J. Korean Soc. Aeronaut. Space Sci. 50(5), 349-357(2022)
DOI:https://doi.org/10.5139/JKSAS.2022.50.5.349
ISSN 1225-1348(print), 2287-6871(online)

무인기용 이중화 비행조종컴퓨터의 고장관리 설계

오태근[1], 윤형식[2]

A Fault Management Design of Dual-Redundant Flight Control Computer for Unmanned Aerial Vehicle

Taegeun Oh[1] and Hyung-Sik Yoon[2]
Agency for Defense Development, Daejeon, Republic of Korea

ABSTRACT

Since the flight control computer of unmanned aerial vehicle (UAV) is a flight critical equipment, it is necessary to ensure reliability and safety from the development step, and a redundancy-based fault management design is required in order to operate normally even a failure occurs. To reduce cost, weight and power consumption, the dual-redundant flight control system design is considered in UAV. However, there are various restrictions on the fault management design. In this paper, we propose the fault detection and isolation designs for the dual-redundant flight control computer to satisfy the safety requirements of an UAV. In addition, the flight control computer developed by applying the fault management design performed functional tests in the integrated test environment, and after performing FMET in the HILS, its reliability was verified through flight tests.

초 록

무인항공기의 비행조종컴퓨터는 비행 안전에 필수적인 장비로써 개발 단계에서부터 신뢰성과 안전성의 확보가 필수적이며, 고장 발생 시에도 정상적으로 기능을 수행할 수 있는 다중화 기반의 고장관리 설계가 요구된다. 무인기의 경우에는 비용, 무게, 전력소모 등을 감소하기 위하여 비행조종시스템의 이중화 설계를 고려하지만, 고장관리를 위한 고장 검출 및 분리 설계에 많은 제약이 있다. 본 논문에서는 무인기용 이중화 비행조종컴퓨터의 신뢰성을 향상시키기 위한 고장 검출 및 고장 분리를 위한 고장관리 설계 방안을 제안한다. 그리고 고장관리 설계를 적용해 개발한 비행조종컴퓨터는 통합시험환경에서 기능 시험을 수행하고 HILS 환경에서 고장 영향성 확인 시험을 수행한 후 무인기에 탑재하여 비행시험을 통해 그 신뢰성을 검증하였다.

Key Words : Flight Control Computer(비행조종컴퓨터), Operational Flight Program(비행운용프로그램), Flight Critical Software(비행안전 필수 소프트웨어), Dual Redundancy (이중화), Fault Management(고장 관리)

Ⅰ. 서 론

기술의 발달과 함께 무인항공기는 다양한 형태와 향상된 성능으로 다양한 분야에서 활용되고 있다[1,2]. 무인항공기는 항공기 비행제어와 임무 수행을 위한 다양한 서브시스템들로 구성된다. 특히 비행조종컴퓨터(FLCC; Flight Control Computer)는 항공기의 센서, 구동면 등과 함께 비행조종시스템의 구성품

† Received : February 15, 2022 Revised : March 16, 2022 Accepted : March 18, 2022
[1] Senior Researcher, [2] Chief Researcher
[2] Corresponding author, E-mail : yhs1117@add.re.kr

중 하나로써 항공기의 비행제어 및 자동비행 기능 등을 수행한다[3,4]. 그러나 비행조종컴퓨터는 무인항공기의 안전한 비행을 위한 필수 장비로써 고장 발생 시 비행제어 및 자동비행이 불가할 수 있으며, 이는 항공기의 손실로 이어질 수 있다. 따라서 비행조종컴퓨터는 고신뢰성을 보장하기 위하여 고장 발생 시에도 정상적인 동작을 유지할 수 있도록 고장허용을 위한 고장관리 설계가 필요하다.

비행조종컴퓨터의 고장관리 설계는 비행조종시스템을 구성하는 장비의 고장(Fault) 발생 시 이를 검출하고 분리하여 이로 인해 비행조종시스템의 오동작 또는 성능 저하를 유발하는 결함(Failure)을 발생하지 않도록 재형상을 수행하는 기능을 설계하는 것이다.

다중화 기법은 고장허용을 위한 설계 방법 중 하나로써 장비의 일부에 발생된 결함을 마스킹하여 결함이 장비의 고장으로 전파되는 것을 방지하는 방식으로 보팅 알고리즘 등이 사용된다. 보팅 알고리즘은 일반적으로 다수결 보팅(majority voting) 방법을 사용하며, 이를 위해서는 3중화 이상의 하드웨어 다중화가 필요하다[5-7]. 그러나 현재 운용되거나 개발 중인 많은 무인기들은 비용, 무게, 전력소모 등을 최소화하기 위하여 최소의 다중화 수준인 이중화 설계를 고려한다[8]. 하지만 3중화 이상에서 사용하던 고장관리 방법으로는 이중화 시스템의 고장을 검출할 수 없어 새로운 고장관리 설계가 필요하다.

본 논문에서는 무인기용 이중화 비행조종컴퓨터의 신뢰성을 향상시키기 위하여 고장관리 설계에서 시스템의 신뢰성에 영향을 미치는 고장 검출, 시간 동기화 및 고장 분리 설계 방법을 제안한다.

II. 본 론

2.1 이중화 비행조종컴퓨터 고장관리 설계

2.1.1 이중화 구조 개요

군용 유인 항공기에서 비행조종시스템은 신뢰성을 향상시키기 위하여 3중 또는 4중의 다중화 시스템 기반의 고장허용 설계를 적용하고 있으며, 고장관리를 위해서 다수의 비행조종컴퓨터 채널들이 일치하는 결과를 선택하는 다수결 보팅 로직을 사용한다. 그러나 현재 운용되거나 개발 중인 많은 무인기들은 비용, 무게, 전력소모 등을 최소화하기 위하여 최소의 다중화 수준인 이중화 설계를 고려하지만, 이중화 구조에서는 고장관리 기능을 위한 고장 검출 및 분리 설계가 어려우며 설계 수준에 따라 무인기의 신뢰성이 영향을 받는다. 이중화 구조의 고장관리 설계를 위해서 물리적인 여분의 채널을 추가하지 않고 다중화 수준을 높이기 위해서 수학적 모델이나 지식기반의 데이터를 이용한 해석적 중복성(analytical redundancy) 구조를 설계하거나 온톨로지 기반 고장

