



RAiO TFT-LCD Controller Tool **(RA8889)** 操作说明

Version 1.1

August 11, 2020

RAiO Technology Inc.

@Copyright RAiO Technology Inc. 2019

Revise History		
Version	Date	Description
1.0	April 26, 2020	Initial Release
1.1	August 11, 2020	1. Add Chapter 2.2 Get the RAiO tool key and enable the software 2. Add Chapter 2.5 LCD Configuration 3. Add Chapter 9.7 Alpha Blending in 32-bit (ARGB) Pixel Mode (Color Depth:24bpp) 4. Modify Figure 1-1, 2-29, 9-1, 9-5, 9-9, 9-12, 9-15, 11-1 and 12-1

Contents

1. 使用前须知.....	5
1.1 在本指南中使用的约定.....	5
1.2 RAiO TFT-LCD Controller Tool(AP Tool)简介.....	5
1.3 系统需求.....	6
2. 软件简介与驱动程序安装.....	7
2.1 安装 USB 驱动程序.....	7
2.1.1 Windows XP 驱动程序安装.....	7
2.1.2 Windows 7/8/10 驱动程序安装.....	11
2.2 取得瑞佑软件密钥(RAiO tool key)与启用软件.....	17
2.3 软件面板介绍.....	19
2.4 软件功能按键介绍.....	20
2.5 LCD 配置.....	21
3. SPI Flash 烧录功能.....	23
4. MPU 写图功能.....	26
5. 字型功能.....	29
5.1 内建字库功能.....	29
5.2 外接集通字库功能.....	31
6. DMA 功能.....	33
7. 几何图形绘图功能.....	38
8. 内存检视功能.....	42
8.1 Ping-pong Buffer.....	42
8.2 卷动功能.....	45
9. BTE Function.....	49
9.1 Memory Copy with ROP.....	49
9.2 Memory Copy with Chroma Keying.....	52
9.3 Memory Copy with Color Expansion.....	55
9.4 Memory Copy with Color Expansion and Chroma Keying.....	58
9.5 Solid Fill.....	59
9.6 Alpha Blending in Picture Mode.....	61
9.7 Alpha Blending in 32-bit (ARGB) Pixel Mode (Color Depth:24bpp).....	64

10. PIP Function.....	67
11. JPEG Function.....	69
12. AVI Function.....	71

RAiO Technology Inc.

1. 使用前须知

1.1 在本指南中使用的约定

符号

符号	说明
	RAiO 联络信息
	步序号码
	注意事项编号

Text

Abbreviations	Description
AP	Application Program
PC	Personal Computer
API	Application Programming Interface
DMA	Direct Memory Access
BTE	Block Transfer Engine
CE	Color Expansion
PIP	Picture in Picture
AP Tool	RAiO TFT-LCD Controller Tool

1.2 RAiO TFT-LCD Controller Tool (AP Tool)简介

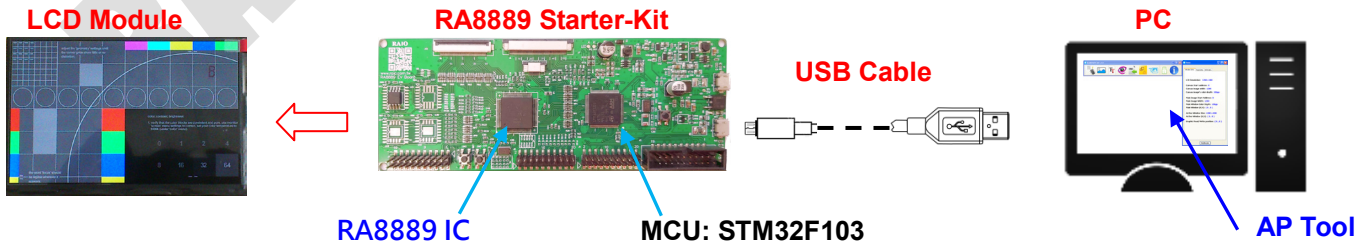


图 1-1 Starter-Kit connection with PC and LCD Panel

RAiO TFT-LCD Controller Tool (AP Tool)是一套在 Windows 操作系统的应用软件，搭配 RA8889 IC 的硬件环境(Starter-Kit)，经由软件简单的操作接口，去控制 IC 执行图形、文字、DMA、BTE、PIP、JPEG、AVI 播放等相关功能。

除了 AP Tool 外，瑞佑科技提供入门开发工具包(Starter-Kit)，MCU 为 STMicroelectronics STM32F103，并使用 KEIL-C 撰写其固件，提供常用的 API 固件，用户开发自己的系统时，方便程序移植到自己的系统中，减少开发时间。

使用此 AP Tool 的目的:

- 快速了解 IC 各个功能，客户可以针对自己的系统，选择适合的功能来使用，增加系统效能，并减少 MCU 的负担
- 帮助了解各个功能如何操作，加速开发时间
- 能实时将操作结果显示于 LCD 上，避免频繁修改参数后，需花费大量时间更新 MCU 固件
- 对于 LCD 模块厂，也可以用此系统来推广自家的 LCD 模块

此 AP Tool 主要是帮助不熟悉 RA8889 IC 如何操作的客户，能快速了解相关功能的参数如何设定与操作，仅提供较常用到的功能，且由于 RA8889 功能强大，为了让使用者易于了解，特将环境设定在简单与易了解的模式下，若想进一步使用到其它的功能，

或对于 AP 有什么建议，可以与 RAiO 联络 ，针对使用者的建议，此 AP 也会不断的修正与更新。

**RAiO**

☎ TEL: +886-3-5637888 EXT: 178 / 188

☎ FAX: +886-3-5601166

☎ Sales E-mail: sales@raio.com.tw

☎ FAE E-mail: fae@raio.com.tw

☎ <http://www.raio.com.tw>

☎ Office Address:

6F-5, No.8, Tai Yuen 1st Street, Chupei City, Hsinchu, Taiwan

1.3 系统需求

- 需要搭配 RA8889 Starter-Kit 硬件环境
- 此软件适用于 Windows XP / 7 / 8 / 10 平台上
- 软件操作前需先安装 STM32F103 MCU 的 USB Driver.
- 使用此软件前，必须取得 RAiO 软件密钥

2. 软件简介与驱动程序安装

2.1 安装 USB 驱动程序

2.1.1 Windows XP Driver 驱动程序安装

- 第一次将 RA6807M Starter-Kit 通过 USB Cable 连接至 PC(Personal computer)时，Windows 会出现 “发现新硬件 STMicronics Virtual COM Port” 的提示信息。

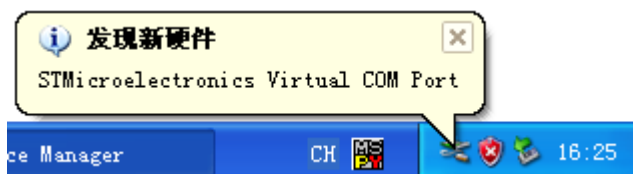


图 2-1

- Windows 会要求安装新硬件的驱动程序，点击 “下一步”。

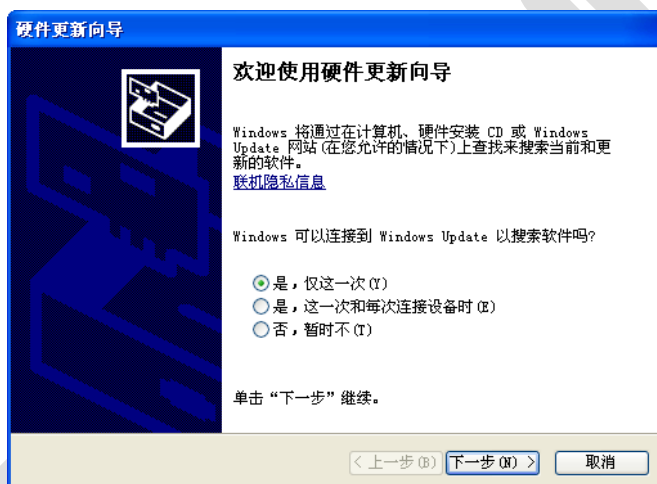


图 2-2

- 选择 “从列表或指定位置安装(高级)”，点击 “下一步”。

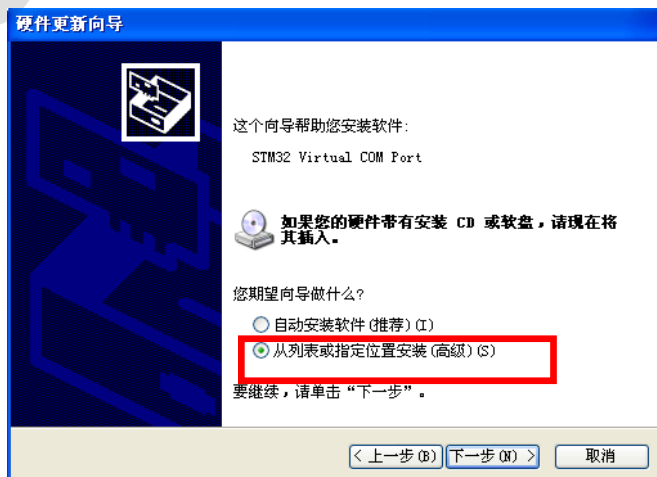


图 2-3

- 4 勾选“不要搜索，我要自己选择要安装的驱动程序”，并点击“下一步”。

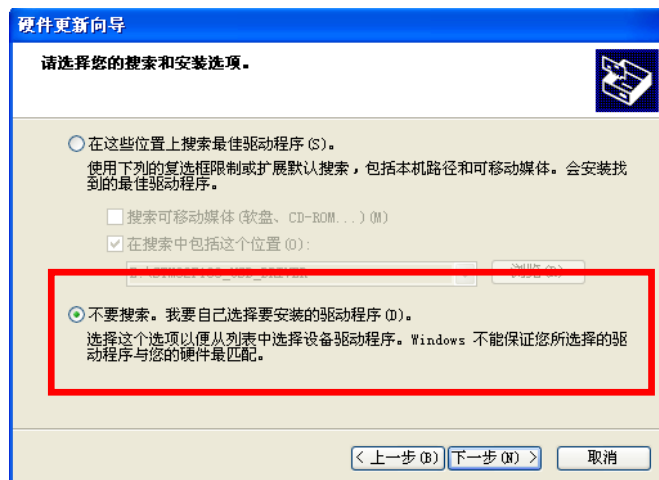


图 2-4

- 5 点击“从磁盘安装”

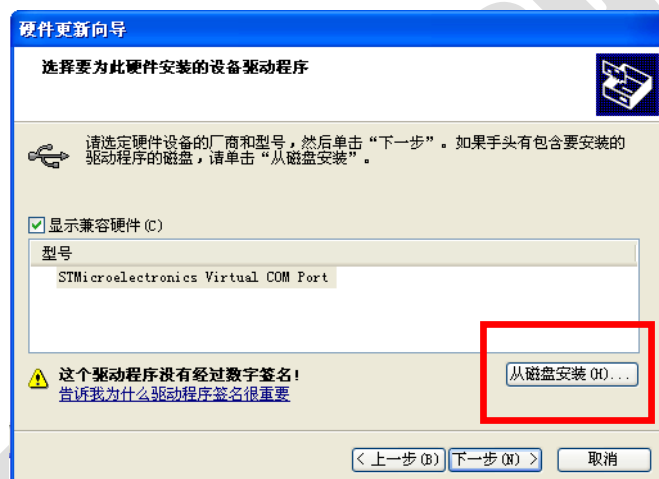


图 2-5

- 6 点击“浏览”，并选择驱动程序路径

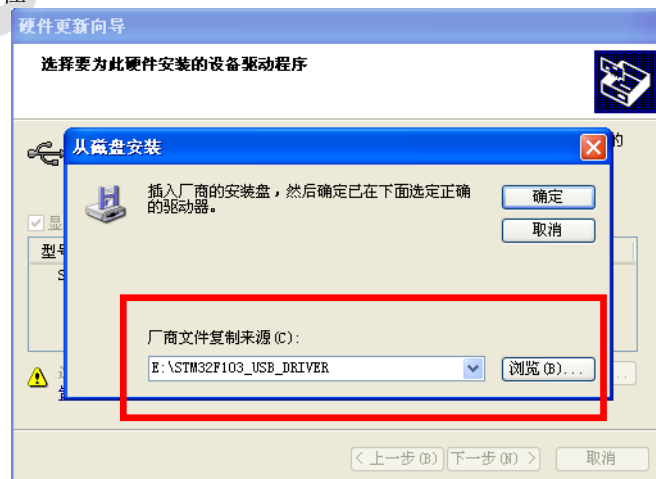


图 2-6

7 点击“下一步”

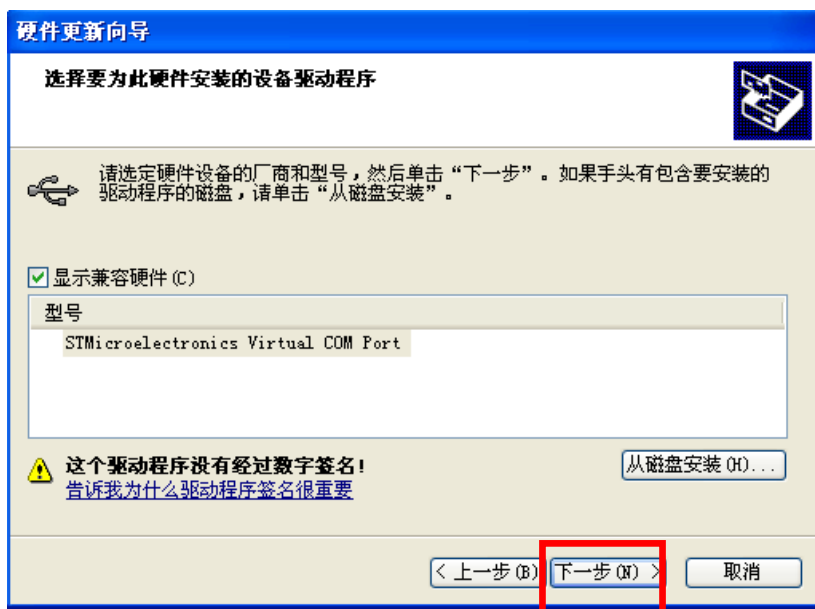


图 2-7

8 开始安装驱动程序，请稍候。

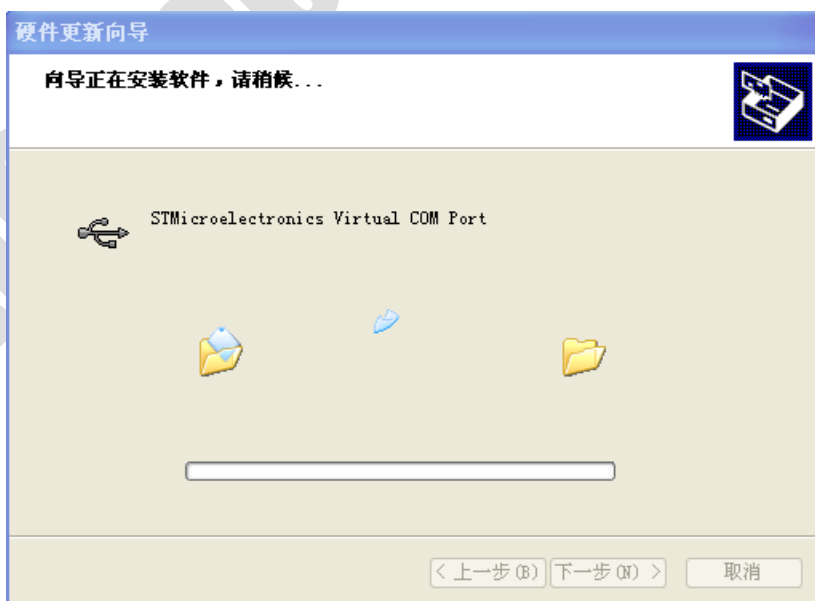


图 2-8

- 9 稍待片刻即完成驱动程序安装，点击“完成”。

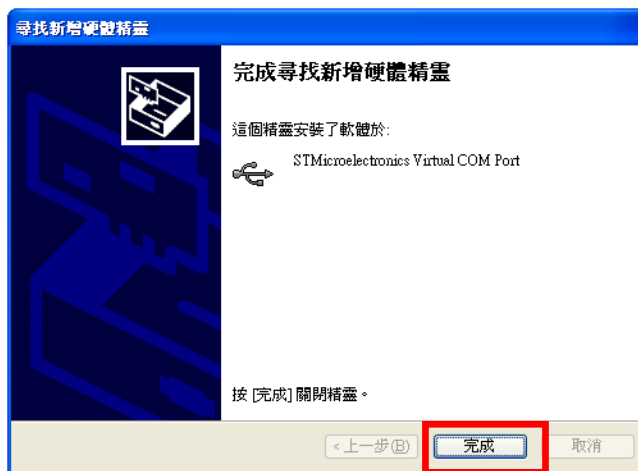


图 2-9

- 10 系统显示新硬件已安装且可使用。

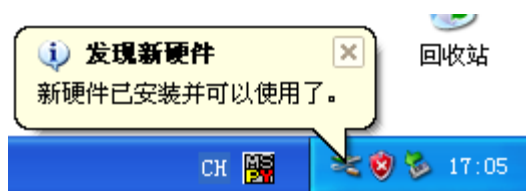


图 2-10

11 由 Windows 控制面板->系统->硬件->设备管理器中，可以看到多了 **STMicroelectronics Virtual COM Port** 装置，已经完成 Starter-Kit 的 USB Driver 安装。

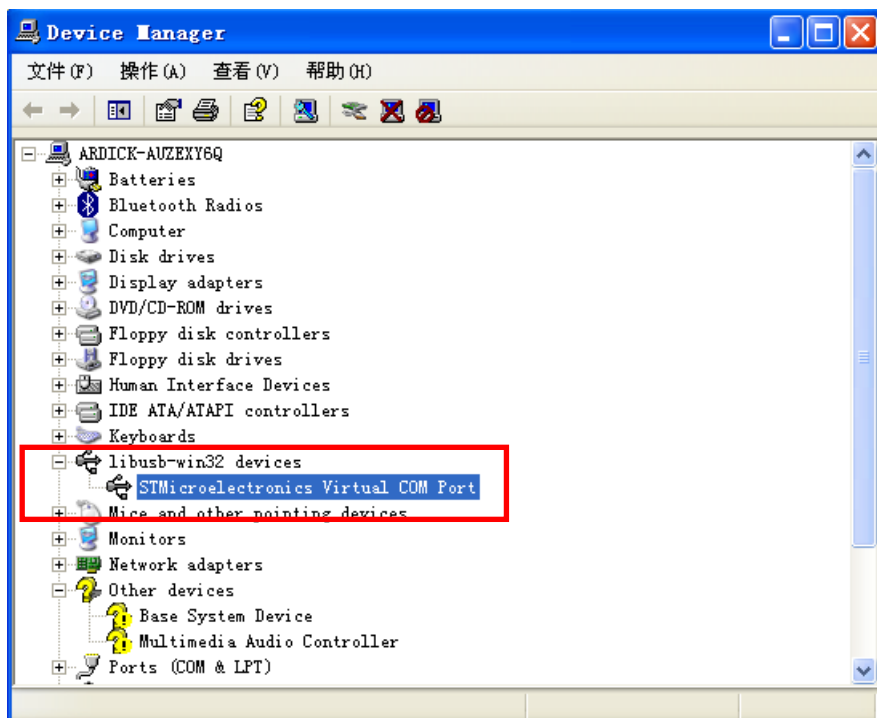


图 2-11

2.1.2 Windows 7/8/10 驱动程序安装

与早期的操作系统不同，Windows 8 的驱动程序有着严格的数字签名限制，正因为如此，安装未经过签属的驱动程序，必须有额外的安装步骤，我们可以参考 [此教学视频](#) (或 [此教学网站](#))，先暂时关闭数字签名的禁用功能，再安装驱动程序。如果系统是在 Windows 7 运行，请跳过此步骤，从第一个步骤开始安装驱动程序。

1 第一次将 RA6807M Starter-Kit 通过 USB Cable 连接至 PC(Personal computer)时，Windows 会出现 “正在安装设备驱动器软件” 的提示信息。

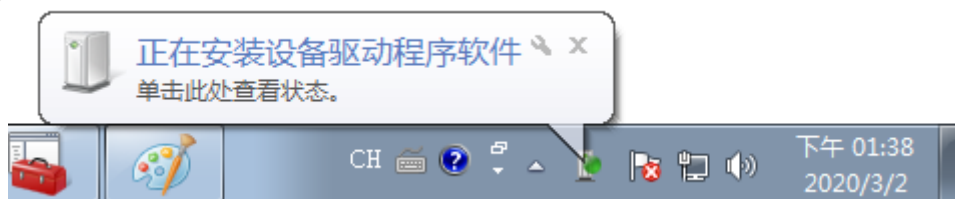


图 2-12

- 2 设备管理器窗口中，点击**其他装置**，于 **STM32 Virtual COM Port Driver** 点击鼠标右键，再点击“更新驱动程序软件”。

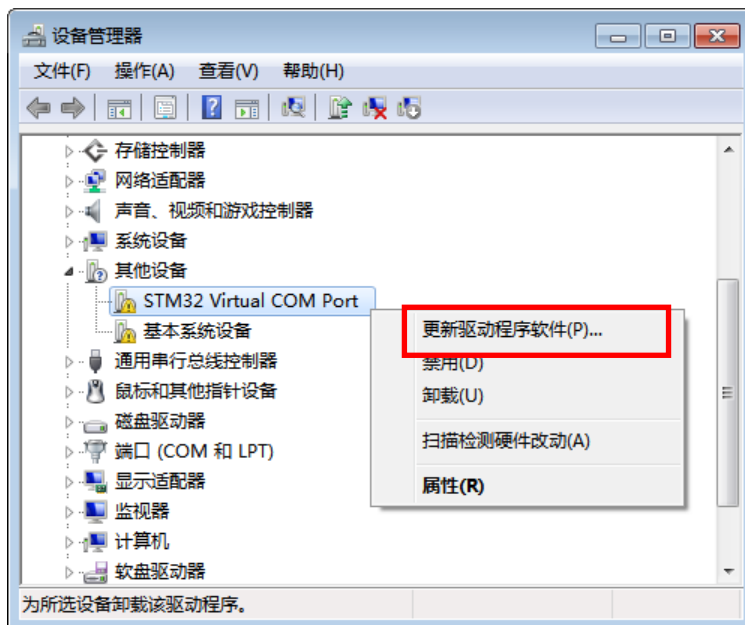


图 2-13

- 3 点击“浏览计算机以查找驱动程序软件”

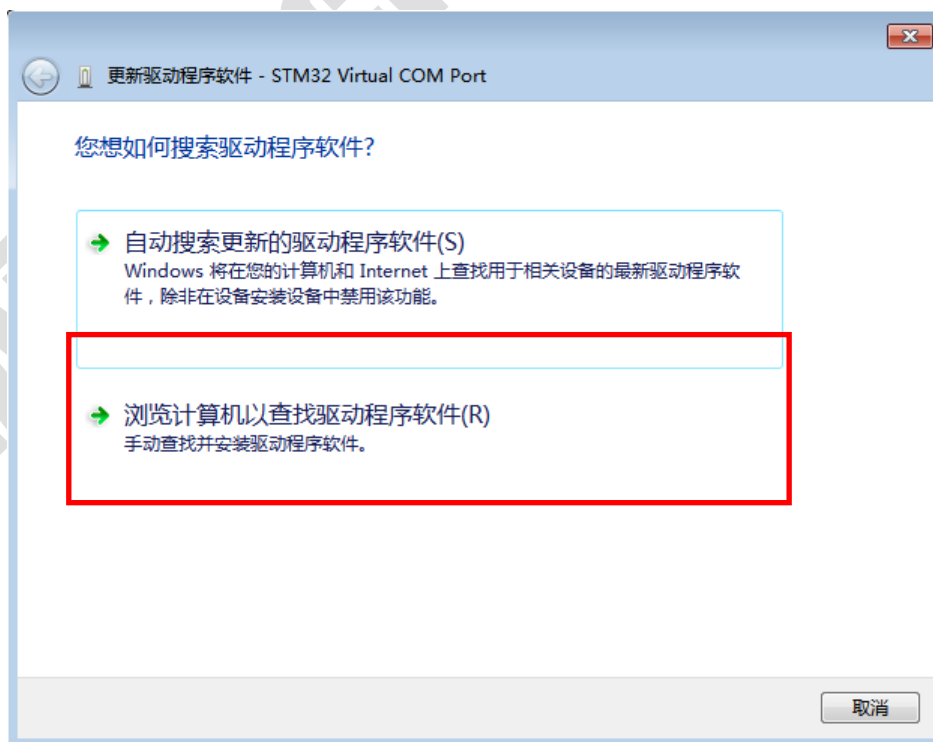


图 2-14

4 点击“从计算机的设备驱动程序列表中选择”

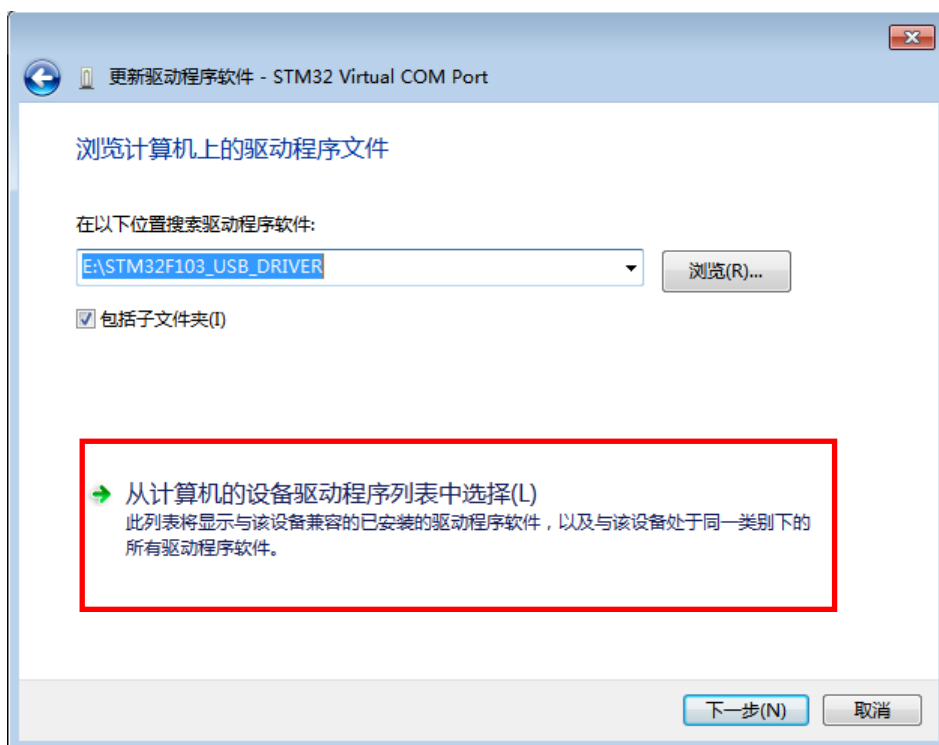


图 2-15

5 点击“从磁盘安装”

