

논리회로 설계 및 실험

- 3주차 : 전가산기 및 반가산기 -

담당 교수: 권동현 교수

조교: 송수현 박사과정

3주차 목표

목표

1. 반가산기와 전가산기를 이용하여 구조적 설계를 이해하고 이를 활용한 HDL 모듈의 확장 실습

가산기 (Adder)

- 덧셈 연산을 수행하는 논리회로
- 한 자릿수 연산을 위해서는 Half adder, Full adder 등이 있음
- 멀티비트의 연산을 위해서는 Ripple carry adder, Carry look ahead adder 등이 있음

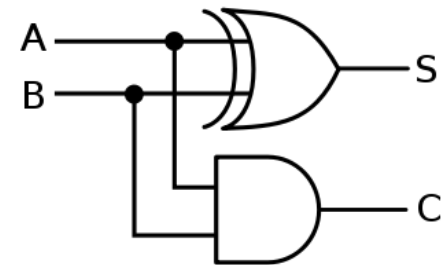
반가산기 (Half adder)

- 한 자릿수 덧셈을 수행하고 합(Sum)과 자리올림수(Carry)를 출력
- Carry는 AND gate, Sum은 XOR gate와 결과가 같음

$$\begin{array}{r} A \quad 1 \\ + B \quad 1 \\ \hline S \quad 0 \\ C \quad 1 \end{array}$$

반가산기의 진리표

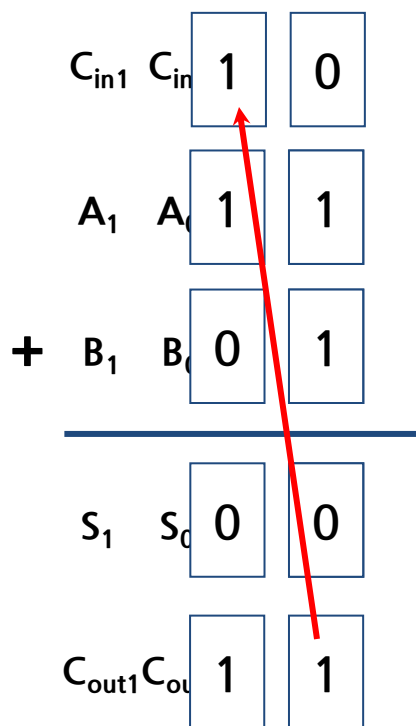
A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



반가산기의 논리회로

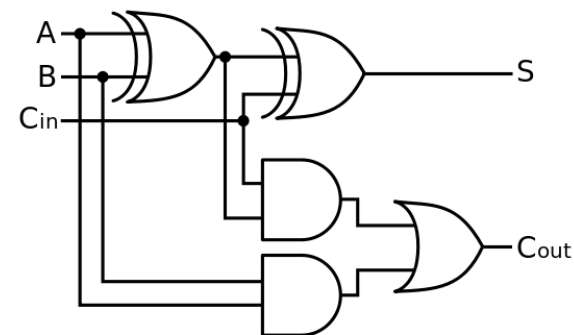
전가산기 (Full adder)

- 한 자릿수 덧셈을 수행할 때 이전 자리의 연산 결과로 받은 Carry를 함께 연산하는 회로
- 두 개의 반가산기와 1개의 OR gate로 구성할 수 있음



전가산기의 진리표

A	B	C_{in}	C_{out}	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

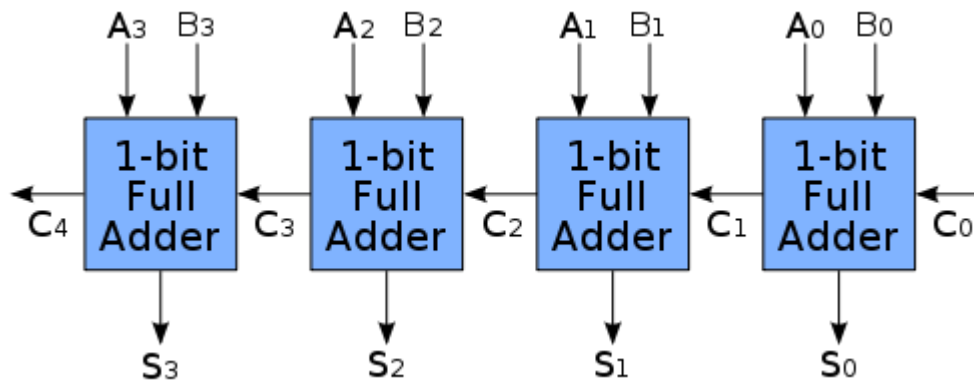


전가산기의 논리회로

리플 캐리 가산기

리플 캐리 가산기 (Ripple carry adder)

