

1 初探 OMAP3530

我在大二的上学期接触了 ARM 这类处理器，在学习其汇编指令集之后，发现她的汇编语言编程思路和 C 语言的特别接近，后来才知道，这款处理器是为 C 语言而设计的，于是，便深深的爱上了 ARM 这个体系结构。在之后的不断学习中，了解到 ARM 比较新的 CORTEX-A8 体系结构以及基于这个体系结构的处理器 OMAP3530，特别想买一块板子学习。但是作为穷学生，总共 2000 多元的花销对我来说，可谓是巨资。存了好久的钱，终于在去年（2010 年）国庆节的时候如愿以偿。下面这张图片就是我的学习平台。

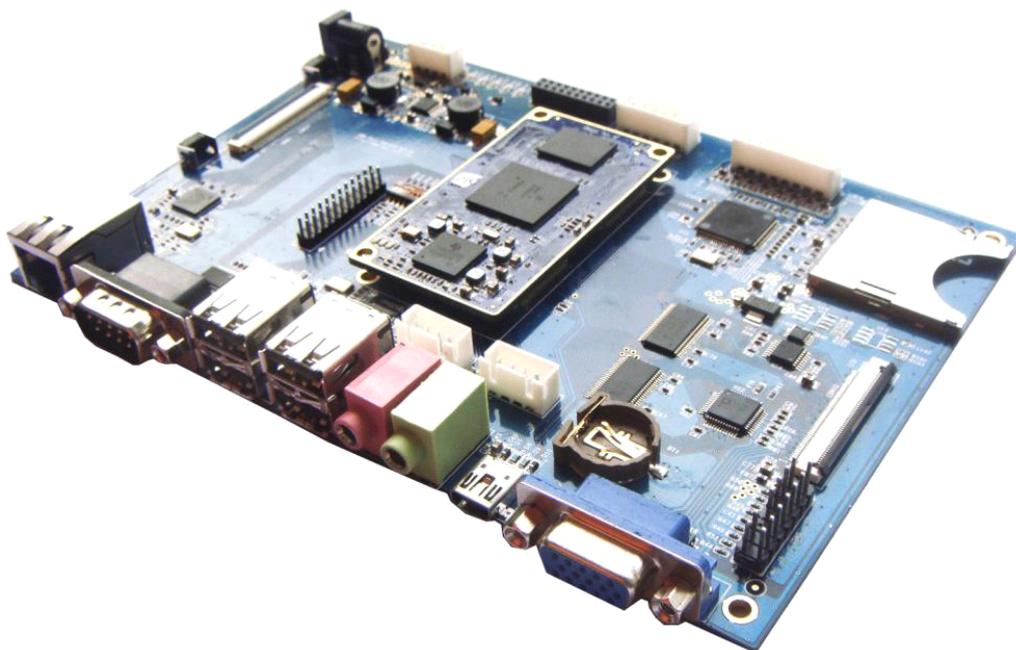


图 1 SBC3530

1.1 OMAP3530 与 TPS659xx 简介

OMAP3530 是 TI 公司推出的一款多核处理器，采用 720 MHz ARM® Cortex™-A8 内核与 520 MHz TMS320C64x+™ DSP。这个芯片使用的是 1.8V 的电平，因此在和许多外设通信时需要电平转换。在看这个芯片的数据手册时，还会发现：哎呀，这么牛的芯片怎么会没有 AD、DA 这些东西？原来 TI 把这些东西放在了他的电源管理芯片上，一般情况下，OMAP3530 会与电源管理芯片 TPS65930（也可以由 TPS65950 代替）一起工作，它们之间通过 I2C 总线进行通信。TPS65930 是一个“贤内助”，不但提供系统所需要的时钟、各种电源，还提供了 A/D、D/A、键盘扫描、PWM、少量的 GPIO 等功能。值得说明的是：一、TPS65930/50 虽然提供了 USB 接口使用的 5V 电源，但是最大值才有 100mA，因此通常不使用这个 LD0；二、SD 卡接口 1 和 2 的 C/D 引脚是由 TPS65930/50 管理的，OMAP3530 是支持 SD 卡启动的，在 SD 卡启动过程中，会不会通过 TPS659xx 来检查卡片是否插入呢？如果是这样，那使用小型的 microSD 卡槽（只有 8 个引脚）又该如何处理？目前这两个问题还不清楚。

1.2 OMAP3530 的数据手册如何查找（以下资料来至网络，部分链接可能无效，我只把我觉得常用的标识出来）

1.2.1 TI 官方资源

TI 中文主页：

<http://focus.ti.com.cn/cn/tihome/docs/homepage.tsp>

或者

<http://www.ti.com>

从“处理器”链接可以链接到每个芯片的资料主页，比如：

<http://focus.ti.com.cn/cn/docs/prod/folders/print/omap3530.html>

或者直接访问

www.ti.com/omap3530

可以下载数据手册，应用文档，用户手册，开发工具等等。

1.2.2 开发环境和软件开发包下载

(1) 建议大家注册一个 myti，方便下载一些软件开发工具和资料，并可以设定邮箱提醒

<http://my.ti.com>

(2) CCS 等升级

<https://www-a.ti.com/downloads/s...ServiceReleases.htm>

(3) 软件开发包 BIOS、CODEC ENG, FC, DSPLINK, NDK, xDais, Linux BSP 等

<https://www-a.ti.com/downloads/s...tcontent/index.html>

(4) 媒体算法和 wince 的 bsp，现在从 TI 的网站上可以免费下载：

<http://www.ti.com/requestfreesoftware>

(5) 版本更新:

