

技術論文

DOI:http://dx.doi.org/10.5139/JKSAS.2012.40.6.542

복수 무인항공기의 운용을 위한 지상통제 소프트웨어 개발

신윤호*, 조상욱*, 조성범*, 김성환*, 유창경*, 최기영**

Development of Ground Control Software for
Operation of Multiple Unmanned Aerial VehiclesYoon-ho Shin*, Sang-wook Cho*, Sungbeom Jo*, Sung-hwan Kim*, Chang-kyung Lyu* and
Keeyoung Choi**

ABSTRACT

Until recently, most of GCS(Ground Control Software) has been required to visualize attitude, position, status of vehicle and to transmit control and mission commands for a single UAV(Unmanned Aerial Vehicle). However, the GCS needs to expand its functions to handle more complex situations. Simultaneous operation of multiple UAVs is emerging as a new practice. Hence, we set up requirements for operation of multiple UAVs and suggest the architecture of GCS that satisfy the requirements. In this study, we analyze the upper requirements and define the total structure of GCS at first. Then we design the inner structure for requirements in detail. Finally, we verify the functions of GCS on PILS(Processor In the Loop Simulation) System.

초 록

현재까지 대부분의 지상통제 소프트웨어는 단일 무인 항공기에 대한 자세, 위치, 상태의 수신 및 가시화와 항공기의 운용을 위한 조종 명령 및 임무 전송의 기능만 갖추고 있었다. 하지만 단일 무인기의 운용에서 다중 무인기의 운용으로 사용자의 요구가 확장됨에 따라 지상통제 소프트웨어도 다중 무인기의 운용을 위한 기능의 확장이 요구되고 있다. 본 연구에서는 다중 무인항공기의 운용을 위한 요구사항을 제시하고 이를 만족시키기 위한 지상통제 소프트웨어의 구조를 제안하였다. 우선, 제시된 상위 요구 성능을 분석하여 그에 따른 전체 구조를 정의하고, 이를 다시 세부 요구사항으로 설정하여 세부구조를 설계하였다. 마지막으로 PILS 환경에서 구성된 지상통제 소프트웨어의 기능을 검증하였다.

Key Words : Ground Control Software(지상통제 소프트웨어), Multiple Unmanned Aerial Vehicle(복수 무인항공기), PILS(Processor In the Loop Simulation)

1. 서 론

† 2012년 1월 26일 접수 ~ 2012년 5월 22일 심사완료

* 정회원, 인하대학교 항공우주공학과

** 정회원, 인하대학교 항공우주공학과

교신저자, E-mai : kchoi@inha.ac.kr

인천광역시 남구 용현동 인하대학교

유인항공기에서 무인항공기로의 발전이 이루어지며 항공기의 자세, 위치, 상태 정보의 수신 및 조종명령을 전송하기 위한 지상제어 소프트웨어의 개발도 필연적으로 이루어져왔다. 지금까지의

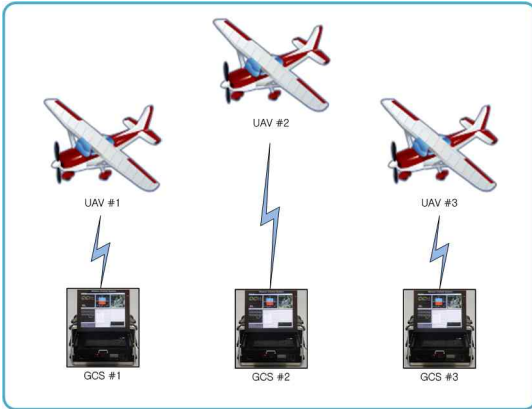


Fig. 1. One-to-One Matching System

무인항공기의 운용은 대부분 단일 무인항공기와 하나의 지상통제 소프트웨어의 일대일 대응 시스템이었다. 하지만 비행제어 컴퓨터의 비약적인 발전 등으로 무인항공기의 사용이 군, 산업에서 광범위하게 적용되기 시작하고, 그 역할이 다양해지면서 사용자의 요구도 단일 무인항공기의 운용에서 복수 무인항공기의 운용으로 확장되었다 [1,2]. 이렇게 사용자의 요구가 확장됨에 따라 지상제어 소프트웨어의 기능 또한 확장되어 적용되어야 한다.

