Filter Study for SFAC Development, Quad-copter Safety Improvement System

Hyeon-Woo Lim[1], Min-Seok Jie[2], Won-Hyuck Choi[3]*

*[1]Department of Aeronautical Systems Engineering Hanseo University
236-49, Gomseom-ro, Nam-myeon, Taean-gun,
32158 Chungcheongnam-do, Republic of Korea
[2,3]Department of Avionics Engineering Hanseo University
236-49, Gomseom-ro, Nam-myeon, Taean-gun,
32158 Chungcheongnam-do, Republic of Korea
[1]gusdnc05@naver.com, {choiwh, jiems}@hanseo.ac.kr*

Abstract

Recently, as the use of quad-copter increases, accidents caused by accidents and non-arrival are frequent. In this paper, we use the value of Kalman filter which is selected first to reinforce the safety of the quadcopter using integrated sensor. The value of the selected Kalman filter is verified by comparing the simulation result value applied in MATLAB with the experimental data value applied in Arduino. In the future, it is aimed to develop Safty Flight Assurance Control (SFAC) safety flight reinforcement system which selects the other type of filter and compares with Kalman filter and stabilizes landing and level maintenance at a certain height. This paper will contribute to the research that needs stabilization of quadrupole for other purposes in the future.

Keywords: Quadcopter, SFAC, Kalman filter, Attitude

1. Introduction

The Quad-copter is a flying craft that allows people to perform their assigned tasks without boarding. The Quad Copter is a small unmanned aerial vehicle capable of vertical takeoff and landing and hovering. It has the advantage of being able to take off and land anywhere regardless of the location by obtaining the propulsion force from the four rotary motors on the fuselage.

Quad Copter is used in various fields such as real-time surveillance, marine structure, broadcasting, and so on to perform rescue work in the event of a disaster such as forest fire. In addition to the rapidly increasing use value and interest, Quadcopter's technology has a position, posture, and altitude control, which plays an important role in enabling anyone to fly safely and easily.

In this paper, we investigate posture control using integrated sensor. In this paper, we select and apply a filter which can reduce the noise of data self-noise phenomenon when measuring the value of sensor. Compare with actual filter. In the future, it is aimed to develop Safty Flight Assurance Control (SFAC) safety flight reinforcement system that can maintain horizontal level at a certain height after comparing with Kalman filter using other type of filter[1].

---

*Corresponding author: WonHyuck Choi,
choiwh@hanseo.ac.kr*

2. Related Research

2.1. Quad-Copter Works

The Quad-copter is a helicopter with four rotors as shown in Fig 1, with two rotating in a clockwise direction and two rotating in a counterclockwise direction. If the speed of the whole rotor is changed, it can cause the up and down of the quad-copter. If the front and rear blade speeds are different, the Pitch moment is generated on the X-axis direction. If the speed of the left and right blades is different, it is possible to cause a Yaw moment by changing the roll moment and the speed of the left and right front and rear pairs at the same time[3][4].
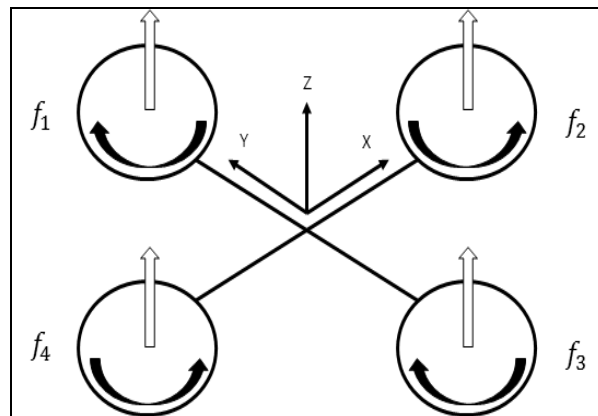
Fig 1. Principle of Quadcopter

