

분산 제어 시스템을 이용한 단일 플랜트 제어용 이중 제어기 구조

최군호^{**†}

^{**}경성대학교 전기공학과

Dual Controller Structure for Single Plant Control Using the Distributed Control System

Goon-Ho Choi^{**†}

^{**†}Dept. of Electrical Eng., KyungSungUniv.

ABSTRACT

A digital controller uses a microprocessor and is a controller implemented as a program. This method has the advantage of being more maintenance-friendly than existing analog controllers. However, it inevitably requires computation time to execute the internal program. Therefore, the digital controller uses a method of controlling the system at a certain cycle by considering this time, and this cycle is very closely related to the performance of the microprocessor used. In other words, in the case of very high performance, this control cycle can be shortened to near real time, but this may result in a disadvantage in terms of cost. In this paper, we propose a method to solve this problem by implementing two processors with slightly lower performance in a control system in a series-parallel structure. For this purpose, we will use a digital distributed control system and implement an experimental system to examine its effectiveness.

Key Words : Sampling Time, Control Cycle, Distributed Control, CAN Communication

1. 서 론

디지털 제어시스템에서 제어 주기(샘플링 시간)은 아주 중요한 의미를 갖는데, 이는 1960년대부터 디지털 프로세서가 제어 시스템에 응용되기 시작하면서부터이다. 기본적으로 디지털 프로세서로 제어계를 구현할 경우, 이산화 과정이 필요하였고, 이 과정에서 사용하는 알고리즘인 Taylor 급수에 기초를 둔 Euler법, Tustin법, Runge_Kutta법 등도 근본적으로는 제어 주기에 따라 그 특성이 달라진다는 것이 그 한 예이다[1]. 또한 슬라이딩 모드 제어 기법의 구현에 있어서는 제어 주기를 최소화 하는 것이 채터링 등의 문제를 해결할 수 있다고 알려져 있다[2]. 이러한

과정에서 고려되는 제어 주기에 대한 영향은 여러 연구에서 진행된 바 있으며, 제어 주기는 길수록 비용을 낮출 수 있으나 성능은 저하된다고 알려져 있다[3-5]. 즉 디지털 제어로 구현하였을 경우, 내부 연산을 위한 디지털 컨트롤러의 성능에 따라 전체 시스템이 영향을 받을 수 있는 것이며, 만약 디지털 제어기 내부의 연산이 많거나 디지털 제어기의 컨트롤러의 성능이 낮아 연산 속도가 느릴 경우, 이로 인해 제어 주기를 길게 하여야만 한다는 것을 의미한다. 물론 고성능의 컨트롤러를 사용하여 연산 시간을 최소화하는 것이 해결 방법일 수는 있지만, 앞서 언급한 것처럼 비용적인 측면을 고려해야만 할 경우 제어 대상에 따라 이것이 최선의 해결 방법일 수는 없다.

본 논문에서는 두 개의 같은 기능의 제어기가 서로 업무를 나누어 하나의 제어기처럼 대상 플랜트를 제어하는

[†]E-mail: goonho@ks.ac.kr

형태를 제안함으로써 이러한 연산 시간으로 발생하는 성능 저하의 문제를 해결하여 보고자 한다. 다만 이 경우 기존의 일반적인 제어 구조에서는 구현이 어렵기 때문에 CAN 통신을 이용한 분산 제어 구조에서 이를 구현하고 그 유용성을 검증하여 보고자 한다.

