

REPORT



부산대학교
PUSAN NATIONAL UNIVERSITY

텀프로젝트 제안서 : 공중전화

목차

1. 프로젝트 목표	3
2. 주제 및 기능	3
- 주제	
- 기능	
3. 구현 내용 및 방법	4
- 컨트롤 유닛	
- 타이머	
- 레지스터	
- 인코더와 디코더	
4. 역할 분담	6
5. 개발 일정	6

1. 프로젝트 목표

- 조합회로와 순차회로로 모듈을 설계할 수 있다.
- Moore model을 활용한 FSM으로 컨트롤 유닛을 설계할 수 있다.
- 모듈과 컨트롤 유닛을 이용하여 공중전화의 회로를 설계하고, 이를 구현할 수 있다.

2. 주제 및 기능

2.1 주제

논리회로 설계 및 실험 과목의 목표는 다양한 논리회로를 설계해 회로의 동작을 확인해보고, 결과적으로 FPGA 보드에 결과물을 구현하는 것이다.

이번에 선정한 과제는 공중전화 설계인데, 이를 구현하기 위해서 가감산기, 타이머, 시프트 레지스터, 메모리 등 다양한 논리회로를 사용해 모듈을 만들고 이를 컨트롤 유닛으로 제어할 것이다. 그 후 설계한 회로를 FPGA 보드로 구현하는데, 버튼을 통한 입력과 7-세그먼트, LED를 통한 출력을 보일 것이다.

2.2 기능

일반적으로 우리가 사용하는 공중전화는, 전화하기 위해 일정 금액 이상 동전을 넣은 후 원하는 전화번호를 누르고 통화하는 과정을 따르게 된다. 우리는 여기에 추가로, 원래의 기능에는 없는 기능인 이전 번호 조회 기능과 현재 상태를 편하게 확인하기 위한 장치인 LED를 추가하였다. 아래는 각 동작 별 기능을 나타낸 것이다.

1) 동전 넣기

공중전화에 들어가면 수화기를 들고 동전을 투입한다. 이번에 구현할 때는 키패드 1번(10원, 2초), 2번(50원, 10초) 버튼에 동전을 투입하는 행동을 대신하도록 했다. 투입한 금액은 7-세그먼트에 표시되며 80원 이상 동전을 넣고 0번을 눌러 동전 투입을 완료한다.

2) 전화하기

동전이 투입되면 전화번호를 누를 수 있다. 키패드 0~9번으로 전화번호를 입력한다. 8개의 숫자를 입력할 수 있고, 입력 후 #을 누르면 통화할 수 있다. 입력한 번호는 7-세그먼트에 표시된다.

3) 전화 중 잔여 금액 처리하기, 통화 종료

잔여 금액이 타이머에 따라 일정 시간마다 감소한다. 잔여 금액은 7-세그먼트로 표시되며 잔여 금액이 0이 되거나, 사용자가 수화기를 놓았을 때 종료되며, 여기서는 키패드 0을 눌렀을 때 수화기를 놓는 행동을 대신하도록 했다. 통화가 종료된 후에 동전 투입 전 상태로 돌아간다.

4) 이전 전화번호 조회

통화 종료 후 마지막으로 전화한 번호를 자동으로 저장하는 기능이다. 통화 전 상태에서 #을 누르면 7-세그먼트로 확인할 수 있고 0을 누르면 원래 상태로 돌아간다.

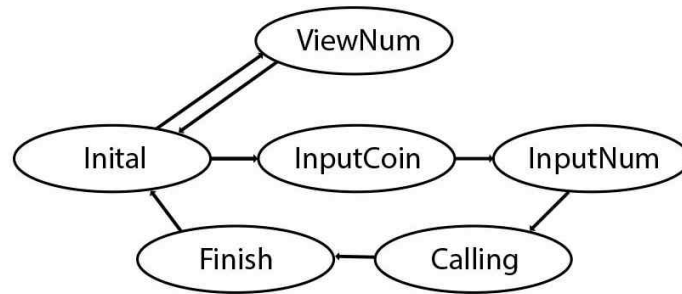
5) 현재 상태 알기

일반적인 공중전화에서는 지금이 무슨 상태인지 확인이 쉽지 않다. 이를 개선해서 LED로 현재 상태를 알도록 한다. 기본(동전 투입 전) 상태에는 LED 1, 동전을 투입하고 있는 상태는 LED 2, 동전 투입을 마치고 전화번호를 누르는 상태에서는 LED 3, 통화 중인 상태는 LED 4, 통화가 종료된 뒤 잠깐 LED 5, 전화한 번호를 확인할 때는 LED 6이 켜진다.

