
CILAB IMU SENSOR

Shop Upload

21th, Jan, 2018



1 IWD-IMU V1

Abstract 제품 설명

CILAB 에서 개발한 IWD-IMU V1 은 ROV, UAV 또는 유사 이동 로봇에 적용이 가능한 멀티 융합 센서로 설계 및 제작 되었습니다. IWD-IMU V1 센서는 일반적인 관성 측정 센서, 압력 센서, 온도 센서로 구성 됩니다.

Key Features:

- 방수 기능(수중에서 사용 가능)
- 융합 센서 데이터 출력
- 다중 센서를 하나의 장치에 융합
- 전원 관리 기능
- I2C 인터페이스
- 장기간 우수한 안정성
- On-chip 인터럽트 컨트롤러

2 특징 및 사양

IWD-IMU V1 센서는 3 축 가속도계(14bit), 3 축 자이로스코프(16bit), 3 축 지자기 센서, 압력 센서(24bit), 온도 센서(24bit)를 통하여 고정밀 방향 측정 값을 제공합니다.

일반 사양

- 다중 센서 데이터 융합
- 가속 범위 $\pm 16g$
- 자이로스코프 범위 $\pm 2000^{\circ}/s$
- 지자기 범위 $\pm 1300\mu T$ (x-, y 축); $\pm 2500\mu T$ (z 축)
- 온도 측정 범위 -20 to 80 $^{\circ}C$
- 압력 측정 범위 0 to 30 bar
- 크기 w 29 x d 46 x h 18
- 무게 62.5g

내부 사양

매개 변수	Symbol	Max	Min	Unit
전원 전압	VDD	3.6	2.4	V
공급 전류	IDD	12.6		mA

작동 온도	T	85	-20	°C
자이로스코프 범위		2000	-2000	°/s
가속도계 범위		16	-16	g
지자계 범위		2500	-2500	uT
압력 측정 범위		30		Bar
온도 측정 범위		85	-20	°C
부팅 시간	TB	400		ms

인터페이스

IWD-IMU V1 은 I2C 인터페이스를 통하여 모든 MCU(micro controller unit)와 통신할 수 있습니다.

매개변수	Max	Min	Unit
Clock Frequency	400		kHz
SCL Period	1.3	0.6	us
SDA Setup Time		0.1	
SDA Hold Time		0	
Setup Time for a repeated Start Condition		0.6	
Hold Time for a Start Condition		0.6	
Setup Time for a Stop Condition		0.6	
Time before a new Transmission can start		1.3	

3 *PRODUCT COMPONENTS*

IWD-IMU V1 의 구성은 아래 그림과 같다.



- IWD-IMU V1 센서
- 방수 컨넥터
- IWD-IMU V1 센서 케이스

4 CODE

IWD-IMU V1 센서 data 받는 방법 :

1. Arduino 설치

링크 : <https://www.arduino.cc/> 를 통하여 이동한 후 아래 그림 1 과 같이 프로그램 다운로드를 선택한다.

