

哈尔滨工程大学

硕士学位论文

基于DSP的图像采集与处理系统的设计

姓名：武颖

申请学位级别：硕士

专业：控制理论与控制工程

指导教师：叶秀芬

20090301

摘 要

图像采集与处理在信息处理技术中占有非常重要的地位，它涉及工业检测、医疗设备、军事、消费电子等诸多领域。传统的图像采集处理系统的架构一般采用计算机，但是这种架构功耗高、体积大、成本高，并且调试困难。针对以上问题，论文采用了 TMS320C6713 DSP 和 CPLD 的架构实现图像的采集与处理。

论文首先给出了系统的总体架构和工作原理。并简要介绍了系统组成中各个模块的功能。然后，结合 TMS320C6713 DSP 的特点和性能，对整个系统的工作流程进行了描述。

其次，论文设计了图像采集子板。该部分主要由图像传感器、图像缓冲器及其控制电路组成。图像传感器将模拟视频信号转化为数字视频数据，并分离出用于图像采集的同步信号。图像数据缓冲器 FIFO 用来暂存图像数据，以使采集时序匹配。图像采集的时序与逻辑控制由 CPLD 完成。论文同时给出了图像采集时序的仿真结果。

再次，设计了图像处理硬件平台，即设计了图像采集基板的硬件部分。该部分包含 DSP 最小系统、图像存储模块和图像传输模块。DSP 采用中断方式由 EDMA 控制器以 DMA 方式将 FIFO 暂存的数据搬运到外扩的 SDRAM 中；采用异步串口通信方式向上位机传送图像数据，上位机可实现图像的显示与保存。为实现系统的脱机运行，DSP 外扩了 Flash，可将程序烧写到 Flash 中，以 Bootloader 方式加载程序。

最后，论文在 DSP 上实现了图像处理算法：包括图像的二值化、采用 SOBEL 算子的图像边缘检测算法、采用 LAPLACE 算子的图像锐化算法以及图像的平滑处理算法等。文章给出了这些算法的原理以及处理结果。上述结果说明了论文设计的图像采集与处理系统的有效性和可行性。

关键词：DSP；CPLD；图像采集；图像处理；SAA7111A

ABSTRACT

Image acquisition and processing is an important branch of information processing technology. It has great utilities in the fields of industry measuring, medical device, military and commercial use. The traditional architecture of image acquisition is based on computer, but this architecture has disadvantages of large power consuming, big size and hard to debugging. Due to above problems, this dissertation proposed an image acquisition and processing system using DSP and CPLD. TMS320C6713 DSP, which is a 32-bit floating pointer processor and can work on 500MHz, was applied as the core processor to realize the algorithms. CPLD is to control video decoder to acquire image data. Because of its high speed and high accuracy, this system provided an ideal platform for image processing.

At the beginning of dissertation, overall architecture and working principle of the system was introduced. Function of each module was also introduced briefly. Moreover, the dissertation also described performance of TMS320C6713 DSP and the system working flow.

In addition, the hardware design of back circuit board was presented. This section consists of image sensor, image buffer and its control circuit. Video decoder, SAA7111A, worked as image sensor to convert analog video signal to digital video data and isolated the synchronous signal for image acquisition. Image data buffer was used to store image data temporarily in order to match acquisition timing between DSP and SAA7111A. Image acquisition timing match and logical control was completed by CPLD. At last, the simulation result of acquisition timing was also given.

Besides, hardware design of image processing platform was introduced in the dissertation. This section can be divided into DSP minimal system, image storage module and image transmission module. EDMA controller of DSP transported image data from FIFO to SDRAM via DMA mode. DSP sent image data to host computer via serial port and a whole image will be displayed on host computer.

An external flash chip is used to store DSP program. Bootloader design of the system was introduced.

At last, image processing algorithms were also realized on the DSP platform, including image binary, image edge detection with SOBEL operator, image sharpening with LPLACE operator and image smoothing. Algorithm principle and results were given in the dissertation.

Key Words: DSP; CPLD; Image acquisition; Image Processing; SAA7111A

哈尔滨工程大学 学位论文原创性声明

本人郑重声明：本论文的所有工作，是在导师的指导下，由作者本人独立完成的。有关观点、方法、数据和文献的引用已在文中指出，并与参考文献相对应。除文中已注明引用的内容外，本论文不包含任何其他个人或集体已经公开发表的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者（签字）：武毅

日期： 2009 年 3 月 10 日

哈尔滨工程大学 学位论文授权使用声明

本人完全了解学校保护知识产权的有关规定，即研究生在校攻读学位期间论文工作的知识产权属于哈尔滨工程大学。哈尔滨工程大学有权保留并向国家有关部门或机构送交论文的复印件。本人允许哈尔滨工程大学将论文的部分或全部内容编入有关数据库进行检索，可采用影印、缩印或扫描等复制手段保存和汇编本学位论文，可以公布论文的全部内容。同时本人保证毕业后结合学位论文研究课题再撰写的论文一律注明作者第一署名单位为哈尔滨工程大学。涉密学位论文待解密后适用本声明。

本论文（在授予学位后即可 在授予学位 12 个月后 解密后）由哈尔滨工程大学送交有关部门进行保存、汇编等。

作者（签字）：武毅

导师（签字）：叶美岩

日期： 2009 年 3 月 10 日

2009 年 3 月 10 日

第 1 章 绪论

1.1 论文研究的背景

随着计算机、多媒体和数据通信技术的高速发展，图像信号的采集与处理在科学研究、工农业生产、医疗卫生等部门得到越来越广泛的应用。同时，具有图像处理功能的嵌入式应用也随之增多，从数码相机、可视电话、多功能移动电话等消费产品到门禁、数字视频监控等工业控制及安防产品^[1]。图像采集需要进行同步信号的处理，比通常意义上的数据采集过程复杂，电路的设计也较为困难。传统 PC 机上的图像采集卡都是在 Philips、Brooktree 等半导体公司提供的接口芯片基础上，由专业公司开发生产的。近几年来，随着图像视频传输协议的不断开放，专用的数字信号处理芯片的不断发展，图像处理系统向着低功耗、小体积、高速和实时处理的方向发展。

1.2 图像采集处理系统发展概述

数字图像处理的发展与计算机以及硬件技术的发展是紧密联系的。最早发表有关计算机图像处理信息的文章的要追溯到 20 世纪 50 年代，随着计算机以及硬件技术的高速发展，性能的大幅度提高，而价格却大幅度下降，这些都无疑推动了图像处理技术的发展。关于图像采集处理系统的发展大致上可以划分为四个阶段^[2]。

1、图像采集处理系统发展的第一阶段

第一阶段的时间是 20 世纪 60 年代到 80 年代中期，这个时期的图像处理系统采用机箱式结构，主流计算机采用小型机，并采用双屏操作方式，所以系统的体积比较大，功能也比较强，当然价格也比较贵。当时的代表作是美国 I²S 公司推出的 MODEL-70、MODEL-50 图像计算机，英国 JOYCELOBEL 公司推出的 MAGISCAN 图像分析系统以及美国 VICOM 系统公司推出的 VICOM-VEM 图像处理工作站。我国图像处理系统的研制起步较晚，主要有清华大学的 TS79 小型通用图像处理系统、TJ82 图像计算机和 TS84 多功能微机图像处理系统。同样是采用机箱式结构，主流计算机采用小型机。

2、图像采集处理系统发展第二阶段

第二阶段的时间是 20 世纪 80 年代中期到 90 年代初期,这个阶段的主要特点是小型化,外形不再是机箱式而是插卡式,绝大部分都采用 PC 系列微机构成图像处理系统,计算机总线采用 ISA(Industrial Standard Architecture)总线,并采用双屏操作方式。图像卡的体积较小,一般图像卡都是采用大规模集成电路甚至是制作专用集成电路,从而使价格降低了。这个时期的代表作是美国 Imaging Technology 公司推出的 PCVISION 图像卡、PCVISION Plus 图像卡,美国 DT 公司推出的 DT2851 图像卡,加拿大 MATROX 公司的一系列图像卡。在国内,80 年代末到 90 年代初,中科院自动化研究所成功研制 CA 系列图像卡,清华大学成功研制 TH 系列图像卡。由于图像卡体积小、价格低、使用方便,所以很受用户欢迎。这阶段的图像卡大都开始采用大规模集成电路或是专用集成电路。这阶段特点是:主流机采用 PC 机,计算机总线采用 ISA 总线。

3、图像处理系统发展的第三阶段

第三阶段的时间是从 20 世纪 90 年代初开始,这一阶段图像处理系统突出特点是单屏方式,以微机 PCI 总线(Peripheral Component Interconnect bus)为支持的单屏方式和以图像压缩传输为特点的图像通信成为主流方式,但仍然主要是依靠微机来进行图像处理,在 Windows 平台上编制图像处理软件包,这个时期的代表有美国 Intel 公司推出的 MMX(多媒体指令系统)等。

4、基于 DSP 的图像处理系统

随着微型计算机的发展和普及,现代的图像处理方式越来越向高速、小型、简洁的方向发展,图像处理逐渐由专用、笨重的图像处理机过渡到通用、小型的微型机方式,但是由于图像的数据量很大,算法复杂程度高,人们经常使用软件来处理,软件往往局限于计算机的配置,使得图像处理速度比较慢、实时性差、价格高,不适宜在小规模、小环境内使用。与此同时数字信号处理各种算法日趋完善,特别是运算能力很强的数字信号处理器(DSP)的问世,使现代图像处理系统进入了和数字信号处理器紧密结合的全数字体制的阶段。以 DSP 为核心的硬件系统同样可以用来进行图像处理,为这个问题的解决带来了新的途径。由于美国 TI 公司在 DSP 市场上的主导地位,使得 TI 公司的图像处理平台在世界上处于领先地位。国内的 DSP 技术起步较晚,但发展很快。90 年代末就有比较成熟的产品出现。典型产品有闻亭公司的

WT6201P/AP、WT6701P/PA 图像处理系统和 WT32EA 通用图像处理系统。

随着 DSP 的运算速度和运算精度不断地提高，片内的存储容量不断地加大，系统接口功能、数据处理能力以及与外部设备的通信功能不断地增强，使得可以开发出，可以完全可以脱离 PC 机基于 DSP 的图像处理系统。这种设计方案的优点是设计简单、灵活，成本比较低，便于实际中使用。

1.3 论文研究的意义

由于图像技术在研究被测对象方面具有非接触、可重复性好以及应用对象广泛等特点，使得图像技术在工业控制、现代医学、现代战争、消费电子等方面具有十分广阔的市场前景。但它也有自身的不足，采集装置的复杂化，图像实时处理对处理器的依赖，图像处理系统的造价及体积都成为限制图像技术应用的瓶颈。比如采用 PCI 或 ISA 总线接口的图像采集卡都必须依赖计算机才可以实现图像的采集和处理，这样的配置使得系统在由于体积导致的额外成本及系统整体架构的复杂度方面的开销大大增加。而如果采用单片机之类的微处理器来取代计算机，又难以实现对数据的实时处理，从而导致系统性能指标的下降。随着半导体技术发展，DSP 的出现使得用微处理器取代计算机的想法得以实现。DSP 广泛应用在高速信号处理方面。它具有运算速度快的突出优点，每秒可以进行上百万次运算。其内部采用改进的哈佛结构，使得数字信号处理器的并行处理能力大大增强^[9]。本课题的提出就是要实现用 DSP 取代计算机来完成图像的采集及处理。课题在以下几个方面进行研究：模拟视频图像的数字化的、数字化图像的存储、图像数据的传输以及对图像数据的处理。由于采用 DSP 取代了计算机，使得本课题设计的图像采集处理系统具有体积小、成本省、功耗低的特点，能够很好的解决系统体积、成本在实际应用中带来的问题。尤其在消费电子领域将具有显著的社会经济效益和广阔的应用前景。

1.4 论文的主要研究内容及章节安排

论文将重点研究图像采集系统的硬件系统的设计与实现，板上各功能部件的驱动和调试以及图像处理算法在 DSP 上的实现等内容。论文的主要工作及章节安排如下：

第一章主要介绍了论文研究的背景、意义，并概述了图像采集处理系统的发展。

第二章设计基于 DSP 的高速图像采集系统的整体架构，并给出系统的工作流程，然后对系统的相关核心器件的选择进行论述。

第三章重点对图像采集子板的电路进行设计。分别对视频解码芯片、图像缓存部分以及采用 CPLD 的逻辑控制部分的实现方案进行了介绍。

第四章设计图像的存储、处理和传输系统。主要设计并实现图像处理算法的硬件平台，图像采集主板的 DSP 最小系统以及 DSP 与上位机的通信。

第五章对系统进行调试并实现 Bootloader 的功能。给出 DSP 实现程序的固化和加载的方法。

第六章将相关的一些图像处理算法移植到 DSP 中进行实现，并给出处理算法对应的实现效果。

结论部分对基于 DSP 实现的图像采集与处理系统设计进行总结，指出现有系统的不足，并针对此提出一些改进的措施。

第 2 章 系统整体架构的设计

基于 DSP 开发的图像采集系统可以满足体积小，成本低，功耗小等各方面的要求。由于图像采集时序的复杂性以及对图像处理速度的要求，使得我们必须选择合适的器件，并采用合理的系统架构。因此，在满足系统要求的前提下，应尽可能的减少系统资源的冗余，提高系统的集成度。

2.1 总体架构及工作流程

图像采集系统可分为前端系统和后端系统两个部分。其基本架构如图 2.1 所示。前端部分：主要有镜头和图像传感器组成。利用镜头将物体所反射的光聚集在图像传感器上，再用 ADC 将模拟图像的数据转化为数字图像的数据。后端部分：接收和处理电路，通常为数字信号处理器或 PC 机。

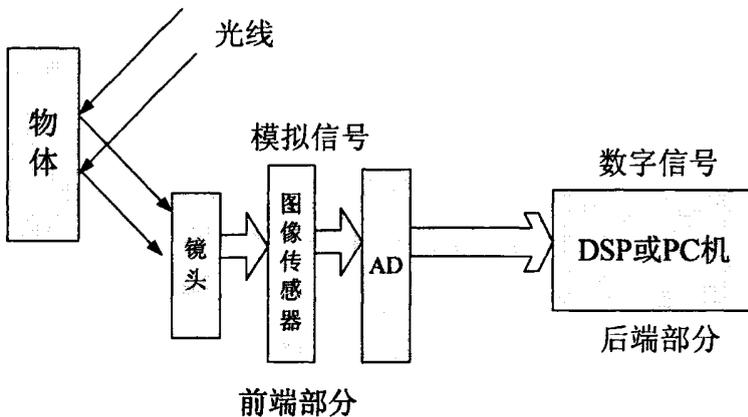


图 2.1 图像采集基本构架

实时图像处理的第一步是对来自图像传感器的视频信号进行实时采集。目前，图像传感器主要有：CCD(Charge Coupled Device)与CMOS(Complementary Metal Oxide Semiconductor)。CCD技术现在已经非常成熟，是高端应用的首选器件。它输出的模拟视频信号包括图像信号、行与场消隐信号、行与场同步信号等七种信号。传统的视频采集系统需要自行设计逻辑电路提取行、场同步信号，控制A/D转换器进行模数转换，再通过地址发生器产生地址，进行数据存储，此类采集控制电路复杂、芯片繁多、开

发周期长、成本高。因此，课题采用视频解码芯片 SAA7111A 来完成上述功能，从而降低了设计的复杂度。

2.1.1 系统总体架构

本论文设计的实时图像采集与处理系统的硬件总体设计方案如图2.2所示。系统主要由视频解码器，CPLD采集控制，FIFO高速缓存，DSP，图像数据传输等部分组成。图像处理模块选用具有适合于数字信号处理的软件和硬件资源的DSP来实现。而图像采集模块由视频解码芯片与CPLD构成，结构简洁，克服了传统采集卡的缺点。其中视频解码芯片SAA7111A可以从视频信号中自动获取行、场同步信号，并完成AD转换。具有在线系统编程(ISP)功能的CPLD可以通过软件实现逻辑重构，将逻辑控制、地址发生器、时序同步等电路全部集成在一块芯片中。整个系统具有体积小，升级容易，开发周期短的特点。

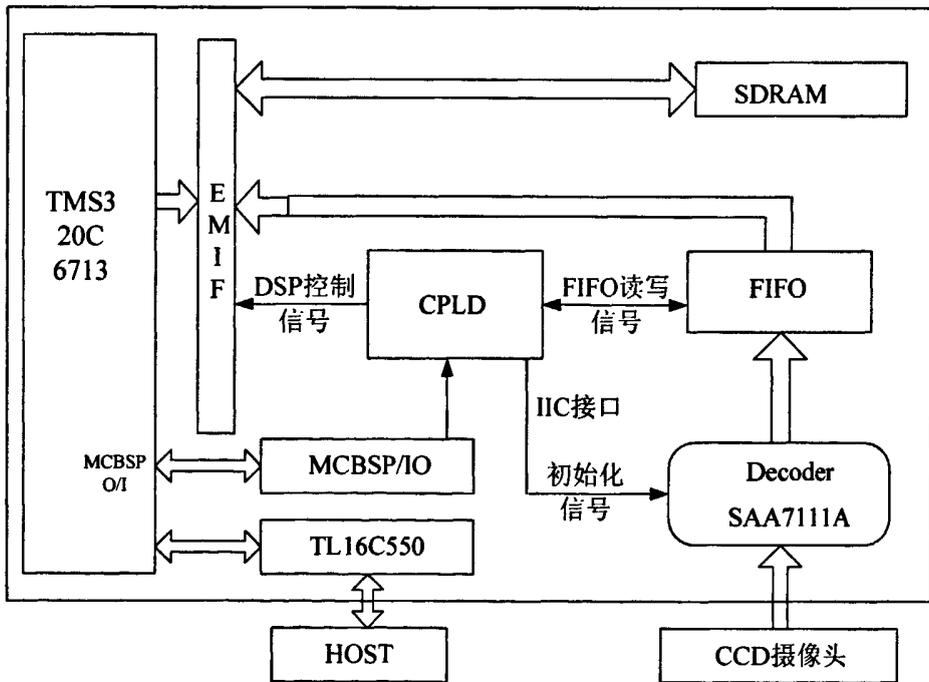


图 2.2 系统总体结构框图

系统各部分功能如下：

- 1、DSP 图像处理电路：该部分实现了整个系统图像采集的流程控制，也是实现图像处理算法的硬件平台。包括 DSP 的最小系统及其外围

- 接口电路。DSP 最小系统包括电源、复位电路、时钟电路、EMIF 外部存储器和 JTAG 仿真接口。
- 2、视频录入设备：使用 CCD 摄像头作为图像采集设备，摄像头输出的模拟信号为亮度信号和同步信号的混合信号，其中亮度信号为图像信号的原始输入；同步信号被视频解码芯片分离后，用于控制图像数据的采集。
 - 3、视频输入处理：该部分采用集成视频解码芯片 SAA7111A，将模拟视频信号转换成数字视频信号，同时输出从模拟视频信号中分离出同步信号和由芯片内部的时钟发生器产生的像素时钟。视频输入处理芯片由 DSP 通过 IIC 接口完成初始化操作。
 - 4、数据缓存 FIFO：该部分作用是完成图像数据的暂存，解决 DSP 与视频解码芯片之间因为速度差异引起的时序匹配问题。
 - 5、数据存储：该部分用于存放图像数据，主要是原始图像数据和图像处理过程中产生的中间数据。由于图像数据量比较大，所以选用容量较大和成本较低的 SDRAM。此 SDRAM 通过 EMIF 总线挂接到 DSP 的 CE0 空间。
 - 6、程序存储：该部分采用 FLASH，用于固化 DSP 的应用程序。FLASH 被分配到了 DSP 的 CE1 空间。
 - 7、时序与逻辑控制部分：该部分主要实现以下几个功能，DSP 寻址空间的译码，根据要求控制采集图像的窗口，根据视频处理芯片输出的同步信号和像素时钟信号生成控制图像数据存储的信号。
 - 8、通讯接口：该部分用于向上位 PC 机传送图像数据或是图像处理结果，采用串行总线接口芯片 TL16C550 实现系统与 PC 机之间的串口通讯。

在实际的硬件结构上，系统由两块电路板组成，如图 2.3 所示。上层为图像采集子板，完成图像转换和采集；下层为图像采集基板主要完成图像的搬运、存储、处理和传输，这样便于系统的调试和扩展。以上两部分将在下面两章中逐一介绍。

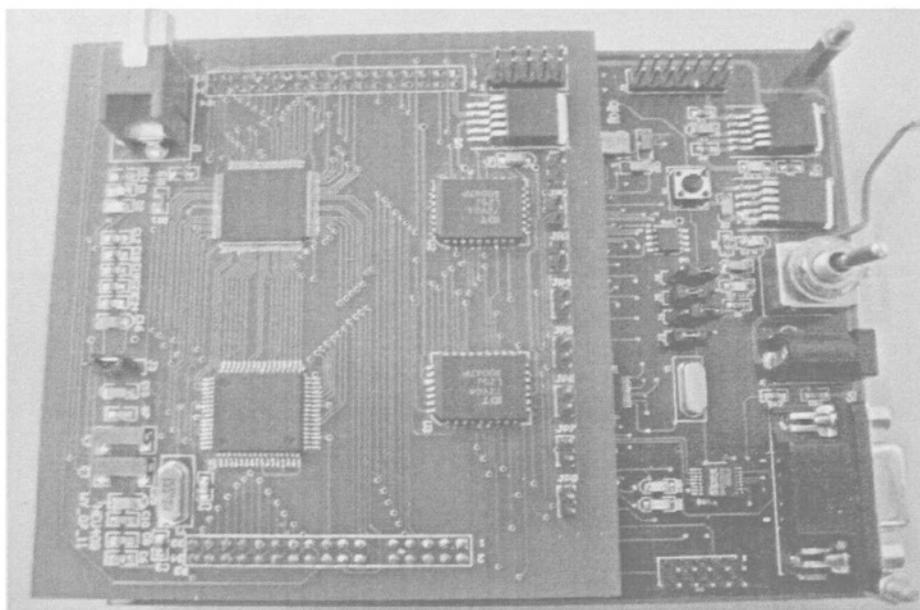


图 2.3 系统实物图

2.1.2 系统的工作流程

当系统执行时,物体会将光线反射到摄像头,CCD 摄像头摄取视频图像,输出标准 PAL 制式的模拟视频信号。视频解码器 SAA7111A 接收模拟视频信号,并将其转换为标准格式的数字视频数据流,转换好的数字图像数据由 SAA7111A 输出,图像数据由 CPLD 控制视频解码器将数据采集到高速缓存 (IDT72V04) 中暂存,从而避免了数字视频数据长时间占用 DSP 外部总线。暂存到高速缓冲器中的图像数据由 DSP 的 EDMA 控制器通过 DMA 方式从高速缓存中将图像数据搬运到 SDRAM 中,采集到的图像数据经过 DSP 图像处理,通过串口将图像数据传送给上位 PC 机。PC 机负责显示图像或显示处理结果。

