

第 5 章 应用系统设计与调试

本章主要介绍基于 S3C4510B 的硬件系统的详细设计步骤、实现细节、硬件系统的调试方法等,通过对本章的阅读,可以使绝大多数的读者具有根据自身的需求、设计特定应用系统的能力。

尽管本章所描述的内容为基于 S3C4510B 的应用系统设计,但由于 ARM 体系结构的一致性、以及外围电路的通用性,本章的所有内容对设计其他基于 ARM 内核芯片的应用系统,也具有很大的参考价值。

本章的主要内容包括:

- 嵌入式系统设计的基本方法。
- S3C4510B 概述。
- S3C4510B 的基本工作原理
- 基于 S3C4510B 的硬件系统设计详述
- 硬件系统的调试方法

5.1 系统设计概述

根据用户需求,设计出特定的嵌入式应用系统,是每一个嵌入式系统设计工程师应该达到的目标。嵌入式应用系统的设计包含硬件系统的设计和软件系统设计两个部分,并且这两部分的设计是互相关联、密不可分的,嵌入式应用系统的设计经常需要在硬件和软件的设计之间进行权衡与折中。因此,这就要求嵌入式系统设计工程师具有较深厚的硬件和软件基础,并具有熟练应用的能力。这也是嵌入式应用系统设计与其他的纯粹的软件设计或硬件设计最大的区别。

本章以北京微芯力科技有限公司(www.winsilicon.com)设计生产的 ARM Linux 评估开发板为原型,详细分析系统的软、硬件设计步骤、实现细节以及调试技巧等。ARM Linux 评估开发板的设计以学习与应用兼顾为出发点,在保证用户完成 ARM 技术的学习开发的同时,考虑了系统的扩展、电路板的面积、散热、电磁兼容性以及安装等问题,因此,该板也可作为嵌入式系统主板,直接应用在一些实际系统中。

图 5.1.1 是 ARM Linux 评估开发板的结构框图,各部分基本功能描述如下:

- 串行接口电路用于 S3C4510B 系统与其他应用系统的短距离双向串行通讯;
- 复位电路可完成系统上电复位和在系统工作时用户按键复位;
- 电源电路为 5V 到 3.3V 的 DC-DC 转换器,给 S3C4510B 及其他需要 3.3V 电源的外围电路供电;
- 10MHz 有源晶振为系统提供工作时钟,通过片内 PLL 电路倍频为 50MHz 作为微处理器的工作时钟;

- FLASH 存储器可存放已调试好的用户应用程序、嵌入式操作系统或其他在系统掉电后需要保存的用户数据等；
- SDRAM 存储器作为系统运行时的主要区域，系统及用户数据、堆栈均位于 SDRAM 存储器中；
- 10M/100M 以太网接口为系统提供以太网接入的物理通道，通过该接口，系统可以 10M 或 100Mbps 的速率接入以太网；
- JTAG 接口可对芯片内部的所有部件进行访问，通过该接口可对系统进行调试、编程等；
- IIC 存储器可存储少量需要长期保存的用户数据；
- 系统总线扩展引出了数据总线、地址总线和必须的控制总线，便于用户根据自身的特定需求，扩展外围电路。

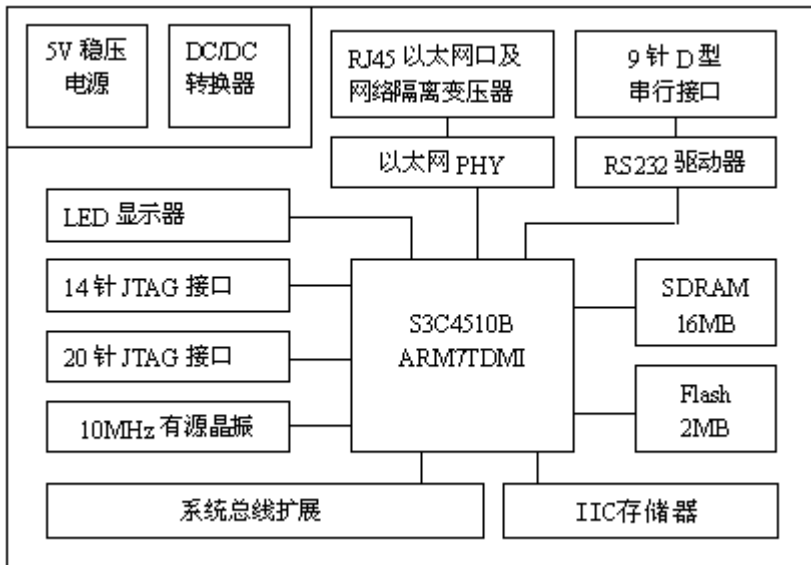


图 5.1.1 ARM Linux 评估开发板的结构框图

5.2 S3C4510B 概述

5.2.1 S3C4510B 及片内外围简介

在进行系统设计之前，有必要对 ARM Linux 评估开发板上的 ARM 芯片 S3C4510B 及其工作原理进行比较详细的介绍，读者只有对该微处理器的工作原理有了较详细的了解，才能进行特定应用系统的设计。

Samsung 公司的 S3C4510B 是基于以太网应用系统的高性价比 16/32 位 RISC 微控制器，内含一个由 ARM 公司设计的 16/32 位 ARM7TDMI RISC 处理器核，ARM7TDMI 为低功耗、高性能的 16/32 核，最适合用于对价格及功耗敏感的应用场合。

除了 ARM7TDMI 核以外，S3C4510B 比较重要的片内外围功能模块包括：

- 2 个带缓冲描述符 (Buffer Descriptor) 的 HDLC 通道
- 2 个 UART 通道
- 2 个 GDMA 通道
- 2 个 32 位定时器
- 18 个可编程的 I/O 口。

片内的逻辑控制电路包括：

- 中断控制器
- DRAM/SDRAM 控制器
- ROM/SRAM 和 FLASH 控制器
- 系统管理器
- 一个内部 32 位系统总线仲裁器
- 一个外部存储器控制器。

S3C4510B 结构框图如图 5.2.1 所示。

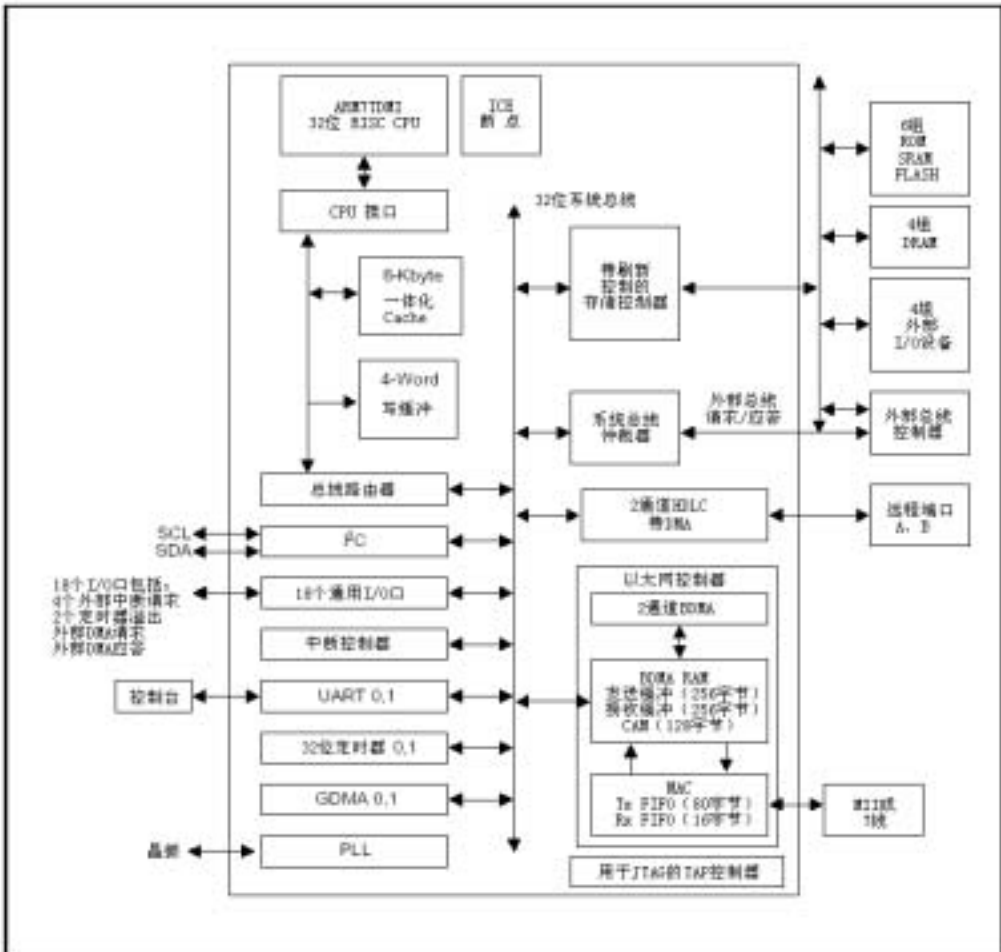


图 5.2.1 S3C4510B 结构框

S3C4510B 的特性描述如下：

体系结构

- 用于嵌入式以太网应用的集成系统
- 全 16/32 的 RISC 架构
- 支持大、小端模式。内部架构为大端模式，外部存储器可为大、小端模式
- 内含效率高、功能强的 ARM7TDMI 处理器核
- 高性价比、基于 JTAG 接口的调试方案
- 边界扫描接口

系统管理器

- 支持 ROM/SRAM、FLASH、DRAM 和外部 I/O 以 8/16/32 位的方式操作
- 带总线请求/应答引脚的外部总线控制器
- 支持 EDO/常规或 SDRAM 存储器
- 可编程的访问周期（可设定 0~7 个等待周期）
- 4 字的写缓冲
- 高性价比的从存储器到外围的 DMA 接口
- 一体化的指令/数据 Cache
 - 一体化的 8K Cache
 - 支持 LRC（近期最少使用）替换算法
 - Cache 可配置为内部 SRAM

IIC 接口

- 仅支持主控模式
- 串行时钟由波特率发生器生成

Ethernet 控制器

- 带猝发模式的 DMA 引擎
- DMA 发送/接收缓冲区（256 字节发送，256 字节接收）
- MAC 发送/接收 FIFO 缓冲区（80 字节发送，16 字节接收）
- 数据对准逻辑
- 支持端模式变换
- 100M/10Mbps 的工作速率
- 与 IEEE802.3 标准完全兼容
- 提供 MII 和 7 线制 10Mbps 接口
- 站管理信号生成
- 片内 CAM（可达 21 个目的地址）
- 带暂停特性的全双工模式
- 支持长/短包模式
- 包拆装 PDA 生成

HDLC (High-Level Data Link Control) 高层数据链路协议

- HDLC 协议特征：标志检测与同步；零插入与删除；空闲检测和发送；FCS 生成和检测（16 位）；终止检测与发送

- 地址搜索模式（可扩展到四字节）
- 可选择 CRC 模式或非 CRC 模式
- 用于时钟恢复的数字 PLL 模块
- 波特率生成器
- 发送和接收支持 NRZ/NRZI/FM/曼切斯特数据格式
- 回环与自动回波模式
- 8 字的发送和接收 FIFO
- 可选的 1 字或 4 字数据传送方式
- 数据对准逻辑
- 可编程中断
- Modem 接口
- 高达 10Mbps 的工作速率
- 基于 8 位位组的 HDLC 帧长度
- 每个 HDLC 有 2 通道 DMA 缓冲描述符用于发送和接收

DMA 控制器

- 用于存储器到存储器、存储器到 UATR、UATR 到存储器数据传送的 2 通道通用 DMA 控制器，不受 CPU 干预
- 可由程序或外部 DMA 请求启动
- 可增减源地址或目的地址，无论 8 位、16 位或 32 位数据传输
- 4 种数据猝发模式

UART

- 2 个可工作于 DMA 方式或中断方式的 UART 模块
- 支持 5、6、7、8 位的串行数据发送和接收
- 波特率可编程
- 1 位或 2 位停止位
- 奇/偶校验
- 间隔信号的生成与检测
- 奇偶校验、覆盖和帧错误检测
- $\times 16$ 时钟模式
- 支持红外发送和接收

定时器

- 2 个可编程 32 位定时器
- 间隔模式或触发模式工作

可编程 I/O 口

- 18 个可编程 I/O 口
- 可分别配置为输入模式、输出模式或特殊功能模式

中断控制器

- 21 个中断源，包括 4 个外部中断源
- 正常中断或快速中断模式（IRQ、FIQ）

— 基于优先级的中断处理

PLL

— 外部时钟可由片内 PLL 倍频以提高系统时钟

— 输入频率范围：10 ~ 40MHz

— 输出频率可以是输入时钟的 5 倍

工作电压

— 3.3V，偏差不超过 5%

工作温度

— 0°C ~ 70°C

工作频率

— 最高为 50MHz

封装形式

— 208 脚 QFP 封装

5.2.2 S3C4510B 的引脚分布及信号描述

图 5.2.2 是 S3C4510B 的引脚分布图

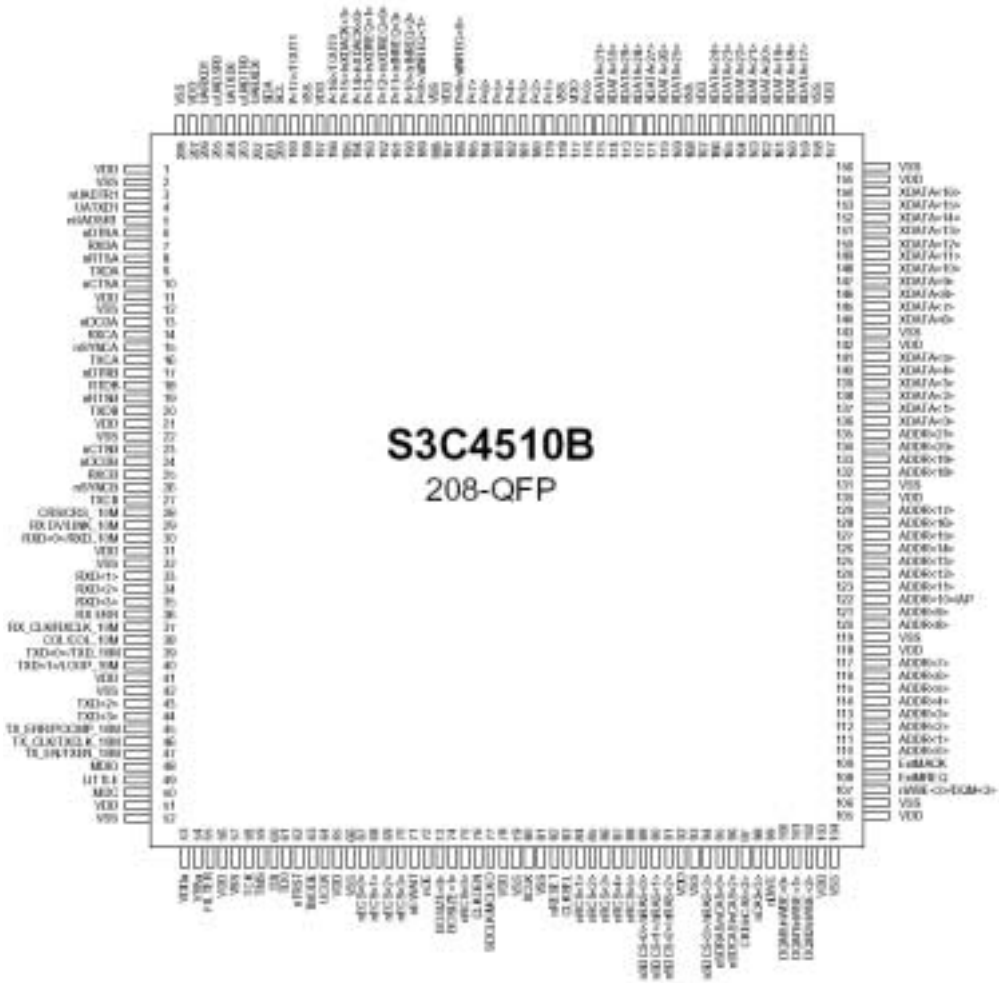


图 5.2.2 S3C4510B 的引脚分布图

各引脚信号描述如下：

表 5-2-1 S3C4510B 的引脚信号描述

| 信号 | 引脚号 | 类型 | 描述 |
|-------------|-----|----|---|
| XCLK | 80 | I | S3C4510B 的系统时钟源。如果 CLKSEN 为低电平，通过 PLL 倍频的输出时钟作为 S3C4510B 的内部系统时钟。如果 CLKSEN 为高电平，XCLK 直接作为 S3C4510B 的内部系统时钟。 |
| MCLKO/SDCLK | 77 | O | 系统时钟输出。SDCLK 为 SDRAM 提供时钟信号 |
| CLKSEL | 83 | I | 时钟选择。如果 CLKSEL 为低电平，PLL 输出时钟作为 S3C4510B 的内部系统时钟。如果 CLKSEL 为高电平，XCLK 直接作为 S3C4510B 的内部系统时钟。 |
| nRESET | 82 | I | 复位信号。nRESET 为 S3C4510B 的复位信号，要使系 |

| | | | |
|--|--|-----|--|
| | | | 统可靠复位，nRESET 必须至少保持 64 个主时钟周期的低电平。 |
| CLKOEN | 76 | I | 时钟输出允许/禁止。高电平允许系统时钟信号输出，低电平禁止。 |
| TMODE | 63 | I | 测试模式选择。低电平为正常工作模式，高电平为芯片测试模式。 |
| FILTER | 55 | AI | 如果使用 PLL，应在该引脚和数字地之间接 820pF 的陶瓷电容。 |
| TCK | 58 | I | JTAG 测试时钟。JTAG 测试时钟信号用于切换状态信息和检测数据的输入输出。该引脚在片内下拉。 |
| TMS | 59 | I | JTAG 测试模式选择。该信号控制 S3C4510B 的 JTAG 测试操作。该引脚在片内上拉。 |
| TDI | 60 | I | JTAG 测试数据输入。在 JTAG 测试操作的过程中，该信号将指令和数据串行送入 S3C4510B。该引脚在片内上拉。 |
| TDO | 61 | O | JTAG 测试数据输出。在 JTAG 测试操作的过程中，该信号将指令和数据串行送出 S3C4510B。 |
| nTRST | 62 | I | JTAG 复位信号，低电平复位。异步复位 JTAG 逻辑。该引脚在片内上拉。 |
| ADDR[21:0]/ ADDR[10]/AP | 117-110 129-120 135-132 | O | 地址总线。22 位的地址总线可寻址每一个 ROM/ SRAM 组、FLASH 存储器组、DRAM 组和外部 I/O 组的 4M 字（64M 字节）的地址范围。 |
| XDATA[31:0] | 141-136 154-144 166-159 175-169 | I/O | 外部数据总线（双向、32 位）。S3C4510B 支持外部 8 位，16 位，32 位的数据宽度。 |
| nRAS[3:0]/ nSDCS[3:0] | 94,91, 90,89 | O | DRAM 行地址锁存信号。S3C4510B 支持最多 4 个 DRAM 组，每个 nRAS 输出控制一组。nSDCS[3:0]用作 SDRAM 的片选信号。 |
| nCAS[3:0] nCAS[0] /nSDRAS nCAS[1] /nSDCAS nCAS[2]/CKE | 98,97, 96,95 | O | DRAM 列地址锁存信号。无论访问哪一个 DRAM 组，4 个 nCAS 输出信号均表示字节选择。nSDRAS 作为 SDRAM 的行地址锁存信号，nSDCAS 作为 SDRAM 的列地址锁存信号，CKE 作为 SDRAM 的时钟使能信号。 |
| nDWE | 99 | O | DRAM 写使能信号。该引脚为 DRAM 组提供写操作信号。（nWBE[3:0]用于为 ROM/SRAM/FLASH 存储器组提供写操作信号。） |
| nECS[3:0] | 70,69, 68,67 | O | 外部 I/O 片选信号。可以有 4 个外部 I/O 组映射到存储空间，每一个外部 I/O 组的地址范围最大为 16KB。nECS 提供每一个外部 I/O 组的片选信号。 |
| nEWAIT | 71 | I | 外部等待信号。该信号用于在访问外部 I/O 设备时，由外设插入等待周期。 |

| | | | |
|---------------------------------|-----------------|-----|---|
| nRCS[5:0] | 88-84,75 | O | ROM/SRAM/FLASH 片选信号。S3C4510B 可访问多达 6 个的外部 ROM/SRAM/FLASH 组。 |
| B0SIZE[1:0] | 74,73 | I | ROM/SRAM/FLASH 存储器组 0 的数据总线宽度设定。ROM/SRAM/FLASH 存储器组 0 常用于程序的启动。 '01' = 字节 (8 位) ; '10' = 半字 (16 位) ; '11' = 字 (32 位) ; '00' = 保留 |
| nOE | 72 | O | 输出使能。当对存储器进行访问的时候, 该信号控制存储器的输出使能。 |
| nWBE[3:0]/ DQM[3:0] | 107, 102-100 | O | 写字节使能。当对存储器进行写操作时, 该信号控制存储器 (DRAM 除外) 的写使能。对于 DRAM 存储器组, 由 nCAS[3:0]和 nDWE 控制写操作。DQM 用于 SDRAM 数据输入/输出的屏蔽信号。 |
| ExtMREQ | 108 | I | 外部总线控制器请求信号。外部总线控制器通过该引脚请求控制外部总线, 当该信号有效时, S3C4510B 将外部总线置为高阻状态, 以便外部总线控制器取得对外部总线的控制。当 ExtMACK 信号为的电平时, S3C4510B 重新取得对外部总线的控制权。 |
| ExtMACK | 109 | O | 外部总线应答信号。 |
| MDC | 50 | O | 管理数据时钟。该引脚产生 MDIO 数据输入输出时所需的时钟信号。 |
| MDIO | 48 | I/O | 管理数据输入/输出。当执行一个读数据的命令时, 该引脚输入由物理层产生的数据, 当执行一个写数据的命令时, 由该引脚输出数据到物理层 (PHY)。 |
| LITTLE | 49 | I | 小端模式选择引脚。当该引脚为高电平时, S3C4510B 工作在小端模式, 当该引脚为低电平时, 工作在大端模式。该引脚在片内已下拉, 因此, S3C4510B 缺省工作在大端模式。 |
| COL/COL_10M | 38 | I | 冲突检测/10M 冲突检测。该引脚显示是否检测到冲突。 |
| TX_CLK/ TXCLK_10M | 46 | I | 发送时钟/10M 发送时钟。S3C4510B 在 TX_CLK 的上升沿驱动 TXD[3:0]和 TX_EN, 当工作在 MII 模式时, PHY 在 TX_CLK 的上升沿采样 TXD[3:0]和 TX_EN。在发送数据时, TXCLK_10M 由 10M 的 PHY 产生。 |
| TXD[3:0] LOOP_10M TXD_10M | 44,43, 40,39 | O | 发送数据/10M 发送数据/10M 回环测试。TXD[3:0]为发送数据引脚, TXD_10M 为 10M 的 PHY 的发送数据引脚, LOOP_10M 由控制寄存器的回环测试位驱动。 |
| TX_EN/ TXEN_10M | 47 | O | 发送使能/10M 发送使能。 |
| TX_ERR/ PCOMP_10M | 45 | O | 发送错误/10M 包压缩使能。 |
| CRS/CRS_10M | 28 | I | 载波侦听/10M 载波侦听。 |
| RX_CLK/ RXCLK_10M | 37 | I | 接收时钟/10M 接收时钟。RX_CLK 为连续的时钟信号, 当其频率为 25MHz 时, 数据传输速率为 100M, 当其频率为 2.5MHz 时, 数据传输速率为 10M。在接收数据时, |