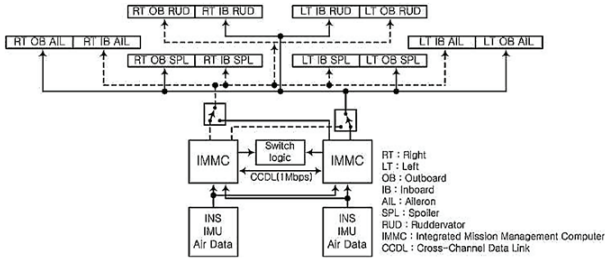

Fig. 1. Hardware Configuration of Dual-Master System [12]

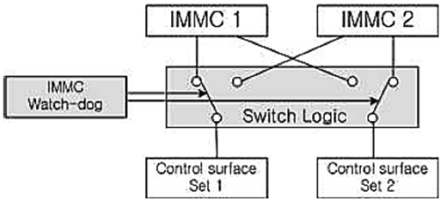

Fig. 2. Switch Logic [12]

진단, 가중치 기반 보팅 알고리즘 등의 다양한 고장진단과 관련한 연구가 있다. 그러나 해석적 중복성 구조 설계의 경우 시스템의 정확한 모델링 어려움으로 인해 실제 적용이 제한적이며, 다양한 고장 진단 관련 연구들도 센서, 엑츄에이터 등 비행조종시스템 연동 장비들에 대한 고장 검출 수준을 높이기 위해 진행되어 비행조종컴퓨터 자체 고장관리를 위한 연구가 부족한 상황이다[9-11].

"Dual-Master" 이중구조를 채택한 고고도 무인 정찰기인 글로벌 호크(RQ-4A)의 경우에는 비행조종컴퓨터의 기능을 수행하는 IMMC(Integrated Mission Management Computer)1, IMMC2에서 센서 및 연동장비의 상태 정보를 이용하여 보팅 로직을 통한 센서 및 연동장비 고장을 판단한다(Fig. 1). 그리고 두 채널의 IMMC 중 한 채널에서 고장이 발생할 경우는 외부에 별도로 개발된 Switch logic에서 고장이 발생한 IMMC를 판단한다. IMMC는 상호 간의 동기신호 생성 및 자기 자신의 고장 상태를 검출하는 BIT(Built In Test) 결과를 생성하고, Switch logic은 각 채널에서 생성하는 BIT 상태 정보 및 WDT(Watch Dog Timer) 결과에 의존하여 고장이 발생한 채널을 결정한다. Switch logic의 구성은 Fig. 2와 같다. 그러나 글로벌 호크의 경우에도 IMMC에서 발생 가능한 다양한 고장들에 대해 IMMC 고장 채널을 분리하기 위한 방법이 제한적이며, Switch logic에 의한 단일 결함점(single point of failure) 발생으로 인한 시스템 신뢰성을 감소시키는 단점이 있다[12-14]. 본 논문에서는 "dual-master" 이중구조를 가진 무인기에서 글로벌 호크 무인기의 단일 결함점에 의한 취약점을 극복하면서 기존의 보팅 로직으로 고장 분리가 불가능하였던 비행조종컴퓨터 자체 고장에 대해 고장 채널 분리가 가능하도록 고장관리 설계 방안을 제안한다.

2.1.2 다중화 신뢰도 영향성

비행조종컴퓨터의 다중화 수준은 비행제어시스템에 할당된 정량적 비행안전성 요구도인 비행제어시스템의 비행 조종 손실 확률(PLOC; Probability-Of-Loss of Control)에 따라 결정된다. 다중화 수준에 따른 비행 조종 손실 확률의 수식은 다음과 같다[15].

$$\text{단일 채널} : PLOC = \frac{T}{MTBCF}$$

$$\text{2중 채널} : PLOC = \frac{T \times C_1^2 \times (1-ST)}{MTBCF}$$

$$\text{3중 채널} : PLOC = \frac{T \times C_2^3 \times (1-ST)}{MTBCF^2}$$

여기서 T는 1회 비행시간, ST는 자가진단 커버리지(탐지 및 분리가 가능한 치명적 결함의 비율), $MTBCF$는 주요 고장 간 평균시간(Mean Time Between Critical Failures)이다. ST는 다음의 고장 탐지율과 고장 분리율 수식을 이용하여 계산 가능하다[16].

$$\text{고장탐지율}(\%) = \frac{N_C^{op}}{N_C} \times 100$$

$$\text{고장분리율}(\%) = \frac{N_C^{sp}}{N_C^{op}} \times 100$$

여기서 N_C^{op}는 운용 중 탐지 가능한 치명적 고장 건수, N_C는 전체 치명적 고장 건수 및 N_C^{sp}는 분리 가능한 치명적 고장 건수이다.

항공기의 다중화 수준이 결정될 때 PLOC를 감소시키기 위해서는 ST값을 증가시켜야 한다. ST는 장비에 발생한 고장을 검출하고 고장이 발생한 채널을 식별할 수 있는 능력을 의미한다. 장비에 고장 발생 시 고장이 발생한 채널을 식별할 수 없는 경우 ST의 값은 0.5이며, 단일 및 2중 채널의 PLOC는 동일하다. ST를 증가시킬수록 PLOC의 값이 감소하므로 ST를 증가시킬 수 있는 고장관리 설계는 신뢰성 향상을 위해 매우 중요하다. 본 논문에서 제안하는 이중화 비행조종컴퓨터 고장관리 설계 기술은 이러한 ST를 증가시켜 무인기의 신뢰도를 향상시키기 위한 방법을 설명한다.

2.1.3 고장검출 설계

고장관리 설계의 첫 번째 단계는 장비에 발생한 고장을 검출하는 단계로써 비행조종컴퓨터 내부 및 외부에서 발생할 수 있는 다양한 고장 원인을 판단하고 검출한다. 비행조종시스템의 고장 원인을 식별하기 위하여 하향식 시스템 안전 및 고장 분석 방법을 통해 고장을 분류하였으며, 고장 원인에 따라 고장을 검출하기 위한 방법은 기존 연구에서 제안하였으며[5], 그 내용은 Table 1과 같다.

