

MAP 통신망 접속기의 성능 분석  
(A Study on the Performance Analysis of the Network Interface for MAP)

임 용제, 김 덕우, 정 범진, 권 옥현  
(Yong Je Lim, Deok Woo Kim, Bum Jin Chung, Wook Hyun Kwon)

서울대학교 제어계측공학과 (Seoul National University,  
Dept. of Control and Instrumentation Engineering)

Modeling of the network interface for MAP and its performance analysis is investigated in this study. The parameters for the network interface are selected and a special interest is concentrated on the parameters related to the performance of the network interface itself. A queuing model of the network interface is proposed and simulation is performed to validate the proposed model of the network interface.

1. 서론

통신망에 대한 표준화 작업은 1977년 ISO(International Standard Organization)에서 계층 모형(layered model)을 제시 하면서 부터 본격화 되었다.[1][2][3] 계층내의 함수 및 기능등을 프로토콜(protocol)이라 하며 각 계층은 고유의 프로토콜을 함유 하고 있다. OSI 모형은 실제로 표준은 아니며 다만 표준을 제정하기 위한 근간(framework) 이다.

통신망 접속기는 표준에 입각한 하드웨어의 제작과 소프트웨어의 작성에 의해 구현된다. 하드웨어는 사용자 시스템(station)들을 망의 전송매체를 통해 물리적으로 연결하는 역할을 담당하며 소프트웨어는 호환성을 유지 하면서 통신망 프로토콜이 표준을 유지 하도록 총괄 운영하는 역할을 담당한다. 이러한 하드웨어와 소프트웨어의 결합체가 망 접속기이다. 망 접속기는 특정한 전송매체 사용(media access) 표준규약에 의해 전송매체로 데이터를 전송하거나 수신 하도록 하드웨어가 구성되며, 통신 표준에 따르는 소프트웨어를 접속기 내에서 실행하도록 설계된다. 본 연구에서 주로 다룰 MAP통신망은 토큰 버스라는 전송매체 사용 규약(IEEE 802.4)에 의해 OSI의 7 계층 모형중 계층 1,2가 구현된 망의 형태이다.[4][5]

본 연구의 주된 목적은 망 접속기의 체계적 설계 및 제작을 위한 자료제시에 있다. 이미 망 접속기의 모델링 및 해석에 대하여 연구된 바 있으나 내용이 너무 일반화 되어있거나, 실제 응용 보다는 연구 그 자체에 의미를 갖는

경우가 대부분 이므로 특정 망에 응용 하고자 하는 경우 크게 도움을 주지 못한다.[6][8]

본 연구는 크게 두가지 방향으로 행하여졌다. 우선 망 접속기의 모델링에 관한 연구이다. 토큰버스망 자체가 실시간 처리를 위한 망으로 가장 적합한 구조를 지니고 있으므로 그 목적에 맞도록 망접속기의 구조적 형태를 제시 하였다. 또 하나는 제시된 모형에 대한 성능해석이다. 성능 해석은 주로 제시된 모형이 과연 주어진 표준안에 맞게 동작할 수 있는지의 여부와 어느정도의 성능을 발휘할 수 있는가에 집중하였다. 성능해석을 위한 방법으로는 시뮬레이션과 해석적 방법을 이용하였다. 시뮬레이션을 통해 망 접속기 모형의 내부 변수가 변경될 경우 세부 모형의 영향을 추정하며 해석적 방법을 통해 망 접속기에 발생한 간섭(perturbation) 효과를 추정한다.

2. 토큰버스의 MAP통신망

(가) 토큰버스(Token Bus)

토큰버스는 물리적으로는 버스를 통해 전송매체에 연결되어 있으면서 논리적으로는 토큰링의 구조로 스테이션들이 연결된 형태이다. 즉, 하나의 버스를 통해 연결되어 있는 분산 시스템들은 논리적 링(logical ring)으로 순차연결되고, 전송 데이터 중 토큰이라는 특수한 형식을 설정하여 이러한 토큰을 소유한 시스템만이 버스의

사용권을 얻고 일정시간 버스를 사용한 뒤 논리적링에서 다음번 시스템으로 토큰을 넘겨주는 형태가 토큰버스인 것이다.

