

本科毕业论文（设计）

题 目： 控制 EDA 课程设计

毕业设计（论文）原创性声明和使用授权说明

原创性声明

本人郑重承诺：所呈交的毕业设计（论文），是我个人在指导教师的指导下进行的研究工作及取得的成果。尽我所知，除文中特别加以标注和致谢的地方外，不包含其他人或组织已经发表或公布过的研究成果，也不包含我为获得_____及其它教育机构的学位或学历而使用过的材料。对本研究提供过帮助和做出过贡献的个人或集体，均已在文中作了明确的说明并表示了谢意。

作者 签 名：_____ 日 期：_____

指导教师签名：_____ 日 期：_____

使用授权说明

本人完全了解_____大学关于收集、保存、使用毕业设计（论文）的规定，即：按照学校要求提交毕业设计（论文）的印刷本和电子版本；学校有权保留毕业设计（论文）的印刷本和电子版，并提供目录检索与阅览服务；学校可以采用影印、缩印、数字化或其它复制手段保存论文；在不以赢利为目的前提下，学校可以公布论文的部分或全部内容。

作者签名：_____ 日 期：_____

学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。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后果由本人承担。

作者签名： 日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权_____大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

涉密论文按学校规定处理。

作者签名： 日期： 年 月 日

导师签名： 日期： 年 月 日

注 意 事 项

1. 设计（论文）的内容包括：

- 1) 封面（按教务处制定的标准封面格式制作）
- 2) 原创性声明
- 3) 中文摘要（300 字左右）、关键词
- 4) 外文摘要、关键词
- 5) 目次页（附件不统一编入）
- 6) 论文主体部分：引言（或绪论）、正文、结论
- 7) 参考文献
- 8) 致谢
- 9) 附录（对论文支持必要时）

2. 论文字数要求：理工类设计（论文）正文字数不少于 1 万字（不包括图纸、程序清单等），文科类论文正文字数不少于 1.2 万字。

3. 附件包括：任务书、开题报告、外文译文、译文原文（复印件）。

4. 文字、图表要求：

1) 文字通顺，语言流畅，书写字迹工整，打印字体及大小符合要求，无错别字，不准请他人代写

2) 工程设计类题目的图纸，要求部分用尺规绘制，部分用计算机绘制，所有图纸应符合国家技术标准规范。图表整洁，布局合理，文字注释必须使用工程字书写，不准用徒手画

3) 毕业论文须用 A4 单面打印，论文 50 页以上的双面打印

4) 图表应绘制于无格子的页面上

5) 软件工程类课题应有程序清单，并提供电子文档

5. 装订顺序

1) 设计（论文）

2) 附件：按照任务书、开题报告、外文译文、译文原文（复印件）次序装

订

指导教师评阅书

指导教师评价:

一、撰写(设计)过程

1、学生在论文(设计)过程中的治学态度、工作精神

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、学生掌握专业知识、技能的扎实程度

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

3、学生综合运用所学知识和专业技能分析和解决问题的能力

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

4、研究方法的科学性;技术线路的可行性;设计方案的合理性

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

5、完成毕业论文(设计)期间的出勤情况

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

二、论文(设计)质量

1、论文(设计)的整体结构是否符合撰写规范?

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、是否完成指定的论文(设计)任务(包括装订及附件)?

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

三、论文(设计)水平

1、论文(设计)的理论意义或对解决实际问题的指导意义

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、论文的观念是否有新意?设计是否有创意?

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

3、论文(设计说明书)所体现的整体水平

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

建议成绩: ☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

(在所选等级前的□内画“√”)

指导教师:

(签名)

单位:

(盖章)

年 月 日

评阅教师评阅书

评阅教师评价：

一、论文（设计）质量

1、论文（设计）的整体结构是否符合撰写规范？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、是否完成指定的论文（设计）任务（包括装订及附件）？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

二、论文（设计）水平

1、论文（设计）的理论意义或对解决实际问题的指导意义

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、论文的观念是否有新意？设计是否有创意？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

3、论文（设计说明书）所体现的整体水平

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

建议成绩： ☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

（在所选等级前的□内画“√”）

评阅教师：

（签名）

单位：

（盖章）

年 月 日

教研室（或答辩小组）及教学系意见

教研室（或答辩小组）评价：

一、答辩过程

1、毕业论文（设计）的基本要点和见解的叙述情况

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、对答辩问题的反应、理解、表达情况

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

3、学生答辩过程中的精神状态

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

二、论文（设计）质量

1、论文（设计）的整体结构是否符合撰写规范？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、是否完成指定的论文（设计）任务（包括装订及附件）？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

三、论文（设计）水平

1、论文（设计）的理论意义或对解决实际问题的指导意义

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

2、论文的观念是否有新意？设计是否有创意？

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

3、论文（设计说明书）所体现的整体水平

☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

评定成绩：☐ 优 ☐ 良 ☐ 中 ☐ 及格 ☐ 不及格

教研室主任（或答辩小组组长）：

（签名）

年 月 日

教学系意见：

系主任：（签名）

年 月 日

原创力文档

max.book118.com

预览与源文档一致, 下载高清无水印

模拟量数据采集

摘要：本次课程设计是一种基于 FPGA 技术的模拟数据采集的设计方案，本文所提出的一种基于 FPGA 技术的模拟量采集，利用 FPGA 的 I/O 端口多，且可以自由编程支配、定义其功能的特点，配以 VHDL 编写的 FPGA 内部执行软件，能很好地解决采集的信号路数多的问题。因为用 VHDL 编写的执行软件内部对每组数字量是按并行处理的，而且 FPGA 硬件的速度是 ns 级的，因此本设计能实时地、快速地监测信号量的变化。本设计鉴于 FPGA 和 VHDL 语言自身的特点，本设计具有较好的扩展性，在监测和控制系统中也具有一定的通用性。系统主要包括：FPGA 芯片区、多路选择与 A/D 采样电路、ADC0809 芯片等几部分。

关键词：FPG； A/D 采样； 模拟量采集； VHDL 语言设计

Analog data acquisition technology

Abstract: The curriculum design is an FPGA-based analog data acquisition technology in the design of the proposed FPGA-based analog technology, digital acquisition and processing system using FPGA's I / O ports and more, and be free programming control, the definition of the functions of the characteristics of the FPGA with VHDL within the preparation of the implementation of software, can be collected very good solution to the problem of multi-signal approach. Because VHDL implementation of the software prepared by the internal volume of the set of figures is based on parallel processing, and FPGA hardware ns speed class, so the design in real time, monitoring of fast changes in the volume signal. In view of the FPGA and VHDL design language of its own characteristics, the design of better scalability, in the monitoring and control system also has some versatility. System mainly includes: FPGA chip area, multi-channel selection and A / D sampling circuit, ADC0809 chip such as several parts.

