논문 2011-2-28

CAN기반 로봇 내부 통신 시스템에 대한 성능평가시스템(RoNSpy) 개발

Development of the RoNSpy : Performance Evaluation System for In-Robot Network System based on CAN

이경중*, 김재오*, 박재한**, 백문홍**, 안현식***

Kyung-Jung Lee, Jae-Oh Kim, Jae-Han Park, Moon-Hong Baeg, Hyun-Sik Ahn

요 약 로봇의 고성능화, 지능화에 따라 다수의 센서, 액츄에이터 및 제어기 적용이 확대되고 있다. 기존의 로봇 내부의 와이어링 기술은 점대점으로 직접 배선을 연결하는 방식이었으나 최근에는 버스 구조의 네트워크를 채택하고 있다. 다양한 버스 구조의 네트워크 중 차량에 이미 적용 중이고 신뢰성이 입증된 CAN(Controller Area Network)의 적용이 보편적이다. 이에 따라 로봇 내부 통신시스템의 성능평가를 통해 네트워크의 실시간성 및 신뢰성에 대한 분석이 요구된다. 본 논문에서는 CAN기반의 로봇 내부 네트워크에 대한 성능평가시스템을 구성하고 네트워크 관리도구 및 응용 모듈 등을 개발하였으며 이를 이용하여 실제 CAN 통신 시스템에 대한 전송시간 및 표준편차를 분석함으로써 네트워크 시스템의 실시간성을 판단할 수 있음을 보인다.

Abstract The high performance intelligent robots have become to use a lot of sensors, actuators and controllers which are connected via bus-type networks while the point to point connection was widely used in-robot networks of existing robots. The bus topology makes the robot system share data internally and it can also reduce the weight of the robot because of smaller wiring harness. The CAN is one of serial bus communication protocols typically used for in-vehicle networks and now it can also be used for in-robot networks because of its low cost and reliability. A performance evaluation system for this type of networked system is developed to analyze the real-time operation and the reliability of a serial-bus network, and the usefulness of the developed evaluation system is shown by some experiments for a network of many CAN nodes.

Key Words : Robots, CAN(Controller Area Network), Performance Evaluation System.

Ⅰ. 서 론

사람과 함께 공존하는 공간에서 동작해야 하는 휴머노이드 로봇은 우선적으로 사람의 안전이 보장되어야 하며, 환경에 잘 어울려야 하고, 오작동 시 위협이 되지 않아야 한다, 그러므로 휴머노이드 로봇은 우선 크기에 제한을 받고, 주변환경의 상황이나 로봇의 상태를 항상 다수의 센서를 통해 감지하며, 감지한 데이터를 실시간으로 처리해 즉시 동작해야 한다. 이를 위해 휴머노이드 로봇에서 사용되는 센서와 액츄에이터들은 용도에 따라 로봇의 구조에서 특정 위치에 설치되어야 하므로 한 곳에 집중될 수 없다는 특징이 있다. 로봇의 이동이나 특정작업을 위한 액츄에이터의 경우도 각각의 기능에 적합한 위치에 분산되어 장착된다. 따라서 제한된 크기의 로봇 내부는 센서, 액츄에이터, 그리고 중앙 제어기 사이에 다양한 종류의 전기케이블의 연결로 인해 매우 복잡하게

*준회원, 국민대학교 전자공학과
**정회원, 한국생산기술연구원 로봇기술연구부
***정회원, 국민대학교 전자공학과(교신저자)
접수일자: 2011.3.6, 수정일자: 2011.4.10
게재확정일자: 2011.4.15

된다[1][2]. 이러한 문제점은 차량의 전자제어시스템 적용이 확대되면서도 동일하게 발생되었으며 이를 극복하기 위하여 차량에서는 가능한 한 점대점(point-to-point)방식이 아닌 버스 구조의 네트워크가 보편적으로 적용되고 있다. 대표적인 차량 내 네크워크(In-Vehicle Network) 프로토콜에는 CAN(Controller Area Network), FlexRay 및 LIN등이 있으며 각 프로토콜의 특성에 따라 차량 내 파워트레인부, 샤시부 및 바디부에 적절히 사용되고 있다.

