



# 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

Levetop Semiconductor Co., Ltd.



诚信・专业・效率

Page -1-

## LT\_IMAGE\_TOOL 使用说明书

LT\_IMAGE\_TOOL.EXE 是 <u>乐升半导体</u> 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:

🖸 Image Tool V1.2	- 🗆 X
<u>Exit</u> <u>B</u> ootloader <u>C</u> ursor <u>F</u> ont BinFile <u>G</u> IFTool	6 I
Hicture	Control
σ 拥     1、点击Input Picture按	Input Picture
← → ▼ ↑ _ ≪ Data (D) > laveton > 480_272_4P ▼ ひ 搜索"480_272_4P" ♪	Single Convert
	All Convert
	Exit
■ 此电脑	RGB
3D 対象	● RGB
	OBGR
■ 图片 2.bmp 2.bmp	
② 文档	
↓ 下载 Fill LCD 意思如素を削減 High Performance LCB Graphics Controller	
La 本地磁盘 (C:)	
Data (D:)	
文件名(N): 1.bmp	
3、点击打开 7冊@ 11冊@	

#### 图 1-2: 导入图片



#### 图 1-3:导入完成

诚信・专业・效率

Page -3-

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。

Image Tool V1.2						
<u>Exit B</u> ootloader <u>C</u> ursor <u>F</u> ont B <u>i</u> nFile <u>G</u> IFTool	1.	Single Co	nvert:	导出当	前选择的-	一张图片
● 另存为 2、选择目标文件夹		All Conve	rt:导出	浙有图	沜	
, $\leftarrow$ $\rightarrow$ $\checkmark$ $\uparrow$ $\square$ « LT_IMAGE_TOOL $\Rightarrow$ Pictures	~ Ū	搜索"Pictures"		9	Single C	onvert
组织 ▼ 新建文件夹			-	•	All Cor	nvert
▲ 型 此电脑 ③ 3D 对象	没有与搜索条件匹配;	的项。			R5G6B5	×
📲 视频					RGB	
▶ 图片					BGR	
↓ 下载						
▲ ● 音乐						
Data (D:) 3、 制八 又1十百				^ 1	bmp	
文件名(N): Picture1				~ 2. 3. 4.	bmp png bmp	
保存类型(T):				~		
▲ 隐藏文件夹	4、点击保存	保存(S)	取消			
				~		

图 1-5: 导出图片 (1/2)



Page -4-



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

Creat Chinese Font Lib
 Font Bitmap
 Font Parameter
 Chinese
 Chinese

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:

字体       ×         字体(E):       デ形(Y):       大小(S):         等後       小四       備定         防冻       一       一         照体       一       小四       原ご         が次       工       小四       原ご         小四       小四       原ご       家体のの         「水小       工       小四       の         小四       「       小四       の         「小四       「       小四       の         「       一       小四       小四         「       一       小四       小四         小四       「       小四       い二         小四       「       小四       小回         小口       小口       小口       小口         小口       小口       小口       小口         小口       小口       小口       小口         小口       小口       小口       小口         放果       「       「       小口         前       小口       小口       小口         放果       「       「          「       小田       AaBbTy/Zz       」         一       一           「       一<	ont Bitmap		Font Parameter
	>体         字体(□):         実体         等线         分束         黑体         楷体         液软雅黑         新宋体         效果         一剰除线(L)         痰色(C):         黒色	字形(1): 常规 新規 耐約 相体 都編約林 小五 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 小二 六号 「、二 一 一 一 一 一 一 一 一 一 一 一 一 一	Font Parameter Chinese ● GB2312 ○ BIG5 ○ 繁体GB2312 Font Size: 16*16 ✓ H Position: Middle ✓ V Position: Middle ✓ V Position: Middle ✓ Color Threshold(0~254): 250 Char Code: BBAA Test Code Select Font Creat Bin Close

