1주차 예비보고서

전공 : 국제한국학과 학년 : 4학년 학번 : 20181202 이름 : 김수미

**1. FPGA란 무엇인지, 그 활용법/사용법을 조사하시오.**

**1) FPGA란 무엇인가?**

FPGA는 Field-Programmable Logic Array의 약자이며, 1985년 Xlinx사에 의해 처음 소개되어 만들어졌다. 말 그대로 프로그램 가능한 논리 소자 배열의 일종으로 사용자가 논리회로를 의도에 맞게 재설정할 수 있는 반도체이며, 이미 설계된 하드웨어를 반도체로 생산하기 직전 최종적으로 하드웨어의 동작 및 성능을 검증하 기 위해 제작하는 중간 개발 물 형태의 집적 회로(IC)이다. 필요하지 않은 게이트가 많이 있어 보통 반도체보다는 큰 편이지만 사용 도중 요구사항이 바뀌면 논리회로를 새로 적용할 수 있기 때문에 반도체 개발 초기 단계에서 중간과정으로 많이 사용한다.

FPGA 기업들은 프로그래밍 소프트웨어를 제작하며, 사용자는 해당 소프트웨어를 이용해 프로그래밍을 진행한 다음 이를 FPGA에 다운로드하여 사용하게 된다.

**2) FPGA 활용법 및 사용법**

AND, OR, XOR, NOT 외에 더 복잡한 디코더나 계산 기능의 ‘조합’ 같은 기본적인 논리 게이트의 기능을 복제하여 프로그래밍할 수 있으며 대부분의 FPGA는 프로그래밍가능 논리 요소에 간단한 filp-flop이나, 메모리 블록으로 된 메모리 요소를 포함하고 있다**.**

FPGA을 위한 프로그램을 작성하기 위해 HDL(Hardware Description Language)을 사용해야 하는데, 가장 대표적인 HDL은 Verilog와 VHDL이다. 이러한 HDL을 이용해 소스코드를 작성한 후 이를 비트 파일로 변환시켜 FPGA를 구성하게 할 수 있다.

초기 설계 입력

**그래픽적인 입력** : schematic capture tool을 통해 논리 회로를 입력한다. 내부의 라이브러리들은 FPGA의 칩 구조를 기반으로 하여 미리 설계된 회로들로 구성되어 있기 때문에 안정적이고 효율적이다.

**문자적인 입력** : 그래픽적인 입력 외에도 부울식, HDL, VHDL, Verilog HDL에 의한 입력을 사용한다. 그래픽적인 입력으로만 프로그래밍을 하면 전체적인 논리를 파악하기 힘들 수 있어 문자적인 방식도 자주 사용한다.

RTL simulation (register-transfer level simulation)

초기 설계 입력 단계에서 사용자가 구현한 것을 simulation으로 검증하는 단계이다.

register란 보통 filp-flop으로 구성되며 clock 신호가 들어가서 회로를 동기화하는 역할을 한다. 메모리처럼 데이터를 읽고 쓸 수 있다.

HDL 코드에서는 이러한 register들을 마치 변수처럼 선언해서 사용한다. 그래서 HDL로는 실제 디지털 회로가 어떻게 구현될지 알기 어렵지만 register level에서는 대략적으로 알 수 있다. 뒤에 이어질 synthesis 과정을 거친 뒤에 FPGA 보드에 넣어서 테스트하기도 한다. 하지만 synthesis 시간이 오래 걸리고 디버깅이 어렵기 때문에 합성 전에 테스트 과정을 거친다. 또한 간혹 RTL simulation 과정을 정상적으로 통과해도 합성이 안 되는 경우가 있어 synthesis 이후에도 테스트를 할 필요가 있다.

synthesis

유저가 구현한 high-level 디자인을 FPGA 보드가 이해할 수 있는 low level로 바꿔주는 단계이다. HDL 코드를 합성해서 netlist 형태의 data로 만든다.

Place & Route

실제적인 회로의 구현 형태가 FPGA의 칩 상에 이루어지는 단계이다.

