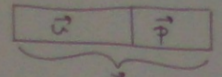
Systematic encoding of cyclic code:

(n,k) code : $u(x)$: $\deg[u(x)] = k-1$

$\Rightarrow v(x) = u(x) \cdot g(x)$

$= u(x) \oplus x^k \cdot p(x)$



Ex: $u(x) = x^3 \oplus x$ | $p(x) = x \oplus 1$ $\quad x^6 x^5 \dots x^0$

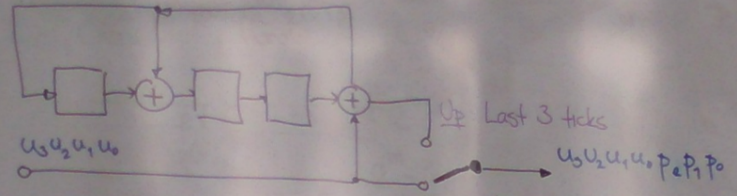
$k=4, n=7$ $\quad \vec{v} = (0101110)$

$v(x) = u(x) \oplus x^4 \cdot p(x) =$

$= x^3 \oplus x \oplus x^4(x \oplus 1)$

$= x^5 \oplus x^4 \oplus x^3 \oplus x$

Ex: Systematic encoder for Hamming (7,4)



Decoder for cyclic codes

→ syndrome decoding

table : syndrom poly. $s(x)$ → error poly. $e(x)$

$v(x) = u(x) \cdot g(x)$

$v(x) = v(x) \oplus e(x)$

$v(x) \cdot h(x) \equiv 0 \mod x^n \oplus 1$

decoder : $v(x) : g(x) =$

$= v(x) : g(x) \oplus e(x) : g(x)$

$= \emptyset \oplus \boxed{s(x)}$

Ex: Mapping table for Hamming (7,4)

$e(x)$	$s(x)$
0	0
1	1
x	x
x^2	x^2
x^3	$x \oplus 1$
x^6	$x^2 \oplus 1$

↓ corrects single err

↓ $e(x) = x^i$

not practical for high t

Trick : Maggit (1960)

$v(x) \rightarrow v^{(1)}(x) \equiv x \cdot v(x) \mod x^n \oplus 1$

$s(x) \rightarrow s^{(1)}(x) \equiv x \cdot s(x) \mod x^n \oplus 1$

⇒ a shift of $v(x)$ shifts also the $s(x)$

Maggit decoder

for cyclic code of length n, checks all n shifts of $v(x)$ against syndromes with $\deg[e(x)] = \max$.

↳ only $x^6 \Leftrightarrow x^2 \oplus 1$ for Hamming $(n=7)$

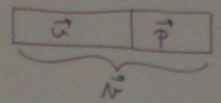
but $x^{14} \dots x^{13} \oplus x^{14} = e(x)$ for Golay $(n=15)$

Systematic encoding of cyclic code:

(n,k) code : $u(x)$: $\deg[u(x)] = k-1$

$\rightarrow v(x) = u(x) \cdot g(x)$

$= u(x) \oplus x^k \cdot p(x)$



Ex: $u(x) = x^3 \oplus x$ | $p(x) = x \oplus 1$ $\quad x^6 x^5 \dots x^0$

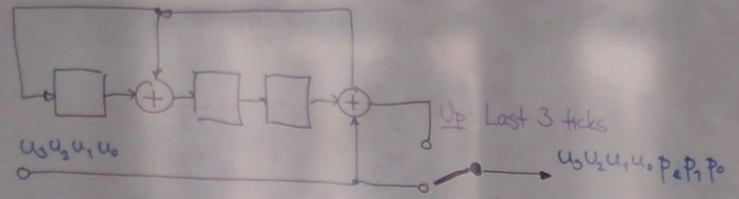
$k=4, n=7$ $\quad \vec{v} = (0101110)$

$v(x) = u(x) \oplus x^4 \cdot p(x) =$

$= x^3 \oplus x \oplus x^4(x \oplus 1)$

$= x^5 \oplus x^4 \oplus x^3 \oplus x$

Ex: Systematic encoder for Hamming (7,4)



Decoder for cyclic codes

→ syndrome decoding

table : syndrom poly. $s(x)$ → error poly. $e(x)$

$v(x) = u(x) \cdot g(x)$

$w(x) = v(x) \oplus e(x)$

$w(x) \cdot h(x) \equiv 0 \mod x^n \oplus 1$

decoder : $w(x) : g(x) =$

$= v(x) : g(x) \oplus e(x) : g(x)$

$= 0 \oplus s(x)$

Ex: Mapping table for Hamming (7,4)

