

## 버스 공유 다중마스터 시스템의 모델링과 설계

\*홍재명, 신준호, 김용득

아주 대학교 전자공학과

경기도 수원시 팔달구 원천동 아주 대학교 컴퓨터 네트워크 연구실

iam-hjm@comnet.ajou.ac.kr

### The Design and Modeling of Shared bus Multimaster System

\*Jae-Myoung Hong, Jun-Ho Shin, Yong-Deak Kim

Dept. of Electronics Engineering Ajou University

Computer Network Lab. Ajou Univ. Woncheon Dong Paldal Gu Suwon KOREA

iam-hjm@comnet.ajou.ac.kr

#### 요 약

본 논문은 다중마스터 시스템의 버스 공유를 큐잉이론으로 모델링하여 정량화하고 이를 바탕으로 최적화된 실시간 시스템을 설계하는 방법을 제안하였으며 그 결과를 실측 실험을 통해서 검증하였다.

다중마스터 모드와 슬레이브 모드를 지원하는 마스터를 이용한 다중 마스터 시스템에서 버스의 공유로 인한 지연과 각 작업의 대기 시간은 각각에 대한 모델링을 통해 정량화 할 수 있으며 이를 통하여 최적화된 시스템을 구성할 수 있게 된다.

본 논문의 실험에서는 VMEbus 상에서 3개의 마스터와 그에 종속된 4개의 슬레이브 시스템을 구성하여 각 마스터들의 버스 요구율과 서비스 시간에 따라 버스를 점유하기 위해 기다리는 시간을 정량적으로 분석하였으며 이를 통하여 개선된 시스템은 각 작업의 버스 상에서의 대기 시간을 최소화하고 효과적으로 버스를 공유함으로써 작업 시간 오차와 오류 발생을 최소화 할 수 있음을 보였다.

#### 1. 서론

컴퓨터 시스템의 규모가 커지면서 기능 확장과 호환의 편의를 위해 버스 시스템 방식의 적용이 확대되었다. 개인용 컴퓨터의 경우는 하나의 마스터에 의한 버스 점유가 일반적이나 특정 목적의 시스템들은 복수개의 마스터들이 버스를 공유하여 각각의 작업을 수행하여야 한다. 대량 작업을 요구하지 않는 시스템들은 다

중 시스템을 구성해도 크게 문제가 되지 않았거나 단순히 충분한 시간적인 지연을 사용하였다. 그러나 시스템의 대형화와 복잡화로 인해 다중마스터 시스템 구성 시에는 예상하지 못했던 오류가 발생하게 된다. 버스 방식으로 다중마스터 시스템을 구성할 때, 각 마스터는 프로세서와 자신의 메모리를 가지고 독립적이면서 상호 연관적인 작업을 하게된다. 각 작업들이 시간적인 제한 조건이 주어지고 상호 연관되어 동작할 때, 버스를 공유함으로써 발생하는 작업순서, 속도 그리고 대기 시간등이 문제가 된다. VMEbus 상에서 버스마스터 역할을 하는 컴퓨터의 수가 증가함에 따라 각 마스터 컴퓨터의 작업량과 작업의 빈도에 따라 대기 시간이 길어지며 실시간 시스템을 구축하기 위해서는 작업의 오차를 미리 예측할 수 있는 데이터가 필요하게 된다. 이런 이유로 예측 가능한 데이터를 얻기 위하여 버스 시스템을 모델링하고 버스 공유를 정량화 시키게 된다.

다중마스터 시스템은 복수개의 마스터 컴퓨터와 이에 종속된 시스템들로 구성된다. 마스터 시스템들은 고유의 종속 시스템을 가지고 독립된 작업을 수행하며, 버스 제어마스터는 각 작업의 시작 요구와 결과를 이용하여 최종 제어 작업을 하며 연속적으로 각 마스터에게 결과를 요구하고 각 마스터는 버스를 통해 접속된 아날로그/디지털 신호 처리 보드와 데이터 입출력 보드등을 제어하여 발생하는 결과를 연속적으로 버스제어용 마스터에게 전송하게 된다. 각 마스터에서 입출력되는 데이터들의 크기와 빈도수, 그리고 버스 제어용 마스터에서의 결과 요청의 빈도수에 따라 각 시스

템이 버스를 점유하기 위한 대기시간이 증가하게 된다. 이때의 버스 점유를 큐잉 이론<sup>[1]</sup>으로 모델링하고 그 결과를 정량적으로 비교하여 실시간 시스템을 구축하는데 필요한 자료를 제공하고 이를 바탕으로 실시간 다중마스터 시스템을 설계하고<sup>[2]</sup> 실측 실험을 통하여 제안 방법의 타당성을 검증하였다.

