

苏州大学

硕士学位论文

基于VHDL的出租车计价器设计

姓名：凌璟

申请学位级别：硕士

专业：集成电路工程

指导教师：李富华

20091001

基于 VHDL 的出租车计价器设计

中文摘要

出租汽车是城市公共交通的重要组成部分，出租汽车计价器是一种专用的计量仪器，它安装在出租汽车上，指示出载客里程数，以及乘客应付费用的总数。出租车计价器在最初使用时具备的基本功能是根据行驶里程计价，要求精度高，可靠性好。随着电子技术的发展以及对计价器的不断改进和完善，产生了诸多的附加功能。

VHDL 是一种应用广泛的硬件描述语言，设计者可以通过它编写代码，通过模拟器仿真验证其功能，完成逻辑综合与逻辑优化，最后通过下载到相应的可编程逻辑器件（如 FPGA）中来实现设计。

出租车计价系统在生活中应用广泛，较多的是利用单片机进行控制，但较易被私自改装，且故障率相对较高；而 FPGA 具有高密度、可编程及有强大的软件支持等特点，所以设计的产品具有功能强、可靠性高、灵活性好等特点。在本设计中针对目前常用的这两种设计方案（单片机设计方案和 FPGA 设计方案）进行了优劣比较，最终确定用 FPGA 来实现系统设计，根据预定的设计要求和设计思路进行了功能设计。

文中给出了计费系统的整体组成框图和各系统模块的详细设计过程。利用 VHDL 语言、PLD 设计出租车计费系统，采用 VHDL 编写租车计费器系统程序，采用 MAX+PLUS II 软件作为开发平台，进行了程序仿真，验证设计实现了出租车计价器的相关功能。

关键词：出租车计价器； VHDL 语言； MAX+PLUS II；

作 者：凌 璟

指导教师：李富华

The Design of Taximeter Based on VHDL

Abstract

The taxi is the most important part in a city public traffic. The taxi mileometer is a special computation instrument, which show the milemeter and the total expense for passengers through installing taxi. In the initial utilization, the basic function of taxi mileometer is calculating the expense by mileage, the requirements focus on the higher precision and better reliability. Now there are lots of additional function along with the electronic technology development and the mileometer improvement.

VHDL is hardware descriptive language in comprehensive application. The designer can write code through it, and through the simulator simulation functionality to complete the logic synthesis and logic optimization. Finally, completing the design through download to corresponding programmable logic instrument (e.g. FPGA)

The taxi pricing system has comprehensive application in our life. Most of them are controlled by SCM. It's easy to refit privately, and the malfunction rate is higher. FPGA is higher density, programmable and strong software support, so the characteristic of designing produce is strong function, higher reliability, good flexibility. In the design, I compare with the tow ordinary project (SCM project and FPGA project), and decide to use FPGA project to achieve the system design. According to the preconcerted design requirements and assumption, I complete the function design.

There are the whole structural chart and the detail design process of each systemic module in the article. Designing the taxi payroll system is by VHDL AND PLD, designing the taxi mileometer system is by VHDL, designing the development platform is by MAX+PLUS II software, meanwhile, testing and verifying the function achievement through the program emulation mode.

Key words: The rental car costs the system; the VHDL language; MAX+PLUS II;

Written by Ling jing

Supervised by Li fuhua

苏州大学学位论文独创性声明及使用授权的声明

学位论文独创性声明

本人郑重声明：所提交的学位论文是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含其他个人或集体已经发表或撰写过的研究成果，也不含为获得苏州大学或其它教育机构的学位证书而使用过的材料。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本人承担本声明的法律责任。

研究生签名： 凌景 日 期： 2009.10.29

学位论文使用授权声明

苏州大学、中国科学技术信息研究所、国家图书馆、清华大学论文合作部、中国社科院文献信息情报中心有权保留本人所送交学位论文的复印件和电子文档，可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外，允许论文被查阅和借阅，可以公布（包括刊登）论文的全部或部分内 容。论文的公布（包括刊登）授权苏州大学学位办办理。

研究生签名： 凌景 日 期： 2009.10.29

导师签名： 李富华 日 期： 2009.10.29

第一章 引言

1.1 设计背景

出租车行业在我国是八十年代初兴起的一项新兴行业，随着我国国民经济的高速发展，出租汽车已成为城市公共交通的重要组成部分。多年来国内普遍使用的计价器只具备单一的计量功能。目前全世界的计价器中有 90% 为台湾所生产。现今我国生产计价器的企业有上百家，主要是集中在北京，上海，沈阳和广州等地。

出租汽车计价器是一种专用的计量仪器，它安装在出租汽车上，能连续累加，并指示出行程中任一时刻乘客应付费用的总数，其金额值是计程和计时时间的函数。我国第一家生产计价器的是重庆市起重机厂，最早的计价器全部采用机械齿轮结构，只能完成简单的计程功能，可以说，早期的计价器就是个里程表。随着科学技术的发展，产生了第二代计价器。它采用了手摇计算机与机械结构相结合的方式，实现了半机械半电子化，在计程的同时还可完成计价的工作。大规模集成电路的发展产生了第三代计价器，也就是全电子化的计价器，其功能在不断完善中。

出租车计价器在最初使用时具备的主要功能是根据行驶里程计价，要求精度高，可靠性好。随着电子技术的发展以及对计价器的不断改进和完善，便产生了诸多的附加功能。例如：(1)LED 显示功能，数码管的使用让计价器实现多屏显示的功能，可同时显示各项营运数据，使乘客一目了然；(2)永久时钟功能，在非营运状态下，日历时钟芯片的使用使计价器可以显示永久时钟；(3)存储功能，可存储多项营运数据，便于查询。新型数据存储器的应用使得计价器的营运数据在掉电情况下还可以保存 10 年。

二十世纪后半期，随着集成电路和计算机技术的飞速发展，数字系统也得到了飞速发展，其实现方法经历了由分立元件、小规模集成电路 SSI (Small-scale Integration)、中规模集成电路 MSI (Medium-scale Integration) 到大规模集成电路 LSI (Large-scale Integrate)、超大规模集成电路 VLSI (Very Large-scale Integrate) 以及特大规模集成电路 ULSI (Ultra Large-scale Integrate) 的过程。同时为了提高系统

的可靠性与通用性，微处理器和专业集成电路 ASIC(Application Specific Integrated Circuit) 逐渐取代了通用全硬件 LSI 电路，而 ASIC 以其体积小、重量轻、功耗低、速度快、成本低、保密性好而脱颖而出。目前，业界大量可编程逻辑器件 PLD (programmable logic device)，尤其是现场可编程逻辑器件 (FPLD) 被大量地应用在 ASIC 的制作当中。在可编程集成电路的开发过程中，以计算机为工作平台，融合了应用电子技术、计算机技术、智能化技术最新成果的电子设计自动化 EDA(Electronic Design Automation) 技术主要能辅助进行三方面的设计工作：IC(Integrated Circuit)设计,电子电路设计以及 PCB(Printed Circuit Board)设计

理想的可编程逻辑开发系统能符合大量的设计要求：它能够支持不同结构的器件，在多种平台运行，提供易于使用的界面，并且有广泛的特征。此外，一个设计系统应该能给设计师提供充分自由的设计输入方法和设计工具选择。Altered 公司开发的 MAX+PLUS II 开发系统能充分满足可编程逻辑设计所有要求。

MAX+PLUS II 设计环境所提供的灵活性和高效性是无可比拟的。其丰富的图形界面，辅之以完整的、可及时访问的在线文档，使设计人员能够轻松、愉快地掌握和使用 MAX+PLUS II 软件。

编程器是一种专门用于对可编程器（如 EPROM(Erasable Programmable ROM),EEPROM(Electrically Erasable Programmable ROM),GAL(Generic Array Logic),CPLD(Electrically Erasable Programmable ROM),PAL 等）进行编程的专业设备

PLD 器件的逻辑功能描述一般分为原理图描述和硬件描述语言描述，原理图描述是一种直观简便的方法，它可以将现有的小规模集成电路实现的功能直接用 PLD 器件来实现，而不必去将现有的电路用语言来描述，但电路图描述方法无法做到简练；硬件描述语言描述是可编程器件设计的另一种描述方法，语言描述可能精确和简练地表示电路的逻辑功能，现在 PLD 的设计过程中广泛使用。常用的硬件描述语言有 ABEL,VHDL 语言等，其中 ABEL 是一种简单的硬件描述语言，其支持布尔方程、真值表、状态机等逻辑描述，适用于计数器、译码器、运算电路、比较器等逻辑功能的描述；VHDL 语言是一种行为描述语言，其编程结构类似于计算机中的 C 语言，在描述复杂逻辑设计时，非常简洁，具有很强的逻辑描述和仿真能力，是未来硬件设计语言的主流。

VHDL(VHSIC(Very High Speed IC) Hardware Description Language)就是超高速集成电路硬件描述语言。覆盖面广,描述能力强,是一个多层次的硬件描述语言。在 VHDL 语言中,设计的原始描述可以非常简练,经过层层加强后,最终可成为直接付诸生产的电路或版图参数描述。具有良好的可读性,即容易被计算机接受,也容易被读者理解。使用期长,不会因工艺变化而使描述过时。因为 VHDL 的硬件描述与工艺无关,当工艺改变时,只需修改相应程序中的属性参数即可。支持大规模设计的分解和已有设计的再利用。一个大规模的设计不可能由一个人独立完成,必须由多人共同承担,VHDL 为设计的分解和设计的再利用提供了有力的支持。

随着出租车行业的发展,对出租车计费器的要求也越来越高。本课题通过软件编程来完成计价器更多的附加功能,具有一定的应用价值。由于科技的发展,芯片中的数据可保持十年不变,且芯片体积小,容量大,因此这种方式具有十分重要的现实意义和广阔的市场前景。

1.2 目前出租车计价器存在的问题和改进方案

1.2.1 出租车计价器存在的问题

随着我国国民经济的高速发展,出租汽车已成为城市公共交通的重要组成部分。近几年来,出租汽车行业在各地蓬勃发展,出租车经营也从无序状态逐渐走入正轨。出租车计价器成为出租车运营中必配的、可靠的计量器具,其使用准确性直接关系到经营者与乘客的经济利益。但在实际运营中,总有不少经营者或乘客反映相同路径,不同的出租车的收费有较大的出入。目前出租车计价器存在的问题主要有:

1 计价器作弊的问题,有些出租汽车司机在计价器上做手脚,多收乘客租金。常见的手段是:输入非法脉冲,计价器主要靠传感器发出的脉冲来计数,脉冲数的多少即汽车行驶里程的多少。作弊者利用这一原理,额外向计价器输入脉冲,其方法多种多样如利用车内收音机、电动剃须刀。

2 毛刺(竞争—冒险)现象。主要影响数字系统设计有效性和可靠性的主要因素,由于毛刺的存在,使得系统存在诸多潜在的不稳定因素,尤其是对尖峰脉冲或脉冲边沿敏感的电路就更是如此。毛刺通常主要对电路的触发清零端 CLR、触发复位端

RESET、CP 端、锁存器的门控端和专用芯片的控制端等产生严重的影响，会使电路发生误动作，从而造成数字系统的逻辑混乱。

1.2.2 改进方法

1、相对于计价器作弊的问题，改进的方法：

(1) 传感器输出编码脉冲。传感器输出的脉冲其脉冲宽度、占空比等参数都是可变的，使计价器与传感器对号入座。

(2) 传感器输出加密脉冲，近年来由于电子技术的发展，已有商品化的加、解密器件供应，只要在传感器与计价器之间加入这种器件，就可以使计价器发出的加密脉冲，而其他传感器或脉冲一概不认。

(3) 加入启动速度的判别电路：利用汽车启动时速度总是由零逐步增高的道理，而非法的外加脉冲总是以固定速度加入的，当加入了判别电路后，如果速度不是由低到高则计价器拒绝接受。

2、在竞争冒险消除毛刺的主要的方法：

(1) 调整路径延时，毛刺归根到底是由于延时不同而引起的，因此只要让门电路的所有输入信号具有相同的延时，毛刺就不会产生。所以消除毛刺最原始、最直接的方法就是调整延时；

(2) 引入选通信号，由于竞争仅仅发生在输入信号变化转换的瞬间，在稳定状态是没有竞争的，所以在输入信号稳定后进行选通就可以彻底消除尖峰脉冲。

3、本设计课题拟解决的问题：

本课题采用 CPLD 芯片为核心，用较少的硬件和适当的软件相互配合主要解决以下问题：

(1) 不同地区的计费方式存在差异；即使同一地区，不同车型的出租车，其计费方式也有差别；另一方面，出租车还面临几年一次的调价或调整计费方式等问题。因此，所设计的计费器不仅要能满足不同地区的要求，而且计费方式的调整也应当很方便。

(2) 由于个别地区对计费器有特殊要求，有时必须修改软件；另外，计费器还面临软件的升级；因而，所设计的计费器应能很方便地重新编程。

(3) 计费器必须要有防作弊功能，能有效防止司机作弊；同时要防止计费器在

营运过程中死机。

1.3 VHDL 简介

在传统的硬件电路设计中，主要的设计文件是电路原理图，而采用 HDL 设计系统硬件电路时主要使用 HDL 编写源程序。所谓硬件描述语言 HDL (Hardware Description Language)，就是该语言可以描述硬件电路的功能，信号连接关系及定时关系。许多公司开发了自己专有的 HDL，包括 Zycad 公司的 ISP，Gateway Design Automation 公司的 Verilog 以及 Mentor Graphics 公司的 BLM。其中，Silicon Compiler 公司的 M 及 Gateway 公司的 Verilog 以 C 语言为基础。UDL/I 在日本以标准 HDL 的形式出现。多年来设计者一直使用这些专用的 HDL。1982 年，各 ASIC 芯片厂商相继开发了用于各自目的的 HDL。1987 年底，IEEE 确认美国国防部开发的 VHDL 为标准硬件描述语言 (IEEE-1076)。之后，各 EDA 公司研制的硬件电路设计工具逐渐向 VHDL 靠拢，VHDL 在电子设计领域得到广泛的接受，1993 年，IEEE 对 VHDL 进行了修订，公布了新版本的 VHDL (即 IEEE-1076-1993)。现在，VHDL 和 Verilog 作为 IEEE 的工业标准硬件描述语言，在电子工程领域，从各公司的设计人员到各大学的教授、学生，都极其重视对其的学习研究，VHDL 已成为事实上的通用硬件描述语言。有专家认为，在 21 世纪中，几乎全部的数字系统设计任务将由 VHDL 与 Verilog 语言承担，VHDL 将是电子工程设计人员的必备知识。VHDL 和其他语言相比，最大的区别在于设计方法上的差别。VHDL 的主要优点有：