Keywords: FPGA; A / D sampling; Analog Acquisition; VHDL Language Acquisition Design

目 录

1 绪论	1
2 课程设计内容及要求	2
2.1 课题设计内容	2
2.2 课题设计要求	2
3 课程设计思路及原理	3
3.1 课程设计总体方案	3
3.2 课程设计原理	4
4 课程设计步骤及过程	7
4.1 ADC0809 的应用	7
4.2 DAC0832 的应用	9
4.3 实验仿真图形	12
5 课程设计心得体会	13
参考文献	14
致 谢	15
附录	16

原创力文档

max.book118.com

预览与源文档一致,下载高清无水印

1 绪论

课程设计是课程教学中的重要环节，不仅是对所学课程的一种实践性总结和应用，而往往是对多门课程的综合性与应用。EDA 是英文“Electronics Design Automation（电子设计自动化）”的缩写，EDA 技术是 90 年代迅速发展起来的，是现代电子设计的最新技术潮流，是综合现代电子技术和计算机技术的最新研究成果，是从事电子线路设计与分析的一门技术，包括电子线路的设计、计算机模拟仿真和电路分析、印制电路板的自动化设计三个方面的内容。EDA 课程设计要求学生根据所学知识设计出一套基于 EDA 的应用系统。它要求学生除了掌握 EDA 的工作原理、接口设计和程序设计之外，还要求学生结合 Protel、电路、模拟电子技术和数字电子技术以及 EDA 等课程对系统进行综合设计。本次课程设计主要应用 VHDL 语言，VHDL 主要用于描述数字系统的结构，行为，功能和接口。除了含有许多具有硬件特征的语句外，VHDL 的语言形式和描述风格与句法是十分类似于一般的计算机高级语言。VHDL 的程序结构特点是将一项工程设计，或称设计实体（可以是一个元件，一个电路模块或一个系统）分成外部（或称可参见部分，及端口）和内部（或称不可视部分），既涉及实体的内部功能和算法完成部分。在对一个设计实体定义了外部界面后，一旦其内部开发完成后，其他的设计就可以直接调用这个实体。这种将设计实体分成内外部分的概念是 VHDL 系统设计的基本点，用 VHDL 编写 FPGA 的内部执行程序，能很好地解决采集的信号路数多的问题。因为用 VHDL 编写的执行软件内部对各组数字量是按并行处理的，而且 FPGA 硬件的速度是 ns 级的，因此本设计能实时地、快速地监测信号量的变化。

2 课程设计及要求

2.1 课题设计内容

本次课程设计的主要旨在通过独立完成一个“模拟量数据采集”的设计，达到对 EDA 技术的熟练掌握，提升对《EDA 技术及应用》课程所学内容的掌握和应用。

在本次课程设计中，使用 Altera 公司的 EPF10K10 系列的 FPGA 芯片，基于实验室现有的 EDA 实验箱，实现“模拟量数据采集”的设计内容。

2.2 课题设计要求

1、由电位器调节的可变电压（0~5V）输入到模/数转换器 ADC0809 的 IN0 通道，利用 FPGA 设计电路控制 ADC0809 对该输入电压进行模/数转换，并将转换的结果（十六进制数）输出至数码管显示。

2、功能键：

（1） 停止

（2） 启动

3、利用 FPGA 设计电路实现对数/模转换器 DAC0832 的驱动，并将模/数转换器 ADC0809 的转换结果（数字量）通过 DAC0832 转换为模拟量的输出。

3 课程设计思路及原理

3.1 课程设计总体方案

3.1.1 设计思路

先将每一个模块分别设计出来，再由软件生成图形，最后用软件连线，定义好每个引脚，最后在试验箱上进行最后的验证

- (1) 设计分频电路 FEN 模块。
- (2) 设计时序控制 AD0809 模块。
- (3) 设计进制转换 DDDF 模块。
- (4) 设计 YM 模块，用于控制数码管的位码。
- (5) 设计 DISP 模块，用于控制数码管的段码。

