

<附錄 1> 무인항공기 비행제어를 위한 시스템개요*

김성수**

목 차

- I. 개요
- II. 무인항공기 개발의 소요기술
- III. 비행제어컴퓨터 요구조건
- IV. 비행제어컴퓨터 시스템 구성
- V. 운용모드
- VI. 시스템의 신뢰도 확보
- VII. 물리적 특성
- VIII. 결론

I. 개 요

무인항공기(UAV: Unmanned Aerial Vehicle)는 정찰감시, 목표물 추적, 통신중계, 공격임무, 기상/자연환경 감시등을 수행하기 위해 전 세계적으로 군사/민간 분야에 있어 큰 관심을 일으키고 있다. 현재 개발 중이거나 실제 임무를 띠고 사용되고 있는 무인항공기를 살펴보면, 길이 15 cm 미만에 중량 100g 이하급인 초소형 비행체(MAV)에서부터 중량 12,000 kg, 탑재체 능력 900 kg, 항속거리 6,000 km에 항속시간이 40시간 이상 되는 대형 무인항공기에 이르기까지 실로 다양하게 개발되어 운용되고 있다. 무인항공기의 가장 큰 장점으로서는 인명피해 없이 비교적 저렴한 비용으로 중요 임무를 수행할 수 있다는 것이다. 이러한 사실은 최근 30

*본고는 “민군겸용 기술개발사업(Dual Use Technology Program)”으로부터 지원받은 과제수행결과 중 일부이며, 민군겸용 기술개발사업의 지원에 감사 드립니다.

** (주) 액티브 적용신호처리팀 연구원

여년 간의 개발과 실전 운용을 통하여 그 효용성이 입증되고 있다.

우리나라에서도 최근 들어 무인항공기에 대한 관심이 고조되고, 그 효용성과 실용성을 인정하면서 대학과 연구기관 그리고 산업체별로 활발한 연구와 개발 실적이 나오고 있다.

현재 실용화 되고있는 무인항공기의 형태로는 고정익이 주류를 이루고 있으며, 회전익과 비행선도 한 자리를 차지하고 있다. 회전익일 경우 우리나라의 지리여건상 매우 이점이 많은 형태로 고전적인 외형을 그대로 사용하는 경우와 ducted fan을 채택하는 경우가 있다. 위에서 열거된 것과 같이 무인항공기는 운용 목적과 운용환경, 개발자에 따라 외형과 추구하는 기능이 매우 다양하다. 하지만 실제 내부 시스템을 살펴보면 외형에 비행 구성요소가 대동소이하다는 것을 볼 수 있다.

본고에서는 이러한 내용들을 바탕으로 무인항공기의 핵심요소중의 하나인 비행제어컴퓨터(FCC: Flight Control Computer)의 개발과 비행제어컴퓨터가 갖추어야 할 기능 및 요구조건을 살펴보았다.

무인항공기(UAV)의 비행제어를 위한 시스템 개발은 우선 UAV의 정확한 임무와 주변 구성요소를 정의하여야 한다. 현대의 전기·전자 구현 기술은 UAV의 요구사항을 만족시키고 집적시키기에 충분하므로, 초기 개발단계에 있어서 정확한 제원의 정의는 개발 시간과 시스템 성능을 만족시키는 중요한 포인트가 된다. UAV를 위한 비행제어시스템은 물리적인 조건, 즉 크기와 중량에 많은 제약을 수반하며, 실제 운용되는 주변 자연환경의 영향을 고려하여야 하고, 특히 시스템의 안정성과 신뢰도를 확보하기 위한 요소 기술이 필요하다.

이러한 여러 가지 조건들을 바탕으로 본문에서는 일반적인 비행제어 시스템의 설계요구사항을 정하고, 구체적인 시스템의 구현 방법에 대하여 소개한다. 끝으로 시스템의 고장진단과 신뢰도 확보에 대한 내용을 소개한다.

II. 무인항공기 개발의 소요 기술

무인항공기의 개발은 운용 범위와 임무, 고도, 체공시간, 사용자에 따라 소요 기술의 차이를 보인다. 용도에 따라 살펴보면 우선 민수용, 산업용, 군사용으로 나누어 볼 수 있으며, 고도에 따라 분류되며, 운용되어지는 자연환경에 따라 개발되는 무인항공기의 기술이 달라진다 할 수 있겠다. 이러한 조건은 설계에서 제작에 이르기까지 적용되는 개념과 기술이 상이하게 달라진다. 이중에서도 특히 Global Hawk와 같이 군용의 목적을 가지고 있을 때, 일반적인 기능의 구현뿐만 아니라 고도에서의 운용조건을 만족시켜야만 한다.

다양한 목적의 무인항공기를 설계하고 개발하는 단계에서 요구사항을 만족시

키기 위해서는 다음과 같은 기술적 어려움을 해결해야 한다.

- High AR : 장기체공을 위해 유도항력을 줄일 수 있는 큰 AR이 요구됨.
- Low Rn : 상대적으로 일반 항공기보다 운용속도가 낮으므로 낮은 레이놀즈 수를 가져야 함.
- 추진기관 : 운용범위가 극지방, 적도, 고고도일 경우 공기의 밀도차 때문에 추진 기관과 배터리의 성능을 저하시킬 수 있음.
- 공탄성 : 높은 AR비에서 익폭이 매우 크므로 날개의 공탄성 문제가 야기됨.
- 고장진단 : 자율비행 조건에서 발생할 수 있는 돌발상황의 경우가 매우 많음. 위에서 언급된 문제점들을 극복하기 위한 요소기술들은 다음과 같다.
- 공력형상 : 높은 양항비를 얻기 위한 Flap 일체식 혁신 날개.
- 기체구조 : 초경량, 고강도 복합재 구조.
- 추진기관 : 고성능, 저 SFC 내연기관, 터보차저와 고효율의 가변피치 프로펠러.
- 항공전자 : Fail-Safe 자동조종장치, GPS 항법 및 위성통신
- 탑재장비 : 고성능, 소형·경량, 저 전력소모, 간편한 조작 및 높은 임무 신뢰성이 요구되는 시스템 개발. 특히 무인항공기의 효율성을 결정하게 되는 임무 신뢰성은 비행시간에 반비례하여 감소하므로 장기체공 무인항공기의 경우 이를 향상시키기 위한 기술개발이 이루어지고 있다.