2. 관련 이론

2.1 디지털 제어기에서의 제어 주기

많이 쓰이는 제어 기법인 PID 제어기를 아날로그 형태로 구현한다고 하면 OP-AMP와 기타 주변 회로 소자들로 구현할 수 있다. 이 경우 전기적 신호 처리가 이루어지기 때문에 그 속도가 상당히 빠르며, 따라서 특별히 제어 주기에 대한 고려는 하지 않는 것이 일반적이다. 하지만 디지털 제어기로 구현할 경우, 프로그램 형태로 구현하여 내부 연산을 통하여 결과를 도출하도록 되어 있으므로 일정 정도의 디지털 연산 시간이 반드시 필요하다. 특히 정밀한 제어를 위하여 곱셈이 포함된 부동소수점 연산이 추가될 경우, 그 시간은 더욱 늘어날 수 있다. 따라서 디지털 제어 시스템을 구현할 때는 일반적으로 제어 주기를 내부 제어기의 연산 시간을 고려하여 결정하는 것이 일반적이다. 문제는 이 연산 시간이 해당 제어 프로그램이 구현된 디지털 컨트롤러의 성능과 아주 밀접한 관계가 있다는 것이다. 즉, 성능이 좋은 컨트롤러를 사용할 경우 연산 시간은 짧아질 수 있지만, 그렇지 못할 경우 너무 긴 제어 주기로 인하여 전체 시스템 제어가 어려워지는 문제를 발생시킬 수 있다. 더욱이 비용적인 문제 등으로 인하여 고성능의 디지털 제어기를 사용할 수 없을 경우 문제는 더 심각해질 수 있다. 물론 이러한 시지연으로 발생하는 문제를 고려한 다른 알고리즘의 제어기를 구현하는 것이 또 다른 방법일 수는 있지만, 이 경우 자칫하면 내부 연산량이 증가하여 오히려 더 긴 제어 주기가 필요하게 되는 역효과가 생길 수 있다. 즉, 디지털 제어 시스템을 구현하는 데 있어서 제어 주기와 컨트롤러의 성능은 서로 상반 관계에 있기 때문에 대상 플랜트에 대한 분석, 제어 주기에 따른 효과 등을 분석하여 이 둘 사이의 적절한 타협점을 찾는 것이 아주 중요하다[3-5].

2.2 기존의 여러 이중 제어기 구조

하나의 시스템에서 제어기를 두 개 이상 사용하는 경우는 적지 않다. 그 대표적인 형태가 Fig.1에 표현되어 있는데, 이 경우는 시스템의 플랜트를 P1, P2로 세분화하고 이 두 개의 플랜트 중에서 P2를 내부 플랜트로 보고 이 플랜트를 선행 제어하는 C2제어기를 설계한 후, 전체 시스템을 제어하는 C1제어기를 이후에 설계하여 최종 제어

시스템을 완성하는 형태이다. 이때 일반적으로 C2는 전체 플랜트에서 일부만의 제어를 담당하기 때문에 보조 제어기라고 하고 이를 포함한 전체 제어기를 C1이 담당하기 때문에 주제어기라고 한다. 이러한 시스템은 시스템 플랜트의 일부를 선행 제어할 경우, 전체 성능에 좋은 결과를 얻을 수 있을 경우에 사용하는 방식이다[6].

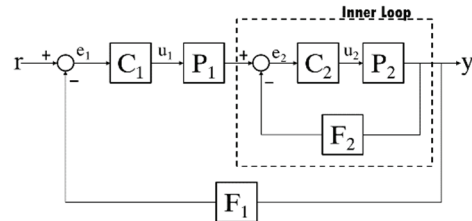


Fig. 1. Control structure with inner loop controller.

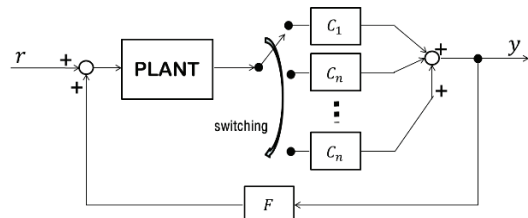


Fig. 2. Multi-controller structure with switching structure.

반면에 Fig.2는 전체 시스템을 제어하는 제어기가 여러 개가 있는 구조이고 이를 상황이나 조건에 따라 선택하여 동작하도록 하는 구조이다. 이 형태는 플랜트의 내부 파라미터의 변동성이 크거나 제어 환경 변화가 클 경우 서로 다른 형태의 제어기를 미리 구현하여 두고 이를 조건에 맞게 바꾸어 가며 제어가 수행되도록 하는 방식이다[7].