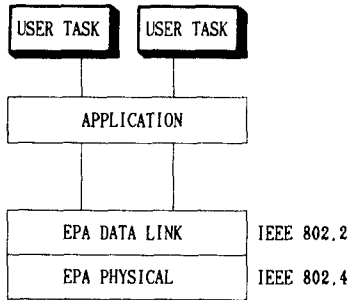
(나) IEEE 802.4

IEEE 802.4 표준은 토큰버스에 대한 다음의 사항들을 구체적으로 설명하고 있다.

- 전송매체의 전기적, 물리적 특성
- 전송매체상에서의 전기적 신호언법(signaling method)
- 전송 프레임(frame)의 형태와 특성
- 프레임을 수신한 스테이션(station)의 동작
- MAC(media access control)계층과 LLC(logical link control)계층간의 개념적 접속부(conceptual interface)에서 이루어지는 서비스 형태.

(다) MAP 통신망

본 연구에서의 시뮬레이션 대상이 mini-MAP 통신망 접속기이므로 이에 대해 간략히 알아보기로 한다. Mini-MAP은 공장환경에서 요구되는 실시간 처리를 위하여 제안된 특별한 구조이다. 이는 OSI 계층모형중 1,2,7 계층으로만 이루어진다. 그림 2.1은 Mini-MAP의 구조를 보여주고 있다. Mini-MAP의 물리적계층은 IEEE 802.4 를 따르고 있다.[7]



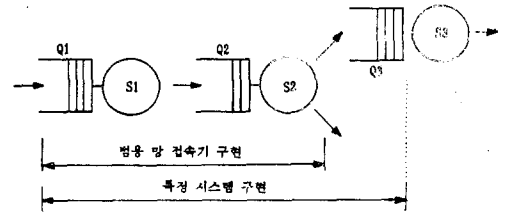
(그림 2.1) Mini-MAP

Mini-MAP의 데이터링크계층은 IEEE 802.2 규격을 따른다. 보통 type 1 서비스를 사용하나 고속처리가 필요한 경우 선택사양으로 IEEE 802.2 LLC type 3 서비스를 사용할 것을 제안하고 있다. Mini-MAP 데이터링크 계층이 따라야 할 조건은 다음과 같다.

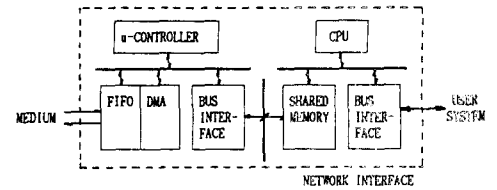
- 최대 SDU(service data unit)크기는 1 kbyte로 한다.
- Management를 위한 LSAP를 제공하여야 한다.
- 응용계층(application) 프로토콜로는 MMS(manufacturing message specification)가 사용되고 있다. MMS는 PLC, Robot, CNC 등에 이용할 수 있도록 만들어진 산업용 LAN을 위한 통신명령어(communication command lanaguage)이다.

3. MAP 통신망 접속기의 모델링

망 접속기를 모델링하기 위하여 큐-서버로 구성되는 큐잉모형을 적용하였다. 그림3.1은 모델링을 위한 개념적 큐잉 모형이다. 그림3.2는 이상의 큐잉모형을 구체적으로 표현해 본 형태이다.[8][9][10]



(그림 3.1) MAP 통신망의 큐잉모형



(그림 3.2) MAP 통신망의 구현형

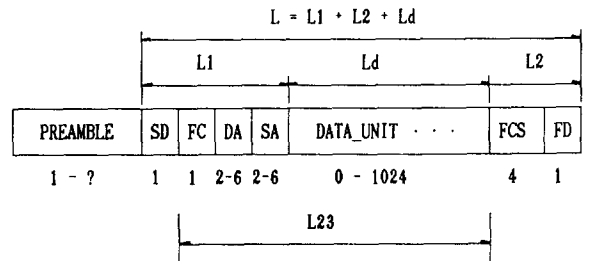
(가) 프레임의 형태

프레임은 전송매체상에서의 인식을 위한 요소(field)와 S2로 넘어가서 처리될 요소로 구분할 수 있다. 그림3.3에서 L1 및 L2가 전자에 속하며 L23가 후자에 속한다. Q1으로 입력되는 프레임의 크기는 최소크기 Lmin과 최대크기 Lmax를 갖는 지수분포로 가정한다.

