

# Adafruit GFX Matrix 16 X 48 LED 전광판

저 전력 / 고 휘도 LED 전광판 ( Arduino Based )

Main H/W Develop

심성훈

Main S/W Develop

권보승

Interface Management

강병근

Controller Management

이호준

Power H/W Develop

최윤석

# 프로젝트 목표

## [ 수행 완료 ]

- [1차] Adafruit (제조사) 의 16 x 16 Flexible 16x16 NeoPixel LED Matrix 를 이용하여 전광판 완성
- [2차] 제조사가 기본 제공하는 8 x 8 GFX Library 를 이용하여 전광판에 문자 출력 및 Flow Effect 출력

## [ 미구현 ]

- [3차] 8 x 8 환경의 GFX Library 를 16 x 16 사양으로 재구현 하고, 256 pixel 에 맞는 Full-Size 한글 Font 출력

# 하드웨어 소개

## [ 메인 하드웨어 ]

- WS2812 내장 컨트롤러가 탑재 된 Flexible LED Matrix \* 3

-> 접을 순 없지만 구부릴 수 있고 얇으며 추가적인 컨트롤러 없이 연결된 다른 LED Matrix 에 신호를 전달할 수 있다.

- Arduino Mega 2560 Micro-Controller ( Based On Atmega2560 )

-> 초기 Atmega 128 (Arduino Uno) 사양이었으나, Uno 보드가 마이크로 컨트롤러에 0x1FF 를 초과하는 신호를 전달하는 경우 2MB 의 메모리 상한 문제로 강제 인터럽트가 발생하는 문제가 있어 고 사양 제품군인 메가 2560 보드로 변경

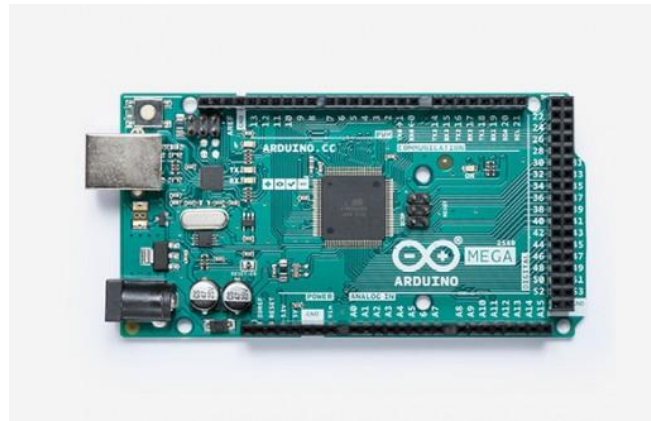
## [ 전기회로/전원부 하드웨어 ]

- AC 220V to DC 5V~12V Voltage Converter / Power Supply

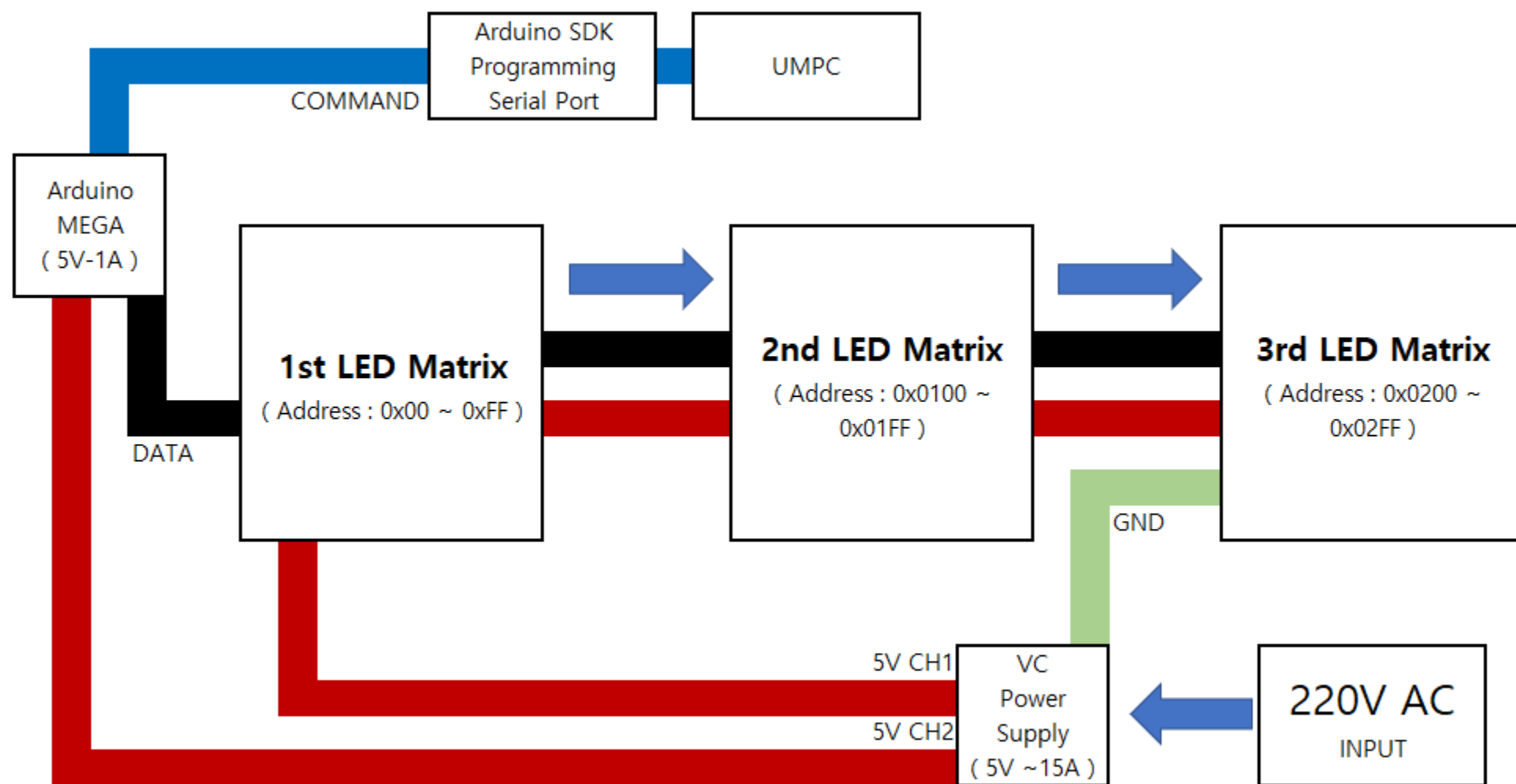
-> 초기 USB Serial 로 부터 전원을 공급받아 5V 2.1A 출력을 할 수 있는 초 소형 전압 컨버터를 장착했으나 접지 불량으로 예상되는 불분명한 사유로 인해 파손되어 고가형 변압기로 변경