图 2-2:选择字体

诚信・专业・效率



Page -7-

 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。

<b>⑦</b> 另存为		×
← → ✓ ↑ 🔜 « LT_IMAGE_TOOL → Fonts	✓ <ul><li>♂</li><li>投索"Fonts"</li></ul>	Font Parameter
<sup>组织 ▼ 新</sup> 2、选择路径		Chinese () GB2312
	修改日期	○BIG5 ○繁体GB2312
③ 3D 対象	没有与搜索条件匹配的项。	Font Size: 16*16 🗸
		H Position: Middle 🗸
		V Position: Middle 🗸
◆ ト <u>転</u> → 音乐		Color Threshold(0~254):
		Char Code:
* 本地磁曲 (C:) — Data (D:) 2 協 ) 文// ク		DDAA
	1、点击	记reat Bin按钮导出
		Creat Bin
		Close
▲ 隐藏文件夹	4、点击保存 保存(5) 取消	
		U I
		· ·

## 图 2-4:保存字库

Page -8-



当显示 (字库)+Font Lib ok 时,即保存成功:

🥑 Creat Chinese Font Lib	×
Font Bitmap	Font Parameter Chinese ④ GB2312 ⑤ BIG5 ⑤ 繁体GB2312 Font Size: 16*16 〜 H Position: Middle 〜 V Position: Middle 〜 V Position: Middle 〜 Color Threshold(0~254): 250 Char Code: BBAA Test Code Select Font Creat Bin Close

图 2-5: 字库制作完成

- | 📝 📙 🖛 | Fonts 文件 主页 共享 查看 小 小 > 此电脑 > Data (D:) > levetop > LT\_IMAGE\_TOOL > Fonts ← ANNONALINE 📌 快速访问 \_\_\_\_ 桌面 \* 🦊 下载 \* \* 🔮 文档 简体16\_16宋体. ■ 图片 \* bin
- 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



诚信・专业・效率

Page -9-

## 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

诚信・专业・效率

Page -10-

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:

💋 GIF Tool	- 🗆 X
GIF Picture ④ 打开 ← → ← ↑ → 此电 <sup>10</sup> · 四 · · · · · · · · · · · · · · · · ·	rl Input Picture Single Convert All Convert Close RSG685 ~ RGB @ RGB O BGR
文件名(N): 5-121204193R0-50.gif v All(*.gif,*.GIF) v All(*.gif,*.GIF) v 3、点击打开 打开(O) 取消 。	

#### 图 3-2: 导入图片



#### 图 3-3:导入完成

Page -11-



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

🖸 GIF Tool		-		×
GIF Picture	Ctrl			
© GIF Tool		Input Pictu Single Conv All Conve Close S56685 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86688 S86685 S8665 S8665 S8665 S8665 S8665 S8655 S8655 S8655 S8655 S8655 S86555 S86555 S86555 S865555 S8655555 S865555555555	ure vert ert	× 数 <sup>133</sup>
<				

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。

🥑 GIF Tool		- 🗆 X
GIF Picture	1、Single Convert: 导出語 All Convert: 导出所有 ・ 0 曖減。	当前选择的一张GIF GIF
组织 ▼ 新建文件夹 ■ 此电脑 ③ 3D 对象 圖 视频 ■ 图片 圖 文档 ↓ 下载 〕 音乐 ■ 桌面 些 本地磁曲 (C:) ■ Data (D:) 3、输入文件名	► ◆ 2 没有与搜索条件匹配的项。	All Convert Close R5G6B5 RGB @ RGB @ BGR 5-121204193R0-50.qif 8864677414617:381654e53ccb528b333 b5a190ddbedf7c7fd37a7812fc337eb9
文件名(N load 保存类型():	∽ ✓ ↓、点击保存 保存⑤ 颐嘴	

图 3-5: 导出图片 (1/2)

Page -12-



減信・专业・效率 <sup>东莞市松</sup>



图 3-6: 导出图片 (2/2)