$$L_{min} = L1 + L2 \quad (3.1)$$

$$L_{max} = L1 + L2 + L_{dm} \quad (3.2)$$

단,  $L_{dm} = \text{MAX}(L_d)$

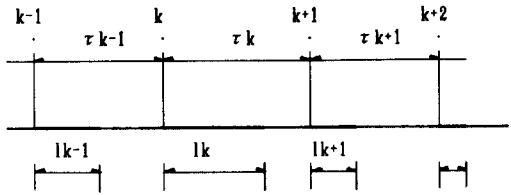


(그림 3.3) 프레임 구성요소

(나) 프레임의 도착 형태(arrivals)

그림3.4에서 프레임 도착간격시간  $\tau_k$ 를 poisson 분포로 가정하면 일반적으로 다음의 크기를 갖는다.

$$0 \leq \tau_k < \infty \quad (3.3)$$



k : k 번째 도착  
 $\tau_k$  : Time between arrivals [k, k+1]  
 $l_k$  : k 번째 프레임의 크기

(그림 3.4) 프레임 도착 형태

그러나 식(3.1), (3.2)에서와 같이 프레임의 길이에 대한 제한 및 토큰이 회전하는 시간이 존재하므로 도착 간격 역시 제한을 받는다. 이것을 고려하여  $\tau_k$ 는 다음의 식으로 표현가능하다.

$$\tau_{\max} = L_{\max} / \lambda_a \quad (3.4)$$

$$\tau_{\min} = L_{\min} / \lambda_a$$

$$\lambda_a = \text{arrival rate}$$

$$\tau_{\min} < \tau_k \leq \tau_{\max} \quad (3.5)$$

그림 3.4에서 보듯이 프레임이 전송되는 시간은 프레임 도착 간격보다 클 수 없다. 이는 토큰버스의 특성에 기인한다.  $\tau_k$ 의 크기에 따라 다음의 식과 같이 프레임 크기  $l_k$ 를 구할 수 있다.

$$\tau_{\max} \leq \tau_k \leq \tau_r \quad (\tau_r = \text{token rotation time}),$$

$$L_{\min} \leq l_k \leq L_{\max} \quad (3.6)$$

$$\tau_{\min} \leq \tau_k < \tau_{\max}, \quad (L_k = \lambda_a \tau_k)$$

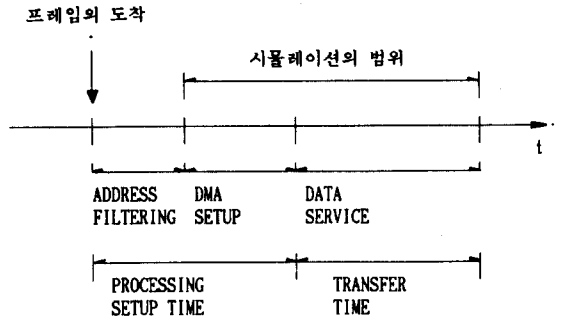
$$L_{\min} \leq l_k \leq L_k \quad (3.7)$$

다) 큐의 크기(queue length)

일반적으로 큐의 모델링은 그 크기가 무한대인 것으로 가정한다. 그러나 실제로 구현될 때는 무한 크기의 큐를 구현할 수 없으므로 한정된 크기의 큐로써 모델링하여야 한다. 간단한 수식에 의해 그 크기를 구하기도 하지만 실제 확률과정에서의 특성은 예측할 수 없으므로 시뮬레이션이나 실측을 통해서 큐의 크기를 정한다.

### 3.2 S1에서의 처리 형태

S1에서 주로 이루어지는 SERVICE는 토큰버스 프로토콜에 의한 전송매체 및 큐의 관리이다. S1에서의 처리형태를 시간축상에서 도식화하면 그림 3.5와 같이 표현할 수 있다.



(그림 3.5) S1의 처리형태

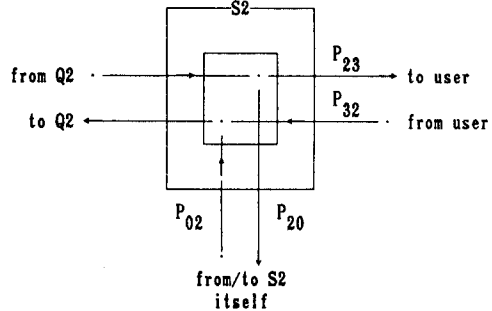
### 3.3 Q2의 형태

Q2를 모델링하는 방법으로는 여러가지 형태가 있을 수 있으나 그 중 가장 많이 이용되는 형태가 링크드 리스트(LL:linked list)구조이다. LL구조를 이용하면 링크의 간단한 변경에 의해 큐의 구조를 동적으로 변화시킬 수 있으므로, 많은 양의 버퍼(buffer)들을 큐로 이용하고자 하는 경우 유리하다.[11]

### 3.4 S2에서의 처리 형태

(가) S2의 처리

S2에서 행하여 지는 서비스 기능은 크게 둘로 나누어진다. S2내에서만 이루어지는 프로토콜 서비스와 외부 사용자 시스템과의 서비스로 구분된다. S2의 서비스 형태를 정리하면 그림 3.6과 같은 표현이 가능하다.