III. 비행제어컴퓨터 요구조건

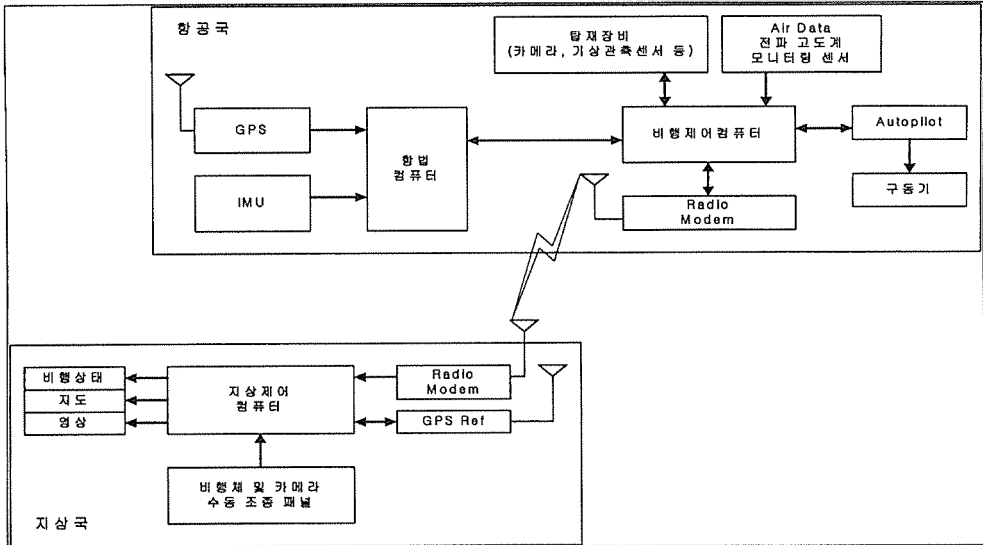
가. 무인항공기 전체 시스템

UAV의 전체적인 시스템은 그림1에서와 같이 크게 항공국과 지상국으로 구분된다. 지상국은 UAV와 RF통신을 하며 현재의 상태를 감시하고 비행경로의 설정과 임무를 선택할 수 있으며, 비상시 자동 무인비행을 해제하여 수동으로 비행체를 안전하게 귀환할 수 있는 기능을 제공한다. 또한 항공국으로부터 전달되는 각종 자료와 영상을 처리하고 분석한다.

항공국은 지상국과의 통신을 위한 RF Modem이 있으며 각종 데이터 수집을 위한 센서와 영상장비, 항공기 자체의 정보를 위한 센서, 자동항법을 위한 GPS와 IMU, 비행제어를 위한 구동기, 그리고 전체 시스템을 관장하는 비행제어컴퓨터로 구성된다.

무인항공기를 구현하기 위한 주요 관련 기술들을 살펴보면 <표 1>과 같다.

< 그림 1> 무인항공기 전체 시스템 구성도



< 표 1> 주요 관련 기술

세부 시스템	핵심 기술
항법 시스템	사세 측정용 GPS
	고장 허용시스템 설계
	항공기용 GPS/INS 알고리즘
자동비행제어 시스템	비행체 비행성능 분석
	제어 알고리즘
	비행상태 모니터링
	실시간 시뮬레이션
	자체 고장진단 및 복구
통신 및 지상 제어 시스템	Data 송수신 알고리즘
	Error Control Coding
	실시간 제어 컴퓨터
	영상 처리 및 Digital Mpa 적용
비행체 개발 및 시험 평가	항공기 안정성 및 제어 변수 계산
	외형 및 복합체 구조 설계
	비행 시험 및 데이터 분석
비행체 제작 및 운용	복합체 기체 제작
	비행 시험 기술

나. 비행제어컴퓨터 시스템 요구사항 및 구성

비행제어컴퓨터(FCC: Flight Control Computer)는 무인항공기의 상태정보를 입력받아 자세와 속도, 고도를 제어하여 입력된 비행경로를 비행하며 주어진 임무를 수행하기 위해 필요한 소형 컴퓨터로서, 비행에 관련된 주변환경 정보와 UAV 자체 정보를 취득하고 RF Modem을 이용하여 지상국과 통신을 가능하게 하여 전체적인 비행을 제어한다. 또한 사용목적에 따라 영상, 각종 데이터 수집장치 등을 위해 확장 가능한 여분의 인터페이스도 제공한다.

