

분산제어방식을 적용한 콤바인 농기계의 제어기 개발

양성수*, 박경석#, 신동원**, 이재식*

A Development of Control System for Agricultural Machinery by using Distributed Control

Sung-Soo Yang*, Kyoung-Seok Park#, Dong-Won Shin**, Jae-Shik Yi*

(Received 17 February 2011; received in revised form 9 September 2011; accepted 6 December 2011)

ABSTRACT

Due to the introduction of the electronic control systems, agricultural machine is getting complicated with its special function. The complex electronics system increases difficulty of maintenances in the wiring harness system. The heavy wiring harness has some poor reliability and workability in the production line of manufacture also. So, in this study, some small local control modules are developed to complement the difficulties with the benefits of distributed control system. All of the local control modules are connected by using CAN communication. This system developed this study has the advantages of modification or alteration of wiring and local modules. All the sensors and actuators can be easily monitored and controlled by the main controller with the appropriate communication protocol made by in this study.

Key Words : Agricultural Machine(농기계), Wiring Harness System(와이어링 하네스), Distributed Control System (분산제어 시스템), CAN Communication(CAN 통신)

1. 서 론

최근 세계적으로 반도체 및 FPD 제조장비 업체를 중심으로 로봇 및 자동화 장비의 제어방식이 중앙집중식에서 Ethernet-IP, EtherCAT, Mechtrolink, RTEX 등과 같은 고속의 네트워크(분산)방식으로 급속하게 전환되고 있는 실정이다.

한편, 농기계분야에서도 전자제어시스템화가 급속도로 진행되면서 안전성, 편리성, 능률 향상, 생산 시간 단축이라는 장점을 이루어 가고 있으나, 중앙집중식 전자제어 시스템화에 따른 와이어링 하네스의 증가와 배선곤란, 중량증가, 신뢰성 저하, 생산라인의 작업성 저하, 와이어링 하네스의 복잡화 등의 부차적인 문제점이 나타나고 있다.

이러한 문제는 다기능, 고기능화가 진행될수록 더욱 심화될 것이고, 이는 와이어링 하네스의 레이아웃 최적화만으로는 해결이 불가능하다.

본 연구에서는 앞서 언급한 농기계 전자제어시스템 구성에 관련한 문제점을 개선하기 위해 모듈 콘트롤

* 금오공과대학교 대학원 자동차공학과

교신저자 : 금오공과대학교 기계시스템공학과

E-mail : kspark@kumoh.ac.kr

** 금오공과대학교 기계시스템공학과

러와 CAN 통신을 응용한 분산제어 시스템^[1-3]과 이를 적용한 새로운 와이어링 하네스 구조를 고안하였다.

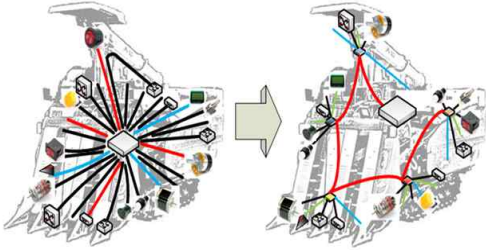


Fig. 1 Concept diagram of the distributed wiring harness structure in this study

일반적으로 컨트롤러와 여러 센서 및 액추에이터들 간의 연결 방법^[4]은 컨트롤러의 I/O라인을 이용하여 직접 연결하는 방식과 로컬 컨트롤러를 두어 통신을 이용하여 연결하는 간접 방식으로 나눌 수 있을 것이다. Fig. 1의 좌측은 기존의 직접연결 방식으로 하나의 메인 컨트롤러에서 모든 신호를 각각의 와이어에 의해서 직접 전달받고 디바이스를 동작시키던 시스템이다. Fig. 1의 우측은 로컬 컨트롤러를 사용하여 구역별로 나누고 로컬컨트롤러와 메인 컨트롤러 간의 직렬 통신을 사용하여 연결하여 동작시키는 네트워크통신 방식을 말한다.

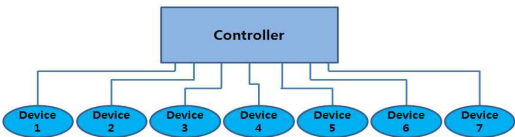


Fig. 2 Direct connected structure between controller and devices

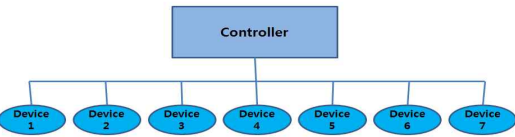


Fig. 3 Wiring structure by using communication between controller and devices

직접연결 방식은 Fig. 2와 같이 신호가 디바이스와 컨트롤러에 직접 연결되므로 신호의 지연이나 손실이

없다. 하지만, 모든 신호선이 컨트롤러로 모이게 되므로 디바이스의 개수가 증가하면서 와이어링 하네스가 두터워지고 실제 와이어링에서 복잡한 구조를 가지게 되는 단점을 가지고 있다.

