

技術論文

항공전자 시스템 개발에 관한 연구

양성욱*, 이상철**

A Study on the Development of an Avionics System

Sungwook Yang*, Sangchul Lee**

ABSTRACT

The importance and cost of avionics system in the integration of an aircraft is continuously increasing. And we can expect enlarged software portion in the system integration for the more intelligent, reliable, and automated avionics system. Both military and commercial avionics community have moved toward commercial-off-the-shelf(COTS) equipment and open systems architecture not only to increase affordability but also to reduce acquisition cost, shorten development time and risk. The same concept is applied in developing avionics test system used for the avionics system integration test. In this paper, we present important topics in the development of avionics system including real-time operating system, interconnect data bus, software development methodology, software development process, and system integration test.

key Words : Avionics System(항공전자 시스템), Open System Architecture(개방형 시스템 구조), Real-Time Operating System(실시간 운영 체제), Software Development Process(소프트웨어 개발 절차)

1. 서 론

현대 항공기에 있어 항공전자 시스템은 중요도와 비용측면에 있어 그 비중이 계속 증가되고 있다. 이는 전기전자기술의 급속한 발달에 따른 항공전자기술의 빠른 발전과 나날이 증가되는 임무 요구사항에 그 원인이 있다 하겠다[1-3]. KT-1 항공기 항공전자 시스템, T-50 항공기 항공전자 시스템, 해상초계기(P-3CK) 임무장비 현대화(Modernization), 한국형헬기(KHP) 임무탑재장비의 개발 등 개발이 이미 완료되었거나 현재 개발이 진행 중인 대규모 사업뿐 아니라 무장관리컴

퓨터 시스템 개발, 항공기 비행제어 시스템 개발 등 다수의 개발 프로그램이 있어 왔으며 향후에도 다양한 항공전자 시스템 개발 및 성능개량 프로그램이 예상된다 하겠다. 현재 항공전자 시스템 개발에 있어 민간항공기와 군용항공기 모두 개발 리스크, 기간, 비용의 절감을 위하여 상용화된 기성개발품(COTS)을 가능한 활용하며 시스템의 확장성, 유연성, 정비성과 유지보수비 절감 등을 위하여 개방형 시스템 구조(Open System Architecture)를 적용하는 추세이다[2-6]. 그러나 아직은 하드웨어와 소프트웨어 공히 항공전자 시스템 개발 시 개발자가 손쉽게 사용할 수 있도록 엄밀한 규격화와 모듈화가 이루어졌다고 보기 어렵다[4]. 군용항공기의 경우 Joint Strike Fighter(JSF) 프로그램에서 상용화된 기성개발품 하드웨어와 소프트웨어를 사용한 항공전자 시스템 개발이 진행 중이다[5]. 더 지능적(Intelligent)이며 자동화된 항공전자 시스템을 개발하기 위해서 소프트웨어의 비중은 계속 늘어날 것이 예상

