

마이크로컨트롤러 TCP/IP 스텝 모터 위치제어기 구현

윤태성, 김선길, 이치환
위덕대학교 정보전자공학과

Implementation of the TCP/IP Step Motor Motion Controller

Tae Seong Yun, Seon Gil Kim, and Chi Hwan Lee
Dept. of Information & Electronic Eng., Uiduk Univ.

ABSTRACT

In this paper, the microcontroller TCP/IP NIE(Network Interface Equipment) which is exclusive of an operating system is implemented and the applied example to a step motor motion controller is then explained. The microcontroller TCP/IP NIE supplies a DHCP(Dynamic Host Configuration Protocol) as well as basic TCP/IP network protocols. So, the automatic setting up of the Dynamic IP(Internet Protocol) is possible. To implement a control of a system with an internet, the microcontroller TCP/IP NIE is applied to a step motor motion controller and a step motor is then controlled with a Microsoft Windows-based Telnet application program.

1. 서 론

산업체에서 일반적으로 사용되는 통신 표준은 RS-232에서 TCP/IP 통신으로 전환되었다. 그 결과, 시스템의 제어 및 관측이 인터넷으로 이루어짐으로써 엄청난 시간과 비용이 절감되었다. 현재 많은 종류의 NIE가 있으며, 그 중 운영체제를 사용한 TCP/IP NIE는 복잡한 하드웨어와 소프트웨어로 구성되어 있다. 그러나 운영체제를 배제한 마이크로컨트롤러 TCP/IP NIE는 간단한 하드웨어와 소프트웨어로 구현이 가능하다^{[1],[2]}. 본 논문에서는 TCP/IP NIE의 개발비용 및 원가절감을 위해 운영체제를 배제한 간단한 하드웨어와 소프트웨어의 구성만으로 TCP/IP NIE를 구현한다. 그리고 TCP/IP 통신으로 마이크로컨트롤러 TCP/IP NIE를 적용한 시스템의 제어를 구현하기 위해 스텝 모터 위치 제어기에 마이크로컨트롤러 TCP/IP NIE를 적용한다. 그런 다음 PC에서 Windows 기반의 텔넷 응용 프로그램을 이용하여 스텝 모터 위치 제어를 제어한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 마이

크로컨트롤러 TCP/IP NIE의 하드웨어 구성 요소와 송·수신되는 프레임 및 소프트웨어에 대해 기술한다. 3장에서는 마이크로컨트롤러 TCP/IP NIE를 구현한 다음 텔넷 응용 프로그램으로 TCP/IP 통신이 이루어짐을 확인한다. 그런 다음 스텝 모터 위치 제어기에 적용하여 TCP/IP 통신을 이용한 스텝 모터의 위치제어를 구현 및 검증한다. 마지막으로 4장에서 본 논문의 결론을 맺는다.

2. 마이크로컨트롤러 TCP/IP NIE

2.1 하드웨어 구성 요소

본 논문에서 구현한 마이크로컨트롤러 TCP/IP NIE의 물리적 구성 요소는 전송매체, 트랜스 & LED 내장형 RJ-45 커넥터, 이더넷 컨트롤러, 마이크로컨트롤러, 그리고 그래픽 LCD로 이루어진다. 전송매체로는 다른 매체에 비해 가격이 낮고 무게가 가벼우며 유연성이 좋아 일반적으로 널리 사용되고 있는 UTP(Unshielded Twisted-Pair) 케이블을 사용한다. UTP 케이블은 100Ω의 임피던스를 가지고 있으며, 이진 신호를 감쇠 없이 100m까지 전송한다^[1].

$$\left(\frac{\text{Primary turns}}{\text{Secondary turns}}\right)^2 = \frac{\text{Primary Impedance}}{\text{Secondary Impedance}} \quad (1)$$

신호원으로부터 최대전력을 얻기 위해서는 임피던스 정합이 필요하다. 식 (1)은 1차 측과 2차 측의 권선수와 임피던스의 관계를 나타내며, 식 (1)에 의해 트랜스의 권선비를 1:1로 설계하고 UTP 케이블의 임피던스와 동일한 임피던스를 삽입함으로써 임피던스를 정합시킨다^[1]. LED는 프레임의 송·수신 및 충돌을 표시하기 위해 사용한다.

