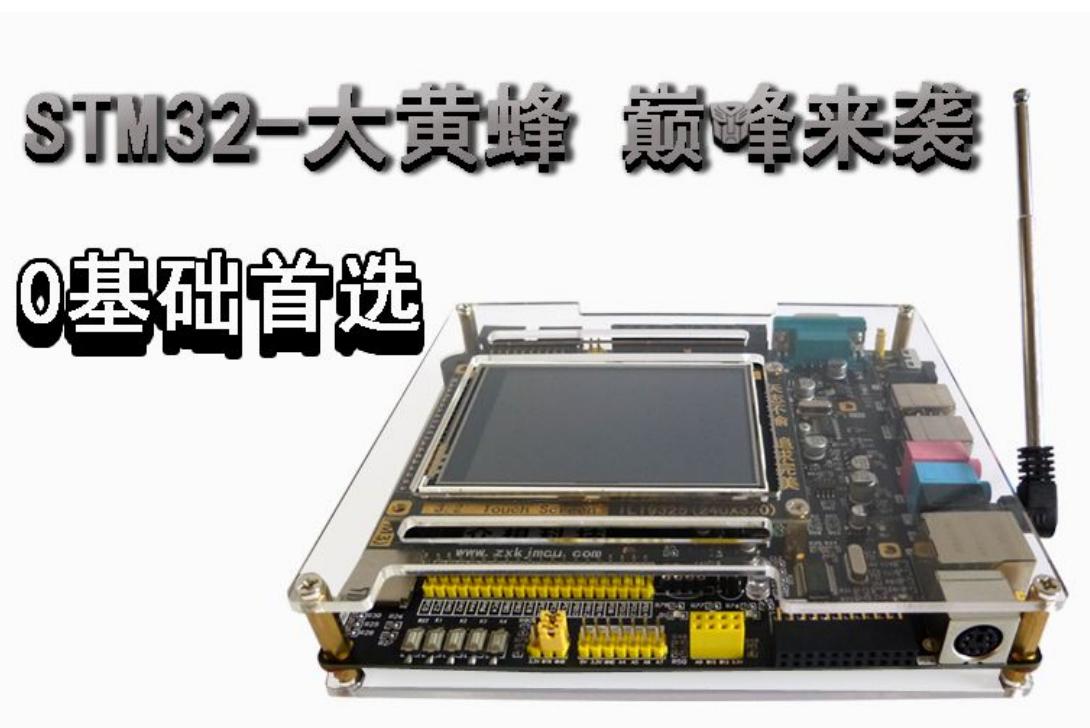


# 学 ARM 从 STM32 开始

STM32 开发板库函数教程—实战篇



官方网站: <http://www.zxkjmcu.com>

官方店铺: <http://zxkjmcu.taobao.com>

官方论坛: <http://bbs.zxkjmcu.com>

刘洋课堂: <http://school.zxkjmcu.com>

## 4. 30 灵活的静态存储控制器 FSMC 原理

### 4. 30. 1 FSMC 功能介绍

- 将 AHB 传输信号转换到适当的外部设备协议
- 满足访问外部设备的时序要求

所有的外部存储器共享控制器输出的地址、数据和控制信号，每个外部设备可以通过一个唯一的片选信号加以区分。FSMC 在任一时刻只访问一个外部设备。

- 具有静态存储器接口的器件包括：
  - 静态随机存储器 (SRAM)
  - 只读存储器 (ROM)
  - NOR 闪存
  - PSRAM (4 个存储器块)
- 两个 NAND 闪存块。
- 16 位的 PC 卡兼容设备
- 8 或 16 位数据总线
- 每一个存储器块都有独立的片选控制
- 每一个存储器块都可以独立配置
- 时序可编程以支持各种不同的器件：
  - 等待周期可编程 (多达 15 个周期)
  - 总线恢复周期可编程 (多达 15 个周期)

—输出使能和写使能延迟可编程(多达 15 周期)

