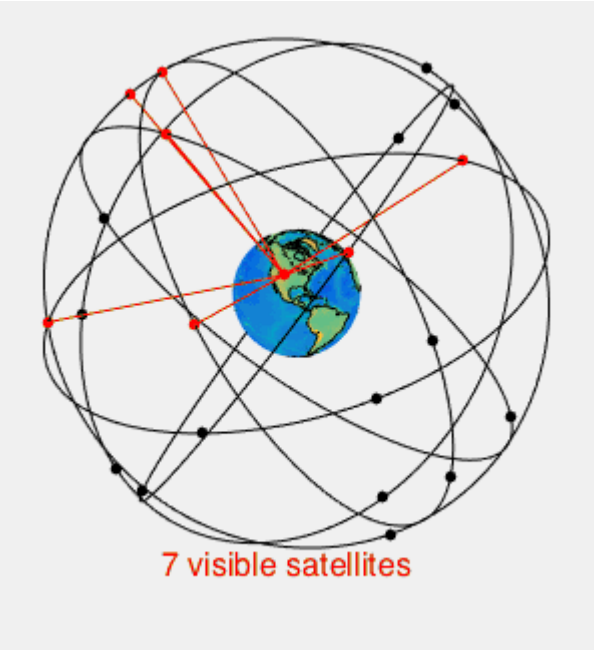
GPS

An **Unmanned aerial vehicle** (UAV) is a Unmanned Aerial Vehicle. UAVs include both autonomous (means they can do it alone) drones and remotely piloted vehicles (RPVs). A UAV is capable of controlled, sustained level flight and is powered by a jet, reciprocating, or electric engine.

GPS 개요

GPS 란?

- GPS(Global Positioning System)
  - 인공위성을 이용하여 지구상의 절대 위치를 알아내는 시스템
- 역사
  - 1970년대 DoD가 제안 NAVSTAR(NAVigation Satellite Timing And Ranging)이라 부름
  - 1978년 군사용 첫 위성 발사
  - 1980년대 민간 용도로 확대
  - 1994년 위성단 구축 완료
    - 지구상의 어떤 위치에서도 언제나 4개 이상의 위성이 보이도록 배치
    - 전체 24개의 6 궤도(21 active, 3 spare)



https://commons.wikimedia.org/wiki/File:GPS24goldenSML.gif

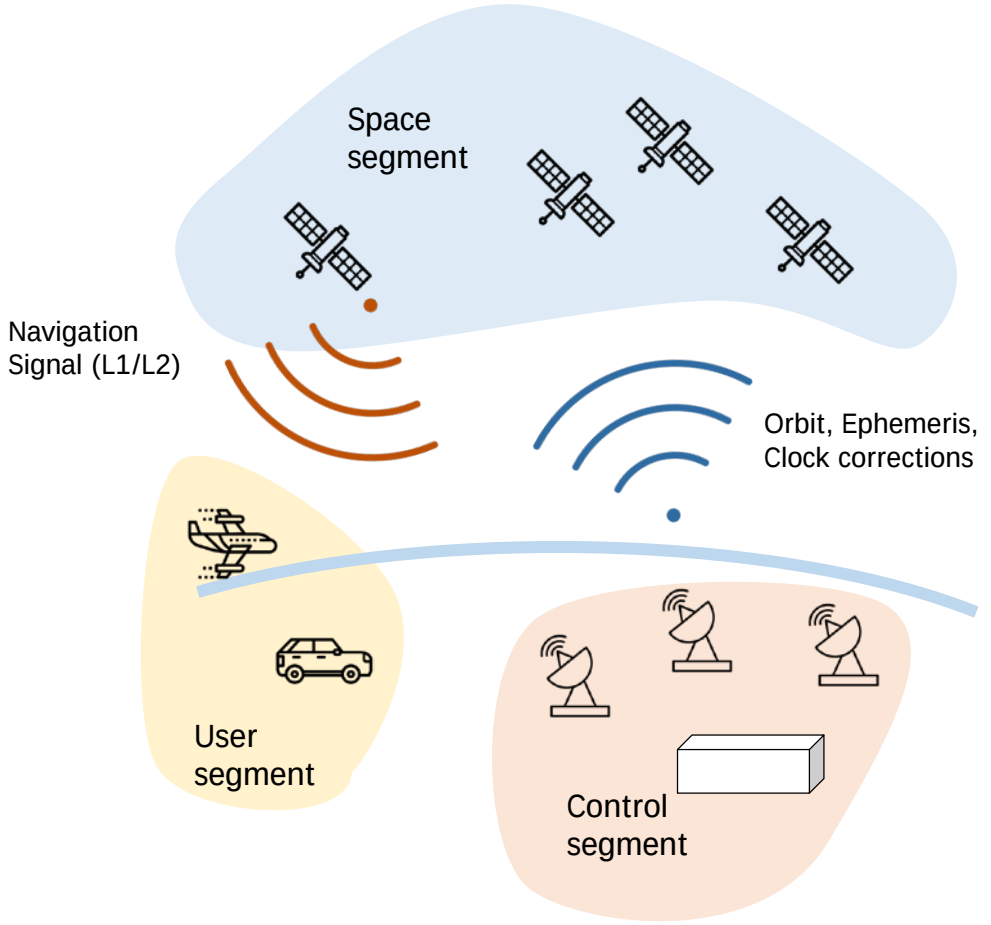
GPS의 특징

- NAVSTAR 서비스의 특징
  - 24시간 365 무료 서비스
  - 전지구적인 3차원 서비스 (극점 제외)
  - 공통의 좌표사용 (WGS84 World Geodetic System 1984)
  - 날씨에 무관
- NAVSTAR외 시스템
  - NAVSTAR 동등
    - GALILEO (EU), GLONASS(Russia)
  - WAAS(Wide Area Augmentation System) 대응
    - EGNOS(EU), MSAS(Japan)

