Analysis and simulation of a syndrome decoder for a constraint length K=5, rate R=1/2 binary convolutional code

Citation for published version (APA):
Schalkwijk, J. P. M., Vinck, A. J., & Rust, L. J. A. E. (1975). *Analysis and simulation of a syndrome decoder for a constraint length K=5, rate R=1/2 binary convolutional code*. (EUT report. E, Fac. of Electrical Engineering; Vol. 75-E-58). Technische Hogeschool Eindhoven.

Document status and date:
Published: 01/01/1975

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 04. Oct. 2023

[illegible]

Analysis and simulation of a syndrome decoder for a constraint length k=5, rate R=$\frac{1}{2}$ binary convolutional code

by

J. P. M. Schalkwijk,
A. J. Vinck and
L. J. A. E. Rust

TECHNISCHE HOGESCHOOL EINDHOVEN
NEDERLAND
AFDELING DER ELEKTROTECHNIEK
VAKGROEP TELECOMMUNICATIE

EINDHOVEN UNIVERSITY OF TECHNOLOGY
THE NETHERLANDS
DEPARTMENT OF ELECTRICAL ENGINEERING
GROUP TELECOMMUNICATIONS

Analysis and simulation of a syndrome decoder for a constraint length k=5, rate R=½ binary convolutional code

by

J.P.M. Schalkwijk,
A.J. Vinck and
L.J.A.E. Rust

TH-Report 75-E-58
April 1975

ISBN 90 6144 058 0

I. INTRODUCTION.

The principle of syndrome decoding for R=½ convolutional codes will be explained using the binary code generated by the encoder of fig (1).

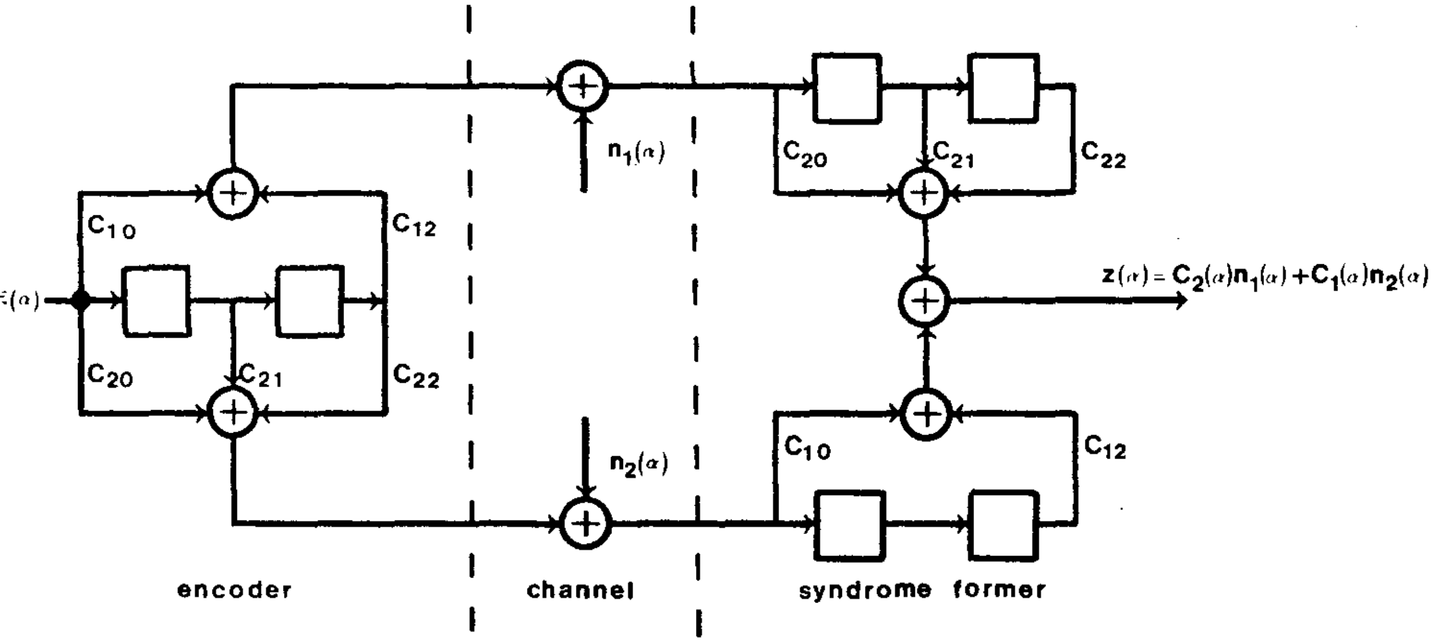


Fig. 1. Encoding and syndrome forming for a R=½ code

The encoder has connection polynomials $C_1(\alpha)$ and $C_2(\alpha)$. Hence, the encoder outputs are $C_1(\alpha)x(\alpha)$ and $C_2(\alpha)x(\alpha)$. The syndrome $z(\alpha)$ only depends on $n_1(\alpha)$ and $n_2(\alpha)$, for

$$\begin{aligned} z(\alpha) &= C_2(\alpha)[C_1(\alpha)x(\alpha) + n_1(\alpha)] + C_1(\alpha)[C_2(\alpha)x(\alpha) + n_2(\alpha)] = \\ &= C_2(\alpha)n_1(\alpha) + C_1(\alpha)n_2(\alpha). \end{aligned}$$

Having formed the syndrome $z(\alpha)$, a recursive algorithm described in [1], determines the noise sequence pair $[\hat{n}_1(\alpha), \hat{n}_2(\alpha)]$ of minimum Hamming weight that can be a possible cause of this syndrome. For a non catastrophic code $C_1(\alpha)$ and $C_2(\alpha)$ are relatively prime. Hence, by Euclids algorithm there exist polynomials $D_1(\alpha)$ and $D_2(\alpha)$ such that

$$D_1(\alpha)C_1(\alpha) + D_2(\alpha)C_2(\alpha) = 1.$$

The estimate $\hat{x}(\alpha)$ of the data sequence $x(\alpha)$ can be written as

$$\hat{x}(\alpha) = [D_1(\alpha)y_1(\alpha) + D_2(\alpha)y_2(\alpha)] + \omega(\alpha)$$

where

$$\omega(\alpha) = D_1(\alpha)\hat{n}_1(\alpha) + D_2(\alpha)\hat{n}_2(\alpha)$$

and

$$y_i(\alpha) = C_i(\alpha)x(\alpha) + n_i(\alpha); \quad i=1,2.$$

The syndrome $z(\alpha)$ can be thought of as generated by the syndrome former of fig. 2.

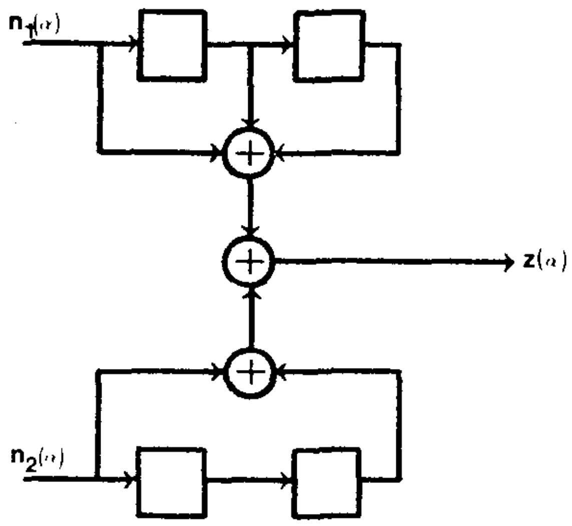


Fig. 2. The syndrome decoder.

If the binary content on time t^- is called the state of the syndrome former, then, after shifting in a bitpair (n_1,n_2) the new state is determined by the content of the ν stages on time t^-, and the incoming bitpair (n_1,n_2). From [1] we have 2^ν classes of states, each containing 2^ν equivalent states. Equivalent states are states which cause the same syndrome digits in response to a given input bitsequencepair. The state in which each of the ν stages of the topregister contain a 0, is taken as the representative for a class of equivalent states.

If we shift in the bitpair (0,0) or (0,1), the new state will be (2 * (old state)) resp. (2 * (old state) + 1), whereas to the bitpair (1,0) or (1,1) we may always add a "0-equivalent" state bit by bit to the content of the registers, without affecting $z(\alpha)$, thus again filling the topregister with a binary zero content. The "0-equivalent" state for the syndrome former of fig. 2 is given in (1,1).

$$\text{So: } = \begin{matrix}1,0\\1,1\end{matrix} \tag{1.1}$$

The D polynomials are

$$\begin{aligned} D_1(\alpha) &= 1+\alpha \\ D_2(\alpha) &= \alpha \end{aligned} \tag{1.2}$$

and the state table is given in table 1.

	bitpair (n_1,n_2)				n_1,n_2				$\hat{n}_1,\hat{n}_2$			
state	00	01	11	10	00	01	11	10	00	01	11	10
	new state				syndrome z				ω			
0	0	1	2	3	0	1	0	1	0	0	1	1
1	2	3	0	1	0	1	0	1	1	1	0	0
2	0	1	2	3	1	0	1	0	0	0	1	1
3	2	3	0	1	1	0	1	0	1	1	0	0

Table 1.

With each state we associate a metric $M_j(k)$, $j=0,1,2,3$; $k=0,1,2,\ldots$, that equals the minimum Hamming weight of a path $[\hat{n}_1(\alpha),\hat{n}_2(\alpha)]^{(\delta)}$, leading from state $j=0$ at time $k=0$ to that particular state.

The metric $M_j(k+1)$ at time $(k+1)$ can be determined recursively, for example, for the syndrome decoder of fig. 2:

$$M_2(k+1) = \bar{z}_k \min\,[M_1(k),M_0(k)+2] + z_k \min\,[M_3(k),M_2(k)+2] \qquad (1.3)$$

Given the value z_k, i.e. $z_k=0$ or $z_k=1$, each $(k+1)$-state can be reached from two k-states, for each of these two k-states add to the metric the Hamming weight of the respective transitions, i.e. of $[\hat{n}_{k1},\hat{n}_{k2}]$, to the particular $(k+1)$-state.

The minimum of the two values is $M_j(k+1)$. In case of a tie we may choose the survivor at random.

For state 1 and 3 we obtain that:

$$M_1(k+1) = \bar{z}_k \min\,[M_2(k)+1,M_3(k)+1] + z_k \min\,[M_0(k)+1,M_1(k)+1] \qquad (1.4)$$

$$M_3(k+1) = \bar{z}_k \min\,[M_2(k)+1,M_3(k)+1] + z_k \min\,[M_0(k)+1,M_1(k)+1] \qquad (1.5)$$

And consequently $M_1(k+1) = M_3(k+1)$. If there is no tie, the path registers for state 1 and state 3 are equal, except for the first stage of the two pathregisters, which are filled with 0 and 1 respectively. In case of a tie we can force equality of the pathregisters, without increasing error probability.

