

3 1972

P. 2229/72

prace

**Instytutu
Maszyn
Matematycznych**

rok XIV



P. 2229/72

prace

Instytutu

Maszyn

Matematycznych

Copyright © 1972 - by Instytut Maszyn Matematycznych
Poland

Wszelkie prawa zastrzeżone

Komitet Redakcyjny

Bartłomiej GŁOWACKI, Andrzej KOJEMSKI, Roman KULESZA /red. naczelny/
Włodzimierz MARDAL /z-ca red. naczelnego/, Jan RELUGA

Sekretarz Redakcji: Romana NITKOWSKA

Adres Redakcji: Instytut Maszyn Matematycznych
Branżowy Ośrodek INTE
Warszawa, ul. Krzywickiego 34
tel. 28-37-29

SPIS TREŚCI ZESZYTU 3

СОДЕРЖАНИЕ
CONTENTS

1. WIATER A.

Badanie symulacyjne niektórych algorytmów podziału czasu procesora w systemach przetwarzania informacji	5
Моделирующее исследование некоторых алгоритмов разделения времени процессора в системах обработки информации /Резюме/	22
Modelling investigation of some processor time-sharing algorithms in information processing systems /Summary/	24

2. HŁAWICZKA A.

O pewnej metodzie syntezy minimalnych bezhazardowych kombinacyjnych układów przełączających TANT	25
O некотором методе синтеза минимальных неазартных комбинированных переключающих схем TANT /Резюме/	46
On a certain method of minimal nonhazardous synthesis of combinative switching TANT element /Summary/	46

AUTORZY ARTYKUŁÓW ZAMIESZCZONYCH W ZESZYCIE 3 "PRAC IMM"	47
--	----

BADANIE SYMULACYJNE NIEKTÓRYCH ALGORYTMÓW
PODZIAŁU CZASU PROCESORA W SYSTEMACH PRZE-
TWARZANIA INFORMACJI

Andrzej WIATER

Pracę złożono 3.12.1971

Artykuł zawiera porównanie wyników teoretycznych i wyników otrzymanych z symulacji czterech najczęściej spotykanych algorytmów podziału czasu procesora: First-Come-First-Served, Round-Robin, Processor-Shared, Feed-Back. Przyjęto nieskończone źródło zadań z wykładniczym rozkładem czasu przetwarzania i rozkładem zgłoszeń Poissona. Do symulacji wykorzystano język ZAM-GPSS/General Purpose System Simulator/. Omówiono przyczyny błędów symulacji oraz wpływ liczby symulowanych zadań na dokładność symulacji.

1. WSTĘP

Jednym z problemów występujących w systemach przetwarzania jest wybór regulaminu obsługi zadań spełniającego narzucone wymagania. W dużej liczbie praktycznych przypadków wymagania te dotyczą czasu reakcji systemu /tzn. czasu jaki upływa od chwili zgłoszenia się zadania do systemu do chwili zakończenia obsługi tego zadania/ oraz długości kolejek zadań oczekujących na obsługę /lub liczby zadań w systemie/.

Zajmijmy się przypadkiem, gdy zadania przybywają do systemu w sposób przypadkowy oraz czas przetwarzania poszczególnych zadań jest losowy. Założmy następujące rozkłady odstępów między zgłoszeniami zadań i czasów przetwarzania [5]:

$$P(t_z \leq t) = \begin{cases} 1 - e^{-\lambda t} & t > 0 \\ 0 & t \leq 0 \end{cases} \quad \lambda > 0 \quad /1/$$

$$P(\tau_p \leq \tau) = \begin{cases} 1 - e^{-\mu \tau} & \tau > 0 \\ 0 & \tau \leq 0 \end{cases} \quad \mu > 0 \quad /2/$$

gdzie: t_z - odstęp czasu pomiędzy zgłoszeniami kolejnych zadań,
 λ - średnia liczba zadań napływających w jednostce czasu,

τ_p - czas przetwarzania zadania,

μ - średnia liczba zadań przetwarzanych w jednostce czasu.

Wprowadźmy współczynnik ρ zwany intensywnością ruchu:

$$\rho = \frac{\lambda}{\mu} \quad /3/$$

Oczywiście, aby średnia liczba zadań oczekujących na przetwarzanie była ograniczona, gdy $t \rightarrow |\infty|$, musi być spełniony warunek:

$$\rho < 1 \quad /4/$$

Oznacza to, że średni odstęp czasu między zgłoszeniem się kolejnych zadań do systemu musi być większy niż średni czas przetwarzania zadania, czyli średnia szybkość przetwarzania musi być większa niż średnia szybkość napływania zadań do systemu.

Teoria kolejek zajmuje się między innymi badaniem czasów reakcji systemu i długości kolejek w zależności od rozkładów zgłoszenia zadań i czasów przetwarzania dla różnych regulaminów obsługi.

Poniżej zostaną opisane cztery z częściej spotykanych regulaminów obsługi:

1. pierwsze zgłoszone - pierwsze obsługiwane
/First - Come - First - Served; FCFS/,

2. cykliczny jednopoziomowy
/Round - Robin; RR/,
3. z podziałem procesora
/Processor - Shared; PS/,
4. cykliczny wielopoziomowy
/Feed - Back; FB_N /.

Podane będą zależności teoretyczne określające czas przebywania zadań w systemie i porównane z wynikami symulacji tych systemów. Systemy te były modelowane za pomocą języka ZAM - GPSS /General Purpose System Simulator/ [6] na maszynie ZAM 41. Oczywiście podane zależności teoretyczne dotyczą stanu równowagi statystycznej systemu ($t \rightarrow \infty$); symulacja tych systemów została przeprowadzona dla skończonego odcinka czasu /dla odcinków czasu, w których zostało przetworzone 1000, 2000 i 3000 zadań/.

Napisanie i uruchomienie programów symulujących opisane cztery algorytmy podziału czasu procesora trwało około trzech tygodni. W tabeli 1 podano liczbę bloków /zdań/ języka ZAM-GPSS w programach symulacyjnych oraz czas pracy maszyny zużyty na przeprowadzenie pojedynczego eksperymentu z modelem /eksperymentu dla ustalonych parametrów strumienia zadań i parametrów regulaminu obsługi/ dla odcinka czasu symulowanego, w którym zostało przetworzone 3000 zadań.

Tabela 1

Liczba bloków programów symulacyjnych i czas pracy maszyny

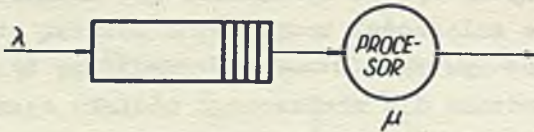
Regulamin	Liczba bloków	Czas pracy maszyny
FCFS	15	15 min.
RR	20	20 min.
PS	40	40 min.
FB_N	30	30 min.

Celem symulacji tych systemów było znalezienie odpowiedzi na pytanie, jaki jest wpływ liczby symulowanych zadań na do-

kładność symulacji systemów masowej obsługi w stosunku do wyników teoretycznych.

2. REGULAMIN PIERWSZE ZGŁOSZONE - PIERWSZE OBSŁUGIWANE /FCFS/

Regulamin FCFS jest najprostszym regulaminem z jedną kolejką oczekiwania /rys. 1/. Zadania wchodzące do systemu ustawiają się na końcu kolejki. Zadania pobierane są z początku kolejki do procesora, gdzie przebywają odcinek czasu konieczny do ich przetworzenia.



Rys. 1. System obsługi FIRST-COME-FIRST-SERVED

Średni czas oczekiwania zadania w kolejce do procesora jest wprost proporcjonalny do średniej liczby zadań w systemie i średniego czasu przetwarzania zadań. A więc średni czas przebywania zadania o czasie przetwarzania t w systemie wynosi [1], [4]:

$$T = E_r \cdot (1/\mu) + t$$

/5/

$$E_r = \frac{\rho}{1 - \rho}$$

gdzie: t - czas przetwarzania,

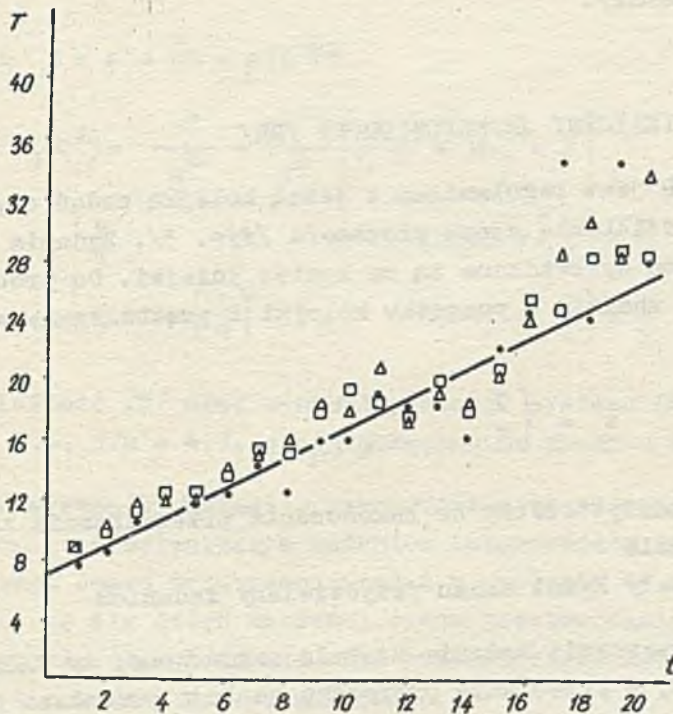
T - średni czas przebywania w systemie,

E_r - średnia liczba zadań w systemie.

Na rysunku 2 pokazano zależność /5/ oraz wyniku symulacji systemu dla: $1/\lambda = 7.4$ $1/\mu = 4.7$.

Regulamin FCFS posiada dwie zasadnicze wady:

- średni czas oczekiwania zadań na zwolnienie procesora jest jednakowy dla wszystkich zadań /zadania o krótkim czasie przetwarzania oczekują na procesor równie długo jak zadania o długim czasie przetwarzania/.
- zadania o długim czasie przetwarzania monopolizują procesor na długi odcinek czasu i nie dopuszczają innych zadań do procesora.



Rys. 2. Średni czas przebywania zadań w systemie FCFS w funkcji czasu przetwarzania

————— zależność /5/

- Wyniki symulacji:
- dla 1000 zadań
 - △ dla 2000 zadań
 - dla 3000 zadań

Zaletą tego regulaminu jest bardzo prosty algorytm wybierania zadań do przetworzenia oraz najmniejszy, dla opisanych regulaminów, stosunek czasu wymiany zadań w procesorze do efektywnego czasu przetwarzania. Regulamin ten może mieć zastosowanie w systemach przetwarzających strumień zadań o małej intensywności ruchu /mała średnia długość kolejki/ z rozkładami odstępów między zgłoszeniami|zadań i czasów przetwarzania o małym odchyleniu standardowym /małe prawdopodobieństwo zgłoszenia się w małych odstępach czasowych grupy zadań z długimi czasami przetwarzania/.

