

单片机原理及应用

**实验指导书**

适用课程\_\_\_单片机原理及应用

适用专业\_ 电子信息工程

广州航海学院 通信系

闫瑞瑞 编写

2018年8月20日

目 录

[实验一 IO口应用 3](#_Toc33884056)

[实验二 定时器/计数器应用 5](#_Toc33884057)

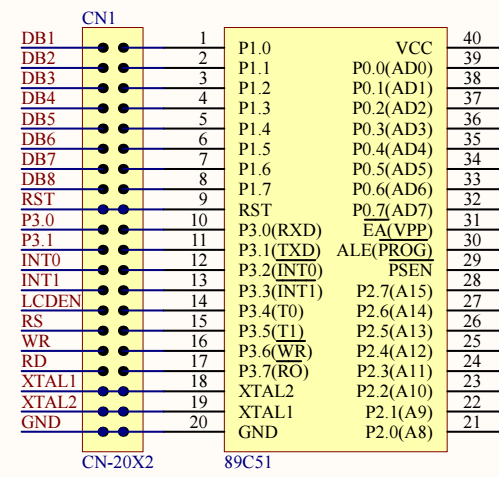
[实验三 串口数据收发 8](#_Toc33884058)

[实验四 D/A应用—信号发生器 11](#_Toc33884059)

[实验五 A/D应用-简易电压表 15](#_Toc33884060)

# 实验一 IO口应用

**------用独立按键模拟电子琴键盘发声**

1. **实验目的**
2. 熟悉Keil软件的使用；
3. 熟悉独立按键状态识别技巧；
4. 熟悉蜂鸣器驱动电路，及蜂鸣器发声频率控制方法。
5. **实验内容及原理**
6. 独立按键

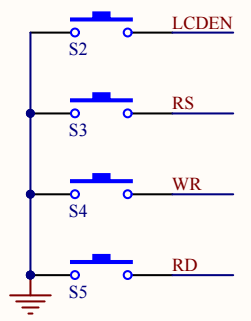
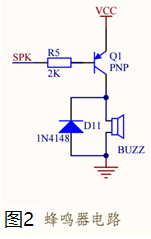


图1 独立按键硬件结构

开发板四个独立按键与单片机STP89C52连接关系如图1。可见，当S2按下时，P3.4引脚为低电平，否则为高电平；S3按下时P3.5为低电平；S4按下时P3.6为低电平；S5按下时P3.7为低电平。

1. 蜂鸣器

开发板电路使用了单片机的I/0口P2.3来控制蜂鸣器。但并没有直接用P2.3控制蜂鸣器，而是通过PNP三极管间接控制。这是因为单片机的I/0口的输入输出电流有限，不能满足蜂鸣器鸣响所需要的电流。而该三极管最大可以提供1A以上的电流，足以驱动蜂鸣器，所以使用P2.3控制PNP三极管的导通和截止，从而达到控制蜂鸣器的目的。

向P2.3写出逻辑“1”时，P2.3输出+5V，三极管的基极电流为0，三极管处于截止状态，电源+5V不能加到蜂鸣器的正极，蜂鸣器不鸣响；向P2.3写出逻辑“0”时，P2.3输出0V，三极管导通，图2中限流电阻R5取合适的值，可以使三极管处于饱和导通状态，电源+5V通过三极管的发射极和集电极加到蜂鸣器的正极，有电流流过蜂鸣器的正极和负极，有源蜂鸣器就开始鸣响。

1. 电子琴键盘音阶频率

电子琴部分音阶与频率对应关系如表1所示。当单片机P2.3输出相应频率的方波给蜂鸣器时，蜂鸣器便发出对应频率的音阶。

表一 电子琴键盘音阶频率对照表（频率单位：Hz）

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 音阶 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 频率 | 261.632 | 293.672 | 329.636 | 349.237 | 392.005 | 440.010 | 493.895 |