The decoding procedure is given in fig. 3.

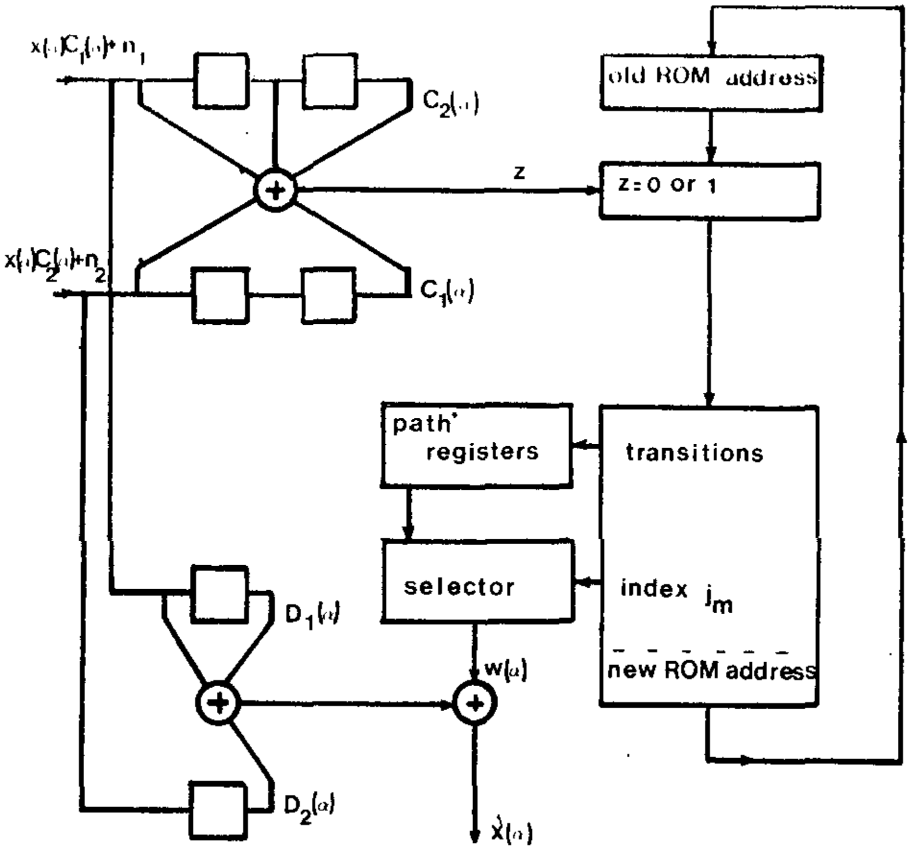


Fig. 3.

Knowledge of the successive survivors for each state, and the state index $j_m(k)$ of the state with minimum metric value suffices to determine the key sequence $\omega(\alpha)$. This information can be stored in a ROM, in which each address corresponds with a certain metric combination.

For both z=0 and z=1, the current ROM address contains information about the survivor, the index j_m, and the new ROM address, respectively. The pathregisters are shuffled and a selector mechanism determines which bit has to be decoded, according to j_m.

The next section describes the differences between Viterbi and syndrome decoding as far as the pathregisters and metric calculations are concerned. It also justifies the selection of a particular k=5 code.

II. Selection of a $\nu=4$ decoder.

First some differences between Viterbi- and syndrome decoding, as far as pathregister organization and metric calculation are concerned, will be explained.

The $R=\frac{1}{2}$ syndrome decoder, has a more complicated pathregister organization, as can be seen from fig. 4 and fig. 5.

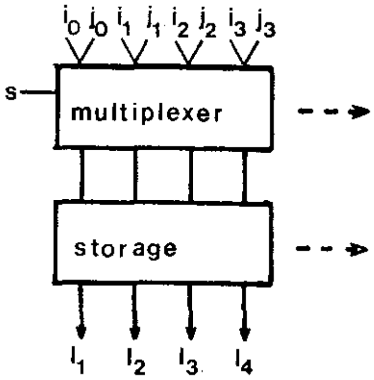


Fig. 4.

Viterbi pathregister organization.

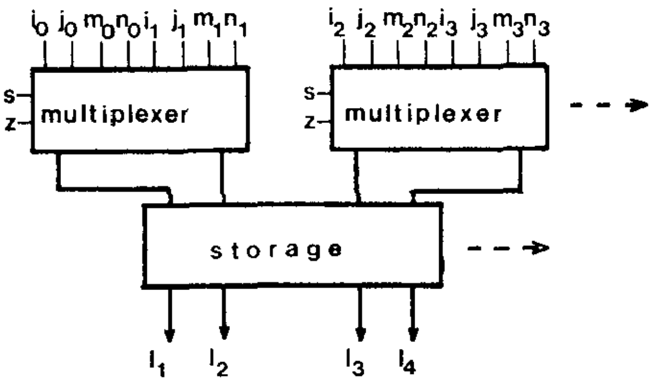


Fig. 5.

Syndrome decoder pathregister organization.

In Viterbi decoding, there are two possible choices for the survivor for each state, whereas for the syndrome decoder there are four alternatives, i.e. two for z=0 and two for z=1. This leads to more complicated pathregisters. However, in many cases the syndrome decoder requires fewer pathregisters than the Viterbi decoder does, thus easily ofsetting the above complication.

An additional advantage of syndrome decoding is a smaller number of metric combinations, and an easier metric calculation.

In Viterbi decoding, we have two incoming states for each state with usually a complementary bitpair.

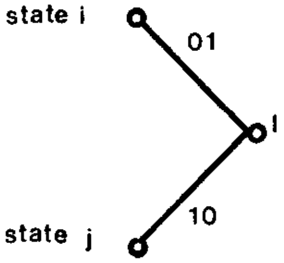


Fig. 6.

After receiving two bits, the new metric for state 1 is

$$M_1(k+1) = \min\left[M_i(k) + d\{(r_1,r_2)-(0,1)\}, M_j(k) + d\{(r_1,r_2)-(1,0)\}\right],$$

(r_1,r_2) being the received data pair.

So we have to determine the distance between (r_1,r_2) and $(0,1)$.

In syndrome decoding, we have after receiving z_k,

$$M_1(k+1) = z_k \min\left[M_i(k)+c_i, M_j(k) + 2-c_i\right] +$$

$$+ \bar{z}_k \min\left[M_m(k) + c_m, M_n(k) + 2-c_m\right].$$

We then determine the minimum of two metrics ofset by some constant. When selecting a decoder one has to look for a code with a small number of distinct metric combinations, that can be realized using few pathregisters. In addition the code must have a good error correcting capability. Therefore a computer program has been developed to determine for the $\nu=4$ codes,

1) the number of bit errors in the paths at d_{min} and $d_{min}+1$, according to [2],
2) the number of distinct metric combinations,
3) the bit error probability for the decoders with PR-length 16 and 11,
4) the minimum possible number of pathregisters.

Three $\nu=4$ codes with minimum distance 7 are listed in table 2.

Code n^0	Connection polynomials	Number of metric combinations	Minimum number of pathregisters	Distance	Number of paths	Total number of bit errors
1	10011	1817	9	7	3	7
	10111			8	3	10
2	11001	1784	12	7	2	4
	11011			8	4	12
3	11001	13641	12	7	2	4
	10111			8	4	12

Table 2.

The state- and metric calculation tables for these three codes are given below.

Code 1.

The "0-equivalent" state is

$$S_0 = \begin{matrix} 1,0,0,0 \\ 1,0,1,1 \end{matrix}$$

The D-polynomials are

$$D_2 = 1 + \alpha + \alpha^2$$

$$D_1 = \alpha + \alpha^2$$

State	new state				z				$\hat{\omega}$			
	n_1,n_2				n_1,n_2				$\hat{n}_1,\hat{n}_2$			
	00	01	11	10	00	01	11	10	00	01	11	10
0	0	1	12	13	0	1	0	1	0	1	1	0
1	2	3	14	15	0	1	0	1	1	0	0	1
2	4	5	8	9	1	0	1	0	1	0	0	1
3	6	7	10	11	1	0	1	0	0	1	1	0
4	8	9	4	5	1	0	1	0	0	1	1	0
5	10	11	6	7	1	0	1	0	1	0	0	1
6	12	13	0	1	0	1	0	1	1	0	0	1
7	14	15	2	3	0	1	0	1	0	1	1	0
8	0	1	12	13	1	0	1	0	0	1	1	0
9	2	3	14	15	1	0	1	0	1	0	0	1
10	4	5	8	9	0	1	0	1	1	0	0	1
11	6	7	10	11	0	1	0	1	0	1	1	0
12	8	9	4	5	0	1	0	1	0	1	1	0
13	10	11	6	7	0	1	0	1	1	0	0	1
14	12	13	0	1	1	0	1	0	1	0	0	1
15	14	15	2	3	1	0	1	0	0	1	1	0

Table 3.

According to state table 3 we can make the metric calculation table for this code.

Metric	$z_k=0$	$z_k=1$
$M_p(k+1)$	min $[M_i(k) + c_i, M_j(k) + 2-c_i]$	min $[M_m(k) + c_m, M_n(k) + 2-c_m]$
M_0	$M_0 , M_6 + 2$	$M_8 , M_{14} + 2$
M_1	$M_8 + 1, M_{14} + 1$	$M_0 + 1, M_6 + 1$
M_2	$M_1 , M_7 + 2$	$M_9 , M_{15} + 2$
M_3	$M_9 + 1, M_{15} + 1$	$M_1 + 1, M_7 + 1$
M_4	$M_{10} , M_{12} + 2$	$M_2 , M_4 + 2$
M_5	$M_2 + 1, M_4 + 1$	$M_{10} + 1, M_{12} + 1$
M_6	$M_{11} , M_{13} + 2$	$M_3 , M_5 + 2$
M_7	$M_3 + 1, M_5 + 1$	$M_{11} + 1, M_{13} + 1$
M_8	$M_{12} , M_{10} + 2$	$M_4 , M_2 + 2$
M_9	$M_2 + 1, M_4 + 1$	$M_{10} + 1, M_{12} + 1$
M_{10}	$M_{13} , M_{11} + 2$	$M_5 , M_3 + 2$
M_{11}	$M_3 + 1, M_5 + 1$	$M_{11} + 1, M_{13} + 1$
M_{12}	$M_6 , M_0 + 2$	$M_{14} , M_8 + 2$
M_{13}	$M_8 + 1, M_{14} + 1$	$M_0 + 1, M_6 + 1$
M_{14}	$M_7 , M_1 + 2$	$M_{15} , M_9 + 2$
M_{15}	$M_9 + 1, M_{15} + 1$	$M_1 + 1, M_7 + 1$

Table 4.

