

1 1976

prace



P. 2229/76

Instytutu Maszyn Matematycznych

rok XVIII

prace

Instytutu

Maszyn

Matematycznych



P.2229/76

Andrzej HŁAWICZKA

IDENTYFIKACJA I ELIMINACJA HAZARDU
PODSTAWOWEGO W WIELOWARSTWOWYCH
UKŁADACH PRZELĄCZAJĄCYCH

Praca doktorska pod kierunkiem
prof. dr inż. Jerzego Śliwińskiego

Zjednoczenie Przemysłu Automatyki i Aparatury
Pomiarowej "MERA"

Instytut Maszyn Matematycznych

Copyright (C) 1976 - by Instytut Maszyn Matematycznych
Poland

Wszelkie prawa zastrzeżone

KOMITET REDAKCYJNY

Jerzy GRADOWSKI, Andrzej JANICKI (z-ca red. nacz.),
Roman KULESZA (redaktor naczelny),
Antoni MAZURKIEWICZ, Tomasz PAWLAK, Ryszard PREGIEL,
Władysław M. TURSKI (z-ca red. nacz.), Zbigniew WIERZBICKI

Sekretarz Redakcji: Romana NITKOWSKA

Redaktor Techniczny: Maria KOZŁOWSKA

Adres Redakcji: Instytut Maszyn Matematycznych
Branżowy Ośrodek INTE
ul. Krzywickiego 34, 02-078 WARSZAWA
tel. 28-37-29

IDENTYFIKACJA I ELIMINACJA HAZARDU PODSTAWOWEGO
W WIELOWARSTWOWYCH UKŁADACH PRZEŁĄCZAJĄCYCH

Andrzej HŁAWICZKA

Pracę złożono 18.07.1973

Przedstawiono metodę identyfikacji k-krotnego hazardu podstawowego w wielowarstwowych kombinacyjnych układach przełączających realizujących funkcje n-zmiennych i zbudowanych na bazie elementów I, LUB, NIE, NAND i NOR. Na podstawie wprowadzonych definicji oraz udowodnionych lematów i twierdzeń zbudowano algorytm I^k . Przedstawiono siedem przykładów wykorzystania algorytmu I^k do identyfikacji k-krotnego hazardu podstawowego za pomocą składania tablicy strukturalnych implikantów i implikentów oraz za pomocą składania siatek Karnaugh'a. Podano sposób eliminacji dowolnych k-krotnych hazardów podstawowych na wyjściu dowolnego elementu logicznego wielowarstwowego układu przełączającego. Przedstawiono dwie metody syntezy minimalnych bezhazardowych układów TANT realizujących dowolne funkcje n-zmiennych.

S p i s t r e ś c i

WYKAZ OZNACZEŃ LITEROWYCH UŻYWANYCH W PRACY	5
WSTĘP	7
1. PRÓBA UPORZĄDKOWANIA POJĘĆ ZWIĄZANYCH Z HAZARDEM I INFORMACJĘ OGÓLNE	
1.1. Podział hazardów i ich definicje	10
1.2. Potencjalowy asynchroniczny automat sekwencyjny i warunki zapewniające jego poprawną pracę	15
1.3. Przykłady szkodliwego wpływu hazardu podstawowego na poprawną pracę potencjalowego asynchronicznego automatu sekwencyjnego	20

2. METODA IDENTYFIKACJI k-KROTNEGO HAZARDU PODSTAWOWEGO ZE WZGLĘ- DU NA ZMIENNĄ x_1 W WIELOWARSTWOWYCH KOMBINACYJNYCH UKŁADACH PRZEŁĄCZAJĄCYCH REALIZUJĄCYCH FUNKCJE n-ZMIENNYCH	
2.1. Wprowadzenie	26
2.2. Definicje podstawowych pojęć	32
2.3. Podstawowe lematy	40
2.4. Algorytm I^k identyfikacji k-krotnego hazardu podstawowego ze względu na zmienną x_1	47
2.4.1. Podstawowa twierdzenia	47
2.4.2. Charakterystyka algorytmu I^k	53
2.5. Zastosowanie algorytmu I^k do identyfikacji k-krotnego ha- zardu podstawowego w kombinacyjnych układach przełączają- cych za pomocą składania tablicy strukturalnych implikantów i strukturalnych implimentów	54
2.5.1. Tablica strukturalnych implikantów i strukturalnych implimentów	54
2.5.2. Przykłady	58
2.6. Zastosowanie algorytmu I^k do identyfikacji k-krotnego hazar- du podstawowego w kombinacyjnych układach przełączających za pomocą składania siatek Karnauga	64
2.6.1. Przykłady	64
2.7. Ogólna charakterystyka metody	76
3. ELIMINACJA k-KROTNEGO HAZARDU PODSTAWOWEGO W UKŁADACH PRZEŁĄ- CZAJĄCYCH	
3.1. Informacje wstępne	79
3.2. Wykorzystanie algorytmu I^k do eliminacji k-krotnych hazar- dów podstawowych w wielowarstwowch kombinacyjnych układach przełączających	80
3.3. Wykorzystanie algorytmu I^k do eliminacji szkodliwych hazar- dów podstawowych w potencjalowych asynchronicznych automa- tach sekwencyjnych	86
3.4. Wnioski	99
4. METODY SYNTEZY BEZHazardOWYCH UKŁADÓW TANT REALIZUJĄCYCH FUNKCJE n-ZMIENNYCH	
4.1. Uwagi wstępne	102
4.2. Algorytm G^{x_1} syntezy bezhazardowych układów TANT	104

4.2.1. Podstawowe definicje i twierdzenia	104
4.2.2. Określenie bazy realizacji (G_{min}, T_{min}) za pomocą redukcji rozwiniętej tablicy AA'	108
4.2.2.1. Rozwinięta tablica AA'	109
4.2.2.2. Technika redukcji rozwiniętej tablicy AA'	112
4.2.3. Charakterystyka algorytmu G^{xi}	115
4.2.4. Przykłady	116
4.3. Adaptacja znanych metod faktoryzacji do algebraicznego spo- sobu syntezy bezhazardowych układów TANT	130
4.3.1. Przykłady	131
4.4. Ogólna charakterystyka metod syntezy bezhazardowych ukła- dów TANT	135
ZAKOŃCZENIE	138
PRZYPISY	140
BIBLIOGRAFIA	153
DÓDATEK	I

WYKAZ OZNACZEŃ LITEROWYCH UŻYWANYCH W PRACY

A	- stan wejścia
B	- idealny element logiczny
B^m	- m-ty element logiczny układu przełączającego
C_U	- koszt pokrycia
C_{TANT}	- koszt układu TANT
E	- wyrażenie dopuszczalne
EN	- nierozszerzone wyrażenie dopuszczalne
F	- funkcja Boole'a
F^m	- funkcja realizowana przez fragment układu przełączającego złożonego z elementu B^m i wycinka W^m
F_P	- funkcja przejściowa
F_S	- funkcja wpisująca
F_R	- funkcja zerująca
G	- zbiór wszystkich głównych implikantów funkcji F
G_{min}	- podzbiór głównych implikantów ze zbioru G
G^{xi}	- algorytm syntezy bezhazardowych układów TANT
H	- początek wyrażenia dopuszczalnego
$HP^k(x_1)$	- k-krotny hazard podstawowy ze względu na zmienną x_1
$HS^k(x_1)$	- k-krotny hazard statyczny w warunkach działania ze względem na zmienną x_1

- $HSO^k(x_1)$ - k-krotny hazard statyczny w warunkach niedziałania ze względu na zmienną x_1
- $HD^k(x_1)$ - k-krotny hazard dynamiczny ze względu na zmienną x_1
- I^k - algorytm identyfikacji $HP^k(x_1)$
- M - zbiór kolumn AA'
- \bar{N} - zbiór kolumn \bar{T}
- PD - implikant dopuszczalny
- PG - implikant główny
- PK_o^m - implikant strukturalny
- QK_o^m - implicent strukturalny
- S^m - wyrażenie strukturalne fragmentu układu przełączającego realizującego funkcję F^m
- \bar{T} - podstawowy czynnik końca
- T - zbiór podstawowych czynników końca
- T_{min} - podzbiór ze zbioru T
- TANT - trójwarstwowy układ przełączający zbudowany z elementów NAND
- U - zbiór wierszy tablicy AA' (pokrycie tablicy AA')
- W^m - m-ty wycinek układu przełączającego
- Z - wektor określający zbiór zmiennych wyjściowych automatu
- Y - wektor określający zbiór zmiennych stanu wewnętrznego automatu
- Y - zmienne stanu wewnętrznego automatu
- X - wektor określający zbiór zmiennych wejściowych automatu
- a - stan logiczny
- c - kolumna \bar{T} tablicy AA'
- d - opóźnienie
- da - czasowy stan logiczny
- f - opis funkcji F
- u - wiersz tablicy AA'
- g, h, i, j, k, l, m, n, o, p, r, s, t - indeksy
- z - zmienne wyjściowe
- y - zmienna poprzedniego stanu wewnętrznego automatu
- x - zmienna wejściowa
- λ - liczba elementów logicznych
- μ - liczba wejść na elementy logiczne
- τ - czas
- $\bigvee_{i=1}^p x_i$ - alternatywa
- $\bigwedge_{i=1}^p x_i$ - koniunkcja

WSTĘP

Możliwość niepożądanego przejścia układu sprawnego technicznie układu przełączającego na zmianę stanu wejścia tego układu ogólnie nazywa się hazardem. Jeżeli hazard występuje w układzie kombinacyjnym lub w kombinacyjnej części układu sekwencyjnego i jest wywołany przejściem stanu wejść w stan sąsiedni, to nazywa się go hazardem podstawowym. Jeżeli hazard podstawowy może objawić się na wyjściu układu przełączającego wielokrotnymi (k -krotnymi)^{*)} niepożądanymi zmianami stanu wyjścia to nazywa się go k -krotnym hazardem podstawowym. Nieszkodliwy zazwyczaj i o przejściowym charakterze w układach kombinacyjnych k -krotny hazard podstawowy może mieć szkodliwy wpływ na poprawną pracę układu, gdy występuje w asynchronicznych układach sekwencyjnych. W układach tych niepożądane reakcje wyjścia mogą zostać utrwalone w pamięci układu.

Elementy logiczne przeznaczone do celów automatyki i sterowania pracują zazwyczaj w warunkach silnych zakłóceń przemysłowych i w związku z tym są najczęściej układami tzw. wysokopoziomowymi, zasilanymi na ogół napięciami powyżej 12 V. Układy te zrealizowane są najczęściej techniką grubowarstwową opierając się na prostych elementach półprzewodnikowych. Obecnie pojawiły się również pierwsze szeregi elementów scalonych przeznaczone do pracy w warunkach silnych zakłóceń przemysłowych (np. elementy serii SN 15300 z firmy Texas Instruments, elementy serii H-100 z firmy SGS, elementy serii FZ-100 z firmy Siemens). Znaczna ich część stanowi systemy minimalne o jednym podstawowym elemencie NAND. Przeprowadzanie syntezy układów przełączających opierając się na elementach NAND sprowadza się zazwyczaj do budowy wielowarstwowych układów przełączających (liczba warstw ≥ 3). Dla uniknięcia w takich układach niepożądanych i szkodliwych skutków istnienia hazardów podstawowych niezbędne są proste inżynierskie metody ich identyfikacji i eliminacji oraz metody realizacji bezhazardowej syntezy.

*) Liczba k jest tutaj jednocześnie przybliżoną informacją o liczbie punktów w układzie, które mogą wywołać przekłamania na wyjściu tego układu

W niniejszej pracy została przedstawiona nowa metoda identyfikacji k -krotnych hazardów podstawowych w wielowarstwowych kombinacyjnych układach przełączających realizujących funkcje n -zmiennych zbudowanych na dowolnych zestawach elementów I, LUB, NIE, NAND, NOR. Przedstawiono także sposób wykorzystania podanej metody identyfikacji do eliminacji każdego k -krotnego hazardu podstawowego w wielowarstwowych układach przełączających. Ma to bardzo ważne znaczenie, zwłaszcza w układach kombinacyjnych, które stanowią część układów sekwencyjnych. Dla tych układów przedstawiono jednocześnie sposób eliminacji tylko szkodliwych hazardów podstawowych.

W pracy przedstawiono również metodę syntezy minimalnych bezhazardowych trójwarstwowych kombinacyjnych układów zbudowanych z elementów NAND i realizujących daną funkcję n -zmiennych.

Znane dotychczas metody identyfikacji hazardu podstawowego są przedstawione w pracach [18], [31] i [35]. Metody te, opisane w rozdziale 2 niniejszej pracy, nie pozwalają jednak na znalezienie wszystkich miejsc układu przełączającego, które są bezpośrednią przyczyną istnienia k -krotnych hazardów podstawowych na wyjściu układu. Utrudnia to tym samym, a czasami wręcz uniemożliwia, eliminowanie hazardów podstawowych w wielowarstwowych układach przełączających. Metoda przedstawiona w niniejszej pracy rozwiązuje ten problem, a ponadto pozwala uniknąć uciążliwego szukania różnych zbiorów wg metod opublikowanych w [18], [31], [35]. Kilka przykładów zastosowania fragmentu przedstawionej metody do identyfikacji hazardu podstawowego podał autor w swojej wcześniejszej pracy [15].

Spotykana w literaturze specjalistycznej metoda eliminacji hazardu podstawowego drogą deminimalizacji wyrażenia strukturalnego dotyczy przede wszystkim dwuwarstwowych układów kombinacyjnych I-LUB oraz LUB-I. Dla układów wielowarstwowych nie przedstawiono dotychczas jednolitego sposobu eliminacji k -krotnego hazardu podstawowego za pomocą deminimalizacji wyrażenia strukturalnego. Przedstawiony w niniejszej pracy sposób eliminacji hazardu podstawowego pozwala na wyeliminowanie dowolnych k -krotnych hazardów podstawowych na wyjściach dowol-

nych elementów logicznych wielowarstwowego układu przełączającego zbudowanego na bazie zestawu elementów I, LUB, NIE, NAND, NOR. Szczególnie przy eliminowaniu tylko szkodliwych hazardów podstawowych, przedstawiony sposób pozwala na budowę tańszych i poprawnie działających wielowarstwowch układów przełączających. Bezhazardowa synteza trójwarstwowch układów kombinacyjnych zbudowanych z elementów NAND, zwanych układami TANT, opracowana została na podstawie metody Gimpala przedstawionej w pracy [11] oraz na podstawie prac autora [16], [17].

Przedstawiona metoda identyfikacji k-krotnego hazardu podstawowego, jak również sposób eliminacji tego hazardu, a szczególnie hazardu szkodliwego, oraz podana metoda bezhazardowej syntezy trójwarstwowch układów zbudowanych z elementów NAND przedstawiają sposoby niespotykane dotąd w literaturze. Ze względu na nieskomplikowane algorytmy postępowania przedstawione metody mogą być szeroko stosowane w praktyce inżynierskiej.

Rozdział pierwszy pracy poświęcony jest bardziej szczegółowemu przedstawieniu problemu hazardu podstawowego i jego szkodliwego wpływu na poprawną pracę potencjałowych asynchronicznych automatów sekwencyjnych. Rozdział ten zawiera również próbę ujednoczenia pojęć związanych z hazardem. Rozdziały drugi, trzeci i czwarty są istotną częścią pracy i przedstawiają opracowane przez autora metody identyfikacji i eliminacji hazardu podstawowego oraz metodę bezhazardowej syntezy. Teoretyczne rozważania przedstawione w tych rozdziałach zostały poparte konkretnymi przykładami, które bazują przede wszystkim na układach zbudowanych z elementów NAND. Metodę identyfikacji i eliminacji k-krotnego hazardu podstawowego autor opracował szczegółowo dla kombinacyjnych układów przełączających. Natomiast dla układów sekwencyjnych można jeszcze ten temat rozwinąć i np. rozpatrzyć sposób eliminacji szkodliwych hazardów podstawowych w impulsowych układach sekwencyjnych.

Definicje, twierdzenia i przykłady ponumerowano podając numer podrozdziału i kolejny numer w danym podrozdziale. Wzory i rysunki mają numerację kolejną w obrębie całej pracy. Umieszczone w pracy zestawienie oznaczeń literowych ma na celu ułatwienie czytania pracy. W przypisach zamieszczonych

na końcu pracy podano wiele istotnych informacji związanych ze stosowanymi pojęciami oraz dodatkowe informacje.

Autor pragnie złożyć bardzo serdeczne podziękowanie prof. dr inż. Jerzemu Siwińskiemu za prowadzenie niniejszej pracy i życzliwą opiekę. Również bardzo serdecznie autor dziękuje całemu zespołowi pracowników Instytutu Automatyki Przemysłowej i Pomiarów, a szczególnie dr Henrykowi Małysiakowi i dr Ferdynandowi Wagnerowi za życzliwość i koleżeńską atmosferę jaką stworzyli mu przy realizacji tej pracy.

1. PRÓBA UPORZĄDKOWANIA POJĘĆ ZWIĄZANYCH Z HAZARDEM I INFORMACJE OGÓLNE

1.1. Podział hazardów i ich definicje

W zależności od formy przekłamania w jakiej hazard objawia się wewnątrz lub na wyjściu układu przełączającego, w zależności od tego w jakiego rodzaju układzie przełączającym występuje i wreszcie w zależności od tego jakich skutków hazard staje się przyczyną i w jakich warunkach się pojawia, wprowadzono w specjalistycznej literaturze światowej, a przede wszystkim w literaturze w języku angielskim, wiele pomocniczych określeń występujących zawsze razem ze słowem "hazard".

D.A. Huffman w swojej pracy [18] zdefiniował pojęcia: "hazard podstawowy" (basic hazard), "hazard statyczny" (static hazard), "hazard dynamiczny" (dynamic hazard), "hazard wielokrotnego stopnia" (multiple-order-hazard) oraz wprowadził pojęcie "bezhazardowy" (hazard-free). W pracy [66] wprowadzono nowe określenia: "hazard przejściowy" (transient hazard), "hazard trwały" (steady-state-hazard) oraz "hazard istotny" (essential hazard). W pracy [27] pojawiły się następne pomocnicze określenia tzw. "hazard kombinacyjny" (combinational hazard) oraz "hazard niestotny" (nonessential hazard). E.B. Eichelberger w pracy [6] rozróżnia ponadto tzw. "hazard logiczny" (logic hazard), "hazard funkcyjny" (function hazard), "hazard logiczny p - zmiennych" (p-variable logic hazard) oraz "hazard M"

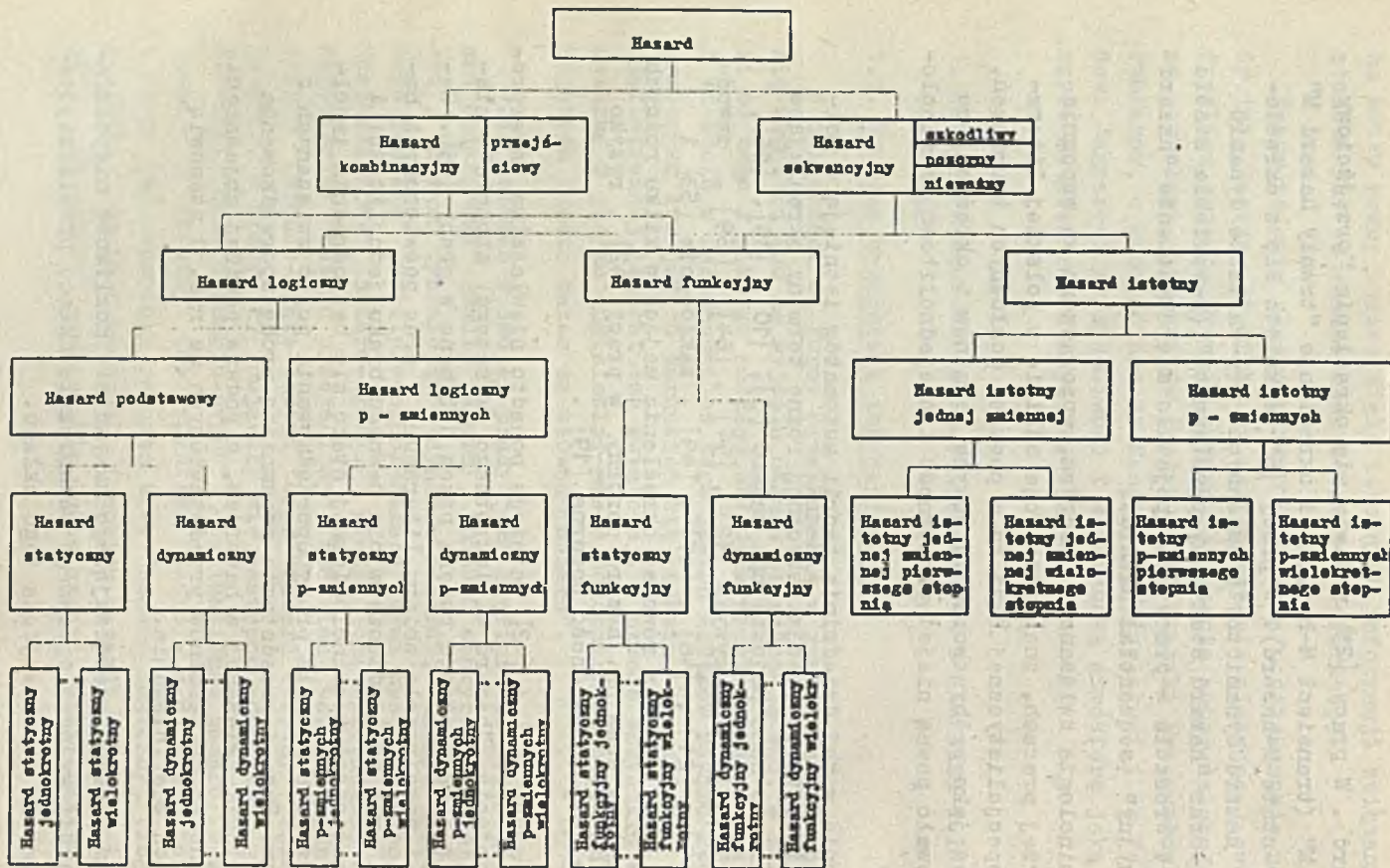
(M-hazard). W pracy [25] pojawia się określenie "przejściowy hazard M" (transient M-hazard) i określenie "trwały hazard M" (steady-state-M-hazard). W pracy [65] zetknięto się z określeniami: "hazard dynamiczny p-zmiennych" (p-variable dynamic hazard) oraz "hazard statyczny p-zmiennych" (p-variable static hazard). Wreszcie w pracy [36] pojawiło się określenie "hazard sekwencyjny" (sequential hazard).

Terminologia związana z hazardem, wprowadzona we wspomnianych wyżej pracach, znalazła swoje odbicie w polskiej literaturze specjalistycznej bądź to w postaci dosłownych tłumaczeń, bądź też jako próby tworzenia nowych terminów i określeń, co spowodowało pewną niejednoznaczność i niejednorodność terminologiczną.

Obecnie w tej dziedzinie teorii automatów istnieją w polskiej literaturze specjalistycznej różne terminy określające dokładnie to samo pojęcie np. hazard [3], [10], [15], [16], [22], [58], [59], ryzyko [12], [54], [64], [69], zawodność [19], [21], [46], [47], [49], zakłócenie [53] i ślizg [3]. Istnieją również określenia mające w kilku różnych pracach inny sens np. hazard pozorny w pracy [42], ryzyko pozorne [69] i zawodność pozorna [19].

W pracach [41] i [42] istnieje ponadto niewłaściwa interpretacja pojęcia "hazard wielokrotnego (wyższego) stopnia" niezgodna z interpretacją podaną przez Huffmana w pracy [18]. Zebranie i przejrzanie zbioru literatury i prac poświęconych hazardowi oraz stwierdzenie w nim pewnego braku jednorodności i jednoznaczności terminologicznej stało się bezpośrednim impulsem do zaproponowania próby uporządkowania pojęć związanych z hazardem. Rys. 1 przedstawia wspomnianą próbę uporządkowania terminologii związanej z hazardem, a podane poniżej objaśnienia zaproponowanych terminów przedstawionych na rys. 1 stanowią próbę ich zdefiniowania.

Hazardem układu przełączającego nazywa się możliwość niepożądanego reakcji wyjścia sprawnego technicznie układu przełączającego na zmianę stanu wejścia tego układu.



Rys. 1. Próba uproszczenia terminologii związanej z hasardem.

Hazardem kombinacyjnym nazywa się każdy hazard występujący wyłącznie w kombinacyjnym układzie przełączającym.

Hazardem sekwencyjnym nazywa się każdy hazard występujący wyłącznie w sekwencyjnym układzie przełączającym.

Hazardem przejściowym nazywa się możliwość wystąpienia krótkotrwałej niepożądanego reakcji wyjścia sprawnego technicznie kombinacyjnego układu przełączającego na zmianę stanu wejścia tego układu.

Hazardem szkodliwym (trwałym) nazywa się możliwość utrwalenia w pamięci sekwencyjnego układu przełączającego niepożądanego reakcji wyjścia na zmianę stanu wejścia tego układu.

Hazardem pozornym nazywa się hazard, który jest zawsze samoczynnie likwidowany przez sekwencyjny układ przełączający.

Hazardem nieważnym nazywa się hazard występujący tylko przy zmianie stanu wejścia sekwencyjnego układu przełączającego, której nie przewidują warunki pracy tego układu.

Hazardem logicznym nazywa się każdy hazard, który można wyeliminować z układu przełączającego drogą rozbudowania struktury tego układu.

Hazardem funkcyjnym nazywa się każdy hazard, który nieodłącznie związany jest z funkcją realizowaną przez układ przełączający i którego nie można wyeliminować z tego układu bez zmiany funkcji, jaką ten układ realizuje.

Hazardem statycznym układu przełączającego realizującego funkcję F nazywa się każdy hazard logiczny wywołany zmianą stanu wejścia A na sąsiedni stan wejścia A' , przy czym $f(A) = f(A')$.

Hazardem dynamicznym układu przełączającego realizującego funkcję F nazywa się każdy hazard logiczny wywołany zmianą stanu wejścia A na sąsiedni stan wejścia A' , przy czym $f(A) \neq f(A')$.

Hazardem podstawowym (zerowego stopnia) nazywa się każdy hazard statyczny lub każdy hazard dynamiczny.

Hazardem statycznym p-zmiennych układu przełączającego realizującego funkcję F nazywa się każdy hazard logiczny wywołany zmianą stanu wejścia A na stan wejścia A'' różniący się p-stanami logicznymi, przy czym $f(A) = f(A'')$.

Hazardem dynamicznym p-zmiennych układu przełączającego realizującego funkcję F nazywa się każdy hazard logiczny wywołany zmianą stanu wejścia A na stan wejścia A'' różniący się p-stanami logicznymi, przy czym $f(A) \neq f(A'')$.

Hazardem logicznym p-zmiennych nazywa się każdy hazard statyczny p-zmiennych lub każdy hazard dynamiczny p-zmiennych.

Statycznym hazardem funkcyjnym układu przełączającego realizującego funkcję F nazywa się każdy hazard funkcyjny wywołany zmianą stanu wejścia A na stan wejścia A'' , przy czym $f(A) = f(A'')$.

Dynamicznym hazardem funkcyjnym układu przełączającego realizującego funkcję F nazywa się każdy hazard funkcyjny wywołany zmianą stanu wejścia A na stan wejścia A'' , przy czym $f(A) \neq f(A'')$.

Hazardem wielokrotnym nazywa się każdy hazard, który może się objawić na wyjściu kombinacyjnego układu przełączającego wielokrotną ilością zmian wartości stanu wyjścia tego układu.

Hazardem istotnym nazywa się hazard sekwencyjny wywołany wyścigiem pomiędzy zmianą stanu wejścia A na sąsiedni stan wejścia A' a zmianą stanu wewnętrznego wywołanego tą zmianą.

Hazardem istotnym p-zmiennych nazywa się hazard sekwencyjny wywołany wyścigiem pomiędzy zmianą stanu wejścia A na stan wejścia A różniący się p-stanami logicznymi, a zmianą stanu wewnętrznego wywołanego tą zmianą.

Hazardem istotnym wielokrotnego stopnia nazywa się hazard sekwencyjny wywołany wyścigiem pomiędzy zmianą stanu wejścia A na sąsiedni stan wejścia A' a ostatnią ze zmian z cyklu kolejno wielokrotnie po sobie następujących zmian stanów wewnętrznych, z których każda zmiana następna spowodowana została zmianą poprzednią, przy czym pierwsza zmiana cyklu wywołana została zmianą stanu wejścia A na sąsiedni stan wejścia A' .

Hazardem istotnym wielokrotnego stopnia p-zmiennych nazywa się hazard sekwencyjny wywołany wyścigiem pomiędzy zmianą stanu wejścia A na stan wejścia A" różniący się p-stanami logicznymi, a ostatnią ze zmian z cyklu kolejno wielokrotnie po sobie następujących zmian stanów wewnętrznych, z których każda zmiana następną spowodowana została zmianą poprzednią, przy czym pierwsza zmiana cyklu wywołana została zmianą stanu wejścia A na stan wejścia A".

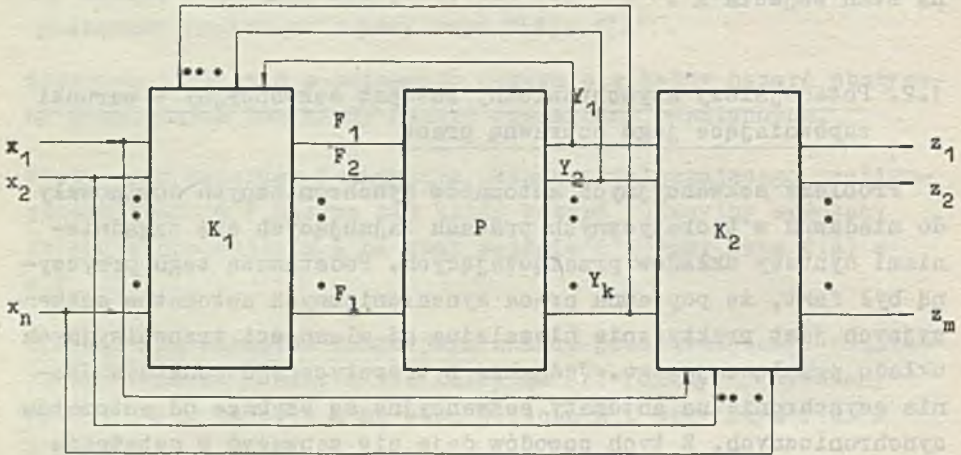
1.2. Potencjałowy asynchroniczny automat sekwencyjny i warunki zapewniające jego poprawną pracę

Problemy sekwencyjnych automatów synchronicznych dominowały do niedawna w teoretycznych pracach zajmujących się zagadnieniami syntezy układów przełączających. Podstawową tego przyczyną był fakt, że poprawna praca synchronicznych automatów sekwencyjnych jest praktycznie niezależna od własności transmisyjnych układu przełączającego. Jednakże z teoretycznego punktu widzenia asynchroniczne automaty sekwencyjne są szybsze od automatów synchronicznych. Z tych powodów daje się zauważyć w ostatnich latach bardzo poważny rozwój prac teoretycznych związanych z syntezą asynchronicznych automatów sekwencyjnych [1], [6], [14], [20], [22], [25], [27], [29], [36], [40], [66], [67], których rezultaty pozwalają obecnie na praktyczne usuwanie możliwości szkodliwego wpływu hazardu na poprawną pracę takich automatów.

Ogólny schemat blokowy^{*)} potencjałowego asynchronicznego automatu sekwencyjnego przedstawiony jest na rys. 2, gdzie dwa bloki K_1 i K_2 stanowią wyodrębnioną część kombinacyjną tego automatu, a blok P jest układem pamięciowym tego automatu. W automacie tym wektor $\mathbf{X} = (x_1, x_2, \dots, x_n)$ określa zmienne wejściowe, wektor $\mathbf{Z} = (z_1, z_2, \dots, z_m)$ określa zmienne wyjściowe automatu, wektor $\mathbf{F} = (F_1, F_2, \dots, F_1)$ określa zmienne stanu wzbudzeń układu pamięciowego oraz wektor

*) Schemat blokowy i jego analizę opracowano opierając się na pracy [64]

$Y = (Y_1, Y_2, \dots, Y_k)$ określa zmienne stanu wewnętrznego automatu. Ponieważ rozpatrywany asynchroniczny automat sekwencyjny jest automatem potencjałowym, więc stan wewnętrzny jest określony wartością sygnałów wzbudzenia. Rytm pracy tego automatu zależy od wielkości odstępów czasowych w jakich następują zmiany stanów wejścia automatu, jak również od właściwości transmisyjnych układu kombinacyjnego K_1 .



Rys. 2. Ogólny schemat blokowy asynchronicznego automatu sekwencyjnego

Opierając się na przedstawionym (rys. 2) modelu asynchronicznego automatu sekwencyjnego i przyjmując, choć nie ma to znaczenia dla ogólnych rozważań, że układy K_1 , K_2 i P wnoszą opóźnienia d_1 , d_2 i d_p , można ustalić następujące czasowe zależności:

$$\begin{aligned}
 F(\tau + d_1) &= \alpha[X(\tau), Y(\tau)] \\
 Y(\tau + d_p) &= \beta[F(\tau)], \\
 Y(\tau + d_1 + d_p) &= \gamma[X(\tau), Y(\tau)] \\
 Z(\tau + d_2) &= \delta[X(\tau), Y(\tau)]
 \end{aligned} \tag{1}$$

Ponieważ opóźnienie d_2 nie ma istotnego znaczenia powyższe zależności po podstawieniu $d = d_1 + d_p$ przyjmują następującą postać:

$$Y(\tau + d) = \gamma [X(\tau), Y(\tau)] \quad (2)$$

$$Z(\tau) = \delta [X(\tau), Y(\tau)] \quad (3)$$

W zależności od rodzaju zastosowanych elementów pamięciowych istnieją dwa typy potencjałowych asynchronicznych automatów sekwencyjnych:

- automaty z pętlami sprzężeń zwrotnych
- automaty z przerzutnikami

W pierwszym przypadku rolę zastępczą elementu pamięciowego spełnia opóźnienie. Natomiast w przypadku drugim elementem pamięci jest przerzutnik. Ponieważ dla poprawnej pracy opisywanego automatu niezbędne jest opóźnienie $d = d_1 + d_p$, więc w przypadku automatu z pętlami logicznych sprzężeń zwrotnych powinny być stosowane specjalne elementy opóźniające zastępujące opóźnienie d_p . Ponieważ wielkość bezwzględna opóźnienia d nie odgrywa roli, więc w praktyce nie buduje się wspomnianych elementów opóźniających. Rolę opóźnienia d spełnia w tym wypadku opóźnienie d_1 , a pamięć tkwi w pętlach logicznych sprzężeń zwrotnych.

Przy założeniu $d_p = 0$ zależność czasowa (1) przyjmie dla automatu z pętlami logicznych sprzężeń zwrotnych następującą postać:

$$Y(\tau) = \beta [F(\tau)]$$

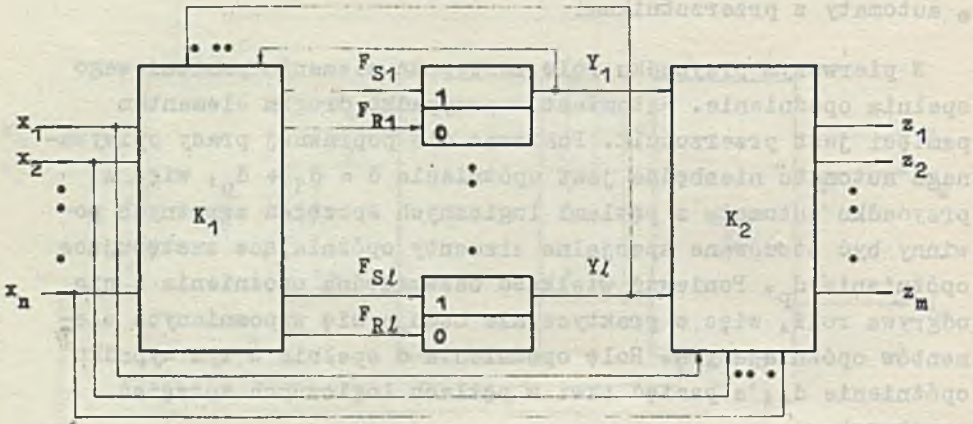
Rys. 3 przedstawia schemat blokowy potencjałowego asynchronicznego automatu sekwencyjnego z blokiem pamięci w postaci przerzutników. Przerzutnikami w tym bloku pamięci są najczęściej dwustanowe przerzutniki z dominującym wejściem wpisującym realizujące funkcje pamięci o postaci:

$$Y_1 = F_{S_1} + \bar{F}_{R_1} \cdot Y_1 \quad (4)$$

lub przerzutniki z dominującym wejściem zerującym realizujące funkcję pamięci o postaci

$$Y_1 = \bar{F}_{R_1} \cdot (F_{S_1} + Y_1) \quad (5)$$

gdzie: Y_i - stan aktualny przerzutnika,
 F_{Si} - funkcja wpisująca przerzutnika,
 Y_{i-1} - stan poprzedni przerzutnika,
 F_{Ri} - funkcja zerująca przerzutnika
 oraz $i \in [1, l]$



Rys. 3. Schemat blokowy asynchronicznego automatu sekwencyjnego z przerzutnikami

Zarówno w przypadku automatu z pętlami logicznych sprzężeń zwrotnych jak i automatu z przerzutnikami prawidłowa praca uzależniona jest od rzeczywistych charakterystyk elementów logicznych i torów sygnałowych układu kombinacyjnego K_1 .

Praca Huffmana [18], a przede wszystkim dalsze jej rozwinięcie w pracy Ungera [66], przedstawia warunki jakie powinny być spełnione, aby na wyjściach potencjałowego asynchronicznego automatu sekwencyjnego nie występowały przekłamania przetwarzanej informacji niezgodne z algorytmem działania zaprojektowanego automatu lub innymi słowy, aby prawidłowość pracy takiego automatu była niezależna od własności transmisyjnych układu kombinacyjnego K_1 .

Warunki zapewniające poprawną pracę potencjałowego asynchronicznego automatu sekwencyjnego:

- a) układ kombinacyjny K_1 nie ma hazardu podstawowego,
- b) zmiana stanów wejścia ograniczona jest wyłącznie do zmiany na sąsiednie stany wejścia,
- c) odległość czasowa pomiędzy kolejnymi zmianami stanu wejścia jest większa od najdłuższego czasu d_1 w przypadku automatu z pętlami logicznych sprzężeń zwrotnych oraz większa od najdłuższego czasu d w przypadku automatu z przerzutnikami,
- d) asynchroniczny automat sekwencyjny nie ma wyścigów krytycznych,
- e) asynchroniczny automat sekwencyjny nie ma hazardów istotnych.

Powyższe warunki są obowiązujące przy następujących założeniach:

- każdy rzeczywisty element logiczny układu przełączającego jest traktowany jako idealna bramka (bez opóźnień) z dołączonymi na każdym wejściu oraz wyjściu elementami opóźnienia
- wartość opóźnienia każdego elementu opóźnienia jest funkcją czasu t i jest zawarta w przedziale $0 \leq d(t) \leq d_{\max}$.

Warunek a, d oraz e stanowią ograniczenia dla projektowanej struktury układu przełączającego. Natomiast warunek b oraz c stanowią ograniczenia dla urządzeń zewnętrznych zasilających wejścia sekwencyjnego automatu. Te dwa ostatnie warunki nie ograniczają jednak w jakiś istotny sposób swobodnej pracy przymysłowych urządzeń zewnętrznych, gdyż prawdopodobieństwo jednoczesnej zmiany dwóch, a tym bardziej większej liczby ich wyjściowych sygnałów, zasilających wejścia sekwencyjnego automatu, jest tak znikome, a czasy przełączeń współczesnych elementów logicznych są tak małe, że praktycznie przyjmuje się, iż te warunki są spełnione automatycznie.

1.3. Przykłady szkodliwego wpływu hazardu podstawowego na poprawną pracę potencjałowego asynchronicznego automatu sekwencyjnego

Przykład A (rys. 4)

Przedstawia potencjałowy asynchroniczny automat sekwencyjny z pętlami logicznych sprzężeń zwrotnych. Jego część kombinacyjną (blok K_1 wg rys. 2) stanowi trójwarstwowy układ przełączający zbudowany z elementów logicznych NAND. Przedstawiony na rysunku sekwencyjny automat nie działa poprawnie. Powodem nieprawidłowej pracy tego automatu jest hazard podstawowy, który posiada część kombinacyjna przedstawionego automatu. Poniżej podano szczegółową analizę zachowania się tego kombinacyjnego układu przełączającego przy niektórych zmianach stanów wejścia.

Przykład szkodliwego wpływu hazardu statycznego ze względu na zmienną x_1 w warunkach działania.

Gdy stan wejścia $x_1 = 1$, $x_2 = 0$, $x_3 = 1$, $y = 1$ zasila wejścia automatu na wyjściach poszczególnych elementów NAND utrzymują się następujące stany logiczne:

wyjście elementu 7 = 1,

wyjście elementu 6 = 0,

wyjście elementu 5 = 0,

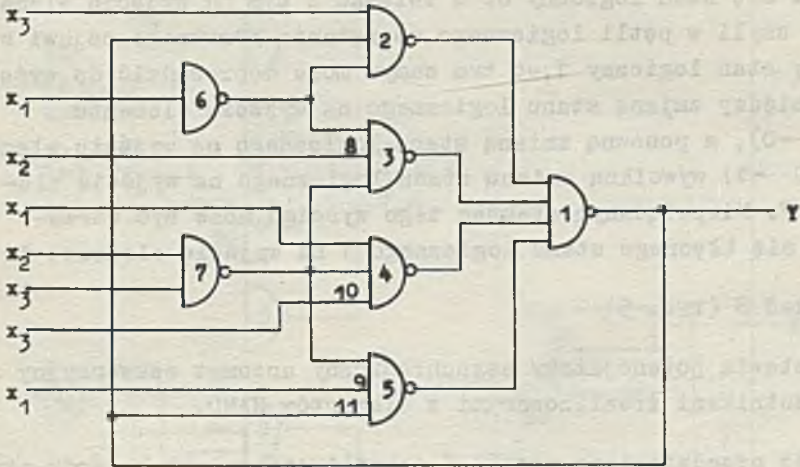
wyjście elementu 4 = 0,

wyjście elementu 3 = 1,

wyjście elementu 2 = 1,

wyjście elementu 1 = 1.

Jeżeli przedstawiony poprzednio stan wejścia zmieni się na sąsiedni stan wejścia $x_1 = 0$, $x_2 = 0$, $x_3 = 1$, $y = 1$, to wyjście elementu 4 i 5 zmieni swój stan logiczny na 1. W związku z tym również wyjście elementu 1 (pętla logicznego sprzężenia zwrotnego) zmieni swój stan logiczny na 0, co tym samym przy odpowiednio niekorzystnym rozkładzie opóźnień w analizowanym układzie przełączającym może doprowadzić do wyścigu na wyjściach elementu 2 pomiędzy zmianą stanu logicznego na wyjściu elementu 6 (0 → 1), a wspomnianą zmianą w pętli sprzężenia zwrotnego (1 → 0).



Rys. 4. Przykład potencjalowego asynchronicznego automatu sekwencyjnego z pętlami logicznych sprzężeń zwrotnych zawierającego hazard podstawowy

Wyścig ten może zakończyć się utrwaleniem na wyjściu elementu 1 błędnego stanu logicznego 0.

Przykład szkodliwego wpływu hazardu statycznego ze względu na zmienną x_2 w warunkach niedziałania.

Wejścia automatu są zasilane następującym stanem wejścia:

$x_1 = 0$, $x_2 = 0$, $x_3 = 1$, $y = 0$. Na wyjściach poszczególnych elementów NAND w tym wypadku utrzymują się następujące stany logiczne:

- wyjście elementu 7 = 1,
- wyjście elementu 6 = 1,
- wyjście elementu 5 = 1,
- wyjście elementu 4 = 1,
- wyjście elementu 3 = 1,
- wyjście elementu 2 = 1,
- wyjście elementu 1 = 0.

