

1 范围

本标准规定了 QR 码符号的要求。它规定了 QR 码模式 2 符号的特征，数据字符编码，符号格式，尺寸特征，[错误纠正纠错](#)规则，参考译码算法，符号质量要求，以及可由用户选择的应用参数，在附录中给出了 QR 码模式 1 符号不同于模式 2 的特性。

2 一致性

QR 码符号（及设计用于生成或识读 QR 码符号的设备）如果满足 QR 码模式 2 或模式 1 规定的要求，应认作符合本规范。然而要注意，模式 2 是推荐用于新的和开放式系统应用的符号方式。

3 引用标准

下列标准文件所包含的条文，通过在本标准中引用而构成本标准的条文。对于注明日期的引用标准，以后进行的补充和修改并不适用，然而，鼓励基于国际标准的协议各方对应用以下标准文件最新版本的可能性进行调研，无注明日期的引用标准适用于提交应用的最近的版本。ISO 和 IEC 的成员仍然是当前有效标准的注册机构。

ISO/IEC 15424 信息技术——自动识别和数据采集技术
——数据载体/符号标识

ISO/IEC 15416 信息技术——自动识别和数据采集技术
——条码印刷质量测试规范
——线性条码

EN 1556 条码——术语

JIS X0201 信息交换用 JIS 8 位字符集

JIS X0208—1997 信息交换用日语图形字符集

ANSI X 3.4 编码字符集——信息交换用 7 位美国国家标准码（7 位 ASCII）

AIM 国际技术规范 扩展解释：第一部分：识别方案与协议（称作“AIM ECI 规范”）

4 术语和定义

EN1556 中的术语和下列各项适用于本标准：

4.1 校正图形(Alignment Pattern)

用于确立矩阵符号位置的一个固定的参照图形，[解码译码](#)软件可以通过它在图象有中等程度损坏的情况下，再同步图像模块的坐标映象。

4.2 字符计数指示符 (Character Count Indicator)

定义某一模式下的数据串长度的位序列。

4.3 ECI 指示符 (ECI designator)

6 位数字，用于标识具体的 ECI 任务。

4.4 编码区域 (encoding region)

在符号中没有被功能图形占用，可以对数据或[错误纠正纠错](#)码字进行编码的区域。

4.5 扩充解释 (Extended Channel Interpretation (ECI))

在某些码制中，对输出数据流允许有与缺省字符集[存](#)不同的解释[的](#)协议。

4.6 扩展图形 (Extension Pattern)

模式 1 中, 不表示数据的一种功能图形。

4.7 格式信息 (Format Information)

一种功能图形, 它包含符号使用的[错误纠正纠错](#)等级以及使用的掩模图形的信息, 以便对编码区域的剩余部分进行译码。

4.8 功能图形 (function pattern)

[包含帮助译码的符号定位或者它的特征识别信息的符号附加成分, 符号中用于符号定位与特征识别的特定图形。](#)

4.9 掩模图形参考 (Mask Pattern Reference)

用于符号中的[三位三位掩模图形](#)标识符。

4.10 掩模 (masking)

在[城内编码区域内](#), 用掩模图形对[在城内编码区的](#)位图进行 XOR 操作, 其目的是使符号中深色与浅色模块数的比例均衡, 并且减少影响图像快速处理的图形出现。

4.11 模式 (mode)

将特定的字符集表示成位串的方法。

4.12 模式指示符 (Mode Indicator)

4 位标识符, 指示随后的数据序列所用的编码模式。

4.13 填充位 (Padding Bit)

值为 0, 不表示数据, 用于填充数据位流最后一个码字中终止符后面的空位。

4.14 位置探测图形 (Position Detection Pattern)

组成寻象图形的三个相同的图形之一。

4.15 剩余位 (Remainder Bit)

值为 0, 不表示数据, 当编码区域不能正好被 8 位的码字填满时, 用于填充最后一个码字后的空位。

4.16 剩余码字 (Remainder Codeword)

[一种填充码字](#), 当所有的数据码字和[错误纠正纠错](#)码字不能正好填满符号的容量时, 用于填充[一种填充码字](#)所空码字位置, 它们紧跟在最后一个[错误纠正纠错](#)码字之后。

4.17 段 (segment)

以同一 ECI 或编码模式编码的数据序列。

4.18 分隔符 (Separator)

全部由浅色模块组成的功能图形, 宽度为一个模块, 用于将位置探测图形与符号的其余部分分开。

4.19 终止符 (Terminator)

用于结束表示数据位流的位图 0000。

4.20 定位图形 (Timing Pattern)

深色与浅色模块交错的图形, 便于决定符号中模块的坐标。

4.21 版本 (Version)

用于表示符号规格的系列。某一特定版本是根据它在所允许的规格系列中的位置来确定的。QR 码所允许规格系列为 21×21 模块（版本 1）～177×177 模块（版本 40）。它也可同时指示符号所应用的纠错等级。

4.22 版本信息 (Version Information)

在模式 2 符号中，包含符号版本的信息及该数据**错误纠正纠错**位的功能图形。

5 运算符（缩略符）

在公式和方程中使用的数学符号，是在它们所在的公式或方程之后定义的。

本规范中使用的数学运算符定义如下：

div 整除运算符

mod 整除后的余数

XOR 逻辑‘异或’功能，当两个输入不等时，它的输出为 1。其符号表示为 \oplus 。

6 有关 QR 码的约定

6.1 模块位置

为便于**参照参考**，**用**在符号中的行、列坐标表示模块的位置，格式为 (i, j)。其中，i 表示模块所在行数（从上向下计算），j 表示列数（从左向右计算），记数从 0 开始，因此模块 (0, 0) 是符号中左上角的模块。

6.2 字节表示

字节的内容用 16 进制表示。

6.3 版本参考

符号版本用版本 V-E 的方式表示，其中 V **标识表示**版本号（1-40），E 表示**错误纠正纠错**等级（L, M, Q, H）。

7 符号描述

本标准的条款规定了 QR 码模式 2 符号的技术规范，除非在附录 M 中作特别说明说明，否则也适用于模式 1 符号。

7.1 基本特征

QR 码为矩阵式二维码，具有如下特征：

a) 编码字符集

- 1) 数字型数据（数字 0～9）；
- 2) 字母数字型数据（数字 0～9；大写字母 A～Z；9 个其他字符：space, \$, %, *, +, -, ., /, :）；
- 3) 8 位字节型数据（与 JIS X 0201 一致的 JIS8 位字符集（拉丁和假名））；
- 4) **日语日本**汉字字符（与 JIS X 0208 附录 1：**变化转换**代码表示法一致的**变化转换** JIS 字符集。注意：在 QR 码中的**日语日本**汉字字符的值为：8140_{HEX} - 9FFC_{HEX} 和 E040_{HEX} - EBBF_{HEX}，可以压缩为 13 位。）
- 5) 中国汉字字符（与 GB18030-2000 附录 A 双字节字符表表示的字符集）

b) 数据表示法

深色模块表示二进制 1，浅色模块表示二进制 0。

c) 符号规格（不包括空白区）

21×21 模块到 177×177 模块（版本 1 到 40，每版本符号每边增加 4 个模块）。

d) 每个符号的数据字符数（最大规格的符号—版本 40-L）

- 1) 数字数据 7089 个字符
- 2) 字母数字数据 4296 个字符
- 3) 8 位字节数据 2953 个字符
- 4) 日本汉字数据 1817 个字符
- 5) 中国汉字数据 21008 个字符

e) 错误纠正纠错的选择

4 种错误纠正纠错等级，可恢复的码字比例为：

L	7%
M	15%
Q	25%
H	30%

f) 符号类型：矩阵

g) 独立定位：是

图 1 为 QR 码符号版本 1 的示例。



图 1 QR 码符号的示例

7.2 其他特征综述

以下 QR 码特征有些是固有的，有些是可选的：

a) 结构链接（可选）

允许把数据文件用最多 16 个 QR 码符号在逻辑上连续地表示。它们可以以任意的顺序扫描，而原始数据能正确地重新连接起来。

b) 掩模（固有）

可以使符号中深色与浅色模块的比例接近 1:1，使因相邻模块的排列造成译码困难的可能性降为最小。

c) 扩充解释（可选）

这种方式使符号可以表示缺省字符集以外的数据（如阿拉伯字符、古斯拉夫字符、希腊

字母等)，以及其他解释（如用一定的压缩方式表示的数据）或者对行业特点的需要进行编码。

7.3 符号结构

每个 QR 码符号由名义上的正方形模块构成，组成一个正方形阵列，它由编码区域和包括寻象图形、分隔符、定位图形和校正图形在内的功能图形组成。功能图形不能用于数据编码。符号的四周由空白区包围。图 2 为 QR 码版本 7 符号的结构图。

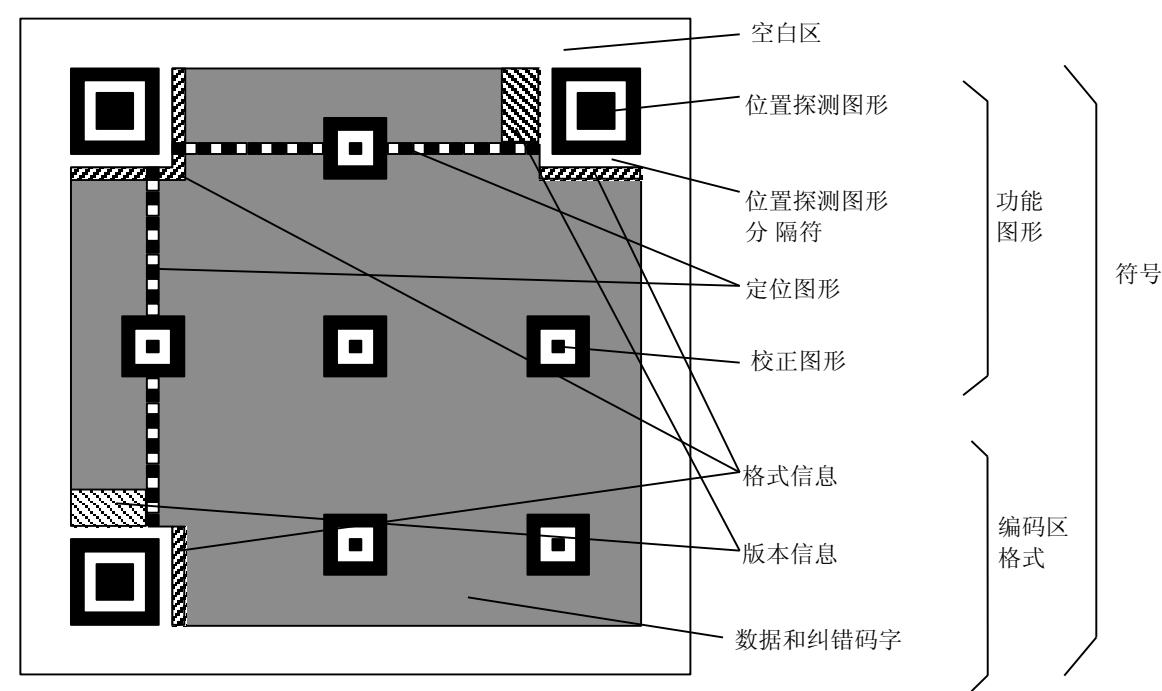


图 2 QR 码符号的结构

7.3.1 符号版本和规格

QR 码符号共有 40 种规格，分别为版本 1、版本 2……版本 40。版本 1 的规格为 21 模块×21 模块，版本 2 为 25 模块×25 模块，以此类推，每一版本符号比前一版本每边增加 4 个模块，直到版本 40，规格为 177 模块×177 模块。图 3 至 8 为版本 1，2，6，7，14，21 和 40 的符号结构。

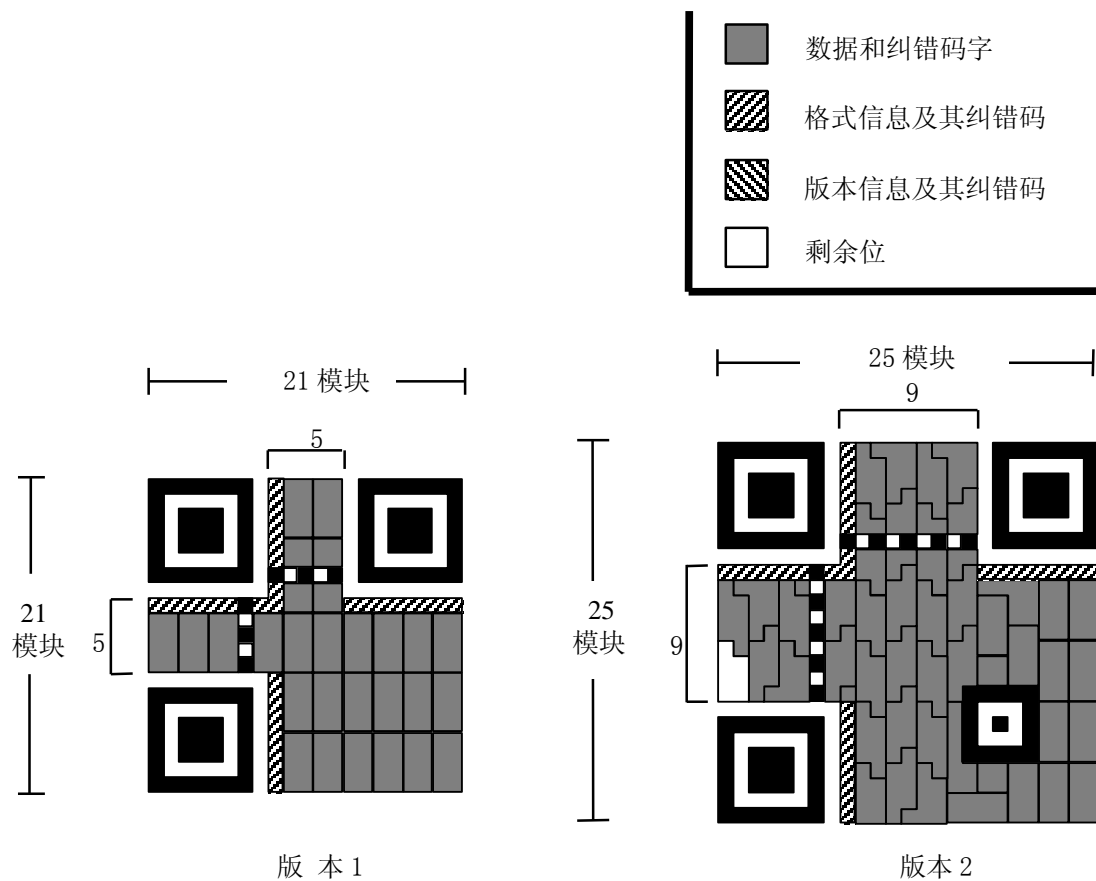
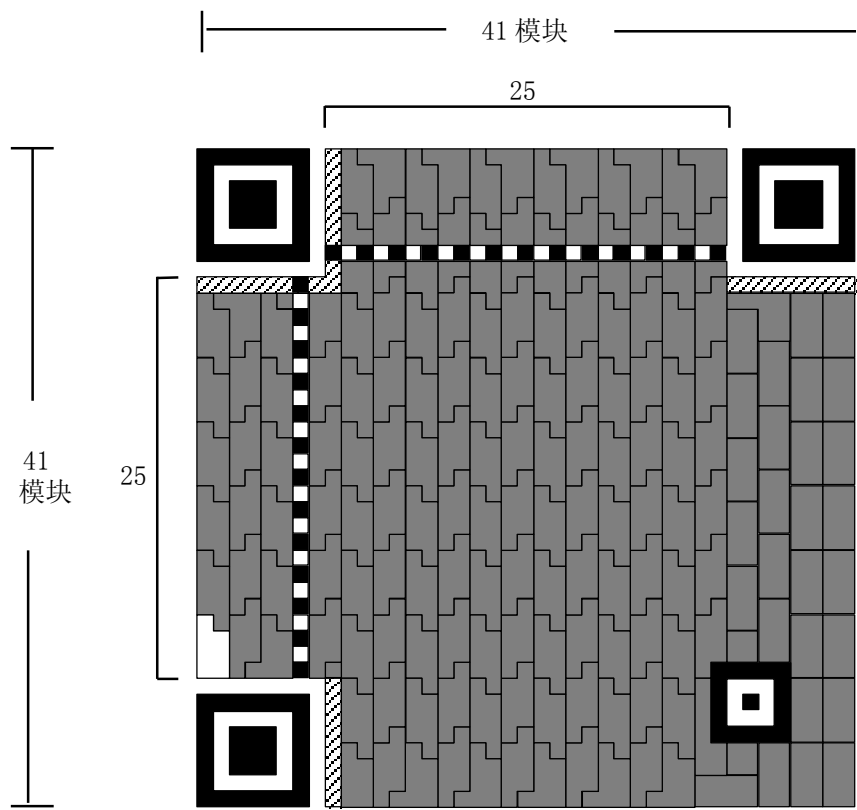
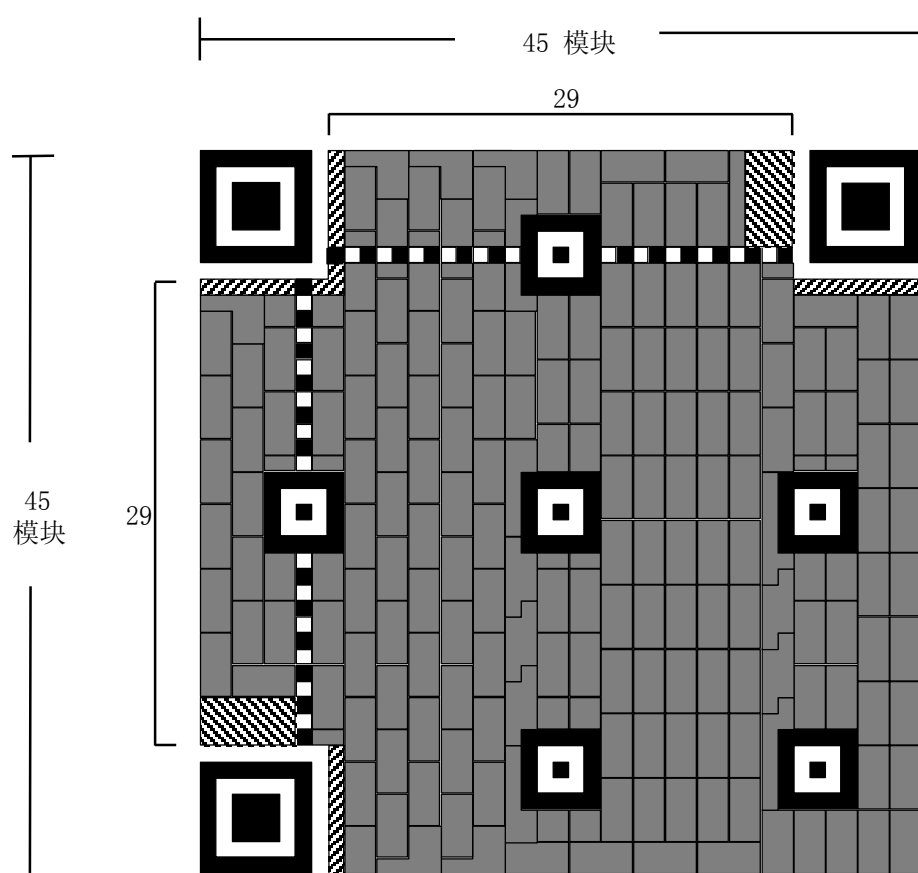