GPS의 구성

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

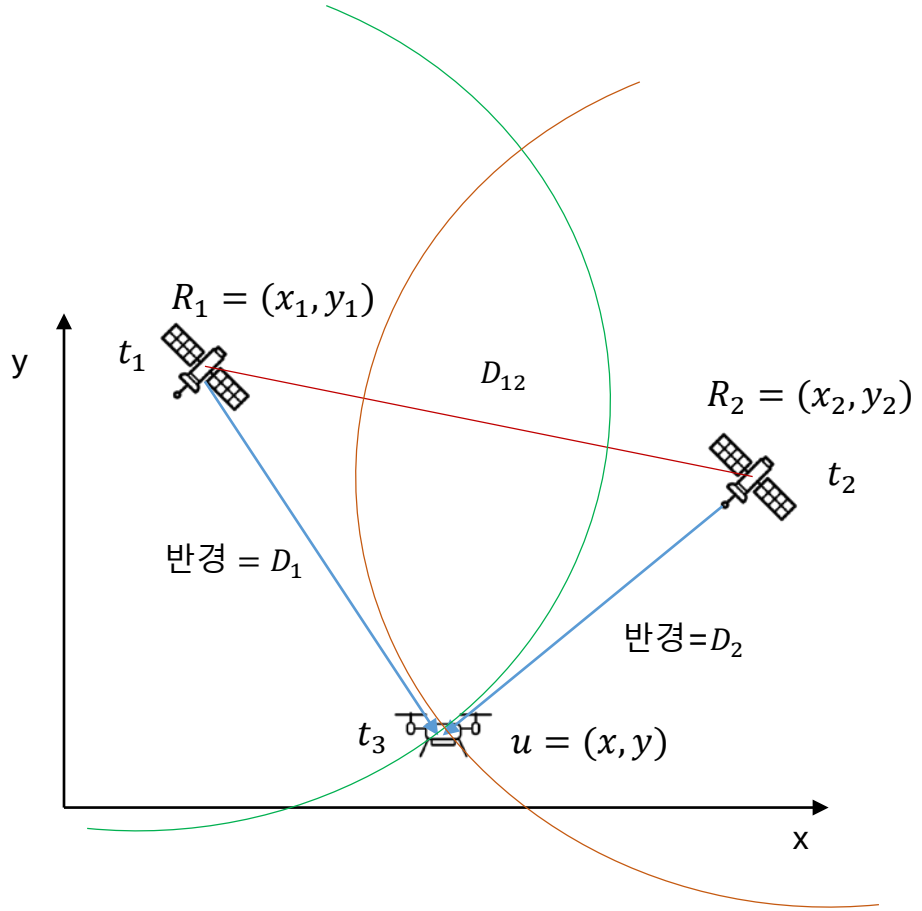
- Space segment
  - 인공위성 24개로 구성된 성단
  - 1궤도에 인공위성 4개
  - 12시간마다 공전 (20,000 km)
- Control segment
  - 위성: 위치, 시간 보정
  - 세계적으로 5군대 위치
- User segment
  - 사용자 위성 신호 분석
  - *Distance = Velocity x Time*



2차원 문제

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

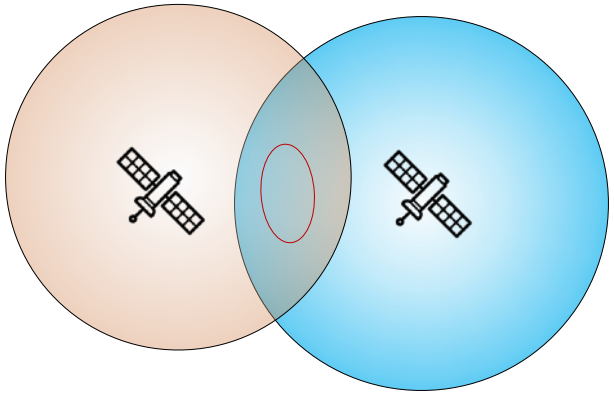
- 2차원 평면에서 2개의 인공위성
  - 가정
    - S_1, S_2의 위치 (x_1, y_1), (x_2, y_2) 는 알고 있음
    - S_1은 t_1시간에 S_2는 t_2시간에 자기 위치와 발신 시각을 전파로 전송
  - triangulation 절차
    - 수신장치 u는 S_1, S_2의 전파로 부터 수신 시각 t_3과 발신 시각 t_1, t_2의 차이로 각각의 거리 R_1, R_2를 계산
    - 다음 2개의 식을 풀어 x, y를 구함
      - $(x - x_1)^2 + (y - y_1)^2 = R_1^2$
      - $(x - x_2)^2 + (y - y_2)^2 = R_2^2$



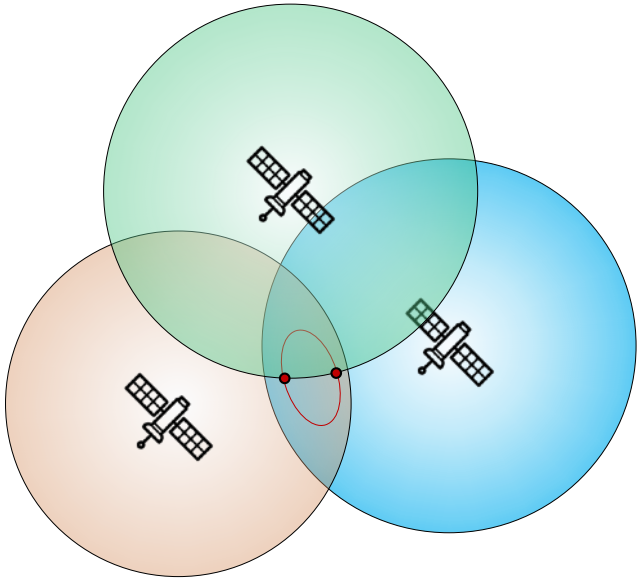
3차원 공간 문제

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

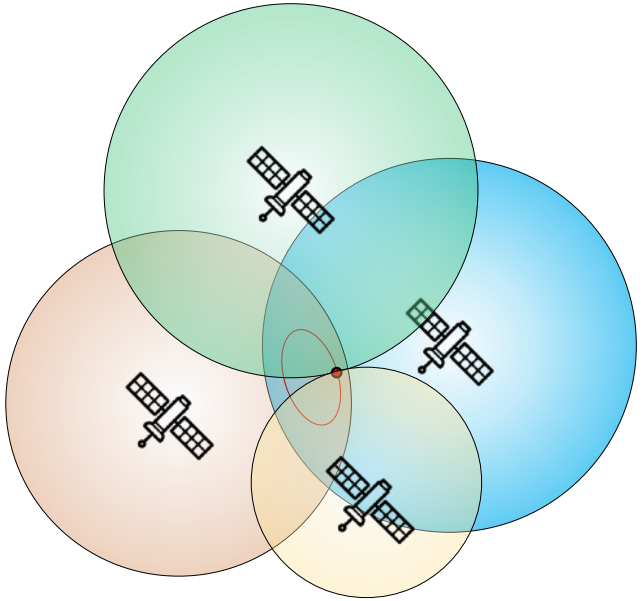
- 3차원 공간에서 n개의 인공위성
  - 2개 구의 교점: 원
  - 3개 구의 교점: 2개의 점
  - 4개 구의 교점: 1개의 점
  - 식 4개 미지수 3개
  - 다음 식을 풀어 x, y, z 를 구함
    - $(x-x_i)^2+(y-y_i)^2+(z-z_i)^2= R_i^2$
      $i = 1, 2, 3, 4$