Table 1. Fault Detection Methods

고장 분류	고장 검출 방법
외부 입력 고장	▪ 센서 입력 비교 : 허용치 초과 불일치 확인 ▪ 센서 입력 검사 : 신호 유효 범위 확인 ▪ 신호 변화율 유효성 확인 ▪ 통신 유효성 검사 : Checksum, CRC 확인
외부 장비 고장	▪ 장비 상태 신호 검사 - 장비 자체결과 확인 - Heartbeat Count 확인
FLCC 채널 하드웨어 고장	▪ 자체점검(Built In Test) ▪ 입력/출력 모니터
FLCC 채널 소프트웨어 고장	▪ 제어 흐름 확인 (control flow check) ▪ 중복 확인(replication check) ▪ 시간성 확인(timing check) ▪ 합리성 확인 (reasonableness check) ▪ 데이터 구조 확인 (Data structural check)

비행조종컴퓨터에서 고장 검출 방법은 고장 원인에 따라 개별 채널에서 수행하는 자체점검으로 판단하거나, 양 채널에서 수신되거나 판단되는 신호들을 비교하고 그 차이가 설정된 한계값(Threshold value)을 초과할 때 고장으로 판단한다.

외부 연동장비의 고장 검출을 위해 각 장비로부터 고장 상태정보를 수신하거나 입력신호의 유효성 범위를 벗어나는 경우 개별 채널에서 고장 판단을 수행한다. 이외의 경우에는 상호 채널 모니터를 통해 보팅 알고리즘을 이용하여 입력신호에 대한 고장 판단을 수행한다. 비행조종컴퓨터 하드웨어(프로세서, 메모리, 통신 모듈 등) 고장에 대해서는 기본적으로 자체점검(Built In Test)을 통해 고장을 판단한다. 또한, 비행조종컴퓨터 소프트웨어 고장을 검출하기 위해 제어 흐름 확인 등의 방법을 이용하여 자신의 채널 상태를 주기적으로 확인하여 고장을 판단한다. 하지만, 고장 채널의 하드웨어 및 소프트웨어 자체점검 결과에 대한 신뢰성이 낮으므로 타 채널에서도 상호 채널 모니터를 통해 상대 채널의 고장을 감시하도록 설계하였다. 자체점검으로 고장을 검출하는 경우는 고장 채널의 식별이 가능하지만, 상호 채널 모니터를 이용하는 경우는 장비의 고장 발생 여부는 판단할 수 있지만 고장 채널의 구분은 불가능하므로 추가적인 고장분리를 위한 알고리즘이 요구된다.

2.1.4 시간 동기화 설계

다중화 장비에서 상호 채널 모니터를 이용하여 입력신호의 고장을 판단하는 경우 각 채널에서 수신하거나 판단하는 신호의 시간 동기화 차이는 비교 신호 오차를 발생시킨다. 따라서 채널 간 시간 동기화의 정확성은 고장 검출의 정확성을 향상시키기 위해 매우 중요하다.

다중 채널로 구성된 비행조종컴퓨터의 시간 동기화 설계를 위해서 기존의 방법은 크게 하드웨어적인 동기화 설계 방법과 소프트웨어적인 동기화 설계 방법이 적용되었다. 하드웨어적인 동기화 방법은 각 채널 간 주기 시간의 차이에 따른 비동기화 상태를 일치시키기 위한 추가적인 공통 클럭을 이용하여 각 채널을 동기화 시킨다. 설계가 단순한 장점이 있지만, 공통 클럭의 결함에 의한 전체 채널의 동기화를 유지할 수 없는 단일 고장점(single point of failure) 문제를 포함하고 있다. 소프트웨어적인 동기화 방법은 개별 채널에서 상호 채널 데이터 링크(CCDL; Cross Channel Data Link) 통신을 이용한 동기화 메시지나 디스크리트 신호를 이용하여 타 채널의 동기화 시작 시간 정보를 획득하고, 동기화 시작 시간이 중간 값을 가지는 채널을 중심으로 채널의 주기 시작시간을 조절하여 채널 동기화를 수행한다. 그러나 이중화 채널에서는 중간 값을 가진 채널 선택이 불가능하다. 두 채널의 동기화 신호 시간 차이의 평균값을 이용하여 동기화 조정을 수행한다면 설계는 단순화할 수 있다. 하지만 채널 클럭 발생을 위한 하드웨어 고장 발생 시 동기화가 유지되더라도 두 채널 모두 주기 시간이 지속적으로 증가하거나 감소하는 문제가 발생할 수 있으므로 이로 인한 고장을 식별하기 위한 설계가 요구된다.

본 연구에서는 이중화 채널의 주기 프레임 시작 시간을 빠르고 안정되게 일치시키기 위하여 이중화 채널들이 동시에 동기화 디스크리트 신호를 타 채널로 전송하고 두 채널이 동시에 동기화 신호의 차이를 보상하여 프레임 시작 시간을 조절하는 프레임 워크(Frame Walk) 방식으로 구현하였으며, 동기화 시간을 단축하기 위하여 Fig. 3과 같이 양방향 동기화를 수행하도록 설계하였다.

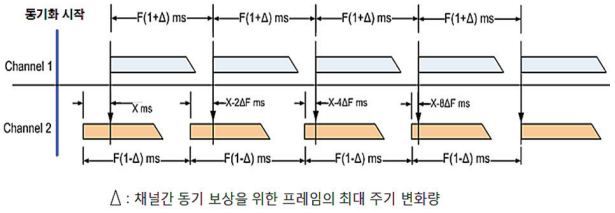

Fig. 3. Concept of Dual Channel Bi-directional Synchronization

(1) 초기 동기화 시간 : 동기화 실행 후 6초 이내
(2) 동기화 최대 허용오차 : 초기 동기화까지는 1,000 μs 이내, 안정화 이후에는 10μs 이내
(3) 타이머 변경 최대치 제한 : 20ms±1ms

상기 요구도를 고려하여 프레임 주기의 보정시간 △과 최대 허용오차는 다음과 같이 결정된다.

$$\triangle = \frac{\left[X \times \frac{F}{T_i^{sync}}\right]}{2}$$

여기서 X는 두 채널의 동기화 신호의 시간 차이, F는 프레임 주기 시간, T_i^{sync}는 초기 동기화 시간이다.

