'2000 춘계학술발표회 논문집
한국원자력학회

원전 통신망의 재전송 지연 영향
Retransmission Delay Effect of Data Communication System in Nuclear Power Plant

송기상, 한병래

한국교원대학교
충북 청원군 강내면 다락리 산 7

서용석, 김동훈

한국원자력연구소
대전시 유성구 덕진동 150

요 약

통신에서 에러는 모든 매체들에서 발생되고 이러한 에러의 발생으로 말미암아 원전 통신망에서는 발전소 운영에서 결정적인 데이터를 잃어버릴 가능성이 있다. 그러므로 발전소의 운영에 장애를 초래하는 에러는 허용되어질 수 없고, 적절한 에러관리가 필요하게 된다. 일반적으로 통신망에서의 에러관리의 양 또는 세부 기술은 에러에 대한 응용들의 허용수준에 의존하는데, 본 논문에서는 메시지의 에러가 있을 경우 재전송을 요구하는 방식에서의 지연을 계산하여 기존 프로토콜의 재전송 지연에 따른 원자력 발전소의 적합성을 평가한다.

Abstract

The communication error occurs in every communication media and it is possible to lose the critical data for nuclear power plant operation. Therefore, it is not allowable communication error for nuclear power plant safety and needs appropriate error management. The amount of error management and detail techniques are dependent error tolerant capability of applications. In this paper, we compute the message delay due to the retransmission after the error is detected and show which protocol is suitable for nuclear power plant application in terms of message delay.

1. 서 론

통신에서 에러는 모든 매체들에서 발생된다. 이러한 에러의 발생으로 말미암아 원전 통신망에서는 운영에서 결정적인 데이터를 잃어버릴 가능성이 있다. 따라서 적절한 에러관리가 필요하게 된다. 에러관리의 목적은 에러가 발생하지 않게 하고 에러가 발생했다면 정정할 수 있게 하며전송되는 정보를 완전하게 받을 수 있게 해 주는 것이다.

일반적으로 전송 중 발생되는 에러의 제어 방법으로는 에코(echo; ACK 혹은 NACK)를 사용하거나 간단히 패리티 체크를 사용하는 방법이 있고[1] 보다 강력한 코드를 사용하는 경우도 있다. 이러한 코드의 사용에는 다항식을 이용한 표준이 사용되며 대표적인 방법으로는 12비트와 16비트 CRC(Check Redundancy Code) 코드, 16 비트의 CRC-CCITT 코드 및 CRC-32 코드가 있다. FDDI와 토큰 링의 경우에는 16비트의 CRC 코드를 사용하며 이러한 코드에서는 버스트 에러를 16비트 CRC 코드의 경우, 단일 및 복수에러, 홀수개의 비트 수를 갖는 모든 에러, 길이가 16이거나 보다 적은 버스트 에러, 17비트 버스트 에러의 99.997퍼센트를, 그리고 18비트 이상의 에러 버스트에 대해서는 99.998퍼센트의 에러를 검출할 수 있다[2].

에러 제어에 대한 다른 접근 방법으로는 Automatic Repeat Request(ARQ)라는 수신측에서 데이터를 수신한 후에 에러가 검출되면 송신측에 재전송을 요구하는 프로토콜이 있다. 이러한 ARQ 프로토콜에는 크게 Stop-and-Wait ARQ와 Continuos ARQ 및 Adaptive ARQ방식이 있다.

2. 재전송에 의한 에러 보정

통신회선에서 에러가 발생했을 경우 수신측은 에러의 발생을 송신측에 알리고, 송신측은 에러가 발생한 프레임을 재전송하게 되는데 이러한 방식을 ARQ라고 한다. ARQ 방식에서 송신측은 항시 수신측으로부터 프레임에 에러가 발생했다는 신호를 받아들일 수 있는 상태에 있어야 한다. 이 때에 송신측은 다음 프레임의 전송 대신에 에러가 발생한 프레임을 재전송해야 한다. 따라서 송신측은 전송 중인 프레임을 기억하고 있어야 할 버퍼가 필요하다. 버퍼의 크기는 기억하고 있어야 할 프레임의 크기와 개수에 의해 결정된다.

(1) 정지-대기 ARQ

송신측에서는 1개의 프레임을 수신측에 송신하고 수신측에서는 수신된 프레임의 에러 유무를 판단해서 송신측에 ACK(Acknowledgement)나 NACK(Noacknowledgement)를 보내는 방식이다. 송신측은 수신측으로부터 ACK를 수신했을 경우 다음 프레임을 전송하며, 수신측으로부터 NACK를 수신하거나 일정시간이 경과할 때까지 응답 프레임 (ACK, NACK)을 수신하지 못할 경우 해당 프레임을 재전송한다. 송신측은 ACK를 수신할 때까지 해당 프레임을 버퍼에 보관하고 있어야 한다.