2개의 위성 교점 : 원



3개의 위성 교점 : 두 점

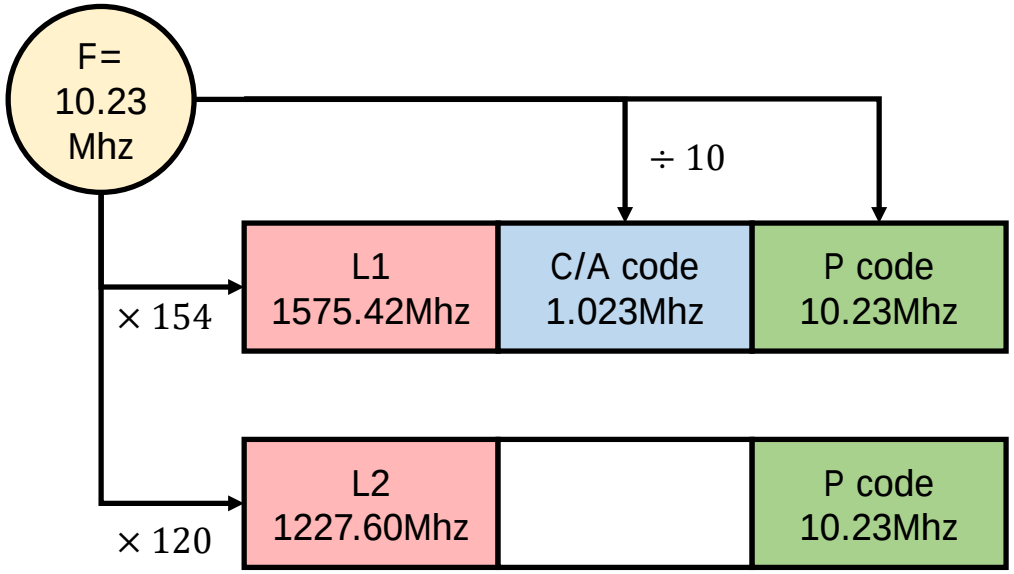


4개의 위성 교점 : 한 점

인공위성의 주파수

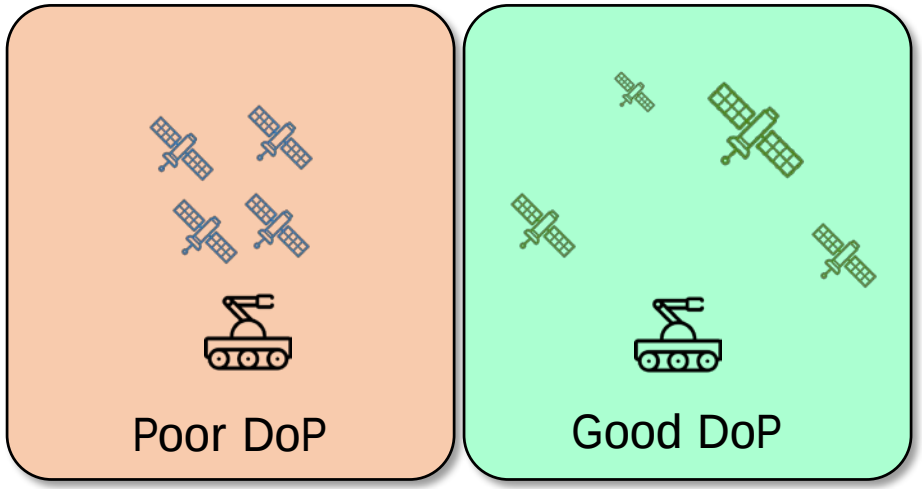
Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- 기본 주파수
  - 10.23Mhz
- L1 주파수: 1575.42Mhz
  - Coarse Acquisition (C/A) code
  - Precision (P) code
- L2 주파수: 1227.60Mhz
  - Precision (P) code



GPS의 오차

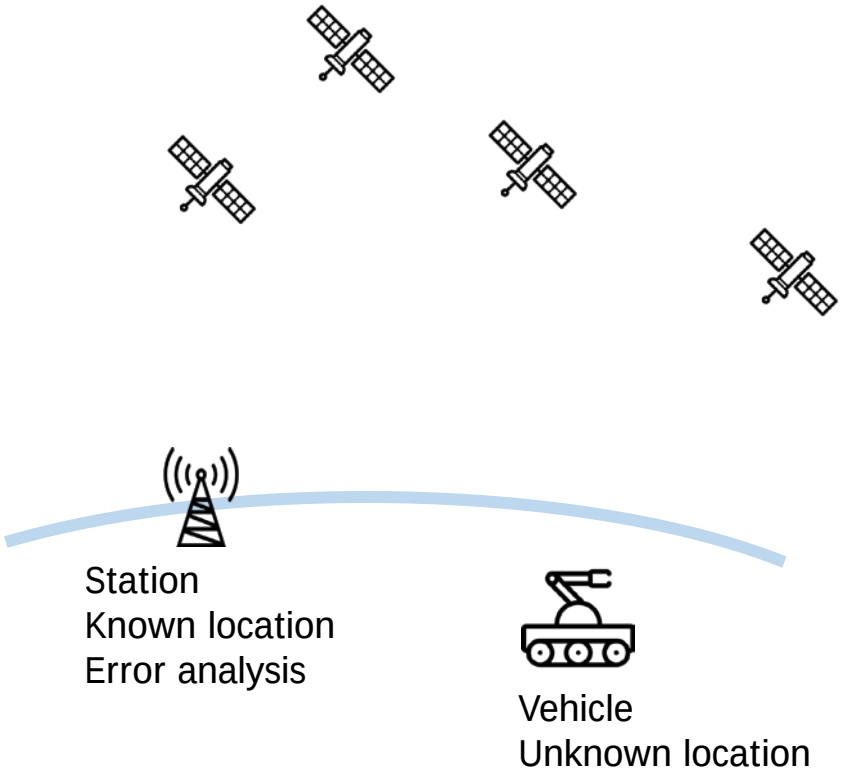
- DoP (Dilution of Precision)
  - 위성의 기하학적 배치 때문에 나타나는 척도
    - 값이 작을 수록 좋음
    - HDOP - Horizontal Only
    - VDOP - Vertical Only
    - PDOP - Positional in 3D
    - GDOP - Geometric in 3D and Time
- 오차 원인
  - 위성: DOP (기하학적 배치, 가림)
    - DoD의 의도적 정밀도 저하. (2002년 이후 중지f)
  - 수신기:
    - Clock 오차, 잡음
  - 환경:
    - 대기의 전달 지연, 전리층(ionospheric)반사, Multi-path



DGPS

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- Differently Corrected GPS
  - 위치를 알고있는 station 을 이용하여 오차 보정 데이터 계산
  - 이 보정 데이터를 전송하여 이동체의 위치를 보정
  - 정지궤도 위성(WAAS) 사용하는 경우도 있음.
  - 정밀도 1m 이내까지



