DonSTU
Computer
Department

LEARNER'S GUIDE
"THEORY OF ANTINOISE CODING"

DonSTU
Computer
Department

Annotation

Laboratory №1

- **DESIGNING THE ENCODER AND THE DECODER FOR SIMPLE PARITY CHECK CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №2

- **DESIGNING THE ENCODER AND THE DECODER FOR GROUP CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №3

- **DESIGNING THE ENCODER AND THE DECODER FOR HAMMING CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №4

- **DESIGNING THE ENCODER AND THE DECODER FOR CYCLIC HAMMING CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №5

- **DESIGNING THE ENCODER AND THE DECODER FOR THE DOUBLE-ERROR-CORRECTING BCH CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №6

- **DESIGNING THE ENCODER AND THE DECODER FOR THE BURST-ERROR-CORRECTING FIRE CODE ON THE BASE OF ACTIVE-HDL**

Laboratory №7

- **DESIGNING THE ENCODER AND THE DECODER OF THE CYCLIC BURST-ERROR-CORRECTING CODE, CONSTRUCTED BY MEANS OF TECHNIQUE INTERLEAVING, ON THE BASE OF ACTIVE-HDL**

Laboratory №8

- ## DESIGNING THE ENCODER AND THE DECODER OF THE BURST-ERROR-CORRECTING CONVOLUTIONAL IWADARE CODE ON THE BASE OF ACTIVE-HDL

Literature

Appendix 1 – Example of the title page
Appendix 2 - The order of defence of a laboratory

TEL. (0622) 301-07-58
301-08-90
FAX. (062) 335-45-89
mailto:do@cs.dgtu.donetsk.ua

83000 Донецк
ул.Артема 58
корпус 4, ауд. 4.14
кафедра “Компьютерная инженерия”

Web design by Dyachenko Oleg

УДК 681.3

Методичні вказівки до виконання лабораторних робіт з курсу «Theory of antinoise coding» (для студентів спеціальності 7.091501)/ Скл.: О.М.Дяченко.- Донецьк: ДонНТУ, 2007. - 35с. - HTML-формат (на електронному носії №19, прот. №7 від 20.06.07).

The problems of development of the group code, the Hamming code, the cyclic Hamming code, the double-error-correcting BCH code, the burst-error-correcting Fire code, the cyclic burst-error-correcting code, constructed by means of technique interleaving, the burst-error-correcting convolutional Iwadare code and designing of the encoders and the decoders are considered on the basis of application of a CAD "ACTIVE-HDL". There is the order and examples of execution of laboratory operations.

Maker: O.N.Dyachenko

Reviewer: S.V.Teplinskiy

Laboratory № 1

DESIGNING THE ENCODER AND THE DECODER FOR THE SIMPLE PARITY-CHECK CODE ON THE BASE OF ACTIVE-HDL

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the simple parity-check code and execute their simulation.

Task variants:

k - amount of information symbols;
n - blocklength;
p - amount of parity-check symbols.

k=4, n=5

The order of performance of work

- 1. In Circuit Editor of Active-HDL draw the principal circuits of the encoder and the decoder for the simple parity-check code detecting odd errors (k=4, n=5).
- 2. Debug program model.
- 3. In Time Diagram Editor perform simulation of the circuit, imitating the encoder, the channel, the decoder. In the channel provide possibility of imitation of errors. Investigate detecting ability of the decoder.

Content of the report

1. Functional circuit of the encoder and the decoder for the simple parity-check code detecting odd errors (k=4, n=5).
2. Principal circuits of the encoder and the decoder for the simple parity-check code detecting odd errors (k=4, n=5) with possibility of imitation of errors (are demonstrated on the PC).
3. Time diagrams of simulation of the encoder and the decoder for the simple parity-check code detecting odd errors (k=4, n=5) in Time Diagrams Editor (are demonstrated on the PC).

Questions

1. Into what types are antinoise codes divided? What is the difference between them?
2. What is "the block length" and "weight" of the code combination?
3. How can the distance be determined between code combinations?
4. What is the connection between the correcting ability of the code and the code distance?
5. What is "the binary symmetrical channel"?
6. What are the properties of the parity-check code?
7. What are the detecting abilities of the parity-check code?
8. How many the parity-check symbols does the parity-check code contain?
9. How is encoding accomplished?
10. How is decoding accomplished?

Example

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the simple parity-check code and execute their simulation.

Initial data

Amount of information symbols k=4.
Blocklength n=5.
Amount of parity-check symbols p=n-k=1.

The simplified model of an information transmission

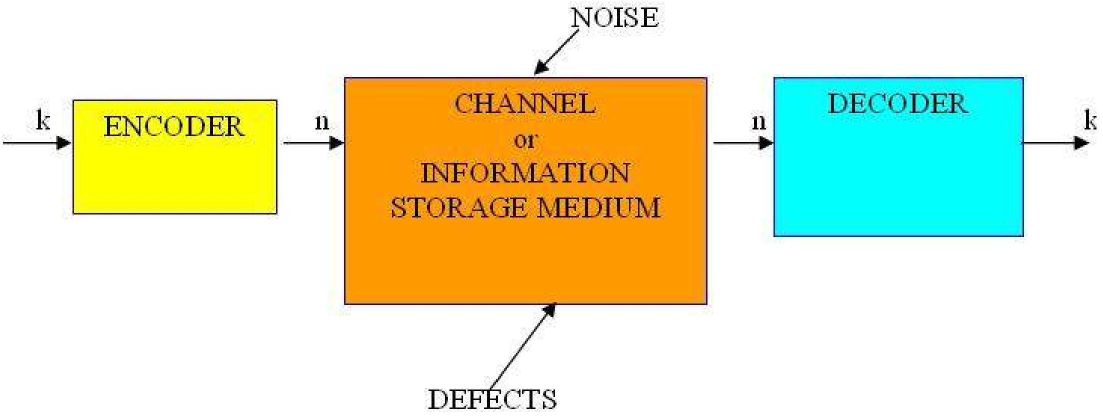


Figure1 - Simplified model of information transmission.

The simple parity-check code is the block code with detection of all odd errors.

The code contains only one parity-check symbol. The parity-check symbol must be equal to such bit in order that total amount of ones in a code combination was even number. The parity-check symbol of a code combination is equal to modulo 2 sum of all its symbols.

k - amount of information symbols;
n - code length;
p - amount of parity check symbols.

Functional circuit of the encoder and the decoder

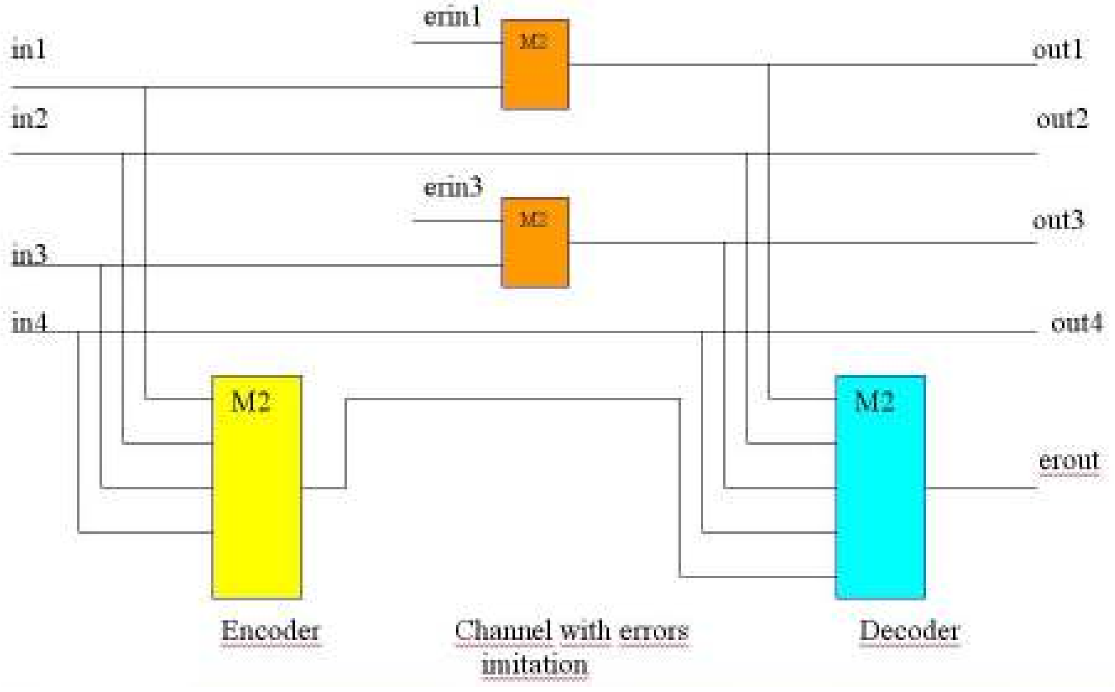


Figure 2 - Functional circuit of the encoder and the decoder for the simple parity-check code

Designation of signals

in1-in4 – inputs of the encoder (information symbols);
out1-out4 – outputs of the decoder (information symbols);
ier1, ier2 – inputs for imitation of the errors in the 1-st and 3-rd symbols;
erout – output of the error.

Principal circuits of the encoder and the decoder

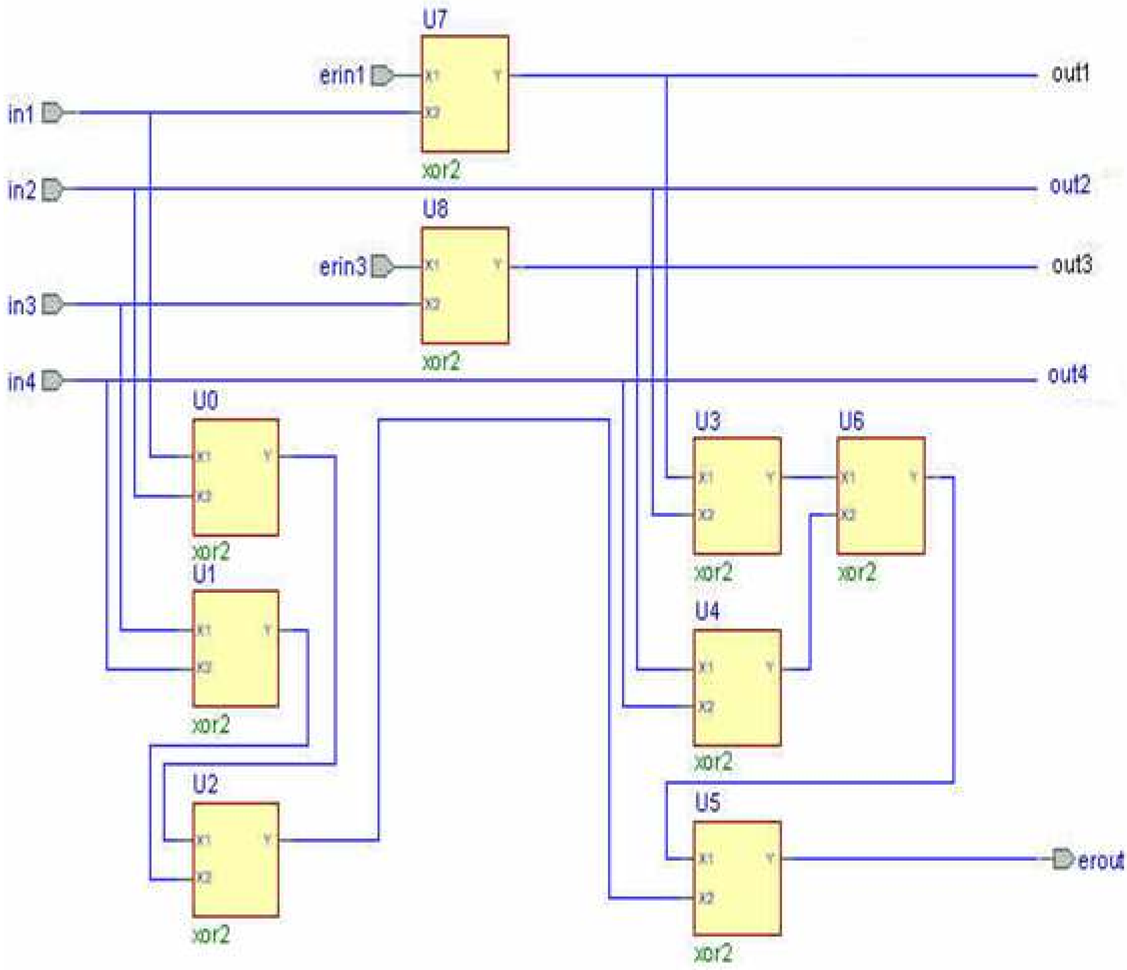


Figure 3 – Principal circuits of the encoder and the decoder for the simple parity-check code

Simulation of the encoder and the decoder for the simple parity-check code

Variant 1

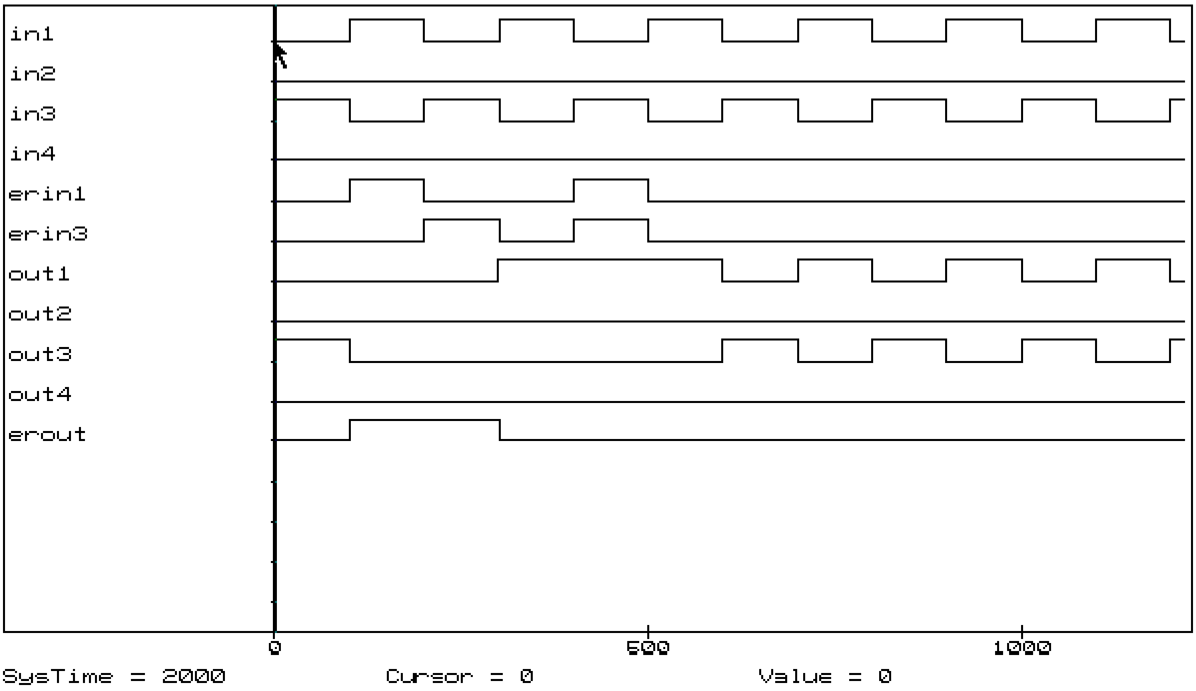
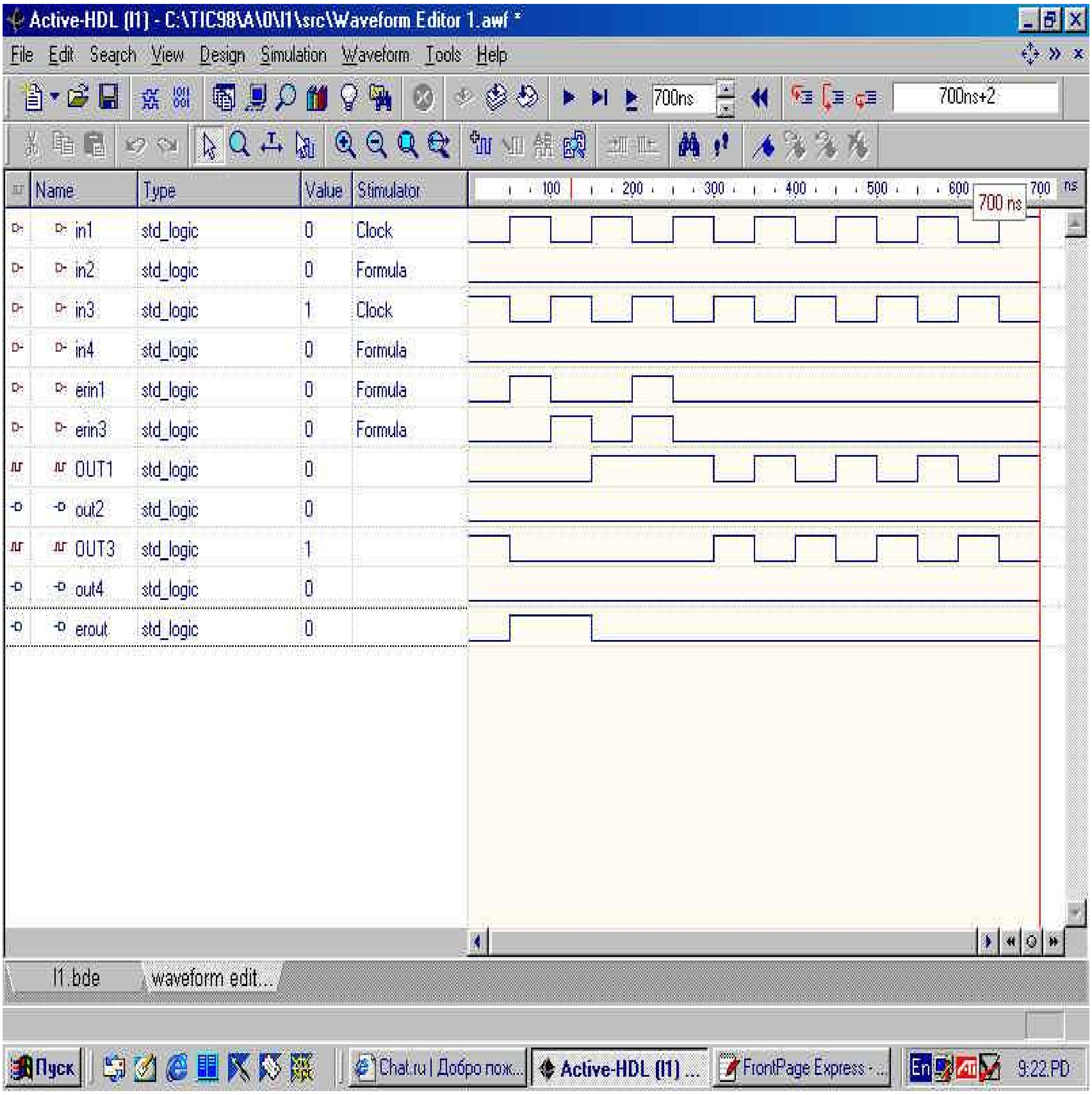


Figure 4 - Requirements to the time diagram of simulation of the encoder and the decoder for the simple parity-check code (variant 1)

Variant 2



Name	Type	Value	Stimulator
in1	std_logic	0	Clock
in2	std_logic	0	Formula
in3	std_logic	1	Clock
in4	std_logic	0	Formula
erin1	std_logic	0	Formula
erin3	std_logic	0	Formula
OUT1	std_logic	0	
out2	std_logic	0	
OUT3	std_logic	1	
out4	std_logic	0	
erout	std_logic	0	

Figure 5 - Requirements to the time diagram of simulation of the encoder and the decoder for the simple parity-check code (variant 2).

Variant 3

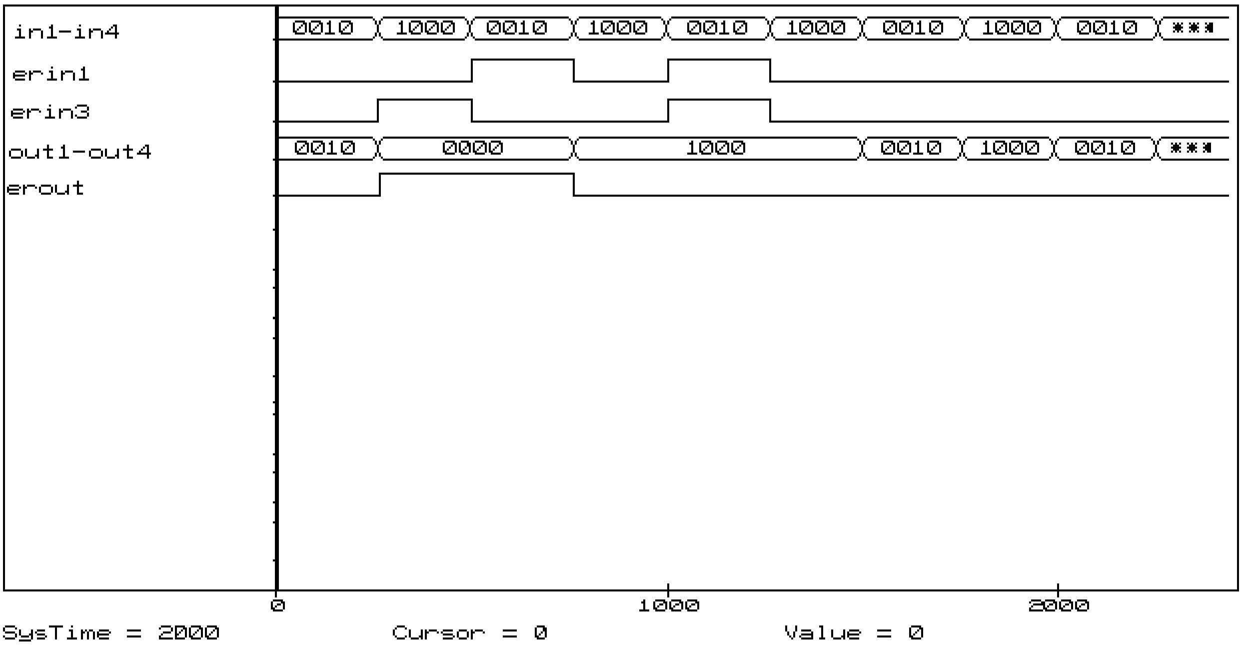


Figure 6 - Requirements to the time diagram of simulation of the encoder and the decoder for the simple parity-check code (variant 3)

Laboratory № 2

DESIGNING THE ENCODER AND THE DECODER FOR THE GROUP CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the single-error-correcting group code and execute their simulation.