(2) 연속적 ARQ

정지-대기 ARQ 방식은 한 프레임을 전송한 후 ACK 나 NACK를 수신할 때까지 다음 프레임을 송신할 수 없으므로 전송효율이 좋지 않다. 이러한 단점을 극복하기 위해서 연속적으로 프레임을 전송하는 방식을 연속적 ARQ 라 한다. 연속적 ARQ 방식에는 Go-back-N ARQ 와

Selective-repeat ARQ 의 2종류가 있다.

① Go-back-N ARQ

송신측은 윈도우 크기만큼의 프레임들을 순서번호를 부여하여 연속적으로 전송한다. 수신측은 수신한 프레임에서 에러를 검출했을 경우 NACK 신호와 함께 이 에러 프레임의 순서번호를 송신측에 전송한다. NACK를 수신한 송신측은 NACK 와 함께 전송되어온 순서번호에 해당하는 프레임부터 다시 연속적으로 윈도우 크기만큼의 프레임들을 수신측에 재전송한다.

이 방식은 송신측이 NACK를 수신하면 에러가 검출된 프레임부터 윈도우 크기만큼의 프레임을 전송해야 하므로 송신측의 버퍼의 크기는 최대 윈도우 크기만큼 잡아야 한다.

② Selective-repeat ARQ

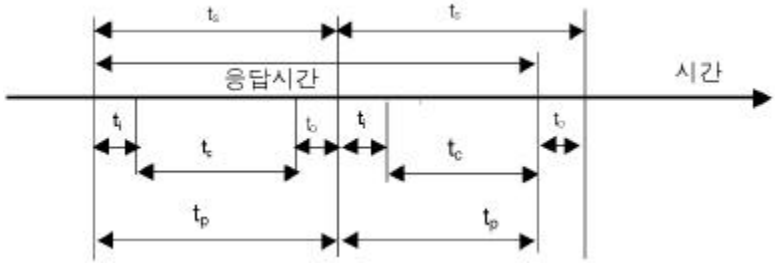
송신측이 NACK를 받은 프레임부터 연속하여 재전송하는 Go-back-N ARQ 와는 달리 NACK 에 해당하는 프레임만을 재전송하는 방식이 Selective-repeat ARQ 방식이다. 이 방식은 Go-back-N ARQ 에 비해 전송효율은 더 좋으나 송신측 입장에서는 상대 수신측의 수신상태에 따라 전송해야 하므로 송신 순서를 벗어난 전송을 많이 하게 되며, 이에 따라 구현방법이 복잡해지는 단점이 있고, 수신측도 마찬가지로 순서없이 오는 프레임을 다시 원래의 순서로 복구시켜야 하는 등 구현할 때의 복잡성 때문에 Go-back-N ARQ 에 비해 적게 사용된다.

③ 적응적 ARQ

전송효율을 최대로 하기 위해서 프레임의 길이를 동적으로 변경시킬수 있는 방식이 적응적 ARQ 방식이다. 수신측은 통신회선의 에러 발생률을 감지하여 송신측에 알려주며, 이 정보에 의해 송신측은 가장 적절한 프레임의 길이를 결정하여 프레임을 전송하게 된다. 물론 이러한 프레임 길이의 결정을 수신측에 통보해야 한다. 이 방식은 통신회선의 품질이 좋지 않아 에러 발생률이 높을 경우 프레임의 길이를 짧게, 에러 발생률이 낮을 경우 프레임의 길이를 길게 조절한다. 이 방식은 전송 효율을 높일수 있는 장점이 있는 반면, 제어회로가 매우 복잡한 단점을 지니고 있다.

2. 원자력 발전소 계측제어 시스템에서의 트래픽 주기와 실시간 요구 사항

원자력 발전소의 MMIS 시스템의 자동화 장비의 타이밍 특징은 다음과 같은 세 가지의 파라미터로 특징 지워질 수 있다.



<그림 1> 간단한 자동화 장치의 타이밍 관계[3]

(1) 샘플링 주기(t_s): 두 개의 프로세스 사이클 사이의 간격

(2) 프로세싱 타임(t_p): 입력 상태 취득 시간(t_i)과 알고리즘 계산시간(t_c) 및 출력 갱신 시간(t_o)이 포함된다.

