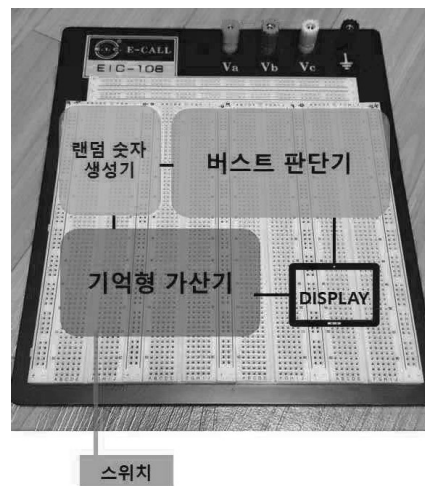


1. 과제 개요

본 프로젝트에서는 논리회로를 이용한 기억형 가산기와 감산기를 개발할 것이다. 현재 시장의 대부분의 가산기와 감산기는 ARM 아키텍처에서 공급하고 있다. ARM 아키텍처의 가산기는 기억능력이 없기 때문에 만약 기억능력을 얻기 위해서는 별도의 레지스터가 필요하다. 반면에 우리 조는 가산기 자체를 사용하여 다비트 감/가산기를 만드는 것이 아니라, 처음부터 기억능력이 있는 플립플롭들을 조합하여 감/가산기를 사용하지 않고 기억성 가산기와 감산기를 개발할 것이다. 우리 조가 개발할 가산기는 레지스터를 추가로 필요로 하지 않기 때문에 기존의 가산기보다 더욱 경제적이다. 이번 프로젝트에서는 가산기를 이용하여 블랙잭 게임을 만드는 데 사용했지만 이와 같은 장치는 통계를 내거나 인원수를 세거나 하는 등 기존에 쓰이고 있는 다양한 분야에 더욱 값싼 가격으로 사용할 수 있을 것이다.

2. 개발 내용



완성된 회로 예상도



동작 순서도

3. 개발 방법

논리회로 과목에서 배웠던 불대수를 이용하여 1과 0으로 디지털 논리회로를 사용할 것이다. 기본 연산인 AND, OR, NOT을 시작으로 Flip-Flop과 Adder를 추가적으로 사용할 것이다.

실험 방법은, 원하는 결과를 얻기 위한 진리표를 작성한 후 필요한 IC칩과 입력이 어떻게 되는 지 분석을 할 것이다. 그 후 피스파이스 프로그램을 이용하여 실제로 회로를 만들어 보기 전에 회로가 제대로 동작을 하는 지 테스트를 해본 후 실제로 브레드보드에 IC칩을 연결하여 개발을 진행할 예정이다 블랙잭 게임을 만들기 위해서는 기억형 가산기 이외에 랜덤 숫자 발생기와 버스트 판단기, 카드 뽑는 장치 등이 추가로 필요하다. 자세한 내용은 다음과 같다.

<랜덤 숫자(포커 카드) 발생기>

블랙잭에서 뽑을 수 있는 카드와 각 카드의 유효 값은 한정되어있다. 숫자 카드는 1~10까지의 값을 가지며 그림카드 J, Q, K는 모두 10의 값을 가진다. 실제 카드를 뽑는 확률을 모방하기 위해 1~10까지의 카운터에 추가로 J, Q, K또한 10으로 뽑히는 가능성을 추가하여 설계할 것이다. 4개의 10을 각각 구별하기 위해서는 2개의 비트가 필요해서 총 6개의 비트가 필요할 것이다. 카운터의 진리표를 작성하면 다음과 같다.

	T						T+1						
	F	E	D	C	B	A	F	E	D	C	B	A	
0	0	0	0	0	0	0	0	0	0	0	0	1	
1	0	0	0	0	0	1	0	0	0	0	1	0	2
2	0	0	0	0	1	0	0	0	0	0	1	1	3
3	0	0	0	0	1	1	0	0	0	1	0	0	4
4	0	0	0	1	0	0	0	0	0	1	0	1	5
5	0	0	0	1	0	1	0	0	0	1	1	0	6
6	0	0	0	1	1	0	0	0	0	1	1	1	7
7	0	0	0	1	1	1	0	0	1	0	0	0	8
8	0	0	1	0	0	0	0	0	1	0	0	1	9
9	0	0	1	0	0	1	0	0	1	0	1	0	10
10	0	0	1	0	1	0	0	1	1	0	1	0	10
10	0	1	1	0	1	0	1	0	1	0	1	0	10
10	1	0	1	0	1	0	1	1	1	0	1	0	10
10	1	1	1	0	1	0	0	0	1	0	1	1	A/11
A/11	0	0	1	0	1	1	0	0	0	0	1	0	2

