

POLSKA AKADEMIA NAUK
KOMITET ELEKTRONIKI I TELEKOMUNIKACJI

ch

KWARTALNIK
ELEKTRONIKI I TELEKOMUNIKACJI
ELECTRONICS AND
TELECOMMUNICATIONS
QUARTERLY

TOM 45 — ZESZYT 2

WYDAWNICTWO NAUKOWE PWN
WARSZAW 1999

RADA REDAKCYJNA

Przewodniczący

Prof. dr hab. inż. ALFRED ŚWIT

członek rzeczywisty PAN

Członkowie

prof. dr hab. inż. DANIEL JÓZEF BEM. — członek koresp. PAN, prof. dr hab. inż. MICHAŁ BIAŁKO — czł. rzecz. PAN, prof. dr hab. inż. STEFAN HAHN — czł. koresp. PAN, prof. dr inż. ANDRZEJ HAŁAS, prof. dr inż. ZDZISŁAW KACHLICKI, prof. dr hab. inż. WŁADYSŁAW MAJEWSKI, prof. dr hab. inż. JÓZEF MODELSKI, prof. dr inż. JERZY OSIOWSKI, prof. dr inż. WITOLD ROSIŃSKI — czł. rzecz. PAN, prof. dr hab. inż. STEFAN WĘGRZYN — czł. rzecz. PAN, prof. dr hab. inż. WIESŁAW WOLIŃSKI — czł. koresp. PAN, prof. dr inż. ANDRZEJ ZIELIŃSKI, prof. dr inż. MARIAN ZIENTALSKI

REDAKCJA

Redaktor naczelny

prof. dr hab. inż. WIESŁAW WOLIŃSKI

Zastępca Redaktora Naczelnego

doc. dr inż. KRYSTYN PLEWKO

Sekretarz Odpowiedzialny

mgr. inż. KRYSTYNA LELAKOWSKA

ADRES REDAKCJI

00-665 Warszawa, ul. Nowowiejska 15/19 Politechnika, pok. 470
Instytut Telekomunikacji, Gmach im. prof. JANUSZA GROSZKOWSKIEGO

Diżury Redakcji: środy i piątki, godz. 14–16
tel/fax (022) 660 77 37, 825 29 18+aut. sekr.

Telefony domowe: Redaktora Naczelnego: 812 17 65

Zast. Red. Naczelnego: 826 83 41

Sekretarza Odpowiedzialnego: 825 29 18

WYDAWNICTWO NAUKOWE PWN

Warszawa, ul. Miodowa 10

Ark. wyd. 13,25. Ark. druk. 10,00

Podpisano do druku w maju 1999 r.

Papier offset. kl. III 70 g. B-1

Druk ukończono w czerwcu 1999 r.

SKŁAD I ŁAMANIE: Agnieszka Chmielewska, Warszawa, ul. Korytnicka 28

DRUK I OPRAWA: Drukarnia Braci Grodzickich, Żabieniec, ul. Przelotowa 7

Szanowni Autorzy

„Kwartalnik Elektroniki i Telekomunikacji” — Electronics and Telecommunications Quarterly jest kontynuatorem tradycji powstałego 45 lat temu kwartalnika p.t. „Rozprawy Elektrotechniczne”.

Kwartalnik jest czasopismem Komitetu Elektroniki i Telekomunikacji Polskiej Akademii Nauk. Wydawany jest przez Wydawnictwo Naukowe PWN SA w Warszawie. Kwartalnik jest czasopismem naukowym, na którego łamach są publikowane artykuły i komunikaty prezentujące wyniki oryginalnych prac teoretycznych i doświadczalnych, a także przeglądowych. Związane są one z szeroko rozumianymi dziedzinami współczesnej elektroniki, telekomunikacji, mikroelektroniki, optoelektroniki, radiotechniki i elektroniki medycznej.

Autorami publikacji są wybitni naukowcy, znani specjaliści o wieloletnim doświadczeniu, a także młodzi badacze — głównie doktoranci.

Artykuły charakteryzują się oryginalnym ujęciem zagadnienia, interesującymi wynikami badań, krytyczną oceną teorii lub metod, omówieniem aktualnego stanu, lub postępu danej gałęzi techniki oraz omówieniem perspektyw rozwojowych. Sposób pisania matematycznej części artykułów zgodny jest z wytycznymi IEC (International Electronics Commission) oraz ISO (International Organization of Standardization).

Wszystkie publikowane w Kwartalniku artykuły są recenzowane przez znanych krajowych specjalistów, co zapewnia że publikacje te są uznawane jako autorski dorobek naukowy. Opublikowanie w kwartalniku wyników prac naukowych zrealizowanych w ramach „GRANTów” Komitetu Badań Naukowych spełnia więc jeden z wymogów stawianych tym pracom.

Czasopismo dociera do wszystkich zajmujących się elektroniką i telekomunikacją krajowych ośrodków naukowych oraz technicznych, a także szeregu instytucji zagranicznych. Jest ponadto prenumerowane przez liczne grono specjalistów i biblioteki.

Każdy Autor otrzymuje bezpłatnie 20 egzemplarzy nadbitek swojego artykułu, co ułatwia przesłanie go do indywidualnych wybranych przez Autora osób i instytucji w kraju, lub za granicą. Ułatwia to dodatkowo fakt, że w Kwartalniku są publikowane artykuły w języku angielskim.

Nadesłane do redakcji artykuły są publikowane w terminie około pół roku, w przypadku sprawnej współpracy Autora z Redakcją. Wytyczne dla Autorów dotyczące formy publikacji są zamieszczone w zeszytach Kwartalnika, można je także otrzymać w siedzibie Redakcji.

Artykuły można dostarczać osobiście, lub pocztą pod adresem zamieszczanym na stronie redakcyjnej w każdym zeszycie.

Redakcja

BIALKO
ANDRZEJ
KI, prof. dr
SKI — czł.
WIESŁAW
MARIAN

M
przy u

ko
Ko
me
me
ko
wy
do
wy
dr
U
me
ar

ró

Róv
nego o
element
wiązyw
począw
towa),
biologi
różnicz
numery
i przes
w tych
równań

Modelowanie równań różniczkowych cząstkowych przy użyciu uniwersalnych sieci neuronowych komórkowych

KRZYSZTOF ŚLOT

Instytut Elektroniki, Politechnika Łódzka

Otrzymano 1998.12.15

Autoryzowano 1999.05.06

Zasadniczym tematem artykułu jest analiza możliwości modelowania równań różniczkowych cząstkowych typu parabolicznego za pomocą Uniwersalnych Sieci Neuronowych Komórkowych (USNK). Podstawową zaletą przyjęcia tego sposobu modelowania jest możliwość uzyskania szybkości przetwarzania zdecydowanie przewyższających inne możliwe metody. W artykule zostaną omówione kolejno zagadnienia modelowania równań różniczkowych cząstkowych liniowych, modelowania układów równań różniczkowych cząstkowych oraz modelowania równań różniczkowych cząstkowych nieliniowych. W części dotyczącej pierwszego z wymienionych zagadnień zaprezentowano odpowiedni sposób wykorzystania sygnałów sieci dla realizacji postawionego zadania. W części poświęconej drugiemu zagadnieniu, zaproponowano metodę modelowania układu równań w strukturze USNK, polegającą na nadaniu przetwarzanej informacji odpowiedniej struktury. Dla celów modelowania równań nieliniowych przedstawiono propozycje odpowiednich modyfikacji architektury komórek sieci, dogodne z punktu widzenia fizycznej realizacji układu.

Słowa kluczowe: uniwersalne sieci neuronowe komórkowe, modelowanie równań różniczkowych, przetwarzanie informacji.

1. WSTĘP

Równania różniczkowe cząstkowe stanowią podstawowy sposób matematycznego opisu szeregu procesów i zjawisk, oparty o wyrażenie rządzących nimi elementarnych zasad i zależności, w postaci sformalizowanej. Umiejętność rozwiązywania takich równań daje możliwość analizy szerokiego zakresu zjawisk, począwszy od fizycznych (dynamika cieczy i gazów, zjawiska falowe, fizyka kwantowa), poprzez reakcje chemiczne, aż do zjawisk z innych dyscyplin nauki, jak biologia, socjologia czy ekonomia. Podstawowym sposobem rozwiązywania równań różniczkowych cząstkowych jest stosowanie czasochłonnych procedur całkowania numerycznego, narzucających konieczność dyskretyzacji równań w dziedzinie czasu i przestrzeni oraz kwantyzacji wartości zmiennych i parametrów występujących w tych równaniach. Wśród rozważanych, alternatywnych sposobów rozwiązywania równań różniczkowych cząstkowych, istotnym kierunkiem prac wydaje się być próba

wykorzystania do tego celu sieci neuronowych komórkowych [1] [2]. Droga ta jest szczególnie obiecująca, jeżeli weźmie się pod uwagę pojawiające się możliwości fizycznej realizacji tej abstrakcyjnej koncepcji, w postaci wyspecjalizowanych, półprzewodnikowych układów scalonych. Podstawową zaletą użycia sieci jako narzędzia modelowania równań różniczkowych cząstkowych jest spodziewane, radykalne zwiększenie szybkości analizy w porównaniu do klasycznych metod numerycznych, wynikające z oferowanego przez sieci neuronowe komórkowe, równoległego sposobu przetwarzania danych.

Wspólnym mankamentem dotychczasowych prac poświęconych modelowaniu równań przy użyciu sieci neuronowych komórkowych, jest rozważanie jako narzędzia realizacji zadania struktur, dla których możliwości realizacji fizycznej układu o znaczeniu praktycznym, wydają się być problematyczne (albo z uwagi na złożoność elementów procesorowych [3] [4], albo skomplikowaną architekturę [5] [6]). Zasadniczym celem tego artykułu jest próba wykazania, że realizowalne w półprzewodnikowej technologii VLSI układy oparte o koncepcję Uniwersalnej Sieci Neuronowej Komórkowej [7], mogą stanowić przydatne narzędzie ultraszybkiego modelowania wybranych kategorii równań różniczkowych cząstkowych (a więc dokonywania analiz przebiegu procesów opisanych za pomocą takich równań). Teoretyczna „uniwersalność” sieć oznacza możliwość wykonania za ich pomocą dowolnego zadania, w szczególności — modelowania równań, w drodze określenia odpowiedniego algorytmu postępowania [8]. Niestety, zastosowanie takiego podejścia w odniesieniu do rozważanej dziedziny zastosowań jest pozbawione uzasadnienia. Próba wyznaczenia rozwiązania w drodze zastosowania procedury algorytmicznej całkowicie eliminuje podstawową zaletę oferowaną przez sieć, jaką stanowi szybkość przetwarzania, ponieważ wymaga dokonania dyskretyzacji procesu przetwarzania w dziedzinie czasu. Zaproponowane w pracy metody modelowania równań zostały w związku z tym określone w sposób, zapewniający uzyskanie maksymalnej możliwej szybkości przetwarzania — odbywa się ono w trakcie pojedynczego procesu ewolucji stanu sieci, inicjowanego zadaniem odpowiednich warunków początkowych i ustaleniem odpowiednich parametrów układu.

Podstawowymi rozwiązaniami zaproponowanymi w zakresie wykorzystania Uniwersalnych Sieci Neuronowych Komórkowych (określanych w dalszej części akronimem USNK) do modelowania równań różniczkowych i przedstawionymi w artykule są:

- określenie sposobu wykorzystania struktury USNK do modelowania równań różniczkowych cząstkowych liniowych typu parabolicznego,
- sformułowanie metody modelowania układu równań różniczkowych cząstkowych przy użyciu USNK,
- określenie niezbędnych modyfikacji układowych, które umożliwiłyby modelowanie wybranych typów równań różniczkowych cząstkowych nieliniowych w strukturze bazującej na koncepcji USNK.

Podstawową wadą wykorzystania fizycznych realizacji USNK do modelowania równań jest mała dokładność przeprowadzonej analizy, spowodowana nieuchronnymi dla technologii półprzewodnikowej niedokładnościami wykonania elementów ukła-

du. W k
fizyczny
paramet
stanowi

W d
wymiar
podstaw
skrótow
zasady
cząstkow
wych dla
dziedzin
równań
czwarte
nieliniow
modyfik

Kon
Neural
stawow
przez m
Archite
wo-cyfr
dokonu
kacje w
neuron

- doda
analiz
— L
Mem
• umo
nywa
w po
Prze
równań
dowoln
równa

du. W konsekwencji, oczekiwanym wynikiem analiz przeprowadzonych przy użyciu fizycznych implementacji USNK byłoby zgrubne oszacowanie zakresów wartości parametrów równań, dających w wyniku pożądaną przebieg modelowanego procesu, stanowiąc punkt wyjścia dla szczegółowych analiz numerycznych.

W dalszej części artykułu przedstawione zostaną metody implementacji dwuwymiarowych równań różniczkowych cząstkowych typu parabolicznego w strukturze podstawowej USNK oraz proponowanych modyfikacjach tej struktury. Na wstępie skrótowo zaprezentowana zostanie koncepcja USNK oraz omówione zostaną ogólne zasady wykorzystania USNK dla celów implementacji równań różniczkowych cząstkowych. Zagadnienie modelowania równań różniczkowych cząstkowych liniowych dla warunków brzegowych określanych na dowolnych krzywych należących do dziedziny równania przedstawiono w części drugiej. Metodę implementacji układu równań różniczkowych cząstkowych w USNK omówiono w części trzeciej. W części czwartej rozważono problem implementacji równań różniczkowych cząstkowych nieliniowych oraz układu takich równań w USNK oraz zaproponowano proste modyfikacje struktury komórki pozwalające na realizację tego zadania.

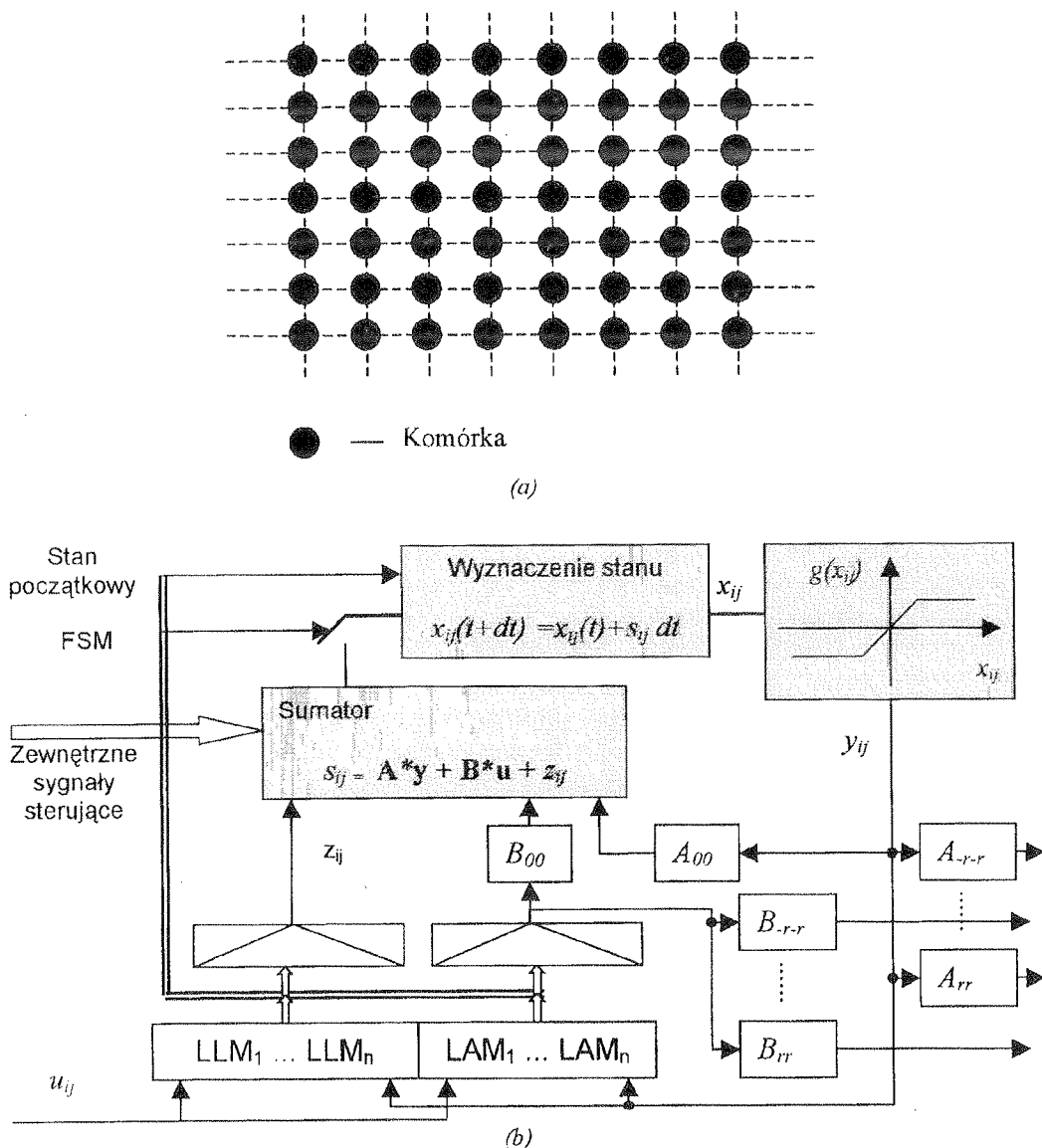
2. UNIWERSALNE SIECI NEURONOWE KOMÓRKOWE A RÓWNANIA RÓŻNICZKOWE CZĄSTKOWE

Koncepcja Uniwersalnych Sieci Neuronowych Komórkowych (ang. *Cellular Neural Network Universal Machine*), sformułowana w [8] stanowi rozwinięcie podstawowego modelu sieci neuronowych komórkowych [1], pozwalające na uzyskanie przez nie uniwersalnych (w sensie Turinga) własności przetwarzania informacji. Architektura USNK (Rys. 1a) jest dwuwymiarową macierzą procesów analogowo-cyfrowych, nazywanych komórkami (ang. *cells*). Każda komórka zawiera część dokonującą analogowego przetworzenia sygnałów sterujących, a zasadnicze modyfikacje wprowadzone w modelu USNK w porównaniu do modelu podstawowego sieci neuronowej komórkowej polegają na (Rys. 1b):

- dodaniu do każdej z komórek sieci zestawu pamięci przechowujących sygnały analogowe — lokalnych pamięci analogowych (ang. *Local Analog Memories* — LAM) i sygnały logiczne — lokalnych pamięci logicznych (ang. *Local Logic Memories* — LLM)
- umożliwieniu blokowania przetwarzania dla wybranych komórek układu, dokonywanego przy użyciu dodatkowych sygnałów sterujących, reprezentowanych w postaci tzw. maski stanów niezmiennych (ang. *Fixed State Map*).

Przetwarzanie sygnału w komórce można opisać za pomocą układu dwóch równań — równania stanu komórki oraz równania wyjściowego. Równanie stanu dowolnej komórki sieci (np. położonej w i -tym wierszu i j -tej kolumnie sieci) jest równaniem różniczkowym zwyczajnym pierwszego rzędu o postaci:

$$\frac{dx_{ij}}{dt} = -x_{ij} + \sum_{kl} A_{kl} y_{i+k, j+l} + \sum_{kl} B_{kl} u_{i+k, j+l} + z_{ij} \quad (1)$$



Rys. 1. Architektura USNK (a) i schemat blokowy jej elementu procesorowego (b)

i opisuje proces ewolucji w czasie sygnału nazwanego stanem komórki ($\{x_{ij}\}$) w funkcji sygnałów sterujących komórką ($\{y_{ij}\}$, $\{u_{ij}\}$ i $\{z_{ij}\}$) oraz parametrów sieci ($\{A_{ij}\}$, $\{B_{ij}\}$). Parametry $\{A_{ij}\}$ i $\{B_{ij}\}$ determinują siłę z jaką sygnały sterujące oddziałują na komórkę i są one jednakowe dla wszystkich komórek sieci. W celu odzwierciedlenia topologii połączeń międzyelementowych są one wyrażane w postaci macierzy kwadratowych nazywanych operatorem sprzężenia zwrotnego (A) i operatorem sterowania (B). Dla rozważanego w artykule przypadku fizycznie realizowalnych USNK, macierze te mają rozmiar 3×3 :

co ozna
nymi k
komork
rki wie
sieci. M
kombin
różnico
Ró
w sygn
postaci

przeds
sieci za
komór

Ro
nego,

gdzie
— oz
o „szy
nielin
W
nume
różni
nej o
równ
cja r
Lapla

$$\mathbf{A} = \begin{bmatrix} A_{-1-1} & A_{-10} & A_{-11} \\ A_{0-1} & A_{00} & A_{01} \\ A_{1-1} & A_{10} & A_{11} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} B_{-1-1} & B_{-10} & B_{-11} \\ B_{0-1} & B_{00} & B_{01} \\ B_{1-1} & B_{10} & B_{11} \end{bmatrix} \quad (2)$$

co oznacza istnienie w sieci bezpośrednich połączeń jedynie między najbliższymi komórkami (przykładowo, element A_{-1-1} pośredniczy w oddziaływaniu na komórkę położoną w i -tym wierszu i j -tej kolumnie struktury, sygnału $y_{i-1,j-1}$ komórki wiersza $i-1$ i kolumny $j-1$). Zbiór parametrów (2) nazywany jest szablonem sieci. Można zauważyć, że równanie stanu komórki może być traktowane jako kombinacja równania różniczkowego (w odniesieniu do dziedziny czasu) i równania różnicowego (w odniesieniu do dziedziny przestrzennej).

Równanie wyjścia komórki opisuje sposób w jaki sygnał stanu jest przekształcony w sygnał wyjściowy komórki ($\{y_{ij}\}$), i ma postać odcinkowo liniowej charakterystyki postaci:

$$y_{ij} = g(x_{ij}) = \frac{1}{2}(|x_{ij}-1| + |x_{ij}+1|) \quad (3)$$

przedstawionej szkiecowo na Rys. 1b. Jeżeli zapewnimy by sygnały stanu komórek sieci zawierały się w przedziale wartości $[-1...1]$, wtedy $y_{ij} = x_{ij}$, a równania stanu komórek takiej sieci stają się równaniami liniowymi o postaci:

$$\frac{dx_{ij}}{dt} = -x_{ij} + \sum_{kl} A_{kl} x_{i+k,j+l} + \sum_{kl} B_{kl} u_{i+k,j+l} + z_{ij} \quad (4)$$

2.1. RÓWNANIA RÓŻNICZKOWE CZĄSTKOWE A KONCEPCJA SNK

Rozważmy równanie różniczkowe cząstkowe rzędu drugiego typu parabolicznego, określone w postaci ogólnej zależnością:

$$\frac{\partial v(\mathbf{r}, t)}{\partial t} = f(v(\mathbf{r}, t) + k \nabla^2 v(\mathbf{r}, t)), \quad (5)$$

gdzie t — jest zmienną oznaczającą czas, $\mathbf{r}$ — jest współrzędną przestrzenną, ∇^2 — oznacza operator Laplace'a (dywergencji), k — dodatnią stałą (decydującą o „szybkości” dyfuzji wartości zmiennej v), zaś $f(v)$ jest pewną, w ogólnym przypadku nieliniową, funkcją zmiennej v .

W celu rozwiązania równania różniczkowego cząstkowego przy użyciu metod numerycznych, konieczne jest przejście od rachunku różniczkowego do rachunku różnicowego w odniesieniu zarówno do zmiennej czasowej jak i zmiennej przestrzennej oraz kwantyzacja zbioru wartości, jakie przyjmować mogą zmienne i parametry równań. W odniesieniu do równań różniczkowych typu parabolicznego, dyskretyzacja równania w dziedzinie zmiennej przestrzennej oznacza zastąpienie operatora Laplace'a jego „różnicowymi” odpowiednikami. Aproksymacje operatora Laplace'a

wyrażają się, dla kolejno zwiększającej się wymiarowości dziedziny równania, zależnościami:

$$\begin{aligned}\nabla^2 v(x) &\cong v(i+1) + v(i-1) - 2v(i) \\ \nabla^2(x, y) &\cong v(i-1, j) + v(i, j-1) + v(i, j+1) + v(i+1, j) - 4v(i, j) \\ \nabla^2 v(x, y, z) &\cong v(i-1, j, k) + v(i, j-1, k) + v(i, j+1, k) + v(i+1, j, k) + \\ &\quad v(i, j, k-1) + v(i, j, k+1) - 6v(i, j, k),\end{aligned}\quad (6)$$

gdzie x, y, z są ciągłymi współrzędnymi przestrzennymi aproksymowanego równania różniczkowego, zaś i, j, k są odpowiadającymi im, dyskretnymi współrzędnymi równania różnicowego. Przyjęcie przedstawionego sposobu aproksymacji operatora Laplace'a sprawia, że (dla przypadku dziedziny dwuwymiarowej) równanie (5) zostaje przekształcone w układ równań różniczkowych zwyczajnych w postaci:

$$\frac{dv_{ij}(t)}{dt} \cong f(v_{ij}(t)) + k \cdot (v_{i-1,j}(t) + v_{i,j-1}(t) + v_{i,j+1}(t) + v_{i+1,j}(t) - 4v_{ij}(t)) \quad (7)$$

określonych w całej dziedzinie przestrzeni dyskretniej i, j . Zauważmy, że zbiór równań (7) jest formalnie identyczny ze zbiorem równań stanu komórek dwuwymiarowej sieci neuronowej komórkowej (4), dla której przyjęte zostały następujące wartości parametrów:

$$\mathbf{A} = k \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{B} = 0, \quad I = 0 \quad (8)$$

przy założeniu, że funkcja $f(\cdot)$ użyta w równaniu (5) jest związana z wielkościami występującymi w opisie komórki w następujący sposób:

$$f(v_{ij}) = -v_{ij}. \quad (9)$$

Oznacza to, że sieci neuronowe komórkowe mogą stanowić dogodne narzędzie modelowania równań różniczkowych cząstkowych rzędu drugiego typu parabolicznego. Jeżeli funkcja $f(\cdot)$ spełnia warunek (9) to modelowania równania w sieci neuronowej komórkowej polega na ustaleniu wag odpowiadających szablonowi (8), wprowadzeniu do sieci sygnałów odpowiadających warunkom początkowym równania (w postaci stanów początkowych komórek sieci), oraz na odpowiednim wymuszeniu warunków brzegowych. Proces „rozwiązywania” równania będzie odpowiadał samorzutnej ewolucji stanu sieci zapoczątkowanego warunkami początkowymi i zmierzającego ku jej końcowemu stanowi stabilnemu (o ile stan taki istnieje). Czas ewolucji stanu sieci zależy od natury modelowanego procesu i od parametrów dynamicznych układu stanowiącego narzędzie modelowania. Czas ten, w przypadku wyboru układu [7] nie przekracza kilku milisekund, dla wszystkich przedstawionych w artykule przypadków. Ponieważ rozmiar sieci — 64×64 komórek — oznacza modelowanie układu 4096 równań różniczkowych zwyczajnych, czas ten

jest bezk
numeryc

W pr
użyciu m
których
nie wyst
zmienny
równań
dyskusji
kompute
charakte
przedmi
chemiczn
odwier
Klasyczn
przestrz
wacją, je
autonom
w oparc

Pod
narzędz

- zagac
- warto
- skoń
- szają
- ograni
- padk
- prob

Pier
stanie v
sieci, d
propon
wyłącza
ków b
cząstk
tury ele

3. M

Ro
nianie c

jest bezkonkurencyjny w porównaniu do stosowania komputerowych obliczeń numerycznych.

W przeciwieństwie do rozwiązywania równań różniczkowych cząstkowych przy użyciu metod numerycznych, w sieciach zrealizowanych fizycznie (a więc takich, dla których przetwarzanie sygnałów w komórkach odpowiada zależnościom (1)–(4)), nie występuje dyskretyzacja równań w dziedzinie czasu oraz kwantyzacja wartości zmiennych tych równań. Zagadnienie błędów wnoszonych przez dyskretyzację równań różniczkowych cząstkowych w dziedzinie przestrzennej wymaga odrębnej dyskusji. Jeżeli przedmiotem modelowania jest samo równanie, użycie jako narzędzia komputera (metod numerycznych) lub sieci komórkowych implikuje przybliżony charakter rozwiązania i wynikające z tego niedokładności rezultatu. Jeżeli jednak przedmiotem modelowania jest zjawisko opisane za pomocą takich równań (fizyczne, chemiczne itp.), dyskretyzacja w dziedzinie przestrzennej może okazać się lepszym odzwierciedleniem natury tego zjawiska niż przyjęcie jego ciągłej interpretacji. Klasycznym przykładem sytuacji, w której przyjęcie modelu dyskretnego w dziedzinie przestrzeni stanowi podejście prowadzące do uzyskania wyników zgodnych z obserwacją, jest występujące dla pewnych warunków zjawisko załamania się propagacji fal autonomicznych [4] w ośrodkach aktywnych, nie posiadające poprawnego opisu w oparciu o model ciągły w dziedzinie przestrzeni.

Podstawowymi problemami związanymi z wyborem fizycznej realizacji sieci jako narzędzia modelowania równań różniczkowych cząstkowych są:

- zagadnienie wpływu niedokładności realizacji parametrów sieci i reprezentacji wartości przetwarzanych w sieci sygnałów na uzyskiwane wyniki
- skończony rozmiar sieci (64×64 komórki, co odpowiada układowi [7]), pogarszający dokładność aproksymacji równań różniczkowych cząstkowych
- ograniczenie zakresu zadań możliwych do bezpośredniego modelowania do przypadku implementacji równań liniowych
- problem określenia sposobu zadawania warunków brzegowych równań.

Pierwsze dwa z wymienionych zagadnień zostały uwzględnione poprzez wykorzystanie w symulacji pracy układu rzeczywistych rozrzutów wykonania parametrów sieci, deklarowanych dla układu [7]. Dla zadawania warunków brzegowych zaproponowano odpowiednie wykorzystanie maski stanów niezmiennych sieci (FSM), wyłączające z przetwarzania wymagane obszary, inicjowane wymaganymi dla warunków brzegowych wartościami. Dla celów modelowania równań różniczkowych cząstkowych nieliniowych zostały zaproponowane odpowiednie modyfikacje struktury elementu procesorowego sieci.

3. MODELOWANIE RÓWNAŃ RÓŻNICZKOWYCH CZĄSTKOWYCH LINIOWYCH ZA POMOCĄ USNK

Rozważmy równanie różniczkowe cząstkowe typu parabolicznego liniowe (równanie dyfuzji) o postaci ogólnej:

$$\frac{\partial v(\mathbf{r}, t)}{\partial t} = a \cdot v(\mathbf{r}, t) + b + k \nabla^2 v(\mathbf{r}, t) \quad (10)$$

i założmy, że celem postępowania jest próba modelowania takich równań przy użyciu rozważanej uniwersalnej sieci neuronowej komarkowej. Uwzględniając, że w przypadku dwuwymiarowym $\mathbf{r} = [x, y] \in \mathbb{R}^2$ po odpowiedniej dyskretyzacji operatora Laplace'a, równanie (10) może zostać zastąpione układem równań różniczkowych zwyczajnych liniowych w postaci:

$$\frac{dv_{i,j}(t)}{dt} = a \cdot v_{i,j}(t) + b + k[v_{i-1,j}(t) + v_{i+1,j}(t) + v_{i,j-1}(t) + v_{i,j+1}(t) - 4v_{i,j}(t)] \quad (11)$$

gdzie: $0 \leq i < M$, $0 \leq j < N$, a M i N określają granice dyskretnej dziedziny równania.

Aby można było dokonać modelowania równania (10) przy użyciu USNK konieczny jest dobór właściwych wartości parametrów szablonu sieci, narzucenie żądanych warunków brzegowych i zapewnienie by równanie stanu komórki było równaniem liniowym o postaci danej zależnością (4). Odpowiednia dla implementacji równania (10) postać szablonu sieci jest następująca:

$$\mathbf{A} = k \cdot \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 + \frac{a+1}{k} & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{B}, \mathbf{Z}. \quad (12)$$

Sposób reprezentacji składnika b w zależności (10) może być różny. Pierwszym możliwym wariantem jest przyjęcie kombinacji wartości $\mathbf{B} = 0$ i $\mathbf{Z} = [b]_{M \times N}$. Drugim rozwiązaniem jest założenie że $\mathbf{B} = [1]$, odpowiednie wartości sygnału wejściowego sieci $u_{ij} = b$, zaś $\mathbf{Z} = 0$. Warunek pozwalający na określenie czy dla parametrów równania (10) i szablonu postaci (12) możliwa jest praca komórek sieci w zakresie liniowym został sformułowany w pracy [9].

Jeżeli modelowane w sieci równanie posiada stabilne rozwiązanie, zmiany wartości sygnałów stanu komórek dążą do zera, co oznacza, że:

$$\forall_{i,j}: \lim_{t \rightarrow \infty} a \cdot v_{i,j}(t) + b + k[v_{i-1,j}(t) + v_{i+1,j}(t) + v_{i,j-1}(t) + v_{i,j+1}(t) - 4v_{i,j}(t)] = 0 \quad (13)$$

a stan stabilny sieci odpowiada rozwiązaniu równania:

$$a \cdot v(\mathbf{r}, t) + b + k \nabla^2 v(\mathbf{r}, t) = 0. \quad (14)$$

Szczególnymi przypadkami powyższego równania o istotnym znaczeniu praktycznym są równanie Laplace'a

$$\nabla^2 v(\mathbf{r}, t) = 0 \quad (15)$$

i równanie Poissona.

Ponieważ wartości wymagane

Różnica równań potencjałów wokół $V_1 < V_2$ konieczne jest wykorzystać o porównaniu określonych wszystkich będących Oznaczenia równań z przedmiotem nych komórek brzegowych twarzą go obrotu potencjału w sieci równe uzyskane dzony dyskretne reprezentacje przed Cz w oparciu z wyznaczenia k dziesiętnik o

$$(10) \quad \nabla^2 v(\mathbf{r}, t) = d(\mathbf{r}). \quad (16)$$

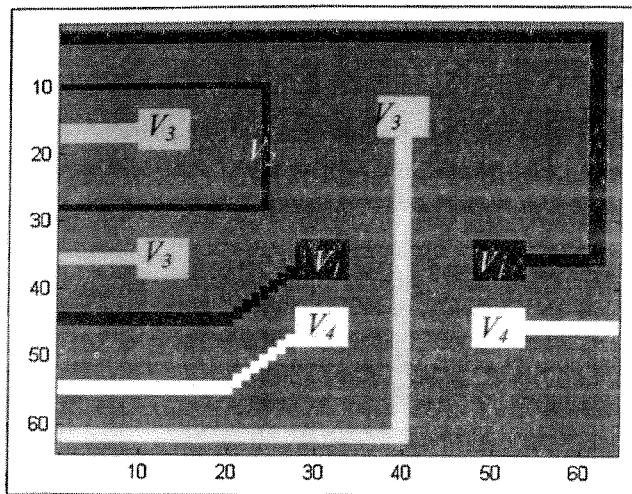
Ponieważ w obydwu powyższych przypadkach parametry równania (10) przyjmują wartości $a=0$, $k=1$, dla modelowania w sieci obydwu przedstawionych równań wymagane jest użycie szablonu zawierającego operator sprzężenia zwrotnego:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -3 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \quad (17)$$

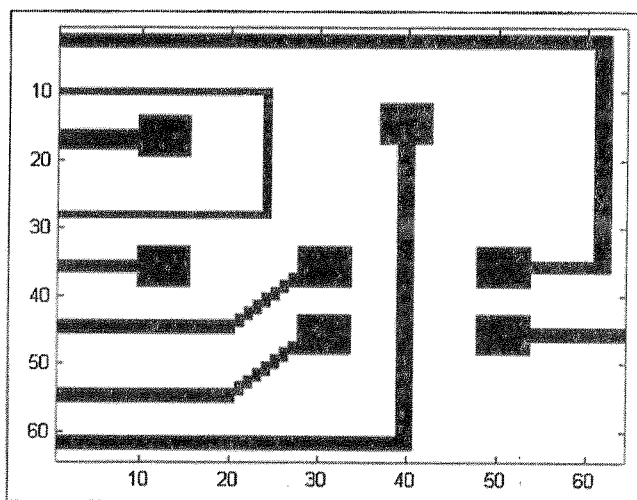
Rozważmy zastosowanie półprzewodnikowej realizacji USNK [7] do rozwiązywania równania Laplace'a i założmy, że celem postępowania jest określenie rozkładu potencjału i natężenia pola elektrycznego w wybranym przekroju poprzecznym wokół układu przewodników naładowanych do potencjałów odpowiednio: $V_1 < V_2 < V_3 < V_4$ (przedstawionych na Rys. 2a). Aby określić rozkład potencjału, konieczne jest rozwiązanie równania Laplace'a w obszarze wokół przewodników, których kształty i potencjały stanowią warunki brzegowe równania. Szablon sieci wykorzystanej w celu rozwiązania zadania zawiera operator sprzężenia zwrotnego $\mathbf{A}$ o postaci (17), a pozostałe dwa elementy zależności (12) — $\mathbf{B}$ i $\mathbf{Z}$, są zerowe. Dla tak określonego szablonu sieci można wykazać [9], że jeżeli warunki początkowe wszystkich komórek sieci znajdują się w zakresie liniowym funkcji wyjścia komórki, będą one także leżały w zakresie liniowym w trakcie całego procesu przetwarzania. Oznacza to możliwość modelowania w takiej sieci równania Laplace'a, będącego równaniem liniowym. Warunki początkowe komórek sieci mają postać identyczną z przedstawioną na Rys. 2a (obraz ma rozmiar 64×64 piksele), zaś w celu odpowiedniego określenia warunków brzegowych, wykorzystana jest maska stanów niezmiennych o postaci przedstawionej na Rys. 2b (zezwalająca na zmiany stanu jedynie dla komórek odpowiadającym białym punktom tego obrazu). Stałe (zerowe) warunki brzegowe zostały określone również na krańcach dziedziny równania. Wynik przetwarzania przeprowadzonego w sieci przedstawiony został w formie dwuwymiarowego obrazu na Rys. 2c (poziomy szarości odpowiadają wartościom wyznaczonego potencjału) oraz trójwymiarowego wykresu na Rys. 2d. Symulacje przetwarzania w sieci zostały przeprowadzone przy założonej niedokładności wykonania parametrów układu i niedokładnej reprezentacji sygnałów wynoszących 2% ([7]). Następnie uzyskany wynik został porównany z wynikami obliczeń numerycznych, przeprowadzonych przy użyciu programu Matlab (dla czterokrotnie większej rozdzielczości dyskretnej dziedziny przetwarzania), a różnica między obydwoma rezultatami, reprezentująca w przybliżeniu błąd modelowania równania w USNK, została przedstawiona na Rys. 2e.

Czas uzyskania stabilnego rozwiązania dla przedstawionego przykładu, określony w oparciu o parametry układu [7] wynosi około 0,5 milisekundy. Korzystając z wyznaczonego rozkładu potencjału pola elektrycznego można, w wyniku wykonania kilku dodatkowych cykli przetwarzania (o czasie trwania wynoszącym kilkadziesiąt mikrosekund) określić wartości modułów natężeń pola elektrycznego, a wynik odpowiednich obliczeń został przedstawiony na Rys. 2f.

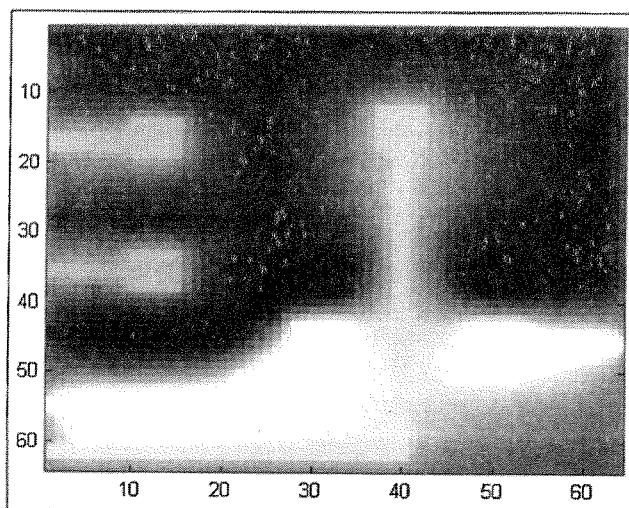
(a)



(b)

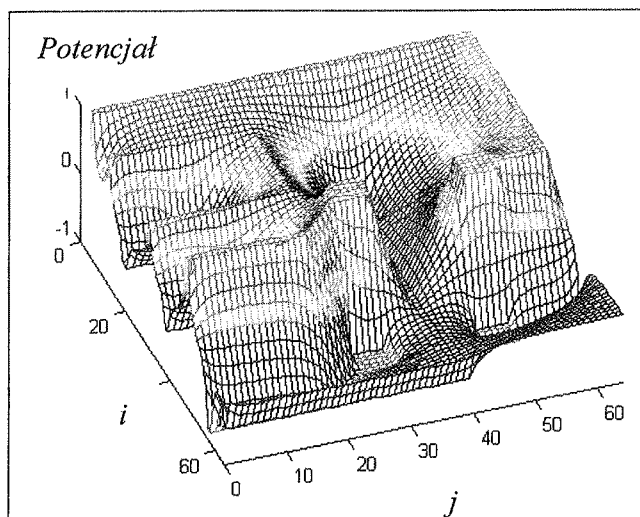


(c)

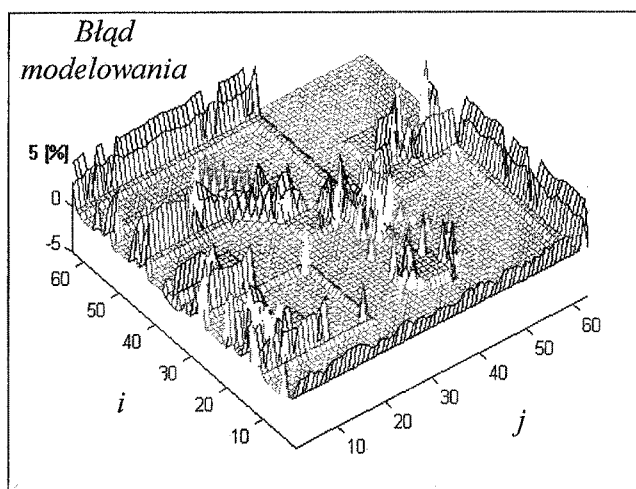


Rys. 2. Θ
początko
o wielu p

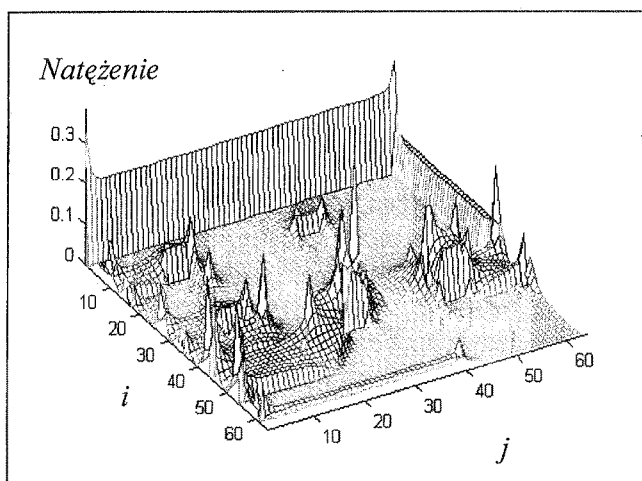
(d)



(e)



(f)



Rys. 2. Obliczanie potencjału i modułu natężenia pola elektrycznego przy użyciu USNK: (a) stan początkowy sieci, (b) maska stanów niezmiennych, (c) wyznaczony rozkład potencjału w formie obrazu o wielu poziomach szarości, (d) trójwymiarowy wykres, (e) błąd określenia wyniku, (f) rozkład modułu natężenia pola

4. MODELOWANIE UKŁADU PARABOLICZNYCH RÓWNAŃ RÓŻNICZKOWYCH CZĄSTKOWYCH LINIOWYCH W USNK

Założmy, że dany jest układ dwóch równań różniczkowych cząstkowych liniowych, o postaci ogólnej:

$$\begin{cases} \frac{\partial v(x, y, t)}{\partial t} = a_v \cdot v(x, y, t) + b_v \cdot w(x, y, t) + c_v + k_v \nabla^2 v(x, y, t) \\ \frac{\partial w(x, y, t)}{\partial t} = a_w \cdot w(x, y, t) + b_w \cdot v(x, y, t) + c_w + k_w \nabla^2 w(x, y, t), \end{cases} \quad (18)$$

gdzie k_v i k_w oznaczają stałe dyfuzji określone dla zmiennych v i w . Odpowiadające mu zestawy równań różniczkowych zwyczajnych, otrzymane w wyniku dyskretyzacji w dziedzinie przestrzeni powyższego układu równań, mają postać:

$$\begin{aligned} \frac{dv_{i,j}}{dt} &= a_v v_{i,j} + b_v w_{i,j} + c_v + k_v [v_{i-1,j} + v_{i+1,j} + v_{i,j-1} + v_{i,j+1} - 4v_{i,j}] \\ \frac{dw_{i,j}}{dt} &= a_w w_{i,j} + b_w v_{i,j} + c_w + k_w [w_{i-1,j} + w_{i+1,j} + w_{i,j-1} + w_{i,j+1} - 4w_{i,j}] \end{aligned} \quad (19)$$

$0 \leq i < M$, $0 \leq j < N$, a M i N stanowią liczbę wierszy i kolumn dwuwymiarowej dyskretnej siatki, będącej dziedziną układu, zaś dla zwiększenia przejrzystości zapisu pominięto jawne odwołania do czasu jako zmiennej niezależnej równań.

W przeciwieństwie do omówionego poprzednio zadania modelowania pojedynczych równań różniczkowych cząstkowych, bezpośrednia implementacja układu takich równań w strukturze USNK nie jest możliwa, z uwagi na wymaganą konieczność reprezentacji przy użyciu elementu procesorowego sieci dwóch niezależnych zmiennych. Podejściem narzucającym się dla realizacji w USNK problemów niewykonalnych przy użyciu pojedynczego szablonu, jest utworzenie odpowiedniego algorytmu postępowania („programu” USNK — [8]). W rozważanym przypadku program taki musi zapewnić skojarzenie z każdą komórką sieci dwóch sygnałów stanu (przy wykorzystaniu lokalnych pamięci komórek) i związane z tym podzielenie wymaganych obliczeń na etapy. Nieuchronność rozwiązywania przedstawionego układu równań w postaci iteracyjnie powtarzanej sekwencji operacji, wprowadzająca element dyskretyzacji obliczeń w dziedzinie czasu, radykalnie pogarsza możliwość do uzyskania szybkość przetwarzania. Biorąc dodatkowo pod uwagę niezwykle małą (w porównaniu z symulacją komputerową) precyzję obliczeń analogowych, stosowanie algorytmów USNK do realizacji postawionego zadania staje się pozbawione uzasadnienia. Alternatywną metodą realizacji zadań niewykonalnych w sieci zawierającej komórki zdefiniowane równaniami (1) i (3) jest odpowiednia modyfikacja struktury sieci [5] [6] lub postaci jej elementów [3], „dopasowująca” je do charakteru postawionego problemu, co zwiększa jednak złożoność układu i jak wcześniej wspomniano, ogranicza możliwość fizycznej implementacji sieci.

Poniżej
do efekty
wymaga
danych o
użycia w
metody je
zwrotnego

gdzie a ,
elementó
siebie zb
suma ind
go, dla k

Rys. 3. Str

Zau

staje się
sieci k
wyłącza
jeżeli d
 T_v i T_w

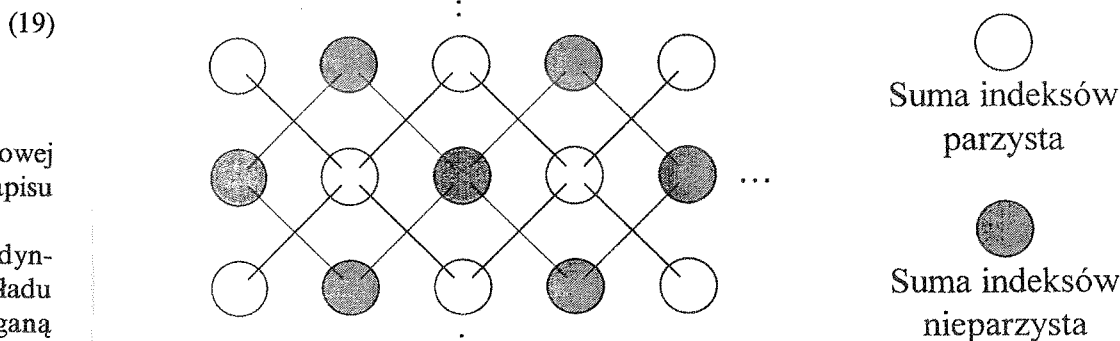
AŃ
NK
ych linio-

Poniżej pokazane zostanie, że możliwa jest inna droga postępowania, prowadząca do efektywnej implementacji układu (19) w USNK. Przedstawiony sposób nie wymaga dokonywania w sieci żadnych zmian i polega na nadaniu przetwarzanym danych odpowiedniej organizacji, pociągającej ze sobą konieczność jednoczesnego użycia w trakcie przetwarzania dwóch różnych szablonów. Podstawą proponowanej metody jest wykorzystanie właściwości szablonów zawierających operator sprzężenia zwrotnego o postaci ogólnej:

$$(18) \quad \mathbf{A} = \begin{bmatrix} a & 0 & b \\ 0 & S & 0 \\ c & 0 & d \end{bmatrix} \quad (20)$$

ne w wyni-
ją postać:

gdzie a, b, c, S są dowolnymi wartościami. Niezerowość wyłącznie diagonalnych elementów szablonu implikuje powstanie dwóch rozłącznych, nie oddziałujących na siebie zbiorów komórek (Rys. 3): pierwszego tworzonego przez komórki, których suma indeksów jest liczbą parzystą (nazywanego dalej „podsiecią parzystą”) i drugiego, dla którego suma ta jest liczbą nieparzystą („podsieci nieparzystej”).



miarowej
ści zapisu

pojedyn-
a układu
wymaganą
niezależ-
oblemów
wiedniego
rzypadku
sygnałów
podzielenie
awionego
wadzająca
żliwą do
e małą (w
osowanie
e uzasad-
wierającej
struktury
arakteru
wcześniej

Rys. 3. Struktura połączeń międzykomórkowych dla operatora sprzężenia zwrotnego zawierającego jedynie elementy diagonalne

Zauważmy, że przyjmując w macierzy (20) szczególne wartości parametrów:

$$\mathbf{A} = k \cdot \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 + \frac{a+1}{k} & 0 \\ 1 & 0 & 1 \end{bmatrix} \quad (21)$$

staje się ona dyskretnym odpowiednikiem operatora Laplace'a dla dwuwymiarowej sieci komórek o organizacji diagonalnej, zapewniając dyfuzję sygnałów stanu wyłącznie między komórkami należącymi do jednej z podsieci układu. Oznacza to że jeżeli dla podsieci parzystej i podsieci nieparzystej określimy odpowiednio szablony T_v i T_w w następujący sposób:

$$T_v: A_v = k_v \cdot \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 + \frac{a_v+1}{k_v} & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad B=0, \quad I=c_v \quad (22)$$

$$T_w: A_w = k_w \cdot \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 + \frac{a_w+1}{k_w} & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad B=0, \quad I=c_w, \quad (23)$$

to tak zdefiniowana sieć będzie modelowała układ równań (18), w którym stałe b_v i b_w są równe zero. Aby uwzględnić oddziaływania między obydwoma systemami równań, konieczne jest zapewnienie niezerowości współczynników b_v i b_w , co jest dokonywane poprzez wprowadzenie dodatkowych elementów do operatorów sprzężenia zwrotnego szablonów obydwu podsieci. W rezultacie, implementacja układu równań (18) w pełnej postaci jest możliwa w sieci określonej przez dwa różne szablony, osobno dla komórek podsieci parzystej i osobno dla nieparzystej, zawierające odpowiednio operatory sprzężenia zwrotnego A_v i A_w :

$$A_v = k_v \cdot \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 + \frac{a_v+1}{k_v} & b_v \\ 1 & 0 & 1 \end{bmatrix} \quad A_w = k_w \cdot \begin{bmatrix} 1 & 0 & 1 \\ b_w & -4 + \frac{a_w+1}{k_w} & 0 \\ 1 & 0 & 1 \end{bmatrix}. \quad (24)$$

Używając przedstawionego sposobu organizacji przetwarzanych sygnałów i stosując podane szablony, możliwe jest wyznaczanie rozwiązania układu równań różniczkowych cząstkowych liniowych w trakcie pojedynczego cyklu ewolucji stanu sieci. Ceną jaką trzeba zapłacić za możliwość jednoczesnego rozwiązywania układu równań przy użyciu przedstawionej metody jest dwukrotne zmniejszenie przestrzennej dziedziny równań w stosunku do liczby komórek sieci.

Korzystając z zaproponowanej metody rozważmy modelowanie w USNK układu następujących równań różniczkowych cząstkowych liniowych:

$$\begin{aligned} \frac{\partial a}{\partial t} &= 5a - 6i + 1 \\ \frac{\partial i}{\partial t} &= -7i + 6a + 1 + \nabla^2 i \end{aligned} \quad (25)$$

stanowiących jedną z pierwszych prób zaproponowanego przez Turinga [10] matematycznego opisu procesu spontanicznego formowania się wzorów (przedstawiony model opisuje oddziaływanie inhibitora — i — i aktywatora — a — „pigmentu”). Rozwiązania powyższego układu równań można dokonać przy użyciu USNK zapewniając odpowiednią organizację przetwarzanej w sieci informacji i stosując odpowiednio szablony:

dla podsieci

dla podsieci

Podstawy komórkowej — brak możliwości funkcji wybranych równań liniowych — będzie wprowadzenie błędów. Jest to zjawisko wartości zmiennej zostanie do

5. MODEL CZĄSTKOWY

Równania — znaczenie nieliniowych w (statycznej) — n — autonomicznej — wowe [3] — [12].

Jedną — wartości — ciedlona — komórki — fizycznym — itd.). W — nych — w których — tem funk-

$$(22) \quad A_a = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 6 & -6 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_a = 0, \quad I_a = 1 \quad (26)$$

dla podsieci parzystej, reprezentującej zmiany koncentracji aktywatora, oraz:

$$(23) \quad A_i = \begin{bmatrix} 1 & 0 & 1 \\ 6 & -10 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad B_i = 0, \quad I_i = 1 \quad (27)$$

tałe b_v i b_w
ni równań,
konywane
nia zwrot-
wnań (18)
osobno dla
powiednio

dla podsieci nieparzystej (koncentracja inhibitora).

Podstawowym problemem wyłaniającym się przy próbie modelowania w sieci komórkowej równań liniowych i układów takich równań jest, w ogólnym przypadku, brak możliwości zapewnienia pracy wszystkich komórek sieci w zakresie liniowym funkcji wyjścia (3). Jeżeli celem postępowania jest modelowanie równania lub układu równań liniowych, skutkiem wchodzenia sygnałów komórek w zakresy nasycenia będzie wprowadzenie do przetwarzania dodatkowych i zwykle narastających w czasie błędów. Jeżeli jednak celem zastosowania sieci będzie modelowanie procesów i zjawisk fizycznych opisanych przy użyciu takich równań, ograniczony zakres wartości zmiennych może okazać się zgodny z naturą modelowanych procesów, co zostanie dokładniej omówione w następnej części artykułu.

(24)

5. MODELOWANIE WYBRANYCH RÓWNAŃ RÓŻNICZKOWYCH CZĄSTKOWYCH NIELINIOWYCH I UKŁADÓW TAKICH RÓWNAŃ

i stosując
różnic-
anu sieci.
a układu
rzestrzen-

Równania różniczkowe cząstkowe nieliniowe typu parabolicznego mają istotne znaczenie dla opisu i modelowania procesów zachodzących w ośrodkach o nieliniowych własnościach, jak na przykład zjawiska spontanicznej samoorganizacji (statycznej — samorzutne różnicowanie struktur biologicznych [10] [11], i dynamicznej — np. reakcja Bielousowa-Żabotyńskiego [11]), powstawanie i propagacja fal autonomicznych (elektryczna aktywność mięśnia sercowego, przewodnictwo nerwowe [3]), zjawiska termodynamiki [3] czy oddziaływania układów kwantowych [12].

(25)

matema-
stawiony
mentu”).
USNK
stosując

Jedną z cech charakterystycznych wielkości fizycznych jest ograniczony zakres wartości jakie wielkości te mogą przyjmować. Cecha ta jest bardzo dobrze odzwierciedlona przez sposób zdefiniowania funkcji wyjścia komórki — sygnały wyjściowe komórki są ograniczone, przez co można je realistycznie wiązać z wielkościami fizycznymi (stopnie koncentracji substancji, polaryzacji elektrycznej, magnetycznej itd.). W pracy [5] przeanalizowane zostały zjawiska zachodzące w układach fizycznych modelowanych przy użyciu nieliniowych równań różniczkowych cząstkowych, w których zależności nieliniowe mają charakter sigmoidalny, a więc zgodny z kształtem funkcji wyjścia komórki. Przedmiotem analizy był układ równań różniczkowych:

$$\begin{cases} \frac{\partial v(x, y, t)}{\partial t} = -v + a_v \cdot f(v) + b_v \cdot f(w) + c_v + k_v \nabla^2 f(v) \\ \frac{\partial w(x, y, t)}{\partial t} = -w + a_w \cdot f(w) + b_w \cdot f(v) + c_w + k_w \nabla^2 f(w), \end{cases} \quad (28)$$

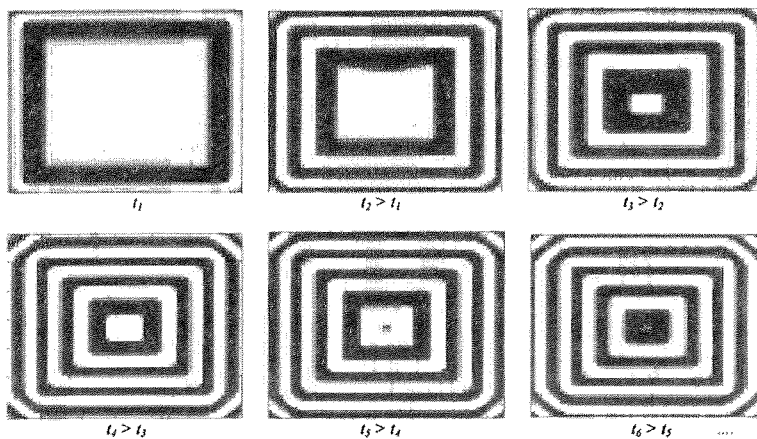
gdzie funkcja nieliniowa $f(\cdot)$ jest identyczna z funkcją wyjścia komórki (3). Zamiast modelowania powyższego układu równań za pomocą hipotetycznej sieci neuronowej komórkowej dwuwarstwowej, jak miało to miejsce w pracy [5], można wykorzystać przedstawioną w poprzedniej części metodę implementacji układu równań przy pomocy USNK. Jednym ze zjawisk fizycznych, charakterystycznych dla aktywnych ośrodków, których modelowanie może być dokonane za pomocą przedstawionego układu równań jest powstawanie tzw. fal koncentrycznych (ang. *concentric waves*, *target waves*). Przykładowe wyniki modelowania zjawiska powstawania fal koncentrycznych w ośrodku opisanym za pomocą układu równań:

$$\begin{cases} \frac{\partial v(x, y, t)}{\partial t} = -v + 1.7 \cdot f(v) - f(w) + 0.3 + 0.1 \cdot \nabla^2 f(v) \\ \frac{\partial w(x, y, t)}{\partial t} = -w + 1.7 \cdot f(w) + f(v) - 0.3 + 0.1 \cdot \nabla^2 f(w) \end{cases} \quad (29)$$

i modelowanego w USNK opisanego szablonami:

$$T_v: A_v = \begin{bmatrix} 0.1 & 0 & 0.1 \\ 0 & 1.3 & -1 \\ 0.1 & 0 & 0.1 \end{bmatrix}, B=0, I=0.3 \quad T_w: A_w = \begin{bmatrix} 0.1 & 0 & 0.1 \\ 1 & 1.3 & 0 \\ 0.1 & 0 & 0.1 \end{bmatrix}, B=0, I=-0.3 \quad (30)$$

zostały przedstawione na Rys. 4.



Rys. 4. Ewolucja procesu tworzenia się fal koncentrycznych, modelowania przy użyciu USNK

W przy
na brzegac
sygnałów k
wartości ś
sygnałów.
przewodnik
nych, do r
liwościowy

W przy
cząstkowy
ich rozwią
modyfikac
w dotychc
diode) [13].
trzech rów
zbliżonej
Chua do
intensywn
elementem
sieci o wię
obecnej r
wydaje się
komórki p
o inne ka

Rozw

i założymy
drugiego:

Przyk
badań na
genów w
niu do ca
których
struktury
realizacji
możliwo
tym sygn
polega n
rów ana

(28)

3). Zamiast neuronowej wykorzystywać warianty przy aktywnych stawianego *intric waves*, al koncent-

W przykładzie założono stałość (zerowość) warunków brzegowych określonych na brzegach kwadratowej siatki o rozmiarze 64×64 , a jako wartości początkowe sygnałów komórek sieci przyjęto szum biały o rozkładzie gaussowskim, zerowej wartości średniej i wariancji wynoszącej 2% pełnego zakresu dynamiki zmian sygnałów. Uwzględnienie niedokładnej reprezentacji założonych parametrów w półprzewodnikowej realizacji USNK prowadziło, w stosunku do symulacji numerycznych, do różnic o charakterze wyłącznie ilościowym, dotyczącym wartości częstotliwościowych przestrzennych i czasowych uzyskanych fal.

(29)

W przypadku gdy funkcja nieliniowa występująca w równaniu różniczkowym cząstkowym ma postać inną niż sigmoidalna, dla uzyskania efektywnego narzędzia ich rozwiązywania za pomocą sieci neuronowych komórkowych, konieczna jest modyfikacja postaci elementu procesorowego. Jedną z najczęściej rozważanych w dotychczasowych pracach postaci tego elementu jest tzw. dioda Chua (ang. *Chua's diode*) [13], będąca nieliniowym elementem analogowym, opisanym za pomocą układu trzech równań różniczkowych, z których jedno jest nieliniowe (o postaci nieliniowości zbliżonej do wielomianu trzeciego stopnia). Wykorzystanie sieci złożonych z diod Chua do modelowania równań różniczkowych cząstkowych było przedmiotem intensywnych prac [3] [6] [14] [15] [16]. Dioda Chua jest stosunkowo złożonym elementem układowym i jak wcześniej wspomniano, realizacje półprzewodnikowych sieci o większych rozmiarach, z komórkami będącymi diodami Chua nie są w chwili obecnej rozważane. Biorąc pod uwagę aspekty fizycznej realizowalności, celowe wydaje się rozważenie czy dokonanie niewielkich modyfikacji podstawowego modelu komórki pozwoli na poszerzenie zakresu możliwych do modelowania w sieci równań o inne kategorie.

), $I = -0.3$

(30)

Rozważmy równanie różniczkowe cząstkowe nieliniowe:

$$\frac{\partial v(x, y, t)}{\partial t} = F(v) + k \nabla^2 v \quad (31)$$

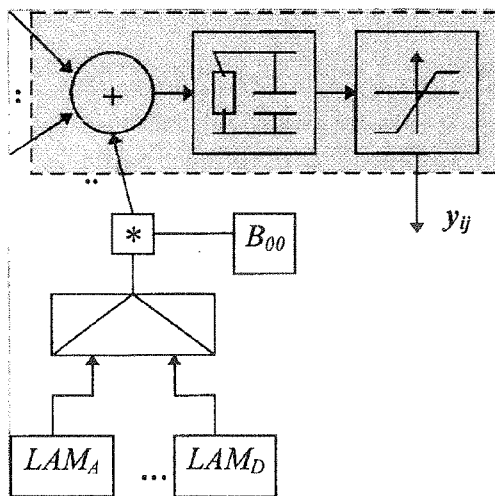
i załóżmy że występująca w nim funkcja nieliniowa $F(\cdot)$ jest wielomianem stopnia drugiego:

$$F(v) = a + b \cdot v + c \cdot v^2. \quad (32)$$

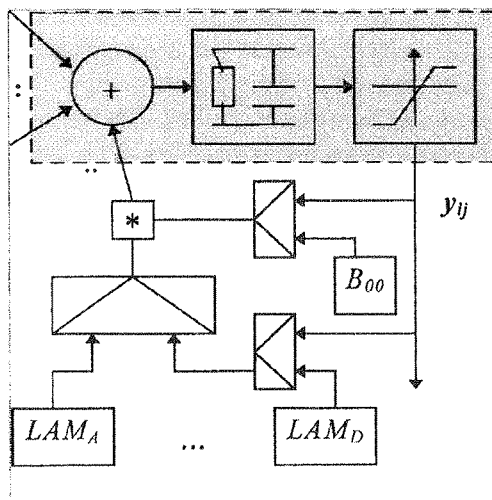
Przykładem równania o takiej strukturze jest zaproponowane we wczesnym okresie badań nad genetyką równanie Fishera, określające wzrost częstości występowania genów warunkujących korzystniejsze z punktu widzenia organizmu cechy w odniesieniu do całej populacji genów [3]. Odpowiednie wartości parametrów funkcji $F(\cdot)$, dla których równanie (31) staje się równaniem Fishera to $a=0$, $b=1$ i $c=-1$. Modyfikacja struktury elementu procesorowego USNK (Rys. 5a), niezbędna dla układowej realizacji tego równania może zostać w prosty sposób dokonana poprzez zapewnienie możliwości sterowania komórki jej sygnałem wyjściowym i jednocześnie, sterowania tym sygnałem odpowiadającej mu wagi (Rys. 5b). Konieczna modyfikacja układowa polega na umieszczeniu w komórce dodatkowo dwóch, dwuwęściowych multipleksów analogowych (kluczy analogowych, o prostej strukturze i małej powierzchni).