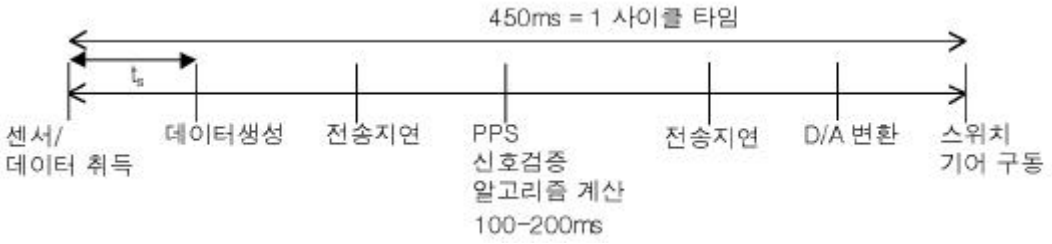
$$t_p = t_i + t_c + t_o$$

(3) 응답시간(t_r): 입력 상태와 여기에 따른 출력 상태 사이의 변화가 나타나는 시간

$$t_r = t_s + t_p = t_s + t_c + t_i + t_o$$

센서와 액튜에이터가 필드 버스와 같은 필드 네트워크 기기로 연결되면, 샘플링 주기가 사이클 타임을 결정하게 된다. 이때, 만일 센서 상태 테이블 갱신 시간(t_i) 및 액튜에이터 갱신 시간(t_o)이 계산 시간과 비슷하게 되면 네트워크에 의한 긴 응답시간이 필요하게 된다. 이에 따라, 로컬 프로세싱과 계산이 이루어지는 동안에 테이블 갱신이 일어나도록 하면 여기에 소요되는 지연을 상당히 줄일 수 있게 된다.

원자력 발전소에서 예상되는 센서 및 액튜에이터 간 사이클 타임은 시스템 간에 차이가 있지만, 일반적으로 다음과 같은 시간이 예상된다.



<그림 2> PPS와 센서 및 액튜에이터 간의 시간 지연 관계[4]

여기서 데이터의 취득과 생성 사이의 시간이 자동화 장비의 상이클 타임 t_s에 해당한다고 볼 수 있다. <그림 2>에서 보이듯이 센서에서 변수 값을 취득하여 신호 검증 및 스위치 구동에까지 소요되는 시간 450ms에서 신호 검증을 위한 지연을 제외하면 통신망에서 소요되는 시간은 250ms 정도가 된다. 이때, 신호검증 과정에 한번의 샘플 데이터를 이용하는 것이 아니라, 5번의 샘플링 데이터가 필요하게 되면 즉 매 20ms 마다 데이터가 샘플링 되어 PPS에 전송되고, 이에따라 5개의 데이터를 이용하여 한 개의 신호를 검증하는 알고리즘의 경우에는 <그림 2>의 관계에서와 같이 매 20ms 마다 PPS로 전송이 이루어져야 한다. 그렇게 되면 통신망에서 허용되는 지연 시간은 최대 20ms를 넘을 수 없으므로, 이와 같은 경성 실시간 조건을 만족시킬 수 있는 통신 실시간 프로토콜이 고려되어야 한다.

원자력 플랜트의 실시간 요구 사항의 한 예로써 <표 1>에는 신형 원자로의 주기적인 트래픽 생성 주기를 나타내고 있다[4].

<표 1> 신형 원자로의 안전 및 비안전 계통의 트래픽 생성 주기 요구조건

구분	기능	마감시한 요구조건
Safety	Control	20 ms
	Monitoring	20 ms
Non-safety	Control	100 ms
	Information processing	500 ms

이와 같은 트래픽 생성 주기 P_i는 결국 실시간 전송 마감시한 D_i와 같다고 가정하면 이 두 값 (P_i, D_i)와 각 프로세서에서 발생되는 메시지 크기와 관련하여 트랙픽 양 C_i가 결정되게 된다. 이러한 세 개의 파라미터에 의하면 실시간 트래픽 (P_i, D_i, C_i) 에 대하여 네트워크의 채널 대역폭이 할당되고, 여기에 따라 발생되는 전송지연 d_i 가 트래픽 발생 주기보다 작도록 네트워크가 설계되어야 한다.

