器将 Bin 檔烧录到 SPI Flash 内，然后透过 USB 转 Uart (RS232) 的控制线对 TFT 串口屏进行模拟，也就是做 TFT 屏显示画面的前期验证。

© Levetop Semiconductor Co., Ltd.

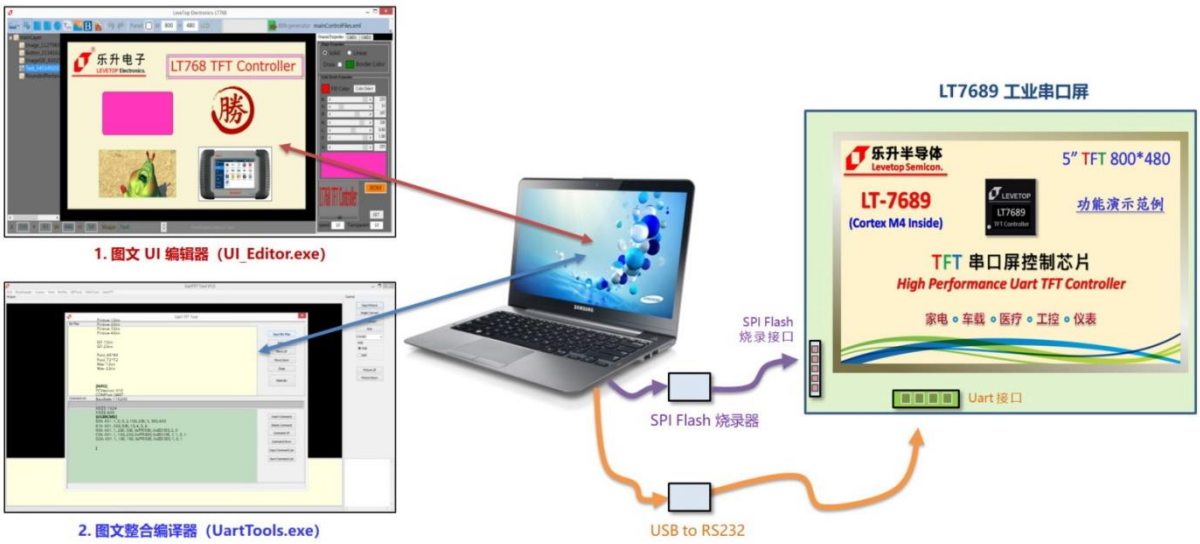


图 1-2：使用上位机软件开发的示意图

上位机软件会依据图片出现的顺序及方式产生指令格式，而前面所说的模拟就是以电脑透过 USB 转 Uart 控制线替代主控端主控发出指令，让开发者在上位机软件上做前期验证，如果上位机软件发出的指令格式都能在 TFT 屏上显示及达到开发者所要的效果，那么主控端最终就在其 MCU 程序上植入这些指令格式，在想要显示图片时送出对应的指令。下图为主控板与 LT7689 串口屏连接的示意图：

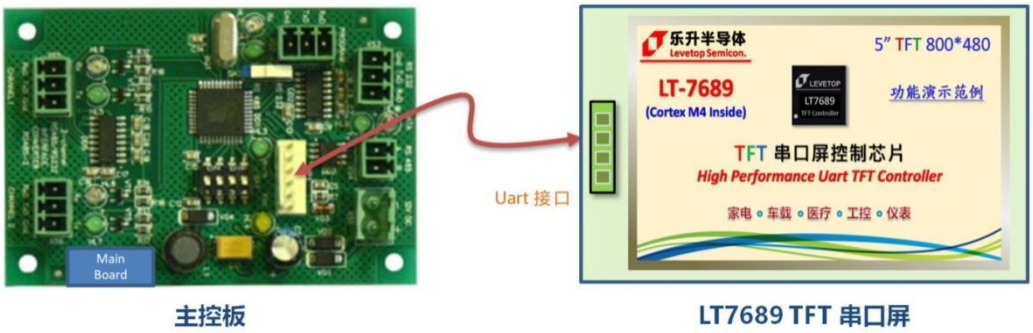


图 1-3：主控板与 TFT 串口屏连接

每个 TFT 串口屏的显示动作都有一个固定的指令，例如 80h 就是显示图片的指令，UI_Editor 会将使用的图片给予编号，在进行编译后会将所有图片、文字、动画等信息产生 Bin 檔，开发者将 Bin 檔烧录到 SPI Flash 内后，于验证的时候当电脑送出 80h、00h 那么 TFT 串口屏就会显示第一张图片，送出 80h、01h 就会显示第二张图片，当 UI_Editor 发出的指令格式都能在 TFT 屏上显示及达到开发者所要的效果，就可以实地将主控端连接到 TFT 串口屏（如上图），而主控端 MCU 程序送出 0xAA(Start) 、80h、00h、1Bh(CRC1)、98h(CRC2) 、0xE4(End1)、0x1B(End2)、0x11(End3)、0xEE(End4) 指令后，TFT 串口屏就会显示第一张图片，同时回传信息 0xAA(Start) 、80h、00h、00h、1Bh(CRC1)、98h(CRC2) ）、0xE4(End1)、0x1B(End2)、0x11(End3)、0xEE(End4) 给主控端，确认整个指令握手协议完成，如下图：

© Levetop Semiconductor Co., Ltd.

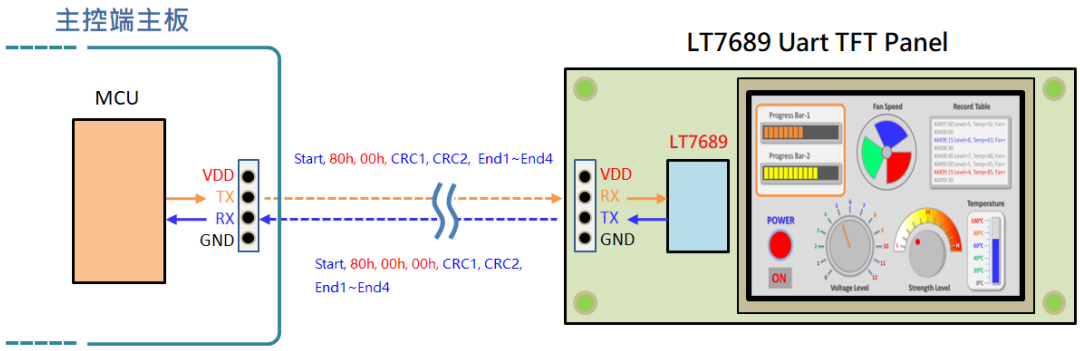


图 1-4：串口屏的指令协议范例一

当主控端 MCU 程序送出 0xAA(Start) 、80h、01h、0Bh(CRC1)、B9h(CRC2) 、0xE4(End1)、0x1B(End2) 、0x11(End3) 、0xEE(End4) 指令后，TFT 串口屏就会显示第二张图片，同时回传信息 0xAA(Start) 、80h、01h、00h、1Bh(CRC1)、98h(CRC2) 、0xE4(End1)、0x1B(End2)、0x11(End3)、0xEE(End4) 给主控端，确认整个握手协议完成，如下图：

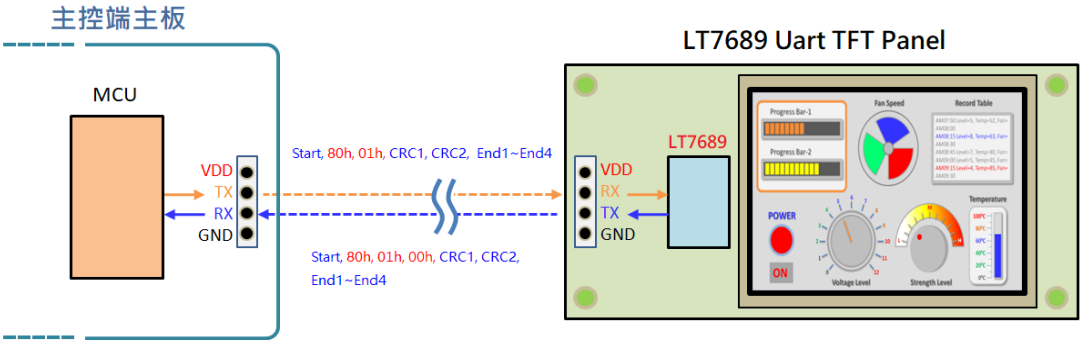


图 1-5：串口屏的指令协议范例二

提示: 为了确保主控端与 TFT 串口屏之间的数据传递正确，主控端 MCU 程序送出的指令还要加上 1 个 Byte 的 **起始码**（固定为 0xAA）、2 个 Byte 的 **CRC 码**、4 个 Byte 的 **结束码**（固定为 0xE4、0x1B、0x11、0xEE），而 TFT 串口屏收到信息或是完成指令后会回传后信息给主控端的 MCU，主控端与串口屏的指令握手协议表请参考手册后面第 2.2 节。有关 2 个字节 CRC 的产生方式请参考手册后面第 2.4 节。

© Levetop Semiconductor Co., Ltd.

1.2 TFT 串口屏的演示板（Demo Kit）

乐升半导体 针对每个串口屏控制芯片都提供演示板（Demo Kit），如下图是 乐升半导体 的 LT7689 TFT 串口屏演示板，可以搭配标准 40Pin 或是 50Pin TFT 屏。



图 1-6：LT7689 串口屏的开发演示板（Demo Kit）

乐升半导体 提供的演示板内都已经含有开发程序，串口屏主板上的 SPI Flash（NOR Flash, 128Mbit）也烧录有演示图片、GIF 动画、字库、Wav 音乐文件等所组合成的 Bin 文件，相关 Demo 文件可以至 乐升半导体 网页（https://www.levetop.cn）的下载专区内下载，或与业务人员联系。

下图是 乐升半导体 的 LT268C TFT 串口屏演示板，搭配了一 480*320（或是作为 320*480 ）的 TFT 屏。



图 1-7：LT268C 串口屏的开发演示板 - 1

© Levetop Semiconductor Co., Ltd.

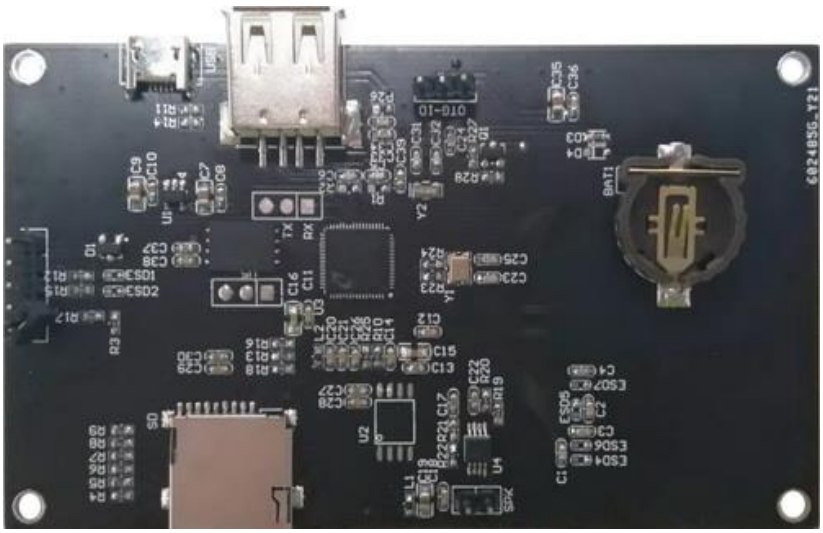


图 1-8：LT268C 串口屏的开发演示板 - 2

© Levetop Semiconductor Co., Ltd.

1.3 串口屏开发流程

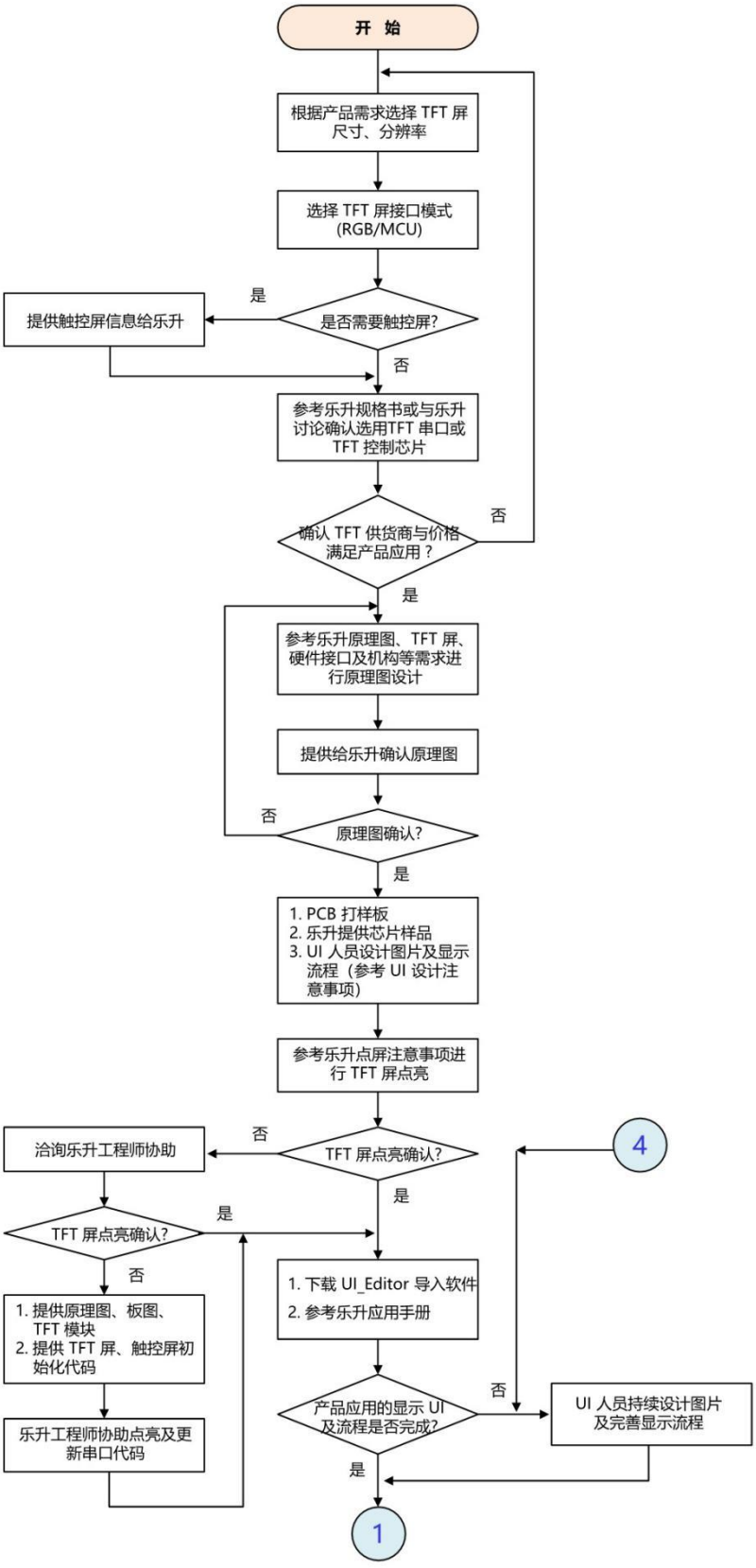


图 1-9：开发流程（1）

© Levetop Semiconductor Co., Ltd.

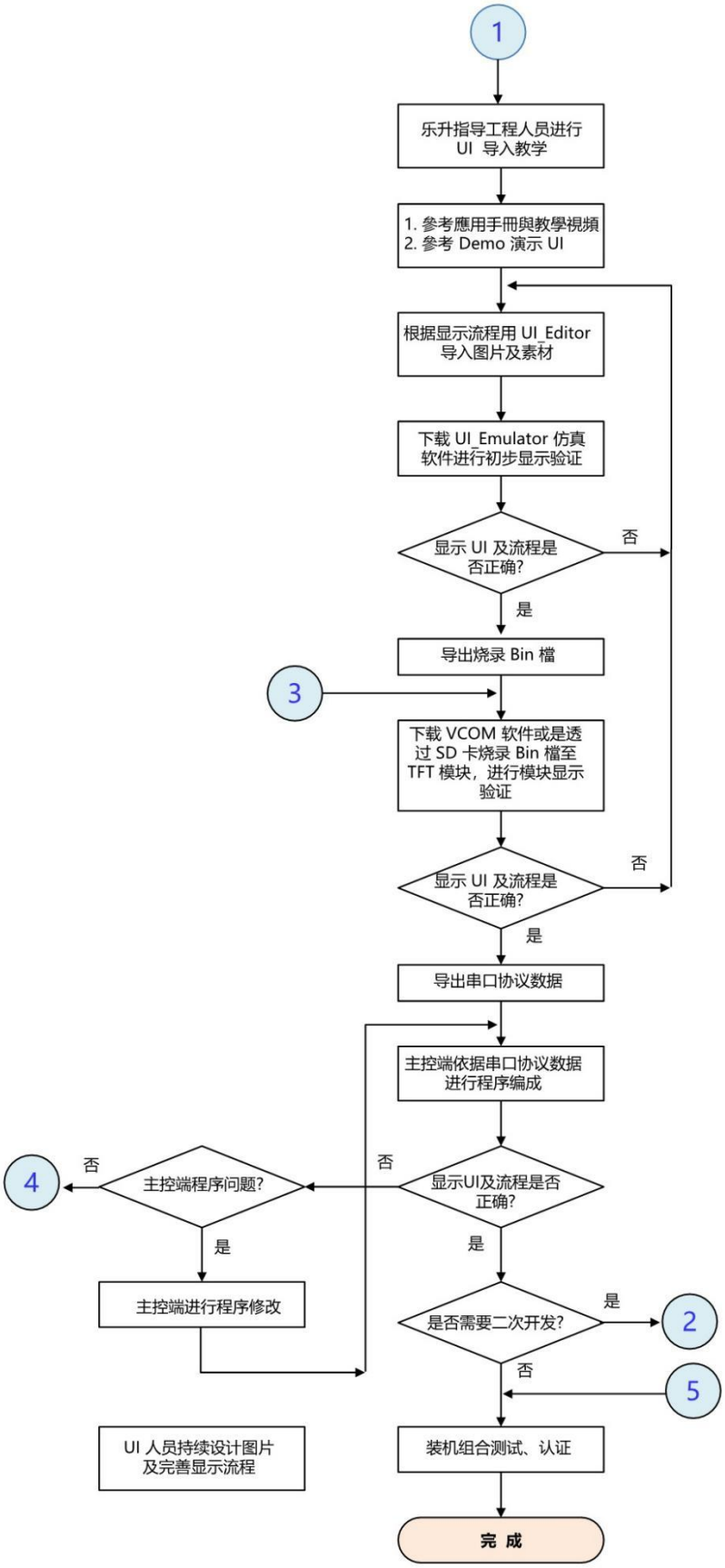


图 1-10：开发流程（2）

© Levetop Semiconductor Co., Ltd.

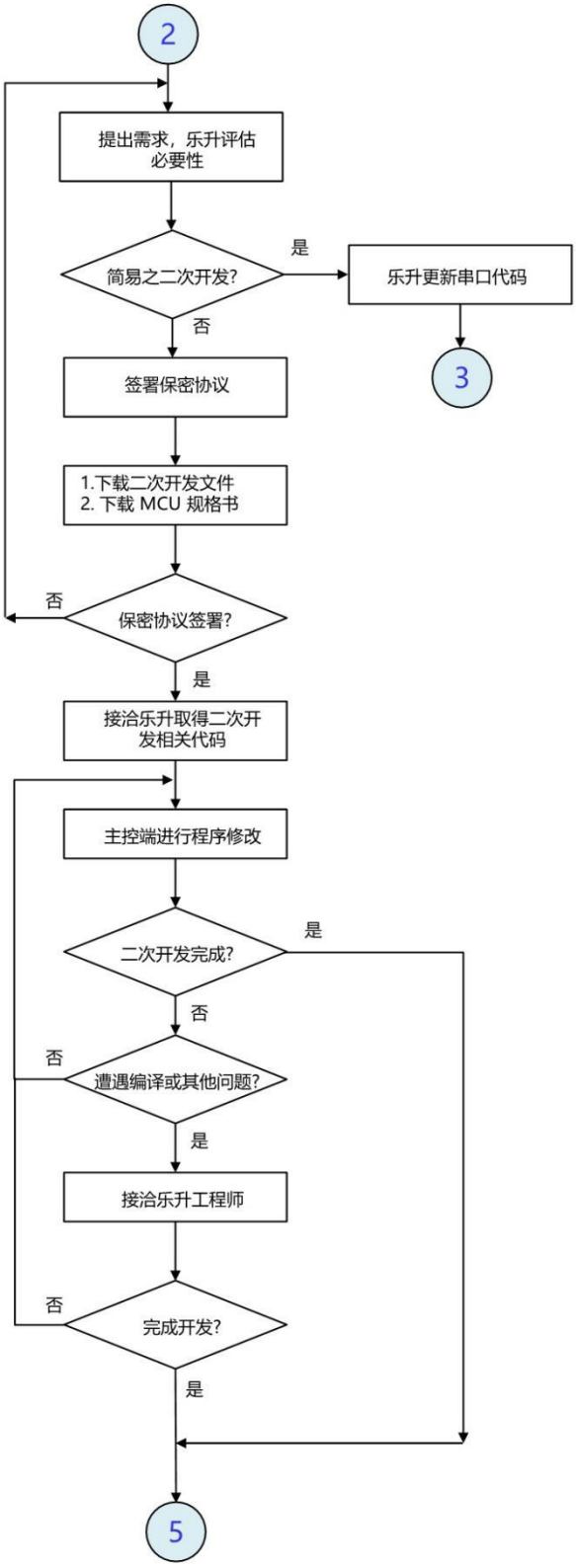


图 1-11：开发流程（3）

© Levetop Semiconductor Co., Ltd.

1.4 TFT 串口屏的升级

一颗新的串口屏控制芯片，需要依次烧录 串口屏程序代码（MCU_Code.bin）、客户应用讯息（UartTFT_Flash.bin）两个 Bin 文件，才可正常工作。其中串口屏程序代码是指控制芯片内部的 MCU 程序，该程序存于控制芯片内部的 Flash 内；客户应用讯息是指客户在显示上需要用到的图片、动画、文字及显示运作流程，该讯息存于外部的 SPI Flash 内。

乐升半导体 每个串口屏控制芯片所支持的升级更新方式不同，请参考下表：

表 1-1：各芯片升级方式

型 号	TFT 接口	升级 Bin 檔	USB 接口升级 (LT_VCOM_GUI)	Uart 串口升级 (LT_Uart_GUI)	SD 卡升级	USB 碟升级
LT7688	RGB	MCU_Code	V	-	-	-
		UartTFT_Flash	V	-	-	-
LT7689	RGB	MCU_Code	-	V	V	V
		UartTFT_Flash	-	V	V	V
LT776	RGB	MCU_Code	-	V	V	-
		UartTFT_Flash	-	V	V	-
LT269	8Bit MCU	MCU_Code	V	V	-	-
		UartTFT_Flash	V	V	V	-
LT268B	8Bit MCU	MCU_Code	V	V	-	-
		UartTFT_Flash	V	V	V	-
LT268C	8/16Bit MCU	MCU_Code	V	V	-	V
		UartTFT_Flash	V	V	V	V
LT268D	8/16Bit MCU	MCU_Code	V	V	-	V
		UartTFT_Flash	V	V	V	V

© Levetop Semiconductor Co., Ltd.

1.4.1 使用 LT_VCOM_GUI 软件更新

其他如 LT7688 及 LT268x 系列可以使用 LT_VCOM_GUI 工具进行更新，可自本公司网站（www.levetop.cn）下载“LT_VCOM_GUI_Vxx”软件。LT_VCOM_GUI 更新程序前需要先将演示板上的“BUSY”引脚拉低，再用 USB 线连接 PCB 板与电脑，如下图所示：



图 1-12：用 LT_VCOM_GUI 更新

以 LT268C 为例：打开电脑软件 LT_VCOM_GUI_Vxx.exe，软件会自动辨识是否连接到 LT268C 及自动获取通讯串口号，如果没有可以尝试点击 Open Comm。（若选择错误的串口号，无法进行下一步操作，防止选错串口号），如下图：

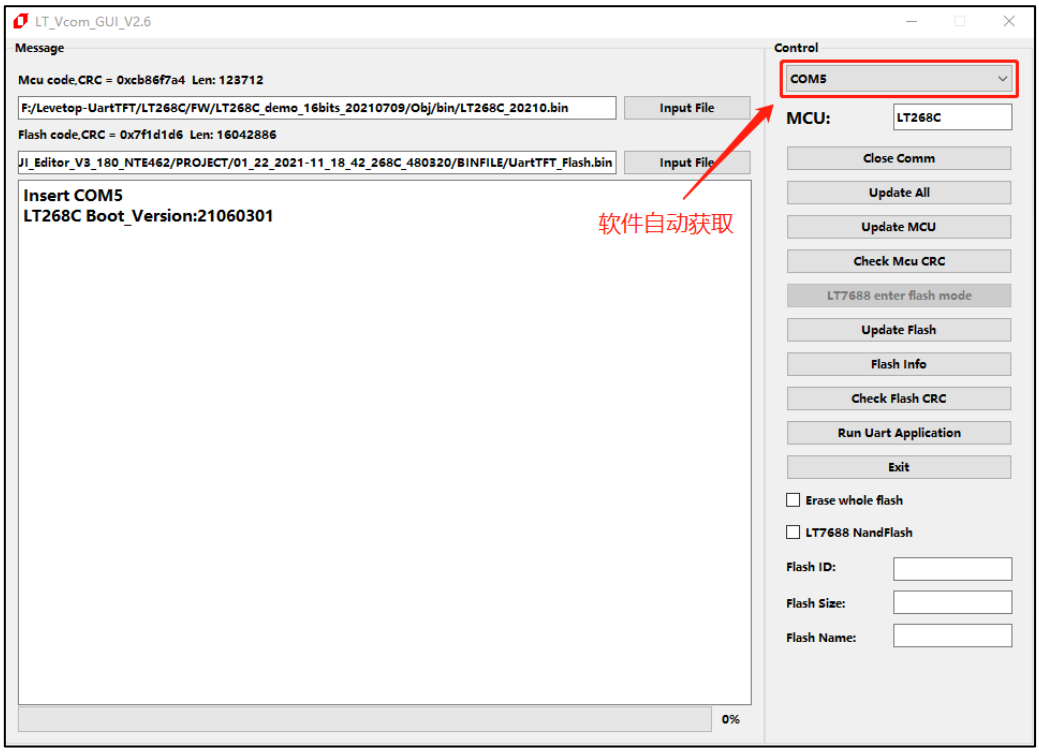


图 1-13：开启软件 LT_VCOM_GUI_Vxx.exe

© Levetop Semiconductor Co., Ltd.

导入需要更新的 MCU，Flash 程序，点击“Update XXX”开启更新，软件界面如下图所示：

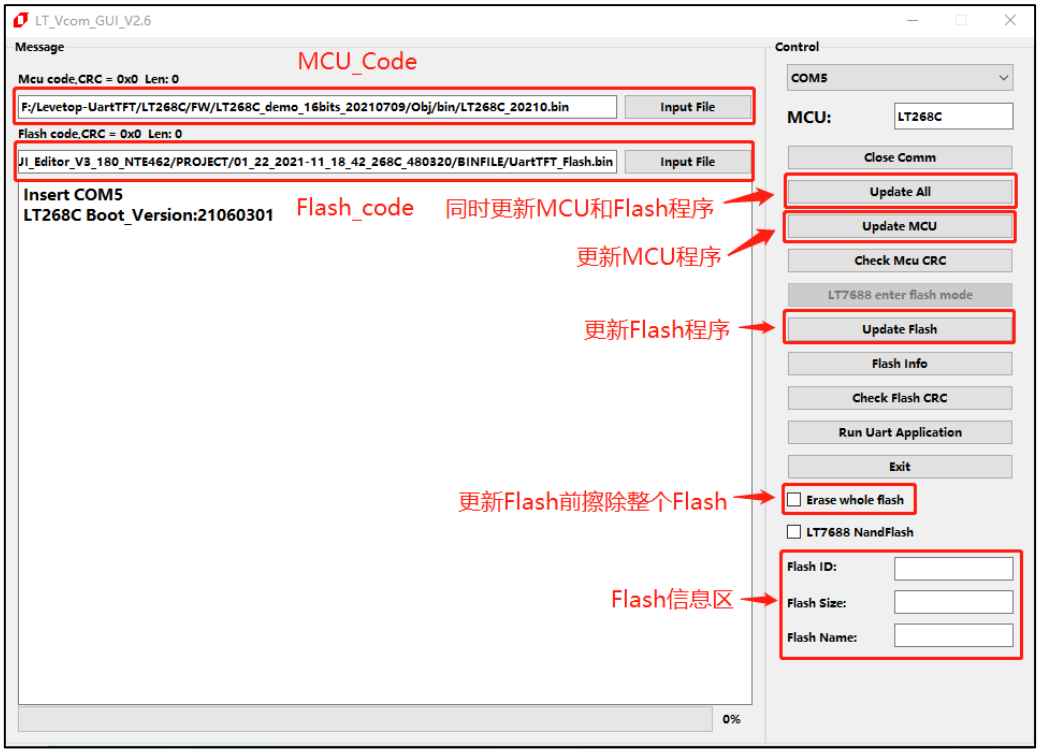


图 1-14：选择更新项目

© Levetop Semiconductor Co., Ltd.

更新完成后可点击“Run Uart Application”进行重置和运行程序，也可重新上电或复位进行重置和运行程序。（注：进行“Run Uart Application”操作时会使 MCU 退出更新模式，使软件不能识别串口，若要重新进入更新模式需按下 RST 按键进行复位。），如下图：

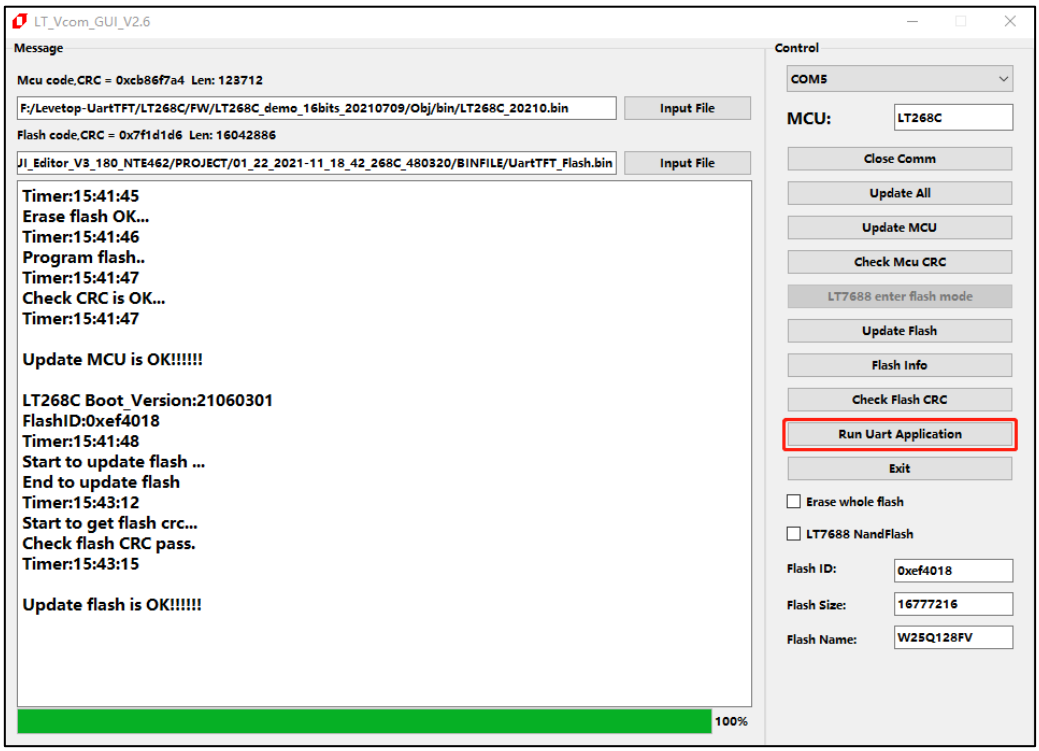


图 1-15：更新完成后进行重置和运行程序

© Levetop Semiconductor Co., Ltd.

1.4.2 使用 LT_Uart_GUI 软件更新

在本公司网站（www.levetop.cn）下载“LT_Uart_GUI_Vxx”软件。通过 LT_Uart_GUI 更新 MCU_Code.bin 或 UartTFT_Flash.bin 时，需要先将演示板上的“BUSY”引脚拉低，再用 USB 转 TTL 线连接 PCB 板上的 TX 和 RX，再用 5V 供电或 USB 线供电，如下图 LT268C 演示板所示：

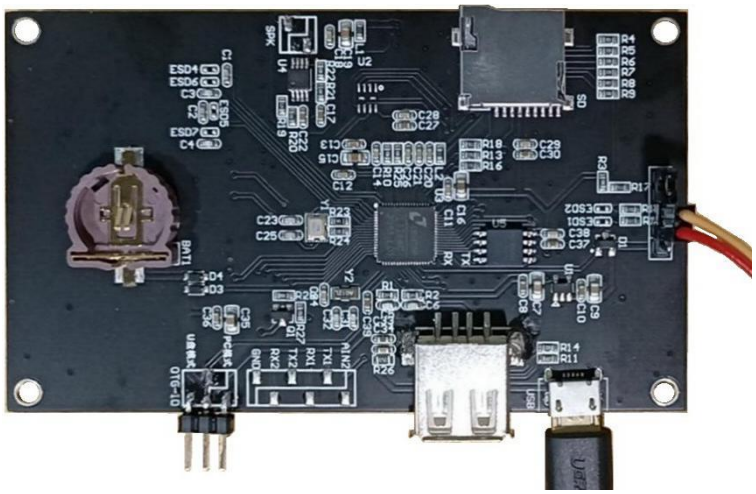


图 1-16：LT268C Uart 升级接线

打开 LT_Uart_GUI 软件，选择 TTL 串口通道，点击“Open Comm”打开串口，会出现 Bootloader 版本号（支持 Uart 更新的 bootloader 版本号是 Version:21060301 或更新版本），导入需要更新的程序，点击“Update MCU”更新 MCU_Code.bin。

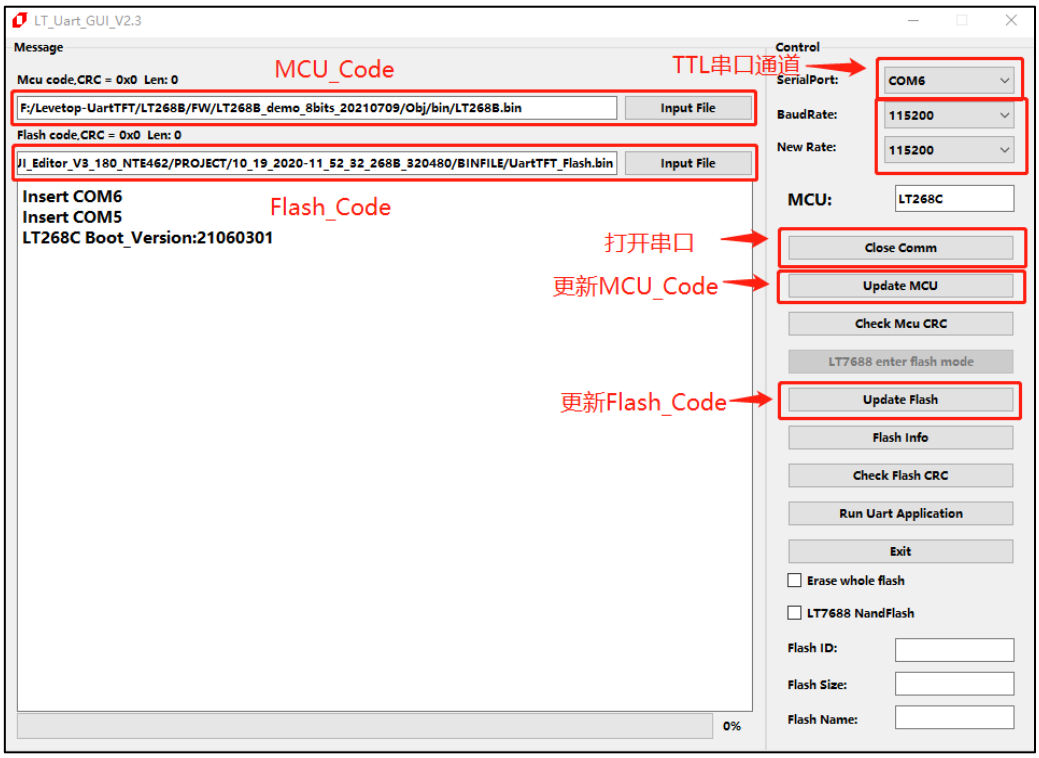


图 1-17：打开 LT_Uart_GUI 软件

© Levetop Semiconductor Co., Ltd.

MCU_Code.bin 更新成功显示如下图，如果不用更新 UartTFT_Flash.bin，点击“Run Uart Application”进行重置和运行程序。

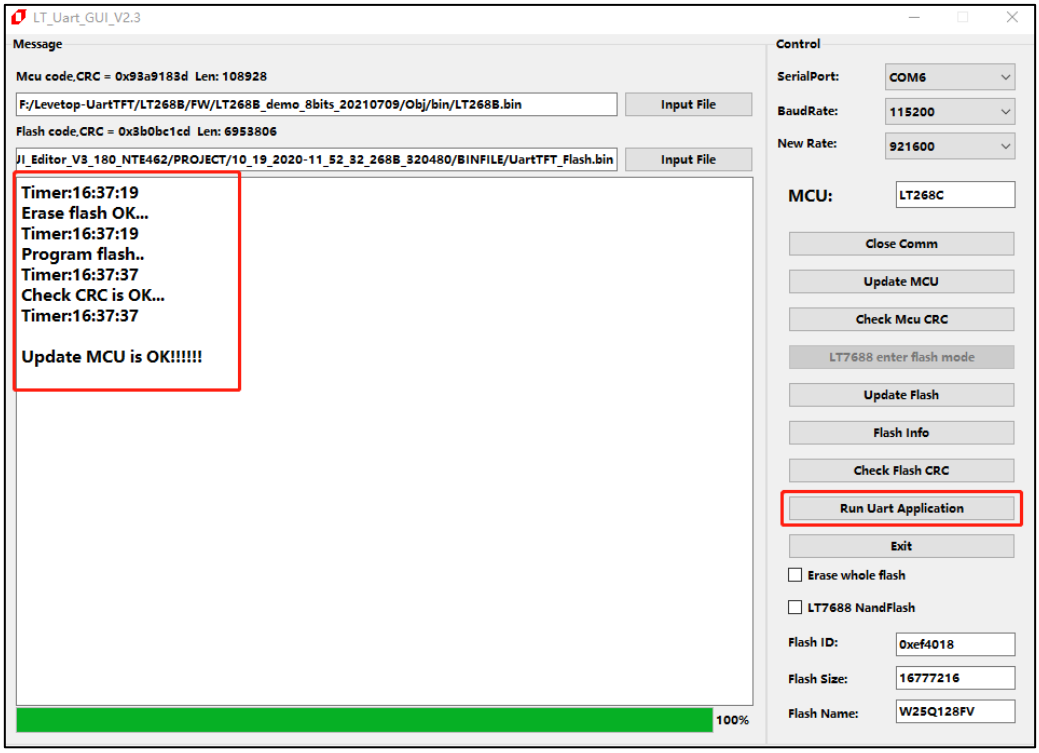


图 1-18：MCU_Code.bin 更新成功

© Levetop Semiconductor Co., Ltd.

如需要更新 UartTFT_Flash.bin，在 Flash code 更新区导入需要更新的 UartTFT_Flash.bin。因默认波特率 115,200bps 更新 Flash 数据太慢，此处建议更换波特率至最快的 921,600bps，方法是在新波特率选项中选择 921,600bps，点击 Close Comm 关闭串口再打开串口，当默认波特率选项自动更新至 921,600bps，则波特率更换成功，如下图所示。点击“Update Flash”开始更新 Flash 程序。

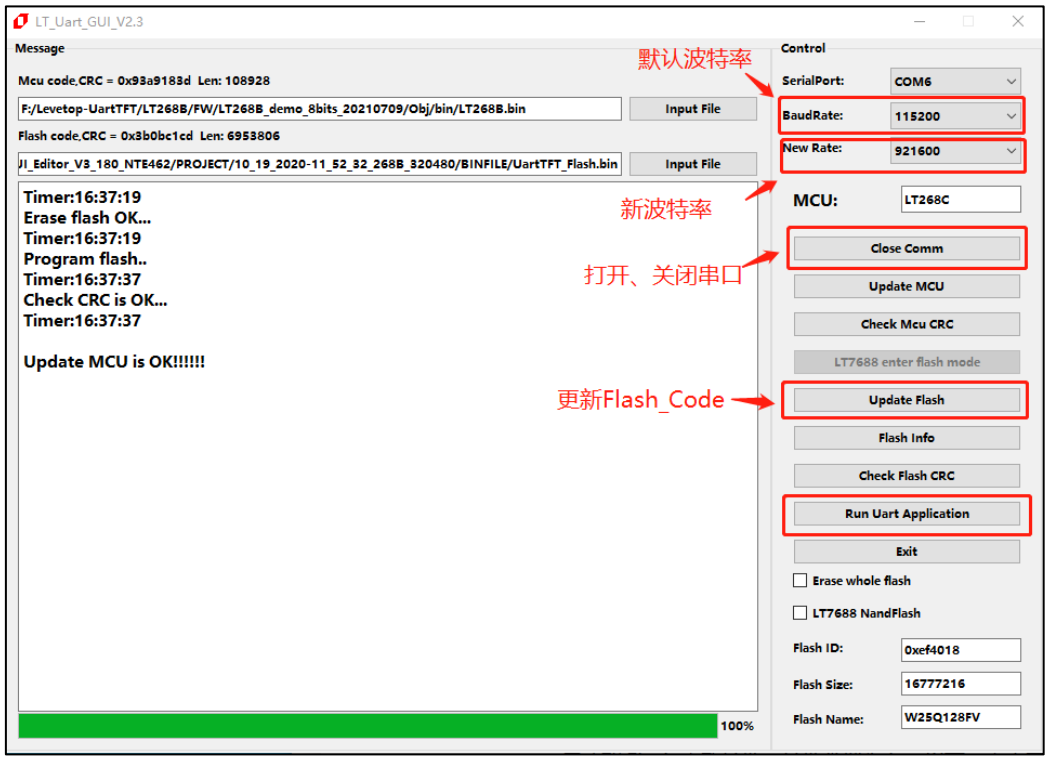


图 1-19：更新 Flash 程序配置

© Levetop Semiconductor Co., Ltd.

