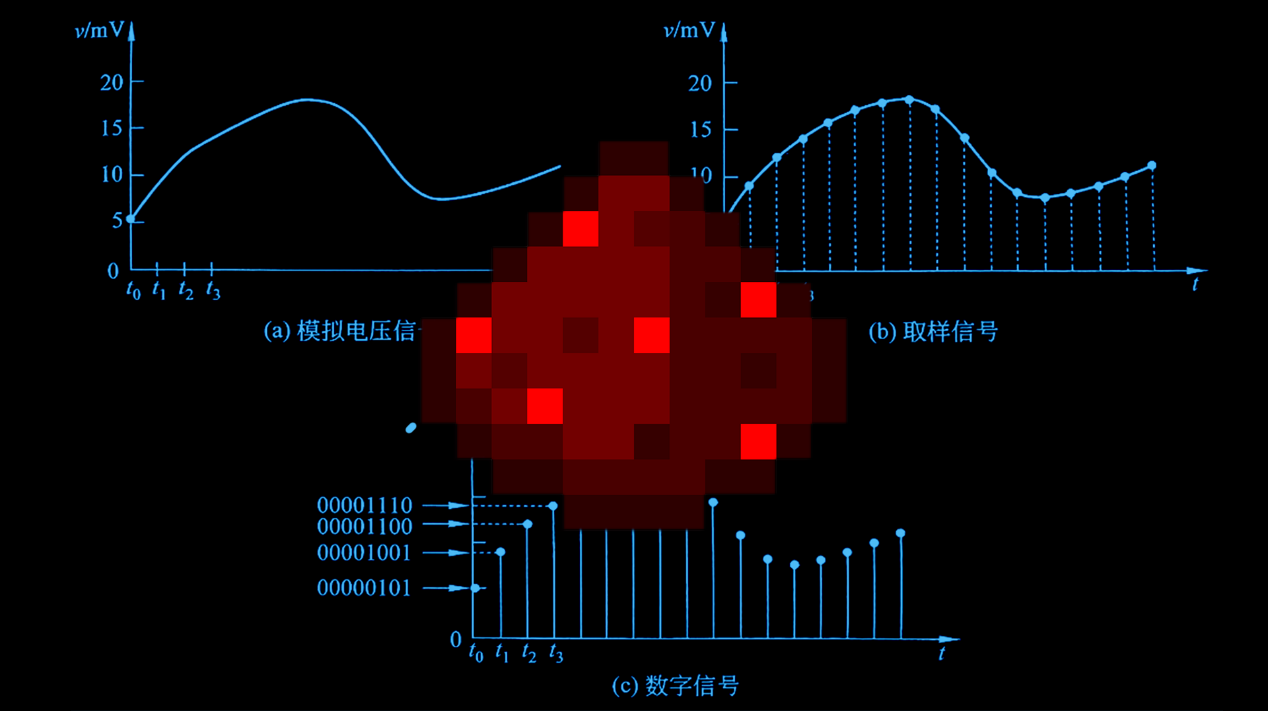
（一）：进制转换及数字信号&模拟信号

【Note】

此文章中，不同形式的字体表示了不同的含义：

正文：正常内容，如果您没有相关基础，请正常阅读。

下划线：不重要的内容，可选择性阅读，不理解的话可以直接跳过。



## 模拟信号与数字信号

**模拟信号**

简单来说，模拟信号是一个“连续”的量，即在时间上或者幅值上连续变化的物理量。如声音信号、一天内的温度变化、某地点海平面的潮汐变化等。这些变化在时间上都是不会中断的（也就是说，不会有什么时候“没有温度”或者“没有海平面”什么的，它们在任意时间点都一定有一个值），且数值都是连续的（比如海平面不可能前一刻还很高后一刻瞬间就降低，它的值在中间必定是连续变化的）

