誤り訂正符号とＱＲコードへの応用について

理学部第一部　数理情報科学科
１４０６０８５　原田　経道
指導教員　柳田　昌宏

１．はじめに

4 年間の勉強の中で最も興味を持ったものは、ハミング符号であった。線形代数を使って、誤りが訂正できる仕組みに感心した。よって今回は、その延長線上にあるリード・ソロモン符号を紹介し、ハミング符号における簡単な誤り訂正の例を挙げる。次に、ＱＲコードの仕組みを、「原田　経道」を符号化することで示し、理解を深めようと思う。

２．理論

２－１．**ガロア体**

ｑ個の元からなる体を**ガロア体**といい、GF(ｑ)で表す。ガロア体には、次の２種類がある。

・**素体**：ｑを素数とし、任意の元 a , b ∈{0, 1, …, ｑ－1}対し、次の加法と乗法が定義すると、体をなす。

（加法）a + b =(a + b) mod ｑ

（乗法）a・b = (a × b)mod ｑ

・**拡大体**：ｑ＝p^m（ｐは素数、ｍは正整数）とし、次の集合Ａ（多項式環）を考える。

$\{a_{m-1}x^{m-1}+\ldots..+a_1+a_0 \mid a_{m-1},\ldots,a_0 \in GF(p)\}$

a(x),b(x)(≠ 0) ∈ A に対し、a(x)をb(x)で割った商q(x)と余りr(x)はa(x) = b(x)q(x) + r(x)(deg(q(x)>deg(r(x)))と一意に定まる。

r(x)=0 のとき、b(x) | a(x)と表す。

また、r(x) = a(x)modq(x)でモジュロ計算を定義する。

任意の元 f(x) , g (x)∈ A に対し、次の加法と乗法を定義すると、Ａは体をなす。

（加法）f(x) + g(x) =[f(x) + g(x)] mod P(x)

（乗法）f(x)・g(x) = [f(x)×g(x)]mod P(x)

(ただし、各係数は素体 GF(p)上の演算を行う。ここでP(x)は、ｍ次既約多項式)

これを、**GF(p)上の**拡大体という。さらに、ｐが素数のべきならば、同様にして **GF(**p^m**)上の**拡大体も作ることが出来る。

拡大体には３種類の表現方法がある。

①$\alpha^i = 1$を満たす最小の正整数 ｉ がｑ－１であるようなα（**原始元**という）∈ $GF(q)$ を適当に選ぶと、$\alpha^0, \alpha^1 \ldots, \alpha^{q-2}$ は非零の互いに異なるｑ－１個の元となるので、これと０を用いてＧＦ（ｑ）は次のように表せ、これを**べき表現**という。

$GF(q) = \{0, \alpha^0, \alpha^1 \ldots, \alpha^{q-2}\}$

②元を多項式で表現したものを**多項式表現**という。

$a_{m-1}x^{m-1} + \ldots\ldots + a_1 + a_0$

ここで、ｍ次既約多項式 P(x)の根をαとし、α^i (i = 0,1, …,q－1)をP(α)で割った余りをα^iに対応させると、多項式表現とべき表現が互いに交換できる。

③元の係数だけを用いて表現したものを**ベクトル表現**という。$(a_{m-1},\ldots,a_1,a_0)$

例えば、$GF(2^3)$ の元は次のように表現される。

べき表現	多項式表現	ベクトル表現
0	0	000
1	1	001
α	x	010
α^2	x^2	100
α^3	x + 1	011
α^4	x^2 + x	110
α^5	x^2 + x + 1	111
α^6	x^2 + 1	101

（ただし、P(α)= α^3 + α + 1 = 0 、α^7 = 1）

２－２．通信路モデル

１９４８年、クロード・シャノン（1916 – 2001）が発表した記念碑的論文「通信の数学的理論」によって情報理論は幕を開けた。その論文の中でシャノンは次のような**通信路モデル**を考えた。

「情報源→符号器→通信路→復号器→受信地」

情報伝達は本質的に５つの部分から成り立っている。

(1)**情報源：**送信したい情報の集合。**X** で表す。この集合の元m（情報）を**情報源記号**という。

(2)**符号器：**情報源の元をｑ元の記号列へ変換する（符号化）。ｑ元の記号の集合を **V** とすると、この変換ｆは **X** からV^*への写像であり、ｆを**符号化関数**と呼ぶ。V^*の部分集合 C= {f(α) | α∈**X**} を（ｑ元）**符号**といいＣの元ｃを**符号語**という。

