

AM37x
Cortex-A8
DM37x

ARM Cortex-A8处理器 原理与应用

——基于TI AM37x/DM37x处理器

李 宁 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

ARM Cortex - A8 处理器原理与应用

——基于 TI AM37x/DM37x 处理器

李 宁 编 著

北京航空航天大学出版社

内 容 简 介

本书介绍了 TI 公司 AM37x/DM37x 处理器的内核以及片上外围子系统的工作原理,并以 Embest 公司的 Devkit 8500 开发套件为对象,介绍了 AM37x/DM37x 处理器上 Android 操作系统移植与应用开发的基本过程。

全书分 12 章,可以分为 3 个部分。第一部分包括第 1~4 章,介绍了 Cortex-A8 处理器的内核结构和编程模型。第二部分包括第 5~8 章,介绍 AM37x/DM37x 处理器上各子系统的工作原理。第三部分包含第 9~12 章,介绍 Android 系统在 AM37x/DM37x 处理器上的移植和应用开发过程。

本书既可作为从事 Cortex-A8 处理器系统开发工程师的参考手册,也可作为高校嵌入式专业研究生的参考书。

图书在版编目(CIP)数据

ARM Cortex-A8 处理器原理与应用:基于 TI
AM37x/DM37x 处理器 / 李宁编著. --北京:北京航空航天大学
大学出版社,2012.4

ISBN 978-7-5124-0737-4

I. ①A… II. ①李… III. ①微处理器—系统设计
IV. ①TP332

中国版本图书馆 CIP 数据核字(2012)第 034427 号

版权所有,侵权必究。

ARM Cortex-A8 处理器原理与应用——基于 TI AM37x/DM37x 处理器

李 宁 编著

责任编辑 董立娟

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱: bhpress@263.net 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:14.5 字数:326 千字

2012 年 4 月第 1 版 2012 年 4 月第 1 次印刷 印数:4 000 册

ISBN 978-7-5124-0737-4 定价:32.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前言

2010年11月,在嵌入式系统联谊会期间,北京航空航天大学出版社的胡晓柏主任曾提出能否写一本关于Cortex-A8的书来满足读者需求。当时笔者觉得非常困难,尽管所在的UP团队有些Cortex-A8处理器应用开发方面的经验。因为Cortex-A8处理器的应用开发涉及知识面非常广,无论是内部结构还是外围电路都非常复杂,而且应用开发都是基于重量级操作系统的。从硬件设计、操作系统移植、底层驱动开发到上层应用设计,开发者所在的层面不同,对知识点的需求完全不同,因此很难写出一本全面介绍Cortex-A8处理器应用开发的书。2011年6月,Embest公司的苏昆经理也找到笔者,希望能出版一本介绍TI公司Cortex-A8处理器的书籍,并建议与TI公司、Embest公司的相关工程师共同讨论书籍的内容。在与工程师多次交流,共同确定这本Cortex-A8处理器书籍的内容大纲之后,开始了本书的编著工作,尽管仍然觉得存在诸多的困难。

本书以TI公司AM37x/DM37x处理器为对象,介绍Cortex-A8处理器的结构与应用开发,共计12章,可以分为3个部分:

第1部分包括第1~4章。介绍了Cortex-A8处理器内核,包括Cortex-A8处理器的编程模型、存储系统、异常处理机制、时钟、复位和电源控制系统。

第2部分包括第5~8章。介绍了AM37x/DM37x处理器的基本结构、存储子系统、多媒体处理部件、通信接口。由于AM37x/DM37x处理器的片上外围电路非常复杂,各个子系统的内容都很丰富,而且多数使用者都无须太关心这些子系统的内部编程,因此本书对其中部分子系统都只是做了特征功能的介绍。

第3部分包括第9~12章。以Embest公司的Devkit 8500开发套件为对象,介绍AM37x/DM37x处理器上Android操作系统移植与应用开发的基本过程。