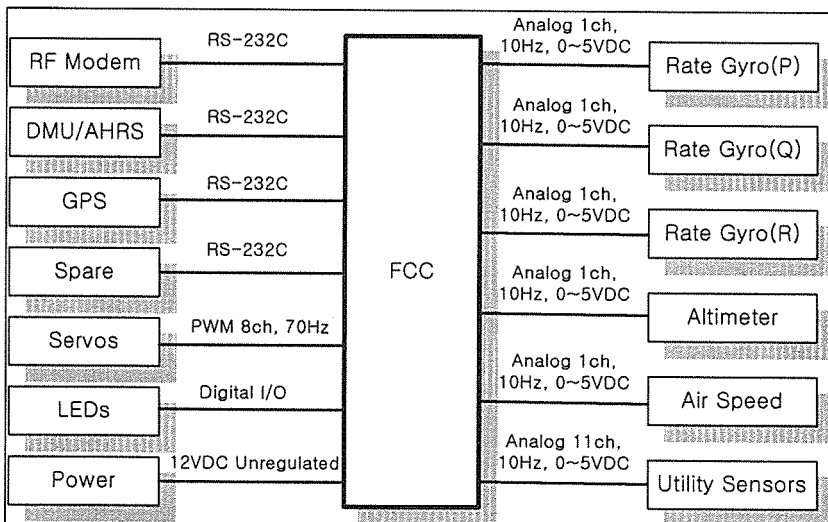
위와 같은 기능을 구현하기 위하여 비행제어컴퓨터는 장비의 운용목적 및 환경이 충분히 고려되어야 하며 시스템의 신뢰성, 정비성 및 확장성을 고려하여 외부 인터페이스(주변장치와의 관계), 물리적 특성(배치, 크기, 중량), 주요기능, 전원요구조건 및 환경조건 등을 만족하도록 구현되어야 한다.

1. 외부 인터페이스

외부 인터페이스는 무인항공기의 임무와 목적에 따라 다양하게 요구되므로, 본 논문에서는 무인항공기의 기본적인 기능을 구현할 수 있는 범위 내에서 기술하였다. 기타 목적에 따른 영상, 환경정보수집 등의 추가장치를 위한 인터페이스는 다양한 신호 형태(Analog, Digital, PWM, Serial등)에 따라 Spare로 제공한다.

각종 주변 장치들에 대해서는 비행제어컴퓨터가 그에 맞는 인터페이스를 제공하여야 하며 그에 따른 프로토콜의 정의, 신호명 그리고 각 신호별 신호수준을 정의하여야 하고 인터페이스를 위한 커넥터의 종류와 신호에 맞는 편번호를 할

< 그림 2> 비행제어컴퓨터 인터페이스 개념도



당하여야 한다.

인터페이스의 요구사항을 결정하기 위해서는 우선 주변 장치가 결정되어야 하고, 결정되어진 장치의 특성에 따라 각종 센서, 통신장비, 구동기 등의 정확한 신호 사양을 정리하여 이를 바탕으로 비행제어컴퓨터의 인터페이스 기능을 설계하게 된다. 기능의 설계가 끝나면 실제 구현에 앞서 주변장치와의 결선도, 배치도, 배터리의 사용방법 등 실제 무인항공기에 탑재되어지는 상황을 충분히 고려하게 된다. 주요 고려사항으로는 정비의 용이성, 조립의 용이성, 시험의 용이성, 간결한 배선의 결선 등이 있다.

또한 초기 계획된 장비만을 탑재하는 것이 아니므로 확장성을 고려하여, 향후 사용이 예상되는 장비의 인터페이스 탑재 공간의 확보도 필요하다.

다음 그림은 비행제어컴퓨터와 주변장치와의 인터페이스를 나타낸다.

- RS-232C 인터페이스: DMU/AHRS, GPS, RF Modem, 시스템의 고장 진단과 동작 테스트를 위한 인터페이스 제공
- Analog I/O 인터페이스: Gyro, Accelerometer, Air Speed, Altimeter, 엔진 RPM 등의 아날로그 센서를 위한 인터페이스 제공
- Digital I/O 인터페이스: 디지털 신호를 출력하는 신호와 상태표시를 위한 LED, 간단한 Digital 장치 제어를 위한 인터페이스 제공
- PWM(Pulse Width Modulation) I/O 인터페이스: 각종 구동기 작동을 위한 인터페이스 제공

2. 시스템 사양

비행제어컴퓨터의 시스템은 무인항공기의 탑재장치의 종류와 신호특성이 정의되면 이를 바탕으로 구현되어야 할 기능이 정의된다. 비행제어컴퓨터는 CPU 사양, 메모리 사양, 다중채널을 지원하는 ADC, PWM 신호를 위한 DPC/PDC, Serial 통신 제어, 디지털 입·출력 등 인터페이스와 관련된 기능과 외부 인터페이스와 관계없이 시스템 전체를 관장하고 테스트할 수 있는 모니터링 기능과 시스템 부팅/리셋을 위한 스위치 회로 그리고 시스템 동작시 오동작이나 고장을 검출할 수 있는 Watchdog 기능이 있어야 한다.

1) CPU

CPU는 주변장치와의 통신, 지상국과의 RF 통신, 데이터 수집 및 처리가 가능하여야 하고 자동비행프로그램이 정상적인 시간에 수행될 수 있는 충분한 성능을 가지고 있어야 한다. 이를 위해 32bit 부동소수점연산이 가능하고 최소 20MIPS 이상의 처리 속도를 가지고있는 DSP 또는 Core가 내장된 프로세서를 사용한다.

아래의 <표 2>는 시스템을 설계할 때 사용하는 기본적인 파라미터들로서 CPU를 선택하기 전에 고찰해야 하는 I/O포트 트래픽의 처리능력 지표로서, 무인항공

< 표 2> 시스템 설계 기본 파라미터

파라미터명	단위	성능	비고
CPU 버스폭	Bit	16, 32	Data
CPU 주파수	MHz	50, 60	Clock
CPU 메모리	KB	2, 4	RAM
CPU 버스 구조	-	S/P	Direct
CPU 명령어 처리	MIPS	25, 30	Cmd

기 전체 시스템의 입·출력 요소들이 각각 가지고 있는 신호특성들과 A-D신호 변환 및 시간상의 데이터 감지기간과 상관 관계를 나타낸다.

