SEGGER emWin 图形库 (emWinGraphics)

1.0

特性

- 该组件集成了 PSoC3 的 emWin 8051 图形库和 PSoC 5 全功能 emWin 图形库 V5.02
- 这些库可与 Keil_PK51、GCC、Keil MDK 和 Keil RVDS 工具链结合使用
- 驱动程序可供图形 LCD 接口和图形 LCD 控制器组件使用

概述

emWin 是一个嵌入式图形库和图形用户界面 (GUI)，旨在为任何通过图形显示工作的应用程序提供有效且独立于处理器和 LCD 控制器的 GUI。它兼容于单任务和多任务环境。由 SEGGER Microcontroller 开发，emWin在嵌入式设备非常流行。赛普拉斯已从SEGGER获得 emWin 库的授权许可，可免费向客户提供全功能的图形库。

何时使用 emWinGraphics

该组件提供 SEGGER emWin 图形库的访问权限及其功能。emWinGraphics 组件是一个完整的、易用的软件程序包，适用于 PSoC 3 和 PSoC 5 目标硬件。软件包由以库形式提供的 emWin GUI、所需的头文件和有关 PSoC 实现的源文件组成。

emWin 图形库组件允许你进行快速高效的 GUI 开发，应用于图形 LCD 工作的任何程序。

入门

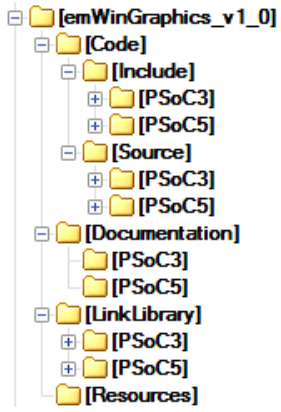
设置

emWinGraphics 组件以 zip 文件的形式提供。你可以在以下位置获取最新版本的组件：

http://www.cypress.com/go/comp_emWin

解压缩到任何文件夹，并保留 zip 文件的目录结构。确保文件在解压后不是只读格式。解压安装的目录结构类似于图 1 所示。

图 1. 目录结构



除 Resources（资源）外，每个主目录都划分成 PSoC 3 和 PSoC 5 子目录。这是因为两个器件系列的实现不同。PSoC 3 支持功能有限的库版本 (emWin 8051)，而 PSoC 5 支持全功能版本的标准 emWin 库。每个库版本提供的详细功能记录在 SEGGER emWin 文档中，文档位于 *Documentation\PSoC3* 和 *Documentation\PSoC5* 目录中。有关 PSoC 3 和 PSoC 5 各个方面功能的详细信息，请参阅本文档的“功能说明” 部分。

下表显示所有主要 emWinGraphics 目录的内容。

目录(Directory)	内容(Contents)
Code\Include (代码\包含文件)	GUI 头文件
Code\Source(代码\源文件)	需要添加到项目的源文件
Documentation(文档)	emWin 用户指南
LinkLibrary	支持各个工具链的 emWin 库
Resources(资源)	支持位图的SEGGER emWin 示例文件和窗口程序

使用 emWinGraphics 库

若要使用 emWinGraphics 库，请按下列步骤操作。所有步骤在后续章节中进一步解释。

1. 集成 emWinGraphics 库到 PSoC Creator 项目。
2. 添加一个可与图形 LCD 面板进行通信的组件到项目。emWinGraphics 库是一个软件库。使用适当的硬件组件，与面板进行通信。赛普拉斯组件库包含两个与 LCD 面板接口的组件：图形 LCD 接口或图形 LCD 控制器。
3. 添加触摸屏组件到项目（可选）。触摸屏组件（如电阻式触摸组件）可与 emWinGraphics 库结合使用。