<http://www.ti.com/myregisteredsoftware>

1.2.3 TI 相关开源资源:

(1) 很有用的 wiki 网页, 下面三个域名是一样的:

<http://wiki.davincidsp.com>

<http://wiki.omap.com>

<http://tiexpressdsp.com>

(2) XDC/RTSC 是完全开源的项目: <http://wiki.eclipse.org/DSDP/RTSC>

(3) TI Linux OMAP community, 可以下载内核等代码, 不需要注册

<http://linux.omap.com>

(4) TI E2E 社区, 现在人气可能还不够旺, 但是有问必答

<https://community.ti.com/forums/32.aspx>

1.2.4 OMAP 和 Davinci 第三方相关资源:

(1) OMAP 开源共享

<http://www.6lic.com/code/forumdisplay.php?fid=291&page=1>

(2) OMAP3 所用的 code sourcery 开发工具链, 免费下载

http://www.codesourcery.com/gnu_toolchains/arm/download.html

(3) OMAP 正式发售 EVM 资料, 由于是 mistral 做的板子, 需要从 mistral 下载相关资料

<http://downloads.mistralsolution.com/omap3/Documentation/>

(4) LogicPD ZOOM SDK

http://www.logicpd.com/products/omap3_development_kit

(5) Spectrum Digital - 很多 DSP 的开发板和仿真器可以从这儿下载资料和升级, 包括 davinci

<http://support.spectrumdigital.com/>

(6) Blackhawk - 有些 560 的仿真器需要从这儿下载驱动和升级

<http://www.blackhawk-dsp.com/>

1.2.5 开源社区 beagleboard 相关资源

(1) beagleboard

beagleboard 主页:

<http://beagleboard.org/>

beagleboard wiki 主页:

<http://elinux.org/BeagleBoard>

beagleboard 代码下载:

<http://code.google.com/p/beagleboard>

beagleboard 硬件手册:

http://www.beagleboard.org/uploads/Beagle_HRM_B4.pdf

beagleboard 硬件设计资料 (包括原始原理图和 allegro PCB):

<http://beagleboard.org/hardware/design>

(2) OpenPandora

OpenPandora 相关资源:

<http://openpandora.org/>

OpenPandora WIKI:

http://pandorawiki.org/Main_Page

(3) RealtimeDSP 的 miniEVM, 是 beagleboard 的低成本克隆:

http://elinux.org/Mini_Board

<http://elinux.org/MiniBoardFAQ>

kernel 更新和文件下载

<http://code.google.com/p/omapminiboard/>

1.2.6 其他相关资源

(1) OMAP3515 和 OMAP3530 带 3D 硬件加速, 采用的是 imagination 的 PowerVR IP 核, 支持 OpenGL ES1.1 和 ES2.0, 可以从下面的网址下载 SDK, 支持 linux 和 wince 版本:

<http://www.imgtec.com/powervr/insider/powervr-sdk.asp>

文档和 API 手册参考 OpenGL ES 的标准及其相关文档:

<http://www.khronos.org/opengles/>

(2) JVM : 有针对 OMAP3 平台的 JVM 可以直接用, 很方便

<http://openjdk.java.net/>

(3) 流媒体软件框架 Gstreamer

<http://gstreamer.freedesktop.org/>

基于 DMAI 的 GST 在 TI 处理器上的版本

<http://gstreamer.ti.com>

https://omapzoom.org/gf/project/gstreamer_ti/

Gstreamer porting on Google's Android

<http://groups.google.com/group/prajnashi?hl=en&pli=1>

(4) ARM 有基于 Cortex-A8 的 Codec 库, 封装进了 OpenMax 框架, 针对只有 Cortex-A8 内核的 OMAP 处理器是很有用的, 因为该 Codec 库使用了 Neon 加速器, 可以做 Player 用。

<http://en.wikipedia.org/wiki/OpenMAX>

1.2.7 一些图形文件系统移植:

(1) Android

http://elinux.org/Android_on_OMAP

<http://beagleboard.org/project/android/>

<https://omapzoom.org/gf/project/omapandroid/wiki/>

<http://labs.embinux.org/android-porting-on-beagle.html>

(2) Debian

<http://elinux.org/BeagleBoardDebian>

(3) Ubuntu

<http://elinux.org/BeagleBoardUbuntu>

(4) Angstrom

<http://www.angstrom-distribution.org/demo/beagleboard>

1.3 如何查看 OMAP3530 的手册

TI 的东西很好，但是手册很多，并且东一个西一个的，让人不知道该如何看。下面总结一下：

(1) 如何操作处理器内部资源

OMAP35xx 是一系列处理器，它们的内部寄存器的地址、功能等是差不多的，因此要想了解内存映射或知道如何操作 OMAP3530 的 GPIO、定时器等，需要查看《OMAP35x Applications Processor Technical Reference Manual》(文档号 SPRUF98)，这个手册看着挺吓人的，3000 多页，但是并不需要全部看，需要用到什么看什么。