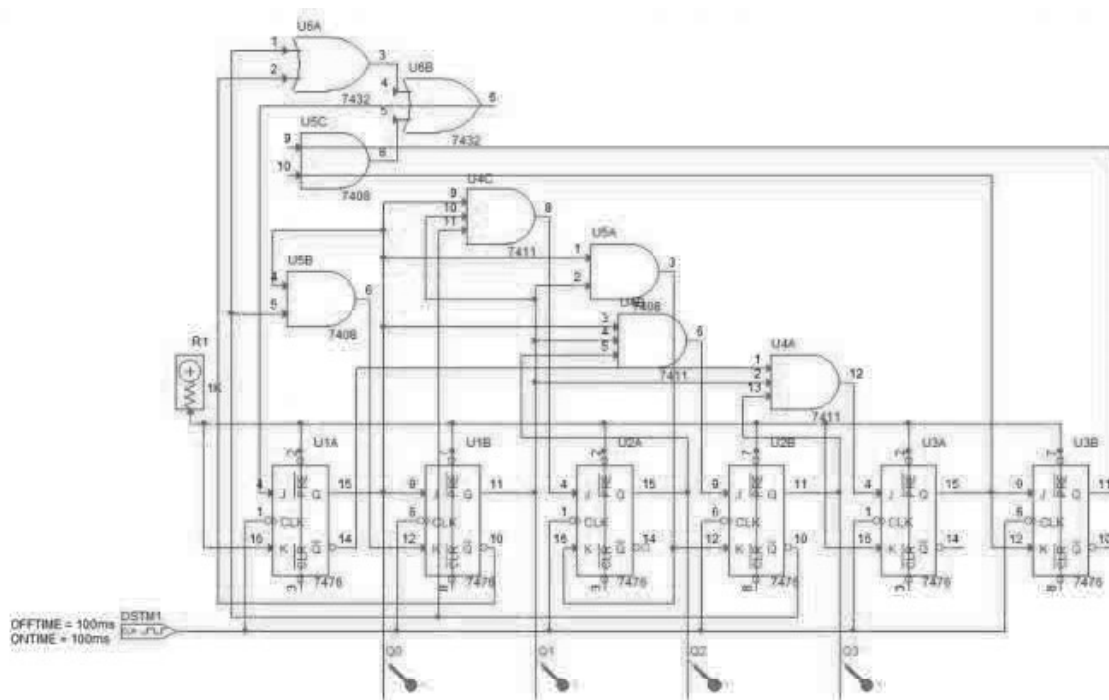
이를 JK 플립플롭으로 구현하려면 다음과 같이 입력을 해주면 된다. (빈칸은 Don't Care)

F		E		D		C		B		A	
J	K	J	K	J	K	J	K	J	K	J	K
0	-	0	-	0	-	0	-	0	-	1	-
0	-	0	-	0	-	0	-	1	-	-	1
0	-	0	-	0	-	0	-	-	0	1	-
0	-	0	-	0	-	1	-	-	1	-	1
0	-	0	-	0	-	-	0	0	-	1	-
0	-	0	-	0	-	-	0	1	-	-	1
0	-	0	-	0	-	-	0	-	0	1	-
0	-	0	-	1	-	-	1	-	1	-	1
0	-	0	-	-	0	0	-	0	-	1	-
0	-	0	-	-	0	0	-	1	-	-	1
0	-	1	-	-	0	0	-	-	0	0	-
1	-	-	1	-	0	0	-	-	0	0	-
-	0	1	-	-	0	0	-	-	0	0	-
-	1	-	1	-	0	0	-	-	0	1	-
0	-	0	-	-	1	0	-	-	0	-	1

위 진리표로 6개 변수 카르노 맵을 작성하여 논리식을 구하면 다음과 같다.

