



UNIwersytet Technologiczno-Przyrodniczy
IM. JANA I JĘDRZEJA ŚNIADECKICH
W BYDGOSZCZY

ZESZYTY NAUKOWE NR 256

**TELEKOMUNIKACJA
I ELEKTRONIKA
13**

WYDZIAŁ TELEKOMUNIKACJI
I ELEKTROTECHNIKI



BYDGOSZCZ – 2010



UNIwersytet Technologiczno-Przyrodniczy
IM. JANA I JĘDRZEJA ŚNIADECKICH
W BYDGOSZCZY

ZESZYTY NAUKOWE NR 256

**TELEKOMUNIKACJA
I ELEKTRONIKA
13**

2

REDAKTOR NACZELNY
prof. dr hab. inż. Janusz Prusiński

REDAKTOR DZIAŁOWY
dr inż. Sławomir Cieślik

OPRACOWANIE TECHNICZNE
mgr Patrycja Fereni-Morzyńska

© Copyright

Wydawnictwa Uczelniane Uniwersytetu Technologiczno-Przyrodniczego
Bydgoszcz 2010

ISSN 1899-0088

Wydawnictwa Uczelniane Uniwersytetu Technologiczno-Przyrodniczego
ul. Ks. A. Kordeckiego 20, 85-225 Bydgoszcz, tel. 52 3749482, 3749426
e-mail: wydawucz@utp.edu.pl <http://www.wu.utp.edu.pl/>

Wyd. I. Nakład 80 egz. Ark. aut. 6,1. Ark. druk. 7,8.
Zakład Małej Poligrafii UTP Bydgoszcz, ul. Ks. A. Kordeckiego 20

Contents

1. Andrzej Handkiewicz, Paweł Śniatała, Grzegorz Pałaszyński, Szymon Szczęsny, Piotr Katarzyński, Michał Melosik, Mariusz Naumowicz – Automated DCT layout generation using AMPLE language.....	5
2. Ireneusz Olszewski - A designing of telecommunication network for the Bydgoszcz code area	15
3. Włodzimierz Pogribny, Dariusz Surma – Modified sliding Wiener-Khintchin transform	31
4. Tomasz Sosnowski, Grzegorz Bieszczad, Mariusz Kastek, Henryk Madura – FPGA-BASED system for image processing in high resolution infrared camera	43
5. Sławomir Andrzej Torbus, Marta Kolasa, Rafał Długosz – Application of the Kohonen neural network in analysis of the measurement results of the polarization mode dispersion	55
6. Tomasz Talaśka, Paweł Przedwojski, Jakub Dalecki, Rafał Długosz – Comparison of different hardware realizations of the Winner Takes All neural network.....	67
7. Iwona Grobelna, Michał Grobelny, Marian Adamski – Petri nets and activity diagrams in logic controller specification - transformation and verification	79
8. Włodzimierz Pogribny, Mariusz Sulima - Implementation of the direct windowed DHT parallel algorithm.....	93
9. Paweł Latos, Bożydar Dubalski, Tomasz Marciniak, Beata Marciniak – Demand forecasting for spare parts	103
10. Jacek Jasielski, Stanisław Kuta, Witold Machowski, Wojciech Kołodziejki - Low voltage, high-speed four-quadrant CMOS transconductance multiplier	115

AUTOMATED DCT LAYOUT GENERATION USING AMPLE LANGUAGE

Andrzej Handkiewicz, Paweł Śniatała, Grzegorz Pałaszynski, Szymon Szczęsny,
Piotr Katarzyski, Michał Melosik, Mariusz Naumowicz

Poznań University of Technology
Chair of Computer Engineering
Piotrowo 3A, 60-965 Poznań, Poland
Szymon.Szczesny@put.poznan.pl

Summary: Designing SI circuits layouts is a demanding task. The process is very time consuming and there is a high risk of making mistakes. It would be much easier if there were a CAD tool doing part of the job for ourselves. This is the place where a possible solution comes in – the AMPLE script language in the ICStation environment. AMPLE is a script language that can be used to generate layouts. Apart from making a layout faster the AMPLE generator enables parametrisation of SI devices and can also be technology-independent. It provides a way for automating and speeding up the process of designing a layout. This paper presents a DCT layout generator which takes advantage of the AMPLE language and offers parametrisation that can make the design process independent from the technology used.

Keywords: AMPLE, layout, discrete cosine transform, current mirror, analog circuit

1. INTRODUCTION

Digital circuits become more and more popular nowadays. There are a lot of CAD tools that help designers in automating most parts of the design flow. However analog circuits are still very important and in some situations irreplaceable. They are, for example, widely used in image processing and video compression. Unfortunately CAD support is unsatisfying and insufficient in this matter, therefore the problem with the automatic generation of an analog layout is currently a very popular topic [1], [2], [3].

Moreover, analog circuits layout generation also requires automation and assistance of software tools which could handle the operations of generating a standard analog cell which would depend on input parameters. For instance, for a current mirror an input parameter could be the current gain, while the size of the coefficient matrix might be a parameter for a DCT implementation in the CMOS technology.

Authors of the article took into consideration all those demands for CAD tools and proposed the following solution - automated generation of a DCT layout using the AMPLE script language - described in the following chapters. The proposed method shows how to reduce total time of designing analog circuits layouts, also the way of parametrising the designing process and shows how to prevent making mistakes.

2. DISCRETE COSINE TRANSFORM

This article uses the concept of the two-dimensional Discrete Cosine Transform (DCT), therefore in this section we shall review some basic information about it.

Using the Discrete Cosine Transformation implementations is very important in the process of compressing images. JPEG and MPEG are only two, most popular examples of many standards which use DCT. It has been observed that the two-dimensional (2-D) Discrete Cosine Transform has the output similar to the one produced by the Karhunen-Loeve transformation [4] and uses image-independent basis functions. This is the reason why DCT-based image coding is applied to all video compression standards [11]. In these standards, the image is divided into 8x8 blocks in the spatial domain and DCT transforms them into 8x8 blocks in the 2-D frequency domain. Such a block size is convenient in respect of computational complexity. Larger sizes are also possible, however they do not offer any significant improvement concerning the level of compression.

2-D DCT in the size of 8x8 can be expressed as [5]:

$$y_{k,l} = \frac{c(k)c(l)}{4} \sum_{i=0}^7 \sum_{j=0}^7 x_{ij} \cos \frac{(2i+1)k\pi}{16} \cos \frac{(2j+1)l\pi}{16}, \quad (1)$$

Where $k, l = 0, 1, \dots, 7$ and

$$c(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0, \\ 1, & k \neq 0. \end{cases} \quad (2)$$

Assuming that the matrices Y and X are composed of elements y_{ij} and x_{ij} , where $i, j = 0, 1, \dots, 7$ respectively, the relation (1) can be also written in the matrix form as:

$$Y = CXC^T \quad (3)$$

and the matrix of coefficients C is as in (4), where:

$$\begin{aligned} a &= \cos(\pi/16); \\ b &= \cos(2\pi/16); \\ c &= \cos(3\pi/16); \\ d &= \cos(4\pi/16); \\ e &= \cos(5\pi/16); \\ f &= \cos(6\pi/16); \\ g &= \cos(7\pi/16); \end{aligned}$$

The main property of 2-D DCT, with respect to implementation, is separability. On the basis of the matrix equation (3), written in the form (5) we can realize 2-D DCT with two 1-D ones. The matrix X denotes one input 8x8 block, and its transposition X^T in the relation $Z = X^T C^T$ means that it is read out column by column. The matrix Z , containing intermediate results, is obtained with the use of 1-D DCT, and can be saved in a memory array.

$$C = \frac{1}{2} \begin{bmatrix} d & d & d & d & d & d & d & d \\ a & c & e & g & -g & -e & -c & -a \\ b & f & -f & -b & -b & -f & f & b \\ c & -g & -a & -e & e & a & g & -c \\ d & -d & -d & d & d & -d & -d & d \\ e & -a & g & c & -c & -g & a & -e \\ f & -b & b & -f & -f & b & -b & f \\ g & -e & c & -a & a & -c & e & -g \end{bmatrix}, \quad (4)$$

$$Y = Z^T C^T, Z = X^T C^T \quad (5)$$

As we can see, the matrix of coefficients (4) only depends on the size of the 2-D Discrete Cosine Transformation. Therefore, it is possible to parametrise the algorithm for designing a circuit which could calculate the DCT. This article proposes the idea of a DCT layout generator based on an AMPLE script. The next part briefly describes the AMPLE language.

3. WHAT IS AMPLE?

AMPLE is an abbreviation for Advanced Multi-Purpose Language which is a standard used in the Mentor Graphics Common User Interface [7], [8] common to all Falcon Framework-based applications. End-users can use AMPLE to modify and extend Falcon Framework-based software functionality by evaluating expressions, creating and assigning variables, defining and executing new functions and directly executing built-in functions. AMPLE supports the C library and module dynamic linking to the existing in-house, as well as third-party solutions bound to the Falcon Framework. Moreover, it supports popular statements like, for instance: iterations, multi-conditional branching, loops, loop's interrupts and terminations.

AMPLE and the Common User Interface contain constructors for defining custom menus, prompt bars, forms, function keys and strokes, as well as user's custom functions and commands. Implemented scripts can be included as an integral part of the general design flow for automating the design of SI circuits. Due to these reasons AMPLE is perfect for our layout customisation.

4. DCT LAYOUT GENERATOR

The idea of writing a layout generator is to design a schematic, create a layout using Schematic Driven Layout (SDL) and finally to analyse this layout in order to create a generator which could take input parameters. All these steps are covered in the sections below.

4.1. SCHEMATIC

Current is the signal that drives information in the SI technology [5]. Therefore, multiplication operations (scaling signals) from (1) can be realised using current mirrors.

Addition operations are realised in the nodes according to the Kirchoff's first rule - the Kirchoff's current law. The schematic of an 8x8 DCT which is shown in Figure 1 has been chosen for the implementation.

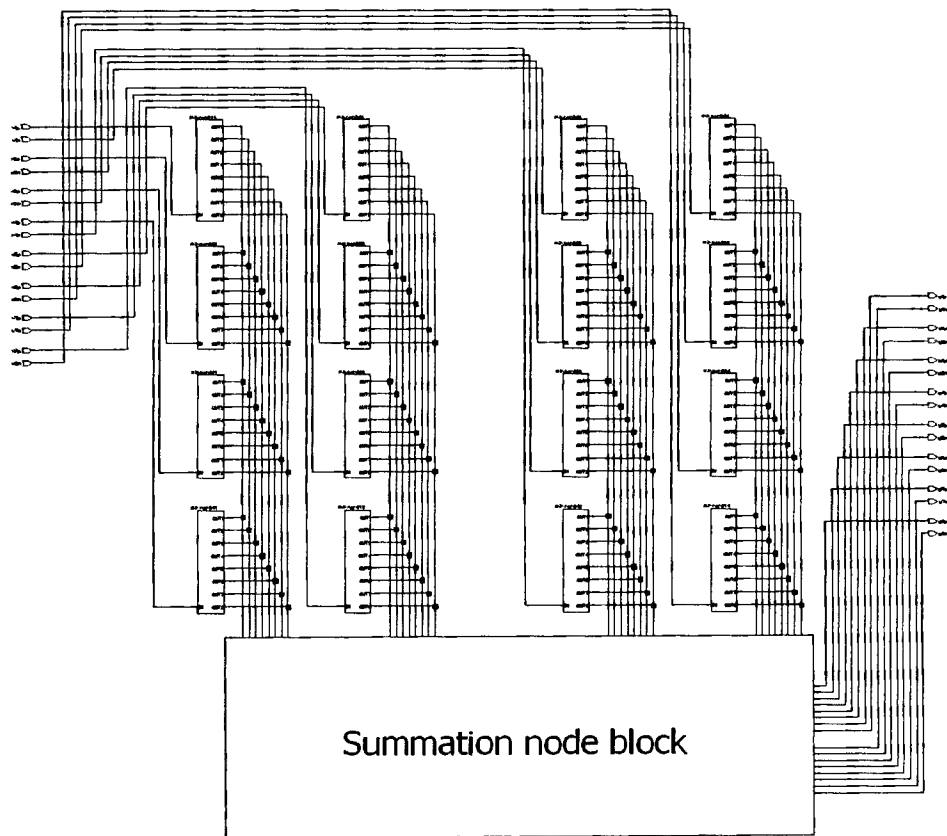


Fig. 1. Schematic of the DCT circuit

The schematic consists of many elements built in the same way, therefore it was constructed hierarchically. Basic cells of the hierarchy are the output stages of current mirrors (in fact they are inverters) with different NMOS and PMOS lengths but the same widths. One of such stages is shown in Fig. 2. The 8x8 DCT includes 8 current mirrors (with one input and eight outputs - Fig. 3) and each one, as an output stage, includes a cell shown in Figure 2. Connections in the block of summation nodes depend on the signs of the coefficients in the matrix C^T . Because of balanced structure, the top-level schematic of 1-D DCT consists of sixteen 1-input, 8-output current mirrors.

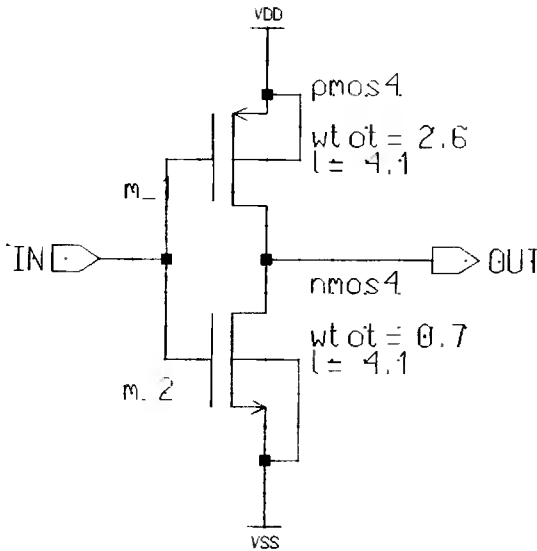


Fig. 2. Output stage of current mirror

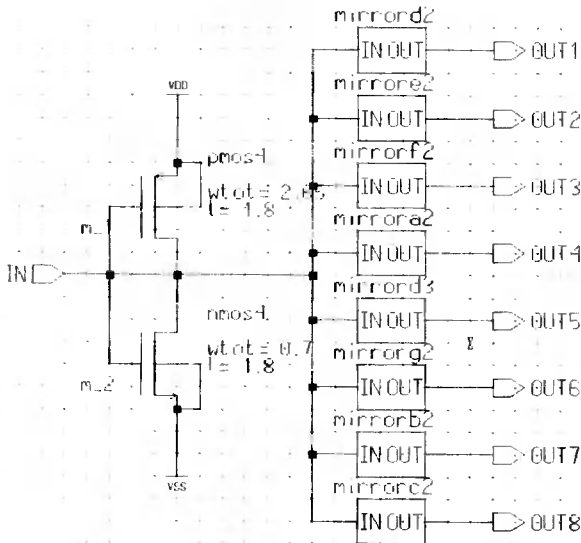


Fig. 3. 8-output current mirror

2-D DCT can be realized in two ways: with and without a memory array. Two circuits shown in Figure 1 are necessary to realize 2-D DCT if the memory array for intermediate results Z in (5) is used [5] and sixteen circuits are necessary for the realization without the memory array [6].

4.2. SCHEMATIC DRIVEN LAYOUT

Providing that we have a schematic we are able to use the ICStation environment to build a layout for this particular circuit [9], [10]. Some features, like the Schematic Driven Layout (SDL, mentioned above) are very helpful in the custom VLSI design. We can also take advantage of tools such as auto-routing or auto-placement. With the support of such software we are able to create a layout, however and unfortunately only for this particular schematic.

If we decide to change the number of inputs and outputs, we have to create a new schematic and repeat the whole process from the very beginning. Moreover, if we change the technology to a newer one, for example: where certain distances between elements can be smaller, we need to re-create the layout once again. Therefore, this is what we demand from our AMPLE script: parametrisation of the DCT layout.

4.3. AMPLE SCRIPT

Our generator was written after taking into consideration the SDL result. Because of the layout's complexity the script was divided into sections and functions responsible for elementary tasks.

The beginning of the script contains declarations of the input data (the size of the current mirror), as well as parameters setting up, for example: distances between elements. This is very useful because thanks to it we can read design rules from a special file, thus making the script technology-independent. In such a situation, the script would open this special file and, taking into consideration the rules, it would minimise the area necessary to make a layout.

The second part of the AMPLE generator includes calculations which check, depending on the values of parameters, for example: dimensions and position of the specified current mirror's layout. Heights and widths of inverters are calculated taking into consideration the sizes of power supply ports and distances between ports and transistors. After the above steps are completed, it is possible to generate the layout of our DCT.

Initially, as a result of a function, a simple input stage of the current mirror is obtained. The previously specified argument defines the length of the NMOS and PMOS transistors. Other arguments stored in a vector define sizes of transistors in output stages. Subsequent steps are shown in Figure 4.

At first, transistors are created. PMOS transistors and NMOS transistor, all of the same length, are placed at the top and at the bottom, respectively. Due to the common width of all PMOS transistors the scaling factors are obtained as the relation (6) of the length L_i of the transistor in the output stage to the length L_{in} of the transistor in the relevant input stage.

The sizes of transistors for the whole current mirror can be automatically calculated using dedicated tools [12]. Thanks to the solution with common widths it becomes easy to design current mirrors with the same heights, which allows to place all of them using the verse strategy (a design strategy of putting cells next to each other in rows), which allows minimisation of the area of the chip.

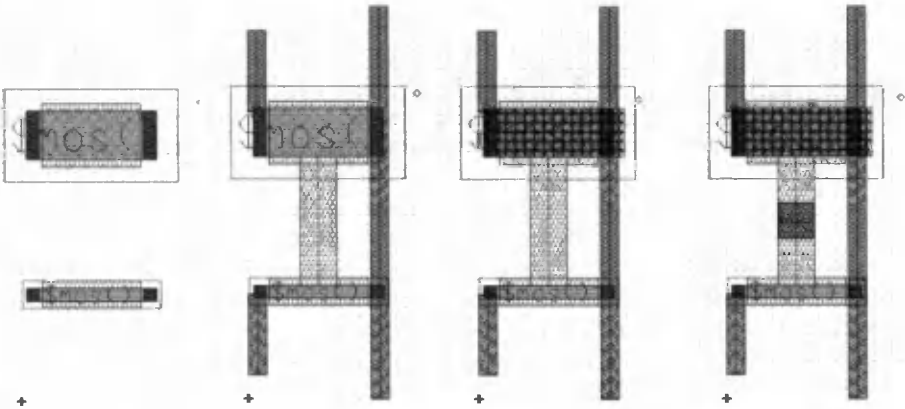


Fig. 4. 8-output current mirror

$$\alpha_i = \frac{L_i}{L_{in}} \quad (6)$$

After placing transistors the connections are drawn. Afterwards FIMP and NTUB layers, and eventually VIA's are drawn. As an example a piece of code with a command which creates a via between the first and the second metal layers is presented below:

```
extern net_out_w = 1.0;
extern nmos_l = 3.775;
```

```
local outvia_x = inv_w(nmos_l) - net_out_w;
local outvia_y = vss_h + vss_to_nmos;
```

```
$add_point_device("$via", @via, [], [x+outvia_x, y+outvia_y], [{"via_w", net_out_w},
{"via_l", net_out_w}, {"t", "m1m2"}, {"cinx", void}, {"ciny", void}, {"dorc", "d"}],
@placed);
```

As we can see it allows us to parametrise dimensions of a via which is defined using the *net_out_w* variable. It is also possible to easily describe the position of a via ($x + outvia_x, y + outvia_y$), which can be calculated using appropriate constants with port sizes and the distance between the VSS port and nmos transistors, also using predefined functions calculating the width of the inverter stage. In the same way all paths, layers and vias can be drawn with the possibility of parametrisation.

Using the initial function and the vector with input arguments, an n-output current mirror can be generated in a loop containing functions which create an output stage, add a relevant input stage and proper connections. The number of output stages in a particular current mirror is the same as the size of the DCT which is set in the part of the script containing parameters. The result has been shown in Figure [5].

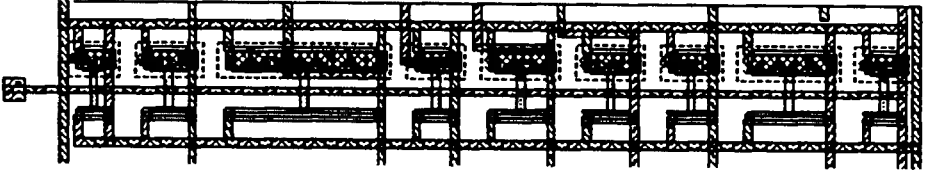
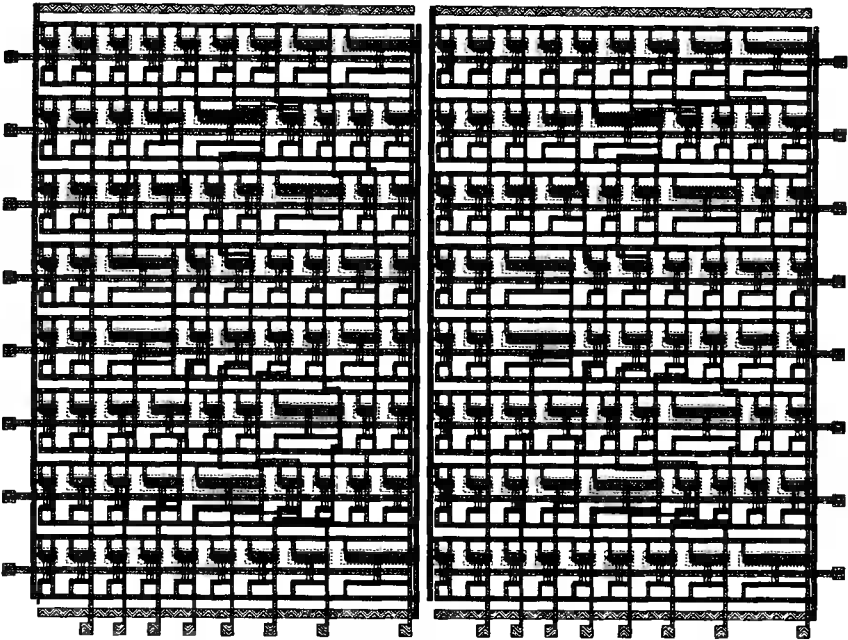


Fig. 5. Layout of a single current mirror

The next part of the AMPLE script uses a function which generates a current mirror layout and builds the DCT. The number of iterations in which the mirrors are generated, depends on the number specified at the beginning of the script. Connections between mirrors are also placed in this part. Finally input and output ports must be created to finish the generation of the DCT layout. This is the place where we can decide if we want them on the left, right, top or bottom side of the layout.

If the size of the DCT is set to 8×8 , the layout shown in Figure 6 will be generated. It is worth noticing that using AMPLE and the presented method of dividing the circuit to 8-output current mirrors we can quickly and automatically obtain the layout of a circuit which consists, in this specific case, of 288 transistors. The method is also a solution for avoiding making mistakes during the process of designing a layout.

Fig. 6. 8×8 DCT Layout

It would take many hours to design such layouts in a traditional way. Furthermore, there would be no possibility for a fast rebuilding of circuits to other sizes. AMPLE allows to obtain circuits in a short time and offers the possibility of rebuilding.

5. SUMMARY

It becomes more and more often required to propose and improve methods for the automation of the analog circuits layout design process. The AMPLÉ script language is a perfect solution in case of a need to have custom layout generators, especially involving circuits that are often used as subcircuits in larger devices. We then only need to set input parameters, run the script and the layout is ready. Many hours of designing a layout are saved for other purposes. In addition, it is possible to make a generator independent from the technology used. In other words, all of the repeatable and time consuming tasks connected with layout generation can be automated using AMPLÉ scripts. In our work we have shown how to quickly design a layout of an analog circuit with the possibility of parametrisation, using the DCT circuit as an example.

BIBLIOGRAPHY

- [1] Yilmaz E., Dundar G., 2009. Analog layout generator for CMOS circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems archive, Volume 28, Issue 1, pp. 32-45.
- [2] Graeb H., Balsa F., Castro-Lopez R., Chang Y.W., Fernandez F.V., Lin P.H., Strasser M., 2009. Analog layout synthesis - Recent advances in topological approaches. Design, Automation & Test in Europe Conference & Exhibition, DATE'09, pp. 274-279.
- [3] Castro-Lopez R., Guerra O., Rocca E., Fernandez F.V., 2008. An Integrated Layout-Synthesis Approach for Analog ICs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume: 27, Issue: 7, pp. 1179-1189.
- [4] Jain A.K., 1999. Fundamentals of Digital Integrated Circuits: Analysis and Design, McGraw-Hill, Inc.
- [5] Handkiewicz A., 2002. Mixed-Signal Systems: A Guide to CMOS Circuit Design. John Wiley and Sons.
- [6] Pankaala M., Virtanen K., Paasio A. i in. 2006. An analog 2-DCT processor. IEEE Trans. Circ. and Syst. for Video Technol. Vol. 16, no. 10, pp.1209-1216
- [7] Mentor Graphics Corporation, 2005. AMPLÉ for IC Flow Reference Manual, 1.1 edition.
- [8] Mentor Graphics Corporation, 2005. AMPLÉ for IC Flow User's Manual, 1.1 edition.
- [9] Mentor Graphics Corporation, 2005. IC Station Reference Manual, 1.1 edition.
- [10] Mentor Graphics Corporation, 2005. IC Station User's Manual, 1.1 edition.
- [11] Bhaskaran V., Konstantinidies K., 1995. Image and Video Compression Standards, Kluwer Academic Publishers, Boston, MA.
- [12] Rudnicki R. Chosen tools for automatic design of switched-current circuits, (in Polish), Ph.D. dissertation, Poznań University of Technology.

AUTOMATYCZNA GENERACJA LAYOUTU UKŁADU DCT
PRZY POMOCY JĘZYKA AMPLE

Streszczenie

Projektowanie layoutów układów SI nie jest zadaniem łatwym. Proces ten wymaga dużych nakładów czasu, istnieje ogromne ryzyko popełnienia pomyłki przez projektanta, a projektowane układy są zależne od technologii, co wymusza ich całkowitą przebudowę w sytuacji zmiany technologii na nowszą. Zadanie to byłoby dużo prostsze, gdyby istniały narzędzia CAD automatyzujące proces projektowania. W obszarze tym możliwe jest wykorzystanie zaproponowanego w artykule rozwiązania – użycie skryptowego języka AMPLE dostępnego w środowisku ICStation. Oprócz możliwości szybszego zaprojektowania prototypu, generator stworzony przy pomocy języka AMPLE umożliwia parametryzację projektowanych urządzeń SI, które stają się niezależne od technologii. Stanowi to daleko idące udoskonalenie procesu projektowania układów scalonych wykonanych w technice SI. Niniejszy artykuł opisuje zaproponowaną metodę automatycznego generowania layoutów przedstawiając jako przykład kolejne etapy realizacji układu DCT.

Słowa kluczowe: AMPLE, layout, dyskretna transformata kosinusowa, zwierciadło prądowe, układ analogowy

A DESIGNING OF TELECOMMUNICATION NETWORK FOR THE BYDGOSZCZ CODE AREA

Ireneusz Olszewski

University of Technology and Life Sciences
Faculty of Telecommunications and Electrical Engineering
Al. Prof. S. Kaliskiego 7, 85-796 Bydgoszcz, Poland

Summary: In this paper an engineering approach to designing a telecommunication network over the Internet Protocol has been presented. The proposed network covers demand for telecommunication services in the Bydgoszcz code area operated by the Polish Telecommunications Co. In this network two classes of services were distinguished: Voice over the Internet Protocol (VoIP) and access to the Internet. The logical structure of the network is a superposition of two structures. First of them contains links that support VoIP and the second one contains links that provide access to the Internet. The physical structure of the network was implemented basing on the Synchronous Digital Hierarchy.

Keywords: router, link, designing

1. INTRODUCTION

This study contains engineering approach to designing of telecommunication network over Internet Protocol and can be an extension of Telecommunications Systems lecture for students of telecommunication. It contains a design of telecommunications network for assumed numeral data of Bydgoszcz code area (former Bydgoszcz province) operated by the Polish Telecommunications Co (PT Co.). National operator offers real-time services as well as the access to the Internet. The designed network should cover the demand for telecommunications services in the Bydgoszcz code area and interoperate with the networks of others operators. Adopted output data, i.e. the number of subscribers of the Polish Telecommunications Co. network, the number of users in the various municipalities, the number of classes of services, etc. differ significantly from the actual data and were adopted only for the project. The design includes the backbone layer of the network without the access layer. Only two classes of traffic: Voice over the Internet Protocol and access to the Internet were taken into account in output data.

The need for writing this article arose due to lack of the holistic approach to designing a network over IP platform in literature. This study was accompanied by a number of explanations and comments to increase its educational value.

Presentation of the whole scheme of network design (traffic structure, logical structure and physical structure) is a contribution of the author.

The whole study consists of six parts. The first part contains the command and output data for the project. The second part determines number of routers of the network and division of the entire area of the network into areas associated with particular routers. The third part shows the phone traffic and Internet traffic structures of the network. The fourth part contains the logical structure of the proposed network, together with accepted assumptions. The fifth part shows the physical structure of the proposed network. Part sixth contains a summary and conclusions.

Design of telecommunication network for Bydgoszcz code area (former Bydgoszcz province)

1. Design requirements:

a) Logical topology:

- determine the number of routers;
- determine the structure of VoIP and the Internet traffic (for **exemplary router draw a flowchart of outgoing and incoming traffic**);
- determine the logical structure of network (bandwidth of links of network and bandwidth of links at points of connection with MAN Bydgoszcz and with network of Polish Telecommunications Co. (PT Co.)

b) Physical topology (on the basis of SDH):

- determine course of transmission routes;
- make the dimensioning of transmission systems (in the case of SDH there is a need to take into consideration the ring configuration);

for a telecommunication network serving traffic generated by 25000 subscribers of VoIP, assuming that 50% of subscribers will apply for access to the Internet.

2. Design assumptions:

a) number of subscribers in the PT Co. network – 150000

b) point of connection for Internet services – MAN Bydgoszcz

c) points of connection of proposed network with PT Co. network:

- for local traffic – local exchanges (LE): LE Chojnice, LE Inowrocław;
- for long-distance traffic and international long-distance traffic – long-distance exchange (LDE): LDE Bydgoszcz;

It should be noticed that LDE Bydgoszcz, LE Chojnice and LE Inowrocław are located in the PT Co. network which is a circuit switched network.

3. The percentage of subscribers, depending on the size of bandwidth of access to the Internet: 10% of subscribers with bandwidth 2048 kbit/s, 20% of subscribers with bandwidth 1024 kbit/s and 70% of subscribers with bandwidth 512 kbit/s;

4. The distribution of subscribers in Bydgoszcz code area is given in Table 1. It should be noted that national operator (PT Co.) offers real-time services as well as the access to the Internet. Therefore, the numbers of subscribers in the particular communities are so small.

Table 1. The number of subscribers in the particular communities in Bydgoszcz code area

No.	Community	Subscribers	No.	Community	Subscribers
1	Bydgoszcz	5000	30	Bukowiec	100
2	Barcin	350	31	Cekcyn	250
3	Brusy	500	32	Dąbrowa	150
4	Chojnice	1500	33	Dąbrowa Biskupia	150
5	Czersk	750	34	Dąbrowa Chełmińska	100
6	Gniewkowo	350	35	Dobrcz	150
7	Inowrocław	2500	36	Dragacz	100
8	Janowiec Wielkopolski	250	37	Drzycim	100
9	Janikowo	300	38	Gąsawa	150
10	Kamień Krajeński	250	39	Gostycyn	200
11	Kcynia	400	40	Jezióra Wielkie	150
12	Koronowo	400	41	Jeżewo	100
13	Kruszwica	350	42	Kęsowo	100
14	Łabiszyn	250	43	Lniano	100
15	Mogilno	750	44	Lubiewo	200
16	Mrocza	200	45	Nowa Wieś Wielka	150
17	Nakło nad Notecią	750	46	Osie	100
18	Nowe	200	47	Osielsko	150
19	Pakość	250	48	Pruszcz	100
20	Sępólno Krajeńskie	750	49	Rogowo	250
21	Solec Kujawski	750	50	Rojewo	150
22	Strzelno	300	51	Sadki	150
23	Szubin	300	52	Sicienko	150
24	Świecie	750	53	Sośno	150
25	Trzemeszno	350	54	Śliwice	100
26	Tuchola	1250	55	Świekatowo	150
27	Więcbork	300	56	Warlubie	150
28	Żnin	750	57	Złotniki Kujawskie	150
29	Białe Błota	200			
				Sum	25000

PROJECT SOLUTION

2. NUMBER OF ROUTERS

The number of routers to some extent is related to the number of subscribers served by a single Call Manager (cluster). Call Manager is installed on the server connected to a router. Single Call Manager can handle up to 7500 subscribers, while the cluster to 30000 subscribers [5]. Assumption: number of routers equal to n .

In this project it was assumed that the entire area served by the network is divided into five equal areas in terms of the number of users: Bydgoszcz area, Chojnice area, Koronowo area, Żnin area and Inowrocław area. Each area will include 5000 users and will be served by a single Call Manager. There are also 5 routers ($n = 5$) in backbone layer. They are named by analogy: Bydgoszcz router, Chojnice router, Koronowo router, Żnin router and Inowrocław router (Each router is associated with appropriate area). Figure 1 shows the division of the entire area of the network into areas associated with particular routers, while Table 2 contains the allocation of communities in these areas.

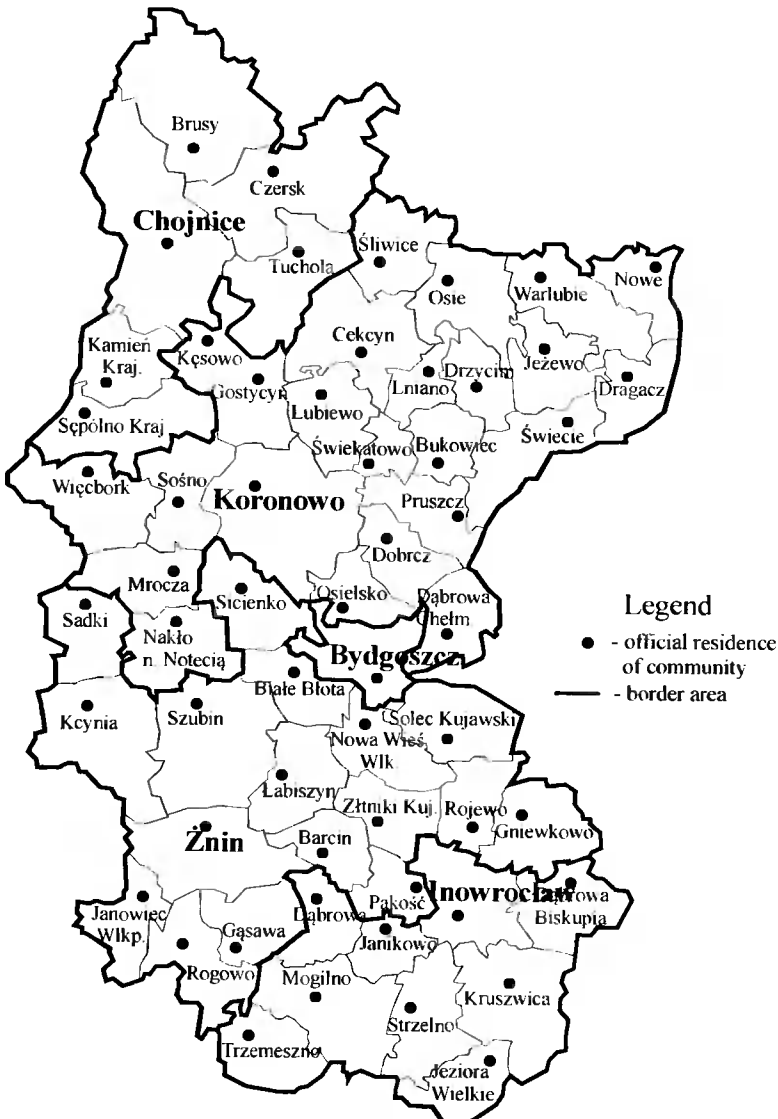


Fig. 1. The division of the entire area of the network into areas associated with particular routers

Table 2. The allocation of communities to individual areas of the network

No.	Area/Community	Users	No.	Area/Community	Users
	Bydgoszcz Area		22	Śliwice	100
1	Bydgoszcz	5000	23	Świekatowo	150
	Chojnice Area		24	Warlubie	150
1	Brusy	500		Żnin Area	
2	Chojnice	1500	1	Barcin	350
3	Czersk	750	2	Gniewkowo	350
4	Kamień Krajeński	250	3	Janowiec Wlkp	250
5	Tuchola	1250	4	Kecynia	400
6	Sępólno Krajeńskie	750	5	Łabiszyn	250
	Koronowo Area		6	Pakość	250
1	Koronowo	400	7	Solec Kujawski	750
2	Mrocza	200	8	Szubin	300
3	Nakło n Notecią	750	9	Żnin	750
4	Nowe	200	10	Białe Błota	200
5	Świecie	750	11	Gąsawa	150
6	Więcbork	300	12	Nowa Wieś Wielka	150
7	Bukowiec	100	13	Rogowo	250
8	Cekcyn	250	14	Rojewo	150
9	Dąbrowa Chełmińska	100	15	Sadki	150
10	Dobrzec	150	16	Sicienko	150
11	Dragacz	100	17	Złotniki Kujawskie	150
12	Drzycim	100		Inowrocław Area	
13	Gostycyn	200	1	Inowrocław	2500
14	Jeżewo	100	2	Janikowo	300
15	Kęsowo	100	3	Kruszwica	350
16	Lniano	100	4	Mogilno	750
17	Lubiewo	200	5	Strzelno	300
18	Osie	100	6	Trzemeszno	350
19	Osielsko	150	7	Dąbrowa	150
20	Pruszcz	100	8	Dąbrowa Biskupia	150
21	Sośno	150	9	Jezióra Wielkie	150

3. STRUCTURE OF TRAFFIC

3.1. STRUCTURE OF PHONE TRAFFIC

Assumption:

L – number of subscribers of the proposed network; $L(PT Co.)$ – number of subscribers of the PT Co. network, L_i – number of subscribers in area i (associated with router i), b – band for VoIP (In this paper it was assumed 32 kbit/s);

Points of connection:

- a) for long-distance traffic (including international long-distance traffic) – LDE Bydgoszcz (point *A*)*;
 b) for local traffic – LE Chojnice and LE Inowrocław (point *B* and point *C* respectively)*;

* – mark A, B, C are introduced to simplify formulas in the remainder of the project.

