**Dekompozycja (*Decomposition*)**

Dekompozycja funkcjonalna jest metodą umożliwiającą realizację wielowejściowego układu kombinacyjnego, opisanego wielowyjściową funkcją boolowską, za pomocą wielu układów połączonych ze sobą w taki sposób, aby zależność funkcyjna realizowanego układu nie uległa zmianie. Przy realizacjach na pamięciach dekompozycja umożliwia realizację funkcji z użyciem kilku tańszych pamięci ROM o mniejszej liczbiei wejść zamiast jednej droższej pamięci ROM o dużej liczbie wejść.

Dekomponując funkcję boolowską zakłada się, że funkcja o *X* argumentach jest dekomponowana na dwa komponenty oznaczane jako blok *G* oraz blok *H*. Argumentami pośrednimi, czyli argumentami bloku *G,* jest zbiór *V*, natomiast argumentami bezpośrednimi, czyli argumentami bloku *H,* jest zbiór *U* oraz wyjścia bloku *G* – *F* = *H*(*U*, G(*V*)). Dodatkowo podstawowy schemat dekompozycji może być rozbudowany do tzw. dekompozycji podwójnej – *F* = *H*(*G*(*U*1), *G*(*U*2)), istotnej przy realizacjach, której komponenty są wyłącznie blokami pamięci.

W celu wyznaczenia dekompozycji funkcji boolowskiej należy wczytać plik danych zawierający funkcję boolowską (wczytana przez użytkownika funkcja zostanie przedstawiona w obszarze danych wejściowych), wybrać argumenty bloku *G* oraz argumenty bloku *H*, wybrać metodę wyznaczania pokrycia kolumnowego, wybrać wyniki, które mają zostać zaprezentowane, a następnie uruchomić procedurę wyznaczania dekompozycji wczytanej funkcji boolowskiej. Po zakończeniu obliczeń wybrane przez użytkownika wyniki obliczeń zostaną zaprezentowane w obszarze prezentacji wyników obliczeń. Aplikacja umożliwia również eksport uzyskanych wyników obliczeń do pliku zewnętrznego.

Dzięki zastosowaniu graficznego interfejsu użytkownika, obsługa aplikacji jest bardzo prosta i intuicyjna. Po uruchomieniu programu jest wyświetlane główne okno aplikacji. Z tego poziomu użytkownik może przeprowadzić wszystkie zaplanowane operacje.

**Literatura**

Iwaszko T., *Dekompozycja funkcji boolowskich z zastosowaniem do syntezy generatorów adresów*. Praca inżynierska, Wydział EiTI Politechniki Warszawskiej, Warszawa 2016.

Chądzyński B., *Implementacja algorytmu dekompozycji układów kombinacyjnych z zastosowaniem metody uzupełniania funkcji boolowskich.* Praca inżynierska, Wydział EiTI Politechniki Warszawskiej, OKNO, Warszawa 2015.