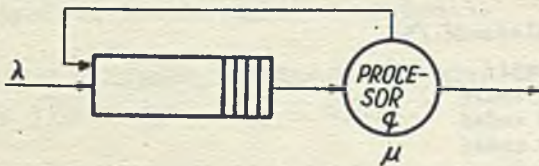
3. REGULAMIN CYKLICZNY JEDNOPOZIOMOWY /RR/

Regulamin RR jest regulaminem z jedną kolejką zadań oczekujących na przydzielenie czasu procesora /rys. 3/. Zadania wchodzące do systemu wprowadzane są na koniec kolejki. Do procesora pobierane jest zadanie z początku kolejki i przetwarzane w czasie t^* równym:

$$t^* = \begin{cases} q & \tau \geq q \\ \tau & \tau < q \end{cases} \quad /6/$$

gdzie: τ - czas potrzebny do zakończenia przetwarzania zadania
 q - stały kwant czasu przydzielany zadaniom

Jeżeli przetwarzanie zadania zostało zakończone, to zadanie opuszcza system, w przeciwnym przypadku zostaje przesłane na koniec kolejki oczekiwać do procesora.



Rys. 3. System obsługi ROUND - ROBIN

Średni czas przebywania zadania o czasie obsługi t w systemie wynosi [1]:

$$T = \frac{\rho k q}{1 - \rho} + \frac{(\lambda/2) \cdot E_1(\tau^2)}{1 - \beta} \cdot [1 - \rho \beta^{k-1}] +$$

$$+ \frac{1}{1 - \rho} \left[\frac{\rho^2}{1 - \rho} \cdot \frac{1}{\mu} - \frac{\rho q}{1 - \beta} \right] \cdot [1 - \beta^k] + \frac{\lambda q e^{-\mu q}}{1 - \beta} \cdot \frac{1}{\mu} [1 - \beta^{k-1}] + t$$

gdzie: $\beta = \rho + (1 - \rho)e^{-\mu q}$

$$E_1(\tau^2) = \frac{2}{\mu^2} - \frac{e^{-\mu q}}{\mu^2} [\mu^2 q^2 + 2\mu q + 2] \quad /7/$$

$$k = \begin{cases} \frac{t}{q} & \text{Entier} \left[\frac{t}{q} \right] = \frac{t}{q} \\ \text{Entier} \left[\frac{t}{q} \right] + 1 & \text{Entier} \left[\frac{t}{q} \right] \neq \frac{t}{q} \end{cases}$$

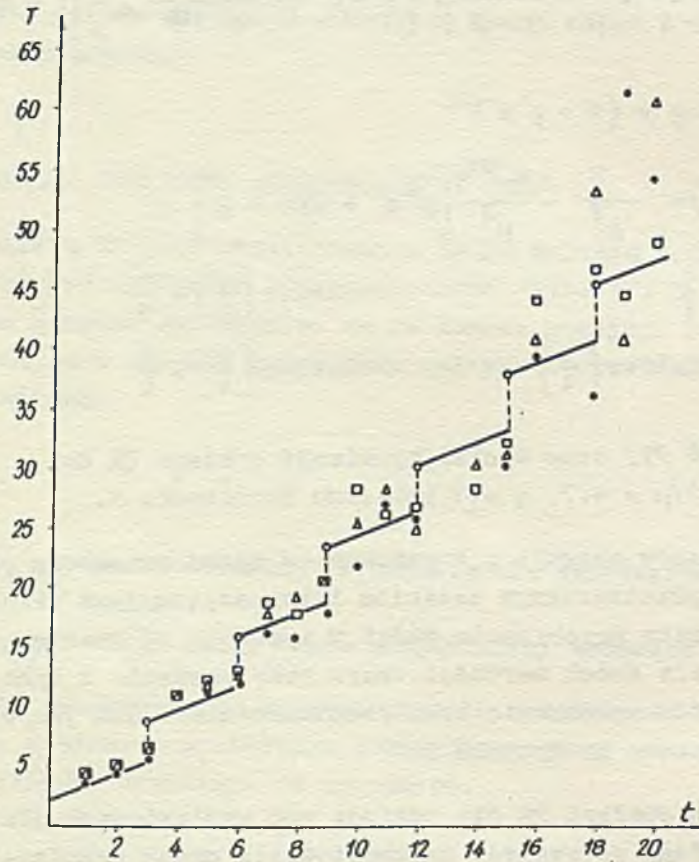
Zależność /7/ oraz wyniki symulacji systemu RR dla $1/\lambda = 7.4$, $1/\mu = 4.7$, $q = 3$ pokazano na rysunku 4.

Dla systemów obsługi z kwantowaniem czasu procesora przydzielonego przetwarzanym zadaniom interesująca jest zależność średniego czasu przebywania zadań w systemie od kwantu q . Zależność tę dla dwóch wartości czasu przetwarzania i tych samych rozkładów zgłoszenia oraz przetwarzania zadań jak na rysunku 4 pokazano na rysunku 5.

Regulamin obsługi RR nie posiada wad wymienionych dla systemu FCFS. Jednak ze względu na kwantowanie czasu przydzielonego zadaniom zależność /7/ nie jest ciągła. Dla dwóch zadań o prawie równych czasach przetwarzania t_1 i t_2 ale spełniających warunek:

$$t_1 \leq kq \quad t_2 > kq \quad k = 1, 2, 3, \dots \quad /8/$$

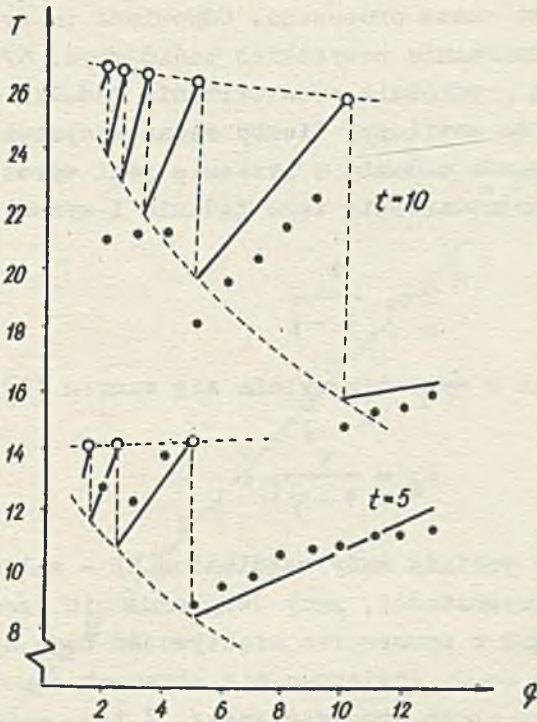
zadanie o czasie przetwarzania t_2 przechodzi cyklicznie przez kolejkę oczekiwań do procesora o jeden raz więcej niż zadanie o czasie przetwarzania t_1 .



Rys. 4. Średni czas przebywania zadań w systemie RR w funkcji czasu przetwarzania

————— zależność /7/

- Wyniki symulacji
- dla 1000 zadań
 - △ dla 2000 zadań
 - dla 3000 zadań



Rys. 5. Średni czas przebywania zadań w systemie RR w funkcji kwantu q dla $t = 5$ i $t = 10$

— zależność /7/
• wyniki symulacji dla 1000 zadań

Z tego względu regulamin RR można efektywnie stosować w systemach przetwarzających strumień zadań o dyskretnym rozkładzie czasów przetwarzania /takim, że czas przetwarzania każdego zadania jest całkowitą wielokrotnością kwantu q /.

Regulamin obsługi RR przekształca się w regulamin FCFS, gdy $q \rightarrow \infty$ /w tym przypadku każde zadanie otrzymuje jednorazowo tyle czasu procesora ile wymaga przetworzenie tego zadania/.

4. REGULAMIN Z PODZIAŁEM PROCESORA /PS/

Regulamin obsługi PS jest granicznym przypadkiem regulaminu RR, gdy $q \rightarrow 0$. Każde zadanie dla małych q szybko przebiega

cyklicznie przez kolejkę i procesor, i za każdym przebiegiem otrzymuje małą ilość czasu procesora. Odpowiada to sytuacji jednoczesnego przetwarzania wszystkich zadań /rys. 6/ z indywidualną chwilową prędkością przetwarzania zadania odwrotnie proporcjonalną do chwilowej liczby zadań w systemie. Stąd średni czas przebywania zadania w systemie jest wprost proporcjonalny do czasu przetwarzania tego zadania i wynosi [1], [4]:

$$T = \frac{t}{1 - \rho} \quad /9/$$

Średnia liczba zadań w systemie wyraża się wzorem [3]:

$$E_r = \frac{\rho}{1 - \rho} \quad /10/$$

Regulamin PS nie posiada wady regulaminu RR - zależność /9/ jest ciągła. W rzeczywistości, przy uwzględnieniu czasu traczonego na wymianę zadań w procesorze efektywność tego systemu dąży do zera, gdy $\rho \rightarrow 0$ /zwiększa się stosunek czasu wymiary zadań do efektywnego czasu przetwarzania/. Z tego względu regulamin PS może być stosowany w systemach z dużą pamięcią operacyjną /mieszczącą wszystkie aktualnie czynne zadania/ i z szybkim systemem obsługi przerw.

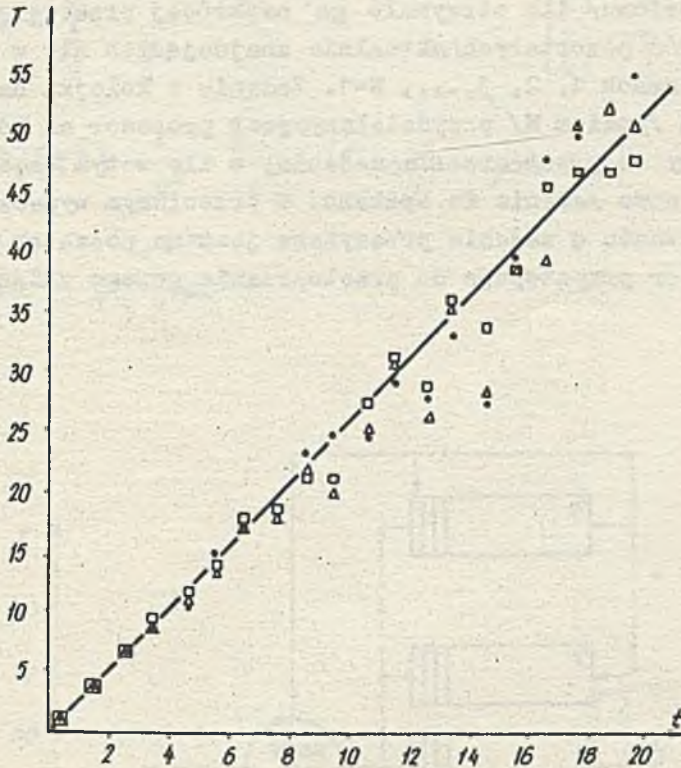


Rys. 6. System obsługi PROCESSOR - SHARED

Zależność /9/ oraz wyniki symulacji systemu PS dla $1/\lambda = 8.1$, $1/\mu = 5$ oraz $\rho = 0.01$, a więc $\rho \ll 1/\mu$ pokazano na rysunku 7.

