[SE273] Term project: A simple calculator design

1. 목적

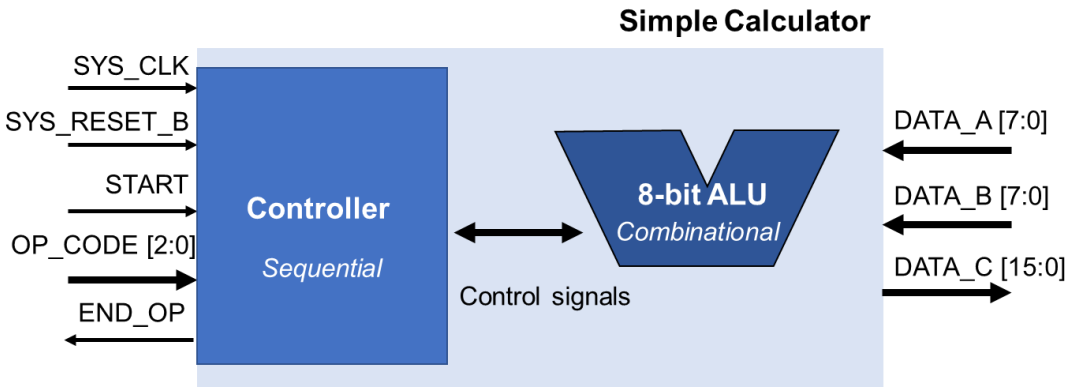
디지털 회로 설계를 하드웨어로 구현함으로써 이를 통해 Combinational Logic 과 Sequential Logic 을 설계하는 능력, Hierarchical Design, Synchronous Design 에 대한 이해, VHDL 언어에 대한 이해 및 설계 능력 및 Simulator 를 이용한 검증 능력 등을 배양한다.

2. 설계 목표

덧셈, 뺄셈, 곱셈을 수행하는 간단한 calculator 설계한다. SM chart 를 사용하여 각 연산을 위한 알고리즘을 구현하되 최소한의 logic 을 사용하여 설계하여야 하며, Top-down 방식으로 설계하되 각 블록의 설계를 검증한 후 전체 블록을 집적하여 설계를 완성한다. Test Bench 을 통해 simulation 을 이용하여 자신이 설계한 회로를 검증한다.

3. 설계 내용

(a) Block Diagram



(b) Signal Description

- System Signals

Name	Type	Description	Note
SYS_CLK	std_logic	System Clock	10 MHz
SYS_RESET_B	std_logic	System Reset	Active Low

- Input Signals

Name	Type	Description	Note
START	std_logic	Start signal	Active High
OP_CODE	std_logic_vector(2:0)	Operation code	3 Commands, 5 Reserved
DATA_A	std_logic_vector(7:0)	Operand A	Signed 8-bit
DATA_B	std_logic_vector(7:0)	Operand B	Signed 8-bit

- Output Signals

Name	Type	Description	Note
END_OP	std_logic	Operation End	Active High
RESULT_C	std_logic_vector(15:0)	Result	Signed 16-bit

(c) Function & OP CODE

Function	OP_CODE[2:0]	Description
ADD	001	DATA_C = DATA_A + DATA_B
SUB	010	DATA_C = DATA_A - DATA_B
MUL	100	DATA_C = DATA_A * DATA_B
Reserved	Others	DATA_C=2'0000000000000000

(d) Design Constraints

✓ **ARITHMETIC LIBRARY 연산 사용 금지. (OPERATOR : +, -, *, / , shift_operator) 이것으로 구현한 경우 0 점 처리. 단, Operator counter module 을 사용할 경우, counter 내부에서 +'1'만 사용 가능**

✓ VHDL 의 3 가지 표현 방식인 Structural Description, Behavioral Description, Data Flow Description 의 방법을 모두 사용하여 설계하여야 한다.

- 8 bit Adder module 은 Data Flow Description 방식으로 설계하여야 한다.
- 8 bit-ALU module 은 Structural Description 방식으로 8 bit Adder 및 기타 회로들을 Component 로 포함하여 설계하여야 한다.
- State machine 설계는 Behavioral Description 방식으로 설계하여야 한다.

