

분산제어시스템을 위한 타이머 제어형 통신망의 주기 및 실시간 비주기 데이터 전송 방식

Transmission Method of Periodic and Aperiodic Real-Time Data on a Timer-Controlled Network for Distributed Control Systems

문 홍 주, 박 홍 성
(Hong-ju Moon and Hong Seong Park)

Abstract : In communication networks used in safety-critical systems such as control systems in nuclear power plants, there exist three types of data traffic: urgent or asynchronous hard real-time data, hard real-time periodic data, and soft real-time periodic data. It is necessary to allocate a suitable bandwidth to each data traffic in order to meet their real-time constraints. This paper proposes a method to meet the real-time constraints for the three types of data traffic simultaneously under a timer-controlled token bus protocol or the IEEE 802.4 token bus protocol, and verifies the validity of the presented method by an example. This paper derives the proper region of the high priority token hold time and the target token rotation time for each station, within which the real-time constraints for the three types of data traffic are met. Since the scheduling of the data traffic may reduce the possibility of the abrupt increase of the network load, this paper proposes a brief heuristic method to make a scheduling table to satisfy their real-time constraints.

Keywords : urgent data, periodic data, token bus, scheduling

I. 서론

1960년대의 프로그램형제어기 및 DDC (Direct Digital Control)의 등장을 거쳐, 1970년에 최초로 Honeywell에 의해 분산제어시스템이 등장하게 되었으며[1], 그후 발전소 및 생산시스템을 비롯한 많은 분야에서 분산제어시스템의 개념이 적용되기 시작했다.

분산제어시스템은 일반적으로 운전기능, 엔지니어링기능, 공정제어기능, 입출력기능, 자료취득 및 저장기능, 기술지원기능 등을 제공한다. 분산제어시스템은 제품 및 특성에 따라 그 구성을 여러 가지 형태로 구분하여 생각할 수 있으며, 대상 시스템에 따라 여러 가지 형태로 구성하여 사용할 수 있다. 그림 1에 분산제어시스템의 구성 예가 나타나 있다. 각 구성요소들은 통신망으로 제어 및 감시 정보를 주고 받으면서 연계동작을 한다.

분산제어시스템의 구성요소중 통신망의 동작 및 특성은 전체 분산제어시스템의 동작 및 특성을 결정짓는 매우 중요한 부분으로 전체시스템이 요구하는 각종 특성을 만족시켜야 한다. 통신망은 분산제어시스템의 계층구조에 따라 복수개의 통신망들이 사용되면, 각각에 연계되는 분산제어시스템의 구성요소의 기능과 동작특성을 만족시키기 위한 특성을 갖추게 된다. 그림 1과 같은 구성에서는 공정제어기들이 연계되는 제어통신망이 가장 중추적인 역할을 담당하게 되며, 대부분의 상용 분산제어시스템 제품에서는 자체적으로 개발한 폐쇄적인 특성의 제어시스템을 사용하는 경우가 대부분이다. 본 논문에서는 그림 1과 같은 구성을 갖는 분산제어시스템에서의 제어통신망에 대한 데이터

전송방식을 살펴본다.

그림 1과 같은 구성을 갖는 분산제어시스템에서의 제어통신망에는 직접적인 제어 및 감시동작을 수행하는 공정제어기가 연계되어 동작하고, 동시에 감시 및 운전조작 등을 수행하는 운전원스테이션이 연계될 수 있으므로, 주기적인 데이터와 비주기적인 데이터를 포함하는 다양한 특성을 갖는 데이터들에 대한 전송 서비스가 제공되어야 한다. 주기적인 데이터의 전송기능과 비주기적인 데이터의 전송기능은 상반된 특성을 갖는 경우가 많으므로 두가지 데이터를 동시에 효과적으로 서비스하는데 어려움이 많으며, 제어통신망으로 사용되는 통신 프로토콜들은 어느 한쪽에 치우친 특성을 갖는 경우가 많다. 최근까지도 이에 관한 많은 연구가 진행되고 있으며[2]-[11], 다양한 현장 데이터 특성을 만족시킬 수 있는 연구개발이 요구되고 있다. FIP[12], FF[13]와 같은 필드버스는 주기 데이터 서비스를 위주로 하고 있으며, IEEE 802.4[14], Profibus-FMS[15]와 같은 타이머 제어 토큰버스형 네트워크는 비주기 데이터 서비스를 위주로 하고 있다. 타이머 제어 토큰버스를 사용하여 주기 데이터 서비스를 제공하기 위한 방법을 연구하고 있지만[2][3][11], 이러한 방법은 비주기 데이터를 주기 데이터로 만들어 주기 데이터처럼 가정하여 연구하고 있다.

본 논문에서는 제어통신망에 널리 사용되는 타이머제어형 토큰 버스 프로토콜을 사용하여 비주기데이터와 주기 데이터를 동시에 효과적으로 처리할 수 있는 방법을 제안한다. 이를 위해, 먼저 2장에서 분산제어시스템에서의 통신데이터 특성을 살펴보고, 3장에서 타이머제어형 토큰버스 프로토콜의 특징과 전송방식에 대한 가정을 기술한다. 4장에서 타이머제어형 토큰버스 프로토콜의 특징을 사용

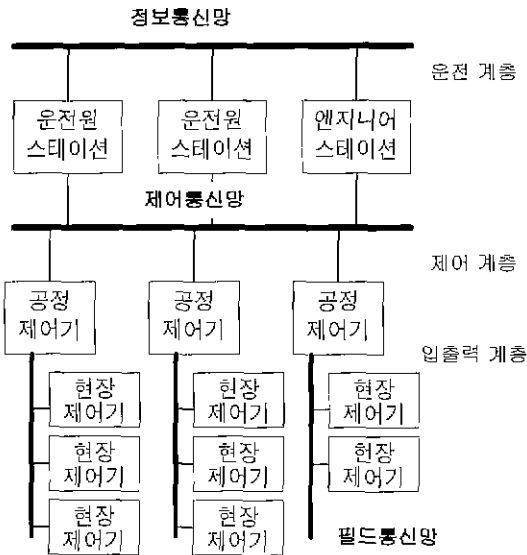


그림 1. 분산제어시스템의 구조 예.
Fig. 1. Example of the architecture of a distributed control system.

하여 분산제어시스템에서의 통신데이터 특성을 만족시키기 위한 방법을 제안한다 5장에서 제안된 방법을 사용한 제어통신망의 구축 및 시험사례를 살펴보고 6장에서 결론을 맺는다.

II. 분산제어시스템에서의 통신데이터

본 장에서는 원자력 발전소의 핵심적인 제어 부분인 발전소 제어 계통에 적용되는 분산제어시스템에서의 통신데이터에 대한 특성을 살펴보고 이를 분류한다. 발전소 제어 계통은 주급수 제어 계통 (FWCS; Feedwater Control System), 증기 우회 제어 계통 (SBCS; Steam Bypass Control System), 원자로 제어 계통 (RRS; Reactor Regulating System), 원자로 출력 급감발 계통 (RPCS; Reactor Power Cutback System), 가압기 압력 제어 계통 (PPCS; Pressurizer Pressure Control System), 가압기 수위 제어 계통 (PLCS; Pressurizer Level Control System), 제어봉 제어 계통 (CEDMCS; Control Element Drive Mechanism Control System)으로 구성된다[16]-[18]. 각각의 계통의 동작은 다음과 같다.