In other words, the quad-copter has the advantage that it is easy to control compared to the existing helicopter because the operation principle is simple. As shown in Fig. 2, the inertial measurement sensor (IMU) required for the control includes an accelerometer and a gyroscope in the form of a MEMS sensor. It consists of 3 axes of accelerometer, 3 axes of each speedometer, and 6 axes in total
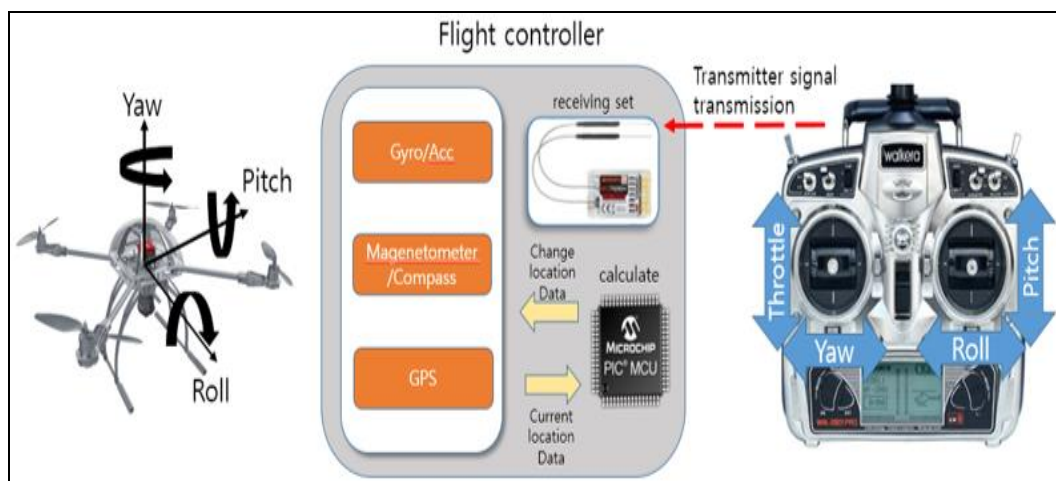
Figure 2. How Quad Copter Works

2.2 Modeling of Quad-Copter

2.2.1 Coordinate Transformation of Quad-copter

A rotation matrix for converting the coordinate system for the gas in the inertial coordinate system, the rotation matrix R, are defined as the following <Equation 1>.

$$R_x(\text{roll}(\phi)) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$
$$R_y(\text{pitch}(\theta)) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\theta \end{bmatrix}$$
$$R_z(\text{yaw}(\psi)) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(1)

The transformation matrix for transforming these equations from the Earth to the body coordinate system can be expressed as <Equation 2>.

$$\begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}$$

(2)

2.2.1 Dynamic Modeling

The final six-degree-of-freedom equation of motion is shown in Equation 3. It is assumed that the aircraft is symmetric about the center of the fuselage, and that the center of the fuselage coincides with the center of gravity of the flight. m is the body weight, I_{xx}, I_{yy}, I_{zz} is the mass moment of inertia of the body, J_r is the moment of inertia of the rotor, and Ω is the rotor rotational angular velocity[2].

$$m\ddot{x} = (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)u_1$$
$$m\ddot{y} = (\cos\phi\sin\theta\cos\psi - \sin\phi\sin\psi)u_2$$
$$m\ddot{z} = (\cos\phi\cos\theta)u_1 - g$$
$$I_{xx}\ddot{\phi} = \dot{\theta}\dot{\psi}(I_{yy} - I_{zz}) - J_r\dot{\theta}\Omega_r + u_2$$
$$I_{yy}\ddot{\theta} = \dot{\psi}\dot{\phi}(I_{zz} - I_{xx}) - J_r\dot{\phi}\Omega_r + u_3$$
$$I_{zz}\ddot{\psi} = \dot{\phi}\dot{\theta}(I_{xx} - I_{yy}) + u_4$$

(3)

$$\ddot{\Phi} = l(-F_1 - F_2 + F_3 + F_4)/J_1$$
$$\ddot{\theta} = l(-F_1 + F_2 + F_3 - F_4)/J_2$$
$$\ddot{\Psi} = (M_1 - M_2 + M_3 - M_4)/J_3$$