- 복수의 전가산기를 이용하여 복수 비트의 덧셈 연산을 할 수 있는 가산기
- 간단한 구조이지만 전가산기의 입력이 이전 전가산기의 출력이므로 전달 지연이 발생함



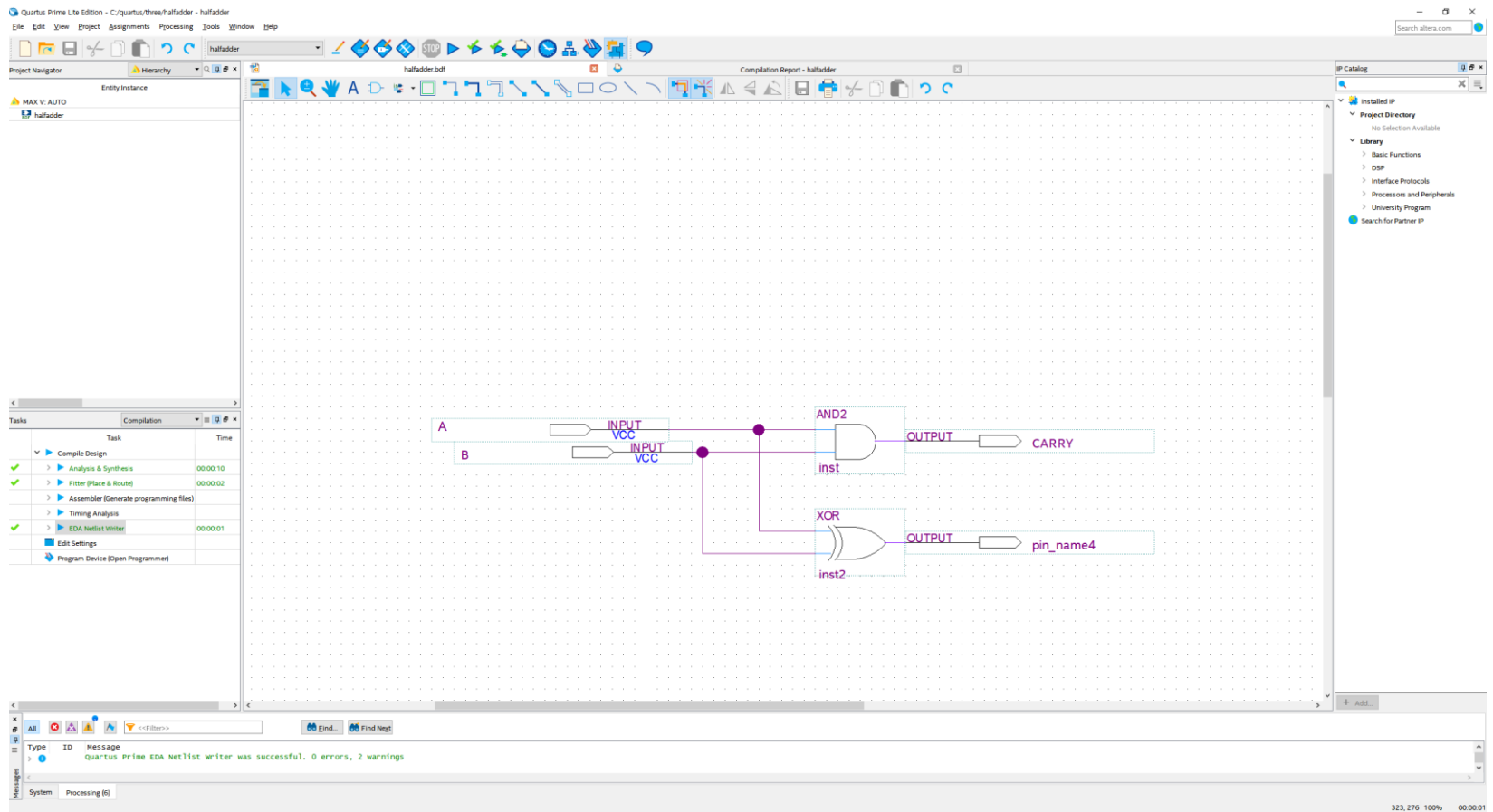
4bit Ripple carry adder의 구조

