



普通高等教育“十三五”规划教材



微电子与集成电路设计系列规划教材

嵌入式系统芯片设计

——基于CKCPU

© 张培勇 严晓浪 著

张培勇 严晓浪 著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

张外借

普通高等教育“十三五”规划教材
微电子与集成电路设计系列规划教材

嵌入式系统芯片设计

——基于 CKCPU

张培勇 严晓浪 著

电子工业出版社
Publishing Industry Press

内 容 简 介

本书介绍 AMBA/AXI 总线嵌入式 CPU 片上系统硬件电路设计, 通过一系列相关实验构建完整的 SoC 硬件电路, 主要内容包括: CKCPU 简介、SoC 芯片设计入门、AXI 总线协议、AXI Master 模块设计、并行接口 LCD 和摄像头控制模块设计、AXI IIC 设计、SPI 模块设计、AHB 总线 CK803、MIPI 全高清摄像 SoC 设计、运动控制与中断、MP3 播放器设计、MJPEG 视频播放器设计。本书提供配套电子课件、程序代码、示范实验微课视频等。

本书可作为高等学校集成电路设计、电子信息工程、通信工程等专业的本科高年级学生、专业学位和学术学位研究生教材, 也可供相关领域的科技工作者学习、参考。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

图书在版编目(CIP)数据

嵌入式系统芯片设计: 基于 CKCPU / 张培勇, 严晓浪著. —北京: 电子工业出版社, 2019.3

ISBN 978-7-121-34929-4

I. ①嵌… II. ①张… ②严… III. ①集成芯片—设计 IV. ①TN430.2

中国版本图书馆 CIP 数据核字(2018)第 196834 号

策划编辑: 王羽佳

责任编辑: 底 波

印 刷: 北京捷迅佳彩印刷有限公司

装 订: 北京捷迅佳彩印刷有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编: 100036

开 本: 787×1092 1/16 印张: 25.5 字数: 652.8 千字

版 次: 2019 年 3 月第 1 版

印 次: 2019 年 3 月第 1 次印刷

定 价: 72.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: (010) 88254535, wylj@phei.com.cn。

前 言

片上系统芯片(System on Chip, SoC)设计是集成电路设计的一个重要方向。本书介绍 AMBA/AXI 总线嵌入式 CPU 片上系统硬件电路设计,通过一系列相关实验构建完整的 SoC 硬件电路。

本书取材于作者在浙江大学电气学院和信电学院开设的“嵌入式系统设计”和“片上系统芯片”课程。本课程经过多年建设,已具备成熟的教学方法,有明确的培养目标。本书可作为高等学校集成电路设计、电子信息工程、通信工程等专业的本科高年级学生、专业学位和学术学位研究生教材,也可供相关领域的科技工作者学习、参考。本书包含大量的实验内容,作者自主设计并录制了丰富的实验项目和指导视频。

在教学方法上,本书将复杂的实验内容拆分成一系列由易到难的实验,学生在每次课程的实验中都能有所收获,并且可以在最终复杂的片上系统设计中综合应用到前期学到的各知识点。

在教学安排上,可以根据教学对象和学时等具体情况对书中的内容进行删减和组合,也可以进行适当扩展,设置 32 学时。可充分利用有效的课堂教学时间,在课堂上进行相关理论课程的教学,课后安排完成相应的实验。本书提供配套电子课件、程序代码、57 个示范实验微课视频等教学辅助资料,请登录华信教育资源网(<http://www.hxedu.edu.cn>)免费注册下载。

在实验安排上,作者在实际授课中,开学第一节课就将实验设备(FPGA 开发板、调试工具、LCD、摄像头、示波器和逻辑分析仪等)登记按组发放给学生,课程结束时收回。这样学生就可以不受实验室开放时间的限制,并且充分激发了学生的兴趣,这项举措受到了学生的欢迎。

本书第 1 章由严晓浪编写,第 2~12 章由张培勇编写。全书由严晓浪和张培勇统稿。浙江大学电气学院的本科生谭天、王誉博和李宜珂设计实现了部分实验内容,信电学院的部分研究生参与设计了部分实验。电子工业出版社的王羽佳编辑为本书的出版做了大量工作。在此一并表示感谢!

由于作者学识有限,书中错误与疏漏之处在所难免,希望广大读者批评指正。

作 者

目 录

第 1 章 CKCPU 简介	1	7.2 SPI 总线加速度传感器	205
1.1 CKCPU 特性	1	7.3 加速度传感器 SoC 设计	207
1.2 CK803 体系结构简介	3	7.4 AXI 接口 SPI 控制模块设计	211
1.3 CK807 体系结构简介	3	7.5 SPI LCD 控制模块设计	228
1.4 实验环境	5	第 8 章 AHB 总线 CK803	236
第 2 章 SoC 芯片设计入门	6	8.1 AHB 总线协议	236
2.1 CKCPU SoC 软件设计环境	10	8.2 CK803 SoC	241
2.2 CKCPU 的 Hello World 实验	11	8.3 AHB to AXI Bridge	241
第 3 章 AXI 总线协议	19	8.4 AHB to AXI Bridge 模块验证	244
3.1 AXI 总线协议介绍	19	第 9 章 MIPI 全高清摄像 SoC 设计	249
3.2 AXI Lite 接口模块设计	27	9.1 MIPI 总线协议	249
3.3 AXI Lite 接口 UART 设计	30	9.2 摄像头控制	250
3.4 UART 驱动软件	64	9.3 MIPI 串行信号接收	252
第 4 章 AXI Master 模块设计	71	9.4 MIPI 信号解码	255
4.1 AXI 突发传输模式	71	9.5 RAW8 格式转换为 RGB 格式	259
4.2 HDMI 控制器硬件设计	73	9.6 RGB 格式转换为 YUV 格式	260
4.3 HDMI 初始化电路设计	77	9.7 AXI 接口 MIPI 控制器	265
4.4 HDMI 输出电路设计	80	第 10 章 运动控制与中断	276
4.5 AXI Lite 接口 HDMI 控制器	87	10.1 两轮平衡车原理	276
4.6 AXI Full 接口 HDMI 控制器	94	10.2 SoC 硬件设计	277
4.7 HDMI TMDS 编码与串行输出 电路设计	103	10.3 驱动软件设计	281
第 5 章 并行接口 LCD 和摄像头控制 模块设计	124	10.4 中断	282
5.1 并行 LCD 接口	124	第 11 章 MP3 播放器设计	290
5.2 并行 LCD 控制器设计	127	11.1 SD 卡读写	290
5.3 并行接口摄像头控制器设计	134	11.2 SD 卡控制器设计	291
第 6 章 AXI IIC 设计	146	11.3 MP3 软件解码	330
6.1 IIC 总线协议	146	11.4 PWM 音频播放	331
6.2 IIC 总线温度传感器	151	第 12 章 MJPEG 视频播放器设计	337
6.3 温度传感器 SoC 设计	155	12.1 JPEG 编码原理	337
6.4 AXI 接口 IIC 控制模块设计	163	12.2 JPEG 文件格式	354
第 7 章 SPI 模块设计	202	12.3 HLS 设计 iDCT 电路	360
7.1 SPI 总线协议	202	12.4 HLS 设计 JPEG 解码电路	376
		12.5 MJPEG SoC 设计	396

