



前言

交互式界面被越来越多地集成于多种应用中，例如医疗设备、过程控制、手机和其它手持设备。这些界面主要基于使用彩色 LCD 的图形 HMI（人机界面）。

对支持彩色的需求在全世界范围内极大增长。本应用笔记旨在讨论这一问题，它说明了如何使用 STM32F10xxx FSMC（灵活的静态存储控制器）来驱动 TFT 彩色 LCD。本文首先说明了怎样将彩色 LCD 连接到 FSMC，然后提供了一个 TFT LCD 接口样例。

与本应用笔记一起交付的演示固件、STM32F10xxx 固件库、以及其它此类固件可从意法半导体网站下载：www.st.com。

目录

1	STM32F10xxx 灵活的静态存储控制器 (FSMC) 概述	6
1.1	FSMC NOR Flash/SRAM 存储区域说明	6
2	TFT LCD（薄膜晶体管液晶显示器）与 FSMC 接口	9
2.1	常用的彩色 LCD 接口	9
2.2	FSMC 与 LCD 模块接口的典型用法	10
2.3	将 FSMC 连接至 LCD Intel 8080（I80）接口	10
2.4	将 FSMC 连接至 LCD Motorola 6800（M68）接口	11
2.5	LCD 与 100 引脚封装的大容量 STM32F10xxx 器件 FSMC 的接口	12
2.6	时序计算	13
2.6.1	LCD 时序	13
2.6.2	FSMC 时序	14
3	Ampire AM-240320L8TNQW00H TFT LCD 与 FSMC 的接口	15
3.1	Ampire TFT LCD 接口	15
3.2	硬件连接	15
3.3	时间计算	16
4	固件说明	18
4.1	LCD 驱动固件说明	18
4.2	运行演示	19
4.2.1	菜单	19
4.2.2	启动演示	20
4.2.3	导航	22
4.3	LCD 演示应用	23
4.3.1	内部 Flash	24
4.3.2	NOR Flash	26
4.3.3	外部 SRAM	27
4.3.4	NAND Flash	29
4.3.5	SD 卡	32

5	彩色 TFT LCD 与 FSMC 接口：性能	34
6	LCD 演示编程	35
	6.1 编程 M29W128 NOR Flash	36
	6.2 LCD 演示编程	39
7	结论	43
8	修订历史	44

表格索引

表 1.	MCU 中 LCD 控制器的优点和缺点	9
表 2.	并行接口读 / 写状态	10
表 3.	LCD Intel 8080 接口引脚.....	11
表 4.	FSMC 和 LCD Intel 6800 引脚	11
表 5.	LCD 8080 接口时序特性.....	13
表 6.	FSMC 和 Ampire LCD 引脚.....	15
表 7.	LCD 8080 接口时序特性.....	16
表 8.	驱动库说明	18
表 9.	高级 LCD 驱动函数	18
表 10.	LCD 显示速度	34
表 11.	文档修订历史	44

图片索引

图 1.	异步 NOR Flash 读访问时序	7
图 2.	异步 NOR Flash 写访问时序	8
图 3.	将 FSMC 连接至 LCD Intel 8080 接口	11
图 4.	将 FSMC 连接至 LCD Motorola 6800 接口	12
图 5.	将 FSMC 连接至 LCD Motorola 6800 接口 ⁽¹⁾	12
图 6.	将 FSMC 连接至 100 引脚封装器件中的 LCD Intel 8080 接口	13
图 7.	TFT LCD 与 FSMC 的硬件连接	16
图 8.	演示菜单的结构	19
图 9.	警告消息	20
图 10.	ST 标识	20
图 11.	STM32 系列	20
图 12.	主菜单	21
图 13.	相应的子菜单	21
图 14.	在演示菜单中导航	22
图 15.	无位图文件（见注释 1）	23
图 16.	将图片复制到内部 Flash	24
图 17.	将图片复制到内部 Flash 完成	24
图 18.	动画显示 STM32F10xxx 的显示速度	25
图 19.	ST 标识	25
图 20.	时间显示子菜单	25
图 21.	时间显示子菜单	26
图 22.	时间显示子菜单	26
图 23.	时间显示子菜单	27
图 24.	将图片复制到外部 SRAM	27
图 25.	将图片复制到外部 SRAM 完成	28
图 26.	时间显示子菜单	28
图 27.	时间显示子菜单	29
图 28.	将图片复制到 NAND Flash	29
图 29.	将图片复制到 NAND Flash 完成	30
图 30.	时间显示子菜单	30
图 31.	NAND 擦除操作	31
图 32.	NAND 擦除操作完成	31
图 33.	将图片复制到 SD 卡	32
图 34.	将图片复制到 SD 卡完成	32
图 35.	时间显示子菜单	33
图 36.	编程 NOR Flash	36
图 37.	选择 STM32_FSMC-LCD_Demo_NORFlash.dfu 文件	37
图 38.	点击 Yes 继续	37
图 39.	NOR Flash 编程	38
图 40.	编程演示	39
图 41.	选择 STM32_FSMC-LCD_Demo.dfu 文件	40
图 42.	点击 Yes 继续	40
图 43.	内部 Flash 编程	41
图 44.	退出 DFU 模式	42

1 STM32F10xxx 灵活的静态存储控制器 (FSMC) 概述

STM32F10xxx 灵活的静态存储控制器 (FSMC) 为嵌入式外部存储控制器，可使 STM32F10xxx 微控制器与多种存储器接口，包括 SRAM、NOR Flash、NAND Flash 及 LCD 模组。

FSMC NOR Flash/SRAM 存储区域适合 MCU 并行彩色 LCD 接口，将在本章讲述。

1.1 FSMC NOR Flash/SRAM 存储区域说明

FSMC 提供了如下特性以控制 NOR Flash/SRAM：

- 选择将要用于映射 NOR Flash/SRAM 的存储区域：有四个独立的存储区域可用来与 NOR Flash/SRAM/PSRAM 接口，使用单独的片选引脚选择每个存储区域。
- 启用或禁用地址 / 数据复用特性
- 选择要使用的存储器类型：NOR Flash/SRAM/PSRAM
- 定义外部存储器数据总线宽度：8/16 位
- 对于 NOR Flash 同步存储器，启用或禁用突发访问模式
- 配置等待信号的法：启用 / 禁用、极性设置、时序配置
- 启用或禁用扩展模式：此模式用于为读写操作使用不同的时序配置来访问存储器。

因为 NOR Flash/PSRAM 控制器支持异步和同步存储器，所以用户应根据存储器特性，仅选择有用的参数。

FSMC 还提供了对多个参数编程，以正确与外部存储器接口的可能。取决于存储器类型，一些参数会用不到。

若使用了外部异步存储器，则用户必须根据存储器数据手册中规定的 AC 时序信息，计算并设置下列参数：

- ADDSET：地址建立时间
- ADDHOLD：地址保持时间
- DATAST：数据建立时间
- ACCMOD：访问模式

此参数可让 FSMC 灵活地访问多种异步静态存储器。若存储器支持此类特性，则有四种扩展访问模式可使读存储器与写访问同时使用不同时序。

当启用扩展模式时，FSMC_BTR 寄存器用于读操作，FSMC_BWR 寄存器用于写操作。

当使用同步存储器时，用户必须计算并设置如下参数：

- CLKDIV：时钟分频比
- DATLAT：数据延迟

请注意，若 NOR Flash 支持此模式，则读操作可为同步，而写操作一般为异步。

当编程同步 NOR Flash 时，该存储器会自动在同步和异步模式间切换，因此在此情况下，必须正确设置所有参数。

图 1 和图 2 显示了在典型 NOR Flash 访问期间的不同时序。

图 1. 异步 NOR Flash 读访问时序

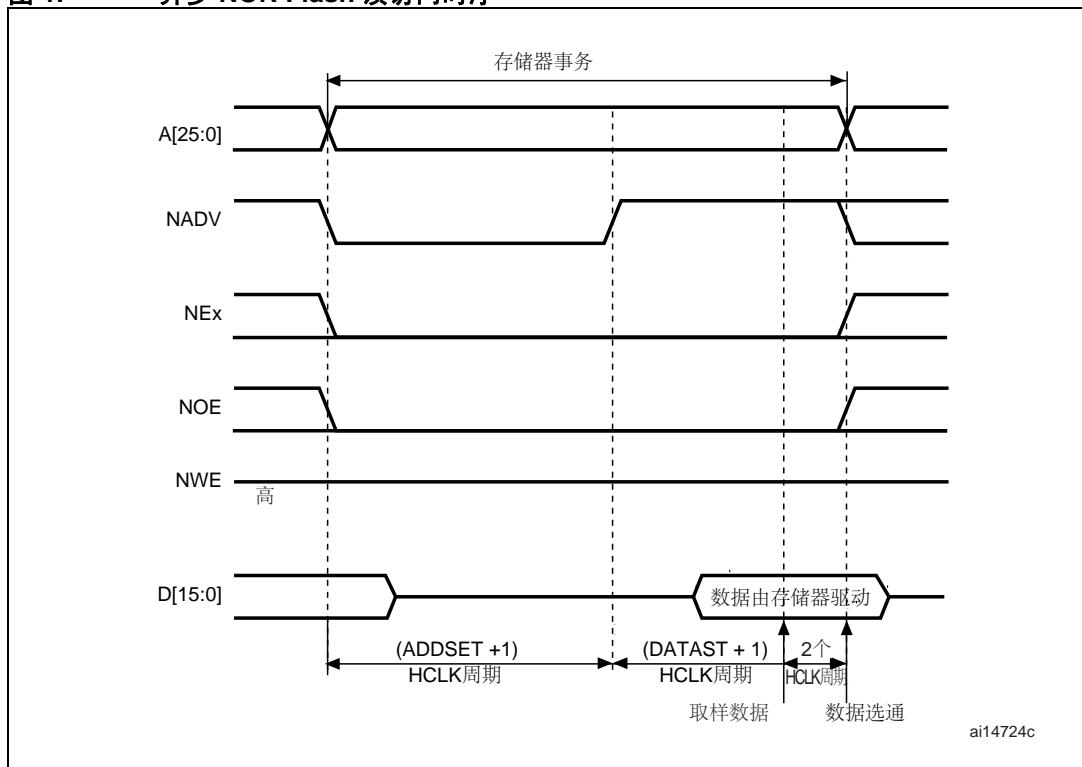
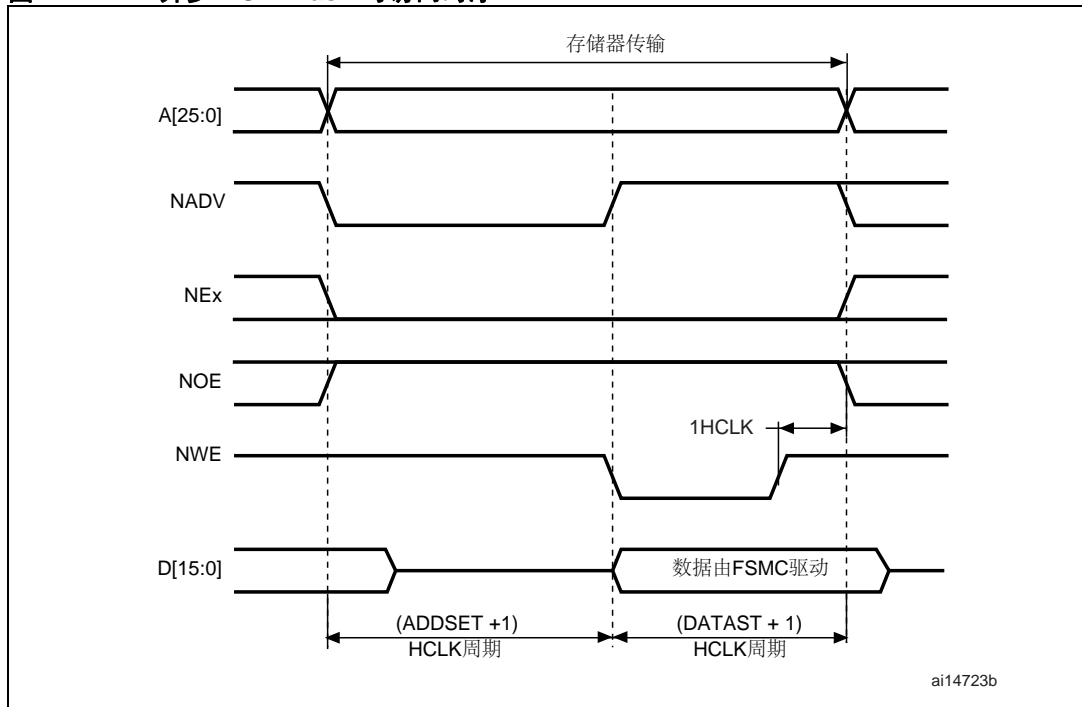


图 2. 异步 NOR Flash 写访问时序



2 TFT LCD（薄膜晶体管液晶显示器）与 FSMC 接口

2.1 常用的彩色 LCD 接口

通常使用硅镶嵌玻璃 LCD 驱动器控制点阵 LCD 单元，可使用串行接口将其与微控制器接口。此类 LCD 单元有嵌入式 RAM 用于显示，不需要任何特殊的微控制器特性。

