

---

# 用灯泡做一个短信发送装置

## 从开关和灯泡理解二进制编码

---

Namishu

这份讲义从一个只能亮或灭的灯泡开始，逐步构造一套可以发送拼音信息的编码方案。读者会在故事中看到二进制、编码表、编码和解码这些概念是怎样自然出现的。

Copyright © 2025 Namishu. All rights reserved.

本作品由 Namishu 发布，包含学习资料、练习题、版式设计及相关内容。你可以将它用于个人学习、研究和非商业分享，也可以为个人用途进行修改。请保留版权信息。

未经书面授权，不得用于商业项目、付费产品、企业或组织场景；不得转售、重新发布原始作品，或声称自己是作者。

完整授权协议：<https://namishu.com/license>

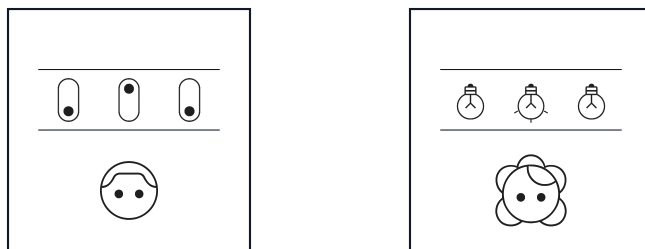
商业授权咨询：[hello@namishu.com](mailto:hello@namishu.com)

# 目录

---

1 点亮一个灯泡	1
2 发送一个字	2
3 发送很多字	3
3.1 汉字编码 .....	3
3.2 拼音编码 .....	3
4 最少几个灯泡	5
5 做一个编码表	7
6 发送信息	8
7 接收信息	8
8 完整方案	9
9 二进制计数	10
10 二进制与计算机	11

现在有两个房间。在一个房间中，放置了一些开关；在另一个房间中，放置了一些灯泡。这些开关可以分别控制灯泡的亮和灭。



阿果在有开关的房间，小红在有灯泡的房间。两个人不能直接说话，只能通过灯泡的亮和灭传递信息。

阿果通过控制开关，向小红发送了一条信息。小红在另一个房间中，看见了灯泡不停的亮和灭。最后，小红明白了阿果发送的信息。

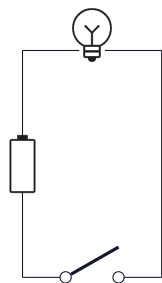
我们想一想，阿果和小红是怎么做到的？

## 1 点亮一个灯泡

先研究一下怎么点亮一个灯泡。需要准备如下材料。



把它们按照下面的方式连接。



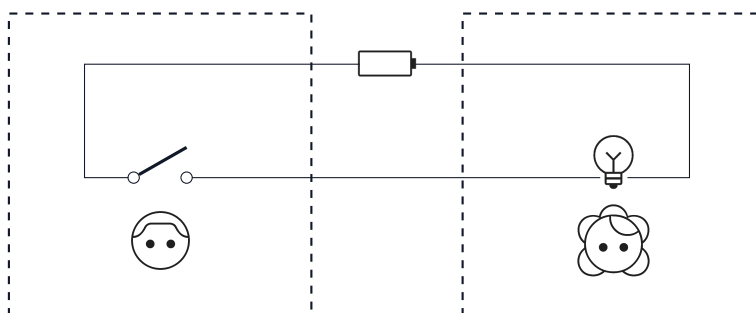
开关闭合时，灯泡就发光。开关打开时，灯泡就熄灭。这样一来，我们可以通过开关来控制灯泡是否发光。

### 一个灯泡的两种状态

灯泡只有“亮”和“灭”两种状态。只要提前约定好这两种状态分别代表什么，它就可以传递两个不同的信息。

## 2 发送一个字

现在把开关和灯泡分别放在两个房间。它们和电池一起，仍然按照上面的方式进行连接，如下图所示。



阿果按下开关，小红看见灯泡亮了。阿果抬起开关，小红看见灯泡灭了。

现在阿果和小红约定好如下内容：灯泡亮的时候，代表数字“1”；灯泡灭的时候，代表数字“0”。这个内容可以写成如下表格的形式。

状态	字符
亮	1
灭	0

于是，阿果可以通过“按下”和“抬起”开关，向隔壁的小红发送数字“1”或者“0”。

这是一套简单的信息发送装置。它包括一个键盘，这个键盘只有一个按键（开关）；还有一个屏幕（灯泡），它只能显示一个“字符”。这个“字符”需要通过查询上面的表格得到。

这个表格就是双方的“通信协议”。如果阿果不想发送数字，也可以约定其他内容（例如下表）。

状态	字符
亮	开心
灭	难过

小红通过观察灯泡，就知道阿果的心情。灯亮表示开心，灯灭表示难过。

### 通信协议

同一套灯泡状态可以表示数字，也可以表示心情。关键不在灯泡本身，而在阿果和小红事先约定的表格。这个表格就是通信协议。

阿果玩了一会儿这个装置。刚开始觉得有意思，但很快就不满意了。

他觉得发送“0”“1”或者“开心”“难过”，其实都是一样的，都是灯泡的两种状态。换句话说，这个装置一次只能表示两种不同的信息：“亮”是一个信息，“灭”是另一个信息。