第1章 CKCPU 简介

本书介绍 AMBA/AXI 总线嵌入式 CPU 片上系统 (System on Chip, SoC) 硬件电路设计, 通过一系列相关实验构建完整的 SoC 硬件电路。

本书取材于作者在浙江大学电气学院和信电学院开设的“嵌入式系统设计”和“片上系统芯片”课程。

设计基于 CK803/CK807 的 SoC, 需要具备用 Verilog/VHDL 在 FPGA 上设计实现数字电路的知识和一定的 C 语言知识。

1.1 CKCPU 特性

CKCPU 是杭州中天微系统有限公司自主设计的 AMBA/AXI 接口的嵌入式 CPU, CKCPU 包含多种型号和多种配置, 本书以 CK803 和 CK807 为例, CK803 性能类似 ARM Cortex-M3, CK807 性能类似 Cortex-A7。

CK803 是面向控制领域的 32 位高效嵌入式 CPU 核, 具有低成本、低功耗、高代码密度等特点。CK803 采用 16/32 位混合编码指令系统, 设计了精简高效的 3 级流水线。

在典型配置下, CK803 的技术参数如表 1.1 所示。

表 1.1 CK803 的技术参数

工 艺	180nm	130nm	90nm
主频 (MHz, Worst-Case)	100	150	230
面积 (mm ²)	0.6	0.3	0.15
功耗 (mW/MHz)	0.2	0.09	0.06

CK803 的主要技术特征如下。

- 精简指令集结构 (RISC), 32 位数据, 32/16 位可变长指令。
- 哈佛结构, 独立的指令总线 and 数据总线。
- 可配置的片上存储系统, 包括 SRAM 和用户自定义存储器等。
- 支持内存访问保护, 支持 4/8 个可配置内存保护区。
- AHB Lite 指令与数据扩展总线接口, 支持用户存储子系统扩展。
- 3 级高性能流水线, 单发射机制。
- 支持内存访问保护, 支持 4/8 个可配置内存保护区。
- 内部硬件调试模块支持片上硬件调试。
- CPU 性能: 1.5 DMIPS/MHz。

CK807 是面向嵌入式系统和 SoC 应用领域的 32 位高效嵌入式 CPU 核, 具有出色的功耗与性能表现。CK807 采用了 16/32 位混合编码的 RISC 指令集, 主要面向对功耗要求严格的高端嵌入式应用, 如高清数字电视、高清机顶盒、移动智能终端、高性能通信、信息安全等。

CK807 采用诸多技术实现出现的性能功耗比。在体系结构方面, CK807 采用自主设计的体

系结构和微体系结构，并重点针对功耗进行了优化，支持短循环低功耗执行、数据高速缓存过滤访问等低功耗技术。在系统管理方面，CK807 支持通过静态设计、动态电源管理和低电压供电来减少功耗，也支持进入省电模式来节省功耗。此外，CK807 支持实时检测并关断内部空闲功能模块，进一步降低处理器动态功耗。

在典型配置下，CK807 的技术参数如表 1.2 所示。

表 1.2 CK807 的技术参数

工 艺	65nm (LP)	40nm
主频 (MHz, Worst-Case)	666	1000
面积 (mm ²)	1.2	0.8
功耗 (mW/MHz)	0.25	0.2

CK807 的主要技术特征如下。

- 精简指令集计算机结构 (RISC)。
- 32 位数据长度，32 位或 16 位可变的指令长度。
- 双发射超标量 7 级流水线，对软件完全透明。
- 按序发射，乱序完成和按序退休。
- 两级内存管理单元，实现虚实地址转换与内存管理。
- 哈佛结构，使用独立的指令总线 and 数据总线。
- 指令高缓和数据高缓大小可配置，支持 8KB、16KB、32KB、64KB。
- AMBA AHB/AXI 总线协议，支持 32/64/128 位总线宽度。
- 支持大端模式和小端模式。
- 2 级多路并行分支预测技术。
- 支持 8 入口硬件返回地址堆栈。
- 支持 8 入口间接跳转分支预测器。
- 短循环低功耗执行技术。
- 指令缓存的低功耗访问技术。
- 低功耗数据缓存过滤技术。
- 非阻塞发射，投机猜测执行。
- 并发操作数旁路技术。
- 快速的数据缓存访问技术。
- 存储器复制加速技术。
- 支持写直和写回操作的数据高速缓存。
- 内部硬件调试模块支持片上硬件调试。
- 支持快速中断，支持向量中断和自动向量中断。
- CPU 性能：2.0 DMIPS/MHz，2.5 CoreMark/MHz。

本书将介绍在 FPGA 环境下，以 CK803/CK807 为核心，构建一个完整的 SoC 硬件电路，以及如何编写这个 SoC 基本的驱动软件。本书采用的方法可用于 AMBA/AXI 接口的嵌入式 CPU，如 ARM、MIPS 和 RISC-V 等。

