

毕 业 设 计 说 明 书

GRADUATE DESIGN

设计题目：基于单片机的智能交通灯控制系统的设计

学生姓名：

专业班级：

学 院：电气工程学院

指导教师：

摘 要

随着现代社会对交通运输的日趋依赖，交通灯成为了人们生活中不可或缺的一部分。传统的交通灯控制系统虽然在一定程度上可以满足指挥路口交通的需要，但随着城市规模的不断扩大，原有的交通灯控制系统已经表现出明显的缺点：红绿灯时间相对固定，不能伴随车流量的改变而调整红绿灯的显示时间。

本设计以 AT89S5 单片机为核心，外接外围电路构成基本电路，使硬件电路能适应所完成的控制功能。在 Keil 软件中编写 C 语言程序，最后用 Proteus 软件进行仿真，基本实现了智能交通灯的模拟。该系统可控制红、绿、黄灯按时间依次变换，并有倒读秒功能。在此基础上，通过传感器对车流量的情况进行数据采集。将采集的数据传送给控制中心，进行分析比较。根据比较的结果，将具体的车流量转换成两相位车流量大小的比值。根据比值转换成对红绿灯时间的控制，使交通信号灯时间可根据车流量改变，提高了交叉口的通行效率。

关键词：单片机；交通灯；倒计时显示；传感检测

Abstract

As modern society increasingly rely on for transportation, traffic lights has become the indispensable part of people's life. Although to some extent, the traditional traffic light control system can meet the needs of command intersection traffic, but with the enlargement of city scale, the original traffic light control systems have shown obvious disadvantages: traffic light time is relatively fixed, not along with the change of the flow adjustment of traffic lights show time.

This design uses AT89S51 microcontroller as the core, constitute the basic circuit, external peripheral circuit that can be adapted to control the functions performed by hardware circuit. In Keil software written in C language program, and finally with the Proteus software simulation, the basic realization of the intelligent simulation of traffic lights. The system can control red, green, yellow light according to time, in turn, transform, and has function of seconds. On this basis, through the sensor to the situation of traffic data collection. Will collect the data transmitted to the control center, carries on the analysis comparison. According to the result of comparison, the specific traffic into two phase flow is the ratio of the size. Translated into traffic lights time according to the ratio control, make the traffic lights time may be changed according to the number of cars, improving the efficiency of the intersection traffic.

Key words: single chip microcomputer; The traffic light; The countdown display; Sensing detection

目 录

摘 要	II
Abstract	III
第 1 章 绪论	1
1.1 课题背景	1
1.2 课题研究的意义	2
1.3 智能交通灯的研究背景	2
1.4 国内外研究现状	3
1.5 我国交通灯现状	3
1.6 论文结构	4
第 2 章 智能交通灯控制系统方案设计	5
2.1 智能交通灯控制系统的通行方案设计	5
2.2 智能交通灯控制系统的功能要求	6
2.3 智能交通灯控制系统的基本构成及原理	7
2.4 本章小结	7
第 3 章 系统硬件电路的设计	8
3.1 主要硬件的选型	8
3.1.1 单片机的选型	8
3.1.2 车流量检测传感器的选型	8
3.1.3 电源电路的选型	10
3.2 系统硬件总电路构成及原理	10
3.2.1 系统硬件电路构成	10
3.2.2 系统工作原理	11
3.3 AT89S51 单片机简介	12
3.3.1 单片机的概述	12
3.3.2 AT89S51 主要引脚功能	12
3.3.3 AT89S51 芯片最小系统	15
3.4 车流量检测模块介绍	16
3.4.1 光电开关的工作原理	17
3.4.2 光电开关的分类	17
3.4.3 光电开关的应用	18
3.5 其它硬件介绍	19
3.5.1 发光二极管	19
3.5.2 七段 LED 数码管	19
3.5.3 电源电路设计	20
3.5.4 蜂鸣器	21
3.6 本章小结	21
第 4 章 软件设计	23
4.1 主程序设计	23
4.2 车流量采样程序设计	23
4.3 显示程序设计	27
4.4 理论基础知识	27
4.4.1 定时器原理	27

4.4.2 软件延时原理	28
4.5 本章小结	28
第 5 章 智能交通灯的仿真	29
5.1 Proteus 软件介绍	29
5.2 仿真过程介绍	29
5.2.1 用 PROTEUS 绘制原理图	29
5.2.2 PROTEUS 对单片机内核的仿真	30
5.2.3 仿真结果与分析	31
5.3 本章小结	37
结束语	38
参考文献	39
谢 辞	40
附 录	41

第 1 章 绪论

1.1 课题背景

随着人口快速的增多，交通工具的爆炸性的发展，以及道路资源的有限性，交通控制就应运而生。在人类的生活、工作环境中，交通扮演着极其重要的角色，人们的出行都无时不刻与交通打着交道。当今，红绿灯安装在各个道口上，已经成为疏导交通车辆最常见和最有效的手段。但这一技术在 19 世纪就已出现了。

1858 年，在英国伦敦主要街头安装了以燃煤气为光源的红、蓝两色的机械扳手式信号灯，用以指挥马车通行。这是世界上最早的交通信号灯。1868 年，英国机械工程师纳伊特在伦敦威斯敏斯特区的议会大厦前的广场上，安装了世界上最早的煤气红绿灯。它由红绿两以旋转式方形玻璃提灯组成，红色表示“停止”，绿色表示“注意”。1869 年 1 月 2 日，煤气灯爆炸，使警察受伤，遂被取消。电气启动的红绿灯出现在美国，这种红绿灯由红绿黄三色圆形的投光器组成，1914 年始安装于纽约市 5 号大街的一座高塔上。红灯亮表示“停止”，绿灯亮表示“通行”。1918 年，又出现了带控制的红绿灯和红外线红绿灯。带控制的红绿灯，一种是把压力探测器安在地下，车辆一接近红灯便变为绿灯；另一种是用扩音器来启动红绿灯，司机遇红灯时按一下喇叭，就使红灯变为绿灯。红外线红绿灯当行人踏上对压力敏感的路面时，它就能察觉到有人要过马路。红外光束能把信号灯的红灯延长一段时间，推迟汽车放行，以免发生交通事故。

信号灯的出现，使交通得以有效管制，对于疏导交通流量、提高道路通行能力，减少交通事故有明显效果。1968 年，联合国《道路交通和道路标志信号协定》对各种信号灯的含义作了规定。绿灯是通行信号，面对绿灯的车辆可以直行，左转弯和右转弯，除非另一种标志禁止某一种转向。左右转弯车辆都必须让合法地正在路口内行驶的车辆和过人行横道的行人优先通行。红灯是禁行信号，面对红灯的车辆必须在交叉路口的停车线后停车。黄灯是警告信号，面对黄灯的车辆不能越过停车线，但车辆已十分接近停车线而不能安全停车时可以进入交叉路口。

交通对于社会的工业经济和人们的生活生产中有着十分重要的意义。交通控制系统是近现代社会随着物流、出行等交通发展产生的一套独特的公共管理系统。要保证高效安全的交通秩序，除了制定一系列的交通规则，还必须通过一定的技术手段加以实现。随着单片机和传感技术的迅速发展，自动检测领域发生了巨大变化，智能交通灯控制系统方面的研究有了明显的进展，并且必将以其优异的性能价格比，逐步取代传统的交通灯控制措施。

1.2 课题研究的意义

科学技术的进步推动了交通工具的现代化，社会经济的发展则导致了交通量的急剧增长并进而加剧了交通拥挤与阻塞的严重程度，城市交通的规模与复杂特征、传统交通控制和交通拥挤一直是困扰世界各国的一大难题，目前美国每年由于交通拥挤造成的直接经济损失达 2370 亿美元以上，而我国国内百万人以上的大城市每年由交通阻塞造成的直接间接经济损失约计 1600 亿元以上，相当于国内生产总值的 3.2%。解决城市交通问题的根本路径大致有两条：一是加快交通基础设施建设；二是加强交通管理。前者是发展城市交通，满足各种交通需求的物质基础，而后者则为合理使用现有交通设施。保证人车的安全，在良好的交通环境下，使现有设施的能力得以发挥。二者相比，由于在大城市新建和扩建道路的可能性受空间制约越来越小。当前城市交通管理的重点也侧重于加强交通管理，对平面交叉口的研究一般都是应用交通信号在时间上给车辆分配通行权，从而实现车辆在时间上的分离。

智能的交通信号灯指挥者人和车辆的安全运行，实现红、黄、绿灯的自动指挥是城乡交通管理现代化的重要课题。在城乡街道的十字交叉路口，为了保证交通秩序和行人安全，一般在每条道路上各有一组红、绿、黄交通信号灯。其中红灯亮，表示该条道路禁止通行；黄灯亮，则表示该条道路上未过停车线的车辆停止通行，已过停车线的车辆继续通行；绿灯亮，则表示该条道路上允许通行。交通灯控制电路自动控制十字路口两组红、黄、绿交通灯的状态转换，指挥各种车辆和行人安全通行，从而实现十字路口城乡交通管理自动化。

智能交通灯控制系统研究的发展，旨在解决人类交通因需求的增多而日益繁重带来的问题，局限于道路建设的暂时不足和交通工具的快速增长，就要使更多的车辆安全高效的利用有限的道路资源，避免因无序和抢行等无控制原因造成的不必要阻塞甚至瘫痪，另外，针对整个交通线路车辆的多少实时调整和转移多条线路的分流也十分必要。

交通网络是城市的动脉，象征着一个城市的工业文明水平。交通关系着人们对于财产，安全和时间相关的利益。具有优良科学的交通控制技术对资源物流和人们出行都是十分有价值的，保证交通线路的畅通安全，才能保证出行舒畅，物流准时到位，甚至是生命通道的延伸。

1.3 智能交通灯的研究背景

智能交通灯是智能交通领域的一个分支。智能交通系统，简称 ITS(Intelligent Transport Systems)，作为一个概念性名词出现于 20 世纪 90 年代初，但其思想早在

20世纪30年代已有萌芽，当时美国通用汽车公司和福特汽车公司倡导和推广过‘现代化公路网’的构想，而20世纪60年代出现的静态路径诱导、计算机交通控制技术等都可谓是 ITS的雏形，不过当时其重要性并不明显，没有受到人们足够的重视。因此，近几年来世界各国都竞相投资 ITS的研发和应用。智能交通灯的应用是解决智能交通系统的关键之一。

1.4 国内外研究现状

目前国内外较为完善的交通信号控制系统主要有英国的 TRANSYT (Traffic Network Study Tool) 和 SCOOT (Split, Cycle and Offset Optimization Technique, 绿信比、周期和相位差优化技术) 系统和澳大利亚的 SCATS (Sydney Coordinated Adaptive Traffic System), 悉尼协调自适应交通系统, 以及美国、日本等国家开发的一些系统, 其中以英国的 SCOOT 系统和澳大利亚的 SCATS 系统相较为著名。它们在中国的城市 (如: 上海、杭州、宁波等用的是 SCATS 系统; 成都、大连、北京等用的是 SCOOT 系统) 也得到了较好的应用。但由于这些系统多为交通信号控制专用系统, 因此开放性较差, 难于同其它系统连接和协调控制, 系统带有一定的局限性, 并且价格比较昂贵, 没有充分考虑我国现有的国情 (如自行车交通流和行人的交通流等)。我国近几年经过深入研究, 也开发出了一些适用于我国交通状况的交通控制系统, 主要有上海交通大学的 SUATS 系统和南京、深圳等地研制的系统。这些系统在深入研究国外先进系统的基础上, 融合了大量交警实际控制经验, 以开放性为前提, 增加了符合国情的特殊功能。但还不成熟, 控制效果也不是非常好, 没有得到广泛应用。因此, 结合我国国民经济, 建立一个相对廉价、获取信息多且准确、工作可靠、具有智能交通控制系统势在必行。

1.5 我国交通灯现状

随着城市机动车增长速度的加快, 1994年我国城市机动车保有量已接近 500 万辆。20世纪90年代以来, 经济的发展加快, 从 1985年到 1995年, 机动车增长率达 13%左右, 近几年更是增多。

然而, 在此同时, 城市道路建设规模也在加大。我国城市普遍存在道路密度、道路面积率偏低的问题。我国城市道路的密度只有 6.8km每平方千米, 而在 20世纪80年代, 世界发达国家就已到达 20km每平方千米。20世纪90年代, 我国部分城市道路面积率, 北京为 5.9%, 上海为 6.4%, 而国外东京为 13.8%, 巴黎为 25%, 普遍高于我国。近几年, 国家虽不断加大城市道路建设的力度, 但仍赶不上车辆的增长速度, 且与世界其他国家相比, 差距仍很大。

目前我国城市街道交叉路口的交通信号灯虽然是自动的，但是仔细观察就会发现红绿灯的交替转换是定时式的，即转换的间隔时间是固定不变的。定时式并不符合实际要求。因为，如果东西和南北两方向车流量相差很大，而信号灯还是平均分配导通时间，就会出现这样的问题：车多的方向导通时间不足，而车少的方向导通时间剩余，造成一方向车挤另一方向车松的不合理的局面，这就是机器自动控制不如人工现场指挥的差别。然而人工指挥劳动强度大，我们应充分发挥计算机的作用，用计算机模拟人的智能来控制交通灯，从而提高经济和社会效益。

1.6 论文结构

基于整个交通控制系统的发展情况，本设计主要进行如下方面的研究：用智能，集成，且功能强大的单片机芯片为控制中心，设计出一套十字路口的交通控制系统，以指挥路口的实时通行状态。

在绪论部分讲述了本课题的研究背景与意义、国内外智能交通控制系统的研究现状以及我国交通灯的现状。

在第二章中，基于绪论部分对单片机智能交通灯控制系统的部分了解以及现实生活中的需要，根据设计要求提出总体设计方案论证与选择，介绍了智能交通灯控制系统的基本构成及原理。

在一、二章的基础上，第三章完成了硬件的选型以及硬件电路的设计。

第四章首先根据软件设计流程图简要介绍了软件设计，并介绍了各个程序模块的基本设计思想。

第五章简要介绍了 proteus 软件及电路绘制并且详细叙述了如何实现电路的仿真。

最后是对本课题的总结与展望，概述了系统实现的功能，前景及致谢、附录、参考文献等关于本次毕业设计的后续工作。附录为系统的程序清单以及整体电路图供阅读参考。

第 2 章 智能交通灯控制系统方案设计

2.1 智能交通灯控制系统的通行方案设计

设在十字路口，分为东西向和南北向，在任一时刻只有一个方向通行，另一方向禁行，持续一定时间，经过短暂的过渡时间，将通行禁行方向对换。其具体状态如下图所示。说明：黑色表示亮，白色表示灭。交通状态从状态 1 开始变换，直至状态 4 然后循环至状态 1，周而复始，即如图 2.1 所示：

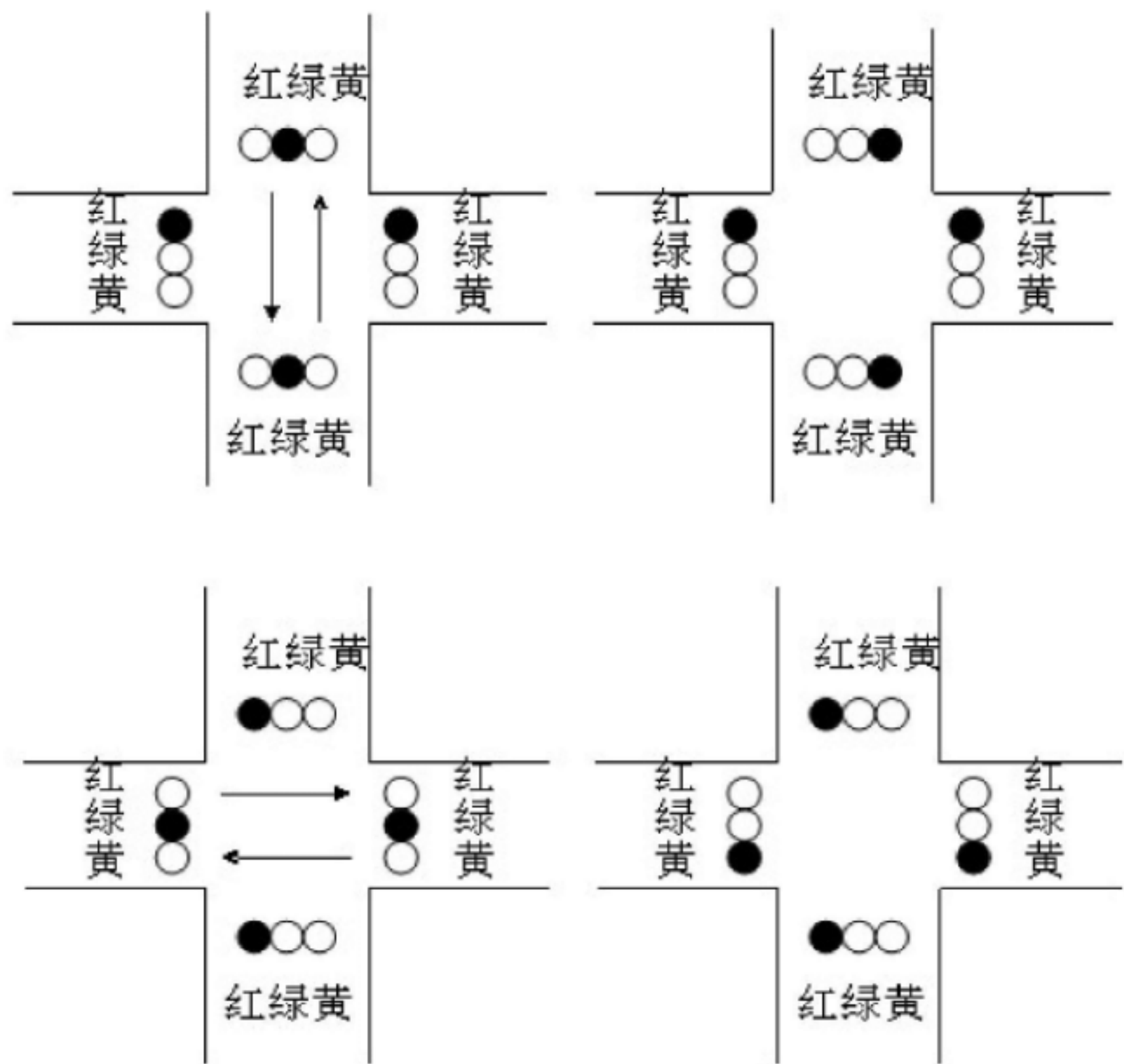


图 2.1 交通状态

通过具体的路口交通灯状态的演示分析我们可以把这四个状态归纳如下：

- （1）南北绿灯亮，东西红灯亮。此状态下，南北允许通行，东西禁止通行。
- （2）南北黄灯亮，东西保持红灯亮。此状态下除了已经正在通行中的其他所以车辆都需等待状态转换。
- （3）东西绿灯亮，南北红灯亮。此状态下，东西允许通行，南北禁止通行。

（4）东西黄灯亮，南北保持红灯亮。此状态下除了已经正在通行中的其他所以车辆都需等待状态转换。

下面用图表表示灯状态和行止状态的关系如下：

表 2-1 交通状态及红绿灯状态

	状态 1	状态 2	状态 3	状态 4
南北向	通行	等待变换	禁行	等待变换
东西向	禁行	等待变换	通行	等待变换
南北红灯	0	0	1	1
南北绿灯	1	0	0	0
南北黄灯	0	1	0	0
东西红灯	1	1	0	0
东西绿灯	0	0	1	0
东西黄灯	0	0	0	1