그림 1은 권선비가 1:1로 설계된 트랜스 & LED 내장형 RJ-45 커넥터이며, 본 논문에서 하드웨어를 최대한 간략화 하기 위해 사용한다.

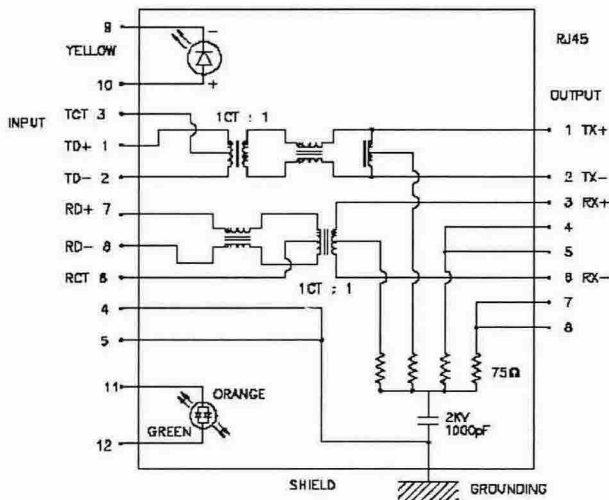


그림 1 트랜스 & LED 내장형 RJ-45 커넥터
Fig. 1 Trans & LED embedded RJ-45 Connector

UTP 케이블로 전송되는 신호는 맨체스터 인코딩된 신호이며, 이더넷 컨트롤러 내부에 있는 디코더에 의해 이진 신호로 복조된다. 이와 반대로 이진 신호를 송신할 때에는 인코더에 의해 맨체스터 인코딩된 신호로 변환된다. 수신된 이진 신호(이더넷 프레임)는 이더넷 컨트롤러 내부의 MAC(Media Access Control) 엔진에 의해 정보에 해당되는 부분만 내부 버퍼로 저장된다. 버퍼에 저장된 정보는 원격 DMA(Direct Memory Access) 방식으로 마이크로컨트롤러의 내부 메모리로 저장한다. 전송매체로 전송되는 데이터는 이더넷 컨트롤러에 의해 하나의 완성된 이더넷 프레임이 된다. 그 외, 이더넷 컨트롤러는 수신된 프레임의 에러를 검출하기 위한 CRC(Cyclic Redundancy Checksum), 충돌 감지를 위한 CSMA/CD(Carrier Sense Multiple Access/Collision Detection) 등 프레임의 송·수신을 위해 필요한 부분을 모두 지원한다^{[1],[3]}. 본 논문에서는 일반적으로 가장 많이 사용되고 보급화 된 칩 중 하나인 Realtek사의 RTL8019AS 이더넷 컨트롤러를 사용한다. RTL8019AS는 TCP/IP 통신을 위해 요구되는 모든 기능을 지원한다^[3].

마이크로컨트롤러는 이더넷 컨트롤러의 초기화 및 레지스터 설정에 필요한 Boot-ROM 역할과 데이터의 송·수신 및 정보처리를 위한 프로토콜 컨트롤러 역할을 병행한다. 그 외, 마이크로컨트롤러 TCP/IP NIE의 상태를 표시하기 위해 그래픽LCD의 문자 출력을 제어한다. 본 논문에서는 128K Bytes의 ISP(In-System Programmable) 플래쉬 메모리가 내장된 Atmel사의 ATmega128 마이크로컨트롤러를 사용한다. ATmega128은 8-bit 마이크로컨트롤러로 RISC 구조를 가지며, 16 MIPS의 명령 처리 속도를 가진다. 그 외, 4K Bytes의 EEPROM

과 SDRAM을 가진다^[4]. 그래픽 LCD는 현대의 HG12605NY-LY(128 × 64) 모델을 사용한다.