本书写作过程中得到各方面的支持和帮助。首先,得到了TI公司和Embest公司的大力支持,在此要对TI公司亚太区大学计划部沈洁经理,Embest公司的张国瑞、苏昆、朱大鹏、李君荣等资深工程师表示感谢。其次,要感谢武汉理工大学计算机科学与技术学院UP团队的硕士研究生:李明、成虎超、卢涛、姚金波、周成、胡飞、瞿华洲、王德锐、贺勇,他们完成了大量而繁杂的资料收集整理工作,本书是他们汗水的结晶。最后要感谢北京航空航天大学出版社的编辑,他们在本书的内容安排、文字校对以及出版等方面给了作者大量有益的建议和帮助。另外,本书借鉴和使用ARM公司网站的内容、TI公司网站内容、Embest公司资料,这些均已得到了ARM公司、TI公司和Embest公司的授权。

正如前所叙述,本书所涉及知识点非常多,有些内容笔者也未能充分领会、有些专业词汇一时无法找到贴切的中文翻译,加上写作时间非常仓促,书中难免会有一些错误。尽管这是一本存在诸多缺憾和不满意的书,但是为了和大家共同进步,就权且算作抛砖引玉吧,敬请各位读者批评指正。

武汉理工大学 计算机科学与技术学院

李 宁 博士

2012 年 1 月



第 1 章 Cortex - A8 处理器简介	1
1.1 Cortex - A8 处理器特点	1
1.2 Cortex - A8 处理器基本结构	2
1.3 AM37x/DM37x 系列处理器	4
1.4 AM37x/DM37x 处理器基本结构	9
1.5 AM37x/DM37x 处理器开发工具	11
第 2 章 Cortex - A8 处理器编程模型	13
2.1 Cortex - A8 架构与指令集	13
2.1.1 Thumb - 2 指令集	13
2.1.2 ThumbEE 指令集	14
2.1.3 Jazelle 扩展体系结构	14
2.1.4 TrustZone 安全扩展体系结构	15
2.1.5 高级 SIMD 体系结构	16
2.1.6 VFPv3 体系结构	16
2.1.7 处理器操作状态	16
2.2 数据类型与存储格式	17
2.2.1 数据类型	17
2.2.2 存储格式	17
2.3 操作模式	18
2.4 寄存器组	18
2.4.1 通用寄存器	18
2.4.2 状态寄存器	20
2.5 处理器系统地址	24
2.6 异常	24
2.6.1 异常入口	25
2.6.2 退出异常	25
2.6.3 复位	25
2.6.4 快速中断请求 FIQ	26
2.6.5 中断请求 IRQ	26
2.6.6 中止 Abort	27

2.6.7 通过 CPSR/SPSR 屏蔽不精确数据中止	28
2.6.8 软件中断指令	29
2.6.9 软件监视指令	29
2.6.10 未定义指令异常	29
2.6.11 断点指令	29
2.6.12 异常向量	30
2.6.13 异常优先级	30
2.7 安全扩展	31
2.7.1 出于安全扩展的软件考虑	31
2.7.2 出于安全扩展的硬件考虑	32
2.8 系统控制协处理器	33
第 3 章 Cortex - A8 存储管理模型	35
3.1 虚拟内存	35
3.1.1 一级页表 L1	36
3.1.2 二级页表 L2	37
3.1.3 节或页尺寸的选择	39
3.2 页表缓存 TLB	39
3.3 存储属性	41
3.3.1 访问许可	41
3.3.2 存储属性	42
3.3.3 域 ID	43
3.4 页表的使用	43
3.4.1 地址空间 ID	44
3.4.2 转换表基址寄存器 0 和 1	44
3.5 存储顺序	45
3.5.1 强顺序型和设备型	45
3.5.2 普通型	46
3.5.3 存储隔离	46
第 4 章 时钟、复位与功耗管理	48
4.1 Cortex - A8 处理器时钟系统	48
4.1.1 主要时钟域	48
4.1.2 AXI 接口时钟 ACLK	49
4.1.3 调试时钟	49
4.1.4 ATB 时钟 ATCLK	49
4.2 Cortex - A8 处理器复位系统	50
4.2.1 上电复位	50
4.2.2 软复位	51

