**Imię i nazwisko:** Jakub Pranica **Grupa:** Informatyka -13i **Data:** 03-04-2017 **Przedmiot:** Metody Programowania

**Wykładowca:** dr inż. Zbigniew Kokosiński **Semestr:** 2

**Temat:** Napisać algorytm realizujący podaną funkcję logiczną przy użyciu multiplekserów MUX 4:1

**Multiplekser** (w skrócie MUX) – [układ kombinacyjny](https://pl.wikipedia.org/wiki/Uk%C5%82ad_kombinacyjny), najczęściej [cyfrowy](https://pl.wikipedia.org/wiki/Uk%C5%82ad_cyfrowy), służący do wyboru jednego z kilku dostępnych sygnałów wejściowych i przekazania go na wyjście. Posiada k wejść informacyjnych, n wejść adresowych i jedno wyjście y.

**Opis programu:** Program wczytuje dane z wcześniej przygotowanego pliku lub generuje je na podstawie informacji podanych przez użytkownika. W tym drugim przypadku program prosi użytkownika o podanie liczby wejść (n=|X|) oraz liczbę stanów określonych (1 <= p <= 2^n). Układ jest opisany za pomocą macierzy binarnych X i Y (z niepełną określonością). Wczytany lub wygenerowany układ zostaje wypisany na ekran (lub do pliku), po czym program rozpoczyna syntezę UK(X,Y) na MUX 4:1 (k=2). Zostaje wybranych k zmiennych adresowych dla 2^L UK(X,Y) poziomu L. Następnie ustalane są kolejno wartości wszystkich kombinacji wartości k zmiennych adresowych, dla których zdekomponowany zostaje każdy UK(X,Y) poziomu L na 2^(L+1) UK(X,Y) poziomu L+1. Program sprawdza czy dany UK poziomu L+1 umożliwia bezpośrednie wyznaczenie kombinacyjnej zależności Y z X w postaci wartości stałej (0 lub 1) lub pojedynczej zmiennej prostej lub zanegowanej - jeżeli tak, to program przypisuje tę wartość/zmienną na odpowiednie wejście MUX, natomiast w przeciwnym wypadku, gdy jest możliwa przynajmniej jedna dalsza dekompozycja, kontynuuje ją w sposób rekurencyjny. Na koniec program wyprowadza rozwiązanie do pliku i/lub na ekran.

**(Pseudokod na odwrocie kartki)**

**Napotkane problemy: 1.**Reprezentacja macierzowa układu kombinacyjnego zapisana, za pomocą tablic w języku programowania c++ **2.**Określenie i wygenerowanie podanej ilości wejść wraz z różnymi stanami określonymi. **3.**Tworzenie podmacierzy (dla UK wyższego poziomu) oraz przekazywanie ich za pomocą funkcji rekurencyjnych **4.**Wypisanie na ekran i do pliku wyników dekompozycji na każdym poziomie wraz z odpowiednim oznaczeniem multiplekserów.

**Przykładowy wynik programu:**

Program dokonuje syntezy układu kombinacyjnego UK(X,Y) oraz realizuje podaną funkcję logiczną przy użyciu multiplekserów MUX4:1.

Gdzie wypisać wyniki? 1 - na ekran 2 - do pliku **1**

**(Przykład 1)** Skąd pobrać dane? 1-wygeneruj losowe 2-wczytaj z pliku **2**

Wczytałem z pliku następujący UK(X,Y):

x1x2x3x4x5 y

X=| 1 0 1 0 1 | Y=| 1 |

| 1 1 0 0 0 | | 1 |

| 0 1 1 1 0 | | 0 |

| 1 1 1 0 1 | | 1 |

| 0 0 1 1 1 | | 1 |

| 1 0 1 1 0 | | 0 |

| 0 1 0 0 1 | | 0 |

| 1 1 1 1 1 | | 1 |

| 1 1 0 1 1 | | 0 |

Objaśnienie formatu: MUXi oznacza MUX poziomu L, którego wyjście podane jest na wejście o numerze dziesiętnym MUX poziomu L-1, 0 <= i <= 2^(k\*L)-1.

Symbol ~ oznacza zaprzeczenie (negacje).

Rozpoczynam syntezę UK(X,Y) jak wyżej na MUX 4:1 (k=2).

