

**LT768x**

## TFT-LCD 绘图加速控制芯片

*High Performance TFT-LCD Graphics Controller*

---

### Bin 文件整合软件

(LT\_IMAGE\_TOOL.EXE)

## 使用说明书

V1.2

[www.levetop.cn](http://www.levetop.cn)

Levetop Semiconductor Co., Ltd.

# LT IMAGE TOOL 使用说明书

LT\_IMAGE\_TOOL.EXE 是 乐升半导体 提供的一个专用程序, 针对 LT768x TFT 控制器外接的 SPI Flash, 可以制作图片 Bin 文件、字库 Bin 文件、GIF 檔的 Bin 文件, 以及制作 LT768x 的开机启动程序、图形光标, 然后将这些 Bin 文件整合起来, 产生可以烧录到 SPI Flash 的 Bin 文件, 此程序的 6 个功能, 分别为:

- 一. 制作「图片 Bin 文件」
- 二. 制作「字库 Bin 文件」
- 三. 制作「GIF 檔 Bin 文件」
- 四. 制作「图形光标」
- 五. 设置「开机启动加载程序」
- 六. Bin 文件整合

## 一、制作「图片 Bin 文件」

1. 打开软件 LT\_IMAGE\_TOOL.EXE:

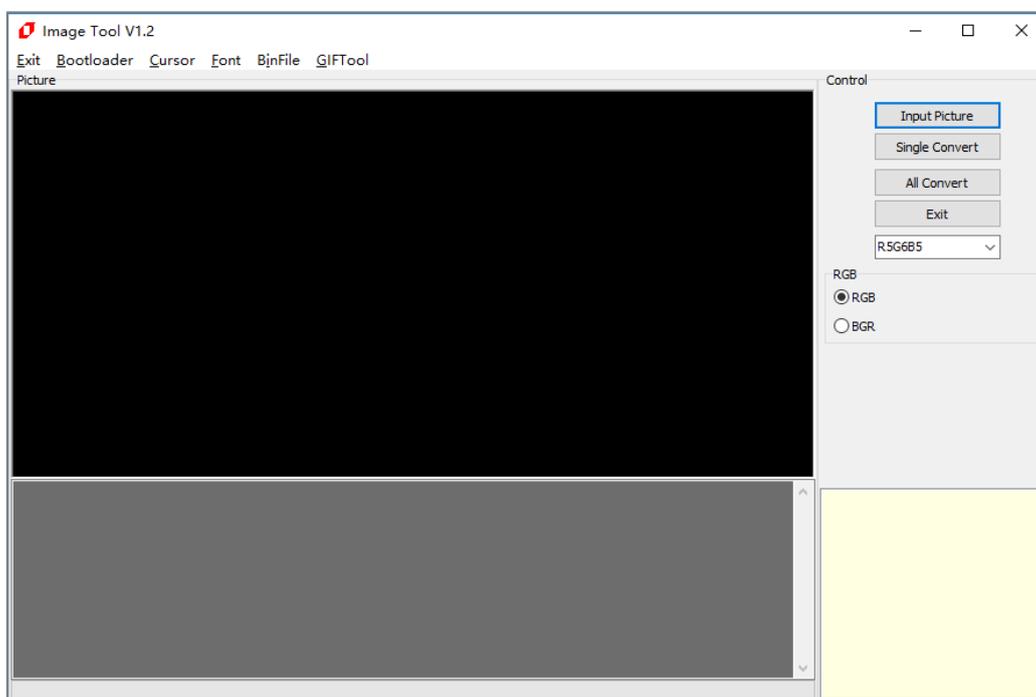


图 1-1: 执行 LT\_IMAGE\_TOOL.EXE

2. 导入图片，点击 Input Picture 按钮，选择需要的图片，点击打开，即可添加此文件夹下的所有图片：

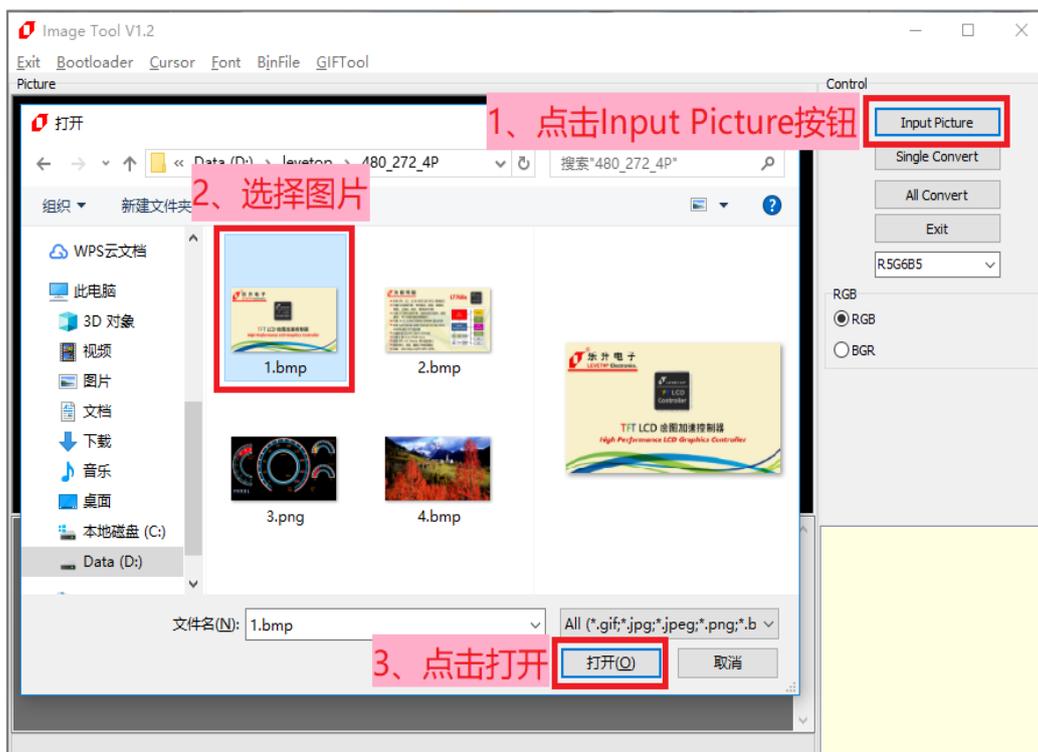


图 1-2：导入图片

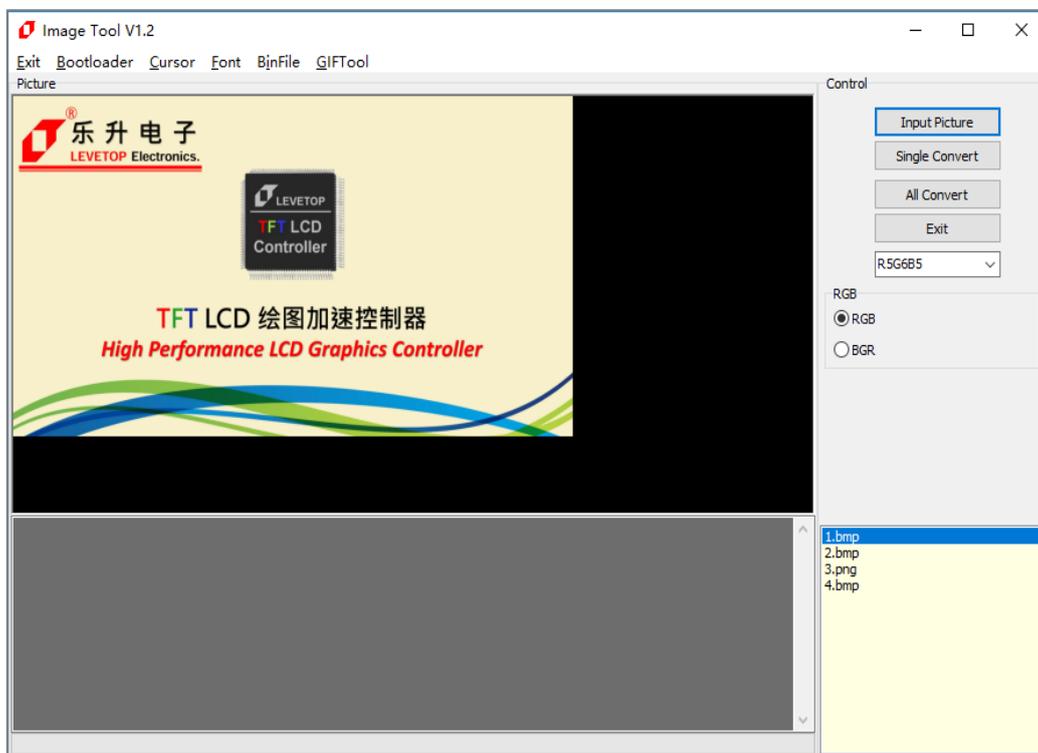


图 1-3：导入完成

3. 图片输出设定, 可选 16bpp 或 24bpp 或 Black\_White 格式, 以及 RGB 或 BGR 格式:  
 (注意: 若需要制作 Black\_White 格式的图片 bin 文件, 源图片文件必须为只有黑色和白色的图片。)



图 1-4: 设定输出格式

4. 导出某一张图片或导出全部图片, 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。

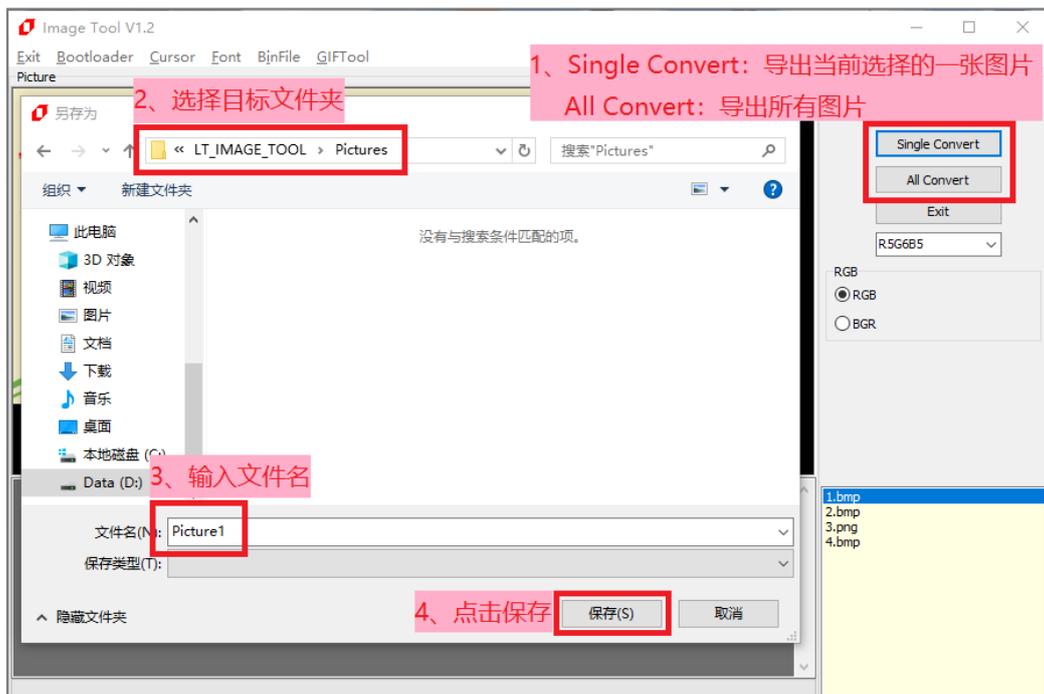


图 1-5: 导出图片 (1/2)