图 3 版本 1 和版本 2 的符号



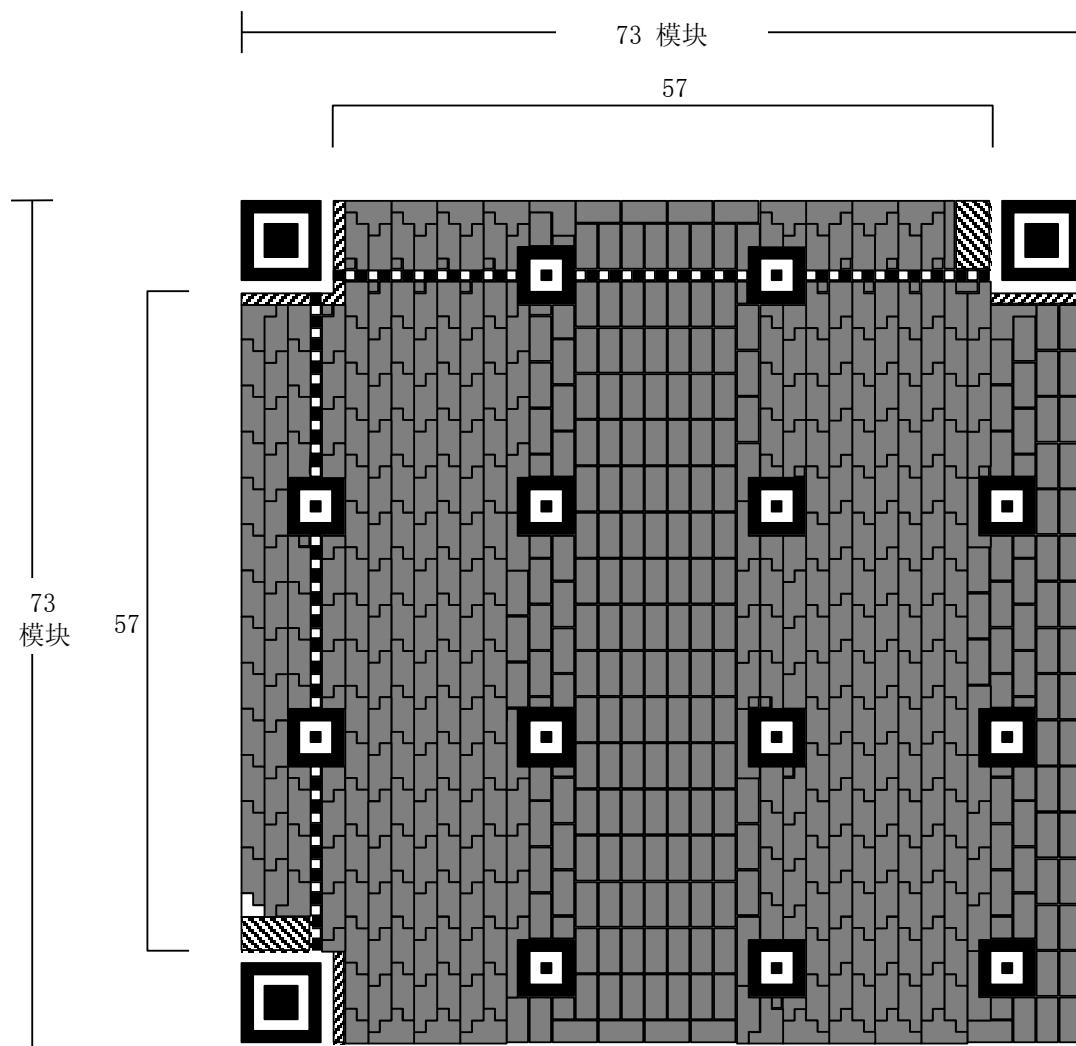
版本 6

图 4 版本 6 的符号



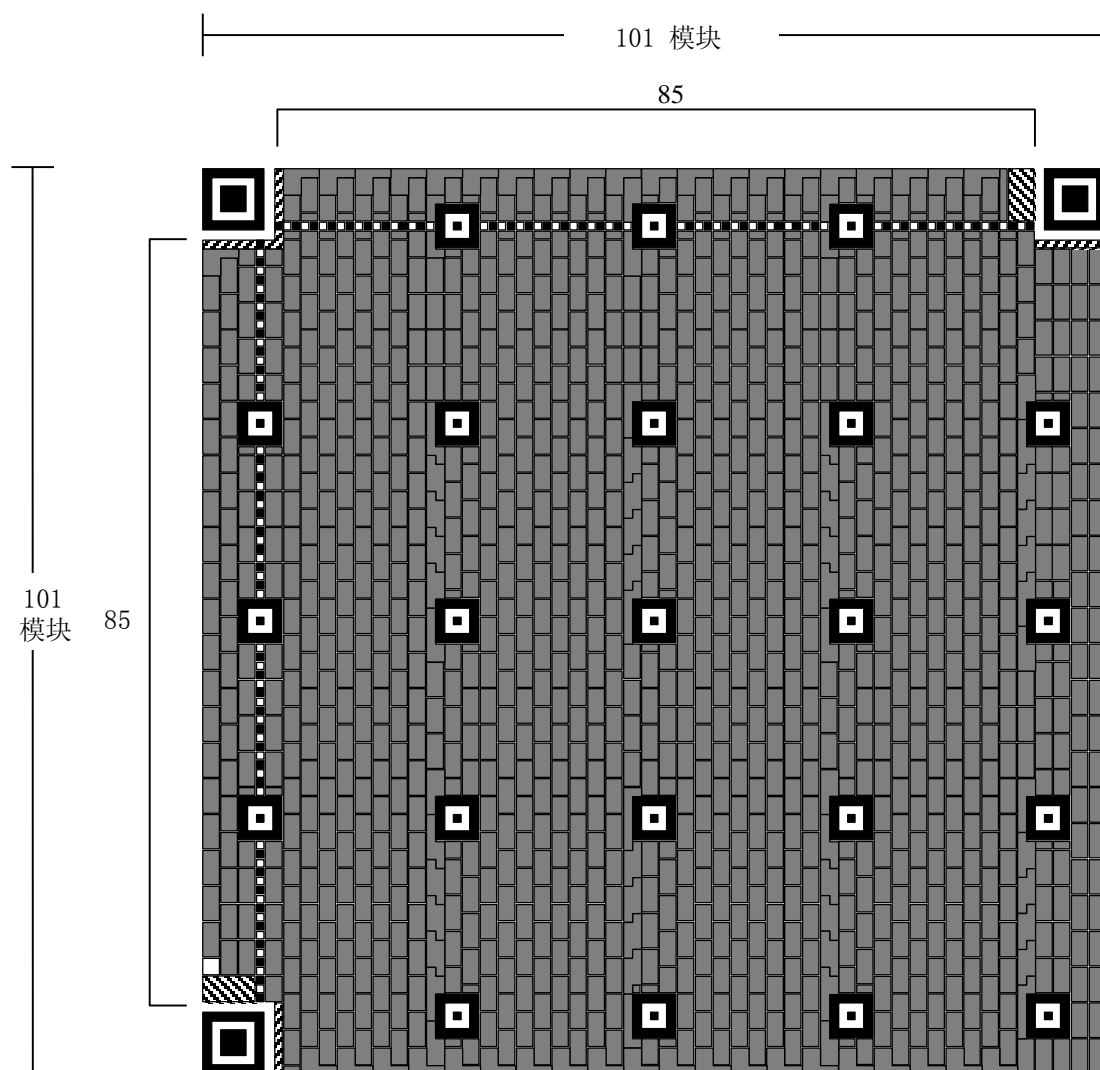
版本 7

图 5 版本 7 的符号



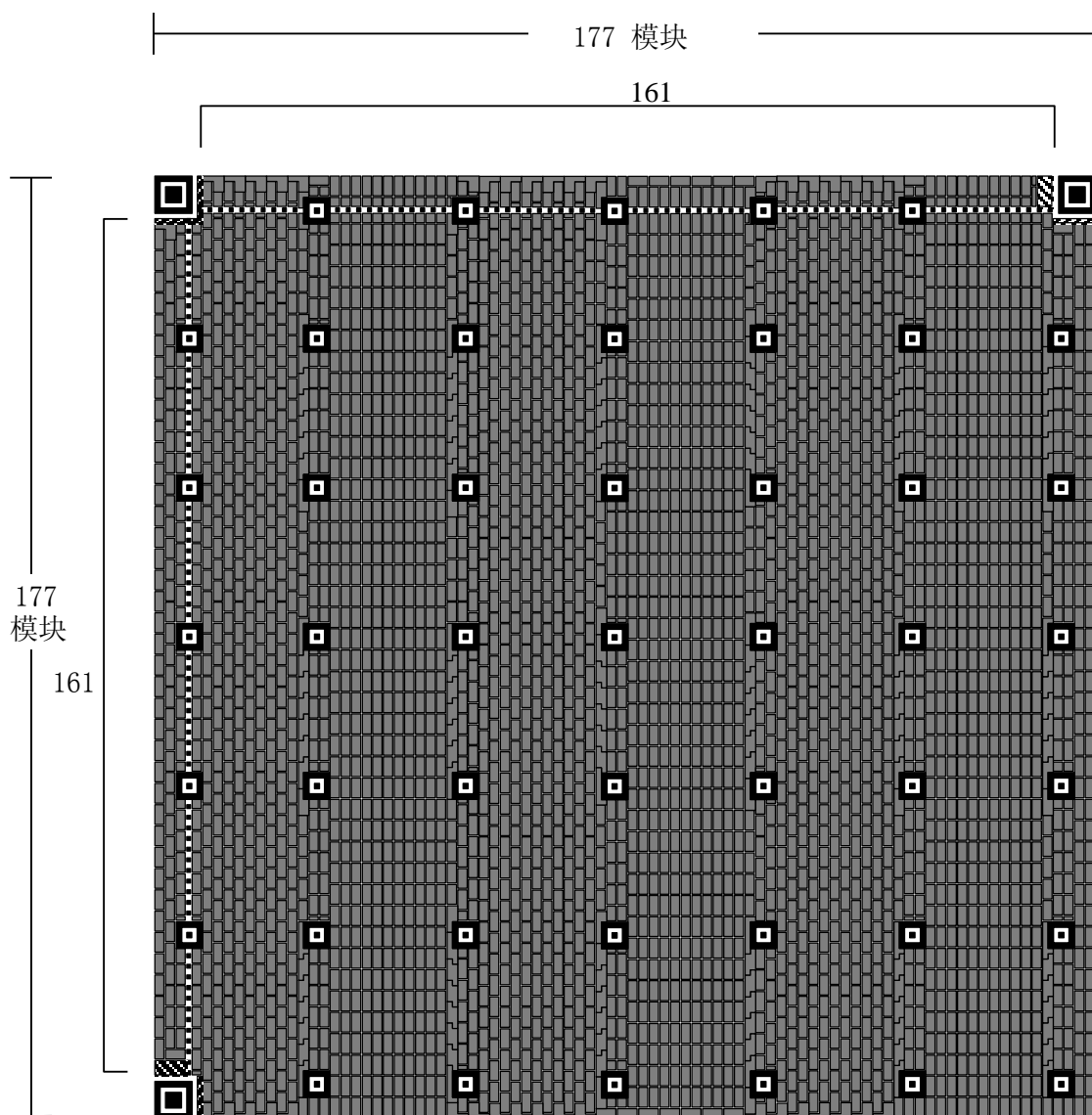
版本 14

图 6 版本 14 的符号



版本 21

图 7 版本 21 的符号



版本 40

图 8 版本 40 的符号

7.3.2 寻象图形

寻象图形包括三个相同的位置探测图形，分别位于符号的左上角、右上角和左下角，如图 2 所示。每个位置探测图形可以看作是由 3 个重叠的同心正方形组成，它们分别为 7×7 个深色色模块、5×5 个浅模块和 3×3 个深色模块。如图 9 所示，位置探测图形的模块宽度比为 1: 1: 3: 1: 1。符号中其他地方遇到类似图形的可能性极小，因此可以在视场中迅速地识别可能的 QR 码符号。识别组成的寻象图形的三个位置探测图形，可以明确地确定视场中符号的位置和方向。

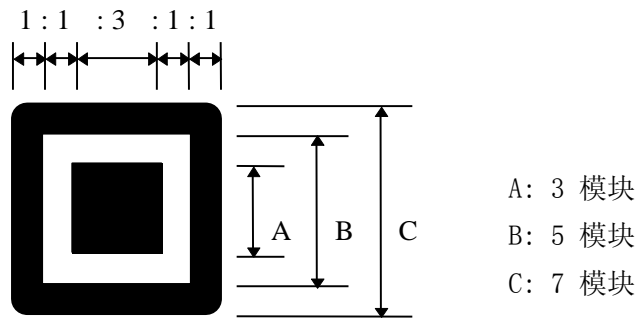


图 9 位置探测图形的结构

7.3.3 分隔符

在每个位置探测图形和编码区域之间有宽度为 1 个模块的分隔符，如图 2 所示，它全部由浅色模块组成。

7.3.4 定位图形

水平和垂直定位图形分别为一个模块宽的一行和一列，由深色浅色模块交替组成，其开始和结尾都是深色模块。水平定位图形位于上部的两个位置探测图形之间，符号的第 6 行。垂直定位图形位于左侧的两个位置探测图形之间，符号的第 6 列。它们的作用是确定符号的密度和版本，提供决定模块坐标的基准位置。

7.3.5 校正图形

每个校正图形可看作是 3 个重叠的同心正方形，由 5×5 个深色模块， 3×3 个浅色模块以及位于中心的一个深色模块组成。校正图形的数量视符号的版本号而定，在模式 2 的符号中，版本 2 以上（含版本 2）的符号均有校正图形，详见附录 E。

7.3.6 编码区域

编码区域包括表示数据码字、错误纠正码字、版本信息和格式信息的符号字符。符号字符的详细内容见 8.7.1，格式信息的详细内容见 8.9，版本信息的详细内容见 8.10。

7.3.7 空白区

空白区为环绕在符号四周的 4 个模块宽的区域，其反射率应与浅色模块相同。

8 要求符号表示方法

8.1 编码方法综述

第一步 数据分析

分析所输入的数据流，确定要进行编码的字符的类型。QR 码支持扩充解释，可以对与缺省的字符集不同的数据进行编码。QR 码包括几种不同的模式（见 8.3），以便高效地将不同的字符子集转换为符号字符。必要时可以进行模式之间的转换更高效地将数据转换，以便为二进制串。

选择所需的错误检测和纠正等级。如果用户没有指定所采用的符号版本，则选择与数据相适应的最小的版本。表 1 为全部符号版本及其容量。

表1 QR码各版本符号的数据容量

版本	每边的模块数 (A)	功能图形模块数 (B)	格式及版本信息模块数 (C)	除 C 以外的数据模块数 (D=A ² -B-C)	数据容量 [码字]* (E)	剩余位
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1079	134	7
6	41	267	31	1383	172	7
7	45	390	67	1568	196	0
8	49	398	67	1936	242	0
9	53	406	67	2336	292	0
10	57	414	67	2768	346	0
11	61	422	67	3232	404	0
12	65	430	67	3728	466	0
13	69	438	67	4256	532	0
14	73	611	67	4651	581	3
15	77	619	67	5243	655	3
16	81	627	67	5867	733	3
17	85	635	67	6523	815	3
18	89	643	67	7211	901	3
19	93	651	67	7931	991	3
20	97	659	67	8683	1085	3
21	101	882	67	9252	1156	4
22	105	890	67	10068	1258	4
23	109	898	67	10916	1364	4
24	113	906	67	11796	1474	4
25	117	914	67	12708	1588	4
26	121	922	67	13652	1706	4
27	125	930	67	14628	1828	4
28	129	1203	67	15371	1921	3
29	133	1211	67	16411	2051	3
30	137	1219	67	17483	2185	3
31	141	1227	67	18587	2323	3
32	145	1235	67	19723	2465	3
33	149	1243	67	20891	2611	3
34	153	1251	67	22091	2761	3
35	157	1574	67	23008	2876	0
36	161	1582	67	24272	3034	0
37	165	1590	67	25568	3196	0
38	169	1598	67	26896	3362	0
39	173	1606	67	28256	3532	0
40	177	1614	67	29648	3706	0