지금까지 사용자가 단순히 지상통제 소프트웨어를 이용하여 단일 무인항공기와 일대일로 데이터를 수신하고 임무를 송신하던 조종사의 개념[3,4]에서 하나의 지상통제 소프트웨어에서 다수의 무인항공기에 대한 모든 정보를 관찰하고, 각각 개별적인 임무 명령을 전달할 수 있는 관리자의 개념으로 확장되어야 한다[1]. 이러한 개념의 확장에는 무인항공기들의 안정성과 시스템 전체의 효율성이 뒷받침되어야 한다.

하지만 기존의 단일 무인항공기의 운용을 위한 구조를 다수 무인항공기의 운용에 적용한다면 항공기의 안정성과 전체 시스템의 효율성에 대한 요구를 만족시킬 수 없다. 다중 무인항공기 운용의 핵심은 다수의 무인항공기가 상호 독립적이 아닌 유기적 관계를 유지하며 임무를 수행하는데 있다. 하지만 Fig. 1과 같은 일대일 대응 시스템에서는 자신이 담당하고 있는 무인 항공기에 대해서만 제한적으로 정보 수신 및 임무 전달이 가능하기 때문에 편대비행과 같은 임무를 계획하거나 임무 수행에 대한 확인이 용이하지 않다.

다음으로 고장발생에 대해 취약한 단점을 가지고 있다. 다수의 무인항공기를 운용하는 중에 특

정 지상통제 소프트웨어나 무선 통신용 모뎀에 고장이 발생했을 경우, 특정 무인항공기는 운용자의 통제권을 이탈하게 된다. 일대일 대응 시스템에서는 이를 대처할 수 있는 방안이 없다.

Fig. 2와 같은 다중구조는 운용하는 모든 항공기에 대해 정보 수신이 가능하다는 장점이 있지만 일대일 대응 시스템과 마찬가지로 고장발생에 취약한 단점이 있다.

따라서 본 연구에서는 Fig. 3과 같이 호스트 장비를 이용하여 보완된 일대일 대응 시스템 구조를 설정하여, 각각의 지상통제 소프트웨어에서 모든 항공기의 정보를 수신할 수 있고 고장발생에 대해서도 대처가 가능하도록 하였다.