5. 成功导出图片 Bin 文件:

0 GIF Tool	- 🗆 X
GIF Picture	Ctrl
	Input Picture Single Convert All Convert Close R5G685 RGB © RGB
Lt_image_tool_v1 X	OBGR
Convert ok 显示Convert ok即成功 OK	5-121204193R0-50.gif 86afe7414617c38ff64e53ccb528b333 b5af90ddbedf7c7fd37a7812fc337eb9
(128, 128, 0x00138000, 0x00008000),//5-121201493R0-50, gif-Frames3 (128, 128, 0x00023800, 0x00008000),//5-121201493R0-50, gif-Frames5 (128, 128, 0x00338000, 0x0008000),//5-121201493R0-50, gif-Frames5 (128, 128, 0x0038000, 0x0008000),//5-121201493R0-50, gif-Frames7 (128, 128, 0x00048000, 0x0008000),//5-121201493R0-50, gif-Frames7 (128, 128, 0x00048000, 0x0008000),//5-121201493R0-50, gif-Frames9 (128, 128, 0x00048000, 0x0008000),//5-121201493R0-50, gif-Frames9 (128, 128, 0x00048000, 0x0008000),//5-121201493R0-50, gif-Frames10 (128, 128, 0x00048000, 0x00088000),//5-121201493R0-50, gif-Frames11	

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:

🥘 loa	d-Add	r.txt - 记事本							_		>	<
文件(E)	编辑	(E) 格式( <u>O</u> ) 登	1]]≣	帮助( <u>H</u> )								
//0 {128,}{}}}}}}}}}}}}}}}}}}}}}}}}}}	128, 128, 128, 128, 128, 128, 128, 128,	0x000000000, 0x00010000, 0x00010000, 0x00018000, 0x00020000, 0x00028000, 0x00038000, 0x00038000, 0x00040000, 0x00048000, 0x00058000, 0x00058000,	0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x0	008000} 008000} 008000} 008000} 008000} 008000} 008000} 008000} 008000} 008000} 008000}	, //5-1 , //5-1 , //5-1 , //5-1 , //5-1 , //5-1 , //5-1 , //5-1 , //5-1	121204 121204 121204 121204 121204 121204 121204 121204 121204 121204 121204	4193R0 4193R0 4193R0 4193R0 4193R0 4193R0 4193R0 4193R0 4193R0 4193R0 4193R0	-50. -50. -50. -50. -50. -50. -50. -50.	gif gif gif gif gif gif gif gif gif gif	-Frames -Frames -Frames -Frames -Frames -Frames -Frames -Frames -Frames -Frames	0 1 2 3 4 5 6 7 8 9 10 11	< >
<											>	.::

图 3-9: 导出的每张图片信息



诚信・专业・效率

Page -14-

## 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



诚信・专业・效率

Page -15-



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。





c、复制图形光标数据:

#### 图 4-3:复制导出的光标数据





#### d、粘贴复制的光标数据到程序内:

<u>File Edit View Project Flash Debug Peripherals Tools SVCS Window H</u> elp
□ 🐸 🛃 🐉 🛝 🖦 📽 💌 🗢 🔶 🗮 🐡 🥵 🕸 🐘 🕼 🕼
🐼 🎬 🐲 🔐 🛛 🗱 LIT768_DEMO 🛛 🖂 🕺 📥 🖷 🗇 🕎 🍪
Image.h
<pre>1 = #ifndef _Image_h 2 #define _Image_h 3  4 = const unsigned char L1768_LCD[256] = { 5</pre>

