

Zespół Szkół Elektronicznych
w Lublinie

QMC 95 F
minimalizacja na bazie
metody Quine'a McCluskeya

Praca dyplomowa
nr 5D/03/99

Autor

Konrad Foks, 5d^T

Konsultant

mgr inż. Marian Matuszyk

Lublin, 1998/99

SPIS TREŚCI

SPIS TREŚCI.....	2
<u>1. WSTĘP.....</u>	<u>3</u>
<u>2. MINIMALIZACJA FUNKCJI LOGICZNYCH METODĄ QUINE’A MCCLUSKEYA.....</u>	<u>4</u>
PRZYKŁAD MINIMALIZACJI FUNKCJI Z TRZEMA POSTACIAMI MINIMALNYMI.....	6
<u>3. OPIS PROGRAMU QMC DLA WINDOWS.....</u>	<u>7</u>
INFORMACJE OGÓLNE.....	7
CHARAKTERYSTYKA PROGRAMU.....	8
STRUKTURA PROGRAMU.....	8
ZASADY OBSŁUGI PROGRAMU QMC DLA WINDOWS.....	8
<u>4. ZAŁOŻENIA DYDAKTYCZNE PROGRAMU.....</u>	<u>10</u>
<u>4. LITERATURA.....</u>	<u>11</u>

1. Wstęp

Zasady algebry Boole'a

Algebra Boole'a określa działania na dwuelementowym zbiorze {prawda, fałsz} odpowiadającemu zbiorowi $\{0,1\}$, na którym określono podstawowe działania: sumy, iloczynu i negacji oraz podstawowe prawa: przemienności, łączności, rozdzielności i De Morgana. Zmienna logiczna (nazywana również zmienną boolowską) przyjmuje więc wartości z tego zbioru : zero albo jeden. Argumenty funkcji odpowiadają stanom wejściowym układu, natomiast wartość funkcji stanowi wyjścia.

Sposoby zapisu funkcji logicznych

	Przykład
Kanoniczna postać sumy	$F = \overline{A}BC + A\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + \overline{A}\overline{B}C$
Kanoniczna postać iloczynu	$F = (A + B + C)(\overline{A} + B + \overline{C})(\overline{A} + \overline{B} + C)$
Liczbowa	$\Sigma = (0,3,4,5,6)_{ABC}$

Ponadto funkcję logiczną można zadawać w postaci opisu słownego, tablicy wartości funkcji (tabeli prawdy) lub wykresu czasowego.

Podstawowe funkcje logiczne

		negacja	suma	iloczyn		
		NIE	LUB	I		
		NOT	OR	AND	EXOR	EXNOR
A	B	$\sim A$	$A+B$	$A*B$	$A \oplus B$	$A \otimes B$
0	0	1	0	0	0	1
0	1	1	1	0	1	0
1	0	0	1	0	1	0
1	1	0	1	1	0	1

Podstawowe prawa algebry Boole'a

Dotyczące sumy	Dotyczące iloczynu
----------------	--------------------

$A + 0 = A$	$A * 1 = A$
$A + \bar{A} = 1$	$A * \bar{A} = 0$
$A + A = A$	$A * A = A$
$A + 1 = 1$	$A * 0 = 0$
$A + B = B + A$	$A * B = B * A$
$A + (B + C) = (A + B) + C$	$A * (B * C) = (A * B) * C$
$\overline{A + B + \dots} = \bar{A} * \bar{B} * \dots$	$A + B * C = (A + B) * (A + C)$
$A + A * B = A$	$\overline{A * B * \dots} = \bar{A} + \bar{B} + \dots$
$\overline{\overline{A}} = A$	$A * (A + B) = A$

2. Minimalizacja funkcji logicznych metodą Quine'a McCluskeya

Zasady minimalizacji

Minimalizacja funkcji logicznych metodą Quine'a-McCluskeya jest metodą systematyczną, wykorzystującą w zalgorytmizowany sposób operację grupowania i obróbki danych, co pozwala na w miarę proste wykorzystanie jej w minimalizujących programach komputerowych. Najczęściej stosuje się ją przy dużej ilości zmiennych, gdzie inne metody (algebraiczna, tablic Karnaugh'a) stają się uciążliwe.