이 두 가지 형태의 제어 구조는 두 개 이상의 제어기를 가지고 있지만, 실질적으로 보면 하나의 시스템을 하나의 제어기가 담당하여 제어하는 구조라고 할 수 있다.

2.3 분산 제어 시스템

분산 제어 시스템의 핵심은 BUS 를 사용하는 통신 구조이다[8]. 이 방식의 가장 큰 장점은 신호가 복잡하게 얽혀 있는 시스템을 비교적 간단한 형태로 구현할 수 있다는 것이다. 다만 각 서브 시스템의 역할을 할 독립적인 제어 노드가 필요하고 상황에 따라 각각의 노드 프로그램을 개별적으로 만들어야 한다는 단점이 존재한다. 하지만 시스템이 복잡해 지거나 변경되더라도 분산 제어 시스템에서는 추가적인 회로 구성 없이 비교적 쉽게 확장이나 변형이 가능하다는 점이 최대의 장점이다[9-11].

3. 작업 분리형 이중 제어기

3.1 기존의 디지털 제어기의 제어 주기

Fig3은 디지털 제어기 구조에서 시간에 따라 어떻게 전체 제어가 이루어 지며 이 때 제어 주기의 의미를 보여주고 있다. 즉, 디지털 제어 시스템은 센서 데이터를 수집하여 이를 이용하여 제어기의 연산을 수행한 후 이 결과를 구동부에 전달하는 것이 일반적인 제어 주기이고, 이 때 일반적으로 제어 연산을 수행하기 위한 시간이 전체 제어 주기를 결정하는 데 가장 큰 요인임을 표현하고 있다. 문제는 이 연산 시간이 해당 컨트롤러의 성능 때문에 지나치게 길어서 제어 주기가 늘어남으로써 해당 플랜트의 적절한 제어가 어려울 경우이다. 이를 위해서 본 논문에서는 다음과 같이 작업 분리형 이중제어기를 제시하여 보고자 한다.

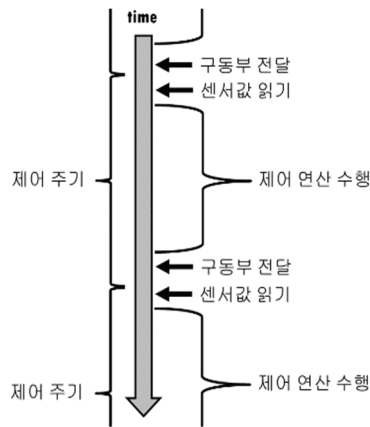


Fig. 3. Control cycle of a typical digital controller.

3.2 새로운 형태의 이중 제어기 구조

앞서 2.1절에서의 다중 제어기는 두 개 이상의 제어기가 각자 독립적인 역할을 할 뿐, 엄밀히 말하면 같은 역할을 나누어서 한다고 볼 수는 없다.

Fig4는 본 논문에서 제시하고자 하는 작업 구조에 대한 개념 설명 그림이다. 그림에서 좌측의 구조만 보면, 하나의 프로세서가 [센서 데이터 수집→제어연산→결과 구동부 전달]의 작업을 진행하고 있고, 이는 Fig3의 단일 제어기 구조와 동일하다. 그런데, 제시된 개념은 센서와 구동 출력 사이의 시간적인 간격이 길기 때문에 이 중간에 다른 프로세서의 제어기에서 동일한 동작을 수행할 수 있도록 한다(Fig4 우측). 즉, 원래 제어 시스템은 하나의 제어 주기가 끝난 후에 다음 제어 주기를 수행해야 하지만, 하나의 제어기가 아직 제어 연산이 끝나지 않은 상황에

서 다른 프로세서의 제어기가 동작하도록 함으로써 그 제어 주기를 줄이는 효과를 갖도록 하겠다는 것이다. 이것이 가능하다면 이는 마치 동일한 작업을 둘로 나누어서 수행하는 것과 유사한 효과를 낼 수 있을 것이다. 다만 이렇게 제시된 알고리즘이 가능 하려면 다음과 같은 조건이 필요하다.