## 2. 다중마스터 시스템의 설계

VMEbus 용 마스터 보드에 다중마스터 시스템 지원 회로를 추가 설계하며 다중마스터 동작을 위한 기능과 슬레이브 모드를 지원하도록 설계한다.

### 2.1 마스터 모드

버스를 관할 할 수 있는 권한을 가지게 되면 자신에 종속된 슬레이브(입출력장치모듈)를 제어 사용할 수 있게된다. 시스템 구성과 설정시 시스템 데이터 처리, 세마포어 레지스터의 설정, 메모리 공유의 설정 등의 요인에 주의 해야한다.

시스템의 데이터 처리방식은 3U와 6U 의 두가지 방식이 있다. VMEbus의 backplane을 보면 96핀 슬롯이 1개씩 또는 2개씩 쌍을 이루어 있는 것을 볼 수 있으며 하나씩 있는 것을 3U, 쌍으로 이루어져 있는 것이 6U이다. 6U는 3U의 데이터와 어드레스 버스와 크기를 확장한 것으로 16bit 데이터와 24비트 어드레스 크기를 32비트 데이터와 32 비트 어드레스 버스로 확장해준다. 개발된 마스터 보드는 6U용이며 MC68030의 32비트 데이터, 어드레스 버스를 효과적으로 처리해준다. VMEbus 액세스 시에 데이터 크기를 결정하는 신호인 Word 신호를 '1' 로 설정해주어야 하며 이 신호는 리셋시에 '0' 으로 초기화되어 있게된다. 또한 시스템에는 2개의 세마포어 레지스터와 인터럽트 레지스터들이 존재하며 인터럽트 순위 1과 7을 사용한다. 세마포어 인터럽트<sup>[6]</sup>를 사용하기 전에 프로그래머는 VIC068의 레지스터를 설정해주고 시작해야 한다. 개발되어진 마스터보드는 메모리 공유를 지원하도록 설계되었으며 기본 개발된 마스터보드와 다른 메모리 맵을 가진 시스템과의 상호 메모리 공유는 슬레이브 제한 어드레스만 소프트웨어적으로 설정해주고 바로 사용가능 하다. 그러나 같은 메모리 맵을 가지는 마스터보드사이에서는 메모리 공유를 위해서 가상적으로 메모리 맵을 달리해서 슬레이브의 메모리를 사용할 수 있도록 어드레스 속이기가 필요하게 된다.

### 2.2 슬레이브 모드

외부의 요구에 의해서 마스터가 슬레이브<sup>[4]</sup>로서 동작하도록 하는 기능이다 마스터의 중앙처리장치가 내부

버스를 외부에 허락하고 슬레이브 모드로 동작하게 된다. 이 모드에서는 내부 DRAM을 VMEbus에 의해서 슬레이브 모드를 이용하여 접근할 수 있다. 이것은 슬레이브 모드로 접근 가능한 어드레스의 상위 8파트 어드레스를 레지스터에 저장하는 것이며 이렇게 저장된 어드레스는 외부의 다른 마스터에 접근해 왔을 때 그 어드레스를 비교하여 사용 여부를 결정하게 된다. 이러한 슬레이브 모드를 위하여 우선 허가 영역을 정의하며 (/CSEXA32) 이것은 슬레이브 동작을 위한 VMEbus 어드레스 비교 레지스터에 그 비교될 어드레스를 기억시킬 때 사용된다. 이처럼 슬레이브 시작 어드레스 레지스터에 정보를 쓸 때 /SLSEL[0..1] 신호의 값을 결정할 수 있으며 이 신호는 VMEbus에서 액세스되는 어드레스가 내부에서 세팅되어 있는 슬레이브 시작 어드레스와 동일할 경우 /SLSEL 이 활성화되게 된다. 이 모드에서는 레지스터를 상호 공유 할 수도 있으며 이 레지스터들은 VIC068의 내부에 8개가 존재하며 이중 5개를 서로 공유하여 자유롭게 쓸 수 있다. 이처럼 서로가 레지스터를 공유할 경우에는 VIC068을 가진 측을 슬레이브로 보는 것이 타당하다. 다중마스터를 지원하기 위하여 버스조정 제어, 슬레이브 모드 제어, VIC068 제어, 세마포어 레지스터 처리 등은 EPLD를 통하여 효과적으로 구성 할 수 있다.