| | | | |
|---------------------|-------------------|-----|---|
| | | | RXCLK_10M 由 10M 的 PHY 产生。 |
| RXD[3:0] RXD_10M | 35, 34, 33, 30 | I | 接收数据/10M 接收数据。 |
| RX_DV/ LINK10M | 29 | I | 接收数据有效/10M 连接状态。 |
| RX_ERR | 36 | I | 接收错误。 |
| TXDA | 9 | O | HDLC Ch-A 发送数据。 |
| RXDA | 7 | I | HDLC Ch-A 接收数据。 |
| nDTRA | 6 | O | HDLC Ch-A 终端准备就绪。nDTRA 引脚指示数据终端设备准备发送或接收。 |
| nRTSA | 8 | O | HDLC Ch-A 传送请求。 |
| nCTSA | 10 | I | HDLC Ch-A 传送清除。 |
| nDCDA | 13 | I | HDLC Ch-A 数据载波检测。 |
| nSYNCA | 15 | O | HDLC Ch-A 同步检测。 |
| RXCA | 14 | I | HDLC Ch-A 接收时钟。 |
| TXCA | 16 | I/O | HDLC Ch-A 发送时钟。 |
| TXDB | 20 | O | HDLC Ch-B 发送数据。 |
| RXDB | 18 | I | HDLC Ch-B 接收数据。 |
| nDTRB | 17 | O | HDLC Ch-B 终端准备就绪。 |
| nRTSB | 19 | O | HDLC Ch-B 传送请求。 |
| nCTSB | 23 | I | HDLC Ch-B 传送清除。 |
| nDCDB | 24 | I | HDLC Ch-B 数据载波检测。 |
| nSYNCB | 26 | O | HDLC Ch-B 同步检测。 |
| RXCB | 25 | I | HDLC Ch-B 接收时钟。 |
| TXCB | 27 | I/O | HDLC Ch-B 发送时钟。 |
| UCLK | 64 | I | 外部 UART 时钟输入。可由外部输入时钟作为 UART 时钟，通常由系统时钟提供 UART 时钟输入。 |
| UARXD0 | 202 | I | UART0 数据接收。 |
| UATXD0 | 204 | O | UART0 数据发送。 |
| nUADTR0 | 203 | I | UART0 数据终端准备就绪。该输入信号通知 S3C4510B，外设（或其他主机）已准备好发送或接收数据。 |
| nUADSR0 | 205 | O | UART0 数据设备准备就绪。该输出信号通知外设（或其他主机），UART0 已准备好发送或接收数据。 |
| UARXD1 | 206 | I | UART1 数据接收。 |
| UATXD1 | 4 | O | UART1 数据发送。 |
| nUADTR1 | 3 | I | UART1 数据终端准备就绪。参见 nUADTR0。 |
| nUADSR1 | 5 | O | UART1 数据设备准备就绪。参见 nUADSR0。 |
| P[7:0] | 185-179, 176 | I/O | 通用 I/O 口。 |
| XINTREQ [3:0] | 191-189, 186 | I/O | 外部中断请求信号，或作为通用 I/O 口。 |

| | | | |
|--------------------------|--|-------|---------------------------|
| P[11:8] | | | |
| NXDREQ[1:0]/ P[13:12] | 193,192 | I/O | 外部 DAM 请求信号, 或作为通用 I/O 口。 |
| nXDACK[1:0]/ P[15:14] | 195,194 | I/O | 外部 DAM 应答信号, 或作为通用 I/O 口。 |
| TOUT0/P[16] | 196 | I/O | 定时器 0 溢出, 或作为通用 I/O 口。 |
| TOUT1/P[17] | 199 | I/O | 定时器 1 溢出, 或作为通用 I/O 口。 |
| SCL | 200 | I/O | I ² C 串行时钟。 |
| SDA | 201 | I/O | I ² C 串行数据。 |
| VDDP | 1, 21, 41, 56, 78, 92, 105, 118, 130, 155, 167, 177, 197 | Power | I/O 口电源。 |
| VDDI | 11, 31, 51, 65, 103, 142, 157, 187, 207 | Power | 芯片内核电源。 |
| VSSP | 2, 22, 42, 57, 79, 81, 93, 106, 119, 131, 156, 168, 178, 198 | GND | I/O 口地。 |
| VSSI | 12, 32, 52, 66, 104, 143, 158, 188, 208 | GND | 芯片内核地。 |
| VDDA | 53 | Power | PLL 电源 |
| VSSA | 54 | GND | PLL 地 |

5.2.3 CPU 内核概述及特殊功能寄存器 (Special Registers)

CPU 内核概述

S3C4510B 的 CPU 内核是由 ARM 公司设计的通用 32 位 ARM7TDMI 微处理器核，图 5.2.3 为 ARM7TDMI 核的结构框图。整个内核架构基于 RISC (Reduced Instruction Set Computer) 规则。与 CISC (Complex Instruction Set Computer) 系统相比较，RISC 架构的指令集和相关的译码电路更简洁高效。

ARM7TDMI 处理器区别于其他 ARM7 处理器的一个重要特征是其独有的称之为 Thumb 的架构策略。该策略为基本 ARM 架构的扩展，由 36 种基于标准 32 位 ARM 指令集、但重新采用 16 位宽度优化编码的指令格式构成。

由于 Thumb 指令的宽度只为 ARM 指令的一半，因此能获得非常高的代码密度。当 Thumb 指令被执行时，其 16 位的操作码被处理器解码为等效的 32 位标准 ARM 指令，然后 ARM 处理器核就如同执行 32 位的标准 ARM 指令一样执行 16 位的 Thumb 指令。也即是 Thumb 架构为 16 位的系统提供了一条获得 32 位性能的途径。

ARM7TDMI 内核既能执行 32 位的 ARM 指令集，又能执行 16 位的 Thumb 指令集，因此允许用户以子程序段为单位，在同一个地址空间使用 Thumb 指令集和 ARM 指令集混合编程，采用这种方式，用户可以在代码大小和系统性能上进行权衡，从而为特定的应用系统找到一个最佳的编程解决方案。

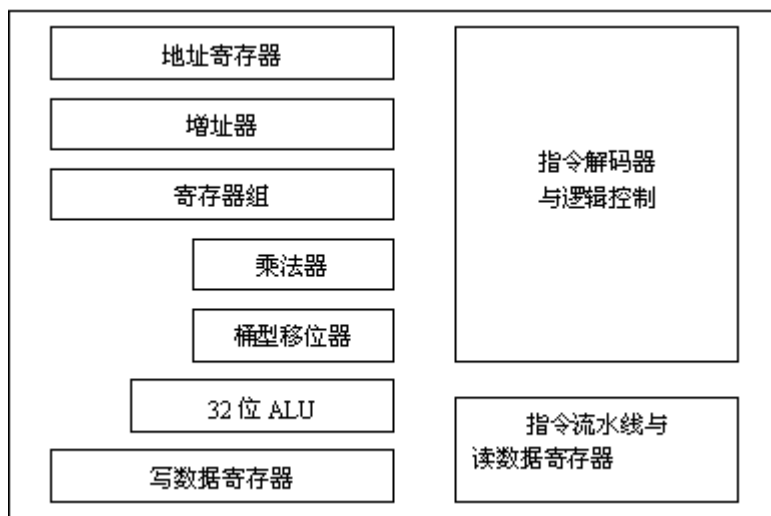


图 5.2.3 ARM7TDMI 核的结构框图

32 位的 ARM 指令集由 13 种基本的指令类型组成，可分为如下四大类：

- 4 类分支指令用于控制程序的执行流程、指令的特权等级和在 ARM 代码与 Thumb 代码之间进行切换。
- 3 类数据处理指令用于操作片上的 ALU、桶型移位器、乘法器以完成在 31

个 32 位的通用寄存器之间的高速数据处理。

- 3 类加载/存储指令用于控制在存储器和寄存器之间的数据传输。一类为方便寻址进行了优化；另一类用于快速的上下文切换；第三类用于数据交换。

- 3 类协处理器指令用于控制外部的协处理器，这些指令以开放统一的方式扩展用于片外功能指令集。

几乎所有的 32 位 ARM 指令都可以条件执行。

16 位的 Thumb 指令集为 32 位 ARM 指令集的扩展，共包含 36 种指令格式，可分为如下四个功能组：

- 4 类分支指令
- 12 类数据处理指令，为标准 ARM 数据处理指令的一个子集
- 8 类加载/存储寄存器指令
- 4 类加载/存储乘法指令

在同一种处理模式下，每一条 16 位的 Thumb 指令都有对应的 32 位 ARM 指令。

工作状态

如前所述，ARM7TDMI 内核支持两种工作状态，并总是处于其中一种工作状态。

工作状态可通过软件或异常处理进行切换：

- ARM 状态：此时执行 32 位字对齐的 ARM 指令。
- Thumb 状态：此时执行 16 位半字对齐的 Thumb 指令。

操作模式：

ARM7TDMI 内核支持 7 种操作模式：

- 用户模式：正常的程序执行状态。
- FIQ (Fast Interrupt Request) 模式：用于支持特殊的数据传送与通道处理。

- IRQ (Interrupt ReQuest) 模式：用于通用的中断处理。
- 管理模式：一种用于操作系统的保护模式。
- 中止模式：当数据或指令预取中止时进入该模式
- 系统模式：一种用于操作系统的特权用户模式。
- 未定义模式：当执行了未定义指令时进入该模式。

可用软件控制操作模式的切换，同时外部的中断和异常处理也会导致操作模式的切换。

绝大多数的用户应用程序运行在用户模式。

当系统响应中断或异常、或访问受保护的系统资源时，处理器会进入特权模式（除用户模式以外的所有模式）。

寄存器

S3C4510B 内建 37 个 32 位的寄存器：31 个通用寄存器，6 个状态寄存器，但并不是所有的寄存器都能总是被访问到。在某一时刻寄存器能否访问由处理器的当前工作状态和操作模式决定。

根据微处理器内核的当前工作状态，可分别访问 ARM 状态寄存器集和 Thumb 状态寄存器集：

- ARM 状态寄存器集包含 16 个可以直接访问的寄存器：R0 ~ R15。除 R15 以外，其余的寄存器为通用寄存器，可用于存放地址或数据值。另外一个（第 17 个）寄存器是当前程序状态寄存器 CPSR，用于保存状态信息。

- Thumb 状态寄存器集是 ARM 状态寄存器集的一个子集。可以访问的寄存器有：8 个通用寄存器 R0 ~ R7，程序计数器 PC、堆栈指针寄存器 SP、连接寄存器 LR 和当前程序状态寄存器 CPSR。

在每一种特权模式下，都有对应的分组堆栈指针寄存器 SP、连接寄存器 LR 和备份的程序状态寄存器 SPSR。

Thumb 状态寄存器集与 ARM 状态寄存器集的对应关系如下：

- Thumb 状态下 R0 ~ R7 寄存器与 ARM 状态下 R0 ~ R7 寄存器是相同的。
- Thumb 状态下的 CPSR 和 SPSRs 与 ARM 状态下的 CPSR 和 SPSRs 是相同的。
- Thumb 状态下的 SP、LR 和 PC 直接对应 ARM 状态寄存器 R13、R14 和 R15。

在 Thumb 状态下，寄存器 R8 ~ R15 不属于标准寄存器集的一部分，但在必要的情况下，用户可以通过汇编语言程序访问他们，用作快速的临时存储单元。

关于寄存器的详细描述，可参阅第二章编程模型的相关内容。

异常

当正常的程序执行流程被中断时，称为产生了异常。例如程序执行转向响应一个外设的中断请求。在优先处理异常时，处理器的当前状态必须保留，以便在异常处理完成之后程序流程能正常返回。并且，多个异常可能会同时发生。

为处理异常，S3C4510B 使用内核的分组寄存器来保存当前状态，原来的 PC 值和 CPSR 的内容被拷贝到对应的 R14（LR）和 SPSR 寄存器中，PC 和 CPSR 中的模式位被调整到相应被处理的异常类型的值。

S3C4510B 的内核支持 7 种类型的异常，每一种异常都有其固定的优先级和对应的特权处理器模式，如表 5-2-2 所示：

表 5-2-2 S3C4510B 的异常类型

| 异常 | 进入模式 | 优先级 |
|-------------------------------|--------|--------|
| 复位 (Reset) | 管理模式 | 1 (最高) |
| 数据中止 (Data Abort) | 中止模式 | 2 |
| FIQ | FIQ 模式 | 3 |
| IRQ | IRQ 模式 | 4 |
| 预取中止 (Prefetch Abort) | 中止模式 | 5 |
| 未定义指令 (Undefined Instruction) | 未定义模式 | 6 (最低) |
| SWI | 管理模式 | 6 (最低) |

S3C4510B 的特殊功能寄存器

表 5-2-3 为 S3C4510B 片内的特殊功能寄存器描述。

表 5-2-3 S3C4510B 的特殊功能寄存器

| 分组 | 寄存器 | 偏移量 | R/W | 描述 | 复位值 |
|-----------|--------|--------|-----|---------|------------|
| 系统 管理器 | SYSCFG | 0x0000 | 读/写 | 系统配置寄存器 | 0x37FFFF91 |
| | CLKCON | 0x3000 | 读/写 | 时钟控制寄存器 | 0x00000000 |

| | | | | | |
|----------------------|-----------|-------------------|-----|--------------------------|------------|
| | EXTACON0 | 0x3008 | 读/写 | 外部 I/O 时序寄存器 1 | 0x00000000 |
| | EXTACON1 | 0x300C | 读/写 | 外部 I/O 时序寄存器 2 | 0x00000000 |
| | EXTDBWTH | 0x3010 | 读/写 | 分组数据总线的宽度设置寄存器 | 0x00000000 |
| | ROMCON0 | 0x3014 | 读/写 | ROM/ARAM/FLASH 组 0 控制寄存器 | 0x20000060 |
| | ROMCON1 | 0x3018 | 读/写 | ROM/ARAM/FLASH 组 1 控制寄存器 | 0x00000060 |
| | ROMCON2 | 0x301C | 读/写 | ROM/ARAM/FLASH 组 2 控制寄存器 | 0x00000060 |
| | ROMCON3 | 0x3020 | 读/写 | ROM/ARAM/FLASH 组 3 控制寄存器 | 0x00000060 |
| | ROMCON4 | 0x3024 | 读/写 | ROM/ARAM/FLASH 组 4 控制寄存器 | 0x00000060 |
| | ROMCON5 | 0x3028 | 读/写 | ROM/ARAM/FLASH 组 5 控制寄存器 | 0x00000060 |
| | DRAMCON0 | 0x302C | 读/写 | DRAM 组 0 控制寄存器 | 0x00000000 |
| | DRAMCON1 | 0x3030 | 读/写 | DRAM 组 1 控制寄存器 | 0x00000000 |
| | DRAMCON2 | 0x3034 | 读/写 | DRAM 组 2 控制寄存器 | 0x00000000 |
| | DRAMCON3 | 0x3038 | 读/写 | DRAM 组 3 控制寄存器 | 0x00000000 |
| | REFEXTCON | 0x303C | 读/写 | 刷新与外部 I/O 控制寄存器 | 0x000083FD |
| 以太网 控制器 (BDMA) | BDMATXCON | 0x9000 | 读/写 | BDMA 接收控制寄存器 | 0x00000000 |
| | BDMARXCON | 0x9004 | 读/写 | BDMA 发送控制寄存器 | 0x00000000 |
| | BDMATXPTR | 0x9008 | 读/写 | 发送帧描述符起始地址寄存器 | 0x00000000 |
| | BDMARXPTR | 0x900C | 读/写 | 接收帧描述符起始地址寄存器 | 0x00000000 |
| | BDMARXLSZ | 0x9010 | 读/写 | 接收帧最大长度寄存器 | 未定义 |
| | BDMASTAT | 0x9014 | 读/写 | BDMA 状态寄存器 | 0x00000000 |
| | CAM | 0x9100- 0x917C | 写 | CAM 内容 (共 32 字) | 未定义 |
| | BDMATXBUF | 0x9200- 0x92FC | 读/写 | BDMA Tx 缓冲测试模式地址(64 字) | 未定义 |
| | BDMARXBUF | 0x9800- 0x99FC | 读/写 | BDMA Rx 缓冲测试模式地址(64 字) | 未定义 |
| 以太网 控制器 (MAC) | MACON | 0xA000 | 读/写 | 以太网 MAC 控制寄存器 | 0x00000000 |
| | CAMCON | 0xA004 | 读/写 | CAM 控制寄存器 | 0x00000000 |
| | MACTXCON | 0xA008 | 读/写 | MAC 发送控制寄存器 | 0x00000000 |
| | MACTXSTAT | 0xA00C | 读/写 | MAC 发送状态寄存器 | 0x00000000 |
| | MADRXCON | 0xA010 | 读/写 | MAC 接收控制寄存器 | 0x00000000 |
| | MACRXSTAT | 0xA014 | 读/写 | MAC 接收状态寄存器 | 0x00000000 |
| | STADATA | 0xA018 | 读/写 | 工作站管理数据寄存器 | 0x00000000 |

| | | | | | |
|--------------|----------|--------|--------|--------------------|----------------|
| | STACON | 0xA01C | 读/写 | 工作站管理控制与地址寄存器 | 0x00006000 |
| | CAMEN | 0xA028 | 读/写 | CAM 使能寄存器 | 0x00000000 |
| | EMISSCNT | 0xA03C | 读/写 | 错误计数寄存器 | 0x00000000 |
| | EPZCNT | 0xA040 | 读 | 中止计数寄存器 | 0x00000000 |
| | ERMPZCNT | 0xA044 | 读 | 远程中止计数寄存器 | 0x00000000 |
| | ETXSTAT | 0x9040 | 读 | 传送控制帧控制状态寄存器 | 0x00000000 |
| HDLC A 通道 | HMODE | 0x7000 | 读/写 | HDLC 模式寄存器 | 0x00000000 |
| | HCON | 0x7004 | 读/写 | HDLC 控制寄存器 | 0x00000000 |
| | HSTAT | 0x7008 | 读/写 | HDLC 状态寄存器 | 0x00010400 |
| | HINTEN | 0x700C | 读/写 | HDLC 中断使能寄存器 | 0x00000000 |
| | HTXFIFOC | 0x7010 | 读/写 | TxFIFO 帧持续寄存器 | _ |
| | HTXFIFOT | 0x7014 | 读 | TxFIFO 帧中止寄存器 | _ |
| | HRXFIFO | 0x7018 | 读 | HDLC Rx FIFO 入口寄存器 | 0x00000000 |
| | HBRGTC | 0x701C | 写 | HDLC 波特率发生时间常数寄存器 | 0x00000000 |
| | HPRMB | 0x7020 | 读/写 | HDLC 前缀常数寄存器 | 0x00000000 |
| | HSAR0 | 0x7024 | 读/写 | HDLC 站地址 0 寄存器 | 0x00000000 |
| | HSAR1 | 0x7028 | 读/写 | HDLC 站地址 1 寄存器 | 0x00000000 |
| | HSAR2 | 0x702C | 读/写 | HDLC 站地址 2 寄存器 | 0x00000000 |
| | HSAR3 | 0x7030 | 读/写 | HDLC 站地址 3 寄存器 | 0x00000000 |
| | HMASK | 0x7034 | 读/写 | HDLC 掩码寄存器 | 0x00000000 |
| | DMATxPTR | 0x7038 | 读/写 | DMA Tx 缓冲描述符指针寄存器 | 0xFFFFFFFF |
| | DMARxPTR | 0x703C | 读/写 | DMA Rx 缓冲描述符指针寄存器 | 0xFFFFFFFF |
| | | HMFLR | 0x7040 | 读/写 | 最大帧长度寄存器 |
| | HRBSR | 0x7044 | 读/写 | DMA 接收缓冲长度寄存器 | 0xXXXX0000 |
| HDLC B 通道 | HMODE | 0x8000 | 读/写 | HDLC 模式寄存器 | 0x00000000 |
| | HCON | 0x8004 | 读/写 | HDLC 控制寄存器 | 0x00000000 |
| | HSTAT | 0x8008 | 读/写 | HDLC 状态寄存器 | 0x00010400 |
| | HINTEN | 0x800C | 读/写 | HDLC 中断使能寄存器 | 0x00000000 |
| | HTXFIFOC | 0x8010 | 写 | TxFIFO 帧持续寄存器 | _ |
| | HTXFIFOT | 0x8014 | 写 | TxFIFO 帧中止寄存器 | _ |
| | HRXFIFO | 0x8018 | 读 | HDLC Rx FIFO 入口寄存器 | 0x00000000 |
| | HBRGTC | 0x801C | 读/写 | HDLC 波特率发生时间常数寄存器 | 0x00000000 |
| | HPRMB | 0x8020 | 读/写 | HDLC 前缀常数寄存器 | 0x00000000 |
| | | HSAR0 | 0x8024 | 读/写 | HDLC 站地址 0 寄存器 |