4. 使用示例代码的编译、链接和测试。emWinGraphics 库提供单任务和多任务环境的示例代码。可通过这些示例来学习库的使用方法。这些示例可在 Resources(资源)目录中找到。
5. 对于多任务应用，调整库使其适用于操作系统 (OS)。如果有多项任务同时访问显示屏，则必须定义 OS 接口程序。有关示例应用的详细信息，请参考第 13 章：“执行模式： *《emWin 用户手册》* 的单任务/多任务”。
6. 使用 emWin 编写你自己的应用程序。此时，你应准备好开始使用 emWin 进行编程。阅读 *《emWin 用户手册》* 中的参考章节，详细了解该库所提供的各种功能。

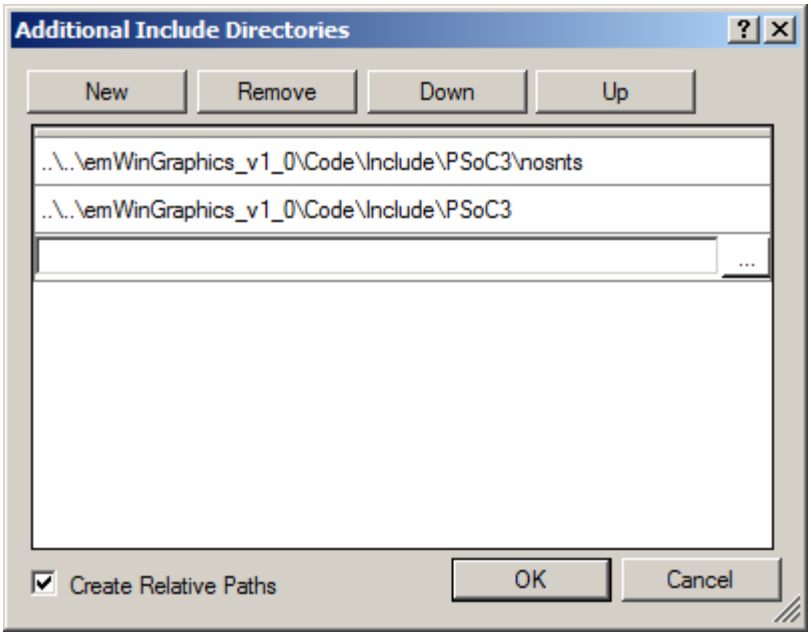
集成 emWinGraphics 库到 PSoC Creator

要将 emWinGraphics 库集成到你的项目中，请执行这些步骤。本示例使用 Keil_PK51 工具链。其他工具链的支持相类似。

1. 首先，决定所需的库。根据你是否使用 RTOS (os)，触摸屏 (ts)， 或两者同时使用做出决定。根据所支持的选项来命名 emWin 库。在本例中，库名为 emWinnosnts.lib，它是一个无 OS (nos) 和触摸屏 (nts) 支持的库。
2. 添加必需的包含文件。emWin include(包含)目录包含：所有应用程序通用的包含文件，以及包含在根据所选项指定子目录中的文件。Compiler（编译器）选项的 **Additional Include Directories**（附加包含目录）字段应含有一个条目，涉及通用和特定库的包含文件路径。

   从菜单 **Project（项目） > Build Settings（编译设置） > Compiler（编译器） > General（常规） > Additional Include Directories（附加包含目录）**中，指定包含文件的路径（如图 2 所示）。

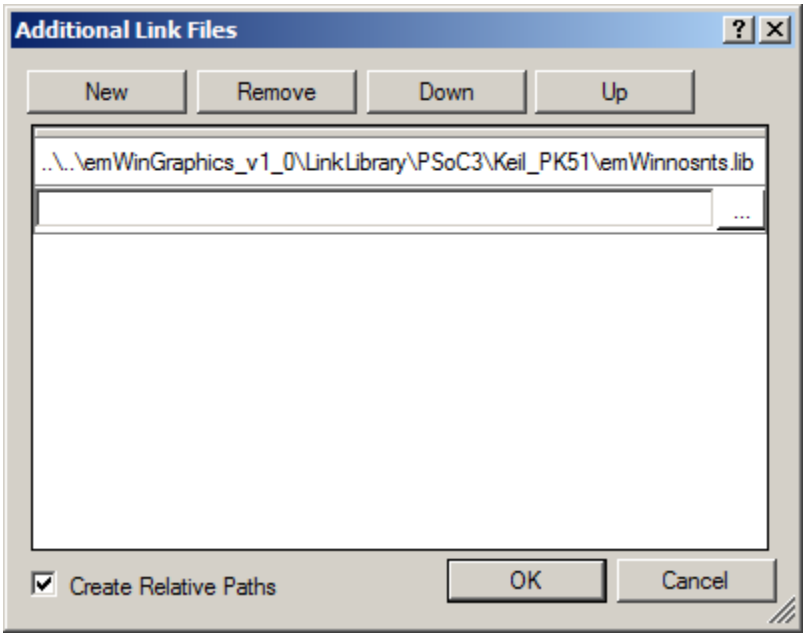
图 2. Additional Include Directories（附加包含目录）对话框