Flowchart of phone traffic in router *i*.

Assumption:

Probability of selection of the desired subscriber from the proposed network and from the PT Co. network is the same;

Probability of selection of the desired subscriber from area *i* is the same as from the other areas.

The total traffic generated in router *i*, specifically generated by the subscribers included in the access network, connected with the router *i*, is divided into long-distance traffic and local traffic. In turn, the local traffic is divided into traffic directed to PT Co. network, (through distinguished connection points B and C) and the local traffic of the network, directed to individual routers. Below Figure 2a, the flowchart of outgoing traffic of router *i* for generic numeral values, was presented. On the basis of the assumptions the contribution of individual components of traffic in the total traffic was marked on the diagram. The assumed statistical multiplexing ratio is 12:1. In turn, Figure 2b shows the corresponding flowchart of traffic for the router Bydgoszcz (in example under consideration). The numbers in circles in this diagram indicate the size of bandwidth, expressed in kbit/s, required to handle traffic. It should be noted that the shown flowchart specifies only the required bandwidth for one direction of transmission. Table 3, designated as *Tout*, shows the size of the required bandwidth, taking into account that the implementation of VoIP connection requires a symmetrical bandwidth between pairs of nodes (router *i* – router *j*, router *i* – A, router *i* – B and router *i* – C) for outgoing traffic. Assuming that the amount of traffic exchanged at the inter-network points of connection (outgoing traffic – incoming traffic) are the same, and knowing (based on analogous calculations) the amount of traffic between individual routers, the flowchart for incoming traffic can be drawn.

Table 3. *Tout*. The required size of bandwidth between pairs of nodes for outgoing traffic (kbit/s)

Router/ Router	Bydgoszcz (<i>i</i> = 1)	Chojnice (<i>i</i> = 2)	Koronowo (<i>i</i> = 3)	Żnin (<i>i</i> = 4)	Inowrocław. (<i>i</i> = 5)	A: LDE	B: LE Chojnice	C: LE Inowrocław.
Bydgoszcz (<i>i</i> = 1)	X	335	335	335	335	1600	5029	5029
Chojnice (<i>i</i> = 2)	335	X	335	335	335	1600	5029	5029
Koronowo (<i>i</i> = 3)	335	335	X	335	335	1600	5029	5029
Żnin (<i>i</i> = 4)	335	335	335	X	335	1600	5029	5029
Inowrocław. (<i>i</i> = 5)	335	335	335	335	X	1600	5029	5029
A: LDE	1600	1600	1600	1600	1600	X	X	X
B: LE Chojnice	5029	5029	5029	5029	5029	X	X	X
C: LE Inowrocław.	5029	5029	5029	5029	5029	X	X	X

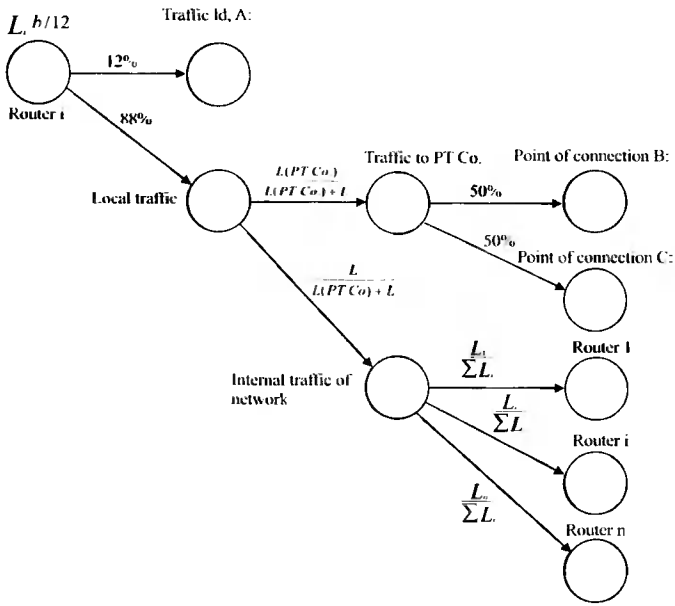


Fig. 2a. The flowchart of outgoing traffic in the router i

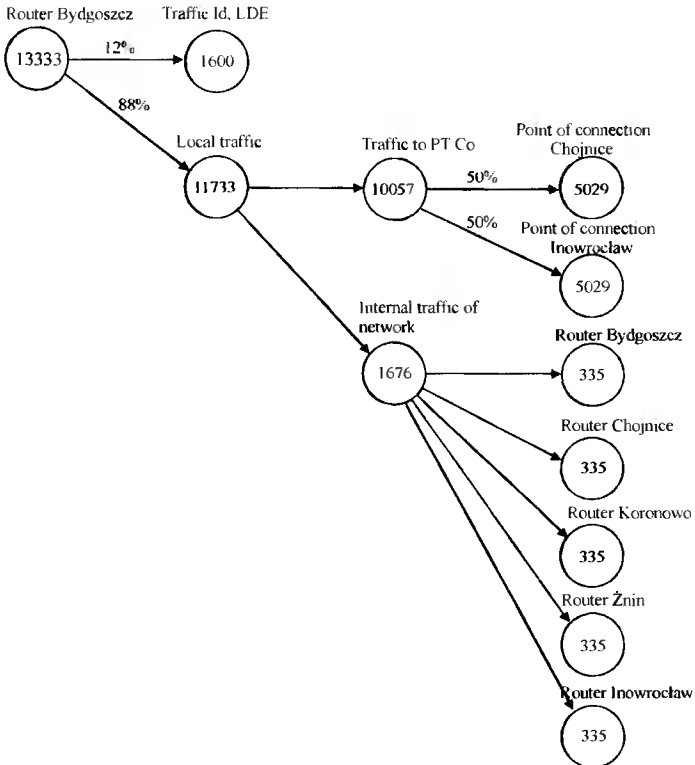


Fig. 2b. The flowchart of outgoing traffic in the router Bydgoszcz (in kbit/s)

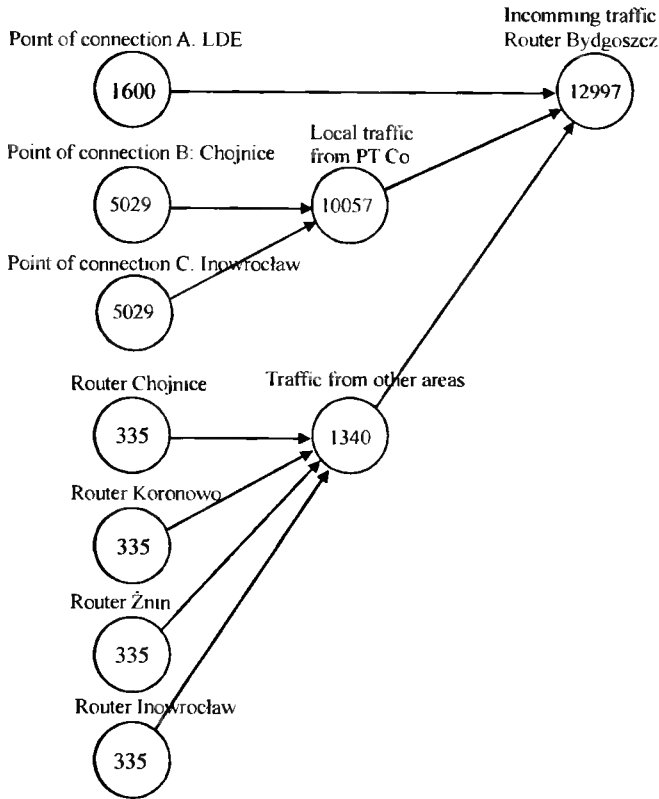


Fig. 3. The flowchart of incoming traffic in router Bydgoszcz

Table 4. T_{in} . The required size of bandwidth between pairs of nodes for incoming traffic (kbit/s)

Router/ Router	Bydgoszcz ($i = 1$)	Chojnice ($i = 2$)	Koronowo ($i = 3$)	Żnin ($i = 4$)	Inowrocław. ($i = 5$)	A: LDE	B: LE Chojnice	C: LE Inowrocław.
Bydgoszcz ($i = 1$)	X	335	335	335	335	1600	5029	5029
Chojnice ($i = 2$)	335	X	335	335	335	1600	5029	5029
Koronowo ($i = 3$)	335	335	X	335	335	1600	5029	5029
Żnin ($i = 4$)	335	335	335	X	335	1600	5029	5029
Inowrocław. ($i = 5$)	335	335	335	335	X	1600	5029	5029
A: LDE	1600	1600	1600	1600	1600	X	X	X
B: LE Chojnice	5029	5029	5029	5029	5029	X	X	X
C: LE Inowrocław.	5029	5029	5029	5029	5029	X	X	X

3.2. STRUCTURE OF INTERNET TRAFFIC

Assumptions:

L_i – number of subscribers in area i (associated with router i), P – number of subscribers (percentage) applying for access to Internet, where: $x\%$ of the bandwidth 512 kbit/s, $y\%$ of the bandwidth 1024 kbit/s and $z\%$ of the bandwidth 2048 kbit/s. The required bandwidth for incoming traffic: 256 kbit/s. Point of connection: MAN Bydgoszcz.

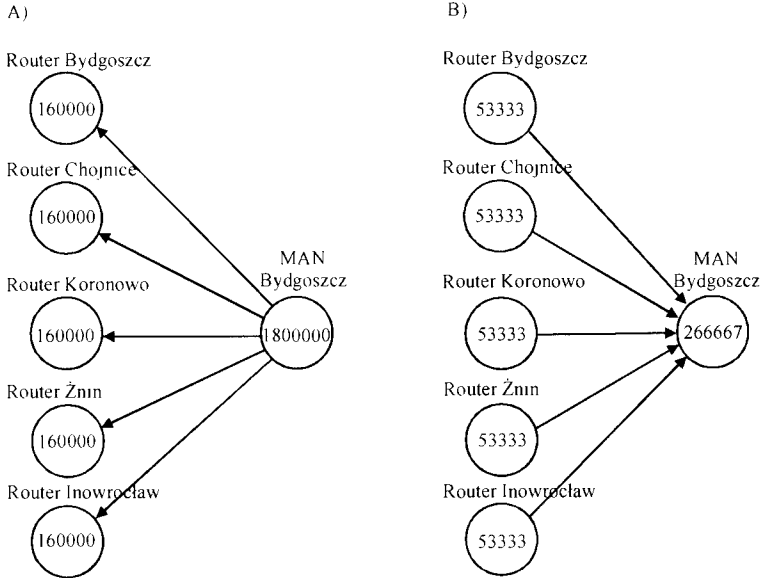


Fig. 4. The structure of Internet traffic (kbit/s): a) incoming; b) outgoing

The required size of bandwidth for the traffic $T_{MAN,i}()$ coming from the MAN Bydgoszcz to the router i for subscribers with bandwidth 512 kbit/s, 1024 kbit/s and 2048 kbit/s can be defined as follows:

$$T_{MAN,i}(512) = L_i P x k 512$$

$$T_{MAN,i}(1024) = L_i P y k 1024 \tag{1}$$

$$T_{MAN,i}(2048) = L_i P z k 2048$$

where:

k is the coefficient of concentration (equal to 12:1).

The total amount of bandwidth for traffic coming from the MAN Bydgoszcz to the router i can be defined as:

$$T_{MAN,i} = L_i P k (x 512 + y 1024 + z 2048) \tag{2}$$

Assuming that the required bandwidth per subscriber for outgoing traffic (regardless of the size of bandwidth for incoming traffic) is 256 kbit/s, the total amount of bandwidth for outgoing traffic can be defined as:

$$T_{i,MAN} = L_i P k 256(x+y+z) \quad (3)$$

For the variant under consideration: $T_{MAN,i}(512) = 74667$ kbit/s, $T_{MAN,i}(1024) = 42667$ kbit/s; $T_{MAN,i}(2048) = 42667$ kbit/s and aggregate value $T_{MAN,i} = 160000$ kbit/s; while $T_{i,MAN} = 53333$ kbit/s for $i = 1,2,\dots,5$. Figure 4 shows the flowchart of outgoing traffic and incoming traffic. It should be emphasized that the required bandwidth, on connections serving the Internet traffic between the router and a MAN Bydgoszcz, is asymmetric (greater value for incoming traffic).

4. LOGICAL STRUCTURE OF THE NETWORK

To determine the logical structure of the network it is necessary to specify the bandwidth of (unidirectional) links between individual pairs of nodes, i.e. between routers of the proposed network, between network routers and exchanges in the PT Co. network (exchanges A, B, C) and between the network routers and MAN Bydgoszcz. The required bandwidth (unidirectional) to serve telephone traffic (VoIP) can be determined summing appropriate elements of the Table T_{out} (describing the required bandwidth to serve outgoing traffic) with the relevant elements of the Table T_{in} (which determine the bandwidth required to serve incoming traffic). Thus, the required bandwidth between pairs of nodes (i, j) can be defined as:

$$T_{i,j} = T_{out_{i,j}} + T_{in_{i,j}} \quad i, j = 1 \dots n+3 \quad (4)$$

In turn, the required bandwidth to serve the Internet incoming and outgoing traffic are defined by $T_{MAN,i}$ and $T_{i,MAN}$ for $i = 1, \dots, n$ (formulas (2) and (3)). Below, Table 5 (T) shows the required bandwidth in the proposed network.

Table 5. T. The required bandwidth in the proposed network (kbit/s)

Router/ Router	Bydgoszcz (i = 1)	Chojnice (i = 2)	Koronowo (i = 3)	Żnin (i = 4)	Inowrocł. (i = 5)	A: LDE	B: LE Chojnice	C: LE Inowrocł.	MAN
Bydgoszcz (i = 1)	X	670	670	670	670	3200	10057	10057	53333
Chojnice (i = 2)	670	X	670	670	670	3200	10057	10057	53333
Koronowo (i = 3)	670	670	X	670	670	3200	10057	10057	53333
Żnin (i = 4)	670	670	670	X	670	3200	10057	10057	53333
Inowrocł. (i = 5)	670	670	670	670	X	3200	10057	10057	53333
A: LDE	3200	3200	3200	3200	3200	X	X	X	X
B: LE Chojnice	10057	10057	10057	10057	10057	X	X	X	X
C: LE Inowrocł.	10057	10057	10057	10057	10057	X	X	X	X
MAN	160000	160000	160000	160000	160000	X	X	X	X

It should be noted that the designated bandwidth of links in Table 5 (T) does not include **overheads** (i.e. the increase of the volume of information transmitted) brought by

the various protocols. In this project it was assumed that the data sent through the Internet links are served by Transmission Control Protocol (TCP), Internet Protocol (IP) and Ethernet protocol, while voice data (VoIP) are served by Real-time Transport Protocol (RTP), User Datagram Protocol (UDP), IP and Ethernet protocol [1]. Assuming that the length of sent/received data through the application layer (from the point of view of TCP/IP model) to/from the Internet is at an average of 1000 bytes and taking into account that the overheads brought by the individual protocols, the total percentage of the overheads brought by those protocols can be specified. The overhead brought by the TCP protocol (TCP header length) is 20 bytes, the IP overhead is 20 bytes and the overhead resulting from the structure of an Ethernet frame is 30 bytes [3]. Thus, the relative percentage of overhead for the data sent to (received from) the Internet can be described as follows:

$$\text{Overhead}_{int} = (1000 + 20 + 20 + 30) * 100\% / 1000 = 107\% \quad (5)$$

In a similar way, there can be defined the total overhead for VoIP data, assuming that the length of the data after encoding is about 40 bytes [1]. RTP overhead is 12 bytes and UDP overhead (the protocol does not use confirmation during transmission) is 8 bytes and overhead Ethernet protocol is 30 bytes in relation to transmitted data (IP packets from the network layer) [3]. The percentage of the total overhead for the VoIP can be defined as before:

$$\text{Overhead}_{voip} = (40 + 12 + 8 + 20 + 30) * 100\% / 40 = 275\% \quad (6)$$

The bandwidth in Table 5 should be adjusted after determining the size of overhead brought by the individual protocols of TCP/IP model for VoIP and Internet connections. Table 5 elements determining the required bandwidth for VoIP ($T_{i,j}$, $i, j = 1, 2, \dots, n, A, B, C, i \neq j$) will be increased by 175%, while the elements that determine the bandwidth of links allowing access to Internet ($T_{i,j}, T_{j,i}, i = MAN, j = 1, 2, \dots, n$) will be increased by 7%. Table 6 shows the required bandwidth taking into account the overhead brought by the individual protocols.

Table 6. The required bandwidth taking into account the overhead brought by the individual protocols in proposed network (kbit/s)

Router/ Router	Bydgoszcz (i = 1)	Chojnice (i = 2)	Koronowo (i = 3)	Żnin (i = 4)	Inowrocł. (i = 5)	A: LDE	B: LE Chojnice	C: LE Inowrocł.	MAN
Bydgoszcz (i = 1)	X	1843	1843	1843	1843	8800	27657	27657	57067
Chojnice (i = 2)	1843	X	1843	1843	1843	8800	27657	27657	57067
Koronowo (i = 3)	1843	1843	X	1843	1843	8800	27657	27657	57067
Żnin (i = 4)	1843	1843	1843	X	1843	8800	27657	27657	57067
Inowrocł. (i = 5)	1843	1843	1843	1843	X	8800	27657	27657	57067
A: LDE	8800	8800	8800	8800	8800	X	X	X	X
B: LE Chojnice	27657	27657	27657	27657	27657	X	X	X	X
C: LE Inowrocł.	27657	27657	27657	27657	27657	X	X	X	X
MAN	171200	171200	171200	171200	171200	X	X	X	X

Figure 5 shows the logical structure of the network, taking into account also the links at the points of connection of the proposed network with the network of PT Co. and MAN Bydgoszcz. It should be noted, however, that the network of PT Co. is a circuit switched network, while the proposed network is based on packet switching, hence, there is a need to use gateways at inter-network points of connection (e.g. on the side of PT Co. network).

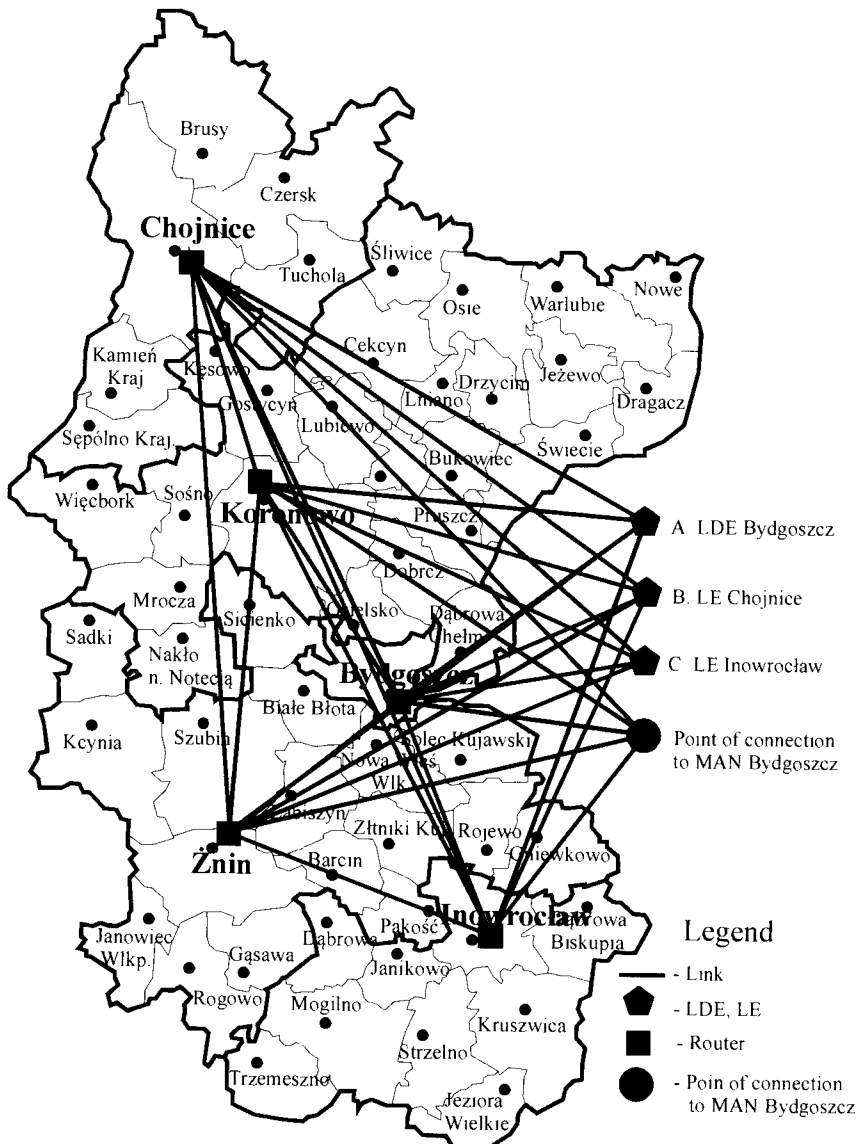


Fig. 5. The logical structure of proposed network

5. PHYSICAL STRUCTURE

Physical structure of the proposed network will be implemented basing on the bi-directional double fiber ring. After determining a bandwidth for each link, the number of virtual containers VC-12, needed to implement these links, should be estimated. For this purpose, the manual of SDH multiplexer TM-160 [2] was used. From this description it follows that the mapping of Ethernet frame 2.16 Mbps can be used in each virtual container VC-12. This means that 50 containers VC-12 (exactly 47 in case of company mapping) is sufficient for transmitting signal (Ethernet frames) of 100 Mbps bandwidth. Thus, the bandwidth determined in Table 6 can be expressed by the required number of containers VC-12.

Table 7. The required size of bandwidth expressed by the number of containers VC-12

Router/ Router	Bydgoszcz (i = 1)	Chojnice (i = 2)	Koronowo (i = 3)	Żnin (i = 4)	Inowrocł. (i = 5)	A: LDE	B: LE Chojnice	C: LE Inowrocł.	MAN
Bydgoszcz (i = 1)	X	1	1	1	1	5	14	14	29
Chojnice (i = 2)	1	X	1	1	1	5	14	14	29
Koronowo (i = 3)	1	1	X	1	1	5	14	14	29
Żnin (i = 4)	1	1	1	X	1	5	14	14	29
Inowrocł. (i = 5)	1	1	1	1	X	5	14	14	29
A: LDE	5	5	5	5	5	X	X	X	X
B: LE Chojnice	14	14	14	14	14	X	X	X	X
C: LE Inowrocł.	14	14	14	14	14	X	X	X	X
MAN	86	86	86	86	86	X	X	X	X

Table 7 shows the required number of containers VC-12 between routers and at inter-network connection points. The dimensioning of the ring can be carried out only now. This ring, as already mentioned bidirectional double fiber, should implement the link between routers (to serve network traffic) as well as links at the points of connection of networks: i.e. links between routers in the proposed network and exchanges in the PT Co. network (exchanges A, B and C) and links between routers and MAN Bydgoszcz. Dimensioning of the ring can be carried out in one of three ways. The first way consists in summing all the needs expressed by the specified number of containers VC-12 between individual pairs of nodes. Since half of the demand (or almost half in the case of odd number of containers VC-12) between a given pair of nodes is performed on one part of the ring, while the second half is performed on complementary part, hence the aggregate number of containers will be the number of containers required for the implementation of a working and backup band (more precisely upper constraint of the band) in the event of failure of the ring. Thus, in the considered example, the aggregate number of routes can be designated as:

$$\sum_{\substack{i=1,2,\dots,n \\ j=1,2,\dots,n \\ i>j}} T_{i,j} + \sum_{\substack{i=A,B,C \\ j=1,2,\dots,n}} T_{i,j} + \sum_{\substack{i=MAN \\ j=1,2,\dots,n}} T_{i,j} \leq 63 \quad (252,1008) \quad (7)$$

This formula is analogous to the formula determining dimensioning of unidirectional double fiber ring. It consists of three components: the first covers bandwidth (symmetric) between the routers, the second covers bandwidth (symmetric as well) implemented at the points of connection between routers of the proposed network and PT Co. network and the third component includes the bandwidth between the routers of the proposed networks and MAN Bydgoszcz. This bandwidth is asymmetric (Table 7, the greater part of traffic is directed from MAN Bydgoszcz to the proposed network than in the opposite direction). However, in the above formula the symmetric bandwidth was adopted. The second way consists in "manual" allocation of the required number of containers VC-12 between pairs of nodes. In this case the band at the point of connection is implemented by a dimensioned ring. The manual dimensioning can reduce the number of virtual containers required to implementing working bandwidth at the ring because it allows to "complement" residual capacity at the circuit of the ring; however, this way of dimensioning is possible only for a small number of nodes [4]. The third way of dimensioning designates the accurate bandwidth at the point of connection to a set of equations and inequalities [4]. The symmetric bandwidth at the point of connection to MAN Bydgoszcz is also adopted in the second and third way of dimensioning.

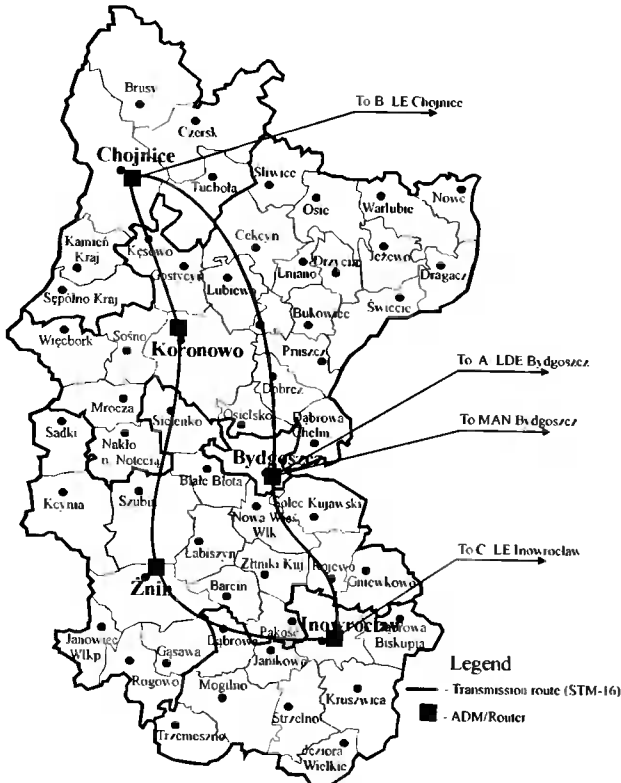


Fig. 6. The physical structure of the proposed network

For the analyzed example, the total number of containers VC-12, determined on the basis of the formula 7, is: $10+165+430 = 605$ VC-12. Therefore, to implement the required bandwidth, the STM-16 transmission system should be used. STM-16 implements 1008 VC-12 containers. (The next two ways of dimensioning is left for students). Figure 6 shows the physical structure of the proposed network.

6. CONCLUSIONS

As mentioned before, the purpose of this paper is to acquaint students with the engineering approach to designing a network over IP platform. The project includes only a backbone layer of the network and omits the access layer. The output data include two classes of services: VoIP and access to the Internet. It should be noted that these services have very different impact on the structure of traffic of the proposed network. VoIP service generates the general structure of traffic that requires a bandwidth between each pair of nodes (routers). However, access to the Internet generates traffic structure of collective node, which requires a bandwidth between each network node and the point of connection to MAN Bydgoszcz. The superposition of these structures defines the logical structure of the network. The physical layer was implemented based on SDH. For the considered variant (which was adopted at the beginning of this paper) SDH ring works as bi-directional double fiber ring with a bandwidth of STM-16 (2488.32 Mbit/s) and meets the requirements concerning the bandwidth. Overhead of individual protocols of TCP/IP model was also taken into account.

Further work, related to the design of the networks will focus on analyzing additional real-time services and greater bandwidth in the access network (to the Internet). In addition, the physical structure of the network will be analyzed taking into account the different technologies: IP / WDM (DWDM) and Ethernet.

BIBLIOGRAPHY

- [1] Cała J., 2001. Organizacja transmisji głosu w sieci IP. Proceedings of the 8th Conference Computer Networks, Krynica, 18 –20 Jun 2001.
- [2] Muplekser SDH TM-160, Instrukcja obsługi, http://www.lanex.pl/dane/pol/IO160-1B_PL.pdf
- [3] Papier Z., 2001. Ruch telekomunikacyjny i przeciążenia sieci pakietowych, WKŁ, Warszawa.
- [4] Pióro M., 1995. Podstawy projektowania cyfrowych sieci telekomunikacyjnych. Francusko-Polska Szkoła Nowych Technik Informatyczno-Komunikacyjnych EFP, Poznań.
- [5] http://www.ventus.pl/cisco/call_manager.php

PROJEKTOWANIE SIECI TELEKOMUNIKACYJNEJ DLA BYDGOSKIEJ STREFY NUMERACYJNEJ

Streszczenie

W artykule przedstawiono inżynierskie podejście do projektowania sieci telekomunikacyjnej na platformie IP. Projektowana sieć pokrywa żądania na usługi telekomunikacyjne, w bydgoskiej strefie numeracyjnej, na obszarze której działa Telekomunikacja Polska (TP S.A.). W projektowanej sieci wyróżniono dwie klasy usług: telefonię internetową (VoIP) oraz dostęp do Internetu. Logiczna struktura sieci jest superpozycją dwóch struktur. Pierwsza z nich zawiera łącza, które realizują telefonię internetową oraz druga, która zapewnia dostęp do Internetu. Fizyczna struktura sieci została zaprojektowana w oparciu o Synchroniczną Hierarchię Cyfrową (SDH).

Słowa kluczowe: ruter, łącze, projektowanie

MODIFIED SLIDING WIENER-KHINTCHIN TRANSFORM

Włodzimierz Pogribny, Dariusz Surma

University of Technology and Life Sciences
7 Prof. Kaliskiego Ave., Bydgoszcz 85-796 Poland
pohry@utp.edu.pl, surma@utp.edu.pl

Summary: The article presents the new approach to increase a speed of Sliding Discrete Wiener-Khintchin Transform (SDW-KT) algorithm on the basis of recurrent correlation analysis (CA) algorithm using. In this case it is not necessary to calculate the whole correlation function by each analyzing window position. There is taken note of analyzing window parameters selecting for sliding DW-KT, FFT, and Sliding Periodogram too. Worked out approach predominance over SFFT and Periodogram is demonstrated on examples of short noisy signal recognition.

Keywords: Time-frequency analysis, Wiener-Khintchin transform, Correlation analysis

1. INTRODUCTION

Random signal analysis in the time domain not always allows to obtain sufficient information about a studied signal. Therefore that analysis is complemented by researches in the frequency domain. In this context Discrete Fourier Transform (DFT) is a basic tool. Its algorithm is as follows [2]:

$$X(k) = \frac{1}{N} \sum_{n=0}^{N-1} x_n \exp(-j \frac{2\pi}{N} nk), \quad \text{for } k, n = \overline{0, N-1} \quad (1)$$

where:

- $X(k)$ – k-th term of a Fourier's complex spectrum;
- $\{x_n\}_{n=0}^{N-1}$ – signal samples, where $x_n = x(nT_s)$;
- T_s – sampling period;
- N – signal length.

Because of this popular method needs too much operations – about N^2 complex multiplications and N additions – instead of it the fast algorithms (FFT) are used which need about $N \log_2 N$ those operations that allows to use FFT in real time systems [2], [3]. But an immediate usage of “motionless” DFT (1) and FFT to non-stationary signal processing is ineffective because of it does not show signal spectrum changes in time. In opposite to it, the sliding algorithms of DW-KT, Periodogram and FFT have not this

downside. However, the correlation analysis which is a base of DW-KT needs many time – consuming operations that limits SDW-KT usage in real time signal processing systems.

Purpose of this work is to compare the sliding algorithms of FFT, Periodogram and DW-KT as well as increase a speed of SDW-KT algorithm on the basis of worked out new recurrent correlation analysis algorithm, select analyzing window parameters and verify the proposed approaches on the basis of short noisy signal recognizing.

2. SLIDING DFT AND PERIODOGRAM ALGORITHMS

Analyzing window of SDFT is being moved along of a signal realization in the next way [4], [5]:

$$X(k,r) = \frac{1}{N_w} \sum_{n=0}^{N_w-1} x_{n+r} \exp(-j \frac{2\pi}{N_w} nk), \text{ for } r = \overline{0, N - N_w - 1}, n = \overline{0, N_w - 1} \quad (2)$$

where:

- r – analyzing window position;
- $X(k,r)$ – k -th Fourier spectrum term at r -th window position;
- N_w – window samples number.

SDFT method resolution (f_{analyz}) [2] depends on N_w and T_s , as $f_{analyz} = 1/T_s N_w$, therefore it is not always sufficient for short non stationary signal analysis. Periodogram is often used to random signal analysis and is built on FFT base:

$$\hat{S}_{xx}^{(P)}(k) = \frac{1}{N} |X(k)|^2, \quad (3)$$

Usually the Welch procedure is applied to a periodogram in order to improve its accuracy. It boils down to averaging a few periodograms on next parts of a signal realization, that limits its use in a real time analysis. Therefore we can propose using the SFFT algorithm (2) to obtain the sliding periodogram:

$$\hat{S}_{xx}^{(P)}(k,r) = \frac{1}{N_w} \left| \sum_{n=0}^{N_w-1} x_{n+r} \exp(-j \frac{2\pi}{N_w} nk) \right|^2, \quad (4)$$

which allows to analyze a signal in a sample by sample regime in opposite to classical periodogram algorithm when a window steps from subset to subset.

3. SLIDING DW-KT WITH USING OF RECURRENT CA ALGORITHM

Discrete variant of the „motionless” Wiener-Khintchin Transform can be presented in the next form [1], [3]:

$$\hat{S}_{xx}(k) = \frac{1}{2P-1} \sum_{m=-(P-1)}^{P-1} \hat{R}_{xx}(m) \exp(-j \frac{2\pi}{2P-1} mk), \quad (5)$$

where:

$\hat{S}_{xx}(k)$ – estimator of k- th term (component) of a spectrum;

$\hat{R}_{xx}(m)$ – correlation function (CF) estimator at m-th time shift;

PT_x – correlation interval, at that $P \leq N$, where $m = \overline{-(P-1), (P-1)}$.

In turn, estimator of „motionless” CF is defined as follows:

- unbiased estimator

$$\hat{R}_{xx}(m) = \begin{cases} \frac{1}{N-m} \sum_{n=0}^{N-m-1} x_n \cdot x_{n+m} & m \geq 0 \\ \hat{R}_{xx}^*(-m) & m < 0 \end{cases} \quad (6)$$

- biased estimator

$$\hat{R}_{xx}(m) = \begin{cases} \frac{1}{N} \sum_{n=0}^{N-m-1} x_n \cdot x_{n+m} & m \geq 0 \\ \hat{R}_{xx}^*(-m) & m < 0 \end{cases} \quad (7)$$

where:

$\overset{\circ}{x}_n = x_n - \bar{x}$ – centered value;

$\bar{x} = \frac{1}{N} \sum_{n=0}^{N-1} x_n$ – mean value;

\hat{R}_{xx}^* – complex conjugated CF.

In many practical applications the Blackman-Tukey method, when the product of the CF estimator and the suitable smoothing window is used [6], Common downside of those algorithms is lack of possibility to use them in the time-frequency analysis immediately. Therefore in this context it is expedient to develop existed methods.

3.1. SLIDING DW-KT DIRECT ALGORITHM

We propose that the algorithm was based on connection of sliding CA and FFT. In this case analyzing window moves by sample by sample but does not step by window by window. At that, CF is calculated at each position of window and next a power spectrum of this part of signal realization is calculated with using of FFT. It allows to obtain a time-frequency map of an analyzed process. Sliding direct DW-KT algorithm can be written as

$$\hat{S}_{xx}(k, r) = \frac{1}{2N_w - 1} \sum_{m=-(N_u-1)}^{N_u-1} \hat{R}_{xx}(m, r) \exp(-j \frac{2\pi}{2N_w-1} mk), \quad (8)$$

$$\text{for } r = 0, N - N_w - 1, m = -(N_u - 1), (N_u - 1)$$

where:

$\hat{S}_{xx}(k, r)$ – power spectrum estimator of k-th component at r-th position of the window.

