

10 조 텁프로젝트 제안서

팀원

톈 흐닌 유 웨이 (202355677)

얏 뚜 카잉 (202355672)

목차

1. 프로젝트 제목 및 목표

2. 주제 및 기능

 2.1 전체 시스템 아키텍처

 2.2 계산기

 2.3 수학 퍼즐 게임

3. 구현 내용 및 방법

 3.1 계산기 설계도 및 알고리즘

 3.2 수학 퍼즐 게임 설계도 및 알고리즘

4. 개발 일정

1. 프로젝트 제목 및 목표

DIP 스위치 및 버튼 스위치를 이용한 다중 모드(계산기, 게임) 시스템 설계

이 프로젝트는 논리회로설계 및 실습에서 배운 Verilog HDL과 디지털 시스템 설계 지식을 실제 FPGA 시스템에 적용하는 것을 목표로 합니다. 구현할 시스템은 8 비트 DIP 스위치를 사용하여 계산기, 퍼즐 게임이라는 두 가지 주요 모드 중 하나를 선택할 수 있습니다. 메인 모드가 선택된 후에는 12 개의 버튼과 DIP 스위치의 조합을 통해 각 모드의 세부 기능(계산기의 연산, 게임의 조작)을 제어하게 됩니다. 또한 16x2 텍스트 LCD를 사용하여 현재 상태나 게임 화면을 표시하고, Peizo를 통해 소리 피드백을 제공하는 복합적인 디지털 시스템을 설계하고 구현합니다.

2. 주제 및 기능

2.1 전체 시스템 아키텍처

본 시스템은 FPGA 보드 위에서 계산기, 수학 퍼즐 게임의 두 가지 모드를 실행하도록 설계되었다. 사용자는 DIP 스위치(DIP SW[2:1]를 이용해 모드를 선택하고 버튼과 스위치로 입력하며 LCD·LED를 통해 결과를 출력한다.

- **입력 장치(Input Unit):** 버튼(BTN[12:1]), DIP 스위치(SW[8:1])를 통해 사용자 입력을 수신한다.
- **제어 장치(Control Unit):** DIP 스위치를 해석해 모드를 결정하고 각 모듈의 동작을 제어한다.
 - a. 00 → 계산기 모드 (Calculator Mode)
 - b. 01 → 퍼즐 게임 모드 (Puzzle Game Mode)
- **연산 및 게임 로직(Processing Unit):** 계산기, 퍼즐 게임의 로직을 독립 모듈로 구성한다.
- **출력 장치(Output Unit):** LCD로 결과·게임 화면·점수를 표시하고 LED를 통해 시각적 피드백을 제공한다.

2.2 계산기 모드 (Calculator Mode)

2.2.1 주제

이 모드는 FPGA 보드를 이용해 DIP Switch[2:1]를 00로 입력하는 경우 기본 사칙연산을 수행하는 계산기를 구현하는 것이다. 버튼과 DIP 스위치를 통해 피연산자 및 연산자를 입력받고 결과를 LCD에 표시한다. 입력, 제어, 연산, 출력 모듈을 결합하여 완전한 계산기 시스템을 구성한다.

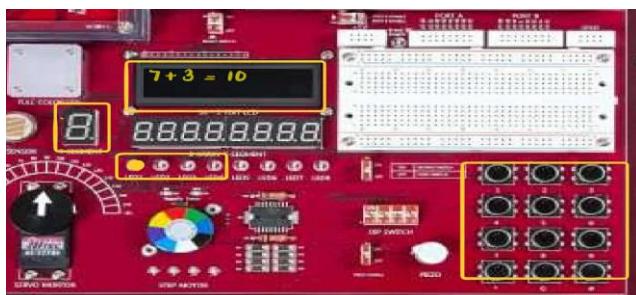
2.2.2 기능

계산기 모드는 두 개의 피연산자와 연산자를 입력받아 결과를 계산하는 기능을 수행한다. 사용자는 버튼으로 숫자를 입력하고 입력한 숫자 하나씩 LCD와 7-segment Display에 표시되면서 DIP 스위치를 통해 연산자를 선택한다. 연산자에 따라 LED 불이 커지면서 Enter 버튼을 입력 시 결과가 LCD에 표시되며 잘못된 입력은 Backspace 버튼으로 수정할 수 있다. 덧셈(00), 뺄셈(01), 곱셈(10), 나눗셈(11)의 기본 연산을 지원한다.