## [ 기타 제작재료 ]

- 포맥스 보드
- 거치대



# 제품 동작 방식



# S/W 구현

- <Adafruit\_GFX> / <Adafruit\_NeoPixel> Function Library 사용
- 이 라이브러리는 8 x 8 Adafruit LED Matrix 를 대상으로 GitHub 에서 Arduino 개발자들이 제공하는 것으로 4배의 해상력을 가진 16 x 16 Matrix 에서는 동일하게 적용이 불가능하다

매트릭스 동작을 위한 사전 정의(Define) / 초기화 세팅 함수	출력의 시작 좌표, 매트릭스의 LED 수, 처리 순서, 출력되는 포트, 속도, 출력할 색상에 대한 선 정의가 가능하다. 글자를 출력할때와 색상 출력 때의 선언문이 달라진다.
<b>Adafruit_NeoMatrix(X Array, Y Array, I/O, Function A + .... + Fucntion Z)</b>  <b>Adafruit_NeoMatrix(48, 16, PIN, NEO_MATRIX_TOP + NEO_MATRIX_LEFT + NEO_MATRIX_COLUMNS + NEO_MATRIX_ZIGZAG + NEO_MATRIX_PROGRESSIVE + NEO_GRB + NEO_KHZ800 )</b>	

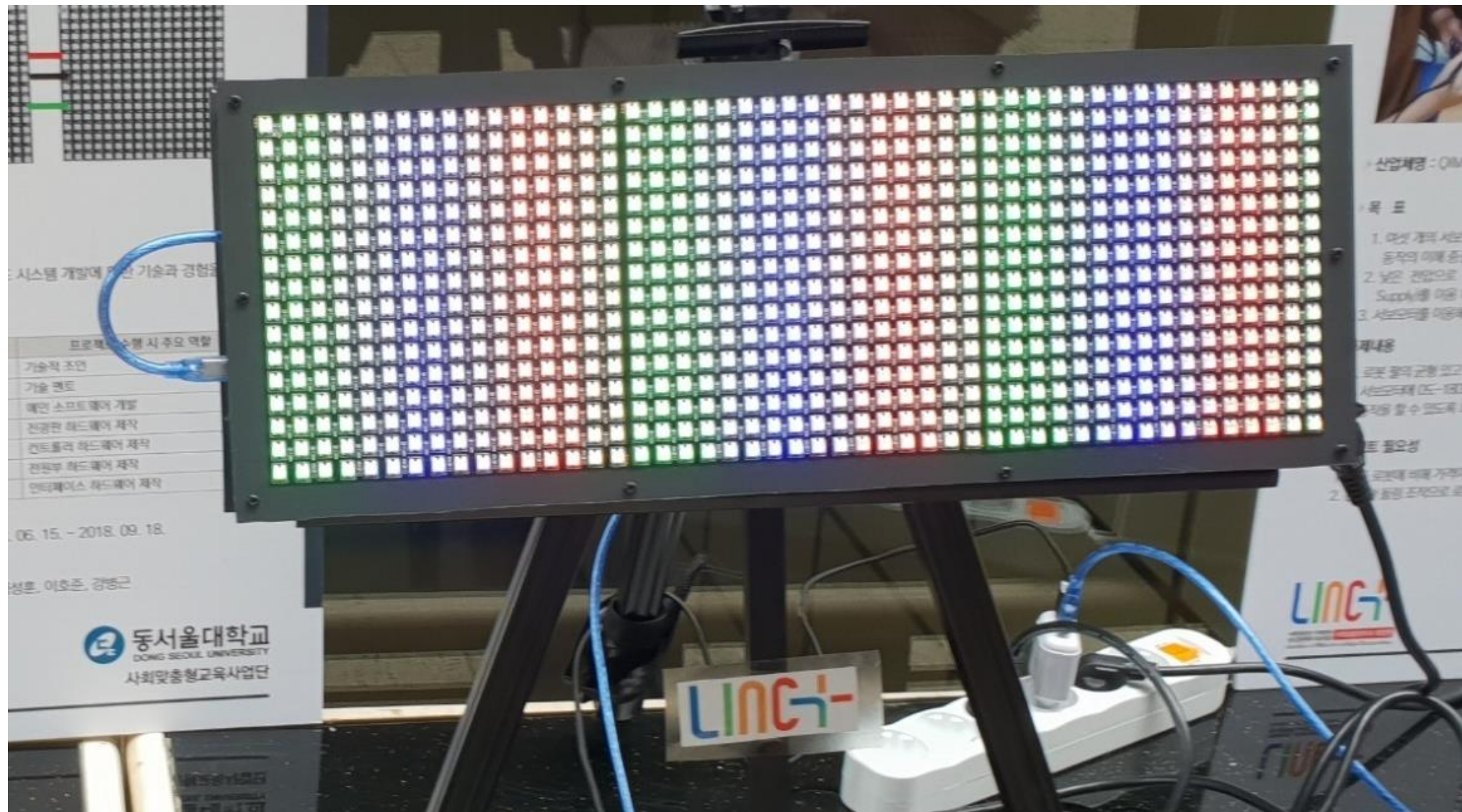
매트릭스 출력 선언 함수	<b>Matrix.begin()</b>	지정된 우선순위에 따라 출력
매트릭스 밝기 선언 함수	<b>Matrix.setBrightness(level)</b>	밝기 단계 선언
출력 좌표 선언 함수	<b>Matrix.setCursor((matrix.width), y)</b>	출력이 시작될 위치 설정
