

일반 전동차량 네트워크의 노드간 MASTER 전환 알고리즘 구현

연준상* · 양오**†

* 청주대학교 전자공학과, **† 청주대학교 반도체공학과

Implementation of Master Changing Algorithm between Nodes in a General Electric Vehicle Network

Jun Sang Yeon* and Oh Yang**†

* Electronics Engineering of Cheongju University

**† Semiconductor Engineering of Cheongju University

ABSTRACT

This paper presents the implementation for the master changing algorithm between nodes in a general electric vehicle. The packet processing method based on the unique network method of an electric vehicle is that the method of processing a communication packet has the priority from the node of a vehicle installed at both ends. An important factor in deciding master or slave in a train is that the request data, the status data, and transmits or control codes of sub-devices are controlled from the node which master becomes. If the request data or the status data is transmitted from the non- master side, it is very important that only one of the devices of both stages be master since the data of the request data may collide with each other. This paper proposes an algorithm to select master or slave depending on which vehicle is started first, which node is master or slave, and whether the vehicle key is operation. Finally experimental results show the stable performance and effectiveness of the proposed algorithm.

Key Words : Master, Slave, Electric vehicle, Changing algorithm, Operation key, FPGA

1. 서 론

기존 전동차들은 대부분 양쪽 차량에서 선두를 지정하여 상/하행선 또는 좌/우측으로 전동차를 운행하도록 되어 있다. 그래서 차량의 운행키(KEY)를 투입한 방향에서만 운전이 가능하며 KEY를 넣은 부분에 방향이 선두가 되어 동작하게 된다[1]. 만약 양쪽 차량 모두 운행키가 꼽혀있다면 각 차량의 기관사에 메시지를 전달하는 모니터를 통해 고장을 지시하고 동작을 멈추게 된다. Fig 1은

전동차량의 운행 키(KEY)와 선두와 후두간의 관계를 나타낸 것이다. 기관사는 차량에 탑승하여 운행 키를 이용하여 배터리 전원을 차량 전체에 투입하고 또한 운행키가 투입된 차량이 중심이 되어 가선을 투입하거나 각종 장치를 점검한다. 하지만 반대편의 차량에 운행 키를 투입시 반대편의 차량이 기능을 대신하게 된다. 따라서 양끝 차량에만 오로지 기관사나 검수요원들이 차량의 상태를 한눈에 볼 수 있도록 각각 모니터가 부착된 장치가 설치되어 있으며 전동차량 양단의 모니터장치는 운행키를 중심으로 선두설정을 확인하고 선두가 되었을 경우 차량의 네트워크의 MASTER 권한으로 전동차량이 Fig 1과 같이 운전하게 된다.

†E-mail: ohyang@cju.ac.kr

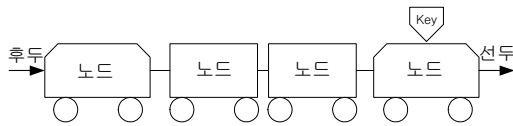


Fig. 1. Leading position when the key is turned to the right node.

즉 요청데이터를 선발적으로 전송하여 각각의 노드에 전달하여 이에해당하는 동작을 하도록 명령어를 전달하거나 각종 상태값을 요청하여 각 차량의 상태나 서비스 장치들의 모니터를 하도록 모든 스케줄관리를 진행하도록 되어있다. 하지만 키가놓인 선두차의 모니터장치가 고장이 발생하였거나 차량용 네트워크의 이상이 발생하였을 경우 차량의 네트워크의 전반적인 동작이 멈추도록 되어있어 이를 해결하고 선두영역권이 후미차로 이동되어 키가놓인 선두차량이 아닌 후미차량이 선두기능을 대신하여 네트워크를 관리할 수 있도록 하는 알고리즘을 제시하도록 한다.

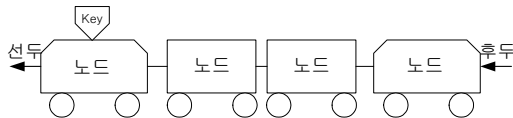


Fig. 2. Leading position when the key is turned to the left node.