(1) VHDL 支持自顶至下的和基于库的设计方法，而且支持同步电路、异步电路、现场可编程门阵列器件 FPGA (field programmable gate array) 以及其他随机电路的设计。VHDL 具有比其他硬件描述语言更强的行为描述能力，基于抽象的行为描述风格避开了具体的器件结构，使设计人员能从逻辑行为上描述和设计大规模电子系统。目前流行的 EDA 工具和 VHDL 综合器大都能实现行为描述到 RTL (Register Transfer Level) 描述的转换。

(2) VHDL 语句的行为描述能力和程序结构决定了它具有支持大规模设计的分解和已有设计再利用的功能，它支持系统的数学模型直到门级电路的描述，并且高层次的行为描述与低层次的门级电路描述、结构描述可以混合使用。这些特点符合 IC 设计的市场要求。VHDL 支持系统级描述，这是它优于其他 VHDL 的最重要的

特点。例如, Verilog 语言是一种门级电路描述语言, 其风格接近于电路原理图, 设计者需要搞清楚具体的电路结构的细节, 因此工作量通常较大。VHDL 语言却最适合于描述电路的行为, 即描述电路的功能, 然后由综合器来生成符合要求的电路网络。设计者在熟悉基本单元电路的描述风格, 积累一定的设计经验后, 就会为用 VHDL 设计同等性能电路的高效率所鼓舞。

(3) VHDL 的硬件描述与具体的工艺技术和硬件结构无关, 当门级或门级以上的描述通过仿真检验后, 再利用相应的工具将设计映射成不同的工艺, 因此电路的设计与工艺的改变是相互独立的。彼此的改变不会产生不良影响, 并且 VHDL 硬件描述语言的实现目标器件的选择范围广泛, 可使用各系列的 CPLD、FPGA 及各种门阵列器件。

(4) VHDL 具有类属描述语句和子程序调用等功能, 对于已完成的设计源程序, 可以通过修改类属参数表和函数的办法来改变设计的规模和结构。VHDL 具有丰富的仿真语句和库函数, 使得门电路级的功能仿真、检查成为可能, 使设计者对整个工程设计的结构和功能的可行性做出决策。

(5) VHDL 作为一种 IEEE 的工业标准, 使 VHDL 的设计成果便于重复利用和交流。这就更进一步推动了 VHDL 语言的推广及完善。另外, 由于其语法严格, 给阅读和使用带来极大的便利。VHDL 的设计流程如图 1.1 所示。

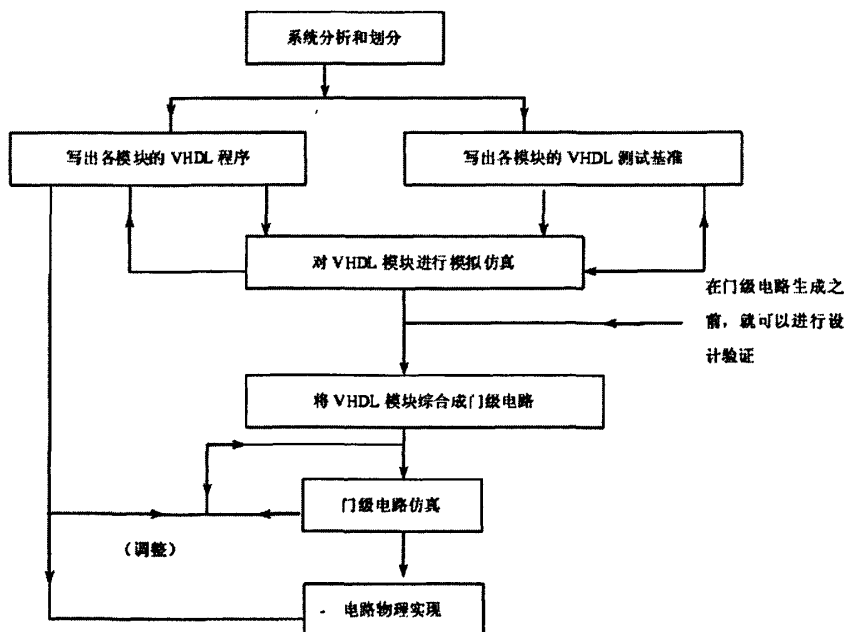


图 1.1VHDL 的设计流程

VHDL 的系统级描述缺乏设计概念上的抽象性,面向对象的 VHDL 可提高设计者在较高的设计层次上描述模型的能力,帮助设计者实现更复杂设计、更大规模的元件的重用。VHDL 面向对象的发展是语言本身进步的方向之一。面向对象的方法在软件开发中已被广泛地接受,它不仅仅是一种新的程序设计技术,而且是一种全新设计和构造软件的思维方法,它是计算机解决问题的方式更加类似于人类的思维方式和更强的管理能力。面向对象的语言必须包含抽象性、可封装性、模块化、层次化及信息机制。抽象性意味着一个对象的特性可以在类描述中文档化。可封装性是指代码和数据必须保存在同一单元中,封装性可有选择性的隐藏信息,使得某些信息对外界不可取。模块化定义了单元的重用。层次化使得对象的行为精炼,不必重复设立前驱中已有的内容。

一般的, VHDL 的设计均自 RTL 开始,模拟验证正确后综合到门级。最后移植到非 VHDL 模拟器上进行处理。这样做是因为 ASIC 开发缺少 VHDL 库。而且存在的 VHDL 库的速度太慢,实用性差。另一方面,在用 VHDL 设计电路时,缺乏统一的、有效的时序处理的描述方法。以前的唯一方法是通过组装实现,但此方法编译太慢,而且 VHDL 库过分依赖模拟器环境。VITAL(VHDL Initiative Towards ASIC Libraries)的应运而生为面向 ASIC 设计的 VHDL 模型的标准化研究开辟了新思路。1995 年 9 月 VITAL 正式通过成为 IEEE std 1076.4-1995 标准。VITAL 标准包括 4 个部分:时序程序包,基本元件包,延时机制,模型建立的规范文档。时序程序包和基本元件包与 std_logic_1164 一起放在 IEEE 库中。VITAL 具有以下特点: (1) 灵活的功能定义: VITAL 以过程和函数的形式提供元件功能的描述。函数计算元件的行为,过程实现电路的结构描述。(2) 延时定义精确: 延时可以以端到端的方式定义,可以依赖信号状态定义条件。(3) 具有精确的时序检查功能: 可进行建立时间和保持时间的检查,提供最小脉冲宽度、周期检查及事件冲突检查功能。(4) 两级模型描述规范: VITAL Level 0 描述外部接口, VITAL Level 1 描述内部实现,定义了统一的、形式化的建模风格,利于在工具内部实现和优化,以提高模拟速度。(5) 以工业标准为基础。

1.4 论文研究内容

本课题实现了出租车计价器的设计要求，且灵活性强，可以通过软件编程来完成更多的附加功能，具有一定的应用价值。本论文从介绍 VHDL 语言开始，设计了出租车计价器的方案，并运用 MAX plus II 对设计的电路进行模拟仿真，最终实现了设计要求。

全文结构如下：

第一章 介绍课题的背景、意义，VHDL 语言的特点和本文研究内容。

第二章 介绍了可实现设计的几种课题方案，并做出了优劣比较，最终决定采用 VHDL、PLD 来实现设计。

第三章 介绍了设计要求和设计思路。

第四章 以第三章的设计思路为基础，运用 VHDL 编写了各功能模块的程序。

第五章 对第四章中各个功能模块进行仿真，以确定其可行性。

第六章 对全文的工作进行总结，并对今后的研究工作进行了展望。

第二章 设计方案的选择

2.1 单片机与 FPGA 的设计比较

信息技术正在快速发展,其应用已经深入到各个领域各个方面。如今越来越多的电子产品向着智能化、微型化、低功耗方向发展,其中有的产品还需要实时控制和信号处理。电子系统的复杂性在不断增加,它迫切要求电子设计技术也有相应的变革和飞跃。使用纯 SSI 数字电路设计系统工作量大,灵活性低,而且系统可靠性差。广泛使用单片机(MCU)设计系统克服了纯 SSI 数字电路系统许多不可逾越的困难,是一个具有里程碑意义的飞跃。

近年来,PLD 器件迅速发展,尤其是 CPLD/ FPGA 向深亚微米领域进军,PLD 器件得到了广泛应用,以 CPLD/ FPGA 为物质基础的 EDA 技术诞生了。它具有电子技术高度智能化、自动化的特点,打破了软硬件最后的屏障,使得硬件设计如同软件设计一样简单。它作为一种创新技术正在改变着数字系统的设计方法、设计过程和设计观念。

单片机,PLD/ EDA 以其各自的特点满足了各种需要,正从各个领域各个层面改变着世界,它们已经成为数字时代的核心动力,推动着信息技术的快速发展。

关于本设计,有着大量的采用单片机设计的方案。单片机是集成了 CPU,ROM, RAM 和 I/O 口的微型计算机。它有很强的接口性能,非常适合于工业控制,因此又叫微控制器(MCU)。它与通用处理器不同,它是以工业测控对象、环境、接口等特点出发,向着增强控制功能,提高工业环境下的可靠性、灵活方便地构成应用计算机系统的界面接口的方向发展。所以,单片机有着自己的特点。

1、品种齐全,型号多样

自从 INTEL 推出 51 系列单片机,许多公司对它做出改进,发展成为增强型 51 系列,而且新的单片机类型也不断涌现。如 MOTOROLA 和 PHILIPS 均有几十个系列,几百种产品。CPU 从 8,16,32 到 64 位,多采用 RISC 技术,片上 I/O 非常丰富,有的单片机集成有 A/D;“看门狗”,PWM,显示驱动,函数发生器,键盘控制等,它们的价格也

高低不等,这样极大地满足了开发者的选择自由。

2、低电压和低功耗

随着超大规模集成电路的发展,NMOS 工艺单片机被 CMOS 代替,并开始向 HMOS 过渡。供电电压由 5V 降到 3V,2V 甚至到 1V,工作电流由 mA 降至 μA ,这在便携式产品中大有用武之地。

采用单片机设计具有成本低、易于控制的优点。设计大多是:通过软件编程,仿真,调试,符合要求后将程序用编程器写入到单片机芯片上。利用单片机的中断响应乘客开车或者等待的请求,进行相应的处理。利用寄存器存放计价器系统运行状态,从而可以根据整个行程的状态进行计价。但这种设计外围电路较多而时序实现能力较弱,调试复杂,抗干扰能力差,特别对这种计程车的计价器需要长时间不间断运作的系统,由于主要是软件运作,容易出错,造成系统不稳定。

EDA(Electronic Design Automation)即电子设计自动化,它是以计算机为工具,在 EDA 软件平台上,对用硬件描述语言 HDL 完成的设计文件自动地逻辑编译、逻辑化简、逻辑分割、逻辑综合及优化、逻辑布局布线、逻辑仿真,直至对于特定目标芯片进行适配编译、逻辑映射和编程下载等。设计者只需用 HDL 语言完成系统功能的描述,借助 EDA 工具就可得到设计结果,将编译后的代码下载到目标芯片就可在硬件上实现。这里的目标芯片就是 PLD 器件(FPGA/CPLD)。FPGA/CPLD 是 EDA 技术的物质基础,这两者是分不开的。可以说没有 PLD 器件,EDA 技术就成为无源之水。EDA 技术作为一种现代电子系统开发方式,具有两方面特点。

1、修改软件程序即可改变硬件

由于 FPGA/CPLD 可以通过软件编程对该硬件的结构和工作方式进行重构,修改软件程序就相当于改变了硬件,这使设计者把思路概念变为实际芯片和电路的周期越缩越短。软件可以使用自顶向下的设计方案,而且可以多个人分工并行工作,这些年来 IP 核产业的崛起,将若干软核结合起来就可以构成一个完整的系统,这一切极大地缩短了开发周期和上市时间,有利于在激烈的市场竞争中抢占先机。

2、速度快,可靠性高

MCU(Micro Controller Unit)和 DSP(Digital Signal Processor)都是通过串行执行指令来实现特定功能,不可避免低速,而 FPGA/CPLD 则可实现硬件上的并行工作,在实时测控和高速应用领域前景广阔;另一方面,FPGA/CPLD 器件在功

能开发上是软件实现的,但物理机制却和纯硬件电路一样,十分可靠;而且与其他的设计方法相比,FPGA 芯片具有比较丰富的存储单元,可以将原本需要外加的存储设备转移到 FPGA 芯片内部实现,使整体设计需要的外围元器件更少。而 MCU 和 DSP 芯片在强干扰条件下,尤其是强电磁干扰下,很可能越出正常的工作流程,出现 PC 跑飞现象。EDA 高可靠性正好克服了它们这一先天不足。

采用 PLD 设计,将所有器件集成在一块芯片上,体积大大减小的同时还提高了稳定性,并且可应用 EDA 软件仿真,调试,易于进行功能扩展,外围电路较少,采用硬件逻辑电路实现,其最大的优点是稳定性好,抗干扰能力强,非常适合作为计程车的计价器系统的控制核心。基于对以上各个因素的考虑,决定采用以 FPGA 为核心,用 VHDL 编程来实现计价器的设计方法,实现出租车计价器的设计,提升设计的可行性。

