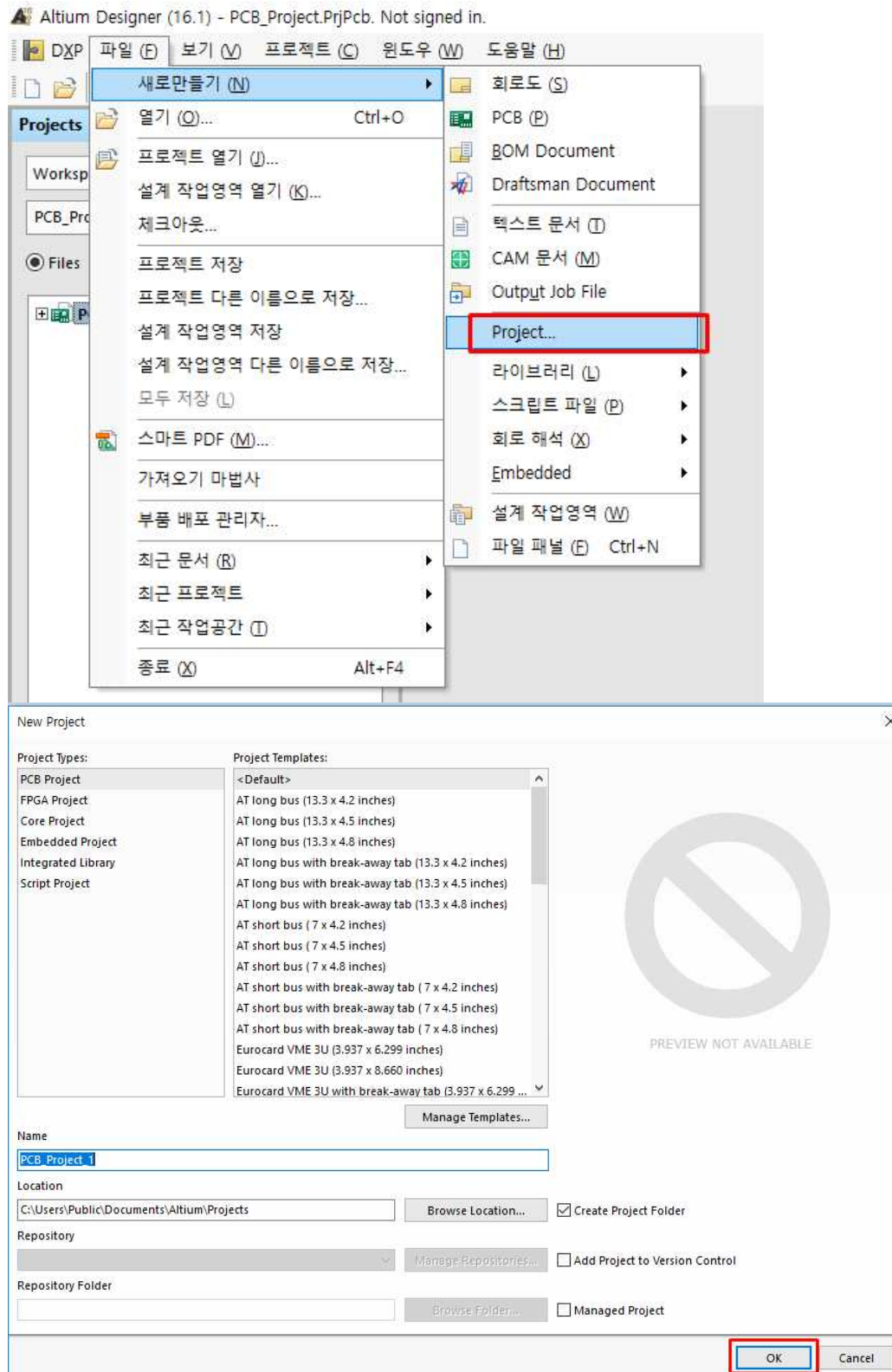


Xilinx Zynq FPGA, TI DSP, MCU 기반의 회로 설계 및 임베디드 전문가 과정

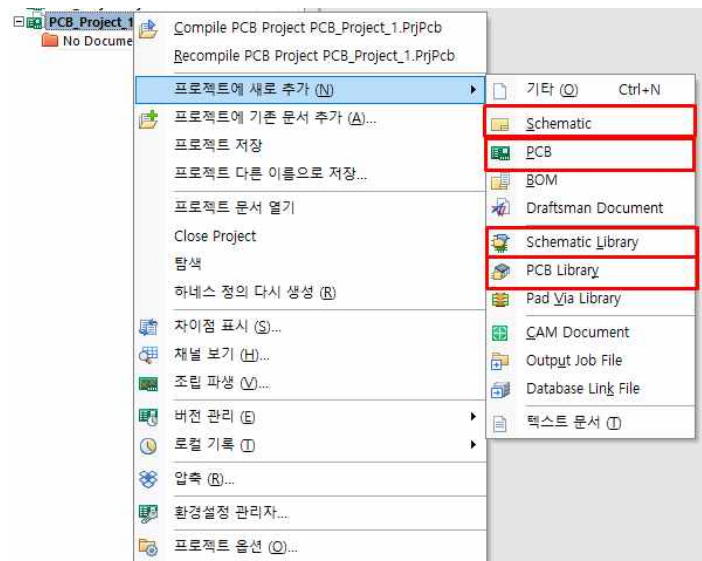
강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 - Hyungjoo Kim(김형주)
mihaelkel@naver.com

Schematic 시작하기

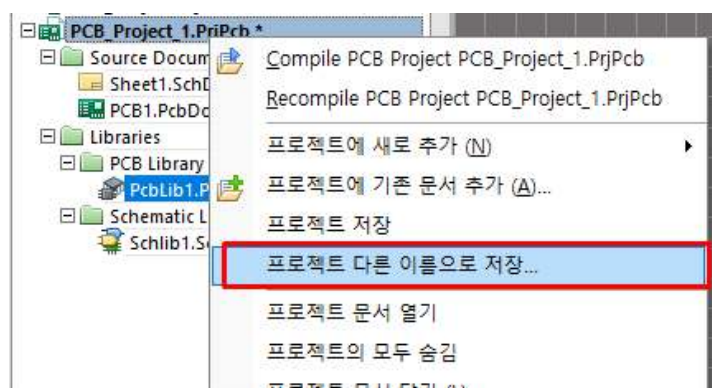
1.파일 > 새로만들기 > Project..



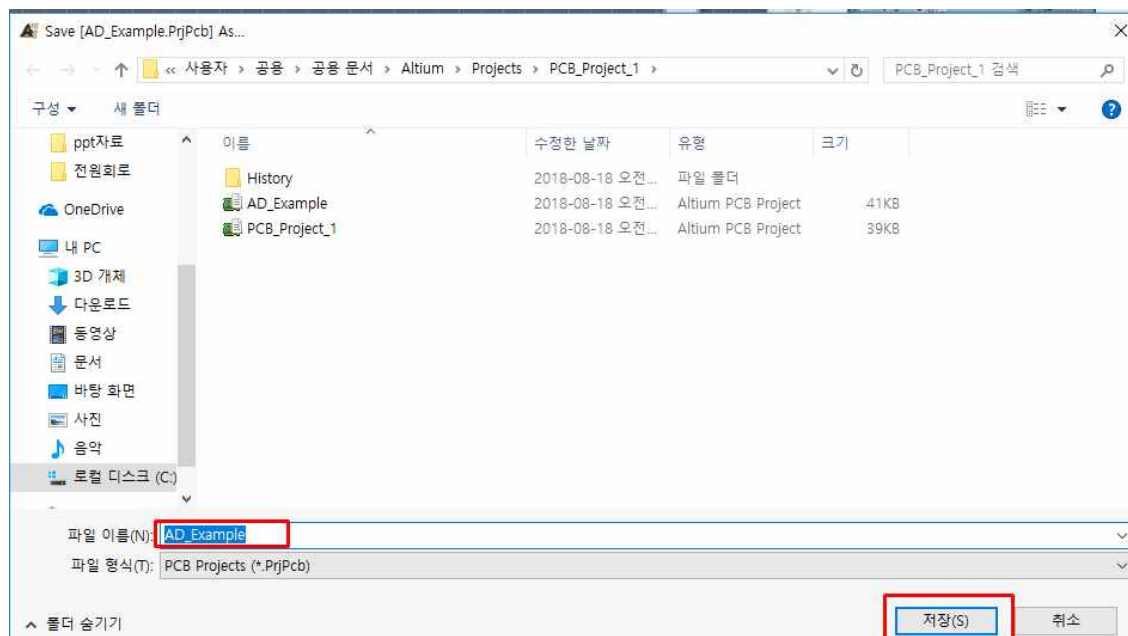
2. 프로젝트에 Schematic, PCB, Schematic Library, PCB Library를 추가한다.



다 추가하고 난 후, 프로젝트 다른 이름으로 저장...을 누른다.

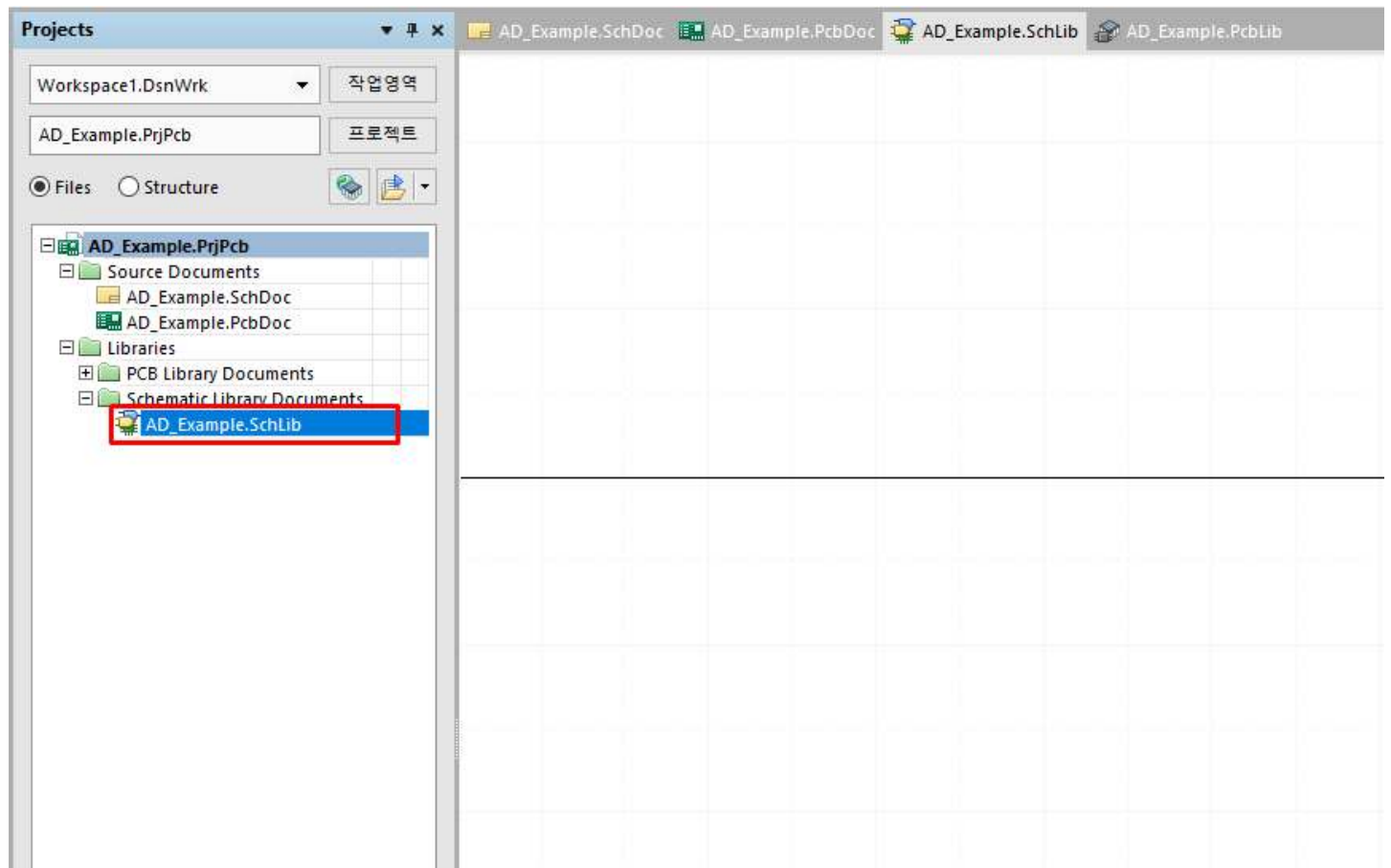


저장할 파일 이름을 전부 다 AD_Example로 지정한다.



3.이제부터 부품 라이브러리를 만들 것이다.

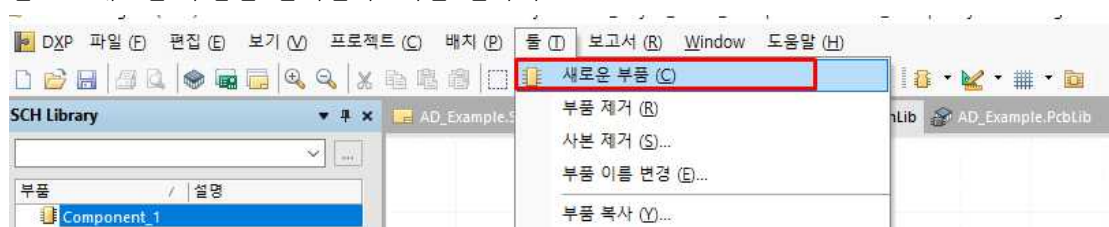
먼저, 아래와 같이 AD_Example.SchLib 파일을 연다.



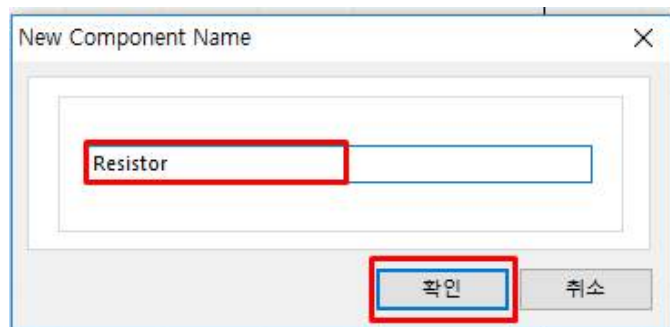
프로젝트 파일 목록 아래쪽에 있는, SCH Library를 클릭한다.



툴 -> 새로운 부품을 클릭한다. 혹은 단축키 T -> C



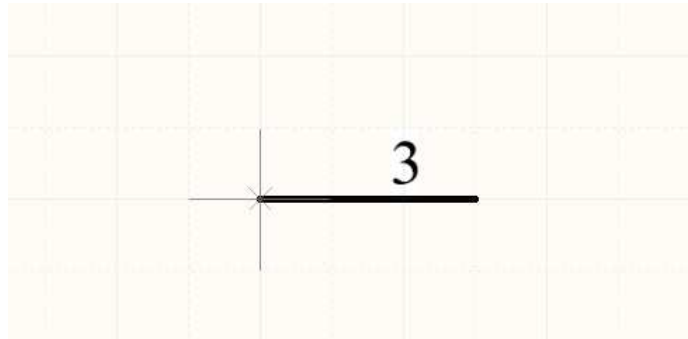
부품 이름을 Resistor라고 적고, 확인을 누른다.