图 4-4: 粘贴导出的光标数据

#### 2、导入及修改图形光标:

a. 复制要导入的光标数据

🥮 Pen.txt - 记事本	复制光标数据	_	$\Box$ ×	
文件(E) 编辑(E) 格式(Q) 查看(V) 帮助(H)				
OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA,	OXAA, OXAA, OXAA, OXAA, OXAA, O	XAA, OXAA	, OXAA, 🛛 🗸	$^{\sim}$
OX96, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA,	0X91, 0X6A, 0XAA, 0XAA, 0XAA, 0	DXAA, OXAA	, OXAA,	
OXA4, OX15, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA,	0XA4, 0X00, 0X6A, 0XAA, 0XAA, 0	DXAA, OXAA	, OXAA,	
OXA9, OXO1, OX1A, OXAA, OXAA, OXAA, OXAA, OXAA,	0XA9, 0X00, 0X46, 0XAA, 0XAA, 0	DXAA, OXAA	, OXAA,	
OXAA, OX40, OX51, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA,	OXAA, OX90, OX14, OX6A, OXAA, O	DXAA, OXAA	, OXAA,	
OXAA, OXA4, OXO5, OX1A, OXAA, OXAA, OXAA, OXAA,	OXAA, OXA9, OX01, OX46, OXAA, (	DXAA, OXAA	, OXAA,	
OXAA, OXAA, OX40, OX51, OXAA, OXAA, OXAA, OXAA,	OXAA, OXAA, OX90, OX14, OX6A, (	DXAA, OXAA	, OXAA,	
OXAA, OXAA, OXA4, OXO5, OX1A, OXAA, OXAA, OXAA,	OXAA, OXAA, OXA9, OXO1, OX46, (	DXAA, OXAA	, OXAA,	
OXAA, OXAA, OXAA, OX40, OX51, OXAA, OXAA, OXAA,	OXAA, OXAA, OXAA, OX90, OX14, (	DX69, OXAA	, OXAA, 📘	
OXAA, OXAA, OXAA, OXA4, OXO1, OX14, OX6A, OXAA,	OXAA, OXAA, OXAA, OXA9, OXO0, (	DX44, 0X1A	, OXAA,	
OXAA, OXAA, OXAA, OXAA, OX40, OX11, OX06, OXAA,	OXAA, OXAA, OXAA, OXAA, OX90, (	X04, 0X41	, OXAA,	
OXAA, OXAA, OXAA, OXAA, OXA4, OXO1, OX10, OX6A,	OXAA, OXAA, OXAA, OXAA, OXA9, (	)X00, 0X44	, OX1A,	
OXAA, OXAA, OXAA, OXAA, OXAA, OX40, OX11, OX1A,	OXAA, OXAA, OXAA, OXAA, OXAA, O	X90, 0X04	, OX1A,	
OXAA, OXAA, OXAA, OXAA, OXAA, OXA4, OXO1, OX1A,	OXAA, OXAA, OXAA, OXAA, OXAA, O	DXA9, 0X00	, OX1A,	
OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OX4O, OX6A,	OXAA, OXAA, OXAA, OXAA, OXAA, O	DXAA, OX95	, OXAA,	
OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA, OXAA,	OXAA, OXAA, OXAA, OXAA, OXAA, O	)XAA, OXAA,	, OXAA	ν.
<			> .	

图 4-5:复制导入的光标数据



诚信・专业・效率

Page -17-

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

诚信・专业・效率



Page -18-

## 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:

Image Tool V1.0			– 🗆 X
Exit Bootloader Cursor Font BinFile			
Picture			Control
🖸 Creat bootloader in spiflash	Grundbaut	×	Input Picture
^ ^	GroupBox 1		Single Convert
	1024		All Convert
	His Size(X):	LCD Interface: Ibbit IFI	Exit
	Puise width: 20	LCD IF RGB : RGB ~	R5G685 ×
	Back-Porch: 140	Main Win Color: 16bpp ~	RGB
	Front-Porch: 160		● RGB
	Vsync	Mode Select	⊖BGR
	Vs Size(Y): 600	PCLK High Active	
	Pulse Width: 3	HSYNC High Active	
	Back-Porch: 20	VSYNC High Active	
	Eront-Porch: 10		
	SPI Flash		
	Read Command : 03h - Normal 🗸	768 SPI Interface: SPI 1 V	
	Address Mode : 24bits 🗸	Picture Address : 512	
Ē	In Linear Mode : Block - 24bpp V	Flash Page Size: 256 V	
	Creat	Close	
	_		
		<u> </u>	-