3. 添加链接库文件。emWin GUI 可与 Keil_PK51、GNU CC、KEIL MDK 或 KEIL RVDS 工具链结合使用，因此，根据选择的工具链选择库。

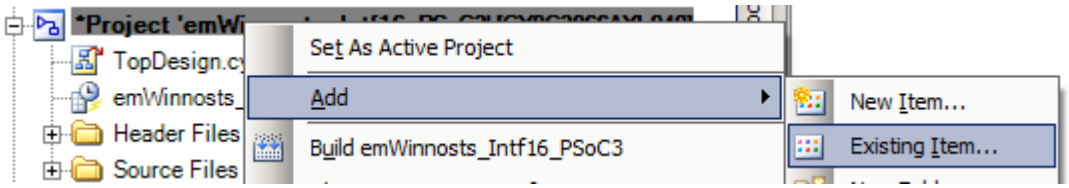
   从菜单 **Project（项目） > Build Settings（编译设置） > Linker（连接器） > General（常规） > Additional Link Files（附加链接文件）**中，选择图 3 所示的库文件。

图 3. Additional Link Files（附加链接文件）对话框

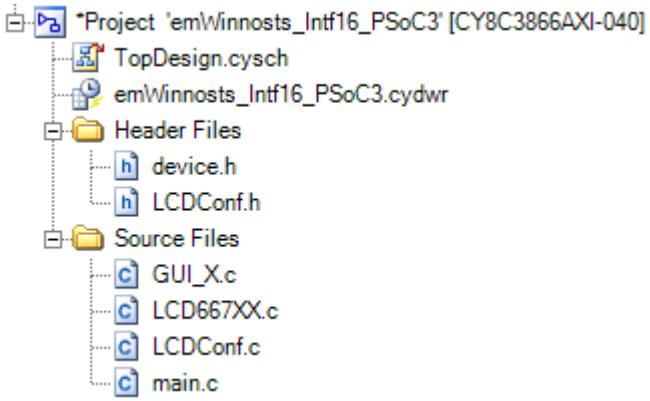


注意 将库添加到连接器的过程与 GCC 工具链有所不同。在这种情况下，将含有库的目录添加到带有“Additional Library Directories”（附加库目录）设置的连接器，并使用“Additional Libraries”（附加库）设置指定库。此设置所用的库名必须带有“lib”前缀，并将“.a”后缀从该名称中删除。

4. 将所需的源文件添加到为 LCD 组件选择的项目。选择接口组件或控制组件。根据上述选择，文件可在 *Source\PSoC3\Graphics LCD Interface Parallel* 目录或 *Source\PSoC3\Graphics LCD Controller* 目录中找到。通常，这些文件针对特定项目进行编辑，因此，建议将所需要的文件复制到项目的根文件夹，然后添加到你的项目中。使用菜单选项 **Project（项目）> Add（添加）> Existing Item（现有项）**将这些文件添加到项目中。如果包含文件并不放置在项目的根目录中，则必须将包含文件的路径添加到编译设置中。



此处显示的项目已添加了所有来自 *Source\PSoC3\Graphics LCD Interface Parallel* 目录的文件（除 *GUI_X_Touch.c* 外）。此示例并不使用 OS 或触摸屏，因此不包括与触摸屏有关的文件。