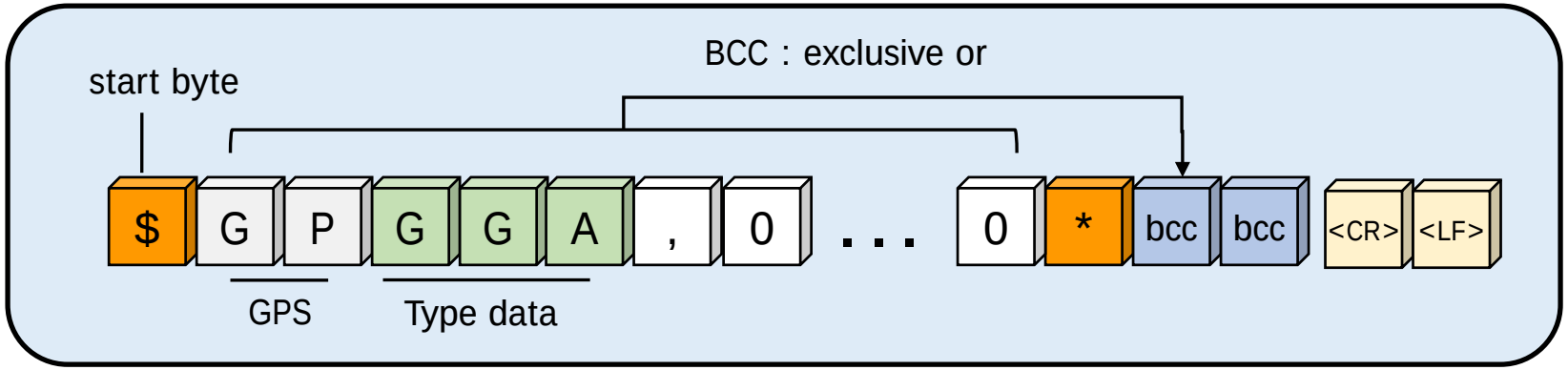
NMEA 프로토콜

Protocol

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

NMEA 0183

- National Marine Electronics Association에서 정의한 GPS, INS 용 프로토콜
  - '**$**' 시작하고 BCC의 앞에는 '*' 를 붙임.
  - 항목들은 ','로 구분 하고 맨 뒤에는 <CR><LF>로 끝남



- 예

$GPGGA,114455.532,3735.0079,N,12701.6446,E,1,03,7.9,48.8,M,19.6,M,0.0,0000*48

Protocol

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

GGA Type 해석 예

- GGA: 시간, 경도, 위도, 고도 등
  - 시각: 113849.00 → 11:38:49.00
  - 위도: 3737.87564,N → 북위 37도37.87564분
  - 경도: 12704.56710,E → 동경127도 4.56710분
  - Fix의 종류: 0/위성 없음, 1/기본 위성, 2/DGPS
  - 사용 위성 개수: 11개
  - Horizontal DoP: 2.04
  - 해수면 고도: 72.8m
  - 지구를 구체와 타원체로 본경 우의 오차: 18.6m
  - DGPS 사용 시각과 기지국 ID: ,,0000
  - BCC: 0x60

GGA Type data

$GPGGA,113849.00,

시간

3737.87564,N,12704.56710,E,2,

위도(latitude) 경도(longitude)

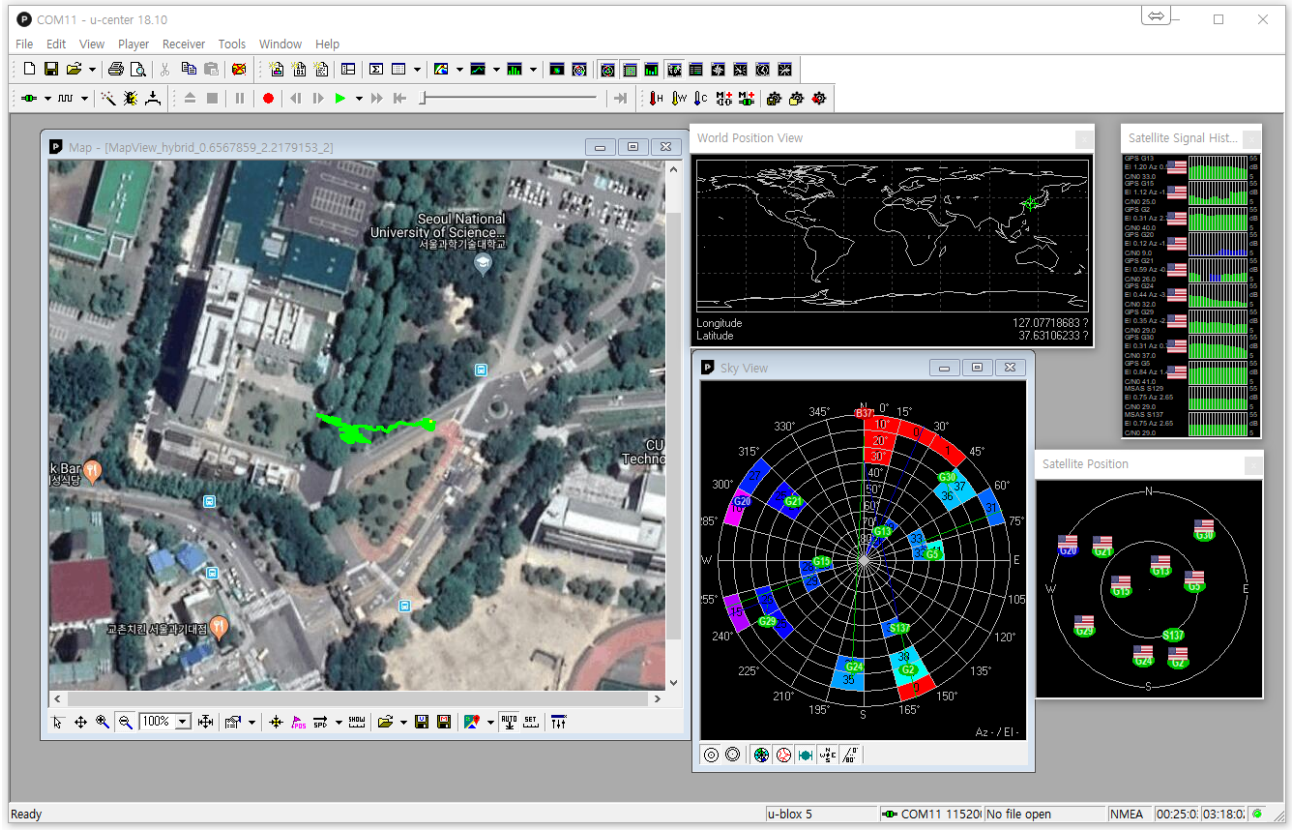
11,2.04,72.8,M,18.6,M,,0000*60

위성수 고도 BCC

GPS 평가 소프트웨어

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- u-center 프로그램
  - GNSS 평가용
  - NMEA 프로토콜
  - 다양한 정보 확인가능
    - https://www.u-blox.com/