4.2.3 APB 和 ATB 复位	52
4.2.4 硬件 RAM 阵列复位	52
4.2.5 存储器阵列复位	53
4.3 Cortex-A8 处理器功耗控制	53
4.3.1 动态功耗管理	53
4.3.2 静态功耗管理或漏电功耗管理	56
第 5 章 AM37x/DM37x 处理器基础	59
5.1 电源复位时钟管理模块 PRCM	59
5.1.1 PRCM 的特点与结构	59
5.1.2 PRCM 的功能	64
5.2 MPU 子系统	71
5.2.1 MPU 子系统结构	72
5.2.2 MPU 各部件功能	73
5.3 互联器子系统	74
5.3.1 术 语	74
5.3.2 处理器内互联器架构	76
5.3.3 L3 互联器	76
5.3.4 L4 互联器	78
5.4 中断控制器	80
5.4.1 概 述	80
5.4.2 MPU INTCPS	82
5.4.3 中断处理过程	83
第 6 章 AM37x/DM37x 处理器存储系统	86
6.1 内存映射	86
6.1.1 全局内存映射	86
6.1.2 L3 和 L4 内存空间映射	88
6.1.3 IVA2.2 子系统内存空间映射	89
6.2 内存子系统	90
6.2.1 通用内存控制器 GPMC	90
6.2.2 SDRAM 控制器 SDRC	93
6.2.3 片上存储器子系统 OCM	94
6.3 内存管理单元 MMU	95
6.4 外部存储卡接口	97
第 7 章 AM37x/DM37x 处理器多媒体系统	101
7.1 IVA2.2 子系统	101
7.1.1 概 述	101
7.1.2 功能特征	102

7.1.3 硬件请求	104
7.1.4 内部结构	105
7.2 SGX 子系统	112
7.2.1 功能特征	113
7.2.2 内部结构及组成	113
7.3 摄像头图像信号处理器	115
7.3.1 功能特征	116
7.3.2 内部结构及组成	119
7.4 显示子系统	128
7.4.1 简介	128
7.4.2 内部结构及功能	128
第 8 章 AM37x/DM37x 处理器通信接口	135
8.1 多主机高速 I ² C 接口	135
8.1.1 概述	135
8.1.2 功能特征	136
8.2 HDQ/1-Wire 总线模块	137
8.2.1 概述	137
8.2.2 功能特征	138
8.3 UART/IrDA/CIR 通信模块	139
8.3.1 概述	139
8.3.2 功能特征	140
8.4 多通道 SPI 接口	141
8.4.1 概述	141
8.4.2 功能特征	142
8.5 多通道缓冲串行端口 McBSP	142
8.5.1 概述	142
8.5.2 功能特征	142
8.5.3 SIDETONE 核	144
8.6 USB OTG 控制器和 USB 主机子系统	145
8.6.1 高速 USB OTG 控制器	145
8.6.2 高速 USB 主机子系统	147
第 9 章 DevKit8500 评估套件	150
9.1 外围芯片	152
9.1.1 TPS65930	152
9.1.2 MT29C4G96MAZAPCJA-5	152
9.1.3 DM9000	152
9.1.4 FE1.1	153

9.1.5	TFP410	153
9.1.6	MAX3232	153
9.2	外围接口	153
第 10 章	Android 操作系统基础	161
10.1	Android 操作系统简介	161
10.1.1	Android 版本历史	162
10.1.2	开放手机联盟	164
10.2	Android 基本架构	165
10.3	Android 源码结构	166
10.3.1	核心工程	166
10.3.2	扩展工程	167
10.3.3	Java 程序包	168
10.4	init 进程	168
10.4.1	init 可执行程序	169
10.4.2	启动脚本 init.rc	170
10.5	shell 工具	172
10.5.1	sh 程序	172
10.5.2	命令工具箱 Toolbox	173
10.6	几个重要系统进程	174
10.6.1	ServiceManager 进程	175
10.6.2	Mediaserver 进程	176
10.6.3	Zygote 进程	176
10.6.4	SystemService 进程	176
第 11 章	Android 操作系统移植	178
11.1	Ubuntu 的安装与配置	178
11.1.1	软件获取	178
11.1.2	创建虚拟机	178
11.1.3	安装 Ubuntu	182
11.2	Android 代码的获取与提交	187
11.2.1	工具配置	187
11.2.2	获取 Android 源代码	190
11.2.3	源代码基本结构	191
11.2.4	提交修改后的源代码	192
11.3	编译 Android 系统	193
11.3.1	描述文件	193
11.3.2	编译过程	194
11.3.3	编译结果	195