Pod wpływem zmiany przedstawionego stanu wejścia na stan sąsiedni wejścia $x_1 = 0$, $x_2 = 1$, $x_3 = 1$, $y = 0$ oraz na skutek

czasu opóźnienia jaki wnosi element 7, na wyjściu elementu 3 pojawi się stan logiczny 0. W związku z tym na wyjściu elementu 1, czyli w pętli logicznego sprzężenia zwrotnego pojawi się błędny stan logiczny 1, co tym samym może doprowadzić do wyścigu pomiędzy zmianą stanu logicznego na wyjściu elementu 2 ($1 \rightarrow 0$), a ponowną zmianą stanu logicznego na wyjściu elementu 3 ($0 \rightarrow 1$) wywołaną zmianą stanu logicznego na wyjściu elementu 7. Niepożądanym efektem tego wyścigu może być utrwalenie się błędnego stanu logicznego 1 na wyjściu elementu 1.

Przykład B (rys. 5)

Przedstawia potencjałowy asynchroniczny automat sekwencyjny z przerzutnikami zrealizowanymi z elementów NAND.

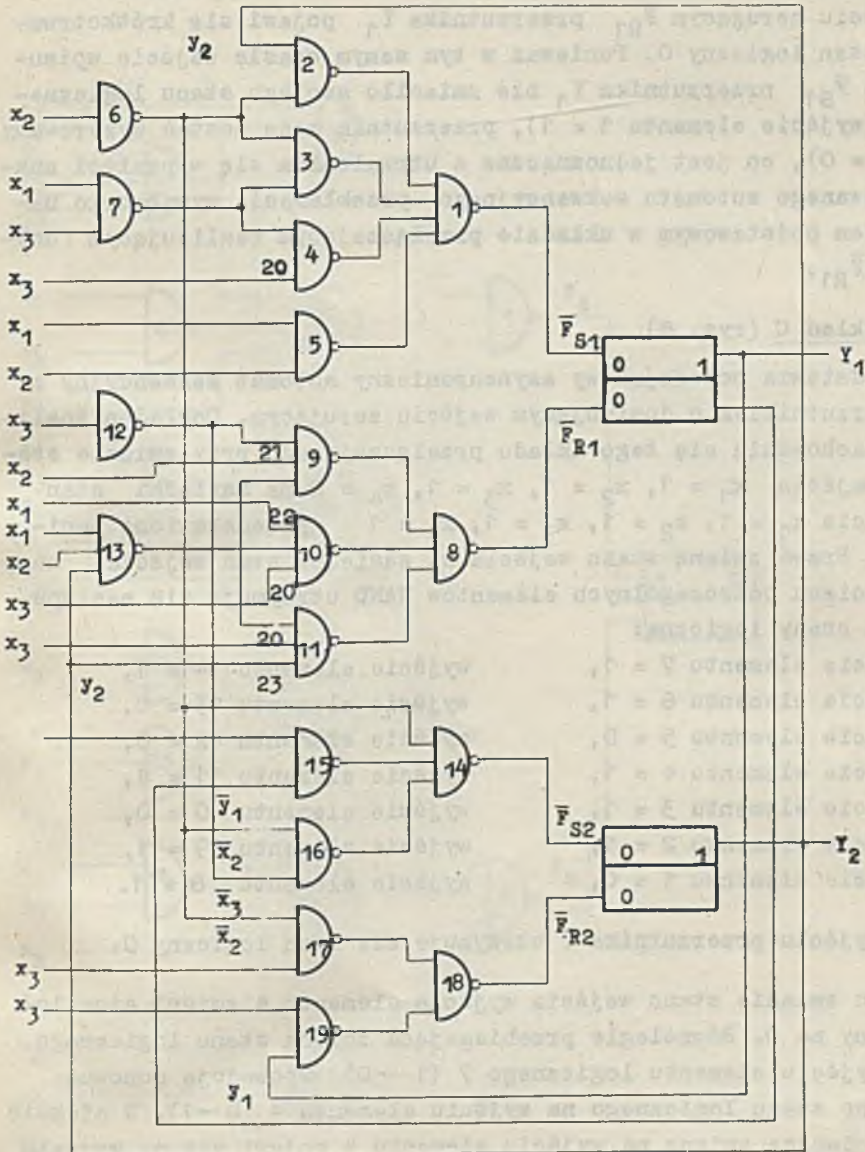
Poniżej przedstawiono przykład szkodliwego wpływu hazardu statycznego ze względu na zmienną x_3 w warunkach działania układu przełączającego realizującego funkcję \bar{F}_{R1} .

Wejścia kombinacyjnych układów przełączających realizujących funkcje \bar{F}_{S1} i \bar{F}_{R1} są zasilane następującym stanem wejścia: $x_1 = 1$, $x_2 = 1$, $x_3 = 1$, $y_2 = 0$, a przerzutnik Y_1 ma na swoim wyjściu Y_1 stan logiczny 1.

Na wyjściach poszczególnych elementów NAND utrzymują się wówczas następujące stany logiczne:

wyjście elementu 7 = 0,	wyjście elementu 13 = 1,
wyjście elementu 6 = 0,	wyjście elementu 12 = 0,
wyjście elementu 5 = 0,	wyjście elementu 11 = 1,
wyjście elementu 4 = 1,	wyjście elementu 10 = 0,
wyjście elementu 3 = 1,	wyjście elementu 9 = 1,
wyjście elementu 2 = 1,	wyjście elementu 8 = 1.
wyjście elementu 1 = 1,	

Jeżeli przedstawiony stan wejścia zmieni się na sąsiedni stan wejścia $x_1 = 1$, $x_2 = 1$, $x_3 = 0$, $y_2 = 0$ to wyjście elementu 10 zmieni swój stan logiczny na 1. A to z kolei spowoduje zmianę stanu logicznego na wyjściu elementu 8 ($1 \rightarrow 0$). Równolegle przebiegająca zmiana stanu logicznego na wyjściu elementu 12 ($0 \rightarrow 1$) spowoduje zmianę stanu logicznego na wyjściu elementu 9 ($1 \rightarrow 0$), co w efekcie stanie się przyczyną ponownej zmiany stanu lo-



Rys. 5. Przykład potencjalowego asynchronicznego sekwencyjnego układu przełączającego z przerzutnikami zrealizowanymi z elementów NAND

gicznego na wyjściu elementu 8 ($0 \rightarrow 1$). W sumie wzięwszy na wejściu zerującym \bar{F}_{R1} przerzutnika Y_1 pojawi się krótkotrwały stan logiczny 0. Ponieważ w tym samym czasie wejście wpisujące \bar{F}_{S1} przerzutnika Y_1 nie zmieniło swojego stanu logicznego (wyjście elementu 1 = 1), przerzutnik może zostać wyzerowany ($Y_1 = 0$), co jest jednoznaczne z utwaleniem się w pamięci analizowanego automatu sekwencyjnego przekłamania wywołanego hazardem podstawowym w układzie przełączającym realizującym funkcję \bar{F}_{R1} .

Przykład C (rys. 6)

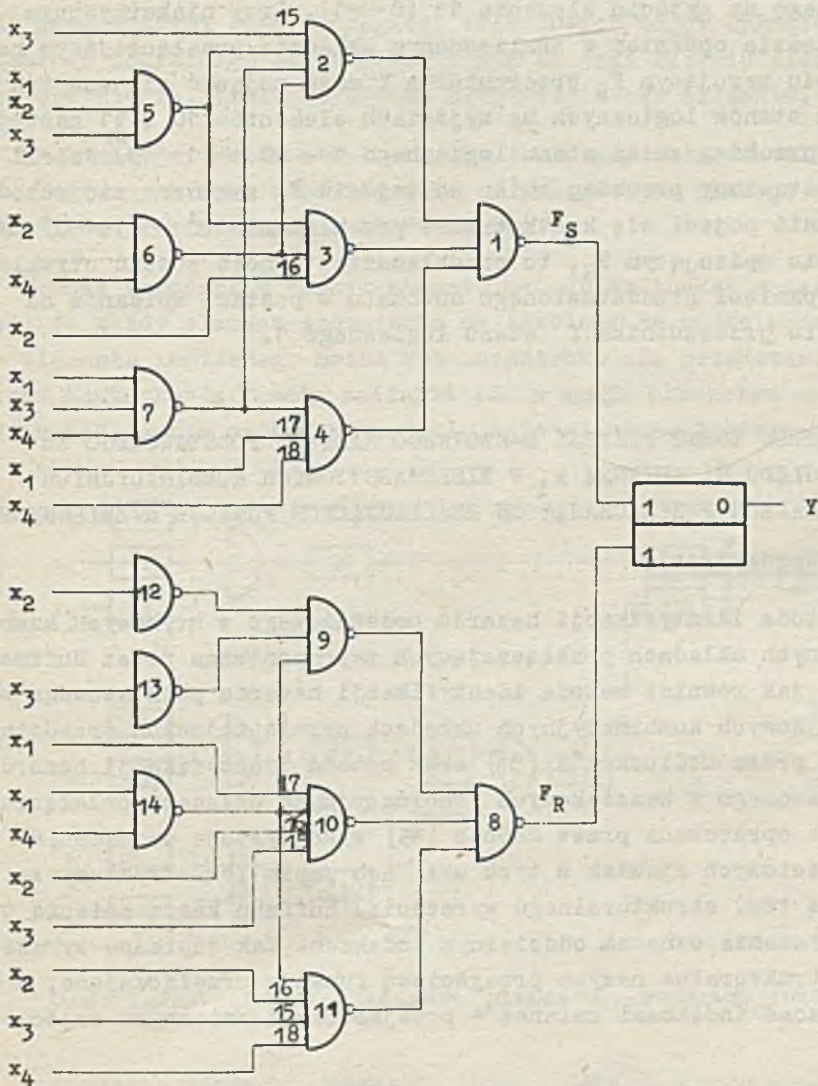
Przedstawia potencjałowy asynchroniczny automat sekwencyjny z przerzutnikiem o dominującym wejściu zerującym. Dokładną analizę zachowania się tego układu przełączającego przy zmianie stanu wejścia $x_1 = 1, x_2 = 1, x_3 = 1, x_4 = 0$ na sąsiedni stan wejścia $x_1 = 1, x_2 = 1, x_3 = 1, x_4 = 1$ przedstawiono poniżej. Przed zmianą stanu wejścia na sąsiedni stan wejścia, na wyjściach poszczególnych elementów NAND utrzymują się następujące stany logiczne:

wyjście elementu 7 = 1,	wyjście elementu 14 = 1,
wyjście elementu 6 = 1,	wyjście elementu 13 = 0,
wyjście elementu 5 = 0,	wyjście elementu 12 = 0,
wyjście elementu 4 = 1,	wyjście elementu 11 = 1,
wyjście elementu 3 = 1,	wyjście elementu 10 = 0,
wyjście elementu 2 = 1,	wyjście elementu 9 = 1,
wyjście elementu 1 = 0,	wyjście elementu 8 = 1.

Na wyjściu przerzutnika Y utrzymuje się stan logiczny 0.

Po zmianie stanu wejścia wyjście elementu 4 zmieni stan logiczny na 0. Równoległe przebiegająca zmiana stanu logicznego na wyjściu elementu logicznego 7 ($1 \rightarrow 0$) spowoduje ponowną zmianę stanu logicznego na wyjściu elementu 4 ($0 \rightarrow 1$). W efekcie ta podwójna zmiana na wyjściu elementu 4 pojawi się na wejściu wpisującym F_S przerzutnika Y w postaci krótkotrwałego stanu logicznego 1.

W tym samym czasie w układzie przełączającym realizującym funkcję F_R wyjście elementu 11 zmieni stan logiczny na 0. Jed-



Rys. 6. Przykład potencjalowego asynchronicznego automatu sekwencyjnego z przerzutnikiem o dominującym wejściu zerującym

nocześnie przebiegająca zmiana stanu logicznego na wyjściu elementu logicznego 14 ($1 \rightarrow 0$) powoduje zmianę stanu logicznego na wyjściu elementu 10 ($0 \rightarrow 1$) oraz ponowną zmianę stanu logicznego na wyjściu elementu 11 ($0 \rightarrow 1$). Przy niekorzystnym rozkładzie opóźnień w analizowanym układzie przełączającym na wejściu zerującym F_R przerzutnika Y może pojawić się skutek zmian stanów logicznych na wyjściach elementów 10 i 11 następujący przebieg zmian stanu logicznego $1 \rightarrow d0 \rightarrow d1 \rightarrow 0$.^{*)} Jeżeli przedstawiony przebieg zmian na wejściu F_R zakończy się wcześniej niż pojawi się krótkotrwałe przekłamanie ($0 \rightarrow d1 \rightarrow 0$) na wejściu wpisującym F_S , to przekłamanie to może zostać utrwalone w pamięci przedstawionego automatu w postaci wpisania na wyjściu przerzutnika Y stanu logicznego 1.

2. METODA IDENTYFIKACJI k-KROTNEGO HAZARDU PODSTAWOWEGO ZE WZGLĘDU NA ZMIENNĄ x_i W WIELOWARSTWOWYCH KOMBINACYJNYCH UKŁADACH PRZEŁĄCZAJĄCYCH REALIZUJĄCYCH FUNKCJE n-ZMIENNYCH

2.1. Wprowadzenie

Metoda identyfikacji hazardu podstawowego w stykowych kombinacyjnych układach przełączających zaproponowana przez Huffmana [18], jak również metoda identyfikacji hazardu podstawowego w bezstykowych kombinacyjnych układach przełączających przedstawiona przez McCluskey'a [31] oraz metoda identyfikacji hazardu podstawowego w bezstykowych kombinacyjnych układach przełączających opracowana przez McGhee [35] wykorzystuje do badania przejściowych zjawisk w tych układach zapis ich struktury za pomocą tzw. strukturalnego wyrażenia. Huffman każdą zmienną tego wyrażenia oznacza oddzielnym indeksem. Tak zapisane wyrażenie strukturalne nazywa przejściową funkcją przełączającą, a oznaczone indeksami zmienne - przejściowymi zmiennymi wejściowymi.

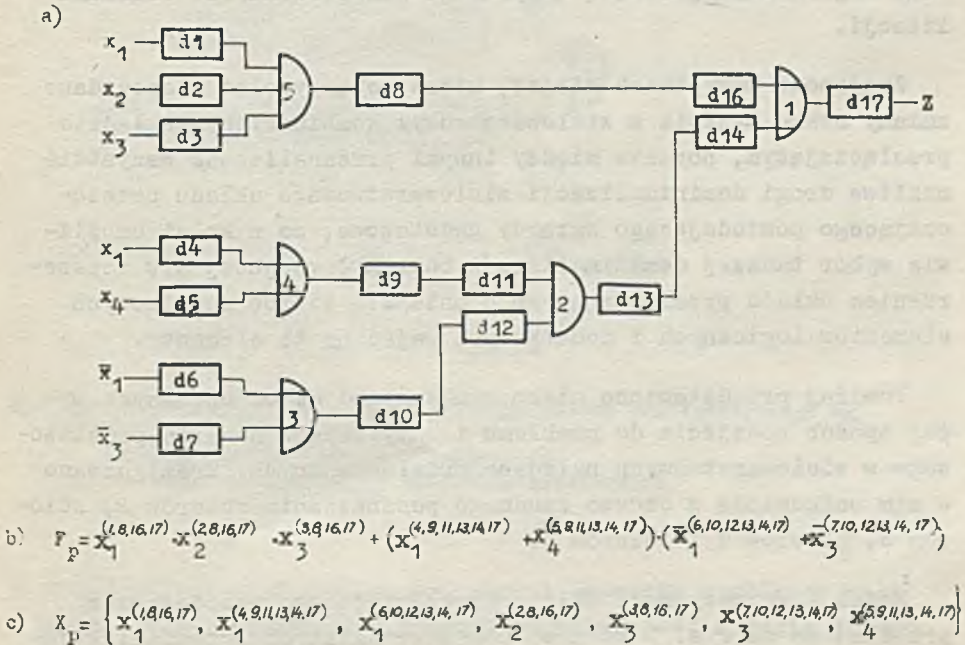
McCluskey wprowadził pewne zmiany w sposobie oznaczania indeksami zmiennych wyrażenia strukturalnego, co pozwoliło mu w korzystać metodę Huffmana do identyfikacji hazardu podstawowego w bezstykowych kombinacyjnych układach przełączających.

^{*)} Zob. definicję 2.2.15.

McCluskey zakłada w swojej metodzie, że każdy element logiczny jest reprezentowany przez idealny, wolny od czasu propagacji element, z umieszczonymi na jego wejściach elementami opóźnienia symulującymi czas propagacji wejściowych torów sygnałowych elementu logicznego oraz umieszczonym na jego wyjściu elementem opóźnienia symulującym czas propagacji tego elementu.

Kombinacyjny układ przełączający zbudowany z takich elementów logicznych oraz odpowiadającą mu przejściową funkcję przełączającą przedstawia przykładowo rys. 7.

McGhee uprościł w pewnym stopniu metodę McCluskey'a zakładając, że każdy element opóźnienia umieszczony na wyjściu dowolnego elementu logicznego można bez uszczerbku dla przedstawionej przez McCluskey'a zasady zastąpić takim samym elementem opóźnienia umieszczonym na każdym wejściu tego elementu logicznego.



Rys. 7. a) Przykład modelu kombinacyjnego układu przełączającego wg propozycji McCluskeya
 b) przejściowa funkcja przełączająca
 c) zbiór przejściowych zmiennych wejściowych

Rys. 8 przedstawia kombinacyjny układ przełączający z rys. 7 z wprowadzonymi do niego modyfikacjami McGhee. Identyfikacja hazardu podstawowego w przedstawionych na rys. 7 i rys. 8 modelach kombinacyjnych układów przełączających polega na szukaniu zbiorów P^{**} , zbiorów S^{**} , zbiorów 1^{**} oraz zbiorów O^{**} , a następnie ich analizie, w wyniku której znajduje się wszystkie hazardy podstawowe występujące w kombinacyjnym układzie przełączającym. Powyższa metoda identyfikacji hazardu podstawowego nie daje jednak pełnej informacji o tych miejscach układu przełączającego, które są bezpośrednią przyczyną istnienia hazardów podstawowych w wielowarstwowym układzie przełączającym.

W konsekwencji, w sytuacji gdy hazardy podstawowe wywierają szkodliwy wpływ na poprawną pracę automatu sekwencyjnego, prowadzi to zazwyczaj do nieoptymalnej deminimalizacji wielowarstwowego układu przełączającego, a w niektórych przypadkach brak tej informacji wręcz komplikuje przeprowadzenie deminimalizacji.

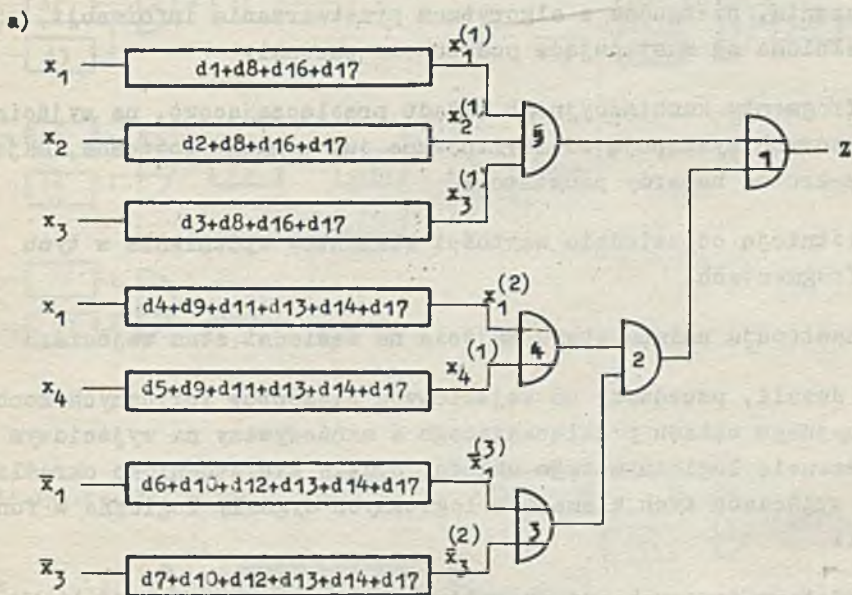
Znajomość wszystkich miejsc, które mogą wywołać niepożądane zmiany stanu wyjścia w wielowarstwowym kombinacyjnym układzie przełączającym, pozwala między innymi przeanalizować wszystkie możliwe drogi deminimalizacji wielowarstwowego układu przełączającego posiadającego hazardy podstawowe, co z kolei umożliwia wybór tańszej deminimalizacji charakteryzującej się rozszerzeniem układu przełączającego o mniejszą liczbę dodatkowych elementów logicznych i dodatkowych wejść na te elementy.

Poniżej przedstawiono nieco odmienny od metod opisanych wyżej sposób podejścia do problemu identyfikacji hazardu podstawowego w wielowarstwowym układach przełączających. Zrezygnowano w nim całkowicie z często żmudnego poszukiwania zbiorów P , zbiorów S , zbiorów 1 i zbiorów O .

Model kombinacyjnego układu przełączającego przedstawiony przykładowo na rys. 7 stanowi podstawę dalszych rozważań. Wyjś-

^{*})Zob. Przypisy do p. 2.1

cie każdego elementu logicznego tego układu traktowane jest jako odrębny sygnał logiczny, którego przebieg czasowy i wartość determinowane są:



$$b) F_p = x_1^{(1)} \cdot x_2^{(1)} \cdot x_3^{(1)} + (x_1^{(2)} + x_4^{(1)}) \cdot (\bar{x}_1^{(3)} + \bar{x}_3^{(2)})$$

$$c) X_p = \{x_1^{(1)}, x_1^{(2)}, x_1^{(3)}, x_2^{(1)}, x_3^{(1)}, x_3^{(2)}, x_4^{(1)}\}$$

Rys. 8. a) Przykład modelu kombinacyjnego układu przełączającego wg propozycji McGhee

b) przejściowa funkcja przełączająca

c) zbiór przejściowych zmiennych wejściowych

- wyrażeniem strukturalnym tej części badanego kombinacyjnego układu przełączającego, którego wyjściem jest wyjście danego elementu logicznego,

- wartościami elementów opóźnienia występujących na wejściach i wyjściach elementów logicznych tej części kombinacyjnego układu przełączającego,

- wartością zmiennych wejściowych,
- zmianami stanów wejścia na stany sąsiednie.

Tak zdeterminowane sygnały logiczne będą miały chwilowe przekłamania, niezgodne z algorytmem przetwarzania informacji, gdy spełnione są następujące podstawowe warunki:

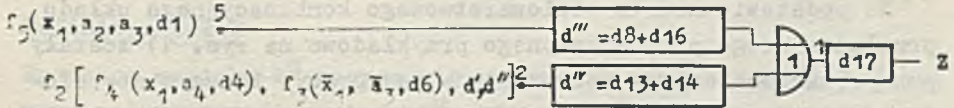
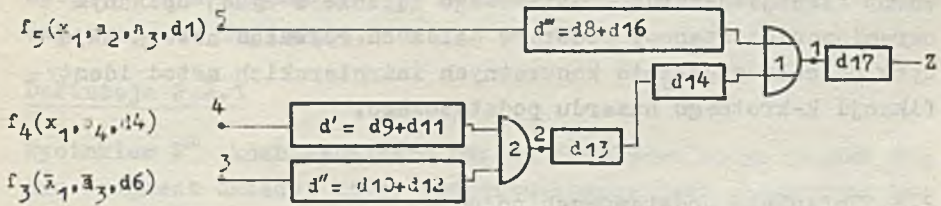
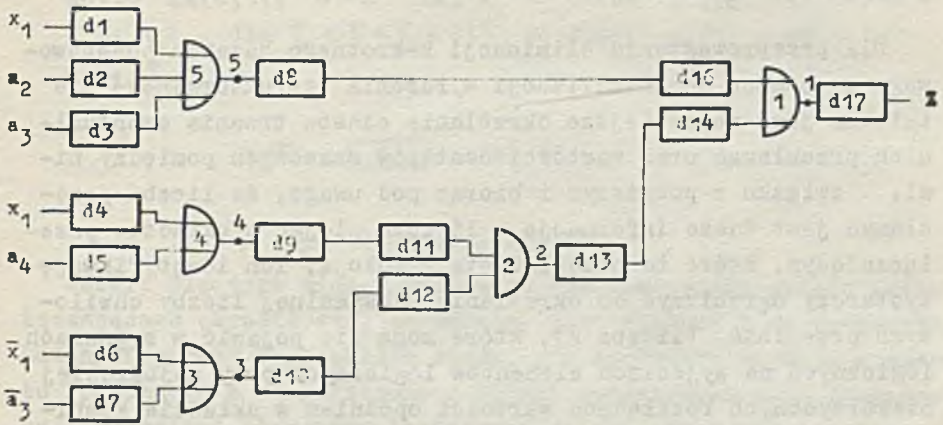
- fragmenty kombinacyjnego układu przełączającego, na wyjściach których występują zdeterminowane już sygnały logiczne, mają k-krotne hazardy podstawowe,
- istnieją odpowiednie wartości elementów opóźnienia w tych fragmentach,
- następuje zmiana stanu wejścia na sąsiedni stan wejścia.

Jeżeli, począwszy od wejściowych elementów logicznych kombinacyjnego układu przełączającego a skończywszy na wyjściowym elemencie logicznym tego układu, będzie się stopniowo określało na wyjściach tych elementów logicznych sygnały logiczne w funkcji:

- zdeterminowanych już sygnałów logicznych na ich wejściach,
- wartości elementów opóźnienia na ich wejściach,
- operacji logicznych tych elementów logicznych,

to uzyska się pełny obraz przebiegów sygnałów logicznych i ich przekłamań na wyjściach wszystkich elementów logicznych. Przeprowadzona w taki sposób analiza pozwala jednocześnie stwierdzić czy przekłamania te są wynikiem operacji wykonanych przez elementy logiczne na przesuniętych w czasie sygnałach wejściowych, czy też wynikiem przedostania się przekłamania z wejść na wyjścia elementów logicznych. Tym samym analiza taka umożliwia zidentyfikowanie hazardów podstawowych i miejsc, które je wywołują.

Rys. 9 przedstawia uproszczony przykład omówionej wyżej analizy zrealizowany na podstawie poprzednio już przedstawionego trójwarstwowego kombinacyjnego układu przełączającego przy założeniu, że na wejściu tego układu zmienia wartość na przeciwną



$$f_1 \left\{ f_2 \left[f_4(x_1, a_4, d_4), f_3(\bar{x}_1, \bar{a}_3, d_6), d', d'' \right], d_8, d_1 \right\}$$

$$f_5(x_1, a_2, a_3, d_1), d''', d'' \Bigg\}^1 \rightarrow d_{17} \rightarrow z$$

Rys. 9. Przebieg kolejnych faz proponowanej analizy wielowarstwowego kombinacyjnego układu przełączającego przedstawionego na rys. 7

zmienna x_1 , a zmienne x_2, x_3, x_4 kolejno równe są stanom logicznym a_2, a_3 oraz a_4 .

Dla przeprowadzenia eliminacji k-krotnego hazardu podstawowego za pomocą deminimalizacji wyrażenia strukturalnego nie istotne jest wcześniejsze określanie czasów trwania ewentualnych przekłamań oraz wartości odstępów czasowych pomiędzy nimi. W związku z powyższym i biorąc pod uwagę, że liczba przekłamań jest także informacją o liczbie miejsc w układzie przełączającym, które te przekłamania wywołują, ich identyfikację wystarczy ograniczyć do określania maksymalnej liczby chwilowych przekłamań (liczba k), które mogą się pojawić w sygnałach logicznych na wyjściach elementów logicznych przy najbardziej niekorzystnych rozkładach wartości opóźnień w układzie kombinacyjnym.

Przedstawione w tym podrozdziale nowe podejście do problemu identyfikacji hazardu podstawowego łącznie z wyżej opisanym ograniczeniem stanowi podstawę dalszych rozważań autora mających na celu uzyskanie konkretnych inżynierskich metod identyfikacji k-krotnego hazardu podstawowego.

2.2. Definicje podstawowych pojęć

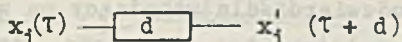
Na podstawie modelu wielowarstwowego kombinacyjnego układu przełączającego przedstawionego przykładowo na rys. 11 zostały przeprowadzone dalsze rozważania w tej pracy. Rzeczywiste elementy logiczne reprezentowane są w nim w postaci idealnych elementów logicznych E o czasie propagacji równym zero, które realizują jedną z funkcji I, LUB, NIE, NAND, NOR, z dołączonymi na ich wejściach elementami opóźnień reprezentującymi czasy propagacji wejściowych torów sygnałowych oraz dołączonymi na ich wyjściach elementami opóźnień reprezentującymi czasy propagacji rzeczywistych elementów logicznych.

Elementy opóźniania (rys. 10) w przedstawionym modelu spełniają następujące warunki:

a) wartość opóźnienia d jest funkcją czasu τ

b) $0 \leq d(\tau) \leq d_{\max}$

o) jeżeli istnieje moment τ_0 , dla którego $x_i(\tau_0 - \varepsilon) = a$, gdzie $a \in [0, 1]$, $\varepsilon > 0$ oraz $x_i(\tau_0) = \bar{a}$ i jeżeli $x_i(\tau') = x_i(\tau_0)$, dla $\tau_0 \leq \tau' \leq \tau_0 + d(\tau)$, to wtedy $x_i'(\tau_0 + d(\tau)) = x_i(\tau_0)$



Rys. 10 Element opóźnienia

Jeżeli dla tego wielowarstwowego kombinacyjnego układu przełączającego określi się, że dowolny jego element B ma podporządkowany fragment tego układu wtedy, gdy wszystkie wejścia elementu logicznego B są połączone z torami sygnałowymi tego fragmentu i wyjściami elementów logicznych należących tylko do tego fragmentu oraz rozpatrywany fragment nie ma ani jednego toru sygnałowego i elementu logicznego nie połączonego w jakikolwiek sposób z tym fragmentem, to można wprowadzić do dalszych rozważań poniższe definicje.

Definicja 2.2.1

Wycinkiem W^m kombinacyjnego układu przełączającego nazywa się ten fragment układu, który podporządkowany jest elementowi logicznemu B^m .

Definicja 2.2.2

Wyjściom wycinka W^m nazywa się wyjście każdego elementu logicznego należącego do wycinka W^m , które połączone jest z wejściem elementu logicznego B^m i każdy tor sygnałowy tego wycinka, który połączony jest z wejściem elementu logicznego B^m .

Definicja 2.2.3

Funkcją F^m nazywa się funkcję realizowaną przez układ przełączający złożony z elementu logicznego B^m i podporządkowanego mu wycinka W^m .

Definicja 2.2.4

Wyrażeniem strukturalnym S^m nazywa się wyrażenie Boole'a, które odzwierciedla strukturę układu logicznego realizującego funkcję F^m .

Definicja 2.2.5

Wyrażeniem strukturalnym S_0^m nazywa się wyrażenie Boole'a, które odzwierciedla strukturę dowolnego fragmentu o wycinka W^m , złożonego z elementu logicznego B^0 o wyjściu połączonym z wejściem elementu logicznego B^m i podporządkowanego mu wycinka W^0 lub takie, które odzwierciedla wchodzący na wejście elementu logicznego B^m niezależny tor sygnałowy dowolnej zmiennej wejściowej x_i .

Definicja 2.2.6

Wyrażenie strukturalne S_0^m nazywa się prostym, jeżeli jest zapisane za pomocą wyrażenia Boole'a w następujących beznawiasowych postaciach:

- alternatywy dowolnych zmiennych zanegowanych i niezanegowanych,
- zanegowanej alternatywy dowolnych zmiennych zanegowanych i niezanegowanych,
- koniunkcji dowolnych zmiennych zanegowanych i niezanegowanych,
- zanegowanej koniunkcji dowolnych zmiennych zanegowanych i niezanegowanych, a także w postaci dowolnej zanegowanej lub niezanegowanej zmiennej.

Przykład:

$$x_1, \bar{x}_2, \bar{x}_1 \cdot x_2 \cdot x_3, x_2 + \bar{x}_3 + x_4, \overline{x_4 \cdot x_5 \cdot x_6}, \overline{\bar{x}_1 + x_4}$$

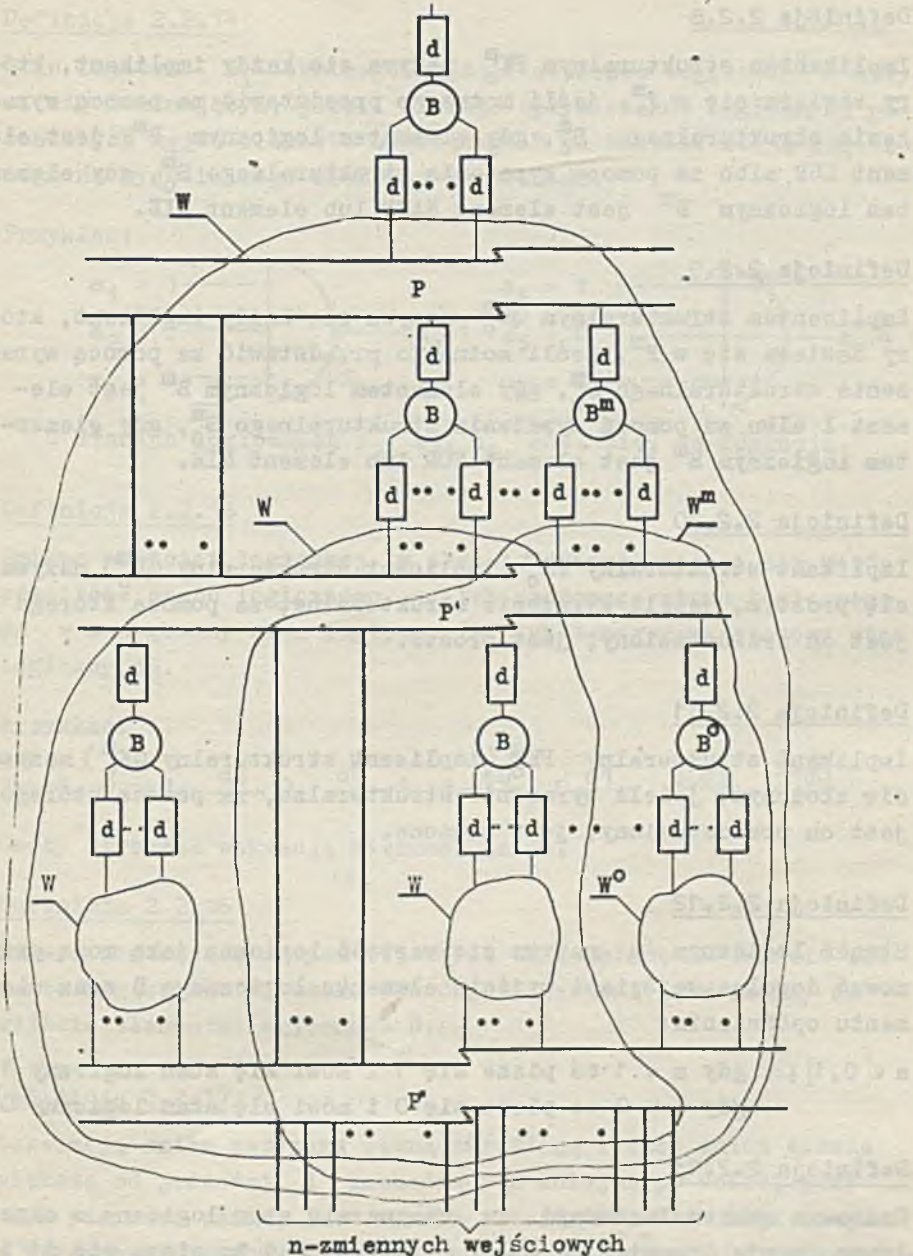
Wyrażenia strukturalne $(\bar{x}_1 \cdot x_2) \cdot x_3, (x_2 \cdot \bar{x}_3) + x_4, x_4 \cdot (\bar{x}_5 \cdot x_6)$ nie są prostymi wyrażeniami strukturalnymi S_0^m .

Definicja 2.2.7

Wyrażenie strukturalne S_0^m , które nie jest proste, będziemy nazywali złożonym.

Przykład:

$$(\bar{x}_1 \cdot x_2) \cdot x_3, (\bar{x}_1 \cdot x_2) + x_3, \overline{\overline{x_4 \cdot \bar{x}_5} + x_6}, x_1 + [x_2 \cdot (x_3 + \bar{x}_4)]$$



n -zmiennych wejściowych

Rys. 11. Wielowarstwowy kombinacyjny układ przełączający realizujący funkcję n -zmiennych z wyszczególnionym m -wycinkiem podporządkowanym elementowi B^m (P , P' , P'' przedstawiają pewne dowolne zbiory połączeń wewnętrznych)

Definicja 2.2.8

Implikantem strukturalnym PK_0^m nazywa się każdy implikant, który zawiera się w F^m , jeśli można go przedstawić za pomocą wyrażenia strukturalnego S_0^m , gdy elementem logicznym B^m jest element LUB albo za pomocą wyrażenia strukturalnego $\overline{S_0^m}$, gdy elementem logicznym B^m jest element NAND lub element NIE.

Definicja 2.2.9

Implicentem strukturalnym QK_0^m nazywa się każdy implicent, który zawiera się w F^m , jeśli można go przedstawić za pomocą wyrażenia strukturalnego S_0^m , gdy elementem logicznym B^m jest element I albo za pomocą wyrażenia strukturalnego $\overline{S_0^m}$, gdy elementem logicznym B^m jest element NOR lub element NIE.

Definicja 2.2.10

Implikant strukturalny PK_0^m (implicent strukturalny QK_0^m) nazywa się prostym, jeżeli wyrażenie strukturalne, za pomocą którego jest on przedstawiony, jest proste.

Definicja 2.2.11

Implikant strukturalny PK_0^m (implicent strukturalny QK_0^m) nazywa się złożonym, jeżeli wyrażenie strukturalne, za pomocą którego jest on przedstawiony, jest złożone.

Definicja 2.2.12

Stanem logicznym a nazywa się wartość logiczną jaką mogą przyjmować dowolne wejścia i wyjścia elementu logicznego B oraz elementu opóźnienia:

$a \in [0,1]$; gdy $a = 1$ to pisze się 1 i mówi się stan logiczny 1,
 gdy $a = 0$ to pisze się 0 i mówi się stan logiczny 0.

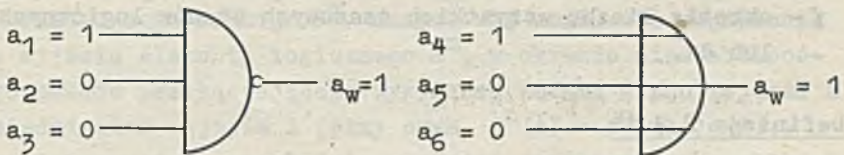
Definicja 2.2.13

Czasowym stanem logicznym da nazywa się stan logiczny o określonym czasie trwania; $da \in [0,1]$; gdy $da = 1$ to pisze się d1 i mówi się czasowy stan logiczny d1, gdy $da = 0$ to pisze się d0 i mówi się czasowy stan logiczny d0.

Definicja 2.2.14

Stan logiczny a_1 dowolnego wejścia elementu logicznego B nazywa się dominującym, jeżeli blokuje wpływ stanów logicznych pozostałych wejść elementu logicznego B na wartość stanu logicznego na wyjściu tego elementu logicznego.

Przykład:

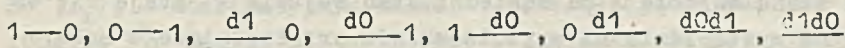


O stanach logicznych a_2, a_3, a_4 mówi się, że dominują.

Definicja 2.2.15

Zmianą wartości logicznej (w skrócie zmianą) nazywa się każde przejście stanu logicznego a lub czasowego stanu logicznego da w zanegowany stan logiczny \bar{a} lub zanegowany czasowy stan logiczny $d\bar{a}$.

Przykład:



Groty strzałek wskazują kierunek zmian.

Definicja 2.2.16

Zmianą dominującą nazywa się każdą taką zmianę na wejściu elementu logicznego B, która powoduje pojawienie się zmiany na wyjściu elementu logicznego B.

Definicja 2.2.17

Sekwencją zmian nazywamy pewną określoną liczbę zmian zawsze większą od jedności i zachodzących kolejno po sobie, przy czym pierwsza zmiana jest zawsze przejściem stanu logicznego a w zanegowany czasowy stan logiczny $d\bar{a}$, natomiast ostatnia zmiana jest zawsze przejściem czasowego stanu logicznego da w zanegowany stan logiczny \bar{a} .

Przykład:

$$\begin{array}{l} \underset{1}{\underbrace{d0^1 d1^{\dots} \cdot d1 d0^\alpha}}_1, \quad \underset{0}{\underbrace{d1^1 d0^{\dots} \cdot d0 d1^\beta}}_0, \\ \underset{1}{\underbrace{d0 d1^1 \dots \cdot d1^1}}_0, \quad \underset{0}{\underbrace{d1 d0^1 \dots \cdot d0^1}}_1 \end{array}$$

α - określa liczbę wszystkich czasowych stanów logicznych $d0$,
 β - określa liczbę wszystkich czasowych stanów logicznych $d1$,
 γ - określa liczbę wszystkich czasowych stanów logicznych $d0$
lub $d1$.

Definicja 2.2.18

Stan wejścia A^* jest pokryty przez strukturalny implikant PK_0^m (strukturalny implicent QK_0^m), jeżeli jest pokryty $^*)$ przynajmniej przez jeden stabilny zbiór 1^* (stabilny zbiór 0^*) wyrażenia strukturalnego implikanta PK_0^m (implicenta QK_0^m).

Definicja 2.2.19

Rozkład wartości opóźnień $d(\tau)$ w kombinacyjnym układzie przełączającym nazywa się niekorzystnym, jeżeli na skutek istnienia hazardu logicznego w tym układzie przełączającym i w efekcie zmiany stanu wejścia A na sąsiedni stan wejścia A' staje się on przyczyną wystąpienia na wyjściu układu przełączającego maksymalnej liczby (spośród wszystkich możliwych) niepożądanych czasowych stanów logicznych.

Definicja 2.2.20

k -krotnym hazardem statycznym w warunkach działania ze względu na zmienną x_i , $HS1^k(x_i)$, w kombinacyjnym układzie przełączającym realizującym funkcje F^m nazywa się możliwość pojawienia się na wyjściu elementu logicznego F^m , w okresie niestabilności tego układu przełączającego wywołanej zmianą stanu wejścia A na sąsiedni stan wejścia A' (przez czym $f^m(A) = f^m(A') = 1$) i przy niekorzystnym rozkładzie wartości opóźnień w tym układzie, sekwencji zmian $\underset{1}{\underbrace{d0^1 d1^{\dots} \cdot d1 d0^k}}_1$. O kombinacyjnym układzie przełączającym realizującym funkcję F^m będzie się mówić w

$^*)$ Zob. Przypisy do p. 2.2

tym przypadku, że posiada $HS1^k(x_1)$. O implikancie strukturalnym PK_0^m , który pokrywa parę sąsiednich stanów wejścia A i A' będzie się mówić w tym przypadku, że posiada $HS1^k(x_1)$.

Definicja 2.2.21

k-krotnym hazardem statycznym w warunkach niedziałania ze względu na zmienną x_1 , $HSO^k(x_1)$, w kombinacyjnym układzie przełączającym realizującym funkcję F^m nazywa się możliwość pojawienia się na wyjściu elementu logicznego B^m , w okresie niestabilności tego układu przełączającego wywołanej zmianą stanu wejścia A na sąsiedni stan wejścia A' (przy czym $f^m(A) = f^m(A') = 0$) i przy niekorzystnym rozkładzie wartości opóźnień w tym układzie, sekwencji zmian $0 \frac{d1^1d0 \dots d0d1^k}{0}$. O kombinacyjnym układzie przełączającym realizującym funkcję F^m będzie się mówić w tym przypadku, że posiada $HSO^k(x_1)$. O implikancie strukturalnym QK_0^m , który pokrywa parę sąsiednich stanów wejścia A i A' będzie się w tym przypadku mówić, że posiada $HSO^k(x_1)$.

Definicja 2.2.22

k-krotnym hazardem dynamicznym ze względu na zmienną x_1 , $HD^k(x_1)$, w kombinacyjnym układzie przełączającym realizującym funkcję F^m nazywa się możliwość pojawienia się na wyjściu elementu logicznego B^m , w okresie niestabilności tego układu przełączającego wywołanej zmianą stanu wejścia A na sąsiedni stan wejścia A' (przy czym $f^m(A) \neq f^m(A')$) i przy niekorzystnym rozkładzie opóźnień w tym układzie, sekwencji zmian $1 \frac{d0d1^1 \dots d1^k}{0}$ lub sekwencji zmian $0 \frac{d1d0^1 \dots d0^k}{1}$. O kombinacyjnym układzie przełączającym realizującym funkcję F^m będzie się mówić w tym przypadku, że posiada $HD^k(x_1)$. O implikancie strukturalnym PK_0^m (implikancie strukturalnym QK_0^m), który pokrywa stan wejścia A lub A' będzie się w tym wypadku mówić, że posiada $HD^k(x_1)$.