(그림 3.6) S2의 동작형태

Q2에서 S2로 전달된 데이터에 대한 S2의 서비스 종류에 따라 두가지 형태의 출력이 나타나며, 이들 출력은 각각 일정한 확률에 의한 배당(routing)이 이루어 진다고 가정할 수 있다.

(나) Q2에 대한 동작에 의한 S2 처리 형태의 구분

물리적으로 볼때 Q2는 S1과 S2에 의해 공유되는 영역이다.

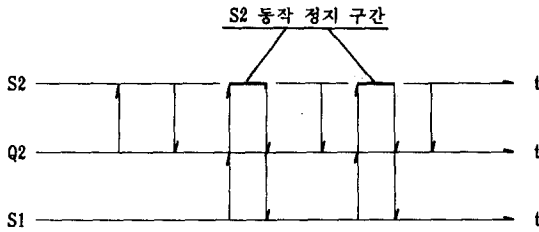
따라서 Q2는 S1과 S2 어느 한쪽에 의해서만 한 순간 사용이 가능하며 양쪽이 동시에 사용할 수는 없다.

(a) 버스 사용권을 이용한 동작

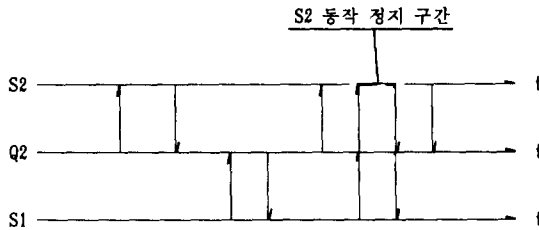
어느 한쪽이 버스사용권을 행사하게 되면 다른 한쪽은 모든 동작을 중단한다. 토큰버스의 특성을 만족하기 위해서는 S1이 사용우선권을 갖는다.

(b) 버스 cycle steal을 이용한 동작

상대방의 완전한(complete) 버스 사이클 사이에서 S1 또는 S2가 Q2를 이용하는 방법이다. S1이 우선권을 갖는 상태에서 S1이 Q2에 대하여 동작을 행하더라도 S2는 자신의 동작이 Q2와 관여하지 않는 상태라면 정지하지 않고 자신의 동작을 계속한다.



(그림3.7) 버스 사용권에 의한 동작



(그림 3.8) 버스 Cycle steal에 의한 동작

3.5 S2에서의 Perturbation

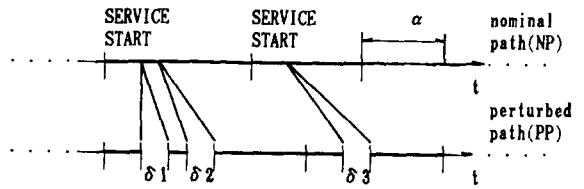
S2 처리시간을 간섭하는 주된 요인은 S1, S2가 Q2를 공유하는데서 발생한다. 간섭효과는 다음과 같다.

(가) 3.4-(나)절의 (a)형태

Q2에 대한 동작이 버스사용권에 의하여 진행되는 형태로 다음의 상황에서 S2 동작에 대한 간섭이 발생한다.[12][13]

- S2에서 이전 도착분 데이터를 처리하고 있을때. (FO:full output)
- 외부사용자 시스템(user system) 또는 self service에 의한 수신동작 중일때 (FO)
- 위의 두 상황에 대하여 S1에서 Q2로의 데이터 입력시간이 첨가됨
- Q2로의 입력이 없을때 (NI :no input)

이상의 상황에 의해 간섭이 처리시간에 첨가(realization) 되는 현상은 다음의 그림3.9와 같이 표현해 볼 수 있다.