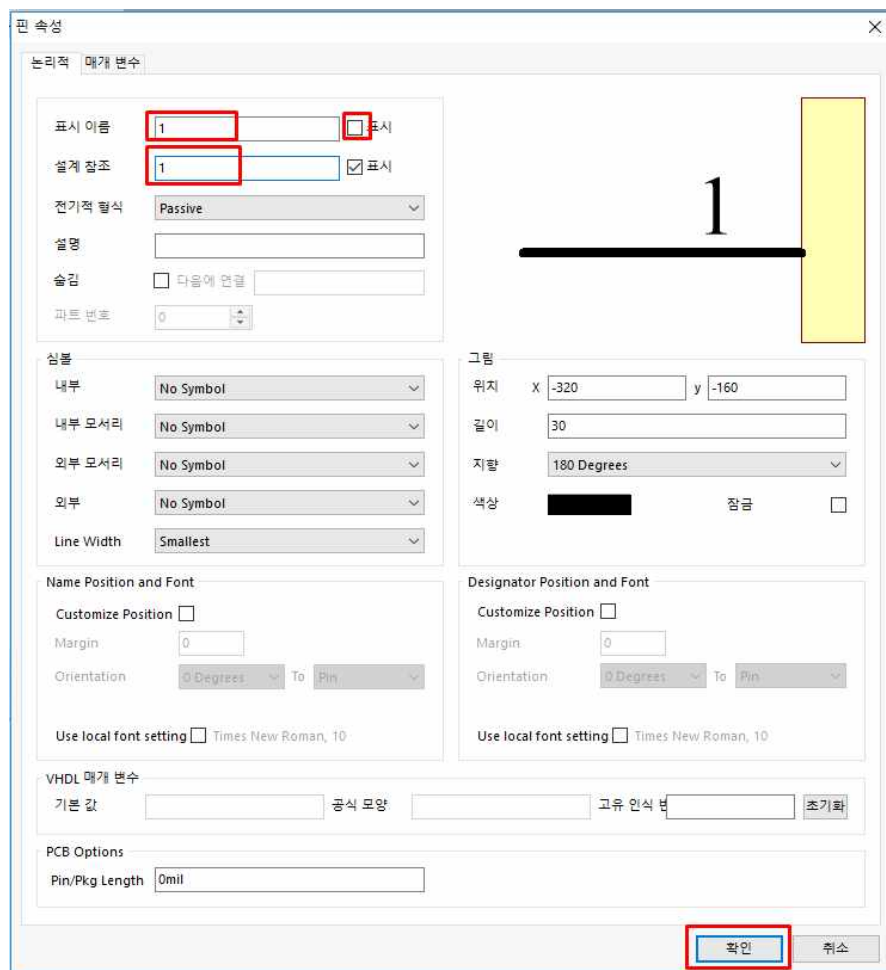
배치 -> 핀을 클릭하여 핀 설정을 할 것이다. 혹은 단축키 P -> P



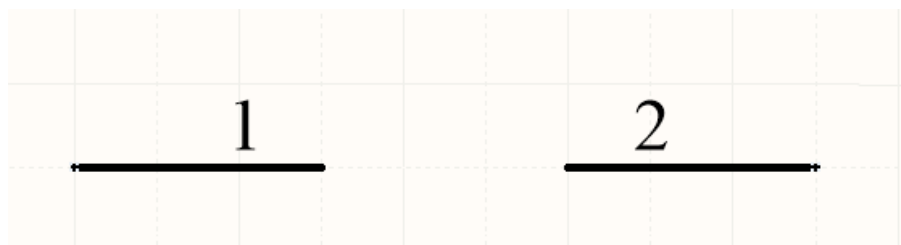
그러면 마우스 커서가 아래와 같이 핀으로 바뀐다.



이 상태로 Tap키를 눌러 속성을 바꿀 수 있다.

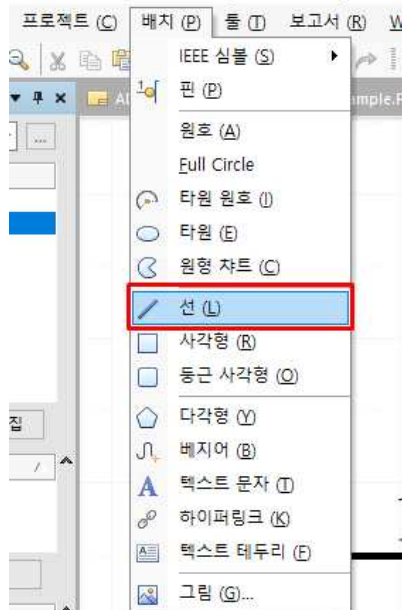


위와 같이 설정한 후, 확인을 누르고 핀을 아래와 같이 배치한다.

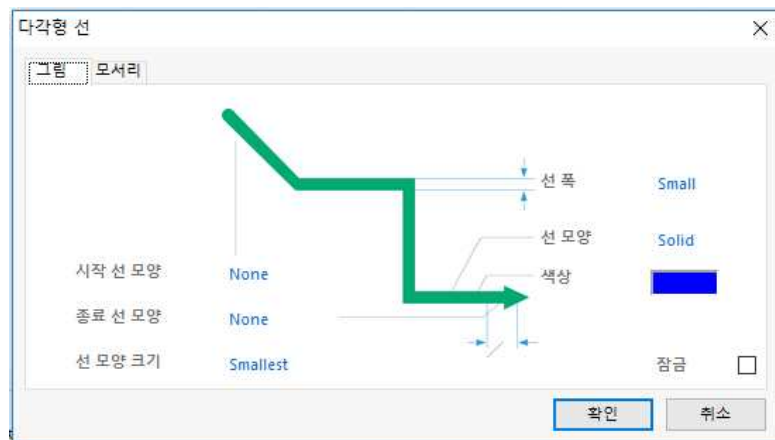


이 때, 핀의 방향에 유의한다.

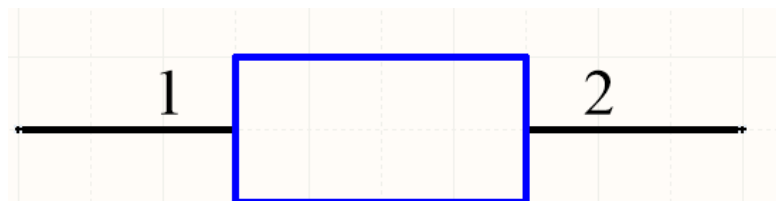
이제 선을 그려줄 것이다. 배치 -> 선 또는 단축키 P -> L을 누른다.



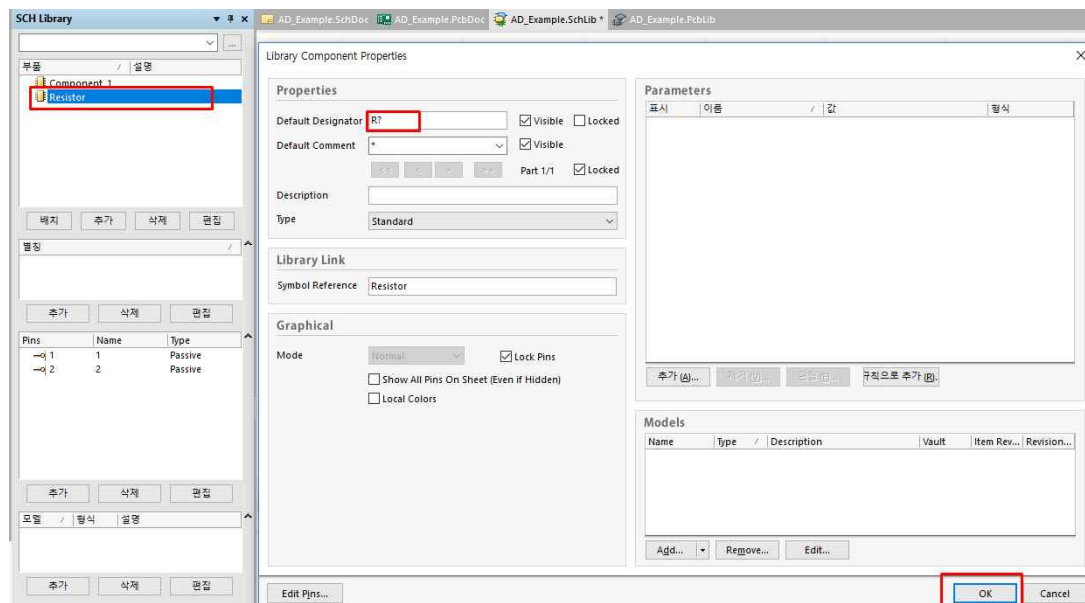
이 상태에서 Tap키를 누르면, 선의 모양, 굵기, 색상 등을 설정할 수 있다.



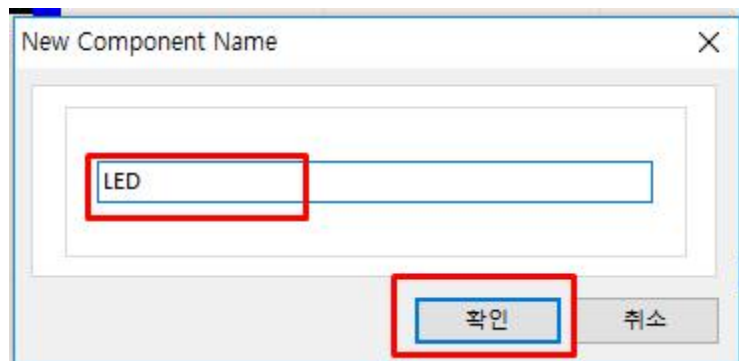
색상만 파란색으로 바꾼 후, 아래와 같이 그린다.



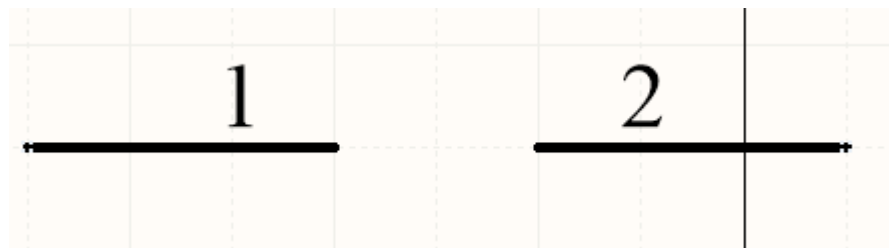
SCH Library에서 Resistor를 더블클릭한다. Designator를 R?로 변경 후, OK를 누른다.



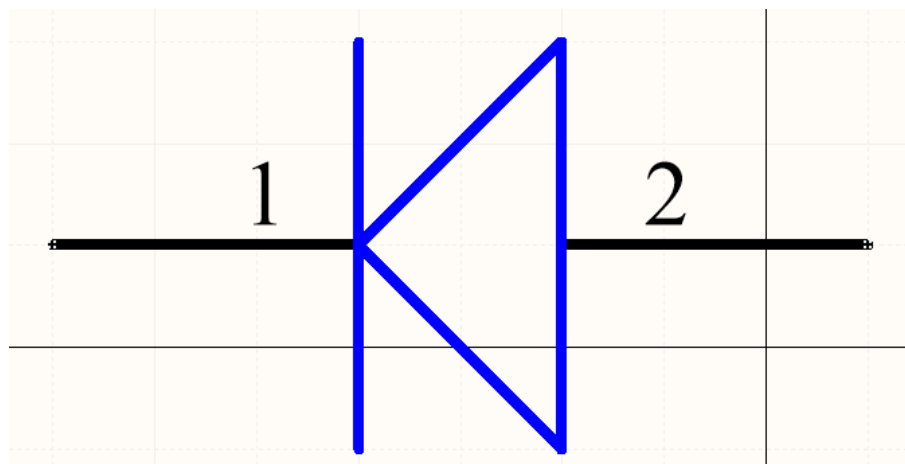
비슷한 방법으로 LED와 Connector를 만들 것이다.
T -> C 후 이름을 LED 라고 쓰고, 확인을 누른다.



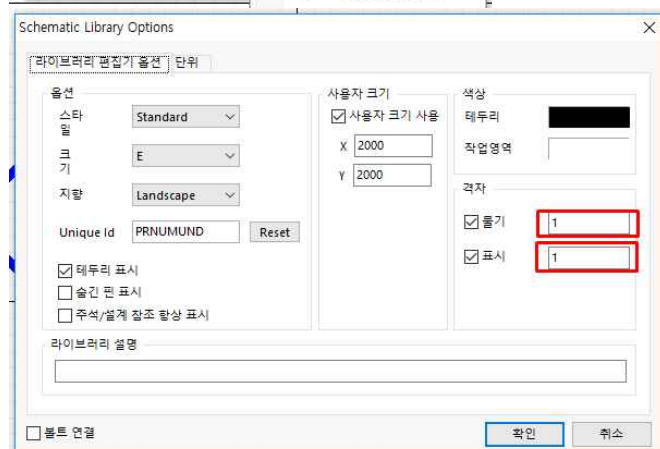
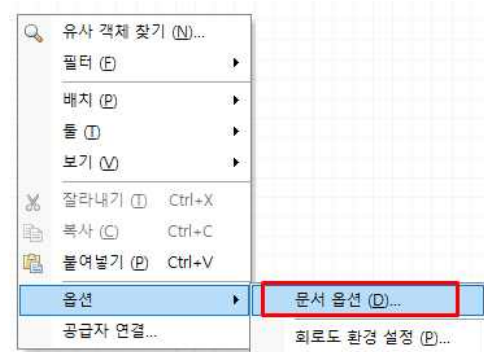
P -> P를 눌러 핀을 아래와 같이 배치한다.



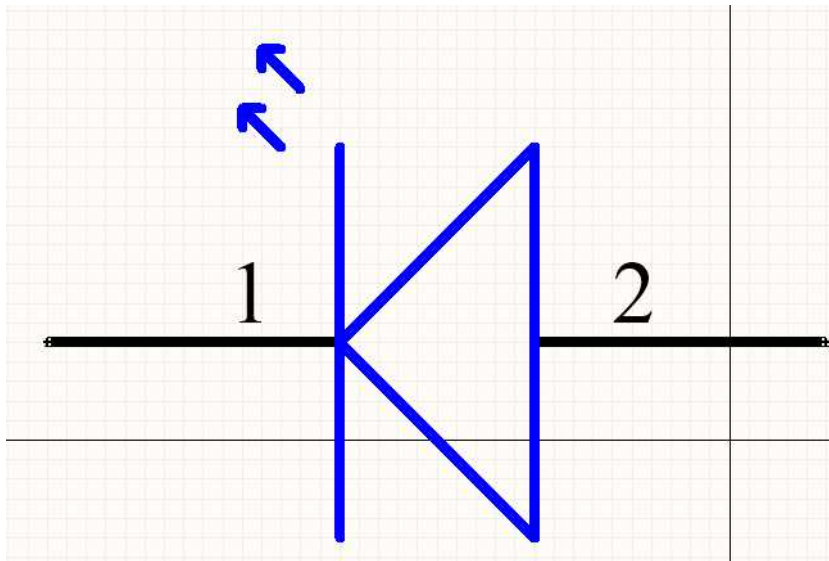
아래와 같이 LED 심볼을 그린다. 대각선을 그리고 싶다면, Space bar를 누르면 선의 방향을 바꿀 수 있다.



우클릭 -> 옵션 -> 문서옵션에서 격자 크기를 1로 변경한다.