(4)

2.3 Integrated Sensor (Acceleration and gyroscope)

The accelerometer is a sensor that measures the acceleration of the current gravity in the X, Y, and Z axes. It is sensitive to vibration and shock due to its characteristics, but it is resistant to errors. Each speedometer is a sensor that measures changes in angular velocity in the X, Y, and Z axes. Each loop has its disadvantage that the angle is calculated through the integral and the error is increased and the error is accumulated at the time of integration. In this paper, pitch and roll are output by using integrated sensor

of acceleration and gyro sensor. Values derived from the integrated sensor are output through a filter.

The MPU-6050 is a sensor module that combines a 3-axis gyro sensor and a 3-axis acceleration sensor. It can use the AVR or Arduino's I2C bus interface to perform complex calculations of sensor values with the DMP function.
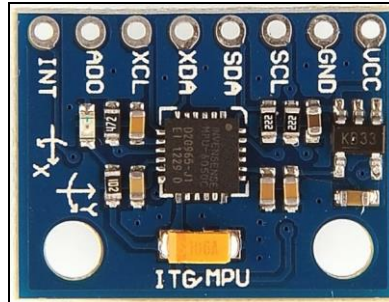
Figure 3. Integrated Sensor (MPU6050)

2.4 Kalman Filter

The Kalman filter is a continuous discrete Kalman filter cycle as shown in Figure 4 below. The time update predicts the current state estimate in advance. The measurement update adjusts the estimate to the actual measurement at that time[5][6][7].
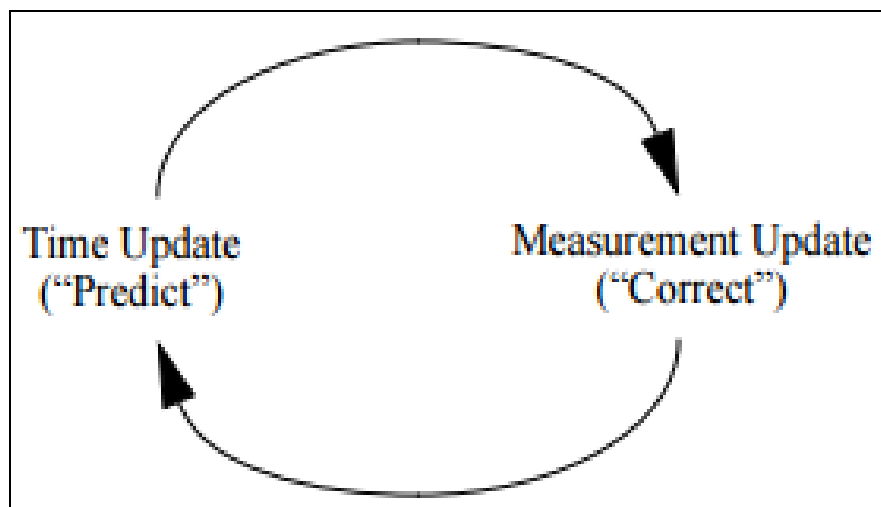
Figure 4. Kalman Filter Algorithm

The Kalman filter is a recursive filter that tracks the state of a linear dynamics system containing noise and iteratively compares estimates that minimize the error from inaccurate measurements. Currently, it is used in various fields such as robotics and radar. Specific equations for time and measurement updates are shown in Table 1 and Table 2 below.