2.2 TMS320C6713 DSP 介绍

2.2.1 DSP 选型

选择合适的 DSP 所应考虑的主要方面有^[4]:

- 1、定点与浮点 DSP。在选用 DSP 时首先考虑是选用定点 DSP 还是浮点 DSP。现有 DSP 以运算数据分为两类:一类是字长为 16/24 位的定

点 DSP, 另一类是字长为 32 位的浮点 DSP。这两类相比: 第一, 浮点 DSP 运算精度高, 动态范围大。定点 DSP 动态范围小, 运算时为防止溢出, 需经常定标, 给编程造成不便。第二, 浮点 DSP 地址总线比定点的宽, 更有利于大数据量信号处理。第三, 浮点 DSP 结构复杂, 单周期内完成功能比定点 DSP 多。第四, 从功耗、体积和成本上看, 定点 DSP 功耗小、体积小、价格便宜适用于小型的不持续供电的设备。

- 2、速度的要求。衡量速度的指标为 MOPS—每秒百万次操作或 PIPS—每秒百万次指令。
- 3、性能指标。主要是指指令执行速度(MIPS)。实时图像处理要求 DSP 处理系统必须在限定的时间内完成任务, 或者在允许的输出一输入相应迟延范围内, 系统的数据输入/输出吞吐率必须达到一定速度。本系统使用 PAL 制式, 每秒 50 场, 因此要求能够在 20ms 内完成对一场图像的处理。
- 4、厂商的技术支持。DSP 本身比较复杂, 如果厂商能够提供完善的开发调试工具, 无疑将大大减少产品的开发时间和开发难度。
- 5、其它性能指标。DSP 的总线结构、封装形式是否符合要求。价格成本。不单指芯片本身价格, 还包括必须的外部配套器件成本。

基于上述几点考虑, 并且考虑 DSP 芯片焊接问题。本系统选用了 TI 公司 C6000 系列中的 TMS320C6713 DSP 作为实时图像处理的核心芯片。

2.2.2 TMS320C6713 简介

C6000 是美国 TI 公司于 1997 年推出的 DSP 芯片^[4]。C6000 片内有 8 个并行的处理单元, 分为相同的两组。DSP 的体系结构采用超长指令字(VLIW)结构, 单指令长为 32 位, 指令里有 8 个指令, 总字长达到 256 位。执行指令的功能单元已经在编译时分配好, 程序运行时通过专门的指令分配模块, 可以将每个 256 位的指令包同时分配到 8 个处理单元, 并由 8 个单元同时运行。

图 2.4 是 TMS320C6713 结构框图, 芯片内部集成了一定大小的片内程序存储器。外设包括直接存储器访问(DMA)、低功耗逻辑、外部存储器接口(EMIF)、多通道缓冲串口、扩展总线、主机口和定时器等。

运算单元共有两组，每组各有四个：

D: 存取外界寄存器数据；

S: 与控制寄存器交换数据；

L: 逻辑运算单元；

M: 16 位乘法器。

通用寄存器也分为 A、B 两组，每组各有 16 个(A0-A15、B0-B15)；

Instruction Fetch、Dispatch、Decode 负责程序的采集，指令的分配及数据路径规划；控制寄存器和控制电路负责 CPU 与其它 DSP 的沟通与设定。

C6713 DSP 除了有一个 32-bit CPU 之外，还有许多外围器件，提供 DSP 与其它接口的连接，包括：

- 1、片内存储器(Internal memory): 分为程序区间和数据区间，采用 2 级存储器结构，第 1 级存储器包括相互独立的程序 cache(L1P)和数据 cache(L1D)，只能作为高速缓存被 CPU 访问。第 2 级存储器(L2)是一个统一的程序/数据空间，可以整体作为 SRAM 映射到存储空间，也可以整体作为第 2 级 cache。C67nx 的片内存储器结构为 4KB 的 L1 Program Cache、4KB 的 L1 Data Cache 以及 64KB 的 L2 Cache。
- 2、外部存储器接口(External Memory Interface, EMIF): EMIF 可连接各种不同类型的内存(如: SDRAM、Flash、Asynchronous Memory)及 I/O 器件，C6713 DSP 的 EMIF 接口可寻址的空间为 512MB。
- 3、增强型直接存储器访问 EDMA(Enhanced Direct Memory Access): EDMA 可在不打扰 CPU 的状态下，进行数据区块的搬移；在 C6713 DSP 中共有 16 个 EDMA 通道。
- 4、主机口(Host Port Interface, HPI): 提供外界主计算机与 DSP 进行数据交换或控制。
- 5、多通道缓冲串口(Multi channel Buffered Serial Port, McBSP): 可进行一般串行数据的传送与接收，可以连接其它通信标准界面，实现多通道的数据传送与接收。
- 6、IIC(Inter-Integrated Circuit): 利用两条信号线(时钟、数据)进行主从式(Master-Slave)串行传输，并且可由多个器件共用相同总线，可由 Master 进行控制和数据存取。

- 7、GPIO(General-Purpose Input/Output): 一般用途的输入输出口。
- 8、定时器(Timer): 片内集成了 32-bit 的通用定时器, 可以用来计时、计数、产生同步信号等。

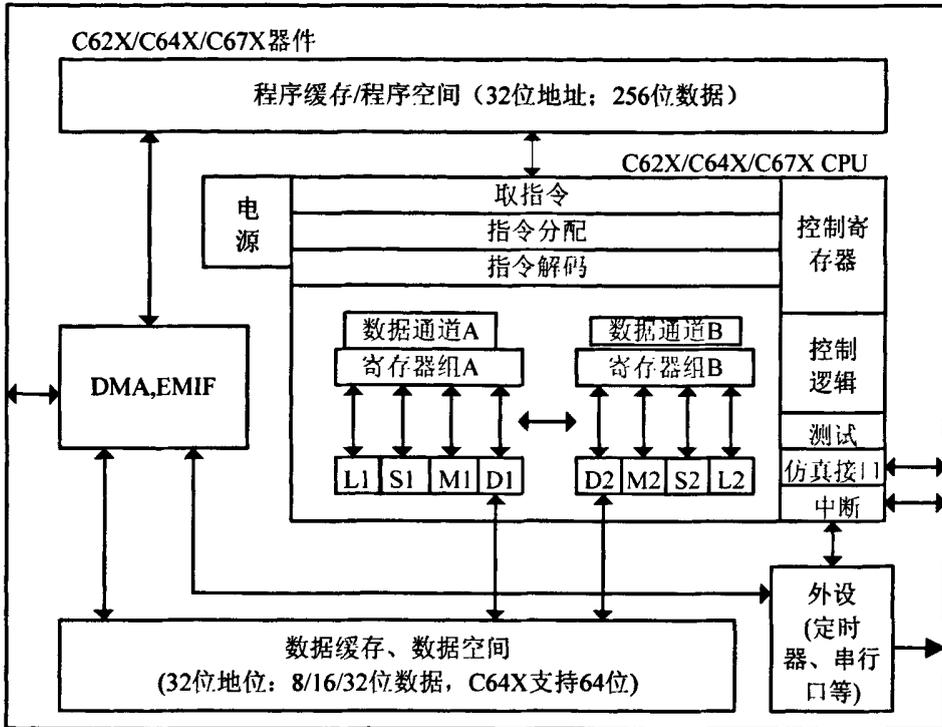


图 2.4 TMS320C6713 结构框图

2.3 本章小结

本章介绍了系统的工作原理、流程及总体架构。为了实现图像处理的高速度和高精度, 在系统的设计中选用了 32 位的浮点处理器 TMS320C6713 DSP 作为核心处理器。文章并对 DSP 的一些性能特点进行了描述。

第3章 图像采集子板设计

3.1 图像采集子板的基本构成

在系统中子板主要完成图像采集的功能。子板的设计包含图像传感器(CCD图像传感器)、图像缓冲器(Frame Buffer)及其控制电路。

图3.1是图像采集子板的原理组成框图。系统设计包含两个图像缓冲器，它们的控制信号通过跳线连接。跳线连接后可实现缓冲器的位扩展，从原来的8位扩展为16位。其中一个缓冲器输入信号为Y信号即图像的灰度信号；另一缓冲器输入信号为UV信号即图像的色差信号。这样把图像的YUV信号都传输给DSP，YUV信号通过DSP的处理后，系统可显示彩色图像。当跳线断开，系统只将Y信号传送到DSP，则可获得灰度图像。

从图3.1中可看到图像传感器是由CPLD所控制，而缓冲器的写入控制信号由传感器提供，读取控制信号则由DSP控制；因此将图像采集子板的硬件设计分为传感器控制电路、图像缓冲器控制电路和逻辑控制模块三部分，以下详细介绍这三部分电路的设计。

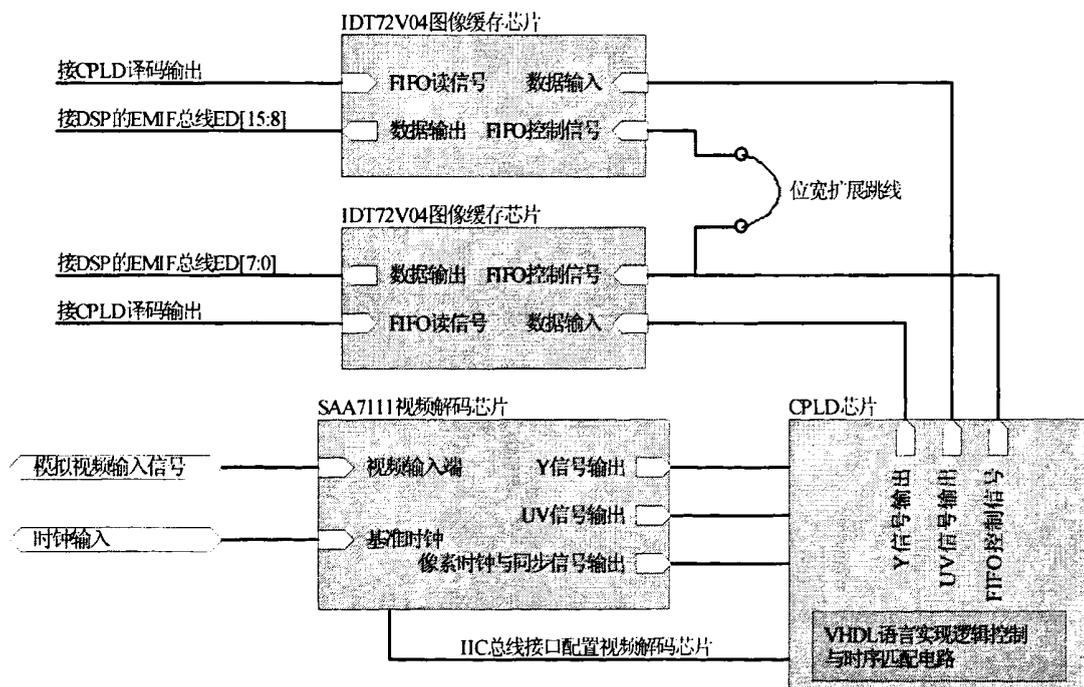


图 3.1 图像采集子板原理组成框图

3.2 视频输入电路的选择和设计

3.2.1 视频解码芯片的选取

TV 解码器可选用 PHILIPS 公司的 SAA7110 / 7111A / 7112 / 7113 / 7114 / 7115 / 7118, SAMSUNG 公司的 KS0127(S5D0127X01)或 ITT 公司的 VPC3211B, 它们可通过 IIC 总线接口控制, 自动识别输入模拟视频信号格式, 然后解码输出 24bit / 16bit 的 RGB 或 YUV 数字信号和 Hsync、Vsync、HAV(HREF)控制信号及采样时钟, 解码后的信号可直接送到视频转换模块或视频采集系统。视频输入处理芯片内部 A/D 转换器的位数越高, 芯片的价格就越昂贵。此外, 芯片对特殊总线接口的支持也使得芯片的价格差异非常大。所以芯片的选取的原则是在满足系统要求的前提下, 使用 A/D 位数少的处理芯片, 尽量减少芯片功能资源的冗余。根据 AVERLogic 公司发布的技术文档。PHILIPS 公司的解码器解码效果最好, 其次是 VPC3211B, KS0127 虽然功能强大价格便宜, 但是解码效果稍逊一些, 而且该芯片的最大缺点是功耗太大(将近 1W)。PHILIPS 公司的 SAA7111A 能够满足系统对采集 256 级灰度图像的要求, 并且在价格上相比其它公司的产品占有很大的优势^[5]。不同公司根据应用场合的不同, 会为其视频输入处理芯片集成不同功能和不同精度的资源。表 3.1 中为不同公司的视频输入处理芯片的性能比较。

表 3.1 几种视频处理芯片的性能比较

器件名称	生产厂商	模拟通道数	A/D位数	输出数据宽度	控制总线	其它总线接口支持
SAA7111	PHILIPS	4	8	16	IIC	无
SAA7110	PHILIPS	6	8	16	IIC	无
ADV7185	AD	6	10	20	IIC	无
TVP5020	TI	4	8	16	IIC	无
BT849	Brooktree	3	12	24	IIC	PCI

本方案采用 SAA7111A, 它采用 CMOS 工艺, 功耗小(小于 0.5W), 电压低(3.3V), 体积小, 温度适应范围广(0°C ~ 70°C)。

SAA7111A 芯片特点:

- 1、自动进行 50 / 60Hz 场频的检测，支持 NTSC 和 PAL 制自动转换，可对 NTSC-M / N / 4.43 / JAPAN、PAL-M / N / BGHI、SECAM 格式视频信号的亮度和色度进行处理。
- 2、拥有 4 路模拟输入：4 路复合视频(CVBS)或 2 路 YC 或 1 路 YC 和 2 路 CVBS。
- 3、可设置 CVBS 或 YC 通道为静态增益控制或自动增益控制(AGC)。
- 4、拥有 2 路亮度和色度梳状滤波器。可对亮度、对比度、光圈和饱和度进行控制。
- 5、行(HREF)、场(VREF)同步、奇偶帧(RES1)和像素时钟(LLC)信号均可由管脚引出。
- 6、可支持以下输出格式：4:2:2(16 位)、4:2:2(CCIR601 8 位)、4:1:1(12 位)YUV 格式或 8:8:8(24 位)、5:6:5(16 位)RGB 格式。
- 7、具有符合 IEEE1149.1 标准扫描逻辑的边界扫描测试电路。
- 8、64 引脚的 LQFP 封装(尺寸:10mm×10mm×1.4mm),功耗小于 0.5W。

SAA7111A的内部功能方框图如图3.2所示。图中，从SAA7111A的四个模拟输入端AI11、AI12、AI21、AI22的某一引脚输入的视频图像信号经模拟处理后。一路可通过缓冲器输出到AOUT端用于监视，另一路经MD转换器后则产生数字色度信号和亮度信号。在分别进行亮度信号处理和色度信号处理后，其亮度信号处理结果的一路将送到色度信号处理器进行综合处理，产生的Y和UV信号经格式化后从VPO(16位)输出；另一路则进入同步分离器，并经数字PLL产生相应的行和场同步信号HS和VS。同时，PLL将驱动时钟发生器，以产生HS锁定的时钟信号LLC和LLC2。SAA7111A的所有功能均是在IIC总线控制下完成的，其中SCL为串行时钟，SDA为串行数据信号。

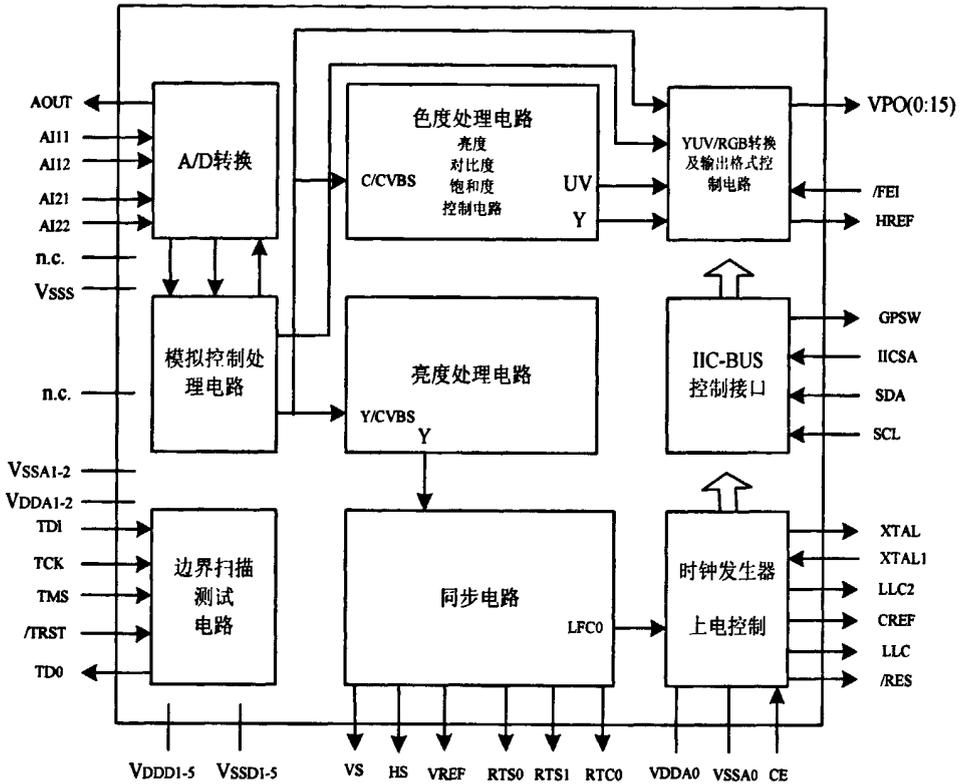


图 3.2 SAA7111A 内部结构

3.2.2 视频输入电路设计

视频信号输入端一般需要使用 27Ω 电阻分压，然后用 22nF 的电容直流耦合。具体电路见图 3.3。SAA7111A 的片选信号管脚 CE 在串接一个限流电阻后接 3.3V 电源，使芯片在上电后一直保持工作状态。在 XTAL 引脚和 XTAL1 引脚间连接一个频率为 24.576MHz 的石英晶体振荡器，用于产生系统工作时钟，再通过该时钟驱动产生行锁定时钟 LLC(27MHz)、LLC2(13.5MHz)和时钟参考信号 CREF(13.5MHz ，相对 LLC2 有一定的相位延时)。

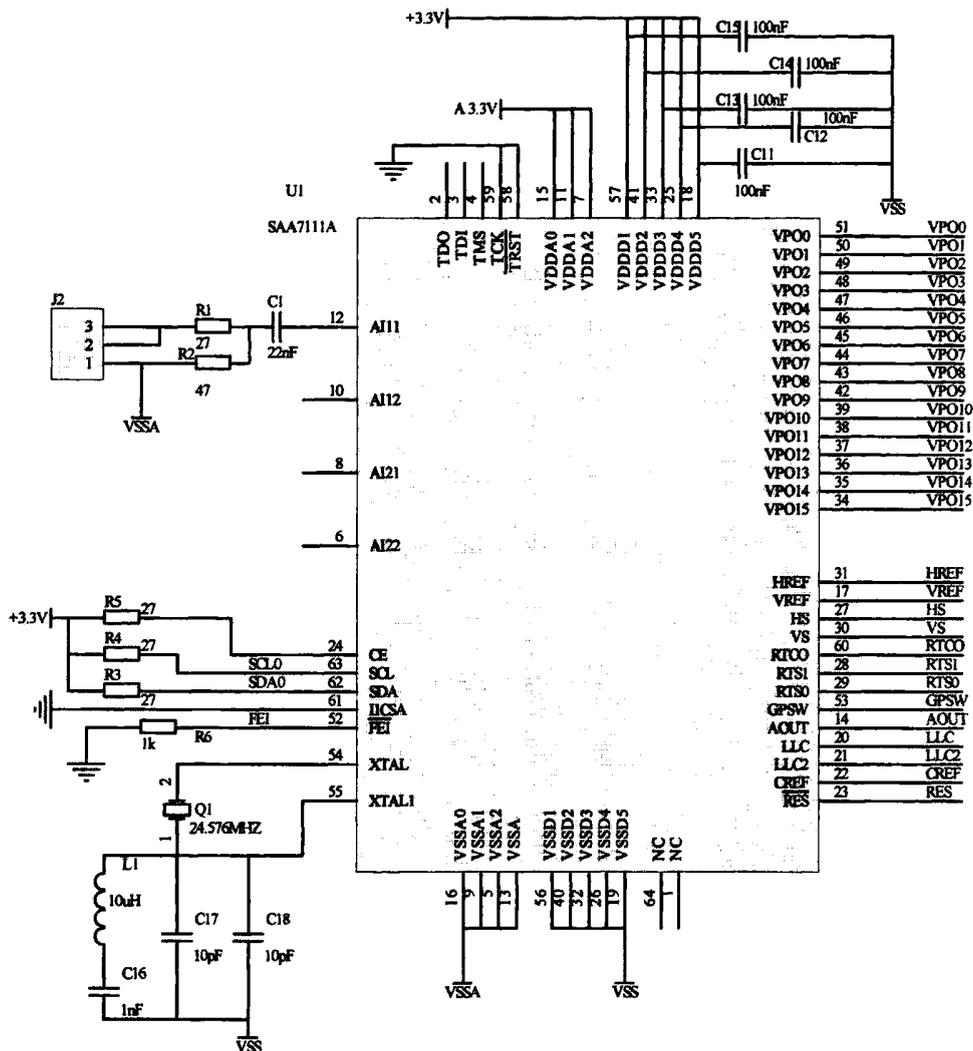


图 3.3 视频输入电路设计

3.2.3 SAA7111A 的初始化设计

SAA7111A 芯片的工作方式通过 IIC 总线进行配置的，IIC 总线控制器通过对 SAA7111A 内部 32 个寄存器的配置来控制芯片内部各个功能模块的工作情况，这 32 个寄存器的地址从 00H 依次到 1FH，对 SAA7111A 寄存器的配置是系统设计中重要的一环^[6]，所以将先介绍 IIC 通讯协议。