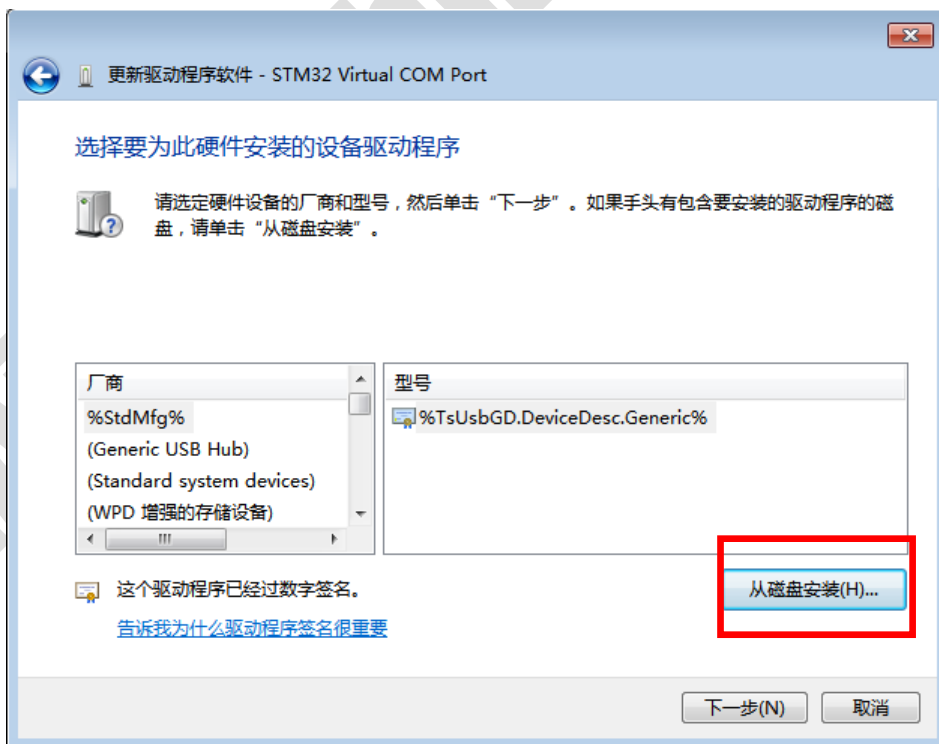


图 2-16

- 6 点击“浏览”，选择驱动程序的所在路径，点击“确定”。

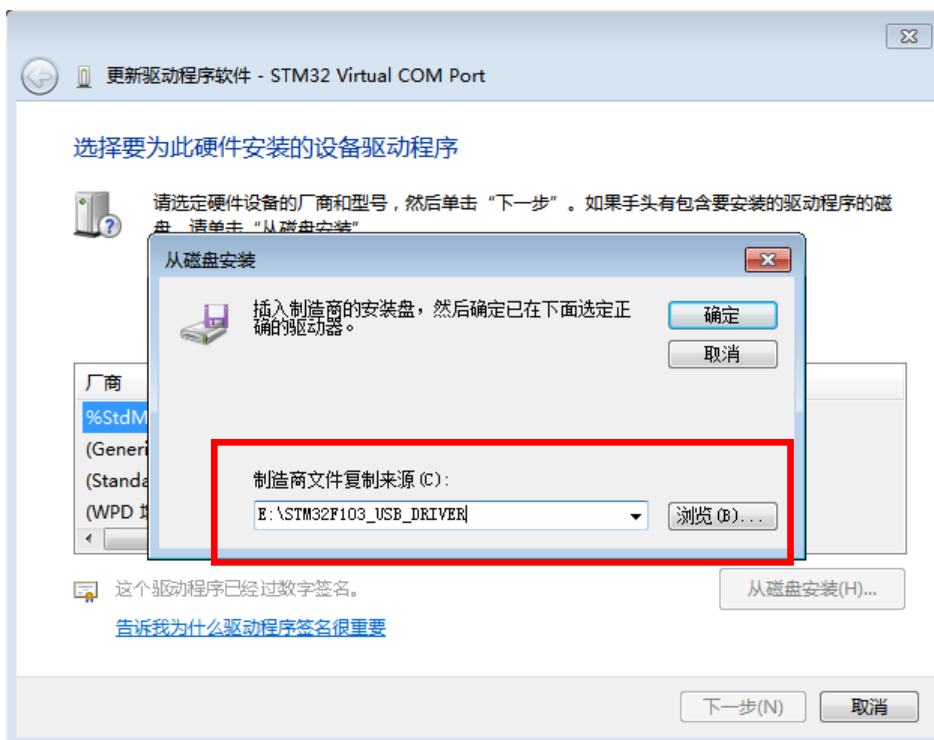


图 2-17

- 7 点击“下一步”

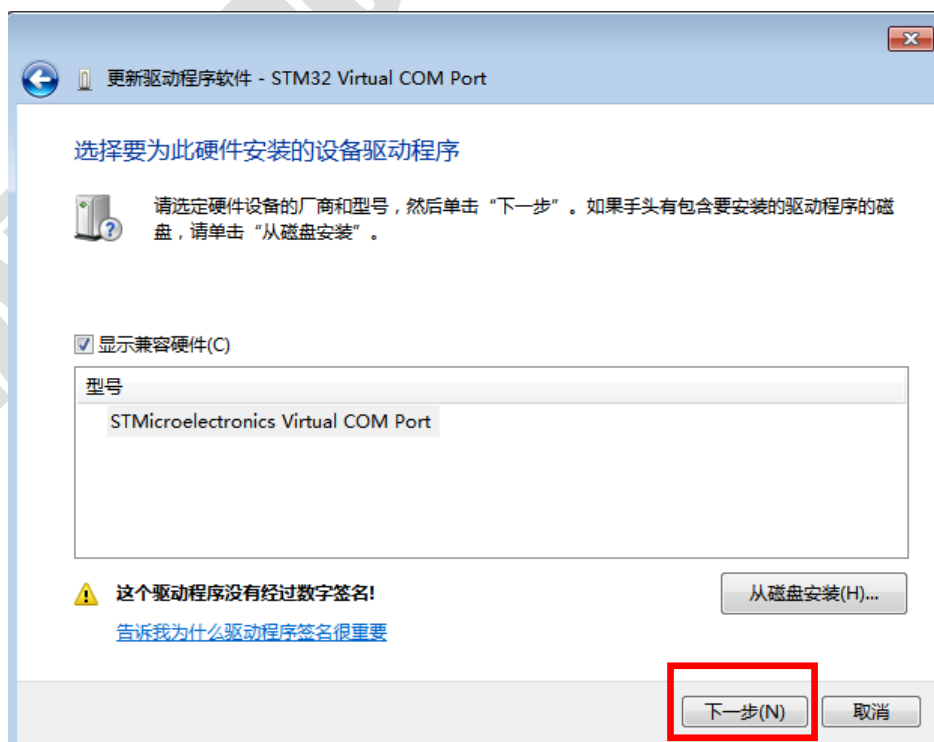


图 2-18

- 8 开始安装驱动程序，请等待。

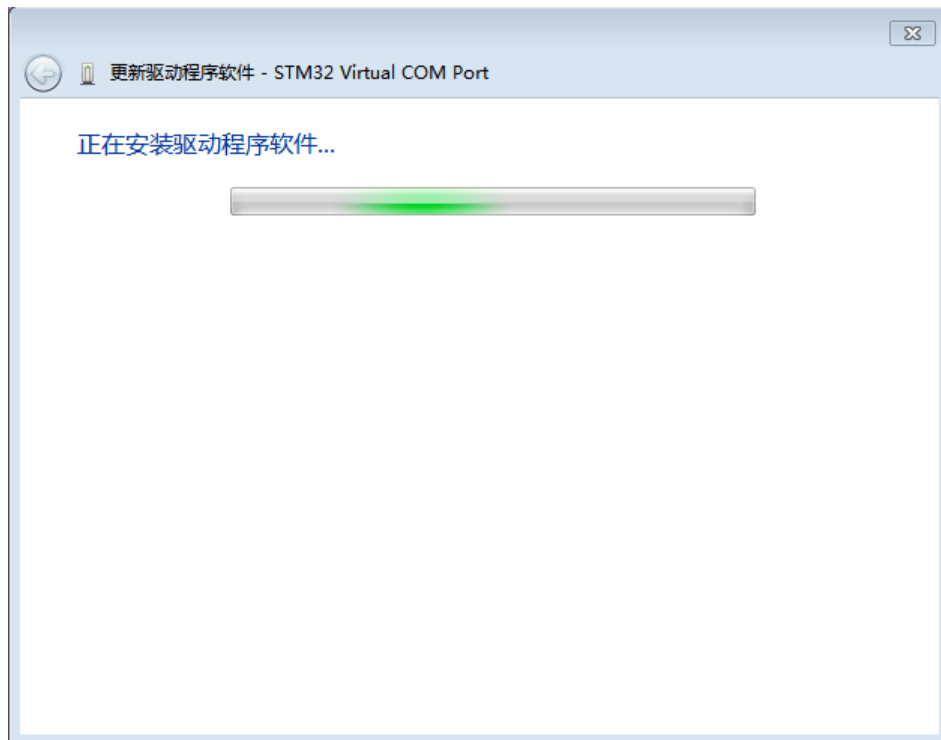


图 2-19

- 9 点击“始终安装此驱动程序软件”

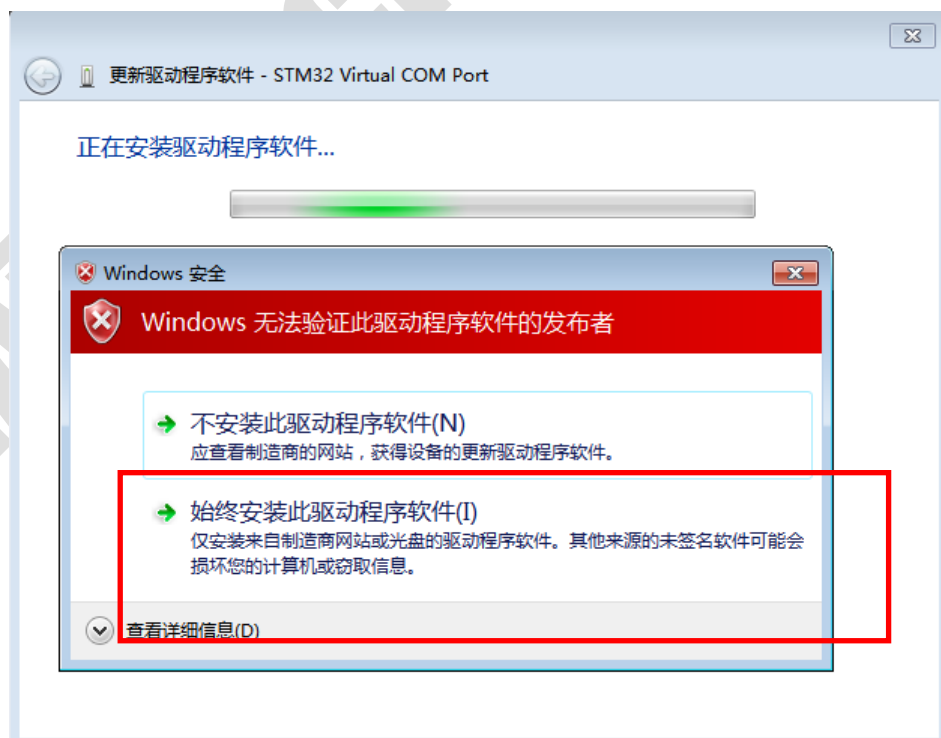


图 2-20

- 10 驱动程序安装完成，点击“关闭”

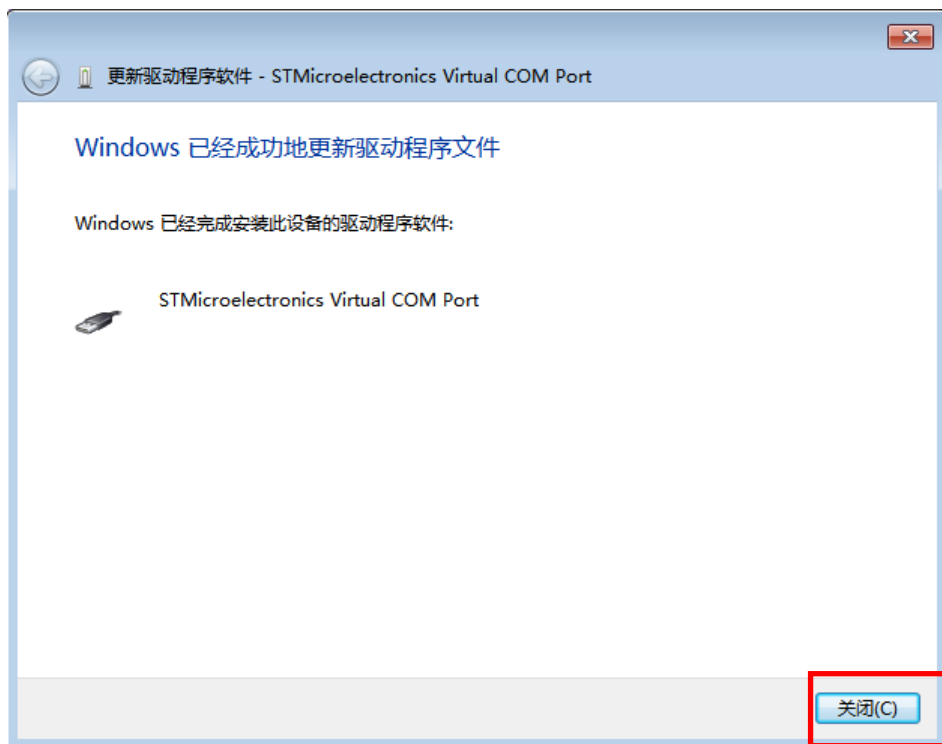


图 2-21

- 11 可以在设备管理器中找到 **STMicroelectronics Virtual COM Port** 装置，驱动程序安装完成并可以与装置通讯，请参考下图。

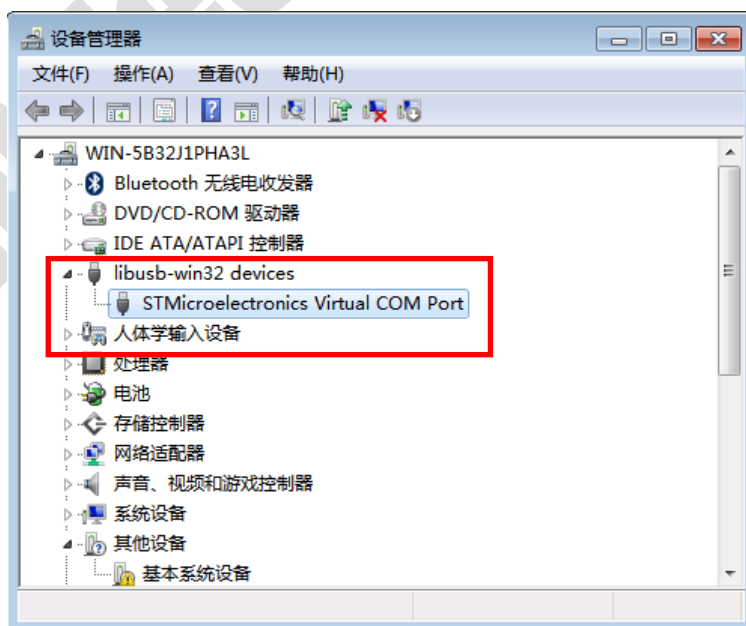


图 2-22

2.2 取得瑞佑软件密钥(RAiO tool key)与启用软件

- 1 执行 RAiO TFT-LCD Controller Tool (RA8889).exe



图 2-23

- 2 软件显示"Activate RAiO Tool" 窗口，提示输入有效的 RAiO tool key，点击  按键，复制 SID code，此 SID code 为此计算机专用，以此 SID code 申请所得的瑞佑软件密钥(RAiO tool key)，为此计算机与此软件专用，无法用于其他计算机或其他瑞佑软件。

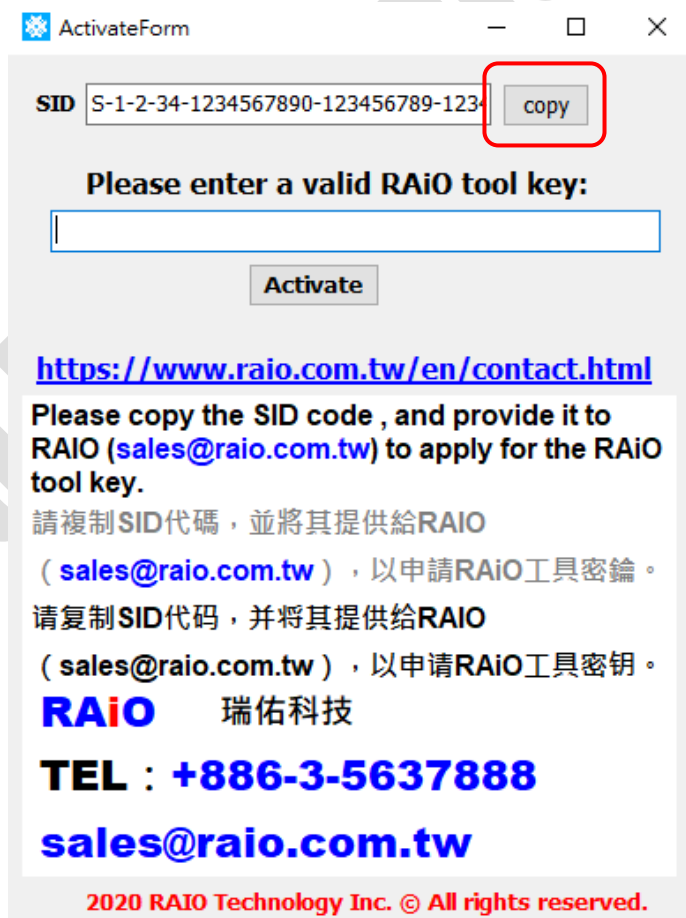


图 2-24

③ 将SID code以e-mail的方式，寄到RAiO 业务人员信箱(sales@raio.com.tw)，提出申请软件密钥需求，RAiO专员将会联系客户，取得客户相关数据并审核完成后，将提供相对应的软件密钥。

④ 客户取得有效的软件密钥后，将相对应的软件密钥输入于下图的字段(图 2-25 红框处)。

图 2-25

⑤ 点击 **Activate** 按钮，软件将进行启用，若软件没有出现警告讯息，则表示成功启用软件，软件可正常操作，若出现警告讯息(图 2-26)，软件密钥不正确，软件无法正常操作，请检查软件密钥是否有效，或重新申请软件密钥。

图 2-26

2.3 软件面板介绍

点击下面图示执行。

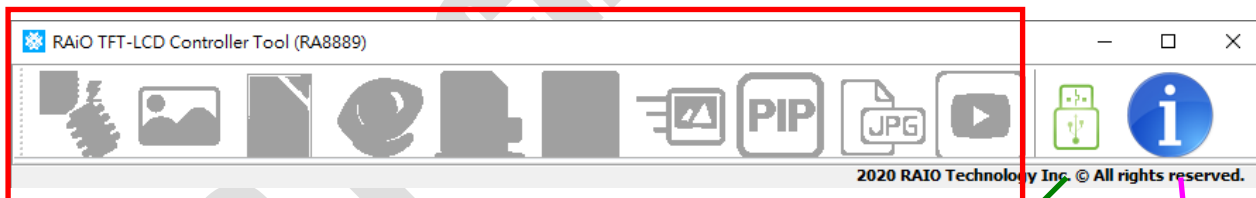


图 2-27

程序执行显示情况如下。

未与 PC USB 正常联机时:

- USB Connection Status (红色) 显示未正常联机
- IC Function Button 关闭, 无法选择所需功能
- Status Window 关闭



IC Function Button Disable

USB Connection Status

Software Information

图 2-28

与 PC USB 正常联机时:

- USB Connection Status (绿色) 显示正常联机
- IC Function Button 开启, 可选择欲操作的功能
- Status Window 开启, 并显示此时相关功能的参数设定值

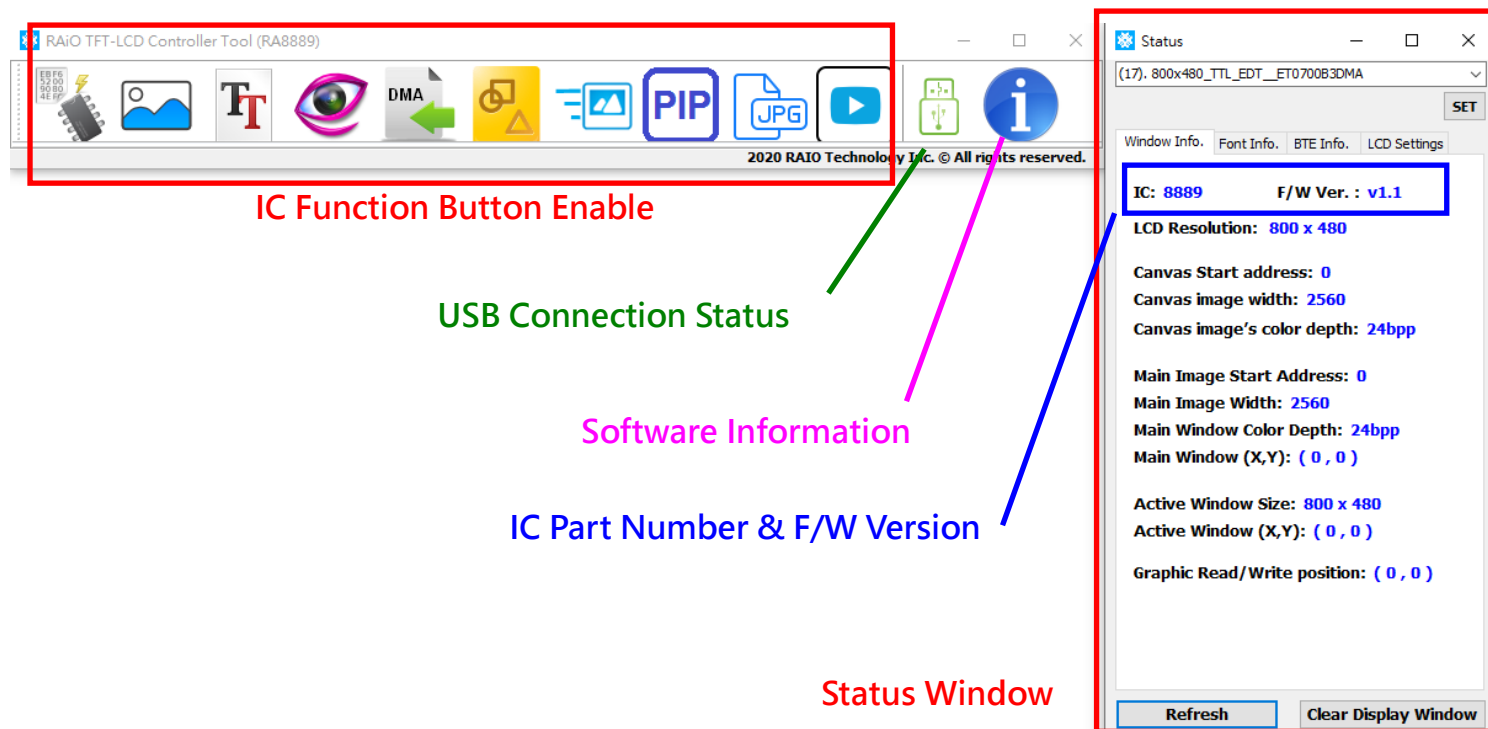


图 2-29

2.4 软件功能按键介绍

IC Function Button 可以选择 RA8889 常用的几个功能, 包含“SPI Flash Tool”, “MPU Write Tool”, “Text Function”, “Memory View”, “DMA Function”, “Draw Function”, “BTE Function”, “PIP Function”, “JPEG Function” and “AVI Function”等功能。

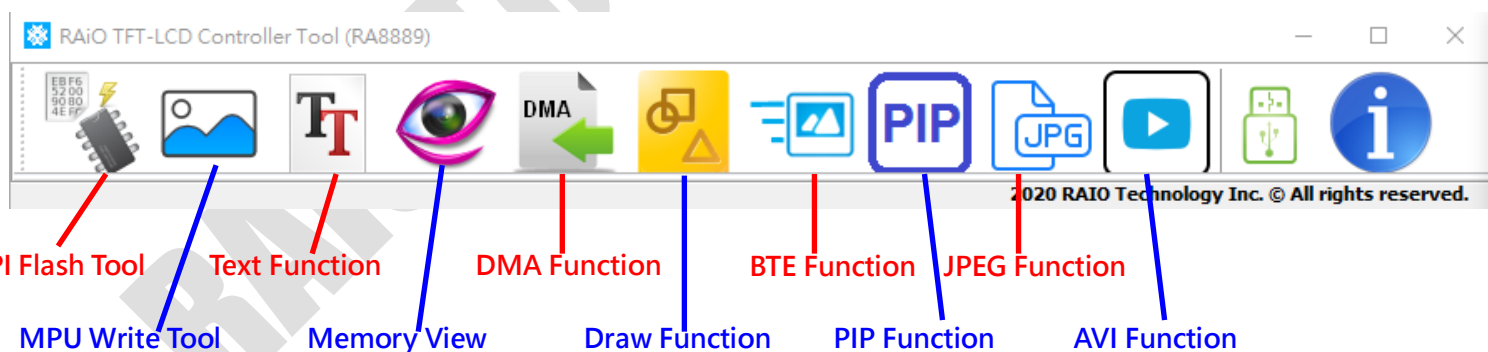
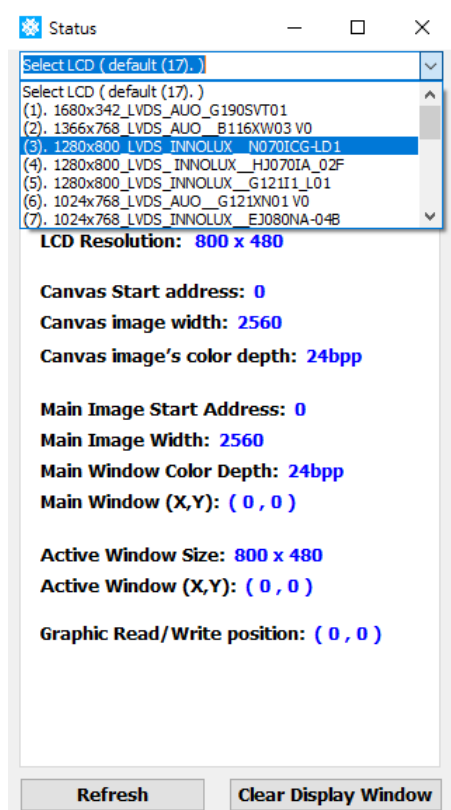


图 2-30

2.5 LCD 配置

我们可以由 Status window 下拉式选单中，选择所使用的 LCD，选择完毕，点击 SET 按钮确认。

1 Choose LCD from the combobox



2 Click SET button

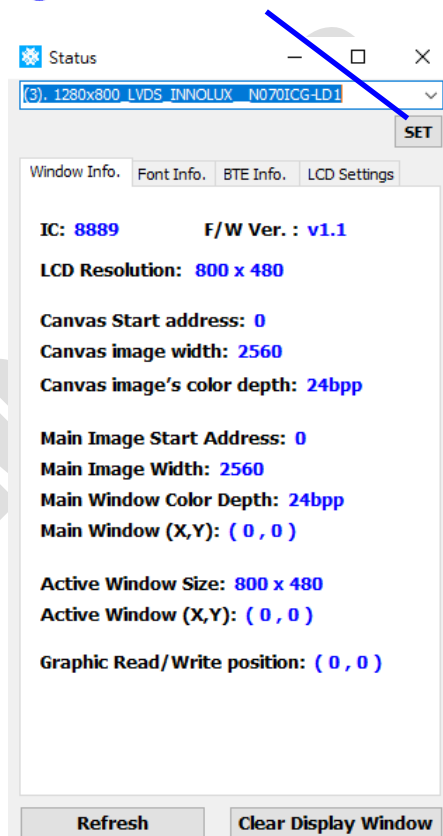
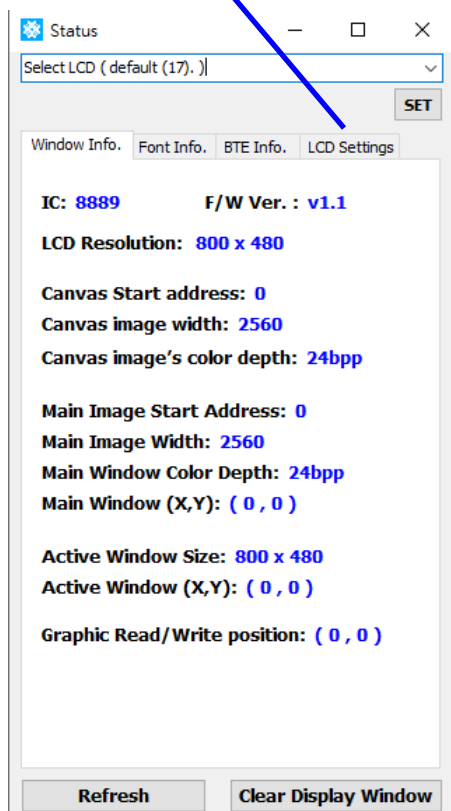


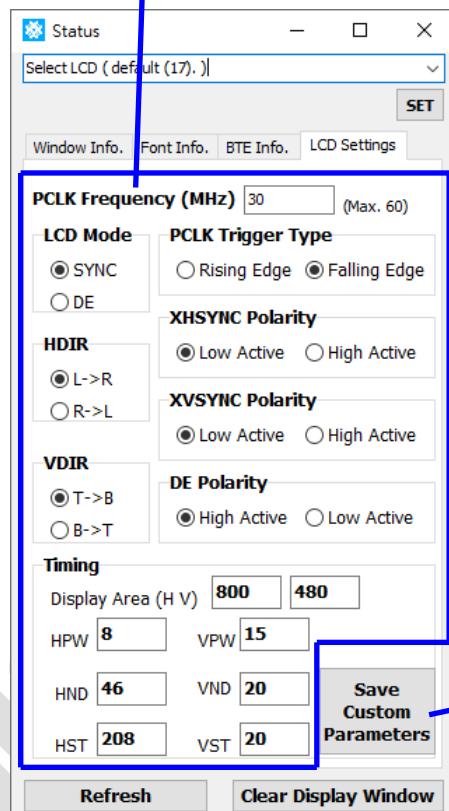
图 2-31

或由 LCD Settings 页面中，客制化设定 LCD 相关参数，并将设定结果储存。

1 Click LCD Settings Page



2 Adjust LCD parameters



3 Save

圖 2-32

客制化设定完毕。

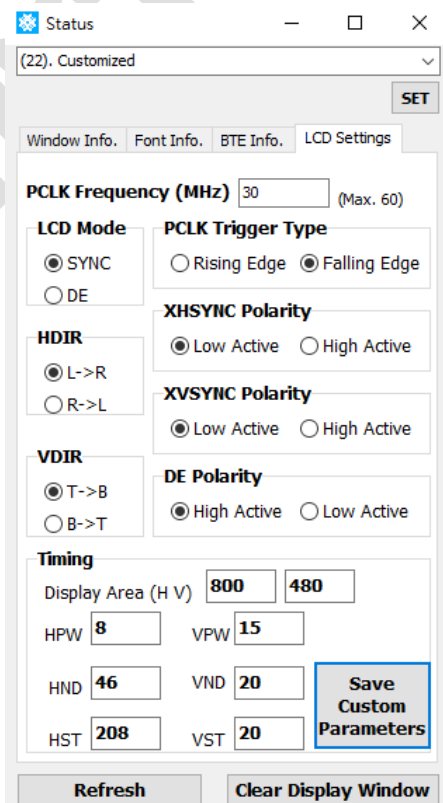


圖 2-33

3. SPI Flash Tool 烧录功能

RA8889 可以外接 SPI Flash, SPI Flash 主要放置图片、视频数据,或使用集通字库(Genitop Font ROM), 可以使用 DMA、JPEG、AVI 与字型功能, 将图片、视频或文字快速由 SPI Flash 搬至显示内存中, 进而在 LCD 画面上显示。