또한 이를 차량용 네트워크에서 MASTER와 SLAVE 권한부여라고 부르면 선두와 후두 차량이 KEY투입 위치에 이동하더라도 본 논문에서 제시한 알고리즘을 적용한다면 차량의 상황이나 네트워크상태를 분석하여 네트워크의 MASTER또는 SLAVE를 결정하게 된다[2].

2. 본 론

2.1 차량의 네트워크 구조

대부분의 전동차 차량의 네트워크 구조는 차량의 모양과 특성에 맞게 일자 형식의 네트워크로 구성되어 있다. 일반적인 사무용이나 산업용으로 사용하는 스타형식이나 원형 그리고 분산형 등의 네트워크는 전동차 구성 모양으로 인해 구성하기가 쉽지 않다[3]. 일반형 전동차량과 고속전철인 경우네트워크의 7계층 중 물리 층에서의 기준으로 살펴보면 네트워크를 구성하기 위한 통신라인에 통신 하고자 하는 모든 장치의 노드(node)들이 연결되어 있는 멀티드롭(multi-drop)방식의 네트워크가 있는데 MASTER에서 패킷을 전송하게 되면 모든 노드들이 한번에 패킷을 수신하게 된다[4]. 따라서 수신된 노드들은 OSI

7층 중에서 네트워크 층이나 또는 그 이상 층에서 자신의 노드인 것만 처리하게 된다[5]. 다른 하나의 방법으로 노드 간 통신 케이블이 연결되어 있어 데이터 패킷들을 노드가 직접 패스하는 방법으로 데이터를 수신 받고 다시 전송하는 토큰 방식의 네트워크가 존재한다[6]. 이러한 전동차량의 특이한 구조로 한 쪽에서 전송된 데이터 패킷은 각각의 노드가 모두 직접 수신하든지 아니면 모든 패킷들이 각 노드를 통해서 전달 하도록 되어 있다. 그래서 각 노드 중 MASTER는 한곳에서 이루어져야 하며 양쪽이 MASTER가 되었을 경우 데이터 패킷이 충돌할 수 있다[7].

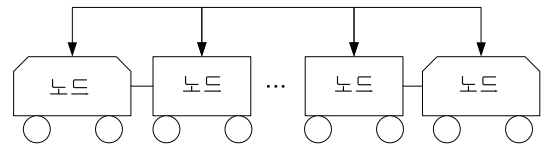


Fig. 3. Network with multi-drop physical layer (WTB, FIP, DIN BUS).

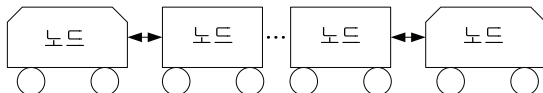


Fig. 4. Network in which the physical layer is configured with token delivery (ALSTOM TORNADO, TOSHIBA SL-NET).

Fig 3과 4는 전동차의 네트워크 시스템의물리 층에서 본 개념이다. 전동차 차량의 기본 네트워크의 기본 개념은 네트워크용으로 사용되는 케이블은 일자로 구성된 네트워크이며 차량에 따른 여러 가지 장비들이 같은 종류의 네트워크 또는 별도의 네트워크 방식으로 연결되어 있는데 종류가 다른 네트워크인 경우 GATEWAY라는 장치를 쌍방간 통신을 중계해주며 차량이 움직이거나 특정한 동작 할 때 양쪽의 차량 중 선두 차량이MASTER가 되어 요청데이터와 상태데이터를 전송하여 여러가지 동작에 필요한 데이터를 주고받도록 되어 있으며 이에 연결된 하부 장치들도 함께 동작을 하게 된다[8]. 하지만 차량 양단 노드 중 MASTER가 되지않은 SLAVE된 노드는 요청데이터를 선행으로 전송하지 않고 MASTER가 요청한 데이터를 수신 받고 이에 합당한 동작을 하거나 상태데이터를 전송하게 된다. 이전의 전동차량들은 가장 중요한 신호들은 전동차에 별도의 케이블을 장착하여 추진을 하거나 제동을 수행하도록 하였지만 최근의 추세는 차량에 설치된 네트워크를 이용하여 서비스 장치들로 정보를 전달할 뿐 아니라 추진, 제동과 같은 중요한 정보도 함께 전송하도록

한다. 그러므로 어느 노드가 선두이고 네트워크가 MASTER 인지 판단하는 것은 상당히 중요한 기능이 되었다.