주어진 실시간 요구 사항은 크게 상하치(Upper Hard Real-Time Deadline: $UHRD$)와 하한치(Lower Hard Real-Time Deadline: $LHRD$)로 구분하여 볼 수 있다. 즉, 상한치의 경우에는 이 시간 이내에 반드시 데이터가 도착해야 하는 범위이고, 하한치는 실시간 범위 내에 들어오기 위하여 전송이 이루어져야 하는 마감시한이라고 볼 수 있다. 일반적으로 경성 실시간 요구 조건에서는 마감시한을 상한치로 두고, 즉, $UHRD = D_i$로 가정하고 시스템을 설계하게 된다.

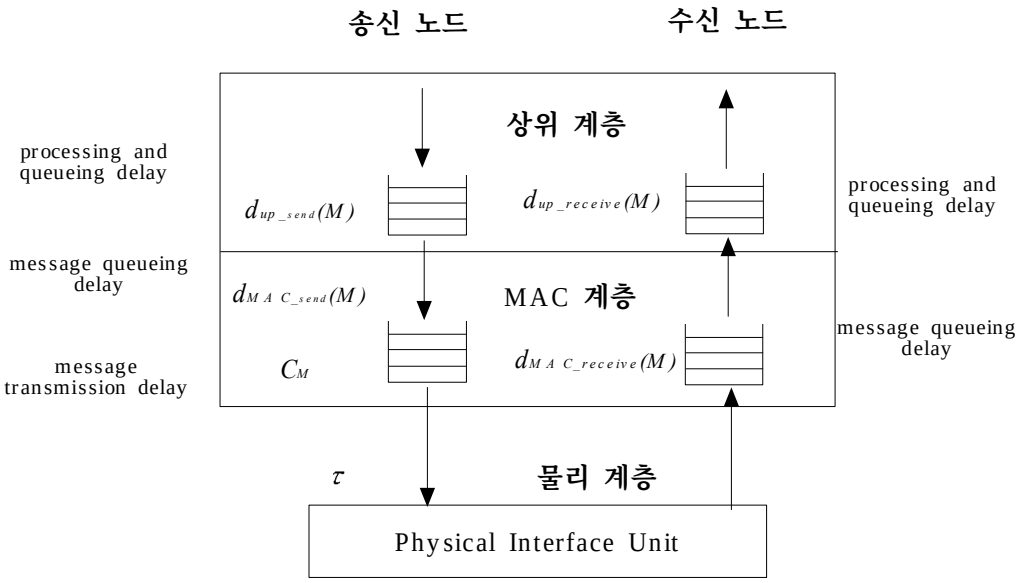
3. 재전송 알고리즘의 선택을 위한 제안

프레임 손실이 크게 예측되는 경우에는 프레임의 에러와 이에 따른 손실을 복구하는 메커니즘이 필요하며, 프로토콜에 따라 다르지만 일반적으로 재전송과 토큰 재 발생 및 네트워크 재구성 방법이 이용된다. LAN과 같은 다중 접근 네트워크에서, 노드들은 하나의 공유된 채널을 경유하여 통신을 하며 다중 접근 네트워크에 있는 노드들은 일반적으로 계층을 이룬 통신 구조를 가지고 있다. 각각의 계층은 각각의 층에 필요한 기능을 수행할 책임이 있는 프로토콜의 다른 집합을 가지고 있다. 따라서, 응용프로그램간에 전송되어질 때 경험되어지는 다양 종류의 지연을 고려해야한다. 이들 지연의 이해는 메시지의 제한시간을 만족하는지를 고려할 때 중요하다.

가) 메시지 지연의 모델링

원자력 발전소와 같이 분산된 경성 실시간 시스템을 지원하기 위하여 다중 접속 네트워크를 사용할 때, 응용 프로그램과 응용프로그램간의 시간지연은 응용 프로그램이 만족 시켜야 하는 제한시간을 만족시킬 수 있는 지를 결정하는 중요한 요인이 된다. 응용 프로그램과 응용 프로그램간의 지연은 응용프로그램 작업들간에 전송되어지는 메시지에 의해 경험되어지는 시간지연이다. 응용프

로그램에서 이루어지는 작업은 자주 다른 노드들에서 실행된 것을 참조로 하게 된다. 따라서 메시지는 전체 네트워크를 가로질러 전송되어져야만 한다. 이 때문에 응용 프로그램간 지연은 시스템의 구현 방법에 의존하게 된다. 이런 종류의 시스템의 구현에 있어서 메시지 M의 전송에 나타나는 시간지연은 아래 <그림 3> 처럼 분해되어 질 수 있다[5].



