Cyclic Codes

- A cyclic code is a parity-check code.
- e.g. cyclic shifts $0001101 \rightarrow 1000110 \rightarrow 0100011 \rightarrow 1010001 \ldots$
- an example cyclic code consists of 4 codewords 000, 110, 011, 101
- a cyclic shift of a codeword is also a valid codeword
- the code is also linear, in that the modulo 2 sum of two code words is also a valid code word, e.g. $110+011=101$
- note that in modulo 2 arithmetic sum and difference are equivalent, and also expressed as a bit-wise XOR
- Cyclic codes are relatively easy to implement in hardware with shift registers

Polynomial representation

The n-length code word polynomial corresponding to $\mathbf{c} = (c_1, c_1, c_2, \ldots, c_n)$ is

$$c(x) = \sum_{i=1}^{n} c_i x^{n-i} = c_1 x^{n-1} + c_2 x^{n-2} + \cdots + c_n$$

The code word polynomial corresponding to a cyclic shift of $\mathbf{c}$:
$\mathbf{c}^{(1)} = (c_2, c_3, \ldots, c_n, c_1)$ is

$$c^{(1)}(x) = c_2 x^{n-1} + c_3 x^{n-2} + \cdots + c_n x + c_1 = xc(x) + c_1(x^n + 1)$$

Therefore

$$c^{(1)}(x) \mathrm{mod}(x^n + 1) = xc(x) \mathrm{mod}(x^n + 1)$$

Cyclic Code Theorem

and on continuing with cyclic shifts,

$$c^{(i)}(x)\mathrm{mod}(x^n+1) = x^i c(x)\mathrm{mod}(x^n+1)$$

In any (n,k) cyclic code all code word polynomials are multiples of a polynomial of degree $(n-k)$ of the form

$$g(x) = x^{n-k} + g_2 x^{n-k-1} + \ldots + g_{n-k}x + 1$$

called the *generator polynomial* where $g(x)$ is a factor of (x^n+1). Furthermore, for any information sequence $\mathbf{p} = (p_1, p_2, \ldots, p_{k-1}, p_k)$, we have the information sequence polynomial

$$P(x) = p_1 x^{k-1} + p_2 x^{k-2} + \ldots + p_{k-1}x + p_k$$

and the code word polynomial corresponding to $\mathbf{p}$ is given by $c(x) = P(x)g(x)$.

Cyclic Code Example

To generate a (7,4) cyclic code we need a generator polynomial of degree $(n-k)$=3, which has to divide (x^7+1). Since,

$$x^7+1=(x+1)(x^3+x^2+1)(x^3+x+1)$$

we could choose either (x^3+x^2+1) or (x^3+x+1). We will use $g(x)=(x^3+x^2+1)$ and multiply it by all $P(x)$ polynomials of the form

$$P(x)=p_1x^3+p_2x^2+p_3x+p_4$$

where the p_is can be 0 or 1. This generates 16 code word polynomials from $c(x)=P(x)g(x)$ and hence the complete set of code words.

These are 0000000, 1111111, seven codes from 0001101 and cyclic shifts, and seven codes from 1110010 and cyclic shifts and is equivalent to Hamming (7,4).

Cyclic Redundancy Check

A cyclic redundancy check (CRC) is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to raw data. It uses a generator polynomial of a degree equal to the number of check bits to be used.

Cyclic Redundancy Check

Example: use $g(x) = (x^3 + x^2 + 1)$ on the 9-bit data 110010110. We perform modulo 2 long division on the message with 3 check bits set to 000 to determine the remainder

	1	1	0	0	1	0	1	1	0	0	0	0
$\oplus$	1	1	0	1								
	0	0	0	1	1	0	1	1	0	0	0	0
$\oplus$				1	1	0	1					
	0	0	0	0	0	0	0	1	0	0	0	0
$\oplus$								1	1	0	1	
	0	0	0	0	0	0	0	0	1	0	1	0
$\oplus$									1	1	0	1
	0	0	0	0	0	0	0	0	0	1	1	1

Cyclic Redundancy Check

Our 3-bit check value (remainder) is 111. To validate the message we perform the calculation again replacing the 000 padding with the check value

	1	1	0	0	1	0	1	1	0	1	1	1
$\oplus$	1	1	0	1								
	0	0	0	1	1	0	1	1	0	1	1	1
$\oplus$				1	1	0	1					
	0	0	0	0	0	0	0	1	0	1	1	1
$\oplus$								1	1	0	1	
	0	0	0	0	0	0	0	0	1	1	0	1
$\oplus$									1	1	0	1
	0	0	0	0	0	0	0	0	0	0	0	0

The remainder should equal zero if there are no detectable errors.

