

관성항법장치가 포함된 비행체의 HWIL 시뮬레이션 시스템 개발 연구

이 아 영^{*,1)}

¹⁾ 국방과학연구소 제1기술연구본부

A Study on the HWIL Simulation System of the Flight Object including Inertial Navigation System

Ayeong Lee^{*,1)}

¹⁾ The 1st Research and Development Institute, Agency for Defense Development, Korea

(Received 27 November 2017 / Revised 27 April 2018 / Accepted 11 May 2018)

ABSTRACT

This paper proposes various methods for constructing a HWIL simulation system including Inertial Navigation System(INS) and Guidance Control Unit(GCU) under the assumption that the INS identifies the initial attitude of an aviation body through its own alignment and that it is a package consisting of an inertial sensor and a navigation computation module. This paper also presents a real-time computing technology and a way to calculate the command of the Flight Motion System(FMS) analogous to the actual flight environment. The proposed HWIL simulation system is constructed by applying the above-mentioned methods and the results of running a series of simulations confirm high effectiveness and usefulness of the system. Finally, minor error factors that could be acquired only in HWIL simulation Environment are analyzed.

Key Words : HWIL Simulation, Inertial Navigation System(관성항법장치), Flight Motion System(비행자세모의기)

1. 서론

항공기, 유도탄, 함정 등 민간 및 군용으로 사용되는 비행체 또는 운항체들은 자신의 현재 위치를 감지하여 목적지까지 정확히 도착하는 임무를 수행하도록 개발된다. 이들은 여러 가지 미션을 수행하는 내장형

시스템으로 설계된 다양한 부체계들로 구성되며, 각각의 부체계는 통합되어 하나의 복합적인 시스템으로 구성된다.

특히, 비행체 시스템들은 정확한 자신의 위치를 측정하기 위해 관성항법장치를 공통적으로 사용한다. 이 장치는 가속도계와 각속도계를 장착한 관성센서(IMU: Inertia Measurement Unit)를 사용하여 각속도와 가속도를 측정 후, 이 신호들을 처리하여 비행체의 위치와 자세각을 추정한다^[1]. 보통은 관성센서의 출력 데이터

* Corresponding author, E-mail: aylee@add.re.kr
Copyright © The Korea Institute of Military Science and Technology

를 적분하여 위치와 자세각을 추정하는 순수항법을 사용하여 자신의 위치를 추정하지만 시간이 지남에 따라 기계적 자이로 스코프의 마찰에 따른 드리프트 등의 영향으로 항법오차가 발생하기 때문에 이를 보정하기 위해 GPS와 같은 보조 센서의 출력과 칼만 필터(Kalman Filter)를 이용한 운항 중 정렬 등의 방법을 함께 사용하기도 한다^{11,12}.

유도탄, 항공기와 같이 관성항법장치, 유도조종장치 및 기타 내장형 부체계가 포함된 복잡한 시스템을 개발함에 있어서 비용 및 기간 등은 중요한 고려 요인이기 때문에, 비용 절감을 위해 컴퓨터 시뮬레이션¹³을 포함한 다양한 시험평가 기법들이 적용되게 되는데, HWIL(HardWare-In-the-Loop) 시뮬레이션도 폭넓게 사용되는 유용한 시험평가 방법 중의 하나이다^{14,15}. HWIL 시뮬레이션은 실물 하드웨어와 소프트웨어로 모델링된 하드웨어의 동작환경(플랜트)을 구현한 후 실시간 시뮬레이션을 수행하여 실물 하드웨어의 성능을 검증하는 시험평가 방법이다. 하드웨어가 포함된 환경에서 비파괴적이고 반복적인 시험환경에서 테스트를 수행하여 인적 및 물적 비용을 절감할 수 있으므로 시험 전 성능 예측, 실패한 시험의 사후 분석, 실제 시험이 불가능 환경의 시스템 성능 예측의 목적으로 다양한 응용 분야에서 광범위하게 사용된다.

항공기, 유도무기와 같이 관성항법장치가 포함되는 시스템들도 성능을 평가하기 위해 HWIL 시뮬레이션을 사용하게 되는데 HWIL 시뮬레이션 시스템을 구성할 때는 관성항법장치가 실제로 동작할 수 있는 환경과 관성항법장치와 연결되는 장비연동 환경을 고려하여 구축해야 한다. 일반적으로 컴퓨터 시뮬레이션 환경과 시스템 부체계, 환경 모의 장비를 연동하여 구현하게 된다. 컴퓨터 시뮬레이션 환경은 실시간 컴퓨터, 실시간 운영체제^{16,17}와 같은 IT 기술들을 이용하여 시스템이 동작하는 플랫폼을 실시간으로 모의하는 실시간 시뮬레이션으로 구현되고, 관성항법장치와 같은 측정장비는 비행자세모의기(FMS: Flight Motion Simulator)와 같은 동작 환경 모의 장비를 사용하여 시스템의 회전운동을 모의한다. 각 장비들의 연동은 다양한 아날로그, 디지털 통신 기술을 이용하여 구성된다.

HWIL 시뮬레이션 환경의 개념은 앞서 설명한 바와 같지만, 실제로 구축될 때는 관성항법장치의 구성, 시스템의 실제 동작 환경, 시험 평가 활용 목적, 구현의 복잡도 및 소요 비용 등에 따라 다른 형태로 개발될 수 있다. 이 논문은 관성센서와 연산 모듈이 하나의

패키지로 합쳐져 두 모듈 사이의 인터페이스를 인터셉트하기 어렵고, 자체 정렬을 통해 비행체의 초기 자세각을 알아내는 관성항법장치라는 가정 하에, HWIL 시뮬레이션 시스템을 구성하기 위한 다음과 같은 기술들을 제시한다.

- HWIL 시뮬레이션 시스템 구성 framework
- 실시간 시뮬레이션 시스템 구성 기술
- 비행체 자세 모의 기법
- HWIL 시뮬레이션 오차요인 분석

본 논문의 구성은 다음과 같다. 2장에서 HWIL 시뮬레이션 환경 및 시스템을 구성하기 위한 구조를 제시하고 3장에서는 관성항법장치가 포함된 HWIL 시스템의 구성을 위한 framework를 제시한다. 그리고 4장에서는 실시간 시뮬레이션 환경을 구현하기 위한 실시간 컴퓨팅기술에 대해 논하며, 5장에서는 비행체의 운동모의 관련하여 실험실 환경의 제약성과 그 해결 방안을 제시한다. 6장에서는 실제 HWIL 시뮬레이션을 수행한 후 획득한 결과를 통해 제안한 시스템 및 방법의 효용성을 보이며, 7장에서 결론을 맺는다.

2. HWIL 시뮬레이션 환경 및 시스템 구성 방법

HWIL 시뮬레이션은 실물 하드웨어의 성능을 확인 검증하는 목적으로 수행되기 때문에 실물 하드웨어의 동작환경을 충실히 모의하는 것이 중요하다. 관성항법장치가 포함된 비행체의 HWIL 시뮬레이션을 수행하기 위해서는 관성항법장치가 정상적으로 동작할 수 있는 환경을 모의하고 그 출력 자료를 비행체의 유도조종장치로 입력하도록 구성해야 한다. 즉, 관성항법장치는 비행체의 병진운동과 회전운동을 측정하는 장비이기 때문에, 비행환경과 동일하게 동작하기 위해서는 비행체의 두 운동을 지상에서 물리적으로 모의해야 한다. 그러나 비행체 회전운동의 경우 비행자세모의기에 탑재하여 모의할 수 있지만 비행체의 병진운동(가속도)은 지상에서 물리적으로 모의하기는 어렵다. 따라서 관성항법장치를 포함한 환경에서 비행체 유도조종 성능을 확인하기 위한 HWIL 시뮬레이션 시스템을 구현하기 위해서는 병진운동 자료를 모의하고 그 출력 자료를 관성항법장치가 측정한 회전운동 자료와 통합할 수 있는 방법이 필요하다.