注 * 码字的长度为8位。

第二步 数据编码

对于采用的模式按照 8.4.1 至 8.4.5 所定义的规则，将数据字符转换为位流。在当需要进行模式转换时，在新的模式段开始前加入模式指示符进行模式转换。在数据序列后面加入终止符。将产生的位流分为每 8 位一个码字。必要时加入填充字符以填满按照版本要求的数据码字数。

第三步 纠错编码

按需要将码字序列分块（见表 13~22），以便按块生成相应的[错误纠正纠错](#)码字，并将其加入到相应的数据码字序列的后面。

第四步 构造最终信息

按 8.6 节（第三步）的描述，在每一块中置入数据和纠错码字，必要时加剩余位。

第五步 在矩阵中布置模块

将寻象图形、分隔符、定位图形、校正图形与码字模块一起放入矩阵。

第六步 掩模

依次将掩模图形用于符号的编码区域。评价结果，并选择其中使深色浅色模块比率最优且使不希望出现的图形最少化的结果。

第七步 格式和版本信息

生成格式和版本信息（如果用到时），形成符号。

8.2 数据分析

分析输入数据，确定其内容并且选择采用缺省的或者其它适当的 ECI 以及适当的模式，对每个序列编码，其详细描述见 8.4。从数字模式到日本汉字模式，每种模式的字符所需的位数依次增加。在符号中可以通过模式的转换使数据的位流长度最短，部分数据用一种模式编码可以比其它部分的编码效率更高。例如：数字序列后跟随字母数字序列。理论上说，用每个数据字符所需的位数最少的模式进行编码是最高效的，但每次模式转换时都需要有相关模式指示符和字符数量指示符等形式的附加开销，因此，对于较少的字符数，模式的转换并不总是能使位流总量最少，有关这方面的说明见附录 H。同时，由于各个版本的容量的增加是步进的，所以不一定在任何情况下都要达到最高的转换效率。

8.3 模式

下列模式是根据缺省的 ECI 有关的字符值与任务定义的。如果采用其他的 ECI，那么选择最佳的压缩模式就应使用字节值而不是具体的字符任务。例如：如果数据字节序列的值在 30_{HEX} 至 39_{HEX} 之间，应使用数字模式。这种情况下，用字节值相应的缺省的数字或者字母数字进行压缩。

8.3.1 扩充解释（ECI）模式

扩充解释（ECI）协议允许输出的数据流有与缺省的字符集不同的解释。ECI 协议在一些码制中有一致的定义。QR 码支持 4 大类解释。

- a) 国际字符集（或码页）
- b) 用于诸如加密或压缩等一般目的的解释
- c) 闭环系统中用户自定义的解释
- d) 无缓冲模式中用于结构链接的控制信息

ECI 协议在 AIM ECI 规范中有全面的定义，协议提供了一个对印刷前和译码后的字节值进行详细解释的方法。

QR 码的缺省解释是 ECI 000020，表示 JIS8 和转换 JIS 字符集。

8.3.2 数字模式

数字模式对十进制数字 0~9（ASCII 值 30_{HEX} 至 39_{HEX}）编码，通常的密度为每 10 位表示 3 个字符。

8.3.3 字母数字模式

字母数字模式对 45 个字符的字符集进行编码，即：10 个数字 0~9（ASCII 值 30_{HEX} 至 39_{HEX}），26 个字母字符 A~Z（ASCII 值 41_{HEX} 至 5A_{HEX}）以及 9 个符号 SP、\$、%、*、+、-、.、/、:（ASCII 值分别为 20_{HEX}，24_{HEX}，25_{HEX}，2A_{HEX}，2B_{HEX}，2D_{HEX}，2E_{HEX}，2F_{HEX}，3A_{HEX}）。通常情况下，两个输入字符用 11 位表示。

8.3.4 8 位字节模式

8 位字节模式用于表示与 JIS X 0201 一致的 8 位拉丁/假名字符集（字符值为 00_{HEX} 至 FF_{HEX}），其编码密度为每个字符 8 位。

8.3.5 日本汉字模式

日本汉字模式用于表示与基于 JIS X 0208 的转换 JIS 系统一致的日本汉字字符。转换 JIS 值由 JIS X 0208 转换而来。具体内容见 JIS X 0208 附录 1 转换代码表示法。每个双字节字符由 13 位二进制码字表示。

8.3.6 中国汉字模式

8.3.7 混合模式

QR 码符号可以包含以 8.3.1 到 8.3.5 中描述的多种模式所表示的混合数据序列。

用混合模式表示已知数据的最高效方法的选择方法见附录 H。

8.3.8 结构链接模式

结构链接模式用于把一个数据文件分开表示为多个 QR 码符号的序列，要求所有的符号可以识读并且数据可以按正确的顺序重新建立。每个符号都要有一个结构链接头，以标识这个序列的长度及该符号在其中的位置，并且检验是否所有识读的符号属于同一个文件。结构链接模式的编码详见第 9 章。

8.3.9 FNC1 模式

FNC1 模式用于表示按 UCC/EAN 应用标识标准或国际 AIM 协会已经同意的具体行业标准格式化的数据。

8.4 数据编码

输入的数据转变为一个位流，如果最开始的 ECI 不是缺省的 ECI，其前面要有 ECI 标头，后面为一个或多个不同模式的段，如果以缺省的 ECI 开始，位流的开头为第一个模式的指示符。

ECI 标头（如果有）应包含如下内容：

- ECI 模式指示符（4 位）
- ECI 指定符（8，16 或 24 位）

位流的其余部分的第一段由以下内容组成，它们包含：

- 模式指示符（4 位）
- 字符计数指示符
- 数据位流

ECI 标头由 ECI 模式指示符的最高位开始，以 ECI 指定符的最低位结束。每个模式段以模式指示符的最高位开始，以数据位流的最低位结束。由于段的长度已经由采用模式的规则以及数据字符数明确地确定，段与段之间没有特定的分隔。

在 8.4.1 至 8.4.6 中定义了在给定的模式中对输入的数据序列编码的步骤。表 2 定义了每个模式的模式指示符。表 3 定义了采用不同模式和符号版本的字符计数指示符的长度。

表2 模式指示符

模式	指示符
ECI	0111
数字	0001
字母数字	0010
8 位字节	0100
日本汉字	1000
中国汉字	1101
结构链接	0011
FNC1	0101（第一位置）
	1001（第二位置）
终止符（信息结尾）	0000

表3 字符计数指示符的位数

版本	数字 模式	字母数字 模式	8 位字节 模式	日本汉字 模式	中国汉字 模式
1~9	10	9	8	8	
10~26	12	11	16	10	
27~40	14	13	16	12	

整个符号的结束由 4 位终止符 0000 表示，当符号数据位流后所余的容量不足 4 位时，终止符将被截短。终止符本身不是模式指示符。

8.4.1 扩充解释（ECI）模式

本模式用于按可选的字节值解释（例如不同的字符集）对某类数据进行编码，可选的字节值解释应符合 AIM ECI 规范中对该类数据进行预处理的规则。模式用模式指示符 0111 引入。不需要引入 QR 码缺省扩充解释（ECI 头 000020，对应于 JIS8/转换 JIS 字符集），尤其是在任何符号的开始。

ECI 只能用于识读者可以传送符号标识的情况，不能传送符号标识的识读者无法从包含 ECI 的符号中传输数据。

输入的 ECI 数据需要编码系统作为一系列 8 位字节的值进行处理。

ECI 序列中的数据可以一种或几种模式对其字节值进行最高效编码，而不必考虑其实际意义。例如：值为 30_{HEX} 到 39_{HEX} 的数据序列可以当作一个数字 0-9 序列，用数字模式进行编码（见 8.4.2），即使也许实际上它并不表示数字数据。字符计数指示符由字节的数量决定（在日本汉字模式中为双字节、中国汉字模式中为双字节）。

8.4.1.1 ECI 指定符

每个扩充解释被分配一个 6 位的任务号，它为紧接在 QR 符号的 ECI 模式指示符后的 1 个、2 个或 3 个码字。其编码规则见表 4。ECI 模式指示符编码为 ASCII/JIS8 字符 5C_{HEX} (\ 或 ISO 646 IRV 中的反斜线, ¥ 或 JIS8 中的日圆符号), 其后跟随 6 位任务号。如果 ASCII/JIS8 字符 5C_{HEX} 本身是作为数据的内容, 在应用 ECI 协议的符号中编码前, 应在数据串中重复该字符。

表4 ECI任务号的编码

ECI 任务号	码字数	码字值
000000 至 000127	1	0bbbbbbb
000000 至 016383	2	10bbbbbb bbbbbbbb
000000 至 999999	3	110bbbb bbbbbbbb bbbbbbbb
		b……b 是 ECI 任务号的二进制值

在[解码译码](#)时, 第一个 ECI 指定符码字 (ECI 模式中模式指示符后的码字) 的二进制图形, 确定 ECI 指定符序列的长度。第一个 “0” 前面的 “1” 的个数量表示 ECI 任务号的第一个码字后的附加码字的数量。第一个 “0” 后面的各位, 是 ECI 任务号的二进制表示。较低的 ECI 任务号有多种编码方式, 最短的方式为首选的。

例如: 假设编码的数据为希腊字母, 字符集为 ISO 8859-7 (ECI000009), 符号版本为 1-H。

要编码的数据	ΑΒΓΔΕ: (字符值 A1 _{HEX} , A2 _{HEX} , A3 _{HEX} , A4 _{HEX} , A5 _{HEX})
符号中的位序列:	
ECI 模式指示符	0111
ECI 任务号 (000009)	00001001
模式指示符 (8 位字节)	0100
字符计数指示符 (5)	00000101
数据:	10100001 10100010 10100011 10100100 10100101
最终的位串:	0111 00001001 0100 00000101 10100001 10100010 10100011 10100100 10100101

该数据译码后传送的例子见 15.2 节。

8.4.1.2 多种 ECI

请参阅 AIM ECI 规范中规定的 ECI 数据段中有关 ECI 指定符的作用的规则。例如, 已经应用某一字符集 ECI 的数据也可用一个与初始的 ECI 共存的非字符集 ECI 加密或压缩, 或者第二个 ECI 可以取消第一个 ECI 并开始一个新的 ECI 段。在数据中出现任何 ECI 指定符都要按 8.4.1.1 在 QR 码符号中编码并开始一个新的模式段。

8.4.1.3 ECI 和结构链接

引入任何 ECI, 必须遵从上述规则及 AIM 的 ECI 规范, 直到编码数据的结束或者出现另一个 ECI (由模式指示符 0111 表示)。如果 ECI 编码的数据以结构链接的模式延伸到两个或更多的符号, 那么, 在每一个继续使用该 ECI 模式的后续符号中, 需提供由 ECI 模式提

示符和 ECI 指定符组成的 ECI 标头, 该标头紧跟在结构链接头之后。

8.4.2 数字模式

将输入的数据每三位分为一组, 将每组数据转换为 10 位二进制数。如果所输入的数据的位数不是 3 的整数倍, 所余的 1 位或 2 位数字应分别转换为 4 位或 7 位二进制数。将二进制数据连接起来并在前面加上模式指示符和字符计数指示符。数字模式中字符计数指示符如表 3 中定义的有 10、12 或 14 位。输入的数据字符的数量转换为 10、12 或 14 位二进制数后, 放置在模式指示符之后, 二进制数据序列之前。

例 1 (符号版本 1-H)

输入的数据: 01234567

1) 分为 3 位一组: 012 345 67

2) 将每组转换为二进制: 012→0000001100

345→0101011001

67 →1000011

3) 将二进制数连接为一个序列: 0000001100 0101011001 1000011

4) 将字符计数指示符转换为二进制 (版本 1-H 为 10 位):

字符数为: 8→0000001000

5) 加入模式指示符 0001 以及字符计数指示符的二进制数据:

0001 0000001000 0000001100 0101011001 1000011

例 2 (符号版本 1-H)

输入的数据: 0123456789012345

1) 分为 3 位一组: 012 345 678 901 234 5

2) 将每组转换为二进制: 012→0000001100

345→0101011001

678→1010100110

901→1110000101

234→0011101010

5 →0101

3) 将二进制数连接为一个序列:

0000001100 0101011001 1010100110 1110000101 0011101010 0101

4) 将字符计数指示符转换为二进制 (版本 1-H 为 10 位):

字符数为: 16→0000010000

5) 加入模式指示符 0001 以及字符计数指示符的二进制数据:

0001 0000010000 0000001100 0101011001 1010100110 1110000101 0011101010 0101

数字模式中位流的长度计算公式如下:

$$B = 4 + C + 10 (D \text{ DIV } 3) + R$$

其中: B=位流的位数

C=字符计数指示符的位数 (根据表 3)

D=输入的数据字符数

$R=0$ 当 $(D \bmod 3) = 0$

$R=4$ 当 $(D \bmod 3) = 1$

$R=7$ 当 $(D \bmod 3) = 2$

8.4.3 字母数字模式

按照表 5，每个输入的字符赋予一个数值 V ，它的值为 0 到 44。

表5 字母数字模式的编码/译码表

字 符	值	字 符	值	字 符	值	字 符	值	字 符	值	字 符	值	字 符	值
0	0	6	6	C	12	I	18	O	24	U	30	SP	36
1	1	7	7	D	13	J	19	P	25	V	31	\$	37
2	2	8	8	E	14	K	20	Q	26	W	32	%	38
3	3	9	9	F	15	L	21	R	27	X	33	*	39
4	4	A	10	G	16	M	22	S	28	Y	34	+	40
5	5	B	11	H	17	N	23	T	29	Z	35	-	41

将输入的数据分为两个字符一组，用 11 位二进制表示。将前面字符的值乘以 45 与第二个字符的值相加，将所得的结果转换为 11 位二进制数。如果输入的数据的字符数不是 2 的整数倍，将最后一个字符编码为 6 位二进制数。将所得的二进制数据连接起来并在前面加上模式指示符和字符计数指示符，按表 3 的规定在字母数字模式中，字符计数指示符的长度为 9、11 或 13 位，其定义见表 3。将输入的字符数编码为 9、11 或 13 位二进制数，放在模式指示符之后，二进制数据序列之前。

例（符号版本 1-H）

输入的数据： AC-42

1) 根据表 5 查出字符的值： AC-41 → (10, 12, 41, 4, 2)

2) 将结果分为 2 个一组： (10, 12) (41, 4) (2)

3) 将每组数据转换为 11 位二进制数： (10, 12) $10 \times 45 + 12 \rightarrow 462 \rightarrow 00111001110$
 (41, 4) $41 \times 45 + 4 \rightarrow 1849 \rightarrow 11100111001$
 (2) $\rightarrow 2 \rightarrow 000010$

4) 二进制数据顺次连接： 00111001110 11100111001 000010

5) 将字符计数指示符转换为二进制（版本 1-H 为 9 位）：

输入的字符数 5 → 000000101

6) 在二进制数据前加上模式指示符 0010 和字符计数指示符：

0010 000000101 00111001110 11100111001 000010

字母数据模式的二进制位流位数由下式计算：

$$B = 4 + C + 11(D \text{ DIV } 2) + 6(D \text{ MOD } 2)$$

其中：B=位流的位数

C=字符计数指示符的位数（根据表 3）

D=输入的字符数

8.4.4 8 位字节模式

在本模式中，一个 8 位码字直接表示一个输入数据字符的 JIS8 字符值，即密度为每个字符 8 位，如表 6 所示。在非缺省的其它 ECI 中，它直接表示一个 8 位字节的值。

表6 JIS8 字符集编码/译码表

字符	值	字符	值	字符	值	字符	值	字符	值	字符	值	字符	值
NUL	00	SP	20	@	40	`	60		80	A0	タ	C0	E0
SOH	01	!	21	A	41	A	61		81	。	A1	チ	E1
STX	02	”	22	B	42	B	62		82	「	A2	ツ	E2
ETX	03	#	23	C	43	C	63		83	」	A3	テ	E3
EOT	04	\$	24	D	44	D	64		84	、	A4	ト	E4
ENQ	05	%	25	E	45	E	65		85	・	A5	ナ	E5
ACK	06	&	26	F	46	F	66		86	ヲ	A6	ニ	E6
BEL	07	'	27	G	47	G	67		87	ア	A7	ヌ	E7
BS	08	(28	H	48	h	68		88	イ	A8	ネ	E8
HT	09)	29	I	49	I	69		89	ウ	A9	ノ	E9
LF	0A	*	2A	J	4A	j	6A		8A	エ	AA	ハ	EA
VT	0B	+	2B	K	4B	k	6B		8B	オ	AB	ヒ	EB
FF	0C	,	2C	L	4C	l	6C		8C	ヤ	AC	フ	EC
CR	0D	-	2D	M	4D	m	6D		8D	ユ	AD	ヘ	ED
SO	0E	.	2E	N	4E	n	6E		8E	ヨ	AE	ホ	EE
SI	0F	/	2F	O	4F	o	6F		8F	ッ	AF	マ	EF
DLE	10	0	30	P	50	p	70		90	ー	B0	ミ	F0
DC1	11	1	31	Q	51	q	71		91	ア	B1	ム	F1
DC2	12	2	32	R	52	r	72		92	イ	B2	メ	F2
DC3	13	3	33	S	53	s	73		93	ウ	B3	モ	F3
DC4	14	4	34	T	54	t	74		94	エ	B4	ヤ	F4
NAK	15	5	35	U	55	u	75		95	オ	B5	ユ	F5
SYN	16	6	36	V	56	v	76		96	カ	B6	ヨ	F6
ETB	17	7	37	W	57	w	77		97	キ	B7	ラ	F7
CAN	18	8	38	X	58	x	78		98	ク	B8	リ	F8
EM	19	9	39	Y	59	y	79		99	ケ	B9	ル	F9
SUB	1A	:	3A	Z	5A	z	7A		9A	コ	BA	レ	FA
ESC	1B	;	3B	[5B	{	7B		9B	サ	BB	ロ	FB
FS	1C	<	3C	¥	5C		7C		9C	シ	BC	ワ	FC
GS	1D	=	3D]	5D	}	7D		9D	ズ	BD	ン	FD
RS	1E	>	3E	^	5E	—	7E		9E	セ	BE	。*	FE
US	1F	?	3F	_	5F	DEL	7F		9F	ソ	BF	。*	FF