주급수제어 계통은 급수 유량을 조절하여 증기 발생기 수위를 적절하게 유지시킨다. 증기 우회제어 계통은 부하 감발시에도 에너지 평형상태를 유지하는 역할을 수행하며, 원자로 제어계통은 빠르고 효과적으로 원자로계통을 제어할 수 있도록 설계된 자동제어 계통이다. 부하변동이 큰 과도현상이나 계통외란에 대한 기능은 원자로 출력 급감발계통이 수행한다. 가압기 압력제어계통 및 가압기수위제어계통은 가압기 내부의 온도, 압력 및 수위를 제어한다. 제어봉 구동장치 제어계통은 제어봉 집합체에 붙어있는 제어봉 구동장치를 제어함으로써 노심에서의 반응도에 대한 주제어실 원거리 운전이 가능하게 한다.

발전소제어계통의 경우 표 1과 같이 다양한 신호가 존

표 1. 발전소제어계통의 신호.

Table 1. Signals of the control system in a power plant.

Index	종류	Description	입력모듈	출력모듈
SFID-1,2	D	증기유량차가를 때 발생	지시계	FWCS-1,2
LCD-1,2	D	증기발생기 수위차가 클 때 발생	지시계	FWCS-1,2
RPO-1,2	D	원자로 출력의 차이가 클 때 발생	지시계	FWCS-1,2
TFF-1,2	A	총 공급수량 신호	데이터취득 시스템	FWCS-1,2
IDFF-1,2	A	공급수량에 대한 내부신호	데이터취득 시스템	FWCS-1,2
ILCI-1,2	A	증기발생기 수위의 내부신호	데이터취득 시스템	FWCS-1,2
RP1-1,2	A	원자로출력	FWCS-1,2	RRS
RCAT-1,2	A	원자로냉각기 평균온도	FWCS-1,2	RRS
RT1-1,2	D	원자로정지 저전압신호	FWCS-1,2	CEDMCS
RCATNL-1,2	A	원자로냉각기 평균온도 저부하 설정치	FWCS-1,2	운전원 스테이션
IRF-1,2	A	재충전 유량치 초기값	FWCS-1,2	운전원 스테이션
...

제한다. 표 1에서 D는 디지털 데이터, A는 아날로그 데이터를 의미한다. 본 논문에서는 이러한 데이터들을 발생특성 및 통신요건에 따라 크게 4가지로 구분하여 논의한다.

1. 긴급 데이터 (UD; Urgent Data)

긴급 데이터는 비주기적으로 긴급한 내용을 담고 발생하는 통신 데이터를 지칭한다. 이 통신데이터는 통신장애가 발생하지 않을 경우, 일정한 전송 시간 및 수신확인을 포함하는 응답시간이 보장되어야 한다. 경보감시기 입력, 모드선택 입력 등을 긴급데이터로 설정할 수 있다.

2. 경성실시간 주기데이터 (HRPD; Hard Real-time Periodic Data)

주기적으로 일정한 형식의 내용을 보내는 데이터중, 통신장애가 발생하지 않을 경우, 주기성을 엄격히 보장할 수 있어야 하는 데이터를 지칭한다. 여기서의 주기성은 정해진 주기내에 한번이상 서비스가 처리되는 것을 의미한다. 수동모드에서의 제어입력, 제어기 사이에 공유하는 제어데이터, 제어기간에 공유하는 상태데이터 등을 경성실시간 주기 데이터로 설정할 수 있다.

3. 연성실시간 주기데이터 (SRPD; Soft Real-time Periodic Data)

주기적으로 일정한 형식의 내용을 보내는 데이터중, 정해진 주기내에 한 번씩 전송서비스가 처리되도록 하되 약간 정도의 시간지연을 허용하는 데이터를 지칭한다. 긴급

서비스의 영향이나 일시적인 통신장애가 발생하지 않는 경우는 주기성을 지키도록 한다. 발전소데이터취득장치, 지시계 등으로 가는 감시데이터들이 연성실시간 데이터로 설정될 수 있다.

4 비실시간/고급통신 데이터 (NRD/HQSD; Non Real-time Data/High Quality Service Data)

짧은 변수값들에 대한 신뢰성 있는 전달 및 획득, 각종 긴 데이터의 다운로드 및 업로드, 프로그램의 원격제어와 같은 고급통신 서비스를 제공하기 위해 발생하는 통신요구에 대한 데이터를 지칭한다. 이 데이터는 주로 시스템의 구성변경과 같은 초기화시에 발생되며, 시스템의 운전중에는 거의 발생하지 않는다.

원자력발전소 제어계통을 중심으로 조사한 데이터를 바탕으로, 분산제어시스템의 제어통신망에서 발생할 수 있는 데이터를 분류한 결과, 표 2와 같이 조사되었다. 고급서비스 데이터는 초기화 방식에 따라 다양하게 변화할 수 있고, 운전중에는 거의 발생하지 않으므로 조사대상에서 제외하였다. 즉, 이러한 4개의 데이터 종류중 본 논문에서는 실제 운전 중에 발생하는 긴급 데이터, 경성실시간 주기 데이터, 연성 실시간 주기 데이터만을 다룬다.

표 2. 원자력발전소 제어계통에서의 데이터 분류.
Table 2. Classification of data in the control system of a nuclear power plant.

종류	데이터 갯수		
	디지털 (1 bit)	디지털 (2 bits)	아날로그
긴급데이터 (UD)	94	22	27
경성실시간 주기데이터 (HRPD)	55	0	18
연성실시간 주기데이터 (SRPD)	121	0	28

조사에서 나타난 바와 같이 적지않은 숫자의 긴급데이터가 존재하였다. 비주기적으로 발생하는 데이터는 충분한 통신대역을 확보하고 있거나, 최대허용 응답지연이 그다지 짧지 않은 경우에는 주기 데이터처럼 처리하는 것이 일반적이다. 그러나, 짧은 최대허용 응답지연 시간을 갖는 긴급데이터와 같은 여러개의 비주기 데이터들을 효과적으로 처리하기 위해서는 비주기성 데이터의 특성을 살리면서 통신대역을 활용하는 것이 바람직하다.

III. 타이머제어형 토큰버스 프로토콜의 특징과 전송방식에 대한 가정

실시간 통신 성능을 보장하기 위해 분산제어시스템의 제어통신망에서 IEEE 802.4[14]와 같이 우선순위를 가지는 타이머 제어형 토큰버스 프로토콜이 널리 사용된다. 본 논문에서는 IEEE 802.4 규약을 중심으로 논의한다.

타이머제어형 토큰버스 프로토콜에서는 전송 데이터의 중요도에 따라 우선순위가 부여되며, 각각의 우선순위를 갖는 데이터들은 별도로 처리된다. 토큰이 수신된 토큰버

스상의 특정 스테이션에서는 우선순위의 순서대로 각 우선순위마다 미리 지정된 시간까지만 매체를 점유하며 데이터를 전송할 수 있는 권리를 부여받게 된다.