1. **实验步骤**
2. 使用Keil编译软件，建立文件夹用来保存当前工程，文件夹以班级学号姓名来命名；新建C51工程文件\*.uvproj并保存在当前文件夹中；建立新的\*.c文件保存并将其加入到当前工程中。单片机芯片选择AT89C52。
3. 在\*.c的文件中编写程序，实现以下功能：

按下独立按键S2时，使蜂鸣器发出“do”的音；按下S3发出“re”的音；按下S4发出“mi”的音，按下S5发出“fa”的音。

独立按键优先级排列顺序依次为：S2🡪S3🡪S4🡪S5

1. 编译链接工程文件，生成\*.Hex文件；
2. 设置晶振频率为11.0125MHz。调试程序，用Keil软件中逻辑分析仪验证频率是否正确。
3. 下载\*.hex文件到单片机开发板，依次按下S2~S4，验证蜂鸣器发声频率正确。
4. **实验源代码**

**五、实验总结**

# 实验二 定时器/计数器应用

**------用定时器实现电子秒表**

1. **实验目的**
2. 熟悉单片机内部定时器的使用和编程方法；
3. 熟悉定时器初值及定时器的计算；
4. 掌握中断处理程序的编程方法。
5. **实验内容及原理**

1、定时器/计数器组成原理图

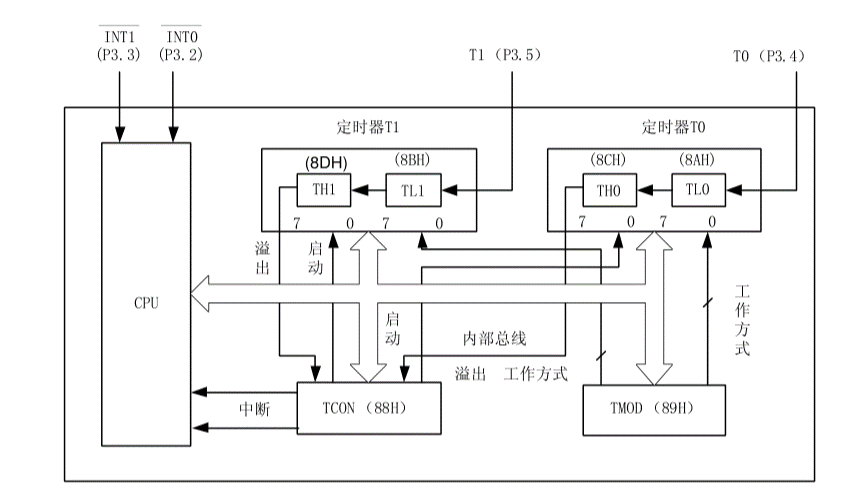


图1 定时器/计数器内部结构框图

51单片机内部有两个16位定时器/计数器，简称为T0和T1，可以独立使用。其中，TH1(高八位)、TL1（低八位）是T1的计数器，TH0(高八位)、TL0（低八位）是T0的计数器.他们的工作状态及工作方式由两个特殊功能寄存器TMOD和TCON的各位的状态或设置来决定。