비트	입력	논리식	비트	입력	논리식
F	J	E	C	J	ABD'
	K	E		K	AB
E	J	A'BD	B	J	A
	K	1		K	AD'
D	J	ABC	A	J	D'+B'+EF
	K	AB		K	1

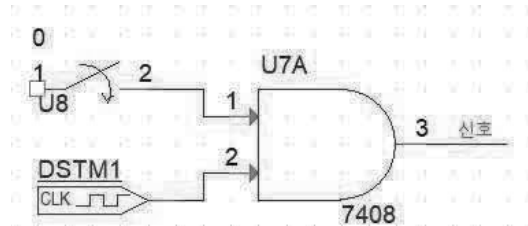
아래의 그림은 위 수식을 바탕으로 피스파이스 프로그램을 사용하여 회로도를 그린 것이다.



<카드 선택/자동 차단 장치>

카드를 선택하는데 스위치를 쓸 것이다. 스위치를 쓸 때 아무리 빠르게 누른다 해도 카운터에서 하나의 숫자만 선택할 정도로 빠르게 누를 수 있는 것은 아니다. 실시간으로 숫자의 합을 보여줄 때 누르는 동안 다수의 신호가 전달되면 정상적으로 카드를 하나만 뽑을 수 없다. 그러므로 스위치를 누르는 시간 간격과 상관없이 하나의 신호만 출력을 해야한다.

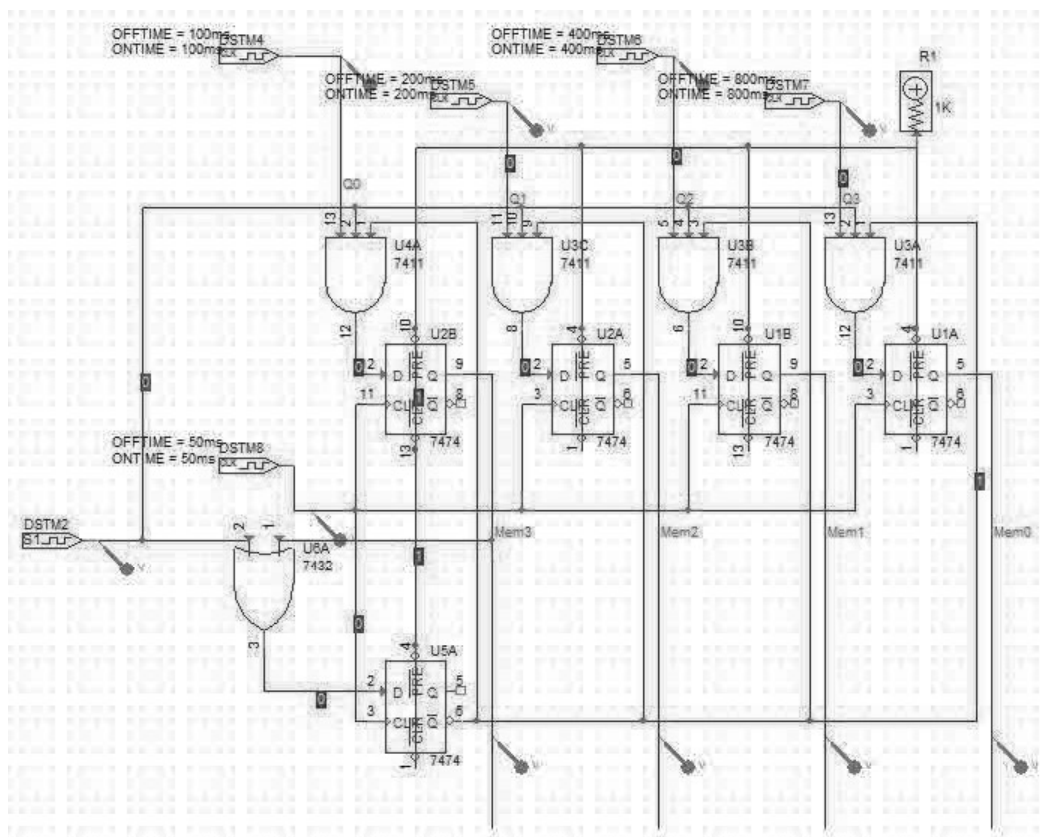
카운터에서 일단 신호를 받으려면 스위치와 입력신호를 AND게이트에 입력을 하면 스위치가 1일 때만 신호가 통과할 수 있다.