| | | | | | |
|--------|-------------|--------|-----|-------------------|--------------|
| | HSAR1 | 0x8028 | 读/写 | HDLC 站地址 1 寄存器 | 0x00000000 |
| | HSAR2 | 0x802C | 读/写 | HDLC 站地址 2 寄存器 | 0x00000000 |
| | HSAR3 | 0x8030 | 读 | HDLC 站地址 3 寄存器 | 0x00000000 |
| | HMASK | 0x8034 | 读 | HDLC 掩码寄存器 | 0x00000000 |
| | DMATxPTR | 0x8038 | 读 | DMA Tx 缓冲描述符指针寄存器 | 0xFFFFFFFF |
| | DMARxPTR | 0x803C | 读/写 | DMA Rx 缓冲描述符指针寄存器 | 0xFFFFFFFF |
| | HMFLR | 0x8040 | 读/写 | 最大帧长度寄存器 | 0xXXXX0000 |
| | HRBSR | 0x8044 | 读/写 | DMA 接收缓冲长度寄存器 | 0xXXXX0000 |
| I/O 口 | IOPMOD | 0x5000 | 读/写 | I/O 口模式寄存器 | 0x00000000 |
| | IOPCON | 0x5004 | 读/写 | I/O 口控制寄存器 | 0x00000000 |
| | IOPDATA | 0x5008 | 读/写 | I/O 口数据寄存器 | 未定义 |
| 中断控制器 | INTMOD | 0x4000 | 读/写 | 中断模式寄存器 | 0x00000000 |
| | INTPND | 0x4004 | 读/写 | 中断悬挂寄存器 | 0x00000000 |
| | INTMSK | 0x4008 | 读/写 | 中断屏蔽寄存器 | 0x003FFFFFFF |
| | INTPRI0 | 0x400C | 读/写 | 中断优先级寄存器 0 | 0x03020100 |
| | INTPRI1 | 0x4010 | 读/写 | 中断优先级寄存器 1 | 0x07060504 |
| | INTPRI2 | 0x4014 | 读/写 | 中断优先级寄存器 2 | 0x0B0A0908 |
| | INTPRI3 | 0x4018 | 读/写 | 中断优先级寄存器 3 | 0x0F0E0D0C |
| | INTPRI4 | 0x401C | 读/写 | 中断优先级寄存器 4 | 0x13121110 |
| | INTPRI5 | 0x4020 | 读/写 | 中断优先级寄存器 5 | 0x00000014 |
| | INTOFFSET | 0x4024 | 读 | 中断偏移地址寄存器 | 0x00000054 |
| | INTOSET_FIQ | 0x4030 | 读 | FIQ 中断偏移量寄存器 | 0x00000054 |
| | INTOSET_IRQ | 0x4034 | 读 | IRQ 中断偏移量寄存器 | 0x00000054 |
| IIC 总线 | IICCON | 0XF000 | 读/写 | IIC 总线控制状态寄存器 | 0x00000054 |
| | IICBUF | 0xF004 | 读/写 | IIC 总线移位缓冲寄存器 | 未定义 |
| | IICPS | 0xF008 | 读/写 | IIC 总线预分频寄存器 | 0x00000000 |
| | IICCONOUT | 0xF00C | 读 | IIC 总线预分频计数寄存器 | 0x00000000 |
| GDMA | GDMACON0 | 0xB000 | 读/写 | GDMA 通道 0 控制寄存器 | 0x00000000 |
| | GDMACON1 | 0xC000 | 读/写 | GDMA 通道 1 控制寄存器 | 0x00000000 |
| | GDMA_SRC0 | 0xB004 | 读/写 | GDMA 源地址寄存器 0 | 未定义 |
| | GDMA_DST0 | 0xB008 | 读/写 | GDMA 目的地址寄存器 0 | 未定义 |
| | GDMA_SRC1 | 0xC004 | 读/写 | GDMA 源地址寄存器 1 | 未定义 |
| | GDMA_DST1 | 0xB008 | 读/写 | GDMA 目的地址寄存器 1 | 未定义 |

| | | | | | |
|------|----------|--------|-----|-------------------|------------|
| | GDMAcnt0 | 0xB00C | 读/写 | GDMA 通道 0 传输计数寄存器 | 未定义 |
| | GDMAcnt1 | 0xC00C | 读/写 | GDMA 通道 1 传输计数寄存器 | 未定义 |
| UART | ULCON0 | 0xD000 | 读/写 | UART 通道 0 行控制寄存器 | 0x00 |
| | ULCON1 | 0xE000 | 读/写 | UART 通道 1 行控制寄存器 | 0x00 |
| | UCON0 | 0xD004 | 读/写 | UART 通道 0 控制寄存器 | 0x00 |
| | UCON1 | 0xE004 | 读/写 | UART 通道 1 控制寄存器 | 0x00 |
| | USTAT0 | 0xD008 | 读 | UART 通道 0 状态寄存器 | 0xC0 |
| | USTAT1 | 0xE008 | 读 | UART 通道 1 状态寄存器 | 0xC0 |
| | UTXBUF0 | 0xD00C | 写 | UART 通道 0 发送保持寄存器 | 未定义 |
| | UTXBUF1 | 0xE00C | 写 | UART 通道 1 发送保持寄存器 | 未定义 |
| | URXBUF0 | 0xD010 | 读 | UART 通道 0 接收缓冲寄存器 | 未定义 |
| | URXBUF1 | 0xE010 | 读 | UART 通道 1 接收缓冲寄存器 | 未定义 |
| | UBRDIV0 | 0xD014 | 读/写 | 波特率除数因子寄存器 0 | 0x00 |
| | UBRDIV1 | 0xE014 | 读/写 | 波特率除数因子寄存器 1 | 0x00 |
| 定时器 | TMOD | 0x6000 | 读/写 | 定时器模式寄存器 | 0x00000000 |
| | TDATA0 | 0x6004 | 读/写 | 定时器 0 数据寄存器 | 0x00000000 |
| | TDATA1 | 0x6008 | 读/写 | 定时器 1 数据寄存器 | 0x00000000 |
| | TCNT0 | 0x600C | 读/写 | 定时器 0 计数寄存器 | 0xFFFFFFFF |
| | TCNT1 | 0x6010 | 读/写 | 定时器 1 计数寄存器 | 0xFFFFFFFF |

5.2.4 S3C4510B 的系统管理器 (System Manager)

概述

S3C4510B 微处理器的系统管理器 (System Manager) 在整个系统工作中起至关重要作用, 只有清楚的了解系统管理器在系统中的作用及工作原理, 才能进行程序设计和系统开发, 但同时, 相对于 8 位或 16 位微处理器而言, S3C4510B 系统管理器的工作原理又是比较复杂的, 因此需要读者认真细致的阅读该部分内容, 并通过编程实践加以掌握。

S3C4510B 微处理器的系统管理器具有以下功能:

- 基于固定的优先级, 仲裁来自几个主功能模块的系统总线访问请求。
- 为访问外部存储器提供必需的存储器控制信号。例如 DMA 控制器或 CPU 要访问 DRAM 组的某地址, 则系统管理器的 DRAM 控制器就会产生必需的 normal/EDO 或 SDRAM 访问信号。可由 SYSCFG[31] 设定访问 normal/EDO 或 SDRAM 的信号。

- 为 S3C4510B 和 ROM/SRAM ,以及外部 I/O 组之间的总线通信提供必需的信号。
- 为外部存储器的数据总线和内部数据总线之间的数据流协调总线宽度的差别。
- 对外部存储器和 I/O 设备, S3C4510B 同时支持小端模式和大端模式的访问方式。

通过产生外部总线请求信号, 外设可访问 S3C4510B 的外部总线。另外, S3C4510B 可通过插入等待周期 (WAIT 信号) 访问低速外设。 WAIT 信号由外设产生, 可延长 CPU 的存储器访问周期。

系统管理器寄存器 (System Manager Registers)

系统管理器使用一组专用的特殊功能寄存器来控制外部存储器的读/写操作, 通过对该组特殊功能寄存器编程, 可以设定:

- 存储器的类型
- 外部数据总线宽度及访问周期
- 定时的控制信号 (例如 RAS 和 CAS)
- 存储器组的定位
- 存储器组的大小

在标准系统配置中访问外设必需的控制信号、地址信号和数据信号, 系统管理器通过设置特殊功能寄存器的值来控制其产生和处理。特殊功能寄存器也被用于控制对 ROM/SRAM/Flash 组的访问, 同时还能控制对多达四个 DRAM 组和四个外部 I/O 组以及一个特殊功能寄存器映射区域的访问。

每个存储器组在组内通过基指针 (Base Pointer) 寻址, 其寻址范围是 64KB (16 位), 而基指针本身为 10 位。因此 S3C4510B 的最大可寻址范围是 $2^{26} = 64\text{MB}$ (或 16M 字)。

在进行系统存储器映射时, 注意两个相连的存储器组的地址空间决不能重叠。图 5.2.4 为 S3C4510B 系统存储器映射。

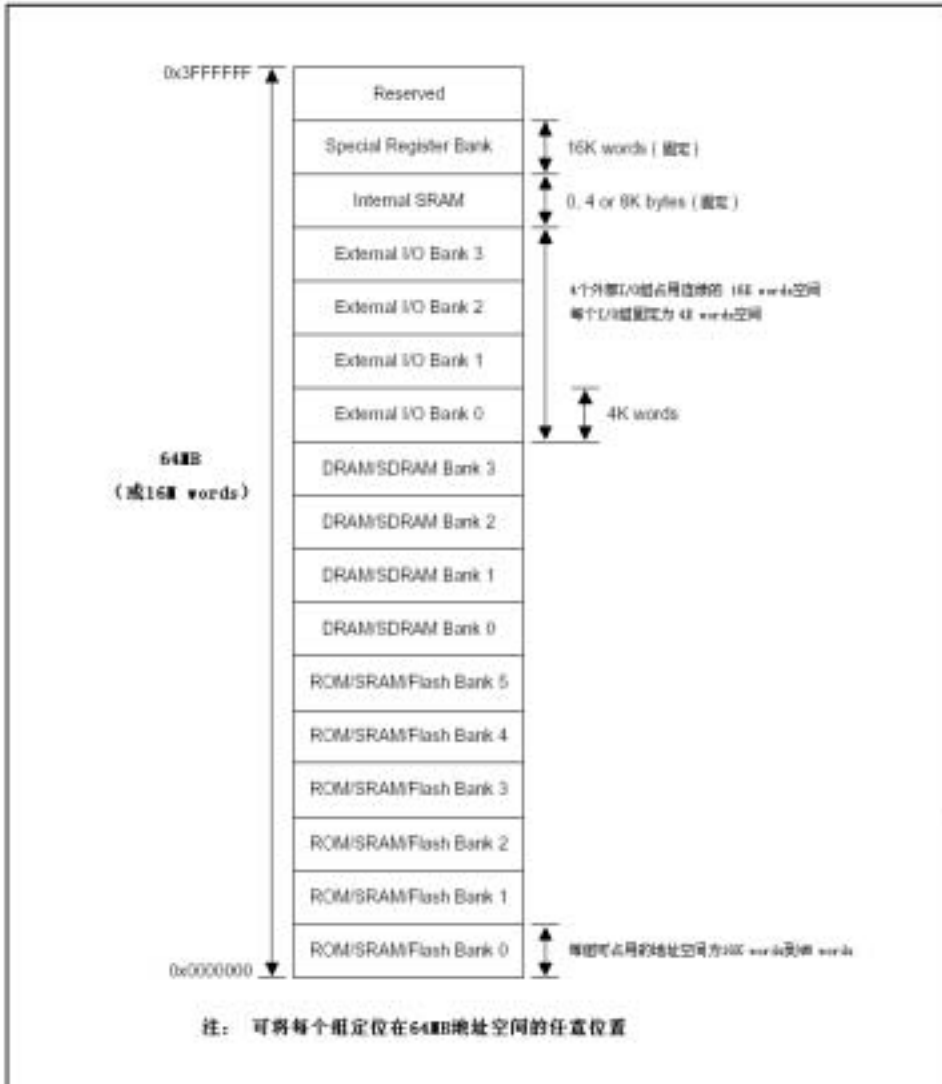


图 5.2.4 S3C4510B 系统存储器映射

系统存储器映射(System Memory Map)

关于 S3C4510B 系统存储器映射，以下是几个应当注意的重点：

- S3C4510B 采用统一编址的方式，将系统的片外存储器、片内存储器、特殊功能寄存器和外部的 I/O 设备，都映射到 64MB 的地址空间，同时，为便于管理，又将地址空间分为如图 5.2.4 所示的若干个存储器组，可以通过配置包含基指针（Base Pointer）和尾指针（End Pointer）的特殊功能寄存器，设定每个存储器组的大小和位置。用户可利用基指针和尾指针设置连续的存储器映射。具体操作如下：即把某个存储器组的基指针的地址设置为前一个存储器组的尾指针的地址。请注意在设定存储器组的控制寄存器时，每两个相连的存储器组的地址空间决不能重叠，即使这些组被禁用。

- 四个外部 I/O 组被定义在一个连续的地址空间中。只需要将基指针分配给

外部 I/O 组 0, 外部 I/O 组 1 的起始地址就等于外部 I/O 组 0 的起始地址+16KB, 同理, 外部 I/O 组 2 的起始地址就等于外部 I/O 组 0 的起始地址+32KB, 外部 I/O 组 3 的起始地址就等于外部 I/O 组 0 的起始地址+48KB。因此, 四个外部组的总的连续的可寻址范围被定义在外部 I/O 组 0 的起始地址+64KB 的地址空间。在整个可寻址的地址空间中, 外部 I/O 组的起始地址并没有被固定。通过设定组的基指针, 可以设定一个具体的组起始地址, 但总的地址空间是连续的 64KB。

- 每个组的起始物理地址为“基指针左移 16 位”, 每组末尾的物理地址为“尾指针左移 16 位 - 1”。

在上电或系统复位后, 所有组的地址指针寄存器都被初始化到其缺省值。这时, 所有的组指针 (ROM/SRAM/Flash 组 0 和特殊功能寄存器组除外) 都被清零。这意味着: 除 ROM/SRAM/Flash 组 0 和特殊功能寄存器组以外, 所有其它组在系统启动时都是未被定义的。这一点很重要, 用户在进行程序设计时, 一般总是要首先通过配置相应寄存器, 定义系统的存储空间。

ROM/SRAM/Flash 组 0 的尾指针和基指针的复位值分别为 0x200 和 0x0。这意味着系统复位后将自动定义 ROM/SRAM/Flash 组 0 的地址空间为 32MB, 实际地址范围为 0x0000,0000 ~ 0x0200,0000-1。ROM/SRAM/Flash 组 0 的这种初始化定义使得系统在上电或复位后, 将系统的控制权交给了由用户编写的启动代码, 当然这些启动代码应存放在外部 ROM 中的, 并映射到 ROM/SRAM/Flash 组 0。当启动代码执行时, 它执行各种系统初始化任务, 并根据应用系统的外部存储器和设备的实际情况来重新配置系统的存储器映射。

特殊功能寄存器组的基指针在系统复位时被初始化为 0x3FF0000, 一般不再改动。

图 5.2.5 是在系统启动或复位时的系统存储器映射。

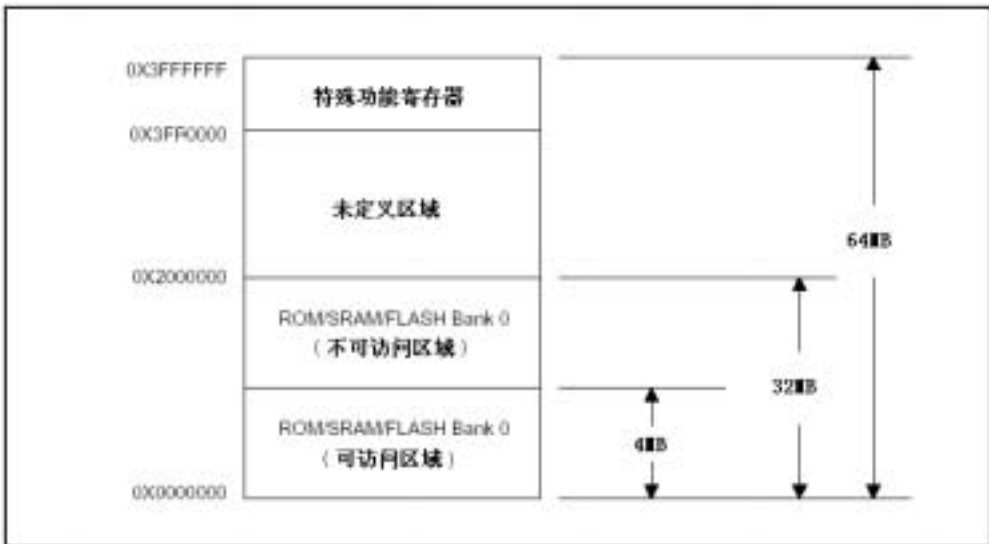


图 5.2.5 系统复位时的存储器映射

表 5-2-4 为系统管理器对应的寄存器, 系统的存储器映射可通过对这些寄存器

的配置来完成。

表 5-2-4 系统管理器对应的寄存器

| 寄存器 | 偏移量 | 操作 | 描 述 | 复位值 |
|-----------|--------|-----|--------------------------|------------|
| SYSCFG | 0x0000 | 读/写 | 系统配置寄存器 | 0x37FFF91 |
| CLKCON | 0x3000 | 读/写 | 时钟控制寄存器 | 0x00000000 |
| EXTACON0 | 0x3008 | 读/写 | 外部 I/O 时序寄存器 1 | 0x00000000 |
| EXTACON1 | 0x300C | 读/写 | 外部 I/O 时序寄存器 2 | 0x00000000 |
| EXTDBWTH | 0x3010 | 读/写 | 分组数据总线的宽度设置寄存器 | 0x00000000 |
| ROMCON0 | 0x3014 | 读/写 | ROM/ARAM/FLASH 组 0 控制寄存器 | 0x20000060 |
| ROMCON1 | 0x3018 | 读/写 | ROM/ARAM/FLASH 组 1 控制寄存器 | 0x00000060 |
| ROMCON2 | 0x301C | 读/写 | ROM/ARAM/FLASH 组 2 控制寄存器 | 0x00000060 |
| ROMCON3 | 0x3020 | 读/写 | ROM/ARAM/FLASH 组 3 控制寄存器 | 0x00000060 |
| ROMCON4 | 0x3024 | 读/写 | ROM/ARAM/FLASH 组 4 控制寄存器 | 0x00000060 |
| ROMCON5 | 0x3028 | 读/写 | ROM/ARAM/FLASH 组 5 控制寄存器 | 0x00000060 |
| DRAMCON0 | 0x302C | 读/写 | DRAM 组 0 控制寄存器 | 0x00000000 |
| DRAMCON1 | 0x3030 | 读/写 | DRAM 组 1 控制寄存器 | 0x00000000 |
| DRAMCON2 | 0x3034 | 读/写 | DRAM 组 2 控制寄存器 | 0x00000000 |
| DRAMCON3 | 0x3038 | 读/写 | DRAM 组 3 控制寄存器 | 0x00000000 |
| REFEXTCON | 0x303C | 读/写 | 刷新与外部 I/O 控制寄存器 | 0x000083FD |

根据外部存储器的宽度决定外部地址译码方法(External Address Translation Method Depends on the Width of External Memory)

与某些 ARM 芯片不同，S3C4510B 应用系统的地址总线的连接方式相对简单。由于 ARM7TDMI 采用 32 位地址总线，所有的地址都可以看作字节地址，地址总线提供 4GB 的线性寻址空间，当发出字访问信号时，存储系统忽略低 2 位 A[1:0]，当发出半字访问信号时，存储系统忽略低位 A[0]，基于以上原因，某些 ARM 系统在与存储器接口时，地址总线的连接需要错开，而 S3C4510B 则通过一个片内的地址总线生成部件，隐藏该过程，用户在设计系统时，只需将 S3C4510B 的地址总线与存储器的地址总线一一对应连接即可（即 S3C4510B 的 A[0]与外部存储器的 A[0]对齐）。表 5-2-5 和图 5.2.6 说明了该过程。

表 5-2-5 地址总线生成

| 数据总线宽度 | 外部地址引脚，ADDR[21:0] | 可访问的存储空间 |
|--------|-------------------|----------|
| 8 位 | A21-A0（内部） | 4M 字节 |
| 16 位 | A21-A1（内部） | 4M 半字 |

| | | |
|------|-------------|------|
| 32 位 | A21-A2 (内部) | 4M 字 |
|------|-------------|------|

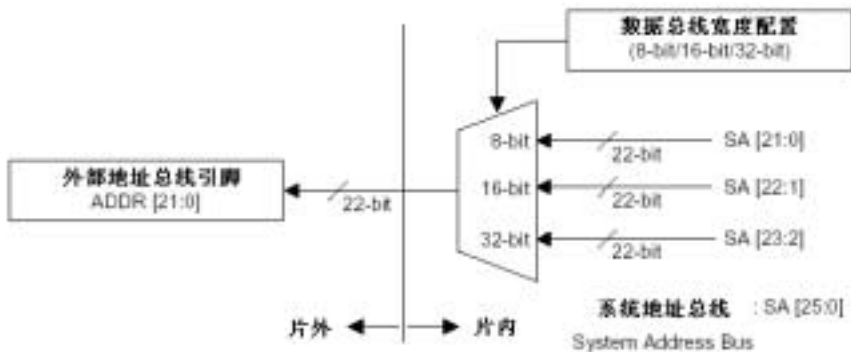


图 5.2.6 地址总线变换

总线仲裁 (Bus Arbitration)