이중화 채널의 시간 동기화를 위하여 각 채널은 프레임 단위의 주기로 디스크리트 동기화 신호를 이용하여 클럭 카운터의 값을 비교하여 채널 간 동기화 신호의 차이 시간을 계산하고, 다음 프레임 주기 시간 조정을 위해 자신의 주기 타이머 클럭을 수정한다. 각 채널에서 동기화 알고리즘의 순서도는 Fig. 4와 같다.

비행조종컴퓨터의 한 채널에서의 타이머 고장 등으로 인한 프레임 주기의 드리프트 고장 발생 시 이중화 비행조종컴퓨터의 전체 프레임 주기 시간 위반 고장을 방지하기 위하여 동기화 로직에 의한 채널 타이머의 주기 변경 최대 값을 제한하고, 타이머 주기 보정 누적 시간의 1ms를 초과하는 경우 동기화 고장으로 판단한다. 그러나 이 경우에도 고장 채널의 식별은 불가능하므로 추가적인 고장분리를 위한 알고리즘이 요구된다.

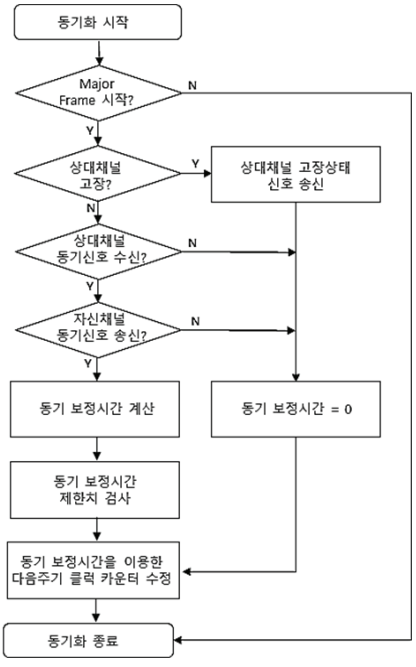

Fig. 4. Flowchart of Dual Channel Synchronization

2.1.5 고장 채널 분리 설계

고장관리 설계의 두 번째 단계인 고장채널 분리 기능은 다중화 장비에서 식별된 고장에 대하여 고장 발생 채널을 식별하는 단계이다. 다중화로 구성된 장비의 고장채널 식별을 위한 일반적인 방법은 각 채널의 신호를 비교하는 다수결 보팅 방법이 사용된다. 그러나 이중화 구조의 장비인 경우에는 다수결 보팅을 사용할 수 없으므로 고장 채널을 식별하기 위해서 다양한 연구가 진행되고 있다. 그러나 많은 연구가 진행되고 있는 해석적 모델을 이용한 다수결 보팅의 경우 정확한 해석적 모델의 개발이 어렵고, 모델의 부정확 정도가 또 다른 오류를 발생시킬 수 있어 적용이 제한적이다. 그리고 이중화 비행조종컴퓨터의 외부 연동장비의 고장에 대한 고장 채널 분리를 위한 연구는 다양하게 이루어지고 있지만, 비행조종컴퓨터의 내부 고장에 대해서 고장 채널을 분리하기 위한 설계 방법에 대한 연구는 상대적으로 활발하게 이루어지고 있지는 않다.

본 논문에서는 이중화로 구성된 비행조종컴퓨터에서 고장 검출은 할 수 있지만, 고장 채널의 분리가 어려운 비행조종컴퓨터 내부 결함, 동기화 고장, CCDL 통신 고장, 보팅 로직에서 데이터 비교를 통한 불일치 발생 등의 경우에 고장 채널을 분리하기 위한 알고리즘을 설계하였으며, 내용은 다음과 같다[17].

(1) 자신의 채널이 먼저 고장을 판단한 경우(Fig. 5)

자신의 채널 소프트웨어에서 먼저 고장을 판단한 경우 자체점검을 통해 자신의 채널 고장 유무를 먼저 판단하고 그 결과를 채널 고장 판단 로직을 통해 타 채널로 송신한다. 이때 자신의 채널이 고장으로 판단되고 타 채널이 정상으로 판단되는 경우 자신의 채널을 고장상태로 전환하지만, 고장 채널의 판단이 불가능한 경우 채널 고장 판단 로직에 의해 채널의 상태가 결정된다.

(2) 타 채널에서 고장을 먼저 판단한 경우(Fig. 6)

타 채널에서 먼저 고장을 판단하고 타 채널로부터 자체점검 요청을 위한 디스크리트 신호 수신 시 높

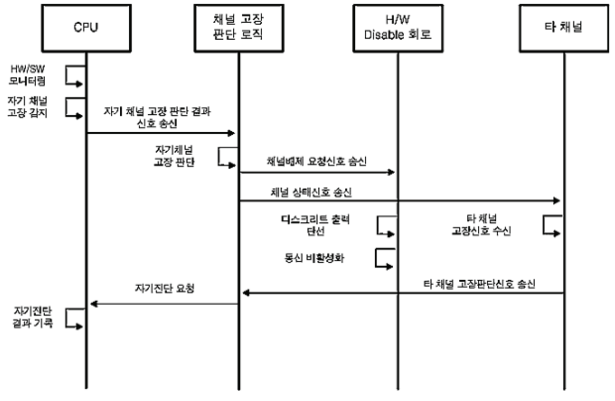

Fig. 5. Self-channel Fault Detection

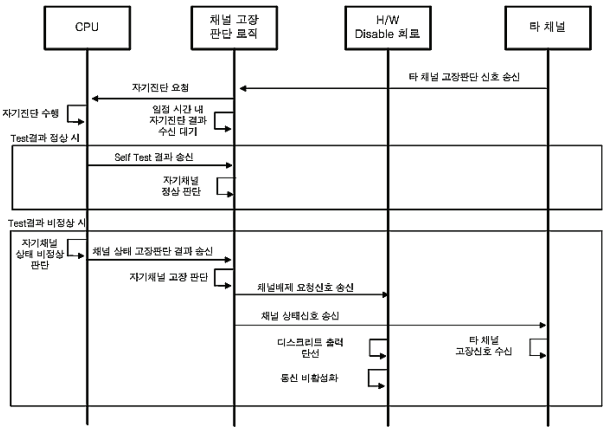

Fig. 6. Fault Detection from Other Channel

은 우선순위를 갖는 인터럽트 서비스 루틴을 이용하여 자체점검을 수행한다. 소프트웨어에서는 자체점검 수행 결과를 채널 고장 판단 로직으로 전송한다. 자신의 채널이 고장으로 판단되는 경우 자신의 채널을 고장상태로 전환하지만, 자신의 채널이 정상으로 판단되는 경우 자신의 채널 상태를 타 채널로 재송신하고 채널 고장 판단 로직에 의해 채널의 상태가 결정된다.