注1 在JIS8字符集中，字节值80_{HEX}到 9F_{HEX}以及E0_{HEX}到 FF_{HEX}没有分配，为保留值。其中，有些值作为转换JIS字符集中的第一个字节，也可能用来区分JIS8和转换JIS字符集。详见JIS X 0208 附录1：转换编码表示。

注2 除5C_{HEX}和 7E_{HEX}之外，JIS8字符集中字节值00_{HEX}到7F_{HEX}应和ISO 646 IRV一致。

将二进制数据连接起来并在前面加上模式指示符和字符计数指示符。按表3的规定，8位字节模式的字符计数指示符为8位或16位，将输入字符数转换为8位或16位二进制数据放在模式指示符之后，二进制数据序列之前。

8位字节模式的位流的位数计算公式如下：

$$D = 4 + C + 8D$$

其中：B=位流的位数

C=字符计数指示符的位数（见表 3）

D=输入数据的字符数

8.4.5 日本汉字模式

在转换 JIS 系统中，两个字节表示一个日本汉字字符。字节的值由 JIS X 0208 的值转换而来。详见 JIS X 0208 附录 1——“编码字符的表示”。将输入数据字符按下面定义转换为 13 位二进制码字。随后将二进制数据连接起来并在前面加上模式指示符和字符计数指示符。日本汉字模式的字符计数指示符的位数按表 3 规定为 8、10 或 12 位，将字符计数指示符转换为相应的 8、10 或 12 位二进制数，放在模式指示符之后，二进制数据序列之前。

1. 对于转换 JIS 值为 8140_{HEX} 到 9FFC_{HEX} 之间的字符：

- 将转换 JIS 值减去 8140_{HEX}；
- 将高位字节乘以 C0_{HEX}；
- 将 b 的结果加上低位字节；
- 将结果转换为 13 位二进制串。

2. 转换 JIS 值为 E040_{HEX} 到 EBBF_{HEX} 之间的字符：

- 将转换 JIS 值减去 C140_{HEX}；
- 将高位字节乘以 C0_{HEX}；
- 将 b 的结果加上低位字节；
- 将结果转为 13 位二进制串。

例：

输入字符	“点”	“茗”
(转换 JIS 值)：	935F	E4AA
1) 减去 8140 或 C140	$935F - 8140 = 121F$	$E4AA - C140 = 236A$
2) 第一字节乘以 C0	$12 \times C0 = D80$	$23 \times C0 = 1A40$
3) 加上第二字节的值	$D80 + 1F = D9F$	$1A40 + 6A = 1AAA$
4) 转换为 13 为二进制数	$0D9F \rightarrow 0\ 1101\ 1001\ 1111$	$1AAA \rightarrow 1\ 1010\ 1010\ 1010$

3. 对于所有的字符：

- 在输入的数据字符的二进制队列前加上模式指示符（1000）和字符计数指示符的二进制表示（8，10 或 12 位）；

日本汉字模式的位流的位数由以下公式计算：

$$B = 4 + C + 13D$$

其中：B=位流的位数

C=字符计数指示符的位数（见表 3）

D=输入的数据字符数

8.4.6 中国汉字模式

8.4.7 混合模式

根据数据内容的要求或者为了增加编码密度，可以从一种模式转换到另一种模式来表

示数据，详见附录 H。每一段数据按 8.4.1 到 8.4.5 中给出的相应模式进行编码。基本结构为模式指示符/字符计数指示符/数据，其后紧跟下一段的模式指示符开始另一段。图 10 为有 N 段数据的结构。

段 1			段 2			段 n		
模式指示符 1	字符计数指示符	数据	模式指示符 2	字符计数指示符	数据	模式指示符 n	字符计数指示符	数据

图10 — 混合模式数据格式

8.4.8 FNC1 模式

有两种模式指示符，它们和 8.3.1 至 8.3.8 以及 8.4.1 至 8.4.6 规定的模式标志结合起来应用，标识表示按特定的行业或应用的格式化的信息的符号，它们（和有关的参数数据一起）放在模式指示符之前。如果使用 FNC1 模式，[解码译码器](#)必需按 15.1 和附录 F 中的规定传输符号标识符。

8.4.8.1 FNC1 在第一位置

该模式指示符标识表示按 UCC/EAN 应用标识标准格式化信息的符号。因此，它在符号中只用一次，并总是放在用于高效对数据进行编码的第一个模式指示符（数字，字母数字，8 位字节或日本汉字、中国汉字）之前，ECI 或者结构链接头之后。在 UCC/EAN 规范中要求 FNC1 字符（在使用这一特殊字符的其它码制中）用作数据段分隔（即在可变长度数据段的末尾）时，QR 码符号将在字母数字模式中用%字符，在 8 位字节模式中用字符 GS(ASCII/JIS8 值 29) 实现这一功能。如果在数据中也有%符号出现，则应按%%进行编码。译码器在这些符号中遇到%时，应按 ASCII/JIS8 值 29 进行传输；如果遇到%%，则按单个%字符传输。

例

输入的数据：

0104912345123459（应用标识符 01=UCC/EAN 物品编码，定长；数据 04912345123459）
15970331（应用标识符 15=保质期，格式 YYMMDD，定长；数据 1997 年 3 月 31 日）
30128（应用标识符 30=数量，不定长；数据 128）（需要分隔符）
10ABC123（应用标识符 10=批号，不定长；数据 ABC123）

数据编码为：049123451234591597033130128%10ABC123

符号中的位序列：

0101（模式指示符，意指 FNC1 在第一位置）
0001（模式指示符，数字模式）
0000011101（字符计数指示符，29）
〈数据：049123451234591597033130128〉
0010（模式指示符，字母数字模式）
000001001（字符计数指示符，9）
〈数据：%10ABC123〉

传送的数据（见 15.1 节和附录 F）

]Q301049123451234591597033130128<ASCII 29>10ABC123

带有%的数据的编码与传送实例：

输入的数据：123%

编码为：123%%

传送为：123%

8.4.8.2 FNC1 在第二位置

该模式指示符标识表示按 AIM 同意的特定行业或者特定应用规范格式化信息的符号。在它后面跟着一个一字节的码字，该码字的值是用于标识 AIM 认可的规范中的应用指示符的值。在这种情况下，它在符号中只使用一次，并总是放在用于高效对数据进行编码的第一个模式指示符（数字，字母数字，8 位字节或日本汉字、中国汉字）之前，ECI 或者结构连接头之后。应用指示符可以采用字符集（a-z，A-Z）中任何一个拉丁字母（用字符的 ASCII 值加 100）或者用一个两位数（直接用它的值）表示，并由译码器作为数据前的一个或两个字符传输。

例：（本例中的应用指示符 37 在本规范发布时还没有分配给任组织，本例中的数据内容纯粹是任意的）

应用指示符：37

输入的数据：AA1234BBB112text text text text<CR>

符号中的位序列：

1001（模式指示符，意指 FNC1 在第二位置）

00100101（应用指示符，37）

0010（模式指示符，字母数字模式）

000001100（字符计数指示符，12）

<数据位：AA1234BBB112>

0100（模式指示符，8 位字节模式）

00010100（字符计数指示符，20）

<数据位：text text text text<CR>>

传送的数据：]Q537 AA1234BBB112text text text text<CR>

8.4.9 终止符

符号的数据结尾由紧跟在最后一个模式段后面的终止符序列 0000 表示，当数据位流数量正好填满符号的容量时，它可以省略，或者当符号所余的容量不足 4 位时它可以截短。

8.4.10 位流到码字的转换

每个模式段的位流需要按顺序连接在一起，最后添加终止符，除非数据位流正好填满符号容量。所得的数据位流将被分为一个个码字；所有的码字的长度都是 8 位，如果位流长度最后一个码字不足 8 位，则用二进制值为 0 的填充位填充至 8 位，填充位应加在数据位流最后 1 位（最低位）的后面。然后按表 7 至 11 所定义的版本和纠错等级交替添加填充码字 11101100 和 00010001，将数据位流扩展，以填满符号的数据容量，所得结果的数据码字序列按 8.5 节进行处理加入纠错码字。为了正好填满符号容量，有些版本也许需要在信息的最后添加 3、4 或 7 个剩余位（全为 0），见表 1。

表7 版本1-8的符号字符数和数据容量

版本	纠错等级	数据码字数*	数据位数**	数据容量				
				数字	字母数字	8 位字节	日本汉字	中国汉字
1	L	19	152	41	25	17	10	
	M	16	128	34	20	14	8	
	Q	13	104	27	16	11	7	
	H	9	72	17	10	7	4	
2	L	34	272	77	47	32	20	
	M	28	224	63	38	26	16	
	Q	22	176	48	29	20	12	
	H	16	128	34	20	14	8	
3	L	55	440	127	77	53	32	
	M	44	352	101	61	42	26	
	Q	34	272	77	47	32	20	
	H	26	208	58	35	24	15	
4	L	80	640	187	114	78	48	
	M	64	512	149	90	62	38	
	Q	48	384	111	67	46	28	
	H	36	288	82	50	34	21	
5	L	108	864	255	154	106	65	
	M	86	688	202	122	84	52	
	Q	62	496	144	87	60	37	
	H	46	368	106	64	44	27	
6	L	136	1 088	322	195	134	82	
	M	108	864	255	154	106	65	
	Q	76	608	178	108	74	45	
	H	60	480	139	84	58	36	
7	L	156	1 248	370	224	154	95	
	M	124	992	293	178	122	75	
	Q	88	704	207	125	86	53	
	H	66	528	154	93	64	39	
8	L	194	1 552	461	279	192	118	
	M	154	1 232	365	221	152	93	
	Q	110	880	259	157	108	66	
	H	86	688	202	122	84	52	

注1：所有码字的长度为8位

注 2：数据位的数量包括模式指示符和字符计数指示符的位数。

表8 版本9-16的符号字符数和数据容量

版本	纠错等级	数据码字数*	数据位数**	数据容量				
				数字	字母数字	8 位字节	日本汉字	中国汉字
9	L	232	1 856	552	335	230	141	
	M	182	1 456	432	262	180	111	
	Q	132	1 056	312	189	130	80	
	H	100	800	235	143	98	60	
10	L	274	2 192	652	395	271	167	
	M	216	1 728	513	311	213	131	
	Q	154	1 232	364	221	151	93	
	H	122	976	288	174	119	74	
11	L	324	2 592	772	468	321	198	
	M	254	2 032	604	366	251	155	
	Q	180	1 440	427	259	177	109	
	H	140	1 120	331	200	137	85	
12	L	370	2 960	883	535	367	226	
	M	290	2 320	691	419	287	177	
	Q	206	1 648	489	296	203	125	
	H	158	1 264	374	227	155	96	
13	L	428	3 424	1 022	619	425	262	
	M	334	2 672	796	483	331	204	
	Q	244	1 952	580	352	241	149	
	H	180	1 440	427	259	177	109	
14	L	461	3 688	1 101	667	458	282	
	M	365	2 920	871	528	362	223	
	Q	261	2 088	621	376	258	159	
	H	197	1 576	468	283	194	120	
15	L	523	4 184	1 250	758	520	320	
	M	415	3 320	991	600	412	254	
	Q	295	2 360	703	426	292	180	
	H	223	1 784	530	321	220	136	
16	L	589	4 712	1 408	854	586	361	
	M	453	3 624	1 082	656	450	277	
	Q	325	2 600	775	470	322	198	
	H	253	2 024	602	365	250	154	

注1：所有码字的长度为8位

注2：数据位的数量包括模式指示符和字符计数指示符的位数。

表9 版本17-24的符号字符数和数据容量

版本	纠错等级	数据码字数*	数据位数**	数据容量				
				数字	字母数字	8 位字节	日本汉字	中国汉字
17	L	647	5 176	1 548	938	644	397	
	M	507	4 056	1 212	734	504	310	
	Q	367	2 936	876	531	364	224	
	H	283	2 264	674	408	280	173	
18	L	721	5 768	1 725	1 046	718	442	
	M	563	4 504	1 346	816	560	345	
	Q	397	3 176	948	574	394	243	
	H	313	2 504	746	452	310	191	
19	L	795	6 360	1 903	1 153	792	488	
	M	627	5 016	1 500	909	624	384	
	Q	445	3 560	1 063	644	442	272	
	H	341	2 728	813	493	338	208	
20	L	861	6 888	2 061	1 249	858	528	
	M	669	5 352	1 600	970	666	410	
	Q	485	3 880	1 159	702	482	297	
	H	385	3 080	919	557	382	235	
21	L	932	7 456	2 232	1 352	929	572	
	M	714	5 712	1 708	1 035	711	438	
	Q	512	4 096	1 224	742	509	314	
	H	406	3 248	969	587	403	248	
22	L	1 006	8 048	2 409	1 460	1 003	618	
	M	782	6 256	1 872	1 134	779	480	
	Q	568	4 544	1 358	823	565	348	
	H	442	3 536	1 056	640	439	270	
23	L	1 094	8 752	2 620	1 588	1 091	672	
	M	860	6 880	2 059	1 248	857	528	
	Q	614	4 912	1 468	890	611	376	
	H	464	3 712	1 108	672	461	284	
24	L	1 174	9 392	2 812	1 704	1 171	721	
	M	914	7 312	2 188	1 326	911	561	
	Q	664	5 312	1 588	963	661	407	
	H	514	4 112	1 228	744	511	315	

注1: 所有码字的长度为8位

注2: 数据位的数量包括模式指示符和字符计数指示符的位数。

表10 版本25-32的符号字符数和数据容量

版本	纠错等级	数据码字数 [*]	数据位数 ^{**}	数据容量				
				数字	字母数字	8 位字节	日本汉字	中国汉字
25	L	1 276	10 208	3 057	1 853	1 273	784	
	M	1 000	8 000	2 395	1 451	997	614	
	Q	718	5 744	1 718	1 041	715	440	
	H	538	4 304	1 286	779	535	330	
26	L	1 370	10 960	3 283	1 990	1 367	842	
	M	1 062	8 496	2 544	1 542	1 059	652	
	Q	754	6 032	1 804	1 094	751	462	
	H	596	4 768	1 425	864	593	365	
27	L	1 468	11 744	3 517	2 132	1 465	902	
	M	1 128	9 024	2 701	1 637	1 125	692	
	Q	808	6 464	1 933	1 172	805	496	
	H	628	5 024	1 501	910	625	385	
28	L	1 531	12 248	3 669	2 223	1 528	940	
	M	1 193	9 544	2 857	1 732	1 190	732	
	Q	871	6 968	2 085	1 263	868	534	
	H	661	5 288	1 581	958	658	405	
29	L	1 631	13 048	3 909	2 369	1 628	1 002	
	M	1 267	10 136	3 035	1 839	1 264	778	
	Q	911	7 288	2 181	1 322	908	559	
	H	701	5 608	1 677	1 016	698	430	
30	L	1 735	13 880	4 158	2 520	1 732	1 066	
	M	1 373	10 984	3 289	1 994	1 370	843	
	Q	985	7 880	2 358	1 429	982	604	
	H	745	5 960	1 782	1 080	742	457	
31	L	1 843	14 744	4 417	2 677	1 840	1 132	
	M	1 455	11 640	3 486	2 113	1 452	894	
	Q	1 033	8 264	2 473	1 499	1 030	634	
	H	793	6 344	1 897	1 150	790	486	
32	L	1 955	15 640	4 686	2 840	1 952	1 201	
	M	1 541	12 328	3 693	2 238	1 538	947	
	Q	1 115	8 920	2 670	1 618	1 112	684	
	H	845	6 760	2 022	1 226	842	518	

注1：所有码字的长度为8位

注2：数据位的数量包括模式指示符和字符计数指示符的位数。。

表11 版本33-40的符号字符数和数据容量

版本	纠错等级	数据码字数*	数据位数**	数据容量				
				数字	字母数字	8 位字节	日本汉字	中国汉字
33	L	2 071	16 568	4 965	3 009	2 068	1 273	
	M	1 631	13 048	3 909	2 369	1 628	1 002	
	Q	1 171	9 368	2 805	1 700	1 168	719	
	H	901	7 208	2 157	1 307	898	553	
34	L	2 191	17 528	5 253	3 183	2 188	1 347	
	M	1 725	13 800	4 134	2 506	1 722	1 060	
	Q	1 231	9 848	2 949	1 787	1 228	756	
	H	961	7 688	2 301	1 394	958	590	
35	L	2 306	18 448	5 529	3 351	2 303	1 417	
	M	1 812	14 496	4 343	2 632	1 809	1 113	
	Q	1 286	10 288	3 081	1 867	1 283	790	
	H	986	7 888	2 361	1 431	983	605	
36	L	2 434	19 472	5 836	3 537	2 431	1 496	
	M	1 914	15 312	4 588	2 780	1 911	1 176	
	Q	1 354	10 832	3 244	1 966	1 351	832	
	H	1 054	8 432	2 524	1 530	1 051	647	
37	L	2 566	20 528	6 153	3 729	2 563	1 577	
	M	1 992	15 936	4 775	2 894	1 989	1 224	
	Q	1 426	11 408	3 417	2 071	1 423	876	
	H	1 096	8 768	2 625	1 591	1 093	673	
38	L	2 702	21 616	6 479	3 927	2 699	1 661	
	M	2 102	16 816	5 039	3 054	2 099	1 292	
	Q	1 502	12 016	3 599	2 181	1 499	923	
	H	1 142	9 136	2 735	1 658	1 139	701	
39	L	2 812	22 496	6 743	4 087	2 809	1 729	
	M	2 216	17 728	5 313	3 220	2 213	1 362	
	Q	1 582	12 656	3 791	2 298	1 579	972	
	H	1 222	9 776	2 927	1 774	1 219	750	
40	L	2 956	23 648	7 089	4 296	2 953	1 817	
	M	2 334	18 672	5 596	3 391	2 331	1 435	
	Q	1 666	13 328	3 993	2 420	1 663	1 024	
	H	1 276	10 208	3 057	1 852	1 273	784	