Table 1. Discrete Kalman Filter Time Update Equations

$$\hat{k}^-_{k+1} = A_k \hat{x}_k + Bu_k \qquad (5)$$

$$P^-_{k+1} = A_k P_k A_k^T + Q_k \qquad (6)$$

Table 2. Discrete Kalman Filter Measurement Update equations

$$K_k = P^-_k H_k^T (H_k P^-_k H_k^T + R_k)^{-1} \qquad (7)$$

$$\hat{x}_k = \hat{x}^-_k + K(z_k - H_k \hat{x}^-_k) \qquad (8)$$

$$P_k = (I - K_k H_k) P^-_k \qquad (9)$$

The first task during the measurement update is to compute the Kalman gain. The next step is to actually measure the process to obtain, and then to generate an a posteriori state estimate by incorporating the measurement as in (8). Again (8) is simply repeated for completeness. The final step is to obtain an a posteriori error covariance estimate via (9).
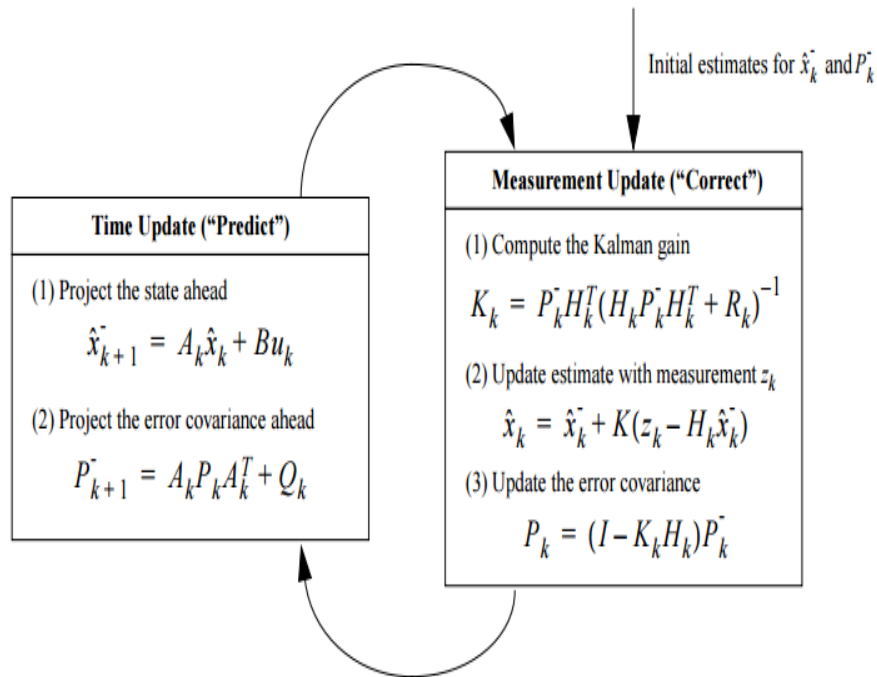
Figure 5. Time & Measurement Update

A complete picture of the operation of the Kalman filter, combining the high-level diagram of Figure 1-1 with the equations from Table 1 and Table 2

3. Safety Flight Assurance Control (SFAC)

The SFAC system in this paper is a system made by reinforcing the posture stir which is an important part in the quadcopter. In order to reduce safety accidents during landing or flying, if a value less than a certain height is input from the object or the ground, the motor is turned to the same hovering state. The SFAC system uses a variety of sensors

and shields to reduce safety accidents and make landing a little more comfortable with attitude control to make the drones' safety and landing instability more prolonged.