실습

- Symbol을 이용한 모듈 구현 및 검증
- 가산기 구현 및 검증
- 비교기 구현 및 검증

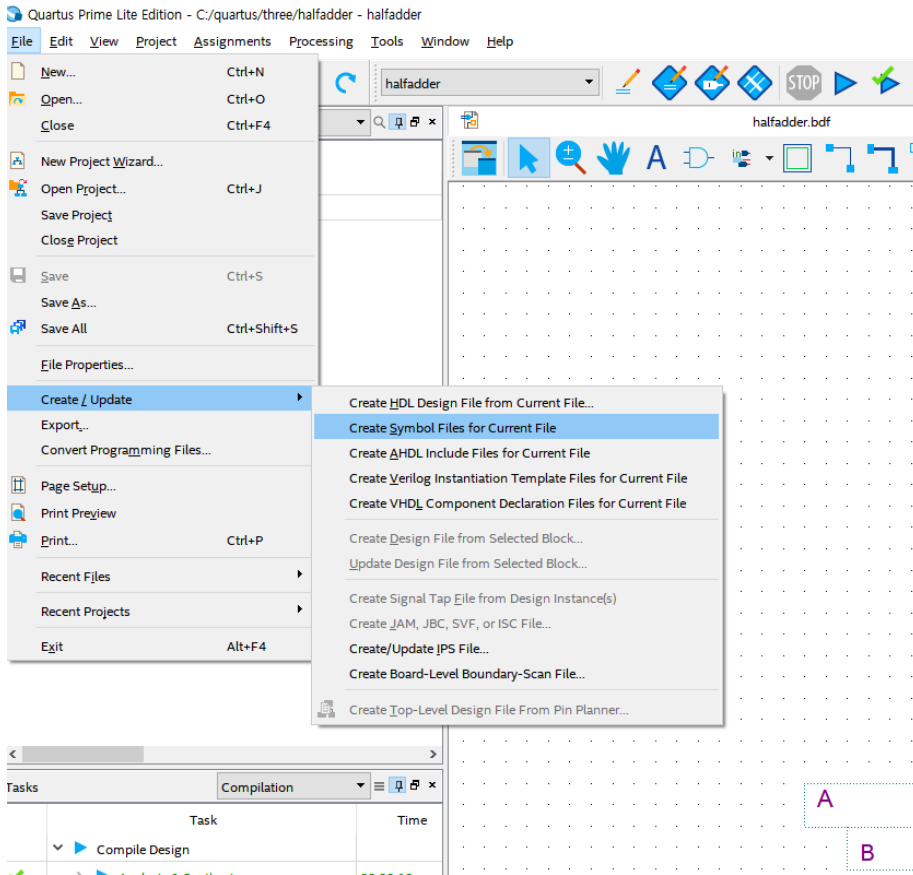
Symbol

- 1) Symbol을 만들 schematic 생성
- 2) Compile