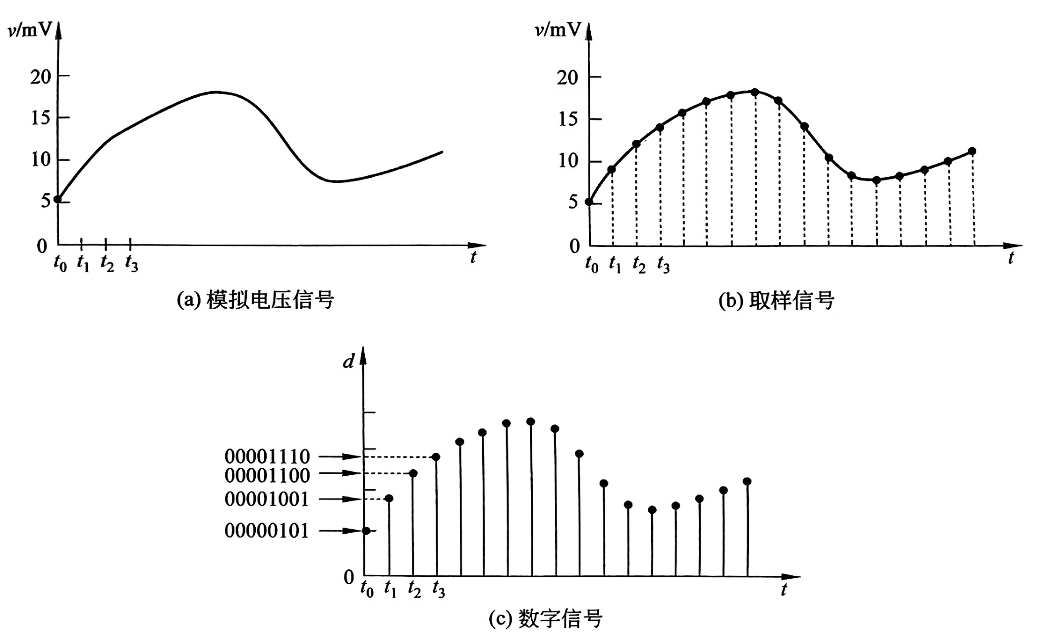
**数字信号**

与模拟信号相反，数字信号在时间和数值上都是离散的，不会像模拟信号那样连续。比如说每天12:00的时候一个人拥有多少块饼干，因为这里以“一天”作为单位进行衡量，因此时间上不是连续的。饼干的数量也是，不会像温度那样连续出现（温度从32度升到33度肯定会经过中间的一系列小数值），而必定为整数值，因此是离散的（没人会拥有2.6、1.25或者11.4514块饼干）

**模拟量的数字表示**

现在计算机都是使用的数字信号。因此，为了便于处理和分析，很多时候都会先将模拟信号转换为数字信号再进行处理

下图所示为模拟信号转为数字信号的过程，图(a)为模拟信号示例。对图(a)中模拟信号进行取样变为时间离散幅值连续的数字信号，如图(b)。*t0*、*t1*、*t2*、*t3*为取样时间点，在该时间点获取信号的幅值。随后，如图(c)所示，将该幅值进行量化（即数字化）取整，并变为时间离散幅值也离散的数字量。



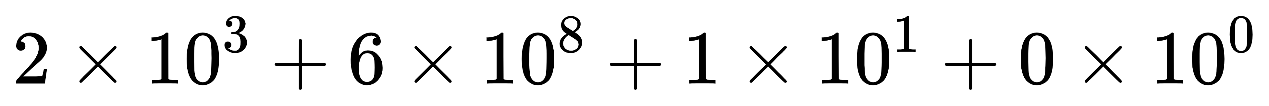
图片来源：清华大学出版社《数字电路与逻辑设计》

## 数制

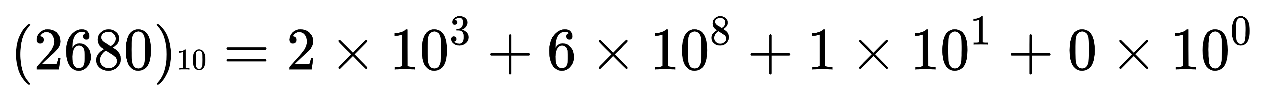
**十进制**

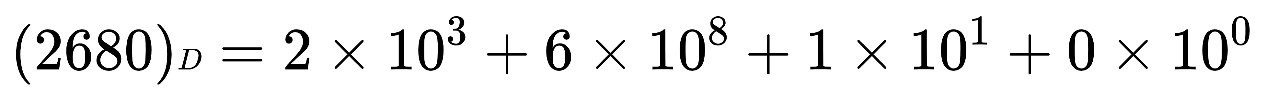
十进制是日常中用的最普遍的一个进制。以10为基数，缝10进1。十进制由十个数码组成，即0、1、2、3、4、5、6、7、8、9。当要表示“十”的时候，则不能用其中任意一位数码来表示了，就需要进位表示为10。

每个数码处于不同位置时，其表示的数值也是不同的，且都为10的整数幂。比如2680这个数就可以表示为：



通常来说，我们会给一个数加上括号并在括号的右下角用角标来表示数的进制，十进制的2680也就是(2680)10。有时候也可以用十进制(Denary)的首字母D来表示，即(2680)D。因此，以上式子也就等同于：





