

ARM 嵌入式最小系统的软硬件架构的改进研究

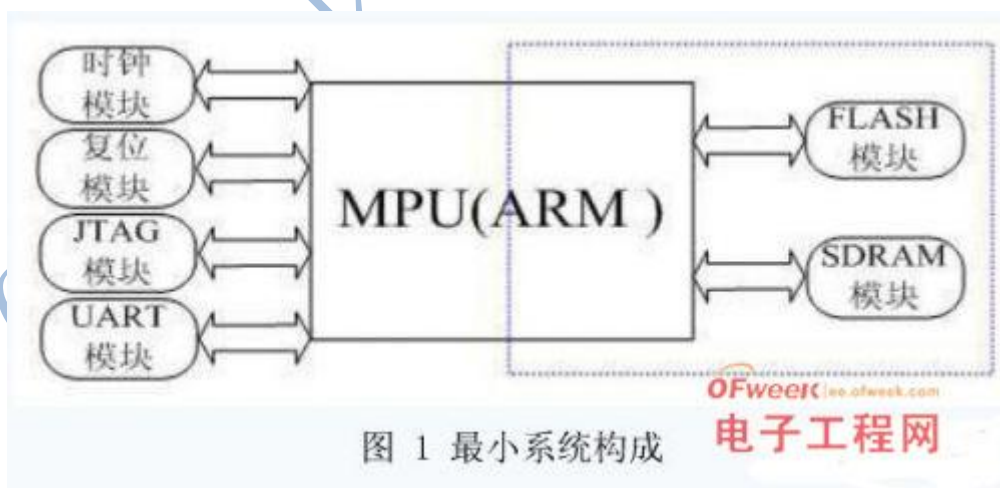
1. 引言

随着嵌入式相关技术的迅速发展，嵌入式系统的功能越来越强大，应用接口更加丰富，根据实际应用的需要设计出特定的嵌入式最小系统和应用系统，是嵌入式系统设计的关键。目前在嵌入式系统开发的过程中，开发者往往把大量精力投入到嵌入式微处理器 MPU(Micro Processing Unit)与众多外设的连接方式以及应用代码的开发之中，而忽视了对嵌入式系统最基本、最核心部分的研究。

当前在嵌入式领域中，ARM(Advanced RISC Machines)处理器被广泛应用于各种嵌入式设备中。由于 ARM 嵌入式体系结构类似并且具有通用的外围电路，同时 ARM 内核的嵌入式最小系统的设计原则及方法基本相同，这使得对嵌入式最小系统的研究在整个系统的开发中拥有着至关重要的意义。本文以基于 ARM 的嵌入式最小系统为平台，从硬件和软件两方面对嵌入式最小系统的架构进行了研究，硬件方面主要介绍了 ARM 处理器与典型外部存储器的接口技术，软件方面重点就嵌入式最小系统的启动架构做了详细分析。

2. 嵌入式最小系统

嵌入式最小系统即是在尽可能减少上层应用的情况下，能够使系统运行的最小化模块配置。对于一个典型的嵌入式最小系统，以 ARM 处理器为例，其构成模块及其各部分功能如图 1 所示，其中 ARM 微处理器、FLASH 和 SDRAM 模块是嵌入式最小系统的核心部分。



时钟模块——通常经 ARM 内部锁相环进行相应的倍频，以提供系统各模块运行所需的时钟频率输入

Flash 存储模块——存放启动代码、操作系统和用户应用程序代码

SDRAM 模块——为系统运行提供动态存储空间，是系统代码运行的主要区域

JTAG 模块——实现对程序代码的下载和调试

UART 模块——实现对调试信息的终端显示

复位模块——实现对系统的复位

3. 外存储器接口技术

ARM 处理器与外部存储器 (Flash 和 SDRAM) 的接口技术是嵌入式最小系统硬件设计的关键。根据需求选择合理的接口方式，可以有效的提升嵌入式系统的整体性能。