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。

🕖 Image Tool V1.0			- 🗆 X
Exit Bootloader Cursor Font BinFile Picture			Control
🚺 Creat bootloader in spiflash		×	Input Picture
^	GroupBox1		Single Convert
	Hsync 1024	LCD RGB IF	All Convert
	Hs Size(X): 202 1	LCD Interface: 16bit TFT V	Exit
	Poise Width: 20	LCD IF RGB : RGB ~	R5G6B5 ~
	Back-Porch: 140	Main Win Color: 16bpp ~	RGB
	Hont+ordi: 160		● RGB
	Vsync	Mode Select	OBGR
	Vs Size(Y): 600	PCLK High Active	
	Pulse Width: 3	HSYNC High Active	
	Back-Porch: 20	DE High Active	
	Front-Porch: 12	DE Mode	
	SPI Flash		
	Read Command : 03h - Normal 🗸	768 SPI Interface: SPI1 V	
	Address Mode : 24bits 🗸	Picture Address : 512	
	In Linear Mode : Block - 24bpp 🗸	Flash Page Size: 256 🗸	
	Creat	Close	
		v	

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)

🚺 Image Tool V1.0			- 🗆 X
Exit Bootloader Cursor Font BinFile			Control
Picture			Control
🗗 Creat bootloader in spiflash		×	Input Picture
^	GroupBox1		Single Convert
	Hsync	LCD RGB IF	All Convert
	Hs Size(X): 1024	LCD Interface: 16bit TFT $\checkmark$	Evit
	Pulse Width: 20	LCD IF RGB : RGB 🗸	Exit
	Back-Porch: 140	Main Win Color: 16bpp 🗸	R5G6B5 ~
	Front-Porch: 160		RGB
	Vsync	Mode Select	OBGR
	Vs Size(Y): 600	PCLK High Active	
	Pulse Width: 3	HSYNC High Active	
	Back-Porch: 20	VSYNC High Active	
	Front-Porch: 12		
	SPI Flash		
	Read Command : 03h - Normal 🗸	768 SPI Interface: SPI 1	
	Address Mode : 24bits ~	Picture Address : 512	
	In Linear Mode : Block - 24bpp 🗸	Flash Page Size: 256 🗸	
	Creat	Close	
U V			

图 5-3:设置 LCD 接口

Page -20-



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

Image Tool V1.0			- 🗆 X
Exit Bootloader Cursor Font BinFile Picture  Creat bootloader in spiflash	GroupBox1 Hsync Hs Size(X): 1024 Pulse Width: 20	LCD RGB IF LCD Interface: 16bit TFT v LCD IF RGB : RGB v	Control Input Picture Single Convert All Convert Exit
	Back-Porch:         140           Front-Porch:         160           Vsync            Vs Size(Y):         600           Pulse Width:         3           Back-Porch:         20           Front-Porch:         12	Main Win Color: 15bpp  Mode Select PCLK High Active HSYNC High Active USYNC High Active DE High Active DE High Active DE High Active	RGB © RGB O BGR
	SPI Flash Read Command : 03h - Normal v Address Mode : 24bits v In Linear Mode : Block - 24bpp v Creat	768 SPI Interface: SPI1 V Picture Address : 512 Flash Page Size: 256 V	
		Ų	-