与点阵 LCD 单元不同，彩色 LCD 需要专门的控制器。

使用彩色 LCD 控制器意味着每个像素需要四条线：三条模拟电压线用于红色、绿色、蓝色（水平方向），一条电压线用于选择（垂直方向）。

一般情况下，可用两种方法与彩色 LCD 接口：

- RGB 接口，使用同步信号
- MPU（微处理器单元）接口（并行或串行）

所有图形 LCD 都有内置驱动器，其形式为芯片或硅玻璃。这些驱动器在 RGB 信号和同步与像素控制之间进行转换。

很多 LCD 还带有 LCD 控制器，进行 MPU 接口与 RGB 信号之间的转换。一些芯片既是驱动器，又是控制器。控制器的作用是持续刷新 LCD。

表 1 列出了集成了控制器的 LCD 与需要 MPU LCD 控制器的 LCD 之间的区别。

表 1. MCU 中 LCD 控制器的优点和缺点

MCU 中的 LCD 控制器	LCD 中的 LCD 控制器
需要外部 RAM（成本）	LCD 控制器中包含了 RAM
连续刷新 LCD 会导致高功耗及高 MCU 使用带宽	仅当 LCD 显示发生变化时 MCU 接口才激活。
适合菜单和界面，不适合电影、游戏、高端 OS（需要 MPU 架构）。	适合菜单和界面，不适合电影、游戏、高端 OS。
没有 MCU 实证（在实际产品中仅有带 LCD 的 MPU 设计）。	节省成本的 QVGA（240 × 320 像素）及以下。在电器及工业市场中广泛使用。

最常见的 LCD MPU 并行接口类型为 Intel 8080（I80）和 Motorola 6800（M68）。

下一节将重点讲述怎样将 LCD Intel 8080 类及 Motorola 6800 类的接口与 STM32F10xxx FSMC 进行连接。

LCD 控制器信号分为两类：数据信号和控制信号。

数据信号连接至 LCD 数据总线，取决于 LCD 色深（8 位、9 位、16 位、18 位或 24 位（真彩色））。

控制信号用于定义操作类型（读或写），以及操作是否包括寻址（写入指令到）LCD 寄存器或显示 RAM。

表2 说明了 LCD Intel 8080 类及 Motorola 6800 接口的控制信号。

表 2. 并行接口读 / 写状态

控制	6800 处理器		8080 处理器		功能
	E	R/W	RD	RW	
RS	E	R/W	RD	RW	
0	1	1	0	1	读寄存器（状态）
0	1	0	1	0	向寄存器写入指令
1	1	1	0	1	读显示数据（RAM）
1	1	0	1	0	写显示数据（RAM）

2.2 FSMC 与 LCD 模块接口的典型用法

STM32F10xxx FSMC 有四个不同的 64 MB 存储区域，以支持 NOR Flash/PSRAM 及类似的外部存储器。

外部存储器与控制器共享地址、数据和控制信号。

每个外部器件通过独有的片选信号访问，但 FSMC 在同一时刻只能访问一个外部器件。每个存储区域都通过专用寄存器配置，包括不同的特性及时间参数。

我们在前面已经看到，FSMC 提供了 LCD 控制器所需的所有信号。用于 LCD 接口的 FSMC 信号说明如下：

- FSMC [D0:D15]：FSMC 数据总线：16 位宽
- FSMC NEx：FSMC 片选
- FSMC NOE：FSMC 输出使能
- FSMC NWE：FSMC 写使能
- FSMC Ax 用于在 LCD 寄存器与 LCD 显示器 RAM 之间选择的一根地址线，其中 x 可为 0 到 25

注：信号名称中的前缀“N”表示信号为低电平有效。

LCD 地址取决于所用的 FSMC NOR Flash/PSRAM 存储区域（NEx）以及驱动 LCD RS 引脚所选的地址（Ax）。

例如：对于 NE2 和 A4，LCD 基址为 0x6400 0000 和 0x6400 0020；对于 NE4 和 A0，LCD 基址为 0x6C00 0000 和 0x6C00 0002。

2.3 将 FSMC 连接至 LCD Intel 8080（I80）接口

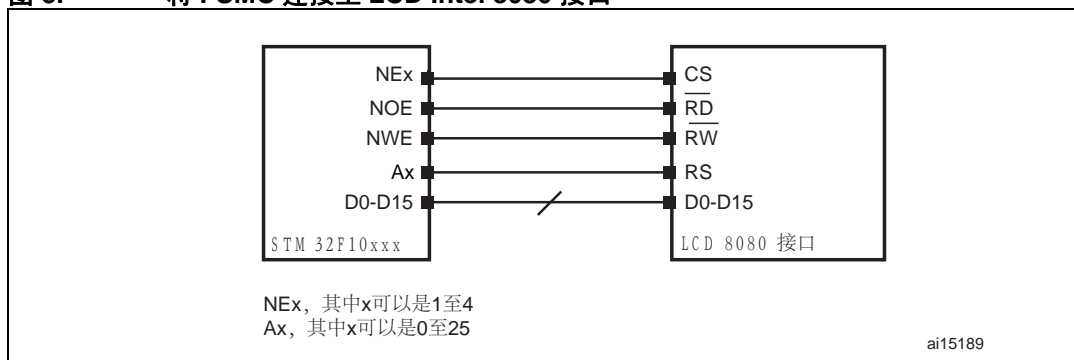
LCD Intel 8080 类 MPU 接口基于四个控制信号和一个数据总线，带宽可变，取决于 MPU 接口能力。表3 说明了这些信号。

表 3. LCD Intel 8080 接口引脚

LCD 信号	信号说明
RS	LCD 寄存器选择
D0-D15	数据 D0-D15
$\overline{\text{CS}}$	片选
$\overline{\text{RD}}$	读操作：低电平有效
$\overline{\text{WR}}$	写操作：低电平有效

图 3 中显示了 FSMC 与 LCD Intel 8080 间的典型连接

图 3. 将 FSMC 连接至 LCD Intel 8080 接口



2.4 将 FSMC 连接至 LCD Motorola 6800（M68）接口

LCD Motorola 6800 类 MPU 接口基于四个控制信号和一个数据总线，带宽可变，取决于 MPU 接口能力。图 4 说明了这些信号。

表 4. FSMC 和 LCD Intel 6800 引脚

LCD 信号	信号说明
RS	LCD 寄存器选择
D0-D15	数据 D0-D15
$\overline{\text{CS}}$	片选
E	启用 / 禁用读写
$\overline{\text{R/W}}$	读操作 / 写操作

图 4 显示了 FSMC 与 LCD Intel 6800 接口间的典型连接。

图 4. 将 FSMC 连接至 LCD Motorola 6800 接口

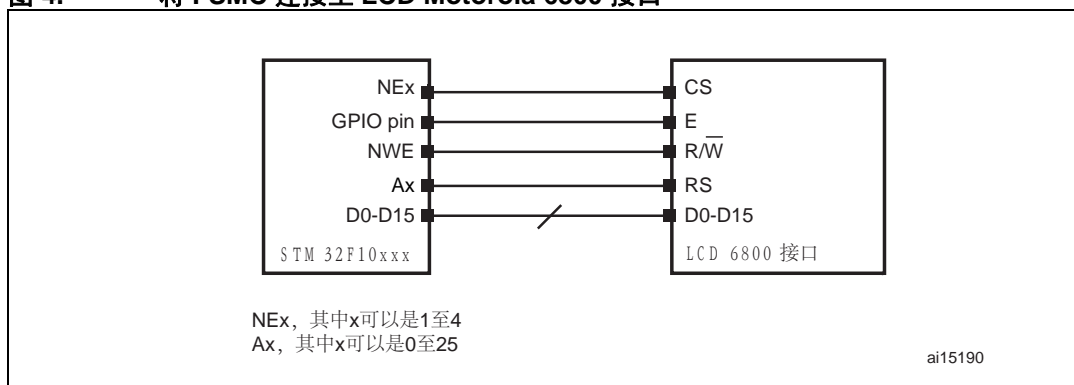
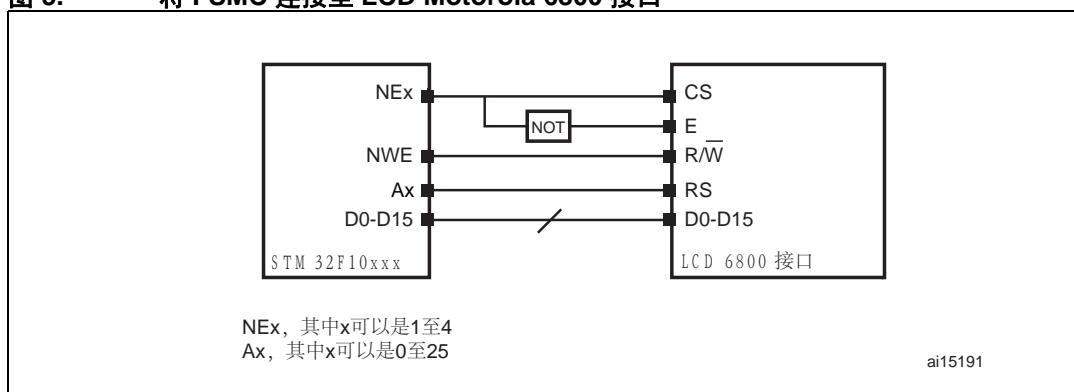


图 5 显示了 LCD Motorola 6800 接口与 FSMC 的另一种连接。

图 5. 将 FSMC 连接至 LCD Motorola 6800 接口 (1)



1. 很多 LCD 有两个互补片选引脚。

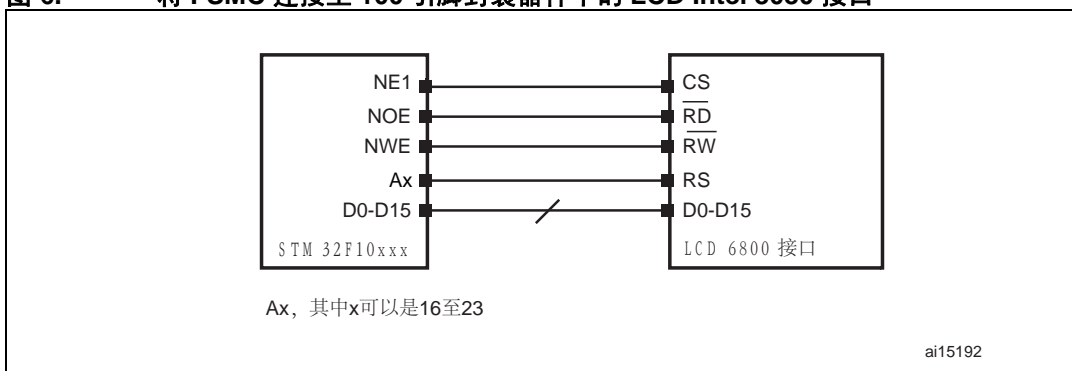
2.5 LCD 与 100 引脚封装的大容量 STM32F10xxx 器件 FSMC 的接口

STM32F101VC/D/E 和 STM32F103VC/D/E 为 100 引脚封装的大容量器件。它们有更少的 FSMC 信号。图 6 给出了彩色 LCD 与 100 引脚封装器件上 FSMC 接口的概览。

此封装中可用于 LCD 接口的 NOR Flash/SRAM FSMC 信号列举如下：

- FSMC D0-D15: FSMC 数据总线: 16 位宽
- FSMC NE1: FSMC 片选: 仅能使用 NOR Flash/SRAM Bank1
- FSMC NOE: FSMC 输出使能
- FSMC NWE: FSMC 写使能
- FSMC Ax 用于在 LCD 寄存器与 LCD 显示器 RAM 之间选择的一根地址线, 其中 x 可为 16 到 23

图 6. 将 FSMC 连接至 100 引脚封装器件中的 LCD Intel 8080 接口



2.6 时序计算

如上所述，对于异步 NOR Flash 类存储器，可能有不同的访问协议。首先，需要定义用户 LCD 使用哪种协议。选择取决于不同的控制信号以及 LCD 在读写传输中的行为。

2.6.1 LCD 时序

表 5 给出了主要彩色 LCD 时间列表，用于计算相应的 FSMC 时序。

表 5. LCD 8080 接口时序特性

符号	参数	条件
t_{AH}	地址保持时间	RS
t_{AS}	地址建立时间	RS
t_{CYC}	系统周期时间	-
$t_{CYC(READ)}$	系统周期时间（读）	-
t_{WRLW}	写操作的低脉冲宽度	\overline{WR}
t_{WRLR}	读操作的低脉冲宽度	\overline{RD}
t_{WRHW}	写操作的高脉冲宽度	\overline{WR}
t_{WRHR}	读操作的高脉冲宽度	\overline{RD}
t_{DS}	数据建立时间	D0-D15
t_{DH}	数据保持时间	D0-D15
t_{ACC}	数据访问时间	D0-D15（此时间取决于负载电容 C_L ）
t_{OD}	输出禁止时间	-

2.6.2 FSMC 时序

通常，LCD 接口使用 ModeB（因此不使用扩展模式），读写操作使用相同的时间。

在此情况下，FSMC 需要三个时间参数：ADDSET、DATAST 和 ADDHOLD。

这些参数根据 LCD 特性计算，为 STM32F10xxx AHB 时钟（HCLK）的函数。

基于图 1 和图 2 中显示的 NOR Flash/SRAM 读写访问时序图，有下列等式：

写或读访问的时间为 LCD 片选信号下降沿和上升沿之间的时间。使用 FSMC 时间参数的函数计算此时间：

$$\text{写 / 读访问} = ((\text{ADDSET} + 1) + (\text{DATAST} + 1)) \times \text{HCLK} = t_{\text{CYC}}$$

DATAST 参数在写操作中写信号的下降沿和上升沿之间测量：

$$\text{写使能信号低电平到高电平} = \text{DATAST} \times \text{HCLK}$$

为确保 FSMC 的正确时间配置，必须考虑下列时间：

- 最大写 / 读访问时间
- 不同的内部存储器延迟。

因此，我们有如下等式：

$$((\text{ADDSET} + 1) + (\text{DATAST} + 1)) \times \text{HCLK} = \max(t_{\text{CYC}}, t_{\text{CYC}(\text{READ})})$$

$$\text{DATAST} \times \text{HCLK} = t_{\text{WRLW}}$$

DATAST 必须为：

$$\text{DATAST} = (((t_{\text{ACC}} + t_{\text{AS}}) + (t_{\text{su(Data_NE)}} + t_{\text{v(A_NE)}})) / \text{HCLK}) - \text{ADDSET} - 4$$