3.2.3.1 IIC 通讯协议

这种总线类型是由飞利浦半导体公司在八十年代初设计出来的，主要

是用来连接整体电路，IIC 是一种多向控制总线^[7]，也就是说多个芯片可以连接到同一总线结构下，同时每个芯片都可以作为实施数据传输的控制源。这种方式简化了信号传输总线。IIC 串行总线一般有两根信号线，一根是双向的数据线 SDA，另一根是时钟线 SCL。所有接到 IIC 总线设备上的串行数据 SDA 都接到总线的 SDA 上，各设备的时钟线 SCL 接到总线的 SCL 上^[8]。典型的 IIC 总线结构如图 3.4 所示。

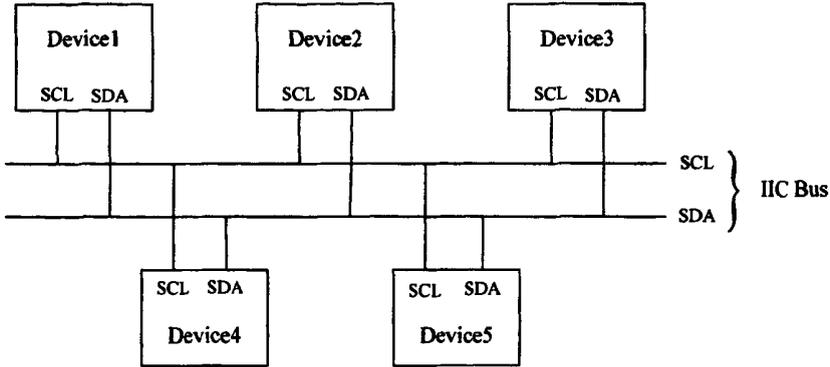


图 3.4 IIC 串行总线硬件连接

3.2.3.2 IIC 总线时序

IIC 动作共有四种模式：开始(Start)、读/写(R/W)、确认(Ack)、停止(stop)，一个完整的 IIC 动作必须包含上述四种模式，器件地址以及寄存器地址宣告，图 3.5 为一完整的 IIC 动作时序。

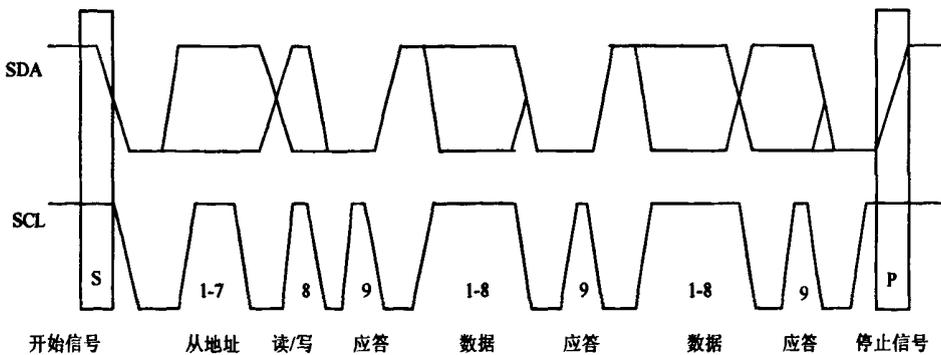


图 3.5 一次完整的数据传输

开始(Start): Master 必须在 IIC Bus 上发出一个开始信号，以取得主控 IIC Bus 的控制权。在没有任何动作的时候，SCL 与 SDA 都为高电位，Master: 必须先将 SDA 转为低电位，经过至少在 $4\mu\text{s}$ 之后，再将 SCL 转为低电位，

这样的动作便是一个开始信号。

器件地址宣告：一个器件地址为 7-bits，由高位依次送出到 IIC Bus 上，符合此器件地址的器件便会被 Master 控制。

读/写(R/W)：与器件地址(7-bits)合为一个 byte。读写控制位于字节高位，代表 Master 要对此 Slave 进行读或写的动作。若此位为高电位，则进行读的动作，若为低电位，则进行写的动作。

确认(ACK)：在进行完器件地址与读写应答、缓存器地址应答与数据读写后，都必须有确认的动作，以确保 Master 与 Slave 的数据传输是正确的；作为 Master 将 SDA 释放为高电位，若传输正确，则由 Slave 将 SDA 拉至低电位，若数据传输不正确则 Slave 不动作，因此 SDA 会维持在高电位。

寄存器地址宣告：一次为 8-bits，由最高位开始传送，结束后必须有确认动作。

数据读写：也是以 8-bits 为一个读写单位，一样由最高位开始，结束后也要有确认动作。

停止(Stop)：其动作过程和开始是相反的，Master 在完成动作以后，将 SCL 释放至高电位，在至少 4 μ s 以后，再将 SDA 释放至高电位。当完成停止动作后，IIC Bus 便空闲，等待下一个开始信号的产生。

3.2.3.3 SAA7111A 的初始化实现

虽然 DSP 有 IIC 接口，但是由于希望本图像采集子板能适用于各种不同的 CPU 或控制器，因此采用了较为普遍的 GPIO (General Purpose I/O) 来控制 SAA7111A。

系统用 McBSP 接口工作在 GPIO 的方式下，用 GPIO 来模拟 IIC 总线接口的时序来对视频解码芯片进行配置。首先，通过配置串口控制寄存器 SPCR1 和 SPCR2 以及引脚控制寄存器 PCR 的禁用 McBSP 功能，将 McBSP 引脚(包括 CLKX、CLKR、DX、DR、FSX、FSR 和 CLKS)作为通用 I/O 口^[9]。以发送器为例，当 SPCR2 的 XRST=0 且 PCR 的 XIOEN=1 时，串口发送器无效，FSX 和 CLKX 用作通用 I/O 引脚。FSX 和 CLKX 作为通用 I/O 端口的引脚设置情况如表 3.2 所列。以 McBSP0 为例，也可以从 FSX0 和 CLKX0 引出两条线分别表示 SDA 线和 SCL 线。

SAA7111A内部有32个寄存器(Sub address 00~1FH)，其中22个是可编程的。00H、1A~1CH、1FH是只读寄存器，其中00H描述的是芯片版本信息；1A~1CH是文本信息检测和解码寄存器，一般很少用到；1FH用来描述芯片的状态。02H~12H是可读写寄存器，其中02H~05H是模拟输入控制寄存器，02H用于设置模拟视频信号输入方式(共8种)，03H~05H用于设置增益控制方式，06H~12H主要用于设置解码方式，通过配置这些寄存器可以置行同步信号的开始和结束位置，并可确定亮度、色度和饱和度的大小以及输出图像数据信号的格式。01H、13H~19H、1DH~1EH寄存器保留使用。需要注意的是，在读00H寄存器前，必须将它初始化为0。在对多个连续的寄存器进行操作时，寄存器地址有自动加1功能。

表3.2 McBSP的管脚作为GPIO用法

管脚	设置方法	作为输出	输出值来自于	作为输入	输入值来自于
CLKX	/XRST=0 XIOEN=1	CLKXM=1	CLKXP	CLKXM=0	CLKXP
FSX	/XRST=0 XIOEN=1	FSXM=1	FSXP	FSXM=0	FSXP
DX	/XRST=0 XIOEN=1	Always	DX_STAT	-	-
CLKR	/RRST=0 RIOEN=1	CLKRM=1	CLKRP	CLKRM=0	CLKRP
FSR	/RRST=0 RIOEN=1	FSRM=1	FSRP	FSRM=0	FSRP
DR	/RRST=0 RIOEN=1	-	-	Always	DR_STAT
CLKS	/RRST=0 RIOEN=1	-	-	Always	CLKS_STAT

SAA7111A的写地址为48H。这里把SAA7111A初始化设定为：一路模拟视频信号输入(AI11)、自动增益控制、625行50Hz PAL制式、YUV 4:2:2(16bit)数字视频信号输出、设置默认的图像对比度、亮度及饱和度。相应的寄存器初始化值如表3.3所列。下面是向SAA7111A的19个连续的子地址寄存器(00H~12H)写入一组数据的程序。图3.6为初始化时读写寄存器的流程图。

表3.3 寄存器初始化值

Sub Address	Data	Sub Address	Data
00H	00H	01H	00H
02H	C0H	03H	23H
04H	00H	05H	00H
06H	EBH	07H	E0H
08H	88H	09H	01H
0AH	80H	0BH	47H
0CH	40H	0DH	00H
0EH	01H	0FH	00H
10H	48H	11H	0CH
12H	01H		

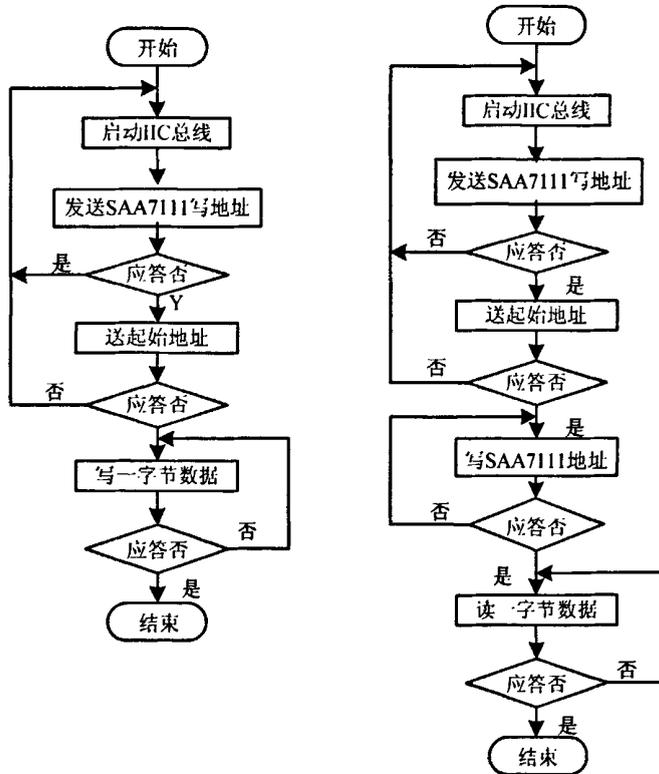


图3.6 IIC初始化写/读流程

3.3 图像缓存存储器的设计

高速视频图像采集处理系统强调的是实时性，而满足实时性的关键就是

要解决采集端与处理端的速度匹配问题，解决这个问题常用的方法就是在高速与慢速器件之间加入数据缓冲池，低速器件采集到的数据按照一定的方式缓冲到数据池中，待达到一定的数据量，通知CPU启动一次数据传输，将数据从缓冲池中取走。常用的可作为数据缓冲的器件有双口RAM、FIFO等。相对来说，采用FIFO比双口RAM在硬件软件设计上要简化得多而且FIFO方案也更适合视频数据的时序特性。因此本系统采用FIFO构建了数据通道的缓冲池，它的前端与A/D相连，采集到的视频数据在A/D的同步时钟驱动下写入FIFO，后端与DSP的扩展总线连接，当FIFO中的缓存数据量达到临界值后，使DSP启动一次EDMA传输，将缓存数据读入RAM^{[10][11]}。在传输过程中它不占用CPU资源，而且利用了中断方式触发CPU读取数据，使整个系统的运算速度得到提高。

3.3.1 FIFO 存储器的选择

FIFO存储器的参数指标直接影响数据的采集速度。首先，FIFO存储器的读写速度要足够快。其次，FIFO存储器的存储容量要适宜。如果容量过大会造成资源浪费；如果容量过小会造成数据采集速度过慢。经过估算，最终选用IDT公司的IDT72V04容量为4KB，其转换频率最大为28.5MHz，读周期时间最小为25ns，完全能够满足DSP的需要。如果对FIFO存储器进行容量扩展，采集写入数据的速度还能够进一步提高。

3.3.2 图像缓存芯片简介

FIFO是一种先进先出的结构，它有如下特点：

- 1、无地址线，只需提供读写和其他控制信号线，布线简单；
- 2、它不可以对任意地址单元操作，实现随机存取，只能是顺序存取；
- 3、只能实现块操作，其读出数据顺序和写入顺序相同。

表3.4 FIFO缓存器的相关引脚及功能

引脚	功能
D0-D7	/WE允许时，数据通过此引脚写入
Q0-Q7	在/RE和/OE允许时，数据通过此引脚读出
/W	控制写数据的允许与否，在/W信号上升沿，数据写入
/R	控制写数据的允许与否，在/R信号上升沿，数据读出

/EF	空标志, 抑制进一步读控制
/HF	半满标志
/FF	全满标志, 抑制进一步写控制
/XI	芯片位或深度扩展信号
/RS	复位信号, 复位时, 读写指针将移到起始位
VCC	电源, 3.3V
GND	地

FIFO读写一次的过程如图3.7所示。首先, 复位FIFO, 使FIFO的读写指针指向内部零地址。当DSP的采集开关打开后, SAA7111A产生的写信号会使FIFO的写指针增加, 当FIFO半满时, 其半满标志 \overline{HF} 产生DSP中断, DSP在中断服务程序完成2K数据的读操作。一次读操作结束, 中断返回, 继续等待中断^[12]。

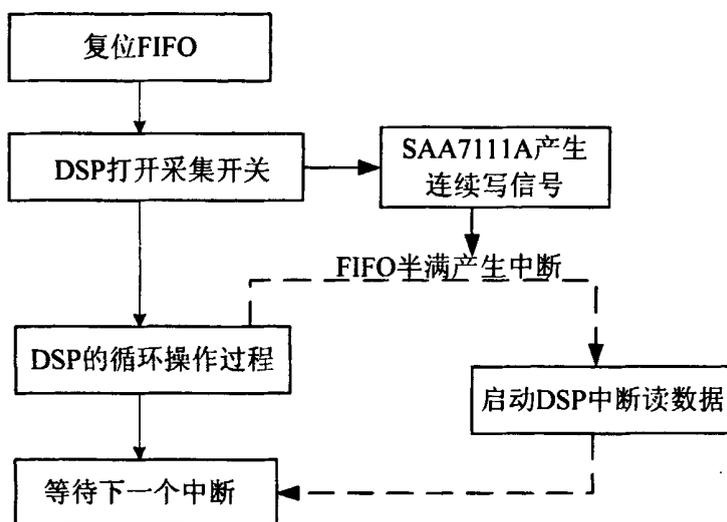


图3.7 FIFO读写过程

3.3.3 视频数据缓存电路的设计

视频解码芯片输出的数据要实时送入FIFO中, 因此两者的写时钟频率必须一致, 所以本系统将视频解码芯片的时钟与FIFO的写信号连接。图3.8为视频解码芯片的时钟像素时序图, 视频解芯片的每个时钟输出一个像素数据。图3.9为FIFO读/写时序图, 在写信号的上升沿数据写入FIFO。将视频解码芯片的像素时钟LL2与FIFO的写信号连接, 可将视频数据实时送入FIFO中。

FIFO可以与DSP的无缝连接, 这样的无缝连接要求FIFO作为一个输入缓

冲器。而当FIFO作为输入缓冲器时，FIFO是EMIF的唯一异步设备。设计中恰好让FIFO作为输入缓冲器，成为挂接到CE2空间的唯一的一个异步设备，因而避免了复杂的接口逻辑。

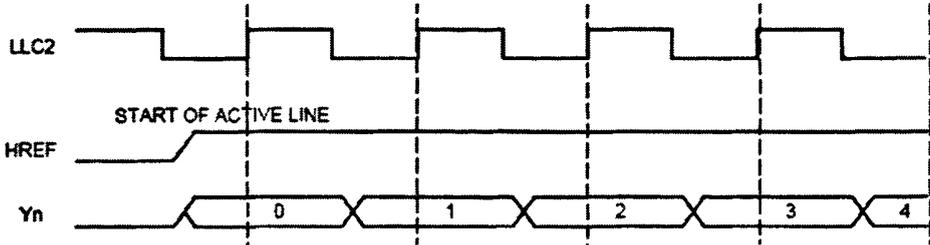


图3.8 SAA7111的时序

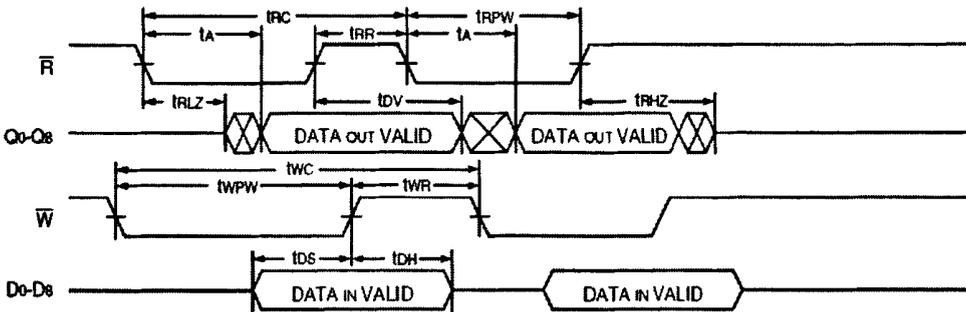


图3.9 FIFO读写时序

图3.10为FIFO半满时序图。FIFO的半满标志 \overline{HF} 作为DSP的中断输入信号。 \overline{HF} 用于告诉DSP此时FIFO中的有效数据是可以读取的(而且容量是FIFO总大小的一半)，以便DSP产生一个中断来读取FIFO中的数据块。

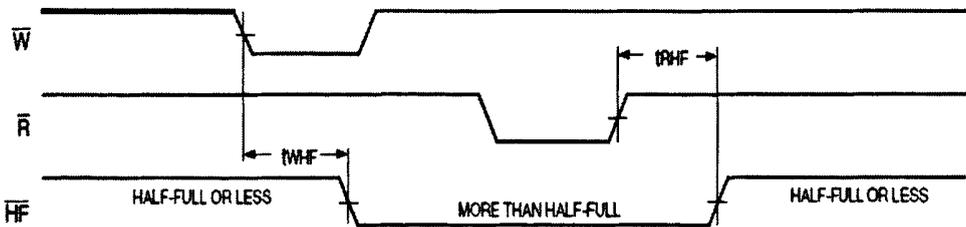


图3.10 FIFO半满时序

时序设定IDT72V04的写入时间(由 \overline{W} 信号控制)不能小于读取时间(由 \overline{R} 信号控制)，且写入时必须考虑到SAA7111A输出速度，因此利用SAA7111A的CLK控制FIFO的 \overline{W} 。SAA7111A的CLK频率为13.5MHz，所以每个图像数据从SAA7111A写入FIFO的时间为 $0.074\mu s$ 。图3.11为DSP读FIFO中数据的时序图，系统中在初始化CE空间控制寄存器(CECTL)时，设定建立时间为2个时钟周期，选通时间为3个时钟周期，保持时间为1个时钟周期；DSP将每个

数据放到SDRAM中用一个时钟周期,这样DSP从FIFO中读数据并将数据放入SDRAM中一共用7个时钟周期。由于本设计中CE空间的频率为125MHz,所以每个图像数据从FIFO搬运到SDRAM中用时为 $0.056\mu\text{s}$ 。FIFO写入时间 $0.074\mu\text{s}$ 大于读出时间 $0.056\mu\text{s}$,故满足FIFO工作要求。

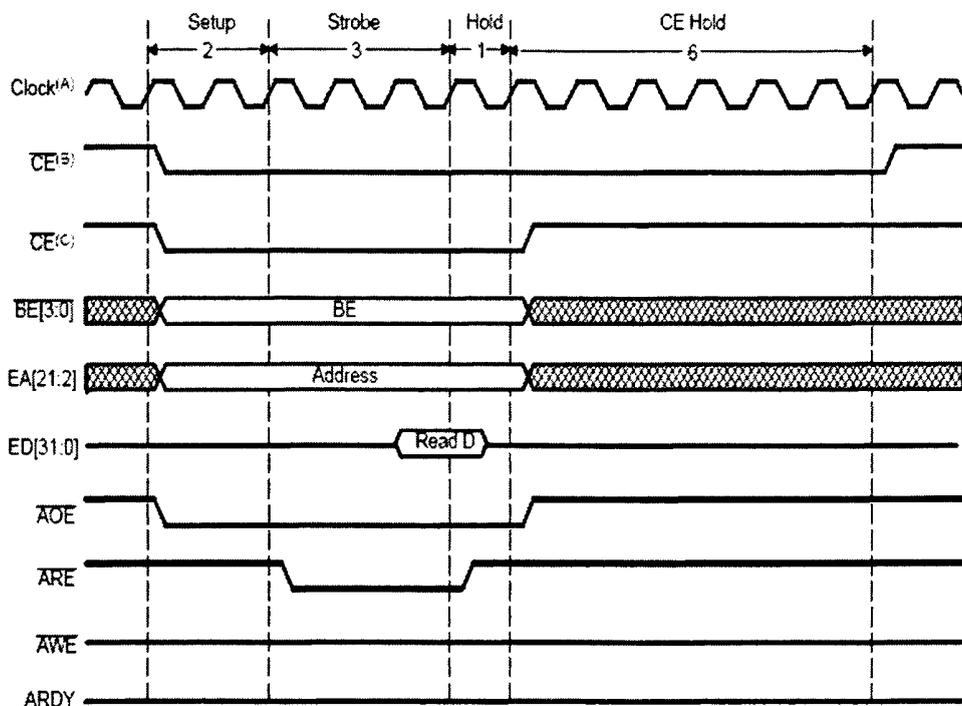


图3.11 DSP读FIFO的时序图

3.4 图像采集逻辑控制模块设计

3.4.1 可编程逻辑器件简介

集成电路的发展促进了电子设计自动化(EDA)技术的发展。先进的EDA技术已经从传统的“自下往上”的设计方法改变为“自上往下”的设计方法。ASIC(Application Specific Integrated Circuit)的设计与制造,已经不再完全由半导体厂商独立承担,系统设计师在实验室里就可以设计出合适的ASIC芯片,并且立即投入到实际使用中,这都得益于可编程逻辑器件PLD(Programmable Logic Device)的出现。现在运用最为广泛的是现场可编程门阵列FPGA,复杂可编程逻辑器件CPLD和可擦除可编程逻辑器件EPLD。