스테이션들은 각 우선순위마다 목적 토큰 회전 시간 (TTRT; Target Token Rotation Time)과 같은 값을 미리 지정하고, 이전에 토큰을 받아서 다른 스테이션으로 전달한 시점부터 토큰을 보유하고 있는 현 시점까지의 시간이 이러한 목적 토큰 회전 시간보다 작은 경우에만 새로운 데이터의 전송을 시작할 수 있도록 한다. 대개의 경우 이전에 토큰을 받은 시점부터 이번에 토큰을 받은 시점까지의 시간으로서의 토큰 회전 시간은 목적 토큰 회전 시간보다 작게된다. 그러나, 마지막에 전송을 시작한 데이터의 전송 시간이 길어 목적 토큰 회전 시간을 초과하는 경우에도 전송을 시작한 데이터는 전송을 완료할 때까지 해당 데이터의 전송을 계속한다.

IEEE 802.4 표준 규약의 경우 4개의 우선순위를 두며, 최상위 우선순위의 전송에는 목적 토큰 회전시간과는 약간 다른 타이머 방식을 사용한다. 다른 스테이션에서 매체를 사용한 시간에 따라 현 스테이션에서 매체를 사용할 수 있는 시간이 달라지는, 목적 토큰 회전 시간에 의한 타이머 제어 방식과 달리, 토큰 수신시 현 스테이션에서 데이터를 전송할 수 있는 최대 시간을 지정하는 방식이다. 최상위 우선순위 토큰 보유 시간 (HPTHT; High Priority Token Hold Time)을 미리 설정하고 한 스테이션에서 토큰을 받은 시점부터 최상위 토큰 보유 시간이 지나기 전까지 최상위 우선순위 데이터를 전송할 수 있다.

본 논문에서는 엄격하게 실시간성을 지켜야하는 긴급데이터와 경성실시간 주기 데이터는 최상위 우선순위인 6을 할당하고 연성 실시간 주기 데이터는 우선순위 4를 할당한다. 이 논문의 일반적인 가정은 다음과 같다. 전송에 대한 응답은 별도로 고려하지 않는다. 긴급데이터와 경성실시간 주기 데이터, 연성실시간 주기 데이터 이외에 다른 데이터는 발생하지 않는다고 한다. 또한, 이 논문에서는 토큰 버스 통신망의 유지보수를 위한 추가시간은 고려하지 않는다

IEEE 802.4에서는 각 스테이션에서의 긴급데이터 및 경성실시간 주기 데이터의 전송과 연성실시간 주기 데이터의 전송은 별도의 큐에서 관리되며, 그림 2와 같은 형태로 서비스를 받게 된다.

한 스테이션에서 토큰을 받은 시점에서 토큰을 다음 스테이션에 넘겨줄 때까지의 소요 시간은 그림 3과 같이 구성된다. 각 데이터전송 이전에 데이터를 큐에서 꺼내어 데이터를 전송하기 시작할 때까지 얼마정도의 시간이 소요되며, 이러한 시간을 T_p 로 명명한다. 다음 스테이션이 토큰을 받는때까지 걸리는 시간을 토큰전달 오버헤드 시간이라 하며 T_o 로 명명한다. 그림 3에서 긴급데이터 전송 이후에 경성실시간 주기 데이터가 전송되는 것처럼 나타났으나 하나의 큐를 사용하는 경우 긴급데이터와 경성실시간 주기 데이터는 섞여서 서비스된다. 그러나, 연성실시간 주기 데이터는 긴급데이터 및 경성실시간 주기 데이터가 서비스된 이후에 처리된다.

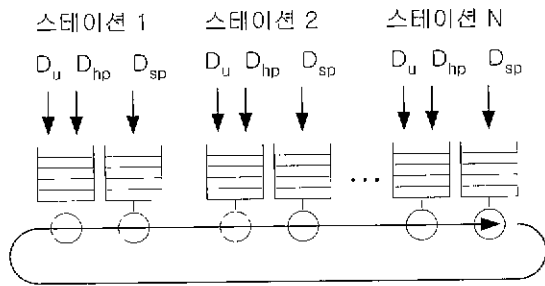


그림 2. 우선순위 있는 토큰 버스에서의 전송.
Fig. 2. Transmission of data in a token bus with priorities.

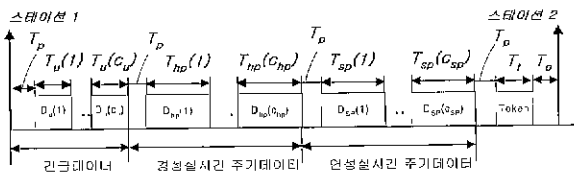


그림 3. 토큰 보유 시간의 구성.
Fig. 3. Bandwidth of a token holding time.

긴급데이터의 프레임 형태는 여러 가지 형태로 정의할 수 있으나 [18]에서의 같이 다음의 구성을 가정한다

프레임 오버헤드	ID	발생 시간	사건 ID	사건 데이터	상태
24	2	2	1	1	1

데이터의 ID는 6만큼 이상의 데이터 처리를 위해 2 byte로 구성하고, 상태를 나타내기 위하여 1 byte를 할당한다. 긴급데이터는 비주기적인 특성을 가지고 사건의 형태로 발생하므로, 발생시간, 사건 ID, 사건데이터에 해당되는 내용을 갖는다.

IEEE 802.4의 경우 프레임 오버헤드에는 프리앰블 (preamble) 10 byte, SD (Start Delimiter) 1 byte, Function code 1 byte, 목적지 주소 2 byte, 전송지 주소 2 byte, 목적지 SAP (Service Access Point) 1 byte, 전송지 SAP 1 byte, 제어 코드 1 byte, FCS (Frame Check Sequence) 4 byte, ED (End Delimiter) 1 byte가 포함된다.

경성실시간 주기데이터 및 연성실시간 주기데이터의 프레임 형태는 다음과 같이 가정한다. [18] 긴급데이터의 경우와 마찬가지로 데이터의 ID와 상태를 갖으며, 데이터는 아날로그 데이터값을 처리하기 위하여 2 byte로 설정한다.

프레임 오버헤드	데이터의 갯수	첫 번째 데이터			두 번째 데이터			...
		ID	데이터	상태	ID	데이터	상태	
24	1	2	2	1	2	2	1	...

한 데이터 프레임에 대한 오버헤드는 긴급데이터의 경우 전체길이의 75%이상이며 매 데이터 전송에 필요한 처

리 지연 시간을 고려하면 이보다 큰 부분을 차지하게 된다 따라서, 여러 데이터를 하나의 데이터 프레임으로 묶어서 전송하면 유효전송률을 높이게 된다.

IV. 비주기데이터 및 주기데이터의 전송

이 장에서는 우선순위가 있는 타임제어형 토큰버스 프로토콜을 사용하여 긴급데이터, 경성실시간 주기 데이터, 연성실시간 주기 데이터가 혼합된 데이터 전송 요구를 효과적으로 처리하기 위한 방법을 제안한다.

비주기적인 긴급데이터와 주기데이터 모두 전송요구의 발생에 대한 제한이 있어야 실시간성을 만족시키는 것이 가능하기 때문에, 본 논문에서는 전송요구의 발생에 대해 다음과 같은 가정 및 제한조건을 정한다.