对于 S3C4510B 微处理器，“系统总线”是指连接各个功能模块的地址和数据总线。S3C4510B 的片内功能模块和片外设备都可请求总线控制权，然后占用总线以完成数据的传输。但 S3C4510B 在系统设计时，任时刻只允许一个部件占用总线，因此，当有两个或两个以上的片内功能模块或外设同时请求总线控制权时，就要求总线控制器进行仲裁。

当某一个片内功能模块或外设取得总线控制权时，其他的总线请求被悬挂（等候处理）直到原来的总线控制设备释放总线控制权时才能被响应。

为便于总线仲裁，S3C4510B 内部的每一个功能模块都设置了优先级，总线控制器就根据这个固定的优先级对各个模块的总线请求进行仲裁。通常，总线控制权总是分配给优先级较高的功能模块。表 5-2-6 列出了优先级。

表 5-2-6 总线仲裁优先级

| 功能模块 | 总线优先级 (分组) |
|----------------------|------------------|
| 外部总线主控器 | A-1 (A 组中的最高优先级) |
| DRAM 存储器刷新控制器 | A-2 |
| General DMA1 (GDMA1) | A-3 |
| General DMA0 (GDMA0) | A-4 |
| 高层数据链路控制器 B (HDLC B) | A-5 |
| 高层数据链路控制器 A (HDLC A) | A-6 |
| MAC 带缓冲 DMA (BDMA) | A-7 (A 组中的最低优先级) |
| 写缓冲器 | B-1 (B 组中的最高优先级) |
| 总线路由器 | B-2 (B 组中的最低优先级) |

注：S3C4510B 的内部功能模块分为 A、B 两组，在每个组内，其优先级根据设置固定。

外部总线控制 (External Bus Mastership)

S3C4510B 微处理器能检测并响应由外部总线主控器产生的总线请求信号

(ExtMREQs)。当 CPU 发出外部总线应答信号 (ExtMACK) 后, 总线控制权就交给外部总线主控器, 此时外部总线请求信号应继续有效。

当 S3C4510B 的外部总线应答信号有效时, 其存储器接口处于高阻状态, 以便外部总线主控器能驱动外部存储器接口。

当 S3C4510B 不控制总线时, 它也不再进行 DRAM 的刷新操作, 因此, 当外部的总线主控器取得总线控制权且会持续一段较长的时间, 必须负责完成 DRAM 的刷新操作。

控制寄存器 (Control Register)

系统配置寄存器(System Configuration Register)

系统管理器中有一个系统配置寄存器 SYSCFG, 该寄存器决定系统管理器中特殊功能寄存器组的起始地址, 以及片内 SRAM 的使用方式和起始地址。在系统存储器映射中, 特殊功能寄存器组的地址空间固定为 64KB, 可参考图 5.2.4。

用户可以通过 SYSCFG 的设定控制诸如写缓冲、缓存模式、DRAM 模式、以及片内 SRAM 的起始地址等。

该寄存器的设置对系统的运行及性能有很大的影响, 系统设计者应仔细理解每一位所代表的意义并正确设置。

SYSCFG 寄存器

| 寄存器 | 偏移地址 | 操作 | 描述 | 复位值 |
|--------|--------|-----|---------|------------|
| SYSCFG | 0x0000 | 读/写 | 系统配置寄存器 | 0x67FFFF91 |

| | | | |
|-------|-------|------------------------------------|----------------------------|
| 31 30 | 26 25 | 16 15 | 6 5 4 3 2 1 0 |
| SDM | PD_ID | Special Register Bank Base Pointer | Internal SRAM Base Pointer |
| | | | CM 0 WE CE SE |

[0]Stall 使能 (SE)

该位必须置为 0。

[1]Cache 使能 (CE)

该位置 1, Cache 操作使能。

[2]写缓冲使能 (WE)

该位置 1, 写缓冲操作使能。

[5 : 4]Cache 模式 (CM)

该两位决定如何分配片内 8KB 存储器为 Cache 和 SRAM。

00 = 4KB SRAM, 4KB Cache

01 = 0KB SRAM, 8KB Cache

10 = 8KB SRAM, 0KB Cache

11 = 系统保留

注: 当设置为 10 时, CE 位自动清零。

[15 : 6]片内 SRAM 的基指针

该设置值左移 16 位即为片内 SRAM 的起始物理地址。

[25 : 16]特殊功能寄存器组的基指针

该设置值左移 16 位即为特殊功能寄存器组的起始物理地址。

[30 : 26]产品号 (PD_ID)

00001=S3C4510X (KS32C50100)

11001=S3C4510B

[31]SDRAM 模式

0 = 4 个 DRAM 组均设置为普通/EDO DRAM 接口

1 = 4 个 DRAM 组均设置为 SDRAM 接口

起始地址设置 (Start Address Setting)

系统管理器中特殊功能寄存器组的起始地址在系统加电或复位时被初始化为 0x3FF0000, 用户也可通过配置 SYSCFG 寄存器的值, 将特殊功能寄存器组的起始地址设定在 64MB 地址空间的任意位置。每一个特殊功能寄存器的实际物理地址为特殊功能寄存器组的起始地址加上该寄存器的偏移地址。

例如, 系统复位时特殊功能寄存器组的起始地址初始化为 0x3FF0000, 特殊功能寄存器 ROMCON 的偏移地址为 0x3014, 因此, ROMCON 的物理地址是:

$$0x3FF0000 + 0x3014 = 0x3FF3014$$

如果用户重新设定特殊功能寄存器组的起始地址为 0x3000000, 则 ROMCON 新的物理地址是 0x3003014。

在大多数情况下, 并不需要去重新设置特殊功能寄存器组的起始地址, 而直接使用其初始化值 0x3FF0000。

Cache 禁止/使能 (Cache Disable/Enable)

用户可通过设置 SYSCFG 中的 CE 位来禁止或使能 Cache, 但由于 Cache 没有自动更新的特性, 用户在使能 Cache 时必须验证数据的一致性。

片内 8KB 的 SRAM 可以通过设置 SYSCFG[5 : 4]用作 Cache, 如果不准备将 8KB 全部用作 Cache, 也可将余下的空间作为片内的 SRAM, 其起始地址由片内 SRAM 的基指针设定。

写缓冲禁止/允许 (Write Buffer Disable/Enable)

S3C4510B 内含 4 个可编程的写缓冲寄存器以提高存储器写操作的速度。当使能写缓冲器, CPU 首先将数据写入写缓冲器, 而不直接写入外部的存储器。4 个写缓冲寄存器可以增强 ARM7TDMI 内核的数据存储操作性能。

系统时钟和 MUX 总线控制寄存器 (System clock and MUX Bus Control Register)

时钟控制寄存器 (Clock Control Register, CLKCON)

在系统管理器中有一个时钟控制寄存器, 可对系统时钟分频。通过设置时钟的分频值可降低系统的工作频率, 该寄存器具体定义与设置, 请参考 S3C4510B 用户手册。

系统时钟 (System Clock)

如果将 S3C4510B 的 CLKSEL 引脚置为高电平, 由 XCLK 引脚输入的外部时钟就直接作为内部系统时钟, 若将 CLKSEL 引脚置为低电平, 外部时钟则会通过片内的 PLL 电路后才作为内部系统时钟, 在这种情况下, 内部系统时钟等于 XCLK × MF (倍乘因子)。例如, 要得到 50MHz 的内部系统时钟, 只需要 10MHz 的外部时钟。

图 5.2.7 为系统时钟电路。

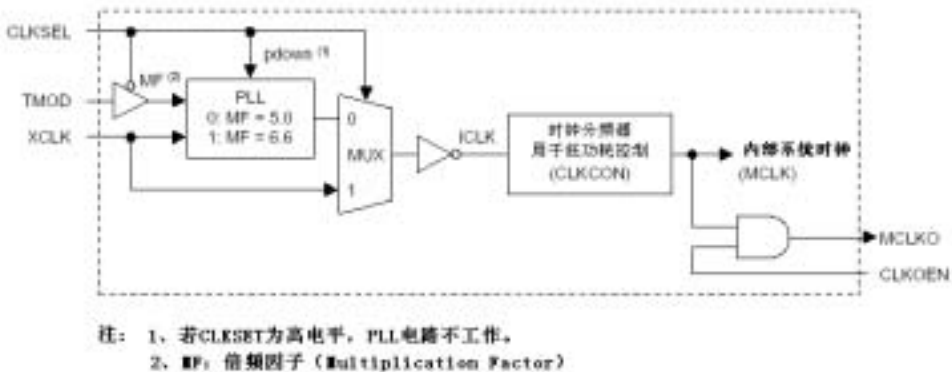


图 5.2.7 系统时钟电路

外部 I/O 访问控制寄存器 (External I/O Access Control Registers, EXTACON0/1)

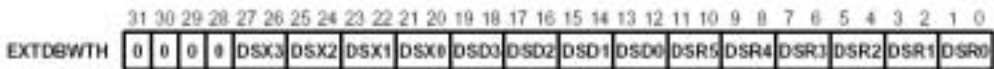
系统管理器内的外部 I/O 访问控制寄存器用于控制外部 I/O 的访问周期，该寄存器具体定义与设置，请参考 S3C4510B 用户手册。

数据总线宽度寄存器 (Data Bus Width Register, EXTDBWTH)

S3C4510B 可以以 8/16/32 位的数据宽度访问外部 ROM、SRAM、Flash 存储器，DRAM、SDRAM 以及外部 I/O 口。通过设置数据宽度寄存器，用户可设定与特定外部存储器和外部 I/O 组相对应的数据宽度。

EXTDBWTH 寄存器

| 寄存器 | 偏移地址 | 操作 | 描述 | 复位值 |
|----------|--------|-----|-------------|------------|
| EXTDBWTH | 0x3010 | 读/写 | 配置每组的数据总线宽度 | 0x00000000 |



[1:0]ROM/SRAM/FLASH 组 0 的数据总线宽度 (DSR0)

DSR0 的属性为只读，ROM/SRAM/FLASH 组 0 在系统中通常作为启动 ROM 区，其值由 B0SIZE[1:0] 引脚的状态决定。

00 = 不允许

01 = 字节 (8 位)

10 = 半字 (16 位)

11 = 字 (32 位)

[3:2]ROM/SRAM/FLASH 组 1 的数据总线宽度 (DSR1)

[5:4] DSR2, [7:6] DSR3, [9:8] DSR4, [11:10] DSR5

00 = 禁止

01 = 字节 (8 位)

10 = 半字 (16 位)

11 = 字 (32 位)

[13: 12] DRAM 组 0 的数据总线宽度 (DSD0)

[15: 14] DSD1, [17: 16] DSD2, [19: 18] DSD3

00 = 禁止

01 = 字节 (8 位)

10 = 半字 (16 位)

11 = 字 (32 位)

[21: 20] 外部 I/O 组 0 的数据总线宽度 (DSX0)

[23: 22] DSX1, [25: 24] DSX2, [27: 26] DSX3

00 = 禁止

01 = 字节 (8 位)

10 = 半字 (16 位)

11 = 字 (32 位)

当选择“禁止”时, S3C4510B 不产生相应的访问信号。

ROM/SRAM/Flash 控制寄存器 (ROM/SRAM/FLASH Control Register, ROMCON)

系统管理器内含 6 个用于控制 ROM、SRAM、Flash 存储器的寄存器, 分别对应于 S3C4510B 所支持的 6 个 ROM/SRAM/FLASH 组。

对于 ROM/SRAM/FLASH 组 0, 其外部数据总线宽度由 BOSIZE[1: 0] 引脚的状态决定:

当 BOSIZE[1: 0] = “01”, ROM/SRAM/FLASH 组 0 的外部数据总线宽度为 8 位。

当 BOSIZE[1: 0] = “10”, ROM/SRAM/FLASH 组 0 的外部数据总线宽度为 16 位。

当 BOSIZE[1: 0] = “11”, ROM/SRAM/FLASH 组 0 的外部数据总线宽度为 32 位。

ROM/SRAM/Flash 控制寄存器

| 寄存器 | 偏移地址 | 操作 | 描述 | 复位值 |
|---------|--------|-----|--------------------------|------------|
| ROMCON0 | 0x3014 | 读/写 | ROM/SRAM/FLASH 组 0 控制寄存器 | 0x20000060 |
| ROMCON1 | 0x3018 | 读/写 | ROM/SRAM/FLASH 组 1 控制寄存器 | 0x00000060 |
| ROMCON2 | 0x301C | 读/写 | ROM/SRAM/FLASH 组 2 控制寄存器 | 0x00000060 |
| ROMCON3 | 0x3020 | 读/写 | ROM/SRAM/FLASH 组 3 控制寄存器 | 0x00000060 |
| ROMCON4 | 0x3024 | 读/写 | ROM/SRAM/FLASH 组 4 控制寄存器 | 0x00000060 |
| ROMCON5 | 0x3028 | 读/写 | ROM/SRAM/FLASH 组 5 控制寄存器 | 0x00000060 |

以下是 ROM/SRAM/Flash 控制寄存器 (ROMCON0 - ROMCON5) 每位的定义:



[1: 0] 页模式配置 (PMC)

00 = 普通 ROM

01 = 4 字/页

10 = 8 字/页

11 = 16 字/页

[3: 2] 页地址访问时间 (tPA)

00 = 5 个周期 01 = 2 个周期
 10 = 3 个周期 11 = 4 个周期

[6:4] 可编程访问周期 (tACC)

000 = 该组禁用 001 = 2 个周期
 010 = 3 个周期 011 = 4 个周期
 110 = 7 个周期 111 = 保留

[19:10] ROM/SRAM/FI ash 组基指针

该设置值左移 16 位即为 ROM/SRAM/FI ash 组的起始物理地址。

[29:20] ROM/SRAM/FI ash 组尾指针

该设置值左移 16 位-1 即为 ROM/SRAM/FI ash 组的结束物理地址。

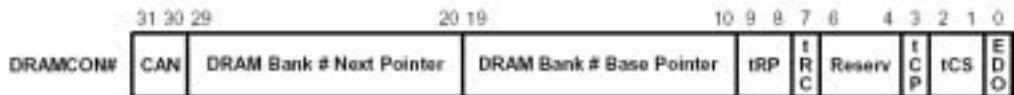
DRAM 控制寄存器 (DRAM Control Registers)

系统管理器内含 4 个 DRAM 控制寄存器, DRAMCON0 ~ DRAMCON3, 分别对应于 S3C4510B 所支持的 4 个 DRAM 组。REFEXTCON 寄存器用于设置外部 I/O 组 0 的基指针。

S3C4510B 支持 EDO、普通、同步 DRAM (SDRAM)。通过将 SYSCFG[31] 置 1 可选择 SDRAM 模式, 如果将该位置 1, 所有的 DRAM 组均选择 SDRAM, 置 0 则所有 DRAM 组均选择 EDO/FP DRAM。

DRAM 及外部 I/O 控制寄存器

| 寄存器 | 偏移地址 | 操作 | 描 述 | 复位值 |
|-----------|--------|-----|-----------------|------------|
| DRAMCON0 | 0x302C | 读/写 | DRAM 组 0 控制寄存器 | 0x00000000 |
| DRAMCON1 | 0x3030 | 读/写 | DRAM 组 1 控制寄存器 | 0x00000000 |
| DRAMCON2 | 0x3034 | 读/写 | DRAM 组 2 控制寄存器 | 0x00000000 |
| DRAMCON3 | 0x3038 | 读/写 | DRAM 组 3 控制寄存器 | 0x00000000 |
| REFEXTCON | 0x303C | 读/写 | 刷新与外部 I/O 控制寄存器 | 0x00000000 |



[0] EDO 模式 (EDO)

0 = 普通 DRAM (快速页模式 DRAM)

1 = EDO DRAM

[2:1] CAS 锁存时间 (tCS)

00 = 1 个周期 01 = 2 个周期

10 = 3 个周期 11 = 4 个周期

[3] CAS 预充电时间 (tCP)

0 = 1 个周期 1 = 2 个周期

[6:4] 系统保留

系统默认值为 000, 但用户在使用时应置为 001

[7] RAS 到 CAS 的延迟 (tRC 或 tRCD)

0 = 1 个周期 1 = 2 个周期

[9: 8]RAS 预充电时间 (tRP)

00 = 1 个周期 01 = 2 个周期

10 = 3 个周期 11 = 4 个周期

[19: 10] DRAM 组基指针

该设置值左移 16 位即为 DRAM 组的起始物理地址。

[29: 20] DRAM 组尾指针

该设置值左移 16 位 - 1 即为 DRAM 组的结束物理地址。

[31: 30] DRAM 组的列地址位数 (CAN)

00 = 8 位 01 = 9 位

10 = 10 位 11 = 11 位

DRAM 接口特性(DRAM Interface Features)

S3C4510B 具有完全可编程的外部 DRAM 接口 ,通过设置相应的 DRAM 控制寄存器 ,用户可方便的控制各种接口特性。这些可编程的特性包括 :

- 外部数据总线宽度。
- 通过 DRAMCON[0]选择快速页模式或 EDO 模式。
- 通过 SYSCFG[31]选择快速页模式/EDO 模式或 SDRAM 模式。
- 每个 DRAM 组的访问周期数、CAS 锁存时间、CAS 预充电时间、RAS 到 CAS 时延、RAS 预充电时间等。

刷新及外部 I/O 控制寄存器 (REFEXTCON), 用于控制 DRAM 的刷新操作及对外部 I/O 组的访问。通过自动产生刷新控制信号 ,S3C4510B 系统不再需要其他的外部刷新信号刷新 DRAM。

通过 23 位内部地址总线 , S3C4510B 可产生用于 DRAM 访问的行、列地址信号。同时 , S3C4510 支持同步或异步 DRAM。

访问 SDRAM (Synchronous DRAM Accesses)

SDRAM 的接口控制信号包括 CKE、SDCLK、nSDCS[3: 0]、nSDCAS、nSDRAS 及 DQM[3: 0] 等。SDRAM 的接口方法将在硬件设计部分详细描述。

DRAM 组的地址空间(DRAM Bank Space)

S3C4510B 的 DRAM 接口支持 4 个 DRAM 组 ,各 DRAM 组可以是不同的配置。用户可通过 DRAM 控制寄存器 (DRAMCON0 ~ DRAMCON3) 设置 DRAM 访问周期和存储器组在地址空间中的位置。

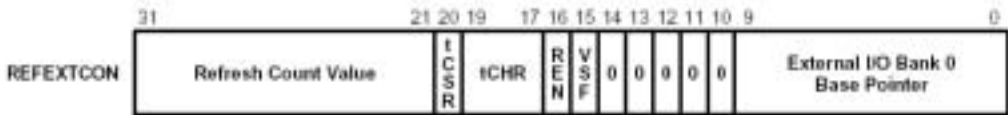
每一个 DRAM 控制寄存器都包含两个 10 位的地址指针 ,首指针和尾指针 ,用以指示 DRAM 组的起始地址和结束地址。10 位的指针值对应 A[25: 16] ,因此 ,在每一个组内 ,其地址空间为 $2^{16} = 64KB$ 。

系统初始化值(System Initialization Values)

当系统初始化时 , 4 个 DRAM 控制寄存器均被置为 0x00000000 ,同时禁用所有的外部 DRAM。

DRAM 刷新与外部 I/O 控制寄存器 (DRAM Refresh and External I/O Control Register , REFEXTCON)

S3C4510B 的 DRAM 接口支持对 ED0/FP DRAM 的 CBR (CAS-before-RAS) 刷新模式和 SDRAM 的自动刷新。通过设置 DRAM 刷新与外部 I/O 控制寄存器 REFEXTCON，可控制 DRAM 的刷新模式、刷新时序及刷新间隔。REFEXTCON 同时还包含 10 位外部 I/O 组 0 的基指针值。



[9 : 0]外部 I/O 组 0 基指针 (基地址)

该设置值左移 16 位即为外部 I/O 组 0 的起始物理地址。外部 I/O 组 0 的结束物理地址为该设置值左移 16 位+16KB-1。

4 个外部 I/O 组的地址空间连续，每组均固定大小为 16KB，因此，通过该基指针可以计算出其他 3 个外部 I/O 组的起始和结束地址。

[15]特殊功能寄存器域有效性 (VSF)

0 = 存储器组不可访问

1 = 存储器组可访问

[16]刷新使能 (REN)

0 = 禁止 DRAM 刷新

1 = 使能 DRAM 刷新

[19: 17]CAS 持续时间 (tCHR)

000 = 1 个周期

001 = 2 个周期

010 = 3 个周期

011 = 4 个周期

100 = 5 个周期

101 = 未用

110 = 未用

111 = 未用

[20]CAS 设置时间 (tCSR)

0 = 1 个周期

1 = 2 个周期

[31 : 21]刷新计数值 (持续时间)

刷新周期 = $(2^{11} - \text{该值} + 1) / \text{FMCK}$

以上是对 S3C4510B 的系统管理器及相关特殊功能寄存器的简单介绍，这些内容将会在程序设计，特别是系统初始化部分的程序设计中反复用到。限于篇幅，还有部分系统管理器的特殊功能寄存器未做介绍，请读者自行参考 S3C4510B 用户手册。

5.3 系统的硬件选型与单元电路设计

从这一节开始，将详细描述系统的硬件选型与设计，希望通过对这些章节的阅读，能使读者具有初步设计特定系统的能力。

尽管硬件选型与单元电路设计部分的内容是基于 S3C4510B 的系统，但由于 ARM

体系结构的一致性和常见外围电路的通用性，只要读者能真正理解本部分的设计方法，从而设计出基于其他 ARM 微处理器的系统，应该也是比较容易的。

需要说明，以下的应用电路可能不是最佳的，但经验证是可以正常工作的。

5.3.1 S3C4510B 芯片及引脚分析

S3C4510B 共有 208 只引脚，采用 QFP 封装，这对于那些常使用 8 位/16 位 DIP 封装微控制器的读者来说，可能会觉得有点复杂，然而，尽管 S3C4510B 引脚较多，但根据各自的功能，分布很有规律。

