Wrocław, 09.10.2024

Układy Cyfrowe i Systemy Wbudowane

Laboratorium 2 – Układy Kombinacyjne

Zadania:

1. Dowolna bramka – funktor – 2-wejścia, 1-wyjście
2. Implementacja funkcji logicznej.
3. Implementacja układu translatora kodu: może być idioto-odporny lub nie

Co chciałbym zobaczyć w sprawozdaniu:

- schemat w Xilinx-ie i informacja o zasadach konstrukcji danego układu,

- zapis procesu minimalizacji funkcji boolowskiej,

- zasady / szkic konstrukcji układu translatora kodu,

- obrazki – wyniki działania symulatora dla kompletu pobudzeń każdego układ,

- krótki raport z przeprowadzonej implementacji sprzętowej.

Uwagi dotyczące pracy z Xilinx-em:

- pracować zgodnie z instrukcją laboratoryjną – partyzantom i kozakom z góry dziękujemy,

- prawidłowo wybrać platformę sprzętową i warunki jej pracy,

- każdy układ budujemy „od zera" jako nowy projekt – nie przerabiamy „starego na nowy",

- bogactwo bibliotek Xilinx-a duże – rozglądać się cierpliwie, robić substytucje jeśli trzeba,

- pamiętać o prawidłowych zaciskach wejściowo-wyjściowych budowanych układów,

- nie bać się „roboczych – diagnostyczno–kontrolnych" zacisków wyjściowych,

- nazwy zacisków i nazwy przewodów inne niż nazwy sygnałów systemowych – po polsku,

- jednowyrazowe, „żołnierskie" nazwy plików – bez cudowania z polskimi znakami,

- zapisywać / nagrywać / sejwować pliki na każdym etapie projektu – nie ma automatu!

- plik <nazwa>.vhd musi być różny od pliku <nazwa>.sch – inaczej klapa!

- symulator: ISIM – koniecznie, ModelSIM – czasem kuleje dostęp do licencji,

- czas momentu zmiany sygnału w symulatorze zawsze liczony od początku symulacji,

- jeśli w schemacie zacisk wyjściowy może być podpięty do wejściowego wprost przewodem – bez funktorów „po drodze" – wstawić „sztucznie coś" co nie zmieni wartości logicznej sygnału – inaczej będzie zgłaszany błąd,

- obrazki do sprawozdania: schemat, rezultaty symulacji: PrintScreen – głupie, ale skuteczne,

- używać pliku UCF zawsze ze strony materiałów do laboratorium.