2) 메모리

비행제어컴퓨터의 메모리는 크게 4부분으로 나누어져 있다. ROM#1은 모니터 링프로그램과 펌웨어가 적재된다. ROM#2는 사용자의 자동비행프로그램이 적재된다. RAM#1은 ROM#2에 적재된 자동비행프로그램이 실제 수행되는 공간을 제공한다. RAM#2는 자동비행에 필요한 제어기의 이득값, 수집 데이터 등을 저장하기 위한 공간을 제공한다.

3) DPC(Digital to PWM Converter)

DPC는 서보모터를 제어하기 위한 신호를 발생시키는데, 서보모터 제어명령은 지상국에서 RF 모뎀을 통해 전달되거나, 자동비행프로그램에 의한 제어명령을 통해 전달된다. 이때 Digital 신호를 PWM신호로 변환시켜 주는 역할을 한다.

4) ADC(Analog to Digital Converter)

비행제어컴퓨터 주변에는 아날로그 센서를 이용하며 많은 데이터를 수집하게 되므로 이를 위한 ADC 기능을 구현한다. Analog신호는 잡음에 민감하므로 입력 신호는 노이즈 제거를 위해 필터를 통과하게 된다. 또한 주변 회로와의 간섭에 의한 잡음이 포함될 수 있으므로 초기 설계에서부터 PCB를 제작할 때까지 각별한 주의가 필요하다. 신호입력 모듈은 Main CPU가 직접 처리하지 않고 별도의 Processor가 처리하게 하여 Main CPU의 부하를 경감시켜 준다.

IV. 비행제어컴퓨터 시스템 구성

비행제어컴퓨터는 무인항공기에 탑재되어야 하므로 중량과 크기에 매우 민감

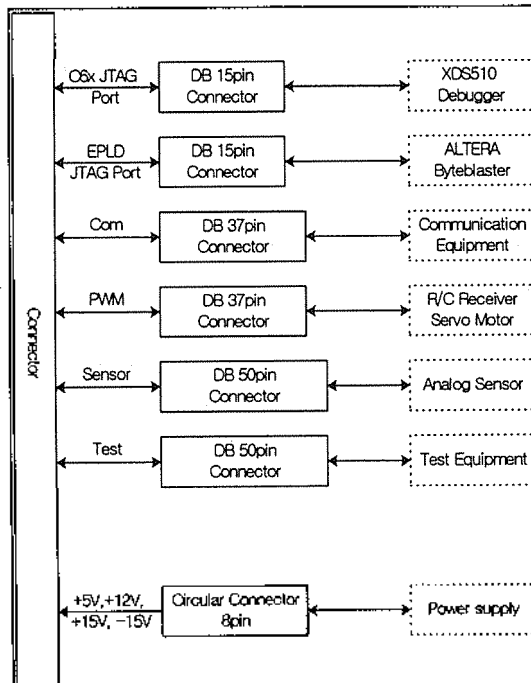
하다. 또한 충격과 주변환경에 대한 내성이 중요하게 작용하고 임무수행중 발생할 수 있는 고장과 기능장애를 극복하거나 적당히 대처 할 수 있어야 한다. 따라서, 일반적인 컴퓨터에서 채용하고 있는 버스 및 프로세서의 구조가 아닌 시스템의 부하와 초[Sec]당 처리 가능한 능력을 감안하여 내이터 버스의 대역폭과 중앙처리장치(CPU)의 클럭 주파수를 결정하고, 다중 입·출력 장치들의 접근에 대한 시간 분할 포트 할당 구조를 가지도록 설계한다.

가. 비행제어컴퓨터의 외부 접속

비행제어컴퓨터의 외부 컨넥터는 일반컴퓨터에서의 접속구조와는 달리 아래의 그림과 같은 특수한 접속구조를 가지게 된다. 이러한 구조는 무인항공기의 자동비행 프로그램 개발과 전체 시스템의 유지/보수 그리고, 자동비행에 사용되는 주변장치를 인터페이스하는데 필요한 컨넥터를 제공한다. 특히 두 개의 JTAG 포트를 이용하여 제어프로그램을 디버깅할 수 있게 한다.

또한 각 컨넥터는 그 모양을 달리하여 사용자가 장치를 연결하는데 있어 혼동을 방지하도록 한다.