<그림 3> 메시지 전송에 나타나는 시간지연 요소

(1) 전송 노드의 상위 계층에서 처리와 대기 행렬에서의 지연

처리 지연은 메시지의 헤더를 만드는데 필요한 시간을 포함한다. 대기 행렬 지연은 메시지가 처리되기 위해 기다려지는 시간과 낮은 계층의 프로토콜을 통과하기 위해 기다리는 시간을 모두 포함한다. 상위 계층의 송신 노드에서 메시지 M 에 의해 경험되어질 수 있는 최대 지연을 $d_{up_{send}}(M)$이라 나타낸다.

(2) 송신 노드의 MAC 계층에서의 대기 행렬 삽입의 지연

이 지연은 MAC 프로토콜에 의해 송신되어지는 메시지가 전송을 기다릴 때 발생한다. 메시지 M이 MAC 계층의 송신 노드에 의해 경험되어질 수 있는 최대 대기 행렬 삽입 지연을 $d_{MAC_{send}}(M)$으로 나타낸다.

(3) 메시지 전송 지연

이 지연은 물리적으로 메시지를 전송하는 데 요구되어지는 시간이다. 메시지 M의 전송 지연은 C_M이라 나타낼 수 있으며, 이상적인 경우는 $C_M = F/R$ 로 정의 된다. 이때, F는 전송을 기다리는 전체 프레임 비트 수이며, R은 링크의 전송율이다.

(4) 전달 지연

이 지연은 한 개의 비트가 채널을 통해 수신노드에 전달되어지는데 필요한 시간이다. 이 지연

을 τ 라고 나타낸다.

(5)수신노드의 MAC 계층에서의 메시지 대기행렬 삽입 지연

메시지가 목적지에 도착하면, 이것은 메시지가 상위 계층의 프로토콜에 전해질 수 있을 때까지 수신 노드의 MAC 계층의 큐에 저장된다. 수신노드의 MAC 계층에 있는 메시지 M에 의해 경험되어질 수 있는 최대값을 $d_{MAC\text{-}receive}(M)$이라고 나타낸다.

(6) 수신 노드의 상위 계층에서의 처리와 대기행렬삽입 지연

메시지가 상위 계층에 전달되고 난 후, 이것은 메시지 헤더가 제거되고 데이터가 수신 응용 작업에 전달되기 전에 다시 대기 행렬에 삽입될 것이다. 수신노드의 상위 계층에서 메시지 M에 의해 경험되어질 수 있는 최대 지연 값을 $d_{up-receive}(M)$이라고 나타낸다.

이와 같은 메시지 지연을 고려한 메시지 M을 위한 응용 프로그램간 지연의 상위 경계 $d_{app}(M)$은 이제 아래와 같이 쓰여질 수 있다.

$$d_{app}(M) = d_{up-send}(M) + d_{MAC-send}(M) + C_M + \tau + d_{MAC-receive}(M) + d_{up-receive}(M) \quad (1)$$

응용 프로그램간 지연의 구성요소들의 몇몇은 네트워크에 의해 결정되어지는 것이 아니고, 호스트에 의해 결정되어진다. 상위 계층 프로토콜 처리와 대기 행렬 삽입 지연 $d_{up-send}(M)$과 $d_{up\text{-}receive}(M)$은 주로 송신과 수신 호스트의 프로세스/메모리와 시스템 소프트웨어속도에 의존한다. 수신 노드의 MAC계층의 대기행렬 삽입지연 $d_{MAC-receive}(M)$은 어떻게 호스트 프로세스가 MAC계층으로부터 “메시지 도착” 인터럽트에 빨리 반응할 수 있는가에 달려 있다. 주어진 시스템에서, 이들 지연의 상위 경계는 종종 결정되어진다[3].

메시지 전송 지연 C_M은 채널의 대역폭에 의해 주어지고, 메시지 전달 지연 τ는 신호전달 속도 및 링크의 길이에 의해 주어진다. 이중에서 메시지 전달지연은 고정 값으로 취급되어질 수 있고, 메시지 전송 지연 C_M은 링크의 환경에 따라 변화될 수 있다. 즉, 동일한 MAC 프로토콜과 전송 매체를 사용한 경우에도 설치된 환경에 따라서 잡음등에 대한 영향을 고려해야 한다. 또한, 송신 노드의 MAC계층의 대기행렬 삽입 지연 $d_{MAC-send}(M)$은 사용되는 MAC프로토콜에 의해 결정된다. 적절히 MAC 프로토콜을 선택함에 따라 보다 지연에 민감한 메시지들은 보다 빨리 전송이 될 수 있는데, 이것은 토큰 링의 우선 순위 제어에서 볼 수 있다.

