

2015년 고급디지털회로설계 Design Project 상세 설계 설명서

1. 제목

PC기반 Function generator (PCFG)

2. 설계 목적

- 설계에서 주어진 기능의 구현을 통해 디지털 회로 설계 능력을 배양한다.
- Block diagram으로 회로를 도식화 하고 Timing diagram을 통해 회로의 동작을 설계하는 능력을 배양한다.
- Xilinx ISE와 Simulator(ISim 또는 ModelSim)를 사용하여 회로를 설계하고 검증하는 능력을 배양한다.

3. 설계 내용

가. 설계 목표

PC로 제어하는 기본적인 파형발생기와 디지털 오실로스코프를 설계한다.

나. 설계 내용

1) Overall Description

가) Function(Mode)

기본적으로 PC Mode, AD Mode, DA Mode가 있으며, 각 mode에 관한 자세한 내용은 이하 mode 설명에서 다룬다.

나) Communication

설계 회로와 PC와의 통신은 USB 통신 module을 통해 이루어진다. PC의 host program으로 회로의 동작을 제어한다.

다) Data Bus Control

USB 통신 module의 in/out data bus 제어를 위한 tri-state buffer를 구현하고, control signal(e.g. MUX select signal)을 생성하여 회로 내부의 input data bus와 output data bus를 제어한다.

라) Board Peripherals

(1) Oscillator / Clock buffer

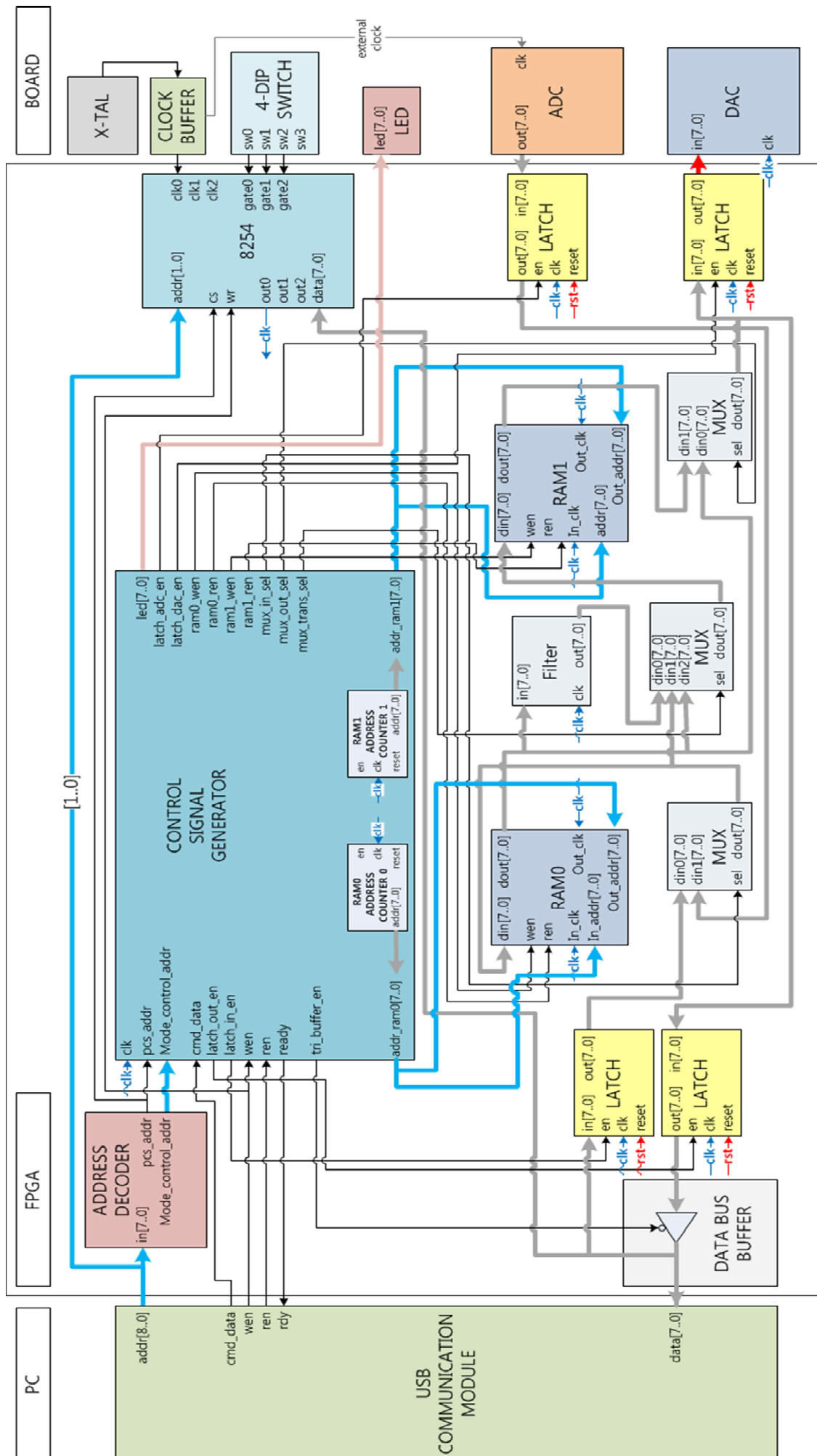
FPGA가 연결되어 있는 PCB(Printed Circuit Board)에 있는 oscillator의 출력이 clock buffer로 입력 되고 clock buffer의 출력이 FPGA로 입력된다. FPGA에 구성된 회로 내부에 있는 8254를 사용하여 입력된 clock signal을 분주하여 system clock으로 사용한다.

