

Jednostka wykonawcza SPI_3

Jednostka wykonawcza SPI_3 wraz z bliźniaczymi modułami SPI_2 oraz SPI_1, wchodzi w skład jednostki arytmetyczno-logicznej, którą steruje-Master. Jednostka SPI_3 wykonuje operacje :

4'b0000 : o_result= i_argA + i_argB ;	DODAWANIE
4'b0001: o_result = i_argA i_argB;	OR
4'b0011: o_result= ~(i_argA & i_argB);	NAND
4'b0010: o_result=(i_argA>>>1);	ARYTMETYCZNE PRZESUNIECIE W PRAWO
4'b0110: o_result=(i_argA<<<1);	ARYTMETYCZNE PRZESUNIĘCIE W
4'b0111: o_result=s_zliczanie0;	ZLICZANIE ZER
4'b0101:o_result=s_OnehotToBinary;	ZAMIANA KODU ONEHOT NA BINARY
4'b0100:o_result=s_binaryToThermometer;	ZAMIANA KODU BINARY NA THERMOMETER
4'b1100:o_result=s_crc;	WYZNACZENIE KODU CRC
4'b1101:o_result=s_U2naGraya;	ZAMIANA KODU U2 NA KOD GRAYA
4'b1111:o_result=U1naU2a;	ZAMIANA KODU U1 NA KOD GRAYA
4'b1110:o_result=s_crcsprawdzenie;	SPRAWDZENIE KODU CRC-3

flagi:

PF-znacznik uzupełnienia do parzystej liczby jedynek;

ZF-znacznik zera informujący, że wynikiem operacji są same zera;

SF-znacznik znaku informujący, że wynik operacji to liczba ujemna;

OF-znacznik jedynki informujący, że wynikiem operacji są same jedynki;

Sygnały:

CLK-sygnał zegarowy, odpowiadający za wysyłanie impulsów zegarowych;

CS-sygnał aktywujący możliwość obliczania przez moduły SPI, aktywowany stanem niskim;

MISO-sygnał Master Input Slave Output, sygnał komunikacyjny, między jednostką Master a SPI, gdzie komunikacja zachodzi od SPI do Master;

MOSI-sygnał Master Output Slave Input, sygnał komunikacyjny, między jednostką Master a SPI, gdzie komunikacja zachodzi od Master do SPI;

Zarządzanie danymi wejściowymi i wyjściowymi:

1. Do urządzenia należy wprowadzić dane w kolejności: i_argA(8 bitów)-0-i_argB(8 bitów)-0-i_oper(4 bity)-0000;

2. Wynikiem obliczeń modułu SPI_3 jest: o_result(8 bitów)-o_PF-o_ZF-o_SF-o_OF-0(16 bitów);

Działanie:

Można wyróżnić 5 stanów działania SPI_3.

1. Czekanie na stan niski sygnału i_cs przy sygnałach enable=0(zakaz przesuwania), a gdy uzyskanie stanu niskiego sygnału i_cs oraz zezwolenie na przesuwanie oraz ustawienie 8 cykli zegara;

2. Wpisanie arg_A;

3. wpisanie arg_B;

4. wpisanie i_oper;

5. Wystawienie wyniku oraz powrót do stanu 1;

Struktura rejestrów:

Rejestr	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
argA	A7	A6	A5	A4	A3	A2	A1	A0
argB	B7	B6	B5	B4	B3	B2	B1	B0
oper	O3	O2	O1	O0	0	0	0	0

Testowanie:

Za pomocą wygenerowanych danych wejściowych oraz oczekiwanych danych wyjściowych, SPI_3 został przetestowany. Jego wybór (spośród innych SPI) jest dokonywany przez 'testbench', a następnie sprawdzane są wartości oczekiwane z rzeczywistymi, porównywane, oraz podsumowana ilość błędów oraz dobrych transmisji.