东西南北四个路口均有红绿黄 3灯和数码显示管 2个，在任一个路口，遇红灯禁止通行，转绿灯允许通行，之后黄灯亮警告行止状态将变换。状态及红绿灯状态如表 2.1所示。说明：0表示灭，1表示亮。

2.2 智能交通灯控制系统的功能要求

本设计能模拟基本的交通控制系统，用红绿黄灯表示禁行，通行和等待的信号发生，还能进行倒计时显示，车流量检测及调整，错误报警等功能。

（1）倒计时显示

倒计时显示可以提醒驾驶员在信号灯灯色发生改变的时间、在“停止”和“通过”两者间作出合适的选择。驾驶员和行人普遍都愿意选择有倒计时显示的信号控制方式，并且认为有倒计时显示的路口更安全。倒计时显示是用来减少驾驶员在信号灯色改变的关键时刻做出复杂判断的1种方法，它可以提醒驾驶员灯色发生改变的时间，帮助驾驶员在“停止”和“通过”两者间作出合适的选择

（2）红绿灯显示

红绿灯显示可以直观的告诉驾驶员禁行，通行和等待的信号。本设计红绿灯有四种状态：首先南北绿灯亮，东西红灯亮。一定时间后，南北黄灯开始闪烁，持续5s，东西向保持红灯亮。接着南北向红灯亮，东西绿灯亮。一定时间后，东西黄灯闪烁，持续5s，南北向保持红灯亮。

（3）车流量检测及调整

随着我国经济建设的蓬勃发展，城市人口和机动车拥有量在急剧增长，交通

流量日益加大，交通拥挤堵塞现象日趋严重，交通事故时有发生。车辆检测器作为智能交通系统的基本组成部分，在智能交通系统中占有重要的地位。现阶段，车辆检测器检测方式有很多，各有其优缺点，如遥感微波检测器、磁感应车辆检测器、红外线车辆检测器等。通过比较南北向和东西向的车流量，调节红绿灯的间隔时间。

2.3 智能交通灯控制系统的基本构成及原理

单片机设计交通灯控制系统，可用单片机直接控制信号灯的状态变化，基本上可以指挥交通的具体通行，当然，接入LED数码管就可以显示倒计时以提醒行使者，更具人性化。本系统在此基础上，加入了车流量检测电路为单片机采集数据，单片机对此进行具体处理，及时调整控制指挥，为了超越视觉指挥的局限性，同时接上蜂鸣器，在听觉上加强了指挥提醒作用。

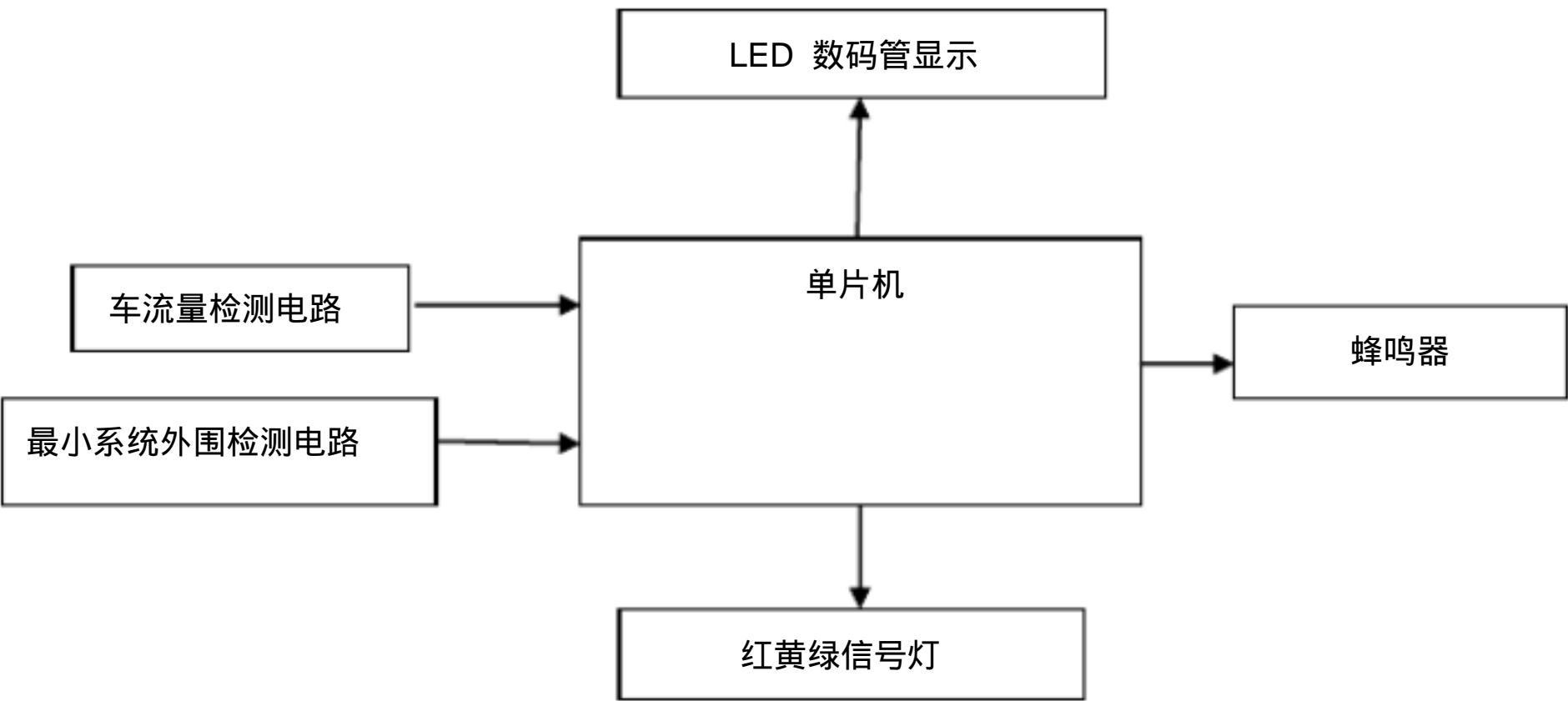


图 2.2 系统的总体框图

本设计系统以单片机为控制核心，由车流量检测模块产生输入，信号灯状态模块，LED 倒计时模块和蜂鸣器状态模块接受输出。系统的总体框图如图 2.2 所示。

2.4 本章小结

本章主要对智能交通灯控制系统方案设计进行了介绍，概述了智能交通灯控制系统的功能要求以及系统的总体框架。

第 3 章 系统硬件电路的设计

3.1 主要硬件的选型

实现本设计要求的具體功能，可以用单片机及外围器件构成最小控制系统，12个发光二极管分成 4组红绿黄三色灯构成信号灯指示模块，8个LED东西南北各两个构成倒计时显示模块，车流量检测传感器采集流量数据，1个蜂鸣器进行报警。

3.1.1 单片机的选型

采用AT89S51 单片机作为主控制器。AT89S51具有两个 16位定时器 / 计数器，5个中断源，便于对车流量进行定时中断检测。32根 I/O 线，使其具有足够的 I/O 口驱动数码管及交通灯。外部存储器寻址范围 ROM、RAM64K，便于系统扩展。其 T0，T1 口可以对外部脉冲进行实时计数操作，故可以方便实现车流量检测信号的输入。选用 AT89S51 单片机跟其他单片机相比，经济实惠，满足设计要求，故选用 AT89S51 单片机作为主控制器。

3.1.2 车流量检测传感器的选型

车流量检测传感器有三种方案如下：

方案一：

采用遥感微波检测器（RTMS）。微波交通检测器是利用雷达线性调频技术原理，通过发射中心频率为 10.525GHz或24.200GHz的连续频率调制微波（FMCW）；在检测路面上，投射一个宽度为 3-4米，长度为 64米的微波带。每当车辆通过这个微波投射区时，都会向 RTMS反射一个微波信号，RTMS接收反射的微波信号，并计算接收频率和时间的变化参数以得出车辆的速度及长度，提供车流量、道路占有率、速度和车型等实时信息。为了检测出车道上车的数量，RTMS在微波束的发射方向上以 2M为一个层面分展探测物体，微波束在 15度范围内投影形成一个分为 32个十层面的椭圆形波束，（椭圆的宽度取决于仪器选择的工作方式），通过这种方式可检测出车量数 RTMS具有两种基本的使用模式，分别是路边侧向模式和前方正向模式。路边侧向模式可以使用一台 RTMS同时检测多至 8条车道，并提供每条车道的交通信息。前方正向模式，用一台 RTMS实时检测一条单一车道的交通情况。RTMS的检测精度高，且是一个全天候的车辆检测器。

方案二：

采用磁感应车辆检测器。这种环形线圈检测器是传统的交通检测器，是目前世界上用量最大的一种检测设备。这些埋设在道路表面下的线圈可以检测到车辆通过时的电磁变化进而精确地算出交通流量。交通流量是交通统计和交通规划的基本数据，通过这些检测结果可以用来计算占用率（表征交通密度），在使用双线圈模式时还可以提供速度、车辆行驶方向、车型分类等数据，这些数据对于交通管理和统计是极为重要的。原理方框图如下：

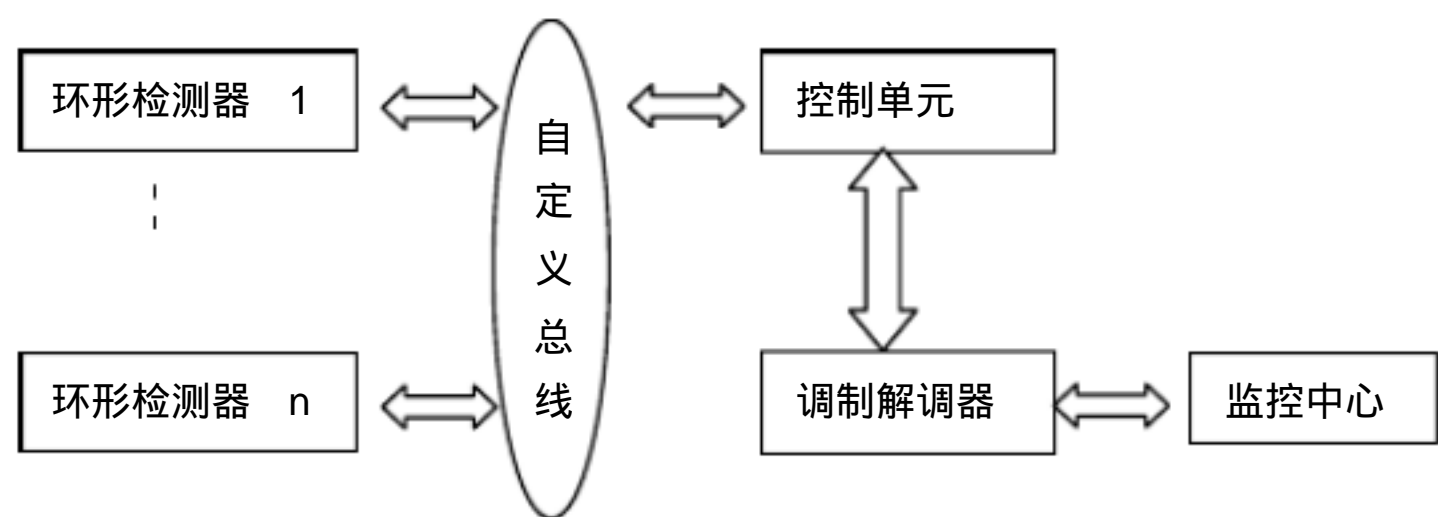


图 3.1 磁检测器方框图

该方案测量精度较好，且性能稳定。

方案三：

利用红外线车辆检测器。红外线车辆检测器是利用被检测物对光束的遮挡或反射，通过同步回路检测物体有无。物体不限于金属，所有能反射光线的物体均可被检测。光电开关将输入电流在发射器上转换为光信号射出，接收器再根据接收到的光线的强弱或有无对目标物体进行探测。如当汽车通过光扫描区域时，部分或全部光束被遮挡，从而实现对车辆数据的综合检测。常利用光电开关技术，具有高速响应，抗干扰性强，不受恶劣气象条件或物体颜色的影响的优点，而且安装简便。

方案一造价高，且易受环境影响，方案二需将检测器埋入地底下，对已建成道路使用不方便。方案三性价比高，且设计简单，权衡利弊，故选用方案三。

在本系统中，采用对射式红外线光电开关 HJS18-M14DNK 检测车流量。该红外线光电开关工作电压为直流 10-30V，检测距离为 10m，响应时间小于 3ms，能在 -25 ~ 55 的温度条件下正常工作。当有车辆通过光电开关之间时，输出端将输出一个开关信号，送入单片机，单片机执行相应程序自动对输入信号进行计数，从而完成对车流量的统计。

3.1.3 电源电路的选型

由于单片机工作时需要 +5V 电压,所以在设计电源电路时 ,需要一个三端稳压器能提供 +5V 电压。

三端稳压器,主要有两种:一种输出电压是固定的,称为固定输出三端稳压器;另一种输出电压是可调的,称为可调输出三端稳压器。其基本原理相同,均采用串联型稳压电路。在线性集成稳压器中,由于三端稳压器只有三个引出端子,具有外接元件少,使用方便,性能稳定,价格低廉等优点,因而得到广泛应用。

三端稳压器的通用产品有 78 系列(正电源)和 79 系列(负电源),输出电压由具体型号中的后面两个数字代表,有 5V, 6V, 8V, 9V, 12V, 15V, 18V, 24V 等档次。

由于7805能够提供 5V 电压的三端稳压电源,在实际的电路控制中应用其作为电源电路较为广泛,在普通的电子元器件商场都有销售易于购买,并且技术相对成熟。7805一脚为电源输入端,二脚为公共接地端,三脚即为我们所需要的 +5V 电压输出端。本文采用最典型的 7805提供电压的电路,即在 7805的1脚和公共接地端(即2脚)之间接入 $0.3\ \mu\text{F}$ 的电容,在公共接地端和三脚 +5V 电压输出端之间接入 $0.1\ \mu\text{F}$ 的电容。

3.2 系统硬件总电路构成及原理

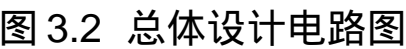
3.2.1 系统硬件电路构成

本系统实现的是对城市十字路口交通的控制,它由三大部分组成:

- (1) 信息的采集部分;
- (2) 单片机自动控制部分;
- (3) 显示部分。

系统以单片机为核心,组成一个集车流量采集、处理、自动控制为一身的开环控制系统。系统硬件电路由车流量检测电路、单片机、状态灯、LED 显示、蜂鸣器组成。其具体的硬件电路总图如图所示。

P0接上拉电阻与 P2控制LED 数码管, P1用于控制红绿黄发光二极管, INT1 口接蜂鸣器, XTAL1 和 XTAL2 接入晶振时钟电路, RESET引脚接上复位电路, T1口接车流量传感器。



通过车流量传感器对东西和南北两条通道的车流量的测量，将车流量信息以脉冲电平的方式传给单片机。单片机能通程序运算得到两条通道车流量的大小来控制路口各方向的红绿灯时长，并由LED显示。以一个周期向传感器取一次数据。

信息采集主要是对路口各方向的车流量进行采集。因为关系到哪个方向通行时间长，哪个方向通行时间短。目前大多采用光学或压力传感器，以确定每个路口在一定时间段车辆通过的次数。本设计采用比例的方法利用红外线传感器只计算出需比较两个相对路口车流量的比值即可。

设计车流量传感器，一个对准东西方向取样，另一个对南北方向取样，分别取得两个代表东西和南北方向车流量 a 和 b 。用单片机巡回检测，并将他们进行比较。若二者相等则按一定时间间隔交替导通。若 $a > b$ ，进入 a 方向绿灯延时程序。反之，进入 b 方向绿灯延时程序。该控制程序又根据具体的比例做时长的变换。这些工作全部由单片机完成。单片机通过接口得到 a 和 b 相对应的电压信号量，然后对其进行处理、分析和判断，改变信号灯输出时长，直接控制信号灯驱动电路，实现单片机对信号灯的智能控制。

本系统先显示状态灯及 LED 数码管，将状态码值送显 P1口，将要显示的时间

值送显 P0口和用 P2口来选通 LED 数码管的显示导通，在此同时以 50ms为周期，用软件方法计时 1秒，到达 1s就要将时间值减 1，刷新 LED 数码管。

该智能交通灯控制系统以四个状态为一周期循环。每满一个状态循环周期，则须将检测到的车流量数据处理一次，判断两个方向的交通轻重缓急状况，再调整下次状态循环的红绿灯时间，以达到自动控制的目的。

3.3 AT89S51 单片机简介

3.3.1 单片机的概述

单片微型计算机简称单片机，又称微控制器，嵌入式微控制器等，属于第四代电子计算机。它把中央处理器、存储器、输入/输出接口电路以及定时器/计数器集成在一块芯片上，从而具有体积小、功耗低、价格低廉、抗干扰能力强且可靠性高等特点，因此，适合应用于工业过程控制、智能仪器仪表和测控系统的前端装置。正是由于这一原因，国际上逐渐采用微控制器（MCU）代替单片微型计算机（SCM）这一名称。“微控制器”更能反映单片机的本质，但是由于单片机这个名称已经为国内大多数人所接受，所以仍沿用“单片机”这一名称。

单片机的主要特点有：

- （1）具有优异的性能价格比。
- （2）集成度高、体积小、可靠性高。
- （3）控制功能强。
- （4）低电压，低功耗。

AT89S51是美国 ATMEL 公司生产的低功耗，高性能 CMOS8位单片机，片内含 4k bytes的可系统编程的 Flash只读程序存储器，器件采用 ATMEL 公司的高密度、非易失性存储技术生产，兼容标准 8051指令系统及引脚。它集 Flash程序存储器既可在线编程（ISP）也可用传统方法进行编程及通用 8位微处理器于单片芯中，ATMEL 公司的功能强大，低价位 AT89S51单片机可为您提供许多高性价比的应用场合，可灵活应用于各种控制领域。