< 그림 3> 비행제어컴퓨터의 외부 접속도

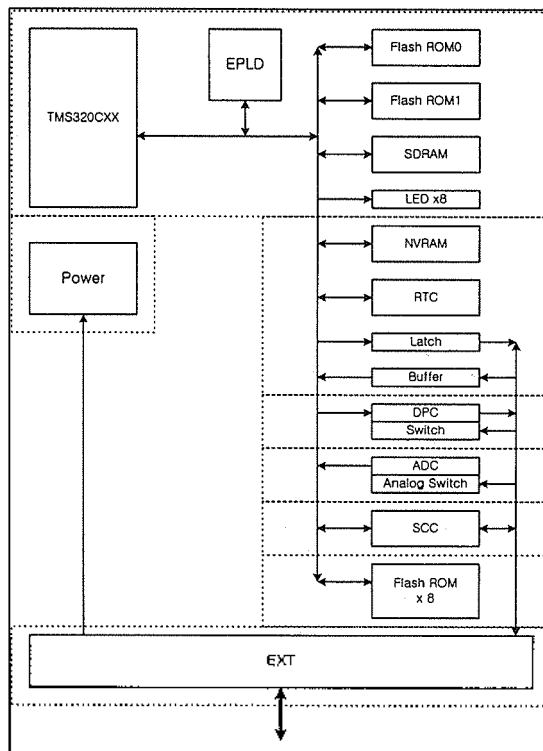


나. 비행제어컴퓨터의 내부 구성

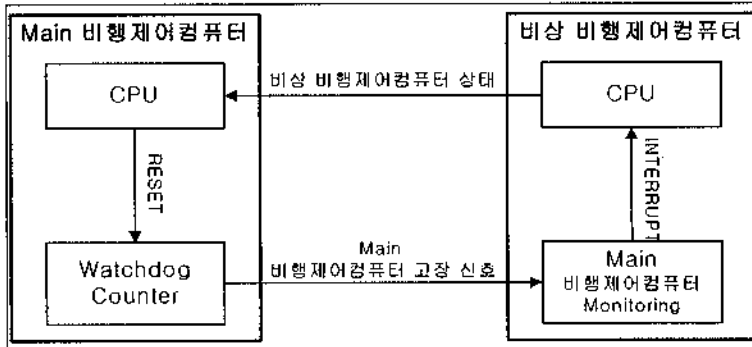
비행제어컴퓨터의 내부 구조는 시스템의 요구조건을 만족시켜야 될 뿐만 아니라 무인항공기라는 특수한 목적에 사용되므로 물리적으로 제한적인 요소가 매우 많다. 이를 위해 각각의 기능을 모듈별로 구성하였으며, 구성된 모듈은 내부 컨넥터에 의해 층층이 연결되어 있다. 모듈을 제어하기 위한 논리 제어회로는 EPLD를 이용하여 SoC(System on Chip)화 한다.

CPU는 고속의 실수연산이 가능한 DSP를 채택하였고, 내부의 프로그램 작업을 위해 JTAG 포트를 통하여 외부의 에뮬레이터와 접속하고 EPLD의 직렬 연결 프로그램 방식을 사용한다. 내부의 소자들을 제어하고 관리하기 위하여, DSP 소자가 비퍼를 통하여 제어신호들 간의 타이밍을 조절하게 되며, 이를 통해 어드레스 버스 신호와 데이터 버스 신호를 동시에 입·출력하고 필요한 연산을 수행하게 한다. 또한, 중앙처리장치(CPU)의 무한 루프 동작이나 오류를 방지하기 위해 Watch-dog 회로를 구성한다.

< 그림 4> 비행제어컴퓨터의 내부 구성도



< 그림 5> 비행제어컴퓨터의 이중화 구조

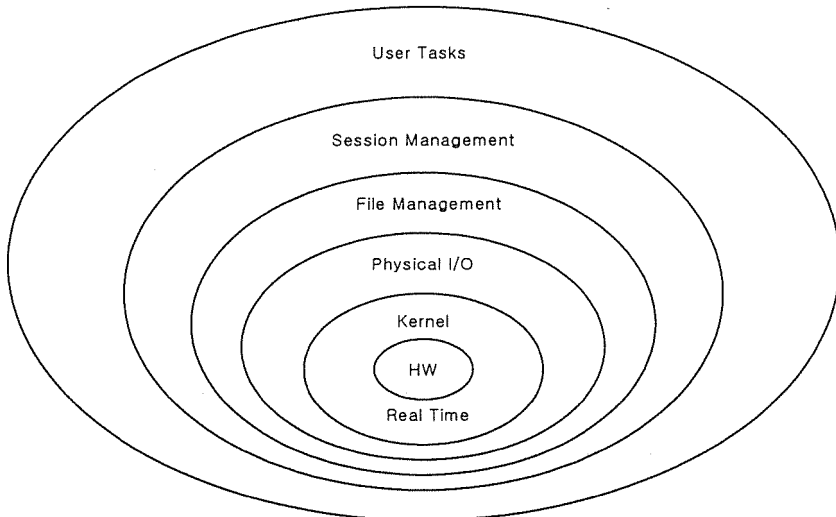


다. 소프트웨어 구성

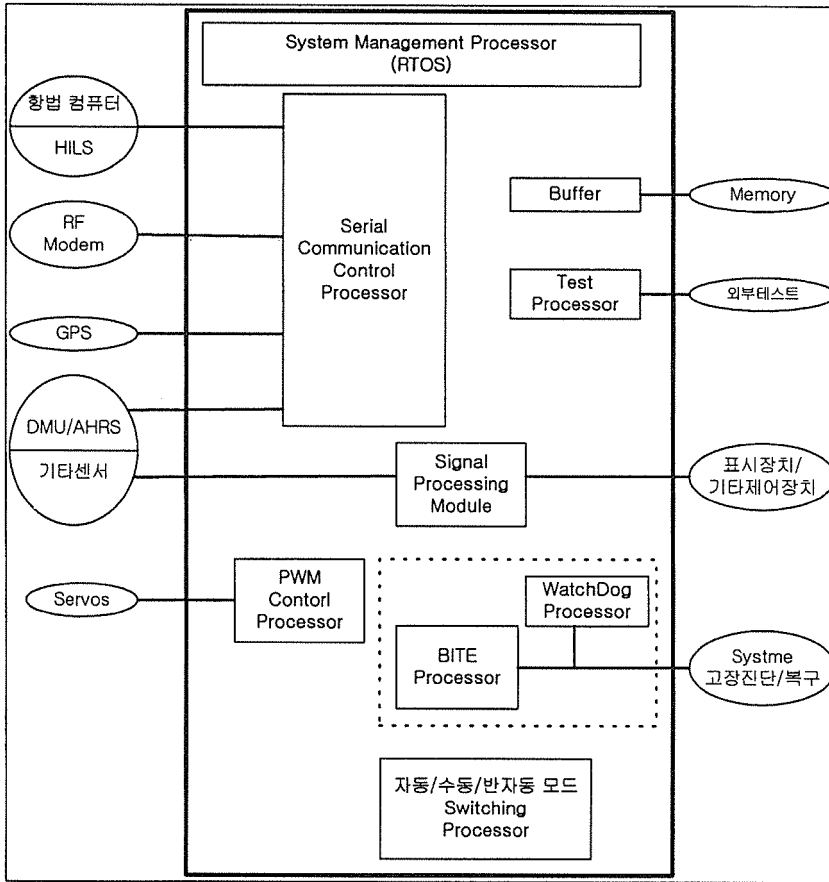
하드웨어에 이어 내장 소프트웨어에 대해 살펴본다. 소프트웨어는 크게 자동비행을 수행하기 위한 것과 전체 시스템을 관장하고 개발의 편의성을 제공하는 두 가지 목적으로 구분된다. 자동비행을 위한 프로그램은 사용자가 임의의 목적에 맞게 작성할 수 있으므로 여기서는 다루지 않는다.

