윈도우즈 XP 시리얼 통신에 실시간성 지원 방법

이승훈°, 송창인*, 주민규*, 이철훈*
°*충남대학교 컴퓨터공학과
e-mail:{sh2011, jjami1234, mingyuman, clee}@cnu.ac.kr

A method to support real-time for serial communication on Windows XP

Seung-Hoon Lee°, Chang-In Song*, Min-Gyu Ju*, Cheol-Hoon Lee*
°*Dept. of Computer Science, Chung-Nam University

● 요 약 ●

네트워크 및 통신 기술의 발전에 따라 임베디드 시스템은 정보통신 기술 및 유무선 네트워크 기술을 적극 수용하면서, 지능형 정보 시스템의 핵심으로 성장하고 있으며, 시리얼 통신은 이러한 임베디드 시스템에서 사용되는 통신 시스템의 일종이다. 윈도우즈 운영체제는 개발환경의 편의성 및 호환성 때문에 임베디드 시스템에서 많이 사용되는 운영체제이다. 하지만 윈도우즈 운영체제는 실시간성을 제공하지 않으며, 윈도우 시스템에서 사용하는 시리얼 통신도 실시간성을 보장하지 못한다. 본 논문에서는 윈도우즈에 디바이스 드라이버 형태로 이식되어 유저 영역에 실시간성을 지원하는 이벤트 기반 RTiK(Real Time implanted Kernel)을 사용하여 시리얼 통신에 실시간성을 지원 할 수 있는 방법을 설계 및 구현하였다.

키워드: 시리얼 통신(Serial Communication), 윈도우즈(Windows), 실시간성(Real-Time)

I. 서론

산업용 제어기기 및 무선시스템에서 기술적인 발전을 추구했던 임베디드 시스템은 정보통신 기술 및 유무선 네트워크 기술을 적극 수용하면서 정보서비스, 군사, 우주항공 및 자동차 분야에서 지능형 정보 시스템의 핵심으로 성장해 나아가고 있다. 이러한 임베디드 시스템에서는 사용자에게 개발 환경의 편의성 및 호환성을 제공하기 위해 윈도우즈 운영체제를 이용하여 개발 및 운영되고 있다. 하지만 윈도우즈 운영체제는 독립적으로는 실시간성을 제공하지 못한다. 윈도우즈에 실시간성을 제공하기 위해 RTX나 INTime 같은 써드파티 운영체제가 사용되며, 이와 같은 제품들은 고가의 구입비용과 라이선스 비용으로 인해 개발 비용의 증가를 초래할 수 있다. 시리얼 포트를 통한 통신은 저렴한 통신 케이블과 소형 커넥터를 사용할 수 있는 이점을 가지고 있어 다양한 임베디드 시스템에서 사용된다. 하지만 컴퓨터의 시리얼 포트를 통한 시리얼 통신은 비동기 통신이기 때문에 주기적인 통신이 이루어지지 않으며, 윈도우즈는 시리얼 통신에 대해 실시간성을 보장하지 않는다. 따라서 개발 비용을 절감하며, 윈도우즈 시리얼 통신에 실시간성을 지원 할 수 있는 방법에 대한 연구가 필요하다.

본 논문에서는 x86기반의 윈도우즈 XP에서 유저영역에 실시간성을 지원하는 이벤트 기반 RTiK을 이용하여 시리얼 통신에 실시간성을 제공 할 수 있는 방법을 연구하였다. 본 논문의 구성은 2장에서 관련 연구로 시리얼 통신에 대해 설명하고, x86 기반의 윈도우즈에 실시간성을 제공하는 RTiK 및 유저 영역에 실시간성을 제공하는 이벤트 기반 RTiK에 대해 기술한다, 3장에서는 x86기반의 윈도우즈 XP 상에서 시리얼 통신에 실시간성을 제공하기 위한 방법과 실험 환경 및 결과를 기술하였으며, 마지막 4장에서 결론을 맺는다.