2.2 VHDL 的设计方法

集成电路从 20 世纪 60 年代开始经历了小规模、中规模、大规模、到目前的超大规模和特大规模阶段。1958 年设计出来的第一块集成电路只有 4 个晶体管,而目前已能制作规模达亿个晶体管以上的单个芯片,并构成一个完整的数字系统或数模混合的电子系统。几乎在集成电路的每个设计环节和整个设计过程都普遍使用 EDA(电子设计自动化)技术。要在几十平方毫米的硅片上完成线宽只有零点几个微米的上百万个器件的整个电子系统设计,只靠手工设计是完全不可能的,必须借用设计自动化(DA)或者 CAD 技术。

2.2.1 数字系统设计策略

任何复杂的数字系统都可以最终分解成基本门和存储器元件,这种分解最好由计算机自动进行。VLSI 设计过程就是把高级的系统描述最终转化成如何生产芯片的描述过程,为了完成这样的转换,人们研究出了描述集成电子系统的抽象方法,这就是层次化、结构化的方法。层次化的设计方法能使复杂的电子系统简化,并能在不同的设计层次及时发现错误并加以纠正;结构化的设计方法能把复杂抽象的系统划分成一些可操作的模块,允许多个设计者同时设计,而且某些子模块的资源可

以共用。

复杂的数字系统，不论是用 VLSI 或现场可编程门阵列实现，还是用印刷电路板实现，较好的策略都是用层次设计与自动设计相结合的方法，基本的设计过程是采用自顶向下的设计，也就是说，从一个行为概念开始，建立越来越详细的层次结构，直至得到一个充分低的级，它能直接变换成物理实现，最后，物理实现完成整个数字系统的功能

1、数字系统自动设计的要求

设计方法选取的主要依据是设计周期、设计成本、芯片成本、芯片尺寸、设计灵活性、保密性和可靠性等。

(1) 设计正确性

设计的正确性是数字系统设计中最基本的要求。设计一旦完成并送交制造厂生产后，再发现有错误，就需要重新制版，重新流片。一个复杂的数字系统设计，电路、版图数据量大，要作一次修改，代价是非常昂贵的。目前专用集成电路一般要求一次投片成功。中小规模集成电路可以人工验证，而对于 VLSI 来说，花费大量人工也无法保证设计的正确性。这就要求：在一个完整设计自动化系统的支持下，在各设计层次上都要进行反复验证和检查，各层次的设计数据都能自动转换和统一处理。由于数字系统设计的限制，需要有功能更强，性能更好的 EDA 设计工具将整个集成电路设计过程统一考虑，前后呼应，使系统设计达到最优。

目前，计算机辅助设计软件及工具几乎渗透到 VLSI 设计的各个步骤中，比如工艺模拟、器件模拟、电路分析、逻辑验证、版图验证及参数提取、布图工具、综合工具、计算机辅助设计、封装工具……

在复杂数字系统设计中，测试是一个十分重要的课题。测试的意义在于检查电路是否能按设计要求正常工作。随着芯片功能的日趋复杂，测试费用所占的比例明显增大，虽然芯片测试是在芯片生产过程当中进行的，但是为了减小测试所需要的资源，往往在电路设计阶段就要充分考虑其可测试性的问题。具体做法是在已有的逻辑设计基础上添加一些专门用于测试的辅助电路。

(2) 设计周期

由于市场竞争的需要，IC 产品要求几周甚至几天就要设计出来。在以往的芯片设计中，版图设计花费的时间最多。1978 年出品的微处理器 Z8000 的设计就是

一例，它含有 17500 个晶体管，版图设计花费了约 6600 个人时，占整个设计时间的一半以上。但今天，随着设计水平的提高，目前一个上百万门的常规数字系统的芯片设计，一个设计团队半个月就可完成设计、验证工作。

（3）设计成本

对于小批量的产品，应着重减小开发费用，对大批量的产品，应增加成品率和增加每一硅片上的芯片数。提高每一硅片上的芯片数主要靠提高工艺水平，减小芯片尺寸，增大硅晶片面积（目前主流为 12 英寸硅片）来实现。通常，小批量 ASIC 采用半定制电路或可编程器件技术，大批量 ASIC 则采用全定制电路技术。另外，要增加实际成品率，又必须减小每个芯片的尺寸，这就要求在高层次设计中优化电路结构，在版图设计中减少布局和布线中的所谓“死区”，提高芯片利用率。

（4）产品性能

IC 的性能主要取决于所选择的电路系统的体系结构、器件工艺结构和版图设计的质量。为了提高 IC 的速度，需要采用流水线体系结构，这样自动化便成了设计的主流。

综上所述，一个复杂数字系统的设计就是在保证产品质量的前提下，正确地选择 IC 体系结构、器件形式和工艺方案，同时要尽可能地减小芯片尺寸，降低设计成本和缩短设计周期。

2.2.2 VLSI 设计的描述域和层次

电路的层次设计过程，可以用 D'D'Gajski 于 1983 年提出的 Y 图描述。层次化、结构化描述方法如图 2.1 所示。在图 2.1 中三个互不相同的设计域由三条射线轴表示。这三个设计域是行为域、结构域和物理域。每个域中有多个抽象的级，而且离中心越远则抽象程度愈高。

行为域从概念上描述一个特定的系统做些什么，要完成什么功能，通常它只表示系统的输入输出间的函数关系；行为域的设计着眼于严密地规定逻辑部件。它根据逻辑部件的规格目标，以考虑给出什么样的输入信号序列、形成什么样的内部状态、发生什么样的输出信号序列等信息为中心进行设计。它对于用什么样的逻辑电路来实现其功能并不特别在意，只是去正确地定义逻辑部件所应完成的功能。行为域是复杂数字系统设计的出发点。

表 2.1 VLSI 设计的层次描述

设计层次	行为域描述	结构域描述	设计考虑
系统级	自然语言描述的性能指标	方框图	系统功能
算法级（芯片）	行为有限状态机、数据流图、控制流图	微处理器、存储器、串（并）行口、、中断控制器	时序、同步、测试
寄存器级（宏单元）	数据流表、真值表、有限状态机、状态表、状态图	寄存器、ALU、计数器、MUX、ROM 等	时序、同步、测试
逻辑（门）级	布尔方程、卡诺图	逻辑门、器件（晶体管）	选用适当的 基本门实现 硬件
电路级	电压、电流的微分方程	晶体管、R、L、C 等	电路性能、 延时、噪声
版图级	几何图形与工艺规则		

采用有效的设计方法是电路与系统设计成功的关键。设计过程的层次化、结构化把完整的硬件设计划分为一些可操作的模块，并允许多个设计者同时设计一个硬件系统中的不同模块，其中每个设计者负责自己所承担的部分。对于复杂数字系统设计，由于集成度太高，不能细分到电路级。因为 VLSI 芯片的结果非常强地依赖局部信息，而且整体性能涉及到局部设计，所以 VLSI 芯片设计是迭代的，要求频繁地调整，为此人们正在开发面向对象的 VHDL 来解决这种设计上的复杂性。

对照 Y 图，一个数字系统的设计可简单描述如下：首先从用户用自然语言的描述开始构建要设计的数字系统的性能、结构规范等。用 VHDL 语言写行为级算法代码，通过综合程序将算法代码自动生成寄存器级 VHDL 代码、寄存器级逻辑电路图，再通过综合程序自动生成门级的网表文件，至此，前端设计完成。当然，在前端设计中，工艺厂家的设计规则的导入及相应的仿真、建模和验证等步骤也是不可缺少的。其次，交由第三方 IP 整合服务公司处理，生成合乎某一流片工艺线需要的版图文件，经过版图生成、布局布线、版图后仿真等步骤，流片生成合格半定制芯片或全定制芯片。

2.2.3 设计自动化

复杂数字系统设计的复杂性,除了使设计周期延长外,还造成了设计人员的缺乏。从 Y 图可以看出,复杂数字系统的设计要求设计者不仅是一位电路设计者,而且是逻辑设计、计算机体系结构与应用软件的专家。为了解决芯片设计的危机,需要有新的电子设计自动化方法,目前有 3 种方法在发展之中。

(1) 计算机辅助的方法

计算机辅助的方法是最早的较为成熟的一种方法,也就是通常说的 CAD 技术。这种方法的基本思想是:所有的设计决定由设计者做出,设计危机可以通过提高设计者的工作效率来解决;给设计者提供完整的 EDA 工具软件,帮助克服设计的复杂性。这种方法是改良性的,因为它企图适应设计者目前的工作风格。设计者的传统工作习惯倾向于先设计出他自己的积木块,然后用这些积木块作元件来实现更高级的结构,即所谓的自底向上的方法。这种方法的优点是设计质量好,但费时间,且容易出错,需要多次迭代和比较。

(2) 可编程的方法

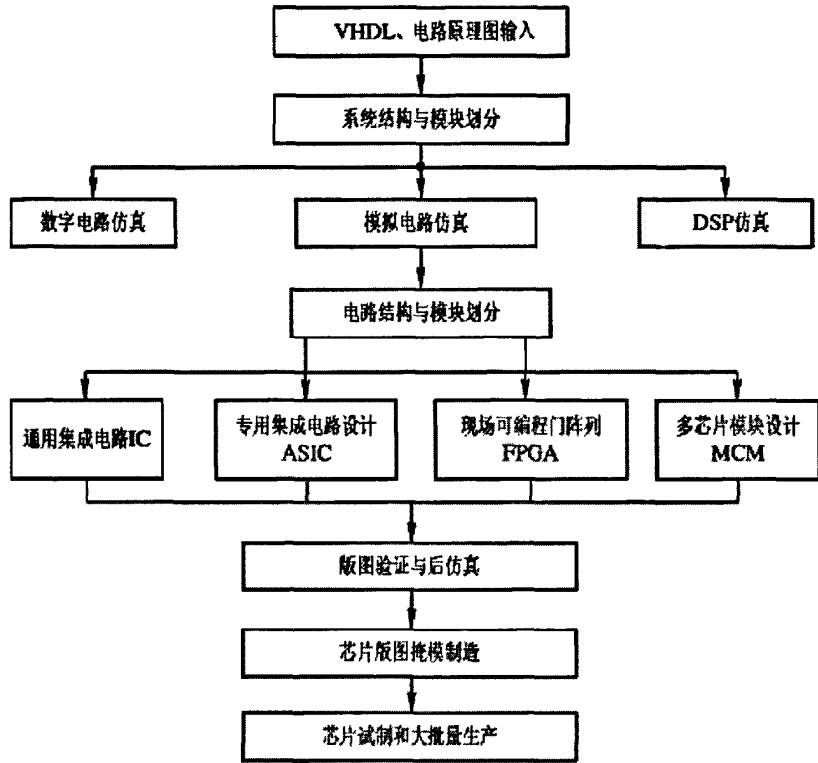
可编程的方法就是利用编译的方法来设计数字系统。这种方法认为知识是算法的,而且可以写出变换程序,从问题的高级描述能自动生成或综合出它的全部或某些部分的解。可编程逻辑阵列、复杂可编程逻辑器件以及都有很好的应用。

(3) 智能的方法

智能的方法就是用人工智能方法设计复杂数字系统。这种方法认为,设计者的知识能存储到一个专家系统的知识库中。专家系统分为三类:一是概念,包括问题域中的基本术语,可从教科书中获得;二是规则,它描述特定的情景与希望完成的动作,这种知识是以经验为基础,是从专家那里得到的;三是策略,它是一些过程,辅助引导搜索知识库,而且当有几个同等合理的规则可使用时,可以帮助解决选择的矛盾。目前实用专家系统有专家布局器,专家布线器等。

目前,符合层次化、结构化设计的 EDA 系统的基本功能如图 2.2 所示。该图列举了和通用数字系统设计软件包的主要模块。

在通用数字系统设计中,用到的主要技术有 VHDL 语言、建模仿真、设计综合、可编程器件以及在深亚微米条件下的延迟计算、静态时序分析。



T 图注释

图 2.2 EDA 系统功能示意图

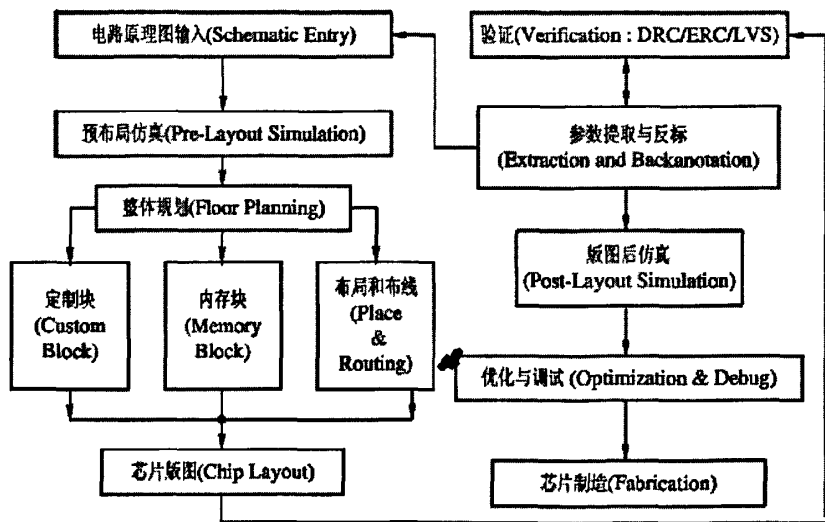


图 2.3 通用数字系统设计软件包的主要模块示意图

2.2.4 基于 VHDL 的 EDA 设计工具

随着半导体技术的迅速发展,在现代数字系统设计中,FPGA 和 CPLD 的使用越来越广泛。与此同时,基于大规模可编程逻辑器件的 EDA 硬件解决方案也被广泛采用。一般来说,EDA 解决方案均采用计算机自顶向下的设计方式;在底层设计时对逻辑进行必要的描述,并依赖特定的软件执行逻辑优化与器件映射,最后再使用由各芯片生产厂商提供的编译器执行布线和网表优化、虽然对于简单的逻辑,采用原始逻辑图或布尔方程输入可以获得非常有效的结果,但对于复杂的数字系统设计,应用以上两种方案就很容易产生错误,而必须依靠一种高层次的逻辑输入方式。

