DAC数模转换

**DAC是数字模拟转换，将数字信号转换为模拟信号（以电流、电压或电荷的形式）。在TPYBoard F407开发板上DAC可输出介于0和3.3V之间的电压。**

**TPYBoard F407开发板上有两个DAC接口，分别是PA4和PA5引脚。**

**程序示例：**

|  |
| --- |
| from pyb import DAC  dac = DAC(1) # create DAC 1 on pin PA4  dac.write(128) # write a value to the DAC (makes X15 1.65V)  dac = DAC(1, bits=12) # use 12 bit resolution  dac.write(4095) # output maximum value, 3.3V |

**输出连续正弦波:**

|  |
| --- |
| **import math**  **from pyb import DAC**  **# create a buffer containing a sine-wave**  **buf = bytearray(100)**  **for i in range(len(buf)):**  **buf[i] = 128 + int(127 \* math.sin(2 \* math.pi \* i / len(buf)))**  **# output the sine-wave at 400Hz**  **dac = DAC(1)**  **dac.write\_timed(buf, 400 \* len(buf), mode=DAC.CIRCULAR)** |

**以12位分辨率输出连续正弦波：**

|  |
| --- |
| **import math**  **from array import array**  **from pyb import DAC**  **# create a buffer containing a sine-wave, using half-word samples**  **buf = array('H', 2048 + int(2047 \* math.sin(2 \* math.pi \* i / 128)) for i in range(128))**  **# output the sine-wave at 400Hz**  **dac = DAC(1, bits=12)**  **dac.write\_timed(buf, 400 \* len(buf), mode=DAC.CIRCULAR)** |

**构造器**

**class pyb.DAC(port, bits=8)**

**构造一个DAC对象。**

* **port 是一个针脚, 或整数(1 或 2)。DAC(1) 是针脚PA4 和 DAC(2) 是针脚PA5。**
* **bits 是指定分辨率的整数，可以是8或12。**

**方法**

**DAC.init(bits=8)**

**重新初始化DAC。 bits 8位或12位。**

**DAC.deinit()**

**初始化DAC使其引脚可用于其他用途。**

**DAC.noise(freq)**

**产生伪随机噪声信号。一个新的随机样本写入DAC输出在给定的频率**

**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 或 DAC.CIRCULAR。**

**同时利用DAC的例子:**

**dac1 = DAC(1)**

**dac2 = DAC(2)**

**dac1.write\_timed(buf1, pyb.Timer(6, freq=100), mode=DAC.CIRCULAR)**

**dac2.write\_timed(buf2, pyb.Timer(7, freq=200), mode=DAC.CIRCULAR)**