注1：所有码字的长度为8位

注2：数据位的数量包括模式指示符和字符计数指示符的位数。

8.5 纠错

8.5.1 纠错容量

QR 码采用纠错算法生成一系列纠错码字，添加在数据码字序列后，使得符号可以在遇到损坏时不致丢失数据。纠错共有 4 个纠错等级，对应四种纠错容量，如表 12 所示。

表12 纠错等级

纠错等级	恢复的容量 % (近似值)
L	7
M	15
Q	25
H	30

纠错等级的选择参见附录 I. 3。

纠错码字可以纠正两种类型的错误，拒读错误（错误码字的位置已知）和替代错误（错误码字位置未知）。一个拒读错误是一个没扫描到或无法译码的符号字符，一个替代错误是错误译码的符号字符。由于 QR 码是矩阵型符号，如果一个缺陷使深色模块变成浅色模块或将浅色模块变成深色模块，将符号字符错误地译码为表面上有效，但却是另一个不同的码字。这种数据替代错误需要两个纠错码字来纠正。

可纠正的替代和拒读错误的数量由下式给出：

$$e + 2t \leq d - p$$

其中：e=拒读错误数

t=替代错误数

d=纠错码字数

p=错误译码保护码字

例如：版本 6-H 符号中共有 172 个码字，其中有 112 个纠错码字（其余 60 个为数据码字）。这 112 个纠错码字可纠正 56 个替代错误或 112 个拒读错误，即符号纠错容量为 56/172 或 32.6%。

在上面的公式中，版本 1-L 符号的 p=3，版本 1-M 符号和版本 2-L 符号的 p=2，版本 1-H、1-Q 和 3-L 符号的 p=1，其他情况下 p=0。p>0（即 1，2 或 3）时，有 p 个码字作为错误检测码字，防止从错误超过纠错容量的符号传输数据。e 必须小于 d/2。例如，在版本 2-L 符号中码字总数为 44，其中数据码字为 34 个，纠错码字为 10 个。从表 13 中可以看出纠错容量为 4 个替代错误（e=0）。代入上面公式：

$$0 + (2 \times 4) = 10 - 2$$

这就意味着纠正 4 个替代错误只需要 8 个纠错码字，剩余的 2 个纠错码字可用于检测（不能纠正）其他错误，如果超过 4 个替代错误，则译码失败。

根据版本和纠错等级，将数据码字序列分为 1 个或多个块，对每一个块分别进行纠错运算。表 13-22 列出了每个版本、每个纠错等级的码字总数、纠错码字总数以及纠错块的结构和数量。

如果某一符号版本需要剩余位填充符号容量中剩余的模块，剩余位都应为 0。

表13 版本1-6的纠错特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
1	26	L	7	1	(26, 19, 2)†
		M	10	1	(26, 16, 4)†
		Q	13	1	(26, 13, 6)†
		H	17	1	(26, 9, 8)†
2	44	L	10	1	(44, 34, 4)†
		M	16	1	(44, 28, 8)
		Q	22	1	(44, 22, 11)
		H	28	1	(44, 16, 14)
3	70	L	15	1	(70, 55, 7)†
		M	26	1	(70, 44, 13)
		Q	36	2	(35, 17, 9)
		H	44	2	(35, 13, 11)
4	100	L	20	1	(100, 80, 10)
		M	36	2	(50, 32, 9)
		Q	52	2	(50, 24, 13)
		H	64	4	(25, 9, 8)
5	134	L	26	1	(134, 108, 13)
		M	48	2	(67, 43, 12)
		Q	72	2 2	(33, 15, 9) (34, 16, 9)
		H	88	2 2	(33, 11, 11) (34, 12, 11)
6	172	L	36	2	(86, 68, 9)
		M	64	4	(43, 27, 8)
		Q	96	4	(43, 19, 12)
		H	112	4	(43, 15, 14)

* (c, k, r): c =码字总数

k =数据码字数

r =纠错容量

† 纠错容量小于纠错码字数的一半，以减少错误[解码译码](#)的可能性

表14 版本7-10的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
7	196	L	40	2	(98, 78, 10)
		M	72	4	(49, 31, 9)
		Q	108	2 4	(32, 14, 9) (33, 15, 9)
		H	130	4 1	(39, 13, 13) (40, 14, 13)
8	242	L	48	2	(121, 97, 12)
		M	88	2 2	(60, 38, 11) (61, 39, 11)
		Q	132	4 2	(40, 18, 11) (41, 19, 11)
		H	156	4 2	(40, 14, 13) (41, 15, 13)
9	292	L	60	2	(146, 116, 15)
		M	110	3 2	(58, 36, 11) (59, 37, 11)
		Q	160	4 4	(36, 16, 10) (37, 17, 10)
		H	192	4 4	(36, 12, 12) (37, 13, 12)
10	346	L	72	2 2	(86, 68, 9) (87, 69, 9)
		M	130	4 1	(69, 43, 13) (70, 44, 13)
		Q	192	6 2	(43, 19, 12) (44, 20, 12)
		H	224	6 2	(43, 15, 14) (44, 16, 14)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表15 版本11-14的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
11	404	L	80	4	(101, 81, 10)
		M	150	1 4	(80, 50, 15) (81, 51, 15)
		Q	224	4 4	(50, 22, 14) (51, 23, 14)
		H	264	3 8	(36, 12, 12) (37, 13, 12)
12	466	L	96	2 2	(116, 92, 12) (117, 93, 12)
		M	176	6 2	(58, 36, 11) (59, 37, 11)
		Q	260	4 6	(46, 20, 13) (47, 21, 13)
		H	308	7 4	(42, 14, 14) (43, 15, 14)
13	532	L	104	4	(133, 107, 13)
		M	198	8 1	(59, 37, 11) (60, 38, 11)
		Q	288	8 4	(44, 20, 12) (45, 21, 12)
		H	352	12 4	(33, 11, 11) (34, 12, 11)
14	581	L	120	3 1	(145, 115, 15) (146, 116, 15)
		M	216	4 5	(64, 40, 12) (65, 41, 12)
		Q	320	11 5	(36, 16, 10) (37, 17, 10)
		H	384	11 5	(36, 12, 12) (37, 13, 12)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表16 版本15-18的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
15	655	L	132	5 1	(109, 87, 11) (110, 88, 11)
		M	240	5 5	(65, 41, 12) (66, 42, 12)
		Q	360	5 7	(54, 24, 15) (55, 25, 15)
		H	432	11 7	(36, 12, 12) (37, 13, 12)
16	733	L	144	5 1	(122, 98, 12) (123, 99, 12)
		M	280	7 3	(73, 45, 14) (74, 46, 14)
		Q	408	15 2	(43, 19, 12) (44, 20, 12)
		H	480	3 13	(45, 15, 15) (46, 16, 15)
17	815	L	168	1 5	(135, 107, 14) (136, 108, 14)
		M	308	10 1	(74, 46, 14) (75, 47, 14)
		Q	448	1 15	(50, 22, 14) (51, 23, 14)
		H	532	2 17	(42, 14, 14) (43, 15, 14)
18	901	L	180	5 1	(150, 120, 15) (151, 121, 15)
		M	338	9 4	(69, 43, 13) (70, 44, 13)
		Q	504	17 1	(50, 22, 14) (51, 23, 14)
		H	588	2 19	(42, 14, 14) (43, 15, 14)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表17 版本19-22的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
19	991	L	196	3 4	(141, 113, 14) (142, 114, 14)
		M	364	3 11	(70, 44, 13) (71, 45, 13)
		Q	546	17 4	(47, 21, 13) (48, 22, 13)
		H	650	9 16	(39, 13, 13) (40, 14, 13)
20	1 085	L	224	3 5	(135, 107, 14) (136, 108, 14)
		M	416	3 13	(67, 41, 13) (68, 42, 13)
		Q	600	15 5	(54, 24, 15) (55, 25, 15)
		H	700	15 10	(43, 15, 14) (44, 16, 14)
21	1 156	L	224	4 4	(144, 116, 14) (145, 117, 14)
		M	442	17	(68, 42, 13)
		Q	644	17 6	(50, 22, 14) (51, 23, 14)
		H	750	19 6	(46, 16, 15) (47, 17, 15)
22	1 258	L	252	2 7	(139, 111, 14) (140, 112, 14)
		M	476	17	(74, 46, 14)
		Q	690	7 16	(54, 24, 15) (55, 25, 15)
		H	816	34	(37, 13, 12)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表18 版本23-26的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
23	1 364	L	270	4 5	(151, 121, 15) (152, 122, 15)
		M	504	4 14	(75, 47, 14) (76, 48, 14)
		Q	750	11 14	(54, 24, 15) (55, 25, 15)
		H	900	16 14	(45, 15, 15) (46, 16, 15)
24	1 474	L	300	6 4	(147, 117, 15) (148, 118, 15)
		M	560	6 14	(73, 45, 14) (74, 46, 14)
		Q	810	11 16	(54, 24, 15) (55, 25, 15)
		H	960	30 2	(46, 16, 15) (47, 17, 15)
25	1 588	L	312	8 4	(132, 106, 13) (133, 107, 13)
		M	588	8 13	(75, 47, 14) (76, 48, 14)
		Q	870	7 22	(54, 24, 15) (55, 25, 15)
		H	1050	22 13	(45, 15, 15) (46, 16, 15)
26	1 706	L	336	10 2	(142, 114, 14) (143, 115, 14)
		M	644	19 4	(74, 46, 14) (75, 47, 14)
		Q	952	28 6	(50, 22, 14) (51, 23, 14)
		H	1110	33 4	(46, 16, 15) (47, 17, 15)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表19 版本27-30的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
27	1 828	L	360	8 4	(152, 122, 15) (153, 123, 15)
		M	700	22 3	(73, 45, 14) (74, 46, 14)
		Q	1 020	8 26	(53, 23, 15) (54, 24, 15)
		H	1 200	12 28	(45, 15, 15) (46, 16, 15)
28	1 921	L	390	3 10	(147, 117, 15) (148, 118, 15)
		M	728	3 23	(73, 45, 14) (74, 46, 14)
		Q	1 050	4 31	(54, 24, 15) (55, 25, 15)
		H	1 260	11 31	(45, 15, 15) (46, 16, 15)
29	2 051	L	420	7 7	(146, 116, 15) (147, 117, 15)
		M	784	21 7	(73, 45, 14) (74, 46, 14)
		Q	1 140	1 37	(53, 23, 15) (54, 24, 15)
		H	1 350	19 26	(45, 15, 15) (46, 16, 15)
30	2 185	L	450	5 10	(145, 115, 15) (146, 116, 15)
		M	812	19 10	(75, 47, 14) (76, 48, 14)
		Q	1 200	15 25	(54, 24, 15) (55, 25, 15)
		H	1 440	23 25	(45, 15, 15) (46, 16, 15)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表20 版本31-34的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
31	2 323	L	480	13 3	(145, 115, 15) (146, 116, 15)
		M	868	2 29	(74, 46, 14) (75, 47, 14)
		Q	1 290	42 1	(54, 24, 15) (55, 25, 15)
		H	1 530	23 28	(45, 15, 15) (46, 16, 15)
32	2 465	L	510	17	(145, 115, 15)
		M	924	10 23	(74, 46, 14) (75, 47, 14)
		Q	1 350	10 35	(54, 24, 15) (55, 25, 15)
		H	1 620	19 35	(45, 15, 15) (46, 16, 15)
33	2 611	L	540	17 1	(145, 115, 15) (146, 116, 15)
		M	980	14 21	(74, 46, 14) (75, 47, 14)
		Q	1 440	29 19	(54, 24, 15) (55, 25, 15)
		H	1 710	11 46	(45, 15, 15) (46, 16, 15)
34	2 761	L	570	13 6	(145, 115, 15) (146, 116, 15)
		M	1 036	14 23	(74, 46, 14) (75, 47, 14)
		Q	1 530	44 7	(54, 24, 15) (55, 25, 15)
		H	1 800	59 1	(46, 16, 15) (47, 17, 15)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表21 版本35-38的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
35	2 876	L	570	12 7	(151, 121, 15) (152, 122, 15)
		M	1 064	12 26	(75, 47, 14) (76, 48, 14)
		Q	1 590	39 14	(54, 24, 15) (55, 25, 15)
		H	1 890	22 41	(45, 15, 15) (46, 16, 15)
36	3 034	L	600	6 14	(151, 121, 15) (152, 122, 15)
		M	1 120	6 34	(75, 47, 14) (76, 48, 14)
		Q	1 680	46 10	(54, 24, 15) (55, 25, 15)
		H	1 980	2 64	(45, 15, 15) (46, 16, 15)
37	3 196	L	630	17 4	(152, 122, 15) (153, 123, 15)
		M	1 204	29 14	(74, 46, 14) (75, 47, 14)
		Q	1 770	49 10	(54, 24, 15) (55, 25, 15)
		H	2 100	24 46	(45, 15, 15) (46, 16, 15)
38	3 362	L	660	4 18	(152, 122, 15) (153, 123, 15)
		M	1 260	13 32	(74, 46, 14) (75, 47, 14)
		Q	1 860	48 14	(54, 24, 15) (55, 25, 15)
		H	2 220	42 32	(45, 15, 15) (46, 16, 15)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

表22 版本39-40的纠错的特性

版本	码字总数	纠错等级	纠错码字数	纠错的块数	每一块的纠错代码 *
39	3 532	L	720	20 4	(147, 117, 15) (148, 118, 15)
		M	1 316	40 7	(75, 47, 14) (76, 48, 14)
		Q	1 950	43 22	(54, 24, 15) (55, 25, 15)
		H	2 310	10 67	(45, 15, 15) (46, 16, 15)
40	3 706	L	750	19 6	(148, 118, 15) (149, 119, 15)
		M	1 372	18 31	(75, 47, 14) (76, 48, 14)
		Q	2 040	34 34	(54, 24, 15) (55, 25, 15)
		H	2 430	20 61	(45, 15, 15) (46, 16, 15)

* (c, k, r): c =码字总数
k =数据码字数
r =纠错容量

8.5.2 纠错码字的生成

将数据码字（必要时包括填充码字在内）按照表 13~22 分为相应数量的块，每一块分别计算出纠错码字并添加到数据码字后。

QR 码的多项式算法用位的模 2 算法和字节的模 100011101 算法。这是加罗瓦域 2^8 以 100011101 表示主模块多项式： $X^8+X^4+X^3+X^2+1$ 。

数据码字为多项式各项的系数，第一个数据码字为最高次项的系数，最低次项的系数是第一个纠错码字前的最后一个数据码字。

纠错码字是数据码字被纠错码多项式 $g(x)$ 除得的余数（见附录 A）。余数的最高次项系数为第一个纠错码字，最低次项系数为最后一个纠错码字，也是整个块的最后一个码字。

用于生成纠错码字的多项式有 31 个，在附录 A.1 中列出。

纠错算法可以用图 11 所示的除法电路来实现。寄存器 b_0 到 b_{k-1} 的初始值为 0。生成编码的状态有两个，在第一种状态，开关位置向下，数据码字同时经过电路以及输出，第一种状态在 n 个时钟脉冲后结束；在第二种状态（ $n+1 \cdots n+k$ 时钟脉冲），开关位置向上，通过保持输入为 0 顺序释放寄存器而生成纠错码字 $\xi_{k-1} \dots \xi_0$ 。

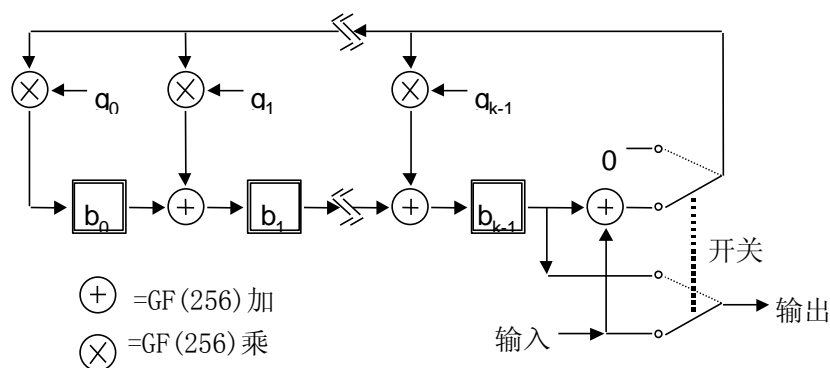


图 11 纠错码字编码电路

8.6 构造信息的最终码字序列

最终码字序列中的码字数应总是与表 7 - 12 和表 12 - 22 所列的符号能够表示的码字总数相同。

按如下步骤构造最终的码字序列（数据码字加上纠错码字，必要时加上剩余码字）。

1) 按表 13-22，根据版本和纠错等级将数据码字序列分为 n 块。

2) 对每一块，按照 8.5.2 和附录 A 计算相应块的纠错码字。

3) 依次将每一块的数据和纠错码字装配成最终的序列：数据块 1 的码字 1；数据块 2 的码字 1；数据块 3 的码字 1；以此类推至数据块 $n-1$ 的最后的码字；数据块 n 的最后的码字；随后，纠错块 1 的码字 1，纠错块 2 的码字 1，...以此类推至纠错块 $n-1$ 的最后的码字；纠错块 n 的最后的码字。QR 码符号所包含的数据和纠错块通常正好填满符号的码字容量，而在某些版本中，也许需要 3、4 或 7 个剩余位，添加在最终的信息流中以正好填满编码区域的模块数。