3.1 Structure of the SFAC

In this study, we use XBee, a 2.4GHz wireless communication module, for transmission and reception. To apply SFAC to the quad copter, we use a gyro and accelerometer integrated sensor which is verified through simulation of attitude control. The overall conceptual diagram is shown in Figure 6 below.
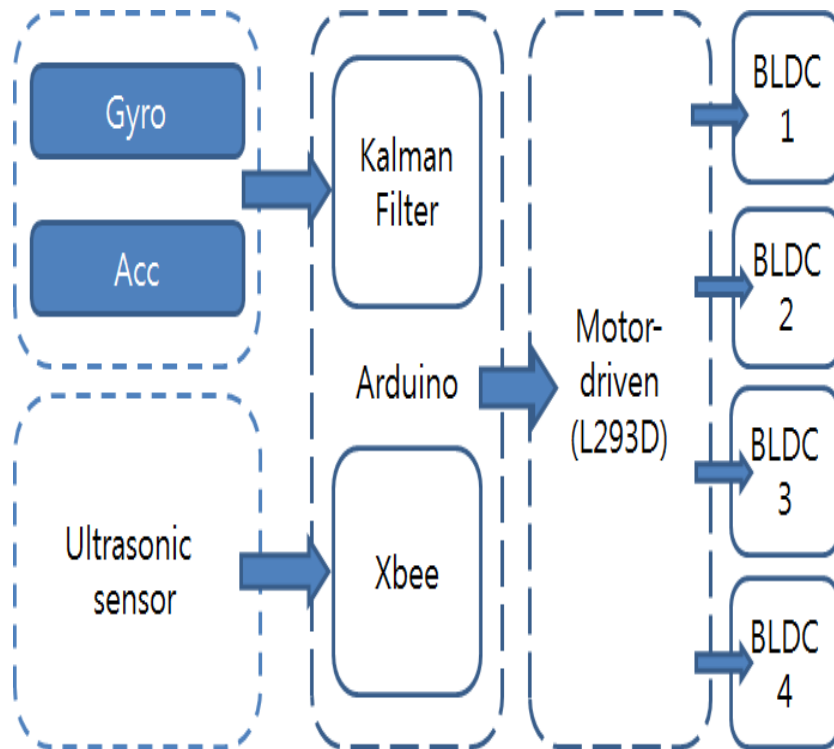
Figure 6. General Diagram of the SFAC System

Quad Copter Keeps level when approaching obstacle or ground when reducing altitude during flight. Complementing the need for time and precision maneuvering during takeoff and landing. Helps take off and land safely from a constant height to a horizontal posture control.

The SFAC system uses a gyroscope and an acceleration sensor to measure the tilt and an altitude sensor to measure the height of the quadrupole. These sensor values pass through the Kalman filter through Arduino and control the motor.

3.2 Mainboard

The mainboard used a generic Arduino Uno. On the main board, integrated sensor and motor control will be done. The L293D motor driver shield was used to control the motor.
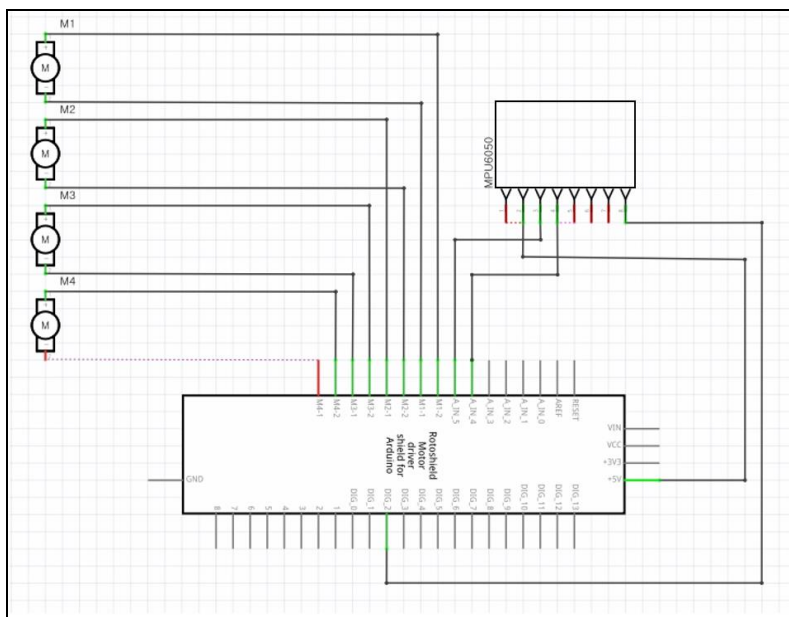
3.2.1 Motor Control