复杂可编程逻辑器件 CPLD 是在 20 世纪 80 年代末 Lattice 公司发明了在线可编程技术以后于 20 世纪 90 年代初出现的。CPLD 是在 EPLD 的基础上发展起来的,采用 E2CMOS 工艺制作,与 EPLD 相比,增加了内部连线,对逻辑宏单元和 I/O 单元也有重大的改进。CPLD 至少包括三种结构:可编程逻辑宏单元、可编程 I/O 单元和可编程内部连线。部分 CPLD 器件内部还集成了 RAM、FIFO 或双口 RAM 等存储器,以适应 DSP 运用的设计要求。其典型器件有 Altera 的 MAX7000 系列,Xilinx 公司的 7000 和 9500 系列,Lattice 的 PLSI/isp LSI 系列和 AMD 的 MACH 系列^[14]。

3.4.2 CPLD 的选型

为了增加系统的集成度、可扩展性以使其具有更高的性能,从而达到小型化的目的,我们增加了一片 CPLD 来完成整个系统的逻辑控制,如控制数据采集、存储器片选、地址译码、数据锁存等功能^[15]。CPLD 的内部结构如图 3.12 所示。考虑到系统中需要 CPLD 控制的输入输出管脚的数目、内部逻辑资源的使用情况以及 Altera 公司 CPLD 芯片的高性能、高集成度、价格合理以及开发工具方便的特点,本系统选用了 MAX7000 系列中的 EPM7128AETC100。MAX7000 系列器件特点^{[16][17]}:

- 1、以第二代多阵列矩阵为基础的高性能 CMOS EPROM 器件;
- 2、逻辑密度为 600~5000 个可用门(器件上提供 1200~10000 个门)的全 EPLD 系列;
- 3、引脚到引脚的逻辑延时为 5ns,计数器工作频率为 178.6MHz;
- 4、可编程功率节省模式,使每个宏单元的功耗降低到 50%或更低;
- 5、可配置的扩展乘积项分配,允许向每个宏单元提供多达 32 个乘积项;
- 6、44 到 208 个引脚的各式封装;
- 7、3.3V 或 5.0V 电源;
- 8、可编程保密位,全面保护专利设计;
- 9、可编程触发器具有单独的清除、置位、时钟使能控制;
- 10、Altera Quartus II 开发系统提供软件设计支持;
- 11、Altera 主编程部件(MPU)或其它厂家的编程硬件支持器件的编程。

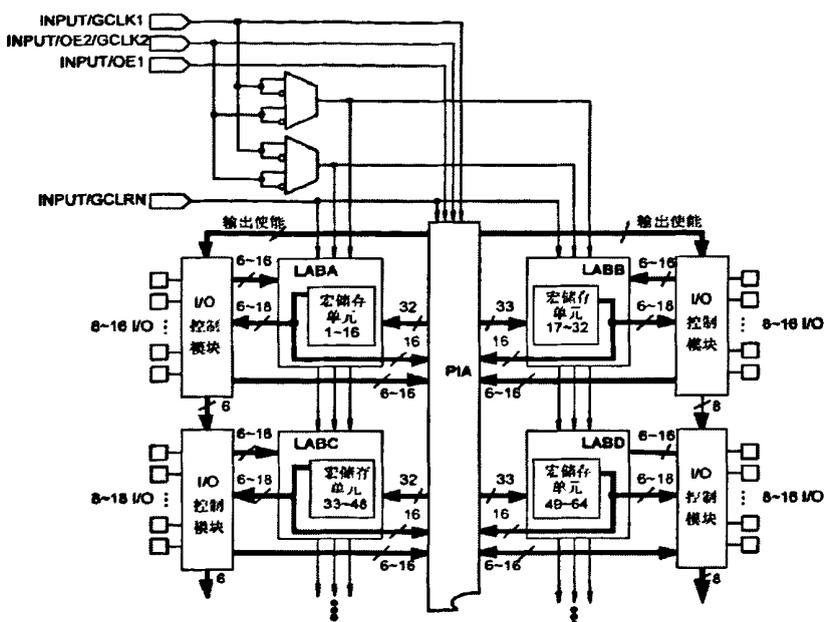


图3.12 CPLD内部结构

3.4.3 CPLD 的控制逻辑

系统采用CPLD控制视频解码器SAA7111A采集图像数据送到高速缓存。SAA7111A芯片提供了很多同步信号来实现系统工作同步，逻辑控制器对这些信号进行逻辑运算完成同步控制。

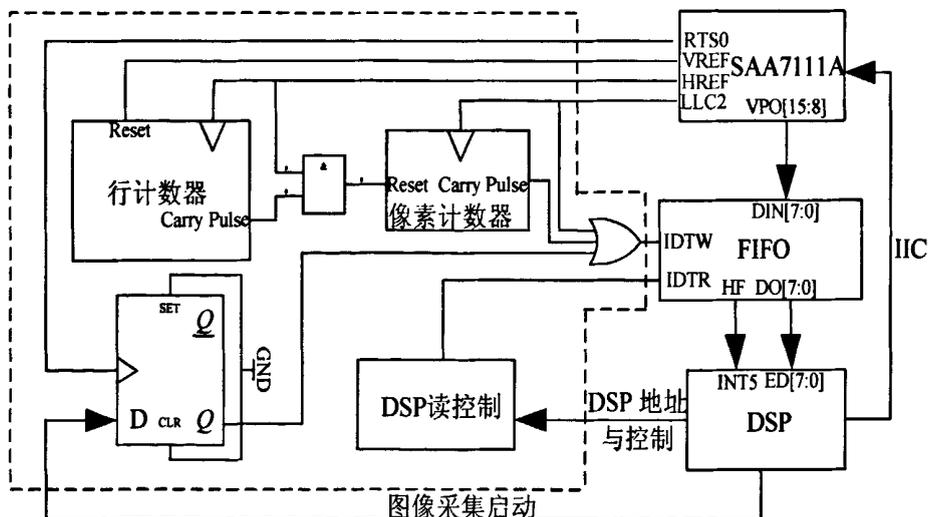


图3.13 控制图像采集逻辑图

图3.13为控制图像采集的逻辑框图。其中虚线框部分由CPLD完成。CPLD上电后，首先使D触发器输出为高电平，则控制FIFO写信号的或门中的一路信号为高电平，这样FIFO写信号禁止，从而关闭图像采集开关。行计数器和像素计数器处于等待计数状态。待DSP、SAA7111A、EDMA和FIFO初始化完成后，由DSP发出启动图像采集的信号。在VREF为高电平且HREF为我们所要行时，行计数器开始计数，达到512行时行计数器复位。同时，在HREF为高电平时且像素是我们想要的像素时，像素计数器开始计数并在达到512个像素时计数器复位。

经过SAA7111A解码得到的PAL制式的图像最大分辨率为720×576，要求采集的图像大小为512×512像素，只取其中一部分像素。SAA7111A的像素时钟LLC2频率为13.5MHz，作为CPLD的工作时钟，由SAA7111A的时序图可知，输出图像分为奇偶单场。在奇偶单场中的VREF高电平都对应有行有效，单场为288行(288个HREF)，输出的VREF的低电平表示场消隐信号，为25行(25个HREF)，由于单场为256行，要求采集图像为512行，则在单场有效行的前16行和后16行的图像数据不采集。此处设计行计数器用来达到取中间256行有效像素的目地。图3.14为采集一场图像的时序仿真图。同样，输出的HREF的高电平表示一行有效像素，为720个LLC2周期，每行要求只采集中间的512个像素。因此系统编写像素计数器，目地是在HREF上升沿出现后取中间的512个像素作为有用的像素，图3.15为采集一行图像的时序仿真图。

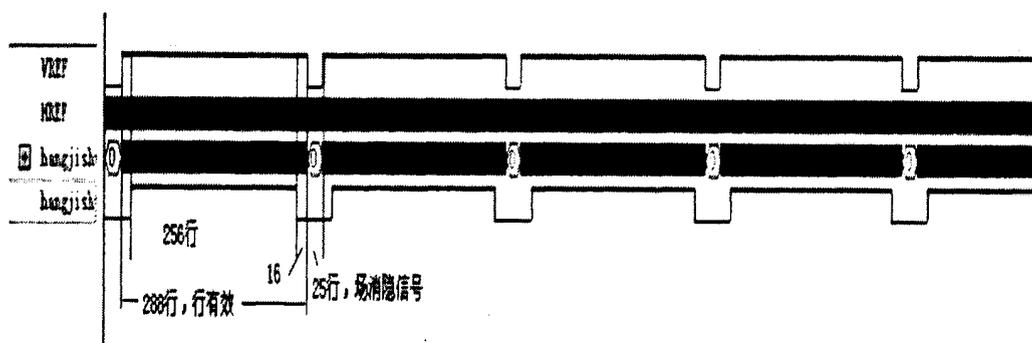


图3.14 采集一场图像的时序仿真

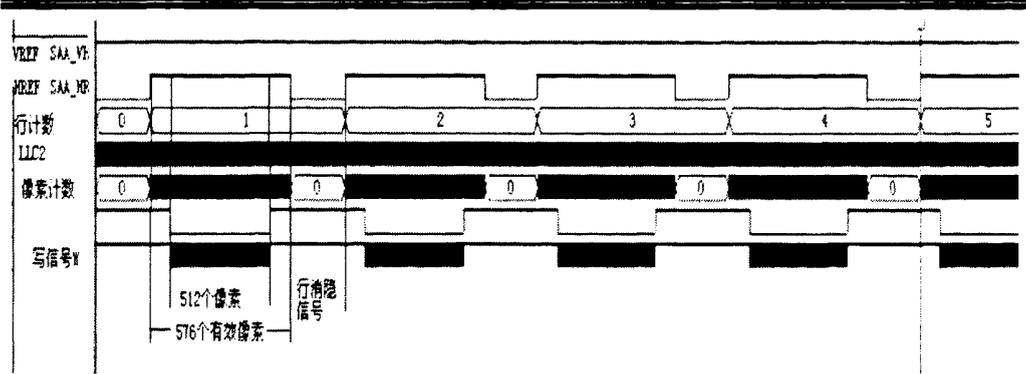


图3.15 采集一行图像的时序仿真

3.5 本章小结

本章首先介绍了图像采集子板的架构。然后对子板包含的图像传感器、图像缓冲器及其控制电路等部分进行了逐一介绍。又从模块功能的角度对全局逻辑电路设计的整体布局作了介绍，然后通过芯片性能对比阐述了复杂可编程逻辑器件的选择依据。最后详细的介绍了模块的具体实现的细节，并给出了仿真结果。

第 4 章 图像处理与传输系统的设计

图像处理与传输系统可分为核心处理器 DSP、图像存储模块和图像传输模块等部分。其主要功能是完成图像采集过程的系统管理，并为图像处理算法的实现提供硬件平台。具体实现上，该部分是一块单独的电路板，称为图像采集基板，其组成框图如下。

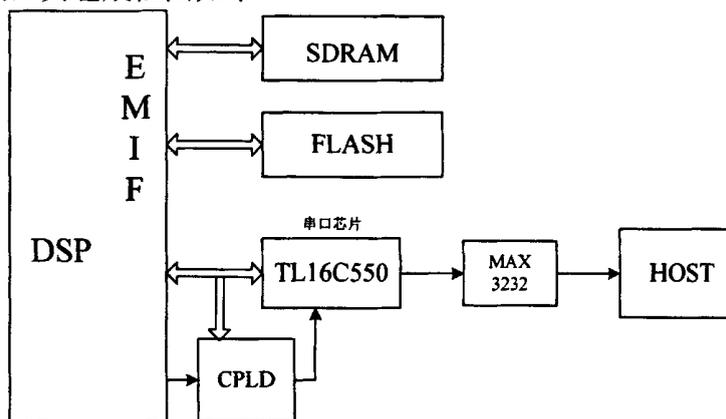


图 4.1 系统基板结构框图

TMS320C6713 DSP 内部集成了 256K 字节的 RAM，可以用于仿真模式下存储数据和程序。但是，一帧 512×512 灰度图像数据就占了 256K 字节的空间，显然 C6713 内部 RAM 容量不足。所以要 DSP 扩展外部 SDRAM 用于存放图像数据。又由于 C6713 DSP 片内 RAM 掉电丢失数据，不能实现 DSP 的脱机工作。因此 DSP 外扩 FLASH 用于固化程序和数据。

4.1 DSP 最小系统的设计

TMS320C6713 DSP 硬件的最小系统包括电源、复位电路、时钟电路、EMIF 外部存储器和 JTAG 仿真接口^[18]。下面对其进行一一介绍。

4.1.1 电源设计

本文图像数据采集系统中，包含了 SAA7111A 模拟器件，还有 CPLD、FIFO、DSP 等数字器件，因此需要模拟和数字两部分电源。低功耗是嵌入式系统设计的重要指标。TI 公司的 TMS320C6713 DSP 采用双电源供电方案，即内核电压和 I/O 电压。其内核电压为 1.2V，工作在 200Mhz 以上，其功耗为

600mA；I/O电压为3.3V，其EMIF总线功耗为75mA。本系统在电路板上设计有5V电源接口，因此系统采用线性稳压芯片实现电压转换，为DSP提供内核电压和I/O电压。采用TPS75933产生3.3V的电源为DSP的I/O口、视频编解码器、视频缓冲存储器、SDRAM和FLASH供电。TPS75701产生了DSP内核需要的1.2V电压。其电路如图4.2所示。模拟电源与数字电源之间用磁珠进行了隔离。

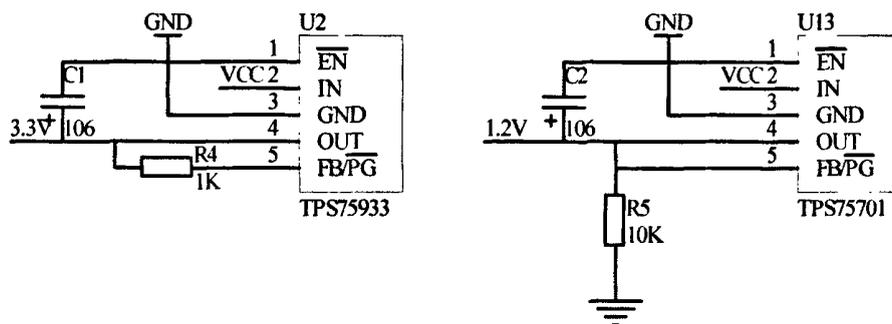


图4.2 电源电路

4.1.2 复位电路设计

DSP 复位电路的设计要求是，系统上电后，提供一个 200ms 左右的复位脉冲。由于系统对脉冲宽度和建立时间等有严格的要求，所以采用专用的复位芯片实现复位电路的功能。其原理图如图 4.3 所示。

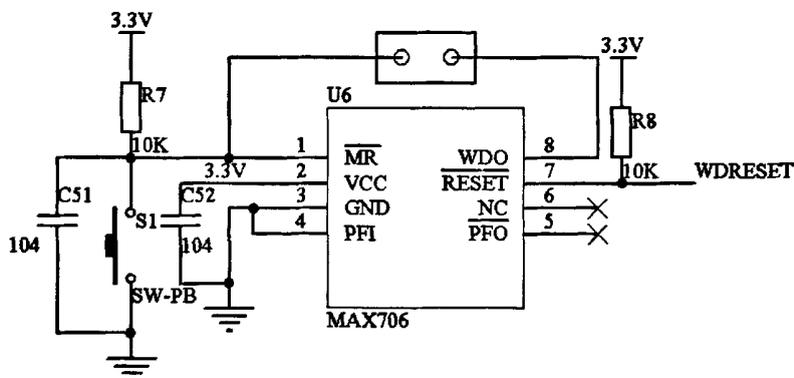


图4.3 DSP复位电路

复位电路采用MAX706S看门狗复位芯片，是一种多功能微处理器监控芯片。除了看门狗功能外，另外还具有上电自动复位、人工复位以及低电压报警等功能。该芯片在上电时或复位键按下时会产生200ms低电平复位脉冲。

满足DSP复位的要求。可以通过在MR和WDO引脚之间加跳线灵活选择复位方式。在系统调试阶段，可以将其断开，关闭看门狗功能，使程序能够稳定装载。当需要时再将两个引脚短接，使其看门狗功能。

4.1.3 PLL 系统时钟

1、PLL控制器概述

PLL控制器(见图4.4PLL控制器结构框图)的特征包括：软件可配置的PLL乘法器、除法器(OSCDIV1、D0、D1、D2和D3)和复位控制器。PLL控制器接受一个由CLKMODE0引脚的逻辑状态确定的输入时钟，来源于CLKIN引脚或片上振荡器输出信号OSCIN。可以通过软件配置乘法器和除法器来获得DSP的CPU以及片内外设所需要的时钟^[19]。

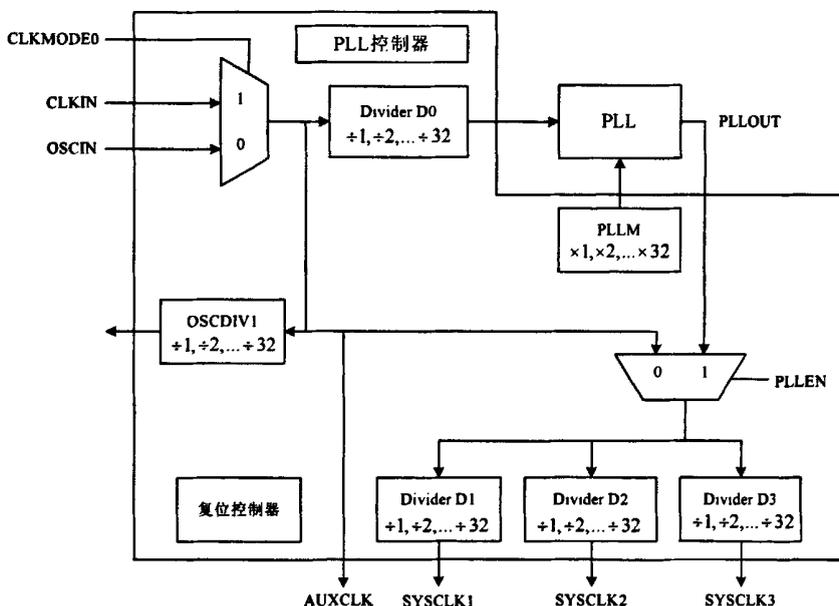


图4.4 PLL控制器结构框图

获得的时钟输出被传送到DSP核、外设和其他DSP内部的模块。

PLL控制器的输入参考时钟：

CLKIN：从外部振荡器(3.3V)输入的信号，CLKMODE0=1；

OSCIN：片上振荡器(1.2V)的输出信号，CLKMODE0=0。

PLL控制器的输出时钟：

AUXCLK：直接来自CLKIN或OSCIN的内部时钟输出信号；

- CLKOUT3: 除法器OSCDIV1的输出;
- SYSCLK1: 除法器D1的内部时钟输出;
- SYSCLK2: 除法器D2的内部时钟输出;
- SYSCLK3: 除法器D3的内部时钟输出。

PLL控制器可以通过PLL乘法器(PLLM)控制寄存器对PLL进行编程,乘法倍频范围从 $\times 1 \dots \times 32$ 。时钟除法器(OSCDIV1、D0、D1、D2和D3)也是可编程的,分频范围从 $\div 1 \dots \div 32$,并且可以被禁止。PLL控制状态寄存器(PLLCSR)的PLL使能位(PLEN)确定PLL控制器模式。当PLEN=1时为PLL模式,D0和PLL被使用;当PLEN=0时为旁路模式,D0和PLL被绕过。

2、系统PLL控制器的配置

时钟系统是DSP的一个重要组成部分。由于DSP的内部指令周期较小,外部晶振的主频不够。DSP的片内锁相环电路(PLL)电路可以根据需要对外接时钟进行倍频或分频,从而为DSP的CPU和片内外设提供合适的时钟。系统的时钟源由一个有源晶振提供,频率为50MHz。在设计时钟电路时必须保证时钟芯片的电源与I/O来自同一个电源。为了减少对外界元器件的干扰,一般时钟源频率不能太高,而是通过DSP内部的PLL电路倍频后为CPU提供时钟。在PCB布线时,应使时钟芯片离DSP尽量近,且附近尽量不要分布其他元件。为使时钟电路能够平稳工作,需要为时钟电路加入EMI滤波器。以下是系统DSP的PLL初始化的清单:

```

*(int *)PLL_CSR  &= ~CSR_PLEN;
*(int *)PLL_CSR  |= CSR_PLLRST;
*(int *)PLL_DIV0  = DIV_ENABLE + 0;
*(int *)PLL_MULT  = 8;
*(int *)PLL_OSCDIV1 = DIV_ENABLE + 9;
*(int *)PLL_DIV1   = DIV_ENABLE + 1;
*(int *)PLL_DIV2   = DIV_ENABLE + 3;
*(int *)PLL_DIV3   = DIV_ENABLE + 3;
*(int *)PLL_CSR  &= ~CSR_PLLRST;
*(int *)PLL_CSR  |= CSR_PLEN;

```

其中: 程序首先设置PLL控制/状态寄存器(PLLCSR)的PLL使能位

PLLEN=0, 用PLL的旁路模式即系统绕过除法器D0和PLL。然后写PLL_CSR的PLLRST=1即PLL复位; 在PLL乘法控制器PLL M中写入8, 即8倍频, 这样得到400Mhz的频率; PLL控制器除法1寄存器使能并2分频, 得到频率为200Mhz, 此频率即是DSP的CPU频率。PLL控制器除法3寄存器使能并4分频, 得到100Mhz频率, 此频率作为DSP外部存储器接口EMIF的频率。最后PLL退出复位状态。

4.2 图像存储模块的设计

4.2.1 TMS320C6713 DSP 的 EMIF 接口

TMS320C6000 DSP 的外部存储器接口(EMIF)可以支持与各种不同类型的存储器连接, 包括 SDRAM、SBSRAM 和 SRAM^[20]。只需要对 DSP 的 EMIF 控制器进行相应的配置就可以实现 DSP 与上述几种存储器的无缝连接。

TMS320C6713 DSP 采用统一寻址方式, 处理器有 20 位片外地址线(因封装形式不同而异)。因此, 扩展异步存储设备, 则只有 1M 的地址可供译码。C6000 EMIF 可扩展四个空间, 分别是 CE0、CE1、CE2、CE3。本设计中扩展了三个空间, 其中 CE0 空间扩展为 SDRAM 用于存放图像数据, CE1 为 FLASH 用于固化应用程序和串口芯片 TL16C550 用于串口通信, CE2 扩展为 FIFO 其用途在上一章中已经讲过。表 4.1 为 EMIF 扩展的片外存储器映射图。