5. REGULAMIN CYKLICZNY WIELOPOZIOMOWY /FB_N/

Regulamin obsługi FB_N [1], [2] jest regulaminem, w którym występuje N kolejek /poziomy 1, 2, ..., N/ /rys. 8/. Zadania



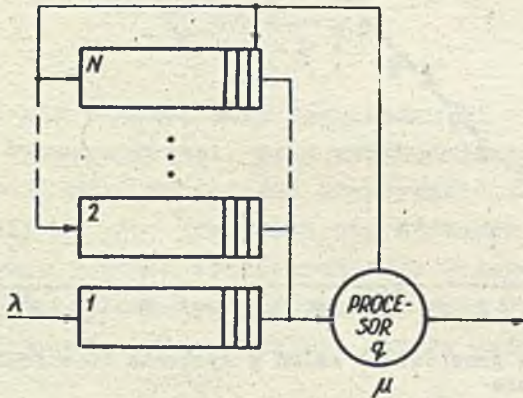
Rys. 7. Średni czas przebywania zadań w systemie PS w funkcji czasu przetwarzania

— zależność /10/

- Wyniki symulacji
- dla 1000 zadań
 - △ dla 2000 zadań
 - dla 3000 zadań

wchodzące do systemu wprowadzane są na kolejkę najniższego poziomu i mają wyższy priorytet w stosunku do zadań z pozostałych poziomów. Zadanie pobierane jest do przetworzenia z najniższej niepustej kolejki, przydzielany jest mu kwant czasu q /w ogólności różny dla różnych poziomów/ i zadanie jest odsyłane /o ile nie zostało zakończone przetwarzanie tego zadania/ na koniec kolejki poziomu wyższego o jeden. Innymi słowy zadaniu z kolejki pierwszej przydzielane jest jednorazowo najwyżej tyle kwantów czasu procesora /o ile w tym czasie nie zgłosiło

się nowe zadanie do systemu oraz nie ma innych zadań w kolejce pierwszego poziomu/ ile otrzymało go najkrócej przetwarzane zadanie spośród pozostałych aktualnie znajdujących się w systemie na poziomach 1, 2, 3, ..., N-1. Zadaniu z kolejki najwyższego poziomu /poziom N/ przydzielany jest procesor na odroinek czasu wymagany do przetworzenia zadania, o ile w tym czasie nie zgłosiło się nowe zadanie do systemu. W przeciwnym wypadku po zakończeniu kwantu q zadanie przesyłane jest na początek kolejki N i procesor przystępuje do przetwarzania nowego zadania.



Rys. 8. System obsługi FEED - BACK

Średni czas przebywania zadania o czasie przetwarzania t w systemie wynosi [1]:

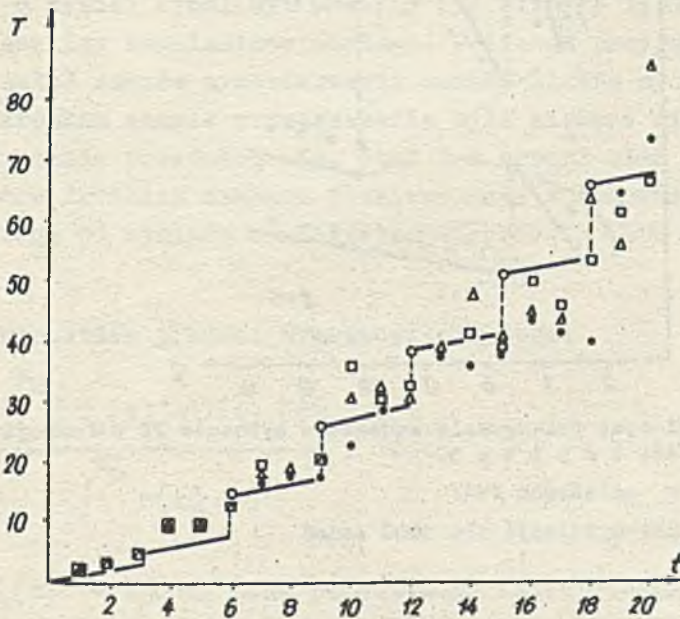
$$T = \begin{cases} \frac{(\lambda/2) [E_k(\tau^2) + \tau_k E_1(\tau^2)]}{[1 - \varphi(1 - e^{-\mu k q})] \cdot [1 - \varphi(1 - e^{-\mu(k-1)q})]} + \frac{\varphi(1 - e^{-\mu(k-1)q})}{1 - \varphi(1 - e^{-\mu(k-1)q})} \cdot (k-1)q + t & 1 \leq k \leq N-1 \\ \frac{\varphi(1/\mu)}{(1 - \varphi)[1 - \varphi(1 - e^{-\mu(N-1)q})]} + \frac{\varphi(1 - e^{-\mu(N-1)q})}{1 - \varphi(1 - e^{-\mu(N-1)q})} \cdot (k-1)q + t & k \geq N \end{cases} \quad /11/$$

gdzie:

$$k = \begin{cases} \text{Entier} \left[\frac{t}{q} \right] + 1 & \text{Entier} \left[\frac{t}{q} \right] \neq \frac{t}{q} \\ \frac{t}{q} & \text{Entier} \left[\frac{t}{q} \right] = \frac{t}{q} \end{cases}$$

$$E_k(\tau^2) = \frac{2}{\mu^2} - \frac{e^{-\mu k q}}{\mu^2} \left[(u k q)^2 + 2\mu k q + 2 \right]$$

$$\gamma_k = \frac{e^{-\mu k q}}{1 - e^{-\mu q}}$$



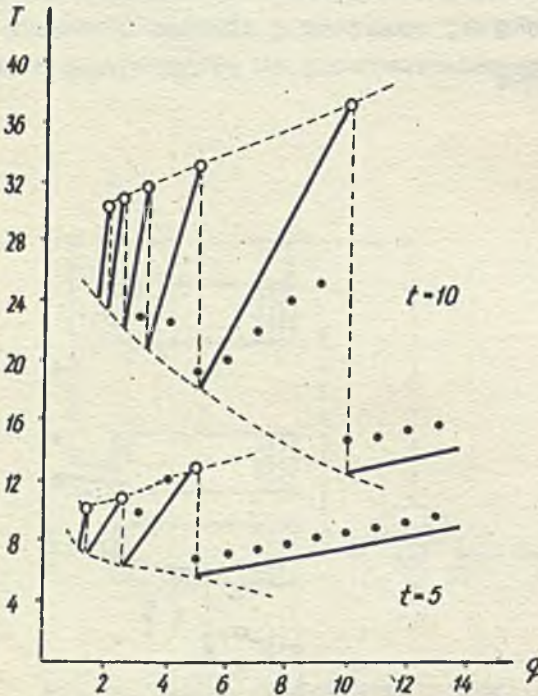
Rys. 9. Średni czas przebywania zadań w systemie FB w funkcji czasu przetwarzania

— zależność /11/

Wyniki symulacji

- dla 1000 zadań
- Δ dla 2000 zadań
- dla 3000 zadań

Zależność czasu przebywania zadania w systemie od czasu przetwarzania jest nieciągła, podobnie jak dla regulaminu RR. W regulaminie FB_N w jeszcze większym stopniu niż w RR uprzywilejowane są zadania o krótkim czasie przetwarzania. Regulamin ten można stosować w systemach przetwarzających strumień zadań zawierający długie ale niepriorytetowe zadania oraz krótkie ale wymagające natychmiastowej reakcji systemu.



Rys. 10. Średni czas przebywania zadania w systemie FB w funkcji kwantu q dla $t = 5$ i $t = 10$

— zaleźność /11/

. Wyniki symulacji dla 1000 zadań

Gdy $q \rightarrow 0$ /przy ograniczonym N /, regulamin FB_N przekształca się w regulamin FCFS. Dla małych q zadania wchodzące do systemu szybko przebiegają $N-1$ poziomów i ustawiają się w kolejce poziomu N , gdzie obsługiwane są zgodnie z regulaminem FCFS. Dla $q \rightarrow \infty$ zadania nigdy nie osiągają poziomu wyższego niż

pierwszy, a więc obsługiwane są również według regulaminu FCFS. Podobnie regulamin FB_1 jest także regulaminem typu FCFS.

Na rysunku 9 pokazano zależność /11/ oraz wyniki symulacji systemu dla $1/\lambda = 7.4$, $1/\mu = 4.7$, $q = 3$, $N = 7$, a na rys. 10 tę samą zależność w funkcji q dla czasów przetwarzania $t = 5$ i $t = 10$.

6. ANALIZA BŁĘDÓW SYMULACJI

Opisane systemy przetwarzania były symulowane w języku ZAM - GPSS [6] na maszynie ZAM 41. Została przeprowadzona analiza zależności błędów symulacji od liczby zadań przetwarzanych w symulowanych modelach. Im większa liczba przetworzonych zadań, tym wyniki symulacji powinny być bliższe wynikom teoretycznej analizy regulaminów obsługi. Ponieważ przyjęto wykładniczy rozkład czasów przetwarzania zadań, liczba symulowanych zadań o krótkim czasie przetwarzania była większa niż zadań o długim czasie przetwarzania. Stąd też średni czas przebywania zadań o krótkich czasach przetwarzania w systemie względnie mało odbiega od wyników analizy teoretycznej /patrz rys. 2, 4, 7, 9/.

Jako kryterium jakości symulacji przyjęto:

$$J = \frac{\int_0^{t_k} |T_S(t) - T(t)| dt}{\int_0^{t_k} T(t) dt} \cdot 100\% \quad /12/$$

gdzie: $T_S(t)$ - średni czas przebywania zadań o czasie przetwarzania t w symulowanym systemie,

$T(t)$ - średni czas przebywania zadań o czasie przetwarzania t w systemie /zależność teoretyczna/

$t_k = 4 \cdot \frac{1}{\mu}$ /dla rozkładu wykładniczego zadania o czasie przetwarzania $t \leq 4 \cdot \frac{1}{\mu}$ stanowią około 98% wszystkich zadań/.

W tabeli 2 pokazano wartości tak zdefiniowanego kryterium dla przebadanych regulaminów obsługi przy różnej liczbie przetwarzanych zadań oraz wartości średniego błędu bezwzględnego dla symulacji 3000 zadań zdefiniowanego następująco:

$$B = \frac{\int_0^{t_k} |T_B(t) - T(t)| dt}{t_k} \quad /13/$$

Tabela 2

Zestawienie błędów symulacji

Regulamin	Wartość kryterium J dla			Średni błąd bezwzględny B dla 3000 zadań
	1000 zadań	2000 zadań	3000 zadań	
FCFS	12%	11%	7%	1.4
RR	11%	9%	7%	1.8
PS	14%	8%	5%	1.5
FB _N	13%	16%	10%	2.8

Błędy symulacji wynikają z następujących przyczyn:

- skończony czas symulacji systemu
Wzory /5/, /7/, /9/ i /11/ określone zostały dla systemu pracującego w czasie $t \rightarrow \infty$,
- niedokładność modelowania wykładniczych rozkładów prawdopodobieństwa.
Generator ciągu liczb o rozkładzie wykładniczym był zbudowany za pomocą generatora liczb pseudolosowych o rozkładzie równomiernym i funkcji odwrotnej do dystrybuanty rozkładu wykładniczego. Wady generatora liczb pseudolosowych nakładały się z błędami modelowania dystrybuanty i powodowały zniekształcenie rozkładów zgłoszenia zdań i czasów przetwarzania.

