5 Ciklično kodiranje (ang. Cyclic Redundancy Code ali CRC)

- Zaporedje binarnih simbolov dolžine $k+1$,

$$b_k\ b_{k-1}\ b_{k-2}\ \ldots\ b_i\ \ldots\ b_1\ b_0$$

predstavimo s polinomom $P_k(x)$ stopnje k:

$$P_k(x) = b_k x^k + b_{k-1}x^{k-1} + \cdots + b_i x^i + \cdots + b_1 x^1 + b_0.$$

Binarni simboli

$$b_i = \{0, 1\}\ \ (i = k, k-1, \ldots, 1, 0)$$

so koeficienti polinoma.

Primer: Simboli: 1 1 0 0 1 1,
Polinom: $P_5(x) = 1x^5 + 1x^4 + 0x^3 + 0x^2 + 1x^1 + 1 = x^5 + x^4 + x^1 + 1.$

- Delilno zaporedje dolžine $r+1$ predstavimo s polinom $G_r(x)$ stopnje r (generator).
- Zaporedju $(k+1)$ simbolov pripišemo r simbolov z vrednostjo nič, $P_k(x) \times x^r$.
- Polinom $P_k(x) \times x^r$ delimo (in sicer po modulu 2) z generatorjem. Dobimo rezultat deljenja $Q(x)$ in ostanek $R(x)$, ki je največ stopnje $(r-1)$:

$$\frac{P_k(x) \times x^r}{G_r(x)} = Q(x) + \frac{R(x)}{G_r(x)}.$$

- Odštejemo/prištejemo (po modulu 2) ostanek deljenja $R(x)$,

$$T(x) = P_k(x) \times x^r - R(x) = P_k(x) \times x^r + R(x),$$

$$P_k(x) \times x^r = Q(x) \times G_r(x) + R(x).$$

$$P_k(x) \times x^r + R(x) = Q(x) \times G_r(x).$$

- Zaporedje $T(x)$ pošljemo v kanal. Sprejemnik sprejme $T(x)'$,

$$T(x)' = T(x) + E(x).$$

V polinomu $E(x)$ so zajete napake. Ko napak ni, je $E(x)$ identično nič. Vsaka napaka na simbolu prispeva en člen k polinomu $E(x)$.

- Sprejemnik preveri pogoj deljivosti - deli sprejeto zaporedje $T'(x)$ z $G_r(x)$,

$$\frac{T(x)'}{G_r(x)} = \frac{T(x) + E(x)}{G_r(x)}.$$

Ker je $T(x)$ deljiv z $G_r(x)$, se deljenje izide, če napake ni ($E(x) = 0$) ali, če je polinom napak $E(x)$ deljiv z $G_r(x)$ brez ostanka.

Takih napak se ne da odkriti.

Primer:

Informacijskemu zaporedju desetih simbolov

$$1\,1\,0\,1\,0\,1\,1\,0\,1\,1$$

bomo določili kontrolne simbole. Naj bo delilno zaporedje

$$1\,0\,0\,1\,1.$$

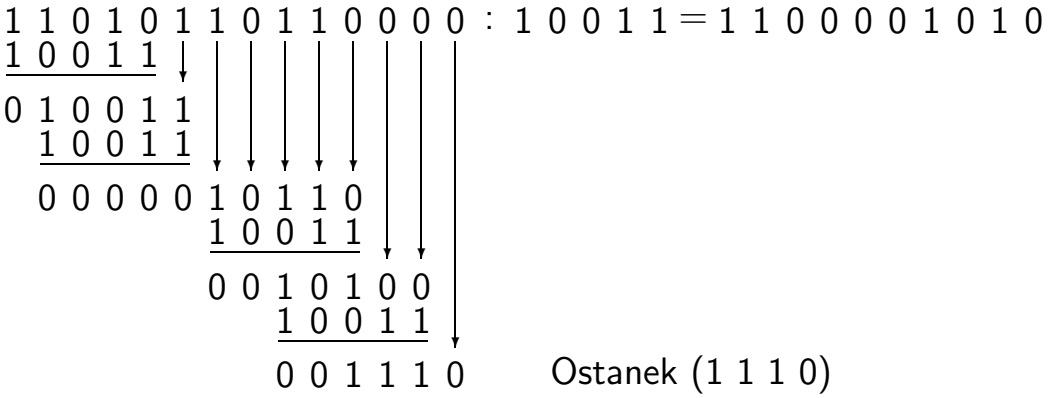
Torej ustreza informacijskemu zaporedju polinom

$$P_9(x) = x^9 + x^8 + x^6 + x^4 + x^3 + x^1 + 1$$

in generatorjev polinom je

$$G_4 = x^4 + x^1 + 1.$$

Informacijskemu zaporedju dodamo štiri ničle ($r = 4$) in ga delimo z delilnim zaporedjem. Čeprav iščemo samo ostanek deljenja, bomo računali tudi rezultat deljenja. Računamo po modulu dva:



V kanal pošljemo zaporedje:

$$1\,1\,0\,1\,0\,1\,1\,0\,1\,1\,1\,1\,1\,0.$$

Sposobnost odkrivanja napak s cilkličnim preverjanjem je odvisna od generatorjevega polinoma.