对于 SPI Flash 存放图片或视频数据的方式, 可以通过 IC 万用烧录器(例如: ALL-100A), 将资料烧录至 SPI Flash 中, 但在图片或视频数据未完全确定的产品开发过程中, 必须常常进行 SPI Flash IC 解焊、烧录、焊接的动作, 直到图片数据完全确定, 这相当费时与费力。

AP 中的 SPI Flash Tool, 则能避免上述繁琐的动作。利用 RA8889 的 SPI Master 功能, 对 SPI Flash 进行清除(chip erase)、烧录数据(Program)、数据比对(Verify)等动作。



点选 IC Function Button 中的  按键, 开启 SPI Flash Tool 功能。

操作画面显示如下图:

Select SPI bus and chip select

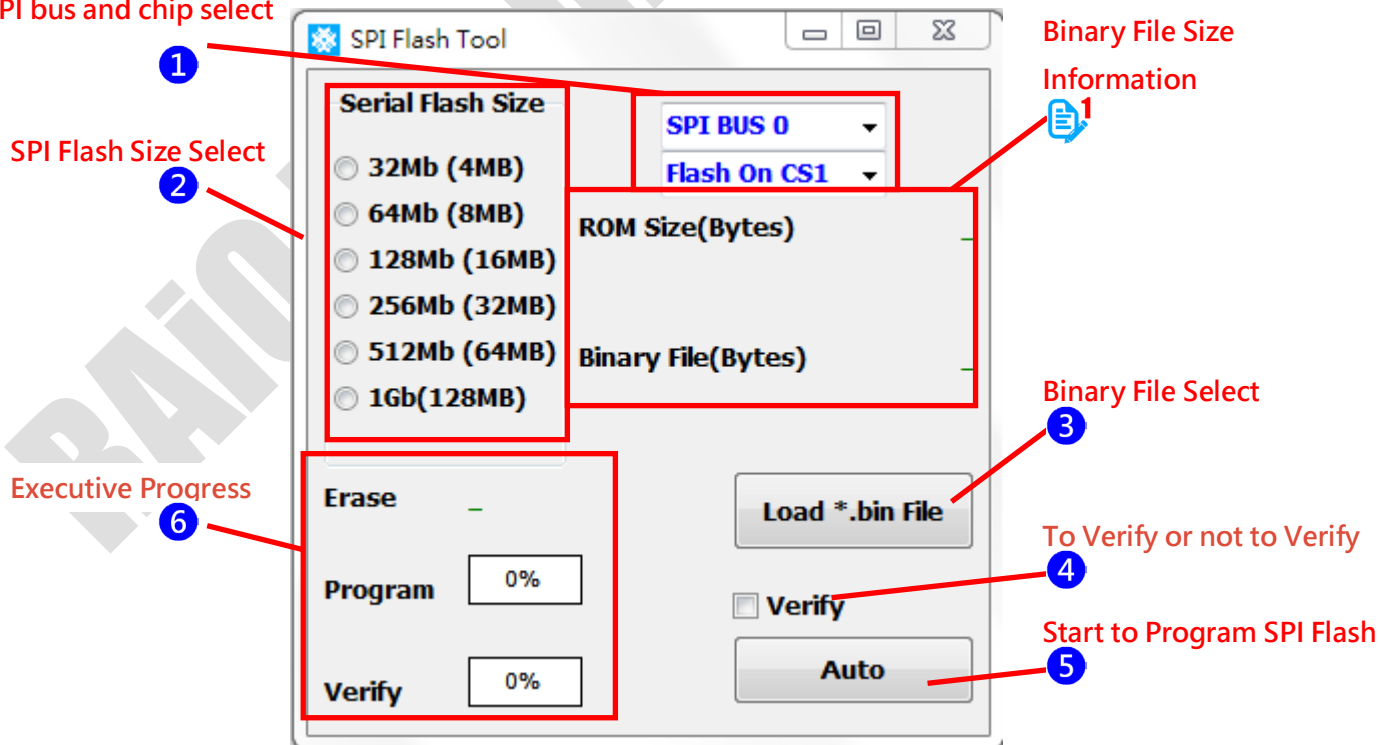


图 3-1

操作 SPI Flash 烧录步骤如下:

- ① 选择 SPI Flash 使用的 SPI BUS 与 CS (chip select)
- ② 选择所使用的 SPI Flash 容量。
- ③ 选取图形数据 Binary File (*.bin)。
- ④ 勾选数据是否进行比对(Verify)动作，进行 Verify 会花比较多的时间。
- ⑤ 点选 Auto 按键，进行烧录动作。
- ⑥ 等待执行进度显示 **Complete**，即完成 Erase、Program (and Verify) 等各项操作。



 若所选的 Binary File (*.bin)文件大小超过 SPI Flash 容量，会出现警告窗口，无法进行烧录动作。

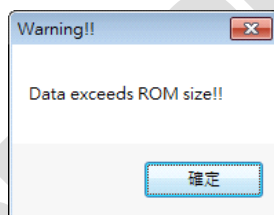


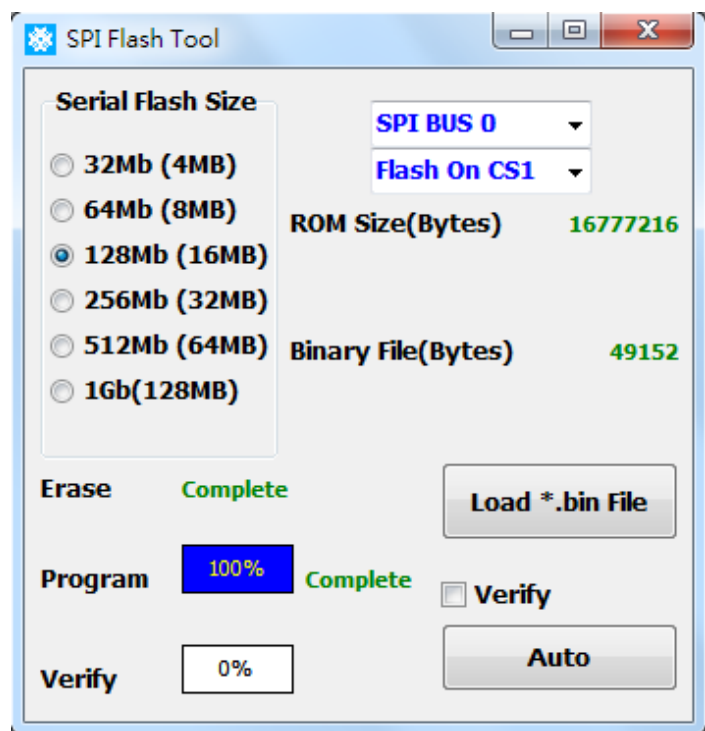
图 3-2

 RAiO 有另外一套转图软件 (Image Tool)，可以将*.bmp、*.jpg 转为*.bin 格式，建议可以使用这套专用软件。

 操作期间若出现 **Timeout !!**、**Fail !!** 等信息，请检查 SPI Flash 的相关电路、固件与软件相关设定是否正确。

例如: SPI Flash 的 BUS 与 CS 设定、SPI Flash 相关信号(protocol)设定、ROM Size 选择是否正确。

没有 Verify 正确烧录完成画面



有 Verify 正确烧录完成画面

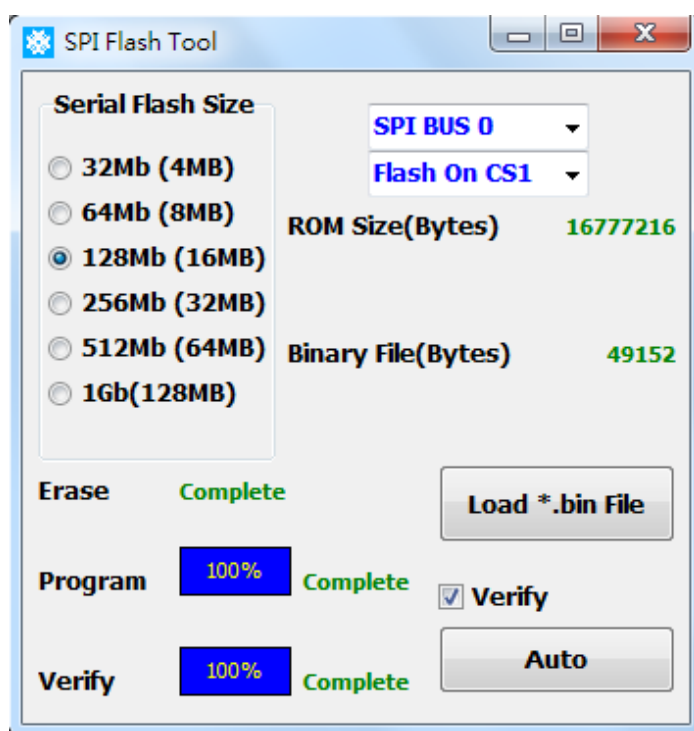


图 3-3

此功能已在下面所列的 SPI Flash 测试完成，若使用其它系列的 IC，发生无法正常烧录的情况，请联络 RAiO 。

MXIC: MX25L6406E, MX25L12835E, MX25L25635E.

Winbond: W25Q256FVFG.

Demo Video:

YouTube

<https://www.youtube.com/watch?v=rjlcG84PE>

youku

http://v.youku.com/v_show/id_XODk4NzUyNTc2.html

4. MPU 写图功能

通过瑞佑提供的 RA8889 Starter-Kit, 此功能将 PC 中的图片数据, 经由 USB 传输与 MCU 写入, 将数据写至 RA8889 的 SDRAM 中, 进而将图片数据显示在 LCD 画面上。同时, 也会基于使用者的设定, 产生一段 C 语言的程序段, 用户可以将此程序段直接移植到他们自己的固件程序中, 由于客户也会同时需要很多子程序去控制 RA8889, 为提高客户的便利性, 故瑞佑已备妥许多相关的 API, 提供给我们的客户, 客户毋需花时间自行撰写。欢迎联络我们 , 或者是拜访我们的官方网站。

本功能支持颜色深度(Color Depth) 24bpp 与 1bpp 的 Bitmap 与 JPEG 原始图形文件, 颜色深度 1bpp 的单色图主要应用在 RA8889 BTE Color Expansion 功能上, 需配合 AP 的 BTE Function 功能。而对于 24bpp 的彩图, AP 会判断 RA8889 此时设定在那一种颜色深度(8/16/18bpp), 会将 24bpp 的图形文件, 转为 IC 所使用的颜色深度数据格式, 再将数据传输给 IC, 完成正常显示, 此应用为一般常用的应用。



点选 IC Function Button 中的  按键, 开启 MPU Write Tool 功能。

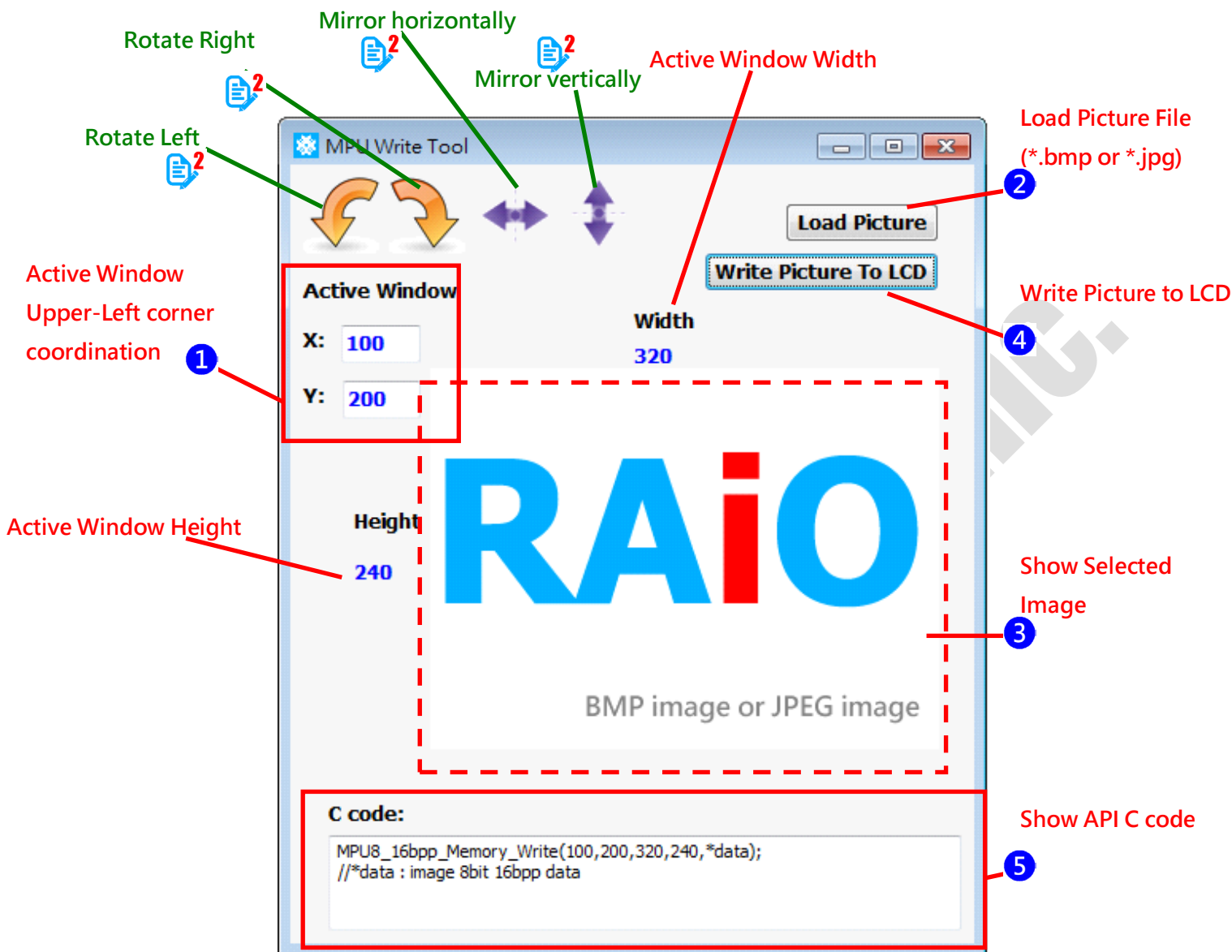


图 4-1

操作 MPU Write Tool 步骤如下:

- ① 依欲存放数据的内存区域，设定 Active Window 起始坐标(X,Y)
- ② 开启图片文件(*.bmp or *.jpg)，支持颜色深度 24bpp 与 1bpp 图形文件，在此 1bpp 图文件应用于 BTE Color Expansion 功能，须配合 BTE Function Tool 实现。
- ③ 开启图片文件后，会显示图形数据、图形宽度与高度。
- ④ 按下 Write Picture To LCD 按键，立即将数据通过 USB 传输，存放于内存设定好的区域，完成操作。
- ⑤ 在 C code 区域内，会显示此操作的 API 程序内容，以方便用户撰写程序。





RA8889 功能强大，相关的缓存器设定也多，RAiO 为了让使用者能快速熟悉，除了提供软件 AP 之外，也提供了固件的 API，将常用到的功能做成子程序，用户只要呼叫子程序，并代入相关的参数与数据，即可完成操作。



图片旋转与镜像的功能，仅支持颜色深度 24bpp 的图片。

实际 LCD (1280x800)画面显示(因照相的关系，颜色与实际情况有误差):

(X=0, Y=0)



图 4-2

Demo Video:

YouTube

<https://www.youtube.com/watch?v=Ly-8XRgCNyY>

youku

http://v.youku.com/v_show/id_XOTAwMTA5ODUy.html

5. 字型功能

5.1 内建字库功能

RA8889 提供了字型的功能，除了内建的 12x24 大小的 ISO/IEC 8859-1/2/4/5 字形 (Internal CGROM) 之外，还支持中国上海集通公司的部份外部字库 ROM (Genitop CGROM)，可以依字型需求，选择有支持的字库 ROM 使用，支持的字库型号为 GT21L16T1W、GT30L16U2W、GT30L24T3Y、GT30L24M1Z、GT30L32S4W、GT30L24F6Y 与 GT30L24S1W。使用者只须将 RA8889 切换为文本模式 (Text mode)，设定使用内建或外接的字型，并写入字型字码，RA8889 会解出相对应的字型数据，并写入设定好位置的内存中，显示于 LCD 画面。



點選 IC Function Button 中的  按键，开启 Text Function 功能。

Select Internal CGROM or Genitop CGROM ①

Select ISO/IEC 8859 (1/2/4/5) ②

Active Window Parameter ⑥

Text Editing Area ⑦

Show C Code ⑨

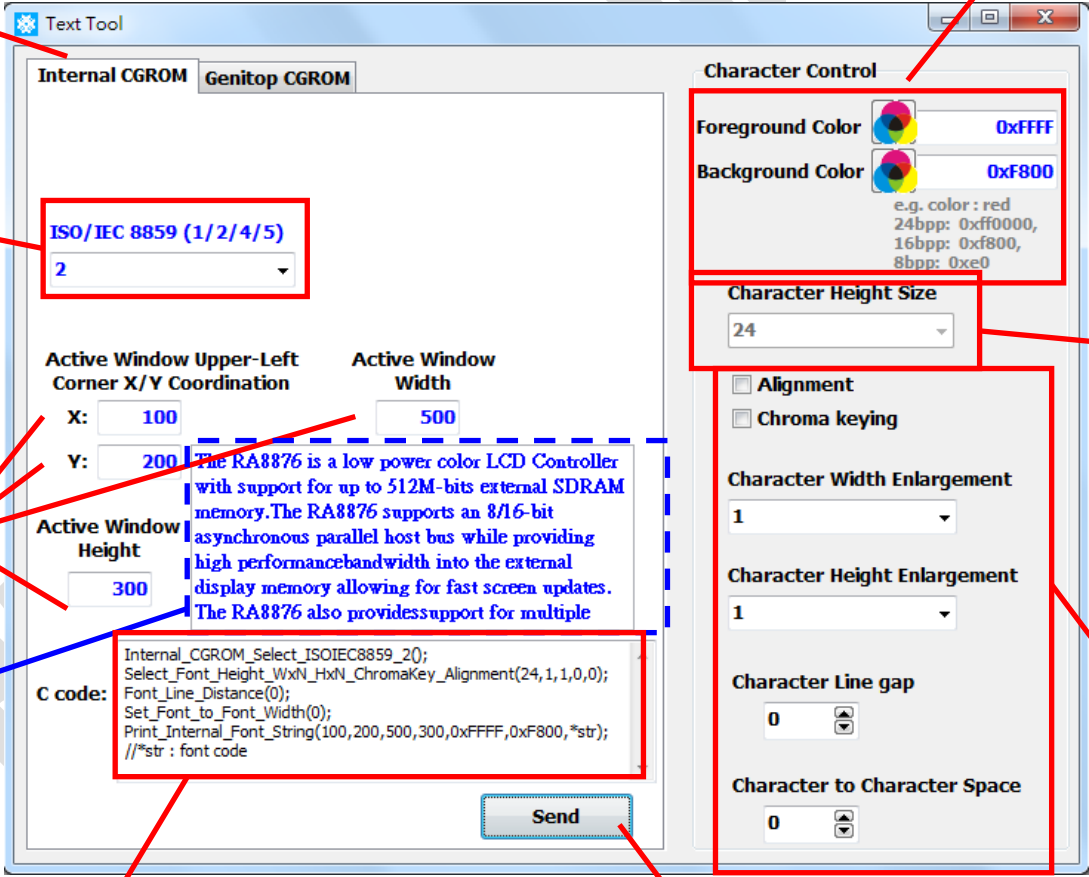
Run Text Function ⑧

Character foreground color and background color ③

Character Height Size ④

Internal CGROM only supports size 12x24 ⑤

Other related functions ⑤



Text Tool

Internal CGROM Genitop CGROM

ISO/IEC 8859 (1/2/4/5)
2

Active Window Upper-Left Corner X/Y Coordination
X: 100 Y: 200

Active Window Width: 500

Active Window Height: 300

The RA8876 is a low power color LCD Controller with support for up to 512M-bits external SDRAM memory. The RA8876 supports an 8/16-bit asynchronous parallel host bus while providing high performance bandwidth into the external display memory allowing for fast screen updates. The RA8876 also provides support for multiple

C code:
Internal_CGROM_Select_ISOIEC8859_2();
Select_Font_Height_WxN_HxN_ChromaKey_Alignment(24, 1, 1, 0, 0);
Font_Line_Distance(0);
Set_Font_to_Font_Width(0);
Print_Internal_Font_String(100, 200, 500, 300, 0xFFFF, 0xF800, *str);
/*str : font code

Send

Character Control

Foreground Color: 0xFFFF

Background Color: 0xF800

e.g. color : red
24bpp: 0xFF0000,
16bpp: 0xF800,
8bpp: 0xE0

Character Height Size: 24

☐ Alignment
☐ Chroma keying

Character Width Enlargement: 1

Character Height Enlargement: 1

Character Line gap: 0

Character to Character Space: 0

图 5-1

操作 Internal CGROM Text Function 步骤如下:

- ① 切换为使用 Internal CGROM
- ② 选择 Select ISO/IEC 8859 (1/2/ 4/5) 所要使用的字型。
- ③ 设定欲显示字的前景色(foreground color)与背景色(background color),输入的颜色资料格式,依 RA8889 当时的 Canvas color depth (8/16/24bpp)设定,可分为三种情况,以全红色数据设定为例:

8bpp, 数据格式设定为 0xe0
16bpp, 数据格式设定为 0xf800
24bpp, 数据格式设定为 0xff0000
- ④ Internal CGOM 仅支持 12x24 字号
- ⑤ 设定其它与字型相关的功能,字对齐(Alignment)、字背景透明(Chroma keying)、字型放大(Enlargement)、字与字间距(Character-to-character space)、行距(Character line gap)。
- ⑥ 设定 Active Window 相关参数, Active Window 起始 XY 坐标、Width 与 Height。 AP 会默认将字型模式下的字型光标(Text Cursor)起始坐标,设定与 Active Window 的起始 XY 坐标相同,之后随着文字增加自动增加地址。
- ⑦ 在文字编辑区域内,编辑有支持的文字与符号。
- ⑧ 按下 Send 按键,将文字写入内存中,显示于 LCD 画面上。
- ⑨ 显示内建字库功能 C 语言程序。

实际 LCD (1280x800)画面显示:

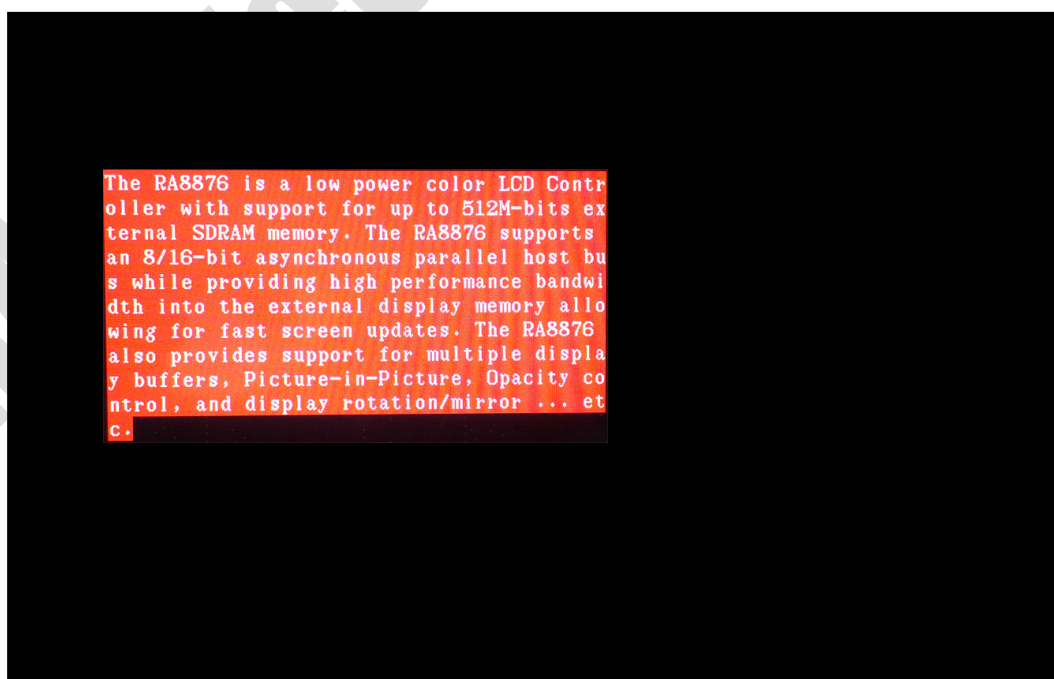


图 5-2

5.2 外接集通字库功能

当 RA8889 外接支持的集通字库 (Genitop CGROM), 通过 AP 简单的设定, 输入欲显示的文字或符号, 即可完成 GB2312、GB12345、BIG5、JIS、Unicode...等文字的显示。

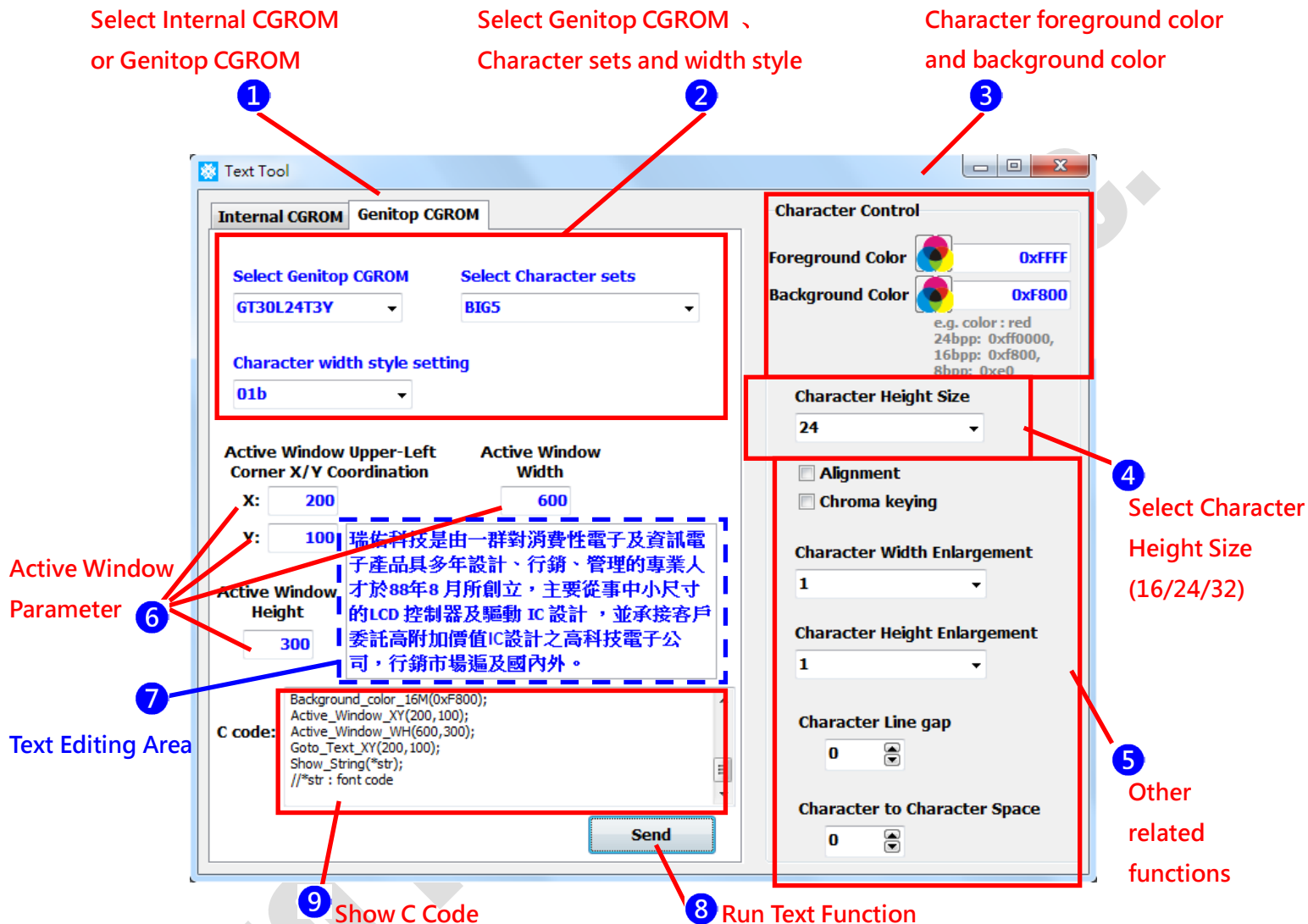


图 5-3

- ① 切换为使用 External CGROM (Genitop CGROM)
- ② 选择外接支持的集通字库 IC 型号, 并选择欲显示字型的类别与字宽风格。
- ③ 设定欲显示字的前景色(foreground color)与背景色(background color), 输入的颜色资料格式, 依 RA8889 当时的 Canvas color depth (8/16/24bpp)设定, 可分为三种情况, 以全红色数据设定为例:

8bpp, 数据格式设定为 0xe0

16bpp, 数据格式设定为 0xf800

24bpp, 数据格式设定为 0xff0000

- 4 选择欲显示字的字型高度(16/24/32)
- 5 设定其它与字型相关的功能，字对齐(Alignment)、字背景透明(Chroma keying)、字型放大(Enlargement)、字与字间距(Character-to-character space)、行距(Character line gap)。
- 6 设定 Active Window 相关参数，Active Window 起始 XY 坐标、Width 与 Height。AP 会默认将字型模式下的字型光标(Text Cursor)起始坐标，设定与 Active Window 的起始 XY 坐标相同，之后随着文字增加自动增加地址。
- 7 在文字编辑区域内，编辑有支持的文字与符号。
- 8 按下 Send 按键，将文字写入内存中，显示于 LCD 画面上。
- 9 显示外部字库功能 C 语言程序。

实际 LCD (1280x800)画面显示(因照相的关系，颜色与实际情况有误差):

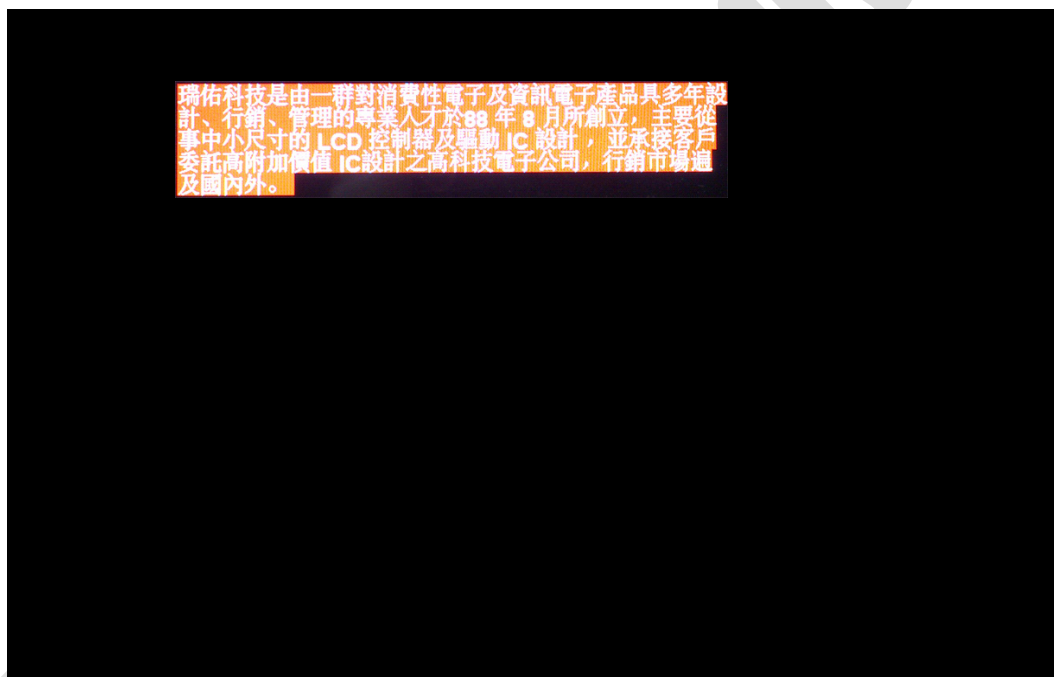


图 5-4

Demo Video:

YouTube

<https://youtu.be/QV2ARVI4iAE>

youku

http://v.youku.com/v_show/id_XMTQyMzgyNTgzMg==.html

6. DMA 功能

RA8889 的 DMA 功能，能将 SPI Flash 的资料，没有 MCU 介入之下，快速搬至 SDRAM 内存中。配合之前介绍的 SPI Flash Tool 功能，可以将所要显示的图形数据存放于 SPI Flash 中，再使用 DMA 快速将图形数据写入 SDRAM 内存中，进而显示于 LCD 画面上。

先利用 RAiO 提供的另一套转图软件 Image Tool，将所使用的图片转换且合并为一个 Binary File，转换完成后会产生 *.bin、*.xlsx 与 *.h 文件，*.bin 文件利用 SPI Flash Tool 功能烧录至 SPI Flash 中，而各图片的相关信息，则列于 *.xlsx 与 *.h 文件中。

关于 Image Tool 的操作，请参阅 Image Tool User Guide，下面以转换 RAiO 特制的六张 128x128 图库 ICON 为例，转换后产生 All_8889_Pic_NOR.bin、All_8889_Pic_NOR.xlsx 与 All_8889_Pic_NOR.h 文件。

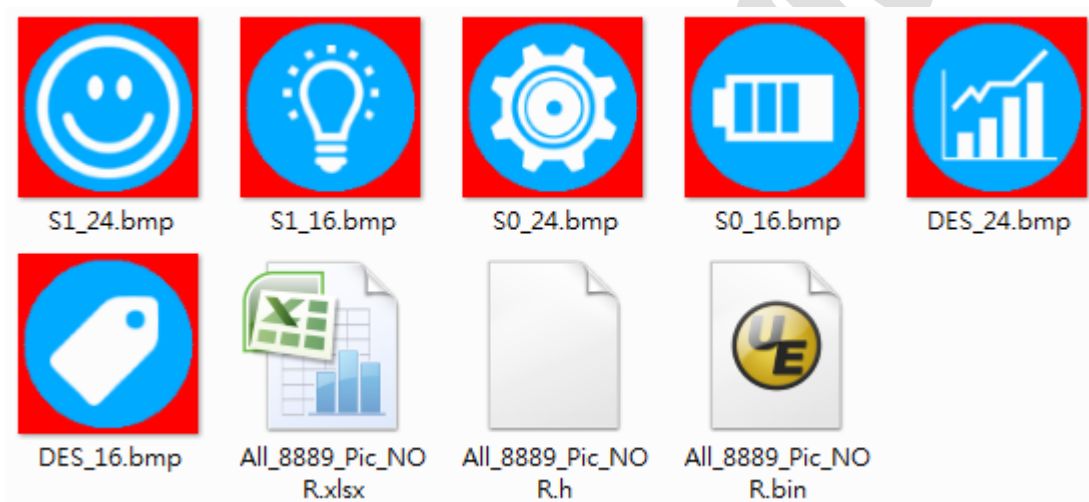


图 6-1

All_8889_Pic_NOR.xlsx 数据内容如下：

	A	B	C	D	E	F	G
1	No.	File Name	Width	Height	Data Format	Size (Bytes)	Start Address
2	1	DES_16.bmp	128	128	16bpp mode (RG	32768	0
3	2	DES_24.bmp	128	128	16bpp mode (RG	32768	32768
4	3	S0_16.bmp	128	128	16bpp mode (RG	32768	65536
5	4	S0_24.bmp	128	128	16bpp mode (RG	32768	98304
6	5	S1_16.bmp	128	128	16bpp mode (RG	32768	131072
7	6	S1_24.bmp	128	128	16bpp mode (RG	32768	163840
8						196608	

图 6-2

将" All_8889_Pic_NOR.bin"烧录至 SPI Flash 后, 开始操作 DMA 功能。



点选 IC Function Button 中的 按键, 开启 DMA Function 功能。

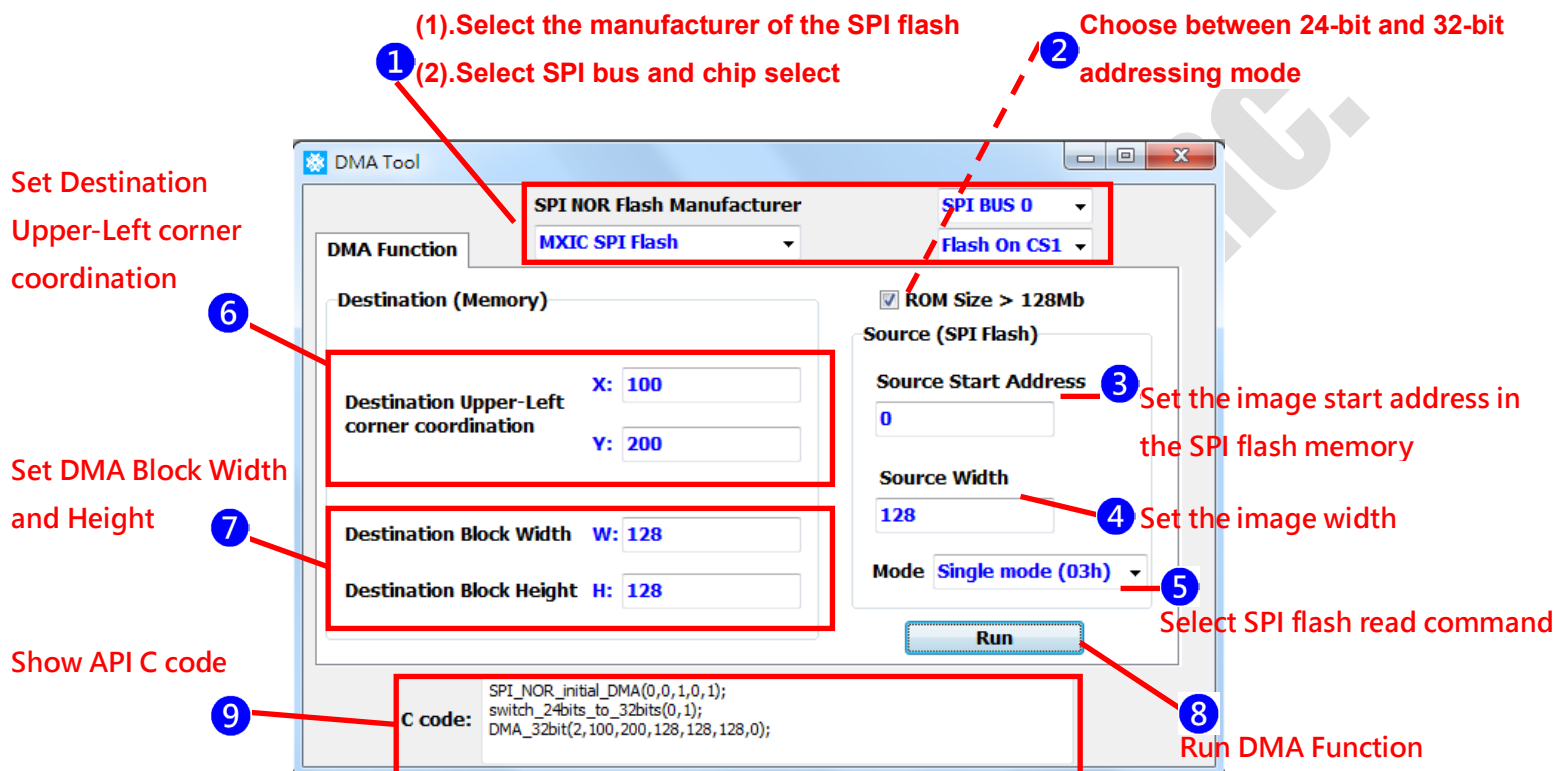


图 6-3

- (1).选择 SPI NOR Flash 厂商(MXIC 或 Winbond, 若使用其它厂商 flash, 请与瑞佑联络) (2).选择操作 SPI NOR Flash 的 SPI bus 与 CS
- 选择操作 SPI NOR Flash 的寻址方式 (>128Mbits : 32-bit addressing mode, ≤128Mbits : 24-bit addressing mode)
- 设定欲显示的图片, 存放于 SPI Flash 的起始地址, 以显示 DES_16.bmp 图形为例, 设定为 0
- 设定原始图片的 Width, 在此设定 128。
- 选择 SPI flash 读取数据命令
- 设定数据经过 DMA 搬至 SDRAM 内存的位置 Canvas 坐标, 例如数据要显示在(X=100,Y=200)位置。
- 设定所要搬动的图片区块的 Width 与 Height, 在此设定搬一整张 128x128 的完整图案数据。
- 按下 Run 按键, 执行 DMA 功能。
- 显示 DMA API C 语言程序。



超过 128Mbits 的 SPI Flash Memory, 要以 32bits 的方式寻址, 128Mbits 以下则以 24bits 寻址, 而 SPI Flash Memory 预设是以 24bits 的方式接受指令, 若选用超过 128Mbytes 的 SPI Flash, 使用前必须下指令, 将 SPI Flash 设定为 32bits 地址模式。

当步骤 2 勾选 ROM Size>128Mb, 执行 DMA 后, 显示的 C code, 会多一行 switch_24bits_to_32bits API 程序段, 且执行 DMA_32bit 的 API 程序, RA8889 也会以 32bits 的地址方式执行 DMA 功能; 若 2 未勾选, 则仅执行 DMA_24bit API 程序, 以 24bits 地址的方式执行 DMA。如下图所示。

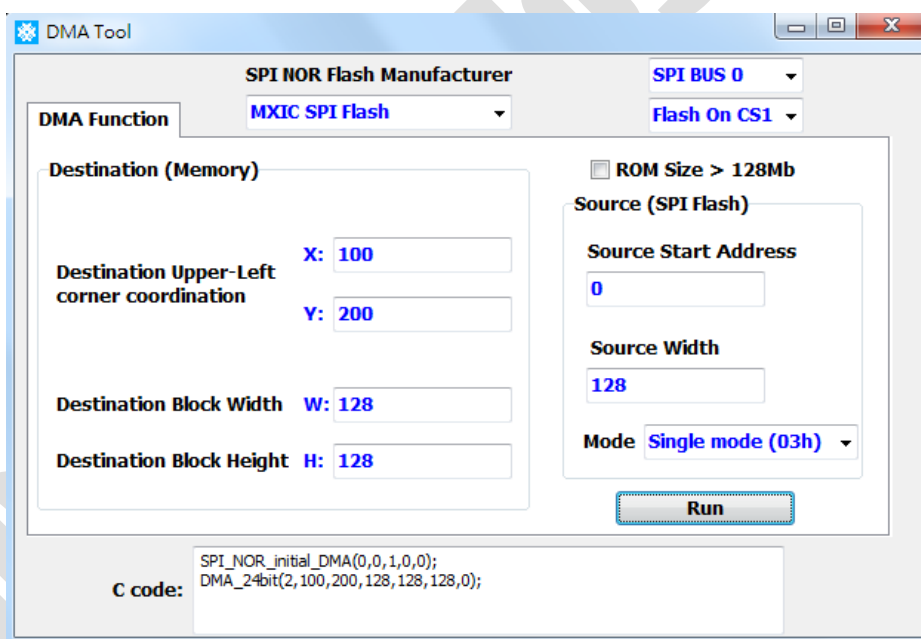


图 6-4

执行后画面显示结果:

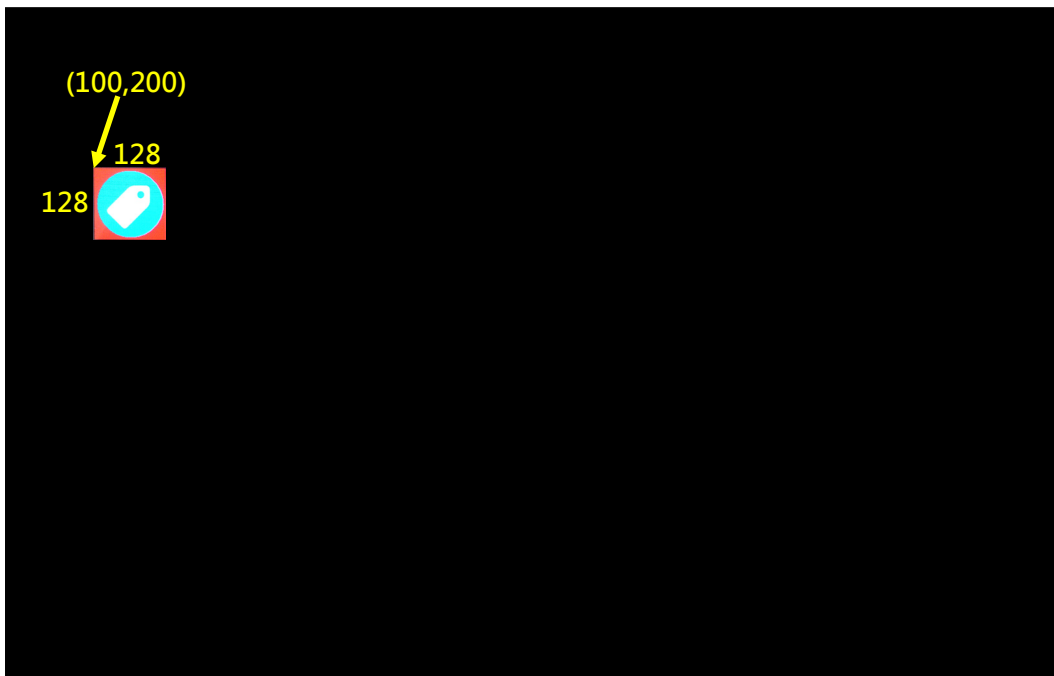


图 6-5

DMA 也可以显示原始图形的部份区块，以显示 S1_24.bmp 的一半宽度图形为例。

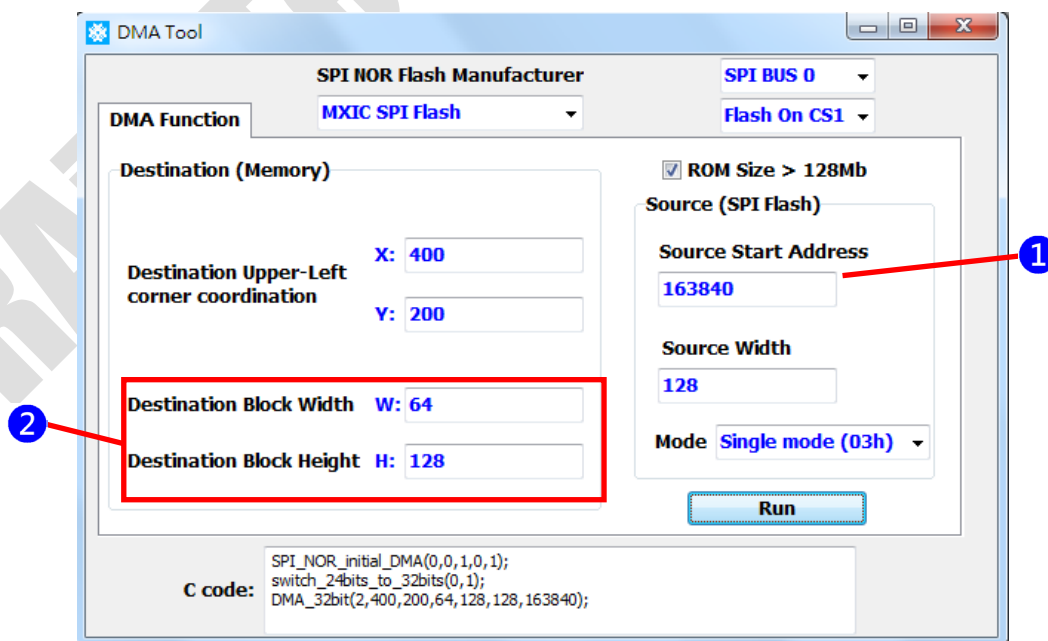


图 6-6

操作步骤大致如前例所述，其中较不一样的地方如下：

- ① S1_24.bmp 图形数据，存放于 SPI Flash Memory 的起始地址，如 All_8889_Pic_NOR.xlsx 所示为 163840。
- ② 由于只显示宽度一半的图形(64x128)，Width 要设定为 64。

执行后画面显示结果：

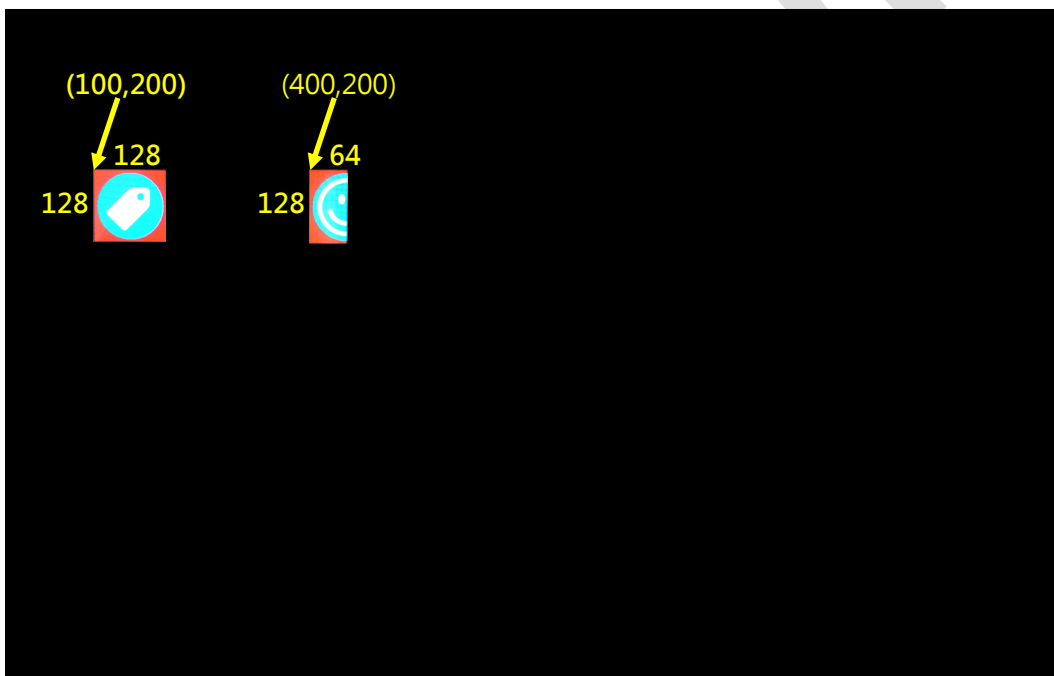


图 6-7

Demo Video:

YouTube

https://www.youtube.com/watch?v=PJ-6_h9sFMk

youku

http://v.youku.com/v_show/id_XOTAwMjAwNDY4.html

7. 几何图形绘图功能

RA8889 提供几何图形画图(Geometry drawing)功能，支持 Ellipse、Circle、Curve、Square、Line、Triangle、Square Of Circle Corner 等图形，可以应用于按键形状或对话框。AP 提供简单的参数设定，即可绘制所需的几何图形。



点选 IC Function Button 中的 按钮，开启 Drawing Function 功能。

Active Window Upper-Left corner
coordination (X,Y)、Active Window Width and
Height

Select Geometric Shape

Drawing
Function
Parameters

Color

Color Fill or
not Fill

The screenshot shows the 'Draw Function' dialog box. It has a title bar 'Draw Function' and standard window controls. The main area is divided into several sections:

- Active Window**: Contains 'X: 0' and 'Y: 0' input fields.
- Width**: A text box containing '1280'.
- Height**: A text box containing '800'.
- Geometric Shape Selection**: A row of buttons: 'Ellipse Curve (UR)', 'Ellipse Curve (LR)', 'Square', and 'Circle Square'. The 'Square' button is selected.
- Start**: Contains 'X: 0' and 'Y: 0' input fields.
- End**: Contains 'X: 1279' and 'Y: 799' input fields.
- Color**: A color picker showing '0x0000' (black). Below it, text explains color formats: 'e.g. color : red', '24bpp: 0xff0000', '16bpp: 0xf800', '8bpp: 0xe0'.
- Color Fill**: A dropdown menu currently set to 'Fill'.
- Run**: A blue button to execute the drawing.
- C code**: A text area at the bottom containing the following code:


```
Active_Window_XY(0,0);
Active_Window_WH(1280,800);
Draw_Square_Fill(0x0000,0,0,1279,799);
```

Numbered callouts point to specific elements:

- 1: Select Geometric Shape (points to the 'Square' button)
- 2: Active Window Upper-Left corner coordination (X,Y)、Active Window Width and Height (points to the X, Y, Width, and Height fields)
- 3: Drawing Function Parameters (points to the Start X, Y fields)
- 4: Color (points to the color picker)
- 5: Color Fill or not Fill (points to the Color Fill dropdown)
- 6: Run (points to the Run button)
- 7: C code (points to the C code text area)

图 7-1

操作步骤:

- 1 选择所要显示的图形功能，在此以矩形(Square)为例，以黑色填满 1280x800 的 LCD 整个画面，这是清除画面的常用功能。
- 2 设定 Active Window 的相关参数，要注意的一点，Drawing Function 只可以在 Active Window 内执行其画图功能。Active Window 起始 XY 坐标设定为(X=0,Y=0)，Width 为 1280，Height 为 800。
- 3 矩形相关参数为左上起始坐标，与右下结束坐标，两点围起的区域即为矩形大小范围，不同形状有不同的参数，请详阅 IC Datasheet。起始坐标设为(X=0,Y=0)，结束坐标为(X=1279,Y=799)。
- 4 如先前的第 5 章字型功能操作步骤 3 所述，Color Depth 16bpp 的黑色设定为 0x0000。
- 5 设定矩形颜色填满(■)或不填满(□)。
- 6 执行 Drawing Function.
- 7 显示 Drawing Function API 程序段。

执行后画面显示结果:

(0,0)

1280

800

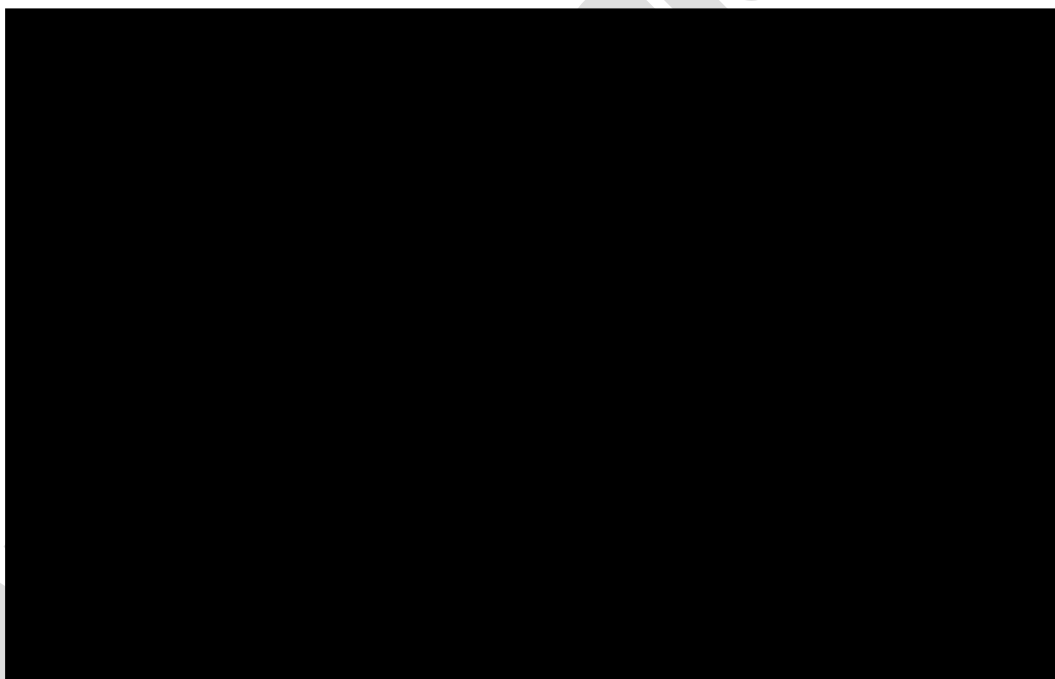


图 7-2

(1279,799)