## 3. 버스 공유의 큐잉 모델링 제안

### 3.1 VMEbus 모델

버스 시스템은 데이터의 송수신을 위하여러가지 인자들이 연결된 집합체로 볼 수 있으며 버스시스템의 특징은 한번에 하나의 작업만이 버스를 점유하여 사용할 수 있다는 것이며 이는 각 버스의 사용주기마다 하나의 마스터가 오직 하나의 슬레이브만 처리함을 의미한다. 이런 이유로 인하여 디지털 보스를 다기능 모듈 중에서 시공유 버스라고 하며 다른 동적 연결 네트워크에 비해서 대역폭이 제한된다.<sup>[8][9]</sup> VMEbus도 역시 시공유 버스에 속하며 이를 위한 모델링을 위하여 큐잉 모델을 사용하며 큐잉 모델<sup>[5]</sup>에서 서버이용률, 대기 시간등을 계산해내기 위해서는 프로세스가 버스시스템에 도착하는 시간의 분포와 버스 서버에서의 처리 시간에 대한 정보가 있어야한다. 각 마스터들은 운영체제에서 처리되는 내부 큐를 가지며 VMEbus를 요구하는 작업에 대해서만 개념적으로 독립된 큐와 서버를 구성할 수 있다. 버스중재는 전체 시스템을 관할하는 시스템에서 하게되며 버스 서버에 의한 서비스 시간은 중재시간과 슬레이브와 마스터 사이에서 데이터를 읽거나 쓰는 작업시간을 모두 포함하게된다.

VMEbus 모델은 각 마스터에서 처리되는 부분과는 독립적으로 버스 상에서 처리되는 부분만 큐잉모델에 적용한다. 그림 1에서 보여지는 마스터의 큐와 서버는 마스터 내부에서 버스처리에 관련되는 부분만 독립시킨 모델이며, 버스의 큐와 서버는 마스터에서 처리되어 최종 VMEbus를 요구한 작업에 대한 큐와 버스 중재 역할을 하는 서버로 구성된 모델이다. 그림에서 n개의 마스터로 나타낸 것처럼 마스터는 21개 이내에서 가변적이며 슬레이브의 수도 n에서 마스터의 수를 뺀 나머지에서 가변적이다. 슬레이브들은 프로세스 큐를 가지지 않으며 슬레이브에서 처리되는 시간은 버스서버의 서비스 시간의 일부로 합쳐진다. 슬레이브의 결과가 마스터에 영향을 미치는 경우도 마스터가 버스를 관할하여 데이터를 주고받으므로 전체흐름은 일정하게 되며 이를 위하여 전체 시스템은 개방형 모델이 적당하다.

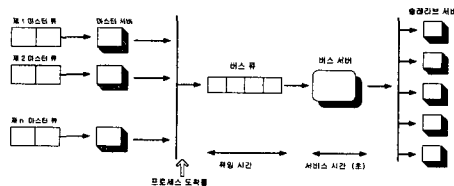


그림 1. 다중마스터 시스템의 큐잉 모델

### 3.2 다중마스터 모델링

큐잉 이론은 해석적 모델링에 주로 사용되며 시스템의 상태나 성능을 수학적으로 계산할 수 있고, 시스템이나 네트워크에서 공유되는 한정된 서버에 의해서 발생하는 지연시간에 따른 효율과 비용적인 문제에 대한 모델화<sup>[6]</sup>를 통하여 수학적으로 분석하고 보다 효율적인 시스템을 구성할 수 있도록 해준다.<sup>[7]</sup>

VMEbus 기반의 다중마스터 시스템의 모델을 Kendall's notation에 의해서 표현하면 아래와 같다.

$$A/B/c/k/m/z = M/M/1/\infty/n/FCFS$$

A는 버스 요구율, B는 요청된 작업의 서비스율, c는 버스 서버의 수이고, 큐용량 k는 마스터 21개내에서 처리 가능하므로 무한대로 표현하고, 고객 m은 21중 하나의 수인 n으로 표시하며 z는 작업처리 우선순위를 의미하므로 기본형태이면서 시스템 구성상 예측이 가능한 FCFS를 따른다.