LED Color 선언 함수	<b>Matrix.Color(R_level,G_level,B_level)</b>	해당하는 pixel 의 RGB 값 설정
LED ON 함수	<b>Matrix.drawPixel(0, Address, color)</b>	LED가 ON될 포인트 및 색 반환

이하의 LED 출력을 위한 잡다한 설정은 다중 for 문을 이용하여 각 Pixel 마다 갖는 color 값이 서로 달라지게끔 할 수 있다.  
 이 경우 matrix.drawPixel(0, adress, 255 - RGB\*3/((i+j)%255)) 등으로 시간단위로 달라지는 값을 color 로 줘야 한다

- 8 x 8 사이즈에 해당하는 기초적인 문자 출력은 주어지는 라이브러리를 통해 행할 수 있으나 LED Flow 출력은 다음과 같은 함수들을 이용하여 직접 작업해야 되었다.



# 완성된 실제 제품



# LED Matrix 에서 글자 출력



```
void textmatrix2() {  
  matrix.fillScreen(0);  
  matrix.setCursor(x, 4);  
  matrix.print(F("—DONG SEOUL UNIVERSITY—"));  
  
  if(--x < -145  
  ) {  
    x = matrix.width();  
    phase=2;  
    pass++;  
  }  
  matrix.show();  
  delay(30);  
  
  if(pass >= 7) pass = 0;  
  matrix.setTextColor(colors[pass]);  
}
```

초등학생도 할 수 있다는 Hello World 뽑아 놓고 만만하게 생각했으나 여기서 문제 발생 !

