**编程实现图灵机XN\*2（C语言实现）**

[编程实现图灵机XN\*2（C语言实现）\_橙橙的橙橙子的博客-CSDN博客\_c语言制作图灵机](https://blog.csdn.net/weixin_44361543/article/details/88876648?utm_medium=distribute.pc_relevant.none-task-blog-2~default~baidujs_baidulandingword~default-1-88876648-blog-111479479.pc_relevant_aa2&spm=1001.2101.3001.4242.2&utm_relevant_index=4)

一、题目分析

所谓的图灵机就是指一个抽象的机器，它有一条无限长的纸带，纸带分成了一个一个的小方格，每个方格有不同的颜色。有一个机器头在纸带上移来移去。机器头有一组内部状态，还有一些固定的程序。在每个时刻，机器头都要从当前纸带上读入一个方格信息，然后结合自己的内部状态查找程序表，根据程序输出信息到纸带方格上，并转换自己的内部状态，然后进行移动。

通过编程的方式模拟实现图灵机的运行过程，我们需要先对图灵机的操作步骤进行抽象概括。在本次实验中我将通过C语言编程的方式模拟图灵机（XN\*2）的运行过程实现求某个整数的倍数运算。

分析：

根据图灵机（XN\*2）的运行过程，可将其操作步骤概括为以下五点：

将输入的十进制数转换成二进制

将二进制数转换成计算机能够直接识别的扩展二进制数

根据指令实现对扩展二进制数按指令操作

因为操作结果仍为扩展二进制数，则需要将其收缩为普通二进制数

将收缩完成的二进制数转化为十进制数输出

结论：

可以根据这五个步骤设计五个函数，通过main()函数依次调用它们，从而实现对图灵机的编程模拟。

二、算法构造

● 设计transform()函数实现将输入的十进制数转换为二进制

● 设计extend()函数实现扩展二进制

● 设计realize()函数实现对扩展二进制的按指令操作

● 设计shrink()函数实现将扩展二进制收缩为普通二进制

● 设计reverse()函数实现将二进制转换为十进制

已知图灵机（XN\*2）的指令如下:

0 0 → 0 0 R，

0 1 → 1 0 R，

1 0 → 0 1 R，

10 0 → 11 1 R，

11 0 → 0 1 STOP。

这五条指令是设计realize()函数的理论依据。

————————————————

版权声明：本文为CSDN博主「橙橙的橙橙子」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/weixin_44361543/article/details/88876648>



# 图灵机模拟程序功能设计

图灵机由无限长的纸带、读写头、状态寄存器、控制规则等四部分组成，纸带上的符号可以是{ 0，1，空格 }。要利用图灵机求解一个问题，需要自己设计图灵机“程序”，即定义一些状态（其中包括初始状态和结束状态），设计给出控制规则，并进行图灵机初始化（设定初始状态和读写头的位置）。图灵机运行时，只要给定问题的一个输入（纸带上的初始符号），则图灵机就能按照设计的控制规则运行得到要求的输出（结束状态时纸带上的符号）。