表 4.1 扩展的片外存储器映射图

扩展空间	扩展存储器名称	空间大小	地址范围
CE0	SDRAM	16M	8000 0000h-80FF FFFFh
	保留	240M	8100 0000h-8FFF FFFFh
CE1	FLASH	1M	9000 0000h-901F FFFFh
	保留	254M	9020 0000h-9FFF FFFF
CE2	FIFO	只占一个地址	A0000000h

EMIF 支持 8 位宽和 16 位宽的 ROM 访问模式, 可以由 CE 空间控制寄存器(CECTL)的 MTYPE 位段设置。当从这些存储器空间读取数据, EMIF 会将多次读的数据打包为一个 32 位宽的值。该模式主要是用于 8 位和 16 位宽的 ROM 器件。在该模式下地址将被左移以便为存储器提供正确的地址。对于 16 位的 ROM, 移位量为 1 位; 对与 8 位 ROM, 移位量为 2。表 4.2 列出

了对于不同宽度的异步存储器，在访问 CE 空间时 EA 总线上的地址位。

表 4.2 用于异步存储器宽度的 EA 映射的字节地址

	EA 总线																			
	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
宽度	逻辑字节地址																			
*32	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2
*16	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
*8	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

系统中 DSP 扩展的 ROM 有 FLASH 和 FIFO。其中 FLASH 是 16 位宽，FIFO 宽度为 8 位，故 DSP 对其宽展的 FLASH 读写时地址要被左移一位。

4.2.2 TMS320C6713 与 SDRAM 接口的设计

1、同步动态 RAM(SDRAM)的选型

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，且具有任意可读/写的属性。因此，SDRAM 在系统中主要用作存储大量的原始图像数据以及其中间的处理结果。由于，图像数据量大，而且图像的搬运过程中要求速度很高^[21]。所以，选用容量大，读写速度快的 SDRAM 作为图像存储器。

系统采用了 HYNIX 公司的 HY57V561620CH-T 型 SDRAM 存储器。SDRAM 占用了 DSP 的 CE0 空间。HY57V561620 的容量是 128Mbit(8M×16bit)，速度可以达到 133MHz，满足系统图像数据的容量和存储速度的要求，采用单电源 3.3V 供电。所有的引脚和 LVTTTL 标准接口兼容。所有的输入和输出引脚都是在时钟的上升沿采样，并具有 UDQM 和 LDQM 数据屏蔽功能。内部为 4 块的结构，具有自动刷新和自刷新两种方式(在 64ms 内刷新 8192 行)，并且对于 CAS 延迟周期以及猝发读写长度和顺序都可以编程控制^[22]。

2、SDRAM 接口电路的设计

C6713 DSP 支持与 SDRAM 的无缝连接。可以通过对 DSP 的 EMIF 存储器映射寄存器配置来产生对 SDRAM 进行读写操作需要的时序。其 DSP 的 EMIF 控制寄存器配置如表 4.3 所示。

表 4.3 SDRAM 的参数配置

寄存器名称	SDRAM 配置
EMIF_GCTL	0000 0078h
EMIF_CE0	FFFF FF93h
EMIF_SDRAMCTL	5311 5000h
EMIF_SDRAMTIM	0000 0578h
EMIF_SDRAMEXT	000A 8529h

其中，全局控制寄存器对整个片外存储空间的公共参数的进行设置，CE0 空间控制寄存器控制相应的 SDRAM 存储空间的接口参数，另外三个 SDRAM 寄存器负责控制所有属于 SDRAM 空间的存储接口情况。其硬件连接如图 4.5 所示。

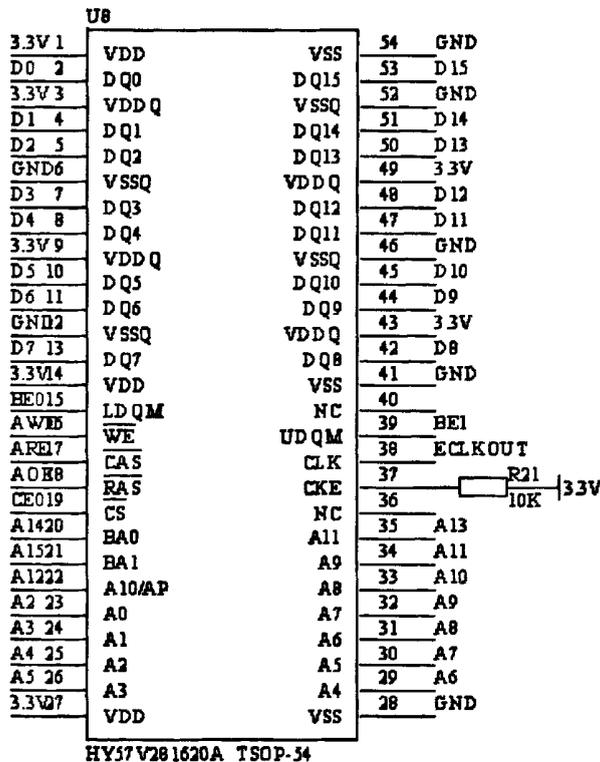


图 4.5 DSP 与 SDRAM 接口电路图

其中，DSP 的 ECLKOUT 与 SDRAM 的 CLK 相连接，作为两者工作的同步时钟。DSP 的 EA[15..14]作为 SDRAM 的 BANK 选择共有 4 个数据块。

每个数据块 $2M \times 16\text{Bit}$ 的数据。DSP 的 EA10~EA2 为 SDRAM 提供了列地址 A8~A0, EA2~EA13 提供了 SDRAM 的行地址 A11~A0。从而每个 BANK 的行列地址和为 21 根地址线, 所以可以有 2M 的寻址空间。

4.2.3 TMS320C6713 与 FLASH 接口的设计

1、FLASH 选型及简介

Flash 存储器又称闪存, 它结合了 ROM 和 RAM 的特点, 不仅具备电子可擦除可编程 (EEPROM) 功能, 而且不会断电丢失数据, 同时能快速读取数据。它具有可以在线电擦写、低功耗、大容量、擦写速度快等特点。

AM29LV800BB-90EC 是 AMD 公司 AM29LV800B 系列的一种器件, 其主要性能如下: 访问时间为 90ns; 存储容量 8Mbit; 工作温度范围 $-55^{\circ}\text{C} \sim +125^{\circ}\text{C}$; 在线编程电压 3.0V~3.6V; 低功耗; 读操作时电流为 7mA; 编程 / 擦除时电流为 15mA^[23]。主要引脚如下:

A0~A18: 19 根地址线, 从而有 512K 的寻址空间;

DQ0~DQ14: 15 根数据线;

$\overline{\text{BYTE}}$: 选择 16 位字模式该引脚置 1 或 8 位字节模式该引脚置 0;

DQ15 / A-1: 当配置为 16 位字模式时, 该引脚为数据 I/O, 组成 16 位字的最高位 DQ15; 当配置为 8 位字节模式时, 该引脚是输入地址线的最低有效位 A-1; 从而可以使 Flash 配置为 $1M \times 8\text{Bit}$ 或 $512K \times 16\text{Bit}$ 。

$\overline{\text{CE}}$: 片选使能;

$\overline{\text{OE}}$: 输出使能;

$\overline{\text{WE}}$: 写使能;

$\overline{\text{RY/BY}}$: 忙 / 闲;

$\overline{\text{RESET}}$: 硬件复位, 低电平有效。

读模式: 为了读取数据, 系统需将 $\overline{\text{CE}}$ 和 $\overline{\text{OE}}$ 设为低电平。同时将 $\overline{\text{WE}}$ 设为高电平。在器件上电或硬件复位后。器件默认设置为读模式。

写模式: 为了向器件写入数据或指令, 系统需将 $\overline{\text{CE}}$ 和 $\overline{\text{WE}}$ 置为低电平, 同时将 $\overline{\text{OE}}$ 置为高电平, 写操作需要 4 个周期, 前 3 个周期向两个特定地址写入 3 个特定字符, 第 4 个周期将所需数据写入相应地址。

复位：向器件的任意地址写入特定指令，该器件复位，复位后默认为读模式。Flash 的操作需要特殊的指令。下面给出 Flash 字节编程的程序清单：

```

*(volatile unsigned short*)(FlashBaseAddr + 0x555*4) = 0xaa;
*(volatile unsigned short*)(FlashBaseAddr + 0x2aa*4) = 0x55;
*(volatile unsigned short*)(FlashBaseAddr + 0x555*4) = 0xa0;
*(volatile unsigned short*)(FlashBaseAddr + Addr*4) = Val;
WriteIsOver();

```

其中：FlashBaseAddr=0x90000000 为的 CE1 空间的基地址。Val 为要写入的数据值。在对 Flash 进行编程命令时首先指定的地址写入命令序列，为 0xaa, 0x55, 0xa0。由于 DSP 的 EMIF 总线的 EA(n+2)对应 Flash 的 An 地址线，所以 Flash 的相应地址需要乘以 4。在写入字节后，需要通过轮询校验的方式来检验编程操作是否正确。在实际调试中，可以先读 Flash 的 ID 号与实际的产品 ID 号对照，以检验软硬件设计是否正确。然后，对 Flash 进行擦除操作，用 DSP 开发软件 CCS 提供的 Memory 观察窗口检验是否为 0xFF。最后可以进行实际的编程操作。

2、FLASH接口电路设计

Flash被映射到EMIF的CE1空间，可以与EMIF总线进行无缝连接。其读写控制信号、地址信号线和数据信号线可直接与DSP的相应的信号线直接连接。根据Flash的读写特点，将CECTL1寄存器配置为16位的异步ROM接口。这样DSP就可以自动的输出符合Flash读写时序的时序，从而DSP可以完成对Flash的读写操作。

如图4.6所示，其中地址线A[0..18]同DSP的地址EA[2..20]相连，16位数据线DQ[0..15]同DSP的数据D[0..15]相连，输出使能信号管脚 \overline{OE} 与DSP的AOE管脚相连，输出使能信号管脚 \overline{WE} 同DSP的AWE管脚相连。

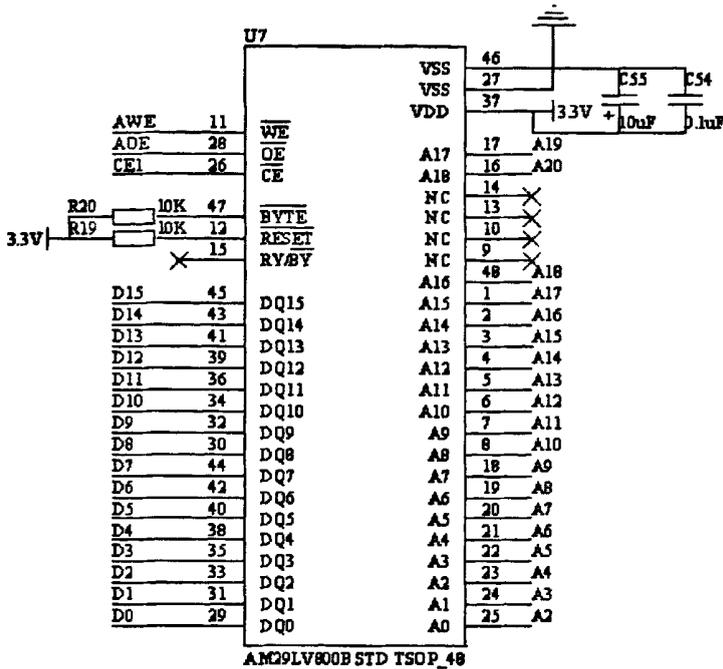


图4.6 DSP的FLASH扩展电路图

4.3 图像搬运模块的设计

TMS320C6713 DSP具有16个EDMA通道，通道间的优先级可设置。EDMA控制器由事件和中断处理寄存器、事件编码器、参数RAM以及硬件地址产生几部分构成。EDMA的参数RAM的容量为2KB，总共可以存放85组EDMA传输控制参数^{[24][25]}。多组参数还可以彼此连接起来，从而实现某些复杂数据流的传输。图4.7给出1组EDMA传输参数的内部结构。

系统使用1片位宽为8Bit的FIFO与DSP进行数据交换，因此，源数据区位宽为8Bit，为了使DSP的EMIF与8Bit的异步存储器接口，必须在CE2的空间控制寄存器中将其设置为8位异步接口模式。由于采集的视频数据要从FIFO缓存搬运到系统的SDRAM中，而SDRAM采用16Bit设计。因此，EDMA传输参数中的目的地址的数据宽度为16Bit。因此对于系统而言，EDMA的传输类型为1D到2D传输。

对EDMA的可选参数控制位(如图4.8所示)设置为0x21300001，即设置为块同步1D到2D传输。EDMA传输源地址设置为0xA0000000，由于读取FIFO时无需改变读地址，因此无需对源地址进行更新。目的地址设置为

0x80000000, PAL制式图像数据先输出奇数场后输出偶数场, 故规定奇数场首地址为0x80000000, 偶数场首地址为0x80000400, 这样设计就是为了能够将1帧图像放在连续的地址进行存放。传输结束后产生中断, CPU开始对图像数据进行处理。

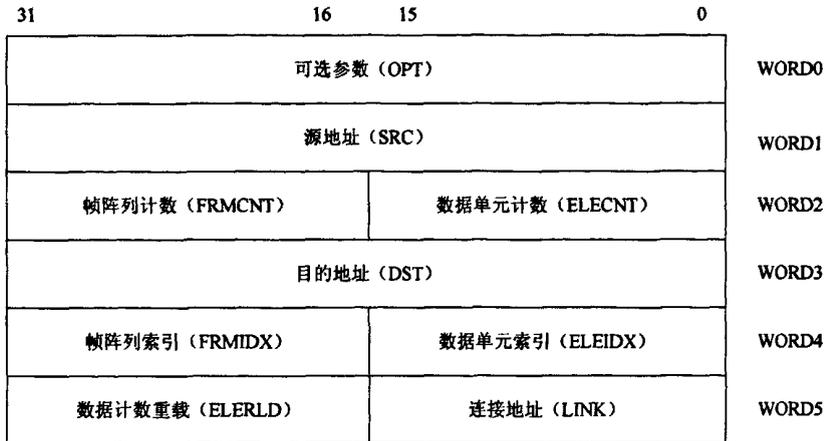


图4.7 EDMA的参数存储结构

经过CPLD对采集时序的控制后, 可以得到512×512的图像数据, 采用16位输出格式, 则一行共有512×8Bit数据, 占用1024个数据单元, 则数据单元计数ELECNT=1024。一帧图像由奇偶两场组成, 一场数据量为512×256×8Bit。如果FIFO的容量超过一半, 则FIFO的 \overline{HF} 管脚为低, \overline{HF} 将一直保持为低直到FIFO读写指针之差小于或等于FIFO的一半。由于系统设计时使用 \overline{HF} 来触发DSP的INT6, 故从FIFO读取一场图像数据需要128次EDMA中断, 每次从FIFO读取2K即4行数据到SDRAM。

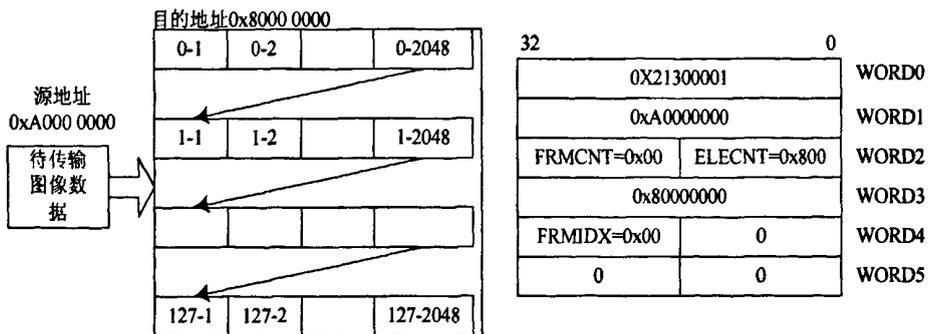


图4.8 块同步1D到2D的传输

4.4 图像传输模块的设计

4.4.1 TL16C550 芯片的介绍

TL16C550是一种具有异步串行通信功能的大规模集成电路芯片。其主要功能是为DCE设备和DTE设备之间提供可靠、灵活的接口服务。TL16C550具有以下特点^[26]:

- 1、在FIFO方式下，16字节缓冲发送器和接收器；
- 2、高达16MHz的时钟和1M波特率的发送速度；
- 3、波特率可编程；
- 4、完全可编程的串行接口特性。可选择5、6、7、8位串行数据位，可设置奇、偶或者无奇偶效验位，可设置1个、1个半或者2个停止位；
- 5、独立控制的发送、接收、线路状态以及中断设置。

对TL16C550的操作主要就是对其内部寄存器进行设置来实现的，对这些寄存器的访问是通过3位地址线来进行选择的，还有LCR控制寄存器的D7位DLAB参与辅助定义。

表4.4列出了如何通过DLAB和A0、A1、A2来选择控制器，表中列出了这11个寄存器的中文名称和英文缩写^{[27][28]}。

表4.4 TL16C550片内寄存器

DLAB	A2	A1	A0	寄存器
0	L	L	L	接收/发送缓冲寄存器(RBR/THR)
0	L	L	H	中断允许寄存器(IER)
X	L	H	L	中断识别寄存器(只读)(IIR)
X	L	H	L	FIFO控制寄存器(写)(FCR)
X	L	H	H	线路控制寄存器(LCR)
X	H	L	L	Modem控制寄存器(MCR)
X	H	L	H	线路状态寄存器(LSR)
X	H	H	L	Modem状态寄存器(MSR)
X	H	H	H	Scratch寄存器(SCR)

1	L	L	L	波特率因子寄存器低位(DDL)
1	L	L	H	波特率因子寄存器高位(DLM)

4.4.2 通信接口电路的设计

C6713没有专门的UART接口,但有多通道缓冲串口(McBSP)为了实现串行通信。很多情况下都用软件对C6713的McBSP口编程来模拟异步串口。许多学者曾对应用McBSP口实现串口进行过研究,软件编程比较复杂,且占用很大的系统资源。本文提出采用专门的异步通信芯片TL16C550来扩展C6713的串口,可以避免上述方法的缺点,使DSP器件实现稳定、准确的串行通信。并将介绍C6713和PC机之间通过16C550进行串行通信的具体实现方法。该方案具有很强的灵活性和实用性,适用于很多需要进行串口通信的场合。PC机与DSP通过16C550扩展RS232串口进行串行通信。其系统框图见图4.9。

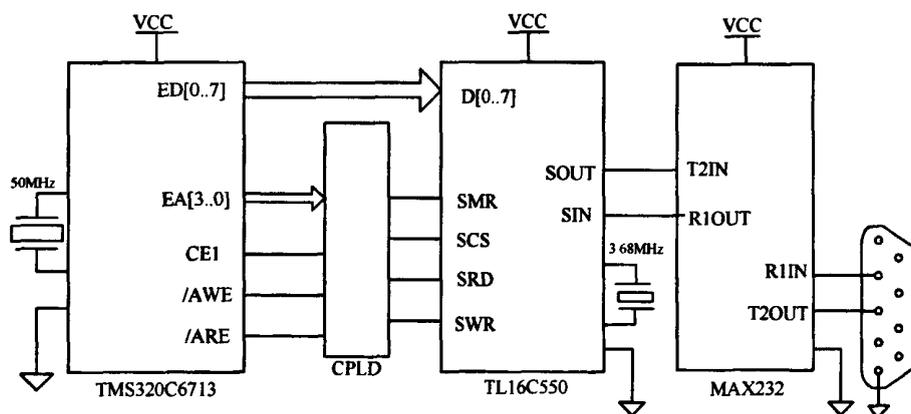


图4.9 串行通信的系统框图

设计中将RD2、WR2均接到了低电平(无效状态),仅由RD1和WR1控制读写,读写信号由与6713连接的附加电路来提供。XIN、XOUT端外接3.6864MHz的晶振作为基准时钟频率。16C550的工作频率可以通过波特率除数低位锁存器LSB和波特率除数高位锁存器MSB来设定。本课题UART的波特率为9600bps,计算得LSB置为18H,MSB设为00H。

SIN为串行数据输入脚, SOUT为串行数据输出脚。这两根信号接到MAX232芯片上,由MAX232实现串行数据的TTL/CMOS电平与RS232电平之间的转换^[29]。

4.4.3 串口通信的软件实现

系统中DSP与16C550之间的时序匹配和逻辑由CPLD完成。在CPLD中软件编写译码器和锁存器，给TL16C550提供复位、读、写和选通信号。其在CPLD软件编程环境Quartus II中的逻辑图如图4.10所示。

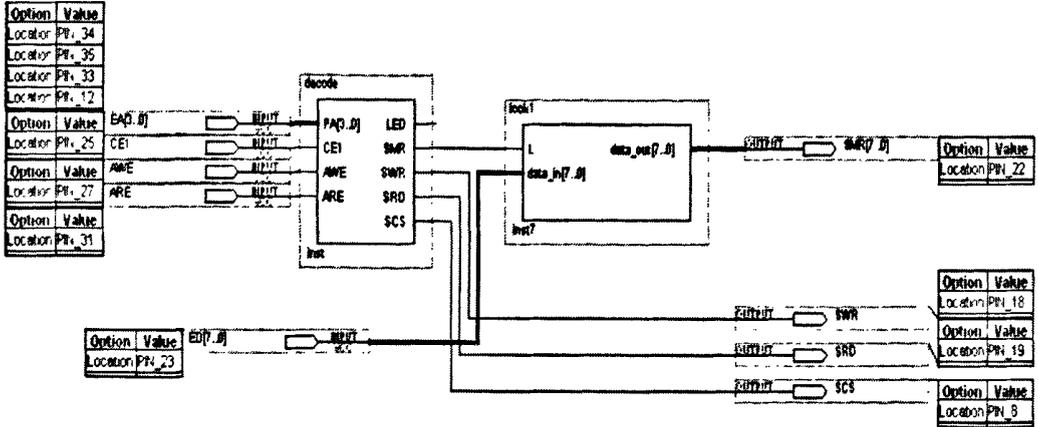


图4.10 DSP与TL16C550之间时序逻辑图

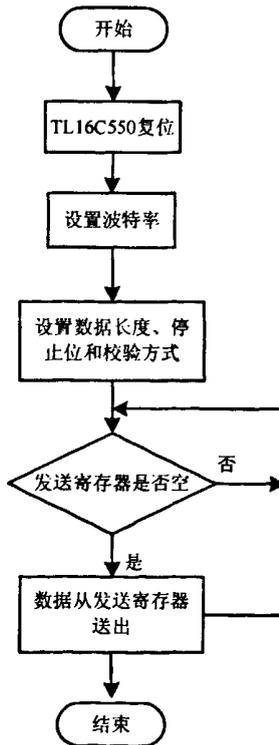


图4.11 串行通信主程序流程图

为了激活扩展口的串行通信功能，需要通过DSP对TL16C550进行一系列的初始化操作。TL16C550初始化的任务是设置工作参数，这些参数包括串行通信数据串的数据位数、停止位数、奇偶校验等。另外，还需要设置发送和接收的波特率及中断方式。串行通信主程序流程图见图4.11。