(3) 자체점검

비행조종컴퓨터의 고장이 판단되고 고장채널 식별이 되지 않은 경우 채널은 우선적으로 자체점검을 수행한다. 이때 수행하는 자체점검은 각 채널의 하드웨어 및 소프트웨어 오류를 판단하기 위하여 프로세서 내부 상태 레지스터 고장 이력 점검, 프로세서 코어 점검을 위한 명령어(instruction) 수행 검사, 하드웨어 고장 이력 점검, WDT 점검을 수행한다. 자체점검은 각 채널이 정상모드 동작 중이므로 실시간 기능 수행에 영향을 최소화할 수 있도록 주기적인 점검을 통한 상태 이력을 이용하여 두 채널 중 고장으로 판정되지는 않았지만 상대적으로 신뢰성이 더 낮은 채널을 판단할 수 있도록 설계하였다.

(4) 채널 고장 판단 로직(Fig. 7)

채널 고장 판단 로직은 비행조종컴퓨터 CPU 고장, CCDL 고장, 소프트웨어 오동작 등과 같이 이중화 구조에서 고장 검출은 가능하지만 고장 채널의 분리가 불가능한 경우 소프트웨어에서 수행한 자체점검 결과를 이용하여 고장 채널을 식별하고, 고장 채널의 분리를 위한 비행조종컴퓨터의 고장 채널에 대한 출력신호를 차단하기 위한 로직이다. 채널 고장 판단 로직은 단일 결함 점에 의한 비행조종시스템의 신뢰도 감소를 방지하기 위하여 채널 외부에서 수행하지 않고 양쪽 채널에서 함께 실행된다. 채널 고장 판단 로직의 기본 개념은 Fig. 7과 같다. 채널 고장 판단 로직은 소프트웨어에서 판단한 고장 판단 결과 정보, 주기적으로 모니터링하는 상태 신호를 이용한 고장 판단 결과 정보, 하드웨어 고장 판단 회로로부터 수

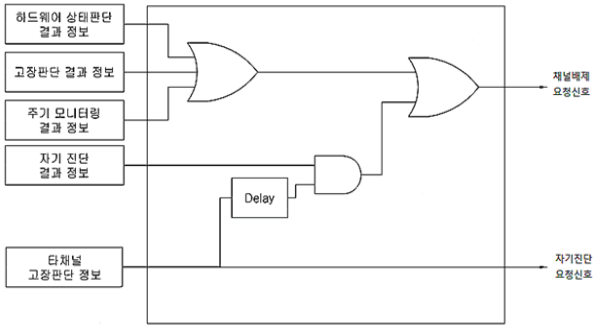

Fig. 7. Channel Fault Detection Logic

신된 하드웨어 상태 판단 결과 정보를 이용하여 고장 채널을 판단한다. 또한 타채널에서 수신된 고장판단 정보가 수신된 경우 펌웨어에서 생성된 자기진단 요청신호에 의해 소프트웨어에서 즉시 자체점검을 수행하여 결과를 채널 고장 판단 로직으로 전달하고, 채널 고장 판단 로직에서는 정해진 일정시간 내에 자체 점검결과를 수신하지 못하거나 자신의 채널에 이상을 판단한 경우 자신의 채널 상태를 고장으로 판단한다. 이때 자기 진단 결과정보 수신을 위한 본 설계에서의 최대 허용 시간은 3 프레임 주기 시간으로 설정하였다. 본 과제에서 채널 고장 판단 로직은 펌웨어로 구현하여 비행조종컴퓨터 비행운용 프로그램과 독립적으로 수행할 수 있도록 설계하였다.

2.2 이중화 비행조종컴퓨터 비행운용 프로그램 검증

2.2.1 이중화 비행조종컴퓨터 개요

비행조종시스템은 물리적으로 분리된 동일한 이중 채널로 구성되었으며, 각 채널은 비행조종컴퓨터와 각종 비행 센서 및 구동장치들이 연동되어 있다(Fig. 8). 비행조종컴퓨터의 각 채널은 CCDL를 통해 상대 채널이 수신한 데이터 및 상태 등을 교환한다.

이중화 비행조종컴퓨터의 비행운용 프로그램(OFP; Operational Flight Program)는 하드웨어의 제어 및 소프트웨어 운영 기능을 비롯하여 외부 장비와의 입

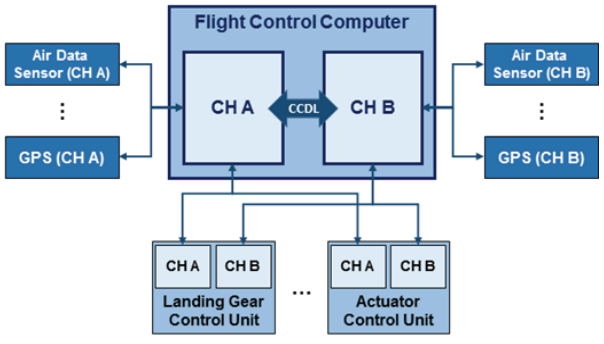

Fig. 8. Structure of Dual-redundant Flight Control System

Table 2. Main Software Functions of Operational Flight Program for Dual-redundant Flight Control Computer

분류	주요 기능
운영	▪ OFP 상태 관리 ▪ 실시간 운영 관리 ▪ Board Support Package 관리
외부 장비 연동	▪ 구성품 연동 기능 ▪ 연동장비 BIT 수행 기능 ▪ 연동결함 판단 기능
지상정비 및 결함관리	▪ 결함 관리 기능 ▪ 자체 점검 기능 ▪ FLCC BIT 기능 ▪ FLCC 상태 모니터링 기능 ▪ 아날로그/디스크리트 입출력 모니터링 기능
이중화 관리	▪ 이중화 채널 동기화 기능 ▪ CCDL 및 상태모니터링 기능 ▪ 연동장비 상태 모니터링 및 입력 선택 기능 ▪ 채널 결함 신호 출력 기능
제어기 연동	▪ 비행모드 관리 기능 ▪ 항로점 관리 기능 ▪ 비정상상황 통제 기능

출력 신호 처리, 연동 장비 및 비행조종컴퓨터의 상태 모니터링 기능 등을 포함한다. 이와 같은 기능들이 실시간으로 스케줄러에 따라 수행되어야 하며, 무인항공기의 비행 제어를 위한 제어법칙의 연산을 지원해야 한다. 이와 같은 독립 채널 운용을 위한 기능 외에도 이중화 채널 간의 동기화 및 데이터 교환 등의 기능을 수행할 수 있어야 한다. 이중화 비행조종컴퓨터 비행운용 프로그램의 주요 기능은 Table 2와 같다.