일반적으로 비행체를 개발할 때, 몬테카를로 시뮬레이션을 통해 성능 확인을 목적으로 시뮬레이션 프로그램이 개발되며, 이 프로그램 내부에는 비행체 동작환경을 모의하는 비행체 6 자유도 운동방정식과 함께 관성항법장치와 유도조종장치를 모델링 한 소프트웨어가 함께 포함된다. 따라서 HWIL 시뮬레이션 시스템을 구성하는데 있어 병진운동 자료(위치, 속도, 가속도)는 시뮬레이션 프로그램에 포함된 관성항법장치 소프트웨어 모델의 출력자료를 사용하고, 회전운동과 관련된 자세각(오일러 각 및 쿼터니언), 각속도는 비행자세모의기에 탑재된 관성항법장치의 출력값을 이용할 수 있다.

위에 언급한 내용들을 고려하여 관성항법장치가 포함된 환경에서 HWIL 시뮬레이션을 수행할 수 있는 시스템으로 Fig. 1과 같이 관성항법 자료 통합 및 통신 모듈, 실시간 시뮬레이션 모듈, 비행자세모의기, 자세각 변환기가 포함된 페루프 시스템을 제안한다.

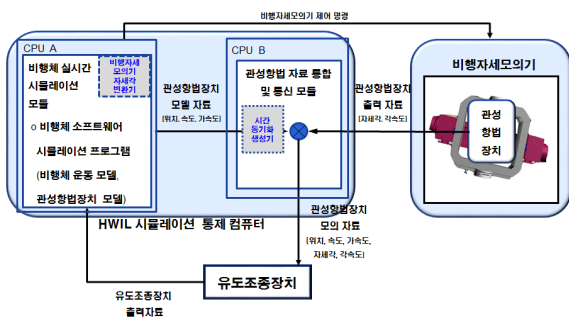


Fig. 1. HWIL simulation system structure

Fig. 1의 비행체 실시간 시뮬레이션 모듈에는 실시간 운영체제를 이용한 내장형 시스템 기술을 사용하여 비행체의 동작환경을 모의하는 비행체의 운동 모델(6 자유도 운동방정식)과 함께 관성항법장치 모델을 포함하는 비행체 소프트웨어 시뮬레이션 프로그램이 탑재된다. 비행체 소프트웨어 시뮬레이션 프로그램은 비행체의 위치, 속도 및 가속도 정보인 병진운동 자료를 실시간으로 생성하고, 동시에 비행체의 회전운동 자료를 생성하기 위해 비행자세모의기에 제어명령을 인가하며, 생성된 병진운동 자료를 관성항법장치자료 통합 및 통신 모듈로 전송한다.

관성항법장치자료 통합 및 통신 모듈은 비행자세모의기에 탑재된 관성항법장치로부터 생성된 비행체의 회전운동 자료와 비행체 실시간 시뮬레이션 모듈의 소

프트웨어로부터 계산된 병진운동 자료를 실시간으로 통합하여 한 개의 통합 프레임으로 구성하고, 구성된 통합 프레임을 관성항법장치가 사용하는 통신 주기에 맞추어 비행체 유도조종장치에 전송한다.

비행자세모의기 자세각 변환기는 비행자세모의기를 사용하여 비행체 비행자세를 모의함에 있어 실험실에서 발생할 수 있는 제약사항을 해결하여 비행자세모의기 제어 명령을 생성한다. 실험실에 설치된 FMS는 세 개의 회전축을 가지는 시스템으로 실험실에 고정되기 때문에 설치 방향을 회전시킬 수 없는 한계로 인해 비행체의 360° 전방위 비행운동을 모의하는데 제약이 따른다. 이 모듈은 이러한 제약사항을 해결하여 비행체의 운동을 모의하기 위한 FMS 제어 명령을 생성한다.

3. 관성항법 자료 통합 및 통신 모듈

비행체 실시간 시뮬레이션 모듈과 관성항법 자료 통합 및 통신 모듈의 구현 위치에 따라 Table 1과 같이 네 가지 방법으로 시스템을 구성할 수 있다.

구성1은 비행체 실시간 시뮬레이션 모듈을 HWIL 시뮬레이션 통제컴퓨터 내의 프로세서 A에 할당하고, 관성항법 자료 통합 및 통신 모듈을 프로세서 B에 할당하여 병렬 처리하는 방법이다. 관성항법 자료 통합 및 통신 모듈에서 처리된 통합 데이터(통합 프레임)가 유도조종장치로 전송될 때 존재하는 상호 간 통신 시간 지연(Latency)을 감소시키고, 통신 데이터 발생 주기의 지터링(Jittering)이 발생하지 않는 장점이 있다. 단점으로는 프로세서간의 통신을 위한 별도의 프로그래밍이 필요하다는 점이다.

구성2는 HWIL 시뮬레이션 통제컴퓨터 내의 프로세서 한 개를 사용한다. 대신 두 개의 태스크를 생성하여 태스크 A에 비행체 실시간 시뮬레이션 모듈을 할당하고, 다른 태스크 B에 관성항법 자료 통합 및 통신 모듈을 할당하는 방법이다. 관성항법 자료 통합 및 통신 모듈에서 처리된 통합 데이터가 비행체 유도조종장치로 전송될 때 존재할 수 있는 상호 간 통신 시간 지연을 감소시키고, 시스템 구성 비용을 최소화시킨다. 단점으로는 멀티태스킹을 위한 프로그래밍 로드가 증가하고, 태스크 간의 스케줄링 방식에 따라 유도조종장치로 출력되는 통합 데이터의 통신주기가 흔들리는 현상(지터링)이 발생할 수 있다.

구성3은 HWIL 시뮬레이션 통제컴퓨터에 비행체 실시간 시뮬레이션 모듈을 할당하고, 별도의 컴퓨터에 관성항법 자료 통합 및 통신 모듈을 할당하는 방법이다. 이 방법은 별도 컴퓨터 시스템 선정 및 구현이 용이/유연하며, 구성1과 같이 관성항법장치 통신 자료의 전송 주기의 지터링이 발생되지 않는 장점이 있다. 그러나 단점으로는 HWIL 시뮬레이션 통제컴퓨터와 관성항법 자료 통합 및 통신 모듈 컴퓨터 간에 데이터를 송수신하기 위한 추가적 외부 통신이 필요하고, 상호 간 통신 시간 딜레이를 증가할 수 있다.

구성4는 HWIL 시뮬레이션 통제컴퓨터에 비행체 실시간 시뮬레이션 모듈을 할당하고, 유도조종장치에 관성항법 자료 통합 및 통신 모듈을 구현하는 방법이다. 이 방법은 통신을 모의하는 추가적인 객체를 사용하지 않기 때문에 통신 시간 지연을 최소화하고, 시스템을 간소화시킴으로 인해 비용절감의 효과가 있다. 그러나 HWIL 시뮬레이션을 수행하기 위해 유도조종장치에 탑재된 별도의 프로그램을 작성해야 하는 단점을 갖는다.