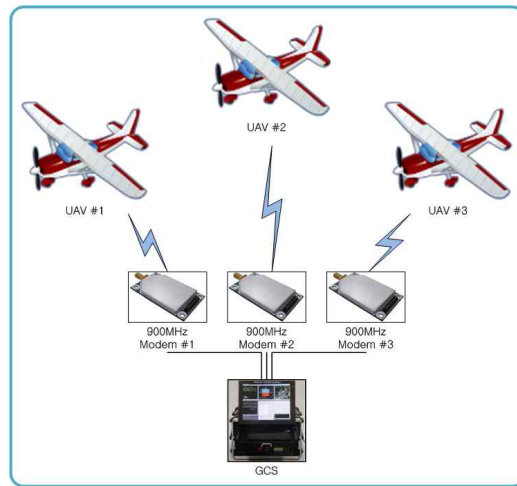


Fig. 2. Multiple System

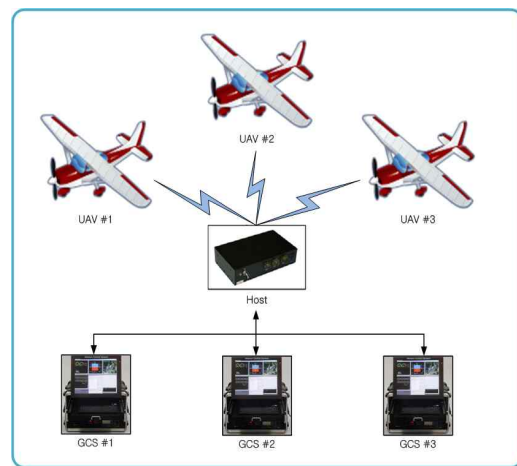


Fig. 3. Complementary One-to-One Matching System

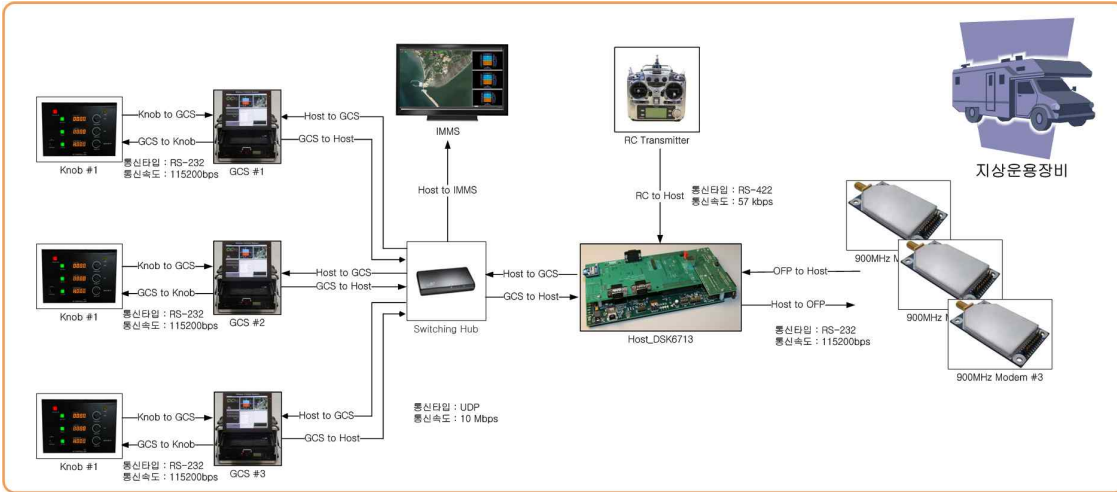


Fig. 4. Composition of Ground Operation Equipment

가장 먼저 요구조건을 확립하였고, 그에 따른 지상통제 소프트웨어의 전체적인 구조를 정의하였다. 다음으로 전체적인 구조에 대해 세부 구조를 설계하고, 보완된 일대일 대응 시스템의 운용 개념에 대해 설명하였다. 마지막으로 이렇게 설계된 지상통제 소프트웨어를 PILS환경에서 검증하였다.

II. 본 론

2.1 요구 조건

무인항공기의 운용이 복수 무인항공기의 운용으로 확장되면서 지상통제 소프트웨어에 요구되는 성능 또한 확장되었다. 이러한 운용 개념의 확장으로 인해 요구되는 지상제어 소프트웨어의 기능은 다음과 같다.

첫 번째로 복수 무인항공기에 대한 데이터 수신 기능이다. 앞서 서론에서 밝힌 바와 같이 무인기를 운용하는 사용자에게 대한 개념이 단순한 조종사의 개념을 넘어 관리자의 개념으로 확장되었다. 이는 기존 단일 무인항공기의 운용에서는 단지 해당 무인항공기의 정보만을 알면 되었지만, 복수 무인항공기의 운용에서는 모든 무인항공기의 정보가 필요하다는 것을 의미한다. 다수의 무인항공기에 대한 정보를 수집하기 위해서는 운용하고 있는 무인항공기 수만큼의 무선 모델 장치가 필요하다. 이를 단순히 지상제어 소프트웨어와 연결을 하는 것은 다수의 지상제어 소프트웨어를 운용하는데 있어 시스템의 관리 및 운

용적 측면에서 비효율적이다. 따라서 본 연구에서는 Fig. 4와 같이 모든 무인항공기의 정보를 수집 및 전달해 주는 호스트 장비를 구성하였다. 이와 같이 호스트 장비를 포함한 구성은 LAN선을 허브에 연결하는 것만으로 모든 무인항공기의 정보를 수신할 수 있으므로, 다수의 지상제어 소프트웨어를 운용하는데 효율적이다. 또한 데이터의 송수신 주기를 정확히 맞춰줄 수 있기 때문에 실시간성이 확보가 된다.