首先，电源和接地引脚有近 50 根，再除去地址总线、数据总线和通用 I/O 口，以及其他的专用模块如 HDLC、UART、IIC、MAC 等的接口，真正需要仔细研究的引脚数就不是很多了，但这些引脚主要是控制信号，需要认真对待，在此先进行简单的分析，其后的单元电路设计里，会有更再详细的说明。

在硬件系统的设计中，应当注意芯片引脚的类型，S3C4510B（也包括其他的微处理器）的引脚主要分为三类，即：输入（I）、输出（O）、输入/输出（I/O）。

输出类型的引脚主要用于 S3C4510B 对外设的控制或通信，由 S3C4510B 主动发出，这些引脚的连接不会对 S3C4510B 自身的运行有太大的影响。

输入/输出类型的引脚主要是 S3C4510B 与外设的双向数据传输通道。

而某些输入类型的引脚，其电平信号的设置是 S3C4510B 本身正常工作的前提，在系统设计时必须小心处理。

S3C4510B 的主要控制信号如下：

LITTLE (Pin49)：大、小端模式选择引脚。高电平 = 小端模式；低电平 = 大端模式；该引脚在片内下拉，系统默认为大端模式，但在实际系统中一般使用小端模式，更符合我们的使用习惯，因此该引脚可上拉或接电源。

FILTER (Pin55)：如果使用 PLL 倍频电路，应在该引脚和地之间接 820pF 的陶瓷电容。在实际系统中，一般应使用 PLL 电路，因此，该电容应连接。

TCK、TMS、TDI、TDO、nTRST (Pin58 ~ Pin62)：JTAG 接口引脚。根据 IEEE 标准，TCK 应下拉，TMS、TDI 和 nTRST 应上拉。S3C4510B 已按此标准在片内连接，只需要与 JTAG 插座直接相连即可，但某些 ARM 芯片并未做相应的处理，在设计电路时应注意。

TMODE (Pin63)：测试模式。高电平 = 芯片测试模式；低电平 = 正常工作模式；用户一般不作芯片测试，该引脚下拉或接地，使芯片处于正常工作模式。

nWAIT (Pin71)：外部等待请求信号。该引脚应上拉。

BOSIZE[1:0] (Pin74, Pin73)：BANK0 数据宽度选择。'01' = 8 位；'10' = 16 位；'11' = 32 位；'00' = 系统保留。

CLKOEN (Pin76)：时钟输出允许/禁止。高电平 = 允许；低电平 = 禁止。一些外围器件（如 SDRAM）需要 CPU 的时钟输出作为自身的时钟源，该引脚一般接高电平，使时钟输出为允许状态。

XCLK (Pin80)：系统时钟源。接有源晶振的输出。

nRESET (Pin82) : 系统复位引脚。低电平复位, 当系统正常工作时, 该引脚应处于高电平状态。

CLKSEL (Pin83) : 时钟选择。高电平 = XCLK 直接作为系统的工作时钟; 低电平 = XCLK 经过 PLL 电路倍频后作为系统的工作时钟。

ExtMREQ (Pin108) : 外部主机总线请求信号。该引脚应下拉。

S3C4510B 的其余引脚为电源线、接地线、数据总线、地址总线以及其他功能模块地输入/输出线, 对 CPU 自身地运行地影响相对较小, 其连接方式也比较简单, 在此不作详述。

5.3.2 电源电路

在该系统中, 需要使用 5V 和 3.3V 的直流稳压电源, 其中, S3C4510B 及部分外围器件需 3.3V 电源, 另外部分器件需 5V 电源, 为简化系统电源电路的设计, 要求整个系统的输入电压为高质量的 5V 的直流稳压电源。系统电源电路如下图所示:

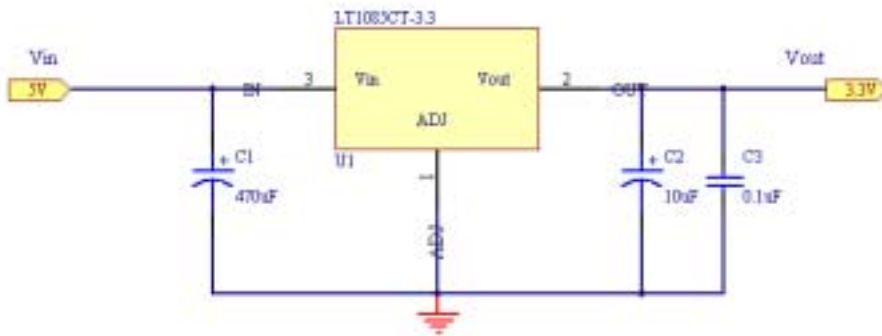


图 5.3.1 系统的电源电路

有很多 DC-DC 转换器可完成 5V 到 3.3V 的转换, 在此选用 Linear Technology 的 LT108X 系列。常见的型号和对应的电流输出如下:

| | |
|--------|------|
| LT1083 | 7.5A |
| LT1084 | 5A |
| LT1085 | 3A |
| LT1086 | 1.5A |

设计者可根据系统的实际功耗, 选择不同的器件。

5.3.3 晶振电路与复位电路

晶振电路用于向 CPU 及其他电路提供工作时钟。在该系统中，S3C4510B 使用有源晶振。不同于常用的无源晶振，有源晶振的接法略有不同。常用的有源晶振的接法如下图所示：

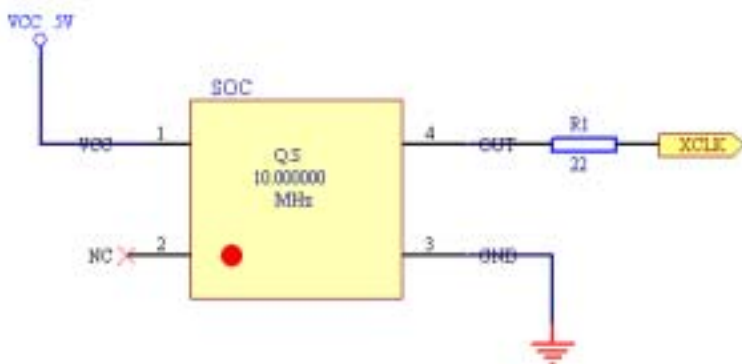


图 5.3.2 系统的晶振电路

根据 S3C4510B 的最高工作频率以及 PLL 电路的工作方式，选择 10MHz 的有源晶振，10MHz 的晶振频率经过 S3C4510B 片内的 PLL 电路倍频后，最高可以达到 50MHz。片内的 PLL 电路兼有频率放大和信号提纯的功能，因此，系统可以以较低的外部时钟信号获得较高的工作频率，以降低因高速开关时钟所造成的高频噪声。

有源晶振的 1 脚接 5V 电源，2 脚悬空，3 脚接地，4 脚为晶振的输出，可通过一个小电阻（此处为 22 欧姆）接 S3C4510B 的 XCLK 引脚。

在系统中，复位电路主要完成系统的上电复位和系统在运行时用户的按键复位功能。复位电路可由简单的 RC 电路构成，也可使用其他的相对较复杂，但功能更完善的电路。

本系统采用较简单的 RC 复位电路，经使用证明，其复位逻辑是可靠的。复位电路如图 5.3.3 所示：

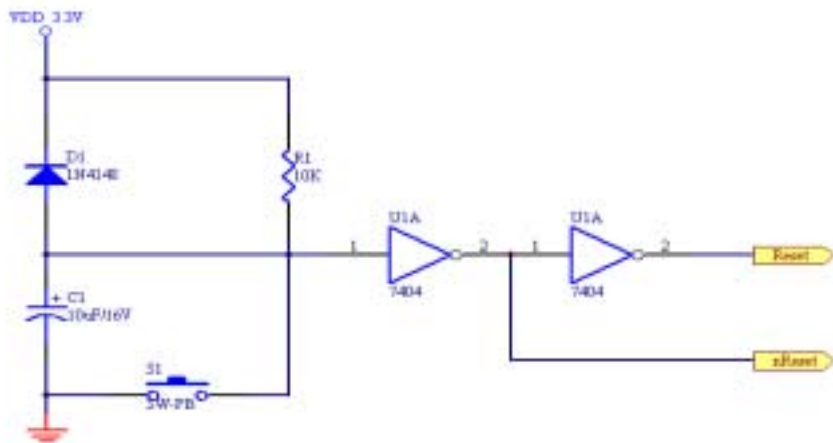


图 5.3.3 系统的复位电路

该复位电路的工作原理如下：在系统上电时，通过电阻 R1 向电容 C1 充电，当 C1 两端的电压未达到高电平的门限电压时，Reset 端输出为低电平，系统处于复位状态；当 C1 两端的电压达到高电平的门限电压时，Reset 端输出为高电平，系统进入正常工作状态。

当用户按下按钮 S1 时，C1 两端的电荷被泻放掉，Reset 端输出为低电平，系统进入复位状态，再重复以上的充电过程，系统进入正常工作状态。

两级非门电路用于按钮去抖动和波形整形；nReset 端的输出状态与 Reset 端相反，以用于高电平复位的器件；通过调整 R1 和 C1 的参数，可调整复位状态的时间。

5.3.4 Flash 存储器接口电路

Flash 存储器是一种可在系统 (In-System) 进行电擦写，掉电后信息不丢失的存储器。它具有低功耗、大容量、擦写速度快、可整片或分扇区在系统编程 (烧写)、擦除等特点，并且可由内部嵌入的算法完成对芯片的操作，因而在各种嵌入式系统中得到了广泛的应用。作为一种非易失性存储器，Flash 在系统中通常用于存放程序代码、常量表以及一些在系统掉电后需要保存的用户数据等。常用的 Flash 为 8 位或 16 位的数据宽度，编程电压为单 3.3V。主要的生产厂商为 ATMEL、AMD、HYUNDAI 等，他们生产的同型器件一般具有相同的电气特性和封装形式，可通用。

以该系统中使用的 Flash 存储器 HY29LV160 为例，简要描述一下 Flash 存储器的基本特性：

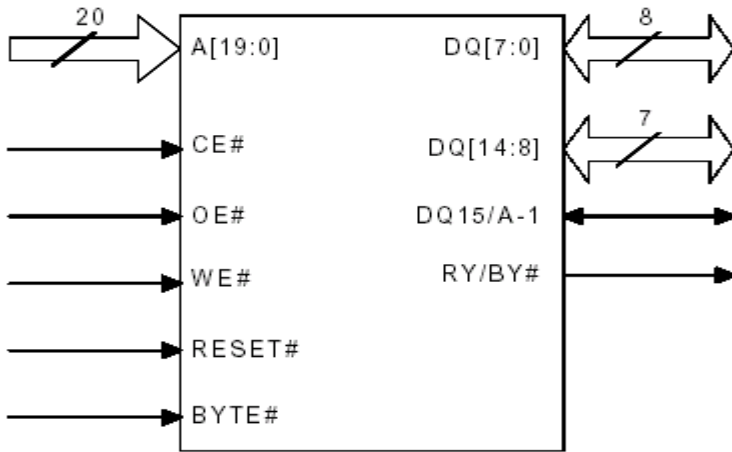


图 5.3.4 HY29LV160 的逻辑框图

HY29LV160 的单片存储容量为 16M 位 (2M 字节), 工作电压为 2.7V ~ 3.6V, 采用 48 脚 TSOP 封装或 48 脚 FBGA 封装, 16 位数据宽度, 可以以 8 位 (字节模式) 或 16 位 (字模式) 数据宽度的方式工作。

HY29LV160 仅需单 3V 电压即可完成在系统的编程与擦除操作, 通过对其内部的命令寄存器写入标准的命令序列, 可对 Flash 进行编程 (烧写)、整片擦除、按扇区擦除以及其他操作。

HY29LV160 的逻辑框图、引脚分布及信号描述分别如图 5.3.4、图 5.3.5 和表 5-3-1 所示:

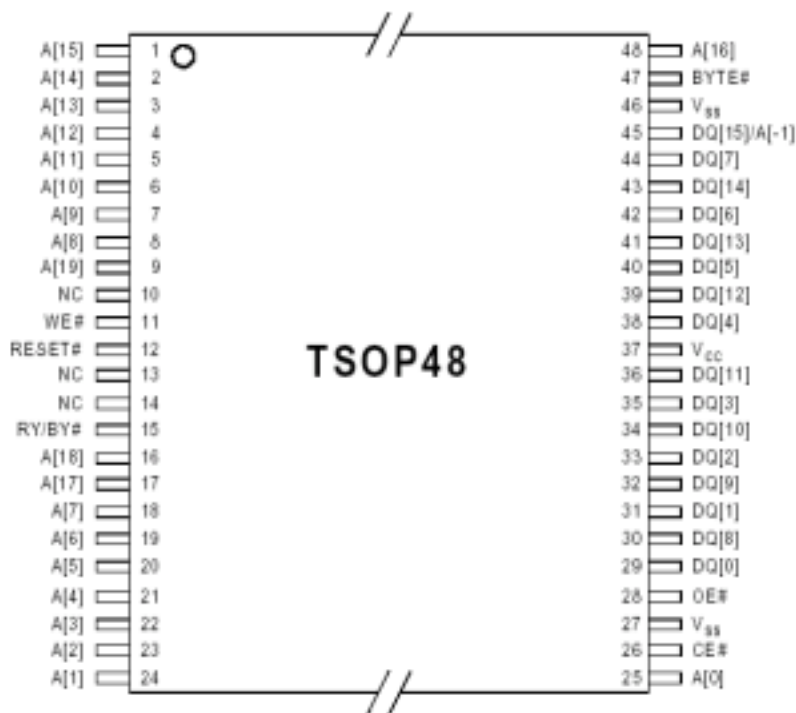


图 5.3.5 HY29LV160 引脚分布 (TSOP48 封装)

表 5-3-1 HY29LV160 的引脚信号描述

| 引脚 | 类型 | 描述 |
|--------------------------|-----------|---|
| A[19:0] | I | 地址总线。在字节模式下, DQ[15]/A[-1]用作 21 位字节地址的最低位。 |
| DQ[15]/A[-1] DQ[14:0] | I/O 三态 | 数据总线。在读写操作时提供 8 位或 16 位的数据宽度。在字节模式下, DQ[15]/A[-1]用作 21 位字节地址的最低位, 而 DQ[14:8]处于高阻状态。 |
| BYTE# | I | 模式选择。低电平选择字节模式, 高电平选择字模式 |
| CE# | I | 片选信号, 低电平有效。在对 HY29LV160 进行读写操作时, 该引脚必须为低电平, 当为高电平时, 芯片处于高阻旁路状态 |
| OE# | I | 输出使能, 低电平有效。在读操作时有效, 写操作时无效。 |
| WE# | I | 写使能, 低电平有效。在对 HY29LV160 进行编程和擦除操作时, 控制相应的写命令。 |
| RESET# | I | 硬件复位, 低电平有效。对 HY29LV160 进行硬件复位。当复位时, HY29LV160 立即终止正在进行的操作。 |
| RY/BY# | O | 就绪/忙 状态指示。用于指示写或擦除操作是否完成。当 HY29LV160 正在进行编程或擦除操作时, 该引脚位低电平, 操作完成时为高电平, 此时可读取内部的数据。 |
| VCC | -- | 3.3V 电源 |
| VSS | -- | 接地 |

以上为一款常见的 Flash 存储器 HY29LV160 的简介, 更具体的内容可参考

HY29LV160 的用户手册。其他类型的 Flash 存储器的特性与使用方法与之类似，用户可根据自己的实际需要选择不同的器件。

下面，我们使用 HY29LV160 来构建 Flash 存储系统。由于 ARM 微处理器的体系结构支持 8 位/16 位/32 位的存储器系统，对应的可以构建 8 位的 Flash 存储器系统、16 位的 Flash 存储器系统或 32 位的 Flash 存储器系统。32 位的存储器系统具有较高的性能，而 16 位的存储器系统则在成本及功耗方面占有优势，而 8 位的存储器系统现在已经很少使用。在此，分别介绍 16 位和 32 位的 Flash 存储器系统的构建。

16 位的 FLASH 存储器系统：

图 5.3.6 为 16 位 Flash 存储器系统的实际应用电路图。

在大多数的系统中，选用一片 16 位的 Flash 存储器芯片（常见单片容量有 1MB、2MB、4MB、8MB 等）构建 16 位的 Flash 存储系统已经足够，在此采用一片 HY29LV160 构建 16 位的 Flash 存储器系统，其存储容量为 2MB。Flash 存储器在系统中通常用于存放程序代码，系统上电或复位后从此获取指令并开始执行，因此，应将存有程序代码的 Flash 存储器配置到 ROM/SRAM/FLASH Bank0，即将 S3C4510B 的 nRCS<0>（Pin75）接至 HY29LV160 的 CE# 端。

HY29LV160 的 RESET# 端接系统复位信号；

OE# 端接 S3C4510B 的 nOE（Pin72）；

WE# 端 S3C4510B 的 nWBE<0>（Pin100）；

BYTE# 上拉，使 HY29LV160 工作在字模式（16 位数据宽度）；

RY/BY# 指示 HY29LV160 编程或擦除操作的工作状态，但其工作状态也可通过查询片内的相关寄存器来判断，因此可将该引脚悬空；

地址总线[A19~A0]与 S3C4510B 的地址总线[ADDR19~ADDR0]相连；

16 位数据总线 [DQ15~DQ0] 与 S3C4510B 的低 16 位数据总线 [XDATA15~XDATA0]相连。

注意此时应将 S3C4510B 的 B0SIZE[1:0]置为‘10’，选择 ROM/SRAM/FLASH Bank0 为 16 位工作方式。

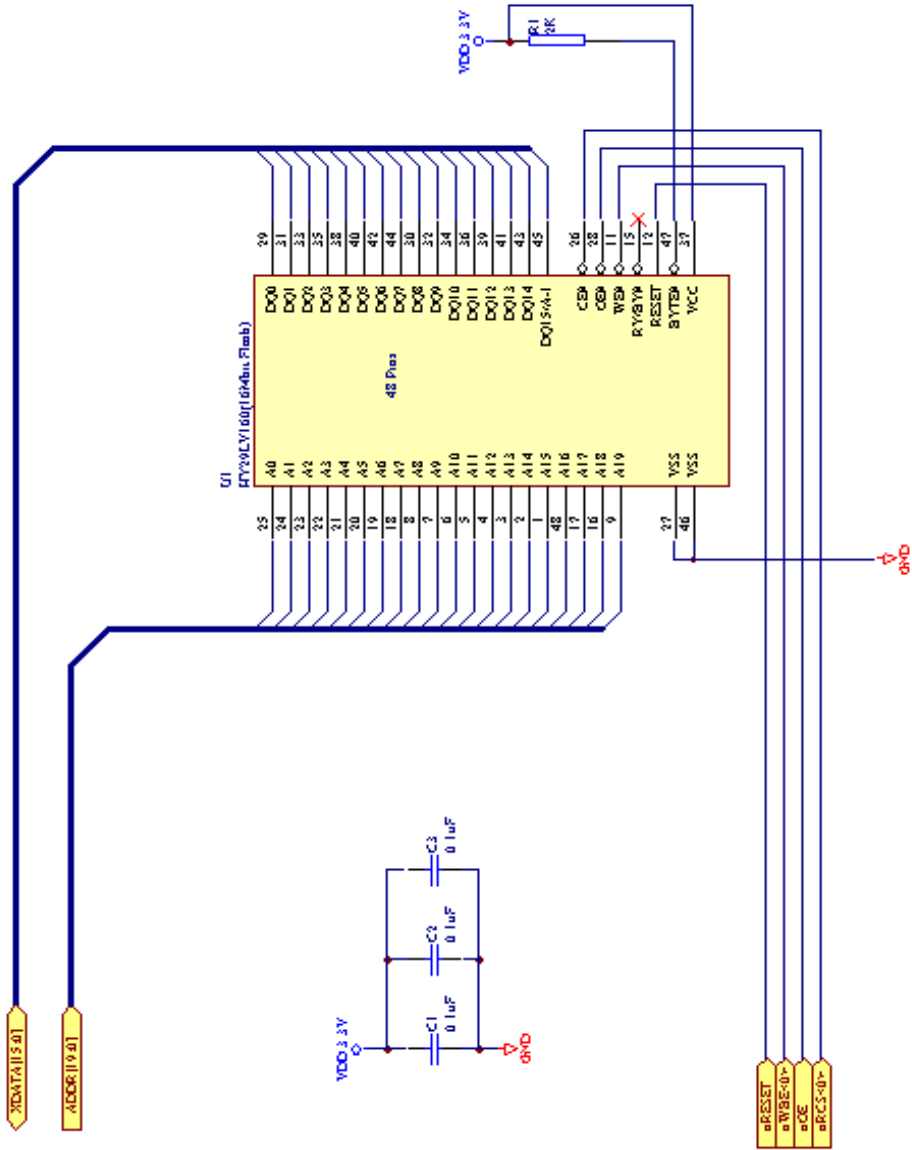


图 5.3.6 16 位 FLASH 存储系统电路图

32 位的 FLASH 存储器系统：

图 5.3.7 为 32 位 Flash 存储器系统的实际应用电路图。

作为一款 32 位的微处理器，为充分发挥 S3C4510B 的 32 性能优势，有的系统也采用两片 16 位数据宽度的 Flash 存储器芯片并联（或一片 32 位数据宽度的 Flash 存储器芯片）构建 32 位的 Flash 存储系统。其构建方式与 16 位的 Flash 存储器系统

相似。

采用两片 HY29LV16 并联的方式构建 32 位的 FLASH 存储器系统，其中一片为高 16 位，另一片为低 16 位，将两片 HY29LV16 作为一个整体配置到 ROM/SRAM/FLASH Bank0，即将 S3C4510B 的 nRCS<0>（Pin75）接至两片 HY29LV16 的 CE# 端；