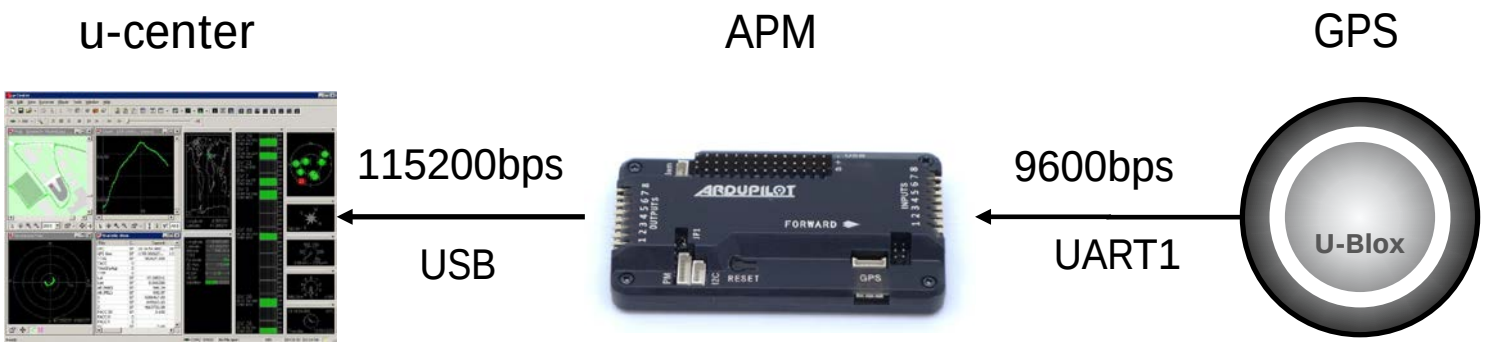


Test code

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

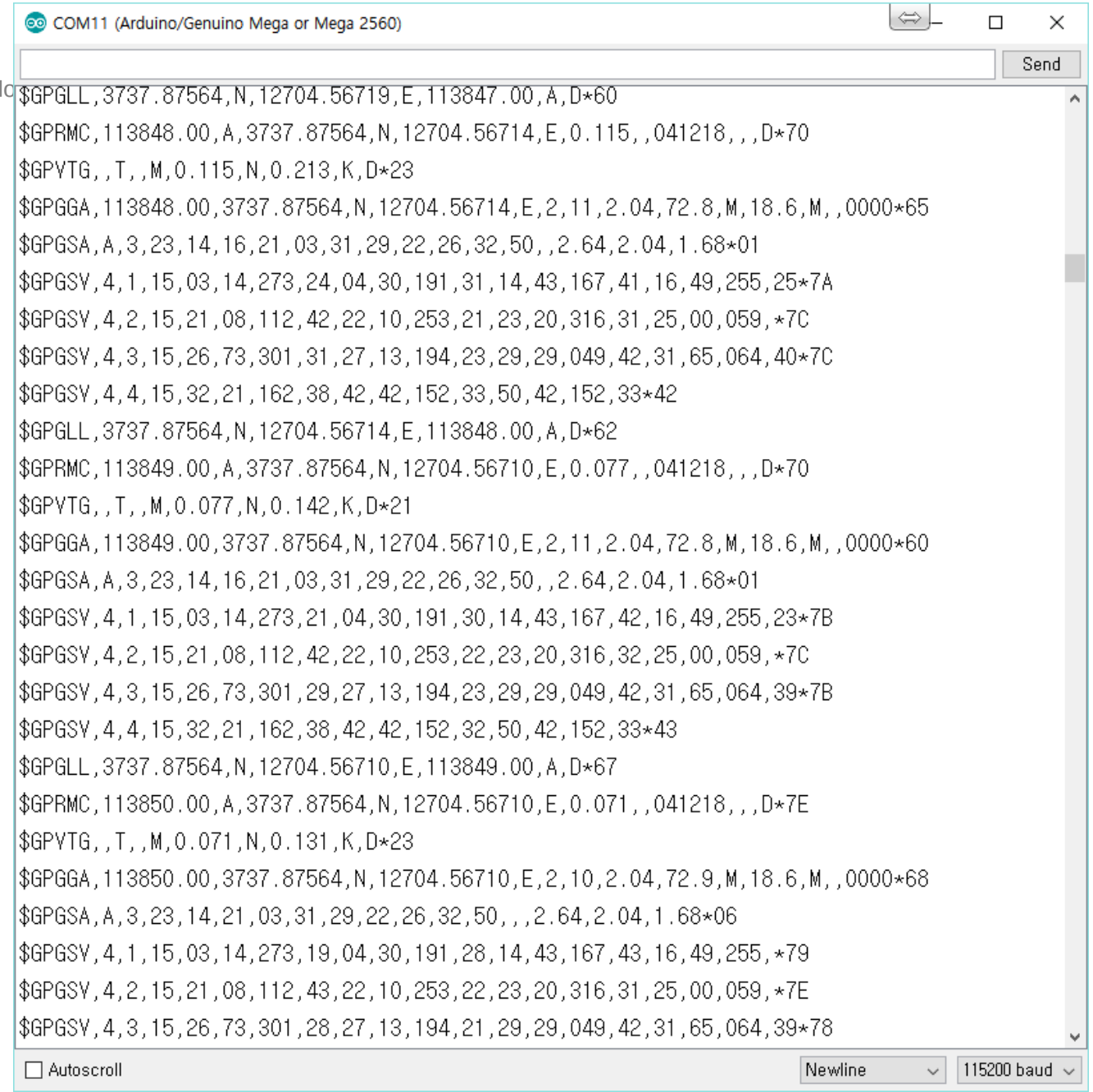
- u-center 용 Bridge code 이용
  - baud rate 변환

```
void setup(void){
  Serial.begin(115200);
  Serial1.begin(9600);
}
void loop(void){
  while(Serial1.available())  {
    Serial.print((char)Serial1.read());
  }
}
```



Packet 출력

- DATA Type
  - GGA: Fix data
  - GSA: DoP and active satellite
  - GSV: 위성의 정보
  - GLL: 경도,위도, 시간
  - RMC: 최소의 데이터
  - VTG: Tracking 과 speed