At that, CF estimator can be presented as unbiased estimator (9) or biased one (10).

$$\hat{R}_{xx}(m, r) = \begin{cases} \frac{1}{N_w - m} \sum_{n=0}^{N_u - m - 1} x_{n+r} \cdot x_{n+r+m} & m \geq 0 \\ \hat{R}_{xx}^*(-m) & m < 0 \end{cases} \quad (9)$$

$$\hat{R}_{xx}(m, r) = \begin{cases} \frac{1}{N_w} \sum_{n=0}^{N_u - m - 1} x_{n+r} \cdot x_{n+r+m} & m \geq 0 \\ \hat{R}_{xx}^*(-m) & m < 0 \end{cases} \quad (10)$$

where all meanings are analogous to expression (7).

3.2. WORKED – OUT FAST RECURRENT CORRELATION ANALYSIS ALGORITHM FOR SDW-KT

We can write down the Sliding CA algorithm for r-th position of the window in next form:

$$\hat{R}_{xx}(m, r) = \frac{1}{N_w - m} \sum_{n=0}^{N_u - m - 1} x_{n+r} \cdot x_{n+r+m} \quad (11)$$

for all positions of window $r = 0, N - N_w - 1$ and time shifts $m = -(N_u - 1), (N_u - 1)$

Here:

$\hat{R}_{xx}(m, r)$ – CF estimator at r-th position of window
and for (r-1) – th position of the window

$$\hat{R}_{xx}(m, r-1) = \frac{1}{N_w - m} \sum_{n=0}^{N_u - m - 1} x_{n+r-1} \cdot x_{n+r-1+m} \quad (12)$$

For obtaining the recurrent algorithm, it is necessary to apply backward differences to CF by adjacent positions of a window:

$$\nabla_r \hat{R}_{xx}(m, r) = \hat{R}_{xx}(m, r) - \hat{R}_{xx}(m, r-1) \quad (13)$$

Then differential CF is:

$$\begin{aligned} \nabla_r \hat{R}_{xx}(m, r) &= \frac{1}{N_u - m} \left[\sum_{n=0}^{N_u - m - 1} x_{n+r} \cdot x_{n+r+m} \right] - \frac{1}{N_w - m} \left[\sum_{n=0}^{N_u - m - 1} x_{n+r-1} \cdot x_{n+r-1+m} \right] = \\ &= \frac{1}{N_w - m} \left[x_{1+r} \cdot x_{1+r+m} + x_{N_u - 1 + r - m} \cdot x_{N_u - 1 + r} - x_{r-1} \cdot x_{r-1+m} \right], \end{aligned}$$

where for $r < 1$ it takes place $x_{r-1} \cdot x_{r-1+m} = 0$.

Hence the CA recurrent algorithm (for $r \geq 1$ only):

$$\begin{aligned} \hat{R}_{xx}(m, r) &= \hat{R}_{xx}(m, r-1) + \nabla_r \hat{R}_{xx}(m, r) = \\ &= \hat{R}_{xx}(m, r-1) + \frac{1}{N_w - m} \left[x_{1+r} \cdot x_{1+r+m} + x_{N_u - 1 + r - m} \cdot x_{N_u - 1 + r} - x_{r-1} \cdot x_{r-1+m} \right] \end{aligned} \quad (14)$$

where:

$$x_{r-1} \cdot x_{r-1+m} = 0 \quad \text{for } r < 1 \text{ by the same initial conditions.}$$

Hence SDW-KT on the basis of the recurrent CA algorithm (14) is:

$$\begin{aligned} \hat{S}_{xx}(k, r) &= \frac{1}{2N_w - 1} \sum_{m=-(N_u-1)}^{N_u-1} \left\{ \hat{R}_{xx}(m, r-1) + \right. \\ &\left. + \frac{1}{N_w - m} \left[x_{r+1} \cdot x_{r+m+1} + x_{N_u+r-m-1} \cdot x_{N_u+r-1} - x_{r-1} \cdot x_{r+m-1} \right] \right\} \exp(-j \frac{2\pi}{2N_w - 1} mk) \end{aligned} \quad (15)$$

We can see that it is not necessary to calculate the whole correlation function by each analyzing window position. There is enough to add only averaged sums of 3 products of suitable samples to the CF at previous position of the window.

4. EXAMPLES OF PROPOSED ALGORITHMS APPLICATION

Worked out algorithms were verified on examples of short noisy signals recognition. It was realized with using a worked out program and MATLAB where were assigned aperture $N = 1024$ samples and sampling rate $f_s = 8000$ Hz. Compound analyzed signal consisted of two sine parts and broadband LFM signal (chirp). Frequency of the sine signal was 2000 Hz and amplitude was $A = 1$. Character of the chirp was as follows:

$$x_n = x(nT_s) = A \cos \left[2\pi \left(\frac{\Delta f}{2N_{ch}} n + f_1 \right) nT_s + \varphi_0 \right] \quad (16)$$

by next parametrs: $\{x_n\}$, $n = \overline{257, 384}$, $N_{ch} = 128$ – the chirp samples number, $\Delta f = f_2 - f_1$ – the frequency increase, $f_1 = 2000 \text{ Hz}$ – initial frequency, $f_2 = 4000 \text{ Hz}$ – final frequency, φ_0 – initial phase. The chirp duration (τ_{ch}) was 16 ms, $A = \sqrt{2}$, and $BT = \Delta f \tau_{ch} = 32$. A frequency- time characteristic of the signal is shown in Fig. 1.

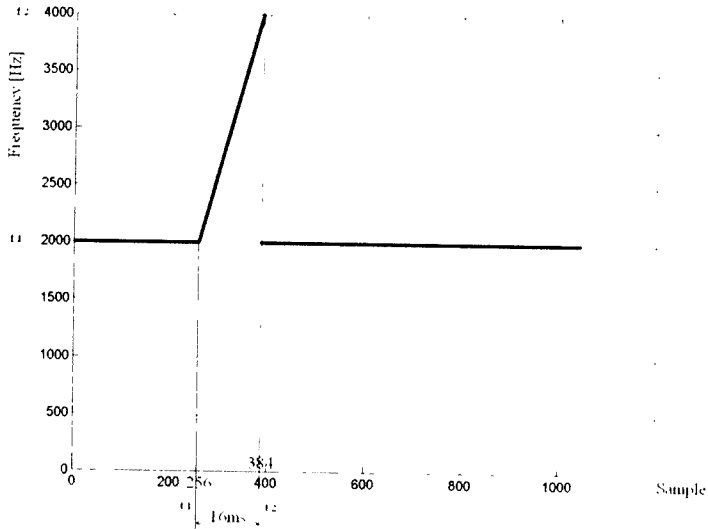


Fig. 1. Frequency-time characteristic of studying signal

This whole signal realization is shown in Fig. 2a and its motionless FFT result – in Fig. 2b. By that there can be done wrong decision that all signal is monochrome whilst it concludes the chirp which amplitude is bigger than one of the sine signal.

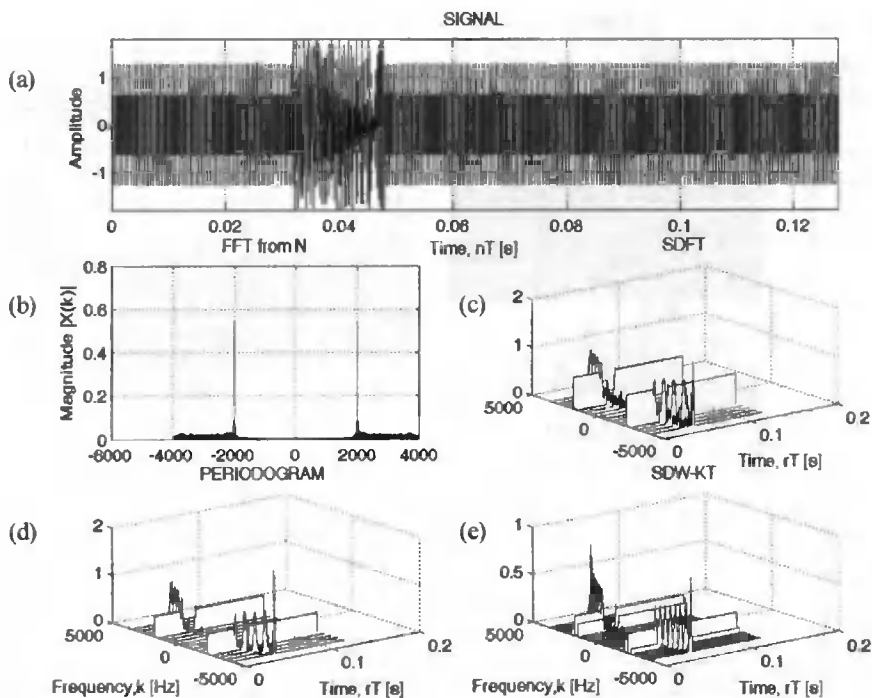


Fig. 2. (a) input noiseless signal, $N = 1024$. (b) "motionless" FFT result of the whole signal realization. A spatial view: (c) SDFT result (d) SPERIOGRAM result. (e) SDW-KT result. For (c), (d) and (e) the window was $N_w = 16$

Analysis of SDFT, Sliding PERIODOGRAM and SDW-KT algorithms was fulfilled for next window length: 8, 16, 32, 64, and 128 samples. These windows slid along a signal realization in a sample by sample regime. Longer windows allow to obtain a better frequency resolution but a worse time one. Note that SDW-KT has twice as big frequency resolution in comparison to SDFT and PERIODOGRAM for the same window length without losing of the time resolution. This fact takes place because of CA, which is part of DW-KT, according to (7), (9), and (10) leads to obtain a double number of CF samples and, accordingly, double number of the power spectrum components (Figs. 2b, 2c, and 2d). Fig. 3 shows the same results but using a view on a plane.

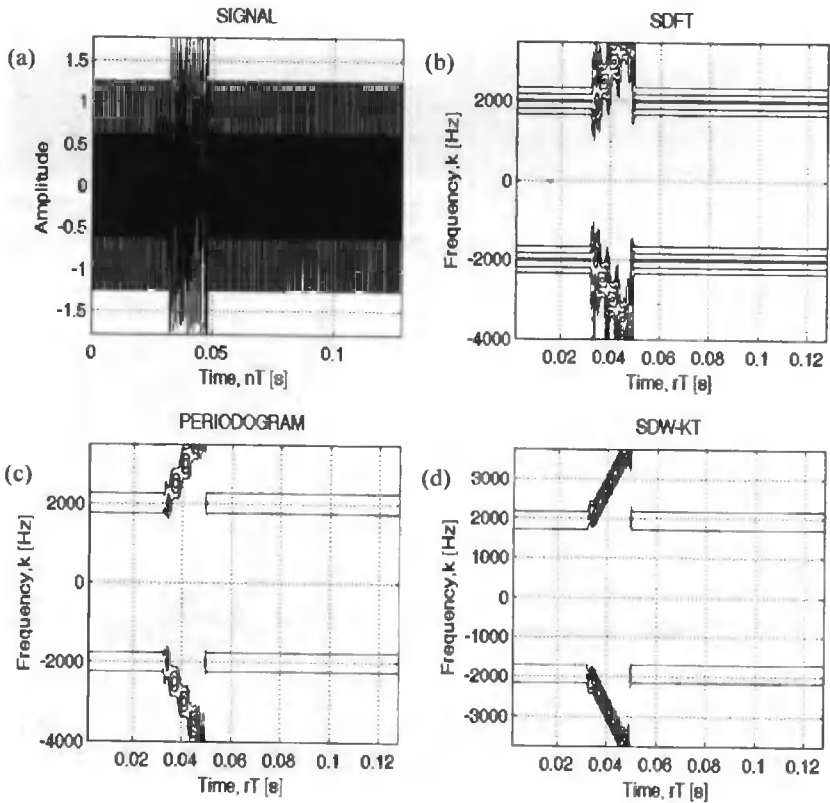


Fig. 3. (a) input noiseless signal, length is $N = 1024$. The view on a plane: (b) SDFT result, (c) Sliding PERIODOGRAM result, (d) SDW-KT result. For (b), (c), (d) the window was $N_w = 16$

As Figures 2 and 3 show, above mentioned methods not only allow to see a signal changes character but determine their appearance moment as well. The best result was obtained for SDW-KT which has twice as big a frequency resolution without worsen a time one.

Next, there was analyzed the SDFT, SPERIODOGRAM and SDW-KT usefulness for recognizing of short noisy signals. At that the computer Gauss' noise „randn” was added according to next SNRs: +10 dB, +3 dB, and 0 dB. Here:

$$SNR = 10 \lg \frac{\Psi_x^2}{\Psi_\xi^2}, \quad (17)$$

and a signal and a noise powers are:

$$\Psi_x^2 = \frac{1}{N} \sum_{n=0}^{N-1} x_n^2, \quad \Psi_\xi^2 = \frac{1}{N} \sum_{n=0}^{N-1} \xi_n^2$$

The noisy signal realization by $SNR = 0 \text{ dB}$ is given in Fig. 3a and Fig. 4a. Other figures show the examples of the signal processing on the basis of SDFT, SPERIDODOGRAM, and SDW-KT (the spatial views and views on a plane accordingly).

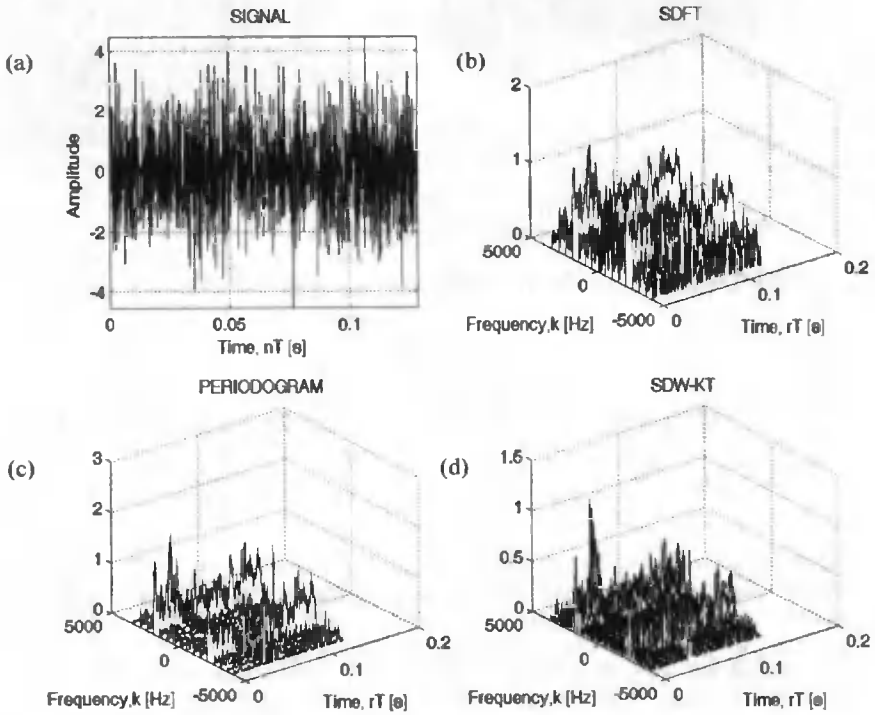


Fig. 4. (a) input noisy signal. The spatial view: (b) SDFT result, (c) SPERIDODOGRAM result, (d) SDW-KT result. Here signal length is $N = 1024$, the window is $N_w = 16$, $SNR = 0 \text{ dB}$ by using the „randn” noise

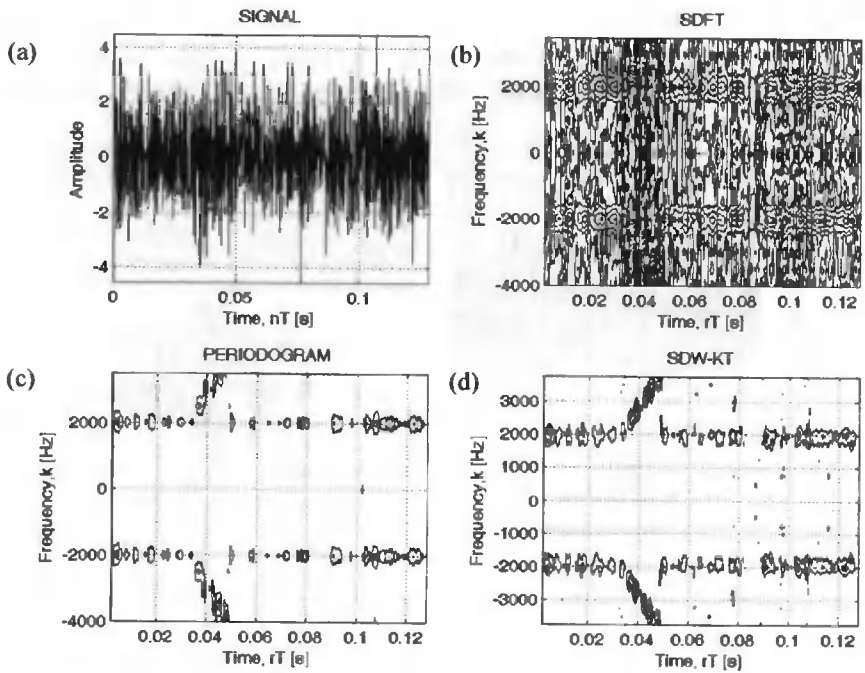


Fig. 5. (a) input noisy signal. The view on a plane: (b) SDFT result, (c) SPERIOGRAM result, (d) SDW-KT result. Here $N = 1024$, $N_w = 16$, $SNR = 0$ dB by using the „randn” noise

As we can see, the best result is obtained for SDW-KT when the appearance moment of the noisy chirp signal is the most expressive (Fig. 5d).

5. CONCLUSION

A comparison between the sliding algorithms of FFT, Periodogram and Discrete Wiener-Khintchin Transform (SDW-KT) is made in this work.

Apart from it, the article presents the new approach to increase a speed of SDW-KT algorithm on the basis of fast recurrent correlation analysis algorithm using. In this case it is not necessary to calculate the whole correlation function by each analyzing window position but there is enough to add only averaged sums of 3 products of suitable samples to the CF at previous position of the window.

There is taken note of analyzing window parameters selecting for the sliding algorithms. Worked out approach predominance over SFFT and Sliding Periodogram is demonstrated on examples of short noisy signals recognition.

BIBLIOGRAPHY

- [1] Bendat J.S., Piersol A.G., 1976. Metody analizy i pomiaru sygnałów losowych, PWN, Warszawa.
- [2] Layons R.G., 1999. Wprowadzenie do cyfrowego przetwarzania sygnałów, WKŁ, Warszawa.
- [3] Otnes R.K., Enochson L., 1978. Analiza numeryczna szeregów czasowych, WNT, Warszawa.
- [4] Pogribny W., 2002. Sliding FFT differential algorithms. Proceedings of the 4th International Conference on Quality, Reliability, and Maintenance, Oxford, 77-80.
- [5] Pogribny W., Surma D., 2008. Sliding discrete Fourier transform in differential format, Information Extraction and Processing, National Academy of Sciences of Ukraine, 48-52.
- [6] Zieliński T.P., 2007. Cyfrowe przetwarzanie sygnałów – od teorii do zastosowań, WKŁ, Warszawa.

ZMODYFIKOWANE ŚLIZGAJĄCE PRZETWARZANIE WIENERA-CHINCZYNA

Streszczenie

W artykule przedstawiono opracowany nowy szybki rekurencyjny algorytm, ślizgającej dyskretnej analizy korelacyjnej (SDW-KT- Sliding Discrete Wiener-Khinchin Transform), pozwalający na istotne zmniejszenie ilości operacji przy kolejnych przemieszczeniach okna analizy, co prowadzi do zwiększenia szybkości przetwarzania SDW-KT. Zwrócono także uwagę na dobrane parametry okien dla ślizgających przetwarzań DW-KT, FFT (Fast Fourier Transform) i PERIODOGRAMU. Przewagi opracowanych podejść nad SFFT i ślizgającym periodogramem przedstawiono na przykładach wykrywania krótkotrwałych zaszumionych sygnałów.

Słowa kluczowe: analiza czasowo-częstotliwościowa, przetwarzanie Wienera-Chinczyzna, analiza korelacyjna

FPGA-BASED SYSTEM FOR IMAGE PROCESSING IN HIGH RESOLUTION INFRARED CAMERA

Tomasz Sosnowski, Grzegorz Bieszczad, Mariusz Kastek, Henryk Madura

Military University of Technology
Institute of Optoelectronics
2 Gen. Sylwestra Kaliskiego Str.
Warsaw, Poland
tsosnowski@wat.edu.pl

Summary: In article a digital system for high resolution infrared camera control and image processing is described. The camera is built with use of bolometric focal plane array of size 640 by 480 detectors. Designed module controls the microbolometer Focal Plane Array (FPA), performs non-uniformity correction, bad pixel mapping and controls the process of displaying the thermal image. The system was designed in such a way, that signal processing algorithms, needed for specific tasks, can be implemented in it without hardware modifications. It was achieved by the application of a FPGA device and microprocessor unit, which both can be re-programmed inside the system. This scientific work is funded as a development project from science funds for years 2009-2011.

Keywords: thermal imaging, signal processing, image processing, FPGA

1. INTRODUCTION

Thermal cameras are more and more often used as observation devices in numerous areas, like security systems, military systems, reconnaissance, detection of chemical agents and many more. In all of the applications it is very important to obtain the accurate image of observed scenery. Because thermal cameras are quite common, they should be easy to operate. As a result, the methods for automatic analysis and processing of thermal image have to be implemented, which simplify the user interface by automatic setting of camera parameters. The applied methods should also enhance the capabilities of a thermal camera in detection and recognition of certain objects and phenomena, making it a more versatile tool than just an observation device. The actual methods implemented in a given device depend on the specific application and the type of analyzed data [1], therefore they couldn't be universal ones or chosen once and for all. Additionally, often the systems for thermal image processing and analysis must have compact size and low power requirements.

In high resolution cameras a robust and efficient image processing system is needed. In handheld and mobile systems the power consumption of such processing system is also a key parameter. Additionally a use of modern infrared image processing module in various types of cameras and applications should be possible without appreciable hardware changes. In infrared image processing module used in high resolution camera,

an amount of data to be processed is significant, but it has to be processed in well defined and short time. The processing delays of image processing modules have to be constant and low, and cannot accumulate in time. Low latency of operations can be obtained with use of parallelization techniques [11]. The amount of a data needed to be processed in thermal camera depends on the resolution of the camera and the frame rate. In Table 1 there is a comparison of data stream volume for three different standard video resolutions and several frame rates.

Table 1. Data stream volume and a maximum time of one pixel(detector) readout time for chosen frame rates and resolutions

Resolution	Pixel (detector) count in frame	Pixel count read in one second and maximum detector readout time			
		25 Hz	30 Hz	50 Hz	60 Hz
320×240	76 800	1 920 000	2 304 000	3 840 000	4 608 000
		520.83 ns	434.03 ns	260.42 ns	217.01 ns
384×288	110 592	2 764 800	3 317 760	5 529 600	6 635 520
		361.69 ns	301.41 ns	180.84 ns	150.70 ns
640×480	307 200	7 680 000	9 216 000	15 360 000	18 432 000
		130.21 ns	108.51 ns	65.10 ns	54.25 ns

Generally speaking, electronic circuitry of a thermal camera consists of three basic modules [2]: focal plane array module, control and image processing module, display module. General, simplified block diagram of thermal camera electronics is presented in Fig. 1. There is an A/D converter inside focal plane array proximity board, which transforms raw image signals into digital data, transferred further to control and image processing module. There are also analog power supplies with proper filtering and noise suppression circuits, providing all main voltages for Focal Plane Array (FPA).

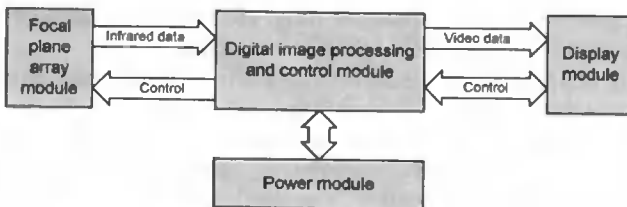


Fig. 1. Block diagram of an uncooled thermal camera

Control and image processing module is responsible for proper Focal Plane Array operation. For proper FPA operation a series of digital and analog signals has to be generated. Digital signals are used to set timing parameters of array, while analog signals are used to bias arrays internal readout circuit with proper voltage levels [3]. Maintaining proper levels of supplying voltages is very important when there is a need to work in the broad temperature range. Those voltages mostly determine the sensitivity of the detectors and the level of output signal. The voltages are driven by automatic microcontroller-based control unit. Microprocessor reads the analog signal from an array by

A/D converter, analyzes it and adjusts the voltage levels accordingly. The voltages are set by D/A converter, which communicates with the microprocessor over SPI bus.

Display module visualizes the video data in VGA format on CRT monitor, LCD or OLED display. RED, GREEN, BLUE video signals and line synchronization (HS) and frame synchronization (VS) pulses are generated by control and image processing module. Video signal is calculated from analog array signals converted to digital form, taking into account calibration coefficients stored in FLASH memory of control and image processing module. The digital video data are then converted back to analog form by a special block consisting of three D/A converters.

2. CONTROL AND IMAGE PROCESSING MODULE

The function of control and image processing module is to control all other camera blocks and to convert the data from detector array. The main operations performed by this module are: control of array module in order to read-out all the detectors, perform non-uniformity correction, bad pixel mapping and prepare data for the display module.

Control and image processing module was built around two main components: FPGA device and a microcontroller. FPGA device performs image data processing, which requires considerable amount of processing power. It deals with the generation of signals controlling array read-out, correction of the non-uniformity of detector response across the array, mapping of bad pixels and generation of input signals for display module. Microcontroller unit controls the camera and performs other tasks that do not require much processing power. Block diagram of control and image processing module with FPGA device is shown in Fig. 2.

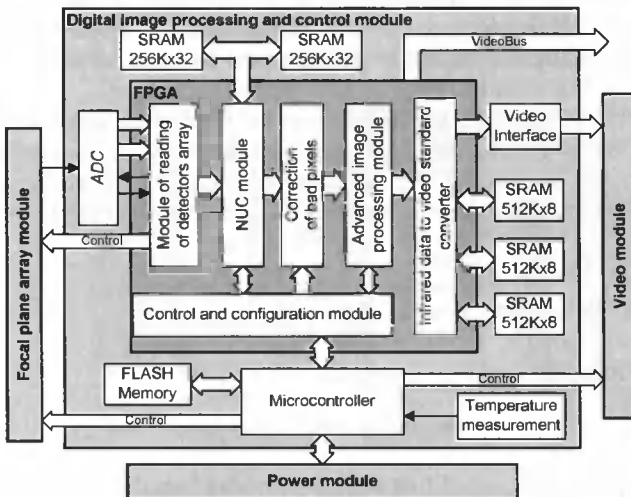


Fig. 2. Functional diagram of FPGA-based control and image processing module

The main role of a microcontroller is to supervise every stage of image processing. Its main tasks are:

- controlling the FPA supply voltages,

- FPA temperature measurement,
- ambient temperature measurement,
- shutter control,
- controlling of auto-calibration process,
- display configuration,
- modification of NUC (NonUniformity Correction) parameters,
- calculation of parameters needed to correct NUC coefficients.

To realize these tasks the STM32F103 microcontroller from STMICROELECTRONICS was chosen. This integrated circuit is a member of 32-bit ARM cortex family. The architecture of the cortex processor is designed to execute 32-bit program code with high efficiency. This type of microcontroller from STM has rich peripherals like 12-bit analog-to-digital converter, 32 timers and communication modules compatible with standards like SPI, I2C, CAN and UART. What is more, a STM32F103 microcontroller is rich in general purpose input-output ports and external memory interface. The external memory interface was used to access peripherals like memory or FPGA.

The EP2K35F672 device by ALTERA is the chosen FPGA type. This device combines processing efficiency with relatively small power consumption. It has a large number of I/O ports, 33 216 logic elements (LEs), 483 840 bits of RAM memory, four PLL circuits for the generation of required signals and 35 embedded hardware multipliers. It should be mentioned that those multipliers make it possible to design functional blocks in FPGA device capable of high-speed complicated calculations with low usage of logic elements and relatively small power consumption.

In every video processing system there is vast amount of data that has to be processed in a given time period [9, 11]. Those data have to be processed in real time without significant delays. It should be pointed out that because of continuity of image processing the introduced delays should be constant, no increase of delays over time is allowed. This means that parallel image processing has to be used.

In order to obtain useful infrared image from the focal plane array a series of image processing algorithms has to be used beginning with detectors nonuniformity correction, bad pixel replacement and ending with image displaying. To realize these image processing tasks in real-time, the following modules were implemented in FPGA device:

- FPA read-out,
- NUC correction module [4],
- bad pixel mapping module,
- image processing module,
- image display module.

2.1. VIDEOBUS

The VideoBus communication bus was implemented in FPGA for image data transfer between all functional blocks contained inside FPGA. As a result the execution order of image processing may be altered without the need to change the functional blocks. This provides considerable flexibility in the design of thermal image processing system. The VideoBus is accessible by a separate connector, which allows monitoring the operation of digital image processing module at every stage of image processing.

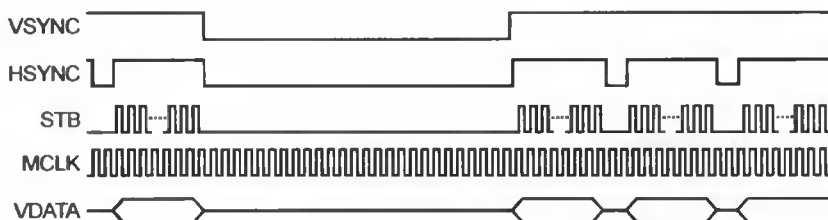


Fig. 3. VideoBus timing diagram

Image data bus consists of 14-bit data bus, vertical (VSYNC) and horizontal (HSYNC) synchronization signals and strobe signal (STB). Signal VSYNC is high during the transmission of the image frame, whereas HSYNC signal is high during consecutive row read-outs. The change of STB signal from low to high (rising edge) indicates that VDATA data bus contains valid (stable) video data for read-out. Those data are kept on VDATA bus as long as STB signal is high. Time characteristics of VideoBus data signals are presented in Fig. 3.

2.2. FPA READ-OUT MODULE

The main tasks of detector array readout module is generation of signals driving internal circuitry of detector array and interfacing with analog to digital converter connected to it. To achieve this there is a need to design a special digital machine which generates a series of signals connected to array via external IO interface. As a result sequence of signals from FPGA, the detector array starts to send analog signals read from consecutive detectors. This signal is a response of the detector and is proportional to the incident infrared radiation. FPA read-out module logic controlling readout process was realized as functional block in FPGA device. Its block diagram is presented in Fig. 4.

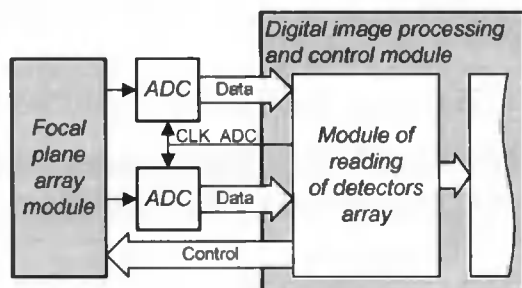


Fig. 4. Schematic of detector array read-out module

High resolution infrared arrays (640x480 and higher) have commonly more than one output, providing simultaneous readout from several detectors. For this type of arrays there is a need to interface to all outputs using several synchronized analog-to-digital converters. The digital signals to microbolometric array have two purposes. One is to provide proper timing during readout from detectors. To do this in UL 04 17 1 detector array a RESET, INT, MC are used [9]. These digital signals provided to array allow an access to analog signals from the detectors. The signals can be then read synchronously with CLK_VIDEO ADC.

The control circuit was designed to read out the data from all array detectors. The purpose of FPA read-out unit is to obtain digital representation of an analog detector output signal, proportional to the incident IR radiation. The data are then transmitted to the next camera modules over image data bus (VideoBus). Further data processing requires the conversion from analog VIDEO signal into digital form (DVIDEO). This conversion is performed by an A/D converter, which requires clock signal (CLK_VIDEO) for the controlling of the conversion process. Result of signal generation in detector array read-out module is shown on Figure 5.

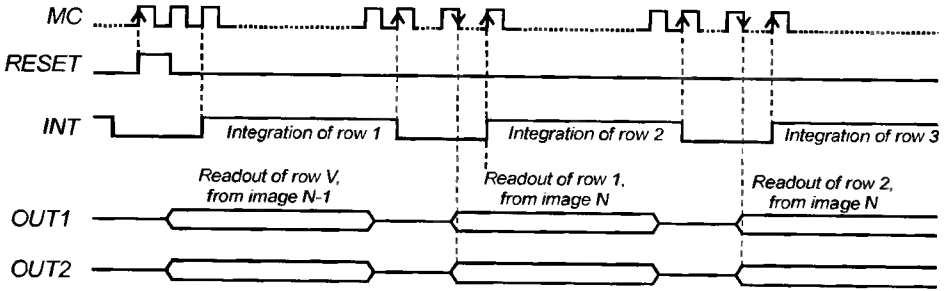


Fig. 5. Timing diagram of control signals for ULIS UL 04 17 1

To generate synchronization signals a module was realized with use of 5-state machine and two counters one for counting clock cycles during row readout, second to count rows in the frame. First counter is used to generate INT signal for row integration while second is used to generate RESET signals during frame initialization. These two counters are also used in generation of VideoBus signals like: VSYNC, HSYNC, STD, MCLK, VDATA0 and VDATA1. The characteristic feature of UL 04 17 1 microbolometer array is, that the signals on the output of array are delayed by the time of one row from the integration trigger signal. This is because of internal structure of the array; first the detectors are integrated and then are read and sent to external system. Example signals generated for UL 04 17 1 microbolometer array are shown on Fig. 6.

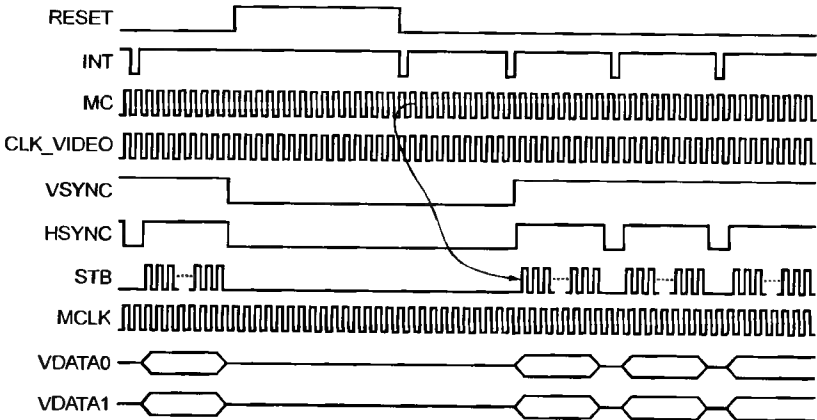


Fig. 6. Timing diagram of control signals for ULIS UL 03 04 1 microbolometer FPA with A/D converter latency taken into account

Used in this solution ADC converter AD9251 from ANALOG DEVICES has its own latency needed to process incoming analog signal equal to 9 clock cycles. This latency has to be taken into account in generation of VideoBus signals in this first module.

2.3. BAD PIXEL MAPPING MODULE

In every real array there are bolometers that do not function properly. The correction for those bad pixels is an important problem. Bad pixels are identified at the thermal camera calibration stage on the basis of their response. The data containing bad pixel locations are stored in memory. Each pixel is described by corresponding one-bit data, where "1" means bad pixel, which has to be corrected, and "0" means good one and no correction is necessary. The content of bad pixel memory determines if the pixel output is displayed, or not. In case of a bad pixel the value from the previous good one is displayed instead. The simplified diagram of the operation of bad pixel mapping module is shown in Fig. 7.

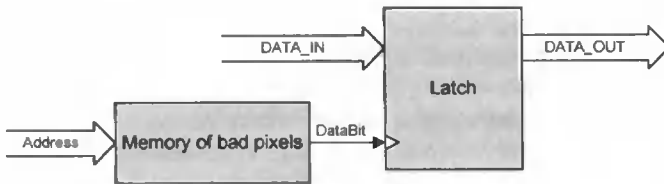


Fig. 7. Scheme of operation for bad pixel mapping module

2.4. IMAGE DISPLAY MODULE

To obtain optimised image quality of infrared image, a specially selected integration time of detector has to be used for readout procedure. This implies that the detector can generate a fixed frame rate which is different than frame rate of standard video output. A mismatch between data read-out frequency and output video frame rate can be overcome with use of special frame rate converter. In case of thermal camera the actual frame rate of video output is often higher than FPA read-out frequency. A special module has to be introduced, which is schematically presented in Fig. 8.