USNK

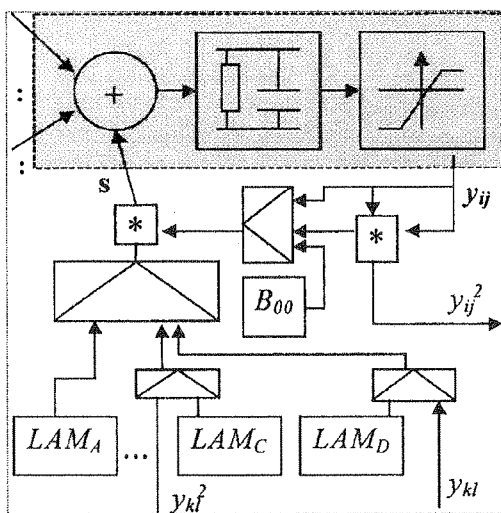
(a)



(b)



(c)



Rys. 5. Fragment struktury komórki USNK (a) oraz proponowany sposób jej modyfikacji dla realizacji funkcji nieliniowej drugiego (b) i trzeciego (c) stopnia

Dla uz
wości wiel
kolejnych
w sieci uk
najprosts
pattern fo
o postaci

Modyf
nieliniowy
parzystej
nej sieci z
prezentacj
dzone zos
(33). Celer
dla który
tworzenie
na podsta
 $c=8$ i $e=$
 α i γ nas
w drodze
symulacji
parametr
wyjściowy
przestrze
numerycz
powiadaj
Wreszcie
Podobnie
maska st
nych roz
wzorów.

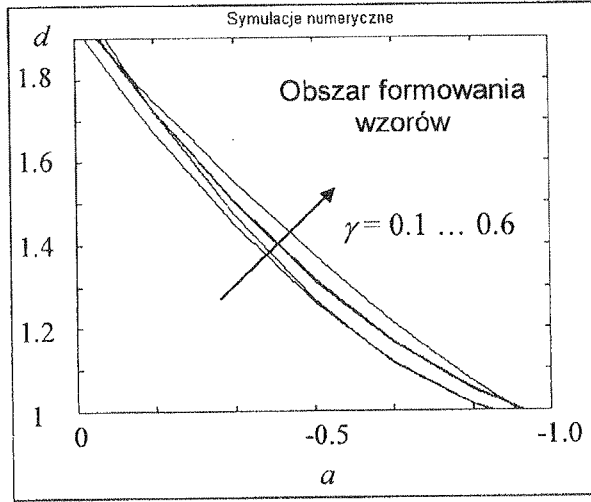
Kom
zastosow
zjawisk i
równania
dynamiki
cję impuls

Dla uzyskania możliwości modelowania równań, w których występują nieliniowości wielomianowe wyższego rzędu, konieczne jest wprowadzenie do struktury kolejnych elementów układowych. Rozważmy dla przykładu problem realizacji w sieci układu równań zaproponowanych przez Schnakenberga [11], stanowiących najprostszy model opisujący zjawisko samorzutnego formowania się wzorów (ang. *pattern formation*) w organizmach żywych (skóra zwierząt, skrzydła motyli itp.) o postaci [11]:

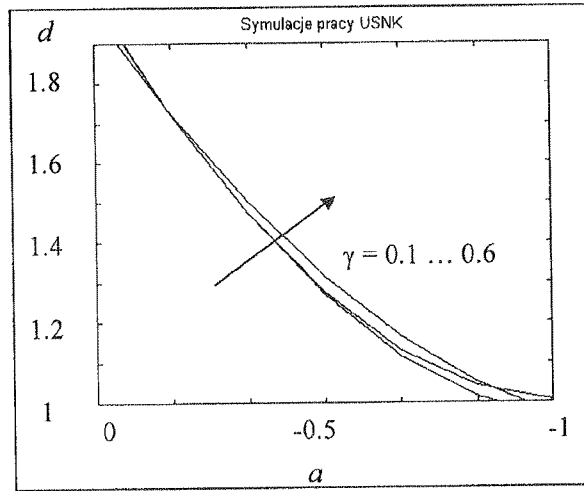
$$\begin{cases} \frac{\partial v}{\partial t} = \gamma(a - b \cdot v + c \cdot v^2 w) + \nabla^2 v \\ \frac{\partial w}{\partial t} = \gamma(d - e \cdot v^2 w) + D \cdot \nabla^2 w. \end{cases} \quad (33)$$

Modyfikacja struktury komórki wymagana dla realizacji w sieci układu równań nieliniowych (33), została przedstawiona na Rys. 5c (przy czym dla komórek podsieci parzystej $s = y_{ij}^2 y_{kl}$, zaś dla komórek podsieci nieparzystej $s = y_{kl}^2 y_{ij}$). Dla hipotetycznej sieci złożonej z przedstawionych elementów (przy założeniu niedokładnej reprezentacji sygnałów i niedokładnego wykonania parametrów układu) przeprowadzone zostały symulacje mające na celu modelowanie zjawisk opisanych równaniami (33). Celem analizy było określenie zakresów zmienności parametrów równania (33), dla których w ośrodku opisywanym przez te równania następuje samorzutne tworzenie się wzorów, a następnie porównanie wyników z wartościami określonymi na podstawie symulacji numerycznych. Dla ustalonych wartości parametrów $b = -2$, $c = 8$ i $e = -8$ równania (33) określono dla jakich zakresów zmienności parametrów a , α i γ następuje samorzutne tworzenie się wzorów, przy czym wyniki uzyskane w drodze obliczeń numerycznych zostały przedstawione na Rys. 6a, zaś wyniki symulacji pracy USNK modelującej układ równań (33) dla zmieniających się parametrów — na Rys. 6b. Na Rys. 6c przedstawiono wykres wariancji sygnałów wyjściowych komórek sieci występujący w stanie ustalonym, obrazujący obszary przestrzeni $a-d$ dla których system jest niestabilny. Różnica wyników symulacji numerycznych i symulacji działania sieci, reprezentująca błąd modelowania odpowiadający przypadkowi przedstawionym na Rys. 6c, została pokazana na Rys. 6d. Wreszcie Rys. 6e zawiera przykładowy wzór samorzutnie generowany w układzie. Podobnie jak to miało miejsce w przypadku modelowania równania Laplace'a, maska stanów niezmiennych (FSM) może być wykorzystana do określenia dowolnych rozkładów warunków brzegowych zadawanych dla procesu formowania się wzorów.

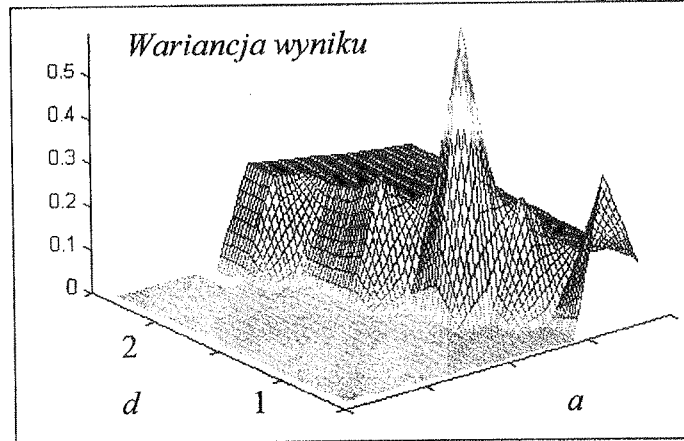
Komórki o strukturze przedstawionej na Rys. 5b, c mogą być z powodzeniem zastosowane do analizy równań stanowiących modele szeregu innych ważnych zjawisk i procesów. Formy podobne do przedstawionej zależnością (33) mają np. tzw. równania Brusselatora [3], stanowiące jedną z podstawowych zależności termodynamiki nieliniowej, czy też równania Fitz-Hugh-Nagumo [3], modelujące propagację impulsów elektrycznych w neuronach.



(a)



(b)



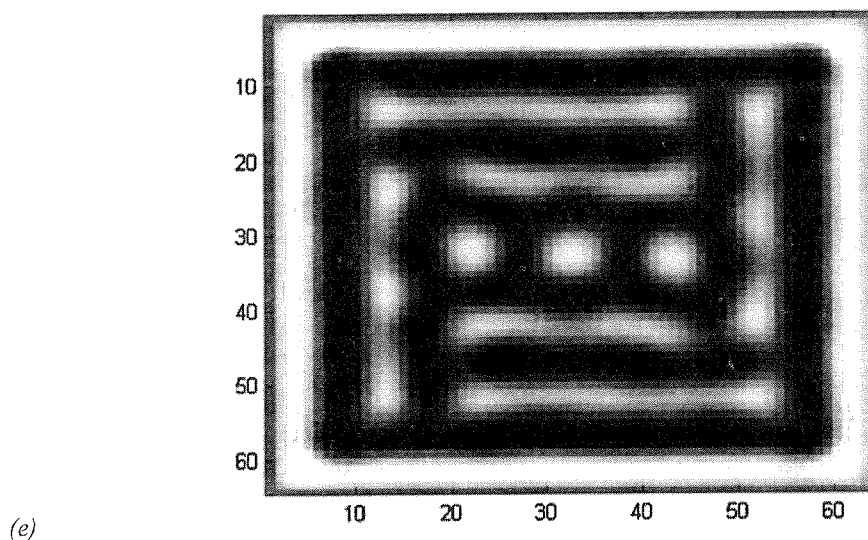
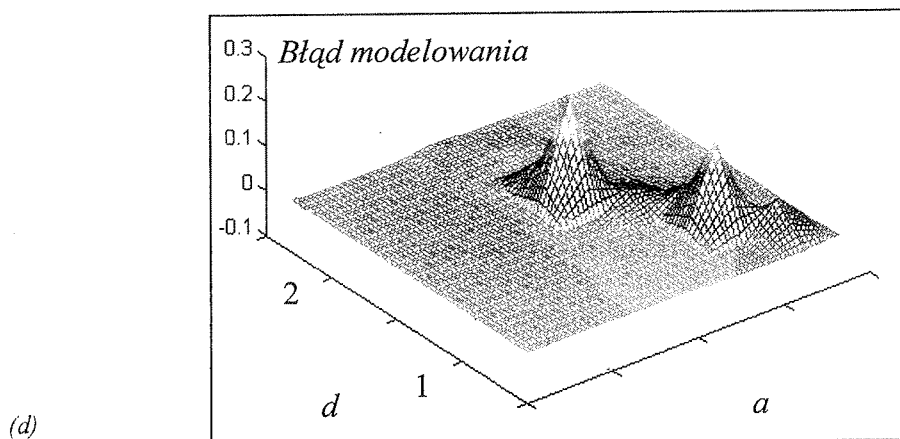
(c)

(d)

(e)

Rys. 6. M
w zapropo
generacji c
trójw

Pods
salne sie
wybrany
sieci dla



Rys. 6. Modelowanie procesów formowania się wzorów przy użyciu USNK o zmodyfikowanych w zaproponowany sposób komórkach: zakresy parametrów wyznaczających granice obszaru samorzutnej generacji określone analitycznie (a) i w drodze symulacji (b), dodatkowo przedstawione w postaci trójwymiarowej (c), (d) błąd modelowania, (e) przykładowy wzór wygenerowany w systemie

PODSUMOWANIE

Podstawowym celem przedstawionego artykułu była próba wykazania, że uniwersalne sieci neuronowe komórkowe mogą stanowić użyteczne narzędzie modelowania wybranych kategorii równań różniczkowych cząstkowych. Zasadniczą zaletą użycia sieci dla realizacji tego zadania jest potencjalna szybkość uzyskiwania rozwiązania,

zdecydowanie przewyższająca możliwości oferowane przez symulacje numeryczne, dokonywane w oparciu o najszybsze istniejące obecnie komputery. Jak zostało pokazane na wybranych przykładach, błędy modelowania nie wpływają na jakościowe zniekształcenie charakteru wyników, przez co użycie sieci do zgrubnej analizy równania wydaje się być uzasadnione.

Zaproponowana w artykule metoda modelowania układu równań różniczkowych cząstkowych może być jednocześnie traktowana jako metoda implementacji modelu sieci neuronowej komórkowej dwuwarstwowej w strukturze USNK. Oznacza to możliwość fizycznego wykonania wielu operacji opracowanych dotychczas dla sieci komórkowych dwuwarstwowych (np. wyznaczania transformaty Radona i innych zastosowań przedstawionych w [17]).

Naturalnym rozwinięciem możliwych do modelowania w sieciach neuronowych komórkowych kategorii równań różniczkowych cząstkowych są równania paraboliczne uzupełnione pochodnymi cząstkowymi rzędu pierwszego względem zmiennej przestrzennej, a dla implementacji takich równań wystarczy odpowiednio zmodyfikować wartości elementów operatora sprzężenia zwrotnego sieci (macierzy A).

BIBLIOGRAFIA

1. L.O. Chua, L. Yang: *Cellular Neural Networks: Theory and Applications*. IEEE Trans. CaS, Oct. 1988, vol. 35, pp. 1257–1290
2. T. Kacprzak, K. Ślot: *Sieci Neuronowe Komórkowe*. PWN, 1995
3. L.O. Chua, M. Hasler, G. Moschytz, Neirynck: *Autonomous Cellular Neural Networks: A Unified Paradigm for Pattern Formation and Active Wave Propagation*. IEEE Trans. CaS, Oct. 1995, vol. 42, No. 10, pp. 559–577
4. P.A. Munuzuri, V. Perez-Munuzuri, M. Gomes-Gesteira: *Spatiotemporal Patterns in Discretely-Coupled Arrays of Nonlinear Circuits: A Review*. Int. Journ. On Bifurcation and Chaos, Feb. 1995, vol. 5, No. 1
5. P. Arena, S. Bagli, L. Fortuna, G. Manganaro: *Complexity in a Two-Layer CNN*. 1996, Proc. of 4-th Int. Conf. on CNNA, pp. 127–132, Sevilla
6. T. Roska, L.O. Chua, D. Wolf, T. Kozek, R. Tetzlaff, F. Puffer: *Simulating Nonlinear Waves and Partial Differential Equations via CNN*. IEEE Trans. on CaS, 1995, vol. 42, No. 10, pp. 807–815
7. R. Dominguez-Castro, S. Espejo, A. Rodriguez-Vazquez, R. Carmona, P. Foldesy, A. Zaraby, P. Szolgay, T. Sziranyi and T. Roska: *A 0.8 μ m CMOS Programmable Mixed-Signal Focal-Plane Array Processor with On-Chip Binary Imaging and Instructions Storage*. IEEE Int. Journal of Solid State Circuits, Jul. 1997
8. T. Roska, L.O. Chua: *The CNN Universal Machine: An Analogic Array Computer*. IEEE Trans. on CaS, II, March 1993, vol. 40, pp. 163–173
9. K.R. Crounse, L.O. Chua: *Methods for Image Processing and Pattern Formation in Cellular Neural Networks: A Tutorial*, IEEE Trans. on CaS, 1995, vol. 42, No. 10, pp. 583–601
10. H. Meinhardt: *Biological Pattern Formation as a Complex Dynamic Phenomenon*. Int. Journal of Bifurcation and Chaos, 1997, vol. 7, No. 1, pp. 1–26
11. J.D. Murray: *Mathematical Biology*, Springer-Verlag, New York 1989
12. 15 W., Towards Nanoelectronics: Possible CNN Implementation using Nanoelectronic Devices. 1998, Proc. of 4-th Int. Conf. on CNN, London, pp. 20–32
13. L.O. Chua: *The Genesis of Chua's Circuit*. Int. Journ. of Electronics and Comm. (AEU), 1992, vol. 46, No. 4, Hirzel-Verlag

14. M. Ogor: *Spatio-Temporal*, vol. 42, No. 10
15. Materiały konferencyjne
16. Materiały konferencyjne
17. L.O. Chua: *Universal Machine*, UC

MODELLING

The main
Neural Network
Reaction-Diffusion
way of modelling
computer-based
implementation
linear Reaction-Diffusion
Machine. An
discussed along
discussed con-
allows for sol-
The method i-
not require an-
The proposed
Networks int-
The last
CNN UM. A
modelling eq-
been obtaine-
used in all sim-
exceed few m-
found to be c-
CNN UM c-
Reaction-Di-

14. M. Ogorzałek, Z. Galias, A. Dąbrowski, R. Dąbrowski: *Chaotic Waves and Spatio-Temporal Patterns in Large Arrays of Doubly-Coupled Chua Circuit*. IEEE Trans. on CaS 1995, vol. 42, No. 10, pp. 706–714
15. Materiały konferencji IEEE 4-th Int. Workshop CNNA-96, Sevilla, 1996
16. Materiały konferencji IEEE 5-th Int. Workshop CNNA-98, London, 1998
17. L.O. Chua, B. Shi: *Multiple Layer Cellular Neural Networks: A Tutorial*. Dec. 1990, UC Berkeley Memo, UCB/ERL/M90/113

K. SŁOT

MODELLING REACTION-DIFFUSION PARTIAL DIFFERENTIAL EQUATION USING THE CELLULAR NEURAL NETWORK UNIVERSAL MACHINE

Summary

The main objective of the following paper is to show that physical implementations of the Cellular Neural Network Universal Machine (CNN UM) paradigm can appear a useful tool for modelling Reaction-Diffusion Partial Differential Equations (PDEs). The main advantage offered by the proposed way of modelling is a processing speed (network's inherent parallelism), which outperforms capabilities of computer-based approaches while its main drawback are inaccurate computations (typical for analog implementations). Three main issues were presented in the paper. The former one concerns modeling of linear Reaction-Diffusion Partial Differential Equations using Cellular Neural Network Universal Machine. An appropriate procedure which is required for implementing equations of this type has been discussed along with its application for Laplace equation solving. The second problem which has been discussed concerns implementation of systems of equations in the considered structure. A method which allows for solving systems of Reaction-Diffusion PDEs within a framework of CNN UM was proposed. The method is based on appropriate organization of data which are to be handled in a network and does not require any modifications, neither of CNN UM architecture, nor of its processing element's structure. The proposed method can be considered also as a means for implementing two-layer Cellular Neural Networks into a CNN UM paradigm framework.

The last issue described in the paper concerns modelling nonlinear Reaction-Diffusion PDEs using CNN UM. Appropriate, minor modifications of CNN UM processing element structure, which allow for modelling equations with low-order polynomial functions were proposed and processing results, which had been obtained for such a structure, were shown. Parameters of an actual network VLSI realization were used in all simulations presented in the paper. For all examples an estimated, total processing time did not exceed few milliseconds. On the other hand, processing was not accurate, however processing errors were found to be of only limited, quantitative nature. As a conclusion, it seems that physical implementations of CNN UM could be used as a tool for an ultra-fast, preliminary analysis of systems described by Reaction-Diffusion PDEs.

M

W
nych op
z zanik
probabi
cyjnego

Sl
analogo

Do zasa
należy zape
ści spełnien
kanałów cy
cyfrowe są
analogowe
Z uwagi n
transmisji,
należy z za
ministyczn

Podcza
bardzo prz
potrzeb sa
kanałów
kanału tra
tyczne rep
 $\underline{Y}(t)$ — sy
addytywn
wejściowy
oznaczają

Modelowanie symulacyjne kanałów binarnych

STANISŁAW LINDNER

Katedra Radioelektroniki Morskiej, Wyższa Szkoła Morska w Gdyni

Otrzymano 1999.01.11

Autoryzowano 1999.05.06

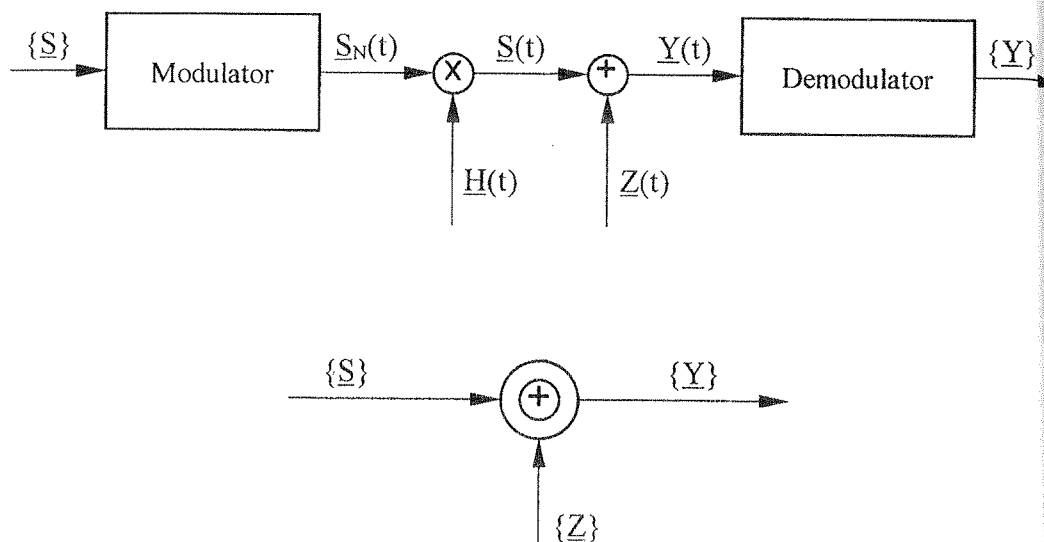
W artykule przedstawiono związki między wielostanowymi modelami kanałów binarnych opisywanymi za pośrednictwem łańcuchów Markowa a kanałami analogowymi z zanikami sygnału użytecznego i szumem gaussowskim. Wykazano, że na podstawie probabilistycznego opisu kanału analogowego można wyznaczyć parametry modelu symulacyjnego kanału binarnego. Przedstawiono przykładowe wyniki obliczeń.

Słowa kluczowe: systemy transmisji, cyfrowe systemy radiokomunikacyjne, kanały analogowe, kanały binarne, grafy modeli kanałów binarnych.

1. WPROWADZENIE

Do zasadniczych wymagań stawianych cyfrowym systemom transmisji informacji należy zapewnienie wysokiej wierności i dużej szybkości przesyłania danych. Możliwości spełnienia tych wymagań zależą od właściwości oraz sposobów wykorzystywania kanałów cyfrowych dostępnych do transmisji. W systemach transmisji danych kanały cyfrowe są na ogół tworzone w oparciu o kanały analogowe, przy czym jako kanały analogowe są wykorzystywane zarówno kanały przewodowe jak i kanały radiowe. Z uwagi na występowanie losowych zniekształceń i zakłóceń sygnałów podczas transmisji, zarówno kanały analogowe jak i tworzone na ich podstawie kanały cyfrowe należy z zasady traktować jako przetworniki dokonujące nieodwracalnych i niedeterministycznych przekształceń sygnałów nadawanych w sygnały odbierane.

Podczas analizy i modelowania cyfrowych systemów radiokomunikacyjnych bardzo przydatne jest korzystanie z modeli kanałów transmisyjnych. W zależności od potrzeb są wykorzystywane zarówno modele kanałów analogowych jak i modele kanałów cyfrowych. Podstawową strukturę blokową analogowego i cyfrowego kanału transmisyjnego przedstawiono na rys. 1, gdzie poszczególne procesy stochastyczne reprezentują odpowiednio: $\underline{S}_N(t)$ — sygnał nadany, $\underline{S}(t)$ — sygnał użyteczny, $\underline{Y}(t)$ — sygnał odebrany, $\underline{H}(t)$ — zniekształcenia multiplikatywne i $\underline{Z}(t)$ — zakłócenia addytywne, natomiast ciągi są reprezentowane odpowiednio przez $\{\underline{S}\}$ — ciąg wejściowy kanału binarnego, $\{\underline{Y}\}$ — ciąg wyjściowy kanału binarnego oraz ciąg $\{\underline{Z}\}$ oznaczający strumień błędów elementarnych kanału binarnego.



Rys. 1. Struktura blokowa kanału transmisyjnego

Opis właściwości statystycznych binarnego kanału cyfrowego może być sprowadzony do opisu właściwości statystycznych strumienia błędów $\{Z\}$, zależnych od cech statystycznych procesów $H(t)$ i $Z(t)$ w kanale analogowym.

2. MODEL KANAŁU ANALOGOWEGO

O postaci sygnału wejściowego odbiornika decyduje zarówno sygnał przenoszący informacje jak i zewnętrzne sygnały zakłócające. W wyniku przejścia sygnału wejściowego przez układy odbiornika, w pasmie wejściowym demodulatora otrzymuje się sygnał wąskopasmowy, o postaci którego decyduje sygnał przenoszący informacje, zewnętrzne sygnały zakłócające oraz sygnał reprezentujący zakłócenia wewnętrzne odbiornika. W dalszych rozważaniach zakłada się, że wąskopasmowy sygnał wyjściowy kanału analogowego można przedstawić następująco:

$$Y(t) = S_N(t) H(t) + Z(t) = S(t) + Z(t), \quad (1)$$

gdzie $Z(t)$ reprezentuje zewnętrzne i wewnętrzne zakłócenia addytywne, natomiast $S(t)$ jest zniekształconym multiplikatywnie sygnałem nadanym, nazywanym dalej sygnałem użytecznym. Sygnał użyteczny można interpretować jako realizację wąskopasmowego procesu stochastycznego w postaci

$$S(t) = A(t) \cos[\omega_0 t + \varphi(t)], \quad (2)$$

gdzie $A(t)$ i $\varphi(t)$ oznaczają odpowiednio obwiednię i odchyłkę fazy chwilowej sygnału stochastycznego $S(t)$. Realizacje procesów $A(t)$ i $\varphi(t)$ są funkcjami wolnozmiennymi

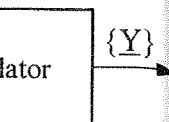
w porównaniu z częstotliwością nośną. Maksymalna szerokość pasma zakłóceń kątowych, na zmianę obwiedni przez kanał a

W celu c... cyfrowego wy... gowy, koniecz... ceń. Matema... obejmować s... wpływ zjawis... demodulacji... ciej przyjmuj... losowa A rep... Rice'a, logar... Z ekspery... cje sygnału... Nakagamię... losowa A o

gdzie $\Gamma(m)$ j

$\Omega = E(A^2)$, m... jako stosun... mocy sygna

W szcze... rozkład Na... rozkładem... co odpowia... Nakagamię... logarytmicz... aproksyma... czteropara... i Rice'a [2]



w porównaniu z $\cos \omega_0 t$, tj. praktycznie nie zmieniają wartości w czasie $T = 2\pi/\omega_0$. Maksymalna szybkość zmian tych funkcji jest odwrotnie proporcjonalna do szerokości pasma zajmowanego przez sygnał użyteczny. W odniesieniu do modulacji kątowych, najczęściej stosowanych w systemach przesyłania informacji cyfrowych, zmiany obwiedni sygnału użytecznego odzwierciedlają zmienne tłumienie wnoszone przez kanał analogowy.

W celu określenia właściwości probabilistycznych strumienia błędów kanału cyfrowego wynikających ze zniekształceń sygnałów przesyłanych przez kanał analogowy, konieczne jest przyjęcie odpowiedniego opisu matematycznego tych zniekształceń. Matematyczne modele zniekształceń addytywnych i multiplikatywnych powinny obejmować szeroką klasę kanałów analogowych i możliwie wiernie odzwierciedlać wpływ zjawisk występujących w tych kanałach na parametry sygnału poddawanego demodulacji. Przy opisie zniekształceń multiplikatywnych sygnału nadanego najczęściej przyjmuje się, że proces stochastyczny $\underline{A}(t)$ jest procesem stacjonarnym a zmienna losowa $\underline{A}$ reprezentująca chwilową wartość obwiedni przyjmuje rozkład Rayleigha, Rice'a, logarytmiczno-normalny lub Nakagamięgo.

Z eksperymentalnych badań zaników w kanałach radiowych wynika, że fluktuacje sygnału użytecznego zazwyczaj mogą być dobrze aproksymowane rozkładem Nakagamięgo [1]. Obwiednię sygnału użytecznego reprezentuje wówczas zmienna losowa $\underline{A}$ o gęstości prawdopodobieństwa

$$p(a) = \frac{2}{\Gamma(m)} \left(\frac{m}{\Omega}\right)^m a^{2m-1} \exp\left\{-\left(\frac{m}{\Omega}\right)a^2\right\} \quad \text{dla } a \geq 0, \quad (3)$$

gdzie $\Gamma(m)$ jest funkcją gamma

$$\Gamma(m) = \int_0^{\infty} x^{m-1} e^{-x} dx, \quad (4)$$

$\Omega = E(\underline{A}^2)$, natomiast $m \geq 1/2$. Parametr m rozkładu Nakagamięgo jest definiowany jako stosunek kwadratu średniej mocy sygnału użytecznego do wariancji chwilowej mocy sygnału użytecznego i może służyć do oceny głębokości zaników

$$m = \frac{[E(\underline{A}^2)]^2}{\text{Var}(\underline{A}^2)}. \quad (5)$$

W szczególności $m = \infty$ oznacza stałość obwiedni sygnału użytecznego, dla $m = 1$ rozkład Nakagamięgo jest identyczny z rozkładem Rayleigha, dla $m = 1,5$ jest rozkładem Maxwella, natomiast dla $m = 0,5$ jest jednostronnym rozkładem Gaussa, co odpowiada najgłębszym fluktuacjom obwiedni jakie można opisać rozkładem Nakagamięgo. Dla $m \gg 1$ rozkład Nakagamięgo zapewnia aproksymację rozkładu logarytmiczno-normalnego o małej wariancji. Rozkład Nakagamięgo umożliwia też aproksymację rozkładu Rice'a a ponadto zapewnia dobre przybliżenie rozkładu czteroparametrowego, którego szczególnymi postaciami są rozkłady Rayleigha i Rice'a [2]. Ponieważ rozkład Nakagamięgo zapewnia znacznie większe możliwości

opisu wyników eksperymentalnych badań zaników w kanałach radiowych niż rozkład Rayleigha, Rice'a czy też rozkład logarytmiczno-normalny, a jednocześnie posiada dogodną do obliczeń postać matematyczną, dlatego zmienna losowa o takim rozkładzie będzie wykorzystywana jako probabilistyczny model obwiedni sygnału użytecznego a tym samym jako model zniekształceń multiplikatywnych sygnału nadanego.

Losowy charakter zniekształceń multiplikatywnych sygnału nadanego oraz przypadkowe zakłócenia addytywne sprawiają, że moc chwilowa sygnału użytecznego i moc chwilowa sygnału zakłócającego są również wielkościami losowymi. Niech r oznacza ciągłą zmienną losową reprezentującą stosunek mocy chwilowej wąskopasmowego sygnału użytecznego $P_s = A^2/2$ do mocy chwilowej wąskopasmowego sygnału zakłócającego P_z w paśmie wejściowym demodulatora

$$r = \frac{P_s}{P_z}. \quad (6)$$

Podstawowym modelem zakłóceń addytywnych przyjmowanym w telekomunikacji jest szum fluktuacyjny, definiowany jako stacjonarny proces gaussowski o zerowej wartości średniej i mocy średniej $N_0 = E(P_z)$. Dla takiej postaci addytywnego sygnału zakłócającego zmienna losowa r jest dana zależnością

$$r = \frac{A^2}{2N_0}. \quad (7)$$

Gęstość prawdopodobieństwa zmiennej losowej r można wyznaczyć w oparciu o wzór (7) oraz odpowiedni model probabilistyczny obwiedni sygnału użytecznego. Jeżeli obwiednia sygnału użytecznego A jest opisana rozkładem Rayleigha, wówczas zmienna losowa r ma rozkład wykładniczy, jeżeli natomiast A ma rozkład Nakagamiego wówczas r przyjmuje postać rozkładu gamma [3]

$$p(r) = \frac{1}{\Gamma(m)} \left(\frac{m}{r_0} \right)^m \exp \left\{ - \left(\frac{m}{r_0} \right) r \right\}, \quad \text{dla } r \geq 0 \quad (8)$$

gdzie $r_0 = E(r)$ jest średnim stosunkiem sygnału do szumu.

Dla parametru $m=1$ rozkład gamma (8) jest identyczny z rozkładem wykładniczym. Z uwagi na wymienione wcześniej zalety zmiennej losowej o rozkładzie Nakagamiiego jako modelu obwiedni modelu obwiedni sygnału użytecznego, rozkład gamma dany wzorem (8) może być wykorzystywany do opisu właściwości probabilistycznych stosunku mocy chwilowej sygnału użytecznego do średniej mocy szumu na wyjściu kanału analogowego. W dalszej części rozważań zmienna losowa o rozkładzie gamma będzie wykorzystywana jako model stosunku rozważań zmienna losowa o rozkładzie gamma będzie wykorzystywana jako model stosunku sygnału do szumu przy modelowaniu cech probabilistycznych strumienia błędów kanału binarnego.

3.

Własności
prawdopodob
cznie niezale
rny i w skróci
o radiowe kan
pamięci kan
radiokomuni
międzysymb
nie kanału b
zmiennej los

3

Z powyższ
traktowaniu
wielostanow
Traktują
kanału binar
zachowuje si
2,..., R). N
przesyłania
wówczas tra
elementami

Najczęś
jednorodnej
numeru prz
i noszą naz
bezwarunko

znajdowani
nego są dla
i noszą naz
stanowego
łańcucha M
 P_{bi} ($i=1, 2$
nych z nią

Najpro
nego, stacj

3. WIELOSTANOWE MODELE KANAŁU BINARNEGO

Własności najprostszego kanału binarnego są opisane jedną liczbą, a mianowicie prawdopodobieństwem błędu elementarnego P_b . Kanał taki generuje strumień statystycznie niezależnych błędów elementarnych, jest bezpamięciowy, symetryczny, stacjonarny i w skrócie będzie oznaczany jako kanał BSS. Kanały binarne utworzone w oparciu o radiowe kanały analogowe na ogół nie są bezpamięciowe, a wynikiem występowania pamięci kanału jest grupowanie się błędów w serie. W cyfrowych systemach radiokomunikacyjnych, w których dodatkowo można pominąć zjawisko interferencji międzysymbolowych, jedną z metod opisu kanału z błędami seryjnymi jest traktowanie kanału binarnego jako symetrycznego i stacjonarnego, o ciągłym rozkładzie zmiennej losowej P_b reprezentującej prawdopodobieństwo błędu elementarnego.

(6)

3.1. MODEL KANAŁU BINARNEGO JAKO ZBIORU STANÓW BSS

Z powyższą interpretacją kanału binarnego blisko związane są modele oparte na traktowaniu rzeczywistego kanału binarnego jako zbioru kanałów BSS, opisanych wielostanowym łańcuchem Markowa.

(7)

Traktując P_b jako zmienną losową dyskretną, można wprowadzić pojęcie stanu kanału binarnego przyjmując, że kanał binarny jest w stanie B_i ($i=1, 2, \dots, R$) jeżeli zachowuje się jak kanał BSS z prawdopodobieństwem błędu elementarnego P_{bi} ($i=1, 2, \dots, R$). Niech $\underline{B}(m)$ dla $m=1, 2, \dots$ oznacza stan kanału binarnego podczas przesyłania m -tego sygnału elementarnego. Ciąg stanów kanału $\{B(1), B(2), \dots\}$ jest wówczas traktowany jako realizacja łańcucha Markowa o macierzy przejść, której elementami są prawdopodobieństwa warunkowe

$$P_{ij}(m+1) = P[\underline{B}(m+1) = B_j \mid \underline{B}(m) = B_i]. \quad (9)$$

(8)

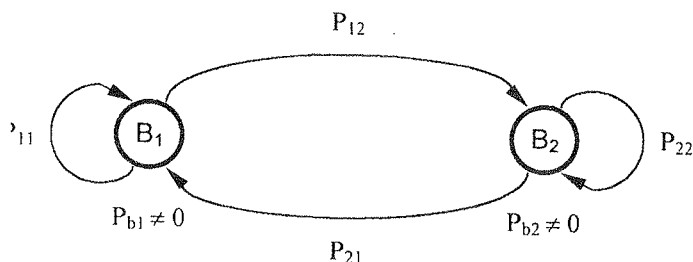
Najczęściej zakłada się, że łańcuch Markowa jest jednorodny i stacjonarny. Dla jednorodnego łańcucha Markowa prawdopodobieństwa warunkowe (9) nie zależą od numeru przesyłanego sygnału elementarnego, są jednakowe dla wszystkich $m=1, 2, \dots$ i noszą nazwę prawdopodobieństw przejść P_{ij} . Natomiast prawdopodobieństwa bezwarunkowe

$$P_i(m) = P[\underline{B}(m) = B_i] \quad (10)$$

znajdowania się kanału w stanie B_i podczas przesyłania m -tego sygnału elementarnego są dla stacjonarnego łańcucha Markowa jednakowe dla każdego $m=1, 2, \dots$ i noszą nazwę prawdopodobieństw stacjonarnych P_i . Do pełnego opisu wielostanowego modelu kanału binarnego za pomocą jednorodnego i stacjonarnego łańcucha Markowa, konieczna jest zatem znajomość wszystkich prawdopodobieństw P_{bi} ($i=1, 2, \dots, R$), znajomość prawdopodobieństw P_{ij} macierzy przejść oraz związanych z nią prawdopodobieństw stacjonarnych P_i .

Najprostszym modelem kanału binarnego opartym na wykorzystaniu jednorodnego, stacjonarnego łańcucha Markowa jest dwustanowy model Gilberta [4] i jego

modyfikacja zaproponowana przez Elliotta [2]. W modelu tym wyróżnia się dwa stany tj. B_1 i B_2 , w których odpowiednio z prawdopodobieństwami P_{b1} i P_{b2} są generowane niezależnie błędy elementarne. Graf modelu przedstawia rys. 2.



Rys. 2. Graf modelu Gilberta-Elliotta

Model ten jest opisany przez macierz przejść o następujących elementach:

$$\begin{aligned}
 P_{12} &= P[\underline{B}(m+1)=B_2 \mid \underline{B}(m)=B_1] \\
 P_{21} &= P[\underline{B}(m+1)=B_1 \mid \underline{B}(m)=B_2] \\
 P_{11} &= 1 - P_{12} \\
 P_{22} &= 1 - P_{21} .
 \end{aligned} \tag{11}$$

Prawdopodobieństwa stacjonarne P_i ($i=1, 2$) są wówczas dane wzorami

$$\begin{aligned}
 P_1 &= P[\underline{B}(m)=B_1] = \frac{P_{21}}{P_{12} + P_{21}} \\
 P_2 &= P[\underline{B}(m)=B_2] = \frac{P_{12}}{P_{12} + P_{21}}
 \end{aligned} \tag{12}$$

natomiast średnie prawdopodobieństwa błędu elementarnego P_0 jest dane zależnością

$$P_0 = P_{b1} \frac{P_{21}}{P_{12} + P_{21}} + P_{b2} \frac{P_{12}}{P_{12} + P_{21}} . \tag{13}$$

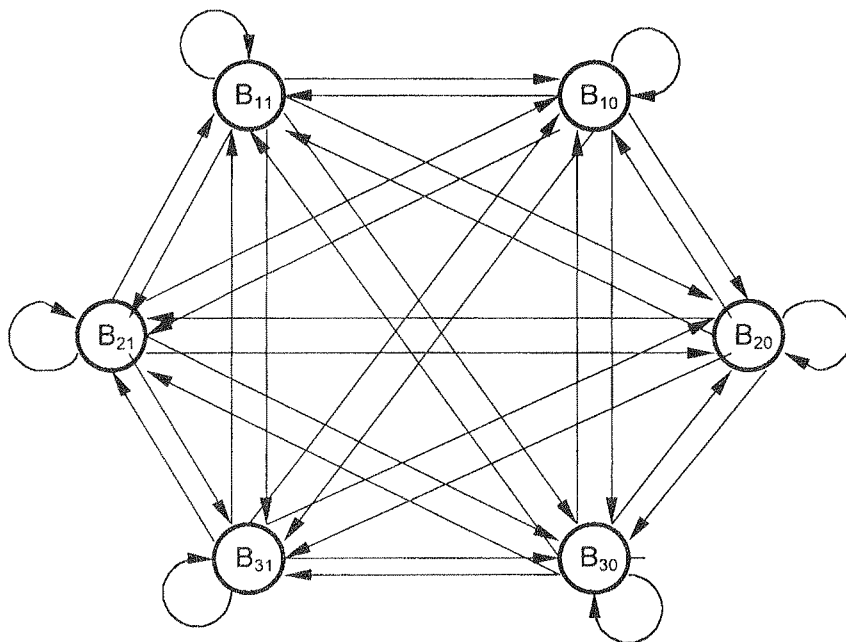
Dwustanowy model Gilberta-Elliotta jest bardzo prosty, jednak na ogół niedostatecznie dokładnie odzwierciedla właściwości probabilistyczne strumienia błędów rzeczywistych kanałów binarnych. Lepszą dokładność modelowania strumienia błędów zapewnia model wielostanowy dla $R > 2$. Jednak mimo matematycznej elegancji opisu kanału binarnego jako zbioru bezpamięciowych, stacjonarnych i symetrycznych stanów B_i , powiązanych wielostanowym łańcuchem Markowa, model taki jako model rzeczywistych kanałów binarnych tworzonych w oparciu o kanały analogowe nie znajduje praktycznego zastosowania z uwagi na niemożność wyznaczenia jego parametrów dla liczby stanów $R > 2$.

Kolejne
ze stanów
natomiast
Liczba mo
kowa rośn
stanami m
rys. 3. Ze
modelu, d
dopodobie

Mimo
 B_{i0} i B_{i1} st
Fritchma
dopodob
zbiór K
bezbłęd
zwolone
możliwo
rów mo
Łącząc v

3.2. MODEL KANAŁU BINARNEGO Z JEDNYM STANEM BŁĘDNYM I STANAMI BEZBŁĘDNymi

Kolejna modyfikacja modelu wielostanowego polega na przedstawieniu każdego ze stanów B_i w postaci dwóch stanów B_{i0} i B_{i1} , z których jeden jest stanem bezbłędnym natomiast w drugim błędy pojawiają się z prawdopodobieństwem równym jedności. Liczba możliwych stanów modelu powiązanych wielostanowym łańcuchem Markowa rośnie zatem dwukrotnie. Dozwolone są też przejścia między dowolnymi stanami modelu. Przykładowo, graf trzystanowego modelu kanału BSS przedstawia rys. 3. Ze względu na dużą liczbę prawdopodobieństw przejść między stanami tego modelu, dla zachowania prostoty rysunku zrezygnowano z zaznaczenia tych prawdopodobieństw.



Rys. 3. Graf trzystanowego modelu kanału BSS

Mimo pozornej dodatkowej komplikacji wielostanowy model ze stanami B_{i0} i B_{i1} stanowi punkt wyjścia do wielostanowego modelu Fritchmana [3]. W modelu Fritchmana wyróżnia się stany bezbłędne kanału oraz stany, w których prawdopodobieństwo błędu jest równe jedności. Kanał jest wówczas przedstawiany jako zbiór K pewnej liczby K_1 stanów błędnych i pewnej liczby $K_0 = K - K_1$ stanów bezbłędnych, powiązanych w łańcuch Markowa. Przejścia między stanami są dozwolone tylko od stanów błędnych do bezbłędnych i odwrotnie, a ponadto istnieje możliwość pozostawania w każdym ze stanów. Ponieważ $K_1 > 1$ obliczenia parametrów modelu stają się uciążliwe, najczęściej przyjmuje się $K_1 = 1$ oraz $K_0 = K - 1$. Łącząc wszystkie stany błędne B_{i1} w jeden stan błędny S_K i pozostawiając stany

B_0 jako stany bezbłędne S_i ($i = 1, 2, \dots, K-1$) otrzymuje się model, w którym kanał jest przedstawiany jako zbiór pewnej liczby stanów bezbłędnych S_i i jednego stanu błędnego S_K , powiązanych w łańcuch Markowa. Ograniczając dodatkowo przejścia między stanami tylko od stanów błędnych do bezbłędnych i odwrotnie, a ponadto dopuszczając możliwość pozostawania w każdym ze stanów, otrzymuje się model Fritchmana z jednym stanem błędnym i wieloma stanami bezbłędnymi.

Łącząc wszystkie stany błędne B_{i1} w jeden stan błędny S_K i pozostawiając stany B_i^0 jako stany bezbłędne S_i ($i = 1, 2, \dots, K-1$) otrzymuje się model, w którym kanał jest przedstawiany jako zbiór pewnej liczby stanów bezbłędnych S_i i jednego stanu błędnego S_K powiązanych w łańcuch Markowa. Ograniczając dodatkowo przejścia między stanami tylko od stanu błędnego do stanów bezbłędnych i odwrotnie, a ponadto dopuszczając możliwość pozostawania w każdym ze stanów, otrzymuje się model Fritchmana z jednym stanem błędnym i wieloma stanami bezbłędnymi.

Parametry tego modelu można wyznaczyć na podstawie rozkładu długości przerw λ ($\lambda = 0, 1, 2, \dots$) między kolejnymi błędami, otrzymanego w wyniku testowych transmisji cyfrowych. Przyjmując, że zmienna losowa λ reprezentuje losową długość przerwy między kolejnymi błędami elementarnymi, otrzymany na drodze pomiarowej rozkład estymat prawdopodobieństw $P_0(\lambda \geq \lambda)$ aproksymuje się sumą funkcji wykładniczych, przy czym liczba składników sumy określa liczbę stanów bezbłędnych S_i ($i = 1, 2, \dots, K-1$) kanału

$$P_0(\lambda \geq \lambda) = A_1 \exp(-u_1 \lambda) + A_2 \exp(-u_2 \lambda) + \dots + A_{K-1} \exp(-u_{K-1} \lambda), \quad (14)$$

gdzie $A_1 + A_2 + \dots + A_{K-1} = 1$.

W oparciu o powyższe wyrażenia wyznacza się parametry A_i oraz u_i ($i = 1, 2, \dots, K-1$). Prawdopodobieństwa przejść K -stanowego łańcucha Markowa oblicza się ze wzorów [3]

$$P_{ii} = \exp(-u_i); \quad P_{iK} = 1 - P_{ii}; \quad P_{Ki} = A_i \exp(-u_i); \quad P_{KK} = 1 - \sum_{i=1}^{K-1} P_{Ki} \quad (15)$$

Model ten jest szczególnie przydatny jako model symulacyjny strumienia błędów. Należy jednak zaznaczyć, że prawdopodobieństwa przejść (15) są wyznaczone w oparciu o estymaty parametrów A_i oraz u_i , otrzymane na podstawie pomiaru rozkładu długości przerw λ między kolejnymi błędami. Podstawowym mankamentem tego modelu jest konieczność każdorazowego prowadzenia testowych pomiarów rzeczywistego kanału cyfrowego w celu wyznaczenia parametrów modelu symulacyjnego. Wynika stąd, że model symulacyjny może wówczas generować jedynie strumień błędów elementarnych o cechach statystycznych identycznych jak w rzeczywistym kanale testowym. Powoduje to małą elastyczność modelu.

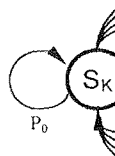
Parametry modelu z jednym stanem błędnym S_K i wieloma stanami bezbłędnymi S_i ($k = 1, 2, \dots, K-1$) można również wyznaczyć przy założeniu, że zbiór stanów kanału jest reprezentowany przez $R = K-1$ stanów B_i ($i = 1, 2, \dots, R$), w których kanał jest traktowany jako stan BSS o prawdopodobieństwie błędu elementarnego P_{bi} . Niech zmienna losowa λ_i reprezentuje długość przerwy między kolejnymi błędami w stanie B_i . Prawdopodobieństwo wystąpienia przerwy nie mniejszej od λ w stanie B_i jest równe

Traktując P

gdzie P_i jest
uwzględniaj
mie postać
Markowa m

$$P_{ii} = 1$$

gdzie $P_0 =$
modelu kan
przedstawi



Rys. 4. G

Zbiore
odpowiad
i R stanar

4. ZW

Opis
z przybliż
sygnału u
którego k
wych pur
dzenie po

kanal jest
nego stanu
wo przejścia
, a ponadto
ie się model

jąc stany B_i^0
m kanal jest
nego stanu
wo przejścia
odwrotnie,
trzyma się
dnymi.

gości przerw
u testiwych
ową długość
pomiarowej
funkcji wy-
ezbłądnych

λ), (14)

2,..., $K-1$).
e wzorów [3]

P_{Ki} (15)

nia błędów.
wyznaczane
ie pomiaru
nkamentem
pomiarów
a symulacyj-
nie strumień
teczywym

błądnymi S_i
nów kanału
h kanal jest
o P_{bi} . Niech
mi w stanie
 P_i jest równe

$$P(\underline{A}_i \geq \lambda) = (1 - P_{bi})^\lambda \quad \lambda = 0, 1, 2, \dots \quad (16)$$

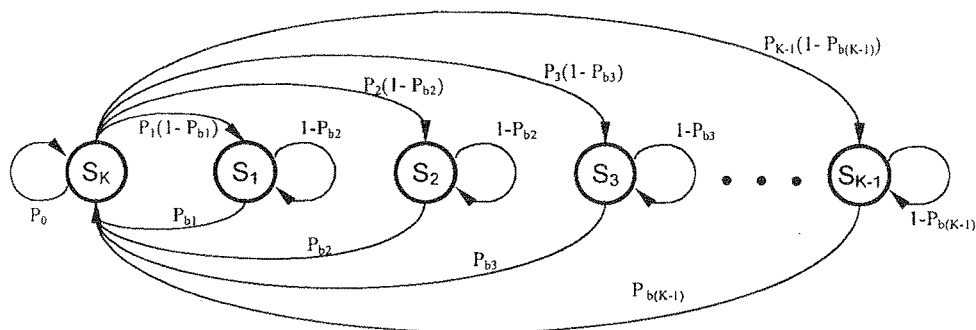
Traktując $P_0(\underline{A} > \lambda)$ jako sumę ważoną

$$P_0(\underline{A} \geq \lambda) = \sum_{i=1}^{K-1} P_i P(\underline{A}_i > \lambda) \quad \lambda = 0, 1, 2, \dots \quad (17)$$

gdzie P_i jest bezwarunkowym prawdopodobieństwem wystąpienia stanu B_i oraz uwzględniając wzór (16), po podstawieniu $(1 - P_{bi}) = \exp(-u_i)$ wyrażenie (17) przyjmie postać (14) dla $A_i = P_i$. Prawdopodobieństwa przejść (15) K -stanowego łańcucha Markowa można zatem przedstawić następująco

$$P_{ii} = 1 - P_{bi}; \quad P_{iK} = P_{bi}; \quad P_{Ki} = P_i(1 - P_{bi}); \quad P_{KK} = 1 - \sum_{i=1}^{K-1} P_i(1 - P_{bi}) = P_0, \quad (18)$$

gdzie $P_0 = \sum_{i=1}^R P_i P_{bi}$ jest średnim prawdopodobieństwem błędu elementarnego. Graf modelu kanału binarnego z jednym stanem błędnym i wieloma stanami bezbłędnymi przedstawia rysunek 4.



Rys. 4. Graf modelu kanału binarnego z jednym stanem błędnym i wieloma stanami bezbłędnymi

Zbiorowi R stanów B_i , w których kanal jest traktowany jako zbiór stanów BSS, odpowiada więc zbiór $K = R + 1$ stanów modelu kanału z jednym stanem błędnym i R stanami bezbłędnymi.

4. ZWIĄZKI WIELOSTANOWEGO MODELU KANAŁU BINARNEGO Z KANAŁEM ANALOGOWYM

Opis kanału binarnego za pomocą zbioru stanów B_i ($i = 1, 2, \dots, R$) wiąże się z przybliżeniem ciągłego procesu stochastycznego $\underline{A}(t)$ reprezentującego obwiednię sygnału użytecznego w kanale analogowym, odpowiednim procesem dyskretnym, którego każda realizacja jest przebiegiem schodkowym o nieciągłościach w losowych punktach czasu $\underline{T}_k$ ($k = 1, 2, \dots$). Powyższe przybliżenie pozwala na wprowadzenie pojęcia stanu kanału analogowego. Tak więc kanal analogowy jest w stanie

B_i ($i=1, 2, \dots, R$) jeżeli zachowuje się jak kanał bez zaników, w którym stosunek sygnału do szumu przyjmuje stałą wartość r_{oi} ($i=1, 2, \dots, R$). Przejścia między stanami mogą odbywać się tylko skokowo i każdorazowo po przejściu do określonego stanu, kanał analogowy znajduje się w nim przez losowy przedział czasu $\tau_k = T_{k+1} - T_k$.

Niech $\underline{B}(\tau_k)$ oznacza losowy stan kanału analogowego w losowym przedziale czasu τ_k , w którym obwiednia sygnału użytecznego jest traktowana jako ustalona, P_i oznacza prawdopodobieństwo wystąpienia stanu B_i kanału analogowego w przedziale czasu τ_k

$$P_i = P[\underline{B}(\tau_k) = B_i] \quad (19)$$

natomiast P_{ij} niech oznacza warunkowe prawdopodobieństwo zmiany stanu kanału analogowego, zdefiniowane jako

$$P_{ij} = \begin{cases} P[\underline{B}(\tau_{k+1}) = B_j \mid \underline{B}(\tau_k) = B_i] & \text{dla } i \neq j \\ 0 & \text{dla } i = j. \end{cases} \quad (20)$$

Przy założeniu stacjonarności kanału analogowego, zarówno prawdopodobieństwa P_i jak i P_{ij} nie zależą od indeksu przedziału czasu τ_k i są jednakowe dla wszystkich $k=1, 2, \dots$. W przypadku stacjonarnego kanału analogowego średni czas τ_{oi} znajdowania się kanału w stanie B_i , po uwzględnieniu szybkości modulacji może być wykorzystywany do oceny szybkości zaników (zaniki szybkie, wolne, bardzo wolne).

Przedstawiony opis właściwości czasowych kanału analogowego pozwala na wyznaczenie elementów macierzy przejść łańcucha Markowa, wykorzystywanego do modelowania kanału binarnego. Traktując w przybliżeniu każdą realizację zmiennej losowej τ jako całkowitą wielokrotność czasu trwania pojedynczego sygnału elementarnego, prawdopodobieństwo stacjonarne P_i można również interpretować jako prawdopodobieństwo przesłania ciągu losowej liczby sygnałów elementarnych przez kanał będący w stanie $B_i(\tau)$. Średnia długość tego ciągu, oznaczana dalej przez l_{oi} zależy od τ_{oi} i od szybkości modulacji. Tak więc każdorazowo po przejściu do stanu $B_i(\tau)$ kanał znajduje się w nim średnio przez czas τ_{oi} i odpowiada to przesłaniu średnio l_{oi} sygnałów elementarnych. Przy założeniu jednorodności łańcucha Markowa, prawdopodobieństwa zmiany stanu kanału binarnego między kolejnymi sygnałami elementarnymi można przedstawić następująco natomiast prawdopodobieństwo pozostania kanału w stanie B_i ma postać

$$P_{ii} = 1 - \frac{l}{l_{oi}} \quad i=1, 2, \dots, R. \quad (22)$$

Z powyższych wzorów wynika, że przy ustalonych właściwościach czasowych kanału analogowego, elementy macierzy przejść łańcucha Markowa zależą od szybkości modulacji. Z zależności tych wynika też, że wraz ze wzrostem średnich długości ciągów l_{oi} przesyłanych przez kanał będący w stanie B_i maleją prawdopodobieństwa przejść P_{ij} do innych stanów i rośnie prawdopodobieństwo P_{ii} pozostania w tym stanie. Tak więc dla bardzo długich ciągów l_{oi} prawdopodobieństwa $P_{ij} \rightarrow 0$ natomiast $P_{ii} \rightarrow 1$. Jeżeli właściwość ta dotyczy wszystkich stanów B_i ($i=1,$

2, ..., R), wó
sygnału uż
kanału bin
można wię
stąpienia p
 P_{bi} w tych
(26) i (29).
binarnego
stanami be

4.

Prawd
z jednym
wzorów (1
o znajomo
Sposób
zniesztak
tywanego
wartości z

gdzie erfc

Powyższe

gdzie $\Gamma(\dots)$
współczy
lacji przy

ym stosunek
ędzy stanami
onego stanu,
 $T_{k+1} - T_k$.
ziale czasu τ_k ,
a, P_i oznacza
ziale czasu τ_k

(19)

stanu kanału

(20)

opodobieństw-
a wszystkich
czas τ_{0i} znaj-
cji może być
rdzo wolne).
pozwala na
tywanego do
cję zmiennej
nału elemen-
etować jako
arnych przez
dalej przez
ści do stanu
aniu średnio
a Markowa,
ni sygnałami
odobieństwo

(22)

a czasowych
a zależą od
em średnich
ależą praw-
odobieństwo
odobieństw-
ów B_i ($i=1$,

2,..., R), wówczas odpowiada to występowaniu bardzo wolnych fluktuacji obwiedni sygnału użytecznego w kanale analogowym. Wyznaczanie parametrów modelu kanału binarnego w przypadku bardzo wolnych zaników w kanale analogowym można więc ograniczyć do znalezienia prawdopodobieństw stacjonarnych P_i wystąpienia poszczególnych stanów B_i oraz prawdopodobieństw błędu elementarnego P_{bi} w tych stanach. Prawdopodobieństwa te można wyznaczyć na podstawie wzorów (26) i (29). Przy bardzo wolnych zanikach w kanale analogowym, jako model kanału binarnego może więc być wykorzystany model z jednym stanem błędnym i wieloma stanami bezbłędnymi, którego graf przedstawiono na rys. 4.

4.1. WPŁYW ZANIKÓW W KANALE ANALOGOWYM NA PARAMETRY MODELU KANAŁU BINARNEGO

Prawdopodobieństwa przejść P_{ij} łańcucha Markowa modelu kanału binarnego z jednym stanem błędnym i wieloma stanami bezbłędnymi można wyznaczyć ze wzorów (15), wykorzystując prawdopodobieństwa P_i oraz P_{bi} wyznaczone w oparciu o znajomość rozkładu zmiennej losowej r .

Sposób określenia tych prawdopodobieństw w oparciu o probabilistyczny opis zniekształceń sygnałów przesyłanych przez kanał analogowy zależy od wykorzystywanego sposobu modulacji i demodulacji. Dla NCFSK, CFSK, DPSK i CPSK wartości zmiennych losowych r i P_b są związane następującymi zależnościami [9]:

$$P_b = \frac{1}{2} \exp(-ar) \quad \begin{cases} a=1/2 & \text{dla NCFSK} \\ a=1 & \text{dla DPSK} \end{cases} \quad (23)$$

$$P_b = \frac{1}{2} \operatorname{erfc} \sqrt{ar} \quad \begin{cases} a=1/2 & \text{dla CFSK} \\ a=1 & \text{dla CPSK} \end{cases} \quad (24)$$

$$\text{gdzie } \operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} \exp(-t^2) dt.$$

Powyższe zależności można wyrazić jednym wzorem [6]

$$P_b = \frac{\Gamma(b, ar)}{\Gamma(b)} \quad (25)$$

gdzie $\Gamma(b)$ jest funkcją gamma, $\Gamma(b, ar)$ jest niepełną funkcją gamma, natomiast współczynniki a i b , w zależności od wykorzystywanego sposobu modulacji i demodulacji przyjmują wartości przedstawione w zestawieniu

	b	
a	1/2	1
1/2	CFSK	NCFSK
1	CPSK	DPSK

Parametry wielostanowego modelu kanału binarnego takie jak: prawdopodobieństwa błędu elementarnego P_{bi} w poszczególnych stanach B_i ($i=1, 2, \dots, R$), elementy macierzy przejść P_{ij} między stanami oraz prawdopodobieństwa stacjonarne P_i , można wyznaczyć na podstawie probabilistycznego opisu kanału analogowego. Znajomość rozkładu zmiennej losowej r reprezentującej stosunek sygnału do szumu w kanale analogowym pozwala bowiem na wyznaczenie prawdopodobieństw P_i oraz P_{bi} . Przyjęcie założenia o skończonym lub przeliczalnym zbiorze stanów B_i ($i=1, 2, \dots, R$) kanału binarnego wiąże się z przybliżeniem ciągłych rozkładów zmiennych losowych r i $P_b = P_b(r)$ odpowiednimi rozkładami dyskretnymi. Dzieląc ciągły zbiór wartości zmiennej losowej r na R przedziałów i oznaczając ich krańce przez r_i oraz r_{i+1} , prawdopodobieństwo wystąpienia stanu B_i można wyrazić wzorem

$$P_i = \int_{r_i}^{r_{i+1}} p(r) dr. \quad (26)$$

Przy uczynionych założeniach, średnie prawdopodobieństwo błędu elementarnego $P_0 = E(P_b)$ oraz stosunek średniej mocy sygnału użytecznego do średniej mocy szumu $r_0 = E(r)$ mają postać

$$P_0 = \sum_{i=1}^R P_i P_{0i} = \int_0^{\infty} P_b(r) p(r) dr \quad (27)$$

$$r_0 = \sum_{i=1}^R P_i r_{0i} = \int_0^{\infty} r p(r) dr, \quad (28)$$

gdzie P_{0i} oraz r_{0i} oznaczają odpowiednio średnie prawdopodobieństwo błędu elementarnego i średni stosunek sygnału do szumu w przedziale $\langle r_i, r_{i+1} \rangle$. Składniki powyższych sum są równe

$$P_i P_{0i} = \int_{r_i}^{r_{i+1}} P_b(r) p(r) dr \quad (29)$$

$$P_i r_{0i} = \int_{r_i}^{r_{i+1}} r p(r) dr. \quad (30)$$

Uwzględniając P_i dane wzorem (26) na podstawie zależności (29) i (30) można wyznaczyć odpowiednio P_{0i} oraz r_{0i} . Ponieważ w stanie B_i kanał binarny jest traktowany jako BSS z prawdopodobieństwem błędu elementarnego P_{bi} , więc prawdopodobieństwa P_{0i} mogą być w przybliżeniu traktowane jako prawdopodobieństwa błędu elementarnego i -tego stanu BSS, czyli $P_{0i} \cong P_{bi}$. Dokładność tego przybliżenia, a tym samym dokładność aproksymacji rzeczywistego kanału binarnego przez model wielostanowy, rośnie wraz ze wzrostem liczby R stanów kanału.

5. WYNIKI OBLICZEŃ

Przykładowe obliczenia wykonano przy założeniu, że informacje są przesyłane przez kanał analogowy z bardzo wolnymi zanikami Nakagamiiego i szumem gaussow-

skim, a d
demodul

Zmie
przynaj
konieczn
liczbie s
stosunku
nierówn
 r przyjm

Uwzglęc
równe c
następuj

Przy
założeni
a średn
wartość
prawd
ści r_{0i} w

R
5

Pra
szumu
na pod
stwa pr
(18), w

podobień-
(R), elementy
cjonalne P_{bi}
analogowego.
u do szumu
ństw P_i oraz
 B_i ($i=1, 2, \dots$,
y zmiennych
ciągły zbiór
przez r_i oraz
n

(26)

u elementar-
edniej mocy

(27)

(28)

ędu elemen-
Składniki

(29)

(30)

(30) można
binarny jest
o P_{bi} , więc
prawdopodo-
bność tego
u binarnego
kanału.

przesyłane
m gaussow-

skim, a do ich przesyłania są wykorzystywane sygnały z modulacją FSK, podlegające demodulacji niekoherentnej.