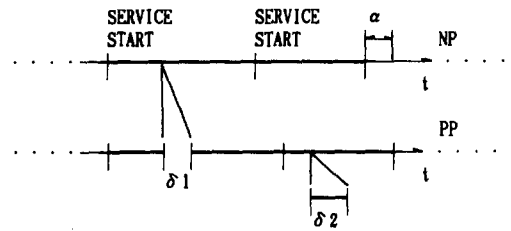
$$\alpha : \text{REALIZED PERTURBATION} = \delta 1 + \delta 2 + \delta 3$$

$$\delta k : \text{PERTURBATION}$$

(그림3.9) 간섭에 의한 처리시간변화(1)

(나) 3.4-(나)절의 (b) 형태

위의 (가)절과 동일한 간섭이 발생하나 각 상황에 대하여 S2가 Q2에 대한 동작을 행할때만 S1에 의해 Q2에 데이터가 입력되는 시간이 realization 되는점이 다르다. 이상의 상황에 의해 간섭이 처리시간에 첨가되는 현상은 다음의 그림 3.10과 같이 표현해 볼 수 있다.



$$\alpha : \text{REALIZED PERTURBATION} = \delta 1 (\delta 2 = \text{LOST})$$

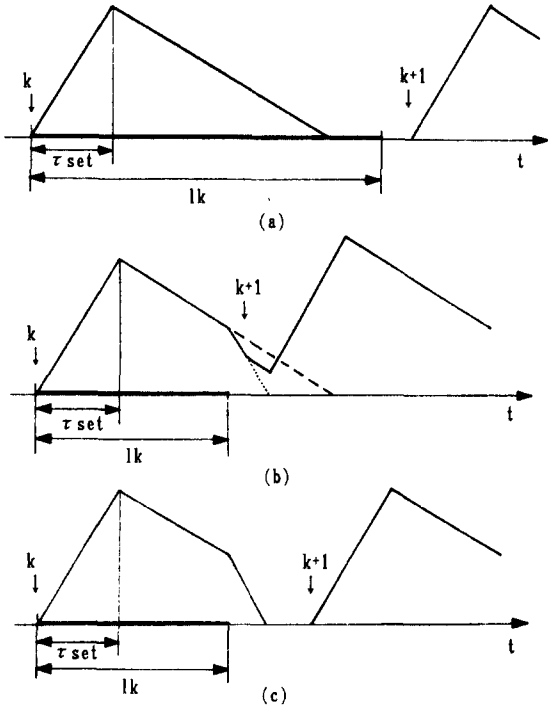
(그림3.10) 간섭에 의한 처리시간변화(2)

4. 해석적 고찰 및 시뮬레이션

4.1. 해석적 고찰

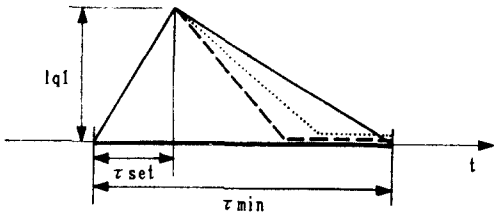
(가) Q1 및 S1에 대한 고찰

그림 4.1은 Q1에 도착되는 프레임과 이를 처리하기 위한 S1의 준비시간 및 S1의 처리시간에 따르는 Q1의 내용변화를 도시한 것이다. (a)그림의 경우는 S1의 처리시간이 충분히 빨라서, 주어진 처리 준비 시간 ( $\tau_{set}$ ) 동안 전송매체로부터 데이터를 저장할 수 있는 버퍼의 크기를 Q1이 갖는다면 데이터의 소실이 발생하지 않는 형태이다. (c) 그림의 경우는 비록 S1의 처리시간이 충분히 빠르지 않더라도 프레임이 도착되는 시간 간격이 크다면 역시 제한된 Q1의 크기라도 데이터의 소실이 발생하지 않는 형태이다.



(그림 4.1) 프레임 도착과 처리형태

그림 4.1 (a)에 대한 가정을 만족시키는 처리율(service rate:  $s_1$ )은 다음의 수식을 통하여 얻을 수 있다. 우선 식(3.4)에서  $\tau_{min}$  즉, 최소 프레임의 전송시간을 얻게 되므로  $\tau_{min}$ 을 넘지 않는 처리율을  $S_1$ 이 갖는다면 (a) 형태가 가능하다. 그림4.2에서 점선의 처리형태를 만족하는  $S_1$ 의 처리율을 구하는 것이다.