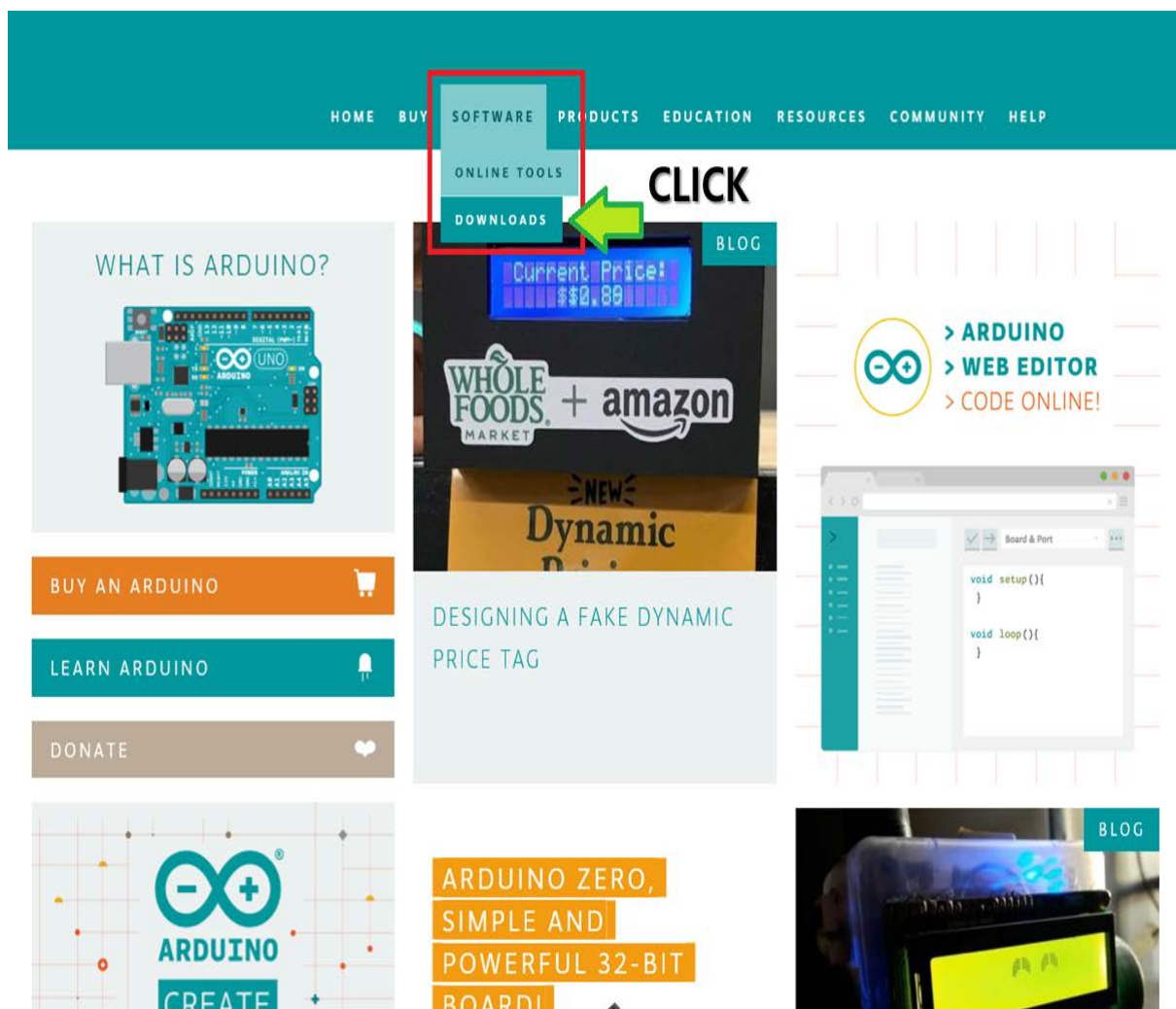



그림 1. Arduino 메인 화면

스크롤을 아래로 내리면 다음 그림 2 와 같은 화면이 나온다.

Download the Arduino IDE



ARDUINO 1.8.5
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.
This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer, for Windows XP and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10
[Get](#)

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

그림 2. Arduino 프로그램 다운

사용자의 컴퓨터 사양에 맞춰 프로그램을 다운로드하세요.