2007년 3월 5일 접수 ~ 2007년 3월 9일 심사완료

* 한국항공대학교 대학원 항공우주 및 기계공학과

** 한국항공대학교 항공우주 및 기계공학부

연락처, E-mail : slee@hau.ac.kr

경기도 고양시 덕양구 화전동 200-1

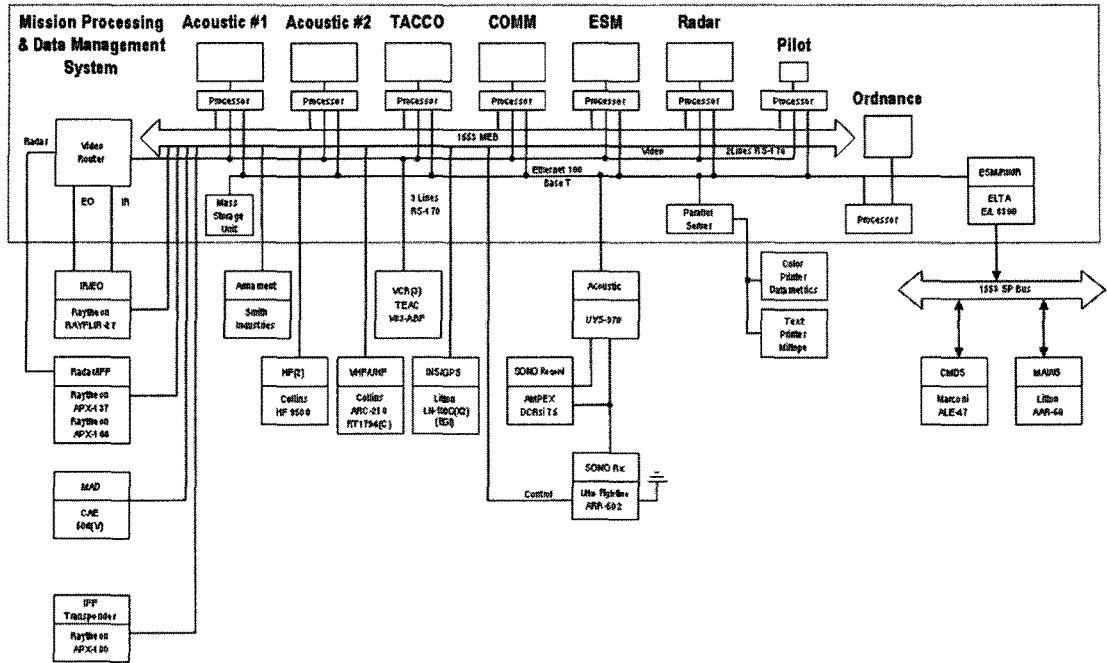


Fig. 1 항공전자 시스템 예

된다. 국내의 많은 산업체들이 시스템과 소프트웨어 영역을 통합시킨 산업체의 프로세스 개선 활동에 대한 평가 모형인 CMMI(Capability Maturity Model Integration) 인증을 획득하고 있는 것도 시스템 통합차원에서 소프트웨어의 중요성에 기인한다고 볼 수 있다. 본 논문에서는 항공전자 시스템 개발 시 고려하여야 할 개발요구도 관련 사항과 실시간 운영체제, 내부통신 데이터 버스, 소프트웨어 개발 및 통합시험에 대하여 살펴보기로 한다.

II. 항공전자 시스템 개발

항공전자 시스템 개발은 개발 요구도의 작성으로부터 시작된다고 볼 수 있다. 주어진 요구도를 분석하여 각각의 장비를 선정할 때 시스템의 임무와 각종 데이터를 처리하는 임무컴퓨터는 우리 인체의 두뇌에 해당된다. Fig. 1은 비교적 복잡한 항공전자 시스템의 한 예이다. 일반적으로 임무컴퓨터를 설계할 경우 최우선 고려 사항은 컴퓨터의 처리속도(throughput)와 메모리 크기이다. 현재의 빠른 처리속도와 대용량의 메모리로 인해 항공전자 시스템은 계속 진화하고 있으며 프로세서와 더불어 실시간 운영체제, 내부통신 데이터 버스의 선정은 항공전자 시스템 개발의 중요한

사항이다. Table 1에서 항공우주 임무시스템을 비교하였다. 상용화된 기성개발품을 활용한 개방형 시스템 구조를 채택하는 개발에 있어 시스템 통합의 측면에서 소프트웨어 개발의 중요성은 점점 커지고 있으며 HMI(Human Machine Interface)를 포함한 항공전자 시스템 요구사항의

Table. 1 항공우주 임무시스템 비교

	Processor	RAM	ROM	통신 프로토콜	RTOS	개발환경
초음속 군용기 IMDC	PPC750 266MHz	32MB (8MB 사용)	EEPROM 16MB (8MB 사용)	1553B	VxWorks	Tornado
한국형 헬기 (예상)	PPC 3, 4세대	32MB	Flash 32MB	1553B, ARINC-429	VxWorks	Tornado
SMS	PPC603e 200/300MHz	64MB	Flash 8MB	1553B	VxWorks	Tornado
AN/ALR 93	PPC7400 133MHz	32MB	Flash 32MB	1553B, Discretes RS-232C, RS-422	VRTX	
KOMPSAT I	80C186 12MHz (16MHz 지원)	512KB	EEPROM 384KB	1553B	VRTX	ICE
KOMPSAT II	80C386 12MHz (33MHz 지원)	512KB	EEPROM 384KB	1553B	VRTX	ICE

많은 부분은 소프트웨어로 구현된다. 통합시험의 효율적 수행을 위한 소프트웨어 엔지니어의 역할은 늘어가는 추세이다.