Wynik: (Przykład 1)

Dekompozycja UK na MUX:

--------------

L=0

A=x1x2

MUX0(00)=1

MUX0(01)=0

MUX0(10)=~x4

MUX0(11)=MUX3

--------------

L=1

A=x3x4

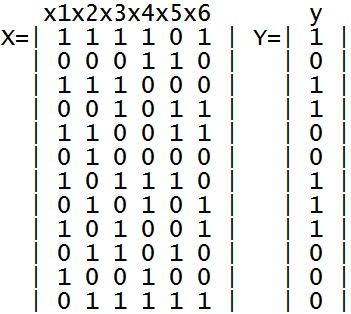
MUX3(00)=1

MUX3(01)=0

MUX3(10)=1

MUX3(11)=1

--------------

**(Przykład 2)** Skąd pobrać dane? 1-wygeneruj losowe 2-wczytaj z pliku **1**

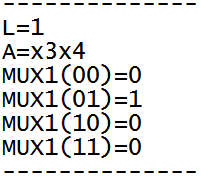
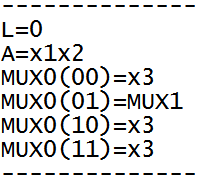
Podaj liczbę wejść n=|X|=**6**

Podaj liczbę stanów określonych (1 <= p <= 64) p=**12**

Wygenerowałem następujący UK(X,Y) **-------------->**

Rozpoczynam syntezę UK(X,Y) jak wyżej na MUX 4:1 (k=2) Wynik:

Dekompozycja UK na MUX:



**Pseudokod:**

**Wypisz na ekran:** "Program Program dokonuje syntezy układu kombinacyjnego UK(X,Y) oraz realizuje podaną funkcję logiczną przy użyciu multiplekserów MUX4:1. \n"

**Wypisz na ekran:** "Skąd pobrać dane? 1-wygeneruj losowe 2-wczytaj z pliku \n"

**Utwórz dwuwymiarową tablicę X oraz jednowymiarową tablicę Y (macierze)**

**Jeżeli użytkownik wpisał 1, wczytaj dane z pliku - w przeciwnym wypadku:**

**Wypisz na ekran:** "Liczba wyjść m=|Y|=1 Podaj liczbę wejść n=|X|="

**Utwórz zmienną n i przypisz do niej wartość podaną przez użytkownika.**

**Wypisz na ekran:** " Podaj liczbę stanów określonych (1 <= p <= " **<< 2^n <<** ") p="

**Utwórz zmienną p i przypisz do niej wartość podaną przez użytkownika.**

**Wygeneruj wartości wejścia (stany określone) oraz wyjścia układu kombinacyjnego UK(X,Y)**

**Wczytane lub wygenerowane dane zapisz odpowiednio w tablicach/macierzach X i Y**

**Zdefiniuj zmienną L oraz nadaj jej wartość początkową 0** (int L = 0;)

**Wybierz 2 zmienne (k=2) adresowe dla 2^L UK(X,Y) poziomu L.**

**Wykonaj funkcję dekomponuj(X,Y,L)**

//...

**Funkcja dekomponuj(wejścia, wyjścia, poziom):**

**Wykonuj kolejno dla wszystkich kombinacji wartości 2 zmiennych adresowych:**

**Zdekomponuj dany UK(X,Y) poziomu L na UK(X2,Y2) poziomu L+1**

**Sprawdź czy UK(X2,Y2) poziomu L+1 umożliwia bezpośrednio wyznaczenie kombinacyjnej zależności Y2 z X2 w postaci wartości stałej (0 lub 1) lub pojedynczej zmiennej prostej lub zanegowanej z macierzy X2.**

**Jeżeli tak, to przypisz tę wartość/zmienną na odpowiednim wejściu MUX poziomu L.**

**W przeciwnym wypadku, jeżeli dalsza dekompozycja jest możliwa, to wykonaj rekurencyjnie funkcję dekomponuj(X2,Y2,L+1)**

//...dalszy ciąg głównej funkcji programu:

**Wyprowadź na ekran wyjście w przykładowym formacie:**

MUXi oznacza MUX poziomu L, którego wyjście podane jest na wejście o numerze dziesiętnym MUX poziomu L-1, 0 <= i <= 2^(2\*L)-1. Symbol ~ oznacza zaprzeczenie (negację).

**Wnioski:** Na podstawie analizy wyników zwracanych przez program i obliczeń wykonanych poza nim, można stwierdzić, iż program działa i prawidłowo generuje układ kombinacyjny UK(X,Y) dla podanych przez użytkownika danych, a następnie dokonuje jego syntezy na MUX 4:1 (k=2) korzystając z uogólnionego Tw. Shannona o dekompozycji logicznej. Czas generacji wydłuża się wraz ze wzrostem podanych przez użytkownika wartości - niska jednak złożoność sprzętowa wynika z niepełnej określoności UK.

**Literatura:**

* pl.wikipedia.org/wiki/Multiplekser
* Kokosiński Zbigniew : Synteza układów kombinacyjnych na multiplekserach i demultiplekserach, ICACIT 2015, Kiev, Ukraine
* Łuba T. : Synteza układów logicznych, Oficyna Wydawnicza PW, Warszawa 2005.