네트워크 통신 방식은 Fig. 3과 같이 하나의 네트워크 통신선만을 사용하기 때문에 디바이스의 개수가 증가하여도 와이어링 하네스가 두터워지거나 복잡한 구조를 가지지 않아도 된다. 하지만, 하나의 통신선으로 개별 디바이스에 관련된 모든 신호가 동일하게 전달되기 때문에 통신량이 많을 때 신호 전달의 지연이 발생하고, 통신상의 에러에 의한 신호 손실이 발생할 가능성이 있다.

2. 개발한 분산제어기 구조

2.1 시스템 구성

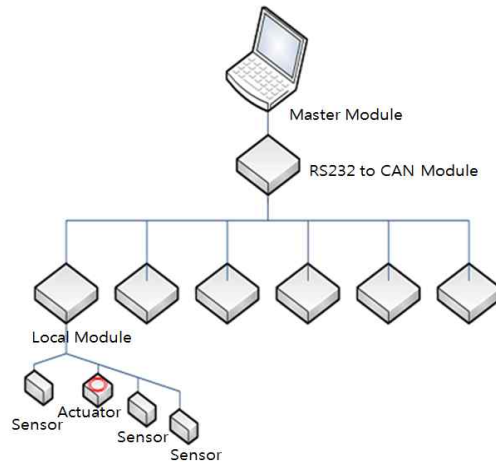


Fig. 4 Distributed control system configuration

본 연구에서는 Fig. 4와 같은 방식으로 분산제어 시스템을 구성하였다. 메인 컨트롤러와 모든 로컬 모듈(로컬 컨트롤러)들은 하나의 CAN 통신으로 연결된다. 메인 컨트롤러는 RS232 신호를 통하여 PC나 모니터링 장치와 연결할 수 있다. 로컬 모듈들은 지역적으로 가까운 위치에 있는 각종 센서와 액추에이터 디바이스들이 연결된다.

로컬 모듈에 속하는 모든 센서와 액추에이터들을 통신모듈들을 별도 장착하여 개별적으로 메인 컨트롤러와 모두 연결하게 되면 개발비용 상승과 데이터 지연 등의 문제점을 야기하므로, 중간에 로컬 모듈을 두어 센서와 액추에이터를 몇 개씩의 작은 단위로 묶어서 연결하였다. 따라서 본 연구에서는 메인 모듈은 통신을 사용하여 로컬 모듈을 제어하고, 로컬 모듈은 센서 및 액추에이터를 직접 제어하는 2중 구조방식을 취하였다.

2.2 데이터 프레임 및 전송방법

본 연구의 분산제어 시스템에서는 CAN 통신을 이용하므로 한 번에 8바이트까지 하나의 전송 패킷단위로 메인과 로컬 모듈 사이에 데이터를 전달할 수가 있다. 또한 한 번에 전달되는 8바이트는 각각의 바이트의 위치에 따라 특정한 의미를 갖도록 할 수 있다.

본 연구에서는 8바이트 중 6바이트를 1개의 프레임으로 구성하여 데이터를 전달하는 방법을 사용하였다. 이는 RS422과 같이 바이트 단위로 전송되는 경우에 데이터의 양이 많아지면 전송속도에 영향을 줄 수 있으므로 최소한의 데이터를 효율적으로 사용할 수 있도록 하기 위함이다.

본 연구에서 사용한 RS232용 데이터 패킷의 구조는 아래와 같다.

Table 1 Packet format for communication

Sync1	Sync2	CMD	ID	DATA 1	DATA 2
-------	-------	-----	----	--------	--------

Sync1, Sync2 : start bytes for a new data frame

CMD : byte for command

ID : byte for data receiving module no.

DATA1, DATA2 : communication data

위 구성에서 Sync1, 2는 RS232 또는 RS422 통신과 같이 전송되는 데이터의 연속적인 흐름 속에서 각 패킷을 구별하는 동기신호가 된다. 이 동기를 나타내는 2바이트를 뒤에 따라오는 네 개의 데이터가 실제 의미 있는 데이터이다.