2.1 개발 요구도

항공전자 시스템의 개발 요구도 작성 시 일반적으로 사용자의 요구사항을 바탕으로 작성된 항공기 요구명세서로부터 항공전자 시스템의 체계 요구사항을 정의하고 이에 따라 각 구성품들의 요구명세서를 정의한다. 개발 초기에 임무요구도 검토회의 등을 통하여 사용자와 개발자간 합의된 명확하고 구체적인 요구사항을 문서로 작성하는 것이 중요하며 이때 양측 모두 현재의 기술수준을 충분히 고려하여야 한다[7]. 항공전자 시스템의 기술 분류는 개발 프로그램에 따라 약간의 차이는 있다. 기본적으로 항공기에 포함되는 비행계기, 통신 및 식별장비, 항법장비, 임무 및 자료처리 시스템이 있으며 항공기의 임무에 따른 비행관리 시스템(Flight Management System), 무장관리 시스템(Weapon Management System), 생존 장비, 전자전 장비, 레이더, 관측 장비(FLIR, EO/IR, MAD, etc.), 음향장비 등 다양한 항공전자 장비가 있다. 이렇게 다양한 항공전자 시스템 장비의 요구도 결정을 위해서는 분야별 전문 인력과 관련 자료의 구축이 필요하며 사용자(소요군)와 개발자간의 긴밀한 협의가 필요하다 하겠

다. 대규모 사업을 수행하는 경우 DOORS[8], RaQuest 등과 같은 요구도 관리 프로그램의 효율적 활용과 연동요구사항(Interface Requirement)의 정확한 작성은 매우 중요하다[7].

2.2 실시간 운영체제

운영 체제는 interrupt 처리, 메모리 관리, 하드웨어의 I/O drivers와 error 관리 등을 제공한다. 특히 항공전자 시스템의 운영 체제는 신속한 처리를 위한 짧은 run-time kernel, 최소화된 latency 등을 가져야 하는 hard real-time 운영 체제를 요구한다[9]. 실시간 운영 체제는 지속적인 처리 속도와 메모리 용량의 증가로 인해 Multi Thread Model에서 Multi Process Model로 변화하고 있다. Multi Thread Model은 운영 체제 kernel과 application이 합쳐져서 서로의 구분이 없는 하나의 큰 프로그램으로 작동하는 구조이며 공통의 작업영역(Memory)을 자유롭게 접근(access)하고, 운영 체제의 크기가 작고 비교적 작은 크기의 시스템에서 구현이 쉽고 빠르다. 하지만 kernel과 application이 하나의 프로그램으로 동작하기 때문에 사소한 버그(bug)가 시스템 전체를 파괴할 수 있다. 대표적인 운영 체제로 VxWorks, OSE, VRTX, pSOS, Nucleus Plus 등이 있다. Multi Process Model은 운영 체제 kernel이나 각 application들이 모두 독립적인 프

Table. 2 Real-Time Operating System

RTOS	Vendor	Processor Supported	Min Size	Development Hosts	Network Support	Specialty Features
pSOS	Integrated Systems >WindRiver	68k	6KB	PC, Sun, HP, native	TCP/IP, Socket	Multiprocessor Support
pSOSelect	Integrated Systems	68k	2.5KB	PC, Sun		
pSOSystem	Integrated Systems	68k, x86, PowerPC, i960	15KB	PC, Sun, HP	TCP/IP, SLIP, FTP, PPP, OSI, Streams, NFS	Multiprocessor Support, SNMP
C Executive	JMI Software Systems	68k, x86, 29k, i960, MIPS, PA-RISC, SPRAC, PowerPC	6KB	PC, Sun	TCP/IP	Multiprocessor Support, SNMP
VRTX	Microtek Research > Mentor Graphic	68k, x86, SPARC	16KB	PC, Sun	TCP/IP, Streams, NFS, RFCs, SLIP	SNMP
OS-9	Microwave Systems	68k, x86, PowerPC	22KB	PC, Sun	TCP/IP, ISDN, SLIP	MS-DOS, Unix
VxWorks	WindRiver Systems	68k, x86, 29k, i960, MIPS, SPRAC, PowerPC	5KB	PC, Sun, HP, SGI, IBM	TCP/IP, Socket, NFS, FTP	Multiprocessor Support, VxMP, SNMP