Task variants:

- Amount of information symbols: k = [(N+5) / 2],
  where N is number given by the teacher.
- Variant A - group code with minimum parity-check symbols (if N is odd). Variant B - group code with minimal hardware of the encoder and the decoder (if N is even).

The order of performance of work

- 1. Find the minimal amount of the parity-check symbols. Construct the generator matrix of the group code for variant A or B.
- 2. Construct the parity-check matrix of the group code. Determine equalities for parity-check symbols and equalities for stages of the syndrome.
- 3. Construct the encoder and the decoder. Construct the decipherer for correction of single error.
- 4. Develop functional and principal circuits of the encoder and the decoder.
- 5. Make and to debug a program model.
- 6. Perform simulation of the circuit, imitating the encoder, the channel, the decoder. In the channel provide possibility of imitation of errors. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 2. The generator matrix of the group code.
- 3. The parity-check matrix for the group code.
- 4. Synthesis of the decoder.
- 5. Functional circuit of the encoder and the decoder.
- 6. Principal circuits of the encoder and the decoder for the group code with errors imitation possibility (are demonstrated on the PC).
- 7. Time diagrams of simulation of the encoder and the decoder for the group code in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. What is the difference between block and continuous codes?
- 2. What is “the weight of the code combination”?
- 3.What are “the weights of the codewords”?
- 4. How can the distance be determined between code combinations ?
- 5. What is "the Hamming distance of the code"?

- 6. What is the connection between the correcting ability of the code and the code distance?
- 7. How can the generator matrix be constructed for the group code ?
- 8. What are conditions of the parity-check submatrix construction?
- 9. What is the algorithm of determining parity-check symbols using the parity-check matrix?
- 10. How can the syndrome be determined using the parity- check matrix, if position of the error in the codeword is known?

Example (N=35)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the single-error-correcting group code and execute their simulation.

Initial data

N=35
Variant B - group code with minimal hardware of the encoder and the decoder
Amount of information symbols: k = [(N+5)/2] = 20

Construction of the group code

- Finding the minimal amount of the parity-check symbols.

$k = 20$
$p \geq [\log_2\{(k+1)+[\log_2(k+1)]\}] = 5.$

If to select p=5, we obtain 10 rows of the parity-check submatrix with weight 2 and 10 rows with weight 3.

If to select p=7, we obtain 20 rows with weight 2, i.e. 20*2=40 of ones in the parity-check submatrix.

- Construction of the generator matrix of group code with minimal hardware of the encoder and the decoder $P_{(27,20)}$

$$P_{(27,20)} = \begin{vmatrix}
100\ 000\ 000\ 0\ 000\ 000\ 000\ 0\ 000\ 001\ 1 \\
010\ 000\ 000\ 0\ 000\ 000\ 000\ 0\ 000\ 010\ 1 \\
001\ 000\ 000\ 0\ 000\ 000\ 000\ 0\ 000\ 100\ 1 \\
000\ 100\ 000\ 0\ 000\ 000\ 000\ 0\ 001\ 000\ 1 \\
000\ 010\ 000\ 0\ 000\ 000\ 000\ 0\ 010\ 000\ 1 \\
000\ 001\ 000\ 0\ 000\ 000\ 000\ 0\ 100\ 000\ 1 \\
000\ 000\ 100\ 0\ 000\ 000\ 000\ 0\ 000\ 011\ 0 \\
000\ 000\ 010\ 0\ 000\ 000\ 000\ 0\ 000\ 101\ 0 \\
000\ 000\ 001\ 0\ 000\ 000\ 000\ 0\ 001\ 001\ 0 \\
000\ 000\ 000\ 1\ 000\ 000\ 000\ 0\ 010\ 001\ 0 \\
000\ 000\ 000\ 0\ 100\ 000\ 000\ 0\ 100\ 001\ 0 \\
000\ 000\ 000\ 0\ 010\ 000\ 000\ 0\ 000\ 110\ 0 \\
000\ 000\ 000\ 0\ 001\ 000\ 000\ 0\ 001\ 010\ 0 \\
000\ 000\ 000\ 0\ 000\ 100\ 000\ 0\ 010\ 010\ 0 \\
000\ 000\ 000\ 0\ 000\ 010\ 000\ 0\ 100\ 010\ 0 \\
000\ 000\ 000\ 0\ 000\ 001\ 000\ 0\ 001\ 100\ 0 \\
000\ 000\ 000\ 0\ 000\ 000\ 100\ 0\ 010\ 100\ 0 \\
000\ 000\ 000\ 0\ 000\ 000\ 010\ 0\ 100\ 100\ 0 \\
000\ 000\ 000\ 0\ 000\ 000\ 001\ 0\ 011\ 000\ 0 \\
000\ 000\ 000\ 0\ 000\ 000\ 000\ 1\ 101\ 000\ 0
\end{vmatrix}$$

- Construction of the parity-check matrix

$$H = \begin{array}{|ccccccccccccccccccccccccccc|}
a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} & a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & a_{18} & a_{19} & a_{20} & b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}$$

- Determination of equalities for parity-check symbols:

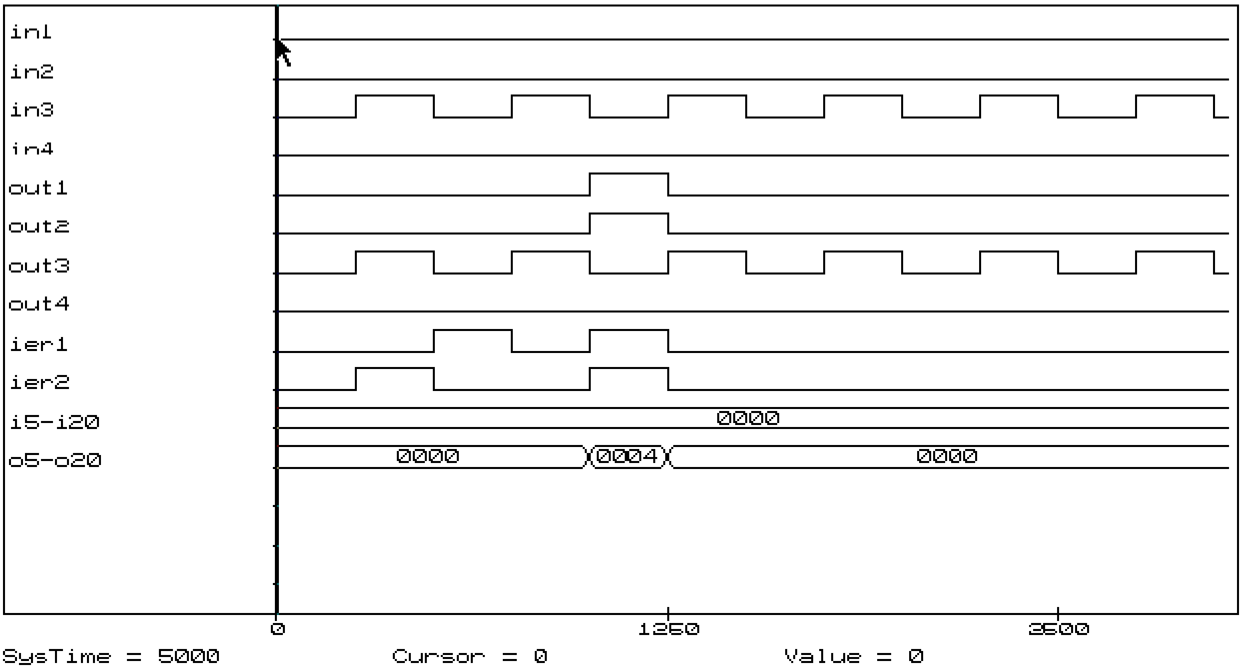
$b_1 = a_6 + a_{11} + a_{15} + a_{18} + a_{20}$
$b_2 = a_5 + a_{10} + a_{14} + a_{17} + a_{19}$
$b_3 = a_4 + a_9 + a_{13} + a_{16} + a_{19} + a_{20}$
$b_4 = a_3 + a_8 + a_{12} + a_{16} + a_{17} + a_{18}$
$b_5 = a_2 + a_7 + a_{12} + a_{13} + a_{14} + a_{15}$
$b_6 = a_1 + a_7 + a_8 + a_9 + a_{10} + a_{11}$
$b_7 = a_1 + a_3 + a_2 + a_4 + a_5 + a_6$

- Equalities for determining the syndrome:

$s_1 = b_1 + a_6 + a_{11} + a_{15} + a_{18} + a_{20}$
$s_2 = b_2 + a_5 + a_{10} + a_{14} + a_{17} + a_{19}$
$s_3 = b_3 + a_4 + a_9 + a_{13} + a_{16} + a_{19} + a_{20}$
$s_4 = b_4 + a_3 + a_8 + a_{12} + a_{16} + a_{17} + a_{18}$
$s_5 = b_5 + a_2 + a_7 + a_{12} + a_{13} + a_{14} + a_{15}$
$s_6 = b_6 + a_1 + a_7 + a_8 + a_9 + a_{10} + a_{11}$
$s_7 = b_7 + a_1 + a_3 + a_2 + a_4 + a_5 + a_6$

The principal circuit of the encoder and the decoder
In format PDF <lab.pdf>

Simulation of the encoder and the decoder



Designation of signals

in1, in2, in3, in4, i5-i20 – inputs of the encoder (information symbols);
out1, out2, out3, out4, o5-o20 – outputs of the decoder;
ier1, ier2 – inputs for imitation of an errors (in this case 4 situation are imitated: without errors, two variants of single corrected errors, double not corrected error).

Laboratory № 3

DESIGNING THE ENCODER AND THE DECODER FOR THE HAMMING CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the single-error-correcting Hamming codes or the single-error-correcting and double-error-detecting Hamming codes and execute their simulation.

Task variants:

- 1. Blocklength of the Hamming code

$$n = [(35 - N) / 2],$$

where N is number given by the teacher.

- 2. Variant A - the single-error-correcting and double-error-detecting Hamming code (if N is even).

  Variant B - the single-error-correcting Hamming code (if N is odd).

The order of performance of work

- 1. Find the minimal amount of the parity-check symbols. Construct the generator matrix for the Hamming code for variant A or B.
- 2. Construct the parity-check matrix for Hamming code. Determine equalities for parity-check symbols and equalities for stages of the syndrome.
- 3. Construct the encoder and the decoder. Use the standard decipher for correction of single error in the decoder.
- 4. Develop functional and principal circuits of the encoder and the decoder.
- 5. Make and to debug a program model.
- 6. Perform simulation of the circuit, imitating the encoder, the channel, the decoder. In the channel provide possibility of imitation of errors. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. The generator matrix for the Hamming code.
- 5. The parity-check matrix for the Hamming code.
- 6. Functional circuit of the encoder and the decoder.
- 7. Principal circuits of the encoder and the decoder for the Hamming code with errors imitation possibility (are demonstrated on the PC).
- 8. Time diagrams of simulation of the encoder and the decoder for the Hamming code in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. What is the principle of construction of Hamming codes ?

- 2. What is the performance of the Hamming codes?
- 3. What are the properties of the Hamming codes?
- 4. How is the Hamming code described and represented?
- 5. What is feature of the decoder for the Hamming code?
- 6. Are there modifications of Hamming codes which can provide better coding gains? What is the difference between them?
- 7. How must the parity-check matrix be constructed for the single-error-correcting Hamming code?
- 8. How must the parity-check matrix be constructed for the single-error-correcting and double-error-detecting Hamming code?
- 9. How can equalities be obtained for the parity-check symbols of the Hamming code?
- 10. How can equalities be obtained for stages of the syndrome of the Hamming code?

Example (N=-5)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the single-error-correcting Hamming codes or the single-error-correcting and double-error-detecting Hamming codes and execute their simulation.

Initial data

N=-5
Variant B - the single-error-correcting Hamming code (d_{min}=3)
Blocklength: n = [(35-N)/2] = 20

Construction of the single-error-correcting Hamming code

- Determination of the minimal amount of the parity-check symbols
  n = 20
  $p = [\log_2 (n+1)] = 5$: $k = n-p = 20-5 = 15$
- Construction of the generator matrix for the Hamming code

The parity-check symbols: b_1, b_2, b_4, b_8, b_{16}. Places of the parity-check symbols in the codeword: 1st, 2nd, 4th, 8th, 16th.
The parity-check matrix:

$$H = \begin{array}{c|cccccccccccccccccccc|}
 & b_1 & b_2 & a_3 & b_4 & a_5 & a_6 & a_7 & b_8 & a_9 & a_{10} & a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & b_{16} & a_{17} & a_{18} & a_{19} & a_{20} \\
\hline
 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0
\end{array}$$

Determination of equalities for parity-check symbols:
$b_1 = a_3 + a_5 + a_7 + a_9 + a_{11} + a_{13} + a_{15} + a_{17} + a_{19}$;
$b_2 = a_3 + a_6 + a_7 + a_{10} + a_{11} + a_{14} + a_{15} + a_{18} + a_{19}$;
$b_4 = a_5 + a_6 + a_7 + a_{12} + a_{13} + a_{14} + a_{15} + a_{20}$;
$b_8 = a_9 + a_{10} + a_{11} + a_{12} + a_{13} + a_{14} + a_{15}$;
$b_{16}= a_{17} + a_{18} + a_{19} + a_{20}$.

Determination of equalities for stages of the syndrome:

$s_0 = b_1 + a_3 + a_5 + a_7 + a_9 + a_{11} + a_{13} + a_{15} + a_{17} + a_{19}$;

$s_1 = b_2 + a_3 + a_6 + a_7 + a_{10} + a_{11} + a_{14} + a_{15} + a_{18} + a_{19}$;

$s_2 = b_4 + a_5 + a_6 + a_7 + a_{12} + a_{13} + a_{14} + a_{15} + a_{20}$;

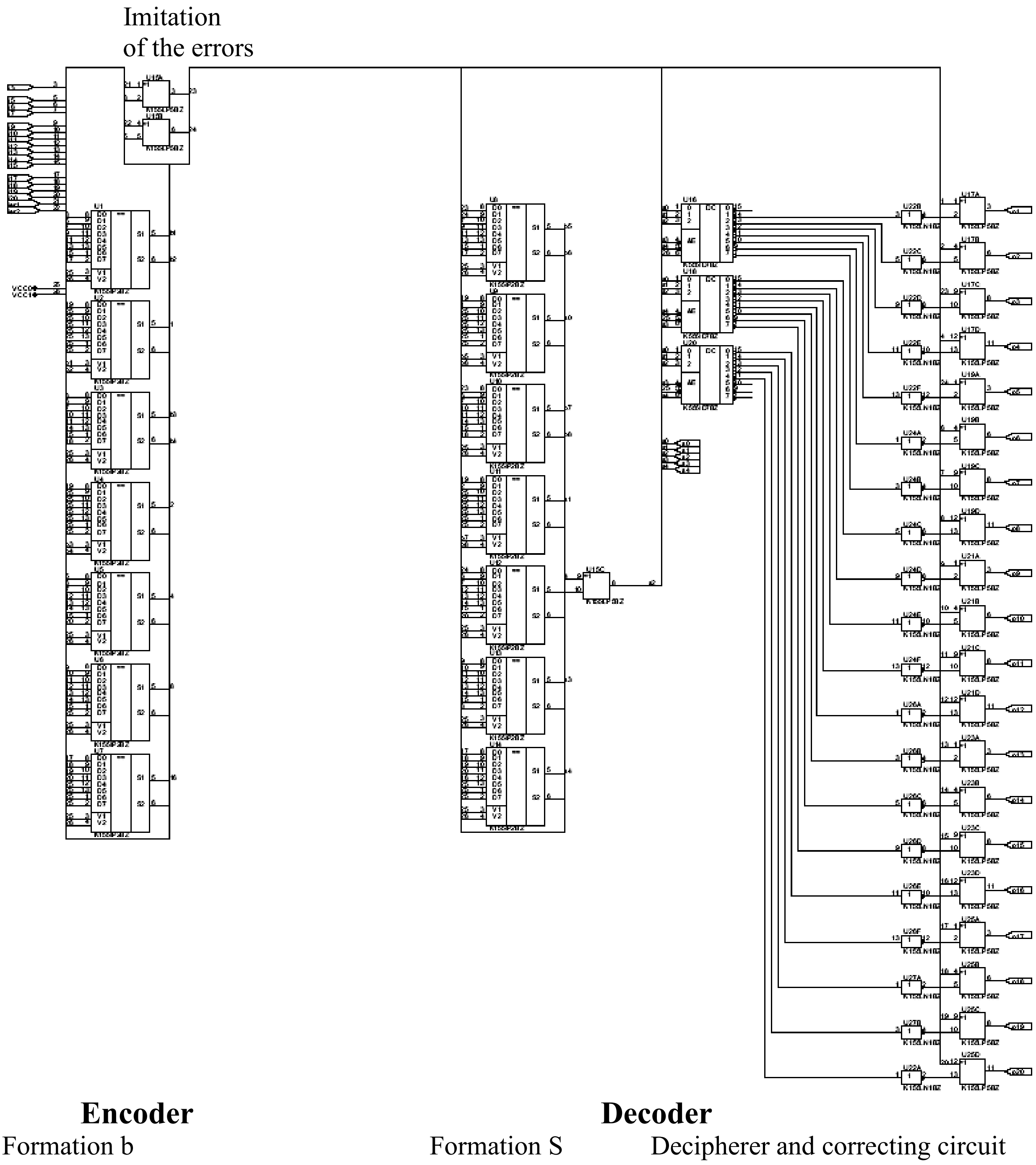
$s_3 = b_8 + a_9 + a_{10} + a_{11} + a_{12} + a_{13} + a_{14} + a_{15}$;

$s_4 = b_{16} + a_{17} + a_{18} + a_{19} + a_{20}$.

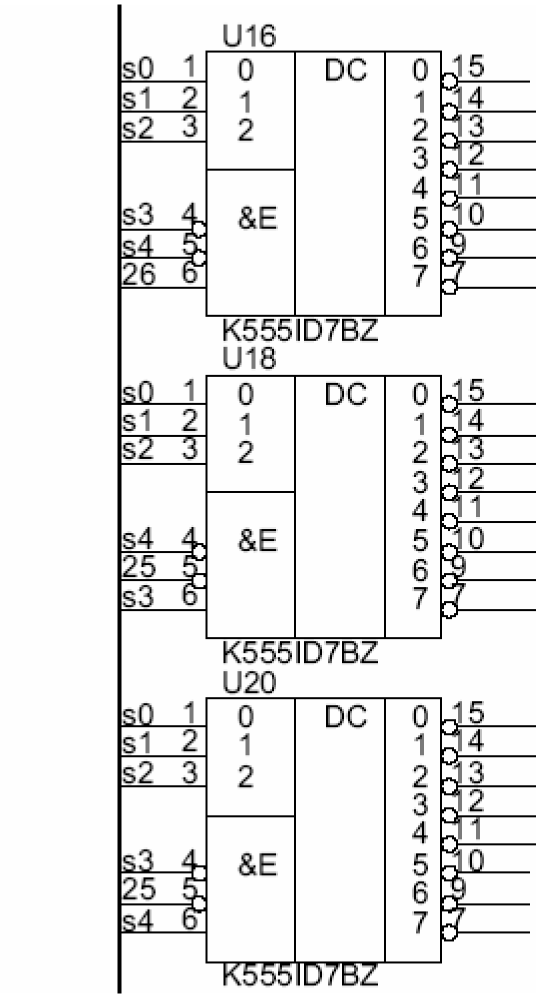
The principal circuit of the encoder and the decoder

1) In format PDF <lab.pdf>

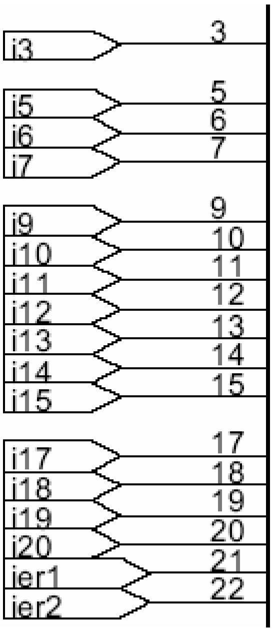
2) The encoder, imitation of errors and the decoder (overview).



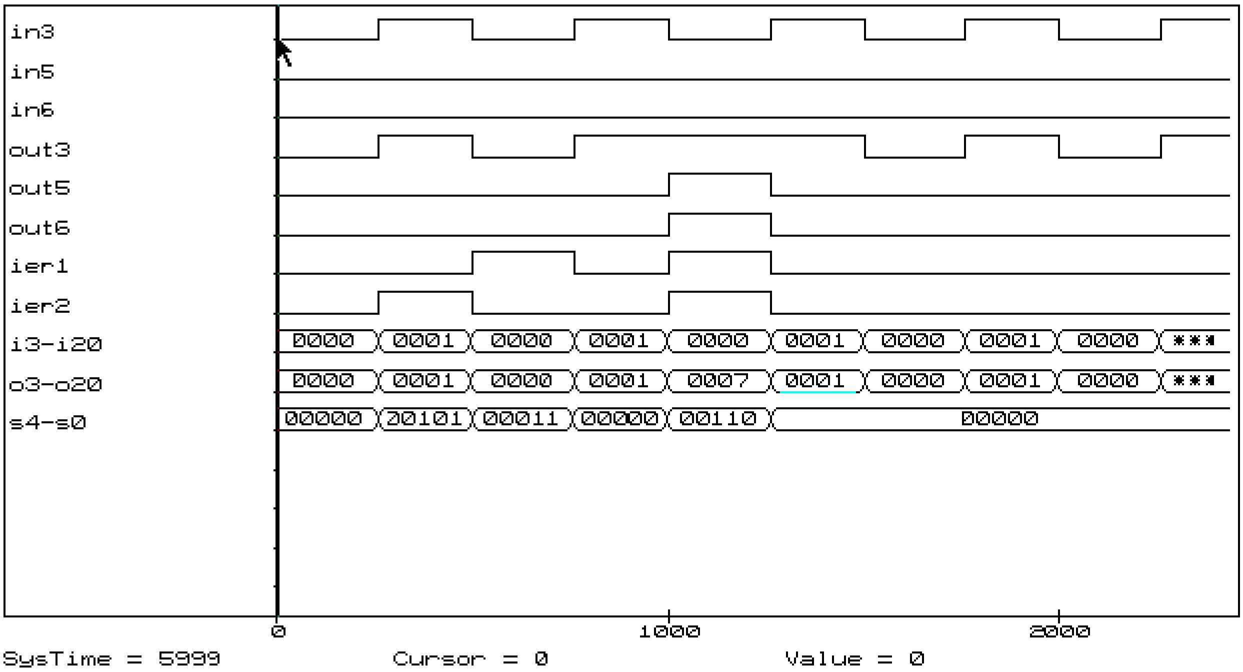
The decipherer of the decoder of the Hamming code (25 is logic 0, 26 is logic 1)



Inputs of the encoder



Simulation of the encoder and the decoder



Designation of signals

in3, in5, in6, i3-i20 – inputs of the encoder (information symbols);
out3, out5, out6, o3-o20 – outputs of the decoder;
ier1, ier2 – inputs for imitation of an errors (in this case 4 situation are imitated: without errors, two variants of single corrected errors, double not corrected error);
s4-s0 – stages of the syndrome.