UartTFT_Flash.bin 更新成功显示如下图，点击“Run Uart Application”进行重置和运行程序。

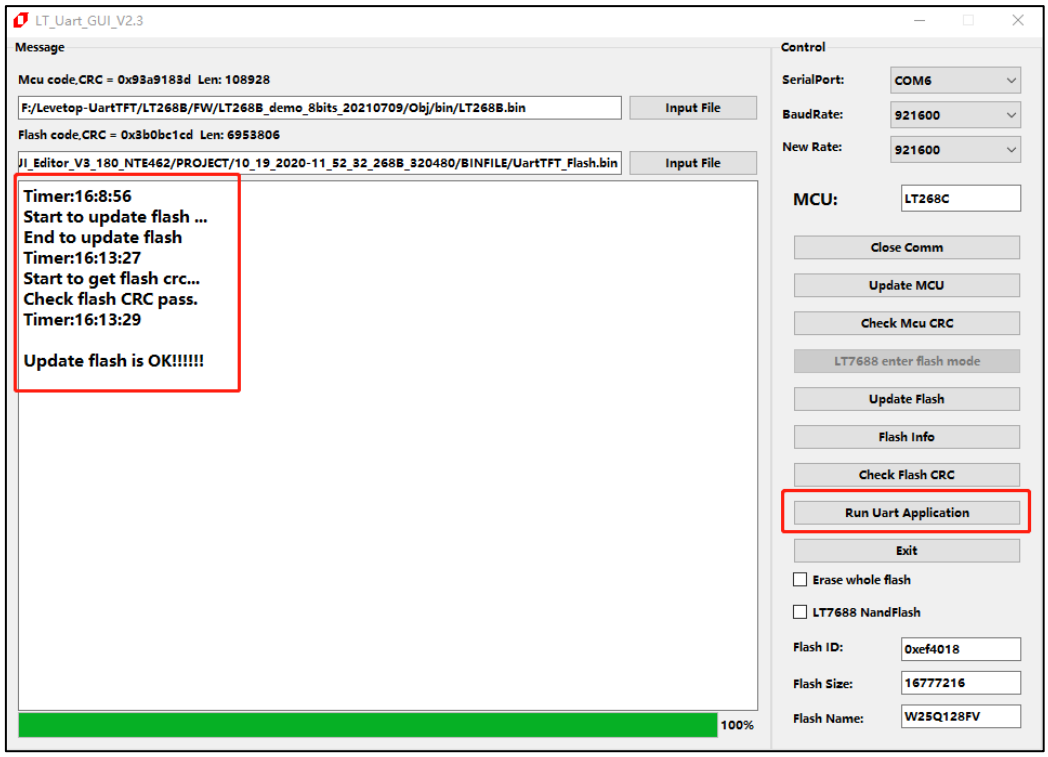


图 1-20：Flash 程序更新完成

目前 LT_VCOM_GUI 或 LT_Uart_GUI 软件已兼容大多数 Winbond 的 NOR Flash，可在软件文件夹中的 Flash.ini 文件中自行添加 Flash 信息和修改 Flash 的片选，在软件的同目录下用记事本的方式打开 Flash.ini 文件，如下图：

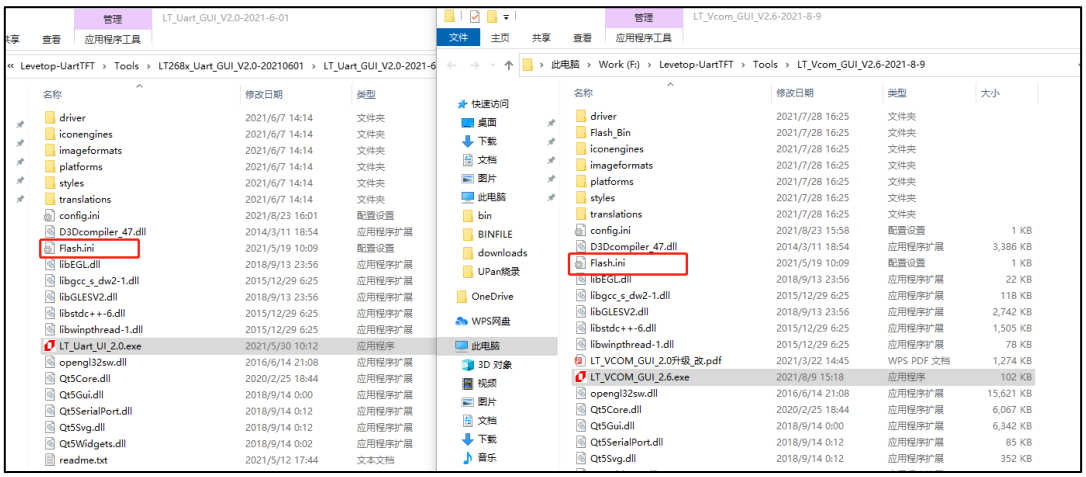


图 1-21：软件文件

© Levetop Semiconductor Co., Ltd.

Flash.ini 文件内容，可以按照格式添加 Flash ID 往后的内存信息，如下图：

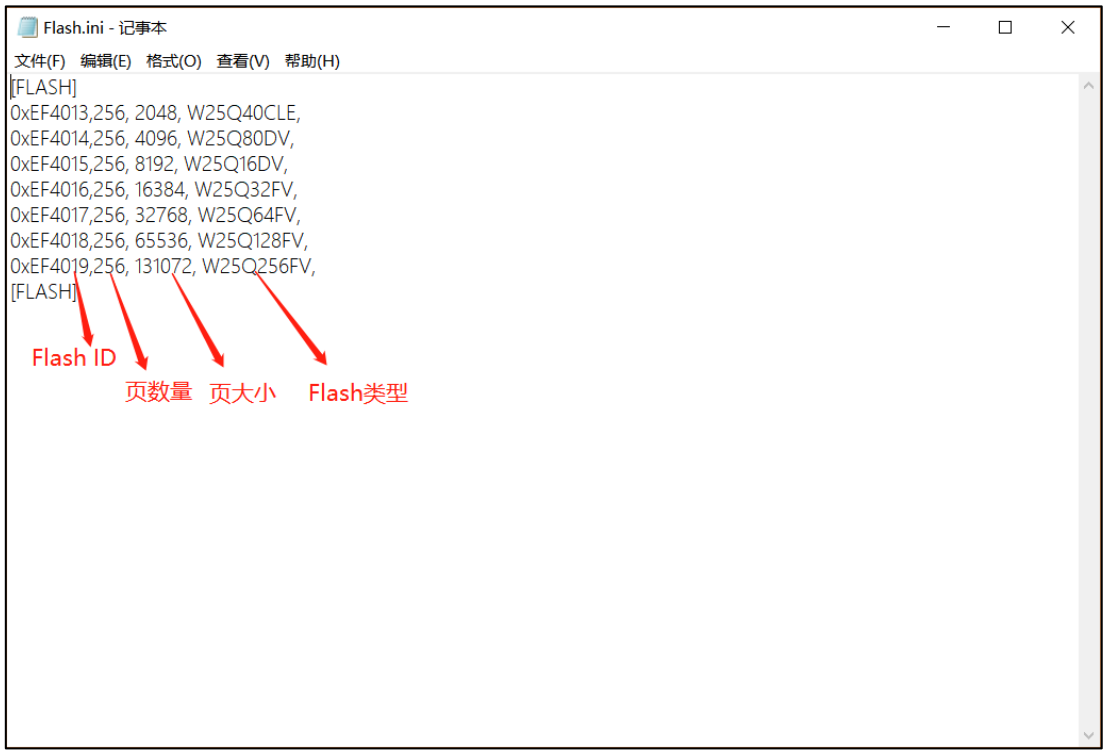


图 1-22：添加 Flash ID

© Levetop Semiconductor Co., Ltd.

1.4.3 用 SD 卡更新 UartTFT_Flash.bin

用 SD 卡更新 Flash 程序，首先将 SD 卡格式化至 FAT32 模式，并在 SD 卡上建立 UartTFT_Flash 子目录，再将要更新的 bin 档案重命名为“UartTFT_Flash.bin”存放在 SD 卡的 UartTFT_Flash 子目录上，给串口屏模组上电，等待显示正常后，将 SD 卡装入模组上的 SD 卡槽内，例如 LT268C 的程序检测到 SD 然会进入更新倒计时画面，如下图：



图 1-23：进入更新倒计时画面

在倒计时 3 秒之后，LT268C 就开始读取 SD 卡上的“UartTFT_Flash.bin”数据，开始进行对 Flash 的烧录动作，如下图所示，烧录结束后 LT268C 会自动重新启动。注意：SD 卡不支持更新 MCU_Code.bin。



图 1-24：UartTFT_Flash.bin 数据更新中

© Levetop Semiconductor Co., Ltd.

1.4.4 用 U 盘更新

通过 U 盘更新 MCU_Code.bin 程序和 UartTFT_Flash.bin。使用 U 盘更新程序需要 U 盘专用的二级 Bootloader（因 U 盘 Bootloader 需要写入屏驱动部分程序，而各家客户使用的屏不同，此项需要定制开发或客户自己修改程序），以 LT268C 为例，通过 LT_VCOM_GUI 软件将 U 盘 Bootloader 烧录到 LT268C 中（需要短接 BUSY 对地），更新完成退出 LT_Vcom_GUI 程序，如下图所示：

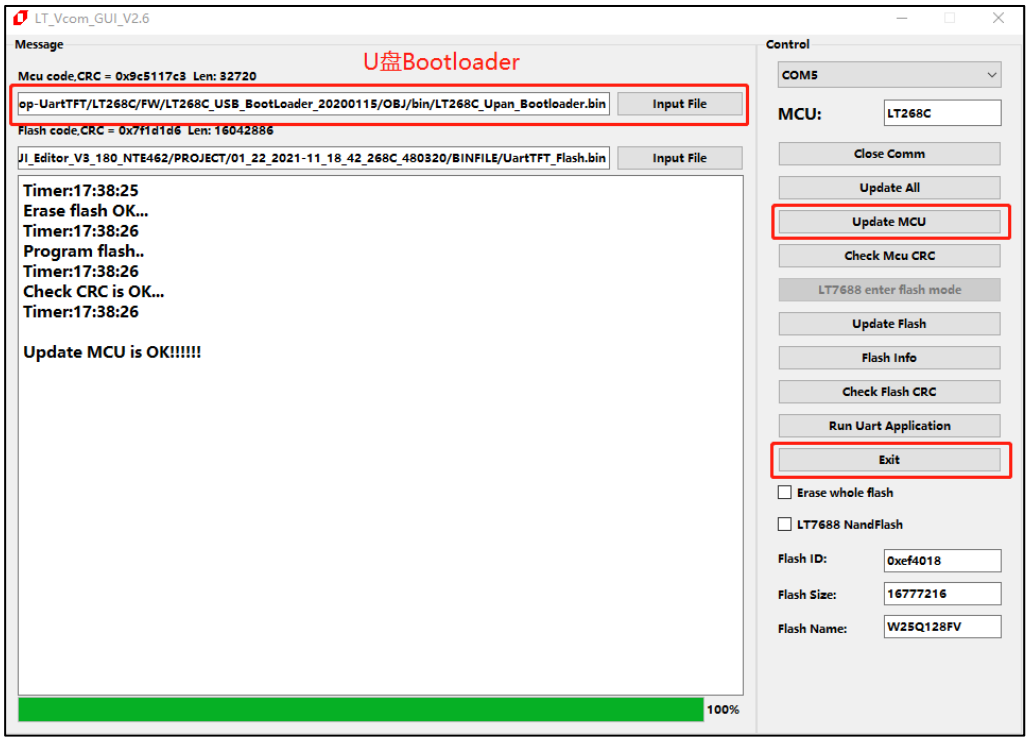


图 1-25：更新 U 盘 Bootloader 程序

U 盘更新方式与 SD 卡相同，首先将 U 盘格式化至 FAT32 模式，并在 U 盘根目录下建立 MCU_Code、UartTFT_Flash、UserInfo 三个文件夹，将需要更新的程序命名为固定的名称（MCU_Code.bin，UartTFT_Flash.bin, UserInfo.bin） 储存到对应的目录里面，如下图所示。不想更新的 Bin 文档不需要储存。

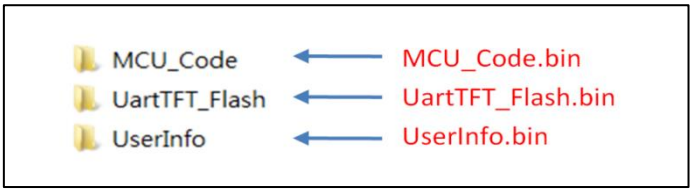


图 1-26：更新的程序名称及对应的文件夹

© Levetop Semiconductor Co., Ltd.

U 盘的文件整理好后，将 U 盘装入 USB 卡槽，短接 OTG 对地（取消短接 BUSY 对地）使用单独 5V 供电（不能直接用 Micro USB 接口来供电），如下图所示：

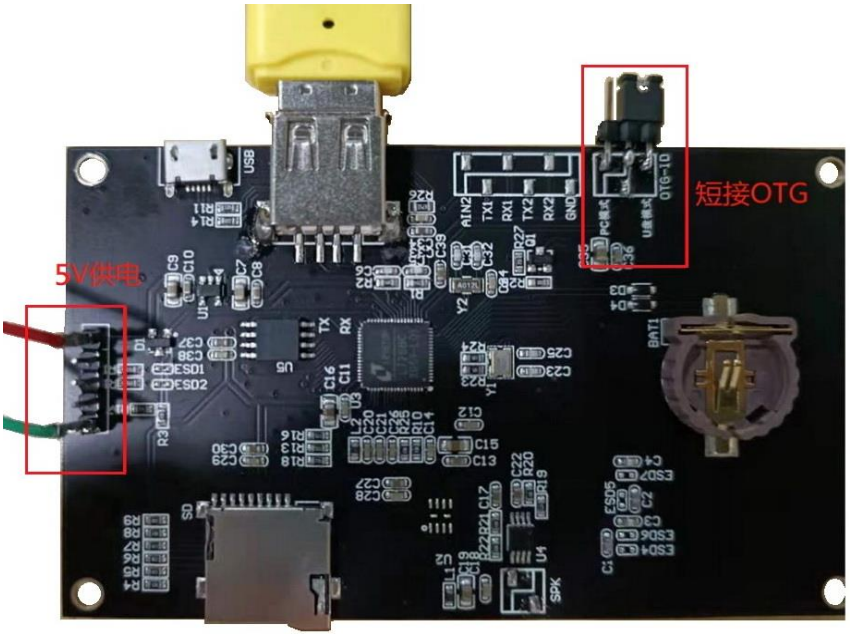


图 1-27：以 SD Card 进行更新中 U 盘更新硬件配置

给 LT268C 模组重新上电，此时串口屏会自动检测及 TFT 显示屏进入更新显示画面，在更新完成后取出 U 盘，然后重新启动 TFT 串口屏即可，如下图所示为升级界面。



图 1-28：U 盘更新程序界面

© Levetop Semiconductor Co., Ltd.

注意 1：如果是更新 UartTFT_Flash.bin，那么通常 SPI Flash 会进行较长时间的擦除及写入动作，因为这是 SPI Flash 本身的特性，请耐心等待。而更新完后 LT268C 内的 MCU 将自动启动，但是建议使用者关机后取出 SD 卡/U 盘，避免下次开机又进入自动更新状态。

注意 2：在 U 盘或 SD 卡格式化时，建议使用快速格式化，分配单元大小选择默认配置，如下图所示：

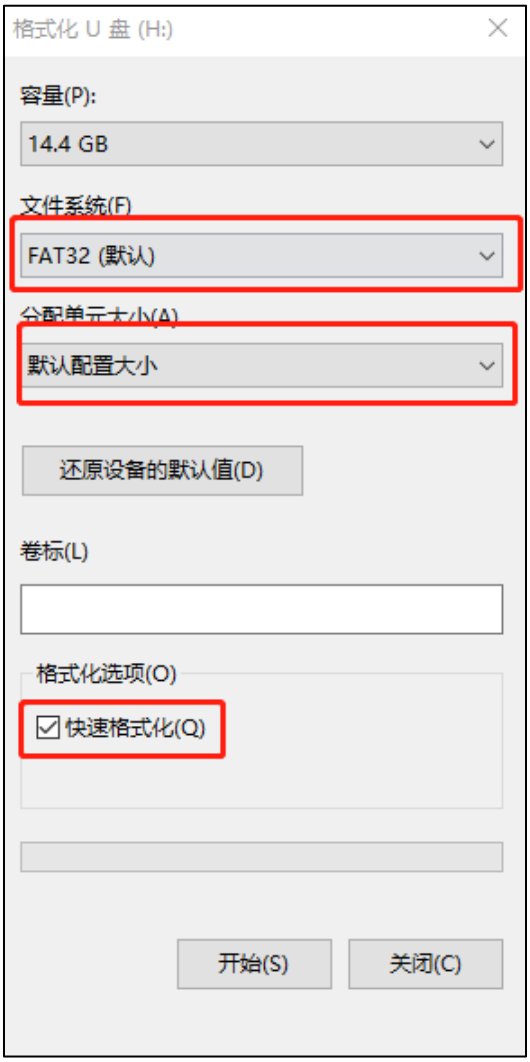


图 1-29：U 盘/SD 卡格式化

© Levetop Semiconductor Co., Ltd.

1.4.5 LT7689 的升级与更新

以 LT7689 为例,透过 USB 盘进行更新 LT7689 的 MCU 程序档案(MCU_Code.bin),或是 SPI Flash 的 Bin 檔 (UartTFT_Flash.bin) 的速度比 SD 卡还要快，更新方式与 SD 卡相同，将需要更新的档案储存在 USB 盘及对应的目录内，将 USB 盘插入 PCB 下方的 USB Disk 卡槽内后对串口屏进行开机或是复位即可自动进入更新模式。

先将 SD 卡/U 盘以 FAT32 方式格式化，然后将更新的 MCU 程序档案（MCU_Code.bin），或是 SPI Flash 的 Bin 檔 (UartTFT_Flash.bin, Userinfo.bin) 透过电脑写入到标准 SD 卡中固定的目录内 (参考前面**图 1-26**），当 TFT 串口屏为关机状态时再将 SD Card/U 盘插入 SD 卡槽/U 盘槽内，注意如果是 U 盘烧录需要短接 OTG 与 GND。

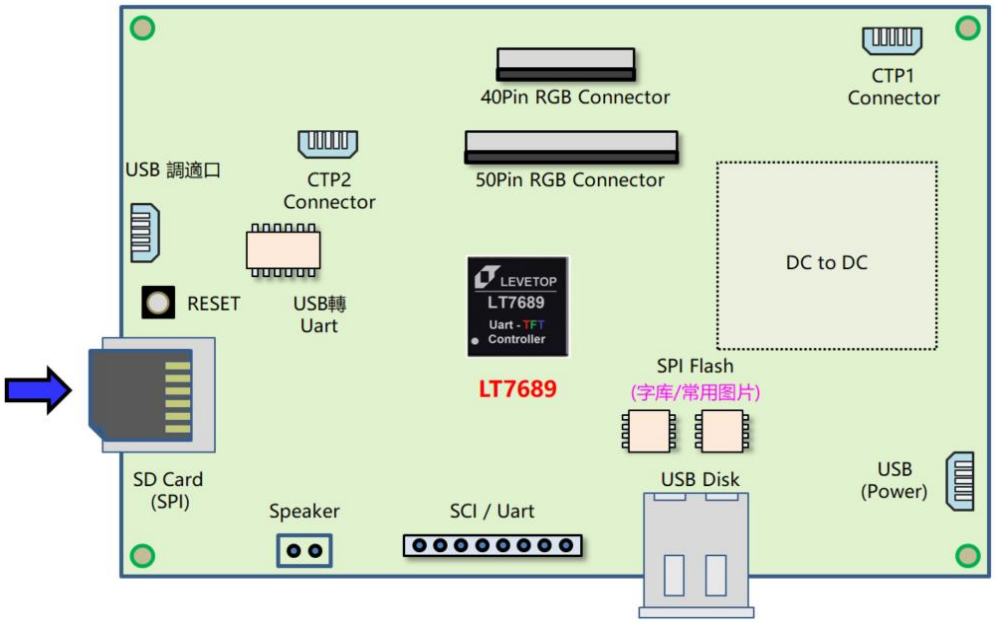


图 1-30：SD Card 插入 SD 卡槽内

© Levetop Semiconductor Co., Ltd.

SD Card 插入 SD 卡槽内后对串口屏进行开机，此时串口屏会自动检测及 TFT 显示屏进入更新显示画面，在更新完成后取出 SD Card，然后重新启动 TFT 串口屏即可，如下图所示为升级 MCU 程序码。

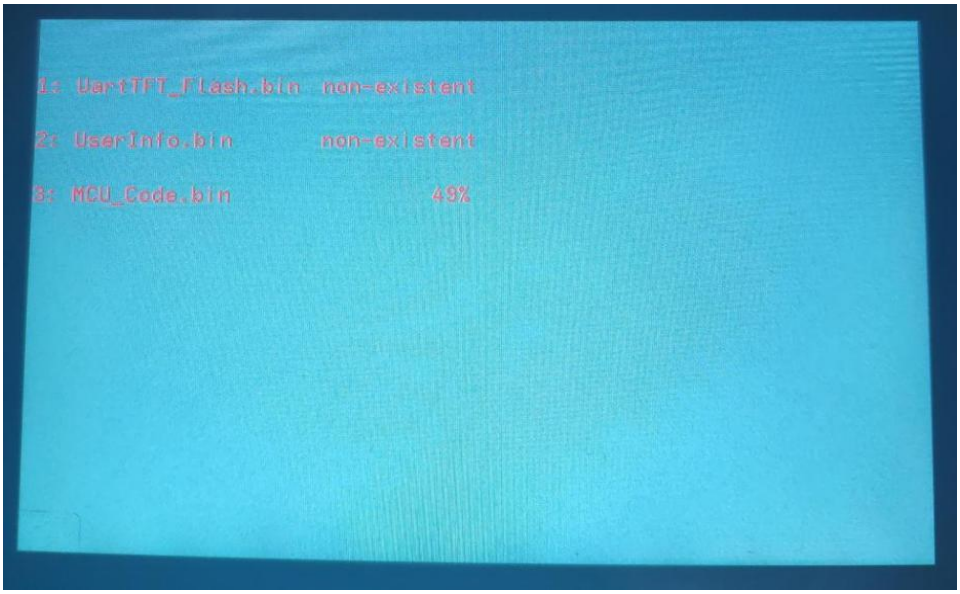


图 1-31：以 SD Card 进行更新中

更新的 MCU 程序档案（MCU_Code.bin），或是 SPI Flash 的 Bin 檔（UartTFT_Flash.bin, UserInfo.bin） 必须储存在固定的目录里面，也就是先在 SD 卡/U 盘内建立 MCU_Code、UartTFT_Flash、UserInfo 三个子目录，然后将要更改的档案储存在该目录里面，如下图所示。不想更新的 Bin 檔不要储存，当 SD Card/U 盘插入 SD 卡槽/U 盘槽内后对串口屏进行开机，串口屏会自动检测存在的档案然后自动进行更新。

注意：更新 Bin 档期间不得关闭电源或是做复位动作，否则将可能造成串口屏 MCU 程序失效及无法再度更新。如果使用 SD 卡/U 盘无法更新或是不会进入**图 1-31** 的更新画面，请与本公司联系。

© Levetop Semiconductor Co., Ltd.

2. 串口指令

为了让主控端的系统或是主板就能够透过 UART、SPI 等串口轻易地在 TFT 屏上显示图片或是信息，在 Levetop TFT 串口屏上规划了一套串口指令集，透过定义好的指令码配合指令参数去改变 TFT 屏上的画面，乐升半导体提供串口屏的开发工具：图文 UI 编辑器（UI_Editor.exe），用户可以进行 TFT 显示屏的方案开发，同时这开发工具可以实时模拟 TFT 显示屏的显示效果，做前期的验证。

2.1 串口屏指令集

表 2-1：串口屏指令集

主功能	细项功能	指令码
显示图片	单张/多张图片	80h, 85h, 8Ah, 8Fh,
	循环播放	81h, 84h
	显示透明图片	82h
	GIF 动画	88h, 89h
	设定显示缓冲区	8Eh
	弹出图片	D8h
	循环卷动	D9h, DBh
	数字图片	90h, 91h
	显示 ASCII 码图片	9Fh
显示控件图片	控件滑条	94h, 95h
	环形触控滑条	96h, 97h
	显示单一控件图片	A0h, A1h
	虚拟控件	A2h, A3h
	全屏滑动图片	B4h
	手势滑动	C9h
	取消控件触控功能	9Eh
	显示底图及控件图片	9Bh, 9Ch
指标与造图	进度条指标图	B0h
	指针指标图	B1h
	环形指标图	DCh
	二维码生成	98h
	曲线显示	B2,B3h
显示字库	字库 - 1~4	C0h, C1h, C2h, C3h
	大字库 - 1~4	D0h, D1h, D2h, D3h
	矢量字库	FCh, FDh
背光亮度	设置亮度	BAh
	On/Off	BCh

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	指令码
设定时钟	设定时钟	8Ch
	读取时钟	8Dh
显示时钟/星期	显示数字时钟	92h
	显示模拟时钟	93h
	显示星期	9Dh
摄像头	显示摄像头画面	AEh, AFh
开机动作与画面	执行开机指令	9Ah/00
指令合并	执行组合指令	9Ah
Wav 音乐文件	播放 Wav 檔	B8h
	停止播放	B9h
几何图形	画点	DFh
	直线	E0h
	空心圆形	E1h
	实心圆形	E2h
	带框实心圆形	E3h
	空心椭圆	E4h
	实心椭圆形	E5h
	带框实心椭圆	E6h
	空心矩形	E7h
	实心矩形	E8h
	带框矩形	E9h
	空心圆角矩形	EAh
	实心圆角矩形	EBh
	带框圆角矩形	ECh
	空心三角形	EDh
	实心三角形	EEh
	带框三角形	EFh
	空心四边形	F0h
	实心四边形	F1h
	空心五边形	F2h
	实心五边形	F3h
	圆柱体	F4h
	方柱体	F5h
	表格视窗	F6h
数字键盘	数字键盘输入	A4h
	取消数字键盘	A5h
寄存器控制	寄存器控制指令	CAh~CFh

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	指令码
全功能键盘	全键盘输入	A6h
	取消全键盘	A7h
电阻屏校验	电阻屏校验指令	8Bh
TFT 屏复位	对串口屏芯片复位	BDh
串口屏侦测	联机检查	BEh
	版本侦测	BFh

© Levetop Semiconductor Co., Ltd.

2.2 主控端与 TFT 串口屏协议表

表 2-2：主控端与 TFT 串口屏协议表

主功能	细项功能	主控端发送 (TFT 串口屏接收)						主控端接收 (TFT 串口屏发送)					
		起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	指令参数	CRC 码 (2Bytes)	结束码 (4Bytes)	起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	信息码/反馈码 (1Bytes)	CRC 码 (2Bytes)	结束码 (4Bytes)
显示图片	显示单张/多张图片	Start	80h	nn		CRC	End	Start	80h	nn	信息码	CRC	End
	取消显示单张/多张图片	Start	85h	nn		CRC	End	Start	84h	nn	信息码	CRC	End
	显示单张/多张图片	Start	8Ah	nn		CRC	End	Start	8Ah	nn	信息码	CRC	End
	显示单张图片	Start	8Fh	nn	X, Y, PNG, Pnn	CRC	End	Start	8Fh	nn	信息码	CRC	End
	循环播放	Start	81h	nn		CRC	End	Start	81h	nn	信息码	CRC	End
	取消循环播放	Start	84h	nn		CRC	End	Start	84h	nn	信息码	CRC	End
	透明图片	Start	82h	nn		CRC	End	Start	82h	nn	信息码	CRC	End
	显示GIF动画	Start	88h	nn		CRC	End	Start	88h	nn	信息码	CRC	End
	取消GIF动画	Start	89h	nn		CRC	End	Start	89h	nn	信息码	CRC	End
	设定缓冲区	Start	8Eh		0, 1	CRC	End	Start	8Eh	00	信息码	CRC	End
	弹出图片	Start	D8h	nn		CRC	End	Start	D8h	nn	信息码	CRC	End
	循环卷动	Start	D9h	nn		CRC	End	Start	D9h	nn	信息码	CRC	End
	取消循环卷动	Start	DBh	nn		CRC	End	Start	DBh	nn	信息码	CRC	End
	数字图片	Start	90h	nn	ddd.d	CRC	End	Start	90h	nn	信息码	CRC	End
	真彩数字图片	Start	91h	nn	ddd.d	CRC	End	Start	91h	nn	信息码	CRC	End
	显示ASCII码图片	Start	9Fh	nn	ddd.d	CRC	End	Start	9Fh	nn	信息码	CRC	End
显示控件图片	全屏滑动图片	Start	B4h	nn		CRC	End	Start	B4h	nn	信息码	CRC	End
	显示单一控件图片	Start	A0h	nn		CRC	End	Start	A0h	nn	信息码	CRC	End
		按下控件图片时						Start	A0h	nn	31h	CRC	End
		放开控件图片时						Start	A0h	nn	30h	CRC	End
	取消单一控件图片	Start	A1h	nn		CRC	End	Start	A1h	nn	信息码	CRC	End
	虚拟控件	Start	A2h	nn		CRC	End	Start	A2h	nn	信息码	CRC	End
		按下控件区域时						Start	A2h	nn	31h	CRC	End
		放开控件区域时						Start	A2h	nn	30h	CRC	End
	取消虚拟控件	Start	A3h	nn		CRC	End	Start	A3h	nn	信息码	CRC	End
	手势滑动	Start	C9h	nn		CRC	End	Start	C9h	nn	Dir+信息码	CRC	End
	取消控件触控功能	Start	9Eh		Code (1Byte)	CRC	End	Start	9Eh	Code (1Byte)	信息码	CRC	End

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	主控端发送 (TFT串口屏接收) 起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	指令参数	CRC码 (2Bytes)	结束码 (4Bytes)	主控端接收 (TFT串口屏发送) 起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	信息码/反馈码 (1Bytes)	CRC码 (2Bytes)	结束码 (4Bytes)
	显示底图及所有控件图片	Start	9Ch	00		CRC	End	Start	9Ch	00	信息码	CRC	End
		屏幕滑动后						Start	9Ch	页号	信息码	CRC	Start
		按下控件图片时						Start	9Bh	图标ID号	31h	CRC	End
		放开控件图片时						Start	9Bh	图标ID号	30h	CRC	End
指标与造图	进度条指标图	Start	B0h	nn	Value (2 Bytes)	CRC	End	Start	B0h	nn	信息码	CRC	End
	指针指标图	Start	B1h	nn	Angle (2 Bytes)	CRC	End	Start	B1h	nn	信息码	CRC	End
	环形指标图	Start	DCh	nn	S_Angle, A_Angle	CRC	End	Start	DCh	nn	信息码	CRC	End
	二维码生成	Start	98h	nn	字符串	CRC	End	Start	98h	nn	信息码	CRC	End
	曲线缓冲	Start	B2h	Chanel (0~3)	Data (n*2bytes)	CRC	End	Start	B2h	Chanel (0~3)	信息码	CRC	End
	显示曲线	Start	B3h	nn		CRC	End	Start	B3h	nn	信息码	CRC	End
触控滑条控制	设置触控滑条	Start	94h	nn		CRC	End	Start	94h	nn	信息码	CRC	End
		触控滑条被按下时						Start	94h	nn	Value (1 Byte)	CRC	End
	移除触控滑条	Start	95h	nn		CRC	End	Start	95h	nn	信息码	CRC	End
	设置环形触控滑条	Start	96h	nn		CRC	End	Start	96h	nn	信息码	CRC	End
		环形触控滑条被按下时						Start	96h	nn	Value (1 Byte)	CRC	End
	移除环形触控滑条	Start	97h	nn		CRC	End	Start	97h	nn	信息码	CRC	End
显示字串	字库 - 1	Start	C0h	nn	字符串	CRC	End	Start	C0h	nn	信息码	CRC	End
	字库 - 2	Start	C1h	nn	字符串	CRC	End	Start	C1h	nn	信息码	CRC	End
	字库 - 3	Start	C2h	nn	字符串	CRC	End	Start	C2h	nn	信息码	CRC	End
	字库 - 4	Start	C3h	nn	字符串	CRC	End	Start	C3h	nn	信息码	CRC	End
	大字库 - 1	Start	D0h	nn	字符串	CRC	End	Start	D0h	nn	信息码	CRC	End
	大字库 - 2	Start	D1h	nn	字符串	CRC	End	Start	D1h	nn	信息码	CRC	End
	大字库 - 3	Start	D2h	nn	字符串	CRC	End	Start	D2h	nn	信息码	CRC	End
	大字库 - 4	Start	D3h	nn	字符串	CRC	End	Start	D3h	nn	信息码	CRC	End
	矢量字库	Start	FCh	nn	字符串	CRC	End	Start	FCh	nn	信息码	CRC	End
		Start	FDh		Name,X0, Y0,X1,Y1, Color,Color-B,Size, H_Ali,V_Ali,Tr, String	CRC	End	Start	FDh	nn	信息码	CRC	End

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	主控端发送 (TFT 串口屏接收)						主控端接收 (TFT 串口屏发送)					
		起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	指令参数	CRC 码 (2Bytes)	结束码 (4Bytes)	起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	信息码/反馈码 (1Bytes)	CRC 码 (2Bytes)	结束码 (4Bytes)
背光亮度	设置亮度	Start	BAh		BL (00~0Fh)	CRC	End	Start	BAh	BL (00~0Fh)	信息码	CRC	End
	On/Off	Start	BCh		00,01,02	CRC	End	Start	BCh	00,01,02	信息码	CRC	End
Wav 檔	播放	Start	B8h		REP(Bit7) + WAV 编号	CRC	End	Start	B8h	REP(Bit7) + WAV 编号	信息码	CRC	End
	停止	Start	B9h			CRC	End	Start	B9h	00	信息码	CRC	End
开机指令	开机指令	Start	9Ah	00		CRC	End	Start	9Ah	00	信息码	CRC	End
合并指令	执行合并指令	Start	9Ah	nn		CRC	End	Start	9Ah	nn	信息码	CRC	End
阻屏校验	电阻屏校验	Start	8Bh			CRC	End	Start	8Bh	00	信息码	CRC	End
TFT 屏复位	对 TFT 屏进行复位	Start	BDh			CRC	End	Start	BDh	00	信息码	CRC	End
几何图形	画点	Start	DFh	nn	X,Y	CRC	End	Start	DFh	nn	信息码	CRC	End
	直线	Start	E0h	nn		CRC	End	Start	E0h	nn	信息码	CRC	End
	空心圆形	Start	E1h	nn		CRC	End	Start	E1h	nn	信息码	CRC	End
	实心圆形	Start	E2h	nn		CRC	End	Start	E2h	nn	信息码	CRC	End
	带框实心圆形	Start	E3h	nn		CRC	End	Start	E3h	nn	信息码	CRC	End
	空心椭圆	Start	E4h	nn		CRC	End	Start	E4h	nn	信息码	CRC	End
	实心椭圆形	Start	E5h	nn		CRC	End	Start	E5h	nn	信息码	CRC	End
	带框实心椭圆	Start	E6h	nn		CRC	End	Start	E6h	nn	信息码	CRC	End
	空心矩形	Start	E7h	nn		CRC	End	Start	E7h	nn	信息码	CRC	End
	实心矩形	Start	E8h	nn		CRC	End	Start	E8h	nn	信息码	CRC	End
	带框矩形	Start	E9h	nn		CRC	End	Start	E9h	nn	信息码	CRC	End
	空心圆角矩形	Start	EAh	nn		CRC	End	Start	EAh	nn	信息码	CRC	End
	实心圆角矩形	Start	EBh	nn		CRC	End	Start	EBh	nn	信息码	CRC	End
	带框圆角矩形	Start	ECh	nn		CRC	End	Start	ECh	nn	信息码	CRC	End
	空心三角形	Start	EDh	nn		CRC	End	Start	EDh	nn	信息码	CRC	End
	实心三角形	Start	EEh	nn		CRC	End	Start	EEh	nn	信息码	CRC	End
	带框三角形	Start	EFh	nn		CRC	End	Start	EFh	nn	信息码	CRC	End
	空心四边形	Start	F0h	nn		CRC	End	Start	F0h	nn	信息码	CRC	End
	实心四边形	Start	F1h	nn		CRC	End	Start	F1h	nn	信息码	CRC	End
	空心五边形	Start	F2h	nn		CRC	End	Start	F2h	nn	信息码	CRC	End

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	主控端发送 (TFT 串口屏接收) 起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	指令参数	CRC 码 (2Bytes)	结束码 (4Bytes)	主控端接收 (TFT 串口屏发送) 起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	信息码/反馈码 (1Bytes)	CRC 码 (2Bytes)	结束码 (4Bytes)
	实心五边形	Start	F3h	nn		CRC	End	Start	F3h	nn	信息码	CRC	End
	圆柱体	Start	F4h	nn		CRC	End	Start	F4h	nn	信息码	CRC	End
	方柱体	Start	F5h	nn		CRC	End	Start	F5h	nn	信息码	CRC	End
	表格视窗	Start	F6h	nn		CRC	End	Start	F6h	nn	信息码	CRC	End
数字键盘	数字键盘输入	Start	A4h	00		CRC	End	Start	A4h	nn	信息码	CRC	End
		按下数字键后						Start	A4h	nn	ASCII + 信息码	CRC	End
		按下 CR 键后						Start	A4h	nn	ASCII + 信息码+内容	CRC	End
	取消数字键盘	Start	A5h	00		CRC	End	Start	A5h	nn	信息码	CRC	End
全功能键盘	全键盘输入	Start	A6h	00		CRC	End	Start	A6h	nn	信息码	CRC	End
		按下 CR 键后						Start	A6h	nn	信息码+内容 (ASCII+GB2312)	CRC	End
	取消全键盘	Start	A7h	00		CRC	End	Start	A7h	nn	信息码	CRC	End
摄像头	显示摄像头画面	Start	AEh	00		CRC	End	Start	AEh	nn	信息码	CRC	End
	取消摄像头画面	Start	AFh	00		CRC	End	Start	AFh	nn	信息码	CRC	End
串口屏侦测	联机检查	Start	BEh			CRC	End	Start	BEh	00	5Ah, or 55h	CRC	End
	版本检查	Start	BFh			CRC	End	Start	BFh	MCU Code(5) + Module Info. (42)	信息码	CRC	End
设定时钟	设定时钟	Start	8Ch		Y, M, D, H, M, S, W (7 Bytes)	CRC	End	Start	8Ch	00	信息码	CRC	End
	读取时钟	Start	8Dh			CRC	End	Start	8Dh	Y, M, D, H, M, S, W (8)	信息码	CRC	End
显示时钟/星期	显示数字时间、日期	Start	92h	nn		CRC	End	Start	92h	nn	信息码	CRC	End
	显示模拟时间、日期	Start	93h	nn		CRC	End	Start	93h	nn	信息码	CRC	End
	显示星期	Start	9Dh	nn		CRC	End	Start	9Dh	nn	信息码	CRC	End
寄存器	执行 9A 指令	Start	CAh	Reg		CRC	End	Start	CAh	nn	信息码	CRC	End
	设定寄存器	Start	CBh	Reg		CRC	End	Start	CBh	nn	信息码	CRC	End

© Levetop Semiconductor Co., Ltd.

主功能	细项功能	主控端发送 (TFT串口屏接收)						主控端接收 (TFT串口屏发送)					
		起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	指令参数	CRC码 (2Bytes)	结束码 (4Bytes)	起始码 (1Bytes)	指令码 (1Byte)	序号 (1Byte)	信息码/反馈码 (1Bytes)	CRC码 (2Bytes)	结束码 (4Bytes)
控制	写入数据	Start	CCh	Data		CRC	End	Start	CCh	nn	信息码	CRC	End
	读取数据	Start	CDh	00		CRC	End	Start	CDh	Data	信息码	CRC	End
	寄存器数据加1	Start	CEh	Reg		CRC	End	Start	CEh	nn	信息码	CRC	End
	寄存器数据减1	Start	CFh	Reg		CRC	End	Start	CFh	nn	信息码	CRC	End

© Levetop Semiconductor Co., Ltd.

2.3 各芯片支持的指令

表 2-3：各芯片支持的指令

No.	指令功能	指令码	MCU + 7680	LT7688	LT7689	LT776	LT268A	LT268B	LT268C LT268D	LT269
1	显示单张或多张图片	80h	V	V	V	V	V	V	V	V
2	显示循环重迭图片	81h	V	V	V	V	V	V	V	V
3	显示带透明效果的单张图片	82h	V	V	V	V				
4	显示变淡	83h								
5	取消显示循环重迭图片	84h	V	V	V	V	V	V	V	V
6	取消显示单张或多张图片	85h	V	V	V	V	V	V	V	V
7	显示 GIF 图片	88h	V	V	V	V	V	V	V	V
8	取消显示 GIF 图片	89h	V	V	V	V	V	V	V	V
9	设定显示单张或多张图片	8Ah	V	V	V	V	V	V	V	V
10	进行电阻屏校验	8Bh	V		V	V	V	V	V	V
11	设定时钟 年/月/日/时/分/秒/周	8Ch	V		V	V		V	V	V
12	回应时钟 年/月/日/时/分/秒/周	8Dh	V		V	V		V	V	V
13	设定显示缓冲区 On/Off	8Eh	V	V	V	V				
14	显示单一图片	8Fh	V	V	V	V	V	V	V	V
15	显示图片式 (客制化) 的数字	90h	V	V	V	V	V	V	V	V
16	显示图片式 (客制化) 的数字	91h	V	V	V	V	V	V	V	V
17	显示 RTC 数字时钟	92h	V		V	V		V	V	V
18	显示 RTC 模拟时钟	93h	V		V	V		V	V	V
19	显示触控滑条	94h	V	V	V	V		V	V	V
20	移除触控滑条	95h	V	V	V	V		V	V	V
21	设置环形触控滑条	96h	V		V	V				
22	移除环形触控滑条	97h	V		V	V				
23	显示二维码	98h	V	Option	V	V		V	V	V
24	执行多组的命令(最多 8 组)，当 nn = 00 时为开机后执行的命令	9Ah	V	V	V	V	V	V	V	V
25	设定控件显示位置及触控后执行的命令	9Bh	V	Option	V	V				
26	显示底图数及控件数量	9Ch	V	Option	V	V				
27	显示星期	9Dh	V		V	V		V	V	V
28	取消单一功能	9Eh	V	V	V	V		V	V	V
29	显示 ASCII 码图片	9Fh	V	V	V	V		V	V	V
30	显示控件图片，及在按下触控后执行的命令（8 组）	A0h	V	V	V	V	V	V	V	V
31	取消显示控件图片及功能	A1h	V	V	V	V	V	V	V	V
32	设定虚拟控件区域，及在按下触控后执行的命令（8 组）	A2h	V	V	V	V	V	V	V	V
33	取消虚拟控件及功能	A3h	V	V	V	V	V	V	V	V
34	显示设置密码输入界面	A4h	V		V	V		V	V	V
35	取消密码输入界面	A5h	V		V	V		V	V	V
36	显示全键盘界面	A6h			V	V				
37	取消显示全键盘界面	A7h			V	V				
38	开启摄像头	AEh				V				
39	关闭摄像头	AFh				V				
40	显示进度条指标图	B0h	V	V	V	V	V	V	V	V
41	显示指针指标图	B1h	V	V	V	V				

© Levetop Semiconductor Co., Ltd.

No.	指令功能	指令码	MCU + 7680	LT7688	LT7689	LT776	LT268A	LT268B	LT268C LT268D	LT269
42	曲线缓冲区	B2h	V		V	V		V	V	V
43	显示曲线	B3h	V		V	V		V	V	V
44	设定触控滑动多张全屏图片	B4h	V	Option	V	V				
45	播放 Wav 檔	B8h	V	Option	V	V		V	V	V
46	停止播放 Wav 檔	B9h	V	Option	V	V		V	V	V
47	调整背光亮度	BAh	V	V	V	V	V	V	V	V
48	显示 On/Off	BCh	V	V	V	V	V	V	V	V
49	串口屏复位	BDh	V	V	V	V	V	V	V	V
50	联机检查 TFT 串口屏	BEh	V	V	V	V	V	V	V	V
51	版本检查	BFh	V	V	V	V	V	V	V	V
52	显示字库 - 1 文字	C0h	V	V	V	V	V	V	V	V
53	显示字库 - 2 文字	C1h	V	V	V	V	V	V	V	V
54	显示字库 - 3 文字	C2h	V	V	V	V	V	V	V	V
55	显示字库 - 4 文字	C3h	V	V	V	V	V	V	V	V
56	WIFI 设定	C4h								
57	手势操作	C9h	V		V	V		V	V	V
58	寄存器操作	CAh	V		V	V	V	V	V	V
59		CBh	V		V	V	V	V	V	V
60		CCh	V		V	V	V	V	V	V
61		CDh	V		V	V	V	V	V	V
62		CEh	V		V	V	V	V	V	V
63		CFh	V		V	V	V	V	V	V
64	设定显示大字库 - 1 文字	D0h	V	V	V	V				
65	设定显示大字库 - 2 文字	D1h	V	V	V	V				
66	设定显示大字库 - 3 文字	D2h	V	V	V	V				
67	设定显示大字库 - 4 文字	D3h	V	V	V	V				
68	显示卷动出现图片	D8h	V	V	V	V	V	V	V	V
69	显示循环卷动图片	D9h	V	V	V	V	V	V	V	V
70	取消循环卷动图片	DBh	V	V	V	V	V	V	V	V
71	显示环形指标图	DCh	V	V	V	V	V	V	V	V
72	画点	DFh	V	V	V	V	V	V	V	V
73	画条直线	E0h	V	V	V	V	V	V	V	V
74	画空心圆形	E1h	V	V	V	V	V	V	V	V
75	画实心圆形	E2h	V	V	V	V	V	V	V	V
76	画带框实心圆形	E3h	V	V	V	V	V	V	V	V
77	画空心椭圆形	E4h	V	V	V	V	V	V	V	V
78	画实心椭圆形	E5h	V	V	V	V	V	V	V	V
79	画带框实心椭圆形	E6h	V	V	V	V	V	V	V	V
80	画空心矩形	E7h	V	V	V	V	V	V	V	V
81	画实心矩形	E8h	V	V	V	V	V	V	V	V
82	画带框实心矩形	E9h	V	V	V	V	V	V	V	V
83	画空心圆角矩形	EAh	V	V	V	V	V	V	V	V
84	画实心圆角矩形	EBh	V	V	V	V	V	V	V	V
85	画带框实心圆角矩形	ECh	V	V	V	V	V	V	V	V
86	画空心三角形	EDh	V	V	V	V	V	V	V	V
87	画实心三角形	EEh	V	V	V	V	V	V	V	V
88	画带框实心三角形	EFh	V	V	V	V	V	V	V	V

© Levetop Semiconductor Co., Ltd.

