



# Digital Combinational Circuit 2 (Implementation)

**Shinwoong Kim**

# 목표

---

- **조합 논리 회로(Combinational logic circuit) 시스템을 설계할 수 있다.**
  - ✓ 주어진 문제에 대한 진리표를 작성할 수 있다
  - ✓ K-map을 사용하여 각 출력에 대한 Boolean 함수를 도출 할 수 있다
  - ✓ 도출된 Boolean 함수를 실제 하드웨어로 설계할 수 있다
- **7-Segment LED 및 구동회로 (74LS48 IC)를 사용할 수 있다.**
  - ✓ Common cathode type의 7-segment LED를 사용할 수 있다
  - ✓ 7-segment LED를 구동하는 칩(74LS48)을 사용할 수 있다

# 실험 내용 설명

## • 자판기 설계 (Vending machine)

### ✓ 문제정의

- 음료수 선택: 500원, 600원 음료수 2가지 종류가 있음 (1bit, S)
- 금액 입력: 500원 단위 (1bit,  $N_2$ ), 100원 단위 (2bits,  $N_1N_0$ ) 입력
- 출력: 음료수 출력 표시 ( $G_1, G_0$ ), 100원 단위 거스름돈 (2bits,  $C_1C_0$ )
- 즉, 입력은 총 4bits ( $S N_2 N_1 N_0$ ), 출력 총 4bits ( $G_1 G_0, C_1 C_0$ )