最短的数据块应在序列的最前面，所有的数据码字在第一个纠错码字的前面。例如，版本 5-H 的符号由 4 个数据和纠错块组成，前两个块分别包括 11 个数据码字和 22 个纠错码字，第 3、4 个块分别包括 12 个数据码字和 22 个纠错码字。在此符号中，字符的布置如下，表中的每一行对应一个块的数据码字（表示为 D_n ）和相应块的纠错码字（表示为 E_n ）；符号中字符的布置可以通过由上向下逐列读表中的各列得到。

	数据码字					纠错码字			
块 1	D1	D2	D11		E1	E2	E22
块 2	D12	D13	D22		E23	E24	E44
块 3	D23	D24	D33	D34	E45	E46	E66
块 4	D35	D36	D45	D46	E67	E68	E88

版本 5-H 符号的最终码字序列为：

$D_1, D_{12}, D_{23}, D_{35}, D_2, D_{13}, D_{24}, D_{36}, \dots D_{11}, D_{22}, D_{33}, D_{45}, D_{34}, D_{46}, E_1, E_{23}, E_{45}, E_{67}, E_2, E_{24}, E_{46}, E_{68}, \dots E_{22}, E_{44}, E_{66}, E_{88}$ 。如果需要，在最后的码字后面加上剩余位（0）。

8.7 码字在矩阵中的布置

8.7.1 符号字符表示

在 QR 码符号中有两种类型的符号字符：规则的和不规则的。它们的使用取决于它们在符号中的位置，以及与其他符号字符和功能图形的关系。

多数码字在符号中表示为规则的 2×4 个模块的排列。其排列有两种方式，垂直布置（2 个模块宽，4 个模块高）；如果需要改变方向，可以水平布置（4 个模块宽，2 个模块高）。当改变方向或紧靠校正图形或其他功能图形时，需用不规则符号字符。

8.7.2 功能图形的布置

按照与使用的版本相对应的模块数构成空白的正方形矩阵。在寻象图形、分隔符、定位图形以及校正图形相应的位置，填入适当的深色浅色模块。格式信息和版本信息的模块位置暂时空置，其具体位置见图 15 和 16，它们对所有版本都是相同的。附录 E 定义了校正图形的位置。

8.7.3 符号字符的布置

在 QR 符号的编码区域中，符号字符以 2 个模块宽的纵列从符号的右下角开始布置，并自右向左，且交替地从下向上或从上向下安排。下面给出了符号字符以及字符中位的布置原则。图 15、16 为使用这些规则的版本 2 和版本 7 的符号。

a) 位序列在纵列中的布置为从右到左，向上或向下应与符号字符的布置方向一致。

b) 每个码字的最高位（表示为位 7）应放在第一个可用的模块位置，以后的放在下一个模块的位置。如果布置的方向是向上的，则最高位占用规则模块字符的右下角的模块，布置的方向向下时为右上角。如果先前的字符结束于右侧的模块纵列，最高位可能占据不规则符号字符的左下角模块的位置（见图 14）。

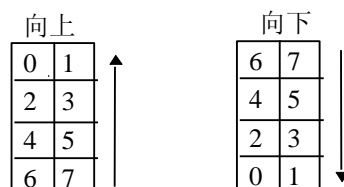


图 12 向上或向下的规则字符的位的布置

c) 如果符号字符的两个模块纵列同时遇到校正图形或定位图形的水平边界，可以在图形的上面或下面继续布置，如同编码区域是连续的一样。

d) 如果遇到符号字符区域的上或下边界（即符号的边缘，格式信息，版本信息或分隔符），码字中剩余的位应改变方向放在左侧的纵列中。



图 13 布置方向改变的符号字符位布置示例

e) 如果符号字符的右侧模块纵列遇到校正图形或版本信息占用的区域，位的布置形成不规则排列符号字符，在相邻校正图形或版本信息的单个纵列继续延伸。如果字符在可用于下一个字符的两列纵列之前结束，则下一个符号字符的首位放在单个纵列中。

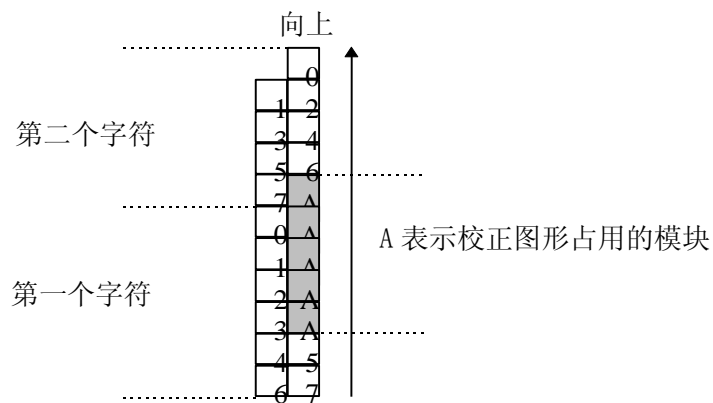


图 14 临近校正图形的位布置示例

还有另一种可供选择的符号字符布置方法，可得到相同的结果，将整个码字序列视为一个单独的位流，将其（最高位开始）按从右向左，按向上和向下的方向交替的布置于两个模块宽的纵列中。并跳过功能图形占用的区域，在纵列的顶部或底部改变方向，每一位应放在第一个可用的位置。

当符号的数据容量不能恰好分为整数个 8 位符号字符时，要用相应的剩余位（如表 1 中所示的 3, 4 或 7）填充符号的容量。在根据 8.8 节进行掩模以前，这些剩余位的值为 0。

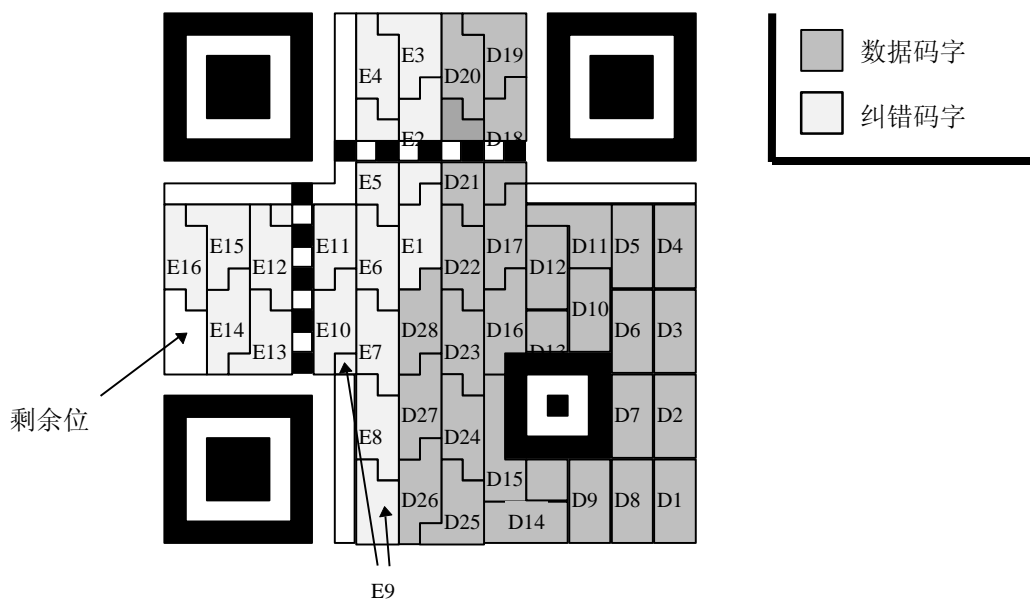


图 15 版本 2-M 符号的符号字符布置

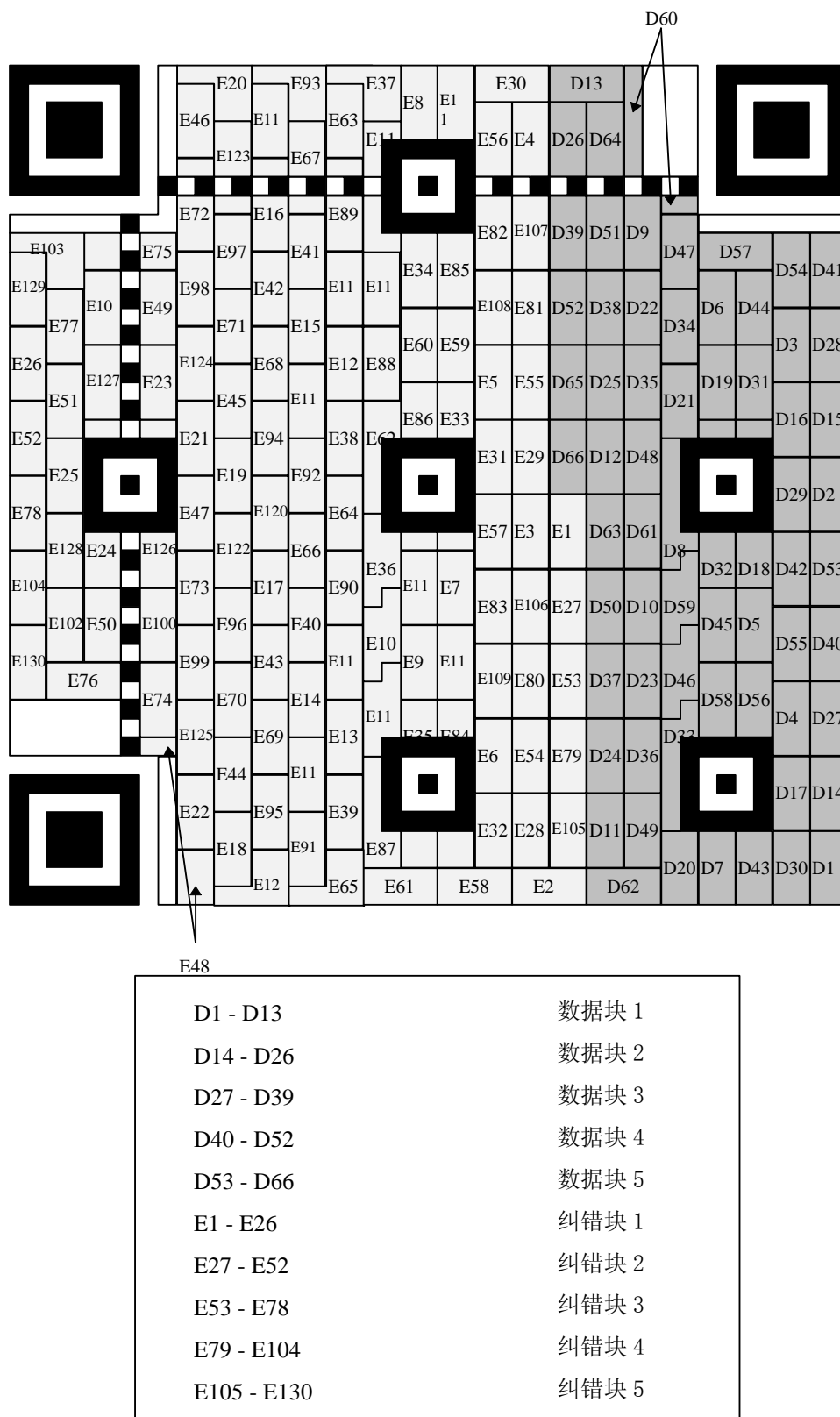


图 16 版本 7-H 符号的符号字符布置

8.8 掩模

为了 QR 码阅读的可靠性，最好均衡地安排深色与浅色模块。应尽可能避免位置探测图形的位图 1011101 出现在符号的其他区域。为了满足上述条件，应按以下步骤进行掩模。

- 1) 掩模不用于功能图形
- 2) 用多个矩阵图形连续地对已知的编码区域的模块图形（格式信息和版本信息除外）进行 XOR 操作。XOR 操作将模块图形依次放在每个掩模图形上，并将对应于掩模图形的深色模块的模块取反（浅色变成深色，或相反）。
- 3) 对每个结果图形的不合要求的部分记分，以评估这些结果。
- 4) 选择得分最低的图形。

8.8.1 掩模图形

表 23 给出了掩模图形的参考（放置于格式信息中的二进制参考）和掩模图形生成的条件。掩模图形是通过将编码区域（不包括为格式信息和版本信息保留的部分）内那些条件为真的模块定义为深色而产生的。所示的条件中， i 代表模块的行位置， j 代表模块的列位置， $(i, j) = (0, 0)$ 代表符号中左上角的位置。

表23 掩模图形的生成条件

掩模图形参考	条件
000	$(i + j) \bmod 2 = 0$
001	$i \bmod 2 = 0$
010	$j \bmod 3 = 0$
011	$(i + j) \bmod 3 = 0$
100	$((i \div 2) + (j \div 3)) \bmod 2 = 0$
101	$(i \bmod 2 + j \bmod 3) \bmod 2 = 0$
110	$((i \bmod 2 + j \bmod 3) \bmod 2) \bmod 2 = 0$
111	$((i \bmod 3 + (i+j) \bmod 2) \bmod 2) \bmod 2 = 0$

图 17 给出了版本 1 符号的所有的掩模图形，图 18 模拟了用掩模图形参考 000 到 111 的掩模结果。

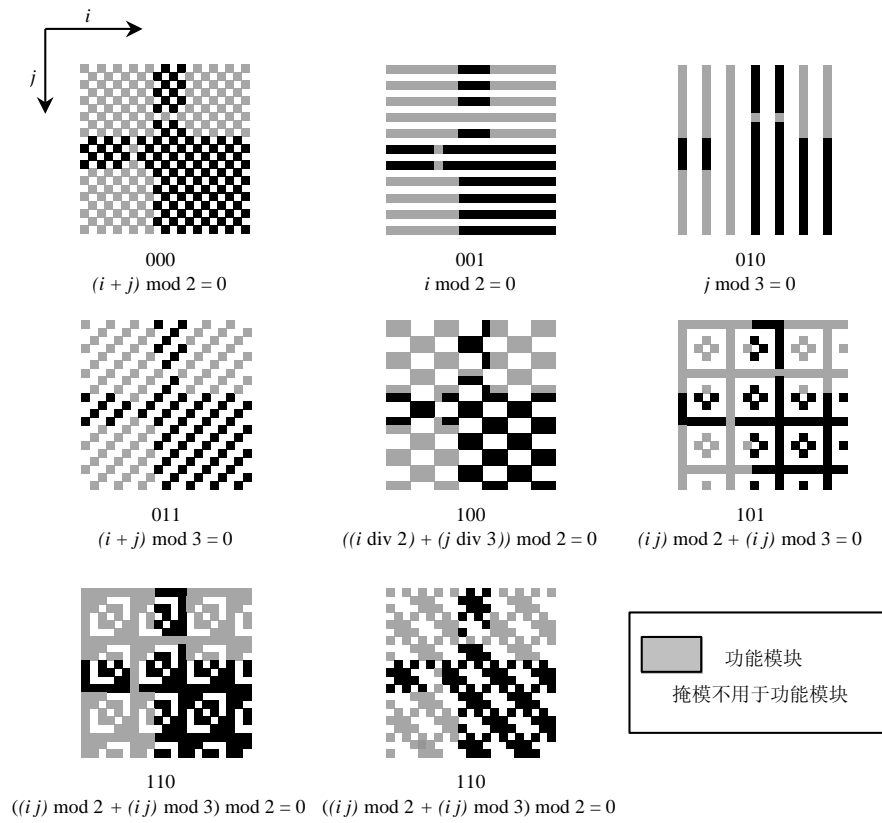


图 17

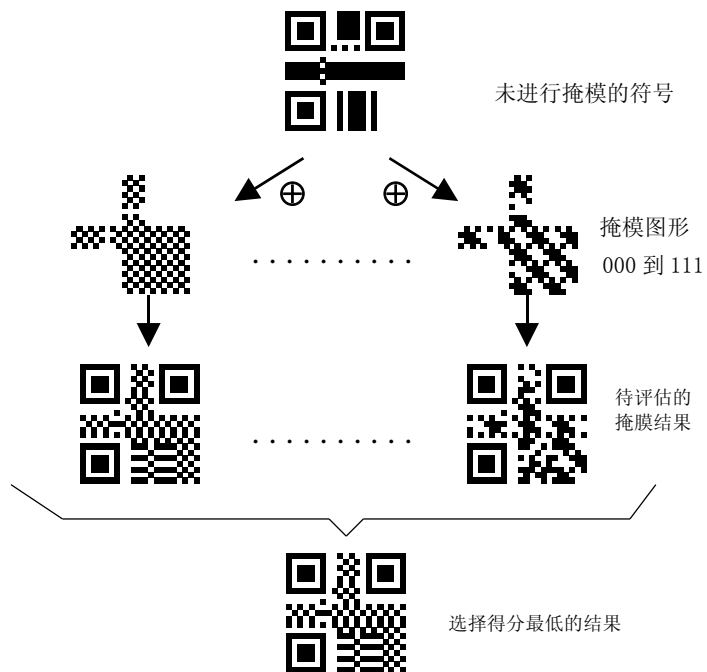


图 18 模式 2 符号的掩模过程

8.8.2 掩模结果的评价

在依次用每一个掩模图形进行掩模操作之后,要通过对每一次如下情况的出现进行罚点记分,以便对每一个结果进行评估,分数越高,其结果越不可用。在表 24 中, N_1 到 N_4 为对不好的特征所罚分数的权重 ($N_1=3$, $N_2=3$, $N_3=40$, $N_4=10$), i 为紧邻的颜色相同模块数大于 5 的次数, k 为符号深色模块所占比率离 50% 的差距, 步长为 5%。虽然掩模操作仅对编码区域进行, 不包括格式信息, 但评价是对整个符号进行的。

表24 掩模结果的记分

特征	评价条件	分数
行/列中相临的模块的颜色相同	模块数 = $(5 + i)$	$N_1 + i$
模块块的颜色相同 颜色相同的模块组成*块	块尺寸 = $m \times n$	$N_2 \times (m - 1) \times (n - 1)$
在行/纵列中出现 1:1:3:1:1(深浅深浅) 图形		N_3
整个符号中深色模块的比率	$50 \pm (5 \times k)\%$ 到 $50 \pm (5 \times (k + 1))\%$	$N_4 \times k$