Podane w tabeli 2 wartości kryterium dotyczą tylko systemów przetwarzających strumień zadań o podanych w poprzednich rozdziałach parametrach rozkładów odstępów między zgłoszeniami

zadań i czasów przetwarzania i mogą być jedynie bardzo ogólną wskazówką przy doborze liczby symulowanych w modelu zadań w zależności od wymaganej dokładności symulacji. Wartości tego kryterium są mniejsze dla strumieni zadań o mniejszej intensywności ruchu i dla rozkładów zgłoszenia zadań i czasów przetwarzania o mniejszym odchyleniu standardowym.

Wyniki jakie uzyskano z symulacji potwierdzają przydatność metod symulacyjnych do badania systemów przetwarzania informacji.

Literatura

- [1] COFFMAN E.G., KLEINROCK L.: Feedback Queueing Models for Time-Shared Systems, Journal of ACM, No 4, October 1968.
- [2] COFFMAN E.G.: Studying Multiprogramming Systems, Datamation, June 1967, 47-54.
- [3] ESTRIN G., KLEINROCK L.: Measures, Models and Measurements for Time-Shared Computer Utilities, Proceedings of 22nd National Conference, Association of Computing Machinery, 1967.
- [4] KLEINROCK L.: Time-Shared Systems: A Theoretical Treatment, Journal of ACM, No 2, April 1967.
- [5] WIĘCKOWSKI A.: Algorytmy szeregowania w systemach operacyjnych, Wybrane zagadnienia systemów operacyjnych, COPAN, Warszawa 1971.
- [6] PERKOWSKI P., HOJDA J.: Język ZAM-GPSS - Instrukcja programowania, Instytut Maszyn Matematycznych, czerwiec 1971.

МОДЕЛИРУЮЩЕЕ ИССЛЕДОВАНИЕ НЕКОТОРЫХ АЛГОРИТМОВ РАЗДЕЛЕНИЯ ВРЕМЕНИ ПРОЦЕССОРА В СИСТЕМАХ ОБРАБОТКИ ИНФОРМАЦИИ

Резюме:

Одной из проблем выступающих в системах обработки информации является выбор распоряжения обслуживания проблем, выполняющего поставленные требования, касающиеся чаще всего времени реакции системы и длины очереди проблем ожидающих обслуживания.

Принимаем следующие параметры источника проблем:

1. Экспоненциальное распределение промежутков между заявками проблем.
2. Экспоненциальное распределение времен обработки.

В работе описаны четыре из чаще встречаемых распоряжений обслуживания: /First-Come-First-Served, Round-Robin, Processor-Shared, Feed-Back/. Эти распоряжения были моделированы на машине ZAM 41 при помощи языка GPSS /General Purpose System Simulator/.

Размеры моделирующих программ выражены числом блоков/предложений/ языка GPSS и время работы машины для проведения одного эксперимента с моделью алгоритма /для 3000 проблем/ для отдельных распоряжений обслуживания составляли:

1. 15 блоков, 15 мин. для FCFS
2. 20 " , 20 " " RR
3. 40 " , 40 " " PS
4. 30 " , 30 " " FB_N

Целью моделирования этих распоряжений было найти соответствующий ответ на вопрос (какое влияние числа моделированных проблем на точность моделирования систем массового обслуживания по отношению к теоретическим результатам. Релятивная ошибка моделирования не превышала 15% для 1000 моделированных проблем, и соответственно 10% для 1000 проблем. Ошибки моделирования этих распоряжений являются результатом:

1. оконченного времени моделирования
2. неточности моделирования экспоненциальных распределений вероятности:
 - а) ошибок генератора псевдослучайных чисел с равномерным распределением,
 - в) ошибок моделирования дистрибуанта экспоненциального распределения

Результаты полученные в ходе моделирования подтверждают пригодность методов моделирования для исследования систем обработки информации.

MODELLING INVESTIGATION OF SOME PROCESSOR TIME-SHARING ALGORITHMS IN
INFORMATION PROCESSING SYSTEMS

Summary

One of the problems arising in processing systems consists in the choice of task service regulation; fulfilling the demands mainly concerning the time of the system reaction and the length of task lines waiting for service.

Let us assume the following parameters of the task source.

1. Exponential distribution of intervals between task notifications.
2. Exponential distribution of service time.

The article describes four, out of more frequently met service algorithms: First-Come-First-Served, Round-Robin, Processor-Shared, Feed-Back. These algorithms were modelled on ZAM 41 computer using the GPSS /General Purpose System Simulation/ language. The size of modelling programs expressed by the number of blocks /sentences/ of GPSS language and the time of machine operation used to carry out one experiment on the algorithm model /for 3000 tasks/ for separate service algorithms were:

1. 15 blocks, 15 min. for FCFS
2. 20 blocks, 20 min. for RR
3. 40 blocks, 40 min. for PS
4. 30 blocks, 30 min. for FB_N

The algorithm modelling aimed at finding the proper answer to the question what is the influence of modelled tasks number on the exactness of mass service system modelling in relation to theoretical results. Modelling error has not exceeded 15% for 1000 modelled tasks and 10% for 3000 tasks respectively.

These algorithm modelling inexactness results from:

1. Finished modelling time
2. Inexactness of exponential probability distribution modelling
 - a. inexactness of generator pseudo-random number with a uniform distribution
 - b. inexactness of modelling the distribuant of exponential distribution

The results obtained from modelling confirm its usefulness for information processing systems testing.

O PEWNEJ METODZIE SYNTEZY MINIMALNYCH
BEZHAZARDOWYCH KOMBINACYJNYCH UKŁADÓW
PRZEŁĄCZAJĄCYCH TANT

Andrzej HŁAWICZKA

Pracę złożono 5.04.1972

Praca zawiera opis metody syntezy minimalnych bez-
hazardowych kombinacyjnych trójwarstwowych układów
przełączających zbudowanych z elementów NAND oraz
podstawowe twierdzenia i definicje dotyczące iden-
tyfikacji i eliminacji hazardu statycznego i dyna-
micznego w takich układach przełączających. Pracę
zakończono licznymi przykładami oraz wnioskami.

SPIS TREŚCI

1. WSTĘP	26
2. PODSTAWOWE TWIERDZENIA I DEFINICJE DOTYCZĄCE IDENTYFIKACJI I ELI- MINACJI HAZARDU STATYCZNEGO I DYNAMICZNEGO W TRÓJWARSTWOWYCH UKŁADACH PRZEŁĄCZAJĄCYCH ZBUDOWANYCH Z ELEMENTÓW NAND	29
3. BEZHAZARDOWA SYNTEZA MINIMALNYCH UKŁADÓW TANT	34
4. PRZYKŁADY ZASTOSOWANIA ALGORYTMU SYNTEZY MINIMALNYCH UKŁADÓW TANT	35
5. WNIOSKI	40
Literatura	42
DODATEK 1	43
DODATEK 2	45

Wykaz podstawowych oznaczeń

F	- funkcja Boole'a
P	- implikant
Pp	- implikant prosty
Pd	- implikant dopuszczalny
Pdm	- maksymalny implikant dopuszczalny
Pdq	- quasi prosty implikant dopuszczalny
Pdg	- główny implikant dopuszczalny
Po	- quasi prosty implikant dominujący
PP	- prosty implikant dopuszczalny
E	- wyrażenie dopuszczalne
En	- nierozszerzone wyrażenie dopuszczalne
H	- początek
T	- czynnik końca
I	- zbiór implikantów dopuszczalnych
$\bigwedge_{a=1}^k x_a$	- konjunktja
$\bigvee_{a=1}^k x_a$	- alternatywa

1. WSTĘP

Na wyjściach kombinacyjnych układów przełączających, gdy ich struktura nie jest zaprojektowana z myślą o niezawodnym działaniu, mogą występować przekłamania przetwarzanej informacji niezgodne z algorytmem działania zaprojektowanego układu przełączającego. Przekłamania te mają charakter przejściowy w kombinacyjnych układach przełączających i wynikają ze struktury sprawnego technicznie układu oraz z rzeczywistych charakterystyk elementów przełączających. O układach kombinacyjnych generujących takie przekłamania mówi się, że posiadają hazard, który w zależności od formy przekłamań w jakiej pojawiają się one na wyjściu układu, nazywa się hazardem dynamicznym lub hazardem statycznym.

Wspomniane wyżej przejściowe przekłamania mogą zostać utrwalone w elementach pamięci asynchronicznego sekwencyjnego automatu, gdy kombinacyjny układ przełączający posiadający hazard jest

częścią tego asynchronicznego automatu. Jeżeli przekłamania stale się utrwalają, to hazard w takim przypadku nazywany jest szkodliwym [10]. Huffman [4], a następnie Unger [8] wykazali, że prawidłowa praca asynchronicznych automatów sekwencyjnych wymaga między innymi, aby ich część kombinacyjna nie posiadała szkodliwego hazardu.

W niniejszej pracy celem uproszczenia dalszych rozważań przyjęto za autorami prac specjalistycznych [3], [5] i [10] zasadę, że przy projektowaniu asynchronicznych automatów sekwencyjnych będzie się dążyć do likwidacji hazardów statycznych i dynamicznych, zarówno szkodliwych jak również nieszkodliwych /pozornych [10]/, które nie wpływają na poprawną pracę asynchronicznych automatów sekwencyjnych.

Ogromny postęp w ostatnim czasie w dziedzinie technologii wytwarzania układów elektronicznych przejawiał się między innymi w postaci miniaturyzacji elementów, zwiększeniu ich niezawodności oraz obniżeniu ich ceny. To ostatnie zjawisko spowodowało, że w układach przełączających, budowanych z elementów scalonych, koszt wykonania połączenia między elementami jest porównywalny z kosztem samego elementu. W związku z powyższym projektant układów przełączających powinien dysponować również takimi metodami syntezy bezhazardowych układów kombinacyjnych, które uwzględniałyby oprócz kryterium kosztów elementów równocześnie kryterium kosztów wejść na elementy, czyli pozwalałyby wybierać najtańsze układy przełączające spośród wszystkich możliwych bezhazardowych układów przełączających, realizujących tę samą funkcję Boole'a. Metody syntezy oprócz wspomnianych wyżej wymagań powinny między innymi spełniać jeszcze następujące [9]:

- możliwość syntezy układów przełączających realizujących funkcje n -zmiennych
- możliwość wyboru kryterium optymalności układu np. najmniejszy czas propagacji układu
- przydatność dla szerokiej klasy elementów

- uwzględnienie ograniczeń właściwych realnym elementom i ograniczeń związanych z ich montażem w realnym układzie
- możliwość uwzględniania dodatkowych wymagań nakładanych na projektowany układ przełączający np. łatwość naprawiania
- zapewnienie otrzymania układów optymalnych lub bliskich optymalnym.

Opracowanie metody syntezy spełniającej te i poprzednie wymagania nie jest rzeczą łatwą, tym bardziej, że spełnienie jednych wymagań wyklucza możliwość spełnienia innych.