- ✓ **ADDER 와 SUBTRACTOR 를 먼저 설계한 후 MULTIPLIER 는 위에서 설계한 ADDER 와 SUBTRACTOR 를 이용하여 제시된 Shift and Add Algorithm 에 의해 SM CHART 를 작성한 후 구현한다.**
- ✓ **모든 operator 는 부호를 고려해야한다.**
- ✓ **반드시 제공한 TOP 파일 내부에 설계할 것. (평가에 사용될 TB 는 해당 TOP 파일을 기준으로 설계되어 있음. - 고려하지 않을시 채점 불가)**
- ✓ **Testbench 시작에 충분한 길이의 sys_reset_b 가 들어감.**

4. 설계 과정

(1) 목표 및 기준설정

- 제시된 기능을 구현하기 위해 필요한 이론 정리
- 덧셈, 뺄셈, 곱셈, 나눗셈 각 연산을 하드웨어로 구현하기 위한 이론 정리
- 설계 문제 및 기준(방법 또는 설계사양)을 정의하기 위한 SM Chart 를 그린다.

(2) 합성 및 분석

- VHDL 로 각 모듈들을 설계한다.
- 각 모듈 별 시뮬레이션을 통해 원하는 구조대로 만들어졌는지 확인한다.
- 각 모듈을 합성하여 전체 설계를 완성한 후 시뮬레이션을 통해 검증.

(3) 결과 도출

- 이상의 내용을 hard copy 를 통해 제출하고 작성한 코드는 파일로 제출하여 검증받음.
- 위의 실험 결과 및 구현 방법을 보고서로 작성한다.

5. 보고서 작성

(1) 도입부 (7pt)

A. 목적 (2pt)

B. 목표 및 기준 설정 (3pt)

C. 팀원간 역할 명시 (2pt) (반드시 적적한 역할 분배를 할 것)

(2) 합성 및 분석 (8pt)

A. State Diagram 을 작성하였는가? (3pt)

B. 시뮬레이션을 통하여 만든 모듈을 검증하였는가? (5pt)

(3) 결과 및 논의 (35pt)

A. 결과 도출 (10pt)

B. 토의 (25pt)

i. 제시된 주요 설계요소 및 제한요소를 만족하였는가? (10pt)(설계 요소 및 제한 요소는 위의 Design Constraints 항목 참조)

1. VHDL 의 3 가지 표현 방식인 Structural Description, Behavioral Description, Data Flow Description 의 방법을 모두 사용하여 설계하였는가? (3pt)

2. Shift and add algorithm 에 의해 SM chart 를 작성한후 multiplier 를 설계하였는가? (3pt)

3. 설계한 Adder, subtractor, multiplier 가 부호를 고려하는가? (4pt)

ii. Top-down 방식의 설계가 적용되었는가? (3pt)

iii. Hierarchical Design 방식의 설계가 적용되었는가? (3pt)

iv. Synchronous Design 방식의 설계가 적용되었는가? (3pt)

v. Testbench 의 작성 요령 및 임의의 입력에 대해서 안정적으로 동작하는가? (5pt)

vi. 기타 논의사항 (4pt)

6. 평가방법 (총 100 점)

(1) 동작여부 - testbench 를 이용하여 검증. (50 점)

(2) 보고서 내용 – (50 점)

(3) 카피 적발 시 모두 0 점 처리

(4) 참여하지 않은 팀원은 개별 0 점 처리 가능

7. 제출방법

(1) 제출기한: 2020/12/4 (Fri) 11:59PM (예외 없음)

(2) 제출방법: 프로젝트 파일과 보고서를 묶어서 이메일 (happy2trees@dgist.ac.kr)로 제출. 제출시 제목을 [SE273_TermProject] 조번호_[학번_이름]_[학번_이름]으로 보낼 것.

(단, 보고서는 .pdf 파일로 변환하여 묶을 것.)

8. 질문 및 문의

(1) 아래 조교 이메일로 질문 및 문의할 것.

(2) **이메일 : 한원국(wkhan@dgist.ac.kr), 김진회(happy2trees@dgist.ac.kr)**