(2) ADC / DAC

FPGA에 연결되어 있는 ADC와 DAC를 통해 AD mode, DA mode를 수행한다. ADC를 통해 function generator에서 발생한 signal을 입력받아 AD mode를 수행한다. RAM에 저장된 data를 DAC로 보내 analog signal을 출력한다.

마) IP Core

회로 구현 시 필요한 block RAM과 Option 1에서 필요한 Divider Generator를 생성하기 위하여 IP Core Generator를 사용한다.



< 그림 1. Overall block diagram >

2) Mode

Mode	ADDRESS (9bits)		Description
	Address Decoder Output Signal	Hex	
System Clock Setting	pcs_addr	140H~143H	8254 Setting
Software Reset	s_reset_addr	180H	Software Reset (모드 동작 중에도 reset)
PC Mode	pc_RAM0_addr	160H	wen, ren signal에 따라 RAM0에 읽기, 쓰기
	pc_RAM1_addr	162H	wen, ren signal에 따라 RAM1에 읽기, 쓰기
Data Transfer Mode	data_transfer_addr	164H	RAM0에서 RAM1로의 data 이동
DA Mode	da_start_addr	166H	RAM1에서 DA mode start
	da_stop_addr	168H	위의 DA mode DAC 출력을 끝냄
AD Mode	ad_RAM0_addr	16AH	RAM0의 chip select, AD mode start
Interpolation Mode (Option)	filter_addr	170H	RAM0에 저장된 신호를 Interpolation 동작을 수행하여 RAM1에 저장

< 표 1. Address memory map >

System clock setting, software reset은 mode가 아닌 회로의 동작환경을 구축하거나 특정 동작을 수행하기 위한 과정 또는 명령이다. Idle mode는 아무런 동작을 수행하지 않고 명령을 기다리는 대기 상태를 의미한다.

가) System Clock Setting (pcs_addr : 140H~143H)

pcs_addr signal을 발생하여 8254의 cs(chip select) signal을 인가한다. 총 3번으로 나뉘어져 들어오는 8254 programming data가 모두 입력되면 8254의 clk0에서 분주된 signal이 발생한다. **분주된 signal은 회로의 system clock으로 사용된다.**

나) Software Reset (s_reset_addr : 180H)

Software reset은 회로의 reset이 필요한 모든 부분을 reset한다. (8254는 reset 제외) Reset이 완료되면 idle mode로 돌아간다. 회로 동작 중 언제나 software reset address가 입력되는 경우 reset을 수행하도록 설계한다.

다) Idle Mode

Address를 기다리는 상태. Reset address를 받으면 이 mode로 돌아가도록 설계한다.