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。

(°I 🔻	<b>7</b> 另存为	×		
	← → × ↑ 📙 « levet → LT_IMAGE	_TOOL ✓ Ö 搜索"LT_IMAGE_TOOL" タ		◎ 新建文档 •
🚺 Imag Exit Bo	<sup>组织▼ 新</sup> 2、选择路径	[≡ ▼ (?)		- 🗆 X
Picture	■ 此电脑 【 名称	^ 修改日期 类型		Control
	<ul> <li>3D 対象</li> <li>▲ WPS支持</li> <li>■ 視環</li> <li>■ 周片</li> <li>※ 西</li> <li>※ 式档</li> </ul>	没有与搜索条件匹配的项。	LCD RGB JF LCD Interface: 15bit TFT v LCD JF RGB : RGB v Main Win Color: 15bpp v	Input Picture Single Convert All Convert Exit RSG6B5 ~
	<ul> <li>▶ 音乐</li> <li>■ 貞面</li> <li>▲ 64WinXP (C:)</li> <li>■ Data (D:)</li> <li>■ 64Win7 (F:)</li> <li>■ 3 協 入 文 任 交</li> </ul>		Mode Select	RGB © RGB O BGR
	→ M路 文件名①: Bootloader	3	DE Mode	
	保存类型(]):	~	768 SPI Interface: SPI0 🗸	
	▲ 隐藏文件夹	4、点击保存 保存(s) 取消	Picture Address : 512 Flash Page Size: 256 V	
	11 1A 13 57 11 1B 13 02 11 16 13 10 11 16 13 10 11 17 13 04 11 18 13 13 11 19 13 01	1、点击Create	Close	-

图 5-5:保存

当显示 ok 时,即保存成功。

Creat bootloader in spiflas	sh	×
61 72 77 63 77 62 78 6 //ID 11 05 13 8A 11 06 13 34 11 07 13 8A 11 08 13 4E 11 09 13 8A 11 0A 13 4E 11 0A 13 4E 11 00 13 80	GroupBox1 Hsync Hs Size(X): 1024 Pulse Width: 20 Back-Porch: 140 Front-Porch: 160	LCD RGB IF LCD Interface: 16bit TFT ~ LCD IF RGB : RGB ~ Main Win Color: 16bpp ~
11 01 13 92 11 01 12 92 11 02 13 40 AA AA AA AA 11 E0 13 29 11 E1 13 03 11 E2 13 0B 11 E3 13 03 11 E4 13 01	Lt_image_tool	Active ligh Active ligh Active Active
AA       AA       AA         11       10       13       04         11       12       13       00         11       13       13       00         11       13       13       00         11       14       13       7F         11       15       13       00         11       14       13       57         11       18       13       02         11       16       13       10         11       16       13       10	SPI Flash Read Command : 03h - Normal V Address Mode : 24bits V In Linear Mode : Block - 24bpp V	768 SPI Interface: SPI0 V Picture Address : 512 Flash Page Size: 256 V
11 18 13 13 11 19 13 01		

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。

	的开始地址	文件大小	×
Files	Start Address	Length(bytes)	
D:\/evetop/LT_IMAGE_TOOL/LT_IMAGE_TOOL_1802(6\Bootloader.bin	0	317	
D: Vevetop V_T_IMAGE_TOOL V_T_IMAGE_TOOL_180206 VPicture 1. bin	317	9707136	
D:\vevetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180206\Picture2.bin	9707453	7372800	
D:\vevetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180206\Picture3.bin	17080253	1131008	
D:\vevetop\v_T_IMAGE_TOOL\v_T_IMAGE_TOOL_180206\16_16宋体.bin	18211261	267264	
D:\Jevetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180206\32_32楷体.bin	18478525	1131008	
File 1     File 2     File 3     File 4     File 5     File 6	File Combine	Close	
			^
			$\sim$