Postać kanoniczna funkcji

W pierwszym kroku należy otrzymać postać kanoniczną funkcji, która będzie zawierała wszystkie składniki jedynek.

Znalezienie prostych implikantów

Tworzymy tabelę zawierającą wszystkie składniki jedynek w postaci zerojedynekowej. Następnie porównujemy każde dwa elementy tej tablicy i sprawdzamy czy obie liczby różnią się tylko na jednej pozycji (np. 0110 i 0100 spełnia ten wymóg, na trzeciej pozycji występuje jedyna różnica między nimi), zaznaczamy oba implikanty, a do nowej tabeli wpisujemy 01-0 (kreska na pozycji występowania różnicy). Po sprawdzeniu pierwszej tabeli wszystkie niezaznaczone implikanty wpisujemy do tablicy prostych implikantów. Podobnie postępujemy z drugą tabelą, z tym że do warunku o różnicy 0,1 na tylko jednej pozycji, dochodzi warunek występowania kreski na dokładnie tych samych pozycjach. Tworząc w ten sposób kolejne tabele, wypełniając stopniowo tablicę prostych implikantów, dochodzimy do momentu w którym żaden z implikantów tabeli, po jej całkowitym porównaniu, nie będzie zaznaczony i wpisując je do tablicy prostych implikantów zakończymy ten etap pracy.

Wyróżnienie istotnych prostych implikantów

Tworzymy tablicę prostych implikantów – tablicę pokrycia, w której każdy wiersz odpowiada kolejnym prostym implikantom, natomiast kolumny składnikom postaci kanonicznej funkcji. Następnie wypełniamy tabelę krzyżykami, tam gdzie dany implikant zawiera w sobie dany składnik postaci kanonicznej.

Po wypełnieniu tabeli szukamy kolumn w których występuje tylko jeden krzyżyk tj. dany składnik postaci kanonicznej funkcji jest pokrywany tylko przez jeden prosty implikant. Ten prosty implikant, zwany implikantem istotnym zaznaczamy w dodatkowej skrajnej kolumnie.

Po odszukaniu wszystkich istotnych implikantów, łączymy je w tzw. jądro funkcji i usuwamy z tabeli pokrycia wszystkie pokrywane przez nie proste implikanty.

Określenie form minimalnych

Z pozostałych prostych implikantów musimy wybrać minimalną ich ilość, która pokryje wszystkie składniki postaci kanonicznej funkcji. Często możemy tego dokonać na wiele sposobów, a więc funkcja ma wiele postaci minimalnych. Ma to duże znaczenie przy minimalizacji funkcji wielowyjściowych, gdzie wykorzystuje się równorzędne składniki poszczególnych funkcji.

Przykład minimalizacji funkcji z trzema postaciami minimalnymi

Dana jest funkcja $F = AB + \overline{A}\overline{B}\overline{C} + ACD + \overline{A}\overline{B}CD + \overline{A}\overline{B}\overline{C}D$

którą poddajemy minimalizacji metodą Quine'a – McCluskeya

Postać kanoniczna funkcji

$$F = \overline{A}\overline{B}\overline{C}\overline{D} + \overline{A}\overline{B}C\overline{D} + \overline{A}B\overline{C}\overline{D} + \overline{A}BC\overline{D} + \overline{A}\overline{B}\overline{C}D + \overline{A}\overline{B}CD + \overline{A}B\overline{C}D + \overline{A}BCD + ABC\overline{D} + ABCD$$

Znalezienie prostych implikantów

Przekształcamy poszczególne składniki postaci kanonicznej na ich odpowiedniki zapisane w systemie binarnym.