S: 제품선택

S	제품
0	500원
1	600원

$G_1$ : 600원 제품 출력  
 $G_0$ : 500원 제품 출력

$N_2N_1N_0$ : 투입금

$N_2$	금액	$N_1N_0$	금액
0	0원	00	0원
1	500원	01	100원
		10	200원
		11	300원

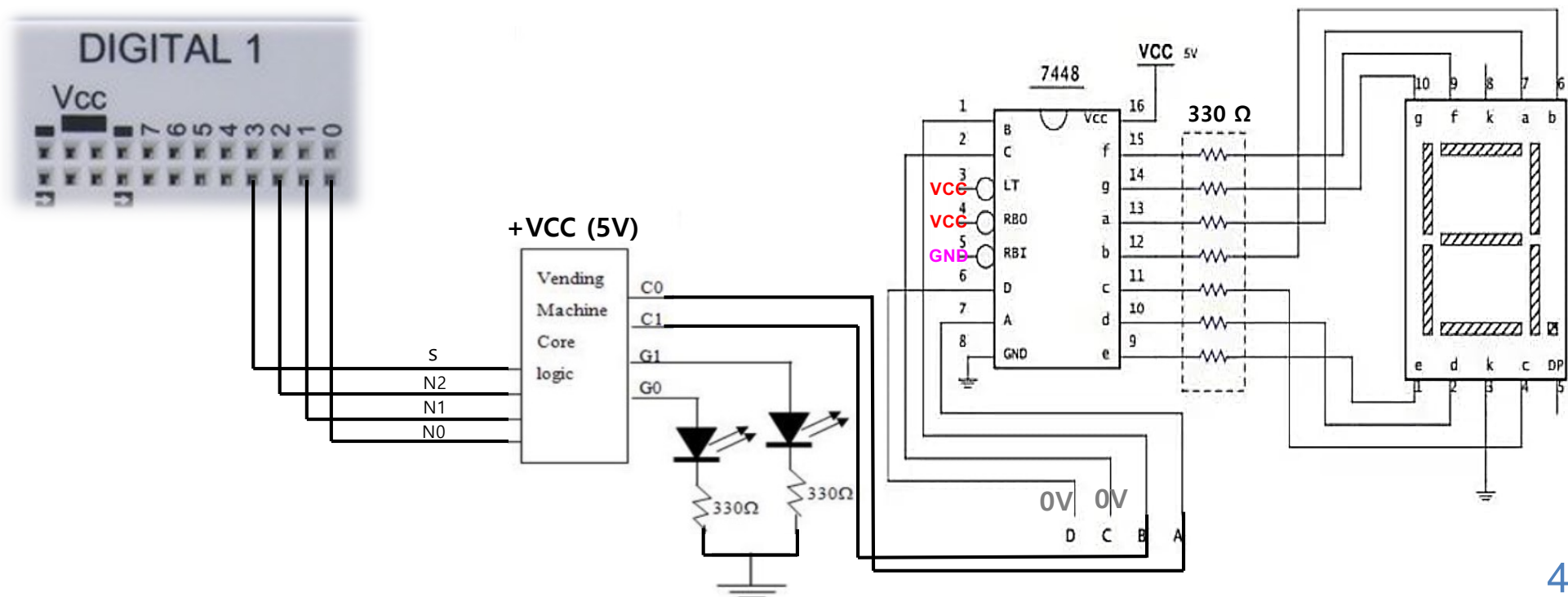


$C_1C_0$	잔돈 출력
00	0원
01	100원
10	200원
11	300원

# 실험 내용 설명

## • 자판기 설계 (Vending machine)

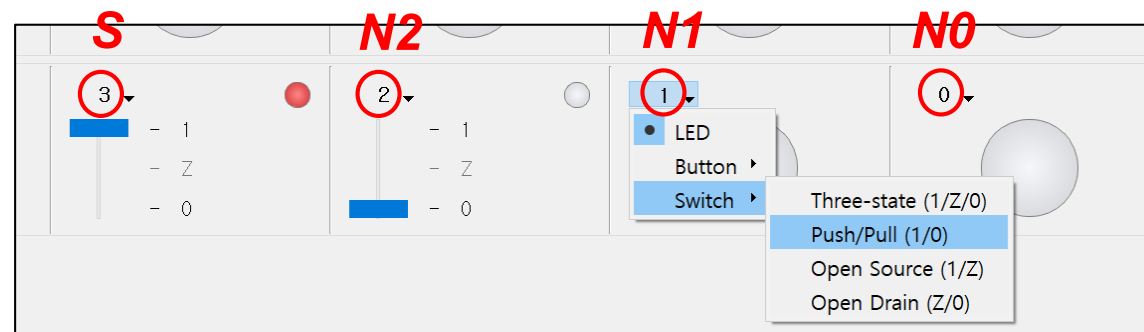
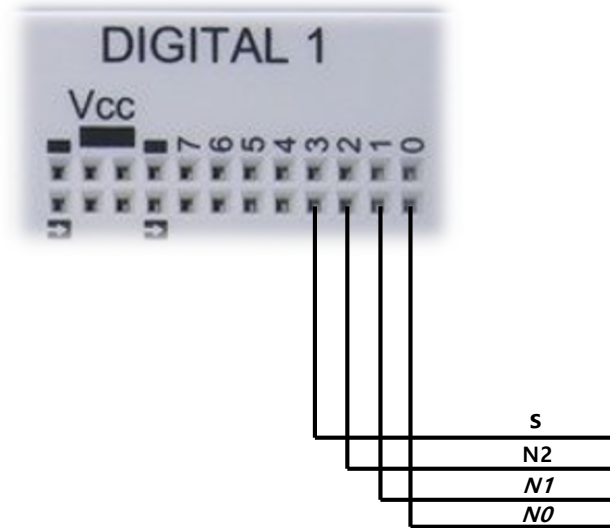
- ✓ TinkerCAD로 설계한 회로를 실제 IC를 사용하여 설계
- ✓ 각 음료수 출력 ( $G_1$ ,  $G_0$ )를 위해 LED(+ 330 $\Omega$ )를 사용
- ✓ 거스름돈 ( $C_1C_0$ ) 표현을 위해서는 7-segment LED + 74LS48 decoder를 사용
- ✓ 입력 control은 EEboard의 DIGITAL1 module 중 4bit (3~0 port) 사용
- ✓ VCC: 5V 사용



# 실험 내용 설명

## • 자판기 설계 (Vending machine)

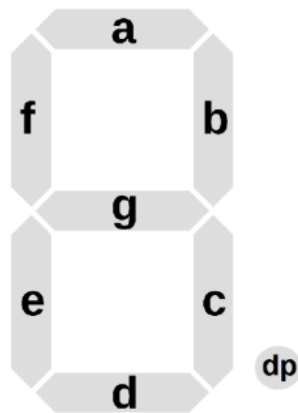
- ✓ 입력 control은 EEboard의 DIGITAL1 module 중 4bit (3~0 port) 사용
- ✓ [SW] StaticIO 속성: Switch → Push/Pull (1/0) 선택



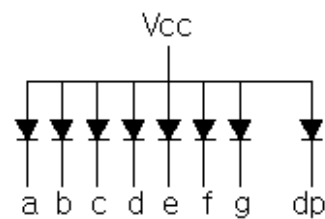
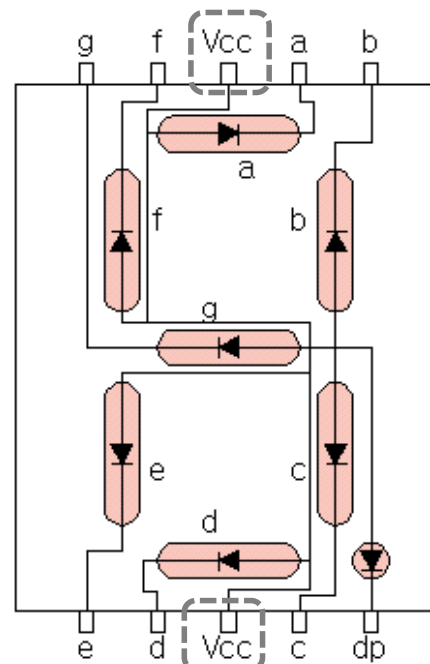
# 실험 내용 설명