2.2.2 비행운용 프로그램 검증

비행조종컴퓨터 OFP는 고신뢰성이 요구되는 비행안전 필수(Flight Safety Critical) 소프트웨어로서 설계 및 검증단계에서 소프트웨어의 신뢰도와 안전성을 확보하기 위한 기술과 경험이 요구된다. 비행운용 프로그램이 요구사항을 충족함을 확인하기 위해서는 비행조종컴퓨터에 대한 개발단계에서 소프트웨어 통합시험, 시스템 통합검증 및 확인시험, 그리고 결함모드 검증 및 확인시험을 수행한다. 개발된 이중화 비행조종컴퓨터 OFP의 통합시험은 기능적 요구도를 검증하기 위해 Open-Loop의 정적 환경에서 하위 수준의 요구도별 기능시험을 수행한다. 소프트웨어 통합시험을 위한 주요 검증 기준은 정확성(Accuracy), 시간성, 초기화 상태, 결함 동작 및 리셋이다. 이러한 기준을 기본으로 소프트웨어 통합시험 수행을 위한 주요 시험항목들은 상태/모드 기능, 초기화 기능, 스

케쥴러 기능, 채널 동기화 기능, 입출력 관리 기능, 다중화 관리 기능, 고장관리 기능으로 구성된다. 개발된 이중화 비행조종컴퓨터 OFP의 기능을 검증하기 위해 상기의 통합시험 환경에서 요구도 기반의 기능시험 및 고장관리 기능시험을 수행하였다.

고장관리 기능의 검증은 비행조종컴퓨터 운용 중 요구되는 시간에 요구되는 고장을 발생시켜야 하므로 일반적으로는 실제 장비를 사용하지 않고 모델링을 통한 시뮬레이션 또는 해석적인 방법을 통해 수행한다. 그러나 이 경우 실제 무인기의 운용 중 발생할 수 있는 고장 상황에 대한 다양한 결과를 예측하기 어려워 본 과제에서는 실제 비행조종컴퓨터 하드웨어 및 소프트웨어를 이용하여 시험할 수 있도록 비행조종컴퓨터 외부 고장 및 내부 고장을 실시간으로 주입할 수 있는 검증환경을 이용하여 고장관리 기능을 검증하였다. 검증환경에 대한 간략한 기능은 2.2.3절에서 설명한다.

고장관리 기능을 검증하기 위하여 고장 주입은 외부 고장과 내부 고장으로 구분하여 주입하였다. 별도의 시험용 코드의 개발 없이 실시간으로 내부 변수의 상태를 모니터링하면서 고장 주입을 위해 사전에 정의한 상태가 발생하였을 때 고장주입을 위한 이벤트로 설정한다. 그리고 이벤트 발생 시 해당 변수의 값을 변경하는 것으로 고장을 주입하였다.

그리고 고장관리 기능에서 주입된 고장의 정확한 식별 및 고장에 따른 정확한 고장분리 기능을 수행하는지를 내부 변수 상태와 출력 신호 확인을 통해 판별하였다. 통합시험환경을 이용하여 수행한 비행조종컴퓨터 OFP의 요구도 기반 시험은 총 86개의 시험 케이스로 구성되었으며, 이중 본 논문에서 설명한 고장관리 기능 관련 시험은 23개의 시험 케이스를 수행하였다.

2.2.3 비행운용 프로그램 통합시험 환경

비행운용 프로그램 통합시험 환경은 이중화 비행조종컴퓨터 OFP의 고장관리 기능을 검증하기 위해 다음과 같은 기능이 필요하다[18,19].

(1) 입출력 신호 모사 및 모니터링 기능 : 외부 연동 및 입력에 따른 기능 검증
(2) 실시간 내부 상태 모니터링 기능 : 비탐침 실시간 디버깅을 통한 소프트웨어 신뢰성 향상
(3) 소프트웨어적 고장주입 기능 : 예외처리 기능 및 이중화 기능 검증

상기의 기능을 지원하기 위한 이중화 비행조종컴퓨터 OFP의 통합시험환경의 주요 요구사항은 Table 3과 같다. 본 과제에서는 이와 같은 요구사항을 기반으로 이중화 비행조종컴퓨터 비행운용 프로그램의 통합시험을 위한 소프트웨어 시험장비(STE; Software Test Equipment)를 제작하였으며, 기본 구성은 Fig. 9와 같다.

Table 3. Main Requirements of Integrated Software Test Environment

분류	주요 요구사항
하드웨어 구성 품목	▪ 통합시험환경 통제 PC ▪ 비행조종컴퓨터 모니터링 PC ▪ Trace 장비 및 운용 장비 ▪ 전원 공급 장비
입출력 인터페이스 기능	▪ Analog/Discrete 입출력 연동 ▪ 시리얼 통신 연동 ▪ MIL-STD-1553B 통신 연동 ▪ 이더넷 통신 연동
신호 모사 기능	▪ Analog/Discrete 입출력 신호 읽기/쓰기 ▪ MIL-STD-1553B RT-BC 쓰기 및 BC-RT 읽기
Trace 장비 제어	▪ 시작/중지 기능 ▪ 리셋 기능 ▪ 메모리 읽기/쓰기 기능 ▪ Breakpoint/Snapshot 기능