두 번째는 50Hz의 주기를 갖는 명령 전송 기능이다. 명령 전송의 주기는 운용하고 있는 모델의 성능과 무선 통신의 제약, 사람의 반응속도 등을 고려해야 한다. 사람의 시각 반응속도는 30Hz정도[5]이므로 무인항공기로 조종 명령을 전송하는 주기는 최소 30Hz 이상이 되어야 한다. 또한 본 연구에서 사용하고 있는 무선 모델의 성능은 115200bps이고, 송수신할 데이터의 ICD는 200byte 정도로 정의되어 있다. 따라서 이론적으로 약 70Hz의 성능을 낼 수 있다. 여기에 시스템의 안정적 동작 등을 고려한 마진을 적용하여 50Hz의 주기를 설정하였다.

2.2 소프트웨어 전체 구조

소프트웨어의 전체적인 구조는 모듈 단위로 구성하였다. 이러한 모듈 단위의 소프트웨어 구조는 상호 독립적으로 동작을 하기 때문에 소프트웨어의 수정이나 새로운 모듈의 추가가 용이하고 효율성을 높일 수 있다[6].

전체 구조는 Fig. 5와 같이 수신 모듈, 가시화 모듈, 데이터 저장 모듈, 송신 모듈, 임무계획 모듈, 외부 입력 모듈 이렇게 총 6개 모듈로 구성

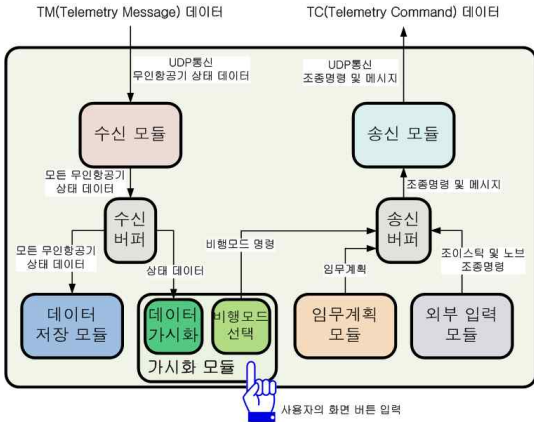


Fig. 5. Total Structure of GCS

되었으며 소프트웨어 동작의 큰 흐름은 다음과 같다. 수신 모듈에서 TM(Telemetry Message) 데이터가 수신되면 이를 모두 수신 버퍼에 저장하고, 데이터 저장 모듈에서는 수신 버퍼에 접근하여 데이터를 저장한다. 가시화 모듈에서는 수신 버퍼에 접근하여 화면을 업데이트하고, 화면상의 버튼으로 사용자로부터 운용할 비행모드 정보를 입력받는다. 외부 조이스틱이나 노브로부터 입력되는 조종 명령은 외부 입력 모듈을 통해 송신 버퍼에 저장되며, 임무계획 또한 임무계획 모듈을 통해 송신 버퍼에 저장된다. 송신 모듈에서는 50Hz의 주기마다 송신 버퍼에 접근하여 가장 최신의 조종 명령이나 임무정보를 무인항공기로 전송한다.

2.3 소프트웨어 세부 구조

2.3.1 수신 모듈

수신 모듈의 기본적인 기능과 모듈의 흐름은 Fig. 6에 순서도로 표현되어있다.

수신 모듈의 기능은 데이터의 수신, 무결성 검사와 데이터 분석으로 나눌 수 있다. 이 모듈에서는 UDP 통신으로 수신되는 모든 무인항공기의 정보를 수신하고, 수신된 데이터를 분석하여 수신 버퍼에 모든 정보를 저장한다.

2.3.2 가시화 모듈

가시화 모듈은 Fig. 7과 같이 주기적으로 수신 버퍼에 접근하여 수신된 무인항공기의 정보를 사용자에게 시각적으로 표현하는 기능과 사용자로부터 운용하고자 하는 비행모드를 화면상의 버튼 클릭으로 입력받는 기능을 한다.