로봇의 경우에도 고성능, 지능형 로봇의 발전에 따라 다양한 특성의 센서 및 액츄에이터 그리고 다수의 제어기가 적용되면서 CAN등에 기초한 버스구조의 로봇 내부 네트워크가 구축되고 있다[3][4]. CAN기반 네트워크 시스템은 공장 자동화 분야뿐만 아니라 차량 내 네트워크에서 오랜 기간 동안 검증된 비교적 안정된 통신방식이다[5][6]. 그러나 로봇의 제어에서는 다수의 액츄에이터가 실시간으로 협조하여 동작해야만 주어진 임무를 달성할 수 있다는 차량에서와 다소 다른 특징이 있다. 또한, 차량에서와 같이 엔진룸 주변의 고열에 강인해야 한다는 등의 요구사항은 약하지만 관절 운동에 의한 버스 케이블의 구부러짐이나 휨 등의 물리적 변형에 강인해야 한다는 특징도 있다.

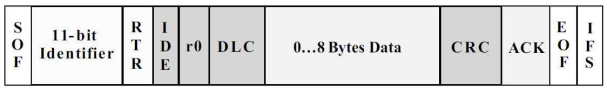
본 논문에서는 다수의 액츄에이터와 센서가 분산되어 있는 휴머노이드 로봇팔 내부의 제어 시스템 설계시, 각 관절의 통신노드와 중앙제어기 간의 네트워크 성능평가를 위한 분석시스템 RoNSpy(Robot Network Spy)를 설계 및 구현한다. 또한, 이 RoNSpy를 이용하여 네트워크 구조 및 통신주기에 따른 표준편차를 분석하여 로봇 내 네트워크의 실시간성 판별이 가능함을 실험을 통하여 보이기로 한다.

Ⅱ. 이론적 배경

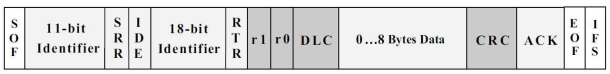
1. CAN(Controller Area Network)

CAN은 실시간 네트워크 기반 제어에 효과적으로 적용될 수 있는 직렬 네트워크 통신 프로토콜이며, 1986년 차량내의 전자제어장치(ECU : Electronic Control Unit) 간의 통신을 위해 차량 부품 업체인 Bosch가 개발하였다. 또한, ISO11898에 의해 표준화 되었고, CiA(CAN in Automation)협회에서는 CAN을 기반으로 하여 ISO/OSI 계층모델에서 응용 계층에 해당하는 통신망 프로토콜을 개발하였다. CAN 통신 프로토콜은 OSI 7계층 중 하위 2개의 계층, 즉 물리층과 데이터 링크 계층에 대해 정의한다.

임베디드 시스템에서 일반적으로 사용되는 표준 CAN(버전 2.0A) 프레임은 11-bit의 식별자를 지원해 이론적으로 2032개의 서로 다른 노드를 하나의 네트워크상에 연결하여 통신할 수 있며, 또한, 29-bit 의 식별자를 지원하는 확장 CAN(버전 2.0B)이 있다. CAN의 주요 특징으로는 비용이 저렴하고, 통신의 견고성이 극대화 되었고, 1Mbps의 빠른 데이터 전송 속도를 지원하며, 탁월한 오류 처리와 오류 제한을 통하여 통신의 신뢰성을 높였다. 또한, CAN은 결함 메시지들의 자동적인 재전송이 가능하고, 물리적으로 결함이 추정되는 노드들의 자동적인 버스 연결 절단이 가능하며, 메시지마다 식별자를 포함하여 그들의 기능과 우선순위에 연관된 식별자만을 선별하여 통신한다[7-10].



(a) 표준 CAN(버전 2.0A) : 11-Bit 식별자



(b) 확장 CAN(버전 2.0B) : 29-bit 식별자

그림 1. CAN 메시지 프레임
Fig. 1. CAN message frames.

