Proteus 프로젝트 만들기

- 1. New Project
- 2. 프로젝트 이름 및 경로 지정
- 3. "Create a schematic from the selected template DEFAULT" 선택
- 4. "Do not create a PCB layout" 선택
- 5. "Create Firmware Project" 선택
 - A. Family: AVR
 - B. Controller: ATmega128
- 6. Finish

7. "Component Mode – P" 선서대로 click하고, LED GREEN 및 RES 선택

ILED_blinking - Proteus 8 Professional - Sc File Edit View Tool Design Graph Image: State	demais Capture Debug Library Template System Help 요 프 말 🖉 @ 冊 季 속 속 속 속 . " ୨ 연 싫 哈 改 물 물 물 물 은 송 양 方 🔀 용 것 금 및 용 요		- D X
Schematic Capture X Source	Code X		
	# Pick Devices	?	×
	Keywards: Ind green Showing local results: 5 Match whole works? Derice LED-BR/SRAPH-GNN USRLy LED-BR/SRAPH-GNN Description Since works? Show only parts with models? LED-BR/SRAPH-GNN USRLy Acrimeted BI-Colour LED model (Green/Amber) with Self-flashing Acrimeted BI-Colour LED model (Green/Amber) acrimeted BI-Colour LED model (Green/Amber) acrimeted BI-Colour LED model (Green/Amber) acrimeted BI-Colo	Preview Schematic Model (LEDA)	
	Keywards: Showing local results: 11766 Keywards: Device Match whole words? Person Match whole words? Person Show only past with models? Public Person Show only past with models? Public Person Show only past with models? Public Person Show only past with models? Show only past with models? Public Person Show only past with models? Show only past with models wi		
C r ↔	(All Sub-schepotre) Bargraph Displays Dot Mark Olsplays LEDs	PCB Preview	

8. LED GREEN 및 RES 추가, ATmega128와 연결



Microchip Studio 프로젝트 만들기

- 1. New Project
- 2. "C/C++ AVR XC8 C Application Project" 선택
- 3. 프로젝트 이름 및 경로 지정
- 4. ATmega128 선택

Abriced Parts				
EPROM (byte Device Info:	Supported Parts	 Show devices: 	All	Device Family:
	bytes)EEPROM (byte	(Kbytes)Data Memory (App./Boot Mem	Name
4096 ATmega128	4096	4096	128	ATmega128
1096 Speed: N/A	4096	8192	128	ATmega1280
1096 V/cc: N/A	4096	8192	128	ATmega1281
1096 Camilus ATmaga	4096	16384	128	ATmega1284
1096 Pathiy. Athega	4096	16384	128	ATmega1284P
1096 Device page for ATmega128	4096	16384	128	ATmega1284RFR2
1096 Datasheet	4096	4096	128	ATmega128A
1096	4096	16384	128	ATmega128RFA1
1096 Supported Tools	4096	16384	128	ATmega128RFR2
512 Atmal ICE	512	1024	16	ATmega16
256	256	2048	16	ATmega1608
256 AVR Dragon	256	2048	16	ATmega1609
312 AVR ONE!	512	1024	16	ATmega162
512	512	1024	16	ATmega164A
512	512	1024	16	ATmega164P
312 <u>X EDBG MSD</u>	512	1024	16	ATmega164PA
512 PAVRISP mkll	512	1024	16	ATmega165A
	512	1024	16	ATmega165P
12	512	1024	16	ATmena165DA
	512	1024	16	ATmena165DA

5. 코드 작성

```
#include <xc.h> // compiler
#include <avr/io.h> // 입출력 포트 (DDRA 및 PORTA 사용)
#include <util/delay.h> // _delay_ms() 함수 사용
int main(void) {
    DDRA = 0xff; // PORTA 출력 모드 설정
    while (1) {
        PORTA = 0x07; // PA0, PA1, PA2 를 1로 설정 (LED on)
        _delay_ms(2000); // delay 2s
        PORTA = 0x00; // PA0, PA1, PA2 를 0로 설정 (LED off)
        _delay_ms(2000); // delay 2s
    }
}
```