11.3.4 系统烧写与运行	196
11.4 基于 Devkit8500 的 Android 系统开发	197
11.4.1 获取 Android 源码	197
11.4.2 编译过程	197
11.4.3 制作文件系统	197
11.4.4 烧写 Android 系统	199
第 12 章 Android 应用程序开发	202
12.1 Android 应用程序开发环境	202
12.1.1 JDK 获取与安装	202
12.1.2 Eclipse 的获取与安装	204
12.1.3 Android SDK 的获取与安装	204
12.2 Android 应用程序开发示例	214
12.2.1 创建新应用程序	214
12.2.2 构建用户界面 UI	216
12.2.3 运行 Android 应用程序	218
参考文献	219

第 1 章

Cortex - A8 处理器简介

2007 年 ARM 公司发布了 Cortex™-A8 微处理器,这是 ARM Cortex 新系列中第一款应用微处理器。该处理器具有出色的性能和效率,适用于各种移动和消费类应用,其中包括移动电话、机顶盒、游戏控制台和汽车导航/娱乐系统。Cortex-A8 处理器的频率可在 600 MHz 到超过 1 GHz 的范围内调节,可为苛刻的消费类应用提供高达 2 000 DMIPS 的性能,能够满足那些需要工作在 300 mW 以下的功耗优化移动设备的要求。总而言之,该处理器在大幅提高处理能力的同时仍保持了前几代移动设备处理器的功率水平,消费类应用时将受益于其更低的热耗散,同时还可降低封装和集成成本。

2011 年 6 月,德州仪器(TI)公司宣布推出采用 1 GHz ARM Cortex-A8 的 Sitara 处理器 AM37x 系列。同年 8 月,TI 公司又推出了最新 DaVinci™处理器 DM37x,其与 AM37x 处理器引脚兼容,差异在于 DM37x 处理器内增加了 800 MHz C64x+™ DSP,适用于更高品质的音视频编解码处理。

本章简要介绍 Cortex-A8 处理器的特征、基本结构,AM37x/DM37x 系列处理器的特点、应用范围以及其产品系列。

1.1 Cortex - A8 处理器特点

Cortex-A8 是第一款采用 ARMv7 架构中所有新技术的 ARM 处理器,包括了一些 ARM 第一次面世的新技术:针对媒体和信号处理的 NEON™技术;双发射、顺序超标量流水线;集成 L2 Cache;加快运行时编译器的 Jazelle® RCT 技术,如即时、动态或预先编译器。还包括最近推出的其他新技术:面向安全的 TrustZone 技术,面向代码密度的 Thumb-2 技术以及 VFPv3 浮点架构。Cortex-A8 处理器的特点如下:

- 完整的 ARMv7-A 指令集;
- 主存储器接口使用带有 AXI 接口的 AMBA 总线架构,可配置为 64 位或 128 位,支持多个未处理事务;

- 带有执行 ARM 整数指令的流水线；
- 带有执行先进 SIMD(单指令多数据)和 VFP(向量浮点)指令集的 NEON 流水线；
- 带有分支目标地址 Cache、全局历史缓存和 8 - Entry 回归堆栈的动态分支预测器；
- 带有 MMU, MMU 具有 32 个 Entry, 每个 Entry 都带有指令与数据分离的 TLB(页表缓冲)；
- L1 Cache 中指令 Cache、数据 Cache 可配置为 16 KB 或 32 KB；
- L2 Cache 可配置为 0 KB、128 KB~1 MB；
- L2 Cache 中可选配校验位和 ECC(纠错码)；
- ETM(内嵌跟踪单元)单元支持非侵入调试；
- 带有 IEM(智能能源管理)的静态和动态电源管理功能；
- 带观测点寄存器和断点寄存器的 ARMv7 调试, 采用 32 位的 APB 总线从接口与 CoreSight 调试系统连接。

