

目 录

- 一、前 言
- 1.1 课题简介
- 1.2 设计目的
- 二、总体设计
- 2.1 设计思路
- 2.2 原件清单
- 三、硬件设计
- 3.1 AT89C51
- 3.2 系统框图
- 3.3 程序框图
- 四、软件设计
- 4.1 硬件设计
- 4.2 单片机时钟电
- 4.3 复位电路
- 4.4 控制电路
- 4.5 工作电路
- 五、软件调试
- 5.1 设计要求
- 5.2 软件的流程图
- 5.3 程序设计
- 六、软件调试
- 七、心得体会
- 八、参考文献

第一章 前言

1.1 课题简介

单片机全称叫单片微型计算机（Single Chip Microcomputer），是一种集成在电路芯片，是采用超大规模集成电路技术把具有数据处理能力的中央处理器 CPU 随机存储器 RAM 只读存储器 ROM 多种 I/O 口和中断系统、定时器/计时器等功能（可能还包括显示驱动电路、脉宽调制电路、模拟多路转换器、A/D 转换器等电路）集成到一块硅片上构成的一个小而完善的计算机系统。

目前单片机渗透到我们生活的各个领域，几乎很难找到哪个领域没有单片机的踪迹。导弹的导航装置，飞机上各种仪表的控制，计算机的网络通讯与数据传输，工业自动化过程的实时控制和数据处理，广泛使用的各种智能 IC 卡，民用豪华轿车的安全保障系统，录像机、摄像机、全自动洗衣机的控制，以及程控玩具、电子宠物等等，这些都离不开单片机。更不用说自动控制领域的机器人、智能仪表、医疗器械以及各种智能机械了。单片机广泛应用于仪器仪表、家用电器、医用设备、航空航天、专用设备的智能化管理及过程控制等领域，大致可分如下几个范畴：1. 在智能仪器仪表上的应用：，例如精密的测量设备 2. 在工业控制中的应用：用单片机可以构成形式多样的控制系统、数据采集系统。例如工厂流水线的智能化管理，电梯智能化控制、各种报警系统，与计算机联网构成二级控制系统等。3.

在家用电器中的应用可从手机，电话机、小型程控交换机、楼宇自动通信呼叫系统、列车无线通信、再到日常工作中随处可见的移动电话，集群移动通信，无线电对讲机等。5. 单片机在医用设备领域中的应用：例如医用呼吸机，各种分析仪，监护仪，超声诊断设备及病床呼叫系统等等。6. 在各种大型电器中的模块化应用：如音乐集成单片机，看似简单的功能，微缩在纯电子芯片中（有别于磁带机的原理），就需要复杂的类似于计算机的原理。

本设计着重在于分析计算器软件 and 开发过程中的环节和步骤，并从实践经验出发对计算器设计做了详细的分析和研究。本系统就是充分利用了 8051 芯片的 I/O 引脚。系统以采用 MCS-51 系列单片机 Intel8051 为中心器件来设计 LED 流水灯系统，实现 8 个 LED 霓虹灯的左、右循环显示，并实现循环的速度可调。

1.2 设计目的

1. 学习基本理论在实践中综合运用的初步经验，掌握电路设计的基本方法、设计步骤，培养综合设计与调试能力。
2. 掌握汇编语言程序设计方法。
3. 培养实践技能，提高分析和解决实际问题的能力。

1.3 设计任务及要求

1. 彩灯用 8 个发光二极管代替。
2. 电路具有控制彩灯点亮右移、左移、全亮及全灭等功能（用按键切换彩灯状态）
3. 彩灯两点移动时间间隔为 1 秒。

二、总体设计思路

2.1 设计思路

本课题使用 AT89C51 单片机时无须外扩存储器。因此，本流水灯实际上就是一个带有八个发光二极管的单片机最小应用系统，即为由发光二极管、晶振、复位、电源等电路和必要的软件组成的单个单片机。

如果要想接在 P1.0 口的 LED1 亮起来，那么只要把 P1.0 口的电平变为低电平就可以了；相反，如果要想接在 P1.0 口的 LED1 熄灭，就要把 P1.0 口的电平变为高电平；同理，接在 P1.1 ~ P1.7 口的其他 7 个 LED 的点亮和熄灭的方法同 LED1。因此，要实现流水灯功能，我们只要将发光二极管 LED1 ~ LED8 依次点亮、熄灭，8 只 LED 灯便会一亮一暗的做流水灯了。同样的道理，可以让 8 个灯左移点亮，全亮、全灭。