## • 7-Segment LED

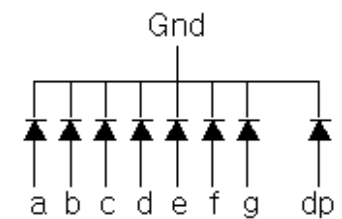
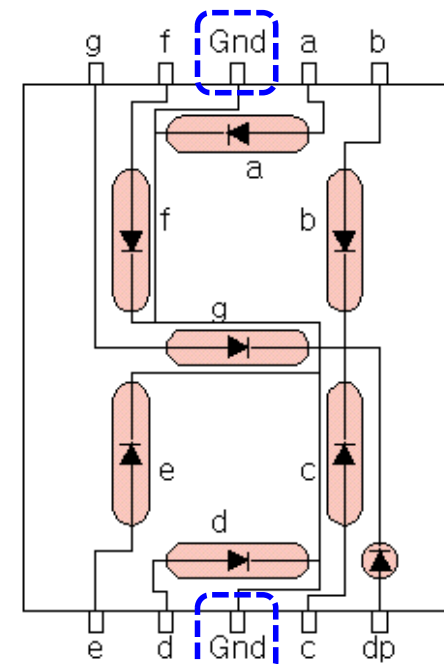
✓ Common Anode vs. Common Cathode



[\\*https://theorycircuit.com/7-segment-arduino-interface/](https://theorycircuit.com/7-segment-arduino-interface/)



common-anode type



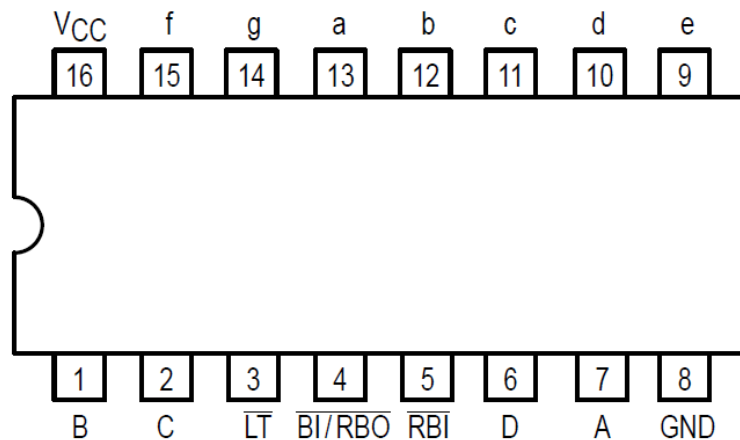
common-cathode type

[\\*https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=jamduino&logNo=220932416728](https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=jamduino&logNo=220932416728)

# 실험 내용 설명

## • BCD to 7-Segment decoder

- ✓ 74LS48: **active high** type
- ✓ 입력 4bit binary code는 **DCBA<sub>(2)</sub>** 순서임



### PIN NAMES

A, B, C, D	BCD Inputs
RBI	Ripple-Blanking (Active Low) Input
LT	Lamp-Test (Active Low) Input
BI/RBO	Blanking Input or Ripple-Blanking Output (Active Low)
BI	Blanking (Active Low) Input

DECIMAL OR FUNCTION	INPUTS						OUTPUTS							NOTE	
	LT	RBI	D	C	B	A	BI/RBO	a	b	c	d	e	f		g
0	H	H	L	L	L	L	H	H	H	H	H	H	H	L	1
1	H	X	L	L	L	H	H	L	H	H	L	L	L	L	1
2	H	X	L	L	H	L	H	H	H	L	H	H	L	H	
3	H	X	L	L	H	H	H	H	H	H	H	L	L	H	
4	H	X	L	H	L	L	H	L	H	H	L	L	H	H	
5	H	X	L	H	L	H	H	H	L	H	H	L	H	H	
6	H	X	L	H	H	L	H	L	L	H	H	H	H	H	
7	H	X	L	H	H	H	H	H	H	H	L	L	L	L	
8	H	X	H	L	L	L	H	H	H	H	H	H	H	H	
9	H	X	H	L	L	H	H	H	H	H	L	L	H	H	
10	H	X	H	L	H	L	H	L	L	L	H	H	L	H	
11	H	X	H	L	H	H	H	L	L	H	H	L	L	H	
12	H	X	H	H	L	L	H	L	H	L	L	L	H	H	
13	H	X	H	H	L	H	H	H	L	L	H	L	H	H	
14	H	X	H	H	H	L	H	L	L	L	H	H	H	H	
15	H	X	H	H	H	H	H	L	L	L	L	L	L	L	
BI	X	X	X	X	X	X	L	L	L	L	L	L	L	L	2
RBI	H	L	L	L	L	L	L	L	L	L	L	L	L	L	3
LT	L	X	X	X	X	X	H	H	H	H	H	H	H	H	4

# 실험 내용 설명

## • 74LS48 Decoder + 7-Segment 연결

- ✓ 330Ω 저항을 사용하여 연결 (74LS48 ↔ 7-segment)