2. CAN 기반 네트워크 시스템의 시간해석

이 장에서는 CAN기반 네트워크 시스템의 시간해석을 하고자 한다. 프로세서 응답시간의 계산은 최악의 경우 프레임의 큐잉 패턴의 범위 안에서 이루어진다. CAN 송신단의 메시지 m이 완전히 전송되어 수신단에 도착하는데 걸리는 최악응답시간은 식(1)과 같다.

$$R_m = J_m + w_m + C_m \tag{1}$$

여기서 R_m은 m의 WCRT(Worst Case Reponse Time)이고, J_m은 메시지 전송이 요청된 후 실제 대기열에 들어가 우선순위 경합을 시작하기 전에 생길 수 있는 시간지연 큐잉 지터를 나타내며, C_m은 메시지 m의

WCET(Worst Case Execution Time)이며, 이는 메시지 m이 순수하게 전송되는데 걸리는 가장 긴 시간을 나타낸다. w_m은 큐잉 지연시간으로 주어지고 식(2)와 같다.

$$w_m = B_m + \sum_{\forall j \in hp(m)} \left\lceil \frac{w_m + J_j + \tau_{bit}}{Tj} \right\rceil Cj \quad (2)$$

여기서 j는 같은 네트워크 내에서 메시지 m보다 높은 우선순위를 갖는 모든 메시지의 집합을 의미한다. τ_{bit}는 CAN의 bit time(1Mbps일 때 1us)을 나타낸다. B_m은 메시지 m보다 낮은 우선순위를 갖는 메시지로 인해 대기하게 되는 가장 긴 시간(blocking time)을 의미하고 식 (3)과 같이 나타낸다.

$$B_m = \max_{\forall k \in lp(m)} (C_k) \quad (3)$$

여기서 k는 메시지 m보다 낮은 우선순위를 갖는 모든 메시지의 집합이다. CAN에서는 낮은 우선순위의 메시지라도 먼저 버스를 점유하여 전송되고 있다면 이 메시지의 전송이 끝나야지만 다른 메시지의 전송이 시작된다. 따라서 B_m은 C_m과 같다(약 $155\tau_{bit}$).

C_m은 식(4)와 같이 나타내며, 이 시간은 frame overhead($47\tau_{bit}$), data contents($8s_m$), extra stuff bits($34+8s_m$)/5 로 표현된다.

$$C_m = \left(\left\lceil \frac{34 + 8s_m}{5} \right\rceil + 47 + 8s_m \right) \tau_{bit} \quad (4)$$

여기서 s_m은 메시지 m의 바이트 크기이다(최대 8byte)[11-12].

3. 제어시스템에 대한 실시간성 분석

CAN네트워크 시스템의 시간해석은 2.2절에서 살펴본 바와 같이 수행할 수 있다. 시간해석은 이론적인 분석이지만 실제 CAN네트워크 시스템에서의 실시간성을 방해하는 요소는 이론적인 분석보다 많은 원인들이 존재한다. 또한, 휴머노이드 로봇의 네트워크의 실시간성 보장에 있어 고려해야 하는 점으로 각 관절의 모터들의 이상, 긴급 상황의 발생 등의 비상시에 시스템의 진단, 모니터링 시스템이 제 기능을 발휘하게 된다. 이에 따라 본 논문에서는 로봇내 CAN기반의 네트워크의 실시간성을 감시할 수 있는 RoNSpy를 개발하여 휴머노이드 로봇 내부 네트워크의 실시간성을 평가분석 하고자 한다.

Ⅲ. RoNSpy 시스템 설계

1. RoNSpy의 시스템 구조

본 논문에서 구현하는 시스템은 크게 로봇의 각 노드의 CAN 네트워크에 직접 연결되어 데이터를 수신하는 CAN 인터페이스 카드와 수신된 데이터를 수집, 디스플레이 및 평가분석까지의 과정을 수행하는 PC부분으로 구분된다. RoNSpy의 시스템은 그림2와 같이 CAN 인터페이스 카드가 CAN 네트워크로부터 수신한 데이터를 펌웨어 계층을 통해 DPRAM으로 저장한다. 또한, PC에서의 32bit 윈도우즈 기반의 Application 계층은 API DLL 계층을 접근할 수 있는 기능을 지원하고, 이러한 API DLL 계층은, Driver DLL 계층과 Hardware driver 계층을 통해 데이터가 저장되어 있는 DPRAM에 접근할 수 있다[13-15].