## 3 发送很多字

阿果想要发送更多的信息。现在这个装置是做不到的。那么，应该如何改进呢。

### 3.1 汉字编码

阿果想到一个办法。就是使用更多的灯泡。把常用的汉字列出来，每个汉字用一个灯泡表示。每个灯泡用一个开关控制。如果灯泡亮了，那么这个汉字被选中。



常用汉字有 3000 多个。他至少要准备 3000 个灯泡、3000 个开关、3000 个电池，还有很多电线。然后按照前面说的方法，把线路连接起来，让每个开关能够独立控制对应的灯泡。

接下来就是通信协议。阿果需要给灯泡编号，然后记录每个灯泡代表的字符，并做成一张表格（如下所示）。

编号	1	2	3	4	5	6	...
字符	你	我	他	天	地	人	...

这个表有 3000 列（不含表头），每列代表一个灯泡，每个灯泡代表一个字符。例如，阿果按下 4 号灯泡的开关，隔壁小红就看见 4 号灯泡亮了。于是小红查表，得到 4 号是“天”字。

通过这种方式，阿果可以发送的信息量从以前的 2 个变成了现在的 3000 个。这样一来，阿果想说的话，都可以通过这个装置发送给小红。

但是阿果仍然不满意。他觉得这么做太麻烦了，因为要用到 3000 个灯泡！他不想让这个装置太复杂。或者说，他想要让灯泡的数量少一些。

#### 直接编码的问题

一个字符对应一个灯泡，虽然容易理解，但是非常浪费。想表示 3000 个汉字，就要准备大约 3000 个灯泡。

### 3.2 拼音编码

他想到了拼音。用拼音代替汉字。拼音只有 26 个字母。注意：在拼音输入中，字母 v 通常用来表示 ü。因此，只需要 26 个灯泡。而且通信协议也更简单。如下表所示，每一个编号，表示一个灯泡；每个灯泡，对应一个字符。

编号	1	2	3	4	5	6	7	8
字符	a	b	c	d	e	f	g	h
编号	9	10	11	12	13	14	15	16
字符	i	j	k	l	m	n	o	p
编号	17	18	19	20	21	22	23	24
字符	q	r	s	t	u	v	w	x
编号	25	26						
字符	y	z						

例如，阿果想给小红发送“你好”，那么就发送拼音“nihao”。从第一个字母 n 开始，查表得到 n 的编号是 14；然后是 i - 9，h - 8，a - 1，o - 15。小红在另一个房间，她先看见 14 号灯泡亮了，然后查表得到 n，接着就是 9 号灯亮，查表得到 i，以此类推，最后得到“nihao”。

阿果还是不满意。他觉得拼音得有音调。另外，两个拼音之间还需要一个“空格”。比如前面说的“nihao”应该是“nǐ hǎo”。因此，还需要增加 5 个灯泡，其中 4 个代表音调，1 个代表空格。

接下来更新通信协议。用数字“1, 2, 3, 4”表示四个音调，用“\_”表示空格符号。完整的表格如下所示。

编号	1	2	3	4	5	6	7	8
字符	a	b	c	d	e	f	g	h
编号	9	10	11	12	13	14	15	16
字符	i	j	k	l	m	n	o	p
编号	17	18	19	20	21	22	23	24
字符	q	r	s	t	u	v	w	x
编号	25	26	27	28	29	30	31	
字符	y	z	1	2	3	4	_	

这样一来，阿果如果想发送“nǐ hǎo”，可以查表找到“ni3\_hao3”中每个字符的灯泡编号，然后依次亮灯。小红看到了点亮的灯泡，接着查表就得到“ni3\_hao3”。于是，她就明白了，这个意思代表“nǐ hǎo”。

太棒了。这个装置是能用的，而且不复杂。

但是阿果不满足，他觉得灯泡数量应该能再少一些。因为他发现，当初用一个灯泡，就能够表示两个“字符”，例如 0 和 1。现在的装置表示 31 个字竟然需要 31 个灯泡。

