



EDDI

Electronic Design
Development Institute

에디로봇아카데미

임베디드 마스터 Lv2 과정

제 1기

2021. 11. 26

김태훈

1. 주어진 회로도가 1차 측과 2차 측이 절연이 되어있지 않음.
현재는 문제가 없는데 MCU를 사용할 경우 문제가 생길 우려가 있어서 절연할 방법을 찾는 중.

2. MCU로는 wifi 통신하고 Deep learning을 위해서 ESP32계열을 쓸 예정

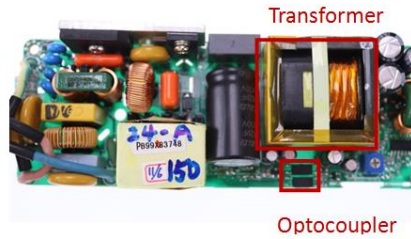


[Profile](#) | [Products](#) | [News](#) | [Service](#) | [Tech Notes](#) | [FAQ](#) | [Distributors](#) | [Contact Us](#) | [Career](#) | [SDG Group Partners Login](#) | [LANGUAGE](#) | [Q](#)

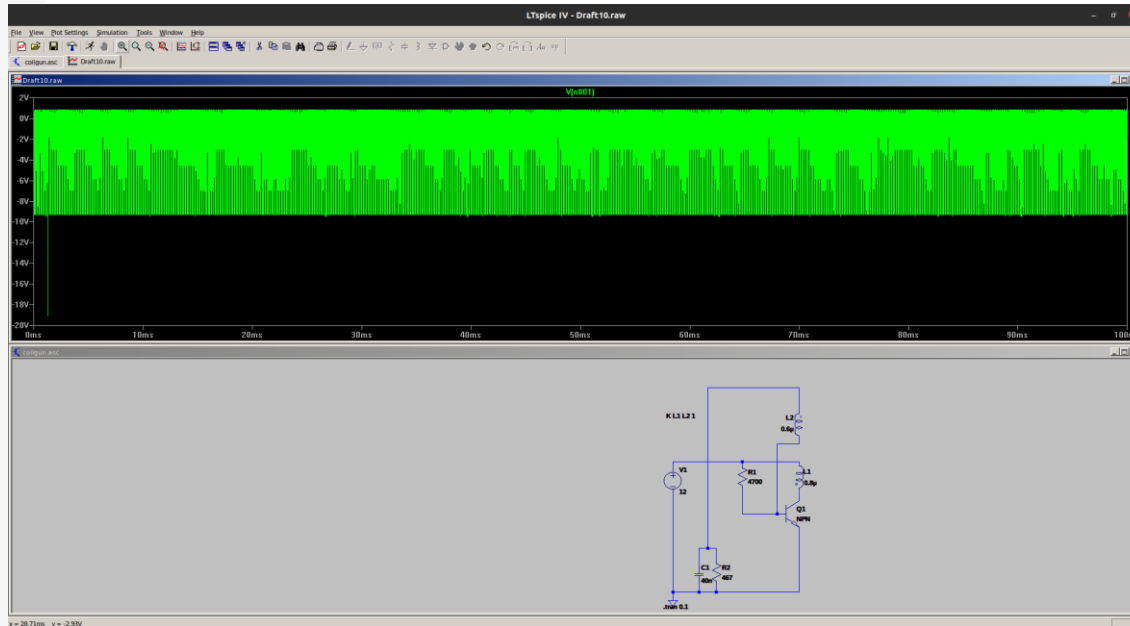
Attention: This article analyzes isolated and non-isolated LED drivers from both safety and life cycle aspects.

1. Safety:

As shown in the figure below, the primary and secondary sides of the isolated LED driver are isolated by a transformer and an optocoupler, and the insulation distance meets the requirements of safety regulations. When an abnormal situation occurs on the primary side, the high voltage will not flow to the secondary side directly, which will not cause any harm to users. It is a safe mechanism whether for users or lamp manufacturers.