2、定时器/计数器的控制

1）定时器/计数器工作模式寄存器TMOD（89H）

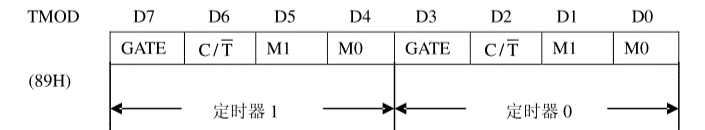


图2 TMOD的各位定义

2）定时器/计数器控制寄存器TCON（88H）

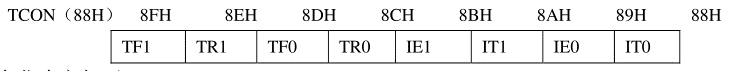


图3 TCON的各位定义

3、定时器/计数器的初始化

1）确定工作方式：对TMOD赋值。

2）预置定时/计数的初值。

4、定时器延时可以使用查询方式实现，也可以使用中断方式实现。与串口中断有关的中断允许控制寄存器IE：

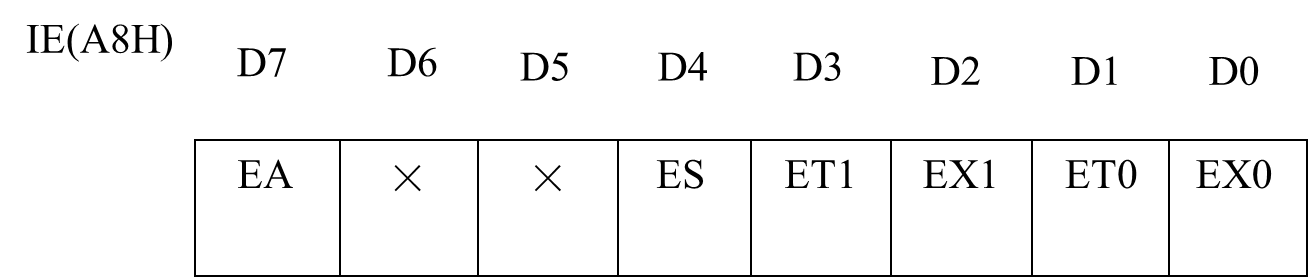


图4 中断允许寄存器IE的格式

1. **实验要求**

1、定时器T0产生周期为1s的方波，送P1.0控制LED闪烁；

2、两位十进制计数每隔1s加1，在00~59间循环；

3、计数结果送两位数码管显示；

4、按钮用作意外情况紧急报警。按钮接外部中断0，当按下按钮时，数码管闪烁显示“E0”，同时峰鸣器发出报警声，各五次，然后退出报警，继续正常1s计数。

1. **实验源代码**
2. **实验总结**

# 实验三 串口数据收发

**UART串口收发双向通信**

1. **实验目的**
2. 熟悉89C52全双工异步串行通信方式；
3. 熟悉SCON、PCON设置方法及其对串口的控制作用；
4. 熟悉串口初始化程序、及收发程序的写法。
5. 掌握串口调试助手的使用方法。
6. **实验内容及原理**
7. USB转串口模块

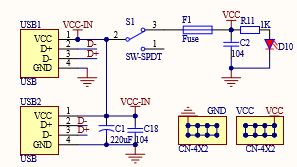
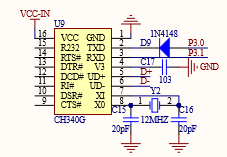
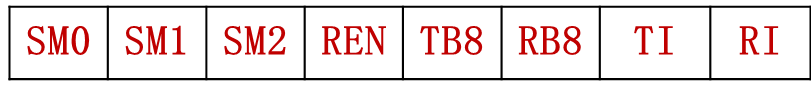


图1 USB转串口电路

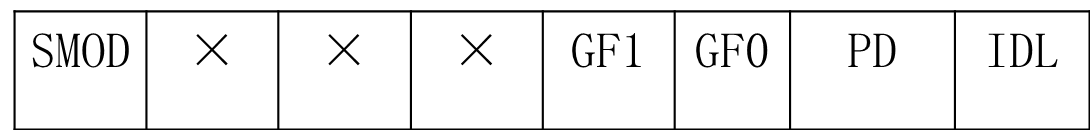
USB接口数据通过CH340G转换为TTL连接至单片机异步串行口P3.0（RXD）和P3.1（TXD）

