

单片机软硬件联合仿真解决方案

本文介绍一种嵌入式系统仿真方法，通过一种特殊设计的指令集仿真器 ISS 将软件调试器软件 Keil uVision2 和硬件语言仿真器软件 Modelsim 连接起来，实现了软件和硬件的同步仿真。

缩略词解释：

BFM: 总线功能模块。在 HDL 硬件语言仿真中，BFM 完成抽象描述数据和具体的时序信号之间的转换。

PLI: Verilog 编程语言接口，是 C 语言模块和 Verilog 语言模块之间交换数据的接口定义。

TCL: 字面意思是工具命令语言，是一种解释执行语言，流行 EDA 软件一般都集成有 TCL。使用 TCL 用户可以编写控制 EDA 工具的脚本程序，实现工具操作自动化。

ISS: CPU 指令集仿真器，可以执行 CPU 的机器码。

TFTP: 简单文件传输协议，Windows 的 tftp.exe 既是该协议的客户端实现。

SMART MEDIA: 一种存储卡，常用于数码相机、MP3。

DMA: 直接内存访问。用于外部设备之间高速数据转移。

MAC: 媒体接入控制器。本文中是指网卡芯片。

前言

传统的嵌入式系统中，设计周期、硬件和软件的开发是分开进行的，并在硬件完成后才将系统集成在一起，很多情况下，硬件完成后才开始进行实时软件和整体调试。软硬件联合仿真是一种在物理原型可用前，能尽早开始调试程序的技术。

软硬件联合仿真有可能使软件设计工程师在设计早期着手调试，而采用传统的方法，设计工程师直到硬件设计完成才能进行除错处理。有些软件可在没有硬件支持的情况下完成任务的编码，如不涉及到硬件的算法。与硬件相互作用的编码在获得硬件之前编写，但只有在硬件上运行后，才能真正对编码进行调试。通过采用软硬件联合仿真技术，可在设计早期开始这一设计调试过程。由于软件的开发通常在系统开发的后段完成，在设计周期中较早的开始调试有可能将使这一项目提早完成，该技术会降低首次将硬件和软件连接在一起时出现意外而致使项目延期完成所造成的风险。

在取得物理原型前,采用软硬件联合仿真技术对硬件和软件之间的接口进行验证,将使你不会花太多的时间在后期系统调试上。当你确实拿到物理原型开始在上面跑软件的时候,你会发现经过测试的软件部分将会正常工作,这会节省项目后期的大量时间及努力。

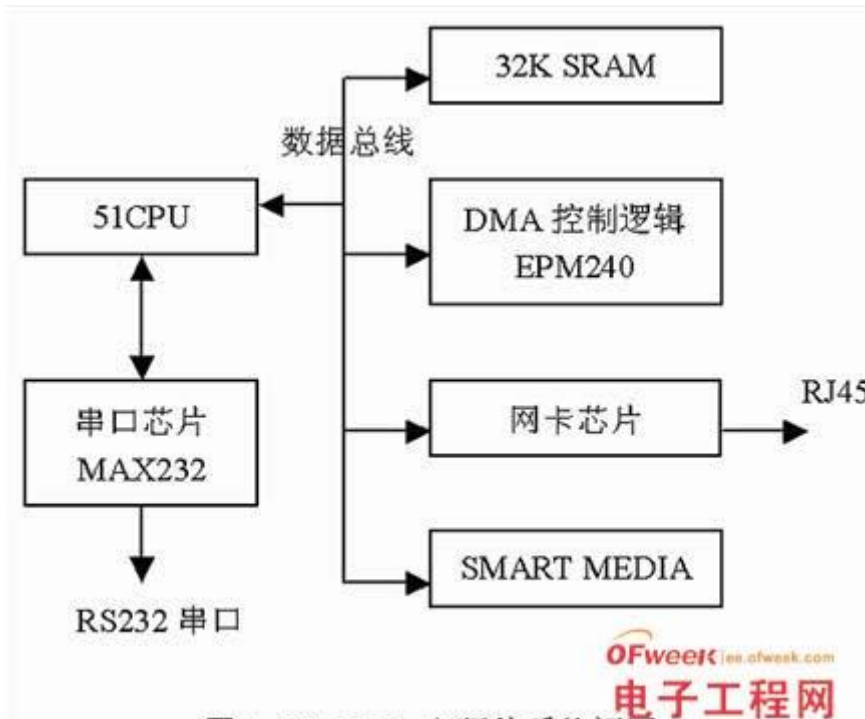


图 1. Mini Web 卡硬件系统框图

软硬件联合仿真系统由一个硬件执行环境和一个软件执行环境组成,通常软件环境和硬件环境都有自己的除错和控制界面,软件通过一系列由处理器启动的总线周期与硬件的交互作用。本文以一个 Mini Web 卡的开发介绍一种软硬件联合仿真系统。