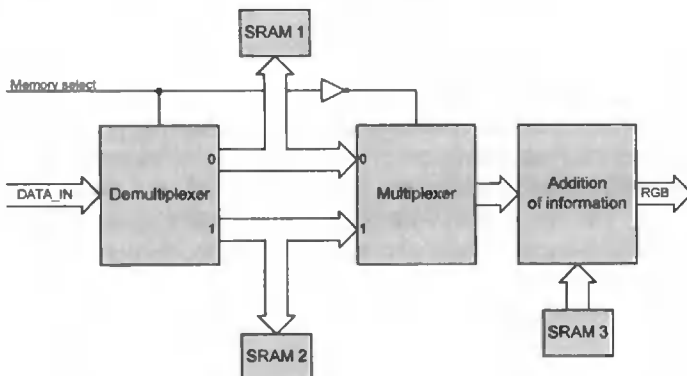


Fig. 8. Block diagram of image display module with additional information overlaying capability

Image display module uses two memories SRAM 1 and SRAM 2. The pixel signals are stored in one of them (e.g. SRAM 1). In the same time the data to be displayed are read from the other memory (in this case SRAM 2). When all of the image pixel data are read from an array, the Memory select signal switches the roles of SRAM 1 and SRAM 2 memories. Now data to be displayed are read from SRAM 1 and pixel data are written into SRAM 2.

Other function performed by image display module is overlaying additional information onto the output video stream. This superimposed information contains user interface controls and camera status data, for example battery level.

Image display module contains also VGA controller, which sends control signals to the display.

2.4. NUC CORRECTION MODULE IN FPGA

Microbolometer detector array exhibits certain non-uniformity of response of detectors to the same power of incident IR radiation [2, 5, 6, 8, 12]. Because of this non-uniformity a fixed pattern noise (FPN) can be observed in the output image, which degrades spatial Noise Equivalent Temperature Difference (NETD) of a thermal camera. Typical non-uniformity level for microbolometer arrays is about 8-10% (std/mean).

There are several methods for non-uniformity correction and most of them rely on digital processing of array output signal. The correction data, so-called NUC coefficients are determined during camera calibration against uniform IR sources.

Basic method of NUC correction is TPC (two-point correction) procedure. TPC algorithm is conducted according to the following formula [8, 14]:

$$N_{ij}^* = G_{ij}N_{ij} + O_{ij}, \quad (1)$$

where N_{ij} is the response of the detector at array coordinates (i, j) , G_{ij} and O_{ij} are the correction coefficients for gain and offset, respectively, and N_{ij}^* is corrected value of detector output signal. In case of TPC corrections the coefficients are given by [8, 14]:

$$G_{ij} = \frac{N(T_H) - N(T_L)}{N_{ij}(T_H) - N_{ij}(T_L)}$$

$$O_{ij} = \frac{N(T_L)N_{ij}(T_H) - N(T_H)N_{ij}(T_L)}{N_{ij}(T_H) - N_{ij}(T_L)}, \quad (2)$$

where $N_{ij}(T_H)$ and $N_{ij}(T_L)$ represent digital values of detector response for high (T_H) and low (T_L) temperature of uniform IR source, $N(T_H)$ and $N(T_L)$ are the mean values of detector response for the temperatures T_H and T_L . Image data bus have width of 14-bit (defined by the resolution of an A/D converter) and NUC correction coefficients are 16-bit [5, 7, 8]. The structure of equation (1) implies that digital circuit for TPC correction has to perform one multiplication and one addition. In case of fixed point numbers the resulting coefficients should be re-scaled in order to increase accuracy. The block diagram of hardware module for TPC correction is shown in Fig. 9

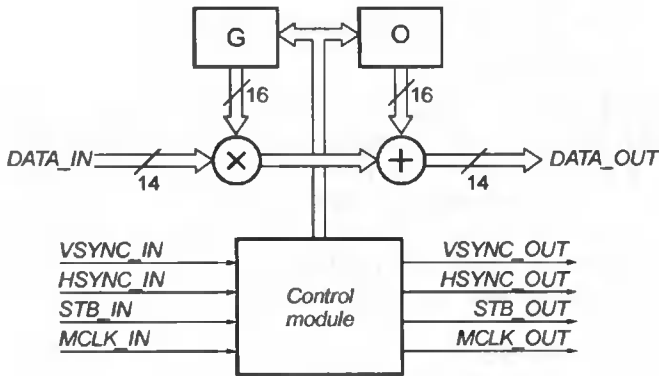
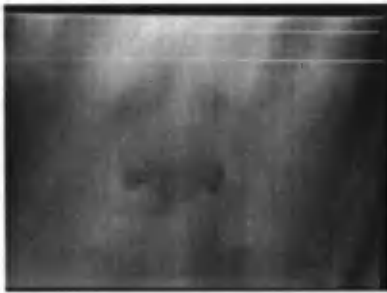


Fig. 9. Block diagram of hardware NUC module

Non-uniformity correction is performed „on the fly”, which means that the time needed to perform the calculations is shorter than single detector read-out time. Additionally the input and output data format for the NUC correction module conforms to the VideoBus standard. Finally the NUC correction module was designed, which performs TPC correction, generates VideoBus output signals and memory address, where correction coefficients are stored.

a)



b)



Fig. 10. Thermal image before (a) and after (b) NUC correction

NUC correction module was described in VHDL language as a parameterized entity. The designer can adjust such parameters as array size (rows and columns) and the length of data word for both input data and calculated correction coefficients. As a results such module can be applied for any given infrared focal plane array. There are two thermal images shown in Fig. 10. First image is obtained without, and the second one after the NUC correction performed by the designed module.

3. SUMMARY

On the basis of functional diagram presented in Fig. 2 the control and image processing module was designed and implemented in a FPGA device. As a result the elec-

tronic board for the read-out and image processing was manufactured on a multi-layer PCB.

The operation of the FPGA-based module was described in VHDL language using several complex finite-state machines. The designed machine use accordingly described internal counters for the switching of their states, by synchronous counting of core clock pulses. VHDL language was also used to define user interface. This interface is used for the setting of camera parameters and for the monitoring of the actual status, using several registers accessible via microprocessor. The additional module that generates simulated A/D converter data was implemented in the project for the testing of array read-out circuit. The project was simulated and then synthesized in ALTERA's Quartus II environment.

Microcontroller circuit was programmed using hybrid method, a mixture of assembler and C programming language. It combined the fast execution of assembler routines with compact code of C procedures in parts where the speed was not critical. The microprocessor with this software successfully controlled the camera with adequate effectiveness.

The simulations and laboratory tests [2, 5] conducted with the use of logic analyzer proved the flawless operation of the designed control and image processing. Simulated timing parameters are in accordance with dynamic parameters described in the specifications of FPA array read-out circuit.

The worst case analysis of timing parameters was also performed for all modules implemented in a FPGA device. Timing analyzer summary revealed that maximal total delay introduced by combinational circuits is 5.548 ns, and theoretical maximum frequency of the VideoBus is 68.1 MHz. These parameters are well above the actual needs as the real frequency of VideoBus in the designed FPA read-out module is only 6.25 MHz.

Additional advantage of the presented design is its flexibility, because the changes in the performed functions and algorithms can be done without hardware modifications. Furthermore, the modules implemented in the FPGA device were described using standard VHDL language, so they can be implemented in any programmable structure.

BIBLIOGRAPHY

- [1] Kastek M., Madura H., Sosnowski T., Polakowski H., 2007. Thermovision method of gas detection in far infrared range, *Advanced Infrared Technology and Applications AITA 9*, Leon.
- [2] Kastek M., Orzanowski T., Sosnowski T., Bareła J., 2007. Test stand for determination of NUC parameters for infrared detector arrays, *Advanced Infrared Technology and Applications AITA 9*, Leon.
- [3] Madura H., 2007. Method of signal processing in passive infrared detectors for security systems, *WIT Transactions on Modelling and Simulation*, 46, pp. 757-768.
- [4] Madura H., Dąbrowski M., Sosnowski T., Trzaskawka P., 2007. Method of automatic recognition of objects flying at low altitudes, *Advanced Infrared Technology and Applications AITA 9*, Leon.
- [5] Bieszczad G., Orzanowski T., Sosnowski T., Kastek M., 2009. Method of detectors offset correction in thermovision camera with uncooled microbolometric focal plane array, *Proceedings of SPIE Vol. 7481*, 748100.

- [6] Orzanowski T., Madura H., Powiada E., Pasierbiński J., 2006. Analysis of readout circuit for a microbolometer focal plane array. *Measurement Automation and Monitoring*, no 9, pp. 16-20.
- [7] Orzanowski T., Sosnowski T., 2006. Implementation of nonuniformity correction algorithms of microbolometer focal plane arrays in FPGA device, *Measurement Automation and Monitoring*, no 11, pp. 8-11.
- [8] Orzanowski T., Madura H., Kastek M., Sosnowski T., 2007. Nonuniformity correction algorithm for microbolometer infrared focal plane array, *Advanced Infrared Technology and Applications AITA 9*, Leon.
- [9] Sosnowski T., Orzanowski T., Kastek M., Chmielewski K., 2007. Digital image processing system for thermal cameras, *Advanced Infrared Technology and Applications AITA 9*, Leon.
- [10] Sosnowski T., Bieszczad G., 2006. Rozpoznawanie obiektów na podstawie analizy obrazu termowizyjnego, VII Krajowa Konferencja Termografia i Termometria w Podczerwieni TTP, Ustroń-Jaszowiec, 6 str. (205÷210) (in polish).
- [11] Wiatr K., 2003. Akceleracja obliczeń w systemach wizyjnych, Wydawnictwa Naukowo-Techniczne, Warszawa (in polish).
- [12] Zhou B., Wang Y., Ye Y., Wu X., Ying J., 2005. Realize multi-point method for real-time correction of nonuniformity of uncooled IRFPA, *Proc. SPIE*, Vol. 5640, pp. 368-375.

MODUŁ DO PRZETWARZANIA OBRAZU Z MIKROBOLOMETRYCZNEJ KAMERY PODCZERWIENI Z ZASTOSOWANIEM UKŁADU FPGA

Streszczenie

W artykule opisano uniwersalny cyfrowy system sterowania i przetwarzania dla kamery termowizyjnej z matrycowym detektorem bolometrycznym rejestrującym promieniowanie w zakresie widmowym w przedziale 8÷12 μm . Najważniejszym zadaniem systemu jest odczytanie sygnałów z poszczególnych detektorów matrycy oraz korekcja wartości wzmocnienia i napięcia przesunięcia charakterystyki czułości dla każdego detektora matrycy. Następnym zadaniem jest przetworzenie analogowych sygnałów z matrycy na postać cyfrową i ich zamiana na obraz termiczny. Dane odczytane z matrycy są przekazywane do następnych modułów kamery termowizyjnej za pomocą magistrali danych obrazowych. Układ sterowania odczytem jest ponadto wyposażony w magistralę sterowania za pomocą, której można ustawić parametry generowanych sygnałów dla matrycy mikrobolometrycznej. Parametry, które mogą podlegać zmianie to liczba obrazów odczytywanych w ciągu sekundy oraz czas całkowania sygnału z detektorów. W kolejnych modułach przetwarzania obrazu dokonywane są operacje takie jak np. korekcja niejednorodności detektorów matrycy, wykrywanie i usuwanie wadliwych pikseli, zaawansowane metody poprawy jakości obrazu. metody wspomagające wykrywanie i identyfikację obiektów. Dzięki zastosowanej architektury systemu możliwa jest adaptacyjna zmiana działania systemu bez konieczności stosowania znaczących zmian sprzętowych. Praca naukowa finansowana ze środków na naukę w latach 2009-2011 jako projekt rozwojowy.

Słowa kluczowe: Termowizja, przetwarzanie obrazu, FPGA

APPLICATION OF THE KOHONEN NEURAL NETWORK IN ANALYSIS OF THE MEASUREMENT RESULTS OF THE POLARIZATION MODE DISPERSION

Sławomir Andrzej Torbus¹, Marta Kolasa¹, Rafał Długosz²

¹ University of Technology and Life Science
Al. S. Kaliskiego 7, 85-796 Bydgoszcz
slator@utp.edu.pl, markol@utp.edu.pl

² Swiss Federal Institute of Technology in Lausanne,
Institute of Microtechnology,
Rue A.-L. Breguet 2, CH-2000, Neuchâtel, Switzerland,
rafal.dlugosz@epfl.ch

Summary: This paper presents a subject of the Polarization Mode Dispersion (PMD). PMD is characteristic for a single mode optical fiber transmission. Several aspects have been presented in the paper, such as the interferometric method for measuring the PMD, as well as the statistical analysis of the measurement results contrasted with the analysis of the same results by use of the Kohonen neural network (KNN).

Keywords: Polarization Mode Dispersion, interferometric method for measuring the PMD, statistical analysis, Kohonen neural network

1. INTRODUCTION

Polarization Mode Dispersion (PMD) is characteristic only for a single mode optical fiber (telecommunication fiber) transmission. It results from the fact that only one mode, LP_{01} , is provided in the telecommunication fiber. It is called the fundamental mode. It is double generated, as it is a combination of two orthogonal modes, which are in two different polarization states.

One can distinguish two axes: **the fast axis** which is the vertical axis responsible for the faster polarization and the **slow axis** which is the horizontal axis responsible for the slower polarization of the optical fiber [1]. When the optical fiber is ideal then both modes are propagated with the same speed. In practice there is no such case and the PMD phenomenon occurs, which results from the following factors:

- **intrinsic factors (fiber imperfections):** noncircular geometry, asymmetrical distribution of refraction factor between the fiber core and the cladding, mechanical stresses and local deformations in the the border between the core and the cladding;
- **external factors (physical stresses):** inflexion, compression, twist, influence of electric and magnetic field and temperature.

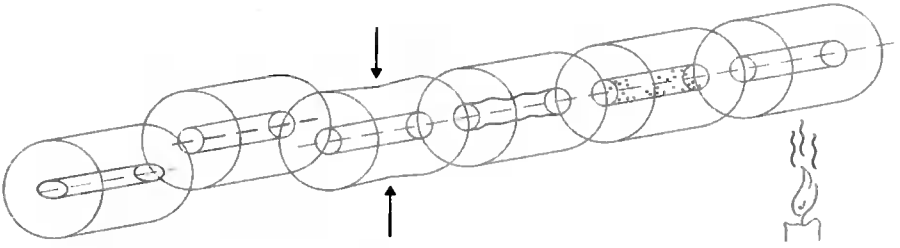


Fig. 1. Graphic presentation of factors influence of PMD

Changes of the speed that are caused by the PMD are quasi-random (probably-random), so the PMD is called statistic or stochastic, because it fluctuates in time (e.g. with the age of optical fiber). Statistic character of this phenomenon causes, that it should be measured when the bitrate in the optical network is bigger than 2.5 Gbps.

Differential Group Delay – DGD between two orthogonal polarization modes has the Maxwell's solution in time, which has been proved both empirically and analytically. The middle result of the DGD is called a delay of the PMD and is shown as the PMD factor [1]. For long optical fibers with the length to exceed one or two kilometers, the PMD factor is defined as $PMD_{LLcoeff}$ and is expressed in $[\text{ps}/\sqrt{\text{km}}]$.

The unit of the PMD shows that this factor does not increase linearly with the length of the fiber but is proportional to the square root of this length, as follows [1]:

$$PMD_{LLcoeff} = \frac{\Delta\tau}{\sqrt{L}} \quad (1)$$

where:

- $\Delta\tau$ – a differential group delay [ps];
- L – the length of the optical fiber [km].

The consequence of the PMD effect is wider optical impulse ($\Delta\tau$). It lowers power of optical signal. To mark (fix) the length of optical fiber in which the PMD effect occurs it is necessary to use some dependents [1]:

- admissible decreasing the sensitivity of the receiver usually settled for 1 dB, so if it is needed to keep loose of power down 1 dB, is fixed that DGD – $\Delta\tau$ should be 0,1 bitperiod – T_{bit} :

$$\Delta\tau_{max} \leq \frac{T_{bit}}{10} \quad (2)$$

where:

- $\Delta\tau_{max}$ – maximum value of differential group delay [ps];
- T_{bit} – bitperiod [ps].

- if $T_{bit} = \frac{1}{B}$, where B is the bitrate, then considering equation (1) the length of optical fiber that depends of the $PMD_{LLcoeff}$ factor can be expressed as follows:

$$L \leq \frac{1}{100 \cdot B^2 \cdot (PMD_{LLcoeff})^2} \quad (3)$$

where: L is the length of optical fiber [km], B is the bitrate [Gbps], $PMD_{LLcoeff}$ is the PMD factor for a long optical fiber [ps/ $\sqrt{\text{km}}$].

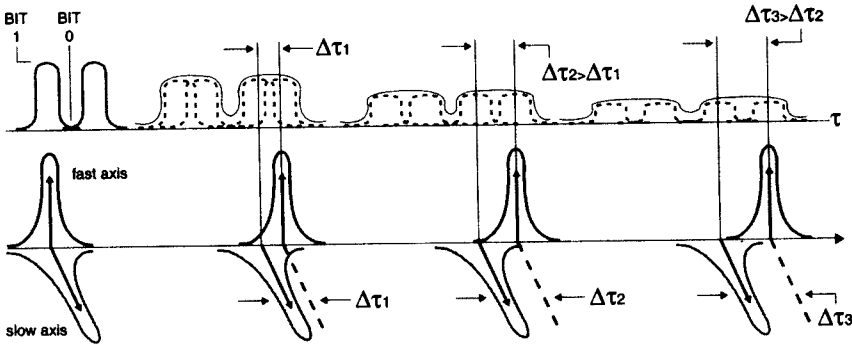


Fig. 2. The PMD effect in the optical signal

Currently used single mode optical fibers that have been described in the Recommendation G.652 [2], should have the $PMD_{LLcoeff}$ factor smaller than $0.5 \text{ [ps}/\sqrt{\text{km}}]$ [1, 2]. Most of the optical fibers, that are currently produced, have the $PMD_{LLcoeff}$ factor less than $0.5 \text{ [ps}/\sqrt{\text{km}}]$, sometimes even ten times less. For the bitrate of 2.5 Gbps, 10 Gbps and 40 Gbps the maximum value of the DGD ($\Delta\tau$) should be equal to 40 ps, 10 ps and 2.5 ps respectively. Theoretically, for $PMD_{LLcoeff} = 0.5 \text{ [ps}/\sqrt{\text{km}}]$, the length of the optical fiber is limited to 6400 km for STM-16, to 400 km for STM-64, and to 25 km for STM-128 [1].

2. EXPERIMENTAL RESULTS AND CHARACTERISTICS OF THE INTERFEROMETRIC METHOD

A 54 kilometers long optical telecommunication line, composed of two optical fibers, has been used to perform described measurements. This line is rented from PKP (Polish Railways) by a Polish company that offers the access the fast Internet and data transmission. This line has been rebuilt to enable transmission of signals with the bitrate in-between 30 and 40 Gbps. On this optical way were some different events like: macrobendings, mechanical weld, electrical welt, connectors. This line has been verified to be in the accordance with the Instruction T - 01 TP S.A. and Recommendation ZN - 96 TP S.A. - 002. This means that: attenuation of the line, attenuation of events and the factor of Chromatic Dispersion (all for two wavelength: 1550 nm and 1625 nm) with OTDR Anritsu model MW9076, was made. The $PMD_{LLcoeff}$ factor has been measured using the PMD analyzer offered by Nexus.

The measurements of the PMD effect have been performed by interferometric method, which takes advantage of the function of autocorrelation of the electromagnetic field (the characteristic of power fluctuation in time). A wideband source of the light (electroluminescence diode – LED or the halogen lamp, which emits the white light) is connected with one end of the optical fiber, while the second end of this fiber is connected to the Interferometer. This method is used to measure the DGD parameter in-between 0.1 ps and 100 ps for the wave length ranging from 60 to 80 nm [3]. To perform the measurements for larger lengths the PMD analyzer was used instead of the Interferometer.

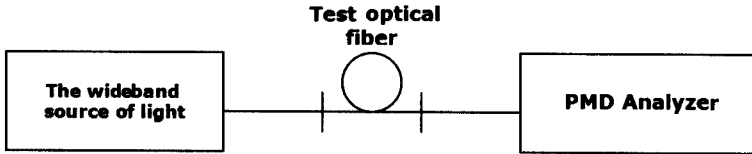


Fig. 3. Schematic diagram of the measurement experiment of the PMD factor

DGD between two orthogonal modes can be marked using the function of autocorrelation. Two examples can be mentioned:

- **the optical fiber with a weakly coupled mode:** a distant between the main and the side peaks is large. In this case the DGD parameter is as follows [3]:

$$\Delta\tau_{g,r} = \frac{2 \cdot dk}{c} \quad (4)$$

where:

dk – is the length of a movable mirror from the position in which distances between the interferometer's mirrors were equal [m], while c is the speed of light in the vacuum equal to $2.998 \cdot 10^8$ [m/s];

- **the optical fiber with the strongly coupled mode:** a distance between the main and the side peaks is small. The DGD parameter can be described using the Gauss resolution for approximation of the characteristic of the autocorrelation function [3]:

$$\Delta\tau_{g,r} = \sigma_{X(t)} \cdot \sqrt{\frac{3}{4}} \quad (5)$$

where:

$\sigma_{X(t)}$ – is the standard deviation of the normal resolution approximating characteristic of the autocorrelation function in the optical fiber with the strong couple mode [ps].

The length of the optical fiber was measured using OTDR, before measurement of the PMD by use of the PMD analyzer. The obtained values were put into the PMD analyzer. The $PMD_{LL\text{coeff}}$ factor will be estimated by use of the length of the line according to equations (1) and (5).

The PMD parameter has static properties that enabled repeating the measurements for ninety times, with the intervals of 16 minutes (ninety results for a single optical fiber). The measurements have been performed for each of the two optical fibers, to

obtain correct results. The transmission will be provided in the III optical window ($\lambda = 1550$ nm), so the measurements were done only for the wavelength of 1550 nm. The PMD factor is measured only in one way such as the Chromatic Dispersion factor.

3. STATISTICAL ANALYSIS OF THE MEASUREMENT RESULTS

Analysis of measurement results the factor PMD for optical fiber is presented in the following order:

- checking the type of the distribution which is formed by gathered measurement results using the harmony test of Pearson or Kolmogorow – Smirnow;
- marking the trust section for the measured PMD factor and ascertainment which of the results don't belong to this section;
- marking the range of bitrate using the trust section for the measured PMD factor for two significance levels α ;
- final conclusions concerning presented statistical analysis of the presented measurement results and expertise of the whole optical telecommunication line.

The measurements of the PMD effect have been performed in two samples of optical fibers A and B with 16 minutes intervals. The statistical analysis is performed only for the A optical fiber. The following results for the A fiber have been achieved. It should be determined which solution is the source of the PMD effect and then this section should be marked to eliminate the incorrect results. Using equations (2) and (3) it should be verified which of the acceptable factors of the PMD can appear in a given optical network that use the bitrates of 30 and 40 Gbps.

Two different significance levels have been selected in the presented analysis, namely $\alpha = 0.05$ and $\alpha = 0.1$.

Firstly, the null hypothesis should be formulated. It is supposed that the resolution of the PMD factor is normal: $H_0 : F(x) \sim N(m, \sigma)$.

Secondly, the normal resolution parameters have to be marked by basing on the data obtained from measurements. To perform this task, the following patterns and calculations have been used:

$$m = \frac{\sum_{i=1}^{90} x_i}{90} = 0.378 \left[\frac{\text{ps}}{\sqrt{\text{km}}} \right] \quad (6)$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{90} (x_i - m)^2}{90}} = 0.136 \left[\frac{\text{ps}}{\sqrt{\text{km}}} \right] \quad (7)$$

where:

- m – expected value (mean, the first moment of the normal);
- x_i – measurement result (factor PMD);
- σ – standard deviation.

Thirdly, the measured data need to be grouped into some distribution series and presented in the table that simplifies the analysis which is required to verify the authenticity of the null hypothesis.

Table 1. The help table for the harmony test of Pearson

I	Section of distributing series	p_i	$n_{\Sigma} \cdot p_i$	l_n	$(l_n - n \cdot p_i)^2$	$\frac{(l_n - n \cdot p_i)^2}{n \cdot p_i}$
1	$(-\infty; 0.3)$	0.281	9.549	32	504.063	52.789
\vdots						
32						
33	$\langle 0.3; 0.4 \rangle$	0.279	9.481	22	156.733	16.532
\vdots						
54						
55	$\langle 0.4; 0.5 \rangle$	0.277	9.413	23	184.614	19.613
\vdots						
77						
78	$\langle 0.5; +\infty \rangle$	0.187	5.539	13	44.163	6.950
\vdots						
90						
\sum_i	$n = 33.981$	≈ 1		90		$\chi^2 = 95.884$

where:

n_{Σ} – sum of the measurement results;

l_n – size range in a number of distributive.

The probability p_i of findings that the PMD value was estimated using the table of the distribution function for the normal resolution in the analyzed section of the distributing series. It has been performed in the following way:

$$\begin{aligned}
 P(x \in (-\infty; 0.3)) &= P(-\infty < x < 0.3) = \\
 &= P\left(\frac{-\infty - m}{\sigma} < u < \frac{0.3 - m}{\sigma}\right) = P\left(\frac{-\infty - 0.378}{0.136} < u < \frac{0.3 - 0.378}{0.136}\right) = 0.281 \quad (8)
 \end{aligned}$$

$$\begin{aligned}
 P(x \in \langle 0.3; 0.4 \rangle) &= P(0.3 \leq x < 0.4) = \\
 &= P\left(\frac{0.3 - m}{\sigma} \leq u < \frac{0.4 - m}{\sigma}\right) = P\left(\frac{0.3 - 0.378}{0.136} \leq u < \frac{0.4 - 0.378}{0.135}\right) = 0.279 \quad (9)
 \end{aligned}$$

$$\begin{aligned}
 P(x \in \langle 0.4; 0.5 \rangle) &= P(0.4 \leq x < 0.5) = \\
 &= P\left(\frac{0.4 - m}{\sigma} \leq u < \frac{0.5 - m}{\sigma}\right) = P\left(\frac{0.4 - 0.378}{0.136} \leq u < \frac{0.5 - 0.378}{0.135}\right) = 0.277 \quad (10)
 \end{aligned}$$

$$\begin{aligned}
 P(x \in \langle 0.5; +\infty \rangle) &= P(0.5 \leq x < +\infty) = \\
 &= P\left(\frac{0.5 - m}{\sigma} \leq x < \frac{+\infty - m}{\sigma}\right) = P\left(\frac{0.5 - 0.378}{0.136} \leq x < \frac{+\infty - 0.378}{0.136}\right) = 0.187 \quad (11)
 \end{aligned}$$

Using the χ^2 – Pearson resolution's table for two different significance levels: $\alpha = 0.05$ and $\alpha = 0.1$, and 89 degrees of freedom we receive:

$$\left. \begin{array}{l} \alpha = 0.05 \\ i-1 = 89 \end{array} \right\} \chi_{\alpha}^2 = \chi_{0.05}^2 = 112.022$$

$$\left. \begin{array}{l} \alpha = 0.1 \\ i-1 = 89 \end{array} \right\} \chi_{\alpha}^2 = \chi_{0.1}^2 = 106.469$$

The following equation has to be used to verify the correctness of the null hypothesis [5]:

$$P(\chi^2 > \chi_{\alpha}^2) = \alpha \quad (12)$$

The null hypothesis can not be rejected, considering the obtained results. It results from the fact that the inequality in equation (12) is not true, as $\chi^2 = 95.884$.

If the resolution of the measured PMD factor is known, the trust section can be marked in order to verify given results during the experiment. For this purpose, Table 1 and a positive number of t_{α} have been used. The positive number characterizes the accuracy of the estimation and it can be obtained from the normal resolution Tables:

$$\left. \begin{array}{l} \alpha = 0.05 \\ i-1 = 5 \end{array} \right\} t_{\alpha} = t_{0.05} = 1.987$$

$$\left. \begin{array}{l} \alpha = 0.1 \\ i-1 = 5 \end{array} \right\} t_{\alpha} = t_{0.1} = 1.662$$

The trust section for the normal resolution is described as follows [5]:

$$\left(m - t_{\alpha} \cdot \frac{\sqrt{\frac{\sum_{i=1}^n (x_i - m)^2}{n-1}}}{\sqrt{n-1}}; m + t_{\alpha} \cdot \frac{\sqrt{\frac{\sum_{i=1}^n (x_i - m)^2}{n-1}}}{\sqrt{n-1}} \right) \quad (13)$$

where:

n – number of the measurement results.

Using equation (13), the trust section can be found for a specified significance level α , while using equation (3) and the length of the optical fiber (L) the range of the bitrate can be determined as follows:

- for $\alpha = 0.05$: $\left(0.378 - 1.987 \cdot \frac{0.136}{\sqrt{89}}; 0.378 + 1.987 \cdot \frac{0.136}{\sqrt{89}} \right) \Rightarrow (0.349; 0.406) \left[\frac{\text{ps}}{\sqrt{\text{km}}} \right]$
 $B \in (33.515; 38.981) [\text{Gbps}]$

- for $\alpha = 0.1 \left(0.378 - 1.662 \cdot \frac{0.136}{\sqrt{89}}; 0.378 + 1.662 \cdot \frac{0.136}{\sqrt{89}} \right) \Rightarrow (0.354; 0.401) \left[\frac{\text{ps}}{\sqrt{\text{km}}} \right]$
 $B \in (33.904; 38.468) \text{ [Gbps]}$

In the conclusion, it can be noticed that the assumed by the system administrator transmission bitrate in-between 30 and 40 Gbps, does not fall within the range which was based on the statistical analysis of the measurement results. This means, that the tested fiber optic should not be used in the transmission system, with a maximum bitrate of 40 Gbps. Accordingly, the transmission can't be realized using SDH (STM – 256), because the maximum bitrate in this system is 40 Gbps. The transmission system will operate correctly for the maximum bitrate of 38 Gbps, or by using a different optical fiber, with a lower value of the PMD factor.

4. KOHONEN NEURAL NETWORKS ALGORITHM

Teuvo Kohonen proposed a new class of neural networks in 1975, that use competitive, unsupervised learning algorithms [6]. The Kohonen neural networks (KNNs) in their classical approach, also called self organized maps (SOM), contain one layer of neurons. This layer is organized as a map. The number of the outputs of the network equals to a total number of neurons. All neurons have common inputs, whose number depending on the application varies in-between two and even several dozen. The SOMs are used in data visualization and analysis [7, 8, 9].

The competitive unsupervised learning in KNNs relies on presenting the network with the learning vectors X in order to make the neurons' weight vectors W resemble presented data. For each training vector X both KNNs determine Euclidean distances (d_{EUC}) between this vector and the weights vectors W in each neuron, which for n network's inputs are calculated using the following formula:

$$d_{\text{EUC}}(X, W_i) = \|X - W_i\| = \sqrt{\sum_{l=1}^n (x_l - w_{il})^2} \quad (14)$$

The neuron, whose weights are the most similar to the training vector X becomes a winner and is allowed to adapt own weights. Two general types of such networks can be distinguished. In the Winner Takes All (WTA) approach only the winning neuron can adapt the weight, while in the Winner Takes Most (WTM) algorithm also neurons that belong to the winner's neighborhood are allowed to adapt the weights, according to:

$$W_j(l+1) = W_j(l) + \eta(k)G(i, j, R, d)[X(l) - W_j(l)] \quad (15)$$

where:

- $\eta(k)$ – learning rate in the k^{th} training epoch;
- $G(i, j, R, d)$ – neighborhood function.

These neurons that belong to the winner's neighborhood are trained with different intensities that depend on the neighborhood function $G()$. In the classical approach a simple rectangular function is used which is defined as [6, 9]:

$$G(i, j, R, d) = \begin{cases} 1 & \text{for } d(i, j) \leq R \\ 0 & \text{for } d(i, j) > R \end{cases} \quad (16)$$

where:

- $d(i, j)$ – topological distance between the winning, i^{th} , neuron and any other, j^{th} , neuron in the map;
- R – range of the neighborhood that is decreased after each epoch.

The common opinion is that much better results can be achieved if the Gaussian function is used instead of the rectangular one [8]. The Gaussian function is defined as follows:

$$G(i, j, R, d) = \exp\left(-\frac{d^2(i, j)}{2R^2}\right) \quad (17)$$

This is important to distinguish the map topology from the neighborhood function. Topology means the grid of neurons i.e. determines which neurons belong to the neighborhood of any neuron in the map [9, 11]. Typical topologies described in literature are rectangular with four of eight neighbors or the hexagonal [6-9].

5. MEASURED DATA ANALYSIS BY KNN SOM

The KNN has been used to analyze the measurement results of the PMD parameter. The number of the inputs has been selected to be equal to 2. The input vectors consist of the following parameters: PMD and B . The number of the training patterns was equal to 90. The simulations have been performed for the Gaussian neighborhood function and the rectangular grid with four neighbors. The map size is 2x2 neurons.

Training the WTM network has been divided into two phases i.e. the *ordering phase* at the beginning of the learning process and the *tuning phase* at the end. In particular phases, the learning rate η , as well as the value of the radius R , have different values. The learning rate η decreases along the learning process starting from a relatively large value $\eta_{\max} = 0.9$ to very small values, on the level of $\eta_{\min} = 0.02$ in this particular case. In the tuning phase the η parameter remains constant and equal to the lower value. The radius R , in the ordering phase, decreases from its maximum value, for which it covers the entire map to zero. In the tuning phase, the value of this parameter remains constant and equals 0. This means that only the winning neuron is allowed to adapt the weights i.e. the SOM works as the WTA network. All these parameters mentioned above have the influence on both the speed and the quality of the learning process, which can be evaluated using the so called quantization error. This error is defined as follows:

$$Q_{err} = \frac{\sum_{j=1}^m \sqrt{\sum_{l=1}^n (x_{jl} - w_{wl})^2}}{m} \quad (18)$$

where:

- m – number of the learning patterns in the input data set;
- n – number of the network inputs.

The measured data have been divided into four distinct classes in the statistical method described above. The number of four results from the real areas. The last class i.e. for $(PMD \in (0.5; +\infty))$ represents these values of the PMD coefficient, which exceed the values in the ITU-T order. The range of values for $PMD \in (-\infty; 0.5)$, or in the real case for $PMD \in (0; 0.5)$, have been divided into three classes with a similar numerical value. Data from particular classes generate similar results. To demonstrate the statistical character of the PMD coefficient, it is necessary to divide data at least into three classes. Division of data into larger number of classes will have no influence on the confidence interval described by (13). On the other hand, increasing the number of classes will increase the time required for analysis, and therefore the number of four is treated as an optimal value.

The statistical method has one important drawback, that is not present in the method based on the Kohonen SOM. The statistical method does not allow for an automatic division of the measured data into classes. Moreover, the user must also perform oneself all required calculations as described in Section 3. The application of the neural network with a given number of neurons allows achieving the same results as in the statistical method without any action from the user. The user does not need to be a specialist in the area of the mathematical statistics.

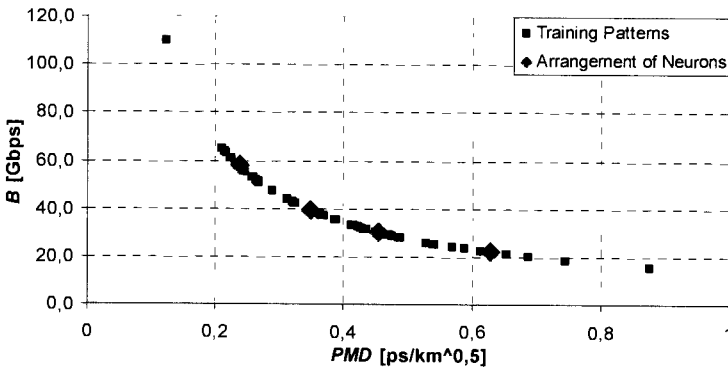


Fig. 4. Training patterns and the final arrangement of neurons after completing the learning process

Software model simulation results, illustrating both training patterns and the final arrangement of neurons after completing the learning process, are shown in Fig. 4. Input data are divided into 4 classes i.e. the number of neurons in the SOM. Each class is represented by a different number of the learning patterns. Particular classes consist of 30, 24, 23, 13 input patterns, respectively. The first class is for $PMD \in (-\infty; 0.286)$, the second for $PMD \in (0.278; 0.392)$, the third for $PMD \in (0.392; 0.523)$, while the fourth for $PMD \in (0.523; \infty)$. The classes obtained using the analysis based on the SOM approximately overlap with the results obtained by use of the statistical method.

Kohonen SOM can be realized in various ways. The authors present the software model simulation results in this paper, but different hardware solutions, based on the application specific integrated circuits (ASIC) have been proposed by the authors earlier [10-14]. Hardware implemented networks offer a fully parallel operation of all neurons

in the map and therefore can become much faster than their software counterparts, while consuming much less energy [12-14]. This opens quite new possibilities of applications of such networks e.g. in low power portable devices. Hardware implemented network of this type will allow for a constant control monitoring of the optic fiber tracts.

Hardware implemented KNN will enable an automatic choice of the optimal fiber optic line. This will enable data transmission to be performed on the constant basis with the bit rate assumed by the operator. An additional advantage of this solution is the possibility of its application in particular nodes of the digital broadcasting networks, in which different transmission paths converge. After determining the bit rate for each of the input paths, assuming the knowledge of the PMD coefficient values for these paths, one can use the proposed solution to make the allocation of the input data streams to the respective output paths. This solution will ensure preservation of the established top bit rate between the boundary (terminal) nodes in the network. In case when the broadcast is performed in one of the four fibers represented by particular classes of the Kohonen map, the remaining fibers can be analyzed using the KNN in real time. After detecting a more favorable transmission path, the network can automatically select this path. This will ensure more reliable operation of the entire transmission system. Sample Flowchart of the proposed solution is shown in Figure 5.