2.2 MASTER 와 SLAVE 의 전환알고리즘

대부분의 전동차는 차량을 처음 기동 시키면 일반 자동차 키와 같은 기능을 하는 기동용 키를 차량의 데스크에서 키를 투입하도록 되어 있는데 투입과 동시에 차량의 배터리 전원이 차량 내 각 장치들에 공급되고 주요 기능의 장치들이 우선적으로 작동하게 된다. Fig 5는 양 끝 노드에 'A' 또는 'B'키가 놓일 수 있는 위치를 그림으로 나타낸 것이며 이중 하나의 KEY만 놓일 수 있다. 기관사나 기타 담당자들은 하나의 키를 놓고 'A' 또는 'B' 방향으로 차량을 운행하게 된다.

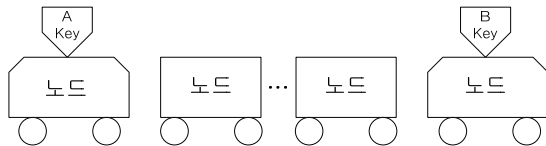


Fig. 5. Vehicle that can put the key of the train.

Fig 6에서는 KEY가 투입 후 양단의 노드 중 어느 장치가 MASTER 또는 SLAVE가 될 것인지를 결정하는 흐름도이다. 여기서는 KEY가 투입되지 않고 전원만 투입되었을

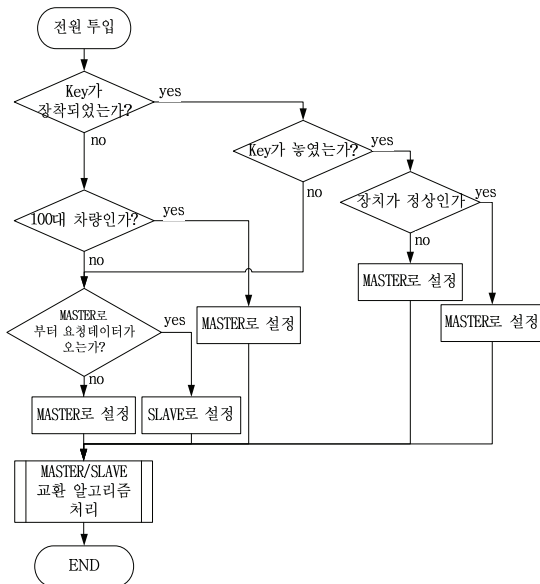


Fig. 6. MASTER and SLAVE changing algorithm when the vehicle is powered on.

경우도 가정하였으며 KEY가 투입되지 않았을 경우 양단의 노드 중에서 특정 노드가 우선적으로 MASTER 권한을 갖도록 하였다. 처음 기동시 KEY가 투입된 차량의 노드가 MASTER로 동작하다가 KEY의 위치가 변경되거나 MASTER 노드의 장치가 문제가 발생하여 본 기능을 발휘할 수 없을 때 즉각적으로 SLAVE 노드가 MASTER의 제어 권한을 요청하거나 강제로 전환하여 정상적인 동작을 계속하도록 한다.

Fig 7은 전동 차량의 운행키를 움직일 때 마다 어느 노드가 MASTER 권한을 가지게 되는지를 알고리즘으로 표현한 흐름도이다.

