



杭州电子科技大学  
HANGZHOU DIANZI UNIVERSITY

## 《单片机原理及应用》作业报告

### 实验报告 5 第三部分：自定义波形生成

学院 卓越学院

学号 23040447

姓名 陈文轩

专业 智能硬件与系统(电子信息工程)

2025 年 5 月 30 日

原题目：DAC0832 与单片机的接线如课堂上所接，参考电压为-2V，请编程实现波形从 +1V 开始增大到 +2V，再下降到 0V，不断循环，周期为 20+ 作业号，单位是 ms。

## 1 实验代码

Code Listing 1: 实验程序

```
1
2 #include <reg51.h>
3 #define DATA P2
4 #define STUDENT_ID 47 // 学号定义为47
5
6 // 波形输出总时间 = (学号 + 20)ms
7 #define TOTAL_TIME (STUDENT_ID/1.375 + 20) // 单位: ms
8 unsigned char mode = 0; // 0:自定义波形, 1:锯齿波, 2:三角波
9
10 // 自定义波形表 - 从+1V到+2V再到0V
11 // 0V对应值为0, 1V对应值为128, 2V对应值为255
12 unsigned char code customWave[256] = {
13     // 0-15: 起始点(+1V)到上升过程
14     128, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 172, 176, 180,
15     184, 188,
16     // 16-31: 继续上升
17     192, 196, 200, 204, 208, 212, 216, 220, 224, 228, 232, 236, 240, 244,
18     248, 252,
19     // 32-47: 到达峰值(+2V)并开始下降
20     255, 255, 250, 245, 240, 235, 230, 225, 220, 215, 210, 205, 200, 195,
21     190, 185,
22     // 48-63: 继续下降
23     180, 175, 170, 165, 160, 155, 150, 145, 140, 135, 130, 125, 120, 115,
24     110, 105,
25     // 64-79: 继续下降
26     100, 95, 90, 85, 80, 75, 70, 65, 60, 55, 50, 45, 40, 35, 30, 25,
27     // 80-95: 接近0V
28     20, 15, 10, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
29     // 96-111: 保持0V一段时间
30     0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
31     // 112-127: 开始回升
32     5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80,
33     // 128-143: 继续回升
```



```
65     EA = 1;           // 开总中断
66     ETO = 1;          // 开定时器0中断
67     TRO = 1;          // 启动定时器0
68 }
69
70 // 设置定时器初值, 使每个点的延时为 TOTAL_TIME/256 ms
71 void setTimer0()
72 {
73     unsigned int delay_us = (TOTAL_TIME * 1000UL) / 256; // 每个点的延时
74     // (微秒)
75     unsigned int reload_value;
76
77     // 假设使用12MHz晶振, 每个机器周期为1us
78     reload_value = 65536 - delay_us;
79     TH0 = (unsigned char)(reload_value >> 8);
80     TL0 = (unsigned char)reload_value;
81 }
82 // 定时器0中断服务函数
83 void Timer0_ISR() interrupt 1
84 {
85     timer_count = 1; // 设置标志, 表示延时完成
86     setTimer0();     // 重新设置定时器初值
87 }
88
89 void main()
90 {
91     unsigned char i;
92     unsigned char waveData;
93
94     Timer0Init(); // 初始化定时器
95     setTimer0();  // 设置定时器初值
96
97     while(1)
98     {
99         for(i=0; i<256; i++)
100         {
101             // 根据mode选择波形
102             switch(mode)
103             {
104                 case 0: // 自定义波形 (使用预先计算好的波形表)
105                     waveData = customWave[i];
106                     break;
```

```
107         case 1: // 锯齿波
108             waveData = getSawWave(i);
109             break;
110         case 2: // 三角波
111             waveData = getTriangleWave(i);
112             break;
113         default:
114             waveData = customWave[i]; // 默认使用自定义波形
115             break;
116     }
117     DATA = waveData;
118
119     // 等待定时器中断发生（延时结束）
120     timer_count = 0;
121     while(!timer_count);
122 }
123 }
124 }
```