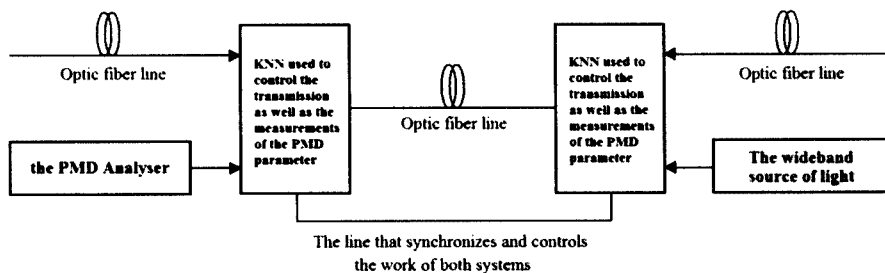


Fig. 5. Block diagram of control – measuring realized based on the KNN

6. CONCLUSIONS

In his paper the Kohonen SOM has been used to analyze the measurement results of the PMD parameter. Obtained results are very similar to the results obtained in the statistical analysis. Using artificial neural network allows for saving the time required for execution of the statistical analysis, as the network perform this analysis and data classification in the real time. In case of the statistical method, this type of analysis can be performed only before opening the fiber-optic highway or in the situation when a reconfiguration has been performed. This opens quite new application possibilities of such networks.

BIBLIOGRAPHY

- [1] Ratuszek M., Zakrzewski Z., Majewski J., Wronikowski M., 2001. Wpływ dyspersji polaryzacyjnej na parametry transmisyjne światłowodów KST'2001 – Bydgoszcz.

- [2] Recommendation ITU – T G.652: Characteristics of a single – mode optical fiber and cable 03/2003.
- [3] Perlicki K., 2002. *Pomiary w optycznych systemach telekomunikacyjnych WKŁ*, Warszawa.
- [4] Torbus S.A., 2007. Measurement and analysis of Polarization Mode Dispersion Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments, Wilga, Proc. of SPIE, 0277-786X, Vol. 6937, 693710.
- [5] Gmurman W.J., 1975. *Rachunek prawdopodobieństwa i statystyka matematyczna WNT*, Warszawa.
- [6] Kohonen T., 2001. *Self-Organizing Maps*, third ed. Springer, Berlin.
- [7] Brocki L., 2007. *Recent Advances in Mechatronics*, Springer Berlin-Heidelberg.
- [8] Mokriš, Forgáč R., 2004. Decreasing the Feature Space Dimension by Kohonen Self-Organizing Maps, 2nd Slovakian – Hungarian Joint Symposium on Applied Machine Intelligence, Herľany, Slovakia.
- [9] Boniecki P., 2005. The Kohonen neural network in classification problems solving in agricultural engineering, *Journal of Research and Applications in Agricultural Engineering*, Vol. 50(1), pp. 37-40, Poznań.
- [10] Verleysen D.M., Jaspers P., Legat J-D., 1993. Analog Implementation of a Kohonen Map with On-Chip Learning, *IEEE Transactions on Neural Networks*, Vol. 4, No. 3, pp. 456-461.
- [11] Fei Li, Chip-Hong Chang L., Siek A., 2009. Compact current mode neuron circuit with Gaussian taper learning capability, *IEEE International Symposium on Circuits and Systems (ISCAS)*, 24-27, pp. 2129-2132.
- [12] Długosz R., Talaska T., Pedrycz W., Wojtyna R., 2010. Realization of a Conscience Mechanism in CMOS Implementation of Winner Takes All Neural Networks, *IEEE Transactions on Neural Networks*.
- [13] Długosz R., Kolasa M., 2009. Optimization of the Neighborhood Mechanism for Hardware Implemented Kohonen Neural Networks, *European Symposium on Artificial Neural Networks (ESANN)*, Bruges, Belgium, pp. 565-570.
- [14] Kolasa M., Długosz R., 2009. Hardware Implementation Issues of the Neighborhood Mechanism in Kohonen Self Organized Feature Maps, *ESANN (European Symposium on Artificial Neural Networks Advances in Computational Intelligence and Learning)*, Bruges (Belgium), pp. 565-570.

ZASTOSOWANIE SIECI NEURONOWEJ KOHONENA DO ANALIZY WNIKÓW POMIARU DYSPERSJI POLARYZACYJNEJ

Streszczenie

W pracy omówiono zagadnienie dyspersji polaryzacyjnej – PMD (ang. Polarization Mode Dispersion), która jest charakterystyczna dla transmisji z wykorzystaniem jednomodowego włókna światłowodowego. Przedstawiono również interferometryczną metodę pomiaru współczynnika dyspersji polaryzacyjnej, statystyczną analizę rzeczywistych wyników pomiaru oraz analizę tych samych wyników pomiaru za pomocą sieci neuronowej Kohonena.

Słowa kluczowe: dyspersja polaryzacyjna, metoda interferometryczna pomiaru PMD, analiza statystyczna wyników pomiarów, sieć neuronowa Kohonena

COMPARISON OF DIFFERENT HARDWARE REALIZATIONS OF THE WINNER TAKES ALL NEURAL NETWORK

Tomasz Talaśka¹, Paweł Przedwojski², Jakub Dalecki³, Rafał Długosz¹

¹ Faculty of Telecommunication and Electrical Engineering
University of Technology and Life Sciences (UTP),
Kaliskiego 7, 85-789 Bydgoszcz, Poland

² Tieto Poland Sp. z o.o.

Grodzka 20, 70-560 Szczecin, Poland

³ STAR- Projekt Lech Starczak

Przemysłowa 8, 85-758 Bydgoszcz, Poland

talaska@utp.edu.pl, pawel.przedwojski@wp.pl, dalecki@o2.pl, rafal.dlugosz@gmail.com

Summary: This paper presents realization and the laboratory tests of the Kohonen winner takes all (WTA) neural network (NN) realized on microcontrollers (μ C) with the AVR and ARM CortexM3 cores, as well as the comparison with the full custom implementation of analog network of this type in the CMOS technology. The two μ Cs have been placed on a single testing board to facilitate the comparison. The board allows for switching between the two μ Cs, it enables selection of either the Euclidean (L2) or the Manhattan (L1) distance measures. It also allows for turning on/off the so-called conscience mechanism. Some signals illustrating the training of the network can be observed directly on the board. The full learning process with all essential parameters can be viewed on PC using the USB port. The prospective application of the system is in on-line analysis of the ECG and EMG biomedical signals in the health care diagnostic systems, as well as in the student laboratories on neural networks and programmable devices.

Keywords: WTA network, digital neural networks, analog neural networks, micro-controller, low energy consumption

1. INTRODUCTION

Artificial NNs are commonly used in such tasks that require processing and classification of “difficult” signals e.g. non-stationary signals, heuristic data etc. in medical health care, telecommunication, electrical engineering and other application areas. In literature one can find various implementation techniques of NNs both the software- and the hardware-based. Considering such criteria as energy consumption, calculation capacity and device size, *full-custom* designed networks are the most efficient solutions [15]. NNs realized as application specific integrated circuits (ASIC) allow, for example, for parallel data processing, and thus consuming significantly less energy are often faster than the software-based networks. The full-custom style allows for a very good matching of the circuit structure to a given task. A disadvantage of this approach is relatively complex design process and large fabrication cost in case of short series. It is also relatively difficult to built-in such a programmable network of this type.

In this paper the authors present realization of the Kohonen WTA NN [2] using an alternative approach based on two μ Cs, as well as the comparison with the analog WTA NN previously implemented in the CMOS 0.18 μ m technology. For this purpose a special testing board has been developed using the Eagle 5.6 environment. To get a better insight into the parameters of the NN realized in this way, two different μ Cs have been used i.e. the 8-bits AVR and the 32-bits ARM CortexM3. Such realized testing board allows for training the NN with different parameters i.e. with two different measures ($L1 / L2$) of similarity between the input training patterns X and the weight vectors W of particular neurons. Both measures are shortly presented in Section III. The network can be trained with or without the so-called conscience mechanism [13, 14, 15]. All these modes can be selected by the manually-operated switches placed directly on the board.

Realizations that involve μ Cs are relatively cheap, which is one of their important advantages. In comparison with custom-designed networks all parameters can be easily reprogrammed. On the other hand, because of serial data processing, networks of this type are relatively slow and thus are suitable for small networks with up to fifty neurons and sampling frequencies not exceeding 100 kHz. However, many applications still can be indicated, in which such parameters are accepted, e.g. in the analysis and classification of the ECG and EMG biomedical signals sampled at max 2 kHz.

Implementation of neural networks on μ Cs is significantly cheaper than realization of such networks by use of digital signal processors (DSP) or in field programmable gate array (FPGA). Furthermore, microcontrollers offered on the market today aid floating point operations, while multiplication operations are performed in a single clock cycle. As a result, the general parameters of both platforms are often comparable. It is worth mentioning that very often core blocks of DSPs are the same like those used in microcontrollers (the ARM core).

The paper is organized as follows. State-of-the-art in the field of Kohonen WTA NNs realized using programmable platforms and the analog technique is presented in next section. An overview of the WTA NN principle is provided in Section III. Section IV presents realization of the testing board described above. In Section V selected measurement results are shown together with a discussion of the achieved parameters. The conclusions are covered in Section VI.

2. STATE-OF-THE-ART STUDY

2.1. REALIZATIONS OF THE KOHONEN NEURAL NETWORKS BASED ON MICROCONTROLLERS

An idea of the implementation of the self organizing map (SOM) [2] using SIMD (Single Instruction Multiple Data) processors has been described in [3]. These processors allow for parallel information processing using a single instruction. They find the application in the newest computer systems. The authors of [3] focused on different methods of detection of the winning neuron i.e. on the so called WTA circuits.

Using SIMD processors is beneficial in case of large NNs with more than 100 neurons, in which data rate is one of the key parameters. In case of smaller networks with sampling frequencies not exceeding 100 kHz such realizations become uneconomical. The SIMD processors are more expensive and dissipate more power than the AVR/ARM μ Cs.

The other reason, important in commercial applications, is that AVR/ARM μ Cs can be easier programmed and require simpler environment (the printed circuit board).

Another implementation has been reported in [4]. This paper presents realization as well as the optimization of selected arithmetic operations, such as multiplication and \tanh using feed-forward multi-layer network implemented in PIC18F45J10 μ C. The authors of [4] conclude that are able to implement up to 256 neuron weights using this μ C. Assuming this network to be a multi-layer architecture, this means that relatively small networks can be realized with the number of neurons not exceeding 50. It is difficult to assess the performance of this network as the achievable data rate, power dissipation and the computational capacity are not provided.

In case of hardware implemented NNs the computational capacity depends on either data processing is performed serially or in parallel, the sampling frequency of the μ C core, the number of neurons and several other parameters. An important issue is the complexity of the training algorithm offered by a NN. μ Cs are rather suitable for such networks that require simple arithmetic operations. For example, the WTA algorithm implemented by the authors requires only multiplications, summations and subtractions. For the comparison the network described in [4] requires \tanh activation functions. It is worth mentioning that the WTA network is trained without supervision that makes this NN relatively faster than their counterparts trained with the supervision, in which an error function is calculated separately for every neuron.

Different applications of NNs realized on μ Cs have been reported. Such networks are frequently used in control and diagnostic devices. A device described in [5] has been used as an intelligent wireless electronic nose node (WENN) used in classification and quantification of binary gas mixtures NH₃ and H₂S. In [6] such implemented NN is used in control of the furnace temperature. One can find the applications in which NNs only cooperate with μ Cs. In [7] software implemented feed-forward Back Propagation (BP) NN exchange data with the PIC16F84A μ C that controls a device detecting damages in textiles produced in the factory.

2.2. REALIZATIONS OF KOHONEN NEURAL NETWORKS AS ASICS

Full-custom developments of Kohonen NNs are not common. Several examples can be found in [15, 16, 17]. They are based on different techniques, such as digital, analog or mixed analog-digital.

A fully digital NN realized in the CMOS 0.5 μ m process is reported in [16]. The main disadvantage in this case is a relatively large chip area, which makes the implementation of large networks fairly impractical. Each processing element (PE), representing a single neuron contains about 10.000 transistors and occupies an area of 4 mm².

Another, mixed analog-digital, implementation has been reported in [17]. In this case some modules of the overall architecture like the distance calculation (L2 in this case) and the WTA blocks are implemented as analog components, while the adaptation process is realized by the use of digital blocks. The memory for the weight storage is implemented as digital counters that can count in both directions, depending on whether an input data x is greater than a neuron's weight w or not. The adaptation mechanism used in this solution differs from the classic algorithm proposed by Kohonen. The learning rate, η , is kept fixed, while its value results from the assumed resolution of the counter (5 bits in this particular case). Moreover, unlike the classic Kohonen's algorithm in

which adaptation process depends on exact values of x and w , in [17] it depends on the value of the sign function, $\text{sign}(x, w)$, always updating the counter value by ± 1 .

A fully analog WTA NN has been implemented by the authors of this paper in the CMOS 0.18 μm technology [15]. In this implementation all main components, including the adaptation mechanism, the Euclidean distance calculation block, as well as the winning neuron detection block (WTA) were realized using analog circuitry. A full design cycle of this network, which started with extensive system level simulations performed in the software model realized in the C++ environment, covered also transistor level simulations, full-custom layout design and finally the chip fabrication and experimental verification. The prototype chip has been realized in Canada in TSMC CMOS 0.18 μm process. Layout of this prototype is shown in Figure 1.

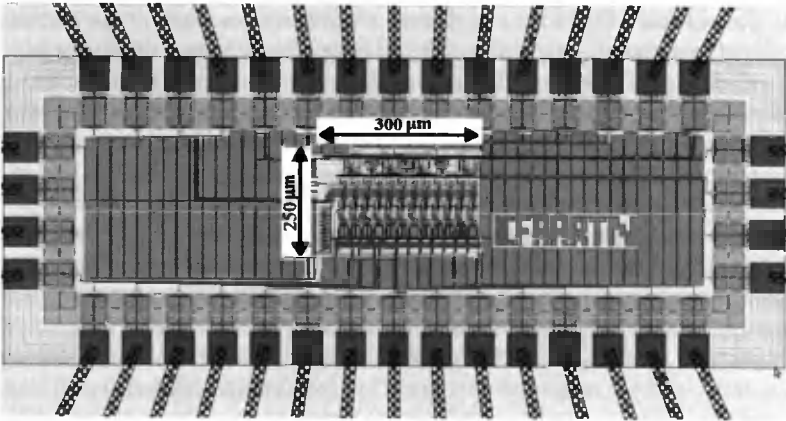


Fig. 1. Layout of the prototype analog WTA NN implemented by the authors in the TSMC CMOS 0.18 μm technology

In measurements the network with 12 channels working in parallel, sampled with 2 MHz clock was able to operate as fast as a standard 2 GHz PC, consuming only 700 μW of power i.e. almost 100 thousands times less than the PC. As this network was a first prototype, only four neurons have been implemented. In case if fifty neurons were realized in a single chip with ten analog inputs, the power dissipation would not exceed 10 mW, while the chip area would equal about 1 mm^2 . Operating at 2 MHz the network would realize 8e09 operations/s. This estimation shows that the analog network realized as ASIC surpasses the networks implemented using the programmable platforms.

3. LEARNING PROCESS IN THE WTA NEURAL NETWORK

The feed-forward network that is in the scope of interests of this paper has been originally proposed by Teuvo Kohonen in [2]. This network features a competitive unsupervised learning. The weights of neurons are modified without any feedback that makes the learning algorithm very fast. Such networks are suitable for the applications, in which data rate is a key parameter e.g. in telecommunications [8-11]. Two types of this network are often distinguished. In this paper the WTA learning method is implemented that is a special case of the winner takes most (WTM) learning algorithm for the

neighborhood range equal to zero. The WTA algorithm is less complex than the WTM one and therefore is much simpler in the hardware implementation.

The learning in Kohonen NNs (KNN) relies on presenting the network with learning patterns, X , in order to make the neurons' weight vectors, W , resemble presented data. For each pattern X the network first determines the distance between this vector and the W vector of each neuron. Different measures of the similarity between these vectors are available. One of them is the Euclidean distance defined as follows:

$$d(X, W_i) = \sqrt{\sum_{l=1}^n (x_l - w_{il})^2} \quad (1)$$

In this work a modified Euclidean measure has been used, in which the rooting operation has been neglected. The results in both cases are the same, as if $a < b$ then always $\sqrt{a} < \sqrt{b}$, while the rooting operation is more difficult in the hardware realization. The Euclidean measure can be denoted as L2. Another frequently used measure, called the "Manhattan" one and denoted as L1, is defined as:

$$d(X, W_i) = \sum_{l=1}^n \text{abs}|x_l - w_{il}| \quad (2)$$

In this case the squaring operations have also been neglected, which enables further simplification of the learning algorithm. Both these measures have been implemented by the authors in the μC and compared with respect to the calculation complexity.

The adaptation of the winning neuron is in the WTA NN performed in accordance with the following formula:

$$W_i(t+1) = W_i(t) + \eta \cdot (X(t) - W_i(t)) \quad (3)$$

where η is the learning rate. Other neurons in the network that lose the competition remain unchanged in this algorithm.

One of the significant problems encountered in the WTA networks are the, so-called, dead neurons i.e. the neurons that take part in the competition but never win and therefore their weights remain unchanged. One of the reasons of this are badly selected initial values of the weights [12]. Such neurons reduce the number of classes that can be discriminated, thus increasing the mapping (quantization) error of the network. For this reason reducing the number of these neurons is an important design objective. One of the efficient methods in this task is by use of the, so-called, conscience mechanism [13, 14]. Its role is to increase the likelihood of winning for all neurons in the network.

The conscience mechanism proposed earlier by the authors and implemented in their analog NN has been also used in the network realized in both μCs . In this case, the real distance between the weight and the training vectors is made higher by adding a signal that is proportional to the number of the wins:

$$d_{\text{cons}}(X, W) = d_{\text{L1/L2norm}}(X, W) + L_{\text{count}} \cdot K \quad (4)$$

where $d_{\text{L1/L2}}(X, W)$ is the real distance determined by use of either the L1 or the L2 metric, $d_{\text{cons}}(X, W)$ is a signal modified by the conscience mechanism and then applied to the WTA block. The L_{count} is the number of the wins of a given neuron. The K coeffi-

cient is the gain factor that allows for controlling and optimizing the learning process by adjusting the strength of the conscience mechanism. The quantization error mentioned above is defined as:

$$Q_E = \frac{1}{z} \cdot \sum_{iter=1}^z \|X(t) - W_j(t)\|^2 \quad (5)$$

where Z is the number of the iterations in each epoch, i.e. the number of all training patterns X in a given input data set. The j index indicates the winning neuron.

4. TESTING BOARD WITH THE WTA NEURAL NETWORK IMPLEMENTED ON MICROCONTROLLERS

The realized testing board with both μ Cs, shown schematically in Fig. 2, has been designed in the Eagle 5.6 environment. The board is composed of several blocks. One of them is the interface block containing the ADC/DAC standard chips that convert analog learning signals X to digital form used then by the μ Cs, as well as the neuron weights to analog form for the observation. The ‘Switching field’ allows for selecting one of the two μ Cs, as well as the distance measure (L1/L2), as shown in Fig. 3.

The conscience mechanism can be turned on/off as well. The proposed system due to manifold of different options can be used both in commercial application and in education. Further development possibilities of this system still exist.

The neuron weights can be observed on-line either as the analog signals at the output of the DAC blocks or directly as the digital signals. The analog signals allow for a rough verification of the learning process on the oscilloscopes. Digital signals can be acquired on PC by use of the USB and RS232 serial ports, for further off-line analysis.

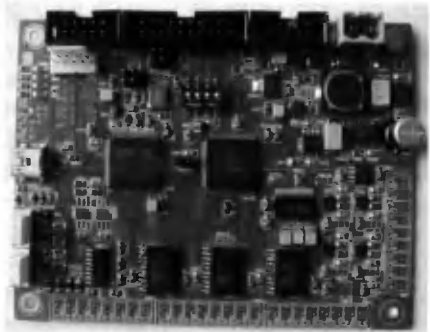
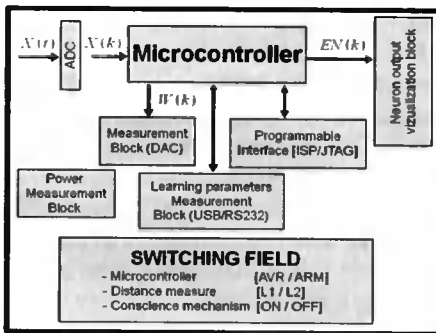


Fig. 2. The proposed testing board of the programmable WTA neural network based on micro-controllers

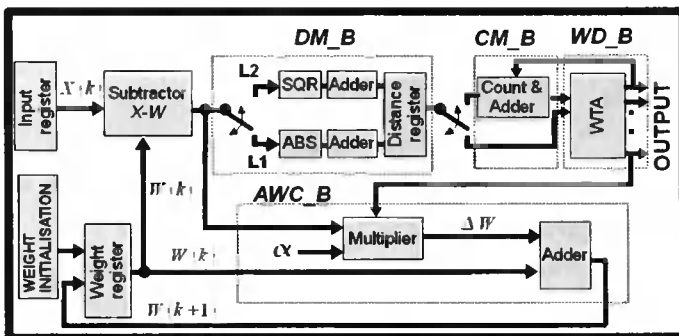


Fig. 3. WTA learning process implemented on μC : DM_B is a distance measurement block, CM_B – the conscience mechanism block, WD_B – the winning neuron detecting block (WTA), AWC_B – weight adaptation block

The board has been designed in such a way to enable measuring on-line the power dissipation, separately for the ADC / DAC blocks and for both μCs . The μCs are programmed by the use of the ISP / JTAG interfaces. The serial ports also allow for acquiring on PC the learning patterns X (as digital signal samples), the calculated distances between the X and the W vectors, the numbers of the wins and the quantization error for detailed analysis of the network performance. The board allows for a full observation of the network with 3 inputs and 4 outputs i.e. 12 neuron weights. To make it possible, a single 4-channel THS1206 ADC and three AD7305 4-channel DACs have been used. The number of neurons can be increased but in this case only selected weights can be observed on-line on the oscilloscopes.

One of the reasons of selecting in this prototype only 12 weights for direct observation was to facilitate a direct comparison with the analog network previously designed by the authors with just 3 inputs and 4 outputs. For a better comparison between both μCs the X and the W signals have in this approach the resolution of 8 bits. This resolution is sufficient in many practical applications. The float type on the ARM μC has not been used in this case, although such a possibility still exists.

5. LABORATORY TESTS OF THE REALIZED WTA NETWORK

The maximum achievable data rate of the network in case of the implementation on μCs depends on the number of the inputs and the outputs, as shown in Fig. 4 for both μCs . The results are present for different parameters of the learning process. The learning speed can be determined using the following formula:

$$f_{\text{data}}(n) = \frac{f_{\text{max}}}{\text{NoC} \cdot n} \quad (6)$$

where f_{max} is the maximum clock frequency of the μC equal 16 MHz for the AVR and 72 MHz for the ARM. NoC is the number of all cycles of the μC required for processing a single vector X in the network with n neurons. Note that the number of the clock cycles in the AVR and the ARM μCs required to complete the same task may differ. The ARM μCs are more efficient, so although f_{max} is in this case 5 times larger, the network is more than 7 times faster than in case of the AVR μC .

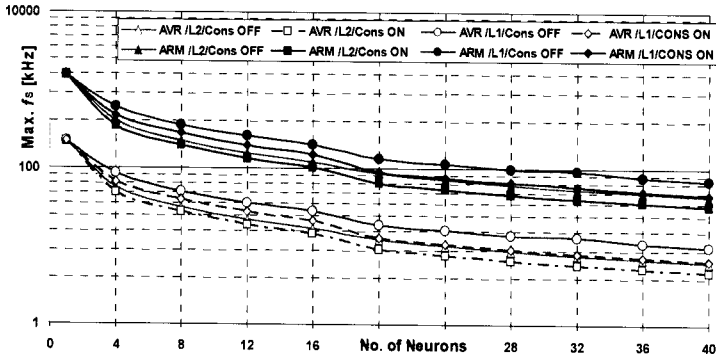


Fig. 4. Achievable maximum data rate of the WTA NN vs. number of neurons (for 3 inputs) for both microcontrollers

The power dissipation of particular system components is shown in Fig. 5. This parameter has been measured for a full performance of the μ Cs, when the network operates at the highest possible data rate. Comparison of the results for the L1/L2 metrics shows that in the first case the achievable data rate is almost doubled for the same number of neurons, while the learning accuracy is maintained. This shows that elimination of the multipliers at least at the stage of the distance calculation is a good idea.

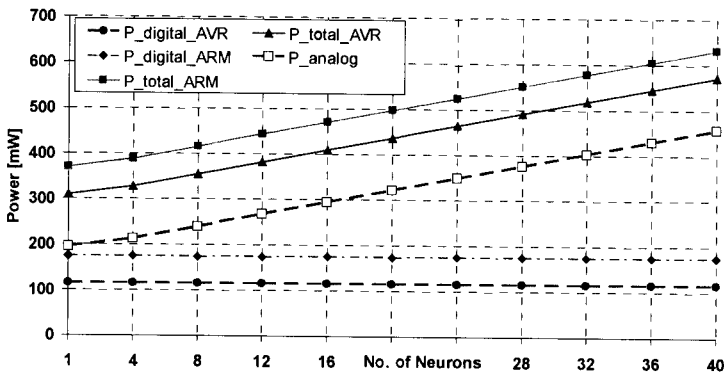
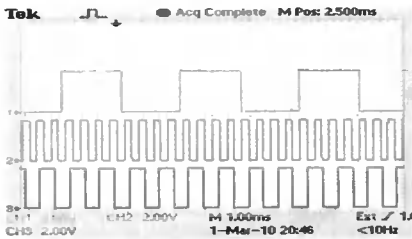
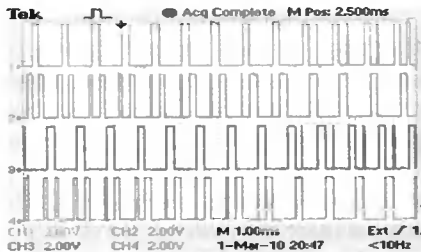


Fig. 5. Estimated power dissipation of the WTA NN vs. the number of neurons for 3 inputs for both used microcontrollers

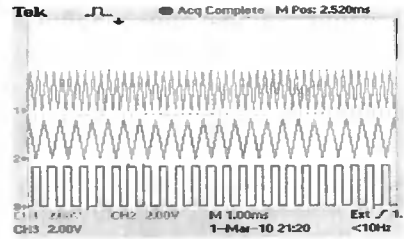
Figure 6 presents selected measurement results of the network with 3 inputs and 4 outputs realized using the ARM μ C sampled at 400 kHz. The results are shown for two different settings of the learning process i.e. for the conscience mechanism being turned on (left) and off (right). When this mechanism was turned-off one of the neurons remained inactive for presented input data.



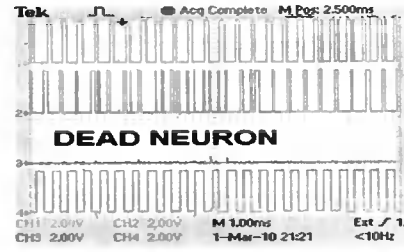
Learning signals (X) – Example 1



Neurons output signals – Example 1



Learning signals (X) – Example 2



Neurons output signals – Example 2

Fig. 6. Selected measurement results of the WTA NN with 3 inputs and 4 outputs. Example 1: the conscience mechanism (Cons) is on; Example 2: the Cons block is off

One of the objectives of this paper was the comparison of the implementation based on μC with the earlier analog realization. The analog NN with the same number of weights, sampled at 1 MHz, dissipated power of 700 μW . This is more than 500 times less than in case of the realization on the μCs . Taking into account also the sampling frequency, it can be demonstrated that the analog network can be even 1000 times more efficient. The main parameters of particular implementations have been collected in Table 1.

Table 1. Comparison between different realizations of WTA NN

	AVR/L1	AVR/L2	ARM/L1	ARM/L2	Analog/L2	PC
P [mW]	310	310	390	390	0.7	2e04
f_s max [kHz]	71	48	480	320	1000	1800
FOM (f_s/P)	0.23	0.15	1.23	0.82	1428	0.09
device sizes	3x4 cm (board with one μC)				1 x 1.5 mm	Large
Price (€)	40 (with one μC)				5 (long series)	c. 300

The results presented for the PC are estimated on the basis of the simulation results of the network model implemented in C++. If the NN is realized as ASIC or on the μC , as opposed to the PC based realizations, the power dissipation can be controlled in such a way to be approximately linearly dependent on the sampling frequency. This feature is an very important advantage, especially in the wearable systems, for example in Wireless Sensor Networks (WSN), which operate in environments with limited access to energy sources [18].

6. SUMMARY

Two different hardware realizations of the Winner Takes All (WTA) Artificial Neural Network (NN) have been compared in the paper. One of them is an analog network designed earlier by the authors in “full-custom” style in the CMOS 180 nm technology. The second implementation, proposed and described in this paper is based on two microcontrollers with the AVR and the ARM cores.

Both microcontrollers are placed on a single testing board, together with the ADC/DAC blocks, the power supply block. This makes the board a fully autonomous system, with the built-in learning abilities.

The measurement results of the prototype devices show that the μ C-based implementation is even ten times more efficient than the PC-based realizations, considering such criteria as the achievable data rate vs. power dissipation. On the other hand the analog networks is thousand times more efficient than the μ C-based network, occupying the area less than 1 mm².

The main disadvantage of the full-custom, transistor level realization is relatively long design process and high fabrication costs in case of short series. The μ C-based realization offers a low cost of the device, medium sizes and large flexibility. If only one μ C will be used, e.g. with the ARM core, the cost will not exceed 40 Euro per piece. The overall device sizes of 3 x 4 cm, achievable in this case, make the proposed system suitable for various portable applications, including medical diagnostic tools.

The WTA neural network presented in this paper is going to be used as a base station in wireless body sensor network for the on-line analysis of various biomedical data. To enable such option the final device will be equipped with the filters for data preprocessing and the wireless communication module to enable communication with the low power sensors placed on the patient's body. The platform is still being developed.

BIBLIOGRAPHY

- [1] Długosz R., Talaška T., Dalecki J., Wojtyna R., 2008. Experimental Kohonen neural network implemented in CMOS 0.18 μ m technology, International Conference on Mixed Design of Integrated Circuits and Systems, pp. 243-248, Poland.
- [2] Kohonen T., 2001. Self-Organizing Maps, Springer Verlag, Berlin.
- [3] Mailachalama B., Srikanthan T., 2002. Area-time issues in the VLSI implementation of self organizing map neural networks, Microprocessors and Microsystems, Vol. 26, No. 9-10, pp. 399-406.
- [4] Cotton N.J., Wilamowski B.M., Dundar G., 2008. A Neural Network Implementation on an Inexpensive Eight Bit Microcontroller, International Conference on Intelligent Engineering Systems, USA, pp. 109-114.
- [5] Young Wung Kim, Jung Hwan Cho, Gi Joon Jeon, 2007. An Intelligent Wireless Electronic Nose Node for Monitoring Gas Mixtures Using Neuro-Fuzzy Networks Implemented on a Microcontroller, IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, Italy, pp.100-104.
- [6] Mousa H.M., El-Bardini M.A., Koutb M.A., 2005. On-Line Neurocontroller Based on Microcontrollers, IEEE International Conference on Industrial Technology, Hong-Kong, pp. 1252-1256.

- [7] Mursalin T.E., Eishita F.Z., Islam A.R., 2008. Fabric defect inspection system using neural network and microcontroller, *Journal of Theoretical and Applied Information Technology*, Vol. 4, No. 7.
- [8] Amerijckx C., Verleysen M., Thissen P., Legat J., 1998. Image Compression by Self-Organized Kohonen Map, *IEEE Transactions on Neural Networks*, Vol. 9, No. 3, pp. 503-507.
- [9] Chang C., Xu P., Rui Xiao, Srikanthan T., 2005. New Adaptive Color Quantization Method Based on Self-Organizing Maps, *IEEE Transactions on Neural Networks*, Vol. 16, No. 1, pp. 237-249.
- [10] Pedrycz W., Vasilakos A., 2001. *Computational intelligence in telecommunications networks*, CRC Press, LLC.
- [11] Kohonen T., Somervuo P., 1998. Self-Organizing Maps of Symbol Strings with Application to Speech Recognition, *Neurocomputing*, Vol. 21, No. 1-3, pp. 19-30.
- [12] Talaśka T., Długosz R., 2008. Initialization mechanism in Kohonen neural network implemented in CMOS technology, *European Symposium on Artificial Neural Networks (ESANN)*, Belgium, pp. 337-342.
- [13] Ahalt S.C., Krishnamurthy A.K., Chen P., Melton D.E., 1990. Competitive learning algorithms for vector quantization, *Neural Networks*, Vol. 3, pp. 131-134.
- [14] DeSieno D., 1988. Adding a conscience to competitive learning, *IEEE Conference on Neural Network*, Vol. 1, pp. 117-124.
- [15] Długosz R., Talaśka T., Pedrycz W., Wojtyna R., 2010. Realization of a Conscience Mechanism in CMOS Implementation of Winner Takes All Neural Networks, *IEEE Transactions on Neural Networks*, Vol. 21, No. 6, pp. 961-971.
- [16] Rajah A., Khalil Han M., 2004. ASIC design of a Kohonen Neural Network microchip, *IEEE International Conference on Semiconductor Electronics*, Kuala Lumpur, Malaysia, pp. 148-151.
- [17] Chung-Yu Wu., Kuo Wen-Kai, 1993. A new analog implementation of the Kohonen Neural Network, *International Symposium on VLSI Technology, Systems, and Applications*, Taipei, Taiwan, pp. 262-266.
- [18] Dubois P., Botteron C., Mitev V., Menon C., Farine P.-A., Dainesi P., Ionescu A., Shea H., 2009. Ad hoc wireless sensor networks for exploration of Solar-system bodies, *Acta Astronautica*, 64 (5-6), pp. 626-643.

PORÓWNANIE RÓŻNYCH SPRZĘTOWYCH REALIZACJI SZTUCZNEJ SIECI NEURONOWEJ TYPU WINNER TAKES ALL

Streszczenie

W pracy przedstawiono projekt oraz wyniki badań laboratoryjnych sieci neuronowej Kohonena typu Winner Takes All (WTA) zaimplementowanej na mikrokontrolerach z rdzeniami AVR oraz ARM. W pracy przedstawiono też porównanie z wcześniejszą realizacją podobnej sieci jako specjalizowany analogowy układ scalony. Dwa mikrokontrolery, na których zaimplementowano algorytm uczący umieszczone zostały na jednej płytce testowej aby umożliwić bezpośrednie porównanie ich parametrów. Za pomocą przełączników umieszczonych bezpośrednio na płytce możliwe jest wybranie jednego z mikrokontrolerów, jednej z dwóch miar podobieństwa między wektorami (Euklidesa L2 lub typu Manhattan L1) oraz włączenie lub wyłączenie mechanizmu sumienia. Niektóre sygnały przedstawiające proces uczenia (sygnału sygnalizującego

zwyczajni neuron) możemy bezpośrednio obserwować na płytce. Proces uczenia możemy też w całości obserwować na komputerze PC, poprzez złącze USB. Do potencjalnych zastosowań wykonanej płytki testowej oraz sprzętowych realizacji sieci neuronowej należą systemy do ciągłego monitoringu zdrowia pacjentów (obserwacja oraz analiza sygnałów typu EKG oraz EMG), a także jako wyposażenie laboratorium studenckiego.

Słowa kluczowe: sieć typu WTA, cyfrowe sieci neuronowe, analogowe sieci neuronowe, mikrokontrolery, niski pobór energii

PETRI NETS AND ACTIVITY DIAGRAMS IN LOGIC CONTROLLER SPECIFICATION – TRANSFORMATION AND VERIFICATION

Iwona Grobelna, Michał Grobelny, Marian Adamski

Institute of Computer Eng. And Electronics
University of Zielona Góra
Zielona Góra, Poland

{i.grobelna, m.adamski}@iie.uz.zgora.pl, m.grobelny@weit.uz.zgora.pl

Summary: The paper presents formal verification method of logic controller specification taking into account user-specified properties. Logic controller specification may be expressed as Petri net or UML 2.0 Activity Diagram. Activity Diagrams seem to be more user-friendly and easy-understanding than Petri nets. Specification in form of activity diagram may afterwards be transformed into Petri net, which may then be formally verified and used to automatically generate implementation (code). A new transformation method dedicated for event-driven systems is proposed. Verification process is executed automatically by the NuSMV model checker tool. Model description based on specification and properties list is being built. Model description derived from Petri net is presented in RTL-level and easy to synthesize as reconfigurable logic controller or PLC. Properties are defined using temporal logic. In model checking process, verification tool checks whether requirements are satisfied in attached system model. If this is not the case, appropriate counterexamples are generated.

