

論文

J. of The Korean Society for Aeronautical and Space Sciences 46(11), 969-973(2018)

DOI:https://doi.org/10.5139/JKSAS.2018.46.11.969

ISSN 1225-1348(print), 2287-6871(online)

항전 비행운용프로그램 검증을 위한 단일 PC 기반 소프트웨어 시험환경 SW 설계

차상철*, 이두환**, 김정열***

Design of STE SW Running on a Single PC to Verify Avionics OFP

Sang-Cheol Cha*, Du-Hwan Lee** and Jeong-Yeol Kim***

Avionics R&D Lab, LIG NEX1 Co. Ltd, Korea

ABSTRACT

Avionics OFP runs on the mission computer and can be operated by interacting with several avionics equipments. In order to verify OFP SW, SIL having real avionics equipments or models is absolutely necessary. Therefore in many cases SIL is implemented concurrently with OFP developing, and only one SIL is provided to developers. So developers sometimes need an alternative to SIL for verifying requirements in the middle of development process. In this paper, we propose a single PC based STE SW that simulates interworking equipments and verifies OFP in a single PC environment without actual interworking equipments or SIL HW interfaces.

초 록

항공전자 비행운용프로그램은 임무컴퓨터에서 실행되면서 각종 항전장비와의 연동을 통해 운용된다. 비행운용프로그램을 검증하기 위해서는 임무컴퓨터와 연동하는 실장비가 탑재된 SIL 또는 STE가 필수적이거나 일반적으로 특정 항공기 체계개발 시 사용되는 SIL은 OFP 개발 일정과 상당 부분 중첩되므로 OFP 개발과정 후반부 또는 SW 통합 시점에 SIL이 개발자에게 제공되며 개발과정 중 개발자가 SW를 검증하기 위한 도구가 제공되지 않는 경우가 많다. 대부분의 경우 SIL은 한 세트만 제작하기 때문에 다수의 개발자가 하나의 SIL을 이용해서 SW 검증을 해야 하므로 시험단계에서 병목현상이 발생할 수밖에 없어 개발 일정 준수에 위험요소로 작용한다. 본 논문에서는 실장비가 탑재된 SIL 또는 STE 대신 개발자 PC와 임무컴퓨터만을 이용하여 OFP 기능을 검증할 수 있는 STE SW 설계에 대해 기술한다.

Key Words : OFP(Operational Flight Program, 비행운용프로그램), STE(Software Test Environment, 소프트웨어 시험환경), SIL(System Integration Lab, 체계통합실험실)

† Received : September 6, 2018 Revised : October 19, 2018 Accepted : October 23, 2018

* Corresponding author, E-mail : sangcheol.cha@lignex1.com

I. 서론

항공기 체계 개발 과정의 일부인 항공전자 비행운용프로그램(이하 OFP, Operational Flight Program) 개발은 요구도 분석 및 도출, 기본/상세 설계, 구현 과정을 거쳐 SW 단위시험, SW 통합시험, 체계 통합 시험을 통해 검증된다. OFP 검증과정에서 단위시험에는 Unit Test 틀이 사용되고, SW 통합시험을 위해 STE(Software Test Environment)가 제공되어야 하며, 체계 통합시험을 위해 SIL(System Integration Lab)이 필요하다. 특히 효과적으로 OFP를 포함한 항전 시스템 검증을 위해 SIL이 반드시 필요하기 때문에 이와 관련된 연구가 진행되었다[1,2].

그러나 일반적으로 특정 항공기 체계개발 시 사용되는 SIL은 OFP 개발 일정과 상당 부분 중첩되므로 OFP 개발과정 후반부 또는 SW 통합 직전에 SIL이 개발자에게 제공되며 개발과정 도중에 개발자가 SW를 검증하기 위한 도구가 제공되지 않는 경우가 많다. 따라서 SW 통합시점이 되어서야 SIL을 이용해서 실장비 연동을 비롯한 SW 검증을 할 수 있다. 대부분의 경우 SIL은 한 세트만 제작하기 때문에 다수의 개발자가 하나의 SIL을 이용해서 SW 검증을 해야 하므로 시험단계에서 병목현상이 발생할 수 밖에 없어 개발 일정 준수에 위협요소로 작용한다.