1. **与串口通信有关的SFR**

**SCON**（98H）

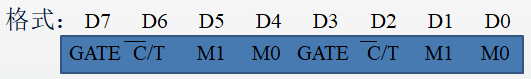
****

**PCON**（87H）

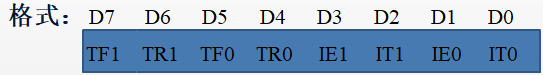


**3、T1方式2常用于串口波特率发生器，与T1有关的SFR**

**TMOD**（89H）



**TCON**（88H）

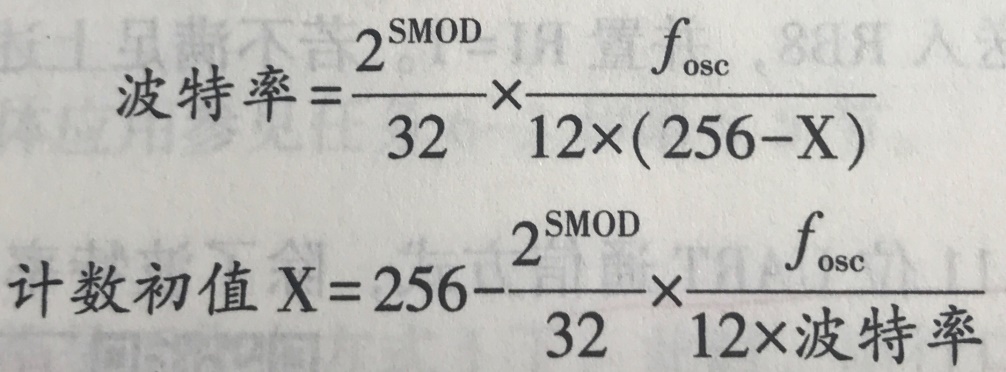


**4、**主频fosc、波特率、计数器初值N 之间的关系：

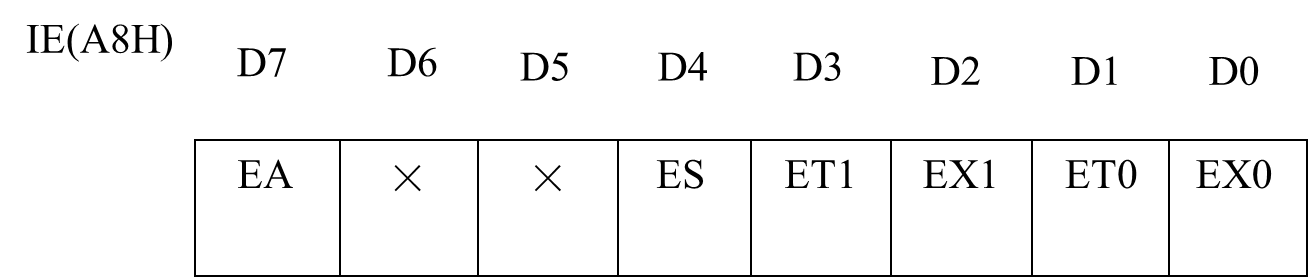
* + - 波特率=2SMOD×T1溢出率/32
    - T1溢出率=fosc/[12\*(2n-N)]

**所以，T1工作在方式2时：**

* **28-N=2SMOD\*fosc/(32\*12)**

****

**5、**串口通信可以工作在查询方式，也可以工作在中断方式。与串口中断有关的中断允许控制寄存器IE：



1. **实验要求**
2. 编写程序实现以下功能：
3. 单片机与计算机之间串行异步通信，波特率9600bps。串口工作方式3，偶校验。定时器1工作方式2，作为串口波特率发生器。
4. 发送字符串：每按下一次开发板独立按键S2键，松开后均会在串口调试助手接收区域显示自己的姓名学号；
5. 能每次接收一字节串口助手发送的数据(16进制)，接收后每按下一次开发板独立按键S3键，松开后将收到的数据用LED灯或两位数码管显示。