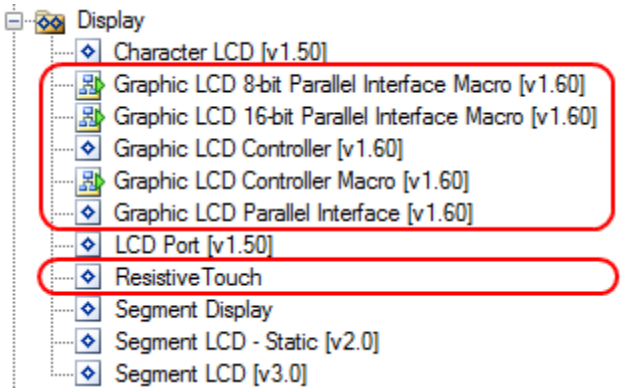


与 emWin 使用的 PSoC Creator 组件

emWinGraphics 库是一个软件库。它并不包含任何驱动 LCD 面板的硬件。使用 emWinGraphics 库构建设计时，需要图形 LCD 组件（图形 LCD 接口或图形 LCD 控制器）。两个组件为多个图形面板提供支持，但它们驱动不同类型的面板。此外，可选的触摸屏组件可与 emWin 结合使用，以实现触摸用户界面。这些组件之间的通信完全通过软件实现。

单个 LCD 图形设计正好含有一个图形 LCD 组件的实例，和可选的一个触摸屏组件的实例。所有这些组件都位于 Component Catalog(组件目录)的 Display (显示)文件夹中（如图 4 高亮显示）。

图 4. 显示文件夹



图形 LCD 接口 (GraphicLCDIntf) 组件

将 GraphicLCDIntf 组件用作面板的接口，以将图形 LCD 控制器和驱动器件集成至 LCD 面板。此类面板还包括帧缓冲器, 它由集成至面板的 LCD 控制器管理。组件对此控制器执行读取和写入操作。

创建 GraphicLCDIntf 应用程序需要执行下列步骤

1. 将图形 LCD 并行接口宏放置在原理图上。根据所用的 LCD 控制器，选择 8 位或 16 位接口。
2. 配置 GraphicLCDIntf 组件，满足对 LCD 控制器执行写入和读取操作的要求。有关配置的详细信息，请参考 GraphicLCDIntf 数据手册。
3. 确保所有关于 GraphicLCDIntf 组件的 emWin 文件都包含在你的项目中。文件位于图形 LCD 并行接口的子目录中，而子目录位于 *Code\Source\PSoC3 或 Code\Source\PSoC5* 目录中，具体取决于 PSoC 器件。
4. 根据你的 LCD 显示硬件修改所需的源文件。可以在不做任何改变的情况下，使用 emWin 发布版本中的大多数源文件，但你必须更改一些配置文件以使目标系统正常工作。有关 emWin 配置的详细信息，请参阅 *《emWin 用户手册》* 中的“配置”章节。

图形 LCD 接口应用程序所需的 emWin 文件

使用 GraphicLCDCtrl 组件来构建 PSoC 3 应用程序，需要下列来自 *Code\Source\PSoC3\Graphics LCD Interface Parallel* 目录的文件。

- *GUI_X.c*：你必须编辑此文件，以提供定时程序（emWin 库需要的时间消息）和调试程序（可选）。
- *GUI_X_Touch.c*（可选）：提供硬件程序实现电阻式触摸界面。此文件无需编辑。
- *LCD667XX.c*：显示驱动程序。此文件无需编辑。
- *LCDConf.h*：你必须编辑此文件，以提供显示驱动程序编译所需的通用配置选项。有关可用配置选项的详细信息，请参阅 *《emWin 用户手册》* 中的“显示驱动程序”章节。
- *LCDConf.c*：你必须编辑此文件，以提供所用的特定显示控制器的初始化程序。