- 어떤 스테이션 S_i 에서 긴급데이터는 일정한 시간 T_{DU} 동안에 발생할 수 있는 긴급데이터의 요구가 A_i 로 제한된다.
- 어떤 스테이션 S_i 에서 경성실시간 주기데이터는 b_i^p 개가 있으며 각각은 주기 $b_i^p(j)$ ($j=1, \dots, b_i^p$)를 갖는다. 주기 p 를 갖는다는 것은 매 p 시간마다 데이터가 발생한다는 의미로 정의한다.
- 어떤 스테이션 S_i 에서 연성실시간 주기데이터는 b_i^s 개가 있으며 각각은 주기 $b_i^s(j)$ ($j=1, \dots, b_i^s$)를 갖는다. 주기성은 경성실시간 주기데이터와 같은 의미로 정의된다.

한 스테이션 S_i 에서 임의의 기간 T 동안에 발생할 수 있는 긴급데이터의 최대량은 $A_i \lceil T/T_{DU} \rceil$ 로 계산되며, 임의의 기간 T 동안에 발생할 수 있는 경성 실시간 주기 데이터 및 연성 실시간 주기 데이터의 최대량은 각각 $\sum_{j=1}^{b_i^p} \lceil T/b_i^p(j) \rceil$, $\sum_{j=1}^{b_i^s} \lceil T/b_i^s(j) \rceil$ 이다. 여기서 $\lceil x \rceil$ 은 $x+1$ 보다 크지 않은 최대 정수를 말한다.

본 논문에서 사용되는 실시간성 조건은 다음과 같다

- 긴급데이터는 모두 T_{DU} 의 최대허용 시간지연 혹은 데드라인을 갖는다. 즉, 임의의 긴급데이터 D_n 에 대하여 다음 조건을 만족하여야 한다.

$$d(D_n) < T_{DU}$$

$d(D)$ 는 임의의 데이터 D 에 대한 서비스 시간을 말한다. 본 논문에서 서비스 시간은 통신의 요구가 발생한 시점부터 데이터가 전송완료되는 시점까지를 의미한다.

- p 의 주기를 갖는 경성실시간 주기데이터 D_{hp} 는 주기 p 내에 서비스되어야 한다. 즉,

$$d(D_{hp}) < p$$

를 만족하여야 한다.

- p 의 주기를 갖는 연성실시간 주기데이터 D_{sp} 는 확률 α (α 는 1에 가깝다)를 갖고 주기 p 내에 서비스되어야 하며, 모든 데이터는 주기 $2p$ 내에 서비스되어야 한다

SRPD의 경우 2주기 내에 전송되는 것이 허용된다. 반드시 한 주기내에 전송되는 조건 대신 이 조건을 사용하는 경우 통신망의 효율성을 높일 수 있으며, 실시간성이 보장되는 통신량의 범위를 넓힐 수 있다. 그러나, 통신량이 많

아짐에 따라 통신망 효율성의 증가는 매우 작으며, 실시간성이 보장되는 통신량의 범위 또한 크게 증가하지는 않으므로, 본 논문에서는 한 주기내에 전송이 완료되는 경우만을 고려한다.

어떤 스테이션 S_i 에서의 최상위 우선순위 토큰 보유시간을 $HPTHT_i$ 라고 하고, 긴급 데이터 프레임의 최대 길이를 A_u , 경성 실시간 주기 데이터 및 연성 실시간 주기 데이터의 최대 길이를 A_b 라고 하자. 경성실시간 주기 데이터 2개 이상이 하나의 프레임으로 구성된다고 하면 $A_b > A_u$ 의 관계를 갖는다.

IEEE 802.4 표준규약은 타이머 제한값이 종료되는 바로 직전 시점에도 새로운 전송의 시작이 가능하므로 최대 토큰 보유시간에 최대 전송 데이터 길이 만큼의 추가적인 토큰 보유 시간을 갖을 수 있다. 따라서, 그림 3에서 나타낸 바와 같이 처리 지연 시간 T_b 를 고려하면 어떤 스테이션 S_i 에서의 최상위 우선순위에 대한 최대 토큰 보유시간은 $HPTHT_i + T_b + A_b$ 가 된다.

이 값이 커질수록 토큰 회전 시간은 늘어나므로 빠른 응답시간을 제공하는데 불리하게 된다. 한 프레임으로 묶는 최적의 주기데이터 갯수는 이와 같은 특성을 고려하여 적절히 설정되어야 한다. A_b 가 작은 값을 갖도록 하면 통신 오버헤드가 차지하는 비율이 높아져서 처리율이 낮아지게 되므로 시스템의 특성 및 요건에 따라 적절한 선택이 필요하다.

각 스테이션 S_i 의 우선순위 k 에 해당되는 큐가 매번 토큰을 받을 때마다 해당 큐에서 대기중인 모든 통신요구를 처리해야 실시간 조건을 만족한다. 즉, IEEE 802.4의 우선 순위 6과 우선 순위 4의 큐에 들어온 데이터를 토큰 수신시 모두 전송하여야 실시간 조건을 만족할 수 있기 때문에, 이를 가능하게 해주는 최상위 우선순위 토큰 보유시간 ($HPTHT$, High Priority Token Holding Time)과 목적 토큰 회전시간 ($TTRT$, Target Token Rotation Time)의 값을 설정해 주어야 한다.

스테이션 S_i 의 우선순위 k 에서 토큰을 전달받은 시점을 기준으로 한 토큰 회전 시간을 TRT_i^k , 이전의 매체사용을 완료한 시점 혹은 토큰을 전달한 시점으로부터 이번의 매체사용 완료시점 혹은 토큰 전달 시점까지의 시간을 $TRT(+)_i^k$ 라고 하고, TRT_i^k 및 $TRT(+)_i^k$ 의 최대값을 T_r^{\max} 라하고, 스테이션 S_i 의 우선순위 4의 목적 토큰 회전시간을 $TTRT_i^4$ 라 하자. 이때, T_r^{\max} 에 의해 최대전송 시간 지연이 결정되고, 실시간성 조건을 만족시키기 위하여 T_r^{\max} 는 최소한 다음 조건을 만족하여야 한다.

$$T_r^{\max} \leq T_D^{\min}, \quad (1)$$

여기서

$$T_D^{\min} = \min_{i=1}^N (T_{DU}, b_i^h(1), \dots, b_i^h(b_i^h), b_i^s(1), \dots, b_i^s(b_i^s))$$

이다.

이때, T_r^{\max} 는 다음을 만족한다

$$T_r^{\max} \leq (\max_{i=1}^N TTRT_i^4 + T_b + A_b) + \sum_{i=1}^N (HPTHT_i + T_b + A_b) + \sum_{i=1}^N (T_b + T_i + T_o) \quad (2)$$

(2)의 우변에서 첫 번째 항은 우선 순위 4의 큐에 저장된 데이터를 전송하는데 소요되는 시간의 합을 의미한다.

