# IMU Sensor 융합

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.

# CONTENTS


**01 Sensor Fusion**
Sensor Fusion에 대하여 알아보자.

**02 Complementary Filter**
Complementary Filter의 원리에 대하여 알아본다.

**03 Programming the Filter**
Complementary Filter 프로그램에 대하여 알아본다.

**04**

**05**

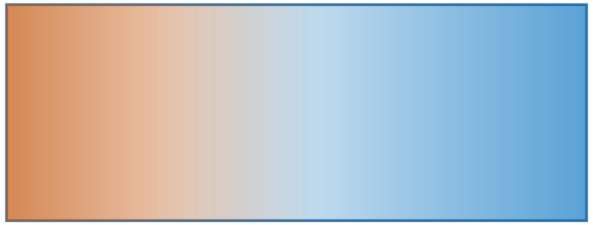

# Sensor Fusion

# Why sensor fusion?

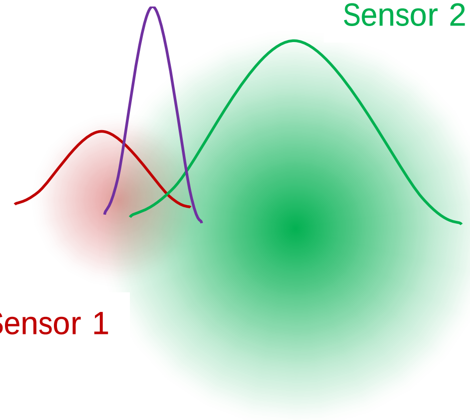
Dept. of Mechanical System Design, Seoul National University of Science and Technology.

## 센서 융합의 방법

- 상보필터 (Complementary filter):
  - 센서의 서로 다른 주파수 특성으로 역할 분담




- 칼만 필터 (Kalman filter):
  - 확률적 특성을 미리 대략 짐작할 수 있고 동역학적 모델을 구하는 것이 가능한 경우



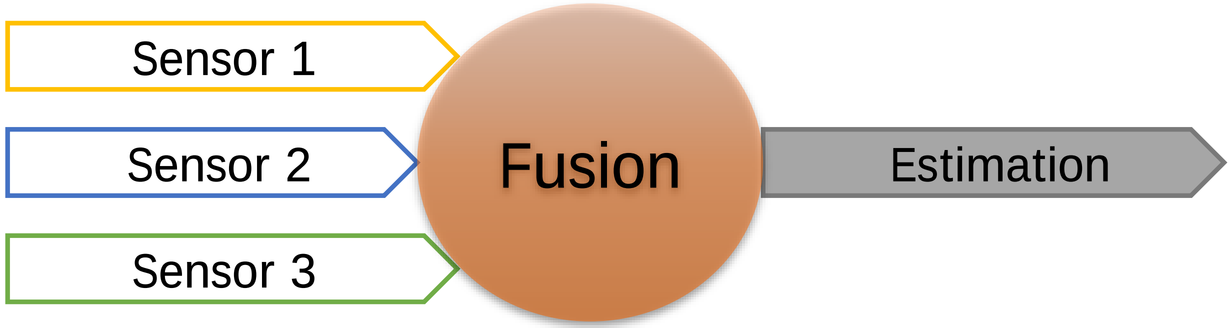

- Magdwick 필터:
  - 9개의 정보에서 3개의 미지수를 찾는 최적화 문제로 접근. 4원수(Quaternion) 사용함.

# Sensor Fusion for IMU



## 센서 융합의 개념

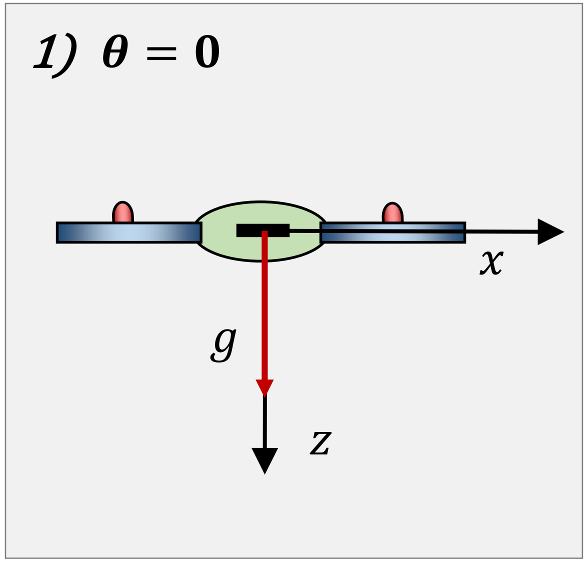
- 한 종류의 물리량에 대하여 서로 다른 특징의 여러 개의 센서로 측정한다.
- 한 센서의 부족한 단점은 다른 센서가 보완해준다.
- 예) 측정 속도가 빠른 부정확한 센서 + 측정 속도가 느린 정확한 센서

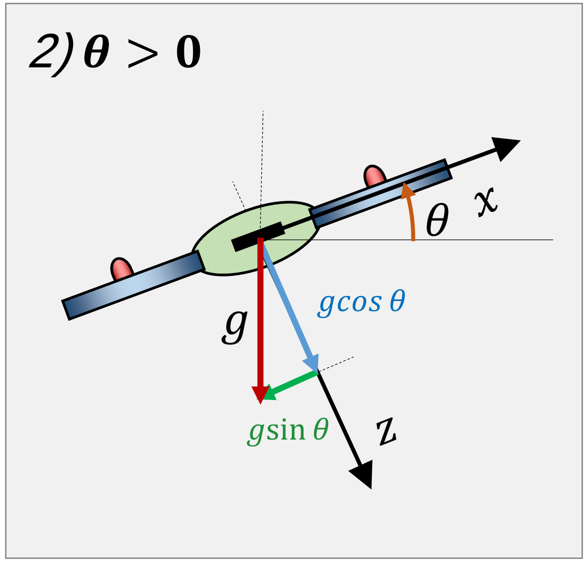


# 각도 측정의 원리 - Accelerometer

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

- roll이 0인 경우 가속도 센서를 이용한 pitch각 $\theta$를 구한다.
  - $\theta = 0$ 인 경우 $a_z = g$ , $a_x = 0$
  - $\theta > 0$ 인 경우 $a_z = g\cos\theta$ , $a_x = -g\sin\theta$
  - $\therefore \theta = \tan^{-1}\dfrac{-a_x}{a_z}$




- 문제점
  - 잡음에 취약하다.
  - 운동 가속도에도 영향을 받는다.

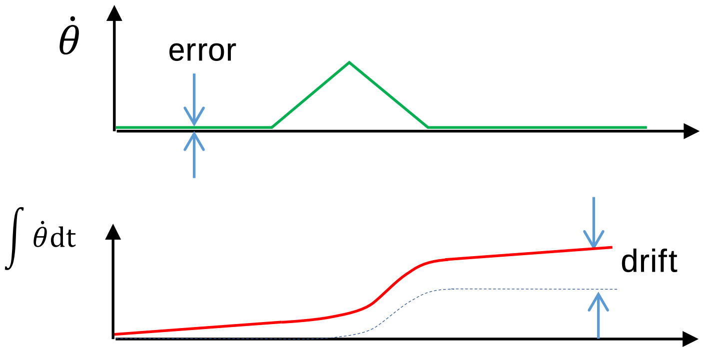
# 각도측정의 원리 - Gyro

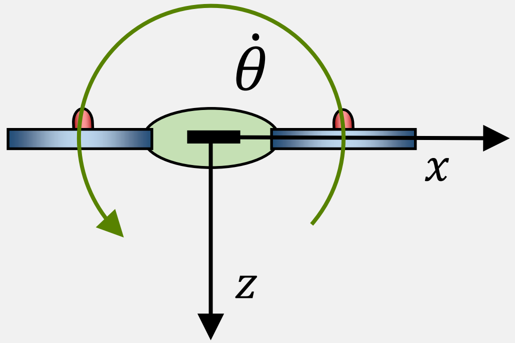
Dept. of Mechanical System Design, Seoul National University of Science and Technology.