또한 SIL은 SW 통합시험이 완료된 후 개발자 사이트에서 통합시험 장소로 이동하는 경우도 있어 통합시험 이후 요구도가 변경되거나 결함 수정이 필요한 경우 개발자가 SIL 설치 장소에 가서 OFP를 검증해야 하는 불편함이 있다.

본 논문에서 제안하는 단일 PC 기반 STE SW는 SIL에 비해 제한적인 SW 검증만 가능하지만 별다른 HW 없이 개발자 PC만을 이용하여 HW 제어 및 인터페이스를 제외한 OFP 기능을 검증할 수 있다는 점에서 기존 연구와 차이가 있다. OFP 기능 중 HW 제어 기능은 OFP 개발 초반에 우선적으로 검증되고 빈번하게 변경되는 요구도의 대부분은 OFP 운용 로직에 집중된다. 따라서 본 논문에서 제안하는 STE SW를 이용한 OFP 검증 대상이 OFP 운용 로직에 한정되지만 전체 개발 주기에 걸쳐 OFP 기능을 효과적으로 검증할 수 있다는 장점이 있다. 그리고 SIL 제작에 필요한 다양한 HW 인터페이스나 장비가 필요 없으며, 임무컴퓨터와 개발자 PC를 이더넷으로 연결함으로써 SW 통합 전에 충분히 SW를 검증하여 SW 통합 소요시간을 단축할 수 있다.

II. 본론

2.1 VxWorks 기반 OFP 구조

일반적으로 VxWorks를 이용하여 OFP를 개발할 경우 Fig. 1의 임무컴퓨터와 같이 I/O 처리 및 HW 인터페이스 제어 기능은 DKM(Downloadable Kernel Module)에서 구현하며 OFP의 주요 로직은 RTP(RealTime Process)에서 구현된다. DKM과 RTP의 메모리 공간은 분리되어 있으므로 공유메모리를 통해서만 상호 연동할 수 있다.

임무컴퓨터에 탑재되는 OFP는 DKM에서 디바이스 드라이버를 이용해 외부로부터 수신한 데이터를 공유메모리에 복사한다. RTP는 공유메모리에서 DKM이 복사한 데이터를 읽어 와서 OFP의 로직에 따라 처리한 후 임무컴퓨터 외부로 송신할 데이터를 다시 공유메모리에 복사한다. DKM은 RTP가 공유메모리에 복사한 데이터를 HW 인터페이스를 통해 임무컴퓨터 외부로 송신하는 기능을 수행한다.

각 HW 인터페이스별 송수신 메시지는 Fig. 2와 같이 공유메모리의 특정 영역에 매핑되어 있으며, RTP는 공유메모리를 통하여 임무컴퓨터 외부와 연동하므로 RTP 입장에서는 공유메모리 너머에서 어떤 인터페이스를 이용하여 메시지가 외부와 연동되는지 알 필요가 없다.