—独立的读写时序和协议，可支持宽范围的存储器和时序

- PSRAM 和 SRAM 器件使用的写使能和字节选择输出
- 将 32 位的 AHB 访问请求，转换到连续的 16 位或 8 位，对外部 16 位或 8 位器件的访问

## 4.30.2 FSMC 功能框图

FSMC 包含以下四个模块：

- (1) AHB 接口（包含 FSMC 配置寄存器）
- (2) NOR 闪存和 PSRAM 控制器
- (3) NAND 闪存和 PC 卡控制器
- (4) 外部设备接口

要注意的是，FSMC 可以请求 AHB 进行数据宽度的操作。如果 AHB 操作的数据宽度大于外部设备（NOR 或 NAND 或 LCD）的宽度，此时 FSMC 将 AHB 操作分割成几个连续的较小的数据宽度，以适应外部设备的数据宽度。

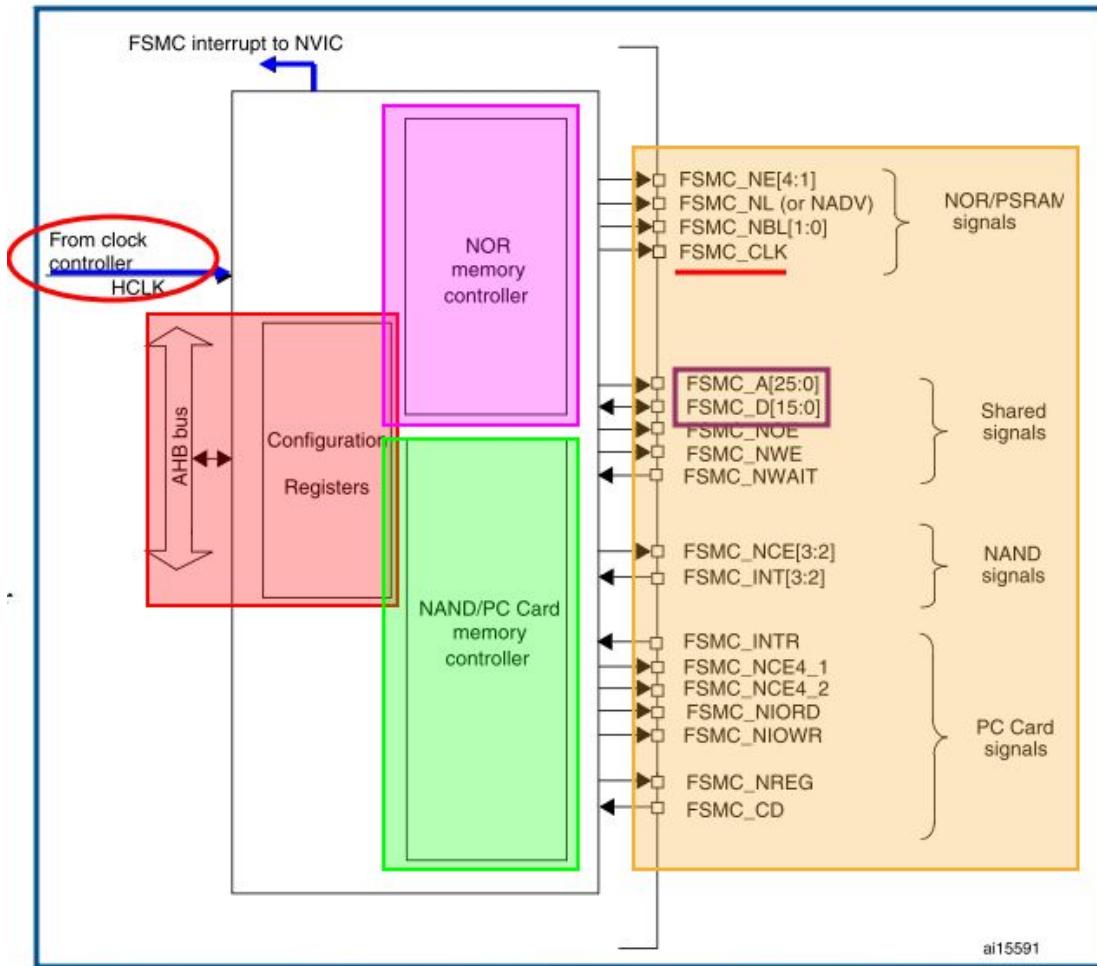


图 4.3.1 FSMC 功能框图

### 4.30.3 FSMC 对外挂设备的地址映像

从 FSMC 的角度看，可以把外部存储器划分为固定大小为 256M 字节的四个存储块，见“图 4.3.2 外挂设备的地址映射”。