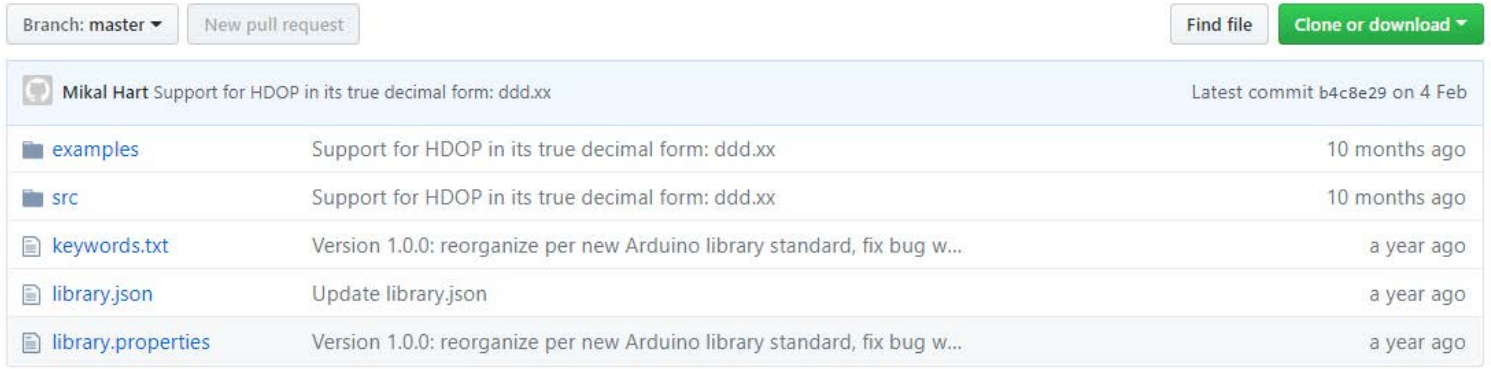


GPS 프로그래밍

TinyGPSPlus library

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- 아두이노용 NMEA 해석 라이브러리
  - download
    - https://github.com/mikalhart/TinyGPSPlus
  - install
    - `C:\Users\user\Documents\Arduino\libraries`
    - TinyGPSPlus 디렉토리 아래 그림 파일들 복사
    - 아두이노 IDE를 다시 시작



TinyGPSPlus 사용법

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

TinyGPSPlus 라이브러리 사용

- 라이브러리 사용

```
#include <TinyGPS++.h>
```

  - 라이브러리 include

- 수신 데이터 처리

```
bool encode(char c);
```

  - *c* : 수신된 1개의 문자 처리

- 유효성 확인

```
bool isValid()
```

  - 해당 구조체의 데이터가 유효한지 확인

- 새로운 데이터 확인

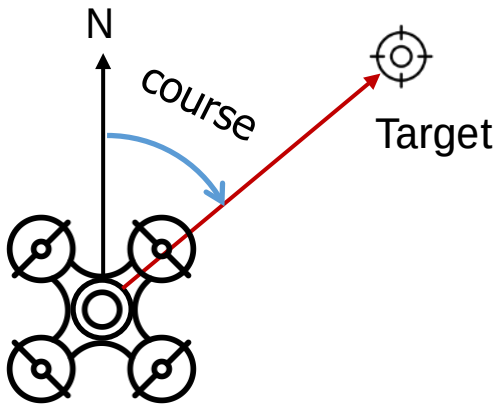
```
bool isUpdated()
```

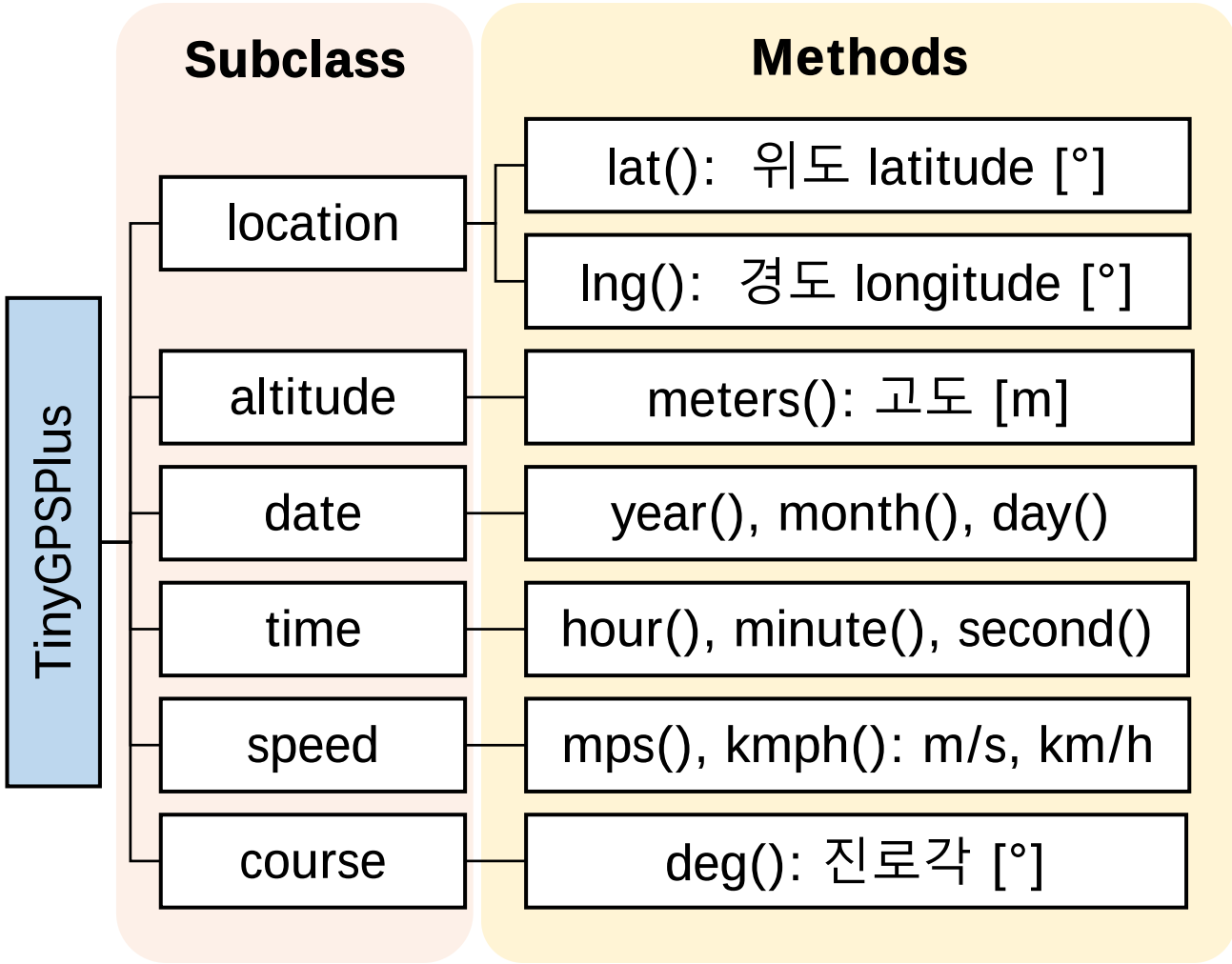
  - 해당 정보가 새로운 정보인지 확인

Class 내의 subclass

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- TinyGPSPlus 부클래스
  - location
  - altitude
  - date
  - time
  - speed
  - course : 진로각