No.	指令功能	指令码	MCU + 7680	LT7688	LT7689	LT776	LT268A	LT268B	LT268C LT268D	LT269
89	画空心四边形	F0h	V	V	V	V				
90	画实心四边形	F1h	V	V	V	V				
91	画空心五边形	F2h	V	V	V	V				
92	画实心五边形	F3h	V	V	V	V				
93	画圆柱体	F4h	V	V	V	V	V	V	V	V
94	画方柱体	F5h	V	V	V	V				
95	画表格视窗	F6h	V	V	V	V	V	V	V	V
96	设定音量	F7h								
97	停止/暂停/继续 视频/音乐播放	F8h								
98	播放 MP3 音乐	F9h								
99	播放 MP4 视频	FAh								
100	播放 MP4 视频	FBh								
101	矢量字库	FCh			V（需要集通字库配合）					
102	矢量字库	FDh			V（需要集通字库配合）					

© Levetop Semiconductor Co., Ltd.

2.4 RS-232(UART) 通讯协议

主控端的系统或是主板透过 UART 串口传递显示指令给 TFT 串口屏时，除了 **指令码、序号、指令参数** 外还要加上 1 个 Byte 的 **起始码**（固定为 0xAA）、2 个 Byte 的 **CRC 码**、4 个 Byte 的 **结束码**（默认为 0x E4、0x 1B、0x 11、0x EE），用户可自定义编辑结束码，后面结束以默认结束为例，指令信息如下表：

表 2-4：串口屏接收的指令信息

起始码	指令码	序号	指令参数	CRC 码	结束码
0xAA (1 Byte)	1 Byte	1 Byte	n Bytes	2 Bytes	**0xE4、0x1B、0x11、0xEE** (4 Bytes)

CRC 码协议如下：

```
chkSum = Rx_CRC_CCITT(txBuf,txLen);
  txBuf[txLen++] = (chkSum>>8)&0xFF;
  txBuf[txLen++] =    chkSum&0xFF;

unsigned int Rx_CRC_CCITT(unsigned char *puchMsg, unsigned int usDataLen)
{
  unsigned char i = 0;
  unsigned short wCRCin = 0x0000;
  unsigned short wCPoly = 0x1021;
  unsigned char wChar = 0;

  while (usDataLen--)
  {
    wChar = *(puchMsg++);
    wCRCin ^= (wChar << 8);
    for(i = 0; i < 8; i++)
    {
      if (wCRCin & 0x8000)
        wCRCin = (wCRCin << 1) ^ wCPoly;
      else
      wCRCin = wCRCin << 1;
    }
  }
  return (wCRCin);
}
```


© Levetop Semiconductor Co., Ltd.

串口屏在收到主控端的系统或是主板指令后会通常会响应 10 个 Byte 信息，包括 **起始码、指令码、序号、信息码、CRC 码、结束码**，第一个 Byte 是**起始码**，然后是传回所收到的指令，第三个是**序号**，第四个传回串口屏执行结果的**信息码**，第五、六个是 **CRC 码**，最后是 4 个 Bytes 的 **结束码**：

表 2-5：串口屏反馈的信息

起始码	指令码	序号	信 息 码	CRC 码	结 束 码
0xAA (1 Byte)	1 Byte	一般指令 (1 Byte) 8Dh 指令 (8 Bytes) BFh 指令 (47 Bytes)	1Byte 0x00：执行完该指令 0x01：串口指令参数错误 0x02：不存在该指令 0x03：指令 Flash 配置溢出 0x04：CRC 码校正错误 0x05：Flash 数据异常 BEh 指令: 0x5A: Ready 0x55: Not Ready 94h 触控滑条指令: 代表进度条指标的百分比位置 A0h, 9Ch/9Bh 控件指令: 0x31: 按下控件 0x30: 放开控件	2 Bytes	0xE4、0x1B、 0x11、0xEE (4 Bytes)

在串口屏反馈的信息结构中，序号在某些指令也代表不同的意思，如控件滑动的 9Ch 指令其序号代表页号、9Bh 指令其序号代表图标 ID 号、设置亮度 BAh 指令其序号代表背光亮度、Wav 播放 B8h 指令其序号代表 WAV 编号、读取时钟 8Dh 指令其序号有 8 个 Bytes 代表时钟信息、版本检查 BFh 指令其序号有 47 个 Bytes 代表串口屏信息。

© Levetop Semiconductor Co., Ltd.

2.5 主控端发送指令的范例

以下是主控端的 MCU 透过 UART 串口传递显示指令给 TFT 串口屏的程序范例，本程序以传送显示第一张图片（80h、00h）为范例，程序内会自动加入 0xAA 起始码、2 个 Byte 的 CRC 码及 4 个 Byte 的 结束码：

```
int main (void)
{
        char c[] = "80 00";                    //发送 80h 指令，和 00 序号
        Send(c);

        while(1);

}

unsigned short Rx_CRC_CCITT(unsigned char *puchMsg, unsigned int usDataLen) //  生成 CRC
{
        unsigned char i = 0;
        unsigned short wCRCin = 0x0000;
        unsigned short wCPoly = 0x1021;
        unsigned char wChar = 0;

        while (usDataLen--)
        {
         wChar = *(puchMsg++);
         wCRCin ^= (wChar << 8);
         for(i = 0; i < 8; i++)
         {
          if (wCRCin & 0x8000)
            wCRCin = (wCRCin << 1) ^ wCPoly;
          else
            wCRCin = wCRCin << 1;
         }
        }
        return (wCRCin);
}


void Send(char *c)                              //发送指令函数
{
        unsigned char Sendbuff[100]={0};
        unsigned short Send_CRC = 0;
        unsigned char C_flag = 0;              //判断是否在" "中

        int i = 0, j = 0;

        if(((c[0]>=0x30 && c[0]<=0x39) || (c[0]>=0x41 && c[0]<=0x5A))   ||   ((c[1]>=0x30 &&
c[1]<=0x39) || (c[1]>=0x41 && c[1]<=0x5A)))                //只有第一个和第二个字符是有效的才是指
令
        {
         while(c[i] != '\0')
         {
          if(c[i] != ' ')                      //排除空格，非空格可以进入
          {
            if(c[i] == '"')                    //当" "中的字符 以 ASCII 输出，不需要输出" 号，
```


© Levetop Semiconductor Co., Ltd.

```
{
 C_flag++;
 i++;
}
if(C_flag == 1)
{
 if(c[i] != '"')
 {
  Sendbuff[j] = c[i];              //ASCII 直接输出
  i++;
  j++;
 }

}
else if(C_flag == 2)              //第二次遇到"
{
 C_flag = 0;
 i++;
}
if(C_flag == 0)
{
 if(c[i] == '/') break;

 if(c[i]>=0x30 && c[i]<=0x39) //0~9
 {
  Sendbuff[j] = ((c[i] - 0x30)<< 4 );
  i++;

  if(c[i]>=0x30 && c[i]<=0x39)
  {
   Sendbuff[j] += (c[i] - 0x30);
   i++;
   j++;
  }
  else if(c[i]>=0x41 && c[i]<=0x5A)
  {
   Sendbuff[j] += (c[i] - 0x37);
   i++;
   j++;
  }
 }

 else if(c[i]>=0x41 && c[i]<=0x5A) //A~Z
 {
  Sendbuff[j] = ((c[i] - 0x37)<< 4 );
  i++;

  if(c[i]>=0x30 && c[i]<=0x39)
  {
   Sendbuff[j] += (c[i] - 0x30);
   i++;
   j++;
  }
  else if(c[i]>=0x41 && c[i]<=0x5A)
  {
   Sendbuff[j] += (c[i] - 0x37);
   i++;
```


© Levetop Semiconductor Co., Ltd.

```
                j++;
               }

              }


             }
            }
            else i++;
          }
          Sendbuff[j] = '\0';
//        printf("%s\r\n",Sendbuff);

          /************************CRC 和针头针尾************************/

          Send_CRC = Rx_CRC_CCITT(Sendbuff,j);

          Sendbuff[j] = Send_CRC>>8 & 0xff;
          Sendbuff[j+1] = Send_CRC & 0xff;

          for(i = 0;i<j+2;i++)
           Sendbuff[j+2-i]=Sendbuff[j+1-i];                    //挪位

          Sendbuff[0] = 0xAA;
          Sendbuff[j+3] = 0xE4;
          Sendbuff[j+4] = 0x1B;
          Sendbuff[j+5] = 0x11;
          Sendbuff[j+6] = 0xEE;

          j+=7;

          for(i = 0; i < j; i++)
           {
            USART_DATA(USART0) = (uint8_t) Sendbuff[i];
            while(usart_flag_get(USART0, USART_FLAG_TBE)==0){};      //循环发送,直到发送完毕
           }

        }
}
```


© Levetop Semiconductor Co., Ltd.

3. 图文 UI 编辑器（UI_Editor.exe）

3.1 UI_Editor 界面的简介

UI_Editor.exe 是 乐升半导体 提供的一款以串口屏为对象的 **图文 UI 编译器**。它的功能是根据客户的需求，将串口屏要用到的图片、文字、配置数据等信息打包生成 BIN 档。客户可以使用 UI_Editor 简单、快捷的制作 UI 界面，之后将生成的 BIN 文档烧录到 SPI Flash 中。

提示： UI_Editor 是在 Microsoft .NET Framework 4.6.2 的环境中编写出来的，所以电脑系统必须安装 Microsoft .NET Framework 4.6.2 才能正常使用。

UI_Editor 的界面由各种按钮和屏幕框组成如下图所示：



图 3-1：UI_Editor 主画面

© Levetop Semiconductor Co., Ltd.

所有的 UI 设计都在屏幕框内完成，用户根据需求选用不同的功能实现设计。其中各种功能键的详细功能如下：

1.  在 UI_Editor 上以图片形式显示文字
2.  添加图片按钮
3.  添加控件按钮
4.  添加 GIF 图按钮
5.  添加数字时钟按钮
6.  添加星期按钮
7.  添加数字按钮（图片）
8.  添加真彩图片按钮（图片）
9.  添加文字按钮（字库）
10.  添加 ASCII 按钮
11.  添加二维码按钮
12.  添加表格
13.  添加画圆环（任意角度）
14.  添加进度条按钮
15.  添加触控滑动条按钮
16.  分别是画矩形、画圆角矩形、画圆、画椭圆
17.  分别是画线、画三角形、画四边形、画五角形、画圆柱体、画长方体
18.  添加指针按钮
19.  添加虚拟按键按钮
20.  分别是撤回操作和恢复操作按钮（当无任何操作时初始画面是 

© Levetop Semiconductor Co., Ltd.

与 UI_Editor 工具同级的有几个文件夹，它们的作用如下图所示。

- ★ ColorNumber 文件夹用来存放真彩数字图片文件
- ★ FONT 文件夹用来存放需要使用的字库
- ★ PICFILE 文件夹可用来先存放需要使用到的图片文件
- ★ PROJECT 文件夹备份着每次 Save 和 Build 的工程文件
- ★ SOURCE 文件夹用来存放音频 BIN 文件

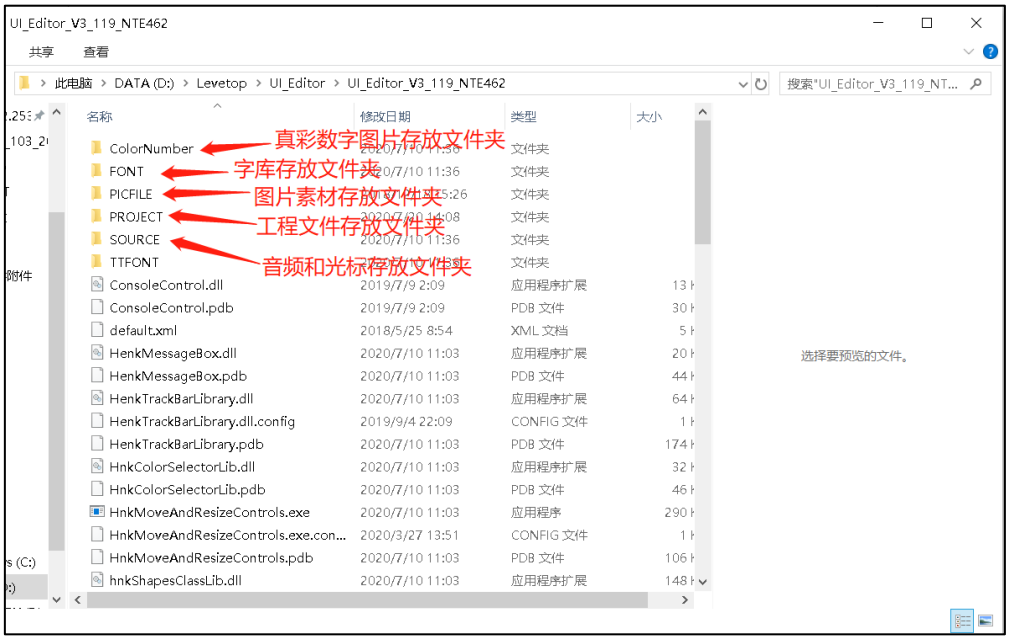


图 3-2：UI_Editor 工具同级文件目录

© Levetop Semiconductor Co., Ltd.

PROJECT 文件夹下级的工程文件里有几个文件夹，它们的作用如下图所示。

- ★ BINFILE 文件夹存放着编译好的 BIN 文件，需要烧录的 UserInfo 和 UartTFT_Flash 就存放在此处。
- ★ COMMANDFILE 文件夹存放着工程储存文件
- ★ PICFILE 文件夹存放着编译后的图片文件
- ★ SRCPIC 文件夹存放着编译前的图片

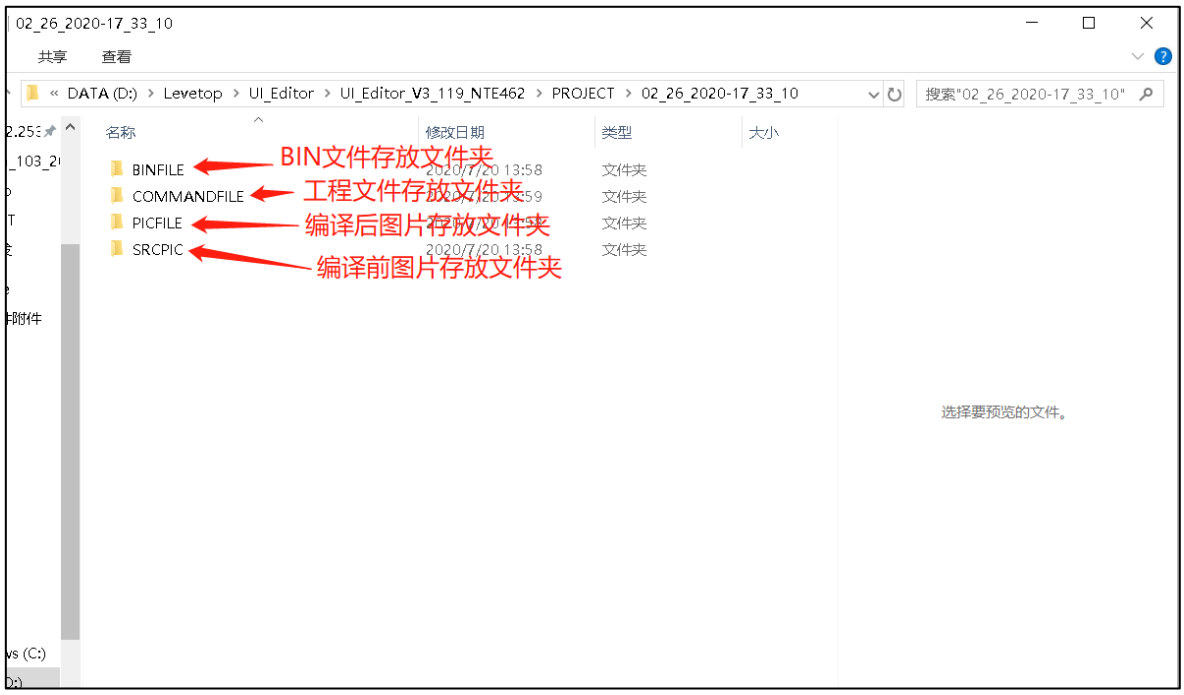


图 3-3：PROJECT 文件夹下级的工程文件目录

© Levetop Semiconductor Co., Ltd.

在菜单按钮里，有 New Project、load、save 三个按钮。分别用来创建新工程、装载工程文件、保存当前工程。按 save 按钮会把工程以 mainControlFiles.xml 文件保存在 PROJECT 下级中以时间命名的 COMMANDFILE 文件夹里。使用 Load 功能在 PROJECT 下级找到对应时间的文件夹里 COMMANDFILE 文件夹的 mainControlFiles.xml 文件，就可以重新加载工程。



图 3-4：UI_Editor 重装载工程文件

© Levetop Semiconductor Co., Ltd.

3.1.1 使用 UI_Editor 的设计流程

下图为用图文 UI 编译器（UI_Editor.exe）开发的详细流程图，用户也可以至 乐升半导体 网站下载各串口屏芯片的 UI_Editor 范例（如 LT7689_UI_Editor_Demo_800x480.rar）来操作，将更快速地了解开发模式。同时建议用户先依据所需功能及 TFT 屏幕大小准备好素材，因为这些显示图片、动画文件、文字库、声音文件等是存放在 SPI Flash 内，资料量都不小，而 SPI Flash 的烧录所需时间较长，因此尽量避免开发中反复对 SPI Flash 进行 UartTFT_Flash.bin 档的烧写，以免延误开发效率。乐升半导体 的 UI_Editor 图文 UI 编译器及 UI_Emulator 串口屏仿真器软件都可以自官网的“串口屏上位机软件”下载专区下载。

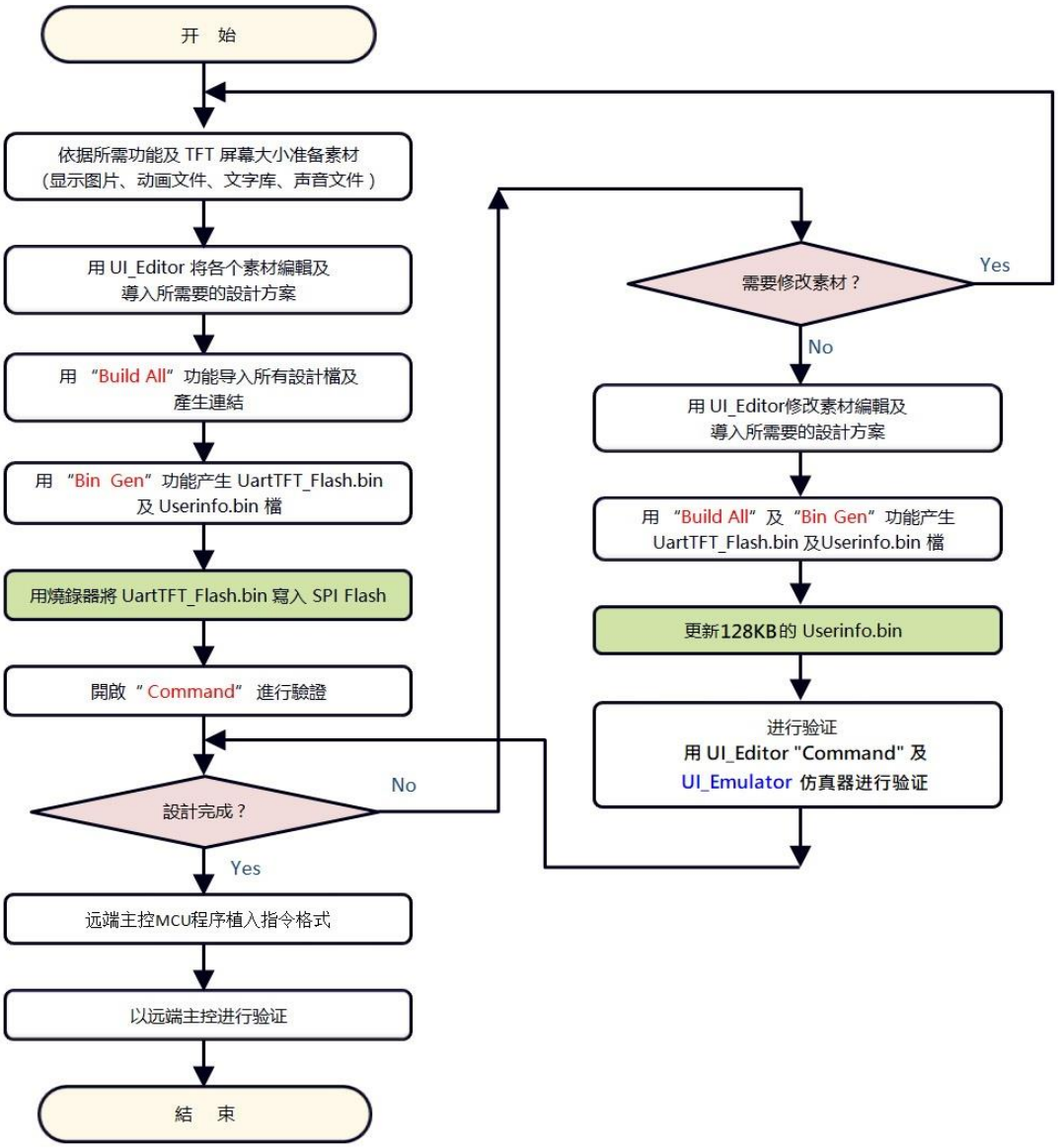


图 3-5：使用 UI_Editor 的设计流程

© Levetop Semiconductor Co., Ltd.

在使用 UI_Editor 工具之前用户需依据产品的需求进行显示规划，这些规划与设计与 UI_Editor 无关，使用者可以用绘图软件去设计图片或是动画（Gif 檔），显示的流程可以用 Word、PowerPoint 或是自己熟悉的工具去记录，规划及准备的方向大致如下所示：

1. 确认显示屏的分辨率，例如 480*272 或是 800*480 等等。
2. 以 1:1 方式设计产品中会需要显示的底图，例如显示屏 480*272，那么底图就以 480*272 分辨率来设计。虽然 UI_Editor 可以改变导入图片的大小，但是不同的算法会导致图片的局部失真，因此强烈建议 UI 工程师以 1:1 方式设计图片。
3. 设计产品中需要显示的各个大小的图片、或是动画（Gif 檔）。
4. 设计产品中需要显示的图片或是动画的显示坐标。
5. 设计显示的流程，例如主控发不同指令后显示屏会更动哪那些画面。这些显示流程跟用户的产品有绝对的关联，必须先思考清楚，而不是在使用 UI_Editor 时一边思考一边导入。
6. 确认是否使用触控屏功能，如果有就要设计触控控件的图片及显示坐标位置。
7. 如果需要音乐或是中文字库，也要准备音源及字库檔利用工具转成 Bin 檔。
8. 在检视整个显示流程无误及准备好所以素材图片之后，就可以开启 UI_Editor 进行显示图片及流程的导入。

以上流程的目的是强调用户在操作 UI_Editor 工具之前的准备工作，这样可以大大地缩短实际设计时间，及减少非常多的编译次数，提升导入串口屏显示的开发效率。

© Levetop Semiconductor Co., Ltd.

3.1.2 使用 UI_Editor 的注意事项

在使用 UI_Editor 工具之前用户需依据产品的需求进行显示规划，这些规划与设计与 UI_Editor 无关，使用者可以用绘图软件去设计图片或是动画（Gif 文件），显示的流程可以用 Word、PowerPoint 或是自己熟悉的工具去记录，规划及准备的方向大致如下所示：

1. 根据产品应用的硬件、软件及显示功能需求去选用本公司芯片，可以参考芯片规格及支持的显示功能，或与本公司业务/FAE 接洽。
2. 至本公司官网（www.levetop.cn）下载最新的 UI_Editor 开发软件及对应芯片的应用手册。
3. 确认显示屏的分辨率，例如 480*272 或是 800*480 等等。
4. 以 1:1 方式设计产品中会需要显示的底图，例如显示屏 480*272，那么底图就以 480*272 分辨率来设计。虽然 UI_Editor 可以改变导入图片的大小，但是不同的算法会导致图片的局部失真，因此强烈建议 UI 工程师以 1:1 方式设计图片。
5. 设计产品中需要显示的各个大小的图片、或是动画（Gif 檔）。
6. 设计产品中需要　显示的图片或是动画的显示坐标。
7. 设计显示的流程，例如主控发不同指令后显示屏会更动哪那些画面。这些显示流程跟用户的产品有绝对的关联，必须先思考清楚，而不是在使用 UI_Editor 时一边思考一边导入。
8. 确认是否使用触控屏功能，如果有就要设计触控控件的图片及显示坐标位置。
9. 如果需要音乐或是中文字库，也要准备音源及字库檔，利用工具转成 Bin 檔。
10. 在检视整个显示流程无误及准备好所以素材图片之后，就可以开启 UI_Editor 进行显示图片及流程的导入。

以上流程的目的是强调用户在操作 UI_Editor 工具之前的准备工作，这样可以大大地缩短实际设计时间，及减少非常多的编译次数，提升导入串口屏显示的开发效率。

在使用 UI_Editor 的制作项目时，建议使用者对不同的项目使用不同的 UI_Editor 目录，可以通过复制 UI_Editor 的文件夹即可。UI_Editor 文件夹可重命名为项目名称，便于区分不同项目保存或编译后产生的工程，如下图：



图 3-6：不同的工程建立不同的项目目录

© Levetop Semiconductor Co., Ltd.

在制作工程时，可通过点击工具下方“SAVE”保存当前制作的工程，或通过编译“Build All”来保存，如下图：

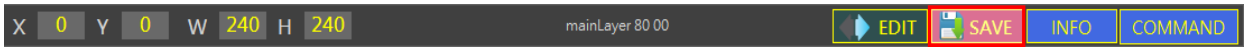


图 3-7：点击“SAVE”保存当前制作的工程



图 3-8：点击编译“Build All”保存当前制作的工程

保存后会自动生成一个以时间命名的工程文件夹，该工程文件夹保持 UI_Editor 路径下的 PROJECT 文件夹中，用户可以自行选择删除不需要的工程文件夹，如下图。若只需保存最新的工程，可在调试完成后删除旧的工程文件夹，只保留最新的即可。

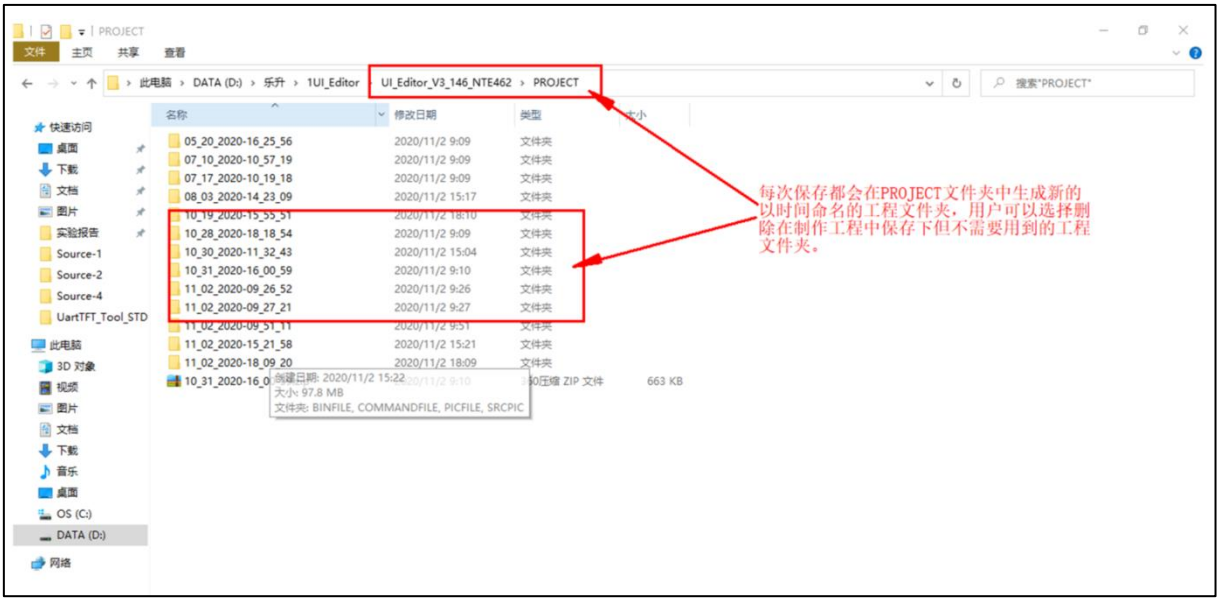


图 3-9：可以自行选择删除不需要的工程文件夹

注意 1：UI_Editor 文件夹名称要根据原来自动产生的为主，不可更改，若是客户已经改了，会 Load project 失败，此时可将文件夹名称再改为原来的年月日时分秒名称即可；若是忘了名称，例如：可打开 Projetct-->COMMANDFILE-->mainControlFiles.xml，找到 <Text> PROJECT\08_26_2020-09_30_40\.... 则“08_26_2020-09_30_40”即为原本的 Project 名称。

注意 2：若是客户需要以特定文件夹名称保存 Proejct，建议客户可用自订文件夹名称内再放入 Proejct，再压缩备份。

© Levetop Semiconductor Co., Ltd.

3.1.3 屏幕初始化和背景图的设定

先打开 UI_Editor。

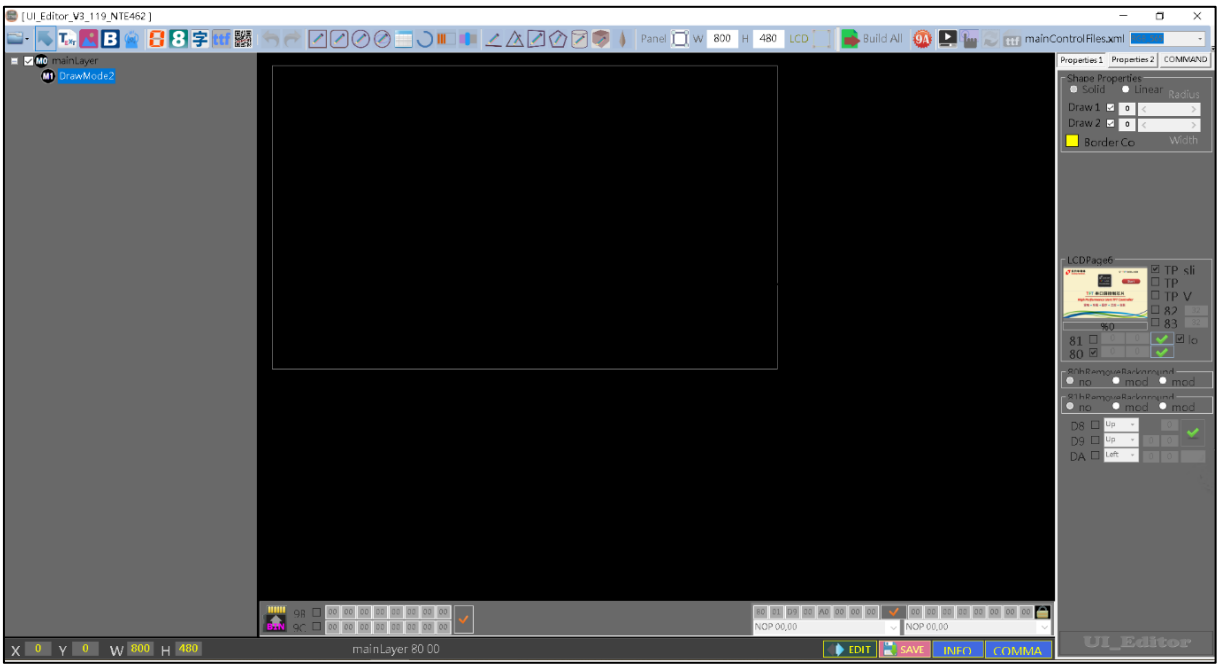


图 3-10：打开的 UI_Editor

先在菜单中选择“New Project”，再点击“屏幕框选择按钮”，最后再双击框内任意位置选择打开要在该图层的图片，如下图所示：

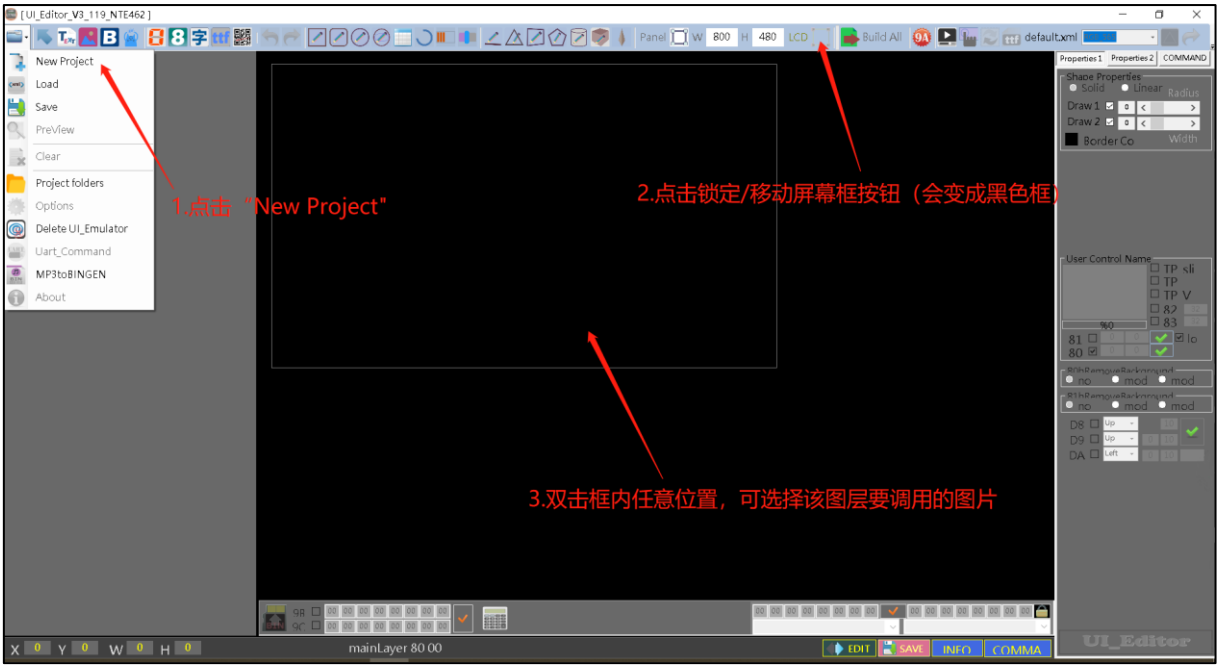


图 3-11：新建工程

© Levetop Semiconductor Co., Ltd.

通过修改右边的屏幕大小内的数字可以改变屏幕的长、宽。要注意的是每个框中的数字修改了之后要按回车来确认，否则无法修改。例如要设定 5 寸 800X480 分辨率的屏幕。则将 W 修改为 800 按下回车键确认，再将 H 修改为 480 按下回车键确认。屏幕大小的设定可以在任何时候进行。在进行 UI 设计时一定要确认设定的屏幕大小没有错误。



图 3-12：设置图层分辨率

点击“锁定/移动屏幕框”按钮后，整个屏幕框可以被拖动到任意位置，可放在视野适合的位置。再次点击该按钮后，即取消了屏幕框选择功能，此时不能修改背景图和拖动屏幕框。



图 3-13：拖动屏幕框到任意位置

© Levetop Semiconductor Co., Ltd.



图 3-14：取消了屏幕框选择功能

© Levetop Semiconductor Co., Ltd.

3.1.4 显示图的设定

初始化了屏幕框之后，默认是屏幕框显示的是一个主图层。我们可以通过在项目栏中主图层选项中右键，选择 Add New Page 来增加图层。



图 3-15：增加图层

如果所建图层只需与主图层共享一个一张图片，这时在此界面点击“取消”即可。

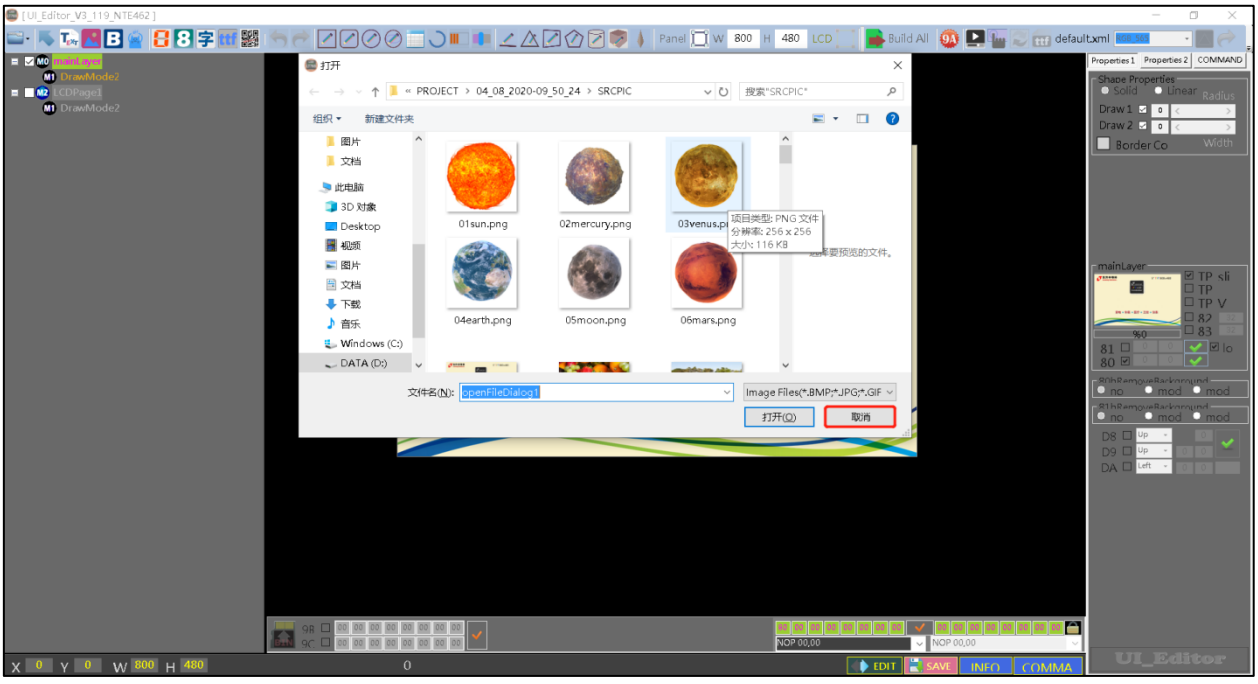


图 3-16：新建图层与主图层

© Levetop Semiconductor Co., Ltd.

如果所建图层需要新建一张独立的图片，点然后击“打开”。

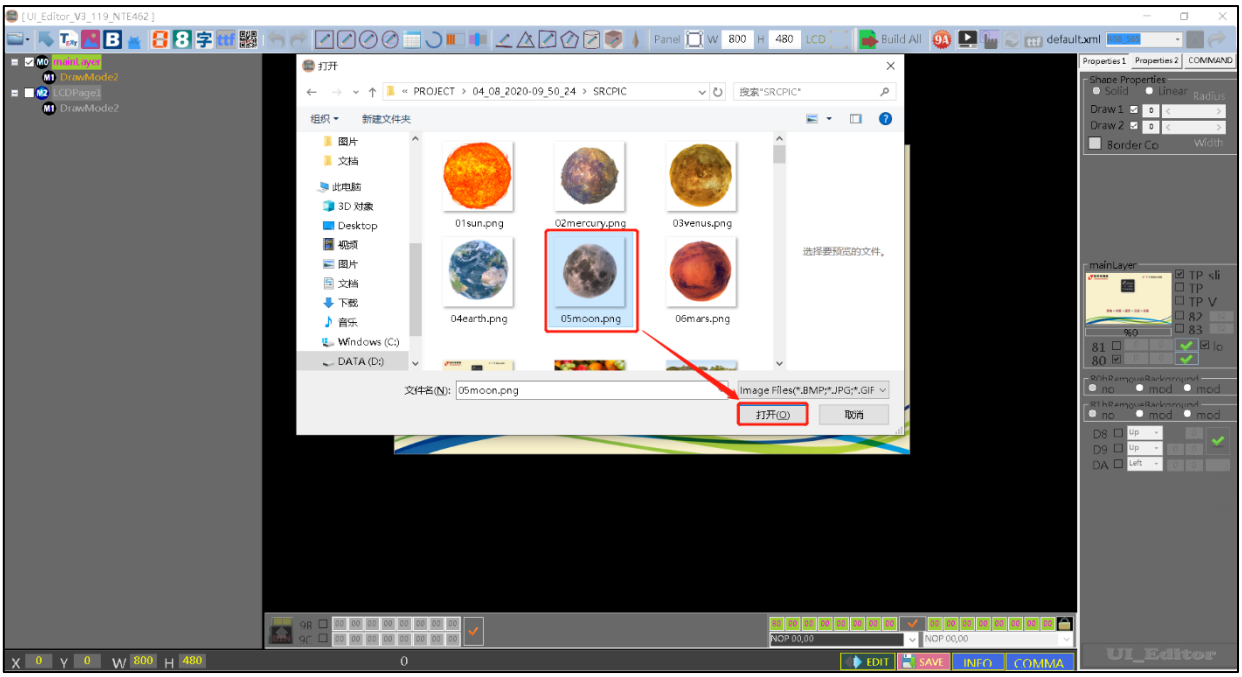


图 3-17：新建独立的图层

如果需要变更次图层的背景图，先勾选需要变更的图层，点击该图层的“DrawMode”，然后再点击“LCDPage”，接着双击缩略的显示框，即可根据需求来选择图层的背景。

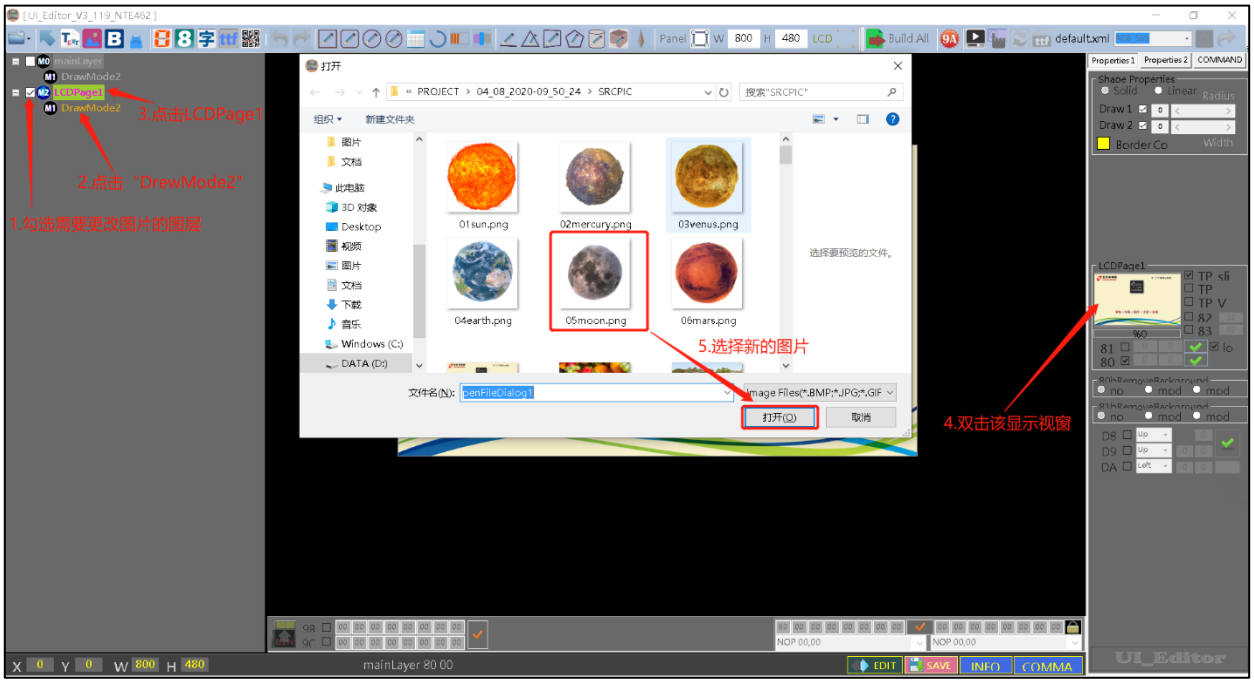


图 3-18：变更图层背景

© Levetop Semiconductor Co., Ltd.