- 存储块 1 用于访问最多 4 个 NOR 闪存或 PSRAM 存储设备。4 个 NOR/PSRAM 区有 4 个专用的片选。
- 存储块 2 和 3 用于访问 NAND 闪存设备，每个存储块连接一个 NAND 闪存。
- 存储块 4 用于访问 PC 卡设备每一个存储块上的存储器类型是由用户在配置寄存器中定义的。

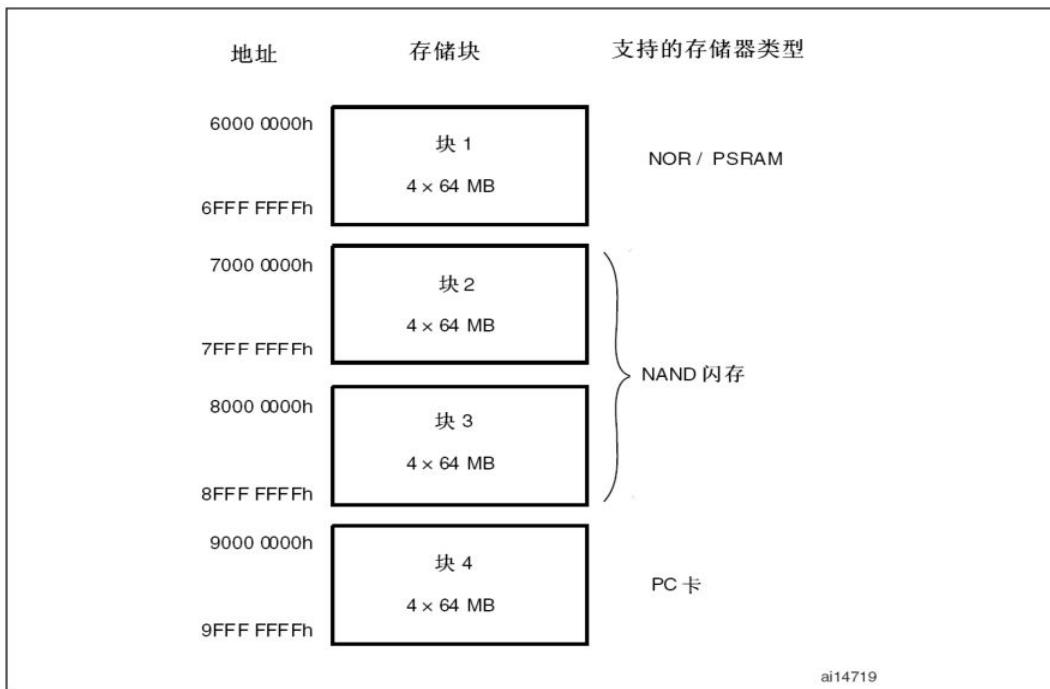


图 4.3.2 外挂设备的地址映射

#### 4.30.4 NOR 和 PSRAM 地址映像

从上图可以看出，FSMC 对外部设备的地址映像从 0x6000 0000 开始，到 0x9FFF FFFF 结束，共分 4 个地址块，每个地址块 256M 字节。可以看出，每个地址块又分为 4 个分地址块，大小 64M。对 NOR 的地址映像来说，我们可以通过选择 HADDR[27:26]来确定当前使用的是哪个 64M 的分地址块，而这四个分存储块的片选，则使用 NE[4:1]来选择。数据线/地址线/控制线是共享的。