한 스테이션 S_i 에서 $TRT(+)_i^4$ 동안에 발생할 수 있는 연성 실시간 주기 데이터에 대한 최대 전송요구가 A_i^{sp} 라고 할 때, 우선순위 4에서의 최대 토큰 보유시간은 $A_i^{sp}(T_b + A^b)$ 이다. 또한 T_r^{\max} 를 사용하여 우선 순위 4의 데이터에 해당되는 연성 실시간 주기데이터에 대한 최대 전송요구는 (3)과 같다.

$$A_i^{sp} \leq \sum_j^b | T_r^{\max} / b_i^s(j) | = b_i^s \quad (3)$$

(2)의 첫 번째 항을 (3)을 사용하여 대체함으로써 다음과 같이 T_r^{\max} 의 보다 엄격한 범위를 구할 수 있다

$$T_r^{\max} \leq \sum_i^N b_i^s (T_b + A^b) + \sum_{i=1}^N (HPTHT_i + T_b + A_b) + \sum_{i=1}^N (T_b + T_i + T_o) = (T_b + A^b) \sum_i^N b_i^s + \sum_{i=1}^N HPTHT_i + N(2T_b + A_b + T_i + T_o) \quad (4)$$

(4)에 의해 주어지는 T_r^{\max} 의 상한값이 T_D^{\min} 보다 작은 경우 (1)을 만족시킨다. 이 관계식을 $HPTHT_i$ 에 대해 정리하면 (5)를 얻을 수 있다.

$$\sum_{i=1}^N HPTHT_i \leq T_D^{\min} - (T_b + A^b) \sum_i^N b_i^s - N(2T_b + A_b + T_i + T_o) \quad (5)$$

한 스테이션 S_i 에서 $TRT(+)_i^6$ 동안에 발생할 수 있는 긴급데이터에 대한 최대 전송요구가 A_i^g 라고 하고, 경성 실시간 주기 데이터에 대한 최대 전송요구가 A_i^{hp} 라고 할 때, $HPTHT_i$ 동안 모두 전송하기 위해서는 다음 조건을 만족하면 된다.

$$(A_i^g - 1)(T_b + A^g) + A_i^{hp}(T_b + A^b) < HPTHT_i \quad (6)$$

이 경우 각 긴급데이터 및 경성실시간 주기데이터는 최대 $TRT(+)_i^6$ 의 전송 지연을 갖는다. 즉, T_r^{\max} 의 최대 전송 지연값을 갖는다.

토큰 회전 시간이 T_r^{\max} 인 경우에 대하여 다음의 관계가 성립한다.

$$A_i^g \leq A_i, | T_r^{\max} / T_{DU} | = A_i, \quad (7)$$

$$A_i^{hp} \leq \sum_j^b | T_r^{\max} / b_i^h(j) | = b_i^h \quad (8)$$

따라서, $HPTHT_i$ 는 (6)과 (7)에서 구한 최대 갯수를 모두 서비스되도록 설정되어야 하므로, (6), (7)과 (8)을 사용하여 (9)를 구할 수 있다.

$$HPTHT_i > (A_i - 1)(T_p + \Delta^u) + b_i^h(T_p + \Delta^h) \quad (9)$$

(5)와 (9)를 이용하면 $HPTHT_i$ 의 유효 범위를 구할 수 있다.

한 스테이션 S_i 에서 $TRT(+)_i^h$ 동안에 발생할 수 있는 연성 실시간 주기 데이터에 대한 최대 전송요구가 A_i^h 라고 할 때, 토큰 회전 시간 TRT_i^h 에 대한 토큰 보유 시간 $TTRT_i^h - TRT_i^h$ 동안 모두 전송하기 위해서는 일반적으로는 다음 조건을 만족하면 된다.

$$(A_i^h - 1)(T_p + \Delta^h) < TTRT_i^h - TRT_i^h$$

우선 순위 4의 큐가 소요할 수 있는 최소 시간은 $TTRT_i^h - T_r^{\max}$ 이다. 연성실시간 주기 데이터가 최대로 들어올 경우라 하더라도 이러한 최소시간 내에서 모든 데이터를 서비스할 수 있어야한다 (3)과 (4)에 의해 주어지는 A_i^h 및 T_r^{\max} 의 상한값을 적용하고 $TTRT_i^h$ 에 대해 정리하면 매우 보존적이지만 다음 식 (10)을 얻을 수 있다.

$$TTRT_i^h > (b_i^h - 1)(T_p + \Delta^h) + (T_p + \Delta^h) \sum_{j=1}^N b_j^h + \sum_{j=1}^N HPTHT_j + N(2T_p + \Delta_p + T_i + T_o) \quad (10)$$

이 경우 각 연성실시간 주기데이터는 $TRT(+)_i^h$ 의 최대 전송 지연을 갖는다. 즉, T_r^{\max} 의 최대 전송 지연값을 갖는다.

원하는 $HPTHT_i$ 들과 $TTRT_i^h$ 들을 구하기 위해서는 미리 주어지는 상수 $T_p, T_i, T_o, \Delta_u, \Delta_p$ 와 통신량 및 실시간성 요구사항에 따라 주어지는 $T_{DU}, A_i, b_i^h(j)$ ($j=1, 2, \dots, b_i^h, i=1, \dots, N$), $b_i^h(j)$ ($j=1, 2, \dots, b_i^h, i=1, \dots, N$)을 사용하여, (5)와 (9)를 만족하는 적당한 $HPTHT_i$ 들을 구한 후, (10)을 만족시키는 적당한 $TTRT_i^h$ 들을 구하면 된다. 여기서 구한 $HPTHT_i$ 들과 $TTRT_i^h$ 들에 의해 얻어지는 긴급데이터, 경성실시간 주기데이터 및 연성실시간 주기데이터의 최대전송지연은 (4)에 의해 얻어지는 T_r^{\max} 이다

이때, $A_i/T_{DU}, 1/b_i^h(j), 1/b_i^h(j), b_i^h$ 및 b_i^h 이 큰 경우 최대 통신요구량의 상한값이 커지고, 토큰을 전달받고 나서 대기중인 모든 통신요구를 수용하기 위하여 큰 $HPTHT_i$ 들과 $TTRT_i^h$ 값을 설정하여야 한다. 이 경우 T_r^{\max} 가 증가하여 실시간성을 만족하기 매우 어렵게 된다. 주기데이터들은 발생하는 시점이 정해져 있으므로, 사전 스케줄링을 하여 전송요구를 고르게 분포시킴으로써 b_i^h 및 b_i^h 을 작게 만들 수 있다. 즉, 공통적인 단일한 주기를 가지고 표 3과 같은 사전 스케줄링한 결과표에 따라 전송하도록 한다.

공통 주기에 동기되어 주기 데이터들을 해당 주기내에 전송하고 긴급 데이터들을 전송할 수 있도록 전송 대역폭을 마련하기 위해서는 주기 데이터들의 전송 시점을 조정해야 한다. 이러한 조정 작업을 스케줄링이라 하며, 표 3과 같이 주기 데이터들을 전송 순서를 공통 주기 단위로 나열

표 3. 스케줄링 테이블의 예.

Table 3. Example of a scheduling table.

전송순서	1	2	3	4	전송데이터수
단위전송열 1	D(1)	D(2)	D(4)	D(7)	4
단위전송열 2	D(1)	D(3)	D(5)		3
단위전송열 3	D(1)	D(2)	D(6)		3
단위전송열 4	D(1)	D(3)	D(4)	D(8)	4
단위전송열 5	D(1)	D(2)	D(5)		3
단위전송열 6	D(1)	D(3)	D(6)		3

한 것을 스케줄링 테이블이라 한다. 표 3의 경우는 D(i) ($i=1, \dots, 8$)의 8개의 주기데이터를 스케줄링한 예를 보이고 있다. 스케줄링 테이블은 단위전송열로 이루어지며 매 공통주기마다 단위전송열을 구성하는 데이터를 순서대로 전송한다.