3.3.2 AT89S51主要引脚功能

AT89S51引脚图如图 3.3所示：

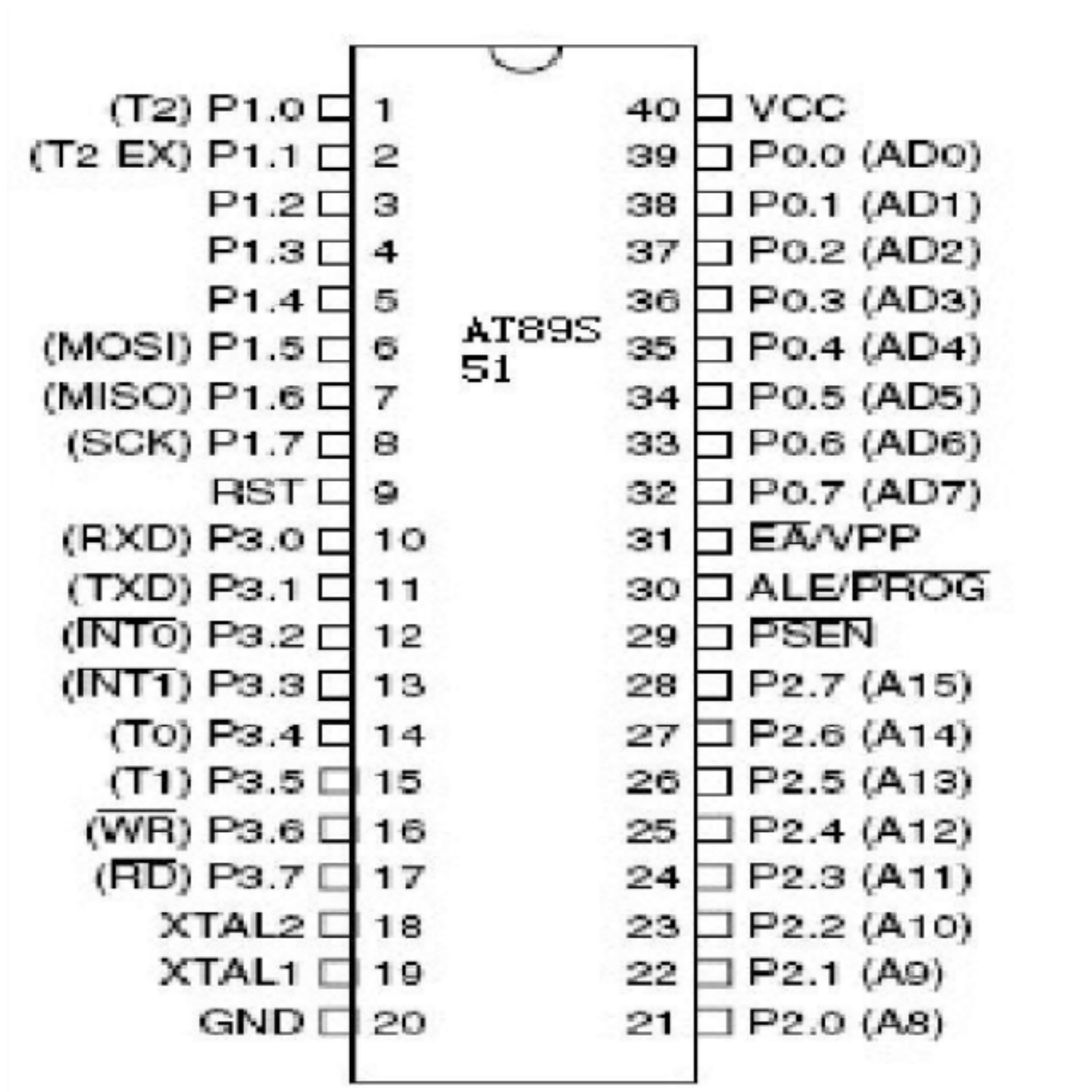


图 3.3 引脚图

VCC：电源电压

GND：地

P0口：P0口是一组 8 位漏极开路型双向 I / O 口，也即地址 / 数据总线复用口。作为输出口用时，每位能驱动 8 个 TTL 逻辑门电路，对端口写 “ 1 ” 可作为高阻抗输入端用。在访问外部数据存储器或程序存储器时，这组口线分时转换地址（低 8 位）和数据总线复用，在访问期间激活内部上拉电阻。

P1口：P1口是一个带内部上拉电阻的 8 位双向 I / O 口，P1 的输出缓冲级可驱动（吸收或输出电流） 4 个 TTL 逻辑门电路。对端口写 “ 1 ”，通过内部的上拉电阻把端口拉到高电平，此时可作输入口。作输入口使用时，因为内部存在上拉电阻，某个引脚被外部信号拉低时会输出一个电流（ IIL ）。Flash 编程和程序校验期间，P1 接收低 8 位地址。P1 口除了作为一般的 I / O 口线外，更重要的用途是它的第二功能，如表 3-1 所示：

表 3-1 具有第二功能的 P1口引脚

端口引脚	第二功能：
P1.5	MOSI（用于 ISP编程）
P1.6	MOSI（用于 ISP编程）
P1.7	MOSI（用于 ISP编程）

P2口：P2是一个带有内部上拉电阻的 8 位双向 I / O 口，P2的输出缓冲级可驱动（吸收或输出电流） 4 个TTL 逻辑门电路。对端口写 “ 1 ”，通过内部的上拉电阻把端口拉到高电平， 此时可作输入口，作输入口使用时，因为内部存在上拉电阻，某个引脚被外部信号拉低时会输出一个电流（ IIL ）。在访问外部程序存储器或 16位地址的外部数据存储器（例如执行 MOVX@DPTR 指令）时， P2口送出高 8位地址数据。在访问 8 位地址的外部数据存储器（如执行 MOVX@Ri 指令）时，P2口线上的内容（也即特殊功能寄存器（SFR）区中P2寄存器的内容），在整个访问期间不改变。 Flash编程或校验时， P2口亦接收高位地址和其它控制信号。

P3口：P3口是一组带有内部上拉电阻的 8 位双向 I / O 口。P3口输出缓冲级可驱动（吸收或输出电流） 4 个TTL逻辑门电路。对 P3口写入 “ 1 ” 时，它们被内部上拉电阻拉高并可作为输入端口。 作输入端时，被外部拉低的 P3口将用上拉电阻输出电流（ IIL ）。P3口除了作为一般的 I / O口线外，更重要的用途是它的第二功能，如表 3-2所示：

表 3-2 具有第二功能的 P3口引脚

端口引脚	第二功能：
P3.0	RXD（串行输入口）
P3.1	TXD（串行输出口）
P3.2	/INT0（外中断 0
P3.3	/ INT1（外中断 1）
P3.4	T0（定时 / 计数器 0外部输入）
P3.5	T1（定时 / 计数器 1外部输入）
P3.6	/ WR（外部数据存储器写选通）
P3.7	/ RD（外部数据存储器读选通）

P3口还接收一些用于 Flash闪速存储器编程和程序校验的控制信号。

RST：复位输入。当振荡器工作时， RST引脚出现两个机器周期以上高电平将使单片机复位。 WDT溢出将使该引脚输出高电平，设置 SFR AUXR 的DISRT0 位（地址 8EH）可打开或关闭该功能。 DISRT0位缺省为 RESET输出高电平打开状态。

ALE/ 错误！未找到引用源。：当访问外部程序存储器或数据存储器时， ALE（地址锁存允许） 输出脉冲用于锁存地址的低 8位字节。即使不访问外部存储器，ALE 仍以时钟振荡频率的 1 / 6输出固定的正脉冲信号， 因此它可对外输出时钟或

用于定时目的。要注意的是：每当访问外部数据存储器时将跳过一个 ALE 脉冲。对 Flash 存储器编程期间，该引脚还用于输入编程脉冲（PROG）。如有必要，可通过对特殊功能寄存器（SFR）区中的 8EH 单元的 D0 位置位，可禁止 ALE 操作。该位置位后，只有一条 MOVX 和 MOVC 指令 ALE 才会被激活。此外，该引脚会被微弱拉高，单片机执行外部程序时，应设置 ALE 无效。

错误！未找到引用源。：片外程序存储器读选通信号输出端，或称片外取指信号输出端。在向片外程序存储器读取指令或常数期间，每个机器周期该信号二次有效（低电平），以通过数据总线 P0 口读回指令或常数。

错误！未找到引用源。/错误！未找到引用源。：外部访问允许。欲使 CPU 仅访问外部程序存储器（地址为 0000H - FFFFH），EA 端必须保持低电平（接地）。需注意的是：如果加密位 LB1 被编程，复位时内部会锁存 EA 端状态。如 EA 端为高电平（接 VCC 端），CPU 则执行内部程序存储器中的指令。Flash 存储器编程时，该引脚加上 +12V 的编程电压 错误！未找到引用源。。

XTAL1：片内反相放大器输入端。
XTAL2：片内反相放大器输出端。外接晶体时，XTAL1 与 XTAL2 各接晶体的一端，借外接晶体与片内反相放大器构成振荡器。

3.3.3 AT89S51 芯片最小系统

一个最简单的单片机系统包括晶振、复位、电源、系统的输入控制、输出显示，以及其他外围模块（如通信、数据采集等）。

(1) 时钟电路

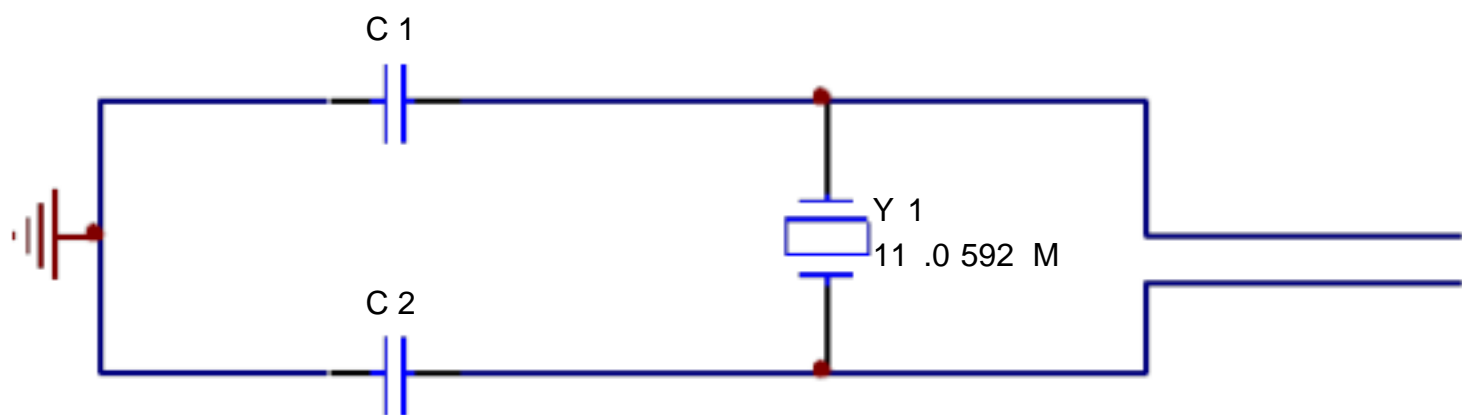


图 3.4 晶振电路图

单片机的晶振电路，即时钟电路。单片机的工作流程，就是在系统时钟的作用下，一条一条地执行存储器中的程序。单片机的时钟电路由外接的一只晶振和两只起振电容，以及单片机内部的时钟电路组成，晶振的频率越高，单片机处理数据的速度越快，系统功耗也会相应增加，稳定性也会下降。单片机系统常用的晶振频率有 6MHz、11.0592MHz、12MHz、本系统采用 11.0592MHz 晶振，电容选 22pF 或 30pF 均可。

(2) 复位电路

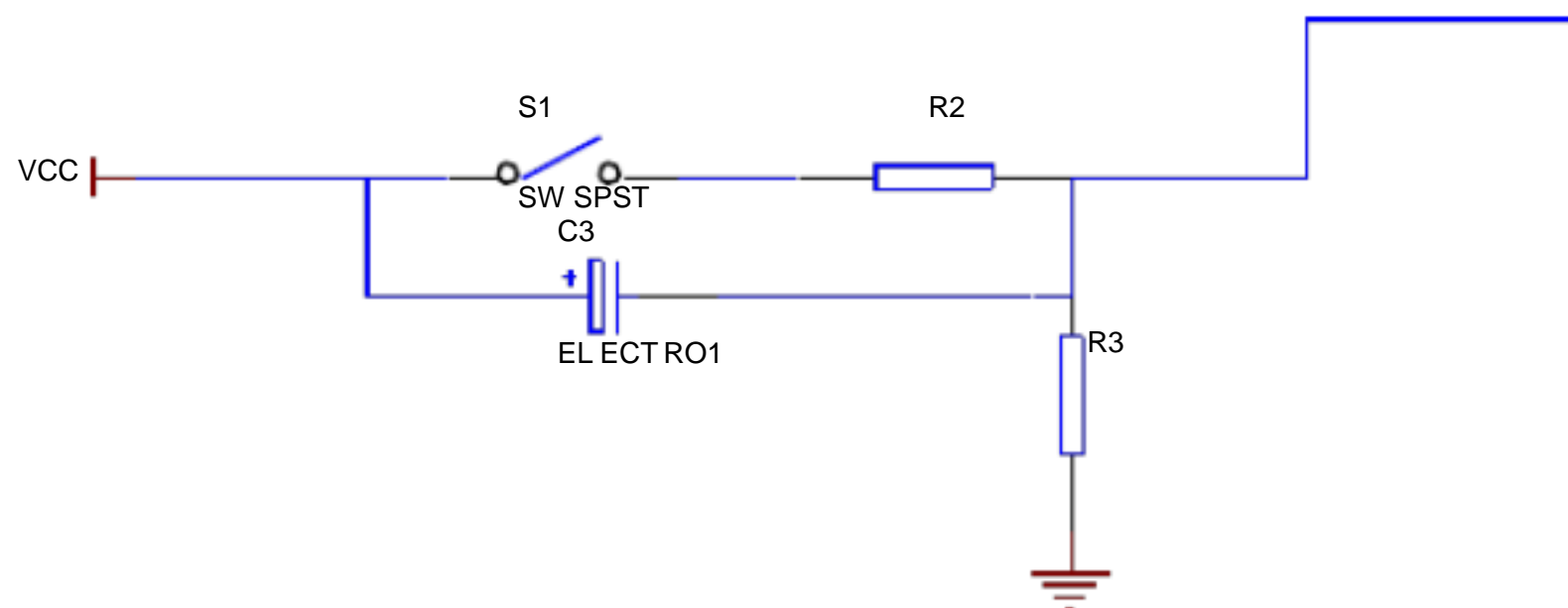


图3.5 复位电路图

系统刚上电时，单片机内部的程序还没有开始执行，需要一段准备时间，也就是复位时间。一个稳定的单片机系统必须设计复位电路。当程序跑飞或死机时，也需要进行系统复位。复位电路有很多种，有上电复位，手动复位等。复位电路的原理是单片机 RST 引脚接收到 2 个机器周期以上的高电平信号，只要保证电容的充放电时间大于 2 个机器周期，即可实现复位。在单片机启动 0.1S 后，电容 C 两端的电压持续充电为 5V，这时候 10K 电阻两端的电压接近于 0V，RST 处于低电平所以系统正常工作。当按键按下的时候，开关导通，这个时候电容两端形成了一个回路，电容被短路，所以在按键按下的这个过程中，电容开始释放之前充的电量。随着时间的推移，电容的电压在 0.1S 内，从 5V 释放到变为了 1.5V，甚至更小。根据串联电路电压为各处之和，这个时候 10K 电阻两端的电压为 3.5V，甚至更大，所以 RST 引脚又接收到高电平。单片机系统自动复位。

(3) EA 脚的功能及接法

单片机的 EA 脚控制程序从内部存储器还是从外部存储器读取程序。由于现在单片机内部的 flash 容量都很大，因此基本都是从内部的存储器读取程序，即不需要外接 ROM 来存储程序，因此，EA 脚必须接高电平。

3.4 车流量检测模块介绍

在本系统中，采用对射式红外线光电开关 HJS18-M14DNK 检测车流量。该传感器工作电压为直流 10-30V，检测距离为 10m，响应时间小于 3ms，能在 -25 ~ 55 的温度条件下正常工作。当有车辆通过光电开关之间时，输出端将输出一个开关信号，送入单片机，单片机执行相应程序自动对输入信号进行计数，从而完成对车流量的统计。

车流量检测传感器可对单片机控制系统提供实时数据，系统对所获数据进行处理。实现红绿灯控制必须解决对当前十字路口的交通状况的检测，

并完成如下工作：

- 1.输入量的采集，系统采集两个输入量，即两个方向的车流量。
- 2.输出量的确认，即红绿灯时间值。

为了采集上述数据，在十字路口的四侧共设置 2个传感器。分别检测两个方向的车流量，车流量检测不是最终目的，在每半个循环周期，系统会检测到两个方向的车流量数据，除以时间， 那么就可以得到单位时间的车流量， 然后比较两个方向单位时间车流量多少，以确定下一次循环红绿灯时间，达到调整的目的。

3.4.1 光电开关的工作原理

光电开关（光电传感器）是光电接近开关的简称，它是利用被检测物对光束的遮挡或反射，由同步回路选通电路，从而检测物体有无的。物体不限于金属，所有能反射光线的物体均可被检测。 光电开关将输入电流在发射器上转换为光信号射出，接收器再根据接收到的光线的强弱或有无对目标物体进行探测。 其工作原理图如图 3.6所示：

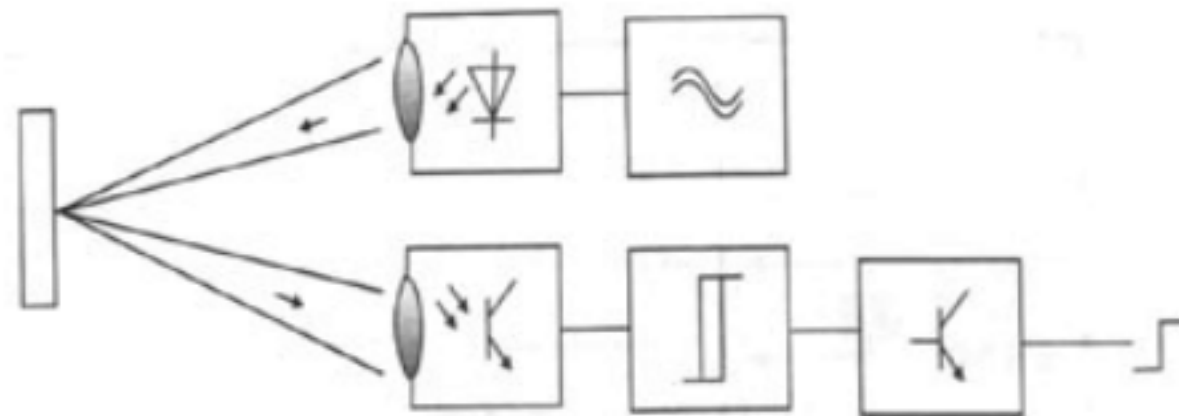


图3.6 光电开关工作原理简图

发送器对准目标发射光束，发射的光束一般来于半导体光源，发光二极管(LED)、激光二极管及红外发射二极管。 光束不间断地发射， 或者改变脉冲宽度。接收器有光电二极管、光电三极管、光电池组成。在接收器的前面，装有光学元件如透镜和光圈等。在其后面是检测电路，它能滤出有效信号和应用该信号。

3.4.2 光电开关的分类

（1）漫反射式光电开关

它是一种集发射器和接收器于一体的传感器，当有被检测物体经过时，物体将光电开关发射器发射的足够量的光线反射到接收器，于是光电开关就产生

了开关信号。当被检测物体的表面光亮或其反光率极高时，漫反射式的光电开关是首选的检测模式。

（2）镜反射式光电开关

它亦集发射器与接收器于一体，光电开关发射器发出的光线经过反射镜反射回接收器，当被检测物体经过且完全阻断光线时，光电开关就产生了检测开关信号。

（3）对射式光电开关

它包含了在结构上相互分离且光轴相对放置的发射器和接收器，发射器发出的光线直接进入接收器，当被检测物体经过发射器和接收器之间且阻断光线时，光电开关就产生了开关信号。当检测物体为不透明时，对射式光电开关是最可靠的检测装置。

（4）槽式光电开关

它通常采用标准的 U 字型结构，其发射器和接收器分别位于 U 型槽的两边，并形成一光轴，当被检测物体经过 U 型槽且阻断光轴时，光电开关就产生了开关量信号。槽式光电开关比较适合检测高速运动的物体，并且它能分辨透明与半透明物体，使用安全可靠。

（5）光纤式光电开关

它采用塑料或玻璃光纤传感器来引导光线，可以对距离远的被检测物体进行检测。通常光纤传感器分为对射式和漫反射式。但光纤式光电开关有一些缺点：适宜空间狭小、电磁干扰大、温差大。