Zmienna losowa r reprezentująca stosunek sygnału do szumu może przyjmować, przynajmniej teoretycznie, wartości od zera do nieskończoności. Ze względu na konieczność podziału zakresu zmienności r na R przedziałów odpowiadających liczbie stanów kanału, obliczenia wykonano dla ograniczonego zakresu zmian stosunku sygnału do szumu. Krańce tego zakresu wyznaczono na podstawie nierówności Czebyszewa [7]. Prawdopodobieństwo, z którym zmienna losowa r przyjmuje wartość w otoczeniu swojej wartości średniej ma wówczas postać

$$P\{E(r) - h[\text{Var}(r)]^{1/2} \leq r \leq E(r) + h[\text{Var}(r)]^{1/2}\} \geq 1 - \frac{1}{h^2} \quad \text{dla } h > 1. \quad (31)$$

Uwzględniając wartość średnią i wariancję zmiennej losowej o rozkładzie (8), równe odpowiednio $E(r) = r_0$ oraz $\text{Var}(r) = r_0^2/m$, nierówność (31) można wyrazić następująco

$$P\left[r_0\left(1 - \frac{h}{\sqrt{m}}\right) \leq r \leq r_0\left(1 + \frac{h}{\sqrt{m}}\right)\right] \geq 1 - \frac{1}{h^2}. \quad (32)$$

Przykładowe obliczenia prawdopodobieństw P_i oraz P_{bi} wykonano dla $h=4$ przy założeniu, że głębokość zaników w kanale analogowym jest określona przez $m=10$, a średnie prawdopodobieństwo błędu elementarnego $P_0=10^{-2}$. Wynikająca stąd wartość średniego stosunku sygnału do szumu $r_0=9.575$. W tabeli 1 zamieszczono prawdopodobieństwa P_i , prawdopodobieństwa błędu elementarnego P_{bi} oraz wartości r_{0i} w poszczególnych stanach kanału dla $R=5$.

T b l i c a 1

Parametry wielostanowego modelu kanału binarnego dla $P_0=10^{-2}$, $m=10$, $r_0=9.575$

R	i	P_i	P_{bi}	r_{0i}
5	1	$1.779 \cdot 10^{-2}$	$7.884 \cdot 10^{-2}$	3.579
	2	$4.026 \cdot 10^{-1}$	$1.784 \cdot 10^{-2}$	6.994
	3	$4.492 \cdot 10^{-1}$	$3.037 \cdot 10^{-3}$	10.537
	4	$1.159 \cdot 10^{-1}$	$4.020 \cdot 10^{-4}$	14.532
	5	$1.341 \cdot 10^{-2}$	$4.921 \cdot 10^{-5}$	18.700
		$\sum_i P_i = 0.999$	$\hat{P} = 9.999 \cdot 10^{-3}$	$\hat{r} = 9.552$

Prawdopodobieństwa te obliczono dzieląc zakres zmienności stosunku sygnału do szumu na R równych przedziałów. W tablicy tej podano też estymaty $\hat{P}_0$ i $\hat{r}_0$ obliczone na podstawie wyznaczonych parametrów modelu wielostanowego. Prawdopodobieństwa przejść P_{ij} między stanami tego modelu można wyznaczyć w oparciu o wzory (18), wykorzystując w tym celu prawdopodobieństwa P_i oraz P_{bi} ($i=1, 2, \dots, R$)

obliczane na podstawie wzorów (26) i (29). Dla $R=5$, $m=10$, $P_0=10^{-2}$ oraz pod warunkiem wykorzystywania NCFSK, macierz przejść modelu Fritchmana ma postać

$$[P]=\begin{bmatrix} 0.92116 & 0 & 0 & 0 & 0 & 0.07884 \\ 0 & 0.98216 & 0 & 0 & 0 & 0.01784 \\ 0 & 0 & 0.99696 & 0 & 0 & 0.00304 \\ 0 & 0 & 0 & 0.99960 & 0 & 0.00040 \\ 0 & 0 & 0 & 0 & 0.99995 & 0.00005 \\ 0.01639 & 0.39542 & 0.44783 & 0.11585 & 0.01341 & 0.01 \end{bmatrix} \quad (33)$$

Wykorzystując probabilistyczny opis kanału analogowego, można więc tworzyć modele symulacyjne kanału binarnego dla różnych głębokości zaników w kanale analogowym i dla różnych rodzajów modulacji i demodulacji przesyłanych sygnałów.

BIBLIOGRAFIA

1. J. R. Benjamin, A. C. Cornell: *Rachunek prawdopodobieństwa, statystyka matematyczna i teoria decyzji dla inżynierów*. WNT, Warszawa, 1977
2. E.O. Elliott: *Estimates of error rates for codes on burst noise channels*. BSTJ, vol. 42, s. 1977–1997, 1963
3. B.D. Fritchman: *A binary channel characterisation using partitioned Markov chains*. IEEE Trans. on Inform. Theory, vol. IT-13, s. 221–227, 1967
4. E.N. Gilbert: *Capacity of burst-noise channel*. BSTJ, vol. 39, s. 1253–1267, 1960.
5. D.D. Kłowski: *Pieredacza dyskretnej komunikacji po radiokanale*. Swiaz, Moskawa, 1969
6. S. Lindner: *Statystyki błędów transmisji binarnej w kanale z bardzo wolnymi zanikami Nakagami i szumem gussowskim*. Archiwum Elektrotechniki, Zeszyt 3/4, s. 957–965, 1987
7. M. Nakagami: *The m-distribution — a general formula of intensity distribution of rapid fading*. W zbiorze: *Statistical methods in radio wave propagation*, ed. W.C. Hoffman, Pergamon Press, 1960
8. R.H. McCulloch: *The binary regenerative channel*. BSTJ, vol. 47, s. 1713–1735, 1968
9. M. Schwartz, W. Bennett, S. Stein: *Communications systems and techniques*. McGraw-Hill, New York, 1966

S. LINDNER

THE SIMULATION MODELLING OF BINARY CHANNELS

Summary

In this article there are presented relations between multistate binary channel models, described by Markov's sequences and analogue channels with fadings of the transmitted signal and gaussian noise. It is shown that, on the basis of probabilistic description of analogue channel there is possible to calculate the parameters of the simulation model of binary channel. Some results of calculations are presented.

Key words: telecommunication systems, digital radiocommunication systems, analog channel, binary channels, graph diagrams of binary channel models.

Simulatio

In
delta-sig
the ma
chaos).
delta-sig
the nec
paper,
The re
and di
simplifi
WIND
The sim

Key wo
analog

Delta-S
conversion.
conversion
difficult to
— the feed
is no analy
setting the
modulators
modulator
that analo
important
There a
be describe

$^{-2}$ oraz pod
chmana ma

84
84
04
40
05

(33)

więc tworzyć
w w kanale
ch sygnałów.

matematyczna

STJ, vol. 42,

s. IEEE Trans.

kawa, 1969
i Nakagamiego

of rapid fading.
on Press, 1960
5, 1968

McGraw-Hill,

, described by
ian noise. It is
o calculate the
esented.

annel, binary

Simulation of delta-sigma analog-to-digital converters with the simulator "Amber"

KRZYSZTOF OLEJARCZYK

Instytut Systemów Elektronicznych, Politechnika Warszawska

Received 1999.01.18

Authorized 1999.04.28

In this paper experiences gathered during creating a simulation program for delta-sigma analog-to-digital converters are presented. The delta-sigma modulator, the main part of the converter, is characterized by chaotic dynamics (deterministic chaos). There is no theory that allows to foresee behaviour of the higher-order delta-sigma modulator (in general cas), so empirical studying of the modulators is the necessity. The specialized simulator, called "Amber", which is described in the paper, is available on author's WWW page (with source code written in C++). The results of exemplary simulations are depicted also, showing main advantages and disadvantages of presented program. Because the user interface is maximally simplified. The simulator can be compiled under different operating systems (UNIX, WINDOWS and DOS). The simulator can be useful in teaching $\Delta\Sigma$ converters. The simulator may be compiled.

Key words: delta-sigma modulation, noise-shaping, oversampling, decimation, digital filters, analog-to-digital conversion.

1. INTRODUCTION

Delta-Sigma modulators play an increasingly important role in analog-to-digital conversion. This technique has become quite popular for achieving high resolution of conversion [1]. Delta-Sigma modulation is conceptually simple, but the system is difficult to analyze. The nature of the modulator's structure prohibits simple analysis — the feedback loop contains nonlinearities and introduces stability problems. There is no analytical method for the verification of stability of delta-sigma modulator and setting the performance measure such as signal-to-noise ratio (SNR) for high-order modulators and for wide class of input signals [1]. Simulation of delta-sigma modulator is still necessary in order to verify a new project and determine the effects that analog nonidealities have on system performance. Because simulation is so important in design process, it should , it should be as quick as possible.

There are several methods of simulating $\Delta\Sigma$ modulators. Following methods will be descibed: simulation based on the use of typical circuit simulator (like SPICE),

simulation based on finite-difference equations, simulation based on behavioral approach and finally simulation based on table-lookup models.

The first of these methods, based on the use of circuit simulator, is slow and has accuracy limited to the order of 90 dB (i.e. value of an upper limit on the measurable SNR is on the order of 90 dB). Delta-sigma modulators are usually implemented as switched-capacitor circuits. In order to obtain accuracy of 90 dB, the simulator needs typically 100–1000 time steps per one clock cycle. On the other hand, if performance of the modulators is estimated on the basis of power spectrum, the simulation should be very long. If the spectrum has the resolution of 1000 bins in the band of interest, then the number of clock cycles is 10,000–1,000,000 and the number of time steps in 100–1000 times greater. Such a long simulation is really time consuming. Worse yet, rounding and truncation errors leads to accuracy degradation on the order of 90 dB [1].

The method of simulating based on finite-difference equations is the simplest and the quickest way of simulating. The most of $\Delta\Sigma$ modulators are implemented as switched-capacitor circuits, and modulators which are realized as continuous-time circuits can be modeled by discrete-time equivalents. The main disadvantage of this method is impossibility of simulating of dynamic errors of analog components such as op-amps. Many simulators based on finite-difference equations have been implemented [1]. Simulator MIDAS (presented on IEEE International Symposium on Circuits and Systems in May 1992, [1]) is destined for oversampled converters. Simulators SWITCAP [2] is destined for wider class of switched-capacitor circuits.

Behavioral modeling approach appears to be the best modeling technique for $\Delta\Sigma$ modulators: the simulation method is quick and can model dynamic errors. Simulations based on behavioral models on the order minutes instead of days for circuit simulator. Behavioral models are designed strictly for transient analysis, on the basis of a set time-domain equations derived for a specific circuit [1]. There are some examples of simulators based on behavioral models: simulator AVEswit developed by Trihy and Rohrer [1], and simulator TOSCA [3]. Both these computer are general-purpose switched-capacitor circuit simulators.

Table-based lookup modeling is circuit simulator. The second step is to perform the transient simulation using the table that was prepared earlier. In other words, this approach is based on extraction a table from the circuit. The method models static errors of analog components but cannot model dynamic errors. Software simulator ZSIM [4] uses the table-based lookup modeling method. The upper limit on the accuracy of this simulator is on the order of 80 dB [4].

Wolff et al. presented in [1] the results of comparison of three methods of simulating. The first-order modulator was simulated on a DECstation 3100. The $\Delta\Sigma$ modulator was run for 40,960 cycles for the finite-difference equations and behavioral methods (a program was developed that could run both methods). The CPU time of the simulations equaled 14 seconds and 1 minute, 42 seconds, respectively. The simulation based on the use of HSPICE was stopped after 34 clock cycles. The CPU time of the simulation equaled 568 seconds, so the estimate for a complete simulation is $568 \cdot 40,960 / 34 \approx 684,273 \text{ sec} \approx 190 \text{ hours}$. Main attributes of the methods are listed in Table 1.

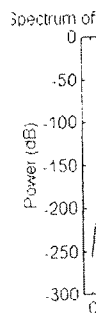
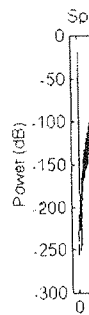
It can be
methods b
them is qu
the best fo
based on o
(which was
[5], as a co
or behavior

The sim
code). The
yields to t
and behav

HSPICE

Behavioral

Difference



F

a behavioral
low and has
measurable
implemented as
ulator needs
performance
ation should
of interest,
time steps in
. Worse yet,
of 90 dB [1].
implest and
emented as
inuous-time
tage of this
ents such as
e been im-
posium on
converges.
or circuits.
nique for $\Delta\Sigma$
mic errors.
of days for
lysis, on the
re are some
developed by
r are gene-
to perform
words, this
odels static
e simulator
mit on the
ethods of
00. The $\Delta\Sigma$
behavioral
PU time of
ively. The
The CPU
simulation
s are listed

It can be concluded that two methods of simulation stand out the others, that is the methods based on the finite-difference equations and behavioral methods. The first of them is quick and often sufficient for qualitative investigations. The second method is the best for accurate analysis of $\Delta\Sigma$ modulators. The only implementation of simulator based on one of these methods, and easily available on World Wide Web, is SWITCAP (which was mentioned above). This simulator is offered by Analogue Integration Ltd. [5], as a commercial product. No other simulator based on finite-difference equation or behavioral models could be found by the author on WWW.

The simulator presented in this paper is freely available on WWW (with the source code). The simulation method is based on finite-difference equations. This simulator yields to the above-mentioned simulators (based on the finite-difference equations and behavioral models) in functional parameters.

Table 1

Comparison of methods of modelling delta-sigma modulator

Simulator	CPU time of transient simulator	Properties
HSPICE	191 hr	Too slow for discrete-time systems; very long simulation may lead to degradation of accuracy
Behavioral	1 min. 42 sec	Can model dynamic errors
Difference Equations	14 sec	Cannot model dynamic errors

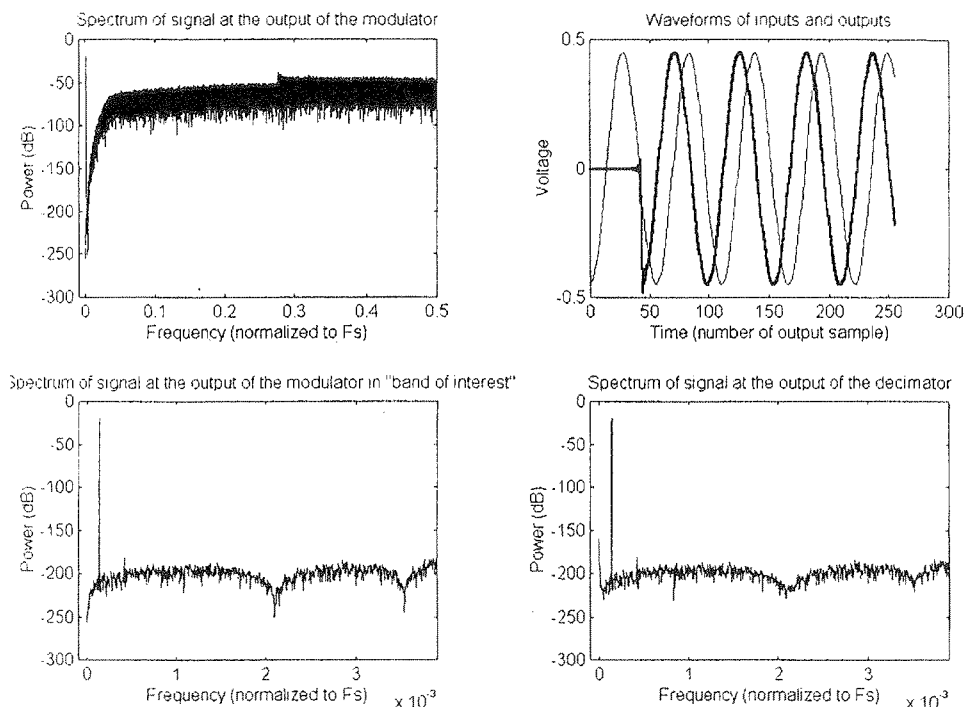


Fig. 1. Results of simulation of a fifth-order modulator with a sinusoidal input signal

The certain designer assumptions were made. The main assumption was that the simulator should work under different operating systems (UNIX, WINDOWS and possibly DOS). Two ways of implementing the simulator were taken into account: a MATLAB script and a computer program written in C++. The advantage of C++ language is the possibility of object-oriented programming. This feature permit the computer program to be easily maintained or extended. On the other hand, software simulator needs a graphical interface for visualization of results. MATLAB appears to be a better choice, from this point of view, because MATLAB scripts use the same functions for graphical visualization under UNIX as well as WINDOWS. Obviously, an implementation of a graphical interface in C++ for different operating systems is time consuming for a programmer.

An intermediate solution has been chosen: the main program has been written in object-oriented C++ language, but the results of simulations are written down into binary data files in MATLAB format. Visualization of the results is provided by MATLAB. Obviously, there are example MATLAB scripts attached to the simulator for graphical presentation of the results. Input data for the simulator is provided through text files. The input data consists of a detailed description of the $\Delta\Sigma$ converter and instructions for the simulator. There are additional MATLAB scripts attached, which are destined for designing $\Delta\Sigma$ modulators and digital filters for the multistage decimator. The most of these tools for designing elements of analog-to-digital converters have been created by Schreier and are freely available from FTP site [6].

2. DESCRIPTION OF PRESENTED SIMULATOR

Various tasks are set before simulators of delta-sigma analog-to-digital converters, as follows:

- Testing stability of the modulator (and testing how imperfections of analog devices impact on stability of the modulator).
- Estimation of modulator parameters.
- Determination of maximum and minimum values of signals at the output of the integrators inside the modulator and at the outputs of the stages inside the decimator.
- Possibly of the viewing of internal waveforms in modulator and power spectra of signals.

There are many definitions of stability of dynamical system such as $\Delta\Sigma$ modulator [1], [7], [8]. The instability consists in that signals at the outputs of the integrators oscillate with a rapidly increasing amplitude. Modulator produces alternating long strings of 1's and 0's (in single-bit quantizer case). Signal-to-noise ratio decreases dramatically, because quantization noise is bounded with wide limits. For the use of this paper the stability will be defined as follows: a delta-sigma modulator is said to be stable for bounded inputs and a given initial state of voltages at the outputs of integrators within the modulator (i.e. the initial state of modulator) are bounded with a certain arbitrary limit. Stability is an absolute necessity if a delta-sigma modulator is

input
signal
generator

to be usable
method of
in the strict
in order to
modulator
to this que
needn't be

When s
is usually
maximal v
oversampl
[9], may be
spectrum
power den
use of DF
applied (le
period in
quantizati
spectrum
for low fre

where N d
spectrum

was that the
WINDOWS and
into account:
advantage of
This feature
e other hand,
s. MATLAB
AB scripts use
WINDOWS.
For different

en written in
n down into
provided by
he simulator
is provided
Σ converter
ts attached,
e multistage
og-to-digital
FTP site [6].

igital conver-

s of analog

output of the
s inside the

power spectra

C modulator

integrators

inating long

o decreases

or the use of

is said to be

outputs of

ounded with

modulator is

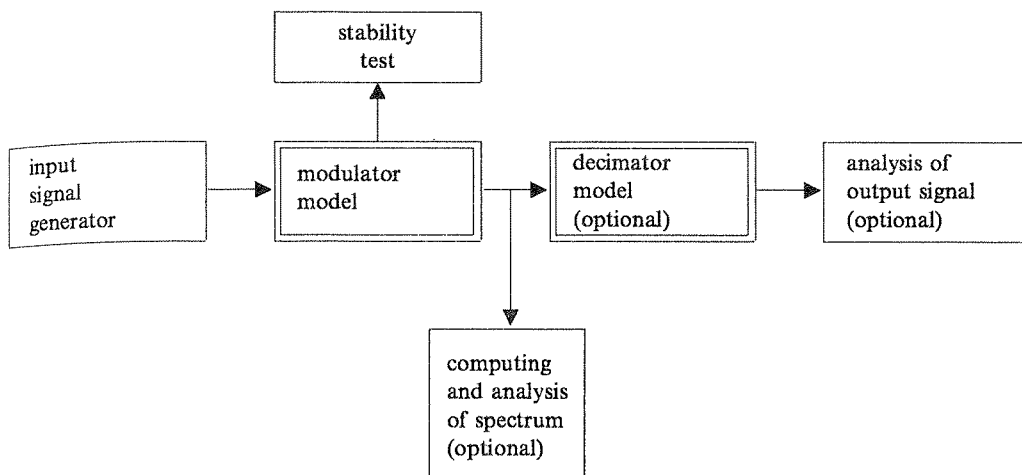


Fig. 2. General model of simulated circuit

to be usable. In the case of higher-order modulators there is no strict analytical method of proving the stability. Simulation can not prove that $\Delta\Sigma$ modulator is stable in the strict sense either. Testing of modulator stability necessitates long simulations, in order to be more credible. The question is what it means 'long simulation'? The modulator may 'explode' after millions of clock cycles [1], so there is no good answer to this question. Because stability tests consider the modulator only, the decimator needn't be simulated and power spectra needn't be calculated.

When stability of modulator is ascertained, the estimation of modulator parameters is usually wanted. The key parameter of modulator which depicts its quality is the maximal value of signal-to-noise ratio at the output of modulator, for a given oversampling ratio. Signal-to-noise ratio or its equivalent, the number of effective bits [9], may be estimated in two ways. The first way is based on the analysis of the power spectrum (obviously, the term 'power spectrum' means the well-known form of the power density spectrum estimate, called periodogram [11]; this estimate is computed by use of DFT). In this method sinusoidal input signal with specific frequency has to be applied (length of the vector of samples should be an integral multiple of the sinusoid period in order to obtain a more readable spectrum). Due to aperiodicity of the quantization noise, the window function is used in the method of estimating the power spectrum [11]. The goal of use of the window is to estimate the spectrum more accurately for low frequencies. The modified Hanning window function of the form (1) is applied.

$$w(n) = \frac{1}{2} \left(1 - \cos \frac{2\pi n}{N} \right) \quad (1)$$

where N denotes the length of the vector of samples, and $n = 0, 1, \dots, N-1$. The power spectrum of a signal $x(n)$ is given by formula (2).

$$P(n) = \left| \frac{1}{N} \sum_{k=0}^{N-1} x(k) \cdot w(k) \cdot \exp \left(-\frac{j2\pi nk}{N} \right) \right|^2 \cdot C \quad (2)$$

Let $x(n)$ be a sinusoid with frequency L/N , where L is an integer number (then N is an integral multiple of the sinusoid period). Scaling factor C equals $8/3$. Now, we obtain equation (3),

$$P(n) = \left| \frac{1}{N} \sum_{k=0}^{N-1} A \cdot \cos\left(2\pi \frac{L}{N} k\right) \cdot w(k) \sum_{k=0}^{N-1} e^{-j2\pi nk} \right|^2 \cdot C = \begin{cases} \left(\frac{A}{4}\right)^2 \cdot C & n=L, n=N-L \\ \left(\frac{A}{8}\right)^2 \cdot C & n=L \pm 1, n=N-L \pm 1 \\ 0 & \text{in other cases} \end{cases} \quad (3)$$

The power spectrum of the window function (1) consists of three bins, and the spectrum $P(n)$ is the convolution of the 'pure' spectrum of the sinusoid with the window function spectrum. It can be verified that $\sum_{k=0}^{N-1} P(n) = \frac{A^2}{2}$.

The analysis of the spectrum consists in calculating of signal power, and the noise power included in the band of interest. The signal-to-noise ratio can be estimated on the basis of shorter simulations then those used for stability testing. In this case, simulations have length on the order 10,000–100,000 clock cycles, depending on the oversampling ratio.

Figure 1 shows results of an exemplary of ADC with fifth-order modulator and oversampling ratio 128. First 65,536 clock cycles were ignored in order to allow FIR decimating filters to reach their steady states. The power spectrum of signal at the modulator output was computed from the vector of 262,144 samples. It yields 2,048-bins of in-band spectrum resolution. In practice, only FFT algorithm is adequately quick in the case of so long vectors of samples. The disadvantage of FFT algorithm is that all the samples of spectrum are established, while only the initial part of the spectrum (i.e. the band of interest) is needed. The Goertzel algorithm [11], which can compute only a part of the spectrum was tested also, but it appeared to be much slower than FFT algorithm for long vectors of samples.

The other way of estimating SNR is based on high-quality decimation filtering and then approximation in least-squares sense [9]. The disadvantage of the method is the need of using high-order digital filters (FIR or IIR) especially in multirate (cascaded) form [1], [10]. The problem is now, that real digital filters are not ideal. The main objective of decimating filter is to attenuate the quantization noise. Oversampling ratio is usually large, so the 'transition band' is comparable with the passband. There is a hyperbolic dependence between the transition band and complexity of decimating filter (i.e. the order of FIR filter) [10]. The concept of use of universal high-performance filter (designed for specific value of oversampling ratio) would lead to needless time-consuming calculations. The other disadvantage of this method of estimating of SNR is that many initial cycles are required to let filters within decimator reach their steady states, before the signal will be analyzed. A merit of this method is a possibility of employing wider class of input signals. Particularly, any sinusoid with arbitrary frequency can be used.

In the :
148 dB (24
frequency
and offset
correspon
case a high
SNR of th
stages (in
greater tha

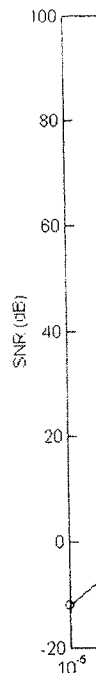


Fig. 4. A s

then N is an
w, we obtain

$$-L \pm 1 \quad (3)$$

ns, and the
id with the

and the noise
estimated on
in this case,
ding on the

ulator and
allow FIR
gnal at the
s. It yields
gorithm is
age of FFT
initial part
rithm [11],
eared to be

on filtering
e method is
a multirate
t ideal. The
Oversamp-
passband.
xplexity of
f universal
would lead
method of
ers within
erit of this
ularly, any

In the above-mentioned example the both methods were used with final result 148 dB (24.3 effective bits). The input sinusoid had amplitude 0.45, no offset, and frequency equals 1). Amplification error of the simulated converter equaled 0.007%, and offset error was $-2e-8$. The conversion time equaled 6915 clock cycles (and corresponds to about 54 periods of the signal at the output of the decimator). In this case a high quality decimating filter was used in order to obtain accurate estimate of SNR of the signal at the output of modulator. The decimator consisted of seven stages (in each stage the decimating rate equaled two; FIR filters had attenuation greater than 150 dB in stopband).

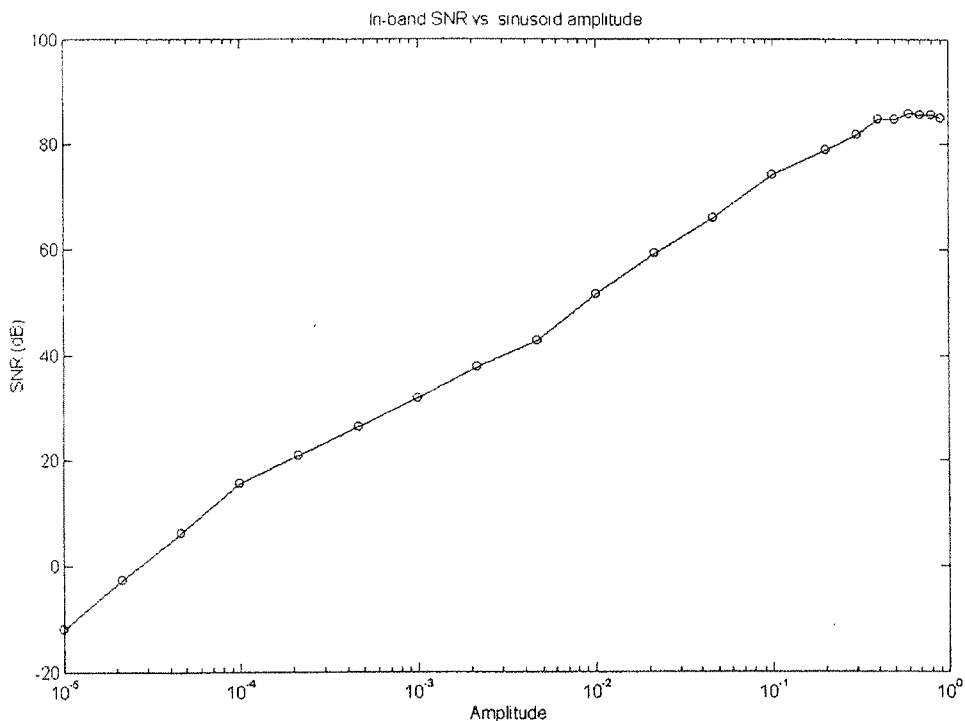


Fig. 3. In-band SNR vs. amplitude of input sinusoid

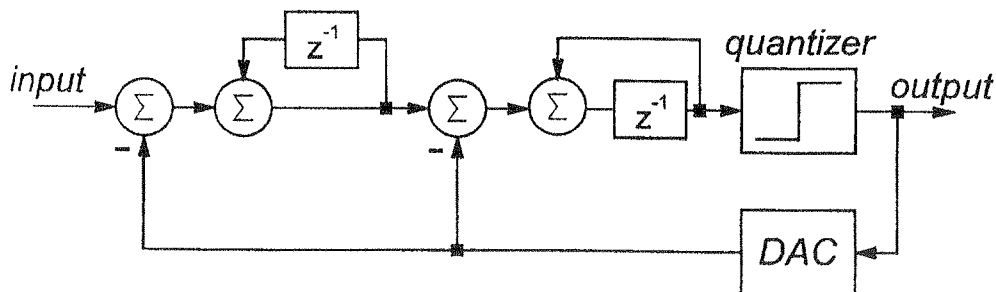


Fig. 4. A standard second-order delta-sigma modulator. The quantizer as well as the DAC have 1-bit resolution

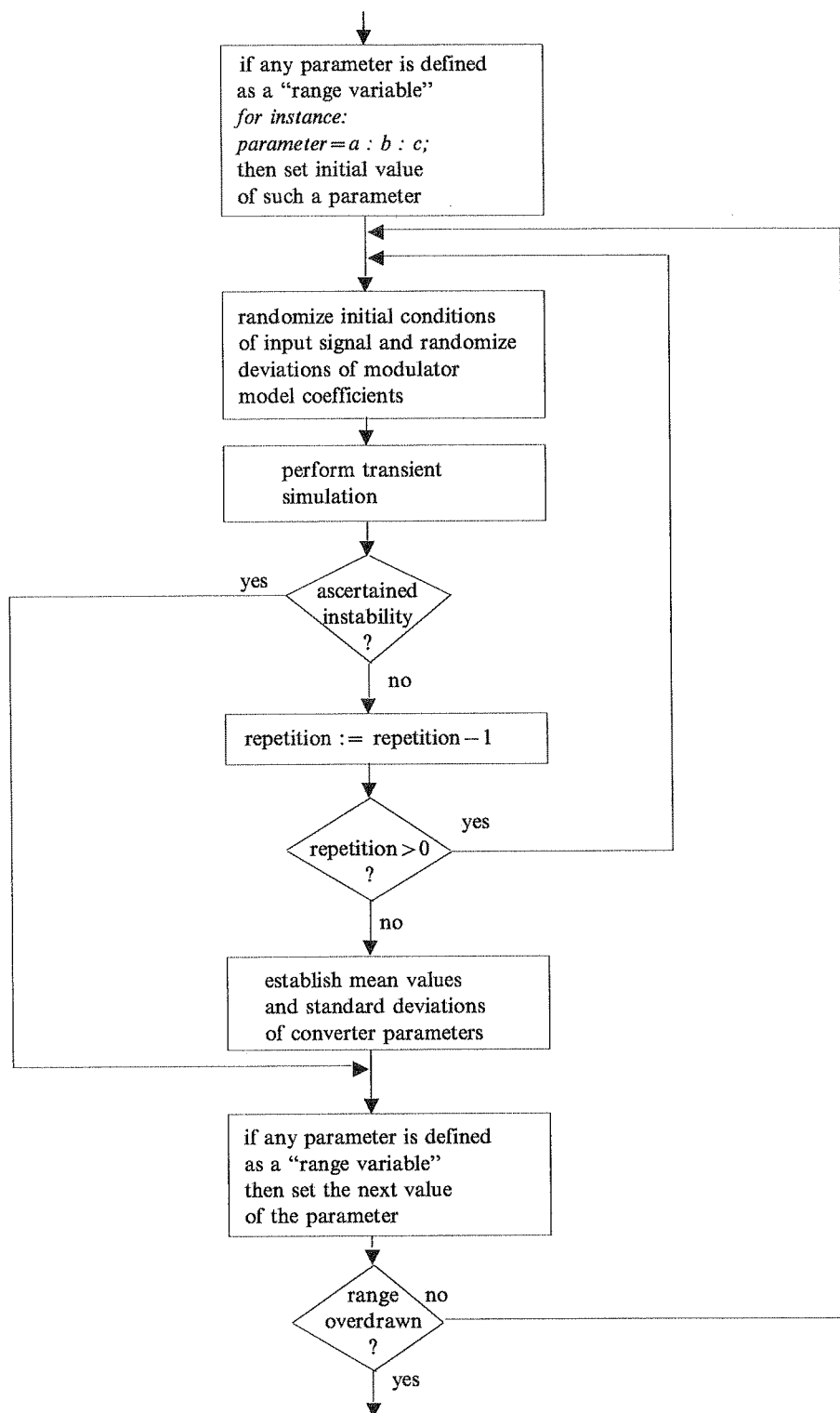


Fig. 5. Simplified block diagram of the program

General flexible and this model quality on same time case, two s of modula simulation

In ord operating simulator, the name Results of MATLAB should de examples

The ty plot of SN a plot. Th 4) with a transient instability simulation for arithm plot).

Sever Figure 5 amplitud parameter and cons an exam

The denotes of the fi

General model of simulated circuit shown in Figure 2 is proposed. This model is flexible and allows to fulfill all above-mentioned requirements. The disadvantage of this model is that there is only one decimating filter. The estimation of the modulator quality on the basis of decimating and approximation, cannot be carried out at the same time as the simulation of a real decimator (not always high-quality one). In this case, two separate simulations would be needed: one for accurate estimation of SNR of modulator (with the use of very good decimator model), and the second for simulation of the 'real' decimator.

In order for the presented program to be easily compiled and used under various operating systems, the user interface is maximally simplified. The input data for the simulator, defined by the user, should be written down into a text file. The user sets the name of this text file as a command-line parameter of the computer program. Results of simulation are printed in console (in brief form) and may be written into MATLAB-style data file. In order for simulator to create the output file, the user should define parameter *mat_file* in the input file. More detail description and examples of input data will be given later.

The typical way of estimating the dynamic range of a modulator is to make the plot of SNR as a function of input sinusoid amplitude. Figure 3 is an example of such a plot. The graph shows the results of simulation of second-order modulator (Figure 4) with an oversampling ratio 128. Each point of the plot is an issue of a single transient simulation. For amplitude equal 1.0, the simulation was broken, because instability of the modulator occurred. The Figure 3 was obtained as a result of two simulation program runs (the first for geometrical progression series and the second for arithmetical progression series — in order to compact the points at the end of the plot).

Several simulation program structures were tested. The structure presented in Figure 5 appeared to be the most flexible. In the above-mentioned example the amplitude is a "range variable", and the parameter *repetition* equals 1. This parameters as well as other used-defined parameters are written down into a text file and consists input data for simulator. The format of the text file will be described with an example:

```
osr = 128;  
modulator = "example2/std2";  
levels = 2;  
stability_limit = 100;  
repetition = 1;  
input = sin(0, 0.1:0.1:0.9, 0.00006103515625);  
outputs = 0 × 400;  
transient = 0 × 10;  
mat_file = "example2/results";
```

The parameter *osr* denotes the oversampling ratio. The next parameter *modulator* denotes the name of a text file that contains definition of modulator model (the syntax of the file will be described later). The parameter *levels* denotes the resolution of

quantizer which is used inside the modulator — resolution equals $\log_2(\text{levels})$. For the sake of simplicity it has been assumed that input voltage range is bipolar and reference voltage values of low-resolution DAC amount to -1 and $+1$. The instability is detected, when the absolute value of voltage at the output of any integrator exceeds the value of parameter *stability_limit*. The definition of input signal is given by the parameter *input* (more detailed description of the parameter will be given later). In above-mentioned example sinusoidal input signal was employed. There is no offset and the amplitude is denoted as a “range variable”, that is $0.1:0.1:0.9$. The frequency of the sinusoid equals $8/(128 \cdot 1024) = 6.10e-5$. The convention is that all frequencies are normalized to sampling frequency. The other user-defined parameters will be described later (the list of all parameters can be seen in Appendix A).

Another instance of a simulation with one “range variable” has been shown in Figure 6. The object of the simulation was a standard first-order modulator with oversampling ratio 128. In this case “range variable” was a value of constant input signal. The “range variable” assumed values from arithmetic series: $-1.000, -0.999, \dots, +1.000$. Each of 2001 points of plot is obtained from a single transient simulation. (Parameter *repetition* equaled 1).

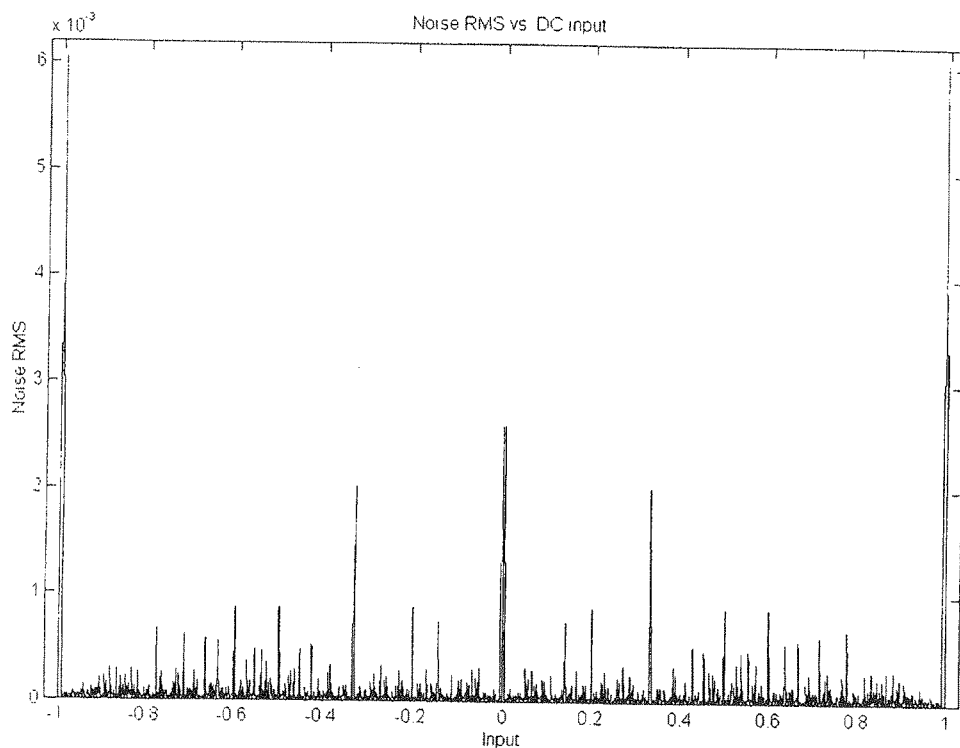


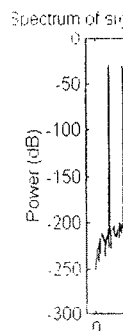
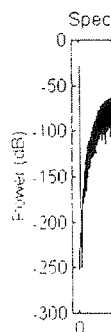
Fig. 6. The in-band noise as a function of the DC input for the standard first-order modulator with oversampling ratio 128, and 1-bit quantizer. The noise exhibits large peaks near inputs which are integers or simple rationals ($\pm 1, 0, \pm 1/2, \pm 1/3, \pm 2/3 \dots$) [1]

- There a
- Constant
 - Sinusoid
 - and freq
 - A random

where $K =$
 ξ, ζ — ran
 a — the la
 There are f
 Example o

In can b
 well as it is

In orde
 conversion
 least-squa



F

3. INPUT SIGNAL GENERATOR DESCRIPTION

There are five forms of input signal implemented:

- Constant input signal with user-defined value.
- Sinusoidal signal (the user sets three parameters of the sinusoid: offset, amplitude and frequency).
- A random signal, defined by the following equation:

$$x[n] = \text{offs} + a \sum_{k=0}^K \xi \cdot \cos(2\pi(f_start + k \cdot f_step) \cdot n + \zeta) \quad (4)$$

where $K = \text{floor}((f_stop - f_start)/f_step)$,

ξ, ζ — random variables, $\xi \sim \mathcal{N}(0; 1)$, $\zeta \sim \mathcal{U}(0; 2\pi)$,

a — the largest number for which the input signal $x[n] \in [\text{offs} - \text{ampl}; \text{offs} + \text{ampl}]$.

There are five user-defined parameters: offs , ampl , f_start , f_step , f_stop .

Example of a signal of this type is shown in Fig. 7.

It can be verified, that the stochastic process of the form (4) is strictly stationary as well as it is stationary in the wide sense.

In order to estimate properly the signal-to-noise ratio for the ADC the value of the conversion time is needed (because the output signal is fitted to delayed input signal in least-squares sense in order to calculate the SNR and other parameters). In such

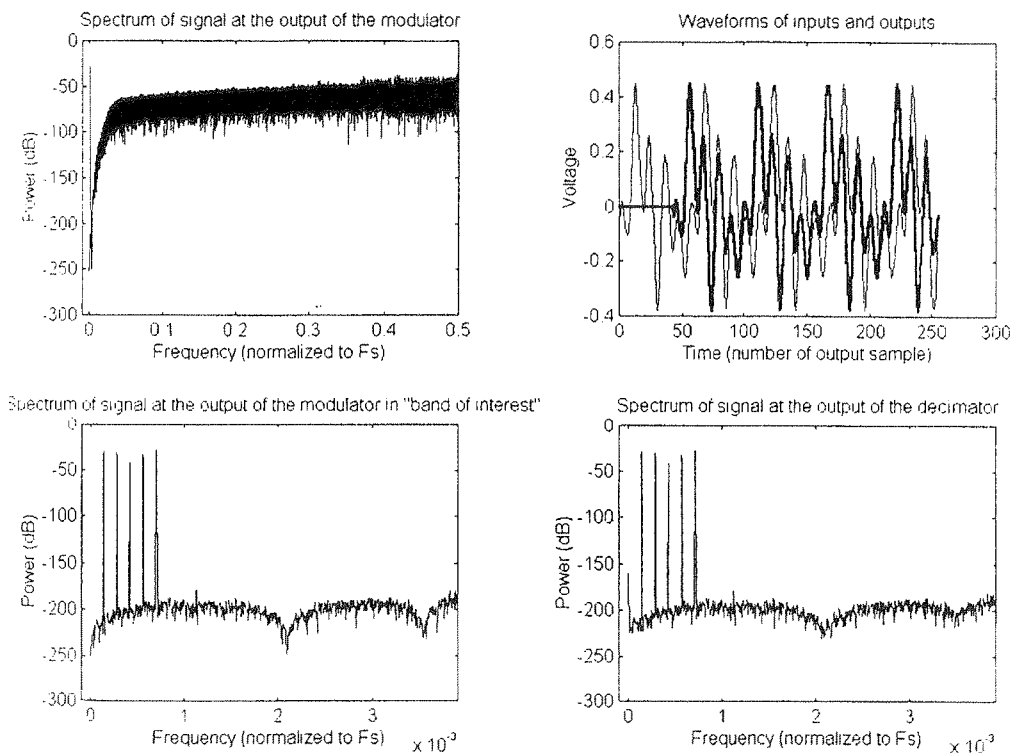


Fig. 7. Results of simulation of a fifth-order modulator with random input signal

a case the user should set the parameter *adc_conv_time*. For low-pass ADCs the value of the conversion time may be obtained as a result of a simulation with sinusoidal input signal of very low frequency.

- Single-sideband suppressed carrier amplitude-modulated sinusoidal signal [1, 4]. This type of input signal is definite with three parameters: amplitude, signal frequency and carrier frequency. In fact this is an ordinary sinusoid with frequency equal to the sum of the signal frequency and the carrier frequency. The information of the value of carrier frequency is necessary in order to properly estimate signal-to-noise ratio. The use of input signal of this type is the simplest way of examination of band-pass modulators. Example of a signal of this type is shown in Figure 8.

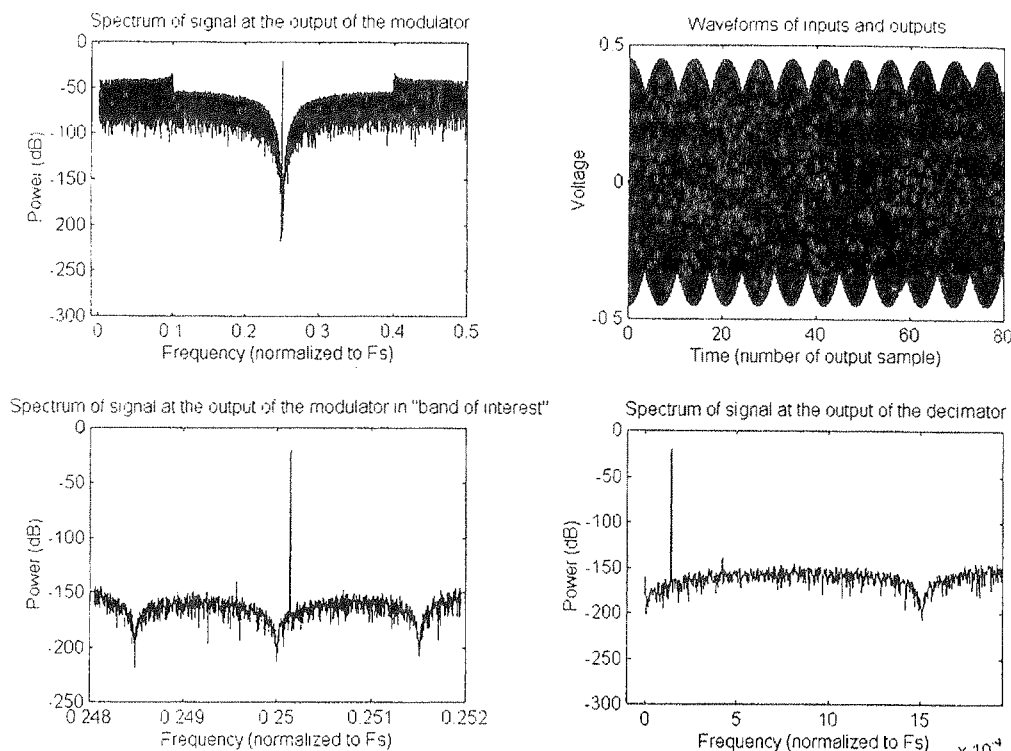


Fig. 8. Results of simulation of a sixth-order band-pass modulator with single-sideband suppressed carrier amplitude-modulated input sinusoidal signal

4. MODULATOR MODEL DESCRIPTION

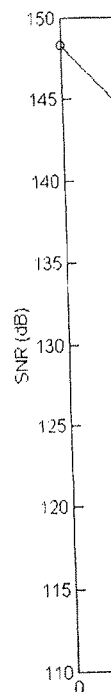
It is assumed that a modulator has a single quantizer and employs a linear loop filter [1]. There are five topologies of the modulator model implemented, as follows:

- feedback form of cascade-of-integrators,
- feedforward form of cascade-of-integrators,
- feedback form of cascade-of-resonators,

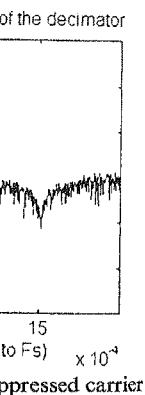
• feedf
• feedb
The topolo
The modul
are implem
text file, th

The param
model coe
extension

The si
coefficient
above) wa
normally
ficients. In
(the param



ss ADCs the
ulation with
al [1, 4]. This
frequency and
al to the sum
the value of
ise ratio. The
of band-pass



a linear loop
as follows:

- feedforward form of cascade-of-resonators,
- feedback form of cascade-of-resonators with delaying quantizer.

The topologies can be seen in Appendix B.

The modulator structures have been chosen, because they are commonly used, and are implemented in Schreier's toolbox for designing of $\Delta\Sigma$ modulators [6]. Exemplary text file, that depicts a fifth-order modulator is shown below:

```
form=CRFB
a=[0.101619 0.179003 0.215454 0.3700379 0.356282];
g=[0.000912 0.001136];
b=[0.101619 0 0 0 0];
c=[0.130912 0.191611 0.379203 0.435599 1.559734];
```

The parameter *form* defines the modulator topology, and the other parameters set the model coefficients. The text file, that defines delta-sigma modulator should have the extension "mod".

The simple analysis of modulator sensitivity to deviations of values of its coefficients is shown in Figure 9. The fifth-order modulator (with coefficient listed above) was simulated. Figure 9 presents SNR as a function of standard deviation of normally distributed random variables which disturb the modulator model coefficients. In this example each point of the plot results from 15 transient simulations (the parameter *repetition* equals 15). Before each transient simulation, values of

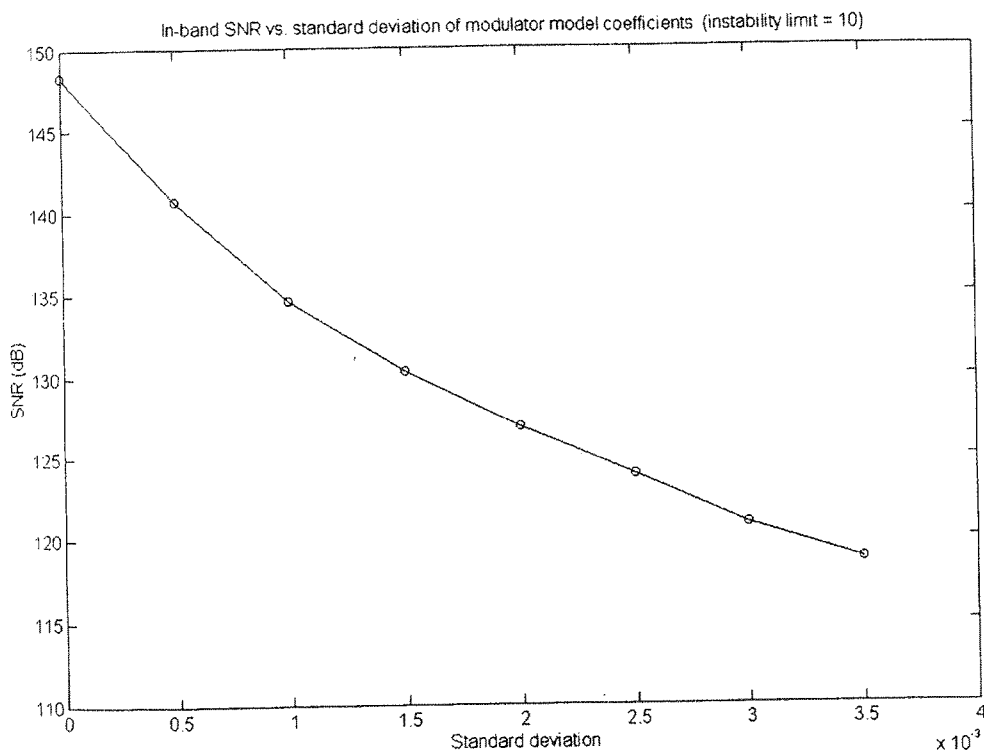


Fig. 9. In-band SNR vs. standard deviation of modulator model coefficients

realizations of every random variables (concerning every modulator model coefficients) were randomized. After each transient simulation the signal power and the in-band noise power were estimated. After whole serial of transient simulations, average values of the signal power and the noise power were calculated. The method of "averaging" leads to more reliable results. The input data for this this simulation is shown below:

```
osr=128;
modulator="example5/ord5";
mod_tol_add=0 : 0.0005 : 0.0035;
leakage_tol=0 : 0.0005 : 0.0035;
repetition=15;
input=sin(0, 0.45, 0.00006103515625);
voltage_range=1;
stability_limit=10;
outputs=0×800;
transient=0×10;
mat_file="example5/results";
```

Parameters *mod_tol_add* and *leakage_tol* are standard deviations of independent random variables (they are equal one to another in the example). More detail description of the parameters as well as the other parameterse can be seen in

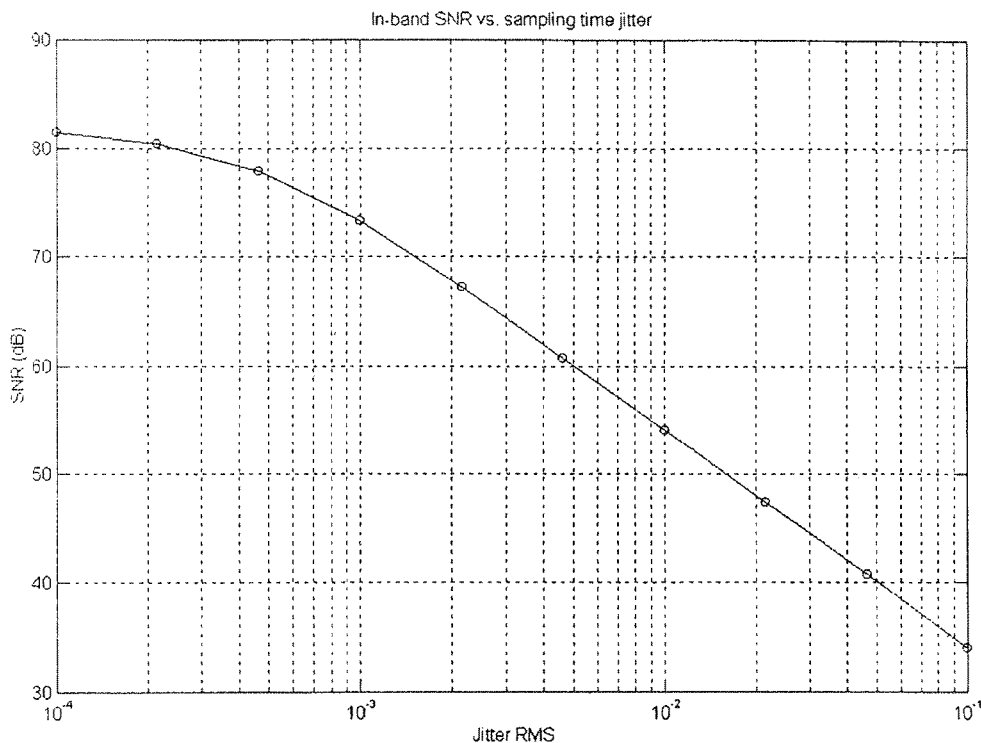


Fig. 10. In-band SNR vs. jitter RMS value

Appendix A
instability f
more reliab

Another
mance is sh
modulators
concerning
carrier amp
case the us
sampling ti
jitter is defi
that the par
10⁻¹). The

Sampling
of the pov
this case.

The c
componen
all out-o
the prese
structure
following

coefficients)
the in-band
average values
"averaging"
known below:

Appendix A. The plot ends at value 0.0035 because long simulations detected instability for the value 0.004 (parameter *repetition* equaled 50 in order to obtain more reliable results).

Another kind of flux of analog devices imperfections on the modulator performance is short-term timing instability (the sampling time jitter). Only band-pass modulators are very sensitive to sampling time jitter. An exemplary simulation concerning a sixth-order band-pass modulator (with single-sideband suppressed carrier amplitude-modulated sinusoidal input signal) is shown in Figure 10. In this case the user sets a standard deviation of a random variable that constitutes the sampling time jitter. In the plot a semilogarithmic scale is used, and the parameter *jitter* is defined as a "range variable" $0.0001 \# 2.15443469 \# 0.1$. This notation means that the parameter takes values from geometrical progression series (10^{-4} , $10^{-4+1/3}$, ..., 10^{-1}). The input data for this simulation is following:

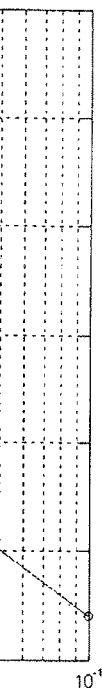
```
osr = 128;
modulator = "example6/bandpass";
jitter = 0.0001 # 2.15443469 # 0.1;
voltage_range = 1;
stability_limit = 10;
outputs = 0 x 800;
transient = 0 x 200;
repetition = 1;
input = AM(0.45, 0.00006103515625, 0.25);
decimator = ((4, f, "example16/b_170", 30)
(2, f, "example16/16_200", 30)
(2, f, "example16/16_200", 30)
(2, f, "example16/16_200", 30)
(2, s, "example16/5_150", 36)
(2, s, "example16/5_150", 36));
fft_modulated = off;
mat_file = "example16/results";
```

Sampling time jitter causes large error of the method of SNR estimation on the basis of the power spectrum, so the high-quality decimating filtering has to be employed in this case.

5. DECIMATOR DEFINITION

The output of the modulator represents a sum of input signal and out-of-band components, mainly modulation noise. The role of the decimator is attenuation all out-of-band noise energy and resampling the signal at the Nyquist rate. In the presented simulation program the decimator model has generally multistage structure and each stage is described separately. Its description is given in the following form:

of indepen-
More detail
be seen in



decimator = ((*description_of_stage*)...(*description_of_stage*));
description_of_stage = (*field1*, *field2*, *field3*, *field4*)

For instance:

decimator = ((2, c, 3, 30)(2, f, "filter1", 30)(2, f, "filter2", 30)(2, f, "filter3", 30)
 (2, s, "filter4", 36) (2, s, "filter5", 36) (2, s, "filter6", 36));

In this example there are seven decimating stages. Field 1 contains value of the decimation factor (a positive integer). The product of the decimation factors of the stages should be equal to the oversampling ratio. If the decimation factor is equal to 1, then the stage reduces to an ordinary filter.

Field 2 contains a single letter that defines a type of the digital filter, as follows:
 c — comb filter (sinc^K),
 f — FIR filter (with coefficients in floating-point form),
 i — IIR filter (with coefficients in floating-point form),
 s — complex filter (with coefficients in signed-digit decomposition form; the decomposition will be described later).

Field 3 contains detailed description of the filter, as it is shown in Table 2.

Table 2

Detailed description of decimating filter

Field 2	Field 3
c	An exponent of sinc^K function, that is K
f	A name of a text file which contains the filter coefficients in floating-point form, for example: 2.526806e-03 3.728009e-04 ... The file name extension has to be ".fir".
i	A name of text files which contain the filter coefficients (in floating-point form) — separately the denominator coefficients and the nominator coefficients (the default extensions of the names are ".A" and ".B" respectively).
s	A name of text files which contain the filter coefficients (in signed-digit decomposition form) — separately the outer filter coefficients and the inner filter coefficients (the default extension of the names are ".q1" and ".q2" respectively).

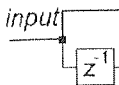
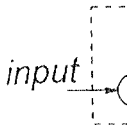
Field 4 contains the number of bits of fix-point data representation at the stage output.

The signed-digit decomposition form may be easily explained by following example:

$-1 \ -3 \ -8$
 $1 \ 1 \ -1$
 $-2 \ -4 \ -9$
 $-1 \ 1 \ -1$

The odd row
first coefficient
 $-2^{-2} + 2^{-4}$
hardware-eff
binary addi

As it wa
have one of
• sinc^K dec



e));

'filter3", 30)

value of the
factors of the
or is equal to

, as follows:

on form; the

Table 2.

Table 2

point form, for

point form) —
s (the default

decomposition
coefficients (the

at the stage

by following

The odd rows contain the power of two, while the even rows contain their signs, so the first coefficients in above-mentioned example is $2^{-2} + 2^{-3} - 2^{-8}$, the next coefficient is $-2^{-2} + 2^{-4} - 2^{-9}$, and so on. Such a form is adequate if the filter has to be hardware-efficient, because instead of each multiply-accumulate operation some binary additions are performed.

As it was mentioned above, a digital filter applied in the decimating stage may have one of following structures:

- sinc^K decimating filter (comb filter), illustrated by Fig. 11,

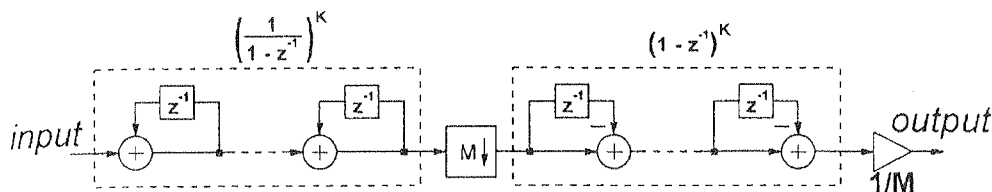


Fig. 11. A decimating stage with sinc^K filter

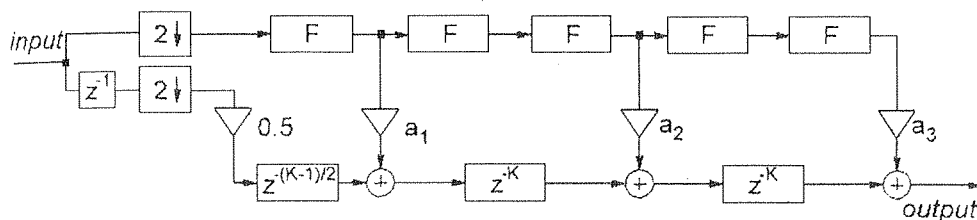


Fig. 12. A decimating stage with cascaded identical inner filters (F)

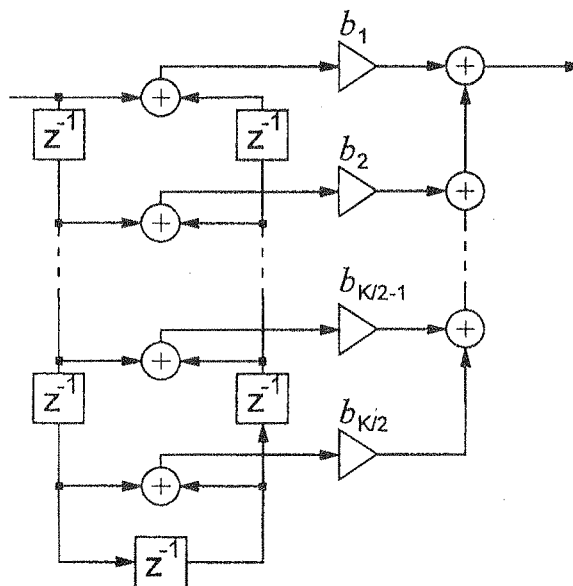


Fig. 13. An inner filter

- FIR filter (the symmetry in the impulse response was assumed) [10, 11],
- IIR direct-form filter structure [10, 11],
- half-band filter based on the procedure described by Saramäki [12, 6]. An exemplary filter of this type is shown in Figures 12 and 13. Identical inner filters (F) are ordinary FIR filters with symmetric impulse response. Coefficients a_i and b_i define precisely the complex filter. The decimating stage of this form should have the decimation factor equal two.

6. CONCLUSIONS

Advantages of the presented simulator:

- The presented computer program can simulate the most of commonly used-sigma ADCs.
- The simulator functions quickly and is optimized in respect to the amount of used operational memory.
- The program may be compiled various operating systems (UNIX, WINDOWS, DOS).
- The form of input text files enables an easy construction of the files by MATLAB procedures, especially with a toolbox created by Richard Schreier [6] (the toolbox is a set of MATLAB programs, which includes among others a tools for designing delta-sigma modulators and designing half-band filters by the way that is based on the procedure described by Saramäki [12]).
- The simulator program as well as the source code in C++ is available on author's home page <http://www.ipe.pw.edu.pl/~kpole>.

Disadvantages of the simulator:

- Dynamic errors of analog devices cannot be modeled. In order to obtain this task, behavioural models of analog components should be implemented.
- More complex topologies of the modulator model are not implemented, for instance: modulators including some quantizers (among others multistage architecture) and modulators that employ dithering.
- Complex (multiplexed) decimating filters (for band-pass decimating) are not implemented.
- There is no possibility of controlling the way of rounding results of arithmetic operations in decimating filters (truncating towards minus infinity is always used).

REFERENCES

1. S. Norsworthy, R. Schreier, G. Temes: *Delta-Sigma Data Converters*. IEEE Press, 1997, pp. 1–43, 141–164, 282–308, 406–468
2. K. Suyama, Y. Tsiividis: *Simulation of mixed switched-capacitor/digital networks with signal-driven switches*. IEEE J. Solid-State Circuits, vol. 25, no. 6, pp. 1403–1413, Dec. 1990
3. V. Liberali, V. Dias, F. Maloberti: *TOSCA: A Simulator for Switched-Capacitor Noise-Shaping A/D Converters*. IEEE Tr. on Computer Aided Design of Integrated Circuits and Systems, vol. 12, no. 9 September 1993

4. G. Brau: *delta-sigma*
5. <http://www>
6. <ftp://next2>
7. L. Risb: *dissertation*
8. R. Schreier: *on Circuits*
9. IEEE Stan
10. R. Cro: *Company*
11. J. Proa: *Company*
12. T. Sara: *Transacti*

1],
[12, 6]. An
er filters (F)
ts a_i and b_i
should have

used-sigma

ount of used

WINDOWS,

MATLAB
the toolbox
or designing
is based on

on author's

in this task,

nented, for
ge architec-

g) are not

f arithmetic
ways used).