使用 GraphicLCDCtrl 组件来构建 PSoC 5 应用程序，需要下列来自 *Code\Source\PSoC5\Graphics LCD Interface Parallel* 文件夹的文件。

- *GUI_X.c*：你必须编辑此文件，以提供定时程序（emWin 库需要的时间消息）和调试程序（可选）。如果应用程序不使用 OS，则应将它包含在项目中。如果你使用 OS，则应使用此文件的 OS 特定版本。
- *GUI_X_embOS.c*：此文件提供适用于 embOS RTOS 的 GUI 内核接口程序。如果应用程序使用 embOS 实时内核，则将此文件包含在你的项目中。通常此文件无需编辑。
- *GUI_X_uCOS.c*：此文件提供适用于 uC/OS RTOS 的 GUI 内核接口程序。如果应用程序使用 uC/OS 实时内核，则将此文件包含在你的项目中。通常此文件无需编辑。
- *GUI_X_Touch.c*（可选）：提供硬件程序实现电阻式触摸界面。此文件无需编辑。

- *GUIConf.c*：分配内存给 emWin 操作。你可以针对特定的芯片和应用程序调整此文件。默认的设置是 8,192 字节 SRAM，可供 emWin 使用。
- *GUIDRV_CompactColor_16.c*：显示驱动程序。此文件无需编辑。
- *LCDConf.h*：表明你正在使用紧凑型彩色驱动程序 (Compact Color Driver)。此文件无需编辑。
- *LCDConf_CompactColor_16.h*：你必须编辑此文件，以提供编译显示驱动程序所需的通用配置选项。有关可用配置选项的详细信息，请参阅 *《emWin 用户手册》* 中的“显示驱动程序”章节。
- *LCDConf.c*：你必须编辑此文件，以提供所使用特定显示控制器的初始化程序。

图形 LCD 控制器 (GraphicLCDCtrl) 组件

GraphicLCDCtrl 组件向带有 LCD 驱动器的 LCD 面板提供接口，但不是 LCD 控制器。此类面板并不包含帧缓冲器。必须提供外部的帧缓冲器。此组件直接驱动控制信号和管理外部 SRAM 中的帧缓冲器。

创建 GraphicLCDCtrl 应用程序需要执行下列步骤

1. 把图形 LCD 控制器宏放置在原理图上。
2. 配置 GraphicLCDCtrl 组件，以满足屏幕刷新要求和 LCD 面板的数据处理定时参数。有关配置的详细信息，请参考 GraphicLCDCtrl 数据手册。
3. 确保所有关于 GraphicLCDCtrl 组件的 emWin 文件都包含在该项目中。文件位于图形 LCD 控制器子目录，该子目录位于 *Code\Source\PSoC3 或 Code\Source\PSoC5* 目录中，具体取决于所使用的 PSoC 器件。
4. 根据你的 LCD 显示屏硬件更改所需的源文件。可以在不做任何改变的情况下使用 emWin 发布版本中的大多数源文件，但你必须编辑一些配置文件以使目标系统正常工作。有关 emWin 配置的详细信息，请参阅 *《emWin 用户手册》* 中的“配置”章节。

图形 LCD 控制器应用程序所需的 emWin 文件

使用 GraphicLCDCtrl 组件来构建 PSoC 3 应用程序，需要下列来自 *Code\Source\PSoC3\Graphics LCD Controller* 目录的文件：

- *GUI_X.c*：你必须编辑此文件，以提供定时程序（emWin 库需要的时间消息）和调试程序（可选）。
- *GUI_X_Touch.c*（可选）：提供硬件程序实现电阻式触摸界面。此文件无需编辑。
- *GUIDRV_Control.c*：显示驱动程序。此文件无需编辑。

- *LCDConf.h*：你必须编辑此文件，以提供编译显示驱动程序所需的通用配置选项。有关可用配置选项的详细信息，请参阅 *《emWin 用户手册》* 中的“显示驱动程序”章节。
- *LCDConf.c*：你必须编辑此文件以提供所使用特定 LCD 面板的初始化程序。