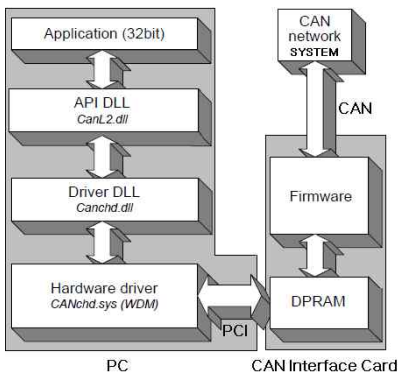


그림 2. RoNSpy 시스템 구조
Fig. 2. System structure of the RoNSpy.

2. RoNSpy의 기능

가. RoNSpy의 개념설계

사람과 같은 동작을 수행하려는 휴머노이드 로봇의 모든 제어는 실시간으로 이루어져야 한다. 예를 들어, 로봇이 뜨거운 물체에 반응하고자 한다면, 감각을 인지하는 온도센서의 데이터를 상위제어기로 전송한다. 상위제

어기는 센서를 통해 전달받은 데이터를 분석하여 현재 상황이 뜨거운 물체에 접촉해 있다는 상황을 인지하고 각 액츄에이터 노드들에게 뜨거운 물체에 반응하는 동작의 행동지령을 내리며, 지령 데이터를 수신한 각 액츄에이터 노드는 지령 데이터에 따라 액츄에이터를 구동한다. 이와 같은 로봇들의 동작수행 과정들은 사람이 무조건반사를 하듯이 실시간으로 제어가 가능하여야 한다.

로봇의 실시간 제어요소에는 중앙처리기의 데이터 처리능력, 구동 알고리즘의 수행속도, 액츄에이터의 동작속도 등 여러 가지 요소들이 뒷받침되어야 하지만 또한 중요한 요소로 강조되는 부분이 데이터를 전송하는 네트워크의 성능이다. 또한, 로봇의 구성이 점차 복잡해짐에 따라 모듈간의 통신을 통해 로봇을 제어하는 기술이 보편화 되면서 로봇 내 네트워크 통신의 실시간성 보장이 매우 중요한 요소가 되었다. 이에 따라, RoNSpy는 CAN기반 로봇 내 네트워크에서 송수신되는 데이터들의 전송주기 및 전송주기 오차, 오차 평균, 분산, 표준편차 분석을 통해 로봇 제어의 실시간성을 평가 분석하는 기능을 수행한다.

나. RoNSpy의 네트워크 관리 도구

RoNSpy의 네트워크 분석 도구는 CAN 메시지의 ID, 길이, 데이터, 주기 등 일반적인 CAN 분석기에서 기본적으로 확인 할 수 있는 정보들을 디스플레이 한다.

다. RoNSpy의 응용 모듈

RoNSpy의 응용 모듈은 로봇의 각 관절의 세부 데이터를 확인 및 분석하기 위한 기능이다. 전체 로봇 구성도를 보여주는 화면에서 세부노드로 진입하게 되면 각 노드의 송수신 데이터, 전송주기, 전송주기 오차에 대한 분석정보를 보다 상세하게 볼 수 있다.

여기서 전송주기의 오차는 표준편차를 기준으로 연산되며, 관측값의 평균값을 m이라 하면, 표준편차의 식은 식(5)와 같이 세울 수 있다.

$$\sigma = \sqrt{\frac{\sum_{k=1}^{n}(x_k - m)^2}{n}} = \sqrt{\frac{\sum_{k=1}^{n} x_k^2}{n} - m^2} \quad (5)$$

또한, 그림 3을 통해 RoNSpy의 전체적인 소프트웨어 구조를 확인 할 수 있다.