**二进制**

二进制就是以2为基数的计数体制，它只采用0和1两个符号，且缝二进一。二进制里没有2这个数，所以1+1=10（注意，这里是“壹零”，不是“十”）。

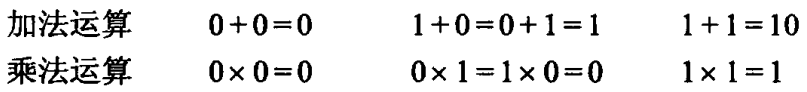
和十进制一样，二进制数N可以被表示为(N)2或者(N)B，B即二进制(Binary)的英文首字母缩写。二进制的各位数都可以表示为2的整数幂，对于一个二进制数101.11，就可以表示为：



图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

因为二进制装置简单可靠，所用元器件较少，且基本运算规则简单，目前数字电路普遍采用二进制。

二进制数的运算规则如下：



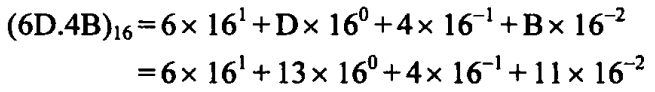
图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

**十六进制和八进制**

二进制虽然运算简单，但是在表示相同的数时二进制所用数位要比十进制多得多，不便于书写和记忆。因此计算机中资料常用十六进制或八进制来表示，以弥补二进制的不足（至于为什么不使用十进制……这个你等会在看了“进制转换”那一章之后就知道了）。

十六进制采用了十六个数码，缝十六进一。除了我们熟悉的0、1、2、3、4、5、6、7、8、9以外，还有A、B、C、D、E、F。其中A、B、C、D、E、F分别对应了十进制中的10、11、12、13、14、15。十六进制的表示形式为(N)16或者(N)H，H代表其英文Hexadecimal。

十六进制数各位都是十六的整数幂，如十六进制数6D.4B，其就可以表示为：



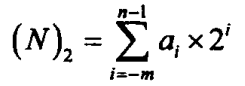
图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

八进制采用8个计数符号：0、1、2、3、4、5、6、7，进位基数是8（缝八进一）。八进制数N也可以用(N)8或者(N)O表示。各个位也都为8的整数幂。对于八进制数374.6，其就是：

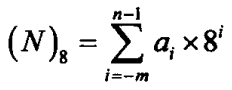


图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

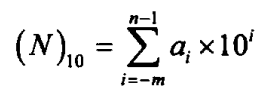
上图这样的式子通常称为“按权展开式”，对于任意二进制数、八进制数、十进制数及十六进制数，其按权表达式如下：



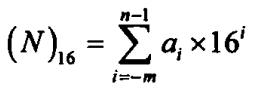
*二进制数的按权表达式*



*八进制数的按权表达式*



*十进制数的按权表达式*



*十六进制数的按权表达式*

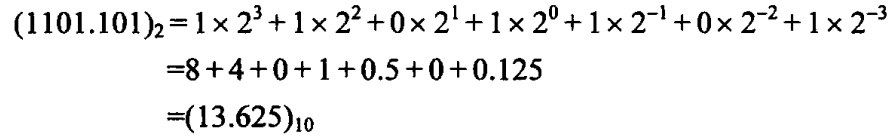
图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

## 进制转换

**二进制、八进制、十六进制转十进制**

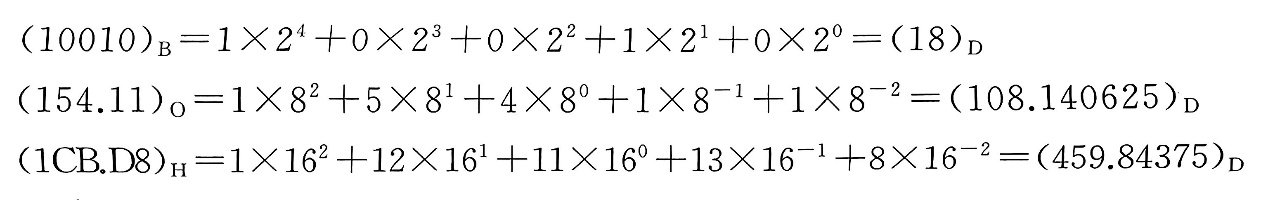
此处仅介绍“按权展开法”，二进制转十进制还有“基数连乘、连除法”，在此不做介绍，感兴趣的读者可以自行查阅。