3.4.3 光电开关的应用

随着我国工业自动化技术的迅速发展，光电开关自动化元件将被普遍采用。应用领域也在不断扩展，采用集成电路技术和 SMD 表面安装工艺而制造的新一代光电开关器件，具有延时、展宽、外同步、抗相互干扰、可靠性高、工作区域稳定和自诊断等智能化功能。这种新颖的光电开关是一种采用脉冲调制的主动式光电探测系统型电子开关，它所使用的冷光源有红外光、红色光、绿色光和蓝色光等，可非接触、无损伤地检测和控制各种固体。新型光电开关具有体积小、功能多、寿命长、精度高、响应速度快、检测距离远以及抗光、电、磁干扰能力强等优点。

目前，这种新型的光电开关已被用作物位检测、液位控制、产品计数、宽度判别、速度检测、定长剪切、孔洞识别、信号延时、自动门传感、色标检出、冲床和剪切机以及安全防护等诸多领域。

3.5 其它硬件介绍

3.5.1 发光二极管

根据本设计的特点，红绿灯的显示不可少，红绿灯的显示采用普通的发光二极管。每个方向上设置红绿黄灯，总共 4 组。如果东西红灯亮，那南北方向就是绿灯亮，反之亦然，所以在硬件上连接图上也是对称分布的，如图 3.7 所示。

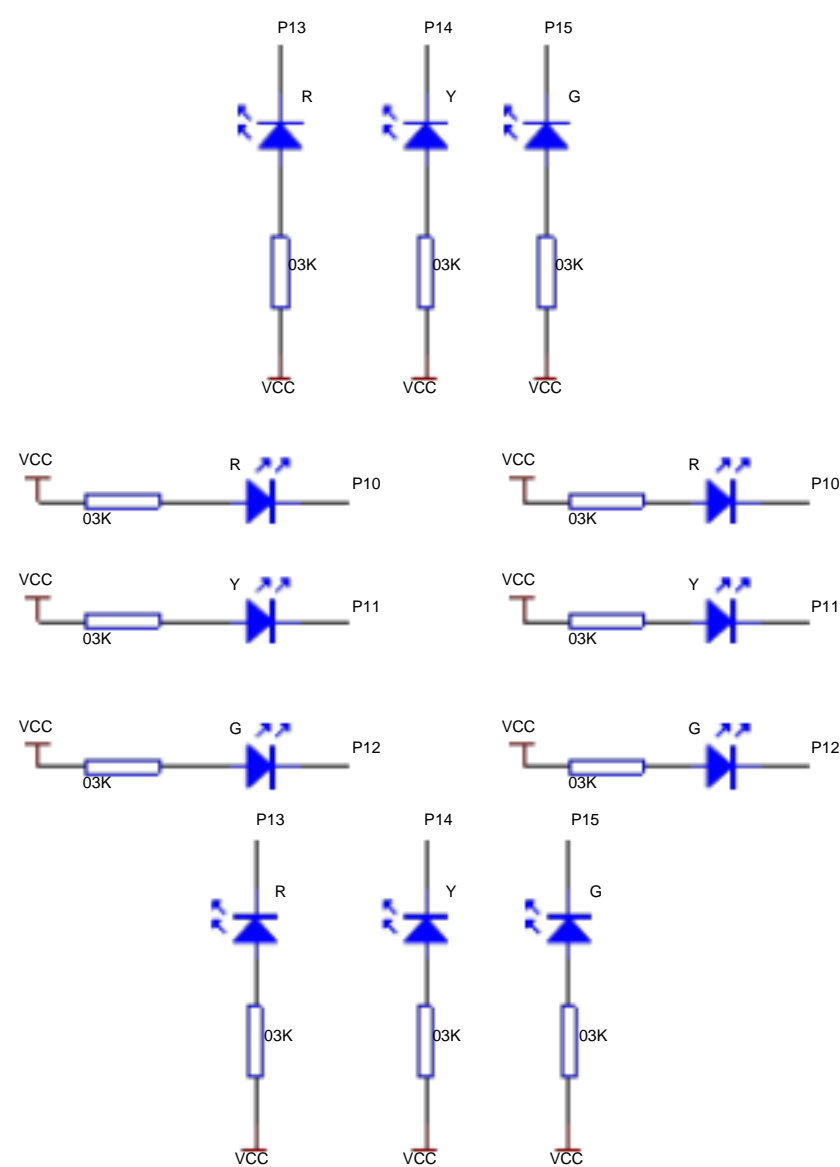


图3.7 信号灯的连接

3.5.2 七段LED 数码管

LED（Light Emitting Diode），发光二极管，它是一种固态的半导体器件，可以直接把电转化为光。LED的心脏是一个半导体的晶片，晶片的一端附在一个支架上，一端是负极，另一端连接电源的正极，使整个晶片被环氧树脂封装起来。半导体晶片由三部分组成，一部分是 P 型半导体，在它里面空穴占主导地位，另一端是 N 型半导体，在这边主要是电子，中间通常是 1 至 5 个周期的量子阱。当电流通过导线作用于这个晶片的时候，电子和空穴就会被推向量子阱，在量子阱内电子跟空穴复合，然后就会以光子的形式发出能量，这就是 LED 发光的原理。而光的波长也就是光的颜色，是由形成 P-N 结的材料决定的。

LED 显示屏作为大型显示设备的一种，具有亮度高、价格低、寿命长、维护

简便等优点。LED数码管的结构简单，分为七段和八段两种形式，也有共阳和共阴之分。以八段共阴管为例，它有 8个发光二极管（比七段多一个发光二极管，用来显示 SP，即点），每个发光二极管的阴极连在一起。这样，一个 LED数码管就有1根位选线和 8根段选线，要想显示一个数值，就要分别对它们的高低电平来加以控制。由于交通灯数码管没有 SP位，本文主要讨论共阴七段 LED数码显示管，其他类形的显示管与其类似。

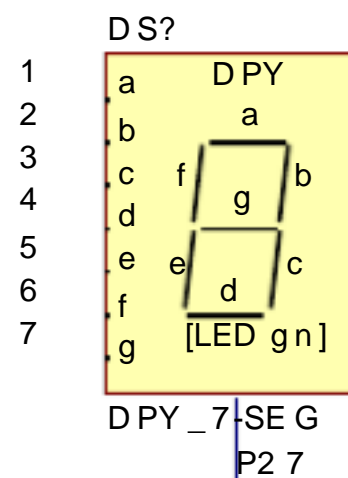


图3.8 LED 数码管

LED 灯的显示原理：通过同名管脚上所加电平的高低来控制发光二极管是否点亮而显示不同的字形，如 a, b, c, d, e, f, g全亮，则数码管显示为 8。采用共阴极连接：

表 3-3 驱动代码表

显示数值	a b c d e f g	驱动代码（ 16进制）
0	1 1 1 1 1 1 0	0x3f
1	0 1 1 0 0 0 0	0x06
2	1 1 0 1 1 0 1	0x5b
3	1 1 1 1 0 0 1	0x4f
4	0 1 1 0 0 1 1	0x66
5	1 0 1 1 0 1 1	0x6d
6	1 0 1 1 1 1 1	0x7d
7	1 1 1 0 0 0 0	0x07
8	1 1 1 1 1 1 1	0x7f
9	1 1 1 1 0 1 1	0x6f

3.5.3 电源电路设计

不管是 AT89S51单片机工作电源、二极管还是数码管的驱动，都要用到 +5V 的直流电源，因此，一个稳定的、持续的 +5V直流电源对本系统十分重要。本设计运用桥式整流电路，将交流转换为直流，可为各部分电路提供恒定的 +5V直流。

220V左右的交流电压，经变压器后转换为15V左右的电压，再经全波整流电桥整流后，得到一幅0-8V左右的波动直流。这一波动的直接经过电解电容C4滤波后，得到一个较平稳的直流，再经过LM7805稳压为+5V后，得到稳定的+5V直流电流，为系统提供稳定电压，保证电路的稳定性和抗干扰性，其电路如图3.9所示。图中C5用于抵消输入线较长时的电感效应，以防止电路产生自激震荡，其容量较小，本设计选择的为 0.31μ 的电容。图中C6用于消除输出电压中的高频噪声，本设计选用 0.1μ 的电容。

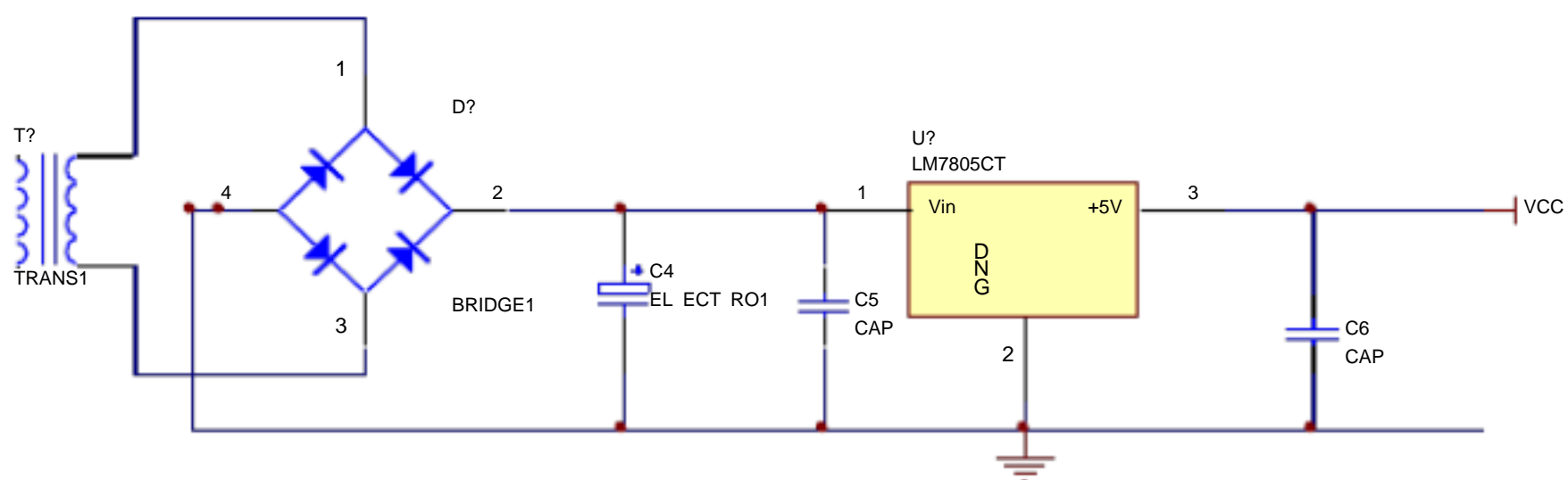


图3.9 电源电路

3.5.4 蜂鸣器

本设计采用一般蜂鸣器，蜂鸣器使用PNP极管进行驱动控制，当连接到单片机上的引脚输出为低电平，PN导通，蜂鸣器蜂鸣；当连接到单片机上的引脚输出高电平时，PN截止，蜂鸣器停止蜂鸣。如图3.10所示

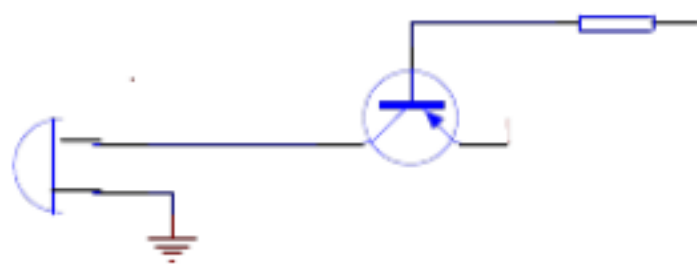


图3.10 蜂鸣器连接

3.6 本章小结

本章首先对单片机、车流量传感器和电源模块进行了选型，接着介绍了总体设计电路图的构成及原理。第三部分详细介绍了AT89S51单片机各个管脚的功能和内部芯片，以及AT89S51单片机的最小系统。第四部分主要介绍了红外线传感器的原理。最后对其他硬件进行了介绍，如发光二极管，LED数码管以及电源电路等。

第 4 章 软件设计

硬件平台结构一旦确定，大的功能框架基本形成。软件在硬件平台上构筑，完成各部分硬件的控制和协调。系统功能是由软硬件共同实现的，因为软件的可伸缩性，最终实现的系统功能可强可弱，差别可能很大。因此，系统是本系统的灵魂。软件采用模块化设计方法，不但易于编程和调试，也可以减小软件故障率和提高软件的可靠性。同时，对软件进行全面测试也是检验错误排除故障的重要手段。这里我采用了常用的 C 语言编程来实现的。

4.1 主程序设计

将整个流程分为四个状态如下：

- (1) 南北绿灯亮，东西红灯亮。此状态下，南北允许通行，东西禁止通行。
- (2) 南北黄灯闪烁 5s，东西保持红灯亮。此状态下除了已经正在通行中的其他所以车辆都需等待状态转换。
- (3) 东西绿灯亮，南北红灯亮。此状态下，东西允许通行，南北禁止通行。
- (4) 东西黄灯闪烁 5s，南北保持红灯亮。此状态下除了已经正在通行中的其他所以车辆都需等待状态转换。

主程序采用查询方式定时，获取交通灯的各种状态。主程序流程图如图 4.1 所示。

4.2 车流量采样程序设计

车流量采样程序主要功能是采样各路口的车流量，每次红灯转换成绿灯前两秒对路口车流量进行采样，然后根据采样后得到的车流量的大小来分配红绿灯的时间。根据两方向车流量的比例分成 5 个区域：小于 0.4，0.4-0.8，0.8-1.2，1.2-1.6，大于 1.6。

时间调整在此只划定 5 个范围。比例小于 0.4，表示南北方向车流量畅通，东西方向车流量拥挤。比例为 0.4-0.8 时，表示南北方向车流量比东西车流量少。比例为 0.8-1.2 时，表示南北车流量与东西车流量差距不大。比例为 1.2-1.6 时，表示南北车流量比东西车流量多。比例大于 1.6 时，表示南北方向车流量拥挤，东西方向车流量畅通。时间设置以 40s 为基准，当比例为 1.2-1.6 时，南北绿灯时间增加 10s，东西绿灯时间减少 10s。当比例为 1.6 以上时，南北绿灯时间增加 20s，东西绿灯时间减少 20s。当比例为 0.4-0.8 时，南北绿灯时间减少 10s，东西绿灯时间增加 10s。当比例为 0.4 以下时，南北绿灯时间减少 20s，东西绿灯时间增加 20s。

程序流程图如图 4.2所示。

根据红绿灯时间调整原理，一个周期下来，count_SN，count_EW中分别存储着南北，东西的车流量，接下来求单位时间车流量，此时南北向时间，东西向时间分别存储在 tt_SN，tt_EW中，则两个方向的流量比例为 (count_SN /tt_SN) / (count_EW/ tt_EW) =(count_SN * tt_EW) / (count_EW * tt_SN),显然该比例是 1左右带小数的值，然而单片机程序中只取整数，重要的数据信息就会丢失，所以本设计中首先将 (R5*R1) 乘以 10，比例就变为 10左右的值。将该比例值放在 A，然后根据 A的大小进行时间调整，具体时间设置表 4-1.

表 4-1 比例及调整时间

南北与东西向比例	0.4以下	0.4-0.8	0.8-1.2	1.2-1.6	1.6以上
调整南北向时间	20	30	40	50	60
调整东西向时间	60	50	40	30	20

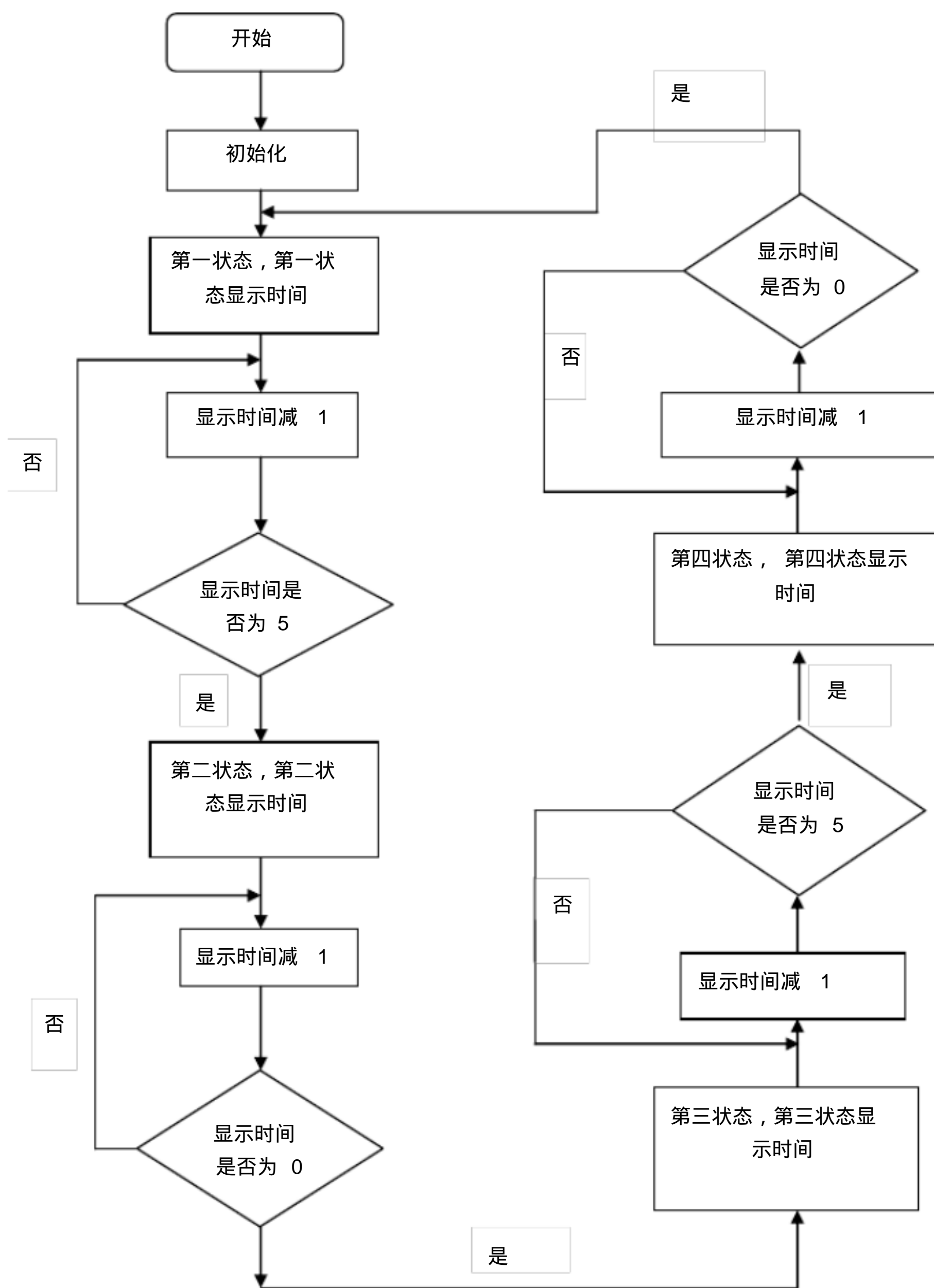
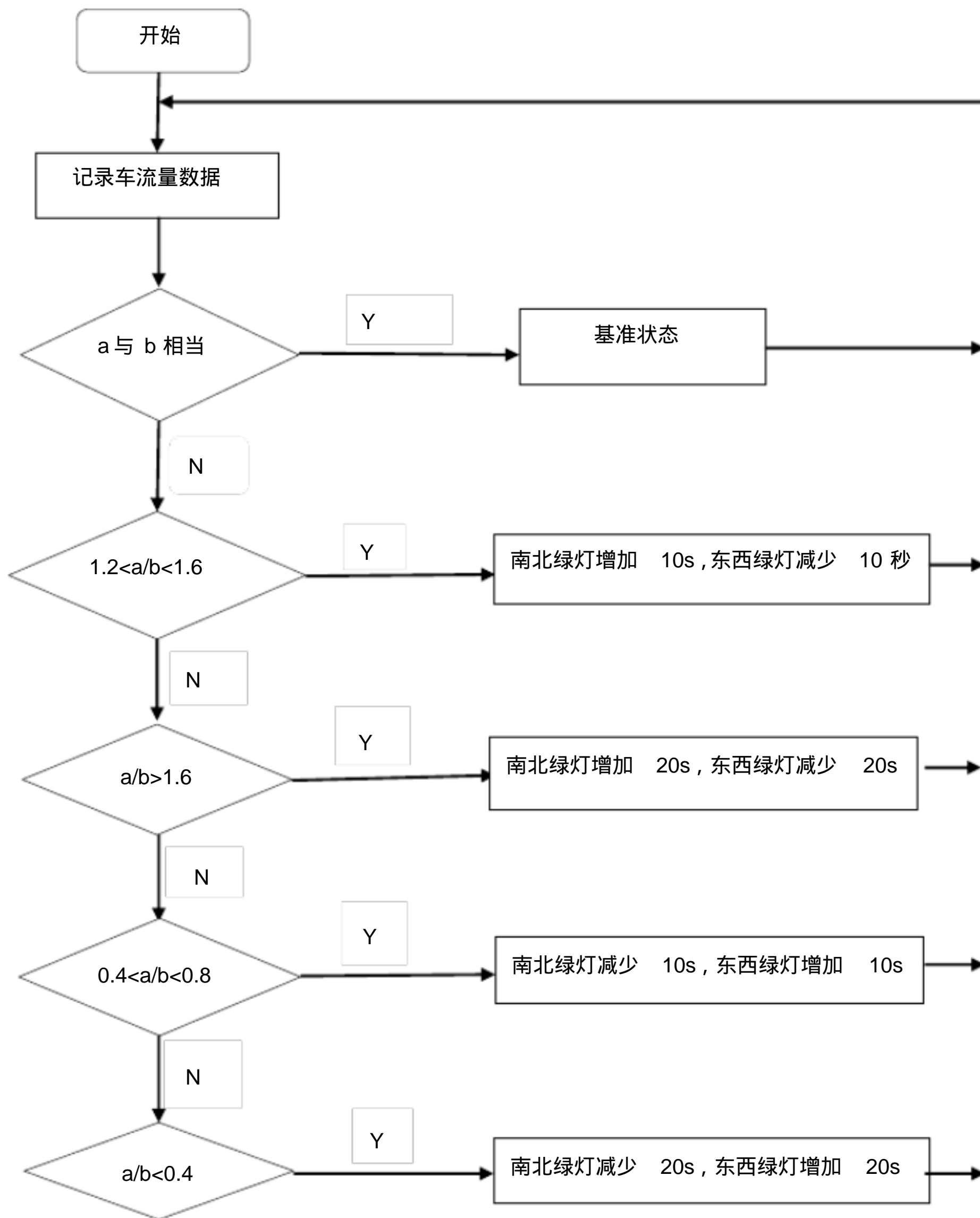