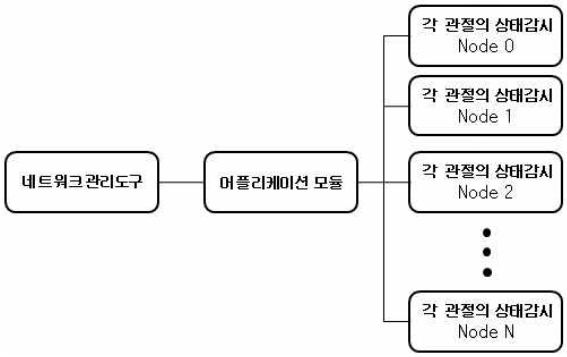


그림 3. RoNSpy의 소프트웨어 구조
Fig. 3. Software structure of the RoNSpy.

Ⅳ. RoNSpy 시스템 구현

1. 시스템의 환경 구성

윈도우즈 기반의 편리한 인터페이스 환경을 구축할 수 있고, 주변기기의 확장이 쉬운 범용 PC를 중심으로 구성하였으며, PC 내 확장슬롯에 Softing사의 CAN Interface Card를 장착하여 RoNSpy의 중앙제어기를 구성한다.

또한, 실시간으로 네트워크를 분석하기 위해서는 CAN통신을 통하여 전달된 데이터를 종류에 따라 구분하여 각각의 형식에 맞추어 확인할 수 있는 사용자 인터페이스를 필요로 한다. 사용자 인터페이스는 Code- Gear 개발 환경에서 C++언어를 기반으로 구현하며 사용자 인터페이스 상에 디스플레이 되는 데이터들은 CAN Interface Card를 통하여 PC로 수신된다. 또한, 수신된 데이터를 RoNSpy에서 분석하여 그 결과를 인터페이스에 디스플레이 한다. 전체적인 시스템 환경 구성도는 그림 4와 같다.

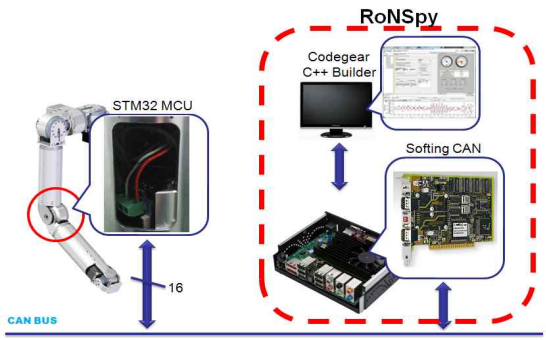


그림 4. 시스템의 적용 환경
Fig 4. Application Environment of the RoNSpy.

2. 휴머노이드 로봇 설계

최종적으로 분석할 로봇은 그림 5와 같이 8자유도의 총 16개의 노드를 갖는 휴머노이드 로봇팔이다. 그러나 본 논문에서는 우선, 팔목과 어깨 두 개의 관절을 가정하여 액츄에이터 노드 2개, 센서노드 2개씩 총 4개의 CAN 통신노드를 이용한 실험환경을 구현하기로 한다.

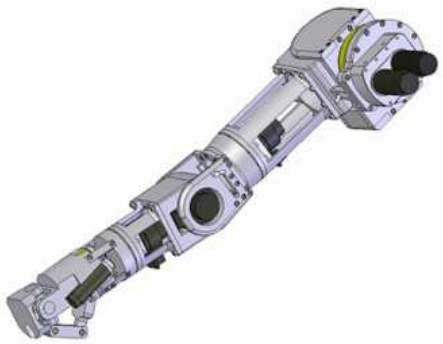


그림 5. 휴머노이드 로봇팔
Fig. 5. Humanoid robot arm.

V. 실험 및 결과

본 논문에서 실험환경은 그림 6과 같이 범용 PC(main controller)와 팔목, 어깨 각 관절마다 액츄에이터 노드 1개, 센서노드 1개를 가정한 총 4개의 CAN 통신노드를 연결한 실험환경을 구현하여 자유도 마다 모듈화 하고 분산형 구조로봇 형태의 시스템을 구성하여 CAN네트워크의 통신 지연 및 오차를 분석, 감시하여 네트워크의 실시간성을 평가 및 검증하기로 한다. 실험에서 기본적인 설정은 다음과 같다.