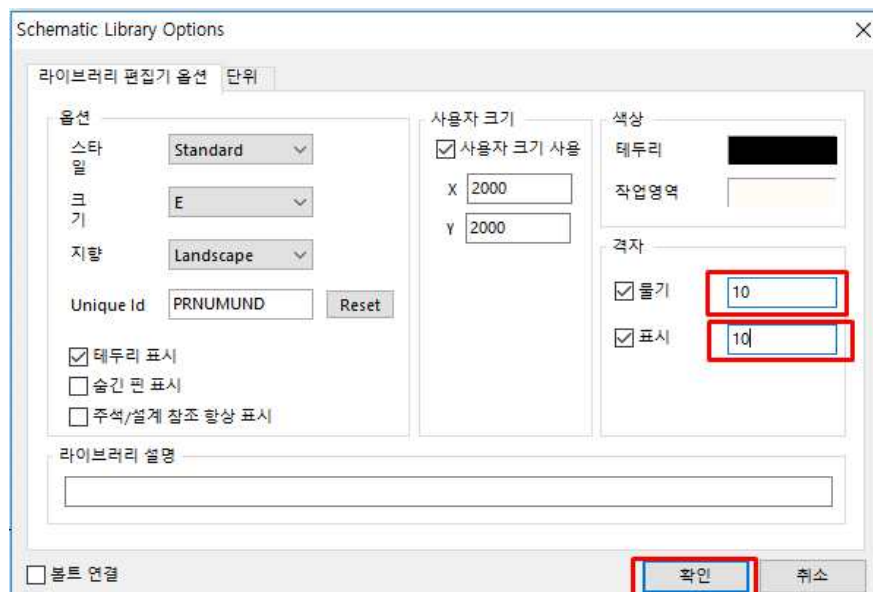


P -> L을 눌러 아래와 같이 심볼을 완성한다.

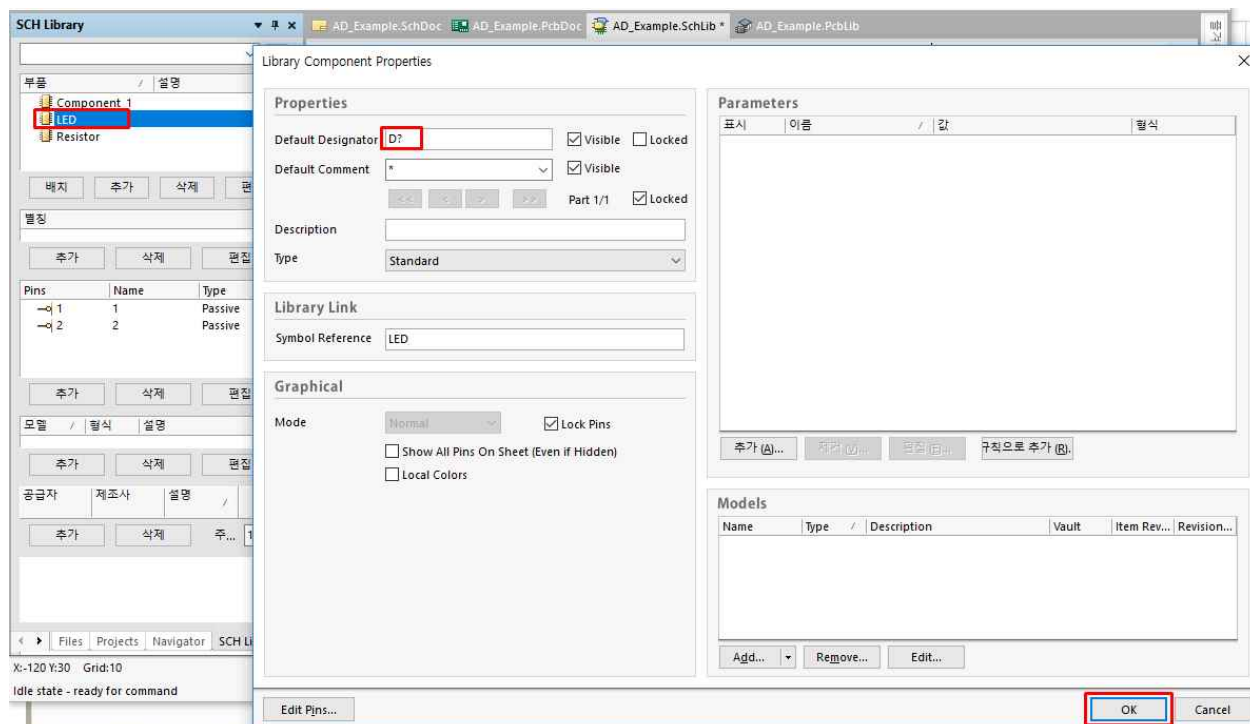


다 그렸다면, 변경했던 격자 크기를 다시 원상복구한다.

우클릭 -> 옵션 -> 문서옵션

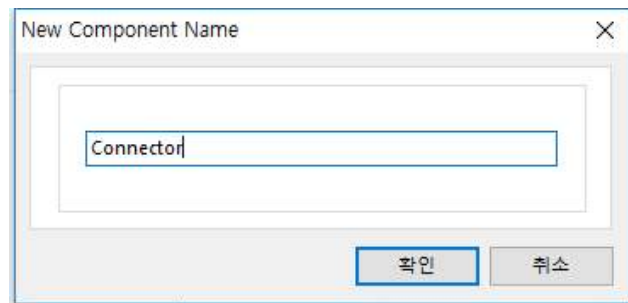


SCH Library의 LED를 더블클릭하여, Designator를 D?로 변경 후 OK를 누른다.

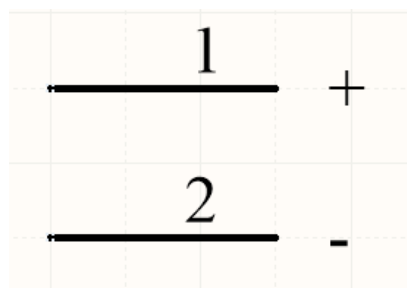
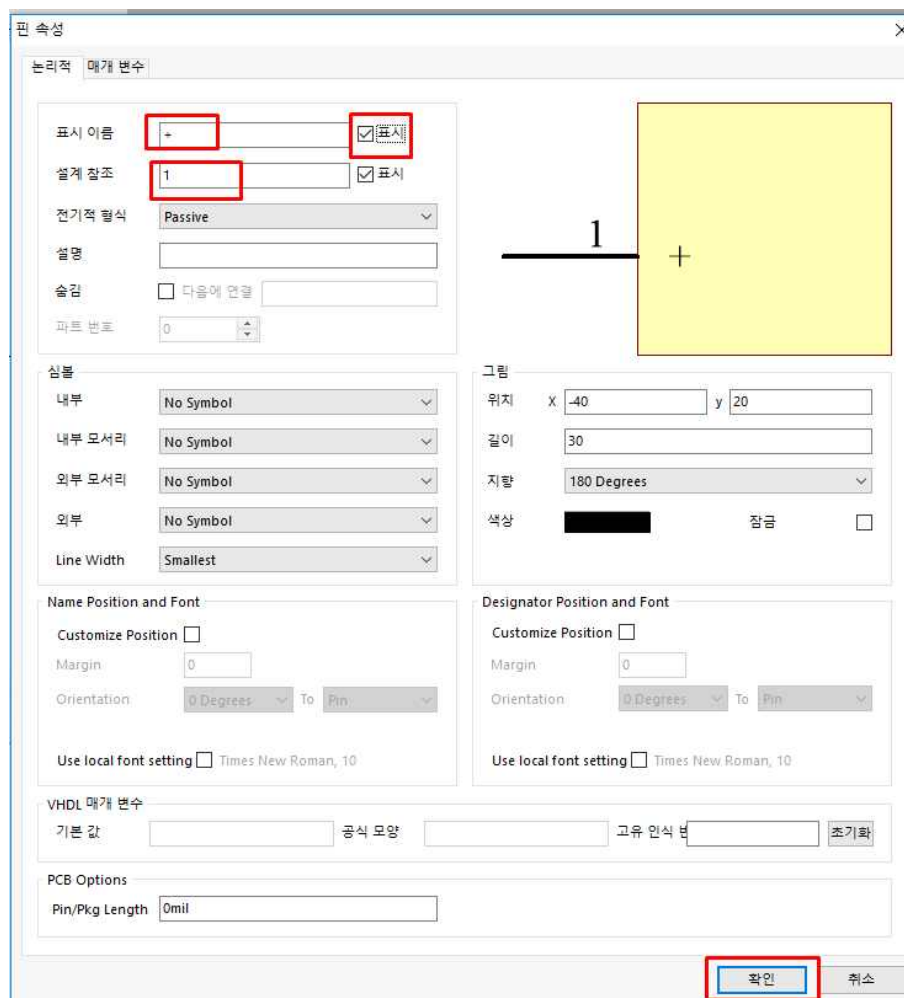


이제 Connector를 생성할 것이다.

T -> C를 누른 후, 이름을 Connector로 지정하여 새로운 부품을 생성한다.

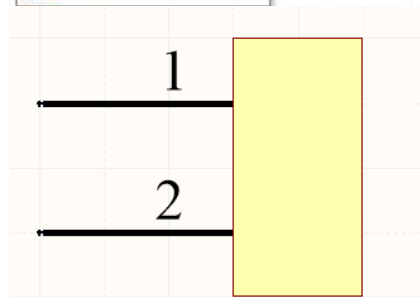
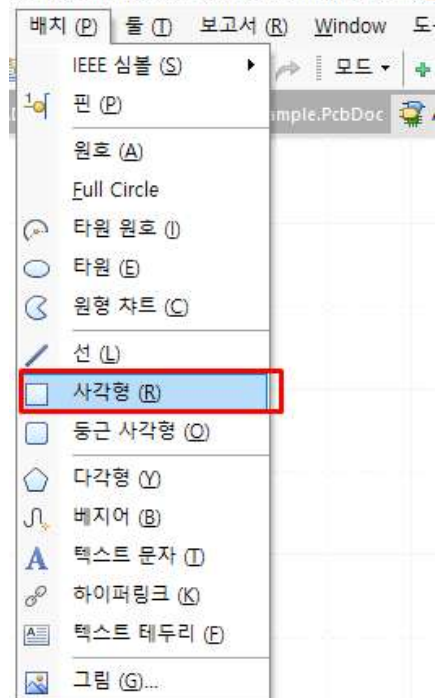


표시 이름을 하나는 +, 다른 하나는 -로 설정한다.



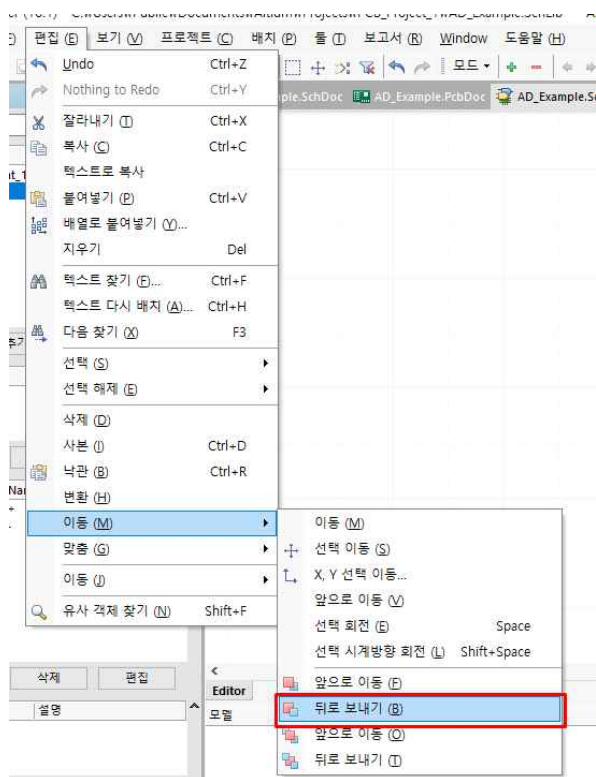
배치 -> 사각형 또는 P -> R을 눌러 사각형을 그린다.

s:\Altium\Projects\PCB_Project_1\AD_Example

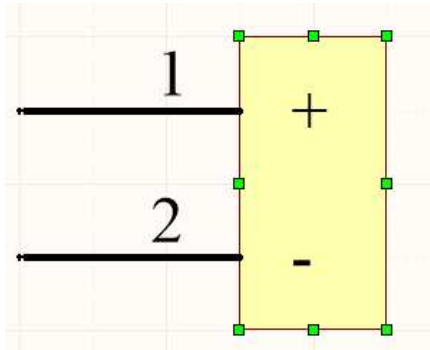


핀 표시가 보이지 않게 되었으므로, 사각형을 맨 뒤로 옮긴다.

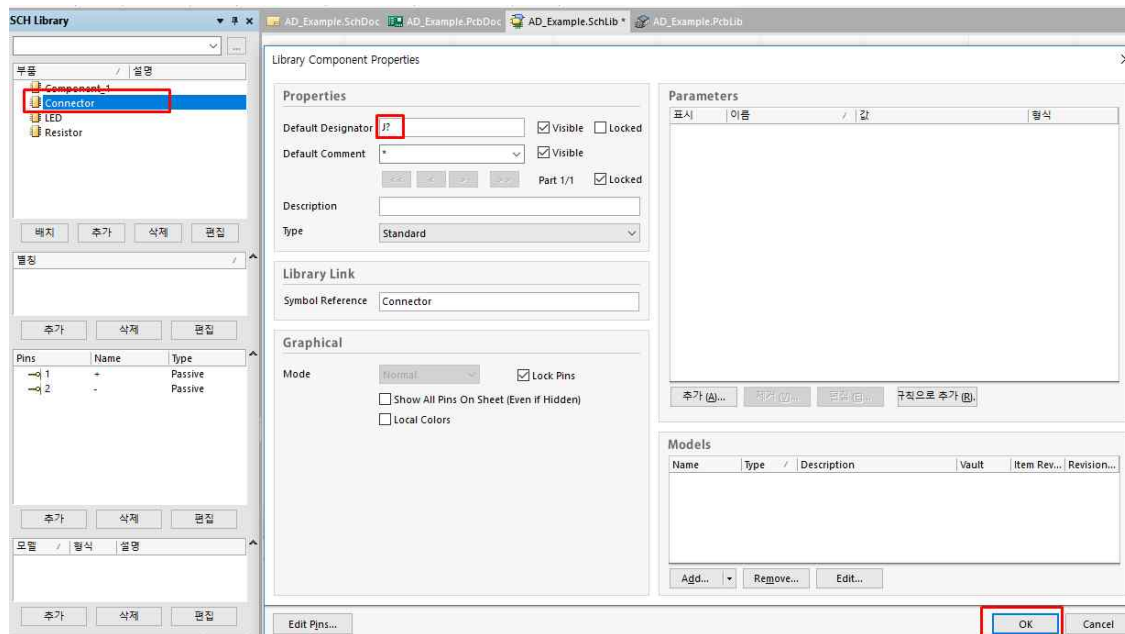
사각형이 선택된 상태로, 편집 -> 이동 -> 뒤로 보내기를 누른다.



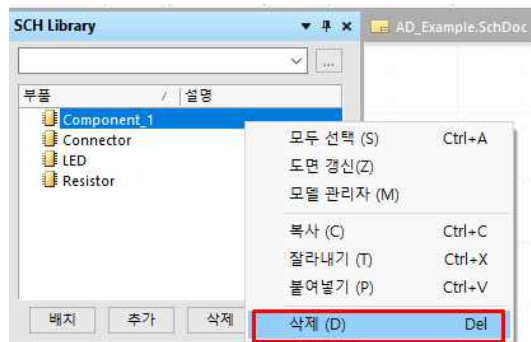
아래와 같이 극 표시가 다시 뜨는 것을 볼 수 있다.



Connector를 더블클릭하여 Designator를 J?로 변경한다.