两片 HY29LV160 的 RESET# 端接系统复位信号；

两片 HY29LV160 的 OE# 端接 S3C4510B 的 nOE（Pin72）；

低 16 位片的 WE# 端接 S3C4510B 的 nWBE<0>（Pin100），高 16 位片的 WE# 端接 S3C4510B 的 nWBE<2>（Pin102）；

两片 HY29LV160 的 BYTE# 均上拉，使之均工作在字模式；

两片 HY29LV160 的地址总线[A19~A0]均与 S3C4510B 的地址总线[ADDR19~ADDR0]相连；

低 16 位片的数据总线与 S3C4510B 的低 16 位数据总线[XDATA15~XDATA0]相连，高 16 位片的数据总线与 S3C4510B 的高 16 位数据总线[XDATA31~XDATA16]相连。

注意此时应将 S3C4510B 的 B0SIZE[1:0]置为‘11’，选择 ROM/SRAM/FLASH Bank0 为 32 位工作方式。

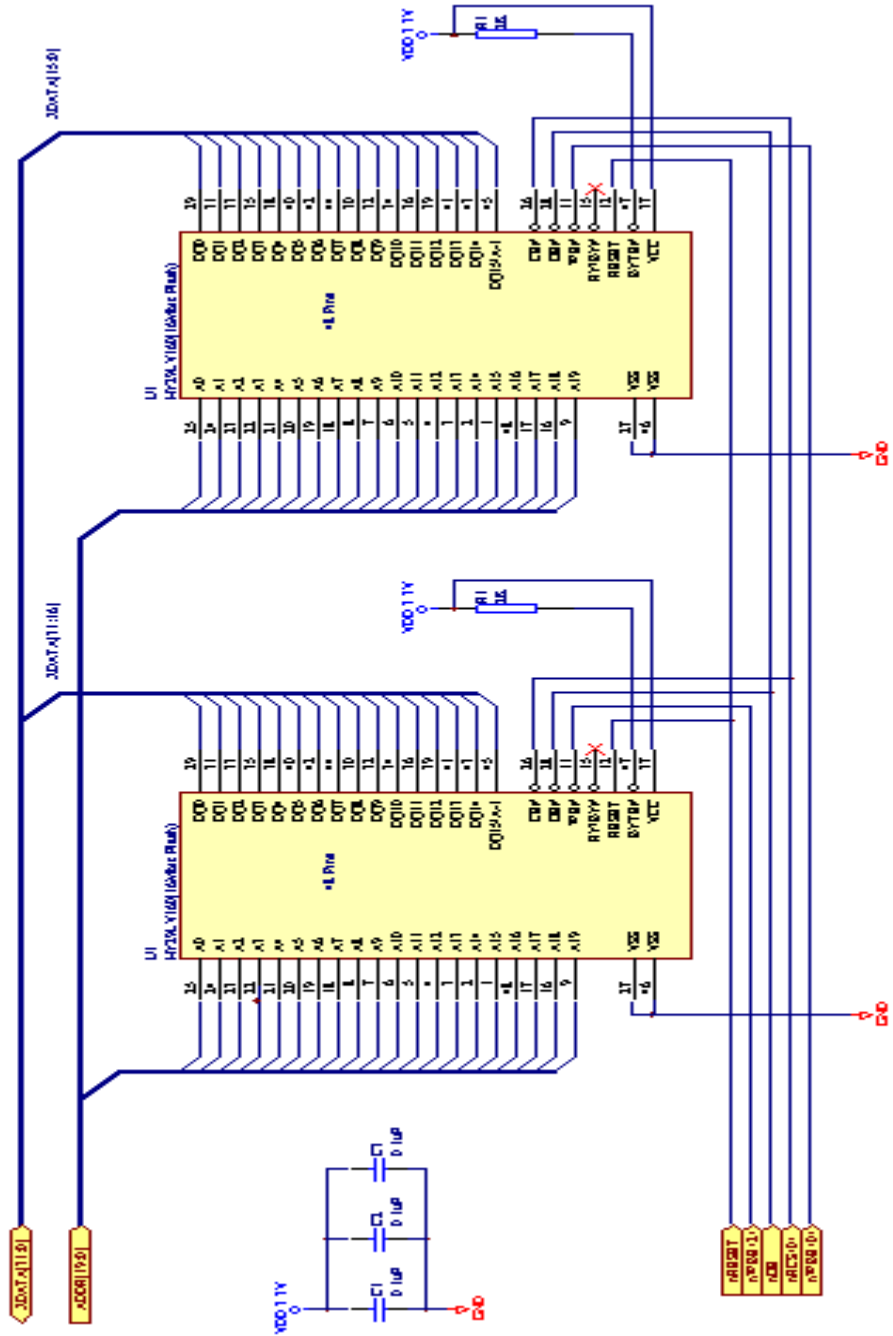


图 5.3.7 32 位 FLASH 存储系统电路图

5.3.5 SDRAM 接口电路

与 Flash 存储器相比较，SDRAM 不具有掉电保持数据的特性，但其存取速度大大高于 Flash 存储器，且具有读/写的属性，因此，SDRAM 在系统中主要用作程

序的运行空间，数据及堆栈区。当系统启动时，CPU 首先从复位地址 0x0 处读取启动代码，在完成系统的初始化后，程序代码一般应调入 SDRAM 中运行，以提高系统的运行速度，同时，系统及用户堆栈、运行数据也都放在 SDRAM 中。

SDRAM 具有单位空间存储容量大和价格便宜的优点，已广泛应用在各种嵌入式系统中。SDRAM 的存储单元可以理解为一个电容，总是倾向于放电，为避免数据丢失，必须定时刷新（充电）。因此，要在系统中使用 SDRAM，就要求微处理器具有刷新控制逻辑，或在系统中另外加入刷新控制逻辑电路。S3C4510B 及其他一些 ARM 芯片在片内具有独立的 SDRAM 刷新控制逻辑，可方便的与 SDRAM 接口。但某些 ARM 芯片则没有 SDRAM 刷新控制逻辑，就不能直接与 SDRAM 接口，在进行系统设计时应注意这一点。

目前常用的 SDRAM 为 8 位/16 位的数据宽度，工作电压一般为 3.3V。主要的生产厂商为 HYUNDAI、Winbond 等。他们生产的同型器件一般具有相同的电气特性和封装形式，可通用。

以该系统中使用的 HY57V641620 为例，简要描述一下 SDRAM 的基本特性及使用方法：

HY57V641620 存储容量为 4 组 × 16M 位（8M 字节），工作电压为 3.3V，常见封装为 54 脚 TSOP，兼容 LVTTTL 接口，支持自动刷新（Auto-Refresh）和自刷新（Self-Refresh），16 位数据宽度。

HY57V641620 引脚分布及信号描述分别如图 5.3.8 和表 5-3-2 所示：

表 5-3-2 HY57V641620 引脚信号描述

| 引 脚 | 名 称 | 描 述 |
|-------------------------|-----------------------|--------------------------------------|
| CLK | 时钟 | 芯片时钟输入。 |
| CKE | 时钟使能 | 片内时钟信号控制。 |
| /CS | 片选 | 禁止或使能除 CLK、CKE 和 DQM 外的所有输入信号。 |
| BA0, BA1 | 组地址选择 | 用于片内 4 个组的选择。 |
| A11 ~ A0 | 地址总线 | 行地址：A11 ~ A0，列地址：A7 ~ A0，自动预充电标志：A10 |
| /RAS , /CAS , /WE | 行地址锁存 列地址锁存 写使能 | 参照功能真值表，/RAS、/CAS 和 /WE 定义相应的操作。 |
| LDQM, UDQM | 数据 I/O 屏蔽 | 在读模式下控制输出缓冲；在写模式下屏蔽输入数据 |
| DQ15 ~ DQ0 | 数据总线 | 数据输入输出引脚 |
| VDD/VSS | 电源/地 | 内部电路及输入缓冲电源/地 |
| VDDQ/VSSQ | 电源/地 | 输出缓冲电源/地 |
| NC | 未连接 | 未连接 |

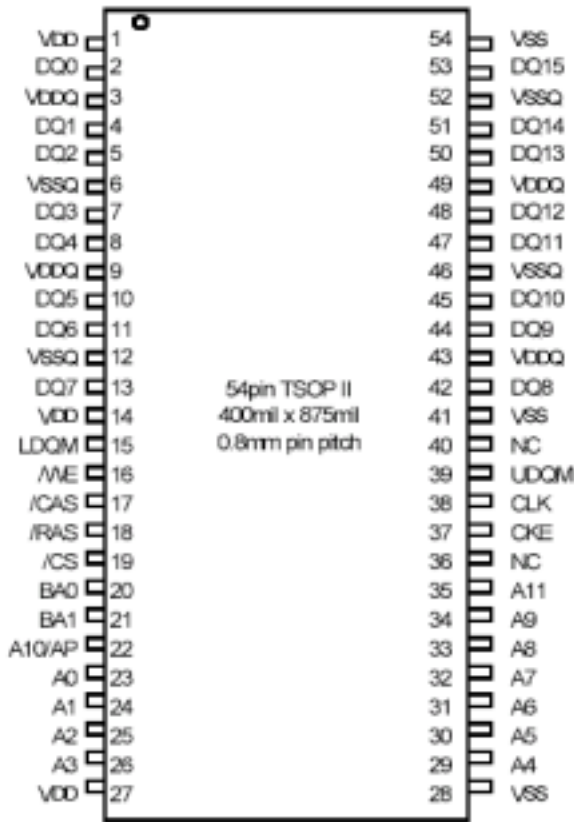


图 5.3.8 HY57V641620 引脚分布

以上为一款常见的 SDRAM HY57V641620 的简介，更具体的内容可参考 HY57V641620 的用户手册。其他类型 SDRAM 的特性与使用方法与之类似，用户可根据自己的实际需要选择不同的器件。

根据系统需求，可构建 16 位或 32 位的 SDRAM 存储器系统，但为充分发挥 32 位 CPU 的数据处理能力，大多数系统采用 32 位的 SDRAM 存储器系统。

HY57V641620 为 16 位数据宽度，单片容量为 8MB，系统选用的两片 HY57V641620 并联构建 32 位的 SDRAM 存储器系统，共 16MB 的 SDRAM 空间，可满足嵌入式操作系统及各种相对较复杂的算法的运行要求。

图 5.3.9 为 32 位 SDRAM 存储器系统的实际应用电路图。

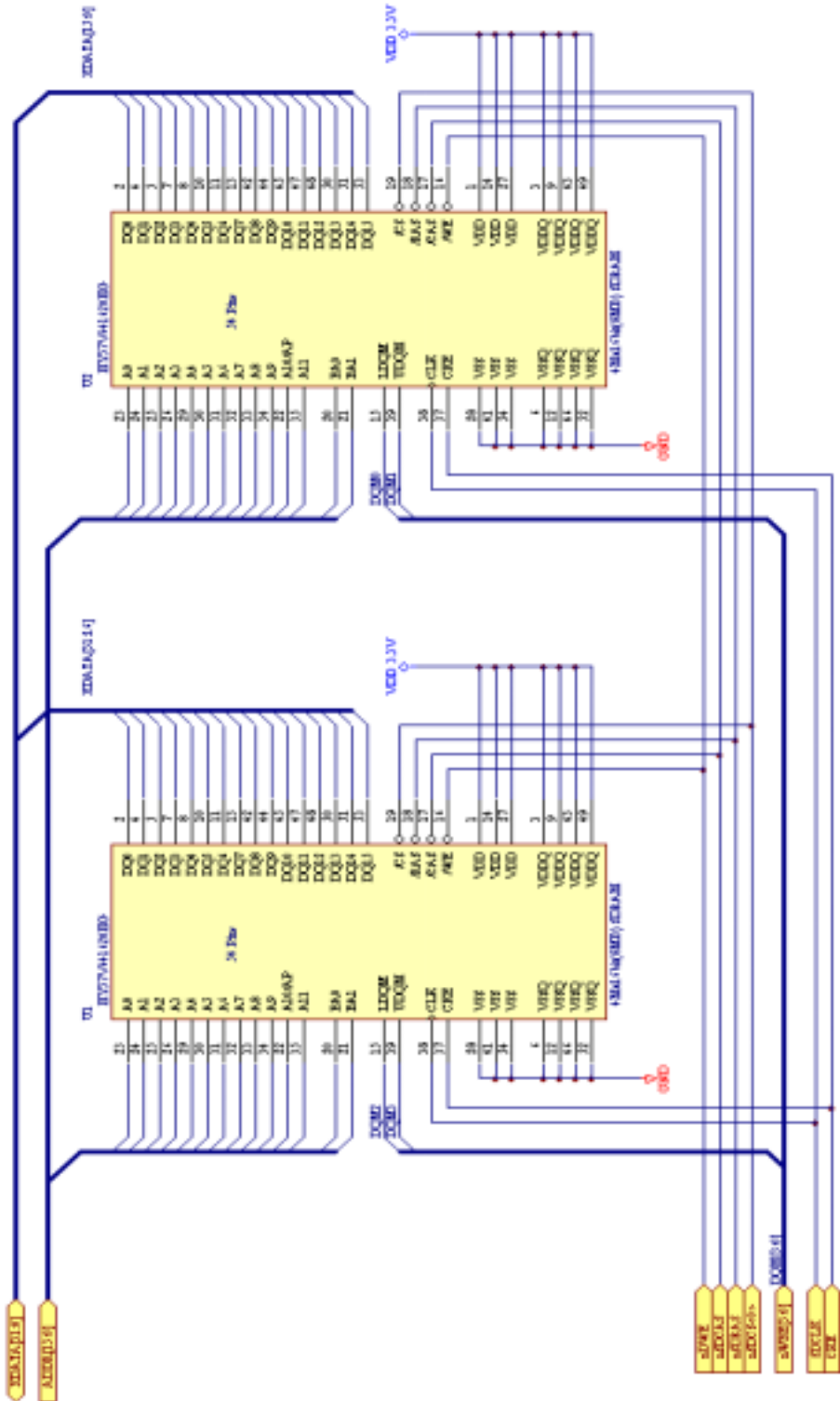


图 5.3.9 32 位 SDRAM 存储器系统的实际应用电路图

与 Flash 存储器相比,SDRAM 的控制信号较多,其连接电路也要相对复杂。

两片 HY57V641620 并联构建 32 位的 SDRAM 存储器系统,其中一片为高 16 位,另一片为低 16 位,可将两片 HY57V641620 作为一个整体配置到 DRAM/SDRAM Bank0 ~ DRAM/SDRAM Bank3 的任一位置,一般配置到 DRAM/SDRAM Bank0,即将 S3C4510B 的 nSDCS<0> (Pin89) 接至两片 HY57V641620 的/CS 端。

两片 HY57V641620 的 CLK 端接 S3C4510B 的 SDCLK 端 (Pin77) ;

两片 HY57V641620 的 CLE 端接 S3C4510B 的 CLE 端 (Pin97) ;

两片 HY57V641620 的/RAS、/CAS、/WE 端分别接 S3C4510B 的 nSDRAS 端 (Pin95)、nSDCAS 端 (Pin96)、nDWE 端 (Pin99) ;

两片 HY57V641620 的 A11 ~ A0 接 S3C4510B 的地址总线 ADDR<11> ~ ADDR<0> ;

两片 HY57V641620 的 BA1、BA0 接 S3C4510B 的地址总线 ADDR<13>、ADDR<12> ;

高 16 位片的 DQ15 ~ DQ0 接 S3C4510B 的数据总线的高 16 位 XDATA<31> ~ XDATA<16>,低 16 位片的 DQ15 ~ DQ0 接 S3C4510B 的数据总线的低 16 位 XDATA<15> ~ XDATA<0> ;

高 16 位片的 UDQM、LDQM 分别接 S3C4510B 的 nWEB<3>、nWEB<2>,低 16 位片的 UDQM、LDQM 分别接 S3C4510B 的 nWEB<1>、nWEB<0>。

5.3.6 串行接口电路

几乎所有的微控制器、PC 都提供串行接口,使用电子工业协会 (EIA) 推荐的 RS-232-C 标准,这是一种很常用的串行数据传输总线标准。早期它被应用于计算机和终端通过电话线和 MODEM 进行远距离的数据传输,随着微型计算机和微控制器的发展,不仅远距离,近距离也采用该通信方式。在近距离通信系统中,不再使用电话线和 MODEM,而直接进行端到端的连接。

RS-232-C 标准采用的接口是 9 芯或 25 芯的 D 型插头,以常用的 9 芯 D 型插头为例,各引脚定义如表 5-3-3 所示:

表 5-3-3 9 芯 D 型插头引脚信号描述

| 引脚 | 名称 | 功能描述 |
|----|-----|---------|
| 1 | DCD | 数据载波检测 |
| 2 | RXD | 数据接收 |
| 3 | TXD | 数据发送 |
| 4 | DTR | 数据终端准备好 |
| 5 | GND | 地 |
| 6 | DSR | 数据设备准备好 |
| 7 | RTS | 请求发送 |
| 8 | CTS | 清除发送 |

| | | |
|---|----|------|
| 9 | RI | 振铃指示 |
|---|----|------|

要完成最基本的串行通信功能，实际上只需要 RXD、TXD 和 GND 即可，但由于 RS-232-C 标准所定义的高、低电平信号与 S3C4510B 系统的 LVTTL 电路所定义的高、低电平信号完全不同，LVTTL 的标准逻辑“1”对应 2V~3.3V 电平，标准逻辑“0”对应 0V~0.4V 电平，而 RS-232-C 标准采用负逻辑方式，标准逻辑“1”对应 -5V~-15V 电平，标准逻辑“0”对应 +5V~+15V 电平，显然，两者间要进行通信必须经过信号电平的转换，目前常使用的电平转换电路为 MAX232，其引脚分布如图 5.3.10。

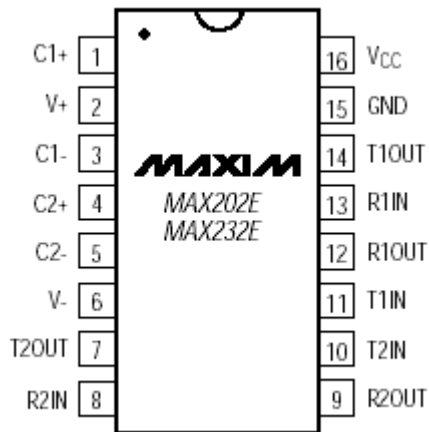


图 5.3.10 MAX232 引脚分布

关于 MAX232 更具体的内容可参考 MAX232 的用户手册。
图 5.3.11 为 MAX232 的常见应用电路图。

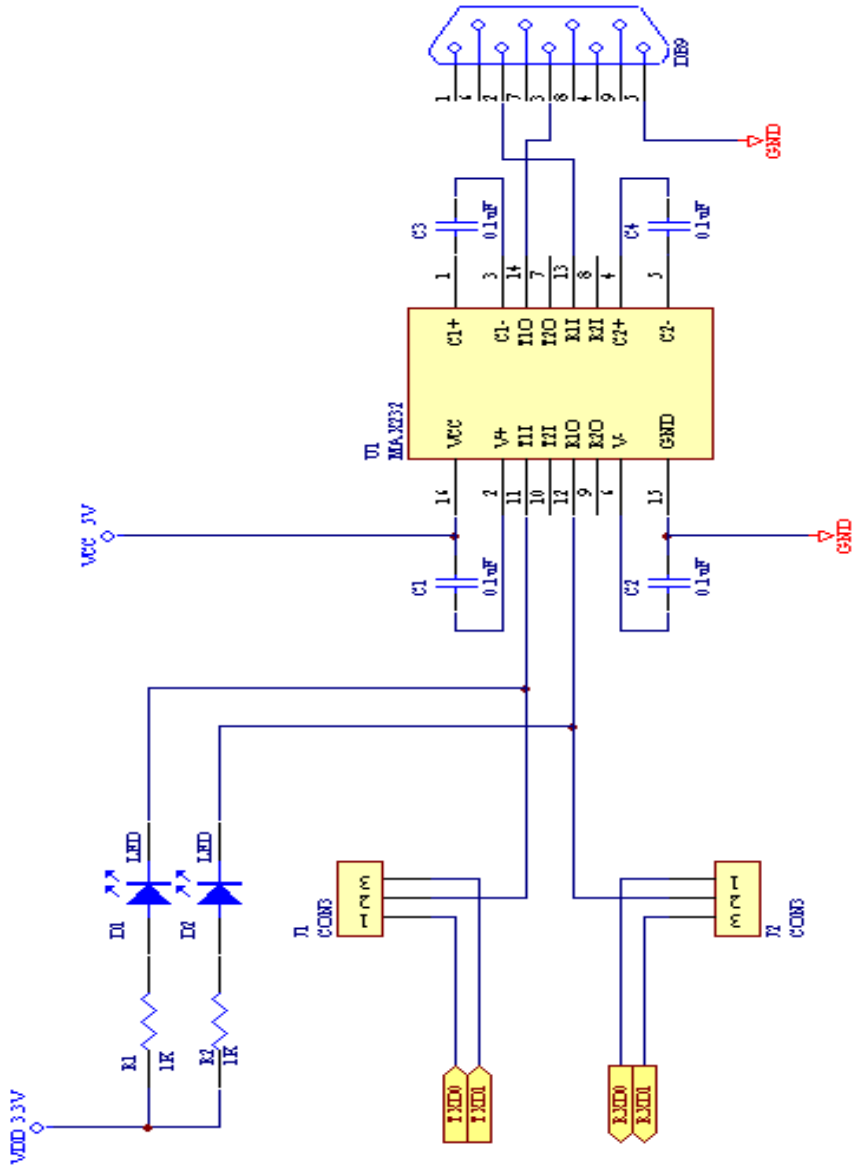


图 5.3.11 MAX232 的常见应用电路图

为缩小电路板面积，系统只设计了一个 9 芯的 D 型插头，通过两个跳线选择 S3C4510B 的 UART0 或 UART1，同时设计数据发送与接收的状态指示 LED，当有数据通过串行口传输时，LED 闪烁，便于用户掌握其工作状态以及进行软、硬件的调试。