Figure 7. Motor Control Circuit Diagram

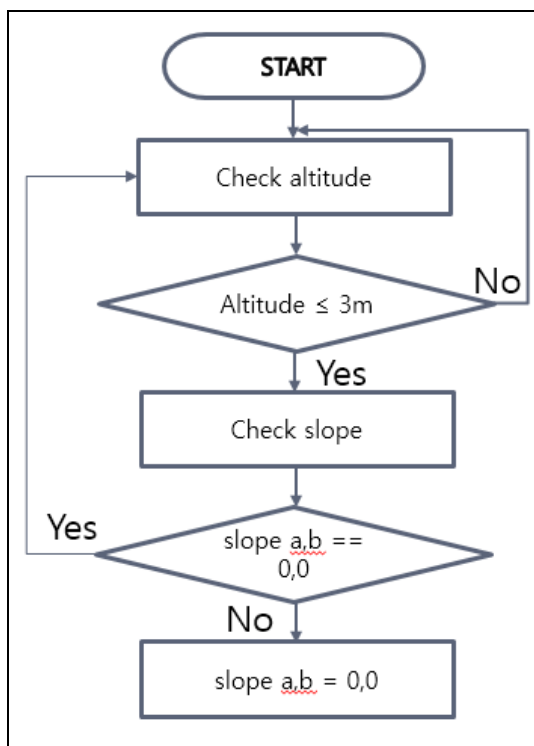### 3.2 Safety Flight Assurance Control Algorithm

Figure 8. SFAC Algorithm

Figure 7 below shows the SFAC algorithm. When the flight starts, check the altitude to see if the ground or obstacle is within 3m. Next, when the ground or obstacle is detected at 3 m, the inclination check is performed. Finally, if the slope is not horizontal, it will

change horizontally. If there is no ground, obstacle or inclination within 3m, it will go back to the beginning and repeat it.

### 3.3 Safety Flight Assurance Control Experiment

To use the SFAC system, it is necessary to check whether it is stable. Sensor values that do not pass through the filter have data self noise. To eliminate this noise, a Kalman filter was used for the actual sensor value, and the value simulated with the Mattel was compared with the output value using the filter on the motherboard, Arduino. In this paper, we use a Kalman filter chosen to stabilize the quadrupole. First, we output the data of the quad-copter sensor through Arduino. The comparison data uses actual measured values. The actual measured values are compared with the simulated values using MATLAB and the Kalman filter used in Arduino. Figure 8 below shows the basic data output from the Arduino and Kalman filter values.

|	Raw data		Kalman data								
2	roll	pitch	roll	pitch		4222	184.9	188.96		185.13	189.22
3	183.83	181.27	183.66	180.8		4223	184.66	189.79		185.12	189.24
4	183.43	182.21	183.65	180.77		4224	183.99	189.65		185.09	189.25
5	178.23	184.77	178.13	185.26		4225	183.67	190.33		185.06	189.27
6	178.54	182.68	178.18	185.11		4226	183.74	190.05		185.03	189.28
7	179.03	180.39	178.22	184.95		4227	183.88	190.51		185.01	189.31
8	178.96	179.76	178.26	184.8		4228	183.56	189.83		184.99	189.32
9	179.65	180.15	178.28	184.67		4229	183.41	189.83		184.96	189.32
10	180.32	180.28	178.3	184.56		4230	183.48	189.96		184.94	189.33
11	179.78	180.45	178.31	184.48		4231	183.91	190		184.92	189.34
12	179.13	180.53	178.32	184.4		4232	183.74	190.12		184.91	189.35
13	179.13	181.66	178.33	184.34		4233	184.34	189.5		184.9	189.35
14	178.89	182.1	178.34	184.28		4234	184.34	189.34		184.9	189.35
15	178.75	181.87	178.35	184.23		4235	184.21	189.8		184.9	189.36
16	179.83	181.73	178.35	184.19		4236	183.41	189.4		184.89	189.36
17	180.88	182.87	178.34	184.16		4237	183.72	189.05		184.88	189.35
18	179.75	183.22	178.33	184.14		4238	183.63	188.49		184.86	189.34
19	179.64	182.83	178.33	184.13		4239	184.31	189.26		184.86	189.34
20	179.78	182.71	178.33	184.13		4240	184.27	189.27		184.86	189.34
21	179.89	182.24	178.34	184.13		4241	184.01	189.27		184.85	189.35
						4242	184.34	188.65		184.84	189.34
						4243	185.12	187.89		184.85	189.32
						4244	186.35	187.69		184.89	189.29
						4245	185.86	187.69		184.91	189.27