(3)**通信路**：情報を伝達するための媒介物。通信路では、通信を妨害する**雑音**が混入し、符号が誤って伝わってしまうことが考えられる。

(4)**復号器**：受信した記号列から、あらかじめ決められた**決定則**Γに従ってもとの符号列を推定して訂正し、さらにその符号列を情報源列へ変換する。

(5)**受信地**：情報の最終的な送り先

２－３．線形符号

V^nをGF(ｑ)上でのｎ次元線形空間とする。これはGF（ｑ）上での拡大体のことである。ＣがV^nのｋ次元部分空間であるとき、Ｃを**[n,k]線形符号**という。

符号化：符号に線形性が加わると、符号化関数（線形写像）が行列で表現できるからである。この行列を**生成行列**といい、Ｇ（ｋ×ｎ型）で現す。符号化関数は、情報源ベクトル**m**∈V^kに対し、**c**＝ｆ（**m**）＝**m**Ｇで与えられる。このとき、Ｇ＝（I_k　Ｐ）の標準形に変形して用いると、**c**の先頭ｋビットが**m**となり、便利である。（I_kはｋ次の単位行列）

通信路：[n,k]線形符号では、通信路での雑音を誤りベクトル **e**∈V^nを用い表現する。すると、受信語ベクトル**w**は**w**＝**c**＋ **e**∈V^nで現せる。

復号化：決定則の一つとして、**限界距離復号法**を紹介する。これは、受信語を“最も近い”符号へ復号する方法である。「近さ」を表現するために、以下で定義される**ハミング距離**を導入する。

定義　**x**＝（$x_1 \ldots x_n$）,**y** = ($y_1 \ldots y_n$) に対し、

$$d_H(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{n} d_1(x_i, y_i)$$

ただし、$d_1(x_i, y_i) = \begin{cases} 1 & (x_i \neq y_i) \\ 0 & (x_i = y_i) \end{cases}$

ハミング距離は、距離の公理を満たす。

次に、**最小距離**ｄと**最小重み**ｗ（**x**）を定義する。

d=d(C) ＝min{$d_H(\mathbf{x},\mathbf{y}) | \mathbf{x},\mathbf{y} \in C, \mathbf{x} \neq \mathbf{y}$}

ｗ（**x**） ＝min{$d_H(\mathbf{x},\mathbf{0}) | \mathbf{x} \in C, \mathbf{x} \neq \mathbf{0}$}

線形符号では、最小距離と最小重みは一致する。

最小距離ｄの[n ,k]線形符号は**[n,k,d]**とも表記される。

さて、符号間の最小距離ｄが 2t+1 以上ならば、ある符号語からハミング距離ｔ以下だけ離れた受信語に対して一番近い符号語はたかだか１つしか存在しない。

したがって、決定則は「最小距離ｄの符号Ｃと受信語**w**が与えられたとき、d_H(**w** , **c**)＜ｄ/２を満たす**c**∈Cが存在すれば**c**に復号し、なければ復号不能と判定する」となる。

例えば、C={000、111}とすると、(000)(100)(010)(001)は(000)へ、(111)(110)(011)(101)は(111)へ復号される。

受信語と符号語の対応表を作れば、表から対応を検索して復号できるが、この検索は時間がかかる。よって、算出によって**w**から**c**を推測するアルゴリズムが欲しい。線形符号では、次の方法ある。

＜**シンドローム復号法**＞

$Gh^T = 0$を満たす**h**∈V^kを**パリティ検査**という。パリティ検査ベクトルの全体はV^nの部分空間であり、次元はｎ－ｋとなる。そして、１次独立なパリティ検査を行ベクトルにもつ行列Ｈを**チェック行列**という。Ｈは、**c** H^T ＝0を満たす。とくに、Ｇ＝（I_k Ｐ）ならば、$Gh^T = 0$よりＨ＝（$P^T I_{n-k}$）となる。