## Gyro 센서를 이용한 pitch $\theta$

- gyro 센서는 근본적으로 각속도를 측정하는 센서
- $\theta$ 를 구하려면 각속도 $\dot{\theta}$를 적분

- 문제점: Drift 현상
  - 작은 오차 적분에 의하여




$$\theta = \int_0^t \dot{\theta} d\tau$$

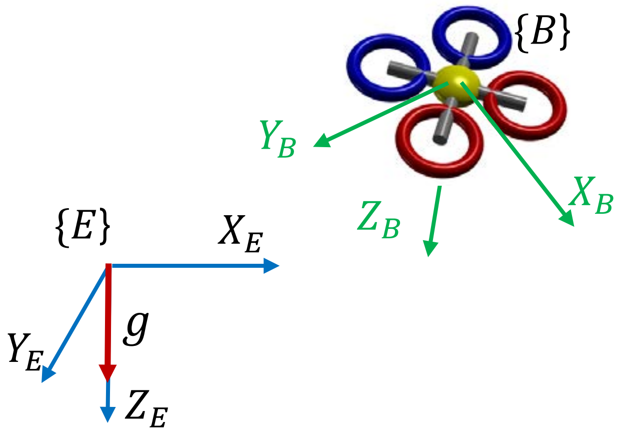
# 가속도 센서로부터의 각도

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

- {E}에서 본 중력가속도 $\boldsymbol{g_E}$를 {B}에서 보면 $\boldsymbol{g_B}$ 이다.
  - {E}에서 본 {B}의 회전행렬을 $R_B^E$ 라고 하면
    - $\boldsymbol{g_E} = R_B^E \boldsymbol{g_B} = [0 \quad 0 \quad g]^T$
    - $\boldsymbol{g_B} = R_E^B \boldsymbol{g_E} = (R_B^E)^T [0 \quad 0 \quad g]^T$

$$= \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\phi s\theta c\psi - c\phi s\psi & s\phi s\theta s\psi + c\phi c\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & c\phi s\theta s\psi - s\phi c\psi & c\phi c\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$$

  - 외부의 가속도가 없다고 가정
    - 가속도 센서 측정값 $\boldsymbol{a} = [a_x \quad a_y \quad a_z]^T$는 $|\boldsymbol{a}| = g$
    - $\boldsymbol{a} = [a_x \quad a_y \quad a_z]^T = [-g \sin\theta \quad g \sin\phi \cos\theta \quad g \cos\phi \cos\theta]^T$
    - $\phi = tan^{-1}(\frac{a_y}{a_z})$
    - $\theta = tan^{-1}\left(\frac{-a_x}{a_y \sin\phi + a_z \cos\phi}\right) = tan^{-1}(\frac{-a_x}{\sqrt{a_y{}^2 + a_z{}^2}})$




$$R_B^E = R(\psi, z)R(\theta, y)R(\phi, x)$$
$$= \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix}$$