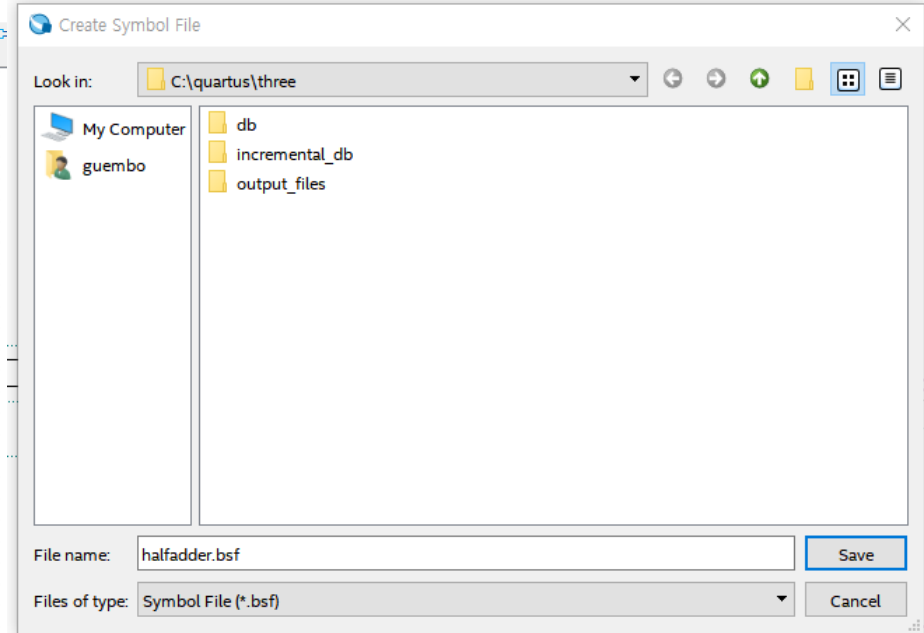


HalfAdder schematic

Symbol

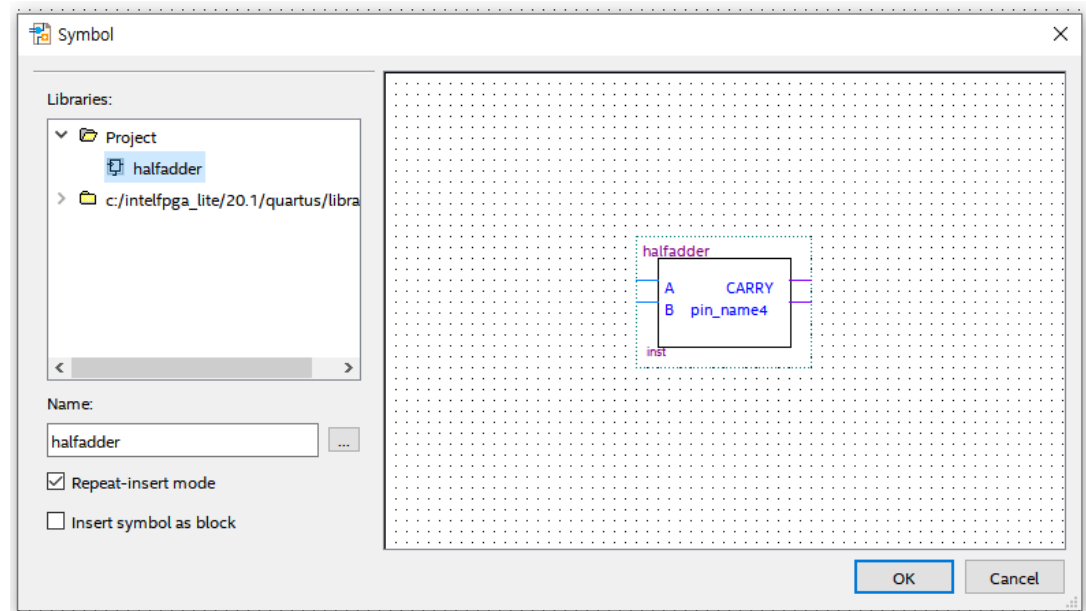
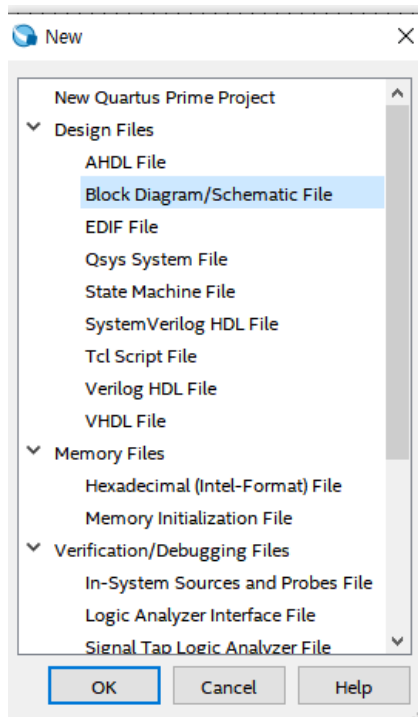