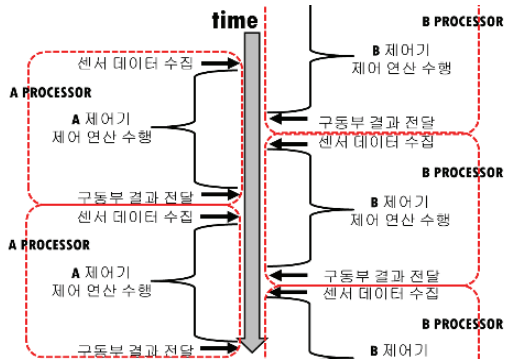


Fig. 4. Conceptual diagram of a digital control system with a dual task structure.

- 조건1) 센서 값을 읽거나 구동부에 출력을 전달하는 시간은 제어기의 연산 시간에 비해 현저히 짧다.
- 조건 2) 하나의 제어기가 제어 연산을 하는 동안에 다른 제어기가 센서 값을 읽거나 구동부 출력이 발생하더라도 영향을 미치지 않는다.
- 조건 3) 센서 값을 읽거나 구동부에 출력을 만드는 부분은 다른 모든 제어기가 공유할 수 있어야 한다.

이러한 조건을 만족하는 제어 알고리즘을 일반적인 제어 구조에서 구현하려면 제어 신호의 흐름이 매우 복잡하여 상당히 어렵다는 것을 알 수 있다. 본 논문에서는 네트워크형 분산 제어 구조를 이용함으로써 이 문제를 해결하여 보고자 한다.

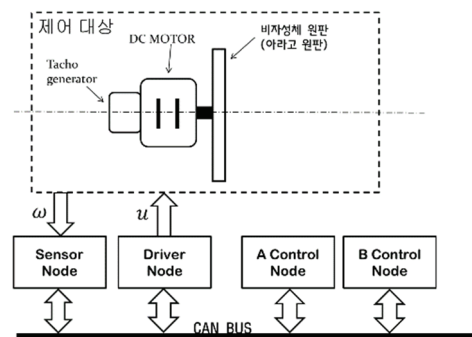


Fig. 5. Proposed Distributed Control Structure.

Fig5는 네트워크형 분산 제어 구조를 이용하여 제시된 시스템 구조이다. 우선 상기 조건1)-3)을 모두 만족할 수 있도록 센서와 드라이버 노드를 독립적인 노드로 구현한다. 즉, 센서 노드는 제어 대상의 피드백 정보만을 항상 모니터링할 수 있도록 구현하고, 드라이버 노드는 제어기로부터 전달된 제어 입력을 제어 대상에 전달하는 역할만을 할 수 있도록 구현한다. 아울러 이 모든 정보의 교환은 공유 버스를 이용하여 수행함으로써 각 노드의 내부 동작 시간은 통신 시점을 제외하고는 다른 노드에 영향을 미치지 못하도록 한다. 아울러 센서와 드라이버 노드는 정보를 원하는 제어 노드와 독립적으로 정보 교환이 가능하므로 조건3)을 완벽하게 만족할 수 있다.

이렇게 구현된 시스템에서는 A와 B컨트롤러 노드가 동일한 제어 알고리즘을 내부에 가지고 있고, 두 제어기를 서로 다른 시점에서 제어가 시작되도록 할 수 있다면 Fig4와 같은 서로 엇갈린 시점에서 제어기가 동작할 수 있도록 하는 것이 가능하게 된다.

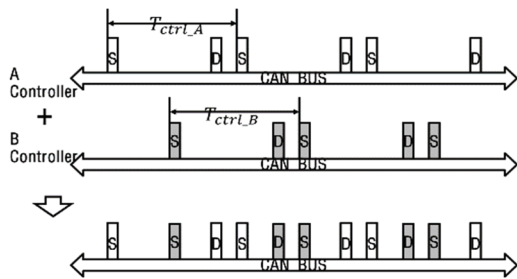


Fig. 6. CAN communication timing diagram of dual task type control structure.