로그로 동작하도록 설계되었고, 각 application은 서로의 메모리가 보호되어 있기 때문에 모듈 단위의 application 개발이나 모듈(기능)의 추가와 변경이 쉽고 안정된 시스템 개발이 가능하다. 대규모의 시스템 개발이 용이한 장점이 있으나 실시간 운영 체제의 크기가 Multi Thread Model에 비해 크기 때문에 작은 시스템 개발에는 오히려 부담이 된다. 대표적인 운용체제로는 QNX, OS-9, LynxOS, RT-Linux, Windows CE 등이 있다. 실시간 운영 체제의 전략적 비교는 Table 2와 같다[9-11].

2.3 내부통신 데이터 버스

항공전자 시스템의 데이터 버스로서 내부통신 데이터 버스의 선정은 매우 중요하다. 신뢰도와 속도가 내부통신 데이터 버스 선정의 중요한 요인이라 할 수 있는데 현재는 MIL-STD-1553B와 ARINC 429가 많이 사용되고 있다. 빠른 전송속도로 인한 Ethernet의 도입과 무장제어 분야에서 MIL-STD-1760의 적용이 주목된다. 최근 ARINC 429는 ARINC 629로 대체되고 있는 추세이나 ARINC 629의 coupling device는 Ethernet에 비해 크기가 크고 MIL-STD-1553B에 비해 무게가 무겁다[12]. Table 3에서 대표적인 내부통신 데이터 버스를 비교하였다[1,12,13].

2.4 소프트웨어 개발

연산속도(throughput)와 메모리 크기의 비약적인 발전 및 사용자의 복잡하고 다양한 요구사항으로 인하여 소프트웨어의 중요성은 점차 증가하고 있다. 소프트웨어 개발 방법은 구조적 방법론(Structured Methodology)에서 재사용성과 확장성이 뛰어난 객체지향 방법론(Object Oriented Methodology)을 사용하는 추세이다. 구조적 방법론은 1970~1980년대 각광을 받은 기법으로 계층적 분할, 논리적 시스템 모델 구축, 하향식(Top-Down) 접근방법이다. 이 방법은 데이터의 흐름과 구조를 중심으로 하며 사용하는 대표적인 프로그램 언어로는 FORTRAN77, C, Ada83 등이 있다. 1990년대 이후, 객체지향 방법론이 각광을 받기 시작했는데 객체지향 개념을 적용한 소프트웨어 설계는 데이터와 데이터를 다루는 Function 및 Procedure를 하나로 묶어 객체(Object)라는 개념을 사용하여 실세계를 표현하는 방식이다. 객체지향 방법에서는 객체의 속성과 연산의 식별이 가능하고, 객체의 관계성을 정의할 수 있으며, 상향식(Bottom-Up) 개발이 주로 사용된다. 캡슐화(Encapsulation), 다형성(Polymorphism), 상속성(Inheritance) 등을 그 특징으로 한다[14-16]. 이 방법을 사용하는 대표적인 프로그램 언어는