在此我们还应注意一点，由于人眼的视觉暂留效应以及单片机执行每条指令的时间很短，我们在控制二极管亮灭的时候应该延时一段时间，否则我们就看不到“流水”效果了。

在此基础上，增加外扩设备，利用 74LS373 和 8255 扩展成 24 个 LED 灯循环显示。

2.2. 原件清单

元件名称	型号	数量/个	用途
单片机	AT89S51	1	控制核心
晶振	12MHZ	1	晶振电路
电容	30pF	2	晶振电路
电阻	Respack8	8	上位电阻
电阻	10k	4	开关电阻
电阻	10k	1	复位电路
电源	+5v	1	提供电源
拨码开关	BUTTON	4	发出信号
数码管	7SEG-MPX1-CC	8	显示电路

三、硬件设计

3.1 AT89C51

1. 芯片由 UNTITLED\ISIS\PROFESSIONAL 实验系统提供
(AT89C51)

1) 主要特性:

与 MCS-51 兼容

4K 字节可编程闪烁存储器

寿命: 1000 写/擦循环

数据保留时间: 10 年

全静态工作: 0Hz—24Hz

三级程序存储器锁定

128*8 位内部 RAM

32 可编程 I/O 线

5 个中断源

可编程串行通道

低功耗的闲置和掉电模式

片内振荡器和时钟电路

2) 管脚说明:

① 电源引脚

Vcc (40 脚): 典型值 +5V。

Vss (20 脚): 接低电平。

②外部晶振

XTAL1 、 XTAL2分别与晶振两端相连接。

③输入输出口引脚：

P0 口：I/O 双向口。作输入口时，应先软件置“1”。

P1 口：I/O 双向口。作输入口时，应先软件置“1”。

P2 口：I/O 双向口。作输入口时，应先软件置“1”。

P3 口：I/O 双向口。作输入口时，应先软件置“1”。

④控制引脚：

RST 、 ALE/-PROG-PSEN、-EA/Vpp组成了 MSC-51的控制总线。

RST (9脚)：复位信号输入端（高电平有效）。

ALE/-PROG(30脚)：地址锁存信号输出端。

第二功能：编程脉冲输入。

-PSEN (29脚)：外部程序存储器读选通信号。

-EA/Vpp(31脚)：外部程序存储器使能端。

第二功能：编程电压输入端 (+21V)。

3) AT89C51单片机的 P 口特点：

P0 口：是一个 8 位漏极开路输出型双向 I/O 端口。作为输出端口时，每位能以吸收电流的方式驱动 8 个 TTL 输入，对端口写 1 时，又可作高阻抗输入端用。在访问外部程序或数据存储器时，它是时分多路转换的地址（低 8 位）/数据总线，在访问期间将激活内部的上拉电阻。

P1 口：P1 口是一个带有内部上拉电阻的 8 位双向 I/O 端口。P1 口的输出缓冲器可驱动（吸收或输出电流方式）4 个 TTL 输入。对端口写 1 时，通过内部的上拉电阻把端口拉到高电位，这时可作输入口。

P2 口作输入口使用时，因为内部有上拉电阻，那些被外部信号拉低的引脚会输出一个电流 (I_{il})。

P2 口：P2 口是一个带有内部上拉电阻的 8 位双向 I/O 端口。P2 口的输出缓冲器可驱动（吸收或输

个 TTL 输入。对端口写 1 时，通过内部的上拉电阻把端口拉到高电位，这时可作输入口。P2 口作输入口使用时，因为内部有上拉电阻，那些被外部信号拉低的引脚会输出一个电流 (I_{il})。在访问外部程序存储器时和 16 位外部地址的外部数据存储器（如执行 `MOVX @DPTR` 时，P2 口送出高 8 位地址。在访问 8 位地址的外部数据存储器（如执行 `MOVX @RI` 时，P2 口引脚上的内容（就是专用寄存器(SFR)区中的 P2 寄存器的内容），在整个访问期间不会改变。

P3 口： P3 口是一个带有内部上拉电阻的 8 位双向 I/O 端口。P3 口的输出缓冲器可驱动（吸收或输出电流方式）4 个 TTL 输入。对端口写 1 时，通过内部的上拉电阻把端口拉到高电位，这时可作输入口。P3 口作输入口使用时，因为内部有上拉电阻，那些被外部信号拉低的引脚会输出一个电流 (I_{il})。

(2) 排阻的作用

所谓排阻就是若干个参数完全相同的电阻，它们的一个引脚都连到一起，作为公共引脚。其余引脚正常引出。所以如果一个排阻是由 n 个电阻构成的，

只引脚，一般来说，最左边的那个是公共引脚。它在排阻上一般用一个色点标出来。排阻一般应用在数字电路上，比如：作为某个并行口的上拉或者下拉电阻用。使用排阻比用若干只固定电阻更方便。