관성항법장치의 모듈 구성 및 HWIL 시뮬레이션 시스템의 구성 환경, 소요 비용 및 시스템 환경에 따라 위에서 제시한 방법들 중에 선택하여 구성하면 되며, 본 논문에서는 구성1안을 선택하여 비행체 실시간 시뮬레이션 모듈은 HWIL 시뮬레이션 통제 컴퓨터의 CPU A(프로세서A)에 할당하고, 관성항법 자료 통합 및 통신 모듈은 HWIL 시뮬레이션 통제 컴퓨터의 CPU B(프로세서B)에 할당하여 HWIL 시뮬레이션 시스템을 구성하였다.

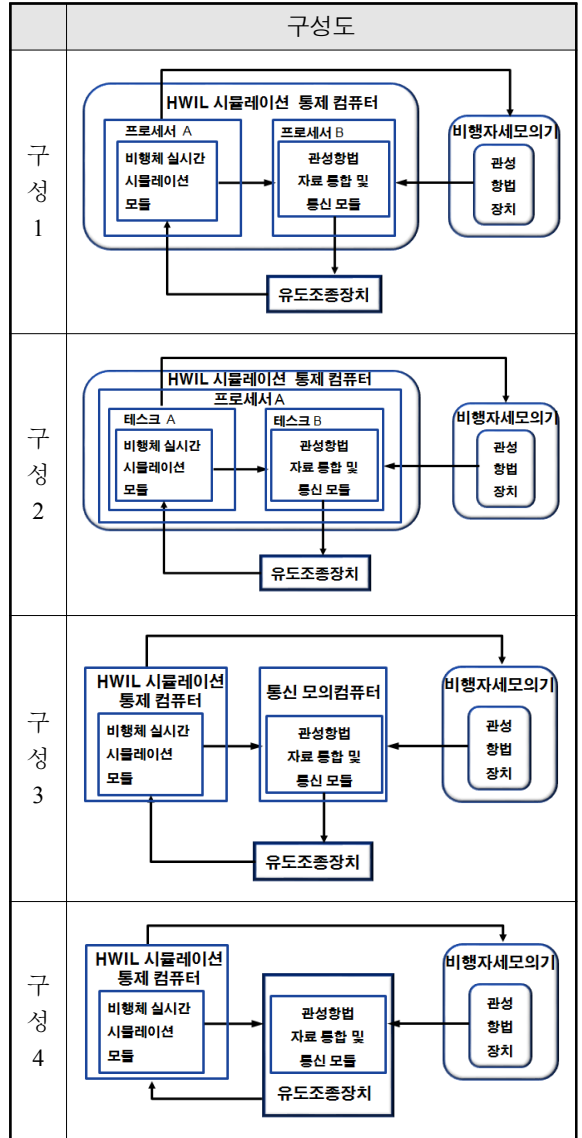
4. 비행체 실시간 시뮬레이션 모듈

4.1 실시간 컴퓨팅

비행체 소프트웨어 시뮬레이션 프로그램은 비행체의 동작환경을 모의하는 비행체의 운동 모델(6 자유도 운동방정식)은 일반적으로 미분방정식 형태로 주어지고, 유도조종 루프 내의 상태방정식과 함께 관성항법장치 모델을 포함한다. 통상적으로 사용되는 적분방식으로는 RK(Runge-Kutta) method, AB(Adams-Bashforth) method, Euler method 등의 방식이 있다^[8]. 적분방법에서 고려하여야 할 사항은 정확도(Accuracy)와 실시간성인데, 이 둘은 상호보완(trade-off) 관계를 가진다. 시뮬레이션의 정확도를 높이기 위해 적분스텝 크기를 줄

이고 고차 적분방정식을 선택하면 되지만 실시간 시뮬레이션에서 연산 속도를 증가시키는 관점에서 실시간성을 보장받기 어렵게 된다.

Table 1. The methods of HWIL simulation system construction



시뮬레이션의 정확도를 높이기 위해 적분스텝 크기를 줄이고 고차 적분방정식을 선택하면 되지만 실시간 시뮬레이션에서 연산 속도를 증가시키는 관점에서 안정적이지 못하게 된다.

주지하는 바와 같이 RK4방식은 AB2방식에 비하여 연산시간이 많이 소요되므로 실시간 연산 목적으로 많이 사용되지 않으나 프로세서의 연산 속도가 빨라져 연고자하는 시뮬레이션 프레임 속도에 크게 문제가 없다면 이용도 가능하지만 빠른 연산속도를 위하여 AB2나 Euler method와 같은 수치해석 알고리즘을 많이 사용한다.

일반적으로 시뮬레이션 진행 시간을 시스템 타임(system time)이라고 부르며 다음 단계의 시스템 타임은 적분간격(step time, ΔT)을 더함으로써 구해진다. 소프트웨어 시뮬레이션을 수행할 때 시스템 타임은 실제 세계의 시간과 무관하게 동작한다. 즉, 시뮬레이션의 한 단계가 증가할 때마다 현재의 시스템 타임에 적분간격을 더하여 시스템 타임이 갱신됨으로써 시뮬레이션을 종료할 때까지 소요되는 시간은 시뮬레이션을 수행하는 시스템의 성능에 종속적이다.

그러나 HWIL 시뮬레이션은 비행체의 부체계들과 연동하여 시뮬레이션을 수행하기 때문에 비행체의 운동방정식 연산도 실시간으로 이루어져야 한다. 여기에서 실시간이란 운동방정식을 연산하는 시뮬레이션의 진행시간(시스템 타임)이 실제 세계의 시간의 경과와 같음을 의미한다.

편의상 실제 세계의 시간과 무관한 적분간격을 논리적 스텝 타임이라고 부르고, 실제 세계와 동기화된 적분간격을 물리적 스텝 타임이라고 표현하며, 한 번의 시뮬레이션 루프를 연산하는데 소요되는 시간을 프레임 타임이라고 표현한다. 실시간 시뮬레이션은 6자유도 운동방정식 해석을 위해 적절한 적분간격을 지정해야 하고, 적분간격은 물리적 스텝 타임이 되어야 한다.

통신 측면에서 HWIL 시뮬레이션은 실제 관성항법장치가 포함된 환경에서 해당 장비와 통신을 수행하며 시뮬레이션을 진행해야 하기 때문에 통신상의 시간지연을 최소화시키는 관점에서는 적분스텝을 줄이는 것이 유리하다. 그러나 실시간 컴퓨터의 연산속도 측면에서는 적분스텝을 늘리는 것이 구현성 측면에서 유리하므로 두 파라미터를 고려하여 적절한 스텝타임과 적분방법을 찾아야 한다.

비행체 실시간 시뮬레이션 모듈은 실시간 컴퓨팅 기술을 이용하여 물리적 스텝타임을 구현하고, 비행체 소프트웨어 시뮬레이션 프로그램이 적분시간 안에 연산될 수 있도록 개발되어야 한다. 또한 다양한 컴퓨터공학적인 방법을 적용하여 실시간성과 연산정확도

를 충족시키는 것이 중요하다. 본 논문에서는 실시간 운영체제를 비행체 실시간 시뮬레이션 모듈을 구현하기 위해 적용된 물리적 스텝타임 구현 방법과 GPS 보정항법을 사용하는 관성항법장치 모델의 실시간 연산을 위해 적용된 방법 및 통신시스템 구현 방안을 제시한다.