Na przykład dążenie do uzyskania jak najmniejszych czasów propagacji układów przełączających zmusza do zmniejszenia liczby warstw układu przełączającego do niezbędnego minimum, pozwalającego jeszcze realizować dowolne funkcje n -zmiennych. Staje się to automatycznie ograniczeniem dla kryterium kosztów, gdyż liczba możliwych do porównania układów przełączających zostaje w ten sposób ograniczona tylko do układów o niezbędnej liczbie warstw. Jeśli jednak ograniczy się zastosowanie metody syntezy kombinacyjnych układów przełączających do funkcjonalnie pełnego systemu w postaci elementu NAND [9], to sprawa poważnie się upraszcza i bardziej realne staje się stworzenie metody spełniającej większość wymienionych wymagań.

Gimpel w swojej pracy [2] przedstawił prostą metodę syntezy minimalnych, trójwarstwowych kombinacyjnych układów przełączających zbudowanych z elementów NAND i zasilanych wyłącznie zmiennymi nieznanymi. Układy takie nazywane są układami TANT, a ich wyrażenia strukturalne - wyrażeniami TANT. Gimpel udowodnił, że układ TANT jest absolutnie minimalny ze względu na kryterium kosztów elementów i kryterium kosztów wszystkich wejść na elementy, jeżeli jego syntezę przeprowadza się na podstawie zbioru dopuszczalnych niższych implikantów /D1/. Ponieważ korzystanie z tego zbioru powoduje jednak, że synteza absolutnie minimalnych układów TANT jest bardzo uciążliwa, Gimpel równolegle udowodnił, że przeprowadzenie syntezy układów TANT opierając się na zbiorze głównych dopuszczalnych implikantów /D1/ jest o wiele mniej pracochłonne, a jednocześnie pozwala

uzyskiwać układy TANT o absolutnie minimalnej liczbie elementów i liczbie wszystkich wejść na elementy niewiele większej od liczby uzyskiwanej w pierwszym przypadku. Metoda Gimpela nie pozwala jednak na projektowanie minimalnych bezhazardowych układów TANT. Dla dwuwarstwowych kombinacyjnych układów przełączających LUB-I oraz I-LUB metodę bezhazardowej ich syntezy przedstawił McCluskey [3] opierając się na pracy Huffmana [4]. Natomiast metodę bezhazardowej syntezy minimalnych trójwarstwowych układów przełączających zbudowanych z elementów NOR przedstawiono w pracy [1]. Na podstawie tej pracy i na bazie metody Gimpela w rozdziałach następnym przedstawiono metodę syntezy minimalnych bezhazardowych układów TANT realizujących funkcje n-zmiennych.

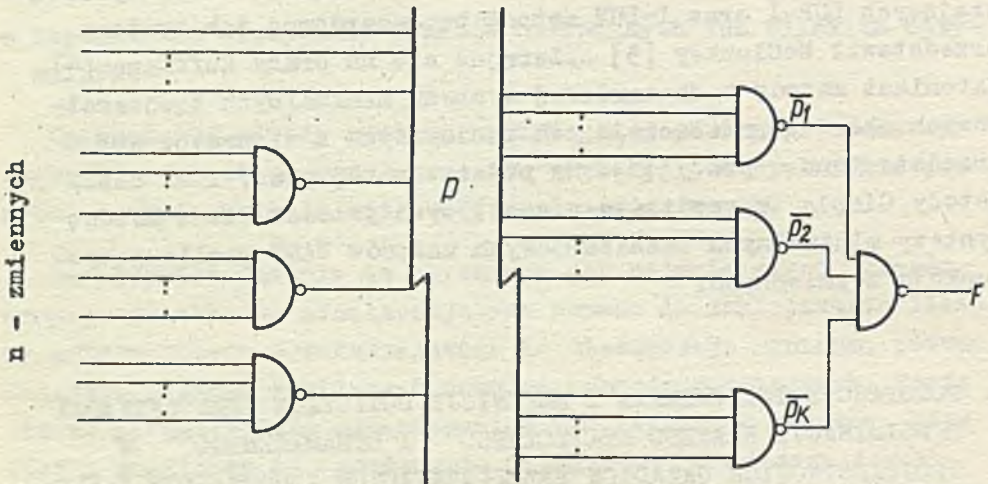
2. PODSTAWOWE TWIERDZENIA I DEFINICJE DOTYCZĄCE IDENTYFIKACJI I ELIMINACJI HAZARDU STATYCZNEGO^κ I DYNAMICZNEGO^{κκ} W TRÓJWARSTWOWYCH UKŁADACH PRZEŁĄCZAJĄCYCH ZBUDOWANYCH Z ELEMENTÓW NAND

W trójwarstwowych układach przełączających zbudowanych z elementów NAND zarówno problem ich bezhazardowej syntezy, czy też identyfikacji w nich HS lub HD komplikuje się poważnie. Siatka Karnaugh lub tablica implikantów w większości wypadków nie jest odbiciem struktury tych układów, a tylko wyłączenie obrazem funkcji realizowanej przez te układy. W związku z tym wybór najtańszego bezhazardowego trójwarstwowego układu przełączającego zbudowanego z elementów NAND za pomocą siatki Karnaugh, czy też tablicy implikantów nie daje rezultatów. Zauważmy jednak, że taki układ przełączający, którego ogólną postać przedstawiono na rys. 1 i który zapisać można w następującej postaci:

$$F = \bigwedge_{a=1}^k \bar{P}_a$$

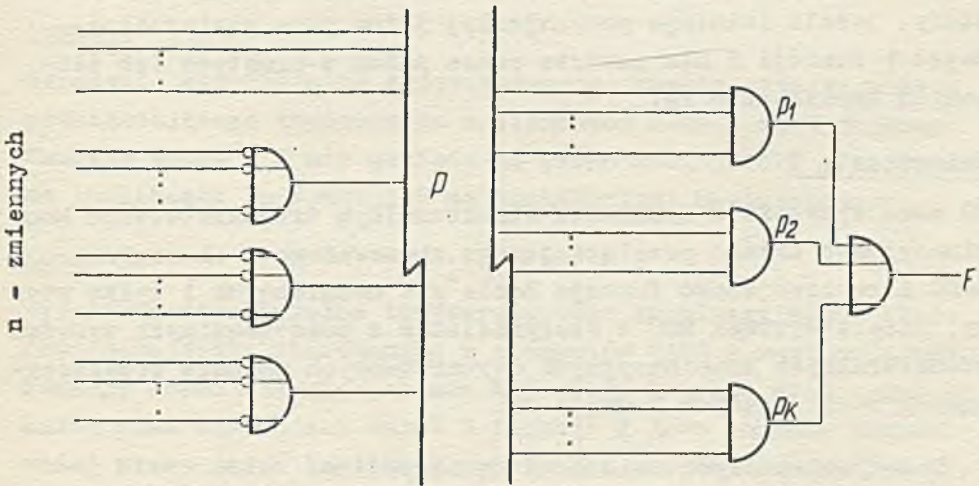
κ) Hazard statyczny w dalszej części pracy oznaczono przez HS

κκ) Hazard dynamiczny w dalszej części pracy oznaczono przez HD



Rys. 1. Ogólna postać trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND, P przedstawia pewien dowolny zbiór połączeń wewnętrznych

można przekształcić, wg zasad podanych w pracach [6] i [11], w układ z rys. 2, przedstawiający równoważny układ przełączający. Alternatywa $\bigvee_{a=1}^k P_a$ jest wyrażeniem strukturalnym powyższego układu, gdzie funkcje P_a są wyrażeniami strukturalnymi dwuwarstwowych układów przełączających. Alternatywę tę można przedstawić za pomocą sumy siatek Karnaugh'a [1], gdzie każda siatka jest obrazem funkcji P_a i równocześnie jest obrazem struktury dwuwarstwowego układu przełączającego realizującego tę funkcję. Tak więc cała suma siatek Karnaugh'a jest obrazem funkcji F oraz jest odzwierciedleniem struktury układu przełączającego realizującego funkcję F . Zsumowanie tych siatek pozwala uzyskać siatkę wypadkową z zaznaczonymi implikantami złożonymi lub prostymi, które są obrazem wejść jedynkowych poszczególnych siatek /funkcji P_a /.



Rys. 2. Zmodyfikowana postać trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND. P przedstawia pewien dowolny zbiór połączeń wewnętrznych

Na podstawie pracy autora [1] można przedstawić następujące twierdzenia i definicje dotyczące istnienia HS i HD w trójwarstwowym kombinacyjnym układach przełączających zbudowanych z elementów NAND.

Twierdzenie 1

HSO może wystąpić w wyrażeniu strukturalnym trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcje Boole'a F n -zmiennych wtedy i tylko wtedy, jeżeli wystąpi HSO w poszczególnych wyrażeniach strukturalnych kombinacyjnych dwuwarstwowym układów przełączających realizujących funkcje P_a .

Twierdzenie 2

HS1 wystąpi w wyrażeniu strukturalnym trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcję Boole'a F n -zmiennych wtedy i tylko wtedy, jeżeli istnieje przynajmniej jedna para sąsiednich wejść 1 funkcji F nie pokryta przez żaden z prostych lub złożonych implikantów P_a .

Twierdzenie 3

HD może wystąpić w wyrażeniu strukturalnym trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcję Boole'a F n -zmiennych i tylko wtedy, jeżeli wystąpi HSO w którymkolwiek z poszczególnych wyrażen strukturalnych kombinacyjnych dwuwarstwowych układów przełączających realizujących funkcję P_a .

Dowody powyższych twierdzeń są oczywiste.

Definicja 1

Bezszkodowym implikantem P_a funkcji Boole'a F jest każda funkcja, która zawiera się w F i która jest przedstawiona w postaci takiego iloczynu prostych implikantów funkcji P_a , że pokrywają one wszystkie pary sąsiednich wejść 0 funkcji P_a .

Odzwierciedleniem przedstawionej definicji są trzy poniższe siatki Karnaugh'a, z których pierwsza przedstawia funkcję F , a dwie pozostałe - postać hazardową oraz postać bezhazardową tego samego implikantu P_a funkcji F .

wx	yz	00	01	11	10
00	0	1	0	0	
01	0	1	1	0	
11	1	1	0	0	
10	0	1	0	0	

wx	yz	00	01	11	10
00	0	1	0	0	
01	0	1	1	0	
11	0	1	0	0	
10	0	1	0	0	

wx	yz	00	01	11	10
00	0	1	0	0	
01	0	1	1	0	
11	0	1	0	0	
10	0	1	0	0	

$F = \bar{w}x + x\bar{y}z + \bar{w}yz$

$P_a = x \cdot (\bar{w} + \bar{y}) (\bar{w} + \bar{z})$
postać bezhazardowa

$P_a = (w+x)(\bar{w}+y+z)(w+y)(\bar{w}+x+y)$
postać hazardowa

Z przedstawionych twierdzeń oraz wprowadzonej definicji wynikają poniższe trzy lematy, które wyznaczają zasady postępowania przy bezhazardowej syntezie trójwarstwowych kombinacyjnych układów przełączających zbudowanych z elementów NAND.

Lemat 1

Wyrażenie strukturalne trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcję Boole'a F nie posiada na pewno HSO, jeżeli poszczególne implikanty P_a funkcji F są implikantami bezhazardowymi.