3.1 常用外存储器简介

(1) Nor Flash 与 Nand Flash

Nor Flash 也称为线性 Flash，可靠性高、随机读取速度快，具有芯片内执行 (XIP, eXecute In Place) 的特点，这样应用程序可以直接在 Flash 闪存内运行，不必再把代码读到系统 RAM 中。常用于擦除和编程操作较少而直接执行代码的场合。

Nand Flash 则是高数据存储密度的理想解决方案，一般用于数据存储和文件存储，以块为单位进行擦除，具有擦除速度快的优点。

(2) 同步动态存储器 SDRAM

SDRAM (Synchronous Dynamic Random Access Memory) 是在早期 DRAM 的基础上改进而来的，它是同步内存，并在接口信号中引入了 CLK 信号，所有数据、地址和控制信号都是和 CLK 上升沿对齐的。此外 SDRAM 还在内部引入了一个命令控制器，处理器访问 SDRAM 都是通过向命令控制器发送命令来实现的。

3.2 ARM 处理器与 Flash 接口技术

3.2.1 ARM 处理器与 Nor Flash 接口技术

Nor Flash 带有 SRAM 接口，有足够的地址引脚，可以很容易的对存储器内部的存储单元进行直接寻址。在实际的系统中，可以根据需求选择 ARM 处理器与 Nor Flash 的连接方式。图 2 给出了嵌入式最小系统在包含两块 Nor Flash 的情况下，ARM 处理器与 Nor Flash 两种不同的连接方式。

1) 双 Flash 独立片选

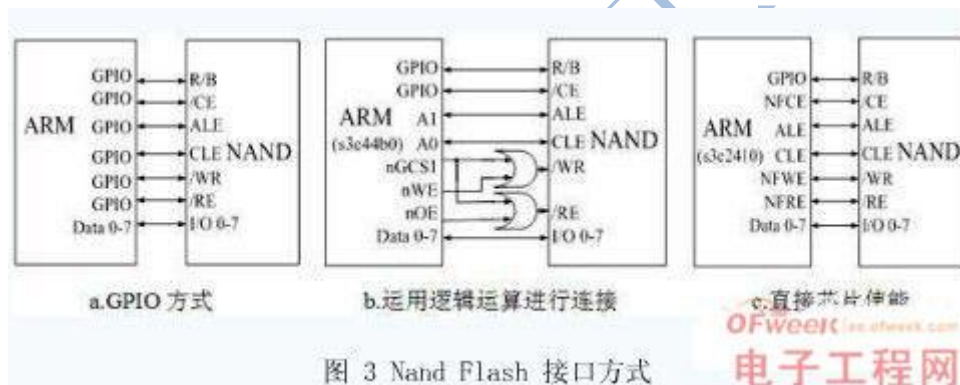
该方式是把两个 Nor Flash 芯片各自作为一个独立的单元进行处理。根据不同的应用需要，可以在一块 Flash 中存放启动代码，而在另一块 Flash 中建立文件系统，存放应用代码。该方式操作方便，易于管理。

(2) 双 Flash 统一片选

该方式是把两个 Nor Flash 芯片合为一个单元进行处理，ARM 处理器将它们作为一个并行的处理单元来访问，本例是将两个 8bit 的 Nor Flash 芯片 SST39VF1601 用作一个 16bit 单元来进行处理。对于 N(N>2) 块 Flash 的连接方式可以此作为参考。

3. 2. 2 ARM 处理器与 Nand Flash 接口技术

Nand Flash 接口信号比较少，地址，数据和命令总线复用。Nand Flash 的接口本质上是一个 I/O 接口，系统对 Nand Flash 进行数据访问的时候，需要先向 Nand Flash 发出相关命令和参数，然后再进行相应的数据操作。ARM 处理器与 Nand Flash 的连接主要有三种方式，如图 3 所示：



运用 GPIO 管脚方式去控制 Nand Flash 的各个信号，在速度要求相对较低的时候，能够较充分的发挥 NAND 设备的性能。它在满足 NAND 设备时域需求方面将会有很大的便利，使得 ARM 处理器可以很容易的去控制 NAND 设备。该方式需要处理器提供充足的 GPIO。

