CHAPTER III

QUEUEING MODEL FOR THE MULTIPLE-MICROPROCESSOR SYSTEMS WITH SHARED MEMORY

Performance model of the shared-memory multiple-microprocessor systems is presented in this chapter. Most often, we are interested in two types of measurments of system performance : response time and throughput. Response time refers to the interval between the request for the performance of a unit of work and the subsequence completion of the work. Throughput refers to the amount of work that can be accomplished per unit time. Typical measurements include operation per second, jobs per second, and transactions per second.

The proposed model is based on the flow equivalence technique [17]. The analysis treats each microprocessor as a server which takes its input from a given queue, and consequently the problems of hardware contention for bus and memory are not considered in detail. This technique is appropriated for the multiple-microprocessor systems in which the tasks are not dynamically reconfigurable as found in most real-time applications.

In particular this chapter presents the queueing model that can be used for calculating the system response time of the multiple-microprocessor systems with arbitrary number of microprocessors and unequal processing rate or the unsymmetrical multiple-microprocessor system. The multiple-microprocessor system response time will be used as a parameter to characterize system performance. Two multiple-microprocessor systems will be considered i.e. the unsymmetrical multiple-microprocessor system with arbitrary number of equal priority microprocessors and the

multiple-microprocessor system with processor priority. In addition to the development of the theoretical queueing model, experiment has been conducted with the proposed shared-memory multiple-microprocessor systems and the result is compared with the theoretical prediction in order to validate the model.

3.1 The Unsymmetrical Multiple-Microprocessor System [26].

The architecture to be considered in this section is shown in Fig. 3.1. The multiple-microprocessor system being considered consists of an arbritrary number of microprocessors sharing the same resources. The shared resources are shared memory, input unit, and output unit. The system is unsymmetrical : all the microprocessors possess their own processing rate and are connected to the shared memory via a single common bus. Each microprocessor also has its local memory for program storage. An external data is sent into the system in a unit called transaction, through the input unit. The input transaction is scheduled to be processed by any available microprocessor upon arrival. When all microprocessors are busy, incoming transaction enters a common waiting line in the shared memory. If several microprocessors are available to an arriving transaction, the transaction is scheduled to the idle microprocessor at random. The processed transaction is then sent to the external device through the output unit. This unsymmetrical multiple-microprocessor system becomes the symmetrical system if all microprocessors have the same processing capability.

3.1.1 System Model.

The following assumptions are made for the queueing model representing the multiple-microprocessor system.