이를 카운터의 실제 숫자가 출력되는 4개의 비트에 각각 연결하면 된다.

카운터와 상관없이 출력된 숫자를 따로 저장하기 위해서 신호의 입력을 D플립플롭에 연결하여 신호가 다시 0이 되어도 출력값이 일시적으로 유지 되도록 한다.

이에 이어 다음 클럭 주기(카운터의 숫자가 변하기 전)에 신호가 다시 0이 될 수 있도록 또다른 D플립플롭을 사용하여 자동으로 스위치가 닫히도록 한다. 기존의 2입력 AND게이트를 3입력 AND게이트로 교체하고 차단기의 출력을 같이 입력한다.



스위치와 차단기가 OR게이트로 더해진 것은 스위치가 ON(1)일 때 Q'이 0이 되도록 하여 신호를 차단시키기 위한 것이며 스위치가 OFF일때는 Q'이 1이 되어 너무 빠르게 스위치가 닫힐 때 조금의 여유를 주기 위한 것이다.

<기역성 가산기>

처음에는 플립플롭만을 이용하여 실제 가산기가 아닌 가산기의 역할을 하는 상태변환 장치를 생각해보았다.

캐리가 없는 반가산기: 입력 1의 변화 관찰

입력 1	입력 2	캐리	상태 변화	변화 원리		특이사항
0	0	0	0 에서 0	리셋	유지	
0	1	0	0 에서 1	셋팅	반전	
1	0	0	1 에서 1	셋팅	유지	
1	1	0	1 에서 0	리셋	반전	캐리 1

위 변화 원리를 생각하면 입력 2가 1일 때 반전이 되며 0일 때 유지가 되어 T플립플롭으로 구현이 가능할 것 같다고 생각을 하였다.

캐리는 두 입력의 AND게이트로 신호를 만들 수 있다.

그다음 캐리를 받을 경우를 생각하였다.

캐리를 받는 전가산기: 입력 1의 변화 관찰

입력 1	입력 2	캐리	상태 변화	변화 원리		특이사항
0	0	1	0 에서 1	셋팅	반전	
0	1	1	0 에서 0	리셋	유지	캐리 1
1	0	1	1 에서 0	리셋	반전	캐리 1
1	1	1	1 에서 1	셋팅	유지	캐리 1

여기서는 캐리가 없을 때와 반대로 입력 2가 1일 때 유지가 되고 0일 때 반전이 된다.

캐리는 두 입력 신호의 OR게이트로 신호를 만들 수 있다.

이러면 먼저 캐리가 발생하는 경우를 생각하여 위에 색칠된 입력 1, 2와 캐리의 특정 조합으로 1이 출력되도록 T입력을 제어하면 될 것이다.

캐리	2	1	반전	캐리
C	B	A	T	C'
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	1
1	0	0	1	0
1	0	1	1	1
1	1	0	0	1
1	1	1	0	1

진리표

A\BC	00	01	11	10
0	0	1	0	1
1	0	1	0	1

T

A\BC	00	01	11	10
0	0	0	1	0
1	0	1	1	1

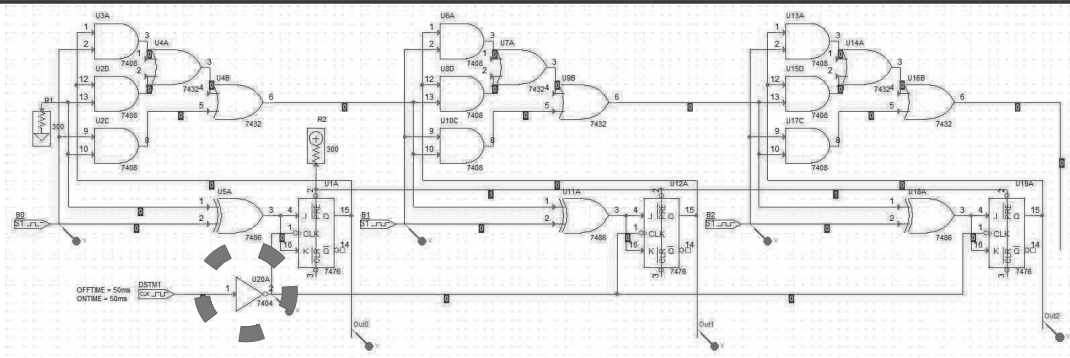
C

카르노맵

논리식은 다음과 같다.