Cyclic Code for correcting errors

- Suppose that a transmitted code word $c(x) = P(x)g(x)$ is received with one or more errors

$$r(x) = P(x)g(x) + e(x)$$

where $e(x)$ contains one or more nonzero terms.
- Define the syndrome potential $S(x)$ as the remainder when the received polynomial is divided by the generator polynomial $g(x)$

$$S(x) = r(x) \mathrm{mod} g(x) = e(x) \mathrm{mod} g(x)$$

- If $S(x)$=0 there are no errors detected. A non-zero $S(x)$ indicates an error is detected, and can be indexed to the position(s) of the error in $r(x)$.
- $r(x)$ can then be corrected by flipping the appropriate bit(s).

Cyclic Code Example

Take the (7,4) cyclic code with generator polynomial $(x^3 + x^2 + 1)$. Consider the code word 0101110 with a one-bit error 010**0**110.

$$\begin{array}{c|ccccccc}
 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
\oplus & & 1 & 1 & 0 & 1 & & \\ \hline
 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
\oplus & & & 1 & 1 & 0 & 1 & \\ \hline
 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\oplus & & & & 1 & 1 & 0 & 1 \\ \hline
 & 0 & 0 & 0 & 0 & 1 & 0 & 1
\end{array}$$

The remainder 101 is indexed to bit position 4 for this code.

The seven non-zero permutations of the syndrome $S(x)$ correspond to the seven bit positions in the codeword, and hence the (7,4) code can correct all single bit errors only.