From table 3 and 4 it follows that except for the first bit

PR 1 ≡ PR 13
PR 3 ≡ PR 15
PR 5 ≡ PR 9
PR 7 ≡ PR 11

and except for the second bit,

PR 7 ≡ PR 15
PR 3 ≡ PR 11
PR 10 ≡ PR 2
PR 6 ≡ PR 14

so the first two bits excluded,

PR 3 ≡ PR 15 ≡ PR 7 ≡ PR 11.

From the above we can see that for this code, we need 9 different pathregisters.

If we analyse the transitions to each state, we can deduce the first two bits for each pathregister, as listed in table 5.

pathregisters	first two bits
0,6,11,13	00
3,5,8 ,14	01
1,7,10,12	10
2,4,9 ,15	11

Table 5.

Candidates as survivor for state 0 are the states 0,6,8 and 14. The first two bits of the corresponding pathregisters are given in table 5, hence, no storage elements are needed for these stages. The organization for pathregister 0 is then given in fig. 7.

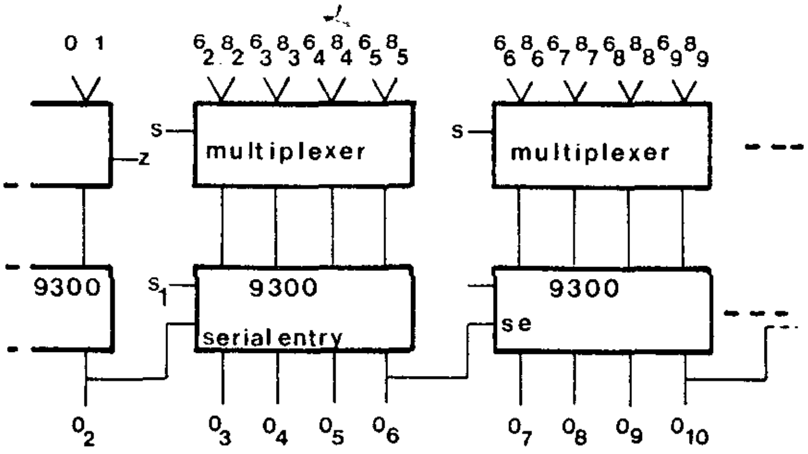


Fig. 7.

Instead of the one out of four multiplexers, we can use here quad-two multiplexers, because PR 6 $\equiv$ PR 14, and if state 0 is selected as a survivor, we can simply shift the contents of pathregister 0 one place to the right. Note that 0_2 is 0 or 1, respectively, according to whether z = 0 or 1.
The same kind of organization is valid of PR 4 and PR 15. So there is no additional hardware needed for these pathregisters.
For PR(1,2,5,6,8,12), we need one out of three multiplexers.
The total amount of hardware, required for the 9 pathregisters, is much less than with classical Viterbi decoding.

Code 2.

The "0-equivalent" state is

$$S_0 = \begin{matrix} 1,0,0,0 \\ 1,1,1,1 \end{matrix}$$

The D-polynomials are

$$D_2 = \alpha^3$$

$$D_1 = 1 + \alpha + \alpha^2 + \alpha^3$$

The state table is given in table 6.

State	new state				z				$\hat{\omega}$			
	n_1,n_2				n_1,n_2				$\hat{n}_1,\hat{n}_2$			
	00	01	11	10	00	01	11	10	00	01	11	10
0	0	1	14	15	0	1	0	1	0	0	1	1
1	2	3	12	13	1	0	1	0	0	0	1	1
2	4	5	10	11	0	1	0	1	0	0	1	1
3	6	7	8	9	1	0	1	0	0	0	1	1
4	8	9	6	7	1	0	1	0	1	1	0	0
5	10	11	4	5	0	1	0	1	1	1	0	0
6	12	13	2	3	1	0	1	0	1	1	0	0
7	14	15	0	1	0	1	0	1	1	1	0	0
8	0	1	14	15	1	0	1	0	0	0	1	1
9	2	3	12	13	0	1	0	1	0	0	1	1
10	4	5	10	11	1	0	1	0	0	0	1	1
11	6	7	8	9	0	1	0	1	0	0	1	1
12	8	9	6	7	0	1	0	1	1	1	0	0
13	10	11	4	5	1	0	1	0	1	1	0	0
14	12	13	2	3	0	1	0	1	1	1	0	0
15	14	15	0	1	1	0	1	0	1	1	0	0

Table 6.

According to state table 6 we can make a metric calculation table.

Metric	z = 0	z = 1
$M_p(k+1)$	min $[M_i(k) + c_i, M_j(k) + 2-c_i]$	min $[M_m(k) + c_m, M_n(k) + 2-c_m]$
M_0	$M_0 \quad , M_7 + 2$	$M_8 \quad , M_{15} + 2$
M_1	$M_8 + 1, M_{15} + 1$	$M_0 + 1, M_7 + 1$
M_2	$M_9 \quad , M_{14} + 2$	$M_1 \quad , M_6 + 2$
M_3	$M_1 + 1, M_6 + 1$	$M_9 + 1, M_{14} + 1$
M_4	$M_2 \quad , M_5 + 2$	$M_{10} \quad , M_{13} + 2$
M_5	$M_{10} + 1, M_{13} + 1$	$M_2 + 1, M_5 + 1$
M_6	$M_{11} \quad , M_{12} + 2$	$M_3 \quad , M_4 + 2$
M_7	$M_3 + 1, M_4 + 1$	$M_{11} + 1, M_{12} + 1$
M_8	$M_{12} \quad , M_{11} + 2$	$M_4 \quad , M_3 + 2$
M_9	$M_4 + 1, M_3 + 1$	$M_{11} + 1, M_{12} + 1$
M_{10}	$M_5 \quad , M_2 + 2$	$M_{13} \quad , M_{10} + 2$
M_{11}	$M_{10} + 1, M_{13} + 1$	$M_2 + 1, M_5 + 1$
M_{12}	$M_{14} \quad , M_9 + 2$	$M_6 \quad , M_1 + 2$
M_{13}	$M_6 + 1, M_1 + 1$	$M_9 + 1, M_{14} + 1$
M_{14}	$M_7 \quad , M_0 + 2$	$M_{15} \quad , M_8 + 2$
M_{15}	$M_8 + 1, M_{15} + 1$	$M_7 + 1, M_0 + 1$

Table 7.

From tables 6 and 7 it follows that except for the first bit

PR(1) ≡ PR(15)

PR(3) ≡ PR(13)

PR(5) ≡ PR(11)

PR(7) ≡ PR(9)

So we need 12 different pathregisters.

It is obvious that the organization for the pathregisters is more complicated than for the first code. The advantage of this code is that there are only 1784 metric combinations.

Code 3.

The "0-equivalent" state is

$$S_0 = \begin{matrix} 1,0,0,0 \\ 1,1,0,0 \end{matrix}$$

The D-polynomials are:

$$D_2 = \alpha$$

$$D_1 = 1 + \alpha$$

The state table is given in table 8.

State	new state				z				$\hat{\omega}$			
	n_1,n_2				n_1,n_2				$\hat{n}_1,\hat{n}_2$			
	00	01	11	10	00	01	11	10	00	01	11	10
0	0	1	2	3	0	1	0	1	0	0	1	1
1	2	3	0	1	0	1	0	1	1	1	0	0
2	4	5	6	7	1	0	1	0	0	0	1	1
3	6	7	4	5	1	0	1	0	1	1	0	0
4	8	9	10	11	1	0	1	0	0	0	1	1
5	10	11	8	9	1	0	1	0	1	1	0	0
6	12	13	14	15	0	1	0	1	0	0	1	1
7	14	15	12	13	0	1	0	1	1	1	0	0
8	0	1	2	3	1	0	1	0	0	0	1	1
9	2	3	0	1	1	0	1	0	1	1	0	0
10	4	5	6	7	0	1	0	1	0	0	1	1
11	6	7	4	5	0	1	0	1	1	1	0	0
12	8	9	10	11	0	1	0	1	0	0	1	1
13	10	11	8	9	0	1	0	1	1	1	0	0
14	12	13	14	15	1	0	1	0	0	0	1	1
15	14	15	12	13	1	0	1	0	1	1	0	0

Table 8.

According to table 8 we can make the metric calculation table.

Metric	z = 0	z = 1
$M_p(k+1)$	$\min\ [M_i(k) + c_i, M_j(k) + 2-c_i]$	$\min\ [M_m(k) + c_m, M_n(k) + 2-c_m]$
M_0	$M_0, M_1 + 2$	$M_8, M_9 + 2$
M_1	$M_8 + 1, M_9 + 1$	$M_0 + 1, M_1 + 1$
M_2	$M_1, M_0 + 2$	$M_9, M_8 + 2$
M_3	$M_8 + 1, M_9 + 1$	$M_0 + 1, M_1 + 1$
M_4	$M_{10}, M_{11} + 2$	$M_2, M_3 + 2$
M_5	$M_2 + 1, M_3 + 1$	$M_{10} + 1, M_{11} + 1$
M_6	$M_{11}, M_{10} + 2$	$M_3, M_2 + 2$
M_7	$M_2 + 1, M_3 + 1$	$M_{10} + 1, M_{11} + 1$
M_8	$M_{12}, M_{13} + 2$	$M_4, M_5 + 2$
M_9	$M_4 + 1, M_5 + 1$	$M_{12} + 1, M_{13} + 1$
M_{10}	$M_{13}, M_{12} + 2$	$M_5, M_4 + 2$
M_{11}	$M_4 + 1, M_5 + 1$	$M_{12} + 1, M_{13} + 1$
M_{12}	$M_6, M_7 + 2$	$M_{14}, M_{15} + 2$
M_{13}	$M_{14} + 1, M_{15} + 1$	$M_6 + 1, M_7 + 1$
M_{14}	$M_7, M_6 + 2$	$M_{15}, M_{14} + 2$
M_{15}	$M_{14} + 1, M_{15} + 1$	$M_6 + 1, M_7 + 1$

Table 9.

From tables 8 and 9 it follows that except for the first bit

PR (1) ≡ PR (3)

PR (5) ≡ PR (7)

PR (9) ≡ PR(11)

PR(13) ≡ PR(15)

So we need 12 different pathregisters.

