



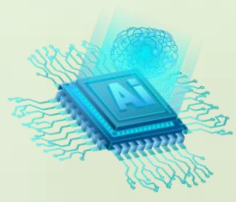
第9章 模数转换与数模转换

9.1 模数转换器ADC的基础知识

9.2 ADC驱动构件及使用方法

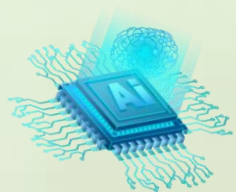
9.3 数模转换 DAC

实验六：ADC模块实验





模数转换ADC是外接模拟量进入MCU的重要一环，本章首先给出模数转换器ADC的基础知识，随后给出ADC驱动构件及汇编语言模式下的使用方法。此外，数模转换DAC也是MCU的基础知识之一，本章将做简要说明。



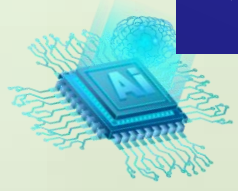


9.1 模数转换器ADC的基础知识

模拟量（Analogue quantity）是指变量在一定范围连续变化的物理量，从数学角度，连续变化可以理解为可取任意值。例如，温度这个物理量，可以有 28.1°C ，也可以有 28.15°C ，还可以有 28.152°C ，……，也就是说，原则上可以有无限多位小数点，这就是模拟量连续之含义。

数字量（Digital Quantity）是分立量，不可连续变化，只能取一些分立值。现实生活中，有许多数字量的例子，如1部手机、2部手机，……，你不能说你买0.12部手机，那它接不了电话！在计算机中，所有信息均使用二进制表示。例如，用一位只能表达0、1两个值，8位可以表达0、1、2、……、254、255，共256个值。

模数转换器（Analog-to-Digital Converter, ADC）是将电信号转换为计算机可以处理的数字量的电子器件，这个电信号可能是由温度、压力等实际物理量经过传感器和相应的变换电路转化而来的。





9.1.1 与A/D转换编程直接相关的基本概念

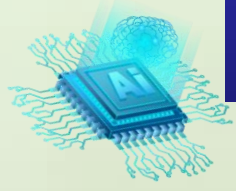
与A/D转换编程直接相关的问题主要有：转换精度、是单端输入还是差分输入、转换速度、A/D参考电压、滤波问题、物理量回归等。

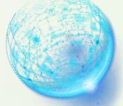
1. 转换精度（重点理解）

转换精度（Conversion accuracy）是指数字量变化一个最小量时对应模拟信号的变化量，也称为分辨率（Resolution），**通常用模数转换器ADC的二进制位数来表征**，通常有8位、10位、12位、16位、24位等，转换后的数字量简称A/D值。

课堂练习：参考电压为3.3V，计算12位的转换精度。

概念：**最低有效位（Least Significant Bit, LSB）、最高有效位（Most Significant Bit, MSB）**





2. 单端输入与差分输入

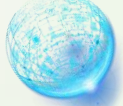
若从微机的一个引脚接入，使用公共地GND作为参考电平，就称为单端输入（Single-ended input）。若从微机的两个引脚接入模拟信号，A/D采样值是两个引脚的电平差值，就称为差分输入（Differential input）。

3. 软件滤波问题

为什么有干扰？软件滤波可以在一定范围内解决干扰问题。
最简单滤波方法：中值滤波、平均值滤波等。

【练习9-2】上网查找一下，有哪些常用的滤波方法？分别适用于什么场景？





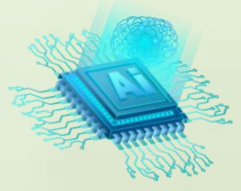
4. 物理量回归问题

在实际应用中，得到稳定的A/D值以后，还需要把A/D值与实际物理量对应起来，这一步称为**物理量回归（Regression）**。

回归方法：线性回归、最小二乘法、神经网络方法等。

物理量回归与仪器仪表“标定（Calibration）”一词的基本内涵是一致的，但不涉及A/D转换概念，只是与标准仪表进行对应，以便使得待标定的仪表准确。而计算机中的物理量回归一词是指计算机获得的A/D采样值，如何与实际物理量值对应起来，也需借助标准仪表，从这个意义上理解，它们的基本内涵一致。