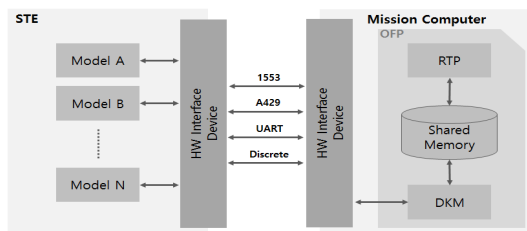


Fig. 1. Simplified software test environment

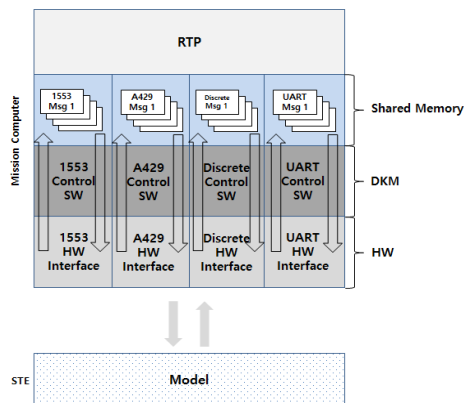


Fig. 2. Data flow between STE and MC

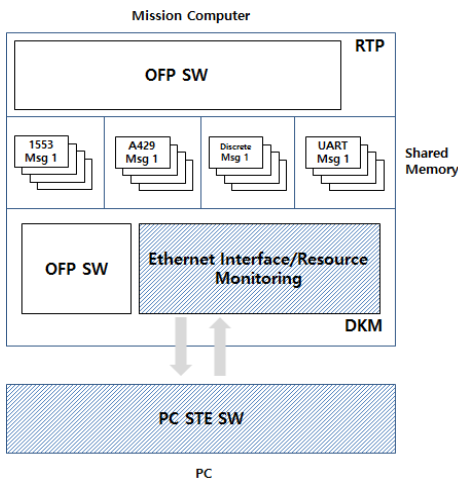


Fig. 3. Structure of STE SW

2.2 STE SW 설계

STE SW는 Fig. 3과 같이 임무컴퓨터에서 동작하는 테스트 에이전트와 PC에서 동작하는 PC STE SW로 구성된다.

2.2.1 테스트 에이전트 설계

테스트 에이전트는 OFP가 MIL-STD-1553, ARINC 429, Discrete 등의 실제 항공전자 버스 HW 인터페이스를 통해 데이터를 송수신하는 기능을 이더넷을 통해 모의하고, OFP가 실행되는 SBC의 CPU 및 메모리 사용률을 측정하는 기능을 제공한다. 임무컴퓨터의 HW를 시뮬레이션하기 위해 가상화(Virtualization) 환경을 이용하는 경우[3,4]도 있으나, 가상화 환경을 이용할 경우 임무컴퓨터에 탑재되는 모든 HW를 가상화해야 하는 문제가 있어 본 연구에서는 고려하지 않았다. 인터페이스를 모의하기 위해 Fig. 2의 OFP DKM에서 동작하는 실제 인터페이스 제어 SW를 Fig. 4와 같이 테스트 에이전트로 교체해서 OFP의 스케줄러가 이더넷 제어 모듈을 실행하도록 수정하였다.

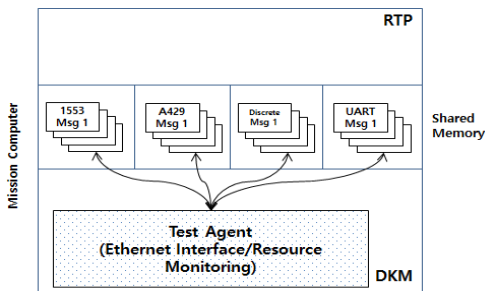


Fig. 4. Implementation of Test Agent

RTP는 공유메모리를 통해 임무컴퓨터 외부와 연동하므로 DKM에서 어떤 HW 인터페이스를 통해 연동하는지 보다 어떤 데이터(구조체)가 송수신되는지가 더 중요하며, 임무컴퓨터 외부와 연동 시 공유메모리에 쓰여지는 구조체 정보만 필요하다. 이더넷을 통해 테스트 에이전트와 PC STE SW간 연동되는 메시지는 공유메모리 이름과 구조체 정보로 이루어진 바이트 스트림이다. 테스트 에이전트가 임무컴퓨터 외부로 메시지를 송신할 경우 공유메모리 이름이 저장된 공유메모리 맵을 순회하여 공유메모리 이름을 획득하고 해당 공유메모리에 쓰여진 데이터를 조합하여 PC STE SW로 송신한다. 반대로 PC STE SW로부터 메시지를 수신한 경우 메시지에 포함된 공유메모리 이름과 구조체 정보를 이용하여 RTP로 전달한다.