**2、**用USB转串口方式连接计算机与单片机开发板；

**3、**调试、编译并下载程序，用串口助手调试并验证程序执行结果。

1. **实验步骤**
2. 根据波特率、单片机时钟频率，及工作方式，计算SMOD及T1初值N：

SMOD=

N=

1. **设置串口波特率步骤：**
2. 写TMOD，设置定时器T1的工作方式；
3. TH1、TL1赋值，设置T1初值；
4. TR=1，启动T1，即启动波特率发生器。
5. **串口收/发过程：**
6. 串口初始化：设置串口工作方式（帧格式）、设置T1工作方式、设置波特率（传输速率）、启动波特率发生器。

若以中断方式传输，则开ES和EA，若以查询方式传输无需开中断。

初始化程序如下：

TMOD=0x02； //T1工作在方式2

TH1=0xFD; //T1初值

TL1=0xFD;

TR1=0; //启动T1

PCON=0x00; //SMOD=0，波特率不翻倍

SCON=0xd0; //串口工作在方式3，允许接收

1. 发送数据/接收数据，即读写SBUF；
2. 判断一帧是否发送完毕，即判断TI是否为1。接收时检测RI标志；
3. TI/RI清零；
4. 跳转到（2），收/发下一帧数据。
5. 编写流程图及实验程序。

**（流程图附下图，实验程序以电子文档提交）；**

3、串行方式连接单片机与计算机；

4、下载程序

5、在串口助手设置：打开串口、设置相应的串口号( )、波特率(9600)、校验方式(偶校验)、停止位(1)。

6、按下松开开发板S2，在串口助手接收区以字符方式接收并验证收到的数据。

7、在串口助手发送区以16进制数形式发送1字节，如55H、80H、01H等，然后按下并松开S3，观察开发板8位LED是否为对应的数据。

**五、实验结果**

（截图贴于下方）

# 实验四 D/A应用—信号发生器

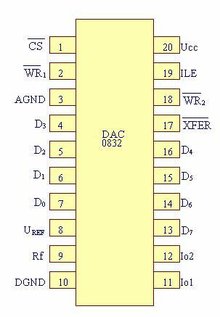
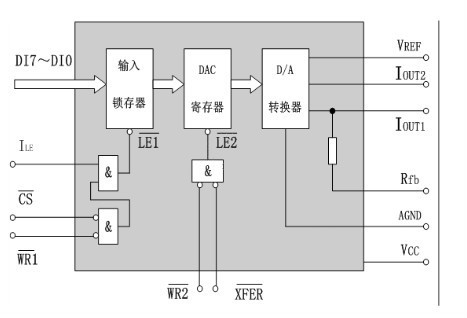
1. **实验目的**
2. 熟悉DAC0832的工作原理；
3. 熟悉 DAC0830的控制代码书写；
4. 利用单片机的D/A转换功能进行简单的设计。
5. **实验内容及原理**
   1. **DAC0832引脚图和工作原理图**

图2.DAC0832工作原理图

图1.DAC0832引脚图

* 1. **DAC0832简要介绍**

DAC0832是8位D-A转换器，它采用先进的COMS工艺制造，采用单片双列直插式封装。DAC0832有三种工作方式:**直通方式**、**单缓冲方式**和**双缓冲方式**。

**（1）单缓冲方式**。单缓冲方式是控制输入寄存器和DAC寄存器同时接收资料，或者只用输入寄存器而把DAC寄存器接成直通方式。此方式适用只有一路模拟量输出或几路模拟量异步输出的情形。

**（2）双缓冲方式。**双缓冲方式是先使输入寄存器接收资料，再控制输入寄存器的输出资料到DAC寄存器，即分两次锁存输入资料。此方式适用于多个D/A转换同步输出的情节。

**（3）直通方式。**直通方式是资料不经两级锁存器锁存，即 CS\*，XFER\* ，WR1\* ，WR2\* 均接地，ILE接高电平。此方式适用于连续反馈控制线路和不带微机的控制系统，不过在使用时，必须通过另加I/O接口与CPU连接，以匹配CPU与D/A转换。