此外，TPS659xx 也有部分 GPIO，如何操作 TPS659xx 的内部资源，首先要关心 OMAP3530 与 TPS659xx 如何使用 I2C 通信的；其次，再关心 TPS659xx 的内部寄存器是怎么样的。有关文档《TPS65930/TPS65920 OMAP™ Power-Management and System Companion Devices Technical Reference Manual》(文档编号 SWCU052) 或《TPS65950 OMAP™ Power-Management and System Companion Devices Technical Reference Manual》(文档编号 SWCU050)。

(2) 芯片的硅片版本 Silicon Revision 不一样，内部寄存器地址等可能不一样，OMAP3530 的版本是 3.1，(1) 中提到的这个文档的内容对 OMAP3530 完全适用，暂时不需要去关心 Silicon Revisions 的文档，在以后看 u-boot 或 linux 代码时遇到硅片版本条件判断的代码，按照版本 3.1 的那条路径走就可以了。

(3) 各个芯片都有一个数据手册，这个手册是讲芯片的电气特性，封装等信息，对硬件制作有用，只搞软件就不用看了，下面列举这些手册：

- 《OMAP3530/25 Applications Processor》(文档编号: SWCS032)
- 《TPS65930/TPS65920 Integrated Power Management\Audio Codec (TPS65930 Only) Data Manual》(文档编号: SWCS037)

- 《TPS65950 Integrated Power Management/Audio Codec Data Manual》(文档编号: SWCS032)

特别需要注意的是OMAP3530有多个封装,有些封装甚至在芯片的顶面也有引脚,我们常用的是CUS封装的芯片。

(4) TI提供了几个比较有用的文档,在画原理图和布线时可以参考:

- 《Powering OMAP™3 With TPS65930/20: Design-In Guide》(文档编号: SWCU059) 这个文档是告诉TPS65930与OMAP3530如何连线。
- 《采用TPS65950为OMAP™3供电:应用设计指南》(文档编号: ZHCU013) 这个文档是告诉TPS65950如何与OMAP3530连线,中文版的资料。
- 《TPS65930/TPS65920 Layout Guide》(文档编号: SWCU058) 或
《TPS65930/TPS65920 Layout Guide》(文档编号: SWCU055)
这两个文档对电源芯片的布线进行指导。
- 《OMAP3530 Easy CUS Package PCB Escape Routing》(文档编号: SPRAB13) 这个文档对CUS封装的OMAP3530如何布线进行指导。

(5) 对上面的内容总结一下

由于我比较感兴趣的是比较偏底层的東西,上面只列举了部分文档,还有许多文档,例如与DSP有关的、与多媒体有关的,以及一些软件和源代码均可在<http://www.ti.com/omap3530>上找到。所有有关寄存器如何设置的资料全部在名字中有“Technical Reference Manual (技术参考手册)”的手册中,数据手册只是电气特性、封装等信息。

1.4 OMAP3530 的内存映射

按照习惯,在学习一款新的处理器时应该先把内存是如何分配的搞清楚,具体的信息可以参考《OMAP35x Applications Processor Technical Reference Manual》(文档号SPRUF98)(这个文档会被经常用到,以后简称《OMAP35x TRM》)的第二章(Memory Mapping)的表2-1(199页)。

4GB的内存刚好被分为4个部分,每部分1GB。

- (1) Q0: GPMC的空间,主要用于Flash;
- (2) Q1: 主要是芯片内部的资源。起始(0x4000 0000)的80KB为内部的ROM,存有固件,有了这个固件,我们才可以从Nand Flash或SD卡启动。之后从0x4020 0000开始的64KB为内部的SRAM,OMAP3530的一级启动代码就被加载到这个SRAM中运行,之后的L4部分的寄存器用于外设,L3部分的寄存器用于系统控制,还有一个IVA2.2是DSP的寄存器。
- (3) Q2: 外接的RAM就从这里开始,但是只有两个Bank,每个Bank最大接512MB。
- (4) Q3: 是一些虚拟地址,可以映射到别处,想了一想,这样还是比较合理的,因为在ARM系统中,Linux使用的是高地址的中断向量(0xFFFF 0000开始),刚好座落在这个地方,通

过设置，可以把这处的地址映射到别出去。

1.5 OMAP 平台的一些名词

- (1) GPMC: General-Purpose Memory Controller, 通用的存储器控制器, 用于连接Nand Flash、SRAM、Nor Flash、OneNand、网络控制器 (Lan9220);
- (2) SDRC: SDRAM Controller, 即连接SDRAM的接口;
- (3) PCRM: Power, Reset and Clock Management, 即电源、复位和时钟管理模块。