II. 관련 연구

1. 관련연구

1.1 시리얼 통신

시리얼 통신이란 일반적으로 컴퓨터 기기를 접속하는 방법의 하나로, 제어기의 값을 컴퓨터로 전송하여 사용자에게 보여줄 수 있는 가장 기본적인 수단이며, 이때 신호가 1비트마다 직렬로 보내지기 때문에 시리얼 통신이라 부른다. 시리얼 통신에는 동기와 비동기 통신의 두 종류의 통신방식이 있다. 동기 통신은 두 개의 디바이스 사이에서 동기 신호를 발생시켜, 그 신호에 맞추어 데이터를 송수신 한다. 데이터의 교환이 없는 동안에도 제어 신호가 흐르고 있으므로 두 디바이스는 동기를 유지하는 것이 가능하다. 실 데이터를 송신 할 때는 그것을 수신하고, 데이터가 없을 때에는 대기 상태를 나타내는 신호를 교환한다. 이처럼 통신이 확립되면 실 데이터를 송수신한 것에 데이터의 시작과 종료를 나타내는 신호가 존재하지 않기 때문에 데이터 전송 속도는 빠르다. 비동기 통신은 송신과 수신 대기 상태를 나타내는 신호가 필요 없다. 그렇지만 데

이터의 처음과 끝에는 반드시 데이터의 개시와 종료를 나타내는 스타트 비트(start bit)와 스톱 비트(stop bit)가 붙는다. 따라서 이들 두 비트의 추가 때문에 비동기 통신의 속도는 동기 통신에 비교하여 약간 늦어진다. 그러나 프로세서는 대기 상태에서 여분의 유휴 문자를 처리할 필요가 없어진다. 컴퓨터의 시리얼 포트는 비동기 시리얼 통신만 지원한다[1-2].

1.2 RTiK

RTiK은 x86 기반의 윈도우즈에 실시간성을 제공하기 위한 기술이며, [그림 1]에서와 같이 윈도우즈 디바이스 드라이버 형태로 이식된다. RTiK은 커널 레벨에서 동작하게 함으로써 하드웨어 접근이나 윈도우즈 커널 자원의 접근을 가능하게 한다[3].

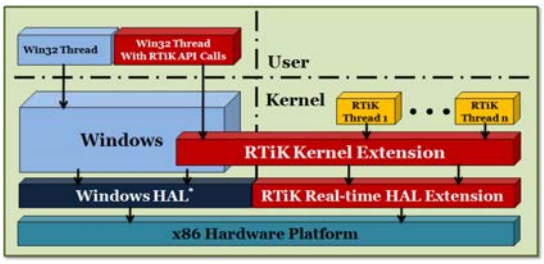

그림 1. RTiK 구조
Fig. 1. RTiK Architecture

1.3 이벤트 기반 RTiK

이벤트 기반 RTiK은 윈도우즈 유저 레벨에서 실시간성을 지원하기 위해 RTiK을 이용한 방법으로 구조는 [그림 2]와 같다. 사용자는 이벤트 기반 RTiK으로부터 제공되는 API를 이용하여 실시간 쓰레드의 수행코드를 작성하고, 생성된 실시간 쓰레드는 커널 영역으로부터 전달 될 신호를 기다린다. 이벤트 기반 RTiK의 타이머가 활성화 되면 사용자가 설정한 주기로 타이머 인터럽트가 발생하며, 지연 처리 호출 과정에서 유저영역에 신호를 전달해 주게 된다. 이때 유저 레벨의 실시간 쓰레드는 윈도우즈에서 관리하는 프로세스 및 쓰레드의 우선순위 중 가장 높은 우선순위로 동작한다. 따라서 윈도우즈에서 동작하는 다른 프로세스 및 쓰레드의 영향을 받지 않고 설정 된 주기를 지키며 동작함으로써 유저영역에 실시간성을 제공해 줄 수 있다[4].