Lemat 2

Wyrażenie strukturalne trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcję Boole'a F nie posiada HS1 wtedy i tylko wtedy, jeżeli każda para sąsiednich wejść 1 funkcji F jest pokryta przynajmniej przez jeden implikant prosty lub złożony P_a funkcji F .

Lemat 3

Wyrażenie strukturalne trójwarstwowego kombinacyjnego układu przełączającego zbudowanego z elementów NAND i realizującego funkcję Boole'a F nie posiada na pewno HD, jeżeli poszczególne implikanty P_a funkcji F są implikantami bezhazardowymi.

Powyższe lematy i definicja 1 pozwalają sprecyzować twierdzenie finalne obejmujące całość problemu bezhazardowej syntezy trójwarstwowych kombinacyjnych układów przełączających zbudowanych z elementów NAND. Brzmi ono następująco:

Twierdzenie 4

Trójwarstwowy kombinacyjny układ przełączający zbudowany z elementów NAND i realizujący funkcję Boole'a F o n -zmiennych jest na pewno bezhazardowym układem przełączającym, jeżeli każda para sąsiednich wejść jedynkowych funkcji F jest pokryta przynajmniej przez jeden bezhazardowy implikant P_a funkcji F .

3. BEZHAZARDOWA SYNTEZA MINIMALNYCH UKŁADÓW TANT

Układy TANT są szczególnym przypadkiem trójwarstwowych kombinacyjnych układów przełączających zbudowanych z elementów NAND i w związku z tym przedstawione poprzednio twierdzenie 4 dotyczy również bezhazardowej syntezy tych układów.

Synteza układów TANT^{**} polega na szukaniu takich podzbiorów zbioru I dopuszczalnych implikantów Pd funkcji Boole'a F, które tworzą niezbędne pokrycia funkcji F. W związku z tym, jeżeli zbiór I będzie zbiorem bezhazardowych dopuszczalnych implikantów Pd, to każde wyrażenie dopuszczalne E implikantu Pd z każdego podzbioru tworzącego niezbędne pokrycie funkcji F nie będzie posiadało HSO.

Twierdzenie 5

Implikant dopuszczalny Pd przedstawiony w postaci nierozszerzonego wyrażenia dopuszczalnego En jest bezhazardowym implikantem dopuszczalnym.

Dowód

Założmy, że istnieje rozszerzone wyrażenie dopuszczalne E implikantu dopuszczalnego Pd. Ponieważ wyrażenie E posiada ozniki końca T powiększone o zanegowanie zmienne początku H, więc tym samym spełnia 1 warunek twierdzenia McCluskey^{***}.

Gimpel udowodnił [2], że układ TANT realizujący funkcję Boole'a F ma postać minimalną, jeżeli każdy implikant dopuszczalny Pd wyrażenia TANT jest głównym implikantem dopuszczalnym Pd_g. Ponieważ każdy główny implikant dopuszczalny w metodzie Gimpela jest generowany w postaci nierozszerzonego wyrażenia dopuszczalnego, więc zupełnie oczywiste jest stwierdzenie, że każdy taki główny implikant Pd_g jest implikantem bezhazardowym.

^{**}) Dodatek 1 dołączony do tej pracy przedstawia podstawowe definicje i twierdzenia metody Gimpela, syntezy minimalnych układów TANT

^{***}) Dodatek 2 przedstawia twierdzenie McCluskey o warunkach istnienia HSO i H₁ w dwuwarstwowych układach logicznych I-LUB oraz LUB-I

W związku z powyższym oczywiście jest również twierdzenie 6.

Twierdzenie 6

Dla dowolnej funkcji Boole'a F wśród realizujących ją bezhazardowych wyrażeń TANT istnieje wyrażenie minimalne takie, że każda para sąsiednich wejść jedynkowych funkcji F jest pokryta przynajmniej przez jeden bezhazardowy główny implikant dopuszczalny Pdg funkcji F .

Algorytm bezhazardowej syntezy minimalnych układów TANT realizujących funkcje F o n -zmiennych:

- 1/ generujemy metodą Gimpla zbiór wszystkich możliwych głównych implikantów Pdg funkcji F
- 2/ tworzymy tabelę głównych implikantów funkcji F , w której kolumny przypisane są poszczególnym parom sąsiednich wejść jedynkowych funkcji F .
- 3/ znajdujemy minimalne wyrażenie strukturalne bezhazardowego układu TANT, stosując znane techniki redukcyjne tabeli implikantów [3].

4. PRZYKŁADY ZASTOSOWANIA ALGORYTMU SYNTEZY MINIMALNYCH UKŁADÓW TANT

Przykład 1

Mamy funkcje o następującej postaci normalnej dysjunkcyjnej

$$F = \bar{w}x + wx\bar{y} + \bar{w}x\bar{y} + y\bar{z}$$

Obraz tej funkcji przedstawiono w poniższej siatce Karnaugh, w której numeryczny zapis dziesiętny odzwierciedla wejścia jedynkowe.

		xz			
		00	01	11	10
yz	00		4	12	
	01		5	13	
	11	3	7		
	10	2	6	14	10

Zastosujemy omówiony wyżej algorytm.

ad 1/ Wygenerowane metodą Gimpla główne implikanty dopuszczalne funkcji F przedstawione są poniżej

$$\text{Pdg1} = y(\bar{w} + \bar{z})$$

$$\text{Pdg2} = y\bar{w}$$

$$\text{Pdg3} = y\bar{z}$$

$$\text{Pdg4} = x(\bar{z} + \bar{y} + \bar{w})$$

$$\text{Pdg5} = x(\bar{z} + \bar{y})$$

$$\text{Pdg6} = x(\bar{y} + \bar{w})$$

$$\text{Pdg7} = x(\bar{z} + \bar{w})$$

$$\text{Pdg8} = x\bar{z}$$

$$\text{Pdg9} = x\bar{y}$$

$$\text{Pdg10} = x\bar{w}$$

$$\text{Pdg11} = xy(\bar{w} + \bar{z})$$

$$\text{Pdg12} = wx(\bar{z} + \bar{y})$$

$$\text{Pdg13} = xz(\bar{y} + \bar{w})$$

ad 2/ Tworzymy tabelę głównych implikantów funkcji F, w której kolumny przypisane są poszczególnym parom sąsiednich wejść jedynkowych zapisanych w postaci dziesiętnej, a wiersze przypisane są poszczególnym głównym implikantom, przy czym kolejność wpisanych od góry w wiersze głównych implikantów uzależniona jest w pierwszym rzędzie od jego kosztów elementów, a następnie od kosztów połączeń, co uwidoczniło w skrajnej prawej kolumnie /tabela 1/.

ad 3/ Stosując znane techniki redukcyjne tabel implikantów /3/ znajdujemy następujące minimalne bezhazardowe wyrażenie TANT funkcji F.

$$F = y(\bar{w} + \bar{z}) + x(\bar{z} + \bar{y} + \bar{w})$$

Jak widać można je zrealizować za pomocą 5 elementów NAND oraz 11 połączeń.

Zapisując funkcję F w postaci alternatywy jej prostych implikantów pokrywających wszystkie pary sąsiednich

wejsc 1 funkcji F otrzymujemy równiez wyrażenie strukturalne bezhazardowego układu TANT

$$F = \bar{w}y + x\bar{z} + \bar{w}x + y\bar{z} + \bar{y}x$$

Ale w tym przypadku tak otrzymane wyrażenie TANT realizuje się za pomocą 9 elementów NAND oraz 18 połączeń. Czyli stosując w tym przykładzie opisany algorytm uzyskaliśmy oszczędność 4 elementów NAND oraz 7 połączeń.

Tabela 1

Główne implikanty funkcji F		2	2	2	3	4	4	4	5	5	6	6	10	12	12	Koszt	
		3	6	10	7	5	6	12	7	13	7	14	14	13	14	el.	poł.
Pdg 2	$y\bar{w}$	X	X		X						X					2	4
Pdg 3	$y\bar{z}$		X	X								X	X			2	4
Pdg 8	$x\bar{z}$					X	X				X			X		2	4
Pdg 9	$x\bar{y}$					X		X	X					X		2	4
Pdg 10	$x\bar{w}$					X	X		X	X						2	4
*	Pdg 1	$y(\bar{w}+\bar{z})$	X	X	X	X					X	X	X			2	5
	Pdg 5	$x(\bar{z}+\bar{y})$				X	X	X		X	X		X	X		2	5
	Pdg 7	$x(\bar{z}+\bar{w})$				X	X	X	X		X	X			X	2	5
	Pdg 6	$x(\bar{w}+\bar{y})$				X	X	X	X	X				X		2	5
*	Pdg 4	$x(\bar{z}+\bar{y}+\bar{w})$				X	X	X	X	X	X	X		X	X	2	6
	Pdg 11	$xy(\bar{w}+\bar{z})$									X	X				2	6
	Pdg 12	$wx(\bar{z}+\bar{y})$											X	X		2	6
	Pdg 13	$xz(\bar{y}+\bar{w})$						X	X							2	6

Przykład 2

Mamy $F = \bar{w}x\bar{y}z + \bar{w}xy + xy\bar{z} + x\bar{y}z + \bar{y}\bar{z}$

Stosujemy przedstawiony algorytm.

ad 1/ Wygenerowane główne implikanty dopuszczalne funkcji F

Pdg1 = $\bar{y}(\bar{w} + \bar{z})$

Pdg2 = $\bar{y}\bar{w}$

Pdg3 = $\bar{y}\bar{z}$

Pdg7 = $x(\bar{w} + \bar{z})$

Pdg8 = $x\bar{y}$

Pdg9 = $x\bar{w}$

$$\text{Pd}g4 = x(\bar{y} + \bar{w} + \bar{z})$$

$$\text{Pd}g5 = x(\bar{y} + \bar{w})$$

$$\text{Pd}g6 = x(\bar{y} + \bar{z})$$

$$\text{Pd}g10 = x\bar{z}$$

$$\text{Pd}g11 = xz(\bar{w} + \bar{y})$$

$$\text{Pd}g12 = xy(\bar{w} + \bar{z})$$

$$\text{Pd}g13 = wx(\bar{y} + \bar{z})$$

ad 2/ Tworzymy tabelę głównych implikantów /Tabela 2/.