실시간 다중마스터 시스템을 설계하는 중요한 데이터가 버스에서의 대기시간 ( $W_q$ )이며 큐잉 모델에서 큐잉시간이 바로 여기에 해당한다. 이 값을 계산하기 위하여 프로세스의 도착 비율과 서비스 비율에 대한 정보가 필요하며 이를 위하여 전체 버스에서의 프로세스 평균 도착율( $\lambda_v$ ), 슬레이브에 의한 평균 버스 서비스 처리시간 ( $B_v$ : 버스요구에 따른 버스서버에 의한 용

답처리,  $Stnx$ : 슬레이브에서의 처리시간,  $Q_t$ : 지연시간)을 통해 계산되는 개별 마스터에 대한 서비스 시간을 구한 후 이를 통해 전체 버스에대한 평균 서비스 시간을 계산한다. 이 값과 서버 이용률 ( $\rho_v$ ), 트래픽 밀도(u)를 이용하여 현재 버스가 사용중일 때 새로 버스를 요구할 경우 기다려야 할 확률  $C(c,u)$ 를 구한다.

$$C(c, u) = \frac{\frac{u^c}{c!}}{\frac{u^c}{c!} + (1 - \rho_v) \sum \frac{u^n}{n!}}$$

위에서 열거한 변수들과 각 계산 값을 이용하여 큐잉 시간을 계산한다. 큐잉 시간은 다음의 식으로 계산되어진다. 아래의 식에서  $E(r)_v$ 는 버스요구작업의 도착 시간 간격으로  $\lambda_v$ 에 의해서 계산된다.

$$W_q = \frac{c(c, u) \cdot E(s)_v}{c \cdot (1 - \rho_v)}$$

$$W_q = \frac{E(s)_v^2}{E(r)_v - E(s)_v}$$

## 4 실험 및 분석

전체실험 환경은 VMEbus rack에 다수의 보드가 장착되어 있고 오실로스코프, 논리 분석기, MC68030용 MDS와 PC로 구성된다. 마스터 보드는 실시간 운영체제가 이식되어 콘솔용 PC로 데이터를 출력하게 되어 있다. 개발된 마스터 보드는 MC68030 주 프로세서와 DRAM과 SRAM, 기본 입출력장치를 가지고 EPLD 기술을 이용하여, VMEbus 접속을 위해 전용 IC인 VIC068로 구성되어 있다. 모델링을 위해 사용된 데이터 측정과 실험결과는 윈도우를 사용하지 않는 기본 OS-9 상에서 이루어졌으며, 추가적으로 슬레이브 장치를 윈도우 상에서 제어하는 실험이 이루어졌다. 실험용 시스템은 마스터 3개와 슬레이브 4개로 구성되며 각 마스터는 자신에 종속된 슬레이브와 짝을 이루며 단, 제 2 마스터는 2개의 종속 슬레이브를 포함하여 구성된다. 실험에서 사용된 슬레이브들은 모두 790nsec 대의 비슷한 처리 속도를 가지며 평균 서비스 시간은 910nsec로 계산되었다. 마스터와 종속 슬레이브로 구성된 독립적인 VMEbus 시스템들이 서로 다른 버스 요구율을 가지고 하루 평균작업시간인 8시간을 기준으로 전체 사이클 수를 계산하고 각각의 독립되었던 시스템을 혼합하여 VMEbus 다중마스터 시스템을 구성한다.

4.1 결과 분석

다중마스터의 결과는 큐잉모델에 의해 계산된 표 1에서 보듯이 독립적으로 측정 계산된 결과에서 어느 정도 손실을 가져오게 된다.