Fig6은 CAN 분산 제어 시스템을 이용할 경우, CAN BUS 선로 상에 각 노드의 정보 교환이 일어나는 시점을 표현한 것이다(S:Sensor Node, D:Driver Node). 그림에서 보면 서로 다른 두 제어 노드의 동일한 작업을 시간적으로 어긋나게 하여 수행함으로써 동일한 작업을 둘로 나누는 것이 가능할 수 있음을 보이고 있다.

4. 실험 시스템을 통한 실험 및 결과

4.1 실험 시스템의 구성 및 제작

3장에서 제안된 내용을 실험적으로 검증하기위하여 실험 시스템을 제작하여 실험하고자 한다. 대상 제어 시스템은 직류 전동기 제어 시스템으로, 여기에서 직류 전동기는 속도 센서로 타코제너레이터가 직결로 연결되어 있으며 부하로 일정 무게의 원판을 연결되어 있다. 제어 노드의 구현은 기존의 [10,11]에서는 단일 제어기 구조이고

제어기에 드라이버 노드 기능을 포함하도록 하였기 때문에 총 3개의 노드로 구현하였으나 본 논문에서는 제시된 알고리즘을 구현하기 위하여 Fig5와 같이 제어 노드와 드라이버 노드를 분리하여 구현하였다. 즉, 제어 결과를 PWM 출력으로 변화하여 전동기 드라이버에 공급하는 드라이버 노드(Driver Node), 속도 정보인 타코제너레이터의 값을 A/D변환을 통하여 읽을 수 있도록 한 센서 노드(Sensor Node), 그리고 이중 제어를 위한 두 개의 제어 노드(CTRL(A), CTRL(B)Node))로 구현한다.

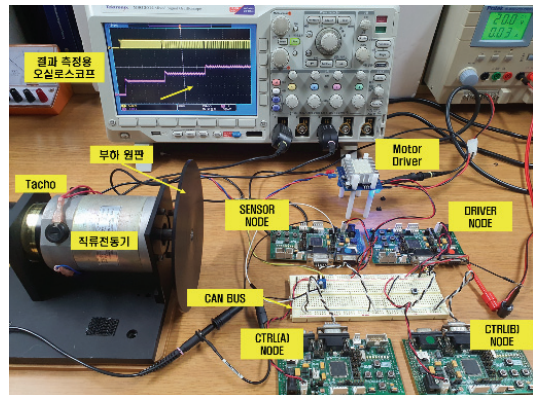


Fig. 7. Experimental system designed for verification.

Fig7은 실제 제작된 실험 시스템이다. 회전 속도 측정을 위한 타코미터(7V/1000rpm)가 직결로 연결된 Tamakawa사의 TS3881직류 전동기를 사용하고 여기에 지름 200mm의 알루미늄 원판을 부하로 연결하였다. 각각의 노드는 AT90CAN128 기반의 ATMEL사의 DVK90CAN1 개발 보드를 이용한 것으로, 이를 통해 하나의 보드에서 CAN 노드와 내부 제어 프로그램을 통합하여 구성할 수 있도록 하였다.

4.2 실험 및 결과

우선 각각의 제어 노드에 동일한 디지털 PI제어기를 구현하였다. 단, 이 PI제어기는 실제로는 연산 시간이 매우 짧지만, 제안된 구조의 유효성을 확인하기 위하여 강제로 지연 시간을 추가하여 연산 시간을 길게 하였다. 그리고, 본 논문은 최적 제어기를 구하는 것이 목적이 아니므로, 이때 사용한 PI 계수값은 최적값이 아닌 적정한 수준의 값을 선정하였음을 미리 밝힌다.