Definicja 2.2.23

O kombinacyjnym układzie przełączającym, który posiada $HS1^k(x_1)$ lub $HSO^k(x_1)$ lub $HD^k(x_1)$ mówi się, że posiada k -krotny hazard podstawowy $HP^k(x_1)$.

Definicja 2.2.24

Hazardowym ze względu na zmienną x_1 strukturalnym implikantem (strukturalnym implicentem) nazywa się każdy implikant strukturalny (implicent strukturalny), który posiada $HP^k(x_1)$.

Definicja 2.2.25

Bezhazardowym, ze względu na zmienną x_1 , strukturalnym implikantem (strukturalnym implicentem) nazywa się każdy implikant strukturalny (implicent strukturalny), który nie posiada $HP^k(x_1)$.

Definicja 2.2.26

Dwa sąsiednie stany wejścia A i A' są pokryte przez bezhazardowy, ze względu na zmienną x_1 , strukturalny implikant (strukturalny implicent), jeżeli są pokryte przez przynajmniej jeden stabilny zbiór 1^* (stabilny zbiór 0^*) wyrażenia strukturalnego tego implikanta (implicenta).

2.3. Podstawowe lematy

W celu ułatwienia czytelnikowi zrozumienia zawartych w podrozdziale 2.4 twierdzeń i ich dowodów, poniżej przedstawiono lematy, które w sposób oczywisty wynikają z definicji sumy logicznej, iloczynu logicznego i prawa De Morgana.

Lemat 2.3.1

Sekwencja zmian $1 \underline{d0^1 d1 \dots d1 d0^k}$ 1 wystąpi na pewno na wyjściu idealnego elementu logicznego LUB, gdy spełnione są na jego wejściach następujące warunki konieczne:

*) Zob. Przypisy do p. 2.2

- I. na żadnym wejściu nie występuje stan logiczny 1,
- II. co najmniej na jednym wejściu występuje sekwencja zmian $1 \underline{d0^1 d1^{1..} \cdot d1 d0^k} \rightarrow 1$ lub co najmniej na jednym wejściu występuje sekwencja zmian $1 \underline{d0 d1^{1..} \cdot d1^r} \rightarrow 0$ albo zmiana $1 \rightarrow 0$ i co najmniej na jednym wejściu występuje sekwencja zmian $0 \underline{d1 d0^1 \cdot \cdot \cdot d0^r} \rightarrow 1$ albo zmiana $0 \rightarrow 1$.
- III. suma wszystkich zmian dominujących $\underline{d0 d1}$ i $0 \underline{d1}$ dowolnych sekwencji zmian występujących na wejściach elementu LUB równa jest $k - 1$ oraz spełnione są na jego wejściach następujące warunki wystarczające:
- ostatnia w czasie ze zmian $1 \rightarrow 0$ i $1 \underline{d0}$ występuje wcześniej od pierwszej w czasie ze zmian $0 \underline{d1}$ i $\underline{d0 d1}$
 - pierwsza w czasie ze zmian $\underline{d1 d0}$ i $\underline{d1} \rightarrow 0$ występuje wcześniej niż $k-2$ zmian dominujących $0 \underline{d1}$ i $\underline{d0 d1}$,
- druga w czasie ze zmian $\underline{d1 d0}$ i $\underline{d1} \rightarrow 0$ występuje wcześniej niż $k-3$ zmian dominujących $0 \underline{d1}$ i $\underline{d0 d1}$,
:
:
- $k-2$ w czasie ze zmian $\underline{d1 d0}$ i $\underline{d1} \rightarrow 0$ występuje wcześniej niż ostatnia ze zmian dominujących $0 \underline{d1}$ i $\underline{d0 d1}$,
 - ostatnia w czasie ze zmian $\underline{d1} \rightarrow 0$ i $\underline{d1 d0}$ występuje wcześniej niż pierwsza w czasie ze zmian $0 \rightarrow 1$ i $\underline{d0} \rightarrow 1$

Dowód:

Lemat wynika z definicji sumy logicznej.

Lemat 2.3.2

Sekwencja zmian $0 \underline{d1^1 d0^{1..} \cdot d0 d1^k} \rightarrow 0$ wystąpi na pewno na wyjściu idealnego elementu logicznego LUB, gdy spełnione są na jego wejściach następujące warunki konieczne:

- I. na żadnym wejściu nie występuje stan logiczny 1, zmiana $0 \rightarrow 1$, zmiana $1 \rightarrow 0$, sekwencja zmian $1 \xrightarrow{d0^1 d1^1 \dots d1^k} 1$, sekwencja zmian $1 \xrightarrow{d0 d1^1 \dots d1^r} 0$ oraz sekwencja zmian $0 \xrightarrow{d1 d0^1 \dots d0^r} 1$.
- II. co najmniej na jednym wejściu występuje sekwencja zmian $0 \xrightarrow{d1^1 d0^1 \dots d0 d1^p} 0$.
- III. suma wszystkich zmian dominujących $\xrightarrow{d0 d1}$ i $0 \xrightarrow{d1}$ dowolnych niezabronionych sekwencji zmian występujących na wejściach elementu LUB równa jest k oraz spełnione są na jego wejściach następujące warunki wystarczające:
- pierwsza w czasie ze zmian $\xrightarrow{d1 d0}$ i $\xrightarrow{d1} 0$ występuje wcześniej niż k-1 zmian dominujących $0 \xrightarrow{d1}$ i $\xrightarrow{d0 d1}$,
 - druga w czasie ze zmian $\xrightarrow{d1 d0}$ i $\xrightarrow{d1} 0$ występuje wcześniej niż k-2 zmian dominujących $0 \xrightarrow{d1}$ i $\xrightarrow{d0 d1}$,
 - k-1 w czasie ze zmian $\xrightarrow{d1 d0}$ i $\xrightarrow{d1} 0$ występuje wcześniej niż ostatnia ze zmian dominujących $0 \xrightarrow{d1}$ i $\xrightarrow{d0 d1}$

Dowód:

Lemat wynika z definicji sumy logicznej.

Lemat 2.3.3

Sekwencja zmian $1 \xrightarrow{d0 d1^1 \dots d1^k} 0$ ($0 \xrightarrow{d1 d0^1 \dots d0^k} 1$) wystąpi na pewno na wyjściu idealnego elementu logicznego LUB, gdy spełnione są na jego wejściach następujące warunki konieczne:

- I. na żadnym wejściu nie występuje stan logiczny 1, zmiana $0 \rightarrow 1$ ($1 \rightarrow 0$), sekwencja zmian $1 \xrightarrow{d0^1 d1^1 \dots d1 d0^k} 1$ oraz sekwencja zmian $0 \xrightarrow{d1 d0^1 \dots d0^r} 1$ ($1 \xrightarrow{d0 d1^1 \dots d1^r} 0$),
- II. co najmniej na jednym wejściu występuje sekwencja zmian $1 \xrightarrow{d0 d1^1 \dots d1^r} 0$ ($0 \xrightarrow{d1 d0^1 \dots d0^r} 1$) lub występuje zmiana $1 \rightarrow 0$ ($0 \rightarrow 1$),

III. suma wszystkich zmian dominujących $\frac{d0d1}{1}$ ($\frac{d0d1}{1}$ i $0 \frac{d1}{1}$) dowolnych niezabronionych sekwencji zmian występujących na wejściach elementu LUB równa jest k oraz spełnione są na jego wejściach następujące warunki wystarczające:

a) (obowiązuje tylko dla sekwencji zmian $1 \frac{d0d1^1 \dots d1^k}{1}$) ostatnia w czasie ze zmian $1 \rightarrow 0$ i $1 \frac{d0}{1}$ występuje wcześniej od pierwszej w czasie ze zmian $0 \frac{d1}{1}$ i $\frac{d0d1}{1}$,

b) - pierwsza w czasie ze zmian $\frac{d1d0}{1}$ i $\frac{d1}{1} \rightarrow 0$ występuje wcześniej niż $k-1$ zmian dominujących $0 \frac{d1}{1}$ i $\frac{d0d1}{1}$,

- druga w czasie ze zmian $\frac{d1d0}{1}$ i $\frac{d1}{1} \rightarrow 0$ występuje wcześniej niż $k-2$ zmian dominujących $0 \frac{d1}{1}$ i $\frac{d0d1}{1}$,

⋮

- $k-1$ w czasie ze zmian $\frac{d1d0}{1}$ i $\frac{d1}{1} \rightarrow 0$ występuje wcześniej niż ostatnia ze zmian dominujących $0 \frac{d1}{1}$ i $\frac{d0d1}{1}$,

c) (obowiązuje tylko dla sekwencji zmian $0 \frac{d1d0^1 \dots d0^k}{1}$) ostatnia w czasie ze zmian $\frac{d1}{1} \rightarrow 0$ i $\frac{d1d0}{1}$ występuje wcześniej niż pierwsza w czasie ze zmian $0 \rightarrow 1$ i $\frac{d0}{1}$.

Dowód:

Lemat wynika z definicji sumy logicznej.

Lemat 2.3.4.

Sekwencja zmian $1 \frac{d0^1 d1 \dots d1 d0^k}{1}$ wystąpi na pewno na wyjściu idealnego elementu logicznego I, gdy spełnione są na jego wejściach następujące warunki konieczne:

I. na żadnym wejściu nie występuje stan logiczny 0, zmiana $1 \rightarrow 0$ zmiana $0 \rightarrow 1$, sekwencja zmian $0 \frac{d1^1 d0 \dots d0 d1^k}{1}$, sekwencja zmian $1 \frac{d0 d1^1 \dots d1^r}{1} \rightarrow 0$ oraz sekwencja zmian $0 \frac{d1 d0^1 \dots d0^r}{1}$,

II. co najmniej na jednym wejściu występuje sekwencja zmian $1 \frac{d0^1 d1 \dots d1 d0^k}{1}$,

III. suma wszystkich zmian dominujących $\frac{d1d0}{1}$ i $1 \frac{d0}{d0}$ dowolnych niezabronionych sekwencji zmian występujących na wejściach elementu logicznego I równa jest k oraz spełnione są na jego wejściach następujące warunki wystarczające:

- pierwsza w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{d0} \rightarrow 1$ występuje wcześniej niż k-1 zmian dominujących $1 \frac{d0}{d0}$ i $\frac{d1d0}{1}$,
- druga w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{d0} \rightarrow 1$ występuje wcześniej niż k-2 zmian dominujących $1 \frac{d0}{d0}$ i $\frac{d1d0}{1}$,
- ⋮
- k-1 w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{d0} \rightarrow 1$ występuje wcześniej niż ostatnia ze zmian dominujących $1 \frac{d0}{d0}$ i $\frac{d1d0}{1}$.

Dowód:

Lemat wynika z definicji iloczynu logicznego.

Lemat 2.3.5

Sekwencja zmian $0 \frac{d1^1 d0 \dots d0 d1^k}{1} \rightarrow 0$ wystąpi na pewno na wyjściu idealnego elementu logicznego I, gdy spełnione są na jego wejściach następujące warunki konieczne:

- I. na żadnym wejściu nie występuje stan logiczny 0,
- II. co najmniej na jednym wejściu występuje sekwencja zmian $0 \frac{d1^1 d0 \dots d0 d1^k}{1} \rightarrow 0$, lub co najmniej na jednym wejściu występuje sekwencja zmian $0 \frac{d1 d0^1 \dots d0^r}{1} \rightarrow 1$, albo zmiana $0 \rightarrow 1$ i co najmniej na jednym wejściu występuje sekwencja zmian $1 \frac{d0 d1^1 \dots d1^r}{1} \rightarrow 0$ albo zmiana $1 \rightarrow 0$,
- III. suma wszystkich zmian dominujących $\frac{d1d0}{1}$ i $1 \frac{d0}{d0}$ dowolnych sekwencji zmian występujących na wejściach elementu logicznego I równa jest k-1 oraz spełnione są na jego wejściach następujące warunki wystarczające:

- a) ostatnia w czasie ze zmian $0 \rightarrow 1$ i $0 \xrightarrow{d1}$ występuje wcześniej od pierwszej w czasie ze zmian $1 \xrightarrow{d0}$ i $d1d0$,
- b) - pierwsza w czasie ze zmian $d0d1$ i $d0 \rightarrow 1$ występuje wcześniej niż $k-2$ zmian dominujących $1 \xrightarrow{d0}$ i $d1d0$,
- druga w czasie ze zmian $d0d1$ i $d0 \rightarrow 1$ występuje wcześniej niż $k-3$ zmian dominujących $1 \xrightarrow{d0}$ i $d1d0$,
- ⋮
- $k-2$ w czasie ze zmian $d0d1$ i $d0 \rightarrow 1$ występuje wcześniej niż ostatnia ze zmian $1 \xrightarrow{d0}$ i $d1d0$,
- c) ostatnia w czasie ze zmian $d0 \rightarrow 1$ i $d0d1$ występuje wcześniej niż pierwsza w czasie ze zmian $1 \rightarrow 0$ i $d1 \rightarrow 0$

Dowód:

Lemat wynika z definicji iloczynu logicznego.

Lemat 2.3.6

Sekwencja zmian $1 \xrightarrow{d0d1^{1..k}} 0$ ($0 \xrightarrow{d1d0^{1..k}} 1$) wystąpi na pewno na wyjściu idealnego elementu logicznego I, gdy spełnione są na jego wejściach następujące warunki konieczne:

- I. na żadnym wejściu nie występuje stan logiczny 0, zmiana $0 \rightarrow 1$ ($1 \rightarrow 0$) sekwencja zmian $0 \xrightarrow{d1^1 d0^{k-1} \dots d0d1^k} 0$ oraz sekwencja zmian $0 \xrightarrow{d1d0^{1..k}} 1$ ($1 \xrightarrow{d0d1^{1..k} \dots d1^k} 0$),
- II. co najmniej na jednym wejściu występuje sekwencja zmian $1 \xrightarrow{d0d1^{1..k} \dots d1^k} 0$ ($0 \xrightarrow{d1d0^{1..k} \dots d0^k} 1$) lub występuje zmiana $1 \rightarrow 0$ ($0 \rightarrow 1$),
- III. suma wszystkich zmian dominujących $d1d0$ i $1 \xrightarrow{d0} (d1d0)$ dowolnych niezabronionych sekwencji zmian występujących na wejściach elementu logicznego I równa jest k

oraz spełnione są na jego wejściach następujące warunki wystarczające:

- a) (obowiązuje tylko dla sekwencji $0 \frac{d1d0^1 \dots d0^k}{1}$) ostatnia w czasie ze zmian $0 \rightarrow 1$ i $0 \xrightarrow{d1} 1$ występuje wcześniej od pierwszej w czasie ze zmian $1 \xrightarrow{d0}$ i $\frac{d1d0}{1}$,
- b) - pierwsza w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{1}$ występuje wcześniej niż $k-1$ zmian dominujących $1 \xrightarrow{d0}$ i $\frac{d1d0}{1}$,
 - druga w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{1}$ występuje wcześniej niż $k-2$ zmian dominujących $1 \xrightarrow{d0}$ i $\frac{d1d0}{1}$,
 \vdots
 - $k-1$ w czasie ze zmian $\frac{d0d1}{1}$ i $\frac{d0}{1}$ występuje wcześniej niż ostatnia ze zmian dominujących $1 \xrightarrow{d0}$ i $\frac{d1d0}{1}$,
- c) (obowiązuje tylko dla sekwencji $1 \frac{d0d1^1 \dots d1^k}{0}$) ostatnia w czasie ze zmian $\frac{d0}{1}$ i $\frac{d0d1}{1}$ występuje wcześniej niż pierwsza w czasie ze zmian $1 \rightarrow 0$ i $\frac{d1}{0}$

Dowód:

Lemat wynika z definicji iloczynu logicznego.

Lemat 2.3.7

Sekwencje zmian $1 \frac{d0^1 d1 \dots d1 d0^k}{1}$, $0 \frac{d1^1 d0 \dots d0 d1^k}{0}$,
 i $\frac{d0d1^1 \dots d1^k}{0}$, $0 \frac{d1d0^1 \dots d0^k}{1}$ wystąpią na pewno na wyjściu idealnego elementu logicznego NOR(NAND), gdy spełnione są warunki konieczne i wystarczające zawarte kolejno:

- dla pierwszej sekwencji zmian w lemacie 2.2.4 (2.2.1)
- dla drugiej sekwencji zmian w lemacie 2.2.5 (2.2.2)
- dla trzeciej i czwartej sekwencji zmian w lemacie 2.2.6 (2.2.3)

przy równoczesnym założeniu, że zmiany i sekwencje zmian przedstawione w tych lemaciach są negacjami zmian i sekwencji zmian, w rzeczywistości występujących na wejściach elementu logicznego NOR (NAND).

Dowód:

Lemat wynika z prawa De Morgana.

2.4. Algorytm I^k identyfikacji k-krotnego hazardu podstawowego ze względu na zmienną x_1

2.4.1. Podstawowe twierdzenia

Lemat 2.4.1.1.*)

Każdy prosty strukturalny implikant PK_0^m i każdy prosty strukturalny implicent QK_0^m jest bezhazardowy.

Dowód:

Lemat jest oczywisty.

Lemat 2.4.1.2.**)

Jeżeli element logiczny B^m jest elementem NAND (NOR) lub elementem logicznym LUB (I), to układ przełączający realizujący funkcję F^m nie posiada na pewno HSO(HS1) i HD, gdy wszystkie strukturalne implikanty PK_0^m (strukturalne implicenty QK_0^m) są proste.

Dowód:

Zbiory liter prostych wyrażeń strukturalnych określających proste strukturalne implikanty PK_0^m (proste strukturalne implicenty QK_0^m) są zawsze stabilnymi zbiorami 1 (stabilnymi zbiorami 0) układu przełączającego realizującego funkcję F^m . Ponieważ warunki^{***)} istnienia HSO(HS1) i HD tworzą niestabilne zbiory 1 (niestabilne zbiory 0), więc lemat jest słuszny.

Twierdzenia 2.4.1.1

Jeżeli element logiczny B^m jest elementem NAND albo elementem LUB, to układ przełączający realizujący funkcję F^m posia-

*) Lematy opracowane na bazie twierdzeń 7.3.3. i 7.3.5. zawartych w pracy [31]

**) Zob. Przypisy do p. 2.4

da na pewno $HS1^k(x_1)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_1 , dla których stan wyjścia układu przełączającego realizującego funkcję F^m równy jest stanowi logicznemu 1,
- II. nie ma bezhazardowego, ze względu na zmienną x_1 , strukturalnego implikantu PK_0^m pokrywającego równocześnie oba stany wejścia A i A' ,
- III. a) istnieje r hazardowych strukturalnych implikantów PK_h^m posiadających $HS1^{\alpha_h}(x_1)$ gdzie $r \geq 0$,
 b) istnieje s hazardowych strukturalnych implikantów PK_1^m posiadających $HS0^{\beta_s}(x_1)$ gdzie $s \geq 0$,
 c) istnieje t hazardowych strukturalnych implikantów PK_w^m posiadających $HD^{\gamma_w}(x_1)$ gdzie $t \geq 0$,
 d) spełnione jest równanie

$$\sum_{h=1}^r \alpha_h - r + 1 + \sum_{l=1}^s \beta_l + \sum_{w=1}^t \gamma_w = k$$

Dowód:

Aby udowodnić to twierdzenie, należy przede wszystkim dowieść, że zbiór stabilnych zbiorów 1 z poszczególnych wyrażeń strukturalnych S_0^m implikantów strukturalnych PK_0^m jest równoważny zbiorowi stabilnych zbiorów 1 i wyrażenia strukturalnego S^m oraz, że nie istnieje stabilny zbiór 1 pokrywający równocześnie oba sąsiednie stany wejścia A i A' .

Stabilne zbiory 1 z poszczególnych wyrażeń strukturalnych S_0^m implikantów strukturalnych PK_0^m otrzymuje się drogą użycia w wyrażeniach strukturalnych S_0^m prawa rozdzielności mnożenia względem dodawania oraz prawa idempotentności [19].

Stabilne zbiory 1 i wyrażenia S^m otrzymuje się także drogą użycia powyższych praw. Ponieważ $S^m = \vee S_0^m$ oraz ponieważ do uzyska-

nia w obu przypadkach stabilnych zbiorów 1, użyto tych samych praw, więc oczywiste jest, że zbiór stabilnych zbiorów 1 z poszczególnych wyrażań S_0^m jest równoważny zbiorowi stabilnych zbiorów 1 wyrażenia strukturalnego S^m . Załóżmy, że istnieje stabilny zbiór 1 wyrażenia strukturalnego któregośkolwiek implikantu strukturalnego PK_0^m , który pokrywa równocześnie oba stany wejścia A i A'. Jeżeli taki zbiór istnieje, to z definicji 2.2.26 wynika, że również istnieje bezhazardowy ze względu na zmienną x_1 strukturalny implikant PK_0^m pokrywający oba stany wejścia A i A'. Ponieważ jest to sprzeczne z warunkiem twierdzenia Huffmana^{*)}, więc na pewno układ przełączający realizujący og funkcję F^m nie posiada HS1 (x_1).

Tym samym udowodniono, że nie może istnieć bezhazardowy ze względu na zmienną (x_1) strukturalny implikant PK_0^m pokrywający równocześnie oba stany wejścia A i A'.

Ponieważ jednak $f^m(A) = f^m(A') = 1$, więc na pewno oba stany wejścia pokryte są równocześnie przez hazardowe strukturalne implikanty PK_h^m posiadające HS1^{αh} (x_1) lub każdy z nich z osobna jest pokryty przez hazardowe implikanty PK_w^m posiadające HD^{rw} (x_1).

Z definicji 2.2.20, 2.2.21, 2.2.22, 2.2.23 i 2.2.24 wynika, że istnienie HS1^{αh}, HSO^{βl} i HD^{rw} w strukturalnych implikantach PK_0^m zapewnia możliwość wystąpienia na wejściach elementu logicznego B^m następujących sekwencji zmian $1 \frac{d0^1 d1^{..} \cdot d1 d0^{..h}}{d1^1 d0^{..} \cdot d0 d1^{βl}} 0, 1 \frac{d0 d1^{1..} \cdot d1^{rw}}{d1 d0^{1..} \cdot d0^{rw}} 1$.

Z lematu 2.3.7 i z warunku III lematu 2.3.1 wynika jednocześnie, że na wyjściu elementu logicznego B^m pojawi się sekwencja zmian $1 \frac{d0^1 d1^{..} \cdot d1 d0^k}{d1^1 d0^{..} \cdot d0 d1^{βl}} 1$, gdy suma wszystkich zmian dominujących na jego wejściach równa jest k-1. W takim razie oczywiste staje się także wniosek, że po transpozycji ilości zmian dominujących w ilość czasowych stanów logicznych d0 musi być także spełniona równość zawarta w warunku III d twierdzenia.

*) Zob. Przypisy do p. 2.4

Twierdzenie 2.4.1.2

Jeżeli element logiczny B^m jest elementem NAND albo elementem LUB, to układ przełączający realizujący funkcję F^m posiada na pewno $HSO^k(x_1)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_1 , dla których stan wyjścia układu przełączającego realizującego funkcję F^m równy jest stanowi logicznemu 0 .
- II. a) istnieje s hazardowych strukturalnych implikantów PK_1^m posiadających $HSO^{\beta_l}(x_1)$ gdzie $s \geq 1$,
 b) spełnione jest równanie $\sum_{l=1}^s \beta_l = k$

Dowód:

Jeżeli $f^m(A) = f^m(A') = 0$, to na pewno nie istnieją implikanty strukturalne PK_0^m pokrywające którykolwiek ze stanów wejścia A lub A' , czyli tym samym nie istnieją hazardowe implikanty strukturalne PK_h^m posiadające $HS1^{ch}(x_1)$ i hazardowe implikanty strukturalne PK_w^m posiadające $HD^{fw}(x_1)$.

Z definicji 2.2.21, 2.2.23 i 2.2.24 wynika, że istnienie HSO^{β_l} w strukturalnych implikantach PK_0^m zapewnia możliwość wystąpienia na wejściach elementu logicznego B^m sekwencji zmian $0 \underline{d1^1 d0^{\dots} \cdot d0 d1^{\beta_l}} 0$.

Ponieważ jednocześnie z lematu 2.3.7 i z warunku III lematu 2.3.2 wynika, że na wyjściu elementu logicznego B^m pojawi się sekwencja zmian $0 \underline{d1^1 d0^{\dots} \cdot d0 d1^k} 0$, gdy suma wszystkich zmian dominujących na jego wejściach równa jest k , więc oczywisty jest wniosek, że musi istnieć co najmniej jeden hazardowy implikant PK_l^m posiadający $HSO^{\beta_l}(x_1)$ oraz, że musi być spełnione równanie w warunku II b twierdzenia.

Twierdzenie 2.4.1.3

Jeżeli element logiczny B^m jest elementem NAND albo elementem LUB, to układ przełączający realizujący funkcję F^m posiada na pewno $HD^k(x_1)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_1 , dla których stan wyjścia $f^m(A) = 0$, a stan wyjścia $f^m(A') = 1$,
- II. a) istnieje s hazardowych strukturalnych implikantów PK_1^m posiadających $HSO^{\beta l}(x_1)$, gdzie $s \geq 0$,
- b) istnieje t hazardowych strukturalnych implikantów PK_w^m posiadających $HD^{\tau w}(x_1)$, gdzie $t \geq 0$,
- c) spełnione jest równanie $\sum_{l=1}^s \beta_l + \sum_{w=1}^t \tau_w = k$.

Dowód:

Jeżeli $f^m(A) = 0$, to na pewno nie ma implikantów strukturalnych PK_0^m pokrywających równocześnie oba stany wejścia A i A' , czyli tym samym nie ma hazardowych implikantów strukturalnych PK_h^m posiadających $HS1^{\alpha h}(x_1)$.

Ponieważ $f^m(A') = 1$ więc na pewno istnieje jakikolwiek strukturalny implikant PK_0^m pokrywający stan wejścia A .

Z definicji 2.2.22, 2.2.23 i 2.2.24 wynika, że istnienie $HSO^{\beta l}$ i $HD^{\tau w}$ w strukturalnych implikantach PK_0^m zapewnia możliwość wystąpienia na wejściach elementu logicznego B^m następujących sekwencji zmian $0 \underline{d1^1 d0^{\dots} \cdot d0d1^{\beta l}} 0$, oraz $0 \underline{d1d0^1 \dots \cdot d0^{\tau w}} 1$ ($1 \underline{d0d1^1 \dots \cdot d1^{\tau w}} 0$).

Ponieważ jednocześnie z lematu 2.3.7 i z warunku III lematu 2.3.3 wynika, że na wyjściu elementu logicznego B^m pojawi się sekwencja zmian $0 \underline{d1d0^{\dots} \cdot d0^k} 1$ ($1 \underline{d0d1^1 \dots \cdot d1^k} 0$), gdy suma wszystkich zmian dominujących na jego wejściach jest równa k , więc oczywisty jest wniosek, że musi być także spełnione równanie zawarte w warunku III o twierdzenia.

Twierdzenie 2.4.1.4

Jeżeli element logiczny B^m jest elementem NOR albo elementem I, to układ przełączający realizujący funkcję F^m posiada na pewno $HS1^k(x_1)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_i , dla których stan wyjścia układu przełączającego realizującego funkcję F^m równy jest stanowi logicznemu 1,
- II. a) istnieje r hazardowych strukturalnych implicantów QK_h^m posiadających $HS1^{\alpha_h}(x_i)$ gdzie $r \geq 1$,
 b) spełnione jest równanie $\sum_{h=1}^r \alpha_h = k$.

Dowód:

Dowód przeprowadza się podobnie jak dowód twierdzenia 2.4.1.2.

Twierdzenie 2.4.1.5

Jeżeli element logiczny B^m jest elementem NOR albo elementem I, to układ przełączający realizujący funkcję F^m posiada na pewno $HSO^k(x_i)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_i , dla których stan wyjścia układu przełączającego realizującego funkcję F^m równy jest stanowi logicznemu 0,
- II. nie ma bezhazardowego ze względu na zmienną x_i strukturalnego implicantu QK_0^m pokrywającego równocześnie oba stany wejścia A i A' ,
- III. a) istnieje s hazardowych strukturalnych implicantów QK_1^m posiadających $HSO^{\beta_l}(x_i)$ gdzie $s \geq 0$,
 b) istnieje r hazardowych strukturalnych implicantów QK_h^m posiadających $HS1^{\alpha_h}(x_i)$ gdzie $r > 0$,
 c) istnieje t hazardowych strukturalnych implicantów QK_w^m posiadających $HD^{\tau_w}(x_i)$ gdzie $t \geq 0$,
 d) spełnione jest równanie $\sum_{l=1}^s \beta_l - s + 1 + \sum_{h=1}^r \alpha_h + \sum_{w=1}^t \tau_w = k$.

Dowód:

Dowód przeprowadza się podobnie jak dowód twierdzenia 2.4.1.1.

Twierdzenie 2.4.1.6

Jeżeli element logiczny B^m jest elementem NOR albo elementem I, to układ przełączający realizujący funkcję F^m posiada na pewno $HD^k(x_1)$ wtedy, gdy spełnione są następujące warunki:

- I. istnieje para sąsiednich stanów wejścia A i A' różniących się stanem logicznym a_1 , dla których stan wyjścia $f^m(A) = 0$, a stan wyjścia $f^m(A') = 1$,
- II. a) istnieje r hazardowych strukturalnych implicitów QK_h^m posiadających $HS1^{\alpha_h}(x_1)$ gdzie $r \geq 0$,
b) istnieje t hazardowych strukturalnych implicitów QK_w^m posiadających $HD^{\tau_w}(x_1)$, gdzie $t > 0$,
c) spełnione jest równanie $\sum_{h=1}^r \alpha_h + \sum_{w=1}^t \tau_w = k$,

Dowód:

Dowód przeprowadza się podobnie jak dowód twierdzenia 2.4.1.3.

2.4.2. Charakterystyka algorytmu I^k

Na podstawie przedstawionych poprzednio lematów i twierdzeń zbudowano następujący algorytm identyfikacji k -krotnego hazardu podstawowego $HP^k(x_1)$:

1. Na schemacie kombinacyjnego układu przełączającego, począwszy od wyjściowego elementu logicznego, numeruje się kolejno wszystkie elementy logiczne oraz wszystkie tory sygnałowe zmiennych wejściowych x wchodzących na te elementy logiczne B^m , które wraz z podporządkowanymi im wycinkami W^m można przedstawić tylko za pomocą złożonych wyrażeń strukturalnych S^m .

2. W zależności od typu operacji logicznych wykonywanych przez wszystkie te elementy logiczne B^m , których podporządkowane wycinki W^m są przedstawione w postaci koniunkcji lub alternatywy tylko prostych wyrażeń strukturalnych S_o^m , określa się za pomocą tychże prostych wyrażeń strukturalnych S_o^m lub za pomocą ich negacji \bar{S}_o^m proste strukturalne implikanty PK_o^m i proste strukturalne implícenty QK_o^m .
3. Wykorzystując lematy 2.4.1.1 i 2.4.1.2 oraz twierdzenia 2.4.1.1 i 2.4.1.5 identyfikuje się $HP^k(x_1)$ na wyjściach elementów logicznych B^m .
4. W zależności od typu operacji logicznej wykonywanej przez funktry B^m , których podporządkowane wycinki W^m są przedstawione w postaci koniunkcji lub alternatywy wyrażeń strukturalnych S_o^m , ze zbioru prostych wyrażeń strukturalnych i ze zbioru złożonych wyrażeń strukturalnych o zidentyfikowanych już w krokach poprzednich $HP^k(x_1)$, określa się za pomocą tychże wyrażeń strukturalnych S_o^m lub za pomocą ich negacji \bar{S}_o^m strukturalne implikanty PK_o^m i strukturalne implícenty QK_o^m , uwzględniając w nich zidentyfikowane już poprzednio $HP^k(x_1)$.
5. Wykorzystując lemat 2.4.1.1 oraz twierdzenia 2.4.1.1, 2.4.1.2, 2.4.1.3, 2.4.1.4, 2.4.1.5 i 2.4.1.6 identyfikuje się $HP^k(x_1)$ na wyjściach elementów logicznych B^m .
6. Krok 4 i 5 powtarza się do momentu zidentyfikowania $HP^k(x_1)$ na wyjściu kombinacyjnego układu przełączającego.

2.5. Zastosowanie algorytmu I^k do identyfikacji k-krotnego hazardu podstawowego w kombinacyjnych układach przełączających za pomocą składania tablicy strukturalnych implikantów i strukturalnych implícentów

2.5.1. Tablica strukturalnych implikantów i strukturalnych implícentów

Tablica PK_o^m (tablica strukturalnych implikantów i strukturalnych implícentów) jest obrazem struktury kombinacyjnego

układu przełączającego. Liczba jej kolumn równa jest liczbie wszystkich par sąsiednich stanów wejścia funkcji realizowanej przez kombinacyjny układ przełączający. Każda z tych par zapisana jest w górnym wierszu tablicy w postaci dwóch cyfr reprezentujących dziesiętny zapis dwóch sąsiednich stanów wejścia. Taki sposób zapisu sąsiednich stanów wejścia przedstawiono po raz pierwszy w pracy [31], gdzie wykorzystano go do bezhazardowej syntezy dwuwarstwowych kombinacyjnych układów przełączających LUB-I. Liczba wierszy tablicy PQK_0^m jest ściśle uzależniona od ilości PK_0^m i QK_0^m , a więc tym samym od struktury kombinacyjnego układu przełączającego zobrazowanej zawsze w lewej części tablicy PQK_0^m (z lewej strony kolumn). Przykład tablicy na s.59.

W tablicy PQK_0^m rozróżnia się wiersze proste i złożone. Każdy wiersz prosty jest obrazem prostego PK_0^m lub QK_0^m , natomiast wiersz złożony składa się z dwóch lub więcej wierszy i jest obrazem złożonego PK_0^m lub QK_0^m . Wiersze odpowiadające PK_0^m lub QK_0^m tej samej funkcji F^m stykają się ze sobą.

Każdy PK_0^m zapisywany jest w poszczególnych kolumnach odpowiadającego mu wiersza za pomocą stanu logicznego 1 i kombinacji stanów logicznych $\begin{matrix} 0 \\ 1 \end{matrix}$ i $\begin{matrix} 1 \\ 0 \end{matrix}$. Natomiast każdy QK_0^m zapisywany jest w odpowiadającym mu wierszu za pomocą stanu logicznego 0 i kombinacji stanów logicznych $\begin{matrix} 1 \\ 0 \end{matrix}$ i $\begin{matrix} 0 \\ 1 \end{matrix}$. Stan logiczny 1 (0) określa, że PK_0^m (QK_0^m) pokrywa daną parę sąsiednich stanów wejścia, natomiast kombinacje stanów logicznych $\begin{matrix} 0 & (1) \\ 1 & (0) \end{matrix}$ i $\begin{matrix} 1 & (0) \\ 0 & (1) \end{matrix}$ określają, który stan wejścia pary sąsiednich stanów wejścia danej kolumny pokryty jest przez PK_0^m (QK_0^m).

Tak więc kombinacja $\begin{matrix} 1 & (0) \\ 0 & (1) \end{matrix}$ oznacza, że górny stan wejścia pokryty jest przez PK_0^m (QK_0^m), natomiast kombinacja $\begin{matrix} 0 & (1) \\ 1 & (0) \end{matrix}$ oznacza, że dolny stan wejścia jest pokryty przez PK_0^m (QK_0^m). HP^k w strukturalnych implikantach PK_0^m i strukturalnych implimentach QK_0^m zapisywany jest w odpowiednich kolumnach wierszy odpowiadających PK_0^m i QK_0^m za pomocą następujących kombinacji stanów logicznych

$$\begin{matrix} 0 & \begin{pmatrix} 0 \\ \beta_l \end{pmatrix} & 1 & \begin{pmatrix} 1 \\ \alpha_h \end{pmatrix} & 1 & \begin{pmatrix} 1 \\ \tau w \end{pmatrix} & 0 & \begin{pmatrix} 0 \\ \tau w \end{pmatrix} \\ 0 & \begin{pmatrix} 0 \\ 0 \end{pmatrix} & 1 & \begin{pmatrix} 1 \\ 1 \end{pmatrix} & 0 & \begin{pmatrix} 1 \\ 0 \end{pmatrix} & 1 & \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{matrix} \quad \text{gdzie:}$$

- kombinacja $\begin{matrix} 0 \\ k \\ 0 \end{matrix}$ $\begin{pmatrix} 0 \\ \beta_l \\ 0 \end{pmatrix}$ oznacza HSO^k (HSO^{β_l})
- kombinacja $\begin{matrix} 1 \\ k \\ 1 \end{matrix}$ $\begin{pmatrix} 1 \\ \alpha_h \\ 1 \end{pmatrix}$ oznacza $HS1^k$ ($HS1^{\alpha_h}$)
- kombinacja $\begin{matrix} 0 \\ k \\ 1 \end{matrix}$ $\begin{pmatrix} 0 \\ \gamma_w \\ 1 \end{pmatrix}$ oraz $\begin{matrix} 1 \\ k \\ 0 \end{matrix}$ $\begin{pmatrix} 1 \\ \gamma_w \\ 0 \end{pmatrix}$ oznaczają HD^k (HD^{γ_w}).

Umieszczone w lewej części tablicy PQK_o^m dodatkowe informacje o operacjach logicznych wykonywanych przez poszczególne elementy logiczne B^m , jak również informacje o operacji negowania danego wiersza (gruba pionowa kreska) są informacjami pomocniczymi ułatwiającymi składanie tablicy PQK_o^m (tabela 5).

Składanie tablicy PQK_o^m , polega na wykonywaniu na niej kroków 2,3,4,5 i 6 algorytmu I^k drogą kolejnego składania wierszy tablicy według następujących zasad:

- a) składane są tylko wiersze odpowiadające PK_o^m lub tylko wiersze odpowiadające QK_o^m ,
- b) do składania wierszy odpowiadających PK_o^m , wykorzystuje się wynikające z twierdzeń 2.4.1.1., 2.4.1.2. i 2.4.1.3. następujące prawa:

$$\begin{aligned}
 & - \left| \begin{matrix} 0 & 1 & 1 & 0 & 0 & 1 \\ & & \alpha_h & \beta_l & \gamma_w & \gamma_w \end{matrix} \right| + \left| \begin{matrix} 1 & 0 & 1 & 0 & 1 & 0 \end{matrix} \right| = \left[\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 \\ & & & & & \end{matrix} \right] \\
 & - \left| \begin{matrix} 0 & 1 & 1 & 0 & 0 & 1 \\ & & \alpha_h & \beta_l & \gamma_w & \gamma_w \end{matrix} \right| + \left| \begin{matrix} 1 & 0 & 1 & 0 & 1 & 0 \end{matrix} \right| = \left[\begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 \\ & & \alpha_h & \beta_l & \gamma_w + 1 & \gamma_w \\ 1 & 0 & 1 & 0 & 1 & 0 \end{matrix} \right] \\
 & - \alpha_{h1} + \left| \begin{matrix} 1 & 0 & 1 & 0 \\ \alpha_{h2} & \beta_l & \gamma_w & \gamma_w \end{matrix} \right| = \left[\begin{matrix} 1 & 1 & 1 & 1 \\ \alpha_{h1} + \alpha_{h2}^{-1} & \alpha_{h1} + \beta_l & \alpha_{h1} + \gamma_w & \alpha_{h1} + \gamma_w \\ 1 & 1 & 1 & 1 \end{matrix} \right] \\
 & - \beta_{l1} + \left| \begin{matrix} 0 & 1 & 0 \\ \beta_{l2} & \gamma_w & \gamma_w \end{matrix} \right| = \left[\begin{matrix} 0 & 1 & 0 \\ \beta_{l1} + \beta_{l2} & \beta_{l1} + \gamma_w & \beta_{l1} + \gamma_w \\ 0 & 0 & 1 \end{matrix} \right]
 \end{aligned}$$

$$- \begin{matrix} 1 \\ \tau_{w1} \\ 0 \end{matrix} + \begin{bmatrix} 1 & 0 \\ \tau_{w2} & \tau_{w2} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \tau_{w1} + \tau_{w2} & \tau_{w1} + \tau_{w2} + 1 \\ 0 & 1 \end{bmatrix}$$

c) do składania wierszy odpowiadających QK_0^m wykorzystuje się, wynikające z twierdzeń 2.4.1.4., 2.4.1.5. i 2.4.1.6., następujące prawa:

$$- \begin{bmatrix} \\ \\ \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ & \beta_l & \alpha_h & \tau_w & \tau_w & \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}$$

$$- \begin{matrix} 0 \\ 1 \end{matrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ & \beta_l & \alpha_h & \tau_w & \tau_w & \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ & \beta_l & \alpha_h & \tau_w + 1 & \tau_w & \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$$- \begin{matrix} 0 \\ \beta_{l1} \\ 0 \end{matrix} \cdot \begin{bmatrix} 0 & 1 & 0 & 1 \\ \beta_{l2} & \alpha_h & \tau_w & \tau_w \\ 0 & 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_{l1} + \beta_{l2}^{-1} & \beta_{l1} + \alpha_h & \beta_{l1} + \tau_w & \beta_{l1} + \tau_w \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$- \begin{matrix} 1 \\ \alpha_{h1} \\ 1 \end{matrix} \cdot \begin{bmatrix} 1 & 0 & 1 \\ \alpha_{h2} & \tau_w & \tau_w \\ 1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ \alpha_{h1} + \alpha_{h2} & \alpha_{h1} + \tau_w & \alpha_{h1} + \tau_w \\ 1 & 1 & 0 \end{bmatrix}$$

$$- \begin{matrix} 0 \\ \tau_{w1} \\ 1 \end{matrix} \cdot \begin{bmatrix} 0 & 1 \\ \tau_{w2} & \tau_{w2} \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ \tau_{w1} + \tau_{w2} & \tau_{w1} + \tau_{w2} + 1 \\ 1 & 0 \end{bmatrix}$$

d) zamianę wiersza odpowiadającego PK_0^m (QK_0^m) w wiersz odpowiadający QK_0^m (PK_0^m) wykonuje się przez wykreślenie w kolumnach wiersza stanów logicznych 1 (0) i wpisanie w puste kolumny wiersza stanów logicznych 0 (1),

e) do negowania wiersza wykorzystuje się poniższe prawo:

$$\begin{bmatrix} 1 & & 1 & 0 & 1 & 0 & 1 & 0 \\ & & & k & k & k & k & \\ & & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ & & & k & k & k & k & \\ & & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

2.5.2. Przykłady

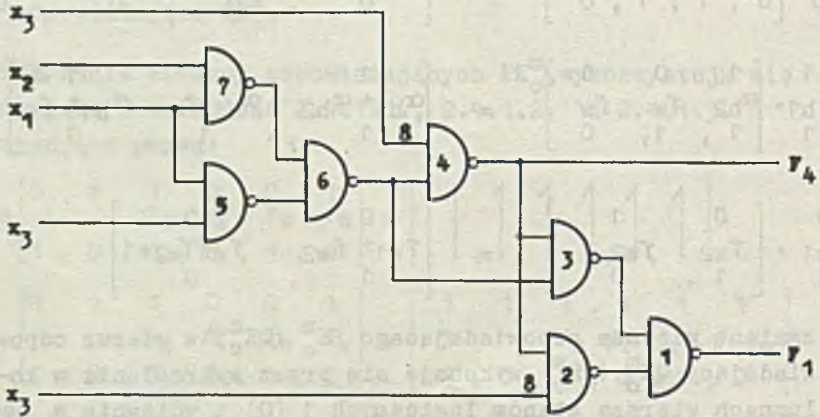
Przykład 2.5.2.1

Rys. 7 przedstawia schemat logiczny trójwarstwowego kombinacyjnego układu przełączającego.

Tablicę PQK^m dla tego układu przełączającego przedstawia tabela 1^{*)}.

Algorytm I^k

- Wynik kroku 2 i kroku 3 przedstawia tabela 2.
- Krok pomocniczy przedstawia tabela 3.
- Wynik kroku 4 i 5 przedstawia tabela 4. Tabela ta jest równocześnie końcowym rezultatem uzyskanym w wyniku zastosowania algorytmu I^k i w związku z tym jest obrazem funkcji F^1 i $HP^k(x_i)$, który posiada analizowany układ przełączający.