As for code 1 we can now determine the first three bits for each PR, which is listed below.

PR	first three bits.
0,1	0 0 0
8,9	0 0 1
4 ,5	0 1 0
12,13	0 1 1
6,7	1 1 0
14,15	1 1 1
2,3	1 0 0
10,11	1 0 1

Table 10.

Conclusions.

The code which yields the syndrome decoder with the minimum amount of hardware is code 1, because the number of pathregisters is only 9 and the metric table contains 1817 entries.

This code has a good error correcting capability as can be seen from fig. 8 , and the amount of hardware of the syndrome decoder is significantly smaller than that of the classical Viterbi decoder.

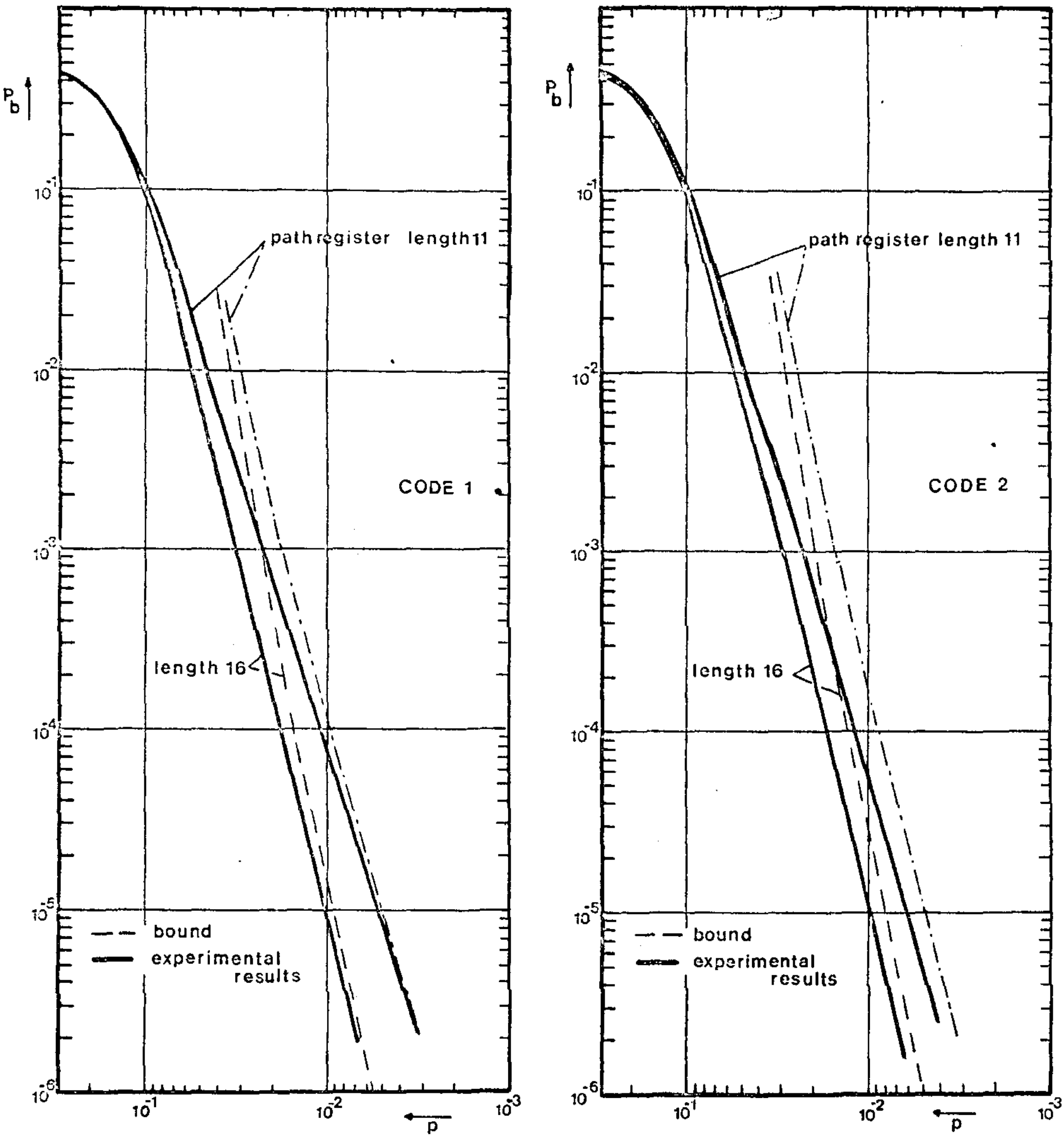


Fig. 8. Bit error rate P_b versus channel transition probability p.

REFERENCES

1. J.P.M.Schalkwijk and A.J.Vinck,"Syndrome decoding of convolutional codes", IEEE Trans. on Communications, to be published.

2. L.v.d.Meeberg, "A tightened upper bound on the error probability of binary convolutional codes with Viterbi decoding",IEEE Trans. Inform. Theory, vol. IT-20,pp. 389-391, May 1974.