Figure 9. Integrated Sensor Experiment Data

Figure 9 is a graph of Kalman filter applied to the mantle. In this graph, the average error is obtained from the data in the interval of 40 to 60 seconds in the actual data. In the case of PITCH, the value is 0.610, and the average error of the ROLL is 0.253.
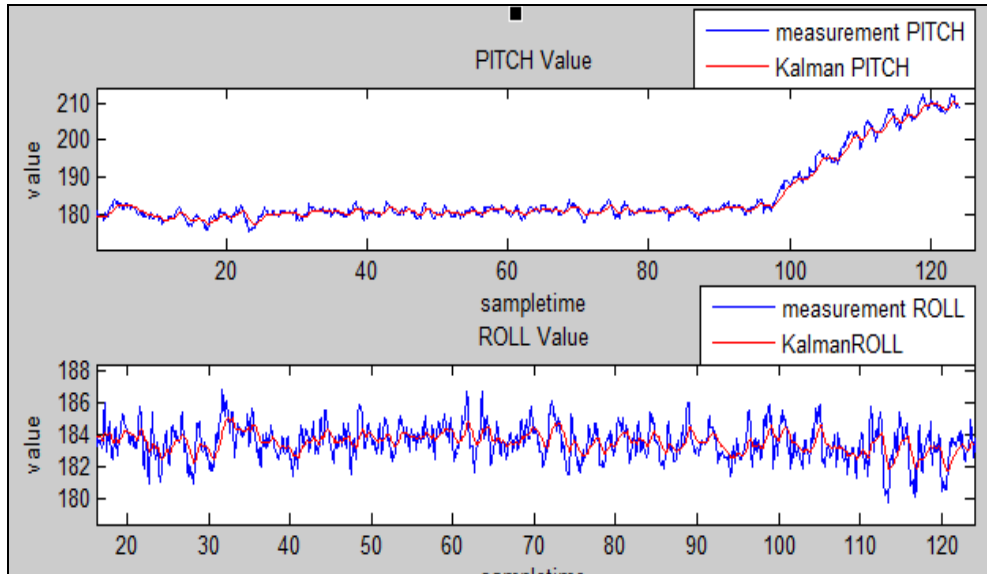
Figure 10. Values Applying Kalman in Matlab

Figure 10 below shows the results of Kalman filtering of the values measured by the MPU6050 through the Arduino. As shown in the graph in Fig. 9, the average error is obtained from 40 to 60 seconds. In the case of PITCH, it was 3.900. The average error of the ROLL is -0.353, which is more stable than the Kalman filter, although it does not appear to be ideal.
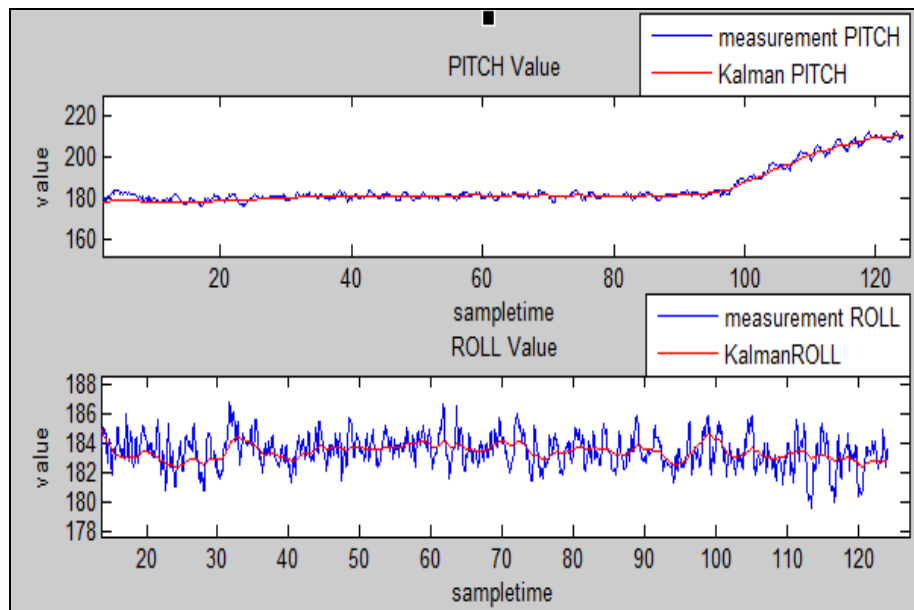
Figure 11. Values Applying Kalman in Arduino