$e(x)$	$s(x)$
0	0
1	1
x	x
x^2	x^2
x^3	$x \oplus 1$
x^4	$x^2 \oplus 1$

↓ corrects single err

$e(x) = x^i$

not practical for high t

Trick : Maggit (1960)

$w(x) \rightarrow w^{(1)}(x) \equiv x \cdot w(x) \mod x^n \oplus 1$

$s(x) \rightarrow s^{(1)}(x) \equiv x \cdot s(x) \mod x^n \oplus 1$

⇒ a shift of $w(x)$ shifts also the $s(x)$

Maggit decoder

for cyclic code of length n, checks all n shifts of $w(x)$ against syndromes with $\deg[e(x)] = \max$.

↳ only $x^6 \Leftrightarrow x^2 \oplus 1$ for Hamming $(n=7)$

but $x^{14} \dots x^{13} \oplus x^{14} = e(x)$ for Golay $(n=15)$

Hoggat algorithm:

1. move all n bits of $w(x)$ into LSR
2. compute $s(x)$ while shifting by $w(x) : g(x)$
3. if $s(x) = 0 \Rightarrow$ do nothing
   else: check for $s(x)$ pattern
   if match: add $e(x)$ to the current $w(x)$
   $w'^{(i)} \leftarrow w^{(i)} \oplus e(x)$
4. continue shifting

$\Rightarrow$ most significant bit is the one that contains error

most sig. bit: $v = (0100101)$ MSB

Ex: $\vec{v} = (00100101)$

$\vec{w} = (00110101)$

1. 0 0 1 1 0 1 0 1 → $s(x) \neq 0$, no match
2. 1 0 0 1 1 0 1 0 → —"—
3. 0 1 0 0 1 1 0 1 → —"—
4. 1 0 1 0 0 1 1 0 → —"—
5. 0 1 0 1 0 0 1 1 → $s(x)$ matches
   0 1 0 1 0 0 1 0 Okay: 01011010
6. 0 0 1 0 1 0 0 1

⋮

etc

Can we construct a cyclic code with given t and k?

↑ number of corrected errors

BCH codes (Bose, Chaudhuri, Hocquenghem 1960)

given: $m, m \geq 3$; $t < 2^m - 1$

provides:
- block length $n = 2^m - 1$
- parity symbols $n - k \leq m \cdot t$
- min. dist. $d_{min} \geq 2 \cdot t + 1$
- generator polynomial

d_{min} may be higher!

Reed-Solomon codes (1960)

→ a BCH with s bit long symbols

widely used: cca since 1990 (decoding)

- storage (SSD, RAID) CD, DVD, BD
- wireless networks (WiMAX)
- satellite communications (DVB-S, NASA)
- bar-codes (QR)
- ADSL, xDSL

Hagglt algorithm:

1. move all n bits of $w(x)$ into LSR
2. compute $s(x)$ while shifting by $w(x)$: $g(x)$
3. if $s(x) = 0 \Rightarrow$ do nothing
   else: check for $s(x)$ pattern
   if match: add $e(x)$ to the current $w(x)$
   $w'(i) \leftarrow w^{(i)} \oplus e(x)$
4. continue shifting

$\Rightarrow$ most significant bit is the one that contains error

most sig. bit: $\vec{w} = (0100101)$ MSB

Ex: $\vec{w} = (00100101)$

$\vec{w} = (00110101)$

① 0 0 1 1 0 1 0 1 → $s(x) \neq 0$, no match
② 1 0 0 1 1 0 1 0 → —"—
③ 0 1 0 0 1 1 0 1 → —"—
④ 1 0 1 0 0 1 1 0 → —"—
⑤ 0 1 0 1 0 0 1 1 → $s(x)$ matches
0 1 0 1 0 0 1 0 Golay: 0 1 0 1 1 1 0 1 0
⑥ 0 0 1 0 1 0 0 1

⋮

etc

Can we construct a cyclic code with given t and k?

t: number of corrected errors

BCH codes (Bose, Chaudhuri, Hocquenghem 1960)

given: $m, m \geq 3$; $t < 2^m - 1$

provides:
- block length $n = 2^m - 1$
- parity symbols $n - k \leq m \cdot t$
- min. dist. $d_{min} \geq 2t + 1$
- generator polynomial

d_{min} may be higher!

Reed-Solomon codes (1960)

→ a BCH with s bit long symbols

widely used: cca since 1970 (decoding)

- storage (SSD, RAID) CD, DVD, BD
- wireless networks (Wi, MAX)
- satellite communications (DVB-S, MASK)
- bar-codes (QR)
- ADSL, xDSL