다수의 무인기를 운용함에 있어서 화면에 표시되는 정보는 상당히 유동적이어야 한다. 따라서

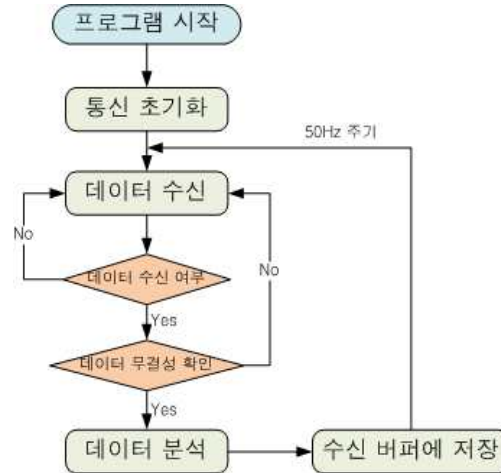


Fig. 6. Process Chart of Receive Module

사용자가 필요한 정보만을 볼 수 있도록 화면의 창 구성을 세세하게 분할한다. 사용자는 자신의 필요에 따라 모든 무인항공기의 상태나 자세만을 볼 수도 있고, 또는 특정 무인항공기의 모든 정보를 볼 수도 있다.

복수 무인항공기의 운용은 앞서 1장에서 밝힌 바와 같이 서로 독립적 임무가 아닌 상호 유기적인 관계를 유지하며 임무를 수행해야 한다. 이러한 임무를 수행하고 사용자가 어떤 비행모드로 운용할 지를 판단하는데 있어서 가장 중요한 정보가 자기 자신과 다른 무인항공기간의 상대적 상태 정보이다. 따라서 지도 창에서는 모든 무인기의 위치, 상대적 거리 등을 모두 표시한다.

2.3.3 데이터 저장 모듈

데이터 저장 모듈은 비행 후 분석을 위해 모든 정보를 저장하는 기능을 한다. 이 모듈은 수신 버퍼가 새롭게 업데이트 되는 즉시 수신 버퍼에 있는 모든 데이터를 파일로 저장한다.

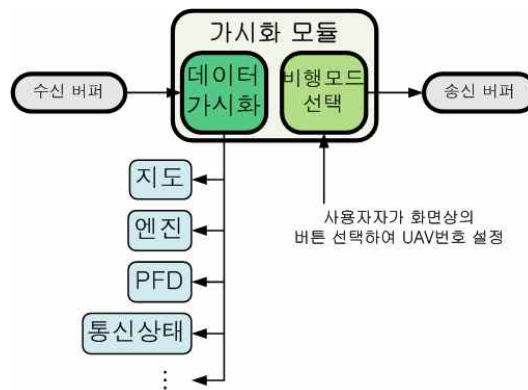


Fig. 7. Visualization Module

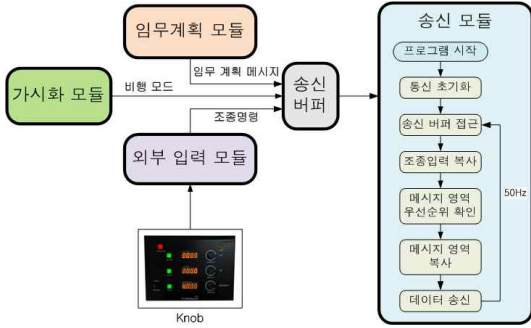


Fig. 8. Transmission Module

2.3.4 외부 입력 모듈

외부 입력 모듈은 외부 조이스틱이나 노브 등을 연결했을 때, 이 정보들을 받아들이는 기능을 한다. 이 모듈은 주기적으로 외부에서 입력되는 정보를 수신하여 송신 버퍼에 저장한다.

2.3.5 임무계획 모듈

임무계획 모듈은 무인항공기의 자동 비행운용에 필요한 임무를 계획하는 기능을 한다. One-Point Navigation, RTB, Waypoint Navigation, 고장 모드 등의 비행 모드나 Waypoint navigation시 비행경로 등을 사용자로부터 입력 받아 송신 버퍼에 저장한다.