4. Conclusion

This paper aims to compare the stability of Kalman filter in Matlab and Arduino to stabilize posture control and to develop SFAC safety flight reinforcement system. Figure 4 and Figure 5 show little difference, but Figure 5 shows a little more stabilization. In the future, we will apply Kalman filter by fusing various sensors and study more stabilized research than present results. This study suggests that the quad - copter used for other

purposes will be useful for the application and development if the posture stabilization is needed.

References

[1]   D. H. Yu, J. H. Seung, J. h. Park, D. J. Lee and K. T. Chong, "Attitude Control for Quadcopter based on Sensor Fusion using Kalman Filter", THE INSTITUTE OF ELECTRONICS ENGINEERS OF KOREA, Vol.37, no.1, (2014), pp.1162-1163.

[2]   T. S Kang, K. J. Yoon, T. H. Ha and G. G. Lee, "H-infinity Control System Design for a Quad-rotor", Institute of Control, Robotics and Systems, vol.21, no.1, (2015), pp. 14-20.

[3]   H. Kim, H. S. Jeong, K. T. Chong and D. J. Lee, "Dynamic Modeling and Control Techniques for Multi-Rotor Flying Robots", The Korean Society of Mechanical Engineers, vol.38, no.2, (2014), pp.137-148

[4]   P. Castillo, A. Dzul, and R. Lozano, "Real-time stabilization and tracking of a four-rotor mini rotorcraft control systems echnology", IEEE Transactions on Control Systems Technology, vol. 12, no. 4, (2004), pp. 510-516

[5]   D.-W. Gu, P. Hr. Petkov, and M. M. Konstantinov, "Robust Control Design with MATLAB", A-jin Publishing Company, 2008

[6]   D. H. Yu, D. Y. Lim, N. O. Sel, J. H. Park and K. T. Chong, "Study of Sensor Fusion for Attitude Control of a Quad-roto", Institute of Control, Robotics and Systems, vol.21, no.5, (2015), pp. 453-458

[7]   Chul Woo Kang, Young Min Yoo, Chan Gook Park, "Performance Improvement of Attitude Estimation Using Modified Euler Angle Based Kalman Filter", Institute of Control, Robotics and Systems, vol.14, no.9, (2008), pp.881-885

Authors
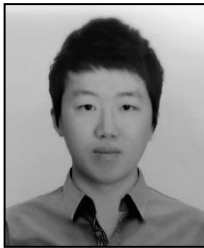
Hyeon-Woo Lim, graduate of the Hanseo University avionics bachelor's and a master's degree and Aviation Systems Engineering in 2015.

Min-Seok Jie, graduated from the Ph.d degree in the department of avionics Engineering from Korea Aerospace University in 2006. he is Hanseo University avionics Engineering Professor in to the present in 2008.

Won-Hyuck Chio, graduated from the Ph.d degree in the department of avionics Engineering from Korea Aerospace University in 2006. the Doowon Technical University Smart Network Professor From 2008 to 2014. he is Hanseo University avionics Engineering Professor in to the present in 2014.