1.2 CK803 体系结构简介

CK803 结构框图如图 1.1 所示。

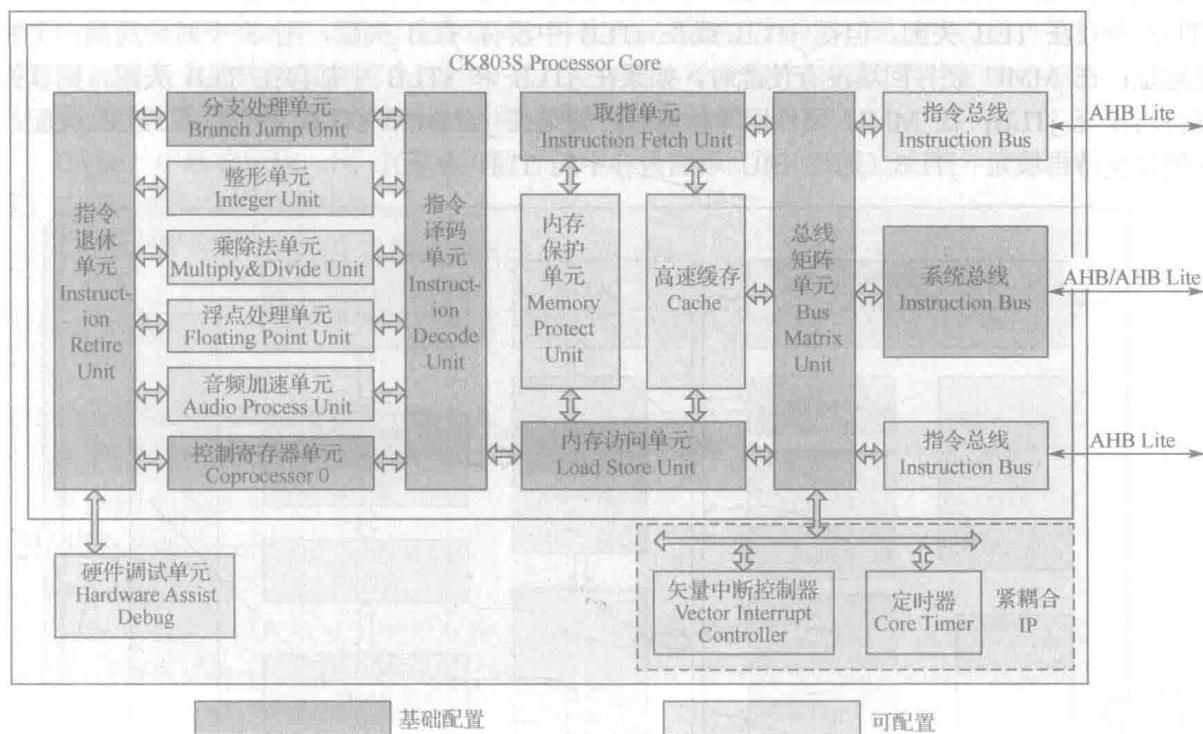


图 1.1 CK803 结构框图

CK803 设计了精简高效的 3 级流水线，分别是取指、译码与执行。对于绝大多数整型指令（包括分支指令、整型运算类指令、加载存储指令）而言，均可在 3 级流水线内完成。对于浮点与音频加速指令而言，则需要额外的一个执行周期。

CK803 体系结构细节请参阅《CK803 用户手册》。需要指出的是，设计基于 CK803 的 SoC 并不需要掌握全部的 CK803 体系结构知识，可以把《CK803 用户手册》当作一本字典来使用，后续章节将会结合具体设计实例介绍部分相关 CPU 体系结构知识。

1.3 CK807 体系结构简介

CK807 结构框图如图 1.2 所示。

CK807 处理器使用了 7 级流水线结构。

指令提取单元一次可最多提取 4 条指令并对其并行处理；可以配备高速缓存，在高速缓存缺失时采用关键指令先取和发射，以及后续指令旁路技术；配备指令暂存器缓存预取指令；采用先进 2 级指令分支跳转预测，可最多同时预测 4 条分支指令，预测精度高。整个指令提取单元拥有低功耗、高发射效率的特点。

指令译码单元可以同时同时对两条指令进行译码，并检测出指令间的数据相关性。指令译码单元根据后继流水线执行情况，及时更新指令的数据相关性信息，并将指令乱序发送至下级流水线执行。指令译码单元支持多达 5 条指令的乱序执行调度。除此之外，指令译码单元还能够分解

LDM/STM 等复杂指令，简化执行逻辑。

内存管理单元 (MMU) 具有 6 表项全相联的数据 μ TLB、4 表项全相联的指令 μ TLB 和 64/128/256 表项 2 路组相联 jTLB 及 4 表项 sTLB。 μ TLB 提升转换速度,对用户透明;jTLB 提高匹配率,用户可配置。如果在 μ TLB 中没有 TLB 失配,则当前周期产生物理地址;如果在 μ TLB 中存在 TLB 失配,但在 jTLB 或在 sTLB 中没有 TLB 失配,则 3 个时钟周期产生物理地址;在 MMU 硬件回填没有使能时,如果在 jTLB 和 sTLB 中都存在 TLB 失配,则软件支持再填充 jTLB;在 MMU 硬件回填使能时,如果在 jTLB 和 sTLB 中都存在 TLB 失配,则硬件支持再填充 jTLB (通过 BIU 取回内存中的 TLB 表项)。

