

Python base program

目录 Contents

第一部分  DAC的基本原理

第二部分  DAC的基本用法

第三部分  DAC输出实例

■ DAC的基本原理

- 经数字系统处理后的数字量，有时又要求再转换成模拟量以便实际使用，这种转换称为“数模转换”。
- 完成数模转换的电路称为数模转换器，简称 DAC（Digital to Analog Converter）。
- 数字量是用代码按数位组合起来表示的，对于有权码，每位代码都有一定的位权。为了将数字量转换成模拟量，必须将每1位的代码按其位权的大小转换成相应的模拟量，然后将这些模拟量相加，即可得到与数字量成正比的总模拟量，从而实现了数字—模拟转换。这就是组成D/A转换器的基本指导思想。

■ DAC的基本用法

- `pyb.DAC(port, bits=8)`
构造一个新的DAC对象
port, 1或2, 对应X5 (PA4) /X6 (PA5)
bits, 输出精度, 可以是8或12
- `DAC.init(bits=8)`
重新初始化DAC, bits可以是8或12
- `DAC.deinit()`
对DAC进行初始化, 使其引脚可用于其他用途
- `DAC.noise(freq)`
产生伪随机噪声信号。以给定的频率将新的随机样本写入DAC输出

■ DAC的基本用法

- DAC.triangle(freq)
以指定频率产生三角波
- DAC.write(value)
写入参数
在8bits时, 参数范围[0-255]
在12bits时, 参数范围[0..4095]
- DAC.write_timed(data, freq, *, mode=DAC.NORMAL)
使用DMA方式周期写入数据
data, 缓冲区数组
freq, 默认使用Timer(6), 用指定频率更新, 也可以指定另外的定时器, 有效的定时器是[2, 4, 5, 6, 7, 8]
mode, DAC.NORMAL or DAC.CIRCULAR

DAC输出实例

需要将示波器和开发板连接就可以了。在python开发板上有两个DAC接口，分别对应引脚X5和X6。上文中我们通过`dac = DAC(1)`定义了1接口，所以需要将示波器的一端接X5，一端接地。



代码运行前，示波器显示一条直线，没有波动。



运行代码后，示波器输出正弦波。



DAC输出实例

以12位分辨率输出连续正弦波的代码如下所示，其实验结果与以8位分辨率输出连续正弦波大体相同。

```
import math                #导入math库
from pyb import DAC        #导入DAC库
buf = bytearray(100)       #创建一个大小为100的缓冲区数组
for i in range(len(buf)):
    buf[i] = 128 + int(127 * math.sin(2 * math.pi * i / len(buf))) #给缓冲区数组赋值
dac = DAC(1)               #定义DAC接口1
dac.write_timed(buf, 400 * len(buf), mode=DAC.CIRCULAR)
#使用DMA方式周期写入数据
```


DAC输出实例

打开终端输入`sudo picocom /dev/ttyACM0`，然后CTRL+D运行文件

```
hcy@hcy-VirtualBox: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
hcy@hcy-VirtualBox:~$ sudo picocom /dev/ttyACM0  
[sudo] hcy 的密码:  
picocom v2.2  
  
port is      : /dev/ttyACM0  
flowcontrol  : none  
baudrate is  : 9600  
parity is    : none  
databits are : 8  
stopbits are : 1  
escape is    : C-a  
local echo is : no  
noinit is    : no  
noreset is   : no  
nolock is    : no  
send_cmd is  : SZ -vv  
receive_cmd is : rZ -vv -E  
imap is      :  
omap is      :  
emap is      : crcrlf,delbs,  
  
Type [C-a] [C-h] to see available commands  
Terminal ready
```

```
hcy@hcy-VirtualBox: ~  
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)  
>>>  
PYB: sync filesystems  
PYB: soft reboot  
MicroPython v1.9.4 on 2018-05-11; PYBv1.1 with STM32F405RG  
Type "help()" for more information.  
>>>
```