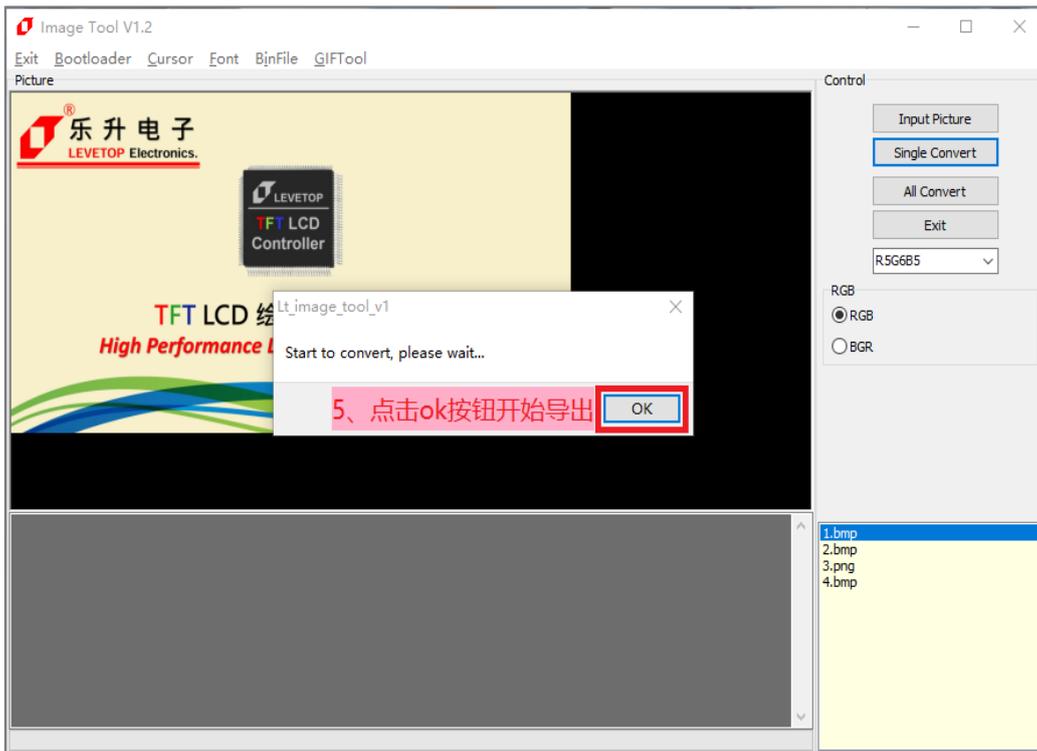


图 1-6: 导出图片 (2/2)

5. 成功导出图片 Bin 文件:

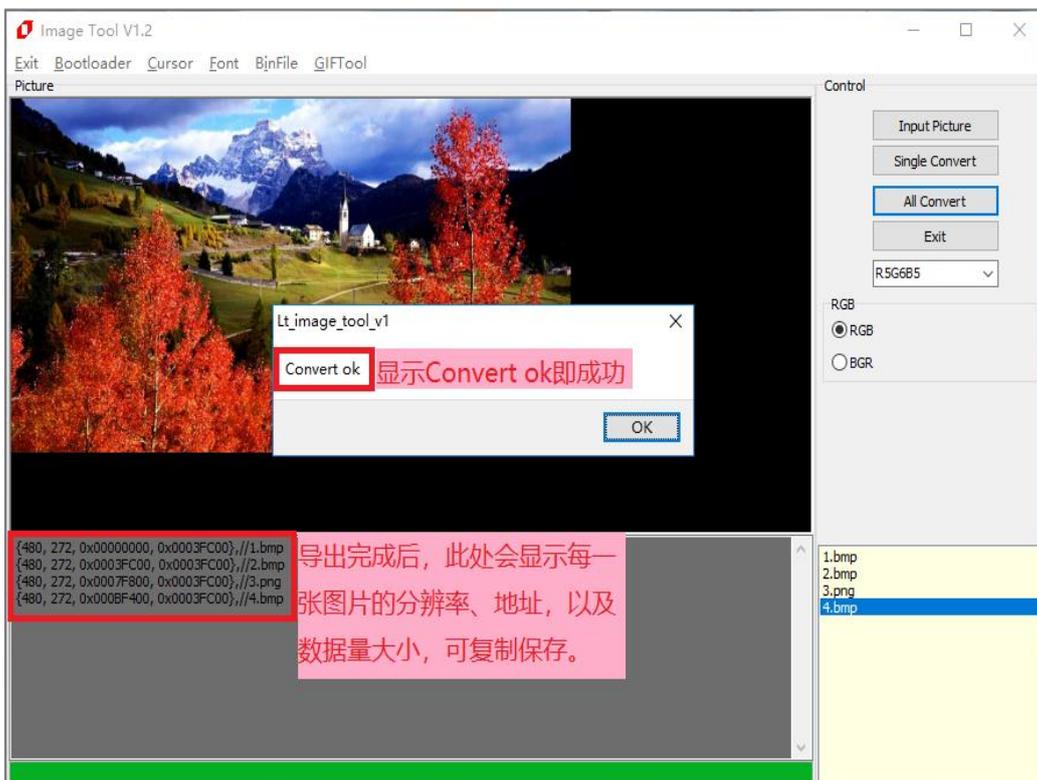


图 1-7: 导出成功

6. 导出图片后可以在目标文件夹中看到导出的 **Picture1.bin** 文件：

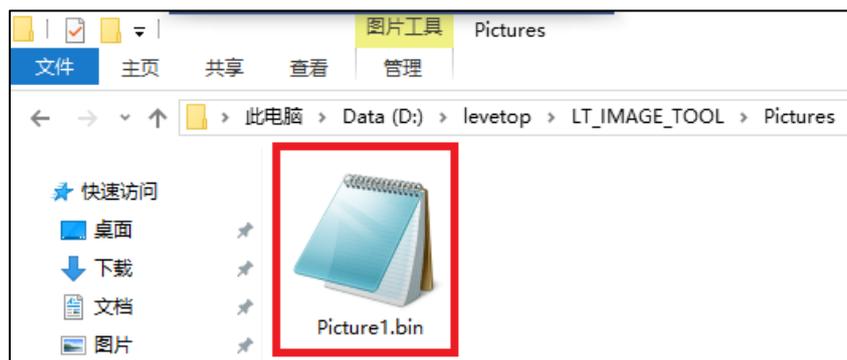


图 1-8: 导出的图片 Bin 文件

## 二、制作「字库 Bin 文件」

1. 点击【LT\_IMAGE\_TOOL 菜单>Font】即可打开中文字库 Bin 文件制作界面：可选择 GB2312 简体字库、BIG5 字库以及 GB2312 繁体字库。

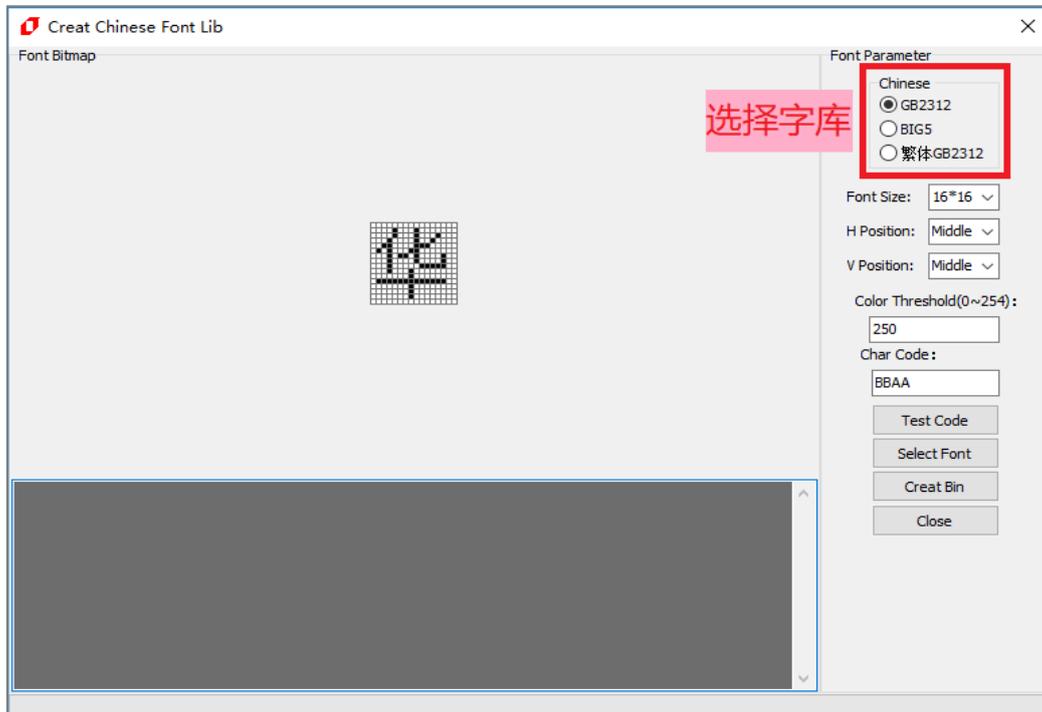


图 2-1：制作中文字库

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：

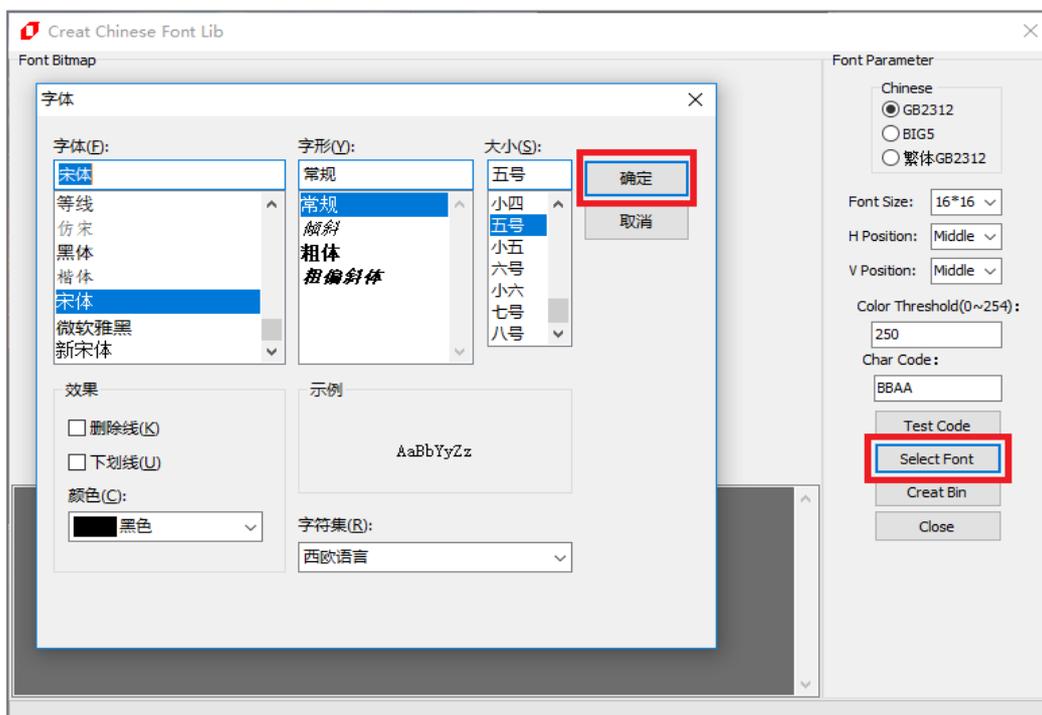


图 2-2：选择字体

3. 设置字库有 16\*16、24\*24、32\*32、48\*48、72\*72 五种字体大小, 你也可以设置字体横向(偏左、居中、偏右)和纵向(偏上、居中、偏下)的位置、颜色阈值(0~254)、和预览文字, 点击【Test Code】按钮即可查看该字符的数据。