应选择掩模结果中罚分最低的掩模图形用于符号掩模。

8.9 格式信息

格式信息为 15 位, 其中有 5 个数据位, 10 个是用 BCH (15, 5) 编码计算得到的纠错位。有关格式信息纠错计算的详细内容见附录 C。第一、二数据位是符号的纠错等级, 见表 25。

表25 纠错等级指示符

纠错等级	二进制指示符
L	01
M	00
Q	11
H	10

格式信息数据的第 3 到第 5 位的内容为掩模图形参考, 见表 23, 按 8.8.2 进行图形的选择。

按附录 C 的方法计算 10 位纠错数据, 并加在 5 个数据位之后,。

然后, 将 15 位格式信息与掩模图形 101010000010010 进行 XOR 运算, 以确保纠错等级和掩模图形合在一起的结果不全是 0。

格式信息掩模后的结果应映射到符号中为其保留的区域内, 见图 19。需要注意的是, 格式信息在符号中出现两次以提供冗余, 因为它的正确译码对整个符号的译码至关重要。图 19 中, 格式信息的最低位模块编号为 0, 最高位编号为 14, 位置为 $(4V+9, 8)$ 的模块总是深色, 不作为格式信息的一部分表示, 其中 V 是版本号。

例:

设定纠错等级为 M: 00

掩模图形参考：101
 数据：00101
 BCH 位：0011011100
 掩模前的位序列：001010011011100
 用于 XOR 操作的掩模图形：101010000010010
 格式信息模块图形：100000011001110

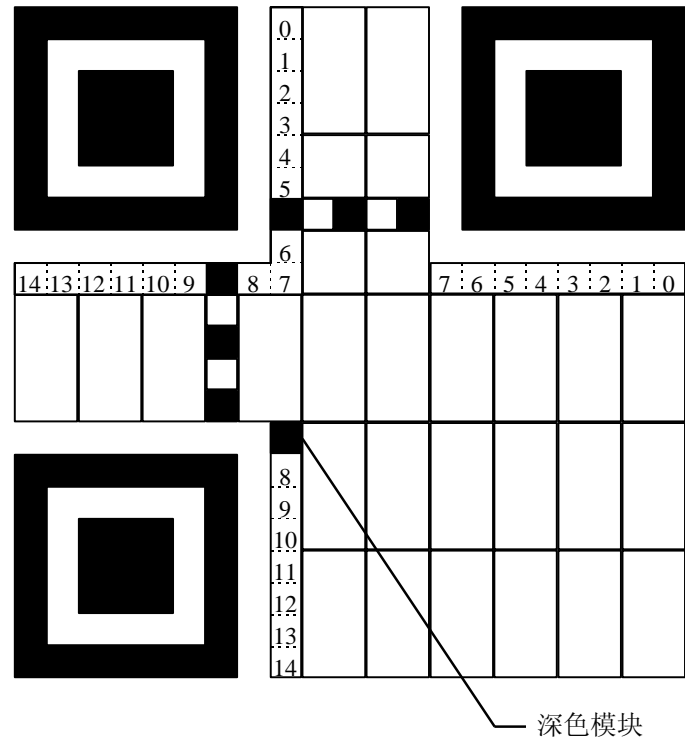


图 19 格式信息位置

8.10 版本信息

版本信息为 18 位，其中，6 位数据位，通过 BCH（18，6）编码计算出 12 个纠错位。版本信息纠错位计算的详细信息见附录 D。6 位数据为版本信息，最高位为第一位。

12 位纠错信息在 6 位数据之后，其计算方法见附录 D。

只有版本 7~40 的符号包含版本信息，没有任何版本信息的结果全为 0。所以不必对版本信息进行掩模。

最终的版本信息应映射在符号中预留的位置，见图 20。需要注意的是，由于版本信息的正确译码是整个符号正确译码的关键，因此版本信息在符号中出现两次以提供冗余。版本信息的最低位模块放在编号为 0 的位置上，最高位放在编号为 17 的位置上，见图 21。

例：
 版本号：7
 数据：000111
 BCH 位：110010010100
 格式信息模块图形：000111110010010100

6 行×3 列模块组成的版本信息块放在定位图形的上面，其右侧紧临右上角位置探测图

形的分隔符，3 行×6 列模块组成的版本信息块放在定位图形的左侧，其下边紧临左下角位置探测图形的分隔符。

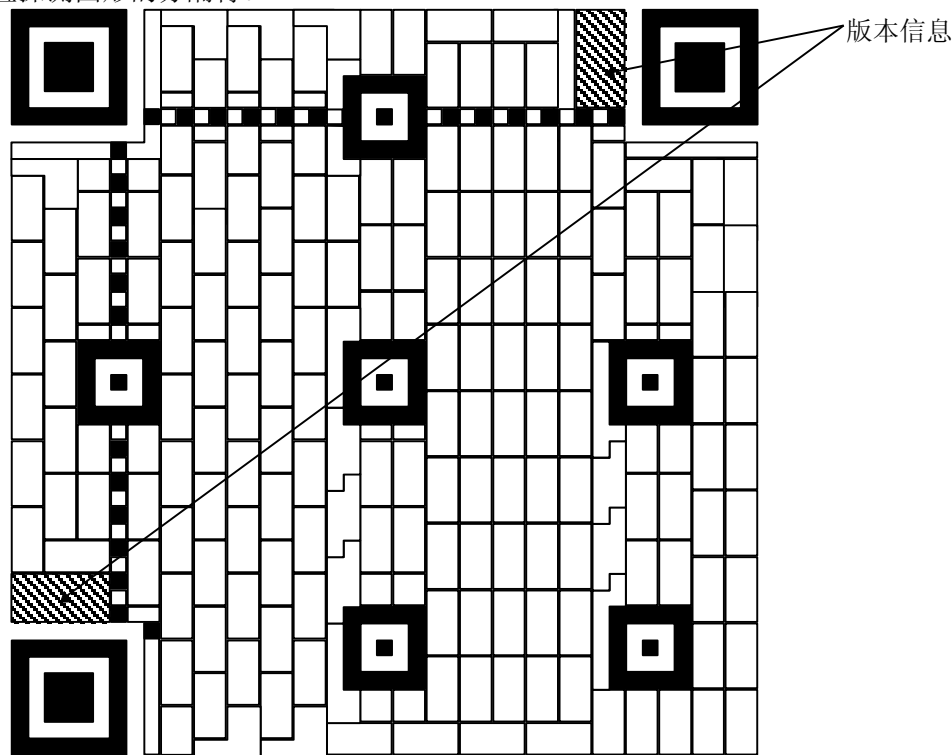


图 20 版本信息位置



图 21 版本信息的模块布置

9 结构链接

9.1 基本原则

可将多达 16 个 QR 码符号以一定的结构方式链接起来，如果一个符号是一个结构链接信息的一部分，那么，它用位于最前边的三个符号字符组成的标头来指示。

结构链接模式指示符 0011 放在第一个符号字符中的四个高位。

结构链接模式指示符之后，紧接两个结构链接码字，它们分布在第一个符号字符的四个低位、第二个符号字符的八位以及第三个符号字符的四个高位。第一个码字是符号序列指示符，第二个码字是奇偶校验数据，它在结构链接信息的所有同一链接的 QR 码符号中是相同的，用以确认所有阅读的符号是同一结构链接信息的一部分。结构链接头后紧接以第一个模式指示符开始的符号的数据码字。如果采用 ECI 缺省模式之外的一个或多个 ECI 模式，那么，

每一个由 ECI 模式指示符与 ECI 模式指定符组成的 ECI 标头应紧接在结构链接头后。

图 22 中的下半部分给出一个结构链接符号的示例，它与图 22 中上半部分的 QR 符号表示相同的数据信息。

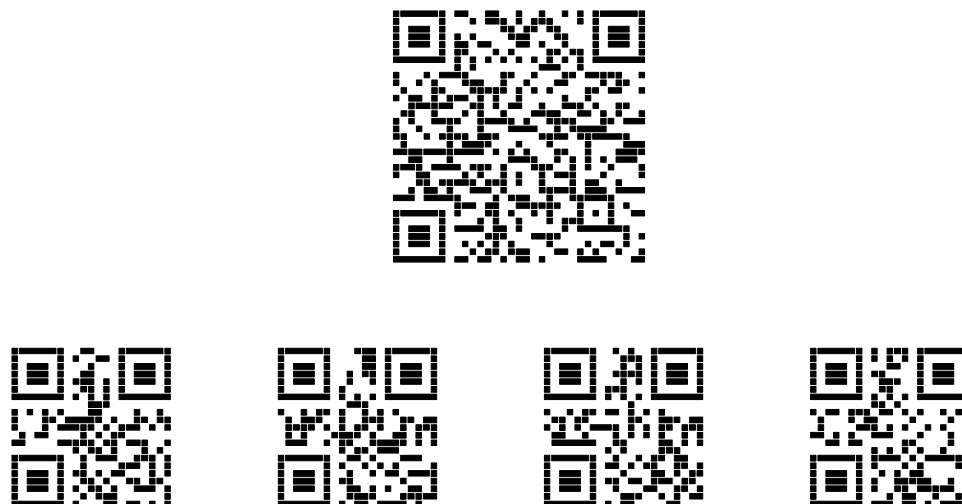


图22 结构链接

9.2 符号序列指示符

此码字表示每一符号在以结构链接方式组成的 QR 码符号集（可包括多达 16 个 QR 码符号）中的位置（以 n 个符号中的第 m 个的形式）。该码字的前四位标识特定符号的位置。后四位标识该结构链接格式中被链接的 QR 码符号的总数。两个 4 位图形应分别与 $(m-1)$ 和 $(n-1)$ 的二进制值相对应。

例：

为表示一个由 7 个符号组成的符号集中的第 3 个符号，编码如下：

第 3 个位置： 0010

符号总数为 7： 0110

字节格式： 00100110

9.3 奇偶性数据

奇偶性数据应是跟在符号序列指示符之后的一个 8 位字节，该奇偶性数据是通过将分成符号块以前的所有原始输入数据的 ASCII/JIS 值逐字节的进行 XOR 运算而获得的一个值。模式指示符、字符计数指示符、填充字节、终止符和填充位不包括在计算中。在该运算中，对日本汉字数据用两字节的转换 JIS 值表示（XOR 运算中，每一字节单独进行），对其它字符用表 6 中的 8 位 JIS 值表示。在 ECI 模式中，应该将在经过数据加密或压缩后获得的 8 位字节数值用于奇偶性数据的计算。

例：

被划分成“0123”，“4567”和“89 日本”的“0123456789 日本”的奇偶性数据计算如下：

第 1 个符号块（“0123”）——hex 的值 30, 31, 32, 33

第 2 个符号块 (“4567”) ——hex 的值 34, 35, 36, 37

第 3 个符号块 (“89 日本”) ——hex 的值 38, 39, 93FA, 967B

通过逐字节地对 “0123456789 日本” 数据连续的进行 XOR 运算计算奇偶性数据信息。

$$30 \oplus 31 \oplus 32 \oplus 33 \oplus 34 \oplus 35 \oplus 36 \oplus 37 \oplus 38 \oplus 39 \oplus 93 \oplus FA \oplus 96 \oplus 7B = 85$$

特别注意, 奇偶性数据计算将根据打印机的容量可在数据被发送到打印机之前或者在打印机中进行。

10 符号印制

10.1 尺寸

QR 码符号尺寸的确定:

X 尺寸: 模块宽度将根据应用要求、采用的扫描技术以及符号生成技术来确定。

Y 尺寸: 模块的高度尺寸必须与模块宽度尺寸相等。

最小空白区: 在符号周围的空白区宽度尺寸为 4X

10.2 供人识读字符

由于 QR 码符号能包含数千个数据字符, 因此供人识读的数据字符包含所有 QR 码所表示的数据信息是不切实际的。作为一种替代, 可用描述性的文本而不是数据原文与符号同时印制在一起。

字符尺寸与字体不作具体规定, 并且供人识读信息可印制在符号周围的任意区域。但不能影响 QR 码符号本身及空白区。

10.3 符号制作导则

可用多种不同的技术印制或标识 QR 码符号。附录 I 给出了用户指导。

11 符号质量

应根据附录 K 规定的二维矩阵码符号印制质量导则, 对 QR 码符号按如下增加和修正的内容进行质量评估。

11.1 获取测试图象

依据附录 K. 1 的要求, 用高精度的图像摄像装置获取被测试符号的一个灰度图象, 照明光及照射方向由应用决定。

11.2 符号质量参数

11.2.1 译码

第 13 章中的参考译码算法可用于测试图象。如果该符号的整个数据信息能被成功译码, 那么译码通过, 译码级别为 “4” 级 (“A”), 否则为 “0” 级 (“F”)。

11.2.2 符号对比度

符号对比度分级依据参考译码算法中定义的符号边缘内的测试图象的所有象素 (包括 4X 宽的空白区) 的灰度值来进行, 具体确定程序见附录 K. 2. 2。

11.2.3 “印制” 增量

译码时, 译码算法首先建立一个高分辨率的二进制数字化测试图象, 然后确定平分符号定位图形的交替模块的中心线的位置。通过检查穿过交替图形的直线的占空比与 50% 的差异来评估印制增量。

取相互独立的两个定位图形（因为水平与垂直增量是有很大的差别的），从与每一定位图形相邻的位置探测图形的外层方块的内边，沿定位图形到与另一位置探测图形相邻接的分隔符的外边，计算遇到的浅色（ N_L ）与深色（ N_D ）的点阵数。在每一方向上印刷增量的结果度量为： $D=N_D/(N_L+N_D)$ ，依据附录 K. 2. 3，按照 $D_{NOM}=0.50$ ， $D_{MIN}=0.35$ ， $D_{MAX}=0.65$ 进行印制增量分级。应按沿着垂直与水平定位图形得到的 D 值中的较低值进行定级。

11.2.4 轴向不一致性

在测试图象的整个区域中参考译码算法最终生成一个数据模块取样点网格，这些取样点准确的水平与垂直间距是评价轴不一致性的基础。

对相邻数据模块之间的水平和垂直间距分别进行计算。计算它们在整个符号的平均值 X_{AVG} 和 Y_{AVG} 。根据它们的两个平均间距的相互接近程度，按附录 K. 2. 4 规定的计算程序对轴向不一致性进行定级。

11.2.5 未使用的纠错

QR 码采用 Reed-Solomon 错误控制编码，较小的符号包含单个纠错字段，较大的符号被划分成两个或多个纠错字段。在所有情况下，应对每一纠错字段按附录 K. 2. 5 独立分级，那么未使用纠错分级应为任一字段中最低的值。然而，该计算不用于格式信息，也不用于版本信息。

11.3 符号的整体分级

一个 QR 码符号的总体印刷质量分级是上述能达到的 5 项指标的分级中的最低值。表 26 汇总了各项测试的定级尺度

表26 符号分级尺度

分级	参考译码	符号对比度	“印刷”增量	轴向不一致性	未使用的纠错
4.0 (A)	合格	$SC \geq 0.70$	$-0.50 \leq D' \leq 0.50$	$AN < 0.06$	$UEC \geq 0.62$
3.0 (B)		$SC \geq 0.55$	$-0.70 \leq D' \leq 0.70$	$AN < 0.08$	$UEC \geq 0.50$
2.0 (C)		$SC \geq 0.40$	$-0.85 \leq D' \leq 0.85$	$AN < 0.10$	$UEC \geq 0.37$
1.0 (D)		$SC \geq 0.20$	$-1.00 \leq D' \leq 1.00$	$AN < 0.12$	$UEC \geq 0.25$
0.0 (F)	不合格	$SC < 0.20$	$D' < -1.00$ 或 $D' > 1.00$	$AN > 0.12$	$UEC < 0.25$

11.4 过程控制测量

有多种工具和方法可以进行有用的测量，对 QR 码符号生成过程进行检测和控制。它们包括：

- 1) 用线性条码检测仪测量符号对比度。
- 2) 用线性条码检测仪对位置探测图形两个轴向的测量，确定水平（与垂直）印制增量。
- 3) 用物理测量确定轴向不一致性。
- 4) 用目测检查位置探测图形与定位图形网格的非一致性和缺陷。

每一种工具与方法见附录 L。

12 译码过程

从识读一个 QR 码符号到输出数据字符的译码步骤是编码程序的逆过程,图 23 为该过程的流程。

- 1) 定位并获取符号图像。深色与浅色模块识别为“0”与“1”的阵列。
- 2) 识读格式信息（如果需要，去除掩模图形并完成对格式信息模块的纠错，识别纠错等级与掩模图形参考。）
- 3) 识读版本信息，确定符号的版本。
- 4) 用掩模图形掩模图形参考已经从格式信息中得出对编码区的位图进行异或处理消除掩模。
- 5) 根据模块排列规则，识读符号字符，恢复信息的数据与纠错码字。
- 6) 用与纠错级别信息相对应的纠错码字检测错误，如果发现错误，立即纠错。
- 7) 根据模式指示符和字符计数指示符将数据码字划分成多个部分。
- 8) 最后，按照使用的模式译码得出数据字符并输出结果。

