⇒ Special cyclic codes:

(1) BCH code [Bose - Chaudhuri - Hocquenghem] code:

→ most efficient error-correcting cyclic codes constructed using polynomials

→ It offers flexibility in choice of code parameters, i.e, block length & code rate.

m → any +ve integer ($m \geq 3$) t → smaller than integer

then ∃ a binary BCH code with the following parameters:

Block length: $n = 2^m - 1$

no: of data length: $n - k \leq mt$

Min. distance: $d_{min} \geq 2t + 1$

→ Hamming single-error correcting codes can be described as BCH code with $t = 1$

$(n - k = m)$.

→ ~~Here~~ precise control over the no: of symbol errors correctable by the code.

i.e, it is possible to design binary BCH codes that can correct multiple bit errors.

⇒ decoding is easy through an algebraic method called syndrome decoding.

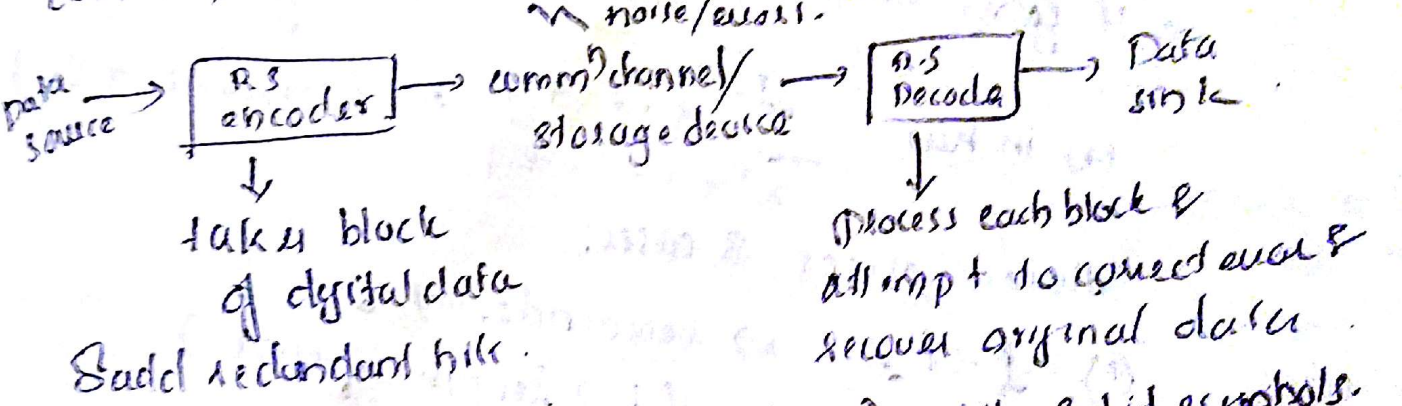
* Applicn: satellite comm, CD player, SSDs.

(2) RS (Reed - Solomon Code):

→ subclass of non-binary BCH codes

→ encoder operates on multiple bits rather than individual bits.

→ block-based error correcting codes used to correct errors in: storage devices (CD, DVD), wireless or mobile comm^n, satellite comm^n.

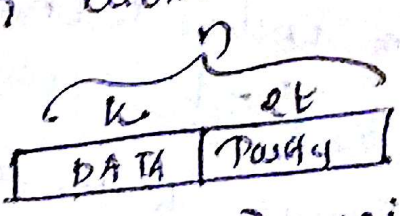


→ Code is specified as RS(n, k) with s-bit symbols.

→ encoder takes k-data symbols of s-bits each & add parity symbols to make an n-symbol codeword. There are n−k parity symbols of s bits each.

⇒ An RS decoder can correct upto t-symbols that contain errors in a codeword, where

$2t = n-k$.



t ↑ → large no: of errors can be corrected.

Eg: RS(255, 223) with 8-bit symbols.

Each codeword contains 255 codeword bytes.

223 bytes for data & 32 bytes for parity

$n-k \quad 2t = 32 \quad t = 16$

t = 16 → decoder can correct any 16 symbol errors in code. i.e, errors up to 16 bytes anywhere in the message can be automatically corrected.

symbol-size s

max. c.w length ; $n = 2^s - 1$

$s \rightarrow$ 8-bit symbol $\quad n = 2^8 - 1 = 255$ bytes

In CRC *

Burst errors:

$\Rightarrow e(x) = x^j + \ldots + x^i$

$\Rightarrow e(x) = x^i(x^{j-i} + \ldots + 1)$

→ If $g(x)$ detect a single error. → i.e, $\frac{x^p}{g(x)}$ = remainder

i.e, in hue. $\frac{x^{j-i} + \ldots + 1}{x^r + \ldots + 1}$ = (remainder $\neq 0$)

thus arises 3 cases:

(1) $j - i < r \Rightarrow$ remainder $\neq 0$:

$j - i = L - 1$ ($L \rightarrow$ length of error)

$L - 1 < r \Rightarrow L \leq r + 1 \Rightarrow L \leq r$

* All burst errors with $L \leq r$ will be detected.

(2) In some rare cases

if $j - i = r$ or $L = r + 1$

syndrome = 0 (error remain undetected).

* probability of undetected burst error of length $r+1$ is $\left(\frac{1}{2}\right)^{r-1}$

i.e All burst errors with $L = r+1$ will be detected with a probability $1 - \left(\frac{1}{2}\right)^{r-1}$.

Eg:
$g(x) = x^{14} + x^3 + 1$ ←
$r = 14$
burst $L = 15$ can slip undetected with probability $\left(\frac{1}{2}\right)^{15-1}$ or $1/10,000$

Eg:
$g(x) = x^{14} + x^3 + 1$

(3) In some rare cases.

$j - i > r$ or $L > r+1 \Rightarrow S = 0$ (error undetected)

* probability of undetected burst error with $L > r+1$ is $\left(\frac{1}{2}\right)^r$.

* All burst errors with $L > r+1$ will be detected with probability $1-(\frac{1}{2})^r$.