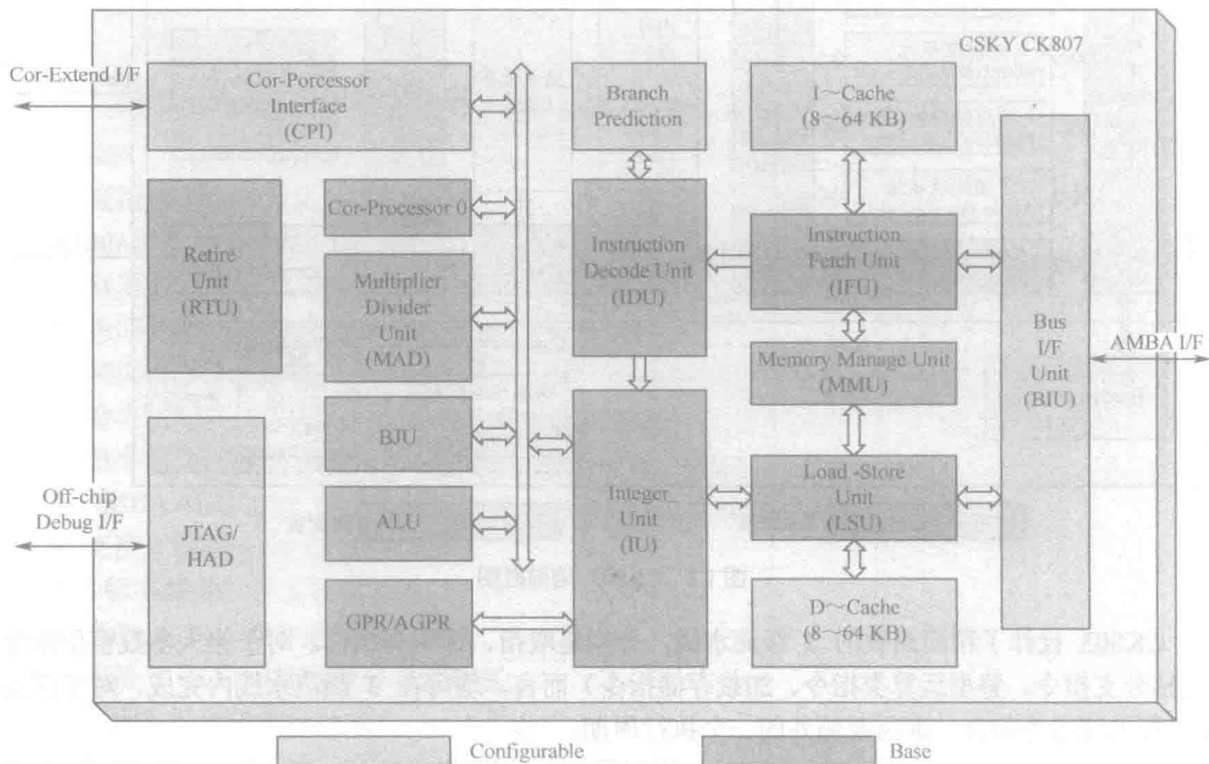


图 1.2 CK807 结构框图

存储载入单元支持存储/加载指令的按序执行,支持高速缓存的非阻塞访问。具有内部前馈机制,消除存储指令回写数据的相关性。支持字节、半字、字和双字的存储/载入指令,并支持字节和半字的载入指令的符号位和 0 扩展。支持非对齐访问。存储/加载指令可以流水执行,使得数据吞吐量达到一个周期存取一个数据。

总线接口单元 (BIU) 支持 AHB/AXI 协议,支持关键字优先的地址访问,可以在不同的系统时钟与 CPU 时钟比例 (1:1, 1:2, 1:3, 1:4, 1:5, 1:6, 1:7, 1:8) 下工作。

执行单元包含 2 条整型流水线和 1 条存储流水线。整型流水线包含 2 个算术逻辑单元 (ALU), 1 个乘除法单元 MAD 和 1 个分支跳转单元 BJU。ALU 执行标准的 32 位整数操作,单周期产生运算结果。ALU 支持快速查找 1/0 算法 (FF1/FF0), 支持移位加操作 (IXH、IXW、IXD) 等。

ALU 通过操作数前馈减少数据真相关,单周期 ALU 指令不存在数据真相关停顿延时。MAD 支持 16×16 、 16×32 、 32×32 整数乘法,支持乘累加、乘累减操作。除法器的设计采用了快速算法,执行周期 4~36 不等。

硬件辅助调试单元 (HAD) 支持各种调试方式, 包括软件设置断点方式、硬件设置断点方式、单步和多步指令跟踪、跳转指令跟踪等 6 种方式, 可以在线调试 CPU、通用寄存器 (GPR)、可选择寄存器 (AGPR)、协处理器 0 (CP0) 和内存。

指令退休单元包括一个 8 表项的重排序缓冲器, 最多支持 8 条指令的并行乱序执行。重排序缓冲器负责指令的乱序回收与按序退休, 并实现结果的按序回写。通过支持指令并行回收与快速退休提高指令退休效率和指令执行带宽。指令退休单元每个时钟周期并行退休与写回两条指令, 负责实现精确异常, 支持普通中断和快速中断。

CK807 中断响应快, 16 个硬件可配置的可选择寄存器用于减少在中断异常处理时花费的时间。支持矢量和自动矢量中断。

CK807 体系结构细节请参阅《CK807 用户手册》。

1.4 实验环境

本书硬件设计实验采用如下 FPGA 开发板:

- 中天 FPGA 开发板
- Digilent Nexys-4 DDR
- Digilent Genesys 2
- Xilinx ZC706

大部分实验可以在所有这些开发板上运行, 其他资源足够的 FPGA 开发板也可完成这些实验。本书以 Xilinx Vivado 软件 Webpack 版本作为 FPGA 设计软件。SoC 的软件开发环境使用中天 C-SKY CPU 软件开发套件 CDS Release V4.3。

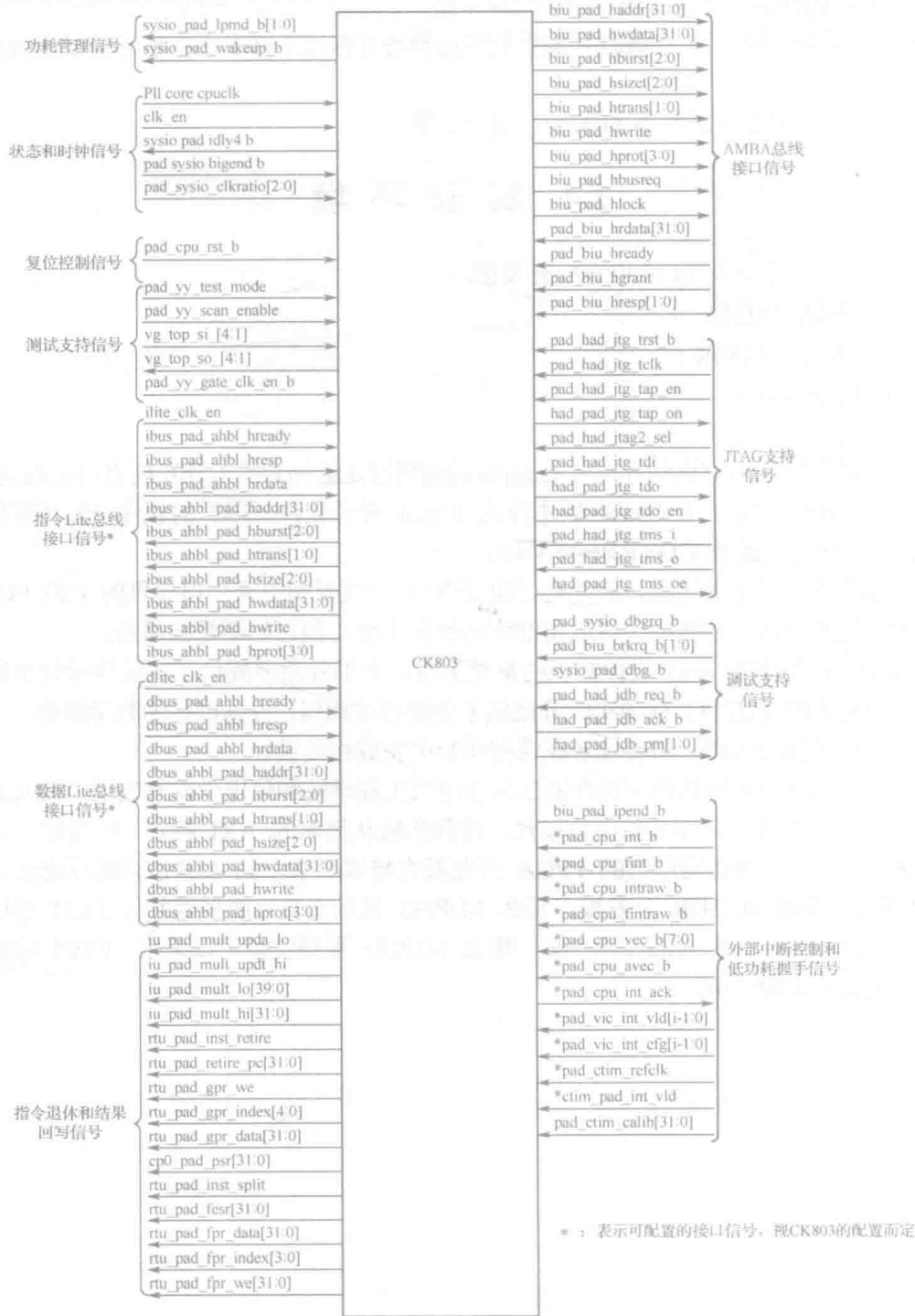
要完成本书各章节中的实验, 必须有一块 FPGA 开发板和中天公司提供的 CKCPU 开发环境。示波器和逻辑分析仪在实验中可以帮助调试硬件电路, 但它们不是必需的。