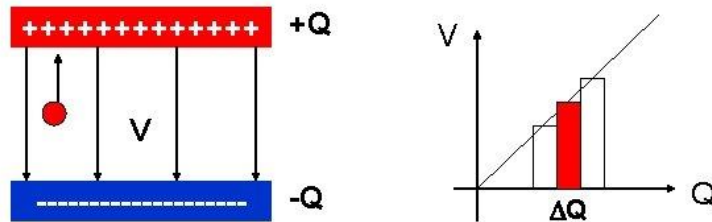


Isolation between primary and secondary sides



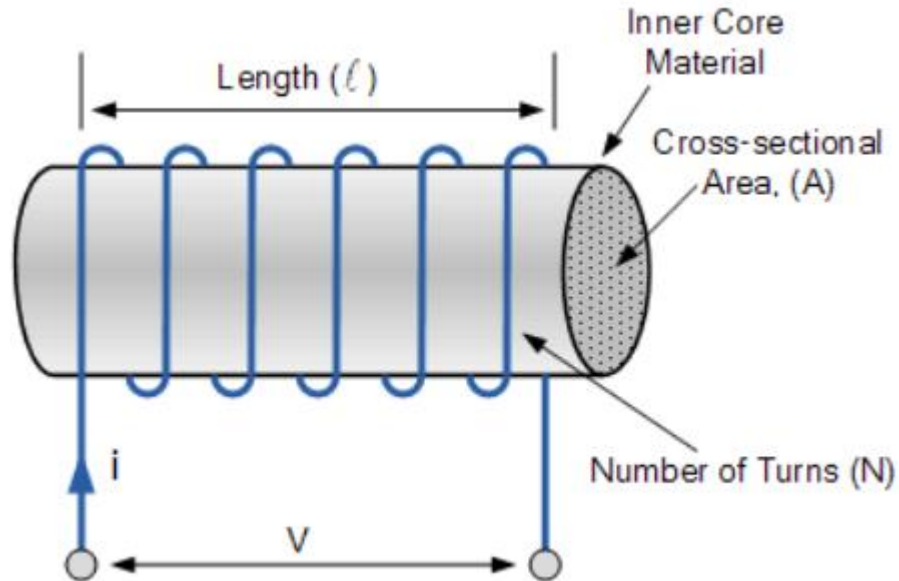
Inverter쪽 회로도 시뮬레이션
Base쪽 저항 값은 4700, 470 고정해야
한다
값을 바꾸면 발진하지 않는 경우가
생길 수 있으므로 주의.

energy stored in a capacitor



- the work done transferring a small amount ΔQ from - to + takes an amount of work equal to $\Delta W = V \Delta Q$
- At the same time, V is increased, since $V = (Q + \Delta Q) / C$
- The total work done when moving charge Q starting at $V = 0$ equals:
 $W = 1/2 QV = 1/2 (CV)V = 1/2 CV^2$
- Therefore, the amount of energy stored in a capacitor equals:

$$E_C = 1/2 C V^2$$



Equation

$$L_{coil} = \frac{\mu_r \mu_0 N^2 A}{l} = \frac{\mu_r \mu_0 N^2 \pi r^2}{l}$$

Where:

L_{coil} = inductance of the coil in henries (H)

μ_r = relative permeability of the core (dimensionless)

μ_0 = permeability of free space = $4\pi \times 10^{-7}$ (H/m)

N = number of turns

A = coil area (m^2)

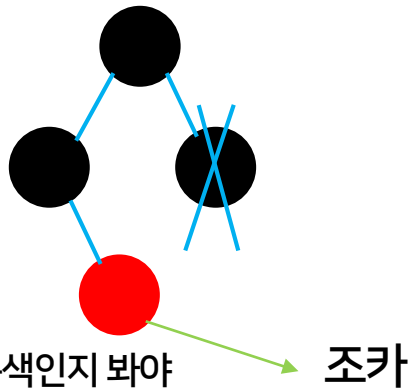
r = coil radius (m)

l = coil length (m)

RED BLACK TREE

삭제

1. RED를 삭제하고 그 자리에 NIL이 오는 경우 : 아무것도 안 해도 된다
2. RED를 삭제하고 RED가 오는 경우 : 불가능(RED violation때문)
3. RED 삭제하고 Black이 오는 경우 : 아무것도 안 해도 된다(설명 예정)
4. BLACK 삭제하고 RED가 오는 경우 : RED 색을 Black으로 바꾸면 된다
5. BLACK 삭제하고 NIL 이 오는 경우 : 제일 문제인데, 부모의 색과 형제의 색이 검은색인지 봐야 하고, 조부모까지도 회전할 수 있음
6. BLACK 삭제하고 BLACK이 오는 경우 : BLACK과 BLACK이 연달아 오는 것은 ROOT 근처가 아니면 흔하지 않은데, 아직 고민하지 못했습니다.