(2) 运用逻辑运算方式进行连接

在该方式下，处理器的读和写使能信号通过与片选信号 CS 进行逻辑运算后去驱动 NAND 设备对应的读和写信号。图 3 中 b 例为 SamSung 公司 ARM7TDMI 系列处理器 S3C44B0 与 Nand Flash K9F2808U0C 的连接方式。

(3) 直接芯片使能

有些 ARM 处理器如 S3C2410 内部提供对 NAND 设备的相应控制寄存器，通过控制寄存器可以实现 ARM 处理器对 NAND 设备相应信号的驱动。该方式使得 ARM

处理器与 NAND 设备的连接变得简单规范,图 3 中 c 例给出了 ARM 处理器 S3C2410 与 Nand Flash K9F2808U0C 的连接方式。

3.3 ARM 处理器与 SDRAM 接口技术

嵌入式最小系统的外部动态存储器模块一般采用 SDRAM。现在的大多数 ARM 处理器内部都集成有 SDRAM 控制器,通过它可以很容易的访问 SDRAM 内部的每一个字节。在实际开发中可以根据需要选用一片或多片 SDRAM。图 4 中给出了两种常用的接口方式。

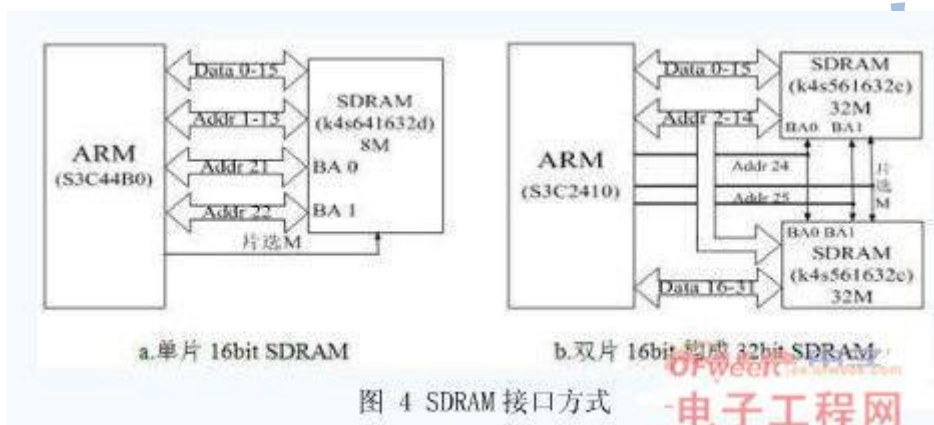


图 4 SDRAM 接口方式

(1) 单片 SDRAM

图 4 中 a 例为 ARM 处理器 S3C44B0 与一个 16bit 的 SDRAM K4S641632d 的连接方式。在对尺寸有严格限制且对动态存储器容量要求不高的嵌入式系统中常采用此种连接方式。

(2) 双片 16bit SDRAM 结合使用

在双片 16bit SDRAM 合成一个 32bit SDRAM 使用时, ARM 处理器的地址线 A2 接 SDRAM 的地址线 A0, 其余地址依次递增, 这是因为在 SDRAM 中字节是存储容量的惟一单位, 而此时 SDRAM 为 32bit 位宽。

SDRAM 的 BA 地址线是其内部 Bank 的地址线, 代表了 SDRAM 内存的最高位。在图 4 中 b 例的 SDRAM 总大小是 64M(64M=), 需要 A25-A0 引脚来寻址, 所以 BA1-BA0 连接到了 A25-A24 引脚上。还需注意的是 SDRAM 内存行地址和列地址是复用的, 所以地址线的数目一般少于 26 条, 具体连接需要参考 ARM 处理器和 SDRAM 的相关手册。