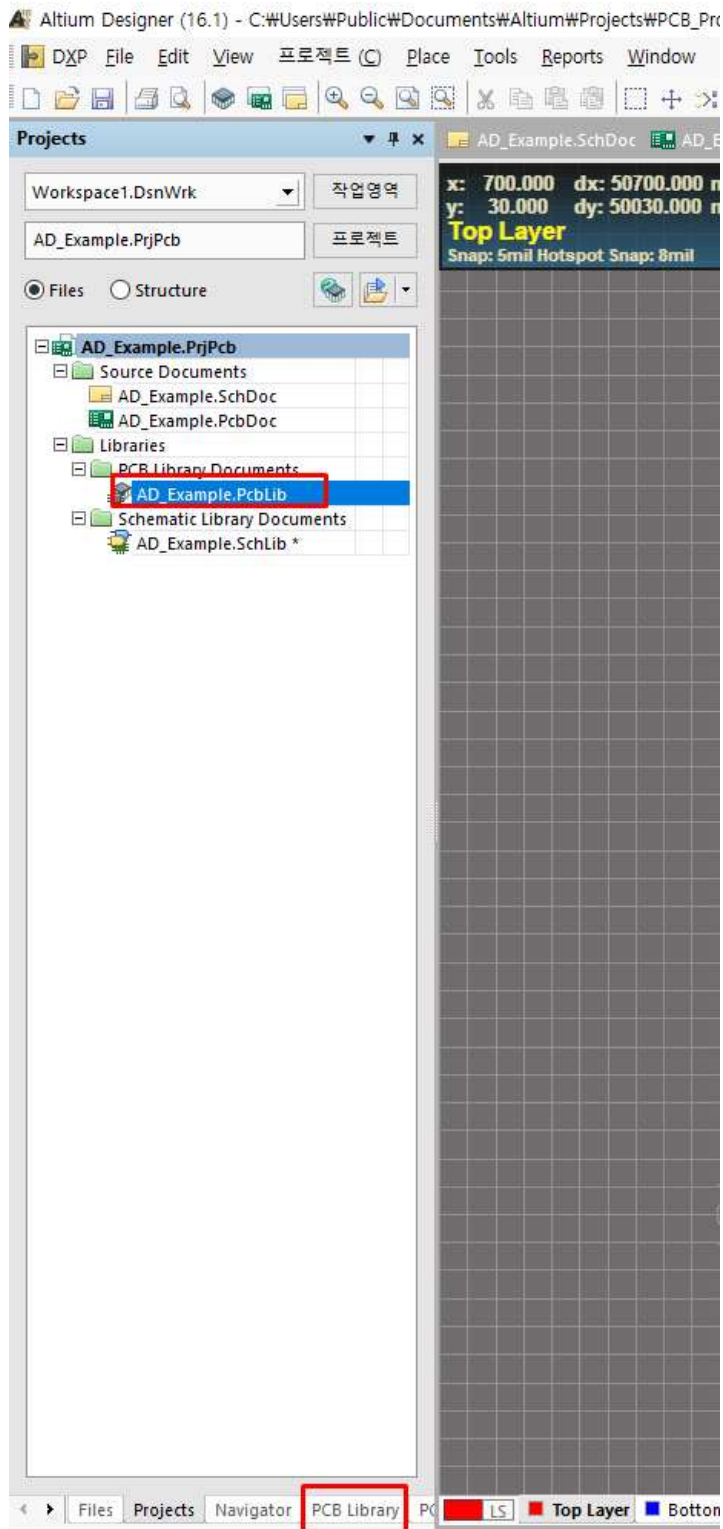


자동 생성된 Component_1은 필요없으므로, 우클릭 -> 삭제를 한다.

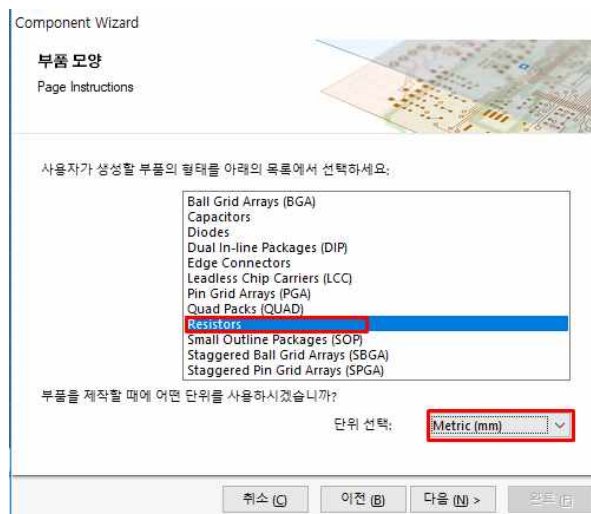
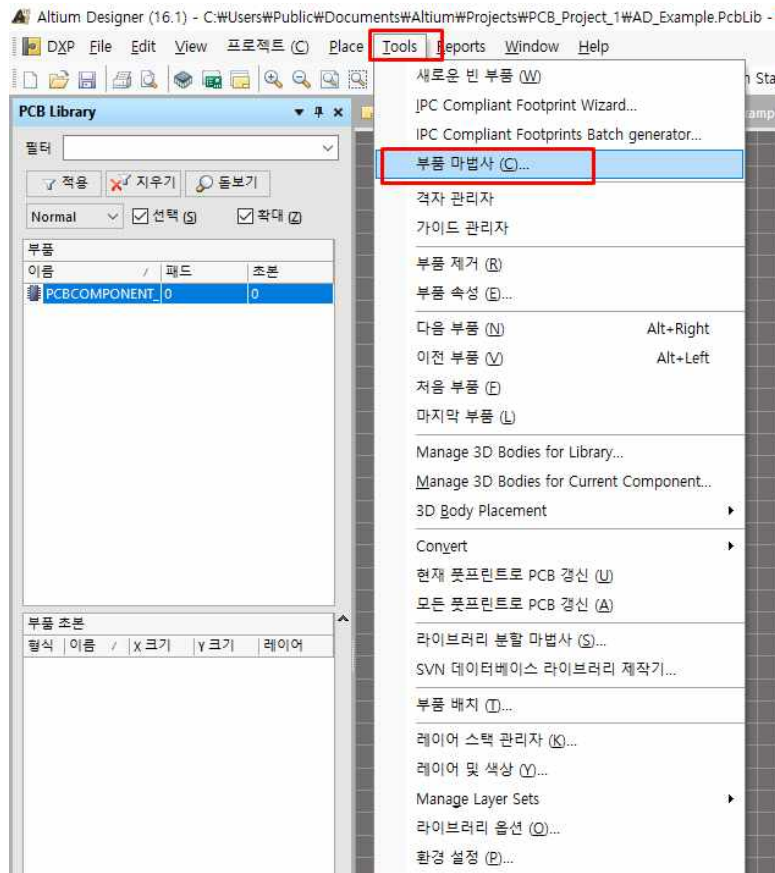


4. 이제 PCB 라이브러리를 만들 것이다.

프로젝트 창에서 PcbLib 파일을 선택한 후, pcb라이브러리를 만든다.




Tools -> 부품 마법사를 클릭한다.



부품 마법사
?
X

저항
기판 기술 정의




저항의 형식은 무엇입니까?
아래의 목록에서 기판 기술 선택:

Surface Mount

취소 (C)
이전 (B)
다음 (N) >
완료 (F)

부품 마법사
?
X

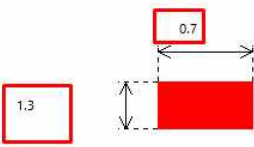
저항
패드 지수 정의



패드의 지수는 무엇입니까?
패드 지수의 값을 입력.

1.3


0.7



취소 (C)
이전 (B)
다음 (N) >
완료 (F)

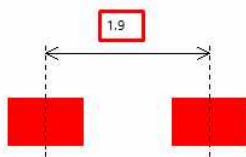
부품 마법사
?
X

저항
패드 형태 정의



패드의 상대적 위치는 어떻게 됩니까?
패드와 패드의 거리를 입력하세요.


1.9



취소 (C)
이전 (B)
다음 (N) >
완료 (F)

부품 마법사
?
X

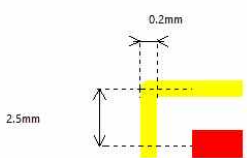
저항
외곽 지수 정의



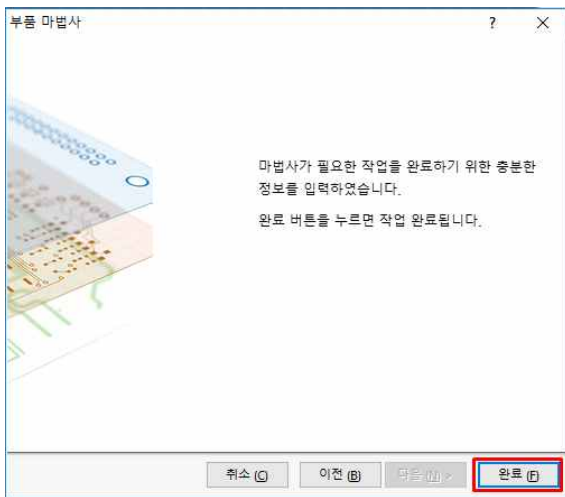
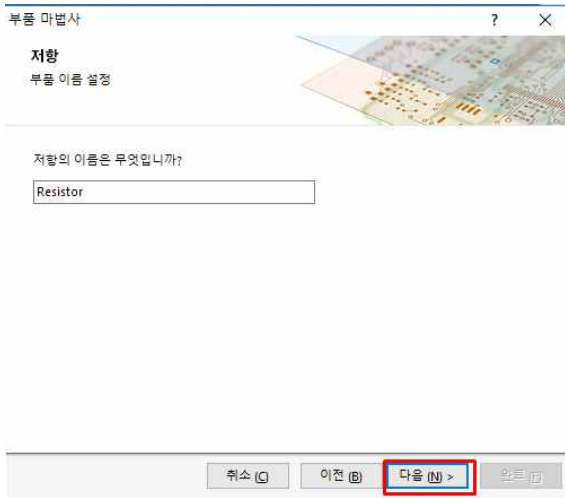
외곽의 높이와 폭은 얼마입니까?
외곽의 높이와 폭을 입력하세요.

0.2mm

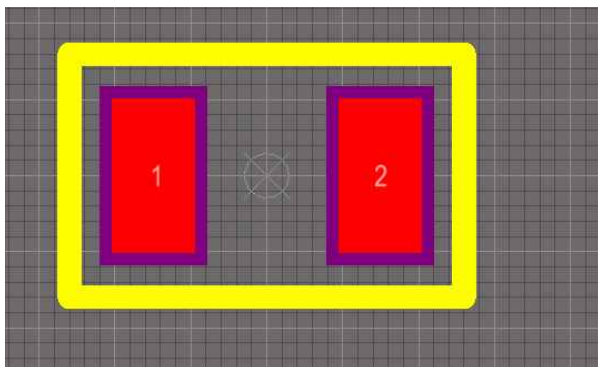
2.5mm



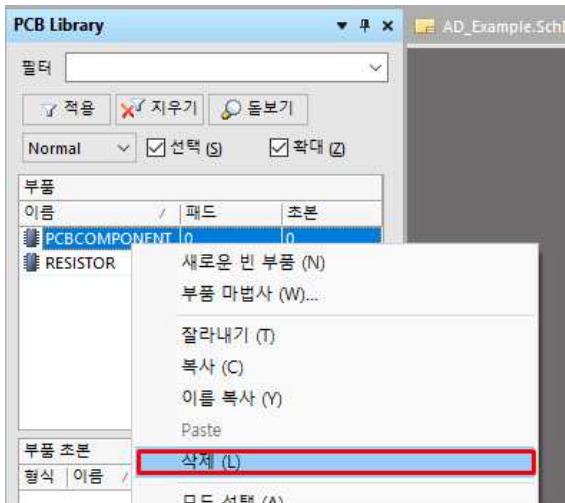
취소 (C)
이전 (B)
다음 (N) >
완료 (F)



아래와 같이 간격을 줄여준다.

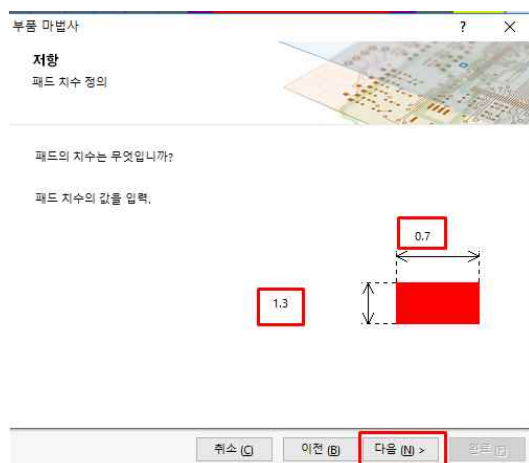
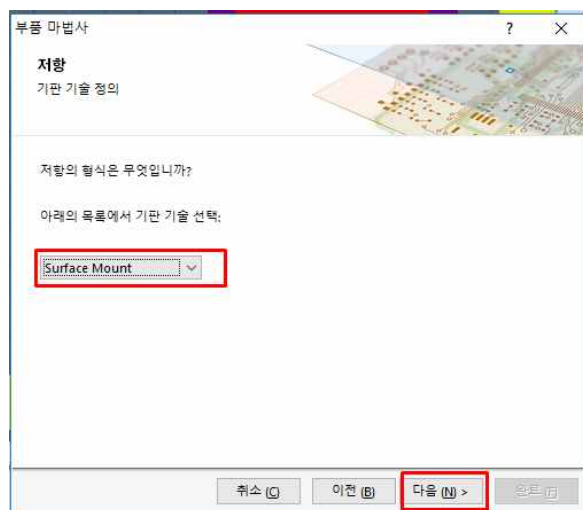
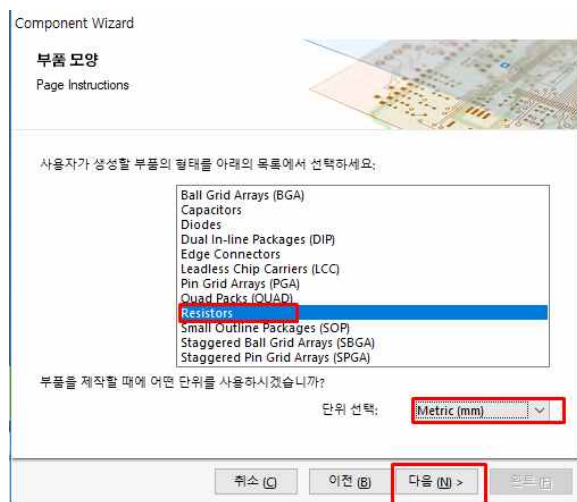
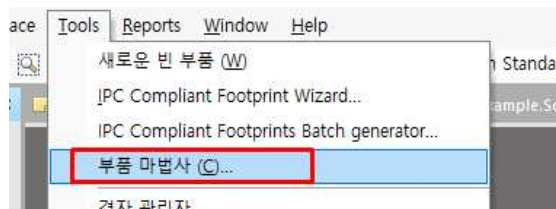


기본으로 만들어진 파일을 삭제한다.



부품 위자드를 다시 실행한다. 이번엔 LED를 만들 것이다.

documents\Altium\Projects\PCB_Project_1\AD_Example.PcbLib * - A



부품 마법사

?

×

저항

패드 형태 정의

패드의 상대적 위치는 어떻게 할까요?

패드와 패드의 거리를 입력하세요.

1.9

취소 (C)

이전 (B)

다음 (N) >

완료 (F)

부품 마법사

?

×

저항

외곽 치수 정의

외곽의 높이와 폭은 얼마입니까?

외곽의 높이와 폭을 입력하세요.

0.2mm

2.5mm

취소 (C)

이전 (B)

다음 (N) >

완료 (F)

부품 마법사

?

×

저항

부품 이름 설정

저항의 이름은 무엇입니까?

LED

취소 (C)

이전 (B)

다음 (N) >

완료 (F)

부품 마법사

?

×

마법사가 필요한 작업을 완료하기 위한 충분한 정보를 입력하였습니다.

완료 버튼을 누르면 작업 완료됩니다.