知识拓展（了解不用抄）

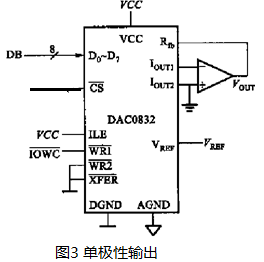
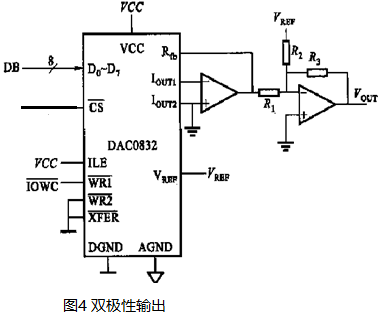
①单极性输出

[单极性电压输出电路](https://baike.so.com/gallery/list?ghid=first&pic_idx=7&eid=5414862&sid=5653004)如图3所示, 由运算放大器进行电流→电压转换，使用内部反馈电阻。输出电压值VOUT和输入数字量D的关系:

VOUT = - VREF ×D/256

D = 0~255， VOUT = 0 ~ - VREF ×255/256

VREF = -5V， VOUT =0~5×(255/256)V

VREF = +5V， VOUT = 0 ~ -5×(255/256)V

②双极性输出

如果实际应用系统中要求输出模拟电压为双极性，则需要用转换电路实现。如右图4所示[双极性电压输出电路](https://p1.ssl.qhimg.com/t01a16a6c840c09aebf.png).

其中 R2=R3=2R1

VOUT= 2×VREF×D/256 -VREF= (2D/256-1)VREF

D = 0时， VOUT= -VREF;

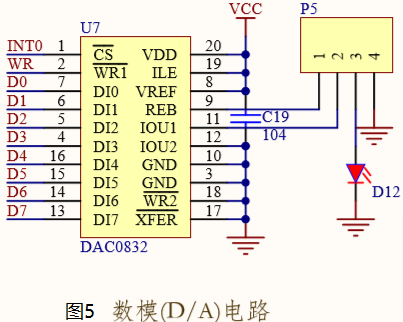
D = 128时， VOUT= 0;

D = 255时，

VOUT= (2×255/256-1)×VREF= (127/128)VREF

即:输入数字为0~255时，输出电压在- VREF ~+ VREF之间变化。

1. **开发板实验电路**



开发板DAC0832电路如图5，图中已连接WR2、XFER到地，已连接ILE到高电平，因此该电路为单缓冲方式，控制信号仅为一级缓冲控制信号CS和WR1。DAC08432为电流输出型器件，可将P5引脚用短接片短接后，使电流流过发光二极管 D12，可通过发光二极管D12的发光强度观察输出电流的变化。

实际应用中，应外接运放将其输出变为单极性或双极性电压输出型，见图3、图4。

1. **实验要求**

1、读懂实验硬件电路图，熟悉DAC0832工作原理

1. 编写程序实现以下功能：

运用STM89C52和DAC0832产生正弦波，锯齿波，三角波；

当按下并弹起一个按键时，输出正弦波，第二次按下并弹起该键输出锯齿波，第三次按下并弹起该键输出三角波。如此，依次循环切换；；

1. 调试，用嵌入式逻辑分析仪观察波形是否正确；
2. 编译并下载程序，用示波器观察DAC0832输出，验证程序执行结果。

**四、实验步骤（参考）**

* 1. 实验电路图单极性输出如图3，双极性输出如图4；
  2. 上机实验，设计流程图；
  3. 编写程序建立不同波形的子函数；
  4. 书写主函数，初始化；
  5. 调用子函数；