Laboratory № 4

DESIGNING THE ENCODER AND THE DECODER FOR THE CYCLIC HAMMING CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the cyclic single-error-correcting Hamming codes and execute their simulation.

Task variants:

- 1. Amount of information symbols k = [(N + 1) / 4], where N is number given by the teacher.
- 2. $\text{N} \bmod 4 = \begin{cases} 0 - \text{variant A;} \\ 1 - \text{variant B;} \\ 2 - \text{variant C;} \\ 3 - \text{variant D.} \end{cases}$
- For variants A and B the generator polynomial K(X) is selected from the table; for variants C и D the generator polynomial K'(X), which is reciprocal (dual) to K(X) from the table.
- For variants A и C construct the systematic code; for variants B and D construct the nonsystematic code.

Table of the primitive polynomials

The degree of the polynomial	The polynomial in octal number system
2	7
3	13
4	23
5	45
6	103

Example.
Determination of the polynomial for 6th degree:
the polynomial in octal number system: 103;
the polynomial in binary form of representation: 001 000 011;
the polynomial form of representation: $X^6 + X + 1$.

The order of performance of work

- 1. Find the minimal amount of the parity-check symbols. According to the variant select the generator polynomial.
- 2. According to the variant construct the generator matrix of the cyclic code.
- 3. Construct the encoder and the decoder on the base of linear switching circuits.
- 4. Develop functional and principal circuits.
- 5. Make and debug a program model.
- 6. Perform simulation of the circuit, imitating the encoder, the channel, the decoder. In the channel provide possibility of imitation of errors. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. Determination of the minimal amount of the parity-check symbols.
- 5. Selection of the generator polynomial.
- 6. Construction of the generator matrix of the cyclic code.
- 7. Functional circuit of the encoder and the decoder.
- 8. Principal circuits of the encoder and the decoder for the cyclic code with errors imitation possibility (are demonstrated on the PC).
- 9. Time diagrams of simulation of the encoder and the decoder for cyclic code in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. Why are the cyclic codes called so?
- 2. Are there different types of the cyclic codes?
- 3. How is the generator polynomial selected for the cyclic code?
- 4. How is the parity-check matrix constructed for the cyclic single-error-correcting code?
- 5. What is “the Meggitt decoder”?
- 6. What is “the shortened cyclic code”?
- 7. How is the operation accomplished for division by the polynomial using linear switching circuit?
- 8. How is the multiplication accomplished for the polynomials using linear switching circuit?
- 9. How can the reciprocal (dual) polynomial be determined?
- 10. What is “the irreducible polynomial”?

Example (N=35)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the cyclic single-error-correcting Hamming codes and execute their simulation.

Initial data

Amount of information symbols k = [(N+1)/4] = 9
Variant C:
the generator polynomial K'(X), which is reciprocal (dual) to K(X) from the table;
the systematic Hamming code.

Construction of the cyclic single-error-correcting systematic Hamming code

- 1. Determination of the minimal amount of parity-check symbols:
  $p = [\log_2\{(k+1)+[\log_2(k+1)]\}] = 4$,
  (in this case square brackets mean a rounding off to the larger closest integer).
  Blocklength n = k+p=9+4 = 13.

- 2. Selection of the generator polynomial. In the table of the irreducible polynomials the primitive polynomial of fourth degree (p=4, therefore, deg K(X) =4) is represented as an octal entry of nonzero coefficients and is equal 23, i.e. 10011 in the binary number system, or in the polynomial form $K(X)=X^4+X+1$ ($23_8 = 010\ 011_2 = 0*X^5 + 1*X^4 + 0*X^3 + 0*X^2 + 1*X^1 + 1*X^0 = X^4+X+1$).

- For variant C generator polynomial K'(X) is reciprocal (dual) to K(X) from the table of the irreducible primitive polynomials.
- $K'(X) = X^{\deg K(X)} * K(X^{-1}) = X^4 *(X^{-4} + X^{-1} + 1) = X^4 *(X^{-4} + X^{-1} + 1) = X^4 + X^3 + 1.$
- 3. Construction of the generator matrix $P_{(n,k)} = I\ H_p$, where I is the identity matrix (the information submatrix), H_p is the parity-check submatrix.
- The information k by k submatrix (size kxk)

$$I = \begin{vmatrix} 100\ 000\ 000 \\ 010\ 000\ 000 \\ 001\ 000\ 000 \\ 000\ 100\ 000 \\ 000\ 010\ 000 \\ 000\ 001\ 000 \\ 000\ 000\ 100 \\ 000\ 000\ 010 \\ 000\ 000\ 001 \end{vmatrix}$$

- The parity-check submatrix H_p consists of the reminders after division of information row, supplemented by p zeros, by the generator polynomial.

- Reminder for the first row:

```
100 000 000 0000  | 11001
110 01            |-------
-------
 100 10
 110 01
 -------
  10 110
  11 001
  -------
   1 111 0
   1 100 1
  ---------
      11 100
      11 001
    --------
       1 010 0
       1 100 1
      --------
         110 10
         110 01
       --------
          00 11
```

Reminder for the second row:

$$\begin{array}{l|l}
100\,000\,000\,0000 & 11001 \\ \cline{2-2}
110\,01 & \\ \hline
\;100\,10 & \\
\;110\,01 & \\ \hline
\;\;10\,110 & \\
\;\;11\,001 & \\ \hline
\;\;\;1\,111\,0 & \\
\;\;\;1\,100\,1 & \\ \hline
\;\;\;\;\;11\,100 & \\
\;\;\;\;\;11\,001 & \\ \hline
\;\;\;\;\;\;1\,010\,0 & \\
\;\;\;\;\;\;1\,100\,1 & \\ \hline
\;\;\;\;\;\;\;\;110\,1 &
\end{array}$$

and so on.
The generator matrix:

$$P_{(13,\,9)} = \begin{array}{ccccccccccccc}
a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & b_1 & b_2 & b_3 & b_4 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1
\end{array}$$

- The decoder
- The full cyclic Hamming code for p=4 has such parameters n and k: $(2^p - 1, 2^p - 1 - p)$ or (15, 11). Thus the code (13, 9) is the shortened (n-i, k-i) cyclic code (15-2, 11-2) , and it is necessary to find the reminder after division $X^{(n-k+i)}$ by the generator polynomial. i=2 because the code (13, 9) is obtained by shortening code (15, 11): (15-2, 11-2). n-k=p=4, n-k+i=4+2=6.

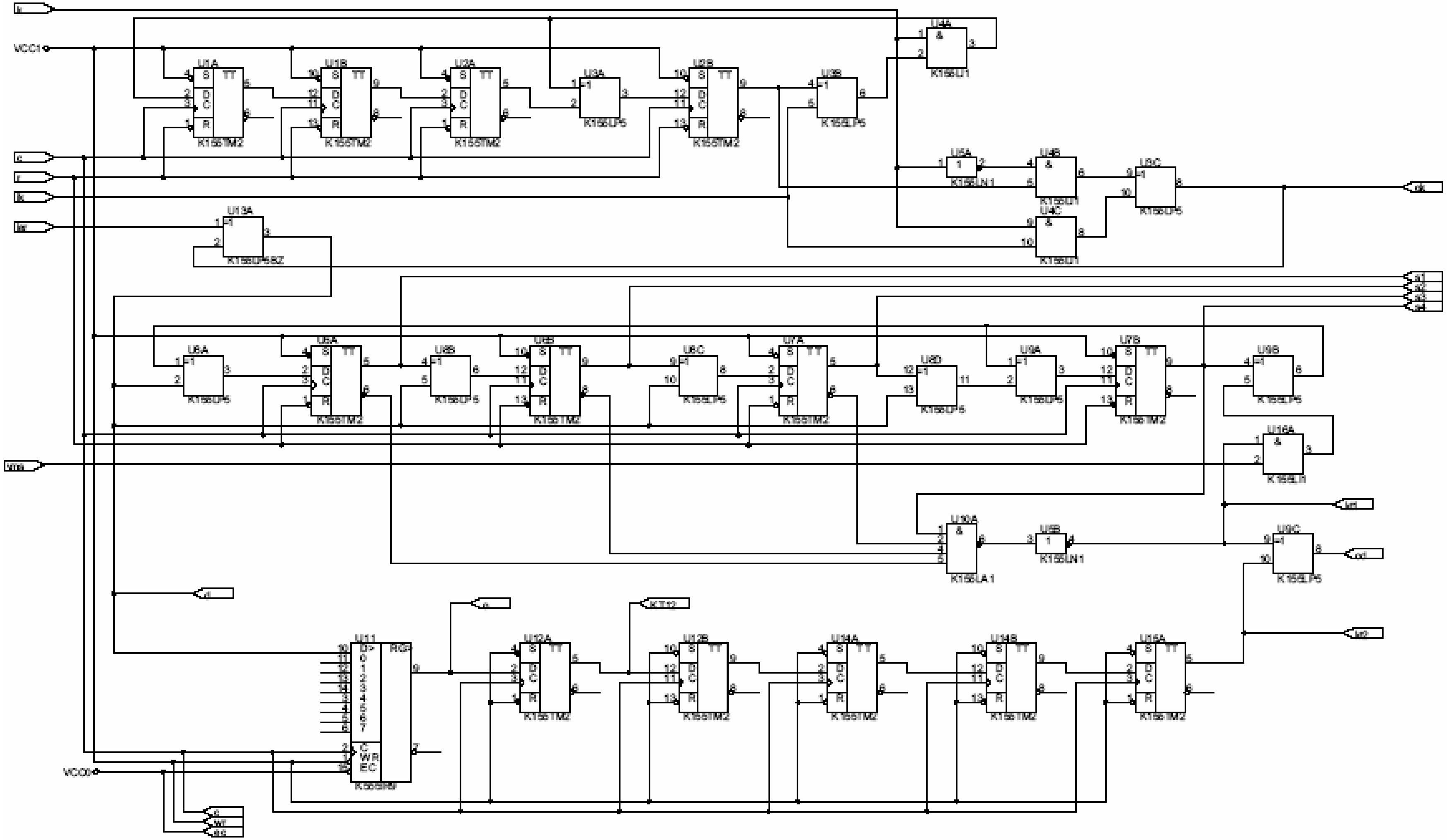
Determination of the remainder after division $X^{n-k+i} = X^6$ by the generator polynomial:

$$R(X^6) = X^3 + X^2 + X + 1$$

$$\begin{array}{l|l}
X^6 & X^4 + X^3 + 1 \\ \cline{2-2}
X^6 + X^5 + X^2 & X^2 + X + 1 \\ \hline
\quad X^5 + X^2 & \\
\quad X^5 + X^4 + X & \\ \hline
\qquad X^4 + X^2 + X & \\
\qquad X^4 + X^3 + 1 & \\ \hline
\qquad\quad X^3 + X^2 + X + 1 &
\end{array}$$

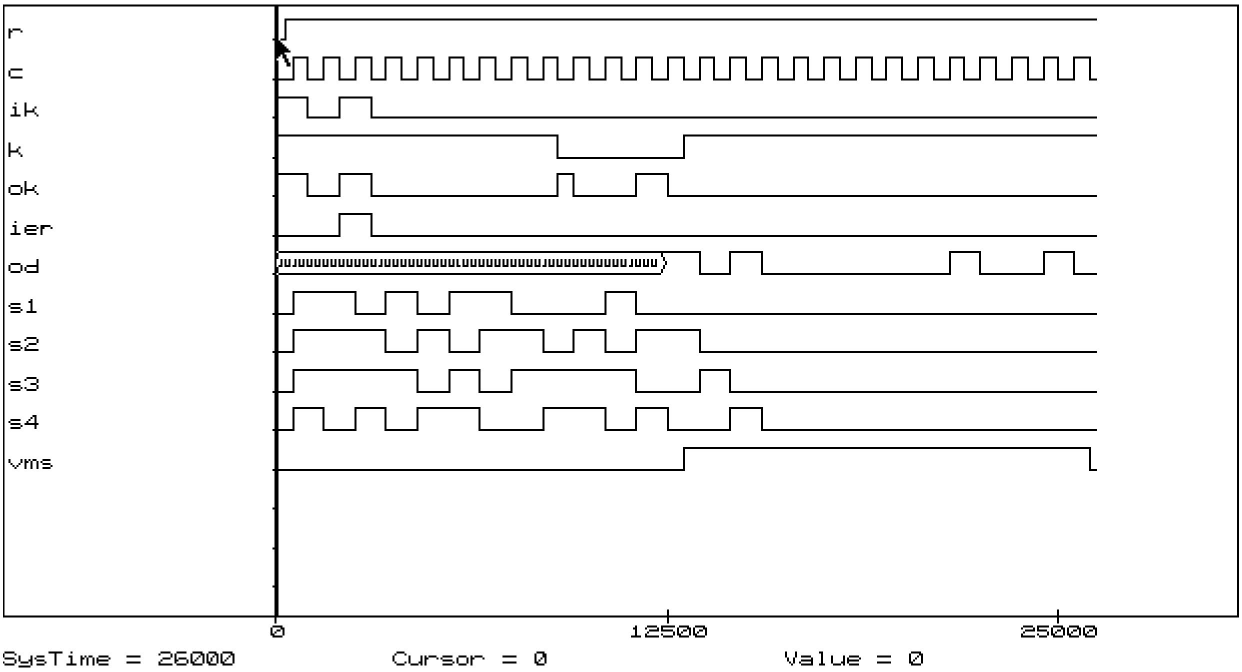
The principal circuit of the encoder and the decoder

1) In format PDF <lab.pdf>
2) The encoder, imitation of errors and the decoder

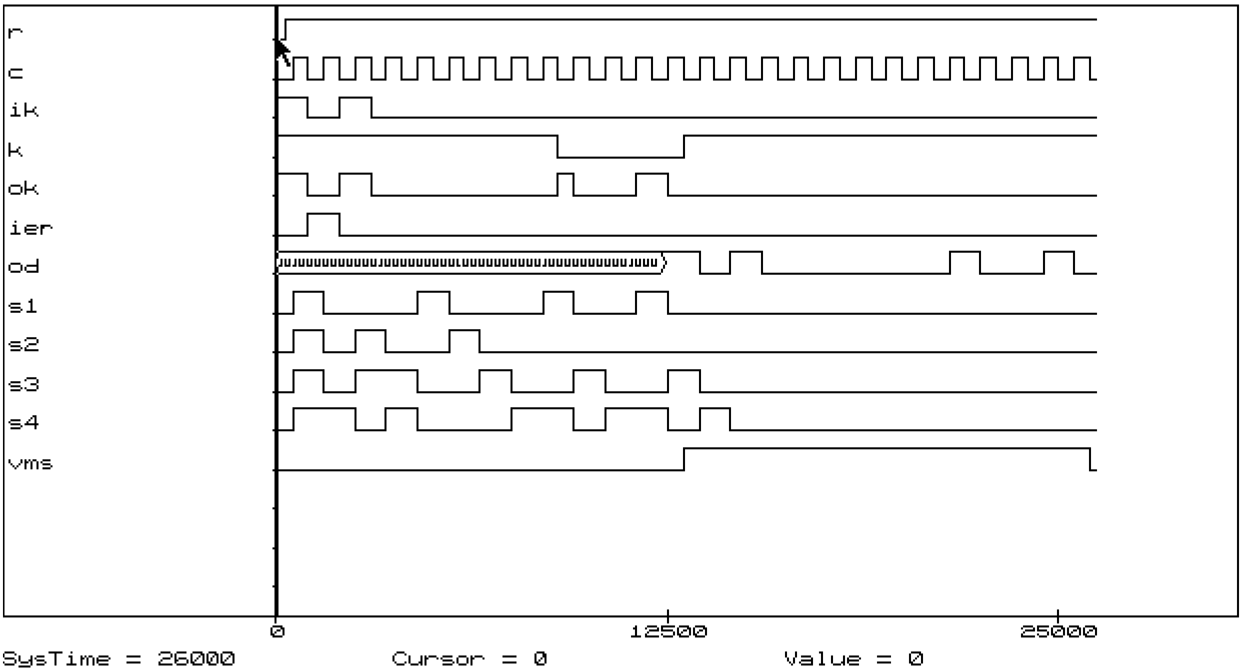


Simulation of the encoder and the decoder

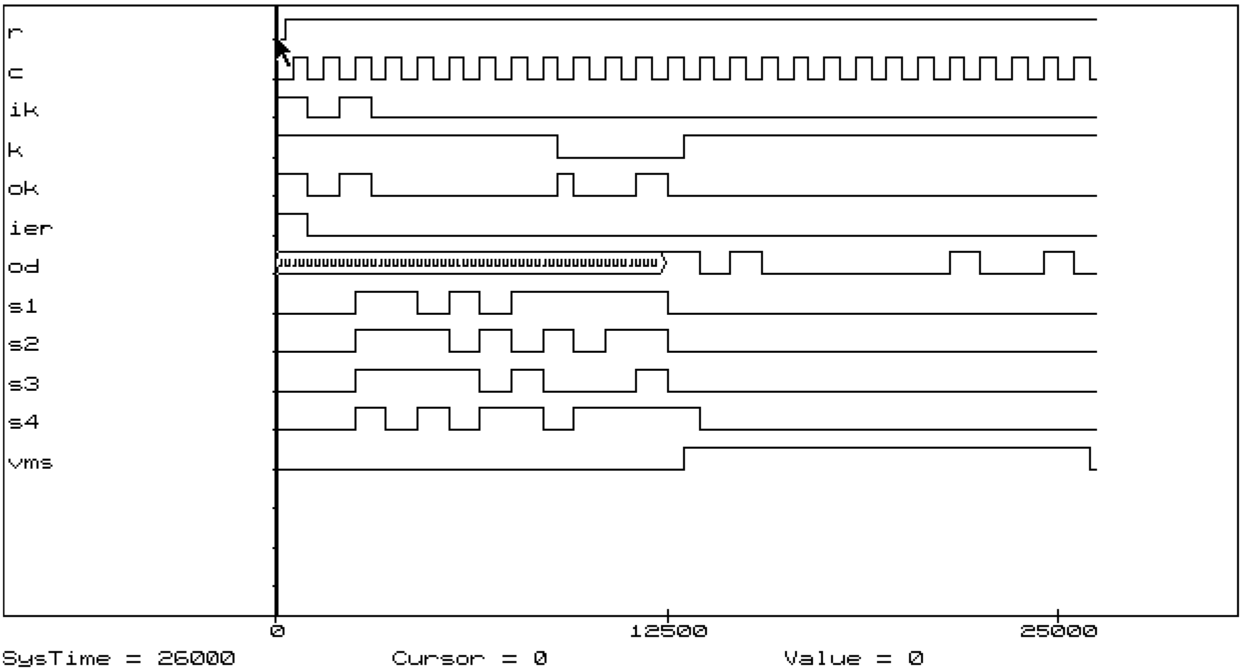
Imitation of an error in the 3-rd information symbol



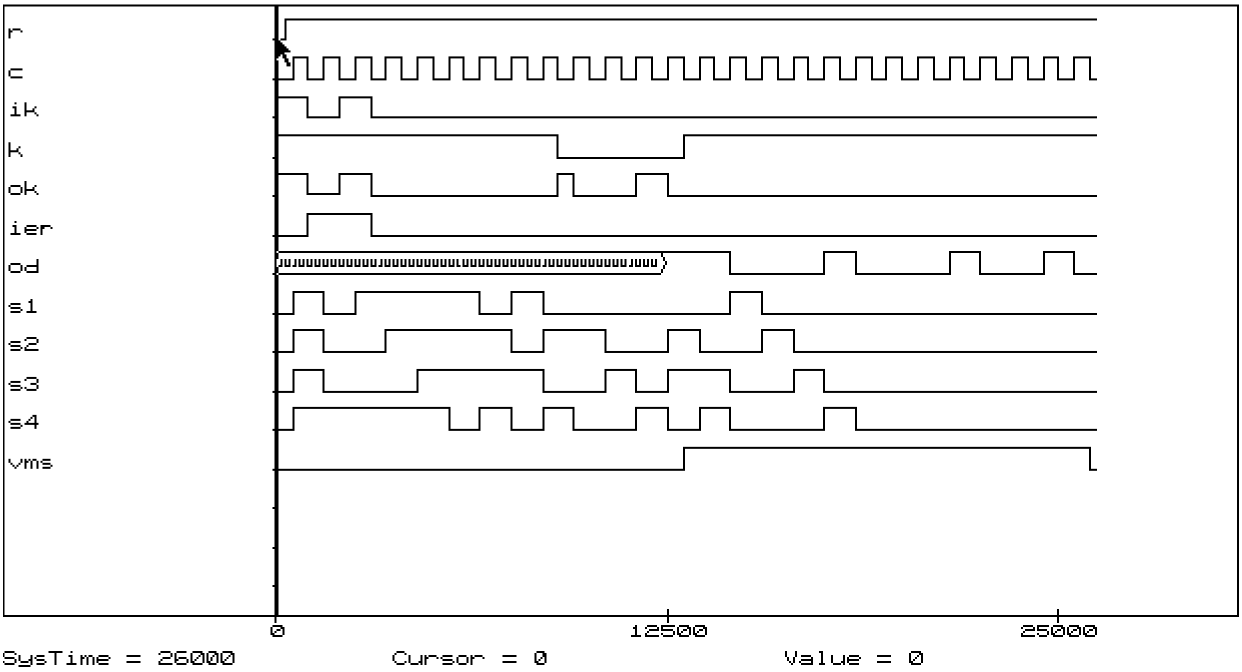
Imitation of an error in the 2-nd information symbol



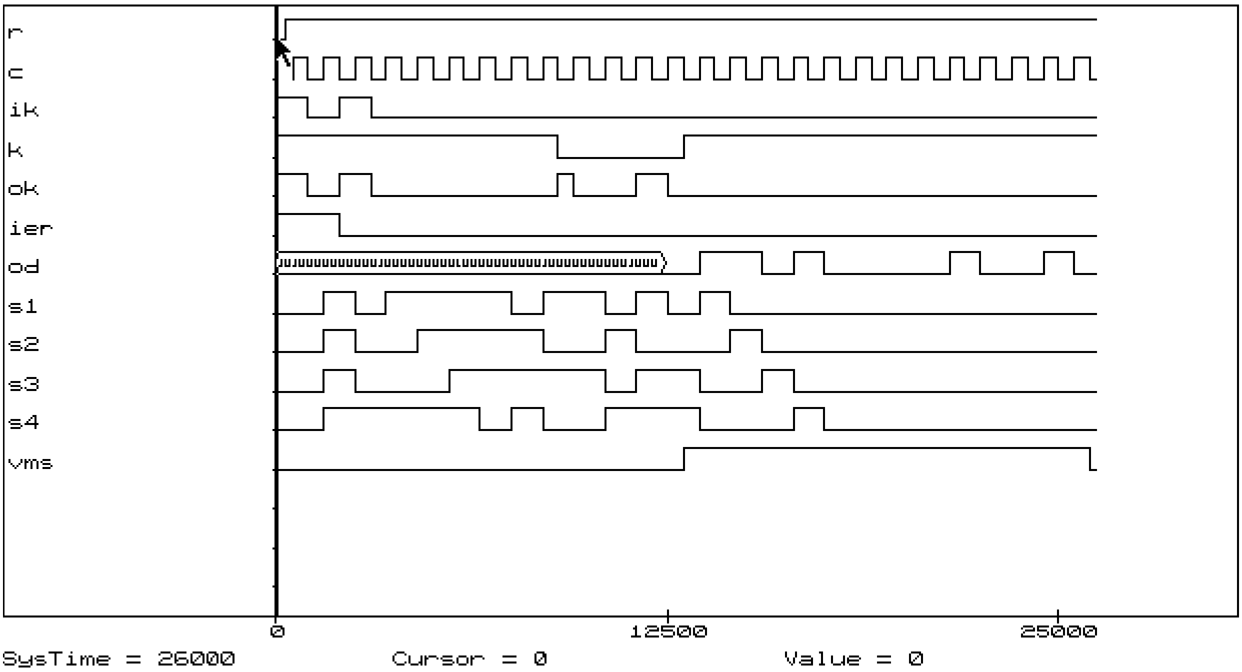
Imitation of an error in the 1-st information symbol



Imitation of an error in the 2-nd and 3-rd information symbols



Imitation of an error in the 1-st and 2-nd information symbols



Designation of signals

r – reset;
c – clock;
ik – input of the encoder;
k – control signal for key;
ok – output of the encoder;
ier – input for imitation of an error;
od – output of the decoder;
s1 – s4 – stages of the syndrome;
vms – enable signal of updating syndrome.