Keywords: formal verification, logic controller, model checking, Petri nets, UML Activity Diagrams

1. INTRODUCTION

Logic controller specification is the first step in the design and development process. It is therefore especially important, that the specification meets user-defined requirements. The specification may be formalized in different forms [12], e.g. by means of Petri Nets or, what may seem more user-friendly, by means of UML 2.0 Activity Diagrams. However, activity diagrams are not well supported by formal verification mechanisms. Nevertheless, they can be transformed into Petri nets which can be then formally verified for consistency between model description and requirements for its behavior. In the article a new transformation method dedicated for event-driven systems is proposed. Activity diagram action nodes are interpreted as Petri net transitions, unlike classical approaches in previous versions of UML where action nodes were interpreted as Petri net places.

Model checking of prepared specification allows to early detect subtle errors resulting from wrong specification interpretation. It is one of formal verification methods

among others like e.g. theorem proving [7] [16]. The paper focuses on a new logical model which derives from Petri net and is presented in RTL-level in such a way that it is easy to synthesize as reconfigurable logic controller or PLC

The article is structured as follows. Section 2 presents some background on formal mechanisms needed to specify logic controller behavior, Petri nets and UML 2.0 Activity Diagrams, and a formal mathematical system used in requirements specification. Section 3 concentrates on transformation aspects from activity diagram into Petri net specification. Section 4 focuses on model checking of formal specification in form of Petri net. The article concludes with short summary and future research directions.

2. FORMAL SPECIFICATION BY MEANS OF PETRI NET AND UML 2.0 ACTIVITY DIAGRAM

This section includes some background on formal specification methods by means of Petri nets and UML 2.0 Activity Diagrams, and on temporal logic.

2.1. PETRI NETS

Petri nets [3] [4] [8] [12] were introduced in 1962 by Carl Adam Petri as a general purpose mathematical model for describing relations between conditions and events. They are currently used in many industrial branches for planning and controlling of production flow, design and programming of microprocessor controllers, system software synthesis, etc. There are some design tools available which allow to automatically generate code from Petri net specification [11].

Graphic representation of Petri net can be understood even by non-technical staff. It allows e.g. to specify such behaviors as parallelism and concurrency, choice, synchronization, memorizing, reading or resource sharing [8].

A Petri net is specified by places (represented by a circle) and transitions (represented by a bar or a box) connected together (represented by directed arcs which indicate relation flow, where places can be connected only to transitions, and transitions can be connected only to places). The number of places and transitions is finite and not zero. States are defined by tokens (represented by small full circles, also called markers) inside some places. A transition can be fired only if each of its input places contains at least one token. Then from each of its input places one token is being removed and added to each of its output places.

Control Interpreted Petri Nets [2] specify and model the behavior of concurrent logic controllers and take into account properties of controlled objects. Local states may change after firing of transitions if some events occur. Transition guards are associated with input signals of controller and places are associated with its output signals. Global state of logic controller is built of simultaneously holding local states.

2.2. UML 2.0 ACTIVITY DIAGRAMS

The Unified Modelling Language 2.0 notation [19] simplifies information flow between team members and enables easy understanding of system behavior by non-experienced staff. It was initially introduced for specification, visualization and documentation of software. However, behavioral embedded system design [4] can also be

facilitated by using some types of UML diagrams, like activity diagrams, state machines or sequence diagrams.

Activity diagrams are currently used in business domain, modeling of information flow [10] [23] and behavioral software and embedded systems design in software/hardware co-design [20].

Most commonly used parts of UML activity diagrams are action nodes (represented with rounded rectangles with the name of the action inside). The flow of activities is described using lines with arrowheads. Additionally every diagram should start with the initial node (big filled dot) and end with the final point (filled dot with a border). Usually embedded controllers or other discreet systems have parallelism in their description. It is possible to represent it using fork and join nodes (notated by horizontal or vertical bars).

2.3. TEMPORAL LOGIC

Temporal logic [5] [15] [17] introduced into computing science in 1977 by Amir Pnuelli derives from modal logic with possibility and necessity operators. Firstly it was used in concurrent and reactive systems. Currently it is used also in program specification, verification, synthesis and logical programming.

Classical temporal logic is Linear-time Temporal Logic (LTL). It describes relations in the system and state sequences. A formula can change any time its value, e.g. in some states it can be true, and in others it can be false. Basic operators are: *always* (G), *sometimes* (F) and *next* (X).

Temporal logic with time branches is Computation Tree Logic (CTL). Time is presented here as a tree branching out into the future with present moment as the root. Characteristic for branching time logics are path quantifiers: the E operator for some paths and the A operator for all paths. They are for paths beginning from a given state, while state quantifiers are for states in a path. State quantifiers are: the F operator for some states and the G operator for all states. Path and state quantifiers are mostly used together, i.e. $EF p$ means that in some paths in some states formula p is true and $AG p$ means that in all paths in all states formula p is true.

3. TRANSFORMATION

In this section a new transformation method dedicated for event-driven systems is presented. Firstly, an example is introduced, which will be used to better illustrate proposed transformation method.

As an example to present transformation method from UML 2.0 Activity Diagrams to Petri nets a simple embedded system for movement control of two vehicles [3] has been taken.

Initially, both vehicles are placed at starting points a and c . After pressing the m button, they begin to move to the right simultaneously. If both vehicles reach their ending points (point b for the first vehicle $W1$ and point d for the first vehicle $W2$) vehicle $W1$ returns to its starting point a . Afterwards, the second vehicle $W2$ returns to its starting point c . Then the process can be started again. The real model of described process is presented in Fig. 1.

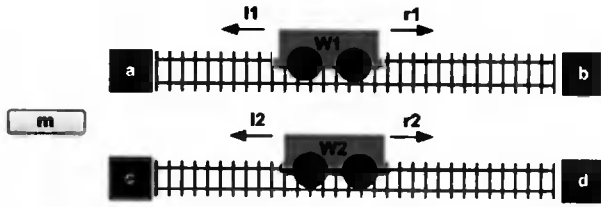


Fig. 1. Real model of analyzed process

The schema of logic controller with input and output signals is presented in Fig. 2. The signals are described in Table 1 and Table 2.

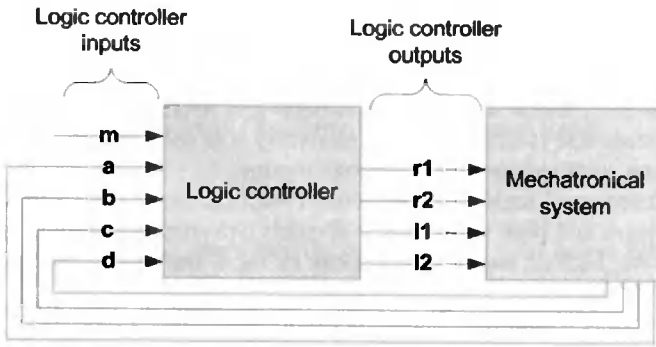


Fig. 2. Logic controller schema

Table 1. Input signals and their meaning

Input signal	Meaning
m	Signal to start the process.
a	The first vehicle <i>W1</i> is at its starting point <i>a</i> .
b	The first vehicle <i>W1</i> is at its ending point <i>b</i> .
c	The second vehicle <i>W2</i> is at its starting point <i>c</i> .
d	The second vehicle <i>W2</i> is at its ending point <i>d</i> .

Table 2. Output signals and their meaning

Output signal	Meaning
r1	The first vehicle <i>W1</i> moves to the right.
r2	The second vehicle <i>W2</i> moves to the right.
l1	The first vehicle <i>W1</i> moves to the left.
l2	The second vehicle <i>W2</i> moves to the left.

Specification of described process, presented in Fig. 3, was prepared by using UML 2.0 Activity Diagrams. Movements to the right are realized simultaneously, while movements to the left (returns) are realized sequentially. Actions are executed when conditions from square bracket are fulfilled (input signals from Table 1 take appropriate values). Values of output signals are specified inside action boxes. They are used for controlling of vehicles movements in the real world (Table 2).

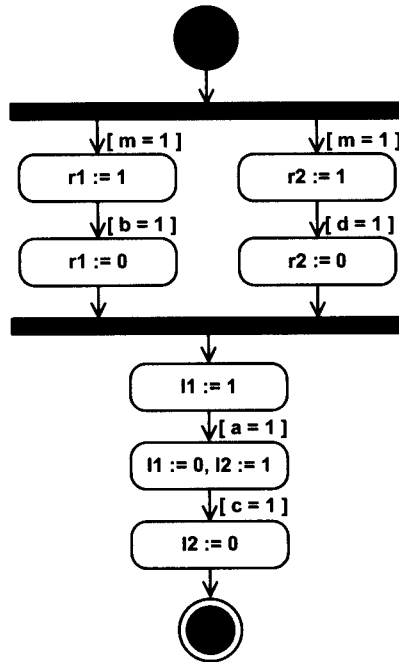


Fig. 3. Activity Diagram for analyzed process

For transformation purposes actions of activity diagrams are treated like transitions [21] [22]. In classical approaches in previous versions of UML action nodes were interpreted as Petri net places. Here, action nodes are interpreted as Petri net transitions. Fork and join nodes (notated by horizontal or vertical bars) are interpreted as fork or join transitions in Petri net. Behavior of controlling process is presented step by step, basing on the interpretation in form of activity diagram. The starting and ending points of activity diagrams refer to appropriate places ($P1$ and $P17$) in Petri net. Additionally, input conditions for the actions were treated like decision blocks before actions. Therefore, each input condition in activity diagram is assigned a transition in Petri net with appropriate firing condition. Additional synchronization places ($P10$ and $P11$) are necessary, because UML syntax enforces synchronization in join node.

Petri net after direct transformation includes 17 places and 16 transitions (Fig. 4a). However, some places and transitions are redundant. The model compression resulting in reduction of unnecessary delays in circuit performance is executed. The proposed reduction method assumes replacing of transition with condition and transition with action with one transition. This procedure is connected with deleting of unnecessary places. As an example, place $P4$ and transition $T4$ may be deleted and their etiquettes may be moved to transition $T2$. Transition $T4$ reflects assigning of value 1 to signal $r1$. According to the activity diagram (Fig. 3) the action may be executed only if signal m is active. It can be therefore assigned to transition which checks the current value of the m signal. After reduction of redundant places and transitions Petri net (Fig. 4b) has 10 places and 9 transitions (amount of places and transitions has been decreased by ca. 40%).

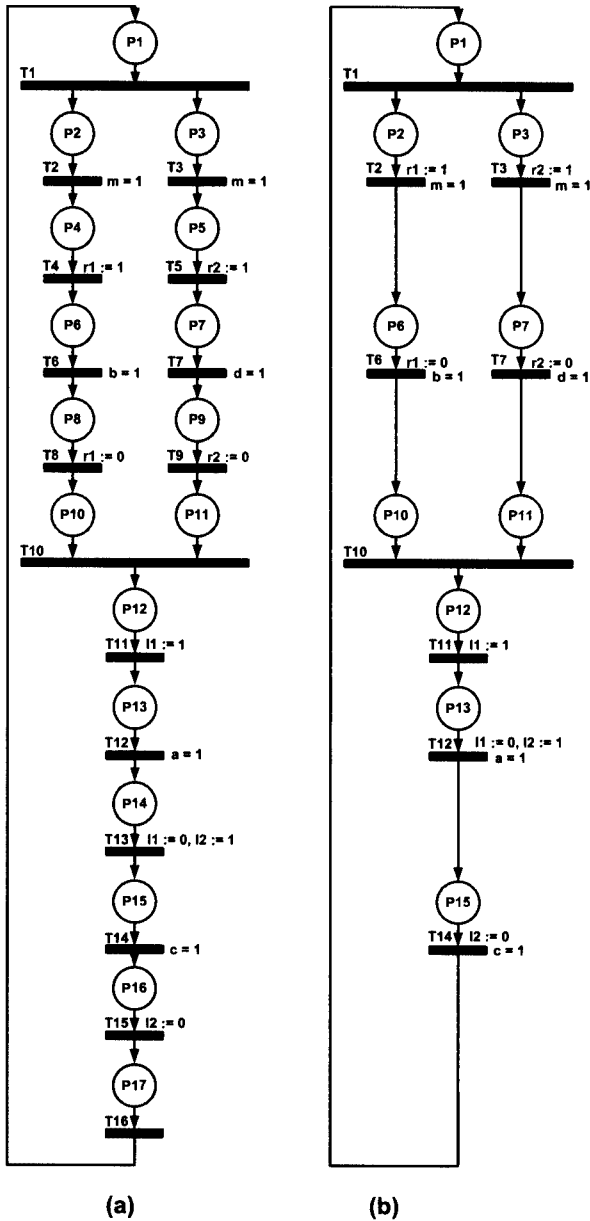


Fig. 4. Petri net after direct transformation from activity diagram (a) and after reduction of redundant places (b)

4. MODEL CHECKING

Model checking technique enables formal verification of logic controller specification. Specification can be checked against behavioral requirements which have to be fulfilled.

First of all, description and requirements list have to be delivered to a model checker tool. A system to be verified should be modeled using the description language of the particular model checker, in our case it is a symbolic model checker NuSMV in the current version 2.4.3 [6] [9]. Requirements list with defined properties should be coded using the specification language of particular model checker. The list should include as many desired properties as possible as only they will be checked. Finally model checker verifies the system and gives an answer whether described model satisfies the specification. In case of detected errors user receives feedback with appropriate counterexamples. What is important is the fact that model checking can be used to verify the whole system or only some part of it. Partial verification is especially valuable in large systems, where the design process is complex and long, as it can be performed step-by-step during the design phase considering each time only a limited subset of requirements.

Design requirements specified by Petri net have to be transformed into the format of the NuSMV model checker. Then the specification can be verified against requirements defined with linear-time temporal logic. Also other specification forms can be formally verified by the model checker tool, an example of algorithmic state machine verification can be found in [13].

The proposed model description derives from Petri net. It is presented in RTL-level (*Register Transfer Level*) in such a way that it is easy to synthesize as reconfigurable logic controller or PLC without any additional changes.

Model description can be prepared as shown in subsections 4.1, 4.2 and 4.3. Requirements list can be defined as presented in subsection 4.4. Subsection 4.5 concentrates on model checking process itself, while subsection 4.6 focuses on the results of performed formal verification.

4.1. VARIABLES DEFINITION

Variables definition (Appendix, lines 2-9) includes global states, input and output signals. Another example of Petri net verification idea is presented in [14] where the verification process is treated more example-specific and focuses rather on controlled objects and its statuses than on global system states.

It is assumed that in one moment only one input signal can be active. Therefore, there is only one variable defined which takes any of possible input signals as its value (the amount of them equals the amount of rows in Table 1 with extra value for no active input signals added).

The number of output signals equals the number of rows in Table 2, where each signal is an independent variable and can be either active or inactive.

Global states are formed of local states from Petri net presented in Fig. 4b. The firing time of transitions is extremely short and only one transition may be fired at particular time unit, what corresponds to one of the important functioning rules of formal model from [1]. However, the nondeterministic automaton with global states of received Control Interpreted Petri Net can be simplified in such a way, that after global state $P2P3$, state $P6P7$ will be reached (transitions $T2$ and $T3$ have the same firing condition so the firing time will be just one after the other and global states $P2P7$ or $P3P6$ would last the minimum amount of time).

System state is defined by the combination of variable values. In the NuSMV system model of described example, there exists 864 possible combinations, but only 24 of them are available.

4.2. INITIAL VALUES OF VARIABLES

Initially, all variables have some default values assigned (Appendix, lines 11-16). The initial values are changing according to the rules defined next.

4.3. ASSIGNMENT OF VALUES TO VARIABLES

All assignments of next values (Appendix, lines 17-60) happen simultaneously. To make the model simpler, in the sample model description no emergency situations have been taken into account. For example, if a vehicle gets stuck during the movement, it will be necessary to push the vehicles by hand to the starting points and start the process again from the beginning.

To simulate the behavior of real system, values of input signals are chosen randomly, but only expected input signals may be active at particular time.

4.4. REQUIREMENTS LIST

Requirements list may be defined by using either Linear-time Temporal Logic LTL or Computation Tree Logic CTL. The second one is better for verification of nondeterministic programs [18]. We have chosen the first one to define behavioral requirements of the designed logic controller. Among the properties there are some safety properties (situations which must not happen) and liveness properties (situations which have to happen).

Required properties (Appendix, lines 61-82) examine behavior of the designed logic controller mainly after occurrence of some input signals. Let us explain some defined properties. The property in lines 67-68 states that always after the occurrence of m input signal, in the next state output signals $r1$ and $r2$ will be assigned value 1 , what means that both vehicles will move to the right. Next property (lines 69-70) indicates that it should never be the case that both output signals $r1$ and $l1$ is assigned at the same time value 1 , what means that the first vehicle can not move to the right and to the left (and as the result stay in one place if the forces for moving to the right and moving to the left are equal) at the same time.

4.5. MODEL CHECKING PROCESS

Model description and requirements list are input data for the NuSMV model checker. The tool compares them and generates an answer whether required properties are satisfied in delivered model description.

4.6. RESULTS

After successfully verification a short report is generated (parts of it are presented in Fig. 5 and Fig. 6). It includes list of checked properties with status whether they are satisfied in the model description or not. If any of the requirements is not satisfied, appropriate counterexample is generated. The counterexample presents a trace which may be used by the designer to follow the incorrect situation.

Sample requirements which are satisfied in the described example are listed below (Fig. 5).

```

-- specification G (input = m -> X (r1 = 1 & r2 = 1)) is true
-- specification G !(l1 = 1 & r1 = 1) is true
-- specification G !(l2 = 1 & r2 = 1) is true
-- specification G (input = b -> X r1 = 0) is true
-- specification G (input = d -> X r2 = 0) is true

```

Fig. 5. Satisfied requirements

From the defined requirements a sample property which can not be satisfied (lines 73-74) indicates that always after occurrence of *b* input signal (the first vehicle reached its ending point *b*) finally the output signal *l1* will be assigned value *1* (the first vehicle will finally return to its starting point *a*). This however can not be guaranteed as it may be the case that the second vehicle never reaches its ending point *d* and so the whole system will remain in global state *P7P10* of the Petri net from Fig. 4b. The situation trace is presented in a generated counterexample (Fig. 6).

```

-- specification G (input = b -> F l1 = 1) is false
-- as demonstrated by the following execution sequence
Trace Description: LTL Counterexample
Trace Type: Counterexample
-> State: 3.1 <-
  state = p1
  input = none
  r1 = 0
  r2 = 0
  l1 = 0
  l2 = 0
-> Input: 3.2 <-
-> State: 3.2 <-
  state = p2p3
-> Input: 3.3 <-
-> State: 3.3 <-
  input = m
-> Input: 3.4 <-
-> State: 3.4 <-
  state = p6p7
  r1 = 1
  r2 = 1
-> Input: 3.5 <-
-> State: 3.5 <-
  Input = b
-> Input: 3.6 <-
-- Loop starts here
-> State: 3.6 <-
  state = p7p10
  input = none
  r1 = 0
-> Input: 3.7 <-
-> State: 3.7 <-

```

Fig. 6. Generated counterexample

5. CONCLUSION

Formally specifying system behavior using UML 2.0 Activity Diagrams is an easy and efficient way to document initial negotiation results between customer and supplier.

The specification is then easy-understandable both for engineers and for non-technical partners. On the other hand, Petri nets seem to be better suited for logic controllers and are currently commonly used in the industry. Furthermore, possibility of implementation (code) generation from Petri net specification and insufficient verification methods of UML Activity Diagrams also have to be taken into account. These aspects suggest the solution of transformation from activity diagram to Petri net specification.

Model checking technique is very valuable for formal verification of designed systems. Although it can not prove that the system model is completely correct, it can prove that it has or has not some user-specified desired properties. An advantage is also the fact that model correctness can be verified before the real system physically exists. Therefore it can potentially prevent errors on an early stage of system development.

However, human interaction is especially needed by counterexamples analysis. Generation of them can be caused either by false model description or by invalid requirements specification and every counterexample has to be carefully analyzed.

Future research directions focus on the improvement of transformation from UML 2.0 Activity Diagrams into Petri nets. The aim is to make the transformation fully automatic so that the outgoing specification form is compact and its interpretation corresponds to the interpretation by means of initial activity diagram. Other research directions concentrate on model checking technique and its application to Petri nets specifications. Different transformation methods from Petri net into description format of the NuSMV model checker are being examined.

APPENDIX

Model description with requirements list for discussed example is attached.

```

1. MODULE main
2. VAR
3.     state: {p1, p2p3, p6p7, p6p11, p7p10,
4.           p10p11, p12, p13, p15};
5.     input: {none, m, a, b, c, d};
6.     r1 : {1, 0};
7.     r2 : {1, 0};
8.     l1 : {1, 0};
9.     l2 : {1, 0};
10. ASSIGN
11.     init(state) := p1;
12.     init(input) := none;
13.     init(r1) := 0;
14.     init(r2) := 0;
15.     init(l1) := 0;
16.     init(l2) := 0;
17.     next(state) := case
18.         state = p1 : p2p3;
19.         state = p2p3 & input = m : p6p7;
20.         state = p6p7 & input = b : p7p10;
21.         state = p6p7 & input = d : p6p11;
22.         state = p7p10 & input = d : p10p11;
23.         state = p6p11 & input = b : p10p11;
24.         state = p10p11 : p12;
25.         state = p12 : p13;
26.         state = p13 & input = a : p15;
27.         state = p15 & input = c : p1;

```

```

28.     1 : state;
29.     esac;
30.     next(input) := case
31.         state = p2p3 : {none, m};
32.         state = p6p7 : {none, b, d};
33.         state = p6p11 : {none, b};
34.         state = p7p10 : {none, d};
35.         state = p13 : {none, a};
36.         state = p15 : {none, c};
37.         1 : none;
38.     esac;
39.     next(r1) := case
40.         state = p2p3 & input = m : 1;
41.         state = p6p7 & input = b : 0;
42.         state = p6p11 & input = b : 0;
43.         1 : r1;
44.     esac;
45.     next(r2) := case
46.         state = p2p3 & input = m : 1;
47.         state = p6p7 & input = d : 0;
48.         state = p7p10 & input = d : 0;
49.         1 : r2;
50.     esac;
51.     next(l1) := case
52.         state = p13 & input != a : 1;
53.         state = p13 & input = a : 0;
54.         1 : l1;
55.     esac;
56.     next(l2) := case
57.         state = p13 & input = a : 1;
58.         state = p15 & input = c : 0;
59.         1 : l2;
60.     esac;
61. LTLSPEC
62.     F (state = p1);
63. LTLSPEC
64.     F (state = p15);
65. LTLSPEC
66.     G (input = b -> F(state = p7p10));
67. LTLSPEC
68.     G (input = m -> X(r1 = 1 & r2 = 1));
69. LTLSPEC
70.     G !(l1 = 1 & r1 = 1);
71. LTLSPEC
72.     G !(l2 = 1 & r2 = 1);
73. LTLSPEC
74.     G (input = b -> F(l1 = 1));
75. LTLSPEC
76.     G (input = b -> X (r1 = 0));
77. LTLSPEC
78.     G (input = d -> X (r2 = 0));
79. LTLSPEC
80.     G (input = a -> X (l1 = 0));
81. LTLSPEC
82.     G (input = c -> X (l2 = 0));

```

BIBLIOGRAPHY

- [1] Andreu D., Souquet G., Gil T., 2008. Petri Net based rapid prototyping of digital complex system, Symposium on VLSI, IEEE Computer Society Annual, pp. 405-410.
- [2] Adamski M., 2001. A rigorous design methodology for reprogrammable logic controllers, The International Workshop on Discrete-Event System Design, DESDes'01, Przytok, Poland.
- [3] Adamski M., Chodań M., 2000. Modelowanie układów sterowania dyskretnego z wykorzystaniem sieci SFC, Wydawnictwo Politechniki Zielonogórskiej, Zielona Góra (in Polish).
- [4] Adamski M., Karatkevich A., Węgrzyn M. (ed.), 2005. Design of embedded control systems, Springer (USA).
- [5] Ben-Ari M., 2005. Logika matematyczna w informatyce. Klasyka informatyki, Wydawnictwa Naukowo-Techniczne, Warszawa (in Polish).
- [6] Cavada R. et al. NuSMV 2.4 User Manual, downloaded from <http://nusmv.fbk.eu/>
- [7] Clarke E.M., Wind J.M. et al, 1996. Formal methods: State of the Art and Future Directions, ACM Computing Surveys, Vol. 28, No. 4.
- [8] David R., Alla H., 1992. Petri Nets & Grafctet. Tools for modelling discrete event systems, Prentice Hall.
- [9] Eshuis R., 2006. Symbolic Model Checking of UML Activity Diagrams, ACM Transactions on Software Engineering and Methodology, Vol. 15, No. 1, pp. 1-38.
- [10] Eshuis R., Wieringa R., 2001. A Comparison of Petri Net and Activity Diagram Variants, Proc. of 2nd Int. Coll. on Petri Net Technologies for Modelling Communication Based Systems, pp. 93-104.
- [11] Frey G., Litz L., 1998. Verification and Validation of Control Algorithms by Coupling of Interpreted Petri Nets, Proceedings of the IEEE SMC'98, Vol. 1, pp. 7-12.
- [12] Gomes L., Barros J.P., Costa A., 2006. Modeling Formalisms for Embedded System Design, Embedded Systems Handbook, Taylor & Francis Group, LLC.
- [13] Grobelna I., 2008. Formalna analiza interpretowanych algorytmicznych maszyn stanów ASM z wykorzystaniem narzędzia model checker, Metody Informatyki Stosowanej nr 3, Tom 16, pp. 107-124 (in Polish).
- [14] Grobelna I., 2008. Formal verification of logic controller specification using NuSMV model checker, X Międzynarodowe Warsztaty Doktoranckie OWD'2008, Archiwum konferencji PTETIS, Vol. 25, pp. 459-464.
- [15] Huth M., Ryan M., 2004. Logic in Computer Science. Modelling and Reasoning about Systems, Cambridge University Press.
- [16] Kern C., Greenstreet M.R., 1999. Formal Verification in Hardware Design: A Survey, ACM Transactions on Design Automation of Electronic Systems (TODAES), Vol. 4, Issue 2, pp. 123-193.
- [17] Klimek R., 1999. Wprowadzenie do logiki temporalnej, AGH Uczelniane Wydawnictwa Naukowo-Dydaktyczne, Kraków (in Polish).
- [18] Lamport L., 1980. "Sometime" is sometimes "not never", On the Temporal Logic of Programs, Proceedings of the Seventh ACM Symposium on Principles of Programming Languages, ACM SIGACT-SIGPLAN, pp. 174-185.
- [19] <http://www.omg.org>
- [20] Schattkowsky T. UML 2.0 – Overview and Perspectives in SoC Design, Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE'05).

- [21] Staines T.S., 2008. Intuitive Mapping of UML 2 Activity Diagrams into Fundamental Modeling Concept Petri Net Diagrams and Colored Petri Nets, 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, pp. 191-200.
- [22] Tričković I., 2000. Formalizing activity diagram of UML by Petri nets, Novi Sad J. Math, Vol. 30, No. 3, pp. 161-171.
- [23] Yen-Liang C., Sammy C., Chyun-Chyi C., Irene C., 2000. Workflow Process Definition and Their Applications in e-Commerce, IEEE 2000, pp. 193-200.

SIECI PETRIEGO I DIAGRAMY AKTYWNOŚCI W SPECYFIKACJI STEROWNIKÓW LOGICZNYCH – TRANSFORMACJA I WERYFIKACJA

Streszczenie

Praca prezentuje metodę formalnej weryfikacji specyfikacji sterownika logicznego uwzględniającą właściwości podane przez użytkownika. Specyfikacja sterownika logicznego może być przedstawiona m.in. w postaci sieci Petriego lub diagramu aktywności języka UML. Diagramy aktywności wydają się być bardziej przyjazne i zrozumiałe dla użytkownika niż sieci Petriego. Specyfikacja w postaci diagramu aktywności może zostać przekształcona do sieci Petriego, która następnie może być formalnie zweryfikowana i wykorzystana do automatycznej generacji implementacji (kodu). Węzły diagramu aktywności konsekwentnie interpretowane są jako tranzycje sieci Petriego, w odróżnieniu od klasycznego podejścia (w starszych wersjach UML) gdzie odwzorowywało się je jako miejsca sieci Petriego. Proces weryfikacji wykonywany jest automatycznie przez narzędzia weryfikacji modelowej. Tworzony jest opis modelu bazujący na specyfikacji oraz lista wymagań. Nowatorskim podejściem jest przedstawienie sieci Petriego na poziomie RTL w taki sposób, że łatwo jest przeprowadzić syntezę logiczną sieci w postaci współbieżnego rekonfigurowalnego sterownika logicznego lub sterownika PLC bez konieczności przekształcania modelu. Wymagania określone są przy użyciu logiki temporalnej. W procesie weryfikacji modelowej narzędzie weryfikujące NuSMV sprawdza, czy model systemu spełnia stawiane mu wymagania. Jeżeli tak nie jest, generowany jest odpowiedni kontrprzykład.

Słowa kluczowe: formalna weryfikacja, sterownik logiczny, weryfikacja modelowa, sieci Petriego, diagramy aktywności UML

IMPLEMENTATION OF THE DIRECT WINDOWED DHT PARALLEL ALGORITHM

Włodzimierz Pogribny, Mariusz Sulima

Faculty of Telecommunications and Electrical Engineering
University of Technology and Life Sciences (UTP)
7 Kaliskiego Ave., Bydgoszcz 85-789, Poland

Summary: Hardware realizations of an algorithm of discrete Hilbert transform (DHT) are made mainly with using the FFT because of its higher resolution [1] [4] in comparison to the direct DHT based on unwindowed impulse response (IR). On the other hand, as it is shown previously in [5] [6] the appropriate use of weight windows for IR provides a higher resolution of the direct DHT in comparison to the DHT method with FFT using. In this work a parallel algorithm and appropriate filter structure for the direct windowed DHT which are faster than the algorithms and structures based on the FFT are presented. The work of the algorithm and filter is illustrated by examples of phase transitions capturing of noisy signals.

Keywords: Discrete Hilbert Transform, DHT, Phase recognition

1. INTRODUCTION

Discrete Hilbert transform for a signal with finite energy can be calculated basing on different algorithms, among which the most widely used algorithm is based on the use of the FFT [7] [8]. The downside of DHT based on FFT is the large number of multiplication operations which limits its use in the processing of broadband signals. On the other hand, the predominance of parallel algorithms of the direct DHT is higher speed and this advantage increases by growing of the number N of samples. For selected types of signals, by applying appropriate weight windows, direct DHT has a higher resolution in comparison to DHT based on FFT [5] [6].

The DHT result of the time series is imaginary one that allows to form an analytical signal from which the amplitudes and phases can be obtained:

$$\left\{ |z|_t \right\}_{t=-N}^N = \left\{ \sqrt{x_t^2 + \tilde{x}_t^2} \right\}_{t=-N}^N \quad (1)$$

$$\left\{ \varphi_t \right\}_{t=-N}^N = \left\{ \arctg \left(\frac{\tilde{x}_t}{x_t} \right) \right\}_{t=-N}^N \quad (2)$$

It is well known fact that the instantaneous frequency is a derivative of the phase which together with (1) allows to realize appropriate kind of the time-frequency analy-

sis [2]. Phase spectrum is often used to identify different processes and physical states, in particular in telecommunications and medicine [3]. However, up to now the application of such algorithms is not sufficiently studied. Therefore purpose of this work is studying of application of the windowed DHT parallel algorithm and structure for signal analyzing especially for determining of phase transitions of noisy signals.

2. DHT METHOD BASED ON FFT

This method uses the property of the harmonic signal DHT based on obtaining of a harmonic signal with phase shifted on 90° and in case of a complex structure signal – by shifting each of its harmonics. Harmonic components are obtained by applying the FFT in order to receive the Fourier series:

$$x(t) = U_0 + \sum_{n=1}^N U_n \cos(n\omega_0 t + \varphi_n) \quad (3)$$

where:

- U_0 – constant component,
- U_n – the amplitude of the n-harmonic,
- ω_0 – angular frequency of the lowest harmonic frequency,
- φ_n – the initial phase of the n-th harmonic.

Ultimately, the result of DHT based on FFT is obtained by IFFT the shifted harmonics:

$$\tilde{x}(t) = \sum_{n=1}^N U_n [\cos(n\omega_0 t + \varphi_n - 90^\circ)] \quad (4)$$

In (4) the constant component is omitted, since the Hilbert transform of constant function is 0.

3. THE METHOD OF DIRECT DHT

Variance of direct DHT was formulated in [1] and [7] as a convolution of the input signal samples and the impulse response (IR):

$$\tilde{x}_n = \sum_{m=-N}^N x_m h_{n-m} \quad (5)$$

In these works the use of unwindowed IR was proposed in next way:

$$h_{n-m} = \begin{cases} \frac{1}{\pi} \frac{1}{n-m}, & n \neq m \\ 0, & n = m \end{cases} \quad (6)$$

A different variance of DHT was formulated by Oppenheim in the book [4]. He defined the Hilbert filter IR weight factors in the form:

$$h_{n-m} = \begin{cases} \frac{2}{\pi} \frac{\sin^2\left(\frac{\pi(n-m)}{2}\right)}{n-m}, & n \neq m \\ 0, & n = m \end{cases} \quad (7)$$

In all cases it is assumed by the Cauchy principal value that the IR factors $\{h_i\}_{i=-2N}^{2N}$ take the value zero and not infinite for all indices $n = m$.

4. THE WORKED OUT WEIGHT WINDOW FOR DIRECT DHT

In this paper, it is proposed to obtain the result of DHT with using of the windowed IR on the basis of the window being worked out by the authors [5]:

$$\tilde{x}_n = \frac{1}{\pi} P \sum_{m=-N}^N \frac{x_m w_{n-m}}{n-m}, \quad (8)$$

where:

$$m, n = \overline{-N, N}.$$

Then the windowed IR $\{h_i^{(w)}\}_{i=-2N}^{2N}$ such a filter takes the values:

$$h_{n-m}^{(w)} = \begin{cases} \frac{w_{n-m}}{n-m}, & m \neq n \\ 0, & m = n \end{cases} \quad (9)$$

The windowed IR weight factors are obtained by solving linear or non-linear equation systems, the choice of which depends on the signal character and a computing power [6]. However, in the simplest case these weight factors can be obtained by solving linear equations and provide the same accuracy of direct DHT as DHT based on FFT. At that the FFT usage allows to calculate the window in next way:

$$\begin{bmatrix} h_0^{(w)} \\ h_{-1}^{(w)} \\ \vdots \\ h_{-2N}^{(w)} \end{bmatrix} = \begin{bmatrix} x_{-N} & x_{-n+1} & x_{-N+2} & \cdots & x_N \\ x_{-N+1} & x_{-N+2} - x_{-N} & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_N & -x_{N-1} & \cdots & -x_{-N+1} & -x_{-N} \end{bmatrix}^{-1} \times \begin{bmatrix} \tilde{x}_{-N}\pi \\ \tilde{x}_{-N+1}\pi \\ \vdots \\ \tilde{x}_N\pi \end{bmatrix} \quad (10)$$

The result of (10) exists only when the inverse matrix exists too. Then the window coefficients are defined by following operation:

$$\{w_i\}_{i=-2N}^{2N} = \left\{ \begin{array}{l} h_i w_i \\ h_i \end{array} \right\}_{i=-2N}^{2N} \quad (11)$$

Example of this window for different lengths of IR is presented in Fig. 1:

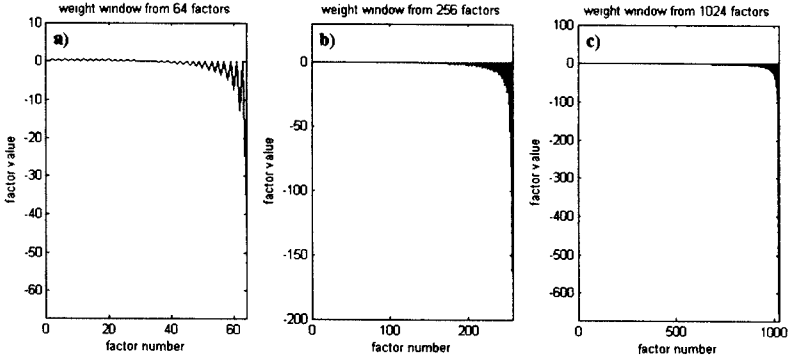


Fig. 1. Weight window for direct DHT improving: a) weight window from 64 factors; b) weight window from 256 factors; c) weight window from 1024 factors

Let's note the values of all odd weight factors are zeros and the value of the last weight factor is decreased radically with window length increasing.

5. THE WORKED OUT DHT PARALLEL ALGORITHM AND FILTER STRUCTURE

Direct DHT algorithm is derived from the convolution:

$$\tilde{x}_n = \frac{1}{\pi} \sum_{m=-N}^N x_m h_{n-m}^{(w)}, \quad (12)$$

where: $\forall m, n = \overline{-N, N}$ and $\forall(m-n) = \{a \mid a = \overline{-2N, 2N}\}$.