2) 메시지 오류에 의한 재전송에 의한 지연

원전용 통신망에서의 메시지 오류에 의한 재전송 지연을 계산하기 위하여서는 상용망 사용의 환경에 따른 문제점으로는 링크 에러에 의한 프레임 및 토큰 손실 문제를 고려해야 한다. 즉, 링크의 오류 이벤트로 인한 가능한 오류들로는 토큰의 손실과 프레임의 손실을 들 수 있는데 링 토폴로지의 토큰의 손실은 전송된 토큰에 오류가 있어서 그것을 토큰으로 인식하지 못함에 따라 발생한다. 이에 따라 링의 초기화와 토큰의 재생성이 필요하므로 이에 따른 네트워크 지연이 발생한

다. 또한, 프레임의 오류에 따른 재 전송지연은 실시간 요구 조건을 만족시키지 못하게 할 수도 있다. 따라서, 본 논문에서는 링크의 오류에 의한 메시지 전송지연을 살펴보고자 한다.

메시지 전송지연 C_M은 채널의 대역폭에 의해 결정되므로 링크 및 전송 에러에 의한 전송지연을 계산하기 위하여서는 전송에 사용된 채널의 유효 대역폭을 계산을 통하여 구할 수 있다. OSI의 2계층에 해당하는 데이터 링크 계층을 LAN 기술에서는 MAC과 LLC의 부분으로 구분하여 구현하고 있다. 이러한 구분에도 불구하고 데이터 링크 계층에서 이루어지는 주요한 기능의 하나인 오류 제어기능이 수행되도록 요구되고 있다.

원전용 통신망 환경에 따른 문제점으로는 링크 에러에 의한 프레임 및 토큰 손실 문제를 고려할 때는 재 전송을 가정하므로 전송 데이터 비트의 유효 비트율을 계산해야 한다. 즉, 유효 데이터율은

$$R_e = \frac{\text{수신측의 총 수신 데이터 비트수}}{\text{총 수신 데이터 비트의 수신 시간}} \tag{2}$$

으로 정의할 수 있다. R_e는 R이 갖는 값에서 재전송 등에 의하여 대역폭이 감소된 경우를 고려한 것이다. R_e를 계산하기 위하여 먼저 하나의 프레임이 제대로 수신될 확율을 계산하면, $j-1$개의 프레임에 오류가 있고 그 다음 프레임의 전송이 성공한다고 하면 총 j개의 프레임 전송이 있어야 한다. 따라서, 확율 P는

$$P[j \text{ 전송}] = P^{j-1}(1-P)$$

이고 기하학적인 확율 분포를 가질 때 성공적인 한 개의 프레임 전송을 위하여 필요한 평균 프레임 전송 개수 $\overline{N_T}$는

$$\overline{N_T} = \sum_{j=1}^{\infty} jP^{j-1}(1-P) = \frac{1}{1-P}$$

가 되며 확율 P는 일반적으로 $P = 1-(1-p_e)^K$로 주어진다. 이때 p_e는 링크의 비트 에러율이며 K는 프레임의 총 비트 수가 된다.

여기에 Stop-and-wait ARQ 방식이 사용된다면 n_h 오버헤드 비트, Acknowledgement 프레임(ACK 및 NACK) 비트 n_a, MAC 액세스 타임 t_{MAC} 및 전달지연 τ를 고려하면 하나의 데이터 블록을 보내고 ACK이나 NACK을 수신하기 위하여 소요되는 시간이 $[(F+n_a)/R]+2(\tau+t_{MAC})$이 된다. 따라서 평균적으로 프레임 전송 개수 $\overline{N_T}$가 제대로 수신이 이루어지기 전에 전송을 시도하므로 유효 데이터 율[1]은

$$R_e = \frac{F - n_h}{\frac{1}{1-P}\{\frac{F+n_a}{R} + 2(\tau + t_{MAC})\}} = \frac{(1-p_e)^F (K - n_h)R}{F + n_a + 2(\tau + t_{MAC})R}$$ 이 된다.

따라서 식 (1)에서 C_M 을 C_e = (트래픽 양)/R_e으로 대치시키면

$$d_{app}(M) = d_{up-send}(M) + d_{MAC-send}(M) + C_e + \tau + d_{MAC-receive}(M) + d_{up-receive}(M) \quad (3)$$

이 되어서 원전의 환경에 따른 메시지 지연을 계산할 수 있다.