如果需要变更主图层的背景图，先勾选需要变更的主图层，点击“锁定/移动屏幕框”按钮后，双击框内任意空白位置，可选择要调用的图片，最后再点击锁定。



图 3-19：变更主图层背景

当存在多个图层的时候，可以通过勾选图层前面的小方格来显示和隐藏当前图层。在当前的图层，右击该图层可对当图层进行删除或者移动。



图 3-20：删除或移动图层

© Levetop Semiconductor Co., Ltd.

3.1.5 界面的编辑与调试

删除操作有两种以下方式

方式一：先选中要删除的目标，点击右键后选择“Remove”删除该控件，如下图所示：



图 3-21：删除操作方式一

方式二：先在项目栏选中要删除的目标后点击右键，然后点击“Remove”即可删除该目标，如下图所示：



图 3-22：删除操作方式二

© Levetop Semiconductor Co., Ltd.

如下图所示，默认的状态下克隆的功能是异位置克隆，先选中要克隆的控件，点击右键后选择“Clone”就会产生一个相同但异位置的控件。



图 3-23：异位置克隆操作

如下图所示，克隆前先双击右下方的灰色框，此时状态会更改为 false，选中要克隆的控件，点击右键后选择“Clone”就会产生一个相同且同位置的控件。



图 3-24：同位置克隆操作

© Levetop Semiconductor Co., Ltd.

如下图所示，选中的控件右键后可出现控件的菜单，选择 copy 复制该控件。



图 3-25：复制功能一

如下图所示，选择 LCDpage 后，右键点击 paste 生成一个与复制来源相同的控件，可选择同位置复制或异位置复制。

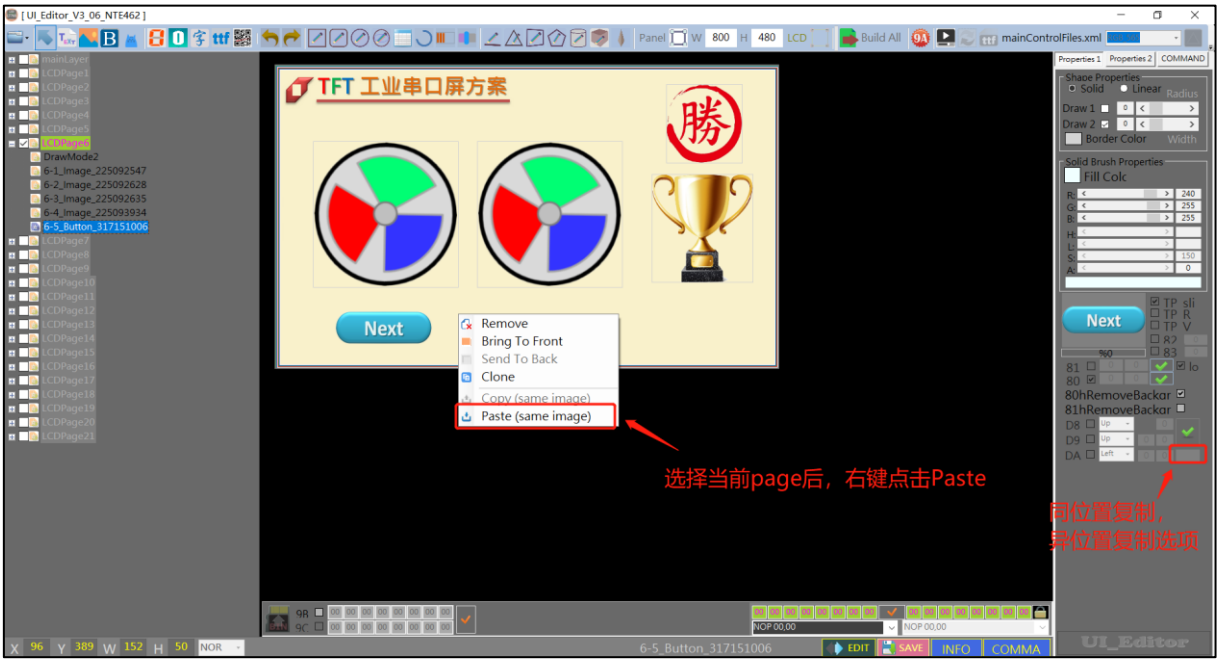


图 3-26：复制功能二

© Levetop Semiconductor Co., Ltd.

如下图所示，复制后会生成一个相同的控件，该控件与来源控件相同。



图 3-27：复制功能三

如下图所示，表示当前处在正常编辑的状态下，所有的添加、删除和编辑功能都能正常执行。



图 3-28：编辑状态

© Levetop Semiconductor Co., Ltd.

如下图所示，表示当前处在调试的状态下，此时不能对界面进行编辑，选中对应的控件点击右键，此时会弹出相关的指令，点击相应的指令即为发送该串口指令，要想重新对界面进行编辑，再次点击下方红色框的这个图标即可。



图 3-29：调试状态

© Levetop Semiconductor Co., Ltd.

3.2 显示图片的设定

3.2.1 设定显示多张图片

先点击图片添加按钮，在屏幕框中点击拖动，出现一个四角带点的白线框。这个白线框就是图片的大小。松开手后，在弹出的资源管理器中选中一张图片。图片的大小可以通过拉动白线框的大小来改变。如果想更换图片可以在图片被选中的前提下，双击图片来更改。如果图片不能被选中（没有出现四角白线框），可以单击项目栏中的项目来选中对应的图片对象。成功添加图片之后，可以看到右边操作栏中有 80h 指令的功能框，80h 指令的序号超过 FF 后 80h 自动变为 8Ah，8Ah 指令功能和操作方法和 80h 指令一致。

显示单张图片：如果想在发送指令后单独显示一张图片，则可以不设置组号。

显示多张图片：要是想要用一个指令显示多张图片，就设定组号，相同组号的图片会一起显示。在设定了组号之后，务必按确认按键完成设定，显示单张/多张图片 80h 指令取消显示指令均为对应序号的 85h 指令，8Ah 没有取消指令。

其中如果图片是 PNG 的话可以勾选去掉 PNG 背景框（80h RemoveBackground）来去掉 PNG 图片的背景（RM1:软件 PNG,RM2:硬件 PNG），如果使用压缩功能再勾选掉 PNG 背景框（80h RemoveBackground）的基础上在勾选 ZIP 压缩，压缩图片不能用作全屏底图，而且支持压缩图片的指令码只限于：80h,81h,84h,85h,88h,89h,8Ah,8Fh,A0h,A1h。如果图片是 JPG 的话可以勾选 JPG 处理背景框（80h RemoveBackground）RM3 来进行图片 JPG 处理,JPG 图片分辨率限制 w*h*2<=90000。



图 3-30：设定显示多张图片

© Levetop Semiconductor Co., Ltd.

3.2.2 设定循环显示重叠图片

在添加了图片之后，可以设定循环显示图片的组号和显示之间的时间间隔。其中时间间隔是以 10ms 为单位。例如图片 A 和图片 B 要以 200ms 时间间隔循环显示。则应该在图片 A 的 81h 指令操作框处设定 1 和 20，按下确定键。在图片 B 的 81h 指令操作框处设定 1 和 0，按下确定键。图片 A 和图片 B 就会形成一个组合 1，在接收到对应的指令后，循环显示。其中如果图片是 PNG 的话可以勾选去掉 PNG 背景框 81h RemoveBackground (mode1:软件 PNG，mode2 硬件 PNG）来去掉 PNG 图片的背景。其中每生成一个 81h 指令（循环显示重叠图片），同时也会生成一个 84h 指令（取消循环显示重叠图片）。



图 3-31：设定循环显示重叠图片

© Levetop Semiconductor Co., Ltd.

3.2.3 设定卷动出现图片

卷动图片效果是从某一个地方按时间间隔逐渐显示一张图片。在添加图片之后，勾选 D8h 操作栏中的小方格，对图片开启卷动功能。之后可以选择图片卷动的方向。有四个选项，分别是：向上卷动、向下卷动、向左卷动和向右卷动。选择一个自己需要的方向后，填写卷动的时间间隔，最后按确认按钮。



图 3-32：设定卷动出现图片

© Levetop Semiconductor Co., Ltd.

3.2.4 设定循环卷动图片

循环卷动图片是在卷动图片的基础上，可以实现一张图片循环卷动出现，或者多张图片持续的顺序卷动出现。具体的操作方法和上面的设定卷动出现图片的基本类似。其中假如要实现一张图片循环卷动，则只在那张图片的 D9h 指令操作栏中设定就可以了。假如要实现多张图片的循环卷动，则在对应图片的 D9h 指令操作栏中同时设定，时间间隔可以不一样。其中每生成一个 D9h 指令（显示循环卷动图片），同时也会生成一个 DBh 指令（取消显示循环卷动图片）。



图 3-33：设定循环卷动图片

© Levetop Semiconductor Co., Ltd.

3.2.5 设定显示 GIF 动画图片

按 GIF 添加按钮，从资源管理器中选一张 GIF 图片。在右边的操作栏中可以找到 GIF 的操作栏。在小演示窗口的右下角的数据框是 GIF 图播放间隔的设定框（以 100ms 为单位），LOOP 为是否循环播放 GIF 图。此时 81h 指令的确认设定按钮变为了 GIF 的确认设定按钮。先设定了播放间隔和是否循环播放 GIF 图后再按确认按钮，即可完成设定。其中每生成一个 88h 指令(显示 GIF 动画图片)，同时也会生成一个 89h 指令（取消显示 GIF 动画图片）。



图 3-34：设定显示 GIF 动画图片

© Levetop Semiconductor Co., Ltd.

3.2.6 设定显示二维码图片

点击二维码添加按钮，在屏幕框中点击拉动，形成白线框。松开，就会弹出二维码设定窗口。在二维码的信息栏中输入要显示的信息。二维码支持显示图标功能。点击了选用图标按钮后，点击图标选择按钮。在弹出的资源管理其中选一个图标。通过图标大小调节条来改变图标的大小。可以点击 Preview 按钮预览二维码显示效果。（注：98h 指令需要的内存为 27k，默认为关闭 98h 指令，若要开启 98h 指令，则必须关闭其他指令使空闲内存达到 27k 以上。）



图 3-35：设定显示二维码

© Levetop Semiconductor Co., Ltd.

3.2.7 设定显示控件功能的图片

可以通过添加按钮控件来实现按键功能，即在屏幕上显示一个按键图片，触摸到这个图片的时候，执行一系列动作。如显示图片、显示 GIF、播放音频等。

先点击按键添加按钮，在屏幕框中点击、拖动生成白线框。在资源管理器中选一张图片作为按键图片。之后可以在屏幕框的下边找到按键的操作框。只有选中了按键图片，它们才会可用。点击第一组指令输入处(两格组成一个指令，最多可执行 8 个指令，若需要执行 8 个以上的指令可使用 9A 进行嵌套)，在下面的指令可选列表中选择要执行的指令。设置好后点击确认设置按钮即可。指令可选列表是由当前工程中其他已生成动作指令组成的。例如在设定按键功能前，已经设定了显示一张图片和一张 GIF 图的指令。则指令列表中会有这两个指令。特别说明，两个按键操作框使用方法一样。如果控件图片是 32 位 PNG 格式，希望去掉背景，则可以勾选 80h RemoveBackground 的去背景框（mode1: 软件 PNG，mode2 硬件 PNG)。其中每生成一个 A0h 指令（显示控件），同时也会生成一个 A1h 指令（取消控件）。



图 3-36：设定显示控件功能的图片

© Levetop Semiconductor Co., Ltd.

3.2.8 设定显示虚拟按键

可以通过添加虚拟按钮控件来实现按键功能，即在屏幕上设定一个区域，但是 user 看不到这个区域的变化（即虚拟按键为完全透明），当触摸到这个区域的时候，执行一系列动作。如显示图片、显示 GIF、播放音频等。

先点击虚拟按键添加按钮，在屏幕框中点击、拖动生成白线框，此框即为虚拟按键的有效范围。选中虚拟按键之后可以在屏幕框的下边找到该虚拟按键的操作框。点击第一组指令输入处（两格组成一个指令，最多可执行 8 个指令），在下面的指令可选列表中选择要执行的指令。设置好后点击确认设置按钮即可。指令可选列表是由当前工程中其他已生成动作指令组成的。例如在设定虚拟按键功能前，已经设定了显示一张图片和一张 GIF 图的指令。则指令列表中会有这两个指令。特别说明，两个按键操作框使用方法一样。其中每生成一个 A2h 指令（显示虚拟控件），同时也会生成一个 A3h 指令（取消虚拟控件）。



图 3-37：设定显示虚拟控件

© Levetop Semiconductor Co., Ltd.

3.2.9 设定开机画面

开机画面设定是用来选择哪些控件在开机时被预先执行，而不需要主控端先发送指令，以下示例假设构建的控件已经都存在：

1. 先点选左上角“mainLayer”，此时会出现 8 组在开机时可预先选择要发送的指令。



图 3-38：点选左上角“mainLay”

© Levetop Semiconductor Co., Ltd.

只要点选任一组数字，即会显示黄色框背景与红色字体，再根据如下图所示位置点选并自动上拉展开 Command 群组，此时即可选择每一组 Command 的指令组合。



图 3-39：选择每一组 Command 的指令组合

2. 开机画面设定如下如所选的指令（User Command）



图 3-40：开机画面所选的指令

© Levetop Semiconductor Co., Ltd.

9A 指令中除了 9A，00 为开机指令外，也可以用来设定执行多组的指令，例如新增加 9A 01、9A 02 等等之后的用户指令，这些指令码主要是让用户可以直接从串口输出 9A 01、9A 02 等等指令，控制 UI_Editor 编辑时预先设定要呼叫的其他指令，范例如下图所示（实际于 LCD 并不会显示 9A 图标）。



图 3-41：以 9A 指令来设定执行多组指令

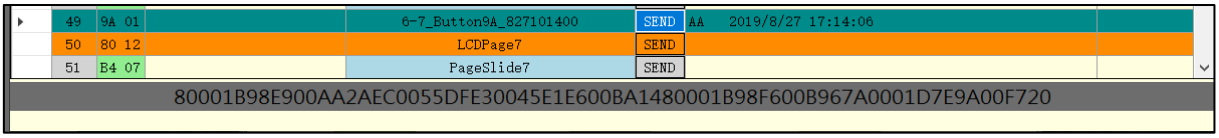


图 3-42：以 9A、01 指令来执行多组指令

9A 指令可以用来合并多个指令，非常好用，建议使用者多加利用，除了简化许多图片显示的设计外，也可以简化主控端的设计。

© Levetop Semiconductor Co., Ltd.

3.2.10 设定触摸进度条图

此功能是结合触控屏的滑动，来改变 TFT 屏上所显示的滑动条长短，同时反馈信息给主控端。

1. 请点 [94h] 滑动条图示，并可开始在屏幕范围内做拖拉移动的动作，此时即可产生滑动条。



图 3-43：设定滑动条及前景颜色与背景颜色

2. 设定滑动条移动时前景颜色与背景颜色
3. 务必要设定滑动条背景图片如下图所示，请拖拉移动一张图片当作滑动条背景，并且要将图片设定为去除边框透明功能。

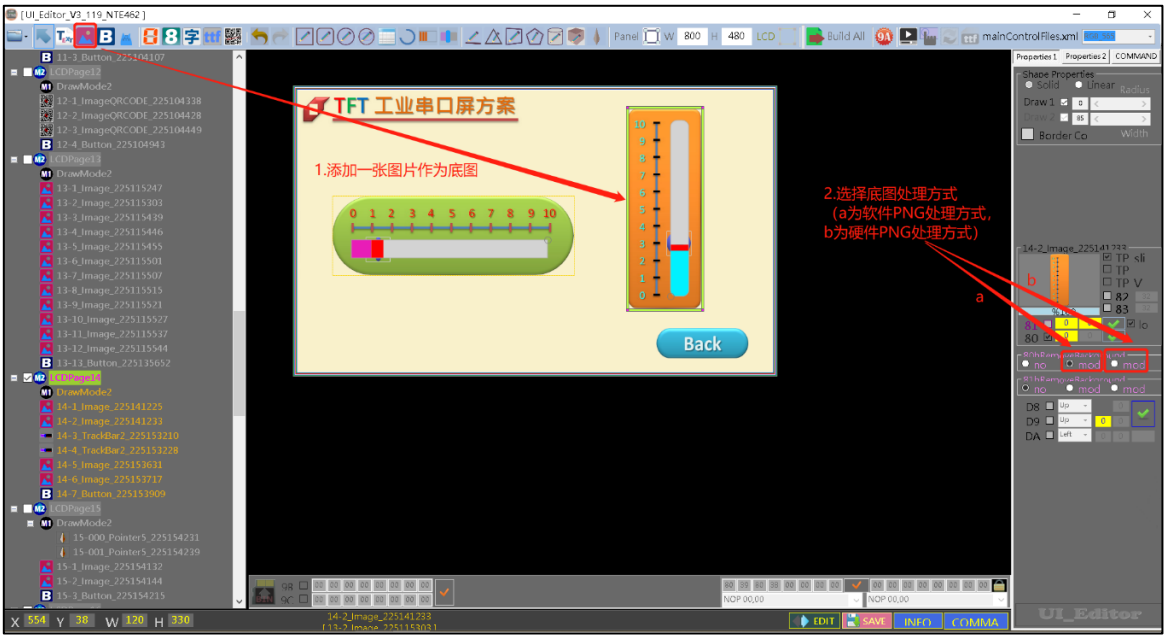


图 3-44：设定滑动条背景图片

© Levetop Semiconductor Co., Ltd.

4. 再新增一张图片当作滑动条的移动图标头，如下图所示，并且也要将图片设定为去除边框透明功能。

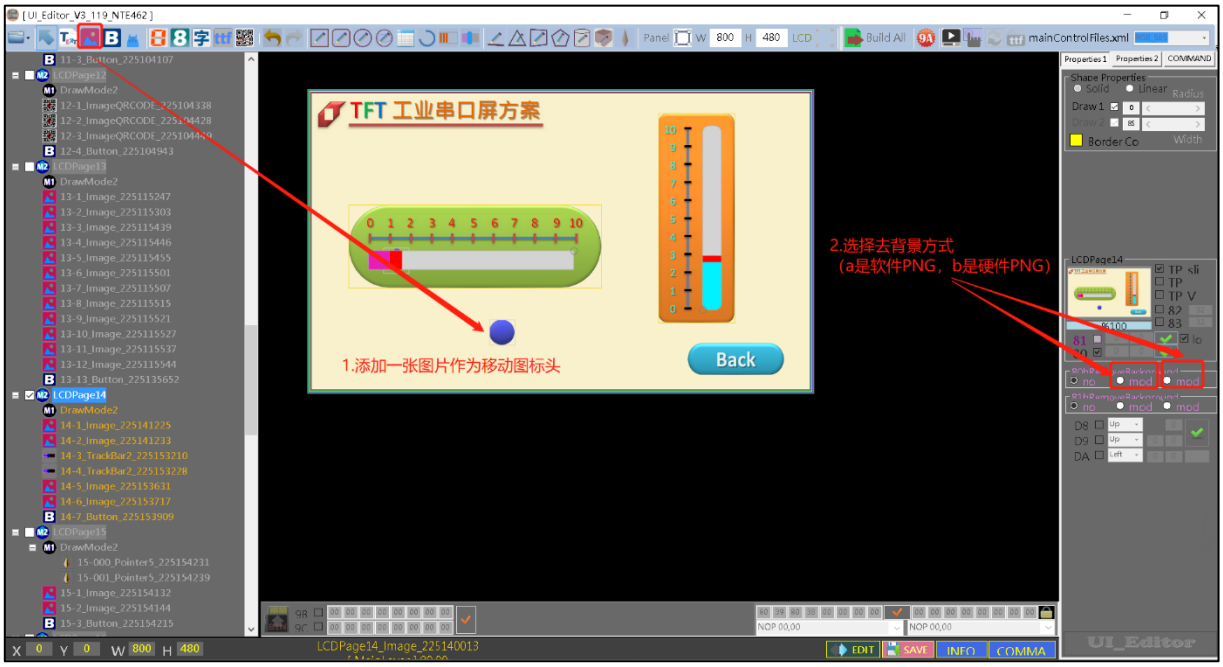


图 3-45：新增滑动条的移动图标头

根据用户需求将滑动条与背景图片与滑动用图标头各自对齐，如下图参考。

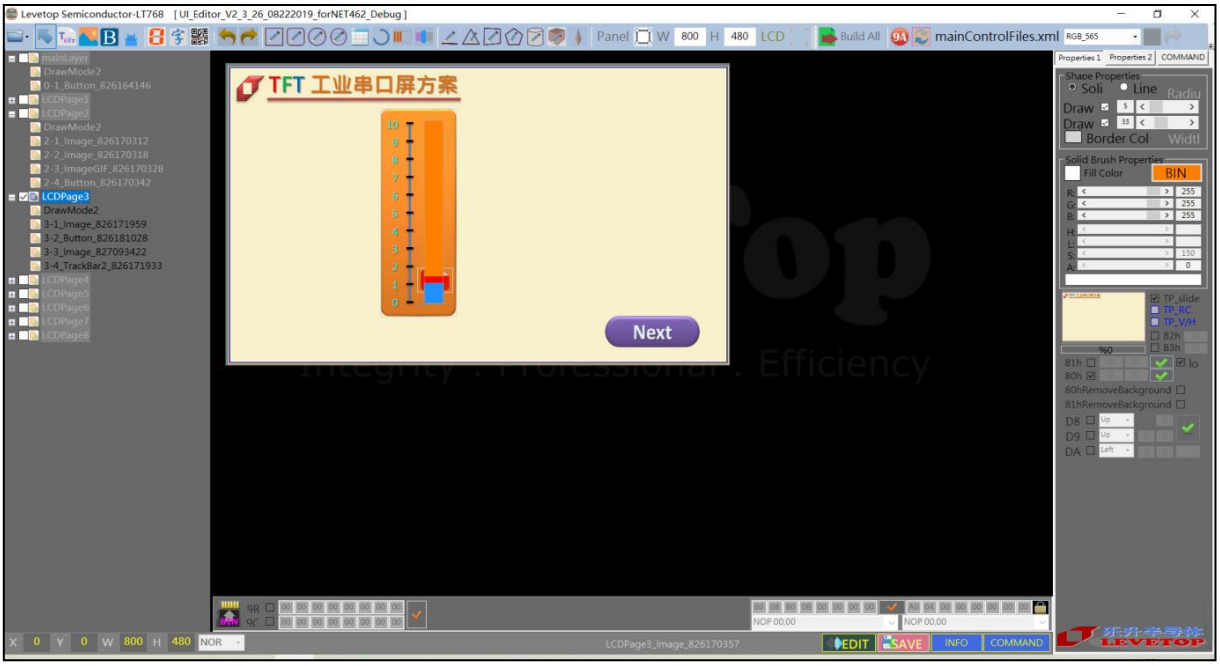


图 3-46：背景图片与滑动图标头各自对齐

© Levetop Semiconductor Co., Ltd.

5. 选择滑动条背景图片 Command 与滑动图标 Command，请先点选滑动条以确保是要设定滑动条背景图片与移动图标模式，接着点选屏幕下方 Command 选择窗口，如下图所示，

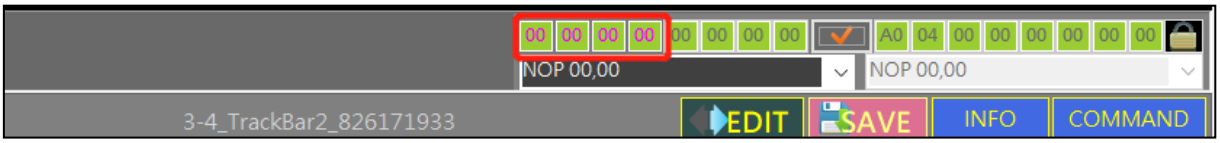


图 3-47：设定滑动条背景图片与移动图标模式

Command 选择窗口，共分两组，第一组是背景图片 Command，第二组是滑动图标头 Command。如下图所示，只要分别点选再展开 Command 列表，再选择需要的显示图片编号即可。

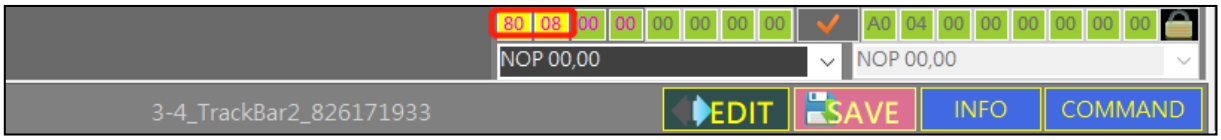


图 3-48：背景图片 Command 选择窗口

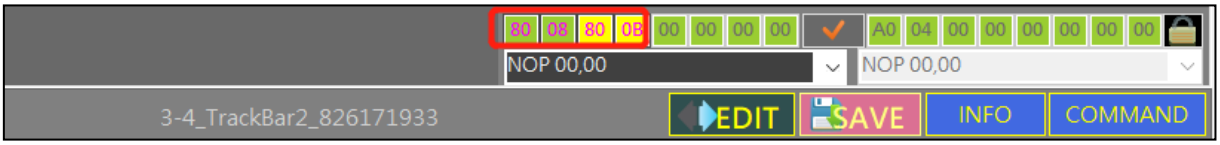


图 3-49：滑动图标头 Command 选择窗口



图 3-50：选择完毕后点击确认

© Levetop Semiconductor Co., Ltd.

3.2.11 设定环形触摸进度条

此功能是结合触控屏的滑动，来改变 TFT 屏上所显示的环形触摸进度的大小，同时反馈信息给主控端。触控进度条底图和触控图标可设置 80 Removebackground，触控图标必须设置在环形触控进度条底图的中心正上分处，设置完成后除了生成对应的显示环形触控滑桥 96h 指令，还会生成对应的移除触控滑桥 97h 指令。

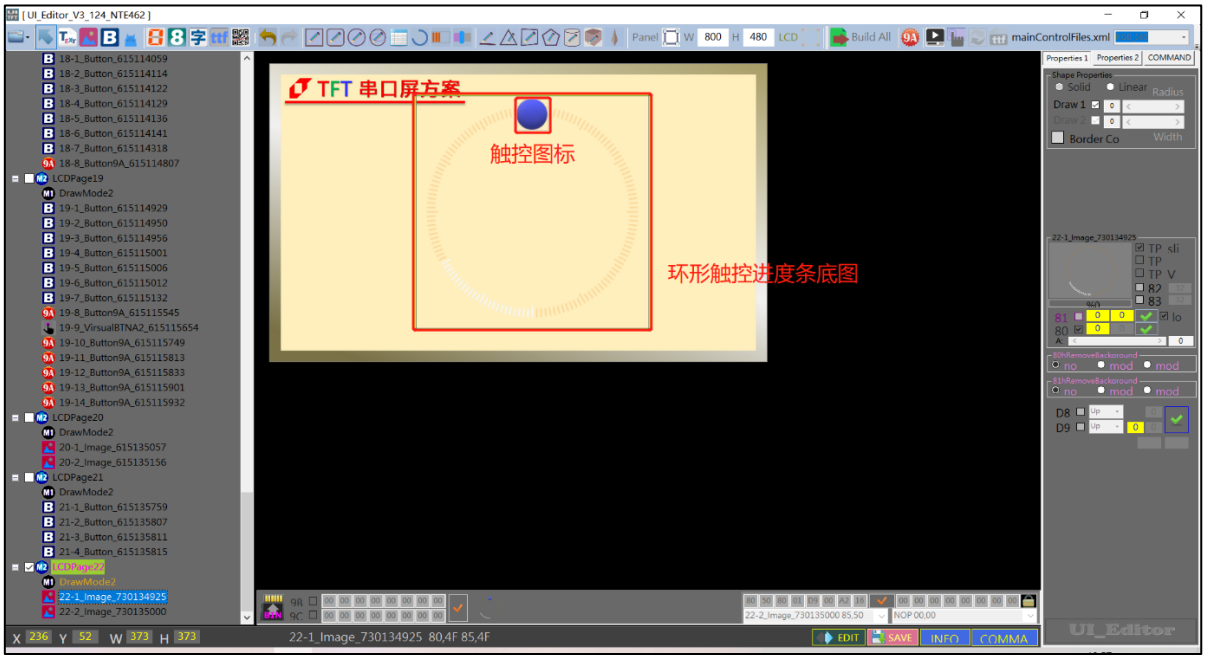


图 3-51：添加环形触控进度条底图和触控图标

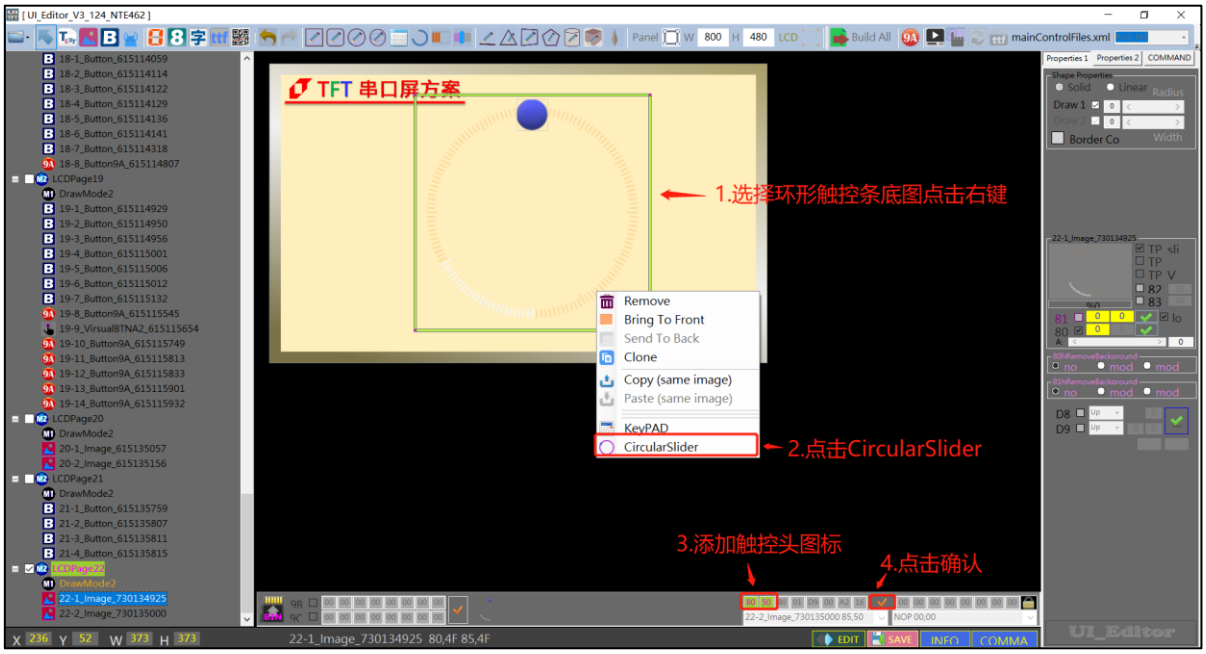


图 3-52：设定环形进度条

© Levetop Semiconductor Co., Ltd.

3.2.12 设定指针图

此功能是设计一个指针图，由主控端传来信息改变指针显示的方向及角度：

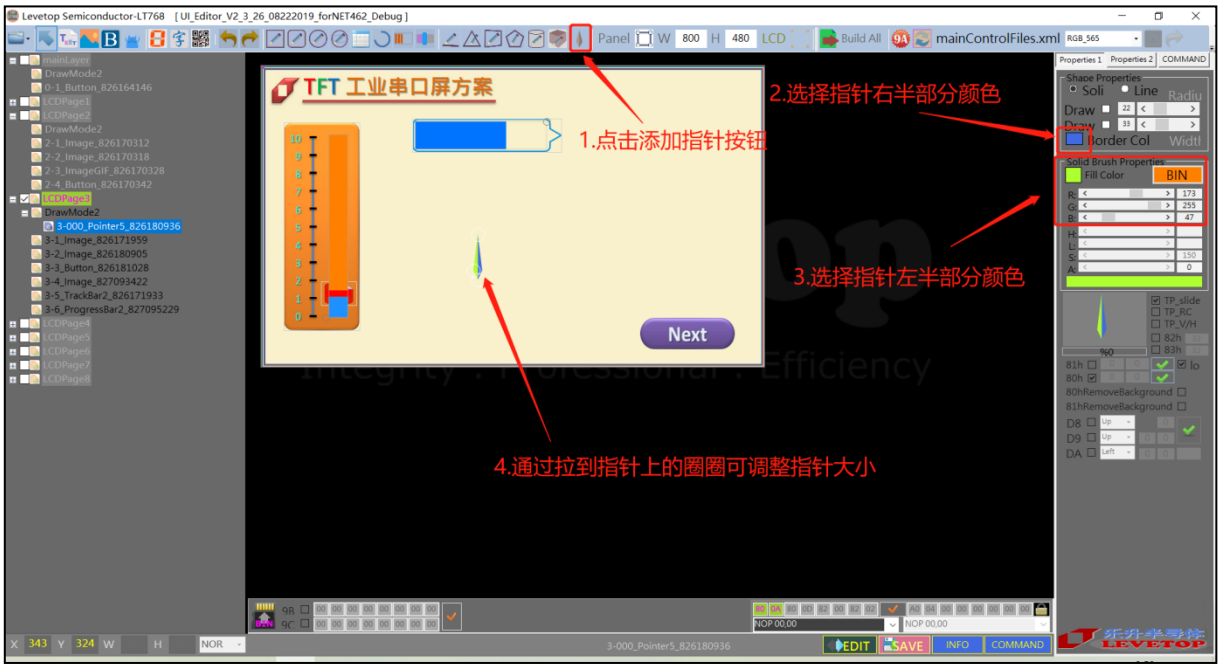


图 3-53：设定指针对左右两侧的背景颜色

1. 点选指针图标，并在屏幕任意位置拖拉移动显示出指针，此时可针对功能需求透过调色盘调整指针左右两侧的背景颜色。

2. 指针上有三个圆圈，当鼠标点选任一点圆圈时即可移动并改变指针尺寸大小。若是鼠标点选至三个圆圈点区域的中间时，鼠标图标会改变成手掌图标，此时表示指针是可以移动的。但若是点选指针都没有任何反应，请先点选左上角树状结构目录上“mainLayer”下的“DrawMode2”，再点选预计选择的指针即可。(EX :3-000_Pointer5_826180936)

© Levetop Semiconductor Co., Ltd.

3. 添加指针背景图，请先点选图标并在屏幕拖拉出一张图标外框，此时 UI_Editor 会自动跳出图片选择视窗，请再点选背景图片即可。如下图所示：



图 3-54：添加指针背景图片

4. 针对功能需求可将背景图片与指针移动重叠显示出仪表的外观，指针也可适时重设大小与颜色，如下图所示：

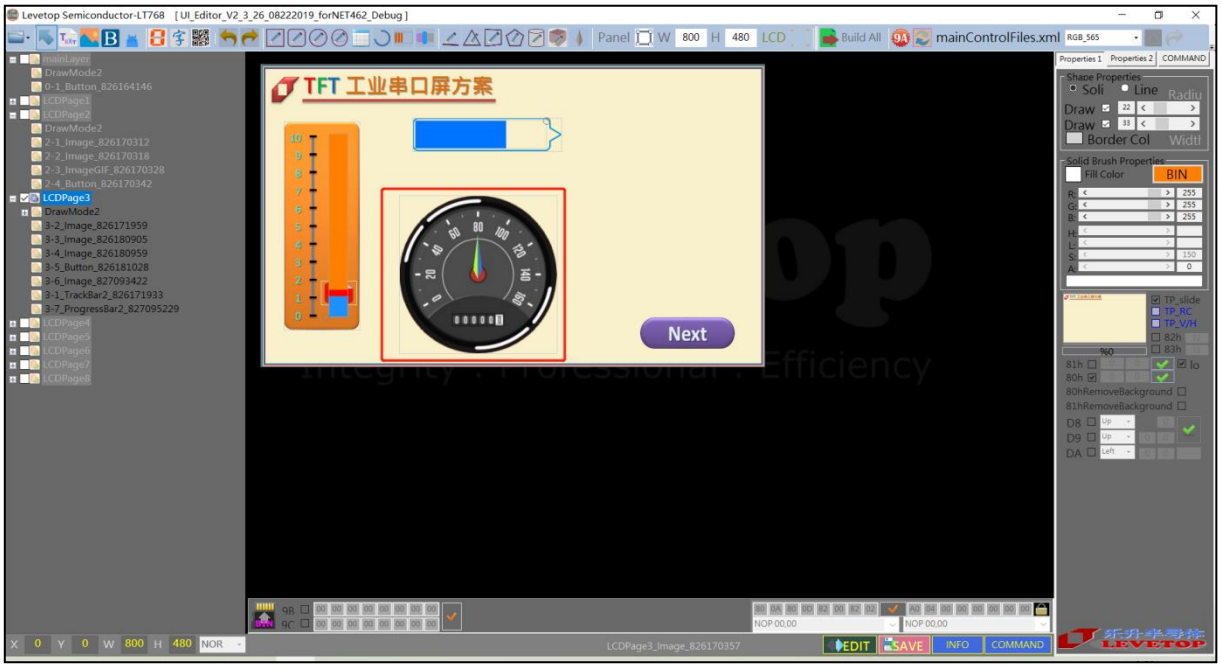


图 3-55：调整指针背景图片显示位置

© Levetop Semiconductor Co., Ltd.

5. 选择指针背景图片 Command，请先点选指针以确保是要设定指针背景图片模式：



图 3-56：选择指针背景图片 Command

点选第一组数据（00 00 任意点选一个，数据背景变成黄色）



图 3-57：点选指针背景图片模式第一组数据

接着点选屏幕下方 Command 选择窗口列表，如下图所示，再展开 Command 列表：



图 3-58：为指针添加背景图片指令

© Levetop Semiconductor Co., Ltd.

3.2.13 设定进度条图

此进度条功能与指针图类似，由主控端传来信息改变进度条显示的长短：

1. 点选进度条图标，并在屏幕任意位置拖拉移动显示出进度条，此时可针对功能需求透过调色盘调整进度条背景与进度条移动时的颜色。



图 3-59：设定进度条背景与进度条移动时的颜色

2. 添加进度条背景图片，如下图所示:



图 3-60：设定背景图片

© Levetop Semiconductor Co., Ltd.

3. 请针对功能需求将进度条与背景图片进行对齐调整，如下图所示：

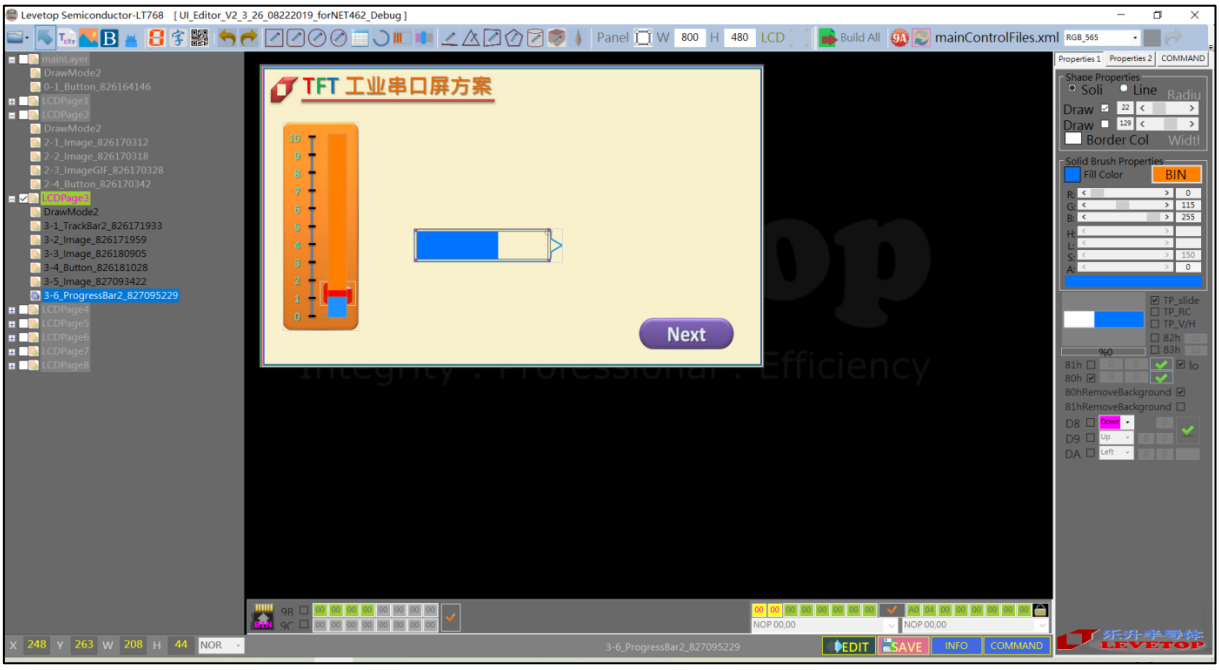


图 3-61：调整进度条背景图片显示位置

4. 选择进度条背景图片 Command，请先点选进度条以确保是要设定进度条背景图片模式，



图 3-62：选择进度条背景图片 Command

点选第一组数据（00 00 任意点选一个，数据背景变成黄色）



图 3-63：点选进度条背景图片模式数据

© Levetop Semiconductor Co., Ltd.

接着点选屏幕下方 Command 选择窗口，如下图所示，再展开 Command 列表：

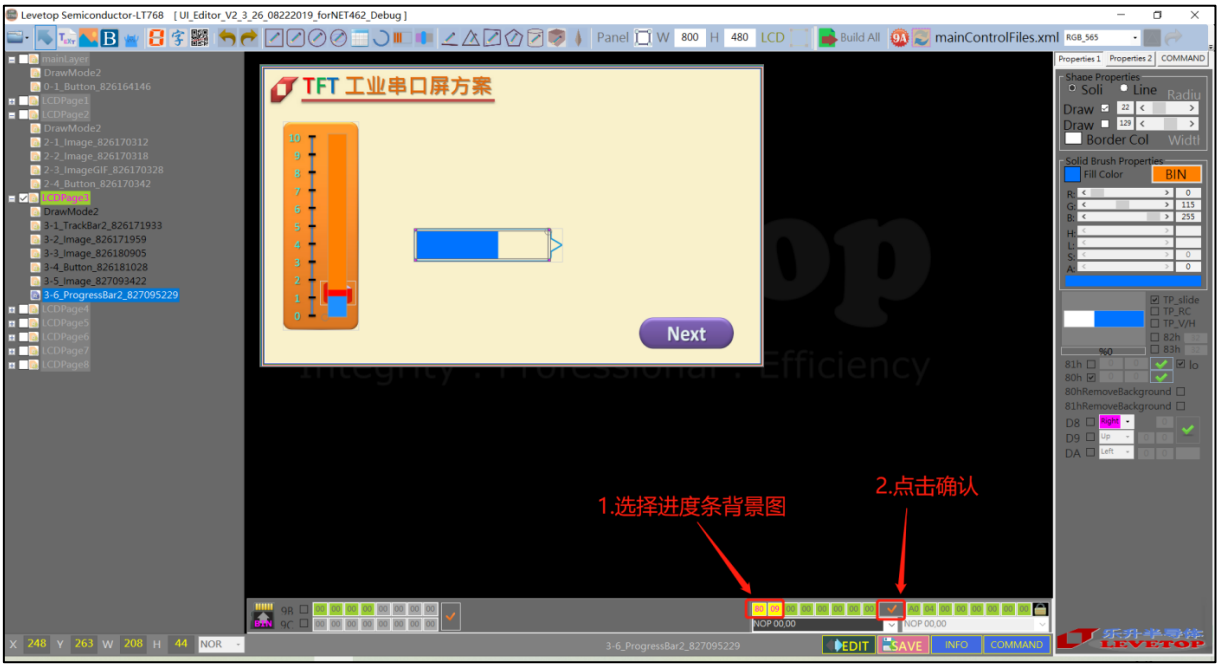


图 3-64：选择进度图的背景图片

© Levetop Semiconductor Co., Ltd.

