

Lista 13

zadanie 1

- a)
- pc_sel wybierze sumator+4.
 - rd_sel wybierze wyjście ALU.
 - rd_we będzie miało stan wysoki.
 - alu_bsel wybierze wyjście pliku rejestrów RS2.
 - alus_asel wybierze wyjście pliku rejestrów RS1.
 - alu_func wybierze odpowiedni kod operacji dodawania dla ALU.
 - mem_we będzie miało stan niski.
- b)
- Licznik instrukcji, Sumator+4, Pamięć kodu, Plik rejestrów, ALU.
- c)
- Pamięć danych, Sumator(ten drugi), Porównanie do zera, Generator stałych

zadanie 2

- a)
- Pamięć danych

Rodzaj operacji	Odsetek instrukcji jeśli używa
OP	Nie używa
I (bez LOAD)	Nie używa
LOAD	25%
STORE	10%
BRANCH	Nie używa
JAL	Nie używa
U	Nie używa
Łącznie	35%

- b)
- Pamięć instrukcji: 100%, gdyż każda instrukcja musi zostać zczytana z pamięci instrukcji.

- c)
- ALU

Rodzaj operacji	Odsetek instrukcji jeśli używa
OP	24%
I (bez LOAD)	20%
LOAD	25%
STORE	10%
BRANCH	11%
JAL	Nie używa
U	8%
Łącznie	98%

- d)
- Generator stałych

Rodzaj operacji	Odsetek instrukcji jeśli używa
OP	Nie używa
I (bez LOAD)	20%
LOAD	25%
STORE	10%
BRANCH	11%
JAL	2%
U	8%
Łącznie	76%

- e)
- Jego zachowanie nie jest dobrze zdefiniowane dla takich sytuacji, może np. generować jakieś śmieci, gdyż jego wyjście i tak jest ignorowane w dalszej części układu.

zadanie 3

- a)
- Jeśli stan wysoki alu_bsel oznacza wybranie wyjścia pliku rejestrów, to niepoprawnie działać będą operacje z rodzaju OP i BRANCH.
W przeciwnym wypadku źle działać będą instrukcje OP-IMM, STORE, LOAD, JALR i AUIPC.
- b)
- Jeśli stan rd_we będzie zawsze niski to niepoprawnie będą działać instrukcje z rodzaju OP, OP-IMM, LOAD, JAL, JALR, LUI, AUIPC.
- c)
- Jeśli stan alu_zero będzie zawsze niski, niepoprawnie będą działać instrukcje z rodzaju BRANCH.

zadanie 5

- a)
- Nowe elementy nie są potrzebne, do obliczenia adresu wykorzystamy ALU, do odczytu i zapisu - pamięć danych.
- b)
- Nie potrzebne są żadne modyfikacje.
- c)
- Nie potrzebne są nowe połączenia.
- d)
- Nie trzeba, do wybrania odpowiedniego wejścia do ALU wystarczą nam sygnały alu_bsel i alu_asel, sygnał alu_func wybierze dodawanie, rd_sel wybierze wyjście pamięci danych, sygnałem rd_we zapewnimy zapisanie wartości pod wskazany rejestr. Sygnał mem_we ustawiamy na niski, a sygnał pc_sel powinien wybrać wyjście sumatora+4.

zadanie 6

- a)
- Będzie potrzebny nowy multiplexer podpięty w miejscu obecnego AD (późniejszego AD1).
- b)
- Plik rejestrów musiałby zostać rozszerzony o dodatkowy port do zapisu, w konsekwencji miałby on wejścia AS1, AS2, AD1, RD1, WE1, **AD2, RD2, WE2** i tak jak poprzednio dwa wejścia RS1 i RS2. Multiplexer obecnie podłączony do wejścia RD(w przyszłości do RD1) musiałby zostać rozszerzony o dodatkowe wejście.
- c)
- Do multiplexera przy wejściu AD1 podpięte powinny zostać bity [7:11] wyjścia pamięci instrukcji (obecne połączenie) oraz bity [15:19] (adres rs1). Do rozszerzonego multiplexera przy RD1 dodatkowo doprowadzone powinno być wyjście pliku rejestrów RS2. Do wejścia AD2 podpinamy bity [20:24] (adres rs2). Na koniec do wejścia RD2 podpinamy wyjście RS1.
- d)
- Potrzebny będzie sygnał sterujący dla nowego multiplexera (np. ad1_sel) oraz sygnał sterujący zapisem drugiego portu do zapisu (np. rd_we2).