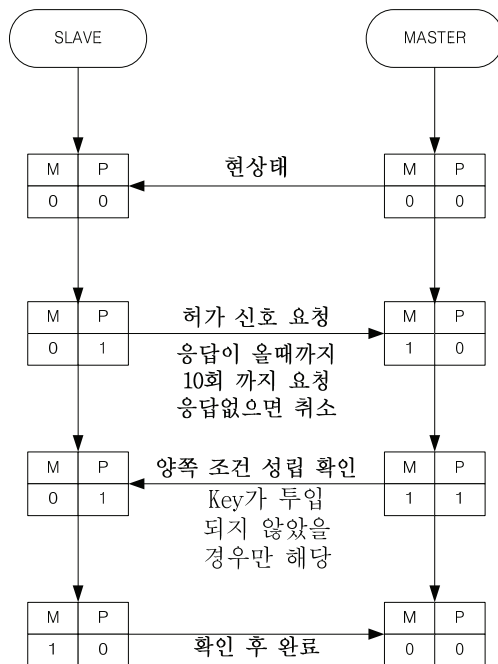


Fig. 7. Flow chart according to MASTER changing.

Fig 7에서 'M'은 마스터 비트인 'P'는 허가 비트를 각각 나타내며 이것을 이용하여 현재 MASTER 권한이 정상적으로 획득이 되는지를 확인한다. 이러한 과정은 전동차의 양끝 노드에 네트워크 시작점이 어느 곳에서 시작될 것인지를 결정한다. 즉 MASTER에 어느 노드가 연결될 것인지를 설정하는 흐름도이다. 차량에 전원이 처음 인가되고 어느 쪽의 KEY가 투입되었는지 또는 어느 차량인지를 확인 후 MASTER / SLAVE가 정해진 상태에서 KEY는 바꾸게 되면 SLAVE는 MASTER에게 'P'비트를 '1'으로 세트 한 후 요청을 하게 되고 MASTER는 현재상태인 'M'과 'P'상태 비트를 SLAVE로 응답한다. SLAVE는 수신 후 다

시 한 번 'P'비트를 1로 세트 후 요청하고 MASTER는 'M'과 'P'비트를 모두 '0'으로 클리어 한 후 다시 한 번 SLAVE로 응답하게 되며 SLAVE 'M'비트는 '1'로 세트 후 'P'비트는 '0'으로 클리어 후 SLAVE로 된 이전 MASTER에게 계속하여 비트를 전달하게 된다.

3. 실험 및 결과분석

본 논문에 제안한 알고리즘의 타당성을 검증하기 위하여 Fig 8과 같이 전동차량의 통신 시스템을 구성하였다. Fig 8을 보면 각각 노드의 전원을 공급하는 Power보드, 아날로그 처리보드, I/O보드, 네트워크보드, CPU보드로 구성되어 있다. CPU보드가 I/O보드를 이용하여 KEY의 투입 여부를 확인 하고 네트워크 보드를 통해 제시된 알고리즘을 구현하도록 하였다.

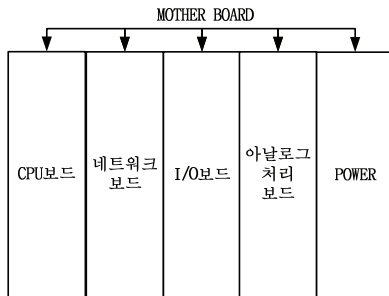


Fig. 8. Configuration of each node.

Fig 9는 실험에 사용된 네트워크 보드의 내부 구성도이다. 기존에 사용된 대부분의 네트워크 보드들에 사용되었던 통신용 칩들은 물리 층이나 데이터 링크 층을 구현하기 위해 기존에 상용화된 제품들을 주로 사용했다[9]. 특히 상용화된 제품 중에는 버퍼 기능을 향상시켜 보다 원활하게 네트워크를 구현하고 CPU가 최소한의 부하로 동작할 수 있도록 하는 개량된 제품들도 있다[10]. 하지만 이러한 사용 제품들의 단점은 철도용이나 특정 산업용에 사용하기에 유연하지 못하여 이를 극복하고 철도용 네트워크에서 MASTER전환을 구현하기 위해 철도차량에 맞는 멀티 다중 칩을VHDL로 직접 개발한 다중 멀티용 칩을 사용하였다[11].