3.2.14 设定带透明度的单张图片

成功添加图片之后，可以看到右边操作栏中有 82h 指令的功能框，勾选 82h 指令后，点击右边的小框，接着就可以根据需求来调节透明度大小。



图 3-65：设定带透明度的单张图片

© Levetop Semiconductor Co., Ltd.

3.2.15 控件滑动及显示

成功添加控件之后，勾选左下方 9Bh 指令的小方框， 双击小方框右侧的指令栏，根据需求选择相应的指令，9Bh 设定最多可执行 2 个指令。放在同一个图层下的控件则视为处于同一个页，底图只能显示主图层的图片，最后要实现控件滑动的功能，只需要发送 9Ch 的指令即可。



图 3-66：控件滑动及显示

© Levetop Semiconductor Co., Ltd.

3.2.16 设定显示数字键盘

数字键盘是由用户交互界面输入信息去反馈给上位机，上位机可以通过反馈信息去执行一些约定的内容，例如开锁等操作。该数字键盘最多可以设置 30 个键盘（包括三个功能键 0x0D：return; 0x08:BackSpace; 0x18:Cancel）每个密码键盘都必须带有三个功能键。用户需要设置最多显示的数字个数，各按键的位置与代表的键值等参数。

添加两张大小和位置一样的图片，一张用于作为数字键盘显示，一张用于数字键盘被按下时在设定的按键区域显示，设置完成后会生成显示密码输入界面 A4 指令和对应取消指令 A5 指令。



图 3-67：添加数字键盘底图

© Levetop Semiconductor Co., Ltd.

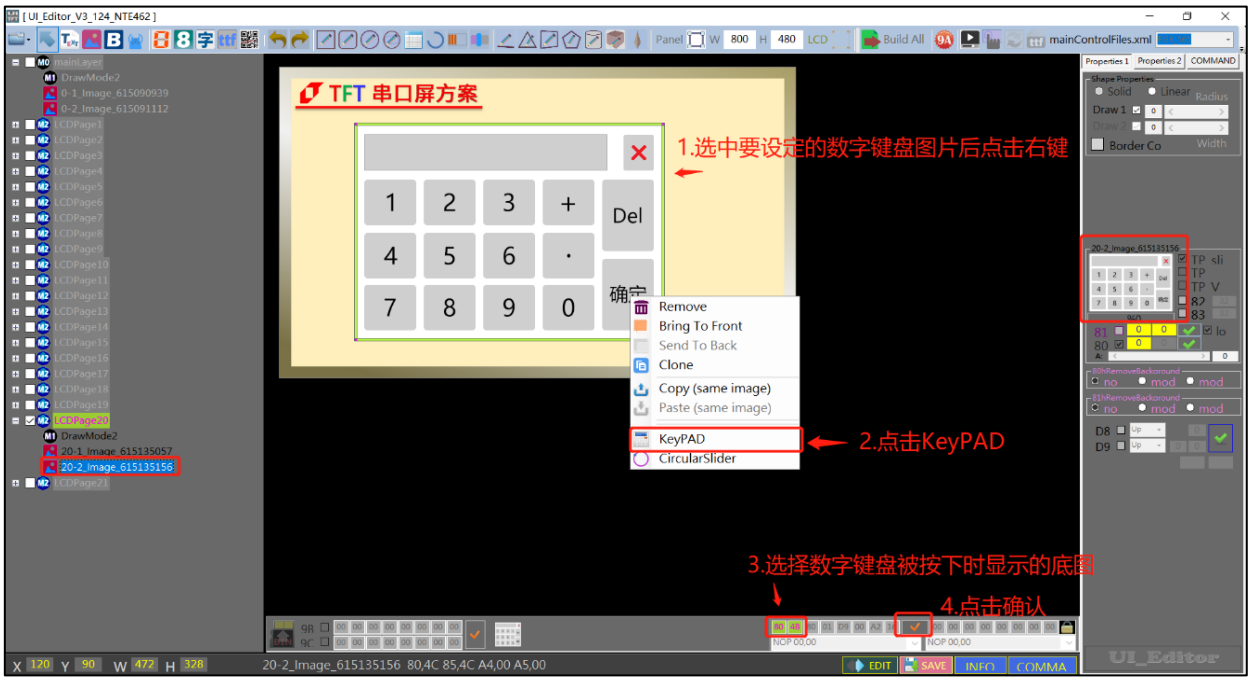


图 3-68：添加数字键盘底图



图 3-69：数字键盘设定

© Levetop Semiconductor Co., Ltd.



图 3-70：数字键盘设定

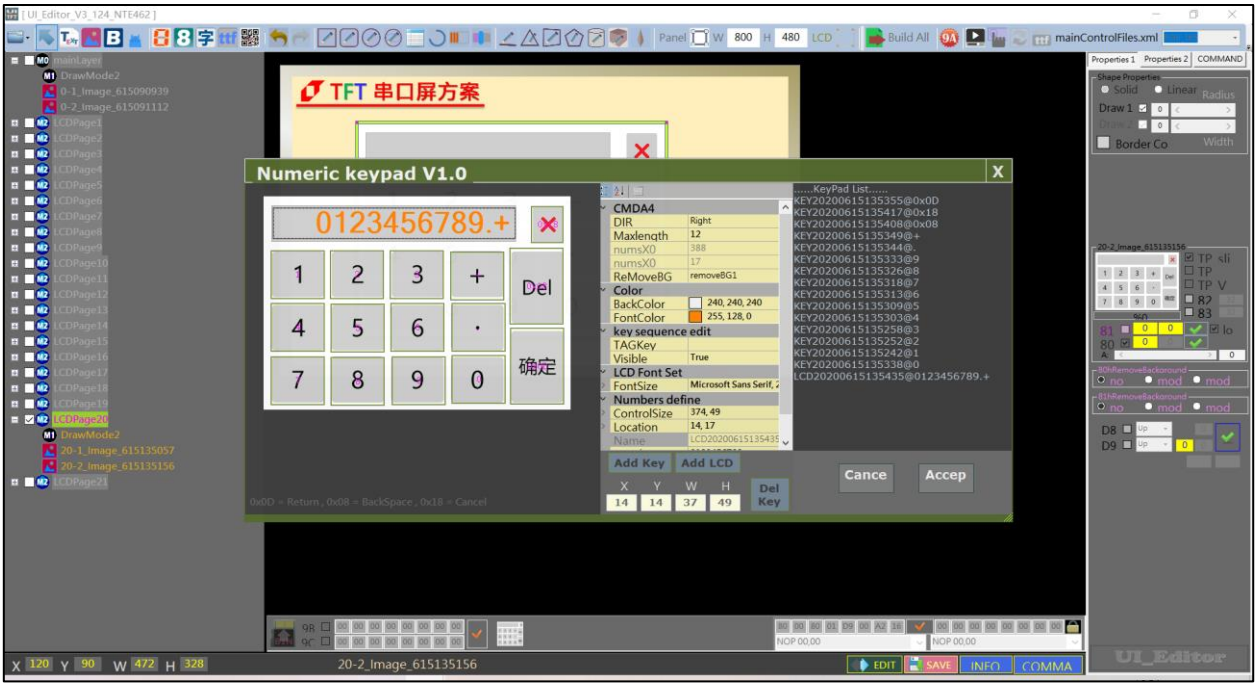


图 3-71：数字键盘设定

© Levetop Semiconductor Co., Ltd.

3.2.17 实时曲线的设定

B2h 及 B3h 指令可实现实时曲线绘制，先由上位机设置好曲线显示的区域及相关参数生成 B3h 指令，再由下位机通过 B2h 指令发送数据进行绘线显示。B2h 发送的指令格式为 B2+Chanel(0-3)(1byte)+Data(N*2bytes)，曲线数据会更新到对应信道的 B3h 显示。

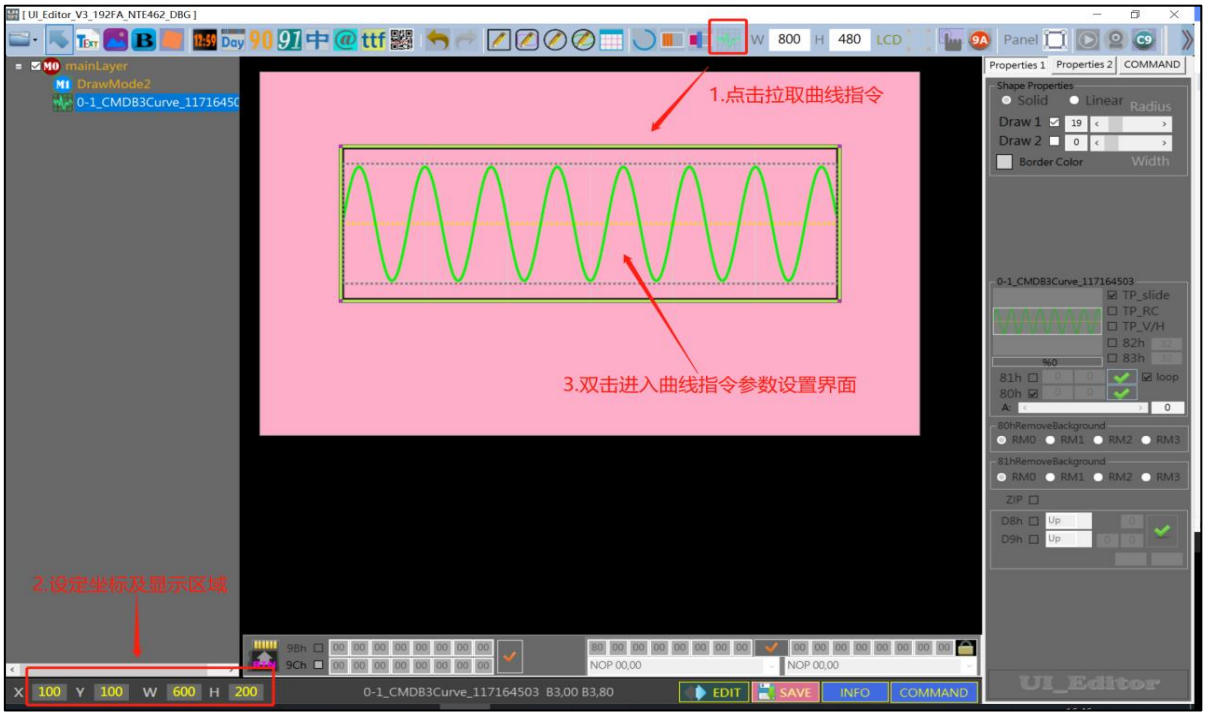


图 3-72：曲线指令设定

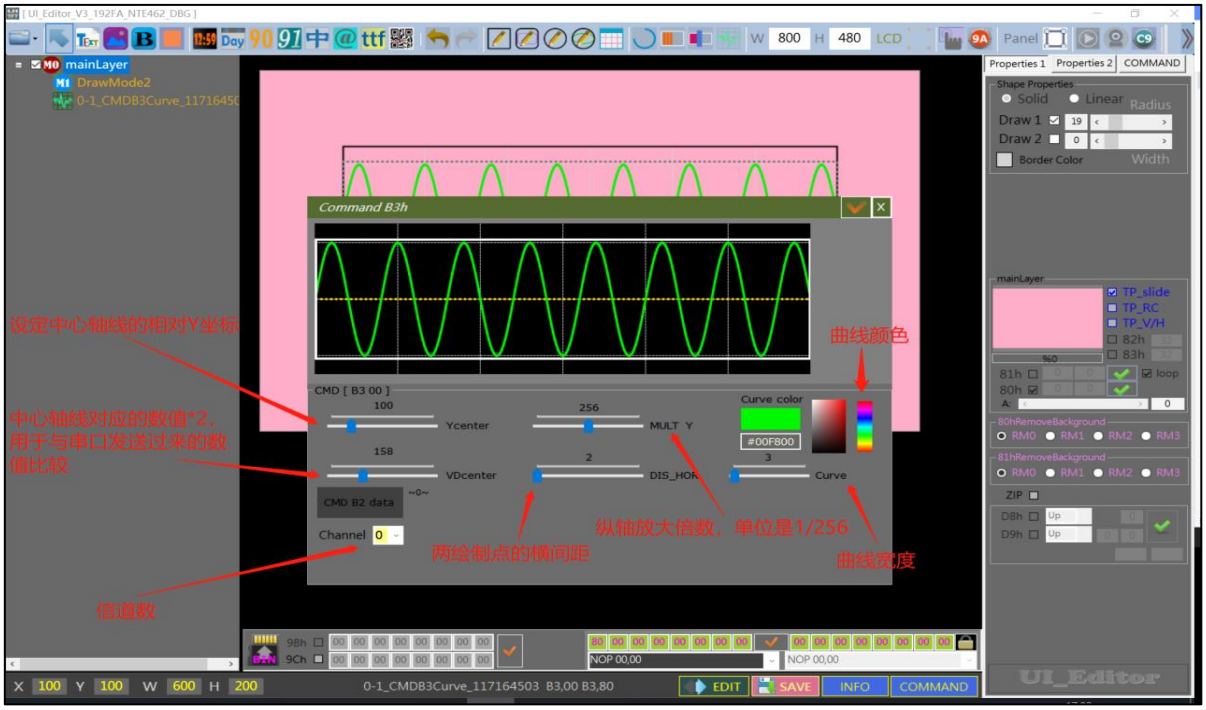


图 3-73：曲线参数设定

© Levetop Semiconductor Co., Ltd.

3.2.18 设定环形指标图

DC 指令可实现显示环状指标图，先由上位机选中环形指标图工具，在编辑界面下拉所需大小，设置好显示区域和相关参数，再由下位机通过 DCh 指令发送数据进行显示，显示的角度为（0，360）任意弧度，发送的角度都是 2 个 Byte 的十六进制单位。

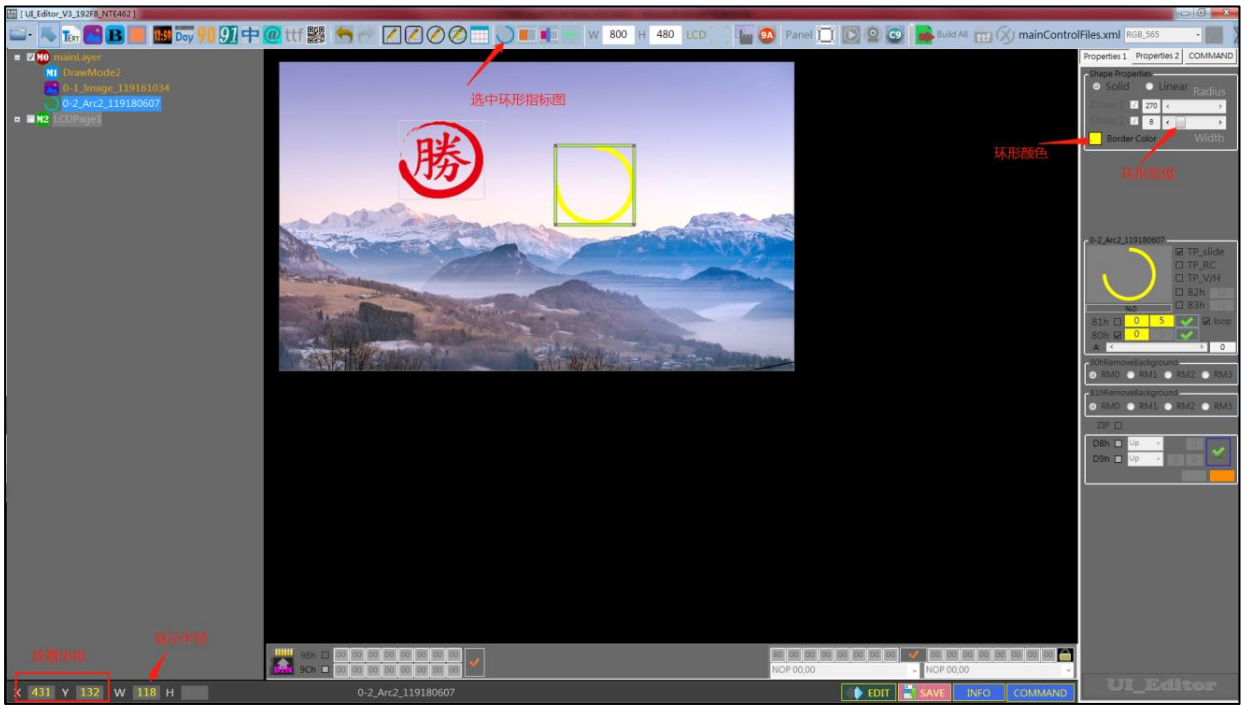


图 3-74：环状指标图参数设定

© Levetop Semiconductor Co., Ltd.

3.2.19 固定图片显示指令

固定图片显示指令只需要透过 UART 或是 SPI 传递给 TFT 串口屏就会执行，它是主控端系统或是主板的固定指令，这指令不需要在上位机软件或是编程软件设定。主控端可以直接使用 8F 固定指令传递给 TFT 串口屏直接将指定的图号显示出来，显示的参数都可以由串口更新。

表 3-1：固定图片显示指令

指令功能	指令码	序号	指令参数	指令说明
直接显示单张图片	**8Fh**	**nn**	X(2), Y(2), PNG(1), Pnn(2)	将编号为 nn 的图片 P 显示在（X，Y）位置。PNG=0 则代表背景为不透明，PNG=1 代表背景为透明（软件 Png 图档），PNG=2 代表背景为透明（硬件 Png 图档），PNG=3 代表使用 Jpg 软解模式。

例如主控端的系统或是主板透过 UART 串口传递显示指令 Start(1Byte) + 8Fh + 00 + 0064h(2Bytes) + 0032h(2Bytes) + 00h(1Bytes) + 0001h(2Bytes) + (CRC(2Bytes) + End(4Bytes) 给 TFT 串口屏，那么 TFT 上就会在（100, 50）显示第 1 张背景为不透明的图片。

传递显示指令 Start(1Byte) + 8Fh + 00 + 0064h(2Bytes) + 00F0h(2Bytes) + 01h(1Bytes) + 0002h(2Bytes) + (CRC(2Bytes) + End(4Bytes) 给 TFT 串口屏，那么 TFT 上就会在（100, 240）显示第 2 张具有 Png 格式的图片。

© Levetop Semiconductor Co., Ltd.

3.2.20 摄像头显示指令

LT776 支持 DCMI 摄像头接口，可以透过摄像头显示指令开启和关闭摄像头的画面。可以直接上位机选中摄像头工具然后在编辑界面内下拉，最后设置显示摄像头的参数即可。目前支持的摄像头型号为 GC0308 和 OV5640，摄像头型号为 OV5640 的可支持任意大小画面的显示，摄像头型号为 GC0308 的只能支持指定大小画面显示。

```
AEh #00:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 92, 69
AEh #01:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 107, 80
AEh #02:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 128, 96
AEh #03:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 160, 120
AEh #04:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 184, 138
AEh #05:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 214, 160
AEh #06:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 256, 192
AEh #07:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 275, 206
AEh #08:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 320, 240
AEh #09:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 366, 274
AEh #0A:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 384, 287
AEh #0B:  0, 0, 128, 128, 128, 0, 2, 4, 80, 80, 427, 319
AEh #0C:  0, 0, 128, 128, 128, 0, 2, 4, 0, 0, 640, 480
AEh #0D:  0, 0, 128, 128, 128, 0, 2, 2, 0, 0, 640, 480
AEh #0E:  0, 0, 128, 128, 128, 0, 2, 0, 0, 0, 640, 480
```

图 3-75：GC0308 支持摄像头

表 3-2：摄像头显示指令

指令功能	指令码	序号	指令参数	指令说明
设定摄像头	**AEh**	**#nn:**	0, 0, 128, 128, 128, 0, 2, Dir(1), X(2), Y(2), Width(2), Height(2),	**摄像头画面**在（X, Y）位置显示；Dir 代表是上下左右出现方向 → 0：正常；2：镜像；4：180 度旋转。位置宽度为 Width；高度为 Height。其他为固定参数。
取消摄像头显示	**AFh**	**nn**		将 D9h 显示循环卷动的图片取消

© Levetop Semiconductor Co., Ltd.

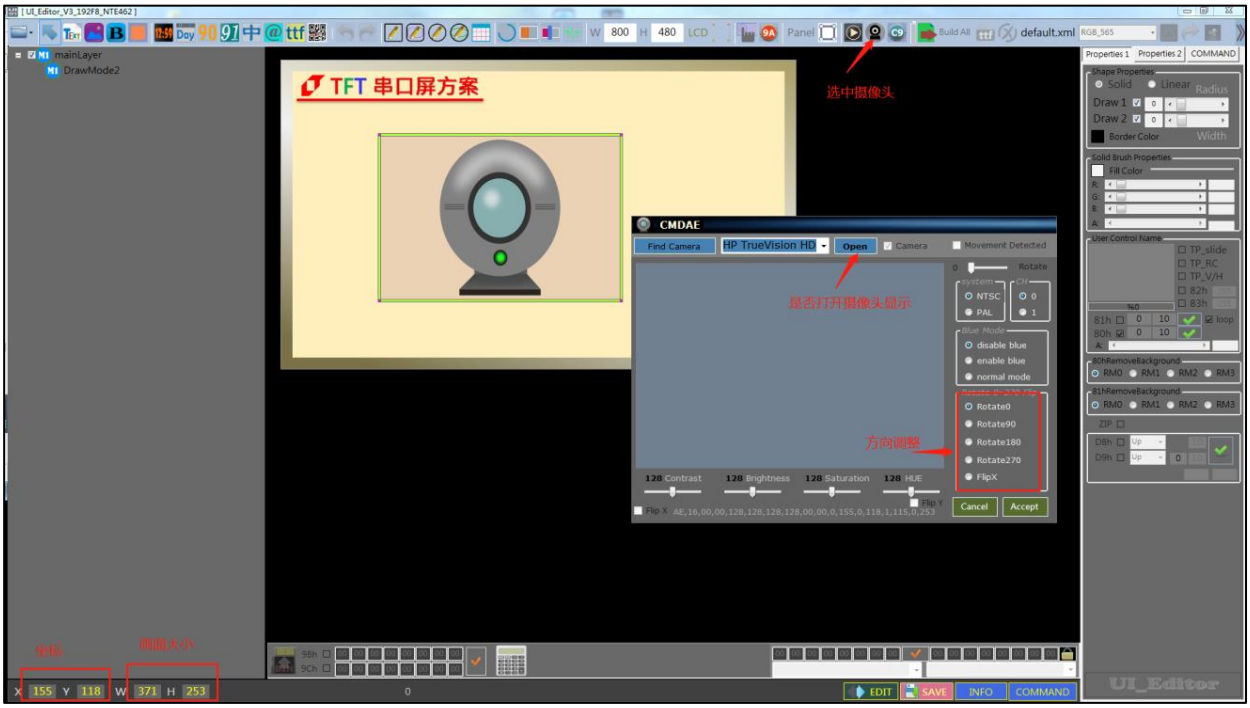


图 3-75：摄像头显示操作

在编译完成后，主控端的系统或是主板透过 UART 串口传递显示指令 AEh 给 TFT 串口屏，那么 TFT 上就会按照设定参数显示对应摄像头画面，传递显示指令 AFh 给 TFT 串口屏，就会取消显示对应摄像头画面。注意：设置的画面大小如果超过 TFT 显示区域，则超出的部分不会显示出来。

© Levetop Semiconductor Co., Ltd.

3.3 显示文字的设定

3.3.1 图像形式显示文字

图片式显示的意思是用图片来显示文字，处理的对象是图片。是在 UI_Editor 上将文字生成为图片，在 TFT 串口屏中以图片形式显示文字。这里的文字是指电脑上可以输入的一切文字，包括中文、英文、数字和符号。这种文字显示只适用于显示固定不变的文字。

点击 UI_Editor 中添加  文字的按钮 ，是在屏幕框中拉一个框，松开按键后会弹出一个设置框。在框中输入要显示的文字。可以根据需要选择心仪的字体和字号，可以改变字体和背景的颜色，还可以选择背景是否透明。注意：选择了背景透明（BackColor 勾选代表不透明，不勾选代表透明），则背景颜色设置失效。确认设置后，要调节显示框的大小使全部文字得以显示。

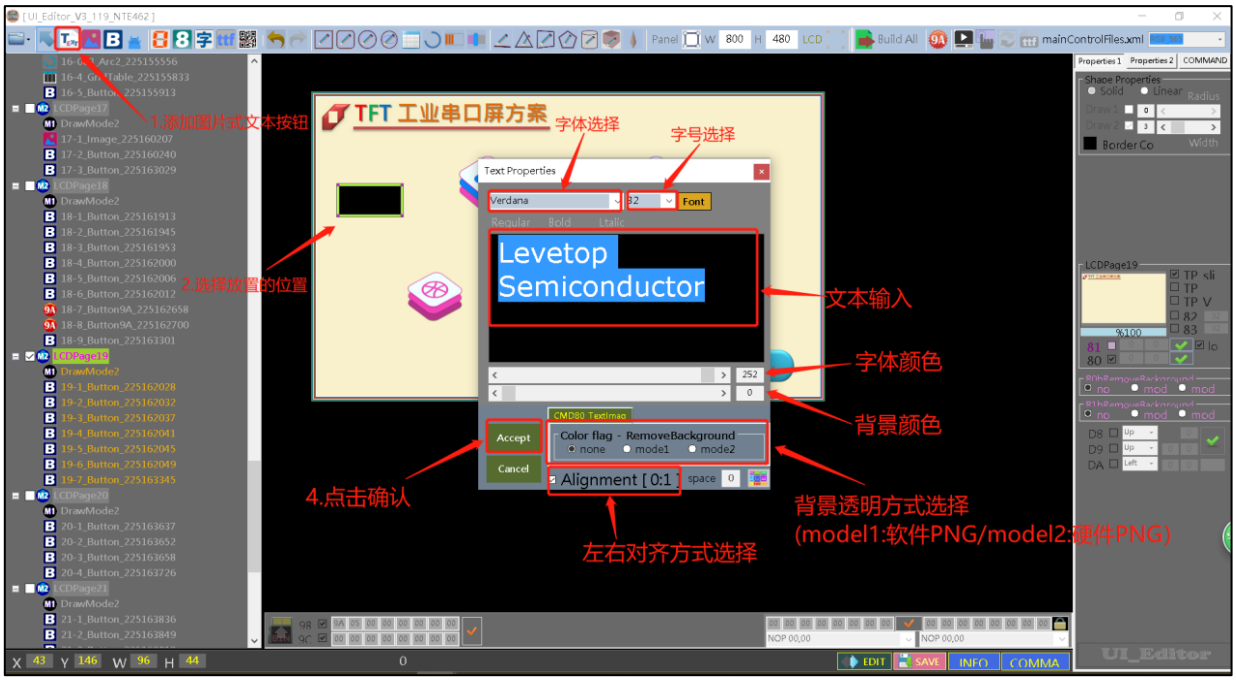


图 3-76：图片形式显示文字

© Levetop Semiconductor Co., Ltd.

3.3.2 图片式的数字显示

图片式的数字显示是指取模电脑系统的字体和大小生成数字图片，根据串口接收到的数据，分析得到应该显示的数字。所以这种数字显示具有多字体和动态显示的优点。

点击 UI_Editor 中的  按钮。在屏幕框中拉一个框，松开按键同样会弹出一个设置框。这个设置框和上面的提到的设置框有不同。这个设置框中的文字不能改变，同时可以设置数字的对齐方式（左对齐或者右对齐）。Alignment 勾选代表左对齐，不勾选代表右对齐。



图 3-77：图片式的数字显示的设置窗口

© Levetop Semiconductor Co., Ltd.

确认设置后，要调整文字框到刚好显示一个数字。如下图所示，每一个小框代表一个数字所占用的面积大小，只需要调整第一个数字框的大小，其他数字框也会跟着改变。在生成 BIN 文件后，用串口和 TFT 串口屏通信的时发送要显示的数字就可以在 TFT 串口屏屏幕中显示出来。



图 3-78：调整图片式数字显示的字体框

© Levetop Semiconductor Co., Ltd.

3.3.3 真彩图片式数字显示

真彩图片式的数字显示是指利用提前制作好“0”～“9”和“.”及“,” 的真彩数字图片，根据串口接收数据，分析得到应该显示的数字，所以这种数字显示具有多字体和动态显示的优点。真彩图片存放在 UI_Editor\ColorNumber 文件内，可以由用户去替换真彩图片。

点击 UI_Editor 中的  按钮。在屏幕框中拉一个框，该框要与真彩数字图片大小一致，松开按键同样会弹出一个设置框，该可以设置数字图片的来源、数字图片对齐方式（左对齐或者右对齐）和图片背景透明。Alignment 不勾选代表左对齐，勾选代表右对齐，BackColor 勾选表示透明背景，不勾选表示不透明。

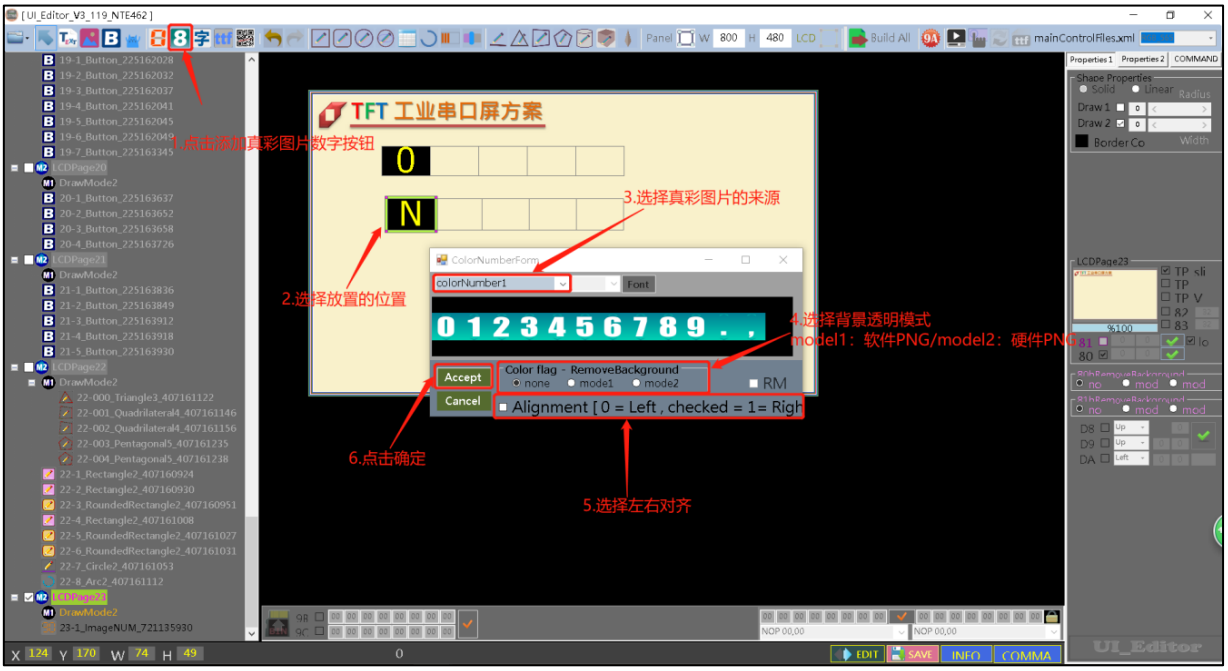


图 3-79：调整真彩图片式数字显示的字体框

© Levetop Semiconductor Co., Ltd.

3.3.4 使用字库显示文字

动态的中文、英文数字显示是利用内部或者 Flash 里的字库直接显示。最多可以选择 8 种字库，其中 C0-C3 字库可以显示数字、英文和中文，D0-D1 字库只能显示中文。



图 3-80：字库显示文字设置

点击文字（字库）添加按钮，点击一下屏幕框，会弹出设置框。框中有相关的各种设置，C0 到 C3 字库的字可以在水平方向和垂直方向分别放大 1 到 4 倍，而 D0 到 D1 的字库不可以。D0 到 D1 的字库可以加粗字体 1 到 2 倍，C0 到 C3 的字库不可以加粗字体。背景颜色、字体颜色、字体对齐等设置都和上面的设置方法一样。

背景透明的设置则和前面的不同，Transparency 被勾选了表示背景透明，不勾选代表背景不透明。设置完成后，文字框中小框的大小不能被改变。在生成 BIN 文件后，用串口和 TFT 串口屏通信的时发送要显示的文字就可以在 TFT 串口屏屏幕中显示出来。

© Levetop Semiconductor Co., Ltd.

3.3.5 图片式的 ASCII 显示

首先用我司开发的取模软件 ASCIIBitmap_Font_Generator_Vxx.exe 取模一组 ASCII 图片，选择字体类型，字体大小，显示方式，随后点击 BIN GEN 进行取模。取模后生成一组 ASCFONT_xx 的文件夹，然后把该文件夹复制到 UI_Editor_XXX\ASCIIFONT 目录下，按照指定格式修改文件夹命名。以上操作完成后在 UI_Editor 引用即可。

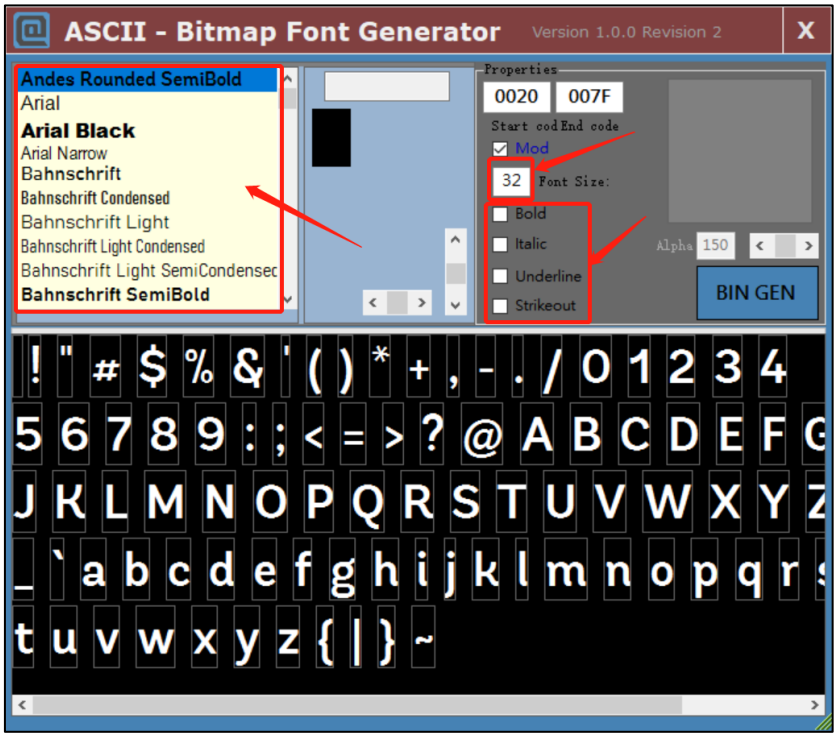


图 3-81：图片式的 ASCII 显示设置

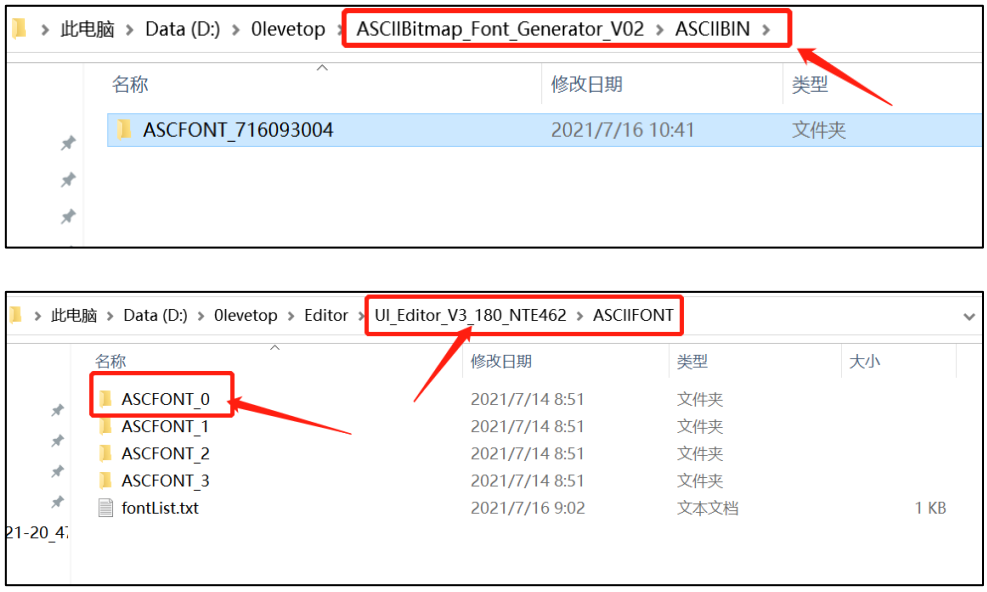


图 3-82：图片式的 ASCII 显示设置

© Levetop Semiconductor Co., Ltd.

点击添加 ASCII 显示控件，在弹出的窗口选取一组 ASCII，设置对齐方式，显示颜色，最后点击 Accep 保存即可。

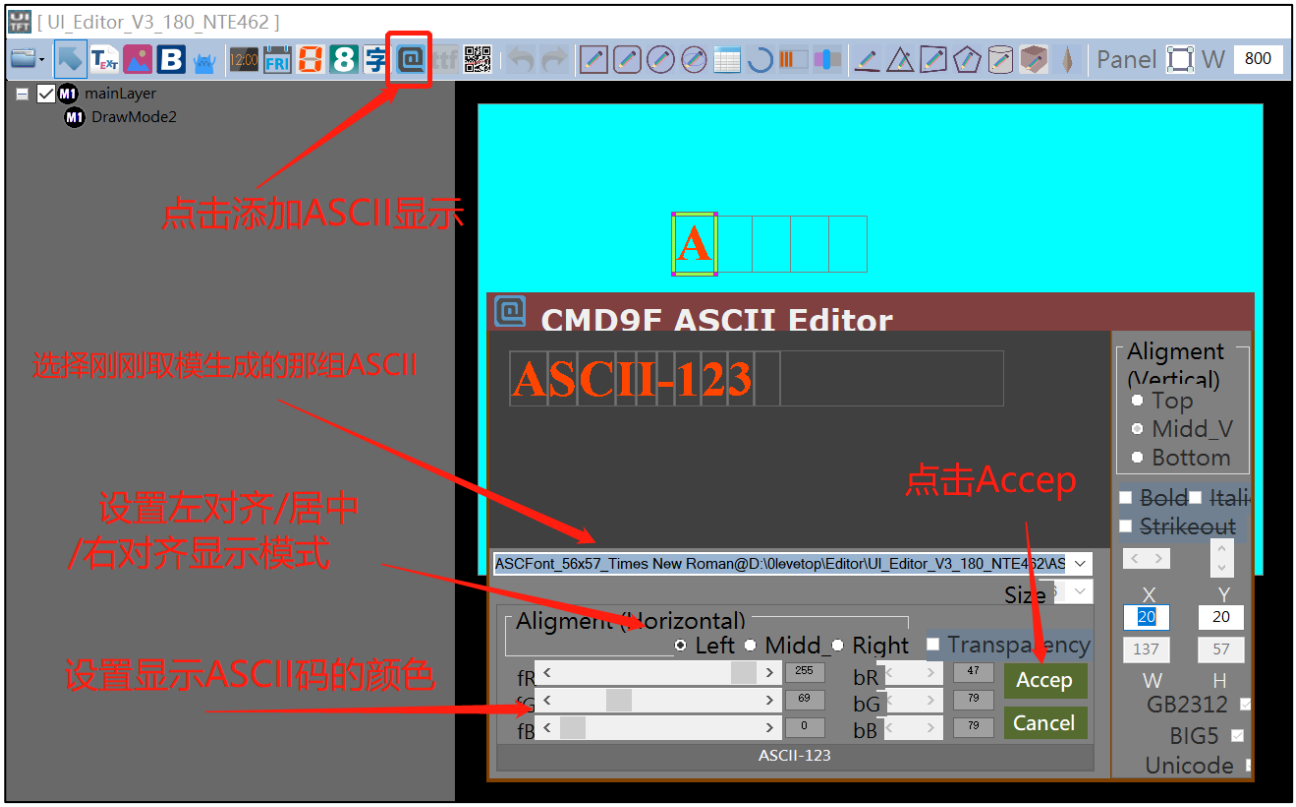


图 3-83：图片式的 ASCII 显示设置

© Levetop Semiconductor Co., Ltd.

3.4 几何绘图功能

3.4.1 基本绘图功能

UI_Editor 一共包含 11 种几何绘图功能，某些是 TFT 串口控制芯片具有的硬件功能，其中 9 种类似的功能可以分为 3 类，如下表所示。

表 3-3：几何绘图功能

	空心、实心、带框实心	空心、实心	带框实心
圆	✓		
椭圆	✓		
矩形	✓		
圆角矩形	✓		
三角	✓		
四边形		✓	
五边形		✓	
圆柱体			✓
方柱体			✓

另外两种功能是画直线和画任意角度的圆环。由于同一类的绘图功能操作基本一致，所以每一类功能只拿一个例子来举例说明。

画圆：点击画圆按钮，在屏幕框中点击并拉动操作框，即可出现圆形。调整外面的操作框可以改变圆的大小。其他的设置都是右边的 Properties1 设置栏中。



图 3-84：画圆设置

© Levetop Semiconductor Co., Ltd.

其中 Draw1 和 Draw2 的具体作用如下：

Draw1： 空心、实心的选择框，Draw2 没有被勾选的前提下，勾选 Draw1 代表选择空心功能，不勾选 Draw1 代表选择实心功能。 如果 Draw2 被勾选了，Draw1 将不受控。

Draw2：带框实心的选择框。勾选上则选择带框实心功能，不受 Draw1 控制。不勾选则选择空心、实心功能。只有在带框功能下才可以调节外边框的粗细。

要改变外边框的颜色，只需要点击外边框颜色选项，在其中选择一种心仪的颜色。要调整内部颜色，可以改变颜色调节框中 RGB 三色的数值，选一种喜欢的颜色，最后要按内部颜色确定按钮。画椭圆、画矩形、画圆角矩形、画三角形的操作和画圆是大体类似的，其中调整三角形的大小是通过拉动三角形的三个顶点来实现的。画四边形，先点击四边形添加按钮，在屏幕框中点击一下，就可以产生一个四边形。通过改变四边形四个角的位置可以更改四边形的形状。拖动四边形可以改变位置。假如无法选中四边形，可以在左边的任务栏中单击对应的任务选项，即可成功选中四边形。

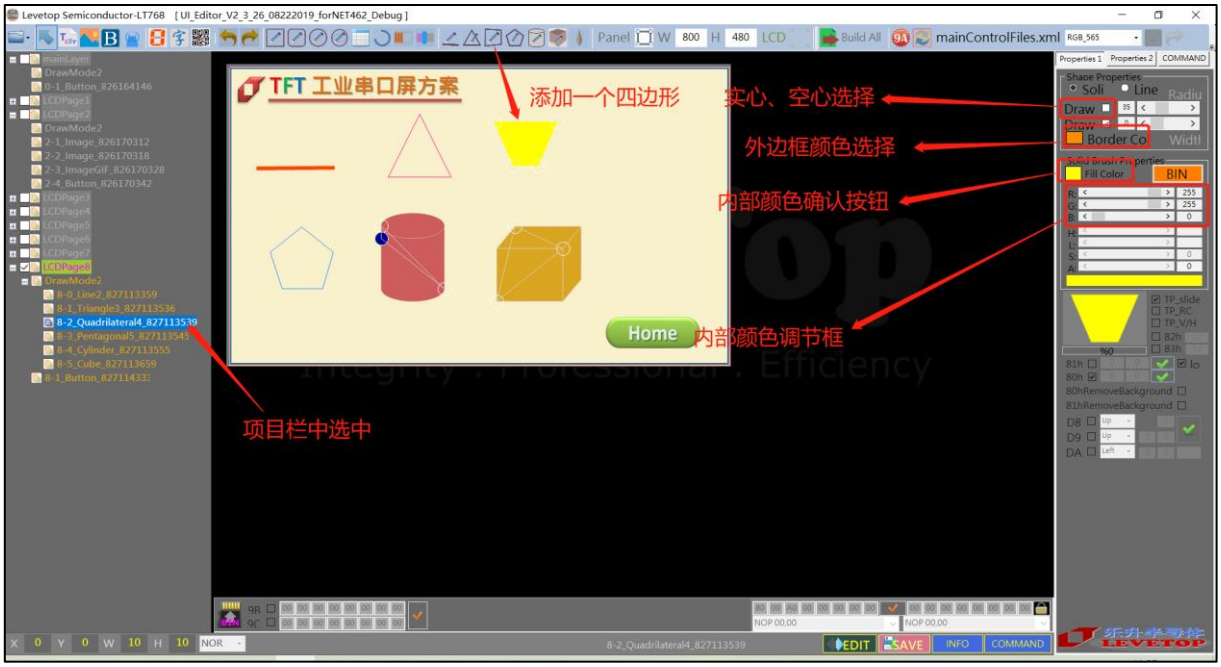


图 3-85：画四边形设置

不是所有 TFT 串口控制芯片都有完整的几何绘图功能，例如 LT7689 的 TFT 串口屏中四边形只有空心和实心功能，所以只有 Draw1 有效，Draw1 被勾选则选用空心功能，Draw1 没有被勾选则选用实心功能。外边框颜色和内部颜色的设置与画圆功能的一样。 画五边形的操作与画四边形的操作一样。 画圆柱体，点击圆柱体添加按钮，在屏幕框内点击一下，就会出现一个圆柱体，改变圆柱体上的三个点可以改变圆柱体的形状。因为 LT7689 的 TFT 串口屏只支持画带外边框实心圆柱体，所以 Draw1 和 Draw2 都失效。边框颜色和内部颜色都和其他绘图功能的一样。

© Levetop Semiconductor Co., Ltd.