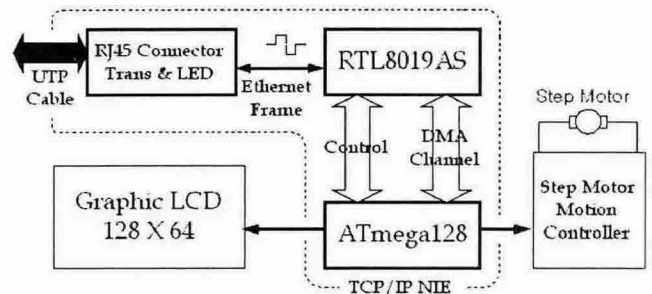


그림 2 마이크로컨트롤러 TCP/IP NIE 구성
Fig. 2 Configuration of the microcontroller TCP/IP NIE

2.2 프레임 및 소프트웨어

이더넷에는 IEEE 802.3 이더넷과 이더넷 II의 두 가지 표준이 있다. 두 표준 프레임은 유사한 구조를 가지며, 현재 두 표준이 함께 사용되고 있다. RTL8019AS는 두 표준 이더넷을 모두 지원한다.

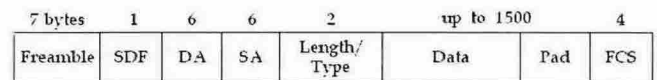


그림 3 이더넷 프레임 구조
Fig. 3 Format of an ethernet frame

그림 3은 이더넷에서 사용되는 프레임의 구조이다. Ethernet II는 SDF 항목을 제외한 8 bytes의 Freamble 로 되어있다. 원격 DMA로 마이크로컨트롤러에 수신되는 항목은 DA(Destination Address), SA(Source Address), Length/Type, 그리고 Data이며, 프레임의 처리 및 응답을 위해 프로그램에서 사용된다. DA와 SA는 각각 수신노드와 송신노드의 물리적인 주소를 나타낸다. Length/Type은 802.3 이더넷 프레임의 데이터 길이를 나타내며, 이더넷 II에서는 입력된 데이터의 프로토콜 정보를 나타낸다. Data 항목은 프로토콜 및 논리주소를 포함한 데이터가 입력되는 부분이며, 최소 46 bytes에서 최대 1500 bytes의 크기를 가진다. Pad 항목은 데이터의 크기가 46 bytes보다 작을 때 크기를 맞추기 위해 추가되는 부분이다. 프레임의 마지막 부분인 FCS(Frame Check Sequence)는 이더넷 프레임의 손상여부를 나타내기 위한 값이며, 보통 CRC값이라고 한다^[1].

마이크로컨트롤러에서 수행되는 프로그램은 수신된 프레임에 대한 처리 및 데이터를 송신하기 위한 프레임의 주소, 프로토콜 정보, 그리고 데이터와 같은 항목들을 생성한다. 원격 DMA(Direct Memory Access)로 마이크로컨트롤러의 내부 메모리에 저

장된 프레임의 항목들은 프로토콜 정보에 따라 각각 해당하는 서브루틴에서 처리한다.

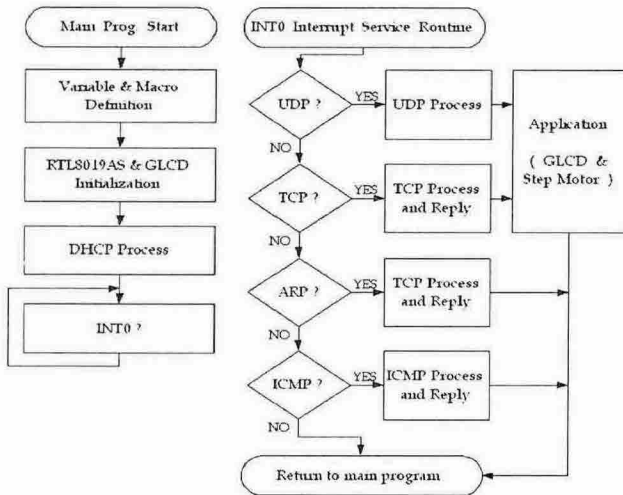


그림 4 흐름 선도
Fig. 4 Flow chart

TCP/IP 통신에서 사용되는 기본적인 프로토콜에는 ARP, ICMP, UDP, TCP가 있다^[1]. 본 논문에서는 기본적인 프로토콜에 대한 처리 및 응답 외에 DHCP(Dynamic Host Connect Protocol)^[2]을 추가하여 유동IP 자동 설정이 가능하도록 프로그래밍한다. 그림 4는 DHCP와 기본적인 TCP/IP 프로토콜의 처리 및 응답을 구현한 프로그램의 전체 흐름 선도이다.