Table. 3 Interconnect Data Bus

	MIL-STD-1553B	RS-232C	ARINC-429	ARINC-629	Ethernet
속도	1 Mbps	최대 19.2 Kbps	12.5 Kbps 또는 100Kbps	2 Mbps	10 Mbps, 100 Mbps, 1 Gbps
적용 분야	항공, 우주, 군사용	일반용	항공, 우주, 군사용	항공용	인터넷
통신 방식	Half-Duplex (반-이중) (1:n 통신)	Half-Duplex (Point-to-Point) (1:1 통신)	Simplex(단방향) with Multiple Receivers (1:n 통신)	Full-Duplex (m:n 통신)	Half-Duplex (반-이중) (m:n 통신)
전송 매체	Coaxial Cable	Twisted Pair	Twisted Pair	Twisted Pair	Coaxial Cable 또는 Optical Fiber
유효전송거리	6 m	20 m	53 m (175ft)	100 m	200 ~ 500 m
특징	Bus Controller가 Remote Terminal 들을 제어한다.	Transmit/Receive를 위한 동일한 Channel이 필요함	Transmit/Receive를 위한 독립된 Channel이 필요함	각각의 terminal이 data bus상의 다른 모든 terminal에 대해 Transmit/Receive이 가능함	Pear-To-Pear 통신을 수행함
항공 시스템 적용 분야	항공/우주분야의 Avionics System Bus에 적용됨	항공/우주 분야의 LRU와 LRU 사이의 저속 통신에 사용됨	항공/우주 분야의 LRU와 LRU 사이의 저속 통신에 사용됨	B777에 최초로 적용된 후 항공/우주 분야에 적용되는 단계임	인터넷 분야에 주로 이용되나, 빠른 통신속도로 인하여 항공/우주 분야에 적용되는 단계임

C++, Ada95, JAVA 등이 있다. 1998년 IEEE, Electronic Industries Association에서는 소프트웨어의 개발에 관한 이해와 적용을 위해 명확한 개념정의와 가이드라인을 제공하는 Software Life Cycle Processes인 IEEE/EIA 12207를 제정하였다. IEEE/EIA 12207는 ISO/IEC 12207의 내용을 전부 포함하고 있으며 같은 해 DoD(미국방성)에 의하여 채용되었다. Fig. 2는 IEEE/EIA의 발전과정이고, Fig. 3은 ISO/IEC 12207 프로세스 구성 체계이다.

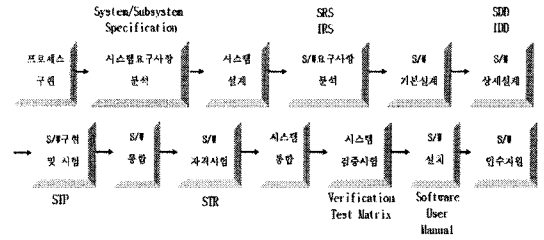


Fig. 4 Software Development Process

평가 모형은 소프트웨어 능력 평가(Software Capability Evaluation)와 프로세스 개선을 목적으로 한다. 소프트웨어 프로세스의 진단 (Software Process Assessments)을 통하여 조직의 현행 소프트웨어 프로세스의 상태를 진단하고 평가하여 소프트웨어 개발 중의 문제점을 개선할 수 있으며 결과적으로 품질 및 개발생산성 향상을 가져온다. 소프트웨어 능력 평가를 통해 발주기관은 소프트웨어 사업자 평가에 프로세스 평가 모형의 결과를 활용하고, 시스템 통합 사업자 평가에 대한 전문성 및 공정성을 확보할 수 있으며, 소프트웨어 사업자는 국제적으로 인정된 기준에 의하여 객관적 능력을 인정받아 대외 신인도를 제고할 수 있다. 국제적으로 인증된 평가 모형으로는 미국 CMU SEI의 CMMI, 유럽의 SPICE(ISO/IEC 15504)가 대표적이다. Fig. 5는 CMMI의 발전과정을 보여주고 있다[18]. CMMI는 완성도(maturity levels)를 통해 분류된 22가지의 process area를 평가함으로써 각각의 process 완성도를 판단할 수 있으며, 또한 개선이 필요한 부분을 찾아낼 수 있다. CMMI에서 제시하는 완성도는 5단계(1~5 Levels)로 구성되어 있다. Table 4는 완성도를 구성하는 process area를 보여준다[18,19].