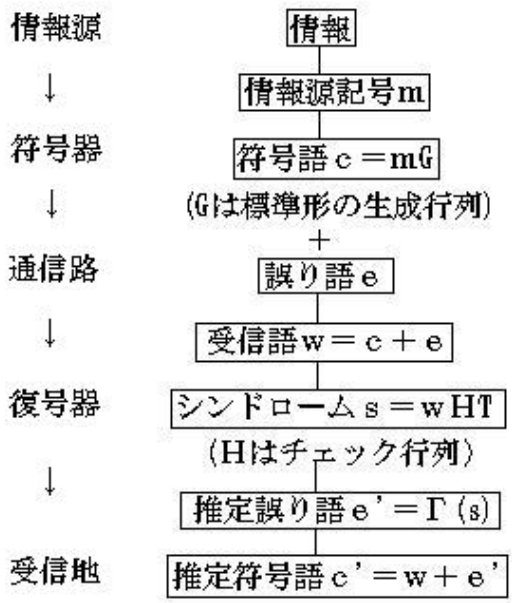
任意のベクトル**s**∈V^{n-k}に対し、

$C(s)=\{r\in V^n \mid rH=s\}$ と定義する。C(s)を**シンドローム s** を持つ**コセット**という。C(s)に属するベクトルの中で最小重みw(r)をもつ元を任意に選び、**コセットリーダーe** と呼ぶ。もし、w(e)＜d/2 を満たせば、各 C(s)には e はたかだか１つしか存在しない。よって、一般に**w**は**w**＝**c**＋**e** と表現されるが、w(e)＜d/2 ならば、**w**を表現する **e** と **c** は一意に定まる。さらに、このとき **c** は**w**に最も近い唯一の符号語となっている。今、R をハミング重み d/2 未満のコセットリーダーの集合とし、ϕ を復号不能を表す記号とする。**e** に **s** を対応づける写像 $\varphi: V^{n-k} \to R\cup\{\phi\}$ を、次のように定義する。

$$\varphi(s)=\begin{cases} e(\in R), & eH=s\text{なる}e\in R\text{が存在するとき} \\ \phi, & \text{それ以外} \end{cases}$$

また受信語**w**に対し、符号語**w**＋φ（**w**H）に復号化するものと定義する。すると、異なる **s**＝**w**H が与えられたとき、**w**＋φ（**w**H）＝**e**＋ φ((**c**＋**e**)H)
＝**c**＋**e**＋ φ(**e**H)＝**c**＋**e**＋**e**＝**c** と最小距離で一意に復号される。

しかし、実際にはφを計算するアルゴリズムを一般的に論じることは難しく、符号の種類ごとに様々なアルゴリズムが研究されている。以上、まとめると



２－４．巡回符号

線形符号の中で、工学的な装置化が容易なものは巡回符号である。

定義

C を GF(q)上の[n ,k , d]線形符号とする。このとき、$(x_1,x_2\ldots,x_n)\in C$ ならば $(x_n,x_1\ldots,x_{n-1})\in C$ が成り立つとき、C を**巡回符号**という。

巡回符号では、ベクトル表現より多項式表現の方が便利である。GF(q)上の符号語 c＝$(c_{n-1},c_{n-2}\ldots,c_0)$ を多項式表現 c(x)＝$c_{n-1}x^{n-1}+\ldots+c_1x+c_0$ で表したもの**符号多項式**という。

巡回符号 C において、０以外の最高次の次数が１であるような最小次数の多項式を特に**生成多項式**といい、g(x)で表す。生成多項式は C において一意に定まり、以下が成り立つ。

(1)任意の c (x)$\in C$ に対し、g(x) | c(x)

(2)g(x) | x^n-1

(3)k = n – deg(g(x))

具体的には、x^n-1 は最高次数が１の既約多項式の積に一意に因数分解されるということが知られているので、g(x)はその既約多項式の積を選んで作ればよい。

巡回符号 C は次のように言い換えることができる。

定理 g(x)=$x^{n-k}+\ldots+g_1x+g_0$ を GF(q)上の最高次数が１で g(x) | x^n-1 な多項式とする。また、m(x)＝$m_{k-1}x^{k-1}+\ldots+m_1x+m_0$ を GF(q)上の k－１以下の多項式とする。このとき、C = {m(x)g(x)} は生成多項式 g(x)の巡回符号である。

このとき c(x)は g(x),xg(x), …, x^{k-1}g(x)の線形結合で c(x)= $m_{k-1}x^{k-1}g(x)+\ldots+m_1x\ g(x)+m_0g(x)$ と表せるので、巡回符号Cの生成行列Gはｋ×ｎ行列となる。

$$G=\begin{pmatrix} x^{k-1}g(x) \\ \vdots \\ g(x) \end{pmatrix}=\begin{pmatrix} 1 & \ldots & g_0 & & 0 \\ & \ldots & \ldots & \ldots & \\ 0 & & 1 & \ldots & g_0 \end{pmatrix}$$