Cortex - A8 处理器使用的 ARMv7 体系结构特点如下：

- 使用 ARM Thumb - 2 指令集, 集 Thumb 指令集的代码密度和 ARM 指令集的性能于一体；
- ThumbEE(Thumb 执行环境), 也就是 Jazelle RCT 技术, 可提供执行环境加速；
- 安全扩展架构, 可增强安全性, 以便实现安全领域的应用；
- 先进的 SIMD 架构, 可提高多媒体应用的性能, 例如 3D 图形、图像处理等；
- 采用 VFPv3(向量浮点架构)进行浮点计算, 与 IEEE754 完全兼容。

1.2 Cortex - A8 处理器基本结构

Cortex - A8 处理器结构如图 1 - 1 所示, 下面简要介绍其主要组成部分。

■ 取指令单元(Instruction Fetch)

取指令单元对指令流进行预测, 从 L1 指令 Cache 中取出指令后放到译码流水线中, 因此, L1 指令 Cache 也包含在取指令单元之中。

■ 指令译码单元(Instruction decode)

指令译码单元对所有的 ARM 指令、Thumb - 2 指令进行译码排序, 包括调试控制协处理器 CP14 的指令、系统控制协处理器 CP15 的指令。指令译码单元处理指令的顺序是：

- 异常；
- 调试事件；
- 复位初始化；
- 存储器内嵌自测(MBIST)；

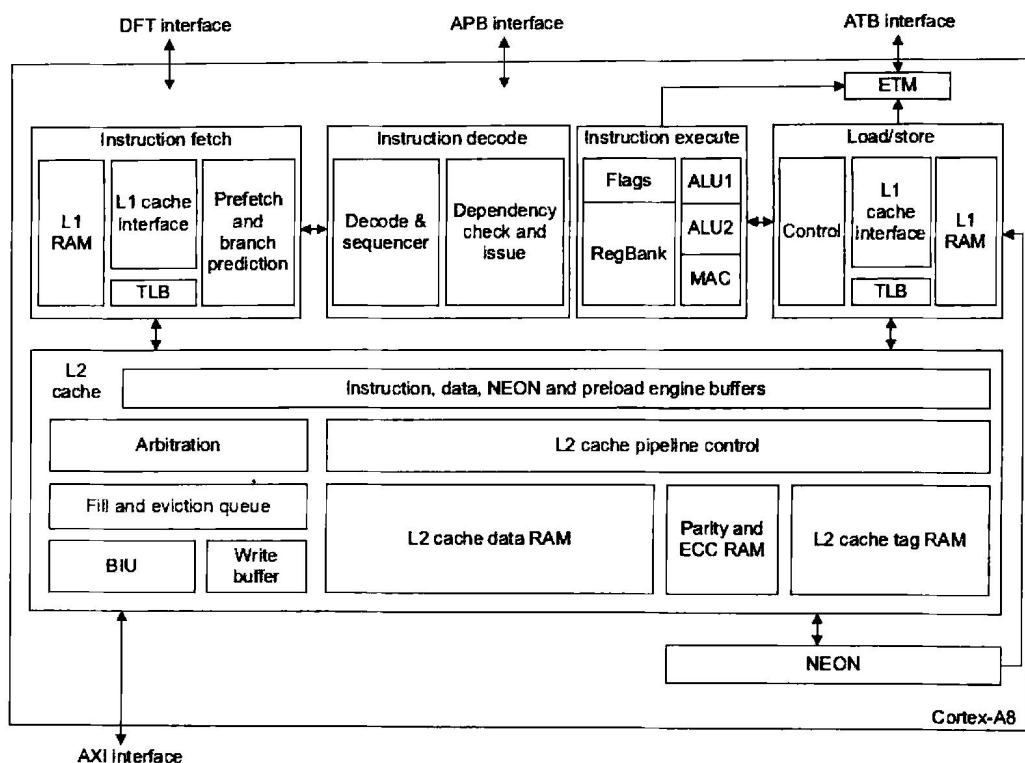


图 1-1 Cortex-A8 处理器结构框图

- 等待中断；
- 其他不常见事件。

■ 指令执行单元(Instruction execute)