1. 현재 파일 심볼 생성



2. 심볼 저장

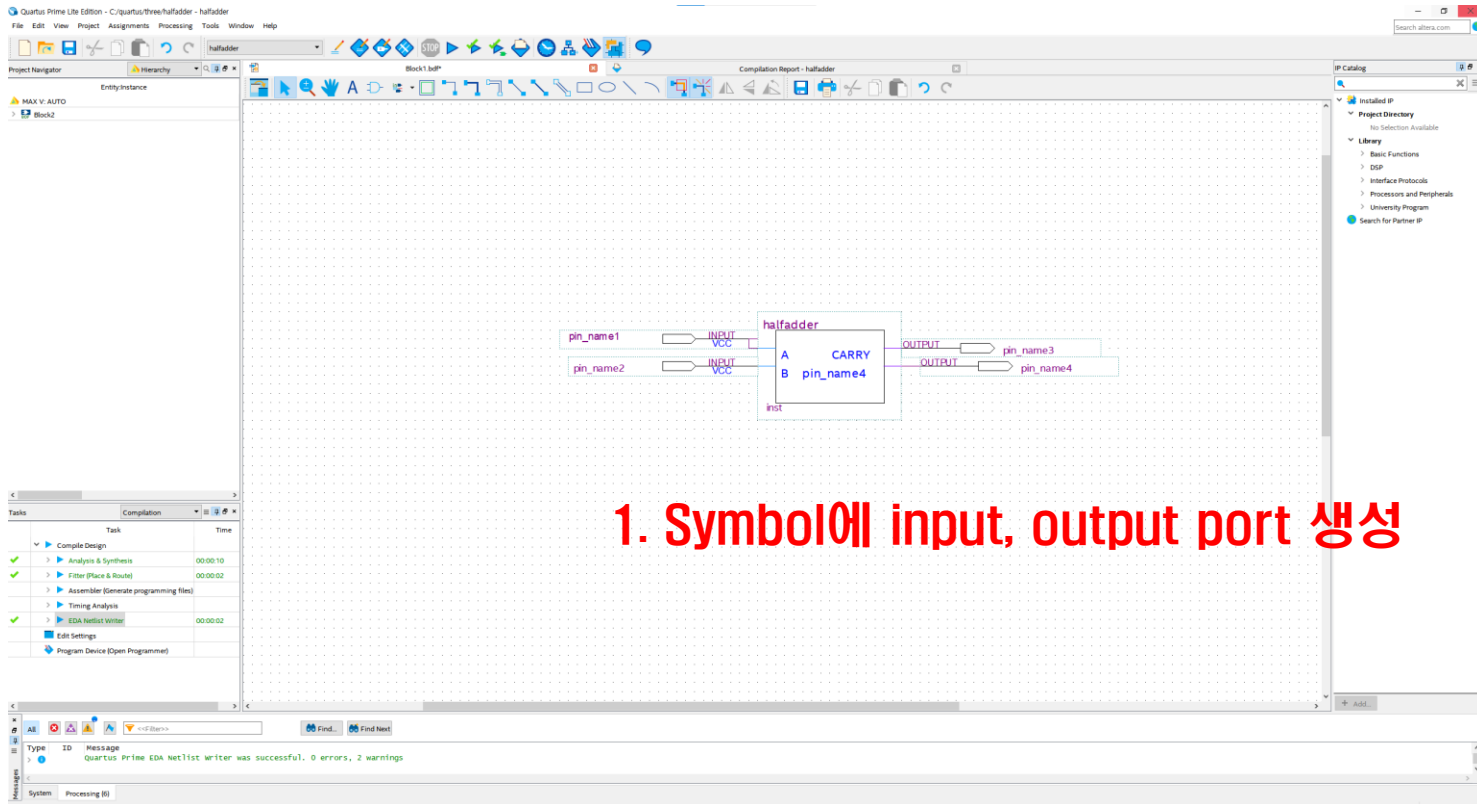
Symbol



2. Project – halfadder 확인

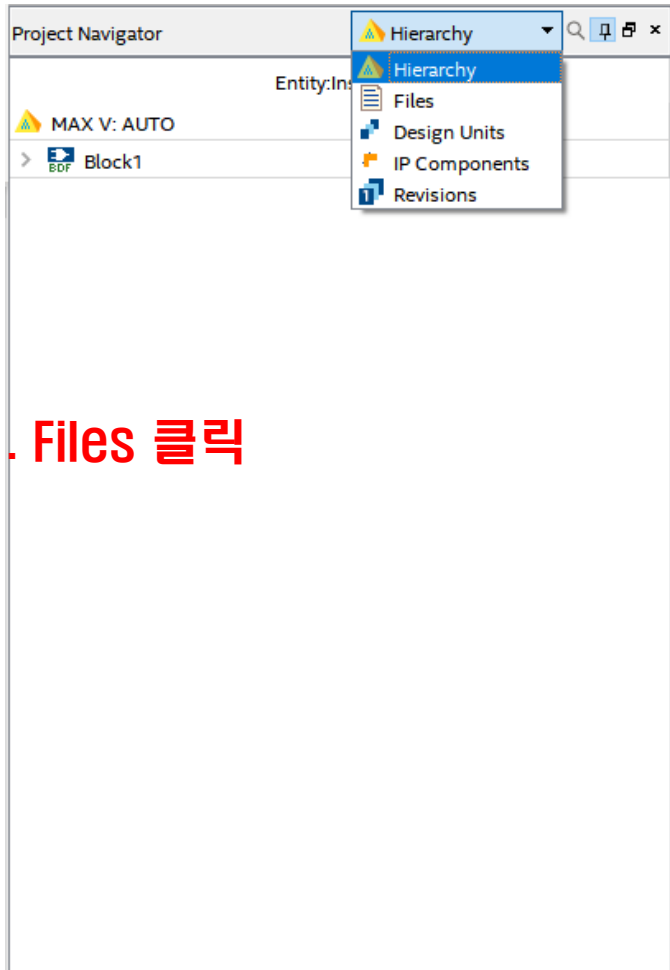
1. 새로운 Schematic File 생성

Symbol

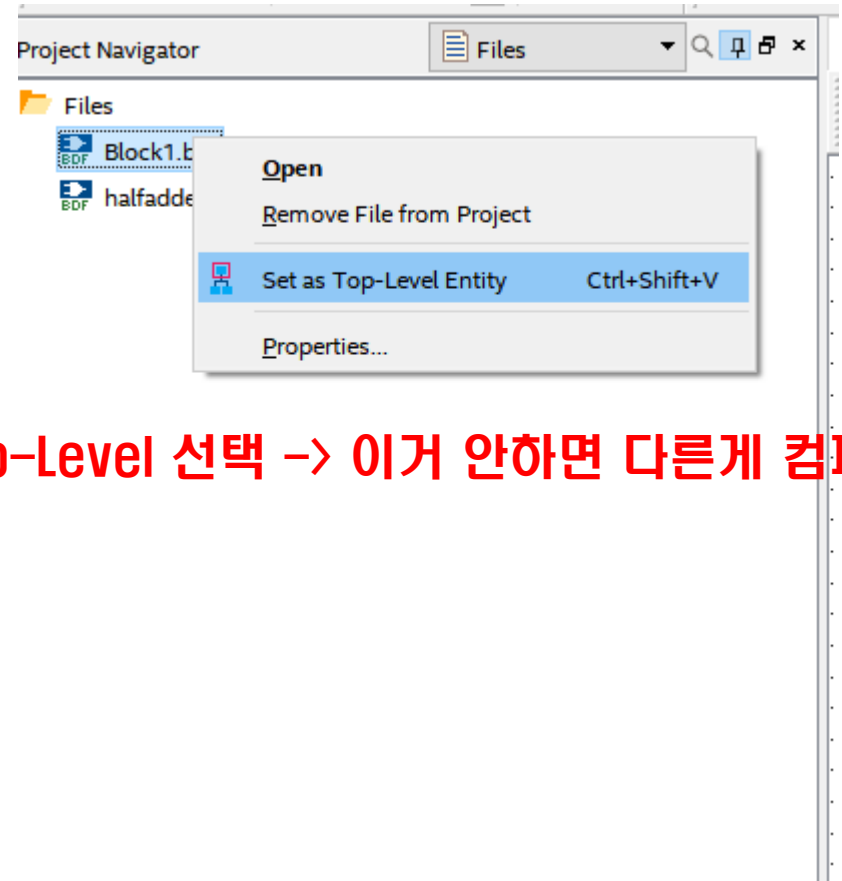


Symbol

1. Files 클릭



2. Top-Level 선택 -> 이거 안하면 다르게 컴파일



Symbol

Tasks		
Compilation		
Task		Time
▼ Compile Design		
✓	> Analysis & Synthesis	00:00:10
✓	> Fitter (Place & Route)	00:00:02
	> Assembler (Generate programming files)	
	> Timing Analysis	
✓	> EDA Netlist Writer	00:00:01
	Edit Settings	