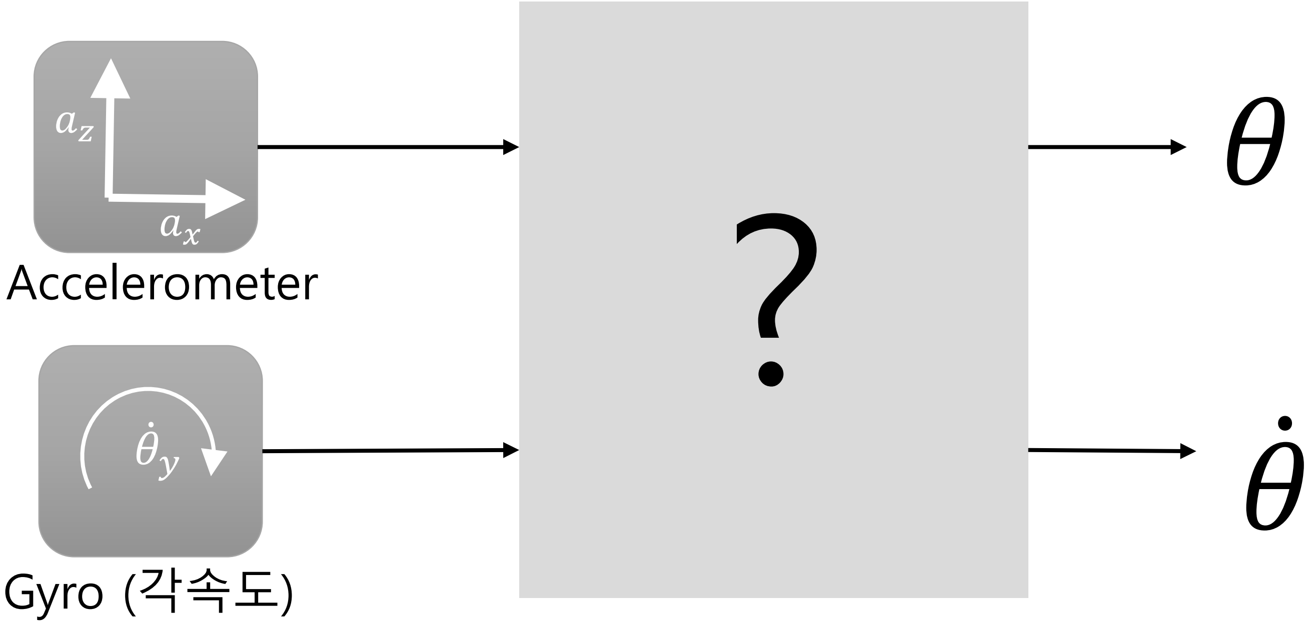

# Complementary Filter

# IMU 센서의 융합

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

## IMU에서의 센서융합 문제

- 가속도 센서와 Gyro의 정보로 부터 최적의 각도 $\theta$ 각속도 $\dot{\theta}$를 구하는 문제

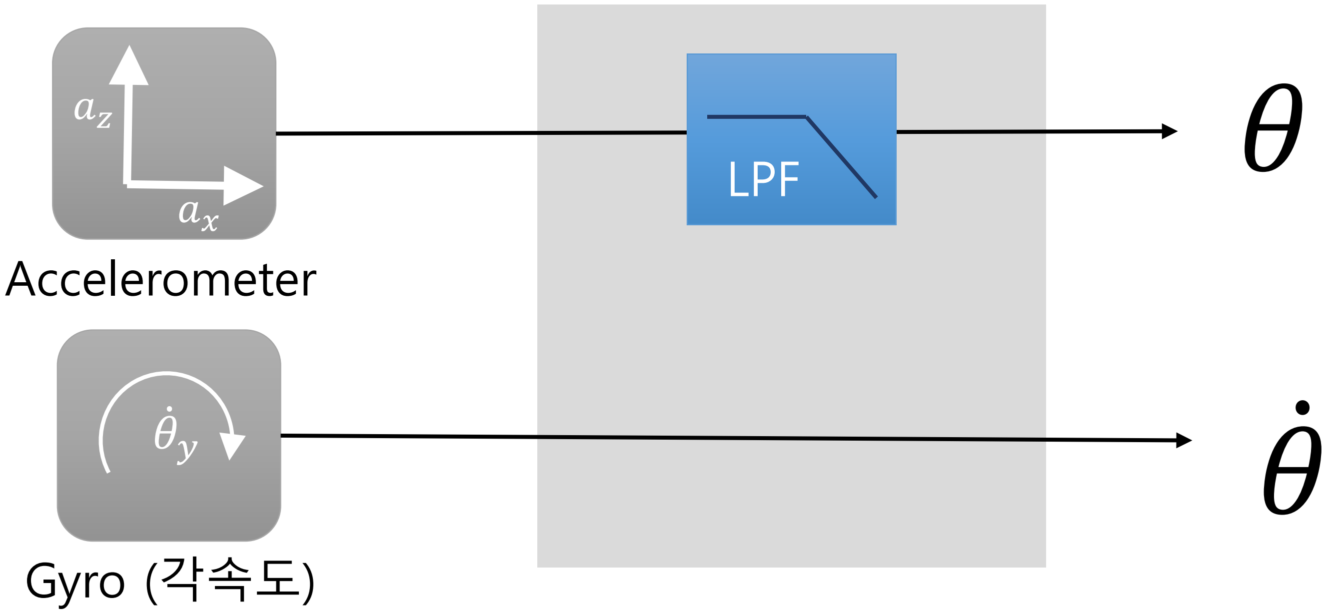


# IMU 센서의 융합

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

## 방법1

- 가속도센서의 고주파노이즈 제거 저역통과 필터(Low pass filter)적용