以绘制矩形圆角为例：

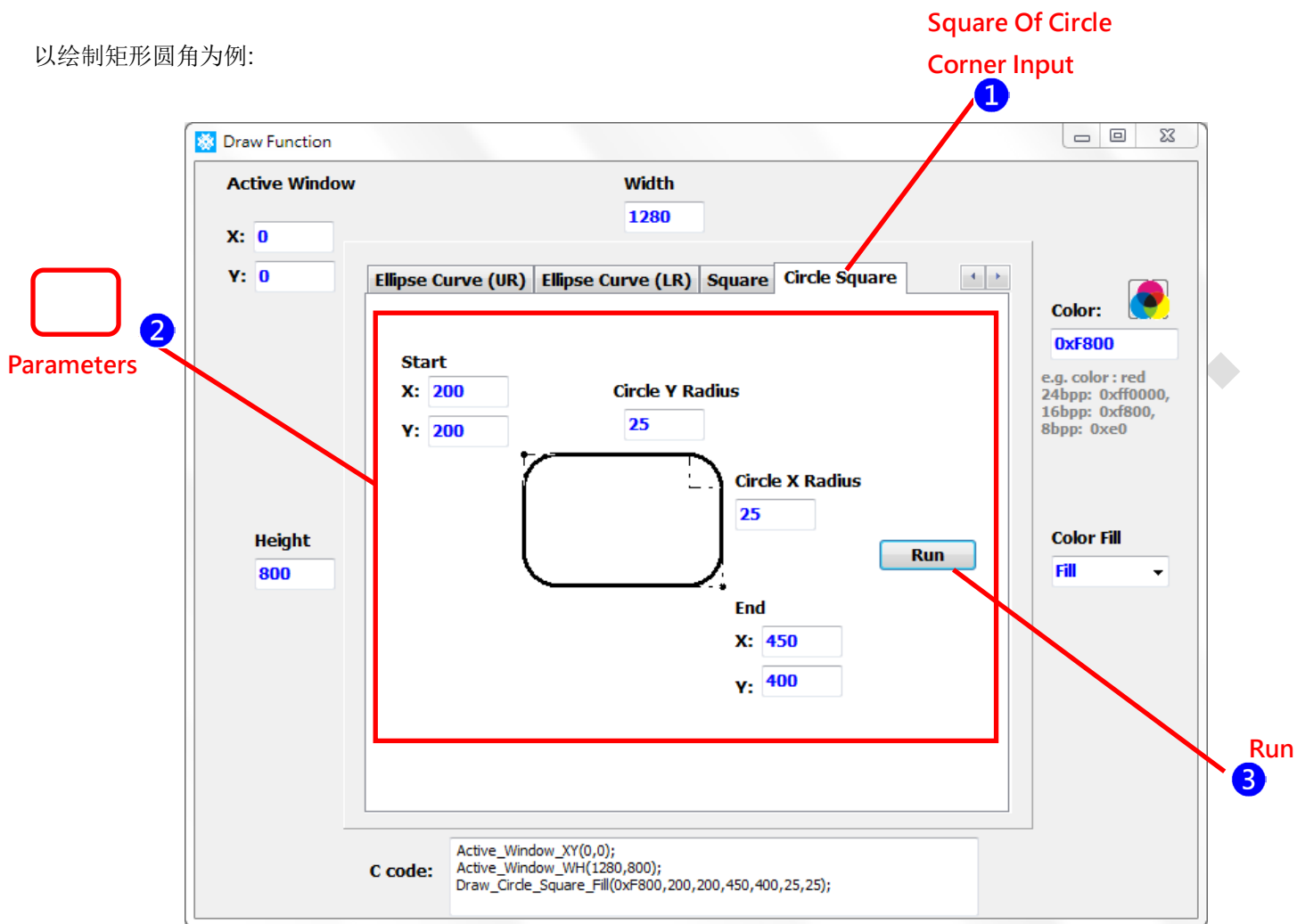


图 7-3

- ① 选择圆角方功能。
- ② 设定图形相关参数，左上起始 XY 坐标(X=200,Y=200)，右下结束 XY 坐标(X=450,Y=400)，圆角的 X 半径 25 与 Y 半径 25。
在此圆角的 X 半径与 Y 半径有条件限制，若填入的参数超过限制，会出现窗口警告，条件限制与警告窗口如下：

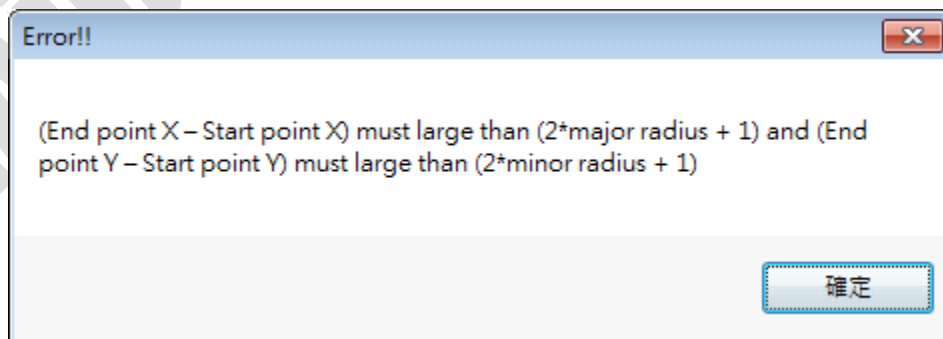


图 7-4

- ③ 执行功能

执行后画面显示结果:

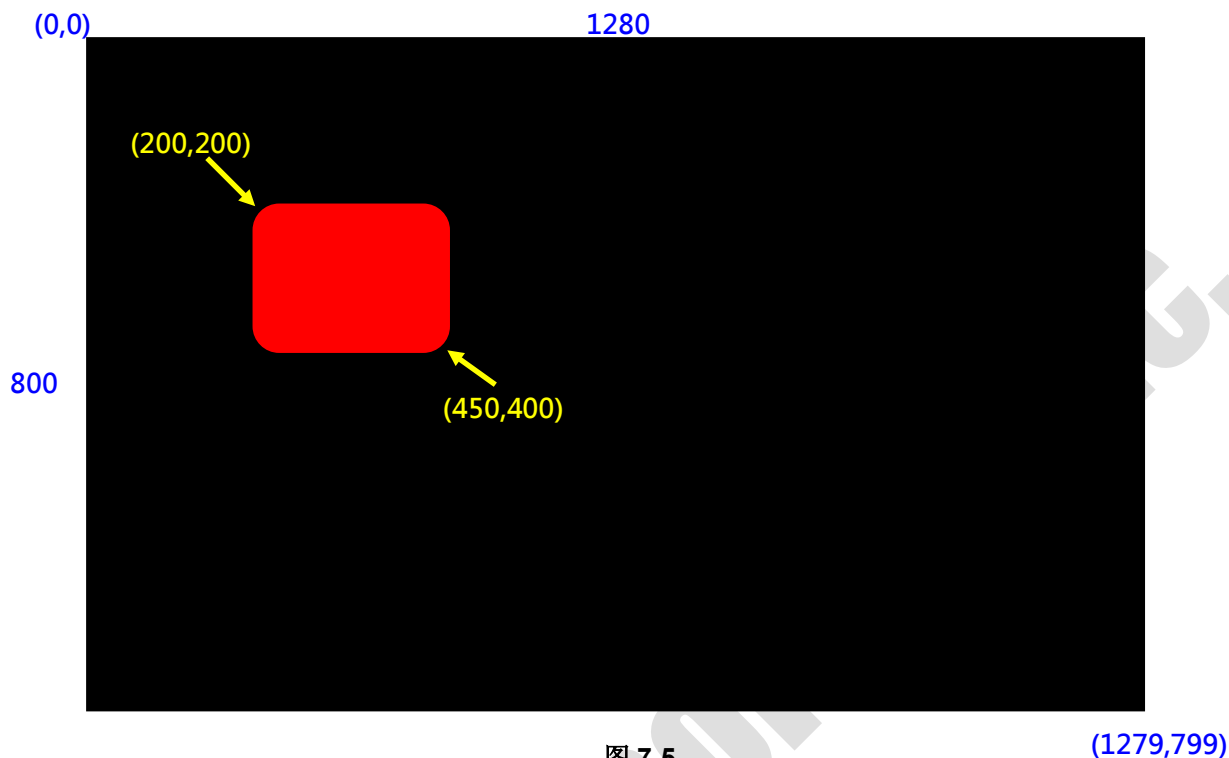


图 7-5

完成圆角方的功能后，可以在图形上面利用 **Text Function** 的功能，并设定字背景透明，将文字写在图形上，即完成简单的按键制作，若需要很多的同样按键图形，可以利用后续介绍的 **BTE Function**，将选定的按键图形，搬至不同的内存位置，达到显示许多按键的功能。

Demo Video:

YouTube

<https://www.youtube.com/watch?v=5JLdPng51fs>

youku

http://v.youku.com/v_show/id_XODk5OTE0NzQ0.html

8. 内存检视功能

8.1 Ping-pong Buffer

RA8889 内建 128Mb SDRAM，可存放处理过的数据，供 LCD 显示使用。通过 IC Function 的 Main Window 功能，切换内存显示的区域，而未显示的区域，则可以做为处理数据的 Image Buffer，利用 Main Window 定时切换显示区域与 Image Buffer，可以达到类似滚动(Scrolling)的功能。

以使用的 LCD 显示分辨率 1280x800(可视范围)为例，Canvas image width 设定为 2560，Active Window Size 为 2560x1600，色深为 16bpp，设置一个 LCD 显示分辨率 4 倍大的 Canvas image 于 SDRAM 内存中。

首先，利用之前介绍的几何图形绘图功能与 DMA 功能，将 2560x1600 内存区块清空且配置为下图所示，可分成 1、2、3、4 区，而每一区大小都是 1280x800，这种情况下，LCD Display Area 默认是显示第 1 区内存数据的画面，而其它 2、3、4 区，则是看不到的，那如果我们要让 LCD 画面分别显示 2、3、4 区呢？可以切换 Main Window Upper-Left corner XY-coordination，将其分别设定(1280,0)、(0,800)、(1280,800)，就可以分次显示每个区域。

依上述的 Function 功能，可以做很多的应用，例如：用户先显示第 1 区数据，趁空档更新第 2 区数据，等第 2 区更新完毕，将 Main Window 切至显示第 2 区，类似 Ping-pong Buffer，这种方式可以解决 MCU 速度不足，用户看到数据更新时的画面。

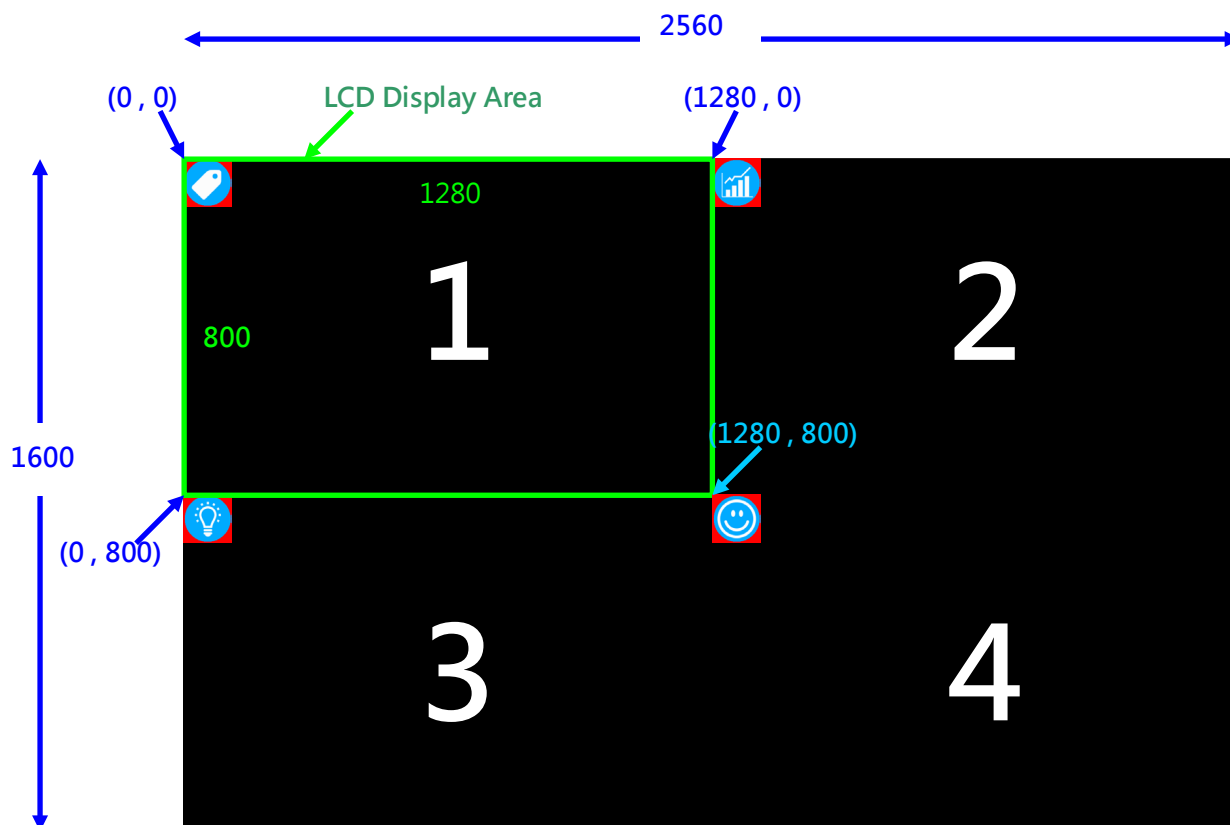
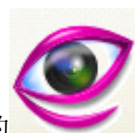
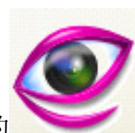


图 8-1



AP 提供简单切换 Main Window XY 坐标的功能，点选 IC Function Button 中的  按键，开启内存检视功能。

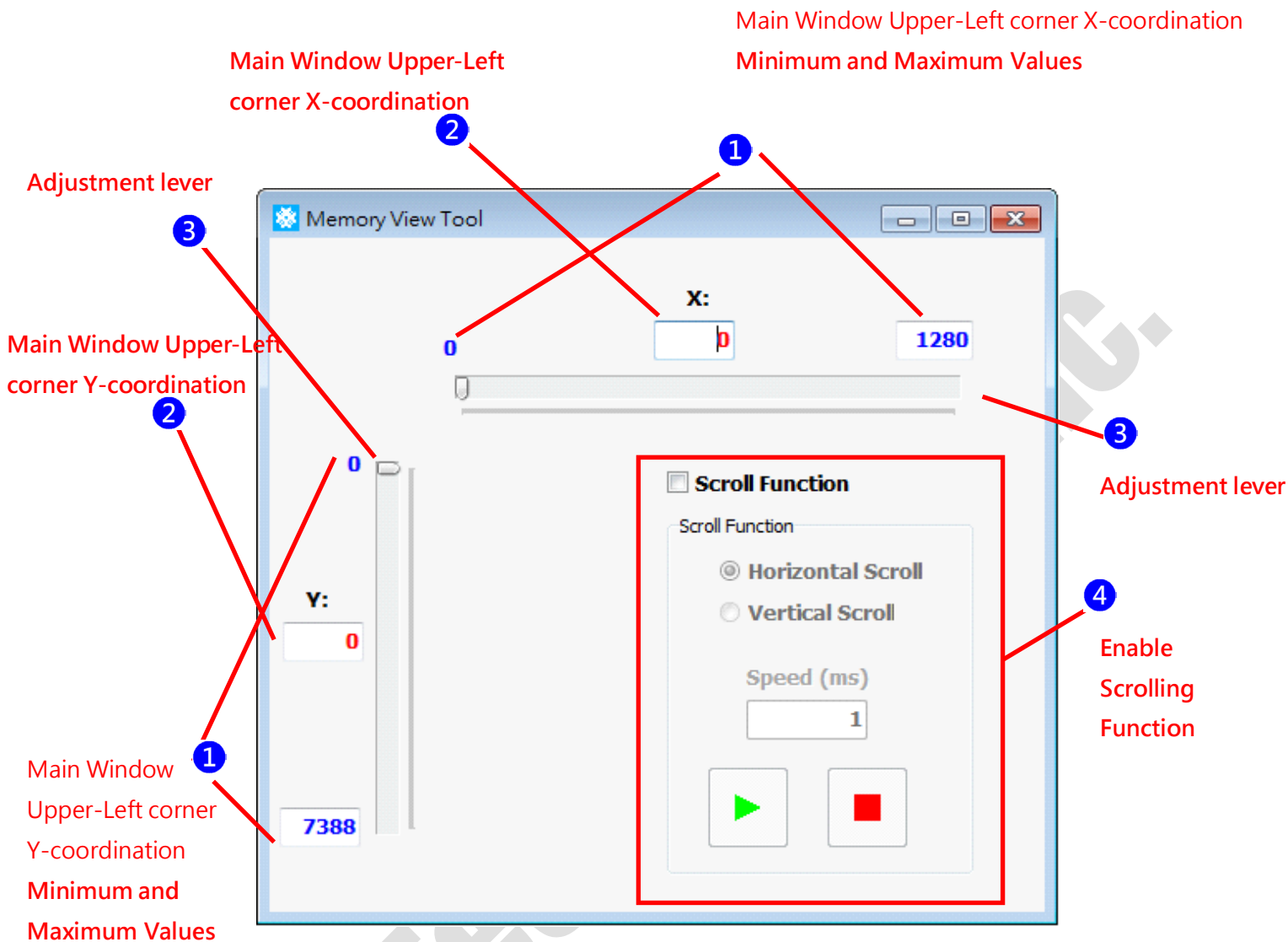


图 8-2

- ① Main Window XY 坐标可设定的最小、最大值，由 AP 自动带出，可以不用设定，其中 X 坐标最大值为 Canvas Width 减掉 LCD Display Width(例: 1280)，所以这里最大可设定的 Main Window X 坐标值为 $2560-1280=1280$ ，而 Y 坐标的可设定范围最大值，是 $8188-800=7388$ ，但必须注意的，虽然 Y 坐标可以设定到 7388 这么大，但还是必须看实际内存容量是否能达到，若无法达到，LCD 会有无法预期的画面显示，关于 Canvas Image 、Main Window 与记忆体容量的计算，请参考 IC Datasheet。
- ② 此时 Main Window X 与 Y 设定坐标值，例如 LCD 显示第 2 区，X 坐标设定 1280，Y 为 0。由于 IC 上设计的限制，X 坐标值要可以被 4 整除，Y 坐标则没有此限制。
- ③ 除了 ② 输入坐标的方式之外，还提供另一种快速设定方式，用鼠标点拉调整杆的方式设定。
- ④ Scroll Function，开启 Scrolling 功能。

显示结果示意图:

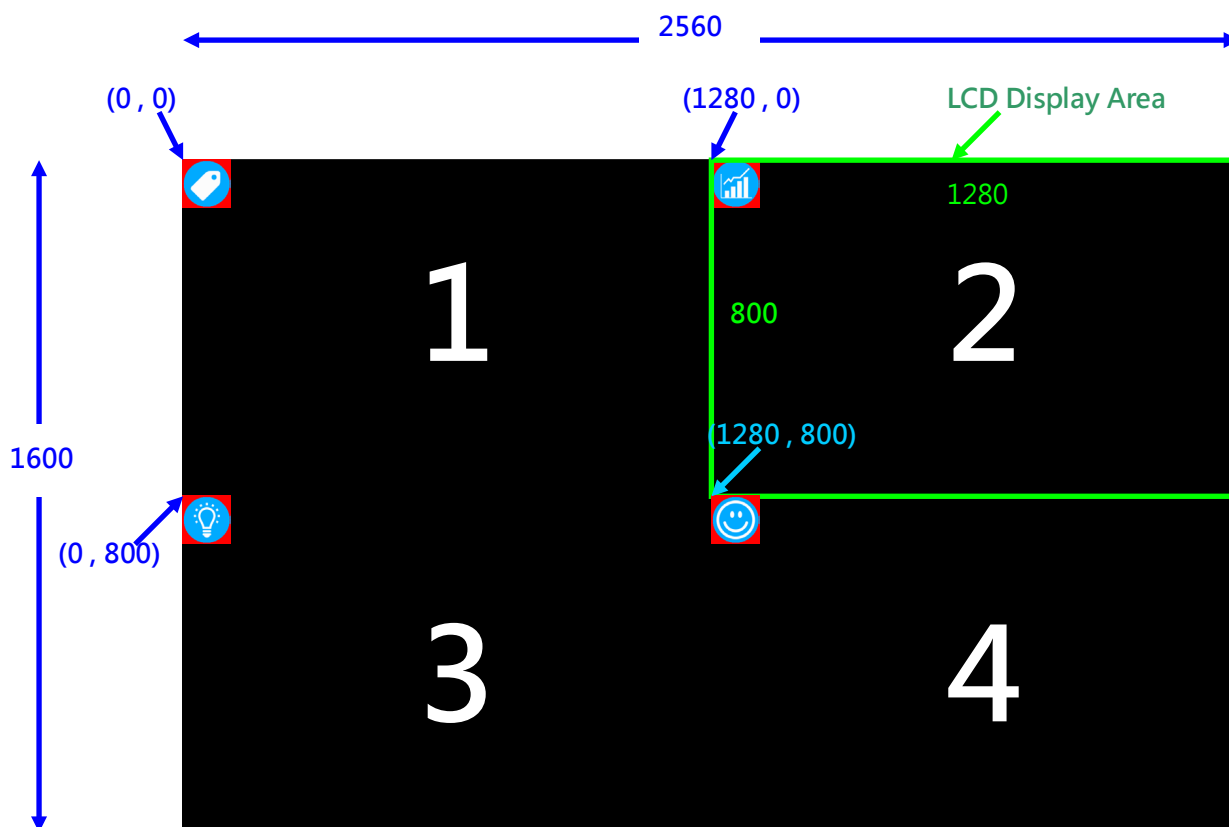


图 8-3

8.2 卷动功能

Main Window 的卷动应用。

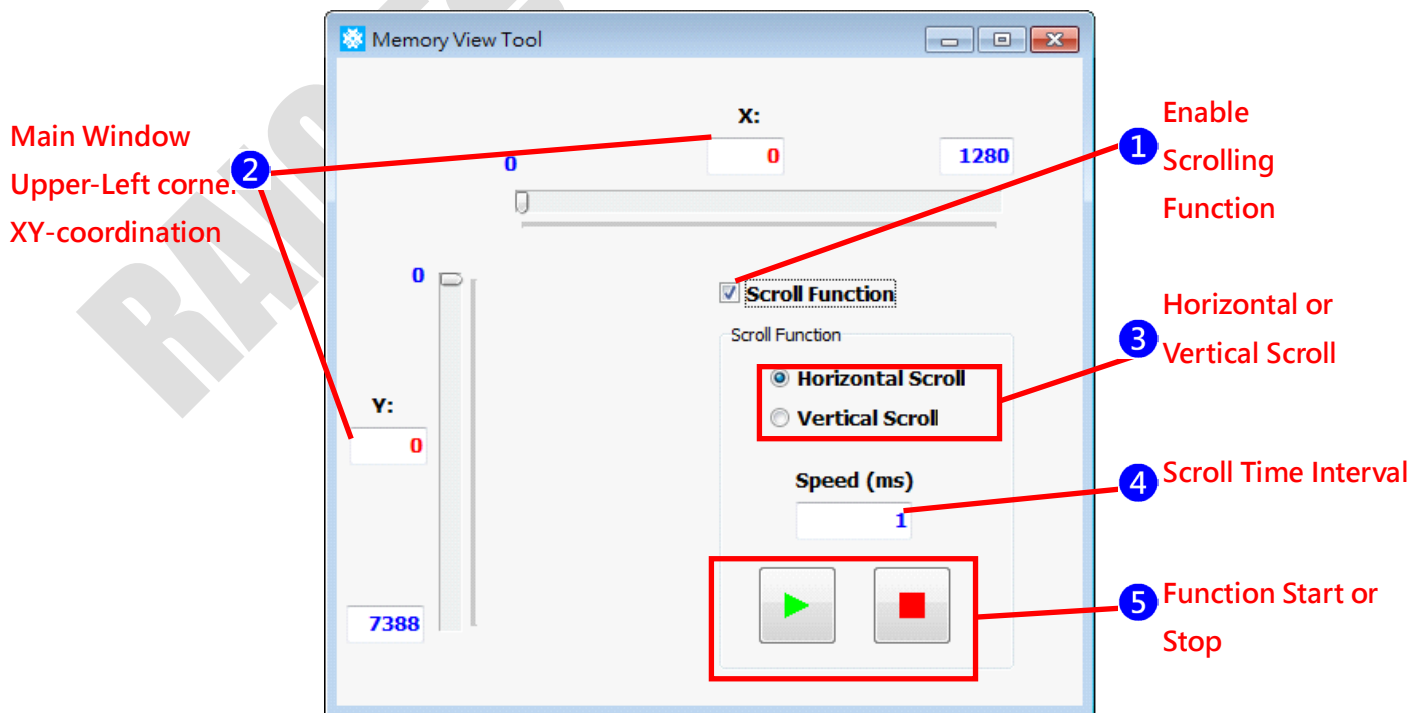


图 8-4

- ① 开启 Scrolling Function
- ② 设定卷动画面的 Main Window XY 起始坐标，卷动功能将会以此坐标卷动到 X 与 Y 的 Main Window 最大坐标值。
- ③ 设定水平卷动(Horizontal Scroll)或垂直滚动(Vertical Scroll)，水平卷动一次增加 4 个 X 坐标值(如前述，IC 设计限制)，垂直滚动一次增加 1 个 Y 坐标值。
- ④ 每次增加坐标量的间隔时间。
- ⑤ 卷动功能开始或停止。

水平卷动示意图:

水平卷动开始之前.