其中：

- $t_{\text{su(Data_NE)}}$: FSMC_NEx 低电平到数据有效
- $t_{\text{v(A_NE)}}$: FSMC_NEx 低电平到 FSMC_A 有效
- $(t_{\text{su(Data_NE)}} + t_{\text{v(A_NE)}}) = 36 \text{ ns}$

注： 当选择 FSMC 时间模式时，请确保 NOE 行为与所用的 LCD 需求相符。请参考 STM32F10xxx 参考手册中的 NOR Flash/SRAM FSMC 时序图说明。

若需灵活的静态存储控制器（FSMC）时间值的更详细信息，请参考 STM32F101xC/D/E 和 STM32F103xC/D/E 大容量数据手册。

3 Ampire AM-240320L8TNQW00H TFT LCD 与 FSMC 的接口

3.1 Ampire TFT LCD 接口

AM-240320L8TNQW00H TFT LCD 集成了一个嵌入式的 LCD 控制器——ILI9320。可将它作为 Intel 8080 或 Motorola 6800 接口。它的色深为 18 位或 16 位，分辨率为 320 × 240 像素。

在本应用笔记中，FSMC 与 Ampire TFT LCD 的接口配置如下：

选择 FSMC Bank1 NOR/PSRAM 4 支持 LCD 器件：

- 存储器类型为 SRAM
- 数据总线宽度为 16 位
- 非复用存储器

所有其它参数必须保持清空。

3.2 硬件连接

通过 FSMC bank1 NOR/PSRAM 4 将 Ampire TFT LCD（ILI9320 控制器）连至 STM32F10xxx FSMC 以控制 LCD 信号。

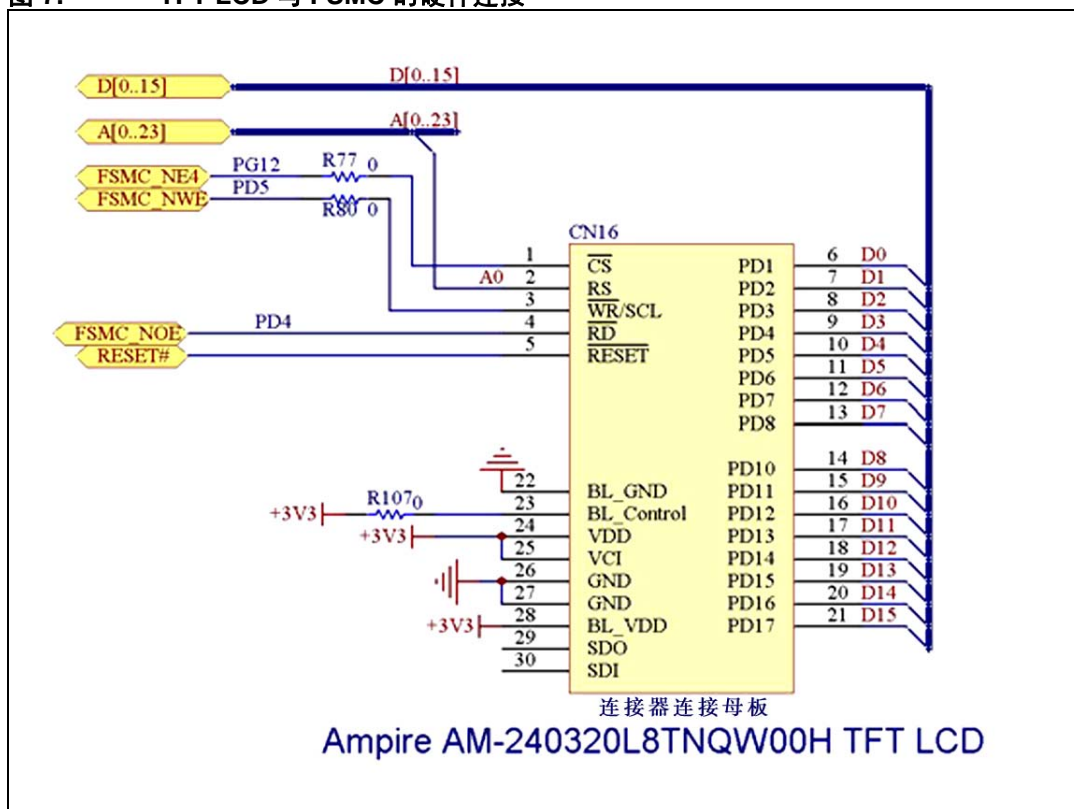
表 6 显示了 LCD 模块引脚和 FSMC 引脚间的对应关系。所有 FSMC GPIO 都配置为复用功能推挽。

表 6. FSMC 和 Ampire LCD 引脚

LCD 信号	FSMC 信号	引脚 / 端口分配	信号说明
RS	A0	PF00	地址 A0
D0-D15	D0-D15	GPIOD/GPIOE	数据 D0-D15
$\overline{\text{CS}}$	NE4	PG12	芯片使能
$\overline{\text{RD}}$ (E)	NOE	PD04	输出使能
$\overline{\text{WR}}$ (R/W)	NWE	PD05	写入使能

图 7 显示了 STM32F10xxx 微控制器和 Ampire LCD（ILI9320 控制器）间的典型连接。此连接为从 STM3210E-EVAL 评估板原理图提取。

图 7. TFT LCD 与 FSMC 的硬件连接



3.3 时间计算

表7给出了 Ampire AM-240320L8TNQW00H 彩色 LCD 时间列表，用于计算相应的FSMC时间。

表 7. LCD 8080 接口时序特性

符号	参数	数值
t_{AH}	地址保持时间	5 ns
t_{AS}	地址建立时间	5 ns 读, 10 ns 写
t_{CYC}	系统周期时间	100 ns
$t_{CYC(READ)}$	系统周期时间 (读)	300 ns
t_{WRLW}	写操作的低脉冲宽度	50 ns
t_{WRLR}	读操作的低脉冲宽度	150 ns
t_{WRHW}	写操作的高脉冲宽度	50 ns
t_{WRHR}	读操作的高脉冲宽度	150 ns
t_{DS}	数据建立时间	10 ns
t_{DH}	数据保持时间	15 ns
t_{ACC}	数据访问时间	最大 100 ns

相应的 LCD FSMC 时序

使用上述公式，我们有：

- HCLK 频率为 72 MHz
- FSMC 访问模式为 ModeB

因此，我们有如下等式：

$$((ADDSET + 1) + (DATAST + 1)) \times HCLK = 100 \quad (\text{仅为写周期，因为我们仅写入 LCD})$$

$$DATAST \times 13.8 = 50$$

DATAST 必须为：

$$DATAST = ((t_{ACC} + t_{AS}) + 36) / HCLK - ADDSET - 4$$

因此，我们有如下 FSMC 时间：

- 地址建立时间：0x1
- 地址保持时间：0x0
- 数据建立时间：0x5

4 固件说明

本应用笔记基于：

- STM32F10xxx 固件库
- LCD 驱动固件（提供控制 LCD 和显示应用所需的主要功能）
- 一个特定的固件，用以调用 LCD 驱动函数以及控制和显示所需的其它函数（main.c 和 stm32f10x_it.c 文件）。

用户可使用相同的库和驱动、不同的接口固件 / 硬件生成任何类似的应用。

4.1 LCD 驱动固件说明

用户可通过驱动应用层直接与 LCD 接口。后文总结了驱动函数。[表 8](#)表示了总的驱动文件组织。

表 8. 驱动库说明

文件	说明
fonts.h, lcd.h, lcd.c	- LCD 定义、类型定义、函数原型 - 基本功能（初始化、读、写寄存器，写 LCD RAM，设置显示窗口，写位图图片，显示字符，清除线，清除 LCD 等等）

高级函数

最终应用调用这些函数即可执行所有需要的配置并执行高端功能（LCD 初始化、写文本、显示位图图片、配置所有硬件元件等等）。[表 9](#)显示了这些功能。

表 9. 高级 LCD 驱动函数

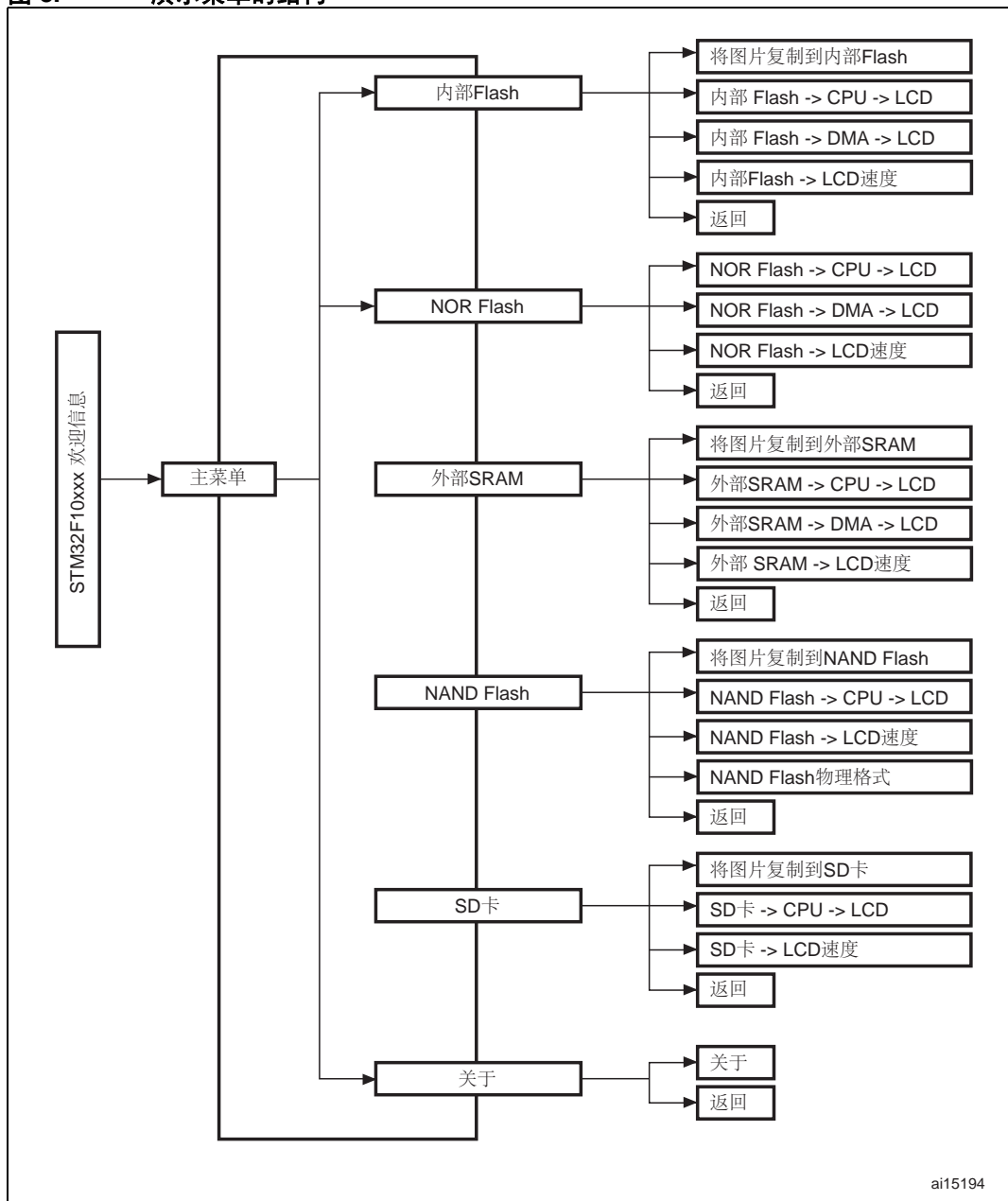
函数名称	说明
STM3210E_LCD_Init	初始化 LCD
LCD_SetTextColor	设置文本颜色
LCD_SetBackColor	设置背景颜色
LCD_ClearLine	清除所选的线
LCD_Clear	清除整个 LCD
LCD_DisplayStringLine	在 LCD 上显示最多 20 个字符
LCD_SetDisplayWindow	设置显示窗口
LCD_WriteBMP	显示加载到内部 Flash 的位图图片

4.2 运行演示

4.2.1 菜单

图 8 显示了 STM32F10xxx LCD 演示的菜单系统。主菜单显示在左手边。用户可使用 UP、DOWN、RIGHT、LEFT 摇杆方向导航，以选择主菜单和子菜单中的条目。若需进入子菜单，请按下 SEL 按钮（当按下摇杆按钮后，会出现 SEL 按钮用于切换）。若需退出子菜单，请选择 Return 菜单然后按下 SEL。

图 8. 演示菜单的结构

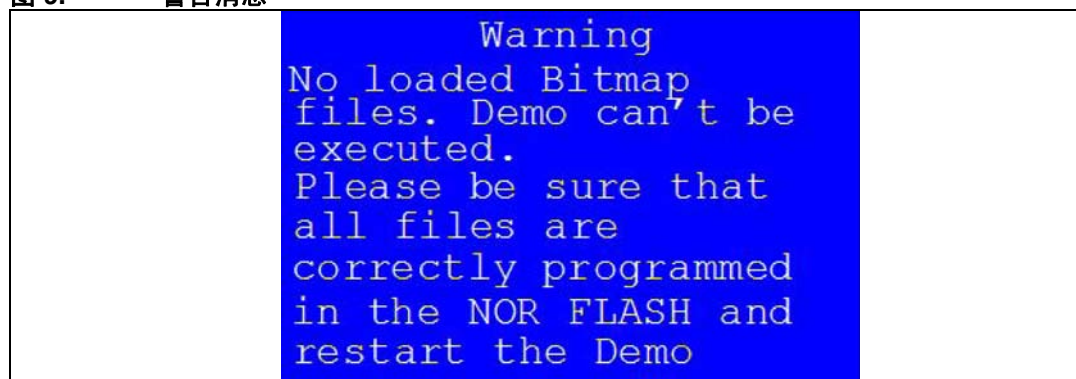


ai15194

4.2.2 启动演示

板子重启之后，在演示开始时，会将图标和位图文件载入 NOR Flash。所有图标都必须正确编程在 NOR Flash 中才能启动演示，因此若有图标缺失，则不能启动演示，LCD 屏幕上会显示图 9 中的消息。