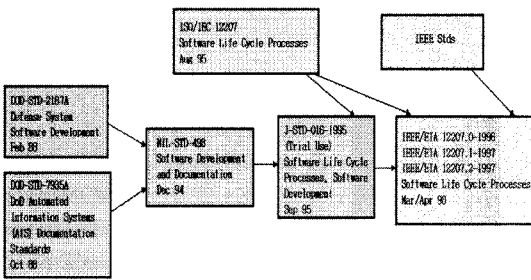


Fig. 2 IEEE/EIA 12207 History

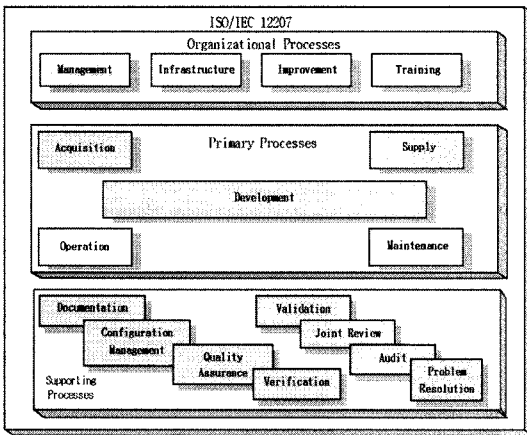


Fig. 3 ISO/IEC 12207 Process

소프트웨어를 개발함에 있어서 고려되어야 사항들은 요구사항 분석, 설계, 구현, 통합, 시험, 설치 및 인수 등 소프트웨어 제품 개발과 관련된 활동이 있으며, 이 사항들의 순차적인 일련의 과정이 소프트웨어 개발의 프로세스이다. Fig. 4는 일반적인 소프트웨어 개발의 프로세스를 나타낸다[17]. 각각의 과정에서는 개발규모 및 성격에 따른 각 개발단계별 Documentation Tailoring 및 작성이 중요하다.

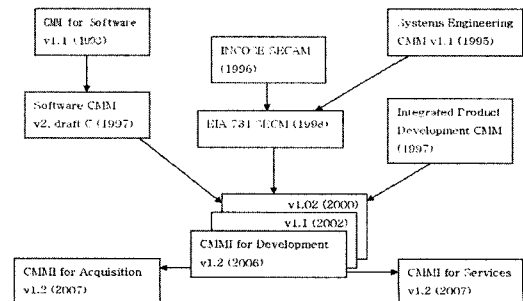


Fig. 5 History of CMMs

Table. 4 Maturity Level and Process Area

Maturity Level	Characteristics	Process Area
5	continually improves its processes based on a quantitative understanding of the common causes of variation inherent in processes	- Causal Analysis and Resolution - Organizational Innovation and Deployment
4	the organization and projects establish quantitative objectives for quality and process performance and use them as criteria in managing processes	- Organizational Process Performance - Quantitative Project Management
3	processes are well characterized and understood, and are described in standards, procedures, tools, and methods.	- Decision Analysis and Resolution - Integrated Project Management +IPPD - Organizational Process Definition +IPPD - Organizational Process Focus - Organizational Training - Product Integration - Requirements Development - Risk Management - Technical Solution - Validation - Verification
2	the projects of the organization have ensured that processes are planned and executed in accordance with policy	- Configuration Management - Measurement and Analysis - Project Monitoring and Control - Project Planning - Process and Product Quality Assurance - Requirements Management - Supplier Agreement Management

성하거나 혹은 달성 가능한 능력 수준을 판단하는 것으로 6개의 Capability Level(0~5 Levels)로 구성되어 있다. Fig. 6는 SPICE의 평가 체계를 보여준다.