Cyclic Code Implementation

Can implement with shift registers

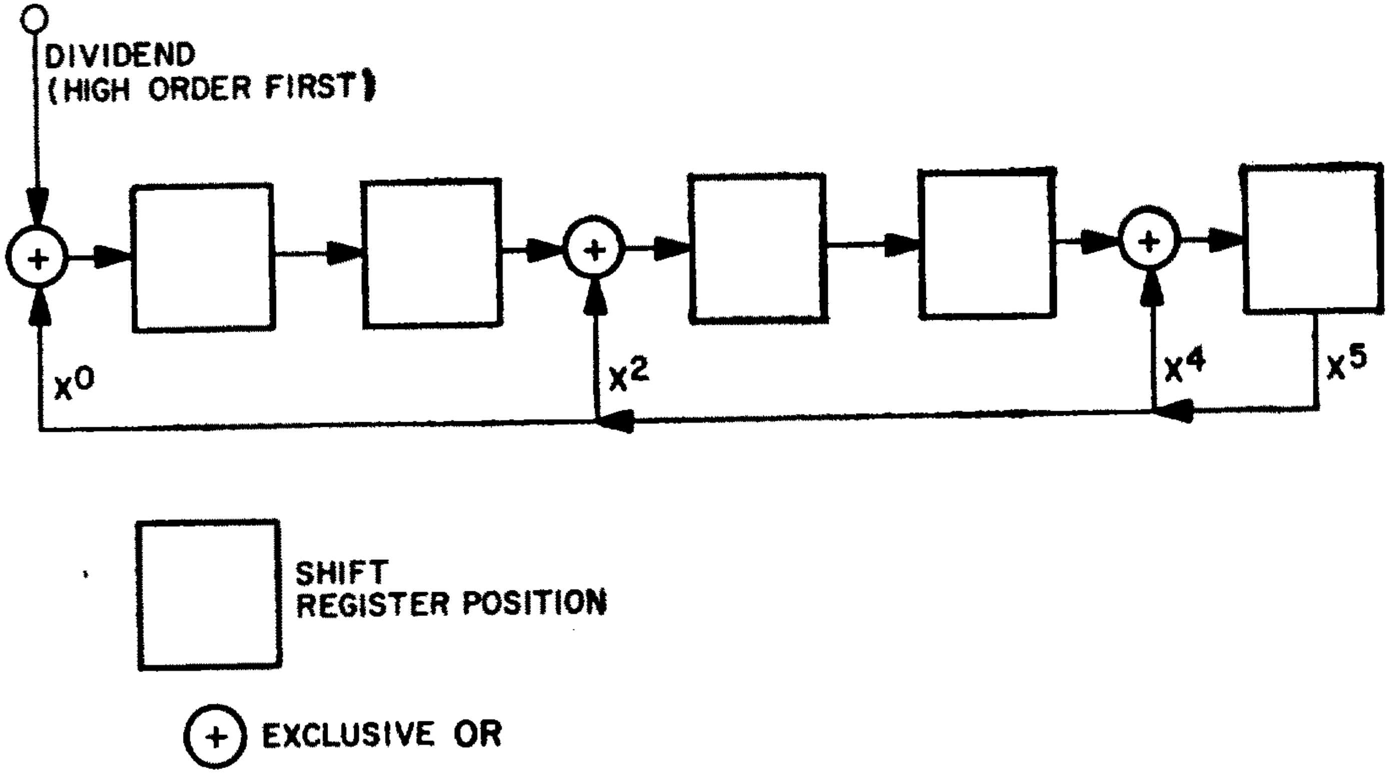


Fig. 1—A shift register for dividing by $1+X^2+X^4+X^5$.

Cyclic Code Implementation

Can implement with shift registers

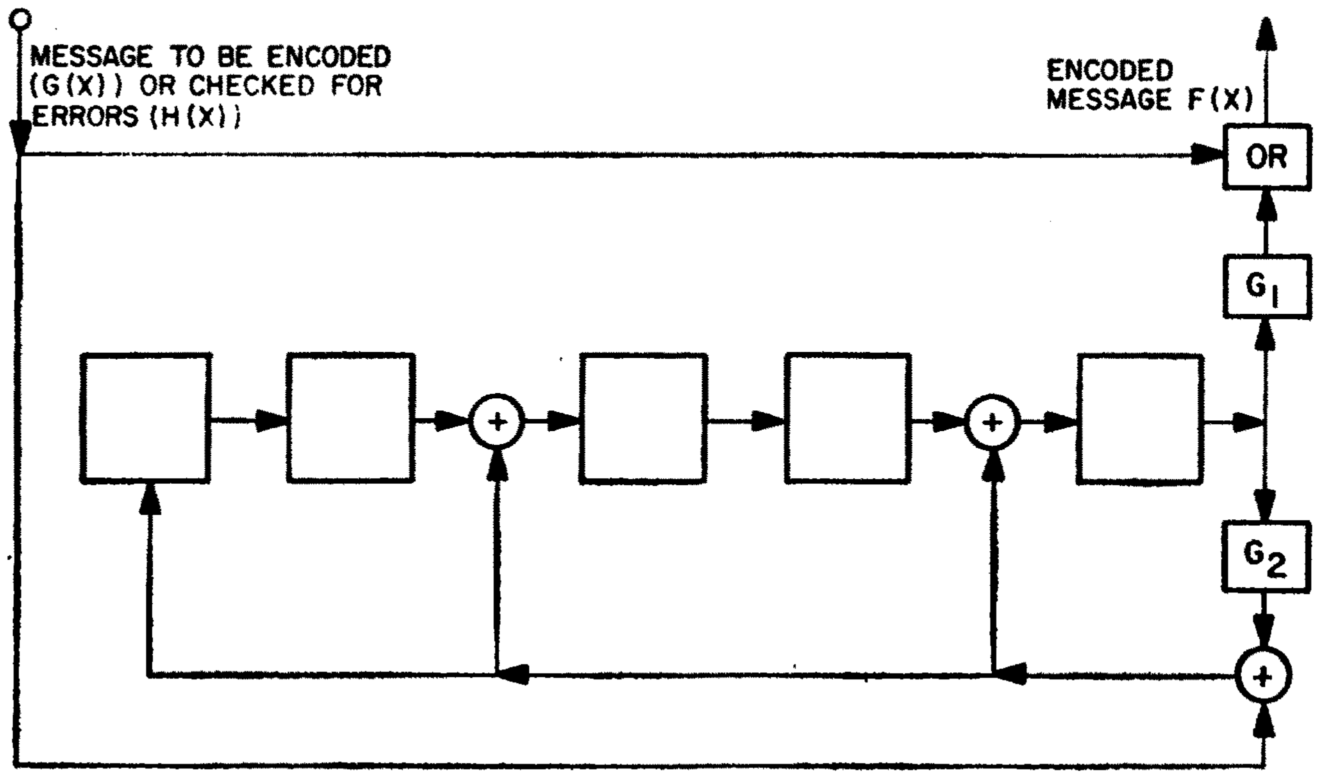


Fig. 3—A more efficient circuit for encoding and error detection. (In this example, $P(X)=1+X^2+X^4+X^5$.)