[실험설정]

· 표준 CAN : 11-bit 식별자
· CAN 전송주기 : 4msec
· CAN baud rates
  1Mbps, 500kbps, 250kbps. 50kbps
· 데이터 길이 : 8byte
· 통신케이블 길이 : 70cm(로봇팔의 길이)

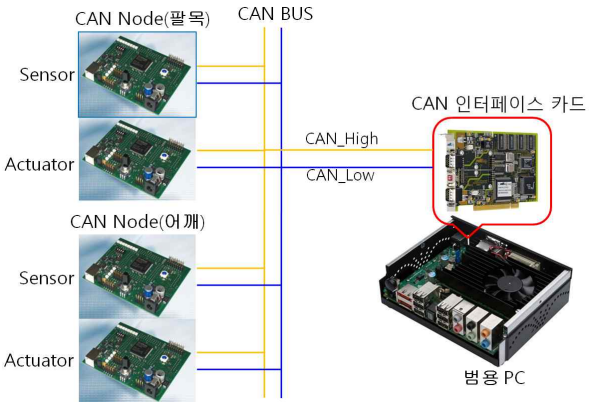


그림 6. 실험을 위한 통신 환경
Fig. 6. Communication environment for experiment.

데이터 전송주기는 각 전송속도에 따른 최악의 데이터 전송시간이 보장되어야 한다. 2절의 식(4)를 통해 산출한 전송시간에 따른 최악의 데이터 전송시간을 살펴보면 표 1에서 확인할 수 있듯이 최대 2.62ms까지 보장이 되어야 하므로 본 논문의 실험설정에서 최악의 데이터 전송시간의 보장을 위해 CAN 전송주기를 최악 메시지 전송시간 보다 큰 4msec로 설정하였다.

표 1. 최악 메시지 전송시간
Table 1. WCET(Worst Case Execution Time)

C_m	1Mbps	500kbps	250kbps	50kbps
(ms)	0.131	0.262	0.524	2.62

RoNSpy를 통해 측정한 데이터를 살펴보면, 네트워크 관리도구에서는 ID, DLC(Data Length Code), DATA 및 메시지 전송주기를 그림 7과 같이 확인 할 수 있다.

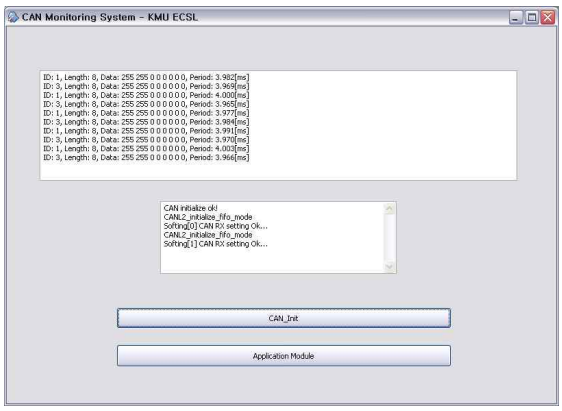


그림 7. RoNSpy의 네트워크 관리 도구
Fig. 7. Network management tool of RoNSpy.

또한, 응용 모듈을 통해 로봇 내부의 각 관절 부위를 세부적으로 들여다 볼 수 있다. 그림 8은 어깨의 관절부 통신 데이터를 측정하는 상태이며, 센서와 액츄에이터 노드로 구분되어 각 노드의 ID, 데이터 전송주기 그리고 전송주기에 따른 표준편차를 확인할 수 있다.