2.3.6 송신 모듈

송신 모듈은 사용자의 조종 입력이나 임무계획 등을 UDP 통신으로 전송하는 기능을 하며 Fig. 8과 같은 구조를 가진다. 이 모듈은 앞서 2.1절에서 밝힌 바와 같이 50Hz의 주기로 송신 버퍼에 접근하여 조종 입력과 임무 정보를 송신한다.

2.4 구조 검증

이번 절에서는 앞서 설정한 소프트웨어의 구조를 기반으로 소프트웨어를 구현하고, 이를 Fig. 9와 같은 Pils 환경에서 검증하였다. 호스트는 세 대 분량의 데이터를 취합하여 데이터를 브로드캐스팅하고, 각각의 GCS로부터 전달받은 조종명령을 각각의 FCS로 전달한다.

다음으로 지상 통제 소프트웨어에서 저장된 데이터와 Pils 환경에서 6DOF 시뮬레이션 컴퓨터에서 저장되는 데이터를 비교하였다. 앞서 소프트웨어의 구조에서도 살펴볼 수 있듯이, 구현된 소프트웨어의 수신 모듈, 데이터 저장 모듈, 가시화 모듈은 모두 수신 버퍼를 공유로 사용하는 구조이다. 따라서 이 둘의 데이터를 비교함으로

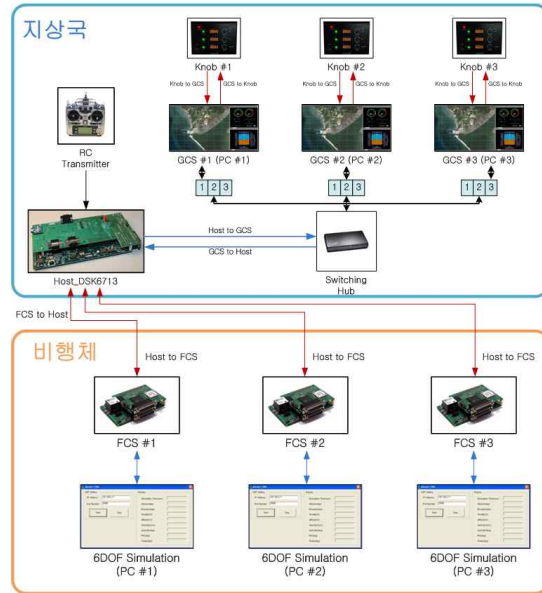


Fig. 9. Pils Environment

써 위의 세 가지 기능을 모두 검증할 수 있다.

마지막으로 Pils 시뮬레이션에서 EP(External Pilot), IP(Internal Pilot), Auto mode 비행을 통해 외부 입력 모듈, 임무 계획 모듈의 검증과 더불어 실제로 운용하는데 있어서의 기능과 성능을 검증하였다.

Table 1을 살펴보면 GCS와 6DOF에서 저장되는 데이터가 같다는 것을 알 수 있다. 다만 저장되는 데이터의 Resolution 때문에 표현되는 자리수만이 다르다. 마지막으로 Fig. 10과 같이 운용상의 검증을 하였다. 소프트웨어는 임무계획이 정상적으로 전송되었을 때만 화면상에 경로점을 표시하도록 되어있다.

호스트 장비의 도입을 통해 각각의 소프트웨어에서 모든 항공기의 데이터를 수신할 수 있도

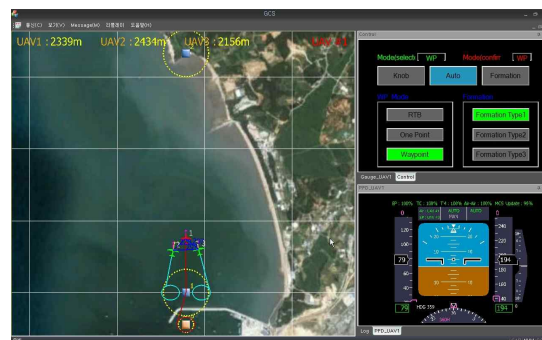


Fig. 10. Formation Flight Simulation

록 하고, 시스템의 실시간성 또한 확보하였다. 또한 소프트웨어의 가시화 모듈에서 모든 항공기의 상태 정보를 세세하게 나누어 표현함으로써 사용자가 자신의 상황에 맞게 임무를 계획할 수 있도록 하였다. 마지막으로 이렇게 완성된 소프트웨어를 PILS환경에서 검증함으로써 본 연구에서 설계한 시스템과 소프트웨어가 요구조건으로 설정한 안정성과 효율성을 모두 만족한다는 것을 알 수 있다.