Laboratory № 5

DESIGNING THE ENCODER AND THE DECODER FOR THE DOUBLE-ERROR-CORRECTING BCH CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the double-error-correcting BCH code and execute their simulation.

Task variants:

$$1.\quad \mathrm{N}\,\mathrm{mod}\,4 = \begin{cases} 0 - \text{variant A}, \mathrm{M}_1(X), \mathrm{M}_2(X); \\ 1 - \text{variant B}, \mathrm{M}_1{}'(X), \mathrm{M}_2(X); \\ 2 - \text{variant C}, \mathrm{M}_1(X), \mathrm{M}_2{}'(X); \\ 3 - \text{variant D}, \mathrm{M}_1{}'(X), \mathrm{M}_2{}'(X). \end{cases}$$

- The minimal generator polynomials are $M_1(X)=X^4 + X + 1$, $M_2(X)=X^4 + X^3 + X^2 + X + 1$
- The polynomials $M_1'(X)$, $M_2'(X)$ are the minimal reciprocal polynomials for $M_1(X)$ и $M_2(X)$.

2. Error in the polynomial form E(X):

$$E(X)= X^{14} + X^{(N \bmod 15)}$$

for all N, when N mod 15 it is not equal 14. If N mod 15 = 14, then $E(X) = X^{14}$, where N is number given by the teacher.

The order of performance of work

- 1. According to the variant determine the generator polynomial of the double-error-correcting (15, 7) BCH code.
- 2. Construct the generator matrix of the systematic double-error-correcting (15, 7) BCH code.
- 3. Construct the encoder and the decoder on the base of linear switching circuits.
- 4. Develop functional and principal circuits of the encoder and the decoder.
- 5. Make and debug program model.
- 6. Perform simulation of the circuit, imitating the encoder, the channel, the decoder. In the channel provide possibility of imitation of errors E(X). Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. Determination of the generator polynomial.
- 5. The generator matrix of the systematic double-error-correcting (15, 7) BCH code.
- 6. Functional circuit of the encoder and the decoder.

- 7. Principal circuits of the encoder and the decoder for the cyclic BCH code with errors imitation possibility (are demonstrated on the PC).
- 8. Time diagrams of simulation of the encoder and the decoder for cyclic BCH code in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. What are differences of construction of the BCH codes from construction of the cyclic codes with d=3?
- 2. How can the generator polynomial be determined for the BCH code?
- 3. How can the generator matrix be constructed for the BCH code?
- 4. What is the polynomial of an error?
- 5. What are versions known of the Meggitt decoders for BCH codes?
- 6. How can the BCH codes be found and described for n=15 and correcting 3, 4, 5, 6, 7 errors?
- 7. How are the encoders constructed for the BCH codes?
- 8. How is the degree determined for the generator polynomial of the BCH code?

Example (N=35)

Task: on the basis of means Active-HDL develop models of the encoder and the decoder for the BCH codes with correction of double errors and execute their simulation.

Initial data

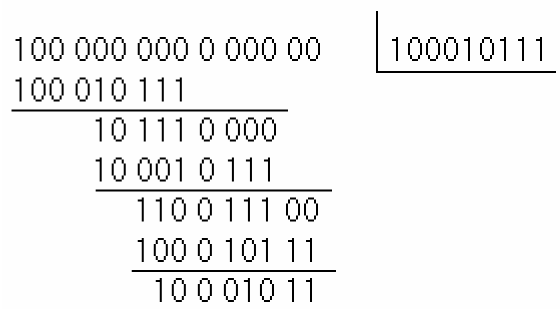
N=35
N mod 4 = 35 mod 4 = 3
Variant C.
The minimal generator polynomials are $M_1(X) = X^4 + X^3 + 1$; $M_2(X) = X^4 + X^3 + X^2 + X + 1$.
An error in the polynomial form $E(X) = X^{14} + X^5$.
The systematic code.

Construction of the BCH code

- Determining of a generator polynomial (15, 7) – BCH-code. $M(X) = \mathrm{LCM}(M_1(X), M_2(X)) =$ $=(X^4 + X^3 + 1)(X^4 + X^3 + X^2 + X + 1) = X^8 + X^7 + X^6 + X^5 + X^4 + X^7 + X^6 + X^5 + X^4 + X^3 + X^4 + X^3 + X^2 + X + 1 = X^8 + X^4 + X^2 + X + 1$.
- The generator matrix of the (15, 7) BCH code.

$$P = \begin{vmatrix} 100\ 000\ 0\ 100\ 010\ 11 \\ 010\ 000\ 0\ 110\ 011\ 10 \\ 001\ 000\ 0\ 011\ 001\ 11 \\ 000\ 100\ 0\ 101\ 110\ 00 \\ 000\ 010\ 0\ 010\ 111\ 00 \\ 000\ 001\ 0\ 001\ 011\ 10 \\ 000\ 000\ 1\ 000\ 101\ 11 \end{vmatrix}$$

- The parity-check submatrix: the code is systematic one; therefore it is necessary to find remainders of division of an information row added p in zero, by the generator polynomial.
- Remainder for the first row:



```
100 000 000 0 000 00    |100010111
100 010 111
        10 111 0 000
        10 001 0 111
            110 0 111 00
            100 0 101 11
             10 0 010 11
```

- Remainder for the second row - 110 011 10
- Remainder for the third row - 011 001 11
- The parity-check matrix for the (15, 7) BCH code.

$$H = \left|\begin{array}{ccccccccccccccc}
a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 & b_8 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right|$$

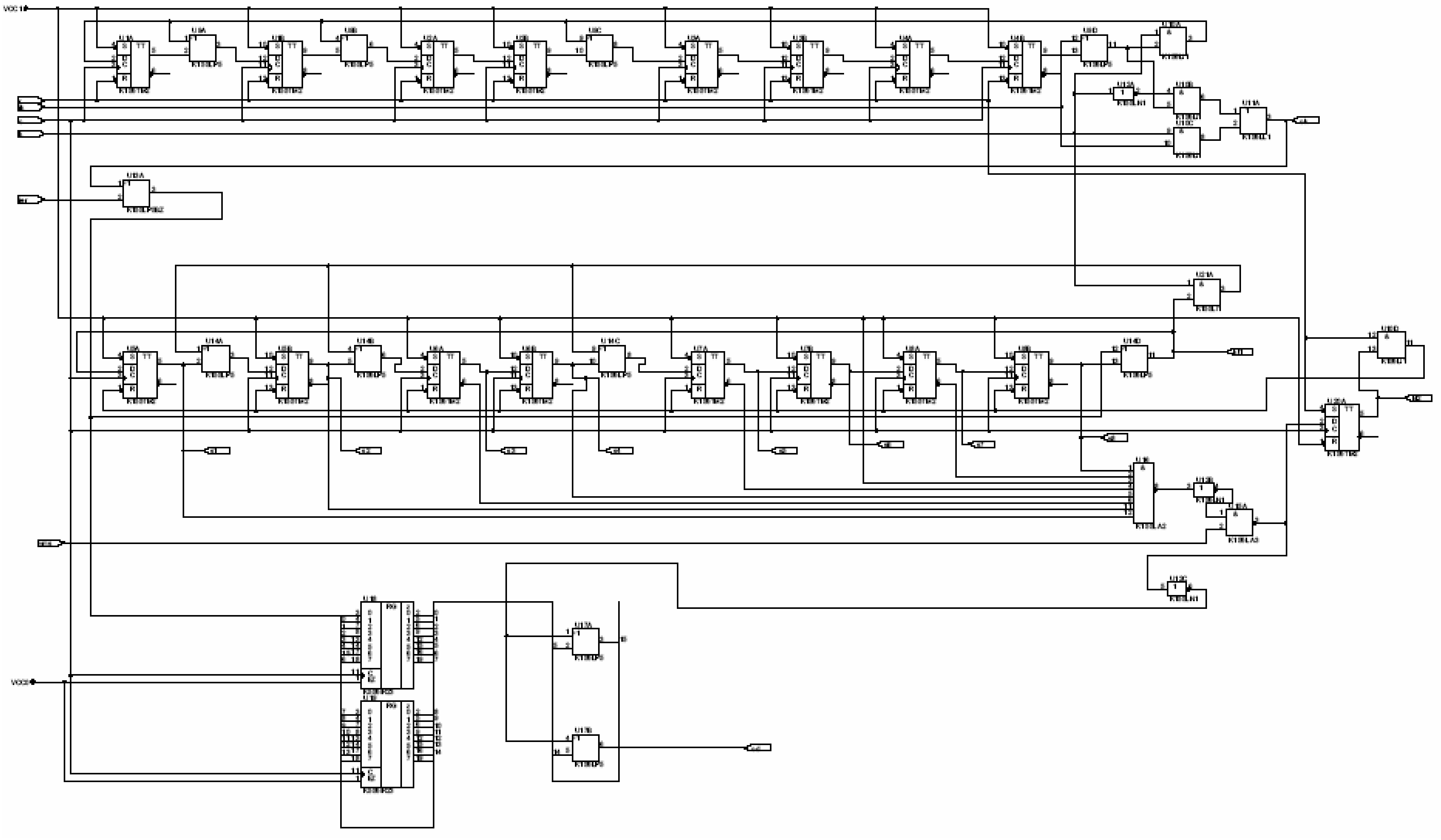
- Syndrome of error in the polynomial form $E(X) = X^{14} + X^5$

$S(X^{14} + X^5) = S(X^{14}) + S(X^5) = (101\ 010\ 11)$

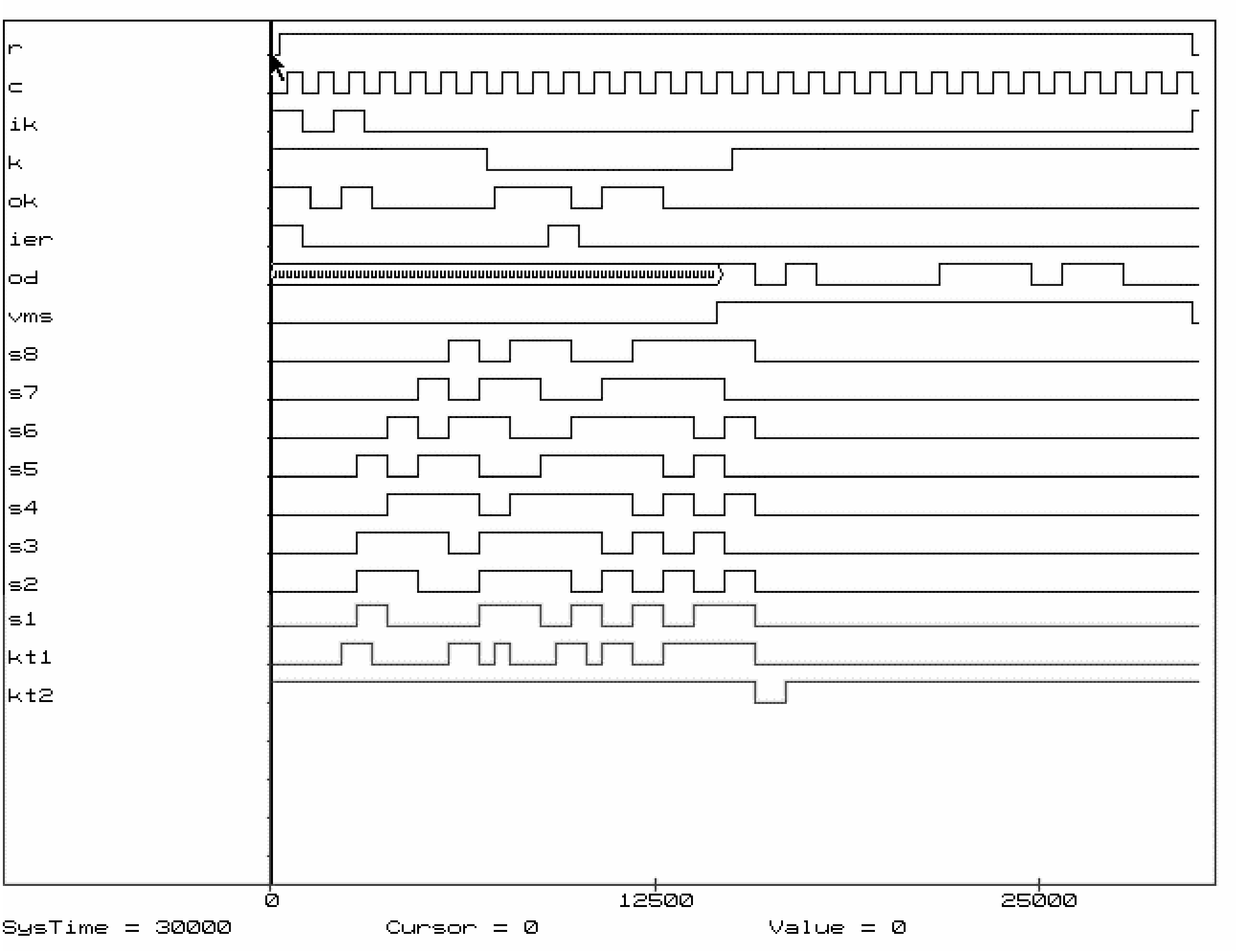
$$\begin{array}{cccccccccc}
 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
+ & & & & & & & & \\
 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\hline
= & & & & & & & & \\
 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1
\end{array}$$

The principal circuit of the encoder and the decoder

1) <u>In format PDF <lab.pdf></u>
2) The encoder, imitation of errors and the decoder



Simulation of the encoder and the decoder



Designation of signals

r – reset;
c – clock;
ik – input of the encoder;
k – control signal for key;
ok – output of the encoder;
ier – input for imitation of an error;
od – output of the decoder;
vms – enable signal of updating syndrome;
s1 - s8 – stages of the syndrome;
kt1 - control point 1;
kt2 - control point 2 (reset of the syndrome generator after correction of errors).

Laboratory № 6

DESIGNING THE ENCODER AND THE DECODER FOR THE BURST-ERROR-CORRECTING FIRE CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for burst-error-correcting Fire code and execute their simulation.

Task variants:

- 1. The generator polynomial:

  $g_1(X)=(X^7 + 1)(X^4 + X^3 + 1)$, if N is even,

  $g_2(X)=(X^7 + 1)(X^4 + X + 1)$, if N is odd.

- 2. Construct the shortened single-burst-error-correcting Fire (105 - i, 94 - i)-code, i=83-[N/2], for burst length t=4, (in this case square brackets mean a rounding off to the larger closest integer), where N is number given by the teacher.

The order of performance of work

- 1. According to the variant determine the generator polynomial for the shortened single-burst-error-correcting Fire code.
- 2. Construct the encoder and the decoder on the base of linear switching circuits.
- 3. Develop functional and principal circuits of the encoder and the decoder.
- 4. Make and debug program model.
- 5. Perform simulation of the circuit, imitating the encoder, the binary channel, the decoder. In the binary channel provide possibility of imitation of burst error in a code combination. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. Determination of the generator polynomial, parameters of the shortened Fire (n,k)-code.
- 5. Determination of the remainder after division polynomial $X^{(n-k+i)}$ by the generator polynomial (by means of simulation of the encoder on the PC).
- 6. Functional circuit of the encoder and the decoder.
- 7. Principal circuits of the encoder and the decoder for the Fire code with burst error imitation possibility (are demonstrated on the PC).
- 8. Time diagrams of simulation of the encoder and the decoder for the Fire code in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. What is “a burst error”?
- 2. What is “a cyclic burst error”?
- 3. How many parity-check symbols must the block linear code contain for correcting all burst error of length t?
- 4. What is the Fire code?
- 5. How is the generator polynomial for construction of Fire code determined?
- 6. How many parity-check symbols must the Fire code contain for correcting all burst errors of length t?
- 7. How is the encoder to be constructed for the Fire codes?

- 8. Draw the scheme of detection and correction of burst errors of length t using the Fire code.
- 9. How are the shortened Fire codes formed?
- 10. What is feature of construction of the decoding device for shortened Fire code?

Example (N=35)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the burst-error-correcting Fire code and execute their simulation.

Initial data

N=35
$g_1(X)=(X^7 + 1)(X^4 + X^3 + 1)$,
i = 83 -[N/2] = 83-18 = 65

Determination of the generator polynomial, parameters of shortened Fire (n,k)-code

The shortened Fire (105-65,94-65) code, that is (40,29) code.

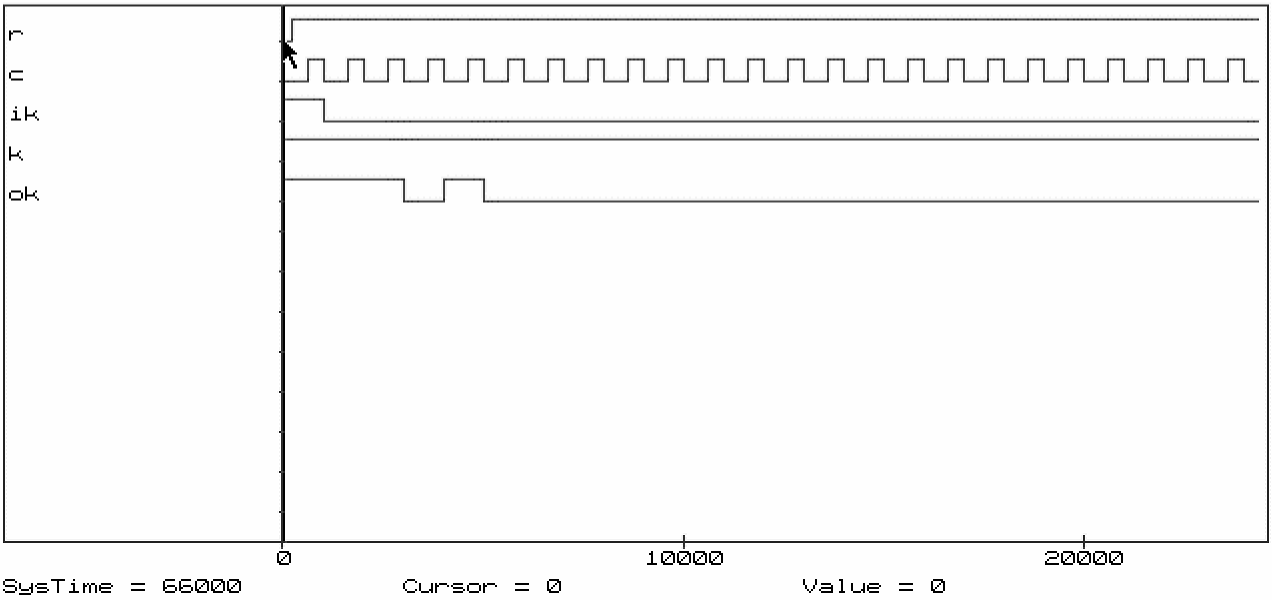
The generator polynomial is $g_1(X)=X^{11} + X^{10} + X^7 + X^4 + X^3 + 1$,

length of a corrected burst error t=4.