“按权展开法”即将该数按权展开，将各个位乘以该位的权重，并将每一位所得结果相加。比如对于二进制数1101.101，将其转换为十进制数的过程如下：



图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

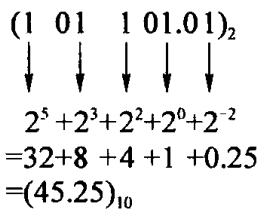
将十六进制数、八进制数转换为十进制数也是同理。



图片来源：清华大学出版社《数字电路与逻辑设计》

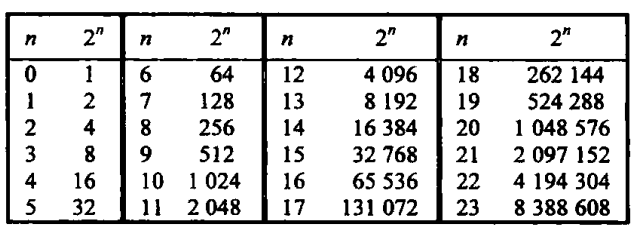
对于二进制数，该过程还可以进一步简化。二进制数中我们只需要将1所在位数的权值相加即可，无需考虑相乘和0所在位数。

比如，将101101.01转换为十进制数：



图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

此方法要求对2的各次幂比较熟悉才能较快实现转换。下表列出了2的各次幂所对应的数：



图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

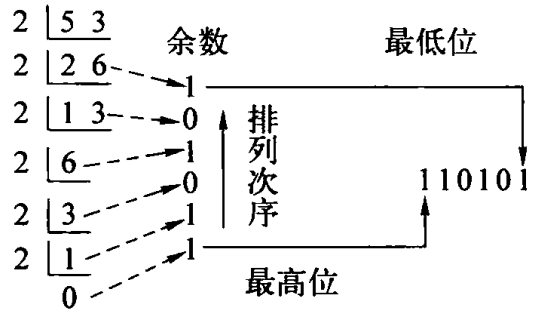
**十进制转二进制、八进制或十六进制**

十进制转任意进制都可以用“除R取余法”或者“乘R取整法”。“除R取余法”适用于将十进制整数转换为R进制整数，而“乘R取整法”则适用于将十进制小数转换为R进制小数。

红石电路中基本不会出现小数的情况，故此处只介绍“除R取余法”。对“乘R取整法”感兴趣的读者可以在网上自行查阅相关资料。

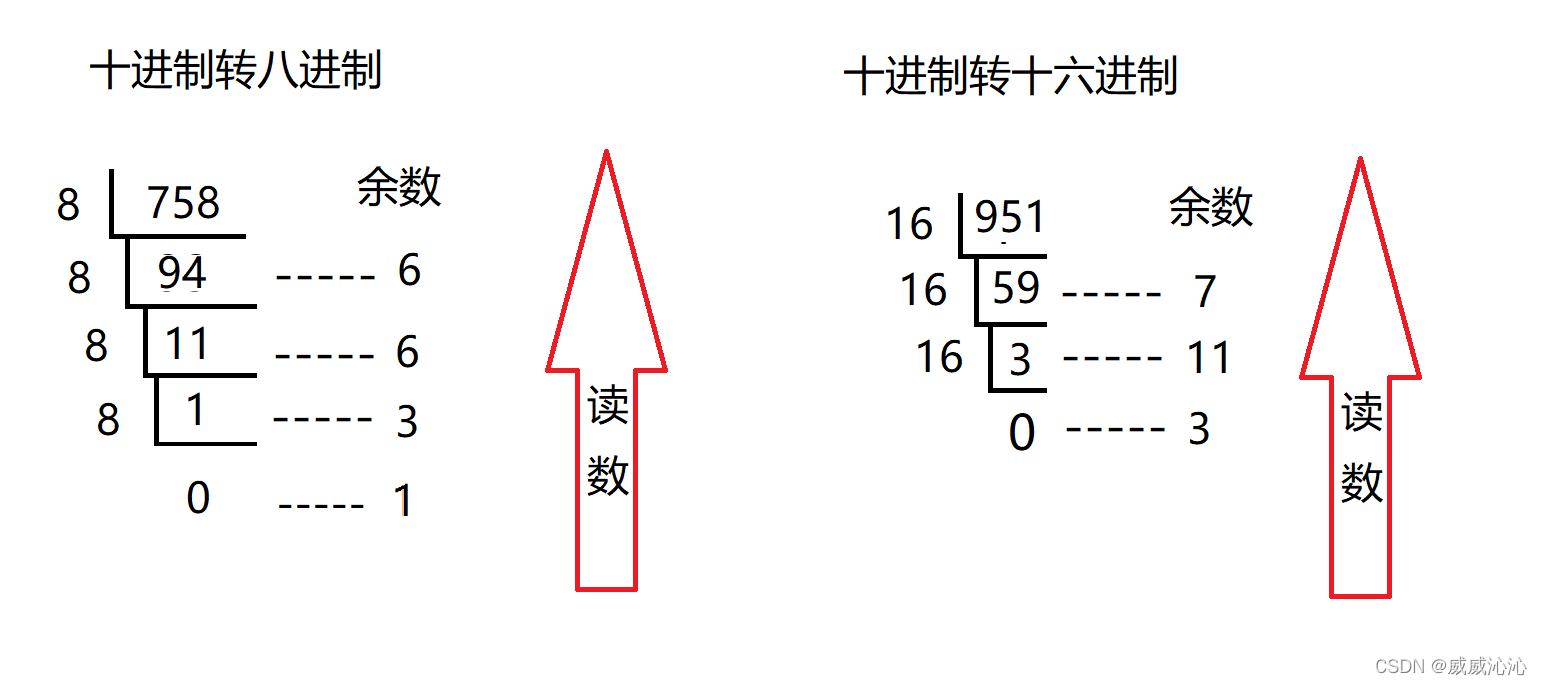
具体来说，我们将该整数除以R（R是要转换为的进制，比如说我们要转换为二进制的话就是除2），得到一个商和一个余数。将余数记录在旁边，然后将商再除以R，接着记录余数和商，直到商为0为止。之后将右侧是余数倒序排列即为得到的数。