使用 GraphicLCDCtrl 组件来构建 PSoC 5 应用程序，需要下列来自 *Code\Source\PSoC5\Graphics LCD Controller* 目录的文件。

- *GUI_X.c*：你必须编辑此文件，以提供定时程序（emWin 库需要的时间消息）和调试程序（可选）。如果应用程序不使用 OS，则应将它包含在项目中。如果使用 OS，则应使用此文件的 OS 特定版本。
- *GUI_X_embOS.c*：此文件提供适用于 embOS RTOS 的 GUI 内核接口程序。如果应用程序使用 embOS 实时内核，则将此文件包含在你的项目中。通常此文件无需编辑。
- *GUI_X_uCOS.c*：此文件提供适用于 uC/OS RTOS 的 GUI 内核接口程序。如果应用程序使用 uC/OS 实时内核，则将此文件包含在你的项目中。通常此文件无需编辑。
- *GUI_X_Touch.c*（可选）：提供硬件程序实现电阻式触摸界面。此文件无需编辑。
- *GUIConf.c*：分配内存给 emWin 操作。可以针对特定的芯片和应用程序调整此文件。默认的设置是 8,192 字节 SRAM，可供 emWin 使用。
- *GUIDRV_Control.c*：显示驱动程序。此文件无需编辑。
- *LCDConf.h*：指定要使用的驱动程序，并含有一些通用配置选项，包括显示方向和颜色深度。可以在必要时编辑此文件。
- *LCDConf.c*：你必须编辑此文件，以提供所使用特定 LCD 面板的初始化程序。

设置模拟触摸屏

连接触摸屏的最常用方法是 4 引脚模拟接口。为 emWin 提供一个驱动程序。emWin 触摸屏驱动程序管理模拟输入（从 A/D 转换器）、去抖和触摸屏校正。

触摸屏驱动程序使用函数 GUI_TOUCH_Exec() 持续地监控和更新触摸面板。如果函数识别发现了某项触摸操作或更改了什么，则会调用适当的触摸屏 API 程序。

使用下列步骤向项目添加触摸屏功能：

1. 确保项目设置使用带有触摸屏支持的 emWin 库。有关如何选择 emWin 库的详细信息，请参阅本文档的“将 emWinGraphics 库集成至 PSoC Creator”一节。
2. 添加 *GUI_X_Touch.c* 文件到你的项目。
3. 将电阻式触摸组件放置在原理图上。组件位于 Component Catalog(组件目录)的 Display(显示)文件夹中。

4. 定期调用 GUI_TOUCH_Exec()。应用程序应以大约 100 次/秒调用它。如果你使用的是实时操作系统，确保调用此函数的最简单方法是创建一个单独的任务。当不使用多任务系统时，你可以使用中断服务程序。
5. 确定面板的 A/D 转换器返回的最小和最大值，并使用这些值更改函数 GUI_TOUCH_Calibrate() 的参数。emWin 需要这些值将测量结果转换成以像素为单位的触摸位置。有关触摸屏配置的详细信息，请参阅 *《emWin 用户指南》* 中的第 19.4.2 章“模拟触摸屏驱动程序”。《用户指南》中引用的文件 *TOUCH_Sample.c* 和 *TOUCH_Calibrate.c* 均可在 *Resources\Examples* 目录下找到。

EmWin 示例文件

emWinGraphics 组件随可供 PSoC 3 和 PSoC 5 使用的许多 emWin 示例文件一同交付。它们放置在 *Resources\Samples* 目录下的 PSoC 3 或 PSoC 5 子目录。如果你不熟悉 emWin，赛普拉斯建议你先编译、连接和测试示例程序，再创建你自己的应用程序。

运行示例文件：

1. 添加一个演示示例文件到你的项目。
2. 编辑主文件 (*main.c*) 包含 *GUI.h*。使能中断，然后按下列代码所示调用 MainTask()。所有 emWin 示例使用 MainTask() 作为进入点。

```
#include <device.h>
#include "GUI.h"

extern void MainTask(void);

void main()
{
    CyGlobalIntEnable;
    MainTask();

    for(;;)
    {
        /* Place your application code here. */
    }
}
```