## 2 实验效果

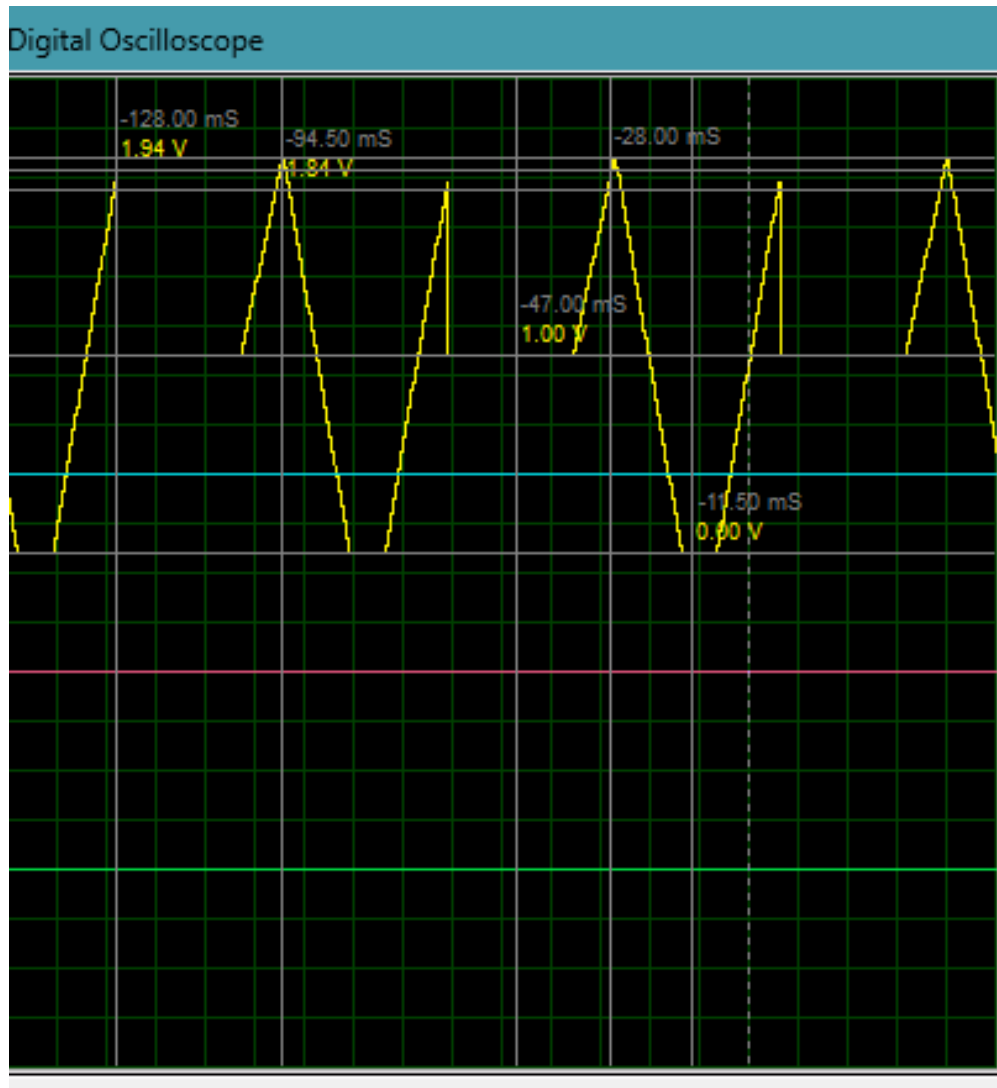


图 1 代码控制效果

示波器光标测量得：周期为 67ms，符合要求；DC 挡位测量电压变化为 0 至 2V，符合实验预期。

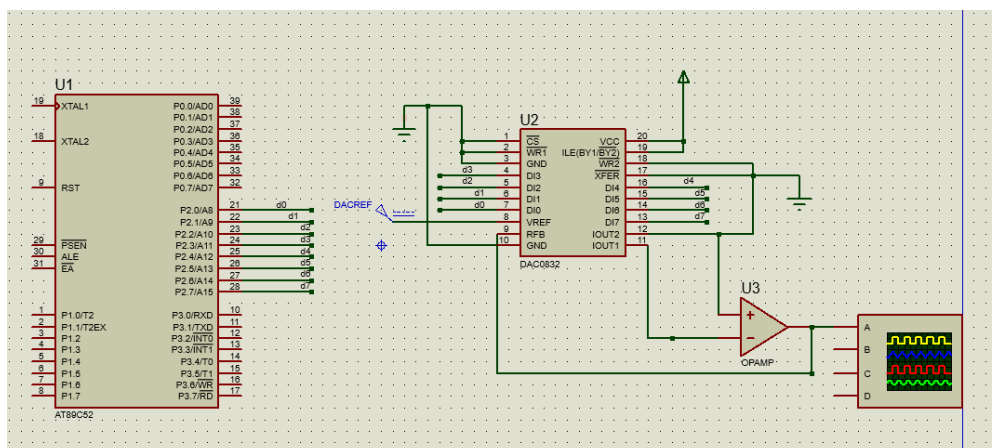


图 2 电路结构

### 3 流程图

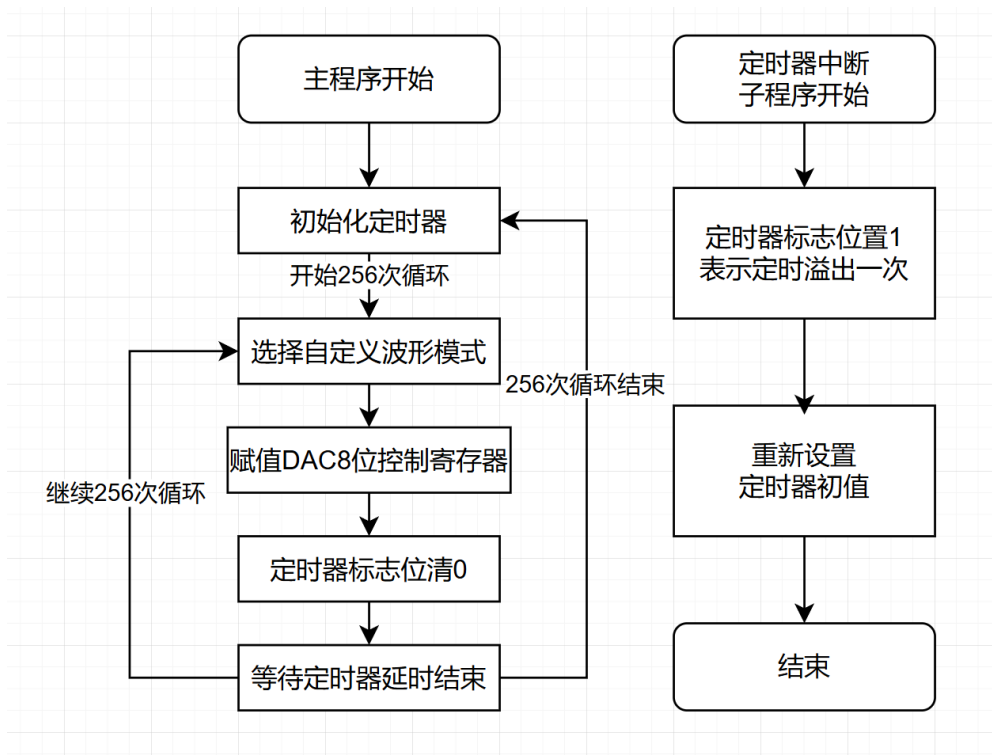


图 3 系统控制流程图

### 4 实验体会

通过本次自定义波形生成实验，我获得了以下几点体会与收获：

1. **DAC 原理与应用的初步理解：**本实验使用 DAC0832 数模转换器与单片机配合工作，加深了我对 DAC 工作原理的理解。通过实践发现，数字量到模拟量的转换过程需要精确的时序控制和值映射，这对应用中实现准确的电压输出至关重要。
2. **波形设计与实现：**通过查表法实现波形输出是一种高效的方法。在设计自定义波形表时，需要精确计算电压与数字量之间的对应关系，例如将 0-2V 的电压范围映射到 0-255 的数字量范围。这种映射关系的掌握对于后续更复杂波形的生成有重要意义。
3. **定时控制的重要性：**波形生成中，时间控制直接影响波形的周期特性。本实验要求实现 (20+ 学号)ms 的周期波形，这需要精确计算每个点的输出时间。通过定时器中断方式实现精确延时，比简单的软件延时更为可靠。
4. **硬件连接的规范性：**DAC0832 与单片机的接线需要严格遵循规范，任何连接错误都会导致输出波形异常。这让我认识到在实际工程应用中，硬件接口的正确连接是系统稳定运行的基础。

总的来说，这次实验将理论知识与实际操作紧密结合，帮助我建立了从数字系统到模拟世界的桥梁。通过自己动手设计并实现特定要求的波形，不仅加深了对单片机与外设协同工作的理解，也提高了解决实际问题的能力。