图 4.1 主程序流程图



4.2 车流量程序流程图

4.3 显示程序设计

交通灯的时间显示主要是通过数码管， 本设计采用动态显示数码管。 其流程图如图 4.3所示。

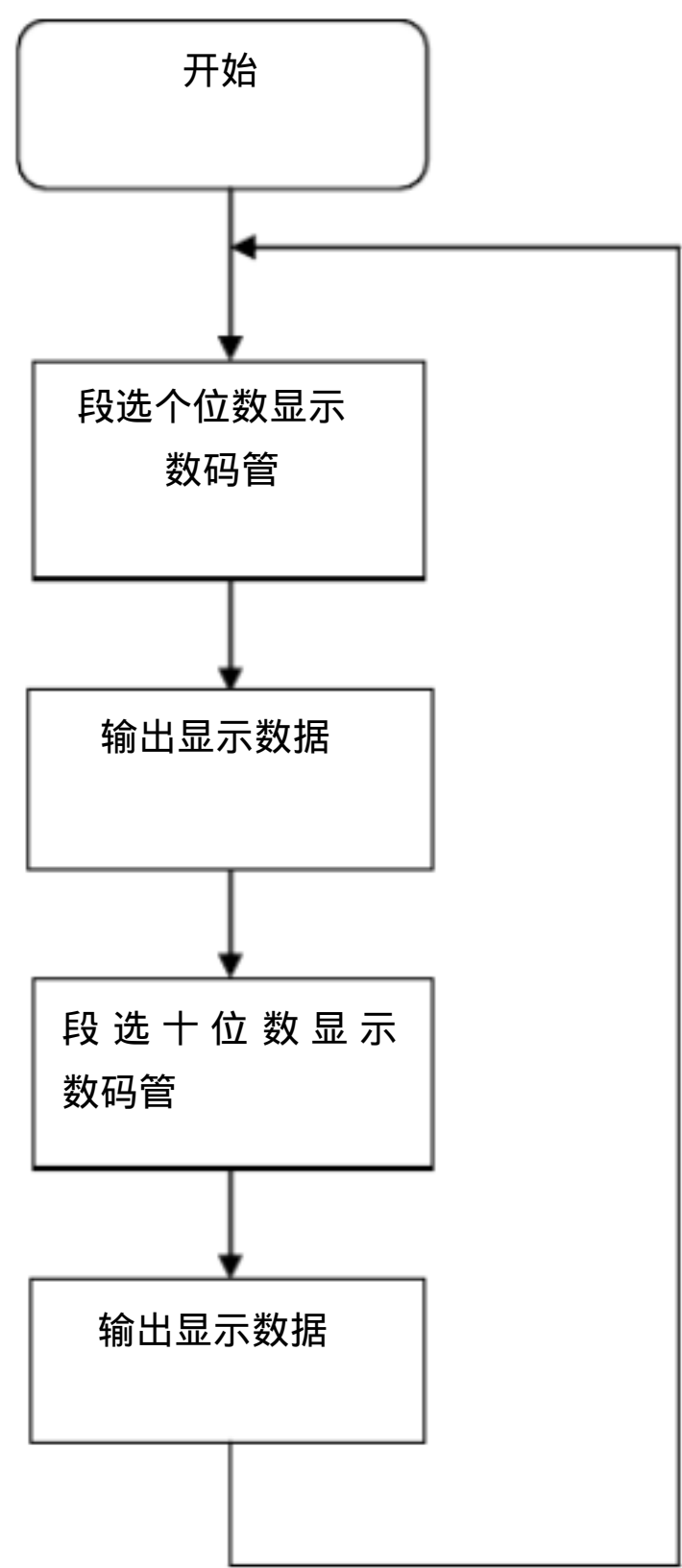


图 4.3 显示程序流程图

4.4 理论知识

4.4.1 定时器原理

定时器工作的基本原理其实就是给初值，让它不断加 1 直至减完为模值，这个初值是送到 TH 和 TL 中的。它是以加法记数的，并能从全 1 到全 0 时自动产生溢出中断请求。因此，我们可以把计数器记满为零所需的计数值，即所要求的计数值设定为 C，把计数初值设定为 TC 可得到如下计算通式：

$$TC=M-C$$

式中，M为计数器模值。计数值并不是目的，目的是时间值，设计1次的时间，即定时器计数脉冲的周期为T0，它是单片机系统主频周期的12倍，设要求的时间值为T，则有 $C=T/T_0$ 。计算通式变为：

$$T=(M-TC)T_0$$

模值和计数器工作方式有关。在方式0时M为8192；在方式1时M的值为65536；在方式2和3为256。就此可以算出各种方式的最大延时。如单片机的主脉冲频率为12MHZ，经过12分频后，若采用方式0最大延时只有8.129毫秒，采用方式1最大延时也只有65.536毫秒。这就是为什么扫描周期为50ms的原因，

若使用软件则会耽搁程序流程，显然不可行。相反，时间计时方面却不可能只用计数器，因为显然1秒钟已经超过了计数器的最大定时间，所以我们还必须采用定时器和软件相结合的办法才能解决这个问题。

4.4.2 软件延时原理

MCS-51的工作频率为12MHZ，机器周期与主频有关，机器周期是主频的12倍，所以一个机器周期的时间为 $12*(1/12\text{MHZ})=1\mu\text{s}$ 。我们可以知道具体每条指令的周期数，这样我们就可以通过指令的执行条数来确定1秒的时间，但同时由于单片机的运行速度很快其他的指令执行时间可以忽略不计。

我们设定一个初值为20的软件计数器和使T0定时50毫秒。这样每当T0到50毫秒时CPU就响应它的溢出中断请求，进入他的中断服务子程序。在中断服务子程序中，CPU先使软件计数器减1，然后判断它是否为零。为零表示1秒已到。设定定时器需要定时50毫秒，故T0必须工作于方式1。要求初值：

$$TC=M-T*T_0=216-50\text{ms}/1\mu\text{s}=15536=3\text{CBOH}.$$

4.5 本章小结

本章主要对智能交通灯软件方面进行了介绍。主要包括了三个方面的软件设计思路及原理：交通灯主程序的设计，车流量采样程序的设计以及数码管显示程序的设计。最后概述了定时器原理和软件延时原理。

第 5 章 智能交通灯的仿真

5.1 Proteus软件介绍

Proteus软件是英国 Labcenter electronics公司开发的电路分析与实物仿真的软件。它运行于 Windows操作系统上，可以仿真、分析（ SPICE）各种模拟器件和集成电路，软件的特点是：

（1）实现了单片机仿真和 SPICE电路仿真相结合。具有模拟电路仿真、数字电路仿真、单片机外围电路组成的系统的仿真、 RS232动态仿真、 IIC 调试器、SPI调试器、键盘和 LCD 系统仿真的功能：各种虚拟仪器，如示波器、逻辑分析仪、信号发生器灯。

（2）支持主流单片机系统的仿真。目前支持的单片机类型有： 6800系列、8051系列、 AVR 系列、 PIC14系列、 PIC16系列、 PIC18系列、 Z80系列、 HC11系列以及各种外围芯片。

（3）提供软件调试功能。硬件仿真系统中具有全速、单步、设置断点等调试功能，同时可以观察各个变量、 寄存器等的当前状态， 因此在该软件仿真系统中，必须具有这些功能；同时支持第三方的软件编译和调试环境，如 Keil C51 uVison4等软件。

（4）具有强大的原理图绘制功能。总之，该软件是一款集单片机和 SPICE 分析于一身的仿真软件，功能非常强大。

5.2 仿真过程介绍

5.2.1 用 PROTEUS绘制原理图

运行PROTEUS VSM的ISIS后出现相应的界面，点状的栅格区域为绘图区。在左侧的上方为电路图概览区，下方则是元器件列表区。单击 P后出现的 Pick Device添加元器件的对话框，输入所要添加的器件名称，则该器件就会出现在右侧，单击 OK按钮，完成一个元器件的添加。重复以上过程，添加好电路中所需的元器件。在元器件列表区选中某元器件后，在电路图概览区会出现该元器件，用鼠标将其拖至绘图区，将所有需要的元器件在绘图区放置好，既开始连线。连线方法很简单，将鼠标移至元器件引脚后出现一个小十字，再次单击左键就完成了一次连线。在布线时，如果需要转弯，可以在要转弯处单击鼠标左键。

5.2.2 PROTEUS对单片机内核的仿真

PROTEUS嵌入式系统仿真软件在设计时就已经注意到和单片机各种编译程序的整合了，如它可以和 Keil，Wave6000等编译模拟软件结合使用。因为 Keil 使用方便，具有强大的软件仿真和硬件仿真功能。把 Proteus和Keil 结合起来调试硬件就方便了好多，我采用的就是 “Proteus+Keil的仿真方法，具体步骤如下：

首先运行 PROTEUS VSM的ISIS,画好原理图后，用鼠标左键双击单片机出现如图 5.1所示定义代码生成工具对话框。

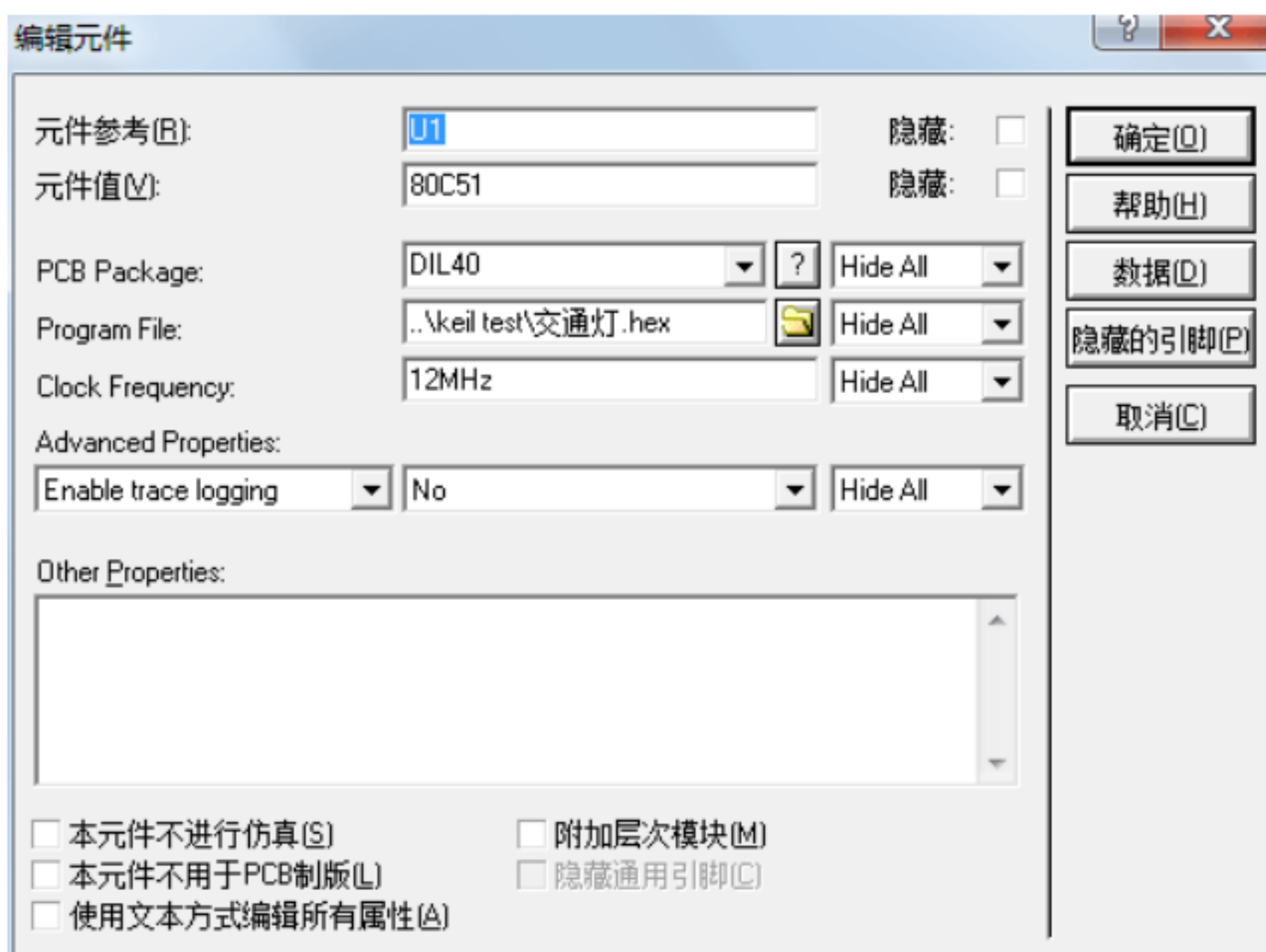


图 5.1 定义代码生成工具对话框

用Keil 软件写好程序，并编译生成相应的 HEX文件。电路图绘制完成后，再添加AT89S52的应用程序。鼠标移至 AT89S51上，双击鼠标左键，打开如图 5.2 所示的对话框。在 Program File 栏添加编译好的十六进制格式的程序文件 AA.hex（可以接受 3种格式的文件），再给 AT89S5输入晶振频率。此处默认为 12MHZ，单击OK按钮完成程序添加工作，下面就可以进行系统仿真了。单击主界面下方的按钮开始系统仿真。PROTEUS VSM进行的是一种交互式仿真，在仿真进行中可以对各控制按钮、开关等进行操作，系统对输入的响应会被真实的反映出来。在这个例子里，开始仿真后，开关按钮通过鼠标单击来改变状态，改变的状态会在LED和数码管显示出来。

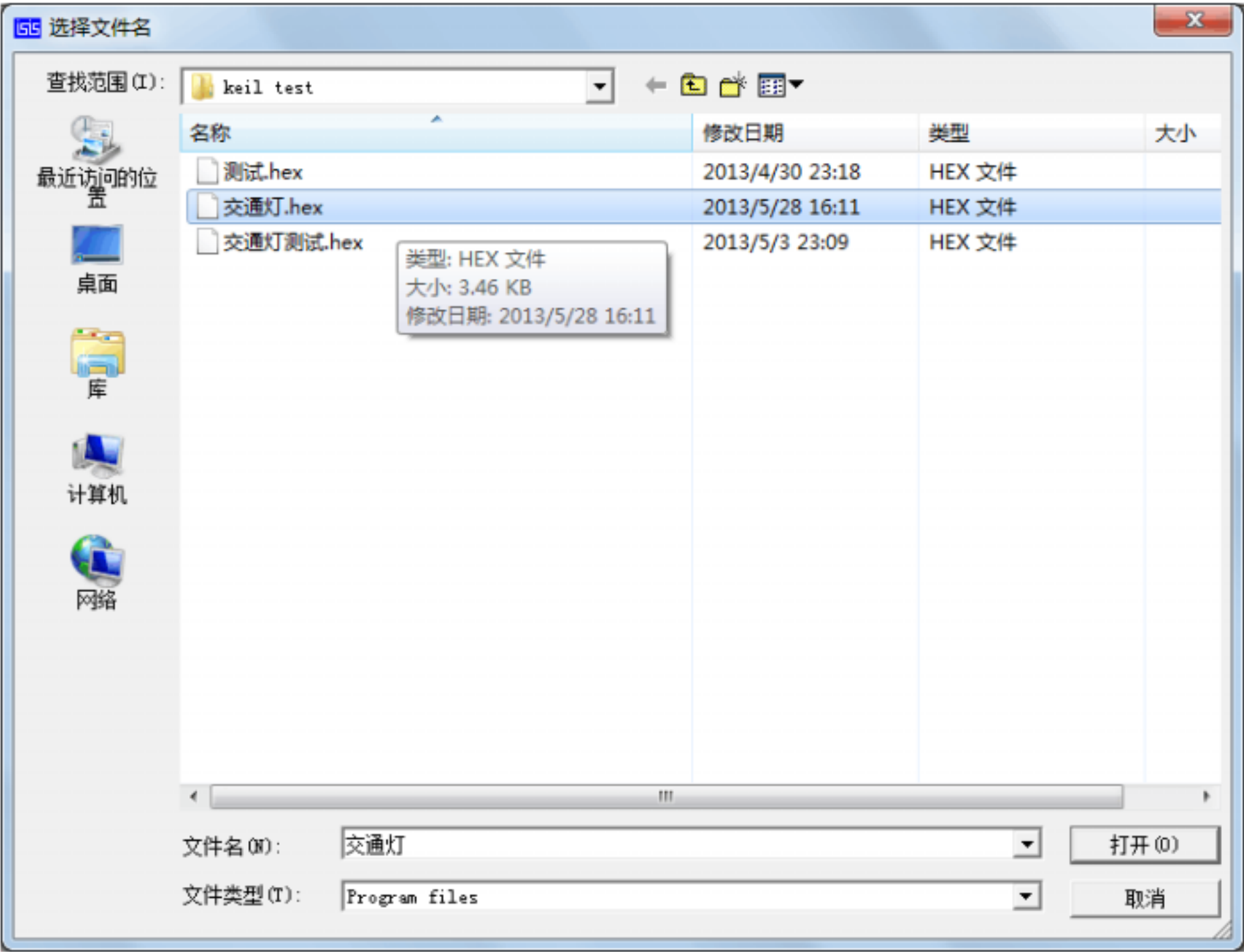


图 5.2 创建源代码对话框

5.2.3 仿真结果与分析

设南北向为 a路口，东西向车为 b路口。

（1）当交通灯开始工作后，执行初始状态。系统自动进入状态 1：南北向绿灯，东西向红灯，倒计时 40秒。倒计时到 5s时，南北向黄灯开始闪烁 5s，东西向保持红灯。然后，东西向绿灯亮，南北向红灯亮，倒计时 40秒。倒计时到 5s时，东西向黄灯开始闪烁 5s，南北向保持红灯。仿真结果如图 5.3 所示。