또한 좀더 빠른 로직으로 향상된 시스템을 구현하기 위해 데이터링크 및 네트워크 층까지 설계된 칩에서 처리하도록 하여 CPU가 동작하는데 유연하게 처리하도록 하였다[12]. 이러한 요구조건에 적합하도록 본 논문에서 제안된 알고리즘을 구현하기 위해 XILINX사의 FPGA를 사용하였으며 제안된 알고리즘을 구현하였다[13].

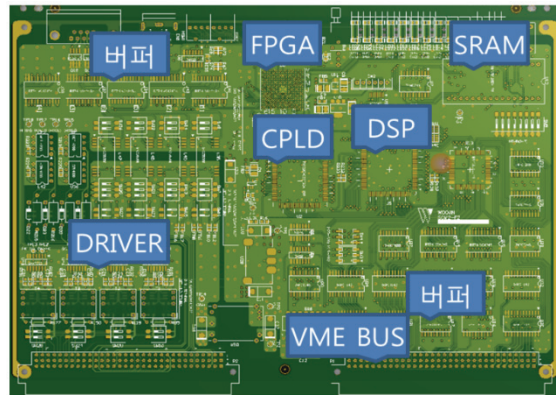


Fig. 9. Block diagram for networking board.

```

tg_IsMaster;tg_IsMaster
value = 0 = 0x0
->
->
-> tg_SetMaster;tg_SetMaster
Master Request Send...
value = 23 = 0x17
-> Permit Please..
Permit Please..
Permitted => Master...

->
-> tg_IsMaster;tg_IsMaster
value = 128 = 0x80

```

Fig. 10. MASTER switch message operated by the console from the node in the SLAVE state.

또한 콘솔의 명령으로 MASTER 또는 SLAVE가 정상적으로 변환 되는지 확인 하기 위해 Fig 10과 같이 RS232C 포트의 콘솔을 이용하여 SLAVE에서 MASTER로 변경되는지 확인하는 메시지를 출력 하여 정상 동작 유무를 확인 한 내용이다.

우선 현재 MASTER가 되어 있는지 확인하는 명령어인 "tg_IsMaster"라는 명령을 입력 후 'M'비트에 '0'으로 클리어 되어 있는 것을 확인 후 'tg_SetMaster'라는 명령어를 입력 후 'M'과 'P'비트를 이용하여 MASTER요청 및 변환이 되고 해당 노드가 MASTER되는 것을 확인하였으며, 다시 "tg_IsMaster"명령어를 입력하여 MASTER가 되어 있는지를 확인하였다.

Fig 11은 네트워크 통신파형을 오실로스코프로 측정한 것이며 통신주기는 100ms마다 통신을 하며 각 주기마다 한 그룹을 확대하여 보면 첫번째 파형은 MASTER 노드에서 데이터요청을 시작으로 각각의 단말 노드에서 마지막 SLAVE노드까지 응답하는 것을 확인할 수 있다. 또한 각각의 제어비트 및 데이터비트는 한개의 패킷에 모두 로

드 하여 전송되도록 되어 있어 주기적으로 100ms마다 송수신이 이루어지는 동안 MASTER의 변환 요청을 하는 패킷과 이를 수용하는 패킷 등은 모두 통신 주기에 동기가 맞춰 이루어지도록 되어 있다. 전동 차량의 운전 KEY가 변경되고 MASTER변환은 Fig. 11에서 보이는 흐름도는 100ms 마다 이루어진다. Fig. 12는 본 논문에서 제안된 알고리즘이 정상적으로 동작하는지 확인하기 위해 실제 차량에 탑재되고 있는 TGIS라는 장치에서 알고리즘을 프로그램으로 구현하여 시험하였다. 이 장치는 전동차 차량과 같은 조건으로 작동하기 위해 전용 시뮬레이터에 연결되었으며 본 시뮬레이터에서는 전동차 차량에서 입/출력되는 수많은 신호들과 차량의 진행방향을 지정하는 KEY 투입도 장치에 직접 넣을 수 있도록 구현된 장치이다. 전원이 처음 인가 되고 KEY가 투입된 방향과 차량의 번호의 조건에 따라 MASTER로 설정되며 각 노드에 설치된 화면의 메시지를 통해 확인하였다.