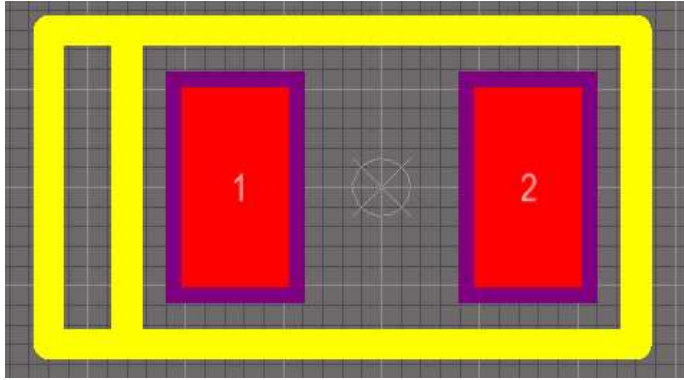
취소 (C)

이전 (B)

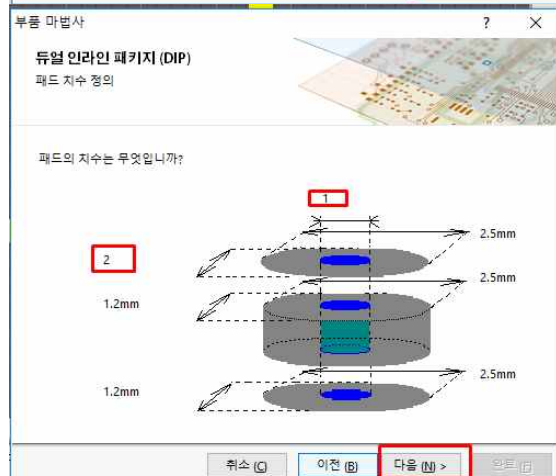
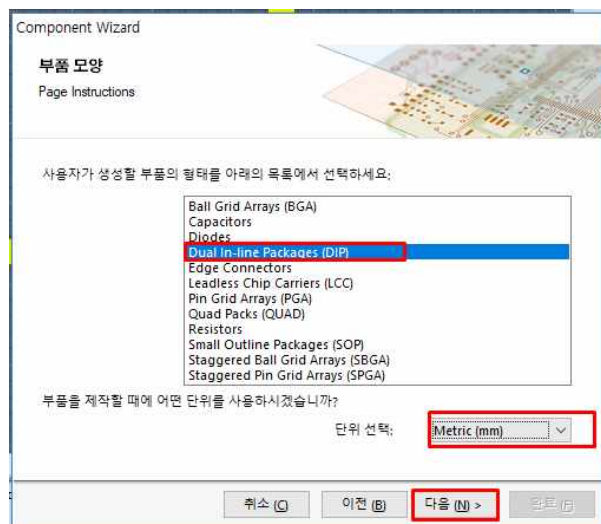
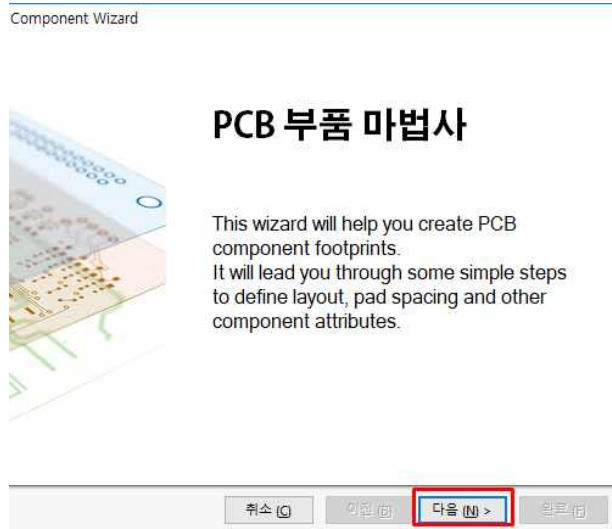
다음 (N) >

완료 (F)

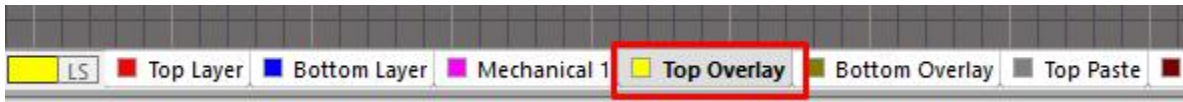
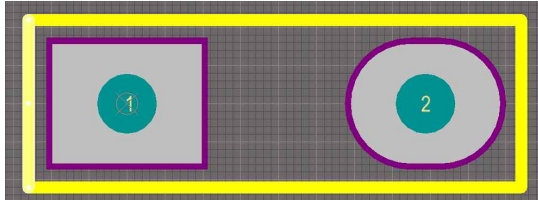
아래와 같이 그림을 수정한다.



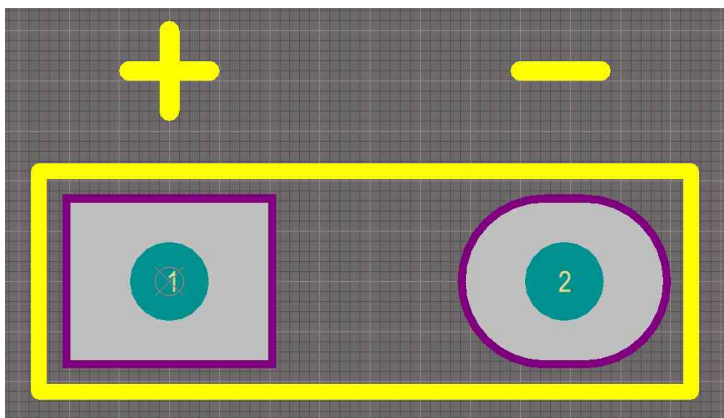
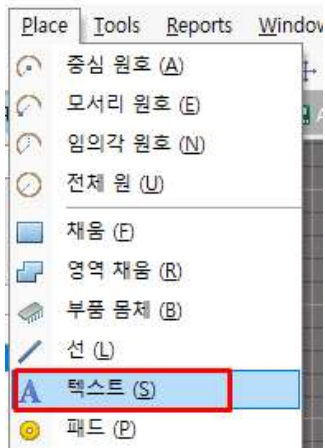
이번엔 Connector를 만들어준다. 똑같이 부품 마법사를 실행한다.



아래 그림과 같이 만들어준다.

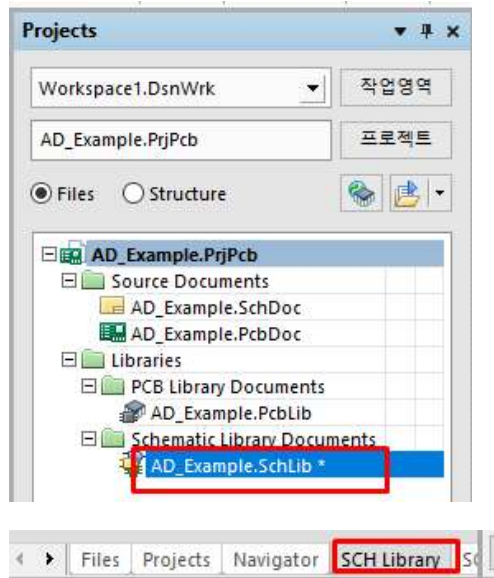


Top Overlay로 가서 Place -> 텍스트 혹은 P -> S 로 텍스트를 배치한다.

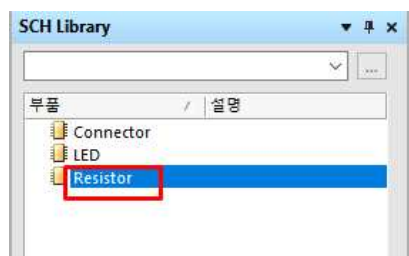


후에 Ctrl + s로 저장한다.

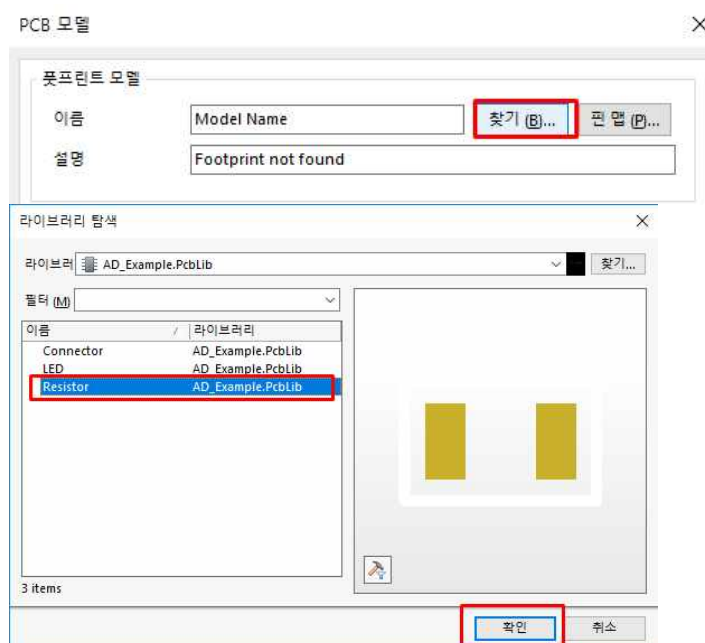
이제 footprint를 추가할 것이다. Projects 탭에서 .SchLib파일을 선택한다. SCH Library 탭으로 간다.



Resistor를 선택하고

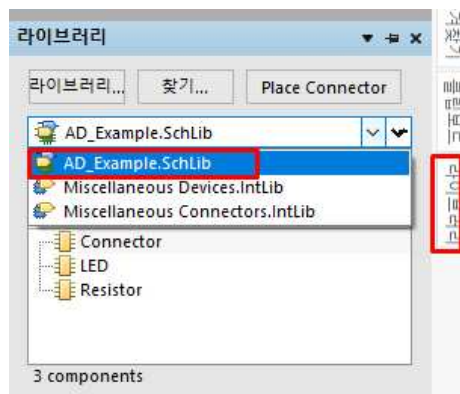
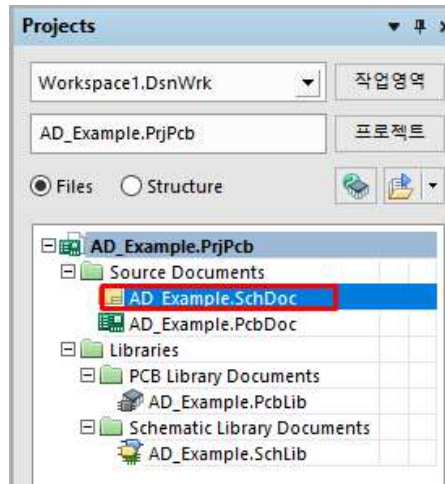


Add Footprint를 클릭한다.

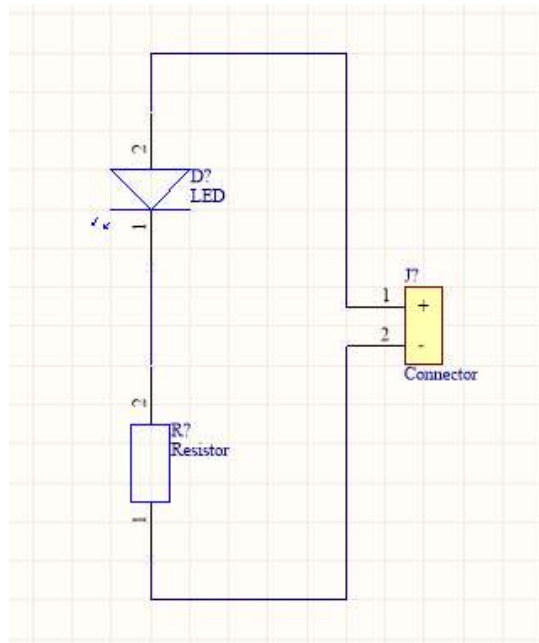


LED와 Connector에 대해서도 같은 작업을 반복한다.
후에 Ctrl + s를 통해 저장한다.

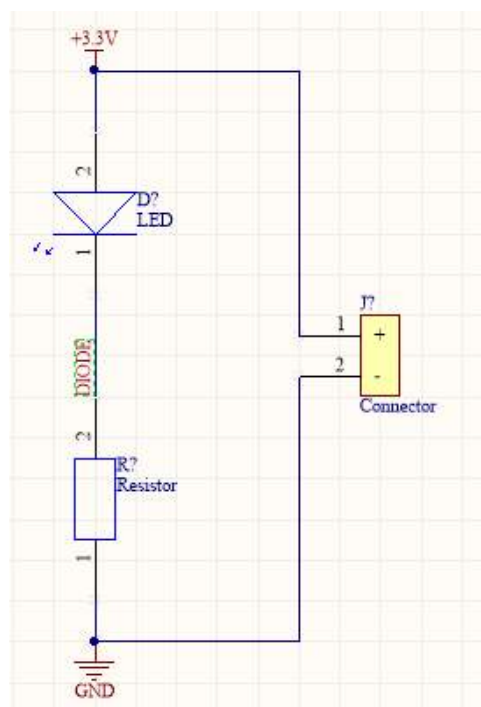
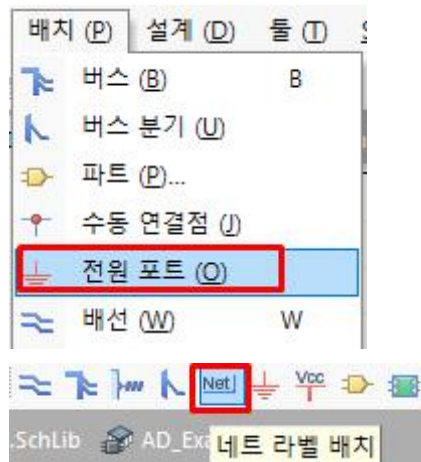
이제 Schematic에 이제까지 만든 부품 라이브러리를 배치할 것이다.



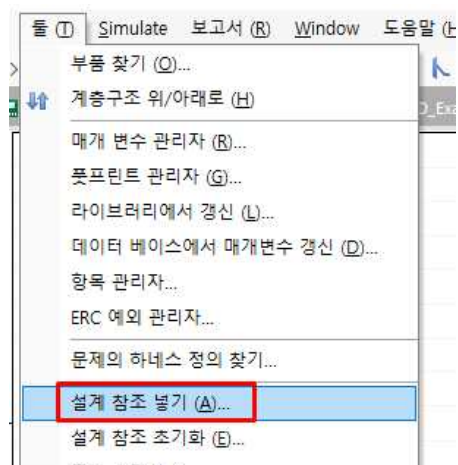
일단 아래와 같이 부품을 배치한다. 부품 회전은 부품을 들고있는 상태에서 Space bar, X, Y로 하면 된다.



전원 포트와 라벨을 아래와 같이 연결한다.



툴 -> 설계 참조 넣기



변경 목록

현재	적용 후	파트 위치
설계 참조	보조	설계 참조
<input checked="" type="checkbox"/> D?	<input checked="" type="checkbox"/> D?	회로도 도면
<input type="checkbox"/> J?	<input checked="" type="checkbox"/> J?	AD_Example.SchDoc
<input type="checkbox"/> R?	<input checked="" type="checkbox"/> R?	AD_Example.SchDoc

주석 달기 결과

Annotation is enabled for all schematic documents. Parts will be matched using 2 parameters, all of which will be strictly matched. (Under strict matching, parts will only be matched together if they all have the same parameters and parameter values, with respect to the matching criteria. Disabling this will extend the semantics slightly by allowing parts which do not have the specified parameters to be matched together.) Existing packages will not be completed. All new parts will be put into new packages.

변경 목록 갱신

Reset All

주석 취소 (B)

변경 확인(ECO 생성)

Information

Change(s) made

3 change(s) were made from previous state

3 change(s) were made from original state

OK

변경 목록

현재	적용 후	파트 위치
설계 참조	보조	설계 참조
<input checked="" type="checkbox"/> D?	<input checked="" type="checkbox"/> D1	AD_Example.SchDoc
<input type="checkbox"/> J?	<input checked="" type="checkbox"/> J1	AD_Example.SchDoc
<input type="checkbox"/> R?	<input checked="" type="checkbox"/> R1	AD_Example.SchDoc

주석 달기 결과

Annotation is enabled for all schematic documents. Parts will be matched using 2 parameters, all of which will be strictly matched. (Under strict matching, parts will only be matched together if they all have the same parameters and parameter values, with respect to the matching criteria. Disabling this will extend the semantics slightly by allowing parts which do not have the specified parameters to be matched together.) Existing packages will not be completed. All new parts will be put into new packages.