Rys. 12. Przykład dwuwyjściowego i pięciowarstwowego kombinacyjnego układu przełączającego na podstawie pracy [55]

*) Generację tej tabeli przedstawiono w przypisach do podrozdziału 2.5.2

Tabela 1

Wagi przypisane poszczególnym zmiennym: $x_1 - 8$, $x_2 - 4$, $x_3 - 2$, $x_4 - 1$

Tablica		PQK ₀ ^m	0	0	0	0	1	1	1	2	2	2	4	4	4	4	8	8	8	8	3	3	5	5	6	6	6	9	9	10	10	10	12	12	7	11	13	14
			1	2	4	3	3	5	9	3	6	10	5	6	12	9	10	12	7	11	7	11	7	13	7	14	11	13	11	14	13	14	13	14	15	15	15	15
F ¹ (LUB)	PK ₅ ¹																									0			0		0	0	0	0	0	1		
																										1			1		1	1	1	1	1	1	1	
	PK ₂ ¹	F ² (I)	QK ₄ ²	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
			1	0	0	0	1	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	

Tabela 2

PQK ₀ ^m		0	0	0	0	1	1	1	2	2	2	4	4	4	4	8	8	8	8	3	3	5	5	6	6	6	9	9	10	10	10	12	12	7	11	13	14
		1	2	4	8	3	5	9	3	6	10	5	6	12	9	10	12	7	11	7	11	7	13	7	14	11	13	11	14	14	13	14	15	15	15	15	
F ¹ (LUB)	PK ₅ ¹																									0			0		0	0	0	0	0	1	
																										1			1		1	1	1	1	1	1	1
	PK ₂ ¹	F ² (I)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		1	0	0	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Tabela 3

PQX_0^m		0	0	0	0	1	1	1	2	2	2	4	4	4	8	8	8	3	3	5	5	6	6	9	9	10	10	12	12	7	11	13	14
		1	2	4	8	3	5	9	3	6	10	5	6	12	9	10	12	7	11	7	13	7	14	11	13	11	14	13	14	15	15	15	15
F^1 (LUB)	PK_5^1																						0			0		0	0	0	0	0	1
	PK_2^1	0		0	1	1	1	0		0	0		0	1	0	1	1	1	1	1	0	0	1	1				1	1	1		1	
		1		1	1	1	1	1		0	1		1	1	1	1	1	0	0	1	1	1	0	0				1	0	0		0	

Tabela 4

PQX_0^m		0	0	0	0	1	1	1	2	2	2	4	4	4	8	8	8	3	3	5	5	6	6	9	9	10	10	12	12	7	11	13	14
		1	2	4	8	3	5	9	3	6	10	5	6	12	9	10	12	7	11	7	13	7	14	11	13	11	14	13	14	15	15	15	15
F^1		0		0	1	1	1	0		0	0		0	1	1	1	1	1	1	1	0	1	1			0	1	1	1	0	1	1	
		1		1	1	1	1	1		0	1		1	1	1	1	0	0	1	0	1	0	1			1	1	1	1	1	1	1	1

Tabela 6

PQK ₀ ^m					0 0 0 1 1 2 2 4 4 3 5 6	
					1 2 4 3 5 3 6 5 6 7 7 7	
F ¹ (NAND)	PK ₂ ¹	F ₂ ² (NAND)	PK ₄ ²	F _F ⁴ (NAND)	PK ₆ ⁴	1 1 1 1 1 1 0 1 1 0 0 0
					PK ₈ ⁴	0 1 1 0 1 1 0 1 1 0
				PK ₈ ²		
	PK ₃ ¹	F ₃ ³ (NAND)	PK ₄ ³	F _F ⁴ (NAND)	PK ₆ ⁴	1 1 1 1 1 1 0 1 1 0 0 0
					PK ₈ ⁴	0 1 1 0 1 1 1 1 1 0
				PK ₆ ³		

Tabela 7

PQK ₀ ^m					0 0 0 1 1 2 2 4 4 3 5 6
					1 2 4 3 5 3 6 5 6 7 7 7
F ¹ (NAND)	PK ₂ ¹	F ₂ ² (NAND)	PK ₄ ²	F _F ¹	1 1 1 1 1 1 0 1 1 0 1 0
				PK ₈ ²	0 1 1 0 1 1 1 1 1 0
	PK ₃ ¹	F ₃ ³ (NAND)	PK ₄ ³	F _F ⁴	1 1 1 1 1 1 0 1 1 0 1 0
				PK ₆ ³	1 1 1 1 1 1 0 0 0 0 0

Algorytm I^k

- Wynik kroku 2 i kroku 3 przedstawia tabela 6.
- Wynik kroku 4 i 5 przedstawia tabela 7. Tabela ta jest równocześnie obrazem funkcji F⁴. Wynika z niej, że układ przełączający realizujący funkcję F⁴ jest bezhazardowy.

- Krok 6 przedstawiony jest w tabeli 8 i 9. Tabela 10 jest wynikiem kroku 6 i jest jednocześnie obrazem funkcji F^1 i $HP^k(x_1)$, który posiada analizowany układ przełączający.

Tabela 8

PQK^m_0				0	0	0	1	1	2	2	4	4	3	5	6
				1	2	4	3	5	3	6	5	6	7	7	7
F^1 (NAND)	PK^1_2	F^2 (NAND)	PK^2_4					0		0		0	1	0	
							1		1		1	1	1	1	
		PK^2_8		1	1	1			1	1	1			1	0
				0	1	1			0	1	0	1			0
PK^1_3	F^3 (NAND)	PK^3_4						0		0		0	1	0	
						1		1		1	1	1	1	1	
	PK^3_6		1	1	1	1	1	1	0	1	1	1	1	1	0
								0	1	0	0	0	0	0	

Tabela 9

PQK^m_0			0	0	0	1	1	2	2	4	4	3	5	6
			1	2	4	3	5	3	6	5	6	7	7	7
F^1 (NAND)	PK^1_2		0			1	1	0		0		1		0
			1			0	1		0	0	0	0		0
PK^1_3						0	0	0	0	0	0		1	
						1		1		1				
						0	1	0	1	0			0	

Tabela 10

		0	0	0	1	1	2	2	4	4	3	5	6
		1	2	4	3	5	3	6	5	6	7	7	7
F^1		0			1	1	0	0	0	0	1		1
		1			0	1	1	0	1	0			0

Wynik identyfikacji $HP^k(x_1)$

HSO^2 w parze sąsiednich stanów wejścia 4,5 oznacza $HSO^2(x_2)$
 HD^1 w parze sąsiednich stanów wejścia 6,7 oznacza $HD^1(x_2)$
 HD^1 w parach sąsiednich stanów wejścia 4,5 i 3,7 oznacza $HD^1(x_1)$

2.6. Zastosowanie algorytmu I^k do identyfikacji k-krotnego hazardu podstawowego w kombinacyjnych układach przełączających za pomocą składania siatek Karnaugh

Metoda przedstawiona w tym rozdziale wykorzystuje, do przedstawiania prostych i złożonych implikantów strukturalnych PK_0^m i implikantów strukturalnych QK_0^m , siatkę Karnaugh. Do przedstawienia w niej zarówno PK_0^m jak również QK_0^m wykorzystuje się zasady podane w rozdziale 2.5.1. Zgodnie z nimi implikanty strukturalne PK_0^m (implikanty strukturalne QK_0^m) zapisywane są w postaci stanów logicznych 1(0) w tych kratkach siatki Karnaugh, które odpowiadają stanom wejścia pokrytym przez te PK_0^m (QK_0^m). Identyfikacja $HP^k(x_1)$ polega, podobnie jak w metodzie składania tablicy PK_0^m , na wykonywaniu kroków 2, 3, 4, 5 i 6 algorytmu I^k drogą kolejnego składania siatek Karnaugh według identycznych zasad, przedstawionych w punktach a), b), c), d) oraz e) rozdziału 2.5.1. Wyjątkowo, dla bardziej czytelnego obrazu siatki, cyfry k , α_n , β_1 oraz γ_w zastąpiono odpowiednią liczbą kresek na styku dwóch kratek siatki Karnaugh odpowiadających dwóm sąsiednim stanom wejścia. Również w celu uproszczenia zapisu, we wszystkich podanych dalej przykładach, w oznaczeniu " PK_0^m " (" QK_0^m ") i " S_0^m " opuszczono indeks "m", np. zamiast $P^5 = PK_3^5 + PK_4^5$ będzie się pisać $P^5 = PK_3 + PK_4$.

2.6.1. Przykłady

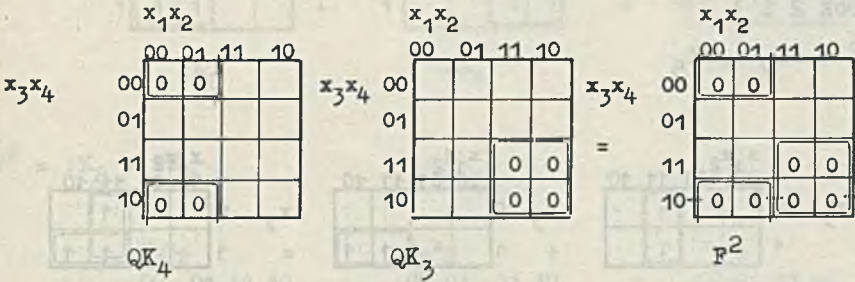
Przykład 2.6.1.1

Rys. 7 przedstawia schemat logiczny trójwarstwowego kombinacyjnego układu przełączającego.

Algorytm I^k

- Krok 2 i 3

$$F^2 = QK_4 \cdot QK_3$$

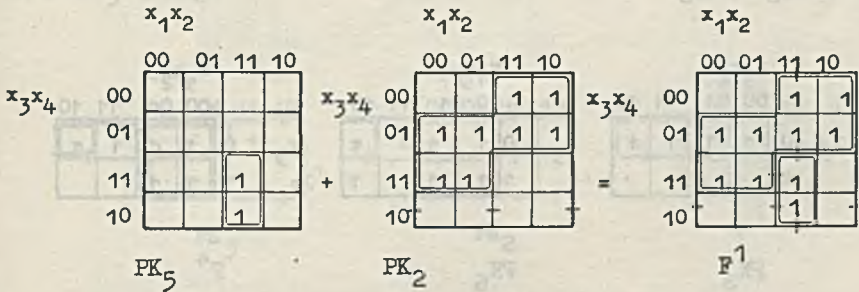


$$(S_4 = x_1 + x_4)$$

$$(S_3 = \bar{x}_1 + \bar{x}_3)$$

- Krok 4 i 5

$$F^1 = PK_5 + PK_2$$



$$(S_5 = x_1 \cdot x_2 \cdot x_3)$$

Wynik identyfikacji HP^k (x₁)

Kombinacyjny układ przełączający posiada:

- HS1¹ (x₃)
- HS1¹ (x₁)
- HD¹ (x₁)
- HSO¹ (x₁)

Przykład 2.6.1.2

Rys. 12 przedstawia schemat logiczny dwuwyjściowego i pięciowarstwowego kombinacyjnego układu przełączającego.

Algorytm I^k

- Krok 2 i 3

$$F^6 = PK_7 + PK_5$$

		$x_1 x_2$			
		00	01	11	10
x_3	0			1	
	1			1	

		$x_1 x_2$			
		00	01	11	10
x_3	0				
	1			1	1

		$x_1 x_2$			
		00	01	11	10
x_3	0			1	
	1			1	1

PK_7

$$(\bar{S}_7 = x_2 \cdot x_1)$$

PK_5

$$(\bar{S}_5 = x_1 \cdot x_3)$$

F^6

- Krok 4 i 5

$$F^4 = PK_8 + PK_6$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1	1	1
	1				

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1		1
	1	1	1		1

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1	1	1
	1	1	1		

PK_8

$$(\bar{S}_8 = \bar{x}_3)$$

PK_6

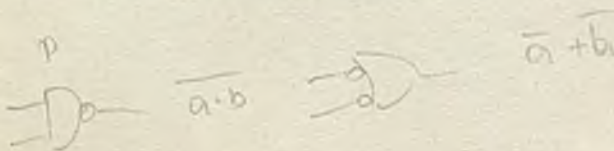
F^4

Wynik identyfikacji $HP^k(x_i)$ dla układu przełączającego realizującego F^4

Układ przełączający jest bezhazardowy

- Krok 6

$$F^3 = PK_6 + PK_4$$



		x_1x_2				
		00	01	11	10	
x_3	0	11	1		1	x_3
	1	1	1			

 $+$

		x_1x_2				
		00	01	11	10	
x_3	0					x_3
	1			1	1	

 $=$

		x_1x_2				
		00	01	11	10	
x_3	0	1	1		1	x_3
	1	1	1	1	1	

$PK_6 \quad PK_4 \quad F^3$

$F^2 = PK_8 + PK_4$

		x_1x_2				
		00	01	11	10	
x_3	0	1	1	1	1	x_3
	1					

 $+$

		x_1x_2				
		00	01	11	10	
x_3	0					x_3
	1			1	1	

 $=$

		x_1x_2				
		00	01	11	10	
x_3	0	1	1	1	1	x_3
	1			1	1	

$PK_8 \quad PK_4 \quad F^2$

$F^1 = PK_3 + PK_2$

		x_1x_2				
		00	01	11	10	
x_3	0			1		x_3
	1					

 $+$

		x_1x_2				
		00	01	11	10	
x_3	0					x_3
	1	1	1			

 $=$

		x_1x_2				
		00	01	11	10	
x_3	0			1		x_3
	1	1	1			

$PK_3 \quad PK_2 \quad F^1$

Wynik identyfikacji $HP^k(x_1)$ dla układu przełączającego realizującego F^1

Kombinacyjny układ przełączający posiada:

- $HSO^2(x_3)$
- $HD^1(x_1)$
- $HD^1(x_3)$

Przykład 2.6.1.3

Rys. 13 przedstawia schemat logiczny trójwyjściowego i sześciowarstwowego kombinacyjnego układu przełączającego.

Algorytm I^k

- Krok 2 i 3

$$F^7 = PK_8 + PK_9$$

		x_1x_2			
		00	01	11	10
x_3	0				
	1			1	1

PK₈

		x_1x_2			
		00	01	11	10
x_3	0				
	1	1			1

PK₉

		x_1x_2			
		00	01	11	10
x_3	0				
	1	1		1	1

F⁷

$$(\bar{S}_8 = x_1 \cdot x_2)$$

$$(\bar{S}_9 = \bar{x}_2 \cdot x_3)$$

Wynik identyfikacji $\bar{S}^k(x_i)$ dla układu przełączającego realizującego F^7

Kombinacyjny układ przełączający jest bezhazardowy

- Krok 4 i 5

$$F^6 = PK_{10} + PK_7$$

		x_1x_2			
		00	01	11	10
x_3	0			1	1
	1			1	1

PK₁₀

		x_1x_2			
		00	01	11	10
x_3	0	1	1	1	1
	1		1		

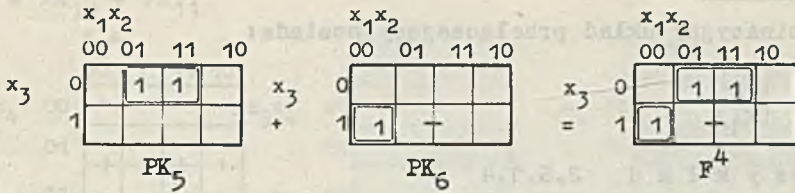
PK₇

		x_1x_2			
		00	01	11	10
x_3	0	1	1	1	1
	1		1	1	1

F⁶

$$(\bar{S}_{10} = x_1)$$

$$F^4 = PK_5 + PK_6$$



$$(\bar{S}_5 = x_2 \cdot \bar{x}_3)$$

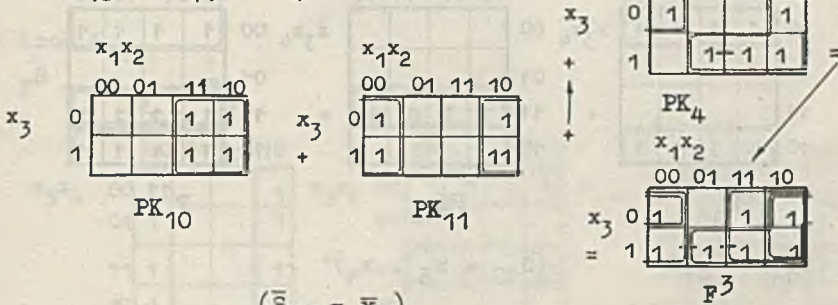
Wynik identyfikacji $HP^k(x_1)$ dla układu przełączającego realizującego F^4

Kombinacyjny układ przełączający posiada:

$$HSO^1(x_1)$$

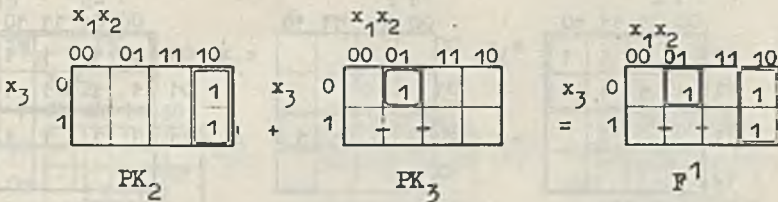
- Krok 6

$$F^3 = PK_{10} + PK_{11} + PK_4$$



$$(\bar{S}_{11} = \bar{x}_2)$$

$$F^1 = PK_2 + PK_3$$



$$(\bar{S}_2 = x_1 \cdot \bar{x}_2)$$

Wynik identyfikacji $HP^k(x_1)$ dla układu przełączającego realizującego F^1

Kombinacyjny układ przełączający posiada:

$HSO^1(x_2)$
 $HSO^1(x_1)$

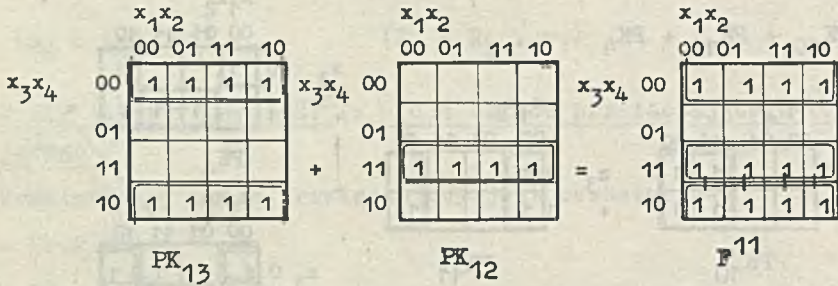
P r z y k ł a d 2.6.1.4

Rys. 14 przedstawia schemat logiczny trójwyjściowego i dziewięciowarstwowego kombinacyjnego układu przełączającego.

Algorytm I^k

- Krok 2 i 3

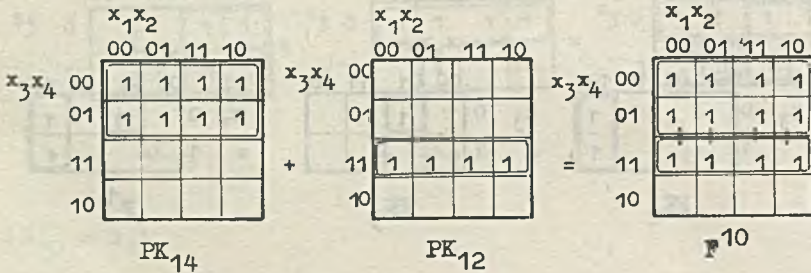
$$F^{11} = PK_{13} + PK_{12}$$



$$(\bar{S}_{13} = \bar{x}_4)$$

$$(\bar{S}_{12} = x_3 \cdot x_4)$$

$$F^{10} = PK_{14} + PK_{12}$$



$$(\bar{S}_{14} = \bar{x}_3)$$

- Krok 4 i 5

$$F^9 = PK_{10} + PK_{11}$$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01				
	11	1	1	1	1
	10	1	1	1	1

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01	1	1	1	1
	11	1	1	1	1
	10				

 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01	1	1	1	1
	11	1	1	1	1
	10	1	1	1	1

PK_{10}
 PK_{11}
 F^9

Wynik identyfikacji $HP^k(x_i)$ w układzie przełączającym realizującym F^9

Układ przełączający posiada:

$$HD^1(x_3)$$

$$HD^1(x_4)$$

- Krok 6

$$F^8 = PK_{15} + PK_9$$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1			1
	01	1			1
	11	1			1
	10	1			1

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1	1	1
	01				
	11	1	1	1	1
	10				

 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1	1	1
	01	1			1
	11	1	1	1	1
	10	1			1

PK_{15}
 PK_9
 F^8

$(S_{15} = \bar{x}_2)$

$$F^7 = PK_8 + PK_9$$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01		1	1	
	11				
	10		1	1	

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1	1	1
	01				
	11	1	1	1	1
	10				

 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1	1	1
	01		1	1	
	11	1	1	1	1
	10		1	1	

PK_8
 PK_9
 F^7

$$F^6 = PK_{15} + PK_8$$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1			1
	01	1			1
	11	1			1
	10	1			1

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01		1	1	
	11				
	10			1	1

 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1			1
	01	1	1	1	1
	11	1			1
	10	1	1	1	1

PK_{15}
 PK_8
 F^6

$$F^5 = PK_6 + PK_7$$

		x_1x_2			
		00	01	11	10
x_3x_4	00		1	1	
	01				
	11			1	1
	10				

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00				
	01	1			1
	11				
	10	1			1

 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00		1	1	
	01	1			1
	11			1	1
	10	1			1

PK_6
 PK_7
 F^5

Wynik identyfikacji $HP^k(x_1)$ w układzie przełączającym realizującym F^5

$HD^1(x_3)$

$HD^1(x_2)$

$HD^1(x_4)$

$HD^4(x_3)$

$HD^4(x_4)$

$$F^4 = PK_{16} + PK_5$$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1		
	01	1	1		
	11	1	1		
	10	1	1		

 $+$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1			1
	01		1	1	
	11	1			1
	10			1	1

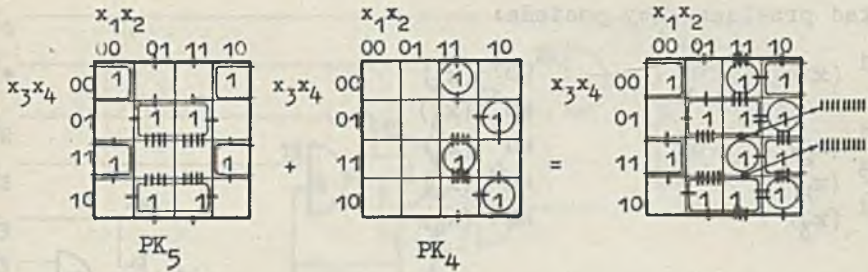
 $=$

		x_1x_2			
		00	01	11	10
x_3x_4	00	1	1		1
	01	1	1	1	
	11	1	1		1
	10	1	1	1	

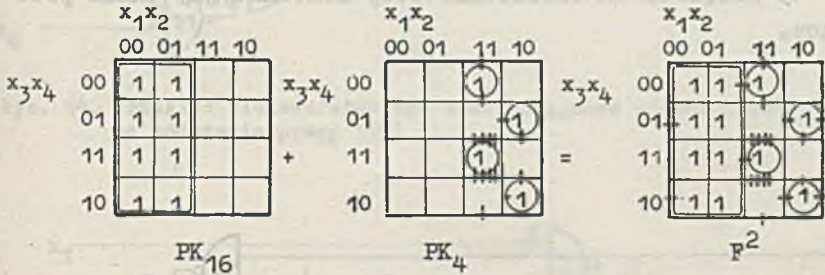
PK_{16}
 PK_5
 F^4

$$(\bar{S}_{16} = \bar{x}_1)$$

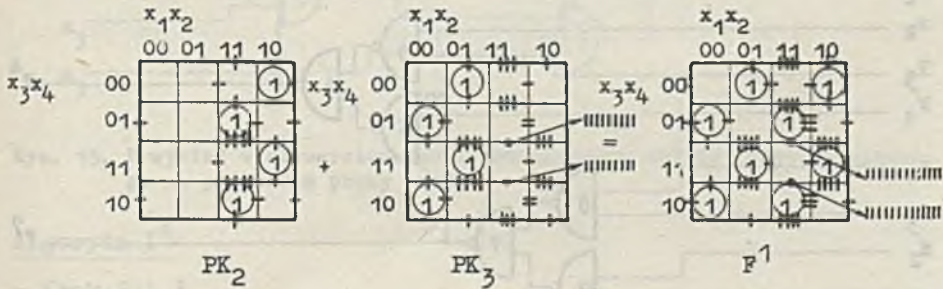
$$F^3 = PK_5 + PK_4$$



$$F^2 = PK_{16} + PK_4$$



$$F^1 = PK_2 + PK_3$$



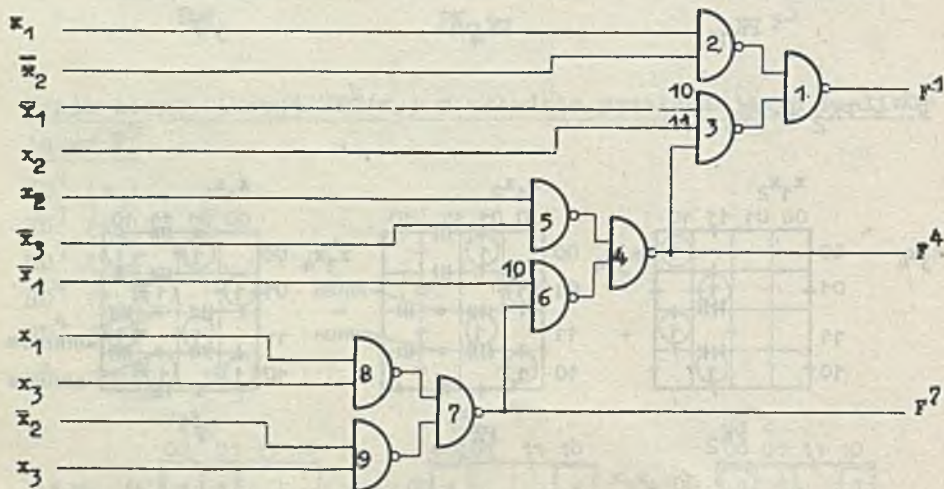
Wynik identyfikacji $HP^k(x_i)$ w układzie przełączającym realizującym F^1

Układ przełączający posiada:

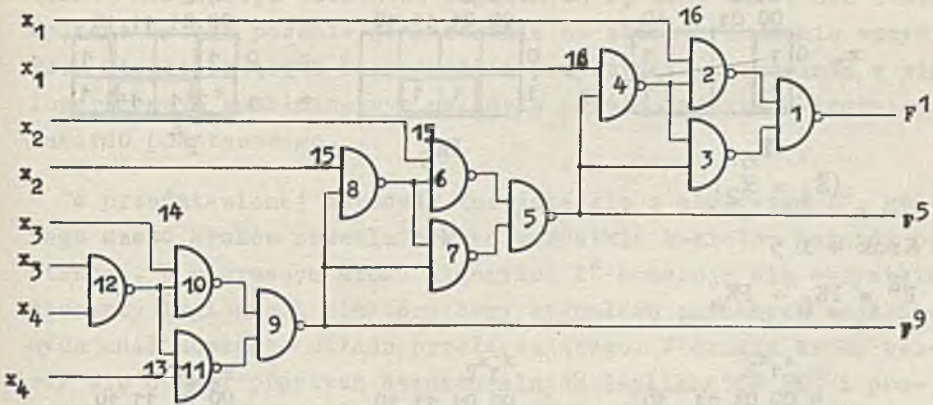
$HD^1(x_1)$	$HD^4(x_3)$
	$HD^{13}(x_3)$
$HD^1(x_2)$	$HD^1(x_4)$
$HD^3(x_2)$	$HD^4(x_4)$
$HD^1(x_3)$	$HD^{13}(x_4)$

Przykład 2.6.1.5

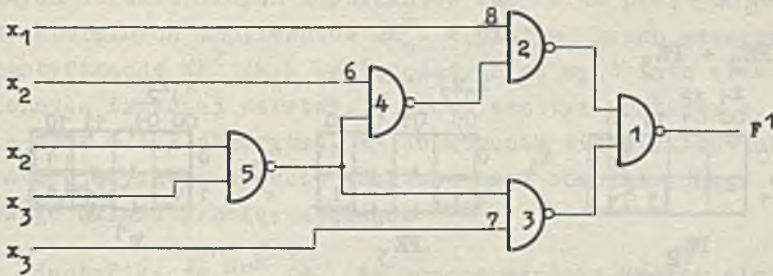
Rys. 15 przedstawia czterowarstwowy kombinacyjny układ przełączający.



Rys. 13. Przykład wielowarstwowego kombinacyjnego układu przełączającego na podstawie pracy [55]



Rys. 14. Przykład wielowarstwowego kombinacyjnego układu przełączającego na podstawie pracy [55]



Rys. 15. Przykład wielowarstwowego kombinacyjnego układu przełączającego na podstawie pracy [13]

Algorytm I^k

- Krok 2 1 3

$$F^3 = PK_7 + FK_5$$

		$x_1 x_2$			
		00	01	11	10
$x_3 x_4$	00	1	1	1	1
	1				

PK₇

$$(\bar{S}_7 = \bar{x}_3)$$

		$x_1 x_2$			
		00	01	11	10
x_3	0				
	1		1	1	

FK₅

$$(\bar{S}_5 = x_2 \cdot x_3)$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1	1	1
	1		1	1	

F³

$$F^4 = PK_6 + PK_5$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1			1
	1	1			1

$$PK_6$$

$(\bar{S}_6 = \bar{x}_2)$

		$x_1 x_2$			
		00	01	11	10
x_3	0				
	1		1	1	

$$PK_5$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1			1
	1	1	1	1	1

$$F^4$$

- Krok 4 i 5

$$F^2 = PK_8 + PK_4$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1		
	1	1	1		

$$PK_8$$

$(\bar{S}_8 = \bar{x}_1)$

		$x_1 x_2$			
		00	01	11	10
x_3	0		1	1	
	1				

$$PK_4$$

		$x_1 x_2$			
		00	01	11	10
x_3	0	1	1	1	
	1	1	1		

$$F^2$$

$$F^1 = PK_2 + PK_3$$

		$x_1 x_2$			
		00	01	11	10
x_3	0				1
	1			1	1

$$PK_2$$

		$x_1 x_2$			
		00	01	11	10
x_3	0				
	1	1			1

$$PK_3$$

		$x_1 x_2$			
		00	01	11	10
x_3	0				1
	1	1		1	1

$$F^1$$

Wynik identyfikacji $HP^k(x_1)$ w układzie przełączającym realizującym F^1

Układ przełączający posiada:

$HS1^1(x_2)$

$HS0^1(x_2)$

$HD^1(x_3)$

2.7. Ogólna charakterystyka metody

Opisana metoda identyfikacji $HP^k(x_1)$ pozwala na wykrycie każdego k-krotnego hazardu podstawowego na wyjściu każdego ele-

mentu logicznego wielowarstwowego kombinacyjnego układu przełączającego realizującego funkcje n -zmiennych i zbudowanego na dowolnej kombinacji elementów logicznych I, LUB, NAND, NOR i NIE. Opisana metoda pozwala jednocześnie na zidentyfikowanie wszystkich miejsc będących bezpośrednią przyczyną występowania w wielowarstwowym kombinacyjnym układzie przełączającym k -krotnego hazardu podstawowego.

W przedstawionej metodzie korzysta się z algorytmu I^k , którego sześć kroków pozwala wykryć wszystkie k -krotne hazardy podstawowe. W pierwszym kroku algorytmu I^k numeruje się wszystkie elementy logiczne i niektóre tory sygnałowe zmiennych wejściowych analizowanego układu przełączającego. W drugim kroku tworzy się obrazy prostych strukturalnych implikantów PK_0^m i prostych strukturalnych implicentów QK_0^m . W kroku trzecim przeprowadza się identyfikację $HP^k(x_1)$ występujących na wyjściach elementów logicznych drugiej warstwy (licząc do wejścia układu przełączającego). W czwartym kroku tworzy się obrazy prostych i złożonych strukturalnych implikantów PK_0^m oraz prostych i złożonych strukturalnych implicentów QK_0^m . W piątym kroku przeprowadza się identyfikację $HP^k(x_1)$ występujących na wyjściach elementów logicznych trzeciej warstwy. W kroku szóstym powtarza się cyklicznie krok 4 i 5 algorytmu, aż do momentu zidentyfikowania $HP^k(x_1)$ występujących na wyjściu analizowanego kombinacyjnego wielowarstwowego układu przełączającego.

Identyfikację $HP^k(x_1)$ za pomocą sześciu kroków algorytmu I^k przeprowadzać można dwoma drogami:

- za pomocą składania siatek Karnaugh'a
- za pomocą składania tablicy PQK_0^m

W pierwszym przypadku proste i złożone strukturalne implikanty PK_0^m oraz proste i złożone implicenty strukturalne QK_0^m przedstawia się za pomocą siatek Karnaugh'a. W przypadku drugim tablica PQK_0^m jest obrazem prostych i złożonych strukturalnych implikantów PK_0^m oraz prostych i złożonych strukturalnych implicentów QK_0^m .

W stosunku do istniejących już metod identyfikacji hazardu podstawowego podanych w pracach [18], [31], [35] przedstawiona metoda dodatkowo pozwala:

- a) na systematyczną identyfikację na wyjściu każdego elementu logicznego wielowarstwowego kombinacyjnego układu przełączającego realizującego funkcje n -zmiennych
 - k -krotnego hazardu statycznego ze względu na zmienną x_i w warunkach działania
 - k -krotnego hazardu statycznego ze względu na zmienną x_i w warunkach niedziałania
 - k -krotnego hazardu dynamicznego ze względu na zmienną x_i oraz identyfikację wszystkich miejsc w układzie, które są bezpośrednią przyczyną istnienia każdego k -krotnego hazardu podstawowego (rozdział 3).
- b) uniknąć
 - tworzenia funkcji przejściowej
 - szukania zbiorów P
 - szukania zbiorów S
 - szukania zbiorów 1 (stabilnych zbiorów 1)
 - szukania zbiorów 0 (stabilnych zbiorów 0)

Szczególnie metoda identyfikacji $HP^k(x_i)$ za pomocą składania siatek Karnaugh, w stosunku do metod przedstawionych w [31], [35], charakteryzuje się zdaniem autora mniejszą pracochłonnością i wydaje się mieć zalety prostej inżynierskiej metody. Stosowanie jej ograniczone jest jednak do układów przełączających realizujących funkcje 6-zmiennych (siatka Karnaugh).

Metoda identyfikacji $HP^k(x_i)$ drogą składania tablicy P,K_0^m została zaprogramowana na komputerze ICL 4-50 o pamięci operacyjnej 100 kbajtów. W pracy [75] opisano rezultaty obliczeń prowadzonych na tym komputerze. Wynika z nich, że dla liczby zmiennych wejściowych zmieniającej się w granicach od 4 do 8 oraz dla liczby warstw układu przełączającego zmieniającej się w granicach od 3 do 5 i w zależności od struktury połączeń, czas pracy programu zmieniał się w granicach od 35 do 90 s bez użycia pamięci zewnętrznych.

3. ELIMINACJA k-KROTNEGO HAZARDU PODSTAWOWEGO W UKŁADACH PRZEŁĄCZAJĄCYCH

3.1. Informacje wstępne

Eliminacja $HP^k(x_i)$ w ujęciu spotykanym w literaturze specjalistycznej [3], [5], [18], [31], [42], [48], [58], [64] i [69] dotyczy przede wszystkim kombinacyjnych układów przełączających opisanych za pomocą alternatywnego wyrażenia normalnego lub konjunkcyjnego wyrażenia normalnego.

W przytoczonych wyżej pracach w celu wyeliminowania $HP^k(x_i)$ proponuje się uzupełniać kombinacyjny układ przełączający dodatkowymi elementami logicznymi i dodatkowymi torami sygnałowymi w taki sposób, aby nowo powstałe wyrażenie strukturalne opisujące rozbudowany układ przełączający przedstawiało sumę prostych implikantów pokrywających wszystkie pary sąsiednich stanów wejścia A i A' , dla których $f(A) = f(A') = 1$ lub iloczyn implikantów prostych pokrywających wszystkie pary A i A' , dla których $f(A) = f(A') = 0$. Czynność tę część autorów nazywa deminimalizacją wyrażenia strukturalnego (układowej funkcji przełączającej).

W innych pracach np. [28] autorzy proponują uzupełniać kombinacyjny układ przełączający kondensatorami odpowiedniej pojemności, aby nowo powstały rozkład wartości czasów opóźnienia w układzie przełączającym uniemożliwił pojawienie się na wyjściu przełączającego układu niepożądanych krótkotrwałych impulsów.

Autor prac [19] i [20] proponuje natomiast uzupełniać wyjścia kombinacyjnych układów przełączających odpowiednimi filtrami, które nie przepuszczają krótkotrwałych impulsów. Chociaż uzupełnienie kombinacyjnych układów przełączających kondensatorami odpowiedniej pojemności, jak również przyłączanie do nich filtrów, umożliwi wyeliminowanie zidentyfikowanych $HP^k(x_i)$ w każdym wielowarstwowym układzie przełączającym, to jednak eliminacja $HP^k(x_i)$ w takim układzie drogą deminimalizacji wyrażenia strukturalnego jest niejednokrotnie drogą prostszą, tańszą

i mającą ponadto inne dodatkowe zalety (np. nierozszerzony asortyment elementów użytych do budowy układu przełączającego).

3.2. Wykorzystanie algorytmu I^k do eliminacji k-krotnych hazardów podstawowych w wielowarstwowych kombinacyjnych układach przełączających

Twierdzenia 2.4.1.1., 2.4.1.2., 2.4.1.3., 2.4.1.4., 2.4.1.5., i 2.4.1.6. określają warunki jakie muszą być spełnione, aby układy przełączające realizujące funkcje F^m posiadały $HP^k(x_i)$. Gdy problem postawi się odwrotnie, to z tych samych twierdzeń wynikają równocześnie niżej przedstawione warunki jakie powinny być spełnione, aby wielowarstwowe układy przełączające realizujące funkcje F^m nie posiadały $HP^k(x_i)$. Brzmia one następująco:

I. Jeżeli element logiczny B^m jest elementem NAND albo elementem LUB, to układ przełączający realizujący funkcję F^m nie posiada na pewno:

- a) $HS\uparrow^k(x_i)$, gdy istnieje przynajmniej jeden bezhazardowy ze względu na zmienną x_i strukturalny implikant PK_0^m
- b) $HSO^k(x_i)$, gdy nie ma ani jednego hazardowego strukturalnego implikantu PK_1^m posiadającego $HSO^{\beta_i}(x_i)$
- c) $HD^k(x_i)$, gdy nie ma ani jednego hazardowego strukturalnego implikantu PK_1^m posiadającego $HSO^{\beta_i}(x_i)$, oraz nie ma ani jednego hazardowego strukturalnego implikantu PK_w^m posiadającego $HD^{\beta_i}(x_i)$.

II. Jeżeli element logiczny B^m jest elementem NOR lub elementem I, to układ przełączający realizujący funkcję F^m nie posiada na pewno:

- a) $HS\uparrow^k(x_i)$, gdy nie ma ani jednego hazardowego strukturalnego implikentu QK_h^m posiadającego $HS\uparrow^{\alpha_h}(x_i)$
- b) $HSO^k(x_i)$, gdy istnieje przynajmniej jeden bezhazardowy ze względu na zmienną x_i strukturalny implikent QK_0^m

c) $HD^k(x_1)$, gdy nie ma ani jednego hazardowego strukturalnego implicentu QK_h^m posiadającego $HS1^{a_n}(x_1)$ oraz nie ma ani jednego hazardowego strukturalnego implicentu QK_w^m posiadającego $HD^{tw}(x_1)$.

W takim razie w celu całkowitego wyeliminowania $HP^k(x_1)$ z wielowarstwowego układu przełączającego należy zidentyfikować go za pomocą algorytmu I^k . Następnie w zależności od tego, czy elementy logiczne B^m są elementami NAND, LUB, NOR oraz I należy pokryć bezhazardowymi, ze względu na zmienną x_1 , strukturalnymi implikantami PK_0^m lub bezhazardowymi, ze względu na zmienną x_1 , strukturalnymi implicentami QK_0^m , wszystkie niepokryte pary sąsiednich stanów wejścia A i A' (różniących się stanem logicznym a_1) wszystkich funkcji F^m , dla których w pierwszym przypadku $f^m(A) = f^m(A') = 1$, a w drugim przypadku $f^m(A) = f^m(A') = 0$.

Przykład 3.2.1

Rys. 5 przedstawia schemat logiczny potencjałowego asynchronicznego układu przełączającego z przerzutnikami o dominującym wejściu wpisującym.

Przedstawiona w rozdziale 1.3 analiza działania układu przełączającego realizującego funkcję \bar{F}_{R1} wykazała możliwość szkodliwego wpływu hazardu podstawowego na prawidłową pracę układu sekwencyjnego. Obecnie przeprowadzi się eliminację wszystkich hazardów podstawowych, które ma układ przełączający realizujący funkcję \bar{F}_{R1} .

Algorytm I^k

- Krok 2 i 3

$$F^9 = PK_{12} + PK_{21} + PK_{13}$$

		$x_1 x_2 x_3$							
		000	001	010	011	100	101	110	111
$y_1 y_2$	00	1	1					1	1
	01	1	1					1	1
	11	1	1					1	1
	10	1	1					1	1

PK₁₂ ($\bar{s}_{12} = x_3$)

		$x_1 x_2 x_3$								
		000	001	010	011	100	101	110	111	
$y_1 y_2$	00	1	1						1	1
	01	1	1						1	1
	11	1	1						1	1
	10	1	1						1	1

PK₂₁ ($\bar{s}_{21} = \bar{x}_2$)

		$x_1 x_2 x_3$								
		000	001	010	011	100	101	110	111	
$y_1 y_2$	00									
	01					1	1			
	11					1	1			
	10									

PK₁₃ ($\bar{s}_{13} = x_1 x_2 y_2$)

		$x_1 x_2 x_3$								
		000	001	010	011	100	101	110	111	
$y_1 y_2$	00	1	1	1				1	1	1
	01	1	1	1				1	1	1
	11	1	1	1				1	1	1
	10	1	1	1				1	1	1

F⁹

$F^{10} = PK_{22} + PK_{13} + PK_{20}$

		$x_1 x_2 x_3$							
		000	001	010	011	100	101	110	111
$y_1 y_2$	00	1	1	1	1				
	01	1	1	1	1				
	11	1	1	1	1				
	10	1	1	1	1				

PK₂₂ ($\bar{s}_{22} = \bar{x}_1$)

		$x_1 x_2 x_3$							
		000	001	010	011	100	101	110	111
$y_1 y_2$	00								
	01					1	1		
	11					1	1		
	10								

PK₁₃

		$x_1 x_2 x_3$							
		000	001	010	011	100	101	110	111
$y_1 y_2$	00	1			1	1			1
	01	1			1	1			1
	11	1			1	1			1
	10	1			1	1			1

PK₂₀ ($\bar{s}_{20} = \bar{x}_3$)

		$x_1 x_2 x_3$							
		000	001	010	011	100	101	110	111
$y_1 y_2$	00	1	1	1	1	1			1
	01	1	1	1	1	1	1		1
	11	1	1	1	1	1	1		1
	10	1	1	1	1	1	1		1

F¹⁰

$$F^{11} = PK_{13} + PK_{20} + PK_{23}$$

		x_1x_2							
		000	001	011	010	110	111	101	100
y_1y_2	00								
	01					1	1		
	11					1	1		
	10								

PK₁₃

		x_1x_2							
		000	001	011	010	110	111	101	100
y_1y_2	00	1				1	1		1
	01	1				1	1		1
	11	1				1	1		1
	10	1				1	1		1

PK₂₀

		x_1x_2							
		000	001	011	010	110	111	101	100
y_1y_2	00	1	1	1	1	1	1	1	1
	01								
	11								
	10	1	1	1	1	1	1	1	1

PK₂₃ ($\bar{S}_{23} = \bar{y}_2$)

		x_1x_2							
		000	001	011	010	110	111	101	100
y_1y_2	00	1	1	1	1	1	1	1	1
	01	1				1	1	1	1
	11	1				1	1	1	1
	10	1	1	1	1	1	1	1	1

F¹¹

- Krok 4 i 5

$$\bar{F}_{R_1} = F^8 = PK_9 + PK_{10} + PK_{11}$$

		$x_1x_2x_3$							
		000	001	011	010	110	111	101	100
y_1y_2	00					1	1		
	01					1			
	11					1			
	10					1	1		

PK₉

		$x_1x_2x_3$							
		000	001	011	010	110	111	101	100
y_1y_2	00							1	1
	01							1	
	11							1	
	10							1	1

PK₁₀

		$x_1x_2x_3$							
		000	001	011	010	110	111	101	100
y_1y_2	00								
	01		1	1					1
	11		1	1					1
	10								

PK₁₁

		$x_1x_2x_3$							
		000	001	011	010	110	111	101	100
y_1y_2	00					1	1	1	1
	01		1	1	1			1	1
	11		1	1	1			1	1
	10					1	1	1	1

F⁸ = \bar{F}_{R_1}

Wynik identyfikacji $HP^k(x_1)$

Układ przełączający realizujący funkcję \bar{F}_{R1} posiada:

- HS1¹ (x_3) (wynika z niespełnienia warunku Ia podanego w p. 3.2)
- HS0¹ (x_2) (wynika z niespełnienia warunku Ib podanego w p. 3.2)
- HD¹ (x_1) (wynika z niespełnienia warunku Ic podanego w p. 3.2)
- HD¹ (y_2) (wynika z niespełnienia warunku Ic podanego w p. 3.2)

Eliminacja HD¹(x_1)

W celu spełnienia warunku Ic (p. 3.2) pokrywa się pary sąsiednich stanów wejścia (0,1,1,0,1)^{*}, (1,1,1,0,1) oraz (0,1,1,1,1), (1,1,1,1,1) funkcji F^{10} nowym strukturalnym implikantem PK_{24} ($\bar{S}_{24} = y_2 \cdot x_2$).