Tabela 2

Główne implikanty funkcji F		0	0	0	1	4	4	4	5	5	6	6	B	12	12	Koszt
		1	4	8	5	5	6	12	7	13	7	14	12	13	14	el. pól
Pdg 8	$x\bar{y}$				X		X		X				X			2 4
Pdg 9	$x\bar{w}$				X	X		X		X						2 4
Pdg 10	$x\bar{z}$					X	X					X		X		2 4
Pdg 5	$x(\bar{y} + \bar{w})$				X	X	X	X	X	X			X			2 5
Pdg 6	$x(\bar{y} + \bar{z})$				X	X	X		X		X	X	X	X		2 5
Pdg 7	$x(\bar{w} + \bar{z})$				X	X	X	X		X	X			X		2 5
* Pdg 4	$x(\bar{y} + \bar{w} + \bar{z})$				X	X	X	X	X	X	X	X	X	X	X	2 6
Pdg 11	$xz(\bar{w} + \bar{y})$								X	X						2 6
Pdg 12	$xy(\bar{w} + \bar{z})$										X	X				2 6
Pdg 13	$wy(\bar{y} + \bar{z})$												X	X		2 6
Pdg 2	$\bar{y}\bar{w}$	X	X		X	X										3 5
Pdg 3	$\bar{y}\bar{z}$		X	X			X					X				3 5
* Pdg 1	$\bar{y}(\bar{w} + \bar{z})$	X	X	X	X	X		X	X			X	X			3 6

ad 3/ Po zastosowaniu techniki redukcyjnej uzyskujemy dwa minimalne wyrażenia TANT

$$F = \bar{y}(\bar{w} + \bar{z}) + w(\bar{y} + \bar{w} + \bar{z}) \quad 6 \text{ elementów } 12 \text{ połączeń}$$

$$y = x(\bar{w} + \bar{z}) + y(\bar{w} + \bar{z}) + x\bar{y} \quad 6 \text{ elementów } 12 \text{ połączeń}$$

Natomiast przedstawiona niżej alternatywa prostych implikantów funkcji F pokrywających wszystkie pary jej sąsiednich wejść 1

$$F = \bar{w}\bar{y} + \bar{y}x + x\bar{w} + \bar{y}\bar{z} + \bar{z}x$$

realizuje układ TANT za pomocą 9 elementów i 18 połączeń.

Przykład 3

Mamy $F = \bar{w}\bar{x}\bar{y} + w\bar{x}\bar{y}\bar{z} + w\bar{x}z$

Stosujemy przedstawiony algorytm

ad 1/ Wygenerowane główne implikanty dopuszczalne funkcji F

$$Pd g 1 = \bar{x}\bar{y}(\bar{w} + \bar{z})$$

$$Pd g 4 = w\bar{z}(\bar{y} + \bar{x})$$

$$Pd g 2 = \bar{x}\bar{y}w$$

$$Pd g 5 = w\bar{z}\bar{y}$$

$$Pd g 3 = \bar{x}\bar{y}\bar{z}$$

$$Pd g 6 = w\bar{z}\bar{x}$$

ad 2/ Tworzymy tabelę głównych implikantów

Tabela 3

Główne implikanty funkcji F		0	0	1	1	Koszt
		1	0	0	1	el./pot
Pdg 5	wz \bar{y}				X	3/6
Pdg 6	wz \bar{x}			X		3/6
Pdg 4	w \bar{z} ($\bar{y} + \bar{x}$)			X	X	3/7
Pdg 2	$\bar{x}\bar{y}w$	X				4/7
Pdg 3	$\bar{x}\bar{y}\bar{z}$		X			4/7
Pdg 1	$\bar{x}\bar{y}(\bar{w} + \bar{z})$	X	X			4/8

ad 3/ Po zastosowaniu techniki redukcyjnej w tabeli uzyskujemy następujące minimalne wyrażenie TANT

$$F = \bar{x}\bar{y}(\bar{w} + \bar{z}) + w\bar{z}(\bar{y} + \bar{x}) \quad \begin{array}{l} 8 \text{ elementów} \\ 15 \text{ połączeń} \end{array}$$

Natomiast normalna dysjunkcyjna postać skrócona funkcji F pokrywająca wszystkie pary jej sąsiednich wejść 1

$$F = \bar{w}\bar{x}\bar{y} + w\bar{y}\bar{z} + \bar{x}\bar{y}\bar{z} + w\bar{z}\bar{x}$$

realizuje układ TANT za pomocą 9 elementów oraz 20 połączeń.

5. WNIOSKI

Podsumowując wyniki uzyskane za pomocą przedstawionej metody należy stwierdzić, że większość wymagań opisanych we wstępie została zrealizowana. Uzyskane w przykładach, w wyniku zastosowania algorytmu, wyrażenia TANT są wyrażeniami minimalnymi spośród wszystkich możliwych bezhazardowych wyrażań TANT. Metoda nadaje się w zupełności do bezhazardowej syntezy funkcji n -zmiennych i w tym celu można opierając się na przedstawionym algorytmie napisać program dla maszyny cyfrowej.

Zupełnie dobre rezultaty przy wykorzystaniu maszyny IBM 7094 uzyskał Gimpel. Czas potrzebny do obliczenia przez maszynę 80 minimalnych wyrażań TANT funkcji 3-zmiennych, na podstawie programu napisanego przy wykorzystaniu jego algorytmu syntezy, wynosił 59 sekund. Biorąc pod uwagę, że metoda przedstawiona w tej pracy jest mniej skomplikowana, ponieważ wymaga wykonania mniejszej liczby operacji przy rozwiązaniu tabeli implikantów, niż metoda Gimpla przy rozwiązaniu tabeli CC, należy przypuszczać, że czas obliczeniowy maszyny powinien być krótszy.

Przedstawione poprzednio przykłady dotyczą bezhazardowej syntezy minimalnych układów TANT realizujących funkcje 4-zmiennych. Przeprowadzanie ręczne przedstawionej metody syntezy dla funkcji o takiej liczbie zmiennych było bardzo pracochłonne. Dla funkcji 5 i 6 ta pracochłonność dodatkowo wzrośnie. Dlatego interesująca byłaby możliwość uzyskiwania dla funkcji o tej liczbie zmiennych krótszą i prostszą drogą takich samych rezultatów lub przynajmniej rezultatów zbliżonych.

Zwróćmy uwagę, że prawie do każdego bezhazardowego wyrażenia TANT przedstawionego za pomocą dysjunkcyjnej normalnej postaci skróconej można zastosować prawo rozdzielności mnożenia względem dodawania, bez naruszania zasad zapewniających bezhazardowość wyrażenia przekształcanego [3]. Jeżeli w procesie przekształcania narzucimy dodatkowy warunek, że w nawiasie mogą się znaleźć tylko zmienne zanegowane, to w wyniku przekształcania będzie można uzyskiwać wyłącznie wyrażenia dopusz-

czalne w postaci bądź to quasi prostych implikantów dopuszczalnych, czyli takich implikantów, które uzyskiwane są w wyrażeniach minimalnych metodą przedstawioną w tej pracy. Prześledźmy to na poprzednio przedstawionych przykładach.

Przykład 1

$F = \bar{w}y + x\bar{z} + \bar{w}x + y\bar{z} + \bar{y}x$ - dysjunkcyjna normalna postać skrócona. Po zastosowaniu prawa rozdzielności mnożenia względem dodawania z zachowaniem narzuconego warunku uzyskujemy wyrażenie następujące:

$$F = y(\bar{w} + \bar{z}) + x(\bar{z} + \bar{y} + \bar{w})$$

Wyrażenie to jest identyczne z wyrażeniem uzyskanym poprzednio.

Przykład 2

$F = \bar{w}\bar{y} + \bar{y}x + x\bar{w} + \bar{y}\bar{z} + \bar{z}x$ - dysjunkcyjna normalna postać skrócona. Po zastosowaniu zaproponowanych przekształceń uzyskujemy następujące wyrażenia:

$$F = \bar{y}(\bar{w} + \bar{z}) + w(\bar{y} + \bar{w} + \bar{z})$$

lub

$$F = \bar{y}(\bar{w} + \bar{z}) + x(\bar{w} + \bar{z}) + x\bar{y}$$

Czyli również w tym przypadku wynik jest taki sam.

Przykład 3

$F = \bar{w}x\bar{y} + \bar{w}\bar{y}\bar{z} + \bar{x}\bar{y}\bar{z} + w\bar{x}\bar{z}$ - dysjunkcyjna normalna postać skrócona. Po zastosowaniu przekształceń uzyskujemy wyrażenie następujące:

$$F = \bar{x}\bar{y}(\bar{w} + \bar{z}) + w\bar{z}(\bar{y} + \bar{x})$$

Wynik znowu jest identyczny z poprzednim.

Weźmy jeszcze inny przykład:

$F = u\bar{z}\bar{w} + u\bar{z}\bar{x} + u\bar{y}\bar{w} + u\bar{y}\bar{x}$ - dysjunkcyjna normalna postać skrócona, 9 elementów i 20 połączeń.

Przekształcając dwukrotnie to wyrażenie otrzymujemy po kolei następujące wyrażenia:

$$F = u\bar{z}(\bar{w} + \bar{x}) + u\bar{y}(\bar{w} + \bar{z}) \quad 6 \text{ elementów } 12 \text{ połączeń}$$
$$F = u(\bar{w} + \bar{x})(\bar{z} + \bar{y}) \quad 4 \text{ elementy } 8 \text{ połączeń}$$

Zarówno pierwsze jak i drugie wyrażenie jest bezhazardowe i o wiele tańsze w realizacji za pomocą elementów NAND.

Taki sposób bezhazardowej minimalizacji jest wyjątkowo prosty. Biorąc pod uwagę, że dla funkcji do 6-zmiennych dysjunkcyjną normalną postacią skróconą można uzyskać stosunkowo łatwo z siatki Karnaugha, walor prostoty tego sposobu dodatkowo wzrasta. Jedynym mankamentem wydaje się być brak pewności, czy w każdym przypadku stosowania tej metody wynik jest minimalny.

Literatura

- [1] HŁAWICZKA A.: Realizacja dowolnych funkcji n-zmiennych za pomocą kombinacyjnych bezhazardowych trójwarstwowych struktur logicznych o funktorach NOR, VKKA Zeszyt 2, Gdańsk 1971.
- [2] GIMPEL F.: The Minimization of TANT Networks, IEEE Transaction on Electronic Computers, VC-16, No 1, February, 1967.
- [3] McCLUSKEY E.I.: Introduction to the Theory of Switching Circuits, New York 1965.
- [4] HUFFMAN D.A.: The Design and Use of Hazard-free Switching Networks, I.A.C.M., January 1957/4.
- [5] SIWIŃSKI J.: Układy przełączające w automatyce, WNT, Warszawa 1968.
- [6] NICHOLS J.: Transistor-Transistor Micrologic Integrated Circuits.
- [7] FRĄCKOWIAK J.: Algorytm G^m generacji wyższych maksymalnych impli-centów dopuszczalnych dla minimalizacji sieci NOR, VKKA, zeszyt 2, Gdańsk 1971.
- [8] UNGER S.H.: Hazard and Delays in Asynchronous Sequential Switching Circuits, IRE Transactions on Circuit Theory, March 1959.
- [9] WIEWIÓRKOWSKI T.: Aktualne problemy syntezy kombinacyjnych układów logicznych budowanych z elementów NAND/NOR, Sympozjum CO PAN, 1971.
- [10] WALIGÓRSKI S.: Układy przełączające. Elementy teorii i projektowania, WNT, Warszawa 1969.
- [11] MIL-STD-806B /norma amerykańska z 26.2.1962/.

DODATEK 1 [D1]

Definicje i twierdzenia metody Gimpela syntezy minimalnych układów TANT

Definicja

Czynnikiem podstawowym T nazywamy iloczyn różnych nie zangowanych zmiennych lub jedynek Boole'a, np. $x, w \cdot x \cdot y \cdot z$

Definicja

Wyrażeniem dopuszczalnym E nazywamy dowolne wyrażenie Boole'a różne od zera w postaci $T_0 \cdot \bar{T}_1 \cdot \bar{T}_2 \cdot \dots \cdot \bar{T}_n$ dla $n \leq 0$, gdzie każde T_i dla $i \in \{0, 1, \dots, n\}$ jest czynnikiem podstawowym.