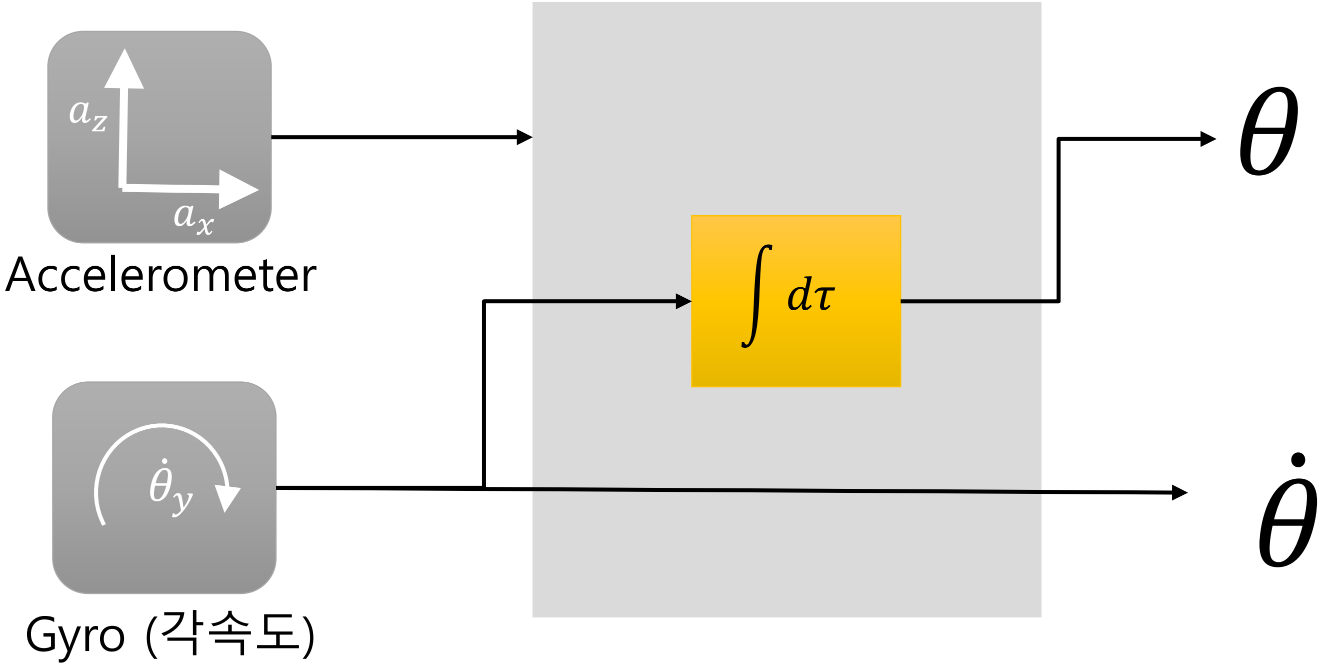

**문제점:**
빠르게 변하는 각도의
변화는 전부 손실됨.

# IMU 센서의 융합

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

## ■ 방법2

- 가속도 센서는 쓰지않고 자이로의 각속도를 적분하여 각도를 구함.



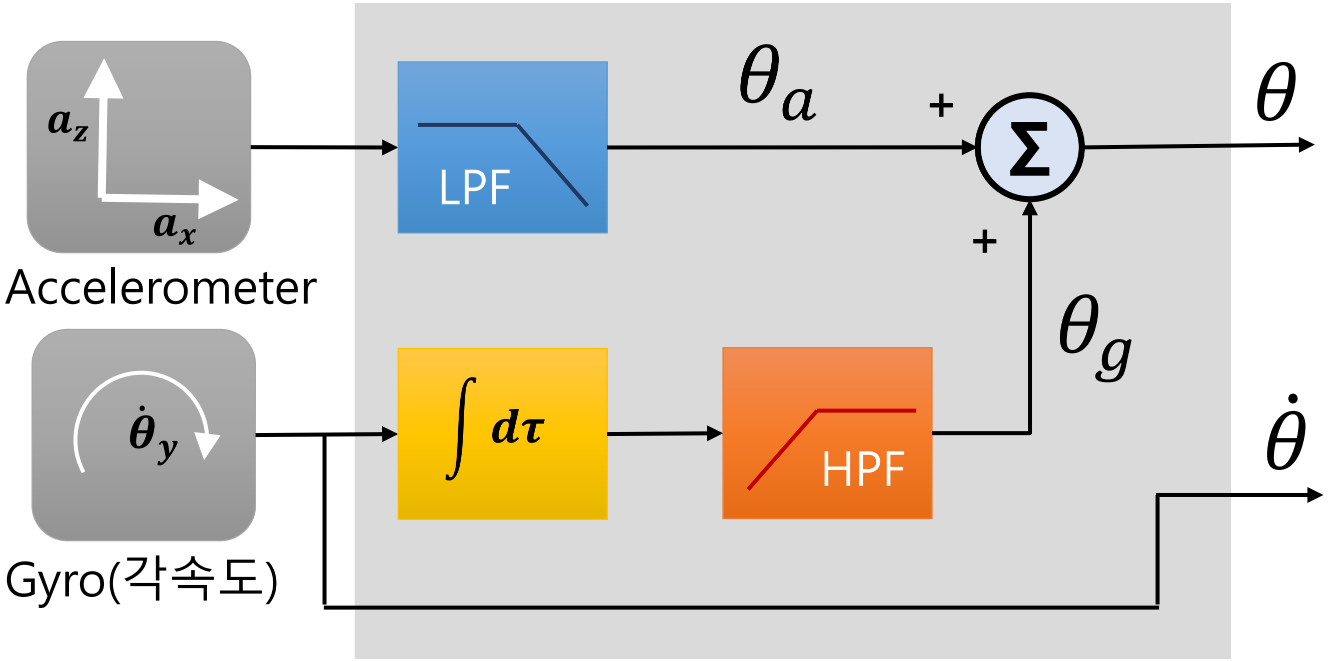

**문제점:**
아주 작은 각도 오차가 누적되어 쌓이는 Drift 현상 발생

# IMU 센서의 융합

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

## 방법3

- LPF(저역통과) + HFP(고역통과필터) + 적분기 사용.