动手设计电路是掌握 SoC 设计知识的重要手段, 本书各章节提供了实验所设计电路的部分代码, 对于较复杂的 MJPEG 视频实验等提供了全部相关代码。根据以往的教学经验, 一些章节中的实验只提供了部分代码, 需要读者在课程实验中完成相关设计。

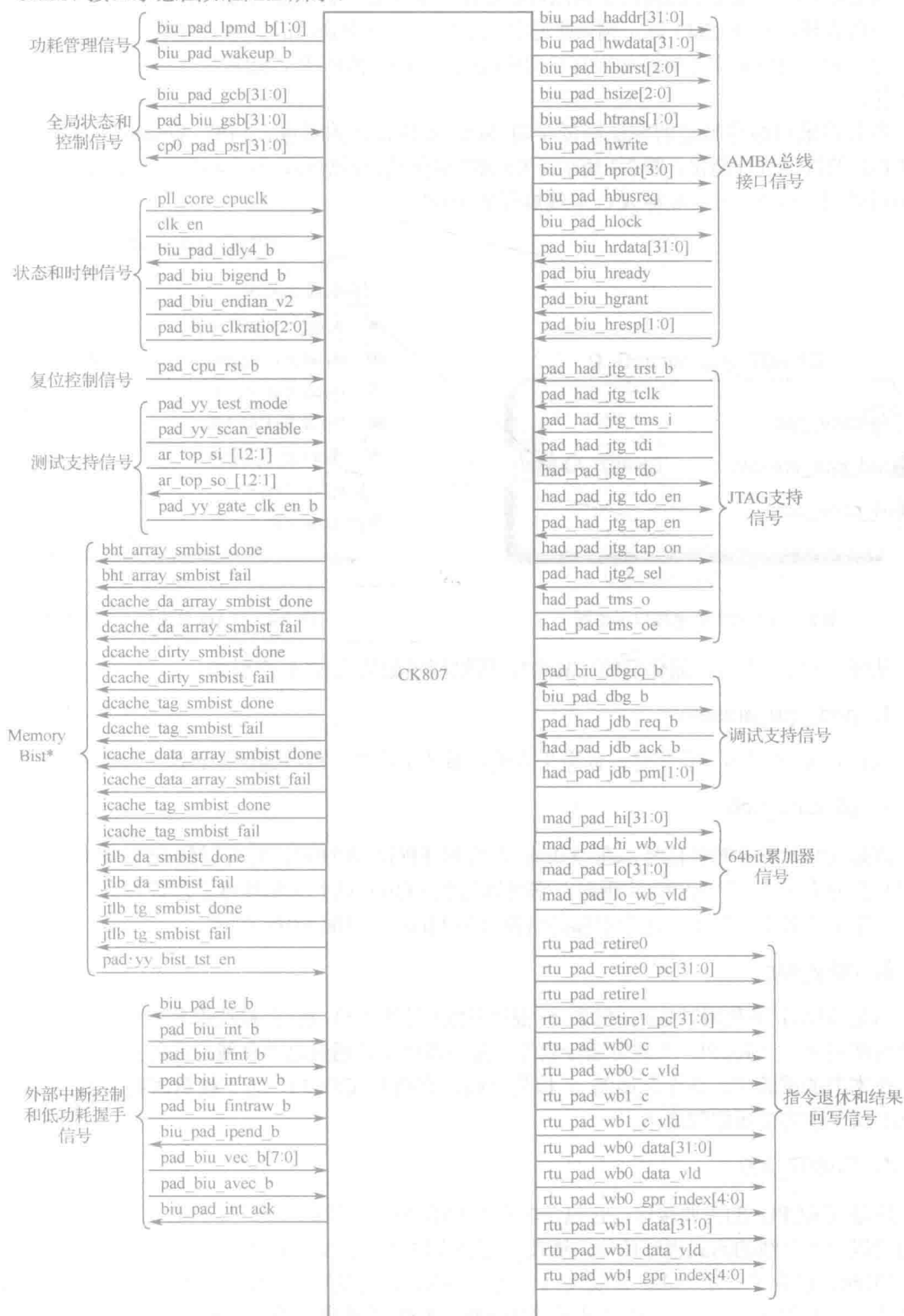
本书接下来的章节顺序按照作者在浙江大学电气工程学院和信电学院开设的“嵌入式系统设计”和“片上系统芯片”课程实验内容设置。这些实验从简单的 AXI 接口 IP 开始, 逐步完成复杂的 SoC 设计。一些实验用到的 FPGA 开发板有特殊要求, 如 MJPEG 视频输出实验, 如果采用 Digilent Nexys-4 DDR 开发板, 那么 MJPEG 视频实验只能采用并行 LCD 输出。如果采用 Digilent Genesys 2 或 Xilinx ZC706, 那么 MJPEG 视频实验可以采用 HDMI 输出。本书提供了相关实验的实验手册。