图 9. 警告消息



若图标正确加载到 NOR Flash 中，则 LCD 上会显示欢迎界面及 ST 标识（见图 10）。

图 10. ST 标识



之后，经过 1 秒，LCD 屏幕上会显示图 11 中的 STM32F10xxx 胶片。

图 11. STM32 系列



在 STM32F10xxx 欢迎界面之后，会出现主菜单。它以一组图标的形式显示，在同一屏幕中还会显示所有子菜单。您可使用 UP、DOWN、RIGHT、LEFT 摇杆方向导航，选择需要的子菜单。若需进入子菜单，请按下 SEL 摇杆按钮，则会显示与所选图标相对应的新的子菜单。

图 12. 主菜单



1. 图 12 中显示的图标来自 http://commons.wikimedia.org/wiki/Crystal_Clear。

选择子菜单之后，应用名称会列于显示的顶部，所有相应的子菜单会列于它之下，如图 13 所示。

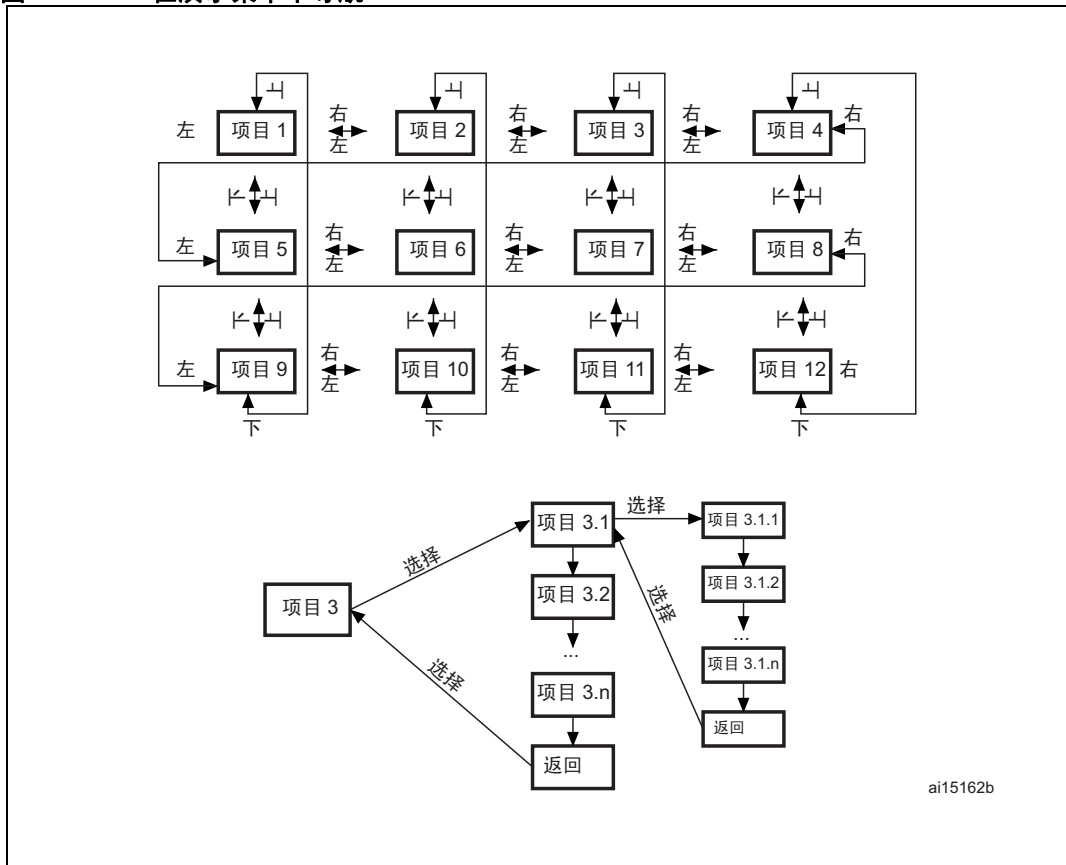
图 13. 相应的子菜单



4.2.3 导航

演示菜单基于方向导航、子菜单选择、条目选择及后退能力，如 [图 14](#) 所示。

图 14. 在演示菜单中导航



用户使用评估板上的 RIGHT、LEFT、SEL、UP、DOWN 摇杆按钮导航。

- UP、DOWN、RIGHT、LEFT 按钮用于在主菜单和当前菜单条目中方向导航
- 按下 SEL 按钮会选择当前条目
- UP 和 DOWN 按钮用于在子菜单中垂直导航
- 若需返回上层菜单，请在 Return 一行按下 SEL

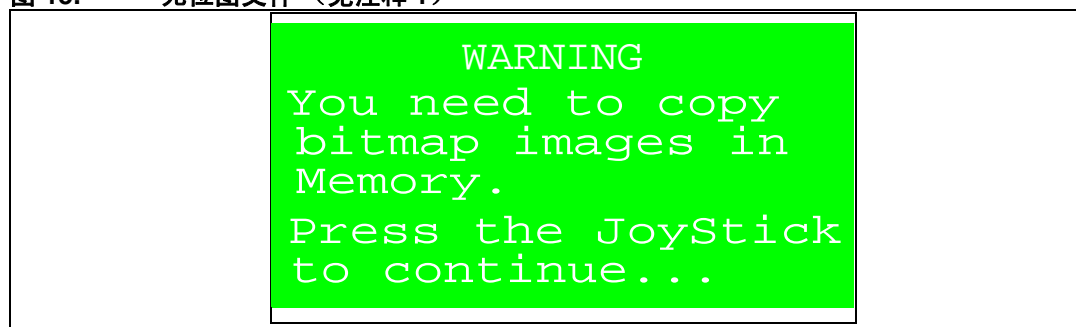
4.3 LCD 演示应用

该 LCD 演示的目的是提供一个完整的样例，从不同存储器位置读取数据并显示在 LCD 上。对于数据传输，我们使用了 CPU 传输和 DMA 传输。对于每种情况，会计算显示时间并显示在 LCD 上。此时间为从所选存储器读取及向 LCD 写入图像的时间之和。

本节提供了每部分演示的详细说明。

- 注：
- 1 在该演示中，内核运行于 $HCLK = 72\text{ MHz}$ 。
 - 2 此演示使用的所有时间都与 STM3210E-EVAL 板存储器有关。若您使用的是不同的存储器，您可获取其它值。STM3210E-EVAL 板具有如下存储器：
 - NOR Flash: Numonyx M29W128F、Numonyx M29W128G、Spansion S29GL128P
 - 外部 SRAM: ISSI IS61WV51216BLL
 - NAND Flash: Numonyx NAND512W3A2CNB
 - SD 卡™: Numonyx SMS128FF
 - 3 所用的位图图片文件属性为：
 - 编码: bitmap
 - 宽: 320
 - 高: 240
 - 每个像素 16 位
 - 图片数据尺寸: 150 KB
- 注：在任何 LCD 演示应用中，若位图图片没有加载到所选存储器（内部 Flash、外部 SRAM）中的保留地址，则 LCD 屏幕上会显示图 15 中的消息。

图 15. 无位图文件（见注释 1）



1. 存储器可为内部 Flash 或外部 SRAM。

4.3.1 内部 Flash

本部分 LCD 演示专门显示存储于内部 Flash 中的位图图片，以及计算出的一帧显示时间。

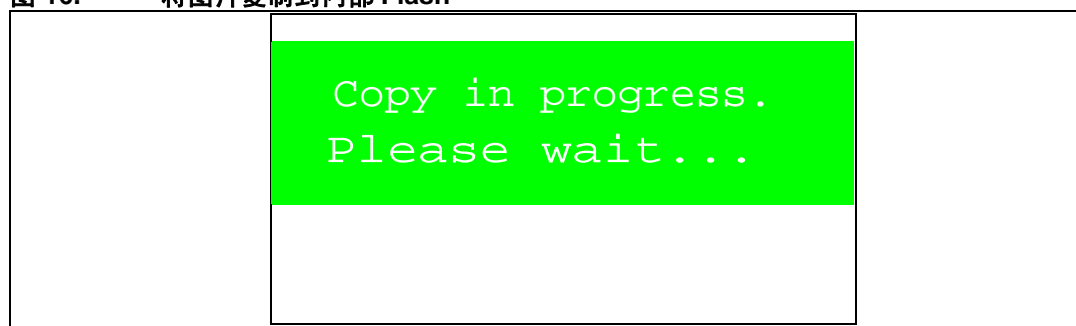
首先需要将相应图片加载到专用的内部 Flash 中。

将图片复制到内部 Flash

该子菜单用于将图片从 NOR Flash 复制到内部 Flash 中的指定地址。

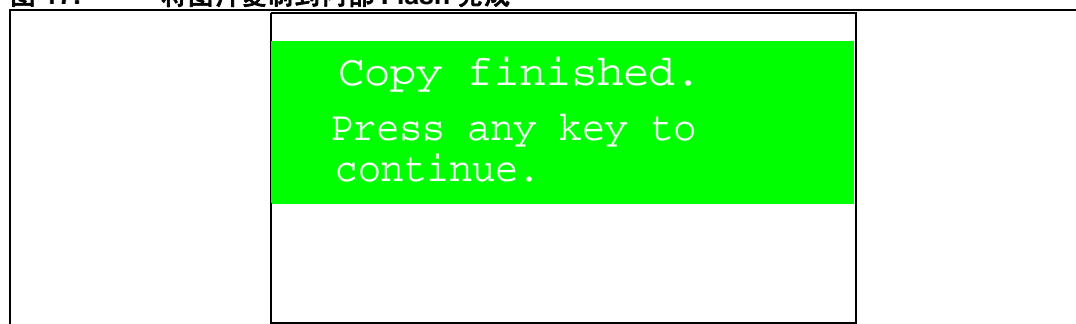
当选择该子菜单时，LCD 上会显示 [图 16](#) 中的消息。

图 16. 将图片复制到内部 Flash



当复制操作完成时，LCD 屏幕上会出现 [图 17](#) 中的消息。

图 17. 将图片复制到内部 Flash 完成



内部 Flash -> CPU -> LCD

该子菜单使用 CPU 从内部 Flash 读取数据图片并将其写入 LCD。两个图片以最大速度显示（无延迟）。

当选择内部 Flash -> CPU -> LCD 菜单时，LCD 屏幕上会显示 [图 18](#) 和 [图 19](#) 中的动画。

图 18. 动画显示 STM32F10xxx 的显示速度

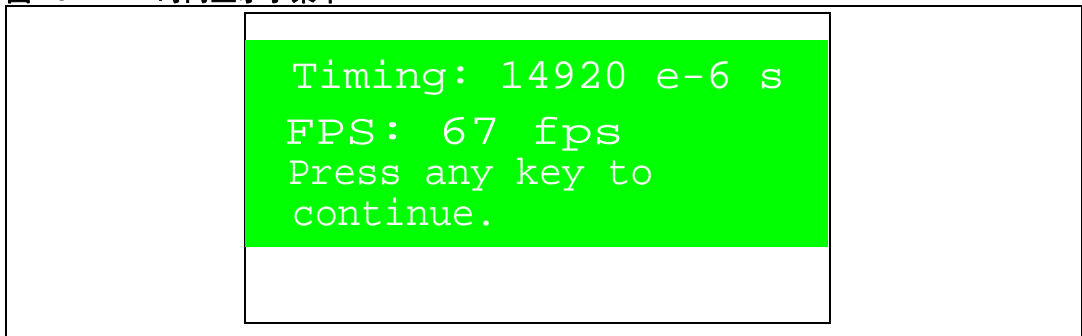


图 19. ST 标识



若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 20 中的子菜单。

图 20. 时间显示子菜单



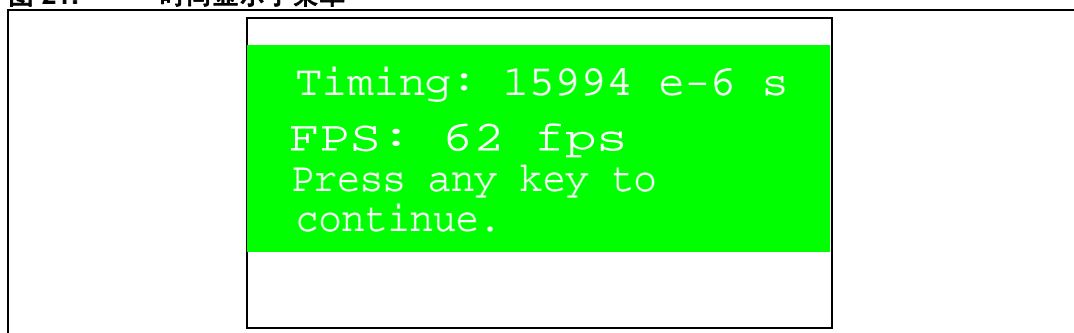
内部 Flash -> DMA -> LCD

该子菜单使用 DMA 从内部 Flash 读取数据图片并将其写入 LCD（0% CPU 负载）。两个图片以最大速度显示（无延迟）。

当选择内部 Flash -> DMA -> LCD 菜单时，LCD 屏幕上会显示图 18 和图 19 中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 20 中的子菜单。