**五、实验结果**

（拍摄不同波形的示波器显示图片并张贴在下方，抄写程序代码）

# 实验五 A/D应用-简易电压表

**一、实验目的**

1、熟悉模拟电压转化成数字的工作原理；

2、熟悉LCD1602的常用指令和编程方法；

3、熟悉ADC0804的使用和程序编写方法。

**二、实验内容及原理**

1、ADC0804芯片

ADC0804是一个8位CMOS型逐次比较式A/D转换器，具有三态锁存输出功能，最短转换时间为100us，其引脚图如下：

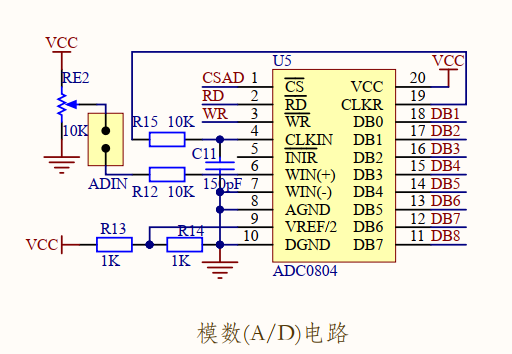


图1 ADC8040引脚图

① 芯片参数：

工作电压：+5V，即VCC=+5V。

模拟转换电压范围：0～+5V，即0≤Vin≤+5V。

分辨率：8位，即分辨率为1/28=1/256，转换值介于0～255之间。

转换时间：100us（fCK = 640KHz时）。

转换误差：±1LSB。

1/2参考电压：2.5V，即Vref = 5V。

② 各个引脚名称及作用：

Vin（+）、Vin（-）：两个模拟信号输入端，可以接收单极性、双极性和差模输入信号。

DB0-DB7：具有三态特性数字信号输出端，输出结果为八位二进制结果。

CLKIN：时钟信号输入端。

CLKR:内部时钟发生器的外接电阻端，与CLK端配合可由芯片自身产生时钟脉冲，其频率计算方式是：fck=1/(1.1RC）。

CS：片选信号输入端，低电平有效。

WR：写信号输入端，低电平启动AD转换。

RD：读信号输入端，低电平输出端有效。

INTR：转换完毕中断提供端，AD转换结束后，低电平表示本次转换已完成。

VREF/2：参考电平输入，决定量化单位。

VCC：芯片电源5V输入。

AGND:模拟电源地线。

DGND:数字电源地线。

③ ADC0804的转换原理

ADC0804是属于逐次逼近式（Successive Approximation Method）A/D转换器，转换速度快，分辨率高。

2、LCD1602液晶

LCD1602液晶模块是两行16个字符，用5×7点阵图形来显示字符的液晶显示器，属于16字×2行类型。内部具有字符发生器ROM( Character- Generator Rom, CG ROM)，可显示192种字符(160个5×7点阵字符和5×10点阵字符)。具有64B的自定义字符RAM( Character- Generator RAM, CG RAM)，可以定义8个5×8点阵字符或4个5×11点阵字符。具有64B的数据显示RAM(Data- Display RAM, DD RAM)。下图为LCD1602的引脚图。

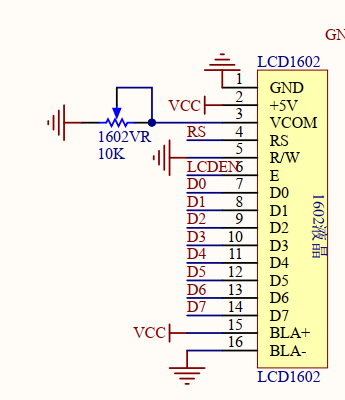


图2 LCD1602液晶引脚

① 模块引脚功能

表1 模块的引脚功能



② LCD1602指令集

表2 LCD1602指令集



**三、实验要求**

1、调节滑动变阻器，对其电压进行采样，采集到的是一个八位的二进制数。

2、LCD1602液晶上的第一行显示采集到的电压值（进制不限）。

3、（选做）LCD1602液晶上的第二行用十进制显示该采样值对应5V的实际电压值，且显示的最小分辨率为0.01V。

**四、实验源代码**

**五、实验结果**

**六、实验小结**