4.1.1 실시간 클럭 제어

실시간 시뮬레이션을 구현하기 위해서는 운동방정식의 적분간격을 물리적 스텝 타임과 일치시켜야 하며, 이를 위해 실시간 클럭을 제어해야 한다. 실시간 클럭 제어는 비행체 실시간 시뮬레이션 모듈과 관성항법 자료 통합 및 통신 모듈에서 모두 사용하며, 타이머 인터럽트와 이진 세마포어를 이용하여 구현할 수 있다. Fig. 2와 같이 시뮬레이션 태스크는 세마포어를 생성한 후, Wakeup될 때까지 대기한다. 타이머 인터럽트의 인터럽트 서비스 루틴(ISR: Interrupt Service Routine)은 타이머 인터럽트에 의해 호출될 때마다 세마포어를 Wakeup 시킨다. 시뮬레이션 태스크는 타이머 인터럽트가 호출될 때마다 Wakeup되므로 물리적 시간의 흐름에 동기화하여 시뮬레이션을 진행할 수 있다.

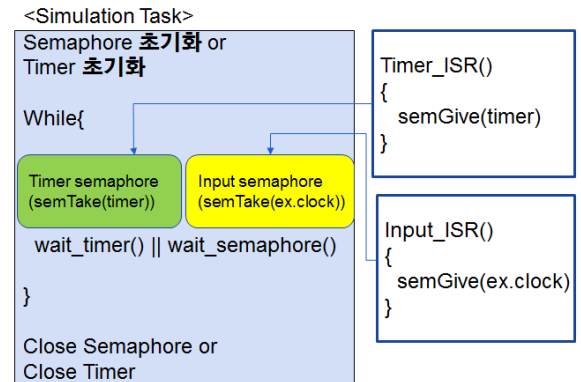


Fig. 2. Real-time clock control using semaphore

실시간 클럭 제어의 또 다른 방법은 외부 클럭을 이용하는 방법이다. 외부에서 통신 자료가 주기적으로 입력된다면, 입력될 때마다 ISR이 동작하도록 설정한 후, ISR에서 시뮬레이션 태스크의 세마포어를 Wakeup 시킬 수 있다. 즉, 통신 자료가 주기적으로 입력된다면 이 자료의 입력주기에 맞추어 ISR을 설정하고 세마포어 Wakeup 명령을 통해 시뮬레이션 태스크를 동작시킬 수 있다. 이러한 기능은 관성항법 자료 통합

및 통신 모듈에서 관성항법장치 통신을 모의하기 위한 목적으로 사용된다.

4.1.2 실시간성 보장을 위한 멀티태스킹

HWIL 시뮬레이션의 실시간성을 보장받기 위해서는 물리적 스텝타임마다 비행체 모델의 연산이 수행되어야 하고 프레임 타임이 물리적 스텝 타임보다 작도록 관리되어야 한다. 하지만 GPS 보정항법을 수행하는 관성항법장치가 포함된 비행체와 같이 복잡한 HWIL 시뮬레이션의 경우 물리적 스텝타임 내에 연산할 수 없는 모델이 존재한다. 일반적으로 GPS 및 보정항법 모델의 경우 연산시간이 길기 때문에 실시간성 보장을 위해서는 비교적 큰 값의 물리적 스텝타임을 사용하면 되지만 이는 시뮬레이션의 정확도가 낮아지는 결과를 초래할 수 있다. 이를 해결하기 위해서 멀티태스킹을 적용하여 한 개의 연산장치 내에서 태스크를 분할하여 한 개의 태스크가 종료된 후 남는 시간에 다른 태스크를 연산하는 방법을 사용할 수 있다.

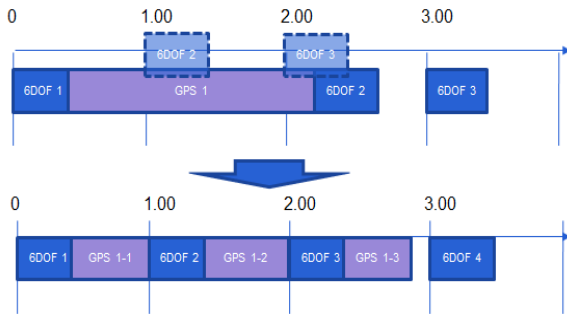


Fig. 3. Example of multitasking

예를 들어, Fig. 3에서 소프트웨어 시뮬레이션 프로그램이 GPS 및 보정항법 모델을 제외한 프로그램(이하 6DOF)과 GPS 및 보정항법 모델(이하 GPS)로 구성되어 있을 때, GPS는 비록 1초에 한 번 연산이 수행되지만 한 번 수행될 때 소요되는 연산시간이 상당히 길기 때문에 GPS를 태스크로 분리하여 6-DOF의 여유 시간에 연산하도록 수정할 수 있다. 적분간격이 1.0 msec이고 GPS 연산에 1.9 msec가 소요되고 6-DOF 연산에 0.3 msec가 소요된다고 가정할 때, Fig. 3의 상단과 같이 모든 연산을 순차적으로 수행하면 GPS 연산 때문에 적분간격 1.0 msec에 맞추어 연산(총 연산시간은 2.2 msec)을 수행할 수 없다. 이런 문제를 해결하기 위해 Fig. 3의 하단처럼 GPS 연산을 6-DOF 연산이

끝난 후 남는 시간에 여러 번 연산하도록 수정하면 실시간으로 소프트웨어 시뮬레이션 프로그램을 연산할 수 있다.

멀티태스킹을 수행하기 위해서는 태스크를 분리하여 실행할 수 있어야 하고, 태스크의 수행에 우선순위를 줄 수 있어야 하며, 우선순위가 낮은 태스크가 동작하는 중간에 우선순위가 높은 태스크의 요구가 들어오면 그 태스크를 먼저 처리할 수 있는 선점형 우선순위 스케줄링(Preemptive priority scheduling) 기법을 제공해야 한다.

4.2 인터페이스 환경

4.2.1 통신 인터페이스

통신 인터페이스는 비행체 실시간 시뮬레이션 모듈 내부의 연산 보드들 사이의 자료 통신과 관성항법장치와 같은 외부 장비와의 통신으로 구분할 수 있다. 비행체 실시간 시뮬레이션 모듈이 병렬처리를 사용할 경우 실시간 운영체제에서 제공하는 메시지 큐, 파일프, 공유메모리와 같은 방법을 사용하여 모듈간의 자료통신과 통신동기화 기법을 설계하는 것이다. 독립적으로 동작하는 각 모듈의 실행 주기를 제어하지 않으면 모듈들은 비동기로 동작하여, 소프트웨어 모델 및 유도조종 제어루프 동작순서 및 결과에 영향을 미치기 때문에 모듈 사이의 동기화하는 메커니즘이 필요하다.

외부장비와의 통신은 아날로그, 이산신호, RS-422등 다양한 통신 메커니즘을 사용하여 비행자세모의기, 관성항법장치, 유도조종장치와 실시간으로 자료를 송수신하는 것을 목적으로 한다. 다양한 통신 기능을 가지는 보드들이 사용되며 CPU와 I/O 보드들이 VME(Versa Module Europa)버스 또는 PCI(Peripheral Component Interconnect)버스를 사용하여 하나의 시스템으로 통합하는 것이 유리하다.

4.2.2 시간동기화 생성기

위에서 언급한 바와 같이 HWIL 시뮬레이션 시스템은 다양한 통신 인터페이스로 구성되어 있기 때문에 이로 인한 통신지연을 항상 고려해야 한다. 본 논문에서 제안하는 HWIL 시뮬레이션 시스템은 관성항법 자료 통합 및 통신 모듈에서 자료를 통합함에 있어서, 모의된 병진운동 자료와 관성항법장치에서 측정된 회전운동 자료 사이의 시간 편차(Δt)가 존재하는데 이를 보상하여 두 자료를 동기화시켜 주어야 한다. 병진

운동의 경우 소프트웨어 시뮬레이션 프로그램에서 계산되기 때문에 시뮬레이션 시간 흐름상에서 현재 시점(Tc)의 비행체 운동을 반영한다. 그러나 관성항법장치에서 측정되는 회전운동의 경우, 비행자세모의기를 제어하고 그 회전운동을 관성항법장치가 측정하기 때문에 현재 시점(Tc)에서 생성된 비행자세모의기 제어 명령은 일정한 시간이 지난 뒤에 회전운동 측정값으로 반영된다. 두 자료의 동기화 문제는 HWIL 시뮬레이션 환경에서만 발생하는 문제이기 때문에 비행환경을 충실히 모의하기 위해서는 두 자료를 동기화시키는 방법이 필요하다.