라) PC Mode

(1) PC RAM0 (pc_RAM0_addr : 160H)

RAM에서 data를 읽어 PC로 전송하거나, PC에서 전송한 data를 RAM에 저장한다. wen signal이 입력된 경우 PC로부터 전송된 data가 주어지는 만큼 RAM0에 저장한다. ren signal이 입력된 경

우 RAM에 존재하는 data가 주어지는 만큼 PC로 전달한다.

(2) PC RAM1 (pc_RAM1_addr : 162H)

위와 동일한 동작을 RAM1에서 수행한다.

마) Data Transfer Mode data_transfer_addr: 164H)

RAM0에 저장된 data를 RAM1으로 저장한다.

바) DA Mode

(1) DA Start Mode (da_RAM1_start_addr : 166H)

PC mode 또는 AD mode를 통해 RAM1에 저장된 data를 DAC로 보내 analog signal을 출력한다. 이 과정을 수행하기 전에 8254 setting을 바꾸어 system clock를 변화시켜 DAC 원하는 출력주파수를 설정하여 출력해본다. RAM1의 0번지부터 마지막 데이터가 저장된 번지까지 반복해서 data를 읽고 DAC로 보내며 동작 중 DA mode stop address가 들어오면 idle mode로 돌아간다.

(2) DA Stop Mode (da_stop_addr : 168H)

DA RAM1 Start Mode를 정지한다. DA RAM1 start mode 동작 중인 경우 본 address를 보내 동작을 중지하고 idle mode로 돌아간다.

사) AD Mode (ad_RAM0_addr : 16AH)

ADC에 입력된 signal을 RAM0에 저장한다. 외부의 ADC를 통하여 입력되는 waveform을 RAM0의 첫 주소부터 순차적으로 data가 주어지는 만큼 저장한다. 저장이 완료되면 idle mode로 돌아간다.

아) Interpolation Mode (filter_addr : 170H)

RAM0로부터 데이터를 불러와서 Interpolatino 동작을 수행한 후 이 결과를 RAM1에 저장한다. 저장이 완료되면 idle mode로 돌아간다.

3) Data Specification

USB-Logic		Controller-RAM		ADC/DAC	
Data Bus	8bit	Data Bus	8bit	Data Bus	8bit
Address Bus	9bit	Address Bus	10bit	Data Type	Unsigned Integer
Clock Frequency					
Clock Buffer를 통해 FPGA로 들어오는 signal의 frequency				40MHz	
8254를 통해 분주하여 system clock으로 사용하는 signal의 frequency				5MHz	
ADC conversion rate				40MHz	
DAC conversion rate				DAC 수행마다 가변	

다. 설계 참고 사항

1) Simple dual Port RAM

Block RAM Size 2kByte

The number of RAMs Required 2 (RAM0, RAM1)

IP Core Generator를 사용하여 생성할 block RAM은 address in/out, data in/out, WEA, write enable, read enable, clock in/out핀을 갖는다. 사용할 Single Port RAM은 synchronous이기 때문에 clock에 동기 되어 동작한다. 생성한 block의 pin의 vector size에 주의한다. (ex. WEA[0..0] -> std_logic이 아니고

std_logic_vector이다.)

Write enable signal이 들어온 경우 data in에 입력된 data를 입력된 주소에 쓴다. Read enable signal이 들어온 경우 입력된 주소에 있는 data를 data out으로 내보낸다. 두 control signal은 모두 active high로 동작하며 address는 10bit data는 8bit이다. 본 설계에서 생성할 RAM의 크기는 1Byte(8bit) * 2048 = 2048Byte이다.

2) Bus control

본 설계에서 구현한 회로에서는 data bus, address bus, 2개의 bus가 필요하다. Bus가 동시에 두 개 이상의 component에 의해 사용되어 충돌이 일어나는 것을 방지하기 위하여 Mux, tri-state buffer 등의 component로 control할 필요가 있다.

가) Mux

Mux(Multiplexer)는 여러 개의 입력 중 하나를 선택하여 출력으로 내보내는 기능을 가지고 있다. 이 기능을 사용하면 bus에 넣을 여러 data 중 원하는 것을 선택하여 넣을 수 있다.