Table 1. Data Comparing

No.	P(Roll Rate)		Q(Pitch Rate)		R(Yaw Rate)	
	GCS	6DOF	GCS	6DOF	GCS	6DOF
1	0	0.20746	-7	-7.33425	0	-0.01851
2	-18	-18.829	-1	-1.64798	1	1.019239
3	-69	-69.043	10	10.65468	3	3.585965
4	-120	-120.73	14	14.84371	6	6.189681
5	-118	-118.25	13	13.83728	7	7.318806
6	-115	-115.69	11	11.64658	6	6.710047
7	-75	-75.637	0	0.84757	5	5.140645
8	-57	-57.843	-18	-18.892	4	4.09989
9	-67	-67.833	-31	-31.2924	3	3.793494
10	-75	-75.548	-32	-32.9827	3	3.035563

III. 결 론

본 연구에서는 운용하고자하는 모든 항공기의 안정성과 시스템의 효율성을 만족시킬 수 있도록 호스트 장비를 이용한 다대다 통신 시스템 구조를 설계하고, 이를 운용할 수 있는 지상통제 소프트웨어의 구조를 설계하였다.

호스트 장비의 도입을 통해 각각의 소프트웨어에서 모든 항공기의 데이터를 수신할 수 있도록 하고, 시스템의 실시간성 또한 확보하였다. 또한 소프트웨어의 가시화 모듈에서 모든 항공기의 상태 정보를 세세하게 나누어 표현함으로써 사용자가 자신의 상황에 맞게 임무를 계획할 수 있도

록 하였다. 마지막으로 이렇게 완성된 소프트웨어를 PILS환경에서 검증함으로써 본 연구에서 설계한 시스템과 소프트웨어가 요구조건으로 설정한 안정성과 효율성을 모두 만족한다는 것을 알 수 있다.

후 기

이 논문은 인하대학교의 지원을 받아 수행되었으며, 이에 감사드립니다.

참고문헌

- 1) Thomas J. Batkiewicz, K.C. Dohse, Vijay Kalivarapu, Thomas Dohse, Bryan Walter, Jared Knutzon, Derrick Parkhurst, Eliot Winer, James Oliver, 2006, "Multimodal UAV Ground Control System", 11th AIAA/SSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, Virginia.
- 2) Huh S. S., Yoo D. I., Moon S. W., Yoon S. H., Jung Y. D., Lee W. H., Sim H. C., 2009, "Development of an Autopilot system and a Ground Monitoring Station for Multiple UAVs", Proceeding of the 2009 KSAS Fall Conference, Gyeongju.
- 3) Francesca De Crescenzo, Giovanni Miranda, Franco Persiani, Tiziano Bombardi, 2007, "Advanced interface for UAV(Unmanned Aerial Vehicle) Ground Control Station", AIAA Modeling and Simulation Technologies Conference and Exhibit, Hilton Head, South Carolina.
- 4) Lee J. H., Ryu H., Kim J. E., Ahn L. K., 2004, "Development of Portable Ground Control System for Operation of Unmanned Aerial Vehicle", Journal of The Korean Society for Aeronautical and Space Sciences, Vol. 32, No. 10, pp. 127~133.
- 5) Hong D.P., Woo W.T., 2008, "Recent Research Trend of Gesture-based User Interfaces", Telecommunication Review, Vol. 18, No. 3, pp. 403~413.
- 6) Ye Hong, Jiancheng Fang, Ye Tao, 2008, "Ground Control Station Development for Autonomous UAV", Lecture Notes in Computer Science, Vol. 5315/2008, pp. 36~44.