该方案的核心是采用一个 51 单片机仿真引擎 GoldBull ISS51 (以下简称 ISS51), ISS51 是 51 单片机开发环境 Keil uVision2 的一个插件, ISS51 具有连接 Keil 和硬件仿真环境 Modelsim 的接口,可以实现软硬件同步仿真。在该系统中, Keil 作为软件调试界面, Modelsim 作为硬件仿真和调试界面, ISS51 负责软件执行、监控软件断点、单步执行、内存和寄存器数据返回给 Keil、CPU 总线时序产生和捕获、内部功能模块(如定时器,串口)的运行等功能。

Mini Web 卡介绍

Mini Web 卡是一个运行在单片机上的 Web 服务器,提供网口连接,有大量文件系统,提供 TFTP 和 HTTP 服务。尽管软件系统比较复杂,但优化编译后,执行代码还不足 25K,为后续升级留下了足够空间。

51CPU 采用 SST89 系列,这种 CPU 具有 ISP 功能,可以通过 RS232 串口,直接将目标码下载到 CPU。

DMA 控制逻辑是一个可编程逻辑器件，采用的是 ALTERA 的 CPLD EPM240，主要功能是实现外围器件之间的 DMA 传递。因为 51CPU 进行 IO 访问是很低效的，需要 24 个时钟周期才能进行一次 IO 访问，在外围设备之间转移数据则需要更多的时钟周期，使用 DMA 控制逻辑可以达到 3 个时钟周期就能转移一个字节。本系统中处理多种网络协议，需要大量报文收发和文件系统访问，采用 DMA 可以极大地提高 51 单片机的数据处理速度。DMA 通道主要有 MAC 芯片与 RAM 之间的数据块转移，SMART MEDIA 和 RAM 之间的数据块转移。

网卡芯片采用的是 AX88796，主要的优点是可以和 51CPU 方便地接口；支持 100M 以太网，速度高；有较大的接收报文缓存，能够平滑网络流量，减少因 51CPU 处理速度慢导致的报文丢弃和重发。

SMART MEDIA 是一个移动存储卡，主要用于存储文件，Mini Web 卡支持 8M 到 256M 的 SMD 卡。

文件系统是 Mini Web 卡的新开发模块，文件系统的测试主要通过 TFTP 来进行，为此 Mini Web 卡上的 TFTP 服务程序进行了特殊设计，支持格式化 SMART MEDIA，获取剩余空间，获取文件名列表，上传、下载和删除文件。

软硬件联合仿真的必要性：

Mini Web 卡软件模块多，软件开发风险较大。软件对硬件的依赖较强，FLASH 存储器的访问驱动、网卡驱动、DMA 驱动，需要软硬件协同调试。

文件系统的开发，在仿真环境下更容易和快捷。比如在仿真结束时，可以将 SMART MEDIA 仿真模型中的数据倒换到磁盘文件中，在仿真开始时，将磁盘文件中的数据加载到 SMART MEDIA 仿真模型中，在定位文件系统的问题时，这一个功能很有用。

采用软硬件联合仿真，便于系统前期设计。51 单片机的外部 RAM 访问效率较低，内存拷贝、外部器件之间的数据块转移很浪费时间。将大量数据的拷贝操作或数据块校验、比较操作在 CPLD 内实现，可以大大改进 51 单片机处理数据的能力。通过软硬件联合仿真，可以评估 CPLD 处理数据对性能的改进。

Mini Web 卡软硬件联合仿真系统：

软硬件联合仿真主要解决的问题是系统功能设计与验证，它不解决电源、滤波电容、总线电平兼容问题。

做系统仿真，首先要对硬件系统建模。我们关注的是系统设计的正确性和可执行性。

系统中的串口只是用来支持 ISP 下载软件，软件部分没有对串口做任何操作，所以系统仿真可以不必考虑。

网卡芯片 AX88796，厂商没有提供仿真模型。它与 CPU 的接口符合 ISA 接口标准，软件对 AX88796 的操作是根据 NE2000 标准网卡芯片设计的，由此我们建立了一个网卡芯片的仿真模型。我们设计了一个 MAC BFM 来仿真网卡芯片的 ISA 接口，NE2000 定义的寄存器在 C 模型中实现，MAC BFM 与 NE2000 寄存器 C 模型通过 PLI 接*换数据。