2.5 통합시험

하드웨어 제작업체로부터 납품된 장비(LRU)에 대한 기능 및 환경시험 결과가 요구 조건을 만족시키는지 확인하고 소프트웨어가 유닛(unit) 테스트, 스레드(thread) 테스트 등을 만족시키면 통합시험 단계로 들어간다. 항공전자 시스템 통합시험은 일반적으로 Avionics Hot Bench(AHB), Avionics System Development Rig(ASDR), Avionics Test System(ATS) 등으로 불리는 통합시험 장비를 통하여 이루어진다. 통합시험 장비는 통합시험을 위한 항공전자 장비 모의, 비행 환경 시뮬레이션, 데이터 감시, 데이터 저장 및 재생, 실 장비와의 연동 등의 기능을 제공한다. 상용화된 기성 개발품을 활용한 개방형 시스템 구조를 채택하는 개발에 있어 시스템 통합의 측면에서 소프트웨어 개발의 중요성은 점점 커지고 있으며 항공전자 시스템 요구사항의 많은 부분은 소프트웨어로 구현되는 추세이다. 따라서 효율적인 통합시험을 위한 가장 중요한 요인 중 하나는 시스템 엔지니어 그룹과 소프트웨어 엔지니어 그룹간의 밀접한 조율이라 하겠다[20]. 통합시험 장비의 개발에 있어서도 항공전자 시스템 개발과 마찬가지로의 이유로 상용화된 기성 개발품을 활용한 개방형 시스템 구조가 채택되고 있다[21].

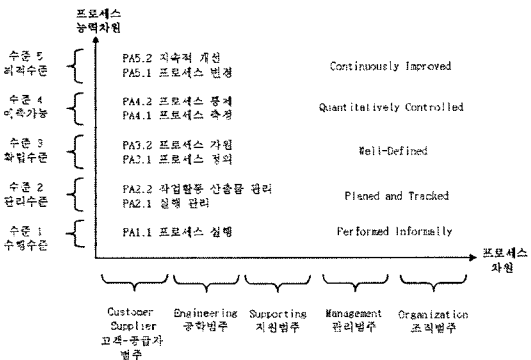


Fig. 6 Two Dimensional Model of SPICE

SPICE (Software Process Improvement and Capability dEtermination : ISO/IEC TR3 15504) 는 S/W 프로세스에 대한 개선 및 능력 측정 기준으로 ISO/IEC 에 의하여 제정되었다. SPICE의 특징은 2가지 차원의 평가 모델을 기준으로 측정한다는 것이다. 2가지 차원은 프로세스 차원 (Process Dimension)과 프로세스 능력 차원 (Process Capability Dimension)으로 구분된다. 우선, 프로세스 차원은 5개의 프로세스 카테고리가 40개의 세부 프로세스로 구성되어 있고, 각 프로세스 별로 목적을 달성하기 위한 기준을 제시하고 있다. 다음으로 프로세스 능력 차원은 조직단위(Organization Unit)가 특정 프로세스를 달

III. 결론

항공전자 시스템의 중요성은 계속 증가하고 있으며 개발에 사용되는 기법도 발전을 거듭하고 있다. 미국과 유럽 공히 개발 리스크, 기간, 비용의 절감을 위하여 상용화된 기성개발품을 활용하고 있으며 시스템의 확장성, 유연성, 정비성과 유지보수비 절감 등을 위하여 공개되고 관리되는 하드웨어 및 소프트웨어 인터페이스를 사용하는 개방형 시스템 구조를 따른다. 하드웨어와 소프트웨어의 모듈화(Modularity) 및 기능의 독자성 (Functional Independence)을 추구하며 더 좋은 기술(technology)의 적용보다는 좋은 기능(function)이 우선시 된다. 이 모든 경향은 궁극적으로 개발비의 감소와 빠른 전자장비의 발전으로 인한 도태장비 관리를 포함하는 군수지원의