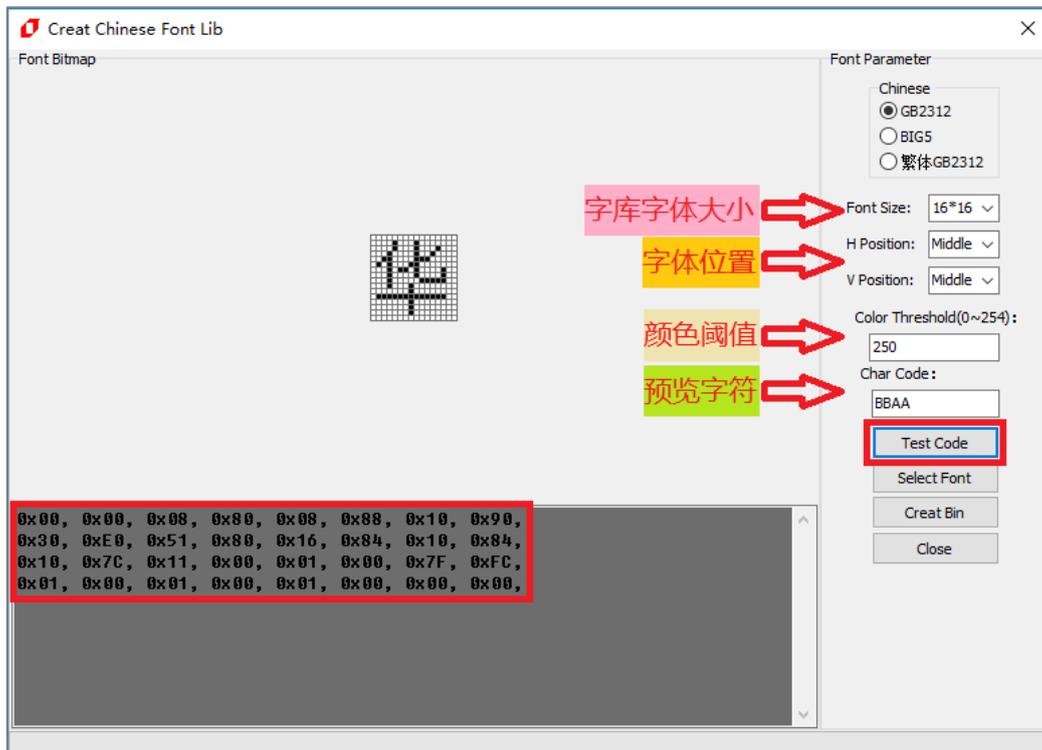


图 2-3: 设置字库

4. 点击【Create 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。

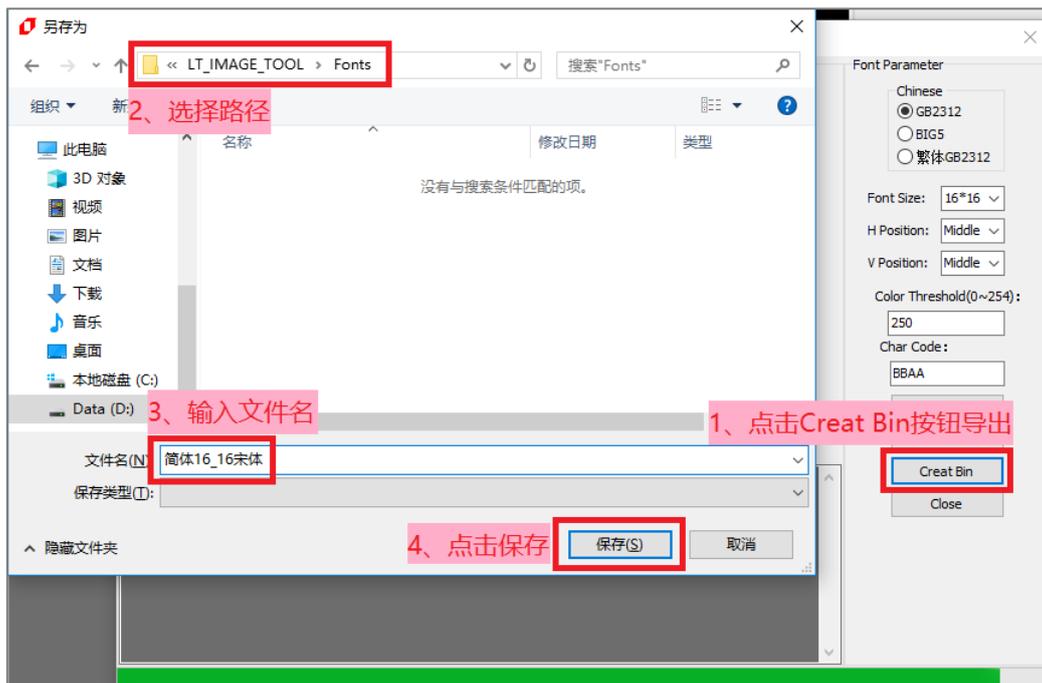


图 2-4: 保存字库

当显示 (字库)+Font Lib ok 时, 即保存成功:

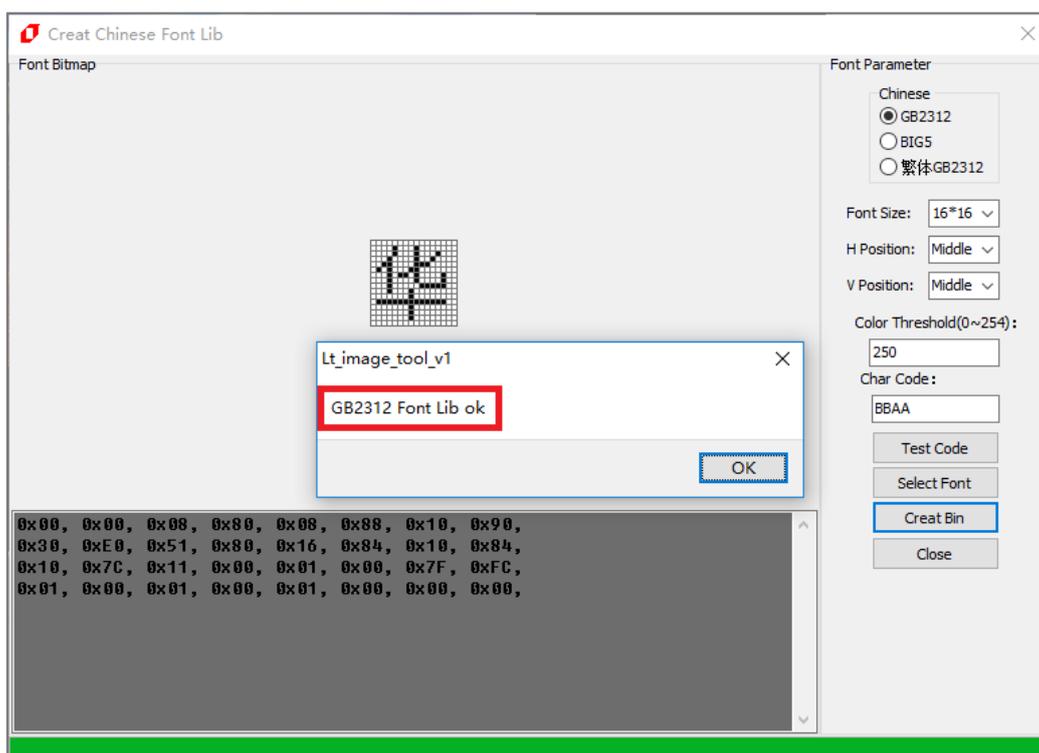


图 2-5: 字库制作完成

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:

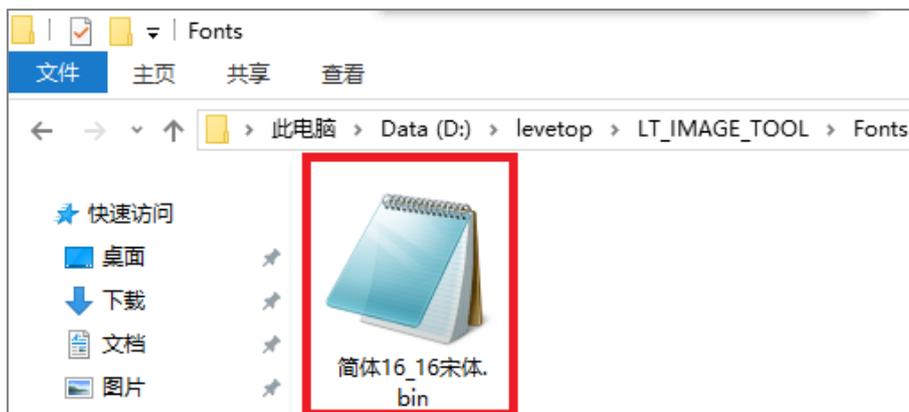


图 2-6: 导出的字库 Bin 文件

### 三、制作「GIF 檔 Bin 文件」

1. 点击【LT\_IMAGE\_TOOL 菜单>GIFTool】即可打开 GIF Bin 文件制作界面：

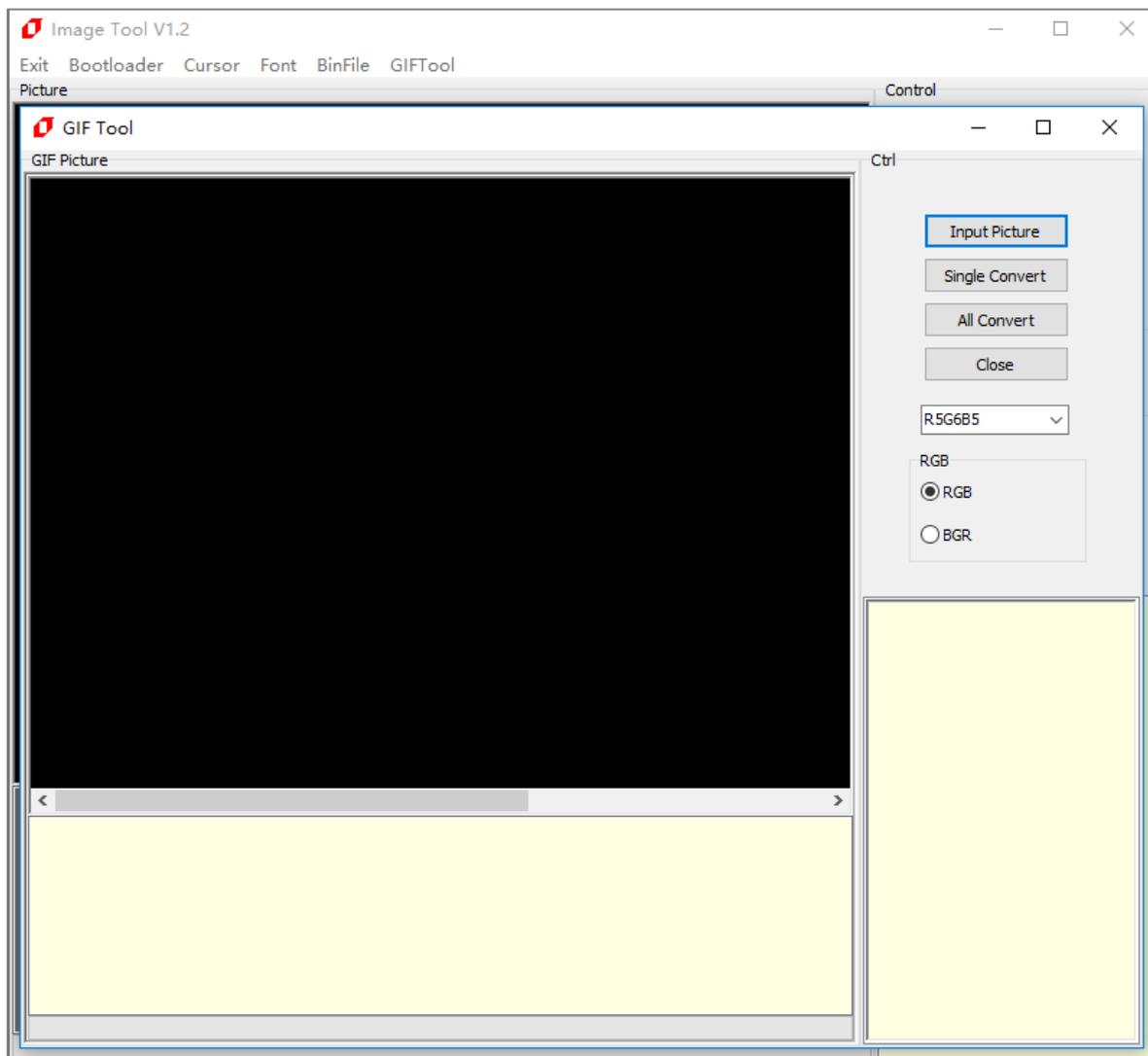


图 3-1：打开 GIF Bin 文件制作界面

2. 导入 GIF 图片，点击 Input Picture 按钮，选择需要的 GIF 图片，点击打开，即可添加此文件夹下的所有 GIF 图片：

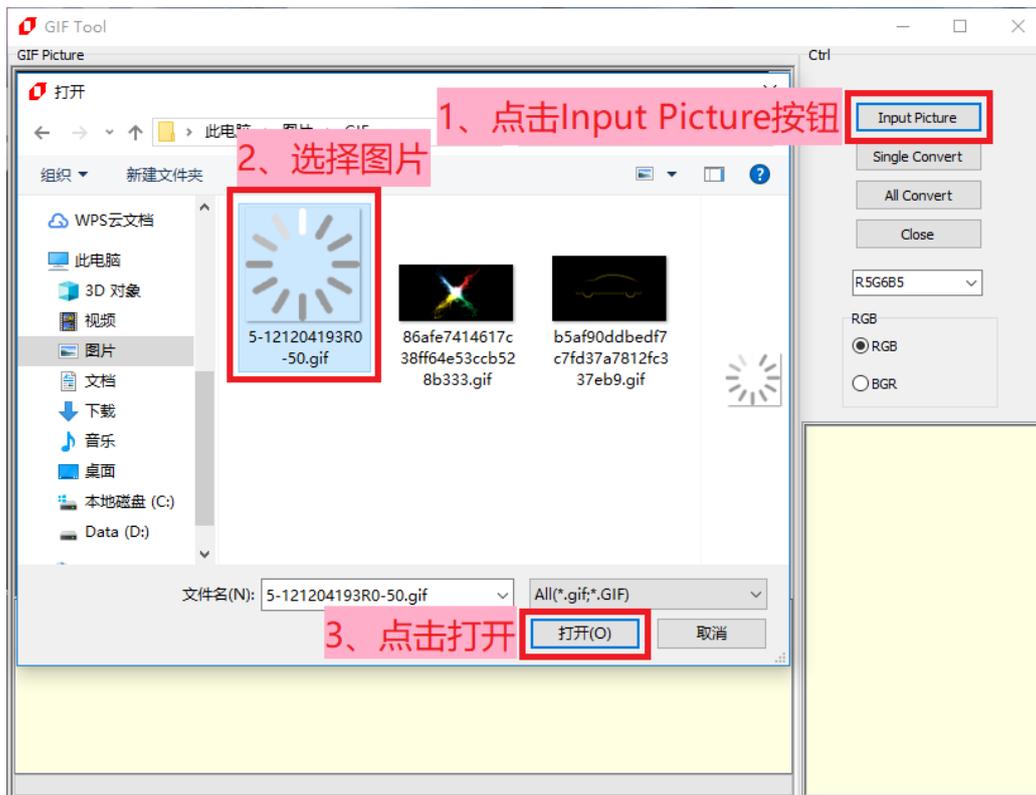


图 3-2：导入图片

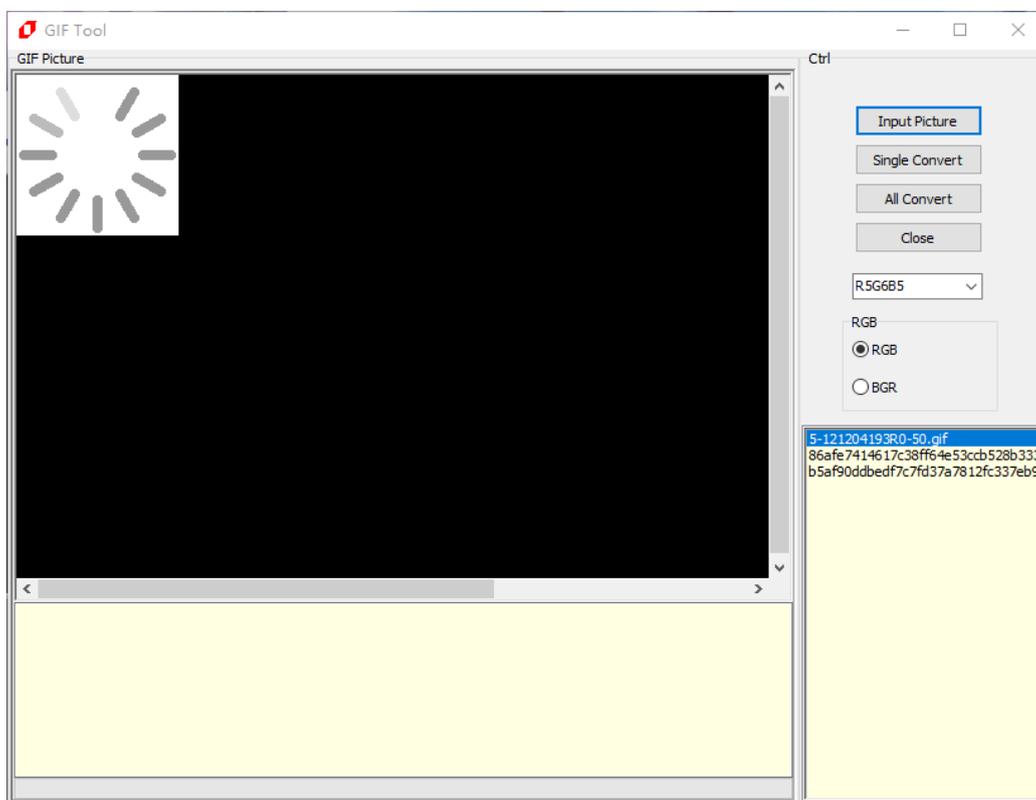


图 3-3：导入完成

3. GIF 图片输出设定, 可选 16bpp 或 24bpp 格式, 以及 RGB 或 BGR 格式:

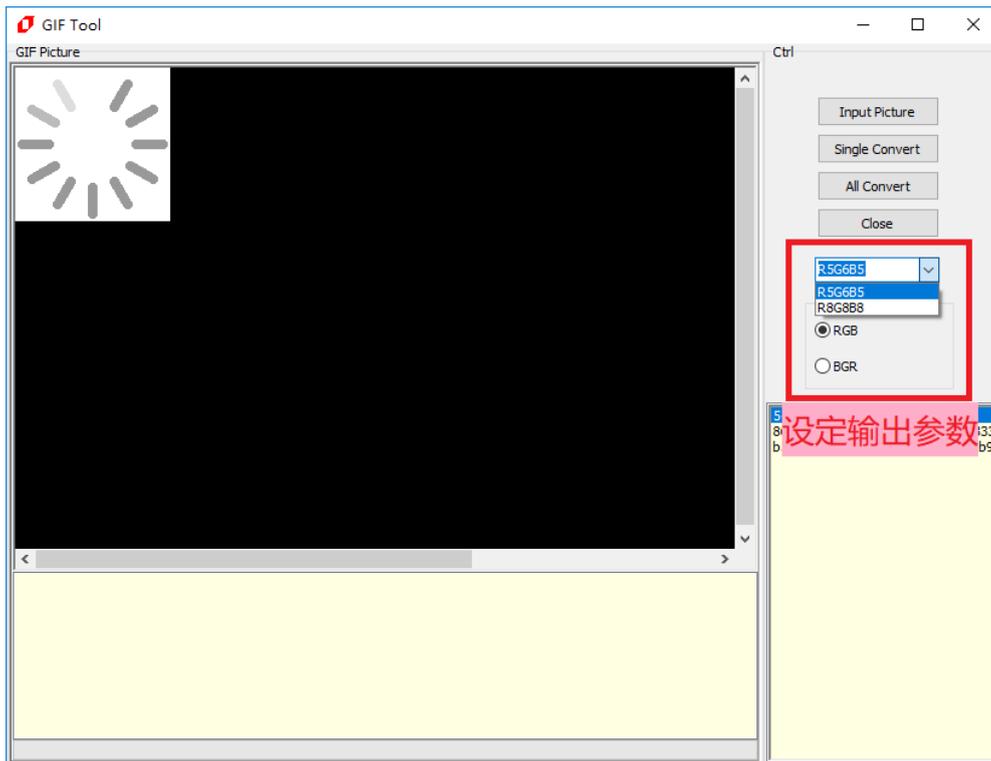


图 3-4: 设定输出格式

4. 导出某一张 GIF 图片或导出全部 GIF 图片, 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。

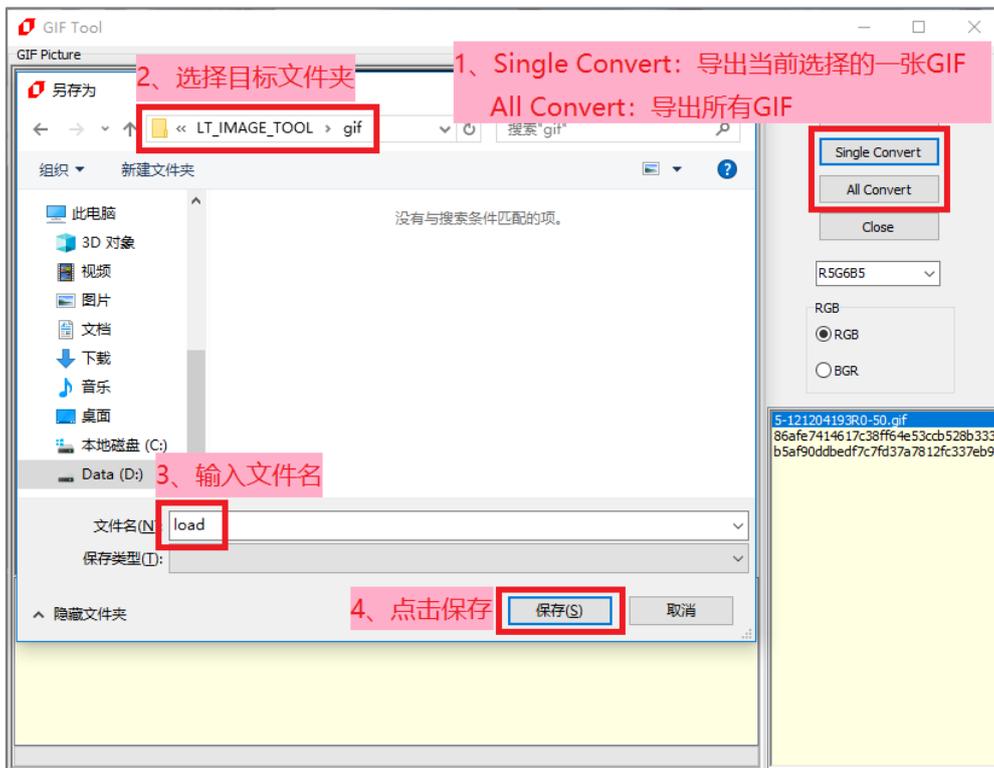


图 3-5: 导出图片 (1/2)

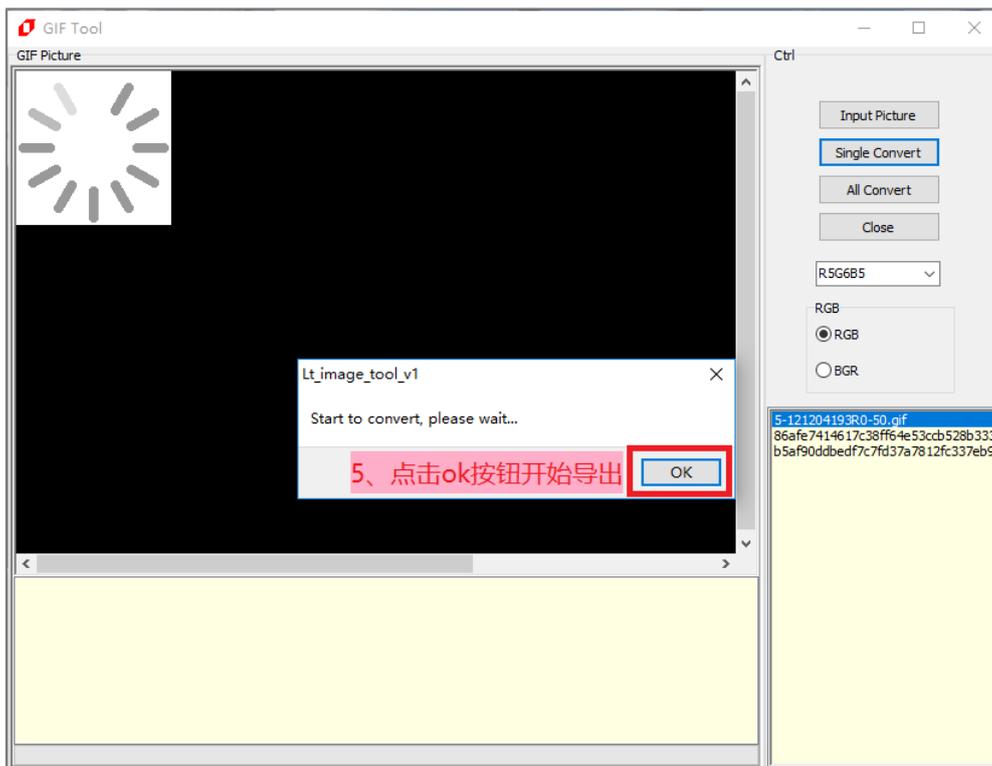


图 3-6: 导出图片 (2/2)