모니터링모듈에서 메인 모듈로 전송될 때 데이터는 RS232를 이용하므로 동기바이트를 포함하여 전송되나 CAN에서는 나머지 4바이트 데이터가 로컬모듈로 전송되어 처리된다. CAN 통신용 데이터 패킷의 구조는 아래와 같다.

Table 2 Packet format for CAN control modules

SID	LEN	CMD	RID	DATA 1	DATA 2
-----	-----	-----	-----	--------	--------

SID : device no. of sending data

Length : length of sending data

CMD : byte for command

RID : device no. of receiving data

DATA1, DATA2 : communication data

모니터링 모듈에서 메인 컨트롤러를 통해 2번 로컬 모듈에 부속되어 있는 3번 장치에 출력신호를 보내는 경우, RS232통신으로 다음과 같은 데이터를 보내게 된다. 동기신호로서는 FFh(=255)를 2번 사용하게 되는데, 본 연구에서 사용하는 데이터의 최대 수는 99까지로 제한하여 사용되므로 255와 혼동되지 않는다. 83의 8은 출력이며 3은 3번째 I/O 출력을 제어하는 명령을 표시한다. 02는 로컬 모듈 번호이며, 01은 출력을 ON 시킨다는 의미이다.

Table 3 Packet example for RS232 communication

FF	FF	83	02	01	01
----	----	----	----	----	----

이 패킷 데이터는 다시 메인 컨트롤 모듈에서 CAN 변환기를 거쳐 아래와 같은 CAN 데이터로 변환되어 로컬 모듈로 전송된다.

Table 4 Packet example for CAN communication

100	04	83	02	01	01
-----	----	----	----	----	----

여기서 100은 메인 모듈 번호이며, 04는 CAN 통신 데이터 8바이트 중에서 유효한 실제 데이터의 길이를 의미한다.

필요한 데이터도 이와 같은 전송방식을 통하여 메인 모듈에서 순차적으로 로컬 모듈을 호출하여 정보 데이터를 요청 하게 되면 각각의 로컬모듈은 요청 신호에 맞추어 필요 데이터를 전송하게 된다. 정보 데이터를 받은 메인 모듈은 그 정보를 바탕으로 제어로직을 통해 데이터를 화면에 표시하거나, 모터, 램프, 밸브 등의 구동을 위한 구동 명령을 로컬모듈로 전송한다.

마이크로 컴퓨터의 I/O포트에 직접 연결되는 대부분 중앙 집중 제어 시스템의 경우에는 입, 출력 신호가 지연되는 경우가 없다. 그러나 분산제어의 경우 모든

입, 출력 명령이나 데이터가 네트워크통신을 통해 순차적으로 전달되므로 로컬 모듈에서 발생한 긴급한 데이터의 경우에는 다음 폴링 사이클까지 지연이 발생할 수 있다. 특히 스위치의 경우 값이 변하게 되면 메인 모듈이 이를 빨리 인지하고 제어를 수행해야 하는데, 폴링의 경우 늦게 대응할 수 있는 문제를 안고 있다.

이러한 분산제어 방식의 폴링방식에 의한 정보 전달의 지연 현상을 해결하기 위해서는 인터럽트 전송 방식이 필요하게 된다. CAN 통신에서는 데이터 ID에 따른 전달의 우선 순위가 부여 되어 있으므로 이를 활용하면 인터럽트 방식의 사용이 가능하다. 연구에서는 데이터의 우선순위를 사용하여 스위치 등의 긴급 데이터들을 실시간으로 즉시 확인하여 대처할 수 있게 하였다. 변동이 느린 아날로그 데이터들은 그 중요도에 따라 다른 시간 간격으로 데이터를 요청하는 방식으로 데이터 지연문제를 해결하였다.

또한, 여러 종류의 많은 데이터들이 전달되는 과정에서 가끔씩 일부 데이터의 손실이 발생하게 되면 제어가 제대로 이루어지지 않는 치명적인 문제점이 발생할 수도 있다.^[5] 본 연구에서는 이런 문제에 대처하기 위해서 모든 액추에이터 구동 명령은 적어도 매 번 3회 이상을 반복하여 전송하도록 하였다.