5.3.7 IIC 接口电路

IIC 总线是一种用于 IC 器件之间连接的二线制总线。它通过 SDA(串行数据线) 及 SCL (串行时钟线) 两线在连接到总线上的器件之间传送信息，并根据地址识别每个器件：不管是微控制器、存储器、LCD 驱动器还是键盘接口。带有 IIC 总线接口的器件可十分方便地用来将一个或多个微控制器及外围器件构成系统。尽管这种总线结构没有并行总线那样大的吞吐能力，但由于连接线和连接引脚少，因此其构成的系统价格低，器件间总线简单，结构紧凑，而且在总线上增加器件不影响系统的正常工作，系统修改和可扩展性好。即使有不同时钟速度的器件连接到总线上，也能很方便地确定总线的时钟，因此在嵌入式系统中得到了广泛的应用。

S3C4510B 内含一个 IIC 总线主控器，可方便的与各种带有 IIC 接口的器件相连。在该系统中，外扩一片 AT24C01 作为 IIC 存储器。AT24C01 提供 128 字节的 EEPROM 存储空间，可用于存放少量在系统掉电时需要保存的数据。

AT24C01 引脚分布及信号描述和应用电路如图 5.3.12、图 5.3.13 所示：

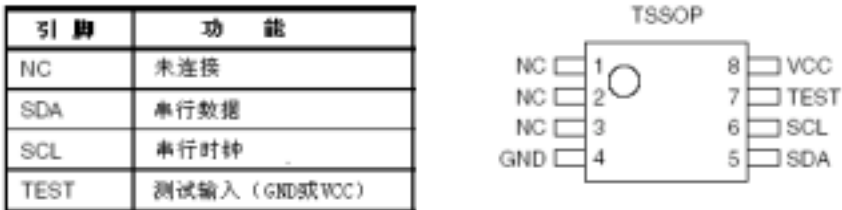


图 5.3.12 AT24C01 引脚分布及信号描述

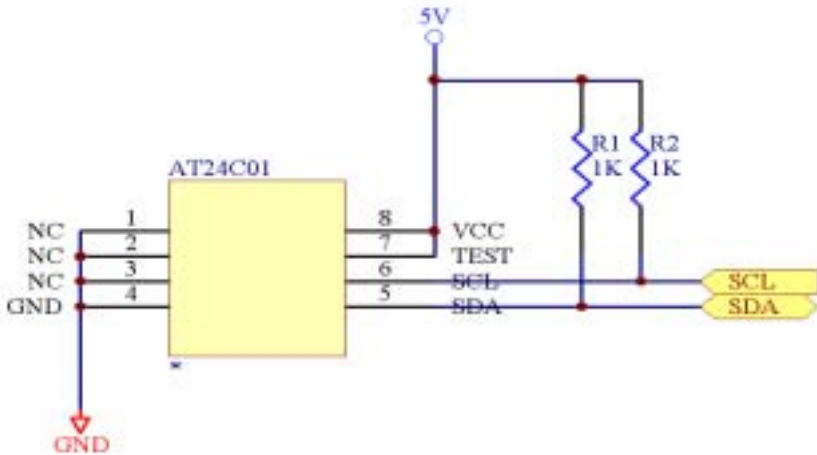


图 5.3.13 AT24C01 应用电路

5.3.8 JTAG 接口电路

JTAG(Joint Test Action Group, 联合测试行动小组)是一种国际标准测试协议,主要用于芯片内部测试及对系统进行仿真、调试, JTAG 技术是一种嵌入式调试技术,它在芯片内部封装了专门的测试电路 TAP (Test Access Port, 测试访问口),通过专用的 JTAG 测试工具对内部节点进行测试。目前大多数比较复杂的器件都支持 JTAG 协议,如 ARM、DSP、FPGA 器件等。标准的 JTAG 接口是 4 线:TMS、TCK、TDI、TDO,分别为测试模式选择、测试时钟、测试数据输入和测试数据输出。

JTAG 测试允许多个器件通过 JTAG 接口串联在一起,形成一个 JTAG 链,能实现对各个器件分别测试。JTAG 接口还常用于实现 ISP (In-System Programmable 在系统编程)功能,如对 FLASH 器件进行编程等。

通过 JTAG 接口,可对芯片内部的所有部件进行访问,因而是开发调试嵌入式系统的一种简洁高效的手段。目前 JTAG 接口的连接有两种标准,即 14 针接口和 20 针接口,其定义分别如下所示。

14 针 JTAG 接口定义:

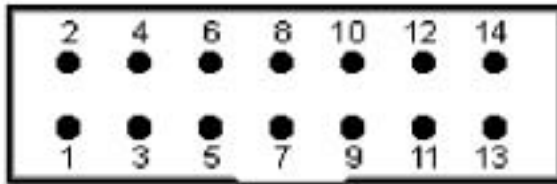


表 5-3-4 14 针 JTAG 接口定义

| 引 脚 | 名 称 | 描 述 |
|---------------|-------|----------|
| 1、13 | VCC | 接电源 |
| 2、4、6、8、10、14 | GND | 接地 |
| 3 | nTRST | 测试系统复位信号 |
| 5 | TDI | 测试数据串行输入 |
| 7 | TMS | 测试模式选择 |
| 9 | TCK | 测试时钟 |
| 11 | TDO | 测试数据串行输出 |
| 12 | NC | 未连接 |

20 针 JTAG 接口定义:

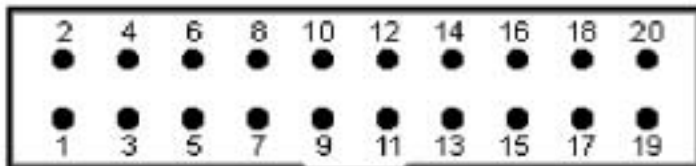


表 5-3-5 20 针 JTAG 接口定义

| 引 脚 | 名 称 | 描 述 |
|-----------------------------|--------|--------------|
| 1 | VTref | 目标板参考电压, 接电源 |
| 2 | VCC | 接电源 |
| 3 | nTRST | 测试系统复位信号 |
| 4、6、8、10、12、 14、16、18、20 | GND | 接地 |
| 5 | TDI | 测试数据串行输入 |
| 7 | TMS | 测试模式选择 |
| 9 | TCK | 测试时钟 |
| 11 | RTCK | 测试时钟返回信号 |
| 13 | TDO | 测试数据串行输出 |
| 15 | nRESET | 目标系统复位信号 |
| 17、19 | NC | 未连接 |

5.3.9 10M/100M 以太网接口电路

作为一款优秀的网络控制器, 基于 S3C4510B 的系统若没有以太网接口, 其应用价值就会大打折扣, 因此, 就整个系统而言, 以太网接口电路应是必不可少的, 但同时也是相对较复杂的。从硬件的角度看, 以太网接口电路主要由 MAC 控制器和物理层接口 (Physical Layer, PHY) 两大部分构成, 目前常见的以太网接口芯片, 如 RTL8019、RTL8029、RTL8039、CS8900、DM9008 等, 其内部结构也主要包含这两部分。

S3C4510B 内嵌一个以太网控制器, 支持媒体独立接口 (Media Independent Interface, MII) 和带缓冲 DMA 接口 (Buffered DMA Interface, BDI)。可在半双工或全双工模式下提供 10M/100Mbps 的以太网接入。在半双工模式下, 控制器支持 CSMA/CD 协议, 在全双工模式下支持 IEEE802.3 MAC 控制层协议。

因此, S3C4510B 内部实际上已包含了以太网 MAC 控制, 但并未提供物理层接口, 因此, 需外接一片物理层芯片以提供以太网的接入通道。

常用的单口 10M/100Mbps 高速以太网物理层接口器件主要有 RTL8201、DM9161 等, 均提供 MII 接口和传统 7 线制网络接口, 可方便的与 S3C4510B 接口。以太网物理层接口器件主要功能一般包括: 物理编码子层、物理媒体附件、双绞线物理媒体子层、10BASE-TX 编码/解码器和双绞线媒体访问单元等。

在该系统中, 使用 RTL8201 作为以太网的物理层接口。图 5.3.14 为 RTL8201 的引脚分布, 表 6-3-6 相关引脚功能描述, 表中仅列出芯片在 100Mbps MII 接口方式下的引脚定义, 当工作于 7 线制网络接口方式, 部分引脚定义不同。更具体的内容和使用方法可参考 RTL8201 的用户手册。

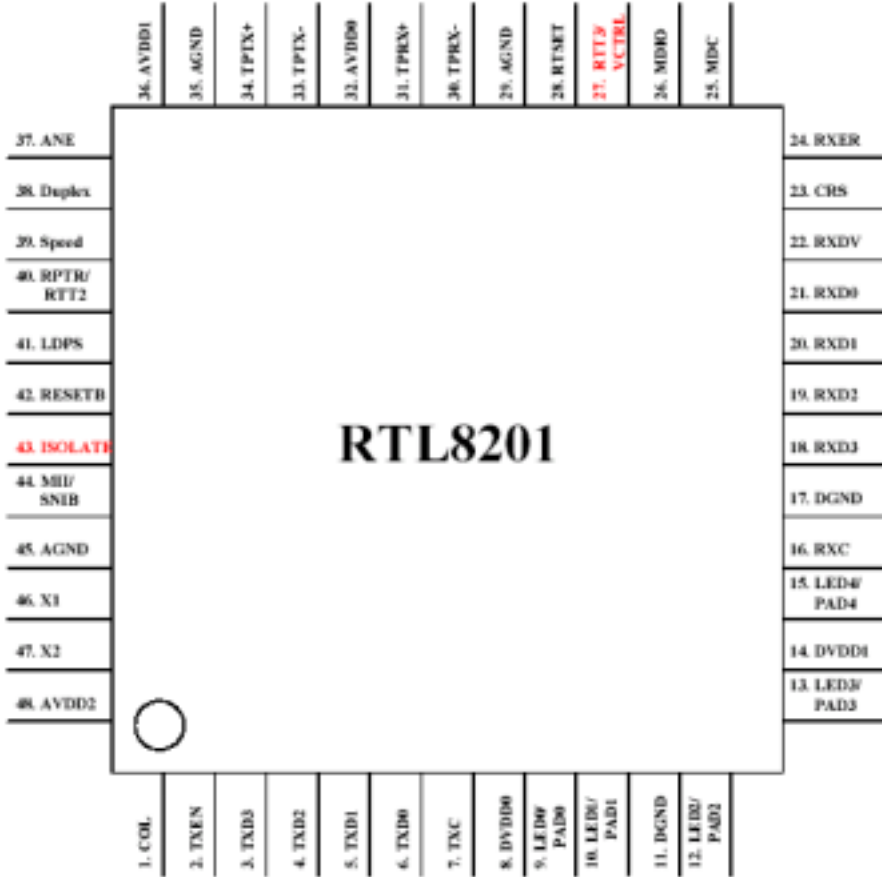


图 5.3.14 RTL8201 的引脚分布

表 5-3-6 RTL8201 引脚功能描述

| 信号 | 类型 | 引脚 | 功能描述 |
|----------|-----|-------------|--|
| TXC | O | 7 | 发送时钟:该引脚提供连续时钟信号作为 TXD[3:0]和 TXEN 的时序参考。 |
| TXEN | I | 2 | 发送使能:该引脚指示目前 TXD[3:0]上的 4 位信号有效。 |
| TXD[3:0] | I | 3,4,5,6 | 发送数据:当 TXEN 有效时,MAC 随 TXC 同步送出 TXD[3:0]。 |
| RXC | O | 16 | 接收时钟:该引脚提供连续时钟信号作为 RXD[3:0]和 RXDV 的时序参考,在 100Mbps 时,RXC 的频率为 25MHz,10Mbps 时为 2.5MHz。 |
| COL | O | 1 | 冲突检测:当检测到冲突时,COL 置为高电平。 |
| CRS | I/O | 23 | 载波侦听:在非 IDLE 状态时,该引脚置为高电平。 |
| RXDV | O | 22 | 接收数据有效:当接收 RXD[3:0]上的数据时,该引脚置高电平,接收结束时置低电平。该信号在 RXC 的上升沿有效。 |
| RXD[3:0] | O | 18,19,20,21 | 接收数据:该引脚随 RXC 同步将数据从 PHY 传送给 MAC。 |

| | | | |
|------------------|--------|---------------|--|
| RXER | O | 24 | 接收错误：当接收数据发生错误时，该引脚置为高电平。 |
| MDC | I | 25 | 站管理时钟信号：该引脚为 MDIO 提供同步时钟信号，但可能与 TXC 和 RXC 时钟异步。该时钟信号最高可达 2.5MHz。 |
| MDIO | I/O | 26 | 站数据输入输出：该引脚提供用于站管理的双向数据信息。 |
| X2 | O | 47 | 25MHz 晶振输出：该引脚提供 25MHz 晶振输出。当 X1 外接 25MHz 振荡器时，该引脚必须悬空。 |
| X1 | I | 46 | 25MHz 晶振输入：该引脚提供 25MHz 晶振输入。当外接 25MHz 振荡器时，该引脚作为输入。 |
| TPTX + TPTX - | O O | 34, 33 | 发送输出。 |
| RTSET | I | 28 | 发送差分电阻连接：该引脚应通过— 2.0K 的电阻下拉。 |
| TPRX + TPRX - | I I | 31, 30 | 接收输入。 |
| ISOLATE | I | 43 | 该引脚置高将 RTL8201 与 MAC 和 MDC/MDIO 管理接口隔离。在该模式下，功耗最小。 |
| RPTR/ RTT2 | I | 40 | 该引脚置高将 RTL8201 设为转发器工作模式。在测试模式下，该引脚重定义为 RTT2。 |
| SPEED | I | 39 | 该引脚置高 RTL8201 以 100Mbps 的速率工作。 |
| DUPLEX | I | 38 | 该引脚置高使能全双工模式。 |
| ANE | I | 37 | 该引脚置高使能自动协议模式，置低为强制模式。 |
| LDPS | I | 41 | 该引脚置高 RTL8201 进入未连接省电（LDPS）模式。 |
| MII/SNIB | I | 44 | 该引脚置高 RTL8201 进入 MII 模式工作。 |
| LED0/PAD0 | O | 9 | 连接 LED 显示。 |
| LED1/PAD1 | O | 10 | 全双工 LED 显示。 |
| LED1/PAD2 | O | 12 | 10M 连接/应答 LED 显示。 |
| LED1/PAD3 | O | 13 | 100M 连接/应答 LED 显示。 |
| LED1/PAD4 | O | 15 | 冲突 LED 显示。 |
| RTT3/CTRL | O | 27 | 目前用作测试，可作为将来的功能扩展。 |
| RESETB | I | 42 | 芯片复位引脚，低电平复位。 |
| AVDD0/ AVDD1 | P | 32, 36 | 模拟电源：为片内模拟电路提供 3.3V 电源，应接去耦合电容。 |
| AVDD2 | P | 48 | 为片内 PLL 电路提供 3.3V 电源，应接去耦合电容并用 100ohm@100MHz 磁珠接到模拟地。 |
| AGND | P | 29, 35, 45 | 模拟地：接地。 |
| DVDD0/ DVDD1 | P | 8, 14 | 数字电源：为片内数字电路提供 3.3V 电源。 |
| DGND | P | 11, 17 | 数字地：接地 |

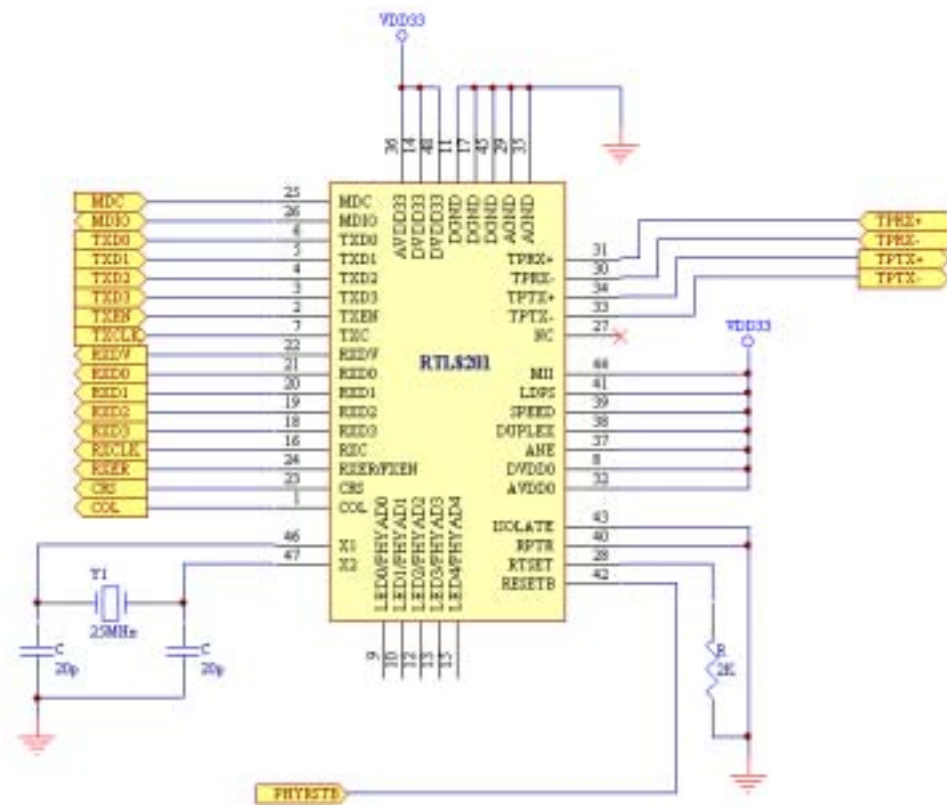


图 5.3.15 RTL8201 应用电路图

由于 S3C4510B 片内已有带 MII 接口的 MAC 控制器，而 RTL8201 也提供了 MII 接口，各种信号的定义也很明确，因此 RTL8201 与 S3C4510B 地连接比较简单。图 5.3.15 为 RTL8201 地实际应用电路图。

S3C4510B 的 MAC 控制器可通过 MDC/MDIO 管理接口控制多达 31 个 RTL8201，每个 RTL8201 应有不同的 PHY 地址（可从 00001B 到 11111B）。当系统复位时，RTL8201 锁存引脚 9，10，12，13，15 的初始状态作为与 S3C4510B 管理接口通信的 PHY 地址，但该地址不能设为 00000B，否则 RTL8201 进入掉电模式。

为减少芯片的引脚数，RTL8201 的 LED 引脚同时复用为 PHY 的地址引脚，因此引脚 9，10，12，13，15 不能直接接到电源或地。图 5.3.16 为引脚 9，10，12，13，15 地连接方法，此时 RTL8201 的 PHY 地址为 00001B。引脚通过 5.1K 的电阻上拉或下拉，决定 RTL8201 的 PHY 地址，在正常工作时，LED 显示 RTL8201 的工作状态，当不需要 LED 状态显示时，LED + 510 欧姆的电阻可去掉。

在图 5.3.15 中，信号地发送和接收端应通过网络隔离变压器和 RJ45 接口接入传输媒体，其实际应用电路见图 5.3.17。

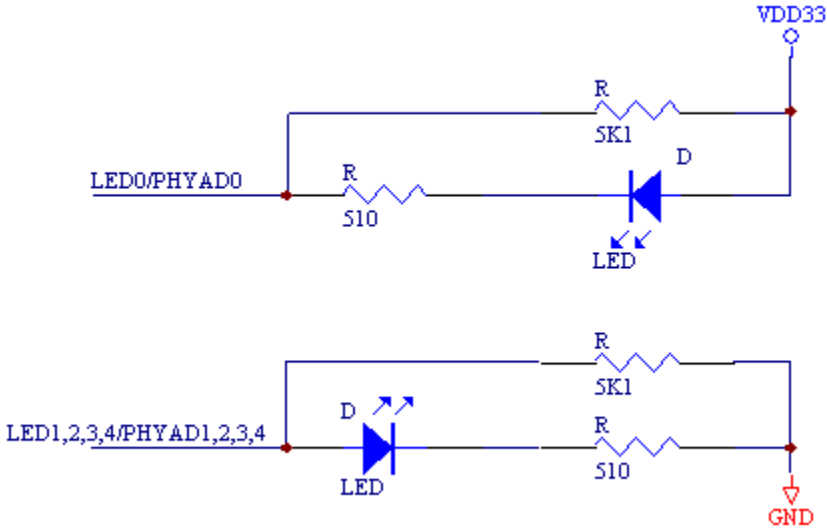


图 5.3.16 RTL8201 的 LED 与 PHY 地址配置

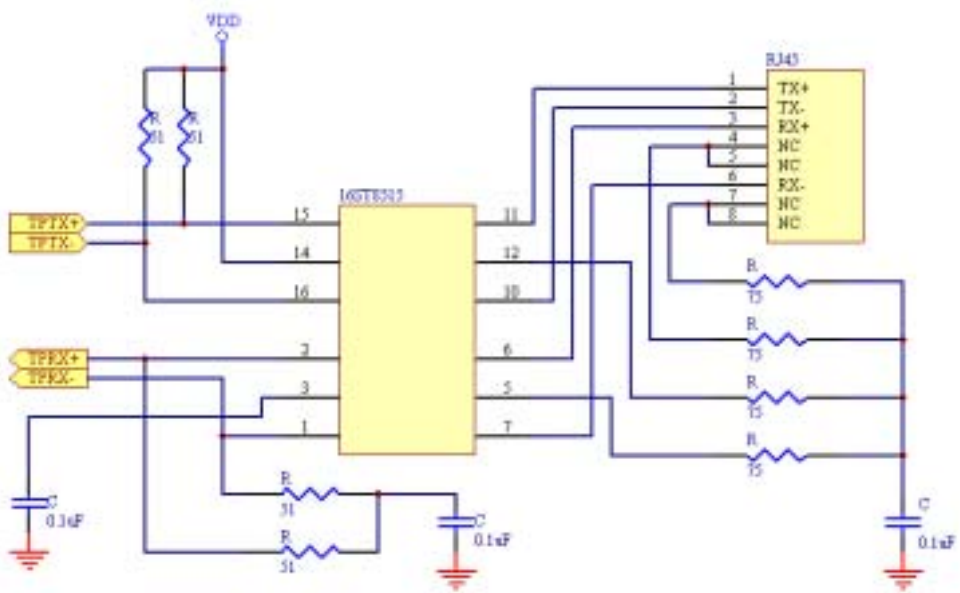


图 5.3.17 RTL8201 与网络隔离变压器及 RJ45 的连接图

5.3.10 通用 I/O 接口电路

S3C4510B 提供了 18 个可编程的 I/O 端口, 用户可将每个端口配置为输入模式、输出模式或特殊功能模式, 由片内的特殊功能寄存器控制。在该系统的设计中, P3 ~ P0 外接 4 只 LED 显示器, 用作程序运行状态的显示或其他输出功能, P7 ~ P4 外接跳线选择高、低电平用作状态输入, 以控制程序流程或其他输入功能, 其应用电路如图 5.3.18。