阿果想进一步优化这个装置。他现在需要思考的问题是：最少几个灯泡可以表示 31 个字符？

## 4 最少几个灯泡

现在的问题是：如果要表示 31 个不同的字符，最少需要几个灯泡？

阿果决定从一个灯泡开始尝试。为了方便描述，他用 0 代表灯灭，1 代表亮灯。容易发现，一个灯泡只有两种状态。为了方便观察，阿果把结果写成表格形式。

灯泡 1	代码
灭	0
亮	1

现在增加一个灯泡。把“灯泡 2”放在“灯泡 1”的左边。容易发现，两个灯泡对应了下面四种状态。

灯泡 2	灯泡 1	代码
灭	灭	00
灭	亮	01
亮	灭	10
亮	亮	11

这就很有意思了。两个灯泡可以表示 4 个字！

那么三个灯泡能表示几个字呢？阿果很期待。他把“灯泡 3”放在“灯泡 2”的左边，继续做实验。

灯泡 3	灯泡 2	灯泡 1	代码
灭	灭	灭	000
灭	灭	亮	001
灭	亮	灭	010
灭	亮	亮	011
亮	灭	灭	100
亮	灭	亮	101
亮	亮	灭	110
亮	亮	亮	111

数了一下，一共是 8 种状态。回想一下前面的结果：1 个灯泡有 2 种状态；2 个灯泡有 4 种状态；3 个灯泡有 8 种状态。阿果猜测这个规律就是：每增加一个灯泡，状态的数量是之前的两倍。因此，4 个灯泡有 16 种状态，5 个灯泡有 32 种状态。

阿果猜测，答案是 5 个灯泡！

### 状态数量

每个灯泡都有亮和灭两种状态。灯泡数量每增加 1 个，所有可能的状态数量就会乘以 2。因此， $n$  个灯泡最多可以表示  $2^n$  种不同的信息。

那就验证一下。数一数 5 个灯泡能表示几种状态。

灯泡 5	灯泡 4	灯泡 3	灯泡 2	灯泡 1	代码
灭	灭	灭	灭	灭	00000
灭	灭	灭	灭	亮	00001
灭	灭	灭	亮	灭	00010
灭	灭	灭	亮	亮	00011
灭	灭	亮	灭	灭	00100
灭	灭	亮	灭	亮	00101
灭	灭	亮	亮	灭	00110
灭	灭	亮	亮	亮	00111
灭	亮	灭	灭	灭	01000
灭	亮	灭	灭	亮	01001
灭	亮	灭	亮	灭	01010
灭	亮	灭	亮	亮	01011
灭	亮	亮	灭	灭	01100
灭	亮	亮	灭	亮	01101
灭	亮	亮	亮	灭	01110
灭	亮	亮	亮	亮	01111
亮	灭	灭	灭	灭	10000
亮	灭	灭	灭	亮	10001
亮	灭	灭	亮	灭	10010
亮	灭	灭	亮	亮	10011
亮	灭	亮	灭	灭	10100
亮	灭	亮	灭	亮	10101
亮	灭	亮	亮	灭	10110
亮	灭	亮	亮	亮	10111
亮	亮	灭	灭	灭	11000
亮	亮	灭	灭	亮	11001
亮	亮	灭	亮	灭	11010
亮	亮	灭	亮	亮	11011
亮	亮	亮	灭	灭	11100
亮	亮	亮	灭	亮	11101
亮	亮	亮	亮	灭	11110
亮	亮	亮	亮	亮	11111

## 5 做一个编码表

阿果数完发现，5 个灯泡有 32 种状态。这 32 种状态用代码表示就是

```
00000 00001 00010 00011 00100 00101 00110 00111
01000 01001 01010 01011 01100 01101 01110 01111
10000 10001 10010 10011 10100 10101 10110 10111
11000 11001 11010 11011 11100 11101 11110 11111
```

接下来只要定义每种状态代表的“字符”就可以了。跟之前的做法一样，用一张表格把代码和字符对应起来。

代码	00000	00001	00010	00011	00100	00101	00110	00111
字符		a	b	c	d	e	f	g
代码	01000	01001	01010	01011	01100	01101	01110	01111
字符	h	i	j	k	l	m	n	o
代码	10000	10001	10010	10011	10100	10101	10110	10111
字符	p	q	r	s	t	u	v	w
代码	11000	11001	11010	11011	11100	11101	11110	11111
字符	x	y	z	1	2	3	4	-