5. 成功导出图片 Bin 文件:

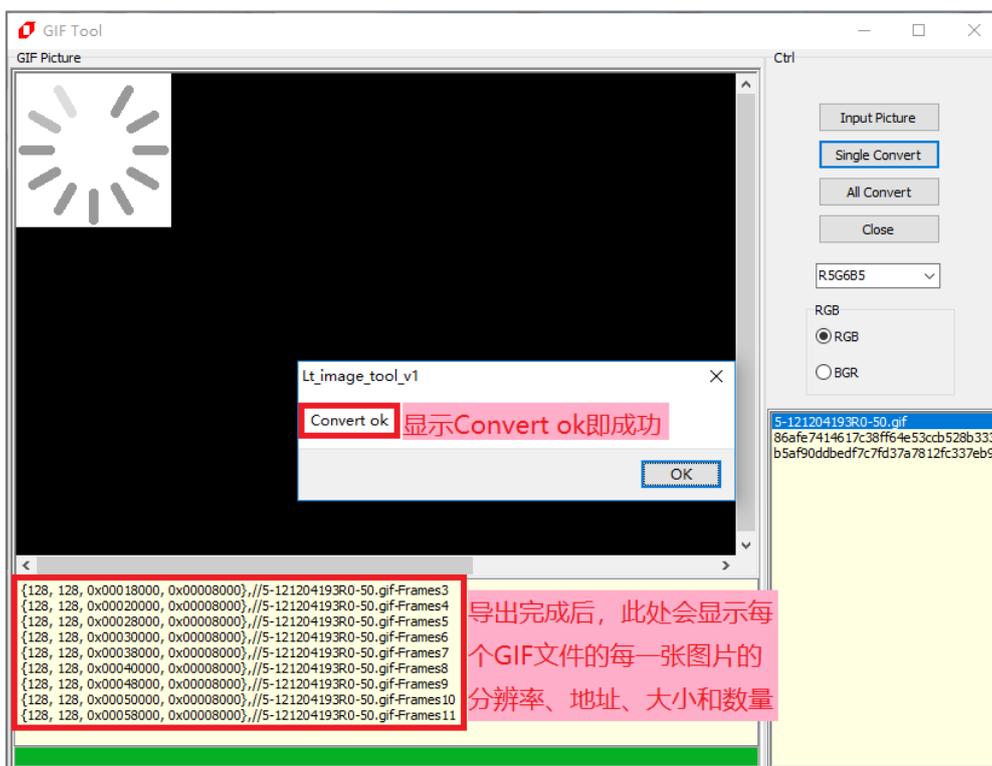


图 3-7: 导出成功

6. 导出图片后可以在目标文件夹中看到导出的 load.bin 文件以及 txt 详情文档:

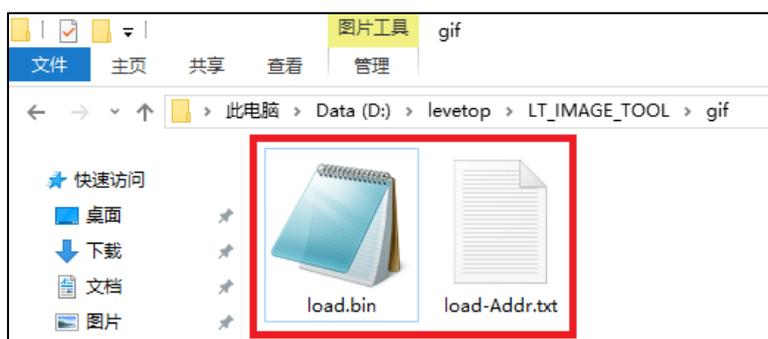


图 3-8: 导出的图片 Bin 文件

在 txt 文档内能看到 Gif 内每张图片的分辨率、地址、大小以及图片数量:

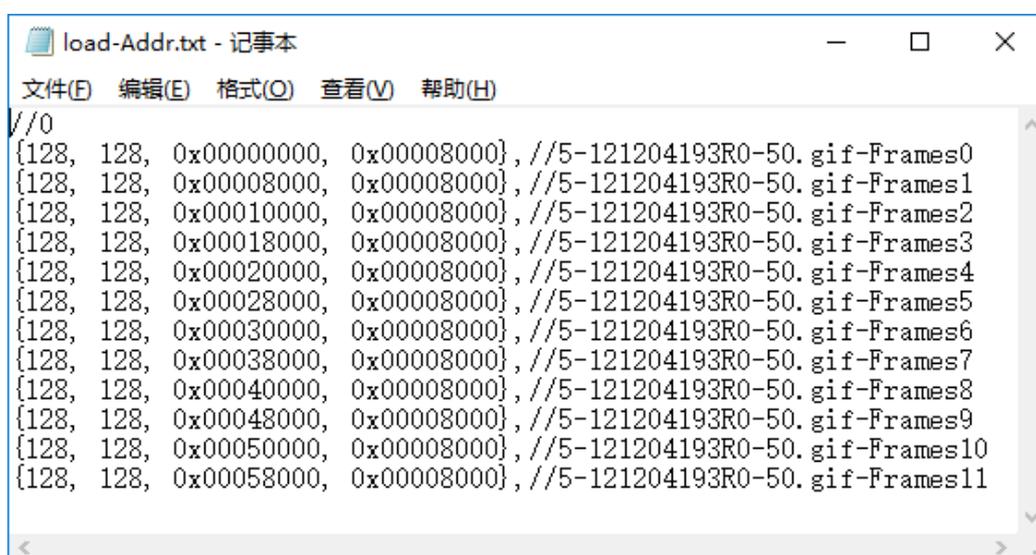


图 3-9: 导出的每张图片信息

## 四、制作及修改「图形光标」

### 1、制作图形光标：

a、点击【LT\_IMAGE\_TOOL 菜单>Cursor】即可打开图形光标制作界面：

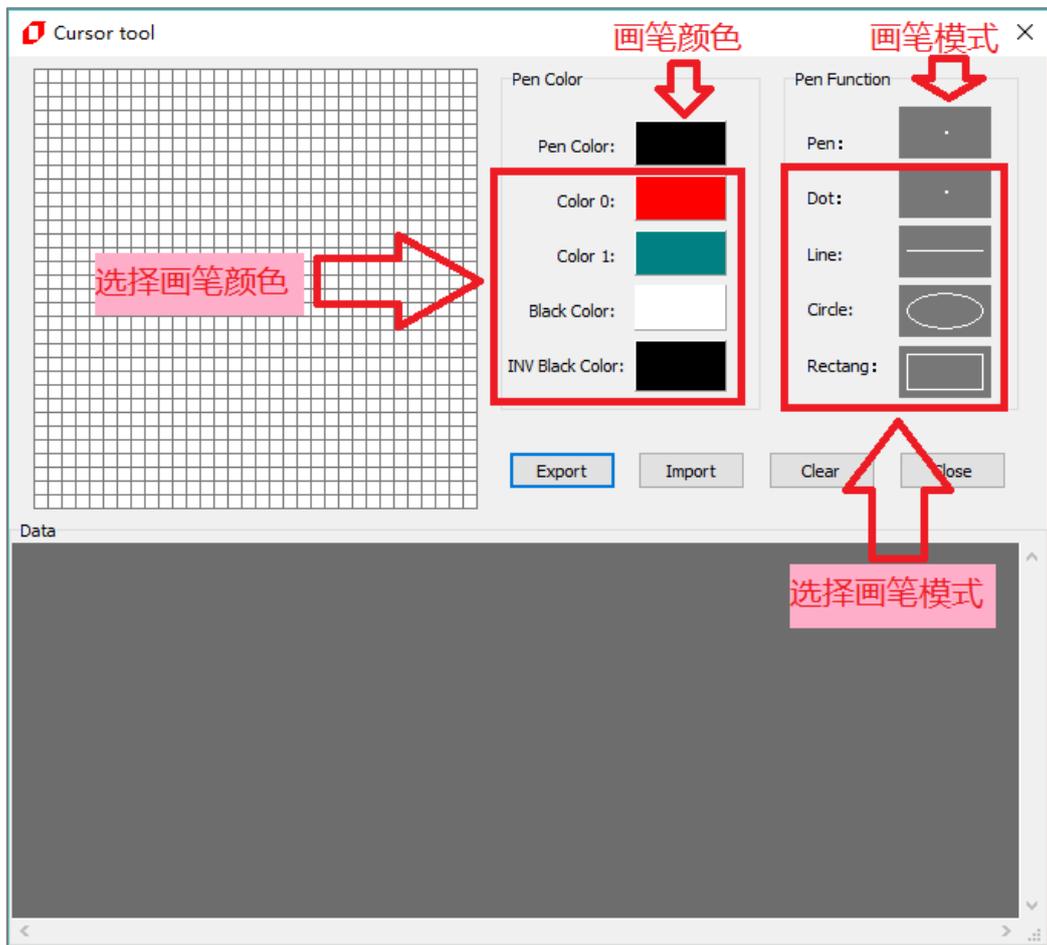


图 4-1：打开图像光标

b、画好图形光标后，点击【Export】按钮，即可看到图形光标数据，位于 Data 区域。

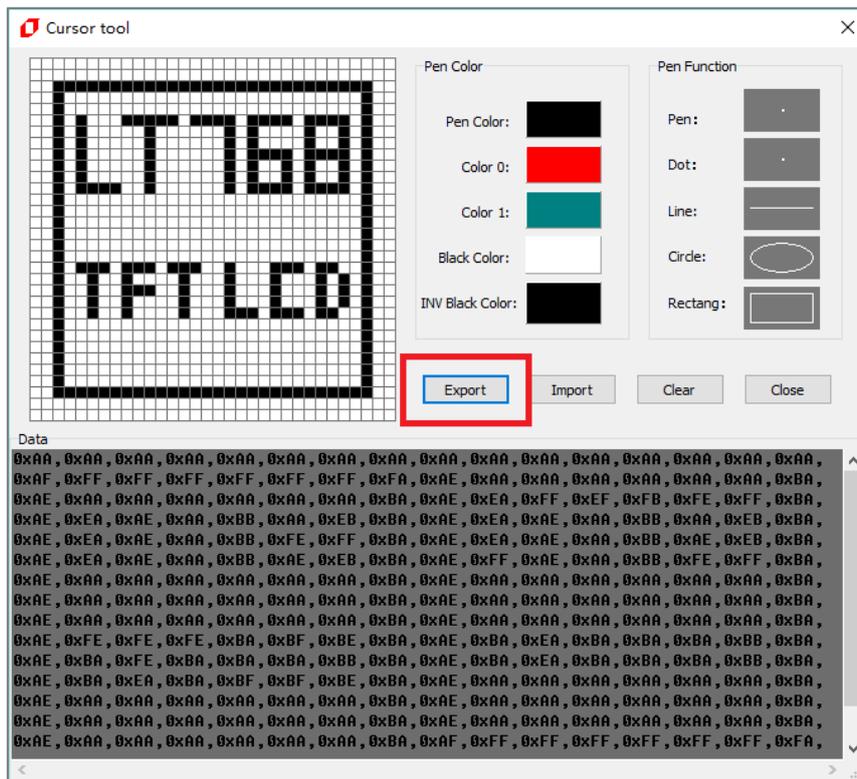


图 4-2：导出图形光标数据

c、复制图形光标数据：

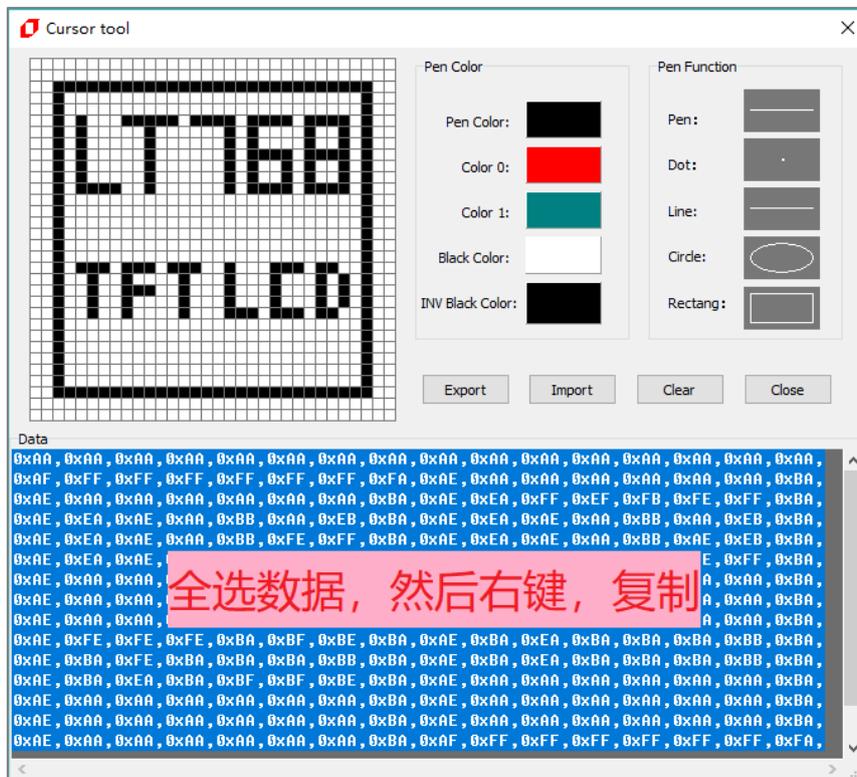


图 4-3：复制导出的光标数据



b. 粘贴到图形光标制作界面的 Data 内后，点击【Import】按钮

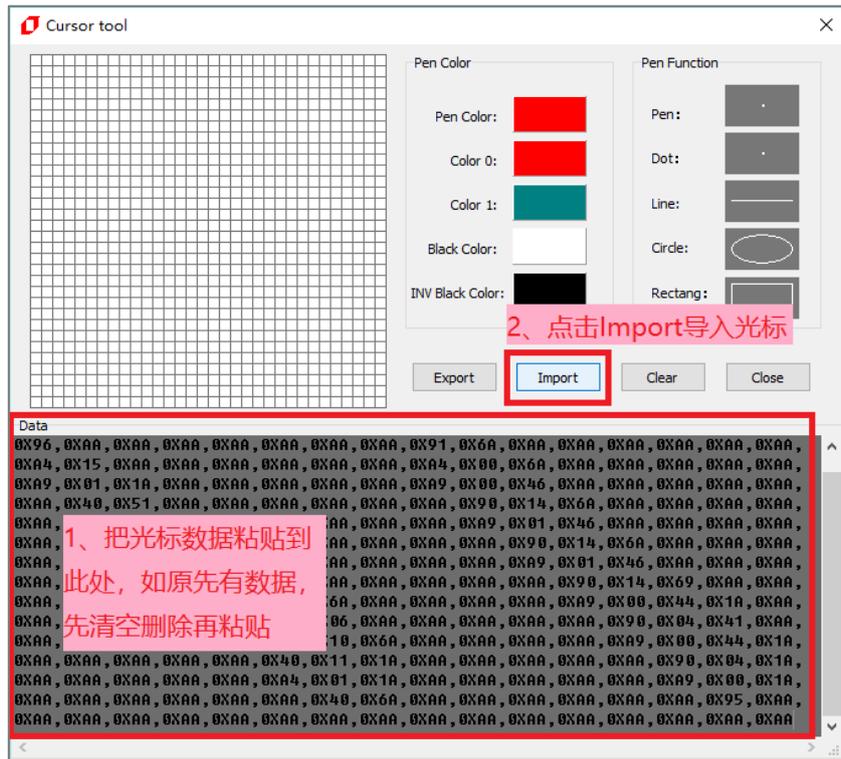


图 4-6：粘贴导入的光标数据

c. 如正确显示要导入的光标，即导入成功，可自行修改，然后按照第一步导出光标数据。

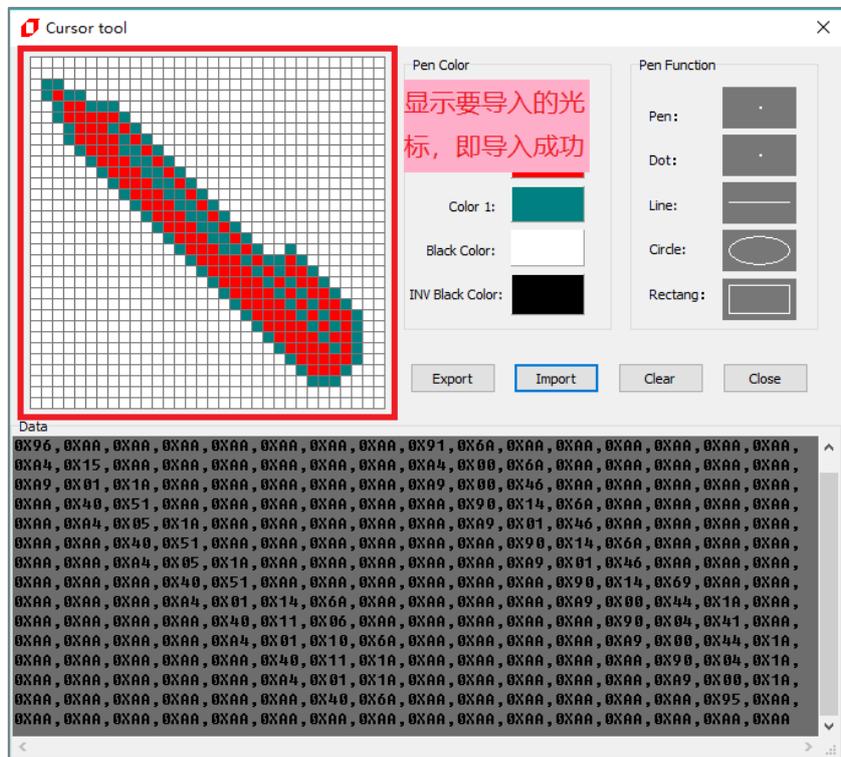


图 4-7：导入成功

## 五、设置「开机启动加载程序」

1. 点击【LT\_IMAGE\_TOOL 菜单>Bootloader】即可打开开机启动设置界面：

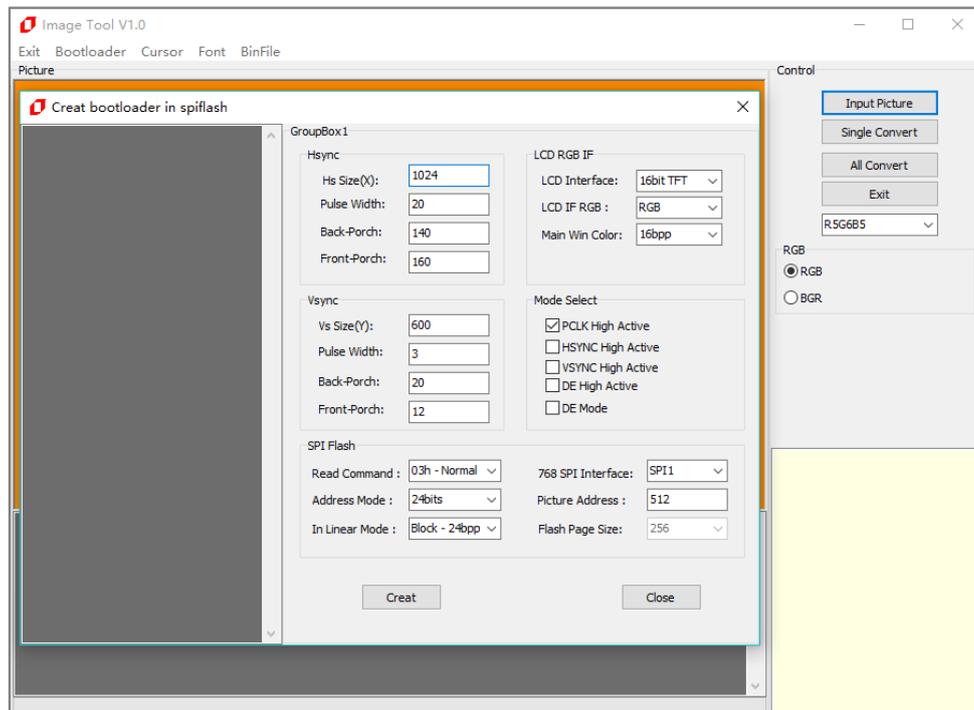


图 5-1: Bootloader 设置

2. 设置屏幕参数：Hsync、Vsync、Mode Select：根据不同 LCD 屏幕设置不同的参数，需参照屏幕资料来修改。

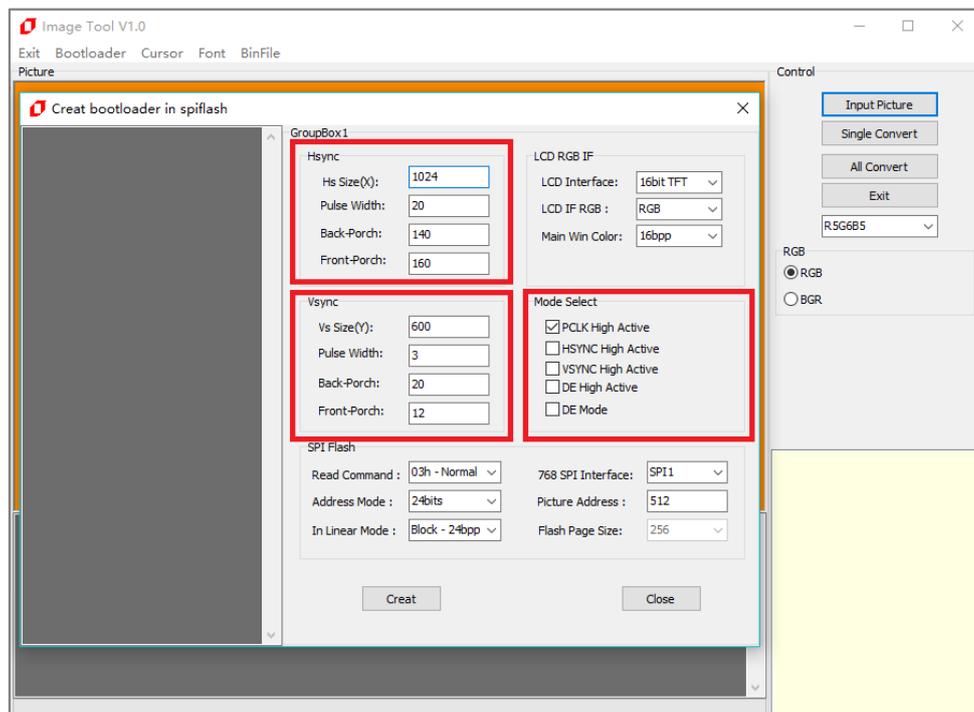


图 5-2: 设置屏幕参数

### 3. 设置 LCD 接口:

a: LCD Interface: 选择 16、18、24bits CMOS 接口面板

b: LCD IF RGB: 选择 RGB 颜色排列

c: Main Win Color: 选择显示色度 24bpp (RGB 8:8:8) 、16bpp (RGB 5:6:5) 、或者是 8bpp (RGB 3:3:2)