4.5 本章小结

本章介绍了图像采集基板的设计。具体对图像搬运部分、图像存储部分和图像传输部分进行了说明。由于采用了EDMA的搬运方式，使得CPU从图像的采集与搬运的过程中解放出来，可以专注于图像的处理任务。只要将图像存储部分配置成乒乓缓存的形式，就可以使图像的处理和图像数据的采集任务同时进行。图像数据的传输方式采用串口通信的方式，主要是因为实现简单。虽然串口数据传输比较慢，但是由于系统最终只需要传输少量的处理结果数据，完全可以满足要求。该部分为图像处理算法实现提供了良好的硬件平台。

第5章 系统的调试及 Bootloader 的实现

5.1 系统的调试

基于DSP的视频采集、处理与存储系统涉及了DSP、CPLD、A/D、FIFO等器件，是比较复杂的系统，影响系统正常工作的原因也是多样的，因此选择合理的系统调试方案是完成系统调试的前提。参考以往系统的调试经验，首先要保证硬件平台的正确，然后再进行软件部分的调试以及软硬件联调，这样有利于问题隔离，可以方便的进行故障定位。

5.1.1 系统硬件的调试

系统的硬件调试遵循了先功能模块，最后再整体模块的调试顺序。同时这两个步骤又不是相互孤立的，而是交互进行的。在功能模块调试之前，一定要先检测电路板有无短路、断路现象，因此把静态调试放在了硬件调试的第一步。具体的调试步骤如图5.1所示。

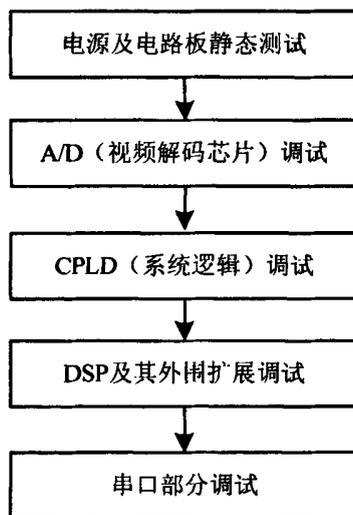


图5.1 系统硬件调试步骤

5.1.1.1 电路板静态及电源的测试

高速电路板的结构复杂，走线、元件密集，对电气性能有较高要求。因此，在拿到PCB电路板而未焊接元件之前，首先要检查有无短路、断路的情

况发生，尤其对于走线比较密集的部分要仔细检查。检查无误后，用万用表对电源、地以及一些重要的时钟线、片选、复位等线路检查。看PCB板内层有无断路和短路情况发生，最后检查信号网络有无连接不良的情况，如有很大电阻则说明传输线路连接不良。以上检查无误后开始对PCB进行焊接。焊接时最好先焊接电源部分，然后再按功能模块进行焊接。

焊接完电源模块后，对PCB板进行上电，用万用表测量输入电压、输出电压是否正确和稳定以及电源模块有无过热现象，如发现异常要立即断电检查。如果电源模块正常工作，再焊接其他功能模块，检测方法和电源模块类似。

在电源模块以及各功能模块焊接完成并通过初步检查后，可以对PCB板进行长时间(比如2小时)供电观察电路板能否稳定工作，如果稳定则可以确定电路板通过静态测试。

5.1.1.2 视频 A/D 的调试

视频A/D部分的调试主要是通过TMS320C6713的IIC通信程序对视频A/D进行寄存器配置，所以调试A/D时一定要调试IIC总线。

根据芯片说明用示波器测试行、场同步信号，时钟信号等管脚输出波形是否正确。如果输出波形正确说明A/D配置成功。

另外可以根据图像的效果对视频A/D的内部参数进行微调，从而获得更好的图像效果。

5.1.1.3 系统逻辑部分的调试

CPLD相对于FPGA使用简单，一般只要保证外围电源以及其JTAG的管脚顺序与下载电缆的相同就可以进行正确的程序下载了。

5.1.1.4 DSP 及其外围扩展芯片的调试

TMS320C6713作为本系统的核心处理芯片，工作频率最高，信号复杂、外围器件多，控制整个硬件系统的运行，因此确保TMS320C6713正常工作是整个系统正常工作的前提。DSP调试分以下几部分进行。

- 1、 TMS320C6713 DSP由外部提供的50MHz晶振经内部PLL倍频到

200MHz。为保证时钟运行正常并工作在200MHz，需检查DSP的CLKOUT，看输出的频率、电平是否正常、稳定；

2、TMS320C6713 DSP的JTAG接口通过仿真器与计算机连接，如果CCS2.2开发界面能够正确的打开，则表明JTAG接口通信正常；然后在CCS2.2编辑界面下使用存储器观察命令查看内部存储空间。如果读写正常，则表明内部存储空间通信正常。最后下载一个小的测试程序，如果运行结果正确，则表明TMS320C6713内部CPU的工作正常，可以进行外部接口的调试工作。如果不能正常打开CCS可检查仿真器是否正确驱动，再进行仿真器复位，电路板硬件复位或断电放电再重新启动等操作；

3、TMS320C6713 DSP通过EMIF接口与帧存储器、外部视频输入、输出数据缓冲器以及FLASH进行通信。将EMIF接口寄存器按软件设计要求设置，在CCS2.2编辑界面下使用存储器观察命令查看外部存储空间。如果读写正常，则表明外部存储空间通信正常。因为总线仲裁申请信号EHOLD具有最高优先级，所以在对EMIF接口调试时，一定要确保EHOLD信号为高电平。否则，TMS320C6713会误认为EMIF空间被外部占用，一直处于等待状态。

5.1.1.5 串口部分的调试

首先确保主板上CPLD部分程序的正确，为串口芯片16C550的寄存器分配地址。然后正确初始化串口芯片，可以做一个小程序向主机传送数据看一下，并用串口调试助手来观察数据。确保无误后，向上位机传送有规律的图像数据，用已编好的图像显示软件看一下是不是期望的图像。

5.1.2 DSP 调试环境 CCS 介绍

CCS是TI公司为了配合TI DSP的硬件开发而推出的软件集成开发环境。内部集成了C6000代码产生工具，包括C6000编译器、汇编优化器、汇编器以及连接器及其可视化软件，软件模拟器，DSP/BIOS实用内核插件，RTDX主机与目标板之间实时数据交换。利用CCS可以完成DSP软件的管理、编辑、编译、调试及其性能测试，为使用者提供了比较完善的软件开发环境。图5.2为DSP开发软件CCS的图形界面。它还允许编辑C代码和汇编源代码，还可以在C代码之后显示与之对应的汇编指令。只需要在菜单VIEW中选择Mix

Source/Asm选项,即可看到C之后跟着的汇编语句。CCS使用工程来管理应用程序的设计文档。工程中包含有源代码、目标文件、库文件、连接命令文件和头文件。在以往的开发工具中,编译、汇编和链接是各自独立的执行程序,开发设计人员需要熟悉每个程序的相关参数,而在CCS集成开发环境下只需要修改这些参数即可。而且CCS能自动查找工程所需要的头文件,并将它加入到工程中^[30]。

CCS的调试工具有下列特性:

- 1、 设置一个或者多个断点;
- 2、 在断点处自动更新窗口;
- 3、 使用Watch窗口查看变量;
- 4、 查看和编辑存储器、寄存器;
- 5、 使用Probe Point工具在主机与目标系统间传输数据流;
- 6、 可对目标系统中的信号绘图显示;
- 7、 使用Profile Point查看执行统计信息;
- 8、 观察目标系统中执行的反汇编和C指令;
- 9、 CCS还提供GEL语言,允许开发者向CCS菜单中增加功能。

在软件开发的分析阶段,传统的调试手段对诊断实时系统中的错综复杂问题无能为力。CCS提供支持实时分析的DSP/BIOS插件,可以用它来实时跟踪和监视一个DSP应用程序,同时对实时性能的影响达到最小。DSP/BIOS API提供以下实时分析特性:

- 1、 程序跟踪:显示写入目标日志的事件,并在程序执行过程中反映动态控制流程;
- 2、 性能监控:跟踪统计目标板资源的使用情况,如处理器负载和线程时序等;
- 3、 文件流:将目标板上的I/O对象与主机上的文件联系在一起。

TI DSP芯片提供片上仿真支持,使CCS能控制程序的运行并实时监控程序的活动。仿真器提供与主机通信的JTAG口,主机与目标DSP通信是通过JTAG接口来完成的,这种连接方式对DSP目标系统的实时性能没有太大的影响,片上仿真硬件提供以下功能:

- 1、 运行、停止、或复位DSP芯片;

- 2、将代码和数据加载到DSP芯片中；
- 3、检查硬件指令或数据相关的断点；
- 4、各种计算功能，包括精确到指令周期的剖切(Profile)功能；
- 5、提供主机和目标系统间的实时数据交换。

CCS支持这些片上的仿真功能，这样可以提供给开发者一个真实的系统工作过程，从而缩短开发时间。

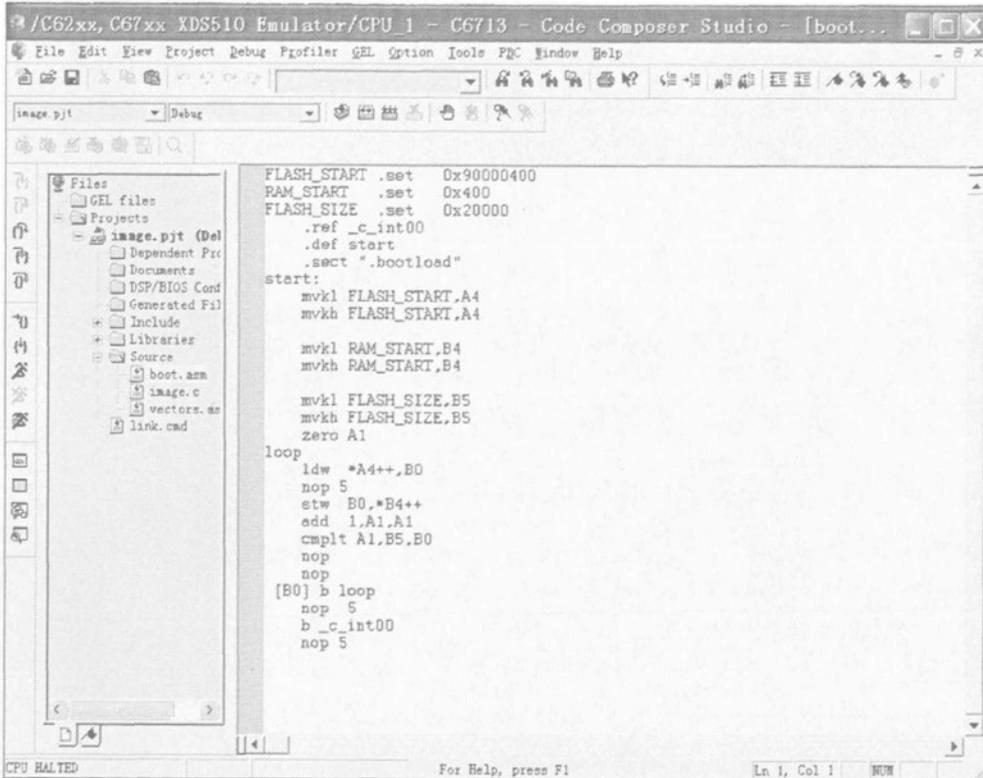


图5.2 CCS的开发界面

软件开发首先选择编程语言编写程序，TI的C6000系列DSP提供了两种编程语言：汇编语言和C/C++。原则上两种语言都可以使用，但对于图像处理的算法一般比较复杂，用汇编来做是不现实的。在系统的开发过程中，采用混合编程的方式^[31]。系统的中断和Bootloader的程序采用汇编语言来完成，而涉及到系统管理和算法的程序用C语言完成。

5.1.3 DSP 系统的软件设计

系统软件包括DSP图像采集程序和上位机的显示程序。其中DSP程序主

要完成对图像数据搬运、处理和传输。其DSP部分的软件流程图如图5.3所示。

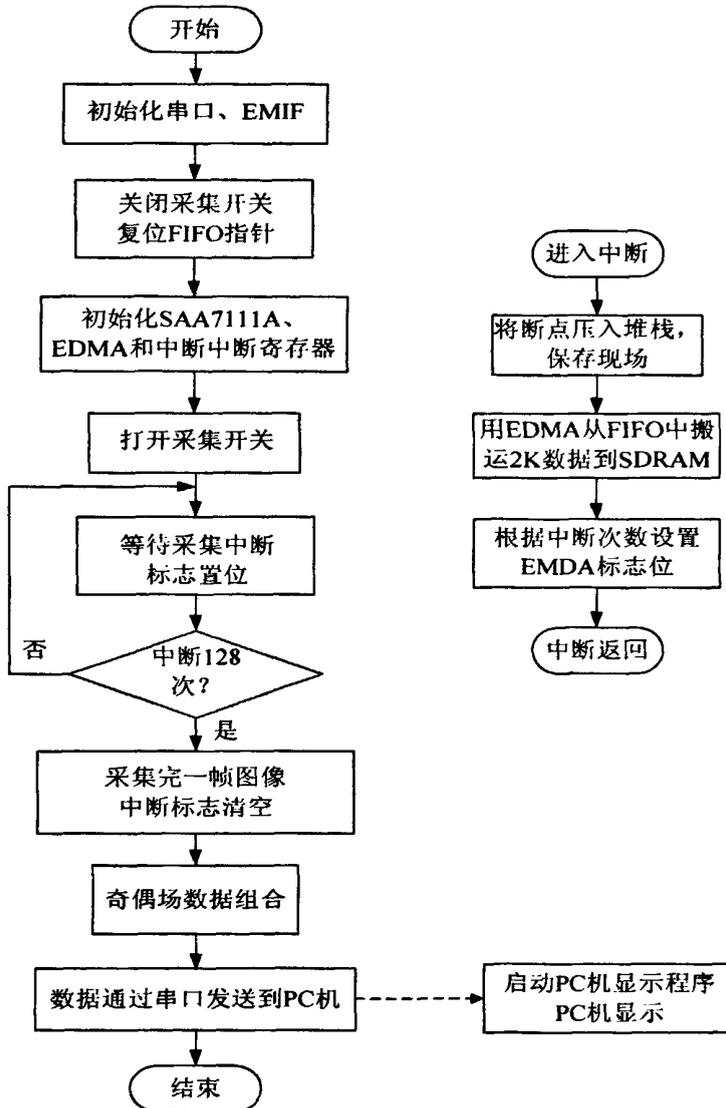


图5.3 软件流程图

DSP复位以后首先对串口和EMIF，锁相环寄存器进行初始化。然后，关闭采集开关(即关闭CPLD中的D触发器)，对FIFO进行复位使其读写指针都指向FIFO的起始地址。接下来初始化中断、EDMA并配置SAA7111A的内部寄存器。上位机向下位机发出开始采集命令(即打开D触发器)，下位机开始进行图像采集。采集到的图像数据由SAA7111A送到FIFO中。当FIFO中存有2K数据达到半满时，触发DSP的INT6中断，进入中断程序。DSP通过EDMA方式将FIFO中已有的2K数据搬运到SDRAM中。程序进行判断是否中断了128次，

若没有则继续等待中断，进行循环采集；若中断有128次则表示采集完一帧完整图像。一帧512×512的图像共有256K字节，所以程序需要有128次中断完成对数据的搬运。得到一帧图像后，将奇偶场数据组合为一幅图像。也可以编程实现各种图像处理算法，采集或处理完的一帧图像通过串口发送到上位机。

5.1.4 系统软硬件总体调试

在完成系统硬件的调试后，需要在硬件平台上进行软件和硬件总体调试，最后达到保证系统各个功能模块协调工作。图5.4为系统总体调试工作平台。

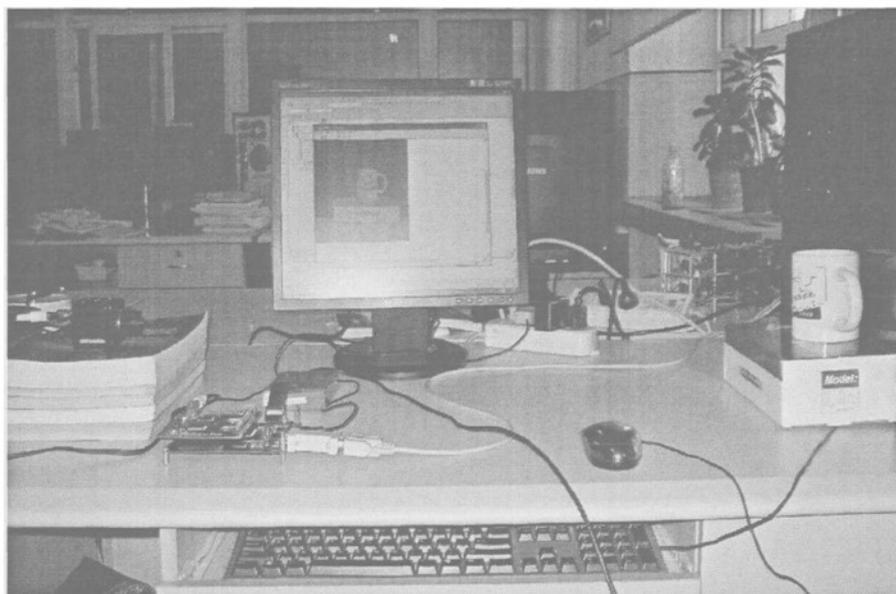


图5.4 系统总体调试工作平台

5.2 系统的 Bootloader 实现

5.2.1 TMS320C6000 DSP 加载过程简介

由于非易失存储器的存取速度较低，因此C6000 DSP内部不设FLASH等非易失存储器。而实际系统调试完成后，必须做到能够脱机运行，只有这样才是一个完整的产品。因此，只能靠执行外部非易失程序存储器的指令代码工作(速度较低)，或通过其它方法将其实际工作的指令代码拷贝到片内或片外其它高速RAM中运行来工作，这个过程就是引导加载，也就是 Bootloader^{[32][33]}。

C6000系列DSP的启动加载方式包括不加载、主机加载和EMIF加载3种。3种加载方式的比较：不加载方式仅限于存储器0地址不是必须映射到RAM空间的器件，否则在RAM空间初始化之前CPU会读取无效的代码而导致错误；主机加载方式则要求必须有一外部主机控制DSP的初始化，这将增加系统的成本和复杂度，在很多实际场合是难以实现的；EMIF加载方式的DSP与外部ROM / Flash接口较为自由，但片上Bootloader工具自动搬移的代码量有限(1KB / 64KB)，C6713的片上Bootloader工具自动搬移的代码量只有1KB。因此，必须编写二次Bootloader程序。系统采用的是常用的EMIF加载方式。

5.2.2 EMIF 的加载分析

实际应用中，通常采用的是EMIF加载方式，把代码和数据表存放在外部的非易失性存储器里(常采用Flash器件)。硬件方面，其与16位宽度的Flash器件的连接，这个在第四章已有介绍。C6713 DSP的配置引脚及其定义如表5.1所列^[34]。

表5.1 C6713的加载方式配置引脚

引脚	意义
HD[4:3] (BOOTMODE)	加载模式配置引脚(BOOTMODE): 00- CE1空间32位加载, HPI/仿真加载。仿真加载是HPI加载的一种; 01- CE1空间8位外部异步ROM默认时序加载; 10- CE1空间16位外部异步ROM默认时序加载; 11- CE1空间32位外部异步ROM默认时序加载。
HD8	字节排序模式(Endian) 0- 系统以大端模式运行; 1- 系统以小端模式运行。

对于C6713 DSP，片上的Bootloader工具只能将1KB的代码搬入内部RAM。系统应用程序的大小超过这个限制，所以，需要在外部Flash的前1KB范围内预先存放一小段程序，待片上Bootloader工具把此段代码搬移入内部并开始执行后，由这段代码实现将Flash中剩余的用户应用程序搬移入内部RAM

中^[35]。此段代码可以被称作一个简单的二级Bootloader。图5.5所示为使用二级Bootloader时的CPU运行流程。首先，编辑一个内存复制程序，即引导加载程序Bootloader，实践证明这种程序都小于1K，并且把这个程序烧写到片外FLASH开始的1K地址空间；然后，再编辑自己需要的应用程序，并且把这段代码放到片外FLASH的1K之后空间里去。这样经过第一级引导加载后，Bootloader在C6713内部RAM运行，完成应用程序从片外FLASH到片内RAM的复制，完成第二级引导加载，同时Bootloader把控制权交给应用程序^{[36][37]}。