堆栈是一个区域，是用来存放数据的，这个区域本身没有任何特殊之处，就是内部 RAM 的一份，特殊的是它存放和取用数据的方式，即所谓的‘先进后出，后进先出’，并且堆栈有特的数据传输指令，即‘**PUSH** 和 瓢 **OP**’，有一个特殊的专为其服务的单元，即堆栈指针 **SP** 每当执一次 **PUSH**指令时，**SP** 就（在原来值的基础上）自动加 1，每当执行一次 **POP**指令，**SP** 就（在原来值基础上）动减 1。由于 **SP** 中的值可以用指令加以改变，所以只要在程序开始阶段更改了 **SP** 值，就可以把堆栈设置在规定的内存单元中，如在程序开始时用一条 **MOVSP, #5FH** 指令，就时把堆栈设置在从内存单元 **60H**开始的单元中。一般程序的开头总有这么条设置堆栈指针的指令，因为开机时，**SP** 的初始值为 **07H**，这样就使堆栈从 **08H**单元 开始往后 **8H**到 **1FH**这个区域正是 8031 的第二、三、四工作寄存器区，经常要被使用，这会造成数的混乱。不作编写程序时，初始化堆栈指令也不完全相同，这是作者的习惯问题。当设置好堆栈区，并不意味着该区域成为一种专用内存，它还是可以象普通内存区域一样使用，只是一般情下编程者不会把它当成通内存用了。

3.3 开发过程

任务分析开始，我们假设已设计并制作好硬件下面就是编写软件的工作。在编写软件之前，首先要确定一些常数、地址，事实这些常数、地址在设计阶段已被直接或间接地确定下来了。如当某器件的连线设计好后，其地址也就确定了，当器件的功能被确定下来后，其控制字也就被确定了。然后用文本编辑器（如EDIT、CCED等）写软件，编写好后，用编译器对源程序文件编译，查错，直到没有语法错误，除了极简单程序外，一般应用仿真机对软件进行调试，直到程序运行正确为止。运行正确后，就可以写（将程序固化在EPROM）。在源程序被编译后，生成了扩展名为HEX的目标文件，一般编程器能够识别种格式的文件，只要将此文件调入即可写片。在此，为使大家对整个过程有个认识，举一说明：

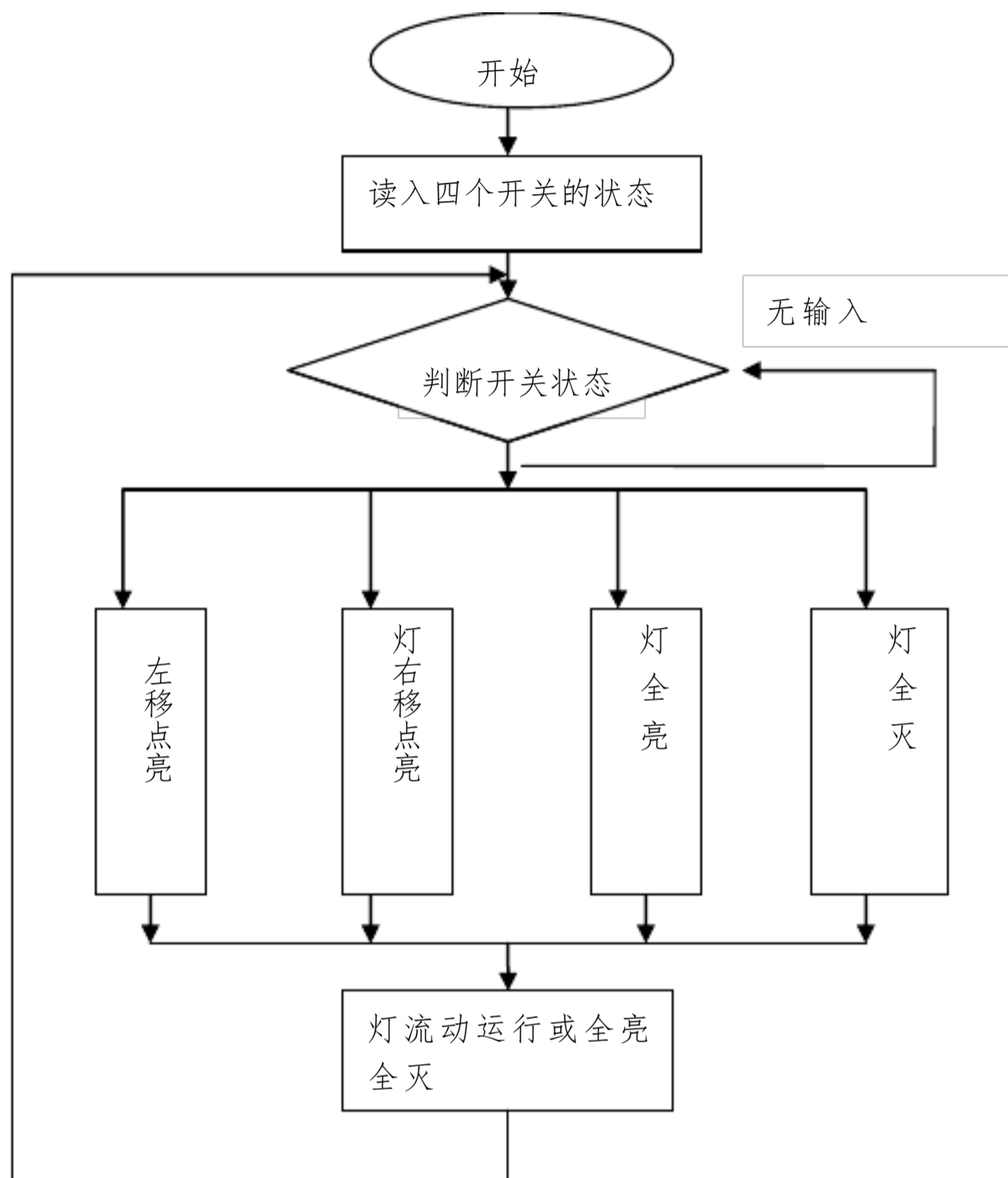
```
ORG0000H LJMPSTARTORG040H START MOVSP,  
#5FH;设堆栈 LOOP NOPLJMPLOOP 循环 END
```