다운 받은 설치 프로그램을 실행하여 아래의 그림 3. 과같이 진행하세요

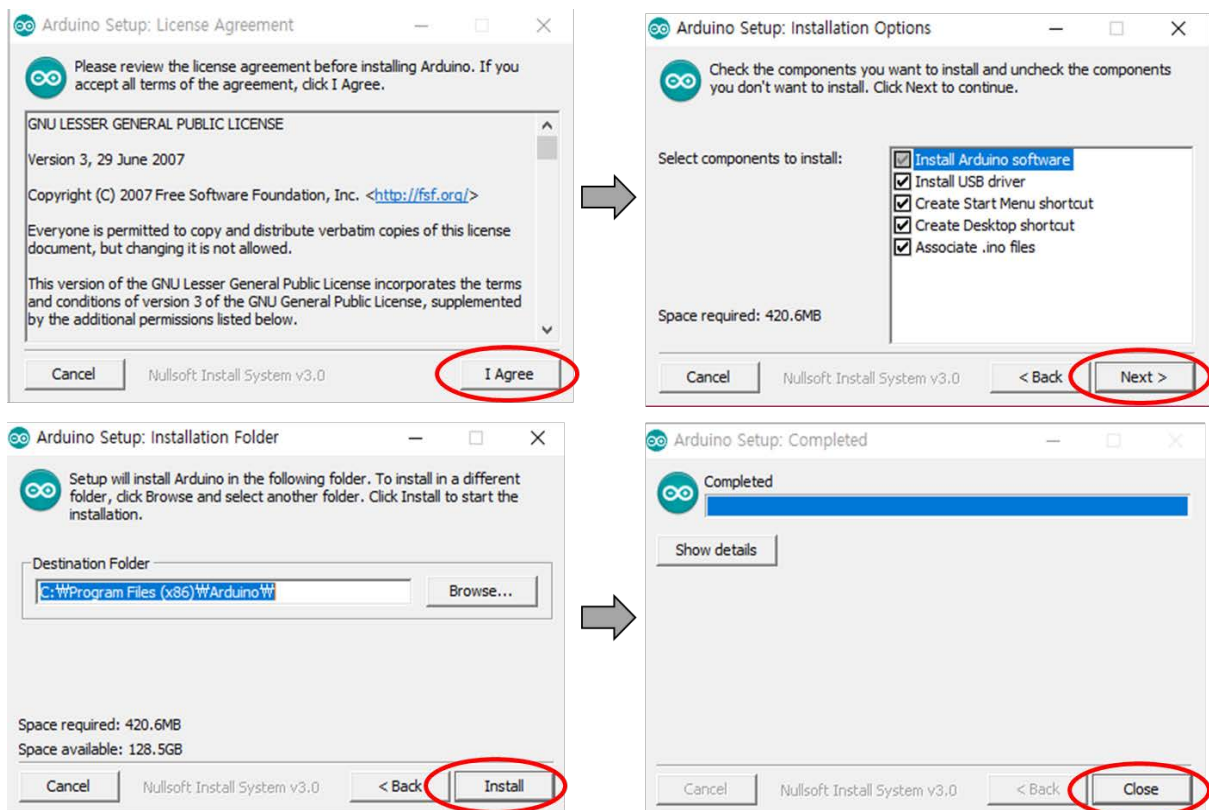


그림 3. Arduino 설치 진행 순서

Arduino 프로그램 실행 후 아래 그림 4 와 같이 진행

2. 예제 불러오기