변경 목록 갱신

Reset All

주석 취소 (B)

변경 확인(ECO 생성)

닫기

기술 변경 명령

변경	대상	영향 문서	상태
활성	활동	영향 대상	검사
<input checked="" type="checkbox"/>	Annotate Component(3)		마침
<input checked="" type="checkbox"/>	Modify	D? -> D1	마침
<input checked="" type="checkbox"/>	Modify	J? -> J1	마침
<input checked="" type="checkbox"/>	Modify	R? -> R1	마침

변경 등록

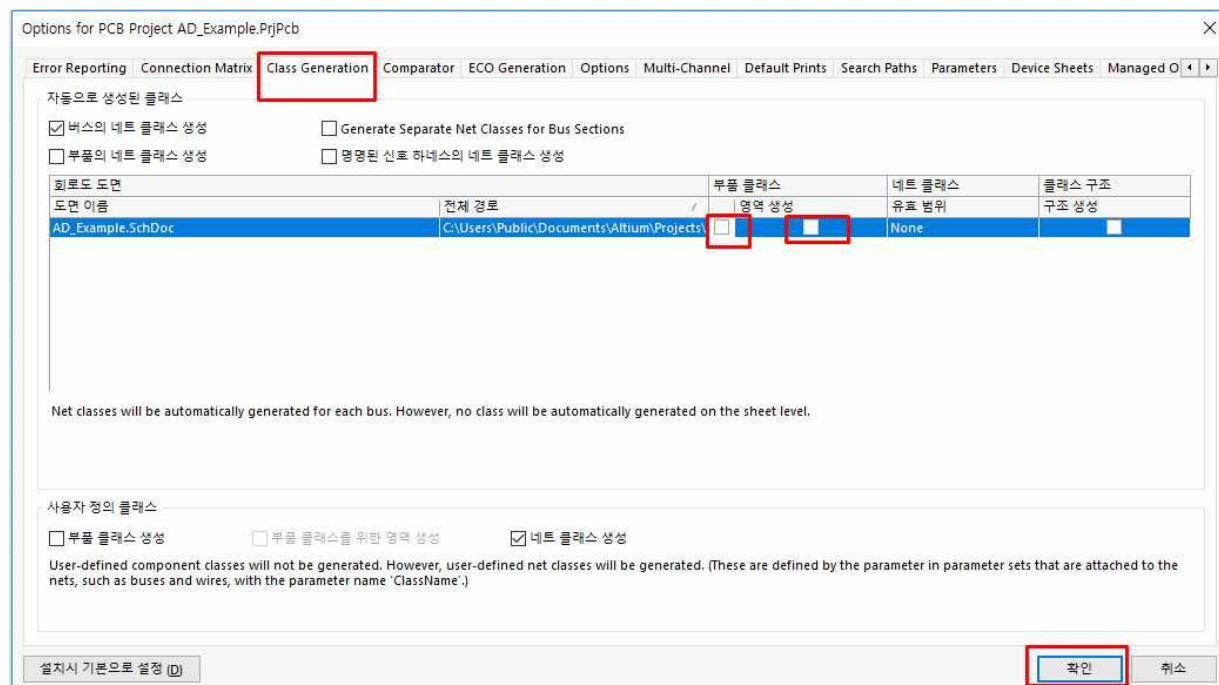
변경 실행

변경 보고서 (B)...

오류만 표시

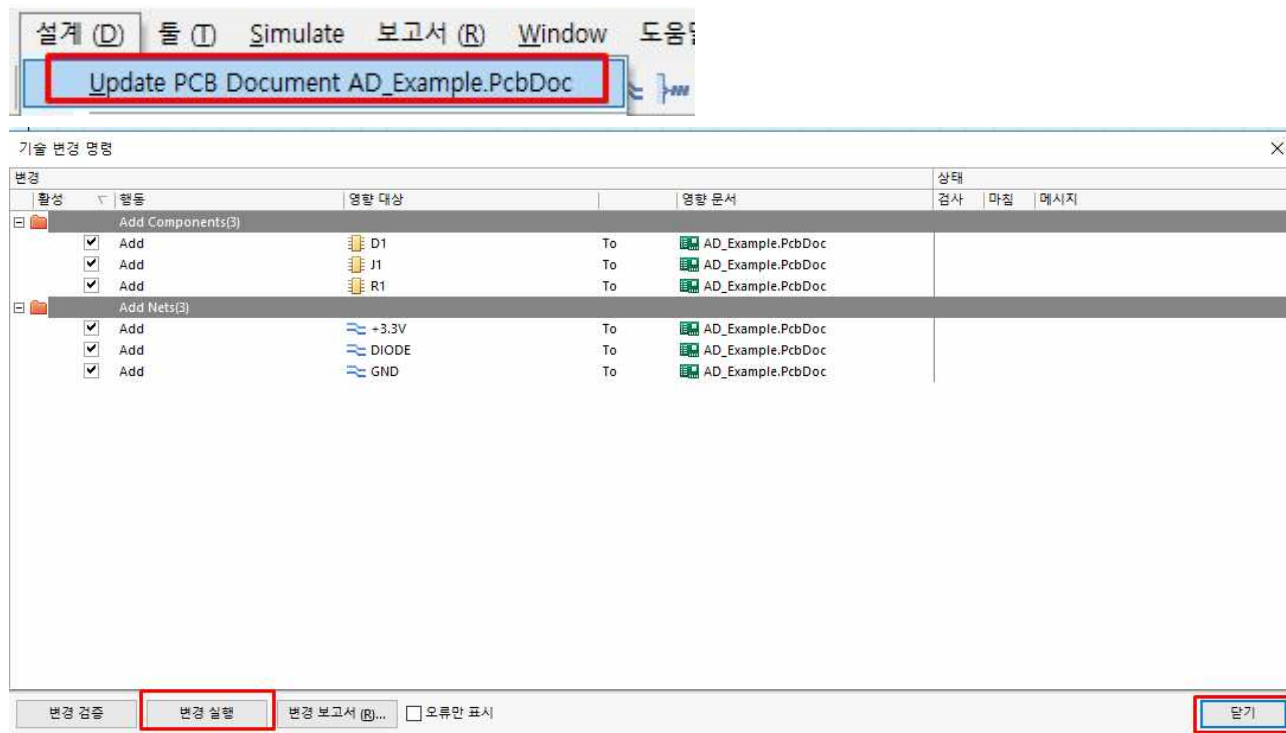
닫기

프로젝트 옵션을 변경해준다.

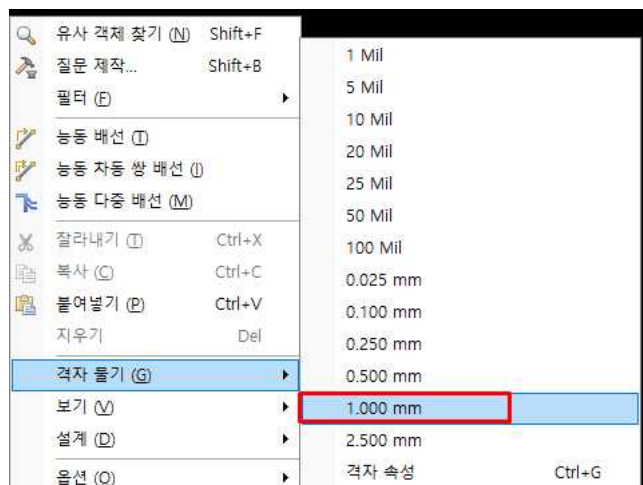


스케메틱에 있는 회로도를 pcb 파일에 갱신한다.

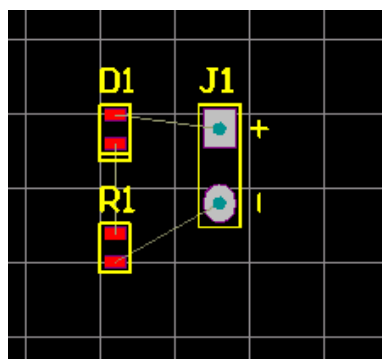
설계 -> Update PCB Document AD_Example.PcbDoc



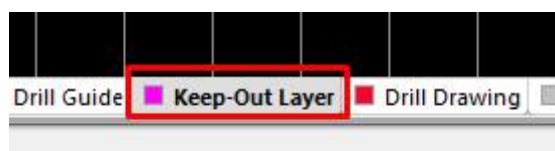
빈 공간에 우클릭을 하여 격자를 생성한다.



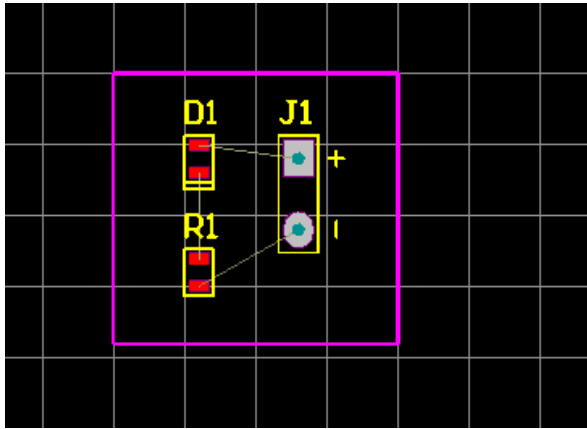
소자들을 아래와 같이 배치한다.



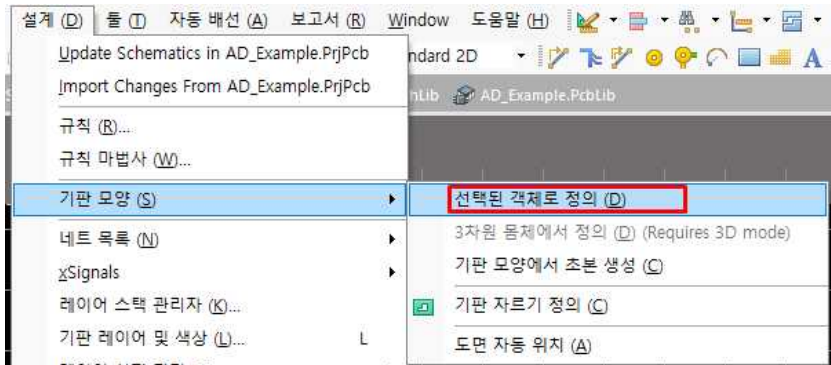
Kepp-Out Layer로 가서



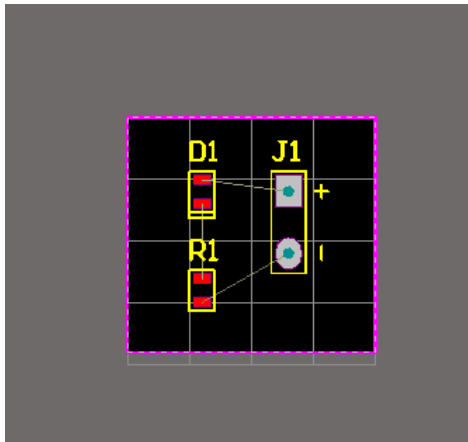
아래와 같이 선을 연결하여 기판의 크기를 결정한다.



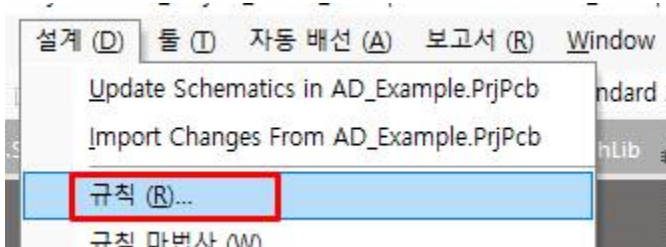
설계 -> 기판 모양 -> 선택된 객체로 정의를 클릭하면



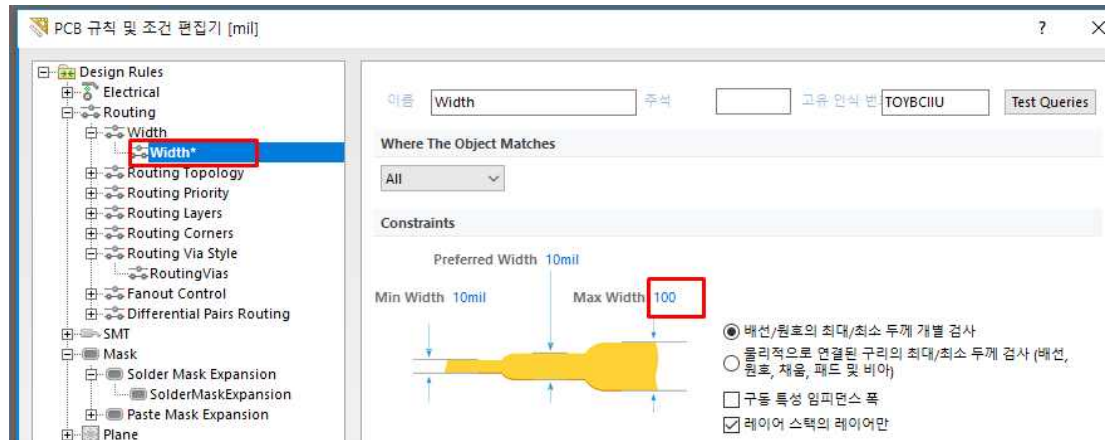
아래와 같이 기판의 크기가 변경된다.



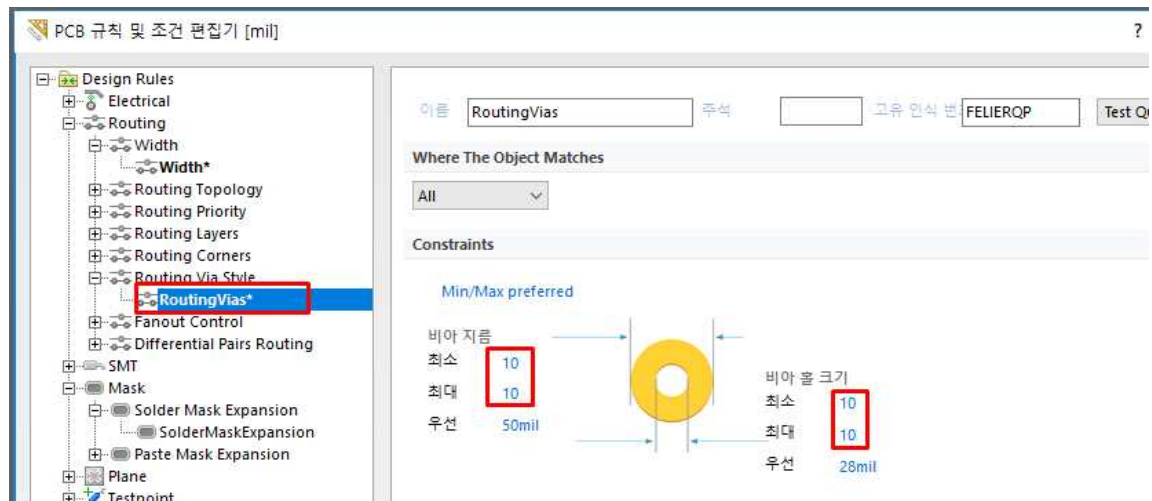
설계 -> 규칙으로 가서



Max width를 100으로 변경한다.