第2章 SoC 芯片设计入门

CK803 接口示意图如图 2.1 所示。



CK807 接口示意图如图 2.2 所示。



注：*表示可配置的接口信号，视 CK807 的配置而定

图 2.2 CK807 接口示意图

从 CK803 和 CK807 的接口示意图可以看出,即使是一个嵌入式 CPU,它的引脚数目也非常多。一般来说,设计 SoC 时,需要把这些 CPU 引脚接上合适的信号,这是一个复杂的工作,需要对 CPU 功能非常了解,如果有任何引脚接了不正确的信号,通常将导致这个 CPU 无法正常工作。

本书将采用循序渐进的方法降低学习 SoC 芯片设计的难度,借助 Vivado 开发环境,将 CKCPU 的接口进行简化,图 2.3 所示是 CK807 简化接口示意图,图 2.4 所示是将 JTAG 总线展开的示意图,图 2.5 所示是将 AXI 总线展开的示意图。

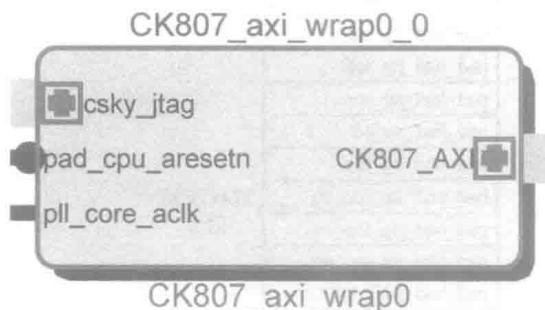


图 2.3 CK807 简化接口示意图

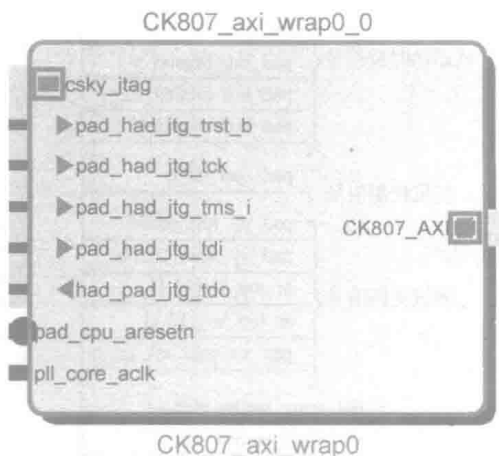


图 2.4 JTAG 总线展开的示意图

从图 2.3 可以看出,简化后的 CK807,可以简单地认为有 4 个引脚。

1. pad_cpu_aresetn

这是 CPU 的异步复位信号,低电平有效。系统上电后,可以通过控制这个引脚完成复位。

2. pll_core_aclk

这是 CPU 的主时钟信号,在 Xilinx 7 系列 FPGA 实现时,它可以到 150 MHz 左右(和 FPGA 型号有关),在 ASIC 实现时,它可以达到 1 GHz 以上(和具体工艺有关)。

在本书的各个实验里,这个引脚通常接 150 MHz 或 100 MHz 时钟。

3. csky_jtag

这是 JTAG 下载调试总线,包含 5 根信号线(见图 2.4),这 5 根线作为一个总线同时使用,这个总线用来向 CKCPU 下载待运行软件,程序调试也是通过这个总线进行的。

在本书的试验中,这个总线的 5 根信号线,直接与 CKCPU 的下载调试器连接,在 Hello World 试验中将说明如何连接。

4. CK807_AXI

这是 CKCPU 的主要接口,它包含非常多的信号线(见图 2.5),在 SoC 的设计中,这些引脚应当以一个总线的形式进行信号的连接,这极大地降低了设计难度。

CK807 包含了一个 AXI 总线接口,通过 AXI 总线接口,CPU 可以进行指令的读取和数据的读写,并和 SoC 中的各模块进行信息交互。AXI 总线将在第 3 章介绍。

CK803 采用的是 AMBA 接口,这是 AXI 的早期版本,它的优点是功耗低,适用于对性能

要求不高,对功耗和成本要求高的场合。在本书的实验中,为了降低设计难度,将 CK803 也封装成 AXI 接口,这样在用 CK807 和 CK803 进行 SoC 设计时,使用方法是一样的。

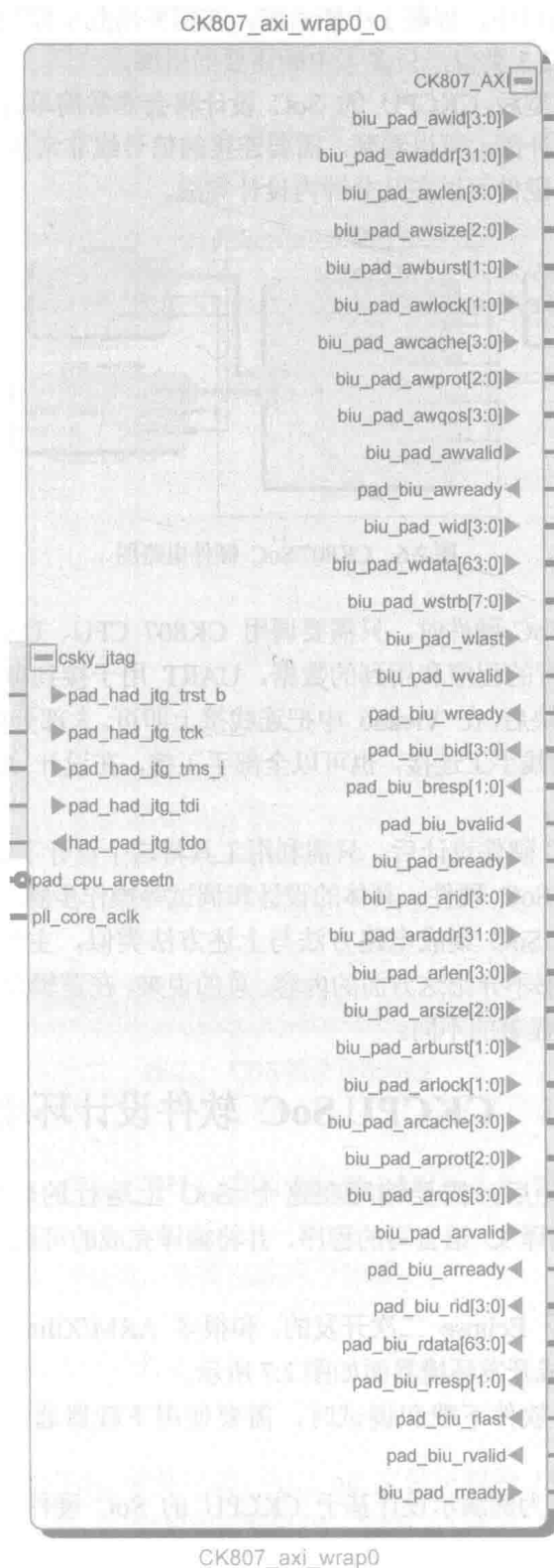


图 2.5 AXI 总线展开的示意图

本书中对性能要求不高的 SoC 实验, CK803 和 CK807 都可以使用, 对性能要求高的实验, 可以使用 CK807, CKCPU 的其他型号都可以用相同的方法进行 SoC 设计。