비행제어컴퓨터를 위한 프로그램 개발에 있어서, 기본적으로 실시간 OS (RTOS)를 탑재하고 각각의 기능별로 Task를 설정하여 개발과 유지/보수가 비교적 수월하게 한다. 기능별로 나뉘어진 Task를 살펴보면 RTOS가 탑재되며 전체시

< 그림 6> 소프트웨어 개념도



< 그림 7 > 비행제어컴퓨터의 S/W 구성도



스텝과 Task간의 제어를 도와주는 System Management Processor가 최상위에 구성된다. 주변장치의 인터페이스를 위해, 시리얼 통신을 관장하는 Serial Communication Control Processor가 있고, 센서 신호의 입·출력을 위한 Signal Processing Module이 있으며, 자동비행 제어신호를 출력하여 서보를 구동시키는 PWM Control Processor, 외부테스트와 메모리 모듈을 위해 Test Processor와 Buffer로 구성한다. 시스템과 항공기의 고장을 진단하고 복구하기 위해 BITE Processor와 Watch-dog Processor가 사용되며, 비행 모드의 선택을 위한 자동/수동/반자동 모드 Switching Processor가 있다.

V. 운용 모드

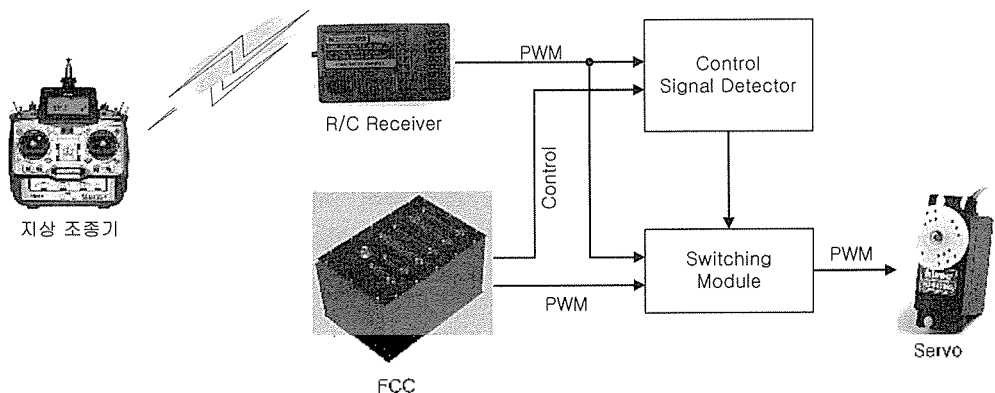
운용모드는 무인항공기를 운용하는데 있어 편리성, 안전성과 비상대응에 대한 선택적 비행 방법을 제공해 준다.

운용모드에는 크게 비행모드와 점검모드가 있다. 비행모드는 수동/반자동/자동으로 구분되어지며, 수동모드는 무인항공기의 제어신호를 지상국에서 직접 발생시킨다. 지상 조종기를 통해 발생하는 제어신호는 RF모뎀에 의해 무인항공기로 전달되어 조종면과 연결된 서보를 작동시킨다. 반자동 모드는 지상국으로부터 고도와 경사각(bank angle)에 대한 정보를 전달받고 무인항공기에 대한 상태 데이터는 탑재된 주변 장치로부터 전달받아 비행에 필요한 제어신호를 생성하여 반자동 비행을 수행한다. 자동모드는 지상의 도움 없이 항공기에 탑재된 장치만을 이용하여, 비행제어컴퓨터에 입력된 항로를 따라 자율비행을 수행한다.

위에 제시된 운용모드에서 비행제어컴퓨터는 무인항공기에 탑재된 각종 센서로부터 고도, 자세, 속도, 위치, 방위각 등의 자료를 획득하여 비행에 필요한 제어신호를 생성하고, 지상국으로 송신한다. 운용모드의 전환은 지상국의 제어신호에 의해 이루어지며, 반자동/자동모드로 비행할 때 자율비행이 불가능한 경우 수동모드로 변환하여 안전한 귀환을 유도할 수 있다.

점검모드는 비행제어컴퓨터의 개발과 자동비행 프로그램의 개발과정에서 제어회로의 이득값을 조정하고, 소프트웨어의 개선을 위해 다운로드 하거나 업로드할 때 사용되며, 초기 시동(booting)과 리셋의 경우 인터페이스하는 장비들의 연결 상태를 점검한다.