图 21. 时间显示子菜单



内部 Flash -> LCD 速度

该子菜单用于以受控的速度显示从内部 Flash 读取的两个图片。您可通过 RV1 调整，控制两个图片（或帧）的显示速度。两帧之间的最大时延约为 1 秒。该值在 RV1 处于最大位置时得到。

4.3.2 NOR Flash

本部分 LCD 演示专门显示存储于 NOR Flash 中的位图图片，以及计算出的一帧显示时间。

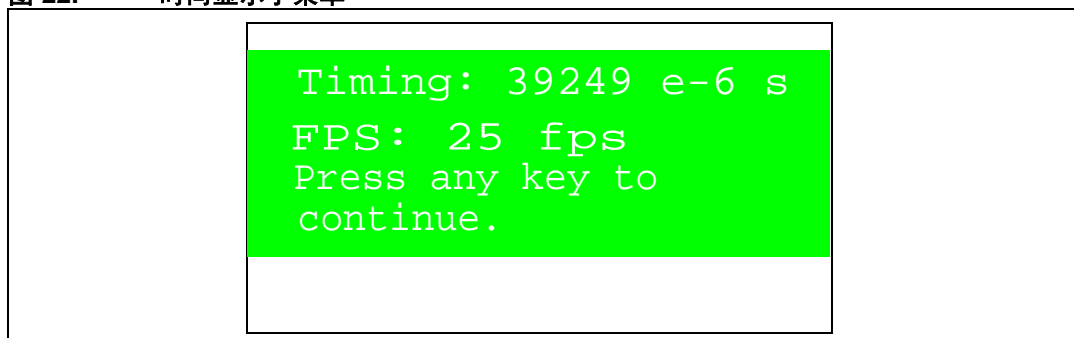
NOR Flash -> CPU -> LCD

该子菜单使用 CPU 从 NOR Flash 读取数据图片并将其写入 LCD。两个图片以最大速度显示（无延迟）。

当选择 NOR Flash -> CPU -> LCD 菜单时，LCD 屏幕上会显示图 18 和图 19 中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 22 中的子菜单。

图 22. 时间显示子菜单

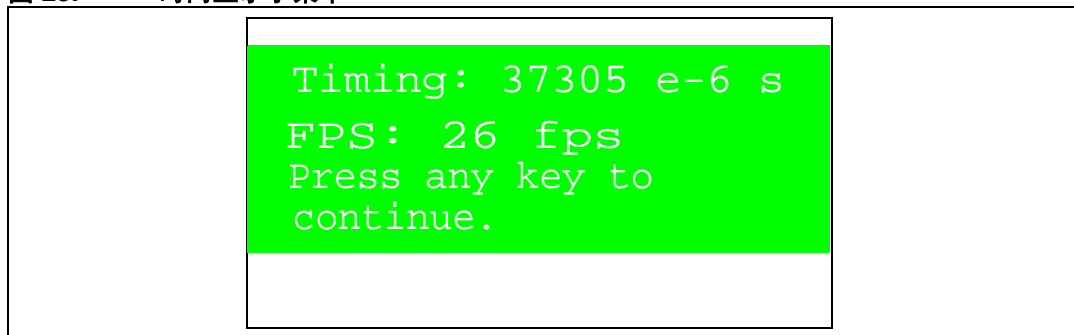


NOR Flash -> DMA -> LCD

该子菜单使用 DMA 从 NOR Flash 读取数据图片并将其写入 LCD（0% CPU 负载）。两个图片以最大速度显示（无延迟）。

当选择 **NOR Flash -> DMA -> LCD** 菜单时，LCD 屏幕上会显示 [图 18](#) 和 [图 19](#) 中的动画。
若您在播放动画时按下了 SEL 按钮，则动画停止并显示 [图 23](#) 中的子菜单。

图 23. 时间显示子菜单



NOR Flash-> LCD 速度

该子菜单用于以受控的速度显示从 NOR Flash 读取的两个图片。您可通过 RV1 调整，控制两个图片（或帧）的显示速度。两帧之间的最大时延约为 1 秒。该值在 RV1 处于最大位置时得到。

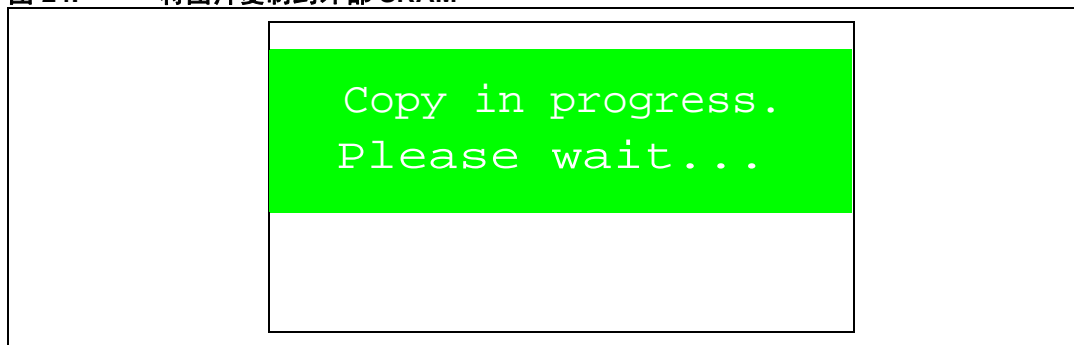
4.3.3 外部 SRAM

本部分 LCD 演示专门显示存储于外部 SRAM 中的位图图片，以及计算出的一帧显示时间。
首先需要将相应图片加载到专用的外部 SRAM 中。

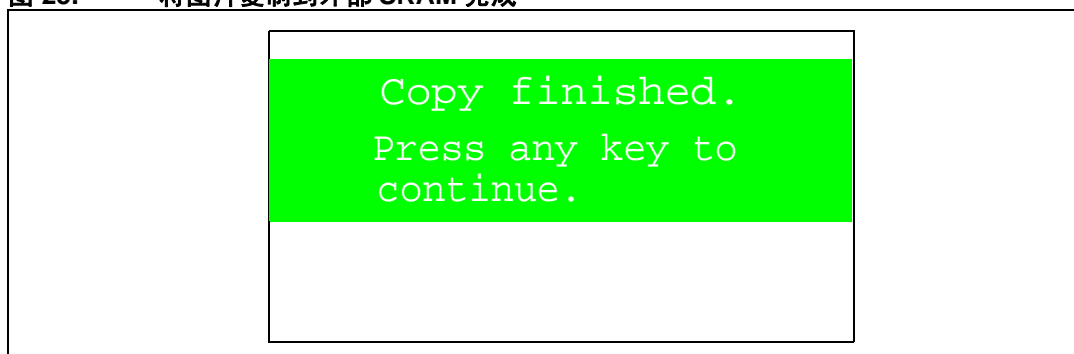
将图片复制到外部 SRAM

该子菜单用于将图片从 NOR Flash 复制到外部 SRAM 中的指定地址。
当选择该子菜单时，LCD 上会显示 [图 24](#) 中的消息。

图 24. 将图片复制到外部 SRAM



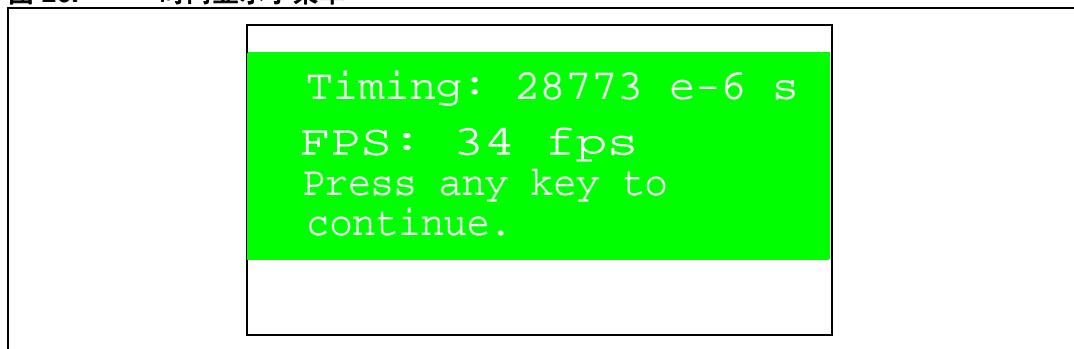
当复制操作完成时，LCD 屏幕上会出现 [图 25](#) 中的消息。

图 25. 将图片复制到外部 SRAM 完成**外部 SRAM-> CPU -> LCD**

该子菜单使用 CPU 从外部 SRAM 读取数据图片并将其写入 LCD。两个图片以最大速度显示（无延迟）。

当选择**外部 SRAM -> CPU -> LCD** 菜单时，LCD 屏幕上会显示图 18 和图 19 中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 26 中的子菜单。

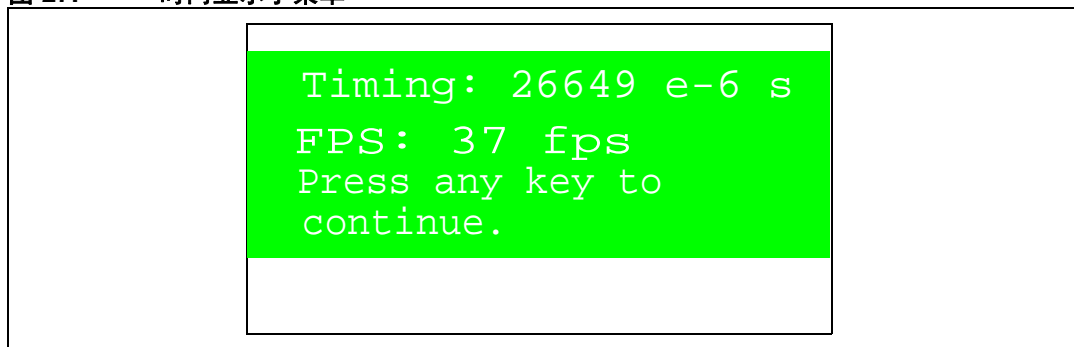
图 26. 时间显示子菜单**外部 SRAM -> DMA -> LCD**

该子菜单使用 DMA 从外部 SRAM 读取数据图片并将其写入 LCD（0% CPU 负载）。两个图片以最大速度显示（无延迟）。

当选择**外部 SRAM -> DMA -> LCD** 菜单时，LCD 屏幕上会显示图 18 和图 19 中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 27 中的子菜单。

图 27. 时间显示子菜单



外部 SRAM-> LCD 速度

该子菜单用于以受控的速度显示从外部 SRAM 读取的两个图片。您可通过 RV1 调整，控制两个图片（或帧）的显示速度。两帧之间的最大时延约为 1 秒。该值在 RV1 处于最大位置时得到。

4.3.4 NAND Flash

本部分 LCD 演示专门显示存储于 NAND Flash 中的位图图片，以及计算出的一帧显示时间。NAND Flash 缓存读取使用内部 16 KB 缓存。

位图图片以 16 KB 读出并写入 LCD。

首先需要将相应图片加载到专用的 NAND Flash 中。

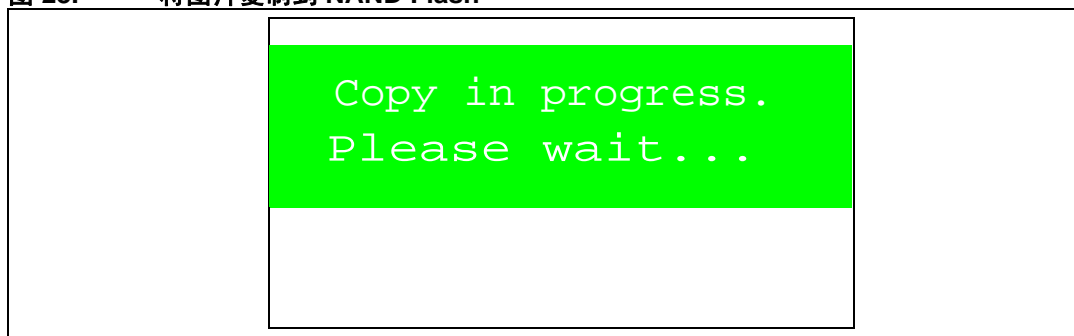
注：若未事先将图片写入 NAND Flash 即进入不同的 NAND Flash 显示菜单，若 NAND Flash 已被擦除则 LCD 屏幕会显示空白图片。

将图片复制到 NAND Flash

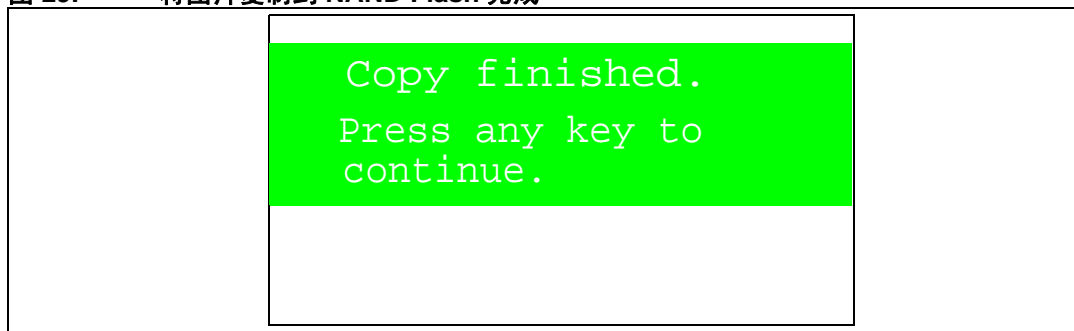
该子菜单用于将图片从 NOR Flash 复制到 NAND Flash 中的指定地址。