(그림 4.2)  $l_{q1}$ 이 최대인  $S_1$ 의 형태

3.1-(가)절에서 제시하였던  $l_{q1}$ 값은 다음과 같이 구한다.

$$l_{q1} = \tau_{set} / \lambda_a \quad (4.1)$$

일단  $l_{q1}$ 이 정하여지면  $S_1$ 의 처리율은 다음의 수식을 통하여 구할 수 있다.

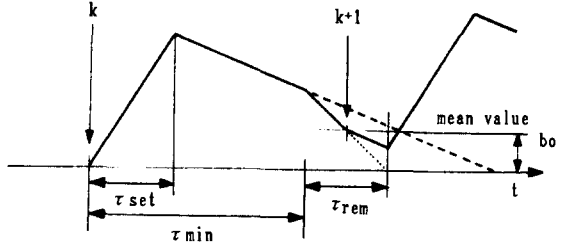
$$\tau_{min} = \tau_{set} + L_{min} / (\lambda_a - \lambda_{sa})$$

therefore,

$$\lambda_{s1} \geq \lambda_{sa} = \lambda_a - L_{min} / (\tau_{min} - \tau_{set})$$

$$\text{where, } l_{q1} \geq \tau_{set} / \lambda_a \quad (4.2)$$

그림4.1(b)에 대한 형태는  $S_1$ 의 처리율에 따라 확률과정에서의 최대크기가 변화하므로 확률적 개념으로 생각하여야 한다. 그림 4.3은 최악의 경우에  $l_{q1}$ 이 커지는 것을 보이고 있다.



(그림 4.3)  $l_{q1}$ 의 예측이 곤란한 경우

하나의 프레임이 완전히 도착하고 다음 프레임이 도착하는데 걸리는 시간은 다음과 같은 확률로 표현된다.

가변시간  $\tau_{rem}$ 의 평균값(mean)을  $\tau_{mean}$ 으로 놓으면 그때의 버퍼크기는 다음과 같이 표현 가능하다.

$$b_0 = \tau_{mean} / \lambda_{s1} \quad (4.3)$$

따라서 최악의 경우  $P_0$ 의 확률을 가진 프레임이 연속해서 한정된 크기의 큐에 놓일 수 있는 데이터의 갯수는 다음과 같이 구한다.

$$l = l_{q1} + nb_0 \quad (4.4)$$

$$n = (l - l_{q1}) / b_0$$

따라서 mean sense 입장에서 버퍼가 모두 사용될 확률은 다음과 같이 구한다.

$$\Pr\{\text{buffer full}\} = P_0^n$$

$Q_1$  및  $S_1$ 에 대하여 마지막으로 고찰할 것은 프레임의 도착 간격시간이 지수분포라는 가정에 대한 문제점이다.  $\tau_{min}$ ,  $\tau_{max}$ ,  $\tau_R$ 이 존재하는 상황에서 지수분포는 보통 다음의 확률표현에 의하여 그 값을 얻어낸다.[2]

$$P_r(x < t) = 1 - \exp(-t/T) \quad (4.5)$$

$t$ 에 따르는 확률값은 그림4.4와 같이 표현된다. 식(4.5)로부터 함수를 만족하기 위하여 시간축상에서 원래의 함수가 만큼 이동된 것으로 본다면  $P_r$ 은 다음과 같이 구한다.

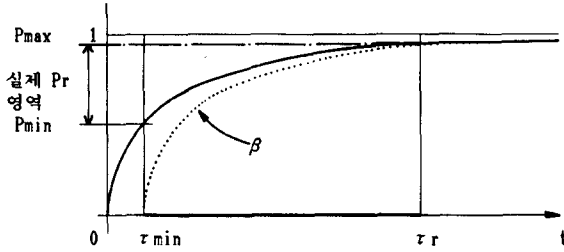
식 (4.5)로부터,

$$\ln(\exp(-(t - \tau_{min})/T)) = \ln(1 - P_r)$$

where,  $0 < P_r < P_{max}$

$$P_{max} = 1 - \exp(-(\tau_{rem} - \tau_{min})/T)$$

따라서,  $t = \tau_{min} - T(\ln(1 - P_r))$



(그림4.4) 지수분포모형

(나) Q2에 관한 고찰

이미 앞절에서 Q2를 LL(linked list)구조로 모델링 하였음을 언급한 바 있다. LL구조에는 구조유지를 위해 데이터와는 별도로 FD, BD, SB 등을 이용하고 있으므로 이들 버퍼와 설계된 데이터 버퍼사이의 관계를 해석하였다.