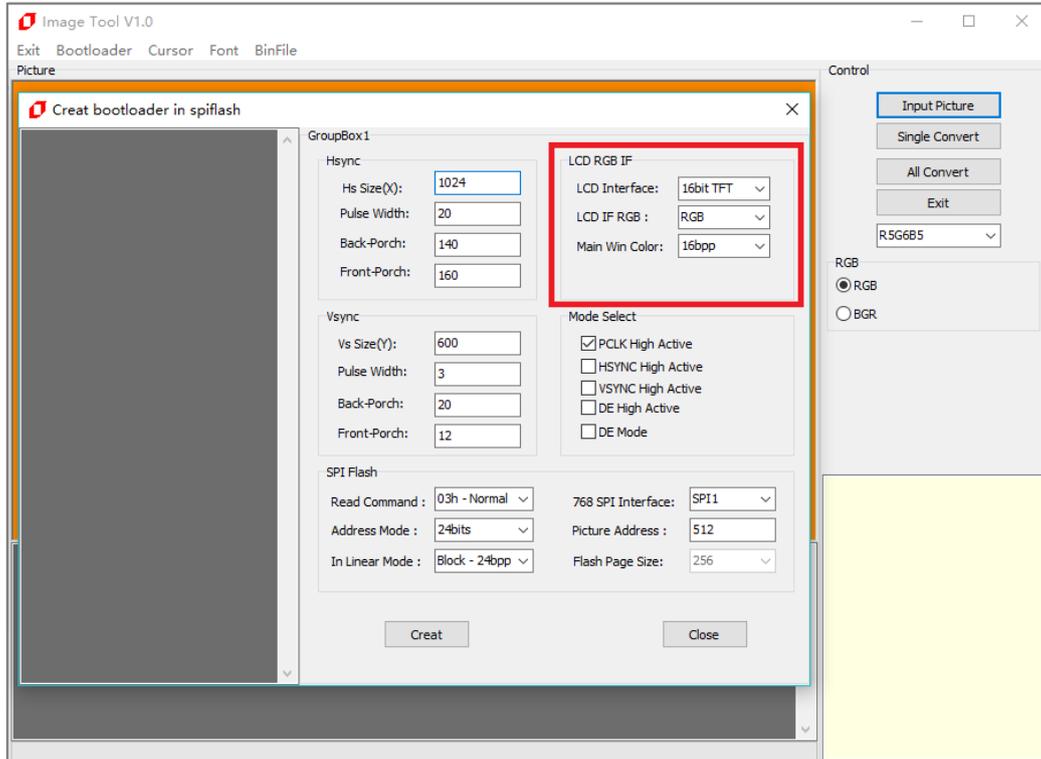


图 5-3: 设置 LCD 接口

#### 4. 设置 SPI Flash 参数:

- a. Read Command: 选择 SPI Flash 读取方式
- b. 768 SPI Interface: 选择 LT768 的 SPI 接口: SPI0 或 SPI1
- c. Address Mode: 选择寻址方式: 24bits 或 32bits
- d. Picture Address: 设定开机显示图片的 Flash 地址
- e. In Liner Mode: 选择内存显示深度

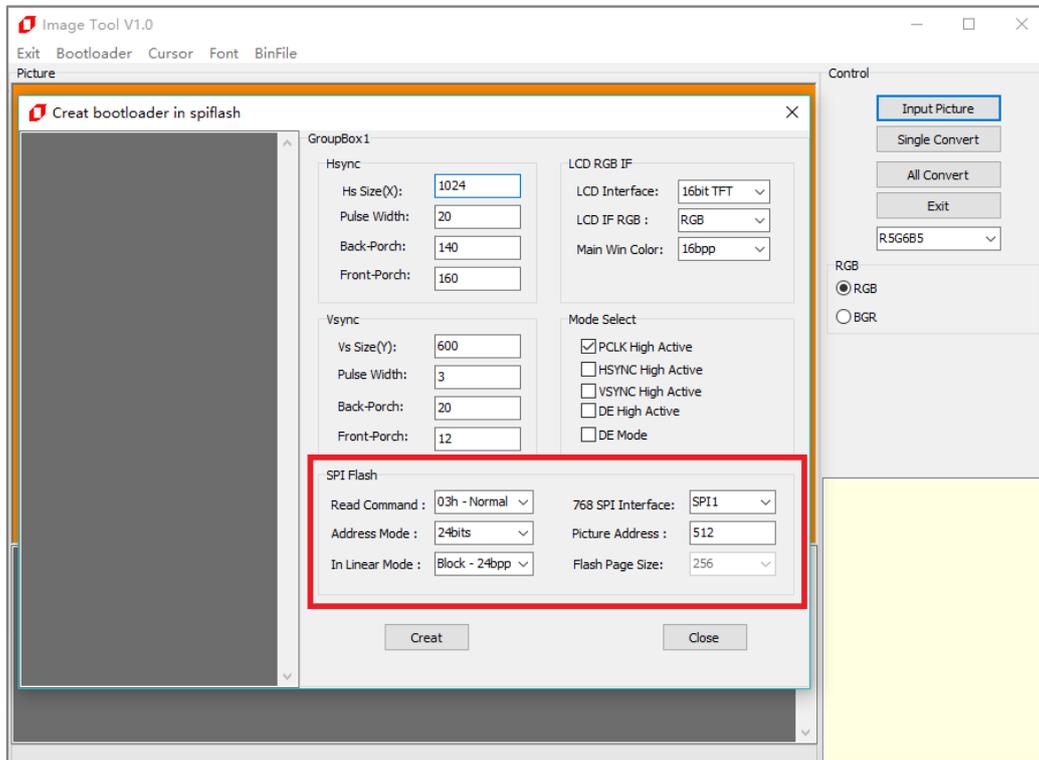


图 5-4: 设置 SPI Flash 参数

5. 以上参数都设定好后, 点击【Create】按钮另存为 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。

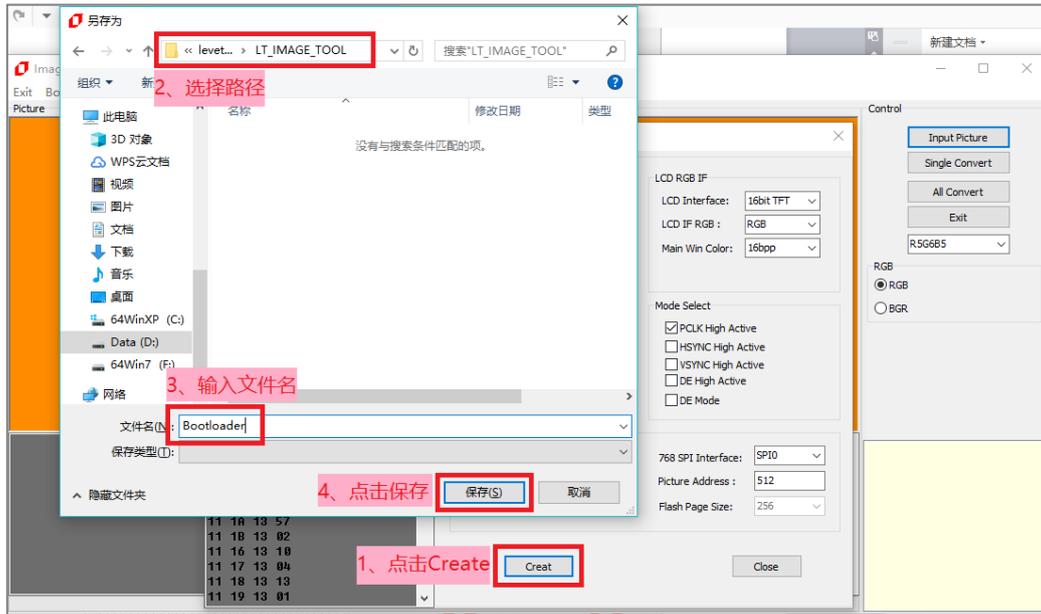


图 5-5: 保存

当显示 ok 时, 即保存成功。

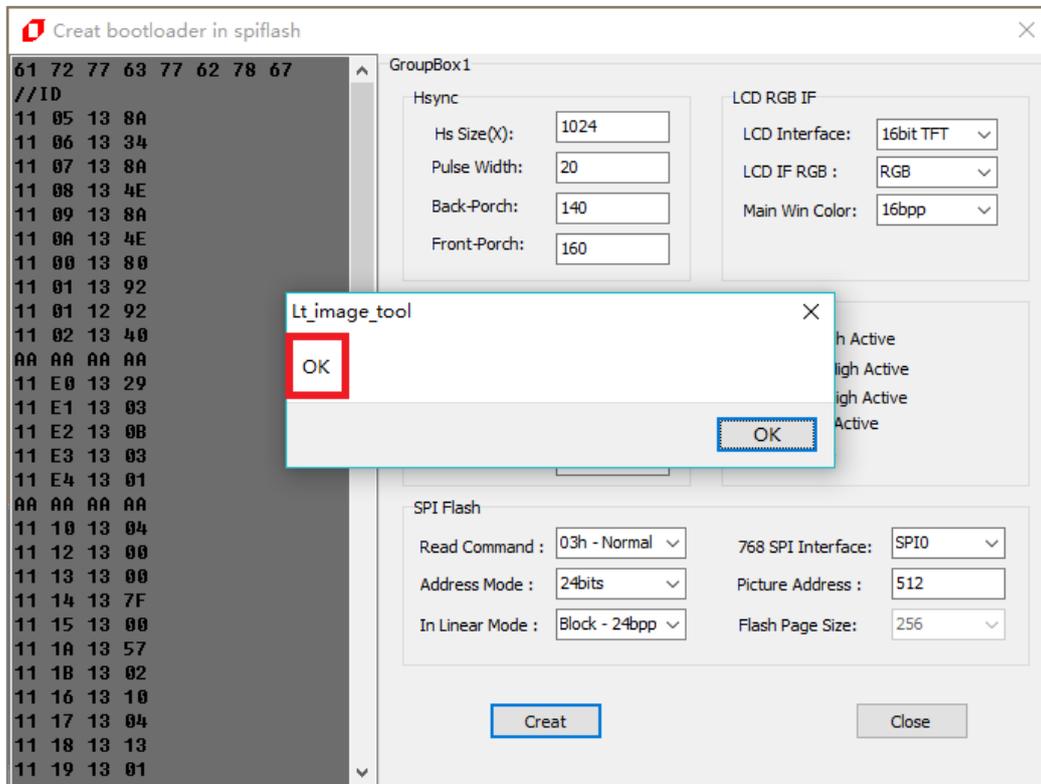


图 5-6: 保存成功

6. 保存后可以在目标文件夹中看到导出的 **Bootloader.bin** 文件:

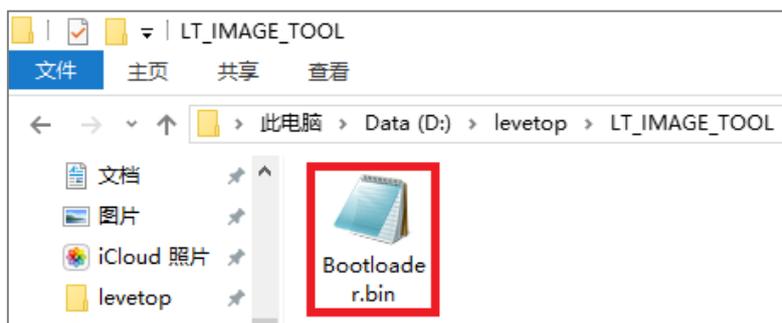


图 5-7: 导出的 **Bootloader.bin** 文件

## 六、Bin 文件整合功能

1. 点击【LT\_IMAGE\_TOOL 菜单>Font】即可打开 Bin 文件整合界面，最多能整合 6 个 Bin 文件，点击【File 1~6】可依次添加。注意，Bootloader.bin 文件需放在地址 0 位置，即 File 1。