在图 2.3 所示的简化接口中, 屏蔽了中断引脚, 在需要用到中断的实验中, 会提供一个包含中断引脚的 CPU, 它和图 2.3 类似, 只多了中断需要的引脚。

采用了图 2.3 的简化模型后, CKCPU 的 SoC 设计将会非常简单, 图 2.6 所示是本章的 Hello World 实验的 SoC 硬件设计图, 可以看到, 需要连接的信号线非常少, AXI 总线不需要一根一根地接信号线, 整个 SoC 硬件可以在几分钟内设计完成。

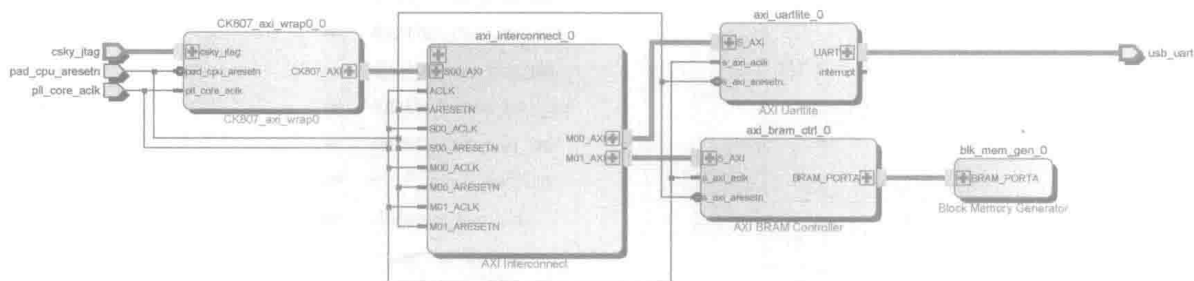


图 2.6 CK807 SoC 硬件电路图

在设计图 2.6 所示的 SoC 硬件时, 只需要调用 CK807 CPU、UART 和一个 SRAM, 其中 SRAM 用于存储 CPU 执行的程序和用到的数据, UART 用于接到串口输出, 实现串口打印调试信息的功能。调用相关模块后, 在 Vivado 中把连线接上即可。大部分内部连线可以利用 Vivado 工具自动连接, 少数剩下的线手工连接, 也可以全部手工接。在设计 ASIC 时, 这部分工作可以通过编写 HDL 代码完成。

完成图 2.6 所示的 SoC 硬件设计后, 只需利用工具将这个设计下载到 FPGA 板上, 就实现了完整的基于 CKCPU 的 SoC 硬件。具体的设计和调试等操作步骤在 2.3 节介绍。

设计基于 CKCPU 的 SoC 集成电路方法与上述方法类似, 主要的不同点在于 FPGA 和 ASIC 设计方法的不同, 本书不介绍这方面的内容。总的说来, 在逻辑设计上, FPGA 和 ASIC 这两种 SoC 的实现方法没有显著的不同。

2.1 CKCPU SoC 软件设计环境

完成 SoC 的硬件设计后, 需要编写在这个 SoC 上运行的软件, 因此需要用到 CDS Workbench 软件, 它用于编译 C 语言写的程序, 并将编译完成的可执行文件下载到 CKCPU 中调试运行。

CDS Workbench 是基于 Eclipse 二次开发的, 和很多 ARM/Xilinx 的开发工具界面类似, 使用方法基本相同。CDS 集成开发环境界面如图 2.7 所示。