- 입력 기능 (Button Switch)



먼저 사용자는 0부터 9까지의 숫자 버튼을 이용하여 첫 번째 숫자를 입력한다. 그 후 DIP 스위치(SW[4:3])를 이용하여 연산자를 선택하며 00은 덧셈(+), 01은 뺄셈(-), 10은 곱셈(x), 11은 나눗셈(÷)을 의미한다. 선택된 연산자에 따라 해당 LED 가 점등되어 현재 연산 상태를 표시한다. 다음으로 사용자는 두 번째 숫자를 입력하고 Enter 버튼을 누른다. 입력 과정에서 LCD에는 입력된 숫자와 연산자가 순서대로 표시되며, 7 세그먼트 디스플레이에는 사용자가 마지막으로 입력한 숫자가 나타난다. Enter 버튼을 누르면 연산이 수행되어 LCD에 결과가 출력된다.



2.3 수학 퍼즐 게임 모드 (Puzzle Game Mode)

2.3.1 주제

이 모드는 FPGA를 이용해 사용자가 DIP Switch[2:1]를 01로 입력하는 경우 산수 퀴즈 형식의 수학 퍼즐 게임을 구현하는 것이다. 사용자는 LCD에 표시된 문제의 정답을 입력하며 제한 시간 내에 올바른 답을 제출해야 한다. 게임은 Easy, Medium, Hard의 3 단계 난이도 모드를 제공하여 사용자가 DIP 스위치를 통해 난이도를 선택할 수 있다. 각 문제는 문제 생성, 정답 검증, 점수 관리, 시간 제어 모듈로 구성되어 있으며 FPGA의 병렬 처리 특성을 활용하여 실시간으로 문제 출제와 피드백을 수행한다.

2.3.2 기능

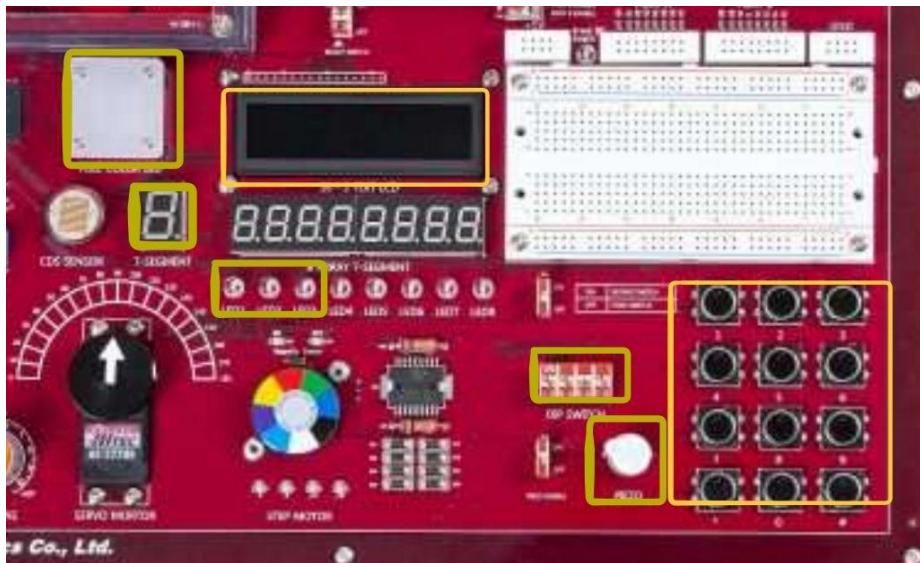


첫 번째 단계에서는 사용자가 게임 시작 전 닉네임을 입력한다. 키패드를 이용하여 알파벳을 입력하고 입력이 완료되면 Submit 버튼을 눌러 닉네임을 확정한다. 입력 중에는 LCD 1 행에 “Enter Nickname:”이라는 문구가 표시되며 사용자가 입력한 문자가 실시간으로 LCD에 반영된다.