指令执行单元包含两个对称的 ALU 流水线、一个用于存取指令的地址生成器和一个乘法流水线。执行单元流水线也执行寄存器回写操作。指令执行单元的功能：

- 执行所有整数 ALU 运算和乘法运算，并影响标志位；
- 根据要求产生用于存取的虚拟地址以及基本回写值；
- 将要存放的数据格式化，并将数据和标志向前发送；
- 处理分支及其他指令流变化，并评估指令条件码。

■ 数据存取单元(Load/store)

数据存取单元包含了全部 L1 数据存储系统和整数存取流水线，由以下部分组成：

- L1 数据 Cache；
- 数据 TLB；
- 整数存储缓冲；
- NEON 存储缓冲；
- 取整数数据对齐、格式化单元；

➤ 存整数数据对齐、格式化单元。

流水线可在每个周期接收一次数据存或取,可以是在流水线 0 或流水线 1 上。对于处理器而言,这将给存取指令的安排带来灵活性。

■ L2 cache

L2 Cache 单元包含 L2 Cache 和缓冲接口单元 BIU。当指令预取单元和数据存取单元在 L1 Cache 中未命中时,L2 Cache 将为它们提供服务。

■ NEON 单元

NEON 单元包含一个 10 段 NEON 流水线,用于译码和执行高级 SIMD 多媒体指令集,NEON 单元包含:

- NEON 指令队列;
- NEON 取数据队列;
- NEON 译码逻辑的两个流水线;
- 3 个用于高级 SIMD 整数指令的执行流水线;
- 2 个用于高级 SIMD 浮点数指令的执行流水线;
- 1 个用于高级 SIMD 和 VFP 的存取指令的执行流水线;
- VFP 引擎,可完全执行 VFPv3 数据处理指令集。

■ ETM 单元

ETM 单元是一个非侵入跟踪宏单元。在系统调试和系统性能分析时,使用它可以对指令和数据进行跟踪,并能对跟踪信息进行过滤和压缩。

ETM 单元通过一个称为 ATB(高级跟踪总线)的外部接口与处理器外部连接。

■ 处理器外部接口

Cortex - A8 处理器有着丰富的外部接口:

- AMBA AXI 接口。AXI 总线接口是系统总线的主要接口,64 位或 128 位,用于执行 L2 Cache 的填充和不可 Cache 指令及数据的访问。AXI 总线时钟与 CLK 输入同步,可以通过 ACLKEN 信号允许。
- AMBA APB 接口。Cortex - A8 处理器通过一个 APB 从接口来访问 ETM、CTI 和调试寄存器。APB 接口与 CoreSight 调试体系结构(ARM 多处理器跟踪调试体系)兼容。
- AMBA ATB 接口。Cortex - A8 处理器通过一个 ATB 接口输出调试跟踪信息。ATM 接口兼容 CoreSight 调试体系结构。

DFT(Design For Test)接口。DFT 接口为生产时使用 MBIST(内存内置自测试)和 ATPG(自动测试模式生成)进行内核测试提供支持。

1.3 AM37x /DM37x 系列处理器

AM37x/DM37x 处理器系列目前共有 AM3715、AM3703、DM3730 和 DM3725 这 4 种,每种处理器均有 CBP、CBC 和 CUS 这 3 种 s - PBGA 封装形式。

AM37x/DM37x 系列中 4 种处理器的配置各不相同,差异如表 1-1 所列。用户可以通过读数据寄存器 CONTROL_IDCODE(物理地址 0x4830 A204)和 CONTROL_DIE_ID(物理地址 0x4830 A218)来获取处理器的设备 ID,从而通过软件来识别芯片的类型。如需了解处理器设备 ID 与配置直接的关系,可查阅 AM37x/DM37x 处理器技术手册。