为了适应自顶向下的设计方法,必须提供良好的自顶向下的设计环境、硬件描述语言提供了丰富的库和与工艺无关的设计输入方式,其中 VHDL 的应用成为新一代 EDA 解决方案中的首选、VHDL 的应用已成为当前以及未来 EDA 解决方案的核心,更是复杂数字系统设计的核心。

(1) 应用 VHDL 的理由

严格地讲,VHDL 是一种用来描述数字逻辑系统的编程语言、它源于美国政府于 1980 年开始启动的超高速集成电路计划。在这一计划的执行过程中,专家们认识到需要有一种标准的语言来描述集成电路的结构和功能,由此,VHDL 便诞生了,并很快被美国电气和电子工程师协会所承认。

VHDL 是为了满足逻辑设计过程中的各种需求而设计的、首先,它能形式化地抽象表示电路的结构和行为,支持逻辑设计中层次与领域的描述、其二,支持电路描述由高层到低层的综合和转换,便于文档管理,易于理解 and 设计重用、最后,VHDL 具有电路仿真与验证功能,可以保证设计的正确性、用户甚至不必编写任何测试向量便可以进行源代码级的调试、而且,设计者可以非常方便地比较各种方案之间的可行性及其优劣,而不需做任何实际的电路实验。

VHDL 能够精确而且简明地描述数字电子系统,可用于从系统级到门级的描述,特别是能以非常抽象的形式反映出系统最本质的性能、VHDL 可为系统级模拟提供方便的手段,使得系统实现之前就可以评价系统的性能、鉴于 VHDL 具有以上诸多优点,只要开发者拥有计算机高级语言的基础,便可以轻松地掌握 VHDL,

使硬件工作软件化。

(2) VHDL 的模块组织

在应用 VHDL 描述数字系统结构时, 我们使用实体和结构 J 架构。

实体描述了数字系统的输入/输出接口, 同时还定义了一些全局常量以及与其它电路之间必要连接的拓扑结构、但在实体中, 我们并不对电路的逻辑做任何描述, 可将其看成是一个所谓的逻辑黑盒子。很明显, VHDL 遵循 EDA 解决方案中自顶向下的设计原则, 并能够保持良好的接口兼容性。

结构对具体的逻辑进行描述, 主要由变量声明和语句描述两部分构成、变量有信号型和内部节点型; 语句包括组合逻辑语句、同步时序逻辑语句、对于异步时序, 则依靠语句排列顺序再加上延时语句来完成。

VHDL 在描述同步时序逻辑时主要使用 process 结构和 wait 语句。在一个结构体中可以同时拥有几个 process, 它是用于描述时序的基本单元。VHDL 描述硬件行为时, 最重要的就是依据时间的变化对信号做出相应的分配和赋值。

在 EDA 解决方案中应用 VHDL 有助于缩短数字系统的开发周期。总的来讲, 一般厂商均遵循 VHDL -87 和 VHDL -93 标准。

2.2.5 数字系统的设计流程

半导体技术和计算机技术的发展, 使数字系统的设计理念和设计方法都发生了深刻的变化。以前, 数字系统主要由一些固定功能的器件加上一定的外围电路构成模块, 再由这些模块进一步形成各种功能电路。在设计时, 几乎没有灵活性可言, 而且, 设计一个系统所需的芯片种类多且数目大。PLD 器件和 EDA 技术的广泛应用给今天的硬件设计者提供了强有力的支持, 同时也使得数字系统的设计思想和设计方式发生了根本性的变化。一般来说, 数字系统的设计有两种思路: 一种是自顶向下的设计思路, 另一种是自底向上的设计思路。

1、自顶向下设计流程

随着集成电路技术的发展, 电子系统的规模与复杂度越来越高, 使用传统的自底向上的设计方法已越来越不适应, 而自顶向下的设计方法却越来越显示出其优越性。

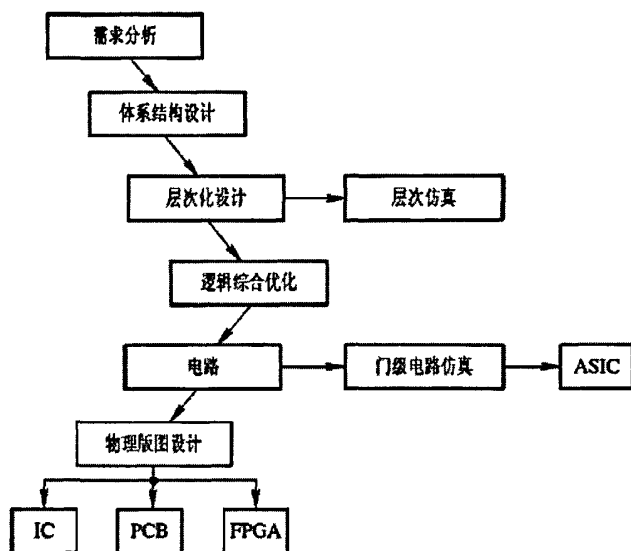


图2.4 自顶向下 (Top-Down) 的设计方法

采用的自顶向下 (Top-Down) 的设计方法,则是从系统总体要求出发,在顶层进行功能方框图的划分和结构设计,并用对高层次的VHDL系统行为进行描述,在系统一级进行仿真、验证然后用逻辑综合优化工具生成具体的门级逻辑电路的网表,最后完成系统硬件的整体设计。这种自顶向下的数字电路设计主要包括以下个抽象层次:(1)行为级,或称算法级。描述的是设计在算法中的行为,独立于硬件实现,电路结构并未确定。(2)寄存器传输级(RTL),或称结构级。描述设计电路的体系结构,也独立于硬件实现。RTL网表由各模块如、或乘法器等级联而成的结构网表组成。模块主要由组合逻辑和寄存器组成,在局部还可能包含一些可逻辑综合的行为描述。(3)逻辑门级。硬件实现已经基本确定,门级设计网表是与芯片制造商工艺库直接相关的网表,电路中所有的工艺器件都要确定。(4)电路级。表示电路设计可以转入制造。

自顶向下的设计方法如图所示,它是指设计电子系统先从系统最抽象的层次出发,作高层次仿真,经过仿真验证后再经整体规划将系统行为操作分为子系统。各个子系统作行为仿真,它和高层次仿真的结果比较,易于发现和修正早期结构设计的错误,当验证合格以后,再经逻辑综合工具自动得到优化的和具体工艺相关的门级描述,作门级仿真,并和高层次仿真的结果作对比,验证合格后经物理设计,即

可得到合格的产品。

可以看出，自顶向下的设计方法有许多突出的优点：

(1) 它克服了大规模电子系统高复杂度所带来的问题，系统可以层次式地划分为易于处理的子系统，再层次式地求精。

(2) 各子系统可以给设计组中的成员同时设计，也加快了设计速度。

(3) 设计错误可以在早期发现，极大地减少了设计的迭代次数。

(4) 逻辑综合优化之前的设计工作和具体采用什么工艺生产芯片是无关的，因而设计的可移植性好，当要采用新的工艺时，可以直接从综合开始。

(5) 自顶向下的设计方法增加了一次性设计成功的可能性。

2、自底向上设计流程

自底向上的方法是一种传统的设计思想、设计者首先将各种基本单元，如各种逻辑门以及加法器。选通器等做成基本单元库，然后调用它们，逐级向上组合，直到设计出自己满意的系统为止。由于缺乏对整个系统的规划，目前这种方法在复杂数字系统设计中，主要应用在建库、IP 模块调整等设计中，只是作为自顶向下方法的一种补充。

3、正向设计和逆向设计

复杂数字系统设计一般采用自顶向下的设计方法，所谓正向设计，也是一种自顶向下的设计方式，它包括从芯片设计到芯片封装的一系列过程。正向设计的流程如图所示。

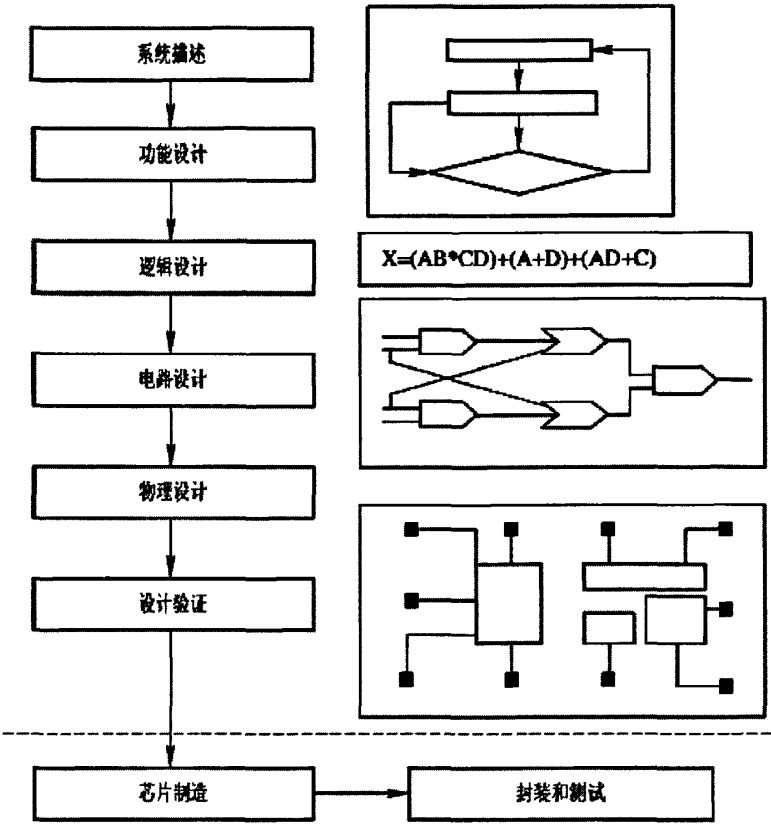


图 2.5 芯片的正向设计流程

(1) 系统描述。系统描述包括系统功能。性能。物理尺寸。设计模式。制造工艺。设计周期。设计费用等。

(2) 功能设计，是将系统功能的实现方案设计出来，通常是给出系统的时序图及各子模块之间的数据流图。

(3) 逻辑设计。逻辑设计是将系统功能结构化，通常以文本。原理图、逻辑图表示设计结果，有时也采用布尔表达式来表示设计结果。

(4) 电路设计。电路设计是将逻辑设计表达式转换成电路实现。

(5) 物理设计。物理设计 I 或称版图设计是 VLSI 设计中最费时的一步、它要将电路设计中的每一个元器件（包括晶体管。电阻。电容。电感等）以及它们之间的连线转换成集成电路制造所需要的版图信息。

(6) 设计验证。设计验证是在版图设计完成以后非常重要的一步工作、它主

要包括设计规则检查、网表提取、电学规则检查、版图网表结构与电路原理图对比和寄生参数提取。

(7) 芯片制造。在复杂数字系统设计中,制造和设计是分离的,一般先做好前端设计后,由第三方公司针对某一特定工艺线进行整合,制造出掩模版并到工艺线上流片。

(8) 封装和测试。在完成芯片制造后,要进行封装和测试、安置在印制电路板上的芯片可封装成双立直插式或引脚阵列式。用于多芯片模上的芯片可不封装。

逆向设计主要用于分析、仿制别人芯片设计,其流程如下:

(1) 提取横向尺寸。剖开芯片封装,进行分块扫描照相,一般需放大几百倍以上,提取出芯片的复合版图;把每张照片拼成整个芯片的复合版图;由产品的复合版图提取电路图、器件尺寸和设计规则;进行电路模拟,验证所提取的电路是否正确;如果模拟正确,局部改造或创新,画出更先进的芯片版图。

(2) 提取纵向尺寸。用扫描电镜、扩展电阻仪等提取氧化层厚度、金属膜厚度、多晶硅厚度、结深、基区宽度等纵向尺寸和纵向杂质分布。

(3) 测试产品的电学参数。电学参数包括开启电压、薄膜电阻、放大倍数、特征频率等。

(4) 确定工艺、有了纵向尺寸和电学参数后,可以确定芯片制造所需的工艺参数、工艺条件和工艺流程等。

由上述内容可见,在由芯片提取出电路图和逻辑关系后,正向设计和逆向设计的设计过程都是一样的了。

第三章 出租车计费系统的设计

3.1 出租车计费规则

实际中出租车的计费工作原理一般分成 4 个阶段：

(1) 车起步开始计费。首先显示起步价（本次设计起步费为 10.0 元），车在行驶 3 km 以内，只收起步价 10.0 元。

(2) 车行驶超过 3 km 后，按每公里 2 元计费（在 10.0 元基础上每行驶 1 km 车费加 2.0 元），车费依次累加。

(3) 行驶路程达到或超过 9 km 后（或者是车费达到 20 元），每公里加收 50% 的车费，车费变成按每公里 3.0 元开始计费。

(4) 车暂时停止（行驶中遇红灯或中途暂时停车）按时间计费，每 3 分钟计费 0.5 元。

若停止则车费清零，等待下一次计费的开始。

本设计中出租车有起动键，停止键和暂停键，起动键为一脉冲信号，用一个按键来实现，当有电平有变换时，表示该计费系统已启动，在汽车运动的过程中，采用一个脉冲信号（计数脉冲）来实现路程的计数，该信号代替了实际的车行里程测速的脉冲信号，在根据相应的计费公式来算出车费，同时显示在数码管上，在刚开始数码管显示的是起始价；当停止键为高电平时，表示汽车停止前进，同时停止发生脉冲，此时路程计数和车费清零；当暂停键为高电平时，表示汽车暂停并停止发生脉冲，此时路程计数暂停，时间计数开始运行，最后把等待时间也这算成车费的一部分。