Appendix A:computer simulation

```
*
SINSERT MACVIT
*
* THIS IS A COMPUTER SIMULATION FØR A K=5 SYNDROME DECØDER
* ******************************
*     GENERAL CØMMENT
*******************************
* THIS PRØGRAM IS BASED ØN 12 PATHREGISTERS
* TØ HAVE A PRØGRAM WHICH CAN BE USED  FØR THE ØTHER K=5 ENCØDERS
       REL
*
* THE CØNNECTIØN PØLYNØMIALS ARE      10011 AND 10111
*
* THERE ARE 1817 DISTINCT METRIC CØMBINATIØNS
* MINIMUM DISTANCE IS 7
* TØTAL NUMBER ØF BITERRØRS IN THE PATHS ØN DISTANCE 7 AND 8 IS
*        17
* THE '0 EQUIVALENT STATE' IS           1000
*                                       1011
*
* THE D PØLYNØMIALS ARE         D2= 111
*                               D1= 011
* A TABLE HAS BEEN MADE,IN WHICH THE RØWNUMBERS CØRRESPØNDEND WITH
*  A METRICCØMBINATIØN
* ALSØ THE INFØRMATIØN FØR THE PATHREGISTERFILLING AND NEW MINIMUM
*  IS PRESENT IN THIS TABLE
*
*******************************
*
*******************************
START NØP
*WHAT IS THE CHANNEL ERRØR PRØBABILITY
BEG1   NØP
      CALL VINI
      DAC PIET
      DAC RNDM
      DAC INSTW
      DAC EINDW
      DAC HUL
      DAC TØTAL
      DAC 0
* A SUBRØUTINE IS CALLED TØ GIVE THE DESIRED ERRØR PRØBABILITY
*  AND THE NUMBER ØF BITERRØRS THAT HAS TØ BE MADE IN THE DECØDING
*  PRØCEDURE
* THIS IS A STARTING PRØCEDURE TØ HAVE A DECREASING ERRØR PRØBABILITY
* WITH STARTING VALUE PIET AND STEPS INSTW UNTIL THE VALUE
* EINDW IS REACHED
* TØTAL IS THE NUMBER ØF BITERRØRS THAT HAS TØ BE MADE EACH RUN
*  RNDM AND HUL ARE VARIABLES FØR THE RANDØM GENERATØR
* ************************************
```

```
****************************************
* GET THE TABLES FØR THE PATHREGISTERFILLING AND METRIC CØMBINATIØNS
*
      CALL PRØG
      DAC Z0ITE
      CALL PRG1
      DAC Z1ITE
      CALL PRG2
      DAC PVOTE
      CALL PRG3
      DAC PVITE
* IN Z0ITE ARE THE NEW RØWNUMBERS FØR Z=0
* IN Z1ITE ARE THE RØWNUMBERS FØR Z=1
* IN PVOTE IS STØRED THE PATHREGISTER FILLING INFØRMATIØN FØR Z=0
*  THE LAST FØUR BITS ØF EACH WØRD INDICATE THE NEW MINIMUM
*  THE SAME FØR Z=1 IN PVITE
****************************************
****************************************
BEG   NØP
* GET A CHANNEL NØISE BIT
      CALL RNDS
      DAC RNDM
      DAC PIET
      DAC NØISE
      DAC CØNT
      DAC 0
*
* A RANDØM SEQUENCE IS CALLED
* TØ GIVE A NØISY BIT WITH PRØBABILITY PIET
* THE BIT IS STØRED IN NØISE
* A CØNTRØLE IS MADE IF SØMETHING IS WRØNG WITH THE GENERATØR
*  THATS WHY A VARIABLE CØNT IS USED
* RNDM IS A CØNSTANT NEEDED TØ FØRM THE RANDØM SEQUENCE
****************************************
****************************************
* FØRM THE CØRRUPTED INPUT N1
      LDA NØISE
* THE LAST BIT ØF WØRD NØISE IS ØNE ØR ZERØ WITH PRØBABILITY PIET
* SØ IT HAS THE FØRM          000000000000000X
      ALL 5
* NØISE SHIFTED 5 PLACES TØ THE LEFT GIVES
*                              0000000000X00000
* IN THE SYNDRØME FØRMER N1 TAKES THE 5 RIGHT PLACES
*       LIKE                   00000000000XXXXX
* NØW ADD  SHIFTED NØISE BIT TØ HAVE
*                              0000000000XXXXXX
      ADD N1
* SHIFT ØNE PLACE TØ THE RHIGHT TØ HAVE THE CØRRECT CØNTENT
* ØF THE SYNDRØME REGISTER WHICH IS
*                              00000000000XXXXX
      ARL 1
      STA N1
* STØRE N1 FØR THE NEXT TRANSMISSIØN
*
* FØRM THE PRØDUCT N1C2
      ANA C2
      STA N1C2
* SAVE THE RESULT
****************************************
```

```
* GET A CHANNEL NØISE BIT
      CALL RNDS
      DAC RNDM
      DAC PIET
      DAC NØISE
      DAC CØNT
      DAC 0
****************************************
* FØRM THE  INPUT N2
      LDA NØISE
      ALL 5
      ADD N2
* NØW FØRM THE PRØDUCT N2C1
      ARL 1
      STA N2
      ANA C1
* FØRM THE EXCLUSIVE ØR ØF N1C2+N2C1
      ERA N1C2
      STA N1C2
* LØØK IN TABLE PART IF THE SYNDRØME Z=1 ØR 0.
* IN TABLE PART THE PARITY ØF 5 BITS ARE MADE
* BECAUSE THIS IS NØT PØSSIBLE WITH THE CØMPUTER IN AN EASY WAY
*
      LDX N1C2
      LDA PART,1
      STA Z
****************************************
* SYNDRØME Z IS FØRMED
****************************************
* NØW FØRM THE PRØDUCT N2D2
      LDA N2
      ANA D2
      STA N2D2
* NØW FØRM THE PRØDUCT N1D1
      LDA N1
      ANA D1
* FØRM THE EXCLUSIVE ØR ØF N1D1+N2D2
      ERA N2D2
      STA N2D2
* LØØK IN TABLE PART IF W=0 ØR W=1
      LDX N2D2
      LDA PART,1
* SHIFT W IN W1 TØ HAVE A DECØDING DELAY
      IAB
      LDA W1
      IAB
      LRL 1
      IAB
      STA W1
***************************************
* N1D1+N2D2 IS FØRMED
****************************************
```

```
*
* GET NEW METRIC AND PATHREGISTER INFØRMATIØN
*
      LDA Z
      SAS 16
      JMP P0
* INFØRMATIØN FØR Z=1
*
* ØLD RØW NUMBER IS STØRED IN BMET
* LØAD X REG WITH BMET
      LDX BMET
* LØØK IN TABLE Z11TE ØN RØW BMET TØ HAVE NEW RØWNUMBER
      LDA Z11TE,1
      STA NWEMET
* IN NWEMET IS NØW THE NEW RØWNUMBER
* LØØK IN TABLE PV1TE TØ HAVE THE CØRRESPØNDING
* PATHREGISTER INFØRMATIØN
* STØRE THIS IN PADRV
      LDA PV1TE,1
      STA PADRV
      JMP P1
P0    LDX BMET
* SAME CØMMENT AS FØR Z=1
*
* INFØRMATIØN FØR Z=0
*
      LDA Z01TE,1
      STA NWEMET
      LDA PV0TE,1
      STA PADRV
P1    LDA NWEMET
      STA BMET
*
* IN BMET IS NØW THE NEW RØWNUMBER FØR THE NEXT TRANSMISSIØN
*
****************************************
* NEW RØWNUMBER(METRICCØMBINATIØN)AND PATH REGISTER FILLING INFØRMATIØ

* IS READY
*****************************************
```

```
****************************************
* REFILL PATH REGISTERS
****************************************
      LDA Z
      SAS 16
      JMP ZISØ
*
* THE PATH REGISTERS MUST BE REFILLED FØR Z = 1
* THE INFØRMATIØN IS STØRED IN PADRV
* FØR INSTANCE PAD0 IS FILLED WITH PATH 8 ØR PATH 14
* THIS DEPENDS ØN THE FIRST BIT IN PADRV
*  JUMP + 2 MEENS THAT  THE FIRST BIT ØF PAD0
* IS 0, JUMP +1 MEENS THAT THIS BIT IS A 1
* THE  CØNTENT ØF PAD0 IS TEMPØRARELY STØRED IN PADH0
* BECAUSE THE ØLD INFØRMATIØN ØF PAD0 IS NEEDED IN THE
* FØLLØWING PATHREGISTER FILLINGS
*
*
* THE SAME GØES FØR THE ØTHER PATHREGISTERS
      PRV AFTER SKIP ØN BIT 1 THE CØNTENT ;
      ØF PADH0 IS PAD8 ØR PAD14 AND JUMP +2
*
*
      PRV AFTER SKIP ØN BIT 2 THE CØNTENT ;
      ØF PADH1 IS PAD0 ØR PAD6 AND JUMP +1
*
      SSP
      STA PADH13
*
* PATH 1 IS EQUAL TØ PATH 13 EXCEPT FØR THE FIRST BIT WHICH IS A C
*
*
      PRV AFTER SKIP ØN BIT 3 THE CØNTENT ;
      ØF PADH2 IS PAD9 ØR PAD15 AND JUMP +1
*
*
      PRV AFTER SKIP ØN BIT 4 THE CØNTENT ;
      ØF PADH3 IS PAD1 ØR PAD7 AND JUMP +2
*
      SSM
      STA PADH15
*
```

```
*
      PRV AFTER SKIP ØN BIT 5 THE CØNTENT ;
      ØF PADH4 IS PAD2 ØR PAD4 AND JUMP +1
*
*
      PRV AFTER SKIP ØN BIT 6 THE CØNTENT ;
      ØF PADH5 IS PAD10 ØR PAD12 AND JUMP +2
*
      SSM
      STA PADH9
*
*
      PRV AFTER SKIP ØN BIT 7 THE CØNTENT ;
      ØF PADH6 IS PAD3 ØR PAD5 AND JUMP +2
*
*
      PRV AFTER SKIP ØN BIT 8 THE CØNTENT ;
      ØF PADH7 IS PAD11 ØR PAD13 AND JUMP +1
*
      SSP
      STA PADH11
*
*
      PRV AFTER SKIP ØN BIT 9 THE CØNTENT ;
      ØF PADH8 IS PAD4 ØR PAD2 AND JUMP +2
*
*
      PRV AFTER SKIP ØN BIT 10 THE CØNTENT ;
      ØF PADH10 IS PAD5 ØR PAD3 AND JUMP +1
*
*
      PRV AFTER SKIP ØN BIT 11 THE CØNTENT ;
      ØF PADH12 IS PAD14 ØR PAD8 AND JUMP +1
*
*
      PRV AFTER SKIP ØN BIT 12 THE CØNTENT ;
      ØF PADH14 IS PAD15 ØR PAD9 AND JUMP +2
*
*
      JMP HAN
*
* THE PATHREGISTER FILLING FØR Z=1 IS DØNE
*
```

```
*
*
* Z=0 FILL THE PATH REGESTERS
*
ZISØ   NØP
**************************************
         PRV AFTER SKIP ØN BIT 1 THE CØNTENT ;
        ØF PADH0 IS PAD0 ØR PAD6 AND JUMP +2
*
*
        PRV AFTER SKIP ØN BIT 2 THE CØNTENT ;
        ØF PADH1 IS PAD8 ØR PAD14 AND JUMP +1
*
        SSP
        STA PADH13
*
*
        PRV AFTER SKIP ØN BIT 3 THE CØNTENT ;
        ØF PADH2 IS PAD1 ØR PAD7 AND JUMP +1
*
*
        PRV AFTER SKIP ØN BIT 4 THE CØNTENT ;
        ØF PADH3 IS PAD9 ØR PAD15 AND JUMP +2
*
        SSM
        STA PADH15
*
*
        PRV AFTER SKIP ØN BIT 5 THE CØNTENT ;
        ØF PADH4 IS PAD10 ØR PAD12 AND JUMP +1
*
*
        PRV AFTER SKIP ØN BIT 6 THE CØNTENT ;
        ØF PADH5 IS PAD2 ØR PAD4 AND JUMP +2
*
        SSM
        STA PADH9
*
*
        PRV AFTER SKIP ØN BIT 7 THE CØNTENT ;
        ØF PADH6 IS PAD11 ØR PAD13 AND JUMP +2
*
*
        PRV AFTER SKIP ØN BIT 8 THE CØNTENT ;
        ØF PADH7 IS PAD3 ØR PAD5 AND JUMP +1
*
        SSP
        STA PADH11
*
*
        PRV AFTER SKIP ØN BIT 9 THE CØNTENT ;
        ØF PADH8 IS PAD12 ØR PAD10 AND JUMP +2
*
*
        PRV AFTER SKIP ØN BIT 10 THE CØNTENT ;
        ØF PADH10 IS PAD13 ØR PAD11 AND JUMP +1
*
*
        PRV AFTER SKIP ØN BIT 11 THE CØNTENT ;
        ØF PADH12 IS PAD6 ØR PAD0 AND JUMP +1
*
*
        PRV AFTER SKIP ØN BIT 12 THE CØNTENT ;
        ØF PADH14 IS PAD7 ØR PAD1 AND JUMP +2
*
```

```
* FILLING ØF ØRIGINAL PATHREGISTERS
* BECAUSE THE PATH REGISTERS ARE STØRED IN HELP PATHS
* WE NØW HAVE TØ CHANGE THIS TØ MAKE THE PATHREGISTERS READY FØR
*
* THE NEXT TRANSMISSIØN
HAN   IMA PADH0
      IMA PAD0
      IMA PADH1
      IMA PAD1
      IMA PADH2
      IMA PAD2
      IMA PADH3
      IMA PAD3
      IMA PADH4
      IMA PAD4
      IMA PADH5
      IMA PAD5
      IMA PADH6
      IMA PAD6
      IMA PADH7
      IMA PAD7
      IMA PADH8
      IMA PAD8
      IMA PADH9
      IMA PAD9
      IMA PADH10
      IMA PAD10
      IMA PADH11
      IMA PAD11
      IMA PADH12
      IMA PAD12
      IMA PADH13
      IMA PAD13
      IMA PADH14
      IMA PAD14
      IMA PADH15
      IMA PAD15
****************************************
* PATHS ARE REFILLED
****************************************
****************************************
* LØØK FØR BITERRØR
****************************************
* LØCATE MINIMUM,STØRED IN PADRV IN LAST 4 BITS
      LDA PADRV
      ANA MASK
      AØA
      STA HULP
      LDX HULP
*
* HULP PLACES BEHIND PADA THE PATH WHICH HAS TØ BE THE MINIMUM
* IS STØRED
*
      LDA PADA,1
* CØMPARE WITH ØRIGINAL W1 FØR CØNTRØLE ØN BITERRØR
      ERA W1
      ANA LTSTE
      SAS E1
      JMP END
* A BITERRØR IS MADE SØ INCREASE ABITF
      LDA ABITF
      AØA
      STA ABITF
****************************************
* DECØDING IS DØNE
****************************************
```

```
************************************************
* ØRGANIZATIØN PART FØR NUMBER ØF TRANSMISSIØNS
* AND PØSSIBLY NEW CHANNEL ERRØR PRØBABILITY
************************************************
      CAS TØTAL
*
* IF THE TØTAL NUMBER ØF BITERRØRS IS EQUAL TØ THE NUMBER IN TØTAL
* WE HAVE TØ START A NEW VALUE FØR THE CHANNEL ERRØRPRØBABILITY
* ELSE GØ ØN WITH THE PRESENT CHANNEL ERRØR PRØBABILITY
* FØR A NEW  CHANNEL ERRØR PRØBABILITY JMP TØ NWESE
*
      JMP NWESE
      JMP NWESE
      JMP GGD
END   NØP
GGD   LDA CØNT
*
* A CØNTRØLE IS MADE IF SØMETHING IS WRØNG WITH THE RANDØM GENERATØR
*IF SØ, WE START A NEW CHANNEL ERRØR PRØBABILITY
*
      SAS 16
      JMP IRSAT
      JMP NWESE
IRSAT  LDA ATRSM
      AØA
      STA ATRSM
      CAS TIEND
      JMP HØGØP
      JMP HØGØP
* IF THE NUMBER ØF TRANSMISSIØNS IS GREATER THAN TIEND WE INCREMENT TIE
                                        D
* AND MAKE ATRSM ZERØ
* IN TIND IS NØW THE NUMBER ØF TRANSMISSIØNS TIMES TIEND
* THE REST IS IN ATRSM
*
      JMP BEG
HØGØP  IRS TIND
      CRA
      STA ATRSM
      JMP BEG
NWESE  NØP
      CALL VJI
      DAC PIET
      DAC TIND
      DAC ATRSM
      DAC ABITF
      DAC 0
* A SUBRØUTINE IS CALLED TØ GET THE ØUTPUT
* THE ØUTPUT IS 1)THE CHANNEL ERRØR PRØBABILITY
*               2)THE TØTAL NUMBER ØF TRANSMISSØNS
*               3)THE NUMBER ØF BITERRØRS TØ BE MADE
*
```

```
*
*
* A  SUBRØUTINE  IS CALLED TØ MAKE A NEW CHANNEL
* ERRØR PRØBABILITY
* A CØNTRØLE IS MADE IF THE VALUE UNTIL WHICH THE PRØBABILITY HAS TØ GØ
* IS REACHED
*
      CALL NUEW
      DAC PIET
      DAC EINDW
      DAC INSTV
      DAC RUDM
      DAC HUL
      DAC CØNTR
      DAC 0
      CRA
* RESET THE VALUES ØF ATRSM,TIND,ARITF
*
* MAKE THE CØNTRØLE FØR THE LAST PRØBABILITY
*
      STA ATRSM
      STA TIND
      STA ABITF
      LDA CØNTR
      CAS HUL
      JMP LEØ
      JMP BEG
      JMP BEG
      NØP
      JMP LEØ
****************************************
* DATA DEFINITIØNS ,END ØF PRØGRAM
****************************************
CØUNT DATA 0
N1    DATA 0
N2    DATA 0
C2    DATA '23
N1C2  DATA 0
MIN   DATA 0
C1    DATA '27
Z     DATA 0
D2    DATA '34
D1    DATA '14
N2D2  DATA 0
W1    DATA 0
BMET  DATA 1
HULP   DATA 0
TØTAL  DATA 0
TIEND  DATA 10000
TIND   DATA 0
EINDW  DATA 1.EE-3
INSTV  DATA 1.EE-3
HUL    DATA 1.EE-3
CØNTR  DATA 0
NWEMET DATA 0
PADRV  DATA 0
NØISE DATA 0
HUP   DATA -30000
BAS   DATA -30000
PART   DATA 0
*
```

```
*
          DATA '1
          DATA '1
          DATA '0
          DATA '1
          DATA '0
          DATA '0
          DATA '1
          DATA '1
          DATA '0
          DATA '0
          DATA '1
          DATA '0
          DATA '1
          DATA '1
          DATA '0
          DATA '1
          DATA '0
          DATA '0
          DATA '1
          DATA '0
          DATA '1
          DATA '1
          DATA '0
          DATA '0
          DATA '1
          DATA '1
          DATA '0
          DATA '1
          DATA '0
          DATA '0
          DATA '1
PADH0     DATA 0
PADH1     DATA 0
PADH2     DATA 0
PADH3     DATA 0
PADH4     DATA 0
PADH5     DATA 0
PADH6     DATA 0
PADH7     DATA 0
PADH8     DATA 0
PADH9     DATA 0
PADH10    DATA 0
PADH11    DATA 0
PADH12    DATA 0
PADH13    DATA 0
PADH14    DATA 0
PADH15    DATA 0
PADA      DATA 0
PAD0      DATA 0
PAD1      DATA 0
PAD2      DATA 0
```

```
*
PAD3    DATA 0
PAD4    DATA 0
PAD5   DATA 0
PAD6   DATA 0
PAD7   DATA 0
PAD8   DATA 0
PAD10   DATA 0
PAD12   DATA 0
PAD14   DATA 0
PAD9    DATA 0
PAD11   DATA 0
PAD13   DATA 0
PAD15   DATA 0
MASK    DATA '17
LTSTE   DATA '40
ZI      DATA 0
HLP1    DATA 0
NUL    DATA 0
CØNT   DATA 0
RNDM   DATA 5.EE-1
PIET   DATA 1.EE-3
TØT    DATA 0
X      DATA 0.0
ALPHA DATA 0
MASK1 DATA 0
ABITF  DATA 0
ATRSH  DATA -0
METTAR DATA 0
Z0ITE  BSZ 1325
Z1ITE   BSZ 1825
PVOTE   BSZ 1325
PV1TE   BSZ 1825
LEØ    CALL EXIT
       END START
BØTTØM
```