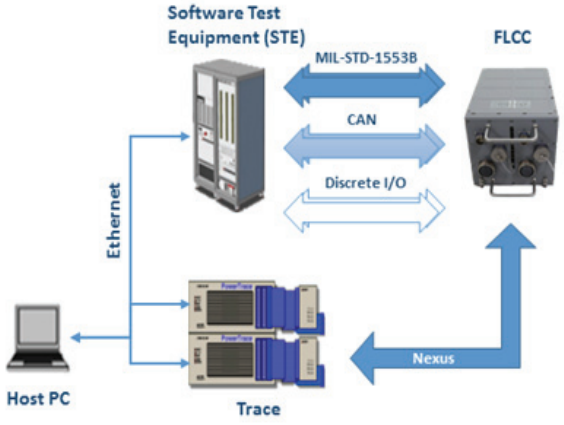

Fig. 9. Structure of Integrated Software Test Environment

시험환경에서 비행조종컴퓨터와 STE는 MIL-STD-1553B 등의 항공전자 데이터버스 및 이산 입출력 신호 등으로 연동되고, Nexus 인터페이스를 통해 Trace 장비를 연동하였다[20].

통합시험환경을 이용하여 고장관리 시험을 위한 고장주입을 위하여 비행조종컴퓨터 외부 고장은 입력신호 연동모델을 통해 주입하고, 내부 고장은 소프트웨어 고장 주입 기능을 활용한다. STE 장비는 Nexus 인터페이스로 연결된 Trace 장비를 통해 비탐침 실시간 모니터링으로 비행운용 프로그램의 내부 변수를 탐침 효과를 배제하면서 확인할 수 있다. 또한 메모리 내 임의 변수에 대한 쓰기 및 프로세서 고장 주입 등을 통하여 다양한 시나리오에 대한 고장관리 기능의 시험이 가능하다. 이처럼 STE와 Trace 장비를 활용한 소프트웨어 통합시험 환경을 구성하여 소프트웨어 구현 단계에서부터 효과적인 디버깅과 검증을 수행함으로써 개발된 소프트웨어의 신뢰성을 향상할 수 있도록 하였다.

2.2.4 고장관리 설계 검증 및 분석 결과

개발된 이중화 비행조종컴퓨터 고장관리 기능을 검증하기 위하여 2.2.3절에서 설명한 통합시험 환경을 이용하여 비행조종컴퓨터 독립적인 고장관리 기능에 대한 검증을 수행하였다. 그리고 항공기의 운용 중인 상황과 유사한 상황에서 이중화 비행조종컴퓨터의 실시간 고장관리 기능을 검증하기 위하여 비행조종 HILS(Hardware in the Loop Simulation) 환경에서 고장 영향성 확인 시험(FMET; Failure Mode Effects Test)을 통해 기능을 검증하였다.

고장관리 기능과 관련된 시험항목 및 판단 기준은 Table 4 및 Table 5와 같으며, 해당 항목들에 대한 검증을 완료하였다.

글로벌 호크의 고장관리 설계는 BIT 수행 결과를 통한 고장 검출과 외부 Switch logic을 이용한 고장 분리를 수행하며, 비행안전 신뢰도 요구가 200번의 임무 비행당 1번 이하의 고장 발생 요구도를 만족하면서 이중화 구조를 유지할 수 있도록 90% 수준의 BIT 커버리지를 가정하고 있다. 본 연구의 고장관리 설계를 적용한 이중화 구조에서 ST를 분석한 결과 비행조종컴퓨터 하드웨어 고장을 기준으로 고장 탐지율은 전원 공급모듈의 hold-up 모듈 기능을 제외한 30 항목에 대해 모두 고장 탐지가 가능하였고, 탐지된 고장에 대해서 고장 채널의 분리가 모두 가능하여 96.8% 수준의 ST를 확보함을 확인하였다.

Table 4. Test Result of FLCC Fault Management Function

고장관리 시험항목	판단 기준
채널 동기화 시험	안정화 후 동기화 시간 차이 < 100us
비행조종컴퓨터 CBIT 기능 시험	CBIT 항목별 고장진단
FLCC 채널 유효 모니터 기능 시험	타채널로 부터 고장 상태 수신시 자체 점검 수행 자체점검 수행 후 비정상 판단시 자신 채널 배제
연동 장비 상태 모니터 기능 시험	연동장비 상태에 따른 고장 판단 연동장비 고장판단에 따른 재형상 수행
FLCC 전원모니터 기능 시험	입력 및 내부 전원 고장판단 전원 고장시 채널 분리
FLCC 이산입력 모니터 기능 시험	이산입력 고장 판단 이산입력 고장 판단에 따른 재형상 수행
FLCC 이산출력 모니터 기능 시험	이산출력 고장 판단 이산출력 고장 채널 식별
FLCC 아날로그출력 모니터 기능 시험	아날로그출력 고장 판단 아날로그출력 고장 채널 식별
CCDL 상태 모니터링 기능 시험	CCDL 상태 모니터링 및 고장판단 고장채널 식별
고장 관리 기능 시험	고장 이력에 따른 자체점검 판단

Table 5. Test Result of HILS Failure Mode Effects

고장관리 시험항목	판단 기준
비행조종컴퓨터 이중화 연동 기능 시험	비행조종컴퓨터 외부 통신 고장에 따른 재형상 수행
대기자료 데이터 사용 우선 순위 시험	대기센서 고장에 따른 재형상 수행
탑재 장비 고장탐지 기능 검증 시험	비행조종계통 구성장비 고장 탐지
탑재 장비 고장 대처 기능 시험	비행조종계통 구성장비 고장에 따른 재형상 수행
FLCC 단일 결함 영향성 시험	비행조종컴퓨터 단일 채널 고장 분리 후 정상동작 수행

III. 결 론

무인기용 비행조종컴퓨터는 비행안전 필수 장비로 고장 발생 시 항공기의 손실로 이어질 수 있으므로 신뢰성 향상을 위해 다중화 구조를 이용한 고장허용 설계를 적용하고 있다. 하지만 이중화 구조를 채택한 비행조종컴퓨터는 고장 검출과 분리를 수행하는 고장관리 기능이 무인기의 신뢰성에 영향을 미치므로 그 설계와 검증은 중요하다. 특히 이중화 비행조종컴퓨터의 고장관리 설계에서 고장 검출, 시간 동기화 및 고장 분리 설계는 고장 검출뿐만 아니라 검출된 고장이 발생한 채널을 분리하여 무인기의 정상적인 운용을 지속하기 위해서 매우 중요하다.