To develop a filter structure it is expedient to determine the convolution results for different n :

$$n = -N: \quad \tilde{x}_{-N} = \frac{1}{\pi} \left(x_{-N} \underbrace{h_{-N-(-N)}^{(w)}}_0 + x_{-N+1} \underbrace{h_{-N-(-N+1)}^{(w)}}_{-1} + \dots + x_0 \underbrace{h_{-N-0}^{(w)}}_{-N} + \dots + x_N \underbrace{h_{-N-N}^{(w)}}_{-2N} \right) \quad (13.1)$$

$$n = -N+1: \quad \tilde{x}_{-N+1} = \frac{1}{\pi} \left(x_{-N} \underbrace{h_{-N+1-(-N)}^{(w)}}_1 + x_{-N+1} \underbrace{h_{-N+1-(-N+1)}^{(w)}}_0 + x_{-N+2} \underbrace{h_{-N+1-(-N+2)}^{(w)}}_{-1} + \dots \right. \\ \left. \dots + x_0 \underbrace{h_{-N+1}^{(w)}}_0 + \dots + x_N \underbrace{h_{-2N+2}^{(w)}}_{-N} \right) \quad (13.2)$$

$$n = 0: \quad \tilde{x}_0 = \frac{1}{\pi} \left(x_{-N} \underbrace{h_{0+N}^{(w)}}_N + x_{-N+1} \underbrace{h_{0+N-1}^{(w)}}_{N-1} + \dots + x_0 \underbrace{h_0^{(w)}}_0 + \dots + x_N \underbrace{h_{0-N}^{(w)}}_{-N} \right) \quad (13.3)$$

$$n = N : \tilde{x}_N = \frac{1}{\pi} \left(x_{-N} \underbrace{h_{N-(-N)}^{(w)}}_{2N} + x_{-N+1} \underbrace{h_{N-(-N+1)}^{(w)}}_{2N-1} + \cdots + x_0 \underbrace{h_{N-0}^{(w)}}_N + \cdots + x_N \underbrace{h_{N-N}^{(w)}}_0 \right) \quad (13.4)$$

On the basis of these results the direct DHT algorithm can be write in a matrix form which is useful for the special processor implementation:

$$\tilde{x}_{-N} = X \frac{\quad}{-N, N} H \frac{(w)}{0, -2N}, \quad (14.1)$$

$$\text{where: } X \frac{\quad}{-N, N} = \begin{bmatrix} x_{-N} & x_{-N+1} & \cdots & x_0 & \cdots & x_N \end{bmatrix},$$

$$H \frac{(w)}{0, -2N} = \begin{bmatrix} h_0^{(w)} \\ h_{-1}^{(w)} \\ \vdots \\ h_{-2N+1}^{(w)} \\ h_{-2N}^{(w)} \end{bmatrix}$$

$$\tilde{x}_{-N+1} = X \frac{\quad}{-N, N} H \frac{(w)}{1, -2N+1}, \quad (14.2)$$

$$\text{where: } H \frac{(w)}{1, -2N+1} = \begin{bmatrix} h_1^{(w)} \\ h_0^{(w)} \\ \vdots \\ h_{-2N+2}^{(w)} \\ h_{-2N+1}^{(w)} \end{bmatrix}$$

• • •

$$\tilde{x}_0 = X \frac{\quad}{-N, N} H \frac{(w)}{N, -N}, \quad (14.3)$$

$$\text{where: } H \frac{(w)}{N, -N} = \begin{bmatrix} h_N^{(w)} \\ h_{N-1}^{(w)} \\ \vdots \\ h_{-N+1}^{(w)} \\ h_{-N}^{(w)} \end{bmatrix}$$

$$\tilde{x}_1 = X \frac{\quad}{-N, N} H \frac{(w)}{N+1, -N+1}, \quad (14.4)$$

$$\text{where: } H \frac{(w)}{N+1, -N+1} = \begin{bmatrix} h_{N+1}^{(w)} \\ h_N^{(w)} \\ \vdots \\ h_{-N+2}^{(w)} \\ h_{-N+1}^{(w)} \end{bmatrix}$$

• • •

$$\tilde{x}_N = X \frac{H(w)}{-N, N} H \frac{(w)}{2N, 0}, \tag{14.5}$$

where:
$$H \frac{(w)}{2N, 0} = \begin{bmatrix} h_{2N}^{(w)} \\ h_{2N-1}^{(w)} \\ \vdots \\ h_1^{(w)} \\ h_0^{(w)} \end{bmatrix}$$

We can represent a specialized parallel processor structure for the direct DHT on the basis of the equations (14.1) – (14.5):

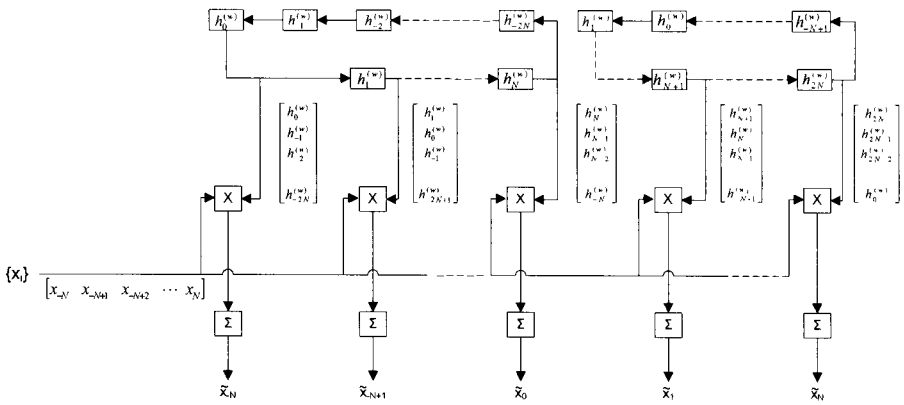


Fig. 2. The special processor parallel structure for direct DHT

This parallel structure contains two parts, both of which have the common AC converter and separate circulating registers for windowed IR shifting. Each from $2N+1$ channels consists of a multiplier and accumulator in which the calculations (14) are realized, for example in the first channel the calculation of \tilde{x}_{-N} is fulfilled according to (14.1). The first part of this structure provides DHT result for indices $i = \overline{-N, 0}$ and the second – for $i = \overline{1, N}$. This structure works correctly for both even and odd number of samples $\{x_i\}$.

The special processor parallel structure can be tested by an example for $\forall m, n = \overline{-2, 2}$ when $\forall (m - n) = \{a | a = \overline{-4, 4}\}$:

$$\tilde{x}_{-2} = \frac{1}{\pi} [x_{-2} + x_{-1} + x_0 + x_1 + x_2] \cdot \begin{bmatrix} h_0^{(w)} \\ h_{-1}^{(w)} \\ h_{-2}^{(w)} \\ h_{-3}^{(w)} \\ h_{-4}^{(w)} \end{bmatrix} = \frac{x_{-2}h_0^{(w)} + x_{-1}h_{-1}^{(w)} + x_0h_{-2}^{(w)} + x_1h_{-3}^{(w)} + x_2h_{-4}^{(w)}}{\pi} \tag{15.1}$$

$$\tilde{x}_{-1} = \frac{1}{\pi} [x_{-2} + x_{-1} + x_0 + x_1 + x_2] \cdot \begin{bmatrix} h_1^{(w)} \\ h_0^{(w)} \\ h_{-1}^{(w)} \\ h_{-2}^{(w)} \\ h_{-3}^{(w)} \end{bmatrix} = \frac{x_{-2}h_1^{(w)} + x_{-1}h_0^{(w)} + x_0h_{-1}^{(w)} + x_1h_{-2}^{(w)} + x_2h_{-3}^{(w)}}{\pi} \quad (15.2)$$

$$\tilde{x}_0 = \frac{1}{\pi} [x_{-2} + x_{-1} + x_0 + x_1 + x_2] \cdot \begin{bmatrix} h_2^{(w)} \\ h_1^{(w)} \\ h_0^{(w)} \\ h_{-1}^{(w)} \\ h_{-2}^{(w)} \end{bmatrix} = \frac{x_{-2}h_2^{(w)} + x_{-1}h_1^{(w)} + x_0h_0^{(w)} + x_1h_{-1}^{(w)} + x_2h_{-2}^{(w)}}{\pi} \quad (15.3)$$

$$\tilde{x}_1 = \frac{1}{\pi} [x_{-2} + x_{-1} + x_0 + x_1 + x_2] \cdot \begin{bmatrix} h_3^{(w)} \\ h_2^{(w)} \\ h_1^{(w)} \\ h_0^{(w)} \\ h_{-1}^{(w)} \end{bmatrix} = \frac{x_{-2}h_3^{(w)} + x_{-1}h_2^{(w)} + x_0h_1^{(w)} + x_1h_0^{(w)} + x_2h_{-1}^{(w)}}{\pi} \quad (15.4)$$

$$\tilde{x}_2 = \frac{1}{\pi} [x_{-2} + x_{-1} + x_0 + x_1 + x_2] \cdot \begin{bmatrix} h_4^{(w)} \\ h_3^{(w)} \\ h_2^{(w)} \\ h_1^{(w)} \\ h_0^{(w)} \end{bmatrix} = \frac{x_{-2}h_4^{(w)} + x_{-1}h_3^{(w)} + x_0h_2^{(w)} + x_1h_1^{(w)} + x_2h_0^{(w)}}{\pi} \quad (15.5)$$

The obtained results correspond to DHT results that confirms the operation correctness of specialized processor parallel structure for direct windowed DHT.

6. APPLICATION EXAMPLES OF THE WINDOWED DHT PARALLEL ALGORITHM FOR DETERMINING OF SIGNAL PHASE TRANSITIONS

The simulation of the specialized direct DHT processor usage was made using DHT to demodulate the binary phase shift keying (BPSK) modulated signals. In all examples below, we can note the DHT result in a shape of a pulse by a phase changes. In the Fig. 3a, 3b, 3c there are shown respectively: binary modulating signal, modulated carrier and DHT result of modulated carrier using parallel direct DHT algorithm. Likewise, in the Fig. 3f there is shown DHT result for multiple random phase changes with the added white Gaussian noise by SNR =10 dB.

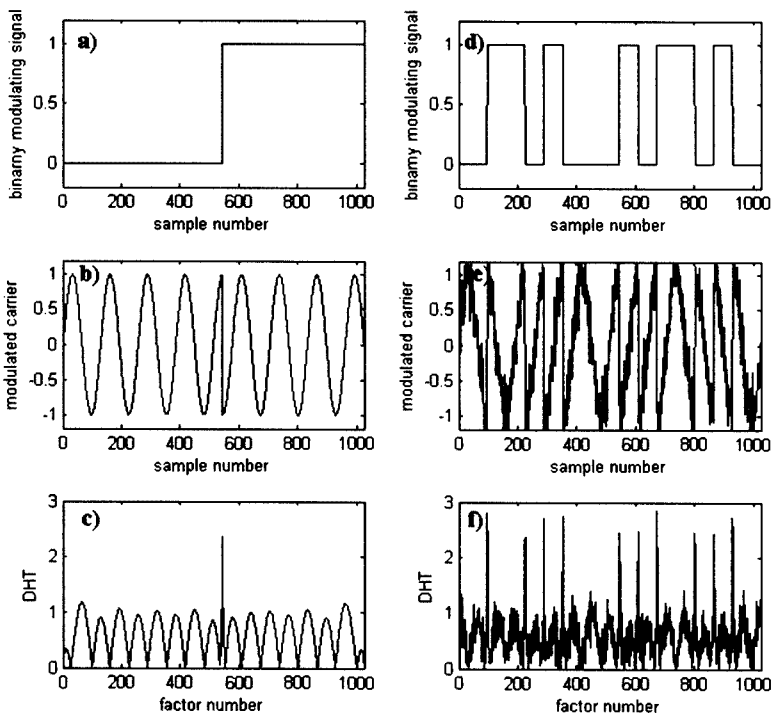


Fig. 3. The DHT result of a few periods BPSK modulated carrier without noise and with adding noise by SNR = 10 dB: a) binary modulating signal; b) modulated carrier without noise; c) DHT result of modulated carrier without noise; d) binary modulating signal; e) modulated noisy carrier; f) DHT result of modulated noisy carrier

In this work the following expression for the SNR was assumed:

$$SNR = 10 \lg \frac{\psi_x^2}{\varsigma_x^2} \quad (16)$$

where:

$$\psi_x^2 \quad \text{-- signal power} \quad \psi_x^2 = \frac{1}{N} \sum_{m=0}^{N-1} x_m^2, \quad$$

$$\varsigma_x^2 \quad \text{-- noise power} \quad \varsigma_x^2 = \frac{1}{N} \sum_{m=0}^{N-1} (x_m - \tilde{x}_m)^2.$$

The generated by computer white Gaussian noise was added to the modulated signal in Simulink Matlab using AWGN procedure:

$$\text{noised_signal} = \text{awgn}(\text{signal}, \text{SNR}, \text{'measured'}); \quad (17)$$

where:

- signal – modulated carrier
- SNR – planned SNR value

The modulated signal power was measured before adding noise to achieve planned SNR value. The DHT result of one modulated carrier period (Fig. 4a) is shown in Fig. 4b and the DHT result of the same signal noised by SNR = 0 dB (Fig. 4c) is shown in Fig. 4d.

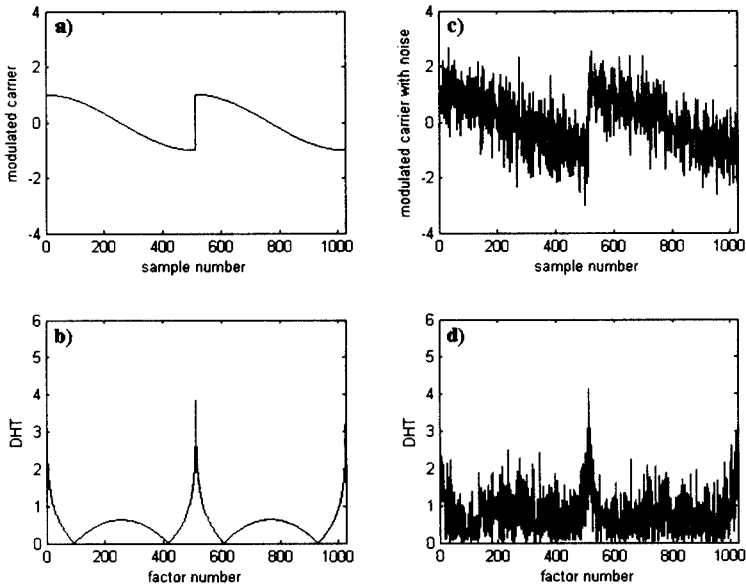


Fig. 4. The DHT result of one period BPSK modulated carrier without noise and with added noise by SNR = 0 dB: a) modulated carrier without noise; b) the DHT result of modulated carrier without noise; c) modulated carrier with noise; d) the DHT result of modulated carrier with noise

In all above shown cases there were used absolute values of DHT result.

7. SUMMARY

The worked out direct DHT algorithm with using the proposed weight windows can ensure equal or better accuracy to the algorithm based on FFT and leads to efficient hardware implementation. The parallel filter structure on the basis of the direct windowed DHT algorithm was proposed and checked on the example. This structure allows to get all convolution results in $2N+1$ processor facts that is $\log_2(2N+1)$ times faster than FFT method. It is also appropriate to be implemented in FPGA technology because of its uniform blocks. An example of a signal phase transition detection on the basis of the direct DHT was presented. The modulating binary signal can be successfully recognized in case up to SNR = -2 dB, therefore DHT leads to fast and reliable BPSK demodulation for example.

BIBLIOGRAPHY

- [1] Гоноровский И.С., 1971. Радиотехнические цепи и сигналы. Советское Радио, Москва.
- [2] Huang N.E. and others, 1998. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proc. Royal Society London., A, #454, London, 903-995.
- [3] Oczeretko E., Borowska M., Ludański T., 2006. Przekształcenie Hilberta w przetwarzaniu sygnałów i obrazów biomedycznych, XXXV KZM, Zakopane-Kościelisko, 12-19 września 2006, 60.
- [4] Oppenheim A.V., Schaffer R.W., 1998. Discrete-time signal processing, second edition. Prentice-Hall, Inc., Upper Saddle River NJ.
- [5] Pogribny W., Sulima M., 2010. Polepszenie dokładności bezpośredniego przetwarzania Hilberta, Zesz. Nauk. WSG w Bydgoszczy, Tom 10, Nr 3, 53-66.
- [6] Pogribny W., Sulima M., 2010. Wybór okien wagowych dla bezpośredniego DHT, Przegląd telekomunikacyjny, rocznik LXXXIII, Nr 8-9, 2010, 1479-1488.
- [7] Poularikas A.D., Ed., 2000. The transforms and applications handbook, second edition, CRC Press LLC, Boca Raton FL.
- [8] Szabatin J., 2003. Podstawy teorii sygnałów, WKŁ, Warszawa.

ZASTOSOWANIE RÓWNOLEGŁEGO ALGORYTMU BEZPOŚREDNIEGO OKIENKOWANEGO DHT

Streszczenie

Sprzętowe realizacje algorytmu dyskretnego przekształcenia Hilberta (DHT) w praktyce wykonywane są przy użyciu DHT opartego na FFT, ze względu na jego wyższą rozdzielczość w porównaniu z bezpośrednim DHT z nieokienkowaną charakterystyką impulsową. Z drugiej strony, jak pokazano pręcej w [5] [6], odpowiednie użycie okien wagowych dla charakterystyki impulsowej pozwala otrzymać wyższą rozdzielczość bezpośredniego DHT w porównaniu z metodą DHT opartą na FFT. W tej pracy zostały zaprezentowane algorytm równoległy oraz odpowiednia struktura filtru dla bezpośredniego okienkowanego DHT, które są szybsze w porównaniu z algorytmami i strukturami opartymi na FFT. Działanie zaprezentowanych algorytmu i filtru zostało zademonstrowane na przykładach przechwytywania zmian fazy sygnałów zaszumionych.

Słowa kluczowe: Dyskretne przekształcenie Hilberta, DHT, Rozpoznawanie zmian fazy

DEMAND FORECASTING FOR SPARE PARTS

Paweł Latos, Bożydar Dubalski, Tomasz Marciniak, Beata Marciniak

University of Technology and Life Sciences
7 Prof. Kaliskiego Ave., Bydgoszcz 85-796 Poland
dubalski@utp.edu.pl, tomasz.marciniak@utp.edu.pl, beata.marciniak@utp.edu.pl

Summary: The subject of the article is an analysis of possibilities of predicting the demand for subsystems for equipment for companies which deal with servicing electronic devices. The analysis has been based on real data from one of service points. The influence of various factors that determine the need for specific replacement parts was presented. Two basic methods of prediction (quantitative and shares) were compared. Modifications of the equity method was proposed, it took into consideration a few of additional factor, such as the number of parts in store or the number of overdue repairs.

Keywords: demand forecast, quantitative analysis, terms of share analysis

1. INTRODUCTION

Recently, on the market of applied electronics there have occurred significant changes which are manifested by a massive scale of the manufactured equipment and its fast development and modification and drop of both production and sale costs [2], [4]. These changes have been caused by progress in the sphere of design, production of components, modules and final devices, packages, accessories logistics and the distribution networks. After -sale service, that is, the Clients service is also an element of this chain [1]. There has been observed disappearance of small service shops where single systems of very different devices have been repaired. The process usually involved exchange of basic components and the repair was made until obtainment of the final effect.

Continuously increasing integration of integrated circuits, miniaturization of elements, improving manufacture of prints and relatively high cost of components, bought in small amounts, has made a repair of small series of devices less profitable. A new type of enterprise providing this kind of services has replaced small service stores. These are big – sometimes worldwide stores which provide an overall array of after-sale services. They cooperate directly with the equipment manufacturers. In this case, the amount of the repaired equipment can be considered in terms of thousands or even tens of thousands items, monthly [3].

For such a massive scale of repair the cost of purchase of components and logistics are very low as compared to the cost of a single service point [2]. It enables not only a decrease in guarantee protection costs which are born by manufacturers but also provides grounds for constructive competitiveness between the service providers.

Another advantage provided by such big companies is the possibility of doing research and introduction of innovation [3]. With such a mass scale of operation it is possible to sacrifice one or several items without any negative consequences. New, more efficient repair methods can be developed, troublesome defects can be diagnosed properly, time and material consumption can be reduced and, moreover, elimination of faulty elements and design error is possible. Thus, the quality of manufactured devices can be significantly improved and the investment costs can be quickly returned which in turn leads to generation of profits.

The costs of supply are affected by the way of cooperation between the equipment manufacturer and the customer service system:

- ‘**traditional**’ – it involves delivery of the faulty equipment by the customer directly to the service point, its repair and return to the customer.
- ‘**goodstock**’ – the service point sends the equipment to the manufacturer who exchanges it for new. The manufacturer comes back into possession of the broken device which is then sent to be repaired. This repair, apart from removal of functional defects, involves bringing the device to a state most similar to the pre-repair one – mainly by its face lifting (removal of traces of its being used before). Repaired in this way equipment is sent back to the manufacturer and goes to ‘goodstock’, that is, to a warehouse of efficient devices from where it is supplied to another customer after a complaint of his/her device is made by him/her.

The second way provides some benefits both measurable (from the point of view of the service the costs of shipment between the manufacturer and the service are minimized thanks to their big amounts), and immeasurable (short time of waiting for new equipment which results in the customer satisfaction with the service). The disadvantage of this system from the manufacturers’ point of view is burdening them with the logistics connected with the customer (reception of damaged equipment and dispatch of the repaired one), the customer’s service and possible losses due to the equipment advertisement which in fact, has no faults (because of e.g. misunderstanding the principles of its operation)

The subject of this paper is presentation of an analysis of supplying the service store with spare parts, and more exactly, a method for forecasting the demand for these elements, its implementation in real conditions and assessment of its application effects.

2. SIGNIFICANCE OF AN EFFICIENT METHOD FOR THE SERVICE STORE SUPPLY

Providing possibilities for making a big number of repairs in a relatively short time (generally accepted standard is 48 hours from the moment of the device registration in the service to dispatching it to the user) with achievement of appropriate financial effect is a pretty difficult challenge. Each of the process critical elements is responsible for keeping the expected index (Turn Around Time) that is the time in which the device is present in the service. When the term fails to be kept for a number of times it involves loss of trust and good relations with the equipment manufacturer. This is affected by the following factors;

- failing to deliver the repaired device to the user on time,

- higher costs of the user service,
- additional costs due to equipment exchange for new forced by a given situation or due to compensation payments in the moment of the user resignation from the repair.

To avoid such a situation it is necessary to provide a proper level of the service operation elements. These include:

- human resources,
- supply with operating materials (e.g. packages),
- supply with spare parts.

The last element, and more precisely, forecasting the demand for spare parts, is the subject the present analysis.

In practice, there often occur situations when there is shortage of many parts for the most popular models of devices in the moments when they are urgently needed. It happens so because of the lack of an efficient method for forecasting the demand for elements. This problem mainly occurs in case of new models of devices in the time when they are being increasingly used. On the other hand, for models which are getting out of common use, the spare parts accumulate to such an extent that it is not possible to utilize or sell them. Apart from the listed negative effects for the producers and relations with them, there occur problems disrupting operation of the service itself. Each device- regardless of the fact whether it has spare parts in stock or not, has to be verified in terms of its series number consistence, accessories and a diagnosis must be made. When it turns out that the needed element is missing this device must be additionally prepared for storage until the delivery of necessary elements. This takes time, space and means.

Solution to these problems is one of the main goals of persons responsible for the service point efficient operation and improving its effectiveness. These actions have been exemplified on the basis of repairs of digital cameras, electronic digital frames, picture printers and pocket video cameras, performed in a service store located in Bydgoszcz.

3. FORECASTING THE DEMAND FOR ELEMENTS NECESSARY FOR SERVICE

Manufacturer of devices which undergo a repair provides the service with information concerning a monthly supply of equipment and the spectrum of these devices supplied models. It seems that such data is out of use for forecasting the demand for a particular spare part as they it does not provide useful information directly, apart from the information on human resources and the company profits. However, after a thorough analysis of the data and its calculation, it can provide basis for prediction of all actions to be taken as well as a reliable point of reference.

The issue of forecasting supplies in a traditional service model is of complex character and requires an analysis of many factors which have an impact on the number of ordered supplies. One of the most important elements of this process is the model lifetime.

For each device model the most important parameter is the time of its existence on the market (in sale) which is reflected by its presence in supplies of equipment to be

repaired. When the product is launched on the market it means that it will appear more and more frequently in the supplies. Therefore, in this time the most important task is to provide the warehouses with subsystems that can meet current demand and also will create a certain safety buffer which can be used when an increase in the number of supplies has significantly exceeded earlier predictions.

The middle of the discussed period (stagnation) is a time when the rising tendency starts decreasing and then it stabilizes, though it does not mean invariability of this value within the next few successive months. Surprisingly, prognosis prepared during this time are burdened with the highest risk. It is so due to unpredictability of the stagnation period end, after which in a short time there can occur very drastic falls of the demand for these elements.

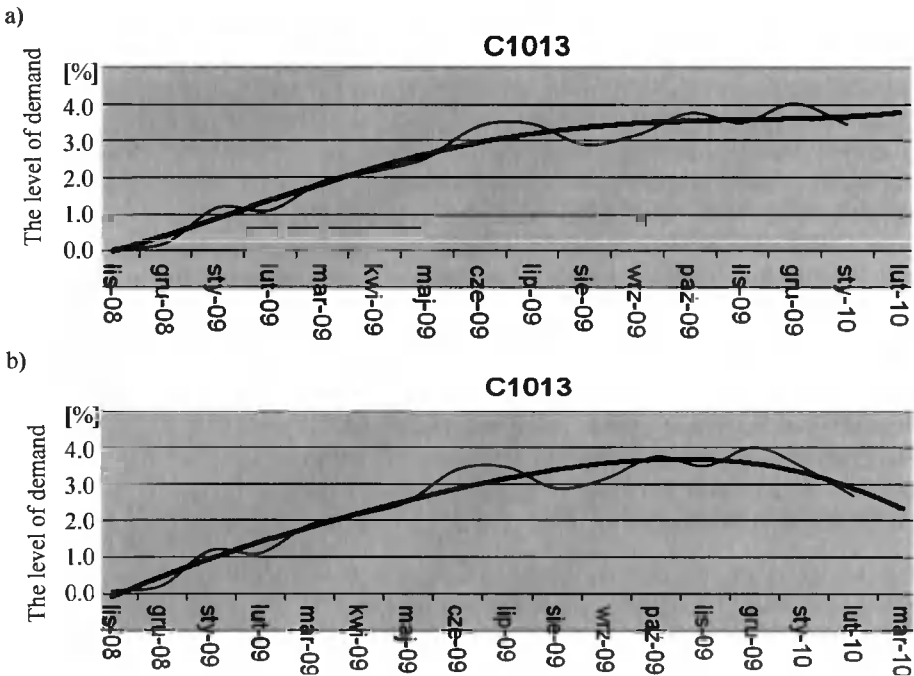


Fig. 1. Observation of the stagnation period end during the model lifetime (a) stagnation period, (b) real demand

The above figure (Fig. 1) illustrate a similar situation. For C1013 model, the stagnation period lasted until October 2009. This tendency continued for the next four months and there was no sign which would predict changes. Therefore, in February 2010 it was most likely that the demand for spare parts would stay at the same level 3.8%. However, it was in February when this tendency changed and the level decreased to approximately 2.7%. This is the type of risk that can occur in the time of stagnation.

During the end of the model lifetime the numbers of supplies decrease systematically. And the same thing should happen to the spare parts which are stored in warehouses. In this period all stock should be used with only a minimal amount of elements left. Time of this cycle depends on the segment which the model comes from, time of sale and in the final period when the supplies are rare, it should depend on

profitability of guaranty repairs. The factor must be taken into consideration for better understanding of the forecasting system and its implementation.

Another important aspect that must be accounted for is the assessment method of the demand for spare parts. Here, a quantitative or share analysis can be applied. One of the first problems that occur while using a quantitative analysis for elaboration of a prognosis method is variability of the number of supplied devices. Depending on the season, the number of repairs varies significantly, according to the data available, the differences reach even several dozen percent. Thus, it is difficult to use it for predicting the demand for necessary elements. This would be possible only if the total number of supplies became constant for each month of the year.

A factor that enables observation of the tendency regardless of different quantities of supplies is the share approach. It involves determination of a given model supply quantity quotient coefficient called 'a share', in relation to the total supply quantity of all models of devices in a given month.

This has been illustrated in the below presented example concerning supplies in the period between 2009 and February 2010.

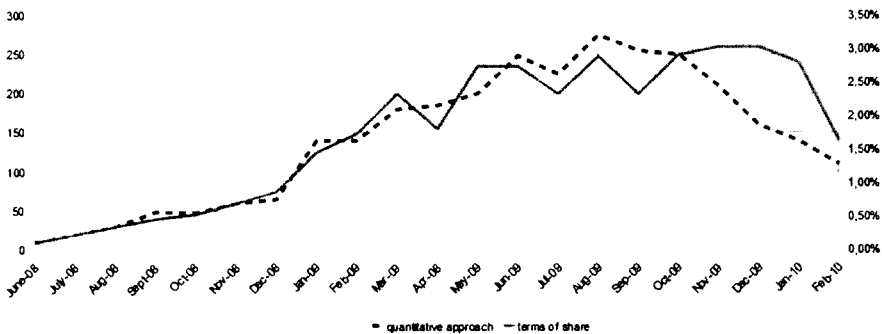


Fig. 2. Comparison of variability of quantitative approach with the share one

On the basis of observation of quantitative changes it can be said that there starts the final time for a given product. In the meantime, a chart of shares shows something else, stagnation reaches the highest values. Fluctuations range to 0.5%. When the supplies increase in the spring, share based forecasting will anticipate it. The result of an analysis with the use of the quantitative approach would be expected, further, significant falls which would result in the lack of spare parts for most of devices of a given model.

The forecasting method of a demand for elements which is still quite widely applied is based on a simple calculation concerning only spare parts without accounting for any other factors. Data on the wear of parts from the three preceding months is being collected, and the mean is being calculated. This mean provides basis for making an order for a manufacturer.

For comparison of both forecasting methods an element called 'COVER AY' – battery plate new design' of a C180 model present in repairs for the last three months (usually an intensive increase in the quantity of supplies starts then) has been used as an example.

Table 1. Comparative analysis of forecasting methods

Month	09 06	09 07	09 08	09 09	09 10	09 11	09 12	10 01	10 02
Quantity of supplies	7	24	34	69	60	78	49	95	117
Share	0.07%	0.24%	0.33%	0.59%	0.68%	1.07%	0.87%	1.76%	2.46%
Monthly forecasting of supplies	6438	6783	6628	5960	5762	5786	5660	4753	5874
Demand forecasting in terms of quantity	x	x	x	8	15	19	25	22	26
Demand forecasting in terms of share	x	x	x	8	17	18	28	33	61
Real use	2	9	12	25	21	28	17	34	42

If for the first three months of forecasting the differences between both methods are slight, they become clear once the model enters the period of intensive development.

Table 2. Comparison of the forecasting systems accuracy

	Summary 2009 09 – 2009 11		Summary 2009 12 – 2010 02		Summary 2009 09 – 2010 02	
	Quantity	Error	Quantity	Accuracy	Quantity	Accuracy
Demand forecasting quantitatively	42	- 41.7%	73	- 21.5%	115	- 31.1%
Demand forecasting in terms of share	43	- 40.3%	122	+ 31.2%	164	- 1.8%
Real use	74		93		167	

Data covering the analyzed forecasting time gives a picture of differences in operation of both methods. In the stage of an intensive increase the share forecasting is of great advantage, that is, progression of predictability. In this way it is possible to compensate inaccuracy of prognosis which occurred during the trend formation. The period from September to November was closed with a similar, quite low accuracy. In the next three month the period a change can already be seen. The share forecasting exceeds a current demand by 30%, compensating, however, the previous prognosis which in effect provides accuracy similar to the equipment real needs. Quantitative forecasting does not, due to its nature, provide any significant correction.

In the chart Fig. 3, a comparison of prognosis with verifying them real data, has been demonstrated.

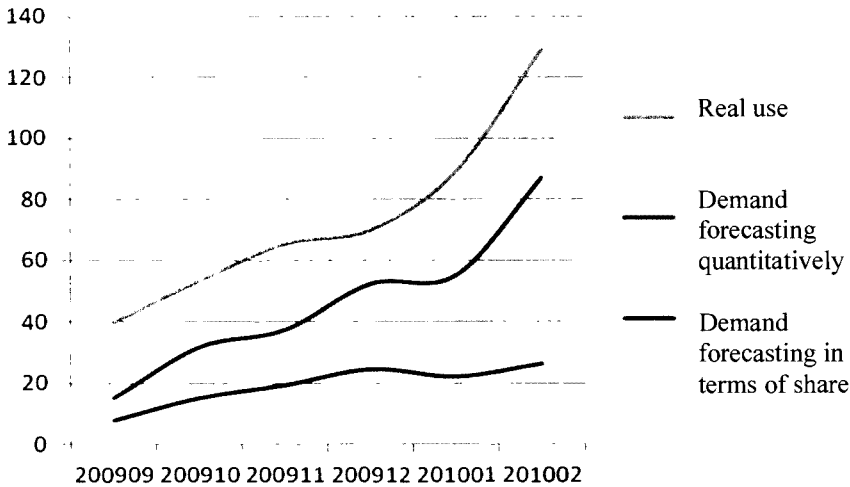


Fig. 3. Comparison of prognosis with real data

Thus, a conclusion can be formulated that the share forecasting has the ability of immediate reaction to the trend changes.

An important element for the forecasting system is the use of the trend line (it is a monotonic component in the model of dependence of the component statistic feature on the time of a given phenomenon observation).

Having accepted the assumption that data on the quantity of a particular model supply and their share are known, it is possible, on their basis, to make a prognosis using its graphic image. From the point of view of statistics, in order to make any prediction, it is necessary to have at least three measurement points (e.g. observation summing up three months). Having this data one can take up the most important action. On the basis of a current data analysis, the line of trend is determined, that is, an approximating line which is a kind of averaging whose aim is to define the general trend for the available set of data.

Basing on the determined trend line in the scope of the determined data it is necessary to increase its continuity by another period.

In this way current tendency is used for prediction of the studied value.

The line trend (in black) shown in the above figure (Fig. 4), built on the basis of real values (in blue) was used for definition of model C180 share in supplies in March 2010. There is the question of choice of the kind of the line. It is possible to choose from several kinds of lines (linear, multinomial, exponential). In the initial phase of the system creation it was found that the best appropriate in terms of accuracy is the line of multinomial approximation one:

The choice of the multinomial degree mainly depends on such factors as:

- The model life phase. The longer it is repaired (the more measurement data) the higher degree of the multinomial ensures better approximation.
- Intensity and fluctuations of the share. Drastic changes in this range can even cause a necessity of using high degrees in the early phase of life.
- Expected changes of the trend. Observing oscillations, it is possible to infer what changes will be likely to happen in the nearest time (e.g. switching a rising trend to a lateral or decreasing one, where the rising trend means an increase in a variable in

time value and decreasing trend means a fall of a variable in time value. Lateral trend stands for lack of distinct rising or falling trend. Each degree of the multinominal is characterized by a different prognosis progression. This is a basis on which the degree adjusted to current needs- should be chosen-progressive in the phase of rising, preservative at the end of the model life.

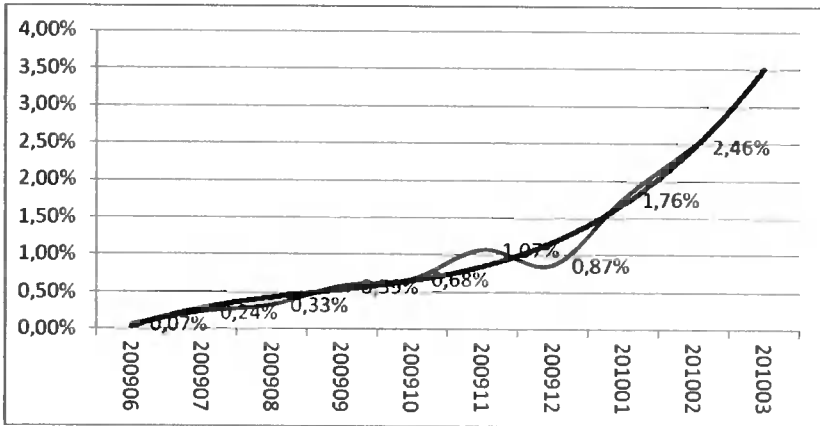


Fig. 4. Visualization of the trend line

4. CONSTRUCTION OF FORECASTING MODELS AND ASSOCIATION OF DATA CONNECTED WITH SUPPLIES OF EQUIPMENT MODELS WITH DATA ON DEMAND FOR SPARE PARTS

Being in the possession of elaborated data on the shares it is possible to go on to prepare a prognosis of the models supplies. Knowing the overall monthly prognosis of supplies and a share of a given model in global supplies one must multiply these two factors in order to receive an absolute number of supplies planned for the next month.