< 그림 8 > 무인항공기의 운용모드 선택 과정



VI. 시스템의 신뢰도 확보

무인항공기에 탑재되는 시스템은 고장이나 작동에러에 대해 매우 민감하다. 매우 짧은 순간의 오동작으로 인해 항공기의 안정성을 해치고 심할 경우 추락의 원인이 된다. 이를 방지하기 위해, 시스템에서 발생할 수 있는 오류나 고장을 찾아내고 이를 해결하며, 그 이전에 고장이나 오류의 발생 확률을 최소화시킨다. 이를 위해 실시간 시스템을 구성하게 되는데, 여기서 말하는 실시간(Real-Time)이란 사용자가 제어행위를 할 때 피제어 대상으로부터의 응답 시간(Response Time)에 대한 요구사항을 피제어 대상이 만족시키는 특성이라고 정의할 수 있다.

실시간 시스템은 시간의 제약서(Deadline)과 예측성을 중요시 여기는 시스템으로, 일반 시스템이 시스템 응답시간을 중요시 여기는 반면 실시간 시스템은 마감시간의 매치를 중요시 여긴다. 실시간 시스템에는 다음과 같이 Hard Real-Time System과 Soft Real-Time 시스템으로 나눌 수 있다.

Hard Real-Time System은 주어진 마감시간 이내에 작업을 완료하지 못하면, 치명적인 손실이 야기 될 수 있는 경우이다. Soft Real-Time System은 주어진 작업을 마감시간 이내에 완수하지 못할 경우 그 손실이 치명적이지는 않지만, 마감시간을 넘는 정도에 따라 점차적으로 증가하는 경우를 말한다.

실시간 시스템은 구성요소들이 우선순위에 따라 부여된 기능을 조화롭게 수행하여, 시스템이 목적했던 성능을 제대로 발휘할 수 있도록 구성한다. 하지만 완벽한 시스템이란 존재하지 않으므로, 개발자가 생각할 수 있는 모든 종류의 결함 원인에 대해서 면역(Immune)되어 있어야 한다. 즉 어떤 종류의 결함 원인에 대해서도 해결책을 가지고 있어야 하며, 생각할 수 있는 모든 종류의 결함에 대해서 그것을 허용해야 한다. 이를 결함 허용(Fault Tolerance)이라고 하는데, 결함 허용을 하고도 시스템은 원래의 성능을 발휘하여야 한다.

위의 내용에서 살펴본 바와 같이 비행제어시스템은 작업시간에 대한 응답이 확실해야 하고, 절대적인 신뢰성을 요구하므로, Hard Real-Time System에 속하게 된다. 이를 위해 지정된 태스크는 한계시간(Deadline)을 기준으로 동작하고, 각각의 작업에 대해서 우선순위(Priority)를 부여하며, 우선순위에 따라 작업이 수행되

< 그림 9> Hard Real-Time vs Soft Real-Time

Task's Limit Time			
Soft Real-Time	Best	Better	Worse
			Worst
Hard Real-Time	Success		Fail

는 시스템을 구성한다. 실시간 시스템의 구성과 함께 실시간 운영체제(Real-Time OS)를 탑재하는데, 이러한 경우 어떤 태스크를 한계시간이 끝나기 전에 수행하는 것이 주관점이 된다. 대부분의 RTOS의 경우 선점형(Preemptive)이므로, 가장 높은 우선순위를 가진 태스크가 항상 실행되고 있는 형태이다. 따라서 가장 빈번히 일어나는 태스크의 우선순위를 가장 높게 책정한다.

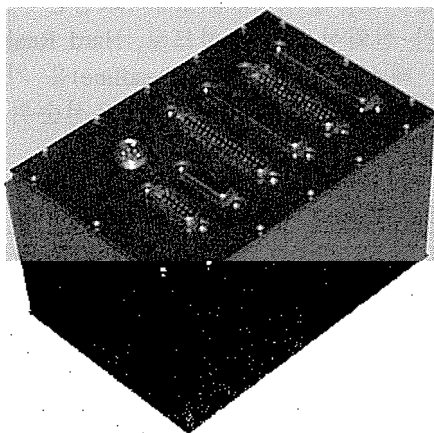
무인항공기는 작업환경이 다양하고, 다양한 임무를 수행하며, 뜻하지 않은 상황을 맞이하게 될 확률이 매우 높다. 따라서 철저한 대비를 위해 실시간 시스템을 구현하고 결함에 대해서 면역성을 가지고 있어야하므로, 하드웨어뿐만 아니라 소프트웨어의 신뢰성과 실시간성이 매우 중요하다.

VII. 물리적 특성

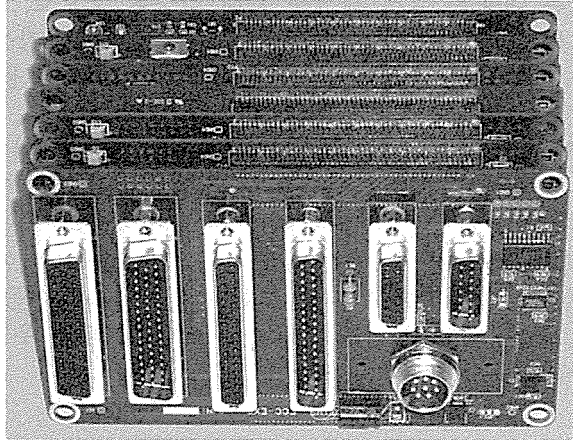
무인항공기에 탑재되는 장비는 물리적 특성을 고려하여 설계한다. 이때 고려해야 할 항목으로는 크기 및 중량, 장착방식, 커넥터의 선정과 장비를 고정시키기 위한 위치 그리고 진동, 온도, 습도, 가속도 등에 충분한 내성을 지니고 있어야 한다. 무인항공기의 Payload를 고려하여, 허용 가능한 최대크기와 중량을 결정하고, 장착위치는 무게중심에 영향을 주지 않는 범위로 제한하여 최소화하는 것이 바람직하다.