두 번째 단계에서는 난이도(Easy / Medium / Hard)를 선택한다. 난이도는 1, 2, 3 버튼을 통해 설정되며 선택된 난이도에 따라 문제의 복잡도와 연산 기호(+, -, ×, ÷)가 달라진다. 예를 들어 Easy 모드에서는 덧셈과 뺄셈 위주로, Hard 모드에서는 곱셈과 나눗셈을 포함한 복합 문제가 출제된다.

- 1 → Easy
- 2 → Medium
- 3 → Hard

LCD 1 행에는 무작위로 생성된 문제가 표시되고 사용자는 버튼으로 정답을 입력한 뒤 Submit 버튼으로 제출한다. 각 문제는 5 초의 제한 시간이 주어지며, 남은 시간은 7-세그먼트 디스플레이에 5 부터 0 까지 표시됩니다. 5 초가 지나면 자동으로 오답 처리됩니다. 정답 여부에 따라 점수와 남은 생명(3 개 기준)이 LCD 에 표시되고, 오답 시 생명이 1 개씩 감소한다. 정답을 입력하면 Peizo 에서 승리음을 출력하고 Full Color LED 가 녹색으로 점등되어 정답임을 시각적으로 알린다. 오답일 경우 Peizo 에서 경고음이 울리고 Full Color LED 가 빨간색으로 점등된다. 또한 생명은 LED 1, 2, 3 번에 표시되며 오답 시 LED 가 하나씩 꺼져 남은 생명을 시각적으로 확인할 수 있다. 모든 생명이 소진된 상태에서 다시 오답을 입력하면 **Game Over** 가 되며 LCD 에는 "Game Over"와 함께 사용자의 닉네임과 최종 점수가 표시된다. 재시작을 위해 DIP Switch[8]를 통해 reset 하고 게임을 다시 시작할 수 있다.



위 사진은 우리가 FPGA 보드에서 사용하기로 선택한 부품들을 보여줍니다.

3.1 계산기

3.1.1 설계도



3.1.2 알고리즘

1. 첫 번째 숫자 입력 단계

- 사용자가 Keypad 로 첫 번째 숫자를 입력한다.
- 입력된 숫자는 LCD 와 7-Segment 에 동시에 표시된다.

2. 연산 선택 단계 (DIP Switch[3:4])

- 00: 덧셈 (+)
- 01: 뺄셈 (-)
- 10: 곱셈 (x)
- 11: 나눗셈 (÷)
- 선택된 연산에 따라 LED 가 점등된다.