3.2 基本设计思想

主要分为三个模块。分别为：秒分频模块、计量模块和译码显示模块。

(1) 脉冲生成模块使整个系统的同步工作，把系统提供的 32M 的晶振频率进行分频，得到我们所需要的秒信号（便于在以后的计时中用到）。同时利用此 32MHz 生成电路工作所需要的 32Hz 工作脉冲。

(2) 计量控制模块也是系统中一个十分重要的模块，它由三个部分组成，分

别是计价部分、计时部分和计程部分，是计程车计价器系统多功能实现的保证。

计价部分又包括两个内容，一个是正常车行里程数这算成价格费用，根据车行路程的远近，还需要增加相应的附加费用；另一部分是在车行过程中等待红灯等暂停状态下，等候时间折算成的价格费用。

在控制模块中设计中还包括按键部分：按键主要是有起动，暂停和停止三个按键，当这三个按键按下时，相对应的电平变为高电平，把该电平传输给控制模块做相应的操作。其实现的过程主要是在硬件上做相应的设计就可以实现，电路如图 3.1。

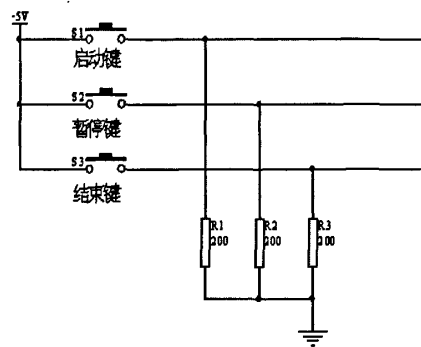


图 3.1 按键模块

（3）最后一个译码显示模块作用在于把计程车的工作情况（价钱、行程数）反馈给乘客。车费的显示由动态扫描电路来完成。用专用模块来实现，完成数据的输入即动态数据的显示。

三个模块有机地结合在一起，实现了基于 FPGA 的多功能计程车计价器的设计。通过分析可以设计出系统的顶层框图如图 3.2 所示：

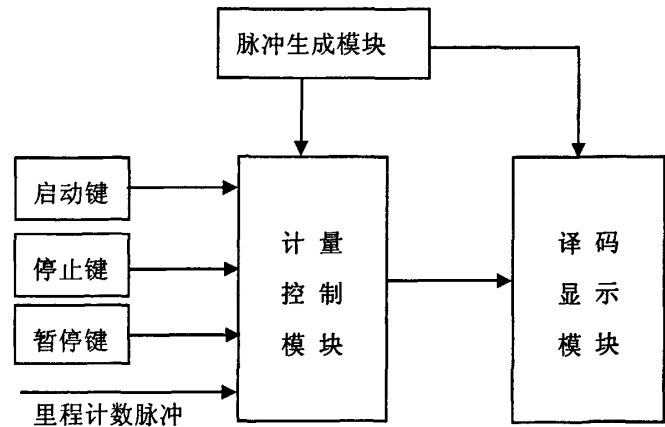


图 3.2 系统的顶层框图

第四章 出租车计费系统的实现

4.1 系统的顶层原理图

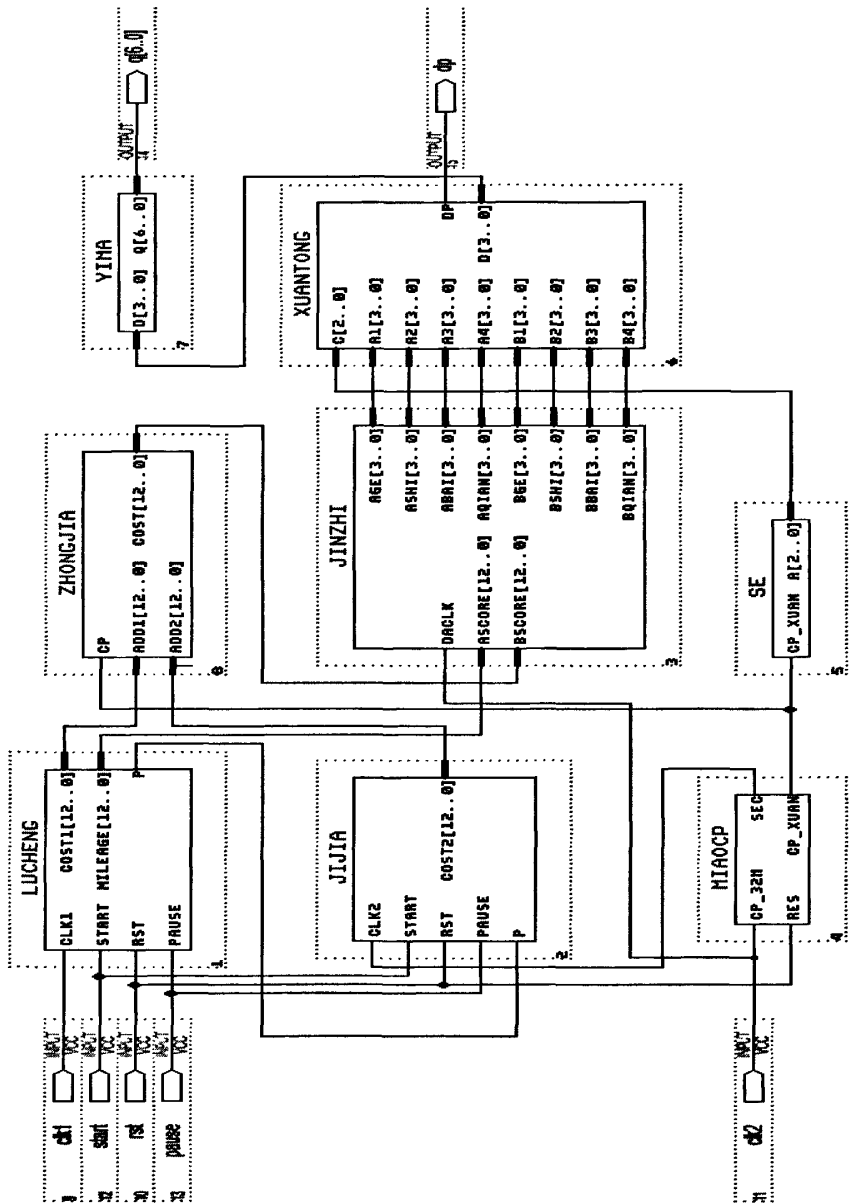


图 4.1 系统顶层原理图

该原理图中输入部分分别是计价器启动信号: `start`; 计价停止信号: `rst`; 车辆暂停运行信号: `pause`; 以及两个秒冲信号, 分别是每行进 10 米一个脉冲的路程脉冲信号: `clk1`; 外接设备工频脉冲信号: `clk2` (为设计器件的工作脉冲, 并由此生成秒脉冲等相关脉冲信号, 本设计选取信号频率为 32MHz)。

输出有两本分, 分别为 `d[6-0]`: 七段数码管的现实码; `DP`: 小数点信号。此数据输送到相应的数码管上, 轮流显示出行驶路程和相应费用。

其中各模块的功能是:

Lucheng: 输入路程脉冲, 计算出相应的行驶路程数向后级传输, 同时把行驶路程折算成行驶费用。

Jijia: 在出租车营运过程中, 必然会遇到等红灯、堵车等情况, 当车速低于一定值是, 此模块开始工作, 把等候时间折算成费用。

Zhongjia: 此模块把行驶路程折算的价格费用与等待时间折算的费用相加, 计算出乘客应付的总价。

miaocp: 用来生成计时所需的秒脉冲, 以及模块 **zhongjia** 所需的工作脉冲 32Hz。同时送到分频模块 **SE** 中, 进行 8 分频, 以作为选通脉冲, 使得数码管显示能比较稳定的为人眼所看到。

Jinzhi: 该模块是把之前采用十进制码总计得的路程、费用, 换算成 BCD 码, 分别按个、十、百、千的顺序向后传输。

Xuantong: 把已经折算成 BCD 码的路程和费用数, 按一定的速度, 按排列顺序每次向后级的数码管传输一位。同时选择显示时小数点所处的位置。

最后是 **Yima**: 把选择传输到数码管显示的那个 BCD 码数转换成数码管的显示码。

4.2 程序流程图

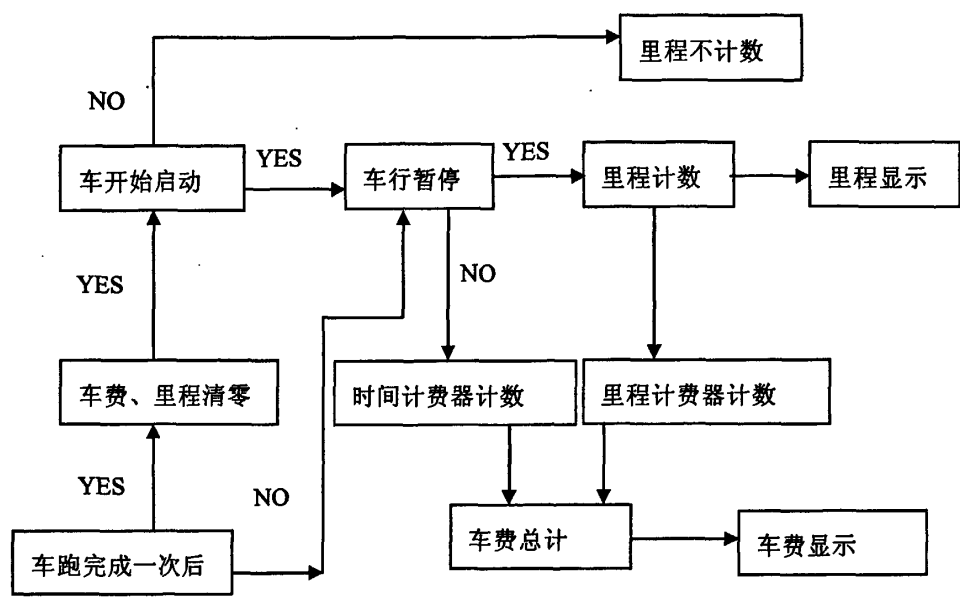


图 4.2 程序流程图

设备先进行“start”按键的判断，再进行“stop”按键的判断，再进行“pause”的判断，当状态为运动时，每 10 米传感器的脉冲计数加一，当计数到达一定的要求时，里程数加一，同时对价钱按要求进行计数，最终让它们显示在数码管上。

4.3 系统各功能模块的实现

4.3.1 模块 lucheng 的实现

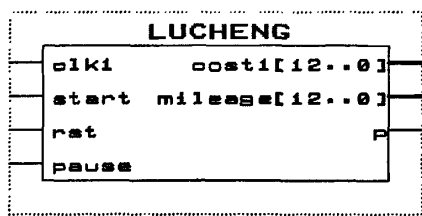


图 4.3 模块 lucheng

模块 lucheng 如图 4.3。输入端口为 START、RST、PAUSE 分别为汽车起动、停止、暂停按键。

当车辆开始进行计费（即 start 输入端口置“1”）时，车辆每行驶过十米路程，

就输入一路程脉冲（即 `clk1`）。根据输入的路程脉冲，开始进行加计数，通过 `mileage` 输出口输出计算得出的行驶路程数；同时也把行驶路程折算成费用通过 `cost1` 口输出。当计费停止（即 `start` 口置“0”）或车辆停止运作（即 `rst` 口为“1”）时，相关数据复位、清零。

在把路程折算成费用是按照如此规律进行：车辆起步价为 10 元 3 公里，超出部分计费标准为：行驶路程小于 9 公里或者费用低于 20 元的部分，按每公里 2 元计费，在程序设计时以 500 米加价 1 元计费；超过 9 公里的部分按每公里 3 元计费，在程序中以 500 米 1.5 元加价。

同时该模块还生成控制信号 `p`，当在 3 公里的起步价内，信号置“0”，使计时模块不工作；当超出起步范围时，开始发出“1”信号。

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

entity lucheng is
port (clk1,start,rst,pause:in std_logic;
      cost1,mileage:out integer range 0 to 8000;
      p: out std_logic);
end lucheng;

architecture analog of lucheng is
begin
process(clk1,start,rst,pause)
variable a,c:std_logic;
variable b:integer range 0 to 50;
variable cos,mile:integer range 0 to 8000;
begin
if (clk1'event and clk1='1') then
    if(rst='1') then          //车停止，未开动时//
        cos:=0;
        b:=0;
```

```
    mile:=0;
    elsif(start='0')then      //计价器未开始计价工作时//
        a:='0';
        cos:=100;      //起步价 10 元//
        mile:=0;
    elsif(start='1' and pause='0')then
        mile:=mile+1;
        b:=b+1;
    end if;
    if(b>=50)then
        a:='1';
        b:=0;
    else
        a:='0';
    end if;
    if (mile<300) then
        c:='0'1;
    elsif (cos<200 and a='1')then
        //超过起步路程后，价格小于 20 元，2 元/公里//
        cos:=cos+10;      //每次计费是 1 元一加//
        c:='1'1;      //超出起步价标志//
    elsif(cos>=200 and a='1') or (mile>=900 and a='1')then
        //超过 20 元或 9 公里后每公里加收 50%车费//
        cos:=cos+15;      //每次计费是 500 米加 1.5 元//
        c:='1'1;
    end if;
end if;
cost1<=cos;
mileage<=mile;
p<=c;
end process;
end analog;
```

4.3.2 模块 jijia

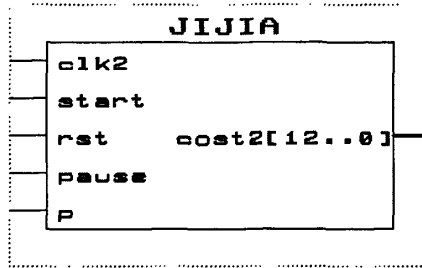


图 4.4 模块 jijia

模块 jijia 如图 4.4，输入端口为 START、RST、PAUSE 分别为汽车起动、停止、暂停按键。

该模块实现了计价过程中把等待时间折算成费用的功能。当车辆行驶超过起步范围（即 p 口为“1”）时，该模块开始运行，每暂停等待三分钟加费用 0.5 元。输出口 cost2 为等待时间折算成的总费用。

其中的 clk2 为计时用秒脉冲。由模块对输入的 32MHz 脉冲进行分频得到。

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

entity jijia is
port (clk2,start,rst,pause,p:in std_logic;
      cost2:out integer range 0 to 8000);
end jijia;

architecture await of jijia is
begin
process(clk2,start,rst,pause,p)
variable a:std_logic;
variable b:integer range 0 to 180;
variable cos:integer range 0 to 8000;
begin
if(clk2'event and clk2='1')then
```

```

    if(rst='1') or (start='0')then
        //车辆停止时，数据清零//
        cos:=0;
        b:=0;
        a:='0';
    elsif(start='1'and pause='1')then
        //计价器工作且车辆暂停时，开始 3 分钟计时//
        b:=b+1;
    end if;
    if(b>=180)then
        a:='1';
        b:=0;
    else
        a:='0';
    end if;
    if(a='1'and p='1')then
        //当超出起步范围，且计时到 3 分钟时，加 0.5 元费用//
        cos:=cos+5;
    end if;
end if;
cost2<=cos;
end process;
end await;