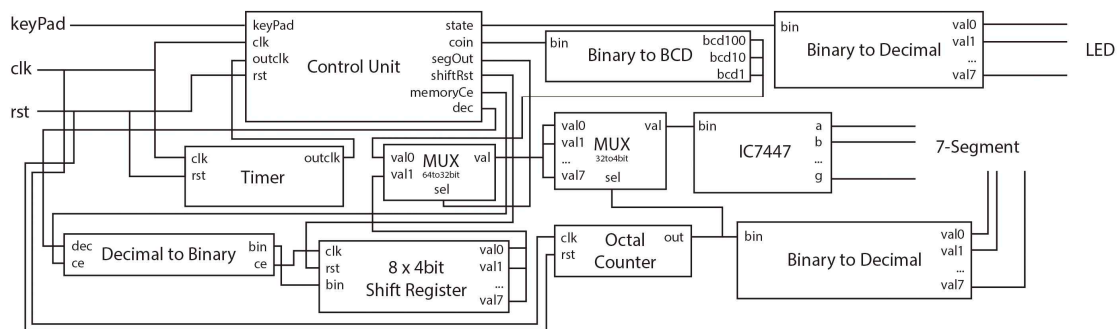
3. 구현 내용 및 방법

3.1 컨트롤 유닛

컨트롤 유닛은 데이터 처리를 할 수 있도록 제어 신호를 공급한다. 현재는 무슨 상태인지, 어떤 조건을 만족했을 때 다른 상태로 넘어가기 위한 동작이 필요하고, 데이터의 흐름을 조절하고 시스템의 정해진 작업을 수행하기 위해 사용한다. 이는 Stadian FSM으로 설계한다. 대략적인 컨트롤 유닛의 상태도는 다음과 같다.



또한, 개략적인 데이터패스 설계도는 다음과 같다.



3.2 타이머

FPGA 보드에서 클럭은 기본적으로 제공하는 클럭을 사용한다. 잔여 금액 감소에 대해 1초에 5원이 감소한다고 할 때 그 1초를 기본적인 클럭을 이용하기 매우 어렵다. 이를 해결하기 위해 1초마다 1클럭이 올라가는 타이머가 필요하다. 클럭의 주파수가 1Mhz이라고 할 때 1초를 정확하게 카운트하기 위해서는 1,000,000진 카운터를 이용하면 되지만, 편의를 위해 $2^{20}(1,048,576)$ 진 카운터를 설계해 모든 out이 1이 되었을 때 Outclk이 올라가도록 한다.

3.3 레지스터

8자리의 전화번호를 차례대로 입력하고 입력한 번호를 순서대로 저장하기 위해 시프트 레지스터가 필요하다. 여기서 하나의 숫자에 4bit에 8자리이므로 8 x 4bit 시프트 레지스터를 사용한다.

3.4 인코더와 디코더

숫자를 입력받고 이를 메모리에 저장할 때 인코더를 통해 2진 코드로 변환해야 하는데 이는 인코더를 통해 인코딩하고 이를 저장한다. 또한, 지금이 어떤 상태인지 표시를 위한 LED는 FSM 설계의 연장으로 컨트롤 유닛에서 나오는 현재 상태를 입력으로 받아 들어와야 할 LED를 Binary로 받아 Decimal로 변환하는 디코더, 전화번호 및 잔여 시간 확인을 위한 세그먼트 표현은 Binary를 입력으로 받아 7-세그먼트에 대한 a~g를 출력으로 갖는 디코더를 설계해 해결할 수 있다.

4. 역할 분담

	FSM 설계
	타이머 설계
	가감산기 설계
	다비트 MUX 설계
	모듈 연결 및 동작 확인
	FPGA 보드 동작 확인

5. 개발 일정

	1주차	2주차	3주차	4주차
일반 모듈 설계				
LED, 세그먼트 제어				
컨트롤 유닛 FSM 설계				
모듈 연결 및 동작 확인				

사용하는 모듈이 많으므로 먼저 각각의 일반 모듈의 설계와 키패드, 7-세그먼트, LED를 제어하는 모듈을 2~3주 이내에 설계한다. 그 후 설계한 회로를 제어하는 컨트롤 유닛을 설계하고, 이 모듈을 모두 연결함으로써 하나의 큰 회로를 완성한다. 마지막으로 FPGA 보드에 파일을 업로드해 동작을 확인한다.