3. 네트워크 제어 시스템의 응용

3.1 농기계 콤바인의 주요 기능

본 연구에서 적용대상으로 삼은 농기계 콤바인의 시스템은 원래 2~3개의 중앙 집중 방식의 컨트롤러로 구성되어 있었으나 분산 네트워크 제어 방식으로 개조하여 총 17개의 로컬 모듈에 각종 센서 및 액추에이터가 분산 배치되도록 하였다.

콤바인의 주요 제어 장치 및 기능을 몇 가지 설명하면 다음과 같다. 예취 높이 제어장치는 지면 상태나 작물의 상태에 따라 예취부의 높이를 자동으로 제어하여 균일한 크기의 작물 취득을 돕는 장치인데 Fig. 6에 사진으로 보였다.



Fig. 6 The photograph of the cutter height control equipment

차체 수평 제어 장치는 Fig. 7과 같으며, 지면의 기울기에 관계없이 항상 차체를 수평으로 유지하여 습지 적응성 및 토양 침하로 인한 작업 성능 저하를 해결한다.^[6]



Fig. 7 The photograph of the leveling control equipment

공급 깊이 제어장치는 Fig. 8과 같으며, 예취·반출되는 작물의 공급 깊이를 자동적으로 조절 하여 탈곡 성능을 항상 양호한 상태로 유지시켜 주기 위한 장치이다.



Fig. 8 The photograph of the feeder control equipment

3.2 제어 알고리즘

본 연구에서 사용한 제어알고리즘에 대하여 도시하면 Fig. 9와 같다. 로컬모듈에 전원이 인가되면 우선 업데이트 항목을 조사한다. 업데이트 항목이 있으면 로컬 모듈의 업데이트를 진행하고, 업데이트 신호가 없으면 EEPROM을 통하여 자신의 ID와 입출력 포트를 설정하여 초기화를 수행한다.

초기 세팅이 끝난 모듈은 입력단자를 통하여 받아들인 스위치 등의 입력신호를 저장하고 있다가 메인 모듈의 명령에 따라 CAN을 사용하여 값을 전송하거나, 명령을 받아서 출력 단자를 통해 출력 신호를 내보내는 루프를 반복하게 된다.

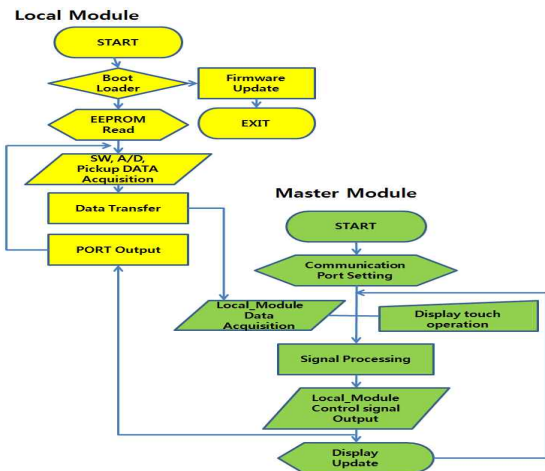


Fig. 9 The flow chart of the Local module cycling functions

메인모듈은 전원이 인가되면 우선 로컬 모듈과 통신을 하기위한 통신포트를 설정한다. 통신 포트의 설정이 끝나면 로컬모듈로부터 데이터 신호를 취득하고 입력된 신호를 처리하여 다시 로컬모듈로 출력 제어 명령을 보낸다. 더불어 디스플레이 화면을 구성하고 화면의 업데이트를 진행한다.

콤바인 운전자가 장치의 작동 필요에 따라 디스플레이에 구성된 화면을 통해 작동 명령을 내리면 처리 과정을 거쳐 로컬 모듈에 출력제어 신호를 보낸다.

3.3 모듈 구성



Fig. 10 The photograph of the embedded controller with touch panel

본 연구에서 사용한 메인 모듈은 Fig. 10과 같이 32Bit의 RISC ARM프로세서를 사용하고 7inch TFT Touch Display를 갖춘 산업용 임베디드 컴퓨터로서 RS232 단자와 입출력 포트를 사용하여 외부 장치를 제어할 수 있다.

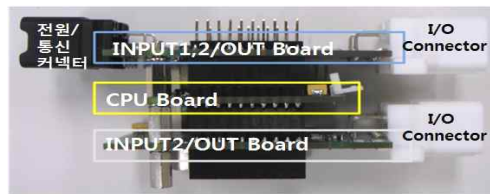


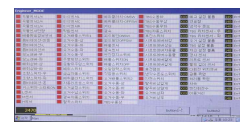
Fig. 11 The photograph of the local control module structure

로컬모듈을 Fig.11과 같이 구성되어 있다. 상·하에 위치한 입출력 보드에서 제어 신호의 출력과 입력이 이루어지고, 중앙에 있는 CPU보드는 제어와 통신을 수행하는 역할을 한다.