A/D转换物理量回归问题，可以转化为数学上的一元**回归分析（Regression analysis）**问题，也就是一个自变量，一个因变量，寻找它们之间的逻辑关系。设A/D值为 x ，实际物理量为 y ，物理量回归需要寻找它们之间的函数关系： $y=f(x)$ 。许多情况下，这种关系是非线性的，人工神经网络可以较好地应用于这种非线性回归分析中。





9.1.2 与A/D转换编程关联度较弱的基本概念

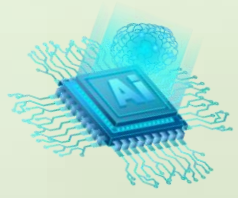
几个A/D转换编程关联度较弱的基本概念，如量化误差、转换速度、A/D参考电压等。

1. 量化误差（重点理解）

在把模拟量转换为数字量过程中，要对模拟量进行采样和量化，使之转换成一定字长的数字量，量化误差（Quantization Error）就是指模拟量量化过程而产生的误差。

通俗地说，最低位无效。

这是理论误差，不可消除。



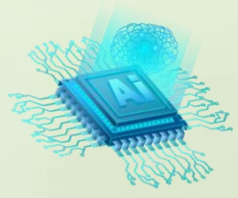


2. 转换速度（了解）

转换速度通常用完成一次A/D转换所要花费的时间来表征。转换速度与A/D转换器的硬件类型及制造工艺等因素密切相关，其特征值为纳秒级。A/D转换器的硬件类型主要有：逐次逼近型、积分型、 Σ - Δ 调制型等。

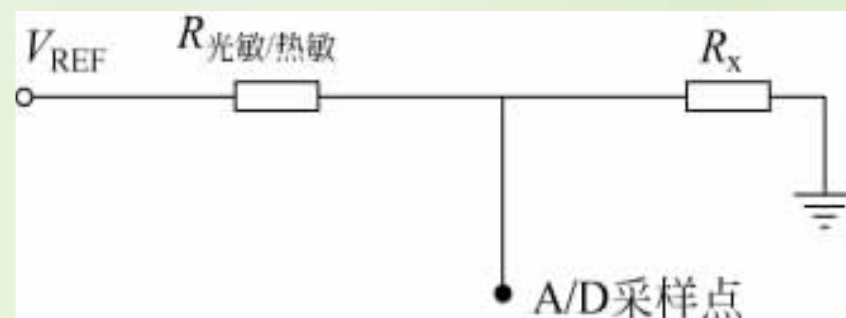
3. A/D参考电压（基本理解）

A/D转换需要一个参考电平。比如要把一个电压分成1024份，每一份的基准必须是稳定的，这个电平来自于基准电压，就是**A/D参考电压**。粗略的情况，A/D参考电压使用给芯片功能供电的电源电压。更为精确的要求，A/D参考电压使用单独电源，要求功率小（在mW级即可），但波动小（例如0.1%），一般电源电压达不到这个精度，否则成本太高。

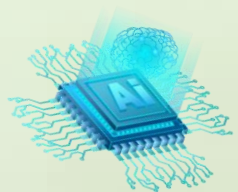




9.1.3 最简单的A/D转换采样电路举例



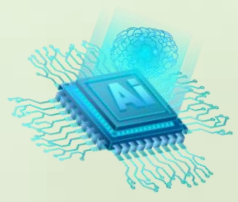
$$V_{A/D} = \frac{R_x}{R_{\text{光敏}} + R_x} \times V_{REF}$$





9.2 ADC驱动构件及使用方法

所有复杂的任务必须逐级简化分解成简单任务，最后达到每个人面对的均是简单任务，但组织起来就是一个复杂任务。ADC驱动构件作为一个复杂工程中一部分，既有其独立性，又有其关联性。在微机原理的学习中，通过学习如何使用各个驱动构件以及各个构件之间协同工作，能更加深刻地理解其原理。