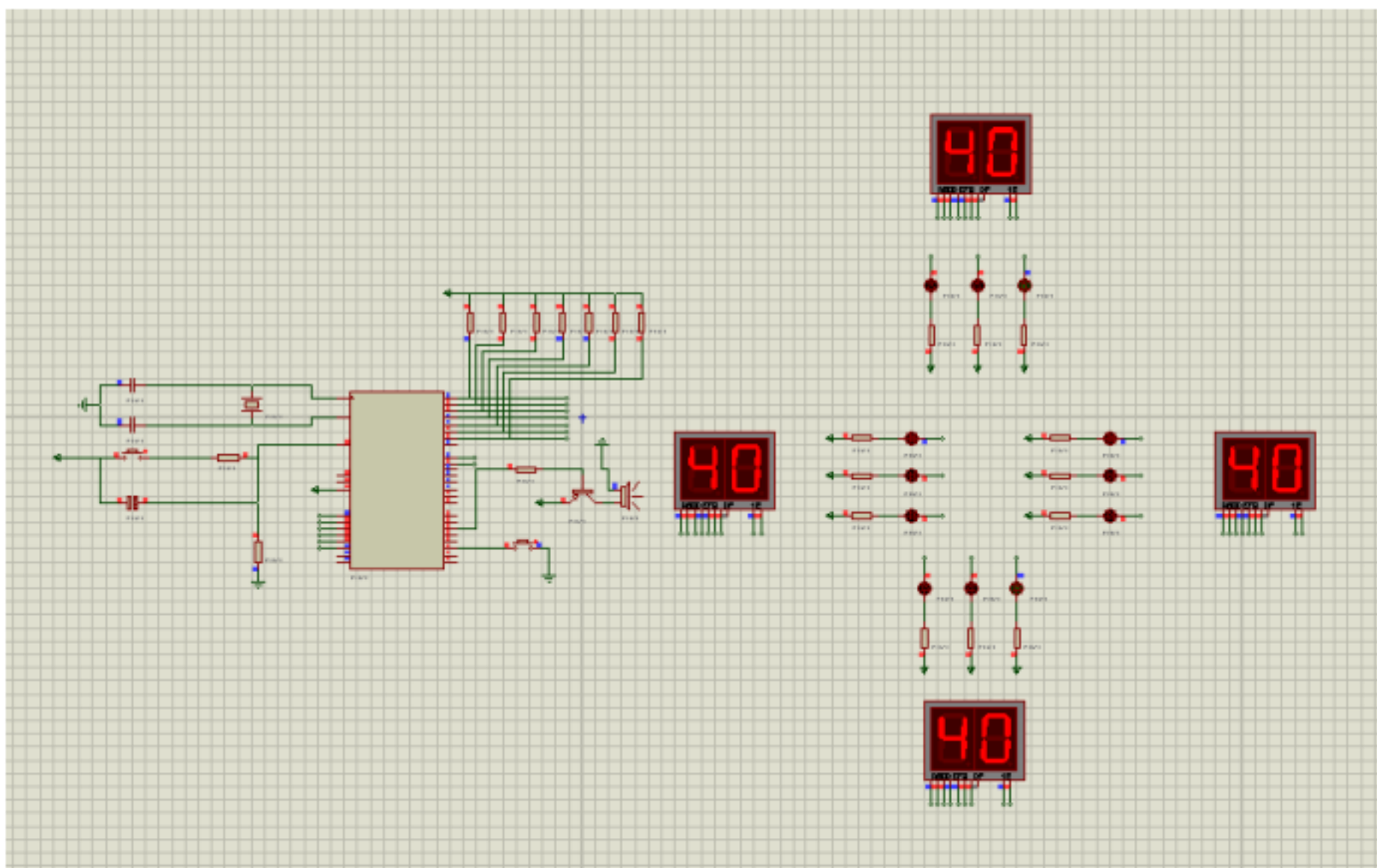
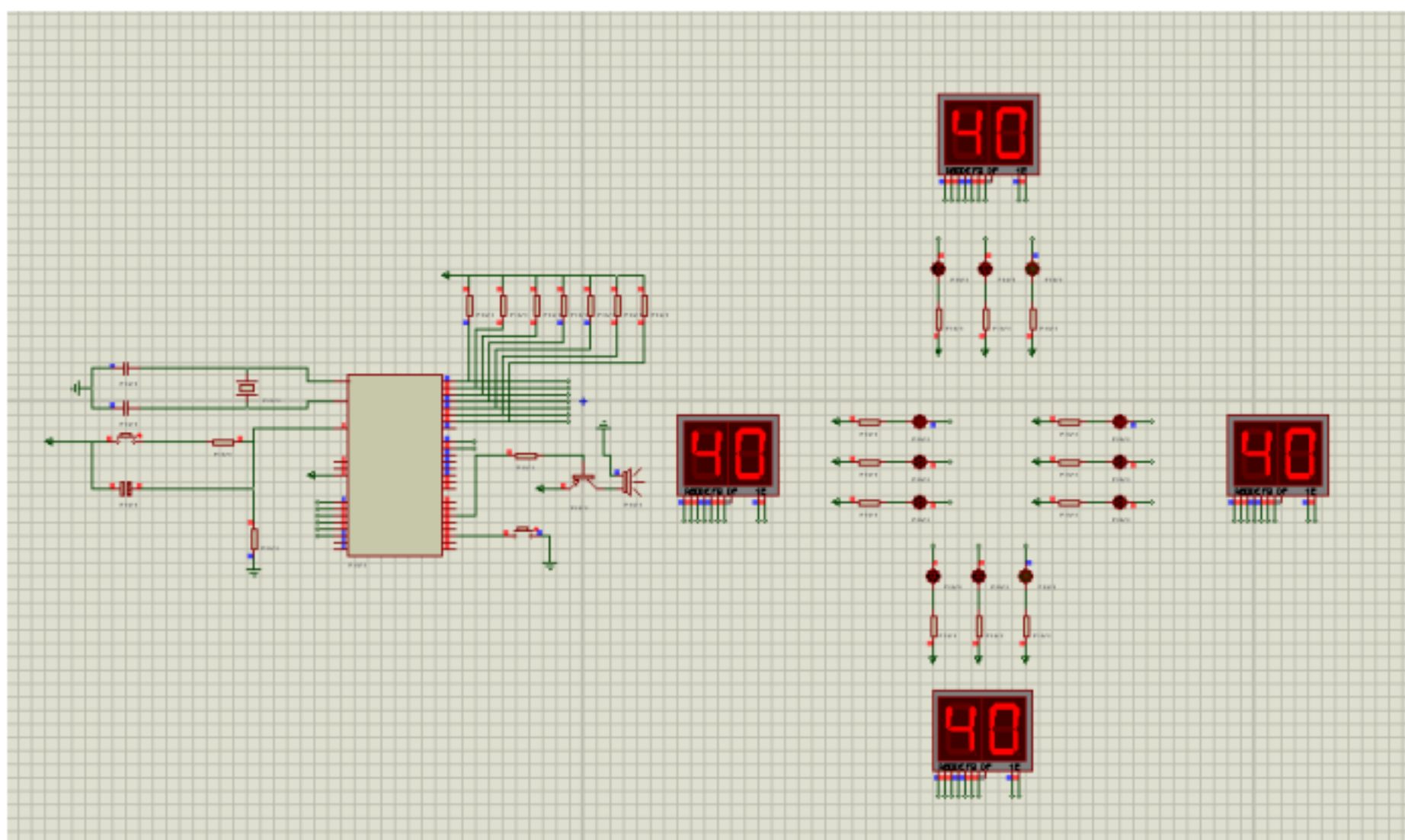


图 5.3 默认状态仿真图

(2) 当手动按下开关，使 a路口车流量与 b路口流量相当时，既 $0.5 < a/b \leq 1.2$ 时。系统将红绿灯时间调整为 40s、40s。南北向绿灯，东西向红灯，倒计时 40 秒。倒计时到 5s 时，南北向黄灯开始闪烁 5s，东西向保持红灯。然后，东西向绿灯亮，南北向红灯亮，倒计时 40 秒。倒计时到 5s 时，东西向黄灯开始闪烁 5s，南北向保持红灯。仿真结果如图 5.4 所示。



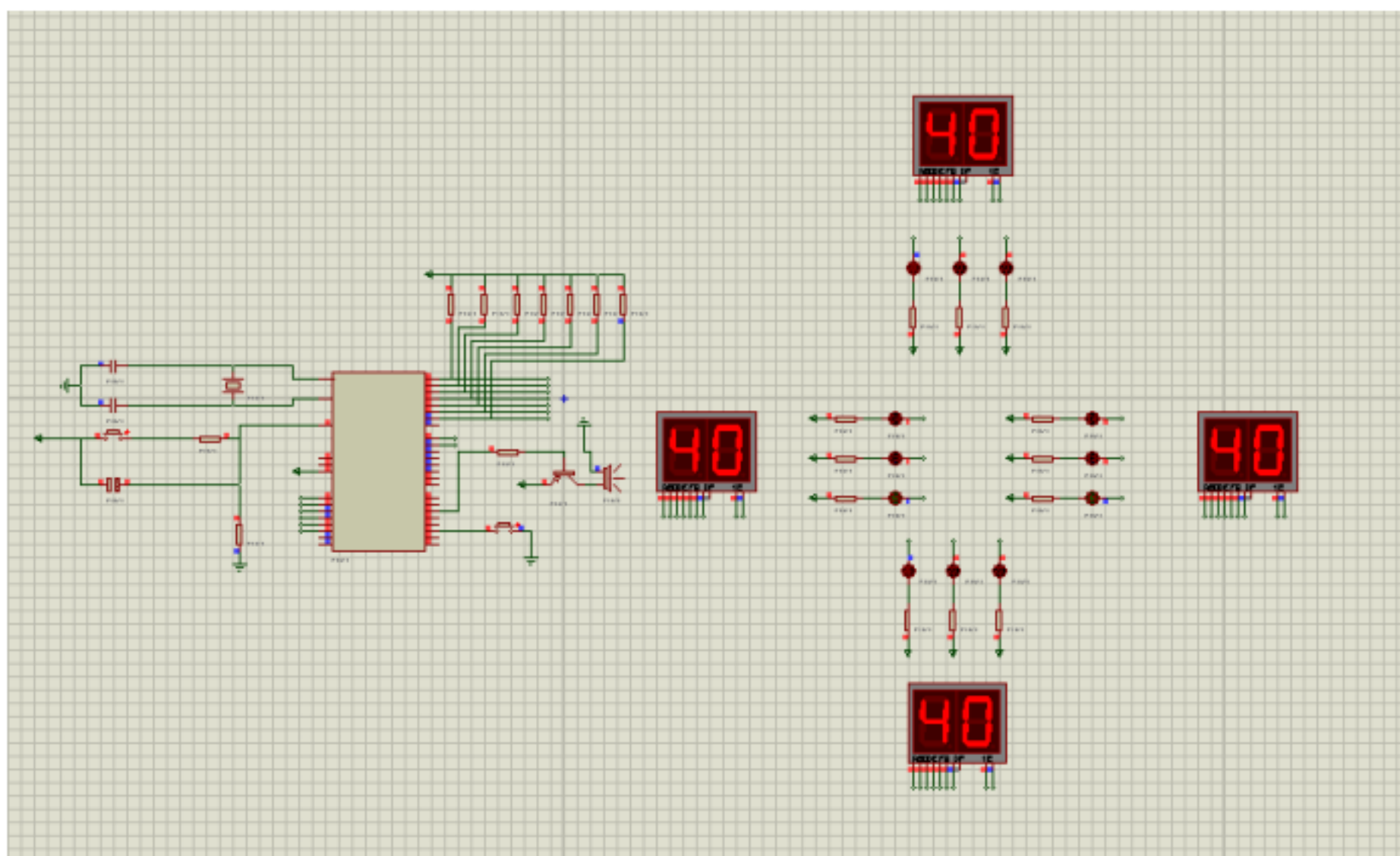
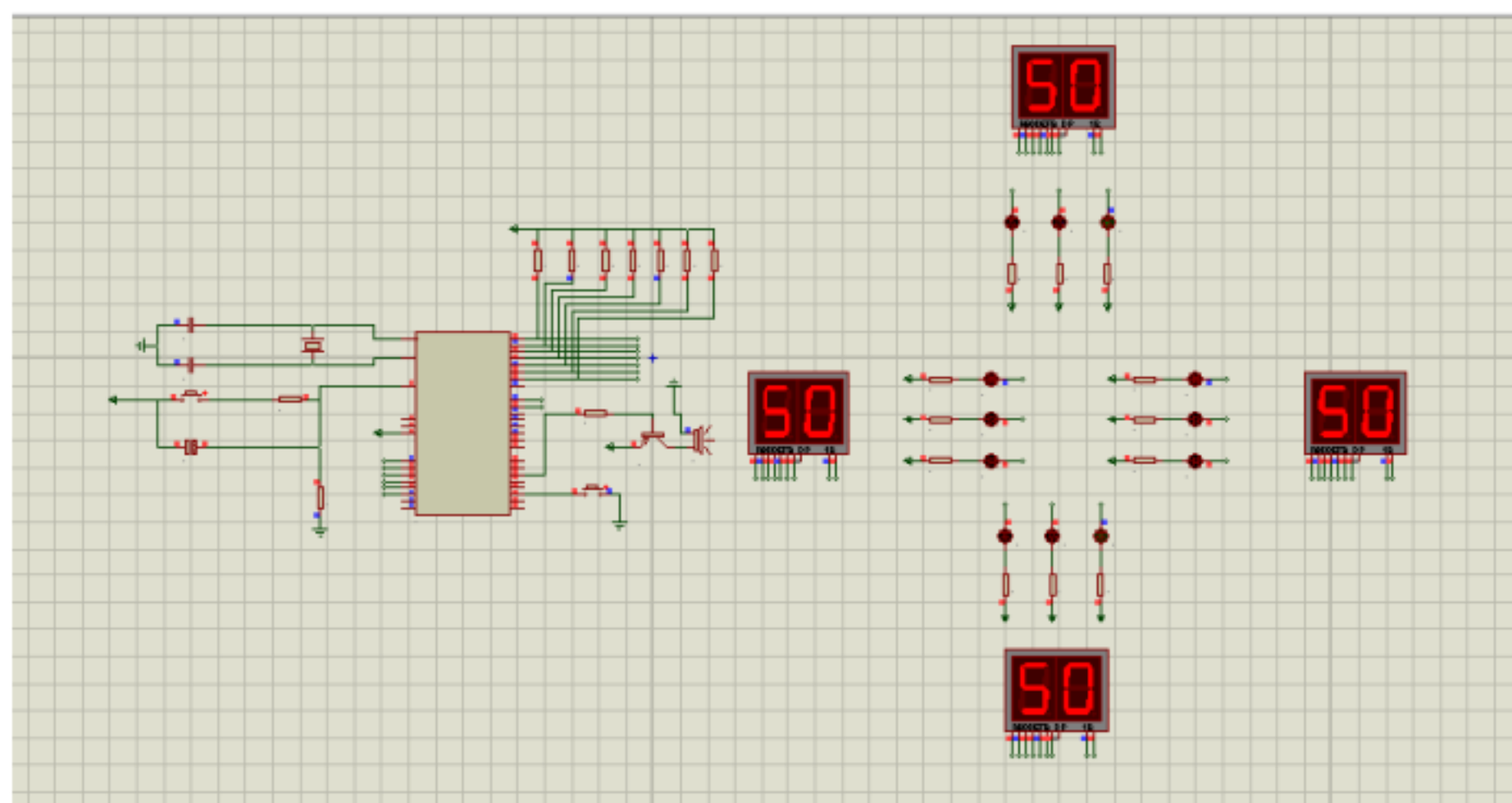


图 5.4 a/b=1时仿真图

(3) 当手动按下开关, 使 a路口车流量大于 b路口流量, 即比例为 1.2-1.6时。系统将红绿灯时间调整为 50s、30s。南北向绿灯, 东西向红灯, 倒计时 50秒。倒计时到 5s时, 南北向黄灯开始闪烁 5s, 东西向保持红灯。然后, 东西向绿灯亮, 南北向红灯亮, 倒计时 30秒。倒计时到 5s时, 东西向黄灯开始闪烁 5s, 南北向保持红灯。仿真结果如图 5.5所示。