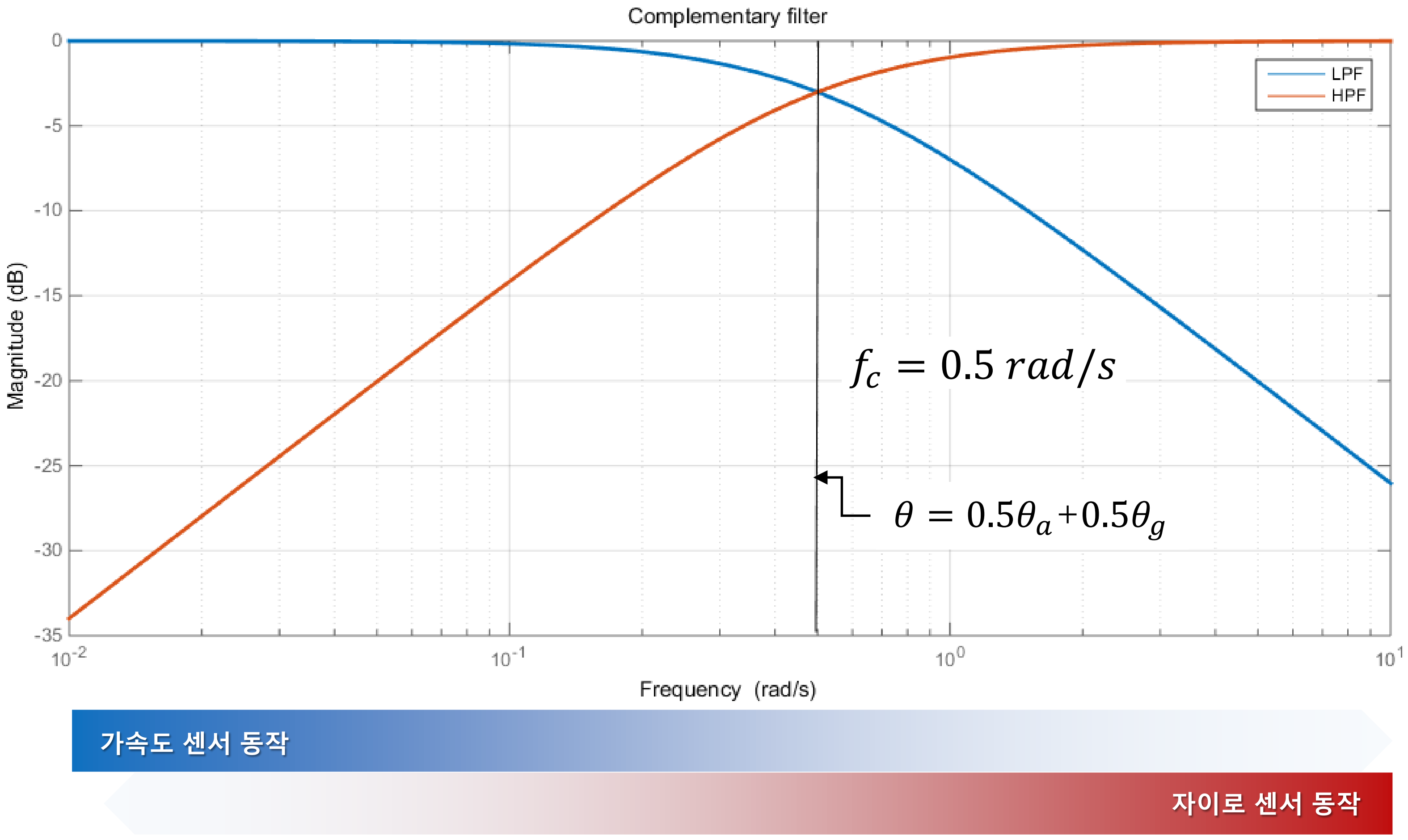



**문제점:**
아주 작은 각도 오차가 누적되어 쌓이는 Drift 현상 발생

# 상보필터의 주파수 응답특성

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

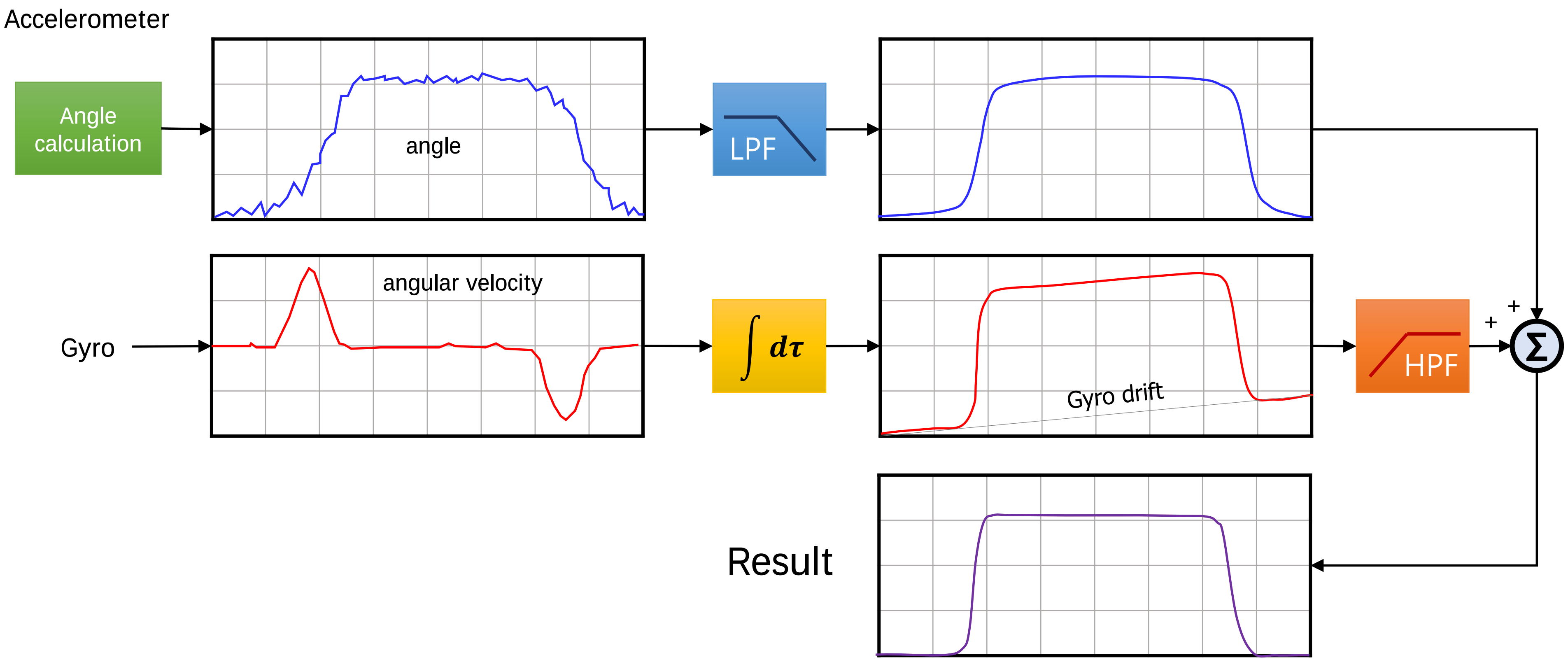
## ■ 주파수응답특성



# 센서융합 (Fusion)

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

## 상보 (Complementary) 필터



# Complementary filter

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

## 1st order filter in continuous time:

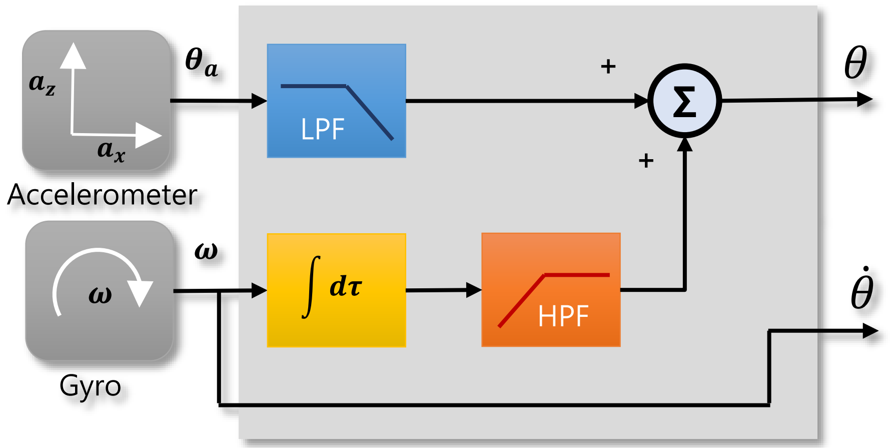
- Transfer function

$$\theta = \frac{1}{T_c s + 1}\theta_a + \frac{T_c s}{T_c s + 1}\frac{1}{s}\omega = \frac{\theta_a + T_c \omega}{T_c s + 1}$$

LPF HPF Integrator

$T_c$: 시상수 $(= \frac{1}{\omega_c})$

$\omega_c$ : cutoff 주파수 [rad/s]



# 이산시간 상보필터 유도

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

## 이산시간 상보 필터 유도

- 전달함수로부터

$$\theta = \frac{\theta_a + T_c\omega}{T_c s + 1}$$

- 미분방정식을 구함

$$T_c\dot{\theta} + \theta = \theta_a + T_c\omega$$

- 이산시간계로 변환하기위하여
  - 샘플시간 $\Delta t$ 에 대하여 Euler 근사식 적용

**오일러 근사화**

$$\dot{x}(t) \cong \frac{x(k) - x(k-1)}{\Delta t}$$

$$\dot{\theta}(t) \cong \frac{\theta(k) - \theta(k-1)}{\Delta t}$$

여기서 $t = kt, \quad k = 0,1,2 \cdots$

# 이산시간 상보필터 유도

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

- 오일러 근사화 적용 하면

$$T_c \frac{\theta(k)-\theta(k-1)}{\Delta t} + \theta(k) = \theta_a(k) + T_c\omega(k)$$

- 이를 정리하면

$$\theta(k) = \frac{T_c}{T_c+\Delta t}\theta(k-1) + \frac{\Delta t}{T_c+\Delta t}\{\theta_a(k) + T_c\omega(k)\}$$

- $\boldsymbol{\alpha} \triangleq T_c/(T_c+\Delta t)$ 라고 정의하면

$$\frac{\Delta t}{T_c+\Delta t} = (1-\alpha)$$

# 이산시간 상보필터 유도

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

- 이산시간계 상보 필터식

$$\boldsymbol{\theta(k) = \alpha(\theta(k-1) + \Delta t\omega(k)) + (1-\alpha)\theta_a(k)}$$

new Angle Last Angle From Gyro From Accelerometer

여기서, $\alpha \ = T_c/(T_c + \Delta t)$

- $\alpha$의 성질
  - $0 \leq \alpha \leq 1$ 임
  - $\alpha = 1$
    - $\boldsymbol{\theta(k) = \theta(k-1) + \Delta t\omega(k)}$ ➔ $\boldsymbol{\omega}$ 의 적분식
  - $\alpha = 0$
    - $\boldsymbol{\theta(k) = \theta_a(k)}$ ➔ 가속도 센서로부터 구한 $\boldsymbol{\theta_a}$ 그대로