Fig. 12는 전동차 차량에 설치된 TGIS라는 모니터 정보 장치의 화면이며 전동차 양 단에 설치되는 노드장치이다. 아울러 Fig. 13은(A)와 (B)장치가 전동차 양단에 장착되어 되어 마스터(Master)와 슬레이브(Slave)간 원활한 전환 동작을 시험하고자 설치된 노드 장치들의 그림이다.

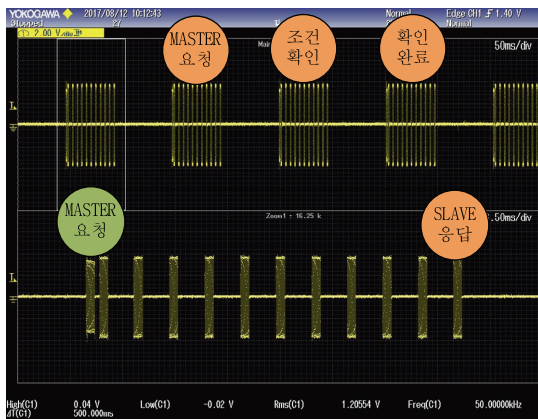


Fig. 11. Network communication waveform by MASTER / SLAVE changing.

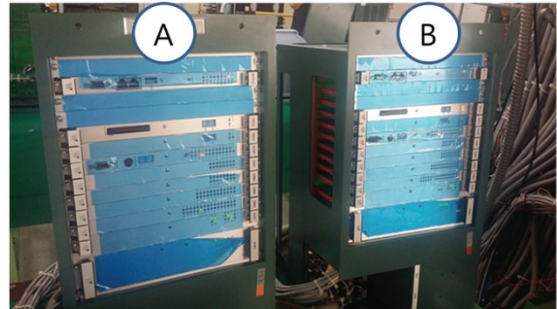


Fig. 13. Monitor device mounted on both ends of a general train.

제안된 알고리즘을 설치하고 각 노드들에 전원을 인가하여 기동 시킨 후, KEY가 투입되지 않은 화면에 출력된 메시지는 SLAVE로 되어 있으며, 반면에 반대 방향의 KEY가 투입된 노드 장치에는 알고리즘에 따라 KEY가 투입된 노드가 MASTER권한을 가져와 화면에 메시지를 출력하는 것을 확인 하였다. 또한 KEY의 위치가 변경 될 때 마다 투입된 방향의 노드가 MASTER권한을 가져오는 것을 확인하였으며 KEY가 투입되어 MASTER권한을 가져 왔던 노드가 임의의 고장을 발생하여 KEY가 투입되지 않는 반대방향의 노드가 특정시간 후 MASTER기능을 대신하는 동작을 확인하였다.

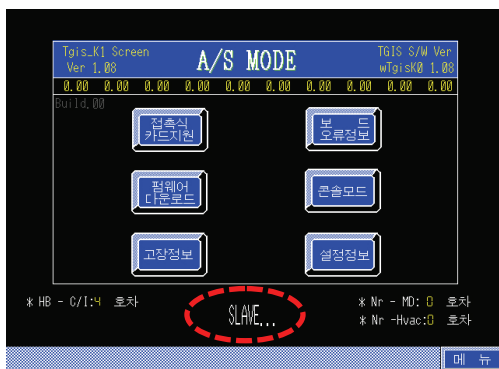


Fig. 12. Screen message with SLAVE.