つまり、 c＝mG と c(x)=m(x)g(x)は同義である。

巡回符号を実際に使う場合は短縮化することが多い。

それには、ｋ×ｎ型生成行列の左上から行と列ｒ個を削り、(ｋ－ｒ)×（ｎ－ｒ）型にすれば、[n-r ,k-r]（短縮）巡回符号になる。

次の定理は、巡回符号の最小距離の下界を与える。

定理（BCH 限界） GF(ｑ)上の[n, k]巡回符号 C の生成多項式が $\alpha^n = 1$ を満たすαの連続する２ｔ個のべき $\alpha^l, \alpha^{l+1} \ldots, \alpha^{l+2t-1}$ を根として持つなら、最小距離 $d(C) \geq 2t+1$ となる。

この BCH 限界を満たすように構成されたれ符号が、次に与える RS 符号と BCH 符号である。

定義（RS 符号）

$\alpha^n = 1$ を満たすαのべき $\alpha^l, \alpha^{l+1} \ldots, \alpha^{l+2t-1}$ を根として持つ GF(q)上の次数が最小で最高次の係数が１の多項式 g(x)を生成多項式とする、符号長ｎ＝ｑ－１のｑ元巡回符号を **RS 符号（リード・ソロモン符号）**という。

RS 符号の生成多項式 g(x)は

$g(x) = LCM[M_l(x), M_{l+1}(x) \ldots, M_{l+2t+1}(x)]$

(ここで、$M_i(x)$ は、α^i を根とする GF(q)上で次数が最小で、最高次の係数が１な多項式（**最小多項式**という）である。また、ＬＣMは最小公倍多項式)

$= (x-\alpha^l)(x-\alpha^{l+1}) \ldots (x-\alpha^{l+2t-1})$ と書け、BCH 限界よりｔ個の誤りを訂正できる。

さらに RS 符号の定義において、$n = q^m - 1$ となるものを**原始 BCH 符号（ボーズ・チャウドウリ・ホッケンガム符号）**という。

２－５．具体例（ハミング符号）

２元 [7, 4, 3]原始ＢＣＨ符号の具体例を挙げる。

情報源ベクトルｍ＝（１０１）を送信する。

まず、ｎ＝７＝2^3-1、ｄ=3=2×1+1 より、$\alpha^7 = 1$ を満たす、α, α^2 を根にもつ生成多項式を考える。今、GF(2)上の $g(x) = x^3 + x + 1$ の根αを $GF(2^3)$ の原始元とすると、$g(\alpha^2) = (\alpha^2)^3 + (\alpha^2) + 1 = 0$ より、α^2 も g(x)の根となるので、g(x)を生成多項式としてよい。すると、生成行列Ｇは

$$G = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

となり、ｍを符号化すると、符号語は

ｃ＝ｍＧ＝（1010011）となる。

ここで、通信路で誤りベクトル e=(0010000)が加わり、受信語ｗ＝(1000011)を受け取ったとしよう。これを正しく復号化したい。チェック行列は、Ｈ＝（$P^T I_3$）

$$= \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

となり、シンドロームはｓ＝ｗ H^T ＝（１１０）となる。ここで、シンドロームと誤りベクトルは一意に対応することを思い出し、パリティ検査ベクトルのｊ列目に一致するシンドロームへ、ｊ列目が１で他が０な誤りベクトルを対応させる。すると、チェック行列を検索することで誤りベクトルが推定できる。個の場合だと、(110)は３列目にあるので、誤りベクトルはｅ（0010000）となる。よって、ｗ＋ｅ=(1010011)となり、正しく復号化された。

シンドローム復号法により受信語と符号語の対応表を検索するよりもシンドロームと誤り語の対応表を検索する方がずっと簡単になっていることを確認したい。

チェック行列を見ると、GF(2)上の **0** でない３次元ベクトルがすべて列に現れる。このような **0** でないｍ次元ベクトルを全て並べたチェック行列から出来る符号を特に $[2^m - 1, 2^m - 1 - m, 3]$ **ハミング符号**という。