단위전송열의 갯수를 스케줄링 테이블의 크기라고 하며, 스케줄링 테이블의 크기를 l 로 표시하고, 공통주기를 p 로 표시하면, lp 마다 주기 데이터들은 반복되어 전송되며, 이러한 반복 주기를 매크로 주기라 한다. k ($k=1, \dots, l$) 번째 전송열은 k 번째 주기에서 전송되는 주기 데이터들로 이루어진다.

사전스케줄링에 의해 단위전송열의 최대길이가 b 가 되면, 통신량의 특성은, 스케줄링의 결과로서 최소주기가 p 이고 각각은 p 이상의 주기를 갖는 b 개의 주기데이터와 같게 된다.

주기가 $p_r(j)$ ($j=1, \dots, b_r$)인 주기데이터들의 공통 전송주기를 p ($p \leq p_r(j), j=1, \dots, b_r$)로 잡으며, 주기가 $p_r(j)$ 인 데이터 발생시점으로부터 $p_r(j) - p$ 이내에 전송요구를 발생시키면 된다. p 가 작을수록 단위전송열의 최대길이가 작아진다. 스케줄링 테이블의 크기가 증가하게 되며 작은 토큰회전시간을 만족하여야 하므로 적절한 선정이 필요하다. 일반적으로 p 는 $p_r(j)$ 들의 최소공약수와 최소값사이의 적절한 값으로 정하면 된다.

사전스케줄링 및 적절한 통신량 매개변수의 설정에 의해 실시간 주기 및 비주기 데이터들의 실시간성을 만족시키기 위한 방법을 정리하면 다음과 같다.

1) 각 스테이션별로 경성실시간 주기데이터 및 연성실시간 주기데이터에 대해 적절한 사전스케줄링을 한다. 공통전송주기와 단위전송열의 최대길이를 경성 및 연성실시간 주기데이터에 대해 각각 $\hat{p}_i^h, \hat{b}_i^h, \hat{b}_i^h, \hat{b}_i^h$ 라고 하면, 이후의 매개변수 설정과정은 경성실시간 주기데이터는 \hat{b}_i^h 개가 각각 주기 \hat{p}_i^h 이고 연성실시간 주기데이터는 \hat{b}_i^h 개가 각각 주기 \hat{p}_i^h 인 통신량에 대하여 계산한다.

2) 1)에서 얻어진 $\hat{p}_i^h, \hat{p}_i^h, \hat{b}_i^h, \hat{b}_i^h$ 및 T_{DU}, A_i 에 대해 (5), (9)를 적용하여 각 스테이션에서 $HPTHT_i$ ($i=1, \dots, N$)를 구한다.

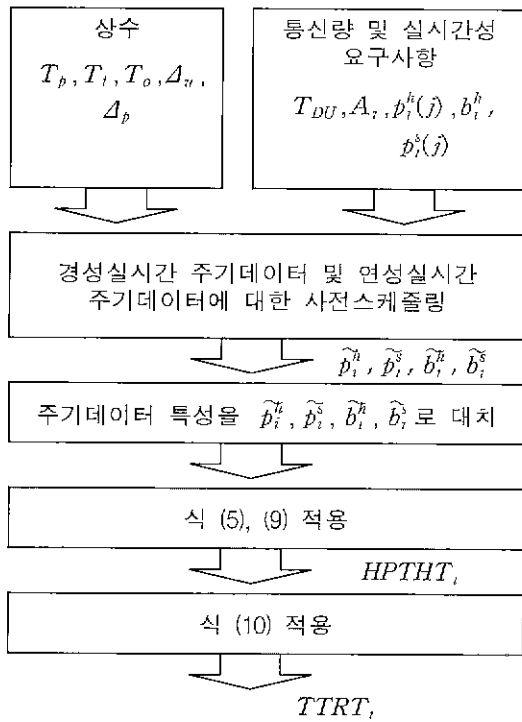


그림 4. 매개변수 설정 절차.

Fig 4 Procedure to set the parameters.

3) 2)에서 얻어진 $HPTHT_i$ ($i=1, \dots, N$)에 대해 (10)을 적용하여 $TTRT_i$ ($i=1, \dots, N$)를 구한다.

그림 4는 이러한 절차를 정리하여 보여주고 있다.

V. 적용예

5 Mbps의 전송률을 가정하면, 긴급데이터의 프레임은 31 byte이므로 Δ_n 는 49.6us이다. 각 주기데이터는 16개씩

표 4. 스테이션들의 통신 발생량.

Table 4. Data traffic of stations.

스테이션 번호 i	긴급 데이터 프레임 T_{DU} ms	간접 데이터 요구 수 A_i	경성 실시간 주기 데이터 종류 및 수 $p_i^h(j)$ ms (b_i^h)	정성 실시간 주기 데이터 종류 및 수 $p_i^s(j)$ ms (b_i^s)	연성 실시간 주기 데이터 양
1	10	4	10, 30, 30, 40 (4)	주기당 16개씩 130, 140, 160, 170, 190, 200, 220, 250, 300, 320, 350, 420 (12)	주기당 16개씩
2	10	8	20, 60, 60, 60 (-4)	주기당 16개씩 150, 160, 180, 190, 200, 210, 240, 250, 350, 360, 380, 400 (12)	주기당 16개씩
3	10	4	80, 80, 90, 100 (-4)	주기당 16개씩 120, 140, 160, 170, 180, 190, 200, 210, 220, 230 (10)	주기당 16개씩
4	20	8	40, 80, 80, 80, 80 (5)	주기당 16개씩 20, 30, 40, 50, 160, 180, 210, 240, 300, 400 (10)	주기당 16개씩
5	20	5	10, 10, 20, 20, 30, 30, 40, 40, 50, 50 (10)	주기당 16개씩 20, 30, 40, 40, 40, 100, 100, 120, 120, 200, 300, 200 (12)	주기당 16개씩
6	20	6	20, 20, 30, 30, 30, 40, 40, 40, 50, 50 (10)	주기당 16개씩 40, 40, 40, 80, 80, 90, 90, 120, 120, 120, 150, 150 (12)	주기당 16개씩

하나의 프레임으로 전송되어진다고 가정하면, 주기데이터 프레임은 최대 105 byte이므로 Δ_p 는 168.0us이다. 토론 프레임은 21 byte로서 T_f 는 33.6us이다. T_p 를 10us로 가정하고, T_o 를 30us로 가정하자. 스테이션의 갯수를 6이며, 각 스테이션의 통신 발생량은 표 4와 같다. 사전 스케줄링은 휴리스틱한(heuristic) 방법을 사용한다.