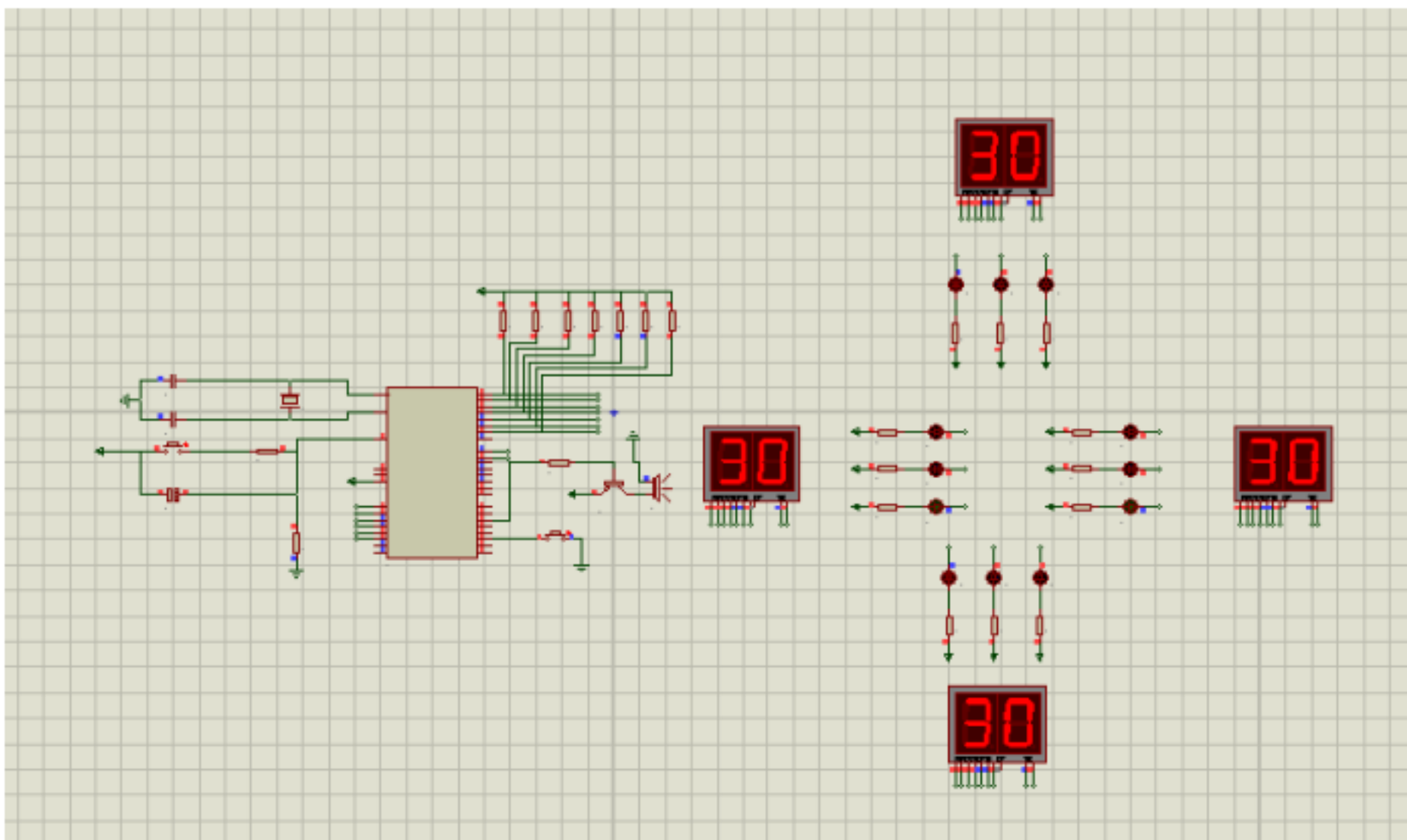
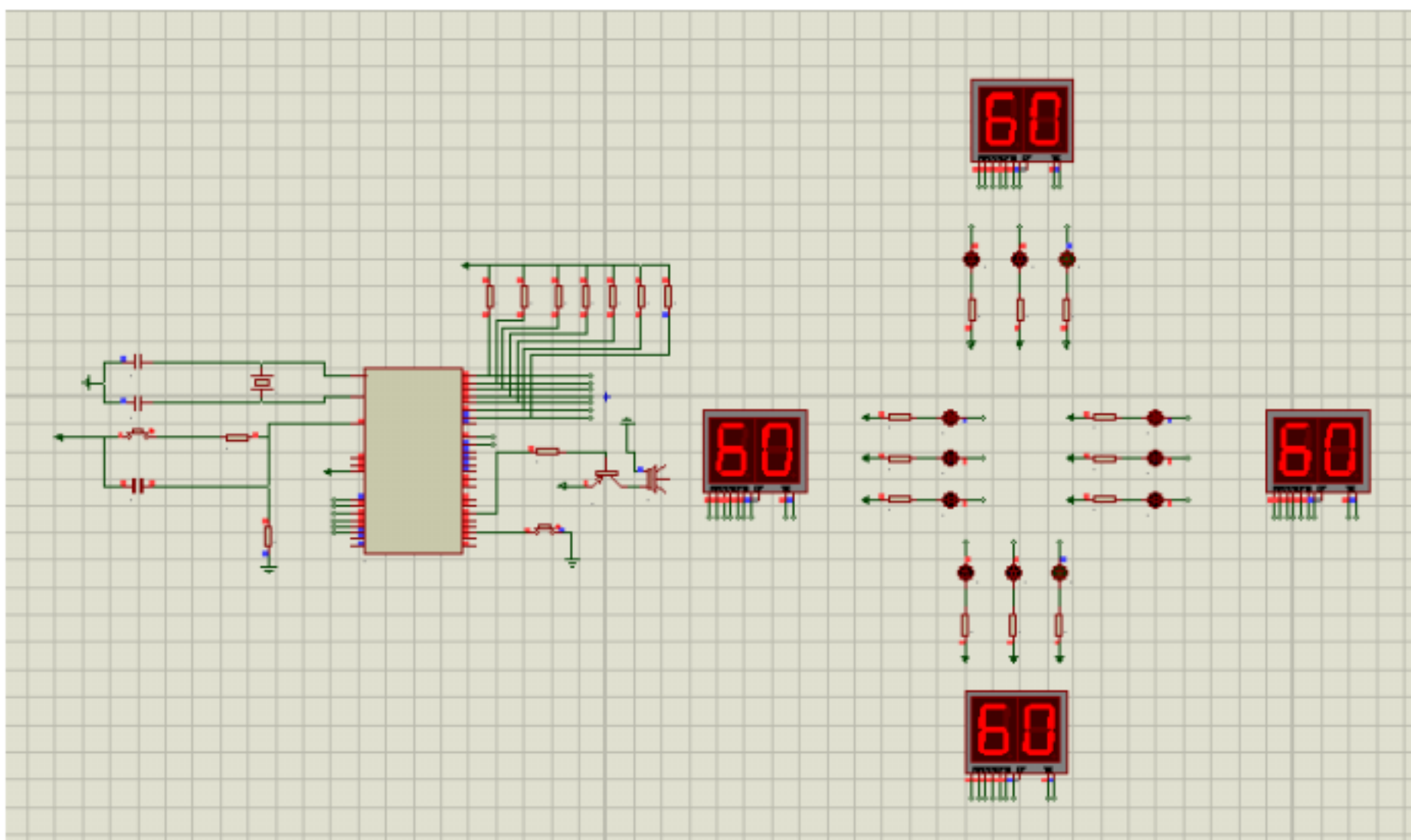


图5.5 比例为 1.2-1.6时仿真图

(4)当手动按下开关，比例为 1.6以上时。系统将红绿灯时间调整为 60、20s。南北向绿灯，东西向红灯，倒计时 60秒。倒计时到 5s时，南北向黄灯开始闪烁 5s，东西向保持红灯。然后，东西向绿灯亮，南北向红灯亮，倒计时 20秒。倒计时到 5s时，东西向黄灯开始闪烁 5s，南北向保持红灯。仿真结果如图 5.6所示。



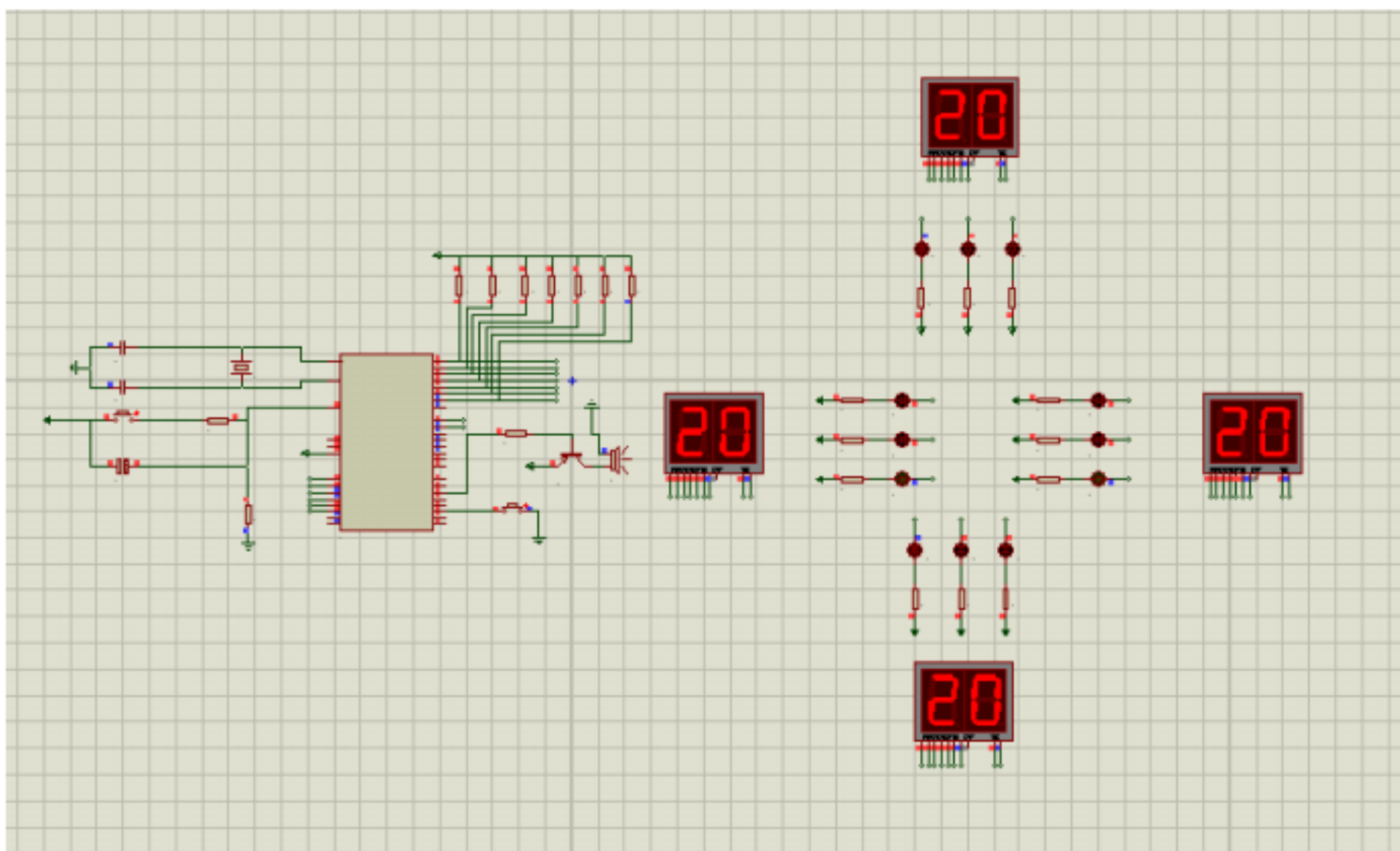
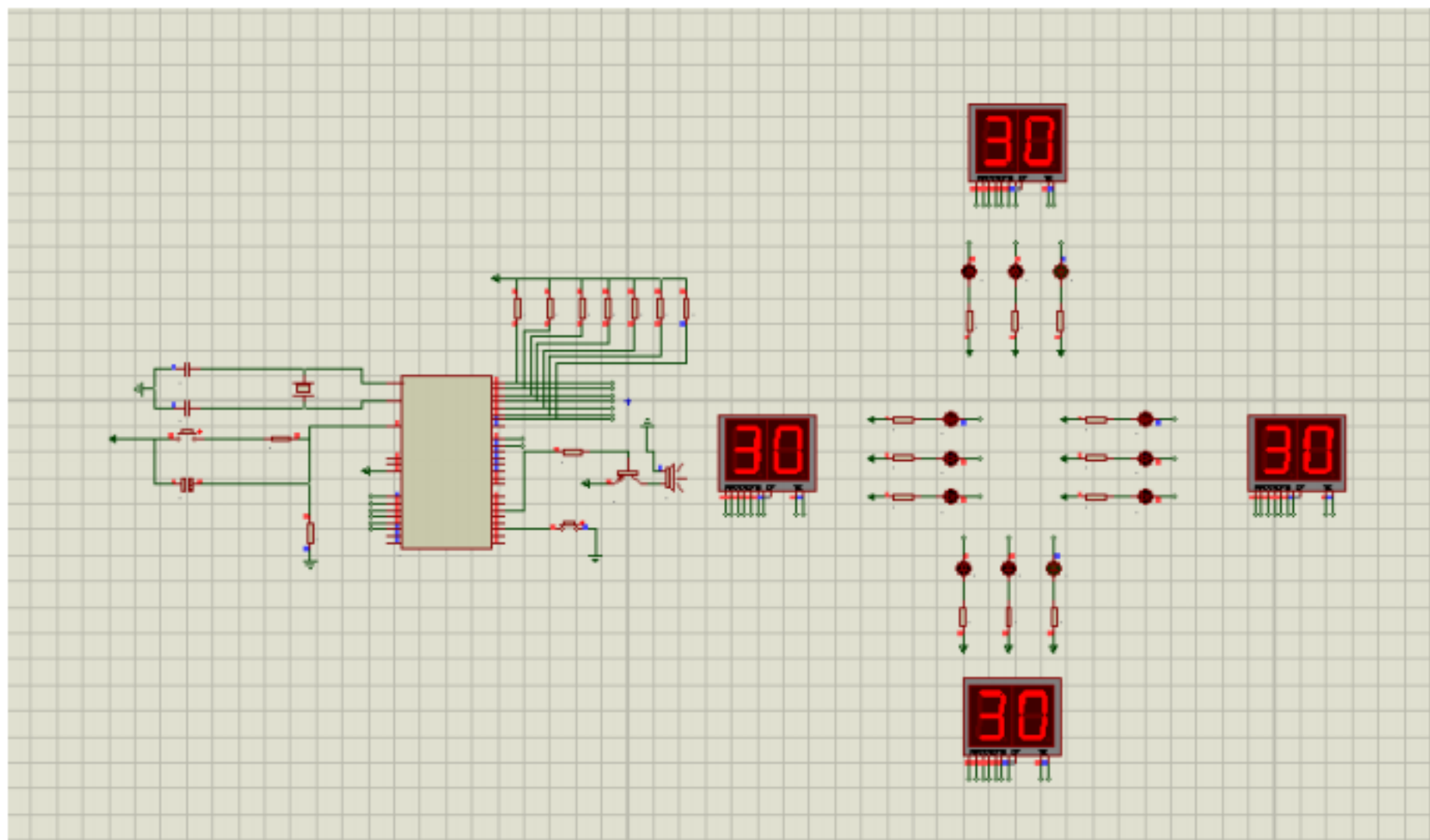


图 5.6 比例大于 1.6 时的仿真图

(5) 当手动按下开关，比例为 0.4-0.8 时。系统将红绿灯时间调整为 30、50s。南北向绿灯，东西向红灯，倒计时 30 秒。倒计时到 5s 时，南北向黄灯开始闪烁 5s，东西向保持红灯。然后，东西向绿灯亮，南北向红灯亮，倒计时 50 秒。倒计时到 5s 时，东西向黄灯开始闪烁 5s，南北向保持红灯。仿真结果如图 5.7 所示。



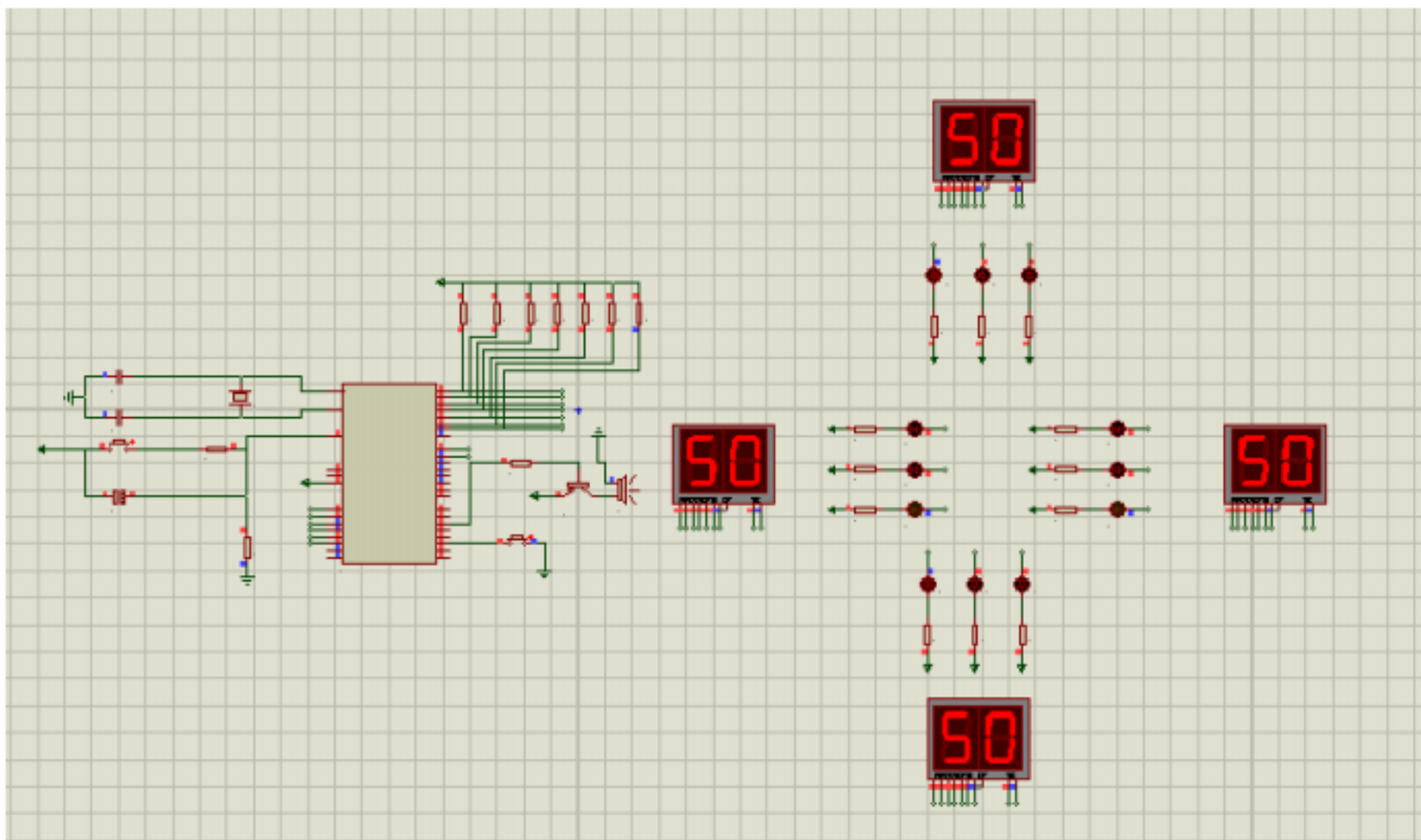
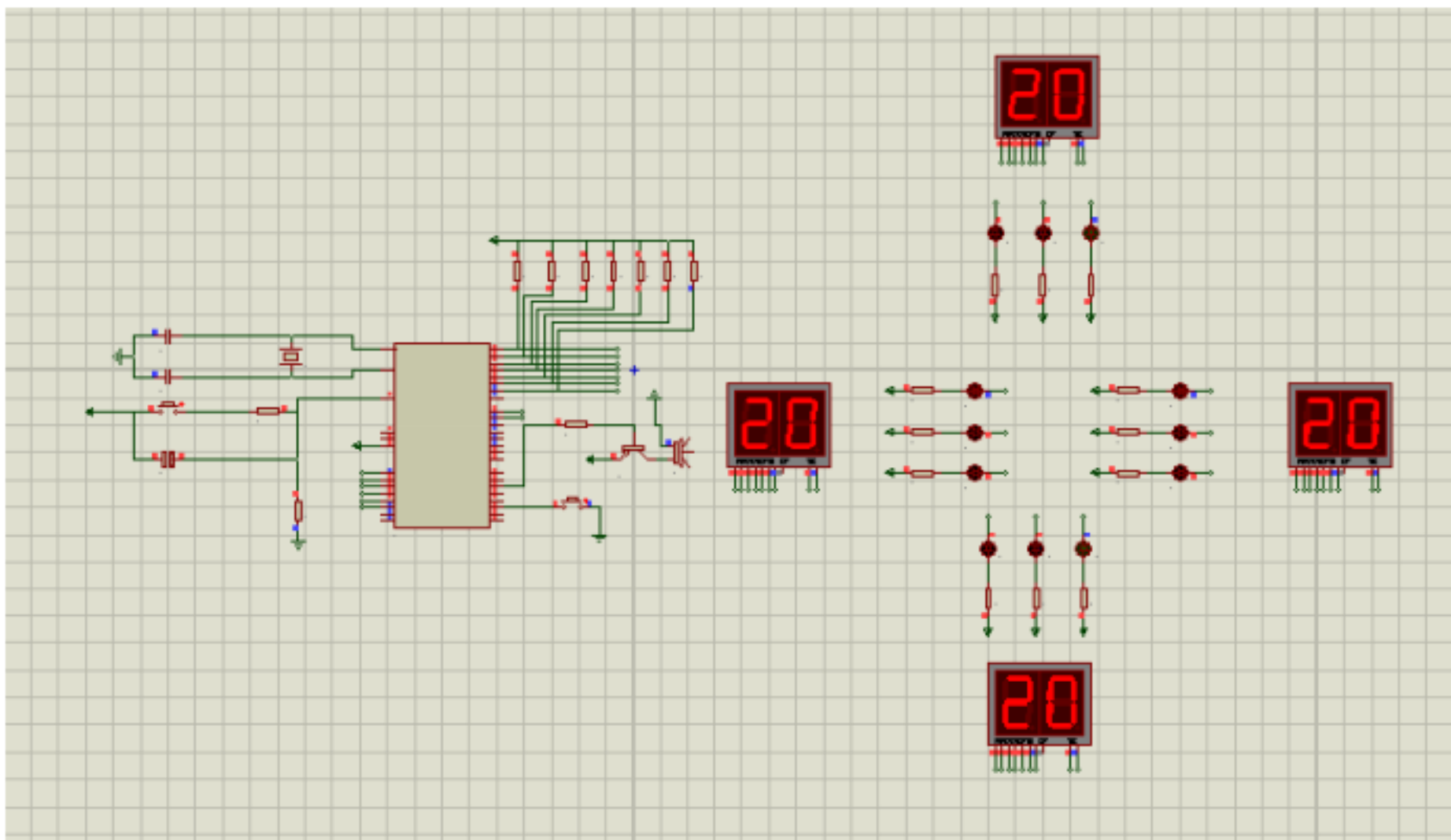