Implikant strukturalny PK_{13} w tej sytuacji jest zbędny.

		$x_1 x_2 x_3$							
		000	001	011	010	110	111	101	100
$y_1 y_2$	00	1	1	1	1	1			1
	01	1	1	1	1	1	1		1
	11	1	1	1	1	1	1		1
	10	1	1	1	1	1			1

F^{10}

PK₂₄

Eliminacja HD¹(y_2)

W celu spełnienia warunku Ic (p. 3.2) pokrywa się pary sąsiednich stanów wejścia (1,1,1,0,0), (1,1,1,0,1) oraz (1,1,1,1,1), (1,1,1,1,0) funkcji F^{11} nowym strukturalnym implikantem PK_{25} ($\bar{S}_{25} = x_1 \cdot x_2$). Implikant strukturalny PK_{13} w tej sytuacji jest zbędny.

* Stan wejścia określany jest dla następującej kolejności zmiennych wejściowych (x_1, x_2, x_3, y_1, y_2)

		$x_1 x_2 x_3$							
		000	001	011	010	110	111	101	100
$y_1 y_2$	00	1	1	1	1	1	1	1	1
	01	1			1	1	1		1
	11	1			1	1	1		1
	10	1	1	1	1	1	1	1	1

F^{11}

PK₂₅

Eliminacja HSO¹ (x_2)

W celu spełnienia warunku Ib (p. 3.2.) pokrywa się pary sąsiednich stanów wejścia (1,1,0,0,1), (1,0,0,0,1) oraz (1,1,0,1,1) (1,0,0,1,1) funkcji F^9 nowym strukturalnym implikantem PK₂₆ ($\bar{S}_{26} = x_1 \cdot y_2$). Implikant strukturalny PK₁₃ jest w tej sytuacji zbędny.

		$x_1 x_2 x_3$							
		000	001	011	010	110	111	101	100
$y_1 y_2$	00	1	1	1			1	1	1
	01	1	1	1		1	1	1	1
	11	1	1	1		1	1	1	1
	10	1	1	1			1	1	1

F^9

PK₂₆

Eliminacja HS¹ (x_3)

W celu spełnienia warunku Ia (p.3.2.) pokrywa się pary sąsiednich stanów wejścia (0,1,1,0,1), (0,1,0,0,1) oraz (0,1,1,1,1), (0,1,0,1,1), funkcji F^8 nowym strukturalnym implikantem PK₂₇ ($\bar{S}_{27} = \bar{x}_1 \cdot x_2 \cdot y_2$), a pary sąsiednich stanów wejścia (1,1,0,0,0), (1,1,1,0,0) oraz (1,1,0,1,0), (1,1,1,1,0) funkcji F^8 nowym strukturalnym implikantem PK₂₈ ($\bar{S}_{28} = x_1 \cdot x_2 \cdot \bar{y}_2$).

		$x_1 x_2 x_3$				
		000	001011	010110	111101100	
$y_1 y_2$	00		1	1	1	1
	01	1	1	1		1
	11	1	1	1		1
	10		1	1	1	1

F^8

FK₂₈
FK₂₇

Wyrażenie strukturalne bezhazardowego układu przełączającego realizującego \overline{F}_{R1} .

$$S^8 = \overline{\overline{x_3 \cdot x_2 \cdot x_1 \cdot y_2 \cdot x_1 \cdot x_3 \cdot x_2 \cdot y_2 \cdot x_3 \cdot y_2 \cdot x_1 \cdot x_2 \cdot x_1 \cdot x_2 \cdot y_2 \cdot x_1 \cdot x_2 \cdot y_2}}$$

Eliminacja $HP^k(x_1)$ wymaga dodatkowego użycia pięciu elementów NAND i czternastu wejść na elementy logiczne.

3.3. Wykorzystanie algorytmu I^k do eliminacji szkodliwych hazardów podstawowych w potencjałowych asynchronicznych automatach sekwencyjnych

W większości sekwencyjnych asynchronicznych układów przełączających część k-krotnych hazardów podstawowych jest nieszkodliwa. Hazardry te nie wpływają na poprawną pracę układu przełączającego i nie muszą być eliminowane. Są to więc wszystkie te $HP^k(x_1)$ (hazardry nieważne - rozdział 1.1.), które chociaż tkwią w strukturze układu przełączającego, to jednak w warunkach pracy automatu narzuconych przez pierwotną siatkę programu nie mogą wogóle pojawić się w postaci przekłamań w rzeczywistym układzie (np. $HP^k(x_1)$ w przejściach pomiędzy różnymi stanami niestabilnymi jak również w przejściach pomiędzy stanem niestabilnym a stanem stabilnym, który jest stanem docelowym).

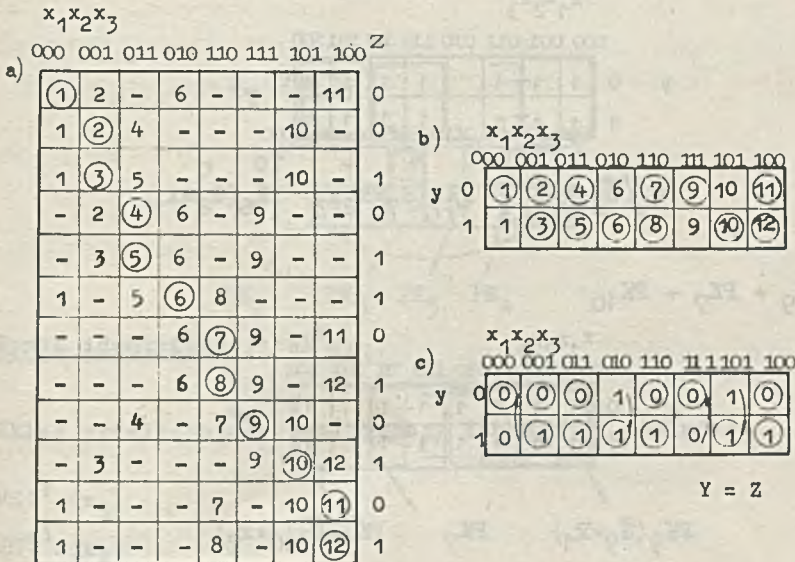
Są to także wszystkie te $HP^k(x_1)$, które pomimo tego, że w warunkach pracy automatu narzuconych przez pierwotną siatkę programu pojawiają się w pętlach sprzężeń zwrotnych sekwencyjnego układu, to jednak są od razu samoczynnie przez ten układ likwidowane (hazardry pozorne - rozdział 1.1.).

W pracy [26] przedstawiono podstawowe zasady identyfikacji w siatce przejść szkodliwych hazardów podstawowych, wykorzystu-

jące do tego celu warunki pracy automatu sekwencyjnego zawarte w pierwotnej siatce programu. Za pomocą tych zasad, algorytmu I^k i zasad eliminacji $HP^k(x_i)$ (warunki w rozdziale 3.2.) można projektować tańsze, prawidłowo pracujące, asynchroniczne automaty sekwencyjne.

P r z y k ł a d 3.3.1

Rys. 4 przedstawia schemat logiczny potencjałowego asynchronicznego automatu sekwencyjnego z pętlami logicznych sprzężeń zwrotnych, zawierającego hazard podstawowy. Pierwotną siatkę programu, zredukowaną siatkę programu oraz siatkę przejść tego układu przełączającego pokazuje rys. 16. Przedstawiona w rozdziale 1.3 analiza działania tego układu przełączającego wykazała możliwość szkodliwego wpływu hazardu podstawowego na prawidłową pracę układu sekwencyjnego.



Rys. 16 Siatki układu przełączającego przedstawionego na rys. 4.

- a) pierwotna siatka programu;
- b) zredukowana siatka programu;
- c) siatka stanów i przejść.

Obecnie przeprowadzi się eliminacje wszystkich szkodliwych hazardów podstawowych.

Algorytm I^k

- Krok 2^x i 3

F²

		$x_1 x_2 x_3$								
		000	001	011	010	110	111	101	100	
y	0	1	1	1	1	1	1	1	1	F ²
	1	1			1	1	1	1	1	

F³ = PK₆ + PK₈ + PK₇

		$x_1 x_2 x_3$								
		000	001	011	010	110	111	101	100	
y	0	1	1	1		1	1	1	1	F ³
	1	1	1	1		1	1	1	1	

PK₈ ($\bar{S}_8 = \bar{x}_2$)

PK₇ ($\bar{S}_7 = x_2 x_3$)

PK₆ ($\bar{S}_6 = x_1$)

F⁴ = PK₉ + PK₇ + PK₁₀

		$x_1 x_2 x_3$								
		000	001	011	010	110	111	101	100	
y	0	1	1	1	1	1	1		1	F ⁴
	1	1	1	1	1	1	1		1	

PK₉ ($\bar{S}_9 = \bar{x}_1$)

PK₇

PK₁₀ ($\bar{S}_{10} = \bar{x}_3$)

*) W celu uproszczenia procedury identyfikacji, w kroku 2 i 4 algorytmu I^k będzie się od razu rysować obrazy funkcji realizowanych na wyjściach elementów logicznych drugiej i trzeciej warstwy.

$$F^5 = PK_7 + PK_9 + PK_{11}$$

		$x_1 x_2 x_3$								
		000	001	011	010	110	111	101	100	
y	0	1	1	1	1	1	1	1	1	F^5
	1	1	1	1	1	1				

\swarrow PK_9 \swarrow PK_7 \swarrow $PK_{11} (S_{11}=y)$

- Krok 4 i 5

$$F^1 = PK_2 + PK_3 + PK_4 + PK_5$$

		$x_1 x_2 x_3$								
		000	001	011	010	110	111	101	100	
y	0			1				1		F^1
	1	1	1	1	1			1	1	

\swarrow PK_2 \swarrow PK_3 \swarrow PK_5 \swarrow PK_4

Wynik identyfikacji $HP^k(x_1)$

Układ przełączający realizujący funkcję F^1 posiada:

$$HS1^1(x_1)$$

$$HS1^1(x_3)$$

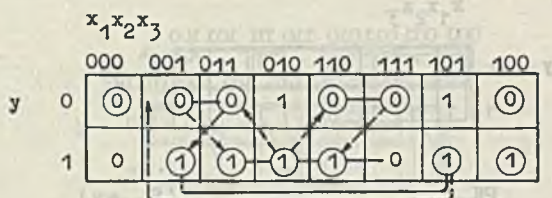
$$HS0^1(x_2)$$

$$HS0^1(x_3)$$

$$HD^1(x_3)$$

Identyfikacja szkodliwych $HP^k(x_1)$

Zidentyfikowane $HP^k(x_1)$ nanosi się na siatkę stanów i przejść (rys. 16c).



- HS¹(x_3) przy przejściu ⑥ → ⑤ może spowodować przeskok do niewłaściwego stanu stabilnego ④
- HS¹(x_1) przy przejściu ⑥ → ⑧ może spowodować przeskok do niewłaściwego stanu stabilnego ⑦
- HSO¹(x_2) przy przejściu ② → ④ może spowodować przeskok do niewłaściwego stanu stabilnego ⑤
- HSO¹(x_2) przy przejściu ④ → ② może spowodować przeskok do niewłaściwego stanu stabilnego ③
- HSO¹(x_3) przy przejściu ⑨ → ⑦ może spowodować przeskok do niewłaściwego stanu stabilnego ⑧
- HD¹(x_3) przy przejściu ⑧ → 9 jest hazardem pozornym.

Eliminacja $HSO^1(x_2)$

W celu spełnienia warunku Ib (p. 3.2.) pokrywa się parę sąsiednich stanów wejścia $(0,0,1,0)^*$, $(0,1,1,0)$ funkcji F^3 nowym strukturalnym implikantem PK_{12} ($\bar{S}_{12} = x_3$). Implikant strukturalny PK_7 jest w tej sytuacji zbędny.

$x_1 x_2 x_3$

	000	001	011	010	110	111	101	100
y = 0	1	1	1		1	1	1	1
y = 1	1	1	1		1	1	1	1

F^3 PK_{12}

Eliminacja $HSO^1(x_3)$

W celu spełnienia warunku Ib (p. 3.2.) pokrywa się parę sąsiednich stanów wejścia $(1,1,0,0)$, $(1,1,1,0)$ funkcji F^4 nowym strukturalnym implikantem PK_{13} ($\bar{S}_{13} = x_2$). Implikant strukturalny PK_7 jest w tej sytuacji zbędny

$x_1 x_2 x_3$

	000	001	011	010	110	111	101	100
y = 0	1	1	1	1	1	1		1
y = 1	1	1	1	1	1	1		1

F^4 PK_{13}

* Stan wejścia określony jest dla następującej kolejności zmiennych wejściowych x_1, x_2, x_3, y .

Eliminacja HS¹(x₃) i HS¹(x₁)

W celu spełnienia warunku Ia (p. 3.2.) pokrywa się parę sąsiednich stanów wejścia (0,1,1,1), (0,1,0,1) oraz (0,1,0,1), (1,1,0,1) funkcji F¹ nowym bezhazardowym strukturalnym implikantem PK₁₄ ($\bar{S}_{14} = x_2 \cdot y \cdot \bar{x}_1 \cdot x_3$).

Eliminacja pozostałego HS¹(x₁)

W celu spełnienia warunku Ia (p. 3.2.) pokrywa się parę sąsiednich stanów wejścia (0,0,1,1), (1,0,1,1), funkcji F¹ nowym strukturalnym implikantem PK₁₅ ($\bar{S}_{15} = \bar{x}_2 \cdot x_3 \cdot y$).

		x ₁ x ₂ x ₃							
		000	001	011	011	110	111	101	100
y	0				1			1	
	1	1	1	1	1			1	1

PK₁₅
PK₁₄
F¹

Wyrażenie strukturalne układu przełączającego pozbawionego szkodliwego hazardu podstawowego

$$S^1 = \overline{\overline{\bar{x}_1 \cdot x_3 \cdot y \cdot \bar{x}_1 \cdot x_2 \cdot x_3 \cdot x_1 \cdot x_3 \cdot \bar{x}_2 \cdot x_1 \cdot y \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot x_2 \cdot y \cdot \bar{x}_1 \cdot \bar{x}_3 \cdot \bar{x}_2 \cdot x_3 \cdot y}}$$

Eliminacja szkodliwych HP^k(x₁) wymaga dodatkowego użycia pięciu elementów NAND i dwunastu wejść na elementy logiczne.

P r z y k ł a d 3.3.2

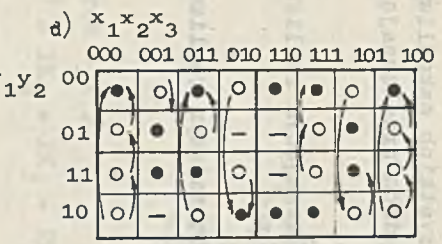
Rys. 17 przedstawia pierwotną siatkę programu, zredukowaną siatkę programu, siatkę stanów oraz siatkę przejść automatu sekwencyjnego przedstawionego na rys. 5.

a) $x_1 x_2 x_3$

	000	001	011	010	110	111	101	100
$y_1 y_2$	00	- 2 -	- - -	7 1 9				
01	10	2 3 -	- - -	- 1 -				
11	- 2	3 6 -	7 - -	- -				
10	- -	11 -	5 4 1 -					
	- -	- 6	5 4 - 9					
	10	- 11	6 5 - -					
	- -	11 -	8 7 1 -					
	- -	- 6	8 7 - 9					
	10	- -	- 8 - 1 9					
	10	12 -	6 - - 9					
	- 12	11 6 -	7 - -					
	10	12 11 -	- - - 13 -					
	- 12	- - -	7 13 9					

b) $x_1 x_2 x_3$

	000	001	011	010	110	111	101	100
$y_1 y_2$	00	10	12 11	6 8 7	1 9			
01	10	12 11	- - -	7 13 9				
11	10	2 3 6 -	7 1 9					
10	10	- 11	6 5 4 1 9					



c) $x_1 x_2 x_3$

	000	001	011	010	110	111	101	100
$y_1 y_2$	00	00	01 00	10 00	00 10 00			
01	00	01 00	-- --	00 01 00				
11	00	11 11	10 --	01 11 00				
10	00	-- 00	10 10	10 11 00				

Rys. 17. Siatki układu przełączającego przedstawionego na rys. 5

- a) pierwotna siatka programu;
- b) zredukowana (zakodowana) siatka programu;
- c) siatka stanów;
- d) siatka przejść.

Obecnie przeprowadzi się eliminację wszystkich szkodliwych hazardów podstawowych występujących w układach przełączających realizujących funkcje \bar{F}_{S1} i \bar{F}_{R1} .

Identyfikację $HP^k(x_1)$ w układzie przełączającym realizującym funkcję \bar{F}_{R1} przedstawiono w rozdziale 3.2.

Identyfikację $HP^k(x_1)$ w układzie przełączającym realizującym funkcję \bar{F}_{S1} przedstawiono niżej.

Algorytm I^k

- Krok 2 i 3

F^2

		$x_1 x_2 x_3$					
		0000	0110	110	111	101	100
$y_1 y_2$	00	1	1	1	1	1	1
	01			1	1	1	1
	11			1	1	1	1
	10	1	1	1	1	1	1

F^2

$$\bar{S}^2 = \bar{y}_2 + x_2$$

$$F^3 = PK_6 + PK_7$$

		$x_1 x_2 x_3$					
		000	0010	110	111	101	100
$y_1 y_2$	00			1	1	1	1
	01			1	1	1	1
	11			1	1	1	1
	10			1	1	1	1

F^3

$$PK_6 (\bar{S}_6 = x_2)$$

$$PK_7 (\bar{S}_7 = x_1 \cdot x_3)$$

$$F^4 = PK_7 + PK_{20}$$

		$x_1 x_2 x_3$					
		000	001	011	010	110	101
$y_1 y_2$	00	1			1	1	1
	01	1			1	1	1
	01	1			1	1	1
	10	1			1	1	1

F^4

$$PK_{20} (\bar{S}_{20} = \bar{x}_3)$$

$$PK_7$$

F⁵

$$x_1 x_2 x_3$$

$$000 \ 001011 \ 010 \ 110 \ 111101 \ 100$$

$y_1 y_2$	00	1	1	1	1				1	1
	01	1	1	1	1				1	1
	11	1	1	1	1				1	1
	10	1	1	1	1				1	1

F⁵

$$F^5 = \bar{x}_1 + \bar{x}_2$$

- Krok 4 i 5

$$F^1 = PK_2 + PK_3 + PK_4 + PK_5$$

$$x_1 x_2 x_3$$

$$000 \ 001011 \ 010110111 \ 101100$$

$y_1 y_2$	00	1	1	1		1	1	-1		
	01	1	1	1		1	1	1	1	
	11	1	1	1		1	1	1	1	
	10	1	1	1		1	1	-1		

F¹ = \bar{F}_{S1}

PK_3 PK_4 PK_5 PK_2

Wynik identyfikacji HP^k(x₁)

Układ przełączający realizujący funkcję \bar{F}_{S1} posiada:

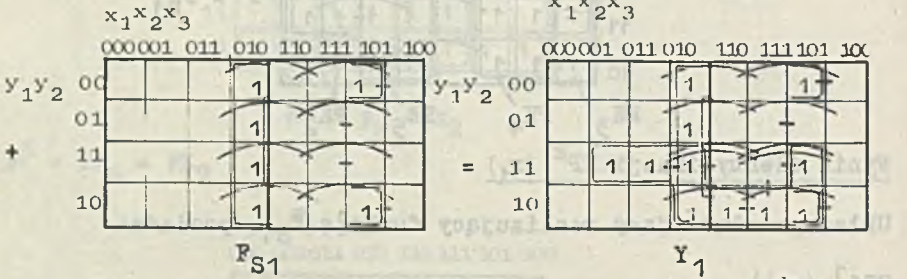
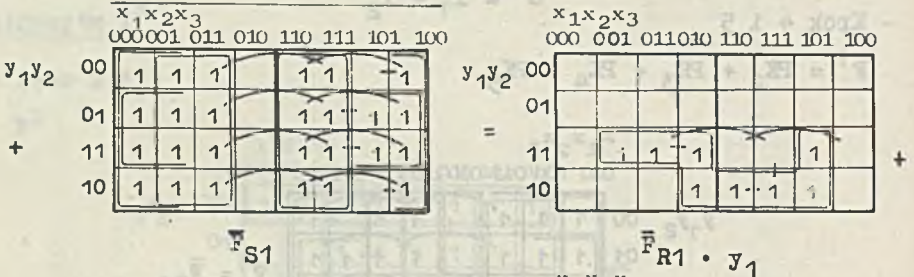
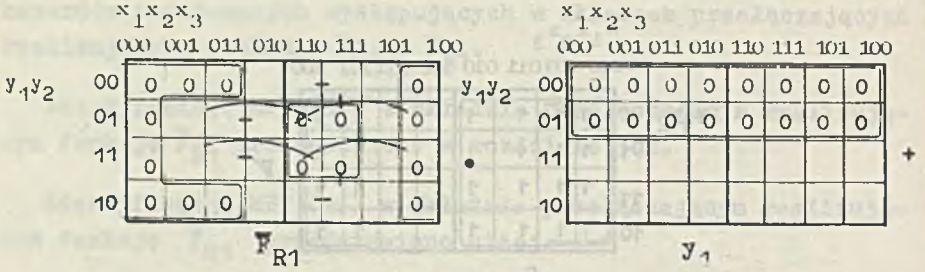
- | | | |
|------------------------------------|---|--|
| HS1 ¹ (x ₁) | } | (wynika z niespełnienia warunku I _c podanego w p. 3.2.) |
| HS1 ¹ (x ₂) | | |
| HS1 ¹ (x ₃) | | |
| HD ¹ (x ₃) | | |

Identyfikacja szkodliwych HP^k(x₁)

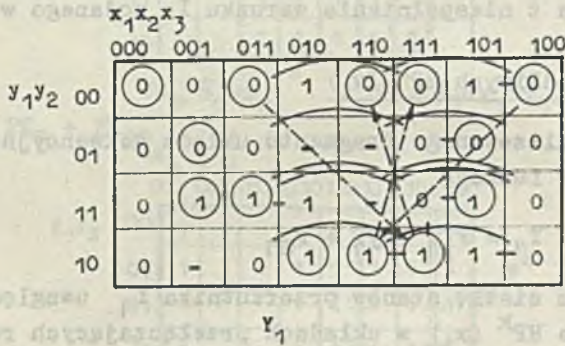
Przerzutnik analizowanego fragmentu układu sekwencyjnego realizuje następującą funkcję

$$Y_1 = \bar{F}_{R1} \cdot y_1 + F_{S1}$$

W związku z tym siatkę stanów przerzutnika Y₁ uwzględniającą zidentyfikowane HP^k(x₁) w układach przełączających realizujących funkcje \bar{F}_{S1} i \bar{F}_{R1} uzyskuje się po wymnożeniu i zsumowaniu poniższych siatek Karnaugh



Zidentyfikowane w siatce stanów przerzutnika Y_1 $H_2^k(x_1)$ nanosi się na siatkę przejść (rys. 17d)



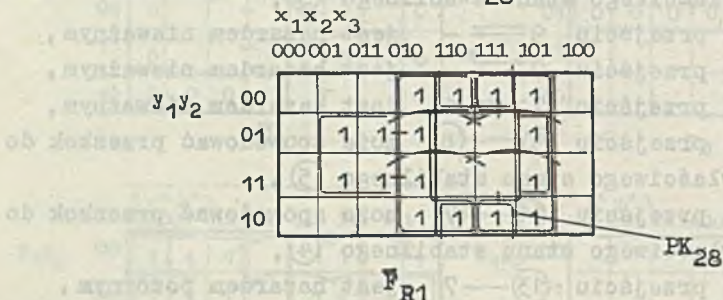
- $HS1^1(x_3)$ przy przejściu $(3) \rightleftharpoons 6$ jest hazardem nieważnym,
 $HS1^1(x_3)$ przy przejściu $(5) \rightarrow (4)$ może spowodować przeskok do niewłaściwego stanu stabilnego (7) ,
 $HS1^1(x_3)$ przy przejściu $(4) \rightarrow (5)$ może spowodować przeskok do niewłaściwego stanu stabilnego (8) ,
 $HSO^2(x_2)$ przy przejściu $9 \rightleftharpoons -$ jest hazardem nieważnym,
 $HSO^1(x_2)$ przy przejściu $9 \rightleftharpoons -$ jest hazardem nieważnym,
 $HSO^1(x_1)$ przy przejściu $7 \rightleftharpoons 11$ jest hazardem nieważnym,
 $HSO^1(x_2)$ przy przejściu $(9) \rightarrow (8)$ może spowodować przeskok do niewłaściwego stanu stabilnego (5) ,
 $HSO^1(x_1)$ przy przejściu $(11) \rightarrow (7)$ może spowodować przeskok do niewłaściwego stanu stabilnego (4) ,
 $HSO^1(x_2)$ przy przejściu $(13) \rightarrow 7$ jest hazardem pozornym,
 $HSO^1(x_2)$ przy przejściu $7 \rightarrow (13)$ jest hazardem nieważnym,
 $HD^1(x_1)$ przy przejściu $(1) \rightarrow 7$ jest hazardem pozornym,
 $HD^2(x_1)$ przy przejściu $7 \rightarrow (1)$ jest hazardem nieważnym,
 $HD^1(x_1)$ przy przejściu $(4) \rightarrow 11$ jest hazardem pozornym,
 $HD^1(x_1)$ przy przejściu $11 \rightarrow (4)$ jest hazardem nieważnym,
 $HD^1(x_2)$ przy przejściu $(1) \rightarrow 7$ jest hazardem pozornym,
 $HD^1(x_2)$ przy przejściu $7 \rightarrow 1$ jest hazardem nieważnym,
 $HD^1(x_2)$ przy przejściu $(5) \rightarrow 9$ jest hazardem pozornym,
 $HD^1(x_2)$ przy przejściu $9 \rightarrow (5)$ jest hazardem nieważnym,
 $HD^1(x_3)$ przy przejściu $1 \rightleftharpoons 9$ jest hazardem nieważnym,
 $HD^1(x_3)$ przy przejściu $(9) \rightarrow 1$ jest hazardem pozornym,
 $HD^1(x_3)$ przy przejściu $1 \rightarrow (9)$ jest hazardem nieważnym,
 $HD^1(y_2)$ przy przejściu $(4) \rightleftharpoons 7$ jest hazardem nieważnym.

Należy więc wyeliminować następujące szkodliwe hazardy podstawowe:

- z układu przełączającego realizującego funkcję \bar{F}_{R1}
 $HS1^1(x_3)$ przy przejściu między stanami wejścia $(1,1,0,1,0) \rightleftharpoons (1,1,1,1,0)$
- z układu przełączającego realizującego funkcję \bar{F}_{S1}
 $HS1^1(x_2)$ przy przejściu między stanami wejścia $(1,0,0,0,0) \rightarrow (1,1,0,0,0)$
 $HS1^1(x_1)$ przy przejściu między stanami wejścia $(0,1,1,0,0) \rightarrow (1,1,1,0,0)$

Eliminacja szkodliwego HS1¹(x₃) w \bar{F}_{R1}

W celu spełnienia warunku Ia (p. 3.2) pokrywa się parę sąsiednich stanów wejścia (1,1,0,1,0), (1,1,1,1,0) funkcji \bar{F}_{R1} nowym strukturalnym implikantem PK₂₈ (Zob. przykład 3.2.1.)



Wyrażenie strukturalne układu przełączającego realizującego \bar{F}_{R1} pozbawionego szkodliwego hazardu podstawowego

$$S^B = \overline{\overline{x_3 \cdot x_2 \cdot \bar{x}_1 \cdot x_2 \cdot \bar{y}_2} \cdot \overline{x_1 \cdot x_3 \cdot \bar{x}_1 \cdot x_2 \cdot \bar{y}_2} \cdot \overline{x_3 \cdot \bar{y}_2 \cdot \bar{x}_1 \cdot x_2 \cdot \bar{y}_2} \cdot \overline{x_1 \cdot x_2 \cdot \bar{y}_2}}$$

Eliminacja szkodliwego hazardu podstawowego wymaga dodatkowego użycia jednego elementu NAND i czterech wejść na elementy logiczne.

Eliminacja szkodliwego HS1¹(x₂) w \bar{F}_{S1}

W celu spełnienia warunku Ia (p. 3.2) pokrywa się parę sąsiednich stanów wejścia (1,0,0,0,0), (1,1,0,0,0) funkcji \bar{F}_{S1} nowym strukturalnym implikantem PK₂₉ ($\bar{S}_{29} = x_1 \cdot \bar{x}_3$).

Eliminacja szkodliwego HS1¹(x₁) w \bar{F}_{S1}

W celu spełnienia warunku Ia (p. 3.2) pokrywa się parę sąsiednich stanów wejścia (0,1,1,0,0), (1,1,1,0,0) funkcji \bar{F}_{S1} nowym strukturalnym implikantem PK₃₀ ($\bar{S}_{30} = x_2 \cdot x_3$). Implikant strukturalny PK₄ jest w tej sytuacji zbędny.

		$x_1 x_2 x_3$							
		000	001	011	010	110	111	101	100
$y_1 y_2$	00	1	1	1		1	1		1
	01	1	1	1		1	1		1
	11	1	1	1		1	1		1
	10	1	1	1		1	1		1

\overline{F}_{S1} \overline{F}_{K30}

→ \overline{F}_{K29}

Usunięcie implikantu strukturalnego \overline{F}_{K4} spowodowało pojawienie się $HS^1(x_2)$, który:

- przy przejściu 12 — (11) jest hazardem pozornym,
- przy przejściu 12 — (11) jest hazardem nieważnym,
- przy przejściu (12) — 11 jest hazardem nieważnym,
- przy przejściu (12) — 11 jest hazardem pozornym,
- przy przejściu — 11 jest hazardem nieważnym.

W związku z powyższym $HS^1(x_2)$ można zignorować.

Wyrażenie strukturalne układu przełączającego realizującego \overline{F}_{S1} pozbawionego szkodliwego hazardu podstawowego

$$S^1 = \overline{y_2 \cdot \overline{x_2} \cdot \overline{x_2} \cdot \overline{x_1} \cdot \overline{x_3} \cdot x_1 \cdot x_2 \cdot x_2 \cdot x_3 \cdot x_1 \cdot \overline{x_3}}$$

Eliminacja szkodliwego hazardu podstawowego wymaga dodatkowego użycia jednego elementu NAND i trzech wejść na elementy logiczne (element logiczny realizujący $\overline{x_3}$ może być wykorzystany z układu realizującego funkcję \overline{F}_{R1}).

3.4. Wnioski

Przedstawiony sposób eliminacji $HP^k(x_i)$ pozwala na wyeliminowanie dowolnych k-krotnych hazardów podstawowych na wyjściach dowolnych elementów logicznych wielowarstwowego układu przełączającego zbudowanego na bazie dowolnej kombinacji elementów logicznych I, LUB, NAND, NOR i NIE. Sposób eliminacji $HP^k(x_i)$ umożliwia szczególnie eliminację wszystkich szkodliwych k-krotnych hazardów podstawowych, a w związku z tym pozwala na budowę

tańszych, poprawnie działających wielowarstwowych układów przełączających.

Do eliminacji $HP^k(x_1)$ w wielowarstwowych układach przełączających wykorzystuje się opisany w rozdziale 2 algorytm I^k oraz przedstawione w punkcie 3.2. rozdziału 3 warunki jakie powinny spełniać wszystkie strukturalne implikanty PK_0^m i strukturalne implikanty QK_0^m bezhazardowego wielowarstwowego układu przełączającego. Do eliminacji w asynchronicznych układach sekwencyjnych szkodliwych $HP^k(x_1)$ wykorzystuje się dodatkowo przedstawione w pracy [28] zasady identyfikacji szkodliwych hazardów podstawowych.

W stosunku do istniejących już metod eliminacji hazardu podstawowego drogą deminimalizacji wyrażenia strukturalnego podanych w pracach [3], [5], [18], [31], [42], [48], [58], [64], [69] przedstawiony sposób pozwala eliminować dodatkowo na wyjściu każdego elementu logicznego w dowolnym wielowarstwowym układzie przełączającym:

- k-krotny hazard statyczny ze względu na zmienną x_1 w warunkach działania,
- k-krotny hazard statyczny ze względu na zmienną x_1 w warunkach niedziałania
- k-krotny hazard dynamiczny ze względu na zmienną x_1 .

Przytoczone przykłady wykazały pełną przydatność opisanego sposobu do eliminacji dowolnego k-krotnego hazardu podstawowego.

Przykład 3.3.2 wyraźnie ilustruje możliwość budowania tańszych układów sekwencyjnych. Przedstawiony w nim układ przełączający realizujący funkcję \bar{F}_{R1} z wyeliminowanymi hazardami szkodliwymi jest tańszy o cztery elementy NAND i dziesięć wejść na elementy logiczne od całkowicie bezhazardowego układu przełączającego realizującego funkcję \bar{F}_{R1} (przykład 3.2.1). Z przytoczonego przykładu 3.2.1 nie można jednak wyciągnąć narzucającego się optymistycznego wniosku:

- każda eliminacja wszystkich $HP^k(x_1)$ w minimalnym wielowarstwowym układzie przełączającym realizującym funkcję F pozwala w efekcie zawsze uzyskać najtańszy spośród wszystkich bezhazardowych układów przełączających realizujących funkcję F

Poniższy przykład nie pozwala takiego wniosku wyciągnąć

a) minimalny trójwarstwowy układ przełączający zbudowany z elementów NAND

		x_1x_2			
		00	01	11	10
x_3x_4	00		1	1	1
	01				1
	11	1			
	10	1	1		1

$$S^1 = \overline{x_3 \cdot \overline{x_2} \cdot \overline{x_4} \cdot \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_1} \cdot \overline{x_4} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_2} \cdot \overline{x_4} \cdot x_1 \cdot \overline{x_3} \cdot \overline{x_1} \cdot \overline{x_2}}$$

b) bezhazardowy trójwarstwowy układ przełączający realizujący funkcję F^1 uzyskany w efekcie eliminacji w układzie a) wszystkich $HP^k(x_1)$

		x_1x_2			
		00	01	11	10
x_3x_4	00		1	1	1
	01				1
	11	1			
	10	1	1		1

14 elementów NAND i 33 wejścia

$$S^1 = \overline{x_3 \cdot \overline{x_2} \cdot \overline{x_4} \cdot \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_1} \cdot \overline{x_4} \cdot x_2 \cdot \overline{x_3} \cdot \overline{x_4} \cdot x_1 \cdot \overline{x_3} \cdot \overline{x_2} \cdot x_1 \cdot \overline{x_4} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_1} \cdot x_2 \cdot \overline{x_4}}$$

c) inny bezhazardowy trójwarstwowy układ przełączający realizujący funkcję F^1 jest tańszy od układu b) o 2 elementy NAND i 5 wejść

		$x_1 x_2$			
		00	01	11	10
$x_3 x_4$	00		1	1	1
	01				1
	11	1	1		
	10	1	1		1

12 elementów NAND i 28 wejścia

$$S^1 = \overline{x_1 \cdot \bar{x}_3 \cdot \bar{x}_2 \cdot \bar{x}_4 \cdot x_1 \cdot \bar{x}_2 \cdot \bar{x}_4 \cdot x_3 \cdot \bar{x}_1 \cdot x_2 \cdot \bar{x}_4 \cdot x_3 \cdot \bar{x}_2 \cdot \bar{x}_4 \cdot x_2 \cdot \bar{x}_4 \cdot x_1 \cdot \bar{x}_3}$$

Przedstawiony sposób eliminacji $HP^k(x_1)$ nie daje więc pewności uzyskania zawsze najtańszego układu spośród wszystkich bezhazardowych wielowarstwowych układów przełączających realizujących daną funkcję.

Problem ten rozwiązano dla układów TANT i w zakresie $HP^k(x_1)$ w rozdziale 4, gdzie przedstawiono metodę syntezy najtańszych bezhazardowych trójwarstwowych układów przełączających zbudowanych z elementów NAND.

4. METODY SYNTEZY BEZHARZARDOWYCH UKŁADÓW TANT REALIZUJĄCYCH FUNKCJE n-ZMIENNYCH

4.1. Uwagi wstępne

Projektant układów przełączających powinien dysponować 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 daną funkcję Boole'a. Metody syntezy oprócz wyżej wspomnianych wymagań powinny między innymi spełniać jeszcze następujące [72]:

- możliwość syntezy układów przełączających realizujących funkcje n-zmiennych,

- możliwość wyboru kryterium optymalności układu np. najmniej-
szy czas propagacji układu,
- przydatność dla szerokiej klasy elementów,
- uwzględnienie ograniczeń właściwych realnym elementom i ogra-
niczeń związanych z ich montażem w realnym układzie,
- możliwość uwzględniania dodatkowych wymagań nakładanych na
projektowany układ przełączający np. możliwość łatwego napra-
wiania,
- zapewnienie otrzymania układów optymalnych lub bliskich opty-
malnym.

Opracowanie metody syntezy spełniającej te i poprzednie wy-
magania nie jest rzeczą łatwą, tym bardziej, że spełnienie jed-
nych wymagań wyklucza możliwość spełnienia innych. Na przykład
dążenie do uzyskania jak najmniejszych czasów propagacji ukła-
dów przełączających zmusza do zmniejszenia liczby warstw ukła-
du przełączającego do niezbędnego minimum pozwalającego jesz-
cze realizować dowolne funkcje n -zmiennych. Staje się to auto-
matycznie ograniczeniem dla kryterium kosztów, gdyż liczba moż-
liwych do porównania układów przełączających zostaje w ten spo-
sób ograniczona tylko do układów o niezbędnej liczbie warstw.
Jeśli jednak ograniczy się zastosowanie metody syntezy kombina-
cyjnych układów przełączających do funkcjonalnie pełnego syste-
mu w postaci elementu NAND [72], to sprawa się upraszcza i bar-
dziej realne staje się stworzenie metody spełniającej większość
wymienionych wymagań.

Gimpel w swojej pracy [11] przedstawił prostą metodę synte-
zy minimalnych, trójwarstwowych kombinacyjnych układów przełą-
czających zbudowanych z elementów NAND i zasilanych wyłącznie
zmiennymi niezanegowanymi. Układy takie nazwane są w tej pracy
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 zbiorze dopuszczalnych niższych implikantów. Ponieważ

korzystanie z tego zbioru powoduje jednak, że synteza absolutnie minimalnych układów TANT jest bardzo uciążliwa, Gimpel równolegle udowodnił, że przeprowadzenie syntezy układów TANT na zbiorze głównych dopuszczalnych implikantów^{*)} jest o wiele mniej pracochłonne, a jednocześnie pozwala uzyskać układy TANT o minimalnej liczbie elementów i liczbie wszystkich wejść na elementy niewiele większej od liczb uzyskiwanych w pierwszym przypadku. Metoda Gimpela nie uwzględnia jednak bardzo istotnego dla asynchronicznych układów sekwencyjnych problemu hazardu podstawowego.

Dla dwuwarstwowych kombinacyjnych układów przełączających LUB-I oraz I-LUB metodę bezhazardowej ich syntezy przedstawił bazując na pracy Huffmana [18], McCluskey. Pewną metodę syntezy trójwarstwowych układów przełączających zbudowanych z elementów NOR i nie zawierających hazardu statycznego w warunkach działania przedstawiono w pracy [10].

Natomiast metodę bezhazardowej syntezy trójwarstwowych układów przełączających zbudowanych z elementów NOR oraz układów TANT przedstawiono w pracach [16] i [17]. Na podstawie tych prac i na bazie metody Gimpela w rozdziale niniejszym przedstawiono metodę syntezy minimalnych bezhazardowych układów TANT realizujących funkcje n-zmiennych.

4.2. Algorytm G^X i syntezy bezhazardowych układów TANT

4.2.1. Podstawowe definicje i twierdzenia

Ogólną postać pokazanego na rys. 18 układu TANT przedstawia wyrażenie strukturalne

$$S = \bigwedge_{o=2}^r \bar{E}_o$$

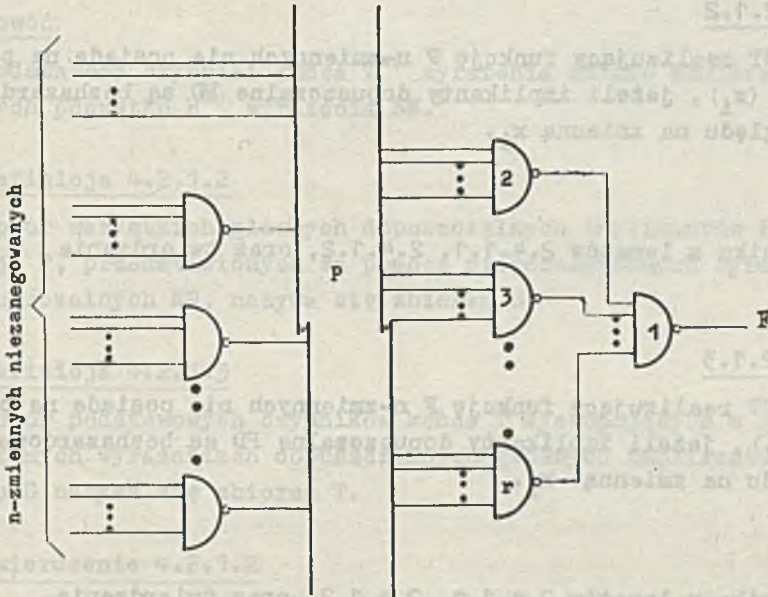
lub tzw. wyrażenie TANT

$$S = \bigvee_{o=2}^r E_o$$

*) Zob. Przypisy do rozdziału 4

Funkcję F realizowaną przez ten układ TANT przedstawia poniższa suma dopuszczalnych^{*)} implikantów PD_0 .

$$F = \bigvee_{\alpha=2}^r PD_\alpha$$



Rys. 18. Ogólna postać układu TANT realizującego funkcje n -zmiennych (P przedstawia pewien dowolny zbiór połączeń wewnętrznych)

Definicja 4.2.1.1

Bezszkodowym układem TANT nazywa się każdy układ TANT, który nie posiada $HP^k(x_i)$ na wyjściu któregośkolwiek elementu NAND.

Lemat 4.2.1.1

Układ TANT realizujący funkcję F n -zmiennych nie posiada na pewno $HS1^k(x_i)$, jeżeli każda para sąsiednich stanów wejścia A i A' , dla których stan wyjścia $f(A) = f(A') = 1$, jest pokryta przynajmniej przez jeden bezszkodowy ze względu na zmienną x_i implikant dopuszczalny PD .

*) Zob. Przypisy do rozdziału 4

Dowód:

Lemat wynika z lematów 2.4.1.1, 2.4.1.2, oraz twierdzenia 2.4.1.1.

Lemat 4.2.1.2

Układ TANT realizujący funkcję F n -zmiennych nie posiada na pewno $HSO^k(x_1)$, jeżeli implikanty dopuszczalne PD są bezhazardowe ze względu na zmienną x_1 .

Dowód:

Lemat wynika z lematów 2.4.1.1, 2.4.1.2, oraz twierdzenia 2.4.1.2.