본 논문에서는 긴급데이터의 발생량에 의해 결정된 T_D^{min} 값 10ms를 기준으로 경성 및 연성 실시간 데이터의 공통 주기를 잡았다. 본 논문에서는 최대 스케줄링 테이블의 길이가 제한되는 경우를 가정하였으며, 최대 스케줄링 테이블의 길이 L 을 10으로 하였다. 또한, 공통주기 \hat{p}_i^h 및 \hat{p}_i^s 이 작을수록 불필요한 통신량이 증가할 수 있으므로, 같은 \hat{b}_i^h 및 \hat{b}_i^s 값을 갖는 스케줄링 테이블의 길이 L 이 10을 만족하는 조건하에서 \hat{p}_i^h 및 \hat{p}_i^s 의 최대값을 잡도록 하였다. 따라서, \hat{p}_i^h 및 \hat{p}_i^s 가 10ms인 경우의 \hat{b}_i^h 및 \hat{b}_i^s 값을 구한 후 이 값을 갖는 \hat{p}_i^h 및 \hat{p}_i^s 의 최대값과 이때의 스케줄링 테이블을 구한다. \hat{b}_i^h 및 \hat{b}_i^s 를 구하는 방법은 다음과 같다

주어진 $p(j)$ ($j=1, 2, \dots, b$)에 대하여 스케줄링 테이블의 길이가 L 로 제한되는 경우를 고려하면, 다음의 관계를 얻을 수 있다. 우선 공통전송주기를 \hat{p} 라고 하면 $\hat{p} \leq p(j)$ ($j=1, 2, \dots, b$)이고, j 번째 데이터는 스케줄링 테이블에서 최소한 $\lceil L / \lceil p(j) / \hat{p} \rceil \rceil$ 번 나타나게 되며, 전송 간격은 $\lceil p(j) / \hat{p} \rceil$ 이하가 되어야 한다. 여기서 $\lceil x \rceil$ 는 x 보다 크지 않은 최대 정수를 말한다.

따라서 스케줄링 테이블 내에서 존재하여 관리되어야 하는 전송 데이터들의 최소 갯수는 $\sum_{j=1}^b \lceil L / \lceil p(j) / \hat{p} \rceil \rceil$ 가 된다. 가장 스케줄링이 잘 된 경우는 전송이 스케줄링 테이블에 고르게 분포하는 경우이고 이때 한 전송열이 가질 수 있는 최대 길이는 $\lambda(\hat{p}) = \lceil (\sum_{j=1}^b \lceil L \cdot \lceil p(j) / \hat{p} \rceil \rceil) / L \rceil$ 이 되며, $\hat{b} \geq \lambda(\hat{p})$ 의 관계를 가진다. 주어진 p_0 에 대하여 $p(j)$ 들보다 작고 p_0 보다 같거나 큰 최적의 공통 전송주기값은, 주어진 범위내에서 $\lambda(\hat{p}) = \lambda(p_0)$ 이고 스케줄링 테이블의 최대전송열의 길이가 $\lambda(p_0)$ 인 \hat{p} 의 최대값, 또는 최대전송열의 길이가 p_0 에 대한 스케줄링 테이블의 최대 전송열의 길이와 같은 \hat{p} 의 최대값이다.

스케줄링은 주기가 가장 작은 데이터로부터 주기가 큰 데이터의 순으로 한다. 각 주기데이터는 스케줄링 테이블의 전송열들의 길이가 최대한 고르게 분포하도록 배치시킨다. 즉, j 번째 주기 데이터의 경우 $\lceil L / \lceil p(j) / \hat{p} \rceil \rceil$ 번의 전송이 $\lceil p(j) / \hat{p} \rceil$ 이하의 전송열간 거리를 가지고 나타나도록 하되 각각은 길이가 가장 작은 전송열에 할당하고 길이가 같은 전송열이 여러개 있을 때는 가장 많이 몰려있는 중간에 위치한 전송열에 할당한다.

위와 같은 방법을 사용하면 표 5와 같이 경성 및 연성 실시간 데이터의 공통주기 \hat{p}_i^h 및 \hat{p}_i^s 가 결정되고 스케줄링 테이블을 얻을 수 있다. 스케줄링 테이블에서 하나의 전송열은 {a, a', b, c..}로 표시되며, a, a', b, c 등은 데이터

의 주기를 나타내며, a 와 a' 는 같은 주기이지만 다른 데이터를 표시한다.

표 5와 (9)에 의해 $HPTHT_i$ ($i=1,2,3,4,5,6$)는 각각 0.53ms, 0.60ms, 0.42ms, 0.60ms, 1.13ms, 1.01ms이상의 값을 가지면 되며, 표 5와 (5)를 만족시키기 위해서 $\sum_{i=1}^N HPTHT_i$ 는 6.18ms보다 작으면 된다. (9)에 의해 얻어지는 $HPTHT_i$ 의 하한값들의 합은 4.28ms이므로 주어질 통신량에 대하여 실시간성 조건을 만족하는 $HPTHT_i$ 들이 존재한다는 것을 알 수 있다. $HPTHT_i$ 가 $i=1,2,3,4,5,6$ 에 대해 각각 0.53ms, 0.60ms, 0.42ms, 0.60ms, 1.13ms, 1.01ms의 값을 가질 경우, (10)에 의해 $TTRT_i^d$ ($i=1,2,3,4,5,6$)는 각각 8.28ms, 8.28ms, 8.10ms, 8.28ms, 8.46ms, 8.46ms보다 큰 값을 가질 수 있다.

분산제어시스템에서는 예측하지 못한 처리시간지연이나 트래픽증가에도 안정적인 동작을 보장하기 위하여 대역폭에 충분한 여유를 두는 경우가 많다. 시스템의 안정적인 동작 및 신뢰성이 중요한 시스템일수록 이 여유분을 크게 하는 경향이 있으며, 시스템의 특성이나 요건을 고려하여 시스템 설계자가 적절히 선택해야 한다. 약간의 여유를 두기 위하여 $HPTHT_i$ ($i=1,2,3,4,5,6$)를 각각 0.7ms, 0.7ms, 0.6ms, 0.7ms, 1.2ms, 1.2ms로 설정하면 $TTRT_i^d$

($i=1,2,3,4,5,6$)는 각각 9.10ms, 9.10ms, 8.92ms, 9.10ms, 9.28ms, 9.28ms보다 큰 값을 가지면 되고, 역시 약간의 여유분을 고려하여 $TTRT_i^d$ ($i=1,2,3,4,5,6$)는 각각 9.3ms, 9.3ms, 9.1ms, 9.3ms, 9.5ms, 9.5ms로 설정하면, 요구되는 실시간성을 만족시킬 수 있다.

앞에서 사용한 전송방식을 사용하면 목적 토큰 회전시간은 충분히 큰 값으로 설정하기만 하면 되고, 그 이상의 작용을 하지 않게 된다. 즉, 최상위 전송 데이터는 최상위 우선순위 토큰 보유 시간으로 제어되고, 그 이하의 우선순위에 대한 전송 데이터는 스케줄링에 기반하여 매체의 사용 및 전송이 제어된다.