当选择该子菜单时，LCD 上会显示图 28 中的消息。

图 28. 将图片复制到 NAND Flash



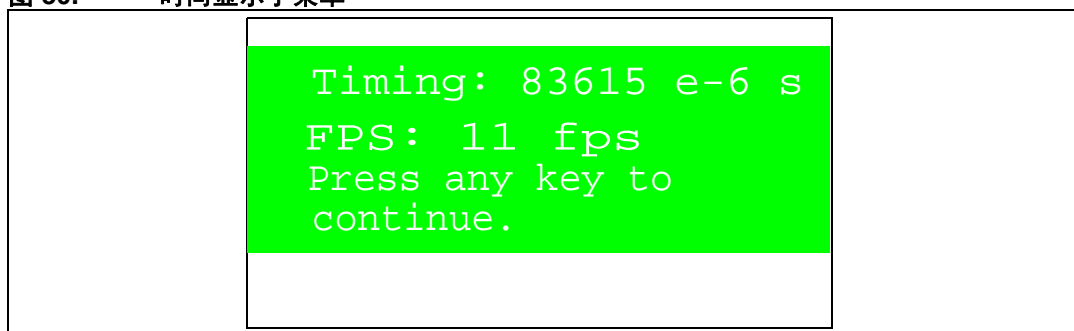
当复制操作完成时，LCD 屏幕上会出现图 29 中的消息。

图 29. 将图片复制到 NAND Flash 完成**NAND Flash -> CPU -> LCD**

该子菜单使用 CPU 从 NAND Flash 读取数据图片并将其写入 LCD。两个图片以最大速度显示（无延迟）。

当选择 **NAND Flash -> CPU -> LCD** 菜单时，LCD 屏幕上会显示 [图 18](#) 和中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示 [图 20](#) 和 [图 19](#) 中的子菜单。

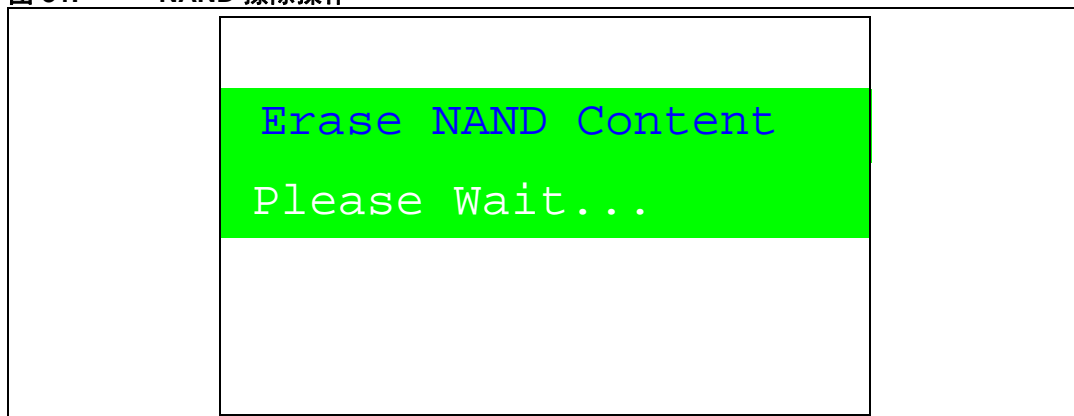
图 30. 时间显示子菜单**NAND Flash-> LCD 速度**

该子菜单用于以受控的速度显示从 NAND Flash 读取的两个图片。您可通过 RV1 调整，控制两个图片（或帧）的显示速度。两帧之间的最大时延约为 1 秒。该值在 RV1 处于最大位置时得到。

NAND Flash 物理格式化

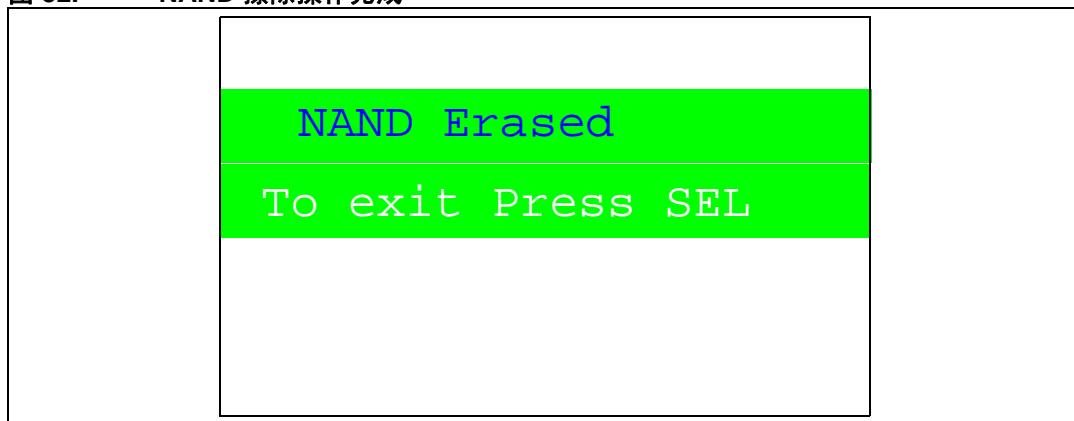
该子菜单用于擦除所有 NAND Flash 内容。当选择该子菜单时，LCD 上会显示 [图 31](#) 中的消息。

图 31. NAND 擦除操作



当 NAND 擦除操作完成时，LCD 上会显示图 32 中的消息。

图 32. NAND 擦除操作完成



若需退出该子菜单并返回到 **NAND 物理格式化**子菜单，请按 SEL。

4.3.5 SD 卡

本部分 LCD 演示专门显示存储于 SD 卡中的位图图片，以及计算出的一帧显示时间。SD 卡缓存读取使用内部 10 KB 缓存。以一次 10 KB 读取位图图片并写入 LCD。

首先需要将相应图片加载到专用的 SD 卡中。

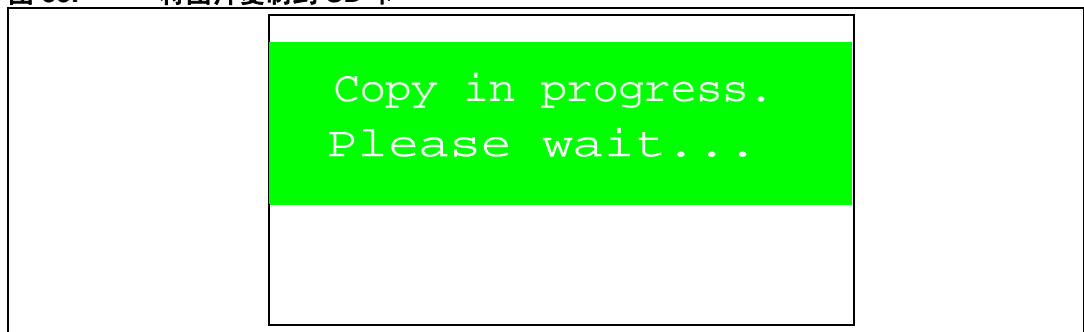
注： 若未事先将图片写入 SD 卡即进入 SD 卡显示菜单，若 SD 卡已被擦除则 LCD 屏幕会显示空白图片。

将图片复制到 SD 卡

该子菜单用于将图片从 NOR Flash 复制到 SD 卡中的指定地址。

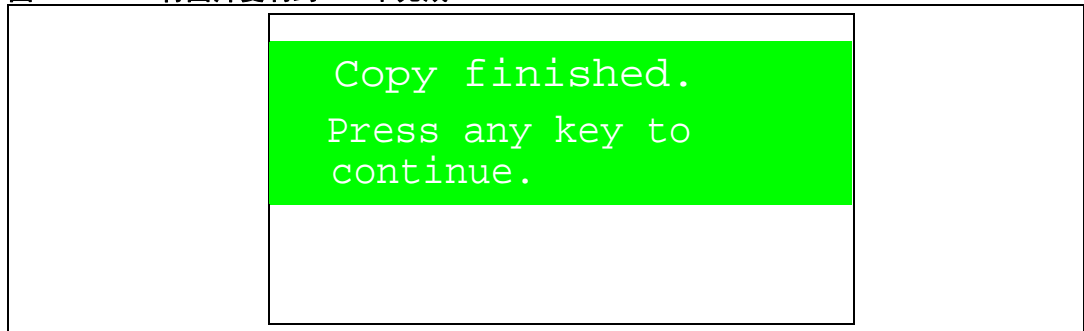
当选择该子菜单时，LCD 上会显示图 33 中的消息。

图 33. 将图片复制到 SD 卡



当复制操作完成时，LCD 屏幕上会显示图 34 中的消息。

图 34. 将图片复制到 SD 卡完成



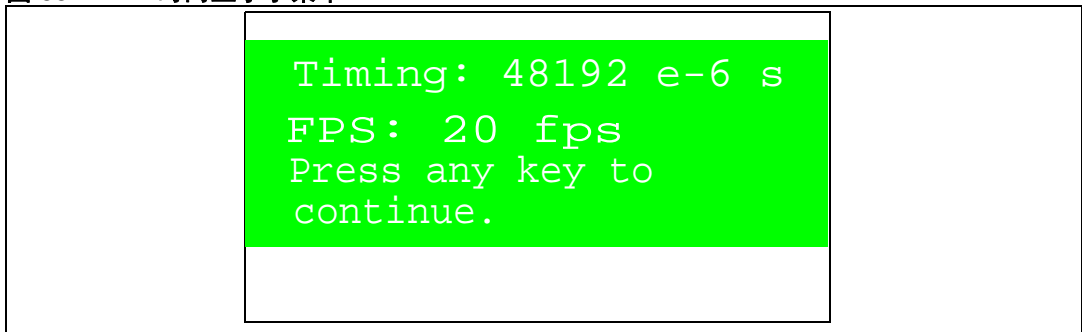
SD 卡 -> CPU -> LCD

该子菜单使用 CPU 从 SD 卡读取数据图片并将其写入 LCD。两个图片以最大速度显示（无延迟）。

当选择 **SD 卡 -> CPU -> LCD** 菜单时，LCD 屏幕上会显示图 18 和图 19 中的动画。

若您在播放动画时按下了 SEL 按钮，则动画停止并显示图 35 中的子菜单。

图 35. 时间显示子菜单



SD 卡 -> LCD 速度

该子菜单用于以受控的速度显示从 SD 卡读取的两个图片。您可通过 RV1 调整，控制两个图片（或帧）的显示速度。两帧之间的最大时延约为 1 秒。该值在 RV1 处于最大位置时得到。

5 彩色 TFT LCD 与 FSMC 接口：性能

表 10 给出了位图图片在不同存储器时的显示速度。

表 10. LCD 显示速度^{(1) (2)}

存储器	帧大小	分辨率	色深	无 DMA 时的显示时间	有 DMA 时的显示时间	每秒帧数 (FPS)
内部 Flash	150 KB	320 × 240	16 位	14.920 ms	15.994 ms	67 fps / 62 fps
外部 SRAM	150 KB	320 × 240	16 位	28.773 ms	26.649 ms	34 fps / 37 fps
NOR Flash	150 KB	320 × 240	16 位	39.429 ms	37.305 ms	25 fps / 26 fps
NAND Flash	150 KB	320 × 240	16 位	83.615 ms	-	11 fps
SD 卡 (1 位)	150 KB	320 × 240	16 位	81.257 ms	-	12 fps
SD 卡 (4 位)	150 KB	320 × 240	16 位	48.192 ms	-	20 fps

1. CPU 频率为 72 MHz，FSMC 频率为 36 MHz。SDIO 时钟频率为 24 MHz。

2. 时间计算使用 SysTick 定时器，以及 Keil 编译器 RVMDK V3.20 - O3，Thumb2 模式。

在第一种情况下，从内部 Flash 传输，CPU 传输要比 DMA 传输快。这是因为内部 Flash 为代码执行做了优化。然而，通常情况下使用 DMA 传输更好，因为这时 CPU 可被释放出来并行执行其它工作。

注：

若 CPU 从内部 Flash 执行代码，则从内部 Flash 进行的 DMA 传输将会减慢。
对于其它情况（位图图片在外部存储器中），多层总线矩阵会确保 CPU 从内部 Flash 或 RAM 执行代码时的无缝 DMA 传输。

6 LCD 演示编程

本章讲解了怎样使用 DFU（设备固件升级）应用，编程 LCD 演示应用。它还讲解了 STM3210E-EVAL 板所需的文件。

若需节省使用 DFU 编程的时间，请如下进行：

- 在 STM3210E-EVAL 板和您的 PC 之间使用经认证的 USB Hub 2.0 高速接口
- 关闭不用的应用，将 PC RAM 的占用量降到最小
- 使用高性能的 PC