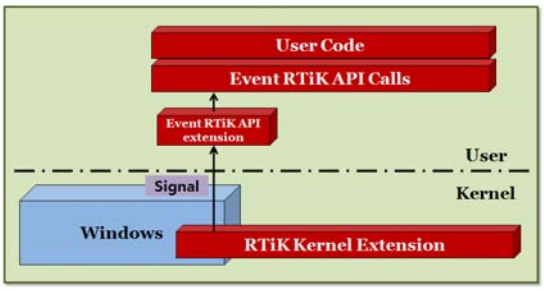

그림 2. 이벤트 기반 RTiK 구조
Fig. 2. Event Based RTiK Architectur

III. 본론

1. 본론

1.1 윈도우즈 XP 시리얼 통신에 실시간성 지원 방법의 설계 및 구현

윈도우즈 XP 시리얼 통신은 [그림 3]과 같이 윈도우즈에서 제공하는 멀티미디어 타이머를 사용하여 시리얼 포트를 통해 데이터를 전송하며, 시리얼 포트 버퍼에 데이터를 넣게 된다. 수신은 시리얼 포트 버퍼를 감시하여 데이터를 읽어온 후 버퍼를 초기화하는 방식으로 동작한다[5-6]. 이와 같이 비동기 시리얼 통신은 멀티미디어 타이머를 이용한 폴링 방식으로 통신을 제어하며, 멀티미디어 타이머는 10ms 이하로 주기 설정 시 정확한 주기 안에서의 실행을 보장하지 못하기 때문에 실시간성이 보장되지 않는다. 또한 송신 컴퓨터와 수신컴퓨터의 주기가 일치하지 않는다면 시리얼 포트 버퍼에 오버플로우가 발생하거나, 정확하지 않은 데이터를 읽어오게 되어 문제가 된다.

윈도우즈 XP 시리얼 통신에 실시간성을 지원하기 위해서는 윈도우즈 운영체제에 실시간성을 지원해 주기 위한 방법이 필요하며, 윈도우즈 운영체제는 커널 코드가 비공개이기 때문에 커널 영역에서 시리얼 통신을 제어 할 수 없다. 이와 같은 이유로 본 논문에서는 앞서 설명한 이벤트 기반 RTiK을 이용하여 윈도우즈 XP 시리얼 통신에 실시간성을 지원 할 수 있는 방법을 설계 및 구현하였다.

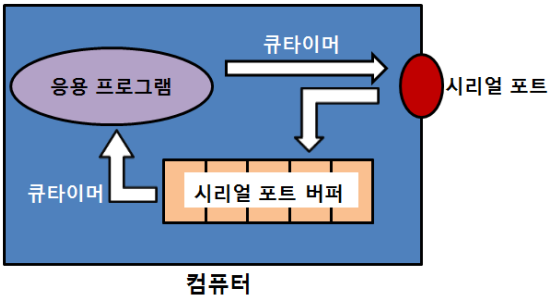

그림 3. 시리얼 통신
Fig. 3. Serial Communication

RTiK을 이용한 시리얼 통신 제어 방법은 [그림 4]와 같이 유저영역에서 윈도우즈 API를 사용하여 호스트와 타겟 컴퓨터의 시리얼 포트를 열고, RTiK API를 사용하여 시리얼 통신에 사용되는 주기를 설정한다. 설정한 주기로 시그널을 받아 동작하게 되는 유저 레벨 실시간 쓰레드에는 데이터 송신과 수신 코드가 작성된다. 데이터를 수신하게 되는 타겟 컴퓨터의 RTiK 타이머를 먼저 활성화 시킨 후 데이터를 송신하는 호스트 컴퓨터의 RTiK 타이머를 활성화 시킨다. 이 방법을 통하여 윈도우즈 XP 시리얼 통신이 주기적으로 송신과 수신을 하게 되며 실시간성을 보장받게 된다.