```

4.3.3 模块 zhongjia

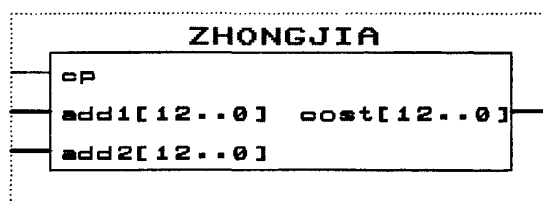


图 4.5 模块 zhongjia

模块 zhongjia 如图 4.5 所示。输入口 add1 和 add2 分别是路程折算成的费用和

等待时间折算成的费用,该模块把两者相加,输出口 cost 输出乘客应付的总的费用。

Cp 为电路的工作频率,为 32Hz。

```
library ieee;
```

```
use ieee.std_logic_1164.all;
```

```
use ieee.std_logic_unsigned.all;
```

```
entity zhongjia is
```

```
port (cp:in std_logic;
```

```
      add1,add2:in integer range 0 to 8000;
```

```
      cost:out integer range 0 to 8000);
```

```
end zhongjia;
```

```
architecture adder of zhongjia is
```

```
begin
```

```
process(cp,add1,add2)
```

```
variable a:integer range 0 to 8000;
```

```
begin
```

```
  if (cp'event and cp='1')then
```

```
    a:=add1+add2;  //把路程和等待时间折算的价格加起来//
```

```
  end if;
```

```
  cost<=a;
```

```
end process;
```

```
end adder;
```

4.3.4 模块 jinzhi 的实现

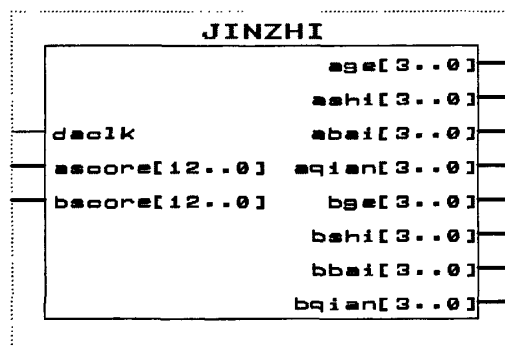


图 4.6 模块 jinzhi

模块 jinzhi 如图 4.6。此数据转换模块是一个模为 10、步长可变的加法计数器，可将计费模块输出的车费和路程转换成 4 位的十进制数，便于数码管显示。

输入口 ascore 为总计的路程数输入口，bscore 为总费用的输入口，两者均以 2 进制码进行十进制编码计数，通过该编码器生成 BCD 码数，输出口分别以 BCD 码表示其个、十、百、千位上的数据。

Dac1k 是工作脉冲，也为 32MHz。

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

entity jinzhi is
    port(dac1k:in std_logic;
    ascore,bscore:in integer range 0 to 8000;
    age,ashi,abai,aqian,bge,bshi,bbai,bqian:out std_logic_vector(3 downto 0));
end jinzhi ;

architecture rtl of jinzhi is
begin
    process(dac1k,ascore) //总路程的进制转换//
        variable comb1:integer range 0 to 8000;
        variable comb1a,comb1b,comb1c,comb1d:std_logic_vector(3 downto 0);
```

```
begin
if(dacclk'event and dacclk='1')then
if(comb1<ascore)then           //加 1 计数还不到应加次数时//
if(comb1a=9 and comb1b=9 and comb1c=9)then
//个、十、百位计数到 9，再来信号，千位加 1，个、十、百位清零//
comb1a:="0000";
comb1b:="0000";
comb1c:="0000";
comb1d:=comb1d+1;
comb1:=comb1+1;
elsif(comb1a=9 and comb1b=9)then
//个、十位计数到 9，再来信号，百位加 1，个、十位清零//
comb1a:="0000";
comb1b:="0000";
comb1:=comb1+1;
comb1c:=comb1c+1;
elsif(comb1a=9)then
//个位计数到 9，再来信号，十位加 1，个位清零//
comb1a:="0000";
comb1b:=comb1b+1;
comb1:= comb1+1;
else
comb1a:= comb1a+1;
comb1:= comb1+1;
end if;
else
ashi<= comb1b;
age<= comb1a;
abai<= comb1c;
aqian<= comb1d;
comb1:=0;
comb1a:="0000";
```

```
comb1b:="0000";
comb1c:="0000";
comb1d:="0000";
end if;
end if;
end process;
process(dac1k,bscore)//总费用的进制转换//
variable comb2:integer range 0 to 8000;
variable comb2a,comb2b,comb2c,comb2d:std_logic_vector(3 downto 0);
begin
if(dac1k'event and dac1k='1')then
if(comb2<bscore)then           //加 1 计数还不到应加次数时//
if(comb2a=9 and comb2b=9 and comb2c=9)then
comb2a:="0000";
comb2b:="0000";
comb2c:="0000";
comb2d:=comb2d+1;
comb2:=comb2+1;
elsif(comb2a=9 and comb2b=9)then
comb2a:="0000";
comb2b:="0000";
comb2:= comb2+1;
comb2c:= comb2c+1;
elsif(comb2a=9)then
comb2a:="0000";
comb2b:=comb2b+1;
comb2:=comb2+1;
else
comb2a:= comb2a+1;
comb2:= comb2+1;
end if;
else
```



```

bshi<=comb2b;
bge<=comb2a;
bbai<=comb2c;
bqian<=comb2d;
comb2:=0;
comb2a:="0000";
comb2b:="0000";
comb2c:="0000";
comb2d:="0000";
    end if;
end if;
end process;
end rtl;

```

4.3.5 模块 xuantong 的实现

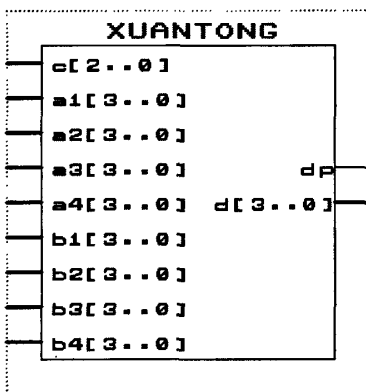


图 4.7 模块 xuantong

本设计的显示部分利用视觉暂留效应，采用动态扫描电路，将进制转化后的里程数和车费的四位十进制数显示在四位的数码管上，节约了硬件资源和能源，具有硬件电路简单，软件工作量少，使产品的性价比高等优点。

模块 xuantong 如图 4.7,经过该八进制选通模块将车费和路程轮流显示出来。

其中 a1、a2、a3、a4 分别是路程的个、十、百、千位的数值输入口，b1、b2、b3、b4 分别是费用的个、十、百、千位的数值输入口。c 是选通地址码的输入口，根据输入的地址码，模块每次只有一位数字向后传输到输出口 d，同时输出数码管

小数点的显示控制信号,使得路程显示为 00.00 公里,费用显示为 000.0 元。

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

entity xuantong is
port(c:in std_logic_vector(2 downto 0);
dp:out std_logic;
a1,a2,a3,a4,b1,b2,b3,b4:in std_logic_vector(3 downto 0);
d:out std_logic_vector(3 downto 0));
end xuantong;

architecture rtl of xuantong is
begin
process(c,a1,a2,a3,a4,b1,b2,b3,b4)
variable comb:std_logic_vector(2 downto 0);
begin
    comb:=c;
    case comb is
        when "000"=>d<=a1;
            dp<='0';
        when "001"=>d<=a2;
            dp<='1';
        when "010"=>d<=a3;
            dp<='0';
        when "011"=>d<=a4;
            dp<='0';
            when "100"=>d<=b1;
            dp<='0';
            when "101"=>d<=b2;
            dp<='1';
            when "110"=>d<=b3;
```

```

dp<='0';
when"111"=>d<=b4;
dp<='0';
when others=>null;
end case;
end process;
end rtl;

```

4.3.6 模块 SE 的实现

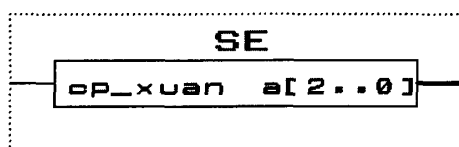


图 4.8 模块 SE

模块 se 如图 4.8，是一个 8 进制数生成器，该模块利用模块工作用的 32Hz 脉冲产生模块 xuantong 所需要的选通地址信号。

```

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

entity se is
port(cp_xuan:in std_logic;
      a:out std_logic_vector(2 downto 0));
end se;

architecture rtl of se is
begin
process(cp_xuan)
variable bt:std_logic_vector(2 downto 0);
begin
if (cp_xuan'event and cp_xuan='1') then
    if (bt="111") then
        bt:="000";
    end if;
end if;
end process;
end rtl;

```

```

else
    bt:=bt+1;
end if;
end if;
a<=bt;
end process;
end rtl;

```

4.3.7 模块 yima 的实现

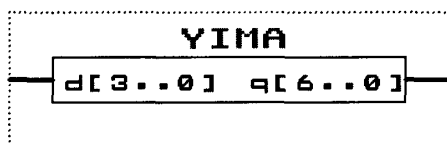


图 4.9 模块 yima

此模块为译码模块，把 0-9 的 BCD 码译成数码管的显示码，如图 4.9。输入口 d 输入 xuantong 模块选出的要显示的 BCD 码数，译成数码管的显示码输出。

在本设计中数码管是共阴数码管，共阴指数码管的公共端接负极，而各段接正极让其在数码管上显示出相应的数据，数码管上的每一段都是由一个端口来驱动，亮度比较高。

数码管的动态扫描是采用 8 位数码扫描显示，其中每个数码管的 8 段 a, b, c, d, e, f, g 和 dp（小数点）都分别连在一起，4 个数码管分别由 4 个选通信号 D1, D2, D3, D4 来选择，被选通的数码管就按要求显示数据，其余的被关闭，在下面动态扫描的 VHDL 代码中用变量 scan 来选择，而在运行中主要是采用 CASE 语句，CASE 语句不存在优先级的问题，条件的输出值只有一种可能，选择值中不允许出现重复，所有表达式的值都必须穷举，不能穷举的用 OTHERS 表示。

在本设计中只采用了一个数码管，对数据的个、十、百、千位实现轮流显示，如需分别显示路程和费用，并且同时显示其个、十、百、千位，只需去除 xuantong 模块，在 jinzhi 模块的输出端分别接入 8 个 yima 模块即可。显示需要的小数点通过硬件线路的连接来显现功能。

```

entity yima is
    port(d:in std_logic_vector(3 downto 0);

```

```

        q:out std_logic_vector(6 downto 0));
    end yima;

    architecture rtl of yima is
    begin
        process(d)
        begin
            case d is
                when "0000"=>q<="0111111";
                when "0001"=>q<="0000110";
                when "0010"=>q<="1011011";
                when "0011"=>q<="1001111";
                when "0100"=>q<="1100110";
                when "0101"=>q<="1101101";
                when "0110"=>q<="1111101";
                when "0111"=>q<="0100111";
                when "1000"=>q<="1101111";
                when others=>q<="1101111";
            end case;
        end process;
    end rtl;

```

4.3.8 秒脉冲及 32Hz 脉冲生成器

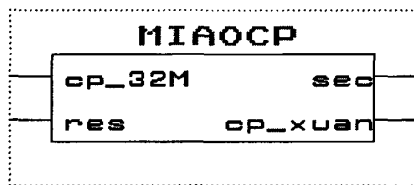


图 4.10 模块 miaocp

由于设计中的显示部分采用动态扫描电路，所以在此之前要进行设置动态扫描的频率，其频率不能低于人视觉暂留现象的频率，视觉暂留现象指当人眼所看到的影像消失后，人眼仍能继续保留其影像 0.1-0.5 秒左右的图像。本设计中输入的系

统时钟为 32MHz 的, 在 CPLD 中对其进行分频, 再对数码管的地址进行扫描。

模块 miaocp 如图 4.10。输入口 res 为车辆停运信号输入口, 当车辆停止运行时, 该模块也停止工作, 清零; 当车辆运行时开始对输入的 32M 脉冲信号进行分频, 分别从输出口 sec 得到秒脉冲, cp_xuan 得到 32Hz 工作脉冲。