(다) 시뮬레이션 기준설정을 위해 고찰

(a) 토큰 회전 시간(token rotation time)의 해석 모형

토큰버스는 토큰의 회전에 의하여 전송매체의 사용권을 분할하므로 망에서의 토큰회전 시간에 대한 해석값을 설정하여야만 시뮬레이션이 가능하다. 이 값은 다음과 같이 구한다.[6]

토큰 회전 시간

$$= (\text{패킷 전송을 위해 모든 스테이션에서 사용된 시간}) + (\text{전송시 토큰을 넘기는데 걸린 시간})$$

$T_{\text{token}}$  (The average token passing time)

$$= (\text{the transmission time of the token packets}) + (\text{the average signal propagation delays incurred by the token in going from the token passing station to the destination station}) + (\text{the interface delay}) = (b_{\text{token}}/R_{\text{network}}) + (L_{\text{pass}}/V_{\text{prop}}) + T_{\text{interface}}$$

where,  $b_{\text{token}}$  ; number of token bits

$L_{\text{pass}}$  ; the average token passing distance

$T_{\text{trans}}$  = (the average elapsed time involved in the transmission of one packet)

+ (the average packet acknowledge delay)

$$= T_{\text{packet}} + T_{\text{ack}}$$

where,

$$T_{\text{packet}} = (\text{amount of time the message sending station spent in transmitting packet overhead bit and user bits}) + (\text{the average signal propagation delay})$$

$$= (b_{\text{control}} + b_{\text{data}}) / R_{\text{network}} + T_{\text{prop}}$$

$$T_{\text{ack}} = (\text{the acknowledgement packet transmission time})$$

+ (the average signal propagation delay)

$$= (t_{\text{back}} / R_{\text{network}}) + T_{\text{prop}}$$

(b) 평균 전송 지연시간 (average message delay)

평균 전송 지연 시간은 다음과 같이 구한다.

$$D_{\text{message}} = T_{\text{queue}} + T_{\text{prop}}$$

$$= W + T_{\text{prop}}$$

$$W = (1/2) \{ \langle NL(1-\sigma) / (1-\lambda_{\text{station}} NL - N\sigma) \rangle +$$

$$\langle N\lambda_{\text{station}} S / (1-\lambda_{\text{station}} NL - N\sigma) \rangle \}$$

where,

$L$  ; token passing delay

$S$  ; the average packet transmission time

$\lambda_{\text{station}}$  ; the mean arrival rate of packet to each station

$$\sigma = \lambda_{\text{station}} S$$

(c) 스테이션의 버퍼크기 (Q1)

Little's formula에 따르면 스테이션에서의 버퍼크기는 다음과 같이 구해진다.

$$N_{\text{buffer}} = T_{\text{queue}} \lambda_{\text{node}} b_{\text{data}}$$

$$= W \lambda_{\text{node}} b_{\text{data}}$$

(d) 각 스테이션에 대한 포화 전송율

$$\lambda_{\text{station}} = 1 / \{ N_{\text{station}} (T_{\text{station}} + T_{\text{token}}) \}$$

## 4.2 시뮬레이션 및 결과

(가) 기준(criteria)의 설정

이러한 Mini-MAP 통신망 접속기의 토큰버스 파라미터는 다음과 같다.[14]

- 16 octets data (typically 16-20 bytes)
- 32 nodes (stations)
- Total length = 1km
- 5 Mbits/sec medium signaling method
- Access time 25msec
- Maximum data size = 1024 bytes

이상의 기준과 4.1절의 해석적 고찰에서의 결과로부터 시뮬레이션에서의 고정 파라미터와 가변 파라미터를 구분하였다.[6][12][15][16]

a) 고정 파라미터

- Message transmission rates on medium = 5 Mbps
- Q1 size = 40 bytes FIFO
- Number of nodes the network supports = 32
- Average frame size = 34

b) 가변 파라미터

- S1의 처리율
- S1의 데이터 전송 준비시간
- S2의 처리율
- 토큰회전시간 (token-rot-time)

c) 구조적 파라미터 (structural parameter)

