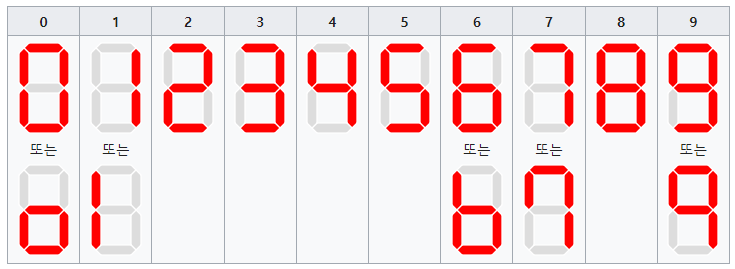
8주차 예비보고서

전공: 물리학과 학년: 3학년 학번: 20191286 이름: 김나현

1. 7-Segment Display에 대해 조사하시오.

7-Segment Display는 7개의 획으로 숫자나 문자를 나타내는 표시 장치입니다. 흔히 디지털 시계나 디지털 계산기에서 숫자를 표현할 때 이 7-Segment Display 장치의 표기를 이용하므로 우리는 7-Segment Display의 표기에 친숙하고 쉽게 이해할 수 있습니다. 다음 <Figure 1>은 일반적으로 0부터 9까지의 숫자에 해당하는 7-Segment Display의 모습입니다.

<Figure 1> 7-Segment Display의 모습

7-Segment Display는 위의 그림처럼 가로 획 두 개, 세로 획 두 개로 이루어진 사각형을 겹쳐 놓은 모양으로 생겼고, 7개의 획은 각각 꺼지거나 켜져서 위의 그림처럼 아라비아 숫자를 표시하거나 문자를 나타냅니다. 이때, <Figure 1>에 보이는 것처럼 0, 1, 6, 7, 9는 둘 이상의 방법으로 표시가 가능하고 7개의 획은 맨 위의 획부터 시계방향으로 A, B, C, D, E, F, G까지 알파벳을 붙여 구분합니다.

7-Segment Display는 일반적으로 발광 다이오드(LED)를 이용하여 각 획을 표시하지만 액정 디스플레이(LCD)나 음극 방전관 등을 이용하는 경우도 있습니다. 또한 7-Segment Display는 가독성을 위해 기울여 나타내기도 하며 앞서 언급하였듯이 문자를 표시할 수도 있는데 다음 <Figure 2>는 카스테레오에서 ‘play’를 나타낼 때 7-Segment Display를 이용한 예시입니다.

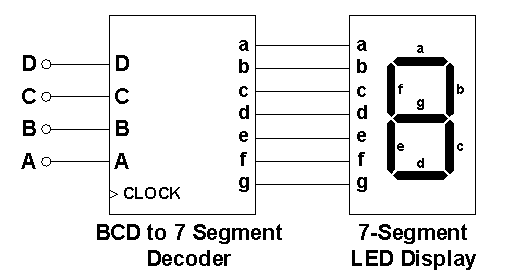
<Figure 2> 7-Segment Display로 재생을 뜻하는 ‘play’를 표시한 모습

1. 7-Segment Display의 동작 원리에 대해 조사하시오.

텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

<Figure 3> 7-Segment Display

7-Segment에는 각 획을 켜기 위해 A부터 G까지의 입력이 들어올 수 있는 7개의 입력 단자가 있습니다. 만약, 7-Segment Display로 0을 나타내고 싶으면 a,b,c,d,e,f의 입력에 1이 들어오면 되고, 3을 나타내고 싶으면 a,b,c,d,g의 입력에 1이 들어오면 됩니다. <Figure 3>의 dp라고 적힌 점은 7-Segment의 위아래를 구분하거나 소수점을 나타내기 위한 점으로, 점이 오른쪽 아래 배치하면 7-Segment Display가 제대로 배치된 것입니다.

<Figure 4> BCD 디코더와 7-Segment Display를 연결한 모습

위의 <Figure 4>와 같이 BCD의 4비트를 7비트로 나누어 주는 BDC 디코더와 7-Segment Display를 연결하면 입력 D, C, B, A가 의미하는 BCD를 7-Segment Display에서 관찰이 가능합니다. 이처럼 7-Segment는 0부터 9까지의 숫자를 한눈에 알 수 있는 아라비아 숫자로 display함으로써 논리 회로의 결과를 직관적으로 관찰할 수 있게 합니다. BCD 디코더에 인가되는 입력과 7-Segment에 인가되는 a,b,c,d,e,f,g에 대해 진리표를 작성해보면 다음과 같습니다.