3. 构建和运行项目。

可用库

emWin GUI 可与 Keil_PK51、GNU CC、KEIL MDK 或 KEIL RVDS 工具链结合使用。对于每个工具链（除 Keil_PK51 外），库提供两个选择 — OS (os) 支持和触摸屏 (ts) 支持。OS 支持提供

初始化图形库期间必需的钩子(hooks)以及 OS 锁定和解锁库中的资源所需的钩子。特定库名支持的选项如下所示：

选项	OS 支持	TS 支持
nosnts	否	否
nosts	否	是
osnts	是	否
osts	是	是

适用于 Keil_PK51 工具链的 emWin 库

该库位于 *LinkLibrary\PSoC3\Keil_PK51* 目录中，其名称如下所示：

emWin<options>.lib

可用选项是 nosnts 和 nosts。

示例：

libemWinnosts.a 是一个无 OS 支持、有触摸屏支持的库。

适用于 GCC 工具链的 emWin 库

该库位于 *LinkLibrary\PSoC5\GCC* 目录中，其名称如下所示：

libemWin<options>.a

可用选项有 nosnts、nosts、osnts 和 osts。

示例：

libemWinosnts.a 是一个有 OS 支持、无触摸屏支持的库。

注意，使用 GCC 工具链时，必须在 **Linker**（连接器）选项的 **Additional Libraries**（其他库）字段中省略库名的“lib”前缀和“.a”文件扩展名。

适用于 Keil MDK 工具链的 emWin 库

库位于 *LinkLibrary\PSoC5\ARM_MDK* 目录中，其名称如下所示：

emWin<options>.lib

可用选项有 nosnts、nosts、osnts 和 osts。

示例：

emWinosts.lib 是一个有 OS 支持和触摸屏支持的库。

适用于 ARM RVDS 工具链的 emWin 库

库位于 *LinkLibrary\PSoC5\ARM_RVDS* 目录中，其名称如下所示：

emWin<options>.lib

可用选项有 nosnts、nosts、osnts 和 osts。

示例：

emWinosts.lib 是一个无 OS 支持和触摸屏支持的库。

放置

完全软件实现。

性能和资源使用

实际的性能和资源使用取决于许多因素（CPU、编译器、内存模型、优化、配置、LCD 控制器接口等）。本节包含图像绘制操作的性能和一个典型配置的内存资源使用信息。

性能

产品	CPU 速度 (MHz)	图形 LCD 组件	bpp	填充[1]	小字体[2]	大字体[2]	Bitmap 1 bpp	Bitmap 8 bpp	Bitmap 16 bpp
PSoC 3	24	图形 LCD 接口（16-位）	16	112K	24K	30K	36K	14K	NA
PSoC 3	60	图形 LCD 接口（16-位）	16	279K	56K	70K	82K	33K	NA
PSoC 3	24	图形 LCD 接口（8-位）	16	63K	20K	24K	28K	13K	NA
PSoC 3	60	图形 LCD 接口（8-位）	16	156K	46K	56K	65K	30K	NA
PSoC 3	24	图形 LCD 控制器	16	34K	21K	23K	26K	12K	NA

[1]根据表格所列的性能值和显示屏的分辨率，可按以下方式计算帧每秒的性能值 (fps)：

fps = 像素每秒/ (LCD_XSIZE × LCD_YSIZE)

[2] 根据这些值，按以下方式计算每秒字符的大概数量：

每秒字符数量 = 像素每秒/ (字符宽度 × 字符高度)