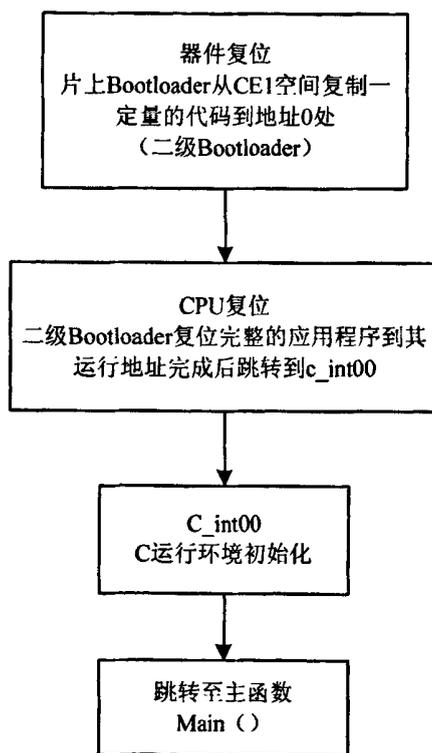


图 5.5 使用二级 Bootloader 时 CPU 运行流程

5.2.3 引导工程的文件流程

系统工程在 CCS 开发环境下使用 C 语言编程，这样可以缩短开发周期，提高工作效率，并具有移植性好的优点^[38]。引导工程的文件流程如图 5.6 所示。

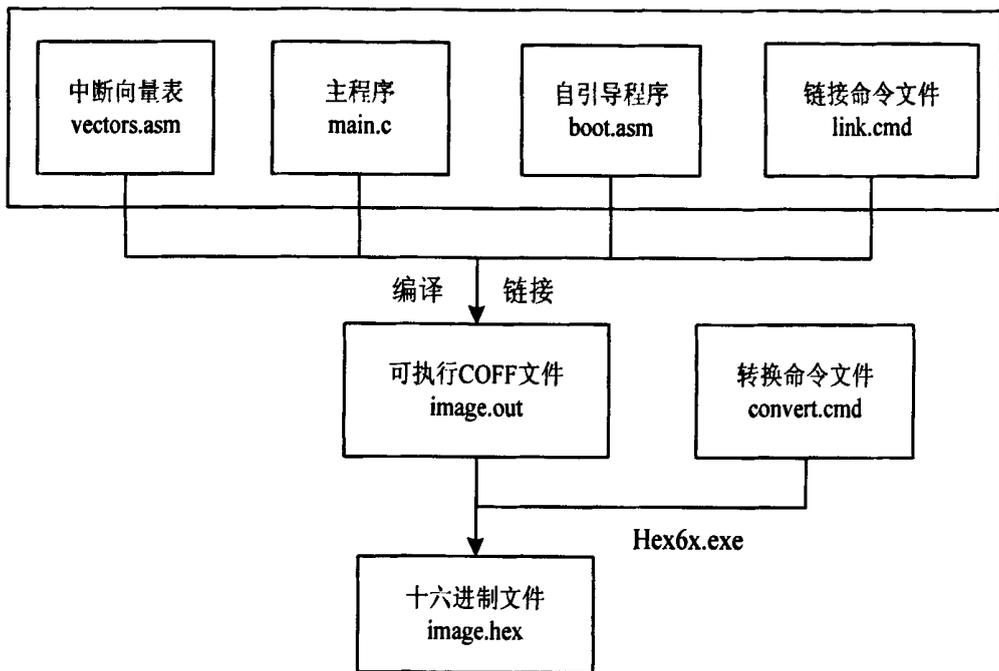


图 5.6 引导工程的文件流程图

5.2.3.1 自引导的汇编程序 boot.asm 的设计

由于执行二级 Bootloader 时 C 的运行环境还未建立起来，所以必须用汇编语言编写。自引导汇编程序主要是配置基本的寄存器，并将保存在外部 FLASH 中的二进制程序拷贝到 DSP 内部的 RAM 中再执行。由于 TMS320C6713 DSP 自动拷贝 1KB，因此起始地址是从 0x00 开始的 0x400 空间，汇编程序如下：

```

.sect ".boot_load" ; 定义数据段
.ref _c_int00 ; 声明外部函数
.global _boot ; 定义全局函数
_boot:
; 先设置控制寄存器，如 EMIF_GCTL 等，(略)
; 拷贝 FLASH 中的程序到 DSP 内部 RAM
mvkl 0x00000400, A4 ; A4 为 RAM 地址指针
|| mvkl 0x90000400, B4 ; B4 为 FLASH 地址指针
mvkh 0x00000400, A4
  
```

```

|| mvkh 0x90000400, B4
zero A1 ; A1 用作计数器
_boot_loop: ; DSP 开始读取 FLASH 中程序
ldb *B4++, B5
mvkl 0x0000F200, B6 ; B6 为需要拷贝的字节数
add 1, A1, A1
|| mvkh 0x0000F200, B6
cmlpt A1, B6, A0
nop
stb B5, *A4++
[B0] b_boot_loop
nop 5
mvkl .S2 _c_int00, B0 ; 循环结束后, 跳转到主函数 main 执行
mvkh .S2 _c_int00, B0
B .S2 B0
Nop 5

```

5.2.3.2 中断向量表 vectors.asm 的设计

中断向量表缺省保存在 DSP 芯片内部 RAM 的 0x800 地址开始的 0x200 字节空间, 上电或复位后, 芯片自动运行复位中断。因此, 复位中断向量设置为引导程序(_boot)的入口地址, 引导程序的主体在 boot.asm 中定义。部分程序如下:

```

.sect ".vectors" ; 段声明
.ref _c_int00 ; 声明外部函数
.ref _func_int6; 声明中断 int6;
RESET_RST: ; 复位中断向量
mvkl _c_int00, B0 ; 装载 C 运行环境程序地址
mvkh _c_int00, B0
B .S2 B0 ; 跳转到 C 运行环境程序执行
NOP 5

```

5.2.3.3 主程序 main.c 的设计

主程序是 DSP 要实现具体功能的主体，其定义的主函数 main() 经编译后在函数 `_c_int00` 中调用，因此在上面的引导程序结束时，将跳转到函数 `_c_int00`，即主函数 main 执行。

5.2.3.4 链接命令程序 link.cmd 的设计

链接命令程序用于定义系统各存储器的地址及大小，并分配编译后各段到相应的存储空间，link.cmd 内容如下：

```
-c
-lrts6700.lib
MEMORY
{
  BOOT_RAM:   o = 00000000h      l = 00000400h
  vecs:       o = 00000800h      l = 00000200h
  IRAM:       o = 00000a00h      l = 0002ffffh
  CE1:        o = 90000000h      l = 00100000h }

SECTIONS
{
  .vectors > vecs fill = 0
  .boot_load > BOOT_RAM fill = 0
  .text > IRAM fill = 0
  .stack > IRAM fill = 0
  .bss > IRAM fill = 0
  .cinit > IRAM fill = 0
  .far > IRAM fill = 0
  .systemem > IRAM fill = 0
  .cio > IRAM fill = 0
}
```

5.2.3.5 转换命令程序 convert.cmd 的设计

上面的工程文件经 CCS 系统编译、汇编后生成可执行 COFF 文件(.out), 它需要转换为二进制文件, 再写入到 FLASH 中。CCS 开发系统带有转换程序: hex6x.exe 将可执行 COFF 文件(.out)转换为十六进制文件(.hex)命令行的格式为:

```
hex6x.exe convert.cmd
```

其中 convert.cmd 内容如下:

```
image.out ; 输入文件名 .out 格式
```

```
-x
```

```
-map image.map ; 生成映射文件
```

```
-image
```

```
-memwidth 8 ; 内存的位宽
```

```
-o image.hex ; 输出文件名 .hex 格式
```

```
ROMS
```

```
{
```

```
    FLASH: org = 000h, len = 0x7900, romwidth = 8, files = {image.hex}
```

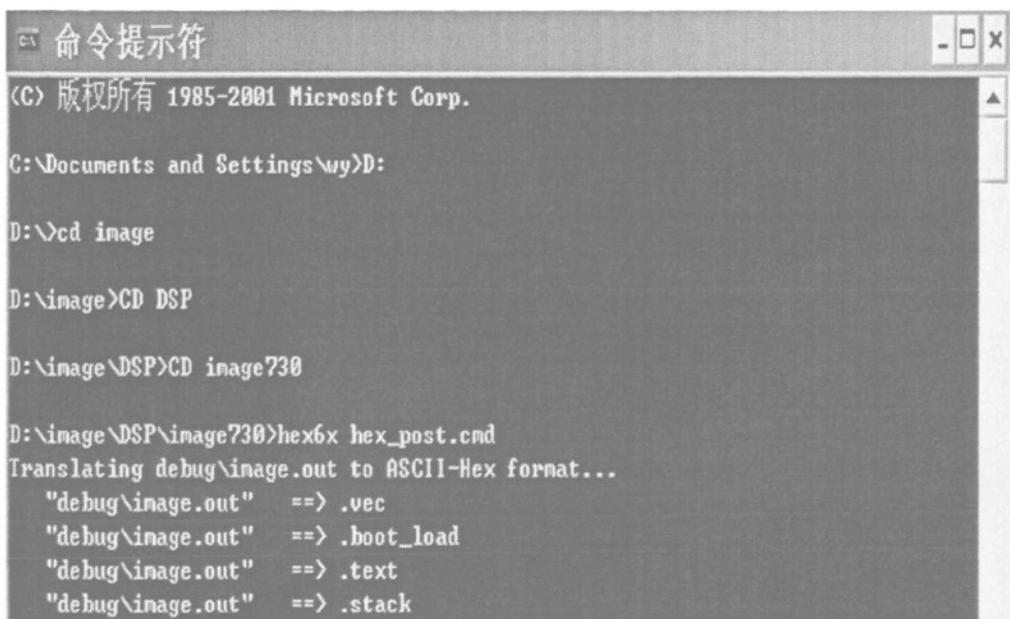
```
}
```

5.2.4 FLASH 的烧写流程

把代码等写入 Flash 的办法大体上可分为以下几种: (1)使用通用烧写器写入。(2)使用 CCS 中自带的 FlashBurn 工具。(3)用户自己编写烧写 Flash 的程序, 由 DSP 将内存映像写入 Flash。其中, 使用通用烧写器烧写需要将内存映像转换为二进制或十六进制格式的文件, 而且要求 Flash 器件是可插拔封装的。这将导致器件的体积较大, 给用户的设计带来不便。使用 TI 公司提供的 FlashBurn 工具的好处在于使用较为直观。FlashBurn 工具提供的图形界面可以方便地对 Flash 执行擦除、编程和查看内容等操作^[39]。

课题用使用 CCS 中自带的 FlashBurn 工具烧写 FLASH。FlashBurn 工具不能识别.out 文件, 只接受.hex 的十六进制文件。因此, 需要将.out 文件转换为.hex 文件。这个转换的工具就是 TI 公司提供的 Hex6x.exe 工具。转换过程的同时, 需要一个 cmd 文件即上文中的 convert.cmd 文件指定作为输入的.out

文件,输出的.hex 文件的格式。制作十六进制.hex 格式文件的过程为启动“命令提示符”:单击“开始”菜单、“程序”、“附件”、“命令提示符”。输入命令进行如下图 5.7 所示:



```
命令提示符
(C) 版权所有 1985-2001 Microsoft Corp.
C:\Documents and Settings\wy>D:
D:\>cd image
D:\image>CD DSP
D:\image\DSP>CD image730
D:\image\DSP\image730>hex6x hex_post.cmd
Translating debug\image.out to ASCII-Hex format...
"debug\image.out" ==> .vec
"debug\image.out" ==> .boot_load
"debug\image.out" ==> .text
"debug\image.out" ==> .stack
```

图 5.7 命令提示符

这样在 D:\image\DSP\image730 目录中生成了 image.hex 文件,下面可利用这个文件烧写到系统基板上的 Flash,以实现自启动功能。

在 CCS 的“Tools”菜单中选择“FlashBurn”项会出现图 5.8 所示的 Flash 烧写界面。先擦除 Flash,然后在相应空格填写文件后,选择菜单“Program”中“Program Flash”烧写程序。烧写完后可通过菜单“Program”中“Show Memory...”检查烧写结果。图 5.9 为 Program 菜单选项。

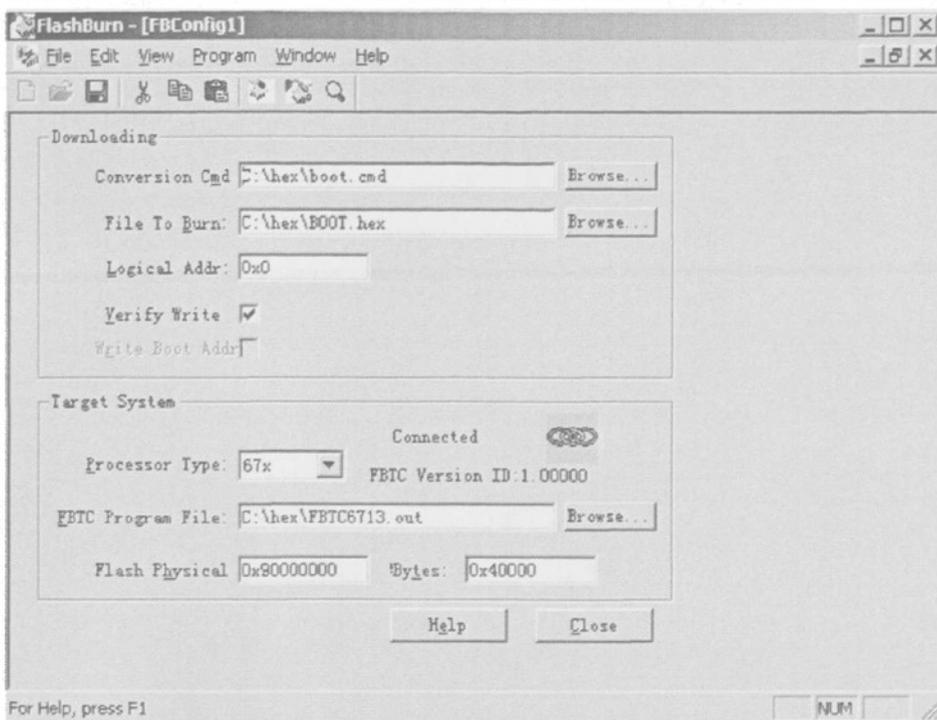


图 5.8 Flash 烧写界面

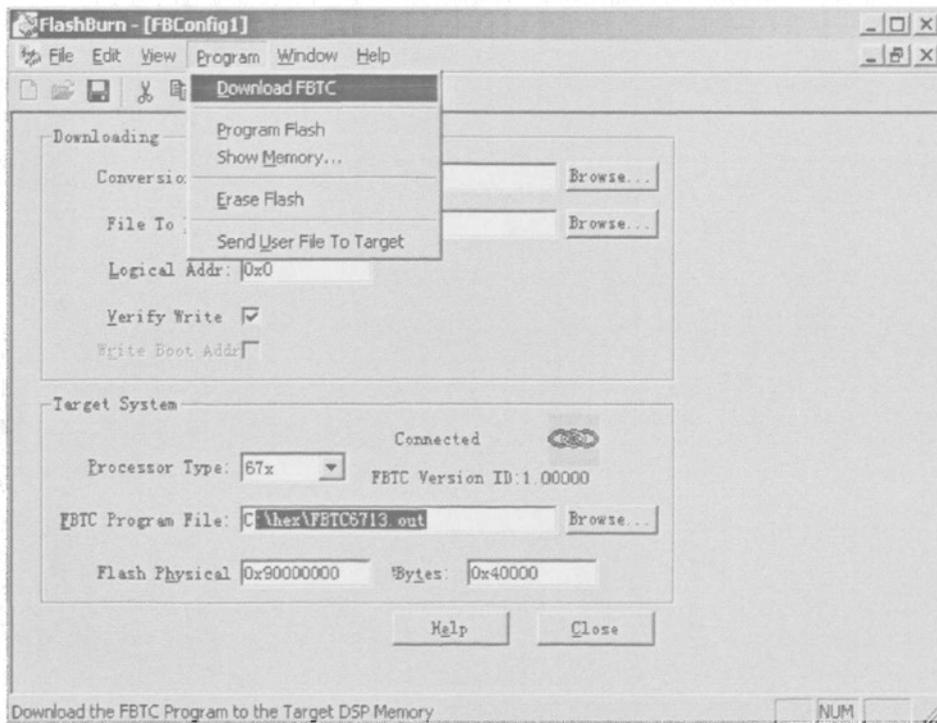


图 5.9 Program 菜单选项

5.3 本章小结

本章分析了系统的硬件、软件调试及系统的自举加载方法和流程。系统硬件调试按模块分别叙述调试方法；软件调试介绍了 DSP 开发环境 CCS 及软件开发流程。DSP 的启动加载有三种方式，本章较为详细的说明了常用的 EMIF 加载方式。

第 6 章 图像处理算法实例

完成了系统本身的硬件架构。图像经过采集、传输后，最后的处理都交由软件系统完成。软件系统的设计是为了更好的存储和管理采集到的图像数据，由于采集的图像可能存在一些曝光不足、模糊等现象，软件系统要能够对图像进行基本的处理。同时从实际要求出发，软件处理系统还应该能够辅助处理和统计采集到的图像中的信息^[41]。本章设计了几个图像处理的应用实验。

6.1 二值化

图像二值化是用灰度变换来研究灰度图像的一种常用的方法，即设定某一阈值Threshold，用Threshold将灰度图像的数据分成两部分：大于阈值的像素群和小于阈值的像素群。例如输入灰度图像函数为 $f(x,y)$ ，输出二值图像函数为 $g(x,y)$ 则

$$g(x,y) = \begin{cases} A & f(x,y) \geq \text{Threshold} \\ 0 & f(x,y) < \text{Threshold} \end{cases} \quad (6-1)$$

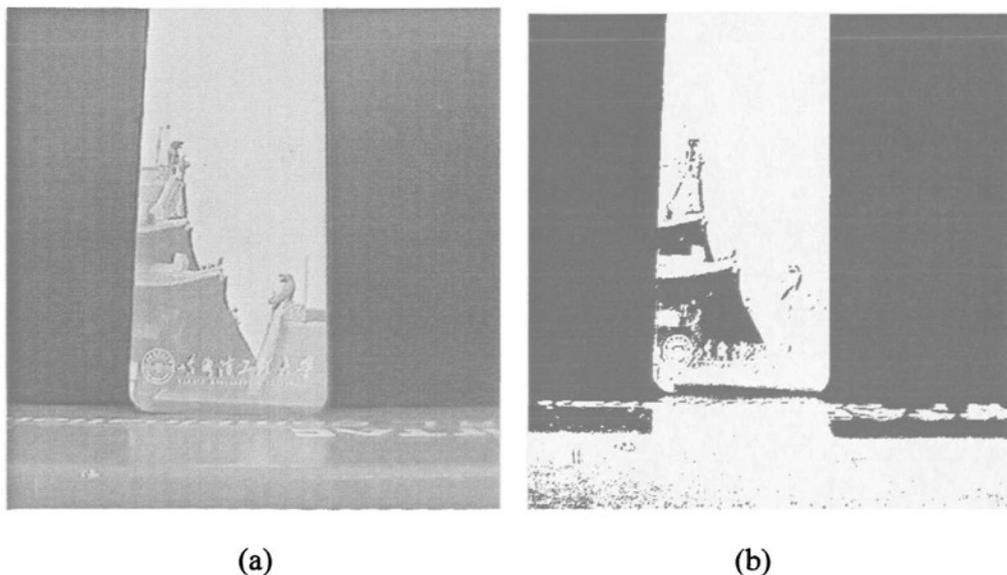


图 6.1 图像二值化处理效果

通过求解阈值Threshold，从而把灰度图像 $f(x,y)$ 分成特征物和背景两个

区域。阈值是把图像和背景区分开的标尺，选取适当的阈值就是既要尽可能保存图像信息，又要尽可能减少背景和噪声的干扰，这是选择阈值的原则^{[42][43]}。

本实例中用的是固定阈值法，对于灰度图像 $f(x,y)$ ，把小于 105 阈值的像素 $f(x,y)$ 置位 0，大于此阈值的像素 $f(x,y)$ 变换为 1。图 6.1 示出图像二值化处理的效果，其中(a)是待处理图像，(b)是处理后的图像。

6.2 数字图像边缘检测(SOBEL 算子)

边缘是图像最基本的特征^[44]。所谓边缘是指图像周围像素灰度有阶跃变化或屋顶状变化的像素的集合，它存在于目标与背景、目标与目标、区域与区域，基元与基元之间。边缘具有方向和幅度两个特征，沿边缘走向，像素值变化比较平缓；垂直于边缘走向，像素值变化比较剧烈，可能呈现阶跃状，也可能呈现斜坡状。因此，边缘可以分为两种：一种为阶跃性边缘，它两边的像素灰度值有着明显的不同；另一种为屋顶状边缘，它位于灰度值从增加到减少的变化转折点。对于阶跃性边缘，二阶方向导数在边缘处呈零交叉；而对于屋顶状边缘，二阶方向导数在边缘处取极值。

经典的边缘检测方法是对原始图像中像素的某小邻域来构造边缘检测算子。常用的边缘检测方法有Roberts算子、Sobel算子、Prewitt算子、Kirsch算子、Laplacian算子、LOG算子、Canny算子等^[45]。

Sobel 算子是边缘检测算子中的一种。Sobel 算子的表达式：

$$s = \sqrt{s_x^2 + s_y^2} \quad (6-2)$$

$$s_x = [f(i-1, j-1) + 2f(i, j-1) + f(i+1, j-1)] - [f(i-1, j+1) + 2f(i, j+1) + f(i+1, j+1)] \quad (6-3)$$

$$s_y = [f(i+1, j-1) + 2f(i+1, j) + f(i+1, j+1)] - [f(i-1, j-1) + 2f(i-1, j) + f(i-1, j+1)] \quad (6-4)$$

用模板表示，即

$$s_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad s_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (6-5)$$

为了简化计算,可用 $s = |s_x| + |s_y|$ 来代替上式,从而得到锐化的图像。Sobel 算子有以下两个特点:

- 1、由于引入了平均因素,因而对图像中的随机噪声有一定的平滑作用;
- 2、由于它是相隔两行或两列之差分,故边缘两侧的元素得到了增强,边缘显得粗而亮。

本系统利用 Sobel 算子对图像进行边缘检测。图 6.2 示出图像边缘检测处理的效果,其中(a)是待处理图像,(b)是处理后的图像。

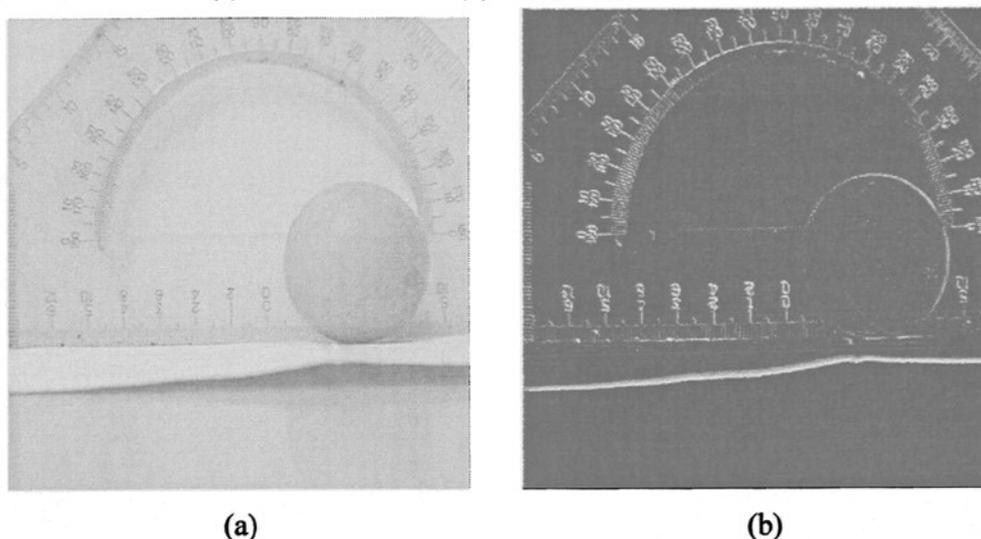


图 6.2 图像边缘检测处理效果

6.3 数字图像锐化(LAPLACE 算子)