请设计开发一个单带（即只有一个读写头）图灵机模拟程序，功能包括：

（1）定义需要的状态；

（2）设计并输入控制规则；

（3）初始化；

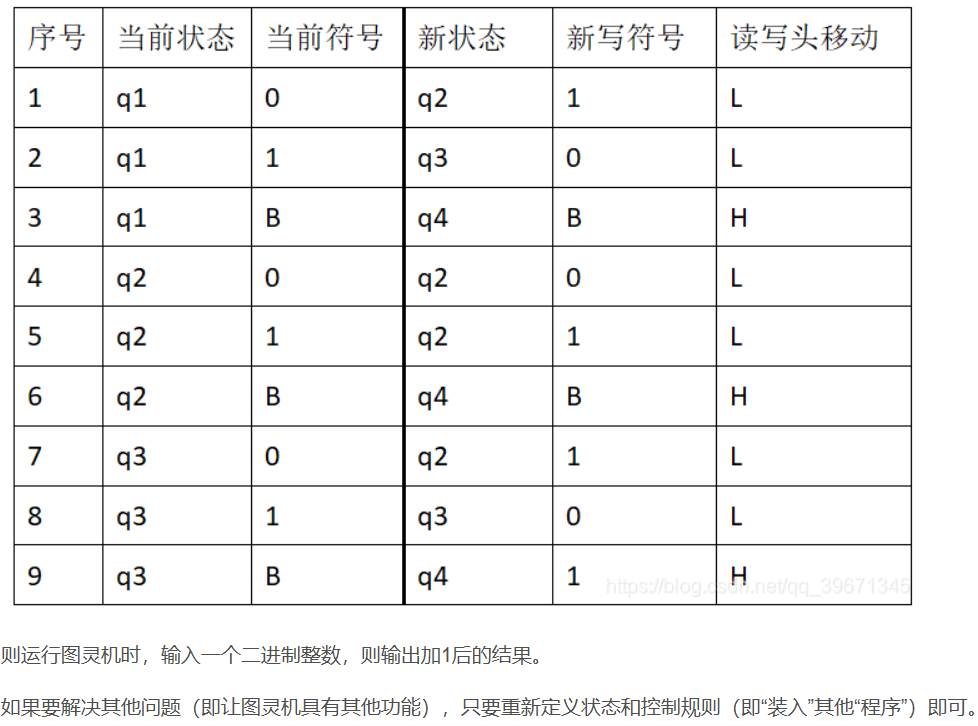
（4）运行时接收输入（即初始化纸带上的符号）；

（5）运行后输出结果；如果运行不正确，应该能报错。

例如，假设图灵机要实现二进正整数的加1运算，则只要给图灵机：

（1）定义4个状态： { q1,q2,q3,q4 }，其中q1是机器的初始状态，q4是机器的结束状态

（2）设计并输入如下的规则：



————————————————

版权声明：本文为CSDN博主「ANAN永不败315」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/qq_39671345/article/details/111479479>

# 【计算理论】图灵机 ( 图灵机设计 )

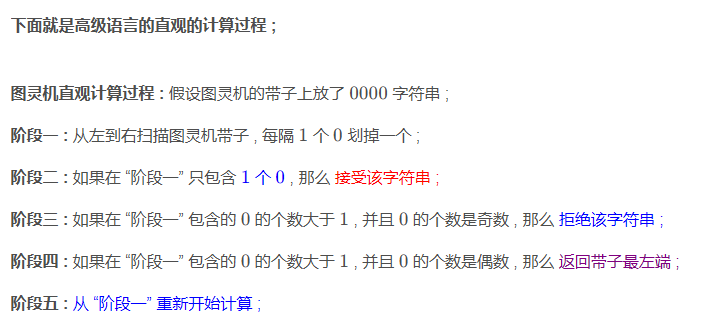
设计一个图灵机M2, 认识一种特定语言 , 该语言由0组成 , 字符串的长度是2^n个, n=0,1,2,\cdots,

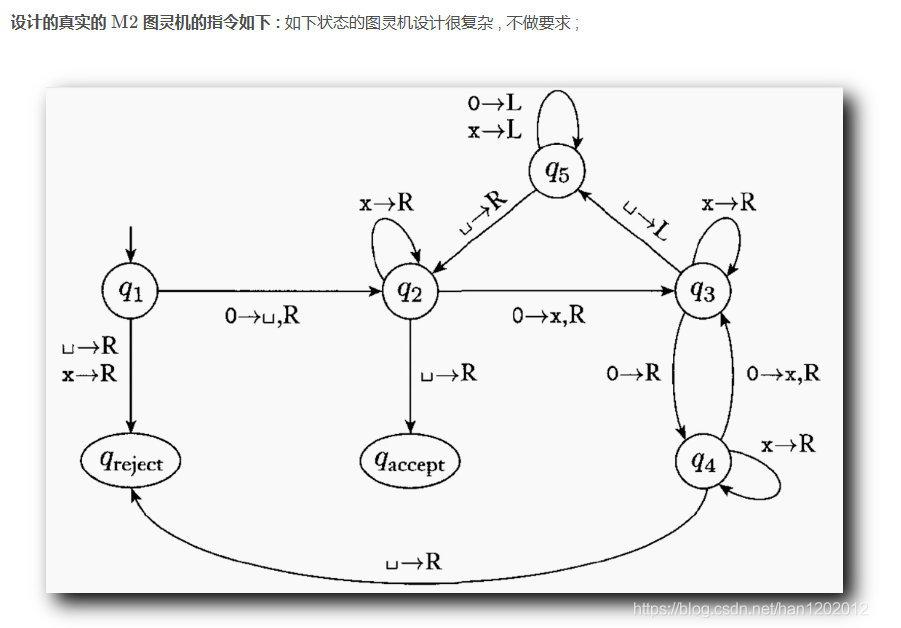
将字符串写到带子上 , 带子上每隔一个0划掉一个 , 数一下剩下的0:

① 如果剩下的0是 1个 , 直接接受该字符串 ;

② 如果剩下的0是奇数个 , 直接拒绝接受该字符串 ;

③ 如果剩下的0是 偶数个 , 继续重新开始循环 ;