Lemat 4.2.1.3

Układ TANT realizujący funkcję F n -zmiennych nie posiada na pewno $HD^k(x_1)$, jeżeli implikanty dopuszczalne PD są bezhazardowe ze względu na zmienną x_1 .

Dowód:

Lemat wynika z lematów 2.4.1.1, 2.4.1.2, oraz twierdzenia 2.4.1.3.

Twierdzenie 4.2.1.1

Układ TANT realizujący funkcję F n -zmiennych nie posiada na pewno $HP^k(x_1)$, jeżeli każda para sąsiednich stanów wejścia A i A' , dla których stan wyjścia $f(A) = f(A') = 1$, jest pokryta przynajmniej przez jeden bezhazardowy ze względu na zmienną x_1 implikant dopuszczalny PD.

Dowód:

Twierdzenie wynika z lematów 2.4.1.1, 2.4.1.2 i 2.4.1.3.

Lemat 4.2.1.4

W każdym wyrażeniu TANT bezhazardowego układu TANT występują tylko nierozszerzone wyrażenia dopuszczalne EN^{*)}.

Dowód:

Podstawowe czynniki końca $\bar{T}^{(*)}$ wyrażenia EN nie zawierają zmiennych początku $H^{(*)}$ wyrażenia EN.

Definicja 4.2.1.2

Zbiór wszystkich głównych dopuszczalnych implikantów $PG^{(*)}$ funkcji F, przedstawionych za pomocą nierozszerzonych wyrażen dopuszczalnych EN, nazywa się zbiorem G.

Definicja 4.2.1.3

Zbiór podstawowych czynników końca \bar{T} występujących w nierozszerzonych wyrażeniach dopuszczalnych głównych implikantów ze zbioru G nazywa się zbiorem T.

Twierdzenie 4.2.1.2

Wśród bezhazardowych układów TANT realizujących daną funkcję F n-zmiennych istnieje na pewno taki minimalny, ze względu na kryterium kosztów elementów logicznych, bezhazardowy układ TANT, że bezhazardowe implikanty PD są głównymi implikantami ze zbioru G.

Dowód:

Twierdzenie wynika z lematu 4.2.1.4 oraz z twierdzenia Gimpe-la^{*)}.

Definicja 4.2.1.4

Minimalnym zbiorem G_{\min} funkcji F nazywa się podzbiór głównych implikantów ze zbioru G, stanowiących pokrycie wszystkich par sąsiednich stanów wejścia A i A', dla których $f(A) = f(A') = 1$

^{*)} Zob. Przypisy do rozdz. 4

i który jest minimalny ze względu na kryterium kosztów elementów logicznych oraz minimalny w zbiorze G ze względu na kryterium kosztów wejść na elementy logiczne.

Definicja 4.2.1.5

Minimalnym zbiorem T_{\min} nazywa się zbiór podstawowych czynników końca występujących w nierozszerzonych wyrażeniach dopuszczalnych głównych implikantów ze zbioru G_{\min} .

Definicja 4.2.1.6

Zbiór (G_{\min}, T_{\min}) nazywa się bazą realizacji minimalnego wyrażenia TANT bezhazardowego układu TANT realizującego daną funkcję F n -zmiennych.

4.2.2. Określenie bazy realizacji (G_{\min}, T_{\min}) za pomocą redukcji rozwiniętej tablicy AA'

Gimpel poszukuje minimalnego zbioru głównych implikantów pokrywających każdy stan wejścia funkcji F za pomocą zmodyfikowanej przez siebie tablicy Luccio-Grassellego, zwanej popularnie tablicą CC^{**}) i wykorzystując do jej redukcji twierdzenia Luccio-Grassellego oraz twierdzenia własne.

Znalezienie pokrycia każdej pary sąsiednich stanów wejścia A i A' jest możliwe po zmodyfikowaniu tablicy głównych implikantów Gimpela^{**}) wg zasad przedstawionych przez McCluskey'a w pracy [31]. W pracy [48] przedstawiono inny oryginalny sposób bezhazardowej syntezy układów przełączających (stykowych) za pomocą podwójnego pokrywania każdej kolumny tablicy prostych implikantów. Jednakże ze względu na popularność w literaturze światowej metody McCluskey'a zdecydowano się skorzystać z zasad przez niego podanych. Zmodyfikowana według tych zasad tablica głównych implikantów^{**}) ma wiersze odpowiadające wszystkim pa-

ⁿ) Opis tej tablicy można znaleźć w pracy [11]

ⁿⁿ) Opis tej tablicy można znaleźć w pracach [16] i [17]

rom sąsiednich stanów wejścia, dla których $f(A) = f(A') = 1$. Tablicę tę, zwaną dalej tablicą AA' , przedstawiono w przykładzie 4.2.4.1. Przyjęcie do rozwiązania tablicy AA' zasad Gimpela, poszukiwania za pomocą tablicy CC minimalnego wyrażenia TANT, powoduje, że synteza minimalnych bezhazardowych układów TANT jest pracochłonna. Jeżeli ponadto nie wprowadzi się do algorytmu rozwiązywania tablicy CC pewnych korekt, to w wielu przypadkach otrzymuje się rozwiązania zawierające HSO lub HD. Celem uniknięcia tych niedogodności proponuje się poszukiwanie minimalnych bezhazardowych układów TANT prowadzić za pomocą specjalnych technik redukcji rozwiniętej tablicy AA' , przedstawionych w tym rozdziale.

4.2.2.1. Rozwinięta tablica AA'

Rozwinięta tablica AA' składa się z tablicy AA' z kolumnami pokryć par sąsiednich stanów wejścia A i A' (kolumny AA') oraz dodatkowych kolumn zwanych kolumnami podstawowych czynników końca \bar{T} (kolumny \bar{T}), których liczba równa jest liczbie różnych podstawowych czynników końca \bar{T} występujących w zbiorze T . W wierszach rozwiniętej tablicy AA' umieszczone są wszystkie główne implikanty PG ze zbioru G .

Jeżeli jakiś główny implikant PG pokrywa parę sąsiednich stanów wejścia, to w miejscu przecięcia wiersza tego głównego implikantu i kolumny odpowiadającej pokrywanej parze sąsiednich stanów wejścia umieszcza się znaczek "x" i mówi się, że wiersz głównego implikantu PG pokrywa tę kolumnę.

Jeżeli nierozszerzone dopuszczalne wyrażenie EN głównego dopuszczalnego implikantu PG posiada podstawowy czynnik końca \bar{T} , to w miejscu przecięcia wiersza tego głównego implikantu i kolumny \bar{T} umieszcza się znaczek "o" i mówi się, że kolumna \bar{T} realizuje wiersz głównego implikantu PG .

Koszt C_{TANT} realizacji funkcji F za pomocą układu TANT można określić jako sumę kosztów wnoszonych przez poszczególne elementy logiczne układu TANT (rys. 18). Koszty te wg [19] są

funkcją liczby elementów logicznych λ oraz liczby wejść na elementy logiczne μ . W związku z tym koszty poszczególnych fragmentów układu TANT można przedstawić za pomocą następujących zależności:

$$C_{T_j} = \varphi(\lambda_{\bar{T}_j}, \mu_{\bar{T}_j})$$

gdzie: j określa j -ty element trzeciej warstwy,

$\lambda_{\bar{T}_j} = 1$ natomiast $\mu_{\bar{T}_j}$ określa liczbę zmiennych niezanegowanych występujących w podstawowym czynniku końca \bar{T}_j ,

np. dla $\bar{T}_j = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3$ $\lambda_{\bar{T}_j} = 1, \mu_{\bar{T}_j} = 3$

$C_{B_0} = \varphi(\lambda_{B_0}, \mu_{B_0})$, gdzie:

o określa o -ty element drugiej warstwy,

$\lambda_{B_0} = 1$ natomiast μ_{B_0} określa sumę liczby zmiennych występujących w początku H_0 wyrażenia EN_0 i liczby podstawowych czynników końca \bar{T} występujących w wyrażeniu EN_0 ,

np. dla $EN_0 = x_1 \cdot x_2 \cdot \bar{x}_3 \cdot \bar{x}_4 \cdot \bar{x}_5$ $\lambda_{B_0} = 1, \mu_{B_0} = 4$

$C_{B_1} = \varphi(\lambda_{B_1}, \mu_{B_1})$ gdzie:

1 określa element wyjściowy układu TANT,

$\lambda_{B_1} = 1$ natomiast μ_{B_1} określa liczbę wyrażeń EN_0 występujących w wyrażeniu $TANT_1$,

np. dla $EN_1 + EN_2 + EN_3$ $\lambda_{B_1} = 1, \mu_{B_1} = 3$

Upraszczając rozumowanie przyjęto za [19], że do porównania kosztów dwóch układów przełączających, w tym przypadku kosztów C_{TANT} dwóch układów TANT realizujących tę samą funkcję F , wystarczy porównać liczbę elementów logicznych oraz liczbę wejść na elementy logiczne tych układów przełączających. Jeżeli z prawej strony każdego wiersza rozwiniętej tablicy AA' umieści się liczbę określającą μ_{B_0} oraz nad każdą kolumną \bar{T} umieści się liczbę określającą $\mu_{\bar{T}}$ i wiedząc, że każdemu wierszowi niezbędnemu do pokrycia kolumn AA' oraz każdej niezbędnej do realizacji wiersza kolumnie \bar{T} odpowiada $\lambda_{B_0} = \lambda_{\bar{T}} = 1$, ponadto

znając liczbę wierszy pokrywających wszystkie kolumny AA' można zawsze określić sumaryczną liczbę elementów NAND oraz sumaryczną liczbę wejść na elementy NAND niezbędną do realizacji funkcji F . Znając przedstawione wyżej wartości można również porównywać koszty wnoszone przez poszczególne wiersze. Ułatwia to redukcję rozwiniętej tablicy AA' i ułatwia wyznaczanie wierszy niezbędnych do minimalnego pokrycia kolumn AA' jak również kolumn \bar{T} niezbędnych do realizacji poszczególnych wierszy.

Definicja 4.2.2.1.1

Pokryciem U rozwiniętej tablicy AA' posiadającej r wierszy nazywa się taki zbiór wierszy $[u_1, u_2, \dots, u_h]$, gdzie $h \leq r$, że każda kolumna AA' tej tablicy jest pokryta przynajmniej przez jeden wiersz u tego zbioru.

Definicja 4.2.2.1.2

Kosztom pokrycia C_U nazywa się sumę kosztów C_{B_0} wnoszonych przez h wierszy pokrycia U i kosztów $C_{\bar{T}_j}$ wnoszonych przez g kolumn \bar{T} realizujących wiersze pokrycia U .

$$C_U = \sum_{o=1}^h C_{B_o} + \sum_{j=1}^g C_{\bar{T}_j}$$

Definicja 4.2.2.1.3

Rozwiązaniem rozwiniętej tablicy AA' nazywa się pokrycie o najmniejszym koszcie C_U (pokrycie minimalne).

Wybór minimalnego pokrycia rozwiniętej tablicy AA' jest bardzo często kłopotliwy i w związku z tym poprzedza się go pewnymi technikami redukcyjnymi, które doprowadzają pierwotną rozwiniętą tablicę AA' do postaci uproszczonej, gdzie wybór minimalnego pokrycia nie sprawia większego kłopotu.

4.2.2.2. Technika redukcji rozwiniętej tablicy AA'

A. Dominacja wierszy

Oznaczmy zbiory kolumn AA' pokryte przez wiersze u_1 i u_2 odpowiednio jako M_1 i M_2 , natomiast zbiory kolumn \bar{T} realizujących wiersze u_1 i u_2 oznaczmy odpowiednio jako N_1 i N_2 .

Definicja 4.2.2.2.1

Wiersz u_1 rozwiniętej tablicy AA' dominuje nad wierszem u_2 tej samej tablicy ($u_1 \rightarrow u_2$), jeżeli:

- 1) $M_1 \supseteq M_2$ i $N_1 \subsetneq N_2$ lub $N_1 = 0$, albo
- 2) $M_1 \supseteq M_2$ i $N_1 = N_2$ lub $N_1 = 0, N_2 = 0$

Definicja 4.2.2.2.2

Wiersz u_1 rozwiniętej tablicy AA' równy jest wierszowi u_2 tej samej tablicy ($u_1 = u_2$), jeżeli:

$$M_1 = M_2 \text{ i } N_1 = N_2 \quad \text{lub} \quad N_1 = 0, N_2 = 0$$

Twierdzenie 4.2.2.2.1

Jeżeli wiersz u_2 rozwiniętej tablicy AA' jest dominowany przez wiersz u_1 tej samej tablicy lub jest równy wierszowi u_1 , to można usunąć z rozwiniętej tablicy AA' wiersz u_2 oraz każdą kolumnę \bar{T} realizującą tylko wiersz u_2 .

Dowód:

Jeżeli $N_1 \subseteq N_2$ lub $N_1 = 0$, to koszt wnoszony przez wiersz u_1 jest mniejszy lub najwyżej równy kosztowi wnoszonemu przez wiersz u_2 i jeżeli dodatkowo wiersz u_2 jest pokryty przez wiersz u_1 lub te pokrycia są sobie równe, to usunięcie wiersza u_2 i kolumn \bar{T} realizujących tylko wiersz u_2 pozostanie bez wpływu na ostateczny wynik minimalizacji.

B. Krytyczne czynniki końca

Gimpel w swojej pracy [11] krytycznym czynnikiem końca funkcji Boole'a F nazywa każdą zanegowaną zmienną, która występuje w postaci podstawowego czynnika końca w maksymalnym dopuszczalnym implikancie^{*)} o początku 1. Jednocześnie udowodnił, że każde wyrażenie TANT funkcji F zawiera wszystkie krytyczne czynniki końca funkcji F .

^{*)}Zob. Przypisy do rozdz. 4

Bez szkody dla końcowego wyniku można nie umieszczać w rozwiniętej tablicy AA' kolumn T z krytycznymi czynnikami końca funkcji F.

C. Zastępowalne czynniki końca

Twierdzenie Gimpela [11] o zastępowalności brzmi następująco: "Czynnik końca \bar{x}_1 jest zastępowalny w funkcji Boole'a F przez czynnik końca $\frac{x_1 \cdot x_2 \cdot \dots \cdot x_g}{x_1}$, gdzie $g \in [2, \dots, n]$, jeżeli dla każdego $i \in [2, \dots, g]$ spełniona jest następująca zależność

$$\bar{x}_i \cdot \left(\frac{F}{\bar{x}_1} \right) \equiv F$$

Zastosowanie tego twierdzenia do redukcji rozwiniętej tablicy AA' może spowodować w niektórych przypadkach, że układ TANT, uzyskany w efekcie rozwiązania rozwiniętej tablicy AA', będzie posiadał HSO lub HD. W związku z tym przy syntezie minimalnych bezhazardowych układów TANT zasięg zacytowanego wyżej twierdzenia musi mieć dodatkowe ograniczenie i w nowej formie brzmieć będzie następująco:

Twierdzenie 4.2.2.2.2

Podstawowy czynnik końca \bar{x}_1 dowolnego dopuszczalnego nierozszerzonego wyrażenia EN głównego implikantu PG jest zastępowalny w funkcji Boole'a F przez inny podstawowy czynnik końca $\frac{x_1 x_2 \dots x_g}{x_1}$, gdzie $g \in [2, \dots, n]$, jeżeli dla każdego $i \in [2, \dots, g]$ spełniona jest zależność

$$\bar{x}_i \cdot \left(\frac{F}{\bar{x}_1} \right) \equiv F$$

oraz zmienna x_1 nie występuje w początku H wyrażenia EN, $(x_1 \cdot x_2 \dots x_g \text{ — } x_1)$.

Dowód:

Ograniczenie twierdzenia wynika z lematu 4.2.1.4

Jeżeli twierdzenie 4.2.2.2.2. jest spełnione w zbiorze G, to można przeprowadzić następującą redukcję rozwiniętej tablicy

AA' . Usuwa się kolumnę \bar{T} z zastępowalnym czynnikiem końca \bar{x}_1 , natomiast wszystkie znaczki "o" w wierszach realizowanych przez tę kolumnę przenosi się w tych samych wierszach do kolumny \bar{T} z podstawowym czynnikiem końca $\bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_g$. Pozwala to zachować prawdziwą informację o kosztach wnoszonych przez te wiersze.

D. Dominacja kolumn AA'

Oznaczmy przez U_1 i U_2 zbiory wierszy pokrywających dwie różne kolumny pokryć c_1 i c_2 .

Definicja 4.2.2.2.3

Kolumna pokryć c_1 rozwiniętej tablicy AA' dominuje nad kolumną pokryć c_2 tej samej tablicy ($c_1 \rightarrow c_2$) jeżeli $U_1 \supseteq U_2$.

Definicja 4.2.2.2.4

Kolumna pokryć c_1 rozwiniętej tablicy AA' jest równa kolumnie pokryć c_2 tej samej tablicy ($c_1 = c_2$) jeżeli $U_1 = U_2$. McCluskey udowodnił w [31], że jeżeli kolumna pokryć tablicy prostych implikantów dominuje nad inną kolumną pokryć tej samej tablicy lub jest jej równa, to można kolumnę dominującą usunąć z tej tablicy. Twierdzenie to odnosi się również do rozwiniętej tablicy AA' i nie ma żadnego ograniczenia jego stosowalności przy redukcji tej tablicy.

E. Wiersze niezbędne

Definicja 4.2.2.2.5

Wierszem niezbędnym nazywa się wiersz, który jako jedyny pokrywa kolumnę AA' .

Twierdzenie 4.2.2.2.3

Wiersz niezbędny oraz kolumny AA' przez ten wiersz pokryte, kolumny \bar{T} , które go realizują, jak również wszystkie wiersze pokryte przez wiersz niezbędny wraz z ich kolumnami \bar{T} , które tylko te wiersze realizują można usunąć z rozwiniętej tablicy AA' .

Dowód:

Twierdzenie jest oczywiste.

Twierdzenie 4.2.2.2.4

Każdy zbiór (G_{\min}, T_{\min}) zawiera główny implikant PG niezbędny wiersza oraz podstawowe czynniki końca kolumn \bar{T} , które ten wiersz realizują.

Dowód:

Twierdzenie jest oczywiste.

Twierdzenie 4.2.2.2.3 można wykorzystać do redukcji rozwiniętej tablicy AA' tylko po usunięciu wszystkich kolumn \bar{T} z zastępowalnymi czynnikami końca. Ograniczenie to jest konieczne z następującego powodu. Jeżeli którakolwiek kolumna \bar{T} z zastępowalnym czynnikiem końca realizuje tylko wiersz niezbędny, to zgodnie z twierdzeniem 4.2.2.2.4 ten zastępowalny podstawowy czynnik końca znajdzie się w zbiorze (G_{\min}, T_{\min}) . W związku z tym koszt bezhazardowego układu TANT będzie droższy o jeden element NAND.

Każdy główny implikant PG oraz każdy podstawowy czynnik końca \bar{T} należący do zbioru (G_{\min}, T_{\min}) oznaczać się będzie dalej w następujący sposób:

- główny implikant - PG•
- podstawowy czynnik końca - \bar{T} •

4.2.3. Charakterystyka algorytmu G^{x_1}

1. Określenie maksymalnych dopuszczalnych implikantów funkcji p^{*k})
2. Określenie krytycznych czynników końca
3. Określenie zbioru G^{*k})
4. Budowa rozwiniętej tablicy AA'
5. Redukcja rozwiniętej tablicy AA'
6. Określenie zbioru (G_{\min}, T_{\min})

* Krok 1, 2 i 3 algorytmu G^{x_1} wykonuje się wg zasad podanych przez Gimpeła w pracy [11]. Dwa przykłady podano w przypisach.

7. Określenie wyrażenia TANT minimalnego bezhazardowego układu TANT

Schemat blokowy algorytmu G^{x_i} przedstawia rys. 19.

4.2.4. Przykłady

Przykład 4.2.4.1

Niech będzie zadana funkcja

$$F = \bar{x}_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 \bar{x}_4 + x_1 x_2 \bar{x}_3 + x_1 \bar{x}_3 x_4$$

Maksymalne implikanty dopuszczalne funkcji F

$$\bar{x}_1 x_3 \bar{x}_2 \bar{x}_4, \bar{x}_1 \bar{x}_2 x_3 x_4, x_2 \bar{x}_4 \bar{x}_1 \bar{x}_3, \bar{x}_1 x_2 x_3 \bar{x}_4, x_1 \bar{x}_3 x_4, x_1 x_2 \bar{x}_3, x_1 x_2 \bar{x}_3 x_4$$

Krytyczne czynniki końca funkcji F

Funkcja F nie posiada krytycznych czynników końca

Zbiór G funkcji F

$$PG_1 = x_1 \cdot x_2 \cdot \bar{x}_3$$

$$PG_2 = x_1 \cdot \bar{x}_3 \cdot x_4$$

$$PG_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3$$

$$PG_4 = \bar{x}_1 \cdot x_3 \cdot \bar{x}_4$$

$$PG_5 = \bar{x}_1 \cdot x_2 \cdot \bar{x}_4$$

$$PG_6 = x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$$

$$PG_7 = x_3 \cdot \bar{x}_1 \cdot x_2 \cdot \bar{x}_4$$

$$PG_8 = x_2 \cdot \bar{x}_4 \cdot x_1 \cdot x_3$$

Budowa rozwiniętej tablicy AA'

Tabela 11 przedstawia rozwiniętą tablicę AA'.

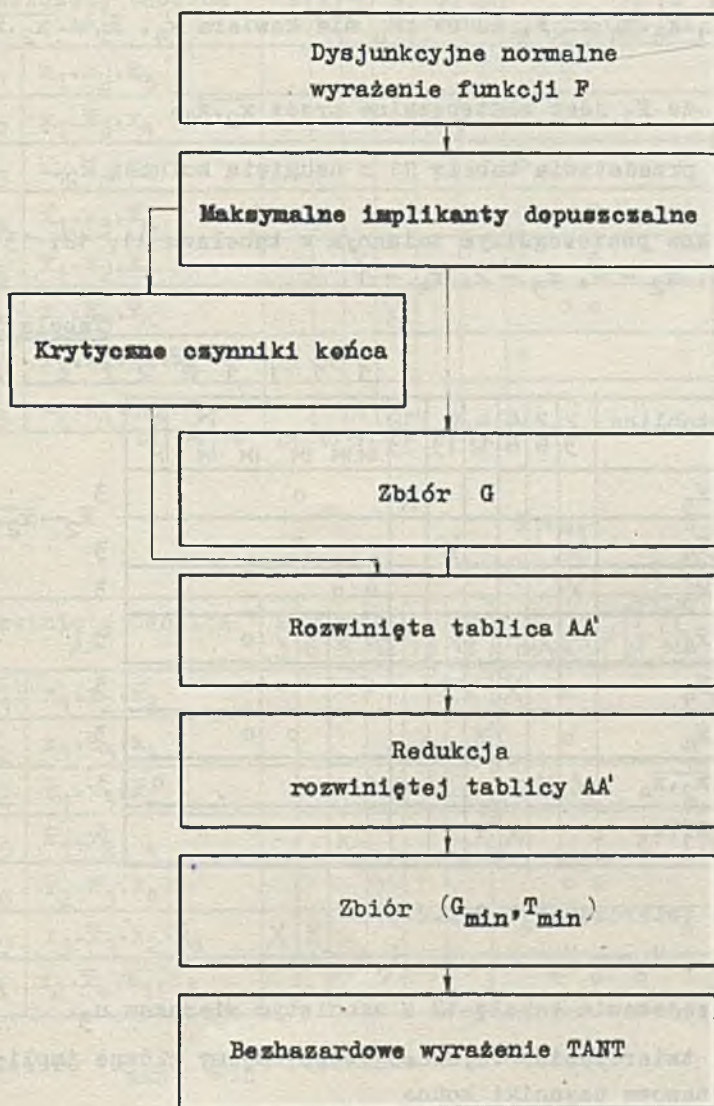
Redukcja rozwiniętej tablicy AA'

Ponieważ z twierdzenia 4.2.2.2.1 nie można skorzystać, należy określić zastępowalne podstawowe czynniki końca.

Stosownie do twierdzenia 4.2.2.2.2

$$\bar{x}_1 \cdot \left(\frac{F}{\bar{x}_3} \right) = \bar{x}_1 \cdot x_2 \cdot \bar{x}_4 \Leftarrow F, EN_1 \text{ i } EN_2 \text{ zawierają } x_1, \bar{x}_3 \not\leftarrow \bar{x}_1 \cdot \bar{x}_3,$$

$$\bar{x}_2 \cdot \left(\frac{F}{\bar{x}_4} \right) = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 \Leftarrow F, EN_5 \text{ i } EN_6 \text{ zawierają } x_2, \bar{x}_4 \not\leftarrow \bar{x}_2 \cdot \bar{x}_4,$$



Rys. 19. Schemat blokowy algorytmu G^{x_1}

$$\bar{x}_3 \cdot \left(\frac{F}{\bar{x}_1} \right) = x_2 \cdot \bar{x}_3 \cdot \bar{x}_4 \subset F, EN_3 \text{ i } EN_4, EN_7 \text{ zawierają } x_3, \bar{x}_1 \cdot \bar{x}_1 \cdot \bar{x}_3,$$

$$\bar{x}_4 \cdot \left(\frac{F}{\bar{x}_2} \right) = \bar{x}_1 \cdot x_3 \cdot \bar{x}_4 \subset F, \text{ żadne } EN_0 \text{ nie zawiera } x_4, \bar{x}_2 \cdot \bar{x}_2 \cdot \bar{x}_4.$$

Stąd wynika, że \bar{x}_2 jest zastępowalne przez $\bar{x}_2 \cdot \bar{x}_4$.

Tabela 12 przedstawia tabelę 11 z usuniętą kolumną \bar{x}_2 .

Wagi przypisane poszczególnym zmiennym w tabelach 11, 12, 13 i 14: $x_1 - 8, x_2 - 4, x_3 - 2, x_4 - 1$

Tabela 11

Rozwinięta tablica AA		2	2	4	4	9	12	12	12	12	12	12	12	12	12	12	
		3	6	6	12	13	13	\bar{x}_1	\bar{x}_2	\bar{x}_3	\bar{x}_4	$\bar{x}_1 \cdot \bar{x}_3$	$\bar{x}_2 \cdot \bar{x}_4$	$\bar{x}_1 \cdot \bar{x}_4$	$\bar{x}_2 \cdot \bar{x}_3$	$\bar{x}_3 \cdot \bar{x}_4$	
PG ₁	$x_1 \cdot x_2 \cdot \bar{x}_3$						X			o							3
PG ₂	$x_1 \cdot \bar{x}_3 \cdot x_4$					X				o							3
PG ₃	$\bar{x}_1 \cdot \bar{x}_2 \cdot x_3$	X						o	o								3
PG ₄	$\bar{x}_1 \cdot x_3 \cdot \bar{x}_4$		X					o		o							3
PG ₅	$\bar{x}_1 \cdot x_2 \cdot \bar{x}_4$			X				o		o							3
PG ₆	$x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$				X					o	o						3
PG ₇	$x_3 \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$	X	X					o							o		3
PG ₈	$x_2 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_3$			X	X						o	o					3

Stosownie do twierdzenia 4.2.2.2.1

$$\frac{u}{7} \text{ — } \frac{u}{3}$$

Tabela 13 przedstawia tabelę 12 z usuniętym wierszem u_3 .

Stosownie do twierdzenia 4.2.2.2.3 otrzymujemy główne implikanty i podstawowe czynniki końca

$$PG_1, PG_2, PG_7, \bar{x}_1, \bar{x}_3, \bar{x}_2 \cdot \bar{x}_4$$

Tabela 14 jest zredukowaną rozwiniętą tablicą AA'. Rozwiązaniem tej tabeli jest wiersz u_8 .

Tabela 12

1 1 1 2 2

Rozwinięta tablica AA		2 3	2 6	4 6	4 12	9 13	12 13	\bar{x}_1	\bar{x}_3	\bar{x}_4	$\bar{x}_1 \cdot \bar{x}_3$	$\bar{x}_2 \cdot \bar{x}_4$
PG ₁	$x_1 \cdot x_2 \cdot \bar{x}_3$						X	o				
PG ₂	$x_1 \cdot \bar{x}_3 \cdot x_4$					X		o				
PG ₃	$\bar{x}_1 \cdot \bar{x}_2 \cdot x_3$	X						o			o	
PG ₄	$\bar{x}_1 \cdot x_3 \cdot \bar{x}_4$		X					o	o			
PG ₅	$\bar{x}_1 \cdot x_2 \cdot \bar{x}_4$			X				o	o			
PG ₆	$x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$				X			o	o			
PG ₇	$x_3 \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$	X	X					o			o	
PG ₈	$x_2 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_3$			X	X				o	o		

$u_7 \rightarrow u_3$

Tabela 13

$\bar{x}_1 \cdot \bar{x}_3$ $\bar{x}_2 \cdot \bar{x}_4$
 (1) (1) 1 2 (2)

Rozwinięta tablica AA		2 3	2 6	4 6	4 12	9 13	12 13	\bar{x}_1	\bar{x}_3	\bar{x}_4	$\bar{x}_1 \cdot \bar{x}_3$	$\bar{x}_2 \cdot \bar{x}_4$
PG ₁	$x_1 \cdot x_2 \cdot \bar{x}_3$						X	o				
PG ₂	$x_1 \cdot \bar{x}_3 \cdot x_4$					X		o				
PG ₄	$\bar{x}_1 \cdot x_3 \cdot \bar{x}_4$		X					o	o			
PG ₅	$\bar{x}_1 \cdot x_2 \cdot \bar{x}_4$			X				o	o			
PG ₆	$x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$				X			o	o			
PG ₇	$x_3 \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$	X	X					o			o	
PG ₈	$x_2 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_3$			X	X				o	o		

(3) PG₁•
 (3) PG₂•
 3
 3
 3
 (3) PG₇•
 3

Do zbioru (G_{\min} , T_{\min}) dodatkowo więc wchodzi:

PG_8 , $\bar{x}_1 \cdot \bar{x}_3$, \bar{x}_4

Tabela 14

$$\bar{x}_4 \cdot \bar{x}_1 \bar{x}_3$$

(1) (2)

Zredukowana tablica AA		4 6	4 12	\bar{x}_4	$\frac{\bar{x}_1 \bar{x}_3}{\bar{x}_4}$	
PG ₅	$\bar{x}_1 \cdot x_2 \cdot \bar{x}_4$	X		o		3
PG ₆	$x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$		X	o		3
PG ₈	$x_2 \cdot \bar{x}_4 \cdot x_1 \cdot \bar{x}_3$	X	X	o o		(3) PG ₈

Zbiór (G_{\min}, T_{\min})

$$G_{\min} = [PG_1, PG_2, PG_7, PG_8]$$

$$T_{\min} = [\bar{x}_1 \bar{x}_3, \bar{x}_2 \bar{x}_4, \bar{x}_4 \bar{x}_1 \bar{x}_3]$$

Minimalne bezhazardowe wyrażenie TANT realizujące funkcję F

$$S = x_1 \cdot x_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_3 \cdot x_4 + x_3 \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4 + x_2 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_3$$

Koszt C_{TANT}

$$\lambda_{TANT} = 10$$

$$\mu_{TANT} = 23$$

Przykład 4.2.4.2

Niech będzie zadana funkcja $F = \bar{x}_1 \bar{x}_2 \bar{x}_3 + \bar{x}_2 x_3 \bar{x}_4 + \bar{x}_1 x_3 \bar{x}_4$

Maksymalne implikanty dopuszczalne funkcji F

$$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4, \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot x_4, x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2, \bar{x}_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4, x_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4$$

Krytyczne czynniki końca

$$\bar{x}_1, \bar{x}_2$$

Zbiór G funkcji F *)

$$PG_1 = x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$$

$$PG_2 = x_3 \cdot \bar{x}_1 \cdot \bar{x}_4$$

$$PG_3 = x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$$

$$PG_4 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3$$

$$PG_5 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$$

$$PG_6 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$$

Budowa rozwiniętej tablicy AA'

Tabela 15 przedstawia rozwiniętą tablicę AA' bez kolumn \bar{x}_1 i \bar{x}_2 .

Redukcja rozwiniętej tablicy AA'

Ponieważ z twierdzenia 4.2.2.2.1 nie można skorzystać, należy określić zastępowalne podstawowe czynniki końca.

Stosownie do twierdzenia 4.2.2.2.2

$$\bar{x}_3 \cdot \left(\frac{F}{\bar{x}_4}\right) = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \Leftarrow F, EN_1, EN_2 \text{ i } EN_3 \text{ zawierają } x_3, \bar{x}_4 \cancel{\bar{x}_3 \cdot \bar{x}_4},$$

$$\bar{x}_4 \cdot \left(\frac{F}{\bar{x}_3}\right) = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4 \Leftarrow F, \text{ nie ma } EN_0 \text{ zawierających } x_4, \bar{x}_3 \cancel{\bar{x}_3 \cdot \bar{x}_4}.$$

Stąd wynika, że \bar{x}_3 jest zastępowalne przez $\bar{x}_3 \cdot \bar{x}_4$.

Tabela 16 przedstawia tabelę 15 z usuniętą kolumną \bar{x}_3 .

Stosownie do twierdzenia 4.2.2.2.1

$$u_6 \text{ --- } u_4$$

Tabela 17 przedstawia tabelę 16 z usuniętym wierszem u_4 .

Stosownie do twierdzenia 4.2.2.2.3 otrzymujemy główne implikanty i podstawowe czynniki końca PG_6 , $\bar{x}_3 \cdot \bar{x}_4$. Usuwa się więc u_6 , u_5 i kolumnę $\bar{x}_3 \cdot \bar{x}_4$. Tabela 18 nie posiada wierszy u_5 i u_6 oraz kolumny $\bar{x}_3 \cdot \bar{x}_4$ i jest zredukowaną rozwiniętą tablicą AA'. Rozwiązaniem tej tabeli jest wiersz u_3 .

Do zbioru (G_{\min}, T_{\min}) dodatkowo więc wchodzi

$$PG_3, \bar{x}_1 \cdot \bar{x}_2, \bar{x}_4.$$

*) Realizacja algorytmu generacji zbioru G przedstawiona jest w przypisach

Zbiór (G_{\min}, T_{\min})

$$G_{\min} = [PG_3, PG_6]$$

$$T_{\min} [\bar{x}_1, \bar{x}_2, \bar{x}_4, \bar{x}_1 \cdot \bar{x}_2, \bar{x}_3 \cdot \bar{x}_4]$$

Minimalne bezhazardowe wyrażenie TANT realizujące funkcję F

$$S = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$$

Koszt C_{TANT}

$$\lambda_{TANT} = 8$$

$$\mu_{TANT} = 15$$

Wagi przypisane poszczególnym zmiennym w tabelach 15, 16, 17 i 18: $x_1 - 4, x_2 - 2, x_3 - 8, x_4 - 1$

Tabela 15

Rozwinięta tablica AA		1 1 2 2						
		0 1	0 8	8 10	8 12	\bar{x}_3	\bar{x}_4	
PG ₁	$x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$				X	o		3
PG ₂	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1$			X		o		3
PG ₃	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$			X	X	o	o	3
PG ₄	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3$	X				o		3
PG ₅	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$		X			o		3
PG ₆	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X				o	3

$\bar{x}_3 \rightarrow \bar{x}_3 \cdot \bar{x}_4$

1 2 2

Tabela 16

Rozwinięta tablica AA		0 1	0 8	8 10	8 12	\bar{x}_4 $\bar{x}_1\bar{x}_2$ $\bar{x}_3\bar{x}_4$	
PG ₁	$x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$				X	o	3
PG ₂	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1$			X		o	3
PG ₃	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$			X	X	o o	3
PG ₄	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3$	X				o	3
PG ₅	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$		X			o	3
PG ₆	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X			o	3

$u_6 \rightarrow u_4$

1 2 (2)

Tabela 17

Rozwinięta tablica AA		0 1	0 8	8 10	8 12	\bar{x}_4 $\bar{x}_1\bar{x}_2$ $\bar{x}_3\bar{x}_4$	
PG ₁	$x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$				X	o	3
PG ₂	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1$					o	3
PG ₃	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$			X	X	o o	3
PG ₅	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$		X			o	3
PG ₆	$\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X			o	3

$x_3 \cdot x_4$

PG₆

$\bar{x}_4 \cdot \bar{x}_1\bar{x}_2$
(1) (2)

Tabela 18

Zredukowana tablica AA		8 10	8 12	\bar{x}_4 $\bar{x}_1\bar{x}_2$	
PG ₁	$x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$		X	o o	3
PG ₂	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1$	X		o	3
PG ₃	$x_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$	X	X	o o	(3) PG ₃

Przykład 4.2.4.3

Niech będzie zadana funkcja $F = x_1 \cdot \bar{x}_2 + x_1 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_4 + \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_4$

Maksymalne implikanty dopuszczalne funkcji F

$\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$, $x_4 \cdot \bar{x}_2 \cdot \bar{x}_3$, $x_1 \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$, $x_1 \cdot x_4 \cdot \bar{x}_2 \cdot \bar{x}_3$, $x_1 \cdot x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$,
 $x_1 \cdot x_2 \cdot x_4 \cdot \bar{x}_3$, $x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$, $\bar{x}_1 \cdot x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$, $x_1 \cdot x_3 \cdot x_4 \cdot \bar{x}_2$, $x_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4$

Krytyczne czynniki końca

\bar{x}_2 .

Zbiór G funkcji F^{**})

$PG_1 = x_1, \bar{x}_2$

$PG_8 = x_1 \cdot x_4 \cdot \bar{x}_3 \cdot \bar{x}_2$

$PG_2 = x_1 \cdot \bar{x}_3$

$PG_9 = x_1 \cdot x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$

$PG_3 = x_1 \cdot \bar{x}_4$

$PG_{10} = x_1 \cdot x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$

$PG_4 = x_1 \cdot \bar{x}_2 \cdot \bar{x}_3$

$PG_{11} = \bar{x}_2 \cdot \bar{x}_3$

$PG_5 = x_1 \cdot \bar{x}_2 \cdot \bar{x}_4$

$PG_{12} = \bar{x}_2 \cdot \bar{x}_4$

$PG_6 = x_1 \cdot \bar{x}_3 \cdot \bar{x}_4$

$PG_{13} = \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$

$PG_7 = x_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4$

Budowa rozwiniętej tablicy AA'

Tabela 19 przedstawia rozwiniętą tablicę AA' bez kolumny \bar{x}_2 .

Redukcja rozwiniętej tablicy AA'

Stosownie do twierdzenia 4.2.2.2.1

$u_4 \rightarrow u_8, u_5 \rightarrow u_{10}, u_6 \rightarrow u_9$.

Rozwiniętą tablicę AA' po tej redukcji przedstawia tabela 20.

Stosownie do twierdzenia McCluskey'a o dominacji i równości kolumn pokrycia uzyskuje się:

$c_2 \rightarrow c_1, c_1 = c_4, c_5 \rightarrow c_1, c_7 \rightarrow c_3, c_3 = c_{12}, c_6 \rightarrow c_8, c_9 = c_{13},$

$c_{14} = c_{11}$

*) Realizacja algorytmu generacji zbioru G przedstawiona jest w przypisach

Wagi przypisane poszczególnym zmiennym: $x_1 - 4, x_2 - 2, x_3 - 8, x_4 - 1$

Tabela 19

Rozwinięta tablica AA														1	1	2	2	2	3				
		0 1	0 4	0 8	1 5	4 5	4 6	4 12	5 7	5 13	6 7	6 14	8 12	12 13	12 14	\bar{x}_3	\bar{x}_4	$\bar{x}_2 \cdot \bar{x}_3$	$\bar{x}_2 \cdot \bar{x}_4$		$\bar{x}_3 \cdot \bar{x}_4$	$\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	
PG ₁	$x_1 \cdot \bar{x}_2$					X		X		X				X									2
PG ₂	$x_1 \cdot \bar{x}_3$					X	X		X		X						o						2
PG ₃	$x_1 \cdot \bar{x}_4$						X	X				X			X		o						2
PG ₄	$x_1 \cdot \bar{x}_2 \cdot \bar{x}_3$					X	X	X	X	X	X			X				o					2
PG ₅	$x_1 \cdot \bar{x}_2 \cdot \bar{x}_4$					X	X	X		X		X		X	X				o				2
PG ₆	$x_1 \cdot \bar{x}_3 \cdot \bar{x}_4$					X	X	X	X		X	X			X					o			2
PG ₇	$x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$					X	X	X	X	X	X	X		X	X							o	2
PG ₈	$x_1 \cdot x_4 \cdot \bar{x}_2 \cdot \bar{x}_3$								X	X								o					3
PG ₉	$x_1 \cdot x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$										X	X									o		3
PG ₁₀	$x_1 \cdot x_3 \cdot \bar{x}_2 \cdot \bar{x}_4$													X	X				o				3
PG ₁₁	$\bar{x}_2 \cdot \bar{x}_3$	X	X		X	X											o						2
PG ₁₂	$\bar{x}_2 \cdot \bar{x}_4$		X	X				X					X				o						2
PG ₁₃	$\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X	X	X	X		X					X								o		2

$u_4 - u_8$

$u_5 - u_{10}$

$u_6 - u_9$

Tabela 20

Rozwinięta tablica AA		0	0	0	1	4	4	4	5	5	6	8	8	12	12								
		1	4	8	5	5	6	12	7	13	7	14	12	13	14	\bar{x}_3	\bar{x}_4	$\bar{x}_2\bar{x}_3$	$\bar{x}_2\bar{x}_4$	$\bar{x}_3\bar{x}_4$	$\bar{x}_2\bar{x}_3\bar{x}_4$		
PG ₁	$x_1\bar{x}_2$					X		X		X				X								2	
PG ₂	$x_1\bar{x}_3$					X	X		X		X					o						2	$c_2 - c_1$
PG ₃	$x_1\bar{x}_4$						X	X				X			X		o					2	$c_1 = c_4$
PG ₄	$x_1\bar{x}_2\bar{x}_3$					X	X	X	X	X	X			X				o				2	$c_5 - c_1$
PG ₅	$x_1\bar{x}_2\bar{x}_3$					X	X	X		X		X		X	X				o			2	$c_7 - c_3$
PG ₆	$x_1\bar{x}_3\bar{x}_4$					X	X	X	X		X	X			X					o		2	$c_3 = c_{12}$
PG ₇	$x_1\bar{x}_2\bar{x}_3\bar{x}_4$					X	X	X	X	X	X	X		X	X						o	2	$c_6 - c_8$
PG ₁₁	$\bar{x}_2\bar{x}_3$	X	X		X	X										o						2	$c_9 = c_{13}$
PG ₁₂	$\bar{x}_2\bar{x}_4$		X	X				X					X				o					2	$c_{14} = c_{11}$
PG ₁₃	$\bar{x}_2\bar{x}_3\bar{x}_4$	X	X	X	X	X		X					X								o	2	
		c_1	c_3	c_5	c_7	c_9	c_{11}	c_{13}															

Tabela 21 przedstawia tabelę 20 z usuniętymi kolumnami pokrycia.