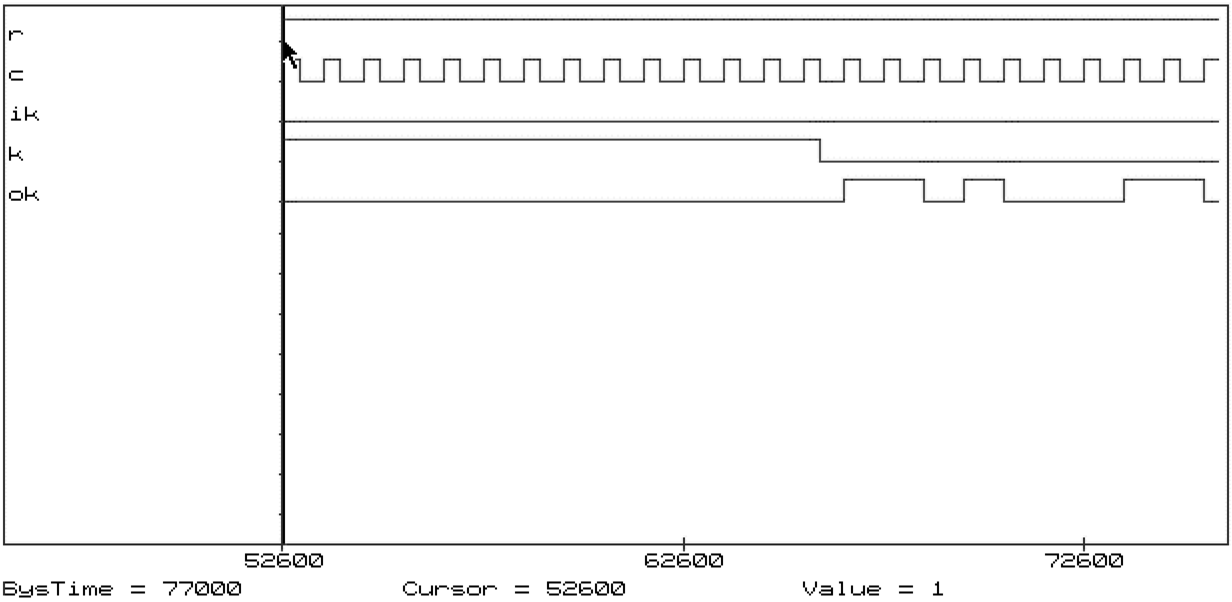
Determination of the remainder after division polynomial $X^{(n-k+i)}$ by generator polynomial

As the (40, 29)-code is the shortened one, for the decoder it is necessary to find remainder after division the polynomial X^{n-k+i} by the generator polynomial. In this case n-k=105-94=11, i=65. The remainder after division X^{11+65} by the generator polynomial is determined by modeling on the PC of the encoder, setting on its input 1 one and 65 zeros. We have obtained 01101000110, or in the polynomial form

$R(X^{11+65}) = X^9 + X^8 + X^6 + X^2 + X$.



SysTime = 66000 Cursor = 0 Value = 0



BysTime = 77000 Cursor = 52600 Value = 1

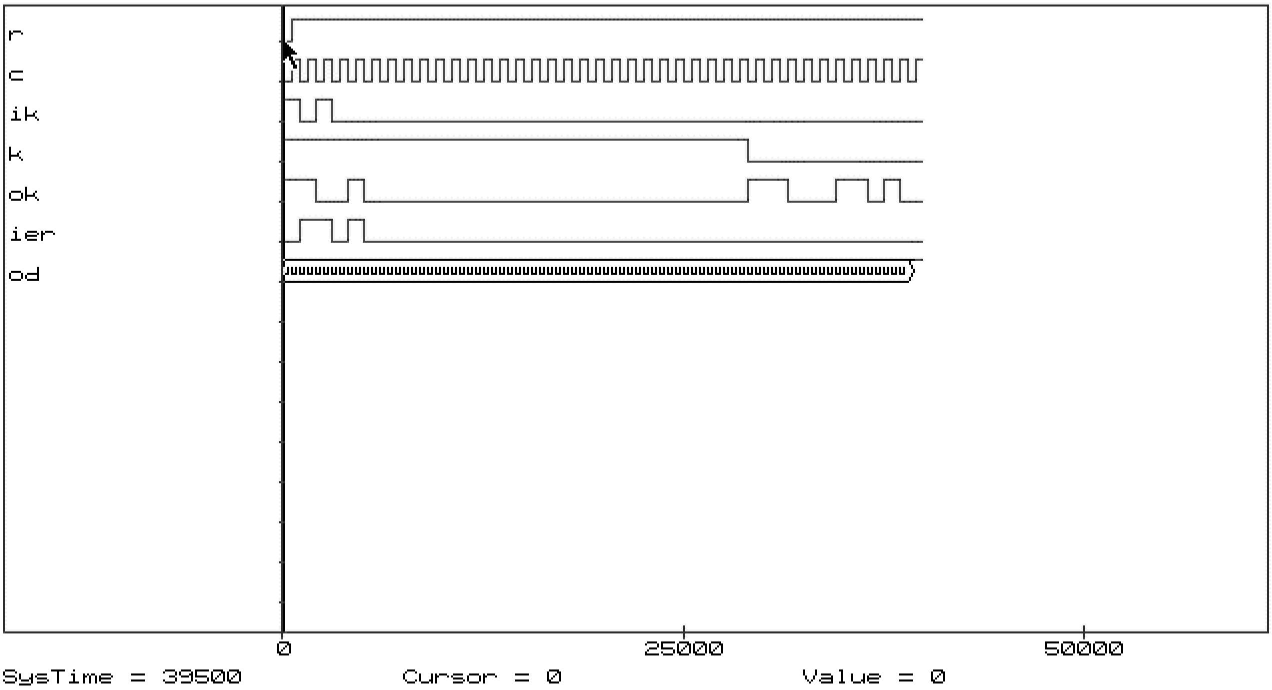
To check up the received result (and at the same time the encoder) it is possible by means of programs of division of polynomials POLY or DIV_POL.

The principal circuit of the encoder and the decoder

In format PDF <lab.pdf>

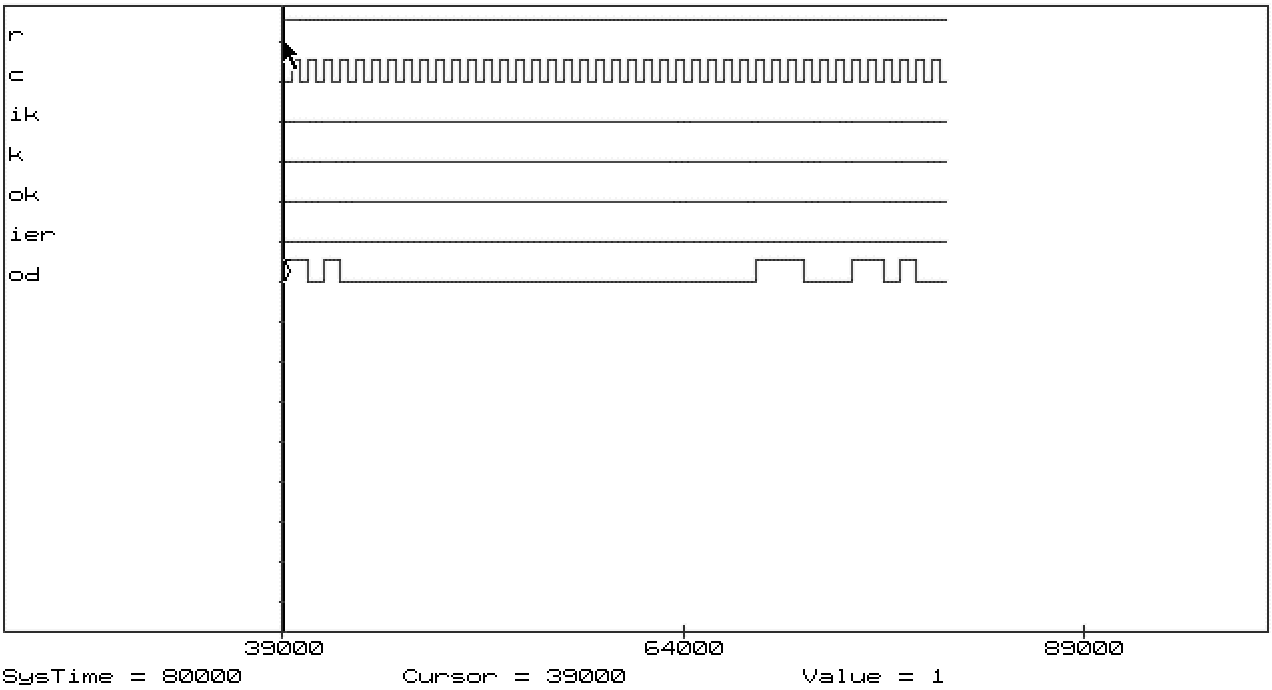
Simulation of the encoder and the decoder

1-st page



SysTime = 39500 Cursor = 0 Value = 0

2-nd page

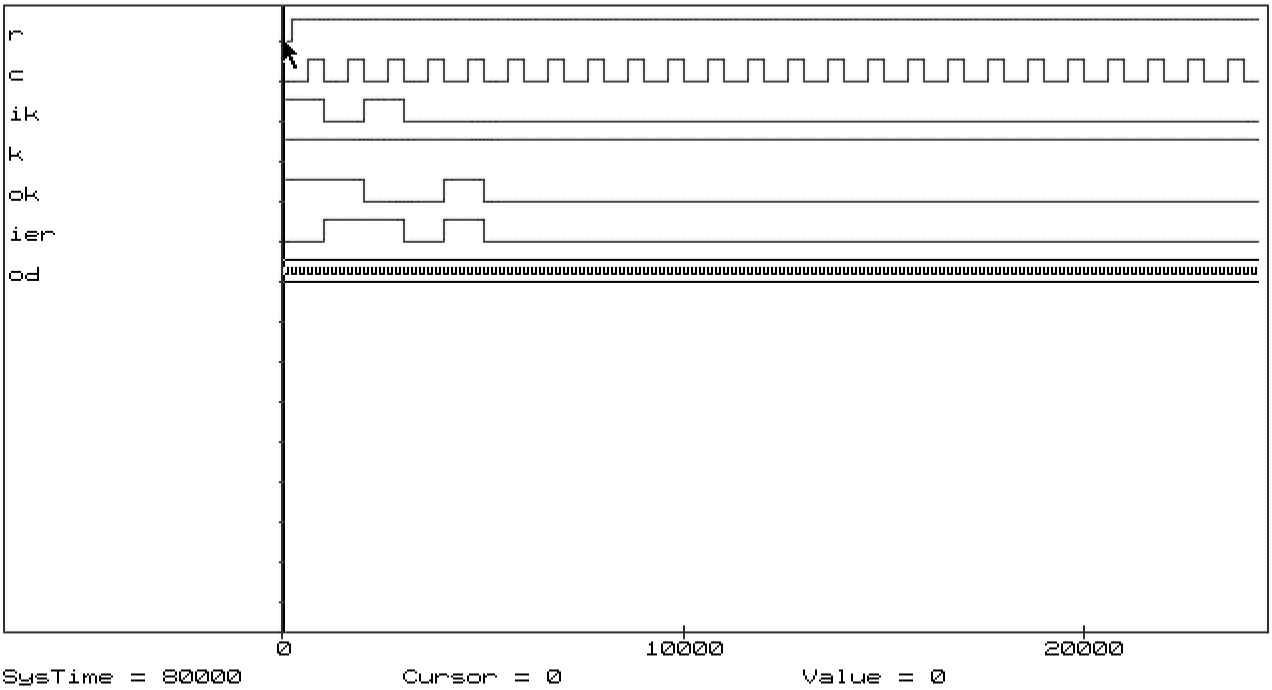


Designation of signals

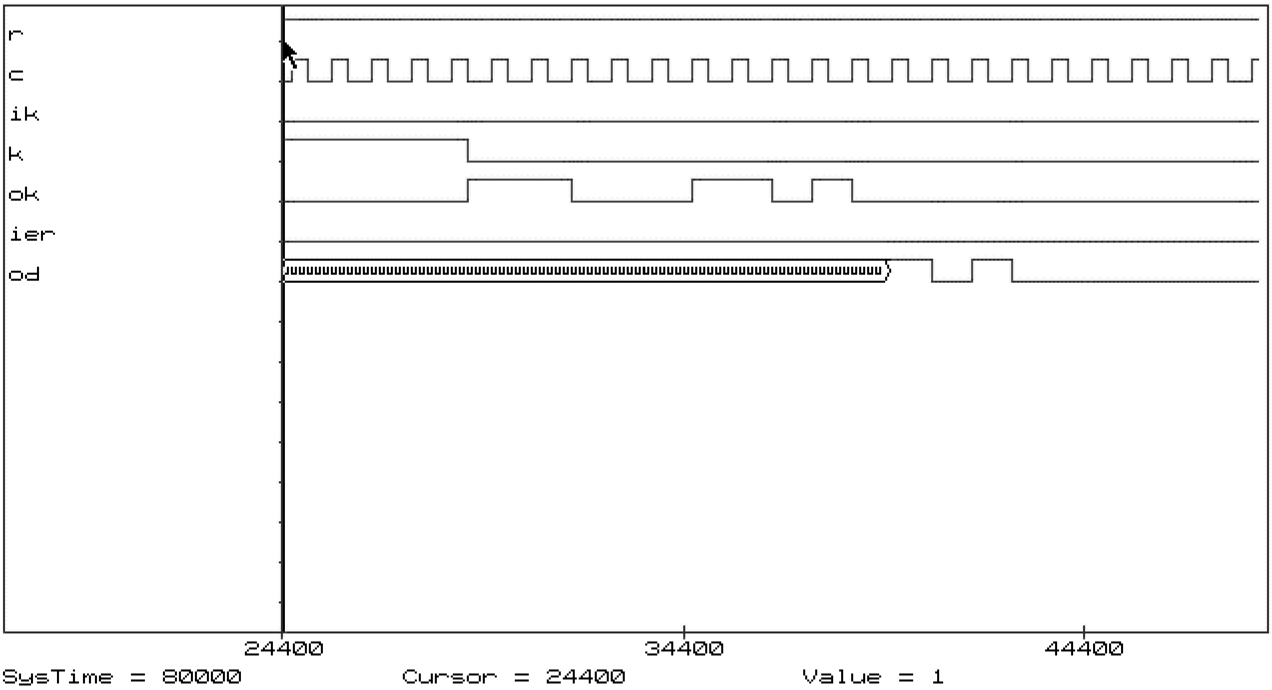
r – reset;
c – clock;
ik – input of the encoder (information sequence of symbols is binary symbols 101 and 26 zeros);;
k – control signal for key;
ok – output of the encoder;
ier – input for imitation of an error (a burst error with length 4 - 1101);
od – output of the decoder.

More in detail

1-st page

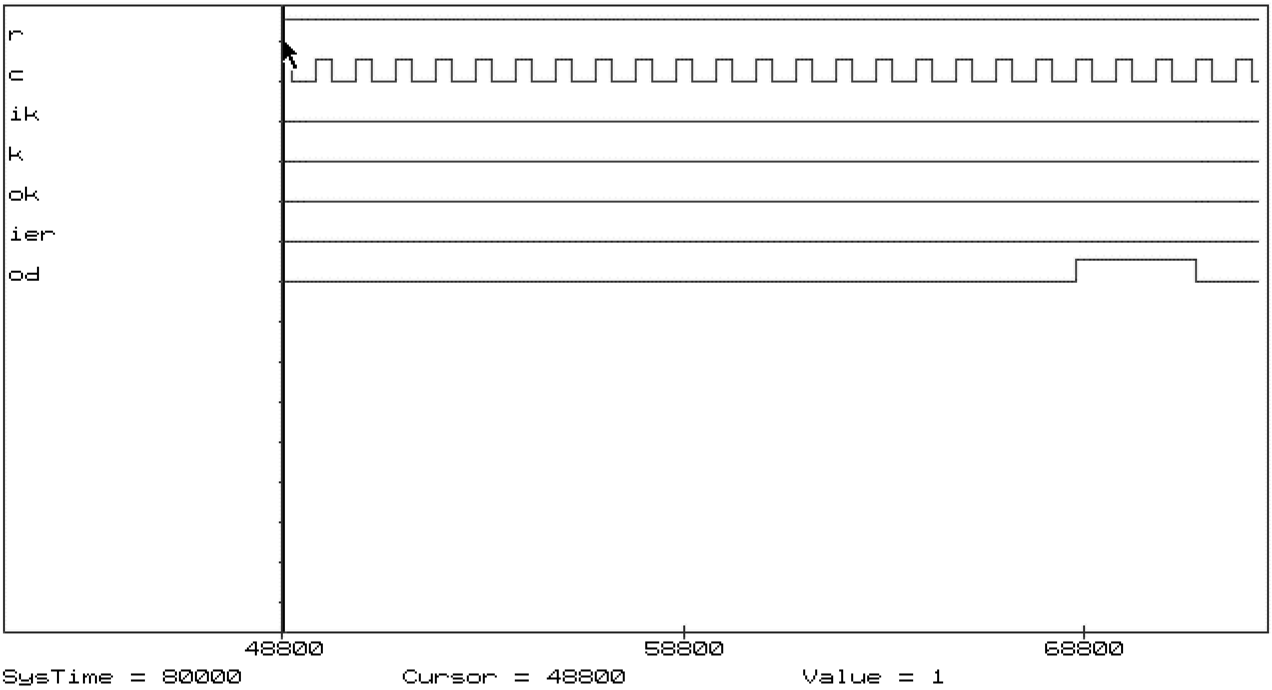


2-nd page (corrected information symbols)



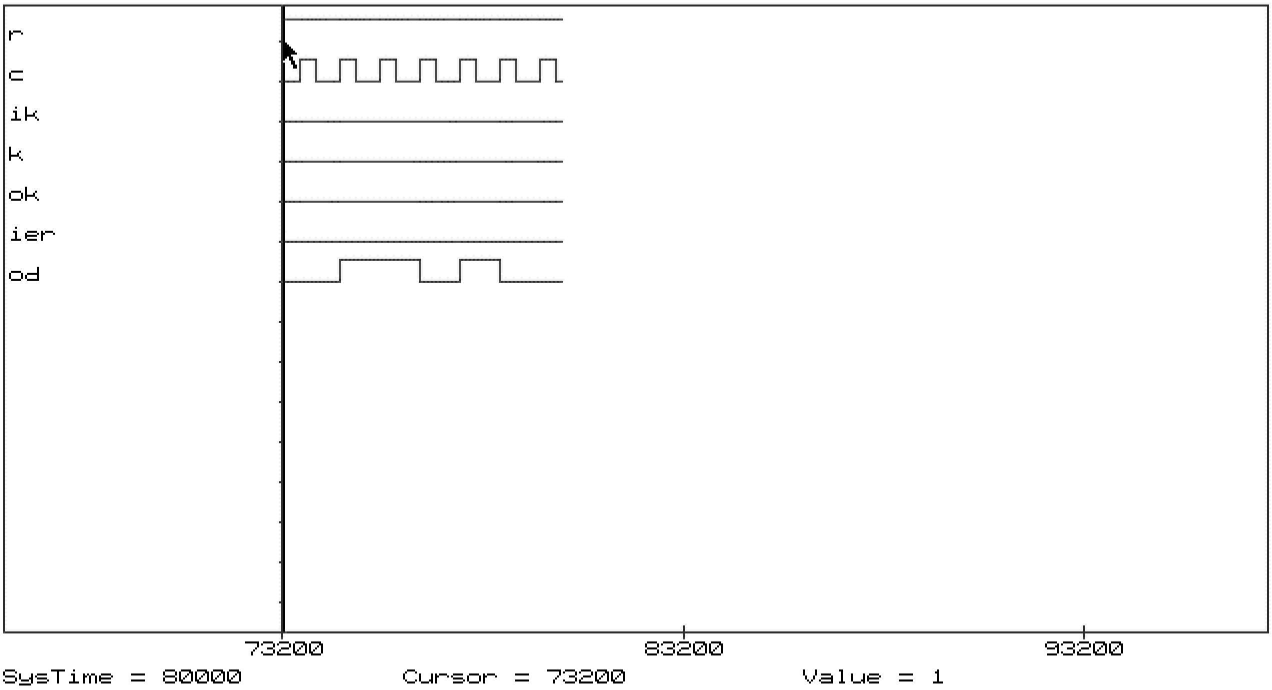
```
SysTime = 80000          Cursor = 24400          Value = 1
```

3-rd page



```
SysTime = 80000          Cursor = 48800          Value = 1
```

4-th page


r
c
ik
k
ok
ier
od
73200
83200
93200
SysTime = 80000
Cursor = 73200
Value = 1

Laboratory № 7

DESIGNING THE ENCODER AND THE DECODER OF THE CYCLIC BURST-ERROR-CORRECTING CODE, CONSTRUCTED BY MEANS OF TECHNIQUE INTERLEAVING, ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the burst-error-correcting code constructed by means of technique interleaving, and execute their simulation.

Task variants:

- 1. The generator polynomial of the (31, 25)-code

$$X^6 + X^5 + X^4 + 1, \text{ if } 2*25 > N > 14.$$

Construct the shortened (2*31 - N, 2*25 - N) single-burst-error-correcting code for burst length t=4.

- 2. The generator polynomial of the (15, 9)-code

$$X^6 + X^5 + X^4 + X^3 + 1, \text{ if } 5 < N < 15.$$

Construct the shortened (2*15 - N, 2*9 - N) single-burst-error-correcting code for burst length t=6.

- 3. The generator polynomial of the (7, 3)-code

$$X^4 + X^3 + X^2 + 1, \text{ if } N < 6.$$

Construct the shortened (3*7 - N, 3*3 - N) single-burst-error-correcting code for burst length t=6.

N is number given by the teacher.

The order of performance of work

- 1. According to the variant determine the generator polynomial for the shortened burst-error-correcting code, constructed by means of technique interleaving.
- 2. Construct the encoder and the decoder on the base of linear switching circuits.
- 3. Develop functional and principal circuits of the encoder and the decoder.
- 4. Make and debug program model.
- 5. Perform simulation of the circuit, imitating the encoder, the binary channel, the decoder. In the binary channel provide a possibility of imitation of burst error in a code combination. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. Determination of the generator polynomial, parameters of the shortened burst-error-correcting code, constructed by means of technique interleaving.
- 5. Determination of the remainder after division polynomial $X^{(n-k+i)}$ by the generator polynomial (by means of simulation of the encoder on the PC).
- 6. Functional circuit of the encoder and the decoder.
- 7. Principal circuits of the encoder and the decoder with burst error imitation possibility (are demonstrated on the PC).