. IEEE Press,

orks with sig-
990
ched-Capacitor
Circuits and

4. G. Brauns, R. Bishop, M. uSteer, J. Paulos, S. Ardalan: *Table-based modeling of delta-sigma modulators using ZSIM*. IEEE Tr. on Computer-Aided Design, vol. 9, no. 2, February 1990.
5. <http://www.analog.co.uk> (this is WWW home page of Analogue Integration Ltd.)
6. <ftp://next242.ece.orst.edu/pub./delsig.tar.Z> (this is FTP site of Richard Schreier's toolbox).
7. L. Risbo: *Σ - Δ Modulators- Stability Analysis and Optimization* Ph.D., June 1994, pp. 23—50. This dissertation is available from WWW site <http://eivind.imm.dtu.dk/publications/>
8. R. Schreier: *An empirical study of high-order single-bit delta-sigma modulators*. IEEE Transactions on Circuits and Systems II, vol. 40, no. 8, August 1993
9. IEEE Standard for Digitizing Waveform Recorders, IEEE Std 1057—1994
10. R. Crochiere, L. Rabiner: *Multirate Digital Signal Processing*, Macmillan Publishing Company, a division of Macmillan, 1988, pp. 401—413, 549, 795—857.
11. J. Proakis, D. Manolakis: *Introduction to Digital Signal Processing*, Macmillan Publishing Company, a division of Macmillan, 1988, pp. 401—413, 549, 795—856
12. T. Saramäki: *Design of FIR as a tapped cascaded interconnection of identical subfilters*. IEEE Transactions on Circuits and Systems, vol. 34, pp. 1011—1029, 1987

APPENDIX A

In Table 1 list of user-defined parameters is given.

Table 1

List of user-defined parameters

parameter	description	default value /parameter is necessary
osr	the oversampling ratio	necessary
levels	the number of output levels of the quantizer; $levels \in \{2; 3; \dots 32\}$	2
stability_limit	the threshold which the absolute values of voltages at the outputs of the integrators should not exceed, if the modulator is called "stable"	10
modulator	the name of the text file that defines delta-sigma modulator model (the extension of this file has to be "mod")	necessary
mod_tol_add mod_tol_mul	standard deviation of normally distributed random variables which are used to disturb values of the modulator model coefficients; each coefficient is described by the following equation: $c_s = c_N \cdot \xi + \eta$ where c_N — nominal value (defined by parameter <i>modulator</i>), c_s — value used in simulation, $\xi \sim \mathcal{N}(1; mod_tol_mul)$, $\eta \sim \mathcal{N}(0; mod_tol_add)$	0 0
leakage leakage_tol	these parameters concern the integrators models; the integrators are described by the following difference equations: $y[n] = x[n] + y[n-1] \cdot (1 - leakage + \xi)$ or $y[n] = x[n-1] + y[n-1] \cdot (1 - leakage + \xi)$ where $\xi \sim \mathcal{N}(0; leakage_tol)$	0 0
thresholds_tol	standard deviation of quantizer thresholds; each threshold value is described by the equation: $U_s = U_N + \xi$ where U_N — nominal value of the threshold that depends only on the parameter <i>levels</i> , U_s — value used in simulation, $\xi \sim \mathcal{N}(0; thresholds_tol)$	0
reference_tol	standard deviation of output levels of the low-resolution DAC; each output level of the DAC is described by the following equation: $V_s = V_N + \xi$, where V_N — nominal value that depends only on the parameter <i>levels</i> , V_s — value used in simulation, $\xi \sim \mathcal{N}(0; reference_tol)$	0
jitter	standard deviation of short-term timing instability (the sampling time jitter); $x[n] = x(T_s \cdot n + \xi)$, where $\xi \sim \mathcal{N}(0; jitter)$, T_s — sampling period; because the convention is that frequencies are normalized, $T_s = 1$;	0
decimator	the multistage decimator description; the possibility of decimator defining is optional	—

Keywords:

input	the definition of input signal may have one of following forms: $input = constant;$ $input = \sin(offset, amplitude, frequency);$ $input = random(off, ampl, f_start, f_stop, f_step);$ $input = AM(amplitude, frequency, carrier_frequency);$	necessary
adc_conv_time	value of delay of converter (normalized to sampling period); this parameter is used if random input signal is employed; the parameter should have a negative value	0
repetition	this parameter has to be set, if a succession of transient simulations is to be performed, in order to establish average values and the standard deviations of the signal-to-noise ratio and other performance parameters of the converter, ("average" power spectrums are computed also in order to "smooth" the spectrums)	1
random_pulse	this parameter is used if <i>repetition</i> is greater than one and a constant input signal is employed; in second and the next transient simulations of the succession the first sample of the input signal is random: $x[0] = random_pulse \cdot \text{sgn}(\xi) \cdot \xi ^{1/2}$ where $\xi \sim \mathcal{U}(-1; +1)$; (if sinusoidal input signal is employed; the first sinusoid has phase equal to zero, and the next sinusoids have random phases)	0.5
outputs	the length of a vector of output samples, which is a basis for estimation of performance parameters of ADC, if parameter <i>decimator</i> is defined; The number of samples at the output of modulator that is used for computing the power spectrum equals $2^{\lfloor \log_2(outputs \cdot osr) \rfloor} \approx outputs \cdot osr$	0×200
transient	the number of initial samples at the output of the decimator which should be ignored, in order for the decimating filters to reach their steady states; (the number of ignored initial samples at the output of the modulator equals <i>osr-transient</i>)	0×200
mat_file	the name of a MATLAB-style output file (the default extension is ".mat"); if the parameter is not set, no output file will be created	—
points	this parameter is used to set the length of waveforms that are written down into the MATLAB-style file (the number of output samples equals <i>points</i> ; the length of internal waveforms equals <i>points \cdot osr</i>)	0
stability_check_only	this is the switch, that takes value "on" or "off"; if it is turned on, all optional analyses are disabled	off
voltage_range	input voltage range	2
fft_modulated	this is the switch, that takes value "on" or "off"; if it is turned off, calculation and analysis of power spectrum of signal at the output of the modulator is disabled	on

APPENDIX B

The modulator structures which are implemented in presented simulator are shown in Figures 1–5.

If the integrators are leaky ones, then integrators models become more complex, as it is shown in Figure 6. Parameter p is defined with the following equation:

$p = 1 - \text{leakage} - \xi$ where $\xi \sim \mathcal{N}(0; \text{leakage_tol})$, leakage and leakage_tol are parameters defined by the user.

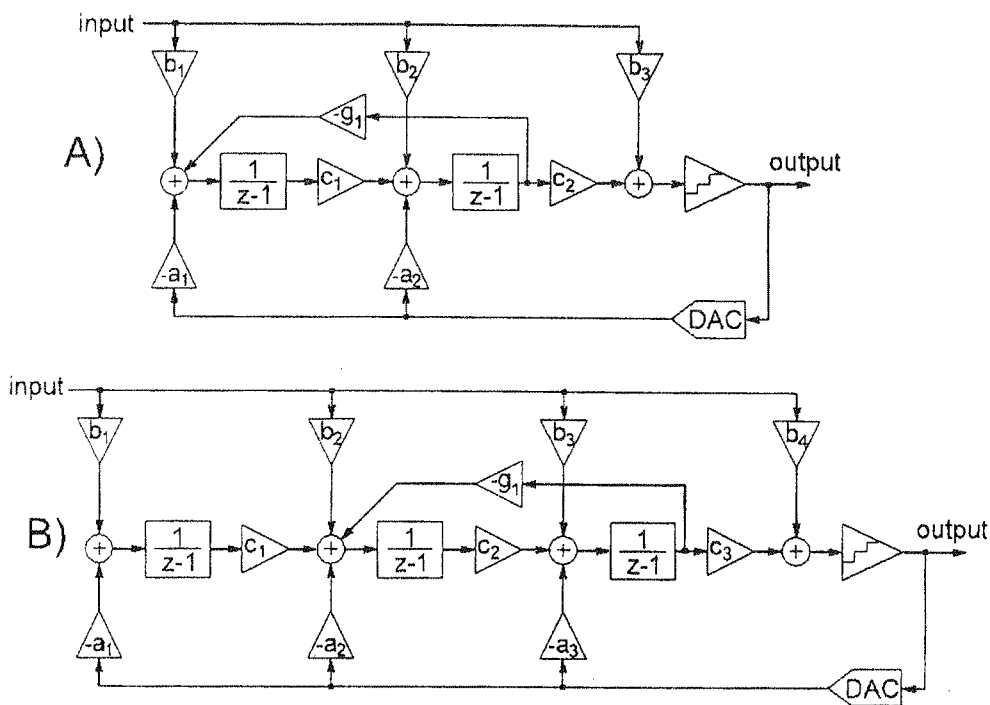


Fig. 1. Cascade-of-integrators structure, feedback form (CIFB). A) Even order. B) Odd order

input

A)

input

B)

Fig. 2. C

A)

input

B)

Fig. 3

...ktr. i Telekom.

...ulator are

...re complex,
...uation:

...ge_tol are

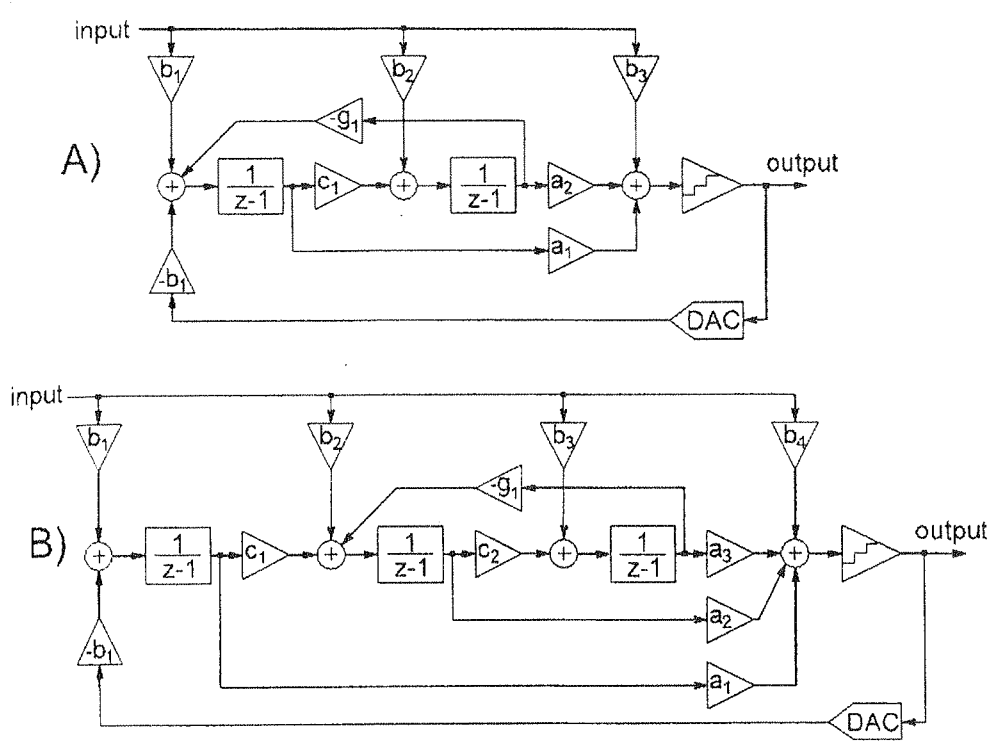


Fig. 2. Cascade-of-integrators structure, feedforward form (CIFF). A) Even order. B) Odd order

output

Odd order

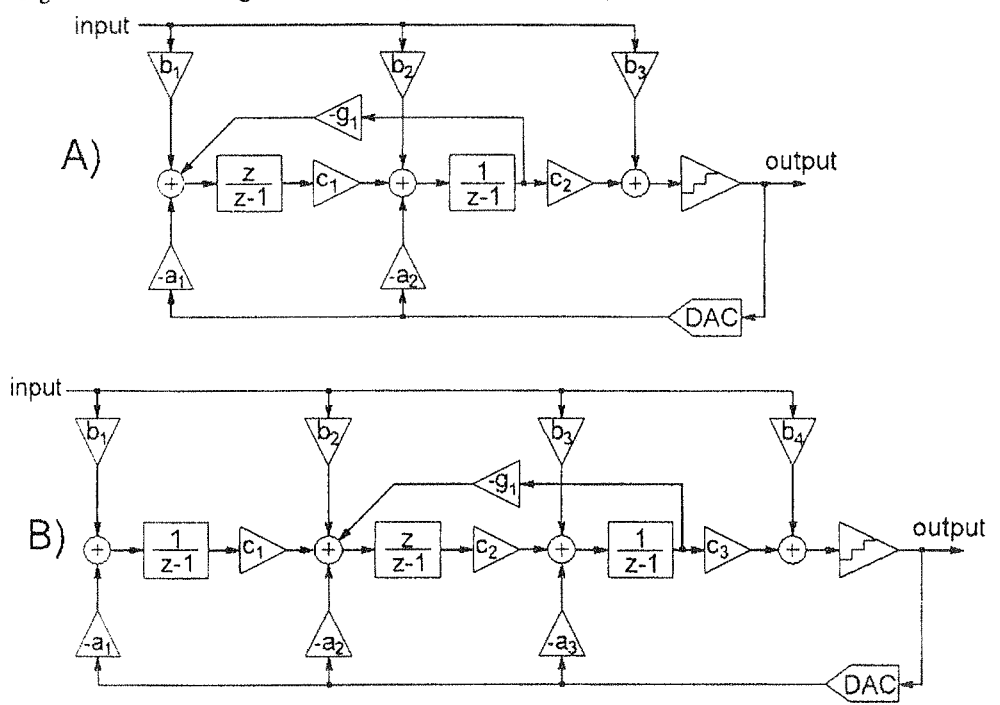


Fig. 3. Cascade-of-resonators structure, feedback form (CRFB). A) Even order. B) Odd order

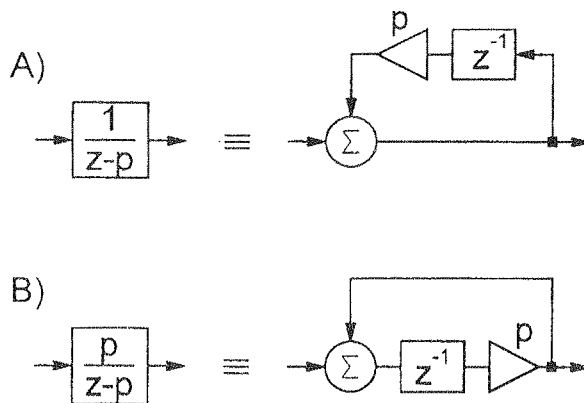


Fig. 6. Integrator model with leakage, A) non-delaying integrator, B) delaying integrator

K. OLEJARCZYK

SYMULACJA PRZETWORNIKÓW ANALOGOWO-CYFROWYCH DELTA-SIGMA Z SYMULATOREM AMBER

Streszczenie

Celem pracy jest przedstawienie doświadczeń zebranych w czasie tworzenia programu do symulacji przetworników analogowo-cyfrowych delta-sigma. Modulator delta-sigma, najważniejszy blok funkcjonalny przetwornika, charakteryzuje chaotyczna dynamika (chaos deterministyczny). Nie została dotąd stworzona teoria pozwalająca przewidzieć zachowanie modulatora delta-sigma (w ogólnym przypadku), dlatego empiryczne badanie modulatorów jest koniecznością. W artykule przedstawiono specjalizowany symulator, o nazwie „Amber”, który został udostępniony na stronie WWW autora wraz z kodem źródłowym w C++. Omówiono również wyniki przykładowych symulacji, pokazujących główne zalety i wady prezentowanego programu. Maksymalne uproszczenie interfejsu użytkownika umożliwia kompilację i użytkowanie programu symulacyjnego pod systemami operacyjnymi INIX, WINDOWS i DOS. Prezentowany symulator ma również istotną wartość dydaktyczną.

Słowa kluczowe: modulacja delta-sigma, kształtowanie szumu, nadpróbkowanie, decymacja, filtry cyfrowe, przetwarzanie analogowo-cyfrowe.

Alg

U
cyfrow
z pod
związ
podzi
Zapre
w Pol

Słowa

$\overline{Y}$
 $\overline{\overline{Y}}$
 Y_1, Y_2
 Y_i
[x]

Różn
PAL —
Program
CPLD —
z kluczow
projektu
(Program
Więks
charakte

Algorytmy podziału wyjść umożliwiające realizację układów cyfrowych w strukturach PLD

DARIUSZ KANIA

Instytut Elektroniki, Politechnika Śląska, Gliwice

Otrzymano 1998.12.12

Autoryzowano 1999.03.17

Układy programowalne mają ograniczoną liczbę wyjść. Problem realizacji układów cyfrowych w oparciu o struktury programowalne jest między innymi ściśle związany z podziałem wyjść. Podział wyjść polega na rozbiciu projektowanego układu na mniejsze związane z poszczególnymi struktury programowalne. Artykuł przedstawia algorytmy podziału wyjść wykorzystujące zewnętrzne transkodery, dekodery lub demultipleksery. Zaprezentowane algorytmy zostały zaimplementowane w systemie Decomp opracowanym w Politechnice Śląskiej.

Słowa kluczowe: podział, dekompozycja, Układy Logiki Programowalnej

WYKAZ OZNACZEŃ

- $\underline{\underline{Y}}$ — zbiór zmiennych wyjściowych
- $\overline{Y}$ — moc zbioru Y
- Y_1, Y_2 — podzbiory zbioru zmiennych wyjściowych Y
- Y_i — i -ta zmienna wyjściowa
- $[x]$ — wartość całkowita liczby x

1. WPROWADZENIE

Różnorodność struktur programowalnych (PLE — Programmable Logic Element, PAL — Programmable Logic Array, GAL — Generic Array Logic, FPLA — Field Programmable Logic Array, FPLS — Field Programmable Logic Sequencer, CPLD — Complex Programmable Logic Devices [1, 5, 14]) spowodowała, że jednym z kluczowych problemów projektowania układów cyfrowych stał się sposób podziału projektu na poszczególne bloki, związany z doбором odpowiednich układów PLD (Programmable Logic Devices).

Większość układów PLD wykorzystuje jedną z trzech struktur AND-OR charakterystyczną dla układów PLE, PAL oraz FPLA. Praktyczne doświadczenia

autora zdobyte podczas projektowania układów cyfrowych na bazie struktur programowalnych, wskazały na możliwość opracowania algorytmów, związanych z podziałem projektu, prowadzących do konstruowania „lepszyc” (tańszych) układów. Algorytmy te mogłyby znacznie zwiększyć możliwości znanych autorowi systemów wspomagających projektowanie układów cyfrowych na bazie struktur PLD, które najczęściej odpowiadają jedynie na pytanie, czy można zrealizować układ na podstawie opisu wprowadzonego przez projektanta. Jeżeli nie można to znaczy, że projektant powinien wybrać „większą” strukturę albo spróbować inaczej opisać układ.

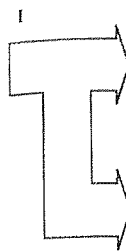
Jak podzielić projektowany układ na mniejsze bloki o ściśle określonych zasobach wewnętrznych, konkretnej liczbie wejść i wyjść?

W literaturze [3, 11] można znaleźć idee podziału wejść, wyjść, termów, lecz nie są przedstawione algorytmy poszukiwania takich rozwiązań. Nie wiadomo jak dobierać zmienne podziału termów, czy też jak poszukiwać zbiorów zmiennych umożliwiających podział wyjść lub wejść. Przedstawione w wielu pracach różnorodne algorytmy dekompozycji bazujące na teorii Curtis'a [2, 5], związane przede wszystkim z podziałem wejść i realizacją układów w strukturach FPGA, najczęściej mają nikłe znaczenie praktyczne w przypadku realizacji układów cyfrowych w prostszych strukturach PLD takich jak np. 16V8, 22V10 itp. Poza tym złożoność tych algorytmów często wyklucza ich praktyczne wykorzystanie.

Tematem artykułu jest jeden, wydaje się, że najprostszy problem — problem podziału wyjść. Do opisu układów wykorzystano tablice przepaleń [7, 11] ponieważ przejrzyste odzwierciedlają one zasoby wewnętrzne struktur programowalnych a dodatkowo czytelnie prezentują przedstawione algorytmy. Tablica przepaleń składa się z dwóch części, pierwszej opisującej matrycę AND i drugiej opisującej matrycę OR. Poszczególne wiersze tablicy przepaleń odpowiadające termom (iloczynom stanowiącym składniki sum), określają aktywne stany wejść (H, L, -) lub wyjść (A, -) projektowanego układu (symbol A oznacza, że dany term jest dołączony do odpowiedniej bramki OR). Nazwijmy tablicę przepaleń opisującą projektowany układ i zawierającą dowolną (konieczną do opisu projektowanego układu) liczbę kolumn Pierwotną Tablicą Przepaleń (PTP). Ogólnie rzecz biorąc, celem rozważań jest takie przekształcenie, podział tzw. Pierwotnej Tablicy Przepaleń (PTP) opisującej projektowany układ na podtablice, który umożliwi bezpośredni podział układu kombinacyjnego na poszczególne bloki, podukłady o ściśle określonej liczbie wyjść. Proces podziału polega więc na podziale PTP opisującej projektowany układ o dowolnej liczbie wyjść, na tzw. Charakterystyczne Tablice Przepaleń odpowiadające konkretnym strukturom programowalnym o ściśle określonej liczbie wyjść [11].

2. PROSTY PODZIAŁ WYJŚĆ

Najprostszą metodą podziału jest prosty podział wyjść. W literaturze można spotkać się z innymi nazwami takimi jak prosta ekspansja liczby wyjść [3, 11] czy też dekompozycja równoległa [9]. Metoda ta wraz ze sposobem podziału Pierwotnej Tablicy Przepaleń przedstawiona jest na rysunku 1.



Rys. 1.

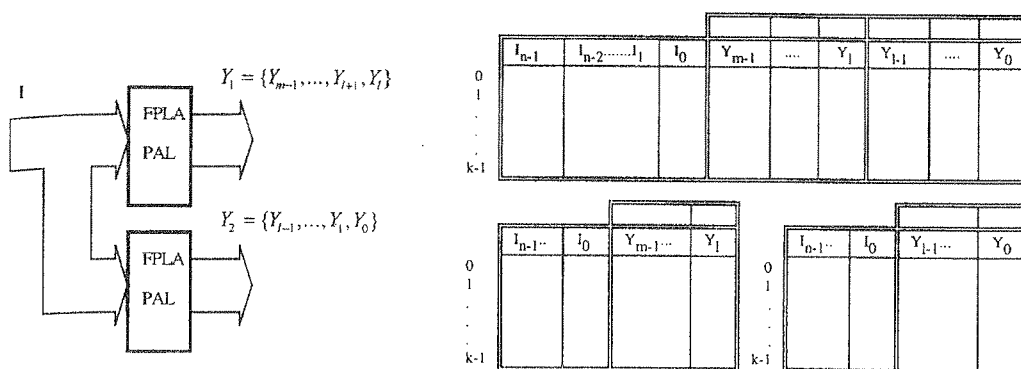
3. PODZIAŁ

Znane
rzystujące
zawartej w
do podzia
PLD. Ide
na rysunk

W og
kombinac
 m_t — liczb
 $k_t \leq 2^{n_t}$; n_t

Definicja

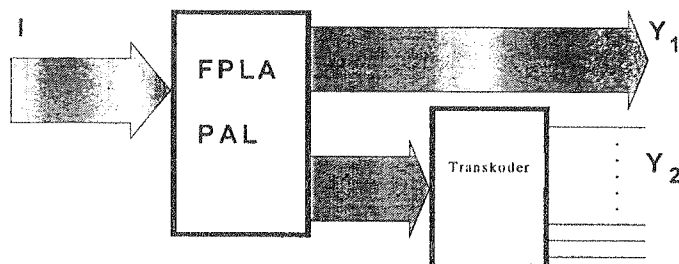
Współ
słów utw
(patrz pr



Rys. 1. Prosty podział wyjść — podział PTP na podtablice opisujące poszczególne produkty

3. PODZIAŁ WYJŚĆ WYKORZYSTUJĄCY ZEWNĘTRZNY TRANSKODER

Znane są w literaturze metody ograniczania długości słowa pamięci wykorzystujące zewnętrzny układ kombinacyjny, polegające na odpowiednim kodowaniu zawartej w niej informacji [10, 12]. Układ kombinacyjny można również wykorzystać do podziału wyjść w przypadku realizacji układów cyfrowych na bazie struktur PLD. Idea podziału wykorzystująca zewnętrzny transkoder przedstawiona jest na rysunku 2.



Rys. 2. Podział wyjść wykorzystujący zewnętrzny transkoder

W ogólnym przypadku transkoder (PLE, PAL, GAL, FPLA) jest układem kombinacyjnym scharakteryzowanym przez trzy parametry, n_t — liczbę wejść, m_t — liczbę wyjść, k_t — liczbę słów kodowych i spełniającym następujące zależności $k_t \leq 2^{n_t}$; $n_t < m_t$. „Najlepszym” transkoderem jest struktura PLE, dla której $k_t = 2^{n_t}$ [6].

Definicja 1:

Współczynnikiem komplikacji $w_k(Y_{s-1}, \dots, Y_{p+1}, Y_p)$ nazywamy liczbę różnych słów utworzonych przez zmienne wyjściowe $Y_{s-1}, \dots, Y_{p+1}, Y_p$ występujące w PTP (patrz przykład tab. 1).

T a b l i c a 1

Przykładowa PTP oraz współczynniki komplikacji $w_k(Y_4, Y_3, Y_2)$ i $w_k(Y_2, Y_1, Y_0)$

Y4	Y3	Y2	Y1	Y0
A	*	*	*	A
A	*	*	A	A
A	*	A	*	A
*	A	A	*	*
*	A	A	A	A
*	A	*	A	A

$$w_k(Y_4, Y_3, Y_2)=4$$

$$w_k(Y_2, Y_1, Y_0)=5$$

Liczbę wyjść PTP można zmniejszyć wykorzystując zewnętrzny transkoder wtedy, gdy istnieje podzbiór Y_2 zbioru zmiennych wyjściowych Y , dla którego spełnione są następujące zależności

$$\overline{\overline{Y}}_2 \leq m_t \quad (1)$$

$$w_k(Y_2) < k_t \quad (2)$$

$$\overline{\overline{Y}}_2 > [\lg_2(w_k(Y_2))]. \quad (3)$$

Dla zbioru Y_2 , dla którego spełnione są zależności (1), (2) minimalna liczba wyjść transkodera wynosi $n_t = [\lg_2(w_k(Y_2))]$ stąd jeżeli jest spełniona zależność (3) to

$$\overline{\overline{Y}}_2 > n_t. \quad (4)$$

Liczba wyjść podtablicy po zastosowaniu zewnętrznego transkodera wynosi $\overline{\overline{Y}} - \overline{\overline{Y}}_2 + n_t$ i jest mniejsza (4) od pierwotnej liczby wyjść stąd możliwe jest zmniejszenie liczby wyjść PTP jeżeli spełnione są zależności (1), (2), (3).

W praktyce spotykamy się najczęściej z następującym problemem. Chcemy zrealizować układ o $\overline{\overline{Y}}$ -wyjściach. Jeżeli liczba wyjść równa $\overline{\overline{Y}}$ jest większa od liczby wyjść dostępnych struktur PLD to konieczny jest podział PTP. Decydujemy się na konkretny transkoder o parametrach n_t , m_t i k_t . Poszukujemy podziału PTP, który w pełni wykorzysta zasoby transkodera (n_t , m_t , k_t) i spowoduje, że podtablica PTP powstająca po podziale wymusza zastosowanie układów o możliwie najmniejszych zasobach.

Algorytm podziału PTP na podtablice PTP przeznaczone do realizacji na elementach PLD i transkoderze

1. Wybieramy podzbiór zbioru Y spełniające następującą zależność: $\overline{\overline{Y}}_2 = m_t$ gdzie $Y_2 \subset Y$.
2. Dla wybranych podzbiorów wyznaczamy współczynniki komplikacji.

3. Potencjała zależności

4. Wybieramy liczy PTP

5. Jeżeli wś pod uw komplik przyjm

Przykład 1:

Niech z stwiona jest

0

1

2

3

4

5

6

7

8

9

10

11

12

Założm
o następuj
podziału P
i transkod
Możliw
transkoder

Tablica 1

Y_0

$Y_4, Y_3, Y_2=4$

$Y_2, Y_1, Y_0=5$

oder wtedy,

pełnione są

(1)

(2)

(3)

liczba wejść

(3) to

(4)

era wynosi

est zmniejsz-

n. Chcemy

ca od liczby

jemy się na

PTP, który

ablica PTP

mnijszych

racji

$=m_t$ gdzie

$z=m_t$

- Potencjalnych rozwiązań należy szukać wśród zbiorów spełniających następujące zależności:

$$w_k(Y_2) < k_t; \quad \overline{\overline{Y_2}} > [\lg_2(w_k(Y_2))]$$

- Wybieramy najlepsze rozwiązanie (maksymalne ograniczenie liczby wyjść podtablicy PTP) szukając wartości maksymalnej następującego wyrażenia

$$\overline{\overline{Y_2}} - [\lg_2(w_k(Y_2))] \quad (5)$$

- Jeżeli wśród podzbiorów o mocy m_t zbioru Y rozwiązanie nie istnieje, to bierzemy pod uwagę podzbiory o mocy $m_t - 1$; ponownie wyznaczmy współczynniki komplikacji itd. (ostatecznie wybieramy rozwiązanie, dla którego wyrażenie (5) przyjmuje wartość maksymalną).

Przykład 1:

Niech zespół funkcji $Y = \{Y_{11}, \dots, Y_1, Y_0\}$ będzie opisany PTP (tab. 2 — przedstawiona jest tylko część opisująca wyjścia).

Tablica 2

PTP opisująca zespół funkcji z przykładu 1

	H	H	H	H	H	H	H	H	H	H	H	H
	Y11	Y10	Y9	Y8	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0	A	*	*	*	*	*	A	*	A	*	*	*
1	*	*	A	*	*	A	*	*	*	*	*	A
2	*	A	*	*	*	*	A	A	*	*	*	A
3	A	*	*	*	*	*	*	A	*	*	*	A
4	*	A	*	*	*	A	A	*	A	*	A	A
5	*	*	A	*	*	*	*	*	A	*	A	*
6	*	*	*	A	*	*	A	*	*	A	*	*
7	*	*	*	*	A	*	*	*	*	A	A	*
8	*	*	*	*	*	A	A	*	*	*	*	*
9	*	*	A	*	*	*	*	*	A	*	*	*
10	A	*	*	*	*	A	A	*	*	*	A	A
11	A	*	*	*	A	*	*	*	*	A	*	A
12	*	*	A	*	A	*	A	*	*	A	A	*

Założmy, że chcemy zmniejszyć liczbę wyjść PTP stosując zewnętrzny transkoder o następujących parametrach: $n_t=5$, $m_t=8$, $k_t=32$. Po zastosowaniu algorytmu podziału PTP na podtablice PTP, przeznaczone do realizacji na elementach PLD i transkoderze, możliwy jest podział tablicy i realizacja przedstawiona na rys. 3.

Możliwy jest również podział PTP i realizacja wykorzystująca „mniejszy” transkoder o parametrach $n_t=3$, $m_t=8$, $k_t=8$ przedstawiona na rysunku 4.

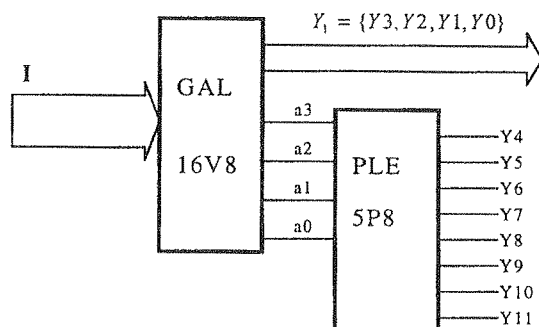
Podtablica PTP

	H	H	H	H	H	H	H	H
	Y3	Y2	Y1	Y0	a3	a2	a1	a0
0	A	*	*	*	*	*	*	A
1	*	*	*	A	*	*	A	*
2	*	*	*	A	*	*	A	A
3	*	*	*	A	*	A	*	*
4	A	*	A	A	*	A	*	A
5	A	*	A	*	*	A	A	*
6	*	A	*	*	*	A	A	A
7	*	A	A	*	A	*	*	*
8	*	*	*	*	A	*	*	A
9	A	*	*	*	*	A	A	*
10	*	*	A	A	A	*	A	*
11	*	A	*	A	A	*	A	A
12	*	A	A	*	A	A	*	*

Tablica transkodera

					H	H	H	H	H	H	H	H
	a3	a2	a1	a0	Y11	Y10	Y9	Y8	Y7	Y6	Y5	Y4
0	L	L	L	H	A	*	*	*	*	*	A	*
1	L	L	H	L	*	*	A	*	*	A	*	*
2	L	L	H	H	*	A	*	*	*	*	A	A
3	L	H	L	L	A	*	*	*	*	*	*	A
4	L	H	L	H	*	A	*	*	*	A	A	*
5	L	H	H	L	*	*	A	*	*	*	*	*
6	L	H	H	H	*	*	*	A	*	*	A	*
7	H	L	L	L	*	*	*	*	A	*	*	*
8	H	L	L	H	*	*	*	*	*	A	A	*
9					*	*	A	*	*	*	*	*
10	H	L	H	L	A	*	*	*	*	A	A	*
11	H	L	H	H	A	*	*	*	A	*	*	*
12	H	H	L	L	*	*	A	*	A	*	A	*

Uwaga: wiersze 5 i 9 (tab. 6) PTP (oznaczone „*”) są identyczne biorąc pod uwagę zmienne Y11, Y10, Y9, Y8, Y7, Y6, Y5 stąd wiersz 9 w tablicy transkodera można wyeliminować



Rys. 3. Tablice po podziale PTP oraz struktura podzielonego układu

Transkoderem może być nie tylko struktura PLE, ale również PLA lub PAL. W tym przypadku bardzo dużą rolę odgrywa parametr k_t , który charakteryzuje zasoby wewnętrzne transkodera. W przypadku realizacji transkodera na strukturze PLA współczynnik $k_t = f(n_t, k_{PLA})$ gdzie n_t jest założoną liczbą wejść transkodera, mniejszą od liczby dostępnych wejść układu, natomiast k_{PLA} jest liczbą termów zawartych w strukturze. Oczywiście jest fakt, że liczba wyjść transkodera $m_t > n_t$ ponieważ tylko wtedy możemy mówić o ograniczeniu liczby wyjść podtablicy PTP powstającej po podziale. Można wykazać, że współczynnik k_t transkodera opartego na strukturze PLA wynosi $k_t = \min(k_{PLA}, 2^{n_t})$.

W przypadku realizacji transkodera w oparciu o strukturę typu PAL wyznaczenie współczynnika k_t jest niemożliwe ponieważ jest on funkcją nie tylko parametrów układu. W strukturze PAL do poszczególnych wyjść dołączona jest ściśle określona

Podtablica PTP

	H	H	H	H	H	H	H	H
	Y11	Y10	Y9	Y8	Y7	Y6	Y5	Y4
0	A	*	*	*	*	*	A	*
1	*	*	A	*	*	A	*	*
2	*	*	A	*	*	*	A	A
3	A	*	*	*	*	*	*	A
4	*	A	*	*	*	A	A	*
5	*	*	A	*	*	*	*	*
6	*	*	*	A	*	*	A	*
7	*	*	*	*	A	*	*	*
8	*	*	*	*	*	A	A	*
9	*	*	*	*	*	*	*	*
10	A	*	*	*	*	A	A	*
11	A	*	*	*	A	*	*	*
12	*	*	A	*	A	*	A	*

liczba termów realizowanych poszczególnych wyjść wykorzystujących ekspansję transkodera problem. możliwa jest powód...

Podtablica PTP

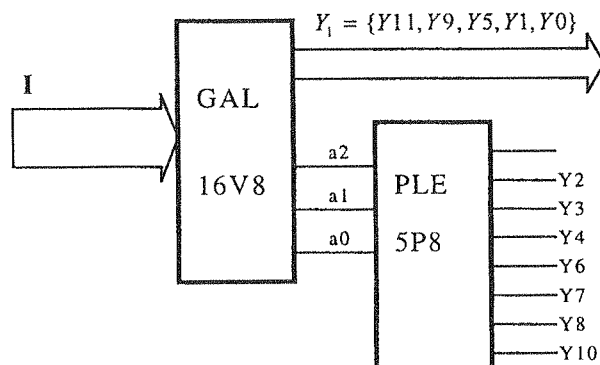
H	H	H
Y6	Y5	Y4
*	A	*
A	*	*
*	A	A
*	*	A
A	A	*
*	*	*
*	A	*
*	*	*
A	A	*
*	*	*
A	A	*
*	*	*
*	A	*

Y11, Y10, Y9,

	H	H	H	H	H	H	H	H
	Y11	Y9	Y5	Y1	Y0	a2	a1	a0
0	A	*	A	*	*	*	*	A
1	*	A	*	*	A	*	A	*
2	*	*	A	*	A	*	A	A
3	A	*	*	*	A	A	*	*
4	*	*	A	A	A	A	*	A
5	*	A	*	A	*	*	*	A
6	*	*	A	*	*	A	A	*
7	*	*	*	A	*	A	A	A
8	*	*	A	*	*	*	A	*
9	*	A	*	*	*	*	*	A
10	A	*	A	A	A	*	A	*
11	A	*	*	*	A	A	A	A
12	*	A	A	A	*	A	A	A

Tablica transkodera

			H	H	H	H	H	H	H	
	a2	a1	a0	Y10	Y8	Y7	Y6	Y4	Y3	Y2
0	L	L	H	*	*	*	*	*	A	*
1	L	H	L	*	*	*	A	*	*	*
2	L	H	H	A	*	*	*	A	*	*
3	H	L	L	*	*	*	*	A	*	*
4	H	L	H	A	*	*	A	*	A	*
6	H	H	L	*	A	*	*	*	*	A
7	H	H	H	*	*	A	*	*	*	A



Rys. 4. Podział PTP i realizacja wykorzystująca 3-we transkoder

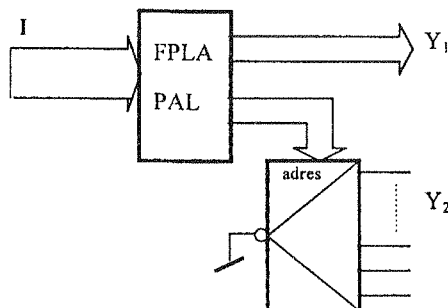
lub PAL. Charakterystyka struktury transkodera, oparta o termów era $m_i > n_i$ ablicy PTP a opartego

znaczenie parametrów określona

liczba termów. Fakt ten sprawia, że pojemność transkodera typu PAL jest zależna od realizowanych w nim funkcji tzn. liczby aktywnych stanów (A) występujących dla poszczególnych wyjść. Jeżeli liczba ta jest większa od k_{PAL} to konieczne jest wykorzystanie dodatkowych wyjść wprowadzając sprzężenia zwrotne lub stosując ekspansję liczby termów [3, 8]. Fakt ten znacznie komplikuje proces realizacji transkodera w postaci struktury PAL. Chciałbym zwrócić uwagę jeszcze na jeden problem. Okazuje się, że przy realizacji transkodera w postaci struktury PAL, możliwa jest minimalizacja poszczególnych funkcji realizowanych w transkoderze, co powoduje zmniejszenie liczby termów potrzebnych do realizacji funkcji wyjściowych.

4. PODZIAŁ WYJŚĆ WYKORZYSTUJĄCY ZEWNĘTRZNY DEMULTIPLEKSER (DEKODER)

W szczególnym przypadku transkoderem może być demultiplekser lub dekodery [4]. Sposób podziału wyjść wykorzystujący tą metodę przedstawiony jest na rysunku 5.



Rys. 5. Idea podziału wyjść wykorzystująca zewnętrzny demultiplekser

W celu przedstawienia algorytmu podziału wyjść wykorzystującego zewnętrzny demultiplekser (dekoder) konieczne jest wprowadzenie niezbędnych definicji.

Definicja 2:

Wyjścia Y_i , Y_k nazywamy rozłącznymi wtedy i tylko wtedy, gdy przyjmują stany aktywne dla różnych termów (tzn.: symbole A występują w różnych wierszach); (patrz tab. 3).

Tablica 3

Przykładowa PTP z wyjściami rozłącznymi i dominującymi

	Y3	Y2	Y1	Y0
	*	*	A	*
	A	A	*	*
	A	*	*	A
	*	*	A	*

wyjście Y3 pokrywa 1, 2 i 3 term

wyjście Y1 pokrywa 0 i 4 term

wyjście Y3 dominuje na wyjściu Y2 i Y0

Y3, Y1 — wyjścia rozłączne

Y2, Y1, Y0 — wyjścia wzajemnie rozłączne

Wprowadźmy dodatkowo pojęcie pokrywania termów przez wyjścia.

Definicja 3:

Mówimy, że wyjście Y_i pokrywa term k -ty wtedy, gdy dla k -tego termu wyjście Y_i przyjmuje stan aktywny (A); (patrz tab. 3).

Przyjmijmy następujące oznaczenia: m — całkowita liczba wyjść PTP, p — liczba wyjść PTP wzajemnie rozłącznych, n — liczba wejść adresowych demultipleksa (dekodera).

Liczbę
 $a_1 2^1 + a_0 2^0$
kombinac
krytych p
Jeżeli

stąd

Zastos
PTP wted
od liczby

Zależn
wykorzyst
rozłączne

Definicja

Wyjście
termów, c
dla który
symbole A
występują

Algorytm

1. Porządkuj wyjścia w sposób zgodny z ich dominacją.
2. Zaznacz wyjścia, które nie są pokrywane przez żadne z poprzednich.
3. Spośród wyjść rozłącznych wybierz te, które dominują nad pozostałymi.
4. Jeżeli liczba wyjść rozłącznych p jest równa liczbie wyjść m , to dekodery są realizowane przez demultipleksy.

NY

lub dekodery
ony jest na

Liczbę dziesiętną p można przedstawić następująco $p_{10} = a_{n-1}2^{n-1} + \dots + a_12^1 + a_02^0$. Załóżmy, że wyjścia rozłączne nie pokrywają wszystkich termów stąd kombinacja $a_{n-1} \dots a_1 a_0 = 0 \dots 00$ przeznaczona jest dla pozostałych termów nie pokrytych przez wyjścia należące do zbioru wyjść wzajemnie rozłącznych.

Jeżeli

$$p \in \langle 2, 3 \rangle \Rightarrow n = 2;$$

$$p \in \langle 4, 7 \rangle \Rightarrow n = 3;$$

$$p \in \langle 8, 15 \rangle \Rightarrow n = 4$$

itd.

stąd

$$n = [\lg_2 p] + 1.$$

Zastosowanie zewnętrznego demultipleksera (dekodera) zmniejsza liczbę wyjść PTP wtedy, gdy spełniona jest zależność: $m > m - p + n$ (liczba wyjść PTP jest większa od liczby wyjść podtablicy PTP powstającej po podziale), tzn.

$$p > [\lg_2 p] + 1. \quad (6)$$

zewnętrzny
inicji.

Zależność (6) jest spełniona dla $p \geq 3$ stąd liczbę wyjść PTP można zmniejszyć wykorzystując demultipleksery (dekodery) wtedy, gdy istnieją co najmniej 3 wyjścia rozłączne. Wprowadźmy dodatkowo kolejne pojęcie wyjść dominujących.

jmują stany
wierszach);

Definicja 4:

Wyjście Y_i nazywamy dominującym nad wyjściem Y_k wtedy i tylko wtedy, gdy zbiór termów, dla których wyjście Y_k przyjmuje stan aktywny jest podzbiorem zbioru termów, dla których wyjście Y_i przyjmuje stan aktywny. (tzn.: wiersze, dla których występują symbole A w kolumnie odpowiadającej wyjściu Y_k są podzbiorem wierszy, dla których występują symbole A w kolumnie odpowiadającej wyjściu Y_i ; (patrz tab. 3).

Tablica 3

rm

tem Y2 i Y0

nie rozłączne

.

u wyjście Y_i

, p — liczba
multipleksa

Algorytm podziału PTP na podtablice PTP przeznaczone do realizacji na elementach PLD i demultipleksery (dekodery)

1. Porządkujemy kolumny w obszarze PTP opisującej wyjścia według liczby aktywnych stanów wyjściowych (symboli A).
2. Zaznaczamy wszystkie wyjścia dominujące (wyjścia te nie są uwzględniane w dalszej części algorytmu).
3. Spośród pozostałych wyjść wybieramy najliczniejszy podzbiór wyjść wzajemnie rozłącznych.
4. Jeżeli

p — moc zbioru wyjść wzajemnie rozłącznych i $p \geq 3$ to układ opisany PTP można zrealizować na układach PLD o mniejszej liczbie wyjść i demultipleksery (dekodery) o n -wejściach adresowych gdzie:

$$n > \lg_2 p.$$

T a b l i c a 4

PTP opisująca zespół funkcji $Y = \{Y_{10}, \dots, Y_1, Y_0\}$ z przykładu 2

	L	L	L	L	L	L	L	L	L	L	L
	Y ₁₀	Y ₉	Y ₈	Y ₇	Y ₆	Y ₅	Y ₄	Y ₃	Y ₂	Y ₁	Y ₀
0	*	*	*	*	*	*	*	A	*	*	A
1	*	*	*	*	*	*	A	*	*	A	*
2	A	*	*	*	*	*	*	*	A	*	*
3	*	*	*	*	A	*	*	*	*	*	A
4	*	*	*	*	*	A	*	*	*	*	*
5	*	*	*	A	*	*	*	*	*	*	*
6	*	*	A	*	*	*	*	*	*	*	*
7	A	A	*	*	*	A	*	*	A	A	*
8	*	A	*	*	*	*	*	*	*	*	*
9	*	A	*	*	*	A	*	*	*	*	*
10	*	*	*	*	*	*	*	*	A	*	*

T a b l i c a 5

PTP po uporządkowaniu kolumn według liczby aktywnych stanów wyjściowych (symboli A) i wyszukaniu zbioru wyjść dominujących, oznaczonych symbolem „x”

	L	L	L	L	L	L	L	L	L	L	L
	Y ₉	Y ₅	Y ₂	Y ₁₀	Y ₁	Y ₀	Y ₈	Y ₇	Y ₆	Y ₄	Y ₃
0	*	*	*	*	*	A	*	*	*	*	A
1	*	*	*	*	A	*	*	*	*	A	*
2	*	*	A	A	*	*	*	*	*	*	*
3	*	*	*	*	*	A	*	*	A	*	*
4	*	A	*	*	*	*	*	*	*	*	*
5	*	*	*	*	*	*	*	A	*	*	*
6	*	*	*	*	*	*	A	*	*	*	*
7	A	A	A	A	A	*	*	*	*	*	*
8	A	*	*	*	*	*	*	*	*	*	*
9	A	A	*	*	*	*	*	*	*	*	*
10	*	*	A	*	*	*	*	*	*	*	*

x x x

Y₂, Y₁, Y₀ — wyjścia dominujące,

PTP po uporządkowaniu kolumn (wyjścia dominujące, reszta wyjść) i wyszukaniu zbioru wyjść wzajemnie rozłącznych oznaczonych symbolem „#”

	L	L	L	L	L	L	L	L	L	L	L
	Y2	Y1	Y0	Y90	Y5	Y10	Y8	Y7	Y6	Y4	Y3
0	*	*	A	*	*	*	*	*	*	*	A
1	*	A	*	*	*	*	*	*	*	A	*
2	A	*	*	*	*	A	*	*	*	*	*
3	*	*	A	*	*	*	*	*	A	*	*
4	*	*	*	*	A	*	*	*	*	*	*
5	*	*	*	*	*	*	*	A	*	*	*
6	*	*	*	*	*	*	A	*	*	*	*
7	A	A	*	A	A	A	*	*	*	*	*
8	*	*	*	A	*	*	*	*	*	*	*
9	*	*	*	A	A	*	*	*	*	*	*
10	A	*	*	*	*	*	*	*	*	*	*
	x	x	x	#			#	#	#	#	#

$\{Y_9, Y_8, Y_7, Y_6, Y_4, Y_3\}$ — najliczniejszy podzbiór wyjść wzajemnie rozłącznych

PTP po uporządkowaniu kolumn (wyjścia dominujące, wyjścia nie należące do podzbioru wyjść dominujących i wzajemnie rozłącznych, wyjścia wzajemnie rozłączne) i przypisaniu kodów poszczególnym wyjściom wzajemnie rozłącznym

	L	L	L	L	L	L	L	L	L	L	L
	Y2	Y1	Y0	Y10	Y5	Y9	Y8	Y7	Y6	Y4	Y3
0	*	*	A	*	*	*	*	*	*	*	A
1	*	A	*	*	*	*	*	*	*	A	*
2	A	*	*	A	*	*	*	*	*	*	*
3	*	*	A	*	*	*	*	*	A	*	*
4	*	*	*	*	A	*	*	*	*	*	*
5	*	*	*	*	*	*	*	A	*	*	*
6	*	*	*	*	*	*	A	*	*	*	*
7	A	A	*	A	A	A	*	*	*	*	*
8	*	*	*	*	*	A	*	*	*	*	*
9	*	*	*	*	A	A	*	*	*	*	*
10	A	*	*	*	*	*	*	*	*	*	*
					a2	#	#	#	#	#	#
					a1	H	H	H	L	L	L
					a0	L	H	L	H	L	H

Przykład 2:

Niech zespół funkcji $Y = \{Y_{10}, \dots, Y_1, Y_0\}$ będzie opisany PTP (patrz tab. 4).

Poszczególne etapy realizacji algorytmu przedstawiono w tablicach 5, 6, 7.

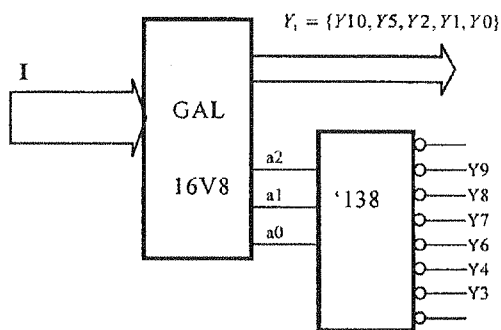
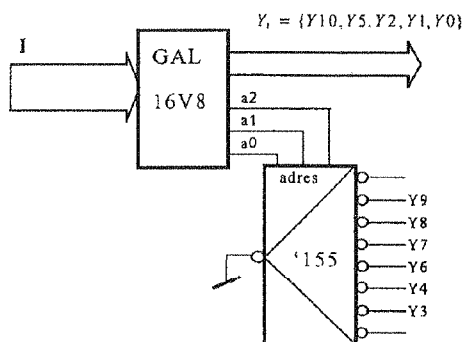
Po zastosowaniu algorytmu podziału PTP na podtablice PTP, przeznaczone do realizacji na elementach PLD i demultiplekserze (dekoderze), możliwy jest podział PTP i realizacja przedstawiona na rys. 6.

Podtablica PTP

	L	L	L	L	L	H	H	H
	Y2	Y1	Y0	Y10	Y5	a2	a1	a0
0	*	*	A	*	*	*	*	A
1	*	A	*	*	*	*	A	*
2	A	*	*	A	*	*	*	*
3	*	*	A	*	*	*	A	A
4	*	*	*	*	A	*	*	*
5	*	*	*	*	*	A	*	*
6	*	*	*	*	*	A	*	A
7	A	A	*	A	A	A	A	*
8	*	*	*	*	*	A	A	*
9	*	*	*	*	A	A	A	*
10	A	*	*	*	*	*	*	*

Tablica transkodera

				L	L	L	L	L	L	L
	a2	a1	a0		Y3	Y4	Y6	Y7	Y8	Y9
0	L	L	L	A	*	*	*	*	*	*
1	L	L	H	*	A	*	*	*	*	*
2	L	H	L	*	*	A	*	*	*	*
3	L	H	H	*	*	*	A	*	*	*
4	H	L	L	*	*	*	*	A	*	*
5	H	H	L	*	*	*	*	*	A	*
6	H	H	L	*	*	*	*	*	*	A
7	H	H	H	*	*	*	*	*	*	A



Rys. 6. Tablice po podziale PTP oraz przykładowa realizacja układu po podziale

Warto zwrócić uwagę jeszcze na jeden problem. Jeżeli liczba wyjść wzajemnie rozłącznych spełnia następujące zależności $p \geq 3$ i $p < d$ (d — liczba wyjść niepełnego dekodera) to układ kombinacyjny opisany PTP można zrealizować na strukturze PLD oraz niepełnym dekodzie o liczbie wejść n równej $n > [\lg_2 p] + 1$. Oczywiście oprócz wszelkiego typu scalonych demultiplekserów i dekodów możliwa jest również realizacja takiego układu wykorzystująca struktury programowalne. Układami, które najlepiej nadają się do takich zastosowań są struktury typu PAL.

W procesie projektowania odpowiedniej struktury. F. Decomp p. struktury p. układu PTP. struktury p. poszczególnego układu. wzajemnie.

W wyniku mogą znaleźć wych na baz. po podziale. turach FOC. takich ukł. w artykule. systemów. programow. nie przydat.

1. Altera
2. R.L. A. s. Symposium
3. M. B. o. l. Company
4. S.D. B. r. Kluwer A.
5. H.A. C. u. Jersey, T.
6. D. K. a. n. cji, 1998,
7. D. K. a. n. wybranych
8. D. K. a. n. w strukt.
9. T. Ł. u. b. October 1
10. W. M. a. j. logiczne w
11. Philips S.

PODSUMOWANIE

W procesie syntezy układów cyfrowych w oparciu o struktury programowalne projektant napotyka na wiele problemów między innymi na problem doboru odpowiedniej struktury programowalnej, czy też podziału projektu na poszczególne struktury. Przedstawione w pracy algorytmy zostały zaimplementowane w systemie Decom przeznaczonym do podziału projektowanego układu na poszczególne struktury programowalne. W części dotyczącej podziału wyjść projektant po opisanu układu PTP lub stworzeniu pliku w formacie PLA, wybiera, korzystając z biblioteki, struktury programowalne. Program Decom próbuje podzielić i przedstawić opis poszczególnych podukładów, podpowiadając równocześnie parametry projektowanego układu, które mogą usprawnić wybór odpowiedniej struktury np. liczbę wyjść wzajemnie rozłącznych itp.

W wyniku licznych eksperymentów okazało się, że przedstawione algorytmy mogą znaleźć wykorzystanie przede wszystkim w procesie syntezy układów cyfrowych na bazie prostych struktur PLD. Zwiększenie sumarycznej liczby wyjść układu po podziale sprawia, że nie znajdują one zastosowania w procesie syntezy w strukturach FOGA, opartych na blokach logicznych z małą liczbą wyjść. W przypadku takich układów najczęściej stosowany jest prosty podział wyjść. Przedstawione w artykule algorytmy mogą stanowić jednak cenne uzupełnienie komercyjnych systemów wspomagających projektowanie układów cyfrowych na bazie struktur programowalnych takich jak ABEL, CUPL, PALASM, SNAP itp., będąc szczególnie przydatnymi w przypadku syntezy różnorodnych dekodów adresów.

BIBLIOGRAFIA

1. Altera: *Applications Handbook*, 1992
2. R.L. Ashenhurst: *The decomposition of switching functions*. Proceedings of an International Symposium on the Theory of Switching, April 1957
3. M. Bolton: *Digital Systems Design with Programmable Logic*. Addison-Wesley Publishing Company, 1990, pp. 133–140
4. S.D. Brown, R.J. Francis, J. Rose, Z.G. Vranesic: *Field Programmable Gate Arrays*. Kluwer Academic Publishers, Boston/London/Dordrecht 1993, pp. 45–86
5. H.A. Curtis: *The Design of Switching Circuits*. D. van Nostrand Company, Inc., Princeton, New Jersey, Toronto, New York, 1962
6. D. Kania: *Coding capacity of Programmable Transcoder*. Kwartalnik Elektroniki i Telekomunikacji, 1998, 44, z. 2, pp. 193–204
7. D. Kania: *Tablicowe metody dekompozycji układów kombinacyjnych. Realizacja tych układów na wybranych strukturach programowalnych*. Rozprawa doktorska, Politechnika Śląska, Gliwice, 1995
8. D. Kania, E. Hryniewicz: *Dekompozycja układów kombinacyjnych pod kątem ich realizacji w strukturach PLD o niewystarczającej liczbie termów*. KST'95, Bydgoszcz, 1995
9. T. Łuba: *Multi-level logic synthesis based on decomposition*. Microprocessors and Microsystems, October 1994, Vol. 18, No. 8, October 1994
10. W. Majewski, T. Łuba, T. Jasiński, B. Zbierzchowski: *Programowalne metody logiczne w syntezie układów cyfrowych*. WKŁ, Warszawa, 1992
11. Philips Semiconductors: *Programmable Logic Devices*. Data Handbook, 1992

D. KANIA

OUTPUT PARTITIONING ALGORITHMS WHICH ENABLE IMPLEMENTATION
OF DIGITAL CIRCUITS IN PLD STRUCTURES

S u m m a r y

Programmable circuits have limited number of outputs. The problem of digital circuit implementation in programmable devices is strictly related to output partitioning. Output partitioning means subdividing a large design into relatively smaller PLDs circuits. This paper presents the algorithms of output partitioning using external transcoder, decoders or demultiplexers. The partitioning algorithms are implemented in Decomp system which has been developed at Silesian Technical University.

Key word: partitioning, decomposition, Programmable Logic Devices

Mo
piezoele

Instytu

O
stosow
przetw
memb
Lamb
wpływ
z wyn

Słowa
brano

1. PI

Czujni
nie w por
takich wie
różnych ga
Wykorzys
np. przy a
wewnętrz
ośrodka.
w układac
nych w r
nikowych
membran
Czujniki z
niskim cz
zmiany jec

Model membranowego czujnika z przetwornikami piezoelektrycznymi do generacji i detekcji ultradźwiękowych fal płytowych typu Lamba

JACEK GOŁĘBIEWSKI

Instytut Elektrotechniki Teoretycznej, Metrologii i Materiałoznawstwa, Politechnika Łódzka

Otrzymano 1999.01.06

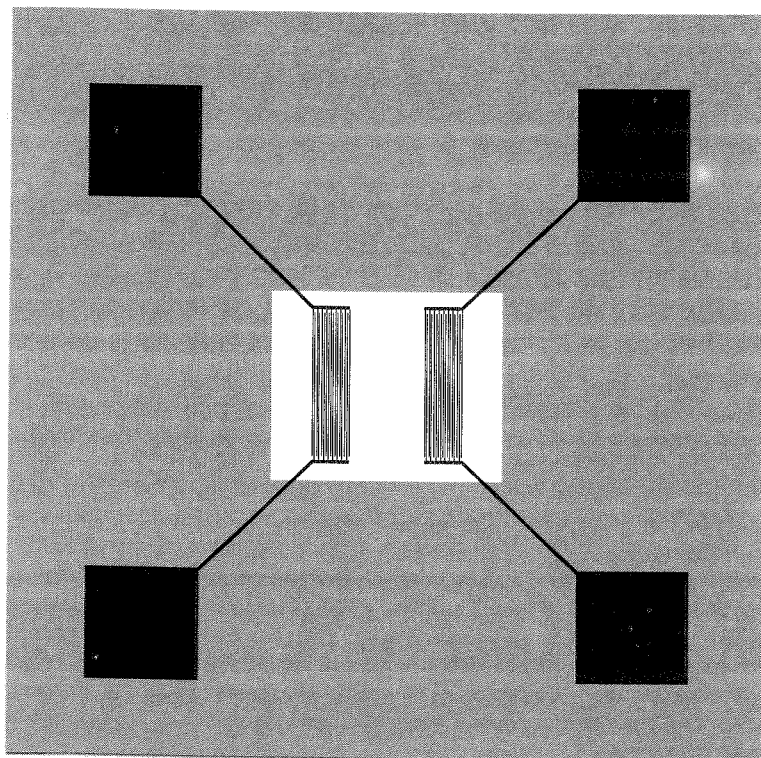
Autoryzowano 1999.05.06

Omówiono konstrukcję międzypalczastych przetworników piezoelektrycznych (IDT) stosowanych w czujnikach z falami Lamba (FPW). Przedstawiono układ zastępczy tego przetwornika stosowanego do generacji i detekcji fali ultradźwiękowej. Rozważono model membranowego czujnika z przetwornikami piezoelektrycznymi do generacji i detekcji fali Lamba. Przeanalizowano charakterystykę częstotliwościową modelu z uwzględnieniem wpływu jego parametrów. Porównano wyniki obliczeń uzyskane dla modelu przetwornika z wynikami pomiarów.

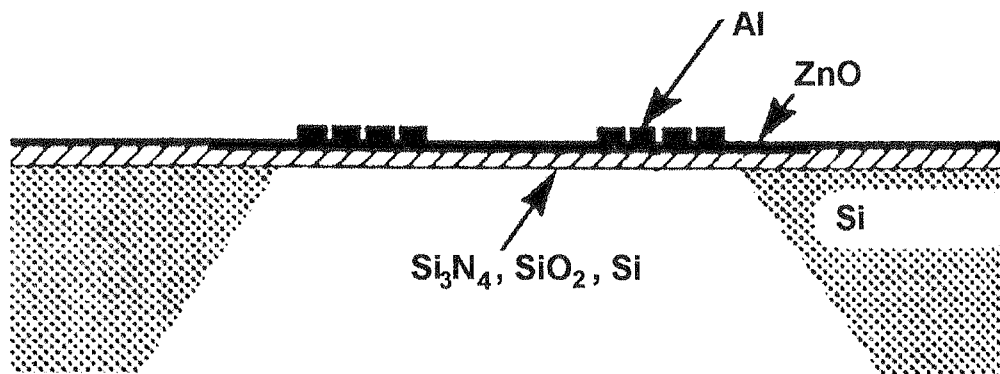
Słowa kluczowe: fale ultradźwiękowe, piezoelektryczne przetworniki międzypalczaste, membranowe czujniki z falami ultradźwiękowymi, modele czujników.

1. PIEZOELEKTRYCZNE PRZETWORNIKI MIĘDZYPALCZASTE

Czujniki z ultradźwiękowymi falami płytowymi typu Lamba znajdują zastosowanie w pomiarach wielkości fizycznych, chemicznych, biologicznych. Dotyczy to takich wielkości jak: ciśnienia, przyspieszenia, lepkość i gęstość cieczy, stężenia różnych gazów, wykrywanie i określanie zawartości białek, bakterii w roztworach [1]. Wykorzystuje się tutaj zjawiska zmiennego powierzchniowego obciążenia masowego np. przy absorpcji gazów na powierzchni czujnika, zjawiska zmiennych napreżeń wewnętrznych w płycie np. przy pomiarach ciśnienia, zjawiska lepkosprężystości ośrodka. Fale ultradźwiękowe typu płytowego mogą być również zastosowane w układach mikrowykonawczych np. miniaturowych pompach, mikserach stosowanych w medycynie, biologii. Zaadoptowanie technologii elementów półprzewodnikowych na bazie krzemu pozwala na wykonanie bardzo cienkich prostokątnych membran o grubościach rzędu kilku μm i pozostałych wymiarach rzędu 1–5 mm. Czujniki z takimi membranami i falą Lamba charakteryzują się małymi wymiarami, niskim częstotliwościami pracy (kilka MHz) oraz dużą czułością masową tzn. małe zmiany jednostkowe masy obciążającej membranę ($\Delta m/m$) powodują względnie duże



Rys. 1a. Widok membrany czujnika z przetwornikami piezoelektrycznymi (IDT)



Rys. 1b. Przekrój poprzeczny membrany czujnika gdzie: Al — aluminiowe elektrody; ZnO — warstwa piezoelektryczna; Si₃N₄, SiO₂, Si — warstwa z azotku krzemu, tlenku krzemu lub krzemu monokrystalicznego

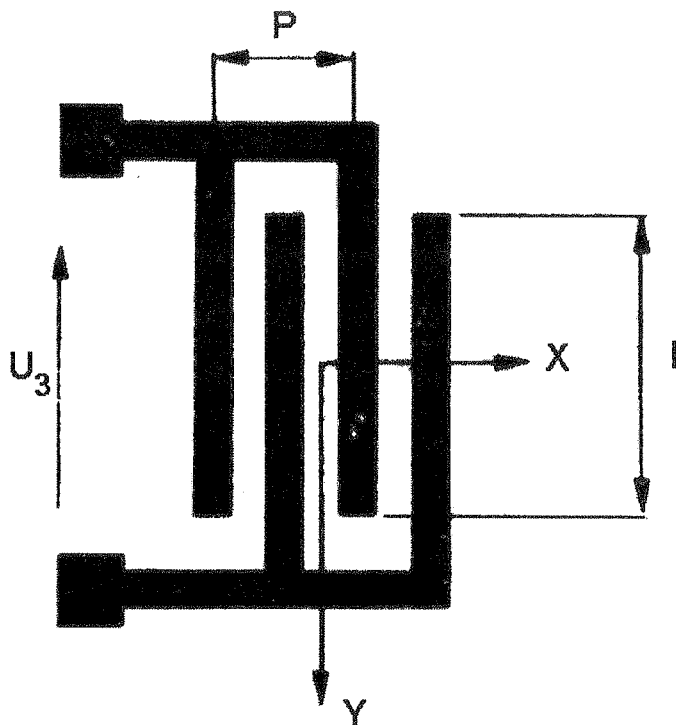
zmiany częstotliwości sygnału wyjściowego ($\Phi f/f$) [10]. Znajduje to zastosowanie w detektorach masy np. czujnikach stężenia gazu.