3.1.2 设计流程图

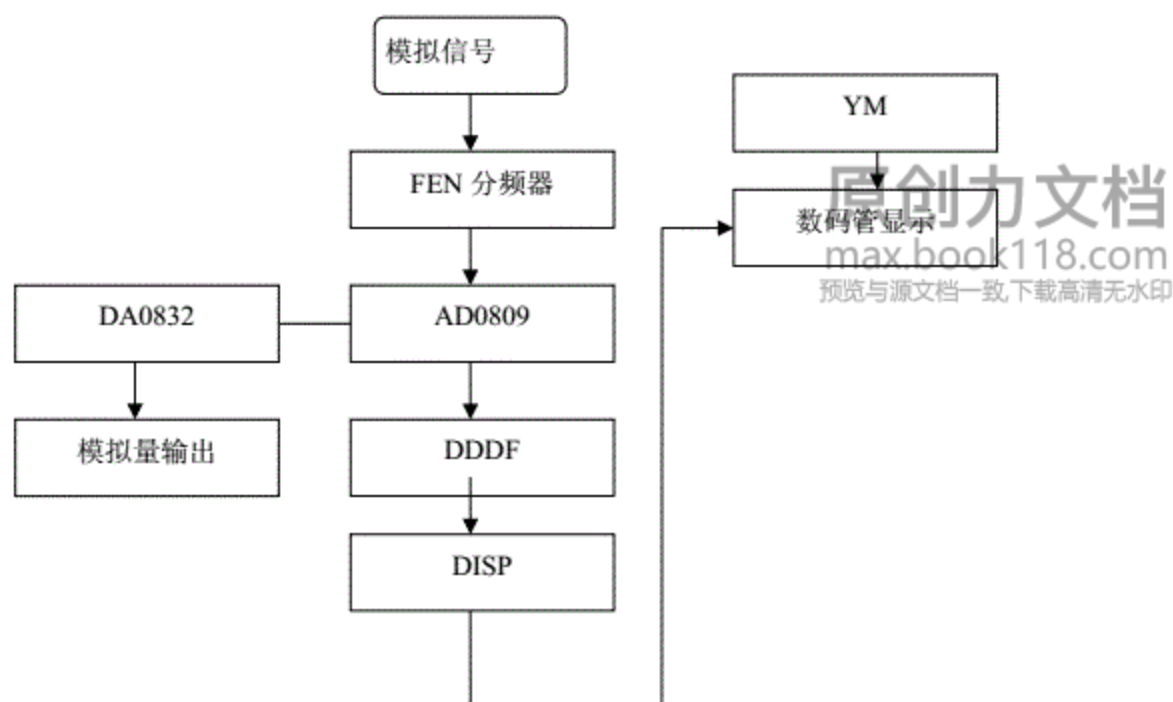


图 3.1 总体设计流程图

3.2 课程设计原理

3.2.1 模拟量采集设计原理

- 1 先通过电位器来调节可变电压
- 2 把调节电压输入到模/数转换器 ADC0809 的 IN0 通道
- 3 利用 FPGA 设计电路控制 ADC0809 对该输入进行模/数转换
- 4 将结果输出到数码管显示
- 5 利用 FPGA 设计电路控制 DAC0832 对该输入进行数/模转换

3.2.2 FPGA 应用原理

FPGA 是英文 Field Programmable Gate Array 的缩写，即现场可编程门阵列，它是在 PAL、GAL、EPLD 等可编程器件的基础上进一步发展的产物。它是作为专用集成电路（ASIC）领域中的一种半定制电路而出现的，既解决了定制电路的不足，又克服了原有可编程器件门电路数有限的缺点。

FPGA 采用了逻辑单元阵列 LCA（Logic Cell Array）这样一个新概念，内部包括可配置逻辑模块 CLB（Configurable Logic Block）、输出输入模块 IOB（Input Output Block）和内部连线（Interconnect）三个部分。

FPGA 的基本特点主要有：

- 1) 采用 FPGA 设计 ASIC 电路，用户不需要投片生产，就能得到合用的芯片。
 - 2) FPGA 可做其它全定制或半定制 ASIC 电路的中试样片。
 - 3) FPGA 内部有丰富的触发器和 I/O 引脚。
 - 4) FPGA 是 ASIC 电路中设计周期最短、开发费用最低、风险最小的器件之一。
 - 5) FPGA 采用高速 CMOS 工艺，功耗低，可以与 CMOS、TTL 电平兼容。
- 可以说，FPGA 芯片是小批量系统提高系统集成度、可靠性的最佳选择之一。

目前 FPGA 的品种很多，有 XILINX 的 XC 系列、TI 公司的 TPC 系列、ALTERA 公司的 FIEX 系列等。

FPGA 是由存放在片内 RAM 中的程序来设置其工作状态的, 因此, 工作时需要对片内的 RAM 进行编程。用户可以根据不同的配置模式, 采用不同的编程方式。

加电时, FPGA 芯片将 EPROM 中数据读入片内编程 RAM 中, 配置完成后, FPGA 进入工作状态。掉电后, FPGA 恢复成白片, 内部逻辑关系消失, 因此, FPGA 能够反复使用。FPGA 的编程无须专用的 FPGA 编程器, 只须用通用的 EPROM、PROM 编程器即可。当需要修改 FPGA 功能时, 只需换一片 EPROM 即可。这样, 同一片 FPGA, 不同的编程数据, 可以产生不同的电路功能。因此, FPGA 的使用非常灵活。

FPGA 有多种配置模式: 并行主模式为一片 FPGA 加一片 EPROM 的方式; 主从模式可以支持一片 PROM 编程多片 FPGA; 串行模式可以采用串行 PROM 编程 FPGA; 外设模式可以将 FPGA 作为微处理器的外设, 由微处理器对其编程。

3.2.3 VHDL 语言

VHDL 语言目前在数字设计领域已为广大设计者所接受, 众多 CAD 厂商纷纷使自己新开发的电子设计软件与 VHDL 兼容, VHDL 语言成了电子设计工程师必须掌握的工具。在目前, CPU 和可编程逻辑器件已经成为数字系统的硬件基础, 从事数字系统的设计必须掌握可编程逻辑器件的设计方法, 而 VHDL 语言作为可编程逻辑器件设计时最重要的输入方法, 为所有可编程逻辑器件厂商所支持。因此要从事可编程逻辑器件的设计工作, 必须掌握 VHDL 语言。

VHDL 的英文 Very-High-Speed Integrated Circuit Hardware Description Language, 诞生于 1982 年。1987 年底, VHDL 被 IEEE 和美国国防部确认为标准硬件描述语言。自 IEEE 公布了 VHDL 的标准版本, IEEE-1076 (简称 87 版) 之后, 各 EDA 公司相继推出了自己的 VHDL 设计环境, 或宣布自己的设计工具可以和 VHDL 接口。此后 VHDL 在电子设计领域得到了广泛的接受, 并逐步取代了原有的非标准的硬件描述语言。1993 年, IEEE 对 VHDL 进行了修订, 从更高的抽象层次和系统描述能力上扩展 VHDL 的内容, 公布了新版本的 VHDL, 即 IEEE 标准的 1076-1993 版本, (简称 93 版)。现在,

VHDL 和 Verilog 作为 IEEE 的工业标准硬件描述语言，又得到众多 EDA 公司的支持，在电子工程领域，已成为事实上的通用硬件描述语言。有专家认为，在新的世纪中，VHDL 于 Verilog 语言将承担起大部分的数字系统设计任务。

VHDL 主要用于描述数字系统的结构，行为，功能和接口。除了含有许多具有硬件特征的语句外，VHDL 的语言形式和描述风格与句法是十分类似于一般的计算机高级语言。VHDL 的程序结构特点是将一项工程设计，或称设计实体

（可以是一个元件，一个电路模块或一个系统）分成外部（或称可是部分，及端口）和内部（或称不可视部分），既涉及实体的内部功能和算法完成部分。在对一个设计实体定义了外部界面后，一旦其内部开发完成后，其他的设计就可以直接调用这个实体。