图 3-86：画方柱体设置

画方柱体的操作和画圆柱体的操作一致。画直线，点击画直线按钮，在屏幕框中点击一下，即可产生一条直线。拖动直线上的两个端点，可以改变直线的长度和方向。改变线条粗细设置框里的数值可以改变线条粗细。线条的颜色可以通过选择颜色按钮里的颜色来改变。



图 3-87：画直线设置

© Levetop Semiconductor Co., Ltd.

画任意角度圆环，点击添加圆环按钮，在屏幕框上点击一下，会出现带调整框的圆环。改变调整框的大小可以改变圆环的半径。圆环的粗细和颜色通过改变相应的按钮处的内容即可。在主控端通过串口与 TFT 串口屏连接后，发送指令和圆环的起始角度与转动角度就可以在 TFT 串口屏的屏幕上显示相应角度的圆环。圆环的转动方向是逆时针的，以 12 点钟方向为 0 角度点。

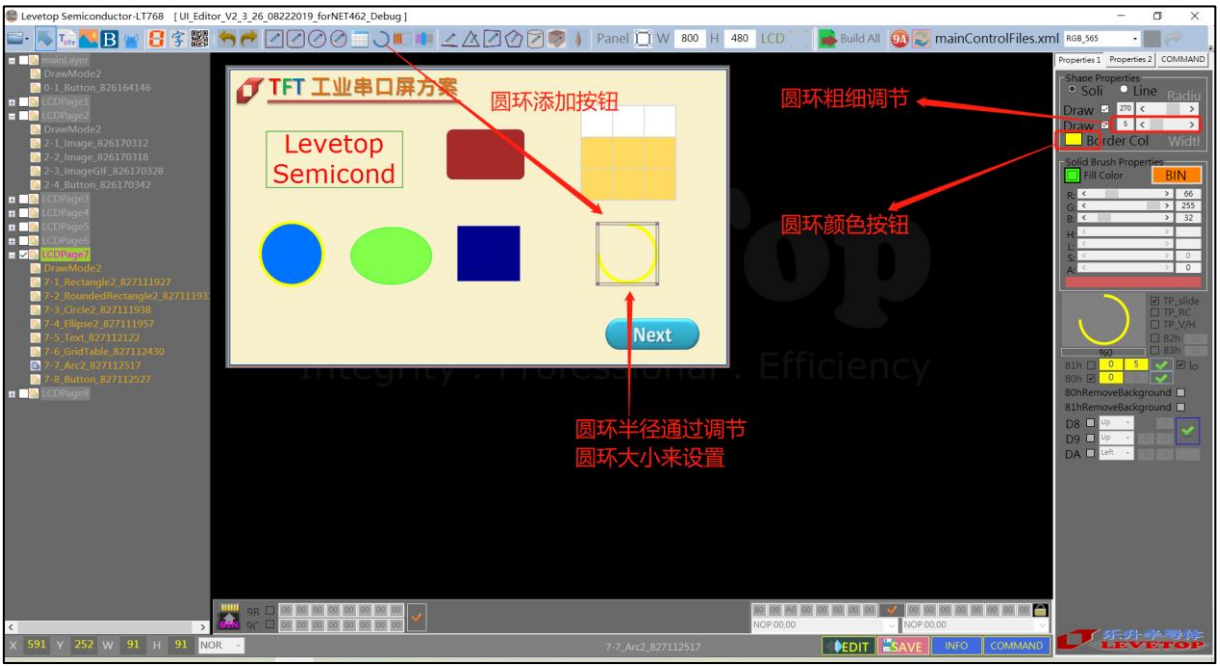


图 3-88：画圆环设置

© Levetop Semiconductor Co., Ltd.

3.4.2 表格制作

按表格添加按钮，在屏幕框内点击一下就会出现一个表格。表格的大小不能通过拉动边框来调整，必须使用右边 Properties2 栏中的 N-Col 和 N-Row 来调整。表格的每一个参数调整都可以在 Properties2 中进行设置。其中如果要更换项目栏的方向（横向、纵向），先选中表格改变 Properties2 中的 Mode 的勾选状态（勾选：横向、不勾选：纵向），再点一下表格就改变项目栏的方向。如果要改变项目栏和内容栏的底色，先选中表格调整对应颜色下面的 RGB 参数，得到自己想要的颜色后，点击一下 Fill color 前面的颜色选项即可。

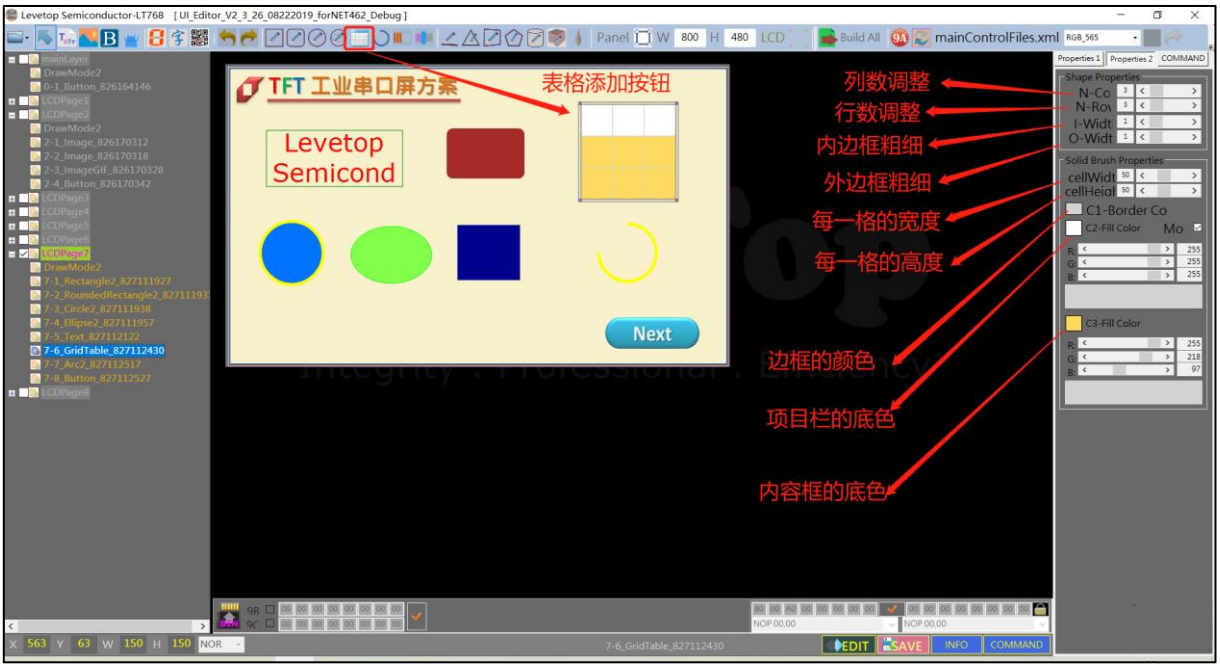


图 3-89：表格设置

© Levetop Semiconductor Co., Ltd.

3.5 其他功能

3.5.1 全屏触摸滑动设定及显示指令

触控滑动控制指令主要是提供触控时，能进行显示画面切换，也就是利用手指在触控屏上进行左右滑动或是上下滑动，显示画面会随滑方向移动，达到平滑切换画面功能，优化显示效能。画面切换是指当前工程中所有图层的底图之间进行切换。

在选中图层，在右边操作栏中找到蓝色的 TP_RC 和 TR_V/H 选项。勾选 TP_RC 代表画面循环切换，不勾选代表不循环切换。勾选 TP_V/H 代表画面切换是在垂直方面切换，不勾选代表水平切换。



图 3-90：触控滑动设定及显示指令

点击 Command，打开串口，点击 B4 指令的 SEND 按钮，再滑动屏幕，屏幕画面就会跟着切换。

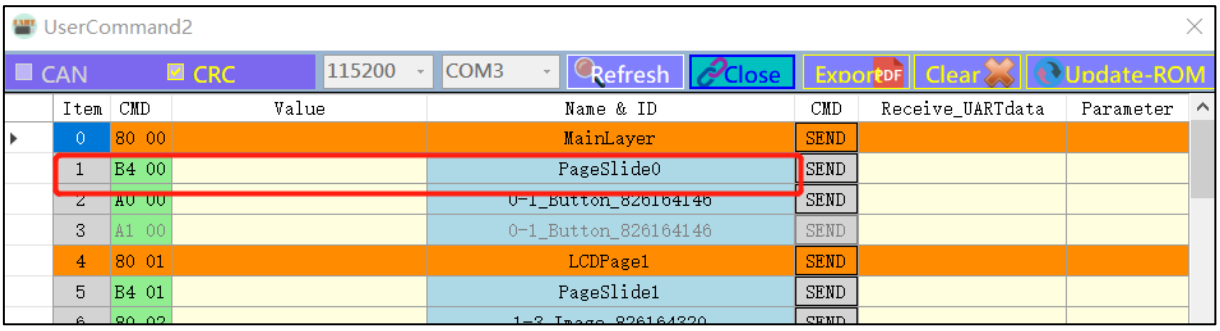


图 3-91：Command 发送 B4 指令

© Levetop Semiconductor Co., Ltd.

3.5.2 背光控制指令

User Cmd 下面默认有 LCD_brightness1 BA. 0A 指令，其中 0A 的大小可以在添加 BA 指令后修改。

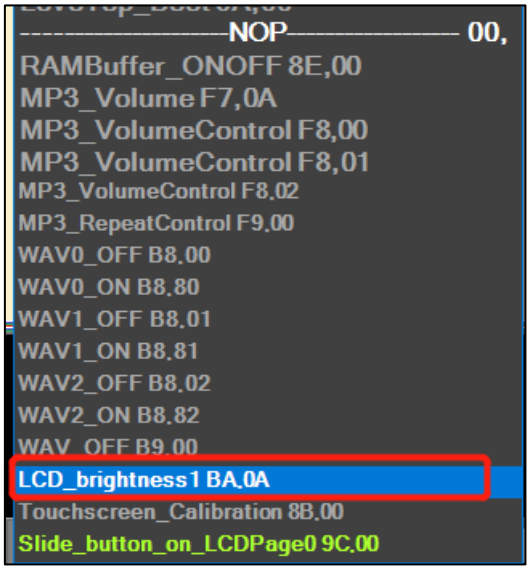


图 3-92：背光控制指令

背景大小调整是通过选中 BA 指令后，右键弹出可变的范围去选择。



图 3-93：背光控制指令

© Levetop Semiconductor Co., Ltd.

3.5.3 起始控制指令

当用 Command 发送指令 BCh 00/BCh 02（00/02 代表 BCh 后面的数值）给 TFT 串口屏，那么 TFT 串口屏的显示会被关闭（Display Off）；发送指令 BCh 01 给 TFT 串口屏，那么 TFT 串口屏的显示会被开启（Display On），如果为 BCh 02 关闭 TFT 串口屏显示，那么点击触控屏也能开启 TFT 串口屏显示。**注意：**显示屏 Off 后，TFT 串口屏将不接受任何绘图功能的指令，直到串口屏收到显示屏 On 指令。

当用 Command 发送指令 BEh 给 TFT 串口屏，如果串口屏初始化完成或是处于 Ready 状态，串口屏会回应 5Ah 给 Command。如果串口屏是处于 Busy 忙状态，串口屏会回应 55h 给 Command。若是 TFT 串口屏与 Command 的 Uart 通讯口未连接则不会响应任何信息。

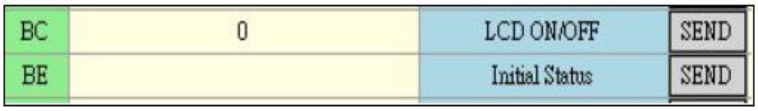


图 3-94：起始控制指令

3.5.4 图片缓冲指令

图片缓冲指令就是通过先在图片在别的图层内缓存完成后，再去显示到主视窗上，从而避免了刷图的显示工程。当收到 8E 00 时，表示要开启缓存区。当收到 8E 01 时，将缓冲区的内容恢复到主视窗上。选中 8E 指令可修改 8E 指令的序号。

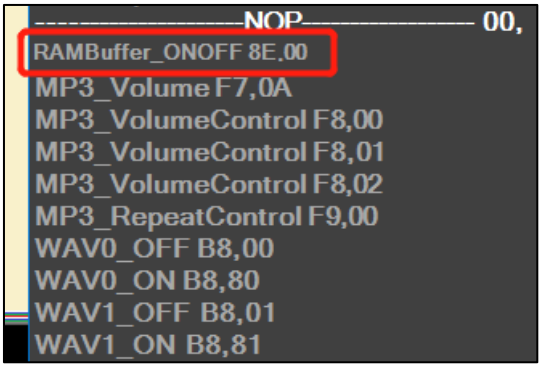


图 3-95：图片缓存指令

© Levetop Semiconductor Co., Ltd.

3.5.5 声音控制指令

UI_Editor 会将 UI_Editor\SOURCE\WAV 路径下的三个 WAV 音频文件分别生成一条 B8h 指令，打开 Command，连接串口。点击对应 B8h 指令的 SEND 键，TFT 串口屏就会播放对应的 WAV 音频，也可以通过按键添加 B8 指令，按下按键播放音频，B8 00 是播放一次，B8 80 是循环播放，对应取消指令是 B9。注意：①要有硬件电路支持才能播放音频。②串口屏最多支持 128 首 WAV 音频。



图 3-96：UI_Editor 里的 WAV 音频文件

B8	00	Wav-1:ON & RP=B7	SEND
B8	01	Wav-2:ON & RP=B7	SEND
B8	02	Wav-3:ON & RP=B7	SEND
B9		Wav-1~3-OFF	SEND

图 3-97：声音控制指令

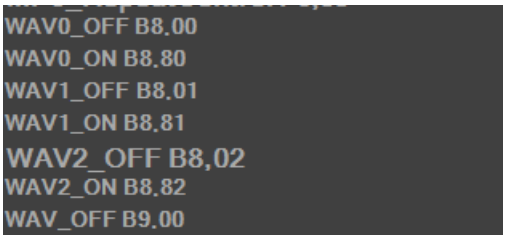


图 3-98：声音控制指令

© Levetop Semiconductor Co., Ltd.

3.5.6 串口屏侦测指令

串口屏侦测指令只需要透过 UART 传递到 TFT 串口屏就会执行，它是主控端系统或是主板的固定指令，这指令不需要在 UI_Editor 上位机软件内设定。

表 3-4：起始控制指令

指令功能	指令码	序号	指令参数	指令说明
检查 TFT 串口屏	BEh	--		串口屏初始化完成或是处于 Ready 状态： 5Ah → Ready； 55h → Not Ready（忙状态）；
检查 TFT 串口屏版本	BFh	--		读取串口屏版本信息：MCU Code 版本（5Bytes） + 串口屏模块. (42Bytes)

当 UART 串口传递指令 BEh 给 TFT 串口屏，如果串口屏初始化完成或是处于 Ready 状态，串口屏会回应 5Ah 给主控端。如果串口屏是处于 Busy 忙状态，串口屏会回应 55h 给主控端。若是 TFT 串口屏与主控端的 Uart 通讯口未连接则不会响应任何信息。

串口指令 BFh 是用来读取 TFT 串口屏版本，包括 5 个 Byte 的程序码版本，及 42 个 Bytes 的串口屏模块信息。当 UART 串口传递指令 BFh 给 TFT 串口屏，串口屏则依序送出这些信息给主控端 MCU。详细的版本信息如下表：

表 3-5：串口屏版本信息

Items	No.	定 义	说 明	举 例	
MCU Code 版本内容	1	年		07	07E3=2019
	2			E3	
	3	月		07	7 月
	4	日		05	5 日
	5	自定义号 0～255		CC	
串口屏保存信息	6	上位机版本号	0x10 代表 V1.0，0x11 代表 V1.1, 0x12 代表 V1.2	10	V1.0
	7	通信接口	0x00-Uart，0x01-SPI，0x02-IIC	00	Uart
	8	通信接口速率	高位在前，低位在后	01	01C200=115200
	9			C2	
	10			00	
	11	PCB 版本号	0x10 代表 V1.0，0x11 代表 V1.1, 0x12 代表 V1.2	10	V1.0
	12	MCU	0x00 代表 8 位单片机，0x01 代表 16 位，0x02 代表 42 位	02	32bit
	13	MCU 厂家	0x00：Levetop，0x01：ST，0x02 = STC	01	STM
	14	Part Number	0x00: 7680A, 0x01: 7680B, 0x02: 7681, 0x03: 7683, 0x04: 7686, 0x05: 7688, 0x06: 268A, 0x07: 268B, 0x08: 3688, 0x09: 7689, 0x0A: 8688, 0x0B: 268D, 0x20: 776	09	LT7689
	15	MCU 与串口控制芯片通信接口	MCU to LTx68 I/F: 0x00 = 8080-8, 0x01 = 8080-16, 0x02 = SPI, 0x03 = I2C	02	SPI
	16	水平像数		01	01E0=480
	17			E0	

© Levetop Semiconductor Co., Ltd.

Items	No.	定 义	说 明	举 例	
	18	垂直像数		01	0110=272
	19			10	
	20	VBPD		00	0014=20
	21			14	
	22	VFPD		00	000C=12
	23			0C	
	24	VSPW		00	0003=3
	25			03	
	26	HBPD		00	008C=140
	27			8C	
	28	HFPD		00	00A0=160
	29			A0	
	30	HFPW		00	0014=20
	31			14	
	32	PCLKRISING		01	1
	33	HSYNCPolarity		00	0
	34	VSYNCPolarity		00	0
	35	DEPolarity		01	1
	36	RGBSequence	000b : RGB,001b : RBG,010b : GRB,011b : GBR,100b : BRG,101b : BGR	00	RGB
	37	ColorDepth	0x00：8bits，0x01：16bits，0x02：24bits	01	16Bits
	38	FlashType	0x00：NorFlash，0x01：NandFlash	00	NOR Flash
	39	FlashSize	单位: 1MBytes	00	0080=128Mbyte
	40			80	
	41	UI Editor 编译工程版本	0x06：06 (ex: 06_04_2019-09_24_53)	07	7 月
	42		0x04：04	05	5 日
	43		0x14：20	14	2019
	44		0x13：19	13	
	45		0x09：09	0A	时
	46		0x18：24	11	分
	47		0x35：53	25	秒

© Levetop Semiconductor Co., Ltd.

3.5.7 User 手动输入指令

双击右下角的 CMD CALL are input by the user 进行 User 手动输入指令模式，此时只能由 User 输入 CMD，再次 CMD CALL are input by the user 退出 User 手动输入指令模式，此时可正常选择指令栏内的 CMD。



图 3-99：User 手动输入指令

© Levetop Semiconductor Co., Ltd.

3.5.8 RTC 数字时钟指令

数字时钟指令 92h，可显示时间和日期，选择一个适合的字体确定后 UI 会自动获取系统当前的信息。当同时显示时间日期的时候，可在设置界面单独设置时间的坐标，在 UI 左下角单独设置日期坐标，效果在模拟器和板子上会呈现出来。

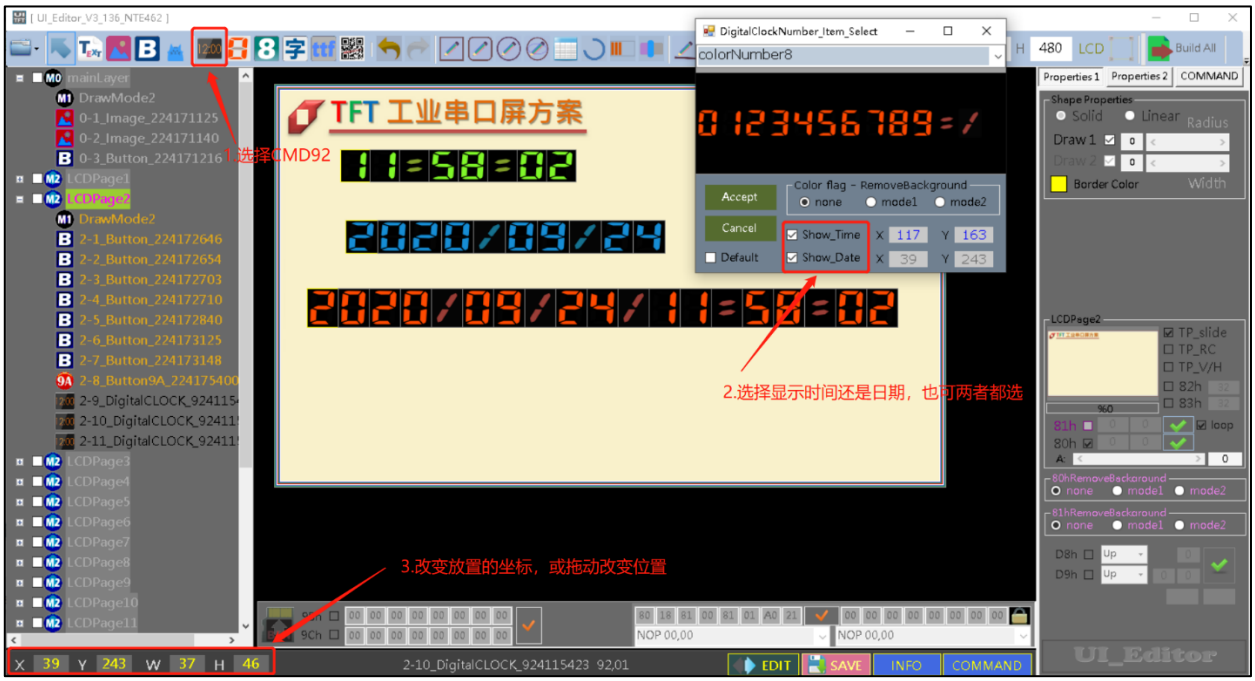


图 3-100：数字时钟指令 92h 设置

© Levetop Semiconductor Co., Ltd.

3.5.9 RTC 模拟时钟指令

首先拉入两张正方形图片（一张时钟背景图一张中心点图标），然后将中心点图标置于背景图的范围内，

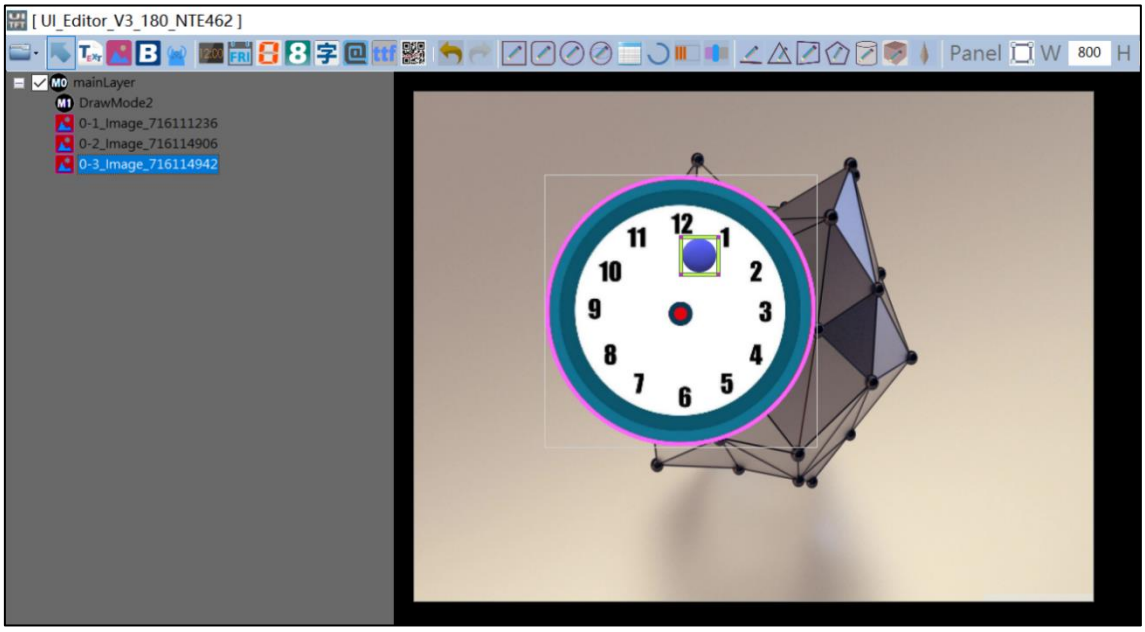


图 3-101：模拟时钟指令 93h 设置

点选背景图，右击选择 AnalogClock，

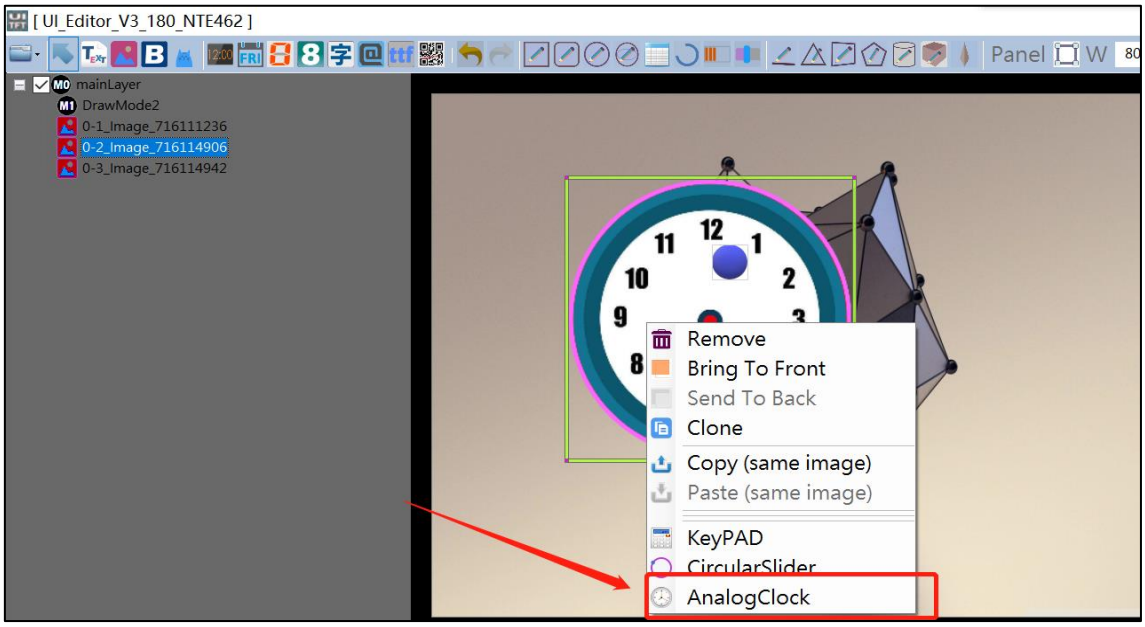


图 3-102：模拟时钟指令 93h 设置

© Levetop Semiconductor Co., Ltd.

在下面指令框设置中心点图标的指令，接着双击下边的小时钟图标，

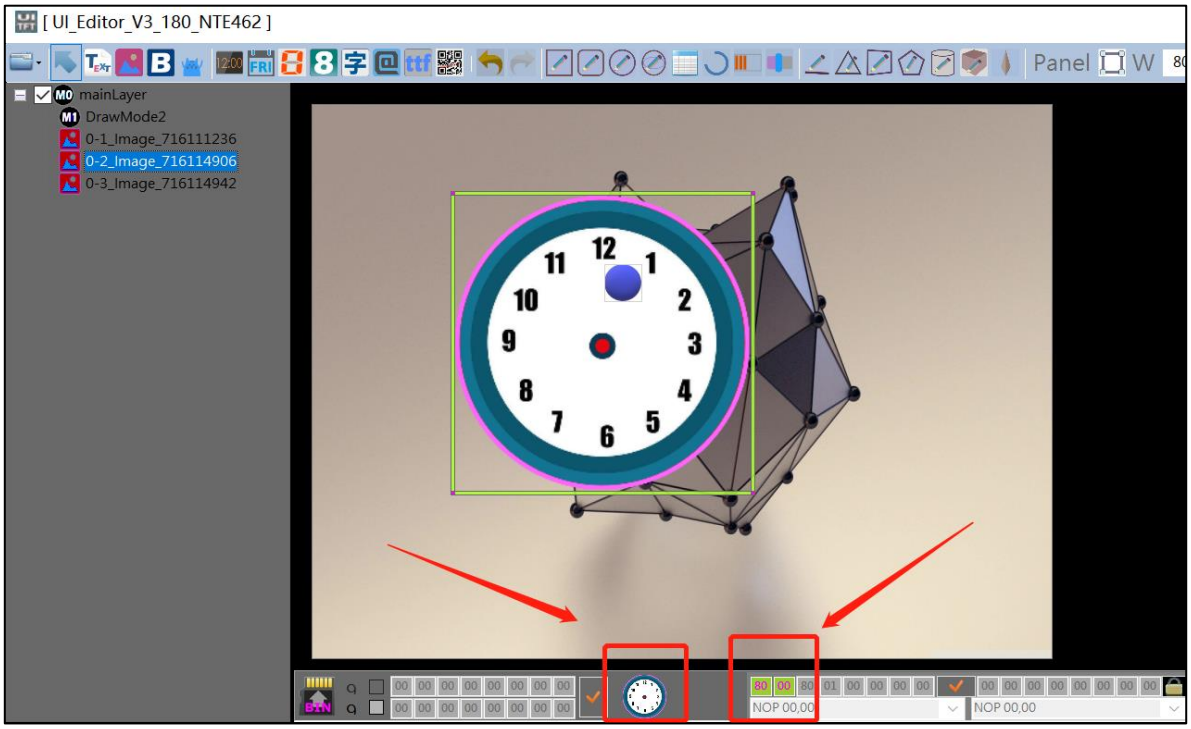


图 3-103：模拟时钟指令 93h 设置

在跳出来的小窗口中设置时针分针秒针的参数，设置好后点勾选保存完成。

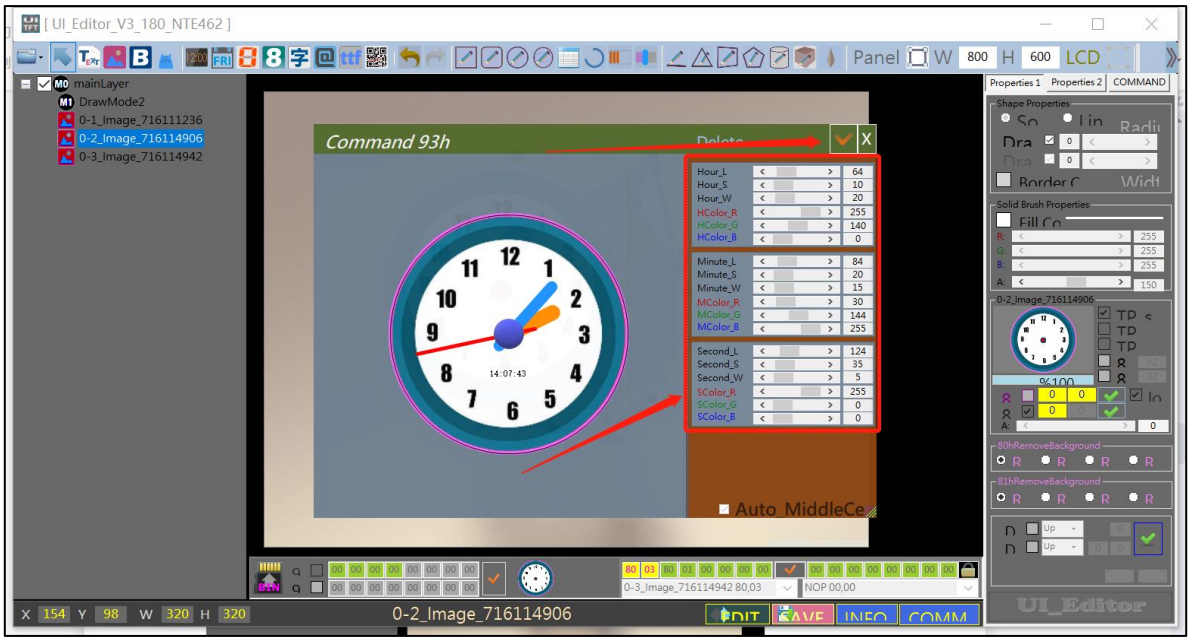


图 3-104：模拟时钟指令 93h 设置

© Levetop Semiconductor Co., Ltd.

3.5.10 显示星期指令

将准备好的图片素材放至指定路径，文件夹名称及文件夹里面图片命名都要按照指定命名来，

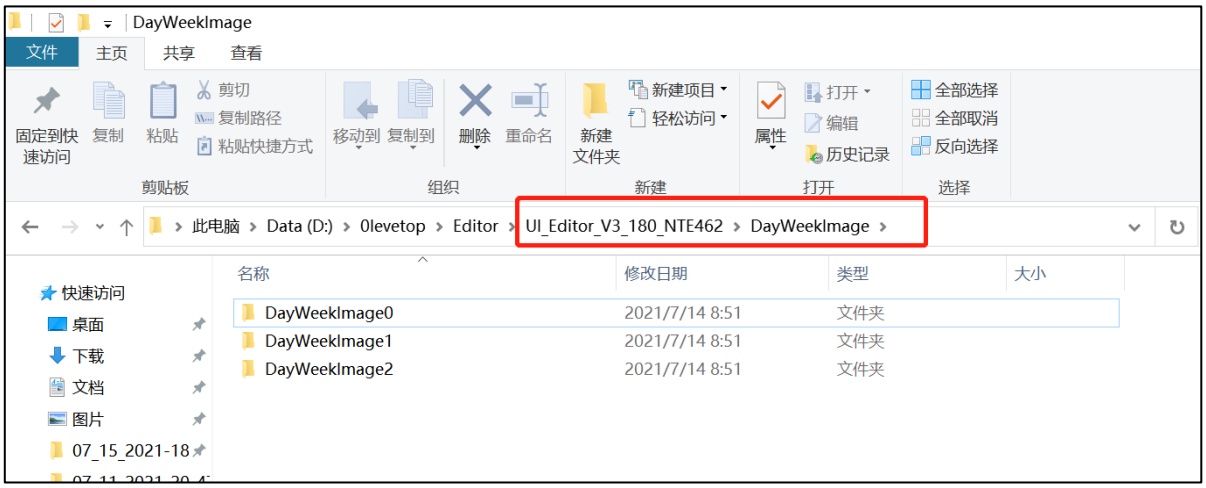


图 3-105：显示星期指令 9Dh 设置

点击添加显示星期指令，在弹出的窗口选择一组准备好的图片素材，再选择图片处理模式后点击 Accep 保存。



图 3-106：显示星期指令 9Dh 设置

© Levetop Semiconductor Co., Ltd.

3.5.11 寄存器操作

CAh ~ CFh 指令主要是对寄存器进行操作，适用于数字变化、亮度调节、音量调节等情况。

CAh：根据寄存器的值来调用 9A 指令，CAh 00h 即调用指令 9Ah 【00】

CBh：此指令为指向寄存器，CBh 00h 即指向寄存器 00；

CCh：此指令为写寄存器，CCh 01h 与 CBh 00h 一起使用即往寄存器 00 写入值 01，此时【00】变成 01；

CDh：此指令为读寄存器，CDh 03h 就是读此时寄存器 03 的值，然后上传到主控端；

CEh：此指令为寄存器内的值加一，CEh 02h 即寄存器 02 此时的值加一，【02】+ 1；

CFh：此指令为寄存器内的值减一，CFh 05h 即寄存器 05 此时的值减一，【05】- 1。

下面以数字变化为例，讲解如何在 UI_Editor 操作使用 CAh ~ CFh 指令。



图 3-107：CA-CF 操作举例说明

© Levetop Semiconductor Co., Ltd.

3.5.12 关闭触控功能指令

9E 指令用于关闭当前界面下所有控件的触控功能，9E A0 表示关闭所有 A0 的触控功能，9E A2 表示关闭所有 A2 的触控功能，此功能适用于某些特定场景。

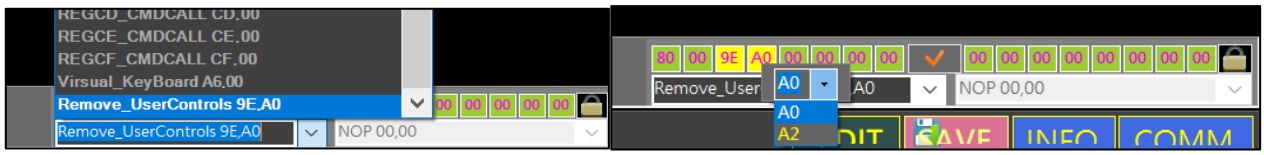


图 3-108：9Eh 操作举例说明

© Levetop Semiconductor Co., Ltd.

3.5.13 全键盘指令

在 UI_Editor 的\SOURCE\VKeyboard 目录下放入 4 个文件(默认自带,若没有可询问我司提供)后,在 UI_Editor 直接设置 A6 指令即可，对应取消指令是 A7 指令。



图 3-109：全键盘相关文件

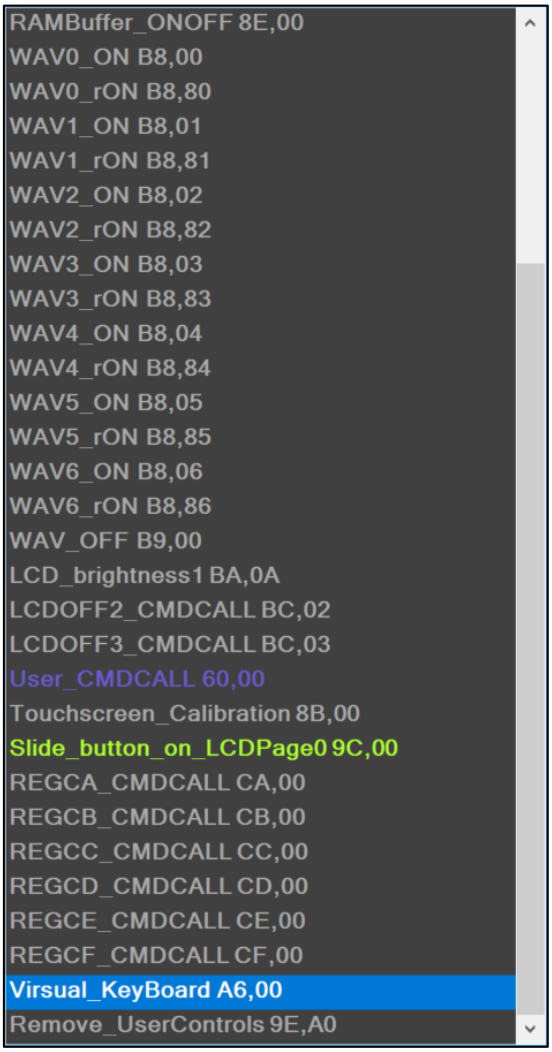


图 3-110：设置 A6 指令

© Levetop Semiconductor Co., Ltd.

3.5.14 滑动手势指令

当用户希望使用滑动屏幕去执行指令时，可通过设置手势操作 C9h 指令实现，C9h 支持上滑、下滑、左滑和右滑执行指令的操作。



图 3-111：设置 C9 指令

© Levetop Semiconductor Co., Ltd.

3.5.15 电阻屏控制指令

电阻屏控制指令只需要透过 UART 或是 SPI 传递到 TFT 串口屏就会执行，它是主控端系统或是主板的固定指令，这指令不需要在上位机软件或是编程软件设定。例如当 UART 串口传递控制指令 8Bh 给 TFT 串口屏，那么就会自动进入电阻屏的 4 个角落校验程序，用户必须依照显示的角落点击完成电阻式触控屏校验程序。

表 3-6：电阻式触控屏控制指令

指令功能	指令码	序号	指令参数	指令说明
进行电阻屏校对	8Bh	--		进行电阻屏的 4 个角落校验点。



图 3-112：电阻屏的 4 个角落校验

© Levetop Semiconductor Co., Ltd.

3.5.16 时钟控制指令

时钟控制指令只需要透过 UART 或是 SPI 传递到 TFT 串口屏就会执行，它是主控端的固定指令，这指令不需要在上位机软件或是编程软件设定。

表 3-7：RTC 时钟控制指令

指令功能	指令码	序 号	指令参数	指令说明
接收时钟数据	**8Ch**	--	Y(1), M(1), D(1), H(1), M(1), S(1), W(1)	主控端设置目前 TFT 串口屏上的时钟：年/月/日/时/分/秒/周 数据，共 7 Bytes，以 16 进制表示。
发送时钟数据	**8Dh**	--		回应主控端目前 TFT 串口屏上的时钟信息：年/月/日/时/分/秒/周 数据。

例如当 UART 串口传递控制指令 8Ch 15 04 1D 0E 30 05 04 给 TFT 串口屏，那么 TFT 串口屏上的时钟（RTC）就会被就设置为 2021 年 4 月 29 日 14 时 48 分 5 秒，星期四。如下说明：

年 15h=2000+21=2021 年
月 04h=4 月
日 1Dh=29 日
时 0Eh=14 时
分 30h=48 分
秒 05h=5 秒
周 04h=星期四

例如 UART 串口传递命令 8Dh 给 TFT 串口屏，那么 TFT 串口屏就会将串口屏上的时钟数据 - 年/月/日/时/分/秒/周上传给主控端。

© Levetop Semiconductor Co., Ltd.

3.5.17 串口屏复位

串口屏复位指令只需要透过 UART 或是 SPI 传递到 TFT 串口屏就会执行，它是主控端系统或是主板的固定指令，这指令不需要在上位机软件或是编程软件设定。例如通过例如 UART 串口传递命令 BDh 给 TFT 串口屏，那么 TFT 串口屏就会进行复位。

3.5.18 设定音量命令

影音播放可以通过 F7 指令来调节音量，调节的范围为 0x00 至 0x1E，用户可以通过打开 Command，连接串口，点击对应 F7h 命令的 SEND 键，就会执行相应的动作，也可以通过按键添加 F7h 指令，按下按键进行设定音量。

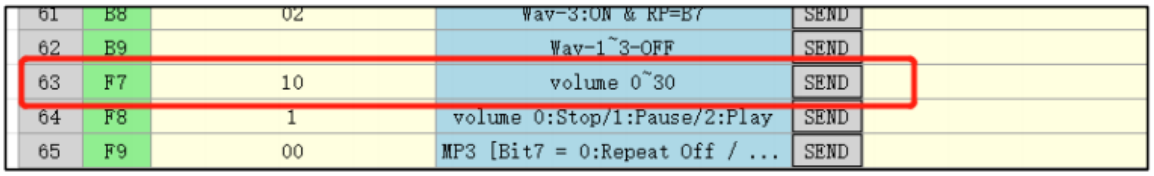


图 3-113：设定音量命令

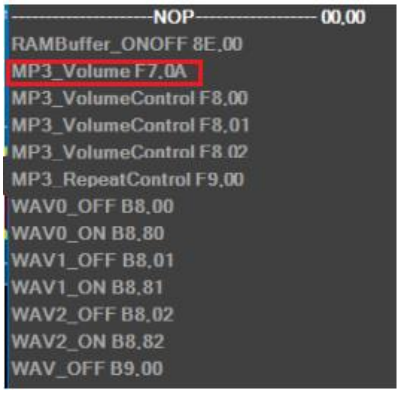


图 3-114：设定音量命令

用户在控件添加完 F700 后，可选中 F7h 指令后点击右键修改数值，如下图：

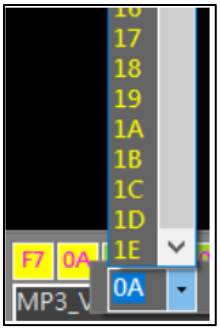


图 3-115：F7 指令设置

© Levetop Semiconductor Co., Ltd.