2.2.2 PC STE SW 설계

PC STE SW는 Java를 이용하여 개발하였으며 Eclipse 플러그인으로 구현하여 Eclipse를 기반으로 개발된 WindRiver Workbench 등의 통합 개발환경에 설치할 수 있도록 하였다. PC STE SW는 임무컴퓨터와 연동하는 장비를 모의하여 임무컴퓨터로 데이터를 송신하거나 임무컴퓨터로부터 수신한 제어명령을 처리하는 기능을 수행한다. 그러기 위해서는 임무컴퓨터가 실제 HW 인터페이스를 통해 연동장비와 주고 받는 메시지에 대한 정보를 PC STE SW가 가지고 있어야 한다.

임무컴퓨터 연동메시지를 PC STE SW가 처리하기 위해 OFP 소스코드를 파싱하여 메시지 정보를 XML 형태로 추출하는 파서를 JavaCC[5]를 이용하여 구현하였다. PC STE SW는 추출된 XML 정보를 이용하여 메시지 구조체 정보를 Fig. 5와 같이 개발자에게 제공한다.

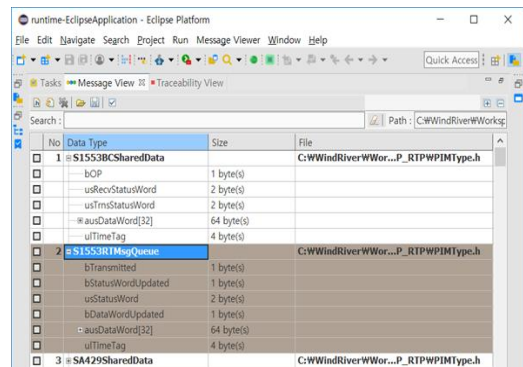


Fig. 5. Message Structure UI

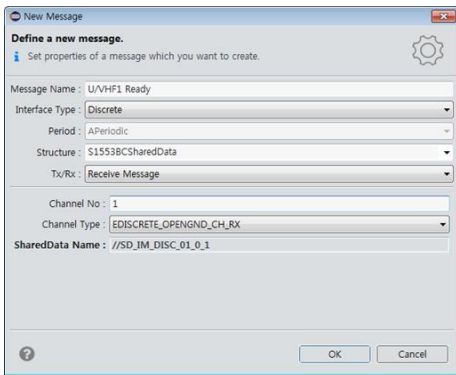


Fig. 6. Defining a new message



Fig. 9. CDU Simulator

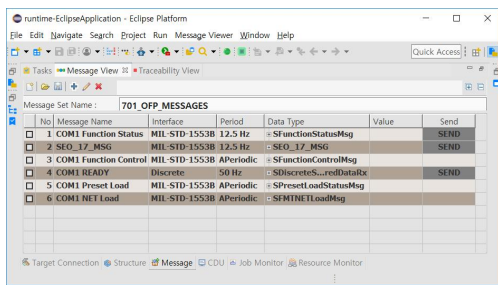


Fig. 7. Message UI

추출된 메시지 구조체 정보를 이용하여 Fig. 6의 UI를 통해 실제로 연동할 메시지 인스턴스를 실제 인터페이스 메시지와 동일하게 정의할 수 있다.

임무컴퓨터 연동메시지 인스턴스를 정의한 후 Fig. 7의 Message UI에서 STE가 임무컴퓨터로 송신하는 메시지의 경우 SEND 버튼을 누르면 메시지 주기마다 해당 메시지가 이더넷을 통해 임무컴퓨터로 전송된다. 이더넷을 통해 임무컴퓨터가 STE로 송신하는 메시지는 Message UI를 통해서 확인할 수 있다.

또한 OFP의 비기능적 요구도 검증을 위해 Fig. 8과 같이 임무컴퓨터에서 동작하는 테스트 에이전트가 측정된 CPU 사용률과 메모리 사용률을 수신하여 실시간으로 모니터링 할 수 있는 기능을 구현하였다.