Stosownie do twierdzenia 4.2.2.2.2

$$\bar{x}_3 \cdot \left(\frac{F}{\bar{x}_2}\right) = \bar{x}_3 \neq F,$$

$$\bar{x}_2 \neq \bar{x}_2 \bar{x}_3, \bar{x}_2 \neq \bar{x}_2 \bar{x}_4, \bar{x}_2 \neq \bar{x}_2 \bar{x}_3 \bar{x}_4$$

$$\bar{x}_4 \cdot \left(\frac{F}{\bar{x}_2}\right) = \bar{x}_4 \neq F,$$

$$\bar{x}_4 \cdot \left(\frac{F}{\bar{x}_3}\right) = \bar{x}_4 x_1 + \bar{x}_4 \bar{x}_2 \Leftarrow F, \text{ żadne } EN_0 \text{ nie zawiera } x_4, \bar{x}_3 \leftarrow \bar{x}_3 \cdot \bar{x}_4$$

$$\bar{x}_2 \cdot \left(\frac{F}{\bar{x}_3}\right) = \bar{x}_2 \neq F,$$

$$\bar{x}_3 \neq \bar{x}_2 \bar{x}_3, \bar{x}_3 \neq \bar{x}_2 \bar{x}_3 \bar{x}_4$$

$$\bar{x}_2 \cdot \left(\frac{F}{\bar{x}_4}\right) = \bar{x}_2 \neq F,$$

$$\bar{x}_4 \neq \bar{x}_2 \bar{x}_4, \bar{x}_4 \neq \bar{x}_2 \bar{x}_3 \bar{x}_4$$

$$\bar{x}_3 \cdot \left(\frac{F}{\bar{x}_4}\right) = x_1 \bar{x}_3 + \bar{x}_3 \bar{x}_2 \Leftarrow F, \text{ żadne } EN_0 \text{ nie zawiera } x_3, \bar{x}_4 \leftarrow \bar{x}_3 \cdot \bar{x}_4$$

Tabela 21

Rozwinięta tablica AA		0	0	5	5	6	6						
		1	8	7	13	7	14	\bar{x}_3	\bar{x}_4	$\bar{x}_2 \bar{x}_3$	$\bar{x}_2 \bar{x}_4$	$\bar{x}_3 \bar{x}_4$	$\bar{x}_2 \bar{x}_3 \bar{x}_4$
PG ₁	$x_1 \cdot \bar{x}_2$				X								
PG ₂	$x_1 \cdot \bar{x}_3$			X		X		o					
PG ₃	$x_1 \cdot \bar{x}_4$						X	o					
PG ₄	$x_1 \cdot \bar{x}_2 \cdot x_3$			X	X	X			o				
PG ₅	$x_1 \cdot \bar{x}_2 \cdot \bar{x}_4$				X	X				o			
PG ₆	$x_1 \cdot \bar{x}_3 \cdot \bar{x}_4$			X		X	X				o		
PG ₇	$x_1 \cdot \bar{x}_2 \cdot x_3 \cdot \bar{x}_4$			X	X	X	X					o	
PG ₁₁	$\bar{x}_2 \cdot \bar{x}_3$	X						o					
PG ₁₂	$\bar{x}_2 \cdot \bar{x}_4$		X					o					
PG ₁₃	$\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X								o		

$$\bar{x}_3 \leftarrow \bar{x}_3 \cdot \bar{x}_4$$

$$x_4 \leftarrow x_3 \cdot x_4$$

Tabela 22

Rozwinięta tablica AA		0 1	0 8	5 7	5 13	6 7	6 14	2 $\bar{x}_2\bar{x}_3$	2 $\bar{x}_2\bar{x}_4$	2 $\bar{x}_3\bar{x}_4$	3 $\bar{x}_2\bar{x}_3\bar{x}_4$	
PG ₁	$x_1 \cdot \bar{x}_2$				X							2
PG ₂	$x_1 \cdot \bar{x}_3$			X		X			o			2
PG ₃	$x_1 \cdot \bar{x}_4$						X		o			2
PG ₄	$x_1 \cdot \overline{\bar{x}_2 \cdot \bar{x}_3}$			X	X	X		o				2
PG ₅	$x_1 \cdot \overline{\bar{x}_2 \cdot \bar{x}_4}$				X		X		o			2
PG ₆	$x_1 \cdot \overline{\bar{x}_3 \cdot \bar{x}_4}$			X		X	X		o			2
PG ₇	$x_1 \cdot \overline{\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4}$			X	X	X	X			o		2
PG ₁₁	$\bar{x}_2 \cdot \bar{x}_3$	X							o			2
PG ₁₂	$\bar{x}_2 \cdot \bar{x}_4$		X						o			2
PG ₁₃	$\bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$	X	X						o			2

$u_6 \rightarrow u_2$,

$u_6 \rightarrow u_3$

$u_{13} \rightarrow u_{11}$,

$u_{13} \rightarrow u_{12}$

Stąd wynika, że \bar{x}_3 oraz \bar{x}_4 zastępowalne jest przez $\bar{x}_3 \cdot \bar{x}_4$.

Tabela 22 przedstawia tabelę 21 z usuniętymi kolumnami \bar{x}_3 i \bar{x}_4 .

Stosownie do twierdzenia 4.2.2.2.1

$u_6 \rightarrow u_2, u_6 \rightarrow u_3, u_{13} \rightarrow u_{11}, u_{13} \rightarrow u_{12}$.

Tabela 23 przedstawia tabelę 22 z uwzględnieniem powyższej redukcji.

Stosownie do twierdzenia 4.2.2.2.3 i twierdzenia McCluskey'a o redukcji kolumn pokrycia otrzymujemy główne implikanty i podstawowe czynniki końca:

$PG_{13} \bullet, \overline{\bar{x}_3 \bar{x}_4} \bullet$

Tabela 23

Rozwinięta tablica AA'								$\overline{x_3 \cdot x_4}$	
		0 1	0 8	5 7	5 13	6 7	6 14	1 1	(2) 3
PG ₁	$x_1 \cdot \overline{x_2}$				X			$\overline{x_2 x_3}$	2
PG ₄	$x_1 \cdot \overline{x_2} \cdot \overline{x_3}$			X	X	X		$\overline{x_2 x_4}$	2
PG ₅	$x_1 \cdot \overline{x_2} \cdot \overline{x_4}$				X		X	$\overline{x_3 x_4}$	2
PG ₆	$x_1 \cdot \overline{x_3} \cdot \overline{x_4}$			X		X	X	$\overline{x_2 x_3 x_4}$	2
PG ₇	$x_1 \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}$			X	X	X	X		2
PG ₁₃	$\overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}$	X	X						(2) PG ₁₃

$c_8 = 0_{10}$

Tabela 24

Zredukowana tablica AA'					$\overline{x_2 \cdot x_3 \cdot x_4}$	
		5 13	6 7	6 14	2 2	(3)
PG ₁	$x_1 \cdot \overline{x_2}$	X				(2) PG ₁
PG ₄	$x_1 \cdot \overline{x_2} \cdot \overline{x_3}$	X	X		o	2
PG ₅	$x_1 \cdot \overline{x_2} \cdot \overline{x_4}$	X		X	o	2
PG ₆	$x_1 \cdot \overline{x_3} \cdot \overline{x_4}$		X	X		(2) PG ₆
PG ₇	$x_1 \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}$	X	X	X	o	(2) PG ₇

Tabela 24 przedstawia zredukowaną rozwiniętą tablicę AA'.
 Rozwiązaniem tej tablicy jest wiersz u_7 lub zbiór $[u_1, u_6]$.
 Do zbioru (G_{\min}, T_{\min}) dodatkowo więc wchodzi

PG₇•, $\overline{x_2 x_3 x_4}$ • lub PG₁•, PG₆•,

Zbiór (G_{\min}, T_{\min})

$$G_{\min} = [PG_{13}, PG_7] \text{ lub } G_{\min} = [PG_{13}, PG_1, PG_6]$$

$$T_{\min} = [\bar{x}_2, \overline{x_3 \cdot x_4}, \overline{x_2 \cdot x_3 \cdot x_4}] \text{ lub } T_{\min} = [\bar{x}_2, \overline{x_3 \cdot x_4}]$$

Minimalne bezhazardowe wyrażenia TANT funkcji F

$$S = \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_1 \cdot \overline{x_2 \cdot x_3 \cdot x_4} \text{ lub } S' = x_1 \cdot \bar{x}_2 + x_1 \cdot \overline{x_3 \cdot x_4} + \bar{x}_2 \cdot \overline{x_3 \cdot x_4}$$

Koszt C_{TANT} obu wyrażeń

$$\lambda_{TANT} = 6$$

$$\mu_{TANT} = 12$$

4.3. Adaptacja znanych metod faktoryzacji do algebraicznego sposobu syntezy bezhazardowych układów TANT

Postacią optymalną^{*)} funkcji Boole'a F nazywa się sumę zawierającą minimalną liczbę prostych implikantów niezbędnych do pokrycia wszystkich par sąsiednich stanów wejścia, A i A', dla których $f(A) = f(A') = 1$. Ponieważ proste implikanty, zgodnie z definicją Gimpela^{**)}, są wyższymi implikantami dopuszczalnymi, więc tym samym wyrażenie o postaci optymalnej funkcji F jest zawsze wyrażeniem TANT. Większość takich wyrażeń TANT można przekształcić z uwzględnieniem pewnego ograniczenia za pomocą prawa rozdzielności mnożenia względem dodawania. Ograniczenie to polega na tym, że w nawiasie sumy logicznej mogą znaleźć się tylko zmienne zanegowane.

$$\begin{aligned} & H \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_{n-4} \cdot \bar{x}_{n-3} + H \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_{n-4} \cdot \bar{x}_{n-2} = \\ & = H \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_{n-4} (\bar{x}_{n-3} + \bar{x}_{n-2}) = \quad (4.3.1) \\ & = H \cdot \bar{x}_1 \cdot \bar{x}_2 \cdot \dots \cdot \bar{x}_{n-4} \cdot \overline{x_{n-3} \cdot x_{n-2}} \end{aligned}$$

*) Metody syntezy postaci optymalnej funkcji n-zmiennych przedstawiono w pracach [48] i [31]

**) Zob. Przypisy do rozdz. 4

Stosowanie powyższego prawa do przekształcenia optymalnej postaci funkcji F pozwala uzyskiwać wyrażenia dopuszczalne przedstawiające quasi proste implikanty dopuszczalne. Kilkakrotnie stosowanie tego prawa zgodnie z poniższą zasadą pozwala uzyskiwać wyrażenia dopuszczalne

$$H.\bar{x}_1.\bar{x}_2 \dots \bar{x}_{h-4}.\bar{x}_{h-1}.x_{h-1}.\overline{x_{h-3}.\bar{x}_{h-2}} + H.\bar{x}_1.\bar{x}_2 \dots \bar{x}_{h-4}.\bar{x}_h.\overline{x_{h-3}.\bar{x}_{h-2}} = H.\bar{x}_1.\bar{x}_2 \dots \bar{x}_{h-4}.\overline{x_{h-3}.\bar{x}_{h-2}.\bar{x}_{h-1}.\bar{x}_h}$$

przedstawiające złożone implikanty dopuszczalne.

Twierdzenie 4.3.1

Każde wyrażenie dopuszczalne E wyrażenia TANT uzyskanego w wyniku przekształceń optymalnej postaci funkcji F za pomocą zależności (4.3.1) przedstawia główny implikant funkcji F.

Dowód:

Każdy quasi prosty implikant uzyskany w wyniku przekształcenia nie zawiera w sobie ani jednego z pozostałych implikantów dopuszczalnych uzyskanych w wyniku tego samego przekształcenia.

Stosowanie zależności (4.3.1) nie narusza warunków zapewniających bezhazardowość przekształconego wyrażenia [31] i biorąc jednocześnie pod uwagę twierdzenia 4.3.1 i 4.2.1.2, należy przypuszczać, że w wyniku kilkakrotnego (o ile to możliwe) stosowania zależności (4.3.1) uzyska się w większości przypadków minimalne lub bliskie minimalnemu bezhazardowe wyrażenie TANT. Przytoczone w następnym podrozdziale przykłady potwierdzają słuszność tego przypuszczenia.

4.3.1. Przykłady

Przykład 4.3.1.1

Niech będzie optymalna postać funkcji F

$$S = \bar{x}_1.\bar{x}_2.x_3 + \bar{x}_1.x_3.y + \bar{x}_1.x_2.y + x_2.\bar{x}_3.y + x_1.\bar{x}_2.y + x_1.\bar{x}_3.y + \bar{x}_2.x_3.y + \bar{x}_1.x_2.\bar{x}_3$$

Po uwzględnieniu zależności (4.3.1)

$$S' = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_3 \cdot y \cdot (\bar{x}_1 + \bar{x}_2) + x_1 \cdot y \cdot (\bar{x}_2 + \bar{x}_3) + x_2 \cdot y \cdot (\bar{x}_1 + \bar{x}_3) + \bar{x}_1 \cdot x_2 \cdot \bar{x}_3$$

$$S'' = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + x_3 \cdot y \cdot \bar{x}_1 \cdot \bar{x}_2 + x_1 \cdot y \cdot \bar{x}_2 \cdot \bar{x}_3 + x_2 \cdot y \cdot \bar{x}_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot x_2 \cdot \bar{x}_3$$

Koszt C_{TANT} dla wyrażenia S''

$$\lambda_{TANT} = 12$$

$$\mu_{TANT} = 29$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 12$$

$$\mu_{TANT} = 35$$

P r z y k ł a d 4.3.1.2

Niech będzie optymalna postać funkcji F

$$S = \bar{x}_1 \cdot x_3 \cdot y_2 + \bar{x}_2 \cdot x_3 \cdot y_2 + \bar{x}_1 \cdot x_2 \cdot y_2 + x_1 \cdot \bar{x}_2 \cdot x_3 + x_1 \cdot x_2 \cdot \bar{y}_2 + x_1 \cdot x_3 \cdot \bar{y}_2 + \bar{x}_1 \cdot x_2 \cdot \bar{x}_3 + x_2 \cdot \bar{x}_3 \cdot \bar{y}_2$$

Po uwzględnieniu zależności (4.3.1)

$$S' = x_3 \cdot y_2 \cdot (\bar{x}_1 + \bar{x}_2) + \bar{x}_1 \cdot x_2 \cdot y_2 + x_1 \cdot x_3 \cdot (\bar{x}_2 + \bar{y}_2) + x_1 \cdot x_2 \cdot \bar{y}_2 + x_2 \cdot \bar{x}_3 \cdot (\bar{x}_1 + \bar{y}_2)$$

$$S'' = x_3 \cdot y_2 \cdot \bar{x}_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot x_2 \cdot y_2 + x_1 \cdot x_3 \cdot \bar{x}_2 \cdot \bar{y}_2 + x_1 \cdot x_2 \cdot \bar{y}_2 + x_2 \cdot \bar{x}_3 \cdot \bar{x}_1 \cdot \bar{y}_2$$

Koszt C_{TANT} dla wyrażenia S''

$$\lambda_{TANT} = 12$$

$$\mu_{TANT} = 29$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 13$$

$$\mu_{TANT} = 36$$

Przykład 4.3.1.3

Niech będzie optymalna postać funkcji F

$$S = x_1 \cdot x_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_3 \cdot x_4 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3 + \bar{x}_1 \cdot x_3 \cdot \bar{x}_4 + \bar{x}_1 \cdot x_2 \cdot \bar{x}_4 + x_2 \cdot \bar{x}_3 \cdot \bar{x}_4$$

Po uwzględnieniu zależności (4.3.1)

$$S' = x_1 \cdot x_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_3 \cdot x_4 + x_3 \cdot \bar{x}_1 \cdot (\bar{x}_2 + \bar{x}_4) + x_2 \cdot \bar{x}_4 \cdot (\bar{x}_3 + \bar{x}_4)$$

$$S'' = x_1 \cdot x_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_3 \cdot x_4 + x_3 \cdot \bar{x}_1 \cdot \overline{\bar{x}_2 \cdot \bar{x}_4} + x_2 \cdot \bar{x}_4 \cdot \overline{\bar{x}_3 \cdot \bar{x}_4}$$

Koszt C_{TANT} dla wyrażenia S''

$$\lambda_{TANT} = 10$$

$$\mu_{TANT} = 23$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 11$$

$$\mu_{TANT} = 28$$

Przykład 4.3.1.4

Niech będzie optymalna postać funkcji F

$$S = x_3 \cdot \bar{x}_2 \cdot \bar{x}_4 + x_3 \cdot \bar{x}_1 \cdot \bar{x}_4 + \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_4$$

Po uwzględnieniu zależności (4.3.1)

$$S' = x_3 \cdot \bar{x}_4 \cdot (\bar{x}_2 + \bar{x}_1) + \bar{x}_1 \cdot \bar{x}_2 \cdot (\bar{x}_3 + \bar{x}_4)$$

$$S'' = x_3 \cdot \bar{x}_4 \cdot \overline{\bar{x}_2 \cdot \bar{x}_1} + \bar{x}_1 \cdot \bar{x}_2 \cdot \overline{\bar{x}_3 \cdot \bar{x}_4}$$

Koszt C_{TANT} dla wyrażenia S''

$$\lambda_{TANT} = 8$$

$$\mu_{TANT} = 15$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 9$$

$$\mu_{TANT} = 20$$

Przykład 4.3.1.5

Niech będzie optymalna postać funkcji F

$$S = x_1 \cdot \bar{x}_2 + x_1 \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_4 + x_1 \cdot \bar{x}_4$$

Po uwzględnieniu zależności (4.3.1)

$$S' = x_1 \bar{x}_2 + x_1 (\bar{x}_3 + \bar{x}_4) + \bar{x}_2 \cdot (\bar{x}_3 + \bar{x}_4) \quad \text{lub} \quad S' = x_1 \cdot (\bar{x}_2 + \bar{x}_3 + \bar{x}_4) + \bar{x}_2 \cdot (\bar{x}_3 + \bar{x}_4)$$

$$S'' = x_1 \cdot \bar{x}_2 + x_1 \cdot \bar{x}_3 \bar{x}_4 + \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 \quad \text{lub} \quad S'' = x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4$$

Koszt C_{TANT} dla dwóch różnych wyrażeń S''

$$\lambda_{TANT} = 6$$

$$\mu_{TANT} = 12$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 9$$

$$\mu_{TANT} = 18$$

Przykład 4.3.1.6

Niech będzie optymalna postać funkcji F

$$S = x_5 \cdot \bar{x}_1 \cdot \bar{x}_2 + x_5 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_5 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_5 \cdot \bar{x}_1 \cdot \bar{x}_4$$

Po uwzględnieniu zależności (4.3.1)

$$S' = x_5 \cdot \bar{x}_2 (\bar{x}_1 + \bar{x}_3) + x_5 \cdot \bar{x}_4 (\bar{x}_1 + \bar{x}_3)$$

$$S'' = x_5 \cdot (\bar{x}_1 + \bar{x}_3) \cdot (\bar{x}_2 + \bar{x}_4)$$

$$S''' = x_5 \cdot \bar{x}_1 \cdot \bar{x}_3 \cdot \bar{x}_2 \cdot \bar{x}_4$$

Koszt C_{TANT} dla wyrażenia S''

$$\lambda_{TANT} = 4$$

$$\mu_{TANT} = 8$$

Koszt C_{TANT} dla optymalnej postaci funkcji

$$\lambda_{TANT} = 9$$

$$\mu_{TANT} = 20$$

4.4. Ogólna charakterystyka metod syntezy bezhazardowych układów TANT

Opisany w niniejszym rozdziale algorytm G^X pozwala na znalezienie bezhazardowego układu TANT realizującego funkcję n -zmiennych oraz zawierającego minimalną liczbę elementów NAND i minimalną (w zbiorze G) liczbę wejść na elementy NAND. Ponieważ algorytm G^X jest częściową adaptacją do syntezy bezhazardowych układów TANT metody Gimpela [11], dlatego spełnia on większość wymagań stawianych w ogólności metodom syntezy, które spełnia także metoda Gimpela. Algorytm G^X spełnia następujące z wymagań stawianych metodom syntezy kombinacyjnych układów przełączających:

- umożliwia syntezę układów przełączających realizujących funkcje n -zmiennych
- umożliwia uzyskanie minimalnych czasów propagacji układu przełączającego
- umożliwia uzyskanie minimalnej liczby elementów logicznych przy założonej bezhazardowej strukturze 3-warstwowej, zbiorze niezanegowanych zmiennych wejściowych i podstawowym elemencie NAND
- umożliwia uzyskanie minimalnej (w zbiorze G) liczby wejść na elementy NAND przy założeniach wymienionych wyżej.

Pełnej adaptacji metody Gimpela narzucono pewne konieczne ograniczenia związane z założeniem syntezy bezhazardowych układów przełączających. I tak syntezę prowadzi się wyłącznie na zbiorze głównych implikantów przedstawionych za pomocą nierozszerzonych wyrażeń dopuszczalnych. Ponadto ograniczono zasięg stosowania twierdzenia Gimpela o zastępowalności podstawowych

czynników końca. Wreszcie zrezygnowano w ostatniej fazie algorytmu G^x z budowy tablicy CC, ponieważ przy bezhazardowej syntezie jest ona zbyt rozbudowana, a w związku z tym jej redukcja jest zbyt pracochłonna. Nie uwzględnienie ograniczenia twierdzenia o zastępowalności podstawowych czynników końca prowadzi często do otrzymania układu TANT z hazardem statycznym w warunkach niedziałania i z hazardem dynamicznym.

Przykładem jest wyrażenie TANT

$$S = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_3 \cdot \bar{x}_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$$

gdzie zmienna x_3 występuje w początku H czynnika końca rozszerzonego wyrażenia dopuszczalnego $x_3 \cdot \bar{x}_3 \cdot \bar{x}_4 \cdot \bar{x}_1 \cdot \bar{x}_2$. Z poniżej przeprowadzonej identyfikacji $HP^k(x_1)$ wynika, że układ TANT będzie posiadał $HSO^1(x_3)$ oraz $HD^1(x_3)$.

	$x_1 x_2$					$x_1 x_2$					$x_1 x_2$			
	00 01 11 10					00 01 11 10					00 01 11 10			
$x_3 x_4$	00	1				00					00	1		
	01	1				01					01	1		
	11					11					11		1	1
	10	1				10	1	1	1		10	1	1	
	$\bar{x}_1 \cdot \bar{x}_2 \cdot (\bar{x}_3 + \bar{x}_4)$				+	$x_3 \cdot (\bar{x}_3 + \bar{x}_4) \cdot (\bar{x}_1 + \bar{x}_2)$				=				

Przyczyną takiego stanu rzeczy jest wprowadzenie zastępowalności $\bar{x}_4 \leftarrow \bar{x}_3 \cdot \bar{x}_4$.

Druga metoda, algebraiczna, pozwala również na znalezienie bezhazardowych układów TANT o minimalnej lub o bliskiej minimalnej liczbie elementów NAND. Wykorzystanie do przekształceń, za pomocą zależności (4.3.1), optymalnej postaci funkcji uzyskiwanej np. prosto z siatki Karnaugh'a, pozwala otrzymać w efekcie przekształceń, nowe bezhazardowe wyrażenie w postaci nierozszerzonych wyrażen dopuszczalnych przedstawiających zawsze główne implikanty funkcji F. Uzasadnia to przypuszczenie, że przekształcona postać może być minimalna lub bliska minimalnej. Rezultaty uzyskane w przykładach 4.3.1.3, 4.3.1.4 i 4.3.1.5 są identyczne z rezultatami uzyskanymi za pomocą algorytmu G^x .

w przykładach 4.2.4.1, 4.2.4.2 i 4.2.4.3 i potwierdzają słuszność tego przypuszczenia. Te same przykłady i przykłady 4.3.1.1, 4.3.1.2 i 4.3.1.6 również wskazują, że układ realizujący postać optymalną funkcji na elementach NAND aczkolwiek jest zawsze bezhazardowym układem TANT to jednak w przeważającej liczbie przypadków jest zawsze zbudowany z większej liczby elementów NAND i wejść na elementy niż bezhazardowy układ TANT, uzyskany w drodze syntezy przeprowadzonej za pomocą metod przedstawionych w niniejszym rozdziale. Dostyć drastycznie ilustruje to przykład 4.3.1.5, w którym bezhazardowy układ TANT uzyskany w wyniku syntezy zaproponowanych w tej pracy jest tańszy o 3 elementy i 6 wejść na elementy.

Reasumując należy stwierdzić, że uzyskane w przykładach w p.4.2.4. i w p. 4.3.1 wyrażenia TANT są wyrażeniami minimalnymi spośród wszystkich możliwych bezhazardowych wyrażeń TANT realizujących daną funkcję. Algorytm G^X nadaje się do bezhazardowej syntezy układów TANT realizujących funkcję n -zmiennych i w tym celu można bazując na nim napisać program na maszynie cyfrową. Zupełnie dobre rezultaty przy wykorzystaniu maszyny IBM 7094 uzyskał Gimpel [11]. Czas potrzebny do obliczenia przez maszynę 80 minimalnych wyrażeń TANT funkcji 3-zmiennych, na podstawie programu napisanego przy wykorzystaniu jego algorytmu syntezy, wynosił 59 s. Biorąc pod uwagę, że algorytm G^X jest mniej skomplikowany, a szczególnie rozwiązanie rozwiniętej tablicy AA' jest mniej pracochłonne, czas obliczeniowy maszyny powinien być krótszy.

Metoda algebraiczna jest tzw. metodą intuicyjną, która przy dużej wprawie inżyniera może (nie zawsze) doprowadzić do wyniku optymalnego. Główną przyczyną niemożliwości uzyskania zawsze optymalnego wyniku jest niepełny zbiór G , którego fragmenty uzyskuje się w większości przypadków w trakcie przekształceń postaci optymalnej funkcji. Metoda algebraiczna, pomimo wymienionej wyżej podstawowej wady, ma również jeden zasadniczy wator - jest metodą prostą i mało pracochłonną. Biorąc pod uwagę, że dla funkcji do 6-zmiennych optymalną postać funkcji można uzyskać prosto z siatki Karnaugha zaleta tej metody dodatkowo wzrasta, a sama metoda dzięki temu ma cechy prostej metody inżynierskiej.

ZAKOŃCZENIE

Obecnie zostaną krótko podsumowane poruszone zagadnienia i osiągnięte w niniejszej pracy wyniki.

Rozdział 1 dosyć szczegółowo wprowadza zagadnienia poruszone w dalszej części pracy. Zawiera próbę uporządkowania i zdefiniowania terminologii związanej z hazardem. Podaje warunki zapewniające poprawną pracę potencjałowych asynchronicznych automatów sekwencyjnych i wykazuje na trzech przykładach możliwość szkodliwego wpływu hazardu podstawowego na prawidłową pracę tych automatów.

Rozdział 2 podaje metodę identyfikacji k -krotnego hazardu podstawowego ze względu na zmienną x_1 w wielowarstwowych kombinacyjnych układach przełączających realizujących funkcje n -zmiennych i zbudowanych na bazie elementów I, LUB NIE, NAND, NOR. Pierwsza część rozdziału zawiera podstawowe definicje, lematy i twierdzenia. Na ich podstawie zbudowano algorytm I^k , którego sześć kroków pozwala wykryć wszystkie k -krotne hazardy podstawowe. Zaproponowano przeprowadzanie identyfikacji k -krotnego hazardu podstawowego dwoma drogami, tj. za pomocą:

- a) składania tablicy strukturalnych implikantów i strukturalnych implimentów
- b) składania siatek Karnaugh

Podano dwa przykłady realizacji algorytmu I^k za pomocą składania tablicy PQK^m oraz podano pięć przykładów realizacji algorytmu I^k za pomocą składania siatek Karnaugh. Szczegółową charakterystykę przedstawionej metody identyfikacji k -krotnego hazardu podstawowego zawiera p. 3.2 rozdziału 2.

Rozdział 3 przedstawia sposób eliminacji dowolnych k -krotnych hazardów podstawowych na wyjściu dowolnego elementu logicznego wielowarstwowego układu przełączającego zbudowanego na elementach I, LUB, NAND, NOR i NIE. W pierwszej części rozdziału podano warunki jakie powinny spełniać wszystkie strukturalne implikanty PK^m i strukturalne implimenty QK^m bezhazardo-

wego wielowarstwowego układu przełączającego. W dalszej części rozdziału przedstawiono, na przykładzie, sposób wykorzystania do eliminacji wszystkich k -krotnych hazardów podstawowych w trójwarstwowym układzie przełączającym, podanego w rozdziale 2 algorytmu I^k i wspomnianych wyżej warunków. Sposób eliminacji wszystkich szkodliwych hazardów podstawowych w potencjałowych asynchronicznych układach sekwencyjnych przedstawiono w dwóch ostatnich przykładach.

Wszystkie przytoczone w rozdziale 3 przykłady wykazały całkowitą przydatność opisanego sposobu eliminacji k -krotnych hazardów podstawowych. Przedstawiony sposób nie daje jednak pewności, że uzyskany bezhazardowy wielowarstwowy układ przełączający jest najtańszym spośród wszystkich bezhazardowych wielowarstwowego układu przełączających realizujących daną funkcję. Problem ten rozwiązano dla pewnych przypadków w rozdziale 4, w którym na bazie metody Gimpela [11] przedstawiono metodę syntezy minimalnych bezhazardowych układów TANT realizujących dane funkcje n -zmiennych. Jako kryterium minimalności przyjęto za Gimpelam:

- a) minimalną liczbę elementów logicznych w układzie TANT
- b) minimalną liczbę wejść na elementy logiczne w układzie TANT spełniającym kryterium a.

Pierwsza część rozdziału zawiera podstawowe definicje i twierdzenia oraz algorytm G^X . W dalszej części podano trzy przykłady ręcznej realizacji algorytmu G^X , ilustrujące jego zastosowanie. Algorytm ten nadaje się całkowicie do bezhazardowej syntezy układów realizujących funkcje n -zmiennych i w tym celu można, bazując na nim, napisać program na komputer. W ostatniej części rozdziału przedstawiono adaptację znanych algebraicznych metod faktoryzacji do bezhazardowej syntezy układów TANT. Adaptacja ta jest intuicyjnym sposobem szukania minimalnego rozwiązania i nie daje pewności uzyskania optymalnego wyniku.

PRZYPISY

1. Przypisy do podrozdziału 2.1

Przejęciową funkcją przełączającą F_P nazywa się każde wyrażenie strukturalne kombinacyjnego układu przełączającego, którego zmienne oznaczone są indeksami zgodnie z zasadą zaproponowaną przez Huffmana [18], McCluskey'a w [31] lub zasadą McGhee w [35].

Przejęciową zmienną wejściową X_P nazywa się każdą oznaczoną indeksem zmienną z przejęciowej funkcji przełączającej [18], [31], [35].

Zbiorem P kombinacyjnego układu przełączającego nazywa się minimalny zbiór przejęciowych zmiennych wejściowych prostych i zanegowanych oraz takich, że wyjście kombinacyjnego układu przełączającego przyjmuje stan logiczny 1, kiedy tylko wszystkie zmienne zbioru P równe są stanowi logicznemu 1 [18], [31].

Zbiorem S kombinacyjnego układu przełączającego nazywa się minimalny zbiór przejęciowych zmiennych wejściowych prostych i zanegowanych oraz takich, że wyjście kombinacyjnego układu przełączającego przyjmuje stan logiczny 0, kiedy tylko wszystkie zmienne zbioru S równe są stanowi logicznemu 0 [18], [31].

Zbiorem 1 kombinacyjnego układu przełączającego nazywa się zbiór niepowtarzających się zmiennych z opuszczonymi indeksami ze zbioru P [31], [35].

Zbiorem 0 kombinacyjnego układu przełączającego nazywa się zbiór niepowtarzających się zmiennych z opuszczonymi indeksami ze zbioru S [31], [35].

2. Przypisy do podrozdziału 1.1 oraz podrozdziału 2.2

Stanem wejścia A kombinacyjnego układu przełączającego realizującego funkcję $F = f(x_1, x_2, \dots, x_n)$ nazywa się zbiór stanów logicznych $[a_1, a_2, \dots, a_n]$ zmiennych wejściowych x_i dla wszystkich $i \in [1, 2, \dots, n]$, [40].

Dwa stany wejścia A i A' nazywa się sąsiednimi, jeżeli $a_i \neq a'_i$ dla tylko jednego $i \in [1, 2, \dots, n]$ oraz $a_j = a'_j$ dla każdego $j \neq i$ [40].

Mówi się, że stan wejścia A zasila kombinacyjny układ przełączający realizujący funkcję $F = f(x_1, x_2, \dots, x_n)$ wtedy, gdy zmienne wejściowe x_i mają wartości stanów logicznych a_i dla każdego $i \in [1, 2, \dots, n]$, [40].

Stanem wyjścia kombinacyjnego układu przełączającego realizującego funkcję $F = f(x_1, x_2, \dots, x_n)$ i zasilanego stanem wejścia A nazywa się stan logiczny $f(a_1, a_2, \dots, a_n)$, [40].

Mówi się, że kombinacyjny układ przełączający zasilany stanem wejścia A jest w stanie stabilnym wtedy, gdy:

- a) wyjścia każdego elementu logicznego B mają stany logiczne określone przez stany logiczne na ich wejściach oraz funkcje logiczne realizowane przez te elementy,
- b) stany logiczne na wyjściach każdego elementu opóźnienia równe są stanom logicznym na ich wejściach [40].

Stabilnym zbiorem 1 nazywa się każdy zbiór 1, który nie zawiera w sobie żadnej pary złożonej z tej samej zmiennej zanegowanej i niezanegowanej [31], [35].

Stabilnym zbiorem 0 nazywa się każdy zbiór 0, który nie zawiera w sobie żadnej pary złożonej z tej samej zmiennej zanegowanej i niezanegowanej [31], [35].

Stan wejścia A jest pokryty przez stabilny zbiór 1:

jeżeli $x_i \in$ do zbioru 1, to wtedy $a_i = 1$ oraz
jeżeli $\bar{x}_j \in$ do zbioru 1, to wtedy $a_j = 0$ [31].

Stan wejścia A jest pokryty przez stabilny zbiór 0:

jeżeli $x_i \in$ do zbioru 0, to wtedy $a_i = 0$ oraz
jeżeli $\bar{x}_j \in$ do zbioru 0, to wtedy $a_j = 1$ [31].

3. Przypisy do podrozdziału 2.4

Twierdzenie 7.3.5 z pracy [31]

Hazard statyczny w warunkach działania (w warunkach niedziałania) istnieje w układzie przełączającym wtedy i tylko wtedy, jeżeli istnieje para stabilnych zbiorów 1 (zbiorów 0) $\{\alpha^x, \dots, \beta^x, \omega\}$, $\{\lambda^x, \dots, \theta^x, \bar{\omega}\}$ takich że:

- a) istnieje tylko jedna zmienna, która jest zanegowana w jednym ze zbiorów 1 (zbiorów 0) i niezanegowana w drugim zbiorze 1 (zbiorze 0) oraz dowolna liczba zmiennych może wystąpić w obu zbiorach 1 (zbiorach 0)
- b) każdy z innych stabilnych zbiorów 1 (zbiorów 0) posiada zmienną ω lub $\bar{\omega}$ lub zanegowaną jedną z innych zmiennych zawartą w obu zbiorach 1 (zbiorach 0) lub zmienną niezawartą w żadnym z obu zbiorów 1 (zbiorów 0).

Twierdzenie 7.3.3 z pracy [31]

Hazard statyczny w warunkach działania (w warunkach niedziałania) istnieje w układzie przełączającym realizującym funkcje F wtedy i tylko wtedy, gdy spełnione są dwa następujące warunki:

- a) istnieje zbiór 1 (zbiór 0) układu przełączającego

$$L = (a, b, \dots, x, \bar{x}, \dots, z^x)$$

w którym tylko jedna zmienna występuje w postaci zanegowanej i niezanegowanej,

- b) istnieje przynajmniej jedna para sąsiednich stanów wejścia A i A' spełniająca następujące warunki:
 - $f(A) = f(A') = 1$ (0)
 - zmienna x jest równa stanowi logicznemu 0 dla stanu wejścia A' i równa stanowi logicznemu 1 dla stanu wejścia A
 - każda inna zmienna (ale nie x) ze zbioru 1 jest równa stanowi logicznemu 1 (0) dla obu stanów wejścia.

Twierdzenie Huffmana [18]

Kombinacyjny układ przełączający realizujący funkcję F posiada hazard statyczny w warunkach działania (w warunkach niedziałania) wtedy i tylko wtedy, gdy:

- a) istnieje para sąsiednich stanów wejścia A i A', dla których $f(A) = f(A') = 1(0)$
- b) kombinacyjny układ przełączający nie ma zbioru 1 (zbioru 0) pokrywającego jednocześnie oba stany wejścia A i A'.

Przypisy do podrozdziału 2.5.2

Generacja tablicy PQK_0^m do przykładu 2.5.2.1

Wagi poszczególnych zmiennych

$$x_1 - 2^3(8), \quad x_2 - 2^2(4), \quad x_3 - 2^1(2), \quad x_4 - 2^0(1)$$

Obliczenie pokryć implikanta PK_5^1 i par sąsiednich tego pokrycia

$$S_5^1 = x_1 \cdot x_2 \cdot x_3, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & 1 & 1 & - \end{array}, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{array} \implies 14 \text{ i } 15$$

Obliczenie pokryć implikentów QK_4^2 i QK_3^2 oraz par sąsiednich tych pokryć

$$S_4^2 = x_1 + x_4, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 0 & - & - & 0 \end{array}, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{array} \implies 0, 2, 6, 4, \quad \begin{array}{c} 0 \\ 2 \\ 4 \\ 6 \end{array} \implies \begin{array}{cccc} 0 & 0 & 2 & 4 \\ 2, & 4, & 6, & 6 \end{array}$$

$$S_3^2 = \bar{x}_1 + \bar{x}_3, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & - & 1 & - \end{array}, \quad \begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{array} \implies 10, 11, 15, 14, \quad \begin{array}{c} 10 \\ 11 \\ 14 \\ 15 \end{array} \implies \begin{array}{cccc} 10 & 10 & 11 & 14 \\ 11, & 14, & 15, & 15 \end{array}$$

Interpretacja wyników identyfikacji uzyskanych w tablicy
2.5.2.4 z przykładu 2.5.2.1

Wynik identyfikacji	Interpretacja wyników identyfikacji
HSO ¹ dla $\frac{2}{10}$	$\begin{array}{cccc} 8 & 4 & 2 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{array} \Rightarrow - 0 1 0 \Rightarrow \text{HSO}^1(x_1) \text{ dla } x_2=0, x_3=1, x_4=0$ x_1
HS1 ¹ dla $\frac{12}{14}$	$\begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{array} \Rightarrow 1 1 - 0 \Rightarrow \text{HS1}^1(x_3) \text{ dla } x_1=1, x_2=1, x_4=0$ x_3
HS1 ¹ dla $\frac{13}{15}$	$\begin{array}{cccc} 8 & 4 & 2 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{array} \Rightarrow 1 1 - 1 \Rightarrow \text{HS1}^1(x_3) \text{ dla } x_1=1, x_2=1, x_4=1$ x_3
HS1 ¹ dla $\frac{7}{15}$	$\begin{array}{cccc} 8 & 4 & 2 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array} \Rightarrow - 1 1 1 \Rightarrow \text{HS1}^1(x_1) \text{ dla } x_2=1, x_3=1, x_4=1$ x_1
HD ¹ dla $\frac{6}{14}$	$\begin{array}{cccc} 8 & 4 & 2 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{array} \Rightarrow - 1 1 0 \Rightarrow \text{HD}^1(x_1) \text{ dla } x_2=1, x_3=1, x_4=0$ x_1

Przypisy do rozdziału 4

Podstawowe definicje i twierdzenia metody Gimpela syntezy minimalnych układów TANT oraz przykłady generacji zbioru G.

Definicja

Czynnikiem podstawowym T nazywa się iloczyn różnych niezanegowanych zmiennych lub jedynek Boole'a np.: $x_1, x_1 \cdot x_2 \cdot x_3 \cdot x_4, 1$.

Definicja

Wyrażeniem dopuszczalnym E nazywa się dowolne wyrażenie Boole'a różne od zera w postaci $T_0 \cdot \bar{T}_1 \cdot \bar{T}_2 \dots \bar{T}_s$ dla $s \geq 0$, gdzie \bar{T}_i dla $i \in [0, \dots, s]$ jest czynnikiem podstawowym. O czynniku podstawowym T_0 będzie się mówiło jako o początku wyrażenia dopuszczalnego i dalej oznaczać się go będzie przez H, natomiast wy-

rażenie $\bar{T}_1 \dots \bar{T}_g$ będzie nazywane końcem wyrażenia E, a każde \bar{T}_i będzie nazywane czynnikiem końca.

Definicja

Wyrażeniem (TANT) nazywa się dowolną sumę wyrażen dopuszczalnych

$$\text{np.: } E_1 + E_2 + \dots + E_k$$

Definicja

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

Definicja

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

Definicja

Nierozszerzonym wyrażeniem dopuszczalnym EN nazywa się podstawowe wyrażenie dla danego implikantu dopuszczalnego PD. Jego czynniki końca nazywa się podstawowymi czynnikami końca.

Definicja

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

Definicja

Maksymalny implikant dopuszczalny PM z początkiem H określane jest jako suma wszystkich implikantów dopuszczalnych z początkiem H.

Twierdzenie

Każdy implikant prosty jest implikantem dopuszczalnym PD.

Definicja

Każdy implikant dopuszczalny zawarty w implikancie prostym nazywa się implikantem niższym.

Definicja

Każdy implikant dopuszczalny, który nie jest niższym nazywa się wyższym.

Definicja

Quasi prostym implikantem dopuszczalnym PV nazywa się każdy implikant dopuszczalny, który ma najwyżej jeden czynnik końca będący sumą zmiennych zanegowanych.

Definicja

Implikantem dominującym nazywa się taki dopuszczalny implikant PO, który z każdym pozostałym implikantem dopuszczalnym PD spełnia następujący test

$$PO \cdot PD = 0 \quad \text{lub} \quad PO \supseteq PD \quad (PD \cdot \overline{PO} = 0)$$

Definicja

Głównym implikantem PG jest dowolny wyższy implikant PP nie zawarty całkowicie w dominującym quasi prostym implikancie.

Twierdzenie

Dla dowolnej funkcji F istnieje minimalne wyrażenie strukturalne TANT takie, że każdy implikant dopuszczalny PO jest głównym implikantem PG.

ALGORYTM GENERACJI ZBIORU G

1. Dysjunkcyjne normalne wyrażenie funkcji F
2. Generacja zbioru prostych implikantów PP funkcji F
3. Generacja zbiorów prostych implikantów zgodnych z początkiem H (spełniających test $PP \cdot H \subseteq PP$)
4. Generacja zbioru maksymalnych implikantów dopuszczalnych PM funkcji F, $PM = H \cdot PP_1 + H \cdot PP_2 + \dots + H \cdot PP_n$

5. Generacja zbioru wyższych implikantów dopuszczalnych PD,
test $PD \subset PP$

6. Test quasi prostego implikanta dominującego PO,
 $PO \cdot PD = 0, PO \equiv PD$

7. Zbiór G

Przykłady realizacji algorytmu generacji zbioru G

A. Dla przykładu 4.2.4.2

Niech będzie zadana funkcja $F = \bar{x}_1\bar{x}_2\bar{x}_3 + \bar{x}_2x_3\bar{x}_4 + \bar{x}_1x_3\bar{x}_4$.

REALIZACJA ALGORYTMU GENERACJI ZBIORU G

- Krok 1

Wyżej przedstawione wyrażenie jest dysjunkcyjnym normalnym wyrażeniem funkcji F.

- Krok 2

Szukamy zbioru prostych implikantów za pomocą siatki Karnaugha

		x_1x_2			
		00	01	11	10
x_1x_2	00	0			
	01	1			
	11				
	10	2	6		10

PP1 = $\bar{x}_1\bar{x}_2\bar{x}_3$

PP2 = $\bar{x}_1\bar{x}_2\bar{x}_4$

PP3 = $\bar{x}_1x_3\bar{x}_4$

PP4 = $\bar{x}_2x_3\bar{x}_4$

- Krok 3

Generujemy zbiór prostych implikantów zgodnych z początkiem H za pomocą tablicy prostych implikantów.

		0	1	2	6	10
PP1	$\bar{x}_1\bar{x}_2\bar{x}_3$	x	x			
PP2	$\bar{x}_1\bar{x}_2\bar{x}_4$	x		x		
PP3	$\bar{x}_1x_3\bar{x}_4$			x	x	x
PP4	$\bar{x}_2x_3\bar{x}_4$			x		x

- Krok 4

Tworzymy całkowity zbiór maksymalnych implikantów dopuszczalnych PM za pomocą poniższej tabeli.