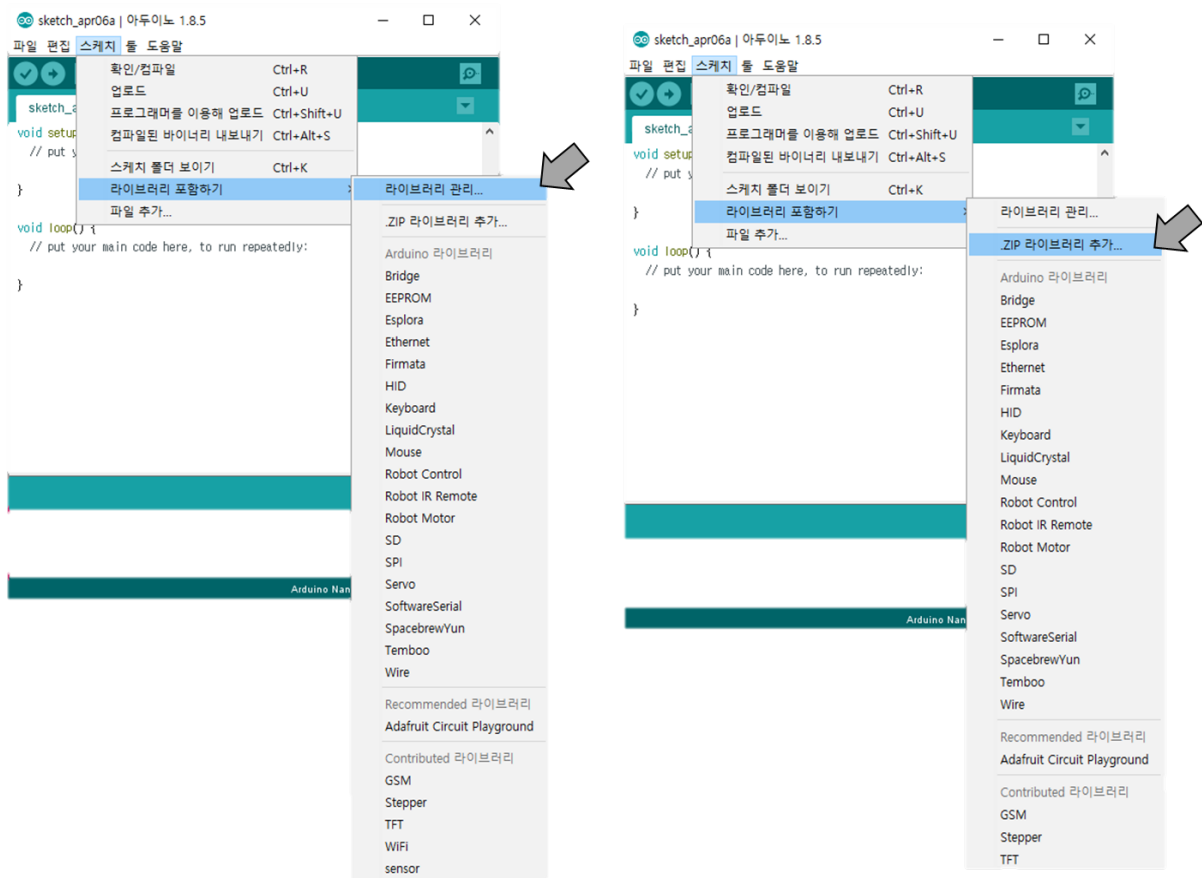


그림 4. Arduino 라이브러리관리

그림 4 와 같이 라이브러리 리스트 업데이트 진행 후, Zip 라이브러리 추가를 통하여 CILAB 에서 제공하는 예제 프로그램인 Sensor 압축 파일을 추가합니다. 추가한 예제 프로그램을 아래 그림 5 와 같이 불러옵니다.