4 课程设计步骤及过程

4.1 ADC0809 的应用

ADC0809 带有 8 通道多路开关，单片 COMS 器件，采用逐次逼近法进行转换。它的转换时间为 $100\mu\text{s}$ ，分辨率为 8 位，转换误差小于 $\pm\text{LSB}/2$ ，单电源 5V 供电，输入模拟电压范围为 $0\sim 5\text{V}$ ，内部集成了可以锁存控制的 8 路模拟多路开关，输出采用三态输出锁存器，电平与 TTL 电平兼容。

4.1.1 ADC0809 内部结构

ADC0809 内部结构如图 4.1 所示：

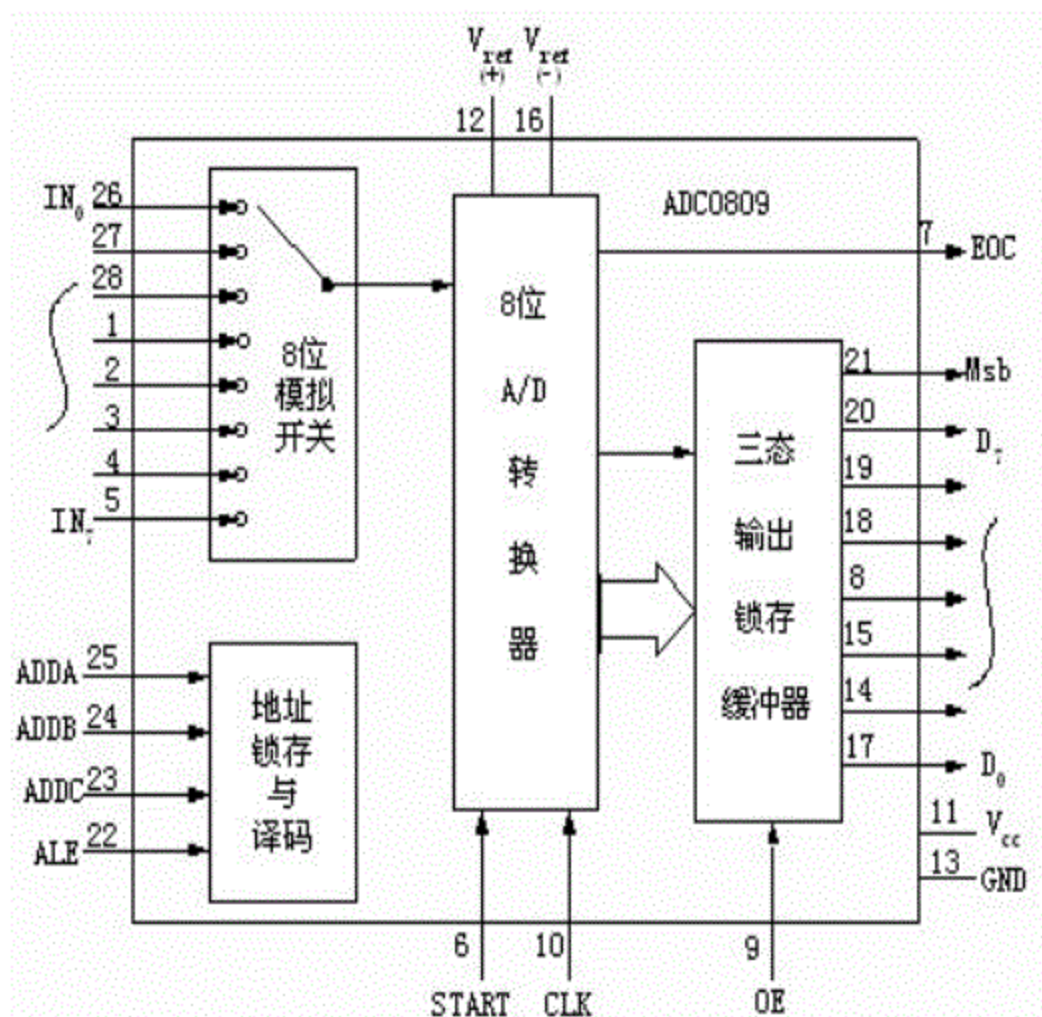


图 4.1 ADC0809 内部结构图

由上图可知, ADC0809 由一个 8 路模拟开关、一个地址锁存与译码器、一个 A/D 转换器和一个三态输出锁存器组成。多路开关可选通 8 个模拟通道, 允许 8 路模拟量分时输入, 共用 A/D 转换器进行转换。三态输出锁存器用于锁存 A/D 转换完的数字量, 当 OE 端为高电平时, 才可以从三态输出锁存器取走转换完的数据。

ADC0809 芯片性能特点: 是一个逐次逼近型的 A/D 转换器, 外部供给基准电压; 单通道转换时间 116 μ s; 分辨率为 8 位, 带有三态输出锁存器, 转换结束时, 可由 CPU 打开三态门, 读出 8 位的转换结果; 有 8 个模拟量的输入端, 可引入 8 路待转换的模拟量。

ADC0809 的数据输出结构是内部有可控的三态缓冲器, 所以它的数字量输出信号线可以与系统的数据总线直接相连。内部的三态缓冲器由 OE 控制, 当 OE 为高电平时, 三态缓冲器打开, 将转换结果送出; 当 OE 为低电平时, 三态缓冲器处于阻断状态, 内部数据对外部的数据总线没有影响。因此, 在实际应用中, 如果转换结束, 要

读取转换结果,则只要在 OE 引脚上加一个正脉冲,ADC0809 就会将转换结果送到数据总线上。

4.1.2 ADC0809 引脚功能

ADC0809 如图 4.2 所示:

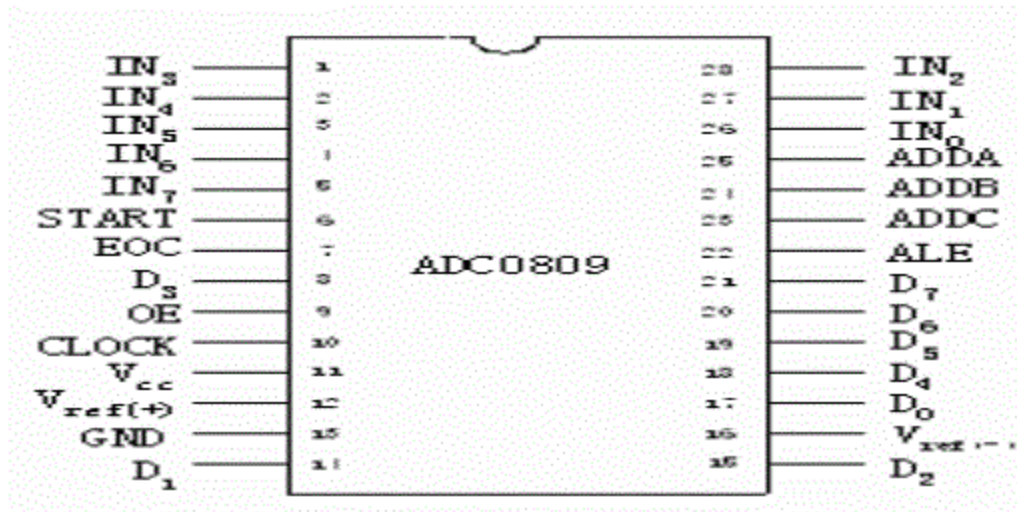


图 4.2 ADC0809 引脚结构

1、模拟信号输入 IN0~IN7: IN0-IN7 为八路模拟电压输入线,加在模拟开关上,工作时采用时分割的方式,轮流进行 A/D 转换。

2、地址输入和控制线:地址输入和控制线共 4 条,其中 ADDA、ADDB 和 ADDC 为地址输入线,用于选择 IN0-IN7 上哪一路模拟电压送给比较器进行 A/D 转换。ALE 为地址锁存允许输入线,高电平有效。当 ALE 线为高电平时,ADDA、ADDB 和 ADDC 三条地址线上地址信号得以锁存,经译码器控制八路模拟开关通路工作。

3、数字量输出及控制线(11 条):START 为“启动脉冲”输入线,上升沿清零,下降沿启动 ADC0809 工作。EOC 为转换结束输出线,该线高电平表示 AD 转换已结束,数字量已锁入“三态输出锁存器”。D0-D7 为数字量输出线,D7 为最高位。ENABLE 为“输出允许”线,高电平时能使 D0-D7 引脚上输出转换后的数字量。

4、电源线及其他(5 条):CLOCK 为时钟输入线,用于为 ADC0809 提供逐次比较所需,一般为 640kHz 时钟脉冲。Vcc 为+5V 电源输入线,GND 为地线。+VREF 和-VREF 为参考电压输入线,用于给电阻网络供给标准电压。+VREF 常和

VDD 相连, $-V_{REF}$ 常接地。

4.2 DAC0832 的应用

DAC0832 是采样频率为八位的 D/A 转换器件。

4.2.1 DAC0832 内部结构

DAC0832 内部结构资料: 芯片内有两级输入寄存器, 使 DAC0832 具备双缓冲、单缓冲和直通三种输入方式, 以便适于各种电路的需要(如要求多路 D/A 异步输入、同步转换等)。D/A 转换结果采用电流形式输出。要是需要相应的模拟信号, 可通过一个高输入阻抗的线性运算放大器实现这个功能。运放的反馈电阻可通过 RFB 端引用片内固有电阻, 海可以外接。

该片逻辑输入满足 TTL 电压电平范围, 可直接与 TTL 电路或微机电路相接, 芯片电路内部结构图如图 4.3 所示:

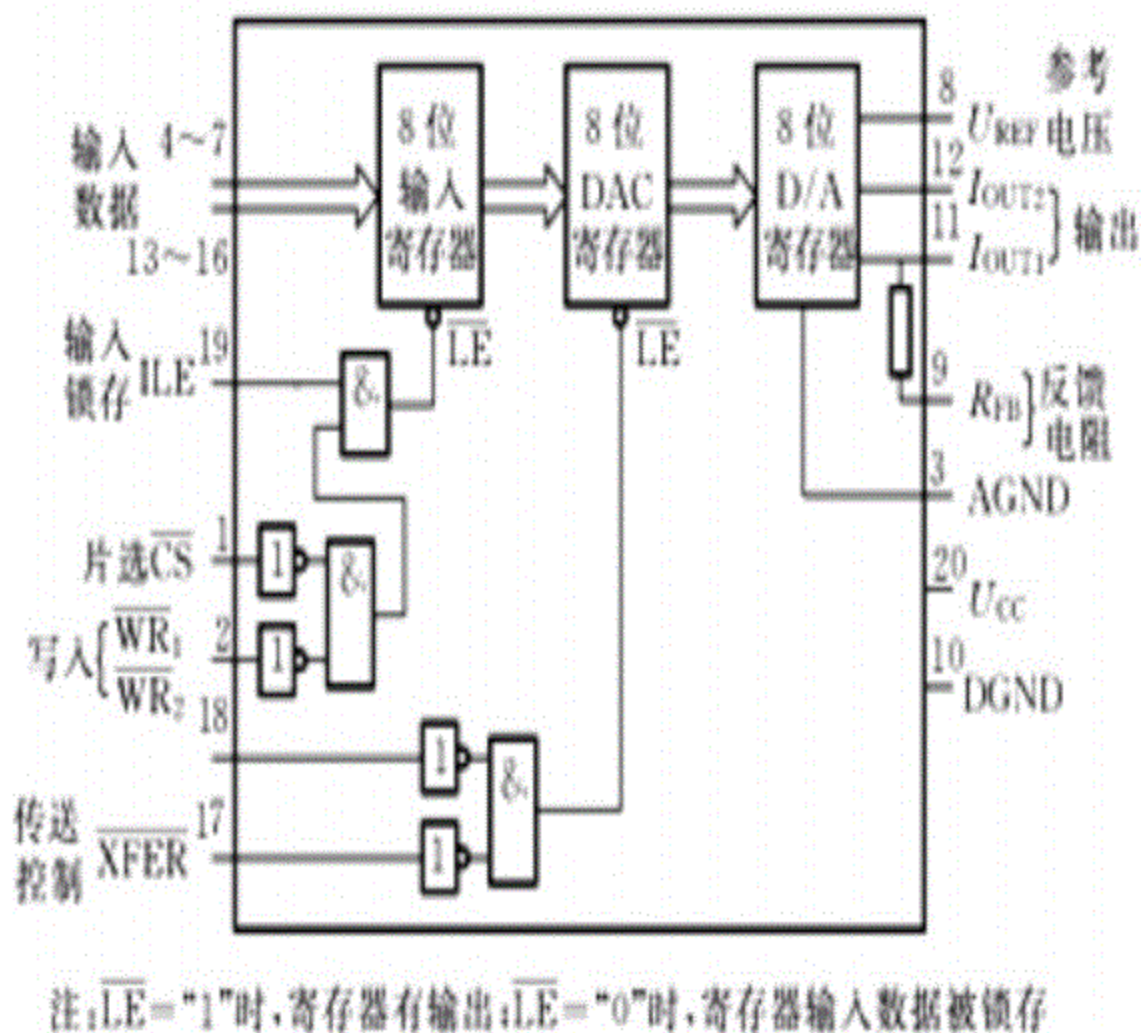


图 4.3 DAC0832 内部结构图

4.2.2 DAC0832 引脚功能

DAC0832 引脚图如图 4.4 所示:

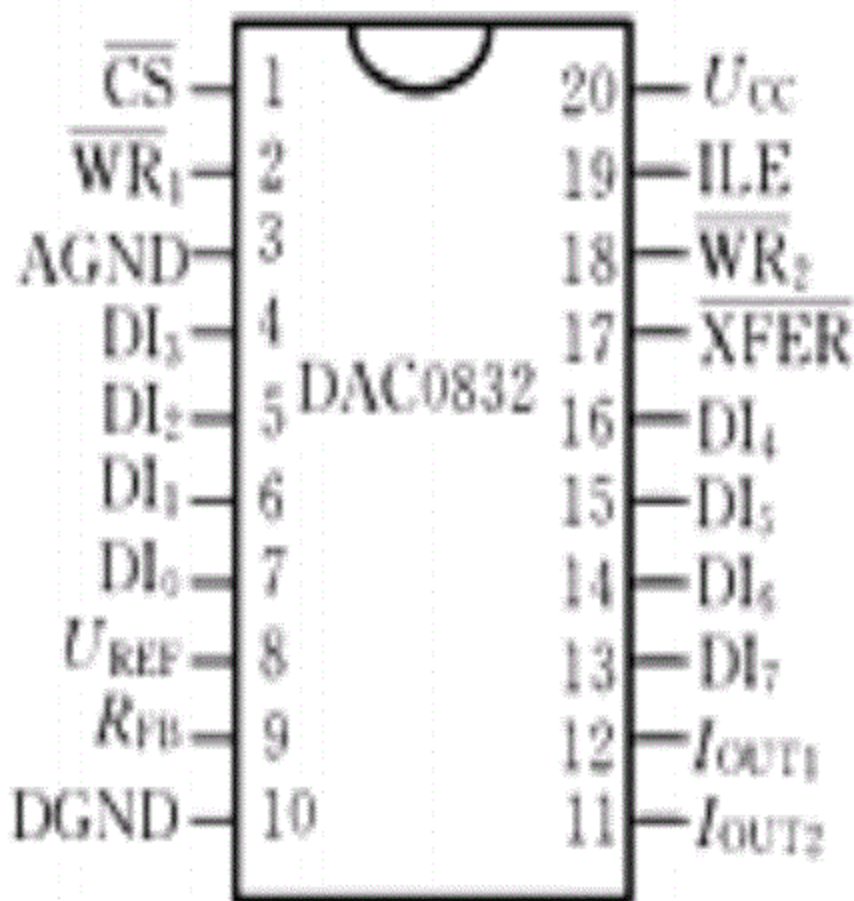


图 4.4 DAC0832 引脚结构图

D0~D7: 8 位数据输入线, TTL 电平, 有效时间应大于 90ns (否则锁存器的数据会出错);

ILE: 数据锁存允许控制信号输入线, 高电平有效;

CS: 片选信号输入线 (选通数据锁存器), 低电平有效;

WR1: 数据锁存器写选通输入线, 负脉冲 (脉宽应大于 500ns) 有效。由 ILE、CS、WR1 的逻辑组合产生 LE1, 当 LE1 为高电平时, 数据锁存器状态随输入数据线变换, LE1 的负跳变时将输入数据锁存;

XFER: 数据传输控制信号输入线, 低电平有效, 负脉冲 (脉宽应大于 500ns) 有效;

WR2: DAC 寄存器选通输入线, 负脉冲 (脉宽应大于 500ns) 有效。由 WR1、XFER 的逻辑组合产生 LE2, 当 LE2 为高电平时, DAC 寄存器的输出随寄存器的输入而变化, LE2 的负跳变时将数据锁存器的内容打入 DAC 寄存器并开始 D/A 转换。

IOUT1: 电流输出端 1, 其值随 DAC 寄存器的内容线性变化;

IOUT2: 电流输出端 2, 其值与 IOUT1 值之和为一常数;

Rfb: 反馈信号输入线, 改变 Rfb 端外接电阻值可调整转换满量程精度;

Vcc: 电源输入端, Vcc 的范围为+5V~+15V;

VREF: 基准电压输入线, VREF 的范围为-10V~+10V;

AGND: 模拟信号地;

DGND: 数字信号地;

4.3 实验仿真图形

实验仿真图形如图 4.5 所示:

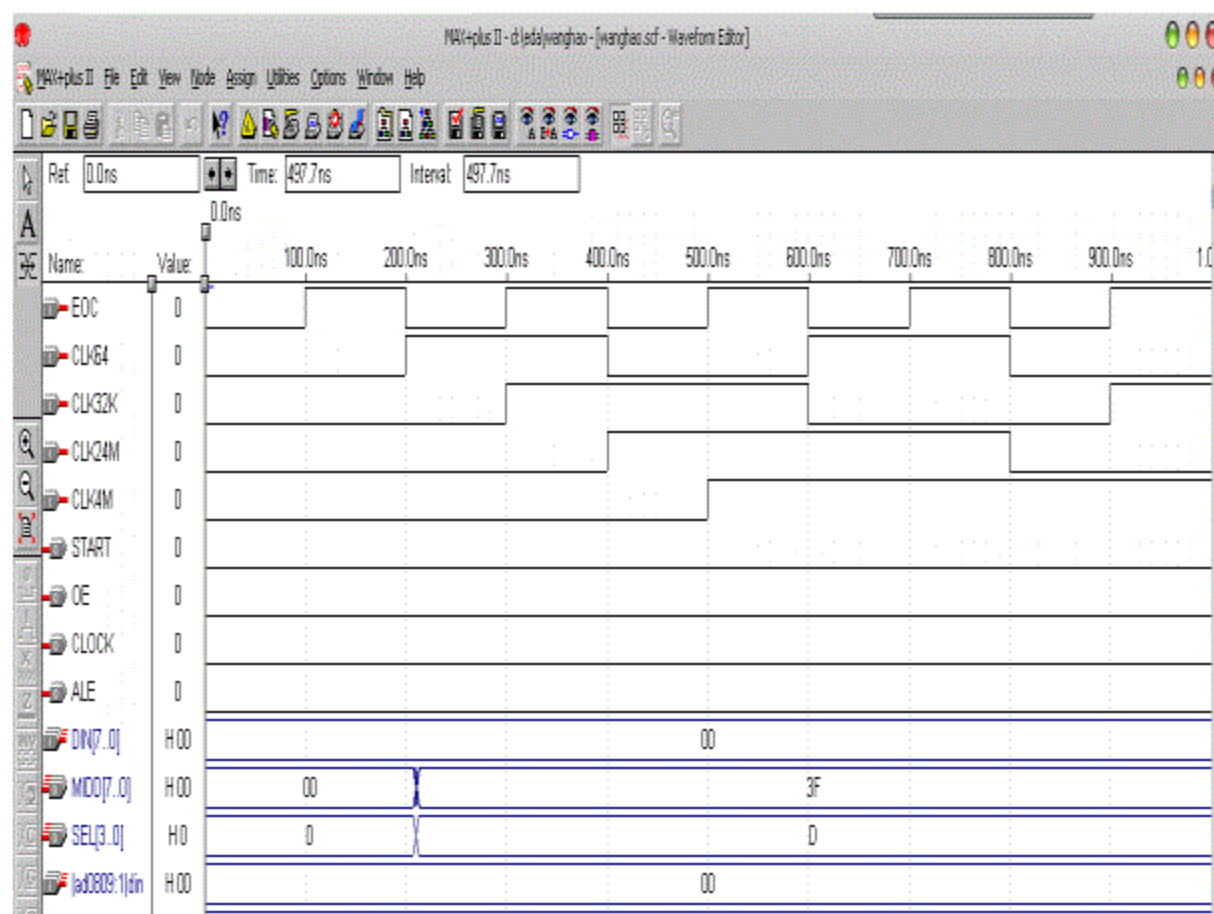


图 4.5 实验仿真图形

5 课程设计心得体会

这次 EDA 课程设计历时两个星期，前面几天就是在寝室学习和熟悉 MAX+plus II 软件，学到了很多很多的东西，同时不仅可以巩固以前所学过的知识，而且学到了很多在书本上所没有学到过的知识。过程中也遇到了很多问题，不过及时的问老师解决问题，使我更加了解了模拟量采集的原理。在自己有余的时间还帮助同学修改性能指标，通过这次设计，进一步加深了对 EDA 的了解，让我对它有了更加浓厚的兴趣。特别是当每一步成功时，心里特别的开心。

通过这次课程设计使我懂得了理论与实际相结合是很重要的，只有理论知识是远远不够的，只有把所学的理论知识与实践结合起来，从理论中得出结论，才能真正为社会服务，从而提高自己的实际动手能力和独立思考的能力。在设计的过程中遇到问题，可以说得是困难重重，这毕竟是第一次接触 EDA 设计，难免会遇到过各种各样的问题，同时在设计的过程中发现了自己不足之处，对以前所学过的知识理解得不够深刻，掌握得不够牢固。

总的来说，这次模拟量采集的设计还是比较成功的，在设计中遇到了很多问题，最后在老师的辛勤的指导下，终于游逆而解，有点小小的成就感，终于觉得平时所学的知识有了实用的价值，达到了理论与实际相结合的目的，不仅学到了不少知识，而且锻炼了自己的能力，使自己对以后的路有了更加清楚的认识，同时，对未来有了更多的信心。

参考文献

- [1] 阎石 主编,《数字电子技术基础》,高等教育出版社,1998
- [2] 谭会生等主编,《EDA 技术及应用》,西安电子科技大学出版社,2001
- [3] 廖裕评等主编,《CPLD 数字电路设计——使用 MAX+plus II 入门篇》,清华大学出版社,2001
- [4] 冯涛等主编,《可编程逻辑器件开发技术: MAX+plus II 入门与提高》,人民邮电出版社,2002
- [5] 杨崇志,《特殊新型电子元件手册》,辽宁科学技术出版社,1999
- [6] 彭介华,《电子技术课程设计指导》高等教育出版社.2000 年出版.
- [7] Mark Zwolinski, Digital System Design with VHDL, 电子工业出版社,2002
- [8] Alan B. Marcovitz Introduction to logic Design, 电子工业出版社,2002

致 谢

同过自己不断的努力以及老师的耐心指导和热情帮助，本设计已经基本完成。在此对李老师和帮助过我的同学表示深深的感谢。

由于自身能力有限，设计中一定存在很多不足之处，敬请各位老师批评指正，我一定虚心接受并进一步改进。

附录

ADC0809 控制模块源程序如下:

```
library ieee;

use ieee.std_logic_1164.all;

entity AD0809 is
port(clk,eoc:in std_logic;
din:in std_logic_vector(7 downto 0);
start,oe,ale:out std_logic;
dout:out std_logic_vector(7 downto 0));
end AD0809 ;

architecture AD0809_arc of AD0809 is
begin
process(clk)
variable x,y:std_logic;
variable state:integer range 0 to 2;
begin
if clk'event and clk='1' then
if x='0' then
case state is
when 0=>ale<='1';
state:=1;
when 1=>start<='1';
state:=2;
when 2=>ale<='0';
start<='0';
state:=0;
```

```

x:='1';
end case;
else
  if eoc='0' then
    y:='1';
    end if;
  if y='1' then
    if eoc='1' then
      case state is
        when 0=>oc<='1';
state:=1;
        when 1=>dout<=din;
state:=2;
        when 2=>oc<='0';
state:=0;
x:='0';
y:='0';
      end case;
    end if;
    end if;
    end if;
  end process;
end AD0809_arc;