W miniaturowych membranowych czujnikach do generowania i detekcji fali ultradźwiękowej typu Lamba (FPW — *flexural plate wave*) wykorzystuje się piezoelektryczne przetworniki międzypalczaste (grzebieniowe) (IDT — *interdigital transducer*) [1, 2]. Na rys. 1a pokazano widok membrany z dwoma symetrycznymi

Rys. 2. Układ
gdzie: P —

przetwornikami IDT umieszczonymi na jej powierzchni. Na rys. 1b pokazano przekrój poprzeczny membrany z przetwornikami piezoelektrycznymi umieszczonymi na cienkiej warstwie sprężystej wykonanej z azotku krzemu, tlenku krzemu lub mnokrystalicznego krzemu.

Na rys. 2 przedstawiono układ przetwornika z dwoma parami symetrycznych elektrod umieszczonych na powierzchni cienkiej piezoelektrycznej membrany. Napięcie sinusoidalnie zmienne $u_3 = U_0 \exp(j\omega t)$ (U_0 — amplituda, ω — pulsacja napięcia) przyłożone do elektrod przetwornika wytworzy okresowo zmieniające się pole elektryczne (rys. 3). Sprężenie między polem elektrycznym i falą FPW jest możliwe, gdy składowe pola elektrycznego sprzęgają się poprzez efekt piezoelektryczny ze składowymi przemieszczeniami fali. Efektywność tego sprzężenia jest tym większa, im większy jest współczynnik sprzężenia elektromechanicznego k_p i im większa jest zgodność rozkładów pól elektrycznych wytwarzanych w przetwornikach międzypalczastych i przez falę ultradźwiękową. Drgania mechaniczne wytworzone w obszarze jednej elektrody będą rozchodzić się wzdłuż osi X i po przemieszczeniu się na odległość $0,5 P$ (rys. 2) zmienią swoją fazę o π . Jednocześnie napięcie wejściowe zmieni swoją fazę o π i pole elektryczne w obszarze następnej elektrody będzie w takiej samej fazie względem fali ultradźwiękowej. Taki układ międzypalczastych elektrod jest przetwornikiem dwukierunkowym. Napięcie U_3 przyłożone do elektrod



Rys. 2. Układ elektrod piezoelektrycznego przetwornika międzypalczastego na powierzchni membrany gdzie: P — okres przetwornika, l — długość elektrod, U_3 — napięcie zasilające, X i Y — układ osi współrzędnych

przetwornika IDT wytworzy składową pola elektrycznego E_x (zgodną z osią X) i składową E_z (zgodną z osią Z — prostopadłą do powierzchni membrany).

Zależności pomiędzy podstawowymi wielkościami mechanicznymi i elektrycznymi opisującymi stan ciała piezoelektrycznego są określone równaniami macierzowymi:

$$\begin{aligned} [T] &= [c^E][S] - [e][E] \\ [D] &= [e][S] + [\epsilon^S][E] \end{aligned} \quad (1)$$

Macierze D , E , są macierzami kolumnowymi opisującymi wektory indukcji elektrycznej i natężenia pola elektrycznego o składowych podanych we współrzędnych prostokątnych x , y , z .

Macierze T , S są macierzami kolumnowymi opisującymi tensory naprężeń i odkształceń, indeksy od 1 do 3 odpowiadają składowym normalnym natomiast indeksy od 4 do 6 odpowiadają składowym równoległym do danej płaszczyzny. Macierze współczynników $[c]$, $[e]$ i $[\epsilon]$ są macierzami współczynników sprężystości, współczynników piezoelektrycznych i współczynników przenikalności dielektrycznej. Macierz $[c]$ jest macierzą symetryczną. Oznaczenia wprowadzone w postaci wskaźników E i S informują, że współczynniki $[c^E]$ zostały wyznaczone przy zerowym polu elektrycznym, a współczynniki $[\epsilon^S]$ przy zerowych odkształceniach.

Równania określające rozchodzenie się fal ultradźwiękowych w piezoelektrykach są opisane równaniami pól akustycznych (ultradźwiękowych) oraz równaniami Maxwella:

$$\begin{aligned} -\nabla \times \mathbf{E} &= \frac{\partial \mathbf{B}}{\partial t} \\ \nabla \times \mathbf{H} &= \frac{\partial \mathbf{D}}{\partial t} \\ \nabla \times \mathbf{T} &= \rho \frac{\partial \mathbf{V}}{\partial t} \\ \nabla_s \times \mathbf{V} &= \frac{\partial \mathbf{S}}{\partial t} \end{aligned} \quad (2)$$

gdzie: $\mathbf{E}$, $\mathbf{H}$ — wektory natężenia pola elektrycznego i magnetycznego, $\mathbf{D}$, $\mathbf{B}$ — wektory indukcji elektrycznej i magnetycznej, $\mathbf{T}$ — wektor naprężeń, $\mathbf{S}$ — wektor odkształceń, ρ — gęstość materiału, $\mathbf{V}$ — wektor prędkości.

Powyższe równania mogą zostać uproszczone dla przypadku fali ultradźwiękowej rozchodzącej się w piezoelektryku, ponieważ długość fali jest znacznie większa od długości fali elektromagnetycznej, jak również może być pominięty wpływ pola magnetycznego. Przy uwzględnieniu, że prędkość fali FPW jest znacznie mniejsza od prędkości fali elektromagnetycznej oraz przy warunkach brzegowych dotyczących kierunku rozchodzenia się fali, tzn. przy założeniu że grubość membrany d jest znacznie mniejsza od długości fali ($d \ll \lambda$) oraz jeżeli długość elektrod l jest znacznie

większa od wymiarowy zerowych na dem metalic o częstotliw

gdzie: V_p — Oznacza to, Układ równ czonęj (4):

gdzie: $\mathbf{E} = \mathbf{E}$ ryczny, $c = c$ ności dielek osi X .

Badanie elektrod i d za-Christoff

2. SCHE

Na rys. elektrodami równoległe składowa E w membran od szerokoś rolę odgryw Równar

większa od długości fali ($l \gg \lambda$), to przetwornik IDT można traktować jako dwuwymiarowy w układzie współrzędnych X i Z . Analizowano nieobciążoną (przy zerowych naprężeniach) membranę piezoelektryczną jednorodną i izotropową z układem metalicznych elektrod, do których doprowadzono napięcie $u_3 = U_0 \exp(j\omega t)$ o częstotliwości f_0 spełniającej warunek:

$$(1) \quad f_0 = \frac{V_p}{P} \quad (3)$$

gdzie: V_p — prędkość fazowa fali Lamba, $\omega = 2\pi f$, P — okres przetwornika. Oznacza to, że w tym przypadku długość fali λ jest równa okresowi P ($\lambda = P$). Układ równań (1) dla przyjętych założeń można przedstawić w postaci uproszczonej (4):

$$\begin{aligned} T &= c \frac{\partial w}{\partial x} - e \cdot E \\ D &= e \frac{\partial w}{\partial x} + \varepsilon \cdot E \end{aligned} \quad (4)$$

gdzie: $E = E_x$ składowa natężenia pola elektrycznego, e — współczynnik piezoelektryczny, $c = c^E$ — współczynnik sprężystości dla $E = 0$, $\varepsilon = \varepsilon^S$ współczynnik przenikalności dielektrycznej dla $S = 0$, w — przemieszczenie cząstek fali, x — współrzędna dla osi X .

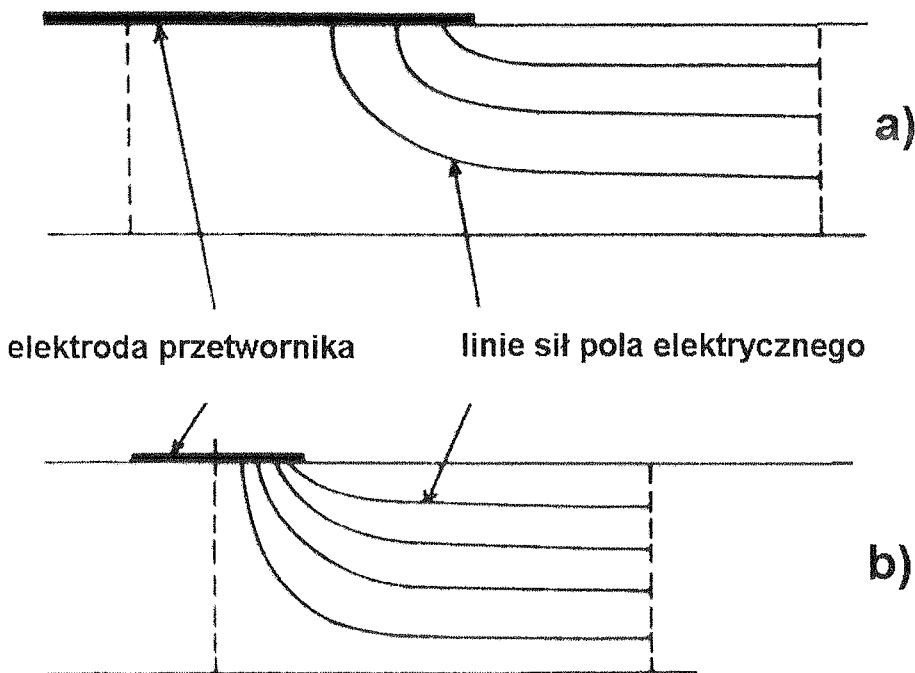
Badanie zależności między rozkładem linii sił pola elektrycznego a konfiguracją elektrod i długością fali było analizowane z zastosowaniem przekształceń Schwarza-Christoffela [4].

2. SCHEMAT ZASTĘPCZY PRZETWORNIKA MIĘDZYPALCZASTEGO

Na rys. 3 pokazano rozkład linii sił pola elektrycznego pomiędzy dwoma elektrodami przetwornika. Można założyć, że linie sił pomiędzy elektrodami są równoległe do powierzchni, natomiast pod elektrodami składową decydującą jest składowa E_z prostopadła do powierzchni. Dla przypadku rozchodzenia się fal FPW w membranie jej grubość d jest znacznie mniejsza od długości fali ($d \ll \lambda$), jak również od szerokości elektrod. Powoduje to, że przy wzbudzaniu i odbiorze fal FPW główną rolę odgrywa składowa pola E_x równoległa do kierunku propagacji fali [3, 5].

Równania ruchu i równanie Poissona dla w/w założeń można napisać w postaci:

$$\begin{aligned} \frac{\partial T}{\partial x} &= \rho \frac{\partial^2 w}{\partial t^2} \\ \frac{\partial D}{\partial x} &= 0. \end{aligned} \quad (5)$$



Rys. 3. Rozkład pola elektrycznego w przetworniku IDT (przekrój prostopadły do powierzchni membrany)
a) dla szerokości elektrody $s=2d$, b) dla $s=2d/3$, gdzie: d — grubość membrany

Korzystając z zależności (4) i (5) otrzymano równanie ruchu:

$$\left(c + \frac{e^2}{\varepsilon}\right) \frac{\partial^2 w}{\partial x^2} - \rho \frac{\partial^2 w}{\partial t^2} = 0. \quad (6)$$

Rozwiązując równanie (6) otrzymano funkcję określającą przemieszczenia elementarne powierzchni membrany:

$$w(x, t) = A_1 e^{-jkx} + A_2 e^{jkx} \quad (7)$$

gdzie: $A_1 = a_1 e^{j\omega t}$, $A_2 = a_2 e^{j\omega t}$, $k = \frac{\omega}{V_p}$, a_1 , a_2 — stałe współczynniki.

Wprowadzając warunki brzegowe dla współrzędnych x_1 i x_2 określających krawędzie elektrod mamy $w=w_1$ dla x_1 oraz $w=w_2$ dla x_2 . Z równania (6) wyznaczono prędkości przemieszczeń w dla x_1 i x_2 :

$$\begin{aligned} \frac{\partial w_1}{\partial t} &= j\omega(A_1 e^{-jkx_1} + A_2 e^{jkx_1}) \\ \frac{\partial w_2}{\partial t} &= j\omega(A_1 e^{-jkx_2} + A_2 e^{jkx_2}). \end{aligned} \quad (8)$$

Z układu równań (8) wyznaczono A_1 i A_2 :

a)

$$\begin{aligned} A_1 &= \frac{1}{2\omega \sin \alpha} \left(\frac{\partial w_2}{\partial t} e^{jkx_1} - \frac{\partial w_1}{\partial t} e^{jkx_2} \right) \\ A_2 &= \frac{1}{2\omega \sin \alpha} \left(\frac{\partial w_1}{\partial t} e^{-jkx_2} - \frac{\partial w_2}{\partial t} e^{-jkx_1} \right) \end{aligned} \quad (9)$$

gdzie: $\alpha = k \times h$, $h = x_2 - x_1$ (dla symetrycznego przetwornika $x_2 - x_1 = P/4$).
Z równań (4), (5) oraz (8) otrzymano zależność:

$$T(x, t) = jk \left(c + \frac{e^2}{\varepsilon} \right) (A_2 e^{jkx} - A_1 e^{-jkx}) - \frac{e}{\varepsilon} D. \quad (10)$$

Korzystając z warunków brzegowych dla $x = x_1$, $x = x_2$ oraz z równania (10) wyznaczono siły działające na krawędziach elektrod przetwornika:

b)

$$\begin{aligned} F_1(t) &= Z \left(\frac{\frac{\partial w_1}{\partial t}}{j \operatorname{tg} \alpha} - \frac{\frac{\partial w_2}{\partial t}}{j \sin \alpha} \right) + \frac{eA}{\varepsilon} D \\ F_2(t) &= Z \left(\frac{\frac{\partial w_1}{\partial t}}{j \sin \alpha} - \frac{\frac{\partial w_2}{\partial t}}{j \operatorname{tg} \alpha} \right) + \frac{eA}{\varepsilon} D \end{aligned} \quad (11)$$

gdzie: $A = d \times 1$, $Z = \rho \times V_p \times A$ jest impedancją falową przetwornika, d — grubość membrany.

(6)

Napięcie u_3 można przedstawić w zależności od natężenia pola E lub od indukcji D (4) a ponieważ okresowo zmienne napięcie będzie wywoływało również okresowe zmiany indukcji można zapisać indukcję D :

a elementar-

$$D = \frac{1}{j\omega A} i_3 \quad (12)$$

(7)

gdzie: i_3 — natężenie prądu.

Wprowadzając zależność (12) do równań (4) otrzymano:

kreślających
wnania (6)

$$u_3 = \frac{e}{j\varepsilon\omega} \frac{\partial w_1}{\partial t} - \frac{e}{j\varepsilon\omega} \frac{\partial w_2}{\partial t} + \frac{1}{j\omega C_0} i_3 \quad (13)$$

gdzie: $C_0 = \frac{\varepsilon A_1}{h}$, $A_1 = d_1 \times l$, d_1 — grubość elektrod. (14)

(8)

Na podstawie równań (11, 13) zwanych równaniami Masona [7] można zbudować równoważny obwód przetwornika piezoelektrycznego w układzie z równoległym polem elektrycznym generującego falę ultradźwiękową typu Lamba.

Równania (11, 13) mogą być zastąpione równoważnym układem równań:

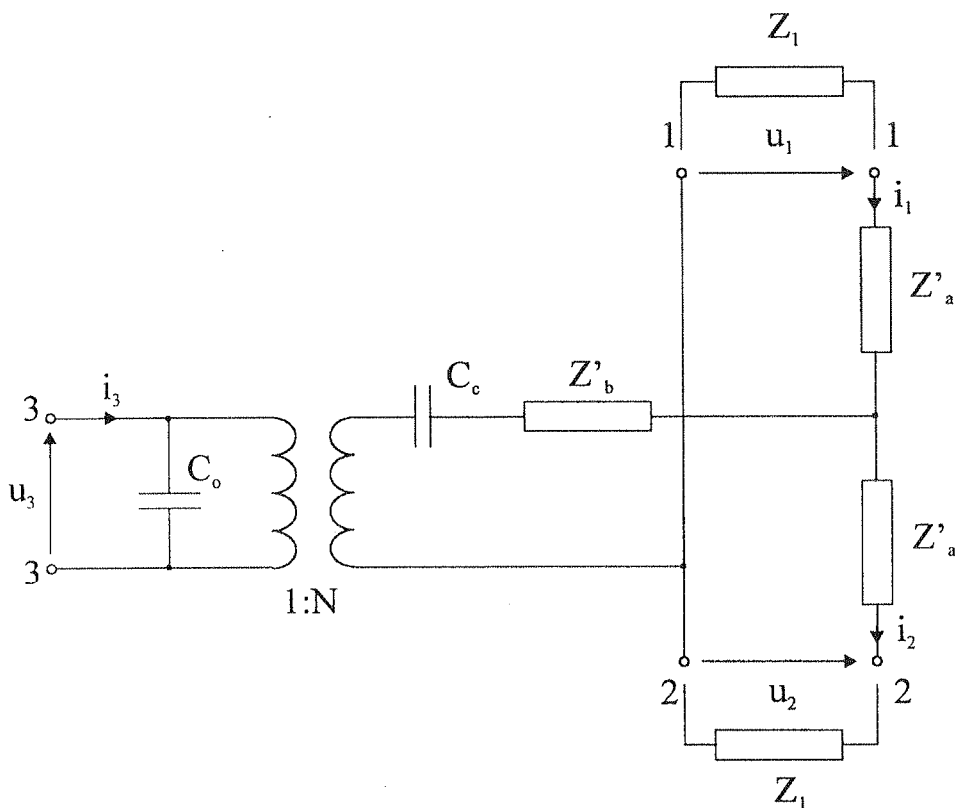
$$\begin{aligned}
 u_1 &= \frac{Z_0}{j \operatorname{tg} \alpha} i_1 - \frac{Z_0}{j \sin \alpha} i_2 + \frac{1}{j \omega C_0} i_3, \\
 u_2 &= \frac{Z_0}{j \sin \alpha} i_1 - \frac{Z_0}{j \operatorname{tg} \alpha} i_2 + \frac{1}{j \omega C_0} i_3, \\
 u_3 &= \frac{1}{j \omega C_0} i_1 - \frac{1}{j \omega C_0} i_2 + \frac{1}{j \omega C_0} i_3,
 \end{aligned} \quad (15)$$

gdzie:

$$Z_0 = \frac{\pi}{\omega_0 C_0 k_p^2}, \quad \omega_0 = 2\pi f_0. \quad (16)$$

W równaniach (15) zmienne i oraz u_1, u_2 są proporcjonalne odpowiednio:

$$i \sim \frac{\partial w}{\partial t}, \quad u \sim F.$$



Rys. 4. Schemat zastępczy obwodu dla przetwornika IDT z jedną parą elektrod gdzie:

$$C_c = -\frac{C_0}{N^2}, \quad Z'_b = -j \frac{Z'_0}{\sin \alpha}, \quad Z'_a = j Z'_0 \operatorname{tg} \frac{\alpha}{2}, \quad Z'_0 = Z_0 N^2, \quad \alpha = \frac{h}{V_p} \omega \quad (17)$$

Na po
o kształcie
występują
i jedno ran
ściowym u

przemieszc

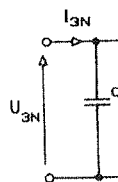
Natomiast
dzonemu r

Na rys
twornika
rozchodze
rys. 4 wys
pięcioweg
wprowadz
i mechan
o przekład

Do ran
cję falową
występują
w której r

gdzie: $G_1(G$

Zakład
ne oraz m
w postaci



(15)

Na podstawie równań (15) można zbudować zastępczy obwód elektryczny o kształcie litery T odpowiadający przetwornikowi piezoelektrycznemu, w którym występują dwa ramiona ultradźwiękowe o sygnałach wejściowych u_1 oraz i_1 , u_2 oraz i_2 i jedno ramię elektryczne o parametrach wejściowych u_3 oraz i_3 . Sygnałom wejściowym ultradźwiękowym u oraz i odpowiadają wytwarzane siły F oraz prędkości przemieszczeń $\frac{\partial w}{\partial t}$ cząstek fali występujące na końcach elektrod przetwornika.

(16)

Natomiast sygnały wejściowe elektryczne na trzecim wejściu odpowiadają doprowadzonemu napięciu u_3 oraz natężeniu prądu i_3 .

dnio:

Na rys. 4 przedstawiono schemat zastępczy obwodu elektrycznego dla przetwornika piezoelektrycznego z polem elektrycznym równoległym do kierunku rozchodzenia się fali Lamba [5, 6]. W schemacie zastępczym pokazanym na rys. 4 występują: pojemność C_0 dołączona równolegle do zacisków źródła napięciowego (odpowiada pojemności jednej pary elektrod) oraz pojemność C_c wprowadzona dla uzyskania równoważności obwodu od strony elektrycznej i mechanicznej [12]. Transformator na rys. 4 jest transformatorem idealnym o przekładni $1:N$.

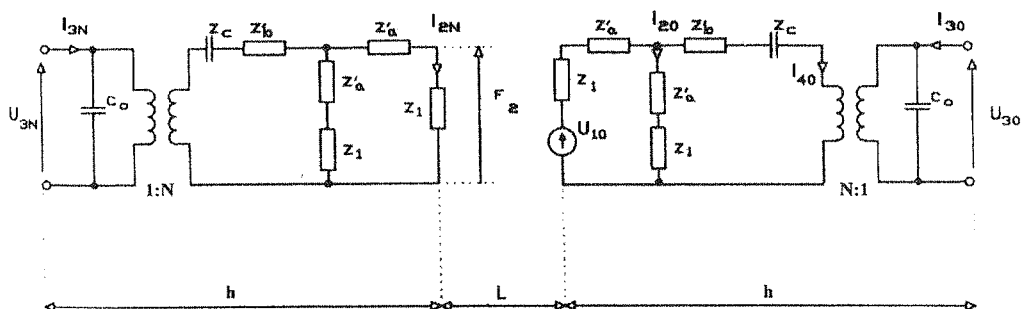
Do ramion ultradźwiękowych obwodu (zaciski 1—1 i 2—2) podłączono impedancję falową Z_1 , która charakteryzuje własności ultradźwiękowe ośrodka. Siłę F_2 występującą na granicy pomiędzy krawędzią elektrody przetwornika a ośrodkiem, w której rozchodzi się fala Lamba wyznaczono z zależności:

$$F_2 = U_2 = I_2 Z_1 = N G_1(j\alpha) U_3 \quad (18)$$

gdzie: $G_1(j\alpha)$ — transmitancja zastępcza przetwornika.

3. MODEL MEMBRANOWEGO CZUJNIKA W POSTACI LINII OPÓŹNIAJĄCEJ

Zakładając, że przetworniki IDT nadawczy i odbiorczy są symetryczne i identyczne oraz mają po jednej parze elektrod można czujnik z falą Lamba przedstawić w postaci linii opóźniającej, której schemat zastępczy pokazano na rys. 5.



Rys. 5. Schemat zastępczy obwodu dla linii opóźniającej z falą Lamba

Założono, że tłumienie rozchodzącej się fali ma charakter wykładniczy (współczynnik tłumienia jest nie zależny od amplitudy fali przy ustalonych warunkach zewnętrznych) co wynika przeprowadzonych badań [2, 7]. Analizując odpowiednie gałęzi obwodu z rys. 4 otrzymano:

$$\begin{aligned} N \cdot U_{3N} &= 2I_{2N} \left(jZ_c - j \frac{Z'_0}{\sin \alpha} \right) + I_{2N} \left(jZ'_0 \operatorname{tg} \frac{\alpha}{2} + Z_1 \right) = \\ &= F_2 \left\{ 1 + j \left[2 \frac{Z_c}{Z_1} - \frac{Z'_0}{Z_1} \left(\frac{2}{\sin \alpha} - \operatorname{tg} \frac{\alpha}{2} \right) \right] \right\}. \end{aligned} \quad (19)$$

Ze wzorów (17) i (18) wyznaczono $G_1(j\omega)$ wykorzystując przekształcenia trygonometryczne:

$$G_1(j\omega) = \frac{1}{1 + j \left(2 \frac{Z_c}{Z_1} - \frac{Z'_0}{Z_1} \operatorname{ctg} \frac{h}{2V_p} \omega \right)} \quad (20)$$

gdzie: $\alpha = \frac{h}{V_p} \omega$.

Siła F działająca na granicy przetwornika odbiorczego z ośrodkiem, w której rozchodzi się fala będzie zależna od siły F_2 wytwarzanej w przetworniku nadawczym oraz od odległości L pomiędzy środkami obu przetworników i współczynnika tłumienia ośrodka δ .

$$F = 2F_2 e^{-\delta L}. \quad (21)$$

Napięcie U_{10} odpowiada sile F działającej na wejściu akustycznym przetwornika odbiorczego (rys. 5) i można je wyznaczyć:

$$U_{10} = \left(Z_1 + jZ'_0 \operatorname{tg} \frac{\alpha}{2} \right) (I_{20} + I_{40}) + jI_{40} \left(-j \frac{Z'_0}{\sin \alpha} + Z_c \right) + N \cdot U_{30} \quad (22)$$

gdzie: U_{30} jest napięciem na wyjściu przetwornika odbiorczego.

Posługując się analizą obwodu (rys. 5) kolejno określono prądy I_{20} oraz I_{40} :

$$I_{20} = \frac{-jI_{40}}{\left(\frac{Z_1}{Z'_0} + j \operatorname{tg} \frac{\alpha}{2} \right) \sin \alpha}, \quad (23)$$

$$I_{40} = \frac{U_{10}}{Z_1 - jZ'_0 \operatorname{ctg} \frac{\alpha}{2}}. \quad (24)$$

Z równania (17) oraz równań (20), (21), (22), (23) wyznaczono napięcie na wyjściu przetwornika odbiorczego:

Wynikiem
określenie

Równanie
obwodu a

W przy
analizowa
elektrod w
tworniku n

Dla ukł
liczbie par

gdzie: L_{nm}

$n -$

$m -$

4. CHA

Wyzna
z asymetry
charakerys
parametra
 $d = 10 \mu\text{m}$,
kość elektr
— grubość

iczy (współ-
warunkach
odpowiednie

$$U_{30}(j\alpha) = \frac{2Z_c e^{-\delta L} U_{3N}}{Z'_0 \left[2\operatorname{ctg} \frac{\alpha}{2} - 2\frac{Z_c}{Z'_0} - j \left(\frac{Z_1}{Z'_0} + 2\frac{Z_c}{Z_1} \operatorname{ctg} \frac{\alpha}{2} - \frac{Z'_0}{Z_1} \operatorname{ctg}^2 \frac{\alpha}{2} \right) \right]} \quad (25)$$

Wynikiem analizy obwodu zastępczego przetworników IDT oraz linii opóźniającej jest określenie modułu stosunku napięcia wyjściowego U_{30} do napięcia wejściowego U_{3N} :

$$(19) \quad \left| \frac{U_{30}}{U_{3N}} \right| = \frac{2Z_c e^{-\delta L}}{Z'_0} \frac{1}{\sqrt{\left(2\operatorname{ctg} \frac{\alpha}{2} - 2\frac{Z_c}{Z'_0} \right)^2 + \left(\frac{Z_1}{Z'_0} + 2\frac{Z_c}{Z_1} \operatorname{ctg} \frac{\alpha}{2} - \frac{Z'_0}{Z_1} \operatorname{ctg}^2 \frac{\alpha}{2} \right)^2}}. \quad (26)$$

nia trygono-

Równanie (26) pokazuje zależności zmian sygnału wyjściowego od parametrów obwodu a także od jego częstotliwości $\left(\alpha = \frac{h}{V_p} \omega \right)$.

(20)

W przypadku gdy przetwornik posiada więcej niż jedną parę elektrod należy analizować w przetworniku odbiorczym kolejne pobudzenia w danym segmencie elektrod wywołane przez siły pochodzące kolejno od segmentów od 1 do n w przetworniku nadawczym.

Dla układu dwóch symetrycznych przetworników nadawczego i odbiorczego o N_p liczbie par elektrod można równanie (26) przedstawić w postaci:

n, w której
nadawczym
półczynnika

$$\begin{aligned} \left| \frac{U_{30}}{U_{3N}} \right| = & 2 \frac{Z_c}{Z'_0} \left[\left(\sum_{n=1}^{N_p} \sum_{m=1}^{N_p} e^{-\delta L_{nm}} \sin \frac{\alpha L_{nm}}{h} \right)^2 + \left(\sum_{n=1}^{N_p} \sum_{m=1}^{N_p} e^{-\delta L_{nm}} \cos \frac{\alpha L_{nm}}{h} \right)^2 \right]^{\frac{1}{2}} \\ & \times \left[\left(2\operatorname{ctg} \frac{\alpha}{2} - 2\frac{Z_c}{Z'_0} \right)^2 + \left(\frac{Z_1}{Z'_0} + 2\frac{Z_c}{Z_1} \operatorname{ctg} \frac{\alpha}{2} - \frac{Z'_0}{Z_1} \operatorname{ctg}^2 \frac{\alpha}{2} \right)^2 \right]^{-\frac{1}{2}} \end{aligned} \quad (27)$$

zetwornika

gdzie: L_{nm} — odległość pomiędzy środkiem n -tej pary elektrod przetwornika nadawczego i środkiem m -tej pary elektrod przetwornika odbiorczego,

(22)

n — odpowiada kolejnemu segmentowi par elektrod przetwornika nadawczego,

m — odpowiada kolejnemu segmentowi par elektrod przetwornika odbiorczego.

I_{40} :

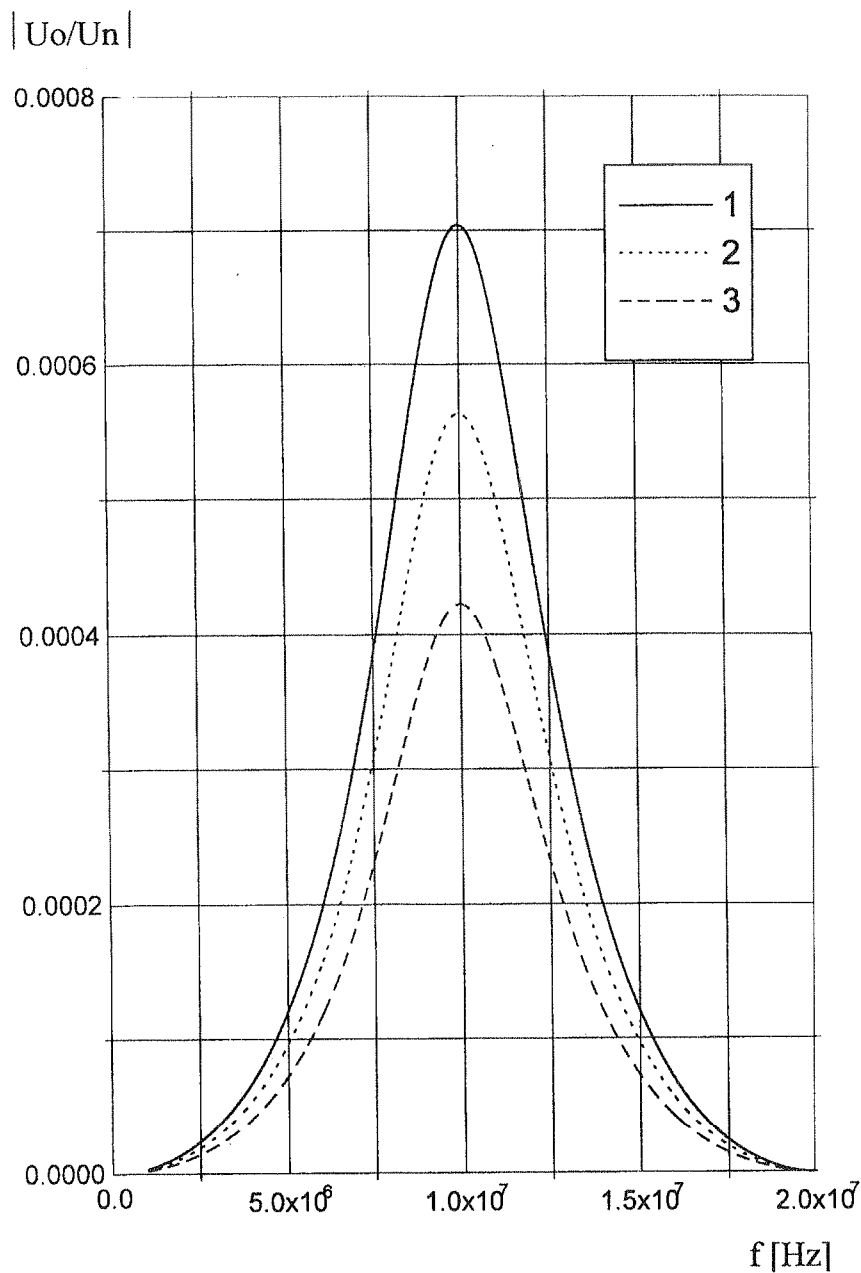
(23)

4. CHARAKTERYSTYKI CZĘSTOTLIWOŚCIOWE MODELU CZUJNIKA

(24)

Wyznaczono charakterystyki częstotliwościowe rozpatrywanego modelu czujnika z asymetrycznymi przetwornikami IDT o dwóch elektrodach. Na rys. 6 pokazano charakterystykę częstotliwościową czujnika z dwoma przetwornikami o następujących parametrach: $l = 2040 \mu\text{m}$, $h = 10 \mu\text{m}$, $L = 1000 \mu\text{m}$, $N_p = 1$, $k_p^2 = 0,005$, $C_0 = 2 \text{ pF}$, $d = 10 \mu\text{m}$, $d_1 = 1 \mu\text{m}$, $a = 2,5 \text{ mm}$, $b = 2 \text{ mm}$ (gdzie: l — długość elektrod, h — szerokość elektrod, L — odległość pomiędzy przetwornikami, d — grubość membrany, d_1 — grubość elektrod, a — długość membrany, b — szerokość membrany) [11]. Na

na wyjściu



Rys. 6. Charakterystyki częstotliwościowe modelu czujnika dla przetworników z jedną parą elektrod ($N_p=1$), gdzie: 1 — dla współczynnika tłumienia $\exp(-\delta L)=0,1$, 2 — $\exp(-\delta L)=0,08$, 3 — $\exp(-\delta L)=0,06$, $|U_o/U_n|$ — moduł ilorazu napięć dla przetworników odbiorczego i nadawczego

|Uo/Un|

0.01

0.01

0.00

0.00

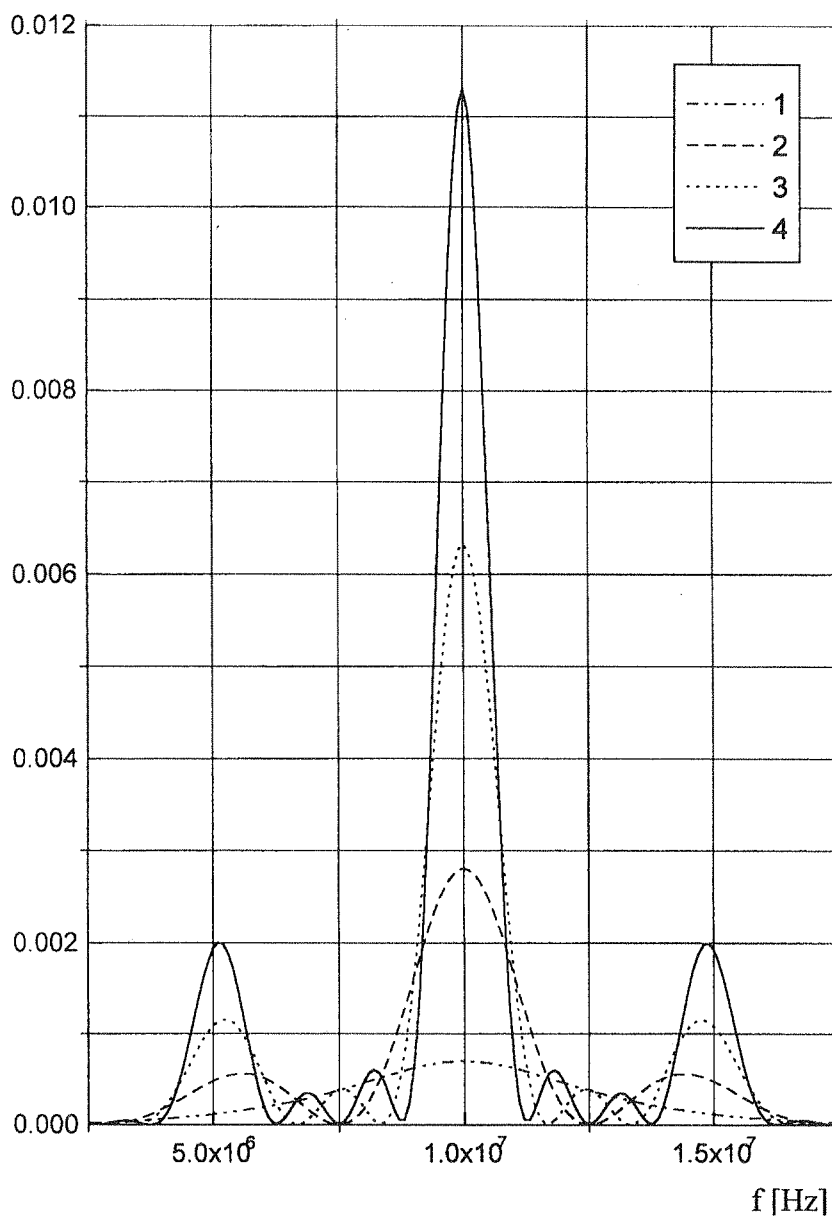
0.00

0.00

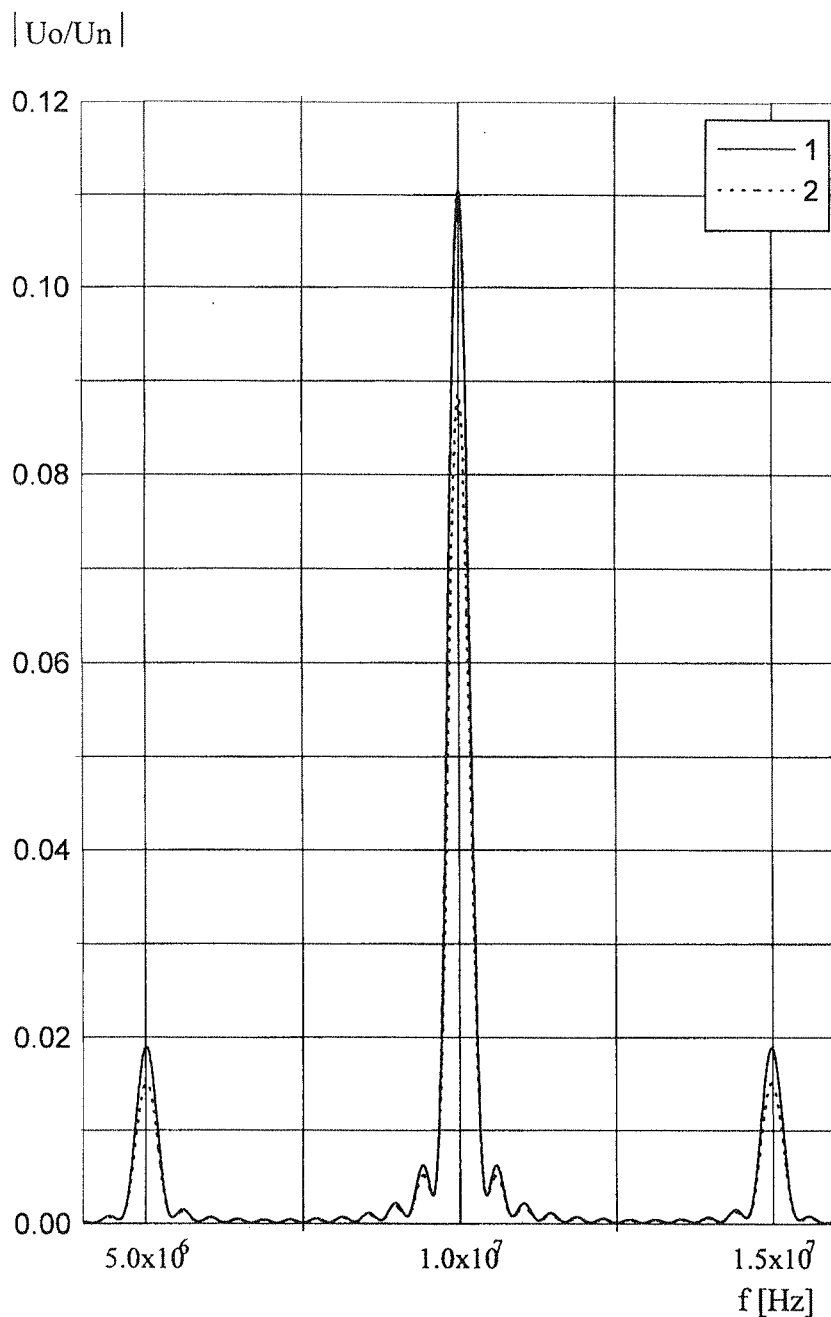
0.00

Rys. 7. Char
o różnej li

$$|U_o/U_n|$$



Rys. 7. Charakterystyki częstotliwościowe modelu czujnika z dwoma symetrycznymi przetwornikami o różnej liczbie elektrod, gdzie: 1 — $N_p=1$, 2 — $N_p=2$, 3 — $N_p=3$, 4 — $N_p=4$; $\exp(-\delta L)=0,1$



Rys. 8. Charakterystyka częstotliwościowa modelu czujnika dla przetworników o dwunastu parach elektrod $N_p = 12$, gdzie: 1 — $\delta = 2220$, 2 — $\delta = 2450$

0.12
0.1
0.08
0.06
0.04
0.02
0
15

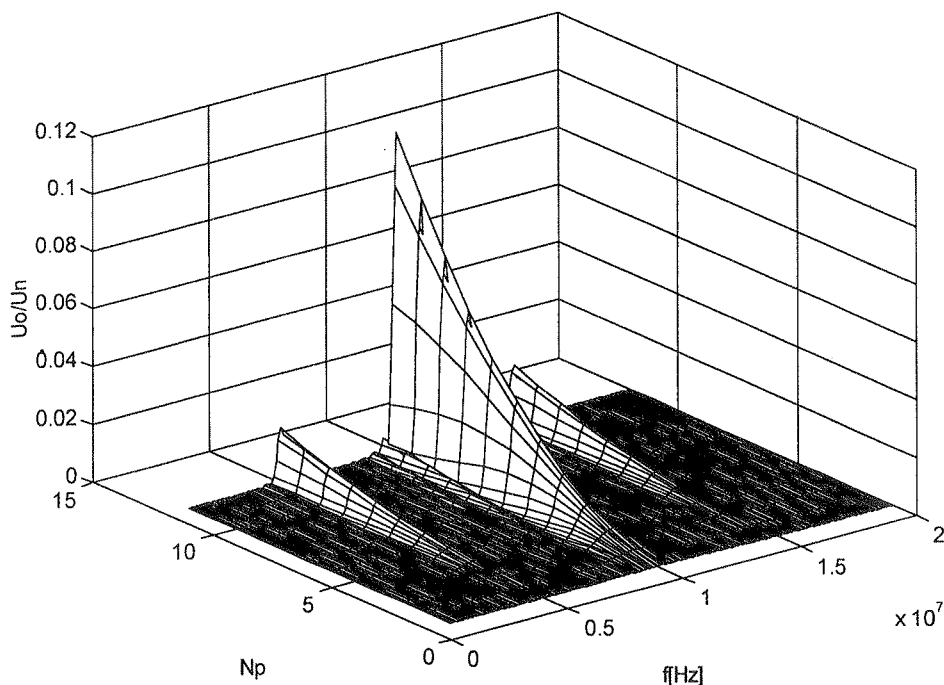
U_o/U_n

Rys. 9. C

materiał p
ników alu
z przetwo
ników tłu
Param

Charakter
par elektro
modelu dla
tłumienia.

Na rys
o liczbie p
mają istot
można wy
kwadratu
ści piezoel
parametr
impedanc
membran
współczyn



Rys. 9. Charakterystyki częstotliwościowe modelu dla liczby par elektrod N_p od 1 do 12, $\delta = 2220$

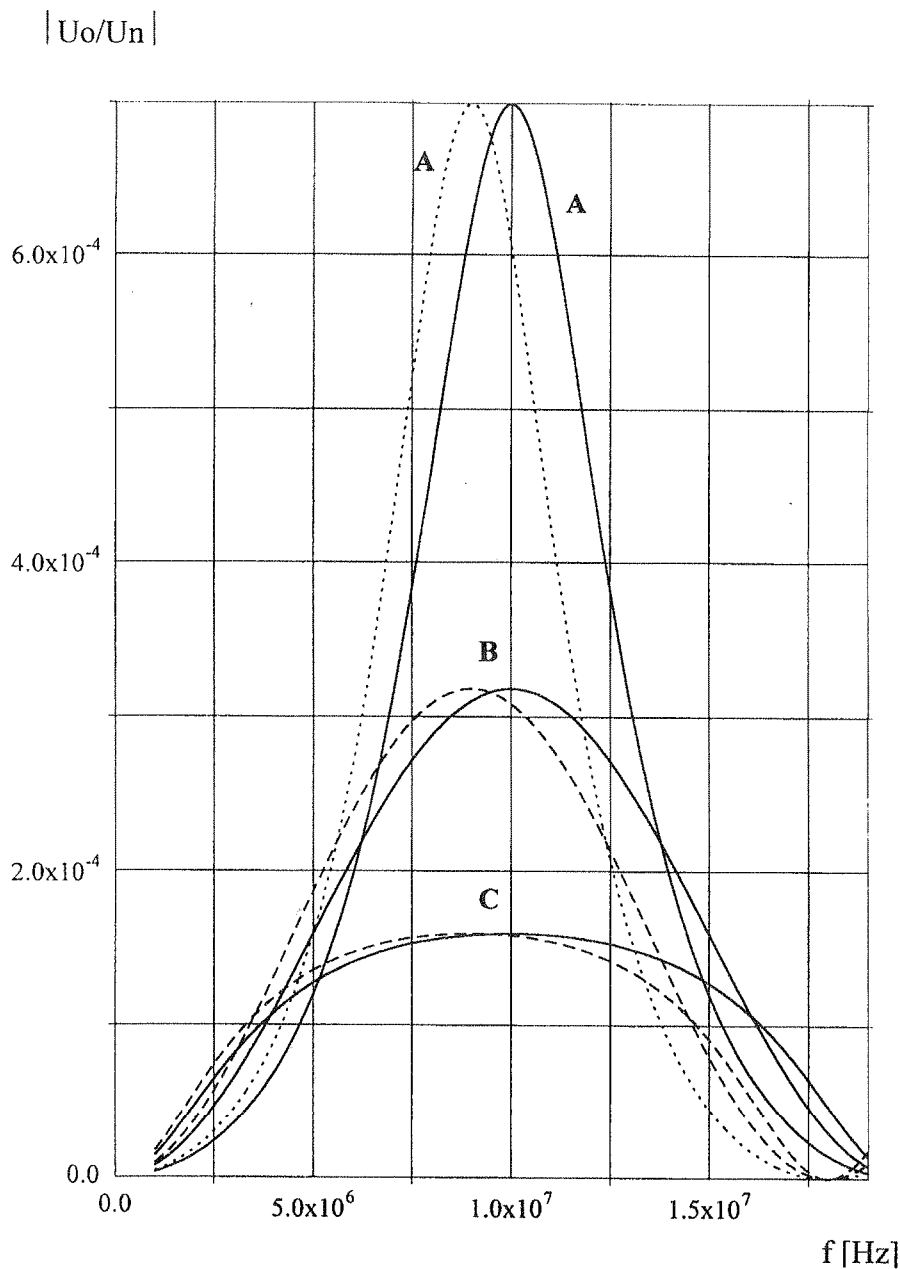
materiał piezoelektryczny membrany wybrano tlenek cynku, a na elektrody przetworników aluminium [8, 9]. Na rys. 6 przedstawiono charakterystyki modelu czujnika z przetwornikami o dwóch elektrodach dla trzech przypadków różnych współczynników tłumienia.

Parametry występujące we wzorze (26) dla w/w danych wynoszą odpowiednio:

$$Z_C/Z'_0 = 0,0016, \quad Z_C/Z_1 = 0,00352, \quad Z'_0/Z_1 = 2,2, \quad \lambda_0 = 40 \mu\text{m}.$$

Charakterystyki częstotliwościowe modelu z przetwornikami dla czterech różnych par elektrod pokazano na rys. 7. Natomiast na rys. 8 przedstawiono charakterystykę modelu dla dwunastu par elektrod przetworników i dwóch różnych współczynników tłumienia.

Na rys. 9 przedstawiono w układzie trzech współrzędnych charakterystyki modelu o liczbie par elektrod N_p od 1 do 12. Na kształt charakterystyki częstotliwościowej mają istotny wpływ wartości parametrów (27) Z_C/Z'_0 , Z'_0/Z_1 , trzeci iloraz Z_C/Z_1 można wyznaczyć z dwóch poprzednich. Wartość ilorazu Z_C^A/Z'_0 jest zależna od kwadratu współczynnika sprzężenia elektromechanicznego k_p , a zatem od właściwości piezoelektrycznych materiału membrany. Na wartość stosunku Z'_0/Z_1 przy stałych parametrach przetworników piezoelektrycznych ma decydujący wpływ wartość impedancji ośrodka Z_1 . W czujnikach pracujących jako detektory masy powierzchnia membrany zostaje obciążona dodatkową masą [10, 15]. Powoduje to zmianę wartości współczynnika tłumienia δ oraz zmianę impedancji Z_1 .



Rys. 10. Charakterystyki częstotliwościowe dla różnych impedancji ośrodka oraz dla dwóch prędkości fazowych; linia przerywana — zmniejszenie prędkości o 10%, $Z_c/Z'_0 = 0,0016$, $\delta = 2220$, $N_p = 1$; A — dla impedancji $Z_1 = \frac{1}{2,2} Z'_0$, B — dla impedancji $Z_1 = Z'_0$, C — dla $Z_1 = 2Z'_0$

Na rys. elektrodom sprzężenia Lamba ule W przypad nie dużych prędkości zależność α (dla małych

W czu zmianę prę zmiany pr oznaczone

W mo impedancj (zasilające oraz odwr sygnału oc Na rys. 10

A tzn. gd

zmniejszer nych przy Z_1) straty zeru. W ro czastych p zbliżoną d

Przyjęt współczyn oraz prę wykorzyst oraz przet nego [11]. w rozdzia

W tab (U_o) do n Dla ba szenia wy $f_0 = 10$ M w modelu

Na rys. 10 pokazano charakterystyki częstotliwościowe przetworników z dwoma elektrodami dla różnych impedancji falowych ośrodka. Założono, że współczynnik sprzężenia elektromechanicznego k_p oraz parametry przetworników są stałe. Fale Lamba ulegają dyspersji, ich prędkość fazowa jest zależna od częstotliwości [1, 10]. W przypadku izotropowej i jednorodnej membrany zmiany prędkości dla odpowiednio dużych wartości $d \geq \lambda$ są niewielkie. W zależności od rodzaju generowanej fali [13] prędkości fazowe mogą być różne, dla fal symetrycznych występuje mniejsza zależność od częstotliwości a dla fal asymetrycznych wartości prędkości są mniejsze (dla małych wartości iloczynu $d \times f$).

W czujnikach zmiana masy obciążającej powierzchnię membrany spowoduje zmianę prędkości rozchodzenia się fali. Na rys. 10 linią przerywaną pokazano wpływ zmiany prędkości fazowej (zmniejszenie o 10% prędkości w stosunku do wykresu oznaczonego linią ciągłą) na charakterystykę częstotliwościową modelu czujnika.

W modelu nie rozważa się efektu odbicia fali w przypadku niedopasowania impedancji, co ma wpływ na sprawność przetwarzania mocy elektrycznej generatora (zasilającego nadajnik piezoelektryczny) w moc mechaniczną fali ultradźwiękowej oraz odwrotnie przy przetwarzaniu mocy fali ultradźwiękowej w moc elektryczną sygnału odbiornika. Minimalne straty energii przetwarzania wystąpią gdy $Z_1 = Z'_0$. Na rys. 10 otrzymano minimalne tłumienie sygnału wyjściowego U_0 dla przypadku

A tzn. gdy $Z_1 = \frac{1}{2,2} Z'_0$, niedopasowanie impedancji ośrodka spowoduje jednak

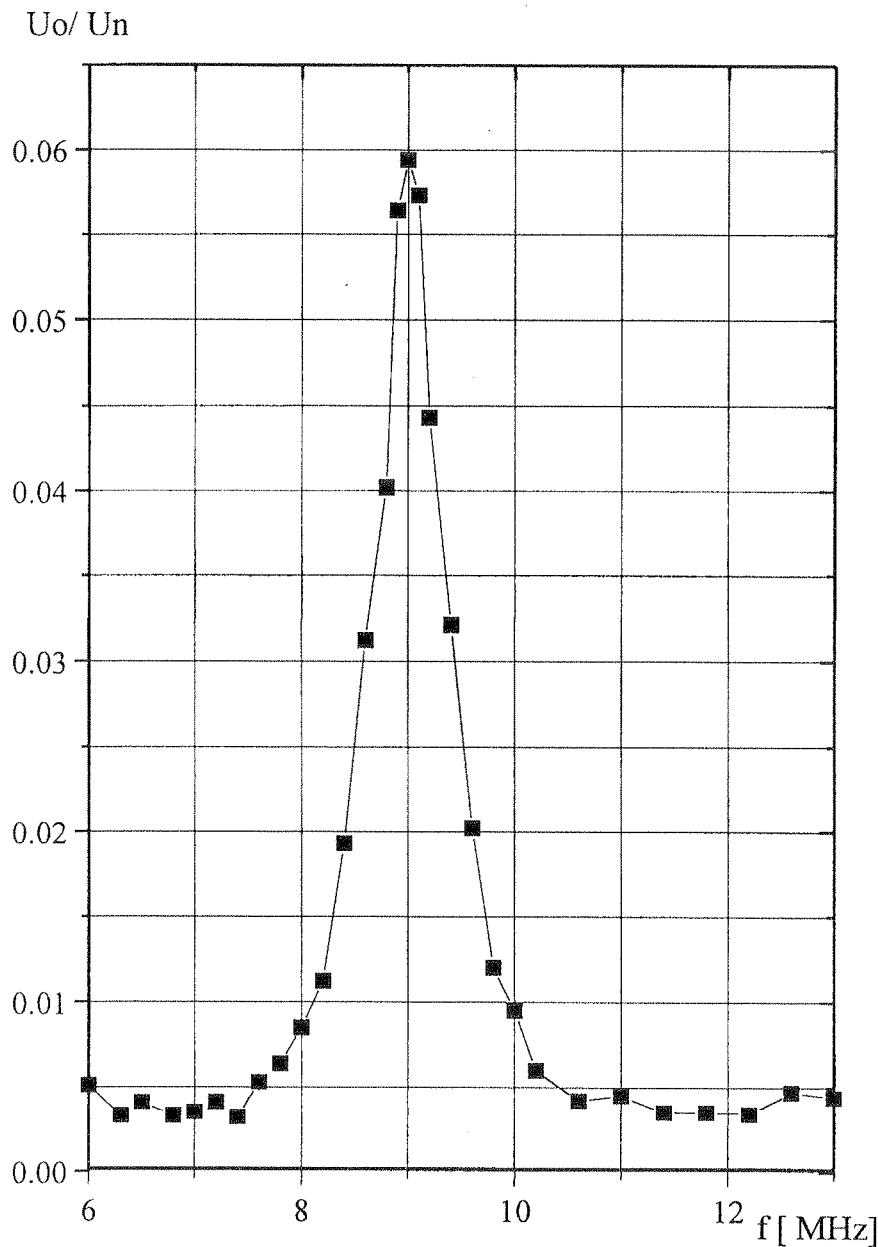
zmniejszenie przetwarzanej energii pomiędzy nadajnikiem a odbiornikiem. W skrajnych przypadkach niedopasowania (dla bardzo małych i bardzo dużych impedancji Z_1) straty energii mogą być bardzo duże i wówczas sygnał wyjściowy będzie bliski zeru. W rozważanym czujniku membranowym ze względu na konstrukcję międzypalczastych przetworników piezoelektrycznych impedancja falowa ośrodka ma wartość zbliżoną do wartości impedancji przetworników.

5. WYNIKI POMIARÓW

Przyjęte do obliczeń wartości współczynnika sprzężenia elektromechanicznego k_p , współczynnika przenikalności dielektrycznej ϵ , współczynnika sprężystości c , gęstości oraz prędkości fazowej dla materiału piezoelektrycznego tlenku cynku (ZnO) wykorzystano z publikowanych danych [8, 9, 18]. Wymiary geometryczne membrany oraz przetworników piezoelektrycznych były mierzone w czasie procesu technologicznego [11]. Badano membranę piezoelektryczną z ZnO o parametrach podanych w rozdziale 4 z przetwornikami międzypalczastymi o 12 parach elektrod.

W tabeli 1 podano uśrednione wyniki pomiarów ilorazu napięcia wyjściowego (U_0) do napięcia wejściowego (U_n).

Dla badanego fizycznego modelu czujnika częstotliwość środkowa pasma przeniesienia wynosiła $f_0 = 9,05$ MHz, natomiast w rozpatrywanym modelu matematycznym $f_0 = 10$ MHz. Również różnice występują w stopniu tłumienia sygnału wyjściowego, w modelu matematycznym tłumienie jest mniejsze a selektywność obwodu większa.



Rys. 11. Charakterystyki częstotliwościowa badanego czujnika

Różnice wartości częstotliwości f_0 wynikają przede wszystkim z przyjętej wartości prędkości fazowej fali (rys. 10). Natomiast zwiększenie tłumienia sygnału w modelu fizycznym spowodowane jest większą wartością współczynnika tłumienia δ oraz mniejszą wartością współczynnika sprzężenia elektromechanicznego k_p . Dokładne

Lp.
Częstotliwość [MHz]
U_o/U_n [$\times 10^{-3}$]

Lp.
Częstotliwość [MHz]
U_o/U_n [$\times 10^{-3}$]

Lp.
Częstotliwość [MHz]
U_o/U_n [$\times 10^{-3}$]

wyznaczenie
określenie
na param
z zaznaczo

Rozwa
wymi typu
nych do ge
Dla wieloe
jące modu
napięcia w
kilku różn
obciążenia
maksymal
nia elektro
nika k_p sp
nika δ zw
jest zależn
fazową fal
przetworn

Tabela 1

Wyniki pomiarów

Lp.	1	2	3	4	5	6	7	8	9	10
Częstotliwość [MHz]	6,0	6,3	6,5	6,8	7,0	7,2	7,4	7,6	7,8	8,0
U_o/U_n [$\times 10^{-3}$]	5,1	3,3	4,1	3,3	3,5	4,1	3,2	5,3	6,4	8,5

Lp.	11	12	13	14	15	16	17	18	19	20
Częstotliwość [MHz]	8,2	8,4	8,6	8,8	8,9	9,0	9,1	9,2	9,4	9,6
U_o/U_n [$\times 10^{-3}$]	11,2	19,3	31,2	40,2	56,4	59,5	57,3	44,3	32,10	20,2

Lp.	21	22	23	24	25	26	27	28	29	30
Częstotliwość [MHz]	9,8	10,0	10,2	10,6	11,0	11,4	11,8	12,2	12,6	13,0
U_o/U_n [$\times 10^{-3}$]	12,0	9,5	6,0	4,2	4,5	3,5	3,5	3,4	4,7	4,4

wyznaczenie współczynnika sprzężenia elektromechanicznego a także dokładne określenie prędkości fazowej dla cienkich warstw jest trudne i ma największy wpływ na parametry modelu. Na rys. 11 pokazano charakterystykę badanego czujnika z zaznaczonymi punktami pomiarowymi.

6. WNIOSKI

Hz]

j wartości
w modelu
ia δ oraz
Dokładne

Rozważany model matematyczny membranowego czujnika z falami ultradźwiękowymi typu Lamba z zastosowaniem międzypalczastych przetworników piezoelektrycznych do generacji i detekcji fal otrzymano przy zastosowaniu równań Masona [7, 8, 12]. Dla wieloelektrodowych symetrycznych przetworników otrzymano wyrażenie określające moduł transmitancji modelu (moduł stosunku napięcia wyjściowego odbiornika do napięcia wejściowego nadajnika). Charakterystyki częstotliwościowe wyznaczono dla kilku różnych parametrów modelu (rys. 6, 7, 8, 9, 10). Wpływ impedancji falowej obciążenia oraz prędkości fazowej na charakterystykę przedstawiono na rys. 10. Na maksymalną wartość modułu transmitancji ma wpływ wartość współczynnika sprzężenia elektromechanicznego k_p oraz współczynnika tłumienia δ . Zwiększenie współczynnika k_p spowoduje wzrost sprawności przetwornika, natomiast zwiększenie współczynnika δ zwiększa straty rozchodzącej się fali. Częstotliwość środkowa charakterystyki jest zależna przede wszystkim od częstotliwości f_o , która określona jest prędkością fazową fali i okresem przetwornika. Zwiększenie liczby par elektrod N_p symetrycznych przetworników nadawczego i odbiorczego powoduje zmniejszenie szerokości przeno-

szego pasma częstotliwościowego (poprawia selektywność obwodu) oraz powoduje wzrost sygnału wyjściowego. Przy dopasowaniu impedancji falowej ośrodka do impedancji przetwornika sprawność przetwarzania jest największa. W rozpatrywanym modelu pominięto wpływ odbicia fali ultradźwiękowej od brzegów membrany a także wynikający z nie dopasowania impedancji przetworników i ośrodka. Wprowadzone założenia dotyczące jednorodności i izotropowości materiału membrany oraz założenie, że kierunek pola elektrycznego jest równoległy do kierunku rozchodzenia się fali ograniczają dokładność rozważanego modelu. W analizie modelu czujnika należy uwzględnić zmienność takich parametrów jak prędkości fazowej i współczynnika sprzężenia elektromechanicznego od częstotliwości oraz wymiarów geometrycznych przetworników piezoelektrycznych [7, 9]. W konstrukcjach miniaturowych czujników membrana zbudowana jest często z kilku warstw różnych materiałów również anizotropowych. Dokładne rozwiązanie analityczne zagadnień dotyczących rozchodzenia się fal ultradźwiękowych w takich membranach jest bardzo trudne [3, 15] i często ograniczone dokładnością pomiaru wszystkich parametrów takich jak współczynniki sprężystości $[c]$, piezoelektryczne $[e]$, dielektryczne $[\epsilon]$ występujące we wzorze (1).

Zastosowanie metod przybliżonych takich jak np. metoda elementów skończonych do analizy rozkładu naprężeń, przemieszczeń jak również rozkładu pól elektrycznych w membranie pozwala często osiągnąć zadowalające wyniki [15, 16, 17]. Metody te wymagają jednak starannego doboru warunków brzegowych i wielkości elementów, a wyniki powinny być weryfikowane za pomocą metod analitycznych oraz badań doświadczalnych.

Najkorzystniejszym rozwiązaniem wydaje się zbudowanie możliwie dokładnego modelu czujnika z zastosowaniem w jego analizie również przybliżonych metod numerycznych.

Praca została wykonana w ramach projektu KBN 8T10C02216.