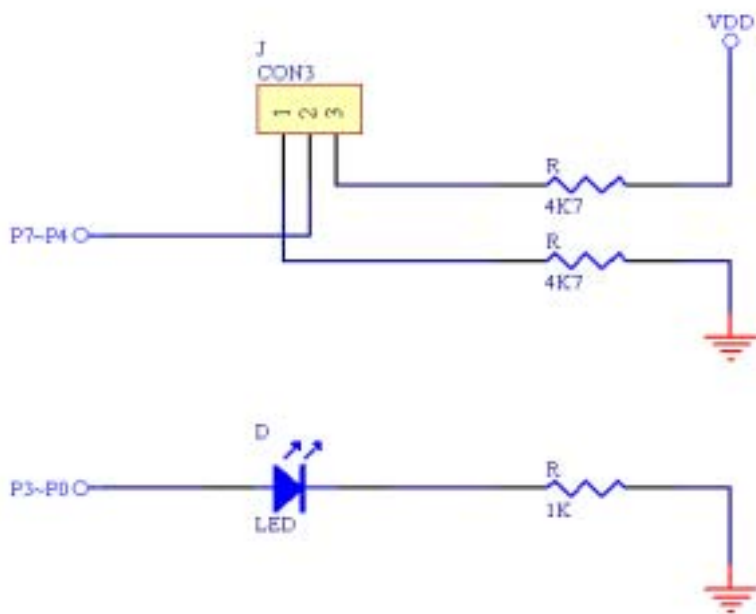


图 5.3.18 通用 I/O 口用作输入输出

5.4 硬件系统的调试

通过 5.3 节对 S3C4510B 应用系统设计方法的阅读，具有一定系统开发经验的读者就可以设计开发自己的特定应用系统，上节所介绍的内容实际上可理解为一个基于 S3C4510B 的最小系统，读者可根据自己的实际需要做适当的增减。

当系统设计制作完成时，必须经过仔细的调试，才能保证系统按照设计意图正常工作。尽管系统的调试与个人对电路工作原理的理解和实际的电路调试经验有很大的关系，但一定的调试方法也是必不可少的。掌握正确的调试方法可使调试工作变得容易，大大缩短系统的开发时间，反之，可能会使整个系统的开发前功尽弃，以失败告终。

本节以单元电路为单位，并结合笔者自身在系统调试时所遇到的一些具有代表性的问题，循序渐进的介绍整个系统的调试过程。在此，笔者建议：当用户的印制电路板制作完毕后，不要急于焊接元器件，请首先对照原理图仔细检查印制电路板的连线，确保无误后方可焊接。同时，尽可能的以各单元电路为单位，一个个焊接调试，以便在调试过程中遇到困难时缩小故障范围，在系统上电后，应先检查电路工作有无异常，芯片在工作时有一定的发热是正常的，但如果有芯片特别发烫，则一定有故障存在，需断电检查确认无误后方可继续通电调试。

调试工具需要示波器、万用表等，同时需要 ARM 调试开发软件 ADS 或 SDT 及相应的仿真器，本系统在调试时使用 ADS1.2 及由北京微芯力科技有限公司开发的 ARM JTAG 仿真器。关于 ADS 的使用方法在以后的章节有详细的叙述。

5.4.1 电源、晶振及复位电路

电源电路、晶振电路和复位电路相对比较简单，按图 5.3.1、图 5.3.2 和图 5.3.3 连接后应该就可以正常工作，此时电源电路的输出因为 DC 3.3V。

用示波器观测，有源晶振的输出应为 10MHz；

复位电路的 RESET 端在未按按钮时输出应为高电平（3.3V），按下按钮后变为低电平，按钮松开后应恢复到高电平。

电源电路、晶振电路和复位电路是整个系统正常工作的基础，应首先保证他们的正常工作。

5.4.2 S3C4510B 及 JTAG 接口电路

在保证电源电路、晶振电路和复位电路正常工作的前提下，可通过 JTAG 接口调试 S3C4510B，在系统上电前，首先应检测 JTAG 接口的 TMS、TCK、TDI、TDO 信号是否已与 S3C4510B 的对应引脚相连，其次应检测 S3C4510B 的 nEWAIT 引脚（Pin71）是否已上拉，ExtMREQ 引脚（Pin108）是否已下拉，对这两只引脚的处理应注意，作者遇到多起 S3C4510B 不能正常工作或无法与 JTAG 接口通信，均与没有正确处理这两只引脚有关。

给系统上电后，可通过示波器查看 S3C4510B 对应引脚的输出波形，判断是否已正常工作，若 S3C4510B 已正常工作，在使能片内 PLL 电路的情况下，SDCLK/MCLKO 引脚（Pin77）应输出频率为 50MHz 的波形，同时 MDC 引脚（Pin50）和其他一些引脚也应有波形输出。

在保证 S3C4510B 已正常工作的情况下，可使用 ADS 或 SDT 通过 JTAG 接口对片内的部件进行访问和控制。

在此，首先通过对片内控制通用 I/O 口的特殊功能寄存器的操作，来点亮连接在 P3 ~ P0 口上的 4 只 LED，用以验证 ADS 或 SDT 调试环境是否已正确设置，以及与 JTAG 接口的连接是否正常。

ADS 和 SDT 均为 ARM 公司为方便用户在 ARM 芯片上进行应用开发而推出的一整套集成开发工具，其中，ADS 为 SDT 的升级版本。该系统的调试以 ADS 为例，同时也适合于 SDT 开发环境。图 5.4.1 为调试系统的硬件连接。

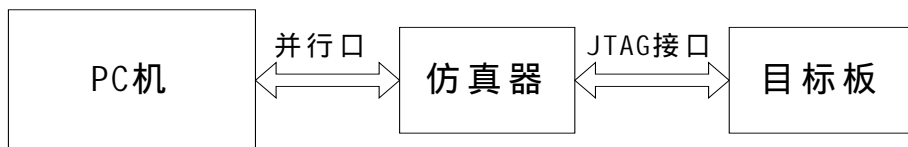


图 5.4.1 调试系统的硬件连接

按图 5.4.1 连接好硬件后，打开 AXD Debugger，建立与目标板（待调试的系统板）的连接，AXD Debugger 有软件仿真方式和带目标系统的调试方式，此时应在带目标系统的调试方式。

选择菜单 System Views Command Line Interface 功能，该选项为 AXD

Debugger 的一个命令行窗口,可在该窗口内输入各种调试命令,使用非常方便。在命令行窗口输入:

```
>setmem 0x3FF5000, 0xFFFF, 32
>setmem 0x3FF5008, 0xFFFF, 32
```

setmem 命令用于对特定的地址设置特定的值,待设定的值可以是 8 位、16 位或 32 位,在此,对通用 I/O 口的模式寄存器 and 数据寄存器设置相应的值,点亮 LED。

S3C4510B 在复位后,特殊功能寄存器的基地址为 0x3FF0000,由表 5-2-3 可知,I/O 口的模式寄存器偏移地址为 0x5000,因此,I/O 口的模式寄存器的物理地址为 0x3FF5000,设定该寄存器的值为 0xFFFF,将 I/O 口置为输出方式。I/O 口的数据寄存器的物理地址为 0x3FF5008,设定该寄存器的值为 0xFFFF,将 I/O 口的输出置为高电平。

在执行完以上两条命令后,连接在通用 I/O 口的 4 只 LED 应被点亮,表示调试系统的软、硬件连接完好,可进行下一步的调试工作,否则,应重新检查调试系统。

用户若使用 SDT 作为调试工具,操作方法类似。按图 5.4.1 连接好硬件后,打开 ARM Debugger for Windows,建立与目标板(待调试的系统板)的连接,选择菜单 View Command 功能,即可显示命令行窗口,在命令行窗口输入:

```
Debug: let 0x3FF5000 = 0xFFFF
Debug: let 0x3FF5008 = 0xFFFF
```

执行完以上两条命令后,连接在通用 I/O 口的 4 只 LED 应被点亮。

关于通用 I/O 口更具体的工作原理和使用方法,可参考 S3C4510B 用户手册。

用户系统若能正常完成上述操作并成功点亮连接在 P3~P0 口上的 LED 显示器,则表明 S3C4510B 已在正常工作,且调试环境也已正确建立,以后的调试工作就相对简单。笔者曾遇到多个用户系统因为不能完成这步工作,使开发者失去信心而最终放弃。

5.4.3 SDRAM 接口电路的调试

在系统的两类存储器中,SDRAM 相对于 FLASH 存储器控制信号较多,似乎调试应该困难一些,但由于 SDRAM 的所有刷新及控制信号均由 S3C4510B 片内的专门部件控制,无需用户干预,在 S3C4510B 正常工作的前提下,只要连线无误,SDRAM 就应能正常工作,反之,Flash 存储器的编程、擦除操作均需要用户编程控制,且程序还应在 SDRAM 中运行,因此,应先调试好 SDRAM 存储器系统,再进行 Flash 存储器系统的调试。

在进行存储器系统调试之前,用户必须深入了解 S3C4510B 系统管理器关于存储器映射的工作原理。

基于 S3C4510B 系统的最大可寻址空间为 64MB,采用统一编址的方式,将系统的 SDRAM、SRAM、ROM、Flash、外部 I/O 以及片内的特殊功能寄存器和 8K 一体化 SRAM 均映射到该地址空间。为便于使用与管理,S3C4510B 又将 64MB 的地址空间分为若干个组,分别由相应的特殊功能寄存器进行控制:

- ROM/SRAM/Flash 组 0~ROM/SRAM/Flash 组 5,用于配置 ROM、SRAM 或 Flash,

分别由特殊功能寄存器 ROMCON0 ~ ROMCON5 控制；

- DRAM/SDRAM 组 0 ~ DRAM/SDRAM 组 3 用于配置 DRAM 或 SDRAM，分别由特殊功能寄存器 DRAMCON0 ~ DRAMCON3 控制；

- 外部 I/O 组 0 ~ 外部 I/O 组 3 用于配置系统的其他外扩接口器件，由特殊功能寄存器 REFEXTCON 控制；

- 特殊功能寄存器组用于配置 S3C4510B 片内特殊功能寄存器的基地址以及片内的 8K 一体化 SRAM，由特殊功能寄存器 SYSCFG 控制；

在该系统中，使用了 Flash 存储器和 SDRAM，分别配置在 ROM/SRAM/FLASH 组 0 和 DRAM/SDRAM 组 0，暂未使用外扩接口器件。

参照表 5-2-4 和 5.2.4 节对应特殊功能寄存器的相关描述可知，当系统复位时，只有 ROM/SRAM/FLASH 组 0 被映射到地址空间为 0x0000,0000 ~ 0x0200,0000 的位置，特殊功能寄存器的基地址被映射到 0x03FF,0000，片内 8K 一体化 SRAM 的起始地址被映射到 0x03FE,0000，它们是可访问的，而其他的存储器组均未被映射，是不可访问的。

因此，要调试 SDRAM 存储器系统，首先应配置相关的特殊功能寄存器，使系统中的 SDRAM 能被访问。表 5-4-1 为针对该系统的与系统管理器相关的特殊功能寄存器的配置，以下详细说明该系统所使用的相关特殊功能寄存器的配置方法。

表 5-4-1 系统管理器相关特殊功能寄存器的配置

| 寄存器 | 偏移量 | 配置值 |
|-----------|--------|-------------|
| SYSCFG | 0x0000 | 0xE7FF,FF82 |
| EXTDBWTH | 0x3010 | 0x0000,3000 |
| ROMCON0 | 0x3014 | 0x0200,0060 |
| DRAMCON0 | 0x302C | 0x1401,0380 |
| REFEXTCON | 0x303C | 0xCE33,83FD |

SYSCFG = 0xE7FF,FF82；其含义为：

为 4 个 DRAM 组选择 SDRAM；特殊功能寄存器组的基地址为 0x03FF,0000；片内 SRAM 基地址为 0x03FE,0000；4KB 配置为 SRAM，另外 4KB 配置为 Cache；使能 Cache 操作。

EXTDBWTH = 0x0000,3000；其含义为：

4 个外部 I/O 组禁用，DRAM/SDRAM 组 0 配置为 32 位数据宽度，其余的 DRAM/SDRAM 组禁用；ROM/SRAM/FLASH 组 0 由 B0SIZE[1:0]的状态配置，其余 ROM/SRAM/FLASH 组禁用。

ROMCON0 = 0x0200,0060；其含义为：

该系统共有 2MB 的 FLASH 存储器，映射到地址空间的 0x0000,0000 ~ (0x0020,0000-1) 处。

DRAMCON0 = 0x1401,0380；其含义为：

该系统共有 16MB 的 SDRAM 映射到地址空间的 0x0040,0000 ~ (0x0140,0000-1) 处。

用户也可将 Flash 存储器和 SDRAM 映射到地址空间的其他位置，但注意组与组

之间的地址不要发生重叠。

DRAMCON0 = 0xCE33,83FD ; 其含义为 :

配置 SDRAM 的刷新计数值,刷新时间、刷新使能等。

系统管理器对应的其他特殊功能寄存器使用其复位值,用户也可根据自身系统的特定情况,对相关特殊功能寄存器进行配置。

在 C:\ 下建立文本文件 memmap.txt,其内容为 :

```
setmem 0x3FF0000,0xE7FFFF82,32
setmem 0x3FF3010,0x00003000,32
setmem 0x3FF3014,0x02000060,32
setmem 0x3FF302C,0x14010380,32
setmem 0x3FF303C,0xCE3383FD,32
```

打开 AXD Debugger 的命令行窗口,执行 obey 命令,配置对应的控制寄存器 :

```
>obey C:\memmap.txt
```

此时,文本文件 memmap.txt 中的几条控制寄存器配置命令已经执行完毕,Flash 存储器和 SDRAM 已分别映射到地址空间的 0x0000,0000 ~ (0x0020,0000-1) 和 0x0040,0000 ~ (0x0140,0000-1) 处。

选择菜单 Processor Views Memory 选项,出现存储器窗口,在存储器起始地址栏输入 SDRAM 的映射起始地址:0x0040,0000,数据区应显示 SDRAM 中的内容,此时所显示的内容为一些随机数。双击其中的任一数据,输入新的值,如输入 0xAA,若对应的存储单元能正确显示刚才输入的数据,则表明 SDRAM 存储器已能正常工作。

在连续的 4 个字节输入 0xAA,然后再输入 0x55,检测 32 位数据是否正确传输,若其中的某一位或几位数据出现错误,则多半是由于对应的数据线不通或连接错误所引起的。

在 SDRAM 能正确访问后,用户就可以将自己编写的各种应用程序,编译并下载到 SDRAM 中运行。

若使用 SDT 调试环境,调试过程与上述步骤相似,简述如下 :

在 C:\ 下建立文本文件 memmap.txt,其内容为 :

```
let 0x3FF0000 = 0xE7FFFF82
let 0x3FF3010 = 0x00003000
let 0x3FF3014 = 0x02000060
let 0x3FF302C = 0x14010380
let 0x3FF303C = 0xCE3383FD
```

打开 ARM Debugger for Windows 的命令行窗口 (View Command),执行 obey 命令 :

```
>obey C:\memmap.txt
```

此时,Flash 存储器和 SDRAM 已分别映射到地址空间的 0x0000,0000 ~ (0x0020,0000-1) 和 0x0040,0000 ~ (0x0140,0000-1) 处。

选择菜单 View Memory 选项,出现存储器的起始地址输入窗口,在此输入 SDRAM 的映射起始地址:0x0040,0000,数据区应显示 SDRAM 中的内容,此时所显示的内容为一些随机数。双击其中的任一数据,输入新的值,如输入 0xAA,若对应的存储单

元能正确显示刚才输入的数据，则表明 SDRAM 存储器已能正常工作。

5.4.4 Flash 接口电路的调试

Flash 存储器的调试主要包括 Flash 存储器的编程（烧写）和擦除，与一般的存储器件不同，用户只需对 Flash 存储器发出相应的命令序列，Flash 存储器通过内部嵌入的算法即可完成对芯片的操作，由于不同厂商的 Flash 存储器在操作命令上可能会有一些细微的差别，Flash 存储器的编程与擦除工具一般不具有通用性，这也是为什么 Flash 接口电路相对较难调试的原因之一，因此，应在理解 Flash 存储器编程和擦除的工作原理的情况下，根据不同型号器件对应的命令集，编写相应的程序对其进行操作。

打开 AXD Debugger 的命令行窗口，执行 obey 命令：

```
>obey C:\memmap.txt
```

此时，2MB 的 Flash 存储器映射到地址空间的 0x0000,0000 ~ 0x001F,FFFF 处，选择菜单 Processor Views Memory 选项，出现存储器窗口，在存储器起始地址栏输入 Flash 存储器的映射起始地址：0x0，数据区应显示 Flash 存储器中的内容，若 Flash 存储器为空，所显示的内容应全为 0xFF，否则应为已有的编程数据。双击其中的任一数据，输入新的值，对应存储单元的内容应不能被修改，此时可初步认定 Flash 存储器已被访问，但是否能对其进行正确的编程与擦除操作，还需要编程验证，通过程序对 Flash 存储器进行编程和擦除操作，放在下一章的内容里说明。

若使用 SDT 调试环境，调试过程与上述步骤相似。

5.4.5 10M/100M 以太网接口电路

以太网接口电路主要由 MAC 控制器和物理层接口（Physical Layer，PHY）两大部分构成，而 MAC 控制器在 S3C4510B 片内，需要用户作硬件调试的只是外接的物理层接口 RTL8201。由于 MAC 控制器的工作原理相对复杂，相应的特殊功能寄存器也比较多，在此不作详述，对此有兴趣的读者可参考已移植到 S3C4510B 的 uClinux 内核代码中对 MAC 控制器的驱动部分。

RTL8201 和 S3C4510B 均有 MII 接口，对应引脚及功能定义明确，只要正确连接，一般都能正常工作。当 RTL8201 正常工作在 100Mbps 状态时，其发送时钟引脚（Pin7）、接收时钟引脚（Pin16）均应有波形输出，同时，对应的 LED 指示灯也能正确指示芯片的工作状态。

5.5 印刷电路板的设计注意事项

在本章结束之前，对该系统的印刷电路板（PCB）设计中应注意的事项作一个简要的说明。

在系统中，S3C4510B 的片内工作频率为 50MHz，其以太网接口电路的工作速

率更高达 100MHz 以上，因此，在印刷电路板的设计过程中，应该遵循一些高频电路的设计基本原则，否则会使系统工作不稳定甚至不能正常工作。印刷电路板的设计人员应注意以下几个方面：

- 注意电源的质量与分配。
- 同类型信号线应该成组、平行分布。

5.5.1 电源质量与分配

在设计印刷电路板时，能给各个单元电路提供高质量的电源，就会使系统的稳定性大幅度的提高。但如何能提高电源的质量，常用的手段有以下几个：

1、电源滤波

为提高系统的电源质量，消除低频噪声对系统的影响，一般应在电源进入印刷电路板的位置和靠近各器件的电源引脚处加上滤波器，以消除电源的噪声，常用的方法是在这些位置加上几十到几百微法的电容。

同时，在系统中除了要注意低频噪声的影响，还要注意元器件工作时产生的高频噪声，一般的方法是在器件的电源和地之间加上 0.1 μ F 左右地电容，可以很好地滤出高频噪声的影响。

2、电源分配

实际的工程应用和理论都证实，电源的分配对系统的稳定性有很大的影响，因此，在设计印刷电路板时，要注意电源的分配问题。

在印刷电路板上，电源的供给一般采用电源总线（双面板）或电源层（多层板）的方式。电源总线由两条或多条较宽的线组成，由于受到电路板面积的限制，一般不可能布得过宽，因此存在较大的直流电阻，但在双面板得设计中也只好采用这种方式了，只是在布线的过程中，应尽量注意这个问题。

在多层板的设计中，一般使用电源层的方式给系统供电。该方式专门拿出一层作为电源层而不再在其上布信号线。由于电源层遍及电路板的全面积，因此直流电阻非常的小，采用这种方式可有效的降低噪声，提高系统的稳定性。

5.5.2 同类型信号线的分布

在各种微处理器的输入输出信号中，总有相当一部分是相同类型的，例如数据线、地址线。对这些相同类型的信号线应该成组、平行分布，同时注意它们之间的长短差异不要太大，采用这种布线方式，不但可以减少干扰，增加系统的稳定性，还可以使布线变得简单，印刷电路板的外观更美观。

以本系统的印刷电路板设计为例，成组的信号线主要是数据线和地址线，可在元器件位置确定后，首先完成他们的布线，尽可能做到成组、平行分布，同时应尽可能的短。然后在进行各种控制信号的布线，最后处理电源和接地引脚。

5.6 本章小节

本章主要介绍 S3C4510B 的基本结构和工作原理，同时介绍了设计一个基于 S3C4510B 的最小硬件系统的详细步骤、实现细节以及硬件系统的调试方法等内容。

需要说明，本章中所使用的器件和电路等，可能不是最优的，但可以保证是能正常工作的，在系统开发的过程中，不同的人会碰到不同的问题，本章的内容只是对笔者系统开发过程的一个描述和简要的总结，希望能对读者的系统设计工作有一点帮助。