```

DAC0832 控制模块源程序如下:

```

library IEEE;
use IEEE.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

```

```

use ieee.std_logic_arith.all;

entity bcd_to_8421 is
port(clk: in std_logic;
reset: in std_logic;
      bcd: in std_logic_vector(7 downto 0);
      b_out: out std_logic_vector(7 downto 0)
);
end bcd_to_8421;

architecture bcd_arch of bcd_to_8421 is
signal da,db: std_logic_vector(7 downto 0);
begin
process(reset,clk,bcd)
begin
if clk'event and clk='1' then
if reset='0' then
da<="00000000";
db<="00000000";
      b_out<="00000000";
else
da<=bcd(7 downto 4)*"1010"+bcd(3 downto 0);
end if;
      end if;
if da>="00110010" then
      db<="11111111";
else
db<=da*"101";
end if;
      b_out<=db;
end process;

```

```
end bcd_arch;
```

分频电路 FEN 模块源程序如下:

```
library ieee;

use ieee.std_logic_1164.all;

entity fen is
port(clk4m:in std_logic;
q:out std_logic );
end fen;

architecture fen_arc of fen is
begin
process(clk4m)
variable cnt:integer range 0 to 3;
variable x:std_logic;
begin
if clk4m'event and clk4m='1' then
if cnt<3 then
cnt:=cnt+1;
else
cnt:=0;
x:=not x;
end if;
end if;

q<=x;
end process;
end fen_arc;
```

用数码管显示模数 A/D 转换器的输入电压源程序如下:

```
library IEEE;

use IEEE.std_logic_1164.all;
```

```

use ieee.std_logic_unsigned.all;

use ieee.std_logic_arith.all;

entity dddf is

port(in_all: in STD_LOGIC_VECTOR(7 downto 0);

out_l,out_h,out_hh,out_hhh:out STD_LOGIC_VECTOR (3 downto 0);

clk24,clr:in std_logic);

end dddf;

architecture def_arch of dddf is

signal b1,b2,b3,b4:std_logic_vector(3 downto 0);

signal a:std_logic_vector(1 downto 0);

signal mid:std_logic_vector(11 downto 0);

signal e,f:std_logic_vector(11 downto 0);

signal big,bigg,biggg:integer range 0 to 10000000;

begin

process(clk24,in_all,clr)

begin

if clk24'event and clk24='1' then

if clr='0' then

a<="00";

b1<="0000";

b2<="0000";

b3<="0000";

b4<="0000";

else

case a is

when"00"=>

e<="0000"&in_all;

biggg<=conv_integer(e);

bigg<=biggg*5000;

```

```
big<=bigg/256;
f<=conv_std_logic_vector(big,12);
    mid<=f;
a<="01";
when"01"=>
if mid>="1010" then
    mid<=mid-"1010";
b2<=b2+1;
if b2="1001" then
b2<="0000";
b3<=b3+1;
if b3="1001" then
b3<="0000";
b4<=b4+1;
end if;
    end if;
else
b1<=mid(3 downto 0);
end if;
a<="01";
when others=>null;
end case;
end if;
end if;
end process;
out_l<=b1;
out_h<=b2;
out_hh<=b3;
out_hhh<=b4;
```

```
end def_arch;
```

4 位二-十进制数转换数码管字符源程序如下:

```
library IEEE;
use IEEE.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
use ieee.std_logic_arith.all;

entity disp is
port(clk64:in std_logic;
inl:in std_logic_vector(3 downto 0);
inm:in std_logic_vector(3 downto 0);
inh:in std_logic_vector(3 downto 0);
inhh:in std_logic_vector(3 downto 0);
midd:out std_logic_vector(7 downto 0);
disp1:out std_logic_vector(1 downto 0)
);
end disp;

architecture arch of disp is
signal mid: std_logic_vector(4 downto 0);
signal disp2: std_logic_vector(1 downto 0);

begin
process(clk64,inl,inm,inh,inhh)
begin
if clk64'event and clk64='1' then
if disp2="11" then
disp2<="00";
else
disp2<= disp2+1;
end if;
```

```

disp1<=disp2;

case disp2 is

when"10"=>mid<='0'&inl;

when"01"=>mid<='0'&inm;

when"00"=>mid<='0'&inh;

when"11"=>mid<='1'&inhh;

when others=>null;

end case;

case mid is

when"00000"=>midd<="00111111";

when"00001"=>midd<="00000110";

when"00010"=>midd<="01011011";

when"00011"=>midd<="01001111";

when"00100"=>midd<="01100110";

when"00101"=>midd<="01101101";

when"00110"=>midd<="01111101";

when"00111"=>midd<="00000111";

when"01000"=>midd<="01111111";

when"01001"=>midd<="01101111";

when"10000"=>midd<="10111111";

when"10001"=>midd<="10000110";

when"10010"=>midd<="11011011";

when"10011"=>midd<="11001111";

when"10100"=>midd<="11100110";

when"10101"=>midd<="11101101";

when"10110"=>midd<="11111101";

when"10111"=>midd<="10000111";

when"11000"=>midd<="11111111";

when"11001"=>midd<="11101111";

```

```

when others=>null;

end case;

end if;

end process;

end arch;

```

数码管阴极扫描二-四线译码源程序如下：

```

library ieee;

use ieee.std_logic_1164.all;

use ieee.std_logic_unsigned.all;

use ieee.std_logic_arith.all;

entity ym is

    port(clk64:in std_logic;

          disp:in std_logic_vector(1 downto 0);

          sel:out std_logic_vector(3 downto 0)

    );

end ym;

architecture arch of ym is

begin

    process(clk64)

    begin

        if clk64'event and clk64='1' then

            case disp is

                when "10"=>sel<="0111";

                when "01"=>sel<="1011";

                when "00"=>sel<="1101";

                when "11"=>sel<="1110";

                when others=>null;

```