VI. 결론

원자력 발전소의 제어시스템과 같은 안전 시스템에 사용되는 통신망에서는 엄격한 실시간성이 요구되는 비주기성 데이터의 전송요구와 엄격한 주기의 준수를 요하는 경성 실시간 주기 데이터의 전송요구가 동시에 존재하며, 역시 연성 실시간 주기 데이터의 전송요구가 존재한다. 비주기적인 데이터와 주기적인 데이터는 발생특성 및 실시간 요구사항의 차이에 의하여 각각 별도의 전송방법을 사용하여 처리하는 경우가 일반적이다. 본 논문에서는 우선순위를 가지는 타이머 제어형 토큰 버스 프로토콜인 IEEE 802.4를 사용하여 비주기적인 특성을 갖는 긴급 데이터와 경성 및 연성 실시간 주기데이터의 실시간 요구사항을 동시에 효과적으로 만족시키는 방법을 제안하였고, 이 방법의 유효성을 예제를 통하여 보였다. 단위시간에 발생하는 비주기 및 주기 데이터의 최대 통신량을 정형화하고, 이로부터 실시간요건을 만족시키기 위한 IEEE 802.4 토큰 버스의 주요 매개변수인 $HPTHT_i$ 와 $TTRT_i^d$ 의 유효 범위 설정방법을 제시하였다. 통신망의 전송능력을 효율적으로 활용하기 위하여 통신요구의 발생시점이 예측가능한 주기 데이터에 대한 처리시점을 사전에 스케줄링하는 방법을 동시에 적용하였다. 논문에서는 주어질 실시간 조건들을 만족시키도록 각 주기 데이터들의 전송시점을 적절히 조정시켜 주는 스케줄링 테이블을 만드는 방법을 휴리스틱하게 제시하였다.

앞으로 사전 스케줄링 방법의 개선, 주어질 성능지표를 최적화 시키는 $HPTHT_i$ 와 $TTRT_i^d$ 의 설정방법 등에 대한 연구가 있을 수 있다.

참고문헌

[1] D. Popovic and V. P. Bhatkar, *Distributed Computer Control for Industrial Automation*, Marcel Dekker, Inc., 1990.
 [2] S. Zhang, *Synchronous Bandwidth Allocation for Time-Critical Communication*, PhD Thesis, Dept of Computer Science, University of York, 1996.
 [3] S. Zhang and A. Burns, *Timing Properties of the Timed Token Protocol*, Tech. Rept. (YCS-94-243), Dept. of Computer Science, Univ. of York, 1994.

표 5. 스테이션들의 스케줄링 테이블과 매개변수들의 설정치

Table 5. Scheduling table and values of parameters of stations.

스테이션 번호 i	T_D ms	A_i	\hat{D}_i^h ms	경성 실시간 주기 데이터의 스케줄링 테이블	\hat{D}_i^k ms	\hat{D}_i^j ms	연성 실시간 주기 데이터의 스케줄링 테이블	\hat{D}_i^s
1	10	1	10	{(10,30), (10,30)', (10,40)}	2	20	{(130,200), (170, 140,250), (130,330), (190), (130,230), (170), (140,350), (160,300), (130,430)}	2
2	10	8	20	{(20), (40), (20), (60)', (20), (60)'}	1	30	{(150,210), (180,250), (180,400), (100,240), (200,350), (150,210), (190,360), (180,250), (160,240), (200,380)}	2
3	10	4	20	{(80), (80)', (80), (100)}	1	10	{(120), (140), (160), (170), (180), (180), (200), (210), (220), (230)}	1
4	20	8	20	{(40), (80), (30)', (80)'', (80)'}	1	10	{(30,160), (30,180), (20,50), (40,210), (20,30'), (40,240), (20,300), (30,50), (20,400), (30,40)}	2
5	20	5	10	{(10,10',20,30), (10,10',20',30'), (10,10',30,40,50'), (10,10',20',30,40'), (10,10',30',50)}	5	10	{(30,40,100'), (20',40',120'), (30,10',120'), (20',100,120''), (20,40,200), (20',40',200'), (20,40',200''), (20',40'), (20',40')}	3
6	20	6	10	{(20,30,40), (20',30',40',50'), (20,30',40'), (20',30',40,50), (20,30',40'), (20',30',40'), (20,30',40,50'), (20',30',30'), (20,30',40',50), (20',30',30',40')}	4	10	{(20,40,100'), (20',40',120'), (20,10',120'), (20',100,120''), (20,40,200), (20',40',200'), (20',40',200''), (20',40'), (20',40')}	3

[4] N. Malcolm and W. Zhao, "The timed-token protocol for real-time communications," *IEEE Computer*, 27(1): pp. 35-41, January, 1994.

[5] H. Kopetz and G. Grunsteidl, "TTP - a protocol for fault-tolerant real-time systems," *IEEE Computer*, vol 27, no. 1, pp. 14-23, January, 1994.

[6] J. D. Decotognie and P. Raja, "Fulfilling temporal constraints in fieldbus," *Proceedings of the IECON'93*, pp. 519-524, 1993.

[7] J. W. Park, Y. S. Kim, and et al, "Network conscious design of distributed real-time systems," *Journal of System Architecture*, vol. 45, no. 2, pp. 131-156, 1998.

[8] K. W. Tindell, H. Hansson, and A. J. Wellings, "Analyzing real-time communications: controller area network(CAN)," *IEEE Real-Time Systems Symposium*, pp. 259-263, 1994.

[9] Helmut Rzehak, Abd E. Elnakhal, and Rudolf Jaeger, "Analysis of real-time properties and rules for setting protocol parameters of MAP network," *The Journal of Real-Time Systems*, 1, pp. 221-241, 1989.

[10] M. Li, *Real-time Communication in an Industrial Network -- Profibus*, PhD Dissertation, Lausanne, EPFL, 1996.

[11] Kang G Shin and Chih-Che Chou, "Design and evaluation of real-time communication for field bus-based manufacturing systems," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 357-367, June, 1996.

[12] FIP committee, FIP technical Description, www.worldfip.org.

[13] F/F committee, F/F technical Description, www.fieldbus.org.

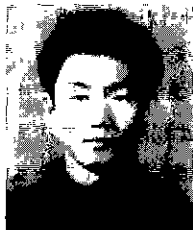
[14] IEEE, *Token passing Bus Access Methods and Physical Layer Specification: ANSI/IEEE Std. 802.4*, 1990.

[15] Profibus committee, Profibus technical Description, www.profibus.com.

[16] 표준경수로 계통설비 (상), (하), 한국전력공사 영광 원전훈련센터, 1996.

[17] K. C. Lish, *Nuclear Power Plant Systems and Equipment*, Industrial Press Inc., 1972.

[18] 원전 자동제어설비의 디지털화 개발 (II) - 최종보고서, 한국전력공사 전력연구원, 1999.



문 홍 주

1968년 5월 22일생. 1991년 서울대학교 제어계측공학과 졸업, 1993년 동대학원 석사, 1998년 동대학원 박사. 1998년~현재 전력연구원 선임연구원. 관심분야는 산업용 통신망, 이산현상시스템, 분산제어시스템, 발전소 제어 및 응용.



박 홍 성

1961년 3월 16일생 1983년 서울대학교 제어계측공학과 졸업, 1986년 동대학원 석사, 1992년 동대학원 박사. 1983년~1990년 삼성전자 근무. 1992년~현재 강원대학교 전기전자정보통신공학부 부교수. 관심분야는 실시간 네트워크, 개방형 구조 제어 시스템, 이산현상시스템, 모발 네트워크 분석.