용이성을 통하여 순기비용(Life Cycle Cost)을 절감하려는 목적으로 사료된다. 항공기의 수명이 30~40년인데 비해 항공전자장비의 수명은 전기전자기술의 급속한 발달로 10년 이내에서 계속 줄어들고 있다. 낮은 전력 소모, 경량, 고성능의 새로운 장비의 출현은 항공전자 시스템 개발의 방향을 바꾸었으며 항공전자 시스템 성능개량 프로그램은 향후 늘어날 것이 예상된다. 지속적인 프로세서 처리속도와 메모리 용량의 증가로 인해 여타 분야에서는 실시간 운영체제로 RT-Linux나 Windows CE 같은 Multi Process Model이 도입되고 있으나 현재 항공전자 시스템의 실시간 운영체제는 Multi Thread Model이 사용되고 있으며 이 중 VxWorks가 가장 많이 사용되고 있다 [9]. 내부통신 데이터 버스는 빠른 전송속도를 가진 Ethernet과 ARINC 629의 적용이 주목된다. 소프트웨어 개발기법은 재사용성과 확장성이 뛰어난 객체지향 방법론 사용이 늘어나고 있다. 평가모형의 인증 획득을 통한 지속적인 개발생산성 및 소프트웨어 품질 향상이 필요하며 항공전자 시스템뿐 아니라 통합시험 장비 개발에 있어서도 상용화된 기성개발품을 활용한 개방형 시스템 구조의 사용이 계속 증가되는 추세이다.

참고문헌

- [1] Moir, I., Seabridges, A., AIAA Education Series Aircraft systems, AIAA, 2001, pp. 315-337.
- [2] 이상철, 김인규, 김영일, "무장관리 컴퓨터 탑재 소프트웨어 개발", 한국항공우주학회지, 제31권, 제5호, 2003, pp. 124-133.
- [3] 오우섭, 김성우, 최이주, 장진석, "한국형헬기(KHP) 임무탑재장비 개발방안 연구", 한국항공우주학회 추계학술발표회, 2005. 11.
- [4] Wilcock, G., Totten, T., Gleave, A., Wilson, R., "The Application of COTS Technology in Future Modular Avionics Systems", Electronics & Communication Engineering Journal, Vol. 13, No. 4, 2001, pp. 183-192.
- [5] Filmer, B., "Open Systems Avionics Architectures Consideration", IEEE Aerospace and Electronic Systems Magazine, Vol. 18, No. 9, 2003, pp. 3-10.
- [6] Roark, C., Kiczuk, B., "Open System Avionics Architectures", IEEE Aerospace and Electronic Systems Magazine, Vol. 10, No. 9, 1995, pp. 18-22.
- [7] Hooks, I. F., Farry, K. A., Customer-Centered Products, AMACOM, 2001.
- [8] Telelogic AB, Using DOORS for Requirements Management, 2003.
- [9] Cedeno, W., Laplante, P. A., "An Overview of Real-Time Operating Systems", Journal of the Association for Laboratory Automation, Vol. 12, No. 1, 2007, pp. 40-45.
- [10] Cooling, J., "Real-Time Operating Systems for the Embedded World", IEE Open Control Systems - The Importance of Industrial Standards, 2004, pp. 4/0-4/10.
- [11] Wind River Systems, Inc., VxWorks Programmer's Guide, March 1997
- [12] Rieckmann, N., ARINC 629 data bus physical layer technology, Microprocessors and Microsystems, Volume 21, Issue 1, 1997, pp. 13-20.
- [13] Audsley, N. C., Grigg, A., "Timing analysis of the ARINC 629 databus for real-time application", Microprocessors and Microsystems, Vol. 21, Issue 1, 1997, pp. 55-61.
- [14] Jacobson, I., Booch, G., Rumbaugh, J., The Unified Software Development Process, Addison-Wesley, 1999
- [15] Douglass, B. P., Real-Time UML - Developing Efficient Objects for Embedded System, Addison-Wesley, 1998
- [16] Smith, M. A., Object-Oriented Software in Ada95, International Thomson Computer Press, 1996
- [17] MIL-STD-498 Software Development and Documentation, 1994
- [18] Carnegie Mellon Software Engineering Institute, CMMI@ for Development, Version 1.2, 2006
- [19] Kossiakoff, A., Sweet, W. N., System Engineering Principles and Practice, John Wiley & Sons, Inc. 2003.
- [20] Braid, D. M., Johnson, C. W., Schillinger, G. D., "An Integrated Test Approach for Avionics System Development", IEEE 20th DASC, Vol. 2, 2001.
- [21] Pizzica, S., "Open Systems Architecture Solutions for Military Avionics Testing", IEEE Aerospace and Electronic Systems Magazine, Vol. 16, No. 8, 2001, pp. 4-9.