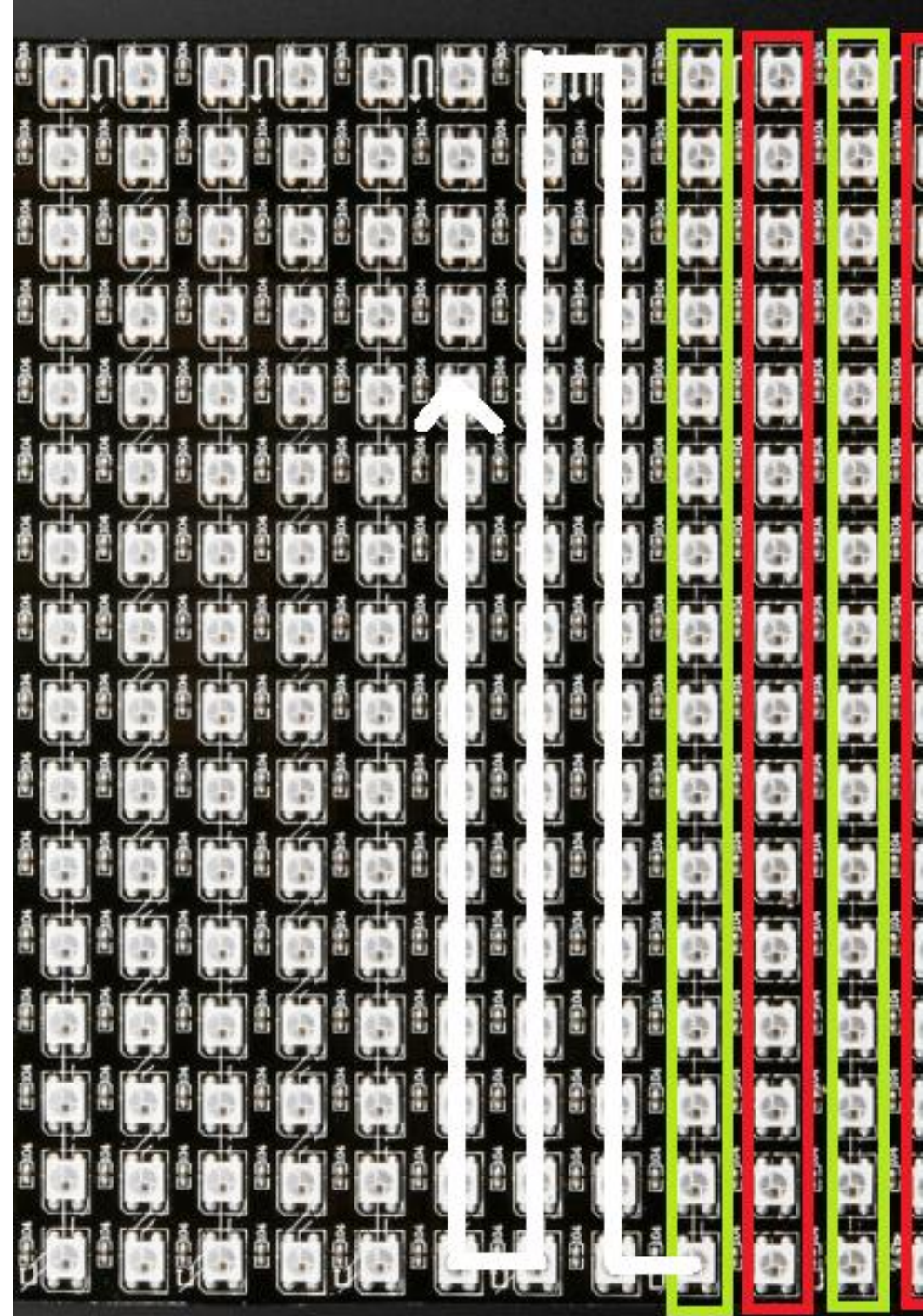
GFX Library 에서 기본 제공하는 영어 출력 함수와, **NEO\_MATRIX\_ZIGZAG** 를 통해 영어로된 문자열은 print 값을 주고 각 문자가 Flow (16개의 Pixel 배열이 0.3초마다 시프트 연산을 통해 다음 배열 포인터로 넘어간다) 될 시간을 정의해 두는 것으로 큰 문제없이 출력이 가능했다 !

근데 ... Matrix 동작 특성이 있어 한글 출력이 안된다 !!!!



# Matrix 동작 특성

Array[0] = 시작점의 주소 0x00, 마지막 주소 0x0F  
Array[1] = 시작점 주소 0x1F, 마지막 주소 0x10 = 프로그래밍적 어려움 발생





# 최종 완성된 제품의 문제점

- 이 결과물은 9월 진행된 1차 캡스톤 대회 출품 시의 사양으로 최종 목적이었던 '한글 출력' 부분이 완성되지 못했다.
- 이는 두가지 문제를 가지고 있는데 초기 GFX Library 로 설계 되어있는 8 x 8 Array 와 16 x 16 Array 의 LED 출력 순서 차이와, 그로인한 Font 설계의 어려움에서 기인한다.
- AVR\_GCC 컴파일러는 한글 String 을 변수나 배열 데이터로 변환할 수 없기에 출력할 한글 자체를 256 pixel Array 로 선 처리 해야 된다.