为了方便描述，我们把这个表称为“编码表”。这个编码表就可以当作阿果和小红的通讯协议。接下来阿果就可以给小红发送信息了。

### 编码表

编码表把“代码”和“字符”对应起来。发送者用它把字符变成代码，接收者用它把代码还原成字符。

## 6 发送信息

我们以拼音“nǐ hǎo”为例来说明发送信息的过程。它对应的字符是“ni3\_hao3”。接下来查表找到每个字符对应的编码（如下表所示）。

字符	编码
n	01110
i	01001
3	11101
_	11111
h	01000
o	01111
3	11101

▪ **编码** 这个步骤称为“编码”，目的是把“字符”变成“代码”。有了代码，接下来就是去执行这个代码：点亮对应的灯泡。

以 01110 为例，它表示点亮灯泡 2-4，其他灯泡灭，如下表所示。

灯泡 5	灯泡 4	灯泡 3	灯泡 2	灯泡 1	代码
灭	亮	亮	亮	灭	01110

## 7 接收信息

小红看见了灯泡的亮和灭。例如，灯泡的状态如下所示。

灯泡 5	灯泡 4	灯泡 3	灯泡 2	灯泡 1
灭	亮	亮	亮	灭

小红知道这表示代码 01110，于是她查“编码表”得知，这个字表示“n”。

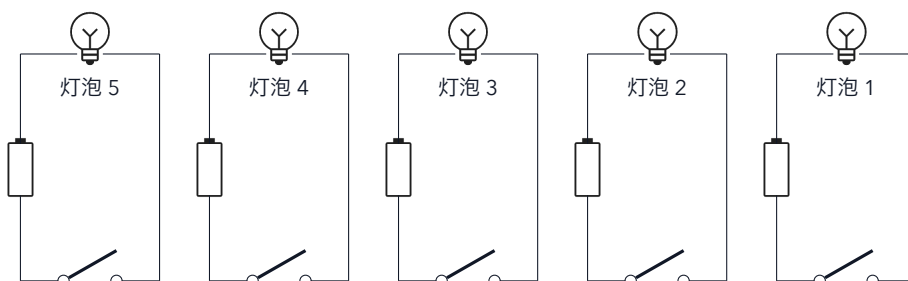
▪ **解码** 这个步骤称为“解码”，目的是把“代码”变成“字符”。小红按照上面的解码步骤，就可以把阿果发送的代码逐个翻译成字符，从而得到完整的信息。

### 发送和接收

发送信息时，阿果做的是“字符 → 代码 → 灯泡状态”。接收信息时，小红做的是“灯泡状态 → 代码 → 字符”。

## 8 完整方案

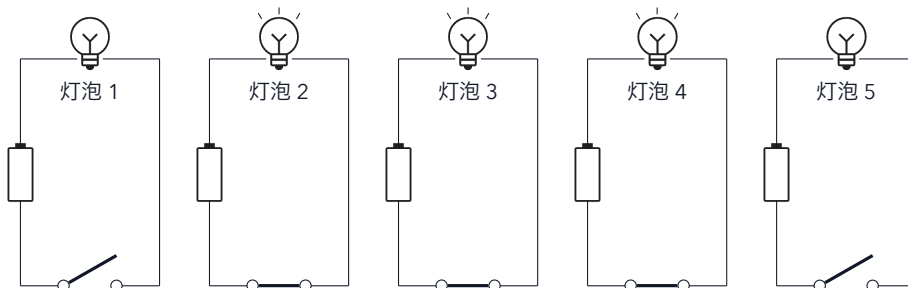
现在阿果有了一个短信发送的装置，支持发送拼音字符。这个装置包含 5 个开关，作为打字的键盘；5 个灯泡作为显示字符的屏幕；还有一些电池和电线，为装置提供能量。装置的示意图如下所示。



它基于下面的编码表进行通信。

代码	00000	00001	00010	00011	00100	00101	00110	00111
字符		a	b	c	d	e	f	g
代码	01000	01001	01010	01011	01100	01101	01110	01111
字符	h	i	j	k	l	m	n	o
代码	10000	10001	10010	10011	10100	10101	10110	10111
字符	p	q	r	s	t	u	v	w
代码	11000	11001	11010	11011	11100	11101	11110	11111
字符	x	y	z	1	2	3	4	-