- 6. 기본의 "Debug"을 "Release"로 바꿈
- 7. Build Build Solution
- 8. Build Solution이 성공되면 ATmega128 프로그래밍을 위한 ".hex" 파일이 생성됨

o led_blinking - Microchip Studio					
File Edit View VAssistX ASF Project Build Debug Tools Window	Help				
◎ - ◎ ② - 创 1 - 1 🖬 🗳 🐰 日 白 ラ - ୯ - 🔠 🔍 🕨 🛤	Release	- B	- 🗊 🗲 😂 🔛 🚅 💋	🖂 - 🛃 - 프 프 프 프 프 📕 1	
MI (b) = → II ▶ ↔ ‡ ? ‡ ┡ T Hex 🔏 🏹 - 🐙 🗐 🕮	🗟 🖫 🚽 🏙 🏯 🛛 🚛 🗰 ATmega128 🁔	None on 🖕			
main.c 😐 🗙 ASE Wizard	<u> </u>	i .			
main.c • D:#Lectures#embedded_programming#example	s#lec_1#led_blinking#led_blinkingi main.c				
8/*					
* Main.C * * Created- 3/11/2025 12-05-28 DM	🔁 Release 🛛 🗙 🕂			- 🗆 ×	
* Author: USER */		···· led_blinking > led_blinking >	Release Rele	ase 검색 으	
#include <xc.h> // compiler #include <xvr io.h=""> // 입물력 포트 (DDRA 및 PORTA 사용) #include <xvr io.h=""> // dalaw er() 한순 사용</xvr></xvr></xc.h>	⊕ 새로 만들기 - 🔏 🕡 👔	🗐 🖄 向 🛝 정렬 -	≣ 보기 ~ …	🖪 세부 정보	
winclude (delayed) in 77 _delay_in(7 B1 -10	🔁 걸러리 이름	^ 수정한 날짜	유형 크	7	
Bint main(void) { DDRA - 0xff; // PORTA 출력 모드 설정	> 📥 dat - 개인 📄 led_blinking.e	ep 2025-03-11 9	.후 12:11 EEP 파일	1KB	
while (1) { PORTA = 9x97: // PA9, PA1, PA2특 1로 설정 (LED on)	led_blinking.e	ff2025-03-11 ♀	·후 12:11 ELF 파일	10KB	
_delay_ms(2000); // delay 2s	led_blinking.h	ex 2025-03-11 오	후 12:11 HEX 파일	1KB	
	LESEC	s 2025-03-11 Q	후 12:11 LSS 파일	8KB	

Proteus 시뮬레이션

- 1. ATmega128에 double-click
- 2. 이전 단계에 생성된 ".hex" 파일을 선택함

000									
Edit Component				? ×					
Part Reference:	UI	Hidden	:0	ОК					
Part <u>V</u> alue:	ATMEGA128	Hidden	:0	Help					
31 Element:		New		Data					
PCB Package	QFP80P1600×1600×120-6	V AA Hide Al		lidden Pins					
Program File:	1	Hide Al	~ E	dit Firmware					
CKOPT (Oscillator Options)	Select File Name								×
BOOTRST (Select Reset Vector)	*** 0(*//)	Deleses			a a in m-				
WDTON (Watchdog timer always on)	90 HAID-	nelease	^		· • • • • • • • • • • • • • • • • • • •				
CKSEL Fuses:	A 94	•			수정한 날짜	유형	크기		
Boot Loader Size:	÷ 📅	led blinking elf			2025-03-11 오후 12:11	ELF 파일		10KB	
SUT Fuses:		ieu oliiking.nex			2023-03-11 == + 12.11			IND	
Advanced Properties:		2							
Clock Frequency ~	바탕 화면								
Other Properties:									
	라이브러리								
	H PC								
Exclude from Simulation	3								
Exclude from PLB Layout	1 E 8 7								
t									
								3	
	파일	0[름(N): lec		,hex			~	열기(0)	
	파일 형식(T): HEX LIBBOE COFE FLE or OBJ Files				, L	최소	-		
								71	

3. 시뮬레이션 시작



* <u>NOTE</u>: 사용하는 Proteus의 버전은 최신 버전이 아니므로, 처음에 시뮬레이션을 시작했을 때 제 대로 돌리지 않은 현상이 있음. 위 2~3 단계를 한 번 더 반복하시면 됨.