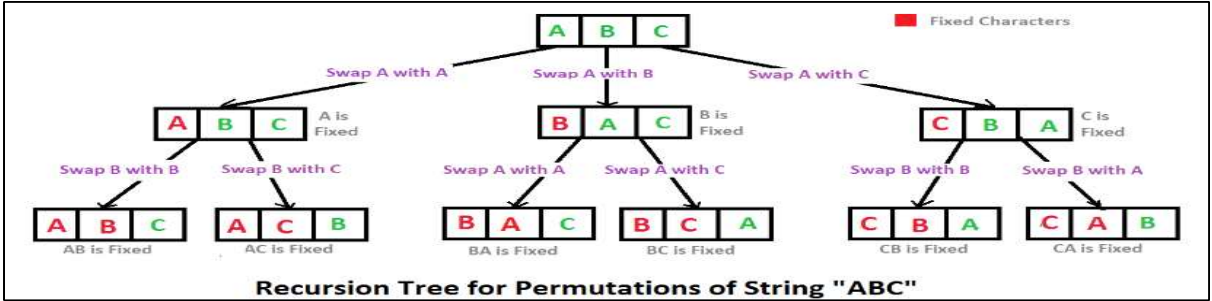
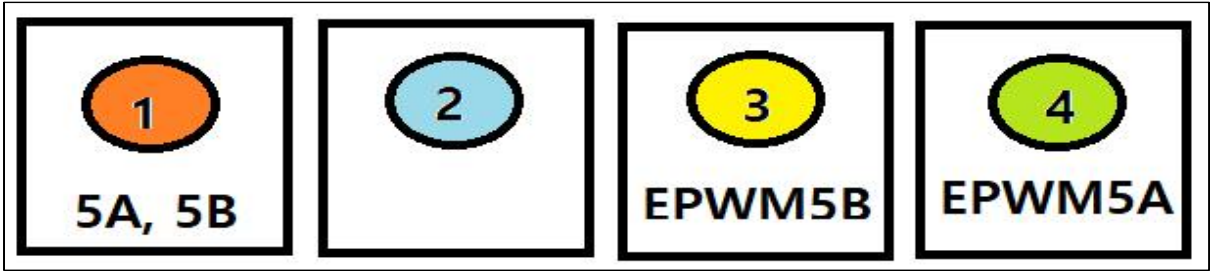


400줄짜리 : 그림은 800줄짜리와 구현동작을 설명하기 위해, 이런식으로 동작한다는 것을 표현하기 위해 설정하였다.



800줄짜리



400줄짜리와 800줄짜리의 차이점은 400줄짜리는 mode 1, 2, 3으로 설정하여, 프로젝트2에서 요구하는 값에 대한 동작을 조건문에 따라 실행되도록 설정하였다.

그런데 8000줄짜리는 위의 그림과 같이 sw에 따라 서로 다른 기능을 부여하여, 그 조건에 맞는 동작을 하기 위해, 스위치의 on, off를 설정한다.

```
if (SW1 == 1 && SW3_cnt % 2 == 0 && SW4_cnt % 2 == 0)
버튼 1번을 누른 것이기에, Sw1=1이라고 설정하였고,
sw3, sw4=0이라고 한 이유는 EPWM 5A, 5B를 기본설정에서 사용한다고 하였기에, 설정하였다.
```

```
PwmDutyRatioA = ((float32)ADC_value01 / (float32)65535 )*(float32)0.475;
EPwm5Regs.CMPA.half.CMPA = (EPwm5Regs.TBPRD + 1) * PwmDutyRatioA;
ADC_Value01의 값은 최소값 0으로 설정하였고, 3V인 65535을 가진다.
이에 비례하는 duty비를 설정하였고, 최댓값이 PWM 주기 절반의 95%로 설정하기 위해, 코드 설정.
```

```
EPwm5Regs.CMPB = (EPwm5Regs.TBPRD + 1) * (1 - PwmDutyRatioA);
EPWM5B와 EPWM5A와 상보적으로 나타나야 하기에, 1에서 CMPA에서 설정한 duty값을 빼준 값이 CMPB로 설정 된다.
```

```
else if (SW1 == 1 && SW3_cnt % 2 == 1 && SW4_cnt % 2 == 0)
EPwm5Regs.CMPB = (EPwm5Regs.TBPRD + 1) * (1 - PwmDutyRatioA);
```

```
else if (SW1 == 1 && SW3_cnt % 2 == 0 && SW4_cnt % 2 == 1)
EPwm5Regs.CMPA.half.CMPA = (EPwm5Regs.TBPRD + 1) * PwmDutyRatioA;
SW3 : EPWM5B동작                                SW4 : EPWM5A동작
```

```
else if (SW3_cnt % 2 == 1 && SW4_cnt % 2 == 1)
EPwm5Regs.CMPA.half.CMPA = 0;                                EPwm5Regs.CMPB = 5000;
```