$\overline{A}\overline{B}\overline{C}\overline{D}$	1100	12
$\overline{A}\overline{B}C\overline{D}$	1101	13
$\overline{A}B\overline{C}\overline{D}$	1110	14
$\overline{A}BC\overline{D}$	1111	15
$\overline{A}\overline{B}\overline{C}D$	0100	4
$\overline{A}\overline{B}CD$	0101	5
$\overline{A}B\overline{C}D$	0011	3
$\overline{A}BCD$	0001	1
$AB\overline{C}\overline{D}$	1011	11

Następnie sortujemy elementy ze względu na ilość jedynek w postaci binarnej co ma ułatwić porównywanie elementów, które dokonujemy teraz pomiędzy składnikami sąsiednich grup

		1. porównanie	2. porównanie	Proste implikanty
	$ABCD$	$ABCD$	$ABCD$	$ABCD$
$\overline{A}\overline{B}\overline{C}\overline{D}$	0100v	-100v	-10-	-10-
$\overline{A}\overline{B}C\overline{D}$	0001v	00-1	11--	11--
$\overline{A}B\overline{C}\overline{D}$	1100v	0-01	11--	-011
$\overline{A}BC\overline{D}$	0011v	110-v		1-11
$\overline{A}\overline{B}CD$	0101v	11-0v		00-1
$\overline{A}B\overline{C}D$	1101v	-011		0-01
$\overline{A}BCD$	1011v	-101v		
$AB\overline{C}\overline{D}$	1110v	11-1v		
$ABCD$	1111v	1-11		
		111-v		

Tablice Karnaugh'a odpowiadające kolejnym etapom minimalizacji

Po pierwszym porównaniu

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	1	1	0	0
	11	1	1	1	1
	10	0	0	1	0

Proste implikanty

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	1	1	0	0
	11	1	1	1	1
	10	0	0	1	0

Implikanty istotne

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	1	1	0	0
	11	1	1	1	1
	10	0	0	1	0

Po drugim porównaniu

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	1	1	0	0
	11	1	1	1	1
	10	0	0	1	0

Implikanty pozostałe

		CD			
		00	01	11	10
AB	00	0	1	1	0
	01	1	1	0	0
	11	1	1	1	1
	10	0	0	1	0

3. Opis programu QMC dla Windows

Informacje ogólne

Program QMC dla Windows minimalizuje wieloargumentowe funkcje algebry boole'a wykorzystując algorytm Quine'a – McCluskeya. Liczba argumentów jest ograniczona ilością liter w alfabecie łacińskim

Charakterystyka programu

Przedmiot minimalizacji – wieloargumentowe funkcje algebry Boole’a

Sposób minimalizacji – metoda Quine’a – McCluskeya

Litery kodujące argumenty funkcji – 26 liter alfabetu łacińskiego (A – Z)

Maksymalna liczba argumentów – 26

Minimalne wymagania sprzętowe - PC 486 system Windows 95/98/NT

Optymalny sprzęt – PC Pentium

Sterownie – myszką lub klawiaturą przy pomocy systemu skrótów

Struktura programu

Rozdział ten przedstawia ogólne mechanizmy pracy QMC od strony programowej.

Program został napisany w całości w Pascalu – kompilator Delphi 2 z wykorzystaniem wbudowanego assemblera. QMC dla Windows jest programem 32 bitowym i może pracować tylko pod kontrolą 32 bitowego systemu operacyjnego jakim jest Windows 95/98/NT.

Porównanie:	QMC	QMC dla Windows
Ilość bajtów zajmowanych przez jeden składnik jedynek	1 – 26 bajtów w zależności od ilości argumentów	8 bajtów bez względu na ilość argumentów
Procedury porównywania	Operacje na ciągach znaków	Operacje bitowe przy wykorzystaniu assemblera
Maksymalna ilość składników jedynek	205	2000
Czas minimalizacji		Do 10 razy krótszy
Wymagania sprzętowe	niskie	Wysokie, jak dla systemu Windows 95
Współpraca z innymi programami	brak	poprzez OLE

Możliwości zmian w programie

Minimalizacja funkcji wielowyjściowych

Zastosowanie tablic dynamicznych dostępnych w Delphi 4

Tablica pokrycia złożona z pojedynczych bitów symulowana na zmiennych wielobitowych