관성항법 자료 통합 및 통신 모듈에 시간동기화 생성기를 추가로 구성함으로써 이를 해결할 수 있다. 기존의 관성항법 자료 통합 및 통신 모듈은 비행체의 회전운동 및 병진운동 자료를 실시간으로 통합하여 한 개의 통합 프레임으로 구성하고, 구성된 통합 프레임을 관성항법장치가 사용하는 통신 주기에 맞추어 비행체 유도조종장치에 전송한다. 반면에 시간동기화 생성기가 포함된 관성항법 자료 통합 및 통신 모듈은 비행체 실시간 시뮬레이션 모듈과 관성항법장치로부터 수신한 데이터를 식 (1)에 따라 데이터 간 생성시간을 보상하여 한 개의 통합 프레임을 구성하고, 통합 프레임을 유도조종장치로 전송함으로써, 비행체의 병진운동 및 회전운동 자료를 통합할 때 발생하는 데이터 간 동기화 문제를 극복할 수 있다.

$$\Delta_L = F_D + I_D \tag{1}$$

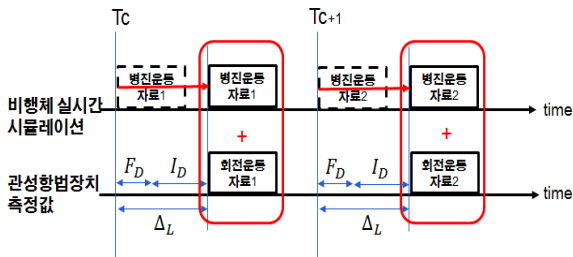


Fig. 4. Time synchronization

즉, Fig. 4와 같이 시간동기화 생성기가 포함된 관성항법 자료 통합 및 통신 모듈은 비행자세모의기를 구동하는데 소요되는 시간(F_D)과 실제 관성항법장치에서 회전운동을 측정하고 자료를 출력하는데 소요되는 시간(I_D)을 미리 측정한 후, 병진운동의 자료를 식

(1)에서 계산된 시간(Δ_L)만큼 지연시켜 통합한 후, 유도조종장치로 출력하면 된다.

5. 비행자세모의기 자세각 변환기

5.1 비행자세모의기 구동 명령 계산법

비행체의 회전운동은 비행자세모의기를 통해 모의하게 되는데 평가받는 비행체 시스템의 동특성에 따라 전기식 또는 유압식의 비행자세모의기가 선택되어야 하고, 세부 성능을 확인하여 시험 평가의 목적에 맞게 적당한 성능의 비행자세모의기가 선택되어야 한다⁹⁾. 일반적으로 HWIL 시뮬레이션에 사용되는 비행자세모의기는 피치, 요, 롤 순서의 오일러 각을 사용하여 제어를 하고, 관성항법장치는 주로 Local Level의 NED좌표계 또는 ENU좌표계 기준으로 동작한다. 따라서 관성항법장치가 실제 비행환경과 동일한 환경에서 동작하도록 모의하기 위해서는 비행체의 소프트웨어 시뮬레이션 프로그램에서 실시간으로 계산된 다양한 기준좌표계의 자세각을 사용하여 관성항법장치 좌표계 기준의 피치, 요, 롤 오일러 순서를 가지는 자세각으로 변환하여 비행자세모의기를 구동해야 한다. 비행체 실시간 시뮬레이션에서 계산된 자세각이 발사대 중심 기준좌표계라고 가정하고, 관성항법장치는 NED좌표계를 사용한다고 가정했을 때, 발사대중심 기준좌표계를 NED좌표계 기준으로 방향코사인행렬((DCM: Directional Cosine Matrix))을 구한 후 이를 활용하여 비행자세모의기의 구동명령인 오일러 각을 계산한다. 발사대 중심 좌표계는 발사점을 기준으로 표적 방향을 X축으로 사용하는 관성좌표계를 의미한다.

$$C_{LNED}^B = (C_{IE}^{LNED} \cdot C_I^{IE} \cdot C_L^I \cdot C_B^L)^T \tag{2}$$

Table 2. Definition of the directional cosine matrix

C_{LNED}^B	Local Level NED to Body DCM
C_B^L	Body to Platform DCM, independent Earth Spin
C_L^I	Platform to Inertial DCM
C_I^{IE}	Inertial to Spin Inertial DCM
C_{IE}^{LNED}	Spin Inertial to Local Level NED DCM

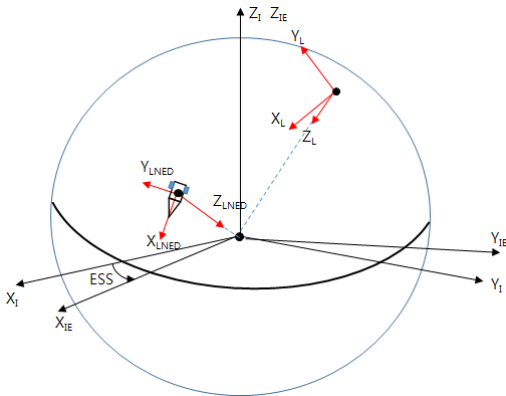


Fig. 5. Coordinate system

C_A^B 의 형태로 표현되는 식은 A좌표계에서 B좌표계로의 좌표변환을 나타내고, 이를 방향코사인행렬이라고 한다. 비행자세모의기의 구동 자세각을 구하기 위하여 Table 2에서 정의한 각각의 방향코사인행렬과 식 (2)를 이용하여 C_{LNED}^B 방향코사인행렬을 구한 후, 식 (3)을 통해 피치, 요, 롤 순서의 오일러 각을 구한다.

비행자세모의기를 제어하기 위해서는 각속도와 각 가속도가 함께 필요하므로 식 (4), 식 (5) 방법으로 각 속도와 각가속도를 구하고 이 자료들을 비행자세모의기 제어 콘솔로 전달한다.

$$\begin{aligned} FMS_{rollPos} &= \text{atan}(-C_{LNED}^B[2][1], C_{LNED}^B[1][1]) \\ FMS_{pitchPos} &= \text{atan}(-C_{LNED}^B[0][2], C_{LNED}^B[0][0]) \\ FMS_{yawPos} &= \text{asin}(C_{LNED}^B[0][1]) \end{aligned} \quad (3)$$

$$\begin{aligned} FMS_{rollRate} &= (FMS_{rollPos}(t_n) - FMS_{rollPos}(t_{n-1})) / \Delta t \\ FMS_{yawRate} &= (FMS_{yawPos}(t_n) - FMS_{yawPos}(t_{n-1})) / \Delta t \\ FMS_{pitchRate} &= (FMS_{pitchPos}(t_n) - FMS_{pitchPos}(t_{n-1})) / \Delta t \end{aligned} \quad (4)$$

$$\begin{aligned} FMS_{rollAcc} &= (FMS_{rollRate}(t_n) - FMS_{rollRate}(t_{n-1})) / \Delta t \\ FMS_{yawAcc} &= (FMS_{yawRate}(t_n) - FMS_{yawRate}(t_{n-1})) / \Delta t \\ FMS_{pitchAcc} &= (FMS_{pitchRate}(t_n) - FMS_{pitchRate}(t_{n-1})) / \Delta t \end{aligned} \quad (5)$$