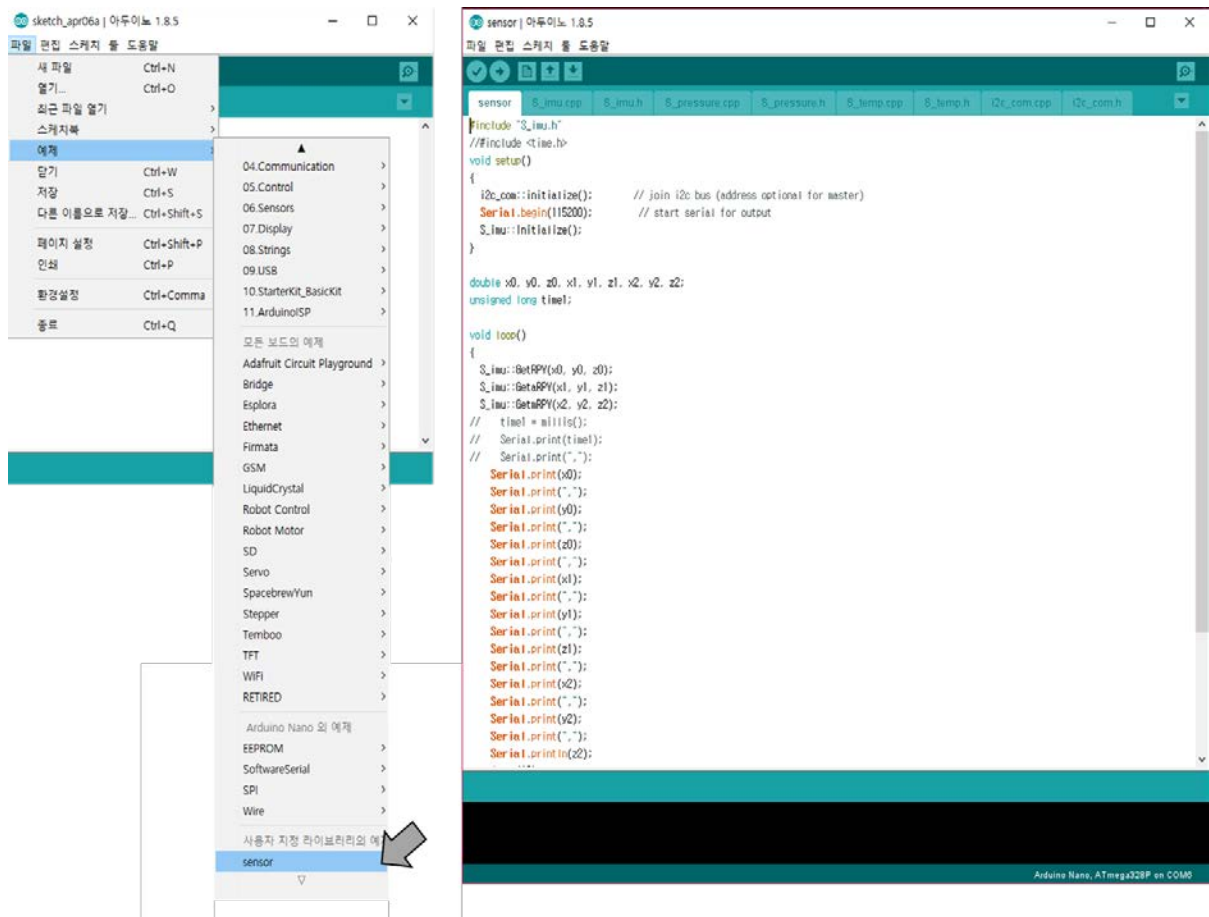


그림 5. 예제 프로그램 불러오기

3. Connect board

IWD-IMU V1 센서를 아래 그림과 같이 연결 후 PC 와 USB 포트로 연결

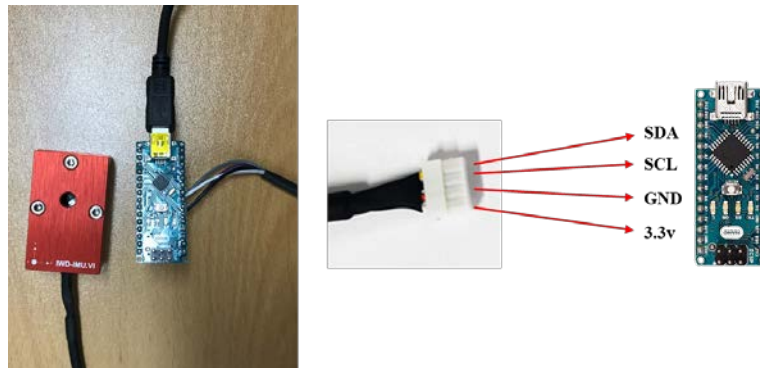


그림 6. IWD-IMU V1 센서 연결