图 8-5

进行水平卷动

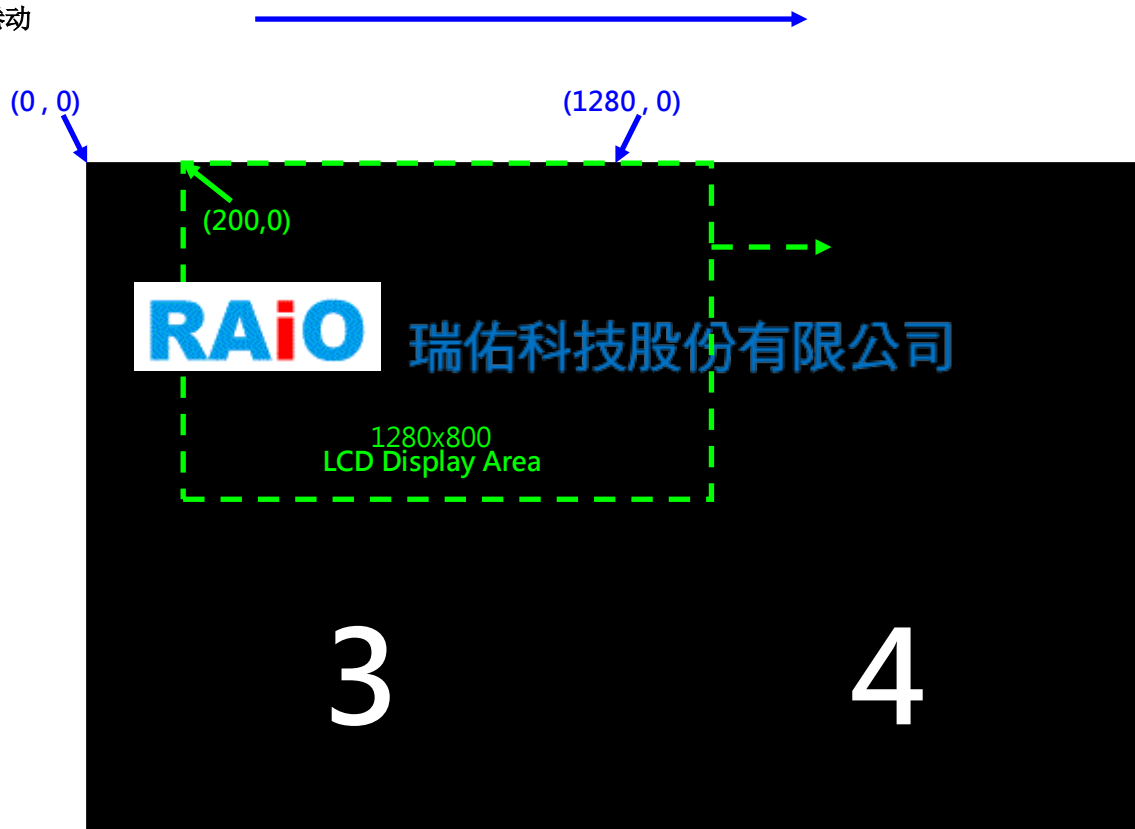


图 8-6

进行水平卷动

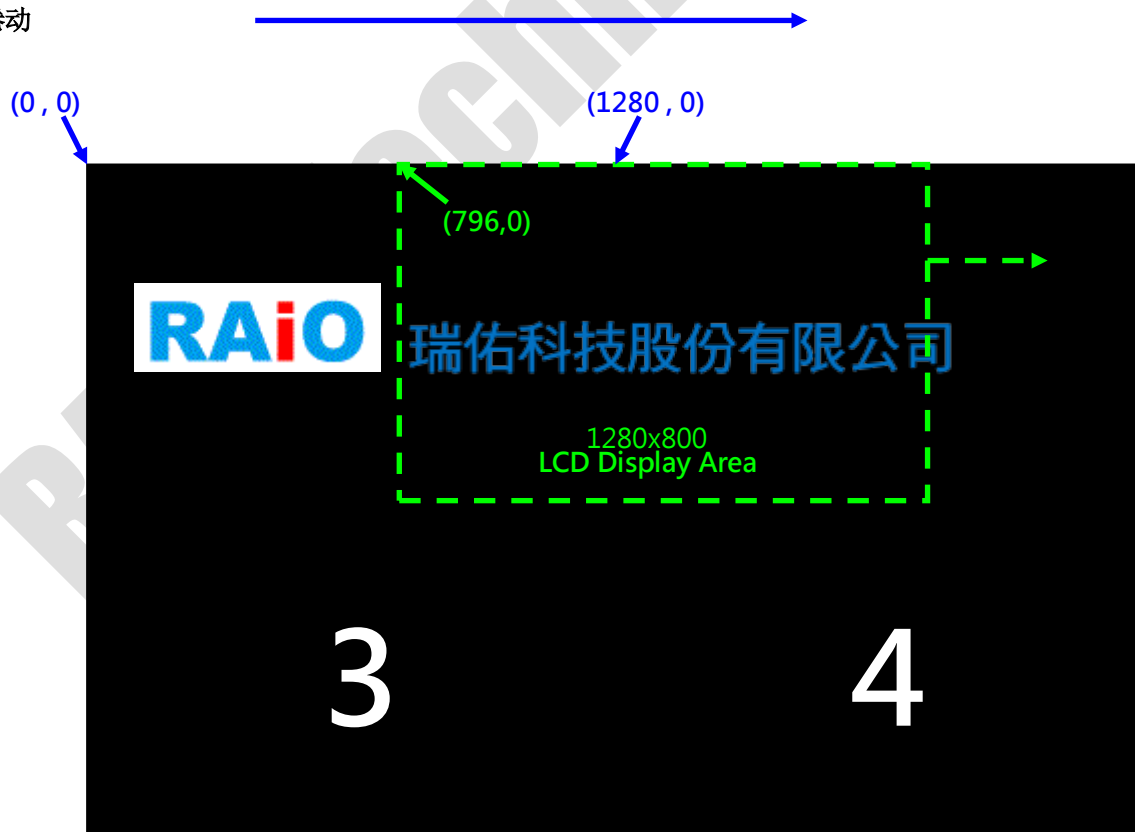


图 8-7

水平卷动结束

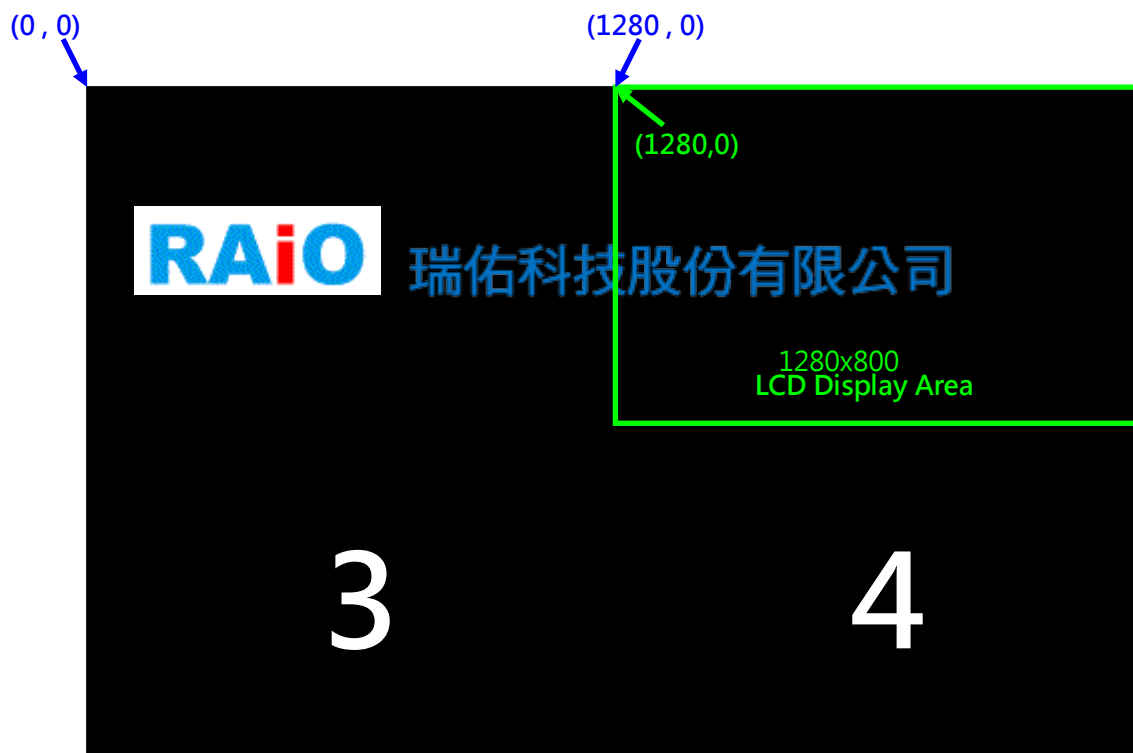


图 8-8

视觉效果有如广告跑马灯，若进行更进阶的应用方式，还可以实时更新卷动画面的资料。

Demo Video:

YouTube

<https://www.youtube.com/watch?v=bliJiKRqhd4>

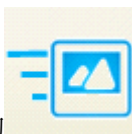
youku

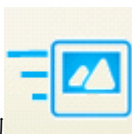
http://v.youku.com/v_show/id_XODk4NzQ0MDQw.html

9. BTE Function

9.1 Memory Copy with ROP

BTE 的详细功能介绍，可以参考 RA8889 Datasheet，AP 提供其中几个常用功能操作接口，让用户能进行简单的操作与了解。



点选 IC Function Button 中的  按键，开启 BTE Function 功能。

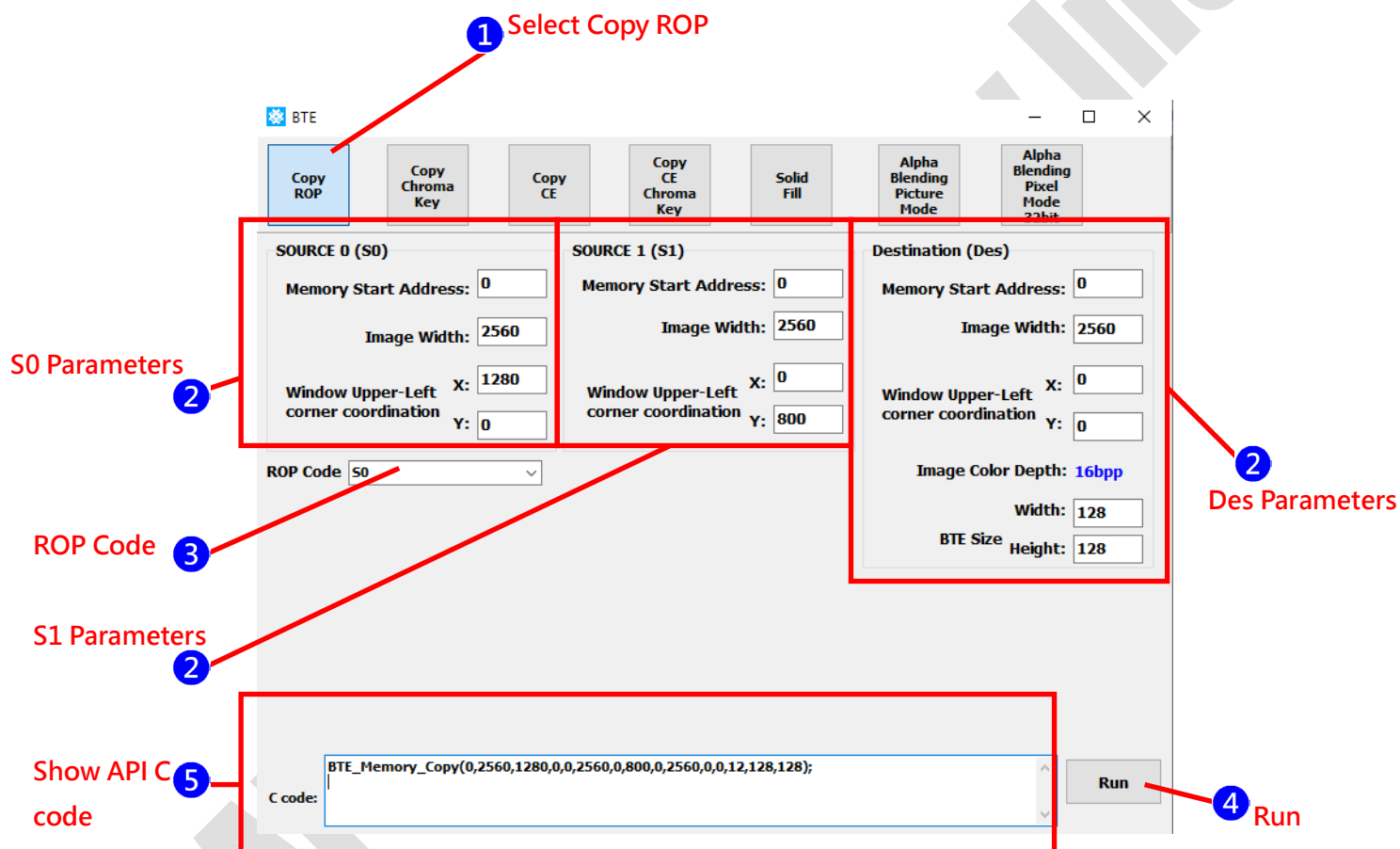


图 9-1

内存数据如下配置，第 1 区设定为 Destination；第 2 区为 S0；第 3 区 S1；第 4 区没有用到：

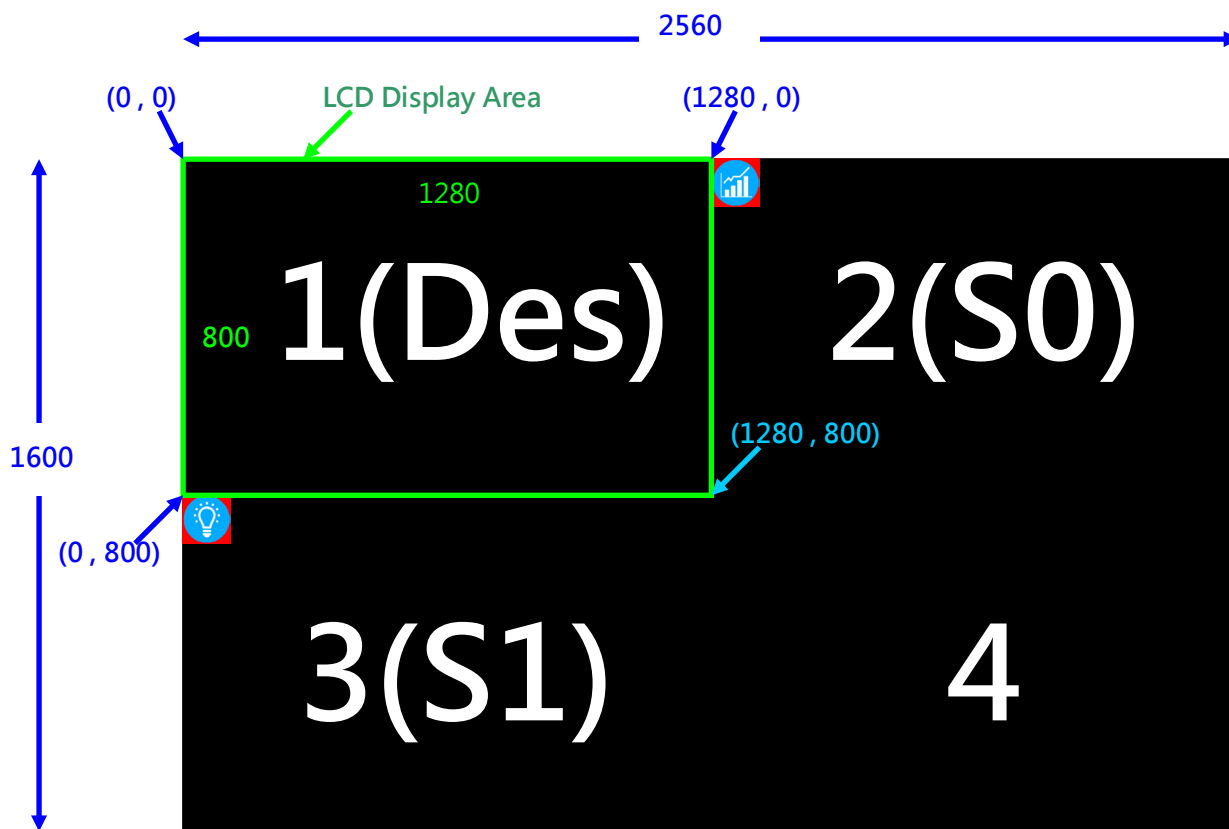


图 9-2

- 1 选择 Copy ROP (Memory Copy with ROP)功能。
- 2 设定 S0、S1、Des 相关参数，Start Address 使用预设 0，Image Width 设定与 Canvas Image Width 相同 2560，S0 XY 坐标设定为(1280,0)；S1 为(0,800)；Des 为(0,0)，BTE Size 设定为 128x128。
- 3 设定 S0 与 S1 逻辑运算功能，以 ROP=S0 为例。
- 4 执行 Copy ROP 功能
- 5 显示 Copy ROP 的 API C code

执行结果如下，当 ROP=S0，将 S0 设定的搬移数据，复制至 Des 设定的坐标区域上。

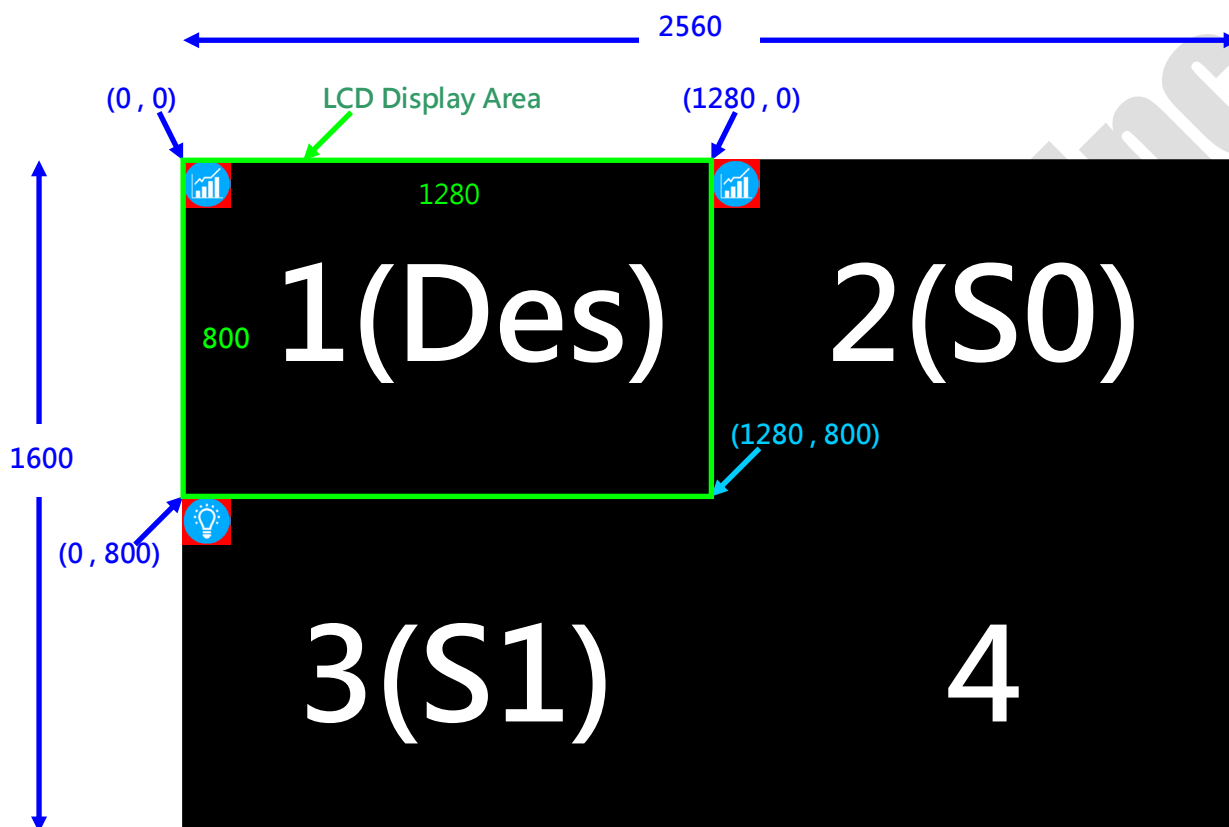


图 9-3

Demo Video:

YouTube

https://www.youtube.com/watch?v=srbvwcZ_w_U

youku

http://v.youku.com/v_show/id_XODk4Njc1NzA4.html

9.2 Memory Copy with Chroma Keying

与 Memory Copy with ROP 功能类似，但没有 ROP 功能，能设定 chroma keying 的颜色，将 S0 的某一个颜色滤除。

内存数据如下配置，第 1 区设定为 Destination；第 2 区为 S0；第 3 区 S1；第 4 区没有用到：

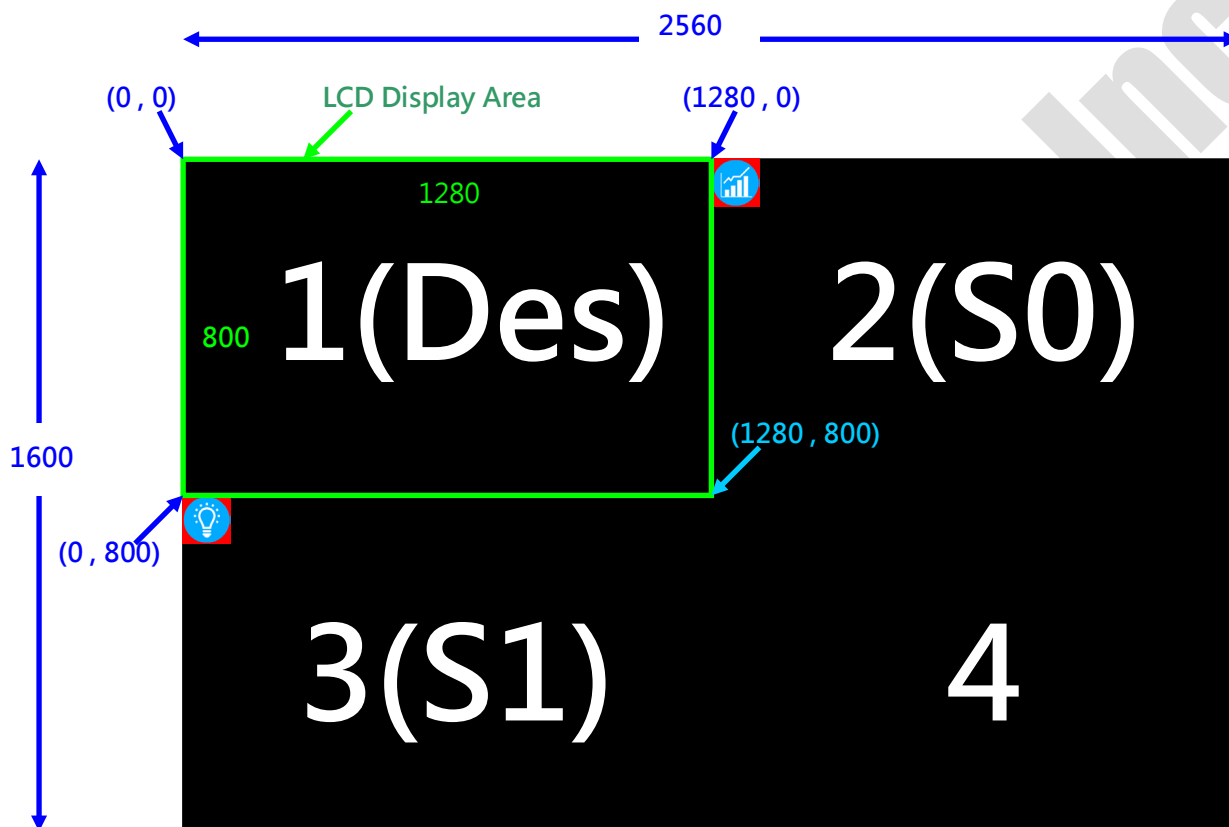


图 9-4

Select Copy
Chroma Key

1

2 Set S0、Des
Parameters

3 Set Chrome
Keying color

4 Run

图 9-5

- 1 选择 Copy Chroma key (Memory Copy with Chroma Keying)功能
- 2 设定 S0 与 Des 相关参数，此功能没有用到 S1，故不须设定相关参数。
- 3 设定 S0 数据要滤除那个颜色。
- 4 执行功能。

执行结果如下，可以发现 S0 的图，经过处理后，背景红色已经被滤除，且数据复制到 Des 区域内。

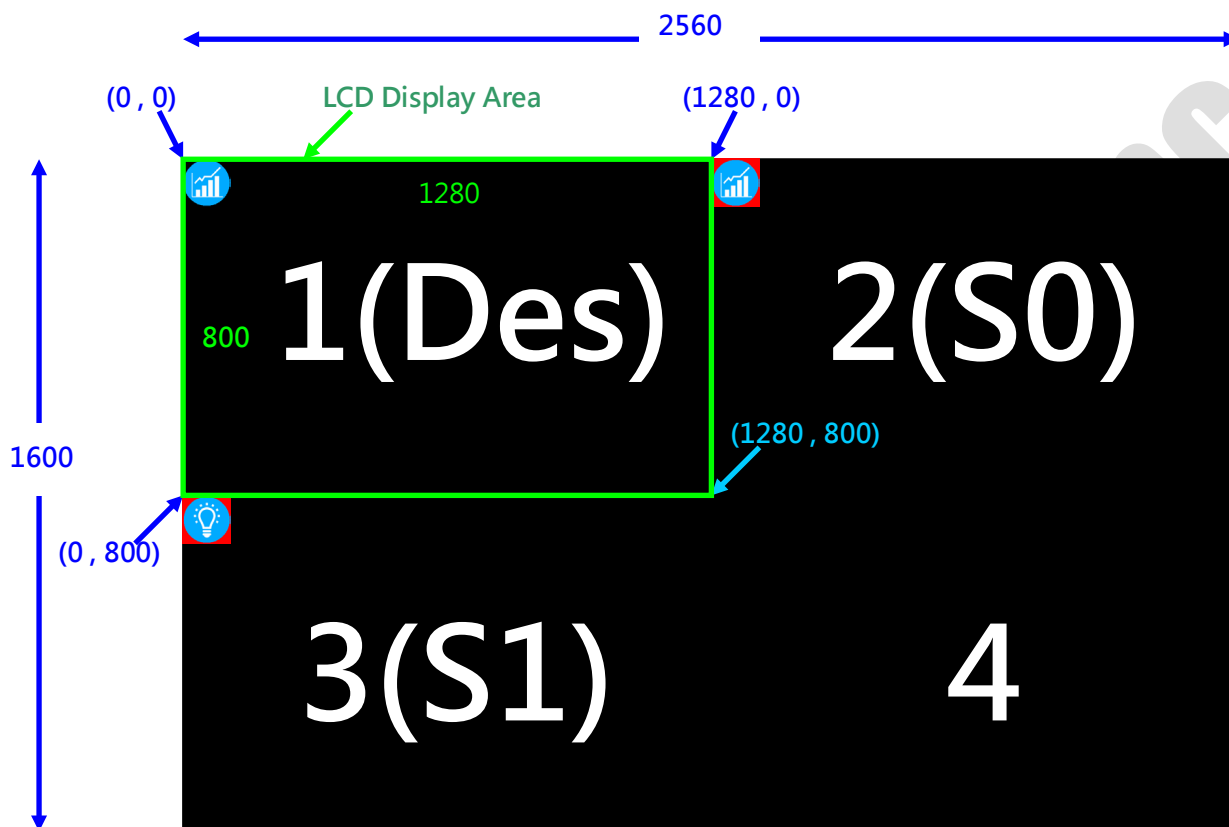


图 9-6

Demo Video:

YouTube

<http://youtu.be/hmidywKHm0Q>

youku

http://v.youku.com/v_show/id_XOTEwODM0NzQ4.html

9.3 Memory Copy with Color Expansion

Memory Copy with Color Expansion 的功能，是将颜色深度 1bpp 的黑白图，转为所设定的前背景两种颜色的彩色图。

先利用 MPU Write Tool 的功能，将颜色深度的图形数据(1024x768)写入到 S0 区域内，由于是 1bpp 的数据，显示出来的画面，感觉宽度被压缩过，这是正常的。

1bpp 原始黑白图:

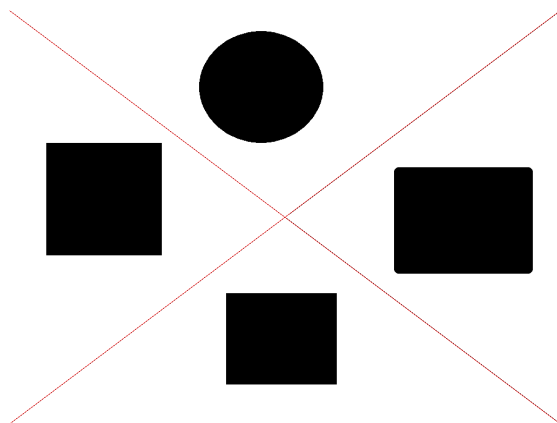


图 9-7

内存数据如下配置，第 1 区设定为 Destination；第 2 区为 S0；第 3 区 S1；第 4 区没有用到：

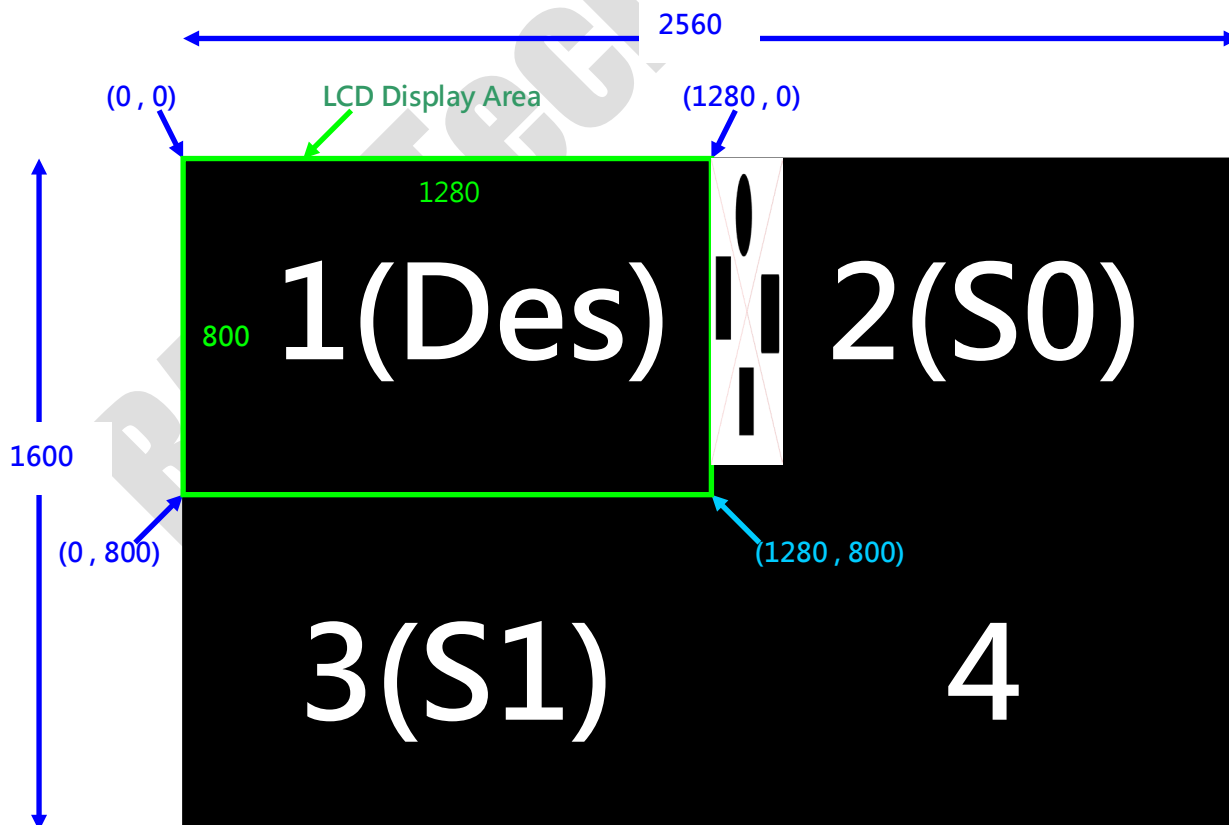


图 9-8

Select
Copy CE

1

2

3

4

Set S0、Des parameters

Set Foreground and Background color

Run

图 9-9

- 1 选择 Copy CE(Memory Copy with Color Expansion)功能
- 2 输入 S0、Des 相关参数
- 3 设定前背景色，单色图的"1"资料会转为前景色，"0"资料转为背景色。
- 4 执行功能。