３．ＱＲコード

３－１．ＱＲコードとは

QR コードとは、1994 年にデンソーの開発部門が開発したマトリックス型二次元コードである。あらかじめ決められた規則に従い、０と１のビット列を作成し、白と黒の格子状のパターンで置き換え、情報を表す。

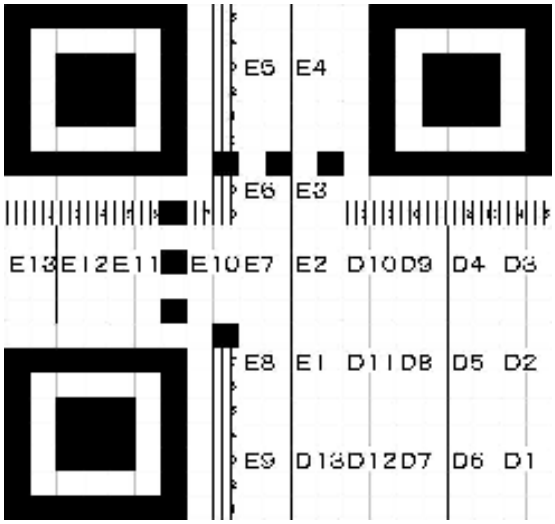
ＱＲコードには、１型から４０型までの**型番**と、７％（Ｌ）、１５％（Ｍ）、２５％（Ｑ）、３０％（Ｈ）の**誤**

り訂正レベルが定められていて、型番と誤り訂正レベルによってＲＳ符号とＢＣＨ符号のパラメータとブロック数が決められている。例えばＲＳ符号の場合、型番１で誤り訂正レベルが

Ｌ→[26,19]、Ｍ→[26,16],Ｑ→[26,13]、Ｈ→[26,9]

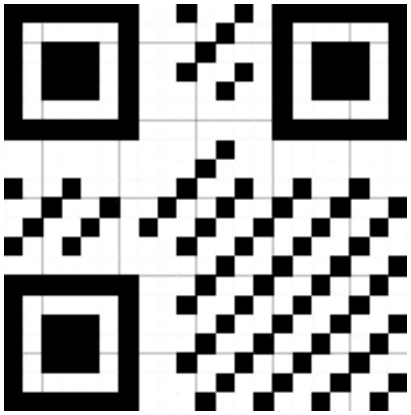
である。今回は、１－Ｑ型を使って符号化する。

１型の型番は以下の通りである。



26 個のブロックに分かれた部分にＲＳ符号を配置し、縦横に伸びる１列にＢＣＨ符号を配置する。

最後に、あらかじめ決められたマスクパターンを使い、各ビットに対し、GF(2)上の加法をする。今回は、以下のマスクパターンを使用する。



３－２．作成

３－２－１．形式情報のＢＣＨ符号化

形式情報とは、2 ビットの誤り訂正指示子と、3 ビットのマスクパターン指示子をあわせたものである。今回は、１１（レベルＱの指示子）＋０１１（上記のマスクパターンの指示子）をあわせたｍ＝（１１０１１）を符号化し、[15, 5, 7]BCH 符号を作る。GF(2^4)の原始元αを根として持つ既約多項式を(x^4+x+1)とし、生成多項式を

$$g(x)=(x^4+x+1)(x^4+x^3+x^2+x+1)(x^2+x+1)$$

とする。(第１式はα,α^2,α^4を、第２式はα^3,α^6を、第３式はα^5を根に持つ）と、生成行列は

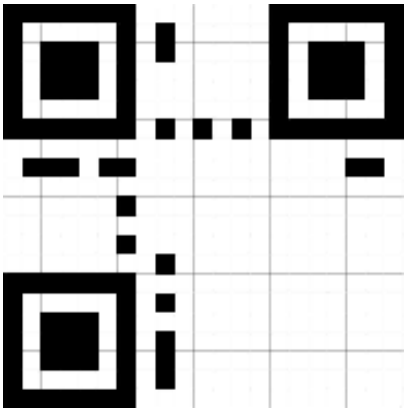
$$G=\begin{pmatrix}1&0&0&0&0&1&0&1&0&0&1&1&0&1&1\\0&1&0&0&0&1&1&1&1&0&1&0&1&1&0\\0&0&1&0&0&0&1&1&1&1&0&1&0&1&1\\0&0&0&1&0&1&0&0&1&1&0&1&1&1&0\\0&0&0&0&1&0&1&0&0&1&1&0&1&1&1\end{pmatrix}$$