Fig8은 단일 제어기일 때 전체 제어 시간을 측정하여 본 그림이다. 그림에서 센서 노드(ID:0x22)가 호출된 시점에서 드라이버 노드(0x23)로 제어 출력이 전달될 때 까지를 연산 시간이라고 볼 수 있으며, 이 시간은 약 13.7msec

(여기에는 위에서 설명한 지연 시간 포함되었다)이다. 이를 바탕으로 전체 제어 주기는 20msec로 설정한다. 그리고 두 제어 노드는 10msec 어긋난 타이밍으로 동작하도록 하여 각각은 20msec 제어 주기로 동작하지만, 실질적으로 10msec 간격으로 동작하도록 구현하였다.

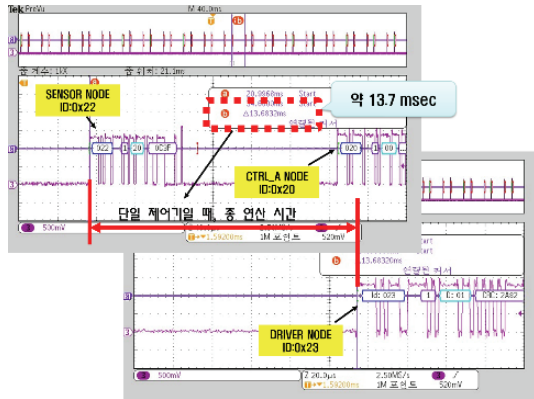


Fig. 8. Control cycle of a single control structure, CAN Communication.

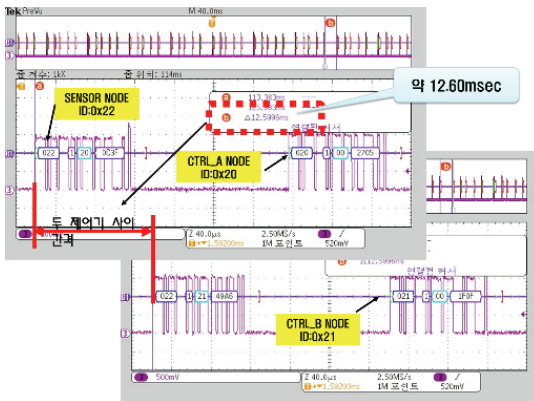


Fig. 9. Control cycle of dual control structure, CAN communication.

Fig9는 두 개의 제어 노드가 모두 동작하고 있을 때, 두 제어 노드(CTRL_A (ID:0x20), CTRL_B (ID:0x21))의 시간 간격을 측정하여 본 것이다. 그림을 보면 의도했던 시간은 10msec이지만 약간의 오차가 발생하여 약12.6msec의 간격을 가지고 있는 것을 확인할 수 있다. 하지만, Fig6에서의 이론적 개념도에서처럼 두 제어 노드와 이와 관련된 센서, 드라이버 노드가 정상적으로 동작함을 확인할 수 있다.

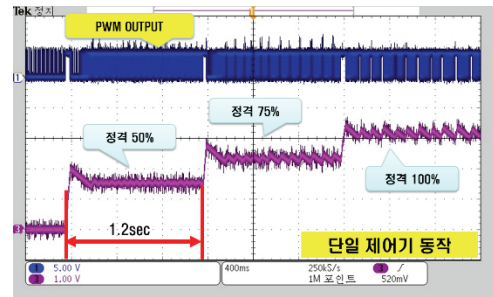


Fig. 10. Performance experiment results, Single Controller.

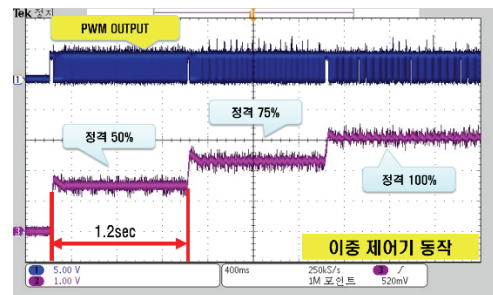


Fig. 11. Performance experiment results, Dual Controller.