테이블이(가) 표시된 사진

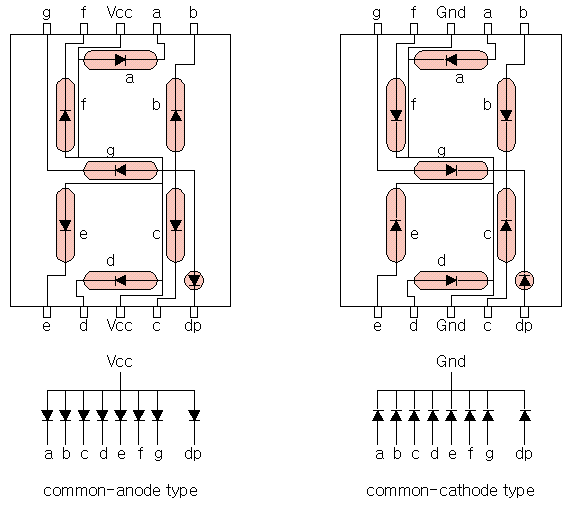
자동 생성된 설명

<Figure 5> BCD 디코더와 7-Segment Display의 진리표

위의 진리표를 통해, BCD 디코더에 인가되는 1비트 4개의 입력들이 디코더를 통해 1비트 7개의 입력으로 바뀌어 7-Segment Display에 인가되어 D, C, B, A가 의미하는 숫자를 그대로 7-Segment Display에 나타낸다는 것을 알 수 있습니다.

1. 7-Segment(Anode Type, Cathode Type)에 대해 조사하시오.

7-Segment Display는 두 가지 종류가 있는데, 하나는 common anode type(+공통)이고 다른 하나는 common cathode type(-공통)입니다. Common anode type은 공통 단자에 Vcc가 연결되어 모든 LED의 +극에 Vcc가 인가되고 각 LED의 -극에 Vcc나 GND를 연결하여 해당 LED에 불이 들어올지, 말지를 결정하는 방법이고 common cathode type은 공통 단자에 GND가 연결되어 모든 LED의 -극에 GND이 인가되고 각 LED의 +극에 Vcc나 GND을 연결하여 해당 LED에 불이 들어올지, 말지를 결정하는 방법입니다.



<Figure 6> 7-segment display의 common anode type과 common cathode type

1. 7-Segment의 구동 방식에 대해 설명하시오.

7-Segment Display의 구동 방식은 크게 두 가지 종류로 나뉘는데, 하나는 static 구동과 다른 하나는 dynamic 구동입니다. Dynamic 구동 방식은 각 LED를 모두 ON하는 것이 아니라 여러 개의 LED들을 번갈아 가면서 빠르게 ON, OFF 해주는 방식으로, 전체 LED를 모두 한 번에 키지 않고 빠르게 순차적으로 점등함으로써 잔상과 착시 현상을 이용해 여러 개의 LED가 동시에 켜져 있는 것처럼 보이게 합니다. 반면, Static 구동 방식은 포트나 플립플롭을 이용하여 모든 LED를 한 번에 ON하므로 dynamic 구동 방식에 비해 전류 소모가 큽니다.

1. 기타 이론

<Figure 4>의 BCD 디코더에서 각 출력, a,b,c,d,e,f,g을 입력 A,B,C,D에 대한 논리 함수로 표현해보기 위해 <Figure 5> BCD 디코더의 진리표와 카르노맵을 이용해보겠습니다.

<Table 1> a의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 1 | 1 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 a를 a=B’D’+C+BD+A라는 논리 함수로 정리할 수 있습니다.

<Table 2> b의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 1 | 1 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 b를 b=B’+C’D’+CD라는 논리 함수로 정리할 수 있습니다.

<Table 3> c의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 1 | 1 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 c를 c=C’+D+B라는 논리 함수로 정리할 수 있습니다.

<Table 4> d의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 d를 d=B’D’+B’C+BC’D+CD’+A라는 논리 함수로 정리할 수 있습니다.

<Table 5> e의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 0 | 1 |
| 01 | 0 | 0 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 0 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 e를 e=B’D’+CD’라는 논리 함수로 정리할 수 있습니다.

<Table 6> f의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 0 | 0 |
| 01 | 1 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 f를 f=C’D’+BC’+BD’+A라는 논리 함수로 정리할 수 있습니다.

<Table 7> g의 논리 함수를 구하기 위한 카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AB\CD | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 1 |
| 01 | 1 | 1 | 0 | 1 |
| 11 | X | X | X | X |
| 10 | 1 | 1 | X | X |

위의 카르노맵와 SOP를 통해 BCD 디코더의 출력 g를 g=B’C+BC’+A+BD’라는 논리 함수로 정리할 수 있습니다.

따라서, BCD 디코더 내부에서는 출력 a,b,c,d,e,f,g를 만들기 위해 위에서 구한 논리 함수(혹은 최적화된 논리 함수)대로 회로가 구현되었음을 예측할 수 있습니다.