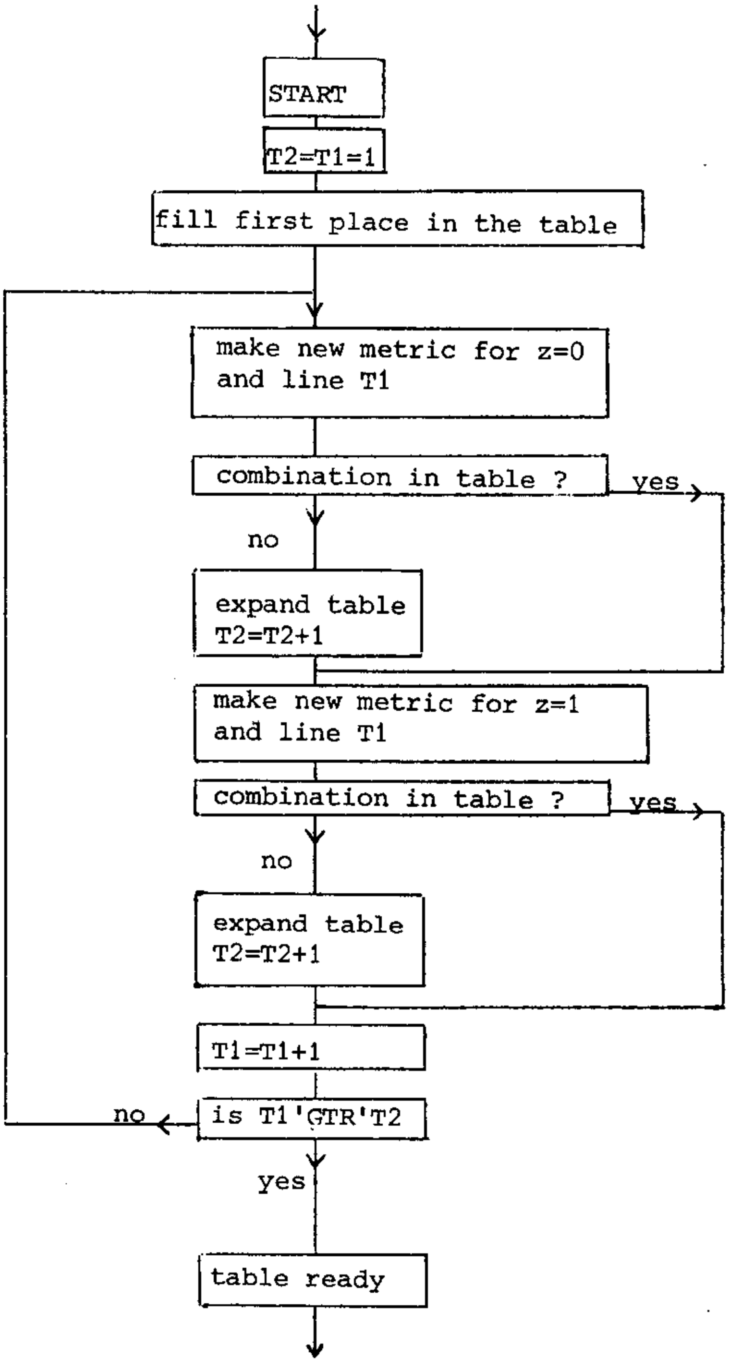
```
THIS IS A MACRO FOR THE PATHREGISTER FILLING
*PADHA IS FILLED WITH PADX OR PADY
*THE FIRST BIT IS SET OR RESET,WHICH DEPENDS ON THE
* VARIABLE GAMMA.
```

```
*
*
**********************************
*                                *
*    MACRO VITL:  MACVIT         *
*                                *
**********************************
*
*  MACRO  NAME  :  PRV  (PAD REGISTER VOLLING)
*  MACRO  CALL  :  PRV ABTER SKIP ON BIT BETA THE CONTENT
*                  OF PADHA IS PADX OR PADY AND JUMP GAMMA.
*  PARAMETERS   :  BETA, PADHA, PADX, PADY, GAMMA.
*  DUMMY WORDS  :  AFTER, SKIP, ON, THE, CONTENT, AND.
*  ARG IDNTIFIER:  BIT, OF, IS, OR, JUMP.
*
*
*
PRV MAC (BIT)=1, (OF)=5, (IS)=2, (OR)=3, ;
        (JUMP)=4, AFTER, SKIP, ON, THE, ;
        CONTENT, AND
      NLSM
      LDA  PADRV
      SAS  <1>
      JMP  &AA
      LDA  <2>
      JMP  &AB
&AA   LDA  <3>
&AB   LRL  1
      JMP  *<4>
      SSM
      STA  <5>
      ENDM
```

```
* THIS SIMULATION IS DONE WITH A PRIME-300 MINI COMPUTER
```