Table 3. Method of making a prognosis of a device model supply

Model	Trend	Supply prognosis	Planned number of devices
C1013	0.0240	5975	143
C140K	0.0410	5975	245
C180K	0.0340	5975	203
C182	0.0120	5975	72
C190	0.0260	5975	155
C300	0.0000	5975	0
C310	0.0000	5975	0
C330K	0.0009	5975	5

With a mass character of being made repairs, forecasting of the number of spare parts indispensable for the store operation is subject to laws of statistics. Each model has faults characteristic for itself which involve replacement of precisely defined parts. Due to recurrence, it is possible, here, to make use of the share approach, but it will refer only to selected parts. This index is calculated in a different way. In this case, a ratio of the number of used parts to the number of repairs performed in the studied period, is taken into consideration. Hypothetically, this time could be relatively short e.g. three months. However, it should be considered that despite of forecasting accuracy, there can occur situations when a part is missing for a longer time, e.g. because of its manufacturer's fault, as they did not provide sufficient supplies at the moment the device was launched on the market. This factor can distort prognosis for a few following months. For this reason, a longer time e.g. 12 months, should be accounted for. This time is sufficient for compensation of some periods of a part unavailability as it enables, to some extent, flexibility of the parts wear division and minimization of the seasonal impact on the demand for spare parts. The mean life time for devices which were used for this analysis is from one and a half to three years. Thus, if during this time there will occur changes within the spectrum of appearing defects it on an every time basis will be possible to be defined to a limited degree.

Table 4. Comparison of shares of C1013 model parts

Model	Part number	Part name	Completed repairs	Amount used parts	The share parts
C1013	2F7112	MONITOR LCD – AUO VJ	2828	111	3.93%
C1013	2F7480	MONITOR LCD – Wintek	2828	18	0.64%
C1013	3F6204	Fuse – 1.25A	2828	1	0.04%
C1013	4F8467	BOARD – main, for Wintek LCD	2828	18	0.64%
C1013	4F8468	BOARD – main, for GPT LCD	2828	54	1.91%
C1013	4F8569	HOLDER – monitor LCD	2828	3	0.11%
C1013	4F8571	BOARD AND FRAME AY – switch	2828	20	0.71%
C1013	4F8572	BOARD AND STROBE AY – power	2828	200	7.07%
C1013	4F8580	COVER AY – front, silver, C1013	2828	95	3.36%
C1013	4F8581	COVER AY – back, silver, C1013	2828	144	5.09%
C1013	4F8582	DOOR AY – battery, silver, C1013	2828	22	0.78%
C1013	4F8583	LENS AY – with CCD	2828	351	12.41%
C1013	4F8584	BOARD – main, for AUO LCD	2828	318	11.24%
C1013	4F8594	MONITOR LCD – GPT 2 LED	2828	15	0.53%
C1013	4F8651	LENS AY – without CCD	2828	167	5.91%

The above presented Table 4 shows exemplary calculations of the part share. Having this data it is enough to multiply it by the planned numbers of the model supplies.

Table 5. Visualization of the demand for spare parts

Model	Part number	Share	*	Estimated number of parts	=	Part forecasting
C1013	2F7112	0.03930	*	143	=	6
C1013	2F7480	0.00640	*	143	=	1
C1013	3F6204	0.00040	*	143	=	0
C1013	4F8467	0.00640	*	143	=	1
C1013	4F8468	0.01910	*	143	=	3
C1013	4F8569	0.00110	*	143	=	0
C1013	4F8571	0.00710	*	143	=	1
C1013	4F8572	0.07070	*	143	=	10
C1013	4F8580	0.03360	*	143	=	5
C1013	4F8581	0.05090	*	143	=	7
C1013	4F8582	0.00780	*	143	=	1
C1013	4F8583	0.12410	*	143	=	18
C1013	4F8584	0.11240	*	143	=	16
C1013	4F8594	0.00530	*	143	=	1
C1013	4F8651	0.05910	*	143	=	8

This provides a rational number of particular parts which will be necessary to ensure the model repairs in the next month.

Determination of the number of necessary parts is not the final stage of the order creation. These factors also have to be accounted for:

- warehouse states, that is, the number of parts which are already in stock. They must be subtracted from the so far done calculations,
- demand for devices whose repair has been stopped until delivery of spare parts, that is, devices for whose repair necessary spare parts were missing already before preparation of an order. These values must be added to the calculations.

The discussed analysis has provided basis for elaboration of a program to be used by companies for forecasting supplies to be implemented in the company network server. Many protections have been created as well as they provide security for data storage, so that an unauthorized person will not have access to it.

The main source of all kinds of data used in the system operation is an administered by the company system of database applications, being the basis of reporting all actions happening every day. These include: the device registration, registration of the part, their location in the warehouse, the repair, wear of components, performance of different tests, protection of accessories, gathering data on the performed deliveries, inventory invoicing and financial matters.

The program provides the following possibilities of revision of the devices and their components in the quantitative and share approach:

- review according to months – displays a list of all devices registered in the assigned period,
- review according to models – displays a list of all items of the device given model whose name has been entered into the field of the form,
- comprehensive review – displays a list of all registered devices without any limiting criteria.

The planned total number of supplies In term of particular months of the current year. Warehouse states are reviewed:

- according to the models, which displays a list with all data on a given model part share,
- according to the part number, which displays information on the share of a particular part,
- according to the part type, which makes it possible to look over a Table in terms of module types, which is expressed by search of the attribute,
- comprehensive review, displays a list of all data on all parts share with no limiting criteria.

5. CONCLUSIONS

The idea of the proposed method for forecasting a demand for spare parts has been tested in comparison with the already existing prediction method.

According to expectations, the system improved the quality of the service supply during a few months to such a degree that repair arrears due to insufficient number of orders were made up. There also followed another significant change. Earlier the orders had been sent twice a month, since then once a week. To justify this, was presented a comparison of real data on forecasting supplies of models and their actual quantities that were provided for repair in March 2010. Seven models of cameras were taken into consideration, whose supplies were the biggest, being thereby the most significant part of this segment.

Table 6. Comparison of supply prognosis of models with actual numbers of supplies in March 2010

Model	C140	C180	C182	C190	C813	C913	C1013
Forecast quantities	149	108	48	84	108	191	90
Number of actual supplies	169	133	48	94	83	177	125
Accuracy of prognosis [%]	88.39	80.86	99.58	88.99	129.58	108.02	71.70

This result proves that the system makes it possible to achieve fairly good accuracy and efficiency which are dependent on appropriate prediction of demand for materials necessary for regular operation.

The above listed improvements have contributed to the corporate image better reputation both in the eyes of the manufacturers and customers. It also increased its reliability as well as and the quality of repairs and trust of cooperating companies. Having rationalized the process of supply made it possible to focus attention and means on making further improvements upon other elements of the offered services.

BIBLIOGRAPHY

- [1] Cieślak M., 2000. Economic Forecasting: methods and applications (in Polish), PWN, Warszawa.

- [2] Jiafu R., Zongfang Z., Fang Z., 2009. The Forecasting Models for Spare Parts Based on ARMA, World Congress on Computer Science and Information Engineering, pp. 499-503.
- [3] Krupa K., 2008. Modeling, simulation and forecasting: continuous systems (in Polish), WNT, Warszawa.
- [4] Yun Ch., Heng Z., Li Y., 2010. Demand Forecasting in Automotive Aftermarket Based on ARMA Model, International Conference on Management and Service Science (MASS), pp. 1-4.

PROGNOZOWANIE ZAPOTRZEBOWANIA NA CZĘŚCI ZAMIENNE

Streszczenie

W artykule zaprezentowana analizę porównawczą różnych metod prognozowania zapotrzebowania na części zamienne w punktach serwisowych. Przedstawiona analiza została praktycznie zweryfikowana w oparciu o rzeczywiste dane pochodzące z jednego z punktów serwisowych. W pracy przedstawiono wpływ różnego typu czynników od których zależy zapotrzebowanie na określone części zamienne. Porównano dwie podstawowe metody predykcji: ilościową i udziałową. Zaproponowano modyfikacje w metodzie udziałowej uwzględniające dodatkowe czynniki jak np. ilości poszczególnych części znajdujące się w magazynach czy konieczność dokonania zaległych napraw.

Słowa kluczowe: prognozowanie zapotrzebowania, metoda ilościowa, metoda udziałowa

LOW VOLTAGE, HIGH-SPEED FOUR-QUADRANT CMOS TRANSCONDUCTANCE MULTIPLIER

Jacek Jasielski¹, Stanisław Kuta¹, Witold Machowski¹, Wojciech Kołodziejski²

¹ Department of Electronics
AGH University of Technology
Kraków, Poland

² Division of Electronics and Telecommunication
State Higher Vocational School in Tarnów
Tarnów, Poland

Summary: The paper presents an analog four-quadrant transconductance multiplier designed in CMOS technology, suitable for low voltage and operating at high-speed. The transconductance multiplier with Gilbert-like architecture uses a cascade of a combination of two linear current dividers implemented by means of the differential pairs to produce a linear dependence between the tail current and the two output currents. To adopt the circuit for low voltage, simple current mirrors have been applied to couple the first- and the second stage of the current dividers cascade. High-speed operation is possible thanks to simple architecture of building blocks using RF CMOS transistors with sufficiently large biasing currents. A complete circuits schematic with input driving peripherals, as well as simulation results of entire multiplier have also been presented.

Keywords: analog VLSI, four quadrant multiplier, CMOS

1. INTRODUCTION

The analog multipliers are versatile building blocks, desired in many applications including continuous time signal processing, automatic variable gain amplifiers, neural networks, and as mixers and modulators in communication systems. They are widely used in contemporary VLSI chips for modulation/demodulation, other non-linear operations including division, square rooting as well as frequency conversion.

The most popular implementation of a four-quadrant multiplier, proposed in the bipolar technology in late 1960's [3], is the famous *Gilbert cell*, which is still the backbone of many different improvement architectures of the contemporary multipliers. The Gilbert cell can be considered as a combination of linear current dividers, implemented by means of a two stage cascode of differential pairs. Each of the differential pairs produces a linear dependence between its tail current and its two output currents. The output currents difference $i_{C1} - i_{C2}$ for bipolar long tail pair is given by:

$$i_{C1} - i_{C2} = I \cdot \operatorname{tgh}(v_X/\varphi_T) \approx I \cdot v_X/\varphi_T \quad (1)$$

where φ_T is the thermal voltage, I is the tail current of the differential pair; for small input differential voltage tgh function may be approximated by its argument itself.

For the bipolar differential pair, linearity with respect to I is guaranteed by the linear dependence g_m versus I (g_m is the transistors' transconductance), which holds with accuracy over an I range of several decades. Linearity with respect to the v_y port, for the "upper" stage of the Gilbert cell, is obtained by proper pre-distortion of the input voltage.

In contrary to bipolar technology, long channel MOSFETs based Gilbert cell implementations have another features. Using the square law relationships (valid for long channel MOSFETs in strong inversion), the currents difference $i_{D2} - i_{D1}$ is given by:

$$i_{D2} - i_{D1} = \frac{I}{2} \sqrt{\frac{\beta \cdot v_X^2}{I} - \frac{\beta^2 \cdot v_X^4}{I^2}} \approx v_X \sqrt{\beta \cdot I} \quad (2)$$

The approximation used in (2) is also valid only for small differential input voltage. Linearity with respect to I is not guaranteed, because opposite to bipolar devices, a long channel MOSFET's transconductance depends on the square root of the bias current:

$$g_m = \sqrt{2\mu_n C_{ox} \frac{W \cdot I_D}{L}} \quad (3)$$

and this relationship points that input signal of the long channel MOSFET counterparts of Gilbert cell is much more limited and the output current swing may be only very small fraction of the bias value. Of course it is possible to use the body driven, or weak inversion MOSFETs structures [7] to imitate bipolar devices, but in this case we obtain significantly worse frequency response, and very small usable output current range.

Contrary to a long channel MOSFET, for the short channel one, the $V_G - I_D$ relationship becomes rather linear than square-type, especially for higher drain current. If the difference $V_G - V_T/L$ becomes significantly larger than E_{sat} (being the field at which carrier velocity drops to half the value extrapolated from low-field mobility, the drain current approaches the value of:

$$I_D = \frac{\mu_n C_{ox}}{2} W (V_{GS} - V_T) E_{sat} \quad (4)$$

and thus short channel MOSFET transconductance is proportional to the drain current:

$$g_m = \frac{\mu_n C_{ox}}{2} W E_{sat} = \frac{I_D}{(V_{GS} - V_T)} = \frac{I_D}{V_{Dsat}} \quad (5)$$

and consequently differential pair comprising such a MOSFETs may have properties similar to that of its bipolar counterpart.

Anyway it should be noted, that CMOS implementation of Gilbert cell architecture – double cascode of differential pairs combination comprise a stock of at least four transistors in the saturation region and therefore its supply requirement usually touches and sometimes even exceeds allowed value for most of contemporary CMOS processes.

In the paper we present four-quadrant transconductance multiplier in Gilbert-like architecture using a two stage cascade of a combination of the differential pairs with short channel MOSFETs. The complete implementation including driving circuits, suitable for low voltage and high speed operation have been also presented.

2. PRINCIPLE OF OPERATION

Fig. 1 shows a block diagram of a four-quadrant transconductance multiplier in Gilbert-like architecture, implemented as a two stage cascade of a small signal transconductors combination. This block diagram follows the idea presented already in [4] more recently used also in [2]. In our solution however, current mirrors have been applied to transfer two output currents of the first transconductor to next stage ones, as tail currents of two differential pairs. The output currents of the first stage fully differential transconductor, I_1 and I_2 in Fig. 1, are related to a common mode current I_{SS} and a differential component $G_m v_Y$, according to:

$$I_1 = \frac{I_{SS}}{2} + \frac{1}{2} G_m v_Y ; I_2 = \frac{I_{SS}}{2} - \frac{1}{2} G_m v_Y . \quad (6)$$

These two outputs currents, coupled by simple current mirrors, create the tail currents for the two small signal transconductors in the next stage of the current divider, which are controlled by v_X input voltage and have transconductances of g_m . From (5) and (6), we easily get the output differential current of the multiplier:

$$\begin{aligned} i_{OUT} &= i_{OUT1} - i_{OUT2} = (I_3 + I_6) - (I_4 + I_5) = \\ &= (I_3 - I_4) - (I_5 - I_6) = \frac{G_m v_Y v_X}{V_{Dsat}} \end{aligned} \quad (7)$$

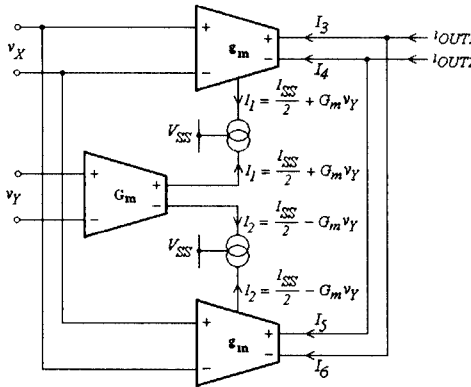


Fig. 1. Block diagram of a four-quadrant transconductance multiplier in Gilbert-like architecture

The block diagram of Fig. 1 has been implemented with transcoductors based on differential pairs with CMOS transistors biased with sufficiently large currents. The schematic of the proposed multiplier circuit is shown in Fig. 2. Because the proposed solution of the multiplier architecture is dedicated rather for high frequency applications, therefore the driving of all the differential pairs should be symmetrical. To achieve this, the additional inverting voltage amplifiers for v_X and v_Y signals have been applied.

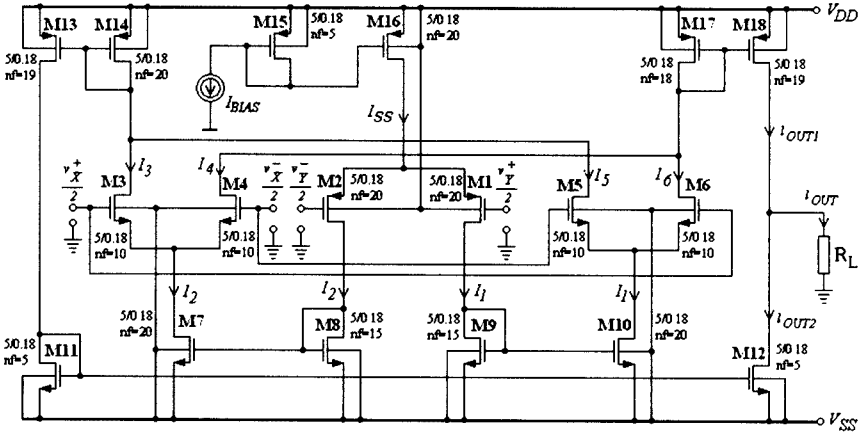


Fig. 2. The schematic of the proposed multiplier circuit

They have the form of simple CMOS inverter (M1, M2) with complementary diode connected transistors load (M3, M4) (Fig. 3). From small signal perspective this circuitry is a transconductor loaded with the same transconductor with shorted input and output ports. Thus resulting voltage gain is -1 .

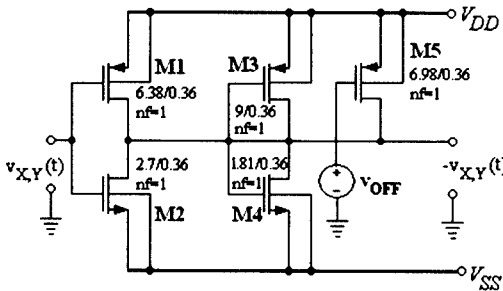


Fig. 3. The schematic of the auxiliary inverting voltage amplifier

It is known that n-MOS conventional long tail pair has better performance (in terms of input swing and linearity) if common mode of the input signal is shifted closer to V_{DD} , especially for lower supply voltages. Therefore M5 transistor has been added (Fig. 3) to perform an appropriate signal conditioning. On the other hand the possibility of aforementioned level shifting may be used for classical offset compensation. The output offset is a consequence of an unmatched threshold voltages of p-MOS and n-MOS in the inverter.

3. SIMULATIONS RESULTS

Circuit-level design and simulation, circuit layout, post-layout verification for UMC's 180 nm CMOS process has been done on the base Foundry Design Kit in the Cadence DF II environment, available via EURORACTICE.

Layout of the completely CMOS transconductance multiplier circuit, containing also the input inverting amplifiers, and utilizing the 18RF_NMOS and 18RF_PMOS cells of the UMC's 180 nm CMOS FDK, is shown in Fig. 4. Circuit has been simulated assuming symmetrical $+0.9/-0.9$ Volt supply (both n- and p-MOSFETs' threshold voltage is about 0.5 Volt). All the presented simulations have been done with postlayout extracted netlist and without I/O cells.

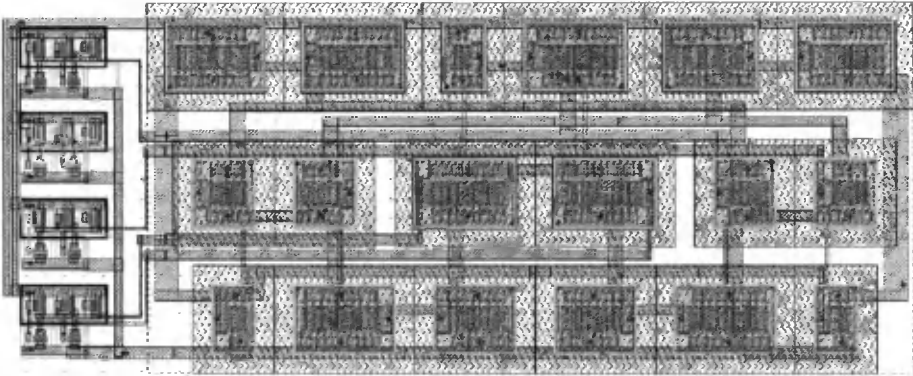


Fig. 4. The entire circuit laid out in UMC 180 nm CMOS technology

Figures 5 and 6 present the DC and small-signal characteristics of the inverting amplifier, respectively. The implementation of such auxiliary yet important building block is no so obvious, so quite satisfactory simulated performance are vital for the rest of the project. From Fig. 5 it can be seen that DC response is linear in the range almost twice wider than required to control the inverting input of the differential pair, while the small signal analysis results indicate relatively wideband response – the 3-dB corner frequency slightly exceeds 3GHz while THD is less than 0.5% at 100 MHz for 200 mV_{p-p} input signal.

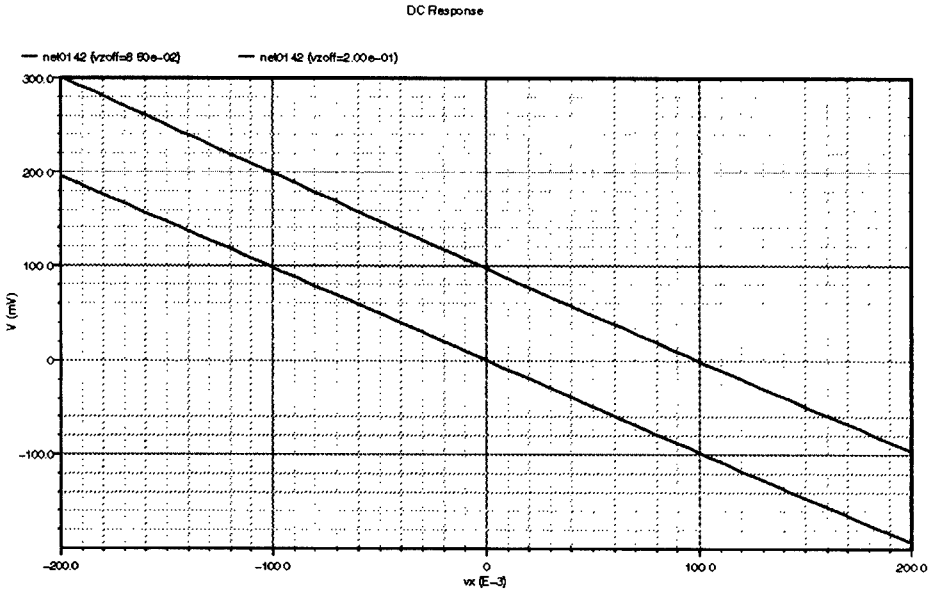


Fig. 5. Simulated DC characteristics of auxiliary voltage inverter

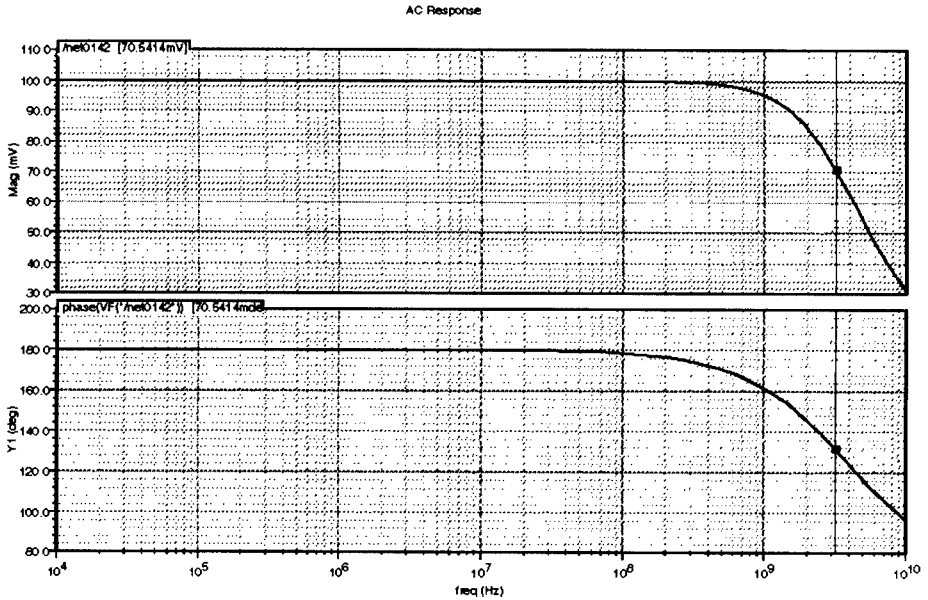


Fig. 6. Small signal characteristics: magnitude and phase of auxiliary voltage inverter

Fig. 7 shows the DC transfer characteristics families for the complete multiplier circuit: $i_{OUT} = f(v_X)$ while $v_Y = \text{const}$ and Fig. 8 $i_{OUT} = f(v_Y)$ while $v_X = \text{const}$, respectively. In both cases, one input voltage was swept from -100 mV to 100 mV , while the other one was stepped from -100 mV to 100 mV by 50 mV .

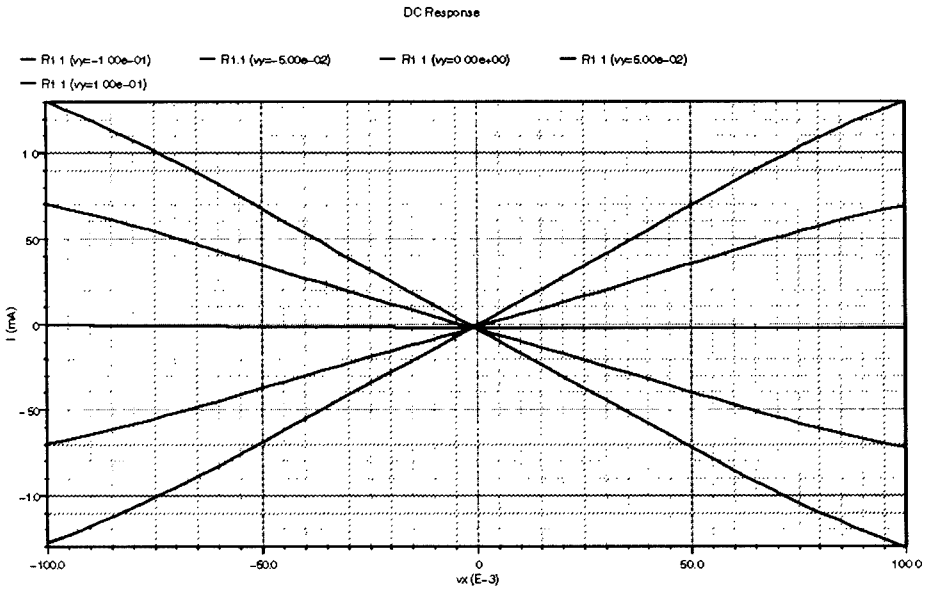


Fig. 7. Simulated DC transfer characteristics families of the complete multiplier circuit:
 $i_{OUT} = f(v_X) @ v_Y = \text{const}$

Comparing this two families of curves, shown in Fig. 7 and 8, we see, that transistor's transconductances of the differential pair for input v_Y signal are proportional to their drain currents and guarantee a linear dependence between the differential pair tail current and its two output currents. To meet this condition, the tail current of the differential pair for input v_Y signal is equal 1.4 mA and RF transistors with minimum channel length $L = 0.18 \mu\text{m}$ have been applied.

The transfer characteristics of the multiplier are linear in the range from -100 mV to 100 mV (the full differential pair input signal is 200 mV). For $|v_X|, |v_Y| < 100 \text{ mV}$, the integral linearity error of the multiplier is less than 0.5%. This value increases to about 1.5% for $|v_X|$ and $|v_Y|$ equal to 150 mV ; which covers about 16% of the supply voltage.

Total harmonic distortion with $100 \text{ mV}_{\text{p-p}}$ input signal at either input terminal with $\pm 50 \text{ mV}$ DC-voltage at the other is less than 0.25% up to 100 MHz and increases to 1% at 500 MHz .

In Fig. 9 AC magnitude frequency response of the multiplier are depicted. From simulation in this domain comes that multiplier 3 dB bandwidth reaches almost 1 GHz .

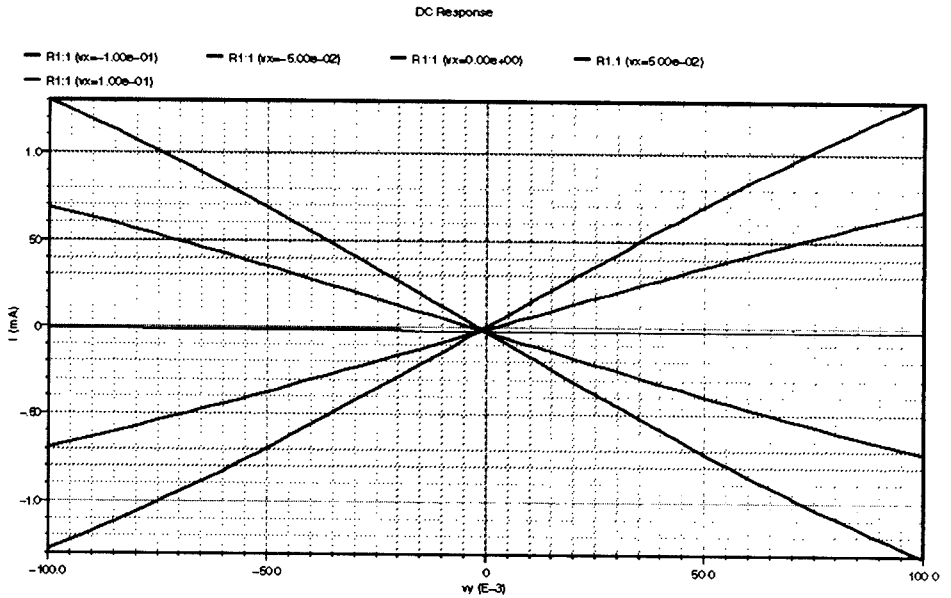


Fig. 8. Simulated DC transfer characteristics families of the complete multiplier circuit:
 $i_{OUT} = f(v_Y) @ v_X = \text{const}$

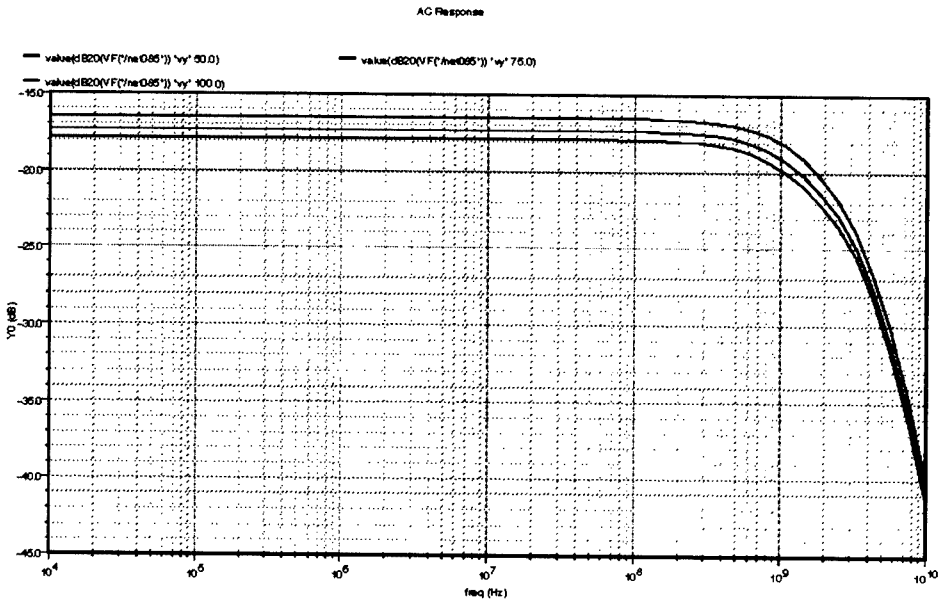


Fig. 9. Simulated AC response of the whole multiplier

Fig. 10 shows transient simulation results for different frequency sinewaves attached to both inputs – and we observe modulation of 100 MHz carrier by 2 MHz signal.

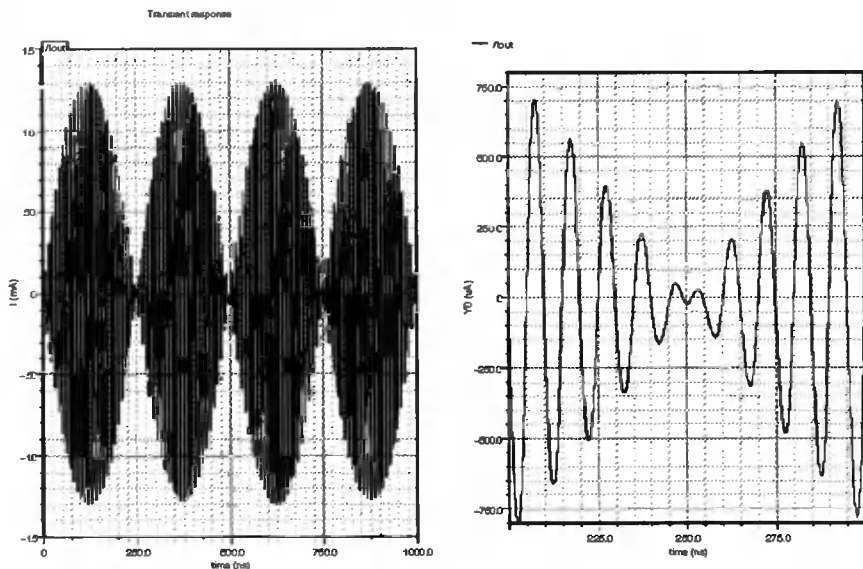


Fig. 10. Transient response for different input frequency AM modulation

Finally Fig. 11 illustrates transients for the multiplier working in phase-and coincidence detector.

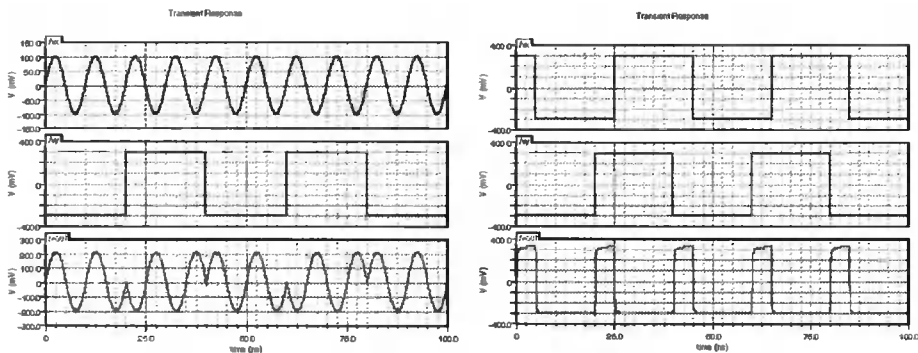


Fig. 11. Simulated transient responses: large signal at one input – phase detector (left) and both inputs overdrive – coincidence detector (right)

4. CONCLUSIONS

An analog four-quadrant CMOS transconductance multiplier in Gilbert-like architecture has been proposed. It is particularly suited for high-speed application due to its cascade architecture composed of very simple building blocks using RF CMOS transistors with sufficiently large biasing currents and may be applicable particularly in RF circuits as mixers, modulators/demodulators in communication systems. The most important benchmarks of the circuit are: quiescent current of 3.5 mA and 3 dB bandwidth

of 1 GHz. The described implementation requires a stock of at most three transistors in saturation region, so it is well suited for low voltage applications.

Both aforementioned features make our proposal an interesting alternative for another already known circuit solutions. The circuit has been sent for manufacturing by Europractice, we received back the fabricated prototypes and preliminary measurements confirm the proper operation of the circuits. An extended test plan including offset, temperature drifts as well as HF performance measurements is currently in progress.

BIBLIOGRAPHY

- [1] Chang C.C., Liu S.I., 1998. Weak inversion four-quadrant multiplier and two-quadrant divider, *Electronics Letters*, vol. 34, Nr 22, pp. 2079-2080.
- [2] Dei M., Nizza N., Lazzerini G.M., Bruschi P., Piotta M., 2009. A four quadrant analog multiplier based on a novel cmos linear current divider, *IEEE PRIME – Research in Microelectronics and Electronics Conference*, pp. 128-131.
- [3] Gilbert B., 1968. A precise four-quadrant multiplier with subnanosecond response, *IEEE J. Solid State Circuits*, vol. SC-3, pp. 365-373.
- [4] Han G., Sánchez-Sinencio E., 1998. CMOS Transconductance Multipliers: A Tutorial, *IEEE Trans. Circuits Syst. II – Analog And Digital Signal Processing*, 45, (12), pp. 1550-1563.
- [5] Salama M.K., Soliman A.M., 2003. Low-Voltage Low-Power CMOS RF Four-Quadrant Multiplier, *AEU – International Journal of Electronics and Communications*, vol. 57, Issue 1, pp. 74-78.
- [6] Sawigun C., Mahattanakul J., 2008. A 1.5 V, wide-input range, high-bandwidth, CMOS four-quadrant analog multiplier, *Proceedings of ISCAS 2008. IEEE*, pp. 2318-2321.
- [7] Walke R.L., Quigley S.F., Webb P.W., 1992. Design of an analogue subthreshold multiplier suitable for implementing an artificial neural network, *IEE Proceedings G*, vol. 139(2), pp. 261-264.

NISKONAPIĘCIOWY SZYBKI CZTEROĆWIARTKOWY TRANS-KONDUKTANCYJNY UKŁAD MNOŻĄCY W TECHNOLOGII CMOS

Streszczenie

W artykule zaprezentowano szybki niskonapięciowy czteroćwiartkowy układ mnożący zaprojektowany w technologii CMOS. Architektura układu oparta jest o strukturę typu Gilberta. W układzie zastosowano kaskadowe połączenie dwóch stopni transkonduktancyjnych zrealizowanych w oparciu o pary różnicowe. Aby układ mógł pracować w zakresie niskich napięć zasilających poszczególne stopnie zostały sprzęgnięte przy pomocy prostych luster prądowych. Duża szybkość działania została osiągnięta dzięki prostej architekturze układu oraz zastosowaniu tranzystorów RF pracujących przy odpowiednio dużych wartościach prądów. W pracy zaprezentowano również wejściowe niskonapięciowe bloki pomocnicze oraz wyniki symulacji kompletnego układu mnożącego.

Słowa kluczowe: analogowe układy VLSI, czteroćwiartkowy układ mnożący, technologia CMOS

ISSN 1899-0088