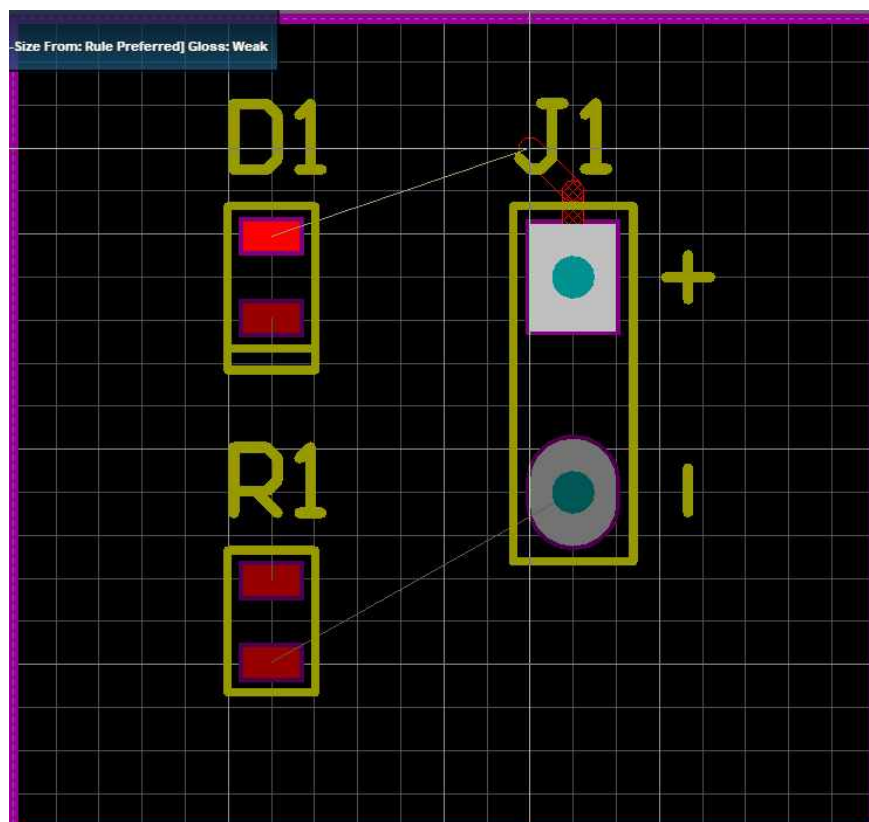
RoutingVias로 가서 비아 지름, 비아 홀 크기를 모두 10으로 변경한다.



이제 배선을 할 것이다. 아래의 아이콘을 클릭한다.



아래와 같이 선을 연결중인 상태에서 Tap키를 누른다.



속성

Use expanded Width
 모든 레이어 적용 ☒
 (Impedance: 0)

레이어 Top Layer Template

배선 폭 조건

Rule 규칙 편집... Trace Width is currently constrained by the rule 'Width' to a minimum of 10mil and a maximum of 100mil.

비아 모양 조건

Via style is currently constrained by the rule 'RoutingVias'. The minimum and maximum hole sizes are 10mil and 10mil. The minimum and maximum diameters are 10mil and 10mil.

핀 바꾸기

☒ 활성 Project needs to be saved

선택 보조 침착 길이 20mil

배선 중격 해결

- ☒ 장야를 무시 (G)
- ☒ 장야를 늘리기 (D)
- ☒ 장야를 줄이기 (W)
- ☒ 자물 장야들에서 접촉 (S)
- ☒ 장야를 늘기 및 감싸기 (H)
- ☒ 현재 레이어에서 자동 배선 (A)
- ☒ 모든 레이어에서 자동 배선 (B)

원격 모드 HugInPush Obstacles

충돌 배선 옵션

- ☐ 90/45도 제한
- ☐ 마우스 클릭 따옴
- ☒ 자동 배선 종료
- ☒ 종로 자동 제거
- ☒ Remove Net Antennas
- ☒ 비아 밀기 허용 (A)
- ☒ Display Clearance Boundaries
- ☐ Reduce Clearance Display Area

배선 마감질(Gloss)

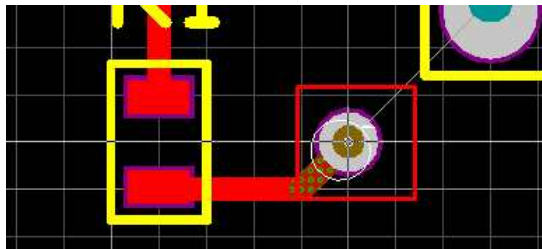
- ☐ 고가 (O)
- ☒ 약탈 (M)
- ☐ 강탈 (S)

Interactive Routing Width Sources

- ☒ 기존 배선을 잡을 때 동일한 선 폭 모드 User Choice
- 비아 크기 모드 Rule Preferred

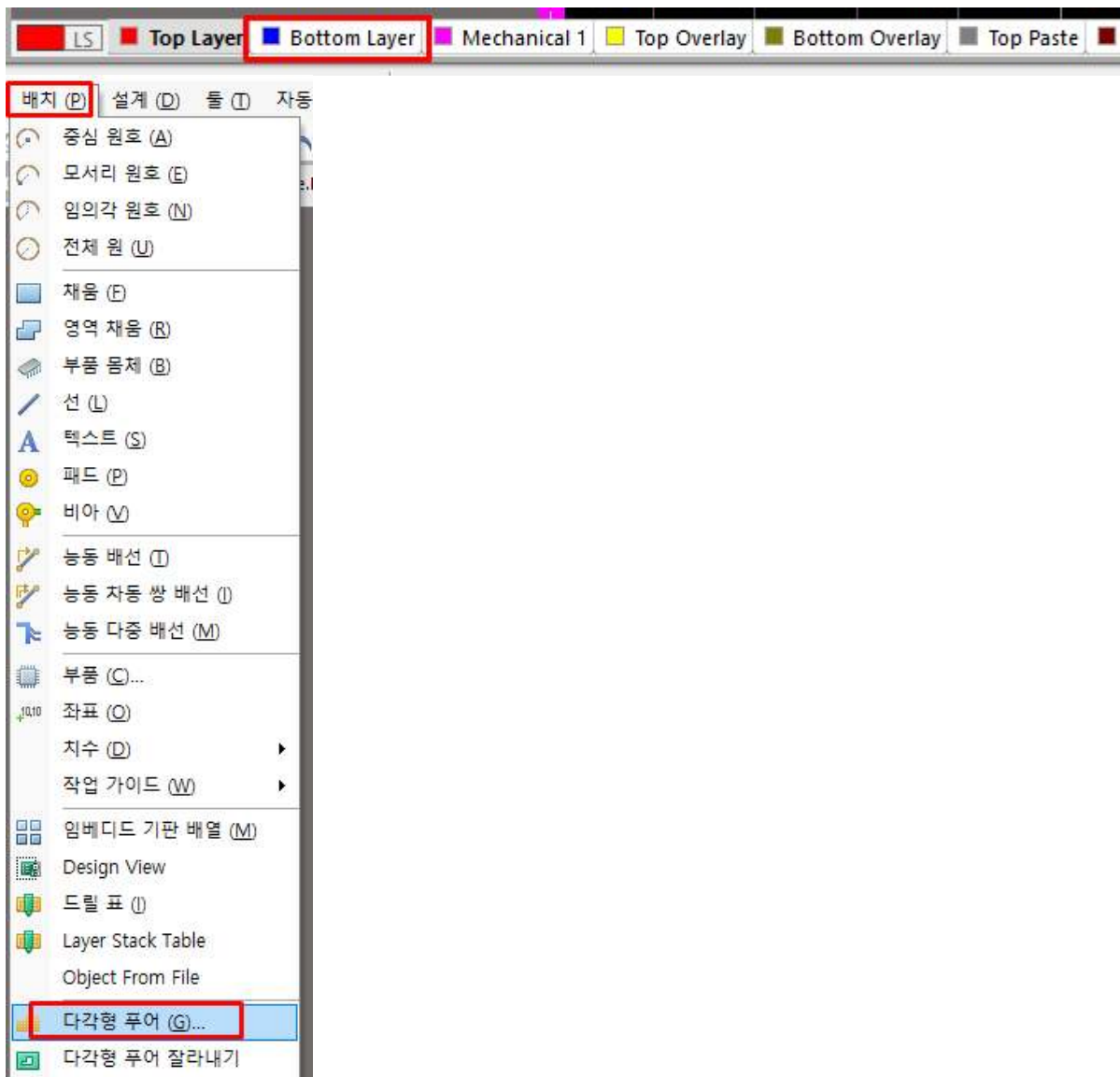
줄거렸기 자주 사용하는 활동 배선 폭 (E)

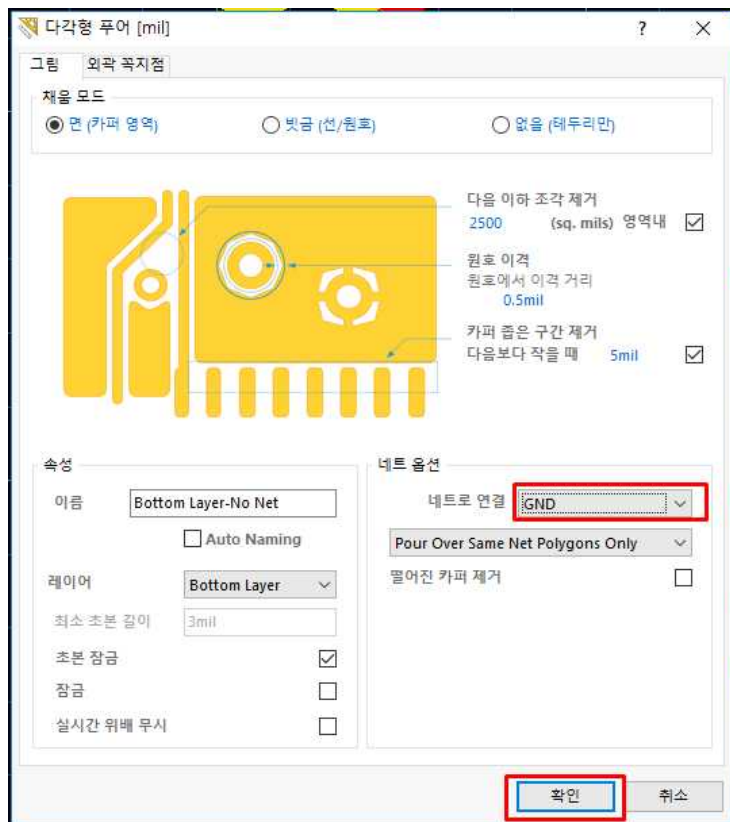
메뉴 (M) 확인 취소



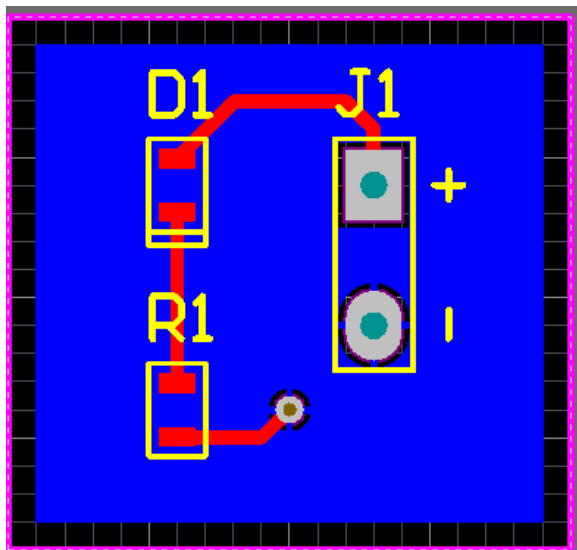


Bottom Layer로 이동한다.





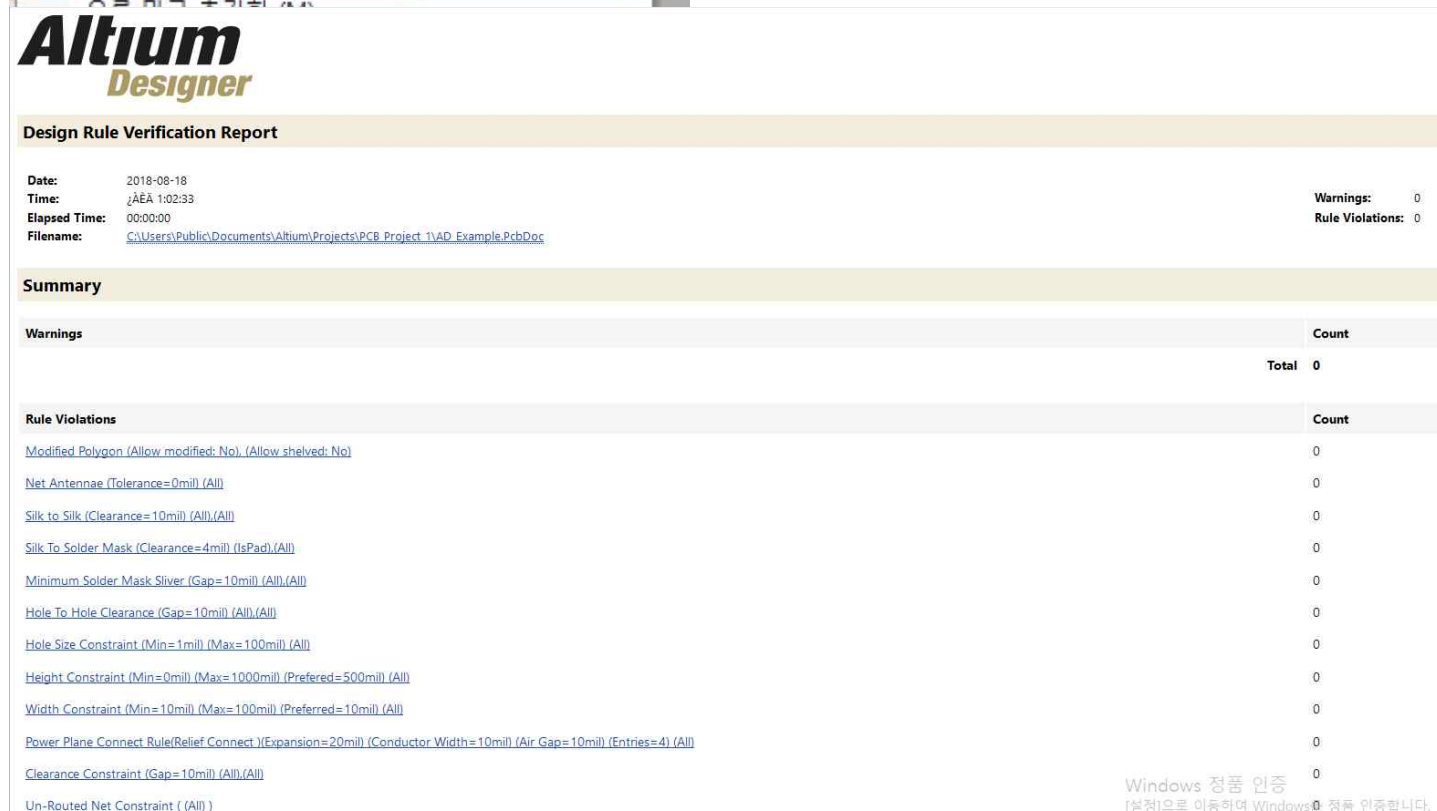
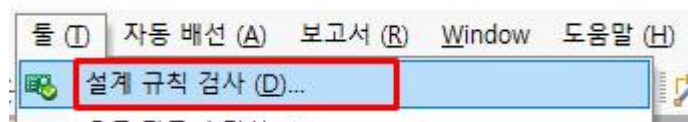
아래와 같이 그려준다.