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。

	at all a state						×
月存为				×			~
← → • • <mark>.</mark> « L	T IMAGE TOOL → bin	∨ ひ 搜索	"bin"	م	Start Address	Length(bytes)	
					0	317	
组织▼ 7 法北	圣胶亿		== -		317	9707136	
ConeDrive			修改日期	类型	9707453	7372800	
此由脑	🥘 16_16宋体.bin		2018/2/6 10:04	BIN 文件	17080253	1131008	
2D 34~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	🥘 32_32楷体.bin		2017/12/20 15:16	BIN 文件	18211261	267264	
	Bootloader.bin		2018/2/9 14:17	BIN 文件	1 点击Fil	e Combine	
	Picture1.bin		2018/2/9 14:18	BIN 文件	т <u>с</u> жанн		1
	Picture2.bin		2017/12/20 10:12	BIN 文件	File Combine	Close	
	Picture3.bin		2017/12/19 15:48	BIN 文件	The combine	Close	
🔮 文档							^
- 🕂 下载							
♪ 音乐							
三 桌面							
🏪 64WinXP (C:)							
🔜 Data (D:)							
🕳 64Win7 (F:) 🗸	<			>			
文件名(N); BinF	ile Combine			~			
保友米刑/		1 -	与土伊方	~			
3、箱	入文件名						· ·
▲ 陶蔚文仕中			保存(S) 取	消			
POINT							

图 6-2:保存整合文件

诚信・专业・效率

Page -24-

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。

🖉 BinFile Combine		×
Files Start Address	Length(bytes)	
D: \evetop \LT_IMAGE_TOOL \LT_IMAGE_TOOL_180209 \bin \Bootloader.bin 0	317	
D:\vevetop\LT_IMAGE_TOC Lt_image_tool X	9707136	
D:\vevetop\LT_IMAGE_TOC Combine over	7372800	
D:\vevetop\LT_IMAGE_TOC	1131008	
D:\/evetop\/LT_IMAGE_TOC	267264	
D:\/evetop\/LT_IMAGE_TOOL\/LT_IMAGE_TOOL_180209\/bin\32_32楷体.bin 18478525	1131008	
File 1         File 2         File 3         File 4         File 5         File 6         File Combine	Close	
{0x0000000, 0x000013D}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Bootloader.bin {0x0000013D, 0x00941E80}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Picture1.bin {0x00941FBD, 0x001708000}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Picture2.bin {0x01049FBD, 0x00114200}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Picture3.bin {0x0115E1BD, 0x00141400}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\16_16宗体.bin {0x0119F5BD, 0x00114200}, //D:\evetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\15_3232楷体.bin		^
这里显示每个源文件的地址和大小,以及保存路径		
		$\sim$

图 6-3: 整合成功

3. 生成的文件:

	_		$\times$
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)			
<pre>{0x00000000, 0x0000013D}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Boo {0x0000013D, 0x00941E80}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Pic {0x00941FBD, 0x00708000}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Pic {0x01049FBD, 0x00114200}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\Pic {0x0119F5BD, 0x00114200}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\16 {0x0119F5BD, 0x00114200}, //D:\levetop\LT_IMAGE_TOOL\LT_IMAGE_TOOL_180209\bin\32_</pre>	tloade ture1. ture2. ture3. 16宋体 32楷体	er.bin bin bin bin bin bin bin bin	^
			$\sim$
<			>

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

诚信・专业・效率



Page -25-

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 // 从显示的 0 地址起开始映像到主规窗图层中 Main\_Image\_Start\_Address(0); 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 // 底图的宽度 Canvas\_image\_width(1024); // 工作视窗: LCD 从主规窗的(0,0)地址开始显示 Active Window  $XY(0, 0_i)$ ; 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, W	/hite); //画白布
/*外挂字库初始化 (字库起始地址: 0xC	)115E1BD, 字库大小: 0x00041400)*/
LT768_Select_Outside_Font_Init(1, 0, 0x01	15E1BD, 1024*600*2, 0x00041400, 16, 1, 1, 1, 1);
/*显示文字*/	
17769 Drint Outside Font String(125 E01	DodWhite(u0*)"左苦士に由フ方阳八ヨ")。

LT768\_Print\_Outside\_Font\_String(425,50,Red,White,(u8\*)"东莞市乐电子有限公司");



Page -26-