전체 크기와 중량의 최소화를 위해 FPGA 또는 ASIC을 이용하여 SoC화하며, 케이스의 재질은 알루미늄 합금을 사용한다. 또한 전자기 간섭의 영향은 외부와 연결되는 커넥터를 통해 많이 발생하므로, 이를 차단할 수 있도록 EMI처리를 한

<그림 10> FCC 외형 사진



< 그림 11 > FCC 내부 사진



다. 각종 주변장치가 좁은 공간에 밀집되어 있어 장치간의 간섭이 발생할 수 있으므로, 이를 해결하기 위해 비행전 철저한 테스트가 요구된다. 컨넥터의 경우 사용자의 실수로 잘못 연결될 수 있으므로, 모양을 달리하여 선정한다.

비행제어컴퓨터의 환경조건은 무인항공기의 운용환경에 따라 결정된다. 유인 항공기에 대한 환경조건은 MIL-STD-810이나 RTCA DO-160에 정의 되어있지만, 아직 무인항공기에 대한 정확한 규정은 확립되지 않은 상태이다. 따라서 무인항공기의 운용상 발생할 수 있는 환경조건에 대하여, 정의하고 이를 적용한다.

VIII. 결 론

비행제어컴퓨터의 개발은 적용되는 무인항공기의 특성에 따라 요구되는 기능과 성능이 다양하게 정의된다. 하지만 전체적인 접근방식은 제시된 내용과 크게 다르지 않다고 본다. 무인항공기 초기 개발 단계에서 주요요구기능, 성능, 주변 인터페이스, 물리적 특성, 환경조건 등을 정의하였으며 이를 바탕으로 비행제어 컴퓨터를 구성하게 된다. 여기에 정비성, 신뢰성, 고장에 대한 진단과 회복에 대한 기능이 앞으로 추가되어야 할 과제이다. 또한 제작하는데 있어 부품의 원활한 조달과 비행제어컴퓨터가 추구하는 기능과 환경조건에 대한 내성을 갖추고 있는지에 대한 정확한 검증이 필요하다.

시스템을 구성하는 문제에 있어서 단순한 기능의 구현은 커다란 기술적 문제를 야기하지 않는다. 단지 소형화와 집적화의 정도에 따라 기술적 문제가 요구될

수 있지만 더 중요한 문제는 전체시스템의 신뢰도 확보이다.

앞서 제시된 모든 내용은 실제 제작을 통해 검증된 것으로, 이를 토대로 유사 장비의 개발과 적용이 가능하다고 보여진다. 특히, 무인항공기용이 아닌 MAV용 비행제어컴퓨터의 개발에 앞선 연구단계라고 생각 할 때 MAV용 비행제어컴퓨터 개발에 있어 기간과 반복되는 실수를 줄이고 더 나은 초소형, 초경량의 장비를 만들 수 있다. 특히 MAV인 경우 현재 개발되어 지는 PDA의 전자적 자원을 이용할 경우 특별한 초기 개발의 과정 없이 충분한 기능을 구현할 수 있다고 본다.

무인항공기 분야는 유인항공기에 비해 기존 선진국과의 기술적 격차가 크지 않으며, 무인항공기에 적용할 수 있는 국제적 규약 또한 미비한 상태이다. 이러한 점을 감안했을 때 초기 개발 투자에 대한 손익을 따져 수동적인 개발과정을 펼쳐나가는 것은, 무인항공기 분야에서도 선진국에 또 다른 우리의 앞마당을 내어주는 것과 같은 결과를 초래한다.

[참고문헌]

- 김중욱 외 4인(1999), 『무인항공기 기술 및 시장동향 조사』, 한국항공우주학회 추계 학술 발표회
- 박무혁 외 3인(2000), 『무인항공기 비행제어컴퓨터 하드웨어개발』, 한국항공우주학회 추계 학술 발표회
- 박승창 외 4인(2001), 『소형 무인항공기용 자동 비행제어컴퓨터의 버스 구조 개발』, 한국군사과학기술학회
- 한국항공우주연구원(2000), 『무인항공기 비행제어컴퓨터 사용자 요구서 기술문서』
- (주)액팀스(2001), 『비행제어컴퓨터 설계서』
- 윤석준(1999), “무인항공기 비행제어시스템”, 『항공산업연구』, 세종대학교 항공산업연구소, 제 52집

[약어설명]

ADC	: Analog to Digital Converter
AHRS	: Attitude and Heading Reference System
ARINC	: Aeronautical Radio Incorporated
DIO	: Digital Input Output
DMU	: Dynamic Measurement Unit
DPC	: Digital to PWM Converter
DSP	: Digital Signal Processor
EMI	: Electromagnetic Interference
EPLD	: Embedded Programmable Logic Device
Ext	: External
FCC	: Flight Control Computer
FPGA	: Field Programmable Gate Array
GPS	: Global Positioning System
HILS	: Hardware In the Loop Simulation
IMU	: Inertial Measurement Unit
INS	: Inertial Navigation System
I/O	: Input Output
JTAG	: Joint Test Access Group
LCFCC	: Low Cost Flight Control Computer
MAV	: Micro Air Vehicles
MIPS	: Million Instructions Per Second
NFCC	: Normal Flight Control Computer
NVRAM	: Non-Volatile Data Output RAM
PWM	: Pulse Width Modulation
R/C	: Radio Control
RF	: Radio Frequency
RTC	: Real Time Clock
RTOS	: Real-Time Operating System
SCC	: Serial Communication Control
SDRAM	: Synchronous Dynamic Random Access Memory
SoC	: System on Chip
UAV	: Unmanned Aerial Vehicle