표 1. 다중마스터 시스템의 손실

작업종류	마스터		
	제 1테스크	제 2테스크	제 3테스크
독립된 8시간작업 Cycle (*10 <sup>9</sup> )	6.063	2.618	1.440
다중마스터일 때 손실 Cycle(*10 <sup>6</sup> )	500.0	97.9	30.1
손실률(%)	8.2	3.7	2.1
다중마스터 경우 초과시간	43분11초	18분 39초	10분 15초

위의 결과를 이용하여 시스템의 동작을 예측하면 사용되는 슬레이브의 응답시간이 느려질수록 평균 서비스 시간은 길어지게 된다. 표 1에서 평균서비스시간은 실시간 스케줄링에 큰 장애가 되는 블록 전송을 배제하고 버스 상에서 한 사이클의 데이터 처리 시간이다. VMEbus 특성상 다중작업을 위한 복수개의 버스서버의 존재는 불가능하며 큐잉모델에 의해 예상된 결과를 바탕으로 효율적인 버스 스케줄링이 필요하다.

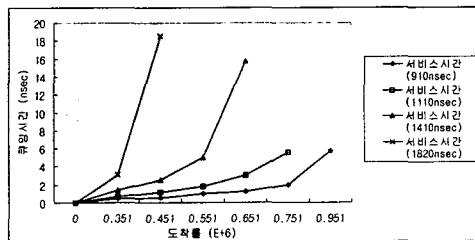


그림 2. 도착율과 평균 서비스 시간에 따른 큐잉 시간

그림 2는 큐잉 시간을 증가시키는 근본적인 두가지 조건, 평균 서비스 시간과 도착율 따른 큐잉시간의 결과이다. 실측 실험을 통하여 얻어진 데이터에서 보면 평균 서비스 시간과 도착율이 증가함에 따라 서버 이용률이 1에 근접함을 알 수 있다. 그러므로 도착율이 높아져서 서버 이용률이 1에 가까워질수록 사이클 손실률의 증가폭이 매우 커진다. 이러한 문제점을 개선

하기 위하여 얻어진 자료를 바탕으로 하여 원하는 대기 시간에 따른 도착율이나 서비스 시간을 조절하면 된다. 상황에 따라 다르겠지만 주로 서비스 시간은 슬레이브 보드에 의한 것이므로 변경이 용이하지 않다. 그러므로 도착율을 조절하게 되는게 각 마스터에서 자신의 VMEbus 요구 테스크의 상대적 중요성에 따라 그 요구율을 정해주게 된다. 이렇게 결정된 시스템은 예측된 수준의 손실을 가지며 다중마스터의 동작을 원활히 수행하게 된다.

5 결론

실시간의 시스템을 구현하는 과정에서는 전용화된 시스템들을 복수 개 사용할 때 버스를 공유함으로써 발생하는 대기 시간을 계산, 예측하여 최적화된 실시간 다중마스터 시스템을 설계해야 한다. 이를 위해서 VMEbus 기반의 실시간 다중마스터 시스템을 큐잉 모델링하여 평균서비스 시간과 버스 요구율의 변화에 따른 버스대기 시간등 시스템 성과 관련된 데이터들을 계산하여 정량적으로 분석하였다. 다중마스터 구성시 예상치 못했던 2.1%에서 8.2%의 테스크 손실률을 정량화 된 버스 공유 데이터를 바탕으로 재설계하여 손실률을 1%이하로 줄이게 되었다.

6 참고문헌

- [1] 조유근, 고건, "운영체제론", 홍릉과학출판사, 1990
- [2] Larry Carlson, "VMEbus/VXibus", Hewlett Packard
- [3] Databook, "VIC068A User's Guide", Cprress Semiconductor, 1992.
- [4] Wade. D. Peterson, " The VMEbus Handbook 2nd", A VITA Publication, 1991.
- [5] Lester R. Lipaky, "Queueing Theory", Macmillan, 1992.
- [6] Mischa Schwartz, "Tellecommunication Networks", Addison-weslet, 1987
- [7] Peter G .Harrison & Naresh M.Pael, "Performance Modelling of COmmunication Networks and Computer Architecture", International Computer Science Series, 1992
- [8] Kai Hwaing, "Advanced Computer Architecture", McGraw-hill. 1993
- [9] John Hennessy. "computer Architecture A Quantitative Approach", Morgan Kaufmann Publishers. 1990