나) Tri-state buffer

Tri-state buffer는 enable 신호에 따라서 출력이 입력의 bypass 혹은 hi-z가 되는 기능을 가지고 있다. 이러한 기능은 in/out port에서 data의 방향을 정할 때 사용할 수 있다. 본 설계에서는 USB 통신 module과 FPGA간의 data bus에서 사용하고 있다.

3) 8254

8254는 timing과 counting 기능을 가진 PIT(Programmable Interval Timer)이다. Control word에 따라서 clock 신호를 분주하거나 rate pulse로 만들어 출력한다. 본 설계에서는 외부에서 입력된 clock을 분주하여 system clock으로 만들기 위해 사용된다. 자세한 설정 방법은 보충 강의 자료에 설명되어 있다.

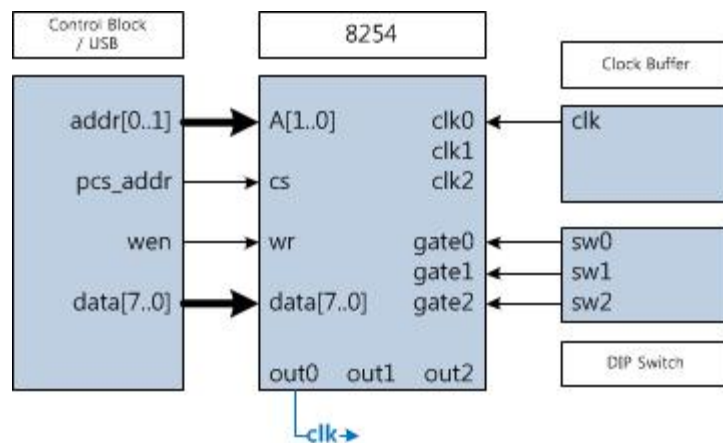
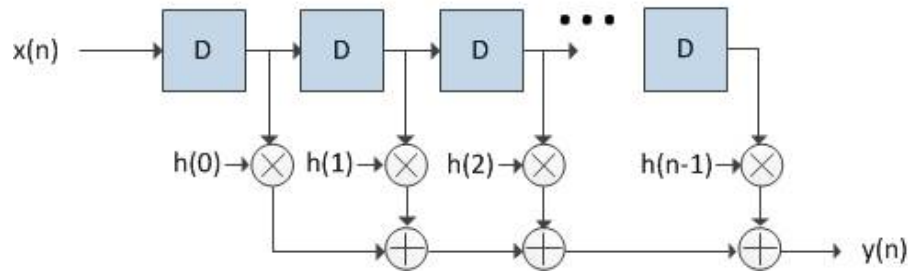


그림 3 8254 연결 구성도

Top(회로 최상위 계층 component)에서 사용할 8254 component는 기타 설계에 필요한 자료들과 함께 제공된다. 제공된 8254를 top에서 component instantiation (component declaration과 port map 설정)을 통해 회로 내부에 넣고 그림 2 또는 그림 3을 참고하여 다른 component와 port에 연결한다. Clock buffer를 거쳐 들어온 signal은 clk0 pin으로 들어오고 control word에 따라서 out0으로 분주된 signal이 출력된다.

4) Filter

Filter는 입력 데이터와 필터 coefficient 간 Convolution 연산을 수행한다. 이를 위해, 필터는 기본적으로 다음과 같은 구조를 가진다.



입력 $x(n)$ 은 clk입력 마다 D flip-flop으로 Shift되고 이 값들은 각각 필터 coefficient $h(0)$, $h(1)$, ... $h(n-1)$ 과 곱해진다. 곱해진 값을 모두 더하여 출력 $y(n)$ 을 얻을 수 있다.

