

# CSED311: Lab 1 (due Mar. 3)

김태연(20220140), 손량(20220323)

Last compiled on: Tuesday 5<sup>th</sup> March, 2024, 20:06

## 1 Introduction

베릴로그 RTL 설계 방법을 익히기 위해 간단한 combinational circuit과 sequential circuit인 ALU와 vending machine을 구현해 본다.

## 2 Design

Vending machine은 크게 다음과 같은 모듈로 나누어 설계하였다.

### 2.1 Coin Storage

이름에서 유추할 수 있듯, 사용자가 넣은 동전을 관리하는 모듈이다. 이 모듈은 다음 입력을 처리할 수 있도록 설계하였다.

- 사용자가 집어넣은 동전에 대한 입력
- 사용자가 구입한 물품의 가격
- 동전 반환 버튼 입력, 시간 초과에 해당하는 입력

Coin storage의 경우 사용자 입력에 대해 바로 출력이 일어날 필요가 없으므로 Moore machine 형태로 설계하였다. 사용자의 입력만 갖고 다음 상태를 계산하기 위해, 상태를 저장하는 레지스터에 대해 다음 상태를 담는 레지스터를 하나 더 추가해 구현하였다. 이 모듈에서 사용하는 레지스터는 다음 정보를 저장한다.

- 현재 저장하고 있는 동전의 총 합
- 반환할 동전의 조합
- 반환 버튼을 누르고 실제 동전 반환이 일어날 때까지 남은 시간

이러한 정보를 저장하는 것에서, coin storage가 가지는 상태는 다음과 같은 4-튜플으로 나타낼 수 있다.

$$(C, R, T, S) \in \{0, 1\}^{32} \times \{0, 1\}^{k\text{NumCoins}} \times \{0, 1\}^2 \times \{0, 1\}$$

여기서 각 변수에 대한 설명은 다음과 같다.

- $C$  – 현재 저장 중인 동전의 총 합

- $R$  – 반환할 동전의 조합
- $T$  – 반환 버튼을 누르고 실제 반환이 일어날 때까지 남은 시간
- $S$  – 반환 버튼을 누르고 기다리는 중인지의 여부

## 2.2 Timekeeper

### 2.3 계산을 위한 combinational logic

동전 조합과 액수 사이의 변환에 대한 계산을 모두 sequential logic 내부에서 계산한다면 sequential logic의 코드가 복잡해진다. 이러한 상황을 피하기 위해 간단한 계산을 수행하기 위한 combinational logic은 별도의 모듈로 구현하도록 설계하였다. 크게 다음과 같은 combinational logic 모듈을 설계하여 사용하였다.

## 3 Implementation

## 4 Discussion

## 5 Conclusion