NOR/PSRAM 是“static memory map”设备：256M 字节的空间需要 28 根地址线寻址。

- HADDR 是内部 AHB 地址总线
- HADDR[25:0]来自外部存储器地址 FSMC[25:0]
- HADDR[26:27]对 4 个 region 分别寻址

附表 1 外部存储器地址

数据宽度	连接存储器的地址线	最大访问存储器空间 (位)
8 位	HADDR[25:0] 与 FSMC_A[25:0] 对应连接	64M 字节*8=512M 位
16 位	HADDR[25:1] 与 FSMC_A[24:0] 对应连接, HADDR[0] 未连接	64M 字节/2*16=512M 位

无论外接 8 位/16 位宽度设备, FSMC\_A[0] 永远连接外部设备的地址 A[0]

- 外接 16 位宽度存储器: HADDR[25:1]      FSMC\_A[24:0]
- 外接 8 位宽度存储器: HADDR[25:0]      FSMC\_A[25:0]

附表 2: HADDR[27:26] 说明

选择的存储快	片选信号	地址范围	HADDR	
			[27:26]	[25:0]
存储器 1 NOR/PSRAM 1	FSMC_NE1	0x6000, 0000~63FF, FFFF	00	FSMC_A[25:0]
存储器 1 NOR/PSRAM 2	FSMC_NE2	0x6400, 0000~67FF, FFFF	01	FSMC_A[25:0]
存储器 1 NOR/PSRAM 3	FSMC_NE3	0x6800, 0000~6BFF, FFFF	10	FSMC_A[25:0]
存储器 1 NOR/PSRAM 4	FSMC_NE4	0x6C00, 0000~6FFF, FFFF	11	FSMC_A[25:0]

#### 4.30.5 100 引脚封装的 FSMC

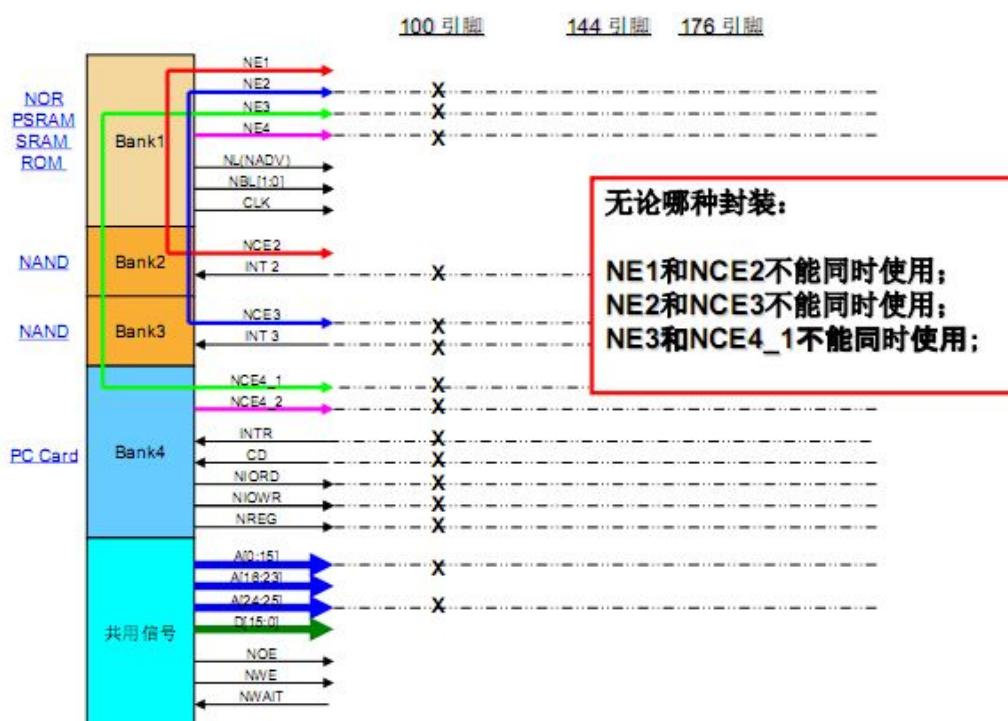


图 4.3.3 100 引脚封装的 FSMC (一)