图像锐化的目的是使模糊的图像变的更加清晰起来,便于提取物体特征进而对物体进行识别和分析。

图像模糊实质就是图像受到平均或积分运算造成的,因此可以对图像进行逆运算如微分运算来使图像清晰化。从频谱角度来分析,图像模糊的实质是其高频分量被衰减,因而可以通过高通滤波操作来清晰图像^{[46][47]}。但要注意,能够进行锐化处理的图像必须有较高的信噪比,否则锐化后图像信噪比反而更低,从而使噪声的增加得比信号还要多,因此一般是先去除或减轻噪声后再进行锐化处理。

图像锐化一般有两种方法:一种是微分法,另一种是高通滤波法。拉普拉斯锐化法是属于常用的一种微分锐化方法。

拉普拉斯(Laplacian)算子是偏导数运算的线性组合,而且是一种各向同性的线性运算。对图像 $f(x,y)$ 其 Laplacian 运算为:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (6-6)$$

对于离散数字图像 $f(x,y)$ 其一阶偏导数为:

$$\begin{cases} \frac{\partial f(i,j)}{\partial x} = \Delta_x f(i,j) = f(i,j) - f(i-1,j) \\ \frac{\partial f(i,j)}{\partial y} = \Delta_y f(i,j) = f(i,j) - f(i,j-1) \end{cases} \quad (6-7)$$

则其二阶偏导数为:

$$\begin{cases} \frac{\partial^2 f(i,j)}{\partial x^2} = \Delta_x f(i+1,j) - \Delta_x f(i,j) = f(i+1,j) + f(i-1,j) - 2f(i,j) \\ \frac{\partial^2 f(i,j)}{\partial y^2} = \Delta_y f(i,j+1) - \Delta_y f(i,j) = f(i,j+1) + f(i,j-1) - 2f(i,j) \end{cases} \quad (6-8)$$

所以,拉普拉斯算子 $\nabla^2 f$ 为:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(i-1,j) + f(i+1,j) + f(i,j+1) + f(i,j-1) - 4f(i,j) \quad (6-9)$$

对于扩散现象引起的图像模糊,可以用下式来进行锐化:

$$g(i,j) = f(i,j) - k\tau \nabla^2 f(i,j) \quad (6-10)$$

这里 $k\tau$ 是与扩散效应有关的系数,该系数取值要合理,如果 $k\tau$ 过大,图像轮廓边缘会产生过冲;反之如果 $k\tau$ 过小,锐化效果就不明显。

如果令 $k\tau=1$,则变换公式为:

$$g(i,j) = 5f(i,j) - f(i-1,j) - f(i+1,j) - f(i,j+1) - f(i,j-1) \quad (6-11)$$

用模板表示如下:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (6-12)$$

本系统利用 Laplace 算子对图像增强的例子如图 6.3 所示,其中(a)是待处

理图像，(b)是处理后的图像。

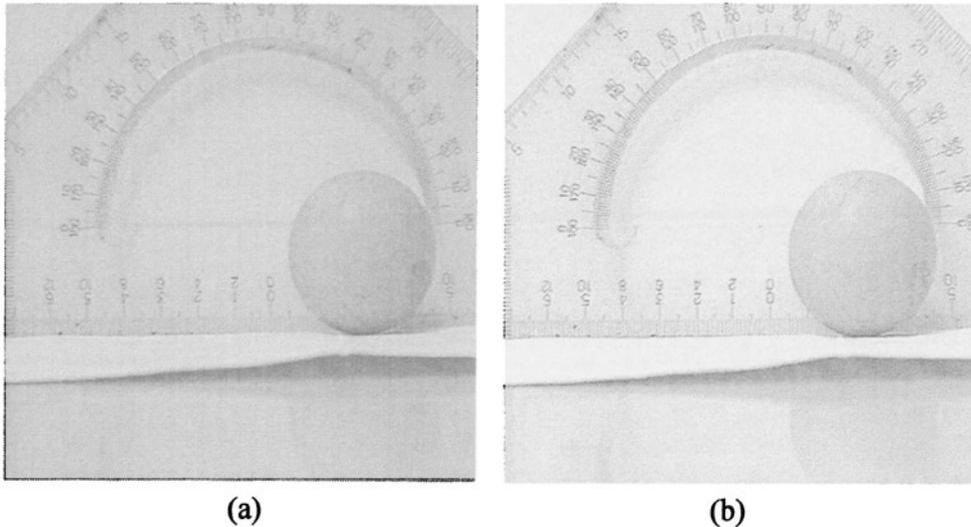


图 6.3 图像锐化处理效果

6.4 图像平滑处理

实际获取的图像一般都因为收到干扰而含有噪声成分。引起噪声的原因主要是：敏感元件灵敏度不均匀、有串入通道中的干扰和数字化装置的量化噪声等。图像平滑处理的作用就在于消除或尽量消弱这些噪声，它是改善图像质量，以便人们对图像进行处理和识别的主要技术手段之一^{[48][49]}。

一般来说，对图像进行平滑处理，可在空域或频域上进行^[50]。由于图像空域平滑处理方法概念直观，便于实现，计算速度快，结果也比较令人满意，因而在图像平滑处理中占有重要地位。

邻域平均法是图像平滑处理中空域上进行的一种方法。这种方法的基本思想是用几个像素灰度的平均值来代替每个像素的灰度。假定有一幅 $N \times N$ 个像素的图像 $f(x, y)$ ，平滑处理后得到一幅图像 $g(x, y)$ 。 $g(x, y)$ 由下式决定：

$$g(x, y) = \frac{1}{M} \sum_{(m, n) \in S} f(m, n) \quad (6-13)$$

式中 $x, y = 0, 1, 2, \dots, N-1$, S 是 (x, y) 点邻域中点的坐标的集合，但其中不包括 (x, y) 点， M 是集合内坐标点的总数。式(6-13)说明，平滑的图像 $g(x, y)$ 中的每个像素的灰度值均由包含在 (x, y) 的预定邻域中的 $f(x, y)$ 的几个像素的灰度值的平均值来决定。

本程序中,用 3×3 均值滤波器对图像平滑处理。当灰度图像在像素 $f(x,y)$ 周围8个像素的平均灰度为 a 时,按照下面方法处理像素 $f(x,y)$;

若 $|f(x,y)-a| < \varepsilon$, 则 $f(x,y) \leftarrow a$;

若 $|f(x,y)-a| \geq \varepsilon$, 则不变。

图 6.4 示出图像用 3×3 均值滤波器平滑处理的效果,其中(a)是待处理图像,(b)是处理后的图像。

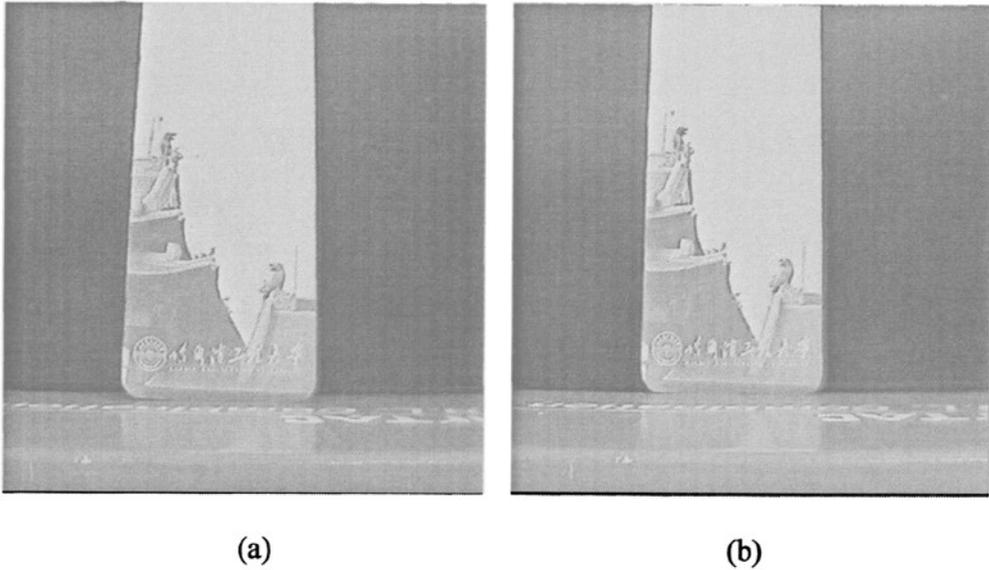


图 6.4 图像平滑处理效果

6.5 图像的外接框检出

本程序先将待处理的图像二值化,然后根据二值化后图像检出图形中黑色区域的外接框。外接框和待检出的目标图形的四边相互平行。所得结果是外接框的上边位置 yup , 下边位置 $ydown$, 左边位置 $xleft$ 和右边位置 $xright$ 。图像处理后可得到外接框的位置坐标,从而可得到目标图像的位置坐标。图 6.5 示出目标图像的外接框检出效果。

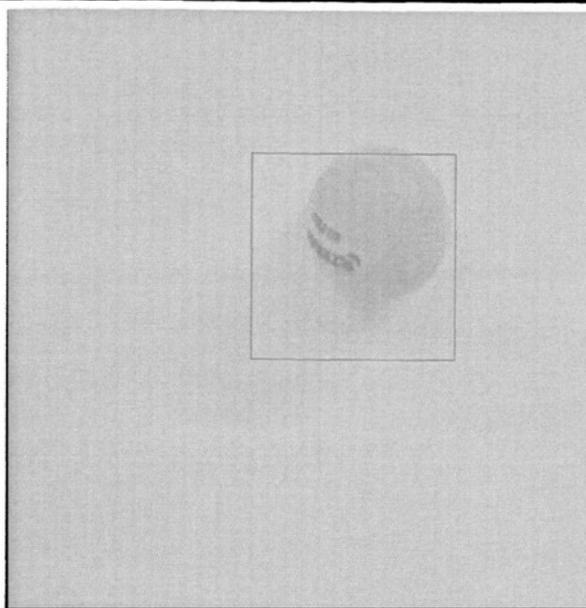


图 6.5 图像外接框检出处理效果

6.6 本章小结

本章设计了几个图像处理的应用实验。这些图像处理算法实例均是由 DSP 处理得到的。本章算法实例包括：二值化、边缘检测、锐化、平滑处理和对目标图像的外接框的检出。这些图像处理方式可由多种方法实现，本章是选用其中一种方法实现的，其它算法的实现均可在 DSP 中完成。最后对这些算法的程序进行了性能分析。

结 论

论文重点研究了图像采集系统硬件的设计与实现，完成了板上各功能部件的调试，在 DSP 上实现了图像采集与处理算法。根据论文的工作，我们得到以下结论：

1、在系统的实现上，基于 DSP 的图像采集系统采用了 DSP+CPLD 的架构。采用这种架构后图像采集与处理平台具有体积小、功耗低、便于调试的优点，并且大大的降低了系统的实现成本和复杂度。论文实现了通用的嵌入式图像采集与处理平台。

2、从系统的软硬件配置来看，由于 DSP 的程序可以在线调试和擦写，使得图像处理的算法软件很容易升级。系统设计中所有的逻辑和时序匹配均采用 CPLD 来完成。由于 CPLD 具有可编程的特点，从而增强了硬件的可重构性与灵活性。

3、系统采用了性价比很高的 TMS320C6713 DSP 作为图像处理的核心处理器，大大增强了系统的处理能力。通过对几种常用图像处理算法的实验结果分析后可知，在达到预期处理效果的前提下，系统能够满足系统实时性的要求。

虽然我们设计的系统具有较好的性能，但是由于系统涉及的技术要点比较多以及时间方面的问题，系统仍然存在一些问题需要完善：

1、彩色图像的采集需要完善。PCB 板上已预留了色差信号的缓存芯片。只需要将 PCB 板上的跳线连接好进行 FIFO 的位宽扩展，就可以将色差信号引入，从而实现灰度信号和色差信号在 DSP 中的组合。

2、进一步可改进系统的架构。采用 DSP 和 FPGA 的方式，在 FPGA 内部嵌入软核。可以用 FPGA 单独完成图像采集的功能，并进行一些基本的图像处理，如将图像从 YUV 格式转换为 RGB 格式。这种设计方式可使 DSP 专注于高级的图像处理算法的实现。

3、改进图像的传输方式，目前的图像传输方式是采用串口通信的方式，速度比较慢，不利于调试。可以考虑采用 USB 传输或是加入图像编码器直接进行 VGA 显示。

4、对于某些实时性要求高的图像处理算法，需采用 C 语言加汇编语言混合编程的方式，从而缩短程序执行时间。此外，采用 TI 为 TMS320C6000 DSP 提供的 Image Library，可以大大缩短开发周期，提高系统的可靠性。

5、在存储架构上，用乒乓存储的方式。DSP 的 EDMA 控制器负责数据搬运，CPU 负责图像处理。两个过程在乒乓存储器中交叉进行。可以实现图像采集与处理连续不间断的进行。

参考文献

- [1] 徐向丽, 陈岳林, 韦必忠. 视频分析技术的研究和应用. 现代计算机. 2008, 07: 66-69 页
- [2] 苏光大. 微机图像处理系统. 清华大学出版社, 2000
- [3] 王念旭. DSP 基础与应用系统设计. 北京航空航天大学出版社, 2002
- [4] 季昱, 林俊超, 余本喜. DSP 嵌入式应用系统开发典型实例. 中国电力出版社, 2007
- [5] 杨胜龙, 包兴, 姚素英. SAA7111A 在视频显示系统中的应用. 电子测量技术. 2007, 30(9): 112-114 页
- [6] 张海英, 胡冰, 赵新田, 吴升艳. 用 SAA7111A 设计模拟视频转换接口. 电子应用技术. 2003, 29(1): 52-53 页
- [7] 张克满, 何格夫. 用虚拟 IIC 总线技术实现 SAA7111 的初始化. 国外电子元器件. 2005, 01: 26-29 页
- [8] 秦承虎, 王金刚. 基于 IIC 总线的图像传感器接口设计. 电子测量技术. 2006, 29(3): 169-170 页
- [9] TMS320C6000 DSP Multichannel Buffered Serial Port (McBSP). Texas Instruments, 2005
- [10] 龙娟, 杨银堂, 乐立鹏. FIFO 存储电路的设计与实现. 微电子学与计算机. 2006, 23(12): 115-117 页
- [11] 顾蕊. TMS320C67 系列 EMIF 与异步 FIFO 存储器的接口设计. 电子工程师. 2005, 31(5): 53-55 页
- [12] 田岩涛, 常丹华, 潘普丰, 张彦. 用基于 DSP 的线阵 CCD 实现二维图像信号采集的系统设计. 应用光学. 2005, 26(6): 8-11 页
- [13] TMS320C6000 DSP Inter-Integrated Circuit (I²C) Module Reference Guide. Texas Instruments, 2002
- [14] 李超纯. 基于 DSP 与 CPLD 的实时图像采集与处理系统的研究. 华中师范大学硕士学位论文. 2007
- [15] 黄素贞. 一种基于 DSP 和 CPLD 的图像采集处理系统的研究. 山东大学

硕士学位论文. 2005

- [16] 李洪伟, 袁斯华. 基于 Quartus II 的 FPGA/CPLD 的设计. 电子工业出版社, 2006
- [17] 林晓春, 过润春. Altera EPLD 系列器件 EPM7128 性能特点及应用. 国外电子元器件. 1998, 11: 42-47 页
- [18] TMS320C6000 DSP Software-Programmable Phase-Locked Loop(PLL) Controller. Texas Instruments, 2006
- [19] TMS320C6000 DSP External Memory Interface(EMIF). Texas Instruments, 2007
- [20] 刘伟, 焦淑红, 刘洋. 基于 SDRAM 的 DSP 片外高速海量存储系统设计. 微型电脑应用. 2006, 22(1): 21-24 页
- [21] 柏翰. 处理器外接 SDRAM 的控制技术介绍. 电子与电脑. 2006, 4: 127-129 页
- [22] 康乐峰, 王和国, 刘书明. DSP 和 Flash 接口技术的实现. 国外电子元器件. 2008, 7: 75-76 页
- [23] TMS320C6000 DSP Enhanced Direct Memory Access(EDMA). Texas Instruments, 2006
- [24] 刘伟生, 孙秋柏, 吴庆洪, 李健兵. DSP 在图像高速采集中的应用. 自动化技术与应用. 2005, 24(12): 22-24 页
- [25] 周小平, 牟永敏. TL16C550 芯片在串行通信中的应用. 北京机械工业学院学报. 2008, 23(2): 49-52 页
- [26] 付保川, 吴志强. 异步串行收发器 16C550 在远程通信中的应用. 苏州城建环保学院学报, 2002, 15(2): 82-87 页
- [27] Texas Instruments Inc. TL16C550 Datasheet[DB/OL]. 1998[2007-01-05]. <http://focus.ti.com.cn/cn/lit/ds/symlink/tl16c550c.pdf>
- [28] 史建民, 黄有方, 嘉红霞. RS-232 串口设备远程通讯功能的实现. 起重运载机械. 2003(6): 30-26 页
- [29] 彭启琮. DSP 集成开发环境. 电子工业出版社, 2004
- [30] 张勇. C/C++语言硬件程序设计. 西安电子科技大学出版社, 2003
- [31] Creating a Second-Level Bootloader for FLASH Bootloading on

- TMS320C6000 Platform With Code Composer Studio . Texas Instruments, 2006
- [32] Texas Instruments Incorporated. TMS320C6711, TMS320C6711B, TMS320C6711C FLOATING POINT DIGITAL SIGNAL PROCESSOR. [Http://www.ti.com](http://www.ti.com),2006
- [33] Texas Instruments Incorporated.TMS320C6000 EMIF to External Flash. Texas Instruments Application Report,2002
- [34] 公茂忠, 刘汉奎, 徐殿国. Flash 存储器的在系统编程及其在 DSP 系统中的应用. 电子技术应用. 2002.3
- [35] TMS320C6000 Assembly Language Tools User's Guide. Texas Instruments,2004
- [36] 王凤臣, 乔卫民, 李庆瑞. TI TMS320C6713 DSP 板 FlashBurn 程序设计. 微计算机信息. 2006, 11
- [37] 朱望纯, 胡汉武, 李智. DSP+FLASH 的二次程序加载实现方法. 理论与方法. 2008, 27(4): 14-20 页
- [38] 左韬, 潘练. DSP 通过 FLASH 并行加载的分析和实例. 微计算机信息. 2007(2): 184-186 页
- [39] 白汉斌, 刘峰, 张明敏. 基于 TMS320C6000 系列 DSP 的 Flash 编程方法. 电子测量技术, 2006, 29(6): 107-110 页
- [40] Rafeal C.Gonzalez,Richard E.Woods.Digital Image Processing.Publishing Housse of Eletronics Industry.BEIJING.2004
- [41] 王勇智. 数字图像的二值化处理技术研究. 湖南理工学院学报. 2005, 18(1): 31-33 页
- [42] 马驰, 张红云, 苗夺谦, 张学东. 改进的多阈值动态二值化方法. 计算机工程. 2006, 32(6): 203-208 页
- [43] 雷丽珍. 数字图像边缘检测方法的探讨. 测绘通报. 2006, 3: 40-42 页
- [44] 张小琳. 图像边缘检测技术综述. 高能量密度物理. 2007, 1: 37-40 页
- [45] 李弼程, 等编著. 智能图像处理技术. 电子工业出版社, 2004
- [46] 张娜. 图像增强技术的研究. 计算机仿真, 2007, 24(1): 192-195 页
- [47] 刘伍丰, 何旭春, 徐杨, 乔卫民. 数据平滑处理的编程. 微计算机信息,

2007, 23(3): 209-211 页

- [48] 朱秀昌, 刘峰, 胡栋. 数字图像处理与图像通信. 北京邮电大学出版社, 2002
- [49] 李俊生. 一种图像平滑处理方法的空域算法与频域分析. 常州工学院学报. 2004, 17(2): 30-33 页

攻读硕士学位期间发表的论文和取得的科研成果

- [1] 武颖, 王洪波, 叶秀芬. “基于 TMS320C6713 嵌入式图像采集系统设计与实现”, 2008 海峡两岸科技与经济论坛, 2008: 201-204 页

致 谢

首先感谢我的导师叶秀芬教授，在我研究生阶段对我的教导和在研究生课题上给予的悉心指导和支持。叶老师治学严谨、工程经验丰富对我的研究生阶段学习和研究有很大的帮助。这里对导师付出的辛勤劳动和提供的帮助表示衷心地感谢。

同时感谢郭书祥教授和自动化学院给我们提供的良好的学习工作环境。感谢自动化学院的各位老师毕业设计过程中给予我的指导和帮助。感谢微型仿生机器人实验室全体教师，给我提供了这样良好的硬件和软件条件，使我学到了很多知识和技能。感谢微型仿生机器人实验室 2006 级全体同学和师弟、师妹们。在整个论文的写作过程中，我们之间的互相帮助、互相交流使我受益匪浅。

我要深切地感谢我的父母，是他们对我的教导和养育才使我有机会去追求理想，接受教育。谢谢你们无私的爱和奉献。

感谢我的男朋友王洪波，在我的研究生阶段一直在帮助我、鼓励我、支持我，给予了我莫大的帮助。

最后要感谢在本课题所涉及到的研究方向上，所有走在前面的人以及本论文所引用参考文献的原作者，你们的研究成果对本论文的完成都有很重要的指导意义。