본 논문에서는 무인기 이중화 비행조종시스템 신뢰성에 영향을 미치는 ST의 확률을 높이기 위해 고장 탐지율과 고장 분리율을 높이기 위한 고장관리 설계 방안을 제안하였다. 기본적으로 글로벌 호크에서 이중화 채널의 고장관리를 위해 외부에 별도로 고장 판단 및 선택 로직을 포함한 시스템이 가지는 단일 결함점 문제는 각 채널에서 고장관리 기능을 수행함으로써 해결하였다. 이때 항공기의 안전에 치명적인 영향을 미치는 비행조종컴퓨터 내부의 CPU 고장, CCDL 고장, 소프트웨어 오동작으로 인한 고장을 채널 고장 판단 로직을 이용하여 탐지하고 식별함으로써 신뢰성을 향상하였다.

제안하는 고장관리 설계가 반영된 비행조종컴퓨터 OFP의 검증을 위하여 소프트웨어 통합시험환경을 구성하여 정상 기능 및 고장관리 기능 요구도의 검증을 수행하고 실제 운용환경을 모사하는 HILS 환경에서 실시간으로 비행조종컴퓨터 내부 및 외부 고장

을 주입하여 고장관리 설계를 검증하였다. 기능 검증이 완료된 비행조종컴퓨터 OFP에 대하여 무인기 지상시험 및 비행시험을 통해 최종 기능과 성능을 확인하였으며, 현재 양산을 위한 규격화를 완료하였다.

References

1) Chang, W., Lee, S. K., Kim, Y. G. and Oh, T. G., "Autonomous Mission Management Software Design and Verification Technique for Unmanned Aerial Vehicles," *Journal of the Korean Society for Aeronautical and Space Sciences*, Vol. 49, No. 6, 2021, pp. 505~513.

2) Kim, Y. G., Chang, W. H., Kim, K. M. and Oh, T. G., "Development of an Autonomous Situational Awareness Software for Autonomous Unmanned Aerial Vehicles," *Journal of Aerospace System Engineering*, Vol. 15, No. 2, 2021, pp. 36~44.

3) Oh, T. G. and Yoon, H. S., "Development of Operational Flight Program for Dual-redundant Flight Control System of Unmanned Aerial Vehicle," *Proceeding of the Korean Society for Aeronautical and Space Sciences Fall Conference,* 2016, pp. 1276~1277.

4) Nam, Y. H., Joo, J. Y. and Jang, S. H., "Development of Dual-Redundant Flight Control computer for Tilt Rotor UAV," *Proceeding of the Korean Society for Aeronautical and Space Sciences Spring Conference,* 2013, pp. 1196~1199.

5) Yoon, H. S. and Kim, Y. G., "A Study on Software Based Fault-Tolerance Techniques for Flight Control Computer," *Journal of The Korean Society for Aeronautical and Space Sciences*, Vol. 44, No. 3, 2016, pp. 256~265.

6) Kim, Y. T., Yoo, H., Choi, I. H. and Park, M. H., "Study on a AFCS OFP Development plan of Domestic Small Helicopter for System Impact Minimization," *Proceeding of the Korean Society for Aeronautical and Space Sciences Spring Conference,* 2013, pp. 882~887.

7) Han, J. P. and Yoon, H. S., "Development of Triplex Flight Control Computer for Unmanned Aerial Vehicle," *Proceeding of the Korea Institute of Military Science and Technology Conference*, 2013, pp. 1465~1466.

8) Zhang, X., Li, H. and Yuan, D., "Dual Redundant Flight Control System Design for Microminiature UAV," *2015 2nd International Conf. on Electrical Computer Engineering and Electronics*, 2015, pp. 785~791.

9) Fourlas, G. K. and Karras, G. C., "A Survey on Fault Diagnosis and Fault-Tolerant Control Methods for Unmanned Aerial Vehicles," *Machines*, 2021.

10) Liu, B., Wu, J., Yao, L. and Ding, Z., "Ontology-based Fault Diagnosis : A Decade in Review," *Proceeding of the 11th International Conference on Computer Modeling and Simulation*, 2019, pp. 112~116.

11) Azadmanesh, A., Farahani, A. and Najjar, L., "Fault Tolerant Weighted Voting Algorithms," *International Journal of Network Security*, Vol. 7, No. 2, 2008, pp. 240~248.

12) Kim, S. S. and et al, "Verification of "Dual-master" Duplication Flight Control System using Simulink Virtual Module," *Journal of the Korean Society for Aeronautical and Space Sciences,* Vol. 36, No. 9, 2008, pp. 867~873.

13) Loegering, G., "Another Approach to Dual-Redundancy-The Global Hawk Experience," *AUVSI*, 2000.

14) Loegering, G. and Evans, D., "The Evolution of The Global Hawk & Mald Avionics Systems," *Proceedings of the 18th Digital Avionics Systems Conference*, October 1999.

15) Park, S. H., Kim, J. Y., Cho, I. J. and Hwang, B. M., "Redundancy Management Design for Triplex Flight Control System," *The Korean Society for Aeronautical and Space Sciences*, Vol. 38, No. 2, 2010, pp. 169~179.

16) MIL-HDBK-470A, *Designing and Developing Maintainable Products and Systems*, DOD Handbook, 1997.

17) Han, J. P. and Yoon, H. S., "Fault-tolerant apparatus and method in multi-computer for Unmanned Aerial Vehicle," Korea Patent, 10-1448013, 2014.

18) Lee, G. H., Hur, G. B. and Kim, Y. K., "Development of Integrated Monitoring Software and Processor Interface as Tools for Verification and Validation of Operational Flight Program in Flight Control Computer," *Proceeding of the Korean Society for Aeronautical and Space Sciences Fall Conference,* 2007, pp. 1175~1178.

19) Han, J. P. and Yoon, H. S., "Research on Software Integration Test Procedure and Result of Integrated Management Computer of Unmanned Aerial Vehicle," *Proceeding of the Korean Society for Aeronautical and Space Sciences Spring Conference*, 2015, pp. 638~641.

20) IEEE-ISO, *The Nexus 5001 Forum Standard for a Global Embedded Processor Debug Interface, Version 3.0*, 2012.