采用单片机定时器产生PWM波，然后控制LED灯的亮度。首先对定0赋初值，使之中断（定时）5ms，再让其中断10次（次数可设定，只是输出波占空比变化快慢不同而已），再定义两个变量cc=0，ee=10H（不同的值会有不同的周期，即PWM波的周期，周期不能太

， $ee--$ ，当 $cc=10H$ ， $ee=0$ 时再 $cc--$ ， $ee++$ 。在主程序中令 P2.0 口当 cc 从 0 到 10H 的时间为高电平，而 ee 从 10H 到 0 的时间为低电平，由于定时器的中断， cc 不断增加， ee 不断减少，则高电平时间越来越长，低电平时间越来越短（总周期不变）。

为了反映出定时器中断对脉冲变化的影响，在 P1.0 口接一按键，如图 5-1，按住时 P1.0 低电平，中断次数 25 次，125ms 脉冲变化一次，可看到波形占空比变化较慢，按键松开时 P1.0 高电平，中断 10 次，则波形占空比变化较快。两个三极管有驱动作用，电感和电容起滤波作用，保证 LED 不至于太过闪烁而看不清亮度变化。

4、系统框图



3. 5、程序框图

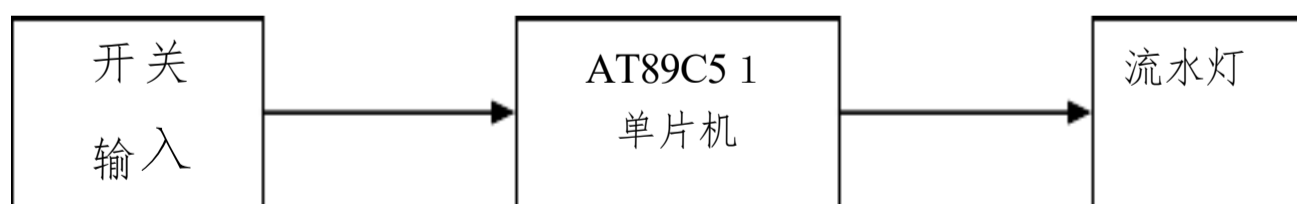


图 2

四、设计步骤

4.1 硬件设计

硬件系统是指构成微机系统的实体和装置，通常由运算器、控制器、存储器、输入接口电路和输入设备、输出接口电路和输出设备等组成。单片机实质上是一个硬件的芯片，在实际应用中，通常很难直接和被控对象进行电气连接，必须外加各种扩展接口电路、外部设备、被控对象等硬件和软件，才能构成一个单片机应用系统。本设计选用以 AT89S51单片机为主控单元。显示部分：8个 LED灯循环亮灭。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/887156105054006115>