5) FPGA 입, 출력 핀

Pin Name	Bit Width	Description
Input		
Address	9	회로의 mode를 지정할 때 또는 8254에 control word를 쓸 때 사용
CMD_data	1	Address signal의 enable 역할
wen	1	회로 전반적인 동작에서 write enable 역할
ren	1	회로 전반적인 동작에서 read enable 역할
clk0~2 (8254)	1	8254 내부의 각 counter에 입력되는 clock
gate0~2 (8254)	1	8254 내부의 각 counter의 enable signal
ADC data	8	ADC로 입력된 신호가 unsigned integer로 sampling 된 것
Output		
Data (DAC)	8	Analog signal을 생성하기 위해 DAC로 보내는 data (unsigned integer)
Clock (DAC)	1	DAC의 clock (ADC와는 달리 FPGA에서 clock 인가)
ready	1	Full-handshake 방식의 USB 통신을 하기 위해 필요한 signal
Input/Output		
Data (PC)	8	PC와 FPGA간의 data 전송을 위한 signal

6) Latch 사용

외부 소자와 회로의 온전한 데이터 입/출력을 위해서는 latch를 사용하는 것이 권장된다. Latch는 본 설계에서 Data in/out bus, ADC, DAC data bus에서 사용된다.

7) Pin, Signal Naming Convention

본 설계 자료에서 제공한 8254 또는 다른 VHDL code를 보면 port name, signal name에 $m_$, $s_$, $_b$ 등이 붙어있다. $m_$ 은 module을 뜻하며 주로 component의 pin name 앞에 붙여 해당 단어가 어떤 module의 pin 임을 쉽게 알 수 있게 한다. $s_$ 는 내부 signal의 이름 앞에 붙이고 목적은 같다. $_b$ 는 pin, signal 관계없이 이름 뒤에 붙이는데, 그 뜻은 bar(not)이다. 이것은 active low 신호 혹은 inverted signal을 표현하는 데 사용된다.

라. 설계 제한 요소

1) 경제요건

State diagram 작성 시 최소한의 state 수를 사용한다.

2) 안정성 및 신뢰성

가) PC mode에서 data 입/출력시 정해진 수만큼의 data가 RAM에 안정적으로 저장되고, PC로 출력되도록 한다.

나) DA mode에서 RAM에 저장된 data를 정해진 수만큼 DAC로 출력하여 파형이 안정적으로 출력되도록 한다.

다) [Optional] AD mode에서 정해진 수만큼의 data가 안정적으로 RAM에 저장되도록 한다.

3) 사회적 영향

해당 설계 과제가 실제 사회에서 응용되는 예와 사회에 어떠한 영향을 미칠 수 있는지를 파악한다.

4. 설계 과정

가. 목표 및 기준설정

1) 설계 배경지식 습득

제시된 설계 내용과 8254 동작 원리(timing diagram), Quick USB 통신 프로토콜, 회로로 구현에 필요한 이론(VHDL, State machine 등)을 이해한다.

2) Block Diagram, Timing Diagram 작성

제시된 회로의 동작을 세부 동작으로 구분하여, 각 세부 동작을 수행하는 블록으로 나타낸다. 각 블록 간에 전달해야할 data 또는 신호가 어떠한 것들이 있으며, 이들이 어떠한 순서로 전달되는 지 고려한다. 앞서 고려한 사항을 바탕으로 기능(mode)별 timing diagram을 완성한다.

나. 합성 및 분석

작성한 timing diagram대로 신호를 발생하기 위해 state machine을 사용하여 controller 회로를 구현한다. 각 block과 controller를 VHDL로 구현한다. 구현한 block들을 포함하는 최상위 block을 만들고 그것으로 FPGA configuration file을 만든다. FPGA configuration file(bit file)을 만드는 과정은 설계 과제 진행 중 공지된다.

다. 시험 및 평가

전체 회로가 완성되면 simulation을 통해 동작을 검증한다. simulation에서 이상 동작이 발견되면 문제점을 찾아 해결한다. 문제가 발생하지 않을 때 까지 simulation, debugging 과정을 반복한다.

라. 결과 도출

구현한 회로를 FPGA에서 동작시켜 안정된 결과를 출력하는지 확인한다. 만약 시뮬레이션에서 성공하였는데 FPGA 상에서 실패했다면 그 이유를 분석하고 다시 회로를 구현한다.