执行后画面显示结果:

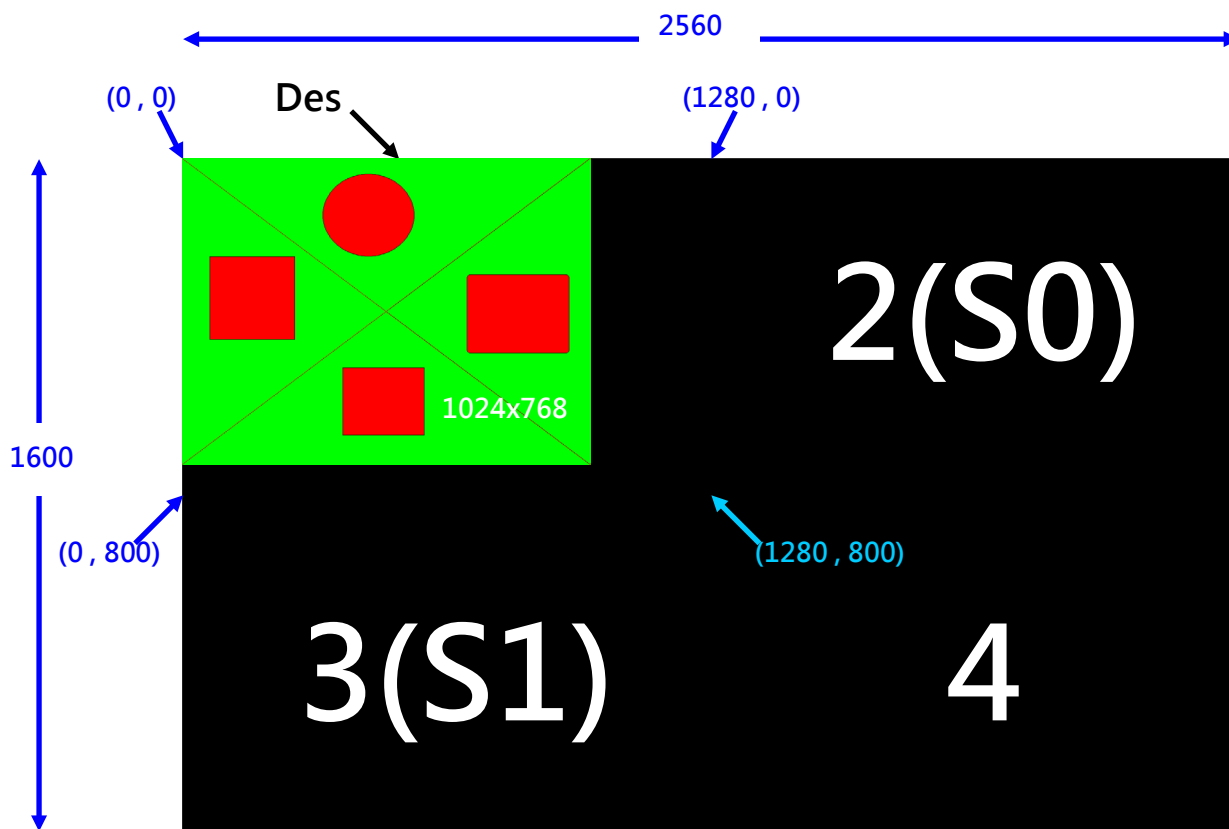


图 9-10

Demo Video:

YouTube

<https://www.youtube.com/watch?v=MRV7CrANIA0>

youku

http://v.youku.com/v_show/id_XODk4NjYwOTI0.html

9.4 Memory Copy Color with Expansion and Chroma Keying

此功能与 Memory Copy with Color Expansion 功能类似，差别在数据“0”并没有转为背景色，而是维持原先 Des 的资料，所以少了设定背景色的设定，其余相同，再此不再列出其执行步骤，而执行结果如下：

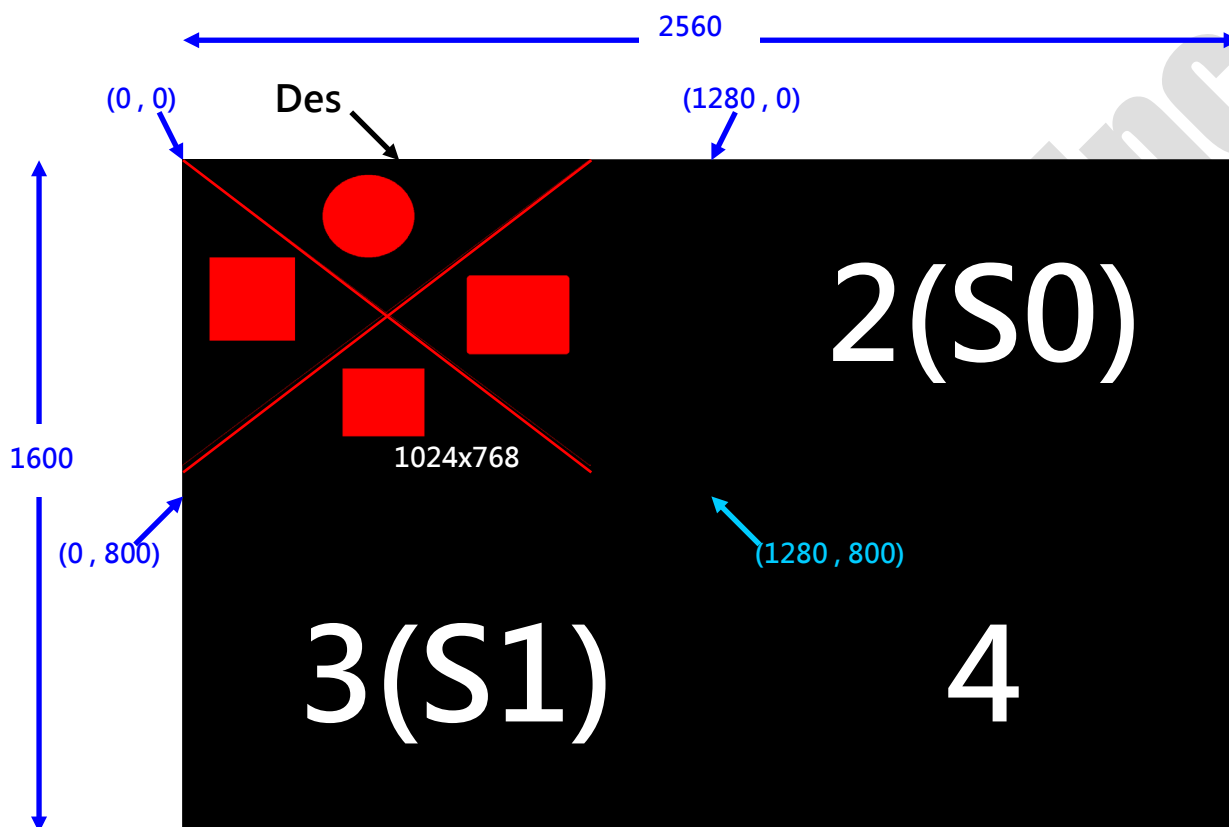


图 9-11

Demo Video:

YouTube

<http://youtu.be/cbwKnBfinPQ>

youku

http://v.youku.com/v_show/id_XOTEwNDYxODMy.html

9.5 Solid Fill

Solid Fill 的功能是将选定的 Des 区块，将它的数据写为所设定的颜色，功能与几何图形绘画功能的矩形功能类似。

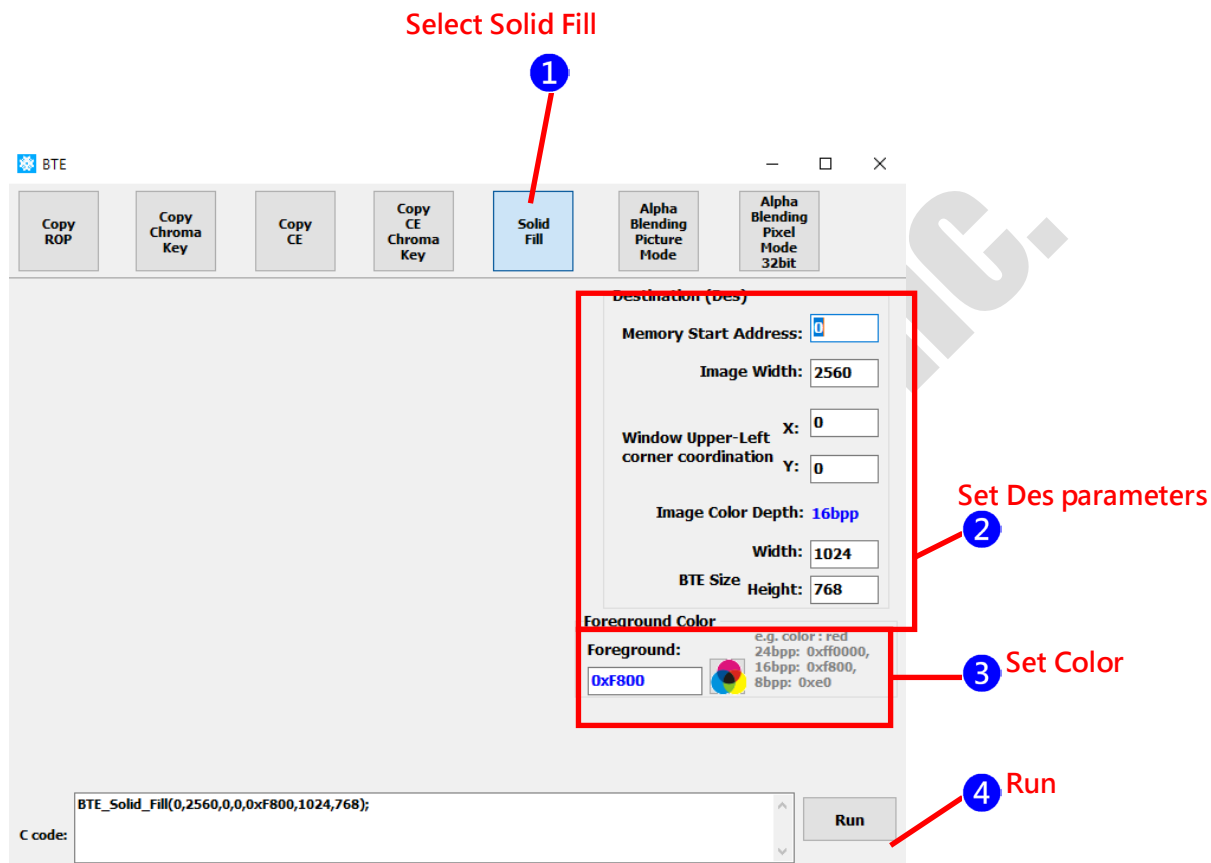


图 9-12

- ① 选择 Solid Fill 功能。
- ② 设定 Des 要填满颜色的区域与大小。
- ③ 设定要填满的颜色。
- ④ 执行功能。

执行后画面显示结果:

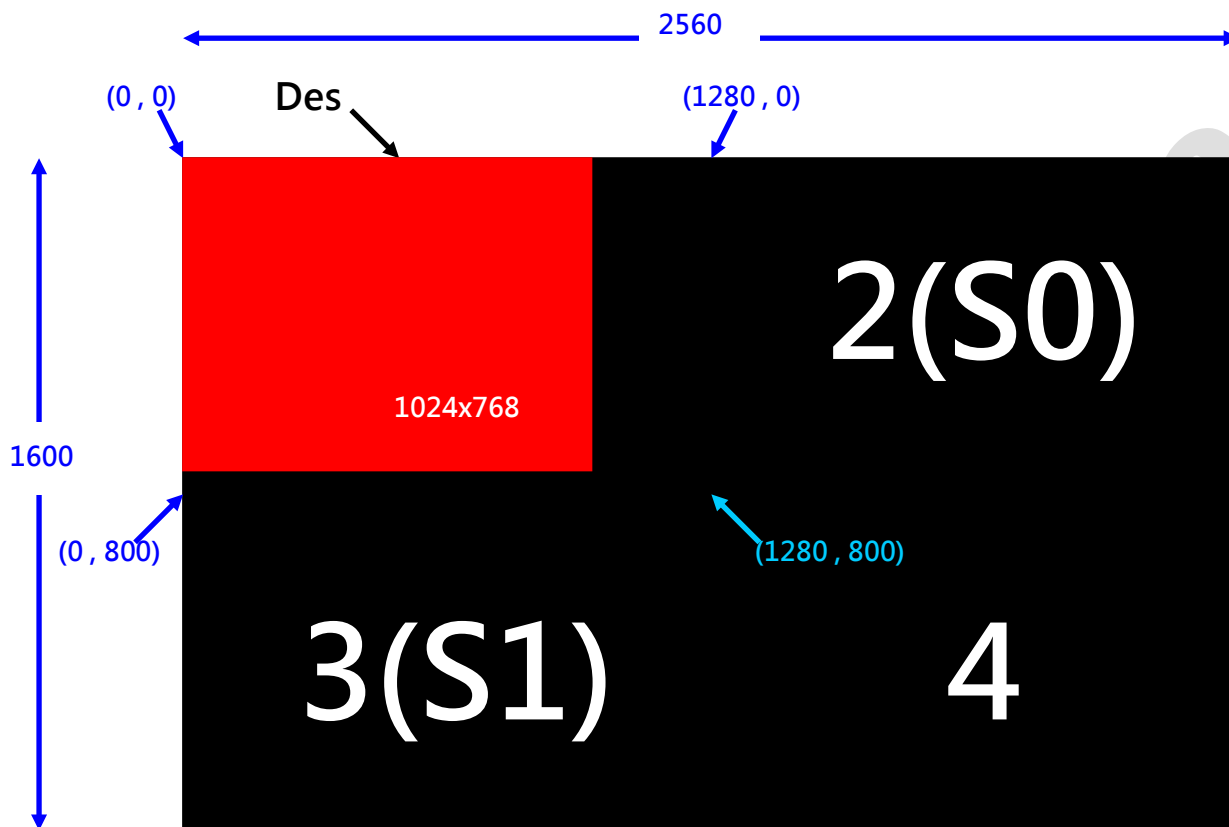


图 9-13

Demo Video:

YouTube

<https://www.youtube.com/watch?v=bbFBLHCvoUE>

youku

http://v.youku.com/v_show/id_XODk4NjkwNjU2.html

9.6 Alpha Blending in Picture Mode

此功能是将 S0 与 S1 的数据做 Alpha Blending 运算，再写到 Des 中，功能类似 Memory Copy with ROP，只是 ROP 功能改为设定 Alpha Level，此功能所选定区域内的数据，其每一个 Pixel 的 Alpha Level 处理方式都是相同的，此为 Picture Mode，RA8889 有支援 16/32bit 的 Pixel Mode，在此并没有列出，RAiO 的 Image Tool 有支持此种格式的图形转换，若有此功能的需求，请联络

RAiO 。

内存数据如下配置，第 1 区设定为 Destination；第 2 区为 S0；第 3 区 S1；第 4 区没有用到：

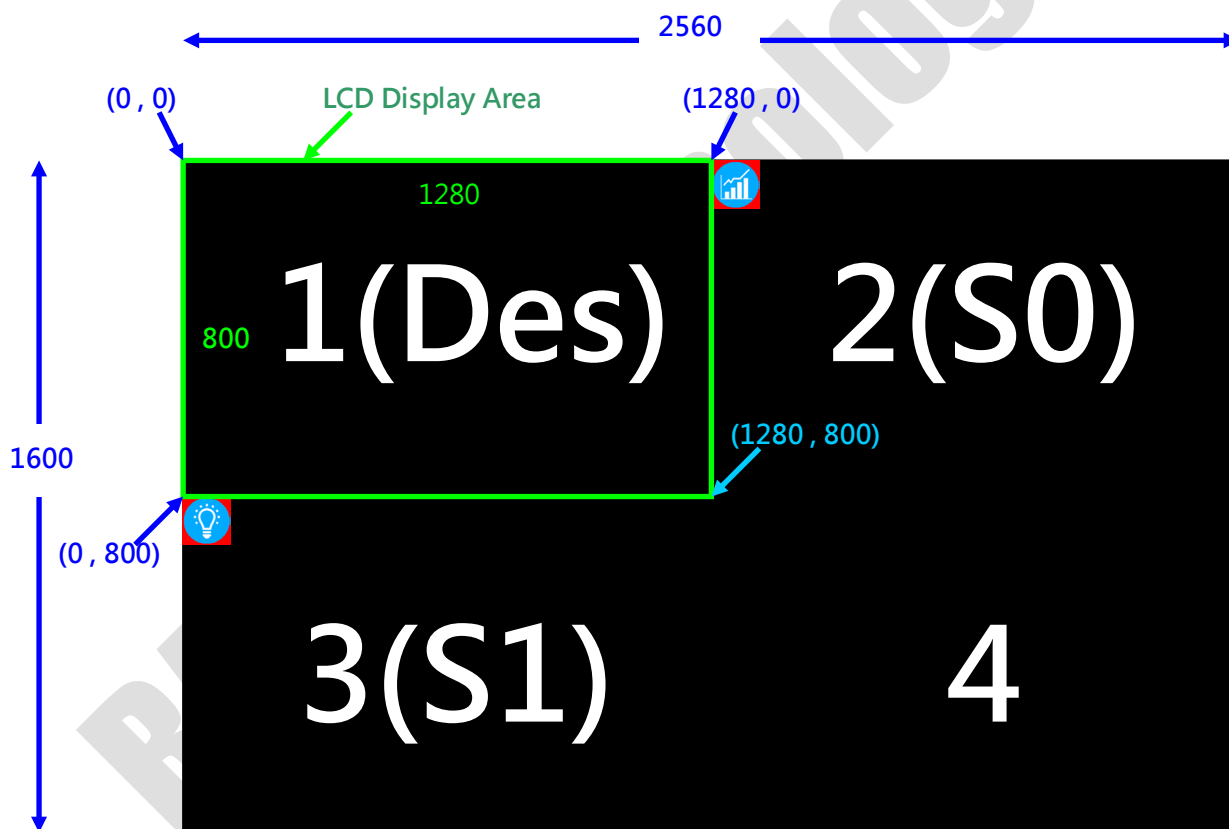
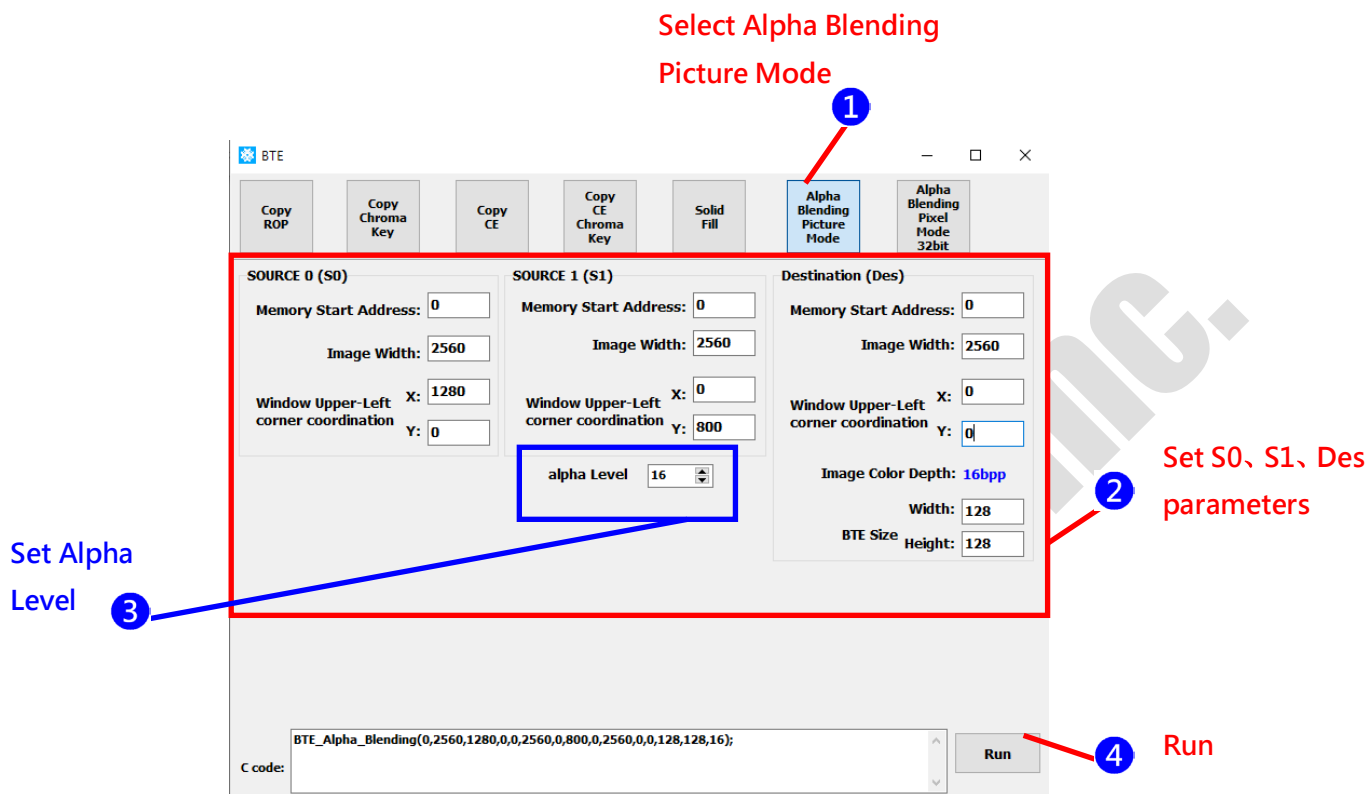


图 9-14


图 9-15

- ① 选择 Alpha Blending Picture Mode(Alpha Blending in Picture Mode)功能。
- ② 设定 S0、S1、Des 相关参数。
- ③ 设定 Alpha Level (0~32)，设定 16，为一半的透明效果。
- ④ 执行功能。

执行后画面显示结果:

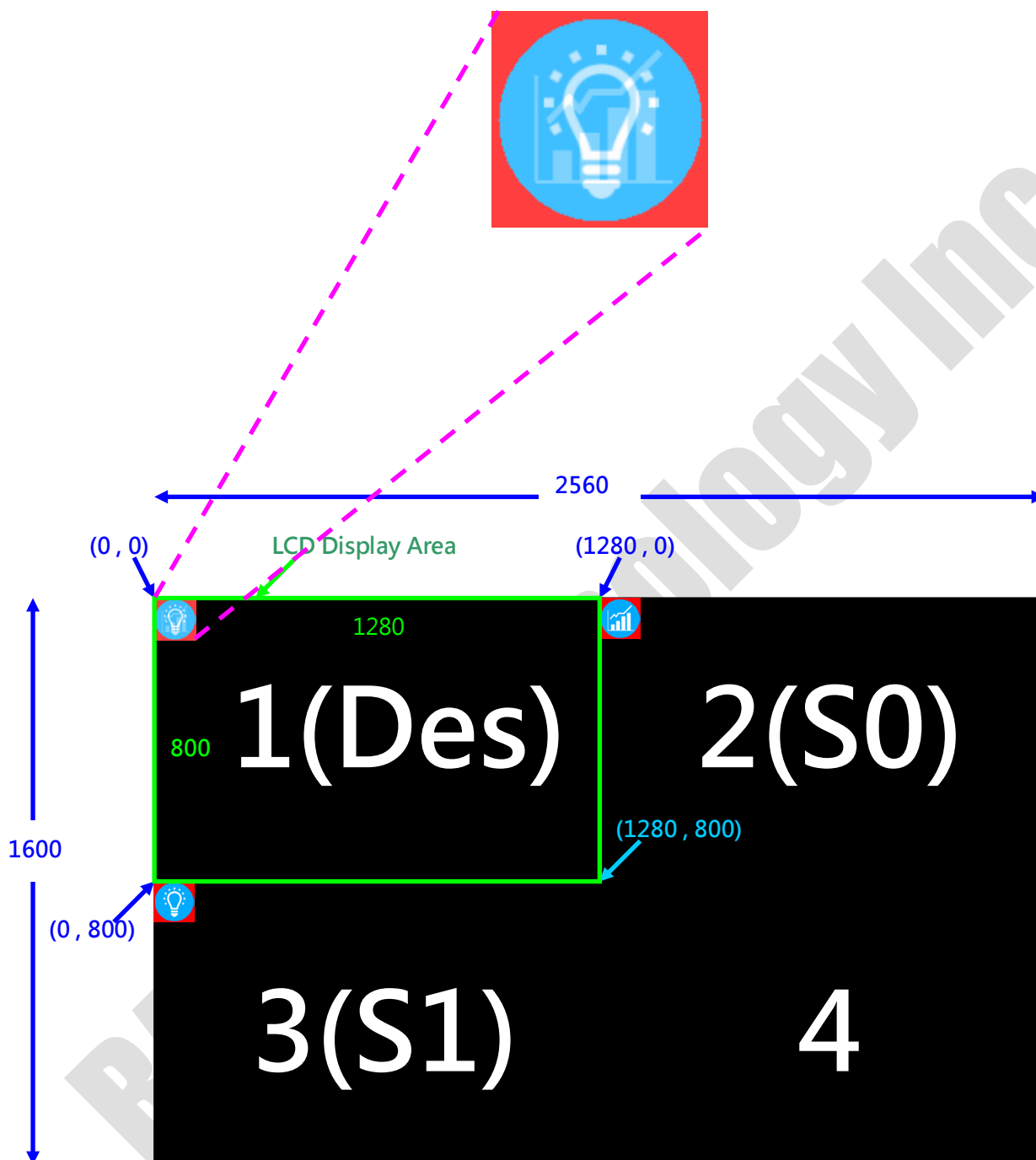


图 9-16

Demo Video:

YouTube

<https://www.youtube.com/watch?v=GG05FbaeARg>

youku

http://v.youku.com/v_show/id_XOTQwMTI2OTMy.html

9.7 Alpha Blending in 32-bit (ARGB) Pixel Mode (Color Depth:24bpp)

Pixel Mode 在 BTE 处理区域内每个像素具有不同的 alpha 透明参数值，这个透明的参数值纪录在每个像素本身的高位中。

色深 24bpp 下，内存数据如下配置，第 1 区设定为 Destination 与 S0；第 3 区为 S1；第 2、4 区没有用到：

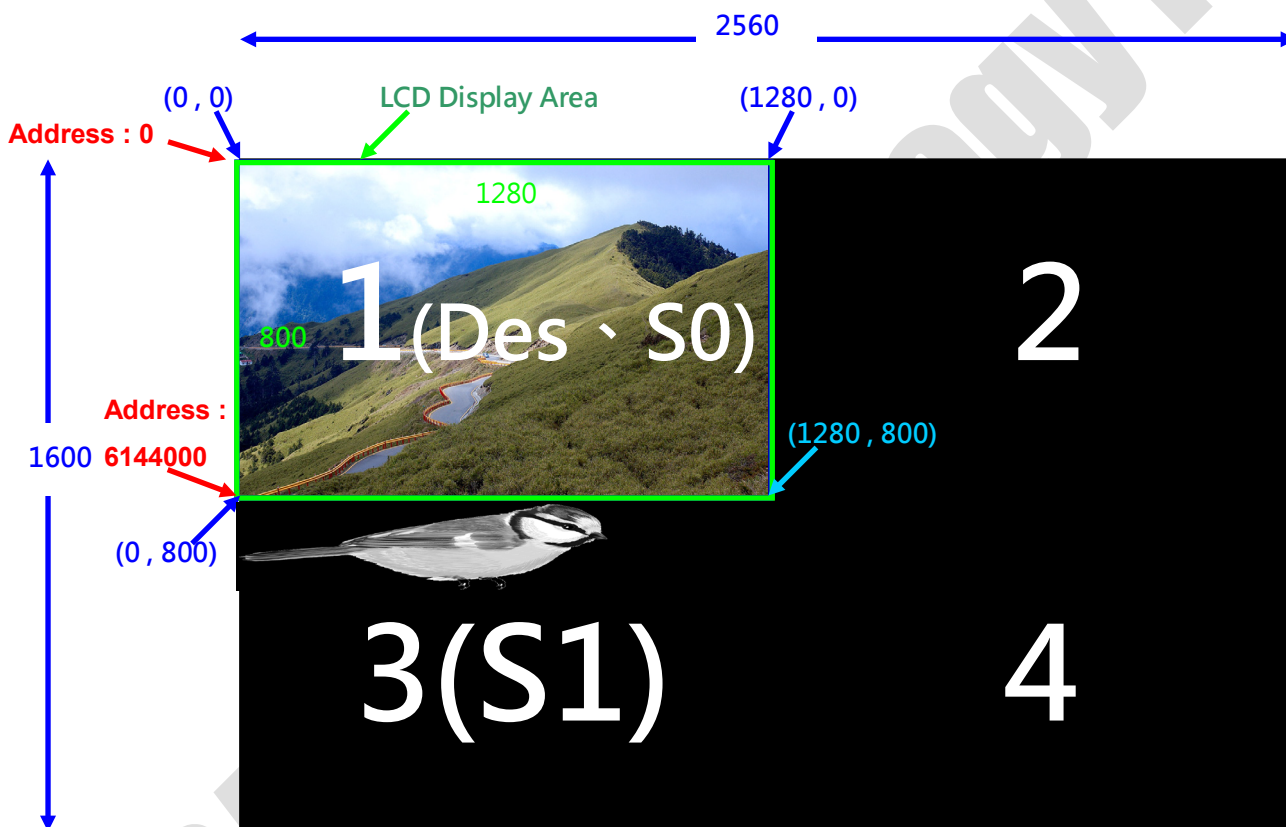


图 9-17

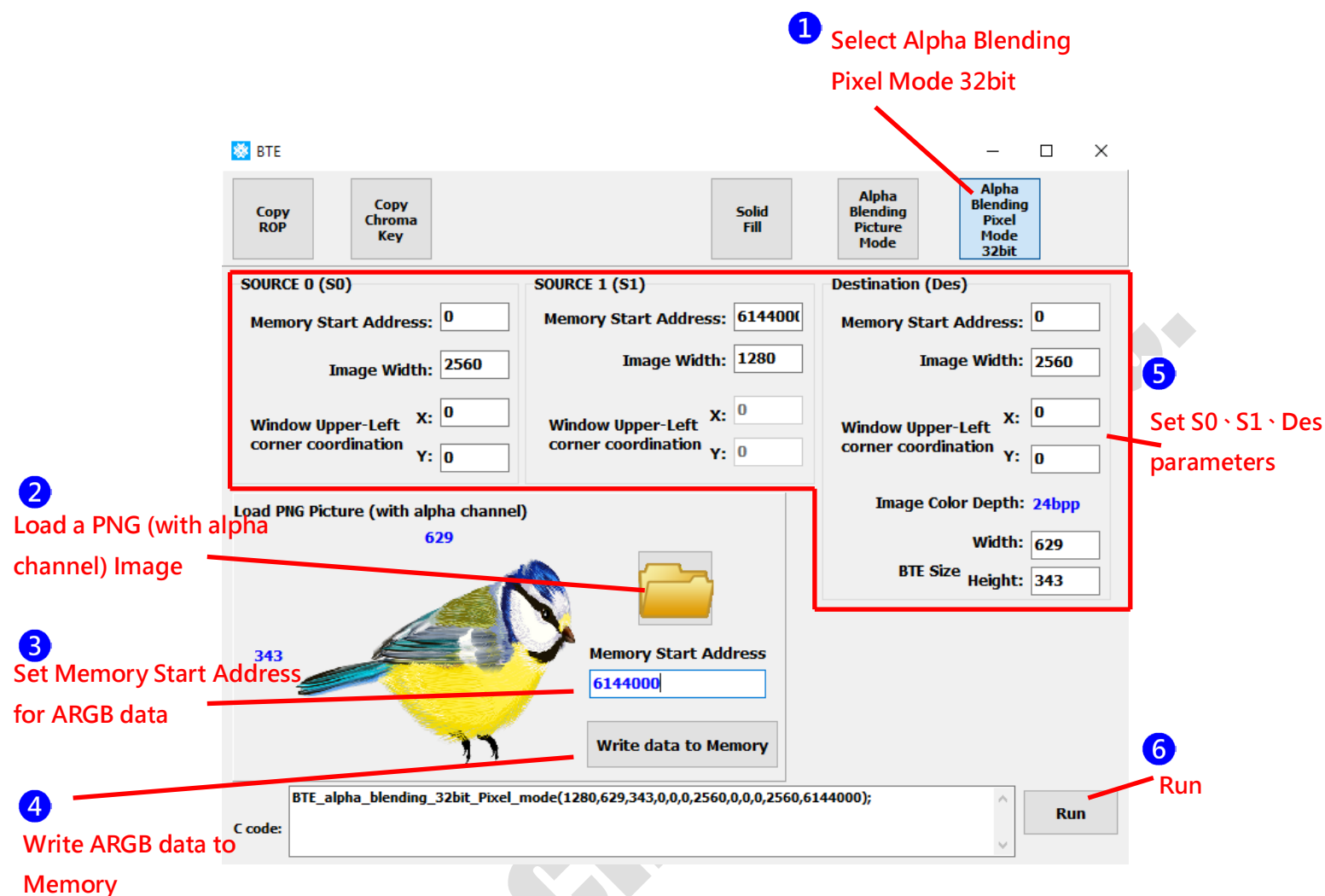


图 9-18

- 1 选择 Alpha Blending Pixel Mode 32bit 功能。
- 2 加载一张具有透明度 PNG 格式图片。
- 3 设定存放 ARGB 数据的内存起始地址， $2560 \times 3 \times 800 = 6144000$
- 4 执行 ARGB 数据写入内存。
- 5 设定 S0、S1、Destination 相关参数
- 6 执行 Alpha Blending Pixel Mode 32bit 功能

执行后画面显示结果:

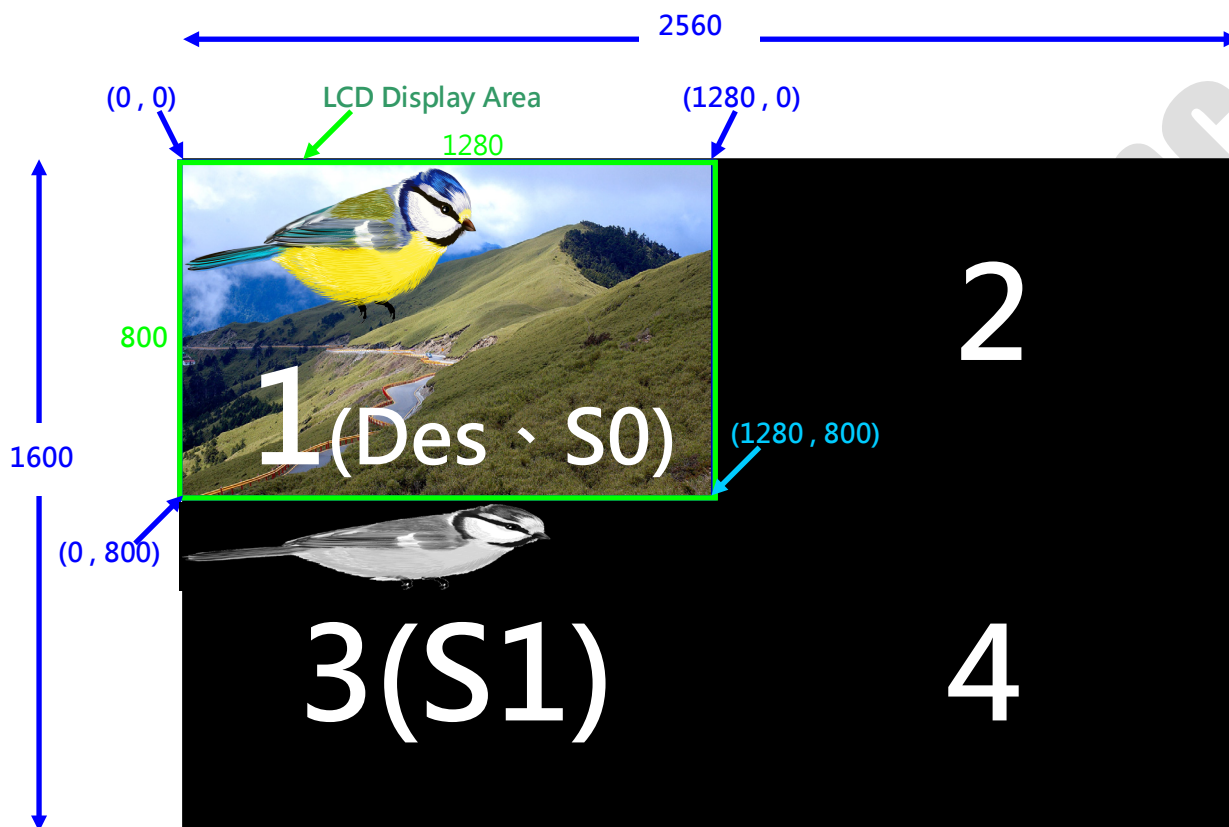


图 9-19

10. PIP Function

RA8889 可以同时显示两组 PIP 窗口于显示窗口上，PIP 窗口不支持透明度的功能，用户可以开启 PIP 功能，选择将某一处内存存放的数据，重迭显示于原来的显示画面上，且不会影响显示画面的原始数据。若 PIP1 与 PIP2 开启，且两者窗口有重迭的情况，PIP1 窗口会在 PIP2 窗口上。



点选 IC Function Button 中的  按键，开启 PIP 功能。

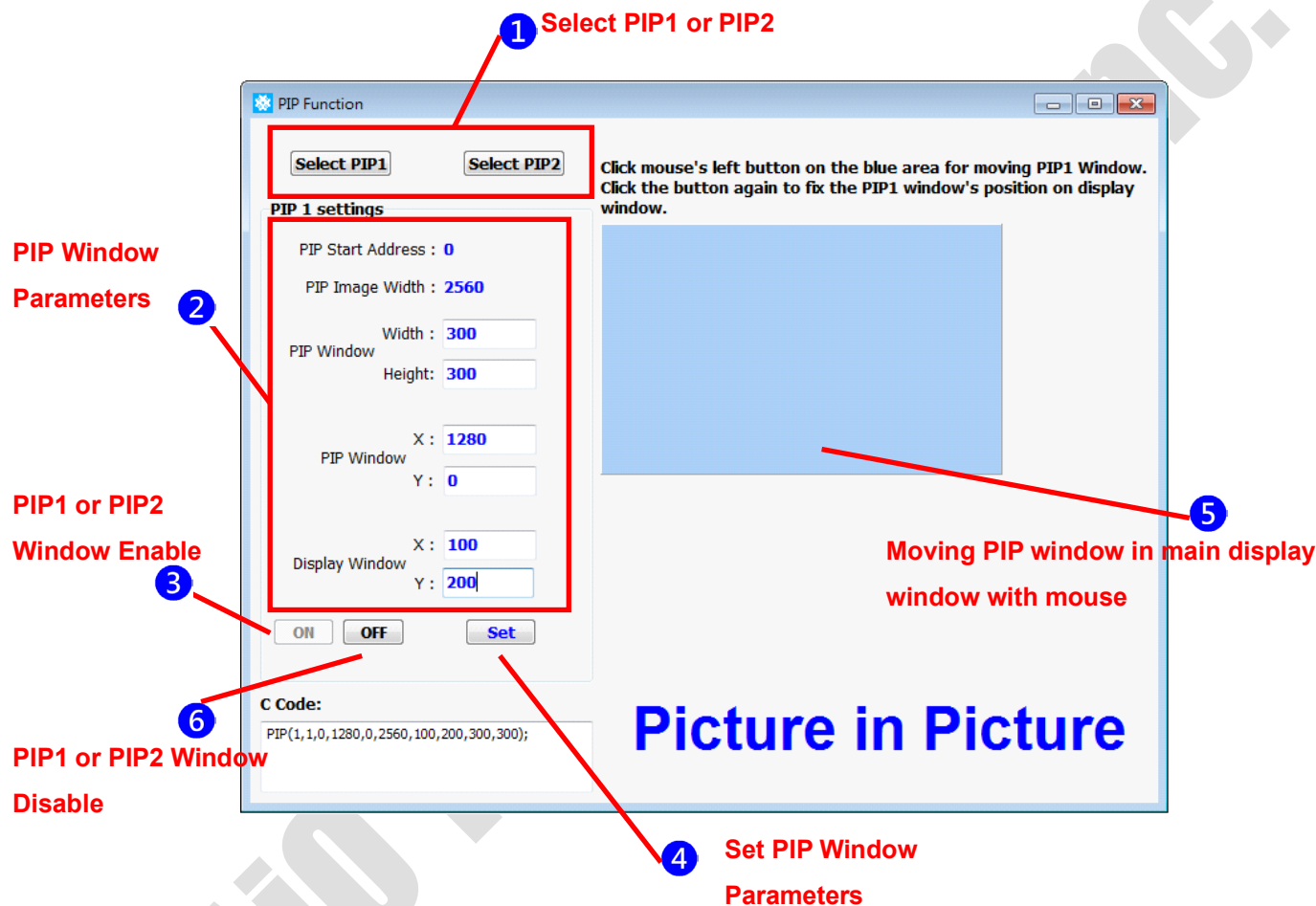


图 10-1

- 1 点击 “Select PIP1” 或 “Select PIP2” 按键，来选择 PIP1 或 PIP2 窗口。
- 2 设定 PIP 窗口相关参数。以此范例说明，PIP1 窗口的数据坐标为(1280, 0)，PIP1 窗口大小为 300x300，设定 PIP1 窗口于显示窗口(100,200)坐标上。
- 3 点击“ON” 按键来开启所选择的 PIP 窗口。
- 4 若有需要而修改 PIP 窗口的相关参数，于修改完相关参数后，点击“Set” 按键来执行修改参数后的 PIP 窗口。
- 5 除了步骤 4 的方式之外，我们还可以通过移动鼠标来设定 PIP 窗口于显示窗口的位置。首先，在蓝色区域点击鼠标左键，鼠标游标由箭头形状  变为十字形状 ，此时，在蓝色区域内移动鼠标，在显示窗口的 PIP 窗口也会随着鼠标移动位置，若再点击鼠标左键，PIP 窗口将会定位于显示窗口上。

- 6 欲开启 PIP2 窗口，请重复步骤 1 至 5。以此范例说明，PIP2 窗口的数据坐标为(1280, 300)，PIP2 窗口大小为 320x240，设定 PIP2 窗口于显示窗口(360,460)坐标上。
- 7 点击“OFF”按键可以关闭所选择的 PIP 窗口。

执行后画面显示结果:

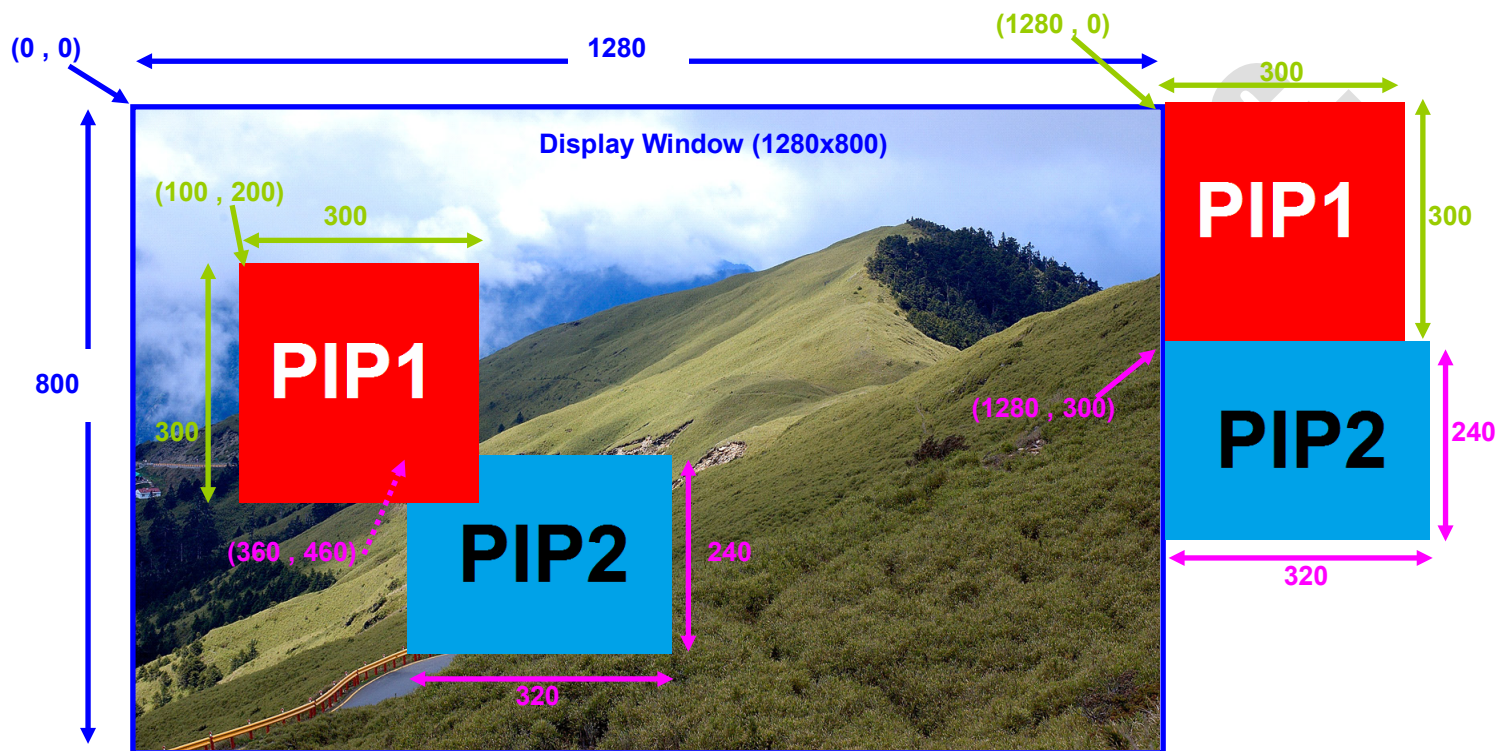


图 10-2

Demo Video:

YouTube

<https://www.youtube.com/watch?v=psNso5ZkLrY>

youku

http://v.youku.com/v_show/id_XODk4NzM2ODUy.html

11. JPEG Function

RA8889 支持宽度为8的倍数, 且为标准型(Baseline) JPEG 图档。将此类的 JPEG 图档烧录至 SPI Flash 中, 点选 IC Function



Button 中的 按键, 开启 JPEG 功能。

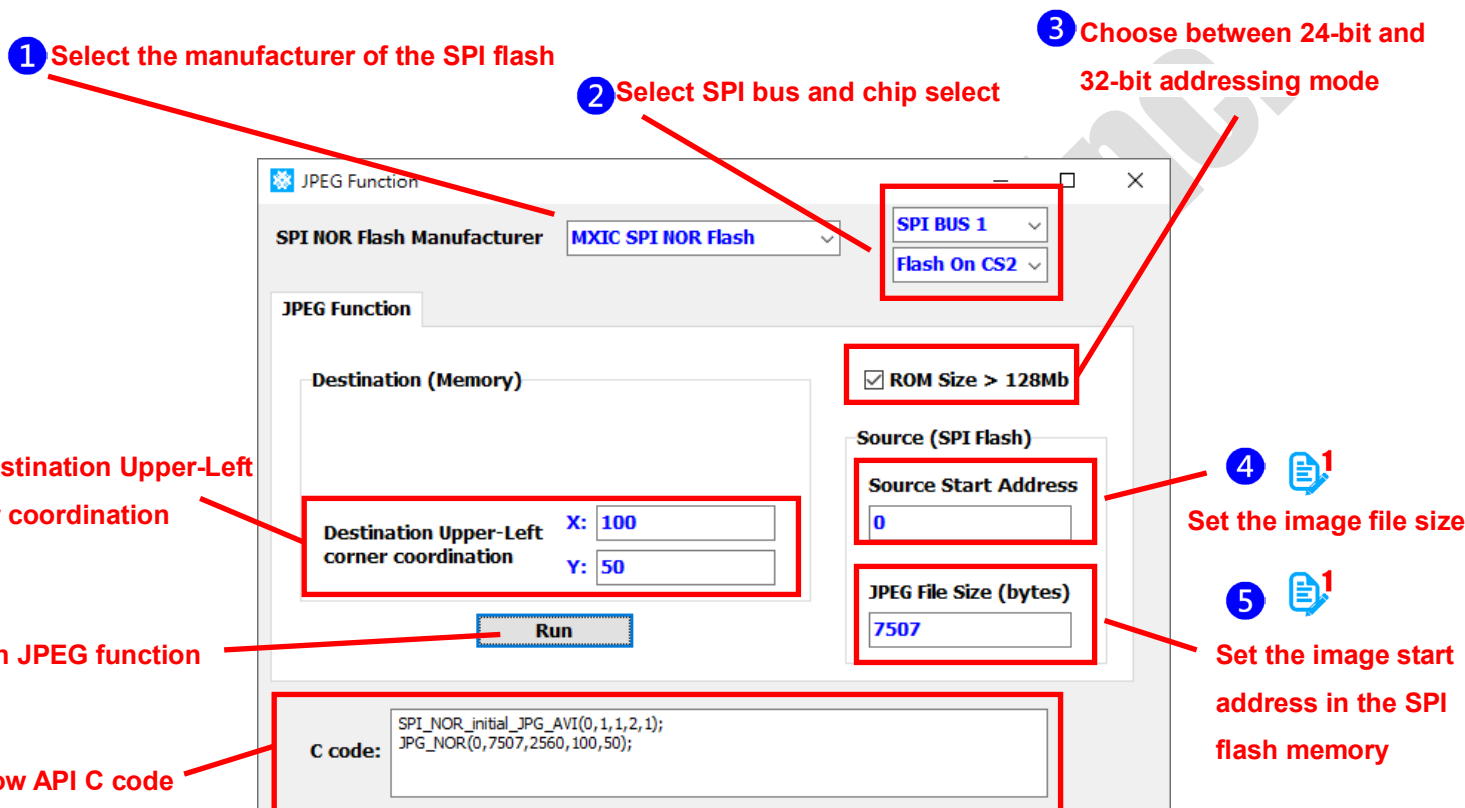


图 11-1

- 1 选择 SPI NOR Flash 厂商(MXIC 或 Winbond, 若使用其它厂商 flash, 请与瑞佑联络)
- 2 选择操作 SPI NOR Flash 的 SPI bus 与 CS
- 3 选择操作 SPI NOR Flash 的寻址方式 (>128Mbits : 32-bit addressing mode, ≤ 128Mbits : 24-bit addressing mode)
- 4 设定图文件位于 SPI Flash 内存中的起始地址
- 5 设定图文件大小(单位:bytes)
- 6 设定数据经过 JPEG 译码后, 搬至 SDRAM 内存的位置 Canvas 坐标, 例如数据要显示在(X=100,Y=50)位置。
- 7 按下 Run 按键, 执行 JPEG 功能。
- 8 显示 JPEG API C 语言程序。



请输入存储在 **SPI Flash** 中 **JPEG** 文件的正确参数，例如源起始地址和 **JPEG** 文件大小。如果相关参数的输入值不正确，当执行此功能时，可能会在显示屏上显示不正确，从而导致系统崩溃。如果出现上述问题，请断开 **RA8889** 演示板上与 **PC** 相连的 **USB** 电缆，暂时终止与 **PC** 的连接，结束 **AP Tool** 软件的工作，并关闭 **RA8889** 演示板上的系统电源，然后再次打开电源。完成上述操作后，将演示板上的 **USB** 电缆再次连接到 **PC**，重新启动并操作与 **AP** 工具相关的功能，然后输入正确 **JPEG** 参数。

执行后画面显示结果:



图 11-2

12. AVI Function

RA8889 支援 Motion JPEG 编码的 AVI 格式视频，但不支持声音播放，将不含声音数据的 AVI 视频(Motion JPEG)烧录于



SPI Flash 中，点选 IC Function Button 中的  按键，开启 AVI 功能。

1 (1).Select the manufacturer of the SPI flash
(2).Select SPI bus and chip select

2 Choose between 24-bit and 32-bit addressing mode

3 Set buffer1 (PIP1 buffer) start address

4 Set buffer2 (Shadow buffer) start address

5 (1).Set the image start address in the SPI flash memory
(2). Set the image file size

6 (1). Set Destination (LCD Screen) Upper-Left corner coordination
(2). Set the width and height of the video

7 Video control buttons : Play、Pause/Resume、Stop

8 Show API C code

图 12-1

- 1** (1).选择 SPI NOR Flash 厂商(MXIC 或 Winbond，若使用其它厂商 flash，请与瑞佑联络 ) (2).选择操作 SPI NOR Flash 的 SPI bus 与 CS
- 2** 选择操作 SPI NOR Flash 的寻址方式 (>128Mbits : 32-bit addressing mode, ≤128Mbits : 24-bit addressing mode)
- 3** 设定 buffer1(PIP1 buffer)起始地址
- 4** 设定 buffer2(Shadow buffer)起始地址
- 5** (1).设定 AVI 视频位于 SPI Flash 内存中的起始地址 (2). 设定 AVI 视频大小(单位:bytes) 

- 6 (1).设定 AVI 视频显示于 LCD 画面的位置, 例如视频要显示在 LCD 画面上(X=200,Y=300)位置。(2).设定 AVI 视频的宽与高。
- 7 AVI 控制按键, Play 视频开始播放、Pause/Resume 暂停播放/继续播放、Stop 视频停止播放。
- 8 显示 AVI API C 语言程序。



请输入存储在 **SPI Flash** 中 **AVI** 文件的正确参数, 例如源起始地址和 **AVI** 文件大小。如果相关参数的输入值不正确, 当执行此功能时, 可能会在显示屏上显示不正确, 从而导致系统崩溃。如果出现上述问题, 请断开 **RA8889** 演示板上与 **PC** 相连的 **USB** 电缆, 暂时终止与 **PC** 的连接, 结束 **AP Tool** 软件的工作, 并关闭 **RA8889** 演示板上的系统电源, 然后再次打开电源。完成上述操作后, 将演示板上的 **USB** 电缆再次连接到 **PC**, 重新启动并操作与 **AP** 工具相关的功能, 然后输入正确 **AVI** 参数。

按下 **Play** 按键, 内存配置状态与 LCD 画面显示结果(以颜色深度为 24bpp 为例):

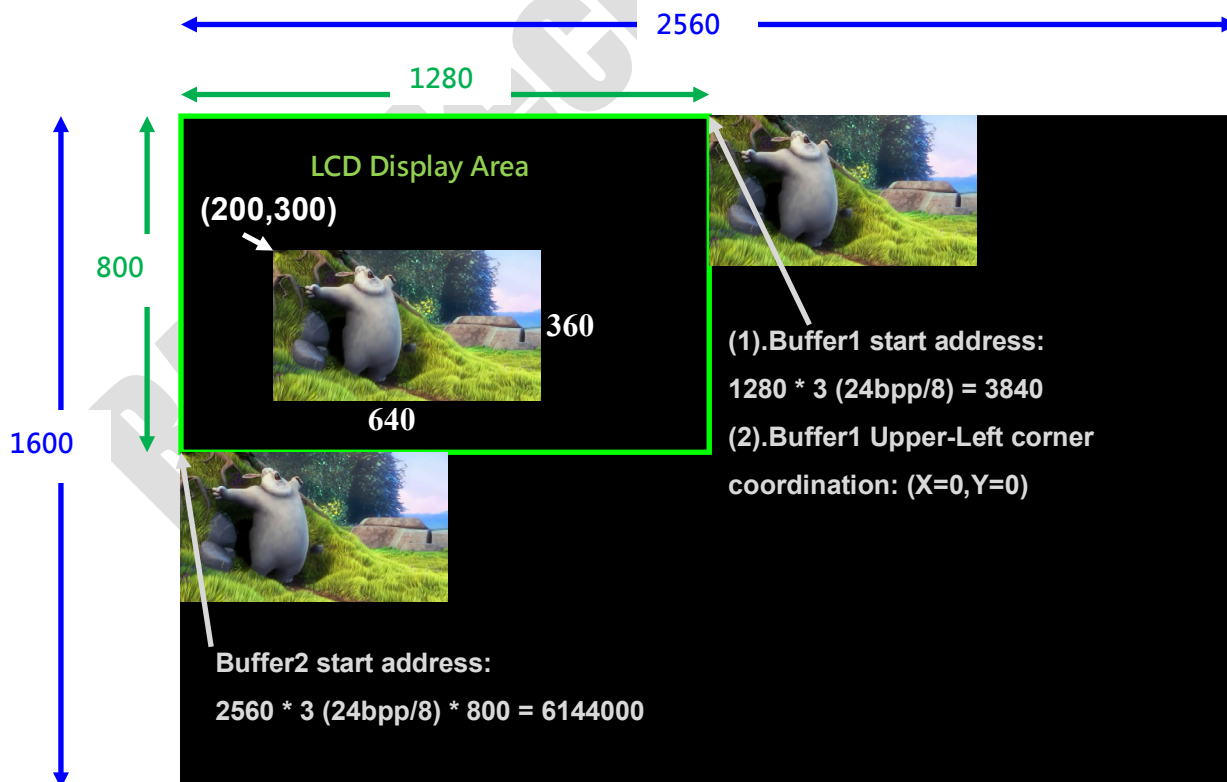


图 12-2