적용예제

- NeoSerial library이용
  - UART0: NeoSerial, UART1: NeoSerial1

```
#include <NeoHWSerial.h>
#include <TinyGPS++.h>
#define GPSBaud 9600
#define PCBaud 115200
TinyGPSPlus gps;
void setup() {
  NeoSerial.begin(PCBaud);
  NeoSerial1.begin(GPSBaud);
  NeoSerial1.attachInterrupt( recvData );
  delay(1000);
}
void loop(){
  if( gps.altitude.isValid()&&gps.altitude.isUpdated()){
    NeoSerial.print("Alt= "); NeoSerial.println(gps.altitude.meters());
  }
```

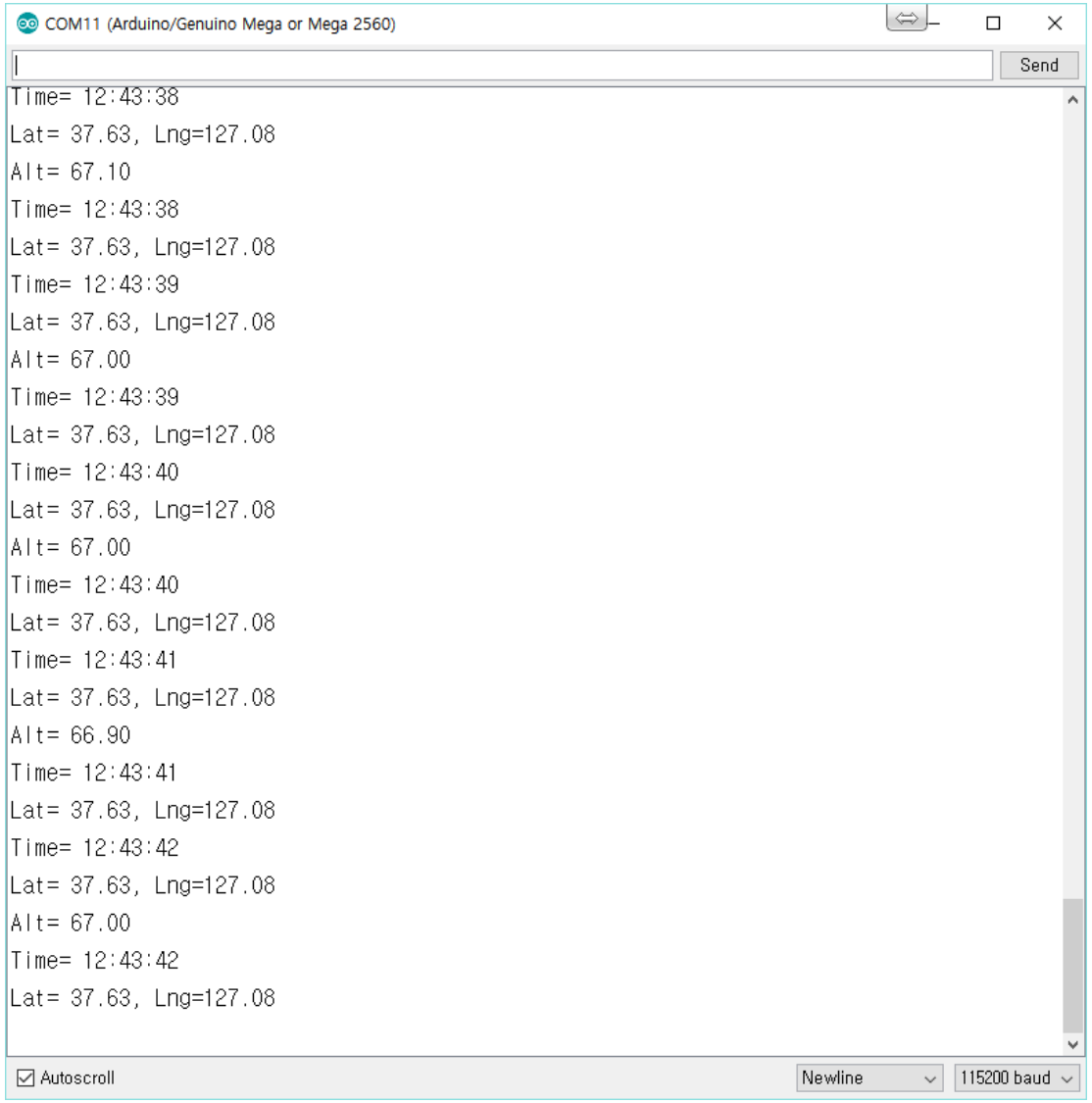
적용예제

주요 내용

- isValid(), isUpdate()
- recvData() : UART1 ISR

```
 if( gps.time.isValid()&&gps.time.isUpdated()){
    NeoSerial.print("Time= ");NeoSerial.print((gps.time.hour()+9)%24);
    NeoSerial.print(":");NeoSerial.print(gps.time.minute());
    NeoSerial.print(":");NeoSerial.println(gps.time.second());
  }
  if( gps.location.isValid()&&gps.location.isUpdated()){
    NeoSerial.print("Lat= "); NeoSerial.print(gps.location.lat());
    NeoSerial.print(", Lng=");NeoSerial.println(gps.location.lng());
  }
}
void  recvData(uint8_t c){
    gps.encode((char)c);
}
```