# 이산시간 상보필터 설계 절차



## ■ 필터 설계 절차

- 샘플간격 $\Delta t$ 와 컷오프 주파수 $f_c$ [Hz]가 주어졌다고 하면
- 다음 식에서 시상수 $T_c$를 구함

$$T_c = \frac{1}{\omega_c} = \frac{1}{2\pi f_c}$$

- 그러면 계수 $\alpha$ 를 구할 수 있음

$$\alpha = \frac{T_c}{T_c + \Delta t}$$

# 상보필터 설계 예제

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

## ■ 예제:

1) 상보필터의 cutoff 주파수 $f_c$가 1.0 Hz , 샘플시간 $\Delta t$ 가 10ms 인 경우

$$\omega_c = 2\pi \text{ [rad/sec]},\ T_c = \frac{1}{\omega_c} = 0.15865,$$

$$\alpha = T_c/(T_c + \Delta t) = \frac{0.15865}{0.15865+0.01} = 0.94$$

$$\theta(k) = 0.94\{\theta(k-1) + 0.01\omega(k)\} + 0.06\theta_a(k)$$

2) 상보필터의 cutoff 주파수 $f_c$가 5.0 Hz , 샘플시간 $\Delta t$ 가 0.5ms 인 경우

$$T_c = \frac{1}{\omega_c} = \frac{1}{2\pi f_c} = 0.0318,$$

$$\alpha = T_c/(T_c + \Delta t) = \frac{0.0318}{0.0318+0.04} = 0.86$$

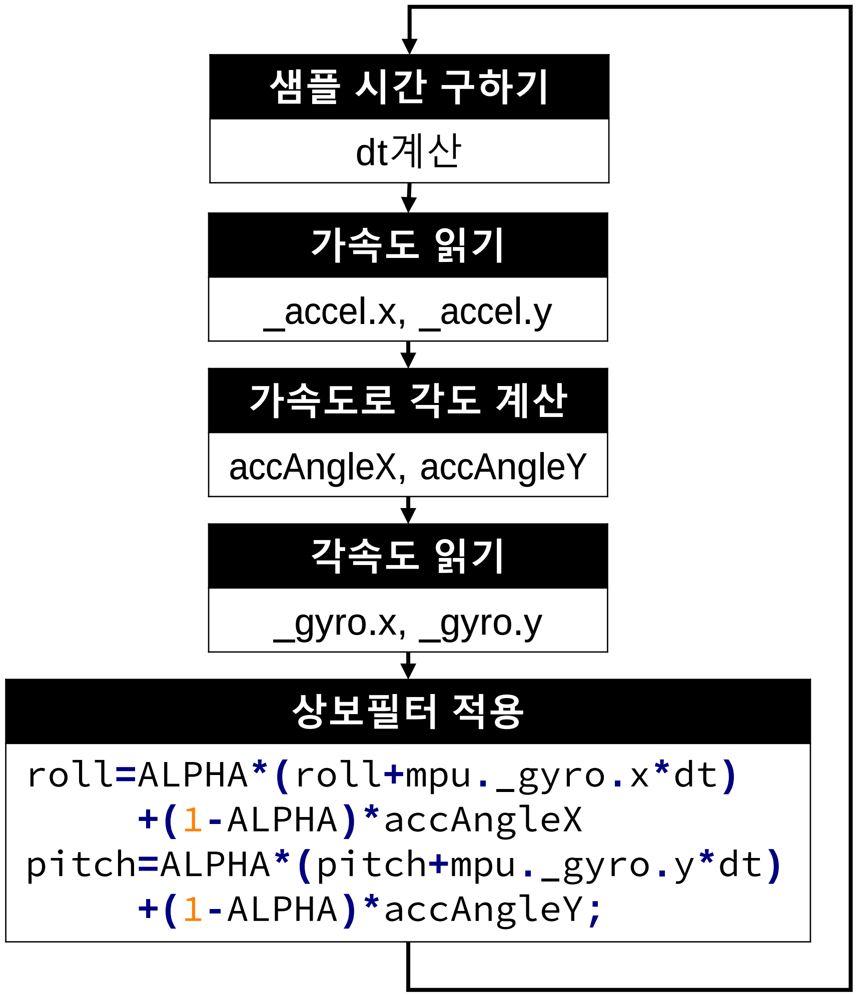
$$\theta(k) = 0.86\{\theta(k-1) + 0.005\omega(k)\} + 0.14\theta_a(k)$$


# Programming the Filter

# 알고리즘 개요

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

- **주요 절차**:
  - 샘플 시간 dt를 구함
  - 가속도값 읽기
  - 가속도로부터 각도 계산
  - 각속도 읽기
  - 상보필터식 적용



# dt 계산



## 샘플 시간 $dt$를 계산

- Gyro 센서가 적분을 하기 때문에 샘플 간격 $dt$ 가 정확해야 함.
- micros() 함수를 이용하여 μs 단위의 $dt$를 계산

```
void calcDT() {
  uint32_t newTime = micros();
  dt = (newTime - prevTime)/1000000.0;
  prevTime = newTime;
}
```

# 가속도로 부터 각도계산

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

## 가속도로 부터 각도 구하기

- 앞서 구한 식을 이용

$$\phi = tan^{-1}(\frac{a_y}{a_z})$$

$$\theta = tan^{-1}\left(\frac{-a_x}{a_y \sin\phi + a_z \cos\phi}\right) = tan^{-1}(\frac{-a_x}{\sqrt{{a_y}^2+{a_z}^2}})$$

- 코드

```
float accAngleX=-atan2(mpu._accel.y,-mpu._accel.z);
float srtAccXY=sqrt(mpu._accel.y*mpu._accel.y
                    +mpu._accel.z*mpu._accel.z);
float accAngleY=atan2(mpu._accel.x,srtAccXY);
```

# 각속도로 부터 각도계산

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

## 각속도로 부터  각도구하기

- 비교를 위하여 추가  Gyro로 부터의 각속도를 이용하여 각도계산
- 단순 적분 계산

```
gyAngleX=gyAngleX+_gyro.x*dt;
gyAngleY=gyAngleY+_gyro.y*dt;
```

## 상보필터를 이용한 센서융합

```
roll=ALPHA* (roll+mpu._gyro.x*dt)+(1-ALPHA)*accAngleX;
pitch=ALPHA* (pitch+mpu._gyro.y*dt)+(1-ALPHA)*accAngleY;
```

# Complementary filter

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

## Programming:

test_MPU6K_fusion_class.ino

```
#include <SPI.h>
#include "MPU6k.h"
#define INTRRUPT_PLOT
#define BARO_CS_PIN 40
#define RadToDeg (180/PI)
#define ALPHA 0.96
extern MPU6000 mpu;
float roll=0, pitch=0, yaw=0,dt;
float gyAngleX=0, gyAngleY=0, gyAngleZ=0;
uint32_t prevTime=0;
void setup() {
  Serial.begin(115200);
  pinMode(BARO_CS_PIN, OUTPUT);  pinMode(12, OUTPUT);
  pinMode(MPU6K_CS_PIN, OUTPUT); digitalWrite(MPU6K_CS_PIN, HIGH);
  digitalWrite(BARO_CS_PIN, HIGH);    //Diselect Barometer
  SPI.begin();
  SPI.beginTransaction(SPISettings(8000000, MSBFIRST, SPI_MODE0));
  delay(100);
  mpu.configureMPU6000();  // configure chip
}
```

# Complementary filter

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

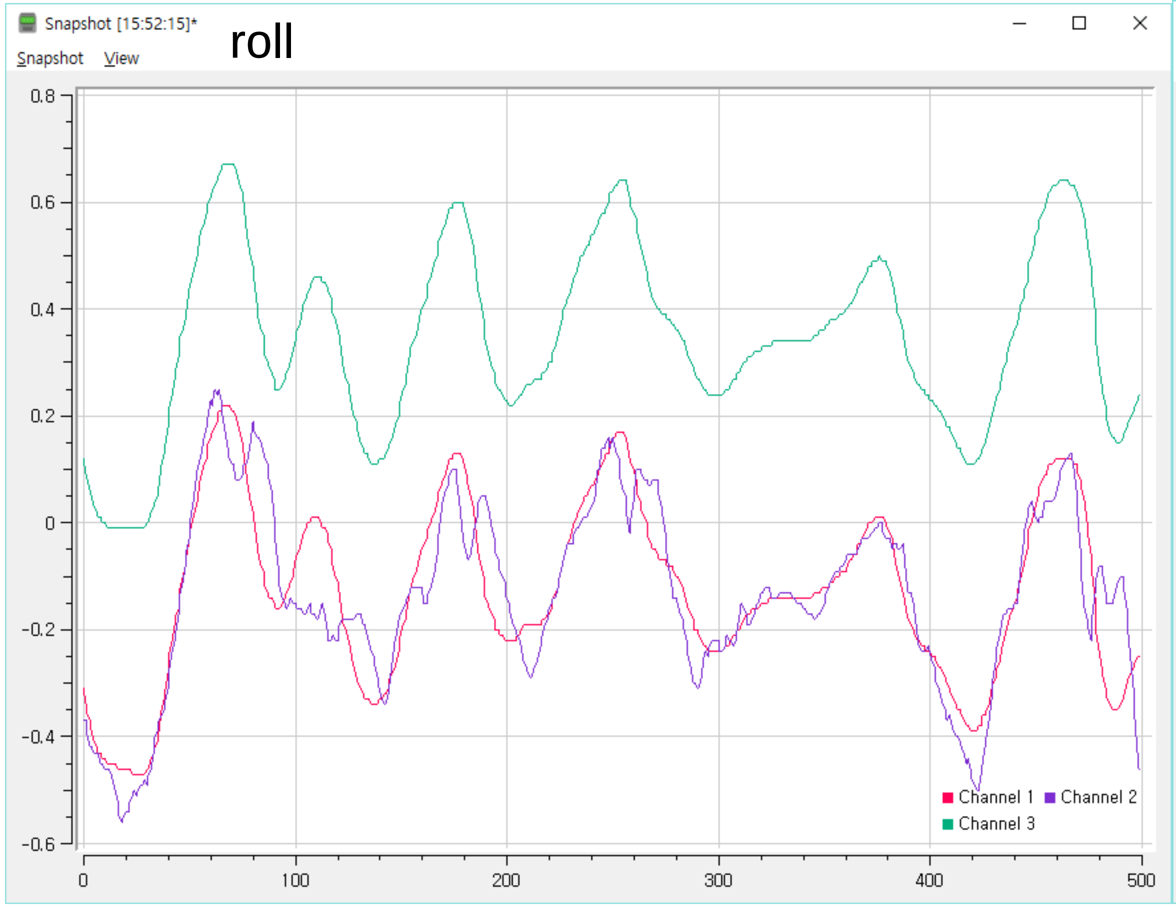
## Programming:

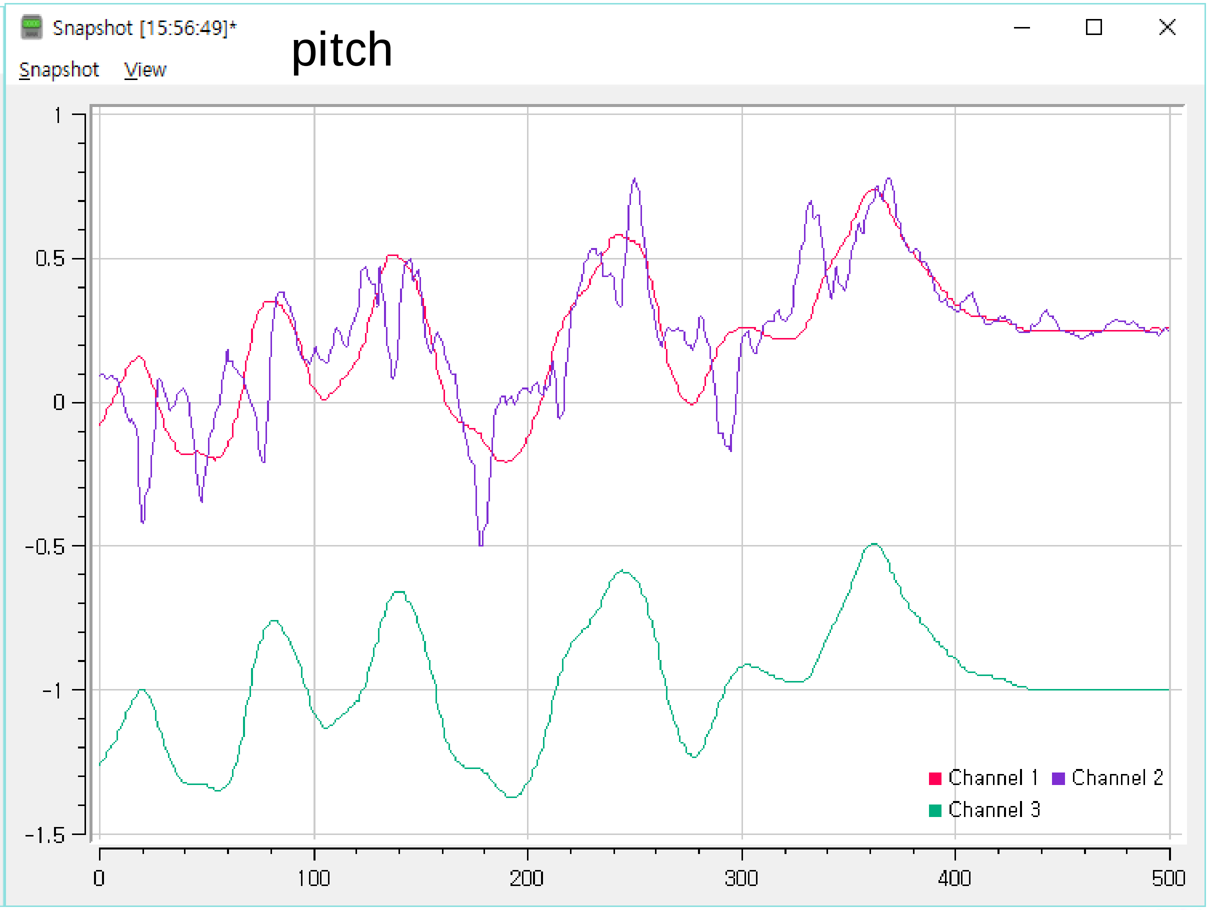
```
void loop() {
  calcDT();
  mpu.updateData( );
  float accAngleX=-atan2(mpu._accel.y,-mpu._accel.z);
  float srtAccXY=sqrt(mpu._accel.y*mpu._accel.y+mpu._accel.z*mpu._accel.z);
  float accAngleY=atan2(mpu._accel.x,srtAccXY);
  roll=ALPHA* (roll+mpu._gyro.x*dt)+(1-ALPHA)*accAngleX;
  pitch=ALPHA* (pitch+mpu._gyro.y*dt)+(1-ALPHA)*accAngleY;
  gyAngleX=gyAngleX+mpu._gyro.x*dt;
  gyAngleY=gyAngleY+mpu._gyro.y*dt;
  gyAngleZ=gyAngleZ+mpu._gyro.z*dt;
  Serial.print(roll*RadToDeg); Serial.print(",");
  Serial.print(pitch*RadToDeg); //.print(",");
  Serial.print("\n");
  delay(0);
}
void calcDT() {
  uint32_t newTime = micros();
  dt = (newTime - prevTime)/1000000.0;
  prevTime = newTime;
}
```