- 8. Time diagrams of simulation of the encoder and the decoder in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. In what the essence of technique interleaving codes consists?
- 2. How is the generator polynomial of the cyclic code, which is turning out using j copies of other cyclic code, determined?
- 3. Draw the scheme of detection and correction of burst errors of length 4 using the cyclic (14, 6)-code, obtained by interleaving the (7, 3)-code with the generator polynomial $X^4 + X^3 + X^2 + 1$.
- 4. How are the shortened cyclic codes formed?
- 5. What is feature of construction of the decoding device for the shortened cyclic code?

Example (N=35)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the burst-error-correcting code constructed by means of technique interleaving, and execute their simulation.

Initial data

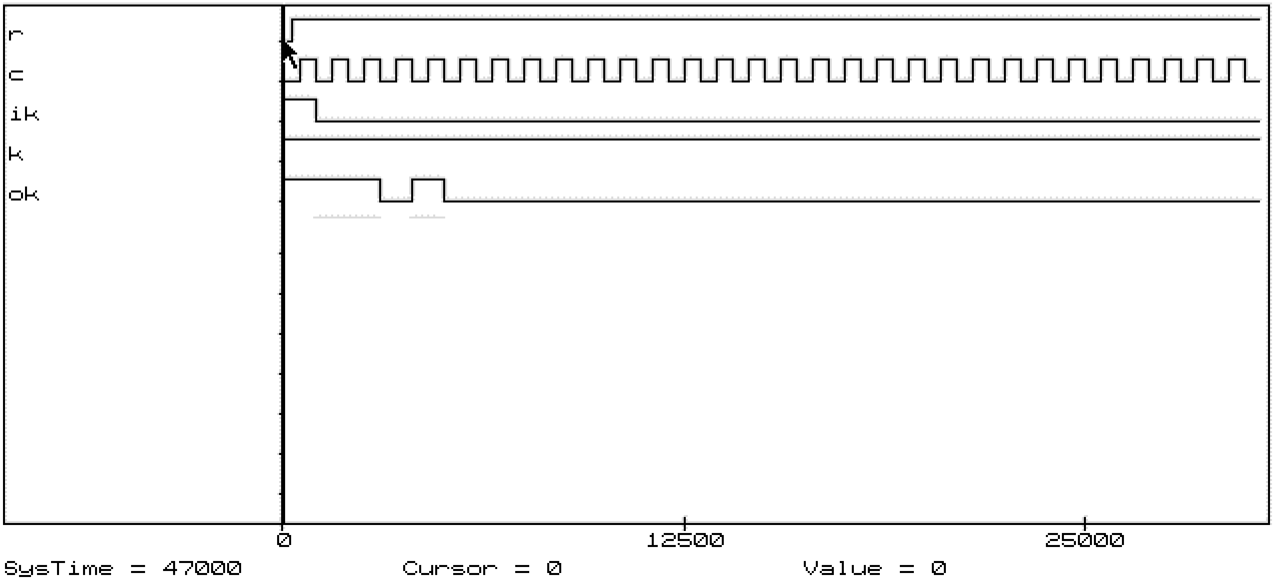
N=35
Generator polynomial of the (31,25)-code is $X^6 + X^5 + X^4 + 1$.
Construct the shortened (2*31-N, 2*25-N)-code, i.e. the (62-35, 50-35)-code or the (27, 15)-code.

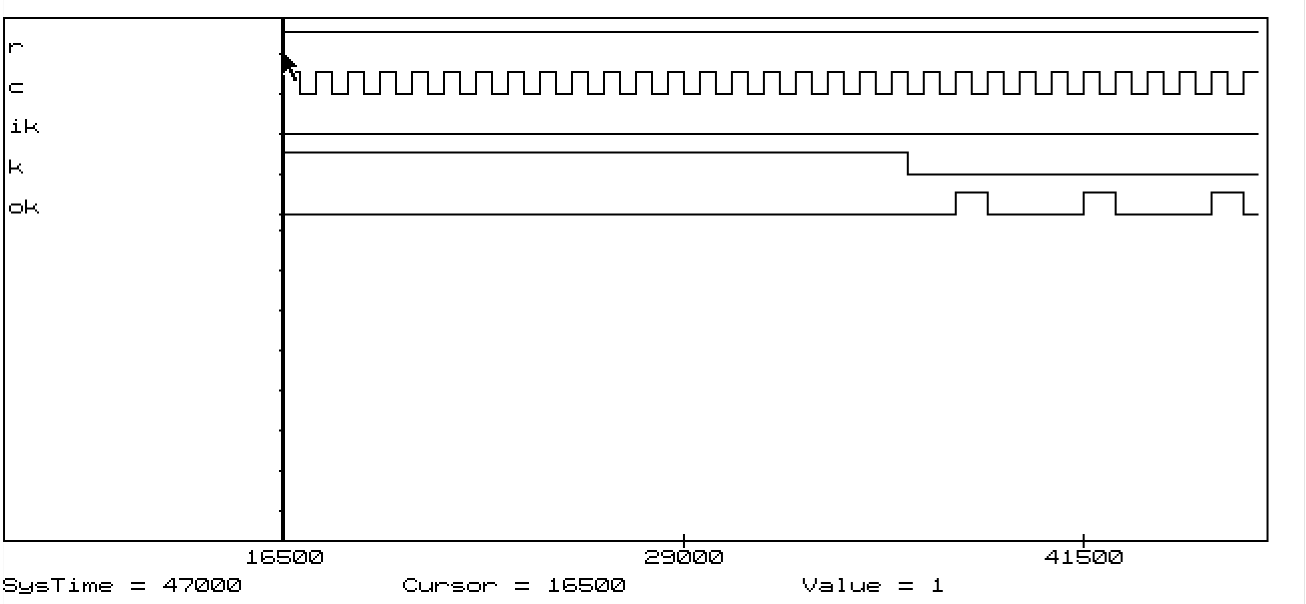
Determination of the generator polynomial, parameters of the shortened burst-error-correcting code, constructed by means of technique interleaving.

The generator polynomial of the (62, 50)-code and the (27, 15)-code:
$X^{2*6} + X^{2*5} + X^{2*4} + 1 = X^{12} + X^{10} + X^8 + 1$,
length of a corrected burst error t=4.

Determination of the remainder after division the polynomial $X^{(n-k+i)}$ by the generator polynomial.

As the (27, 15)-code is the shortened one, for the decoder it is necessary to find remainder after division the polynomial X^{n-k+i} by the generator polynomial. In this case n-k=27-15=12, i=35. The remainder after division X^{12+35} by the generator polynomial is determined by modeling on the PC of the encoder, setting on its input 1 one and 65 zeros. We have obtained 001000100010, or in the polynomial form
$R(X^{12+35}) = X^9 + X^5 + X$.





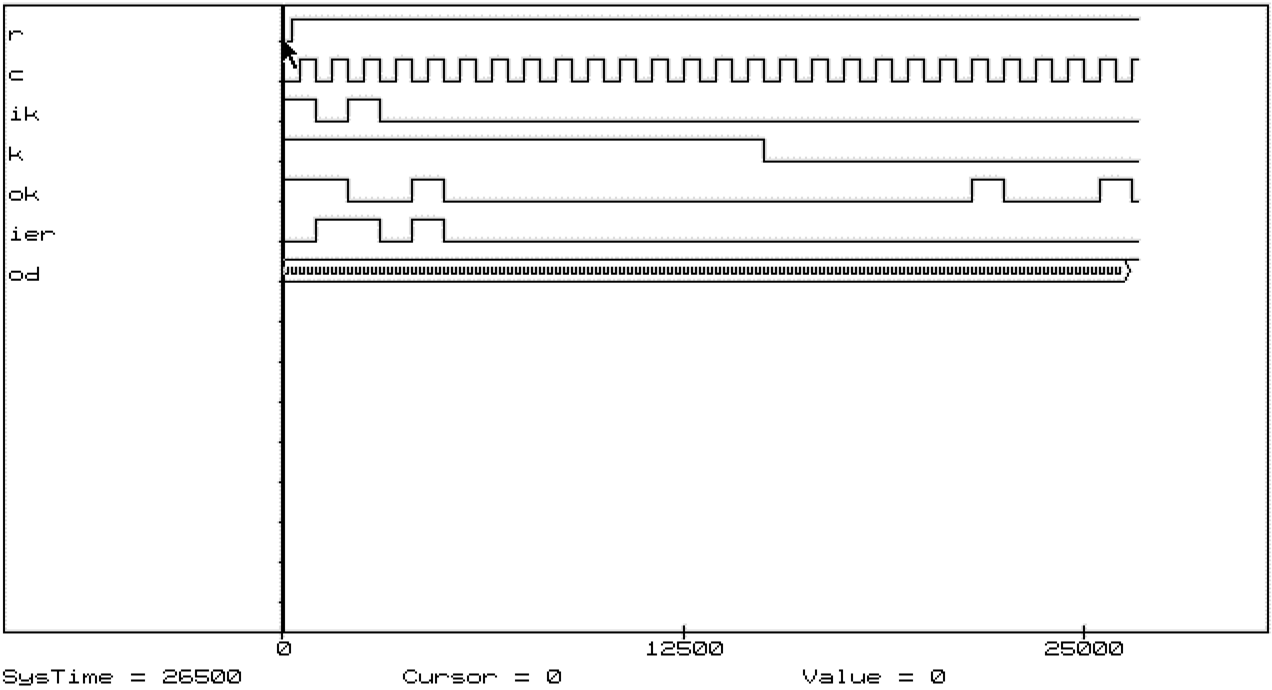
To check up the received result (and at the same time the encoder) it is possible by means of programs of division of polynomials POLY or DIV_POL.

The principal circuit of the encoder and the decoder

In format PDF <lab.pdf>

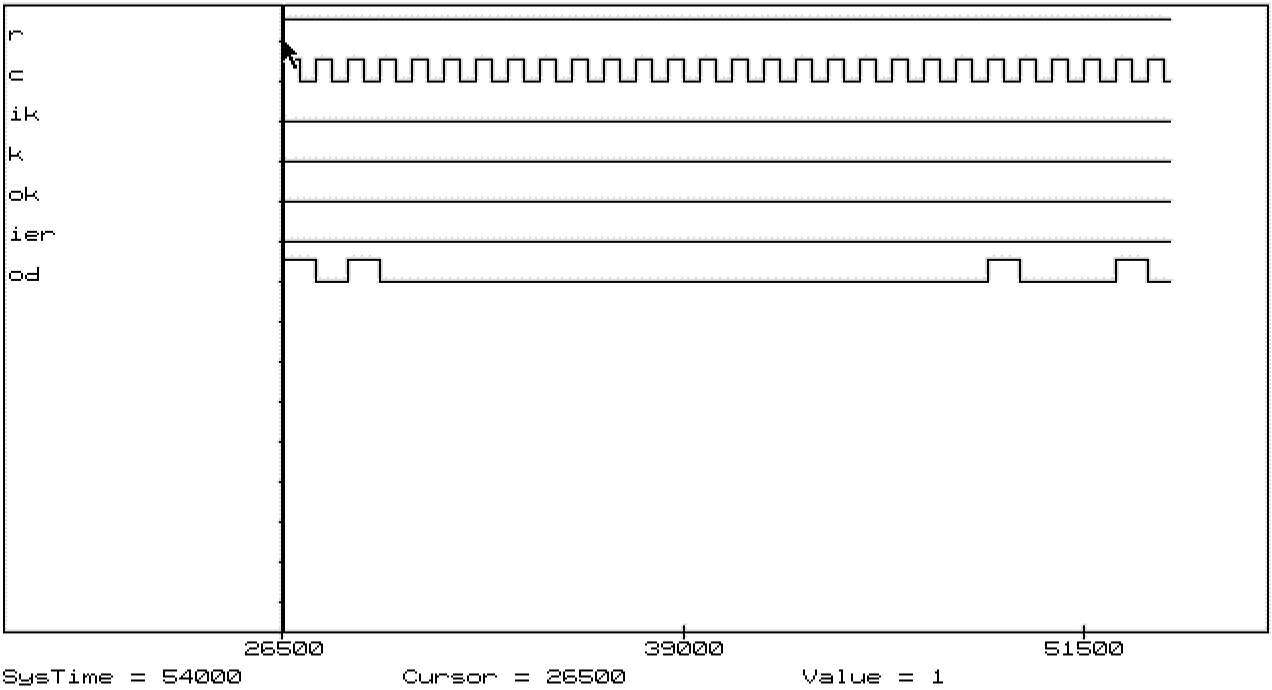
Simulation of the encoder and the decoder

1-st page



```
SysTime = 26500          Cursor = 0          Value = 0
```

2-nd page



```
SysTime = 54000          Cursor = 26500      Value = 1
```

Designation of signals

r – reset;
c – clock;
ik – input of the encoder (information sequence of symbols is binary symbols 101 and 12 zeros);;
k – control signal for key;
ok – output of the encoder;
ier – input for imitation of an error (a burst error with length 4 - 1101);
od – output of the decoder.

Laboratory № 8

DESIGNING THE ENCODER AND THE DECODER OF THE BURST-ERROR-CORRECTING CONVOLUTIONAL IWADARE CODE ON THE BASE OF **ACTIVE-HDL**

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the burst-error-correcting convolutional Iwadare code and execute their simulation.

Task variants:

The burst-error-correcting convolutional (22, 11) Iwadare code (λ=3, n_0=2) for correcting burst of length 6.

Error in the polynomial form E(X):
$E(x) = x^5 + N$,
where N is number given by the teacher in the polynomial form.

The order of performance of work

- 1. Determine the generator matrix for the burst-error-correcting convolutional (22, 11) Iwadare code (λ=3, n_0=2) for correcting burst of length 6.
- 2. Determine the syndrome polynomial.
- 3. Develop functional and principal circuits of the encoder and the decoder.
- 4. Make and debug program model.
- 5. Perform simulation of the circuit, imitating the encoder, the binary channel, the decoder. In the binary channel provide possibility of imitation of burst error in a code combination. Investigate correcting ability of the decoder.

Content of the report

- 1. The title page.
- 2. Task.
- 3. Initial data.
- 4. Determination of parameters of the Iwadare code.
- 5. The generator matrix for the burst-error-correcting convolutional (22, 11) Iwadare code (λ=3, n_0=2) for correcting burst of length 6.
- 6. The syndrome polynomial for the Iwadare code.
- 7. Functional circuit of the encoder and the decoder.
- 8. Principal circuits of the encoder and the decoder with possibility of imitation of burst error (are demonstrated on the PC).
- 9. Time diagrams of simulation of the encoder and the decoder in Time Diagrams Editor (are demonstrated on the PC).

Questions

- 1. In what does difference between convolutional and block codes consist?
- 2. What is constraint length?
- 3. What is information length k of a word, code length n of the block?
- 4. On what length of a codeword is influence of one frame of the information symbols kept?
- 5. What is the systematic tree code?
- 6. What is the convolutional code?
- 7. What is the generator polynomial matrix?
- 8. How is the parity-check matrix of the Wyner-Ash code constructed?
- 9. In what does difference between convolutional and block codes correcting the burst errors consist?
- 10. What is the Iwadare code?
- 11. How are the generator and the parity-check matrices of the Iwadare code constructed?

- 12. What is the length of burst errors, which is capable to correct the decoder of Iwadare code, equal?
- 13. What is the syndrome polynomial?

Example (N=35)

Task: on the base of Active-HDL tools develop models of the encoder and the decoder for the burst-error-correcting convolutional Iwadare code and execute their simulation.

Initial data

N=35
The design parameter of the Iwadare code $\lambda=3$, the frame of a codeword $n_0=2$.

Determination of parameters of the Iwadare code

The most important tree codes are those known as convolutional codes. Unlike the block codes, the codeword of a tree code n_0 (the codeword frame) is formed in dependence not only on the information block k_0 (the information frame), but also on m already transmitted information frames. Let $v=mk_0$. This v is an important descriptor of a convolutional code. It is called the constraint length. There are several other length measures for a tree code. Let $k=(m+1)k_0$. This k is closely related to the constraint length. We call it the wordlength of the convolutional code. The corresponding measure after encoding is called blocklength n. It is given by

$$n=(m+1)n_0=k\frac{n_0}{k_0}.$$

The blocklength is the length of codeword that can be influenced by a single information frame. Because of implementation considerations, practical tree codes use very small integers for k_0 and n_0; typically k_0 is one.

Let λ and n_0 be any positive integers. An Iwadare code is a systematic binary burst-error-correcting convolutional code with an n_0 - 1 by n_0 matrix of generator polynomials given by

$$\mathbf{G}(x)=\begin{bmatrix} 1 & & & & g_1(x) \\ & 1 & & & g_2(x) \\ & & \ddots & & \vdots \\ & & & 1 & g_{(n_0-1)}(x) \end{bmatrix}$$

where matrix element $g_{in_0}(x)$ has been abbreviated to $g_i(x)$ and is given by

$$g_i(x)=x^{(\lambda+1)(2n_0-i)+i-3}+x^{(\lambda+1)(n_0-i)-1} \qquad \text{for } i = 1, \ldots, n_0-1.$$

The largest-degree generator polynomial is $g_1(x)$, which has a degree of $(\lambda+1)(2n_0-1)-2$. Therefore an Iwadare code is a $((m+1)n_0, (m+1)(n_0-1))$ convolutional code where m, the number of frames, is:

$$m=(\lambda+1)(2n_0-1)-2.$$

This code corrects any burst error of length λn_0 or less.

In this case $m = (\lambda+1)(2n_0-1)-2 = 4(4-1)-2 = 12-2 = 10;$ $(m+1)n_0 = 11*2 = 22;$ $(m+1)(n_0-1)=11.$ $\lambda n_0=3*2=6$.

The generator matrix for the burst-error-correcting convolutional (22, 11) Iwadare code ($\lambda=3$, $n0=2$) for correcting burst of length 6

So, it is required to construct (22,11) Iwadare code.

The generator matrix looks like

$G(x) = [1 \ g_1(x)]$,

where

$$g_1(x) = x^{(\lambda+1)(2n_0-1)+1-3}+x^{(\lambda+1)(n_0-1)-1}=x^{4*3+1-3}+x^{4-1} = x^{10}+x^3.$$

$G(x) = [1 \;\; x^{10} + x^3]$

The code corrects any burst error of length no more $\lambda n_0 = 6$.

The generator polynomial

Because $n_0 - k_0 = 1$, there is only a single syndrome polynomial:

$$s(x) = \sum_{i=0}^{n_0-1} g_i(x) e_i(x) + e_{n_0}(x) =$$

$$= e_{n_0}(x) + \sum_{i=0}^{n_0-1} \left[x^{(\lambda+1)(n_0-i)-1} + x^{(\lambda+1)(2n_0-i)+i-3} \right] e_i(x).$$

The polynomial $c_2(x)$ is the polynomial of parity symbols, and $c_1(x)$ is the polynomial of information symbols. The corresponding error polynomials are $e_1(x)$ and $e_2(x)$. The syndrome polynomial is

$s(x) = e_2(x) + (x^3 + x^{10})e_2(x)$

where for a burst error beginning in the zero frame,

$e_2(x) = e_{20} + e_{21}x + e_{22}x^2,$

$e_1(x) = e_{10} + e_{11}x + e_{12}x^2 + e_{13}x^3.$

In this case for determination and analysis of a syndrome polynomial it is necessary to consider four consecutive frames of the Iwadare code.

	Frames						
	3-rd		2-nd		1-st		0-th
	┌	┐	┌	┐	┌	┐	
	↑	↓	↑	↓	↑	↓	
$e_1(x)$	e_{13}	↓	e_{12}	↓	e_{11}	↓	e_{10}
	↑	↓	↑	↓	↑	↓	↑
$e_2(x)$	e_{23}	↓	e_{23}	↓	e_{21}	↓	e_{20}
	↑	↓	↑	↓	↑	↓	↑
		└	┘	└	┘	└	┘

Notice that e_{23} must be zero if the burst begins in the zeroth frame, i.e. e_{10} or e_{20} are not zero. Further, if e_{10} is not zero, then e_{13} is zero.

Let's consider the coefficients of $s(x)$ for the two cases, in which the burst error begins at e_{10} or e_{20}.

Syndrome polynomial

	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	s_{10}	s_9	s_8	s_7	s_6	s_5	s_4	s_3	s_2	s_1	s_0

equals:

if burst starts at e_{10}

	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	$\mathbf{s_{10}}$	s_9	s_8	s_7	s_6	s_5	s_4	$\mathbf{s_3}$	s_2	s_1	s_0
…	0	0	0	12	11	10	0	0	0	0	12	11	10	22	21	20

if burst starts at e_{20}

	s_{15}	s_{14}	s_{13}	s_{12}	s_{11}	$\mathbf{s_{10}}$	s_9	s_8	s_7	s_6	s_5	s_4	$\mathbf{s_3}$	s_2	s_1	s_0
…	0	0	13	12	11	0	0	0	0	13	12	11	0	22	21	20

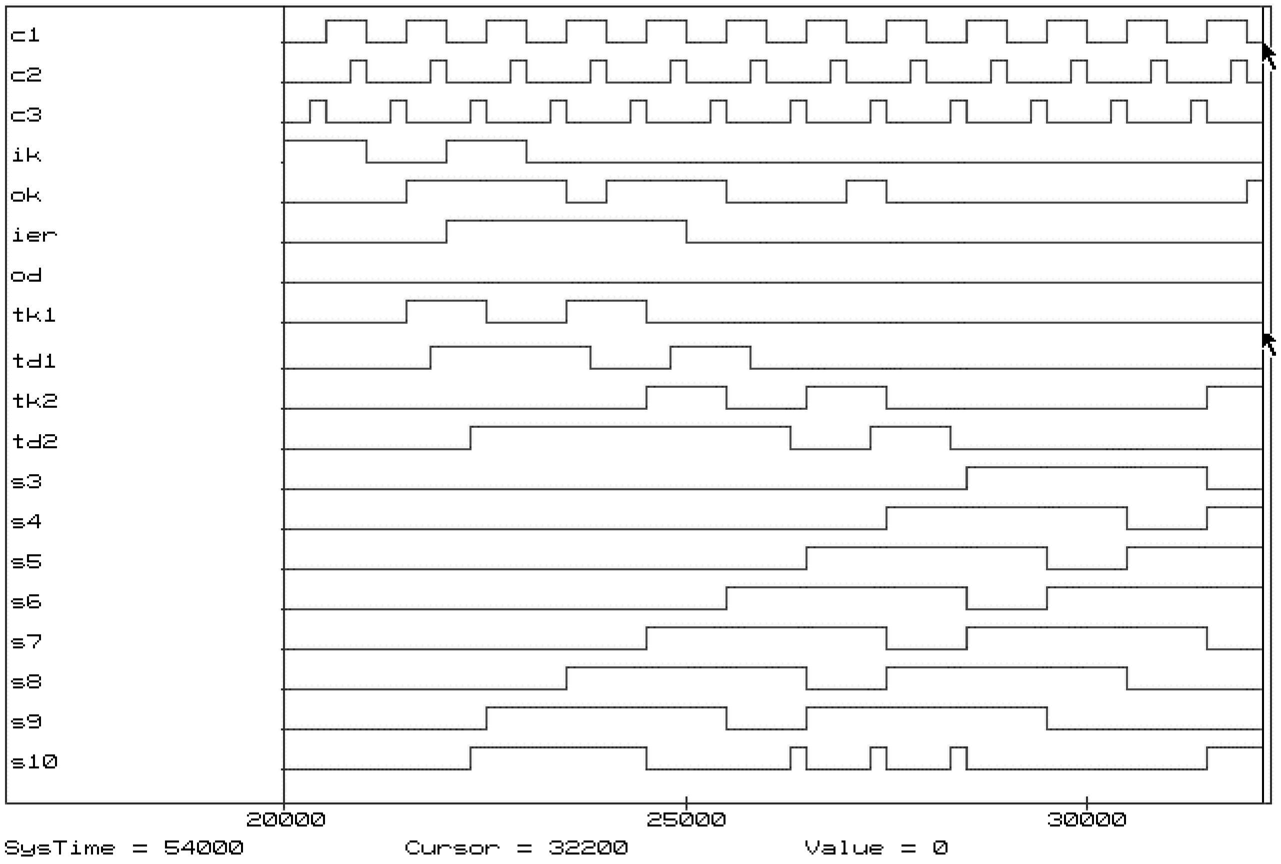
For construction of the decoder it is necessary to test syndrome positions s_3 and s_{10}. If both are equal to one, then it is necessary correction.