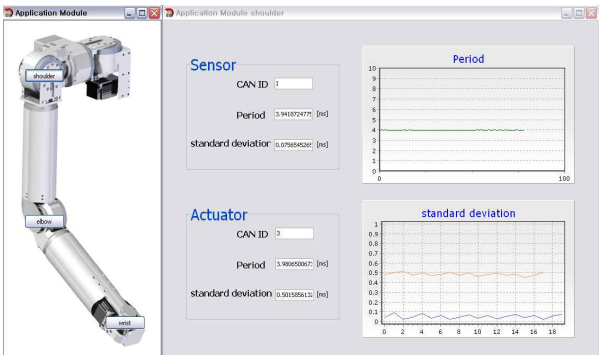


그림 8. RoNSpy의 응용 모듈
Fig. 8. Application module of RoNSpy.

이와 같은 실험환경을 통하여 실험 네트워크의 환경에서 전송속도(Buad Rate)에 따른 네트워크의 전송시간의 표준편차를 측정하고, 표준편차의 크기에 따른 네트워크 실시간성을 비교분석한다.

CAN 통신 시스템에서 각 전송속도에 따른 실제 전송시간을 측정하고 그 표준편차를 구하면 그림 9와 같다. 여기서, 전송속도는 50kbps부터 1Mbps까지이며, 1Mbps, 250kbps, 500kbps, 50kbps의 순서로 CAN 네트워크 전송주기 오차가 낮으므로 1Mbps일 때 실시간성이 가장 높다고 할 수 있다.

그러나 전송속도에 따른 오차의 차이가 최대 약9us로 매우 근소한 차이이기 때문에 이 시스템에서는 모든 전송속도에서 실시간성이 보장된다고 판단할 수 있다.

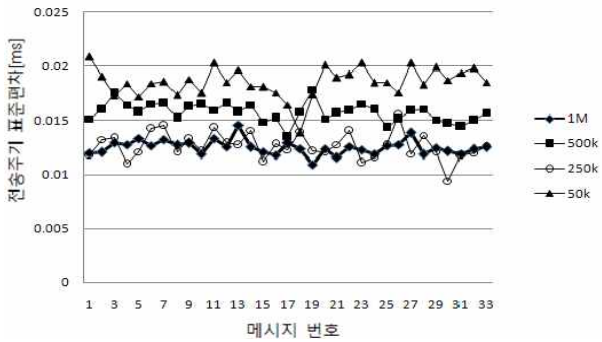


그림 9. 전송속도에 따른 전송시간의 표준편차
Fig. 9. Standard deviations of each transmission time according to CAN baud rates.

VI. 결 론

로봇 내의 네트워크 시스템에서는 다양한 데이터들이 하나의 네트워크를 공유하므로 만일 네트워크 시스템 또는 시스템 구동 알고리즘이 부적절하게 설계된다면 실시간 데이터의 전송지연시간이 미리 지정된 한계치를 초과하여 네트워크가 적용된 응용 시스템의 성능을 저하 시키거나, 불안정한 상태에 도달할 수 있다.

본 논문에서는 이러한 문제를 분석하고 효과적으로 대처하기 위하여 로봇 내 CAN 통신 시스템 데이터를 분석하고, 다양한 시각적 효과를 이용하여 실시간으로 확인 가능하도록 구현한 CAN 네트워크 분석 시스템 RoNSpy를 개발하였다. 또한, RoNSpy를 이용하여 휴머노이드 팔의 일부 노드를 가정한 실험환경을 구성하고 실제 실험을 통해 RoNSpy의 분석 기능을 검토하였다. 앞으로 RoNSpy에 실제 휴머노이드 로봇팔 전체 노드를 연결하고 CAN 기반 로봇 내부 통신시스템의 실시간성 및 신뢰성을 분석할 것이다.