5.2 전방위각 사격 모의 기법

5.2.1 전방위각 사격 모의 제약사항

5.1절의 방법으로 오일러각을 구하면 비행자세모의기 구동명령을 계산할 수 있다. 하지만 이와 같은 방법은 실험실에 비행자세모의기가 설치된 특정 방위각 비행환경만 모의가능하다는 단점이 있다. 다시 말하자면, 일반적으로 비행자세모의기는 실험실의 한 지점에

고정되어 독립적인 좌표계(이하 FMS 좌표계)를 사용한다. 하지만 소프트웨어 시뮬레이션 프로그램은 비행체의 방위각에 제한이 없이 시뮬레이션을 수행하고 다양한 기준좌표계를 가진다. 따라서 비행자세모의기를 사용하여 비행체의 회전운동을 표현하기 위해서는 각축의 회전 순서와 좌표계를 고려하여 실시간 시뮬레이션 컴퓨터에서 계산하는 비행체의 자세각을 비행자세모의기의 자세각과 매핑시키는 방법을 고려해야 한다.

예를 들어, Fig. 6과 같이 실험실의 비행자세모의기는 진북방향을 중심으로 11.5도 회전되어 설치되었고 가정한다면 실험실에서의 비행체 기준 좌표계는 비행자세모의기가 설치된 위치가 발사점이 되고 북쪽을 중심으로 11.5도 회전된 방향이 X축이 되는 좌표계로 고정된다. 따라서 사격방위각이 11.5도인 시뮬레이션 환경에서 유도탄의 오일러 각을 사용하여 비행자세모의기를 구동하고 관성항법장치의 자세각을 측정하면 사격방위각과 비행자세모의기의 방위각이 정확히 일치하기 때문에 정상적인 시뮬레이션을 수행할 수 있다.

하지만 사격방위각이 11.5도가 아닌 180도 환경일 경우, 실제 비행환경에서는 관성항법장치 측정값이 사격방위각에 따라 변하게 되지만, 실험실 환경에서는 비행자세모의기가 설치된 방위각에 따라 항상 일정한 값을 가지게 된다. 즉, FMS를 임의의 실제 사격방위각만큼 물리적으로 돌려 설치하지 않고서는 NED 좌표계를 사용하는 관성항법장치로부터 올바른 출력값을 얻을 수 없다.

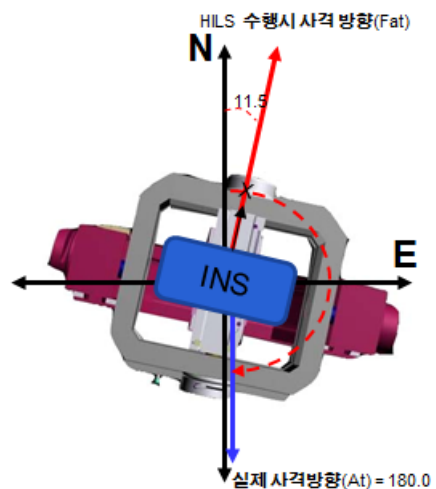


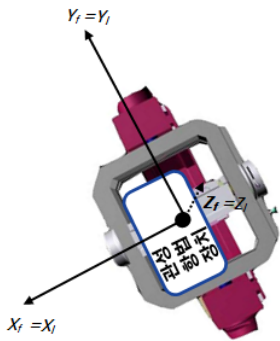
Fig. 6. INS mounted in FMS(fixed at a specific location)

그러므로 발사점 중심의 표적 방향을 X축으로 사용하는 좌표계를 사용하는 경우, 임의의 발사 방위각을 가지는 전 방향의 HILS를 수행하기 위해서는 사격 방위각과 오일러 각을 모두 고려하여 관성항법장치의 출력값을 실제 비행환경과 일치시킬 수 있는 모의 방법이 필요하다.

이 논문에서 제안하는 기법은 회전운동을 모의함에 있어 비행체 회전운동을 표현하는데 사용되는 기준 좌표계(비행체의 소프트웨어 시뮬레이션 프로그램에서 사용)와 비행자세모의기의 회전운동을 표현하는데 사용되는 비행자세모의기 좌표계(비행자세모의기 설치 방향에 고정) 사이의 불일치로 인해 발생하는 비행체 사격방위각 제한 문제를 비행자세모의기의 자세각을 변환하여 극복하는 방법이다.

5.2 전방위 사격모의를 위한 자세각 변환기

비행자세모의기 자세각 변환기를 구현하는 기술은 Fig. 7과 같다. 비행자세모의기를 구동하는 자세각은 비행체 소프트웨어 시뮬레이션 프로그램에서 계산된 관성좌표계 기준의 오일러 각, 두 좌표계의 차이(δ) 각, 비행체의 사격방위각을 함께 고려하여 비행자세모의기를 회전시키는 방법이다. 이 방법은 마치 비행자세모의기가 비행체 사격방위각 방향으로 설치된 것처럼 구동하도록 각 축의 구동 자세각을 새롭게 구하는 방법이다. 이 방법을 구현하기 위해 사격방위각과 관성좌표계의 오일러각을 연동시키는 방법은 식 (6)과 같다. C_I^B 는 관성좌표계를 비행체 동체 좌표계와 일치시키는 방향코사인행렬이며, C_F^I 는 비행자세모의기 좌표계를 관성좌표계와 일치시키는 방향코사인행렬이다.



비행자세모의기 자세각 변환기

Fig. 7. Attitude angular converter using FMS

$$CFB = C_I^B C_F^I \tag{6}$$

사격방위각이 고려된 비행자세모의기의 새로운 오일러 각은 CFB 방향코사인행렬을 이용하여 구하며, 이때 비행자세모의기 구동축의 오일러 순서 정의에 따라 식 (7)과 같이 피치, 요, 롤 순서로 구할 수 있다.

$$\begin{aligned} FMS_{rollPos} &= \text{atan2}(-CFB[0][2], CFB[0][0]) \\ FMS_{yawPos} &= \text{asin}(CFB[0][1]) \\ FMS_{pitchPos} &= \text{atan2}(-CFB[2][1], CFB[1][1]) \end{aligned} \tag{7}$$

앞에서와 동일한 방식으로 비행자세모의기를 제어하기 위해서는 각속도와 각가속도가 함께 필요하므로 식 (4), 식 (5) 방법으로 각속도와 각가속도를 구하고 이 자료들을 비행자세모의기 제어 콘솔로 전달하면 실제 비행환경과 동일한 사격방위각으로 비행하는 비행체를 모의할 수 있다.

6. 시스템 적용 결과

6.1 비행체 실시간 시뮬레이션 모듈 결과

위에서 제시한 다양한 기법들을 적용하여 개발된 HWIL 시스템의 성능을 검증한 결과를 제시하고자 한다. 먼저 구성1안을 적용하여 HWIL 시스템을 구성하였고, 이 때 사용한 CPUA(프로세서A), CPUB(프로세서 B)는 1.3 GHz 클럭, 1 GByte 메모리를 장착한 MVME7100보드 2개를 사용하였다. 소프트웨어 시뮬레이션 프로그램은 AB2적분방식을 사용하였고 적분간격은 1 msec으로 지정하였다. 또한 실시간성을 보장받기 위하여 실시간 클럭 제어, 멀티태스킹을 적용하였고, 충실하고 효과적인 관성항법장치 모의를 위하여 병렬 처리 및 전방위 사격 모의 기법을 적용하였다.