3. TCP/IP 스텝 모터 위치 제어기 구현

먼저 마이크로컨트롤러 TCP/IP NIE를 구현한 다음 TCP/IP 통신이 원활히 이루어지는지를 확인한다. 마이크로컨트롤러 TCP/IP NIE의 전원인가 후, DHCP 서버로부터 할당받은 유동IP는 그래픽 LCD를 통해 확인할 수 있었다. 그리고 할당받은 유동IP로 텔넷을 통해 마이크로컨트롤러 TCP/IP NIE에 접속을 시도한 결과 접속이 이루어짐을 확인할 수 있었다.

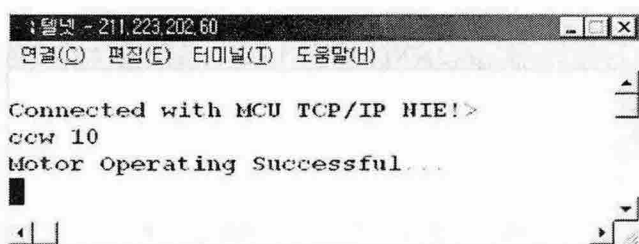


그림 5 텔넷 응용 프로그램 화면
Fig. 5 Picture of a telnet application program.

그림 5는 텔넷을 통해 실제 구현한 마이크로컨트롤러 TCP/IP NIE에 접속한 모습이며, 마이크로컨트롤러 TCP/IP NIE는 “Connected with MCU TCP/IP NIE!>” 라는 배너의 응답으로 접속이 완료되었음을 텔넷 응용 프로그램에게 알린다. 그리고 텔넷을 통해 “ccw 10” 명령어 전송 후, “Motor operating successful...” 이라는 응답으로 스텝 모터가 정상적으로 작동했음을 알린다.

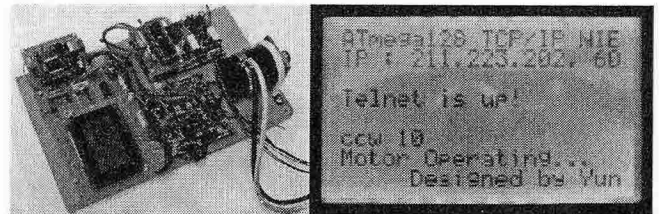


그림 6 구현한 TCP/IP 스텝 모터 위치 제어기
Fig. 6 The TCP/IP step motor motion controller

그림 6은 실제 구현한 마이크로컨트롤러 TCP/IP NIE를 스텝 모터 위치 제어기에 적용하여 TCP/IP 통신으로 스텝 모터의 위치제어를 구현한 모습이다. 텔넷 응용 프로그램으로 PC에서 “ccw 10” 명령을 전송한 후 스텝 모터가 전송한 명령의 회전수만큼 구동되는 것을 스텝 모터와 그래픽 LCD를 통해 확인할 수 있었다. 이것으로 마이크로컨트롤러 TCP/IP NIE를 적용한 스텝 모터의 위치제어가 TCP/IP 통신을 통해 원활히 이루어짐을 검증한다.

4. 결론

본 논문에서는 마이크로컨트롤러를 이용하여 운영체제를 배제한 TCP/IP NIE를 구현함으로써, 하드웨어와 소프트웨어를 간략화 하였다. 그리고 스텝 모터의 위치 제어기에 적용하여, TCP/IP 통신을 통한 스텝 모터의 위치제어를 구현하였다. 본 논문에서 구현한 마이크로컨트롤러 TCP/IP NIE는 개발비용 및 원가를 낮춤으로써, 스텝 모터 외에 다양한 제어 및 관측시스템에 적용되어질 것으로 전망한다.

참고 문헌

- [1] Fred Eady, “Networking and Internetworking with Microcontrollers”, Newnes Pub., 2004.
- [2] Jeremy Bentham, “TCP/IP Lean: Web Servers for Embedded Systems”, 2nd Edition, CMP books, 2002.
- [3] RTL8019AS Data Sheet, REALTEK Semiconductor Corp., 2001.
- [4] ATmega128 Data Sheet, Atmel Corp., 2002.