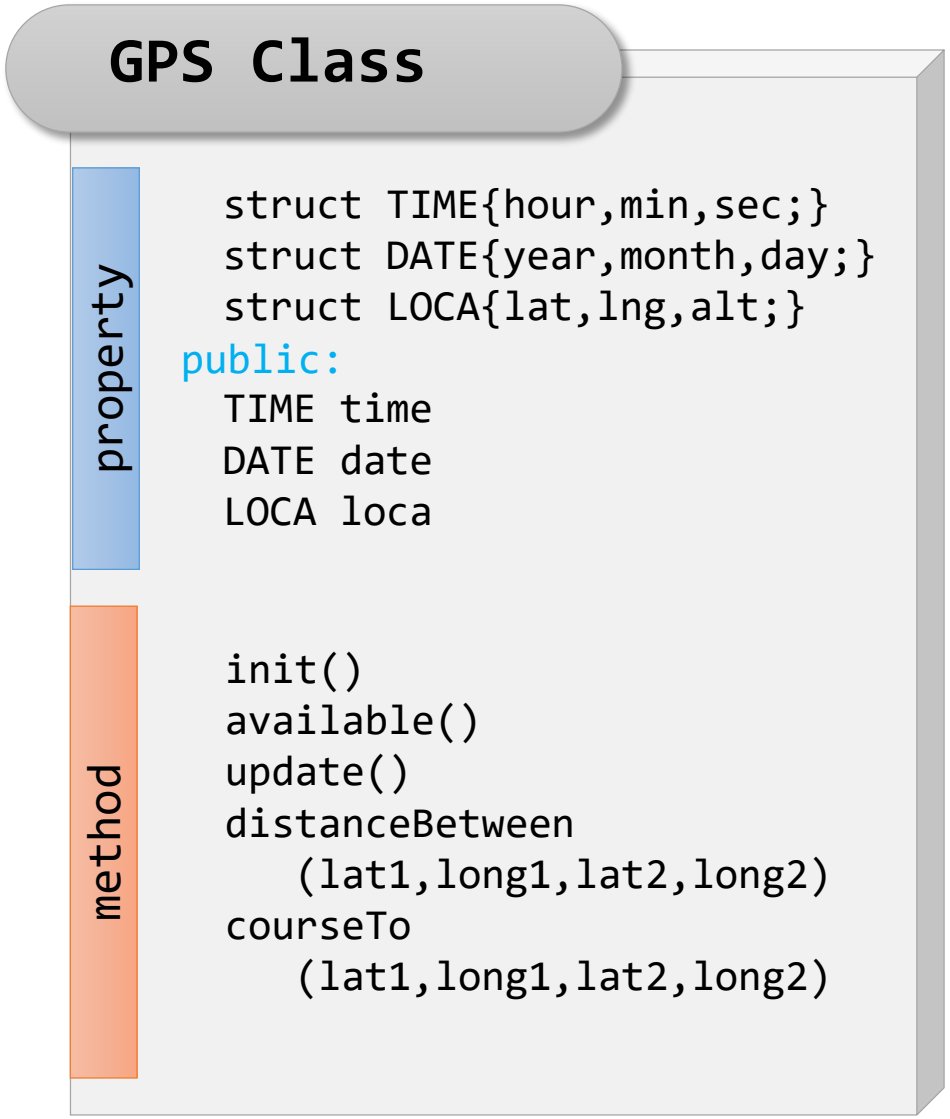
실행결과


COM11 (Arduino/Genuino Mega or Mega 2560)
Send
Time= 12:43:38
Lat= 37.63, Lng=127.08
Alt= 67.10
Time= 12:43:38
Lat= 37.63, Lng=127.08
Time= 12:43:39
Lat= 37.63, Lng=127.08
Alt= 67.00
Time= 12:43:39
Lat= 37.63, Lng=127.08
Time= 12:43:40
Lat= 37.63, Lng=127.08
Alt= 67.00
Time= 12:43:40
Lat= 37.63, Lng=127.08
Time= 12:43:41
Lat= 37.63, Lng=127.08
Alt= 66.90
Time= 12:43:41
Lat= 37.63, Lng=127.08
Time= 12:43:42
Lat= 37.63, Lng=127.08
Alt= 67.00
Time= 12:43:42
Lat= 37.63, Lng=127.08
Autoscroll
Newline
115200 baud

GPS class 화

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

- GPS class 정의
  - Property
    - time: 시간 구조체 변수
    - date: 날짜 구조체 변수
    - loca: 위치 구조체 변수 (위도,경도,고도)
  - Method
    - init(): 초기화 함수
    - available(): 데이터가 유효성 및 갱신 여부
    - update(): 데이터 읽기
    - distanceBetween(): 두 지점의 거리
    - courseTo(): 1에서 2로 가기 위한 방향각



헤더파일

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

주요내용

GPS.h

```
#ifndef __GPS_h
#define __GPS_h
#define GPSBaud 9600
#include <NeoHWSerial.h>
#include <TinyGPS++.h>
#include <Arduino.h>
void  recvData(uint8_t c);
class GPS{
   struct TIME{uint8_t hour,min,sec;};
   struct DATE{uint8_t year,month,day;};
   struct LOCA{float lat,lng,alt;};
  public:
    void init();
    bool available();
    void  update();
    float distanceBetween(float lat1,float long1,float lat2, float long2);
    float courseTo(float lat1,float long1, float lat2,float long2);
    TIME time;
    DATE date;
    LOCA loca;
};
#endif
```

C++ 파일

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

주요내용

GPS.cpp [1/2]

```
#include "GPS.h"
TinyGPSPlus g;
void GPS::init(){
  NeoSerial1.begin(GPSBaud);
  NeoSerial1.attachInterrupt( recvData );
  delay(1000);
}
bool GPS::available(){
  bool a=g.altitude.isValid()&& g.altitude.isUpdated();
  bool b=g.location.isValid()&&g.location.isUpdated();
  return (a||b)? 1:0;
}
void GPS::update(){
  loca.alt=g.altitude.meters();
  loca.lat=g.location.lat();
  loca.lng=g.location.lng();
  time.hour=(g.time.hour()+9)%24; // UMT +9 hour
  time.min=g.time.minute();
  time.sec=g.time.second();
}
```

C++ 파일

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

주요내용

GPS.cpp [2/2]

```
float GPS::distanceBetween(float lat1,float long1,float lat2, float long2){
  return g.distanceBetween(lat1, long1, lat2, long2);
}
float GPS::courseTo(float lat1,float long1, float lat2,float long2){
  float ang = g.courseTo(lat1, long1, lat2, long2);
  return (ang>180)? ang-360:ang;  //+- 180 dgree
}
void  recvData(uint8_t c){
    g.encode((char)c);
}
```

C++ 파일

Dept. of Mechanical System Design, Seoul National University of Science and Technology.

주요내용

tinyGPS_Class.ino

```
#include <NeoHWSerial.h>
#include "GPS.h"
#define PCBaud 115200
GPS gps;
void setup() {
  NeoSerial.begin(PCBaud);
  gps.init();
}
void loop(){
  if(gps.available()) gps.update();
  NeoSerial.print("Time= ");NeoSerial.print(gps.time.hour);
  NeoSerial.print(":");NeoSerial.print(gps.time.min);
  NeoSerial.print(":");NeoSerial.print(gps.time.sec);
  NeoSerial.print(", Alt= "); NeoSerial.print(gps.loca.alt);
  NeoSerial.print(", Lat= "); NeoSerial.print(gps.loca.lat);
  NeoSerial.print(", Lng=");NeoSerial.print(gps.loca.lng);
  NeoSerial.print(", Crs=");
  NeoSerial.println(gps.courseTo(37.63,127.08,37.63,126.99));
  delay(100);
}
```

실행결과

Dept. of Mechanical System Design, Seoul National University of Science and Tec

COM11 (Arduino/Genuino Mega or Mega 2560)

Send

```
Time= 14:49:28, Alt= 77.10, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.10, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.10, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:29, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 77.00, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:30, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:31, Alt= 76.90, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:31, Alt= 76.80, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:31, Alt= 76.80, Lat= 37.63, Lng=127.08, Crs=-89.97
Time= 14:49:31, Alt= 76.80, Lat= 37.63, Lng=127.08, Crs=-89.97
```

Autoscroll | Newline | 115200 baud

THANK YOU

Powerpoint is a complete presentation graphic package it gives you everything you need to produce a professional-looking presentation