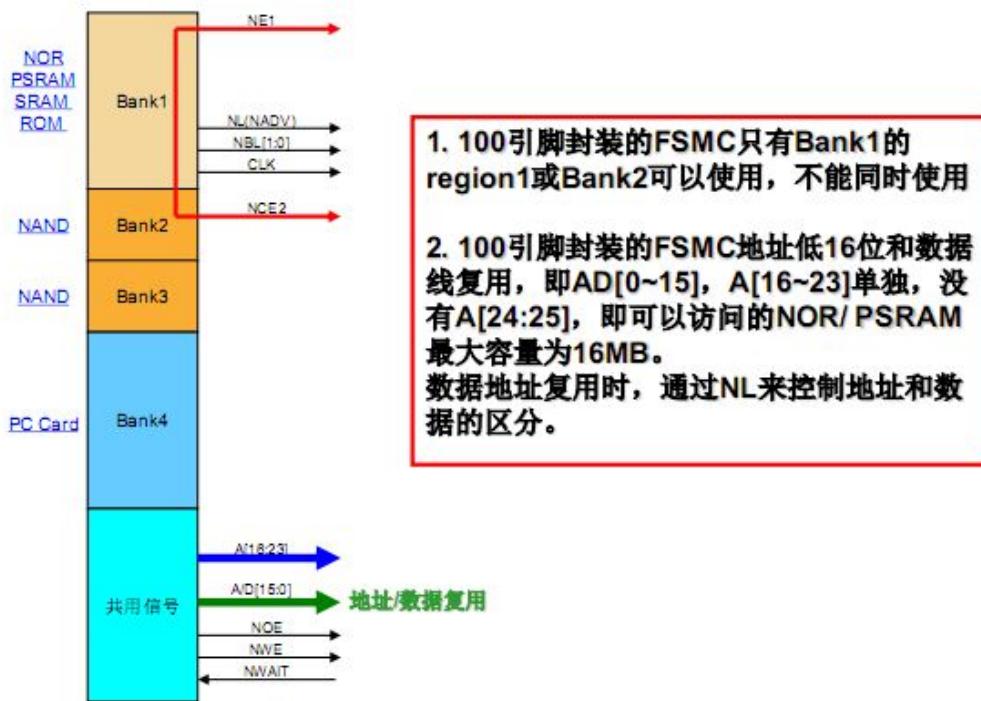


图 4.3.4 100 引脚封装的 FSMC (二)

#### 4.30.6 把 TFT-LCD 当成 SRAM 设备使用

这里我们介绍下为什么可以把 TFT-LCD 当成 SRAM 设备用：首先我们了解下外部 SRAM 的连接。外部 SRAM 的控制一般有：地址线(如 A0–A18)、数据线(如 D0–D15)、写信号(WE)、读信号(OE)、片选信号(CS)，如果 SRAM 支持字节控制，那么还有 UB/LB 信号。

TFT-LCD 信号包括：RS、D0–D15、WR、RD、CS、RST 和 BL 等，其中真正操作 LCD 的时候需要用到的就只有：RS、D0–D15、WR、RD 和 CS。其操作时序和 SRAM 的控制完全类似，唯一不同就是 TFT-LCD 有 RS 信号，但是没有地址信号。

TFT-LCD 通过 RS 信号来决定传送的数据是数据还是命令，本质上可以理解为一个地址信号，比如我们把 RS 接在 A0 上面，那么当 FSMC 控制器写

地址 0 的时候，会使得 A0 变为 0，对 TFT-LCD 来说，就是写命令。而 FSMC 写地址 1 的时候，A0 将会变为 1，对 TFT-LCD 来说，就是写数据了。这样，就把数据和命令分开了，他们其实就是对应 SRAM 操作的两个连续地址。当然 RS 也可以连接在其他地址线上，众想科技公司的 LY-STM32 大黄蜂开发板是把 RS 连接在 A16 上面的。