BIBLIOGRAFIA

1. J. Gołębiowski: *Mikroczujniki z falą ultradźwiękową*. Elektronizacja 1997, nr 9, s. 17–21
2. K. Toda: *Lamb-wave delay lines wit interdigital electrodes*. J. Appl. Phys., 1973, vol. 44, no. 1, pp. 56–62
3. P. Heyliger: *Exact solutions for simply supported laminated piezoelectric plates*. J. Appl. Mech., 1997, vol. 64, pp. 299–306
4. Y. Jin, S.G. Joshi: *Coupling of interdigital transducer to ultrasonic Lamb waves*. Appl. Phys. Lett., 1991, vol. 58, no. 17, pp. 1830–1832
5. T. Nakamoto, T. Moriizumi: *A theory of a Quartz crystal microbalance based upon a Mason equivalent circuit*. Jap. J. Appl. Phys., 1990, vol. 29, no. 5, pp. 963–969
6. F. Craciun, L. Sorba, E. Molinari, M. Pappalardo: *A coupled-mode theory for periodic piezoelectric composites*. IEEE Trans. on Ultrason., Ferroelectr., and Freq. Control, 1989, vol. 36, no. 1, pp. 50–56
7. K. Uozumi, K. Ohson, R.M. White: *Generation and detection of ultrasonic Lamb waves in a thin deposited film by using interdigital transducers*. Appl. Phys. Lett., 1983, vol. 43, no. 10, pp.
8. L.P. Solie: *Piezoelectric waves on layered substrates*. J. Appl. Phys., 1973, vol. 44, no. 2, pp. 619–627

9. E.L. A
IEEE T
10. J. G o
torów n
11. J. G o
Membr
1998, ss
12. J. G o
dźwięko
wych M
13. J. G o
ultradzw
14. J. G o
ultrason
Zakopan
15. P. H a
bending.
16. J. G o ł
Mertolo
17. S. K a r
263–26
18. J. G o ł
sorów. M
ss. 222–
19. L.F. B r
Trans. o

THE MODE
THE GENE

The cons
type waves (Q
detection of t
transducers to
the model incl
the transducer
applied in the

Key words: u
membrane ser

9. E.L. Adler: *Electromechanical coupling to Lamb and shear-horizontal modes in piezoelectric plates*. IEEE Trans. on Ultrason., Ferroelectr., and Freq. Control, 1989, vol. 36, no. 2, pp. 223–230
10. J. Gołębiowski: *Analiza właściwości mikroczujników z falami ultradźwiękowymi jako detektorów masy*. Kwartalnik Elektroniki i Telekomunikacji, 1998, 44, z. 1, ss. 45–60
11. J. Gołębiowski, T. Budzyński, P. Grabiec, J. Jaźwiński, J. Koszur: *Membrany w mikroczujnikach z falą ultradźwiękową typu Lamba*. Mater. V Konfer., COE'98, Jurata, 1998, ss. 239–243
12. J. Gołębiowski: *Model przetwornika piezoelektrycznego w membranowych czujnikach z ultradźwiękową falą Lamba*. Mater. Konfer. VIII Sympoz. Modelowanie i Symulacja Systemów Pomiarowych MiSSP'98, Krynica, 1998, ss. 143–150
13. J. Gołębiowski, J. Kowalski: *Modelowanie generatora z membranowym czujnikiem ultradźwiękowym w pętli sprzężenia zwrotnego*. Mater. V Konfer. COE'98, Jurata, 1998, pp. 223–237
14. J. Gołębiowski: *Modelling of the structures in the composite membranes of the silicone ultrasonic sensors*. Mater. Konfer. — 9th International Symposium on System Modeling Control, Zakopane, 1998
15. P. Haylinger, S. Brooks: *Exact solutions for laminated piezoelectric plates in cylindrical bending*. ASME J. Appl. Mech., vol. 63, 1996, pp. 903–910
16. J. Gołębiowski: *Analiza dynamiczna membrany czujnika z falą ultradźwiękową*. Mater. Konfer. Mertologia Wspomagana Komputerowo MWK'97, Zegrze, 1997, ss. 43–48
17. S. Kang, S. Im: *Finite Element analysis of wrinkling membranes*. J. Appl. Mech., 1997, vol. 64, pp. 263–269
18. J. Gołębiowski: *Analiza naprężeń w wielowarstwowych membranach krzemowych mikrosensorów*. Mater. Konfer. Modelowanie i Symulacja Systemów Pomiarowych, MiSSP'97, Krynica, 1997, ss. 222–227
19. L.F. Brown, D.L. Carlson: *Ultrasound transducer modes for piezoelectric polymer films*. IEEE Trans. on Ultrason., Ferroelectr., and Freq. Control, 1989, vol. 36, no. 3, pp. 313–318

J. GOŁĘBIOWSKI

THE MODEL OF THE MEMBRANE SENSOR WITH THE PIEZOELECTRIC TRANSDUCERS TO THE GENERATION AND DETECTION OF THE ULTRASONIC PLATE WAVES LAMBA TYPE

Summary

The construction of the interdigital piezoelectric transducers (IDT) used in the sensors with the Lamba type waves (FPW) was discussed. The equivalent scheme of these transducers used to generation and detection of the ultrasonic wave was presented. The model of the membrane sensor with the piezoelectric transducers to generation and detection of Lamba type wave was discussed. The frequency characteristic of the model including the influence of the model parameters was analysed. The results of the calculations for the transducer model were compared with the experimental results obtained for the transducers that were applied in the sensor.

Key words: ultrasonic waves, ultrasonic wave sensors, interdigital piezoelectric transducer, model of membrane sensor.

W

rzecz
biolo
Przec
netyc
i głę
falow

Słowe

Rak su
20% wszy
30 lat wys
operacyjn
zwiększyły
kalectwem
nymi. Pom
a zabieg c
podstawo
wczesnego
na celu po

Działan
wzrost du
z jego prz
tkanek.

Zastoso
wych, jest
organolept
stadium ch

Wieloczęstotliwościowa termografia mikrofalowa

BRONISŁAW STEC, ANDRZEJ DOBROWOLSKI

Wydział Elektroniki Wojskowej Akademii Technicznej w Warszawie

Otrzymano 1999.02.05

Autoryzowano 1999.04.28

Artykuł prezentuje problematykę związaną z promieniowaniem termicznym ciał rzeczywistych w zakresie mikrofalowym oraz opis własności transmisyjnych tkanek biologicznych rozpatrywanych w aspekcie diagnozowania nowotworu złośliwego sutka. Przedstawiono opis matematyczny transmisji niejednorodnej płaskiej fali elektromagnetycznej przez model warstwowy tkanki oraz opisano istotę pomiarów temperatury i głębokości położenia ogniska za pomocą wieloczęstotliwościowego termografu mikrofalowego.

Słowa kluczowe: termograf mikrofalowy, radiometr, promieniowanie termiczne

1. WPROWADZENIE

Rak sutka jest najczęściej występującym nowotworem złośliwym i stanowi około 20% wszystkich chorób onkologicznych u kobiet w Polsce. Podejmowane od przeszło 30 lat wysiłki w celu poprawienia wyników leczenia drogą radykalizacji zabiegów operacyjnych oraz dołączenie radioterapii i hormonoterapii profilaktycznej nie zwiększyły w sposób istotny liczby wyleczeń, obciążając dodatkowo pacjentki kalectwem pooperacyjnym, uszkodzając promiennymi i zaburzeniami hormonalnymi. Ponieważ wyleczenie zależy w znacznym stopniu od wczesnego rozpoznania, a zabieg chirurgiczny wykonany w tym czasie daje największą szansę przeżycia, podstawowego znaczenia nabiera upowszechnienie nowych i skutecznych metod wczesnego wykrywania nowotworów. Wprowadzenie termografii mikrofalowej ma na celu poszerzenie dostępnego arsenału urządzeń i metod diagnostycznych.

Działanie termografów opiera się na fakcie, że obecność nowotworu powoduje wzrost dużej ilości naczyń krwionośnych w jego sąsiedztwie, co w połączeniu z jego przyspieszonym metabolizmem powoduje wzrost temperatury otaczających tkanek.

Zastosowanie termografu mikrofalowego pozwala na wykrycie komórek rakowych, jest średnicy rzędu 1 mm (dla porównania mammografia około 5 mm, badanie organoleptyczne powyżej 1 cm). Wykrycie tak małych niejednorodności (wczesne stadium choroby) pozwala na wyeliminowanie interwencji chirurgicznej i leczenie

przy wykorzystaniu chemioterapii. W odróżnieniu od termografii w zakresie podczerwieni, która umożliwia pomiar wyłącznie temperatury powierzchni, termografia mikrofalowa pozwala na wykrywanie niejednorodności termicznych w głębi próbek biologicznych.

2. PODSTAWY FIZYCZNE TERMOGRAFII MIKROFALOWEJ

Wszystkie ciała o temperaturze wyższej od zera bezwzględnego emitują promieniowanie elektromagnetyczne i pochłaniają energię elektromagnetyczną z otaczającej przestrzeni. Moc mikrofalowa w paśmie (f_1, f_2) wysyłana do elementarnego kąta bryłowego $d\omega$ półprzestrzeni z elementu powierzchni dS_1 ciała rzeczywistego o temperaturze T , charakteryzujące się emisyjnością $e(f)$, w kierunku wyznaczonym przez kąt φ , zgodnie z prawem Rayleigha — Jeansa, wynosi [5]

$$dP = e(f) \cdot \frac{2}{3} \cdot \frac{kT}{c^2} \cdot (f_2^3 - f_1^3) \cdot dS_1 \cdot \cos\varphi d\omega. \quad (1)$$

Promieniowanie elektromagnetyczne propaguje się w tkankach, które ze względu na brak własności magnetycznych, można zaliczyć do dielektryków stratnych. Rozwiązując wektorowe równania falowe Helmholtza dla pól elektrycznego i magnetycznego w dielektryku stratnym otrzymujemy rozwiązania podstawowe o postaci [6]:

$$\mathbf{E} = \mathbf{E}_0 e^{\pm(\alpha + j\beta)\mathbf{n} \cdot \mathbf{r} + j\omega t}, \quad \mathbf{H} = \mathbf{H}_0 e^{\pm(\alpha + j\beta)\mathbf{n} \cdot \mathbf{r} + j\omega t}. \quad (2)$$

Ponieważ pochłanianą moc można wyrazić jako funkcję kwadratu natężenia pola, absorpcję mocy mikrofalowej charakteryzuje się często przez tzw. głębokość wnikania:

$$\delta = \frac{1}{2\alpha} \quad (3)$$

tj. przez dystans, na którym moc niesiona przez falę zmniejsza się e -krotnie.

Korzystając z układu równań Maxwella otrzymujemy zależność określającą głębokość wnikania jako funkcję stałych materiałowych tkanki

$$\delta = \left\{ 2 \cdot \sqrt{\frac{\omega^2 \varepsilon_0 \varepsilon_r \mu_0}{2} \left[\sqrt{1 + \left(\frac{\sigma}{\omega \varepsilon_0 \varepsilon_r} \right)^2} - 1 \right]} \right\}^{-1}. \quad (4)$$

Błony komórkowe o pojemności około $1 \mu\text{F}/\text{cm}^2$ tworzą dla niskich częstotliwości warstwy izolacyjne tak, że prąd płynie jedynie przez obszar międzykomórkowy, co tłumaczy niską przewodność tkanki w tym zakresie częstotliwości. Ze wzrostem częstotliwości konduktywność tkanki rośnie [7], czego rezultatem jest wzrost pochłaniania i w efekcie malenie głębokości wnikania. Dla częstotliwości większych od 10 GHz głębokość wnikania jest tak mała, że mierzone termometrem mikrofalowym

promieni
kiej wars
zatem po
w obszar

Ponie
rozważyć
Maxwella
wany wsp
 α . W przy

przy czym
natomiast
 $\cos\beta = q \exp$

Korzy
czynnik t
funkcją z
padania α

Proces
poza jej c
przybliżeni
sutka, tłuszcz
warstwy i c
naczynia k
warstwy tk
pozwala na
w obszarze

Wszystk
obszaru o
emituje do
poszczegól
apertury an

promieniowanie umożliwia wykrycie niejednorodności termicznych zaledwie w cienkiej warstwie skóry. Wyniki uzyskiwane w zakresie fal milimetrowych powinny być zatem podobne do wyników otrzymywanych za pomocą termometrów pracujących w obszarze podczerwieni.

Ponieważ tkanki biologiczne złożone są z wielu zróżnicowanych warstw należy rozważyć załamanie fali na granicy warstw. Korzystając z równań różniczkowych Maxwella i warunków brzegowych oraz prawa Snelliusa można określić zmodyfikowany współczynnik załamania o wartości rzeczywistej, będący funkcją kąta padania α . W przypadku propagacji fali z ośrodka 1 do ośrodka 2 otrzymujemy [7]

$$n' = \sqrt{\sin^2 \alpha + \left(\frac{n_2 q}{n_1 \kappa_1} \right)^2 (\kappa_2 \cos \gamma - \sin \gamma)^2} \quad (5)$$

przy czym zespolony współczynnik załamania wyrażony jest w postaci $\underline{n} = n(1 - j\kappa)$, natomiast cosinus kąta odbicia jest wielkością zespoloną przedstawioną w postaci $\cos \beta = q \exp(j\gamma)$.

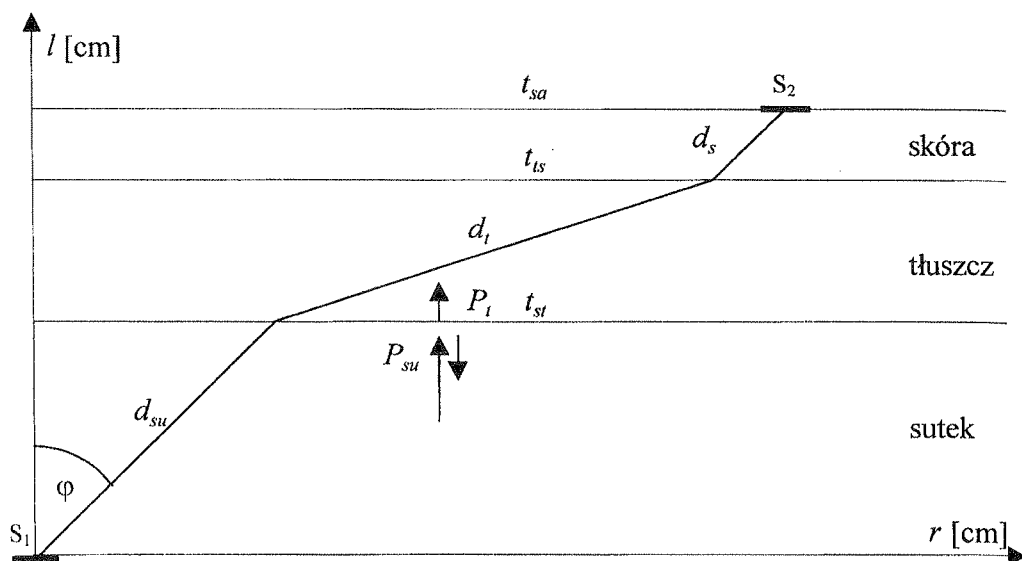
Korzystając ze wzorów Fresnela, otrzymujemy wyrażenie opisujące współczynnik transmisji dla fali propagującej się z ośrodka 1 do ośrodka 2 będący funkcją zespolonych przenikalności dielektrycznych ośrodków ϵ_1 i ϵ_2 oraz kąta padania α [5]

$$t_{12} = \frac{P_2}{P_1} = 1 - \frac{1}{2} \left(\left| \frac{\frac{\epsilon_2}{\epsilon_1} \cos \alpha - \sqrt{\frac{\epsilon_2}{\epsilon_1} - \sin^2 \alpha}}{\frac{\epsilon_2}{\epsilon_1} \cos \alpha + \sqrt{\frac{\epsilon_2}{\epsilon_1} - \sin^2 \alpha}} \right|^2 + \left| \frac{\cos \alpha - \sqrt{\frac{\epsilon_2}{\epsilon_1} - \sin^2 \alpha}}{\cos \alpha + \sqrt{\frac{\epsilon_2}{\epsilon_1} - \sin^2 \alpha}} \right|^2 \right). \quad (6)$$

3. MODEL WARSTWOWY TKANKI BIOLOGICZNEJ

Proces transmisji promieniowania od „gorących ośrodków” wewnątrz tkanki poza jej obszar graniczny jest bardzo skomplikowany i dlatego w pierwszym przybliżeniu przyjęty został prosty model warstwowy złożony z izotropowych warstw sutka, tłuszczu i skóry, o parametrach dielektrycznych zależnych jedynie od rodzaju warstwy i częstotliwości (rys. 1). Pomijane są inne obszary wewnętrzne, takie jak np. naczynia krwionośne. Ze względu na bardzo małe wartości promieniowanej mocy warstwy tkanek przyjęte do analizy traktowane są jako liniowe. Izotropowość warstw pozwala na przyjęcie założenia o prostoliniowym rozchodzeniu się promieniowania w obszarze każdej warstwy.

Wszystkie warstwy zakłada się jako izotermiczne o temperaturze T_0 , za wyjątkiem obszaru o powierzchni S_1 , którego temperatura jest o T wyższa. Powierzchnia ta emituje do półprzestrzeni dodatkowy strumień mocy P , który przechodząc przez poszczególne warstwy, ulega tłumieniu i dwukrotnemu załamaniu, zanim dotrze do apertury anteny odbiorczej o powierzchni S_2 .



Rys. 1. Model warstwowy tkanki biologicznej

W świetle dotychczasowych rozważań emisyjność przestrzenną rozpatrywanego modelu, wyrażoną jako funkcję kąta φ , mierzonego względem normalnej do powierzchni S_1 , możemy wyrazić zależnością [4]

$$e(\varphi) = \exp\left(-\frac{d_{su}}{\delta_{su}}\right) \cdot t_{st} \cdot \exp\left(-\frac{d_t}{\delta_t}\right) \cdot t_{ts} \cdot \exp\left(-\frac{d_s}{\delta_s}\right) \cdot t_{sa}, \quad (7)$$

gdzie: d_{su}, d_t, d_s — długość dróg w warstwach sutka, tłuszczu i skóry,
 $\delta_{su}, \delta_t, \delta_s$ — głębokość wnikania w warstwach sutka, tłuszczu i skóry,
 t_{st}, t_{ts} — współczynniki transmisji na granicach warstw,
 t_{sa} — współczynnik transmisji uwzględniający charakterystykę anteny.

Jeżeli założymy, że powierzchnia S_1 jest znacznie mniejsza od kwadratu promienia modelu oraz charakteryzuje się równomiernym rozkładem powierzchniowej gęstości natężenia promieniowania, to możemy ją potraktować jako punktowe źródło promieniowania i wówczas, w efekcie dalszych przekształceń, otrzymujemy wyrażenie, określające moc promieniowaną do pierścienia ograniczonego kątami φ_1 i φ_2 , wykrojonego w płaszczyźnie powierzchni S_2 , o postaci

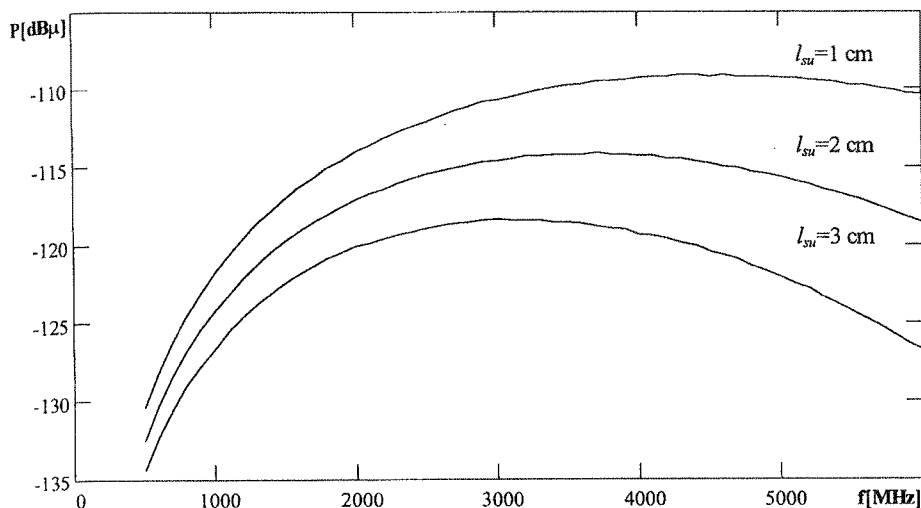
$$P = \frac{4\pi}{3} \cdot \frac{kT}{c^2} (f_2^3 - f_1^3) S_1 \int_{\varphi_1}^{\varphi_2} e(\varphi) \cos \varphi d\varphi. \quad (8)$$

Na podstawie zależności (8) wyznaczony został rozkład widmowy mocy promieniowanej przez kulistą powierzchnię S_1 o promieniu 2 mm i temperaturze wyższej o 5 K od temperatury otoczenia, odbieranej anteną o promieniu 7 mm, w dziesięcioprocentowym paśmie częstotliwości. Do obliczeń przyjęto szerokość warstwy tłuszczu $l_t = 2$ cm, szerokość warstwy skóry $l_s = 2.5$ mm i szerokość warstwy sutka l_{su} zmienianą parametrycznie (1 cm, 2 cm, 3 cm).

Wykr
 podstaw
 danej głę
 której od
 szerokość
 Efekt
 termiczne
 częstotliw
 ciowych r
 [1, 2, 3].

Wpływ
 redukcji m
 chni mod
 Poniew

to korzy
 ciepła T i



Rys. 2. Rozkład widmowy mocy odbieranej anteną umieszczoną centralnie ponad ogniskiem

Wykres obrazujący wyniki obliczeń przedstawiony jest na rys. 2. Na jego podstawie można stwierdzić, że dla danej szerokości warstwy sutka, a tym samym dla danej głębokości źródła promieniowania istnieje ściśle określona częstotliwość, na której odebrana będzie maksymalna moc. Analogiczny jakościowo efekt daje zmiana szerokości warstwy tłuszczu lub skóry.

Efekt ten wynika z faktu jednoczesnego wzrostu intensywności promieniowania termicznego (prawo Rayleigha — Jeansa) oraz spadku głębokości wnikania w funkcji częstotliwości (wzrost konduktywności). Badając kształt charakterystyk częstotliwościowych można więc określić rzeczywistą temperaturę źródła ciepła i jego głębokość [1, 2, 3].

4. METODYKA POMIARÓW WIELOCZĘSTOTLIWOŚCIOWYCH W PRZYPADKU TKANKI ZŁOŻONEJ Z WARSTW SUTKA, TŁUSZCZU I SKÓRY

Wpływ emisyjności przestrzennej $e(f)$, uwzględnionej w zależności (1), polega na redukcji mocy szumów cieplnych i w efekcie temperatura rejestrowana na powierzchni modelu T' jest odpowiednio mniejsza.

Ponieważ:

$$dP = \frac{2}{3} \cdot \frac{kT'}{c^2} \cdot (f_2^3 - f_1^3) \cdot dS_1 \cdot \cos \varphi \, d\omega \quad (9)$$

to korzystając z (1) otrzymujemy związek łączący rzeczywistą temperaturę źródła ciepła T i odpowiadającą jej temperaturę na powierzchni modelu T'

$$T' = T \cdot e(f) \quad (10)$$

W typowym przypadku praktycznym podczas określania głębokości ogniska raka sutka mamy do czynienia z trzema warstwami: sutka, tłuszczu i skóry oraz anteną umieszczoną bezpośrednio ponad ogniskiem. W tym przypadku zależność (10) można, korzystając z (7), rozwinąć do postaci:

$$T' = T \cdot \exp\left(-\frac{d_{su}}{\delta_{su}}\right) \cdot t_{st} \cdot \exp\left(-\frac{d_t}{\delta_t}\right) \cdot t_{ts} \cdot \exp\left(-\frac{d_s}{\delta_s}\right) \cdot t_{sa} \cdot K_T, \quad (11)$$

gdzie K_T oznacza wypadkowe wzmocnienie radiometru, rozumiane jako stosunek wskazywanej temperatury do odbieranej mocy.

Ze względu na znikomą grubość skóry (w przypadku sutka rzędu 1 ÷ 2 mm), czynnik tłumiący w skórze można dla danej częstotliwości uznać za wielkość stałą. Ponadto współczynniki transmisji na granicach warstw, charakterystyka anteny oraz transmitancja termometru również są dla danej częstotliwości (czyli egzemplarza radiometru) stałe. Wprowadzając wypadkowy współczynnik przeniesienia radiometru

$$K_r = t_{st} \cdot t_{ts} \cdot \exp\left(-\frac{d_s}{\delta_s}\right) \cdot t_{sa} \cdot K_T \quad (12)$$

otrzymujemy zależność opisującą związek temperatury na powierzchni tkanki z temperaturą ogniska i jego głębokością:

$$T' = T \cdot \exp\left(-\frac{d_{su}}{\delta_{su}} - \frac{d_t}{\delta_t}\right) \cdot K_r. \quad (13)$$

Współczynnik K_r należy wyznaczyć eksperymentalnie w procesie kalibracji, a głębokość wnikania można przyjąć z tablic lub również wyznaczyć eksperymentalnie, co jest dokładniejsze na etapie badań wstępnych realizowanych z wykorzystaniem konkretnych tkanek.

Kalibracji radiometru należy dokonać wykonując trzy pomiary przy trzech różnych szerokościach warstw:

$$\begin{cases} T_1 = T \cdot \exp\left(-\frac{d_{su1}}{\delta_{su}} - \frac{d_{t1}}{\delta_t}\right) \cdot K_r \\ T_2 = T \cdot \exp\left(-\frac{d_{su2}}{\delta_{su}} - \frac{d_{t1}}{\delta_t}\right) \cdot K_r \\ T_3 = T \cdot \exp\left(-\frac{d_{su1}}{\delta_{su}} - \frac{d_{t2}}{\delta_t}\right) \cdot K_r \end{cases} \quad (14)$$

Po rozwiązaniu układu równań otrzymujemy:

$$\delta_{su} = \frac{d_{su2} - d_{su1}}{\ln\left(\frac{T_1}{T_2}\right)}; \quad \delta_t = \frac{d_{t2} - d_{t1}}{\ln\left(\frac{T_1}{T_2}\right)} \quad (15)$$

Znając
częstotliw
i głębokoś
częstotliw

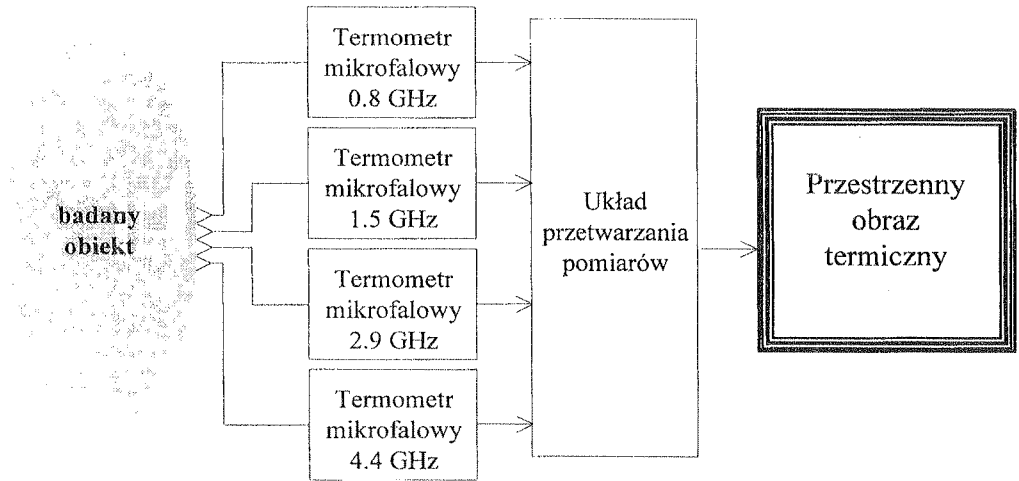
bad
obi

Dokon
metrów i l

Po rozwią
warstw th

$$K_r = \frac{T_1}{T} \cdot \exp\left(\frac{d_{su1}}{\delta_{su}} - \frac{d_{t1}}{\delta_t}\right) = \frac{T_2}{T} \cdot \exp\left(\frac{d_{su2}}{\delta_{su}} - \frac{d_{t1}}{\delta_t}\right) = \frac{T_3}{T} \cdot \exp\left(\frac{d_{su1}}{\delta_{su}} - \frac{d_{t2}}{\delta_t}\right) \quad (16)$$

Znając parametry δ_m , δ_t i K_r dla trzech radiometrów pracujących na różnych częstotliwościach można wyznaczyć zarówno rzeczywisty przyrost temperatury jak i głębokość ogniska realizując pomiar wieloczęstotliwościowy. Idea pomiaru wieloczęstotliwościowego pokazana jest na rys. 3.



Rys. 3. Idea mikrofalowego termografu 3D

Dokonując pomiaru wieloczęstotliwościowego z wykorzystaniem trzech termometrów i korzystając z (13) można napisać

$$\begin{cases} T_{f1} = T \cdot \exp\left(-\frac{d_{su}}{\delta_{suf1}} - \frac{d_t}{\delta_{tf1}}\right) \cdot K_{r1} \\ T_{f2} = T \cdot \exp\left(-\frac{d_{su}}{\delta_{suf2}} - \frac{d_t}{\delta_{tf2}}\right) \cdot K_{r2} \\ T_{f3} = T \cdot \exp\left(-\frac{d_{su}}{\delta_{suf3}} - \frac{d_t}{\delta_{tf3}}\right) \cdot K_{r3} \end{cases} \quad (17)$$

Po rozwiązaniu układu równań otrzymujemy wyrażenia określające szerokości warstw tłuszczu i sutka oraz rzeczywistą temperaturę ogniska:

$$d_t = \frac{\delta_{tf1}\delta_{tf2}\delta_{tf3}[a_2\delta_{suf3}(\delta_{suf1}-\delta_{suf2})-a_1\delta_{suf2}(\delta_{suf1}-\delta_{suf3})]}{\delta_{suf3}\delta_{tf2}(\delta_{suf1}-\delta_{suf2})(\delta_{tf1}-\delta_{tf3})-\delta_{mf2}\delta_{tf3}(\delta_{suf1}-\delta_{suf3})(\delta_{tf1}-\delta_{tf2})} \quad (18)$$

$$d_{su} = \frac{a_1 - d_t \left(\frac{1}{\delta_{tf2}} - \frac{1}{\delta_{tf1}} \right)}{\frac{1}{\delta_{suf2}} - \frac{1}{\delta_{suf1}}} \quad (19)$$

$$T = \frac{T_{f1}}{K_{r1}} \cdot \exp\left(\frac{d_{su}}{\delta_{suf1}} + \frac{d_t}{\delta_{tf1}}\right) = \frac{T_{f2}}{K_{r2}} \cdot \exp\left(\frac{d_{su}}{\delta_{suf2}} + \frac{d_t}{\delta_{tf2}}\right) = \frac{T_{f3}}{K_{r3}} \cdot \exp\left(\frac{d_{su}}{\delta_{suf3}} + \frac{d_t}{\delta_{tf3}}\right) \quad (20)$$

gdzie:

$$a_1 = \ln\left(\frac{T_{f1} \cdot K_{r2}}{T_{f2} \cdot K_{r1}}\right) \quad (21)$$

$$a_2 = \ln\left(\frac{T_{f1} \cdot K_{r3}}{T_{f3} \cdot K_{r1}}\right). \quad (22)$$

Rzeczywista głębokość źródła ciepła określona jest równaniem:

$$d = d_{su} + d_t + d_s \quad (23)$$

gdzie grubość warstwy skóry $d_s = 1 \div 2$ mm.

PODSUMOWANIE

Wyprowadzone zależności obowiązują w idealizowanym przypadku zakładającym brak dyfuzji ciepła. W ramach przedstawionego tematu realizowane są badania uwzględniające bardziej realny rozkład temperatur w tkance. Jednocześnie wdrażane są nowatorskie rozwiązania konstrukcyjne termometrów mikrofalowych, realizując kompensację współczynnika odbicia od tkanki [8] oraz nowoczesne oprogramowanie przeznaczone do obsługi termografu.

BIBLIOGRAFIA

1. L. Dubois, J. Pribetich, J. Falbre, M. Chive, Y. Moschetto: *Non-invasive microwave multifrequency radiometry used in microwave hyperthermia for bidimensional reconstruction of temperature patterns*. Int. Journal of Hyperthermia, 1993, Vol. 9, pp. 415–431
2. S. Mizushina, T. Shimizu, K. Suzuki, M. Kinomura, H. Ohba, T. Sugiura: *Retrieval of temperature-depth profiles in biological objects from multi-frequency microwave radiometric data*. Journal of Electromagnetic Waves and Applications, 1993, Vol. 7, pp. 1515–1548
3. H. Ohba, M. Kinomura, M. Ito, T. Sugiura, S. Mizushina: *Multifrequency microwave radiometry for non-invasive thermometry using a new temperature profile model function*. Asia Pacific Microwave Conference, 17 (1), Tokyo, 1994
4. B. Stec: *Termografia mikrofalowa w biologii i medycynie*. AND 4, 1990
5. B. Stec, A. Dobrowolski: *Analiza własności transmisyjnych tkanek biologicznych w zastosowaniu do przestrzennej termografii mikrofalowej*. VIII Krajowe Sympozjum Nauk Radiowych (URSI'96), Wrocław 1996
6. B. Stec, A. Dobrowolski: *Przestrzenna termografia mikrofalowa*. Krajowe Sympozjum Telekomunikacji'97, Bydgoszcz 1997
7. B. Stec, A. Dobrowolski: *Analiza własności transmisyjnych tkanek biologicznych w zastosowaniu do przestrzennej termografii mikrofalowej*. Biuletyn WAT, Rok XLVI, Nr 11 (543), Warszawa, 1997
8. B. Stec, M. Żurawski: *Compensated microwave thermometer for medical applications*. Asia Pacific Microwave Conference, 17 (2), Tokyo 1994

This p
diagnose of
which enab
possible to

Microv
visual infor
information
of results o

The pa
tissues tran
The analys
consisting o
blood vess
isothermal,
degrees hig
of energy to
before it re
thermograph
analyse equ
transforma

Key words:

B. STEC, A. DOBROWOLSKI

MULTIFREQUENCY MICROWAVE THERMOGRAPHY

Summary

This paper analyses the possibility of application of multifrequency microwave thermography to diagnose of higher fever affection of small area in variable diseases. In contrast to infrared thermography, which enables us to measure temperature on the surface exclusively, microwave thermography makes it possible to detect thermal heterogeneity inside the tissues.

Microwave thermograph is a device consisting of a microwave radiometer and computer block of visual information. Radiometer enables measurement of body thermal radiation research, which gives information about its absolute temperature. The block of visual information carries out digital processing of results of measurement and shows the spatial distribution of temperature inside the research tissue.

The paper presents the issue of real substance thermal radiation of microwave, and the description of tissues transmitting ability, which, due to lack of magnetic property, can be considered as dielectric loss. The analysis of thermal radiation transmissino is carried out on the basis of tissue-in-layers model, consisting of three homogeneous layers, which stand for: breast, fat and skin. Another inner areas, such as blood vessels and the problems of heat diffusion, are neglected here. All layers are assumed to be isothermal, with the exception of a small area located on the model's bottom, which temperature is a few degrees higher than the temperature of its surrounding. The area bordering surface emits an additional flux of energy towards half-space, which is damped and refracted twice, while passing through respective layers, before it reaches the aperture of thermograph receiving antenna. The length of waves used in microwave thermography ($1\text{ m} \div 3\text{ cm}$) exceeds considerably the dimension of most cells, which makes it possible to analyse equally the entire range of frequency. In conclusion of the paper there are presented the reverse transformation the results of measurement to spatial distribution of temperature inside the tissue.

Key words: microwave thermograph, radiometer, thermal radiation

Wyko
w c

rozp
obuc
oper
syste
zwła
logic
rozp
rozp
tów
i pod
o k

prze

Główn
dzielenie
a następ
przynależ
szeregowi
akcentow
informacji
dokonany
dopiero p
dokonywa
Spośród
jest na d

Wykorzystanie wieloprocessorowego układu TMS320C80 w dziedzinie przetwarzania i rozpoznawania obrazów

MIROSLAW GAJER

Katedra Automatyki

Akademia Górniczo-Hutnicza im. S. Staszica w Krakowie

Otrzymano 1999.01.10

Autoryzowano 1999.02.04

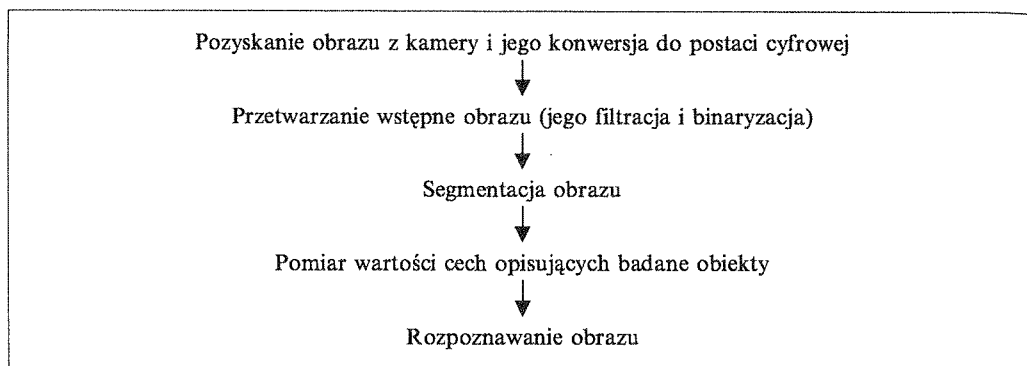
W artykule omówiono budowę systemu wizyjnego przewidzianego do automatycznego rozpoznawania typów układów scalonych na podstawie oznaczeń umieszczonych na ich obudowach. System taki może współpracować z robotem przemysłowym, dokonującym operacji montażu układów scalonych. Przedstawiono szczegółowo wszystkie etapy pracy systemu automatycznego przetwarzania i rozpoznawania obrazów. Uwagę zwrócono zwłaszcza na takie operacje, jak filtracja medianowa obrazu, jego binaryzacja, filtracja logiczna, erozja obrazu, obrót obrazu, segmentacja oraz ekstrakcja cech opisujących rozpoznawany obiekt. Omówiono także zastosowaną minimalno-odległościową metodę rozpoznawania obrazów oraz podano uzyskaną w wyniku przeprowadzonych eksperymentów sprawność procesu rozpoznawania. W artykule podano także skrótowy opis budowy i podstawowych zasad funkcjonowania wieloprocessorowego układu TMS320C80, w oparciu o który zrealizowany został opisany system wizyjny.

Słowa kluczowe: systemy wieloprocessorowe, obliczenia równoległe, automatyczne przetwarzanie i rozpoznawanie obrazów

1. WSTĘP

Głównym celem systemu automatycznego rozpoznawania obrazów jest wydzielenie z całości obrazu jedynie jego fragmentu reprezentującego nieznany obiekt, a następnie zaliczenie tego obiektu do jednej ze zdefiniowanych wcześniej klas przynależności. Zanim to jednak nastąpi, obraz musi zostać poddany całemu szeregowi operacji przetwarzających mających na celu poprawę jego jakości, zaakcentowanie i uwypuklenie istotnej z punktu widzenia procesu rozpoznawania informacji, segmentacji na fragmenty reprezentujące poszczególne obiekty oraz dokonany być musi pomiar wartości opisujących badane obiekty cech, po czym dopiero przystąpić można do procesu rozpoznawania obrazu [1]. Ciąg operacji dokonywanych na obrazie w procesie jego rozpoznawania przedstawiono na rys. 1.

Spośród wymienionych wyżej operacji jedynie pozyskanie obrazu dokonywane jest na drodze czysto sprzętowej, natomiast wszystkie pozostałe wymagają już



Rys. 1. Ciąg operacji dokonywanych na obrazie w procesie jego rozpoznawania

opracowania i realizacji programu komputerowego. Operacje te zostaną szerzej omówione na przykładzie systemu wizyjnego mającego za zadanie rozpoznawanie układów scalonych na podstawie oznaczeń na ich obudowach. Taki system wizyjny w powiązaniu z robotem przemysłowym może posłużyć do całkowitej automatyzacji stanowiska montażu układów scalonych w produkowanych urządzeniach elektronicznych. Dzięki wyposażeniu robota w system wizyjny, może on uzyskać informację o typie układu scalonego, podlegającego aktualnie operacji montażu i na tej podstawie wykonać odpowiedni ciąg czynności montażowych. Omawiany system wizyjny został zrealizowany w oparciu o współpracującą z komputerem PC kartę układu akwizycji obrazów z kamery, wyposażoną we własną wieloprocesorową jednostkę przetwarzającą firmy Texas Instruments typu TMS320C80. W punkcie kolejnym zostanie podana skrótowa charakterystyka wieloprocesorowego układu TMS320C80 oraz omówione zostaną podstawowe zasady jego funkcjonowania.

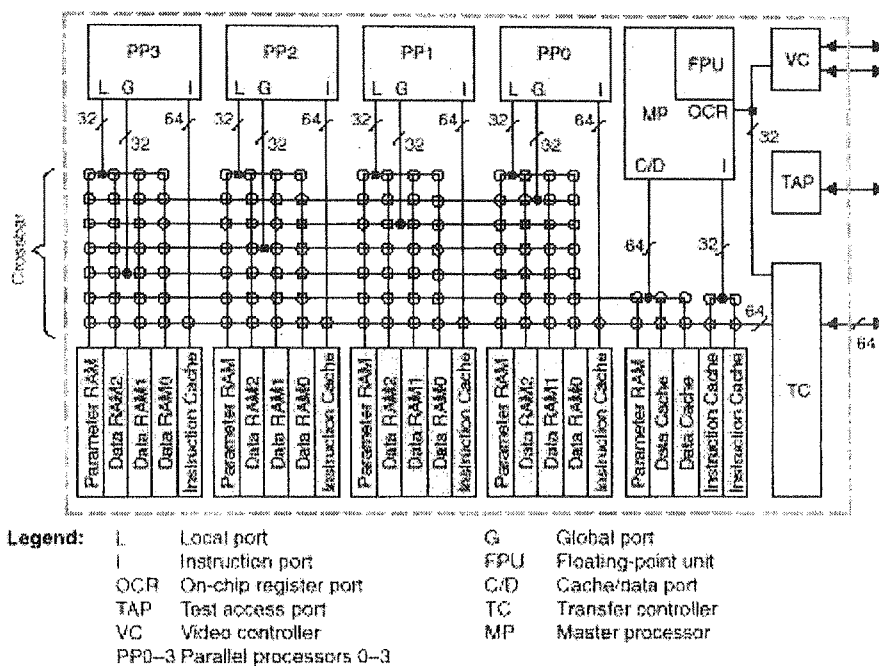
2. BUDOWA I ZASADY FUNKCJONOWANIA SYSTEMU TMS320C80

Wieloprocesorowy układ TMS320C80 jest urządzeniem wieloprocesorowym, wykonanym w postaci pojedynczego układu scalonego i składa się z następujących bloków funkcjonalnych:

- 32-bitowy zmiennoprzecinkowy procesor nadrzędny MP (ang. Master Processor) typu RISC
- cztery stałoprzecinkowe równoległe procesory PP (ang. Parallel Processor) typu DSP
- kontroler przesyłu danych do pamięci i urządzeń zewnętrznych TC (ang. Transfer Controller)
- kontroler akwizycji sygnałów wizyjnych VC (ang. Video Controller)
- blok 50 KB wewnętrznej pamięci SRAM
- przełączalna sieć połączeń międzyprocesorowych (ang. crossbar switching network)

Na rys. 2 pokazano strukturę wewnętrzną układu TMS320C80 z uwzględnieniem wszystkich wymienionych wyżej bloków funkcjonalnych [4]. Poszczególne urządzenia

systemu
poprzez
pięć ró
procesor
czonych
• L —
• I — p
• OCR
rejestr
• TAP
• VC —
• PP0-3
• G —
• FPU
• C/D —
pamięć
• TC —
zewnę
• MP —
Każe
2 KB p
przezna



Rys. 2. Budowa układu TMS320C80

systemu (procesor MP, procesory PP oraz kontroler TC) komunikują się ze sobą poprzez wspólną pamięć. Wewnętrzna pamięć SRAM systemu została podzielona na pięć równych bloków o wielkości 10 KB każdy, które zostały przypisane kolejno procesorowi MP i każdemu z czterech procesorów PP. Znaczenie napisów zamieszczonych na rys. 1 jest następujące:

- L — port lokalny procesorów PP
 - I — port służący do pobierania kodów instrukcji procesorów PP i MP
 - OCR — port, poprzez który procesor MP może dokonać zapisu lub odczytu rejestrów sterujących pracą układu kontrolera akwizycji sygnałów wizyjnych
 - TAP — port testowy, wykorzystywany przez oprogramowanie uruchomieniowe
 - VC — kontroler sterujący procesem akwizycji sygnałów wizyjnych
 - PP0-3 — równoległe stałoprzecinkowe procesory DSP
 - G — port globalny procesorów PP
 - FPU — zmiennoprzecinkowa jednostka arytmetyczno-logiczna procesora MP
 - C/D — port, poprzez który procesor MP dokonuje odczytu zawartości podręcznej pamięci danych
 - TC — układ kontrolera zarządzającego transferem danych z pamięci i urządzeń zewnętrznych
 - MP — procesor nadrzędny systemu, zmiennoprzecinkowy o architekturze RISC
- Każdy z bloków pamięci podzielony jest na pięć mniejszych bloków o wielkości 2 KB każdy, wśród których można wyróżnić bloki pamięci Parameter RAM przeznaczony do przechowywania stosu, wektorów przerwań i parametrów dla

układu TC, stanowiącego sprzęg z pamięcią zewnętrzną. Następnie wyróżnić można bloki pamięci danych Data RAM 0, Data RAM 1 i Data RAM 2, pamięć podręczną do przechowywania kodu aktualnie wykonywanego programu Instruction Cache, ponadto procesor MP posiada pamięć podręczną do przechowywania najczęściej wykorzystywanych danych Data Cache [5]. Każdy z wymienionych bloków pamięci dołączony jest do układu przełącznicy (ang. *crossbar*). Z drugiej strony z układem przełącznicy połączone są procesory. Procesory PP posiadają trzy porty (port lokalny L, port globalny G oraz port instrukcji I), procesor MP posiada dwa porty (port danych C/D oraz port instrukcji I) ponadto kontroler zewnętrznej przestrzeni adresowej posiada jeden port [6]. Takie rozwiązanie pozwala na realizację aż piętnastu równoległych dostępu do pamięci. Przełącznica łączy porty procesorów z żądanym blokiem pamięci. Jednakże nie wszystkie urządzenia mogą komunikować się z dowolnym z rozważanych bloków pamięci. Na przykład z pomocą portu I dany procesor może pobierać instrukcję jedynie ze swojego lokalnego bloku Instruction Cache, podobnie port L może służyć jedynie do pobierania danych lokalnych. W przypadku gdy w danym cyklu dwa różne urządzenia usiłują uzyskać dostęp do tego samego bloku pamięci, wówczas układ przełącznicy przydziela prawo dostępu na drodze rozstrzygania ważności priorytetów. Różne operacje realizowane przez różne urządzenia posiadają następujące priorytety:

1. przesłania blokowe PP (ang. *packet transfer*) do lub z pamięci zewnętrznej realizowane przez kontroler TC typu Low Priority PP
2. operacje dostępu do pamięci realizowane przez procesor PP
3. przesłania blokowe PP (ang. *packet transfer*) do lub z pamięci zewnętrznej realizowane przez kontroler TC typu High-Priority PP
4. operacje dostępu do pamięci realizowane przez procesor MP
5. przesłania blokowe PP (ang. *packet transfer*) do lub z pamięci zewnętrznej realizowane przez kontroler TC typu Urgent PP

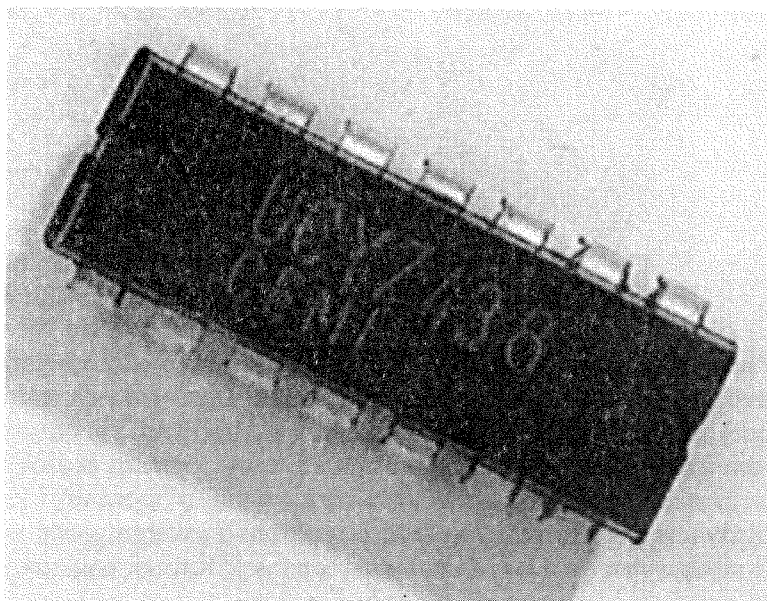
W pierwszej kolejności uzyskuje dostęp do pamięci urządzenie posiadające aktualnie wyższy priorytet. W przypadku jednoczesnej próby dostępu do danego bloku pamięci przez dwa lub więcej procesorów PP, posiadających jednakowy priorytet stosowany jest algorytm szeregowania karuzelowego (ang. *round robin*), oparty na krążącym żetonie i łańcuszku priorytetów, co zapewnia wszystkim procesorom PP równy dostęp do zasobów systemu.

W rozważanym systemie wizyjnym układ TMS320C80 umieszczony został na współpracującej z komputerem PC za pośrednictwem złącza PCI karcie SDB posiadającej rozbudowane układy akwizycyjne umożliwiające jednoczesne zbieranie obrazów z dwóch kamer, akwizycję dwóch stereofonicznych sygnałów audio, odtwarzanie stereofonicznego sygnału audio oraz wyświetlanie obrazów na ekranie monitora przy wykorzystaniu własnego sterownika graficznego. Karta SDB posiada ponadto rozbudowane bloki pamięci służące do gromadzenia zebranych i następnie przetworzonych danych obrazów bądź sygnałów audio. Pamięć zewnętrzna systemu składa się z 8 MB DRAM, 2 MB VRAM związanych ze sterownikiem graficznym karty oraz 4 MB kolejki FIFO do gromadzenia pikseli pozyskanych z kamery obrazów.

W punkcie kolejnym zamieszczono opis wykorzystania karty SDB do wykonywania operacji automatycznego przetwarzania obrazów układów scalonych i ich rozpoznawania na podstawie umieszczonych na obudowach układów scalonych napisów.

3. SYSTEM WIZYJNY ROZPOZNAJĄCY UKŁADY SCALONE

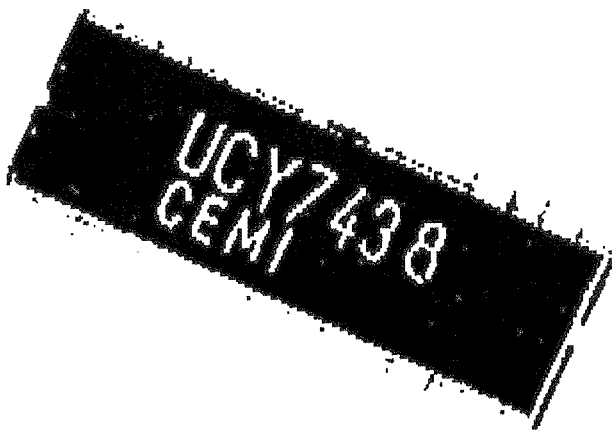
Pozyskany obraz z kamery o wymiarach 512 na 512 piksli poddany został filtracji dolnoprzepustowej, której głównym celem jest usunięcie dodanego do obrazu szumu. Szum ten jest wynikiem zsumowania się szumu termicznego generowanego w strukturze CCD kamery oraz szumu zebranego w torze transmisji sygnału [3]. Spośród wielu możliwych rozwiązań wybrano filtrację medianową, której działanie polega na posortowaniu piksli od najmniejszej do największej wartości ich stopni szarości, a następnie wybraniu wartości leżącej dokładnie pośrodku, czyli tzw. mediany posortowanych liczb i przypisaniu jej rozważanemu pikselowi. Zastosowano filtrację medianową o pięcioelementowej masce, obejmującej rozważany piksel oraz jego czterech najbliższych sąsiadów, czyli piksele leżące bezpośrednio nad nim, pod nim oraz z jego prawej i lewej strony. Zaleta filtracji medianowej w porównaniu z innymi metodami np. filtracji konwolucyjnej polega na tym, że filtr medianowy wygładzając obraz zachowuje jednocześnie krawędzie i kontury obiektów, które w przypadku filtracji konwolucyjnej ulegają rozmyciu i zamazaniu. Maskę rozważanego filtru medianowego może posiadać oczywiście większy rozmiar (najczęściej stosuje się dziewięcioelementowe maski filtrów), ale trzeba pamiętać że złożoność obliczeniowa, a co za tym idzie czas obliczeń operacji sortowania rośnie szybko z wymiarem



Rys. 3. Obraz układu scalonego po filtracji medianowej

problemu. W rozważanym tutaj przypadku zastosowanie pięcioelementowej maski filtru okazało się zupełnie wystarczające, gdyż obraz miał dużo lepszą jakość niż w przypadku gdy był pozbawiony jakiegokolwiek filtracji, a stosowanie masek filtru o większych rozmiarach nie przynosiło widocznej poprawy. Obraz układu scalonego po operacji filtracji medianowej przedstawiono na rys. 3.

Po filtracji obraz poddany został operacji binaryzacji polegającej na przypisaniu każdemu z pikseli barwy czarnej lub białej, w zależności od tego czy ich poziomy jasności znajdują się powyżej, czy też poniżej określonej wcześniej wartości progowej [2]. Z operacją tą wiąże się znaczna redukcja informacji zawartej w obrazie i w związku z tym bardzo istotną sprawą jest odpowiedni dobór wartości progowej, poniżej której wszystkie piksele staną się czarne, tak aby nie utracić istotnej z punktu widzenia procesu rozpoznawania informacji zawartej w obrazie. Doboru wartości progowej dokonano metodą eksperymentalną, ustalając go ostatecznie na 58 poziom szarości obrazu (obraz pozyskany z kamery posiadał 256 poziomów szarości, a stanowisko laboratoryjne oświetlone było dodatkowo lampką halogenową o mocy 20 W). Dzięki takiemu rozwiązaniu obraz po procesie binaryzacji przedstawiał czarną obudowę układu scalonego z białymi napisami, umieszczoną na białym tle, co w etapie kolejnym znacznie ułatwiło przeprowadzenie segmentacji obrazu. Obraz rozpoznawanego układu scalonego po jego binaryzacji zamieszczono na rys. 4.

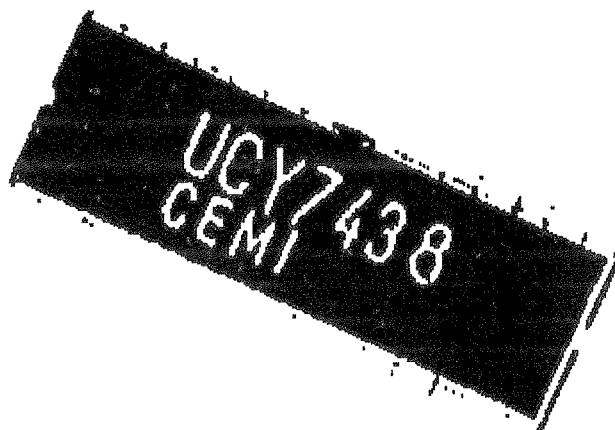


Rys. 4. Obraz po binaryzacji

Uzyskany binarny obraz wymagał filtracji logicznej, której celem było usunięcie zakłóceń w postaci pojedynczych białych pikseli na czarnym tle lub pojedynczych czarnych pikseli na białym tle oraz linii o szerokości jednego piksela umieszczonych na tle o odmiennej barwie [1]. Zastosowano złożenie dwóch filtracji logicznych o trójelementowych maskach, z których pierwsza zwana filtracją logiczną poziomą, o masce obejmującej rozważany piksel i jego najbliższych sąsiadów z prawej i lewej strony eliminuje pojedyncze piksele i pionowe linie o szerokości jednego piksela występujące na tle o odmiennej barwie. Efekt działania operacji filtracji logicznej poziomej pokazano na rys. 5.

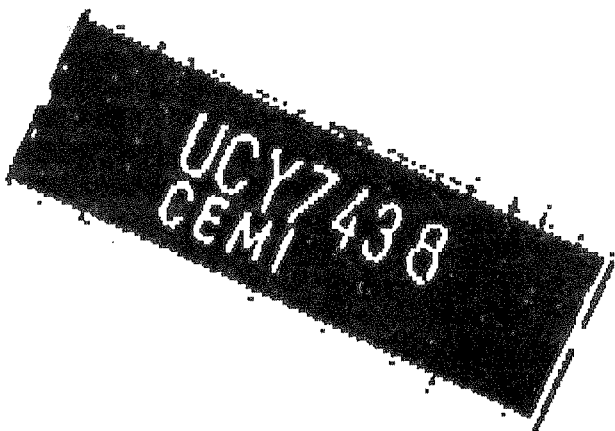
Ana
piksel c
pojedyn
o odmie
towan

Rozp
orientac
ułatwie
reprezen
obrazu c
nym zna
orientacj
obrazu,



Rys. 5. Obraz po filtracji logicznej poziomej

Analogicznie maska filtracji logicznej pionowej obejmuje rozważany aktualnie piksel oraz jego bezpośrednich sąsiadów z góry i z dołu. Eliminuje ona zarówno pojedyncze piksele jak i poziome linie o szerokości jednego piksela występujące na tle o odmiennej barwie. Obraz układu scalonego po filtracji logicznej pionowej zaprezentowano na rys. 6.



Rys. 6. Obraz po filtracji logicznej pionowej

Rozpoznawane układy scalone mogły, w ogólnym przypadku, posiadać dowolną orientację przestrzenną, to znaczy mogły być one obrócone o dowolny kąt. W celu ułatwienia dokonania operacji wydzielenia z całego obrazu jedynie jego fragmentu reprezentującego układ scalony postanowiono dokonać operacji obrotu całego obrazu o kąt o wartości równej kątowi skrócenia układu scalonego, lecz o przeciwnym znaku. Dzięki temu na obrazie wynikowym układy scalone przyjmowały taką orientację przestrzenną, że linie brzegowe obudowy, były równoległe do krawędzi obrazu, co znacznie ułatwiło jego późniejszą segmentację.

Ponieważ linia brzegowa obudowy układu scalonego nie jest idealnie gładka, ale posiada liczne rozgałęzione wypustki, koniecznym stało się poddanie obrazu, przed pomiarem kąta skrzywienia operacji erozji, która sprawia, że brzeg obudowy układu staje się bardziej podobny do linii prostej. Brak tej operacji powodował, że wartość kąta skrzywienia układu obarczona była znacznym błędem. Obraz po zastosowaniu operacji erozji można zobaczyć na rys. 7. Algorytm pomiaru kąta skrzywienia układu scalonego ma postać następującą. Ponieważ obraz podlegający przetwarzaniu w systemie wizyjnym robota posiada szerokość 512 piksli należy wybrać dwa punkty charakterystyczne leżące na górnej krawędzi obrazu po prawej i lewej stronie punktu znajdującego się w połowie jej długości i odległe od niego o szerokość 64 piksli. Zatem odcięte tych punktów wynoszą odpowiednio 192 i 320, a odległość pomiędzy nimi 128 piksli. Następnie należy posuwać się od tych punktów charakterystycznych pionowo w dół i analizować wartość każdego napotkanego po drodze piksela. Jeżeli napotkany piksel posiada wartość 255, oznacza to że jest on koloru białego i należy do tła obrazu, zatem należy kontynuować operację i sprawdzić piksel leżący poniżej. Natomiast jeżeli napotkany piksel posiada wartość 0, to posiada on kolor czarny i jest pierwszym napotkanym pikselem należącym do linii brzegowej obudowy układu scalonego i w związku z powyższym należy zapamiętać wartość jego rzędnej oraz zaprzestać sprawdzania wartości dalszych piksli. Podczas opisanej operacji zakłada się, że tło obrazu posiada wszystkie piksele w kolorze białym o wartości 255, a w szczególności nie znajduje się tam żaden piksel w kolorze czarnym o wartości 0. Założenie takie jest dopuszczalne ponieważ system wizyjny robota obserwuje układy scalone umieszczone na białym tle o wysokiej wartości piksli bliskiej 255, które w procesie binaryzacji obrazu z progiem ustalonym na 58 poziom szarości obrazu zostaną w zdecydowanej większości zaliczone do zbioru piksli znajdujących się powyżej wartości progowej i przypisana zostanie im wartość 255. Ewentualnie występujące zaburzenia struktury tła w postaci pojedynczych piksli o wartości 0 lub linii o małej szerokości zbudowanych z takich piksli, zostaną skutecznie wyeliminowane w procesie filtracji logicznej poziomej i pionowej oraz podczas operacji erozji obrazu.

Mając dane wartości dwóch zmiennych $\langle top_1 \rangle$ oraz $\langle top_2 \rangle$, będących rzędnymi dwóch punktów charakterystycznych należących do linii brzegowej obudowy układu scalonego można łatwo obliczyć wartość kąta $\langle fi \rangle$ o jaki obrócony jest układ scalony względem osi skojarzonego z obrazem układu współrzędnych, zgodnie z poniższym wzorem:

$$fi = \arctan \frac{top_1 - top_2}{128}. \quad (1)$$

Aby linie brzegowe układu scalonego stały się równoległe do brzegu obrazu, obraz musi zostać obrócony o kąt minus $\langle fi \rangle$. Trzeba w tym miejscu zaznaczyć, że operacja obrotu obrazu dokonywana jest na obrazie pierwotnym, który nie został poddany działaniu procedury erozji obrazu. Operacja erozji obrazu wygładza co prawda linię brzegową układu scalonego, pozbawiając ją wszelkiego rodzaju nieregul-

odka, ale
u, przed
y układu
wartość
sowaniu
a układu
u w sys-
punkty
e punktu
i. Zatem
nimi 128
pionowo
spotkany
y do tła
poniżej.
rny i jest
y układu
lnej oraz
zakłada
ości 255,
artości 0.
je układy
55, które
ci obrazu
ących się
entualnie
ości 0 lub
yelimino-
acji erozji

będących
ej obudo-
ocony jest
a, zgodnie

(1)

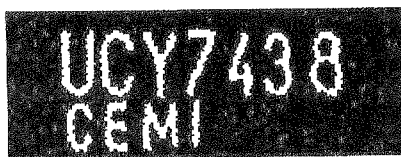
u obrazu,
naczyć, że
nie został
gładza co
u nieregul-



Rys. 7. Obraz po erozji



Rys. 8. Obraz po operacji obrotu



Rys. 9. Obraz układu UCY7438 po operacji maskowania



Rys. 10. Obraz układu UCY7453 po operacji maskowania



Rys. 11. Obraz układu SN7406n po operacji maskowania

larności, ale przy okazji pogarsza znacznie czytelność napisów znajdujących się na wierzchu obudowy układów scalonych, co jest ze względu na dalszy proces rozpoznawania tych układów scalonych całkowicie niedopuszczalne. Zatem obraz uzyskany w wyniku zadziałania procedury erozji obrazu jest produktem ubocznym całego ciągu operacji wstępnego przetwarzania obrazu, który służy tylko i wyłącznie do wyznaczenia wartości kąta $\langle \text{fi} \rangle$ i nie jest już nigdzie dalej wykorzystywany.

Operacja obrotu obrazu, względem początku układu współrzędnych może być dokonana zgodnie z poniższym wzorem:

$$x_new = x_old \cdot \cos(-\text{fi}) + y_old \cdot \sin(-\text{fi}) \quad (2a)$$

$$y_new = -x_old \cdot \sin(-\text{fi}) + y_old \cdot \cos(-\text{fi}). \quad (2b)$$

Zmienne $\langle x_new \rangle$, $\langle y_new \rangle$, $\langle x_old \rangle$, $\langle y_old \rangle$ określają wartości współrzędnych pikseli odpowiednio w obrazie przetworzonym i pierwotnym. Obrót względem dowolnego punktu wymaga odpowiedniego przeskalowania współrzędnych. Rozwiązanie to posiada jednak dość poważną wadę polegającą na tym, że w obrazie przetworzonym powstają liczne dziury, będące pikslami o współrzędnych nie wyliczonych przez powyższą procedurę dokonującą operacji obrotu obrazu. Rozwiązanie takie prowadzi do znacznego pogorszenia jakości rozpoznawanego obrazu i jest zupełnie nie do zaakceptowania w systemie wizyjnym robota.