4. 결 론

일반형 전동차 차량에 사용하는 모니터장치인 TGIS에 제시한 알고리즘을 적용하여 실험한 결과 KEY를 투입한 노드에 따라 MASTER가 전환되는 것을 확인하였으며 SLAVE노드에서 KEY가 투입되면 제시된 알고리즘에 맞게 동작하여 MASTER의 권한을 가져오며 MASTER는 다시 SLAVE로 변환되었다. 그 결과 양단 두 개의 노드가 동시에 MASTER되거나 SLAVE가 되지 않는 것을 확인하였다. 또한 차량을 사용하지 않는 대기 중인 상태를 가정하여 양쪽 노드에 동시에 KEY가 놓지 않았을 때도 어느 한 쪽은 항상 MASTER로 놓이게 되어 차량에 필요한 정보들은 요청에 의해 계속하여 응답하고 수집하게 되었다. 또한 논문에서 제안한 알고리즘을 사용하여 KEY가 투입된 MASTER가 된 노드에서 비정상적인 동작을 하거나 고장

이 발생하여 시스템이 멈추게 되면 반대편 KEY가 투입되지 않은 SLAVE 노드가 일정 시간 경과 후 강제로 MASTER가 되어 문제가 있는 상대방 MASTER를 대신하여 정상적으로 동작 하는 것을 확인하였다. 따라서 차량마다 모니터 장치는 하나 이지만 KEY가 투입된 차량 이외에 KEY투입되지 않은 상대방 차량의 모니터 장치도 MASTER기능으로 충분히 동작 할 수 있어 2중계의 효과 및 고장 발생시에도 효과적으로 대처할 수 있는 장점을 실험을 통해 확인하였다. 아울러 양단의 노드가 동시에 MASTER권한을 가져가지 못하므로 하나의 네트워크에 동시에 필요한 데이터를 동시에 요청하거나 기타 필요한 데이터 패킷을 양쪽 노드에서 동시에 전송되는 경우가 발생되지 않고 안정적으로 신뢰성 있는 통신이 이루어지는 것을 실험을 통해 확인하였다.

참고문헌

1. Woojin, "TGIS report for repair manual of gwacheon line", pp. 1.3-1.20, 2005.
2. Dong-kyu lee, "Implementation of network management protocol for railway cars and master fault tolerance algorithm", Master thesis, pp. 8-9,45-49, Inha university, 2000.
3. Jae-chul Lee, "A Study on Implementation of PC based Distributed Industrial Network", Doctoral thesis, pp.16-33, Mokwon University, 2009.
4. International Electro technical Commission (IEC), "Chapter 4(Train Communication Network for WTB (Wire Train Bus))", IEC 61375, first edition, pp. 298-327,2003.4.
5. J.DE AZEVEDO, "The World FIP Protocol Standard and Specifications", World FIP EN50170, pp. 5-7, 1998.10.
6. Jonathan, "Token Ring Solutions", IBM Redbooks, pp. 9-18, 2000.
7. Rhonda Alexis Dirvin and Arthur R. Miller, Motorola, Inc. "MC68824 Token Bus Controller", IEEE 802.4, pp. 15-25, 1986.
8. Jae-sun Han, "A Study on Efficient Gateway Message Transmission Technique between CAN (Controller Area Network) Communication Networks", "CAN (Controller Area Network)", Yonsei University, Master thesis, pp. 719-722, 2007.
9. ZILOG, "Z80C30/Z85C30 CMOS SCC Serial Communications Controller Product Specification", pp1-14, pp. 27-39, 2002.
10. SIEMENS, "SAF/SAB82532 Datasheet Enhanced Serial Communication Controller for HDLC", pp. 88-108, 2007.
11. Sung-Gyu Ok, Oh Yang, "IC design for Multi-Channel Asynchronous Communications using a FPGA", Journal of the Institute of Electronics Engineers of Korea, pp. 28-37, 2010. 1.
12. Sung-Gyu Ok, Oh Yang, "IC design for Multi-Channel Synchronous Communications using a FPGA", Journal of the Semiconductor & Display Technology, pp. 1-6, 2011. 9.
13. XILINX, "Spartan-3 FPGA Family: Complete Data Sheet", pp. 47-55, pp. 107-108, pp. 128-270, 2013.

접수일: 2017년 9월 7일, 심사일: 2017년 9월 18일,
 게재확정일: 2017년 9월 18일