となり符号語は、ｃ＝ｍＧ

＝（１１０１１１０００１０１００）

となる。これにマスクパターン

（１０１０１０００００１００１０）を GF(2)上で加えて、（０１１１０１０００００１１０）を得る。これを、規則に従い配置すると、以下のようになる。



３－２－２．データビット列のＲＳ符号化

データビット列は、以下の順で並べる。

１）4 ビットのモード指示子：

→今回は、漢字モードを使用するので、１０００

２）8 ビットの文字数指示子：

→5 文字を符号化するので、０００００１０１

３）2 進 16 ビットシフトＪＩＳコードを 13 ビットに圧縮したもの：

→シフトＪＩＳコードを調べ、もしその文字が

・８１４０～９ＦＦＣの間にあれば、８１４０を引き、上位バイトにＣ０をかけ、それに下位バイトを加える。

・Ｅ０４０～ＥＢＢＦの間にあれば、Ｃ１４０を引き、上位バイトにＣ０をかけ、それに下位バイトを加える。

今回の場合は、

原→８ＣＢ４→（変換）→８Ｂ４→0100010110100

田→９３６３→（変換）→ＤＡ３→0110110100011

→８１４０→（変換）→　　０→0000000000000

経→８Ｃ６Ｆ→（変換）→８６Ｆ→0100001101111

道→９３Ｂ９→（変換）→ＤＦ９→0110111111001

４）終端ビット　0000

５）埋め草ビット 11101100　又は　00010001

→データビット列の長さが 104 に満たないとき、８ビット以下までを０で、それ以上を埋め草ビットで埋める。

よって今回符号化するビット列は、(1000000 00101010001011010001101101000110000000000000 00100001101111011011111100100000000000011101 1 0000010001)となる。これを８ビットごとの固定長で区切り、既約多項式 $x^8+x^4+x^3+x^2+1$ の根αを原始元とする $GF(2^8)$ 上でのベクトル表現からべき表現へ変換すると、

D1→10000000→α^7　　D2→01010100→α^{143}

D3→01011010→α^{19}　　D4→00110110→α^{249}

D5→10001100→α^{49}　　D6→00000000→α^{255}

D7→00001000→α^3　　D8→01101111→α^{61}

D9→01101111→α^{61}　　D10→11001000→α^{196}

D11→00000000→α^{255}　D12→11101100→α^{122}

D13→00010001→α^{100}　である。よって、ｍ＝

（α^7，α^{143}，α^{19}，α^{249}，α^{49}，α^{255}，α^3，α^{61}，α^{61}，α^{196}，α^{255}，α^{122}，α^{100}）

となる。1-Q 型では[255, 242, 14]RS 符号を短縮化し、2^8 元[26,13,14]ＲＳ符号を使う。生成多項式は、

$g(x)=(x-\alpha)(x-\alpha^2)(x-\alpha^3)(x-\alpha^4)(x-\alpha^5)(x-\alpha^6)$

$(x-\alpha^7)(x-\alpha^8)(x-\alpha^9)(x-\alpha^{10})(x-\alpha^{11})(x-\alpha^{12})$

生成行列ＧはＧ＝（ＩＰ）（Ｉは単位行列）と置くと、

$$P=\begin{pmatrix}
243 & 221 & 99 & 43 & 144 & 39 & 114 & 250 & 228 & 247 & 243 & 1 & 164\\
86 & 217 & 34 & 26 & 193 & 82 & 92 & 71 & 67 & 88 & 75 & 185 & 37\\
214 & 180 & 150 & 81 & 41 & 251 & 0 & 169 & 8 & 47 & 36 & 137 & 86\\
8 & 142 & 202 & 31 & 185 & 188 & 3 & 166 & 195 & 77 & 84 & 187 & 127\\
49 & 251 & 224 & 143 & 195 & 137 & 0 & 229 & 252 & 69 & 174 & 40 & 237\\
159 & 87 & 128 & 215 & 102 & 197 & 254 & 21 & 110 & 176 & 216 & 180 & 140\\
62 & 77 & 99 & 254 & 54 & 239 & 194 & 155 & 37 & 169 & 203 & 102 & 160\\
82 & 9 & 118 & 254 & 122 & 220 & 10 & 124 & 200 & 125 & 225 & 118 & 111\\
33 & 83 & 104 & 72 & 176 & 87 & 45 & 249 & 223 & 87 & 235 & 194 & 181\\
103 & 168 & 57 & 192 & 128 & 20 & 46 & 163 & 227 & 244 & 76 & 83 & 136\\
58 & 29 & 188 & 191 & 39 & 18 & 25 & 210 & 187 & 39 & 24 & 225 & 71\\
248 & 178 & 243 & 6 & 232 & 123 & 217 & 128 & 173 & 193 & 13 & 112 & 152\\
74 & 152 & 152 & 100 & 86 & 100 & 106 & 104 & 130 & 218 & 206 & 140 & 78
\end{pmatrix}$$