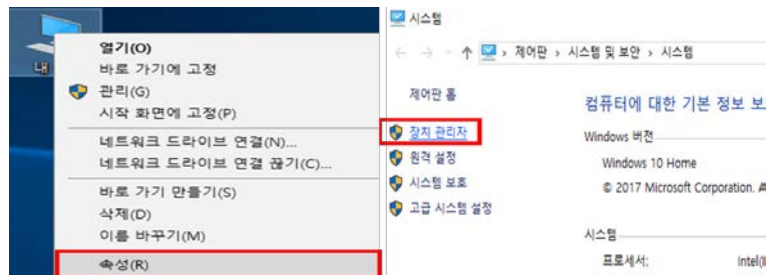


그림 7. 시스템 속성에서 장치 관리자

그림 7 과 같이 내 PC 에서 마우스 우 클릭 후 위 그림과 같이 장치 관리자를 클릭

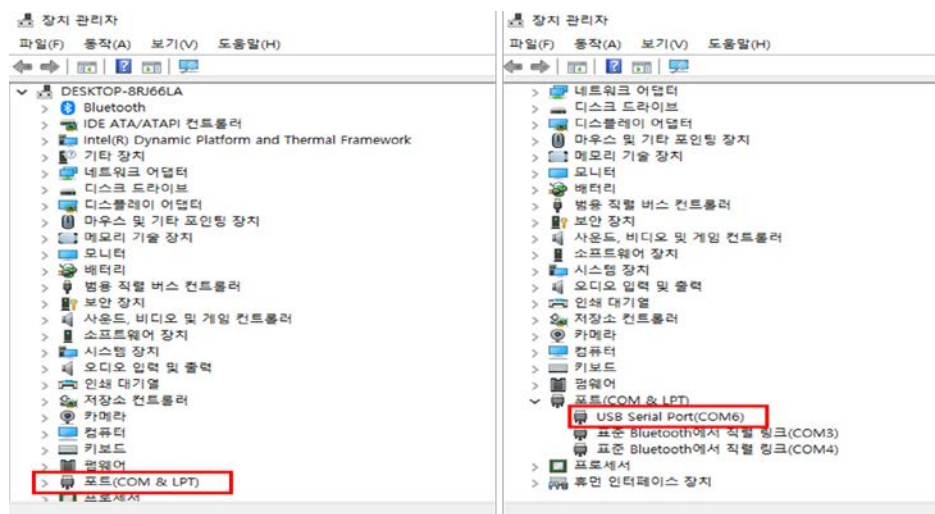


그림 8. Com port 확인

그림 8 과 같이 IWD-IMU V1 센서가 연결된 Com port 확인 (COM6)