Appendix B: Generation of syndrome metric table

Block scheme for the generation of the syndrome table.



```
        BURROUGHS B6700 BEATHE COMPILER, VERSION 2.6.000,  THURSDAY, 04/(


                                   E / E C A / S Y N D R O M E T A B L E
                                   = = = = = = = = = = = = = = = = = = =


'BEGIN''FILE'IN(KIND=READER),OUT:
'INTEGER' AA,BB,C,J,V,T,X,K,L;
'INTEGER'T2,T1;
'REAL' P,PP;
'INTEGER' G,H,J,M,F;
'INTEGER'I;
'REAL''ARRAY'                              O[1:199  ];
'INTEGER''ARRAY'S[0:11],D[0:11]                 ,AB[0:15];
'INTEGER''ARRAY'                             A[0:11],BA[0:15];
'INTEGER'SS;
'INTEGER''PROCEDURE'METRIC0(S,D,AB);
 'INTEGER''ARRAY'S[0],D[0],AB[0]:
'BEGIN''INTEGER'AA,BB,C,CC,J,V,FF;
       'INTEGER''PROCEDURE'M(AA,BB,FF);
'VALUE'AA,BB;'INTEGER'AA,BB,FF;
'BEGIN''INTEGER'P;
P:=AA:
'IF'BB<P'THEN'P:=BB;
M:=P;
'END';
FF:=1:
S[0]:=M(D[0],D[1]+2,FF);
S[1]:=M(D[6]+1,D[7 ]+1,FF);
S[2]:=M(D[1],D[0]+2,FF);
S[3]:=M(D[8],D[7]+2,FF);
S[4]:=M(D[1 ]+1,D[2]+1,FF);
S[5]:=M(D[7],D[8]+2,FF);
S[6 ]:=M(D[9]   ,D[10]+2,FF);
S[7 ]:=M(D[3 ]+1,D[4]+1,FF);
S[8 ]:=M(D[10]   ,D[9]+2,FF);
S[9 ]:=M(D[5 ]   ,D[4]+2,FF);
S[10]:=M(D[10]+1,D[11]+1,FF);
S[11]:=M(D[4 ]   ,D[5 ]+2,FF);
C:=S[0]:'FOR'U:=1'STEP'1'UNTIL'11'DO''BEGIN'
'IF'S[U]'LSS' C'THEN'            C:=S[U]:
'END':
'FOR' V:=0'STEP'1'UNTIL'11'DO'S[V]:=S[V]-C;
'END';
'INTEGER''PROCEDURE'METRIC1(S,D,BA);
 'INTEGER''ARRAY'S[0],D[0],BA[0]:
'BEGIN''INTEGER'AA,BB,C,J,V,FF,CC:
       'INTEGER''PROCEDURE'M(AA,BB,FF);
'VALUE'AA,BB;'INTEGER'AA,BB,FF;
'BEGIN''INTEGER'P;
P:=AA;
'IF'BB<P'THEN'P:=BB;
M:=P;
'END';
FF:=1;
S[0 ]:=M(D[6 ]   ,D[7 ]+2,FF):
S[1 ]:=M(D[0 ]+1,D[1 ]+1,FF);
```

```
S[3 ]:=M(D[2 ]   ,D[1 ]+2,FF);
S[4 ]:=M(D[7 ]+1,D[8 ]+1,FF);
S[5 ]:=M(D[1 ]   ,D[2 ]+2,FF);
S[6 ]:=M(D[3 ]   ,D[4 ]+2,FF);
S[7 ]:=M(D[9 ]+1,D[10]+1,FF);
S[8 ]:=M(D[4 ]   ,D[3 ]+2,FF);
S[9 ]:=M(D[11]   ,D[10]+2,FF);
S[10]:=M(D[4 ]+1,D[5 ]+1,FF);
S[11]:=M(D[10]   ,D[11]+2,FF);
C:=S[0]:'FOR'U:=1'STEP'1'UNTIL'11'DO''BEGIN'
'IF'S[U]'LSS' C'THEN'          C:=S[U];
'END';
'FOR' V:=0'STEP'1'UNTIL'11'DO'S[V]:=S[V]-C;
'END';
'REAL   ''PROCEDURE' OPZOEKEN(O,T,P);
'VALUE'T,P;
'REAL   ''ARRAY'O[1];'INTEGER'T   ;
'REAL'P;
'BEGIN''INTEGER' AAA,BBB,CCC;
BBB:=T;
'FOR'AAA:=1'STEP'1'UNTIL' BBB'DO'
'BEGIN''IF'O[AAA]=P'THEN''BEGIN'BBB:=0;CCC:=1'END'
'ELSE'CCC:=2;
'END';
OPZOEKEN:=CCC;
'END';
'REAL''PROCEDURE'GAR(A,P);
'VALUE'P;'REAL'P;'INTEGER''ARRAY'A[0];
'BEGIN''INTEGER'I,J;
      'FOR'I:=1'STEP'1'UNTIL'12
        'DO' A[I-1]:=P.[4*I-1:4];
  'END';
'REAL''PROCEDURE' ARP(PP,A);
'INTEGER''ARRAY'A[0];
'REAL' PP;
'BEGIN''INTEGER'T;PP:=0;
     'FOR'T:=1'STEP'1'UNTIL'12'DO'
      PP:=PP&A[T-1][4*T-1:4];
  'END';
'FOR'T1:=0'STEP'1'UNTIL'11'DO' READ(IN,/,A[T1]);
ARP(O[1],A);
T1:=T2:=1;
WRITE(OUT,<///,
"OLD-METRICCOMB----NEW-COMB-FOR-Z=0----NEW-COMB-FOR-Z=1">);
'FOR'T1:=1'STEP'1'UNTIL'T2'DO'
'BEGIN'
P:=O[T1];
GAR(A,P);
WRITE(OUT,</,"-">);
'FOR'K:=0'STEP'1'UNTIL' 11'DO' WRITE(OUT,<I1>,A[K]);
WRITE(OUT,<  "---------">);
    METRICO(S,A,AB);
'FOR'K:=0'STEP'1'UNTIL' 11'DO' WRITE(OUT,<I1>,S[K]);
WRITE(OUT,<  "---------">);
ARP(PP,S);
'IF'OPZOEKEN(O,T2,PP)=1
'THEN'
'ELSE''BEGIN'T2:=T2+1;O[T2]:=PP;'END';
```

```
METRIC1(S,A,BA);
'FOR'K:=0'STEP'1'UNTIL' 11'DO' WRITE(OUT,<I1>,S[K]);
ARP(PP,S);
'IF'OPZOEKEN(O,T2,PP)=1
'THEN'
'ELSE''BEGIN'T2:=T2+1;O[T2]:=PP;'END';
'IF'T2>19  7'THEN'T2:=0;
'END';
'END'.
```

first lines of table for code 3

```
OLD-METRICCOMB----NEW-COMB-FOR-Z=0----NEW-COMB-FOR-Z=1
-052434444443--------052434444443--------303143342233
-303143342233--------240214223344--------314230133343
-240214223344--------123201213240--------232032241424
-314230133343--------321323334042--------012230132203
-123201213240--------122331214110--------221322230012
-232032241424--------122022301122--------234223032432
-321323334042--------342423032332--------222031201123
-012230132203--------021223230012--------113231213210
-122331214110--------011220030002--------221222323021
-221322230012--------232022031222--------122001201120
-122022301122--------112130112232--------220212022232
-234223032432--------102132322221--------032433242233
-342423032332--------203122222221--------042234442233
-222031201123--------212130112133--------230212022322
-021223230012--------032022031322--------102001101120
-113231213210--------010210120002--------221321223021
-011220030002--------011022030012--------012111212210
-221222323021--------232322032222--------221021101021
-232022031222--------102022101121--------032223032232
-122001201120--------112130112110--------220212020012
-112130112232--------121221223032--------010110022102
-220212022232--------101101112120--------032032231223
-102132322221--------130212222223--------312230133132
-032433242233--------032234243343--------103122322232
-203122222221--------230212222222--------212230132132
-042234442233--------052234233443--------303123122232
-212130112133--------221221123042--------010110012202
-230212022322--------102101211120--------032032231222
-032022031322--------012133312232--------012223032232
-102001101120--------110110112110--------110210020012
-010210120002--------021012020011--------112011211210
-221321223021--------232322032122--------121021201011
-011022030012--------011022011222--------012111012231
-012111212210--------021221221111--------211221121021
-232322032222--------102122121121--------032233332232
-221021101021--------212120012122--------130112012122
-102022101121--------110110112222--------110210022132
-032223032232--------012233233332--------012233232233
-112130112110--------121221121012--------121221123011
-220212020012--------212012021221--------032012211221
-121221223032--------132322032132--------111021101112
-010110022102--------011212120011--------012031111210
-101101112120--------120211112110--------111120120012
-032032231223--------032133212233--------213223032332
-130212222223--------022101111120--------222032231322
-312230133132--------321323133032--------012130112102
-032234243343--------032334334443--------103132132233
-103122322232--------130212223232--------312230132233
-230212222222--------122101111120--------232032231222
-212230132132--------221223133032--------012120112102
-052234233443--------032333434443--------102133142233
```

EINDHOVEN UNIVERSITY OF TECHNOLOGY
THE NETHERLANDS
DEPARTMENT OF ELECTRICAL ENGINEERING

Reports:

1) Dijk, J., M. Jeuken and E.J. Maanders
AN ANTENNA FOR A SATELLITE COMMUNICATION GROUND STATION (PROVISIONAL ELECTRICAL DESIGN). TH-report 68-E-01. March 1968. ISBN 90 6144 001 7

2) Veefkind, A., J.H. Blom and L.Th. Rietjens
THEORETICAL AND EXPERIMENTAL INVESTIGATION OF A NON-EQUILIBRIUM PLASMA IN A MHD CHANNEL. TH-report 68-E-2. March 1968. Submitted to the Symposium on a Magnetohydrodynamic Electrical Power Generation, Warsaw, Poland, 24-30 July, 1968. ISBN 90 6144 002 5

3) Boom, A.J.W. van den and J.H.A.M. Melis
A COMPARISON OF SOME PROCESS PARAMETER ESTIMATING SCHEMES. TH-report 68-E-03. September 1968. ISBN 90 6144 003 3

4) Eykhoff, P., P.J.M. Ophey, J. Severs and J.O.M. Oome
AN ELECTROLYTIC TANK FOR INSTRUCTIONAL PURPOSES REPRESENTING THE COMPLEX-FREQUENCY PLANE. TH-report 68-E-04. September 1968. ISBN 90 6144 004 1

5) Vermij, L. and J.E. Daalder
ENERGY BALANCE OF FUSING SILVER WIRES SURROUNDED BY AIR. TH-report 68-E-05. November 1968. ISBN 90 6144 005 X

6) Houben, J.W.M.A. and P. Massee
MHD POWER CONVERSION EMPLOYING LIQUID METALS. TH-report 69-E-06. February 1969. ISBN 90 6144 006 8

7) Heuvel, W.M.C. van den and W.F.J. Kersten
VOLTAGE MEASUREMENT IN CURRENT ZERO INVESTIGATIONS. TH-report 69-E-07. September 1969. ISBN 90 6144 007 6

8) Vermij, L.
SELECTED BIBLIOGRAPHY OF FUSES. TH-report 69-E-08. September 1969. ISBN 90 6144 008 4

9) Westenberg, J.Z.
SOME IDENTIFICATION SCHEMES FOR NON-LINEAR NOISY PROCESSES. TH-report 69-E-09. December 1969. ISBN 90 6144 009 2