The principal circuit of the encoder and the decoder

In format PDF <lab.pdf>

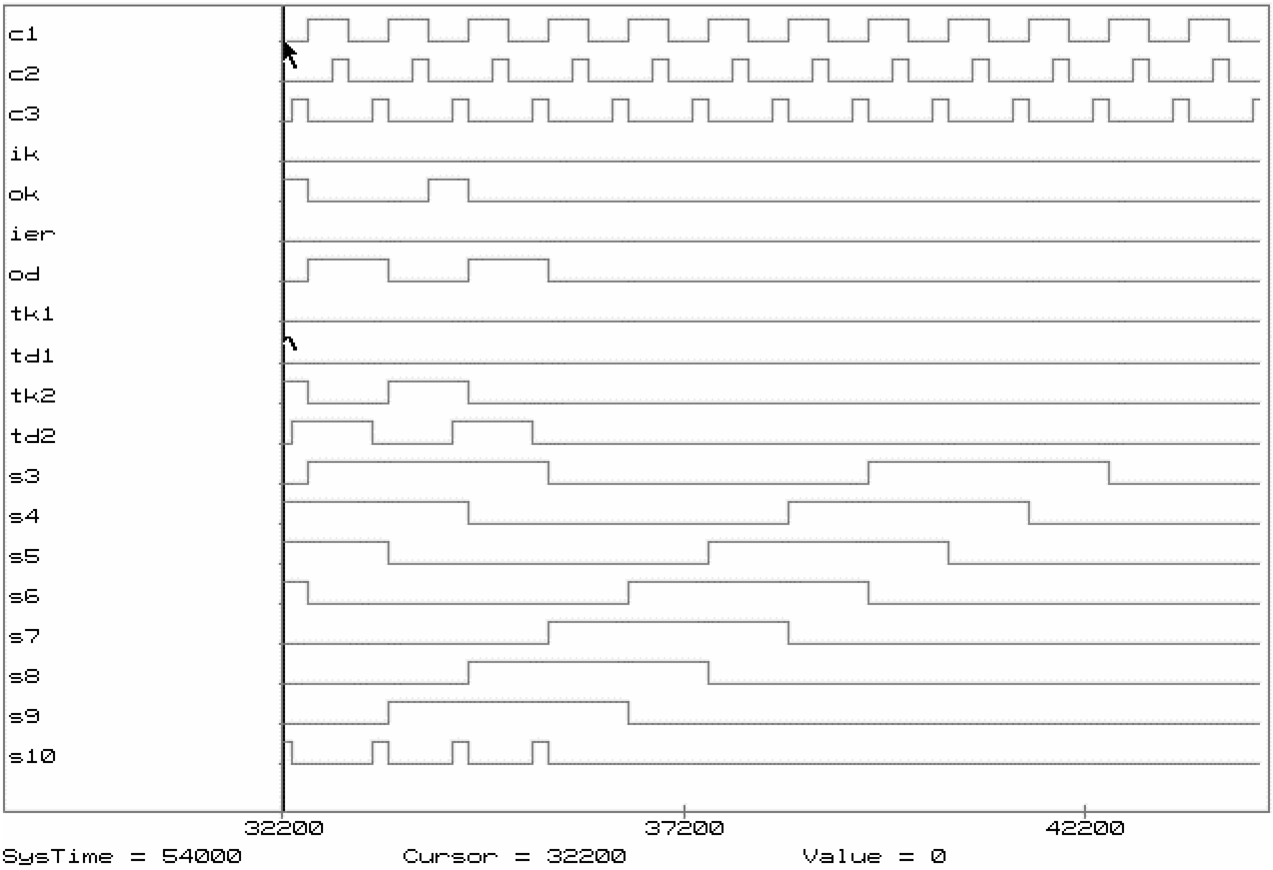
Simulation of the encoder and the decoder

1-st page



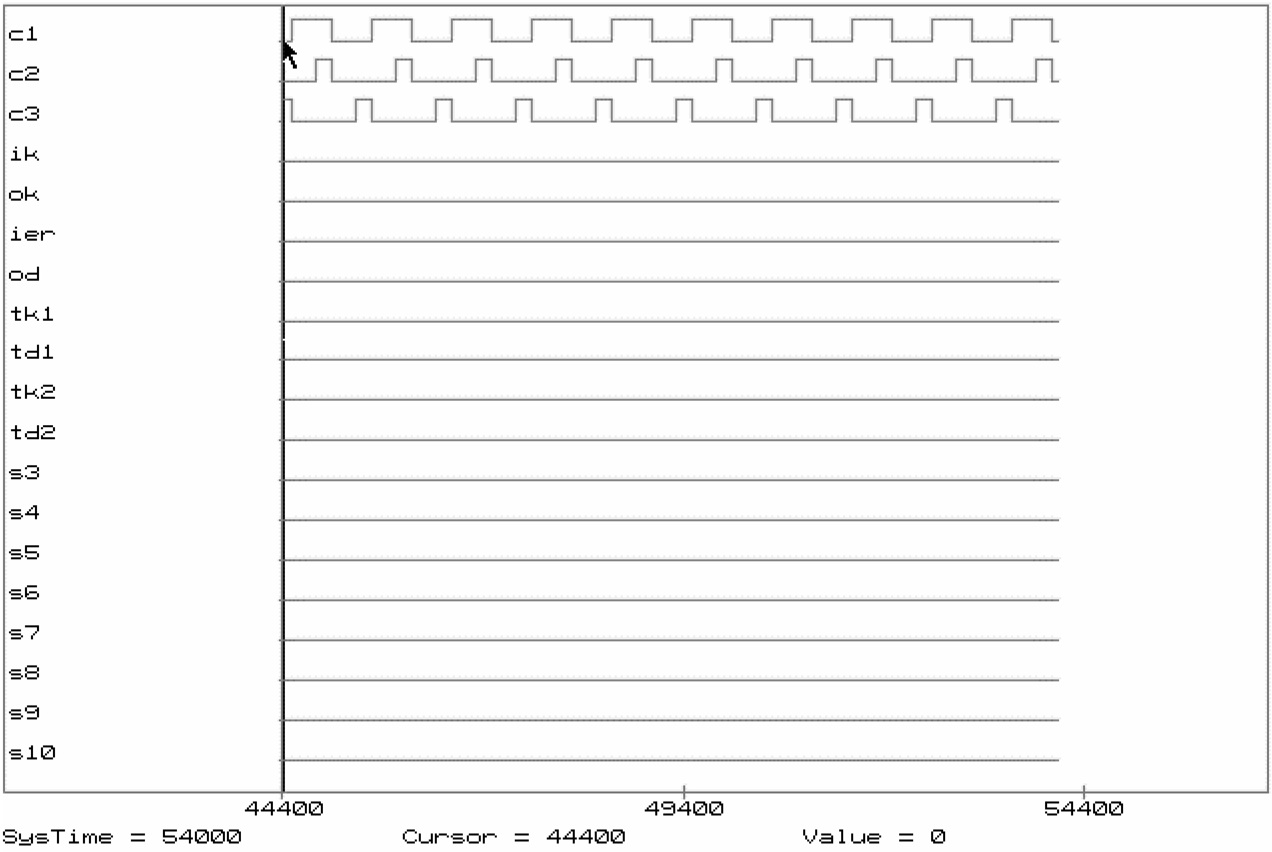
```
SysTime = 54000          Cursor = 32200          Value = 0
```

2-nd page



SysTime = 54000 Cursor = 32200 Value = 0

3-rd page



SysTime = 54000 Cursor = 44400 Value = 0

Designation of signals

VCC0, VCC1 – levels 0 and 1;

c1, c2, c3 - clock (c1 is used for transformation of a parallel code in consecutive and vice versa, and also synchronization of the trigger on the output of the decoder; c2 - synchronization of the register of information symbols $c_1(x)$; c3 - synchronization of the register of parity-check symbols $c_2(x)$);

ik – input of the encoder (information sequence of symbols is binary symbols 101);

ok – output of the encoder after imitation of the burst error;

ier – input for imitation of an error (a burst error with length 6 – 111111);

od – output of the decoder;

tk1, tk2 – test points of the encoder;

td1, td2 – test points of the decoder;

s3-s10 – stage of the syndrome register from 3-rd to 10-th inclusive.

The note: on the time diagram (see page 2) simultaneous values s3=1 and s10=1 (namely in this case correction is carried out) occurs three times and only information symbols are corrected. Three errors from burst errors of length 6 distort information symbols (the decoder corrects these symbols), other three errors distort parity-check symbols (the decoder does not correct these symbols as on the output of the decoder only information symbols are used).

LITERATURE

Main:

1. Richard E Blahut. Theory and Practice of Error Control codes/ Addison-Wesley Publishing Company, 1986.– 576 p.
2. Peterson W.W., Weldon E.J.,Jr. Error-correcting codes.- 2nd ed.- Cambridge (Mass.): MIT Press., 1971.– 595 p.
3. Блейхут Р. Теория и практика кодов, контролирующих ошибки: Пер. с англ. - М.: Мир, 1986. - 576 с.: ил.
4. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки. - М.: Мир, 1976. - 595с.: ил. (pp. 134-138, 304-321, 251-273)
5. Кузьмин В. П., Кедрус В. А. Основы теории информации и кодирования. - К.: Вища шк. Головное изд-во, 1986. – 238 с.
6. Цымбал В. П. Теория информации и кодирования: Учебник. - К.: Вища шк., 1992. - 263 с.: ил.

Additional:

7. Сидельников В.М. Теория кодирования. – М.: 2006. – 289 с.
8. Shu Lin, Daniel J. Costello. Error Control Coding.Fundamentals and Applications/ Prentice-Hall, 1983. - 617 p.
9. Michael Purser. Introduction to Error-Correcting Codes/ Artech House, 1995.-133 p.
10. W. Carry Huffman, Vera Pless. Fundamentals of Error-Correcting Codes/ Cambridge University Press., 2003. - 662 p.
11. Robert H. Morelos-Zaragoza. The art of error correcting coding/ SONY Computer Science Laboratories, Inc. JAP, John Wiley & Sons, Ltd, 2002. – 219 p.
12. Todd K. Moon. Error Correction Coding. Mathematical Methods and Algorithms/ Wiley-Interscience, John Wiley & Sons, Inc., 2005.- 755 p.
13. Огнев И. В., Сарычев К. Ф. Надежность запоминающих устройств. - М.: Радио и связь, 1988. - 224 с.
14. Дяченко О.Н. Графический способ представления сверточных кодов// Наукові праці Донецького національного технічного університету. Серія “Інформатика, кібернетика і обчислювальна техніка” (ІКОТ-2007). Випуск 8 (120) - Донецьк: ДонНТУ, 2007. – С. 89-98.
15. Дяченко О.Н. Аппаратная реализация и корректирующие возможности кодов Рида-Соломона// Наукові праці Донецького національного технічного університету. Серія “Проблеми моделювання та автоматизації проектування динамічних систем” (МАП-2007). Випуск: 6 (127) - Донецьк: ДонНТУ. - 2007. – С. 113-121.

MINISTRY OF EDUCATION AND SCIENCE, YOUTH AND SPORTS OF UKRAINE

DONETSK NATIONAL TECHNICAL UNIVERSITY

Laboratory report №__

Course of studies
“The name of discipline”

Theme “The name of a theme”

Student of the group _______

Donetsk-20XX

THE ORDER OF DEFENCE OF A LABORATORY

Defence of a laboratory consists of two parts:

1. **Demonstration of laboratory results on a computer.**
2. **Defence of the laboratory report.**

Demonstration of laboratory results on a computer should be with explaining comments:

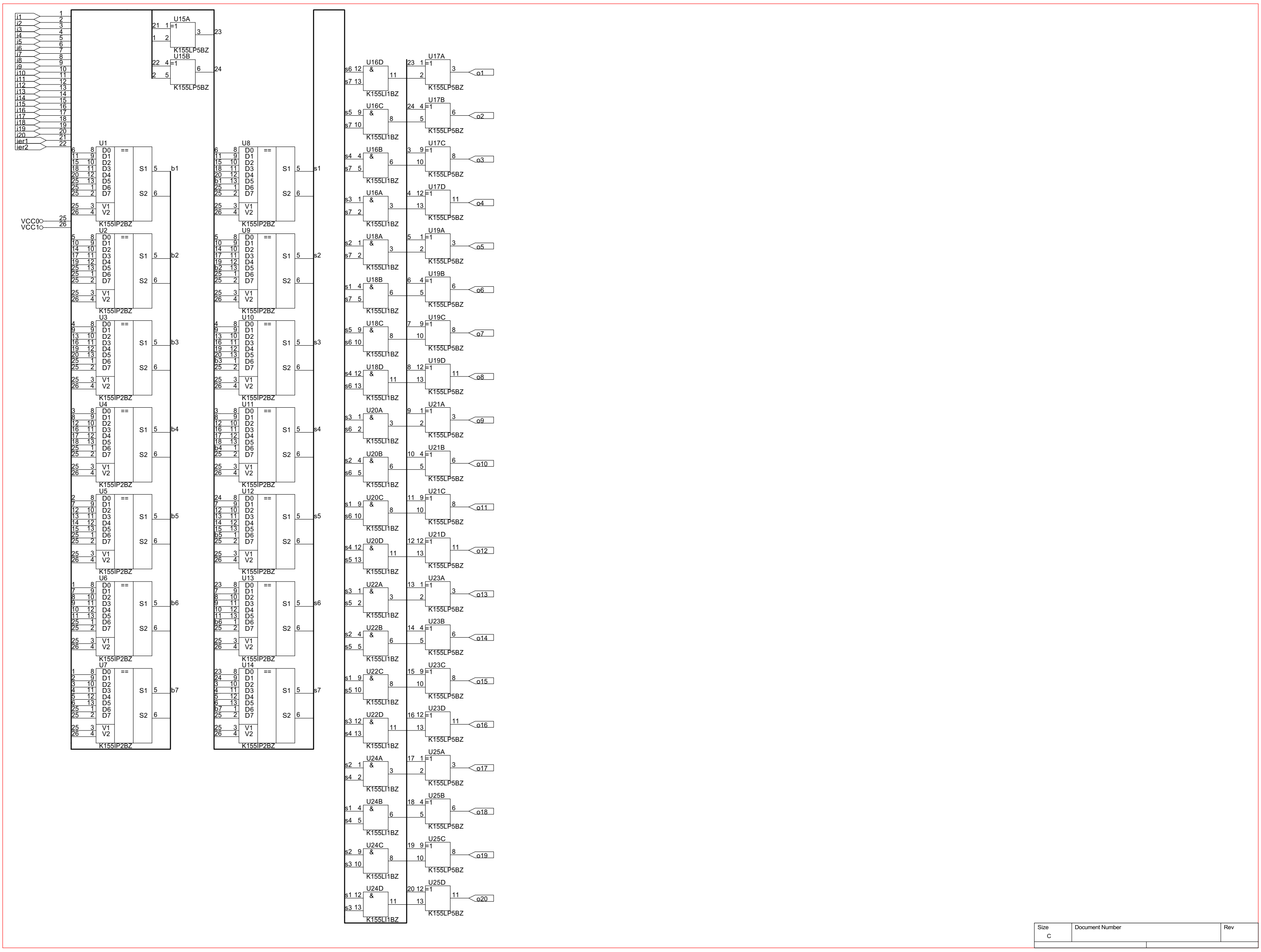
1. **The name of a code.**
2. **Correcting abilities of a code.**
3. **Show on the circuit the encoder, imitation of errors (specify in what positions of a codeword the errors are simulated) and the decoder.**
4. **Features of hardware of the encoder and the decoder.**
5. **Show on the time diagram detection (or correction) errors. Examine three situations:**

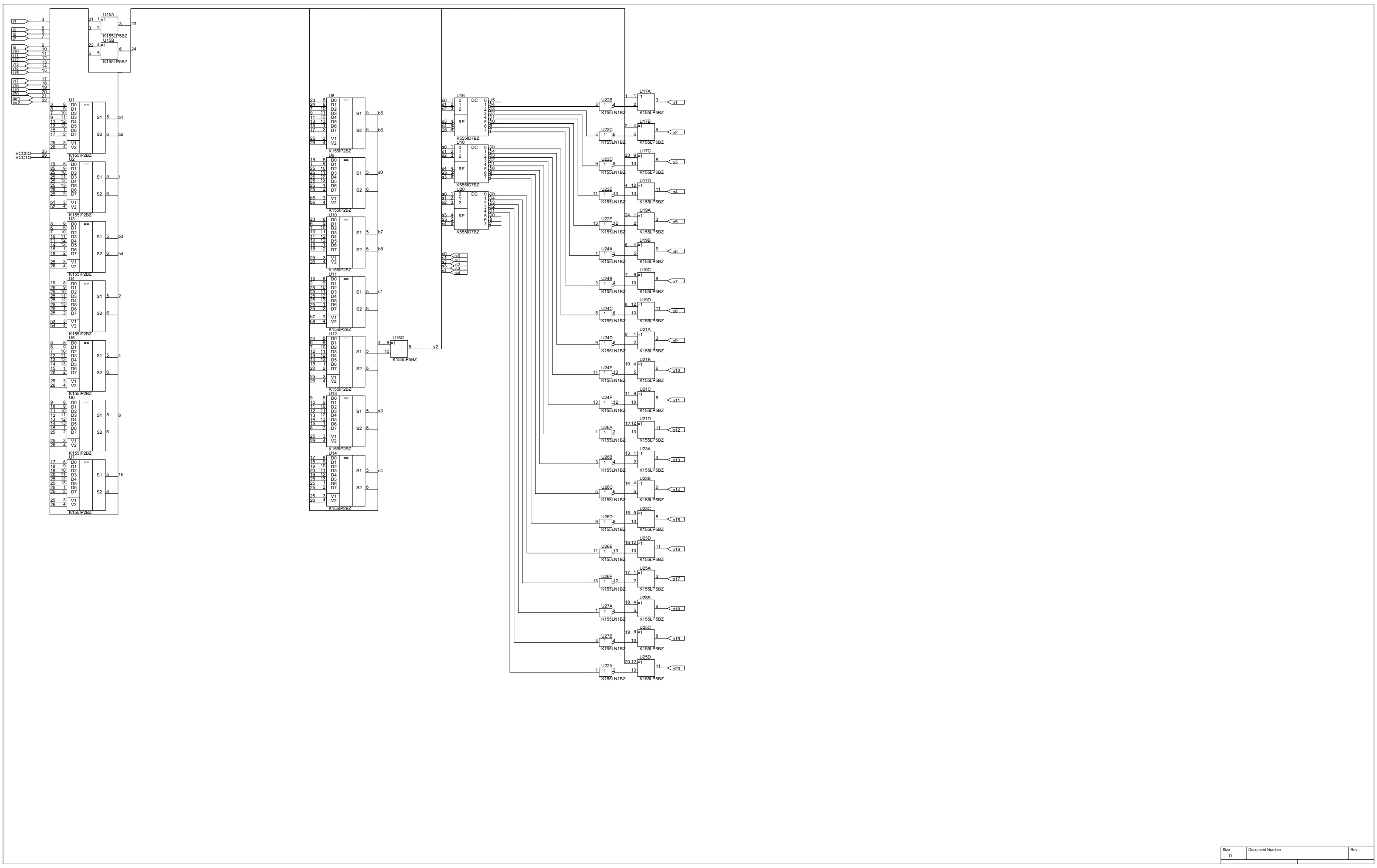
   1) variant without imitation of errors;

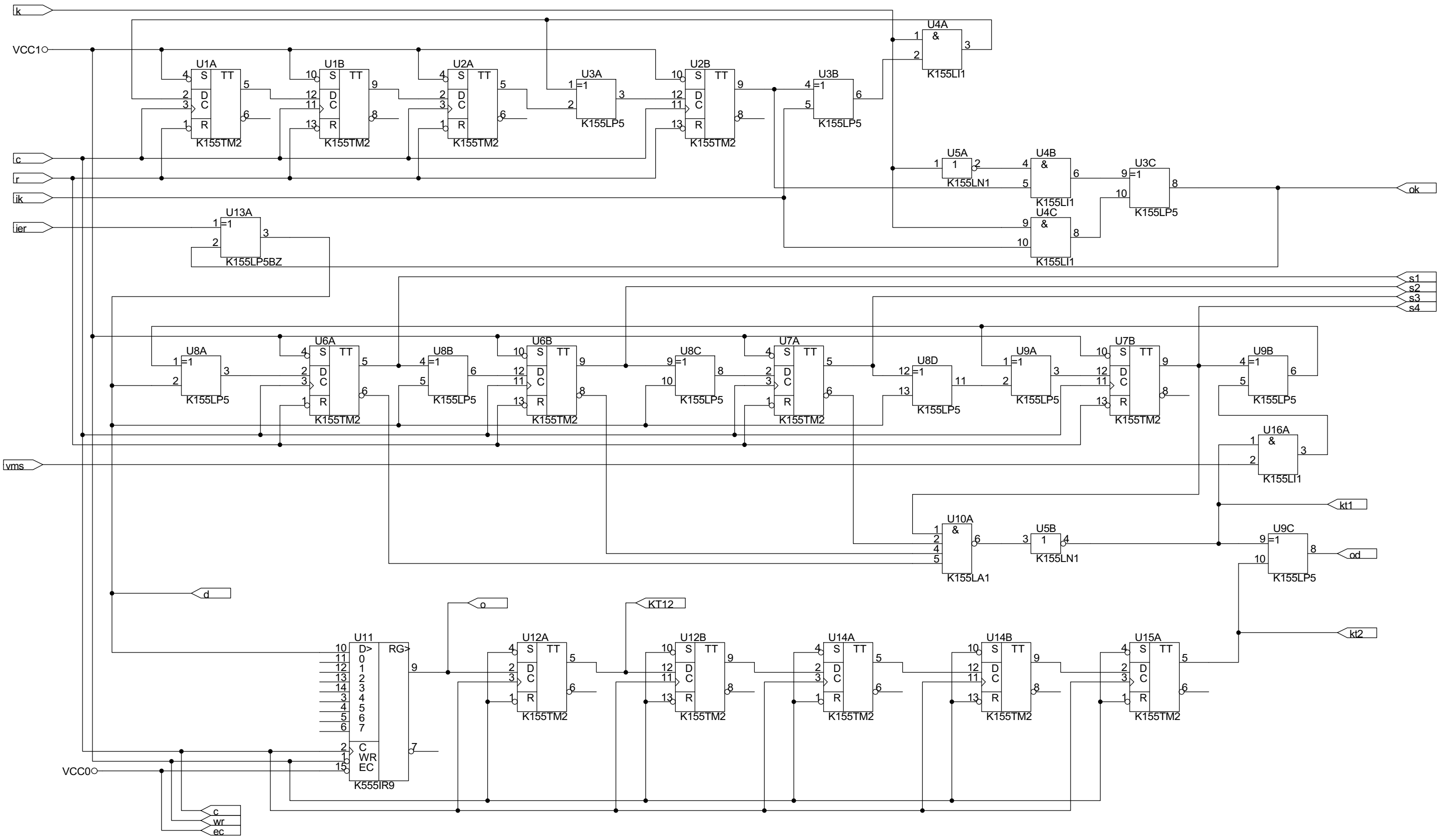
   2) variant with imitation of errors which are corrected (or are detected);

   3) variant with imitation of errors which exceed correcting abilities of a code.