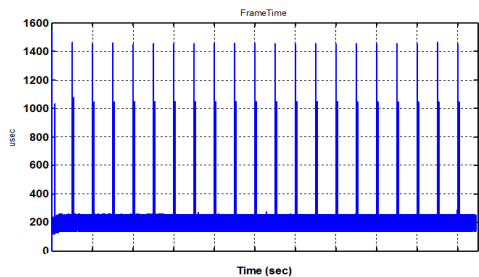


Fig. 8(a). Frame Time before multitasking

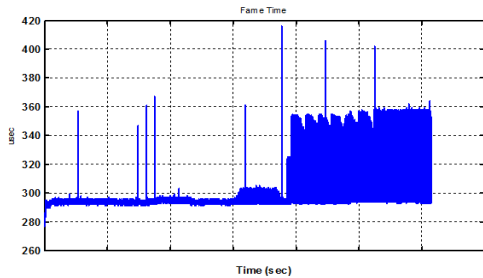


Fig. 8(b). Frame Time after multitasking

Fig. 8은 HWIL 시뮬레이션의 프레임 타임 결과이다. Fig. 8(a)는 연산 소요 시간이 일정하지 않고 특정시간에 많은 시간이 소요되어 HWIL 시뮬레이션이 실시간성을 만족하지 못함을 보여준다. 이러한 문제를 극복하기 위해 연산 시간이 긴 GPS 및 보정항법 모듈을 독립적인 태스크로 분리한 후 멀티태스킹으로 처리하도록 설계하였고, 그 결과 Fig. 8(b)와 같이 전 구간에 걸쳐 연산 소요시간이 적분간격보다 작음을 확인할 수 있다.

6.2 비행자세모의기 자세각 변환기 모듈 결과

앞에서 제시한 자세각 변환기를 관성항법장치자료 통합 및 통신 모듈에 탑재하여 임의의 사격방위각으로 HWIL 시뮬레이션을 수행하였다. 변환된 관성항법장치의 각속도 출력을 비행체의 소프트웨어 시뮬레이션 프로그램의 항법 모델에서 연산된 각속도 출력자료와 비교하였다. Fig. 9는 비행체의 롤, 피치, 요 각속도를 비교한 결과인데 두 그래프가 거의 유사함을 확인할 수 있다. 임의의 사격방위각이 아닌 FMS 방위각에 맞추어 사격방위각을 조정하여 HWIL시뮬레이션을 수행해도 비슷한 수준의 차이는 발생함을 확인하였다. 따라서 제시된 방법이 HWIL 시뮬레이션을 수행함에 문제가 없음을 확인할 수 있다.

6.3 HWIL 시뮬레이션 추가 오차 고찰

Fig. 9 그래프의 차이는 HWIL 시뮬레이션 환경, 부체계 모델과 실제 하드웨어의 차이에 의한 영향으로, 관성항법장치가 포함된 HWIL 환경은 다음과 같은 오차요인을 가진다.

- FMS 동특성에 의한 시간 지연
- 관성항법장치 자료 출력 시간 지연
- 관성항법장치 장착 오차

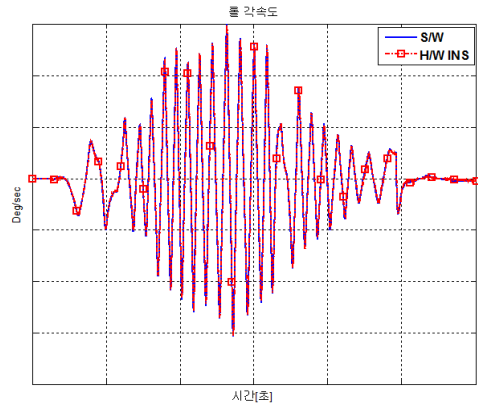


Fig. 9(a). Roll angle velocity after using attitude angular converter using FMS

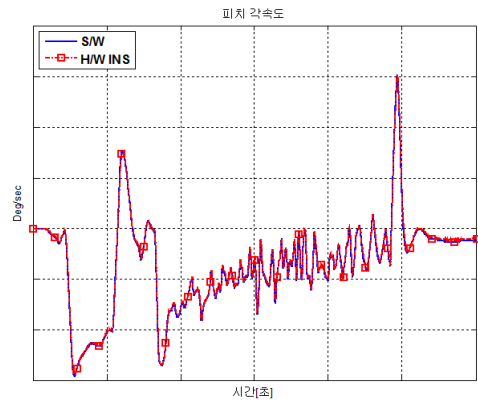


Fig. 9(b). Pitch angle velocity after using attitude angular converter using FMS

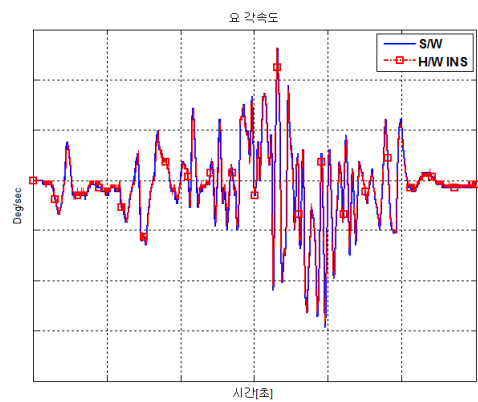


Fig. 9(c). Yaw angle velocity after using attitude angular converter using FMS

비행체의 자세를 모의해주는 FMS도 하나의 제어시스템이기 때문에 구동하는데 시간이 소요된다. 소요되는 시간으로는 HWIL 시뮬레이션 통제 컴퓨터가 FMS 제어 콘솔과 통신하는데 소요되는 시간과 FMS 제어 콘솔이 실제 테이블을 구동하는데 소요되는 시간이 포함되며, 이것들이 FMS 동특성에 의한 시간 지연이라 할 수 있다.

또한 관성항법장치 자료 출력 시간 지연은 관성항법장치의 각속도 센서가 FMS의 기동을 측정한 후, 그 결과를 유도조종장치로 출력하는데 걸리는 시간을 의미한다.

Fig. 10은 HWIL 환경에서 발생할 수 있는 시간지연을 측정한 그래프이다. 정현파로 FMS를 구동한 후, FMS 명령, FMS 구동응답신호, 관성항법장치 출력 값을 비교하였다. FMS 동특성에 의한 시간지연은 대략 3~5 msec이고, 본 논문의 결과 검증에 적용된 관성항법장치는 FMS 동특성에 의한 시간지연을 포함하여 자료 출력 시간 지연은 약 20 msec 정도임을 확인할 수 있다.