10) Koop, H.E.M., J. Dijk and E.J. Maanders
ON CONICAL HORN ANTENNAS. TH-report 70-E-10. February 1970. ISBN 90 6144 010 6

11) Veefkind, A.
NON-EQUILIBRIUM PHENOMENA IN A DISC-SHAPED MAGNETOHYDRODYNAMIC GENERATOR. TH-report 70-E-11. March 1970. ISBN 90 6144 011 4

12) Jansen, J.K.M., M.E.J. Jeuken and C.W. Lambrechtse
THE SCALAR FEED. TH-report 70-E-12. December 1969. ISBN 90 6144 012 2

13) Teuling, D.J.A.
ELECTRONIC IMAGE MOTION COMPENSATION IN A PORTABLE TELEVISION CAMERA. TH-report 70-E-13. 1970. ISBN 90 6144 013 0

14) Lorencin, M.
AUTOMATIC METEOR REFLECTIONS RECORDING EQUIPMENT. TH-report 70-E-14. November 1970. ISBN 90 6144 014 9

15) Smets, A.J.
THE INSTRUMENTAL VARIABLE METHOD AND RELATED IDENTIFICATION SCHEMES. TH-report 70-E-15. November 1970. ISBN 90 6144 015 7

16) White, Jr., R.C.
A SURVEY OF RANDOM METHODS FOR PARAMETER OPTIMIZATION. TH-report 70-E-16. February 1971. ISBN 90 6144 016 5

17) Talmon, J.L.
APPROXIMATED GAUSS-MARKOV ESTIMATIONS AND RELATED SCHEMES. TH-report 71-E-17. February 1971. ISBN 90 6144 017 3

18) Kalásek, V.
MEASUREMENT OF TIME CONSTANTS ON CASCADE D.C. ARC IN NITROGEN. TH-report 71-E-18. February 1971. ISBN 90 6144 018 1

19) Hosselet, L.M.L.F.
OZONBILDUNG MITTELS ELEKTRISCHER ENTLADUNGEN. TH-report 71-E-19. March 1971. ISBN 90 6144 019 X

20) Arts, M.G.J.
ON THE INSTANTANEOUS MEASUREMENT OF BLOODFLOW BY ULTRASONIC MEANS. TH-report 71-E-20. May 1971. ISBN 90 6144 020 3

21) Roer, Th.G. van de
NON-ISO THERMAL ANALYSIS OF CARRIER WAVES IN A SEMICONDUCTOR. TH-report 71-E-21. August 1971. ISBN 90 6144 021 1

22) Jeuken, P.J., C. Huber and C.E. Mulders
SENSING INERTIAL ROTATION WITH TUNING FORKS. TH-report 71-E-22. September 1971. ISBN 90 6144 022 X

23) Dijk, J. and E.J. Maanders
APERTURE BLOCKING IN CASSEGRAIN ANTENNA SYSTEMS. A REVIEW. TH-report 71-E-23. September 1971. ISBN 90 6144 023 8

24) Kregting, J. and R.C. White, Jr.
ADAPTIVE RANDOM SEARCH. TH-report 71-E-24. October 1971. ISBN 90 6144 024 6

25) Damen, A.A.H. and H.A.L. Piceni
THE MULTIPLE DIPOLE MODEL OF THE VENTRICULAR DEPOLARISATION. TH-report 71-E-25. October 1971. ISBN 90 6144 025 4 (In preparation).

26) Bremmer, H.
A MATHEMATICAL THEORY CONNECTING SCATTERING AND DIFFRACTION PHENOMENA, INCLUDING BRAGG-TYPE INTERFERENCES. TH-report 71-E-26. December 1971. ISBN 90 6144 026 2

27) Bokhoven, W.M.G. van
METHODS AND ASPECTS OF ACTIVE-RC FILTERS SYNTHESIS. TH-report 71-E-27. 10 December 1970. ISBN 90 6144 027 0

28) Boeschoten, F.
TWO FLUIDS MODEL REEXAMINED. TH-report 72-E-28. March 1972. ISBN 90 6144 028 9

29) REPORT ON THE CLOSED CYCLE MHD SPECIALIST MEETING. Working group of the joint ENEA/IAEA international MHD liaison group. Eindhoven, The Netherlands, September 20-22, 1971. Edited by L.H.Th. Rietjens.
TH-report 72-E-29. April 1972. ISBN 90 6144 029 7

30) Kessel, C.G.M. van and J.W.M.A. Houben
LOSS MECHANISMS IN AN MHD GENERATOR. TH-report 72-E-30. June 1972. ISBN 90 6144 030 0

31) Veefkind, A.
CONDUCTING GRIDS TO STABILIZE MHD GENERATOR PLASMAS AGAINST IONIZATION INSTABILITIES. TH-report 72-E-31. September 1972. ISBN 90 6144 031 9

32) Daalder, J.E. and C.W.M, Vos
DISTRIBUTION FUNCTIONS OF THE SPOT DIAMETER FOR SINGLE- AND MULTI-CATHODE DISCHARGES IN VACUUM. TH-report 73-E-32. January 1973. ISBN 90 6144 032 7

33) Daalder, J.E.
JOULE HEATING AND DIAMETER OF THE CATHODE SPOT IN A VACUUM ARC. TH-report 73-E-33. January 1973. ISBN 90 6144 033 5

34) Huber, C.
BEHAVIOUR OF THE SPINNING GYRO ROTOR. TH-report 73-E-34. February 1973. ISBN 90 6144 034 3

35) Bastian, C. et al.
THE VACUUM ARC AS A FACILITY FOR RELEVANT EXPERIMENTS IN FUSION RESEARCH. Annual Report 1972. EURATOM-T.H.E. Group "Rotating Plasma". TH-report 73-E-35. February 1973. ISBN 90 6144 035 1

36) Blom, J.A.
ANALYSIS OF PHYSIOLOGICAL SYSTEMS BY PARAMETER ESTIMATION TECHNIQUES. 73-E-36. May 1973. ISBN 90 6144 036 X

37) Lier, M.C. van and R.H.J.M. Otten
AUTOMATIC WIRING DESIGN. TH-report 73-E-37. May 1973. ISBN 90 6144 037 8 (vervalt zie 74-E-44)

38) Andriessen, F.J., W. Boerman and I.F.E.M. Holtz
CALCULATION OF RADIATION LOSSES IN CYLINDRICAL SYMMETRICAL HIGH PRESSURE DISCHARGES BY MEANS OF A DIGITAL COMPUTER. TH-report 73-E-38. October 1973. ISBN 90 6144 038 6

39) Dijk, J., C.T.W. van Diepenbeek, E.J. Maanders and L.F.G. Thurlings
THE POLARIZATION LOSSES OF OFFSET ANTENNAS. TH-report 73-E-39. June 1973. ISBN 90 6144 039 4 (in preparation)

40) Goes, W.P.
SEPARATION OF SIGNALS DUE TO ARTERIAL AND VENOUS BLOOD FLOW IN THE DOPPLES SYSTEM THAT USES CONTINUOUS ULTRASOUND. TH-report 73-E-40. September 1973. ISBN 90 6144 040 8

41) Damen, A.A.H.
COMPARATIVE ANALYSIS OF SEVERAL MODELS OF THE VENTRICULAR DE-POLARISATION; INTRODUCTION OF A STRING-MODEL. TH-report 73-E-41. October 1973. ISBN 90 6144 041 6

42) Dijk, G.H.M. van
THEORY OF GYRO WITH ROTATING GIMBAL AND FLEXTURAL PRIOTS.
TH-report 73-E-42. November 1973. ISBN 90 6144 042 4

43) Breimer, A.J.
ON THE IDENTIFICATION OF CONTINUOUS LINEAR PROCESSES. TH-report
74-E-43, January 1974. ISBN 90 6144 043 2

44) Lier, M.C. van and R.H.J.M. Otten
CAD OF MASKS AND WIRING. TH-report 74-E-44. February 1974.
ISBN 90 6144 044 0

45) Bastian, C. et al.
EXPERIMENTS WITH A LARGE SIZED HOLLOW CATHODE DISCHARGE FED WITH
ARGON. Annual Report 1973. EURATOM-T.H.E. GRoup "Rotating Plasma".
TH-report 74-E-45. April 1974. ISBN 90 6144 045 9

46) Roer, Th.G. van de
ANALYTICAL SMALL-SIGNAL THEORY OF BARITT DIODES. TH-report 74-E-46.
May 1974. ISBN 90 6144 046 7

47) Leliveld, W.H.
THE DESIGN OF A MOCK CIRCULATION SYSTEM. TH-report 74-E-47. June 1974.
ISBN 90 6144 047 5

48) Damen, A.A.H.
SOME NOTES ON THE INVERSE PROBLEM IN ELECTRO CARDIOGRAPHY. TH-report
74-E-48. July 1974. ISBN 90 6144 048 3

49) Meeberg, L. van de
A VITERBI DECODER. TH-report 74-E-49. October 1974. ISBN 90 6144 049 1

50) Poel, A.P.M. van der
A COMPUTER SEARCH FOR GOOD CONVOLUTIONAL CODES. TH-report 74-E-50.
October 1974. ISBN 90 6144 050 3

51) Sampic, G.
THE BIT ERROR PROBABILITY AS A FUNCTION PATH REGISTER LENGTH IN THE
VITERBI DECODER. TH-report 74-E-51. October 1974. ISBN 90 6144 051 3

52) Schalkwijk, J.P.M.
CODING FOR A COMPUTER NETWORK. TH-report 74-E-52. October 1974.
ISBN 90 6144 052 1

53) Stapper, M.
MEASUREMENT OF THE INTENSITY OF PROGRESSIVE ULTRASONIC WAVES BY MEANS
OF RAMAN-NATH DIFRACTION. TH-report 74-E-53. November 1974.
ISBN 90 6144 053 X

54) Schalkwijk, J.P.M. and A.J. Vinck
SYNDROME DECODING OF CONVOLUTIONAL CODES. TH-report 74-E-54.
November 1974. ISBN 90 6144 054 8

55) Yakimov, A.
FLUCTUATIONS IN IMPATT-DIODE OSCILLATORS WITH LOW q-SECTORS.
TH-report 7--E-55. November 1974. ISBN 90 6144 054 6

56) Plaats, J. van der

ANALYSIS OF THREE CONDUCTOR COAXIAL SYSTEMS.
TH-report 75-E-56. March 1975. ISBN 90 6144 056 4.

57) Kooy, C.
RE-OPTICAL ANALYSIS OF A TWO DIMENSIONAL APERTURE RADIATION PROBLEM.
TH-report 75-E-57. April 1975. ISBN 90-6144-057-2.