	Program Device (Open Programmer)	

1. 컴파일

Simulation Waveform Editor - C:/quartus/three/halfadder - halfadder - [halfadder_20210902100411.sim.vwf (Read-Only)]

File Edit View Simulation Help

Search altera.com



2. 똑같은 HalfAdder 확인

다음 모듈들을 Symbol로 만드시오

1. Decimal-to-Binary 변환기 -> input : (0~11)
2. 2bit 비교기
3. 반가산기
4. 전가산기 (반가산기를 이용해 구현)
5. 4bit 리플 캐리 가산기 (전가산기를 이용해 구현)

입력		출력	
A	B	A=B	A>B
A ₁ A ₂	B ₁ B ₂	F ₁	F ₂
0 0	0 0		
	0 1		
	1 0		
	1 1		
0 1	0 0		
	0 1		
	1 0		
	1 1		
1 0	0 0		
	0 1		
	1 0		
	1 1		
1 1	0 0		
	0 1		
	1 0		
	1 1		

2bit 비교기(같거나 크고 작고 비교)

아래 기능을 만족하는 가산기를 구현하시오

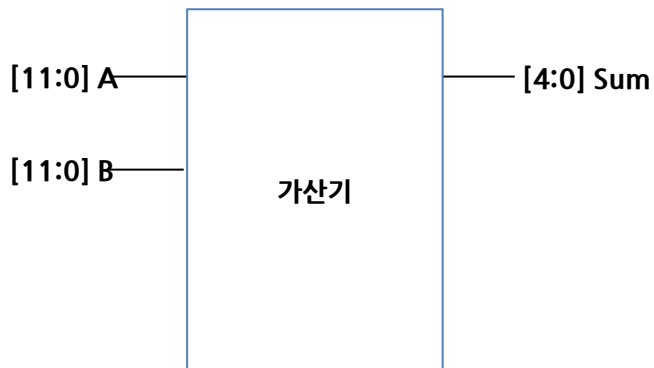
입력 : [11:0] A, [11:0] B

출력 : [4:0] Sum

입력 A, B는 one-hot code이며 십진수 입력을 뜻한다
A와 B로 입력 받은 두 수를 더하여 5bit의 2진수로 출력한다

예를 들어, A2, B9에 입력을 받으면 $2+9=11$ 이므로 $\text{Sum}=\{01011\}$ 이다

4bit 리플 캐리 가산기와 Decimal-to-Binary 회로를 이용하여 구현



아래 기능을 만족하는 4bit 비교기를 구현하시오

입력 : [3:0] A, [3:0] B

출력 : Gt, Eq, Lt

4bit 입력 A와 B를 비교하여 $A > B$ 이면 $Gt=1$, $A=B$ 이면 $Eq=1$, $A < B$ 이면 $Lt=1$ 을 출력하는 4bit 비교기

단, 2bit 비교기 2개를 사용하여 4bit로 확장하라