O czynniku podstawowym T_0 będziemy mówili jak o początku wyrażenia dopuszczalnego i dalej oznaczać go będziemy przez H . Wyrażenie $\bar{T}_1 \cdot \dots \cdot \bar{T}_n$ będzie nazywane końcem wyrażenia E , a każde \bar{T}_i będzie nazywane czynnikiem końca.

Definicja

Wyrażeniem TANT nazywamy dowolną sumę wyrażeń dopuszczalnych, np. $E_1 + E_2 + \dots + E_k$.

Definicja

Niezbędnym wyrażeniem dopuszczalnym nazywamy wyrażenie, z którego nie można usunąć żadnego czynnika bez zmiany podstawowej funkcji.

Definicja

Dopuszczalnym implikantem P_d funkcji Boole'a jest funkcja, która zawiera się w F i którą można przedstawić w formie wyrażenia dopuszczalnego E .

Definicja

Nierozszerzonym wyrażeniem dopuszczalnym E_n nazywamy podstawowe wyrażenie dla danego implikantu dopuszczalnego P_d . Jego czynniki końca nazywają się podstawowymi czynnikiem końca.

Definicja

Prostym dopuszczalnym implikantem lub implikantem PP funkcji F nazywamy każdy taki dopuszczalny implikant, z którego jeżeli zostanie usunięty jakikolwiek czynnik końca, to otrzymana funkcja nie będzie się zawierała w F .

Definicja

Maksymalny implikant dopuszczalny P_{dm} z początkiem H określony jest jako suma wszystkich implikantów dopuszczalnych z początkiem H .

Twierdzenie

Każdy implikant prosty P_p jest implikantem dopuszczalnym P_d .

Definicja

Każdy implikant dopuszczalny zawarty w implikancie prostym nazywamy implikantem niższym.

Definicja

Każdy implikant dopuszczalny, który nie jest niższy nazywamy wyższym.

Definicja

Quasi prostym implikantem dopuszczalnym P_{dq} nazywamy każdy implikant dopuszczalny, który ma co najmniej jeden czynnik końca będący sumą zmiennych zanegowanych.

Definicja

Quasi prostym implikantem dominującym nazywamy taki dopuszczalny implikant P_o , który z każdym pozostałym implikantem dopuszczalnym P_d spełnia następujący test $P_o \cdot P_d = 0$ lub $P_o \supseteq P_d$.

Definicja

Głównym implikantem P_{dg} jest dowolny wyższy implikant PP nie zawarty całkowicie w dominującym quasi prostym implikancie P_o .

Twierdzenie

Dla dowolnej funkcji F istnieje minimalne wyrażenie TANT takie, że każdy implikant dopuszczalny P_d jest głównym implikantem P_dg .

DODATEK 2 [D2]

Twierdzenie 7.3.5 z pracy McCluskey "Introduction to the theory of switching circuits", New York 1965, dotyczące istnienia HSO i HS1 w dwuwarstwowych układach logicznych I-LUB oraz LUB-I. HS1/HSO/ istnieje w sieci jeśli i tylko jeśli istnieje para stabilnych zbiorów 1 /zbiorów 0/ takich, że:

- 1/ Istnieje tylko jedna zmienna, która jest zanegowana w jednym ze zbiorów 1 /zbiorów 0/ i niezanegowana w drugim zbiorze 1 /zbiorze 0/, przy czym dopuszczalna jest dowolna liczba liter w obu zbiorach 1 /zbiorach 0/.
- 2/ Każdy z innych stabilnych zbiorów 1 /zbiorów 0/ posiada w lub \bar{w} lub zanegowaną jedną z innych liter włączoną w każdy z dwóch zbiorów 1 /zbiorów 0/ pary, lub zmienną, która nie pojawi się w żadnym z dwóch zbiorów 1 /zbiorów 0/ przedstawionej pary.

О НЕКОТОРОМ МЕТОДЕ СИНТЕЗА МИНИМАЛЬНЫХ НЕАЗАРТНЫХ КОМБИНИРОВАННЫХ ПЕРЕКЛЮЧАЮЩИХ СХЕМ TANT

Резюме

Работа заключает описание метода синтеза минимальных неазартных комбинированных трехслойных переключающих схем, построенных из элементов NAND а также основные теоремы и дефиниции касающиеся идентификации и элиминации статического и динамического азарта в таких схемах переключения.

В заключении поданы предложения и много примеров применения алгоритма синтеза минимальных неазартных схем TANT.

ON A CERTAIN METHOD OF MINIMAL NONHAZARDOUS SYNTHESIS OF COMBINATIVE SWITCHING TANT ELEMENT

Summary

The paper encloses a description of a method of minimal nonhazardous synthesis of combinative three layer switching elements built of NAND elements and the basic theorems and definitions concerning the identification and elimination of static and dynamic hazard in such switching elements.

In the end conclusions and numerous examples are given of the application of the algorithm of minimal nonhazardous synthesis of TANT elements.

AUTORZY ARTYKUŁÓW ZAMIESZCZONYCH W ZESZYCIE 3 "PRAC IMM"

BADANIE SYMULACYJNE NIEKTÓRYCH ALGORYTMÓW PODZIAŁU CZASU
PROCESORA W SYSTEMACH PRZETWARZANIA INFORMACJI

Andrzej WIATER mgr inż. W roku 1970 ukończył Wydział Elektroniki Politechniki Warszawskiej w specjalności Maszyny Matematyczne. Od 1969 roku pracuje w Instytucie Maszyn Matematycznych w Zakładzie Sterowania, obecnie na stanowisku starszego asystenta. Zajmował się automatyzacją projektowania połączeń logicznych na pakietach mikroelektronicznych, a następnie zagadnieniami symulacji procesów dyskretnych oraz sieciami transmisji systemów przesyłania informacji.

O PEWNEJ METODZIE SYNTEZY MINIMALNYCH BEZHAZARDOWYCH
KOMBINACYJNYCH UKŁADÓW PRZEŁĄCZAJĄCYCH TANT

Andrzej HŁAWICZKA mgr inż. Ukończył studia w 1965 r. na Wydziale Elektrycznym Politechniki Śląskiej w Gliwicach. W 1968 r. ukończył dwuletnie podyplomowe studia automatyki i telemekhaniki na Wydziale Automatyki Politechniki Śląskiej. W latach 1965-1971 zajmował się projektowaniem i wdrażaniem cyfrowych układów sterowania jako starszy konstruktor, a następnie starszy projektant. W tym czasie opublikował szereg prac z zakresu zaprojektowanych przez siebie układów cyfrowego sterowania oraz zgłosił do Urzędu Patentowego kilka patentów i wzorów użytkowych. Od 1971 r. pracuje na stanowisku adiunkta i kieruje Pracownią Urządzeń Cyfrowych w Śląskim Oddziale Instytutu Maszyn Matematycznych. Dodatkowo zajmuje się problemami teorii automatów, a szczególnie zawodnością strukturalną bezstykowych układów przełączających. Opublikował w tej dziedzinie trzy prace.

WYDAWNICTWA IMM

Branżowy Ośrodek Informacji Naukowo-Technicznej i Ekonomicznej Instytutu Maszyn Matematycznych wydaje:

ALGORYTMY - półrocznik, zawiera artykuły na temat teorii programowania i zastosowania elektronicznych maszyn cyfrowych. Do nabycia w księgarni ORWN PAN oraz w Domach Książki. Cena zeszytu 40,- zł.

PRACE IMM - 3. numery w roku, zawierają publikacje naukowe i badawcze pracowników IMM w zakresie projektowania i budowy elektronicznych maszyn cyfrowych oraz systemów przetwarzania informacji. Do nabycia w księgarni ORWN PAN oraz w Domach Książki. Cena zeszytu 60,- zł.

Elektroniczna Technika Obliczeniowa - NOWOŚCI - kwartalnik, zawiera artykuły przeglądowe z dziedziny maszyn matematycznych, opracowane na podstawie najnowszej literatury światowej. Prenumeratę prowadzi Centrala Kolportażu Prasy i Wydawnictw "RUCH". Cena prenumeraty krajowej 240,- zł rocznie.

Automatyzacja Przetwarzania Informacji - INFORMACJA EKSPRESOWA - miesięcznik. Prenumeratę prowadzi Centrala Kolportażu Prasy i Wydawnictw "RUCH". Cena prenumeraty krajowej 240,- zł rocznie.



WYDAWNICTWA PRZEMYSŁU MASZYNOWEGO "WEMA"
oferują usługi wydawnicze

Od 5 lat działa w Warszawie specjalne wydawnictwo resortowe powołane do świadczenia usług wydawniczych na rzecz jednostek organizacyjnych resortu przemysłu maszynowego.

Do szczególnych zadań Wydawnictw Przemysłu Maszynowego "WEMA" należy:

- prowadzenie działalności wydawniczej zgodnie z potrzebami resortu,
- koordynacja działalności wydawniczej w jednostkach organizacyjnych resortu,
- koordynacja i nadzór nad prawidłowym wykorzystaniem maszyn i urządzeń poligraficznych,
- prowadzenie własnego ośrodka poligraficznego,
- prowadzenie ośrodka informacji wydawniczej.

Od ubiegłego roku Wydawnictwo znacznie rozszerzyło zakres usług i obecnie wydaje:

- katalogi branżowe i karty katalogowe

oraz na zlecenie przedsiębiorstw przemysłowych różnego rodzaju literaturę firmową, jak:

- katalogi zakładowe,
- katalogi części wymiennych,
- informatory techniczno-handlowe,
- dokumentacje techniczno-ruchowe, instrukcje obsługi i instrukcje naprawcze,
- dokumentacje techniczne kapitalnych remontów,
- wydawnictwa reklamowe, jak prospekty, foldery, ulotki itp.

Katalogi branżowe wydaje się w porozumieniu i we współpracy z właściwymi gestyjnie zjednoczeniami.

Sprzedają katalogów WPM "WEMA" zajmują się następujące księgarnie:
Księgarnie "WSPÓLNEJ SPRAWY":
Warszawa, ul. Marszałkowska 28, tel. 21-66-60
Warszawa, ul. Marchlewskiego 35, tel. 20-49-69

"DOM KSIĄZKI":

Główna Księgarnia Techniczna, Warszawa, ul. Świętokrzyska 14,
tel. 26-63-38.

Księgarnie te prowadzą sprzedaż odręczną i wysyłkową.

Literaturę firmową WPM "WEMA" wykonują na konkretne zamówienie przedsiębiorstw przemysłowych.

WPM "WEMA" znacznie skróciły cykle wydawnicze i zapewniają obecnie terminową realizację zamówień.

Wszelkich informacji na temat warunków przyjmowania i realizacji zamówień wydawniczych udziela Sekretariat Wydawnictwa, Warszawa, ul. Daniłowiczowska 18, pokój nr 7, tel. 27-49-47, skr. poczt. 90.

BIBLIOTEKA GŁÓWNA
Politechniki Śląskiej

P 2229/72