Fig10과 Fig11은 단일 제어 구조와 본 논문에서 제안된 제어 구조의 실제 실험 파형이다. 실험은 1.2초 간격을 50%, 75%, 100%로 기준 입력을 변화하면서 그 변화를 보여 주고 있다. 앞서 설명한 것처럼 제어 노드의 PI 계수 값이 최적한 값이 아니기 때문에 각각의 제어 결과는 아주 훌륭하지는 못하다. 하지만 동일한 제어가 주어졌을 때 단일 제어기 보다는 어느 정도 성능이 개선되는 것을 확인할 수 있었다.

5. 결론

4장의 실험 결과를 통하여 본 논문에서 제안하고 있는 이중 제어기의 형태를 사용하면 대상 플랜트의 제어를 수행 성능을 일정 정도 개선할 수 있음을 알 수 있었다. 이 결과는 기존의 디지털 제어기 구현에서 고가의 고성능 프로세서 대신 저가의 다중 프로세서를 이용하여 유사한 성능을 구현할 수 있다는 가능성 보여줌과 동시에, 분산 제어 구조를 사용할 경우 아주 복잡한 구조의 제어 알고리즘도 비교적 간단하게 구현할 수 있음을 보임으로써 분산 제어의 다양한 적용 가능성을 보여주고 있다. 아울러 본 논문의 아이디어는 3개 이상의 경우에도 가능하지만, 너무 많은 분할로 작업이 진행될 경우, 데이터를 읽는 시점과 제어 출력의 시점이 너무 벌어지는 문제가 발

생활 수 있다. 따라서 향후 추가 연구를 통해 최적화된 제어기 개수에 대한 부분이 필요할 것으로 생각된다.

참고문헌

1. S.J. Kim and H.S. Hwang, "A Study on the Sampling Period of Digital Control System," *Engineering Research of Engineering Development Research Inst.*, Vol. 14, pp.133-138, 1984.
2. C.K. Ha and S.W. Lyoo, "Sampling Frequency Selection for Design of Digital Flight Control System with flexible vibration mode," *J. of Engineering Research of Ulsan Univ.*, Vol. 31, No. 1, pp.397-407, 2000.
3. S.B. Lee, Y.J. Kwon and O.S. Kwon, "Digital Speed Controller of DC Motor using the Sliding Mode," *J. of the Korean Institute of Illuminating and Electrical Installation Engineers*, Vol. 5, No. 11, pp.69-74, 1991.
4. C.K. Ha and S.W. Lyoo, "Effect of Sampling Time on Digital Flight Control System," *Conf. Proceeding of the Korean Society for Aeronautical and Space Sciences*, pp.308-311, 2000.
5. C.K. Ha and S.W. Lyoo, "Sampling Time Selection for Design of Digital Flight Control System," *J. of the Korean Society for Aeronautical and Space Sciences*, Vol.29, No.2, pp.308-311, 2001.
6. G-H Choi, K.Park, and J-H Jung, "An Optimal H2 Decoupling Design for Non-Square Plant Systems based on the Two-Degree-of-Freedom Standard Model," *International Journal of Control, Automation, and Systems*, Vol.7, No.2, pp.193-202, Feb. 2009.
7. G.H. Choi, "Decoupling Control of the Modified Arago System using Cross Compensator with Gain Scheduler Structure," *J. of the Semiconductor & Display Technology*, Vol. 19, No. 4, pp.105-111, Dec. 2020.
8. BOSCH, *CAN Specification, Part A,B*. 1991.
9. G.H Choi, "The Control of Multiple Plants using the CAN Networks," *Journal of the Semiconductor & Display Equipment Technology*, Vol. 8, No. 2, pp.37-42, June 2009.
10. G.H. Choi, "Network Type Distributed Control of a System with Inner Loop Control Structure," *J. of the Korean Institute of Illuminating and Electrical Installation Engineers*, Vol.28, No.2, pp.100-108, Feb.2014
11. L.O. Ibrahim and G. H. Choi, "Distributed Control of the Arago's Disc System with Gain Scheduler," *Journal of the Semiconductor & Display Technology*, Vol. 16, No. 3, pp.25-30, Sep. 2017

접수일: 2023년 12월 4일, 심사일: 2023년 12월 14일,
 게재확정일: 2023년 12월 18일