CDS Workbench 进行软件下载和调试时, 需要使用下载器通过 JTAG 接口与开发板连接。

接下来以 Hello World 为例演示设计基于 CKCPU 的 SoC 硬件, 并在这个 SoC 上进行软件的开发和调试。

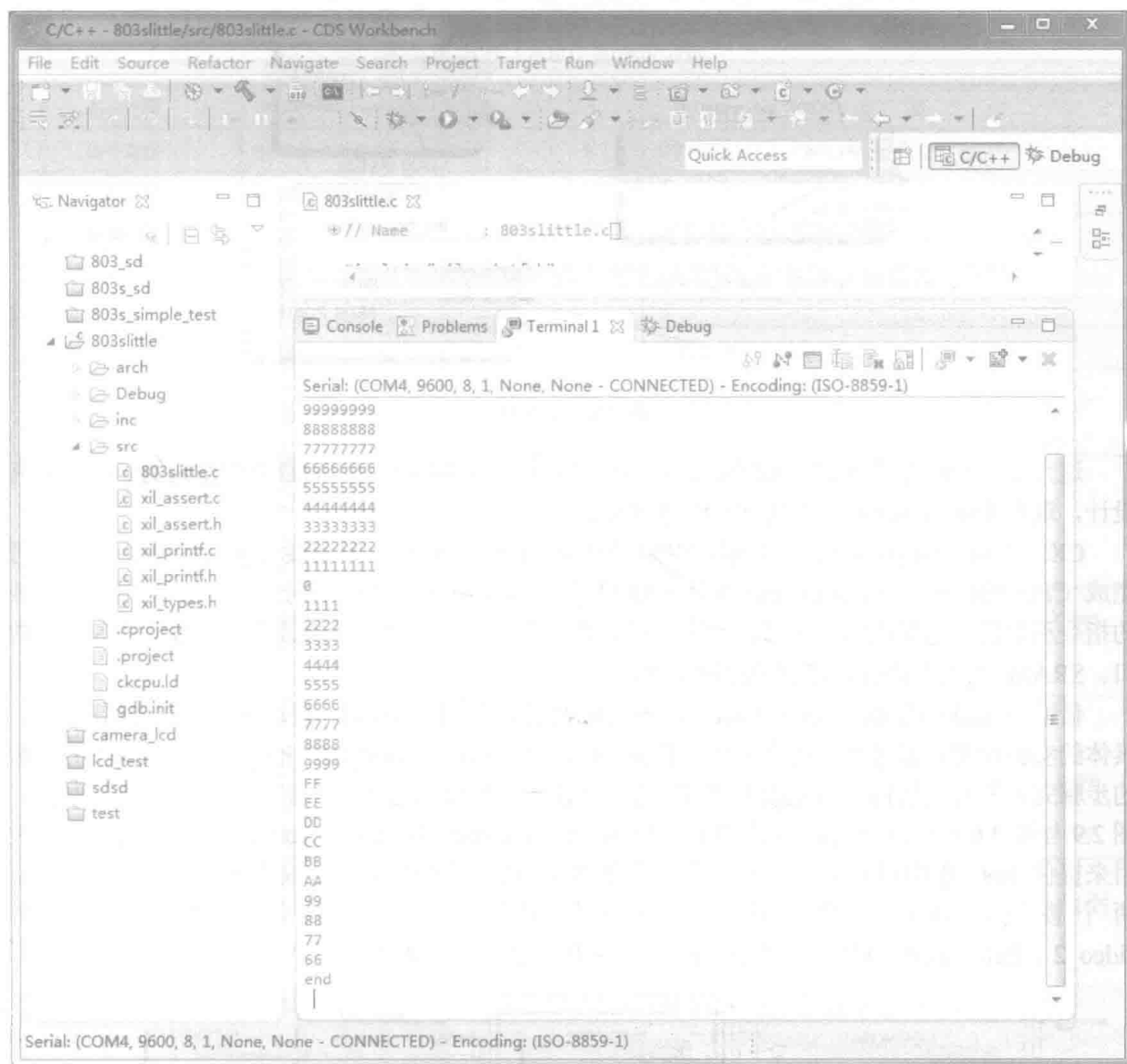


图 2.7 CDS 软件开发环境

2.2 CKCPU 的 Hello World 实验

本实验作为 SoC 设计入门实验，需要完成两个步骤。

- (1) 搭建 SoC 硬件。
- (2) 在 SoC 硬件上运行相应软件。

1. 搭建 SoC 硬件

首先搭建如图 2.6 所示的 SoC 硬件电路，生成 bit 文件下载到 FPGA 开发板。用 CK807 作为 SoC 中的 CPU，在 Vivado 开发环境中添加 IP，如图 2.8 所示。

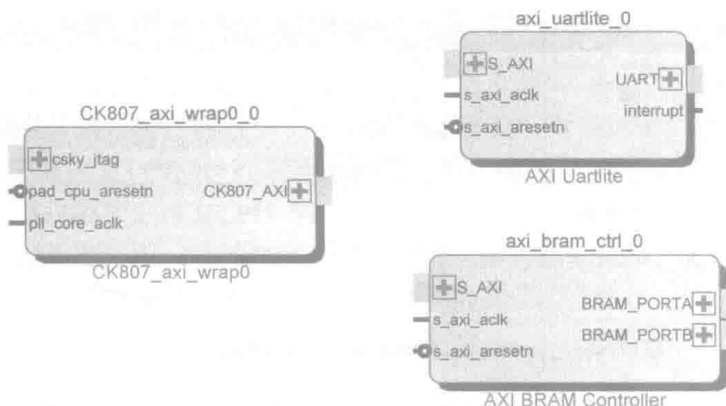


图 2.8 加入 IP

这几个 IP 包含了 SoC 基本的模块,其中 UART 模块将在后面的章节中作为例子介绍如何设计,现在先用 Vivado 自带的 IP 构建 SoC。

CK807_axi_wrap0 是已经封装好的接口简化的 CK807 CPU,只需要连接它的 4 个接口即可完成 CPU 的连接。axi_bram_ctrl_0 是 AXI 接口的 SRAM 控制器,用于存储 CPU 运行时需要的指令和数据,它默认为双端口,本实验只需要一个端口,因此可以修改它的配置,改为一个端口。SRAM 的大小通过设置它的地址实现。

借助 Vivado 的 Run Connection Automation 功能,可以让 Vivado 自动连接模块之间的连线,具体的实验步骤可参考本书配套的实验视频 video_2.1_hello_world_807_hw_1_zju.ogv,按照视频的步骤完成信号连接后,即完成了 SoC 的硬件设计,生成的电路图如图 2.9 所示,可以看出,图 2.9 与图 2.6 有一点不同,多了 Clock Wizard 和 Processor System Reset 两个模块,这两个模块用来提供 SoC 的时钟和复位信号。由于这个 SoC 的时钟和复位信号很简单,因此可以跳过这两个模块,直接从片外接入时钟信号和复位信号,具体步骤可参考视频 video_2.2_hello_world_807_hw_2_zju.ogv。这两种方法没有本质的不同。

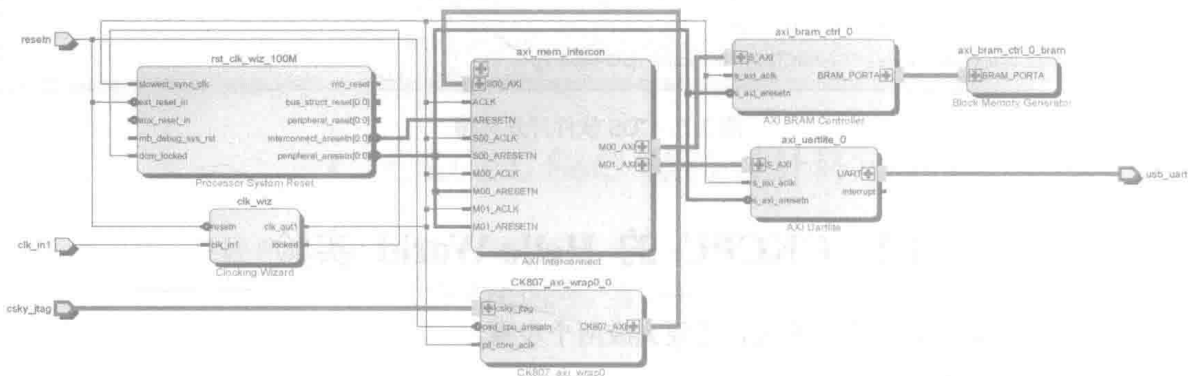


图 2.9 SoC 硬件原理图

连接各个模块后,需要给 SRAM 和 UART 指定访问地址,由于 CK807 上电后从 0x0000_0000 开始执行指令,所以需要把 axi_bram_ctrl_0 的基地址设置为 0x0000_0000,它的大小可以设置为一个合理的值,本实验需要的内存空间很小,4KB 已经足够存放指令和数据了。CK807 的地址分配如表 2.1 所示。