图 5.7 比例为 0.4-0.8 时的仿真图

(6) 当手动按下开关，比例小于 0.4时。系统将红绿灯时间调整为 20、60s。南北向绿灯，东西向红灯，倒计时 20秒。倒计时到 5s时，南北向黄灯开始闪烁 5s，东西向保持红灯。然后，东西向绿灯亮，南北向红灯亮，倒计时 60秒。倒计时到 5s时，东西向黄灯开始闪烁 5s，南北向保持红灯。仿真结果如图 5.8所示。



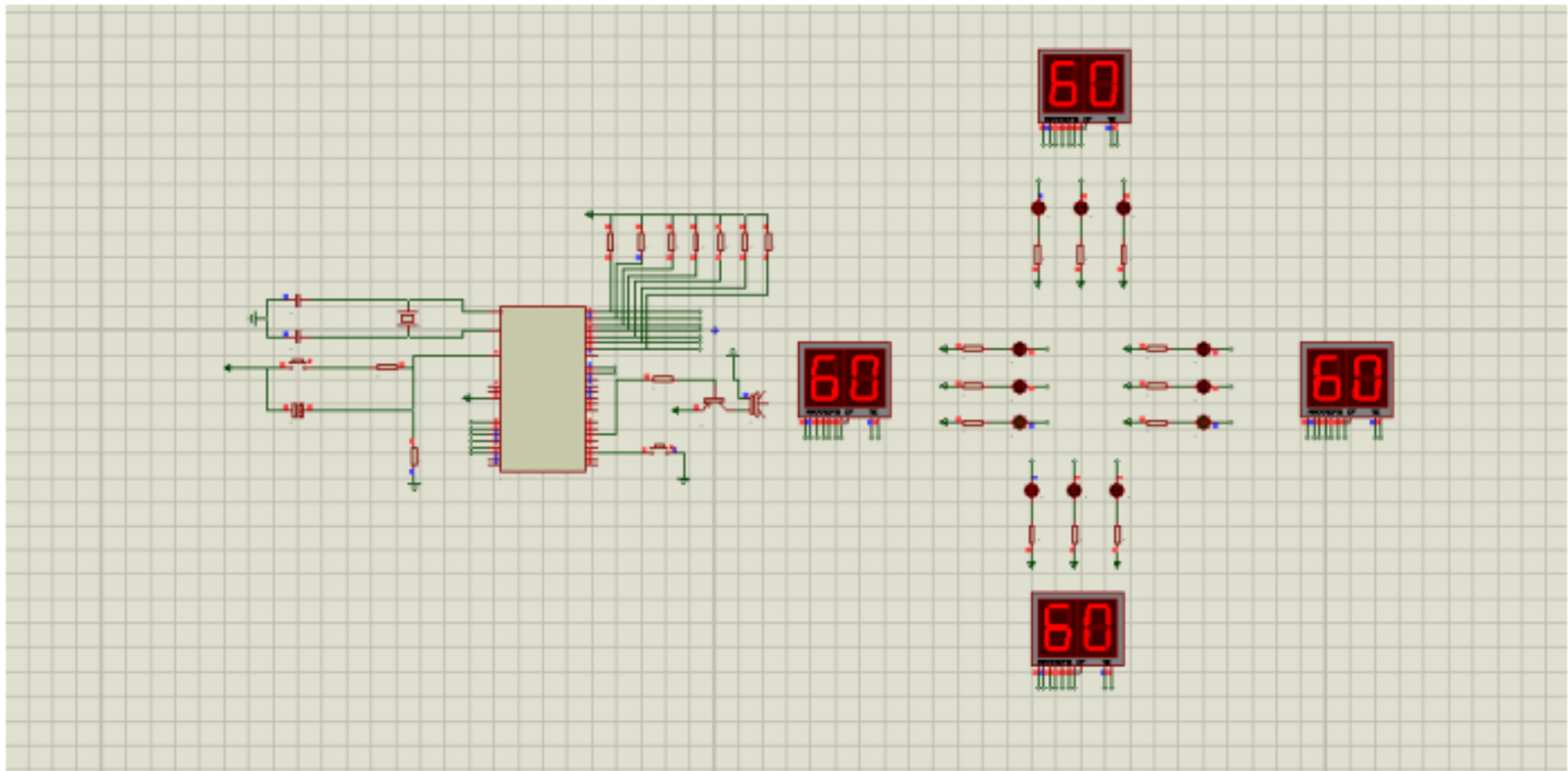


图 5.8 比例小于 0.4 时的仿真图

5.3 本章小结

本章首先对 proteus 软件进行了介绍，后面展示了各种情况的仿真模拟图。

结束语

交通灯控制在交通运输领域有着非常重要的作用。 本系统采用以单片机为中心控制器，实现了传统的红绿灯显示和倒读秒功能， 并能根据车流量的变化自动调整红绿灯时间。

首先通过信息采集， 主要是对路口各方向的车流量进行采集。 本设计采用比例的方法利用红外线传感器只计算出需比较两个相对通行路口车流量的比值即可。该系统可完成红绿灯传统的控制功能， 红绿黄灯按一定的规律依次循环显示， 并采用 LED显示器，以动态扫描的方式完成倒计时功能。在此基础上，通过传感器对车流量的情况进行数据采集。 将采集的数据以脉冲信号的方式传送还给控制中心，进行分析比较。 根据比较的结果， 将具体的车流量转换成两车流量大小的比值。根据比值与红绿灯时长的关系输出时长控制信号， 使交通信号灯可根据车流量改变，提高了交叉口的通行效率。

本论文以现有智能交通灯理论为基础， 突破传统思路，使问题简化。 将难点不再放在具体的车辆数的获得， 以及车辆数与红绿灯时间的数学模型的建立。 通过传感器取得两路口车辆数的比值。 因红绿灯以秒为单位， 这在控制中是精度很低的，将比值和秒数对应化数学模型很容易建立，便于实现， 使智能交通信号灯从理论进入实践成为一种可能。并且系统的结构简单，成本较低。 经进一步研究和完善，有望推向实际应用。

参考文献

- [1] 黄卫 . 智能运输系统 (ITS) 概论 [M] . 北京 : 人民交通出版社 , 2000.
- [2] 中国智能运输系统体系框架专题组 . 中国智能运输系统体系框架 [M] . 北京 : 人民交通出版社 , 2003
- [3] 陆化普 . 解析城市交通 [J] . 中国水利水电出版社 , 2001
- [4] 郭天祥 . 新概念 51 单片机 C 语言教程 — 入门、提高、开发、拓展 [M]. 电子工业出版社 .2009
- [5] 戴佳等 . 51 单片机应用系统开发典型实例 [M] . 中国电力出版社 , 2002
- [6] 杨欣等 . 电子设计从零开始 [M]. 清华大学出版社 .2010
- [7] 阎石 . 数字电子技术基础 (第五版) [M]. 高等教育出版社 .2006
- [8] 孙玉芳 . 基于单片机的智能交通灯控制系统研究 [D]. 哈尔滨工程大学 .2009
- [9] 温志达 , 梁桂荣 , 陈碧铭 , 高素萍 . 基于车流量的智能交通灯控制系统 [J]. 自动化技术与应用 .2009(6)
- [10] 余发山 . 单片机原理及应用技术 [M] . 中国矿业大学出版社 . 2001 : 25-58 页
- [11] 胡汉才 . 单片机原理及其接口技术 [M] . 清华大学出版社 , 2003 : 10-101 页
- [12] 王成勇 . 智能交通灯灯控制系统 [J]. 广东技术师范学院学报 .2006:92-94 页
- [13] 周润景 , 张丽娜 . 基于 PROTEUS 的电路及单片机系统设计与仿真 [M]. 北京航空航天大学出版社 , 2006
- [14] 胡伟 , 季晓衡 . 单片机 C 程序设计及应用实例 [M]. 北京 : 人民出版社 , 2003.
- [15] 范风强 . 单片机语言 C51 应用实战集锦 [M]. 北京 : 电子工业出版社 , 2003.
- [16] 徐玮 , 徐福军 , 沈建良编著 . C51 单片机高效入门 [M]. 北京 : 机械工业出版社 , 2006 : 32-38.
- [17] 卢庆林 . 数字动态扫描显示电路的设计 [J] . 2007 : 56—60 页
- [18] 王中亮 . 治理城市交通拥堵的对策思考 —— 以上海为例 [J] . 上海经济研究 , 2006:76-81 页
- [19] V. Yu. Teplov, A. V. Anisimov. Thermostatting System Using a Single-Chip Microcomputer and Thermoelectric Modules Based on the Peltier Effect [J], 2002.
- [20] Meehan Joanne, Muir Lindsey. SCM in Merseyside SMEs: Benefits and barriers [J]. TQM Journal. 2008.

谢 辞

本次设计是在翻阅了大量资料，以及在何佳老师的大力帮助和支持下完成的。他在设计中悉心指导，提出了许多宝贵的意见和建议，帮助我们解决了许多难题，对我们的设计起到了至关重要的作用，充分显示了一位经验丰富的教师精神之高尚、学识之渊博。

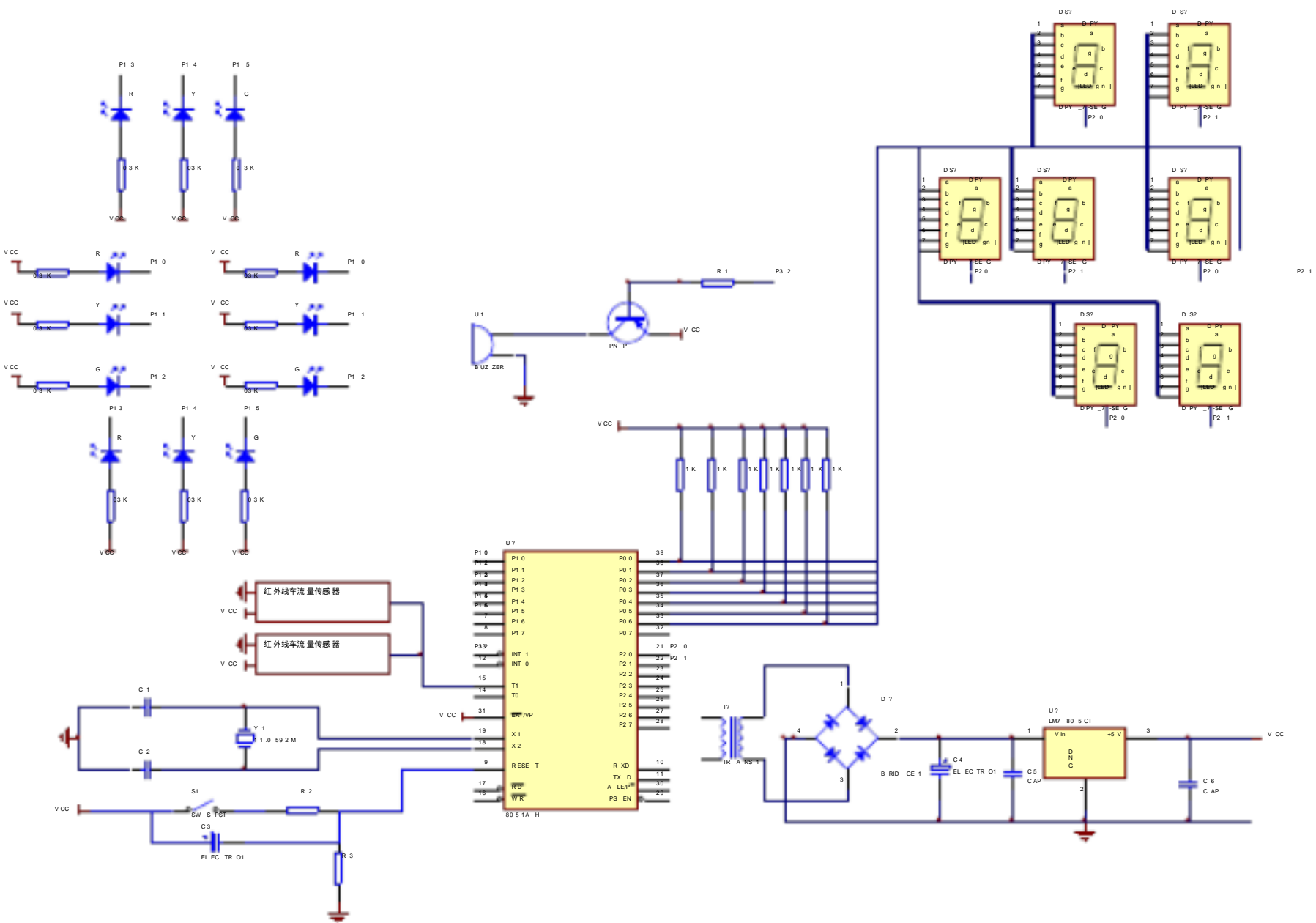
在设计过程中，我受益非浅，不仅使我由对课本上的知识粗浅认识有了更进一步的理解，而且还积累了很多课本以外的知识，开阔了我的视野，丰富了我的内涵，同时还积累了很多宝贵的经验，增强了独立思考的能力，当然在这个过程中也发现自己还有许多不足之处，但整个过程对我来说就像一个实际的眼帘，这对我来说无疑是一笔宝贵的财富。四年的大学生活即将结束了，但这并不意味着学习的终止，在今后的工作和学习中，我还要不断的学习，不断地丰富和发展自己的知识，不断地提高和完善自己，不断地进步。

设计过程是充实的，内容是丰富的，我从中获得了更深层的知识，同时也锻炼了我的毅志，培养了我的耐心和细致的工作作风和严谨的科学态度，了解了工作的艰辛，这对我以后的工作和学习是大有帮助的。

本次设计是在何老师的耐心指导和其他同学的大力协助下完成的，在此深表感谢。

附录

附录 A：交通灯电路图



附录 B: 交通灯控制程序

```
#include<reg51.h>
#define uint unsigned int
#define uchar unsigned char
sbit P32=P3^2;
sbit P35=P3^5;
unsigned char code table[]={
0x3f,0x06,0x5b,0x4f,
0x66,0x6d,0x7d,0x07,
0x7f,0x6f};
uchar
Operation_Type=1,count_SN,count_EW,count,tt=40,num,shi,ge,tt_SN=40,tt_EW=40;
uint aa,A;
void delay(uint z)           //延时程序 ( 1ms)
{
    uint x,y;
    for(x=z;x>0;x--)
        for(y=110;y>0;y--);
}
void Traffic_light()         //交通灯子函数
{
    switch(Operation_Type)
    {
        case 1:
            {
                P1=0x1e;
            }
            break;
        case 2:
            {
                P1=0x2e;
                delay(200);
                P1=0x3e;
                delay(200);
            }
    }
}
```



```
        }
        break;
    case 3:
        {
            P1=0x33;
        }
        break;
    case 4:
        {
            P1=0x35;
            delay(200);
            P1=0x37;
            delay(200);
        }
        break;
    }
}

void count_car()                //检测车流量子函数
{
    switch(Operation_Type)
    {
        case 1:
        {
            if(P35==0)
            {
                count++;
                delay(500);
            }
        }
        break;
        case 2:
        {
            count_SN=count;
        }
        break;
    }
}
```

```

    case 3:
    {
        if(P35==0)
        {
            count++;
            delay(500);
        }
    }
    break;
    case 4:
    {
        count_EW=(count-count_SN);
    }
    break;
}
}
void main()                                //主函数
{
    EA=1;                                    //开总中断
    TMOD=0X01;                              // 设置定时器
    TH0=(65536-50000)/256;
    TL0=(65536-50000)%256;
    ET0=1;                                  //定时器 0 允许中断
    TR0=1;                                  //开启定时器 0
    while(1)
    {
        Traffic_light();
        count_car();
        if(A<0)
        {
            P32=0;
        }
    }
}
void time0() interrupt 1                    //定时器中断

```

```
{
  TH0=(65536-50000)/256;
  TL0=(65536-50000)%256;
  aa++;
  if(aa==20)
  {
    aa=0;
    tt--;
    if(tt==5)
    {
      if(Operation_Type==5)
      {
        count=0;
        Operation_Type=1;
      }
      Operation_Type++;
      if(Operation_Type==4)
      {
        tt=5;
      }
      if(Operation_Type==2)
      {
        tt=5;
      }
    }
    if(tt==0)
    {
      Operation_Type++;
      if(Operation_Type==5)
      {
        count=0;
        Operation_Type=1;
      }
      if(Operation_Type==3)
      {
```

```
    tt=tt_EW;
}
if(Operation_Type==1)
{
    tt=tt_SN;
}
}
}
shi=tt/10;
ge=tt%10;
num++;
if(num==3)
{
    num=1;
}
switch(num)
{
    case 1:{P2=0xc2;P0=table[shi];}break;
    case 2:{P2=0xc1;P0=table[ge];}break;
    default:break;
}
delay(10);
if(Operation_Type==4)
{
    if(tt_SN=40,tt_EW=40)
    {
        A=(count_SN*40*10)/(count_EW*40);
    }
    else if(tt_SN=20,tt_EW=60)
    {
        A=(count_SN*60*10)/(count_EW*20);
    }
    else if(tt_SN=30,tt_EW=50)
    {
        A=(count_SN*50*10)/(count_EW*30);
```

```

    }
    else if(tt_SN=50,tt_EW=30)
    {
        A=(count_SN*30*10)/(count_EW*50);
    }
    else
    {
        A=(count_SN*20*10)/(count_EW*60);
    }
    if(A<=12&&A>8)
    {
        tt_SN=40;
        tt_EW=40;
    }
    if(A<=16&&A>12)
    {
        tt_SN=50;
        tt_EW=30;
    }
    if(A>16)
    {
        tt_SN=60;
        tt_EW=20;
    }
    if(A<=8&&A>4)
    {
        tt_SN=30;
        tt_EW=50;
    }
    if(A<4&&A>0)
    {
        tt_SN=20;
        tt_EW=60;
    }
    }
}

```

}