W celu ominięcia powyższych trudności należy posłużyć się transformacją odwrotną operacji obrotu obrazu. W tym przypadku należy wziąć po kolei wszystkie piksele należące do obrazu przetworzonego, o współrzędnych $\langle x_new \rangle$, $\langle y_new \rangle$, a następnie znaleźć współrzędne $\langle x_old \rangle$, $\langle y_old \rangle$ odpowiadających im w obrazie pierwotnym piksli. Obliczając wartości współrzędnych $\langle x_old \rangle$, $\langle y_old \rangle$ należy najpierw sprawdzić czy nie leżą one poza brzegiem obrazu, czyli czy ich wartości nie wychodzą poza przedział [1, 511]. Gdyby tak było, wówczas odpowiadającym im w obrazie przetworzonym pikslom o współrzędnych $\langle x_new \rangle$, $\langle y_new \rangle$ należy przypisać wartość 255, czyli kolor biały, gdyż piksele te będą należały do tła obróconego o kąt minus $\langle \text{fi} \rangle$ obrazu. Obraz układu scalonego po operacji obrotu przedstawiono na rysunku 8.

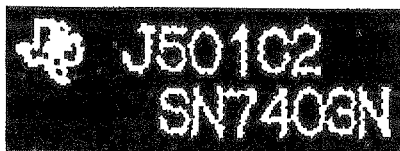
W kroku kolejnym obraz poddany został segmentacji polegającej na nałożeniu na fragment obrazu, reprezentujący układ scalony prostokątnej maski, wycinającej z całości jedynie fragment istotny z punktu widzenia procesu rozpoznawania. Obraz układu scalonego po operacji maskowania zamieszczono na rys. 9. Natomiast obrazy układów scalonych po operacji maskowania, reprezentujących siedem pozostałych klas przynależności pokazano na rys. 10–16.

Wycinek obrazu o wymiarach 200 na 75 piksli reprezentujący fragmentu obudowy układu scalonego, z widocznymi na niej oznaczeniami, dzielony jest następnie dwoma pionowymi liniami na trzy sektory, z których dwa skrajne, lewy i prawy mają wymiary 70 na 75 piksli, a sektor środkowy posiada wymiary 60 na 75 piksli. Jako cechy opisujące obraz przyjęto liczby białych piksli w poszczególnych sektorach. Należy zwrócić uwagę na fakt, że na tym etapie przetwarzania obrazów nastąpiła drastyczna redukcja informacji zawartej w obrazie, ponieważ zamiast wielu tysięcy

piksli,
z punk
dostęp
minima
rezultar
odległo
tycznych
przypis
odległo



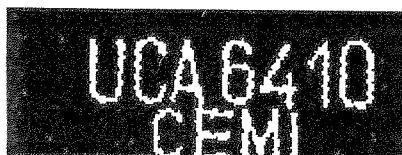
Rys. 12. Obraz układu UCY7404 po operacji maskowania



Rys. 13. Obraz układu SN7403n po operacji maskowania



Rys. 14. Obraz układu UCY7417 po operacji maskowania



Rys. 15. Obraz układu UCA6410 po operacji maskowania



Rys. 16. Obraz układu UL1201 po operacji maskowania

piksli, cały obraz opisywany jest jedynie przez trzy liczby niosące całą istotną, z punktu widzenia procesu rozpoznawania, informację o obrazie. Spośród wielu dostępnych metod rozpoznawania obrazów zdecydowano się na wybór metody minimalno-odległościowej, ze względu na jej matematyczną prostotę i dobre rezultaty rozpoznawania. Istota tej metody polega na pomiarze w przestrzeni cech odległości jaka dzieli rozpoznawany obiekt od punktów wzorcowych charakterystycznych dla każdej ze zdefiniowanych uprzednio klas obiektów. Ostatecznie obiekt przypisywany jest do tej klasy, od której w przestrzeni cech dzieli go najmniejsza odległość.

Taki opis obrazu jedynie w postaci trzech liczb naturalnych, okazał się całkowicie wystarczający do jego prawidłowego rozpoznania, co potwierdziły liczne eksperymenty. Przeprowadzono łącznie kilkaset prób rozpoznawania ośmiu różnych układów scalonych, z czego ponad 99% rozpoznań kończyło się sukcesem, czyli wytypowaniem właściwej klasy przynależności danego układu scalonego. Wyniki rozpoznawania zamieszczono w tabeli 1. Tabela z zamieszczonymi wynikami rozpoznawania układów scalonych wymaga krótkiego komentarza. Otóż rozpoznawany układ scalony mógł znajdować się w dwóch możliwych położeniach, albo w pozycji tzw. normalnej, albo w tzw. pozycji odwróconej (w przypadku gdy napis na jego obudowie był odwrócony o 180 stopni). Zatem każda z klas przynależności musiała zostać podzielona na dwie podklasy, reprezentujące powyższe przypadki, a uzyskane wyniki rozpoznawania zamieszczono w tabelach również dla każdej podklasy oddzielnie.

T a b e l a 1

Wyniki rozpoznawania układów scalonych

Typ układu scalonego	Liczba rozpoznań prawidłowych dla pozycji normalnej układu scalonego	Liczba rozpoznań błędnych dla pozycji normalnej układu scalonego	Liczba rozpoznań prawidłowych dla pozycji odwróconej układu scalonego	Liczba rozpoznań błędnych dla pozycji odwróconej układu scalonego
7453	50	0	50	0
7406N	50	0	50	0
7404	50	0	50	0
7403N	50	0	50	0
7417	50	0	49	1
7438	50	0	50	0
6410	48	2	50	0
1201	50	0	50	0

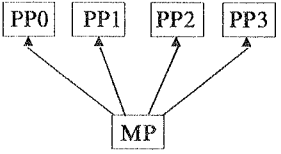
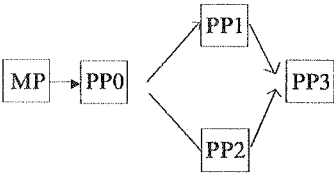
4. ZRÓWNOLEGIENIE OBLICZEŃ

Wszystkie operacje związane z przetwarzaniem i rozpoznawaniem obrazów są bardzo pracochłonne i do ich wykonania w sposób sekwencyjny procesor MP potrzebował aż 6,3 sekundy. Podjęto zatem próbę zrównoleglenia obliczeń oraz przetestowano kilka możliwych do zrealizowania w obrębie układu TMS320C80 konfiguracji wieloprocesorowych. Ogólnie rzecz ujmując zastosowano dwa odmienne podejścia. W pierwszym przypadku dokonano dla systemu TMS320C80 implementacji architektury typu SIMD (ang. *single instruction multiple data*). Przetwarzany obraz podzielony został na cztery równe części, które przypisano do odpowiednich procesorów PP. Zatem wszystkie procesory PP wykonywały te same operacje, ale na

odmiennych zbiorach danych. Rozwiązanie alternatywne polegało na implementacji architektury typu MIMD (ang. *multiple instruction multiple data*), gdzie z pięciu procesorów systemu TMS320C80 utworzony dwa typy struktur potokowych. Zrealizowane architektury wieloprocessorowe opisano dokładniej w tab. 2.

Tabela 2

Struktury systemów wieloprocessorowych zrealizowanych w oparciu o układ TMS320C80

Typ konfiguracji wieloprocessorowej	Zadania wykonywane przez poszczególne procesory
Macierzowa 	Wszystkie procesory PP wykonywały te same zadania, ale każdy na innej ćwiartce obrazu: • filtracja medianowa • binaryzacja obrazu • filtracja logiczna pozioma i pionowa • erozja obrazu • obrót obrazu procesor MP przydzielał poszczególnym procesorom PP zadania, synchronizował pracę całości, następnie dokonywał ekstrakcji cech, obliczał odległość w przestrzeni cech i typował klasę przynależności
Potokowa 	Z procesorów utworzono strukturę potokową o pięciu stacjach, w których wykonywane były następujące operacje: 1. MP: filtracja medianowa 2. PP0: binaryzacja obrazu, filtracja logiczna pozioma 3. PP1: filtracja logiczna pionowa, erozja obrazu 4. PP2: obrót obrazu 5. PP3: ekstrakcja cech, pomiar odległości w przestrzeni cech i typowanie klasy przynależności
Hybrydowa 	Z procesorów utworzono strukturę potokową o czterech stacjach, przy czym stacja trzecia składała się z dwóch procesorów PP pracujących współbieżnie na połówkach obrazu. W poszczególnych stacjach potoku wykonywane były następujące operacje: 1. MP: filtracja medianowa 2. PP0: binaryzacja obrazu, filtracja logiczna pozioma i pionowa, erozja obrazu 3. PP1, PP2: obrót obrazu 4. PP3: ekstrakcja cech, pomiar odległości w przestrzeni cech i typowanie klasy przynależności

5. ZAKOŃCZENIE

W artykule omówiono budowę systemu wizyjnego robota przemysłowego, przewidzianego do pracy na stanowisku automatycznego montażu układów scalonych. Omówiony system wizyjny zrealizowany został w oparciu o wieloprocessorowy układ TMS320C80, w ramach którego zrealizowano kilka różnych struktur wieloprocessorowych. Najkrótszy czas obliczeń uzyskano dla konfiguracji macierzowej, gdzie

wszystkie procesory PP wykonywały te same operacje, ale na różnych zbiorach danych i były przez cały czas równomiernie obciążone. Zdecydowanie najgorsza okazała się konfiguracja potokowa, ze względu na różny czas trwania operacji wykonywanych w poszczególnych stacjach potoku. Operacja obrotu obrazu wymagała najdłuższego czasu obliczeń, co powodowało konieczność wstrzymania pracy pozostałych stacji potoku w celu zapewnienia synchronizacji. Pewną poprawę przyniosło tutaj zastosowanie konfiguracji hybrydowej, gdzie stację potoku dokonującą obrotu obrazu zrealizowano w postaci dwóch pracujących współbieżnie na połówkach obrazu procesorów PP.

Za trafnością wyboru minimalno-odległościowej metody rozpoznawania obrazów przemawia uzyskana wysoka sprawność procesu rozpoznawania wynosząca ponad 99%. Ponadto należy jeszcze zwrócić uwagę na uniwersalność zastosowanej w niniejszej pracy metody rozpoznawania, polegającą na całkowitym uniezależnieniu efektywności procesu rozpoznawania układów scalonych od charakteru oznaczeń umieszczonych na obudowach układów scalonych (w szczególności mogą to być znaki alfanumeryczne, znaki alfabetu japońskiego, znaki firmowe korporacji produkujących układy scalone itp.). Jedynym warunkiem poprawnego działania systemu jest zachowanie przez firmy produkujące układy scalone identycznej postaci oznaczeń na obudowach wszystkich układów scalonych tego samego typu. Jednakże jak wykazały przeprowadzone eksperymenty oznaczenia umieszczone na układach scalonych charakteryzują się tak dużym stopniem powtarzalności (dobra technika wykonania druku oznaczeń), tak że reprezentowany system automatycznego rozpoznawania obrazów nie miał najmniejszych kłopotów z zaliczeniem testowych układów scalonych (czyli takich w oparciu o które system nie był wcześniej uczony) do właściwej klasy przynależności.

Należy zwrócić jeszcze uwagę na fakt, że przyjęto stałą odległość kamery od rozpoznawanego układu scalonego. Rozwiązanie takie jest jednakże często stosowane w przypadku systemów wizyjnych robotów przemysłowych, gdzie kamera obserwująca przestrzeń pracy robota umieszczana jest na stałym postumencie (rozwiązanie alternatywne polega na umieszczeniu kamery bezpośrednio na ramieniu robota).

Oprogramowanie dla wieloprocessorowego systemu TMS320C80 zbudowane jest z wielu modułów, które kompilowane są przy pomocy dwóch kompilatorów (oddzielne kompilatory dla procesora nadrzędnego MP oraz dla procesorów równoległych PP), a następnie łączone w jeden moduł, który w kolejnym etapie ładowany jest do pamięci programu karty SDB i system automatycznie rozpoczyna jego wykonywanie. Komunikacja pomiędzy poszczególnymi jednostkami przetwarzającymi system TMS320C80 odbywa się poprzez wspólną pamięć za pośrednictwem przesyłanych wiadomości.

BIBLIOGRAFIA

1. R. Tadeusiewicz: *Systemy wizyjne robotów przemysłowych*. WNT, Warszawa 1992
2. Ch. Watkins, A. Sadun, S. Marenka: *Nowoczesne metody przetwarzania obrazu*. WNT, Warszawa 1995

3. J. W
4. TMS
5. TMS
6. TMS
7. TMS
8. TMS

Pracę w
puterow

THE US

In th
of an int
system ca
integrate
threshold
Further t
were pre
presented

Key word

3. J. Woźnicki: *Podstawowe techniki przetwarzania obrazu*. WKŁ, Warszawa 1996
4. TMS320C80 Code Generation User's Guide. Materiały katalogowe firmy Texas Instruments
5. TMS320C80 Master Processor User's Guide. Materiały katalogowe firmy Texas Instruments
6. TMS320C80 Parallel Processor User's Guide. Materiały katalogowe firmy Texas Instruments
7. TMS320C80 System-Level Synopsis. Materiały katalogowe firmy Texas Instruments
8. TMS320C80 Multitasking Executive User's Guide. Materiały katalogowe firmy Texas Instruments

Pracę wykonano w ramach grantu KBN 8T 11A 03914 „Analiza i projektowanie systemów komputerowych czasu rzeczywistego o różnym stopniu rozproszenia”.

M. GAJER

THE USAGE OF THE TMS320C80 MULTIPROCESSOR SYSTEM IN THE FIELD OF IMAGE PROCESSING AND RECOGNITION

S u m m a r y

In the article the vision system was described that can be used to the automatic recognition of the type of an integrated circuit basing on the descriptions placed on the integrated circuit packages. Such a vision system can co-operate with an industrial robot and can be used to perform the montage operation of these integrated circuits. In the article all the necessary image processing operations such as median filtering, thresholding, logic filtering, image erosion, segmentation and feature extraction were described in details. Further the minimal distance image recognition method was discussed and the obtained recognition results were presented. In the other part of the article the multiprocessor system TMS320C80 for which the presented algorithms were implemented was also described.

Key words: multiprocessor systems, parallel processing, automatic image processing and recognition

2

i an
asp
szcz
Ins
sto
prz
nyc
tech

Sło
ukł

Celer

rów wizy
wych or
przeznac
stawowy
przetwar
(i wskaza
MISD. W
pory w li
wego prz
pracach c
bibliogra
systemów
systemów

Architektura MISD procesorów specjalizowanych w systemach wizyjnych czasu rzeczywistego na tle aktualnych prac badawczych

KAZIMIERZ WIATR

Katedra Elektroniki, Akademia Górniczo-Hutnicza im. S. Staszica w Krakowie

Otrzymano 1999.01.25

Autoryzowano 1999.03.15

W pracy przedstawiono przegląd publikacji dotyczących specjalizowanych procesorów i architektur, przeznaczonych do realizacji operacji przetwarzania obrazów. Podstawowym aspektem rozważań jest czas przetwarzania danych wizyjnych i z tego punktu widzenia na szczególne wyróżnienie zasługuje nie stosowana wcześniej architektura MISD (ang. *Multiple Instruction-stream Single Data-stream*). W przeglądzie uwzględniono szereg aspektów stosowania procesorów specjalizowanych, od specjalnego konstruowania algorytmów, przeznaczonych do implementacji w strukturach sprzętowych, aż po omówienie współczesnych nurtów konstruowania układów programowalnych FPGA, będących płaszczyzną technologiczną implementacji procesorów specjalizowanych.

Słowa kluczowe: procesory specjalizowane, architektury systemów, przetwarzanie obrazów, układy programowalne.

1. WSTĘP

Celem prac prowadzonych przez autora nad badaniem specjalizowanych procesorów wizyjnych było zaproponowanie optymalnej struktury elementów obliczeniowych oraz architektury ich połączeń w wieloprocessorowym systemie wizyjnym przeznaczonym do wstępnego przetwarzania obrazów w czasie rzeczywistym. Podstawowym kryterium stosowanym do oceny badanych systemów była szybkość przetwarzania. Przedstawione w pracach [324, 335] rozważania dotyczą przebadania (i wskazania jako najlepszej w pewnych zastosowaniach) wybranej architektury typu MISD. Warto dodać, że systemów o tego typu architekturze nie proponowano do tej pory w literaturze dotyczącej sprzętowych metod usprawniania procesu komputerowego przetwarzania obrazów, co powoduje, że wyniki badań opisywanych w tych pracach dość istotnie wykraczają poza obszar wiedzy dostępnej w znanej autorowi bibliografii przedmiotu. Badania przeprowadzono na przykładzie szczególnej klasy systemów tj. takich systemów do przetwarzania i analizy obrazów, które są częścią systemów stosowanych dla potrzeb automatyki i sterowania w czasie rzeczywistym.

Narzuca to bardzo ostre wymagania w zakresie szybkości działania tworzonego systemu, ma jednak także pewne aspekty ułatwiające jego zaprojektowanie. Konkretnie znane z góry przeznaczenie rozważanego w pracy systemu poważnie ogranicza stopień jego komplikacji, ponieważ z góry znane są rodzaje operacji, jakie trzeba będzie wykonać na obrazie w trakcie jego przetwarzania. Takie ograniczenie pola zastosowań pozwoliło efektywnie sformułować ogólne zasady umożliwiające skuteczne opracowanie dedykowanej architektury wieloprocesorowej związanej ze wstępnym przetwarzaniem sygnałów wizyjnych w projektowanych systemach. Założeniem prowadzonych przez autora poszukiwań było także oparcie prac badawczych na realnych możliwościach implementacji w dostępnych technologiach. Niniejsze opracowanie ma na celu szersze odniesienie tych zagadnień do aktualnego stanu wiedzy i prowadzonych prac badawczych w tym zakresie.

2. SYSTEMY WIZYJNE CZASU RZECZYWISTEGO

Systemy wizyjne są ważną klasą systemów obliczeniowych. Przeznaczenie, ogólną koncepcję, realizowane funkcje, parametry i własności tych systemów przedstawiono w wielu opracowaniach [11, 19, 68, 84, 98, 132, 136, 138, 168, 170, 187, 195, 223, 229, 235, 243, 251, 260, 274, 275, 276, 277, 288]. Podstawowym elementem systemów wizyjnych jest kamera pozyskująca obraz do dalszego przetwarzania. Przełomem w technice kamer wizyjnych było opracowanie półprzewodnikowego czujnika CCD [59, 228, 250] używanego obecnie powszechnie zamiast wcześniej stosowanej lampy analizującej — widikonu. Aktualnie budowanych jest wiele odmian czujników CCD, głównie z uwzględnieniem zróżnicowanej rozdzielczości (liczba kolumn $\times$ liczba wierszy), częstotliwości pracy (liczba obrazów na sekundę) i czułości (związanej m.in. z rozmiarami geometrycznym matrycy półprzewodnikowej) [130, 185, 211, 252, 278, 351]. Dla potrzeb niniejszego opracowania ograniczono się jedynie do rozpatrywania tzw. wielkich standardów (CCIR, RS-170), które dominują na światowym rynku wizyjnym. Ważne jest jednak ciągle śledzenie wzrostu rozdzielczości kamer (np. HDTV — ang. *High Density TV*) oraz wzrostu liczby przekazywanych przez nie obrazów na sekundę [127, 130, 211, 217, 256]. Obie te tendencje prowadzą do poważnego wzrostu częstotliwości napływania danych obrazowych (piksli) do systemu wizyjnego, co powoduje dalszy wzrost zapotrzebowania na szybkie metody przetwarzania obrazów, w tym także rozważane przez autora metody wstępnego przetwarzania.

Z powyższym zagadnieniem wiąże się problem pojęcia „czasu rzeczywistego” dla rozważanych systemów wizyjnych. Przyjęto uważać za systemy czasu rzeczywistego te systemy, w których czas reakcji systemu na zmiany na obserwowanym obiekcie jest niezauważalny (w wymiarze skali czasu dla zmian zachodzących na tym obiekcie). Za bezpośrednie odniesienie skali czasu takich systemów uznano czas trwania jednego obrazu. W związku z tym czas trwania jednego piksla, wynikający z częstotliwości pracy przetwornika A/C, przyjęto jako podstawową jednostkę czasu [311, 318]. Zachowanie tak wygórowanych wymagań odnośnie czasu reakcji systemu wizyjnego

nie zawsze jest konieczne. Można sobie wyobrazić system dla potrzeb automatyki pracujący znacznie wolniej, jeśli stałe czasowe sterowanego obiektu są odpowiednio duże. Jednak postawienie zadania w taki sposób, jak podano wyżej, gwarantuje, że ze strony procesu przetwarzania obrazu nie będzie dalszych ograniczeń szybkości działania sensora wizyjnego, bo limitującym czynnikiem jest tu i tak proces akwizycji obrazu.

Analizowany obraz zorganizowany jest w systemach wizyjnych najczęściej w postaci tabeli [187, 216, 267, 273, 343], a analizie poddawany jest zazwyczaj tylko centralny fragment obrazu, przy czym ważne jest aby analizowane piksele miały kwadratowy kształt [311, 324].

3. ALGORYTMY PRZETWARZANIA W SYSTEMACH WIZYJNYCH

W systemach wizyjnych autor wprowadził umowny podział realizowanych operacji na trzy poziomy. Podział taki znajduje oparcie także w innych opracowaniach [55, 101, 295, 299]. W [324, 335] wyraźnie sprecyzowano rodzaj realizowanych na poszczególnych poziomach operacji. Takie rozgraniczenie algorytmów pozwala na lepsze dopasowanie architektury procesorów obliczeniowych i ich połączeń do realizowanych przez nie zadań. Rozważane algorytmy obejmują obliczenia związane ze wstępnym przetwarzaniem obrazów i w tym zakresie autor korzystał z dostępnych opracowań dotyczących przetwarzania obrazów [18, 36, 46, 67, 68, 69, 89, 98, 132, 138, 168, 170, 181, 187, 194, 223, 229, 230, 235, 245, 251, 253, 260, 273, 274, 275, 276, 298, 343] nie próbując wprowadzać nowych metod.

Należy zauważyć, że przyjęty podział operacji przetwarzania i analizy obrazów, posiada swoje odzwierciedlenie w odmiennym sposobie ich realizacji, wyraża się to poprzez zróżnicowane podejście do struktury i zasobów sprzętowych, przeznaczonych do ich realizacji. Najbardziej odpowiednią architekturą, przeznaczoną do realizacji zróżnicowanych algorytmów, jest zróżnicowana, heterogeniczna architektura systemu [39, 75]. Opracowana przez autora architektura przeznaczona jest do współpracy z pozostałą częścią systemu, realizującą wyższe poziomy przetwarzania i zrealizowaną w odmienny sposób. Warto zauważyć, że przetwarzanie wstępne bazuje na informacji o całym obrazie (standardowo 512×512 pikseli), co w przypadku systemu działającego w czasie rzeczywistym nakazuje zmaksymalizować wysiłki zmierzające do przyspieszenia obliczeń. Dalsze etapy przetwarzania często analizują ograniczoną, specjalnie wyselekcjonowaną część informacji [101], co czyni łatwiejszym uzyskanie dla nich zadowalającej szybkości działania.

Wiedza na temat algorytmów wstępnego przetwarzania danych wizyjnych jest znana od wielu lat, lecz nadal prowadzone są prace związane z dalszym jej rozwijaniem i udoskonalaniem. Prace te dotyczą formułowania nowych i modyfikacji wcześniej znanych rozwiązań. Przykładem niestandardowego podejścia jest filtracja medianowa, a jej modyfikacje związane są z badaniem kolejnych algorytmów [15, 38, 140, 147, 154, 167, 236, 296, 346, 355, 360, 361] oraz analizą wpływu wielkości i kształtu okna na skuteczność filtracji obrazu wizyjnego [244]. Analogiczne prace

dotyczy także innych operacji: konwolucji [220], detekcji krawędzi [121, 157, 239], binaryzacji [199], filtracji obrazów binarnych [264] oraz operacji morfologicznych [52, 119, 137, 144, 183, 184, 202, 262, 297].

Intensywnie prowadzone są także prace związane z takim formułowaniem zapisu realizacji znanych wcześniej algorytmów, aby był on jak najlepiej dostosowany do realizacji w strukturach równoległych (układy wieloprocessorowe, układy specjalizowane VLSI) [44, 106, 108, 124, 134, 150, 156, 272]. Dorobek badawczy w tym zakresie jest bardzo przydatny dla omawianych prac, ponieważ realizacja algorytmów w układach programowalnych FPGA, poza swoją specyfiką, może czerpać z dorobku implementacji algorytmów w specjalizowanych układach VLSI [87, 114, 118, 155, 171, 209, 213]. Interesujące w tym zakresie są opracowania, które powstały specjalnie dla układów FPGA [14, 77, 164, 175, 200, 234, 254, 366].

4. ARCHITEKTURY SYSTEMÓW WIZYJNYCH

Architektury systemów komputerowych decydują o ich szybkości i możliwościach użytkowych [7, 8, 81, 109, 126, 139, 195]. W szczególności dotyczy to systemów przeznaczonych dla robotyki [57, 104, 112, 143, 146, 201] i systemów wizyjnych [71, 105, 145, 153, 188]. Systemy czasu rzeczywistego dla szybkiej robotyki cechuje dążność do uzyskania maksymalnej liczby operacji wykonywanych w jednostce czasu. Osiągane to jest poprzez ciągle zwiększanie szybkości zegara traktującego pracą procesorów (ograniczone możliwościami technologicznymi) i doskonalenie struktury wewnętrznej procesorów. Niezależnie rozwijana jest działalność badawcza związana ze stosowaniem równoległych struktur przetwarzających [37, 42, 53, 70, 125, 148, 159, 169, 174, 196, 249, 299, 332, 337]. Zwiększenie stopnia zrównoleglenia obliczeń poprzez zwielokrotnienie liczby procesorów wiąże się jednak z trudnościami wymiany informacji pomiędzy tymi procesorami i dlatego konieczne są poszukiwania takich topologii połączeń (architektur) systemów obliczeniowych, które będą najbardziej efektywne.

Przegląd architektur wieloprocessorowych, pod kątem topologicznym, należy przeprowadzać uwzględniając jakościowo różne rozwiązania, stosowane szczególnie w systemach wizyjnych. Porządkująca w tym zakresie jest klasyfikacja wprowadzona przez Flynn'a [81], rozróżniająca systemy wieloprocessorowe według liczby strumieni instrukcji i danych. Są to systemy SISD (ang. *Single Instruction-stream Single Data-stream*), SIMD (ang. *Single Instruction-stream Multiple Data-stream*), MISD (ang. *Multiple Instruction-stream Single Data-stream*) i MIMD (ang. *Multiple Instruction-stream Multiple Data-stream*). Znane są także inne klasyfikacje systemów wieloprocessorowych. Jedną z nich, przeprowadzoną przez Ercegovać'a i Lang'a, za podstawę klasyfikacji przyjmuje strukturę algorytmów obliczeń, rozróżniając organizację komputerów: sekwencyjną, grupowo-sekwencyjną, z luźno powiązanymi strumieniami i ogólnie współbieżną (cytowane za [148]). Inna klasyfikacja, zaproponowana przez Treleaven'a i Lima, uwzględnia rodzaje mechanizmów sterowa-

nia (s
cem)
katów
kacją
archi
waża
system
wyod
kich
uznaj
czne.

Pi
proce
Cellu
tipe
ASAI

G
puter
tami
także
Sarno
i syste
w tym
miesz
i SPM

W
zasług
Oc
sor),
obraz
Video
Parall
turach
plexer

W
i akwi
grabbe
modu
wstep
no-log
rzeczy
grabbe
[61, 62
stanow

nia (samosterowanie, sterowanie danymi, sterowanie na żądanie, sterowanie wzorcem) i rodzaje mechanizmów wymiany danych (wspólne dane, przesyłanie komunikatów) (cytowane za [269]). Klasyfikacja opracowana przez Duncana jest modyfikacją klasyfikacji Flynn'a, powstała poprzez wyeliminowanie z niej niektórych architektur i uzupełnienie jej o nowe elementy [72]. Tanenbaum w swoich rozważaniach całą uwagę koncentruje na strukturach MIMD, wyróżniając wśród nich systemy ściśle i luźno powiązane (z pamięcią wspólną i prywatną, a w tych z kolei wyodrębniając systemy szynowe i przełączane) [279]. Cechą wspólną prawie wszystkich klasyfikacji systemów jest to, że odnoszą się one do klasyfikacji Flynn'a, uznając tym samym jej pierwszeństwo nie tylko historyczne, ale także metodologiczne.

Pierwszym reprezentantem grupy SIMD jest architektura oparta na matrycach procesorów (ang. *array machines*) zastosowana m.in. w systemach CLIP4 (ang. *Cellular Logic Array for Image Processing*), Connection Machine, PICAP3, Centipede i CLIP7 [138, 214, 280]. Najnowsze opracowania w tej grupie to systemy: ASAP [29], CAP [362], PIRAM [280] i IMAP [214].

Grupa MIMD jest reprezentowana przez systemy Ncube i architektury transputerowe [35, 138]. Kolejną grupę stanowią struktury systoliczne, której reprezentantami są systemy WARP [138], MATRIX [198] i inne [208, 257]. Na uwagę zasługuje także cytokomputer (system Cyto HSS) [138] i architektura piramidalna (system Sarnoff Pyramid Vision System oparty na procesorach Sarnoff Pyramid Chip i system PAPIA oparty na Square Processor Arrays) [138]. Najnowsze propozycje w tym zakresie to system „Micro Data Flow” w grupie MIMD [42] oraz architektury mieszane: M-SIMD (ang. *Multiple Single Instruction-stream Multiple Data-stream*) i SPMD (ang. *Single Program Multiple Data*) [92, 198].

W ramach grupy SISD, w zastosowaniach do systemów wizyjnych, na uwagę zasługują komputery wektorowe CRAY-1, CYBER 205 i NEC SX [148].

Odrębną grupę stanowią procesory sygnałowe DSP (ang. *Digital Signal Processor*), w tym procesory DSP opracowywane specjalnie dla potrzeb przetwarzania obrazów. Opracowania w tym zakresie były realizowane w strukturach VLSI jako Video DSP [353], VSP (ang. *Video Signal Processor*) [48, 96], HiPAR (ang. *High Parallel DSP*) [247] i ISMPs (ang. *Image Signal Multiprocessor*) [205] oraz w strukturach FPGA, jako DSPLM (ang. *DSP Logic Module*) i MBLM (ang. *Multiplexer-Based Logic Module*) [4].

W pracach [3, 6, 63, 76, 192] przedstawiono wybrane moduły pozyskiwania i akwizycji sygnału wizyjnego, najbardziej charakterystyczne dla modułów frame grabber'ów. Z punktu widzenia niniejszych rozważań, najbardziej interesujące są te moduły akwizycji obrazu, które wyposażono w dodatkowe funkcje, związane ze wstępnym przetwarzaniem tego obrazu (tablic LUT [3], jednostka arytmetyczno-logiczna [63], mikroprocesor [6, 76]). Moduły te stosuje się w systemach czasu rzeczywistego [85, 136, 240, 241]. Ważne są także opracowania uniwersytecie frame grabber'ów (MAVEC [141], DSPACE [330]) i całych systemów wizyjnych (ACRC [61, 62], BDFP [7], CESARO [101, 218, 275], PAPRICA [31], PIPE [313, 314]), stanowiące w wielu przypadkach wzorzec dla przyszłych rozwiązań przemysłowych.

5. DEDYKOWANA ARCHITEKTURA DO WSTĘPNEGO PRZETWARZANIA OBRAZÓW

Poszukiwania nowych architektur wieloprocessorowych, o bardzo dużych mocach obliczeniowych, są przedmiotem nieustających i intensywnych prac badawczych [11, 19, 55, 66, 160, 166, 259, 270, 215, 338, 339]. Przykładem obszaru, w którym manifestuje się potrzeba poszukiwania nowych struktur dla tych systemów są zastosowania wizyjne. Przedmiotem szczegółowych badań autora jest analiza oraz dobór architektury elementów obliczeniowych i ich połączeń. Jako punkt wyjścia dla tych rozważań przyjęto klasyfikację systemów wieloprocessorowych, wprowadzoną przez Flynn'a [81]. Przeprowadzona analiza zadań i funkcjonowania systemu wizyjnego uwzględnia przede wszystkim czas realizacji zadanych operacji, z uwzględnieniem czasu obliczeń i czasu transmisji danych oraz rozkazów. Przeprowadzone badania wykazują jednoznacznie poważną przewagę procesorów specjalizowanych nad innymi rozwiązaniami. Przewagę tę osiąga się poprzez wielokrotne przyspieszenie obliczeń w specjalizowanym procesorze oraz poprzez opracowanie specjalnej potokowej architektury ich połączeń, dzięki czemu osiąga się całkowite wyeliminowanie opóźnień związanych z przesyłaniem danych obrazowych pomiędzy procesorami. Zaprezentowane wyliczenia teoretyczne znalazły poparcie w przeprowadzonych symulacjach i pomiarach weryfikacyjnych na procesorach testowych [321, 323, 324, 325, 327, 328, 329].

Czas wykonania poszczególnych operacji zależy wprost od realizujących te operacje elementów obliczeniowych. Celem maksymalnego skrócenia tego czasu, w ramach prac prowadzonych przez autora opracowano specjalne procesory specjalizowane, wykonujące tylko jedną instrukcję, zależną od schematu ideowego procesora. W ten sposób całkowicie wyeliminowano opóźnienia związane z przesyłaniem i dekodowaniem kodu rozkazu (praktycznie kilku rozkazów składających się na realizację każdego, nawet bardzo prostego algorytmu) oraz znacznie przyspieszono wykonanie tej operacji poprzez dedykowany sprzęt. W celu jak najszybszej realizacji potrzebnych w przetwarzaniu obrazów operacji arytmetycznych opracowano szereg specjalnych struktur elementów procesora, eliminując lub poważnie ograniczając wszelkie przeniesienia szeregowo (poważnie wydłużające czas obliczeń) oraz maksymalnie rozszerzając strukturę obliczeniową bramek celem skrócenia drogi sygnału [324, 327, 333, 334, 335].

Procesory specjalizowane znane są z implementacji w strukturach VLSI [21, 51, 64, 73, 99, 100, 106, 133, 149, 150, 151, 210, 212, 225, 300]. W szczególności wykorzystywano je do realizacji operacji wizyjnych: filtracji medianowej [47, 90, 140, 266], konwolucji [110, 248], detekcji krawędzi [80], liczenia histogramu [1, 203, 247], operacji morfologicznych [152, 261, 262, 362], szukania minimum-maksimum [161], szkieletyzacji [29]. Struktury takich opracowań często bazowały na architekturze systolicznej [78, 90, 131].

W późniejszym okresie, wraz z opracowaniem technologii układów programowalnych FPGA, powstały kolejne opracowania [34, 85, 101, 102, 221, 287]. Opracowane procesory realizowały operacje filtracji medianowej [41], korelacji [122], konwolucji [23], wydobywania krawędzi [177] oraz pracowały jako mnożarki [25].

Znane są także opracowania firmowe specjalizowanych procesorów, przeznaczonych do realizacji konkretnych operacji przetwarzania obrazów. Wśród nich są przede wszystkim procesory realizujące operację konwolucji: A110 [258, 312, 324], HSP48908 [113, 312] i PDSP16488 [91, 231, 312]. Ponadto znane są procesory liczące histogram — HSP48410 [113], dokonujące detekcji krawędzi — PDSP16401 [91, 231] oraz filtry obrazowe — HSP48901 [113]. Przedstawione procesory firmowe realizują ściśle określone funkcje i nie ma możliwości ich rekonfigurowania, co zapewniają implementacje w układach FPGA.

Poszukiwania w zakresie procesorów i systemów rekonfigurowanych są obecnie bardzo intensywnie prowadzone [13, 20, 107, 166, 197, 291, 352, 356], co jest następstwem z jednej strony wzrastających potrzeb w zakresie wymaganych mocy obliczeniowych, z drugiej zaś nowych możliwości implementacyjnych w układach programowalnych FPGA [20, 107, 291, 356]. Prowadzone są prace ogólne, o charakterze rozważań teoretycznych i symulacyjnych dla architektur rekonfigurowalnych [13], a pierwsze próby implementacji takich architektur do przetwarzania obrazów podejmowano już w strukturach VLSI [197]. Kolejnym problemem jest potrzeba szybkiej rekonfiguracji. Czas na rekonfigurację układu programowalnego może być bardzo ograniczony przez zewnętrzne uwarunkowania. Przykładowo, w systemie wizyjnym, może to być czas powrotu pionowego plamki na obrazie. Jednak na przykładzie opracowanej architektury potokowej widać, że czas ten może być znacznie krótszy i właśnie dla takich potrzeb opracowuje się dynamicznie rekonfigurowalne (rekonfigurowalne tylko partiami) układy FPGA [265, 268].

Dla skrócenia czasu realizacji obliczeń przez procesory specjalizowane (tzw. sprzętowe) intensywnie rozwijane są obecnie metody szybkiej realizacji wybranych operacji logicznych i arytmetycznych. Problem ten jest analogiczny do rozważań wcześniej problemów architektur systemów, ale dotyczy skali mikro i wiąże się z takim rozczłonkowaniem zadań danej operacji arytmetycznej lub logicznej, aby czas jej wykonania był sumarycznie jak najmniejszy. Pomimo, iż w grę wchodzi niewielkie opóźnienia czasowe (rzędu od kilku do kilkudziesięciu ns), to operacje te wykonywane wiele milionów razy wnoszą znacząco duże opóźnienia. Zagadnienie staje się problemem poważnym także wówczas, jeżeli dana operacja ma być realizowana w trybie pracy synchronicznej i musi się zmieścić w ściśle określonym przedziale (kwancie) czasu. W rozpatrywanej w pracach [324, 335, 336] architekturze potokowej taki kwant czasowy wynosił 66 ns i zdarzało się, że dla niektórych operacji należało czas ten zwielokrotnić, aby zachować synchronizm współpracy procesorów. Dlatego opracowania w tym zakresie są bardzo ważne, chociaż dotyczą stosunkowo prostych struktur (np. bardzo szybki sumator wielobitowy) [5, 24, 58, 115, 204, 219, 232].

Opracowana przez autora architektura potokowa procesorów specjalizowanych i związany z nią standard magistrali potokowej mają bardzo dobre parametry czasowe [308, 309, 310, 312, 318, 324], szczególnie w zestawieniu z metodą software'ową, realizowaną przez procesory ogólnego przeznaczenia [240, 241] i dedykowanymi rozwiązaniami sprzętowymi [3, 6, 61, 62, 63, 76, 101, 218, 275]. Przedstawiona powyżej koncepcja architektury potokowej i procesorów specjalizowanych jest wkładem autora w aktualny nurt badań nad dedykowanymi dla użytkownika

strukturami obliczeniowymi CCM (ang. *Custom Computing Machines*) [16, 60, 103, 226, 233, 237, 265, 342].

Pełne wykorzystanie opracowanej architektury potokowej procesorów specjalizowanych związane jest z odpowiednim jej otoczeniem, umożliwiającym odpowiednio szybkie odebranie przetworzonych danych wizyjnych. W pracach [117, 186, 206, 227, 293, 302, 304, 306, 315, 316, 317, 359] omówiono dostępne standardy magistral uzasadniając wybór magistrali VME. Ponadto w [307, 310] przedstawiono opracowany, odpowiednio zorganizowany bank pamięci wyjściowej, zapewniającej efektywną współpracę architektury potokowej z magistralą w standardzie VME. Dalsze możliwości przyspieszenia komunikacji międzyprocesorowej poprzez transmisję dużych bloków danych do wielu procesorów i komunikację z kilkoma jednostkami jednocześnie zawarto w [317, 323]. Dzięki nowoczesnym pamięciom wieloportowym [191, 284] opracowano translator międzyliniowego sygnału wizyjnego do postaci kolejnoliniowej, zapewniający bezstratną w sensie czasowym współpracę kamery z architekturą potokową [305, 311]. Efektem współpracy powyższych modułów jest rozbudowany, niejednorodny system wizyjny, którym winno zarządzać oprogramowanie, spełniające wymagania pracy w trybie czasu rzeczywistego [108, 162, 186, 315, 316, 323]. Szczególnie ważną kwestią w takich systemach jest synchronizacja i komunikacja pomiędzy procesami i procesorami [238, 303, 320]. Wymogi te bardzo dobrze spełnia zastosowany przez autora w trakcie testowania architektury potokowej procesorów specjalizowanych system operacyjny OS-9 [65, 193, 303] wspomagany pakietem graficznym MGR [74].

6. IMPLEMENTACJE W UKŁADACH FPGA

Pełne przetestowanie opracowanych przez autora procesorów specjalizowanych, współpracujących z magistralą potokową, było możliwe dzięki szybkiemu rozwojowi w ostatnim okresie układów programowalnych FPGA [2, 9, 10, 12, 17, 281, 347], których stosowanie wspomagane jest bogatym zestawem narzędzi projektowych i testujących [28, 45, 135, 179, 180, 215, 285, 289, 301, 348, 349]. Autor korzystał z układów FPGA jako platformy technologicznej, która pozwoliła praktycznie zweryfikować opracowaną koncepcję. Dla jej przetestowania zaprojektowano i zbadano następujące procesory specjalizowane: logiczny, LUT, medianowy, konwolucji, dwukrotnej konwolucji (gradient Sobela w dwu kierunkach) oraz zliczania histogramu [318, 323, 324, 325, 326, 327, 328]. Implementowanie tych procesorów w strukturach programowalnych FPGA, których konfiguracja zapisywana jest w pamięci statycznej RAM, pozwala rekonfigurować procesory w trakcie normalnej pracy systemu (*on-line*), umożliwiając dynamiczną zmianę parametrów i algorytmów przetwarzania [30, 178, 286, 290, 294]. Właściwości takiej w zasadzie nie posiadały procesory specjalizowane realizowane w technologii VLSI. Możliwość zapisu konfiguracji układu programowalnego w pamięci RAM umożliwiła także opracowanie uniwersalnego procesora rekonfigurowalnego, który po odpowiednim zaprogramo-

waniu może realizować dowolną operację wstępnego przetwarzania danych wizyjnych [319, 321, 322, 324, 331].

W pracy [335] przeprowadzono szczegółową analizę opóźnień w procesorze zaimplementowanym w układzie FPGA. Wskazano na główne elementy składowe wnoszonego przez procesor opóźnienia. Wskazania te zbieżne są z pracami nad taką modyfikacją konstrukcji układów programowalnych FPGA, aby wnoszone opóźnienia były jak najmniejsze [27, 88], w szczególności dla sygnału zegarowego [364]. Analizowane w pracach opóźnienia wynikają zarówno ze sposobu zorganizowania zasobów wewnętrznych układów programowalnych, jak również ze sposobu ich optymalnego wykorzystania. W tym zakresie powstało wiele prac analizujących optymalny *partitioning* i *routing* koniecznej logiki do zasobów programowalnych [32, 49, 176, 255, 271, 282, 341, 344, 345]. Na uwagę zasługują prace dotyczące aspektu synchronizacji [207] i opóźnienia, analizowanego już na etapie *routing'u* [50, 163, 365]. W tym zakresie cenne są metody szybkie prototypowania układów FPGA [165]. Efektywne algorytmy automatycznego *routing'u* są ściśle związane z metodami optymalnej dekompozycji i syntezy danej funkcji logicznej [22, 33, 242]. Wspomaganie procesów optymalizacji czasowej może następować już na poziomie syntezy logicznej układu, zorientowanej na przyszłą implementację w układach FPGA [43, 56, 357]. Kolejnym zagadnieniem jest w miarę optymalne wykorzystanie wyprowadzeń układów programowalnych i taka ich alokacja, aby wnoszone opóźnienia były minimalne. Zagadnienie to jest szczególnie ważne przy łączeniu kilku kostek układów programowalnych FPGA [78, 116, 129, 142, 173, 226, 286, 354].

Nowe możliwości projektowania dedykowanych procesorów w strukturach reprogramowalnych FPGA spowodowały rozwój nowego myślenia o kompleksowym projektowaniu systemów. Wymienność realizowania algorytmów przez sprzęt i oprogramowanie daje możliwość elastycznego dzielenia zadań pomiędzy hardware i software oraz opracowywania procesorów z każdorazowo kształtowaną listą instrukcji [26, 54, 82, 97, 120, 123, 190, 222, 237, 263]. Technika ta znana pod nazwą *Hardware/Software Co-Design* ma duże znaczenie dla architektur CCM.

Szybkiemu rozwojowi dedykowanych architektur i procesorów specjalizowanych sprzyja bardzo dynamiczny rozwój układów programowalnych FPGA. Celem prac w tym zakresie jest zwiększenie pojemności tych układów oraz taka modyfikacja wewnętrznych struktur logicznych [79, 172, 283] i połączeniowych [86, 93, 158], aby można je było coraz bardziej efektywnie wykorzystywać. Największym problemem jest minimalizacja wnoszonych opóźnień w rozrastających się matrycach przełączających. Przydatne w tych pracach są analogiczne doświadczenia związane z układami VLSI [94, 95, 358, 363]. Dla opracowywania najbardziej dostosowanych do potrzeb użytkownika układów FPGA projektanci nowych struktur z renomowanych firm uczestniczą w realizacji najbardziej wyszukanych projektów, w tym właśnie dla potrzeb systemów wizyjnych [340]. Bardzo szybki rozwój układów programowalnych FPGA spowodował, że pojemności tych układów przekroczyły już milion bramek, a ich struktura wewnętrzna i parametry czasowe są coraz lepsze, szczególnie z punktu widzenia przedstawionych w prach rozważań [350].

7. PODSUMOWANIE

Przedstawione przez autora prace, a w szczególności [324, 335, 336], są istotnym uzupełnieniem do aktualnie stosowanych architektur wieloprocessorowych, ukazującym szerokie pole zastosowań architektury MISD, szczególnie do systemów przetwarzających duże ilości danych. Autor przeciwstawia się przy tym często wyrażanej tezie o małej użyteczności architektury MISD. Wystarczy przytoczyć kilka cytatów: „organizację MISD wprowadzono głównie dla kompletności klasyfikacji, trudno bowiem wskazać istniejący system komputerowy” [148], „klasyfikacja ta upraszcza stan faktyczny, gdyż nie istnieją praktyczne realizacje systemów typu MISD” [269], „MISD — ta klasa dodana została sztucznie do kompletu, ponieważ nie ma przykładów takiej architektury” [40], „MISD — modelowi temu nie odpowiada żaden ze znanych komputerów” [279]. W klasyfikacji Duncana [72] całkowicie wyeliminowano architekturę MISD. W celu potwierdzenia postawionej tezy autor zbudował oraz przebadął odpowiednie układy i udowodnił, że architektura MISD może znaleźć szerokie zastosowanie, uzasadnione znaczną akceleracją obliczeń, szczególnie w systemach wymagających dużych mocy obliczeniowych, których przykładem są systemy wizyjne.

Osiągnięte przez autora w trakcie prac badawczych wyniki, dotyczące czasu realizacji operacji wstępnego przetwarzania obrazów wizyjnych pokazują, że przyspieszenie wnoszone przez proponowaną architekturę wynosi kilka rzędów w stosunku do najnowszych opracowań, w tym także do opracowań specjalizowanych (66 ns zamiast 180 ms dla liczenia histogramu, 132 ns zamiast 250 ms dla odejmowania obrazów 102 μ s zamiast 51,9 ms dla operacji konwolucji) [335]. Pomimo, że dalszy postęp w zwiększeniu szybkości obliczeń przez nowo opracowywane procesory skróci dystans dzielący je od opracowanych przez autora procesorów i architektury, to należy zauważyć, że prezentowane rozważania mają charakter jakościowy, a nie konstrukcyjny i w miarę rozwoju technologii, mierzonej postępem w zakresie szybkości realizacji obliczeń przez procesory ogólnego przeznaczenia, szybkości opracowanych przez autora struktur będą także wzrastały, zachowując jakościowy dystans w stosunku do systemów ogólnego przeznaczenia.

Wartość postawionej tezy, o istotnych walorach stosowania architektury MISD w systemach wizyjnych czasu rzeczywistego, podnosi fakt opracowania szybkich specjalizowanych elementów obliczeniowych (procesorów) i wydajnej dedykowanej struktury ich połączeń (architektury), z możliwością dynamicznej rekonfiguracji (w strukturach FPGA), jako wkład do światowych prac badawczych na polu dedykowanych dla użytkownika struktur obliczeniowych CCM (ang. *Custom Computing Machines*), których ważną cechą jest rekonfigurowalność architektury systemów RC (ang. *Reconfigurable Computers*) i VC (ang. *Virtual Computers*).

Realizacja struktur CCM w układach FPGA pozwoli w przyszłości na wyposażenie komputerów w rekonfigurowalne koprocessory sprzętowe. Dorobek autora na tym polu stanowi opracowanie i zbadanie architektury potokowej specjalizowanych procesorów sprzętowych, która może być zaimplementowana w dowolnym środowisku. Zbadana architektura potokowa może być implementowana w uniwersalnej

TOM
arch
dan
złoż
1. M
E
P
2. A
3. A
4. M
II
P
5. A
II
7,
6. A
7. W
D
Pr
8. N
9. A
10. A
11. A
Pr
N
12. A
13. H
&
Ar
14. J.
Th
15. J.
78
16. P.
So
Ca
17. AT
At
18. G.
19. P.
Hi
Im
20. P.J.
Ar
Ca
21. M.
Ma

architekturze rekonfigurowalnej jako wysokowydajne narzędzie do przetwarzania danych wizyjnych i innych podobnie zorganizowanych, w szczególności także jako złożony koprocesor wizyjny.

BIBLIOGRAFIA

1. M. Abdelguerfi, S. Khalaf, A.K. Sood: *Bit-Serial Parallel Processing Unit for the Histogramming Operation*. IEEE Transactions on Circuits and Systems, Vol. 37, No. 7, July 1990, pp. 948–954
2. ACTEL: *ACT Family Field Programmable Gate Array Databook*, Sunnyvale CA, Actel Corp. 1992
3. *Advanced Imaging*, New York, PTN Publishing Corp. 1994–1995
4. M. Agarwala, P.T. Balsara: *An Architecture for a DSP Field-Programmable Gate Array*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 3, No. 1, March 1995, pp. 136–141
5. A. Aggoun, M.K. Ibrahim, A. Ashur: *Bit Level Pipelined Digit-Serial Array Processors*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 45, No. 7, July 1998, pp. 857–868
6. ALACRON: *AL860/FT200 Computing Subsystems*. Nashua, Alacron Inc. 1995
7. W.E. Alexander, D.S. Reeves, C.S. Gloster: *Parallel Image Processing with the Block Data Parallel Architecture*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 947–968
8. N. Alexandridis: *Design of Microprocessor-Based Systems*. New Jersey, Prentice Hall, 1993
9. ALTERA: *Data Book*. San Jose CA, Altera Corp. 1993
10. ALTERA: *Flex 8000 Handbook*. San Jose CA, Altera Corp. 1994
11. A. Antola, A. Avai, L. Breveglieri: *Modular Design Methodologies for Image Processing Architectures*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 1, No. 4, December 1993, pp. 408–414
12. AMD: *MACH 1, 2, 3 and 4 Family Data Book*. Sunnyvale CA, AMD Inc. 1995
13. H.R. Arabnia, J.W. Smith: *A Reconfigurable Parallel Imaging System*. Proc. of the IASTED & IEEE International Conference: Signal and Image Processing — SIP-95 Las Vegas NV 1995, Anaheim—Calgary—Zurich, IASTED Acta Press 1995, pp. 21–23
14. J. Aranda, J. Climent, A. Grau: *A FPGA Implementation of a Video Rate Multi-target Tracking System*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 70–73
15. J. Astola, P. Haavisto, Y. Neuvo: *Vector Median Filter*. Proceedings of the IEEE, Vol. 78, No. 4, April 1990, pp. 678–689
16. P. Athanas, R. Hudson: *Using Rapid Prototyping to Teach the Design of Complete Computing Solutions*. Proceedings of the IEEE Symposium on FGPAs for Custom Computing Machines, California 1996, pp. 90–97
17. ATMEL: *Configurable Logic Design & Application Book — PLD, FPGA, Gate Array*, San Jose CA, Atmel Corp. 1995
18. G.W. Awoock, R. Thomas: *Applied Image Processing*. New York, Mc Graw Hill 1995
19. P. Baglietto, M. Maresca, M. Migliardi, N. Zingrian: *Image Processing on High-Performance RISC Systems*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 917–930
20. P.J. Bekkes, J.J. Plessis, B.L. Hutchings: *Mixing Fixed and Reconfigurable Logic Array Processing*. Proceedings of the IEEE Symposium on FGPAs for Custom Computing Machines, California 1996, pp. 118–127
21. M.A. Bayoumi: *VLSI Design Methodologies for Digital Signal Processing Architecture*. Norwell MA, Kluwer, 1994

22. A. Bender: *Optimal System-Level Synthesis of Digital Systems for Real-time Applications*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 59–64
23. A. Benedetti, A. Prati, N. Scarabottolo: *Image Convolution on FPGAs: the Implementation of Multi-FPGA FIFO Structure*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 123–130
24. J. Biernat: *Arytmetyka komputerów*. Warszawa, PWN, 1996
25. E.I. Boemo, S. Lopez-Buedo, J.M. Meneses: *Some Experiments About Wave Pipelining on FPGA's*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 2, June 1998, pp. 232–237
26. I. Bolsen, H.J. Man, B. Lin, K. Rompaey, S. Vercauteren, D. Verkest: *Hardware/Software Co-Design of Digital Telecommunication Systems*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 391–418
27. G. Borriello, C. Ebeling, S.A. Hauck, S. Burns: *The Triptych FPGA Architecture*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 3, No. 4, December 1995, pp. 491–500
28. G. Bostock: *Programmable Logic Handbook*. Manchester, Newnes, 1993
29. N. Bourbakis, N. Seffensen, B. Saha: *Design of an Array Processor for Parallel Skeletonization of Images*. IEEE Transactions on Circuits and Systems, Vol. 37, No. 1, January 1990, pp. 284–298
30. V.M. Bove, J.A. Watlington: *Cheops: A Reconfigurable Data-Flow System for Video Processing*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 2, April 1995, pp. 140–149
31. A. Broggi: *Parallel and Local Feature Extraction: A Real-Time Approach to Road Boundary Detection*. IEEE Transactions on Image Processing, Vol. 4, No. 2, February 1995, pp. 217–223
32. S.D. Brown, J. Rose, Z.G. Vranic: *A Stochastic Model to Predict the Routability of Field-Programmable Gate Arrays*. IEEE Transactions on Computer — Aided Design of Integrated Circuits and Systems, Vol. 12, No. 12, December 1993, pp. 1827–1838
33. M. Burns, M. Perkowski, L. Józwiak: *An Efficient Approach to Decomposition of Multi-Output Boolean Functions with Large Sets of Bound Variables*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 16–23
34. D. Caban: *Hardware Implementation of a Real Time Positional Filter*. Proc. of the 5th School: Computer Vision and Graphics, Wrocław, 1994, pp. 195–200
35. V. Cantoni, M. Ferretti, L. Lombardi: *A Comparison of Homogeneous Hierarchical Interconnection Structures*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 416–428
36. K.R. Castleman: *Digital Image Processing*. New Jersey, Prentice Hall, 1996
37. T.L. Casavant, M. Singhal: *Distributed Computing Systems*, New York, IEEE 1994
38. C. Chakrabarti: *Efficient Stack Filter Implementations of Rank order Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL 1993, pp. 958–961
39. P. Chalermwat, N. Alexandridis, P. Piamsa-Nga, M. O'Connell: *Parallel Image Processing in Heterogeneous Computing Network Systems*. Proceedings of the IEEE International Conference on Image Processing, Lausanne, 1996, pp. 161–164
40. B.S. Chalk: *Organizacja i architektura komputerów*. Warszawa, WNT, 1998
41. S.C. Chan, H.O. Ngai, K.L. Ho: *A Programmable Image Processing System using FPGA*. Proc. of the IEEE International Symposium on Circuits and Systems, London 1994, pp. 2.152–2.128
42. C.M. Chang, S.L. Lu: *Design of a Static MIMD Data flow Processor Using Micropipelines*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 3, No. 3, September 1995, pp. 370–378
43. S.C. Chang, M. Marek-Sadowska, T.T. Hwang: *Technology Mapping for TLU FPGA's Based on Decomposition of Binary Decision Diagrams*. IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 10, October 1997, pp. 1226–1236
44. L.W. Chang, J.H. Lin: *A Bit Level Systolic Array for Median Filter*. IEEE Transactions on Signal Processing, Vol. 40, No. 8, August 1992, pp. 2079–2083

45. Y.W. Chang, D.F. Wong, C.K. Wong: *Design and Analysis of FPGA/FPIC Switch Modules*. Proc. of the Int. Conference on Computer Design: VLSI in Computers & Processors, IEEE Computer Society Press 1995, pp. 394—401
46. R. Chellappa: *Digital Image Processing*. New York, IEEE, 1992
47. C.T. Chen, L.G. Chen, J.H. Hsia: *VLSI Implementation of a Selective Median Filter*, IEEE Transactions on Consumer Electronics, Vol. 42, No. 1, February 1996, pp. 33—41
48. C.C. Chen, C.W. Jen: *A Programmable Concurrent Video Signal Processor*. Proceedings of the IEEE International Conference on Image Processing, Lausanne 1996, pp. 1039—1042
49. C.D. Chen, Y.S. Lee, A.C.H. Wu, Y.L. Lin: *Tracer-FPGA: A Router for RAM-Based FPGA's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 14, No. 3, March 1995, pp. 371—374
50. C.S. Chen, Y.W. Tsay, T.T. Hwang, A.C.H. Wu, Y.L. Lin: *Combining Technology Mapping and Placement for Delay-Minimization in FPGA Designs*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 14, No. 9, September 1995, pp. 1076—1084
51. Y.Y. Chen, S.J. Upadhyaya, C.H. Cheng: *A Comprehensive Reconfiguration Scheme for Fault-Tolerant VLSI/WSI Array Processors*. IEEE Transactions on Computers, Vol. 46, No. 12, December 1997, pp. 1363—1371
52. F. Cheng, A.N. Venetsanopoulos: *An Adaptive Morphological Filter for Image Processing*. IEEE Transactions on Image Processing, Vol. 1, No. 4, October 1992, pp. 533—539
53. A.N. Choudary, J.H. Patel: *Parallel Architectures and Parallel Algorithms for Integrated Vision Systems*. Norwell MA, Kluwer 1990
54. D.A. Clark, B.L. Hutchings: *Supporting FPGA Microprocessors Through Retargetable Software Tools*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 195—205
55. M.J. Colaitis, J.L. Jumpertz, B. Guerin, B. Cheron, F. Battini, B. Lescure, E. Gautier, J.P. Geffroy: *The Implementation of P³ I, a Parallel Architecture for Video Real-Time Processing: A Case Study*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 1019—1037
56. J. Cong, Y. Ding: *On Area/Depth Trade-Off in LUT-Based FPGA Technology Mapping*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 2, No. 2, June 1994, pp. 137—148
57. G. Conte, D. Corso: *Multi-Microprocessor System for Real-Time Applications*. Norwell MA, Kluwer, 1985
58. P. Corbett, R. Hartley: *Designing Systolic Arrays Using Digit-Serial Arithmetic*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 39, No. 1, January 1992, pp. 62—65
59. Y. Cristol: *Solid State Video Cameras*. Pergamon Press, 1986
60. W.B. Culbertson, R. Amerson, R.J. Carter, P. Kuekes, G. Snider: *Exploring Architectures for Volume Visualisation on the Teramac Custom Computer*. Proceedings of the IEEE Symposium on FPGA for Custom Computing Machines, California 1996, pp. 80—89
61. E.L. Dagle: *Real Time Vision Aided Traffic Monitoring*. Applied Mathematics and Computer Science, vol. 3, no. 1, Zielona Góra 1993, pp. 125—134
62. E.L. Dagle: *Image Processing Hardware and Algorithms*. Proc. of the 5-th School: Computer Vision and Graphics, Wrocław, 1994, pp. 9-34
63. DATA TRANSLATION: *Image Processing Handbook*, Marlboro MA, Data Translation Inc. 1989
64. A.L. Decegama: *The Technology of Parallel Processing — Parallel Processing Architectures and VLSI Hardware*. New Jersey, Prentice Hall 1989
65. P.C. Dibble: *OS-9 INSIGHTS an Advanced Programmers Guide*, Des Moines — Iowa, Microware System Corporation, 1992
66. F. Distanto, M. Sami, R. Stefanelli, G. Storti-Gajani: *Mapping Neural Nets onto a Massively Parallel Architecture: A defect-Tolerance Solution*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 444—460

67. E.R. Dougherty, J. Astola: *Mathematical Nonlinear Image Processing*. Norwell MA, Kluwer, 1993
68. E.R. Dougherty, P.A. Laplante: *Introduction to Real-Time Imaging*. New York, IEEE Press, 1995
69. E.R. Dougherty, D. Sinha: *Computational Gray-scale Mathematical Morphology on Lattice*. Real-Time Imaging 1995, vol. 1, pp. 69–85
70. M.J.B. Duff: *Parallel Processors for Digital Image Processing, Advances in Digital Image Processing*. New York, Plenum Press 1979
71. M.J.B. Duff: *Computing Structures for Image Processing*, London, Academi Press Ltd. 1983
72. R. Duncan: *A Survey of Parallel Computer Architectures*. Computer, Vol. 23, No. 2, February 1990, pp. 5–16
73. S. Dutta, W. Wolf, A. Wolfe: *VLSI Issues in Memory-System Design for Video Signal Processors*, Proc. of the Int. Conference on Computer Design: VLSI in Computers & Processors, IEEE Computer Society Press 1995, pp. 498–505
74. EKF: *System — PAK 1/MGR ver. 1.3.4*. Reccoware Systems, Wolfgang Ocker, 1993
75. I. Ekmeçic, I. Tartalja, V. Milutinovic: *A Survey of Heterogeneous Computing: Concepts and Systems*. Proceedings of the IEEE, Vol. 84, No. 4, August 1996, pp. 1127–1144
76. ELTEC: *VMEbus Image Processing*. California, Eltec Corporation, 1991
77. J.B. Evans: *Efficient FIR Filter Architectures Suitable for FPGA Implementation*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 41, No. 7, July 1994, pp. 490–493
78. W.J. Fang, A.C.H. Wu: *A Hierarchical Functional Structuring and Partitioning Approach for Multiple-FPGA Implementation*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 10, October 1997, pp. 1188–1195
79. A.H. Farrahi, M. Sarrafzadeh: *Complexity of the Lookup-Table Minimization Problem for FPGA Technology Mapping*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 13, No. 11, November 1994, pp. 1319–1332
80. M.N. Fesharaki, G.R. Hellestrand: *A Real-Time Edge Detection ASIC Design*, Proc. of the IEEE International Symposium on Circuits and Systems, London, 1994, pp. 3.81–3.84
81. M.J. Flynn: *Very High-Speed Computing-Systems*. Proc. IEEE, 1966, Vol. 54, No. 12
82. M.J. Flynn: *Introcution to: Influence of Programming Techniques on the Design of Computers*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 467–469
83. FORCE: *SYS 68K/CPU-30 User's Manual ver. 3.0*. Campbell — CA, Force Computer Inc. 1991
84. K.S. Fu, R.C. Gonzalez, C.S.G. Lee: *Robotics: Control, Sensing, Vision and Inteligence*. New York, Mc Graw Hill, 1987
85. O. Fucik, J. Honec, P. Valenta, P. Zemcik: *Cost Efficient Image Digitising and Processing System Using FPGAs*. Proc. of the 5-th School: Computer Vision and Graphics, Wroclaw, 1994, pp. 225–232
86. K. Fujiyoshi, Y. Kajitani, H. Niitsu: *Design of Minimum and Uniform Biparties for Optimum Connection Block of FPGA*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 11, November 1997, pp. 1377–1383
87. K. Gaedke, H. Jeschke, P. Pirsch: *A VLSI Based MIMD Architecture of a Multiprocessor System for Real-Time Video Processing Applications*. A Special Issue of: Journal of VLSI Signal Processing, vol. 5 nos. 2–3 — 1993, pp. 159–169
88. D.D. Gajski, S. Narayan, L. Ramachandran, F. Vahid, P. Fung: *System Design Methodologies: Aiming at the 100 h Design Cycle*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 4, No. 1, March 1996, pp. 70–81
89. J.M. Gauch: *Multiresolution Image Shape Description*. New York, Springer, 1992
90. J.C. Gealow, F.P. Hermann, L.T. Hsu, C.G. Sordini: *System Design for Pixel-Parallel Image Processing*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 4, No. 1, March 1996, pp. 32–41
91. GEC PLESSEY: *Digital Signal Processing — IC Handbook*. Scotts Valley CA, GEC Plessey Semiconductors, 1990