3.5.19 影音播放控制命令

影音播放（即 MP4 和 MP3）有三条控制指令，F8 00（停止播放音频）、F8 01（暂停播放音频）和 F8 02 (继续播放音频)。用户可以通过打开 Command，连接串口，点击对应 F8h 命令的 SEND 键，就会执行相应的动作，也可以通过按键添加 F8h 指令，按下按键进行音频播放控制。

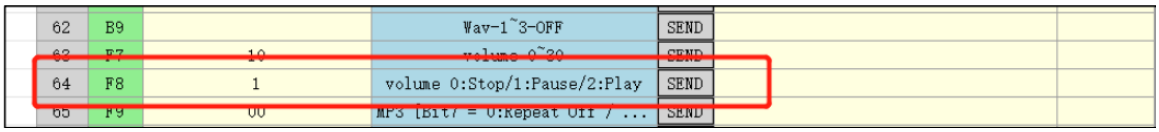


图 3-116：影音播放控制命令

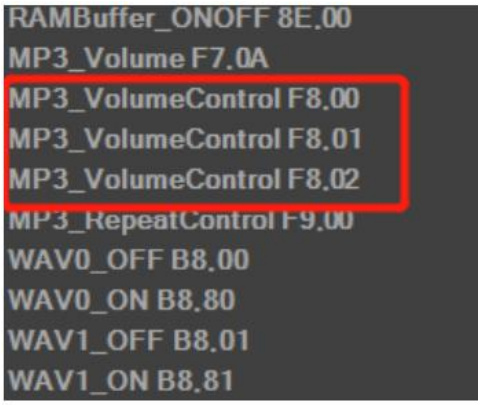


图 3-117：影音播放控制命令

© Levetop Semiconductor Co., Ltd.

3.5.20 影音播放命令

LT3688 支持 MP3 及 MP4 的影音格式播放，透过串口指令可以播放存在 SD 卡的影音档案。主控端或主板只需要透过 UART 口传递命令到 LT768x 的 TFT 串口屏就会执行，它是主控端的固定命令，这些固定命令与上位机软件或是编程软件无关。用户可以通过打开 Command，连接串口，点击对应影音播放命令的 SEND 键，就会执行相应的动作，也可以通过按键添加对应影音播放指令，按下按键进行音频播放控制。

播放 MP3 指令不需要写在指令文件内，例如当 UART 串口传递命令 F9h、00h 给 TFT 串口屏，那么 TFT 串口屏的喇叭就会播放存在 SD 卡内的第一个 MP3 檔（0.MP3）音乐。当 UART 串口传递命令 F9h、81h 给 TFT 串口屏，那么 TFT 串口屏就会不断循环播放第二个 MP3 檔 (1.MP3) 音乐。当 UART 串口传递命令 F8h、00h 给 TFT 串口屏，那么 TFT 串口屏就会停止播放 MP3 音乐。

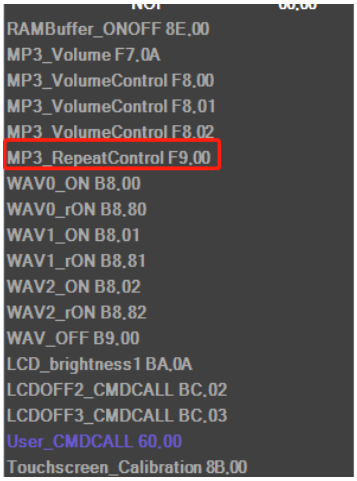


图 3-118：影音播放命令

用户在控件添加完 F9 00 后，可选中 F9h 指令后点击右键修改数值，如下图：

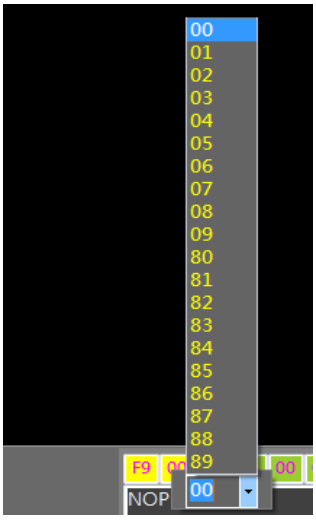


图 3-119：影音播放命令

© Levetop Semiconductor Co., Ltd.

如果设定 MP4 视频播放，则按 MP4 视频播放按钮，在屏幕框拉动出现视频播放窗口。然后可以设置 FB 指令执行时播放的 SD 内的 MP4 视频与播放模式，F8 00 对应 SD 卡目录下的 SD:\RomFiles\Video 的 0.MP4 视频文件，FB 01 对应 SD 卡目录下 SD:\RomFiles\Video 的 1.MP4 视频文件。注意：设置窗口的大小必须要不 MP4 视频的分辨率大小一致。



图 3-120：MP4 视频播放

FAh 播放 MP4 指令不需要在上位机上设置参数，直接通过串口发送参数显示。例如当 UART 串口传递命令 FAh、0010h、0020h、00FFh、00FFh、00h、02h 给 TFT 串口屏，那么 TFT 串口屏就会在（16, 32）坐标播放 256*256 大小且存在 SD 卡内的 MP4 檔（2.MP4）。

表 3-8：FAh 影音播放设定指令

指令功能	指令码	序号	指令参数	指令说明
播放 MP4 视频 - 1	**FAh**	--	X(2), Y(2), Width(2), Height(2), Mode(1), Name(1)	将视频显示在（X, Y）位置，宽度为 Width、高度为 Height。Name 为档案序号名称。Mode 为 0：播放一次，Mode 为 1：循环播放。

© Levetop Semiconductor Co., Ltd.

3.5.21 使用矢量字库的文字设定及显示指令

LT3688 支持矢量文字显示，透过串口指令可以调用存在 SD 卡的矢量字库档案（.TFT）。FC 矢量字库指令调用矢量字库显示的参数可在上位机上设定，按 ttf 矢量字库按钮，在屏幕框拉动出现矢量字库显示窗口，设定显示参数，最后主控端通过串口发送显示数据即可调用矢量字库显示。

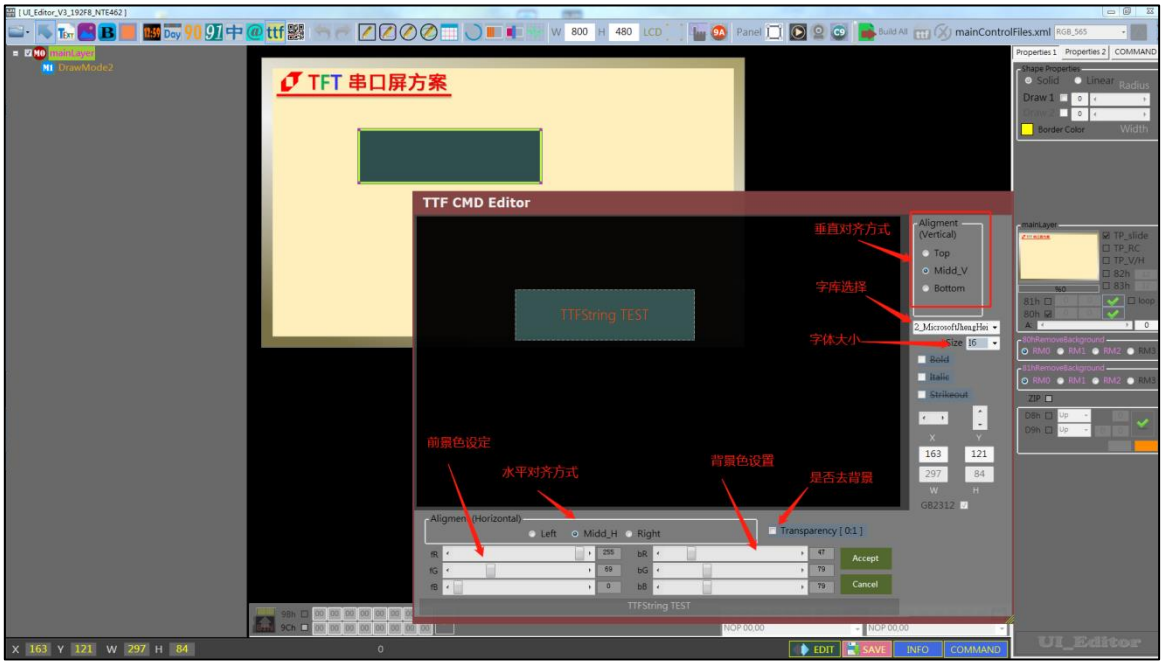


图 3-121：矢量字库设定

如果是调用 FDh 指令来显示矢量字库。FDh 指令和 FCh 类似，但是 FDh 指令只需要透过 UART 口传递 LT768x 的 TFT 串口屏就会执行，它是主控端的固定命令，这些固定命令与上位机软件或是编程软件无关。

表 3-9：使用矢量字库的文字设定指令 - 2

指令功能	指令码	序号	指令参数	指令说明
设定显示矢量字库 - 2	FDh	--	Name(1), X0(2), Y0(2), X1(2), Y1(2), Color-F(3), Color-B(3), Size(1), Align-H(1), Align-V(1), Tr(1), String	调用矢量字库序号档（Name.TTF）将文字显示在 (X0, Y0) 与 (X1, Y1) 的区域，文字前景颜色为 Color-F、背景颜色为 Color-B，Size 为文字大小。Align-H 为水平对齐；Align-H 为垂直对齐（0：不对齐；1: 对齐）。Tr = 0：不透明; 1: 透明。 String 为要显示出的字符串。

© Levetop Semiconductor Co., Ltd.

3.6 WAV 音频、字库文件的说明

3.6.1 WAV 音频文件说明

WAV 音频文件存放于 UI_Editor\SOURCE\WAV 路径下。



图 3-122：Wav BIN 文件

有关 WAV 音频文件的制作请参考第 4.2 节的内容。

3.6.2 字库文件的说明

字库文件存放于 UI_Editor\FONT 路径下。其中 fontList.txt 文件的内容是所有字库文件的名称按排列顺序编写，如下图：

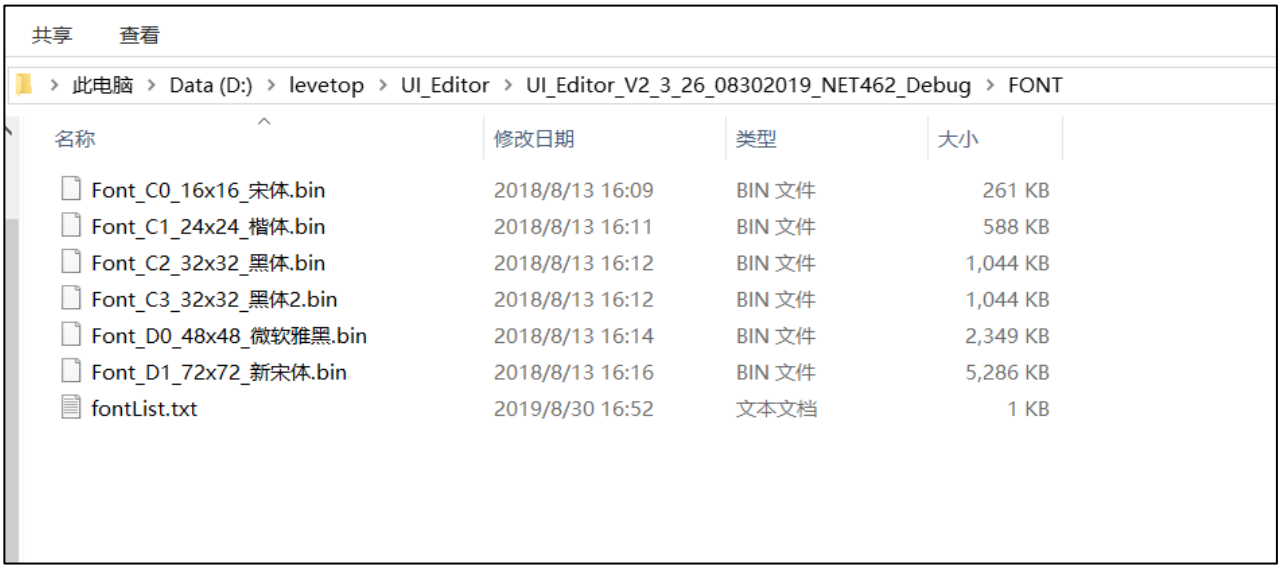


图 3-123：FONT 文件

有关字库文件的制作请参考本文第 4.1 节的内容。

© Levetop Semiconductor Co., Ltd.

3.6.3 真彩数字图片文件说明

真彩数字图片文件存放于 UI_Editor\ColorNumber 路径下。

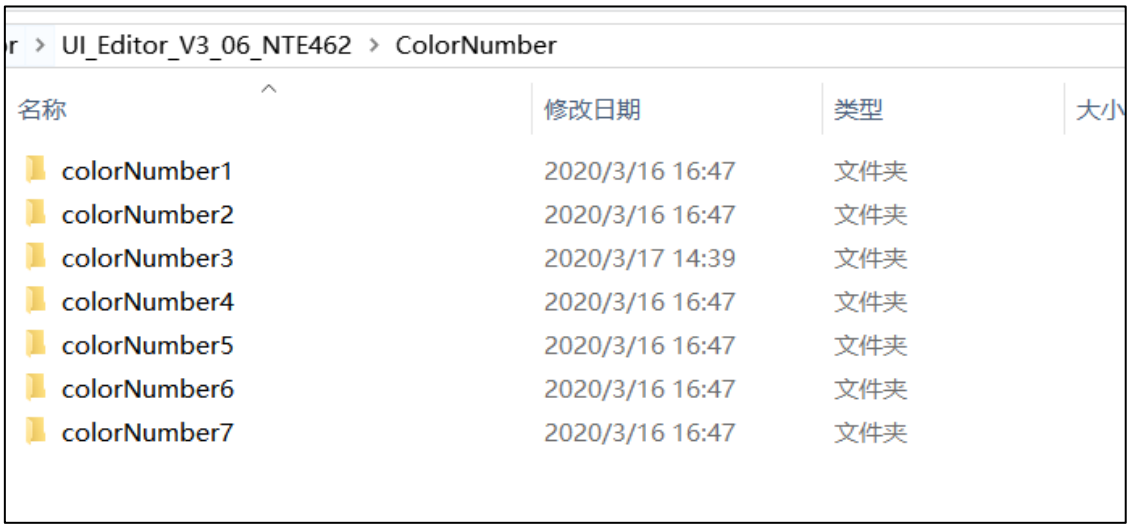


图 3-124：真彩数字图片文件

© Levetop Semiconductor Co., Ltd.

3.7 BIN 文件的生成和串口发送指令

生成 BIN 文件之前，要设置 TFT 串口屏中屏幕的参数和主控端与 TFT 串口屏通信的参数。通过点击 UI_Editor 界面中的 INFO 按钮来打开参数设置。设置完后，点击 INI SAVE 按钮确定设置。

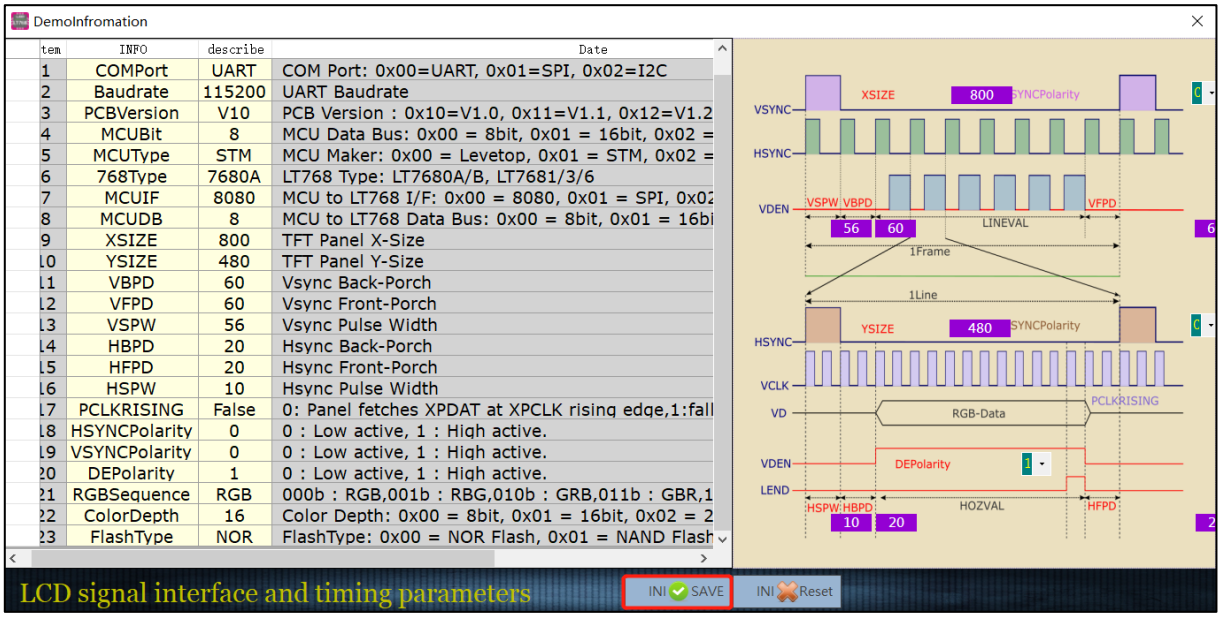


图 3-125：INFO 内配置参数

设置好 INFO 中的参数后，按工程编译按钮。



图 3-126：工程编译

© Levetop Semiconductor Co., Ltd.



图 3-127：工程编译成功

等待工程编译成功，之后再按 BIN GEN 按钮生成 BIN 文件到 BINFILE 下级文件夹中按时间命名的文件夹中。

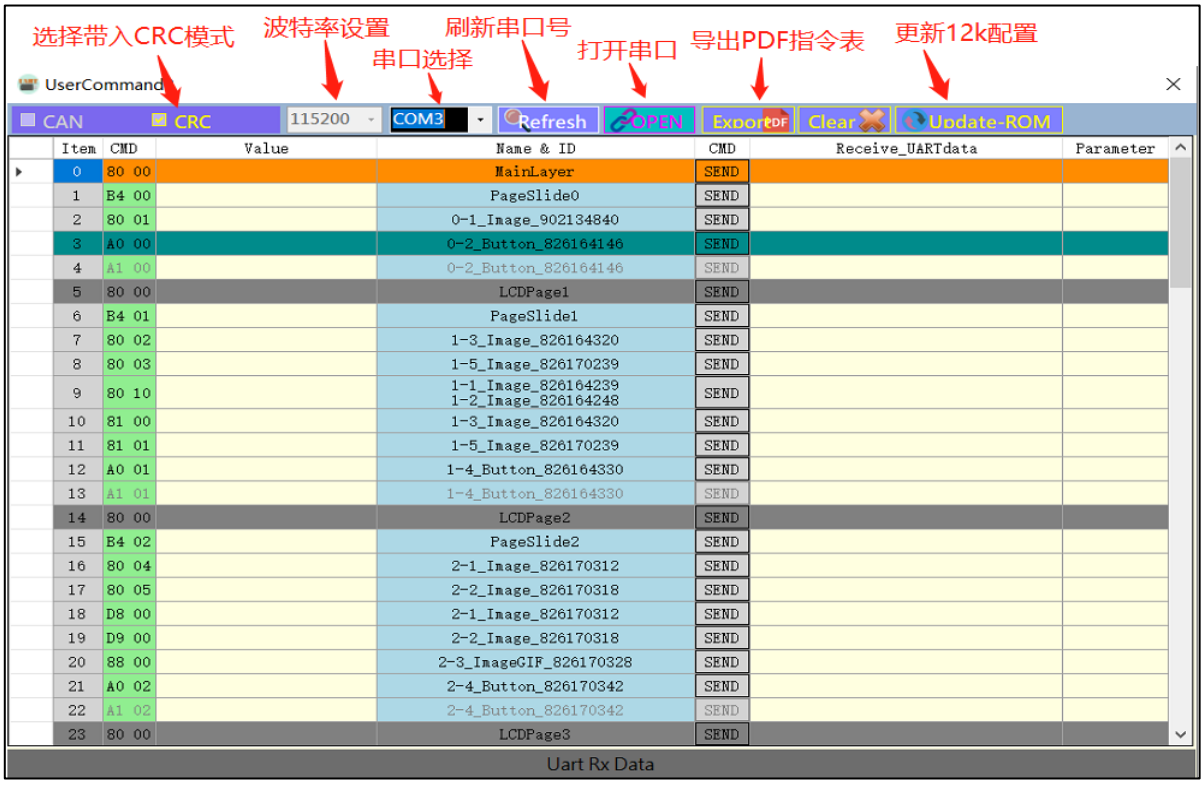


图 3-128：Command 发送指令给 TFT 串口屏

© Levetop Semiconductor Co., Ltd.

通过烧录工具将生成的 BIN 文件烧录到 TFT 串口屏中，连接 TFT 串口屏和主控端。点击 Command 按钮，弹出指令发送界面。设置串口波特率，选择串口号，打开串口，就可以发送指令。如果是要带数值的指令，Value 框中会有默认的数值，可以修改数值来实现其他的效果。按 SEND 按钮即可发送指令给 TFT 串口屏。

当开发者用 UI_Editor 将产品的 UI 接口规划及验证后，可以导出整个 UI 所运用到的指令表，主控端依据这些指令在其 MCU 程序上植入这些指令格式，在想要显示图片或是要求 TFT 串口屏做出动作时只要送出对应的指令及 CRC 码即可，同时主控端可以依据 TFT 串口屏的响应判断是否被正确执行。下图为设计完成后导出的指令表。

由图上举例来说，主控端 MCU 程序送出 80h、00h、1Bh(CRC1)、98h(CRC2) 指令后，TFT 串口屏就会显示第一张 MainLayer 图片；主控 MCU 程序送出 80h、01h、0Bh(CRC1)、B9h(CRC2) 指令后，TFT 串口屏就会显示第二张图片 Image_902134840，以此类推。有关 2 个字节 CRC 的产生方式请参考手册前面第 2.4 节。

LT7680A ：LCD Serial Communication Command Reference

Author : LeveTop

Run Date : 2019/9/9

Item	CMD	Value	Name & ID	CMD	Receive_UARTdata	Parameter
0	80 00		MainLayer	SEND		
1	B4 00		PageSlide0	SEND		
2	80 01		0-1_Image_902134840	SEND		
3	A0 00		0-2_Button_826164146	SEND		
4	A1 00		0-2_Button_826164146	SEND		
5	80 00		LCDPage1	SEND		
6	B4 01		PageSlide1	SEND		
7	80 02		1-3_Image_826164320	SEND		
8	80 03		1-5_Image_826170239 1-2_Image_826164248	SEND		
9	81 00		1-3_Image_826164320	SEND		
10	81 01		1-5_Image_826170239	SEND		
11	A0 01		1-4_Button_826164330	SEND		
12	A1 01		1-4_Button_826164330	SEND		
13	80 00		LCDPage2	SEND		
14	B4 02		PageSlide2	SEND		
15	80 04		2-1_Image_826170312	SEND		
16	80 05		2-2_Image_826170318	SEND		
17	D8 00		2-1_Image_826170312	SEND		
18	D9 00		2-2_Image_826170318	SEND		
19	88 00		2-3_ImageGIF_826170328	SEND		
20	A0 02		2-4_Button_826170342	SEND		
21	A1 02		2-4_Button_826170342	SEND		
22	80 00		LCDPage3	SEND		

图 3-129：指令表导出 PDF 格式

© Levetop Semiconductor Co., Ltd.

下表是用 UI_Editor 设计后所能支持的指令总表，如果 UI_Editor 设计时没用到的功能将不会在导出的 PDF 内出现。

表 3-10：指令总表

指令码	序号	指 令 参 数	指 令 功 能
80h	nn		显示图片
81h	nn		循环显示重叠图片
82h	nn		显示带透明效果的单张图片
84h	nn		取消循环显示重迭图片
85h	nn		取消 80 指令的显示图片
88h	nn		显示 GIF 图片
89h	nn		取消显示 GIF 图片
8Ah	nn		显示单张或多张图片
8Ch		Y, M, D, H, M, S, W	设定时钟 年/月/日/时/分/秒/周
90h	nn	ddd.d	显示图片式的数字
91h	nn	ddd.d	显示图片式的数字
92h	nn		显示数字时钟
93h	nn		显示模拟时钟
94h	nn		显示触控滑条
95h	nn		移除触控滑条。
96h	nn		显示环形触控滑条
97h	nn		移除环形触控滑条
98h	nn	String	显示二维码
9Ah	nn		执行多组的指令，当 nn=00 时为执行开机指令
9Ch	00		执行显示控件滑动
9Dh	nn		显示星期
9Eh	nn		取消控件触控功能
9Fh	nn		显示 ASCII 码图片
A0h	nn		显示控件图片
A1h	nn		取消显示控件图片及功能

© Levetop Semiconductor Co., Ltd.

指令码	序号	指 令 参 数	指 令 功 能
A2h	nn		设置虚拟控件
A3h	nn		取消显虚拟控件及功能
A4h	nn		设置数字键盘输入界面
A5h	nn		取消数字键盘输入界面
B0h	nn	Value(2Bytes)	显示进度条指标图
B1h	nn	Angle (2 Bytes)	显示指针指标图
B4h	nn		滑动多张全屏图片
B8h		REP(Bit7) + WAV	播放 Wav 檔
B9h			停止播放 Wav 檔
BAh		BL（00~0Fh)	调整背光亮度
BCh		00 or 01	显示 On/Off
BDh			串口屏复位
BEh			检查 TFT 串口屏
BFh			版本侦测
C0h	nn	String	显示字库 - 1 文字
C1h	nn	String	显示字库 - 2 文字
C2h	nn	String	显示字库 - 3 文字
C3h	nn	String	显示字库 - 4 文字
CAh	Reg		执行 9A [Reg] 指令
CBh	Reg		设定指到的寄存器
CCh	Data		写入数据到寄存器内
CDh	00		读取寄存器的内容值
CEh	Reg		寄存器内的数据加 1
CFh	Reg		寄存器内的数据减 1
D0h	nn	String	显示大字库 - 1 文字
D1h	nn	String	显示大字库 - 2 文字
D2h	nn	String	显示大字库 - 3 文字
D3h	nn	String	显示大字库 - 4 文字
D8h	nn		显示卷动出现图片

© Levetop Semiconductor Co., Ltd.

指令码	序号	指 令 参 数	指 令 功 能
D9h	nn		显示循环卷动图片
DBh	nn		取消循环卷动图片
DCh	nn	S_Angle, A_Angle	显示环形指标图
DFh	nn	X, Y	显示画点
E0h	nn		显示一条直线
E1h	nn		显示一空心圆形
E2h	nn		显示一实心圆形
E3h	nn		显示一带框实心圆形
E4h	nn		显示一空心椭圆形
E5h	nn		显示一实心椭圆形
E6h	nn		显示一带框实心椭圆形
E7h	nn		显示一空心矩形
E8h	nn		显示一实心矩形
E9h	nn		显示一带框实心矩形
EAh	nn		显示一空心圆角矩形
EBh	nn		显示一实心圆角矩形
ECh	nn		显示一带框实心圆角矩形
EDh	nn		显示一空心三角形
EEh	nn		显示一实心三角形
EFh	nn		显示一带框实心三角形
F0h	nn		显示一空心四边形
F1h	nn		显示一实心四边形
F2h	nn		显示一空心五边形
F3h	nn		显示一实心五边形
F4h	nn		显示一个圆柱体
F5h	nn		显示一画方柱体
F6h	nn		显示一表格视窗

© Levetop Semiconductor Co., Ltd.

3.8 UI_Editor 范例下载

关于 UI_Editor 的使用，用户可以至 乐升半导体 网站下载 UI_Editor 的范例（如 LT7689_UI_Editor_Demo_800x480.rar）来操作：

1. 自本公司网站（www.levetop.cn）下载 UI_Editor 压缩文件（UI_Editor_Vx_x_x.rar），然后解压缩“UI_Editor_Vx_x_x_rar”.

2. 下载 UI_Editor 范例（如 LT7689_UI_Editor_Demo_800x480.rar）解压缩及打开压缩包。

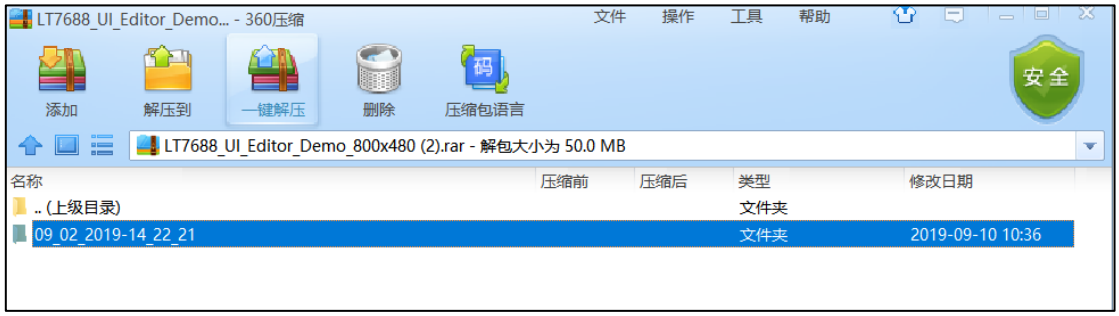


图 3-130：解压缩及打开压缩包

3. 将演示范例目录如“09_02_2019-14_22_21”文件夹解压到 UI_Editor 的 PROJECT 文件夹内。



图 3-131：范例目录拷贝到 UI_Editor 的 PROJECT 文件夹内

注意：

① 不能修改压缩包文件夹名称及内容。

② 压缩包内的文件夹必须解压到 UI_Editor 的 PROJECT 文件夹内。

© Levetop Semiconductor Co., Ltd.

4. 打开 UI_Editor，打开菜单选项后点击 Load：



图 3-132：打开 UI_Editor 点击 Load

5. 打开文件夹内的 COMMANDFILE，选择工程文件后点击打开。

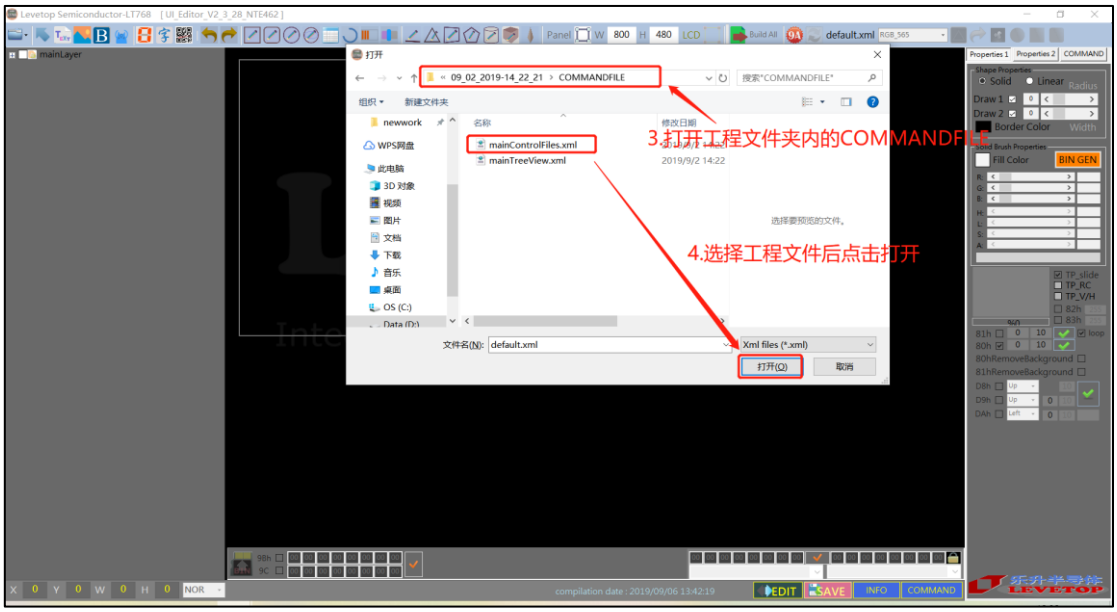


图 3-133：打开 UI_Editor 演示范例

© Levetop Semiconductor Co., Ltd.

3.9 支持压缩图片的指令

部分的串口屏芯片在使用图片显示时支持本公司的专用压缩模式，可以减少 SPI Flash 的使用量，支持该压缩图片的指令如下：

表 3-11：各芯片支持压缩图片的指令

指令码	LT268A	LT268B	LT268C/D	LT269	LT776
80h	V	V	V	V	V
81h	V	V	V	V	V
88h	V	V	V	V	V
8Ah	V	V	V	V	V
8Fh	V	V	V	V	V
91h	-	-	-	-	-
92h	-	-	-	-	-
9Dh	-	-	-	-	-
A0h	V	V	V	V	V
93h	-	-	-	-	V
94h	-	-	-	-	V
96h	-	-	-	-	V
A4h	-	-	-	-	V（只支持背景图）
D8h	-	-	-	-	-
D9h	-	-	-	-	-

注意： LT268x 串口屏芯片在使用压缩图片时仍有部分限制：

1. 压缩图片不能用作全屏底图；
2. 压缩图片不能用作 png 叠加时的底图；

© Levetop Semiconductor Co., Ltd.

4. 制作字库与 Wav 档的 Bin 文件

4.1 制作字库的 Bin 文件

4.1.1 全字库制作

在应用端如果要使用到中文字库，Levetop 提供一个制作字库 Bin 文件的程序 - “BWFont.exe”，能将字库信息转成 Bin 文件，然后透过 DMA 传输方式将字库数据存到 LT7689/LT7688 的内建显示内存中，之后如果要在 TFT 屏上显示中文，MCU 只须送 GB 码（2 个 Bytes）就可以在设定的位置上显示出中文，因此可以提升中文显示效能，还降低 MCU 处理中文显示的负担。对于字库 Bin 文件的制作，使用者可以参考以下的说明，举例产生一个 16*16 的宋体字库 Bin 文件：

1、 执行“BWFont.exe”软件即可打开中文字库 Bin 文件制作界面：

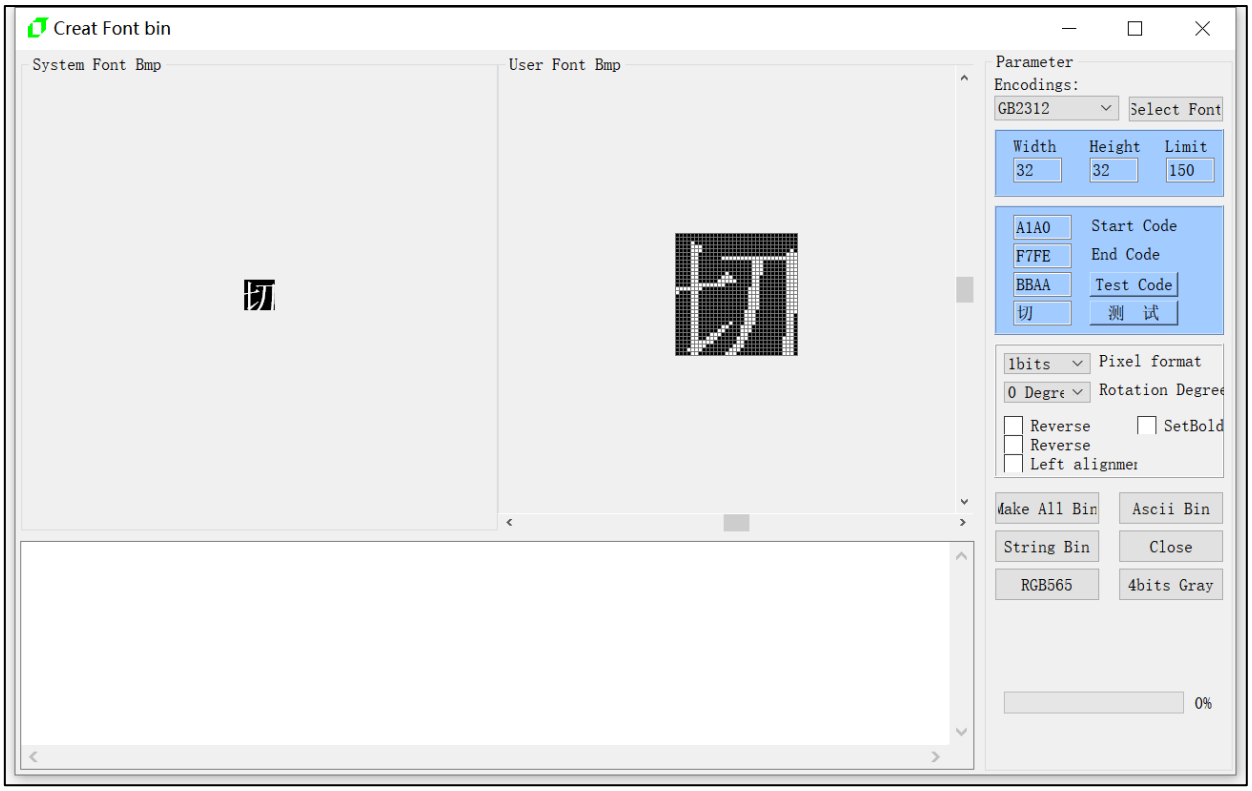


图 4-1：制作中文字库

© Levetop Semiconductor Co., Ltd.

2、 点击【Select Font】按钮，可设置字体、字形、大小等，设置完毕后，按确定保存：

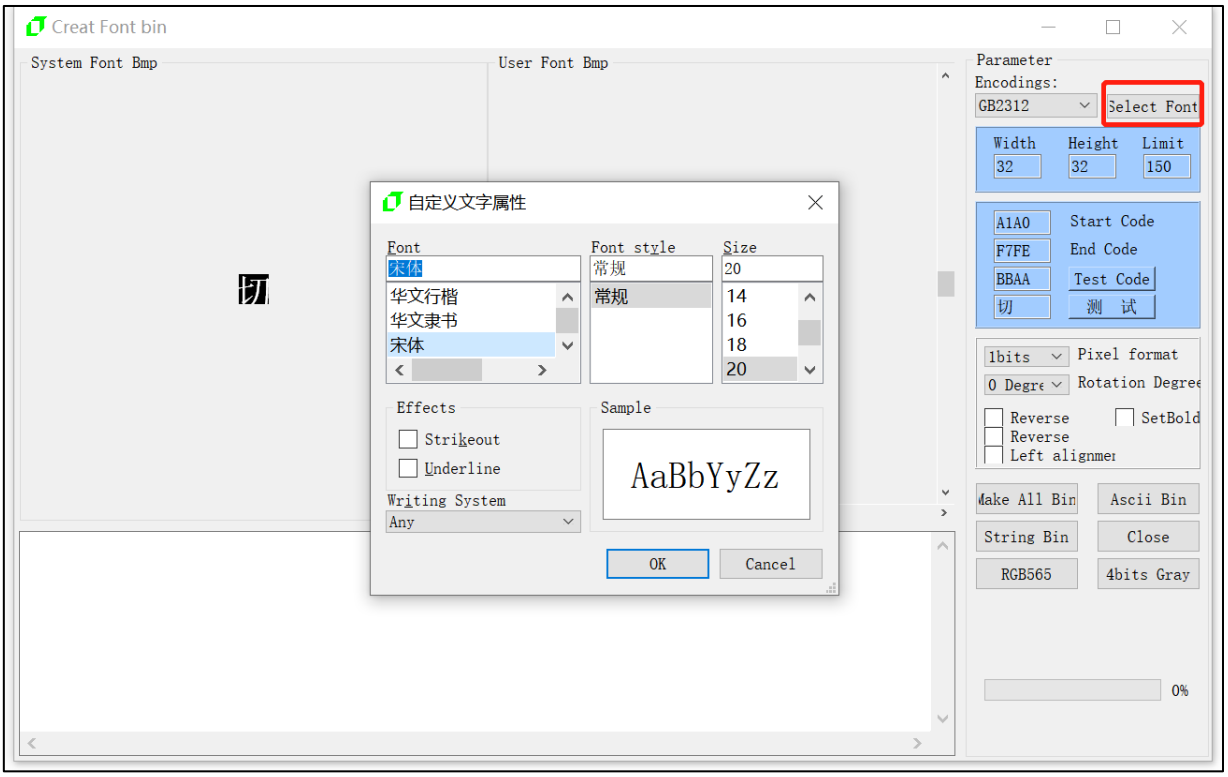


图 4-2：选择字体

3、设置字库可设置任意字体大小，点击【Test Code】按钮即可查看该字符的数据。

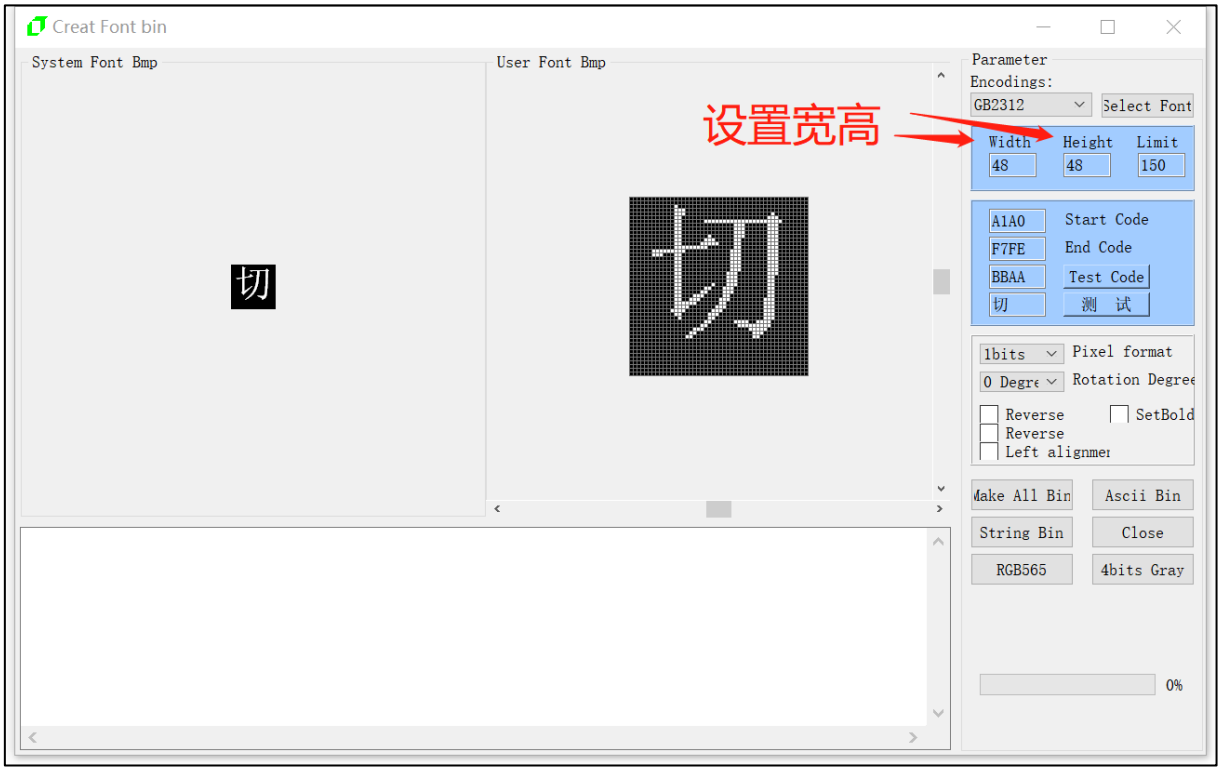


图 4-3：设置字库

© Levetop Semiconductor Co., Ltd.