타겟 컴퓨터의 RTiK 타이머를 먼저 활성화 해주는 이유는 시리얼 포트 버퍼의 오버플로우를 막기 위한 것이며, 타겟 컴퓨터를 먼저 활성화 시킨다고 하여도 시리얼 포트 버퍼에 값이 들어오지

않으면 데이터를 읽지 않기 때문이다.

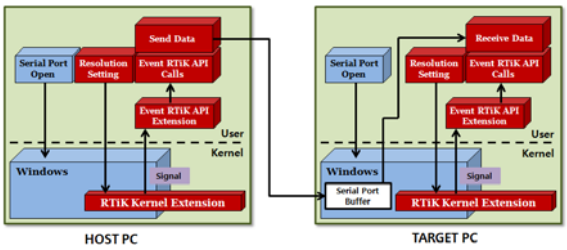

그림 4. Event RTiK을 이용한 시리얼 통신
Fig. 4. Serial Communication with Event RTiK

1.2 실험 환경 및 결과

윈도우즈 XP 시리얼 통신이 주기적으로 정상 동작하는지 확인하기 위해 [그림 5]와 같이 호스트 컴퓨터와 타겟 컴퓨터를 시리얼 케이블로 연결하여 시스템을 구성하였다.

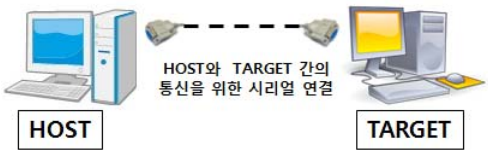

그림 5. 시스템 구조
Fig. 5. System Architecture

호스트와 타겟 컴퓨터의 환경은 [표 1]와 같으며, 실험 방법은 호스트 컴퓨터에서 주기적으로 일정한 길이의 데이터를 송신하고, 타겟 컴퓨터에서 일정한 길이의 데이터를 주기적으로 수신하는지 확인한다. 이와 같은 방법을 통해 윈도우즈 XP 시리얼 통신에 실시간성이 지원되는 것을 보여준다.

표 1. 호스트 및 타겟 PC 환경
Table 1. Host and Target PC environment

항목	호스트 PC	타겟 PC
CPU	Pentium(R) 4 CPU 3.00 GHz	Pentium(R) 4 CPU 2.66 GHz
OS	Microsoft Windows XP Professional K Version 2002 Service Pack 3	Microsoft Windows XP Professional Version 2002 Service Pack 2
Development tool	Microsoft Visual Studio 2008	Microsoft Visual Studio 2005

실험은 4ms로 8개의 데이터를 1000번 송수신하여 측정하였다. 수신 시 데이터를 읽어 시리얼 포트 버퍼에 넣게 되고, 버퍼에 있는 데이터를 화면에 출력하는데 4ms가 소요되기 때문에 시리얼 통신 시 최소 주기는 4ms가 되며, [그림 6]과 [그림 7]은 송수신 시 주기 설정 및 데이터 확인 결과이다.

그림 6. 송신 주기 설정 및 송신 데이터 길이
Fig. 6. Send Resolution Setting and Send Data Length

그림 7. 수신 주기 설정 및 수신 데이터 길이
Fig. 7. Receive Resolution Setting and Receive Data Length

[그림 8]과 [그림 9]는 4ms로 송신 및 수신 시 실행 주기를 측정한 실험 결과이다.