3. 두 번째 숫자 입력 단계

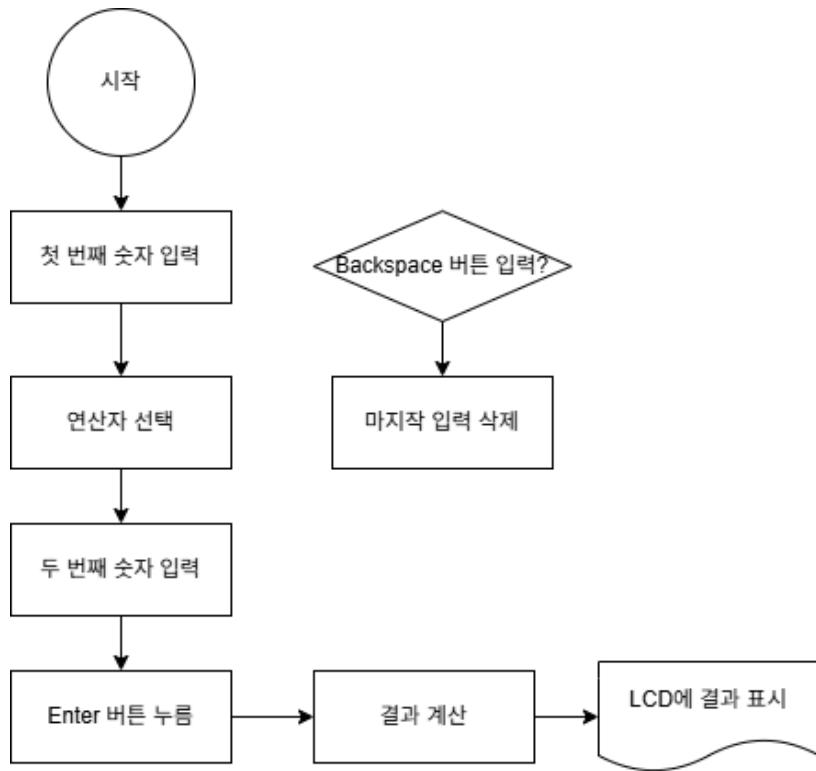
- 사용자는 두 번째 숫자를 Keypad 로 입력한다.
- LCD 에 실시간으로 표시된다.

4. 결과 계산 단계 (Enter 버튼)

- Enter 버튼을 누르면 ALU 에서 연산 수행
- 계산 결과가 LCD 에 “A + B = 결과” 형태로 표시된다.

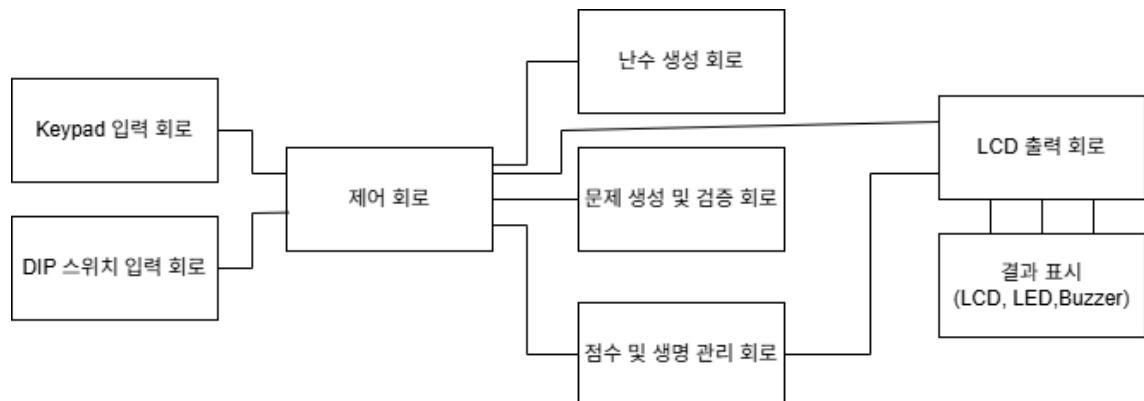
5. 수정 기능 (BTN_Backspace)