Cyclic Code for correcting Burst Errors

- From Hamming distance concept, a code with minimum distance 2t+1 can correct any t errors.
- A linear block code that corrects all burst errors of length t or less must have at least 2t check symbols.

BCH Codes

- Bose, Ray-Chaudhuri and Hocquenghem (BCH) codes are a subclass of cyclic codes that can be designed for correction of t errors. These codes are versatile in design and there exists an efficient decoding algorithm.
- For any m and t, there exists a BCH code with $n = 2^m - 1$, $(n-k) \leq mt$ and $d_{\text{min}} = 2t + 1$.
- for example, t=2, m=4 provides a (15,7) BCH code with a generator polynomial corresponding to 111010001.
- Reed-Solomon codes are a subset of BCH codes. Reed-Solomon codes are non-binary codes with elements belonging to a q-ary alphabet, where $q = 2^k$ is a power of 2.
- For an (N, K) Reed-Solomon code, K q-ary symbols are mapped into N q-ary symbols.

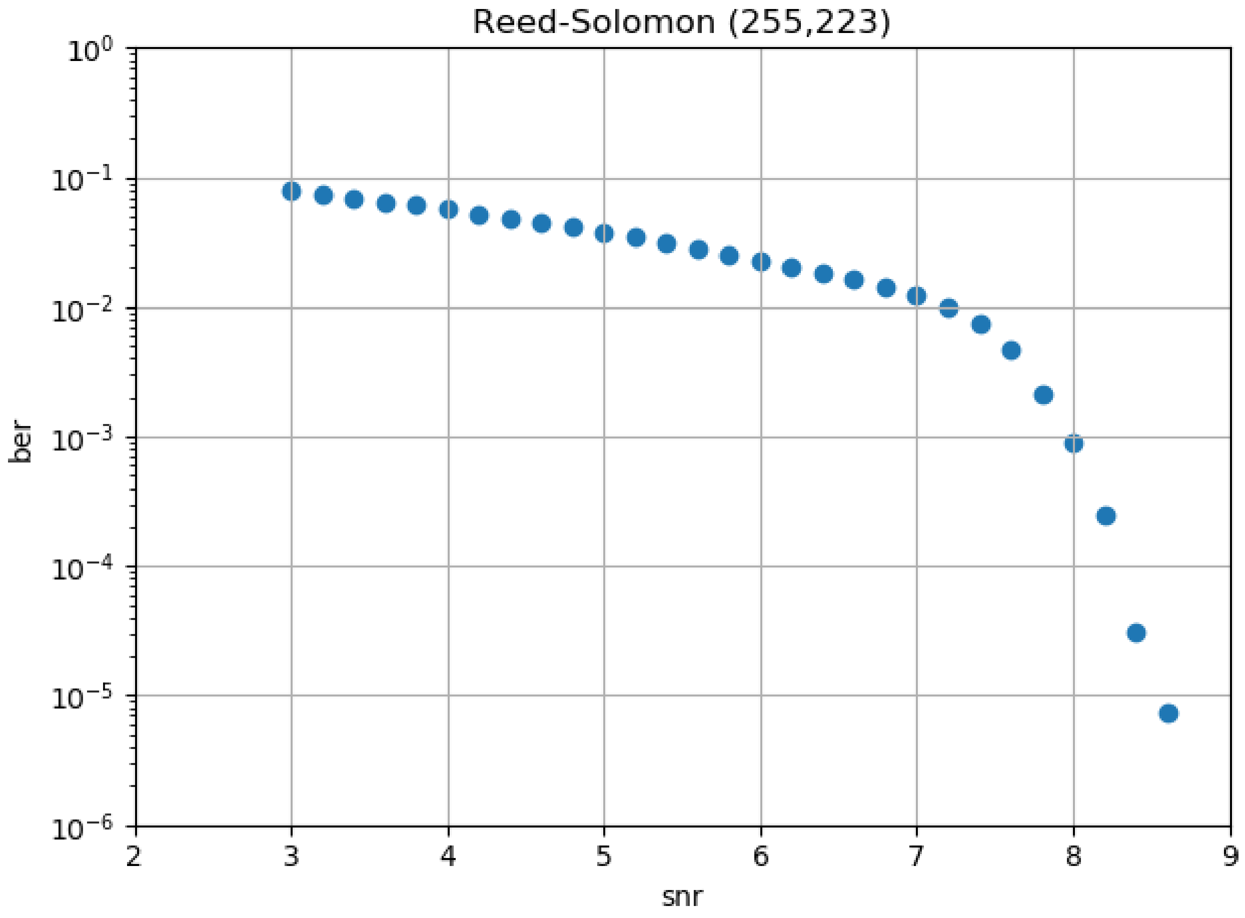
Reed-Solomon Codes

Reed-Solomon codes are block-based error correcting codes with a wide range of applications in digital communications and storage:

- Storage devices (including tape, Compact Disk, DVD, barcodes, etc)
- Wireless or mobile communications (including cellular telephones, microwave links, etc)
- Satellite communications
- Digital television / DVB
- High-speed modems such as ADSL, xDSL, etc.