여기서 MAC 계층의 제한 시간인 D_M은 메시지 M의 전송을 위한 제한시간을 정의하는 것이다. 만약 메시지 M이 제한시간 D_M을 가지고 t 시간에 송신노드의 MAC 계층에 도착한다면, 송신노드는 시간 $t + D_M$안에 메시지의 전송을 마쳐야만 한다. 메시지는 목적지까지 여러 개의 지연을 경험한다. 메시지 제한시간의 만족은 이들 지연과 직접적으로 관련되어 있다. 특별히, 메시지 M의 MAC 계층의 제한시간은 아래 수식과 같다면 만족되어 질 수 있다.

$$D_M \geq d_{MAC-send}(M) + C_M \quad (4)$$

여기에서 $d_{MAC-send}(M)$은 메시지 M이 송신노드에서 경험되어 질 수 있는 지연의 대기행렬 삽입 지연의 최대 값이다. 그리고, C_M은 메시지 M의 전송 시간이다. 이때, C_M은 주어진 네트워크의 상태에 따라 달라질 수 있다. 또한, 재전송에 의한 대기지연이 늘어날 경우에는 MAC 프로토콜에 따라 $d_{MAC-send}(M)$의 값도 달라진다. 이와 같은 이유들 때문에 실제의 시스템에서, 메시지의 시간 제한은 응용계층 제한시간에 의해 주어 질 수 있지만, D^M_{app}로 표현되는 메시지 M의 응용 계층 제한시간은 전송되어지는 응용 작업에 의해 메시지가 수신되기 전에 지나갈 수 있는 시간의 최대 값을 제공한다. 따라서 지연 범위의 사용은 메시지의 M의 MAC 계층의 제한시간을 다음과 같이 응용 계층의 제한시간으로 표현할 수 있다.

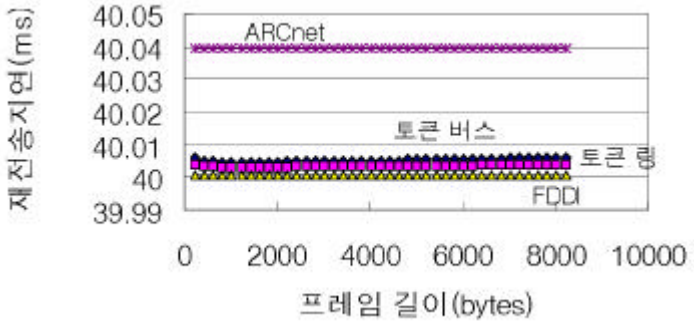
$$D_M = D^M_{app} - d_{up-send}(M) - d_{up\_receive}(M) \quad (5)$$

그래서, 수식(3)은 응용 계층의 제한시간을 MAC계층의 제한시간으로 바꿔준다. 각 프로토콜에 따른 D_M, R_e, D^M_{app}을 계산해 보면 다음과 같다. 계산에 사용된 파라미터는 <표 2>와 같다.

<표 2> 주요 프로토콜의 파라미터

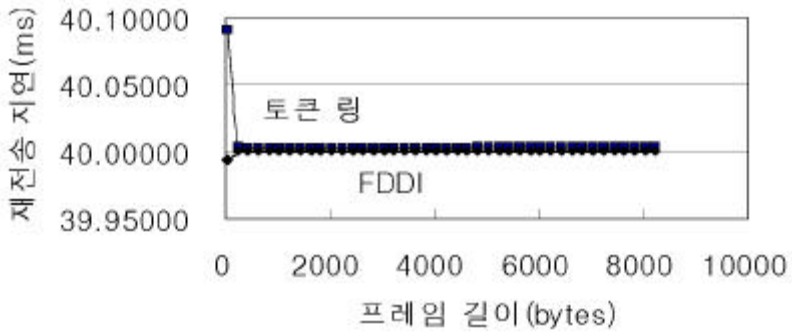
파라미터	주요 프로토콜			
	토큰 버스	토큰 링	FDDI	ARCnet
프레임 길이(K): bytes	26 - 8196	25 - 17967	32 - 4500	516
Aknowledgement 프레임 길이(n_a): bytes	26	25	32	1
오버헤드 (n_h): bytes	26	24	32	8
링크 전송율 (R): Mbits/sec	10	16	100	2.5
전달지연 (τ): μsec (100m)	0.5	0.5	0.5	0.5
MAC 액세스 타임 (t_{MAC}):50노드: μsec	5	3.125	0.5	20
링크 오류율 (p_e)	STP: 5.0×10^{-6}	STP: 5.0×10^{-6}	광섬유: 10^{-9}	STP: 5.0×10^{-6}

또한, $d_{up-send}(M)$, $d_{MAC-send}(M)$, $d_{MAC-receive}(M)$ 및 $d_{up-receive}(M)$은 하드웨어의 성능이나 프로토콜 스택 등에 의존하므로 일반적인 값을 계산할 수는 없으나, 네트워크 인터페이스 카드나 프로토콜 스택이 지정되면 구할 수 있다. 만일 이들 값이 각각 10 $msec$를 필요로 한다고 하면, 재전송에 의한 지연은 원전 통신망이 토큰 버스, 토큰 링, FDDI 및 ARCnet으로 설계된 경우에 대하여 재전송 지연은 <그림 4>와 같이 나타난다.