- 사용자가 잘못 입력한 숫자를 삭제할 수 있다.
- LCD 표시 내용이 즉시 갱신된다



3.2 수학 퍼즐 게임

3.2.1 설계도

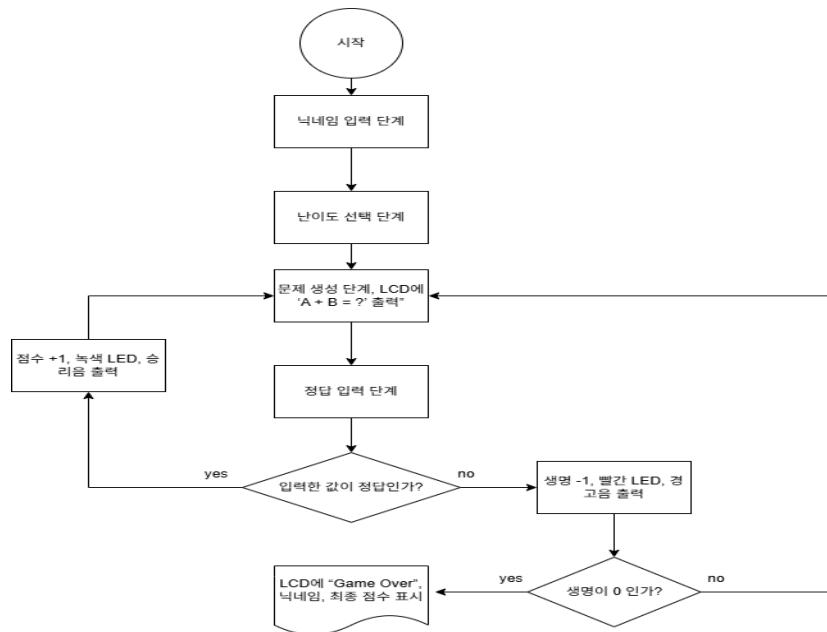


3.2.2 알고리즘 (Algorithm)

1. 닉네임 입력 단계 - Keypad 로 문자를 입력하여 LCD 1 행에 표시, 'submit' 버튼을 누르면 닉네임 입력 완료
2. 난이도 선택 단계 - 1,2,3 버튼으로 Easy / Medium / Hard 선택, 난이도에 따라 문제의 연산 종류 및 숫자 범위 결정
3. 문제 생성 단계 - 내부 난수 생성기를 통해 두 개의 피연산자와 연산자(+, -, ×, ÷) 생성, LCD 1 행에 "A + B = ???" 형태로 출력
4. 정답 입력 단계 - 사용자는 Keypad 로 정답을 입력하고 LCD 2 행에 표시, 'Enter' 버튼으로 제출
5. 시간 제한 단계 - 5 초 카운터 시작, 제한 시간 초과 시 자동 오답 처리 및 생명 감소
6. 정답 검증 단계 - 입력된 값과 정답 레지스터 비교

정답 시: 점수 +1, Full Color LED(녹색), Peizo 승리음, 오답 시: 생명 -1, Full Color LED(빨간색), Peizo 경고음

7. 생명 관리 단계 - 생명은 LED1~3 으로 표시 (꺼질수록 생명 감소), 생명 0 개일 때 다시 오답 → Game Over
8. 게임 종료 단계 - LCD 에 "Game Over" 및 닉네임, 최종 점수 표시, Dip Switch [8] (reset) 입력 시 닉네임 입력 단계로 복귀



4. 개발 일정

| | 11 월 2 주 | 11 월 3 주 | 11 월 4 주 | 12 월 1 주 | 12 월 2 주 | 12 월 3 주 |
|---------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 제안서 및 계획 발표 | | | | | | |
| Calculator | | | | | | |
| Puzzle Game (Nickname 입력) | | | | | | |
| Peizo, Random function | | | | | | |
| 모듈연결 동작확인 | | | | | | |
| 최종 보고서 작성 | | | | | | |

먼저 1-2 주 차에 계산기 모드를 완성합니다. 이 기간에 키패드 입력, 기본 연산(ALU), 그리고 LCD 숫자 표시 기능 구현을 마무리합니다. 3주차에는 퍼즐 게임에 필요한 영어 입력 로직을 구현하고 이어서 Peizo, 난수 생성(Random function) 모듈 그리고 LED, Full Color LED, LCD 디스플레이 관련 로직을 모두 개발합니다. 이러한 개별 모듈 구현이 완료되면 시스템 전체의 모듈 연결 동작을 확인하고 통합 디버깅을 진행합니다. 특히 계산기와 퍼즐 두 가지 모드가 DIP 스위치로 올바르게 전환되는지 집중적으로 테스트합니다. 마지막 주차에는 최종 보고서 작성과 데모 시연을 준비하며 프로젝트를 마무리할 예정입니다.