글씨도 아래와 같이 추가한다.(배치 -> 텍스트)



설계가 정상적으로 되었는지, 체크를 한다. (DRC)

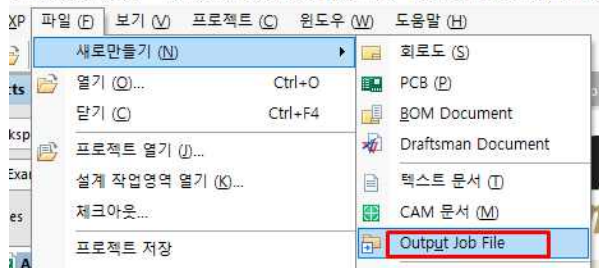


위와 같이 Warnings와 Rule Violations가 없다면, 일단 에러는 없는 것이다.

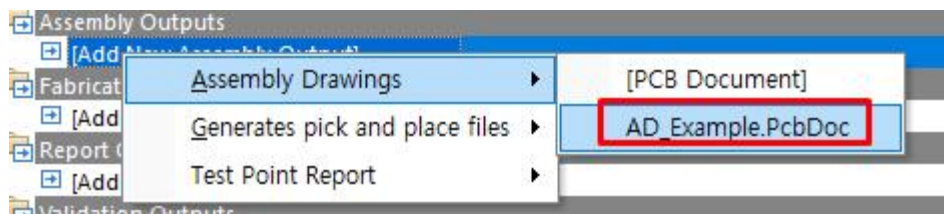
이제 아웃풋 파일을 만들어보자

파일 -> 새로만들기 -> Output Job File

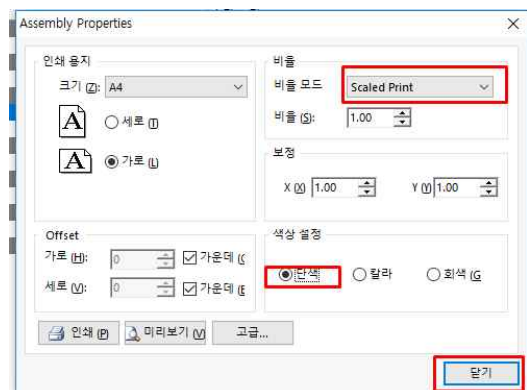
um Designer (16.1) - Workgroup [Workspace1.DsnWrk] - file:///C:/Users/Public/Doc



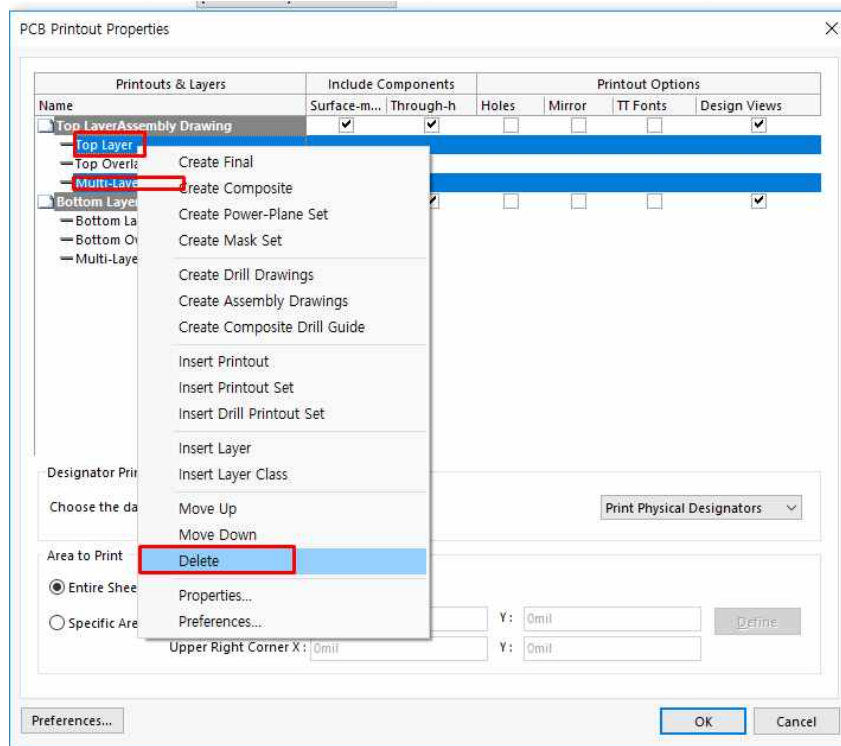
Assembly Drawings를 추가한다.



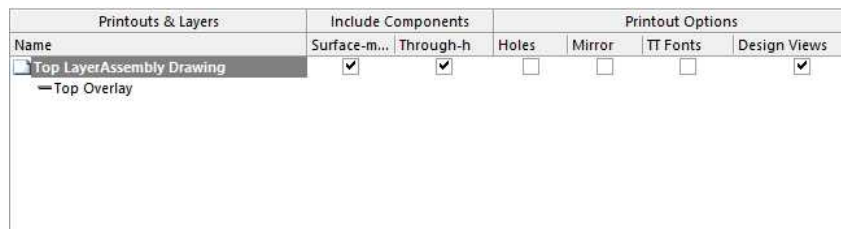
우클릭 후 페이지 설정에서 아래와 같이 설정한다.



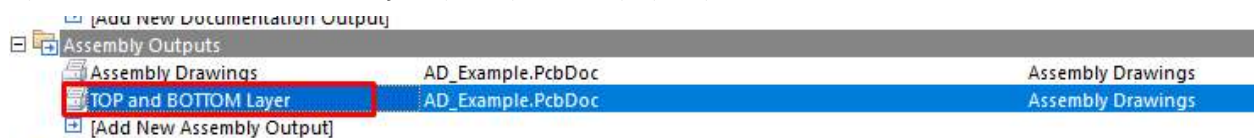
우클릭 후 환경설정에서 Top Layer와 Multi-Layer를 삭제한다.



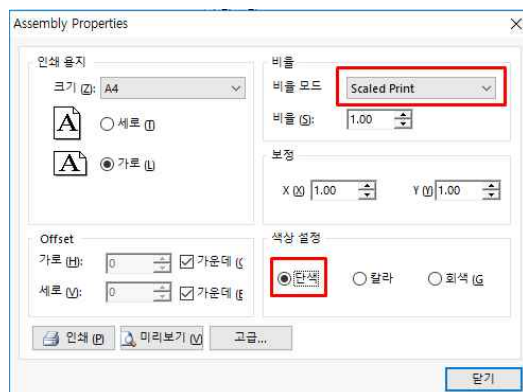
Bottom Layout은 전부 삭제한다.



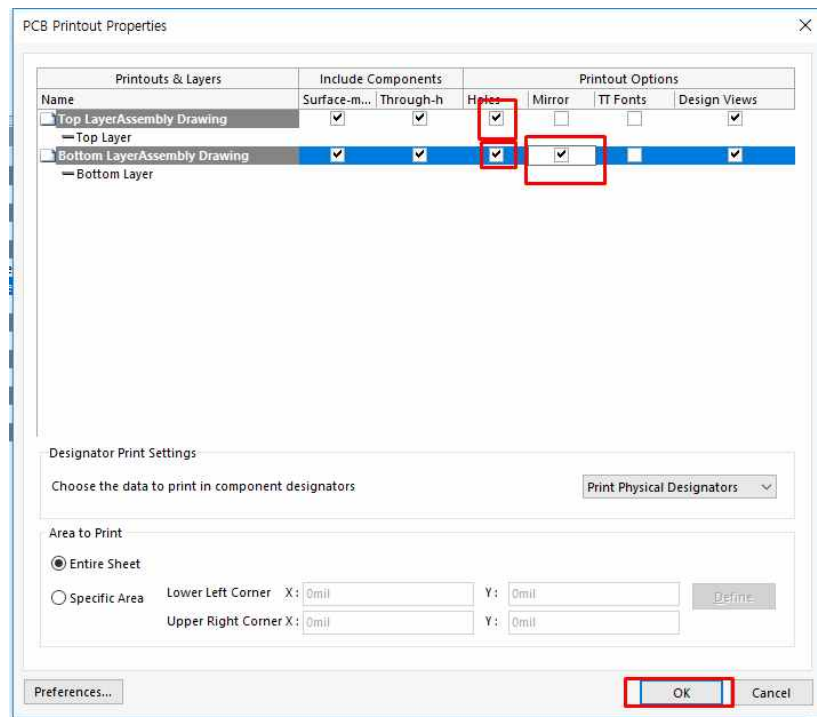
이번엔 TOP and BOTTOM Layer라는 이름으로 추가한다.



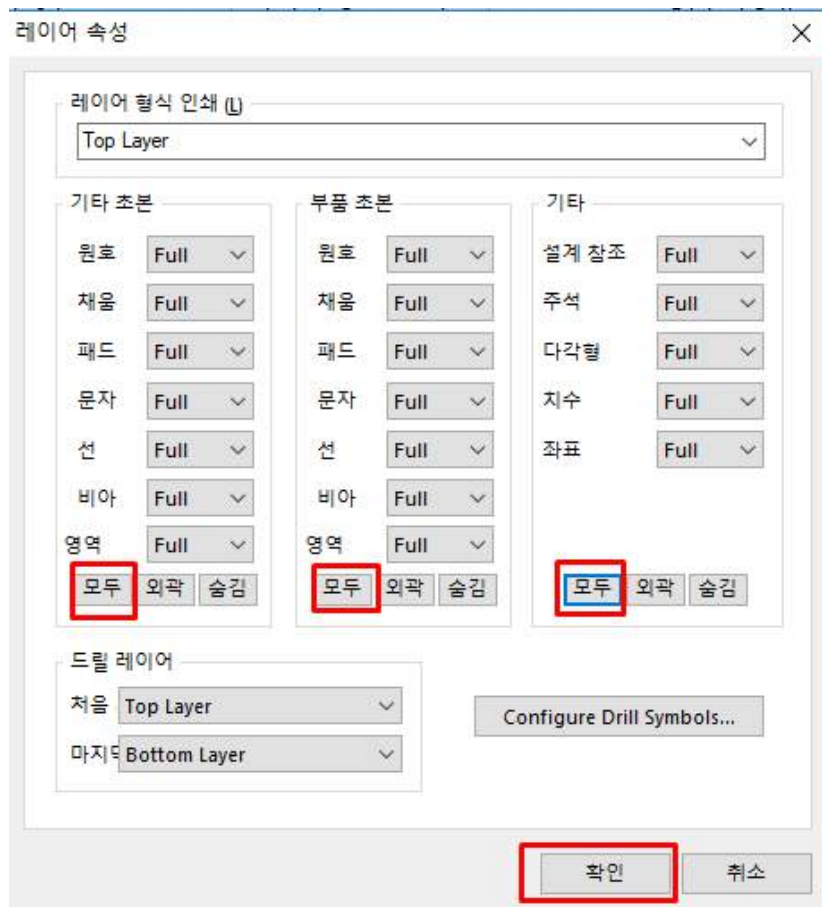
페이지설정도 위와 똑같이 해준다.



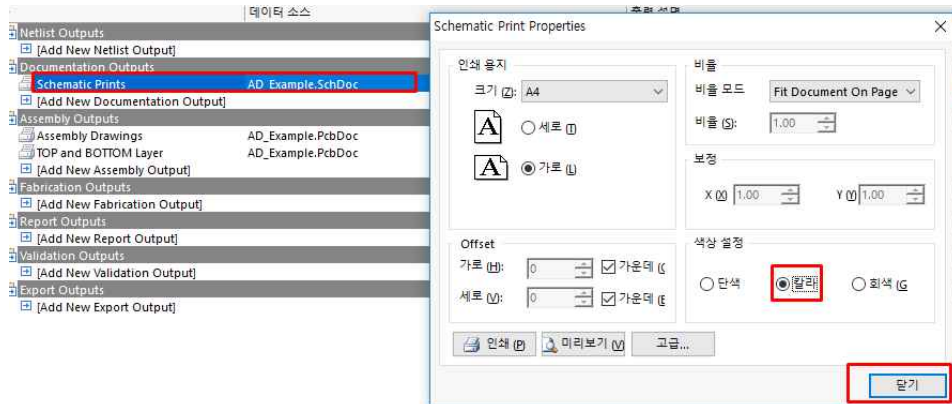
환경설정은 아래와 같이.



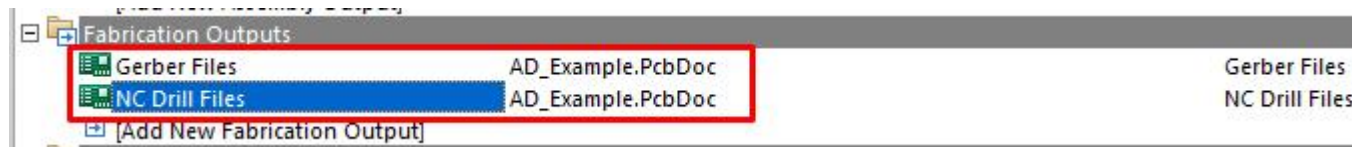
Top Layer와 Bottom Layer를 더블클릭하여 모두를 다 눌러준다.



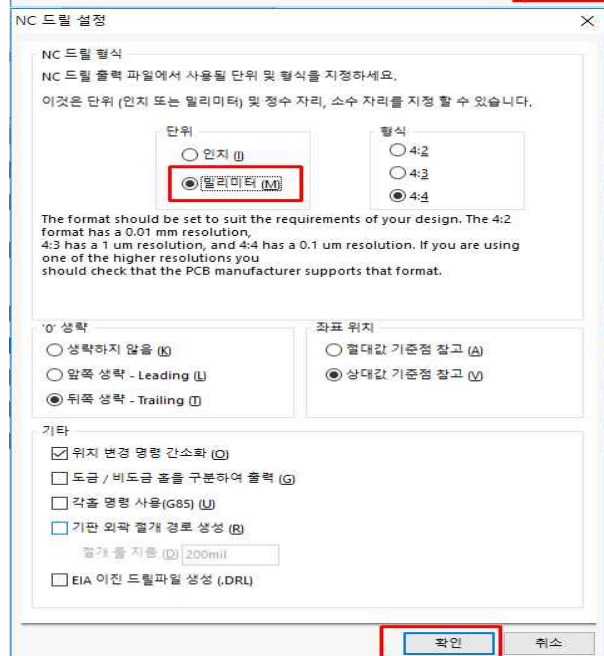
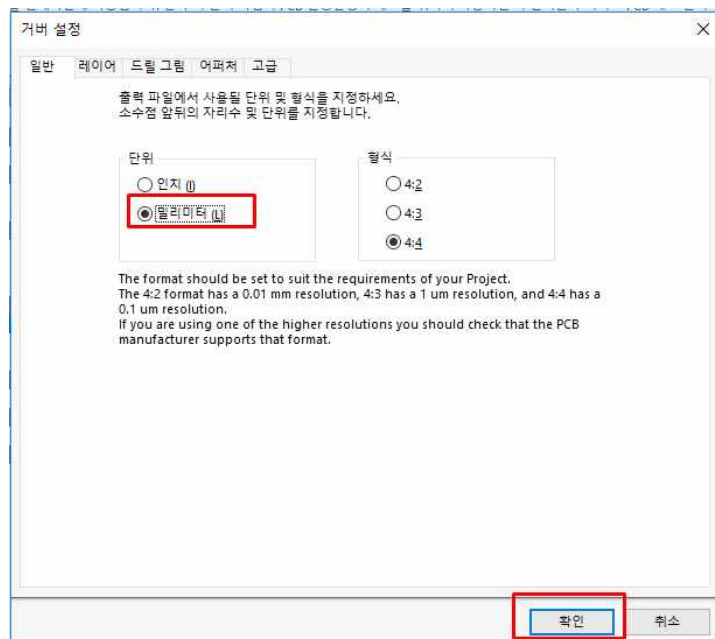
Schematic Prints를 생성하여 페이지 설정에서 칼라로 바꾼다.



Fabrication Outputs에서 거버 파일과 드릴 파일을 추가한다.



설정을 아래와 같이 해준다.



BOM 파일도 추가한다.