有关字符大小的详细信息，请参阅 *《emWin 用户手册》* 的“字体”一节。

产品	CPU 速度 (MHz)	图形 LCD 组件	bpp	填充[1]	小字体[2]	大字体[2]	Bitmap 1 bpp	Bitmap 8 bpp	Bitmap 16 bpp
PSoC 3	60	图形 LCD 控制器	16	75K	48K	54K	61K	29K	NA
PSoC 5	24	图形 LCD 接口（16-位）	16	662K	156K	177K	195K	150K	589K
PSoC 5	60	图形 LCD 接口（16-位）	16	1.64M	315K	357K	400K	306K	1.40M
PSoC 5	24	图形 LCD 接口（8-位）	16	300K	124K	136K	149K	117K	178K
PSoC 5	60	图形 LCD 接口（8-位）	16	570K	231K	252K	277K	219K	335K
PSoC 5	24	图形 LCD 控制器	16	147K	131K	130K	135K	114K	152K
PSoC 5	60	图形 LCD 控制器	16	224K	200K	192K	207K	184K	229K

内存要求

emWin 库的内存使用差异很大，主要取决于所用的应用程序和特性。下表提供一些示例应用程序的大小（包括代码、字体、引导程序、图像等），以帮助你了解基本内存要求。若要更加精确地计算，请参阅 emWin 库的 SEGGER 文档。

采样	Keil 8051		GCC-4.4.1		说明
	Flash（闪存）	SRAM	Flash（闪存）	SRAM	
Hello World 应用程序	19.4 KB	3.6 KB	24.5 KB	2.9 KB	本示例项目仅使用显示文本的基本功能。
2D 图形应用程序	33 KB	3.7 KB	34 KB	2.9 KB	此示例使用不同的字体显示文本，并执行 2D 图形库的一些绘制操作。
Window 应用程序	不可用	不可用	84 KB	11 KB	本项目是使用窗口管理器和各种小组件的“典型”应用程序。

应用程序编程接口

emWinGraphics 组件并不包括任何不存在于标准 emWin 库中的 API 函数。请参考 SEGGER emWin 文档，了解可用 emWin 程序的详细说明。

功能描述

详细的功能记录在 SEGGER emWin 文档中。本节确定与 PSoC 3 和 PSoC 5 有关的 SEGGER 功能的主要类别。

SEGGER emWin 库特性	PSoC 3	PSoC 5
2-D 图形库	完全支持	完全支持
显示位图文件	不支持编译位图格式以外的其他格式。 不支持 16-位位图。完全支持位数等于和小于 8 的位图。	支持所有编译的位图格式。 其他格式 (JPEG、GIF、PNG) 可能获得支持，但可能由于 RAM 要求而不可行
字体	仅支持比例字体	支持所有字体类型
位图转换器	编译的窗口应用程序以二进制码提供。	
色彩支持	完全支持	完全支持
内存设备	不支持	完全支持
多任务 (RTOS)	不支持	完全支持
窗口管理器	不支持	完全支持
窗口对象（小组件）	不支持	完全支持
虚拟屏幕/虚拟页面	完全支持	完全支持
多层/多显示	不支持	不支持
指针输入设备	仅完全支持触摸屏	仅完全支持触摸屏
子画面和游标	不支持	完全支持
抗混叠	不支持	完全支持
外语	不支持	不支持
VNC	不支持	不支持
驱动程序	紧凑型彩色驱动程序和赛普拉斯定制驱动程序	

组件更改

版本 1.0 为 emWinGraphics 库的首次发行版。

© 赛普拉斯半导体公司，2012。此处所包含的信息如有更改，恕不另行通知。除赛普拉斯产品的内嵌电路之外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不根据专利或其他权利以明示或暗示的方式授予任何许可。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯产品不保证能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

PSoC® 是赛普拉斯半导体公司的注册商标，PSoC Creator™ 和 Programmable System-on-Chip™ 是赛普拉斯半导体公司的商标。此处引用的所有其他商标或注册商标归其各自所有者所有。

所有源代码（软件和/或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和/或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定的用途之外，未经赛普拉斯的明确书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对此材料提供任何类型的明示或暗示保证，包括（但不仅限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不做出通知的情况下对此处所述材料进行更改的权利。赛普拉斯不对此处所述之任何产品或电路的应用或使用承担任何责任。对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

产品使用可能受适用的赛普拉斯软件许可协议限制。