由于此处生成了秒脉冲, 可以利用此信号对设计添加功能, 如时钟功能。如晚间营运需增添附加费用, 可在增加时钟功能的基础上, 在费用计算的模块中添加相应时钟控制程序, 使得白天和夜间采用不同的起步价格。

```
library IEEE;
    use IEEE.STD_LOGIC_1164.ALL;
    use IEEE.STD_LOGIC_ARITH.ALL;
    use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity miaocp is
    port (cp_32M,res:in std_logic;
          sec,cp_xuan:out std_logic);
end miaocp;

architecture adder of miaocp is
begin
    process(cp_32M,res)
        variable c : integer range 0 to 32000000;
        variable d : integer range 0 to 1000000;
    begin
        if res='1' then
            c:=0;
        elsif rising_edge(cp_32M) then
            if (c<32000000) then
                c:=c+1;
            else
                c:=1;
            end if;
        end if;
```

```
        IF (d<1000000) then
            d:=d+1;
        ELSE
            d:=1;
        end if;
    end if;
    if (c>=16000000) then
        sec<='1';
    else
        sec<='0';
    end if;
    if (d>500000) then
        cp_xuan<='1';
    else
        cp_xuan<='0';
    end if;
    end process;
end adder;
```

第五章 系统仿真

5.1 模块 lucheng 的仿真结果

对 lucheng 模块进行仿真，时序仿真结果如图 5.1，观察波形，当启动键（start）为一个脉冲时，表示汽车已起动，车费（cost1）显示初价，路程（mileage）开始计数；当停止键（stop）为‘1’时，表示汽车停止，车费（cost1）和路程（mileage）都为‘0’；当暂停键为‘1’时，车费（cost1）和路程（mileage）的值都保持不变。在仿真中，先设定车子停止-运行-停止一个周期，再设一个停止-运行-停止周期，其中对暂停功能进行检测，其结果基本符合要求。

输入端为：start, rst, pause, clk1

输出端为：p, cost1, mileage;

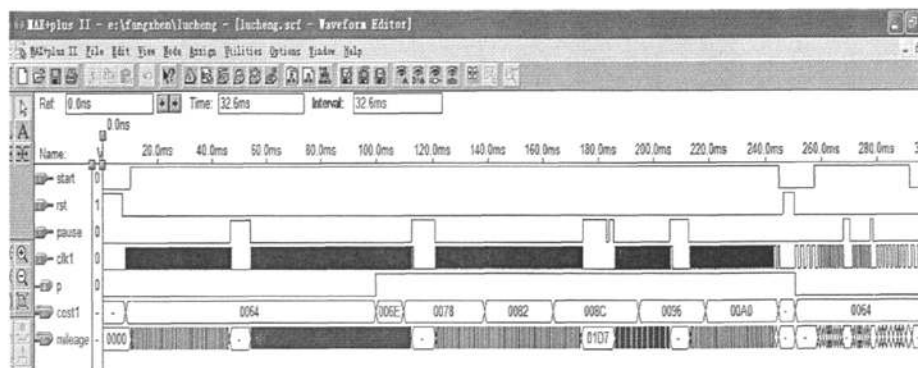


图 5.1 模块 lucheng 仿真波形图

由仿真结果可以看出由于车速的快慢，输入路程脉冲 clk1 的频率也会不断变化。当行驶路程小于 3 公里时，费用的显示一直为 10 元。随着行驶路程变长，费用按照预定规律，也随着不断增加。当费用超过起步价后，模块输出 p 信号。

当车辆停止（即 rst 为“1”）时，模块数据清零，车辆开始运行，但还未开始计价（即 rst 为“0”，start 为“0”）时，预置初始值，起步价 10 元。

本设计可计路程数为 0-80.00 公里，费用 0-800.0 元。

5.2 模块 jijia 的仿真结果

该模块是把车辆营运时的等待（如堵车，红灯等情况下的车辆暂停）时间折算成费用，仿真结果如图 5.2 所示：

输入端口为：stop, start, pause, p, clk2;

输出端口为：cost2;

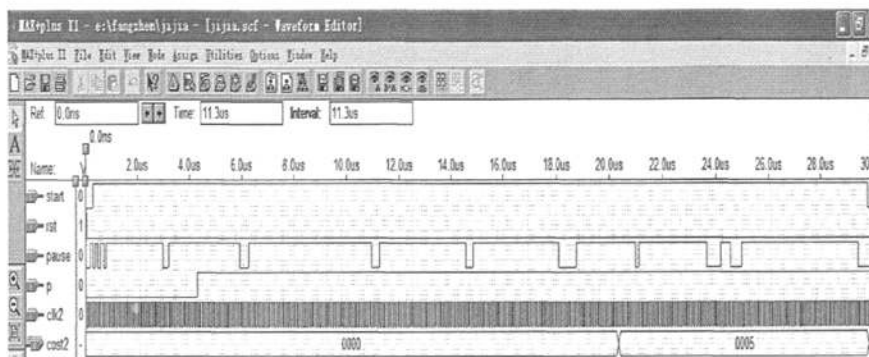


图 5.2 模块 jijia 仿真波形图

由仿真结果可以看出车辆启动后，产生秒脉冲 clk2 输入该模块。当行驶路程小于 3 公里或计费不超过 10 元时（即 p 为“0”），等待时间不折算成费用。随着行驶路程变长，超出起步路程或价格后（即 p 为“1”），车辆每暂停等待 3 分钟，费用增加 0.5 元。

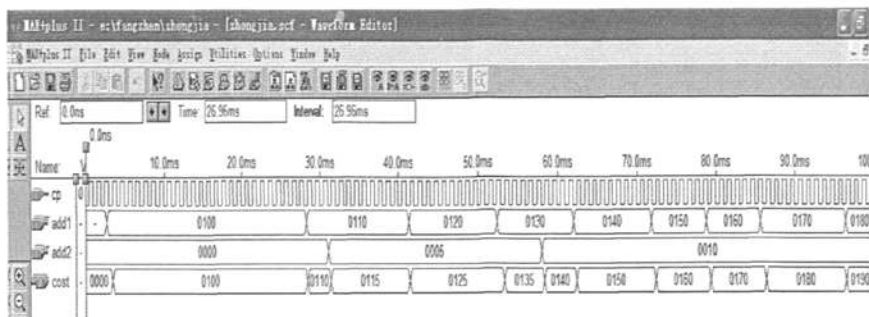
当车辆停止（即 rst 为“1”）时，模块数据清零，车辆开始运行（即 rst 为“0”，start 为“0”）时，数据清零。

5.3 模块 zhongjia 的仿真结果

该模块是一加法器，把车辆营运时的路程折算的费用和等待时间折算成的费用相加，计算出乘客所需付的总价格，仿真结果如图 5.3 所示：

输入端口为：cp, add1, add2;

输出端口为：cost;

图 5.3 模块 `zhongjia` 仿真波形图

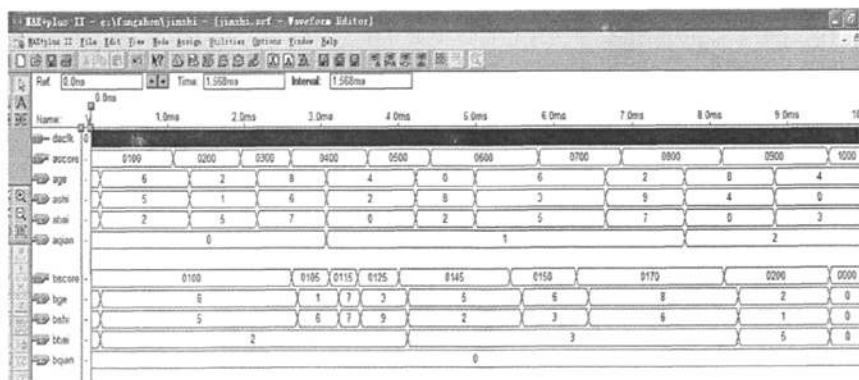
由仿真结果可以看出车辆启动后，每来一个工作脉冲 `cp`，模块就把输入的两部分费用加起来，计算出乘客需付的总费用。本设计可及费用范围为 0-800.0 元

5.4 模块 `jinzhi` 的仿真结果

该模块把车费和路程转化为 4 位 BCD 编码数，仿真结果如图 5.4 所示：

输入端为：`daclk, ascore, bscore`;

输出端为：`age, ashi, abai, aqian, bge, bshi, bbai, bqian`;

图 5.4 模块 `jinzhi` 仿真波形图

由仿真结果可以看出车辆启动后，每来一个工作脉冲 `daclk`，模块就把记录的路程数和费用数进行进制转换，把初始的二进制码记的十进制数转换成 BCD 码数输出。

此部分的仿真比较难实现，需要设置适当频率的 `daclk` 信号与输入数据 `ascore`、`bscore` 的长度相匹配，才能实现实际运行情况的仿真结果。

5.5 模块 xuantong 的仿真结果

该模块通过选通地址码把车费和路程轮流传输到数码管显示，仿真结果如图

5.5 所示：

输入端为：c, a1,a2,a3,a4,b1,b2,b3,b4;

输出端为：dp,d;

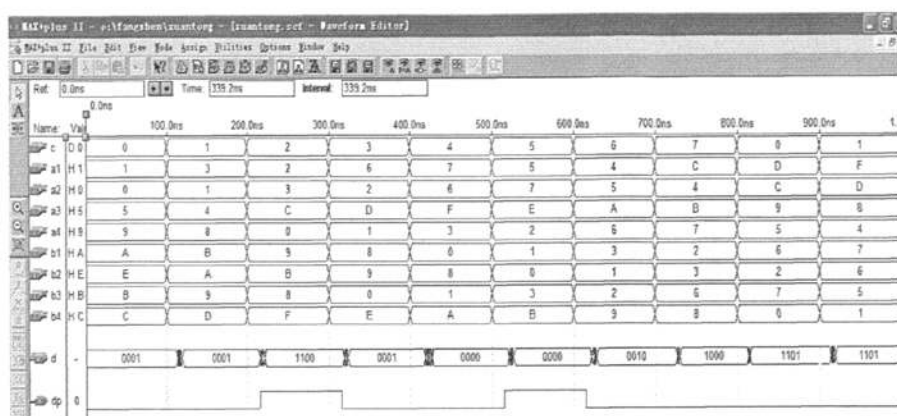


图 5.5 模块 xuantong 仿真波形图

由仿真结果可以看出每来一个选通地址 c，模块就把选到的那一位十进制输入值向后输出，同时在路程的百位和费用的十位显示小数点。输出顺序为路程个位——路程十位——路程百位+小数点——路程千位——费用个位——费用十位+小数点——费用百位——费用千位——路程个位，如此循环往复。由此可知路程可显示范围为 0-99.99 公里，费用为 0-999.9 元。

5.6 模块 se 的结果验证

模块 se 仿真如图 5.6 所示：

输入端为：cp_xuan

输出端为：a

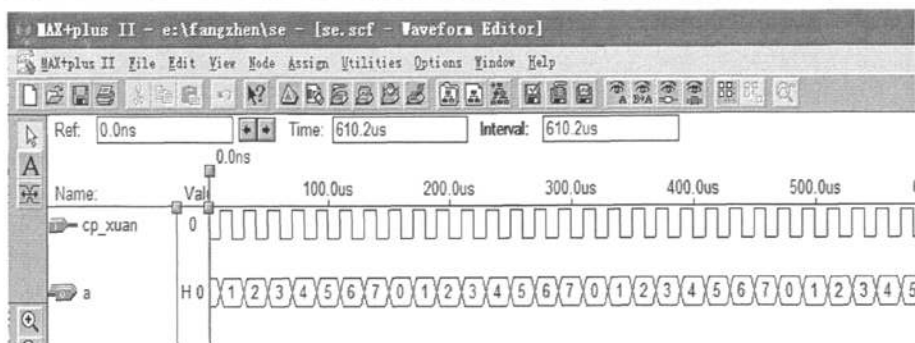


图 5.6 模块 SE 仿真波形图

由仿真结果可以看出此模块为八进制数生成器，对输入脉冲进行计数，输出 0-7 作为 xuantong 模块的地址选择码，当模块计数满 7 后再来脉冲信号，又回复到 0 重新开始计数。

5.7 模块 yima 的结果验证

译码模块 yima 的时序仿真结果如图 5.7 所示

输入端为：d

输出端为：q

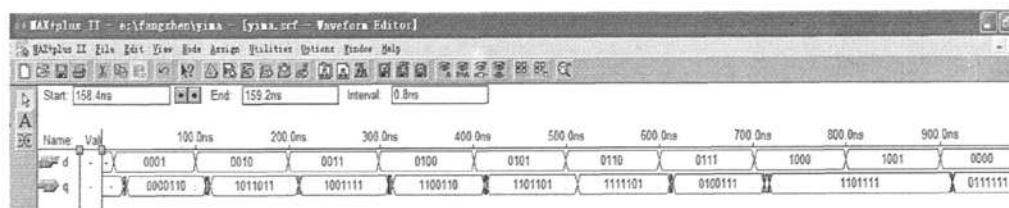


图 5.7 模块 yima 仿真波形图

由仿真结果可以看出此模块为数显译码器，对输入的 0-9 十进制数进行译码，输出共阴数码管的显示码。

5.8 模块 miaocp 的结果验证

该模块由于分频系数过大，给仿真带来了一定的困难。所以仿真时，我们采用较低的频率来进行仿真，其原理是相同的。如下图 5.8 所示，其中输入脉冲 cp_32M 占空比为 50%。在程序中都可以设置需要的任意频率值和占空比值。