참 고 문 헌

[1] 임을균, 조재일, 황대환, "지능형 로봇 내부 실시간 제어 통신망 기술," ETRI, 전자통신동향분석 제 20권 제 2호, 21~32쪽, 2005

[2] 원지운, 김태환, 홍원기, 이용두, "CAN 네트워크에서의 동적 마스터 시스템에 관한 연구," 한국정보과학회 학술발표 논문집 제32 제2호, 559~561쪽, 2005

[3] 구자봉, 허욱열, 김진걸, "휴머노이드 로봇에 대한 CAN(Controller Area Network) 적용," 대한전기학회, 정보 및 제어 심포지엄 논문집, 77~79쪽, 2004

[4] 구자봉, 허욱열, 김진걸, 김병륜, "CAN내장 휴머노이드 로봇에 대한 진보된 우선순위 적용," 대한전기학회 논문지 제 53권 10호, 714~719쪽, 2004

[5] Othman, H.F., Aji, Y.R., Fakhreddin, F.T., Al-Ali, A.R., "Controller Area Networks: Evolution and Applications," Information and Communication Technologies, 2006

[6] Xiaofeng Yin, Qiulin Zhao, Jingxing Tan,

"Development of VeNetSpy: a Generic Monitoring Software for CAN-based ECUs and In-Vehicle Networks," SAE 2010 World Congress & Exhibition, Detroit, MI, USA, 2010
[7] BOSCH, "CAN Specification Ver2.0," 1991
[8] Embedded Systems Korea, "CAN/LIN 기술정보,"
[9] Steve Corrigan, "Introduction to the Controller Area Network," Texas Instruments, 2002
[10] Gerhard Goller, "Atmel Wireless & Microcontrollers CAN Tutorial," Atmel, 2000. 10
[11] Tindell, K, Burns, A., "Guaranteed Massage Latencies for Distributed Safety Critical Hard Real-Time Networks," YCS 229, Dept, Computer Science, Univ. of York, 1994
[12] K.Tindell. A. Burns and a Wellings, "Calculating Controller Area Netrwork(CAN) Message Response Times," IFAC workshop on Distributed Computer Control Systems(DCCS), Toledo, Spain, 1994
[13] 안봉주, 박병률, 기영훈, 안현식, 정구민, 김도현, "CAN 기반 제어시스템의 시간해석을 위한 모니터링 툴 개발," 한국자동차공학회, 춘계학술대회 논문집 Vol.3 1742~1747쪽, 2006
[14] Renjun Li, Chu Liu, Feng Luo, "A design for automotive CAN bus monitoring system," Vehicle Power and Propulsion Conference, 2008
[15] 양승현, 이석원, "Controller Area Network (CAN 통신과 임베디드 시스템을 이용한 자동차 감시 시스템 구현," 대한전기학회, 대한적기학회 학술대회 논문집, 2690~2692쪽, 2005

※ 본 연구는 지식경제부 산업원천기술개발산업(No.10033788-2010-12, 로봇 H/W 개발 솔루션, 기능별 모듈 및 응용기술 개발)의 지원과 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2011-C1090-1121-0005)

저자 소개

이 경 중(준회원)

- 2010년 국민대학교 전자공학부 학사 졸업
- 2010년~현재 국민대학교 대학원 전자공학부 석사과정

<주관심분야 : USN 기술, 로보틱스 응용>

김 재 오(준회원)

- 1994년 국민대학교 전자공학부 학사 졸업
- 1996년 국민대학교 전자공학부 석사 졸업
- 2008년~현재 국민대학교 대학원 전자공학부 박사과정

<주관심분야 : USN 기술, 로보틱스 응용>

박 재 한(정회원)

- 부산대학교 전자공학과 공학석사
- 2010년 고려대학교 전기전자전파공학과 박사수료
- 현재 한국생산기술연구원 융복합 연구부문 지능형로봇연구그룹 선임연구원

<주관심분야 : RFID/USN 기술, 이동무선통신, 안테나 및 전파전파, 통방융합기술>

백 문 홍(정회원)

- 1995년 동경대학교 전기전공 공학 박사
- 현재 한국생산기술연구원 융복합 연구부문 지능형로봇연구그룹 수석연구원

<주관심분야 : 3차원 물체인식, 3차원 공간인지>

안 현 식(정회원)

- 1992년 서울대학교 제어계측공학과 박사
- 현재 대한전기학회 정보 및 제어부문 이사
- 현재 국민대학교 전자공학부 교수

<주관심분야 : 지능형 로보틱스 및 차량 전자 제어>