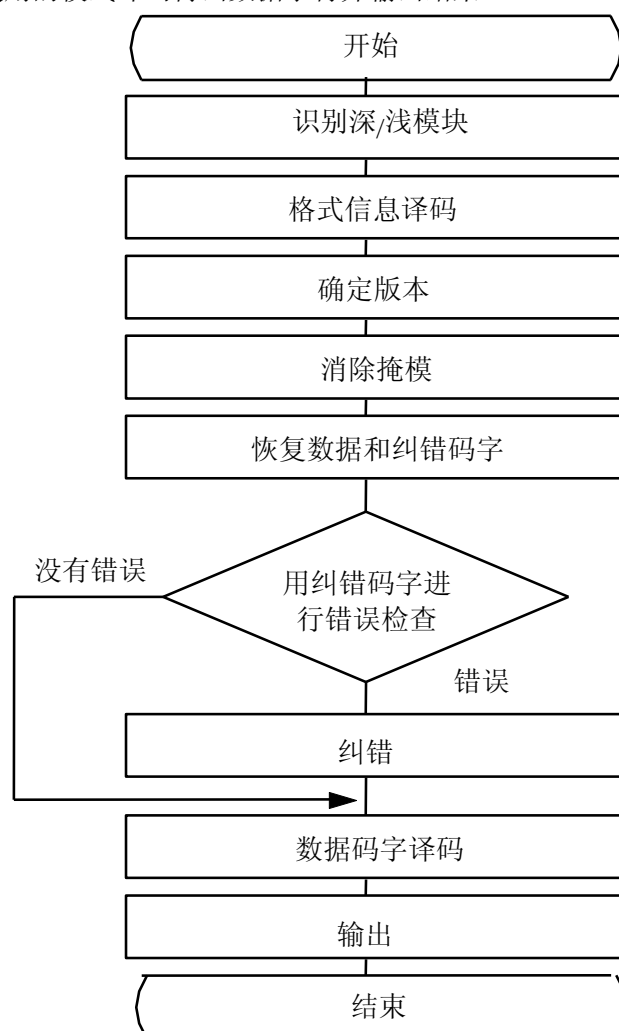


图 23——QR 码译码步骤

13 QR 码的参考译码算法

参考译码算法在图像中寻找符号并进行译码，译码算法参照图象中的深色浅色状态。

1) 选择图象的反射率最大值与最小值之间的中值确定阈值，使用阈值将图象转化为一系列深色与浅色像素。

2) 确定寻象图形，在 QR 码中的寻象图形由位于符号的 4 个角中的 3 个角上的 3 个相同的位置探测图形组成。如 7.3.2 所述，每一位置探测图形的模块序列由一个深色—浅色—深色—浅色—深色次序构成，各元素的相对宽度的比例是 1:1:3:1:1。对本译码算法，每一元素宽度的允许偏差为 0.5（即单个模块的方块的尺寸允许范围为 0.5~1.5，3 个模块宽度的方块的宽度允许尺寸范围为 2.5~3.5）。

a. 当探测到预选区时，注意图像中一行像素与位置探测图形的外边缘相遇的第一点和最后一点 A 和 B（图 24）。对该图像中的相邻像素行重复探测，直到在中心方块 X 轴方向所有穿过位置探测图形的直线被全部识别。

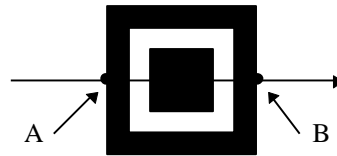


图 24—位置探测图形扫描线

b. 重复步骤 a，在图象的 Y 轴方向，识别穿过位置探测图形中心方块的所有像素行。

c. 确定探测图形中心，通过在 X 轴方向穿过位置探测图形中心块的最外层的像素线上 A、B 两点连线的中点连一直线，用同样方法在另一垂直方向上划一直线，两条直线的交点就是位置探测图形的中心。

d. 重复步骤 a 至 c，确定其它两个位置探测图形的中心位置。

3) 通过分析位置探测图形中心的坐标，识别哪一个位置探测图形是左上角图形以及符号的旋转角度来确定符号的方位。

4) 确定：a) 距离 D，是左上角位置探测图形中心与右上角位置探测图形中心之间的距离，b) 两个探测图形的宽度， W_{UL} 和 W_{UR} 。

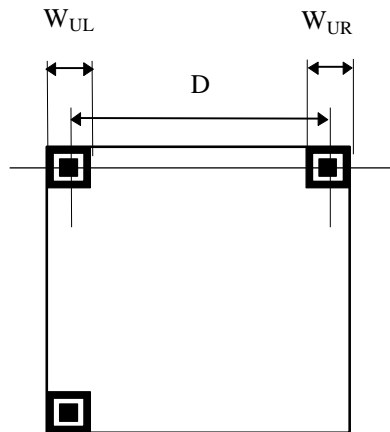


图 25—上部位置探测图形

5) 计算符号的名义模块宽度尺寸 X

$$X = (W_{UL} + W_{UR}) / 14$$

6) 初步确定符号的版本

$$V = [(D/X) - 10] / 4$$

7) 如果初步确定的符号版本等于或小于 6，那么该计算值即为版本号。如果初步确定的符号版本等于或大于 7，那么版本信息应按下列步骤译码：

- a) 用 7 除以右上角位置探测图形的宽度尺寸 W_{UR} ，得到模块尺寸 CP_{UR}

$$CP_{UR} = W_{UR} / 7$$

b) 见图 26，由 A、B 和 C 找出通过三个位置探测图形中心的导向线 AC、AB。根据与导向线相平行的直线、位置探测图形的中心坐标和模块尺寸 CP_{UR} 确定在版本信息 1 区域中每一模块中心的取样网格。二进制值 0 和 1 根据采样网格上的深色浅色的图形来确定。

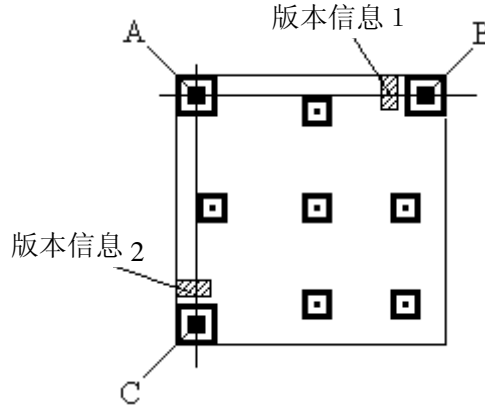


图 26—位置探测图形与版本信息

c) 通过检测并纠错确定版本，如果有错，根据 BCH 纠错原理，对版本信息模块出现的错误进行纠错，参见附录 D。

d) 如果发现错误超过纠错容量，那末计算左下方位置探测图形的宽度尺寸 W_{DL} ，并按上述步骤 a)，b)，c) 对版本信息 2 进行译码。

8) 对于没有校正图形的版本 1 符号，继续按照附录 M. 14 中的算法的第 7 步和第 8 步进行译码，然后转至本算法的第 9 步。版本 2 以及更高版本的符号的译码要求由 7.3.5 和附录 E 定义的坐标决定的每一校正图形的中心坐标来确定取样网格。

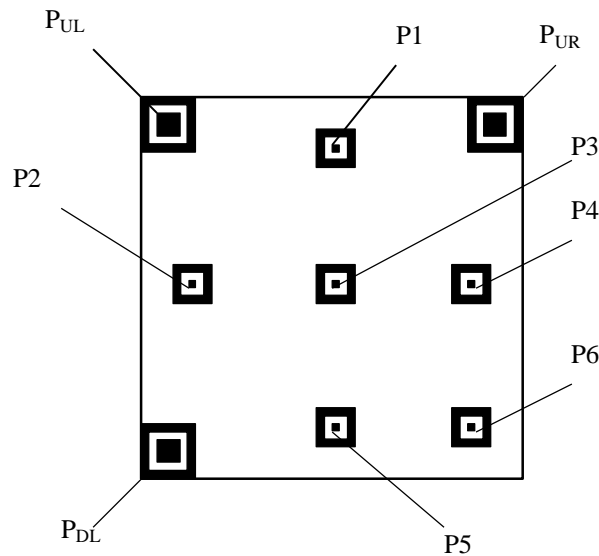


图 27—位置探测图形与校正图形

a) 左上角位置探测图形的宽度 W_{UL} 除以 7，计算模块尺寸 CP_{UL}

$$CP_{UL}=W_{UL}/7$$

b) 根据左上角位置探测图形 P_{UL} 的中心 A 的坐标，平行于从第 7c) 得到的导向直线 AB 和 AC 的直线以及模块尺寸 CP_{UL} 初步确定校正图形 P_1 和 P_2 的中心坐标。

c) 从初定的中心坐标的象素开始，扫描校正图形 P_1 和 P_2 中的空白方块的轮廓，确定实际的中心坐标 X_i 和 Y_j (见图 28)

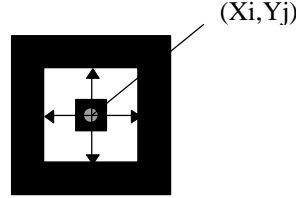


图 28—校正图形的中心坐标

d) 根据左上角位置探测图形 P_{UL} 的中心坐标和在 c) 中得到的校正图形 P_1 和 P_2 的实际中心坐标值，估计校正图形 P_3 的初步中心坐标。

e) 按照 c) 中同样的步骤找到校正图形 P_3 的实际中心坐标。

f) 确定 L_x 和 L_y ， L_x 是指校正图形 P_2 和 P_3 两中心之间的距离， L_y 是指校正图形 P_1 和 P_3 两中心之间的距离。用校正图形的已定义的间距除 L_x 和 L_y ，获得位于符号左上角区域下边的模块节距 CP_x 和右边的模块节距 CP_y 值。

$$CP_x=L_x/AP$$

$$CP_y=L_y/AP$$

其中，AP 是校正图形中心的模块间距（见表 E. 1）

以同样方式，找出 L_x' 和 L_y' ， L_x' 是左上部位置探测图形 P_{UL} 与校正图形 P_1 的中心坐标之间的水平距离。 L_y' 是左上部位置探测图形 P_{UL} 的中心坐标与校正图形 P_2 的中心坐标之间的垂直距离。由下面给出的公式计算符号左上角区域中上边的模块节距 CP_x' 和左边的节距 CP_y' 值。

$CP_x' = L_x' / (\text{校正图形 } P_1 \text{ 的中心模块的列坐标} - \text{左上部位置探测图形 } P_{UL} \text{ 的中心模块的列坐标})$

$CP_y' = L_y' / (\text{校正图形 } P_2 \text{ 的中心模块的行坐标} - \text{左上部位置探测图形 } P_{UL} \text{ 的中心模块的行坐标})$

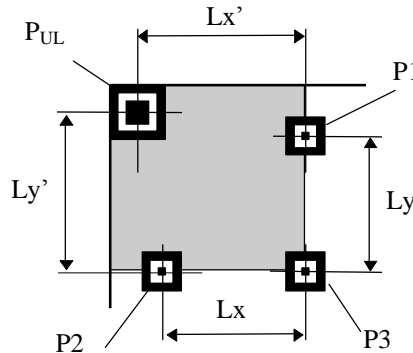


图 29—符号的左上区域

g) 依据代表符号左上区的每一边的模块节距值 CP_x , CP_x' , CP_y 和 CP_y' , 确定覆盖符号的左上区的采样网格。

h) 在同样方式下, 确定符号右上区 (被右上角位置探测图形 P_{UR} , 校正图形 P_1 , P_3 和 P_4 所覆盖) 和符号左下区 (被右上区位置探测图形 P_{UR} , 校正图形 P_2 , P_3 和 P_5 覆盖) 的采样网格。

i) 对校正图形 P_6 (见图 30), 由校正图形 P_3 、 P_4 和 P_5 的间距, 穿过校正图形 P_3 和 P_4 , P_4 和 P_5 的中心的导向直线以及这些图形的中心坐标值得到的模块间距 CP_x' 和 CP_y' 值, 估计它的初步的中心坐标。

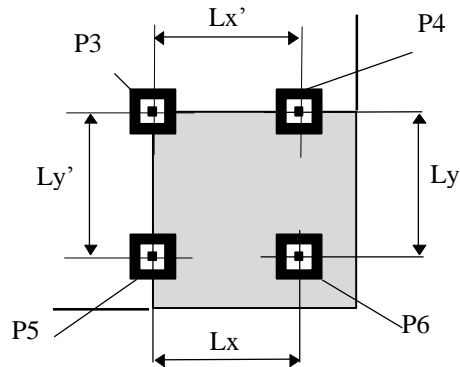


图 30—符号右下区

j) 重复步骤 e) — h), 确定符号右下区的采样网格。

k) 用同样原则确定符号未覆盖区的采样网格。

9) 对网格的每一交点上的图象像素取样, 并根据阈值确定是深色块还是浅色块, 构造一个位图, 用二进制的 “1” 表示深色的像素, 用二进制的 “0” 表示浅色的像素。

10) 对与左上区位置探测图形相邻的格式信息译码, 得到纠错等级和用于符号的掩模图形。如果检测出错误超过格式信息的纠错容量, 那么, 采用同样的程序对与右上部和左下部位置探测图形相邻的格式信息进行译码。

11) 用掩模图形对符号编码区进行异或 (XOR) 处理, 解除掩模并恢复表示数据和纠错码字的符号字符。这与在编码程序中采用的掩模处理过程的作用正好相反。

12) 根据 8.7.3 中的排列规则确定符号码字。

13) 根据符号版本和纠错等级需求, 用在 8.6 中第 3 步规定的交替处理的逆过程, 重新将码字序列按块排列。

14) 按照附录 B 中的错误检测与纠错译码程序, 纠正替代错误与拒读错误, 直到符号版本和纠错等级所规定的最大纠错容量。

15) 通过重新组配数据块序列, 恢复原始信息的位流。

16) 将数据位流分成若干段, 每一段由模式指示符开始, 段的长度由在模式指标符后的字符计数指示符确定。

17) 根据采用的模式的规则, 对每一段进行译码。

14 自动鉴别能力

QR 码可以与许多其它码制符号一起用于自动识别环境中（见附录 J）。除此而外，通过对格式信息掩模图形的分析，可自动识别模式 1 和模式 2 符号。设备供应商可提供选择仅对模式 1 或仅对模式 2 或对两种模式都可进行译码。

15 数据传输

所有编码的数据字符都应包括在数据传输之列。功能图形、格式信息与版本信息、纠错字符、填充和剩余字符不传输。所有数据的缺省传输模式是它们的 8 位 JIS8 值或 16 位转换 JIS 值。由于字符值的设定，它准确地传输数字、拉丁字母、假名、日本汉字和中国汉字数据序列。工作在缓冲模式下的译码器在传输之前将数据文件重新链接，不传输结构链接头。如果译码器在非缓冲模式下工作，结构链接头应作为每个符号的前两个字节进行传输。包含扩展解释的数据传输在内的更复杂的解释将在下面叙述。

15.1 符号标识符

ISO/IEC 15424 提供了一个标准的程序，根据译码器的设置选择和符号的自身特性报告已经阅读的码制。

一旦数据结构（包括使用的 ECI 模式）被识别，译码器将适当的符号标识符作为一个段首标记追加到被传输的数据上；如果使用 ECI 模式，就需要符号标识符。用于 QR 码的符号标识与选择值见附录 F。

15.2 扩展解释

在支持 ECI 协议的系统中，每一传输都要求传输符号标识符。在任何时候遇到 ECI 模式指示符，它应作为转义字符 5C_{HEX} 被传输，依据 JIS X 0201，5C_{HEX} 在 QR 码的缺省编码中代表字符“¥”（在 AIM ECI 规范和 ASCII 字符集中，“¥”字符值对应反斜杠字符“\”）。按照在表 4 中定义规则的逆运算，表示 ECI 指示符的码字将转化成一个 6 位数，这些 6 位数将被传输为 30_{HEX} 至 39_{HEX} 之间的相对应的 8 位值，紧接在转义字符之后。

应用软件识别到 \nnnnnn 之后，将所有后续字符解释为来自 6 位数字的指示符定义的 ECI。该解释在下述两种情况出现之前一直有效：

- a) 编码数据的结束；
- b) 按 AIM ECI 规范所定义的规则，通过模式指示符 0111 表示改变为一个新的 ECI。

当返回到缺省解释方式时，译码器应输出适合的转义序列作为数据的前缀。

如果字符“¥”需要作为被编码的数据，应按如下方式进行传输：每当字符 5C_{HEX} 作为数据出现，应传输两个该值的字节，因此每当单个值出现，总是一个转义字符，连续两次出现则表示真正的数据。

例 1：

1) 被编码的数据：ABC ¥ 1234

被传输的数据：ABC ¥¥ 1234

2) 被编码的数据：ABC 后面紧跟按照 ECI 123456 的规则编码的<后续数据>。

被传输的数据：ABC ¥ 123456<后续数据>。

例 2：（使用 8.4.1.1 中的数据）

该信息包含：ECI 模式指示符、ECI 指定符、模式指示符、字符计数指示符以及数据，

形式如下：

0111 00001001 0100 00000101 10100001 10100010 10100011 10100100 10100101

符号标识符]Q2（见附录 F）必须附加到数据传输之中。

传输（hex 值）5D 51 32 5C 30 30 30 30 39 A1 A2 A3 A4 A5

以 ECI000009 编码的数据 ABΓ △E

在结构链接模式中，如果符号的开始就遇到 ECI 模式指示符，后续序列数据字符应被解释为当前模式，前一符号的终止端使用的 ECI 无效。

注：反斜杠字符“\”，ASCII 值 5C_{HEX} 等同于 JIS X 0201 中的“¥”。

15.3 FNC1

在第 1 或第 2 个位置隐含 FNC1 的模式中，由于没有 ASCII 或 JIS8 值与该字符相对应，该隐含字符不能被直接传输。因此必须通过相关符号标识符（）Q3，）Q4，）Q5，）Q6）传输，指示隐含 FNC1 出现在第一个或第二个位置。在这些符号的其它位置，依据相关应用规则，FNC1 也可作为一个数据字段分隔符出现。在字母数字模中用字符“%”表示；在 8 位字节模式中用字符“GS”（ASCII/JIS 8 值 1D_{HEX}）表示。在两种模式中，译码器应传输 ASCII/JIS 8 值 1D_{HEX}。

如果字符“%”在字母模式中是编码数据的一部分，它在符号中用“%%”表示，如果译码器遇到这种情况将以单个字符“%”传输。