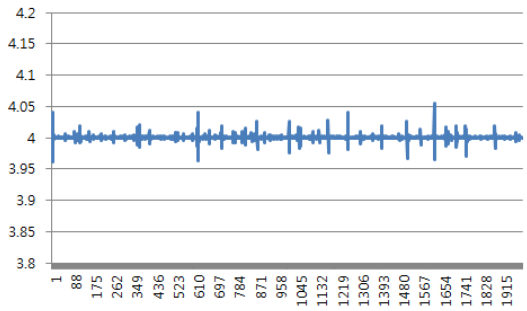

그림 8. 송신 주기를 4ms로 설정 후 결과 화면
Fig. 8. Result of 4ms-Cycle Send Test

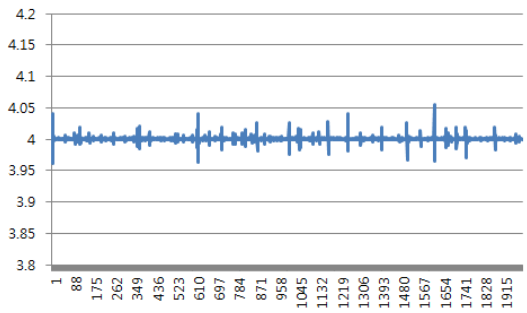

그림 9. 수신 주기를 4ms로 설정 후 결과 화면
Fig. 9. Result of 4ms-Cycle Receive Test

표 2. 4ms 주기로 통신 시 실험 결과
Table 2. Result of 4ms-Cycle Communication test

	최대값	최소값
송신	4.054641	3.964425
수신	4.041833	3.933853

[표 2]에서 볼 수 있듯이 시리얼 포트를 통한 송수신시 1%의 오차를 가지는 것을 확인할 수 있다. 이를 통해 설정한 주기 값을 크게 벗어나지 않고 시리얼 통신이 동작하는 것을 확인할 수 있다.

Ⅳ. 결론

시리얼 통신은 저렴한 통신 케이블과 소형 커넥터를 사용하기 때문에 임베디드 시스템에서 많이 사용된다. 임베디드 시스템의 특성상 시리얼 통신을 통해 주기적으로 정확한 데이터를 송신 및 수신하여 시스템을 동작시키기 위해서 시리얼 통신에 실시간성이 요구된다. 그러나 임베디드 시스템에서 많이 사용되고 있는 윈도우즈 운영체제는 독립적으로 실시간성을 제공하지 못하며, 윈도우즈 시스템에서 사용하는 시리얼 통신에 실시간성을 보장하지 못하는 문제점이 있다. 이에 따라 본 논문에서는 x86 기반의 윈도우즈 유저 영역에 실시간성을 제공해 주는 이벤트 기반 RTiK을 이용하여 윈도우즈 XP 시리얼 통신에 실시간성을 지원하기 위해 방법을 설계 및 구현하였으며, 실제 실험을 통하여 시리얼 통신에 실시간성을 보장 할 수 있음을 검증하였다. 향후 연구과제로는 임베디드 시스템에서 사용하는 다양한 통신 시스템에 실시간성을 제공하기 위한 연구가 필요하며, 윈도우즈 7 상의 시리얼 통신에 실시간성을 지원해 주기 위한 방법이 필요하다.

참고문헌

[1] D.A.Godse and A.P.Godse, “Microprosessors”, Technical Publications Pune, pp.432-472, 2007

[2] O. Bailey, “Embedded systems : desktop intergration”, Wordware Publishing, Inc., pp.34-56, 2005

[3] jwlee, mhcho, jjkim, hmjo, yspark, chlee, “Design and Implementation of Real-time Implanted Kernel, RTiK to Support Real-Time for a Test Set based on Windows,” Journal of The Korea Contents Association, Vol. 10, No. 10, pp. 36-44, Oct. 2010.

[4] mgju, jwlee, csjang, shkim, chlee “A Method to Support Real-time for User-level Robot Components on Windows,” Journal of The Korea Contents Association, Vol. 11, No. 7, pp. 36-44, Jul. 2011.

[5] J. Axelson, “Serial Port Complete : Programming and Circuits for RS-232 and RS 485 Links and Networks”. Lakeview Research, pp75-90, 1998

[6] Windows Multimedia Timers, http://msdn.microsoft.com/en-us/library/windows/desktop/dd743609