Example: A popular Reed-Solomon code is RS(255,223) with s=8-bit symbols. Each codeword contains n=255 code word bytes, of which k=223 bytes are data and $n-k$=2t=32 bytes are parity. For this code the decoder can correct any t=16 symbol errors in the code word: i.e. errors in up to 16 bytes anywhere in the codeword can be automatically corrected. Reed-Solomon codes are particularly well suited to correcting burst errors.

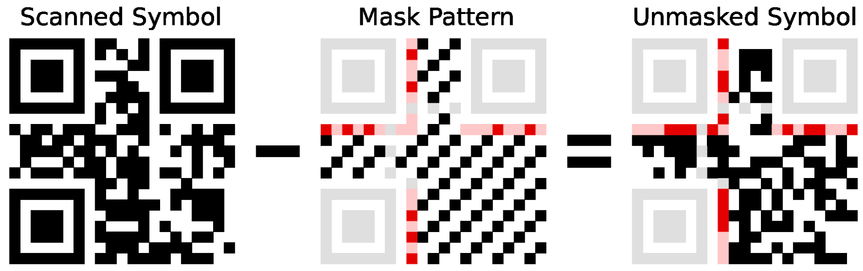
Reed-Solomon Codes

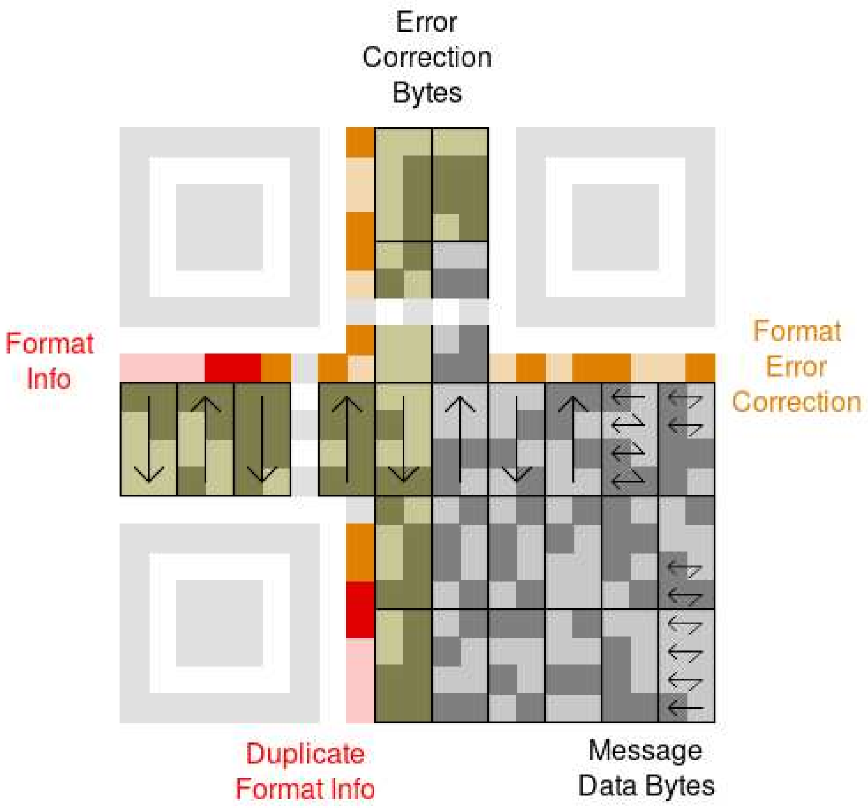


`https://pypi.org/project/reedsolo`

Example: QR Codes

A QR code (abbreviated from Quick Response code) is a type of matrix barcode (or two-dimensional barcode) first designed in 1994.





Level L	up to 7% data bytes
Level M	up to 15% data bytes
Level Q	up to 25% data bytes
Level H	up to 30% data bytes

The QR specification defines the block sizes so that no more than 30 error-correction symbols appear in each block.

“Unmasked” QR code v1(M)