92. W. Gehrke, K. Gaedke: *Associative Controlling of Monolithic Parallel Processor Architectures*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 5, October 1995, pp. 453–464
93. J. Ghosh, A. Varma, N. Krishnamurthy: *Distributed Control Schemes for Fast Arbitration in Large Crossbar Networks*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 2, No. 1, March 1994, pp. 54–67
94. J. Ghosh, A. Varma: *Reduction of Simultaneous-Switching Noise in Large Crossbar Networks*. IEEE Transactions on Circuits and Systems, Vol. 38, No. 1, January 1991, pp. 86–99
95. A. Gloria, P. Faraboschi, M. Olivieri: *Design and Characterization of a Standard Cell Set for Delay Insensitive VLSI Design*. IEEE Transactions on Circuits and Systems — II. Analog and Digital Processing, Vol. 41, No. 6, June 1994, pp. 410–415
96. J. Goodenough, R.J. Meacham, J.D. Morris, N.L. Seed, P.A. Ivey: *A Single Chip Video Signal Processing Architecture for Image Processing, Coding, and Computer Vision*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 5, October 1995, pp. 436–445
97. G. Goossens, J. Praet, D. Lanner, W. Geurts, A. Kifli, C. Liem, P.G. Paulin: *Embed Software in Real-Time Signal Processing Systems: Design Technologies*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 436–454
98. R.C. Gonzalez, P. Wintz: *Digital Image Processing*. Reading MA, Addison-Wesley, 1987
99. M. Gorgoń: *Wykorzystanie systemu specjalizowanych procesorów hardware'owych we wstępnym przetwarzaniu obrazu*. Elektrotechnika, t. 11, z. 4, Krakow, 1992, ss. 411–432
100. M. Gorgoń: *VLSI Based Pipeline Architecture for Real-Time Image Data Preprocessing*. Proceedings of the 5-th School: Computer Vision and Graphics, Wrocław 1994, pp. 233–238
101. M. Gorgoń: *Ocena specjalizowanych procesorów pod kątem ich przydatności do wstępnego przetwarzania obrazu*. Praca doktorska, Kraków, AGH, 1995
102. M. Gorgoń: *FPGA Implementation of Dedicated Image Processor*. Proc. of the 7-th School: VLSI and ASIC Design, Wrocław 1995, pp. 201–208
103. P. Graham, B. Nelson: *Genetic Algorithms in Software and in Hardware — A Performance Analysis of Workstations and Custom Computing Machine Implementations*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 216–225
104. J.H. Graham: *Computer Architectures for Robotics and Automation*. New York, Gordon and Breach Science Publishers, 1987
105. H.J.M. Groil: *Traffic Assignment Problems Solved by Special Purpose Hardware with Emphasis on Real-Time Applications*. Delft University, 1992
106. J.I. Guo, C.M. Liu, C.W. Jen: *A General Approach to Design VLSI Arrays for the Multi-Dimensional Discrete Hartley Transform*. Proceedings of the IEEE International Symposium on Circuits and Systems, London 1994, pp. 235–238
107. B. Gunther, G. Milne, L. Narasimhan: *Assessing Document Relevance with Run-Time Reconfigurable Machines*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 10–17
108. W.A. Halang, A.D. Stoyenko: *Constructing Predictable Real-Time Systems*. Norwell MA, Kluwer 1991
109. M. Hamdi: *Modified Mesh-Connected Computers for Image Processing Applications*. IEEE Trans. 1995, pp. 139–145
110. D.W. Hammerstrom, D.P. Lulich: *Image Processing Using One-Dimensional Processor Arrays*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 1005–1018
111. A. Hansford: *The AD9502 Video Signal Digitizer and Its Applications*. Application Note. Norwood MA, Analog Devices, 1990
112. R.M. Haralick, L.G. Shapiro: *Computer and Robot Vision*. California, Addison-Wesley Publishing 1992
113. HARRIS: *Digital Signal Processing*. Melbourne, Harris Semiconductors Inc. 1993

114. D.K. Harris-Dowset, S. Summerfield: *Low Latency Architectures for Wave Digital Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, Cichago IL 1993, pp. 1881–1884
115. R. Hartley, P. Corbett: *Digit-Serial Processing Techniques*. IEEE Transactions on Circuits and Systems, Vol. 37, No. 6, June 1990, pp. 707–719
116. S. Hauck, G. Borriello: *Pin Assignment for Multi-FPGA Systems*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 9, September 1997, pp. 956–964
117. N. Hauser: *Technology boosts VMEbus Data Transfer to 560 MBytes/sec*. Proc. of the VITA Conference: Open Bus Systems'92, Zurich, 1992, pp. 61–66
118. Z. He, M.L. Liou: *A New Array Architecture for Motion Estimation*. Proceedings of the IEEE Wrokshop Visual Signal Processing and Communications, New Jersey 1994, pp. 148–151
119. H.J.A.M. Heijmans: *Composing Morphological Filters*. IEEE Transactions on Image Processing, Vol. 6, No. 5, May 1997, pp. 713–723
120. M. Heinrich, D. Ofelt, M.A. Horowitz, Hennessy: *Hardware/Software Co-Design of the Standard FLASH Multiprocessor*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 455–466
121. P.V. Henstock, D.M. Chelberg: *Automatic Gradient Treshold Determination for Edge Detection*. IEEE Transactions on Image Processing, Vol. 5, No. 5, May 1996, pp. 784–787
122. B. Herzen: *Signal Processing at 250 MHz Using High-Performance FPGA's*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 2, June 1998, pp. 238–246
123. G.M. Hopper, J.W. Mauchly: *Influence of Programming Techniques on the Design of Computers*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 470–474
124. J.H. Hsiao, L.G. Chen, T.D. Chiueh, C.T. Chen: *Novel Systolic Array Deisgn for the Discrete Hartley Transform with High Throughput Rate*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL 1993, pp. 1567–1570
125. K. Hwang, F. Briggs: *Computer Architecture and Parallel Processing*. New York, Mc Graw-Hill, 1984
126. K. Hwang: *Advanced Computer Architecture*. New York, Mc Graw-Hill 1993
127. J. Hyncek: *Low-Noise and High-Speed Charge Detection in High-Resolution CCD Image Sensor*. IEEE Transactions on Electron Devices, Vol. 44, No. 10, October 1997, pp. 1679–1688
128. IDT: *Specialised Memories, FIFOs & Modules*. Santa Clara CA, Integrated Device Technology Inc. 1995
129. T. Isshiki, W.W.M. Dai: *Bit-Serial Pipeline Synthesis for Multi-FPGA Systems with C++ Design Capture*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 38–47
130. K. Itakura, T. Nobusada, Y. Toyoda, Y. Saitou inni: *A 2/3-in 2.0 M.-Pixel CCD Imager with an Advanced M-FIT Architecture Capable of Progressive Scan*. IEEE Transactions on Electron Devices, Vol. 44, No. 10, October 1997, pp. 1625–1632
131. Y. Iwata, M. Kawamata, T. Higuchi: *Design of Fine Grain VLSI Array Processor for Real-Time 2-D Digital Filtering*. Proc. of the IEEE International Symposium on Circuits and Systems, Chicago 1993, pp. 1559–1562
132. A.K. Jain: *Fundamentals of Digital Image Processing*. New Jersey, Prentice Hall, 1989
133. M.M. Jamali, S. Ravindranath, S.C. Kwatra, A.G. Eldin: *ASIC Design of a Generalized Covariance Matrix Processor for DOA Algorithms*. Proceedings of the IEEE International Symposium on Circuits and Systems, London 1994, pp. 75–78
134. S.D. Jean, C.M. Liu, C.C. Chang, Z. Chen: *A New Algorithm and its VLSI Architecture Design for Connected Component Labelling*. Proc. of the IEEE International Symposium on Circuits and Systems, London 1994, pp. 2.565–2.568
135. J.H. Jenkins: *Designing with FPGAs and CPLDs*, New Jersey, Prentice Hall, 1994
136. E. Jezierski, D. Zarychta, A. Jezierski: *Tracking of Robot Arm in 3D Space using Stereo Vision*. Proc. of 6th Int. Symposium on Measurements and Control in Robotics, Brussels 1996, pp. 439–443

137. X.C. Jin, S.H. Ong, Jayasooriah: *A Domain Operator for Binary Morphological Processing*. IEEE Transactions on Image Processing, Vol. 4, No. 7, July 1995, pp. 1042–1046
138. P.P. Jonker: *Morphological Image Processing: Architecture and VLSI Design*. Norwell MA, Kluwer 1992
139. R.Y. Kain: *Advanced Computer Architecture*. New Jersey, Prentice Hall 1996
140. B.K. Kar, K.M. Yusuf, D.K. Pradhan: *Bit-Serial Generalized Median Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, London 1994, pp. 3.85–3.88
141. M. Karaczyn, K. Pucher: *The MAVEC FPBA-Based Image Acquisition and Processing Board*. Proc. of the Conference: Programmable Devices and Systems PDS'95, Gliwice, 1995, pp. 59–64
142. C. Kim, H. Shin: *A Performance-Driven Logic Emulation System: FPGA Network Design and Performance-Driven Partitioning*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 15, No. 5, May 1996, pp. 560–568
143. W.E. King, T.H. Dreyer, R.W. Conners, P. Araman: *Using MORRPH in an Industrial Machine Vision System*. Proceedings of the IEEE Symposium on FPGAs for Custom Computer Machines, California 1996, pp. 18–26
144. B. Kisacanin, D. Schonfeld: *A Fast Thresholded Linear Convolution Representation of Morphological Operations*. IEEE Transactions on Image Processing, Vol. 3, No. 4, July 1994, pp. 455–457
145. J. Kittler, M.J.B. Duff: *Image Processing System Architecture*. New York, John Wiley & Sons 1985
146. B. Klaus, P. Horn: *Robot Vision*. New York, McGraw-Hill, 1986
147. S.J. Ko, Y.H. Lee: *Center Weighted Median Filters and Their Applications to Image Enhancement*. IEEE Transactions on Circuits and Systems, Vol. 38, No. 9, September 1991, pp. 984–993
148. S. Kozielski, Z. Szczerbiński: *Komputery równoległe*. Warszawa, WNT 1994
149. I.C. Kraljic, G.M. Quenot, B. Zavidovique: *From Real-Time Emulation to ASIC Integration for Image Processing Applications*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 4, No. 3, September 1996, pp. 391–404
150. J. Kreyszig: *Techniques for the Implementation of Arithmetic Functions in ASICs*. Proc. of the 7-th School: VLSI and ASIC Design, Wrocław, 1995, pp. 99–113
151. A. Krikelis, R.M. Lea: *A Modular Massively Parallel Computing Approach to Image-Related Processing*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 988–1004
152. R.K. Krishnamurthy, R. Sridhar: *A CMOS Wave-pipelined Image Processor for Real-Time Morphology*. Proceedings of the IEEE International Conference on Computer Design: VLSI in Computer & Processors, Austin TX, 1995, pp. 638–640
153. B. Kruse: *System Architecture for Image Analysis*. Structured Computer Vision, Boston, Acad. Press, 1980
154. A. Kundu, J. Zhou: *Combination Median Filter*. IEEE Transactions on Image Processing, Vol. 1, No. 3, July 1992, pp. 422–429
155. H.K. Kwan, T.S. Okullo-Oballa: *2-D Systolic Arrays for Realization of 2-D Convolution*. IEEE Transactions on Circuits and Systems, Vol. 37, No. 2, February 1990, pp. 267–273
156. A.Y. Kwentus, M.J. Werter, A.N. Willson: *A Programmable Digital Filter IC Employing Multiple Processors on a Single Chip*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 2, No. 2, June 1992, pp. 231–244
157. S.H. Kwok, A.G. Constantinides: *A Fast Recursive Shortest Spanning Tree for Image Segmentation and Edge Detection*. IEEE Transactions on Image Processing, Vol. 6, No. 2, February 1997, pp. 328–332
158. Y.T. Lai, P.T. Wang: *Hierarchical Interconnection Structures for Field Programmable Gate Arrays*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 5, No. 2, June 1997, pp. 186–196
159. H.W. Lawson: *Parallel Processing in Industrial Real-Time Applications*. New Jersey, Prentice Hall, 1992

160. R.M. Lea, J.P. J alowiecki: *Associative Massively Parallel Computers*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 469–479
161. C. Lee, S.C. Juan, W.W. Yang: *A Parallel Bit-Level Maximum/Minimum Selector for Digital and Video Signal Processing*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 41, No. 10, October 1994, pp. 693–695
162. Y.H. Lee, C.M. Krishna: *Real-Time Systems*. New York, IEEE 1993
163. Y.S. Lee, A.C.H. Wu: *A Performance and routability-Driven Router for FPGA's. Considering Path Delays*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 2, February 1997, pp. 179–185
164. I. Lemberski: *Modified Approach to Automata State Encoding for LUT FPGA Implementation*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 196–199
165. D.M. Lewis, D.R. Galloway, M. Ierssel, J. Rose, P. Chow: *The Transmog-rifier-2: A 1 Million Gate Rapid-Prototyping System*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 2, June 1998, pp. 188–198
166. H. Li, Q.F. Stout: *Reconfigurable SIMD Massively Parallel Computers*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 429–443
167. J.S.J. Li, A. Ramsingh: *The Relationship of the Multi-Shell to Multistage and Standard Median Filters*. IEEE Transactions on Image Processing, Vol. 4, No. 8, August 1995, pp. 1165–1169
168. R.L. Libbey: *Signal & Image Processing Sourcebook*. New York, Van Nostrand Reinhold, 1994
169. B. Liebowitz, J. Carson: *Multiple Processor System for Real Time Applications*. New Jersey, Prentice Hall, 1985
170. J.S. Lim: *Two-dimensional Signal and Image Processing*. New Jersey, Prentice Hall, 1990
171. Y.S. Lin, J.I. Guo, C.B. Shung, C.W. Jen: *A Multi-Phase Shared Bus Structure for the Fast Fourier Transform*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL 1993, pp. 1575–1578
172. C.C. Lin, M. Marek-Sadowska: *On Designing Universal Logic Blocks and Their Application to FPGA Design*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 5, May 1997, pp. 519–527
173. H. Liu, D.F. Wong: *Network-Flow-Based Multiway Partitioning with Area and Pin Constraints*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 17, No. 1, January 1998, pp. 50–59
174. K.C. Lo: *Parallel Random Sampling with Multiprocessor System*. Proc. of the IEEE International Symposium, Chicago 1993, pp. 1979–1982
175. L. Louca, T.A. Cook, W.H. Johnson: *Implementation of IEEE Single Precision Floating Point Addition and Multiplication on FPGAs*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 107–117
176. A. Lu, E. Dagless, J. Saul: *DART: Delay and Routability Driven Technology Mapping for LUT Based FPGAs*. Proc. of the Int. Conference on Computer Design: VLSI in Computers & Processors, IEEE Computer Society Press, 1995, pp. 409–414
177. W. Luk, N. Shirazi, P.Y.K. Cheung: *Modelling and Optimizing Run-Time Reconfigurable Systems*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 168–177
178. P. Lysaght, J. Stockwood: *A Simulation Tool for Dynamically Reconfigurable Field Programmable Gate Arrays*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 4, No. 3, September 1996, pp. 381–390
179. T. Łuba, M.A. Markowski, B. Zbierzchowski: *Komputerowe projektowanie układów cyfrowych w strukturach PLD*. Warszawa, WKiŁ, 1993
180. T. Łuba, M.A. Markowski, B. Zbierzchowski: *Kompilatory układów logicznych*. Warszawa, OWPW 1995
181. M. Majorek, L. Wojnar: *Komputerowa analiza obrazu*. Kraków, FotoBit-Design, 1994

182. F. Mallet, F. Boeri, J.F. Duboc: *Hardware Architecture Modelling using an Object-oriented Method*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 147—153
183. P. Maragos, R.W. Schafer: *Morphological Systems for Multidimensional Signal Processing*. Proceedings of the IEEE Vol. 78, No. 4, April 1990, pp. 690—710
184. P. Maragos: *Differential Morphology and Image Processing*. IEEE Transactions on Image Processing, Vol. 5, No. 6, June 1996, pp. 922—937
185. M. Marimoto, K. Orihara, N. Mutoh, A. Toyodaiinni: *A 1-inch 2-M. Pixel HDTV CCD Image Sensor with Tungsten Photo-Shield and H-CCD Shunt Wiring*. IEEE Transactions on Electron Devices, Vol. 42, No. 1, January 1995, pp. 50—57
186. B. Marzec: *Wprowadzenie do standardu VMEbus*. Warszawa, WNT, 1994
187. A. Materka: *Elementy cyfrowego przetwarzania i analizy obrazów*. Warszawa—Łódź, PWN, 1991
188. A. Materka, W. Górnisiewicz, M. Kukula: *Mikrokomputerowe systemy przetwarzania obrazów*. Informatyka nr 10, Wrszawas, 1990, ss. 9—17
189. J. Meerbergen, P. Lippens, B. Mc Sweeney, W. Verhaegh, A. Van der Werf, A. Zanten: *Architectural Strategies for High-Throughput Applications*. A Special Issue of: Journal of VLSI Signal Processing, vol. 5, nos. 2—3 — 1993, pp. 201—220
190. G. Micheli, R.K. Gupta: *Hardware/Software Co-Design*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 349—365
191. MICRON: *DRAM Data Book*. Boise — Idaho, Micron Technology Inc., 1992
192. MICROSYS: *MicroSys VME. Made for Proffesionals*. Munchen, Microsys Inc. 1991
193. MICROWARE: *Proffesional OS-9 ver. 2.3*, Des Moines — Iowa. Microware System Corporation, 1989
194. Z. Mikrut, R. Tadeusiewicz: *Automatyczna wizualna identyfikacja obiektów na zrobotyzowanych stanowiskach pracy*. Elektrotechnika, t. 9, z. 3—4, Kraków, 1990, pp. 485—499
195. D. Milford: *Architectures for Computer Vision*. Applied Mathematics and Computer Science, vol. 3, no. 1, Zielona Góra 1993, pp. 135—140
196. L.L. Miller, A.R. Hurson, S.H. Pakzad: *Parallel ARchitecture for Data/Knowledge Based Systems*. New York, IEEE 1995
197. R. Miller, V.K. Prasanna-Kumar, D.I. Reisis, Q.F. Stout: *Image Computations on Reconfigurable VLSI Arrays*. IEEE Trans. 1988, pp. 925—930
198. E. Mirsky, A. De Hon: *MATRIX: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 157—166
199. S. Mo, J. Mathews: *Adaptive, Quadratic Preprocessing of Document Images for Binarization*. IEEE Transactions on Image Processing, Vol. 7, No. 7, July 1998, pp. 992—999
200. J. Monteiro, S. Devadas, A. Ghosh: *Sequential Logic Optimization for Low Power Using Input-Disabling Precomputation Architectures*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 17, No. 3, March 1998, pp. 279—284
201. V.J. Mooney, D. Ruspini, O. Khatib, G. Micheli: *Hardware-Software Run-Time Systems and Robotics: A Case Study*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 162—167
202. A. Morales, R. Acharya, S.J. Ko: *Morphological Pyramids with Alternating Sequential Filters*. IEEE Transactions on Image Processing, Vol. 4, No. 7, July 1995, pp. 965—977
203. S. Muller: *A New Programmable VLSI Architecture for Histogram and Statistics Computation in Different Windows*. Proceedings of the IEEE International Conference on Image Processing, Washington DC, 1995, pp. 73—76
204. C. Nagendra, M.J. Irwin, R.M. Owens: *Area-Time-Power Tradeoffs in Parallel Adders*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 43, No. 10, October 1996, pp. 689—702
205. H. Nakahira, M. Maruyama, H. Ueda, H. Yamada: *An Image Processing System using Image Signal Multiprocessors (ISMPs)*. A Special Issue of: Journal of VLSI Signal Processing, vol. 5, nos. 2—3 — 1993, pp. 133—140

206. J.E. Namara: *Technical Aspects of Data Communication*. Digital Press, 1988
207. L.B. Nguen, M. Perkowski, L. Jóźwiak: *Design of Self-Synchronized Component FSMs for Self-Timed Systems*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 253—260
208. S. Nichani, N. Ranganathan: *SAP: Design of a Systolic Array Processor for Computations in Vision*. Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers & Processors, Massachusetts MA, 1990, pp. 315—318
209. S.S. Nikolaidis, D.E. Metafas, C.E. Gouits: *CORDIC Based Pipeline Architecture for All-Pass Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL, 1993, pp. 1917—1920
210. T. Nishitani, P.H. Ang, F. Catthoor: *VLSI Video/Image Signal Processing*. Norwell MA, Kluwer 1993
211. T. Nomoto, S. Hosokai, T. Isokawa, R. Huga-inni: *A 4-M-Pixel CMD Image Sensor with Block and Skip Access Capability*. IEEE Transactions on Electron Devices, Vol. 44, No. 10, October 1997, pp. 1738—1746
212. J. Nossek: *Parallel Processing on VLSI Arrays*. Norwell MA, Kluwer 1991
213. K. Nourji, N. Demassieux: *Optimization of Real-Time VLSI Architecture for Distributed Arithmetic-Based Algorithms: Application to HDTV Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, London, 1994, pp. 223—226
214. S. Okazaki, Y. Fujita, N. Yamashita: *A Compact Real-Time Vision System Using Integrated Memory Array Processor Architecture*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 5, October 1995, pp. 446—452
215. OrCAD: *Electronic Design Automation Tools*, Beareton — Oregon, OrCAD Inc. 1994
216. M. Ostrowski: *Informacja obrazowa*. Warszawa, WNT, 1992
217. T. Ozaki, H. Ono, H. Tanaka, A. Sato, M. Nakai, T. Nishida: *A High-Packing Density Pixel with Punchthrough Read-Out Method for an HDTV Interline CCD*. IEEE Transactions on Electron Devices, Vol. 41, No. 7, July 1994, pp. 1128—1135
218. P. Pachowicz, R. Tadeusiewicz: *Porównanie systemu CESARO z wybranymi systemami analizy i rozpoznawania obrazów*. Elektrotechnika, t. 3, z. 2, Kraków, 1984, ss. 173—182
219. K.K. Parhi: *A Systematic Approach for Design of Digit-Serial Signal Processing Architectures*. IEEE Transactions on Circuits and Systems, Vol. 38, No. 4, April 1991, pp. 358—375
220. M.G. Parker, M. Benaissa: *Fault-Tolerant Linear Convolution Using Residue Number Systems*. Proceedings of the IEEE International Symposium on Circuits and Systems, London, 1994, pp. 441—444
221. A. Patel, S. Gudvangen: *Implementation of the Fermat Number Transform on FPGAs*. Proc. of the 7-th School: VLSI and ASIC Design, Wrocław, 1995, pp. 243—250
222. P.G. Paulin, C. Liem, M. Cornero, F. Nacabal, G. Goossens: *Embedded Software in Real-Time Signal Processing Systems: Application and Architecture Trends*. Proceedings of the IEEE, Vol. 85, No. 3, March 1997, pp. 419—435
223. T. Pavlidis: *Grafika i przetwarzanie obrazów*. Warszawa, WNT, 1987
224. N.T. Patsis, C.H. Chen, M.E. Larson: *SIMD Parallel Discrete-Event Dynamic System Simulation*. IEEE Transactions on Control Systems Technology, Vol. 5, No. 1, January 1997, pp. 30—41
225. G.G. Pechanek, M. Stojancic, S. Vassiliadis, C.J. Glossner: *MFAST: A Single Chip Highly Parallel Image Processing Architecture*. Proceedings of the IEEE International Conference on Image Processing, Washington DC, 1995, pp. 69—72
226. J.B. Peterson, R.B. O'Connor, P.M. Athanas: *Scheduling and Partitioning ANSI-C Programs onto Multi-FPGA CCM Architectures*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 178—187
227. W.D. Peterson: *The VMEbus Handbook*. Vita Publications, 1990
228. *Photonics*. Pittsfield MA, A Laurin Publication, 1992—1994
229. J. Pitas: *Digital Image Processing*. New Jersey, Prentice Hall, 1993
230. J. Pitas, A.N. Venetsanopoulos: *Order Statistic in Digital Image Processing*. Proceedings of the IEEE, Vol. 80, No. 12, December 1992, pp. 1893—1921

231. PLESSEY GEC: *Digital Signal Processing — IC Handbook*. Scots Valley CA, GEC Plessey Semiconductors, 1990
232. B. Pochopień: *Arytmetyka systemów cyfrowych*. Gliwice, Wydawnictwo Politechniki Śląskiej, 1997
233. A. Postula, S. Chen, L. Jóźwiak, D. Abramson: *Automated Synthesis of Interleaved Memory Systems for Custom Computing Machines*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 115–122
234. B. Pottier, J.L. Llopis: *Revisiting Smalltalk-80 Blocks: A Logic Generator for FPGAs*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 48–58
235. W.K. Pratt: *Digital Image Processing*. New York, John Wiley & Sons, 1987
236. G. Qiu: *An Improved Recursive Median Filtering Scheme for Image Processing*. IEEE Transactions on Image Processing, Vol. 5, No. 4, April 1996, pp. 646–648
237. S. Rajamani, P. Viswanath: *A Quantitative Analysis of Processor — Programmable Logic Interface*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 226–234
238. R. Rajkumar: *Synchronization in Real-Time Systems*, Norwell MA, Kluwer, 1991
239. A. Rangarajan, R. Chellapa, Y.T. Zhou: *A Model-Based Approach for Filtering and Edge Detection in Noisy Images*. IEEE Transactions on Circuits and Systems, Vol. 37, No. 1, January 1990, pp. 140–144
240. O. Ravn, N.A. Andersen: *A Test Bed for Experiments with Intelligent Vehicles*. Proc. of the 1st International Workshop on Intelligent Autonomous Vehicles, Hampshire UK, 1993
241. O. Ravn, N.A. Andersen: *An Intelligent Vehicle — Indoor Navigation*. Proc. of the 9 Fachgesprache uber Autonome Mobile Systeme, Munchen 1993
242. M. Rawski, T. Łuba, L. Jóźwiak, A. Chojnacki: *Efficient Logic synthesis for FPGAs with Functional Decomposition Based on Information Relationship Measures*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 8–15
243. D.R. Reddy, R.W. Horn: *Computer Architectures for Vision*. Computer Vision and Sensor Based Robots, New York, Plenum Press, 1979
244. S.J. Reeves: *On the Selection of Median Structure for Image Filtering*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 42, No. 8, August 1995, pp. 556–558
245. T.H. Reiss: *Recognizing Planar Objects using Invariant Image Features*. Berlin, Springer, 1993
246. C.V. Reventlow, M. Talmi, S. Wolf, M. Ernst, K. Muller, C. Stoffers: *System Considerations and the System Level Design of a Chip Set for Real-Time TV and HDTV Motion Estimation*. A Special Issue of: Journal of VLSI Signal Processing, vol. 5, nos. 2–3 — 1993, pp. 237–248
247. K. Ronner, J. Kneip: *Architecture and Applications of the HiPAR Video Signal Processor*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 6, No. 1, February 1996, pp. 56–66
248. P.J. Rose, B.G. Koether: *A 75 MHz CMOS Digital Convolver*. Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers & Processors, Massachusetts MA, 1990, pp. 311–314
249. A. Rosenfeld: *Parallel Image Processing using Cellular Arrays*. Computer, 1983, no 1
250. M. Rusin: *Wizyjne przetworniki optoelektryczne*. Warszawa, WKŁ, 1990
251. J.C. Russ: *The Image Processing Handbook*. CRC Press, 1995
252. N. Sakaguchi, N. Nakamura, S. Ohsawa, Y. Endo: *Dark Current Fixed Pattern noise Reduction for the 2/3-in Two-Milion Pixel HDTV STACK-CCD Imager*. IEEE Transactions on Electron Devices, Vol. 44, No. 10, October 1997, pp. 1658–1662
253. J. Serra, P. Soille: *Mathematical Morphology and Its Applications to Image Processing*. Norwell MA, Kluwer, 1994

254. I. Schaffer, M.A. Perkowski: *Synthesis of Multilevel Multiplexer Circuits for Incompletely Specified Multioutput Boolean Functions with Mapping to Multiplexer Based FPGA's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 12, No. 11, November 1993, pp. 1655–1664
255. M. Schlag, J. Kong, P.K. Chan: *Routability-Driven Technology Mapping for Lookup Table-Based FPGA's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 13, No. 1, January 1994, pp. 13–26
256. D. Scheffer, B. Dierick, G. Meynants: *Random Addressable 2048 × 2048 Active Pixel Image Sensor*. IEEE Transactions on Electron Devices, Vol. 44, No. 10, October 1997, pp. 1716–1720
257. U. Schmidt, S. Mehrgardt: *Wavefront Array Processor for Video Applications*. Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers & Processors, Massachusetts MA, 1990, pp. 307–310
258. SGS THOMSON: *Image Processing Databook*. Roma, SGS Thomson Inc. 1990
259. D.H. Shaefer: *The Characterization and Representation of Massively Parallel Computing Structures*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 461–468
260. L. Shapiro, A. Rosenfeld: *Computer Vision and Image Processing*. Boston, Academic Press, 1992
261. M.H. Sheu, J.F. Wang, J.S. Chen, A.N. Suen, Y.L. Jeang, J.Y. Lee: *A Data-Reuse Architecture for Gray-Scale Morphologic Operations*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 39, No. 10, October 1992, pp. 753–756
262. M.H. Sheu, J.F. Wang, J.Y. Lee, L.Y. Liu: *An Expandable Chip Design for Gray-Scale Morphological Operations*. Proc. of the IEEE International Symposium on Circuits and Systems, Chicago 1993, pp. 1563–1566
263. M. Shobaki: *Verification of Embedded Real-Time Systems using Hardware/Software Cosimulation*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 46–50
264. N.D. Sidiropoulos, J.S. Baras, C.A. Berenstein: *Optimal Filtering of Digital Binary Images Corrupted by Union/Intersection Noise*. IEEE Transactions on Image Processing, Vol. 3, No. 4, July 1994, pp. 382–403
265. S. Singh, J. Hogg, C. McAuley: *Expressing Dynamic Reconfiguration by Partial Evaluation*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 188–194
266. J. Siu, J. Li, S. Luthi: *A Real-Time 2-D Median Based Filter for Video Signals*. IEEE Transactions on Consumer Electronics, Vol. 39, No. 2, May 1993, pp. 115–121
267. W. Sarbek: *Metody reprezentacji obrazów cyfrowych*. Warszawa, PLJ, 1993
268. V. Skylarov, N. Lau, R.S. Monteiro, A. Melo, A. Oliveira, K. Kondratjuk: *Design of Virtual Digital Controllers Based on Dynamically Reconfigurable FPGAs*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 200–203
269. A. Skorupski: *Podstawy budowy i działania komputerów*. Warszawa, WKiŁ, 1997
270. J.P. Strong: *Computations on the Massively Parallel Processor at the Goddard Space Flight Center*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 548–558
271. Y. Sun, T.C. Wang, C.K. Wong, C.L. Liu: *Routing for Symmetric FPGA's and FPIC's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 1, January 1997, pp. 20–31
272. S. Sunder, V. Ramachandran: *Systolic Implementation of Multidimensional Nonrecursive Digital Filters*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 3, No. 6, December 1993, pp. 399–407
273. A. Śluzek: *Komputerowa analiza obrazów*. Warszawa, Politechnika Warszawska, 1991
274. R. Tadeusiewicz: *Komputerowa analiza obrazów i jej zastosowania*. Elektrotechnika, t. 1, z. 2, Kraków 1982, ss. 85–94
275. R. Tadeusiewicz: *Systemy wizyjne robotów przemysłowych*. Warszawa, WNT, 1992

276. R. Tadeusiewicz, M. Flasiński: *Rozpoznawanie obrazów*. Warszawa, PWN, 1991
277. R. Tadeusiewicz: *Computer System Dedicated to Image Processing*. Proc. of the 5-th School: Computer Vision and Graphics, Wrocław, 1994, pp. 111–126
278. Y. Takemura, K. Suda, H. Serita, H. Kotaki, T. Sugiki, A. Iida, O. Osonoe, T. Sakurai, N. Sasano, H. Sekine: *HDTV Compact Digital Camera using 2/3" 1.3 m. CCD Image Sensor*. IEEE Transactions on Consumer Electronics, Vol. 41, No. 1, February 1995, pp. 81–88
279. A.S. Tanenbaum: *Rozproszone systemy operacyjne*. Warszawa, PWN, 1997
280. S.L. Tanimoto: *Memory System for Highly Parallel Computers*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 403–415
281. TEXAS INSTRUMENTS: *Field Programmable Gate Array — Databook*. Bedford — UK. Texas Instruments Ltd. 1992
282. S. Thakur, Y.W. Chang, D.F. Wong, S. Muthukrishnan: *Algorithms for an FPGA Switch Module Routing Problem with Application to Global Routing*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 1, January 1997, pp. 32–46
283. S. Thakur, D.F. Wong: *Simultaneous Area and Delay Minimum K-LUT Mapping for K-Exact Networks*. Proc. of the Int. Conference on computer Design: VLSI in Computers & Processors. IEEE Computer Society Press, 1995, pp. 402–408
284. TOSHIBA: *MOS Memory Products. Data Book*. Tokyo, Toshiba Inc. 1988
285. S.M. Trimberger: *Field Programmable Gate Array Technology*. Norwell MA, Kluwer, 1994
286. A. Tsutsui, T. Miyazaki: *ANT-on-YARDS: FPGA/MPU Hybrid Architecture for Telecommunication Data Processing*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 2, June 1998, pp. 199–211
287. A. Tsutsui, K. Yamada, T. Miyazaki, N. Ohta: *Special Purpose FPGA for High-speed Digital Telecommunication Systems*. Proc. of the Int. Conference on computer Design: VLSI in Computers & Processors. IEEE Computer Society Press, 1995, pp. 486–491
288. L. Uhr, S. Levialdi, K. Preston, M.J.B. Duff: *Evaluation of Multicomputers for Image Processing*. Boston, Academic Press, 1986
289. VIEWLOGIC: *WorkView*. Marlboro MA, ViewLogic Systems Inc. 1991
290. J. Villasenor, C. Jones, B. Schoner: *Video Communications Using Rapidly Reconfigurable Hardware*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 5, No. 6, December 1995, pp. 565–567
291. J. Villasenor, B. Schoner, K.N. Chia, C. Zapata: *Configurable Computing Solutions for Automatic Target Recognition*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 70–79
292. J. Vincente, J. Lanchares, R. Hermida: *RSR: A New Rectilinear Steiner Minimum Tree Approximation for FPGA Placement and Global Routing*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 192–195
293. VITA: *VME revision D*. VITA — Technical Committee of the VMEbus International Trade Association, 1990
294. J.E. Vuillemin, P. Bertin, D. Roncin, M. Shand, H.H. Touati, P. Boucard: *Programmable Active Memories: Reconfigurable Systems Come of Age*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 4, No. 1, March 1996, pp. 56–69
295. C.L. Wang, P.B. Bhat, V.K. Prasanna: *High-Performance Computing for Vision*. Proceedings of the IEEE, Special Issue on Parallel Architectures for Image Processing, Vol. 84, No. 7, July 1996, pp. 931–946
296. X. Wang: *Generalized Multistage Median Filters*. IEEE Transactions on Image Processing, Vol. 1, No. 4, October 1992, pp. 543–545
297. D. Wang, D.C. He: *A Fast Implementation of 1-D Grayscale Morphological Filters*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 41, No. 9, September 1994, pp. 634–636

298. C.D. Watkins, A. Sadun, S. Marenka: *Nowoczesne metody przetwarzania obrazu*. Warszawa, WNT, 1995
299. C.C. Weems: *Architectural Requirements of Image Understanding with Respect to Parallel Processing*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 537–547
300. K.A. Wen, S.C. Chen, J.T. Yao: *Single Processor Design for 2-D Wiener Filter*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL 1993, pp. 2039–2042
301. A. Wenban, G. Brown: *A Software Development System for FPGA-Based Data Acquisition Systems*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 28–37
302. K. Wiatr: *Wieloprocesorowy, modułowy system do sterowania procesów lub urządzeń w czasie rzeczywistym, o architekturze optymalizowanej ze względu na parametry niezawodności i szybkości obsługi*. Sprawozdanie z realizacji projektu badawczego KBN nr 7 0827 91 01, Kraków, AGH, 1991
303. K. Wiatr, P. Rajda: *Możliwości kształtowania komunikacji międzymodułowej dla różnych architektur software'u w systemie operacyjnym OS-9*. Mat. Konf. Sztuczna Inteligencja i Cybernetyka Rozwoju — CIR'93, Warszawa, 1993, ss. 287–297
304. K. Wiatr, J. Kasperek: *Możliwości synchronizacji i optymalizacji parametru szybkości przetwarzania systemu wieloprocesorowego poprzez przesyłanie informacji w trybie „Message Broadcast”*, ZN AGH, Automatyka, z. 67, Kraków, 1993, ss. 114–123
305. K. Wiatr, P. Rajda, J. Kasperek: *Sposób i układ do przetwarzania sygnału wizyjnego*. Patent nr P-300692 z dnia 12.10.1993 r.
306. K. Wiatr, P. Rajda: *Sprzętowe rozwiązania pamięci dwuportowej dla potrzeb systemów wieloprocesorowych w standardzie VME*. ZN AGH, Automatyka, z. 67, Kraków 1993, ss. 125–141
307. K. Wiatr, J. Kasperek: *Dopasowanie trybu dostępu do pamięci dynamicznych z możliwościami transmisji blokowej standardu VME*. Elektrotechnika, t. 12, z. 3, Kraków, 1993, ss. 111–139
308. K. Wiatr, J. Kasperek, P. Rajda: *Multiprocessor Unit for a Fast Image Data Preprocessing in Real-Time Application*. Proc. of the 5-th School: Computer Vision and Graphics, Wrocław, 1994, pp. 339–344
309. K. Wiatr, J. Kasperek, P. Rajda, M. Gorgoń: *Układ procesorów potokowych do obróbki sygnału wizyjnego*. Patent nr OWP-I-P/218 z dnia 20.4.1994 r.
310. K. Wiatr: *System wieloprocesorowy do sterowania urządzeń w czasie rzeczywistym wykorzystujący algorytmy analizy obrazów pochodzących z kamery wizyjnej o architekturze optymalizowanej ze względu na parametr szybkości przetwarzania*. Sprawozdanie z realizacji projektu badawczego KBN nr 8 8518 92 03, Kraków, AGH, 1994
311. K. Wiatr, J. Kasperek: *Możliwości dopasowania sygnału standardowej kamery dla potrzeb potokowej obróbki danych w systemach wizyjnych czasu rzeczywistego*. Elektrotechnika, t. 15, z. 2, Kraków, 1996, ss. 97–106
312. K. Wiatr, P. Rajda: *Architektura potokowa procesorów specjalizowanych dla potrzeb cyfrowej obróbki sygnału wizyjnego w czasie rzeczywistym na przykładzie operacji konwolucji*. Elektrotechnika, t. 15, z. 2, Kraków, 1996, ss. 119–136
313. K. Wiatr, J. Kasperek, P. Rajda: *Xilinx FPGA — Based Frame Grabber for Image Preprocessing*. Proc. of the 7-th School: VLSI and ASIC Design, Wrocław 1995, pp. 273–279
314. K. Wiatr, J. Kasperek, P. Rajda: *FPGA Application in Real-Time Pipeline Image Processing System*. Proc. of the Conference: Programmable Devices and Systems PDS'95, Gliwice, 1995, pp. 65–69
315. K. Wiatr: *System wieloprocesorowy do sterowania urządzeń w czasie rzeczywistym wykorzystujący algorytmy analizy obrazów pochodzących z kamery wizyjnej o architekturze optymalizowanej ze względu na parametr szybkości przetwarzania*. Mat. Konf. Automatyka — Granty'95, Warszawa, 1995, s. 78–88
316. K. Wiatr: *Rozpoznawanie kształtów i ruchu obiektów dla potrzeb sterowania urządzeń w czasie rzeczywistym z wykorzystaniem dedykowanych architektur systemu wieloprocesorowego i potokowej obróbki wstępnej danych obrazowych*. Mat. Konf. Automatyka — Granty'95, Warszawa, 1995, s. 162–170

317. K. Wiatr, J. Kaspererek, P. Rajda, P. Russek: *Sposób transmisji po magistrali do kilku odbiorców*. Patent nr OWP-I-P/270 z dnia 12.7.1995 r.
318. K. Wiatr: *A Pipelined Multiprocessor Architecture for a Fast image Data Preprocessing in Real Time Applications*. Proc. of the IASTED & IEEE International Conference: Signal and Image Processing — SIP-95, Las Vegas NV 1995, Anaheim — Calgary — Zurich, IASTED Acta Press, 1995, pp. 13–16
319. K. Wiatr: *Pipelined Architecture of Reconfigurable Specialised Processors for a Real-Time Image Data Pre-Processing*. Proc. of the IEEE International Conference on Signal Processing — ICSP-96, Beijing — China 1996, IEEE Press 1996, pp. 649–652
320. K. Wiatr, P. Russek: *Synchronizacja pracy procesorów w systemie do analizy obrazów wizyjnych w czasie rzeczywistym*. Elektrotechnika, 1996, t. 15, z. 4, Kraków, ss. 335–346
321. K. Wiatr: *Rekonfigurowalna architektura procesorów potokowych do obróbki wstępnej danych wizyjnych w czasie rzeczywistym*. Elektrotechnika, 1996, t. 15, z. 4, Kraków, ss. 291–301
322. K. Wiatr: *Reconfigurable Specialised Hardware Processor for a Real-Time Fast Image Data Pre-Processing*. Proc. of the IASTED International Conference: Signal and Image Processing — SIP-96, Orlando — FL 1996, Anaheim — Calgary — Tokyo, IASTED Acta Press 1996, pp. 100–104
323. K. Wiatr: *Rozpoznawanie kształtów i ruchu obiektów dla potrzeb sterowania urządzeń w czasie rzeczywistym z wykorzystaniem dedykowanych architektur systemu wieloprocessorowego i potokowej obróbki wstępnej danych obrazowych*. Sprawozdanie z realizacji projektu badawczego KBN nr 8 S505 008 07, Kraków, AGH, 1997
324. K. Wiatr: *Architektura specjalizowanych procesorów potokowych do obróbki wstępnej danych wizyjnych w czasie rzeczywistym*. Kwartalnik Elektroniki i Telekomunikacji, PAN, 1997, Vol. 43, z. 1, ss. 121–148
325. K. Wiatr: *A Dedicated Hardware Logic Processor for a Real-Time Image Data Pre-Processing*. Proc. of the IASTED & IEEE International Conference: CONTROL'97, Cancun — Mexico 1997, Anaheim — Calgary — Tokyo, IASTED Acta Press 1997, pp. 207–210
326. K. Wiatr: *Specialised Architecture of Dedicated Hardware Processor for a Real-Time Image Data Pre-Processing*. Proc. of the EUROMICRO Conference: Real-Time Systems, Toledo — Spain, 1997, Washington — Brussels — Tokyo, IEEE Computer Press 1997, pp. 180
327. K. Wiatr: *Dedicated Hardware Processor for a Real-Time Image Data Pre-Processing Implemented in FPGA Structure*. Proc. of the 6th International Conference on Image Analysis and Processing — ICIAP'97, Florence — Italy 1997, Berlin, Springer-Verlag, 1997, vol. II, pp. 69–75
328. K. Wiatr: *Specialized Hardware Median Processor Implemented in FPGA Structure for Pipeline Architecture to a Real-Time Image Data Pre-Processing*. Proc. of the IASTED International Conference: Parallel and Distributed Computing and Systems, Washington 1997, Anaheim — Calgary — Tokyo, IASTED Acta Press 1997, pp. 319–324
329. K. Wiatr: *A pipelined Architecture of Specialised Hardware Processors in FPGA Structure for a Real-Time Image Data Pre-Processing*. *Image Processing and Communications*. An International Journal, Bydgoszcz, Published by the Institute of Telecommunications 1977, vol. 3, no. 1–2, pp. 71–84
330. K. Wiatr, T. Salamon, T. Uhl: *Realizacja sterowania wizyjnego w środowisku Matlab/Simulink/dSPACE*. Krajowa Konferencja: Metody i systemy komputerowe w badaniach naukowych i projektowaniu inżynierskim — podejście mechatroniczne, Kraków, 1997, ss. 653–660
331. K. Wiatr: *Pipelined Architecture of Specialised Reconfigurable Processors in FPGA Structures for Real-Time Image Data Pre-Processing*. IEEE Computer Society: Proceedings of the 24th EUROMICRO Conference — Digital Systems Design, Vasteras — Sweden 1998, pp. 131–138
332. K. Wiatr: *Dedicated System Architecture for Parallel Image computation used Specialised Hardware Processors Implemented in FPGA Structures*. International Journal of Parallel and Distributed Systems and Networks, Pittsburgh, 1998
333. K. Wiatr, M. Gorgoń: *Implementation and Optimization of FPGA Systems Designing in Real-Time Vision Systems*. Proc. of the International Conference: Programmable Devices and Systems, Gliwice, 1998, s. 135–142

334. K. Wiatr: *Pipelined Architecture of Specialised Processors in FPGA Chips for a Fast Image Data Pre-Processing*. Proc. of the 5th International Workshop on Systems, Signals and Image Processing — IWSSIP'98, Zagreb — Croatia, 1998, s. 255–258
335. K. Wiatr: *Analiza parametrów czasowych architektury potokowej specjalizowanych procesorów sprzętowych*. Kwartalnik Elektroniki i Telekomunikacji PAN, 1998, Vol. 44, z. 3, ss. 87–108
336. K. Wiatr: *Architektura potokowa specjalizowanych procesorów sprzętowych do wstępnego przetwarzania obrazów w systemach wizyjnych czasu rzeczywistego*. Kraków, Wydawnictwa AGH, 1998
337. H.A.G. Wijshoff: *Data Organisation in Parallel Computers*. Norwell MA, Kluwer, 1988
338. N.B. Wilding, A.S. Trew, K.A. Hawick, G.S. Pawley: *Scientific Modeling with Massively Parallel SIMD Computers*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 574–585
339. S.S. Wilson: *Teaching Network Connectivity using Simulated Annealing on a Massively Parallel Processors*. Proceedings of the IEEE, Vol. 79, No. 4, April 1991, pp. 559–566
340. R.D. Witting, P. Chow: *OneChip: An FPGA Processor with Reconfigurable Logic*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 126–135
341. R.G. Wood, R.A. Rutenbar: *FPGA Routing and Routability Estimation Via Boolean Satisfiability*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 6, No. 2, June 1998, pp. 222–231
342. R. Woods, A. Cassidy, J. Gray: *VLSI Architectures for Field Programmable Gate Arrays: A Case Study*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 2–9
343. J. Woźnicki: *Podstawowe techniki przetwarzania obrazu*. Warszawa, WKiŁ, 1996
344. Y.L. Wu, M. Marek-Sadowska: *Routing for Array-Type FPGA's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 5, May 1997, pp. 506–518
345. Y.L. Wu, M. Marek-Sadowska: *Graph Based Analysis of 2-D FPGA Routing*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 15, No. 1, January 1996, pp. 33–44
346. T. Viero, K. Oistamo, Y. Nuevo: *Three-Dimensional Median-Related Filters for Color Image Sequence Filtering*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 4, No. 2, April 1994, pp. 129–142
347. XILINX: *The Programmable Logic Data Book*. San Jose CA, Xilinx Inc. 1996
349. XILINX: *XACT Development Systems*. San Jose CA, Xilinx Inc. 1994
XILINX: *XCELL — The Quarterly Journal for Xilinx Programmable Logic Users*. San Jose CA, Xilinx Inc. 1994–1996
350. XILINX: *The Programmable Logic Data Book*. San Jose CA, Xilinx Inc. 1998
351. M. Yamagishi, M. Negishi, H. Yamada, T. Tsunakawaini: *A 2 Million Pixel FIT-CCD Image Sensor for HDTV Camera Systems*. IEEE Transactions on Electron Devices, Vol. 38, No. 5, May 1991, pp. 976–980
352. T. Yamauchi, S. Nakaya, N. Kajihara: *SOP: A Reconfigurable Massively Parallel System and its Control-Data-Flow Based Compiling Method*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 148–156
353. H. Yamauchi, Y. Tashiro, T. Minami, Y. Suzuki: *Architecture and Implementation of a Highly Parallel Single-Chip Video DSP*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 2, No. 2, June 1992, pp. 207–220
354. X.D. Yang: *Design of Fast Connected Components Hardware*. IEEE Trans. 1988, pp. 937–944
355. X. Yang, P.S. Toh: *Adaptive Fuzzy Multilevel Median Filter*. IEEE Transactions on Image Processing, Vol. 4, No. 5, May 1995, pp. 680–692
356. D. Yeh, G. Faygin, P. Chow: *RACER: A Reconfigurable Constraint-Length 14 Viterbi Decoder*. Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines, California 1996, pp. 60–69
357. C.W. Yeh, L.T. Liu, C.K. Cheng, T.C. Hu, S. Ahmed, M. Liddel: *Block-Oriented Programmable Design with Switching Network Interconnect*. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 2, No. 1, March 1994, pp. 45–53

358. H. Youssef, R.B. Lin, E. Shragowitz: *Bounds on Net Delays for VLSI Circuits*. IEEE Transactions on Circuits and Systems — II: Analog and Digital Signal Processing, Vol. 39, No. 11, November 1991, pp. 815–824
359. J. Zalewski: *Advanced Multimicroprocessor Bus Architecture*. New York, IEEE 1995
360. B. Zeng, A.N. Venetsanopoulos: *A Split Sorting-Algorithm for Generalized Order Statistic Filters*. Proceedings of the IEEE International Symposium on Circuits and Systems, Chicago IL 1993, pp. 946–949
361. B. Zeng: *Convergence Properties of Median and Weighted Median Filters*. IEEE Transactions on Signal Processing, Vol. 42, No. 12, December 1994, pp. 3515–3518
362. D. Zhao, D.G. Daut: *A Real-Time Column Array Processor Architectures for Images*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 2, No. 1, March 1992, pp. 38–48
363. D. Zhou, F.P. Preparata, S.M. Kang: *Interconnection delay in Very High-Speed VLSI*. IEEE Transactions on Circuits and Systems, Vol. 38, No. 7, July 1991, pp. 779–790
364. K. Zhu, D.F. Wong: *Clock Skew Minimization During FPGA Placement*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 4, April 1997, pp. 376–385
365. K. Zhu, D.F. Wong: *Switch Bound Allocation for Maximizing Routability in Timing-Driven Routing of FPGA's*. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 17, No. 4, April 1998, pp. 316–323
366. Z. Zilic, Z.G. Vranesic: *Using Decision Diagrams to Design ULMs for FPGAs*. IEEE Transactions on Computers, Vol. 47, No. 9, September 1998, pp. 971–982

K. WIATR

THE MIDS ARCHITECTURE OF SPECIALISED PROCESSORS IN THE REAL-TIME VISION SYSTEMS ON THE BACKGROUND NEW SEARCHING WORKS

Summary

This paper presents new searching works for real-time visions systems. In particular the paper show many publications connected with CCD chips and video signal standards, analog image to digital signal convert, real-time visions systems, algorithms for image computation (to implementation in the VLSI and FPGA chips), parallel and multiprocessors vision systems architecture, dedicated pipelined architecture of specialised hardware processors, and implementation specialised processors in the FPGA chips. On this background author presents results of his searching works.

These searching works consisted in striving after parallel arrangement of image preprocessing computations, by selection of such computational elements (processors) and such a transmission between them (architecture), that the total computation time was the least possible. Structures of specialised hardware processors have been worked out, whose internal diagram univocally defines the type of only one image pre-processing operation (realised in hardware) and allows for considerable shortening of the time of the operation. To verify the concept the following processors have been worked out: look-up-table, median filter, convolver, logic processor, and morphological processor. The implementation of these processors in programmable FPGA structures allows for dynamic change of image processing functions in the course of normal operation of the system. As a result of the work described, a concept of universal reconfigurable processor has been developed, in which one can implement any desired operation of image pre-processing (from the system bus level). The concepts presented have been verified by experiments, using processors implemented in FPGA structures.

The work presented is a contribution to world wide intense research on developing user dedicated "Custom Computing Machines" and "Reconfigurable Computers".

Key words: specialised processors, systems architecture, image computation, programmable logic devices

SPIS TREŚCI

K. Ślot: Modelowanie równań różniczkowych cząstkowych przy użyciu uniwersalnych sieci neuronowych komórkowych	125
S. Lindner: Modelowanie symulacyjne kanałów binarnych	149
K. Olejarczyk: Symulacja przetworników analogowo-cyfrowych delta-sigma z symulatorem Amber	163
D. Kania: Algorytmy podziału wyjść umożliwiające realizację układów cyfrowych w strukturach PLD	189
J. Gołębiowski: Model membranowego czujnika z przetwornikami piezoelektrycznymi do generacji i detekcji ultradźwiękowych fal płytowych typu Lamba	203
B. Stec, A. Dobrowolski: Wieloczęstotliwościowa termografia mikrofalowa	225
M. Gajer: Wykorzystanie wieloprocessorowego układu TMS 320C80 w dziedzinie przetwarzania i rozpoznawania obrazów	235
K. Wiatr: Architektura MISD procesorów specjalizowanych w systemach wizyjnych czasu rzeczywistego na tle aktualnych prac badawczych	251

CONTENTS

K. Ślot: The modelling reaction-diffusion partial differential equations of the cellural neural network universal machine	125
S. Lindner: The simulation modelling of binary channels	149
K. Olejarczyk: Simulation of delta-sigma analog-to-digital converters with the simulator Amber	163
D. Kania: Output partitioning algorithms which enable implementation of digital circuits in PLD structures	189
J. Gołębiowski: The model of the membrane sensor with the piezoelectric transducer to the generation and detection of the ultrasonic plate waves Lamba type	203
B. Stec, A. Dobrowolski: Multifrequency microwave thermography	225
M. Gajer: The usage of the TMS 320C80 multiprocessor system in the field of image processing and recognition	235
K. Wiatr: The architecture of MISD specialised processors in the real time vision systems on the background searching works	251

INFORMATION FOR THE AUTORS OF K.E.T.

The KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI (K.E.T.) — ELECTRONICS AND TELECOMMUNICATION QUARTERLY is a journal of the Committee for Electronics and Telecommunication of the Polish Academy of Sciences and is a direct continuation for 44 years of the quarterly ROZPRAWY ELEKTROTECHNICZNE, that has been published also under the auspices of the Polish Academy of Sciences.

Its pages contain scientific articles concerned with theory and application of electronics, telecommunication, microelectronics, optoelectronics, radioengineering and medical electronics.

Accordingly, its pages contain scientific articles, that are original works both contributory and concerned with comparative analyses of methods or systems synthetic reviews of a given field, state-of-art discussions of a given field, and presentations of developments in a given technology.

Regular papers are accepted for publication in Polish or English, but title, the extensive summary and figure caption must be bilingual. Bilingual summary must show the significance of the results shown in the paper, emphasizing limitations and advantages, possible applications and recommendations for further works.

K.E.T. is meant for specialists and students with above subjects.

FORMAL PREPARATION OF MANUSCRIPT

Texts

Texts to be submitted to K.E.T. may be typed on one side of the paper only with double spacing between the lines and a 4 cm margin or using of wordprocessing with text formatters based on MS DOS, APPLE/MAC such as Microsoft word. In any case the text formatter must be specified on the Author/s, their Affiliations and relevant addresses.

All text pages should be numbered. Regular papers should have a length of no more than 30 pages.

Printing types and symbols

For letteral symbols, units and graphical symbols, the recommendation of IEE (International Electrotechnical Commission) and ISO (International Organisation of Standardisation) must be followed.

- In case of manuscripts conventionally typed (i.e. without wordprocessing), letters to be printed in italic should be underlined; letters to be printed in bold should be double interlined.
- Formulae and footnotes should be numbered in accordance with the quotation order followed in the text.
- The serial number of each formulae should be written at right side, between round brackets.
- The serial number of footnotes should be written between round brackets and upper.
- Mathematical details, ancillary to the main discussion of the paper may be included in one or more appendixes, that are printed after conclusions and before references.

In manuscript containing lot of symbols, a glossary may be included after introduction.

References

References are usually gathered at the end of the text and numbered according to the order of quotation in the text. The serial numbers should be written without square brackets (i.e. 1, 2...)

Illustrations

The originals of illustrations may be either drawings or photographs (black or colour). Each copy of paper must include two complete sets of illustrations.

All rights reserved. No part of the publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means: electronic, photocopying, recording or otherwise, without the prior permission of the copyright owner — Redaction K.E.T.

INFORMACJE DLA AUTORÓW

Redakcja przyjmuje do publikowania prace oryginalne, przeglądowe i monograficzne wchodzące w zakres szeroko pojętej elektroniki. Ponieważ KWARTALNIK ELEKTRONIKI I TELEKOMUNIKACJI jest czasopismem Komitetu Elektroniki i Telekomunikacji Polskiej Akademii Nauk, w związku z tym na jego łamach znajdują się prace naukowe dotyczące podstaw teoretycznych i zastosowań z zakresu elektroniki, telekomunikacji, mikroelektroniki, optoelektroniki, radiotechniki i elektroniki medycznej.

Artykuły powinny charakteryzować oryginalne ujęcie zagadnienia, własna klasyfikacja, krytyczna ocena (teorii lub metod), omówienie aktualnego stanu, lub postępu danej gałęzi techniki oraz omówienie perspektyw rozwojowych.

Artykuły publikowane w innych czasopismach nie mogą być kierowane do druku w Kwartalniku Elektroniki i Telekomunikacji w drugiej kolejności zgłoszenia.

Objętość artykułu nie powinna przekraczać 30 stron po około 1800 znaków na stronie.

Wymagania podstawowe: Artykuły należy nadsyłać w maszynopisie pisany jednostronnie lub na wyraznym czarno-białym wydruku komputerowym, w 2 egzemplarzach, w języku polskim lub angielskim wybranym przez autora. Tekst artykułu musi być poprzedzony tytułem pracy, imieniem i nazwiskiem Autora wraz z podaniem miejsca jego pracy. Wszystkie strony muszą mieć numerację ciągłą.

Sposób pisania tekstu: Tekst powinien być pisany bez używania wyróżnień, a w szczególności nie dopuszcza się spacjiowania, podkreślania i pisania tekstu dużymi literami z wyjątkiem wyrazów, które umownie pisze się dużymi literami (np. FORTRAN). Proponowane wyróżnienia Autor może zaznaczyć w maszynopisie zwykłym ołówkiem za pomocą przyjętych znaków adustacyjnych np. podkreślenie linią przerywaną oznacza spaciowanie (rozstrzeżenie), podkreślenie linią ciągłą — pogrubienie, podkreślenie wężym — kursywa. Tekst powinien być napisany z podwójnym odstępem między wierszami, tytuły i podtytuły małymi literami. Marginesy z każdej strony powinny mieć około 35 mm. Przy podziale pracy na rozdziały i podrozdziały cyfrowe ich oznaczenia nie powinny być większe niż III stopnia (np. 4.1.1).

Sposób pisania tablic: Tablice powinny być pisane na oddzielnych stronach. Tytuły rubryk pionowych i poziomych powinny być napisane małymi literami z podwójnym odstępem między wierszami. Przypisy (notki) dotyczące tablic należy pisać bezpośrednio pod tablicą. Tablice należy numerować kolejno liczbami arabskimi, u góry każdej tablicy podać tytuł. Tablice umieścić na końcu maszynopisu. Przyjmowane są tablice algorytmów i programy na wydrukach komputerowych. W tym przypadku zachowany jest ich oryginalny układ.

Sposób pisania wzorów matematycznych: Rozmieszczenie znaków, cyfr, liter i odstępów powinno być zbliżone do rozmieszczenia elementów druku. Wskaźniki i wykładniki potęg powinny być napisane wyraźnie i być prawidłowo obniżone lub podwyższone w stosunku do linii wiersza podstawowego. Znaki nad literami i cyframi, całkami i in. symbolami (strzałki, linie, kropki, daszki) powinny być umieszczone dokładnie nad tymi elementami, do których się odnoszą. Numery wzorów cyframi arabskimi powinny być kolejne i umieszczone w nawiasach okrągłych z prawej strony. Nazwy jednostek, symbole literowe i graficzne powinny być zgodne z wytycznymi IEE (International Electrotechnical Commission) oraz ISO (International Organization of Standardization).

Powołania: Powołania na publikacje powinny być umieszczone na ostatnich stronach tekstu pod tytułem „Bibliografia”, opatrzone numeracją kolejną bez nawiasów. Numeracja ta powinna być zgodna z odnośnikami w tekście artykułu. Przykłady opisu publikacji:

- periodycznej: F. Valdoni: A new millimetre wave satellite, E.T.T. 1990, vol. 2, nr 5, p. 553
- nieperiodycznej: K. Andersen: A resource allocation framework. XVI International Symposium, Stockholm (Sweden), may 1991, paper A 2.4
- książki: Y.P. Tvidis: Operation and modeling of the MOS transistors. Mc Graw-Hill, New York 1987, p. 141 — 148

Materiały ilustracyjne: Rysunki powinny być wykonane wyraźnie, na papierze gładkim, lub milimetrowym w formacie nie mniejszym niż 9×12 cm. Mogą być także w postaci wydruku komputerowego. Fotografie lub diapozytywy przyjmowane są raczej czarno-białe w formacie nie przekraczającym 10×15 cm. Na marginesie każdego rysunku i na odwrocie fotografii powinno być napisane ołówkiem imię i nazwisko Autora oraz skrót tytułu artykułu, do którego są przeznaczone. Spis podpisów pod rysunki i fotografie powinien być umieszczony na oddzielnej stronie.

Streszczenia: Do każdego artykułu musi być dołączone obszerne — do 60 wierszy — streszczenie z tytułem artykułu. Streszczenia (Summary) muszą być w języku polskim i angielskim. Streszczenie powinno wyjaśniać główny cel pracy, wskazywać korzyści i ograniczenia, możliwe zastosowania i zalecenia dla dalszego rozwoju danej gałęzi techniki. Pod streszczeniami powinny być podane słowa kluczowe.

Autorowi przysługuje bezpłatnie 20 odbitek artykułu. Dodatkowe egzemplarze odbitek, lub cały zeszyt Autor może zamówić u wydawcy na własny koszt.

Autora obowiązuje korekta autorska, którą powinien wykonać w ciągu 3 dni od daty otrzymania tekstu z Redakcji i zwrócić osobiście, lub listownie pod adresem Redakcji. Korekta powinna być naniesiona na przekazanych Autorowi szpaltach na marginesach ew. na osobnym rkuszu w przypadku uzupełnień tekstu większych niż dwa wiersze. W przypadku nie zwrócenia korekty w terminie, korektę przeprowadza Redakcja Techniczna Wydawcy.

Redakcja prosi Autorów o powiadomienie ją o zmianie miejsca pracy i adresu prywatnego.

Subs

T

insti

PW)

fax.

Tele

Sub

acco

1-43

50%

For

Pol

Ru

yea

Subscription for external subscribers:

The promotional subscription price in 1999 is \$ 90 including postage for institutions. Subscriptions should be sent to the publisher, Polish Scientific Publishers PWN Ltd, Journal Division, Miodowa 10, 00-250 Warsaw, POLAND, fax. (48) (22) 826 09 50, (48) (22) 826 71 63, with a copy by fast to Electronics and Telecommunications Quarterly 00-665 Warsaw, ul. Nowowiejska 15/19, r. 470. Subscription is occupied after showing a cheque or transfer documents. Our bank account is as follws: Bank account: PBK VIII O/Warszawa nr 11101037-1052-2700-1-43. At subscripber's request this journal will be air mailed at additional postage to 50% dross proce to European countries and 65% overseas.

Subscription orders available through the local press distribution or through the Foreign Trade Enterprise ARS Polona, 00-068 Warszawa, Krakowskie Przedmieści 7, Poland.

Ruch S.A. fulfils foreign customers orders, starting from any issue in the calendary year: tel. (48) (22) 620 120 39, fax. (48) (22) 620 17 62.