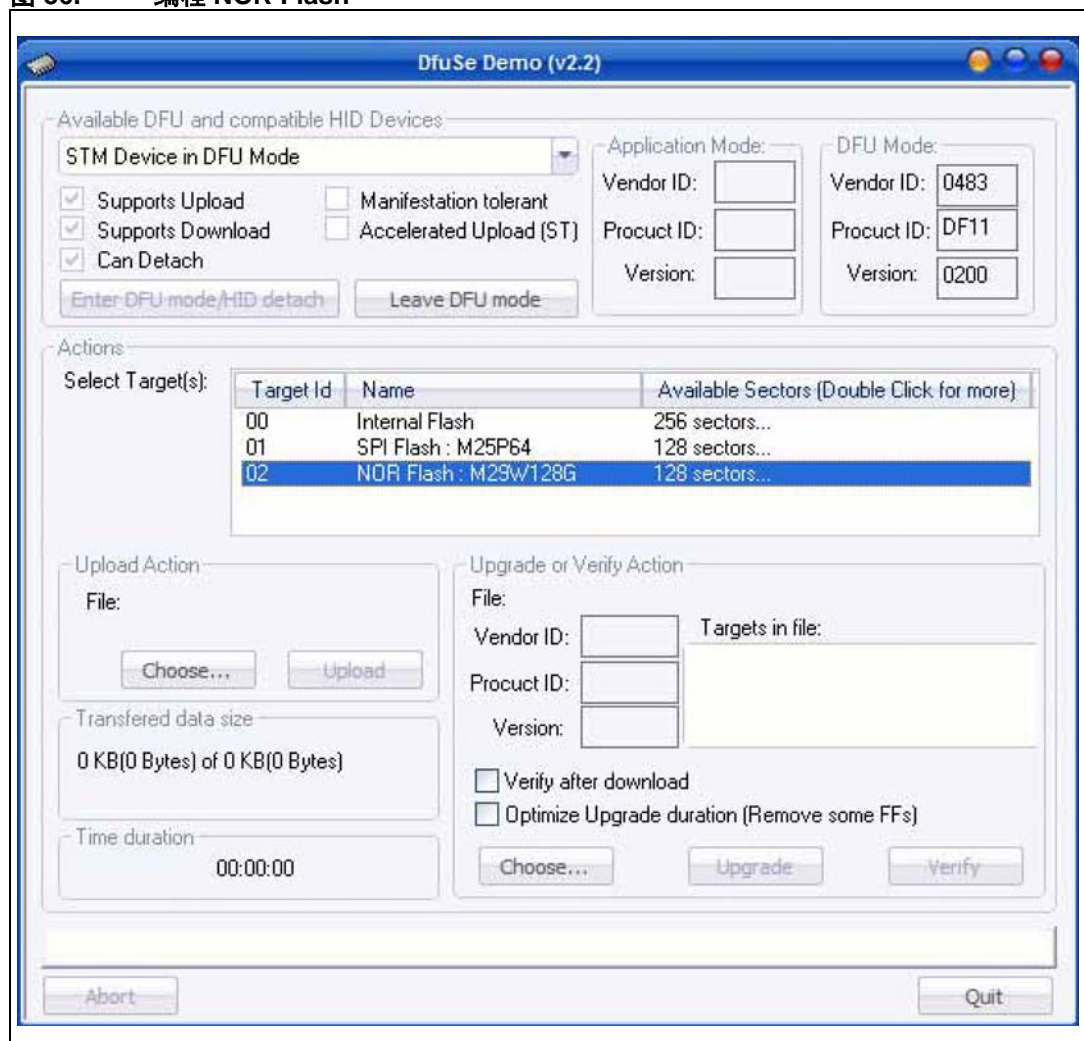
注： 当使用 DFU 应用时，请确保 STM3210E-EVAL 板内的跳线 JP14 位于位置 2-3。

6.1 编程 M29W128 NOR Flash

若需编程 NOR Flash, 请遵循如下步骤:

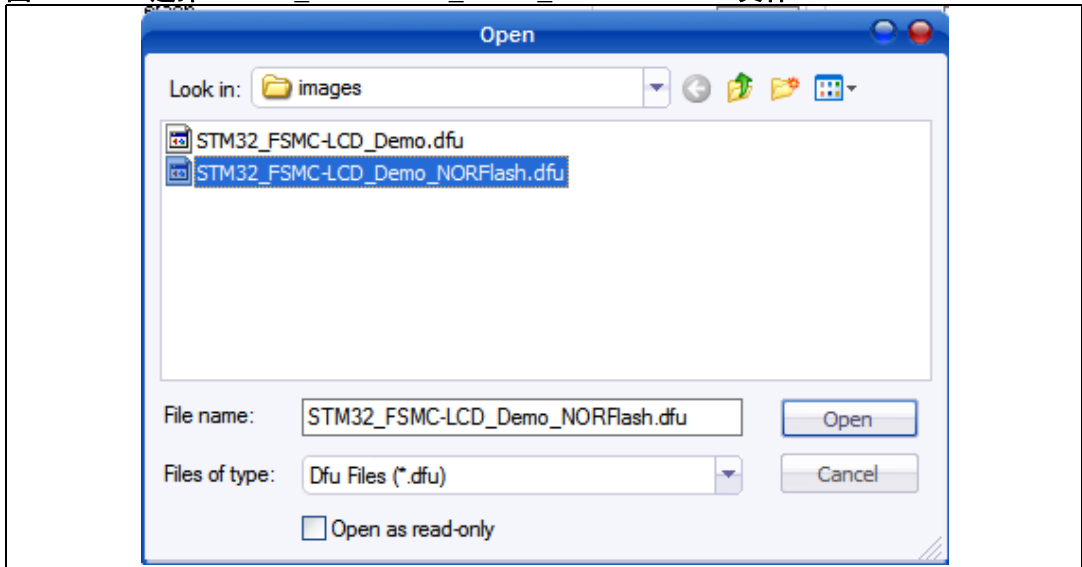
1. 若您是第一次执行此操作, 请使用您首选的工具链加载 DFU 固件项目, 并使用 JTAG 加载 DFU 映像。
2. 使用 USB 线将 STM3210E-EVAL 板连接到 PC。若您是第一次进行此操作, 将会要求您为板子安装驱动。驱动在安装目录中。
若需进入 DFU 模式, 请按下 STM3210E-EVAL 板上的 Key 按钮。
3. 运行 *DfuSeDemo.exe* 程序
4. 图 36 中的窗口出现。在 **Actions** 框内的 **Select Target(s)** 区域, 选择 **NOR Flash: M29W128G**。在 **Upgrade or Verify Action** 框内, 点击 **Choose...** 按钮。

图 36. 编程 NOR Flash



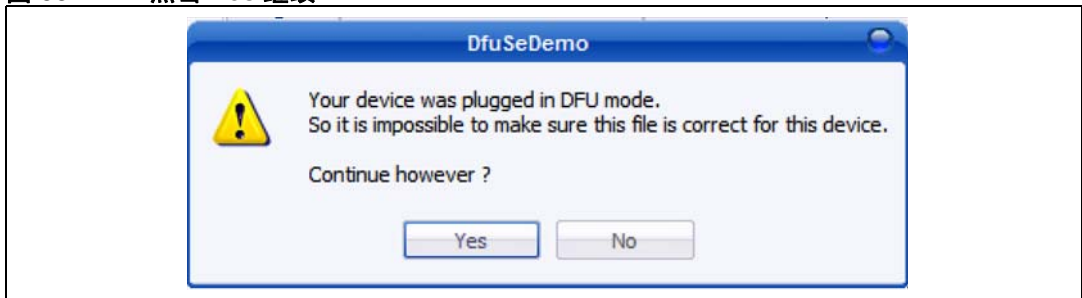
- 5. 这会打开 [图 37](#) 中所示的窗口。选择 `STM32_FSMC-LCD_Demo_NORFlash.dfu` 文件。

图 37. 选择 `STM32_FSMC-LCD_Demo_NORFlash.dfu` 文件



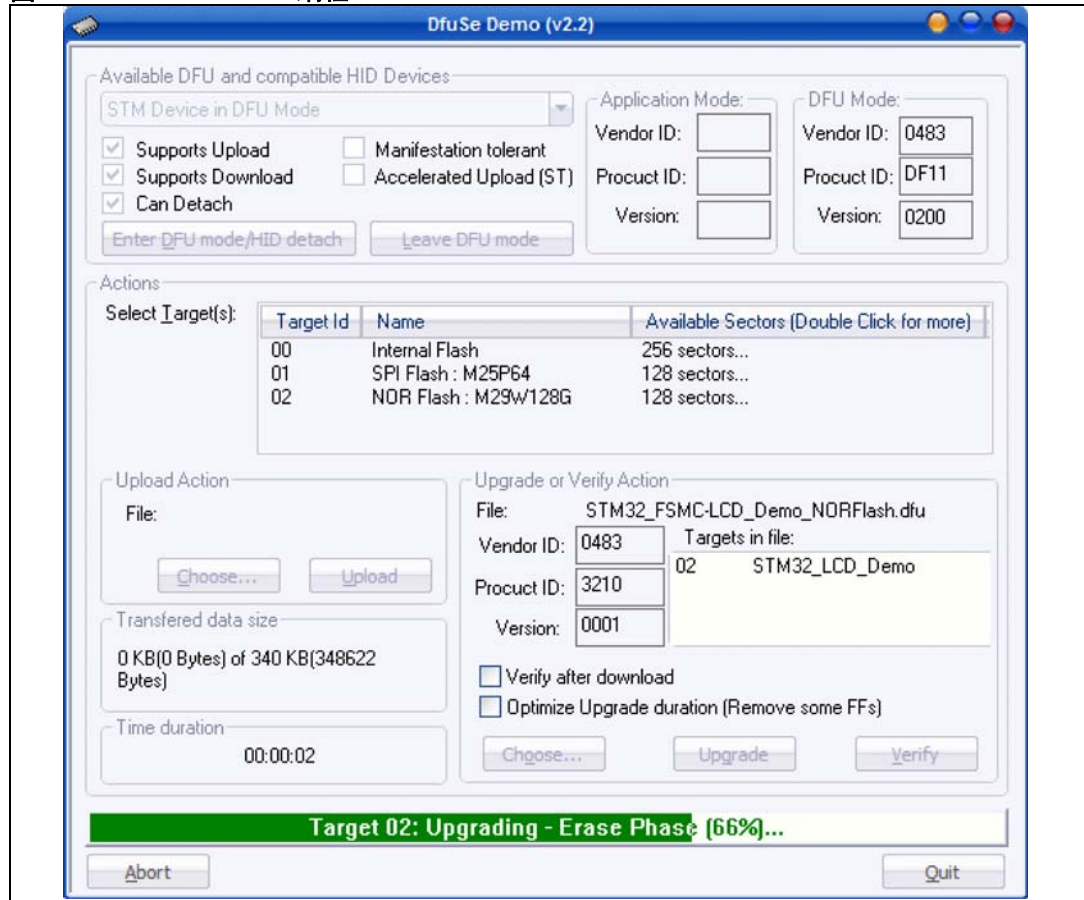
- 6. 然后单击 **Open** 按钮。当之前的窗口再出现时，单击 **Upgrade** 按钮，出现对话框窗口，选择 Yes 继续（见 [图 38](#)）。