但在实际运用中实现此功能是否正常，还需要通过把程序下载到 FPGA 芯片

中, 结合相关电路及外部设备进行测试。

输入端为: res, cp_32M

输出端为: set, cp_xuan

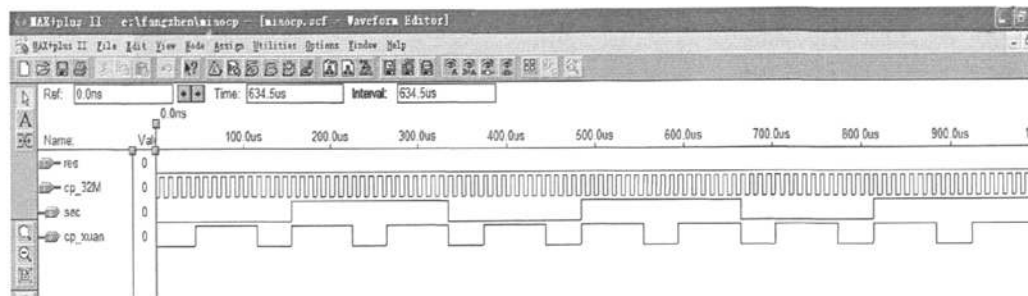
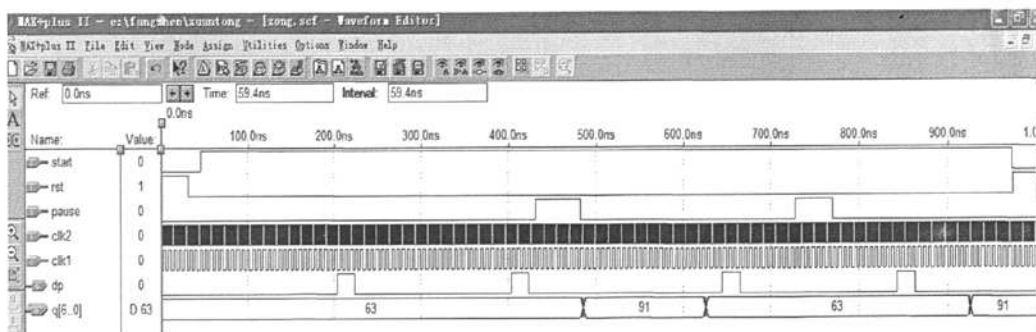


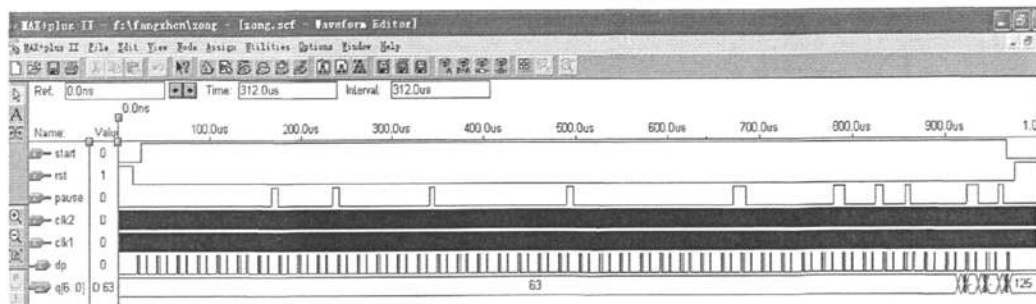
图 4.8 模块 miaocp 仿真波形图

由仿真结果可以看出此模块为分频器, 当车辆运行时, 对输入脉冲 cp_32M 进行分频输出, 其中 sec 是 32 分频输出信号, cp_xuan 为 10 分频输出信号。

5.9 总体电路的结果验证



(a)



(b)

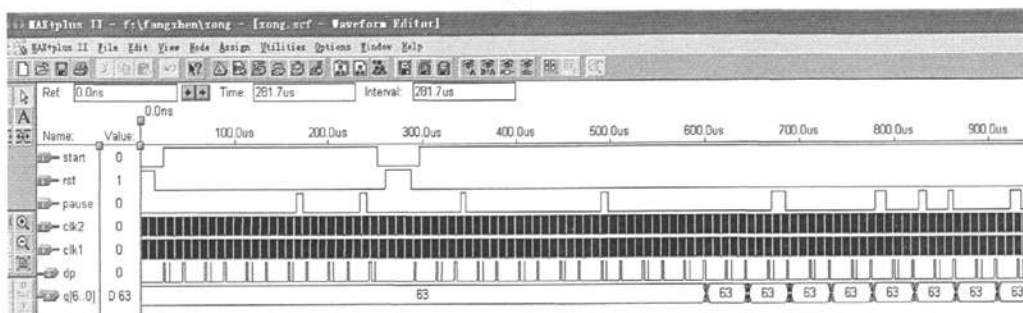
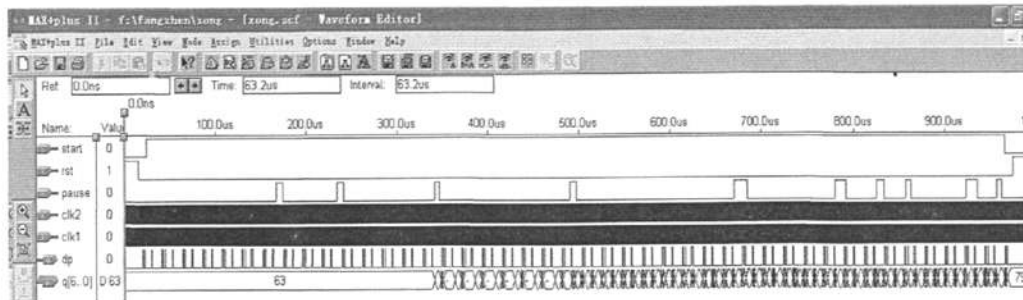


图 4.9 总体电路的仿真波形图

由仿真结果可以得出本设计功能都得以实现,当车辆开始营运(rst为“0”,stard为“1”)时,对路程脉冲(clk1)进行计数,同时把行驶的路程和暂停等候时间(由输入的32MHz脉冲生成分频秒脉冲计来时)均折算成费用,通过选通模块和译码模块把数据轮流传输到外接的显示设备(共阴数码管)上。以方便乘客直观的知晓目前的行驶里程和相应的费用。

其中的仿真图 4.9 (a)、(b)、(c) 均为单次营运仿真结果。(a) 图是行驶路程在 3 公里内的仿真结果, 先显示行驶费用, 后者是本次营运的总费用。(b) 图是行驶路程超出 3 公里的仿真。(c) 图仿真可看出在车辆行驶过程总, 行驶速度会有快慢, 相应的脉冲也会有多少、快慢。(d) 图仿真了连续两次营运的情况。

第六章 总 结

出租车计费器系统的设计已全部完成，能按预期的效果进行模拟汽车启动、停止、暂停等功能，并设计动态扫描电路显示车费数目，由动态扫描电路来完成。车暂时停止不计费，车费保持不变。若停止则车费清零，等待下一次计费的开始。

在出租车计费系统的 8 个模块中，重点在于 lucheng 模块，jijia 模块，jinzhi 模块、xuantong 模块。lucheng 模块主要有汽车的启动、停止、暂停按键；jijia 模块主要是把等候时间折算成费用；jinzhi 模块主要是把车费和路程转化为 4 位 BCD 编码数；xuantong 模块是将车费和路程显示出来。

各模块完成后，在将它们组合成完整的出租车系统，在设计过程中还需要改进的是控制系统的纠错功能。出租车计费系统的设计中体现了 VHDL 覆盖面广，描述能力强，是一个多层次的硬件描述语言及 PLD 器件速度快，使用方便，便于修改等特点，本设计在实用方面具有一定的价值。

设计由于采用 CPLD 大规模可编程逻辑器件，整机功耗小、抗干扰能力强、系统稳定、工作可靠、升级方便。另外，由于器件资源尚有较大的扩展空间，根据实际需要，系统可方便地增加所需功能，如通过器件内部编程增加时钟功能，既可为司机和乘客提供方便，又能为夜间行车自动调整收费标准提供参考；还可以通过 FPGA 的输出引线控制语音器件，对顾客发出问候语，提醒其所要收缴的费用等；也可以通过 FPGA 的输出引线控制打印机打印发票等。

通过对出租车计费器的设计经过软件验证，对芯片设计有如下几点结论：

(1)虽然 Verilog HDL 语言与 C 语言很相似，但是用其语言必须具备一定的专业知识；用后者设计思想实现的，并不一定能通过最后的逻辑综合和编译；

(2)在设计中，特别是设计计数器时，一定要主要计数器同步与异步的区别；

(3)设计时也应注意顺序语句与并发语句。比如说，设计时序电路时，一般在用一个 ALWAYS 语句实现，而其中的语句是顺序语句；但是组合逻辑电路设计时，一般用连续赋值语句 ASSIGN 语句来实现；

(4)用 MAXPLUSII 软件，可以很容易贯穿 TOP—DOWN 设计思想，图形化的设计。一方面，分块进行设计，很大程度上提高了设计的效率。

参考文献

- [1] 侯伯亨,顾新.VHDL 硬件描述语言与数字逻辑电路设计.修订版电子工程师必备知识 [M].西安: 西安电子科技大学出版社,1999
- [2] 张昌凡, 等.可编程逻辑器件及 VHDL 设计技术 [M].广州: 华南理工大学出版社,2001
- [3] 曾繁泰,陈美金.VHDL 程序设计 [M].北京: 清华大学出版社, 2001
- [4] 潘松, 等.VHDL 实用教程 [M].成都: 电子科技大学出版社,2000
- [5] 高健, 沈庆宏, 高敦堂. ITS 中车辆检测器设备的研究与实现[J].交通与计算机, 2002. 20 (5):15218
- [6] 王小平, 王彦芳, 等. 基于 EDA 技术的出租车费计价单片系统
- [7] 卢庆利. 将 VHDL 硬件描述语言引入数字电路教学的探索[J]. 实验室研究与探索, 2000, 19 (5): 67270
- [8] 刘艳. EDA 技术在数字系统电路设计实验中的应用[J]. 实验室研究与探索, 2002, 21 (3): 63264
- [9] 谭会生,张昌凡. EDA 技术及应用.西安: 电子科技大学出版社.2001
- [10] 徐颖.Protel 99 SE EDA 技术及应用.西安: 机械工业出版社.2005
- [11] 焦素敏, 等.EDA 技术及应用.北京: 清华大学出版社.2005
- [12] 康华光.电子技术基础[M].北京: 高等教育出版社, 1998.
- [13] 江冰.电子技术基础及应用[M].北京: 机械工业出版社, 2001.
- [14] 陈耀和.VHDL 语言设计技术[M]. 北京:电子工业出版社.
- [15] 毕满清.电子技术实验与课程设计[M].北京: 机械工业出版社, 2001.
- [16] 樊国梁.VHDL 语言设计数字电路[J]. 国外电子测量技术,2005 (2).
- [17] 刘丽华. 专用集成电路设计方法[M]. 北京:北京邮电大学出版社,2000.
- [18] 张明.Verilog HDL 实用教程[M].成都:电子科技大学出版社,1999
- [19] 王小军编著.VHDL 简明教程[M].乔长阁,边计年,薛宏熙译.北京:清华大学出版社,1997
- [20] 蒋敬旗,岚松松,刘明业.集成电路计算机硬件描述语言标准 Verilog 的标准化工

- 作[M].中国标准化,2000(3)
- [21] 夏宇闻.一种高层次的支持模拟和数模混合信号电路与系统设计的语言 Verilog-A/AMS[M].电子技术应用,1999(8)
- [22] 周清华, 马善农, 谢勇勤. 基于 CPLD 的出租车计价器的设计与研究 [J]. 科技广场, 2007(1): 205-206.
- [23] 杨建潮. 出租汽车计价器的计价原理 [J]. 检定与规程, 2001(4): 44-45.
- [24] 洪传文. 出租车计价器的原理及应用 [J]. 技术监督与服务, 1997(3): 17-18.
- [25] 黄再银. 基于 uPD78F0034 单片机的出租车计费器的设计与实现 [J]. 国外电子元器件, 2004(8): 21-24.
- [26] 何伟, 张玲. 消除 CPLD / FPGA 器件设计中的毛刺 [J]. 重庆大学学报, 2002, 25(12): 69.
- [27] SAMIR Palnit, ar. Verilog HDL: A guide to Digital Design and Synthesis(B, /CD-ROM). Prentice Hall PTR, 1996
- [28] J.Bhas, er. Verilog HDL Synthesis: A Practical Primer. Star Galaxy Publishing, PA, 1998.
- [26] Ur, owitz H. Energy detection of un, nown deterministic signals[J].Proceeding of the IEEE, Apr. 1967,55(04): 523 ~ 531.
- [29] Mitola J. Cognitive Radio: an Integrated Agent architecture for Software Defined Radio [J]. Dissertation, Royal Institute ofTechnology, 2000.
- [30] Cabric D, Mishra S M, Brodersen R W. Implementation Issues inSpectrum Sensing for Cognitive Radios[J]. The Thirty-eighthAsilomar Conference on Signals, Systems and Computers, Sep.2004,1:77 ~ 776.

攻读学位期间公开发表的论文

1. 凌璟, 《用 VHDL 设计示波器的方法探讨》吉林省教育学院学报 2009 年第七期

致 谢

经过近一年的查找、整理参考资料,编写程序、仿真,到最后写作论文,终于完成了毕业设计。看着厚厚的札记和研究期间的纪录,求学期间的点点滴滴历历涌上心头,时光匆匆飞逝,三年的努力与付出,随着论文的完成,工程硕士的学习生活将划上句号。看着心血凝聚而成这么一本论文,欣慰和高兴之余,感慨颇多。首先衷心感谢尊敬的导师李富华老师,因为论文是在李老师的悉心指导下完成的。李老师渊博的专业知识,严谨的治学态度,精益求精的工作作风,诲人不倦的高尚师德,严以律己、宽以待人的崇尚风范,朴实无华、平易近人的人格魅力对我影响深远。本论文从选题,开题到完成定稿,每一步都是在李老师的指导下完成的,倾注了李老师的大量心血。

感谢苏州大学多位老师的教导与帮助,是他们的辛勤付出给了我宝贵的新知识,他们渊博的知识让我受益匪浅。同时感谢我的同事兼同学们,在论文完成过程中给予我很多的帮助,你们给予我理解、支持和鼓励,这些都成为我向前迈进的动力。

三年的时光如白驹过隙,我身边所有的老师、同学、朋友和亲人,谢谢三年来对我的关照与宽容,一起走过的缤纷时代,将会是我一生最珍贵的回忆。最后,要感谢在百忙中抽出时间对本文进行评审并提出宝贵意见的各位专家。谢谢大家!