<그림 4> 원전 통신망에 적용되는 각 프로토콜의 재전송 지연 비교

만일 FDDI와 토큰 링이 사용되는 경우에는 프레임의 길이에 따라 실제 전송율이 변화하므로, 이에 따른 재전송 지연 역시 다르게 된다. 즉, <그림 5>에는 이 두 개의 프로토콜이 사용된 경우의 재전송 지연을 보이고 있다.



<그림 5> FDDI와 토큰 링 프로토콜의 프레임 길이와 재전송 지연

<그림 5>에서 보듯이 같은 프로토콜이라도 제전송 지연 시간이 프레임 길이에 따라 다르므로, 원자력 발전소의 통신에서 예상되는 최악의 지연을 고려하여 적절한 프레임 길이를 사용하는 것이 필요한데, 프레임 길이가 최소 200 바이트 이상은 되어야 할 것이다. 또한, 유효 전송율은 프레임의 크기를 어떻게 하는냐에 따라 크게 의존적이므로 각각의 프로토콜에서 최적 유효율을 갖도록 프레임의 크기를 정할 필요가 있다. 또한, 유효 전송율과 프레임 길이 및 링크 에러율은 서로 밀접한 관계가 있으므로 이것 역시 고려되어야 한다. <그림 5>에서 볼 때, 링크의 전송율이 작을수록 재전송 지연 값이 커지므로 보다 프레임의 길이를 최적화 하는 곳이 필요하다. 각 프로토콜에서 링크에 의한 에러 없이 전송 가능한 시간이 응용에 따라서 지정되어 있으므로, 프로토콜의 선정이나 네트워크의 설계 시에 이러한 조건하에서 응답조건이 만족되는 지를 점검하여야 한다.

4. 결론

원전 통신망에 적용 가능한 각 프로토콜을 대상으로 Stop-and-Wait ARQ 방식을 사용하여 에러에 따른 메시지 재전송이 있는 경우에 정확한 메시지를 받기까지 걸리는 지연 시간을 계산할 수 있도록 하였다. 주어진 조건에서 토큰버스, 토큰 링, FDDI 및 ARCnet 프로토콜이 보이는 재전송 지연시간은 정상 동작시의 약 0.1% 범위정도 증가함을 알 수 있다. 이에 비하여 프레임 길이에 따른 재전송 지연 시간은 프레임 길이가 짧을 때는 0.25%까지 증가 하는 것을 알 수 있다. 이런 정도의 재전송 지연은 충분히 메시지의 오류를 재전송 방식으로 제어할 수 있다는 것을 알 수 있게 한다. 그러나, 이런 지연 시간은 순전히 데이터 전송에 관련된 시간이므로 에러 신호의 처리나 재전송 순서에 따른 지연의 증가를 예상할 수 있다. 그러므로, 확율적인 방식으로 계산되는 재전송 지연보다 더 엄격한 메시지 재전송 경우를 고려한 에러 관리 구조가 필요된다.

참고 문헌

1. J. D. Spragins et. al., Telecommunications, Protocols and Design, Addison-Wesley Publishing Company, (1991).
2. A. S. Tanenbaum, Computer Networks, 3rd ed. Prentice Hall. (1996).
3. P.Pleinevaux and J.-D. Decotignie, "Time Critical Communication Networks: Field Buses," IEEE Network, Vol.2, No. 3, pp. 55-63, (1988).
4. K.-S. Song et. al., "Analysis of Several Digital Network Technologies for Hard Real-time Communications in Nuclear Power Plant," Journal of the Korean Nuclear Society, Vol 31. No. 2, pp. 226-235, April (1999).
5. N. Malcolm and W. Zhao, "Hard Real-time Communication in Multiple-Access Networks," Real-Time Systems, Vol. 8, pp. 35-77, (1995).
6. EPRI, "Integrated Instrumentation & Control System Upgrade Plan," EPRI NP-7343, (1992).