SRAM 仿真模型是很容易获取的，很多器件生产商都提供 Verilog 仿真模型，但器件生产商提供的 Verilog 仿真模型都包含复杂的延时控制代码，这会影响仿真速度。根据经验，我们可以确保 SRAM 在单板设计中被正确应用，不会产生时序问题，所以我们可以采用一个简化的 SRAM 仿真模型，这是我们自己设计的，有效代码只有十几行。

51CPU BFM 负责单片机管脚时序的产生和捕获。51CPU BFM 是与 ISS51 紧密捆绑的，由 ISS51 安装程序提供。

SMART MEDIA 是三星公司提供的仿真模型，我们使用的也是三星公司的同类型存储卡。该模型可以用于验证软件操作 SMART MEDIA 的正确性和 DMA Controller 的接口时序。

DMA Controller 是 Mini Web 卡硬件开发的一部分，将逻辑设计代码应用于仿真，既能检测逻辑设计的正确性，又能使整个仿真系统得以正常运转。

将上述硬件模型连接起来，产生下图所示硬件系统模型图(点击查看大图)：

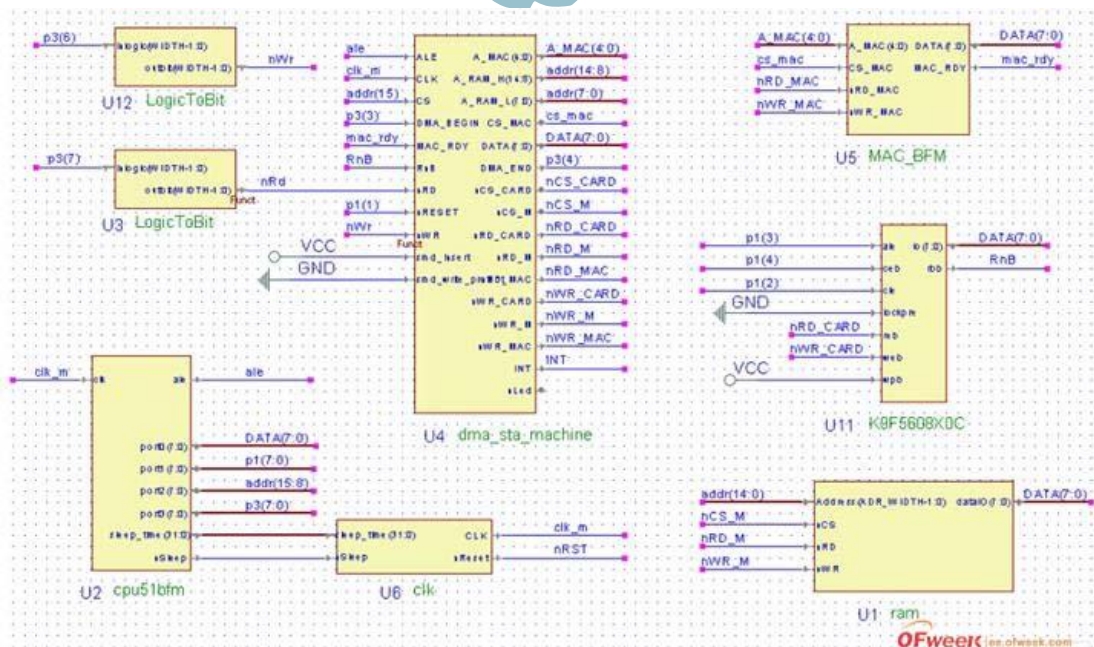


图 2. Mini Web 卡硬件模块电路图

图 2 中 U11 为 SMART MEDIA 仿真模型，U4 为 DMA Controller 模型。

虚拟网卡

做系统仿真，必须输入来自真实世界的激励，并将仿真系统的输出传递到真实世界。即便是不能连接到真实世界，也应该提供模拟真实世界的输入，并对仿真系统的输出进行检测和分析。