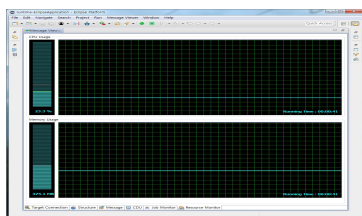


Fig. 8. Resource monitor

2.3 실험

본 논문에서 제안한 STE SW를 유인항공기 OFP 개발에 실제로 적용하여 OFP 로직을 검증하였다. 임무컴퓨터 OFP 요구도 총 529개 중 검증을 위해서 실제 HW 인터페이스가 필요한 요구도 71개를 제외한 458개의 요구도를 본 연구에서 제안한 시험환경 SW를 통해서 검증할 수 있는 것으로 분석되어 SIL을 이용한 요구도 검증 대비 86%의 테스트 커버리지를 확보할 수 있었다. 본 시험환경 SW를 이용해 검증 가능한 458개의 요구도 중 임무컴퓨터와 ARINC 429 인터페이스를 통해 연동하는 CDU(Control and Display Unit) 관련 요구도 검증을 위해 175개의 요구도를 본 시험환경 SW를 이용하여 검증하였다. 이를 위해 PC STE SW에 CDU 모의기를 구현하여 통합하였고, 앞서 기술한 메시지 구조체 파싱, 메시지 인스턴스 정의를 통해 임무컴퓨터와 CDU 모의기간의 메시지 연동을 모의하여 Fig. 9와 같이 실제 CDU에서 시현되는 화면을 CDU 모의기에서 재현하여 CDU 화면이 설계대로 시현되는지, CDU 버튼을 누를 경우 정상동작 여부 등의 기능을 검증할 수 있었다.

III. 결 론

본 논문에서는 단일 PC만을 이용하여 항공전자 OFP를 검증할 수 있는 소프트웨어 시험환경 SW 설계에 대해 기술하였다. 본 논문에서 제안한 소프트웨어 시험환경 SW는 SIL과 달리 OFP 검증을 위해 별도의 HW 없이 상용 PC만을 이용하여 OFP를 검증할 수 있기 때문에 개발자에게 SW 통합 이전에 OFP를 검증할 도구를 제공하고, SIL이 가용하지 않은 경우에도 OFP 검증을 할 수 있어 품질향상 및 개발일정 준수에 효

과적이다.

본 연구에서는 실제 HW 인터페이스를 이더넷 인터페이스로 교체하는 과정에서 불가피하게 OFP(DKM)를 수정하였으나, 추후 연구에서는 시험 대상 OFP를 수정하지 않고 OFP 기능 검증을 할 수 있도록 테스트 에이전트를 OFP와 완전히 분리할 예정이다.

PC STE SW의 실행환경으로 윈도우를 선택하였으며 그로 인해 소프트웨어 시험환경 SW의 실시간성 확보가 어려웠으나, PC STE SW를 Linux로 포팅하여 최소한 Soft Realtime 환경을 확보하고 SIL과 최대한 유사한 환경을 제공할 것이다. 마지막으로 테스트 시나리오 기능을 추가해 SW 통합시험 자동화 및 2016년부터 개발되는 무기체계 SW에 적용되는 요구도 기반 신뢰성시험 자동화 방안에 대해 연구할 계획이다.

References

- 1) Jo, Y. W., Kim, B. G., Park, J. S., and Lee, J. U., "Development of System Integration Laboratory for the Verification of UAV Avionics System Requirements," *Journal of The Korean Society for Aeronautical and Space Sciences*, Vol 40, No. 3, 2012, pp. 446-453.
- 2) Seo, M. G., Jang, W. H., Park, J. S., Seung, D. B., and Kim, S. W., "Development and Verification of Models in System Integration Laboratory for the Avionics System," *Proceeding of the Korean Society for Aeronautical and Space Sciences Fall Conference*, 2015, pp. 2074-2077.
- 3) Engblom, J., Girad, G., and Werner, B., "Testing Embedded Software using Simulated Hardware," *Proceeding of Embedded Real-Time Software and Systems*, 2006.
- 4) Engblom, J., and Mattias Holm, C. W., "A Fully Virtual Multi-Node 1553 BUS Computer System," *Proceeding of Data Systems in Aerospace*, Berlin, May 2006.
- 5) JavaCC, <https://javacc.org>