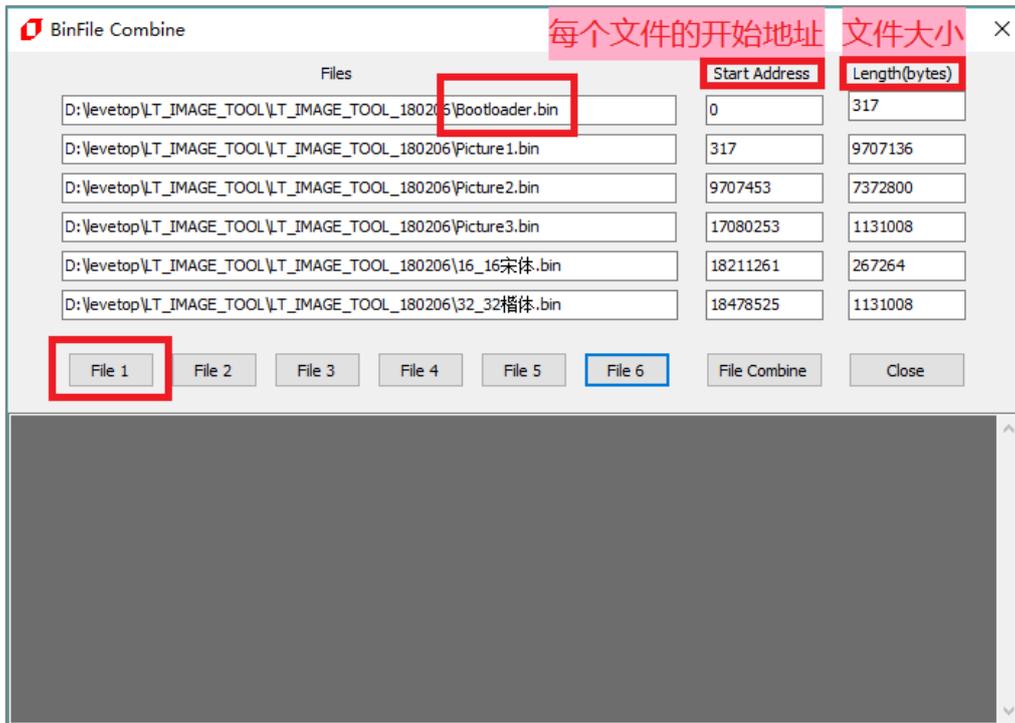


图 6-1: Bin 文件整合

2. 点击【File Combine】按钮保存整合文件，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。

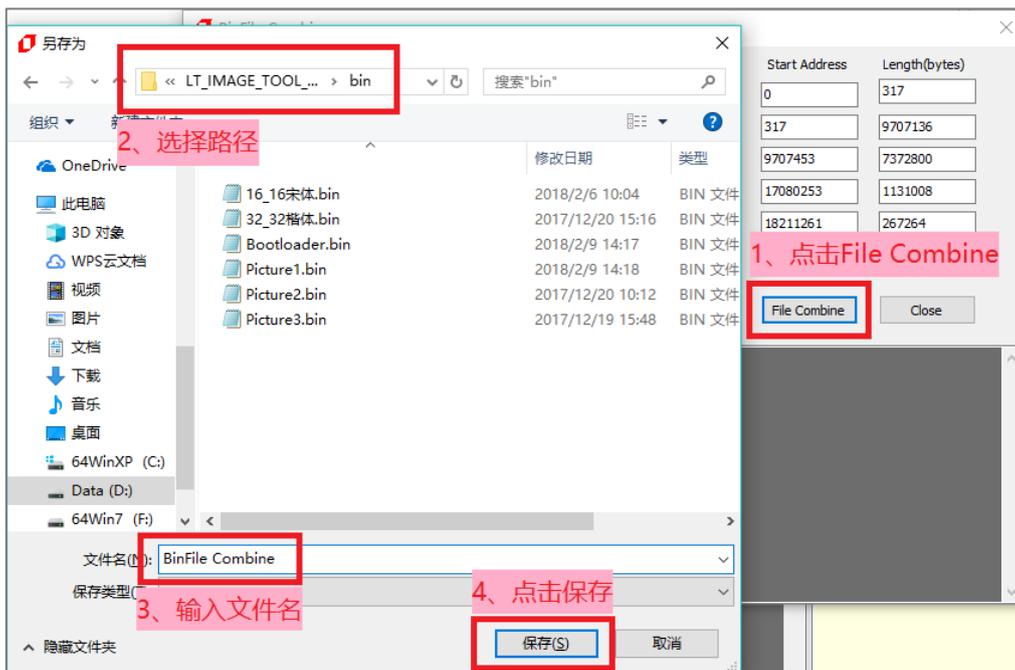


图 6-2: 保存整合文件

当显示 Combine over 时，即整合成功，并显示每个源文件的地址和大小，同时生成一个 BinFile Combine Addr.txt 文件，便于查阅每个源文件的地址、大小等详细信息。

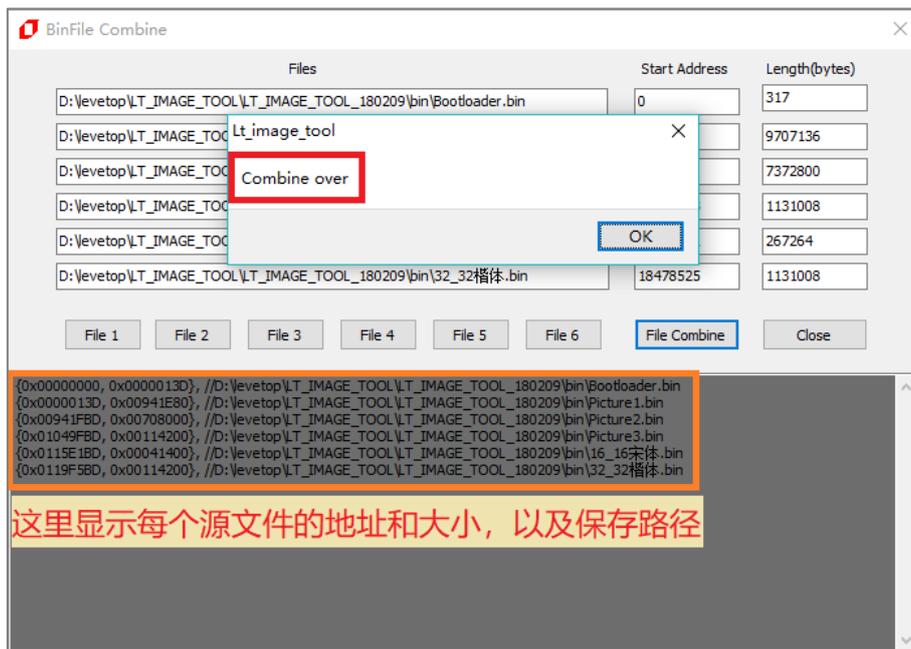


图 6-3：整合成功

### 3. 生成的文件：

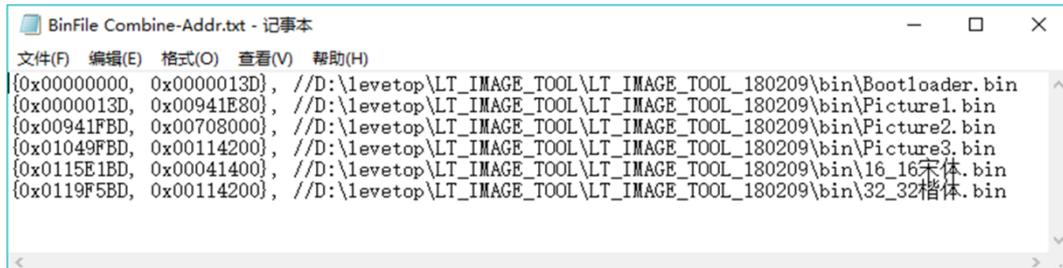


图 6-4：保存文件信息

整合完成后可以在目标文件夹中看到导出的 BinFile Combine.bin 文件，然后使用者可以用 SPI Flash 烧录器将此档案烧录到连接至 LT7868x 的 SPI Flash。

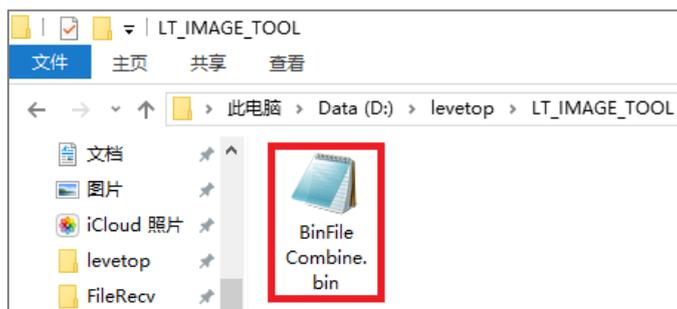


图 6-5：导出的 Bin 整合文件

#### 4. 程序调用烧录到 SPI Flash 内的 Bin 文件资料

以下用 2 个例子说明程序如何调用已经烧录到 SPI Flash 内的 Bin 文件数据，一个是在程序中显示图片【Picture1】和【Picture2】，另一个是在程序中调用 SPI Flash 内的字库（16\_16 宋体）。

**范例一：**使用 DMA 的方式从 LT768 的 CS0 中外挂的 Flash 中的【Picture1】地址读取第一张和第二张图片，并循环显示。两张图片的分辨率为 1024\*600、色深为 16bit。函数的详细参数设置，请参考 LT768 的应用手册【LT768\_AP-Note\_Vxx\_CH.pdf】档案。

```
Select_Main_Window_16bpp();           // 设置主规窗的色深 16bit 的深度
Main_Image_Start_Address(0);          // 从显示的 0 地址起开始映像到主规窗图层中
Main_Image_Width(1024);               // 主规窗的宽度
Main_Window_Start_XY(0, 0);           // 主视窗的起始坐标主规窗从(0, 0)地址开始
Canvas_Image_Start_address(0);        // 从底图（显示内存）的 0 地址开始写数据
Canvas_image_width(1024);             // 底图的宽度
Active_Window_XY(0, 0);                // 工作视窗：LCD 从主规窗的(0,0)地址开始显示
Active_Window_WH(1024, 600);          // 工作视窗：LCD 显示的宽为 1024，长为 600
while(1)
{
/*-----显示第一张图片，地址为 0x0000013D-----*/
LT768_DMA_24bit_Block(0, 0, 0, 0, 1024, 600, 0x0000013D);
delay_ms(500);
/*-----显示第二张图片，地址为 0x0000013D+1024*600*2-----*/
LT768_DMA_24bit_Block(0, 0, 0, 0, 1024, 600, 0x0000013D+1024*600*2);
delay_ms(500);
}
```

**范例二：**从 LT768 的 CS0 中外挂的 Flash 中的【16\_16 宋体】地址读取 16\*16 宋体字库数据，并显示“东莞市乐电子有限公司”，且不放大字体的高度和宽度、背景色透明、字体对齐。函数的详细参数设置，请参考【LT768\_AP-Note\_Vxx\_CH.pdf】档案。

```
Select_Main_Window_16bpp();           // 设置主规窗的色深 16bit 的深度
Main_Image_Start_Address(0);          // 从显示的 0 地址起开始映像到主规窗图层中
Main_Image_Width(1024);               // 主规窗的宽度
Main_Window_Start_XY(0, 0);           // 主视窗的起始坐标主规窗从(0, 0)地址开始
Canvas_Image_Start_address(0);        // 从底图（显示内存）的 0 地址开始写数据
Canvas_image_width(1024);             // 底图的宽度
Active_Window_XY(0, 0);                // 工作视窗：LCD 从主规窗的(0,0)地址开始显示
Active_Window_WH(1024, 600);          // 工作视窗：LCD 显示的宽为 1024，长为 600
LT768_DrawSquare_Fill(0, 0, 1024, 600, White); //画白布
/*-----外挂字库初始化（字库起始地址：0x0115E1BD，字库大小：0x00041400）-----*/
LT768_Select_Outside_Font_Init(1, 0, 0x0115E1BD, 1024*600*2, 0x00041400, 16, 1, 1, 1, 1);
/*-----显示文字-----*/
LT768_Print_Outside_Font_String(425,50,Red,White,(u8*)"东莞市乐电子有限公司");
```