$$T = B'C + BC'$$

$$C = AB + AC + BC$$



회로도에 표시된 붉은색 부분은 자동 차단 장치에서 쓰인 D플립플롭과 기억성 가산기에 쓰인JK플립플롭의 작동에지를 맞추주기 위해, T플립플롭 가산기를 사용할 때 클럭에 NOT게이트를 연결하였다.

<숫자 비교기>

21은 2진수로 10101

22는 2진수로 10110

위의 두 숫자를 비교하기 위해 5비트 비교기를 쓸 수 있지만, 상위 4비트만을 비교해도 된다. 앞의 4비트가 1011이면 21이하의 모든 숫자보다 크기 때문에 앞의 4비트를 1011과 비교해서 같다 또는 크다가 출력이 되면 PRESET을 0으로 입력하여 가산기가 11111이 출력되게 한다. 비교기에 이어 게임을 초기화(재시작)하기 위한 스위치도 함께 설계했다.

만약 입력값이 22보다 같거나 크다는 결과를 H가 1일 때, 스위치를 ON 했을 때를 S가 1일 때로 정하면 다음과 같은 진리표를 만들 수 있다.

H	S	PRESET	CLEAR	상태	결과
0	0	1	1	(합 < 22) (스위치 OFF)	P/C OFF
0	1	1	0	(합 < 22) (스위치 ON)	Clear - 초기화
1	0	0	1	(합 > 21) (스위치 OFF)	BUST 출력
1	1	1	0	(합 > 21) (스위치 ON)	Clear - 초기화

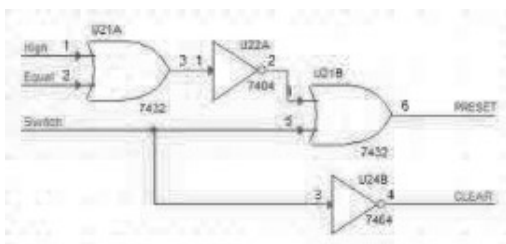
위 진리표로 논리식을 다음과 같이 만들었다.

$$P = H'S' + H'S + HS \text{ 또는 } P = (HS')' = H'+S$$

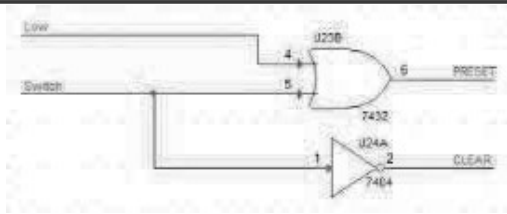
$$C = S'$$

위의 H는 다음과 같이 신호를 만들 수 있다.

비교기의 Higher/Equal의 OR 출력으로 22와 같거나 클 때, 비교기의 Lower의 NOT 출력으로 22보다 작을 때로 스위치와 함께 연결하면 된다.



방법1 : Higher/Equal의 OR 출력



회로가 간단한 방법2를 선택할 것이다.

4. 작품의 재료

작품의 재료는 브레드보드를 장착시킨 HBE-LogicLab-B, 회로를 구성하는 각종 IC 칩이다. 작품을 만드는 데 사용할 예정인 IC 칩과 장비는 다음과 같다

장비/부품명	기능/규격		수량(개)
HBE-LogicLab-D			1
신호 연결 케이블			1
호스트 PC			1
IC 칩	74LS32	OR 게이트	2
	74LS08	AND 게이트	1
	74LS11	3입력 AND	3
	74LS86	XOR 게이트	1
	74LS76	JK 플립플롭	3
	74LS74	D 플립플롭	6
	74LS83	4비트 병렬 가산기	1
7- Segment(3) LED			2

5. 과제추진계획 및 일정

수행 내용	일정		
	9월	10월	11월
결과표 작성 및 피스파이스 시뮬레이션			
필요한 IC부품 구입 및 작품 제작			
최종 작품 제작			
보고서 작성			