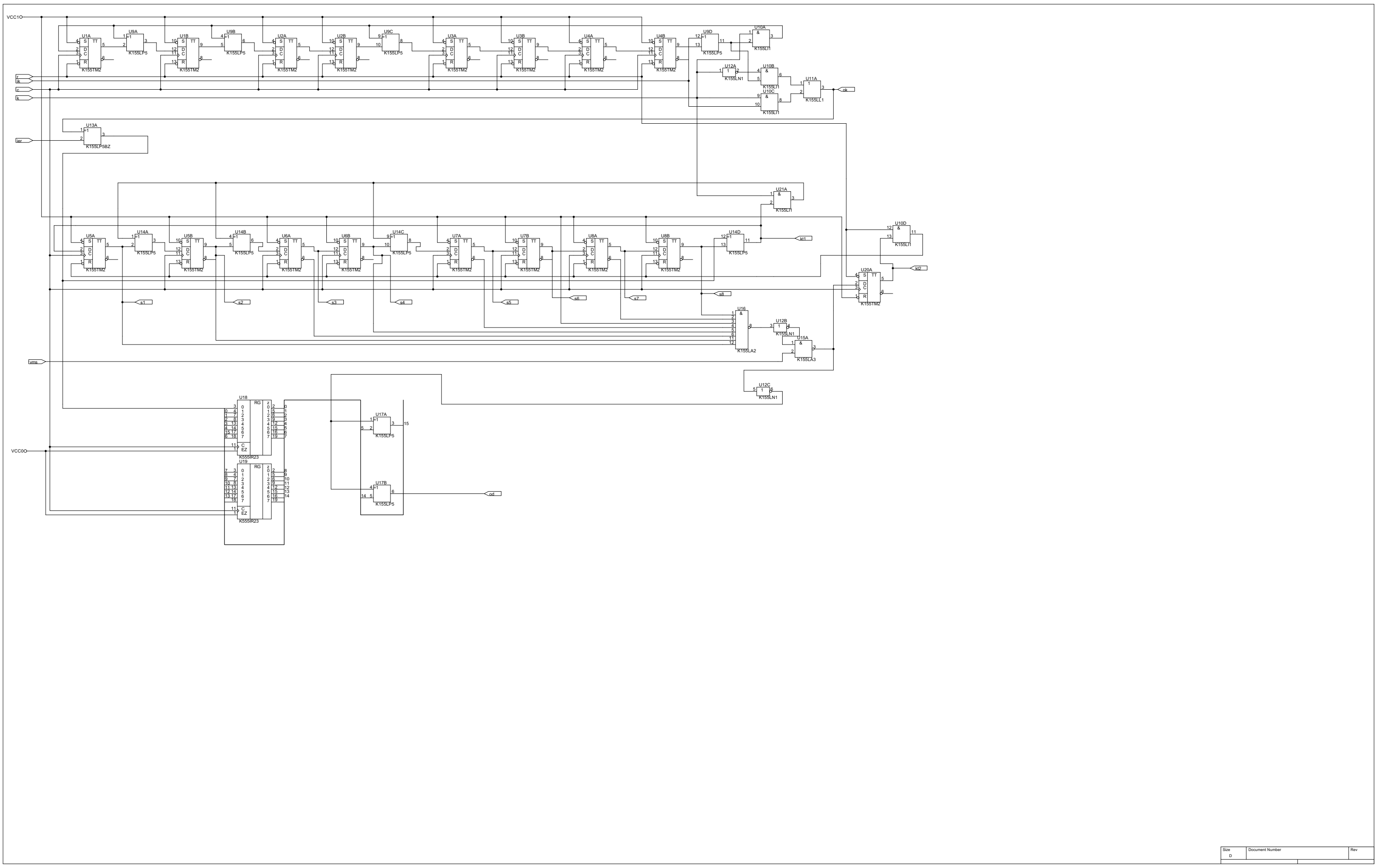
   Explain the obtained results.

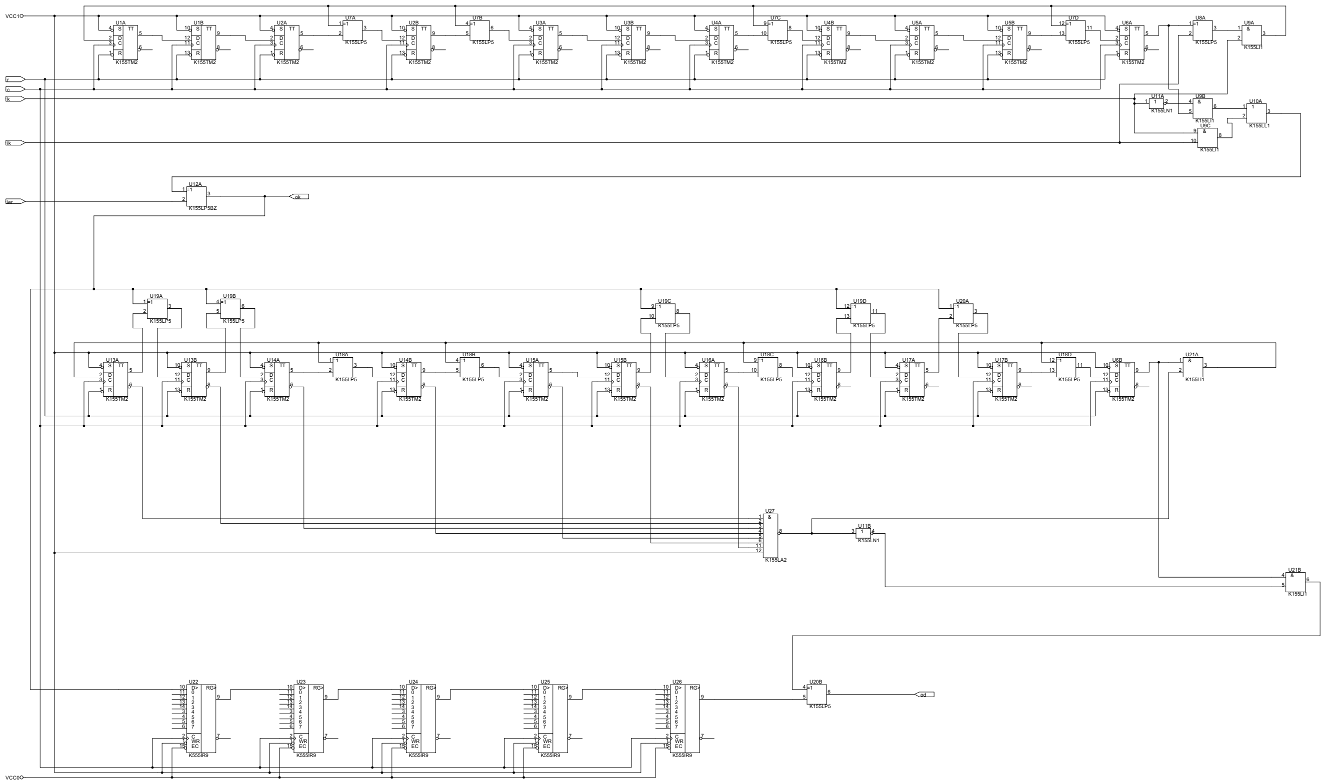

i1
i2
i3
i4
i5
i6
i7
i8
i9
i10
i11
i12
i13
i14
i15
i16
i17
i18
i19
i20
ier1
ier2
VCC0
VCC1
U15A
U15B
K155LP5BZ
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
K155IP2BZ
D0
D1
D2
D3
D4
D5
D6
D7
V1
V2
S1
S2
b1
b2
b3
b4
b5
b6
b7
s1
s2
s3
s4
s5
s6
s7
U16A
U16B
U16C
U16D
U18A
U18B
U18C
U18D
U20A
U20B
U20C
U20D
U22A
U22B
U22C
U22D
U24A
U24B
U24C
U24D
K155LI1BZ
U17A
U17B
U17C
U17D
U19A
U19B
U19C
U19D
U21A
U21B
U21C
U21D
U23A
U23B
U23C
U23D
U25A
U25B
U25C
U25D
o1
o2
o3
o4
o5
o6
o7
o8
o9
o10
o11
o12
o13
o14
o15
o16
o17
o18
o19
o20
Size
C
Document Number
Rev


U15A
U15B
U15C
K155LP5BZ
U1
U2
U3
U4
U5
U6
U7
U8
U9
U10
U11
U12
U13
U14
K155P2BZ
U16
U18
U20
K555ID7BZ
U17A
U17B
U17C
U17D
U19A
U19B
U19C
U19D
U21A
U21B
U21C
U21D
U23A
U23B
U23C
U23D
U25A
U25B
U25C
U25D
U22A
U22B
U22C
U22D
U22E
U22F
U24A
U24B
U24C
U24D
U24E
U24F
U26A
U26B
U26C
U26D
U26E
U26F
U27A
U27B
K155LN1BZ
VCC0
VCC1
Size
D
Document Number
Rev



Size	Document Number	Rev
B		

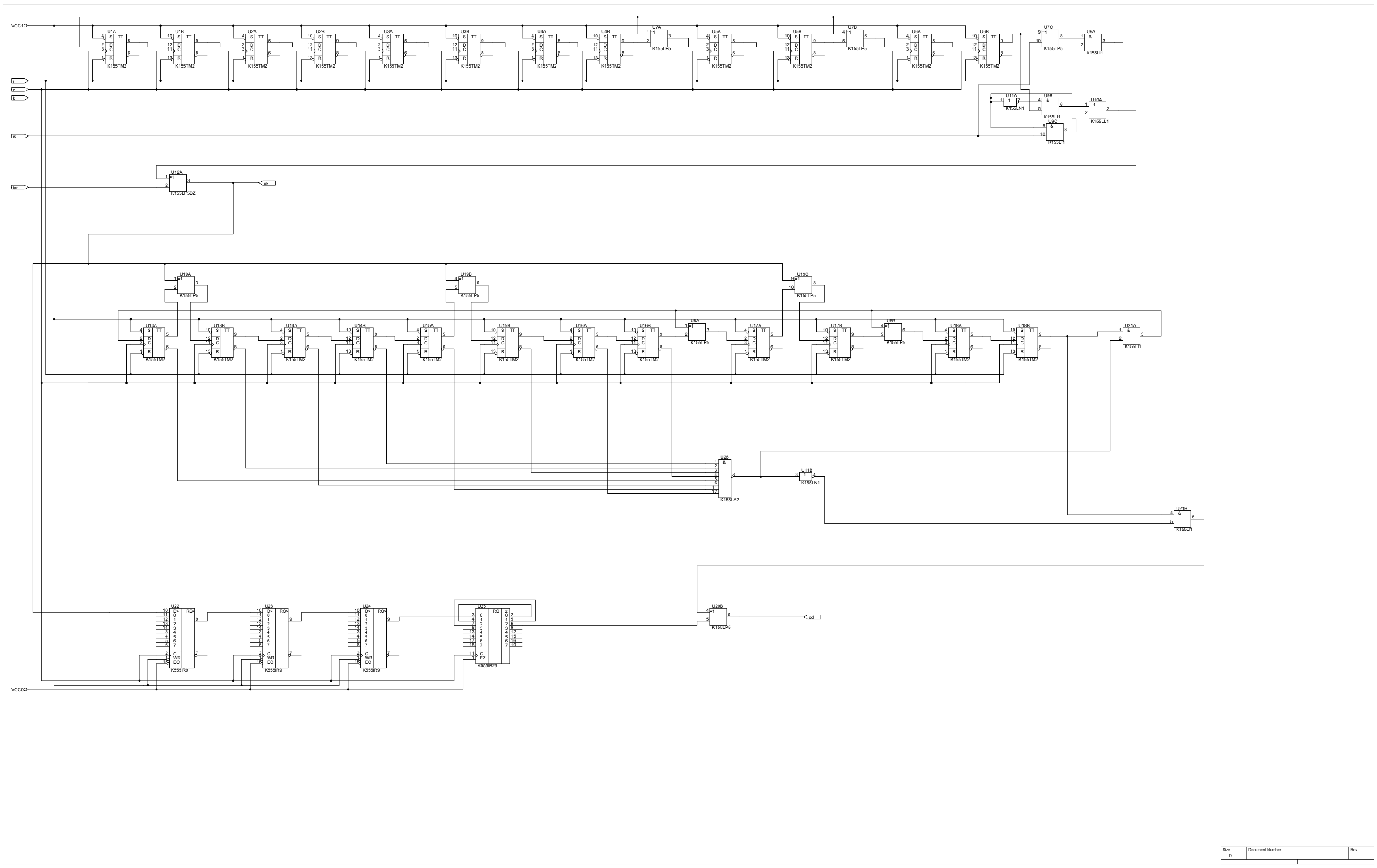

VCC1
r
ik
c
k
ier
U1A K155TM2
U9A K155LP5
U1B K155TM2
U9B K155LP5
U2A K155TM2
U2B K155TM2
U9C K155LP5
U3A K155TM2
U3B K155TM2
U4A K155TM2
U4B K155TM2
U9D K155LP5
U10A K155LI1
U12A K155LN1
U10B K155LI1
U10C K155LI1
U11A K155LL1
ok
U13A K155LP5BZ
U21A K155LI1
U5A K155TM2
U14A K155LP5
U5B K155TM2
U14B K155LP5
U6A K155TM2
U6B K155TM2
U14C K155LP5
U7A K155TM2
U7B K155TM2
U8A K155TM2
U8B K155TM2
U14D K155LP5
kt1
U10D K155LI1
kt2
U20A K155TM2
s1
s2
s3
s4
s5
s6
s7
s8
U16 K155LA2
U12B K155LN1
U15A K155LA3
U12C K155LN1
vms
U18 RG K555IR23
U19 RG K555IR23
VCC0
U17A K155LP5
U17B K155LP5
15
od
Size
D
Document Number
Rev

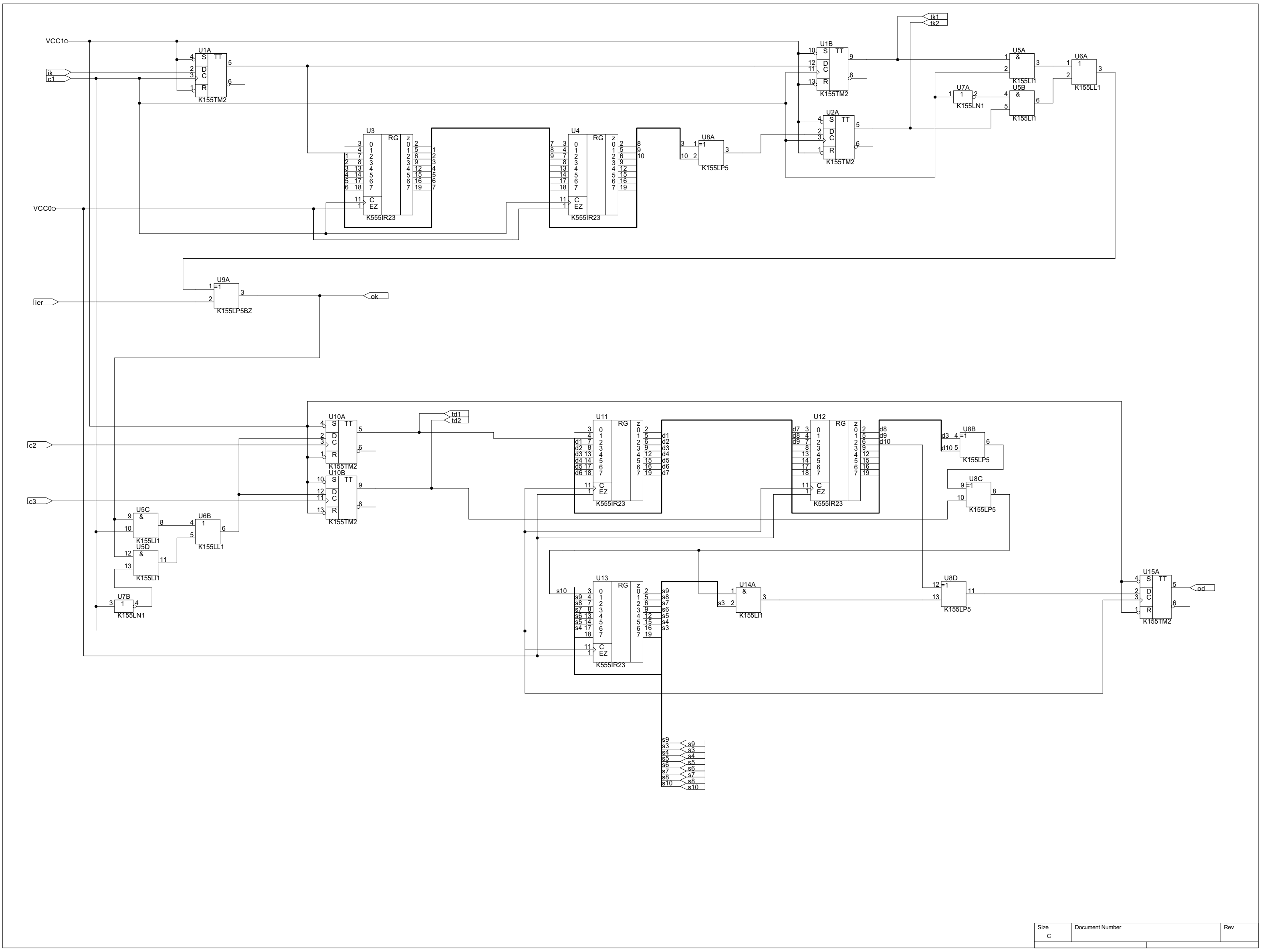


Size
D

Document Number

Rev


VCC1
r
c
k
ik
ier
ok
VCC0
od
U1A K155TM2
U1B K155TM2
U2A K155TM2
U2B K155TM2
U3A K155TM2
U3B K155TM2
U4A K155TM2
U4B K155TM2
U7A K155LP5
U5A K155TM2
U5B K155TM2
U7B K155LP5
U6A K155TM2
U6B K155TM2
U7C K155LP5
U9A K155LI1
U11A K155LN1
U9B K155LI1
U9C K155LI1
U10A K155LL1
U12A K155LP5BZ
U19A K155LP5
U19B K155LP5
U19C K155LP5
U13A K155TM2
U13B K155TM2
U14A K155TM2
U14B K155TM2
U15A K155TM2
U15B K155TM2
U16A K155TM2
U16B K155TM2
U8A K155LP5
U17A K155TM2
U17B K155TM2
U8B K155LP5
U18A K155TM2
U18B K155TM2
U21A K155LI1
U26 K155LA2
U11B K155LN1
U21B K155LI1
U22 K555IR9
U23 K555IR9
U24 K555IR9
U25 K555IR23
U20B K155LP5
S
TT
D
C
R
=1
&
1
RG
D>
WR
EC
EZ
z
Size
D
Document Number
Rev


VCC1
VCC0
ik
c1
c2
c3
ier
ok
tk1
tk2
td1
td2
od
U1A K155TM2
U1B K155TM2
U2A K155TM2
U3 K555IR23
U4 K555IR23
U5A K155LI1
U5B K155LI1
U5C K155LI1
U5D K155LI1
U6A K155LL1
U6B K155LL1
U7A K155LN1
U7B K155LN1
U8A K155LP5
U8B K155LP5
U8C K155LP5
U8D K155LP5
U9A K155LP5BZ
U10A K155TM2
U10B K155TM2
U11 K555IR23
U12 K555IR23
U13 K555IR23
U14A K155LI1
U15A K155TM2
RG
TT
S
D
C
R
EZ
d1
d2
d3
d4
d5
d6
d7
d8
d9
d10
s3
s4
s5
s6
s7
s8
s9
s10
Size
C
Document Number
Rev