4. Com port 및 baudrate 설정

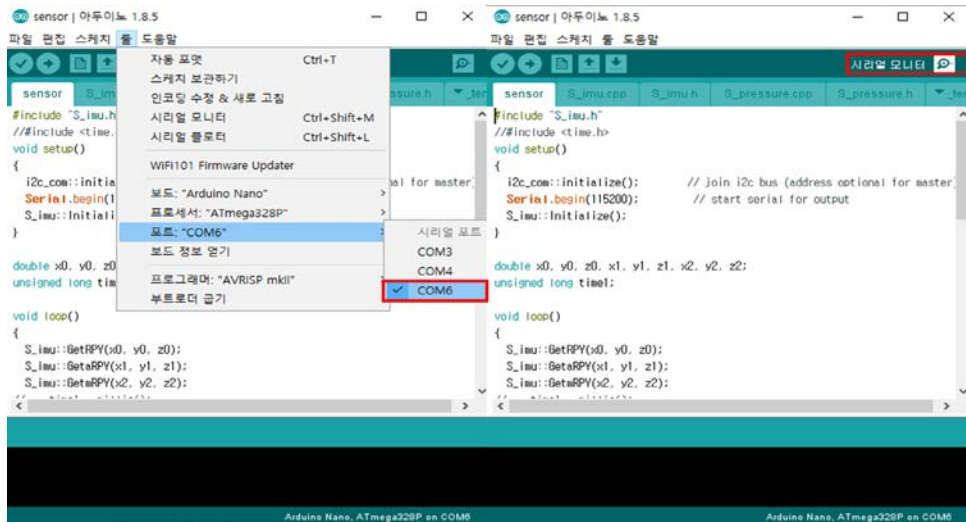


그림 9. Com port 설정

확인한 Com port(Com6)에 맞춰 그림 9 와 같이 포트 설정 후 상단에 시리얼 모니터 클릭

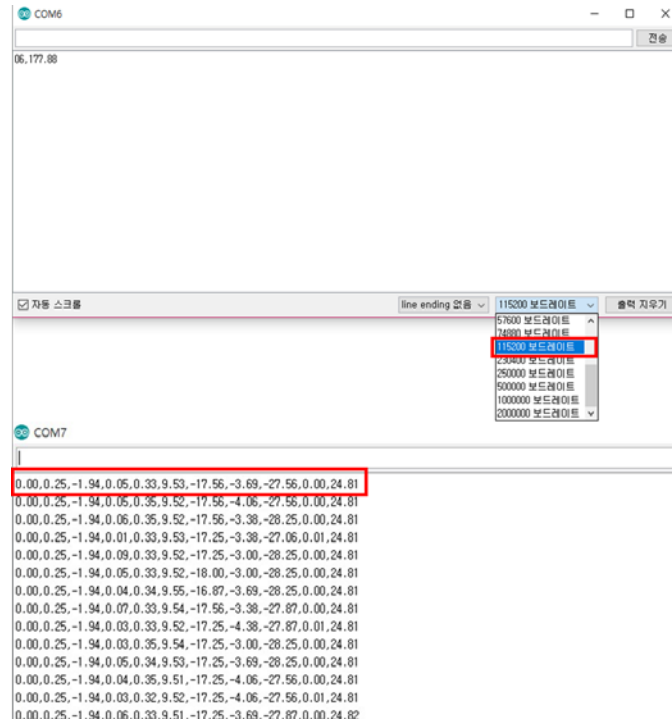


그림 10. Baudrate 설정 및 데이터 출력

그림 9 에서 시리얼 모니터를 클릭하면 그림 10 과 같이 실행 창이 생성되며, 실행 창 하단에 baudrate 설정(115200)한 후 센서 데이터 출력

5. 데이터 출력

0.00, 0.25, -1.94, 0.05, 0.33, 9.53, -17.56, -3.69, -27.56, 0.00, 24.81

- ➔ Yaw, Pitch, Roll, X-Acceleration, Y-acceleration, Z-acceleration, X-지자계, Y-지자계, Z-지자계, 수심, 온도
- ➔ 수심 정보의 경우는 해수면을 기준으로 기본 설정이 되어있으므로, 초기 OFFSET 을 설정한 후 사용을 권장합니다.

6. 프로그램 설명

```
#include "S_imu.h"
#include "S_pressure.h"
#include "S_temp.h"
```

```
//#include <time.h>
```

```
double base = 0;
```

```
void setup()
```

```
{
    i2c_com::initialize();           // join i2c bus (address optional for master)
    Serial.begin(115200);           // start serial for output
    S_imu::Initialize();
    S_temp::Initialize();
    while(!S_pressure::Initialize())
    {
        ;
    }
    S_pressure::Getpressure(base);
}
```

I2C 통신 설정

```
double p, t, x0, y0, z0, x1, y1, z1, x2, y2, z2;
```

```
unsigned long time1;
```

변수 설정

```
void loop()
```

```
{
    S_imu::GetRPY(x0, y0, z0);
    S_imu::GetRPY(x1, y1, z1);
    S_imu::GetmRPY(x2, y2, z2);
    S_pressure::Getpressure(p);
    S_temp::Gettemp(t);
    Serial.print(x0);
    Serial.print(",");
    Serial.print(y0);
    Serial.print(",");
    Serial.print(z0);
    Serial.print(",");
    Serial.print(x1);
    Serial.print(",");
    Serial.print(y1);
    Serial.print(",");
    Serial.print(z1);
    Serial.print(",");
    Serial.print(x2);
    Serial.print(",");
    Serial.print(y2);
    Serial.print(",");
    Serial.print(z2);
    Serial.print(",");
    Serial.print(p-base);
    Serial.print(",");
    Serial.println(t);
    delay(10);
}
```

데이터 출력