对于 Mini Web 卡来说，它和真实环境是通过网口连接的。使用虚拟网卡技术，能够将图 3 中的 MAC C Model 与虚拟网卡进行通讯。

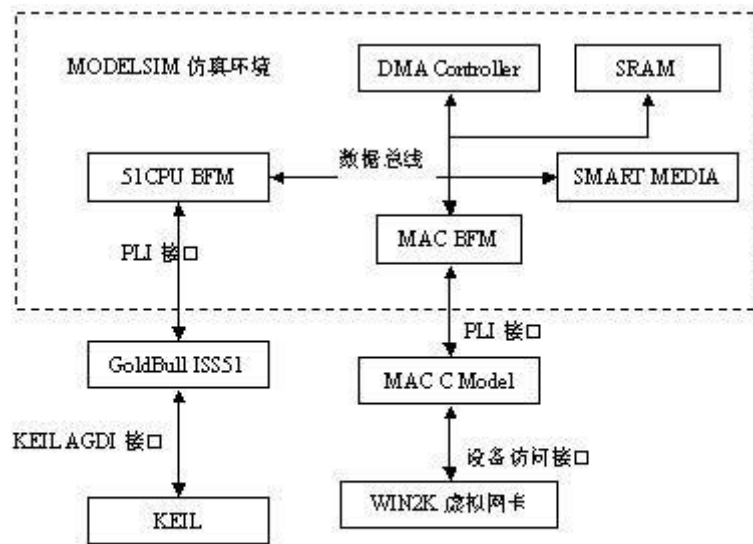


图 3. Mini Web 卡软硬件联合仿真系统框图

对于运行在 Windows 系统上的应用程序来说，它并不知道网卡是虚拟的还是真实的，应用程序通过虚拟网卡收发数据，事实上是与仿真系统在进行网络通信。

这样就可以使用 TFTP 向 Mini Web 卡仿真系统传递网页文件，使用 IE 浏览 Mini Web 卡仿真系统中的网页，Mini Web 卡的所有功能都能够被检验。

使用网络嗅探器 Sniffer 可以监控虚拟网卡的报文流，方便协议调试。

仿真加速技术

软硬件联合仿真，影响仿真速度的瓶颈在 HDL 代码部分的仿真。如果不设法提高 HDL 代码部分的仿真速度，软件调试就非常低效。

提高硬件仿真速度的方法之一是软件硬件仿真采用事件同步，只在 CPU 访问 I/O 时保持软件和硬件是同步的。

仿真加速方法之二是硬件仿真系统时钟休眠。对于 Mini Web 卡来说，只有 DMA Controller 是受时钟控制的，软件没有操作 DMA Controller 的期间，DMA Controller 的运作是毫无意义的，所以可以在非 DMA 操作期间，对时钟进行休眠；ISS51 在每次 I/O 访问时，给出与上次 I/O 访问的时间差，这个时间差经过处理可以作为时钟休眠的时间段。如果 ISS51 连续进行 I/O 访问，就不会产生时钟

休眠了。DMA Controller 工作于查询方式，可以采用时钟休眠技术，而不会导致仿真与真实结果的不一致。

方法之三是，缩短 SMART MEDIA 仿真模型中的一些长延时的时间参数。因为在等待 SMART MEDIA 进入就绪状态时，CPU 必须连续查询 IO，影响仿真速度。我们主要用于软件功能验证，这种修改也是可以接受的。

方法之四，在软件设计上，谨慎使用外部中断，因为一旦中断启动，ISS51 需要在每个机器周期查询是否有中断信号，导致软件仿真和硬件仿真在每个指令上都进行同步，影响仿真速度。如果一定要使用外部中断，建议用 C 模型代替 Verilog 模型，这样可不影响仿真速度；或者由用户根据外部模块产生外部中断的时机，使用 ISS51 的控制命令，在恰当时刻使能 ISS51 的中断模块。

在一个普通 PC（CPU 为 AMD 速龙 1000，SDRM512M 133），运行 Mini Web 卡仿真系统，使用 PING 命令测试 Mini Web 卡仿真系统的响应速度：

```
Reply from 10.10.112.76: bytes=32 time=64ms TTL=128
```

使用 IE 打开 Mini Web 卡仿真系统中的网页文件，感觉和拨号上网的速度差不太多。创建多个 TFTP 连接，同时向仿真系统传递或下载网页文件，同时使用 IE 进行网页浏览，都无响应中断现象出现。

总结

使用软硬件联合仿真，Mini Web 卡不需要硬件就能进行全部功能的仿真，增强了系统设计成功的信心。软硬件联合仿真方便系统设计调整，可以在设计前期*估性能，方便软件和硬件的 debug，是一个值得推广的技术。