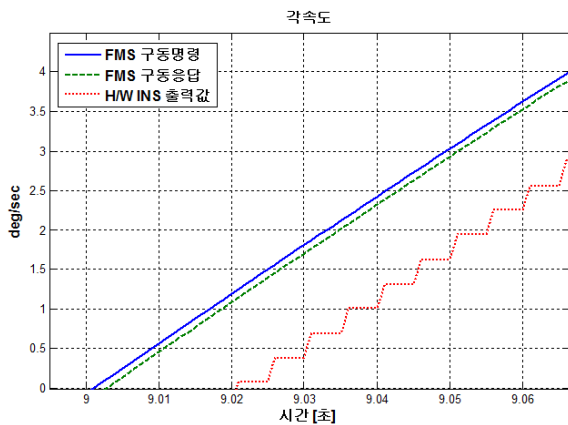


Fig. 10. Time delay for the FMS dynamic characteristics and INS output data

관성항법장치 장착오차는 관성항법장치를 장착할 때마다 달라지지만 대략 0.5도를 넘지 않는 범위에서 측정된다. 관성항법장치의 장착각 오차의 영향성을 확인하기 위해 관성항법장치에서 출력되는 자세각 자료에 식 (8)과 같이 장착오차를 보상하여 HWIL 시뮬레이션을 수행하였다. C_I^B 는 관성좌표계를 비행자세모의기 좌표계와 일치시키는 방향코사인행렬이며, C_F^B 는

관성항법장치 장착각 오차를 이용하여 비행자세모의기 좌표계를 비행체 동체 좌표계와 일치시키는 방향코사인행렬의 역행렬이다.

$$CIB = C_F^B C_I^F \tag{8}$$

관성항법장치 장착각오차가 고려된 비행자세모의기의 새로운 오일러 각은 CIB 방향코사인행렬을 이용하여 구하며, 이 때 비행자세모의기 구동축의 오일러 순서 정의에 따라 식 (9)과 같이 피치, 요, 롤 순서로 구할 수 있다. 즉, 관성항법장치로부터 측정된 장착각오차를 반영하여 비행자세모의기 구동명령을 계산함으로써, 비행자세모의기에 탑재된 관성항법장치는 장착각오차가 없는 것처럼 자세각, 각속도 값을 출력하게 된다.

$$\begin{aligned} FMS_{rollPos} &= \text{atan2}(-CIB[0][2], CIB[0][0]) \\ FMS_{yawPos} &= \text{asin}(CIB[0][1]) \\ FMS_{pitchPos} &= \text{atan2}(-CIB[2][1], CIB[1][1]) \end{aligned} \tag{9}$$

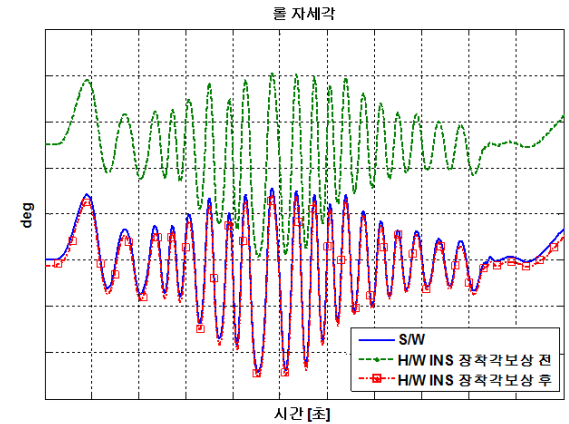


Fig. 11. Roll angle after mounting error compensation

Fig. 11은 관성항법장치의 장착각 오차 요인을 보상한 HWIL 시뮬레이션 결과 가운데 롤 자세각 결과이다. 비행체의 소프트웨어 시뮬레이션 프로그램의 항법 모델에서 연산된 출력값과 장착각오차를 보상하기 전과 후의 관성항법장치 출력값을 비교한 그래프로, 장착각 오차 보상 후의 결과가 비행체의 소프트웨어 시뮬레이션 프로그램의 항법 모델에서 연산된 출력값과 더 유사함을 확인할 수 있으며, 장착각오차가 HWIL 시뮬레이션 환경 오차요인임을 알 수 있다.

위에서 분석한 세 가지 오차요인 가운데 FMS 동특성에 의한 시간 지연은 HWIL 시뮬레이션 환경에 존재하는 전형적인 오차요인이며 나머지는 실제 비행환경에서도 존재할 수 있는 요인이다. 위의 요인들은 자세각과 각속도의 진동 또는 제어루프의 발산 등 유도조종 알고리즘에 크게 영향을 미치기 때문에 HWIL 시뮬레이션 시스템이 개발된 후 정량적인 오차요인 분석이 필요하다. 또한 HWIL 시뮬레이션 환경에서만 발생하는 오차요인과 실제 비행환경에서도 영향을 주는 오차요인인지에 대한 식별이 반드시 필요하다.

7. 결 론

HWIL 시뮬레이션은 비행체 하드웨어성능 및 유도조종성능을 확인하기 위해 수행되는데, 시험의 신뢰도를 높이기 위해서는 실제의 비행환경을 충실하게 모의하는 것이 중요한 요소이다. 비행환경의 충실한 모의에는 비행체의 유도조종과 관련된 중요한 구성품인 관성항법장치와 유도조종장치가 포함한다는 의미와 함께 실제 비행체 환경에서 각 구성품이 출력되는 자료와 유사한 자료들을 HWIL 시뮬레이션 환경에서 출력할 수 있다는 의미를 포함한다.

본 논문에서는 관성항법장치가 HWIL 시뮬레이션 환경에 포함되는 경우 발생하는 병진운동을 모의할 수 없는 물리적인 한계와 자료를 통합할 때 발생하는 시간 동기화 문제를 극복할 수 있는 해결책을 제시하였다. 또한 실시간 컴퓨팅과 관련된 내용으로 GPS 및 보정 항법 등의 모델들의 연산시간이 길 때 적분간격(ΔT) 시간 내에 연산할 수 있는 방법을 제시하였고, 통신 인터페이스와 관련된 내용으로 관성센서와 연산모듈이 통합된 관성항법장치를 사용하여 HWIL 시뮬레이션 시스템을 구성할 때 병진운동을 모의하기 위한 컴퓨터 시스템 구성 및 인터페이스 방법을 제시하였다. 마지막으로 비행체의 운동모의 관련하여 발사점 중심 관성좌표계로 표현되는 비행체의 자세를 비행자세모의기로 모의하는 방안을 제시하였으며, 제시된 방법들은 실제 시스템을 구현한 후 관성항법장치를 탑재한 HWIL 시뮬레이션을 수행한 결과를 토대로 실시간성 및 비행자세모의 정확성을 검증하여 HWIL 시뮬레이션에 적용될 수 있음을 확인하였다.

References

- [1] O. J. Woodman, "An Introduction to Inertial Navigation," Technical Report, UCAN-CL-TR-696, Univ. of Cambridge, Aug. 2007.
- [2] Y. M. Kwon, "GPS/INS Fusion Using Multiple Compensation Method Based on Kalman Filter," Journal of the Institute of Electronics and Information Engineer, Vol. 52, No. 5, May 2015.
- [3] D. M. Curry, "The Synergistic Roles of Test and Simulation In the Life Cycle of the Army TACMS Missile System," Proceedings of SPIE Vol. 7663, 2010.
- [4] R. S. Karvande, B. R. Kummar, "Development of Hardware-In-Loop Simulation Testbed for testing of Navigation System-INS," International Conference Machine Intelligence Research and Advancement, 2013.
- [5] R. M. Gravitz, W. F. Waite, "Validation Methodologies for Complex, Hybrid, HWIL, 6DOF Missile Simulations - A Structured Approach -," Proceedings of the 1988 Winter Simulation Conference, 1988.
- [6] D. T. Winters, J. C. Garcia, T. M. Neighoff, "Development of a Kinetic Warhead Hardware-In-the-Loop Simulation," Proceedings of SPIE Vol. 4366, 2001.
- [7] W. Waite, S. J. Swenson, A. C. Jolly, C. S. Shepherd, and R. Gravitz, "Validation of Hardware-In-the-Loop(HWIL) and Distributed Simulation Systems," Foundations for V&V in the 21st Century Workshop, Oct. 2002.
- [8] S. D. Conte, C. D. Boor, Elementary Numerical Analysis An Algorithm Approach, McGraw-Hill, pp. 362-376, 1980.
- [9] R. W. Mitchell, "Selection Criteria For Hydraulic and Electric Motion Simulators," Proceedings of SPIE Vol. 5408, Bellingham, WA, 2004.