Zasady obsługi programu QMC dla Windows

Wprowadzanie funkcji i zasady edycji

Po uruchomieniu program jest gotowy do wprowadzania nowej funkcji przy pomocy klawiszy A-Z dla postaci literowej funkcji lub 0-9 dla dziesiętnej. Program sam rozpoznaje postać funkcji i „zadba” o jej właściwy wygląd (znak wartości funkcji, znaki odstępu, znaki niedozwolone) i

kodowanie. Kolejność liter kodujących można ustawić wciskając wyróżniony prostokąt zawierający te litery (obok przycisku Minimalizacja). Chcąc uzyskać zanegowany argument należy wcisnąć odpowiednią literę wraz z klawiszem SHIFT przy czym położenie przełącznika Caps Lock nie wpływa na działanie tej funkcji. Znaki między kolejnymi składnikami jedynki najwygodniej wprowadzać za pomocą klawisza spacji, lub też za pomocą właściwych znaków $<+>$ lub $<,>$.

Widok głównego okna programu

Odczyt i zapis funkcji na dysku

Funkcje możemy zapisywać i odczytywać z dysku przy pomocy poleceń menu Funkcja / Otwórz, Zapisz jako.

Plik programu QMC zawiera funkcję wejściową w postaci identycznej z występującą w raporcie (małe litery jako argumenty zanegowane). Plik ten możemy edytować dowolnym edytorem tekstowym, jednak zmiany mogą być powodem błędnego odczytania pliku przez program QMC.

Rozpoczęcie minimalizacji

Rozpoczęcie minimalizacji następuje po:

- ☐ wciśnięciu przycisku MINIMALIZACJA
- ☐ wybraniu odpowiedniego polecenia z menu Funkcja/Minimalizuj
- ☐ wciśnięciu skrótu klawiaturowego Ctrl+M.

Następnie na ekranie pojawi się okno podające na bieżąco kolejne kroki minimalizacji i dotychczasowy czas jej trwania. Po wciśnięciu przycisku POKAŻ SZCZEGÓŁY możemy zapoznać się z ilością elementów i porównać między nimi kolejnych tablic występujących w minimalizacji metodą Quine'a McCluskeya. Jeżeli szczegóły będą uwidocznione okno pozostanie widoczne po skończeniu minimalizacji. W przeciwnym wypadku zniknie automatycznie. Minimalizację można w każdej chwili przerwać przyciskiem Przerwij.

Okno pojawiające się podczas minimalizacji

Przeglądanie wyników minimalizacji

Bezpośrednio po skończeniu minimalizacji są dostępne następujące dane

- Postać kanoniczna funkcji wejściowej
- Postać dziesiętna tej funkcji
- Proste implikanty występujące w tej funkcji
- Minimalną postać (jedną lub wiele)
- Ilość liter i grup funkcji wejściowej i jej postaci minimalnej

Zawartość każdego okna możemy przesuwając trzymając wciśnięty lewy klawisz myszy. Ponadto wszystkie są dostępne w raporcie

Raport

O ile okno programu służy do bezpośredniego przeglądania wyników minimalizacji, to ten sposób nie jest wygodny przy dłuższych funkcjach. W raporcie możemy wygodnie przeglądać dłuższe funkcje a także kopiować za pomocą schowka wyniki do innych programów, drukować je i zapisywać w postaci tekstowej.

Zmiana parametrów programu

W programie poprzez menu Narzędzia / Czcionka możemy zmienić parametry czcionki jaką będą wyświetlane wszelkie funkcje w oknie programu. Podczas wyboru będą jedynie dostępne czcionki o jednakowej szerokości liter (Courier itp.) aby właściwie odwzorować wyświetlane funkcje.

4. Założenia dydaktyczne programu

Program został napisany w celu wykorzystania go w realizacji programu dydaktycznego w ramach przedmiotu „Podstawy techniki cyfrowej”

4. Literatura

Opis metody minimalizacji

J. Siwiński „*Układy przełączające w automatyce*”

Programowanie