**Place (배치)** : mapping된 block들을 FPGA의 특정 logic block 위치에 지정한다.

**Route (배선)** : 배치된 각 block들을 회로의 연결 정보에 따라 연결 연결선 및 전기적 스위치를 사용하여 상호 연결한다.

설계 검증 및 성능 추정

P&R이 끝나고 실제 회로의 형태를 가지고 있는 상황에서 timing simulation을 진행한다.

P&R 직후에 cell들을 연결하면 wire의 길이가 달라질 수 있으며 이 때 delay 문제가 생길 도 있는데, 이러한 timing error 없이 회로가 잘 돌아가는지를 검증하는 과정이다. 또한 특정 조건에서 performance를 어떻게 발휘하는지도 측정하고, 추후 어떤 성능을 가질지 추정한다.

**2. FPGA 의 장단점 및 활용 분야를 조사하시오.**

**1) FPGA의 장점**

속도 : 하드웨어의 병렬성을 기반으로 하기 때문에 순차적이지 않으며 clock 주기당 더 많은 작업 수행을 한다. 따라서 속도가 빠르며 신호 처리가 중요한 분야에서 주로 사용한다.

안정성 : 프로세서 기반의 시스템은 어떠한 프로세서 코어든지 하나의 task를 한 번에 실행하며 시간 결정적 task의 위험 상태에서 서로를 선점한다. 이 때문에 하드웨어 안정성 문제를 최소한으로 줄여준다.

시장 출시 기간 : ASIC 디자인과 비교하여, ASIC는 긴 제조 과정을 거쳐야 하지만 FGPA는 그러지 않고도 아이디어나 개념을 하드웨어에서 검증할 수 있다. 또한 상용 하드웨어는 FPGA 칩에 연결된 다른 유형의 I/O와도 사용할 수 있다. 이러한 검증 기간의 단축과 범용성 덕분에 시장에 출시되기까지의 기간이 ASIC보다 짧다.

비용 : 여러 유저들은 맞춤 하드웨어 기능을 필요로 한다. FPGA는 요구에 따라 점진적으로 변경해 나가는 비용이 ASIC를 새로 구축하는 것보다 적게 들어간다.

장기 유지 관리 : 프로그래밍이 가능하며 현장에서 바로 업그레이드가 가능한 FPGA의 특성상 재설계를 할 필요 없이 바로 필요한 만큼 업그레이드를 하면 되기 때문에 장기적으로 유지할 수 있다는 장점이 있다.

**2) FPGA의 단점**

대량 생산 시 비용 문제 : 한 번 구축하면 양산할 때의 비용은 적은 ASIC에 비해 FPGA의 양산 단가는 비싸다.

칩의 면적 : 칩의 면적이 커서 사이즈가 작은 제품은 FPGA로 설계하기 어렵다.

**3) 활용 분야**

우리는 제품을 개발할 때 제품 설계를 완성한 다음 하드웨어를 바꾸지 않아도 기능을 업데이트 할 수 있다. 이러한 것은 대부분 마이크로프로세서를 사용한 시스템에서 ROM의 프로그램을 업데이트 하는 과정을 거친다. 그런데 프로그램 이외에 IC(칩) 자체의 기능 역시 업데이트가 가능하다. 이를 가능하게 하는 것이 바로 FPGA 이다.

최근 의료 바이오 시장에서 FPGA는 박테리아, 바이러스, 암 등을 검진하고 치료에 활용할 수 있는 Lap-On-a- Chip, PCR과 같은 체외 진단 기기의 기능 집약화를 통한 소형 임베디드 장치와 FPGA 하드웨어를 적용하는 분야, 그리고 PC기반 의료영상진단기기의 영상처리 속도 향상, 정밀한 레이저 제어를 통한 치료기기, 수술용 로봇 시스 템 등에서 사용되고 있다.

이 외 FPGA는 로봇제어(알파고), 우주선, 인공위성 등 압도적으로 빠른 하드웨어 프로그래밍을 요구로 하는 현장에서 주로 사용한다.