值得注意的是, 有些 ARM 处理器如 SamSung 公司的 ARM940T 系列处理器 S3C2510, 其地址总线与其它标准 ARM 处理器不太一样, 它的地址线分为内部地址线和外部地址线两种类型。根据所连接的存储器数据总线宽度, 内部地址线进行相应的移位, 对应到外部地址线 A23-A0, 从而对外提供固定的地址线 A23-A0。其实质与典型的 ARM 处理器是一致的。

4. 嵌入式系统启动架构

启动架构是嵌入式系统的关键技术。掌握启动架构对于了解嵌入式系统的运行原理有着重要的意义。嵌入式系统在启动时，引导代码、操作系统的运行和应用程序的加载主要有两种架构，一种是直接从 Nor Flash 启动的架构，另一种是直接从 Nand Flash 启动的架构。

需要注意的是，在嵌入式系统启动引导的过程中会有多种情况出现，如 vxWorks 的启动代码 BootRom 就有压缩和非压缩，驻留和非驻留方式之分，而操作系统本身也多以压缩映像方式存储，所以启动代码在执行和加载过程中需要根据不同的情况，作出相应的处理。

4.1 从 Nor Flash 启动

Nor Flash 具有芯片内执行 (XIP, eXecute In Place) 的特点，在嵌入式系统中常做为存放启动代码的首选。从 Nor Flash 启动的架构又可细分为只使用 Nor Flash 的启动架构和 Nor Flash 与 Nand Flash 配合使用的启动架构。图 5 给出了这两种启动架构的原理框图。

4.1.1 单独使用 Nor Flash

在该架构中，引导代码、操作系统和应用代码共存于同一块 Nor Flash 中。系统上电后，引导代码首先在 Nor Flash 中执行，然后把操作系统和应用代码加载到速度更高的 SDRAM 中运行。另一种可行的架构是，在 Nor Flash 中执行引导代码和操作系统，而只将应用代码加载到 SDRAM 中执行。

该架构充分利用了 Nor Flash 芯片内执行的特点，可有效提升系统性能。不足在于随着操作系统和应用代码容量的增加，需要更大容量昂贵的 Nor Flash 来支撑。

4.1.2 Nor Flash 和 Nand Flash 配合使用

Nor Flash 的单独使用对于代码量较大的应用程序会增加产品的成本投入，一种的改进的方式是采用 Nor Flash 和 Nand Flash 配合使用的架构。在该架构中附加了一块 Nand Flash，Nor Flash (2M 或 4M) 中存放启动代码和操作系统 (操作系统可以根据代码量的大小选择存放于 Nor Flash 或者 Nand Flash)，而 Nand Flash 中存放应用代码，根据存放的应用代码量的大小可以对 Nand Flash 容量做出相应的改变。

系统上电后，引导代码直接在 Nor Flash 中执行，把 Nand Flash 中的操作系统和应用代码加载到速度更高的 SDRAM 中执行。也可以在 Nor Flash 中执行引导代码和操作系统，而只将 Nand Flash 中的应用代码加载到 SDRAM 中执行。该架构是当前嵌入式系统中运用最广泛的启动架构之一。

4.1.2 从 Nand Flash 启动

有些处理器如 Samsung 公司的 ARM920T 系列处理器 S3C2410 支持从 Nand Flash 启动的模式，它的工作原理是将 NandFlash 中存储的前 4KB 代码装入一个称为 Steppingstone (BootSRAM) 的地址中，然后开始执行该段引导代码，从而完成对操作系统和应用程序的加载。该方式需要处理器内部有 NAND 控制器，同时还要提供一定大小额外的 SRAM 空间，有一定的使用局限性，在实际开发中较少使用。

5. 结束语

本文作者创新点：根据 ARM 体系结构的相通性，以基于 ARM 的嵌入式最小系统为平台，从硬件和软件两方面对其架构进行了研究。硬件方面就 ARM 处理器与常用外部存储器的接口方式进行了阐述，软件方面给出了嵌入式最小系统典型的启动架构及其选择标准。

OFweek 电子工程网