(52') 35 min

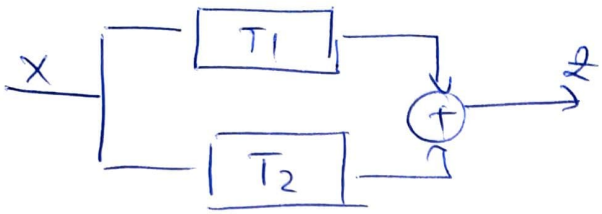
30413

# Problem 4

GF(5)

$T_1 = 1 + \Delta^3$ $\quad T_2 = 1 + 2\Delta + \Delta^4$

$T_3$ is for (30413)

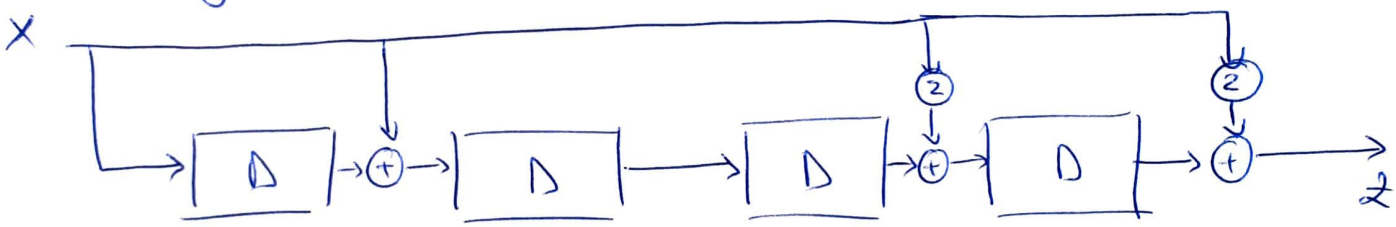



a) the transfer function is calculated according to the formula:

$$T_3 = T_1 + T_2 = 1 + \Delta^3 + 1 + 2\Delta + \Delta^4 =$$

$$= 2 + 2\Delta + \Delta^3 + \Delta^4$$

drawing the schematic:




c) the impulse response of $T_3$ is $h$, which is the response to the input sequence 1000... and in our case $h = 22011$

d) We have the input sequence: 2123

- in preparation: $2 * (22011) = 44022$

$3 * (22011) = 11033$

$h = 22011$

| | |
|---|---|
| $2h$ | 44022 |
| $2\Delta h$ | 44022 |
| $\Delta^3 h$ | 220 |

$h = 22011$

| | |
|---|---|
| $2h$ | 44022 |
| $\Delta h$ | 022011 |
| $2\Delta^2 h$ | 002402 2 |
| $3\Delta^3 h$ | 0001103 3 |
| response | 4412430300... |

the respone to 2123 is the sequenca 4412430300...

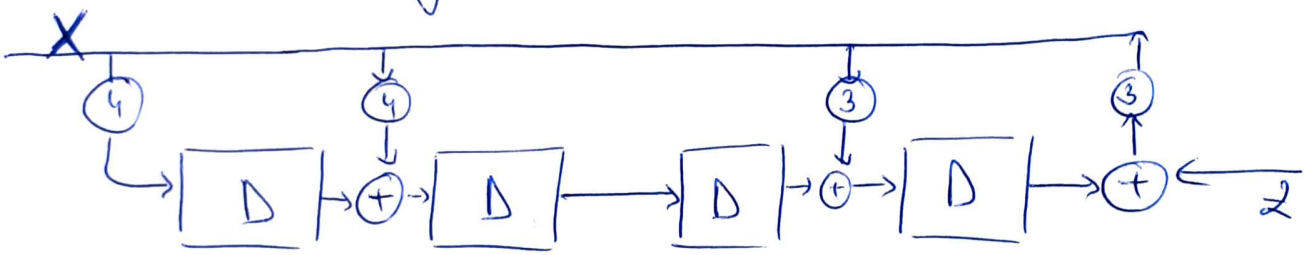
e) We have $T_3 = 2 + 2\Delta + \Delta^3 + \Delta^4$

$2^{-1}$ (in GF(5)) is $= 3$ (exists) $\quad 2\cdot 1 = 2,\ 2\cdot 2 = 4,\ 2\cdot 3 = 1$

$-2 = 3$

$-1 = 4$

With this knowledge in mind, the reverse automaton is:




$$T_3^{-1} = \frac{1}{T_3} = \frac{1}{2 + 2\Delta + \Delta^3 + \Delta^4}$$

b) -- because we have 4 possible states in GF(5), we use 3 bits

a) multiplier by 2 in GF(5):

| $X_2$ | $X_1$ | $X_0$ | $P_2$ | $P_1$ | $P_0$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | x | x | x |
| 1 | 1 | 0 | x | x | x |
| 1 | 1 | 1 | x | x | x |

b) library IEEE
use IEEE.std_logic_1164.all;
use IEEE.std_logic_arith.all;
use IEEE.std_logic_unsigned.all;

entity aut is
port(X: in std_logic_vector(2 downto 0); -- 3 Leitungen (5)
2: out std_logic_vector(2 downto 0));
end aut;

architecture arh of aut is
signal m1, m2, d3, d1, d0: std_logic_vector(2 downto 0);

begin
firstplus: adder port map(A => d3, B => X, out1 => d1);
multiplier2: mul2 port map(nr => X, result => m1);
secondplus: adder port map(A => m1, B => d1, out1 => d0);
multipl2: mul2 port map(nr => X, result => m2);
lastplus: adder port map(A => d0, B => m2, out1 => 2);
end arh;

~~-- multiplier~~

~~entity mul2~~
component adder is
port(A, B: in std_logic_vector(2 downto 0);
out1: out std_logic_vector(2 downto 0));
end component;

Component mul2 is
port(nr: in std_logic_vector(2 downto 0);
result: out std_logic_vector(2 downto 0));
end component;