입출력 보드는 각각 6개의 입력 또는 8개의 출력단자를 가지고 있으며, 입출력 보드를 구성하는 방법에 따라 최대 12개의 입력 또는 최대 6개의 입력과 최대 8개의 출력이 가능하다.



(a) Input



(b) Output

Fig. 12 Screen configuration for IN/OUTPUT

Table 5 Summary of IN/OUTPUT functions

INPUT			
Grain L/M/H/F SW	곡물센서	Lubrication SW	윤활유 주입
Warning SW	배출완료	Cutter SW	예취
Mono L/R SW	모노레버	Thresh SW	탈곡
Cutter Up/Dw SW	예취 상하	CutterClutch ON/OFF SW	예취 자동
Steer L/R SW	조향	FeedChain ON/OFF SW	피드체인
TailLight SW	미등	EngineStop SW	엔진정지
L/R Blink SW	방향지시등	PumpingClutch ON/OFF SW	배출클러치
Hom SW	혼	OilPressureSW	오일압력
StartOn SW	시동	WorkLightingSW	작업등
L/M/H sensor SW	이삭깊이	TBS Up/Dw SW	TBS 상하
EarSensor L/C/R SW	이삭센서	TBS L/R SW	TBS 좌우
GrainBlocking SW	막힘센서	TBSAuto SW	TBS 자동
AugerPumping SW	배출	ShiftLever Up/Dw SW	탈곡깊이 상하
Auger Up/Dw SW	오거 상하	ShiftLever A/M SW	탈곡깊이 자동
Auger L/R SW	오거 좌우	MovingBack SW	후진스위치
AugerSet SW	오거세팅	Temp SW	냉각수 온도
AugerReturn SW	오거수납	Safety SW	안전스위치
GeupSimAuto SW	급심	ChargerLamp sw	충전
CutterClutch SW	예취클러치	CutterHeightVol	예취높이
AutoLift SW	자동리프트	FuelLevel	연료량
AutoUnion SW	결속자동	WaterTemp	냉각수온도
MovingLockSW	주행제한	TBSPosition_L/R	TBS L/R 높이
AugerSetVol	오거설정값	AugerPositionSensor	오거 위치센서
TBSSetVol	TBS 설정값	FeedSpeed	급동속도
HeightVol	탈곡깊이센서	DispdeSpeed	처리통폭업
TiltingAngle	경사센서	VehSpeed	차속
LiftSensor	리프트센서	EngineRPM	엔진RPM
ThreshDepthVol	탈곡깊이센서		

OUTPUT			
RearBrkLamp	후면	WorkLampLED	작업등 LED
RearTailLamp	후면미등	TBSAutoLED	TBS 자동 LED
RearBlinkLamp L/R	후면 시그널	TBS L Up/Dw Valve	TBS 좌 상하
FrontBlinkLamp L/R	정면 시그널	TBS R Up/Dw Valve	TBS 우 상하
FrontTailLamp	정면미등	SteeringVal L/R	조향 밸브 좌우
WorkLamp	작업등	Auger Up/Dw Valve	오거 상하 밸브
AugerPumpingLED1	배출LED	Cutter UP/Dw Valve	예취 상하 밸브
GeupSimAutoLED	급심자동 LED	OperationVal	작동 밸브
CutterClutchLED	예취클러치 LED	WarningBuzzer	경보 버저
AutoLiftLED	자동상승 LED	Horn	혼
CutterClutch ON/Off	예취클러치	FuelStopRelay	연료차단
AugerSetLED	오거설정 LED	PreheatRelay	예열
AugerReturnLED	오거수납 LED	LubricationMotor	주유펌프
AutoUnionLED	결속자동 LED	StartMotor	시동모터
BackBuzzer	후진경보	Thresh Dp/Si Out	달고길이
FeedChain ON/OFF	피드체인	AugerMove L/R Out	오거 좌우
PumpingClutch ON/OFF	배출클러치	AugerBreakOut	오거 브레이크
ScreenMotor	스크린모터		

Fig. 12는 임베디드 컨트롤러를 사용하여 로컬모듈을 제어하기 위해 화면을 구성한 것이다. (a)는 약 80여 개의 스위치 신호와 12개의 아날로그 입력, 4개의 속도와 함께 우측에는 통신장치가 발생한 모듈을 표시하고 있다. 스위치의 값은 스위치가 닫히게 되면 노란 아이콘으로 표시되고 에러가 발생한 스위치는 빨간 아이콘으로 표시된다.