Iloczyn podstawowy		Początek H	Suma implikantów prostych zgodnych z początkiem H	Maksymalny implikant dopuszczalny	
	$x_1x_2x_3x_4$				
0	0 0 0 0	1	$\bar{x}_1\bar{x}_2\bar{x}_3+\bar{x}_1\bar{x}_2\bar{x}_4$	PM1	$\bar{x}_1\bar{x}_2 \bar{x}_3\bar{x}_4$
1	0 0 0 1	x_4	$\bar{x}_1\bar{x}_2\bar{x}_3$	PM2	$x_4\bar{x}_1\bar{x}_2\bar{x}_3$
2	0 0 1 0	x_3	$\bar{x}_1\bar{x}_2\bar{x}_4+\bar{x}_1x_3\bar{x}_4+\bar{x}_2x_3\bar{x}_4$	PM3	$x_3\bar{x}_4\bar{x}_1\bar{x}_2$
6	0 1 1 0	x_2x_3	$\bar{x}_1x_3\bar{x}_4$	PM4	$x_2x_3\bar{x}_1\bar{x}_4$
10	1 0 1 0	x_1x_3	$\bar{x}_2x_3\bar{x}_4$	PM5	$x_1x_3\bar{x}_2\bar{x}_4$

- Krok 5

a) Tworzymy wszystkie implikanty dopuszczalne

$$\begin{array}{l} \text{PM1} \\ \bar{x}_1\bar{x}_2 \bar{x}_3\bar{x}_4 \\ \bar{x}_1\bar{x}_2\bar{x}_3 \\ \bar{x}_1\bar{x}_2\bar{x}_4 \end{array}$$

$$\text{PM2} \quad x_4\bar{x}_1\bar{x}_2\bar{x}_3$$

$$\begin{array}{l} \text{PM3} \\ x_3\bar{x}_4\bar{x}_1\bar{x}_2 \\ x_3\bar{x}_4\bar{x}_1 \\ x_3\bar{x}_4\bar{x}_2 \end{array}$$

$$\text{PM4} \quad x_2x_3\bar{x}_1\bar{x}_4$$

$$\text{PM5} \quad x_1x_3\bar{x}_2\bar{x}_4$$

b) Odrzucamy implikanty dopuszczalne niższe PM2, PM4 i PM5

$$\text{PM2} \subset \text{PM1}, \quad \text{PM4} \subset \text{PM3}, \quad \text{PM5} \subset \text{PM1}$$

c) Zbiór wyższych implikantów dopuszczalnych
Quasi proste implikanty dopuszczalne

$$PV1 = x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4$$

$$PV2 = x_3 \bar{x}_4 \bar{x}_1 \bar{x}_2$$

Proste implikanty dopuszczalne

$$PP1 = \bar{x}_1 \bar{x}_2 \bar{x}_3$$

$$PP2 = \bar{x}_1 \bar{x}_2 \bar{x}_4$$

$$PP3 = \bar{x}_1 x_3 \bar{x}_4$$

$$PP4 = \bar{x}_2 x_3 \bar{x}_4$$

- Krok 6

Test quasi prostego implikanta dominującego nie jest spełniony przez żaden quasi prosty implikant dopuszczalny.

- Krok 7

Wszystkie wyższe implikanty dopuszczalne uzyskane w kroku 5 są głównymi implikantami dopuszczalnymi.

Zbiór G

$$PG_1 = x_3 \bar{x}_2 \bar{x}_4$$

$$PG_3 = x_3 \bar{x}_4 \bar{x}_1 \bar{x}_2$$

$$PG_5 = \bar{x}_1 \bar{x}_2 \bar{x}_4$$

$$PG_2 = x_3 \bar{x}_1 \bar{x}_4$$

$$PG_4 = \bar{x}_1 \bar{x}_2 \bar{x}_3$$

$$PG_6 = \bar{x}_1 \bar{x}_2 \bar{x}_3 \bar{x}_4$$

B. Dla przykładu 4.2.4.3

Niech będzie zadana funkcja $F = x_1 \bar{x}_2 + x_1 \bar{x}_3 + x_1 \bar{x}_4 + \bar{x}_2 \bar{x}_3 + \bar{x}_2 \bar{x}_4$

ALGORYTM GENERACJI ZBIORU G

- Krok 1

Wyżej przedstawione wyrażenie jest dysjunkcyjnym normalnym wyrażeniem funkcji F.

- Krok 2

Szukany zbioru prostych implikantów za pomocą siatki Karnaugh.

$$x_3 x_4$$

		$x_1 x_2$		
		00	01	11
00	0			12
01	1			13
11				11
10	2			14

$$PP1 = x_1 \bar{x}_2 \quad PP4 = \bar{x}_2 \bar{x}_3$$

$$PP2 = x_1 \bar{x}_3 \quad PP5 = \bar{x}_2 \bar{x}_4$$

$$PP3 = x_1 \bar{x}_4$$

- Krok 3

Generujemy zbiór prostych implikantów zgodnych z początkiem H za pomocą tablicy prostych implikantów

		0	1	2	8	9	10	11	12	13	14
PP1	$x_1 \bar{x}_2$				x	x	x	x			
PP2	$x_1 \bar{x}_3$				x	x			x	x	
PP3	$x_1 \bar{x}_4$				x		x		x		x
PP4	$\bar{x}_2 \bar{x}_3$	x	x		x	x					
PP5	$\bar{x}_2 \bar{x}_4$	x		x	x		x				

- Krok 4

Tworzymy całkowity zbiór maksymalnych implikantów dopuszczalnych PM za pomocą poniższej tabeli

Iloczyn podstawowy		Początek H	Suma implikantów prostych zgodnych z początkiem H	Maksymalny implikant dopuszczalny	
	$x_1 x_2 x_3 x_4$				
0	0 0 0 0	1	$\bar{x}_2 \bar{x}_3 + \bar{x}_2 \bar{x}_4$	PM1	$\bar{x}_2 \bar{x}_3 \bar{x}_4$
1	0 0 0 1	x_4	$\bar{x}_2 \bar{x}_3$	PM2	$x_4 \bar{x}_2 \bar{x}_3$
2	0 0 1 0	x_3	$\bar{x}_2 \bar{x}_4$	PM3	$x_3 \bar{x}_2 \bar{x}_4$
8	1 0 0 0	x_1	$x_1 \bar{x}_2 + x_1 \bar{x}_3 + x_1 \bar{x}_4 + \bar{x}_2 \bar{x}_3 + \bar{x}_2 \bar{x}_4$	PM4	$x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4$
9	1 0 0 1	$x_1 x_4$	$x_1 \bar{x}_2 + x_1 \bar{x}_3 + \bar{x}_2 \bar{x}_3$	PM5	$x_1 x_4 \bar{x}_2 \bar{x}_3$
10	1 0 1 0	$x_1 x_3$	$x_1 \bar{x}_2 + x_1 \bar{x}_4 + \bar{x}_2 \bar{x}_4$	PM6	$x_1 x_3 \bar{x}_2 \bar{x}_4$
11	1 0 1 1	$x_1 x_3 x_4$	$x_1 \bar{x}_2$	PM7	$x_1 x_3 x_4 \bar{x}_2$
12	1 1 0 0	$x_1 x_2$	$x_1 \bar{x}_3 + x_1 \bar{x}_4$	PM8	$x_1 x_2 \bar{x}_3 \bar{x}_4$
13	1 1 0 1	$x_1 x_2 x_4$	$x_1 \bar{x}_3$	PM9	$x_1 x_2 x_4 \bar{x}_3$
14	1 1 1 0	$x_1 x_2 x_3$	$x_1 \bar{x}_4$	PM10	$x_1 x_2 x_3 \bar{x}_4$

- Krok 5

a) tworzymy wszystkie implikanty dopuszczalne

PM1	$\bar{x}_2\bar{x}_3\bar{x}_4$ $\bar{x}_2\bar{x}_3$ $\bar{x}_2\bar{x}_4$
PM2	$x_4\bar{x}_2\bar{x}_3$
PM3	$x_3\bar{x}_2\bar{x}_4$
PM4	$x_1\bar{x}_2\bar{x}_3\bar{x}_4$ $x_1\bar{x}_2\bar{x}_3$ $x_1\bar{x}_2\bar{x}_4$ $x_1\bar{x}_3\bar{x}_4$ $x_1\bar{x}_2$ $x_1\bar{x}_3$ $x_1\bar{x}_4$
PM5	$x_1x_4\bar{x}_2\bar{x}_3$ $x_1x_4\bar{x}_2$ $x_1x_4\bar{x}_3$
PM6	$x_1x_3\bar{x}_2\bar{x}_4$ $x_1x_3\bar{x}_2$ $x_1x_3\bar{x}_4$
PM7	$x_1x_3x_4\bar{x}_2$
PM8	$x_1x_2\bar{x}_3\bar{x}_4$ $x_1x_2\bar{x}_3$ $x_1x_2\bar{x}_4$
PM9	$x_1x_2x_4\bar{x}_3$
PM10	$x_1x_2x_3\bar{x}_4$

b) Odrzucamy implikanty niższe PM2, PM3, PM7, PM9, PM10 oraz

$x_1x_4\bar{x}_2$, $x_1x_4\bar{x}_3$, $x_1x_3\bar{x}_2$, $x_1x_3\bar{x}_4$, $x_1x_2\bar{x}_3$, $x_1x_2\bar{x}_4$

c) Zbiór wyższych implikantów dopuszczalnych

Quasi proste implikanty dopuszczalne

$$\begin{aligned}PV1 &= \bar{x}_2 \bar{x}_3 \bar{x}_4 \\PV2 &= x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 \\PV3 &= x_1 \bar{x}_2 \bar{x}_3 \\PV4 &= x_1 \bar{x}_2 \bar{x}_4 \\PV5 &= x_1 \bar{x}_3 \bar{x}_4 \\PV6 &= x_1 x_4 \bar{x}_2 \bar{x}_3 \\PV7 &= x_1 x_3 \bar{x}_2 \bar{x}_4 \\PV8 &= x_1 x_2 \bar{x}_3 \bar{x}_4\end{aligned}$$

Proste implikanty dopuszczalne

$$\begin{aligned}PP1 &= x_1 \bar{x}_2 \\PP2 &= x_1 \bar{x}_3 \\PP3 &= x_1 \bar{x}_4 \\PP4 &= \bar{x}_2 \bar{x}_3 \\PP5 &= \bar{x}_2 \bar{x}_4\end{aligned}$$

- Krok 6

Test quasi prostego implikanta dominującego nie jest spełniony przez żaden quasi prosty implikant dopuszczalny

- Krok 7

Wszystkie wyższe implikanty dopuszczalne uzyskane w kroku 5 są głównymi implikantami dopuszczalnymi

Zbiór G

$$\begin{aligned}PG_1 &= x_1 \bar{x}_2 & PG_8 &= x_1 x_4 \bar{x}_2 \bar{x}_3 \\PG_2 &= x_1 \bar{x}_3 & PG_9 &= x_1 x_2 \bar{x}_3 \bar{x}_4 \\PG_3 &= x_1 \bar{x}_4 & PG_{10} &= x_1 x_3 \bar{x}_2 \bar{x}_4 \\PG_4 &= x_1 \bar{x}_2 \bar{x}_3 & PG_{11} &= \bar{x}_2 \bar{x}_3 \\PG_5 &= x_1 \bar{x}_2 \bar{x}_4 & PG_{12} &= \bar{x}_2 \bar{x}_4 \\PG_6 &= x_1 \bar{x}_3 \bar{x}_4 & PG_{13} &= \bar{x}_2 \bar{x}_3 \bar{x}_4 \\PG_7 &= x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4\end{aligned}$$

BIBLIOGRAFIA

- [1] ARMSTRONG D.B., FRIEDMAN A.D.: Realization of Asynchronous Sequential Circuits Without Inserted Delay Elements, IEEE Transactions on Computers, 1968, t. C-17, nr 2
- [2] BROMIRSKI J.: Macierzowa metoda minimalizacji funkcji Boole'a 5 i 6-zmiennych, Praca III KKA, Politechnika Śląska, Gliwice 1964, t. 2
- [3] BROMIRSKI J.: Teoria automatów, WNT, Warszawa 1969
- [4] BURKE R.E., VAN BOSSE J.G.: NAND-AND Circuits, IEEE Transactions on Electronic Computers, 1965, t. EC-14, nr 1
- [5] CALDWELL S.H.: Switching Circuits and Logical Design, J. Wiley, New York 1960
- [6] EICHELBERGER E.B.: Hazard Detection in Combinational and Sequential Switching Circuits, IMB JRD 1965, t. 9, nr 2
- [7] FRĄCKOWIAK J.: Metoda rozkładu tablic Karnaugh'a jako metoda faktoryzacji. Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 15
- [8] FRĄCKOWIAK J.: Zastosowanie metody rozkładu tablic Karnaugh'a do faktoryzacji funkcji logicznych, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 16
- [9] FRĄCKOWIAK J.: Algorytm G^m generacji wyższych maksymalnych impli-centów dopuszczalnych dla minimalizacji sieci NOR, Prace V KKA, Sekcja 2, Teoria Automatów, Politechnika Gdańska, 1971
- [10] FRĄCKOWIAK J.: Projektowanie bezhazardowych układów logicznych z elementami NOR, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 19
- [11] GIMPEL J.F.: The Minimization of TANT Networks, IEEE Transactions on Electronic Computers, 1967, t. EC-16, nr 1
- [12] GŁUSZKOW W.M.: Synteza automatów cyfrowych, WNT, Warszawa 1968
- [13] HELLERMAN L.: A Catalog of Three - Variable or - Invert and AND-Invert Logical Circuits, IEEE Transactions on Electronic Computers, 1963, t. EC-12, nr 2
- [14] HŁAWIČKA J.: Essential Hazard Correction Without the Use of Delay Elements, IEEE Transactions on Computers, 1970, t. C-19, nr 3
- [15] HŁAWIČKA A.: Identyfikacja hazardu oraz jego eliminacja w kombinacyjnych układach logicznych zbudowanych z elementów NOR, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 19
- [16] HŁAWIČKA A.: Realizacja dowolnych funkcji n-zmiennych za pomocą kombinacyjnych bezhazardowych trójwarstwowych struktur logicznych o funktorach NOR, Prace VKKA, Politechnika Gdańska, 1971
- [17] HŁAWIČKA A.: O pewnej metodzie syntezy minimalnych bezhazardowych kombinacyjnych układów logicznych TANT, Prace Instytutu Maszyn Matematycznych, 1972, nr 3
- [18] HUFFMAN D.A.: The Design and Use of Hazard-free Switching Networks, Journal of the Association for Computing Machinery, 1957, t. 4

- [19] JACZEWSKI J.: Układy logiczne dla zastosowań przemysłowych, PWN, Warszawa 1970
- [20] JAKUBAJTIS E.A.: Sintez asinchronnych konečných avtomatov, Izdatelstvo Zinatne, Ryga 1970
- [21] KACZANOWSKI S., OLSZEWSKI M., WAŃSKI Z.: Płynowe elementy i układy logiczne, WKL, 1971
- [22] KAMIONKA-MIKUŁA H.: Przypadki nieszkodliwości hazardu statycznego w automatach sekwencyjnych Archiwum Automatyki i Telemekhaniki, 1972, t. XVII, z. 1
- [23] KARNAUGH H.: The Map Method for Synthesis of Combinational Logic Circuits, Trans. A.I.E.E. Part 1, Communications and Electronics, 1953, t. 72, nr 2
- [24] KINTNER P.M.: A Simple Method of Designing NOR Logic, Control Engineering, luty 1963
- [25] LANGDON G.G., JR.: Analysis of Asynchronous Circuits Under Different Delay Assumptions, IEEE Transactions on Computers, 1968, t. C-17, nr 12
- [26] LASCARIS C., MATSUKIS J.N.: Optimization of NOR (NAND) Switching Circuits for N-variables, Electronic Engineering, marzec 1968
- [27] LERNER S.B.: Hazard Correction in Asynchronous Sequential Circuits, IEEE Transactions on Electronic Computers, 1965, t. EC-14, nr 2
- [28] MALEY G.A., EARLE J.: The Logic Design of Transistor Digital Computers, Englewood Cliffs, Prentice-Hall, New York 1963
- [29] MAŁYSIAK H.: Synteza synchronicznych automatów sekwencyjnych z zastosowaniem przerzutników, rozprawa doktorska, Politechnika Śląska, Gliwice 1969
- [30] McCLUSKEY E.J., STONE H.: Hazard-free Combinational Switching Networks, Communications of the Association for Computing Machinery, 1960, t. 3, nr 7, s. 392
- [31] McCLUSKEY E.J.: Introduction to the Theory of Switching Circuits, McGraw-Hill, New York 1965
- [32] McCLUSKEY E.J.: Minimization of Boolean Functions, BSTJ 1956, t. 35, s. 1417 - 1444
- [33] McCLUSKEY E.J.: Transients in Combinational Logic Circuits, w pracy zbiorowej: Redundancy Techniques for Computing Systems, Spartan Books, Waszyngton 1962
- [34] McCLUSKEY E.J., BARTEE T.C.: A Survey of Switching Circuits Theory, McGraw-Hill New York 1962
- [35] McGHEE R.B.: Some Aids to the Detection of Hazards in Combinational Switching Circuits, IEEE Transactions on Computers, 1969, t. EC-18, nr 6
- [36] MEISEL W.S., KASHEF R.S.: Hazards in Asynchronous Sequential Circuits, IEEE Transactions on Computers, 1969, t. EC-18, nr 8
- [37] MERGLER H.: A Systematic Approach, The Key to Optimum Logic Design, Control Engineering, styczeń 1967

- [38] MICHALSKI R.S.: Minimalizacja funkcji logicznych za pomocą metody pokrycia, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 14
- [39] MICHALSKI R.S.: Metoda określenia wyrażenia minimalnego dwuwartościowej funkcji logicznej bezpośrednio na podstawie jej graficznego obrazu w tablicy Veitcha-Karnaugh, Prace IV KKA, Akademia Górniczo-Hutnicza Kraków, Sekcja Teorii Automatów, 1967, t. 2
- [40] MILLER R.E.: Teorija pereključatelnych schem, Izdatelstvo - Nauka - Moskwa 1971, t. II
- [41] MYSTKOWSKI A.: Hazard stopni wyższych w układach sterowania przekąźnikowego, Pomiar, Automatyka, Kontrola 1966, nr 3
- [42] MYSTKOWSKI A.: Projektowanie układów sterowania przekąźnikowego obrabiarek, WNT, Warszawa 1970, s. 184 - 192
- [43] NICHOLS J.: Transistor-Transistor Micrologic Integrated Circuits, Fairchild Application Bulletin, luty 1967, APP - 131
- [44] PACH A.: Elementy rzeczywiste i idealne w układach przekąźnikowych, Prace III Krajowej Konferencji Automatyki, Gliwice 1964, t. 2
- [45] PACH A.: Zagadnienia niezawodności pracy układów przekąźnikowych, Budown. Okręt. 1966, nr 8, s. 244 - 248
- [46] PACH A.: Możliwość eliminowania dynamicznej zawodności strukturalnej przy pomocy rozszerzonej macierzy stanów, Zeszyty Naukowe Politechniki Śląskiej, Automatyka z. 14
- [47] PACH A.: Analiza zawodności strukturalnej stykowych układów przełączających, Praca doktorska, Akademia Górniczo-Hutnicza, Kraków 1964
- [48] PACH A.: Niezawodność pracy a zagadnienie minimalizacji układów przekąźnikowych, Archiwum Automatyki i Telemekhaniki, 1967, t. XII, z. 1
- [49] PACH A.: Pewna modyfikacja macierzy M. Karnaugh, Prace IV KKA, Akademia Górniczo-Hutnicza, Sekcja Teorii Automatów, Kraków 1967, t. 2
- [50] POCHOPIEŃ B.: Metoda syntezy automatów sekwencyjnych z zastosowaniem przerzutników, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, Gliwice 1972, z. 19
- [51] RICHARDS R.K.: 6 More Ways to Make Logic Circuits, Control Engineering, 1967, nr 1
- [52] ROGIŃSKI W.N.: Rabota relejnych schem v perechodnom periode, Avtomat, Tielemech. 1959, nr 10
- [53] ROGIŃSKI W.N.: Metody syntezy przekąźnikowych układów sterowania, WNT Warszawa 1967
- [54] SAPIECHA K.: Analiza sieci kombinacyjnych z ograniczonymi opóźnieniami szkodliwymi, Archiwum Automatyki i Telemekhaniki, 1972, t. XVII, z. 3
- [55] SIEMENS, Halbleiter Schaltbeispiele - Integrierte Schaltungen, 1972
- [56] SIWIŃSKI J., MAŁYSIAK H., NIEDERLIŃSKI, WAGNER F.: Prace Katedry Automatyki Procesów Przemysłowych w zakresie teorii automatów, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 14

- [57] SIWIŃSKI J.: Proektirovanie mnogotaktnych avtomatov s ispolzovaniem tablic posledovatel'nosti vklucenij, Materiały Międzynarodowej Konferencji "Automation and Control Systems - ARS69", Morawska Ostrawa, kwiecień 1969
- [58] SIWIŃSKI J.: Układy przełączające w automatyce, WNT, Warszawa 1968
- [59] SIWIŃSKI J.: Układy przekaźnikowe w automatyce, WNT, Warszawa 1964
- [60] SMITH R.A.: Minimal Three-Variable NOR and NAND Logic Circuits, IEEE Transactions on Electronic Computers, 1965, t. EC-14, nr 1
- [61] SORKING R.B.: Integrated Electronics, McGraw-Hill, Book Company, 1970
- [62] SPEGEL M.: A Method for a Low-Cost Realization of Logical Functions. Vth Yugoslav International Symposium on Information Processing Bled, 1969
- [63] Synchronism, Asynchronism, Hazards and Races, rozdział z pracy Further Sequential Theory
- [64] TRACZYK W.: Projektowanie tranzystorowych układów przełączających, WNT, Warszawa 1966
- [65] Transients and Hazards in Switching Circuits
- [66] UNGER S.H.: Hazards and Delays in Asynchronous Sequential Switching Circuits, IRE Transactions on Circuit Theory, 1969, t. CT-6
- [67] UNGER S.H.: A Row Assignment for Delay-Free Realizations of Flow Tables Without Essential Hazards, IEEE Transactions on Computers, 1968, t. C-17, nr 2
- [68] WAGNER F.: Minimalizacja pewnej klasy funkcji logicznych za pomocą zmodyfikowanych tablic Karnauga, Zeszyty Naukowe Politechniki Śląskiej, Automatyka, z. 15
- [69] WALIGÓRSKI S.: Układy przełączające, elementy teorii i projektowania, WNT, Warszawa 1971
- [70] WAWIŁOW E.N., PORTNOJ G.P.: Synteza układów elektronicznych maszyn cyfrowych, WNT, Warszawa 1967
- [71] WIECHERT K.: Zum Entwurf von logischen Schaltungen mit NOR Schaltkreisen, Messen - Steuern - Regeln, 1969, nr 3
- [72] WIEWIÓRKOWSKI T.: Aktualne problemy syntezy kombinacyjnych układów logicznych zbudowanych z elementów NAND/NOR/, Sympozjum CO PAN, 1971
- [73] WOOD P.E.: Hazards in Pulse Sequential Circuits, IEEE Transactions on Electronic Computers, 1964, t. EC-13
- [74] YOELI M., RINON S.: Application of Ternary Algebra to the Study of Static Hazards, Journ. ACM, 1964, t. 11
- [75] ZASADA W., KIBNER G.: Zastosowanie EMC do analizy dowolnego kombinacyjnego układu przełączającego oraz identyfikacji k-krotnego hazardu podstawowego przy pomocy algorytmu I^k, oprac. wew. IMM O/Śl.
- [76] ZISSOS D., COPPERWHITE G.W.: The Design of Minimal NOR/NAND Logical Circuits, Electronic Engineering 37, wrzesień 1965

- [77] ZISSOS D., COPPERWHIDE G.W.: Further Developments in the Design of Minimal NOR and NAND Switching Circuits for N-Variables, Electronic Engineering 38, lipiec 1966
- [78] ZISSOS D., A List of Minimal NOR/NOR and NAND/NAND Combination Logical Circuits, A.T.E.J. 20, 1964, nr 2
- [79] MIL-STD-806B norma amerykańska z 26.2.1962

D O D A T E K

Pragniemy zwrócić uwagę na fakt, że rozprawa doktorska przedstawiona w tym zeszycie została napisana w roku 1972. To, że dociera dziś do czytelnika, związane było między innymi z długim cyklem wydawniczym. Zawarta w rozprawie bibliografia obejmuje w związku z powyższym stan wiedzy w zakresie poruszonych w rozprawie problemów tylko do pierwszej połowy roku 1972.

Zagadnienia identyfikacji hazardów i ich eliminacji w układach przełączających były w ostatnich trzech latach rozpatrywane przez kilku autorów zarówno krajowych jak i zagranicznych. Dostępną w kraju literaturę tych autorów z lat od 1972 do 1975 omawiającą te zagadnienia, przedstawia wykaz bibliograficzny na końcu tego dodatku. Część tej literatury podaje nowe spojrzenie i nieco inne propozycje rozwiązań problemów poruszonych i rozwiązanych w drugim, trzecim i czwartym rozdziale przedstawionej rozprawy doktorskiej. Najbardziej interesujące - zdaniem autora - rezultaty osiągnięte w tych pracach w zakresie problematyki poruszonej w ww rozdziałach, dotyczą analizy skutków hazardu występującego w asynchronicznych automatach sekwencyjnych. Szczegółową analizę tych skutków przedstawiono w pracach [13], [14], [15] i [16]. W pracach tych podano także wiele przypadków dla różnych typów struktur automatów sekwencyjnych, w których zjawisko hazardu jest nieszkodliwe.

Postać realizowanej funkcji decyduje o możliwości wystąpienia w układzie zjawiska hazardu danego typu. W układach przełączających realizowanych w postaci alternatywnej może wystąpić zjawisko hazardu statycznego w warunkach działania, ale nie może wystąpić zjawisko hazardu statycznego w warunkach niedziałania, ani zjawisko hazardu dynamicznego. Natomiast w układach realizowanych w postaci konjunkcyjnej może wystąpić zjawisko hazardu statycznego w warunkach niedziałania, ale nie może wystąpić zjawisko hazardu statycznego w warunkach działania, ani zjawisko hazardu dynamicznego.

Właściwości te zostały wykorzystane w pracy [13] do uproszczenia tych układów, w których zjawisko hazardu danego typu nie

jest szkodliwe dla prawidłowej pracy całego układu. W pewnych przypadkach chwilowy zanik "jedynki" dla sygnału sterującego układem pamięci nie zostaje przez ten układ zapamiętany. W pracy [13] stwierdzono, że występuje to dla tych układów, dla których spełniony jest warunek, że w każdej chwili najwyżej jeden z sygnałów sterujących układem pamięci jest w stanie logicznym "1". Dzięki temu można znacznie uprościć te układy sekwencyjne, które wykorzystują przerzutniki RS i inne spełniające podany wyżej warunek, bowiem alternatywna postać części kombinacyjnej tych układów nie musi zawierać implikantów anty hazardowych.

Dokładną analizę skutków chwilowych przekłamań występujących na wejściach przerzutników różnych typów oraz analizę możliwych bezpośrednich przejść w elementach pamięci przedstawiono w pracach [14] i [16]. Opierając się na rezultatach tych analiz i fakcie, że dwupoziomowe układy przełączające są zawsze wolne od jednego z rodzajów logicznego hazardu statycznego, w pracach [14] i [16] podano wytyczne do syntezy minimalnych asynchronicznych układów dwupoziomowych: NAND, NOR, I-LUB, LUB-I sterujących przerzutnikami. Zastosowanie tych wytycznych w algorytmach minimalizacji funkcji logicznych jest bardzo proste i czasem sprowadza się do znanych algorytmów minimalizacji.

Prace [14] i [16] podają też sposób wykorzystania do dodatkowego uproszczenia układów informacji o ograniczeniach zmian wejść.

Przedstawiony w czwartym rozdziale niniejszej pracy algorytm G^{x_1} umożliwia przeprowadzenie syntezy minimalnego układu spośród wszystkich układów TANT pozbawionych hazardu podstawowego. Podobny algorytm, rozważany jednak z punktu widzenia tylko hazardu statycznego, przedstawiony jest w pracach [7] i [8]. Natomiast w pracach [15] i [16] udowodniono, że algorytm G^{x_1} pozwala otrzymywać minimalne układy TANT pozbawione logicznego hazardu p-zmiennych. Gdyby nie korzystać z wyników wspomnianej wyżej analizy skutków hazardu występującego w asynchronicznych automatach sekwencyjnych, synteze minimalnych układów TANT sterujących przerzutnikami należałoby przeprowadzać za pomocą algorytmu G^{x_1} , niezależnie od tego jakim przerzutnikiem i którym

jego wejściem steruje układ. Korzystając z przedstawionych w pracach [15] i [16] lematów i twierdzeń minimalizowany układ TANT można uwolnić od zbędnych obwodów anty hazardowych za pomocą:

- uwolnienia wyrażenia TANT od niektórych zbędnych implikantów
- dopuszczenia wszystkich lub tylko niektórych rozszerzeń czynników końca.

W celu praktycznego wykorzystania tych wniosków w pracy [15] i [16] przedstawiono różne modyfikacje algorytmu Gimpela, zależnie od tego, od jakiego rodzaju hazardu należy uwolnić minimalizowany układ TANT sterujący przerzutnikami.

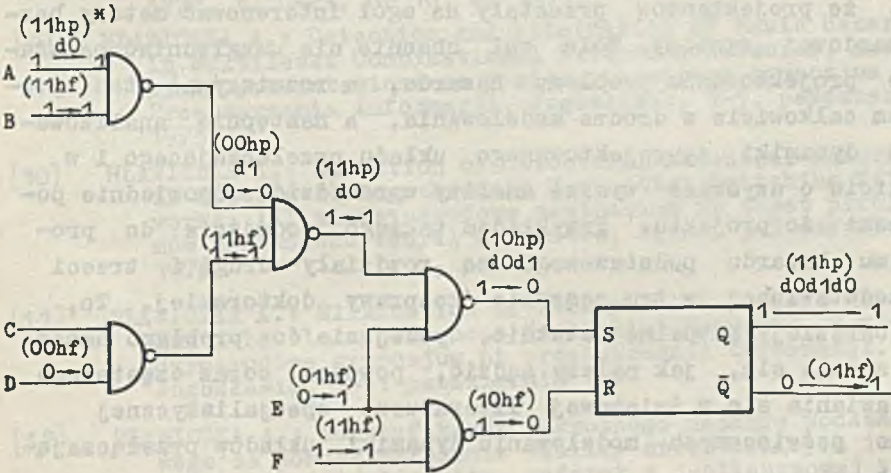
Nową metodę identyfikacji hazardu istotnego przedstawiono w pracy [10]. Zdefiniowano w niej pojęcie strukturalnego hazardu istotnego i przedstawiono prostą metodę identyfikacji tego hazardu w wielowarstwowych asynchronicznych sekwencyjnych układach przełączających. Do identyfikacji tego hazardu proponuje się w tej pracy wykorzystywać algorytm I^k realizowany bądź za pomocą składania siatek Karnaugh'a bądź za pomocą składania tablicy strukturalnych implikantów i implicentów.

W pracach [9], [11] i [12] powtórzono z pewnymi poprawkami i w innym ujęciu rezultaty autora przedstawione w 2 i 3 rozdziale niniejszej pracy.

Problematyka identyfikacji i eliminacji w układach kombinacyjnych hazardów logicznych p-zmiennych i hazardów funkcyjnych bardzo szczegółowo omówiona została w pracach [1], [2], [3], [4], [5], [6], [7], [17] i [18]. Ponieważ problematyka ta nie była rozpatrywana przez autora w żadnym z rozdziałów niniejszej pracy, dlatego zasygnalizowano tutaj tylko istnienie ww literatury celem zwrócenia na nią uwagi zainteresowanych czytelników.

Bardzo interesujące rezultaty dotyczące identyfikacji w układach kombinacyjnych i sekwencyjnych hazardów logicznych p-zmiennych uzyskał Breuer w pracy [19].

Należy zwrócić uwagę na fakt, że analiza hazardów zaprezentowana w tej pracy w wielu szczegółach została wcześniej, aczkolwiek w innej formie, opisana przez autora w pracach [9] i [11], a przede wszystkim w rozdziale drugim rozprawy doktorskiej przedstawionej w tym zeszycie. Na przykład dyskusja przeprowadzona w lematach 2.3.1. ÷ 2.3.7. dotycząca sposobu tworzenia sekwencji zmian oraz sposobu ich przesyłania przez bramki I, LUB, NIE, NAND i NOR jest bardzo podobna do wyników rozważań Breuera dotyczących tworzenia i przesyłania przez ww bramki hazardów logicznych p-zmiennych. Niektóre rezultaty tej dyskusji, a w szczególności łatwość identyfikacji hazardów dynamicznych są identyczne w obu przypadkach. Ilustruje to wyraźnie poniższy rysunek.



* wg metody Breuera

Również dyskusje dotyczące sposobu tworzenia warunków bezhazardowości i przesyłania ich przez bramki wielowarstwowego sekwencyjnego układu przełączającego z przerzutnikami RS przedstawione w pracy [19] i rozprawie doktorskiej są do siebie podobne.

W pracach poświęconych eliminacji hazardów wywołanych równoczesną zmianą stanów p-zmiennych wejściowych udowodniono, że istnieją funkcje, które nie posiadają swoich układowych realizacji pozbawionych hazardów logicznych i hazardów funkcyjnych. Eichelberger udowodnił to w zakresie hazardów funkcyjnych. Natomiast Frąckowiak w pracy [7] przedstawił algorytm, który albo znajduje minimalną, pozbawioną hazardu logicznego, dwuwarstwową realizację funkcji przełączającej f , albo udowadnia, że funkcja f nie ma realizacji bezhazardowej. Można więc w chwili obecnej stwierdzić, że projektanci układów przełączających nie dysponują metodami syntezy umożliwiającymi tworzenie układów przełączających pozbawionych całkowicie hazardów. W związku z powyższym projektanci nie zawsze zdają sobie sprawę, czy zaprojektowane przez nich układy przełączające są bezhazardowe, czy też nie. Spowodowało to, że projektantów przestały na ogół interesować metody bezhazardowej syntezy. Wolą oni obecnie nie uwzględniać na etapie projektowania problemu hazardu, a rozwiązywać ten problem całkowicie w drodze modelowania, a następnie analizowania dynamiki zaprojektowanego układu przełączającego i w oparciu o uzyskane wyniki analizy wprowadzić odpowiednie poprawki do projektu. Przykładem takiego podejścia do problemu hazardu podstawowego są rozdziały drugi i trzeci przedstawionej w tym zeszycie rozprawy doktorskiej. To, najbardziej aktualnie ostatnio, podejście do problemu hazardu stanie się, jak należy sądzić, powodem coraz częstszego pojawiania się w światowej literaturze specjalistycznej prac poświęconych modelowaniu dynamiki układów przełączających.

Bibliografia z lat 1972-1975

- [1] BEISTER J.: Beiträge zur Theorie der Hasards in Schalt-
netzen, praca doktorska, Karlsruhe, RFN, 1972
- [2] BEISTER J.: Die Richtungsfunktion zur Behandlung stati-
scher Hasards, materiały przedstawione na seminarium
w Karlsruhe Univ., Karlsruhe, RFN, maj 1972
- [3] BEISTER J.: A Unified Approach to Combination Hazards,
IEEE Transactions on Computers, czerwiec 1974,
t. C-23, nr 6, s. 566
- [4] BEISTER J.: Ein Monotoniekriterium für statische und
dynamische Hazards, materiały seminaryjne Instytutu
Teorii Automatów i Sieci Przełączających, Stowarzy-
szenie matematyków i informatyków, sprawozdanie nr 39,
Bonn, 1972
- [5] BREDESON J.G., HULINA P.T.: Elimination of Static and
Dynamic Hazards for Multiple Input Changes in Com-
binational Switching Circuits, Inform. Contr., 1972,
t. 20, s. 114-124
- [6] FANTAUZZI G.: An Algebraic Model for the Analysis of
Logical Circuits, IEEE Transactions on Computers,
czerwiec 1974, t. C-23, nr 6, s. 576
- [7] FRĄCKOWIAK J.: Methoden der Analyse und Synthese von
hasardarmen Schaltnetzen mit minimalen Kosten, Ha-
bilitationsschrift, Karlsruhe, RFN, czerwiec 1973
- [8] FRĄCKOWIAK J.: The Synthesis of Minimal Hazardless TANT
Networks, IEEE Transactions on Computers, październik
1972, t. C-21, nr 10, s. 1099
- [9] HŁAWICZKA A.: Detection and Elimination of Basic Hazards
in Multilevel Combinational Switching Networks, mate-
riały przedstawione na 8 międzynarodowym symposium nt.
Przetwarzania Informacji, Jugosławia, 1-5, październik
1973
- [10] HŁAWICZKA A.: Detection of Structural-Essential-Hazards
in Multilevel Asynchronous Sequential Switching Net-
works, III międzynarodowe seminarium na temat proble-
mów zastosowań teorii automatów, Warna, 3-7 czerwiec
1975
- [11] HŁAWICZKA A.: Elimination of Steady-State-Hazards in
Multi-Level Asynchronous Sequential Networks, 8 mię-
dzynarodowe symposium nt przetwarzania informacji,
Jugosławia, 7-11 październik 1974
- [12] HŁAWICZKA A.: Identyfikacja k-krotnego hazardu podstawo-
wego za pomocą składania tablicy strukturalnych im-
plikantów i implimentów, referat z jubileuszowej kon-
ferencji naukowej Wydziału Automatyki i Informatyki
pn. "Współczesne Problemy Automatyki i Informatyki,
Sekcja B", Gliwice, 18-20 wrzesień 1973

- [13] KAMIONKA-MIKUŁA H.: Przypadki nieszkodliwości hazardu statycznego w automatach sekwencyjnych, Archiwum Automatyki i Telemechaniki, PWN Warszawa 1972, t. XVII, z. 1, s. 21
- [14] KAMIONKA-MIKUŁA H.: Synteza dwupoziomowych układów minimalnych sterujących przerzutnikami, Archiwum Automatyki i Telemechaniki, PWN Warszawa 1974, t. XIX, z. 1, s. 73
- [15] KAMIONKA-MIKUŁA H.: Synteza bezhazardowych układów minimalnych TANT sterujących przerzutnikami, Archiwum Automatyki i Telemechaniki, PWN Warszawa 1974, t. XIX, z. 2, s. 183
- [16] KAMIONKA-MIKUŁA H.: Analiza przypadków nieszkodliwości hazardu w automatach sekwencyjnych, praca doktorska, Politechnika Śląska, 1974
- [17] SAPIECHA K.: Analiza sieci kombinacyjnych z nieograniczonym opóźnieniem bramek, Archiwum Automatyki i Telemechaniki, PWN Warszawa 1974, t. XIX, z. 1, s. 63
- [18] SAPIECHA K.: Eliminacja M ryzyka w realizacjach funkcji boolowskich, Archiwum Automatyki i Telemechaniki, PWN Warszawa 1974, t. XIX, z. 2, s. 199
- [19] BRUER M.A.: Procedures for Eliminating Static and Dynamic Hazards in Test Generation, IEEE Transactions on Computers, październik 1974, t. C-23, nr 10, s. 1069

ИДЕНТИФИКАЦИЯ И ЭЛИМИНАЦИЯ ОСНОВНОГО РИСКА В МНОГОСЛОЙНЫХ ПЕРЕКЛЮЧАЮЩИХ СХЕМАХ

Резюме

Указан метод идентификации k -кратного основного риска в многослойных комбинационных переключающих схемах, реализующих функции n -переменных и построенных на базе элементов И, ИЛИ, НЕТ, И-НЕТ, ИЛИ-НЕТ. На основе введенных дефиниции, а также на основе доказанных лемм и теорем, был построен алгоритм I^k . Подано семь примеров использования алгоритма I^k для идентификации k -кратного основного риска при помощи составления таблицы структурных импликантов и имплицентов, а также при помощи составления диаграмм Карно. Подан способ элиминации любых основных k -кратных рисков на выходе любого логического элемента многослойной переключающей схемы. Указаны два метода синтеза минимальных безрисковых схем TANT реализующих любые функции n -переменных.

BASIC HAZARD DEDECTION AND ELIMINATION IN MULTI-LEVEL SWITCHING NETWORKS

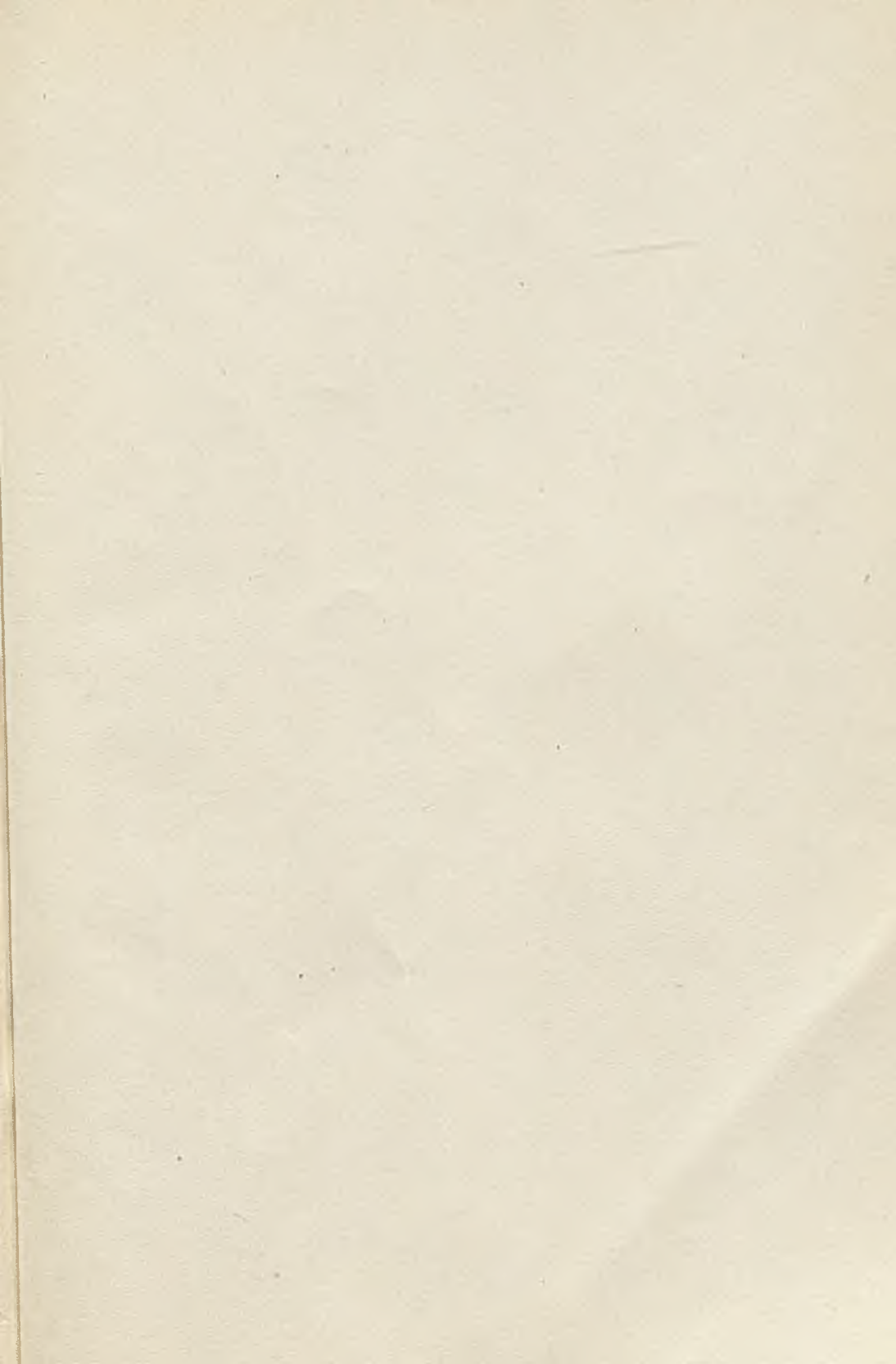
Summary

Presented a method of k basic hazard dedection in multi-levels combinational switching networks realizing n -variable function built on elements AND, OR, NOT, NAND and NOR. On the basis of the introduced definitions, proved lemmas and theorems the I^k algorithm was built. Presented 7 examples of I^k algorithm use for k basic hazard dedection by means of composing table of structural implicants and implicates and, by means of composing Karnaugh maps. Given the ways of any basic k hazard elimination on the output of any logic element of the multi-level switching networks. Two methods of synthesis of minimal hazard-free TANT networks realizing any n -variable function are presented.

INFORMACJE O AUTORZE

Andrzej Hławiczka dr inż. Ukończył studia w 1965 r. na Wydziale Elektrycznym Politechniki Śląskiej. W roku 1968 ukończył dwuletnie podyplomowe studia automatyki i telemekhaniki na Wydziale Automatyki Politechniki Śląskiej. W roku 1973 obronił pracę doktorską na temat "Identyfikacja i eliminacja hazardu podstawowego w wielowarstwowych układach przełączających". Od roku 1965 do roku 1973 zajmował się projektowaniem i wdrażaniem cyfrowych układów sterowania jako starszy konstruktor, a następnie jako starszy projektant. W tym czasie opublikował wiele prac z zakresu zaprojektowanych przez siebie układów cyfrowego sterowania oraz zgłosił do Urzędu Patentowego kilka patentów i wzorów użytkowych.

Od 1971 roku pracuje na stanowisku adiunkta i kieruje Zakładem Konstrukcji Cyfrowych w Śląskim Oddziale Instytutu Maszyn Matematycznych. Dodatkowo zajmuje się problemami teorii automatów, a szczególnie zawodnością strukturalną bezstykowych układów przełączających. Opublikował z tej dziedziny cztery prace.



Cena 60 zł