마. 설계고려사항

최소한의 logic을 사용하여 설계하도록 노력하여야 하며, Top-down 방식으로 설계하되 각 블록의 설계를 검증한 후 전체 블록을 집적하여 설계를 완성한다. State diagram으로 회로 구현 시, 구현의 편리성과 가독성을 위해 one hot state를 사용한다. 각 mode 동작 시 임의의 test vector에 대해서 안정된 동작을 하도록 설계한다. (ex. 잘못된 address 무시, 동작 중의 다른 address 입력 무시 등의 예외 처리)

(※ Test Vector : 회로의 동작 검증을 위한 임의의 입력 값 / Vector : 여러 개의 bit)

5. 설계 평가

가. 회로 검증 (Demonstration)

구현한 회로의 FPGA configuration file(bit file)을 사용하여 설계한 회로를 평가한다. 본 설명서에서 제시된 기능이 온전히 구현되었는지 평가한다.

회로 검증은 8254 setting으로 system clock을 생성하고, software reset으로 회로의 동작 환경을 초기화시킨 후 진행한다. 8254 setting 또는 software reset 기능이 동작하지 않는 경우 회로 검증을 할 수 없으므로 실패한 것으로 간주한다.

1) PC Mode

data가 RAM0, 1에 온전히 입/출력 되는지 확인한다.

2) Data Transfer Mode

PC mode 또는 AD mode를 통해 회로에 저장된 신호가 RAM0에 저장되고, RAM0에 저장된 data가 RAM1으로 이동하는지 확인한다.

3) DA Mode

RAM1의 data가 DAC를 통해 출력되어 파형으로 나타나는 지 확인한다.

4) AD Mode

ADC를 통해 data가 저장된 것을 PC mode와 DA mode를 통해 확인한다.

※ 아래 mode에 대한 평가는 나머지 mode가 전부 동작 했을 때만 진행한다.

5) Interpolation Mode

RAM0에서 데이터를 읽어 Interpolation 모드를 수행한 후 RAM1에 저장하고, PC모드를 통해 RAM1에 데이터가 정확한지 확인한다.

나. 결과물 제출

1) FPGA configuration file

추후 공지될 제출일까지 조교의 email로 email 제목과 file 제목을 형식에 맞춰 제출한다. Email 제목은 DP1_BIT_조번호_학번이름1_학번이름2 형식으로 정하고, FPGA configuration file 제목은 DP1_BIT_조번호_학번이름1_학번이름2.bit 형식으로 정한다. soft copy와 ISE project는 검증 당일 본인이 지참한다.

조교 email 주소

swahn08@gmail.com

Email 제목 예시

DP1_BIT_01_2011XXXX현대차_2011XXXX기아차

bit 파일 제목 예시

DP1_BIT_01_2011XXXX강정호_2011XXXX박병호.bit

2) 보고서

회로 검증 성공한 날짜를 기준으로 3일 내에 조교의 email로 레포트 soft copy를 제출한다. 메일 제목은 DP1_조번호_학번이름1_학번이름2으로 정한다. 보고서 soft copy는 아래아 한글 또는 마이크로소프트 워드 형식을 사용한다. 보고서 file의 제목은 DP1_조번호두자리_학번이름1_학번이름2.확장자 형식으로 정한다. 보고서 file은 ISE project folder와 같이 zip 형식으로 압축하여 제출한다. Hard copy는 추후 공지일 까지 R관 806호로 제출한다.

보고서 및 프로젝트 폴더 제출 기한 예시

화요일 성공 -> 금요일 자정까지 제출

Email 제목 예시

DP1_RESULT_01_2011XXXX손흥민_2011XXXX헤리케인

6. 보고서 작성요령

1. 제목
2. 목적
3. 목표 및 기준 설정 - 설계 과정1
4. 합성 및 분석 - 설계 과정2
5. 시험 및 평가 - 설계 과정3
6. 결과 도출 - 설계 과정4
7. 토의
 - 1) 제시된 주요 설계 요소와 제한요소들을 만족하는 설계를 하였는지 논의
 - 2) Top-down 방식으로 설계하였는지 논의
 - 3) 임의의 입력에 대해서도 안정적으로 동작하는 논의
 - 4) logic량은 최적화 되었는지 논의
 - 5) one hot state로 회로를 구현하였을 때의 장점에 대해서 논의
 - 6) 해당 설계 과제가 실제 사회에서 응용되는 예와 사회에 어떠한 영향을 미칠 수 있는지를 파악
 - 7) 설계결과를 실제 제품개발에 사용될 수 있는지를 검토한다.