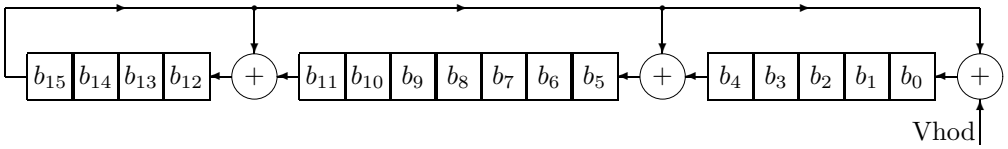
Nekateri standardni polinomi:

$$\begin{aligned} CRC-12 &= x^{12}+x^{11}+x^3+x^2+x^1+1 \\ CRC-16 &= x^{16}+x^{15}+x^2+1 \\ CRC-CCITT &= x^{16}+x^{12}+x^5+1. \\ Ieee802.3 &= x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1 \end{aligned}$$

Vezja za ciklično preverjanje

Primer:

$$CRC-CCITT = x^{16}+x^{12}+x^5+1.$$



Poenostavljena shema.

Za odkrivanje napak je potrebno enako vezje in enak postopek.

Vezja, ki se praktično uporabljajo, so nekoliko drugačna.

Ciklično preverjanje s tabelo ostankov

- $G_{16}(x)$ generatorjev polinom stopnje 16,
- $P(x)$ začetnih nekaj bajtov okvirja (sporočila),
- $R(x)$ trenutni ostanek deljenja.

$$P(x)x^{16} = Q(x)G_{16}(x) + R(x).$$

Dodajmo k $P(x)$ še naslednji bajt okvirja.

$$P'(x) = P(x)x^8 + B(x),$$

$$B(x) = b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x^1 + b_0.$$

$$P'(x)x^{16} = Q'(x)G_{16}(x) + R'(x).$$

$$\begin{aligned}
[P(x)x^8 + B(x)]x^{16} &= P(x)x^{16}x^8 + B(x)x^{16} = \\
&= [Q(x)G_{16}(x) + R(x)]x^8 + B(x)x^{16} = \\
&= Q(x)G_{16}(x)x^8 + R(x)x^8 + B(x)x^{16} = \\
&= Q(x)G_{16}(x)x^8 + [R(x) + B(x)x^8]x^8.
\end{aligned}$$

$$R(x) = R_H(x)x^8 + R_L(x)$$

$$[R(x) + B(x)x^8]x^8 = R_L(x)x^8 + [R_H(x) + B(x)]x^{16}.$$

$$R'(x) = R_L(x)x^8 + Ostanek\{[R_H(x) + B(x)]x^{16}]\}$$

$$V(x)x^{16} = [R_H(x) + B(x)]x^{16}$$

Algoritem za ciklično preverjanje s tabelo ostankov T:

1. postavi začetni ostanek na nič, $R = 0$,
2. Prištej po modulu 2 naslednji bajt sporočila k zgornjemu bajtu ostanka, $V = R_H + B$.

   Vrednost V je indeks v tabelo ostankov T, vrednost ostanka je $T[V]$.
3. Pomakni R za osem mest levo. S tem se zgornji del R_H izgubi.

   Rezultat je R_L pomaknjen za osem mest levo.
4. prištej po modulu 2 vrednost ostanka $T[V]$ k R, $R = R + T[V]$.
5. ponavljaj korake 2-4 do konca okvirja.

Tabela ostankov

```
0000 1021 2042 3063 4084 50a5 60c6 70e7 8108 9129 a14a b16b c18c d1ad e1ce f1ef
1231 0210 3273 2252 52b5 4294 72f7 62d6 9339 8318 b37b a35a d3bd c39c f3ff e3de
2462 3443 0420 1401 64e6 74c7 44a4 5485 a56a b54b 8528 9509 e5ee f5cf c5ac d58d
3653 2672 1611 0630 76d7 66f6 5695 46b4 b75b a77a 9719 8738 f7df e7fe d79d c7bc
48c4 58e5 6886 78a7 0840 1861 2802 3823 c9cc d9ed e98e f9af 8948 9969 a90a b92b
5af5 4ad4 7ab7 6a96 1a71 0a50 3a33 2a12 dbfd cbdc fbbf eb9e 9b79 8b58 bb3b ab1a
6ca6 7c87 4ce4 5cc5 2c22 3c03 0c60 1c41 edae fd8f cdec ddcd ad2a bd0b 8d68 9d49
7e97 6eb6 5ed5 4ef4 3e13 2e32 1e51 0e70 ff9f efbe dfdd cffc bf1b af3a 9f59 8f78
9188 81a9 b1ca a1eb d10c c12d f14e e16f 1080 00a1 30c2 20e3 5004 4025 7046 6067
83b9 9398 a3fb b3da c33d d31c e37f f35e 02b1 1290 22f3 32d2 4235 5214 6277 7256
b5ea a5cb 95a8 8589 f56e e54f d52c c50d 34e2 24c3 14a0 0481 7466 6447 5424 4405
a7db b7fa 8799 97b8 e75f f77e c71d d73c 26d3 36f2 0691 16b0 6657 7676 4615 5634
d94c c96d f90e e92f 99c8 89e9 b98a a9ab 5844 4865 7806 6827 18c0 08e1 3882 28a3
cb7d db5c eb3f fb1e 8bf9 9bd8 abbb bb9a 4a75 5a54 6a37 7a16 0af1 1ad0 2ab3 3a92
fd2e ed0f dd6c cd4d bdaa ad8b 9de8 8dc9 7c26 6c07 5c64 4c45 3ca2 2c83 1ce0 0cc1
ef1f ff3e cf5d df7c af9b bfba 8fd9 9ff8 6e17 7e36 4e55 5e74 2e93 3eb2 0ed1 1ef0
```