

Dekompozycja funkcji logicznej

Potrzebna w praktyce np. do podzielenia funkcji między kilka bloków w układzie FPGA

$$y = f(x_{n-1}, x_{n-2}, \dots, x_1, x_0) = f(A, B)$$

podział zmierzonych wejściowych na zbiór wejściowy A i zbiór wyjściowy B

Tablica dekompozycji

Macierz o wierszach indeksowanych elementami A i kolumnach indeksowanych elementami B

Krotność kolumnowa $v(A|B)$

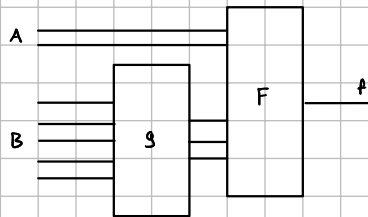
liczba istotnie różnych kolumn w tablicy dekompozycji

Twierdzenie o dekompozycji

$$f(A, B) = F[g_{i-1}(B), \dots, g_1(B), g_0(B), A]$$

\Leftrightarrow

$$v(A|B) \leq 2^i$$

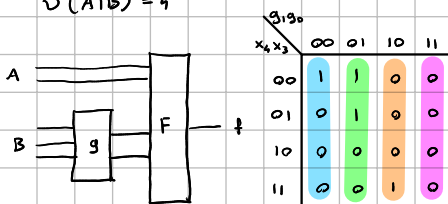


Dekompozycja funkcji 5 zmiennych i przykład

$$y = f(x_4, x_3, x_2, x_1, x_0) \quad A = \{x_4, x_3\} \quad B = \{x_2, x_1, x_0\}$$

		B							
		x_2, x_1, x_0							
A	x_4, x_3	000	001	010	011	100	101	110	111
	00	1	1	1	1	0	0	0	0
	01	0	1	1	1	0	0	0	0
	10	0	0	0	0	0	0	0	0
	11	0	0	0	0	1	1	1	0

$$v(A|B) = 4$$



$$f = F[g_1(x_2, x_1, x_0), g_0(x_2, x_1, x_0), x_4, x_3]$$

x_2	x_1	x_0	g_1	g_0
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

- grupowanie równoważnych kolumn
- zakodowanie (ponumerowanie) unikatowych kolumn
- zdefiniowanie na tej podstawie funkcji g_i

Dekompozycja funkcji nie u pełni dekodowej

cde ab	000	001	010	011	100	101	110	111
00	1	-	0	1	-	0	1	0
01	-	-	-	-	1	1	-	-
10	-	0	1	0	0	-	0	1
11	0	1	-	-	-	-	-	-

$$\cup (A|B) = ?$$

Szukamy takiego sposobu pogrupowania kolumn
który da najmniejszą krotkość

Zgodność kolumn

Kolumny są zgodne jeśli nie istnieją wiersze,
u których obie są dekodowane i sprzeczne

Kolumny zgodne można łączyć

Zgodność nie jest przechodnia

Zgodność jest zwrotna i symetryczna

Pary zgodne umożliwiając utworzenie
maksymalnych grup kolumn zgodnych
(maksymalnej klasy zgodności - MKZ)

Obliczanie MKZ metoda 1 (bezpośrednia)

$$\begin{array}{l} \{a,b\} \\ \{b,c\} \rightarrow \{a,b,c\} \\ \{a,c\} \end{array} \quad \begin{array}{l} \{a,b,c\} \\ \{b,c,d\} \\ \{a,c,d\} \\ \{a,b,d\} \end{array} \rightarrow \{a,b,c,d\}$$

i tak dalej, dopóki da się łączyć

Obliczanie MKZ metoda 2

1. wypisujemy wszystkie pary zgodne

2. Sklejamy

- pary u trójki i odznaczamy wszystkie pary
- trójki u czwórki i odznaczamy wszystkie trójki
- itd dopóki da się łączyć

3. Zostają same niezłączone zbiory \rightarrow MKZ

cde ab	000	001	010	011	100	101	110	111
00	1	-	0	1	-	0	1	0
01	-	-	-	-	1	1	-	-
10	-	0	1	0	0	-	0	1
11	0	1	-	-	-	-	-	-

0,3 1,3 2,5 3,4 4,5 5,7

0,4 1,4 2,7 3,6 4,6

0,6 1,5

1,6

0,3 ✓

0,4 ✓

0,6 ✓

1,3 ✓

1,4 ✓

1,5 ✓

1,6 ✓

2,5 ✓

2,7 ✓

3,4 ✓

3,6 ✓

4,5 ✓

4,6 ✓

5,7 ✓

0,3,4 ✓

0,3,6 ✓

0,4,6 ✓

1,3,4 ✓

1,3,6 ✓

1,4,5

1,4,6 ✓

2,5,7

3,4,6 ✓

MKZ

0 3 4 6

1 3 4 6

1 4 5

2 5 7

Tablica zgodności Petricka Szukam minimalnego pokrycia

		0	1	2	3	4	5	6	7
0	3 4 6	A	✓			✓	✓		✓
1	3 4 6	B		✓		✓	✓		✓
1	4 5	C		✓		✓	✓		
2	5 7	D			✓		✓		✓
		A	B ∨ C	D	A ∨ B	A ∨ C	C ∨ D	A ∨ D	D

Metoda analityczna

Zapisuję jako wyrażenie Booleanne i przekształcam równość
Zamiana postaci iloczynowej na sumacyjną

$$\begin{aligned}
 f(A,B,C,D) &= A(B+C)D(A+B)(A+B+C)(C+D)(A+B)D \\
 &= A(B+C)D(A+B)(A+B+C)(C+D) \\
 &= A(A+B)(B+C)(A+B+C)D(C+D) \\
 &= A(B+C)D \\
 &= ABD + ACD \rightarrow 2 możliwości dla minimalnego pokrycia
 \end{aligned}$$

MK2		Minimalne pokrycie	Bez postereku kolumn
0 3 4 6	0 3 4 6	0 3 4 6	0 3 6
1 3 4 6	1 4 5	1 4 5	1 4 5
1 4 5	2 5 7	2 5 7	2 7
2 5 7			

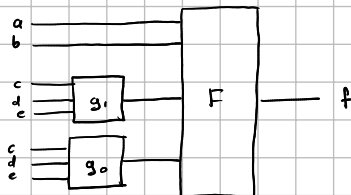
$$U(A|B) = 3$$

cde \ ab	000	001	010	011	100	101	110	111
00	1	-	0	1	-	0	1	0
01	-	-	-	-	1	1	-	-
10	-	0	1	0	0	-	0	1
11	0	1	-	-	-	-	-	-

g ₀ \ ab	00	01	10	11
00	1	0	0	-
01	-	1	-	-
10	0	0	1	-
11	0	1	-	-

F

c	d	e	g ₁	g ₀
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	0
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	0



Dekompozycja FPGA

Dąży do upakowania funkcji w blokach
o danej liczbie wejść

Proces iteracyjny (zuszyczeń)

Strategia od wyjść do wejść

- zapewnić odpowiednią liczbę wejść dla bloku związanego
- dekompozycja bloku wolnego

Strategia od wejść do wyjść

- zapewnić odpowiednią liczbę wejść dla bloku wolnego
- dekompozycja bloku związanego