接下来介绍如何发送短信。以发送字母 n 为例。n 对应的代码是 01110。于是我们闭合“灯泡 2-4”对应的开关。



接收短信的步骤是反过来的。接收人观察灯泡，得到代码 01110。然后查询编码表得到 n。

### 完整流程

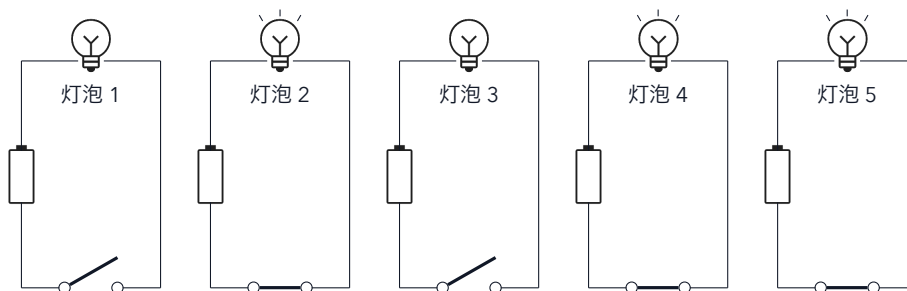
只要双方使用同一张编码表，发送者就能把文字变成灯泡状态，接收者也能把灯泡状态还原成文字。

## 9 二进制计数

阿果用发短信的时候，突然有了一个新的想法。他想换一个编码表。既然它可以对拼音进行编码，那么它也能对数字进行编码。比如像下面这样。

代码	00000	00001	00010	00011	00100	00101	00110	00111
字符	0	1	2	3	4	5	6	7
代码	01000	01001	01010	01011	01100	01101	01110	01111
字符	8	9	10	11	12	13	14	15
代码	10000	10001	10010	10011	10100	10101	10110	10111
字符	16	17	18	19	20	21	22	23
代码	11000	11001	11010	11011	11100	11101	11110	11111
字符	24	25	26	27	28	29	30	31

于是，这个装置就可以当作计数器！例如，下图代表数字“11”。



阿果接着意识到，其实不需要灯泡。因为**代表数字的是“代码”，而不是灯泡**。灯泡只是用来显示“位置”。打个比方，第一个灯泡是个位，第二个灯泡是十位，第三个灯泡是百位，以此类推。

阿果想起了数数。考虑两种方法计数。

- 第一种方法用到符号“0-9”，称为“十进制”。从第一个位置开始数，从0数到9；下一个数进位，得到“10”。然后11，12，以此类推。
- 第二种方法用到符号“0-1”，称为“二进制”。从第一个位置开始数，从0数到1；下一个数进位，得到“10”；接着是“11”；下一个进位，得到“100”，以此类推。

通过数数，他得到十进制和二进制的对应关系，如下所示。

二进制	0	1	10	11	100	101	110	111
十进制	0	1	2	3	4	5	6	7
二进制	1000	1001	1010	1011	1100	1101	1110	1111
十进制	8	9	10	11	12	13	14	15
二进制	10000	10001	10010	10011	10100	10101	10110	10111
十进制	16	17	18	19	20	21	22	23
二进制	11000	11001	11010	11011	11100	11101	11110	11111
十进制	24	25	26	27	28	29	30	31

阿果现在明白了一个事实，那就是同一个“数”可以用不同的符号来表示。它可以用十进制表示，也可以是二进制。例如，二进制“10010”和十进制“18”是同一个“数”；二进制“1010”和十进制“10”也是同一个数。

#### 同一个数，不同写法

二进制和十进制只是写数的方法不同。只要表示的是同一个数，它参与加法、减法、乘法和除法时，数的大小不会因为写法改变。

接下来，阿果试着把这两个二进制的数加起来。他注意到二进制“1+1”等于“10”，相当于从“1”开始数数，那下一个数就是“10”。

$$\begin{array}{r}
 1\ 0\ 0\ 1\ 0 \\
 +\ 0\ 1\ 0\ 1\ 0 \\
 \hline
 1\ 1\ 1\ 0\ 0
 \end{array}$$

他发现二进制的“11100”，就是十进制的“28”，等于十进制的“18+10”。换句话说，加法不区分“进制”。用十进制也好，二进制也好，加出来的数只是写法不同，大小是相同的。

既然加法不区分进制，阿果猜测减法也不区分。那乘法和除法呢。阿果想到这里，拿起了纸和笔，他要验证这个想法……

故事先讲到这里。

## 10 二进制与计算机

在计算机中，所有的数据——无论是文字、图片、音乐还是视频——都被转换成了二进制。你可以把二进制看作是计算机的“语言”，它用0和1来表示所有信息。

比如说，图片和视频的每一个像素点，都是通过二进制数字来存储的。音乐文件也是一样，声音的波形被转化成数字，通过一串0和1来表示。即使是你正在玩的一款游戏，所有的图像、动画和声音效果背后，实际上都是二进制数据在运行。

这种看似简单的数字组合，实际上蕴含了复杂的计算和信息处理能力，正是通过这些基本的0和1，计算机能够完成各种我们日常使用的任务。