となる。（簡便のためのαの係数だけ書いた）

これより符号語は、ｃ＝ｍＧ＝

（α^7，α^{143}，α^{19}，α^{249}，α^{49}，α^{255}，α^3，α^{61}，α^{61}，α^{196}，α^{255}，α^{122}，α^{100} α^{19}，α^{243}，α^{225}，α^{143}，α^{43}，α^{98}，α^{84}，α^{235}，α^{199}，α^{223}，α^{22}，α^{235}，α^{122}）となる。

後ろ１３ビットをＥブロックに格納し、ベクトル表現に直すと、

α^{19}→E1→=01011010　　α^{243}→E2→=01111101

α^{225}→E3→=00100100　　α^{143}→E4→=01010100

α^{43}→E5→=01110111　　α^{98}→E6→=01000011

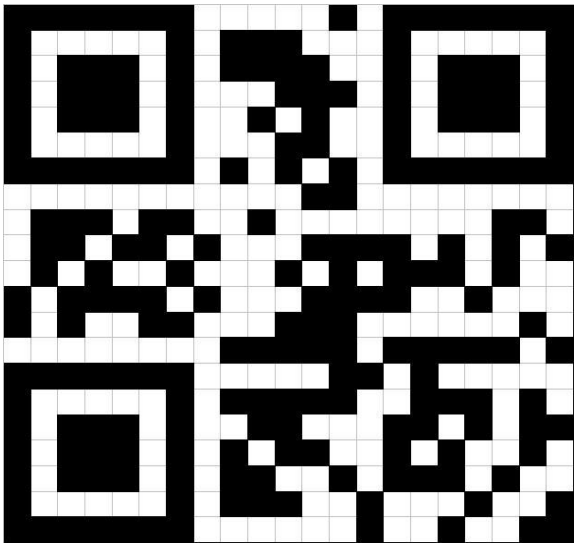
α^{84}→E7→=01101011　　α^{235}→E8→=11101011

α^{199}→E9→=00001110　　α^{223}→E10→=00001001

α^{22}→E11→=11101010　　α^{235}→E12→=11101011

α^{122}→E13→=11101100

以上、得られたＤ１～Ｄ１３とＥ１～Ｅ１３を規則にしたがって並べ、最後にマスク処理を行うと、以下のＱＲコードが完成する。



４．おわりに

今回の研究を通して、代数学が社会に応用されている様に、改めて感動した。また、今まで大学で学んだ数学の知識が、この研究を通じて社会とリンクできたことにも、とてもやりがいを感じた。

５．謝辞

６．参考文献

[1]池田和興、例題が語る符号理論、共立出版（２００７）

[2]G．A．ジョーンズ/Ｊ．M．ジョーンズ、情報理論と符号理論、シュプリンガー・ジャパン（２００６）

[3]濱屋進、符号理論入門　工学社（２００８）

[4]今井秀樹、符号理論、電子情報通信学会（１９９０）

[5]今井秀樹、情報・符号・暗号の数理、コロナ社（２００４）

[6]イエルン・ユステセン/トム・ホーホルト、誤り訂正符号入門、森北出版株式会社（２００５）

[7]松坂和夫、代数系入門、岩波書店（１９７６）

[8]ハーバート・シルト、独習Ｃ、翔泳社（１９９４）

[9]堀部安一　Information & Computing のための数理序論

[１０]堀部安一　ベクトルと行列―線形の代数・幾何入門

[１１]ダレル・ハーディ/キャロル・ウォーカー、応用代数学入門　ピアソン・エデュケーション　（２００５）

[１２]平松豊一、応用代数学、裳華房（１９９７）

[１３]藤原良・神保雅一,符号と暗号の数理、共立出版（１９９３）