图 38. 点击 Yes 继续



7. M29W128 NOR Flash 编程操作开始，如 图 39 中所示。

图 39. NOR Flash 编程

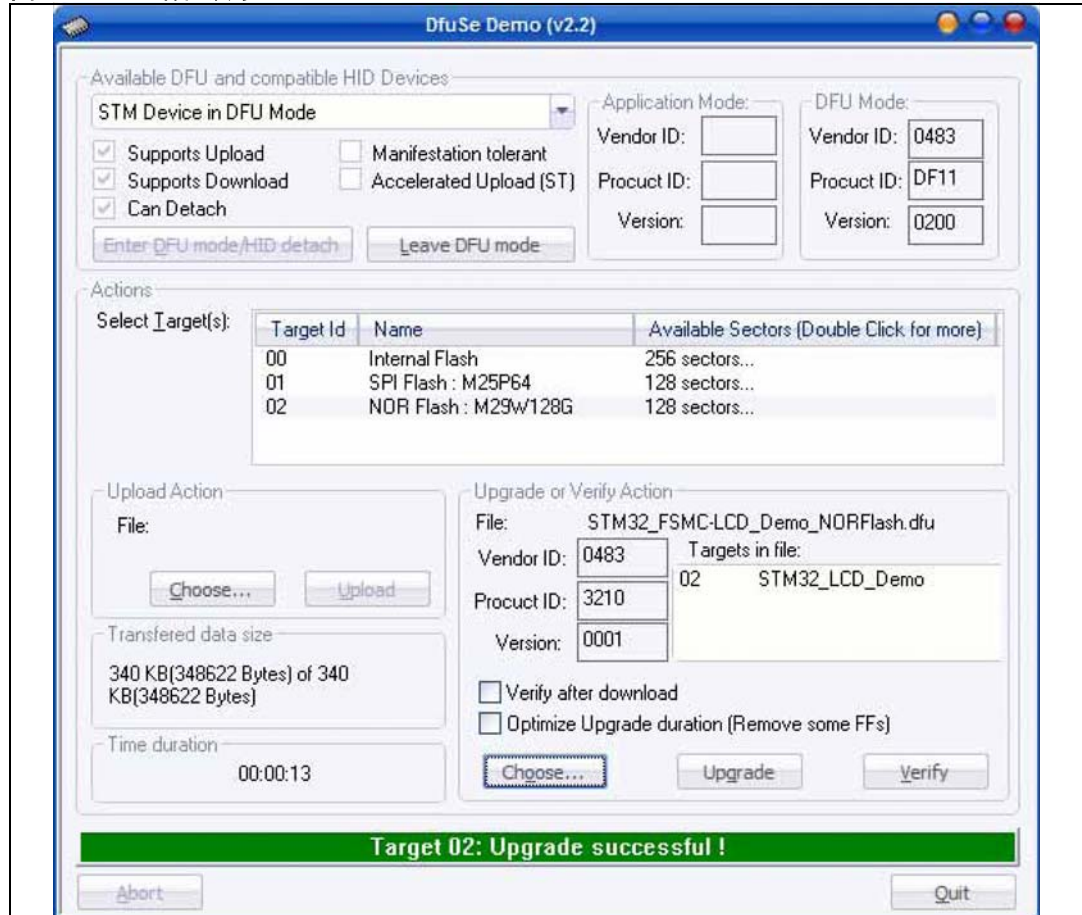


6.2 LCD 演示编程

若需编程内部 Flash 中的 *STM32_FSMC-LCD_Demo.dfu* 文件，请遵循如下步骤：

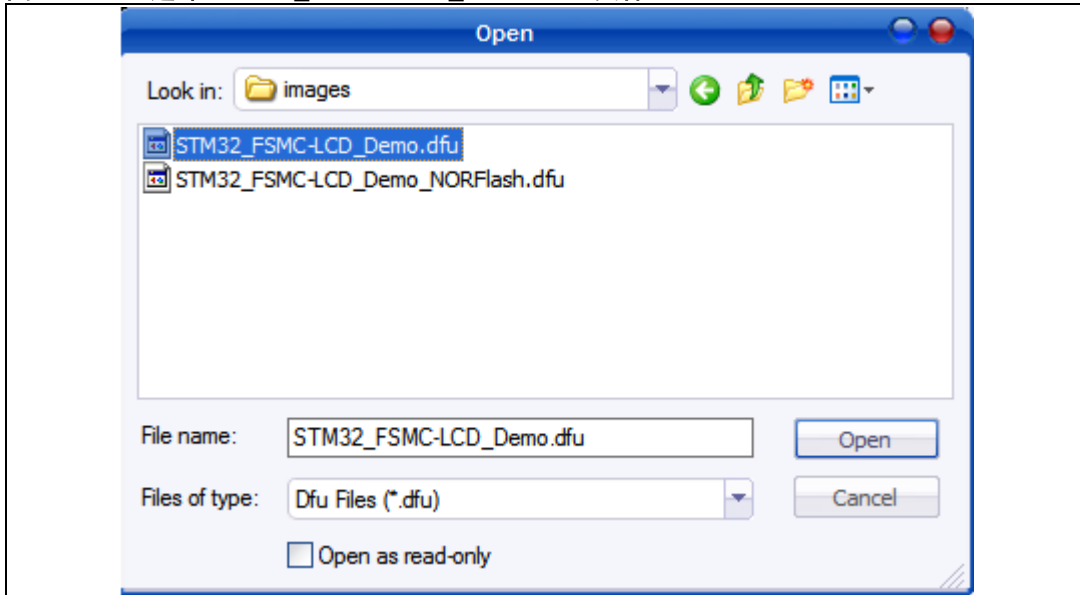
1. 当 NOR Flash 编程结束时，在 **Action** 框内的 **(Select Target(s))** 区域，选择 *Internal Flash*，在 **Upgrade or Verify Action** 框内，点击 **Choose...** 按钮。

图 40. 编程演示



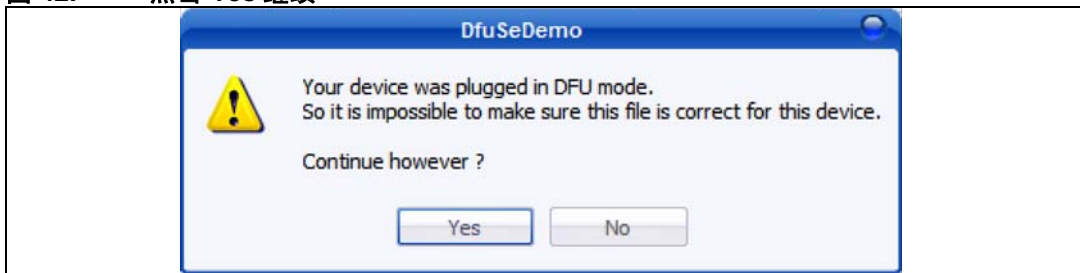
2. 这会打开图 41 中所示的窗口。如图 41 中所示，选择 STM32_FSMC-LCD_Demo.dfu 文件。

图 41. 选择 STM32_FSMC-LCD_Demo.dfu 文件



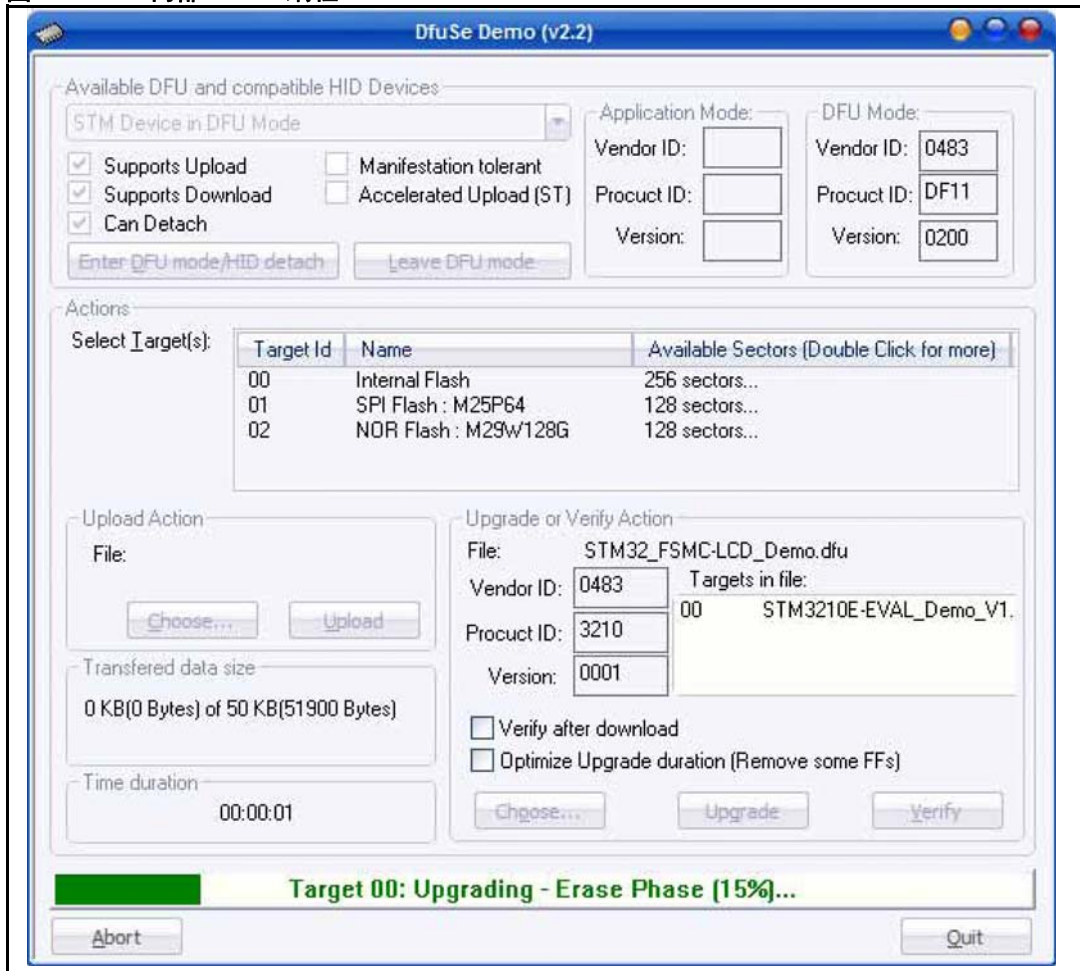
3. 然后单击 **Open** 按钮。当之前的窗口再出现时，单击 **Upgrade** 按钮，此时会出现对话框，如图 42 中所示。选择 **Yes** 继续。

图 42. 点击 Yes 继续



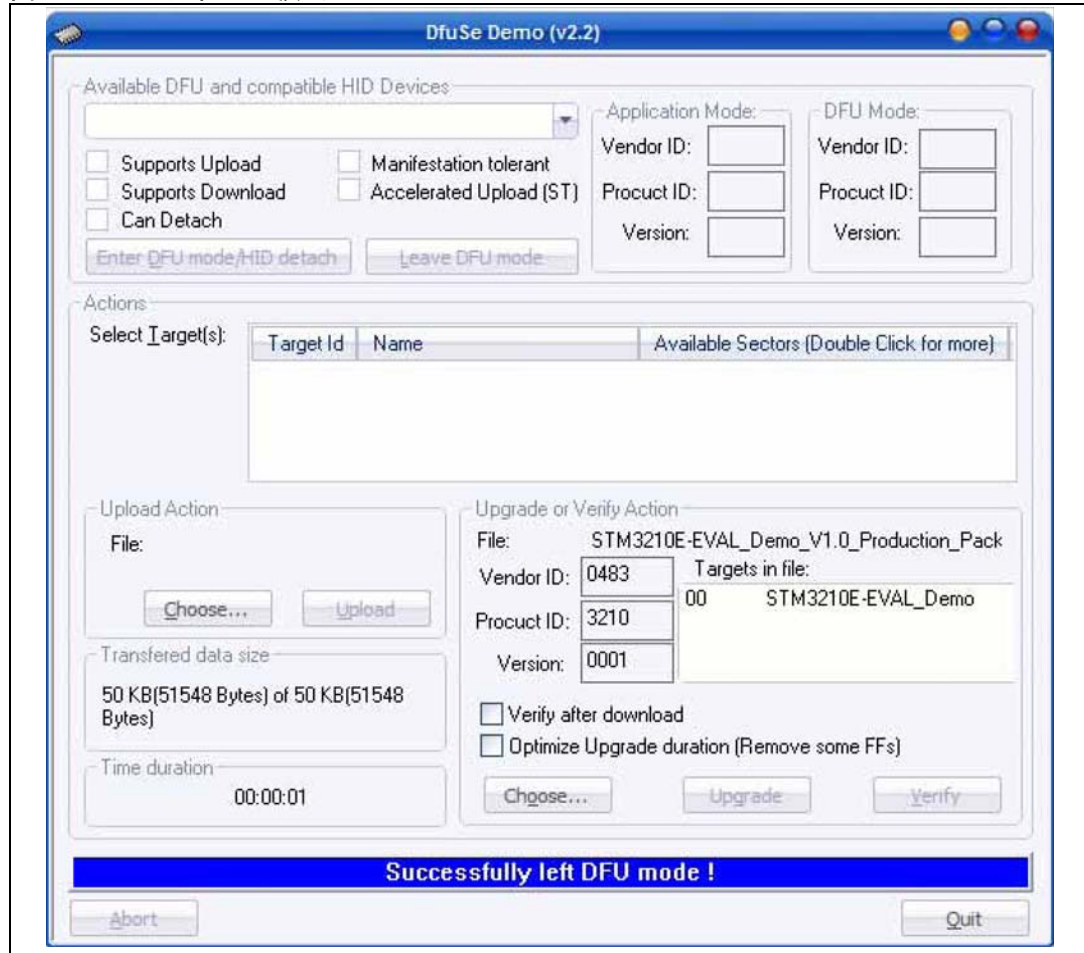
4. STM32F10xxx 的内部 Flash 开始编程，如 图 43 中所示。

图 43. 内部 Flash 编程



- 5. 当编程操作完成时，点击 **Leave DFU Mode**。这会令目标从 DFU 模式切换到应用模式，如 [图 44](#) 中所示。然后演示开始。

图 44. 退出 DFU 模式



7 结论

FSMC 因其在性能和灵活性上的优势，可方便地与彩色 LCD 接口，为构建多媒体应用服务。FSMC 适合低功耗、中分辨率 LCD 的显示器。大容量 STM32F10xxx FSMC 加上 MCU 性能、信号处理及控制功能等能力，可直接与 LCD 显示器接口，极大降低应用系统的成本和复杂性。

本应用笔记演示了怎样使用 STM32F10xxx FSMC 与 LCD 接口、使用少量 STM32F10xxx 处理能力创建显示应用。

8 修订历史

表 11. 文档修订历史

日期	修订	变更
2008 年 7 月 22 日	1	初始版本。
2008 年 9 月 19 日	2	删除了对 COSMORAM 的引用。 从图 1: 异步 NOR Flash 读访问时序和图 2: 异步 NOR Flash 写访问 时序中删除了 NWAIT 信号。

请仔细阅读：

中文翻译仅为方便阅读之目的。该翻译也许不是对本文档最新版本的翻译，如有任何不同，以最新版本的英文原版文档为准。

本文中信息的提供仅与 ST 产品有关。意法半导体公司及其子公司（“ST”）保留随时对本文档及本文所述产品与服务进行变更、更正、修改或改进的权利，恕不另行通知。

所有 ST 产品均根据 ST 的销售条款出售。

买方自行负责对本文所述 ST 产品和服务的选择和使用，ST 概不承担与选择或使用本文所述 ST 产品和服务相关的任何责任。

无论之前是否有任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为 ST 授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在 ST 的销售条款中另有说明，否则，ST 对 ST 产品的使用和 / 或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

意法半导体的产品不得应用于武器。此外，意法半导体产品也不是为下列用途而设计并不得应用于下列用途：（A）对安全性有特别要求的应用，例如，生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）汽车应用或汽车环境，且 / 或（D）航天应用或航天环境。如果意法半导体产品不是为前述应用设计的，而采购商擅自将其用于前述应用，即使采购商向意法半导体发出了书面通知，采购商仍将独自承担因此而导致的任何风险，意法半导体的产品规格明确指定的汽车、汽车安全或医疗工业领域专用产品除外。根据相关政府主管部门的规定，ESCC、QML 或 JAN 正式认证产品适用于航天应用。

经销的 ST 产品如有不同于本文档中提出的声明和 / 或技术特点的规定，将立即导致 ST 针对本文所述 ST 产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大 ST 的任何责任。

ST 和 ST 徽标是 ST 在各个国家或地区的商标或注册商标。

本文档中的信息取代之前提供的所有信息。

ST 徽标是意法半导体公司的注册商标。其他所有名称是其各自所有者的财产。

© 2015 STMicroelectronics 保留所有权利

意法半导体集团公司

澳大利亚 - 比利时 - 巴西 - 加拿大 - 中国 - 捷克共和国 - 芬兰 - 法国 - 德国 - 中国香港 - 印度 - 以色列 - 意大利 - 日本 - 马来西亚 - 马耳他 - 摩洛哥 - 菲律宾 - 新加坡 - 西班牙 - 瑞典 - 瑞士 - 英国 - 美国

www.st.com