(b)는 출력신호를 내보내기 위한 화면이다. 화면에 구성되어있는 탭을 체크하면 로컬모듈에 연결되어 해당되는 디바이스들이 동작하게 된다.

Table 5는 입, 출력 되는 데이터들을 정리하여 나타낸 것이다.

4. 장비개발 효과 및 결과

전자 제어 시스템의 다양화, 복잡화, 고기능화 등은 농기계 분야에서도 예외는 아니다. 이러한 변화로 인해 다기능의 농기계에 있어 와이어링 하네스의 복잡성이 문제화 되고 있는 상황이다. 본 연구에서는 이를 해결하기 위해 와이어링 구조 변경과 더불어 효율적인 로컬 제어 방식을 고안하였다. 본 연구를 통하여 도출한 결과를 정리하면 다음과 같다.

- 기존 와이어링 하네스는 컨트롤러와 연결되는 전선의 가닥이 수십 가닥의 뭉치로 묶여 있어 큰 부피와 무게를 차지하고 있으나 본 연구에서는 전체 와이어링의 구조를 백본 와이어링 4가닥(전원2, 통신2)으로 간소화하였다.



(a) Original wiring (b) Developed wiring
Fig. 13 The photographs of the original and newly developed wiring harnesses for comparison

Fig. 13(a)는 기존의 복잡한 덩어리 형태의 와이어링 하네스 구조를 나타낸 것이다. 사진에서 노란박스로 표시한 부분이 메인 컨트롤러이고 여기에 다량의 센서와 액추에이터들이 복잡하게 얽혀 있음을 볼 수 있다. 한편, Fig. 13(b)는 본 연구에서 새로 고안한 와이어링 구조를 나타낸 것이다. 가로방향의 검은색 백본 와이어링에 대해 소형 로컬 컨트롤러들이 세로로 분기되어 있는 배치되어 있다.

- 기존의 많은 입, 출력 신호들은 메인 컨트롤러에 직접 연결되지 않고 통신으로만 연결되어야 하므로 효율적인 통신 방법이 필요하다. 따라서 본 연구에서는 6바이트를 1개의 패킷단위로 구성하여 모든 입, 출력을 표현할 수 있는 프로토콜을 고안하였으며 시간 지연을 최소화할 수 있는 데이터 스케줄링 방안을 고안하였다.

5. 결론

위와 같이 본 논문에서 제시하는 하드웨어 및 소프트웨어 방식을 실제로 적용하여 본 결과 다음과 같은 결론을 얻을 수 있었다.

기존 차량용 와이어링과 제어 시스템의 경우에는 장치의 확장 또는 변경을 위해서 새로운 와이어링 하네스의 제작이 필요하지만, 본 연구에서는 4가닥의 백본 와이어링의 분기 또는 경로 변경만으로 가능하였다. 따라서 이러한 방식이 시스템 확장 또는 개조 측면에서 매우 편리함을 확인하였다.

또한, 로컬 제어 모듈들의 표준화 및 규격화를 통해

새로운 장치의 첨가 또는 변경과 같은 소량 다품종 형태의 장비 개발에 대하여 유연성이 매우 뛰어나도록 확인할 수 있었다.

참고문헌

- 1) Kimita Ishida, kazuya Akashi, and takao Kato, "Multiplex Application on the wiring system", SAE P-241, 910407.
- 2) Katumi Miyake, "A Multiplexed Body Wiring System for Practical Use", SAE SP-806, 900704.
- 3) Seunghoon Kim, Wootaik Lee, Myoungcho Sunwoo, "A Co-simulation Toolbox for Distributed Real-Time Control System", KSAE Vol. 11, No. 1, 2003.
- 4) 이석, 김만호, 이경창 "차량용 네트워크 기술 연구 동향", 한국정밀공학회지 Vol. 23, No. 9, 2006.
- 5) B. wittenmark, J. Nilsson, M. Tornngren, "Timing Problems in Real-time Control System", In Proceedings of the American Control Conference, 1995.
- 6) 이상식, 문정환, 박원엽, 이충호, 이규승, 황 현, "콤바인 차체의 유압수평제어 장치에 관한 연구", 바이오시스템공학 Vol. 29, No 5, pp. 425-432(2004).