# Filtering Result

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

- ch1: 상보필터, ch2: 가속도 이용, ch3: 각속도 적분

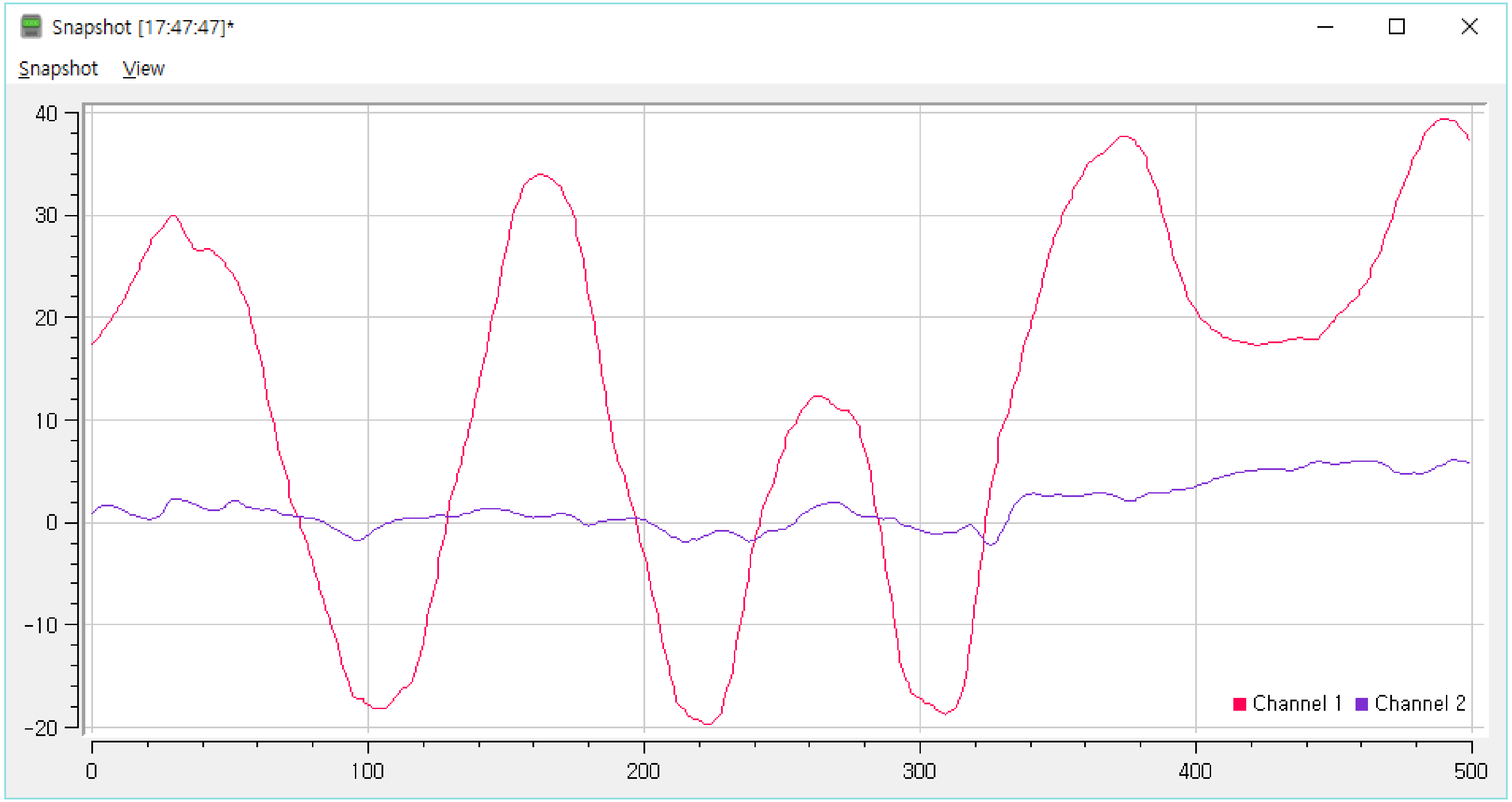


# Filtering Result

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

- ch1: roll, ch2: pitch

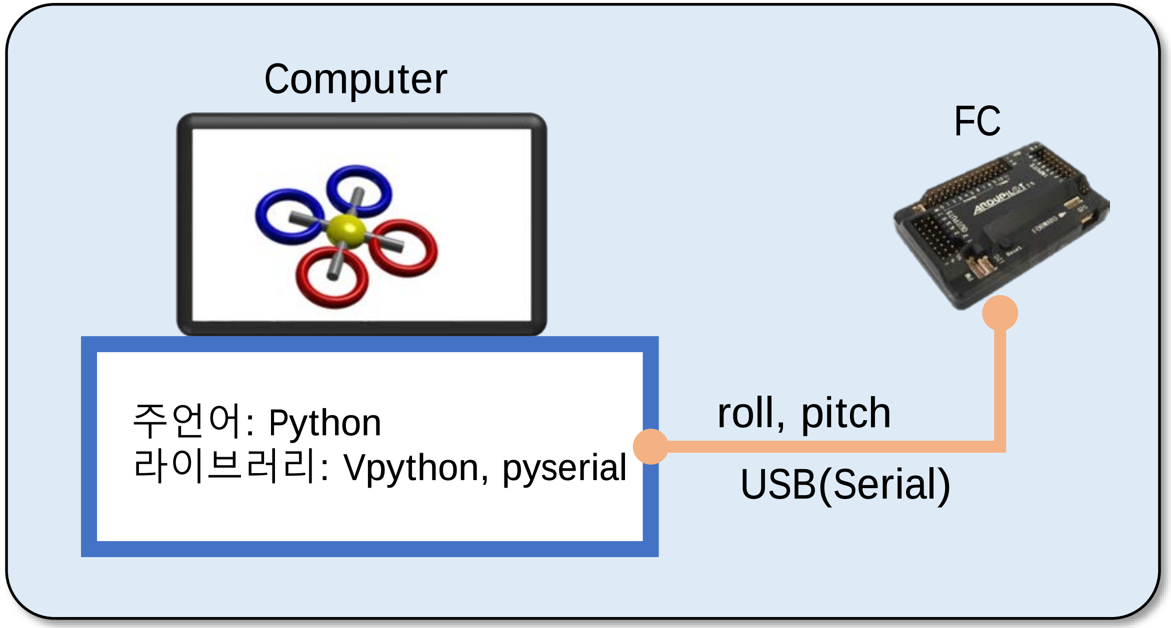


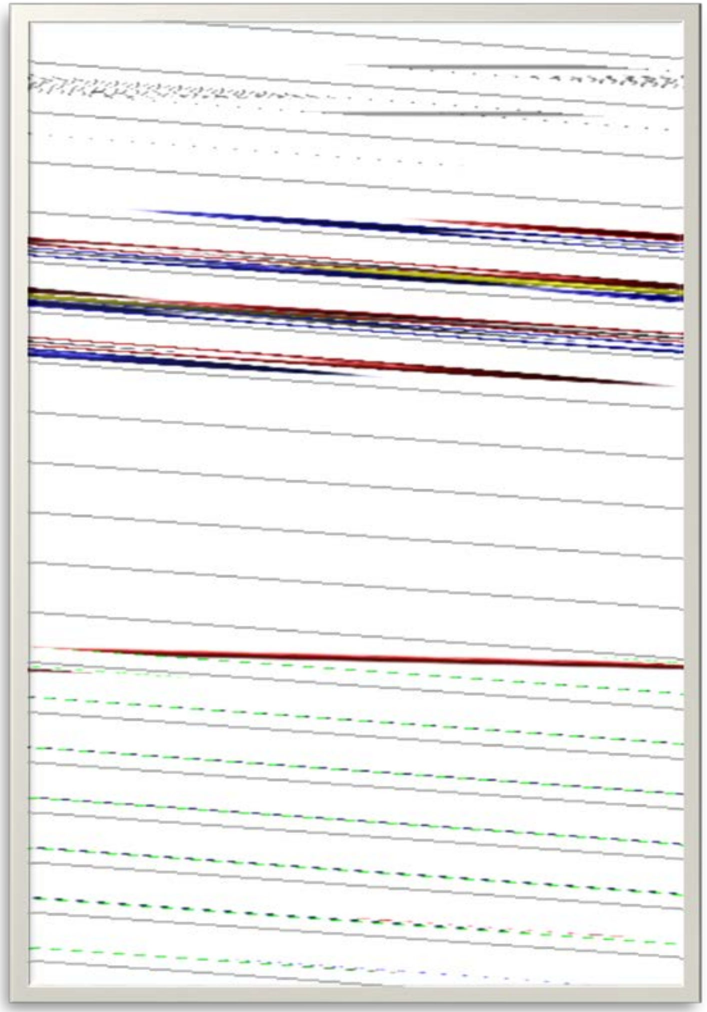
# 상보필터 프로그램 결과

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

## 검증을 위하여 시뮬레이터 구현

- python + vpython (3D graphics)
- threading: 통신과 그래픽 동시에 실행
- pyserial: 직렬통신



# **THANK** YOU

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