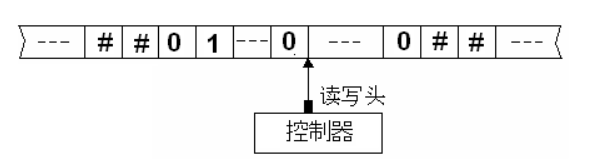


版权声明：本文为CSDN博主「韩曙亮」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

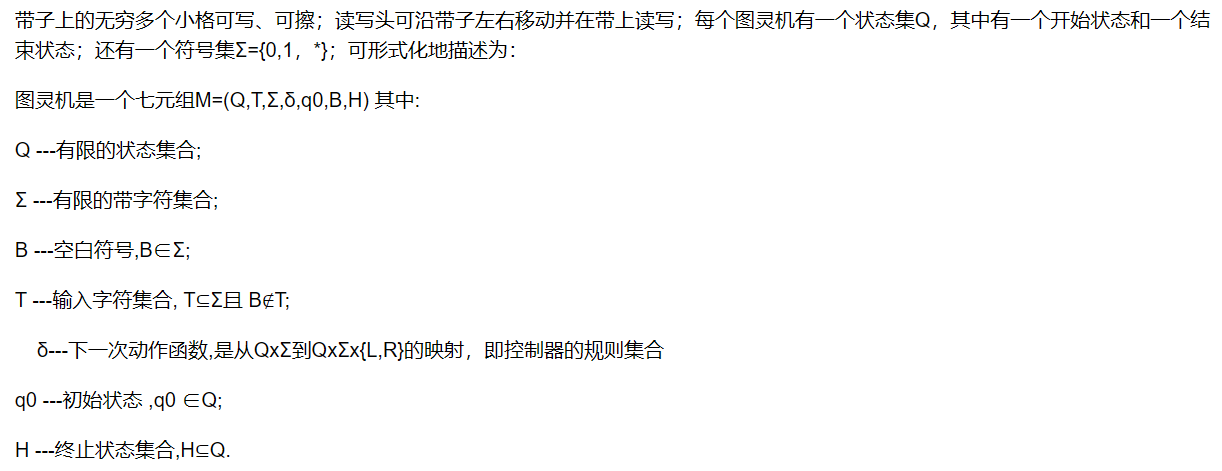
原文链接：<https://blog.csdn.net/shulianghan/article/details/110563801>

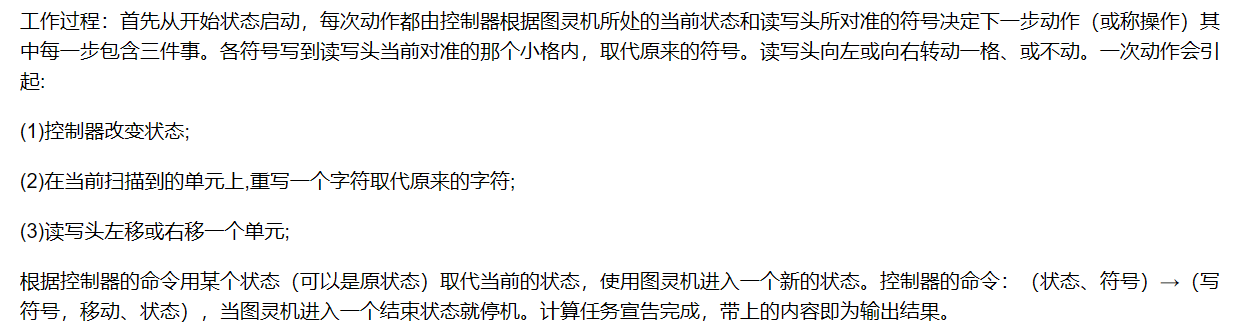
图灵机(TM)是一种重要的计算模型，它由英国数学家A．M．Turing于1936年提出。这个模型很好的描述了计算过程。无数的事实表明，任何算法都可以用一个图灵机来描述，这就是著名的丘奇论题。图灵机在可计算性理论中起着重要作用。可以证明图灵机识别的语言就是0型语言。

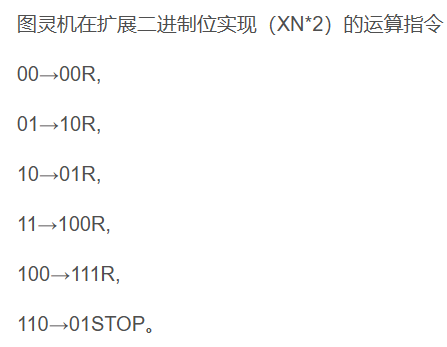
Components. Alan Turing sought to describe the most primitive model of a mechanical device that had the same basic capabilities as a human "computer."图灵机The machine consists of the following compone的组成如下图所示：



它由一个状态控制器，一个读写头和一个输入带组成。其中输入带左右端可以无限伸长。带上的每一格恰好有一个字符。开始时，带上从编号为0开始的n个格存放着由有限输入字母表上的字符组成的字符串，第0格及其左边和第n＋1格及其右边各格均为空白。空白是一个特殊的带符号，它不属于输入字母表。读写头一次可以在带上读或写一个字符，并可根据指令向左或向右移一格。状态控制器根据当前的状态，读到输入字符并发布指令。指令的内容包括状态转换，在带上的一格写上(更换)字符，以及读写头向左或向右移动一格等。











————————————————

版权声明：本文为CSDN博主「将卿。」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/qq\_42272391/article/details/88857331/