만약 그 스위치 3, 4번이 동시에, 동작하면 상보적으로 동작할 수 있도록 동작을 구성하였다.

```
else if (SW1 == 2)
```

- B. 버튼 A 2번 클릭 : 가변저항 A의 변경에 따른 PWM 주파수 변화.
- SW2를 사용하지 않고, A를 2번 눌렀다는 것을 SW1 == 2로 표현하였다.

EPwm5Regs.CMPA.half.CMPA = (EPwm5Regs.TBPRD + 1) * PwmDutyRatioA;

EPwm5Regs.CMPB = (EPwm5Regs.TBPRD + 1) * (1 - PwmDutyRatioA);

- 우선 기본 설정이 EPWM 5A, 5B를 상보적으로 동작한다고 하였기에, 위와 같이 설정하였다.

가장 중요한 추가설정 사항의 경우, EPWM 5A와 5B는 상보적으로 동작하며, 가변 저항 A의 변경에 따라 PWM의 주파수는 최소 100kHz, 최대 300kHz로 가변.

을 구현하기 위해서, 아래와 같은 코드를 사용하였다.

PWM_CARRIER = 100E3 + 200E3 * (float32)ADC_value01 / (float32)65535;

이때 100E3는 최소 100kHz를 구현하였고, **“200E3 * (float32)ADC_value01 / (float32)65535;”**는 가변 저항 A 값의 변화에 따른 주파수 값을 설정하기 위해, 사용하였다.

마지막으로 버튼 A 3번 클릭을 하기 위해서, **else if (SW1 >= 3)**를 사용하였다.

EPwm5Regs.CMPA.half.CMPA = (EPwm5Regs.TBPRD + 1) * 0.5;

EPwm5Regs.CMPB = (EPwm5Regs.TBPRD + 1) * (1 - 0.5);

앞서 언급한 것처럼 같은 이유로 EPWM 5A, 5B를 상보적으로 동작한다고 하였기에, 설정하였다.

앞서 그림에서 스위치에 따라 동작하는 것들이 다르다고 하였는데, SW3가 1이 되는(버튼 C를 클릭) 경우,

“EPwm5Regs.CMPA.half.CMPA = 0;”를 통해, EPWM 5A의 동작이 중단되는 것을 설정하였고, **“%”**라는 것을 통해, 2번 클릭하면 0의 값을 갖기 때문에, 다시 동작한다는 것을 구현한다.

마찬가지로, 버튼 D를 누르면,(SW4가 1이 되는) **“EPwm5Regs.CMPB = 5000;”**를 통해, EPWM 5B의 동작이 중단되는 것과 **“%”**를 통해 다시 동작하는 것을 설정하였다.

<pre> if (SW3_cnt % 2 == 1) { EPwm5Regs.CMPA.half.CMPA = 0; } </pre>	<pre> if (SW4_cnt % 2 == 1) { EPwm5Regs.CMPB = 5000; } </pre>
--	---