4、点击【Make All Bin】即可输出字库 Bin 文件。**注意：**输入文件名时文件名中不能包含下面这些字符，如：? * / \ < > : " |，否则无法保存。

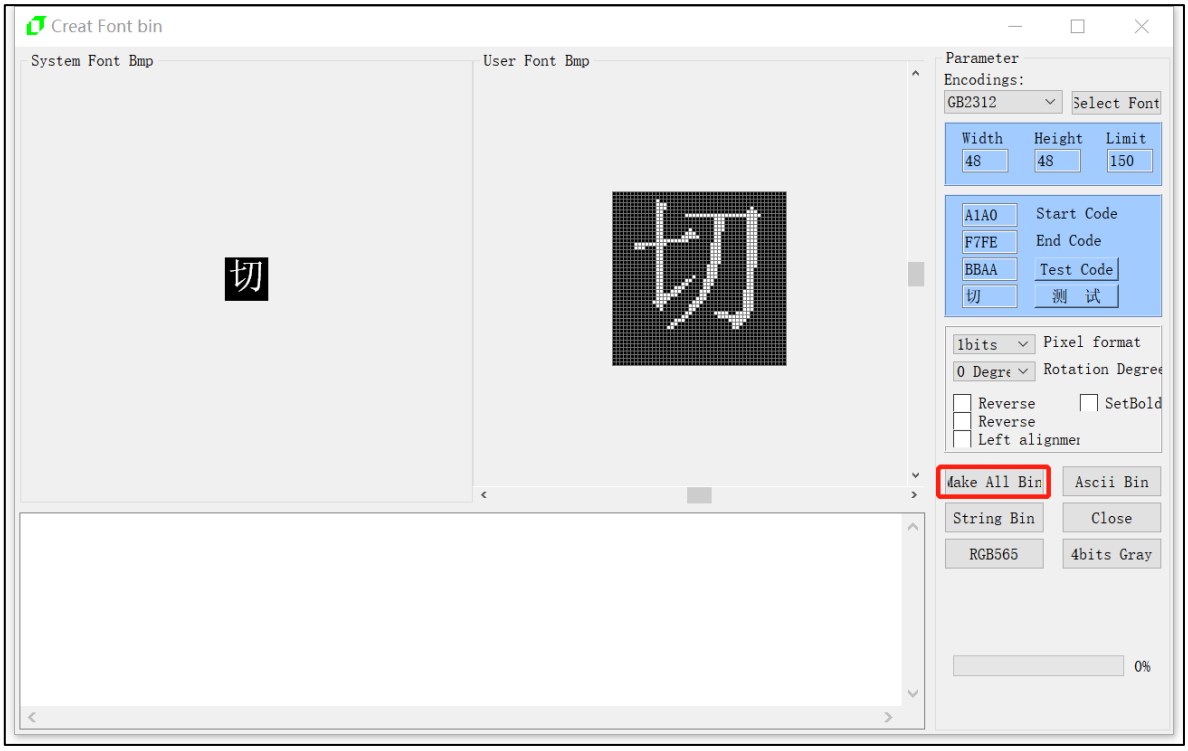


图 4-4：保存字库

当显示 Font Lib ok 时，即全字库制作成功：

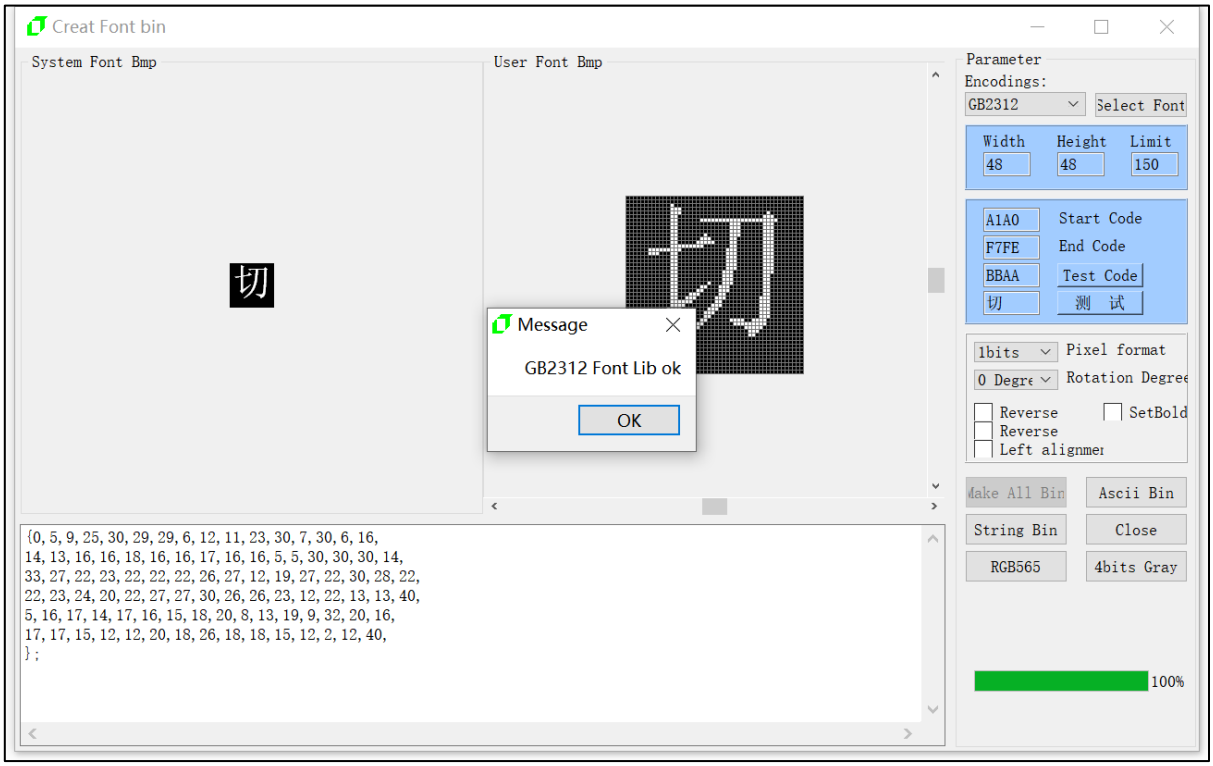


图 4-5：字库制作完成

© Levetop Semiconductor Co., Ltd.

5、制作完成后可以在目标文件夹中看到导出的 简体 16_16 宋体.bin 文件：



图 4-6：导出的字库 Bin 文件

4.1.2 自定义字库制作

由于大字库容量相对比较大，考虑到会占用 Flash 比较多的空间，因此该主控端提供了自定义字库的功能，客户可根据需要提前想好要调用的文字，从而仅仅将这部分文字打包成一个字库，进而减少 Flash 容量的浪费。对于自定义字库 Bin 文件的制作，使用者可以参考以下的说明，举例产生一个 48*48 的宋体自定义字库 Bin 文件：

1. 前期选择字体大小和形状，步骤与 4.2.5.1 全字库制作相同，点击【String Bin】后选择一个存放要调用文字的 txt 文档：

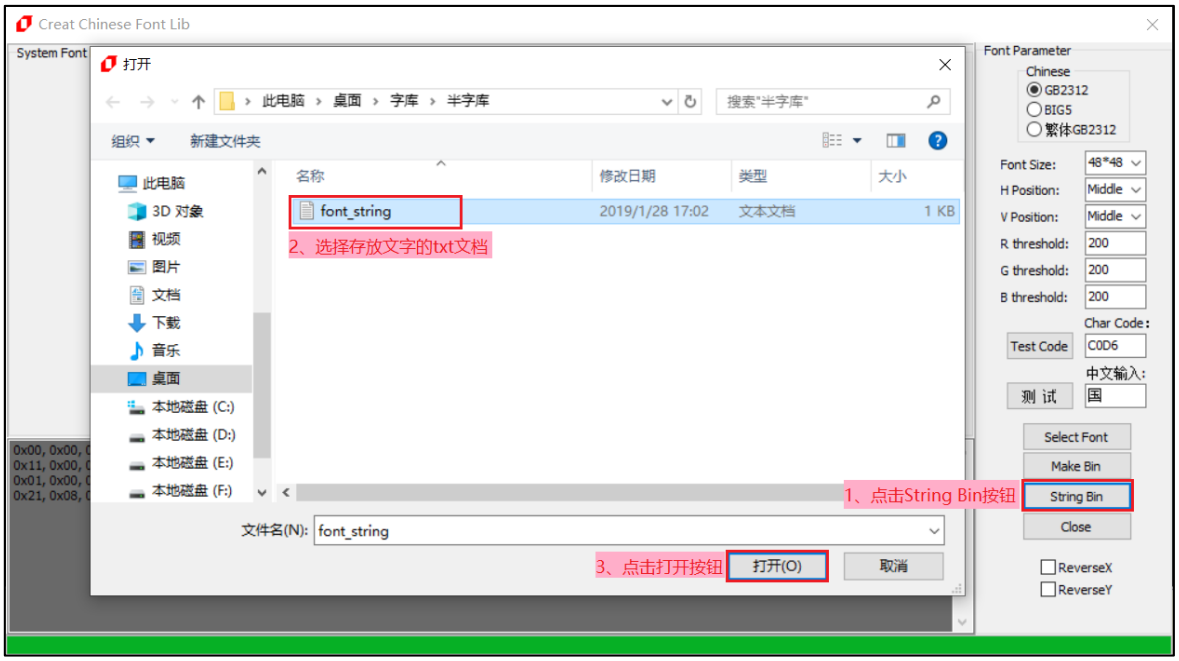


图 4-7：选择 txt 文档

© Levetop Semiconductor Co., Ltd.

2. 提前将需要调用显示的文字存放在一个记事本下：

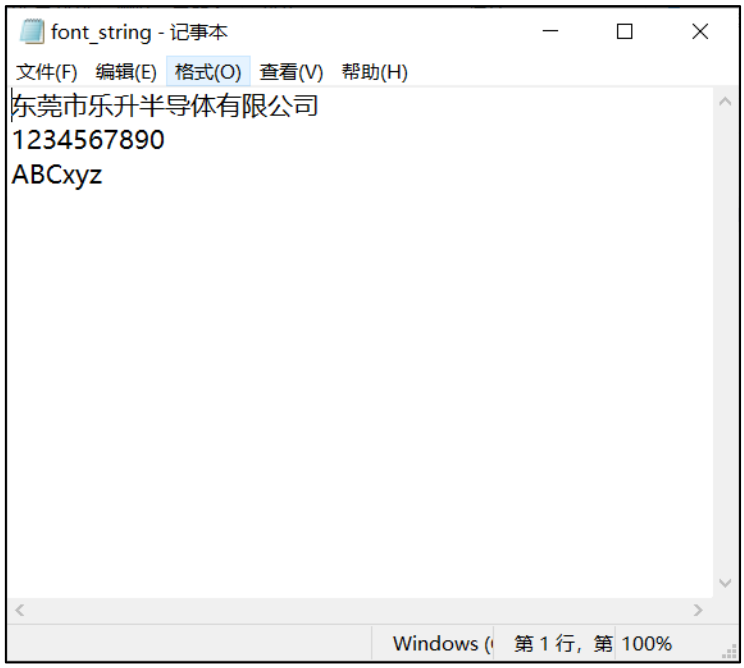


图 4-8：存放文字的 txt 文档

3. 指定一个路径存放将要生成的自定义字库，命名好后点击保存：

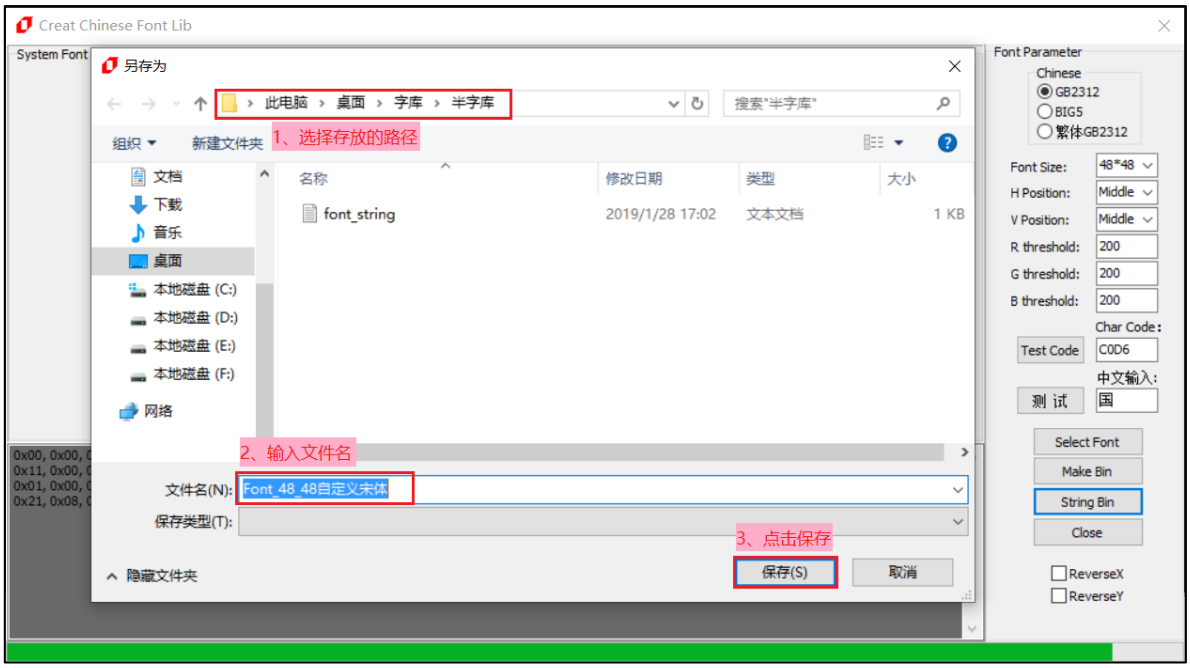


图 4-9：保存字库

© Levetop Semiconductor Co., Ltd.

当显示 String Font Lib ok 时，即自定义字库制作成功：

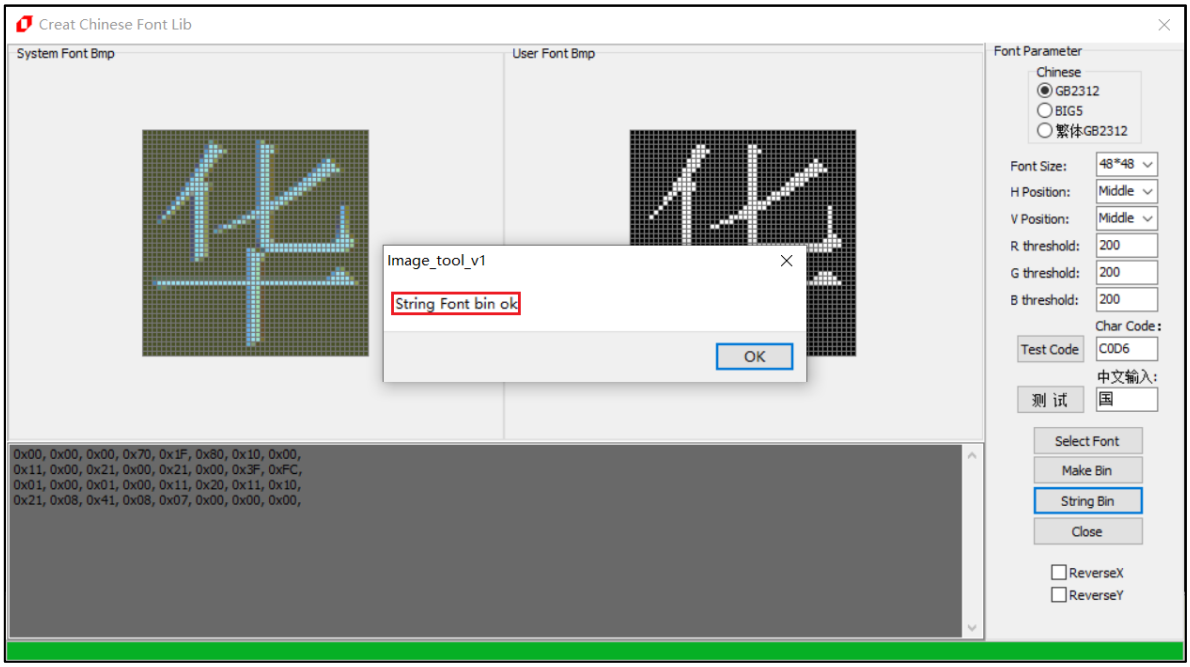


图 4-10：字库制作完成

4. 完成后可以在目标文件夹中看到导出的 Font_48_48 自定义宋体.bin 文件：

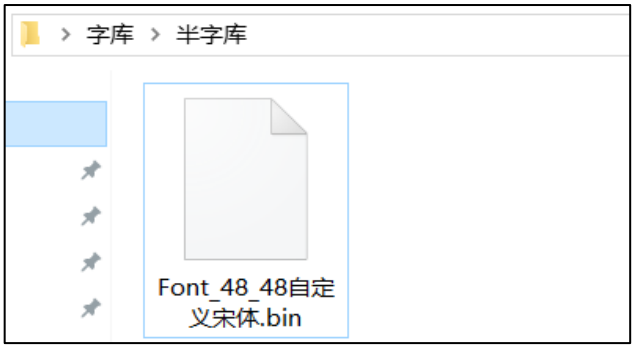


图 4-11：导出的字库 Bin 文件

© Levetop Semiconductor Co., Ltd.

4.2 制作 Wav 檔的 Bin 文件

4.2.1 音频文件转 WAV

1. 若音频素材格式不是 WAV 格式，则需要通过格式转换来获取 WAV 格式文件。下面以“格式工厂”免费版为转换平台进行操作。首先打开软件，选择音频，选择“-> WAV”，进入添加文件界面。

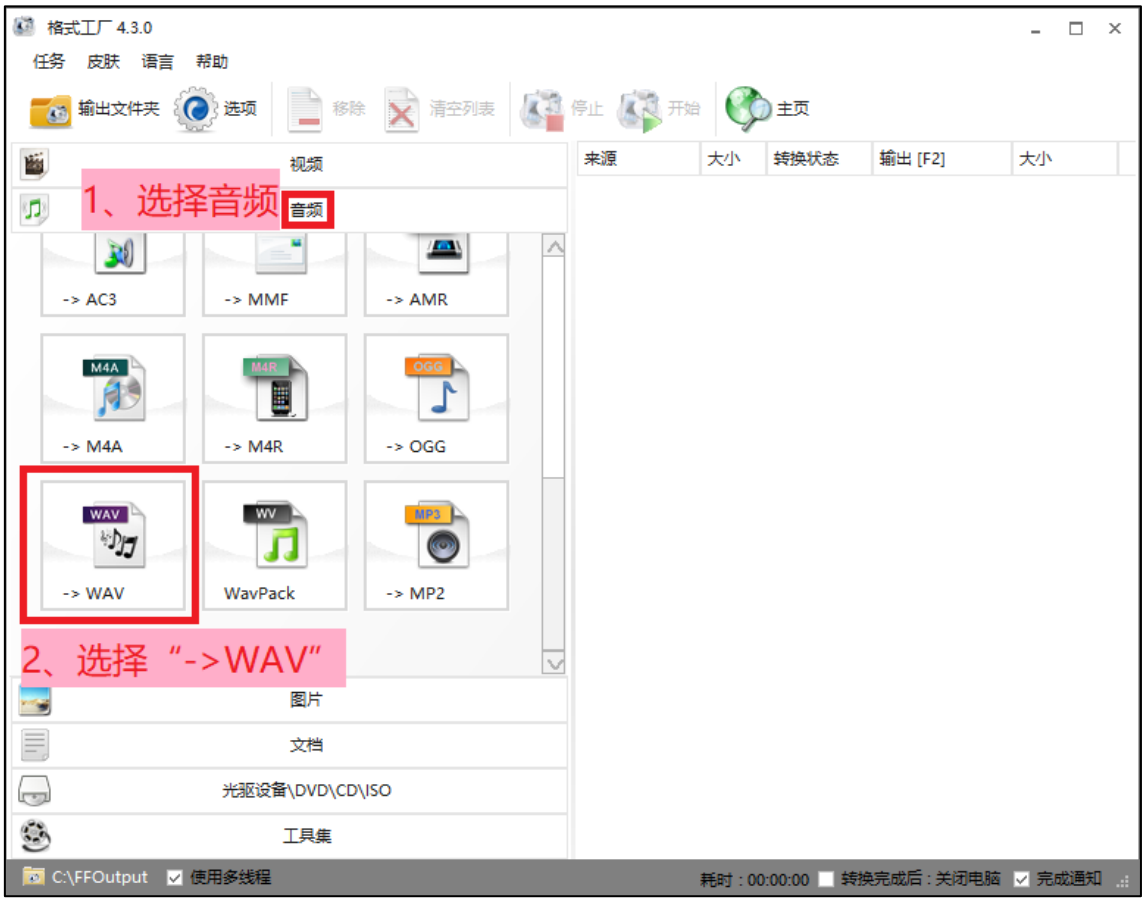


图 4-12：选择功能

© Levetop Semiconductor Co., Ltd.



图 4-13：wav 功能界面

2. 点击添加文件按钮，选择需要转换的音频文件。

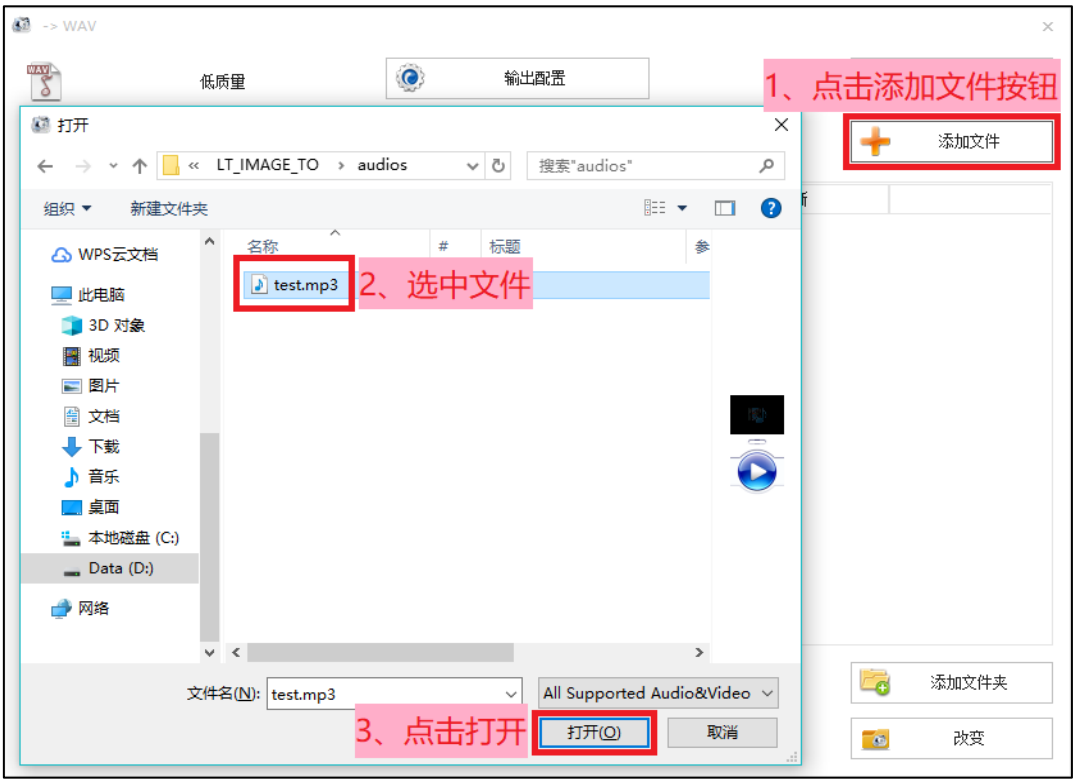


图 4-14：添加 wav 文件

© Levetop Semiconductor Co., Ltd.

3. 点击输出配置按钮，进行音频设置，其中，采样率可选 11025 或 22050。由于采用低成本的功放方案，同时为了节省 bin 文件存放空间，推荐选择采样率为 11025。



图 4-15：输出配置

4. 选中文件，点击剪辑按钮，进入音频剪辑界面，可选择需要的音频段。如不需要，请直接转到步骤 5。



图 4-16：进入剪辑功能

© Levetop Semiconductor Co., Ltd.

剪辑音频，可调节音量大小，以及截取其中的片段，剪辑完成后，点击确定按钮保存：

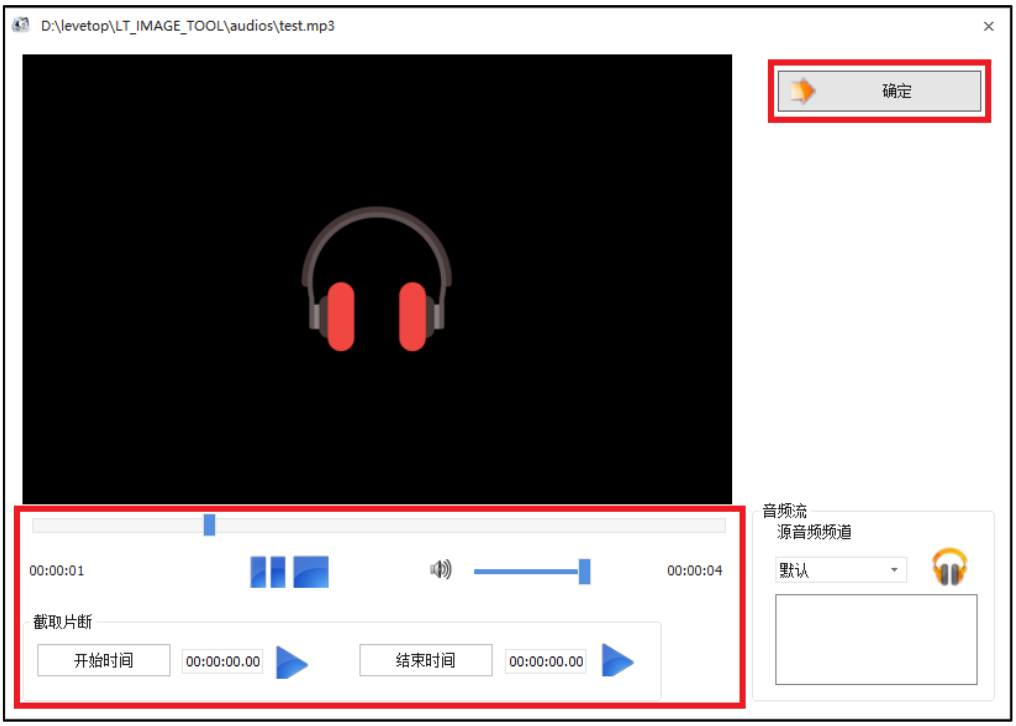


图 4-17：剪辑界面

5. 点击右下角的改变按钮，可重新选择输出目标文件夹，点击确定按钮添加任务。

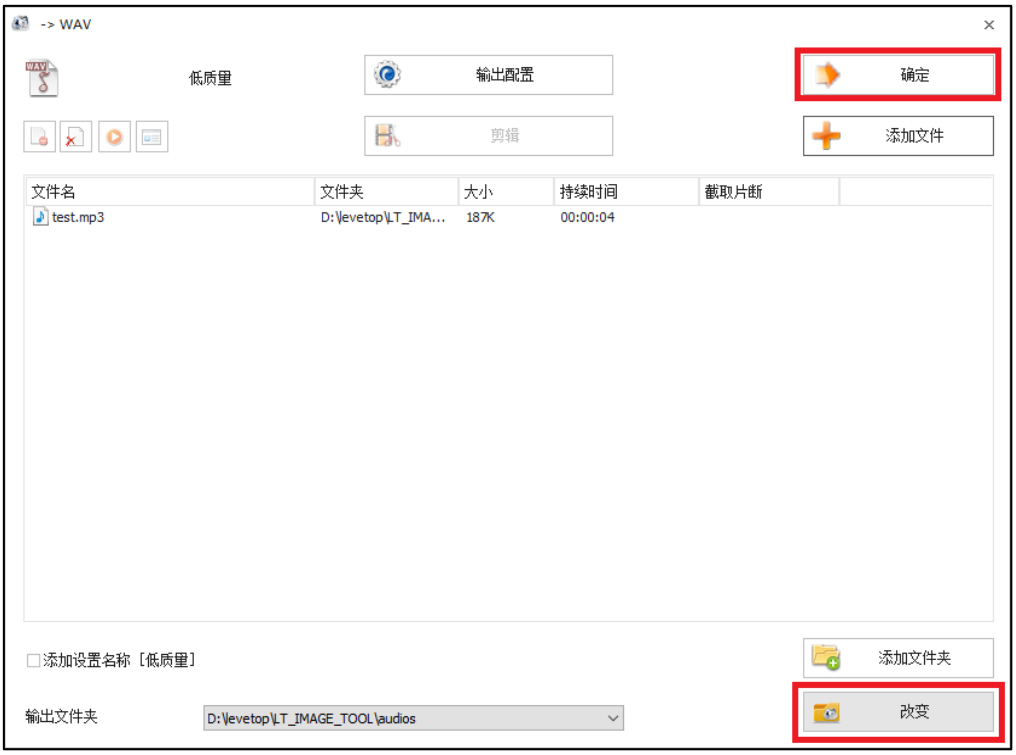


图 4-18：选择输出文件夹

© Levetop Semiconductor Co., Ltd.

6. 点击开始按钮，开始转换，转换完毕后，可在目标文件夹查看导出的 WAV 文件。



图 4-19：开始转换



图 4-20：导出的 wav 文件

© Levetop Semiconductor Co., Ltd.

4.2.2 制作 WAV 档 Bin 文件

1. 执行“WavTool.exe”软件即可打开 WAV Bin 文件制作界面：

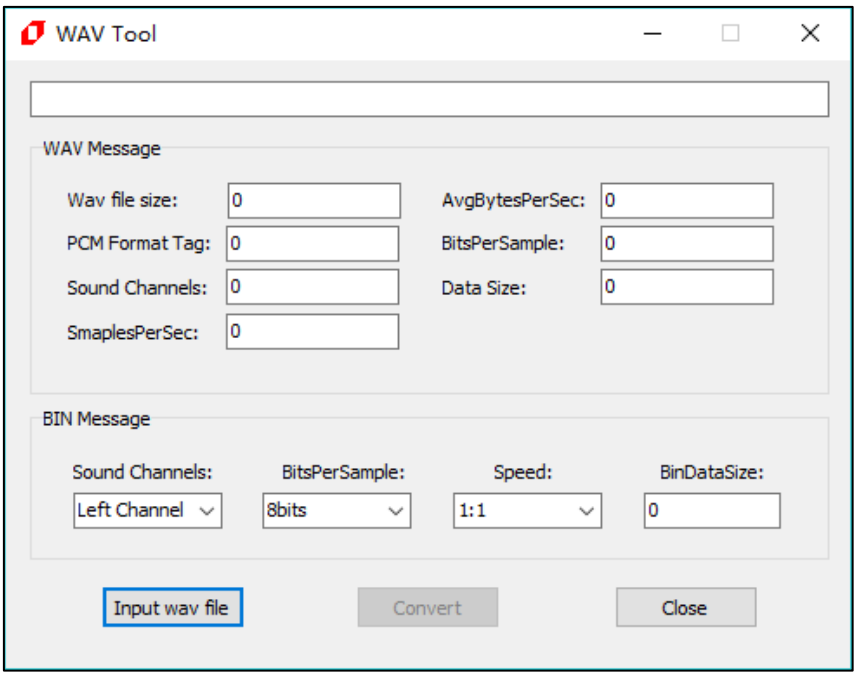


图 4-21：WAV Bin 文件制作界面

2. 导入 WAV 文件，点击 Input wav file 按钮，选择需要转换的 WAV 文件，点击打开，即可添加：

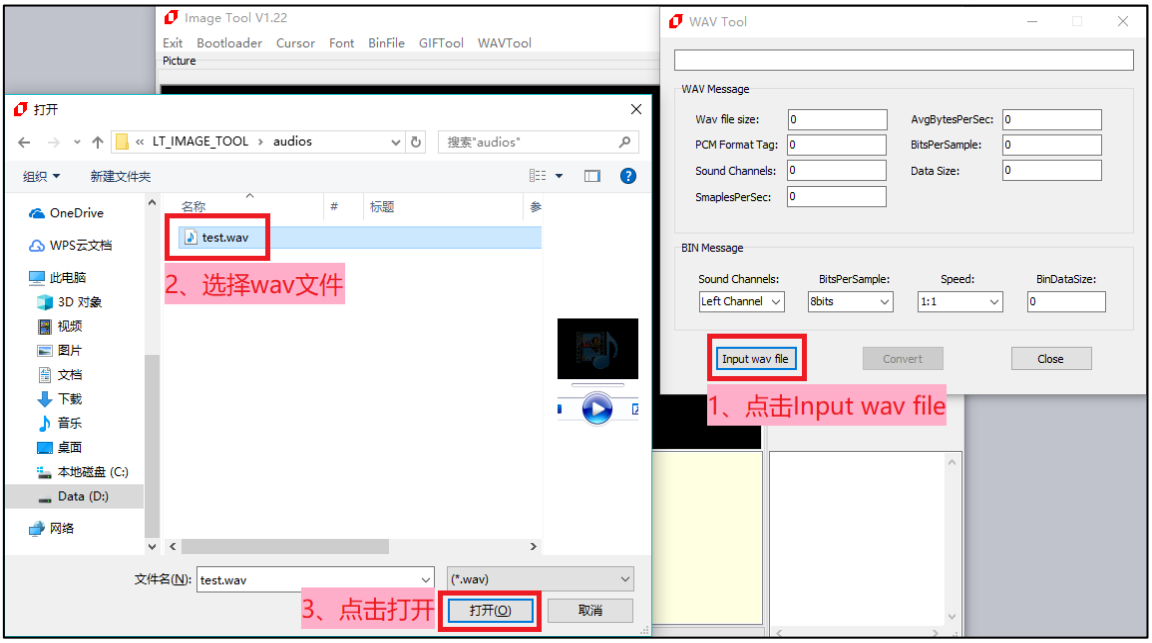


图 4-22：添加 wav 文件

© Levetop Semiconductor Co., Ltd.

添加成功后，在界面上方，可以看到文件所在路径，并显示 wav 文件的相关数据（如下图）。如果采样率（SamplesPerSec）不是 11025 或 22050，建议改变 Speed 的比率，或是先用其他音频软件将采样率变成 11025 或 22050。

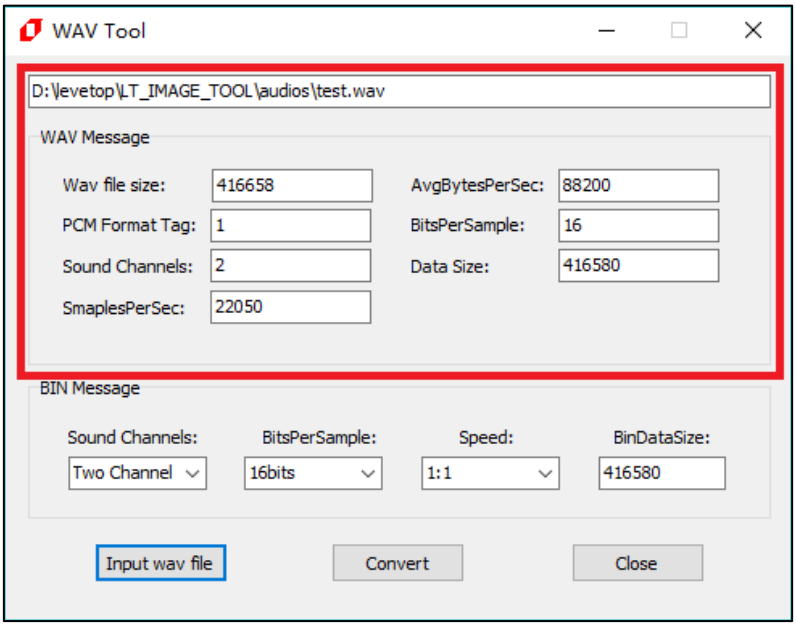


图 4-23：添加成功

3. 设置 bin 文件的参数，Sound Channels 选项可选左声道、右声道或双立体声道，BitsPerSample 选项可选 8bits 或 16bits。一般选用单声道、8bits 就足够，bin 文件较小，便于储存。Speed 选项选择生成 bin 文件时的采样速度，速度越高，音质也会相对下降，同时需要改变程序的定时器更新数据的时间。

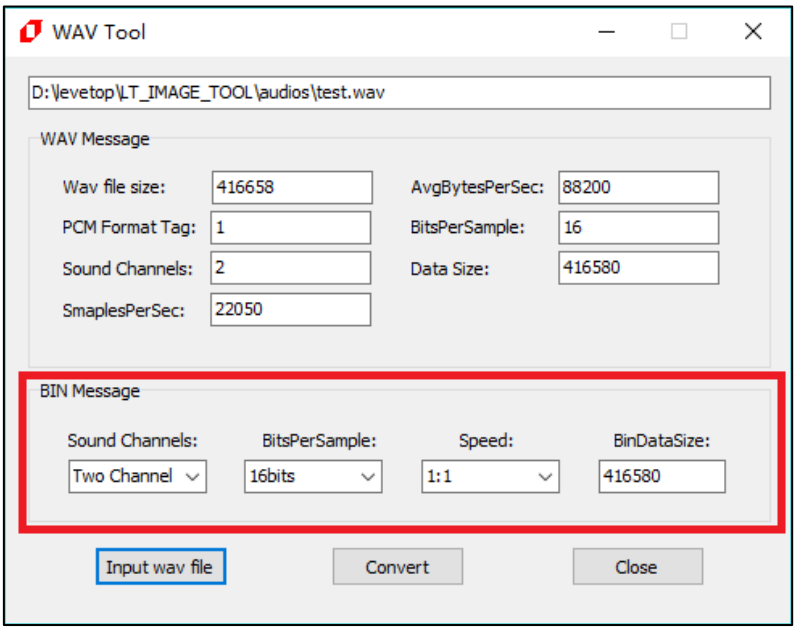


图 4-24：WAV Bin 文件制作界面

© Levetop Semiconductor Co., Ltd.

4. 导出 bin 文件，**注意：**输入文件名时文件名中不能包含下面这些字符，如：? * / \ < > : " |，否则无法保存。

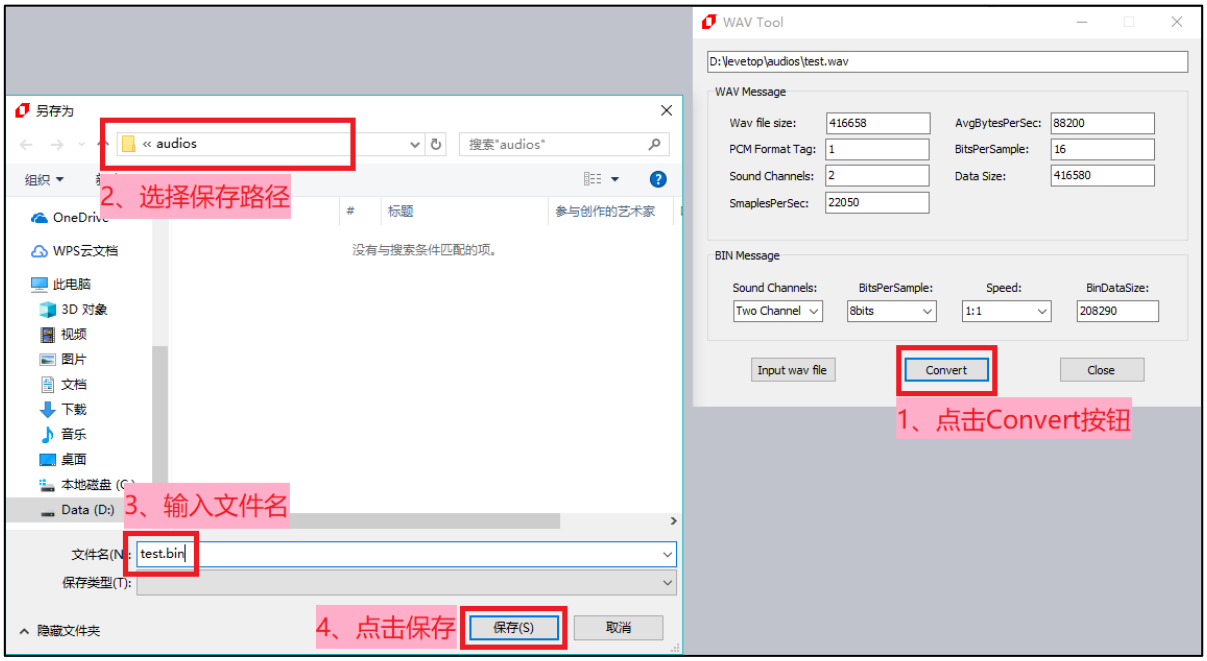


图 4-25：导出 bin 文件

若该音频文件与 bin 设置参数不符合，则会提示 Can't suport this bits，请按照步骤 3 重新设置：

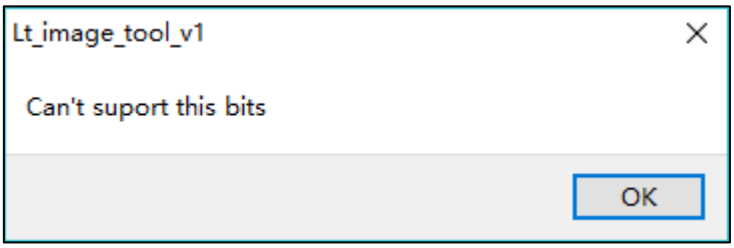


图 4-26：错误提示

若符合要求，则点击 OK 按钮开始导出：

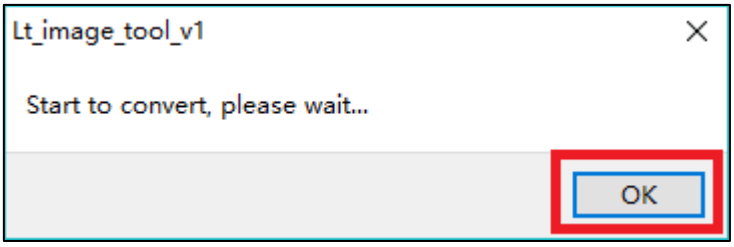


图 4-27：点击 OK 导出

© Levetop Semiconductor Co., Ltd.

5. 成功导出 WAV bin 文件，并可在目标文件夹看到导出的 test.bin 文件：



图 4-28：导出成功

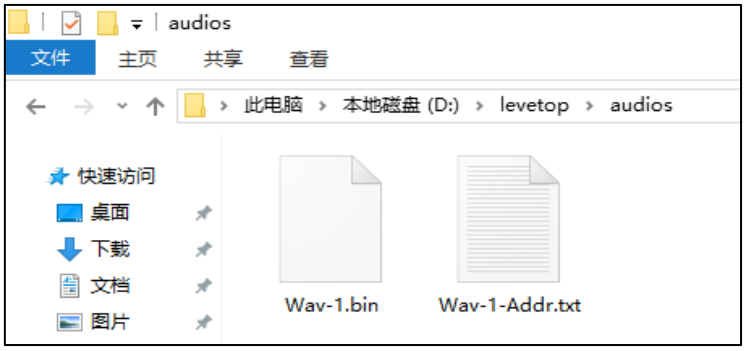


图 4-29：生成的 bin 文件

4.2.3 典型音频驱动电路

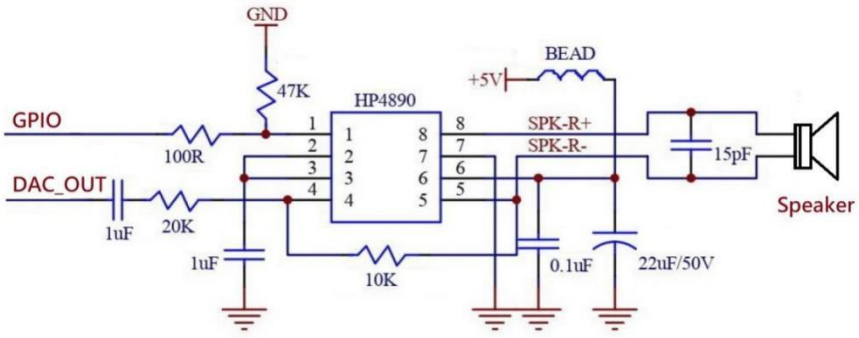


图 4-30：用 DAC_OUT 输出及小功率放大器的音频驱动电路

© Levetop Semiconductor Co., Ltd.

5. 版权说明

本文件之版权属于 深圳市乐升半导体 所有，若需要复制或复印请事先得到 乐升半导体 的许可。本文件记载之信息虽然都有经过校对，但是 乐升半导体 对文件使用说明的规格不承担任何责任，文件内提到的应用程序仅用于参考，乐升半导体 不保证此类应用程序不需要进一步修改。乐升半导体 保留在不事先通知的情况下更改其产品规格或文件的权利。有关最新产品信息，请访问我们的网站 **Https://www.levetop.cn** 。