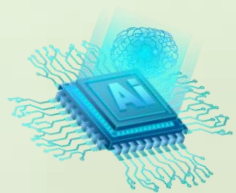


9.2.1 ADC驱动构件要素分析

1. CH573芯片的ADC引脚

表9-1 CH573芯片ADC模块通道引脚表

| 通道号 | 宏定义 | MCU引脚名 | AHL-CH573引脚号 |
|-------|-------------------------------|----------------|--------------|
| 0~1 | ADC_CHANNEL_0~ADC_CHANNEL_1 | PA4~PA5 | 14、13 |
| 2~5 | ADC_CHANNEL_2~ADC_CHANNEL_5 | PA12~PA15 | 7、8、10、9 |
| 8~9 | ADC_CHANNEL_8~ADC_CHANNEL_9 | PB0、PB6 | 未引出 |
| 12~13 | ADC_CHANNEL_12~ADC_CHANNEL_13 | PA8、PA9 | 用户串口使用 |
| | ADC_CHANNEL_VREFINT | 内部电压监测 | |
| | ADC_CHANNEL_TEMPSENSOR | 内部温度检测 | |
| | ADC_CHANNEL_DIFF_0 | 差分1（通道0与通道2组成） | 14、7 |
| | ADC_CHANNEL_DIFF_1 | 差分2（通道1与通道3组成） | 13、8 |





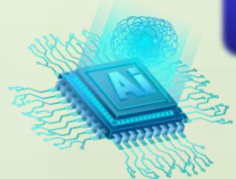
2. ADC模块寄存器概述（略）

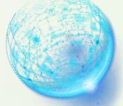
3. A/D转换构件要素（重点掌握）

表9-3 A/D转换构件要素

| 序号 | 函数 | | | 形参 | | 宏常数 | 备注 |
|----|---|----------|----------|---------|-----|-----|----|
| | 简明功能 | 返回 | 函数名 | 英文名 | 中文名 | | |
| 1 | 初始化 | 无 | adc_init | Channel | 通道号 | 用 | |
| | | | | NC | 未使用 | 用 | |
| 2 | 读一次 A/D 值 | uint16_t | adc_read | Channel | 通道号 | 用 | |
| 说明 | <p>（1）初始化的说明：通道号一般与引脚名对应，若是内部物理量，则无引脚号对应，实参应该是物理量名称，在 User 头文件中宏定义与引脚或内部通道联系起来，才能满足可移植性；A/D 转换精度（即位 数）直接在初始化时给出最大值，不作为函数参数，才能保证应用层程序可移植性。每个通道单独初始化，不论芯片结构如何，要满足该要求；若芯片有硬件滤波直接使用最大值。</p> <p>（2）adc_read 的说明：adc_read 的返回值使用 uint16_t，实际的有效值仅仅只有低 12 位。</p> | | | | | | |

4~6 参见源代码





9.2.2 ADC驱动构件使用方法

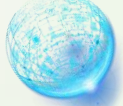
1. 初始化

```
li a0, (通道号, 使用宏常数)    // a0←通道号
li a1, 0                        // a1 ←0, 本参数未使用
call adc_init                   //调用AD转换初始化函数
```

2. 读取AD采集值 （默认函数返回值在a0中）

```
li a0, (通道号, 使用宏常数)    // a0←通道号
call adc_read                   //调用AD转换读出函数
```





3. printf输出AD值

(1) printf格式变量定义处

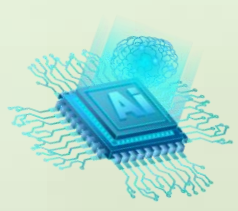
```
printf_format2:           // 定义一个printf使用的数据格式控制符  
    .ascii "%d\n\0"       // 一个参数，十进制
```

(2) AD值读出语句之后

```
mv a1,a0                  //把a0存到a1中，作为printf的第2个参数
```

(3) 调用printf，输出AD值

```
la a0, printf_format2    //printf的第1个参数是a0  
call printf
```





9.2.3 ADC驱动构件使用实例

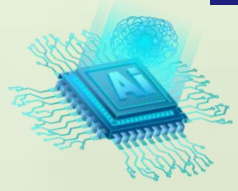
实例运行，课堂练习

9.3 数模转换 DAC

（了解，掌握基本概念）

数/模转换器（Digital-to-Analog Converter, DAC），是利用程序把微型计算机内部的数字量转化为从微型计算机引脚以电压形式输出的模拟量。

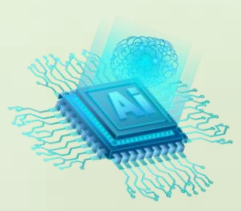
本书的样例芯片CH573内部没有DAC模块，但是可以通过PWM模拟实现DAC的效果。





实验六：ADC模块实验

在学院网站上传实验报告及实验程序，注意注释规范。





本章作业： 1~6