- Q2의 형태 (예 : LL 구조)
- Q2의 공유형태 (S1, S2에 의한)
- S2에서의 슬롯 할당 확률

(나) 시뮬레이션 결과

시뮬레이션을 통해 얻어진 각 파라미터 간의 영향으로부터 다음을 알 수 있다. S1의 처리속도는 빠를수록 좋다. S1의 처리속도가 해석적 고찰에서 제시하였던 최소 처리율 이상으로 빠른 경우 Q1의 크기가 결정론적으로 주어지므로, 확률 과정에서 데이터 소실을 방지할 수 있다. Q2 및 S2에 관련된 파라미터는 망 접속기 전반에 대해서 영향을 받는다. Q2의 크기와 구조적 형태는 토큰버스 고유의 파라미터(토큰 회전 시간, 패킷의 평균 크기 등) 및 S2의 처리속도에 밀접한 관계가 있다. S2의 처리형태는 multi-tasking 이 바람직하다. S2의 처리속도에 따라 S2가 수신에 관여하는 작업시간의 비중이 점점 커지며 S2가 거의 전체 동작 시간을 수신하는데 사용한다. 따라서 망 프로토콜을 전달하는 S2의 처리기능을 보아 어떤 처리 작업에 대해서도 공정한 multi-tasking 형태가 바람직 하다. S2의 처리시간 간섭요소는 S1과 S2의 Q2를 공유하는 형태에 따라 차이가 많다. 특히 토큰버스의 실시간 처리 특성을 감안 한다면 S1 및 S2는 버스 cycle steal 을 이용하는 형태가 바람직 함을 알 수 있다.

5. 결론

본 논문에서는 MAP 통신망 접속기의 성능 해석을 위하여 접속기를 모델링하고 그 모형에대한 해석을 행하였으며, 시뮬레이션을 통해 여러가지 성능 파라미터간의 영향을

파악하였다. 특정 사용자 시스템에대한 접속기를 구현하는 경우 사전에 망정보(전송 데이터의 평균 크기, 전송 빈도, 전송 비율 등) 및 접속기 내에서 동작하는 프로토콜 소프트웨어에 관한 정보(계층 수, 각 계층의 프로토콜 구현 정도, 그 크기, 처리시간 등)에 대한 정확한 인식이 필요함을 알 수 있었다.

<참고문헌>

- [1] Knowles, Larmouth and Knightson, Standard for Open Systems Interconnection, BSP, 1987.
- [2] Schwartz, Telecommunication Networks - Protocols, Modeling, and Analysis, Addison-Wesley, 1987.
- [3] "LAN 의 표준화 동향," 한국 전자 통신 연구소, 주간 연구 동향 88-07, pp.23-39, 1988.
- [4] ANSI/IEEE 802.2 Logical Link Control, IEEE, 1985.
- [5] ANSI/IEEE 802.4 Token-Passing Bus Accesss Method and Physical Layer Specifications, IEEE, 1985.
- [6] Dennis S. Mok, "A Software Tool for Performance Analysis and Evaluation of Local Area Network," Proc. 11th conference on Local Area Computer Netwroks, pp.77-88, Oct., 1986.
- [7] MAP Task Force, MAP v3.0, General Motors, May,1987.
- [8] J. W. Wong and Werner Bux, "Analytic Modeling of an Adapter to Local Area Networks," IEEE Trans. on Comm., vol.32, pp.82-89, 1988.
- [9] H.Kobayashi, Modeling and Analysis, Addison-Wesley, 1987.
- [10] Kleinrock, Queueing Systems vol.I: Theory, Wiley, 1985.
- [11] Horowitz and Shani, Fundamentals of Data Structure, Computer Science, 1983.
- [12] Y. C. Ho and X. Cao, "A New Approach to the Anaysis of Discrete Event Dynamic Systems," Automatica, vol.19, pp.149-167, 1983.
- [13] Y. C. Ho and X. Cao, "Perturbation Analysis and Optimization of Queueing Networks," JOTA, vol.40, pp.559-582, Aug., 1983.
- [14] M. H. MacDougall, Simulating Computer Systems Techniques and Tools, MIT Press, 1987.
- [15] D. W. Jacobson, "High Performance Reliable Token Bus for the MAP Network Architecture," Proc. 11th Conference on Local Area Computer Networks, pp.26-33, Oct., 1986.
- [16] Madhav V. Marathe and Robin A. Smith, "Performance of a MAP Network Adapter," IEEE Network, pp.82-89, 1988.

알림 : 본 연구는 한국 과학 재단의 연구비 지원에 의 해 수행되었음 .