比如我们要将53转换为二进制，其过程如下：



图片来源：《数字逻辑电路与系统设计第二版 (蒋立平著)》

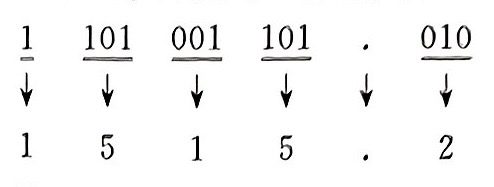
转换为八进制和十六进制也是同理，不过如果将十进制转换为十六进制的时候余下了高于9的数字，则应将其记录为对应的字母（A=10，B=11，C=12，D=13，E=14，F=15）。

图片来源：CSDN论坛

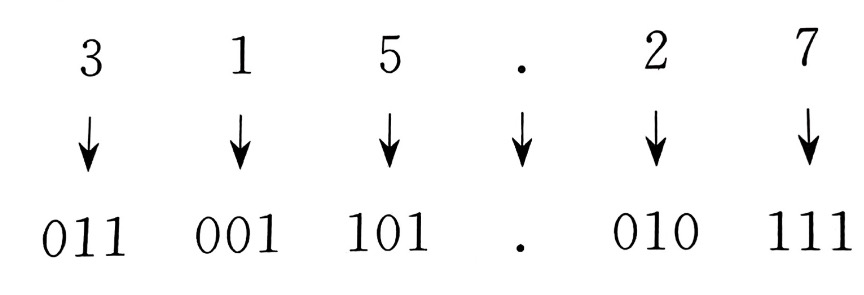
**二进制与八进制、十六进制的转换**

将二进制转换为八进制和十六进制则简单很多，因为2的3次方是8，且2的4次方是16，因此将二进制和八进制互相转化只需要每3位逐个转换就行了，十六进制则是每4位逐个转换，反之亦然。

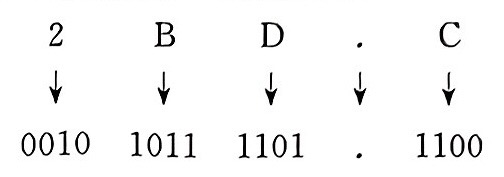
二进制数1101001101.01转换为八进制：



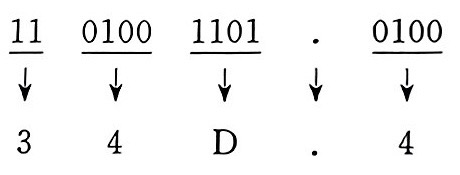
八进制数315.27转换为二进制：



十六进制数2BD.C转换为二进制：



二进制数1101001101.01转换为十六进制：



图片来源：清华大学出版社《数字电路与逻辑设计》

【第一期END，敬请期待第二期】