表 1-1 AM37x/DM37x 系列处理器配置差异

子系统	DM3730	DM3725	AM3715	AM3703
IVA 2.2 子系统	有	有	无	无
2D/3D 图形加速器	有	无	有	无

下面给出该系列中配置最全的 DM3730 处理器的技术特征,其他处理器可查阅所使用处理器的数据手册。

DM3730 处理器的技术特征如下:

■ 兼容 OMAP 3 体系架构。

■ ARM 微处理器(MPU):

- 高达 1 GHz 的 ARM Cortex-A8 核;
- 也支持 300、600、800 MHz 操作;
- NEON SIMD 协处理器。

■ 高性能图像、视频、音频加速子系统(IVA2.2TM):

- 高达 800 MHz 的 TMS320C64+DSP 核;
- 也支持 260、520 和 660 MHz 的操作;
- 增强 DMA 控制器(128 独立通道);
- 视频硬件加速器。

■ POWERVR SGX 图形加速器:

- 基于 Tile 结构,能达到每秒处理 20M 个多边形(20 MPoly/s)的能力;
- 通用可扩展着色引擎,该多线程引擎合并了像素着色功能和顶点着色功能;
- 支持工业标准 API:OpenGL ES 1.1、2.0 和 OpenVG 1.0;
- 支持细粒度任务切换、负载均衡和功耗管理;
- 具有可编程高质量图形边缘反锯齿功能。

■ 先进的超长指令字的 TMS320C64+DSP 核:

- 8 个高度的功能单元;
- 6 个 ALU 单元(32/40 位),每个 ALU 都可在每个时钟周期处理单个 32 位、2 个 16 位或 4 个 8 位数据;
- 2 个乘法器,每个乘法器都可在每个时钟周期处理 4 个 16×16 位(结果为 32 位)或 8 个 8×8(结果为 16 位)位运算;
- 支持非对齐的存取结构;

- 64 个 32 位通用功能寄存器；
- 指令可压缩,以减少代码尺寸；
- 所有指令都可以是有条件执行的；
- C64+TM 的其他增强功能：
 - ✓ 保护模式操作；
 - ✓ 预测支持错误检测和程序跳转；
 - ✓ 硬件支持按模循环操作。

■ C64x+TM L1/L2 存储结构：

- 32 KB 的 L1P 程序 RAM/Cache(直接映射)；
- 80 KB 的 L1D 数据 RAM/Cache(2 路组相联映射)；
- 64 KB 的 L2 统一映射 RAM/Cache(4 路组相联映射)；
- 32 KB 的 L2 共享 SRAM 和 16 KB 的 L2 ROM。

■ C64x+L1/L2 指令集：

- 按字节编址(8/16/32/64 位数据)；
- 8 位溢出保护；
- 提供位域提取、设置和清除功能；
- 提供归一化运算、饱和运算和位计数运算；
- 具有支持复杂乘法的专门指令。

■ 外部存储器接口：

- SDRAM 控制器(SDRC)：
 - ✓ 具有 1 GB 地址空间的 16、32 位存储控制器；
 - ✓ 带有低功耗 SDRAM 接口；
 - ✓ SDRAM 存储调度器(SMS)和循环引擎。
- 通用存储控制器(GPMC)：
 - ✓ 16 位宽,地址/数据总线复用；
 - ✓ 最多 8 个片选引脚,每个片选引脚可选 128 MB 地址空间；
 - ✓ 提供与 NOR Flash、NAND Flash(带 ECC 海明码计算)、SRAM 以及伪 SRAM 的无粘性接口(即不针对某一特别类型)；
 - ✓ 提供灵活的异步协议,控制与用户定制逻辑电路(FPGA、CPLD、ASIC 等)的接口；
 - ✓ 具有地址/数据总线不复用模式(限制为 2 KB 的地址空间)。

■ I/O 端口电压:1.8 V,MMC1(多媒体卡)电压:3 V;处理器核电压:0.9~1.2 V 自适应;核逻辑电压:0.9~1.1 V 自适应。以上电压为默认的操作点电压,可以通过 SmartReflex AVS 优化到更低的电压。

■ 具有商用、工业和扩展温度等级。

■ 串行通信：

- 5 个多通道缓冲串行端口(McBSP)：