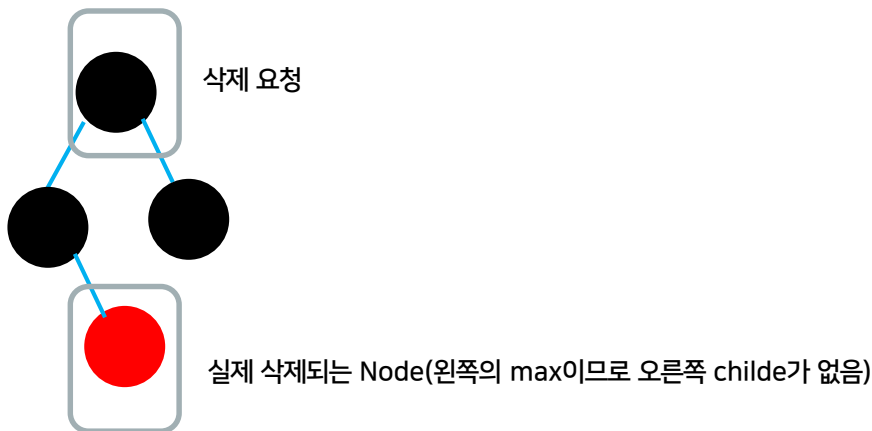


시간관계상 더 알아내지 못했습니다.

RED BLACK TREE

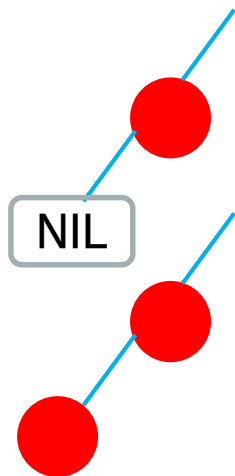
삭제의 전제

1. 숫자 X를 삭제한다고 X가 있는 node가 삭제되는 것이 아니라
왼쪽 tree의 max가 삭제된다는 것을 잊지 말기(Binary Tree의 delete)
2. 1)의 삭제과정에서, 삭제되는 node는 오른쪽 child가 없을 수 밖에 없음
(여기선 NIL일 수 밖에 없음)



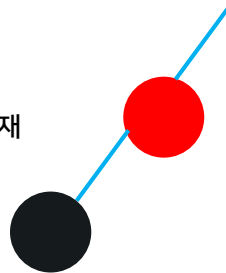
RED BLACK TREE

1. RED를 삭제하고 그 자리에 NIL이 오는 경우 : 아무것도 안 해도 된다
2. RED를 삭제하고 RED가 오는 경우 : 불가능(RED violation때문)
3. RED 삭제하고 Black이 오는 경우 : 아무것도 안 해도 된다(설명 예정)



Black의 level을 생각해보면 어차피 0이고,
Parent입장에서도 왼쪽 black의 level은 달라지지 않음.

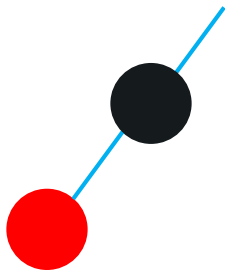
이미 RED violation이므로 insert 함수에서 처리되어서 존재
하지 않는 경우.



RED 삭제하고 BLACK이 오는 것도
사실 NIL과 같다고 보면 된다
부모입장에선 black level이 변하지 않으므로
아무것도 안 해도 됨.

RED BLACK TREE

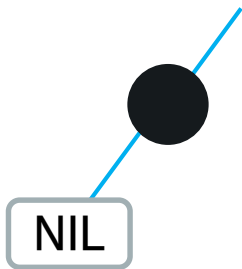
- 4. BLACK 삭제하고 RED가 오는 경우 : RED 색을 Black으로 바꾸면 된다
- 5. BLACK 삭제하고 NIL 이 오는 경우 : 제일 문제인데, 부모의 색과 형제의 색이 검은색인지 봐야 하고, 조부모까지도 회전할 수 있음
- 6. BLACK 삭제하고 BLACK이 오는 경우 : BLACK과 BLACK이 연달아 오는 것은 ROOT 근처가 아니면 흔하지 않은데, 아직 고민하지 못했습니다.



Black을 삭제하는 순간 Black level이 1 감소하지만
어차피 Black의 부모는 RED일 가능성이 높고,
RED를 그대로 두면 RED Violation이 뜨므로
색을 바꾸거나 회전 해야 하는데
색을 검은색으로 바꾸면 Black violation도 사라지므로
색만 바꾸면 된다.

RED BLACK TREE

5번 케이스는 구현 중인데, 아직 완벽하게 구현하지 못했습니다.
제 생각에는 부모 색이 검정색일 때, 조카가 존재할 때
이런 것들이 회전해야하는 문제가 되는 것 같습니다.



RED BLACK TREE

*** 코드는 지금 거의 스파게티 코드이므로 안 보시는 게 좋을 것 같습니다..