1. There are c microprocessors in the system. The data transaction processing time for each microprocessor is a random

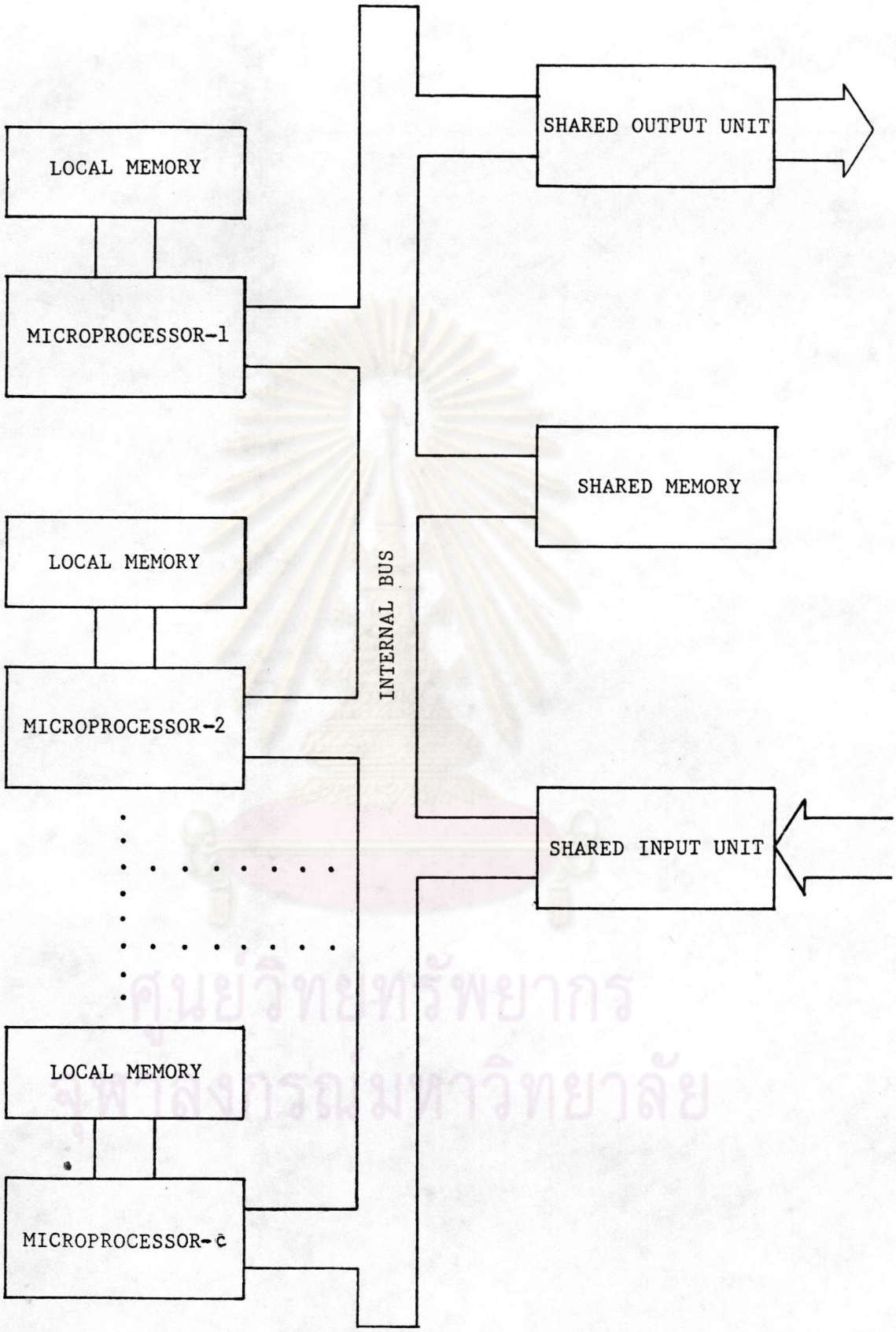

Fig 3.1 The unsymmetrical multiple-microprocessor system block diagram.

variable with exponential distribution. The mean processing rate of microprocessor n is μ_n , n = 1,2,3,.....,c.

2. The external data transactions arrive randomly to the input unit. The transaction arrival rate is a random variable observing Poisson distribution with a mean of λ .

3. The response times of the shared memory, the input unit, the output unit, and the internal bus are very short compared with the processing time of the microprocessors.

4. The shared memory capacity is very high compared with the amount of the external input data.

5. The possibility that more than one microprocessor finishes their tasks simultaneously is so small that it can be neglected.

Hence the multiple-microprocessor system can be analyzed by the queueing model as shown in Fig.3.2. The queueing model consists of c unidentical servers servicing to a single common queue. The model becomes M/M/c queueing model with unequal service rate. Only a solution for the identical servers model are currently available [27,28]. In the next section, we will present the solution for M/M/c queueing model with unequal service rate.

3.1.2 Analysis of the Model.

The queueing model for the multiple-microprocessor system consists of c servers with mean service rate of μ_n , n = 1,2...,c shared a common single queue. Consider the mean service rate of this system, if there are more than c transactions in the system, all the c servers are busy and each is putting out at a mean of μ_n , n = 1,2..,c and the mean system output rate is thus $\mu_1 + \mu_2 + \ldots + \mu_c$. When there are fewer than c transactions in the system, n < c, only n of c servers are busy and the system is processing at a mean rate of $nc/\sum_{n=1}^{c}(1/\mu_n)$. Thus the state of the system can be defined by a number of transactions in the system (n) and the state diagram of the system

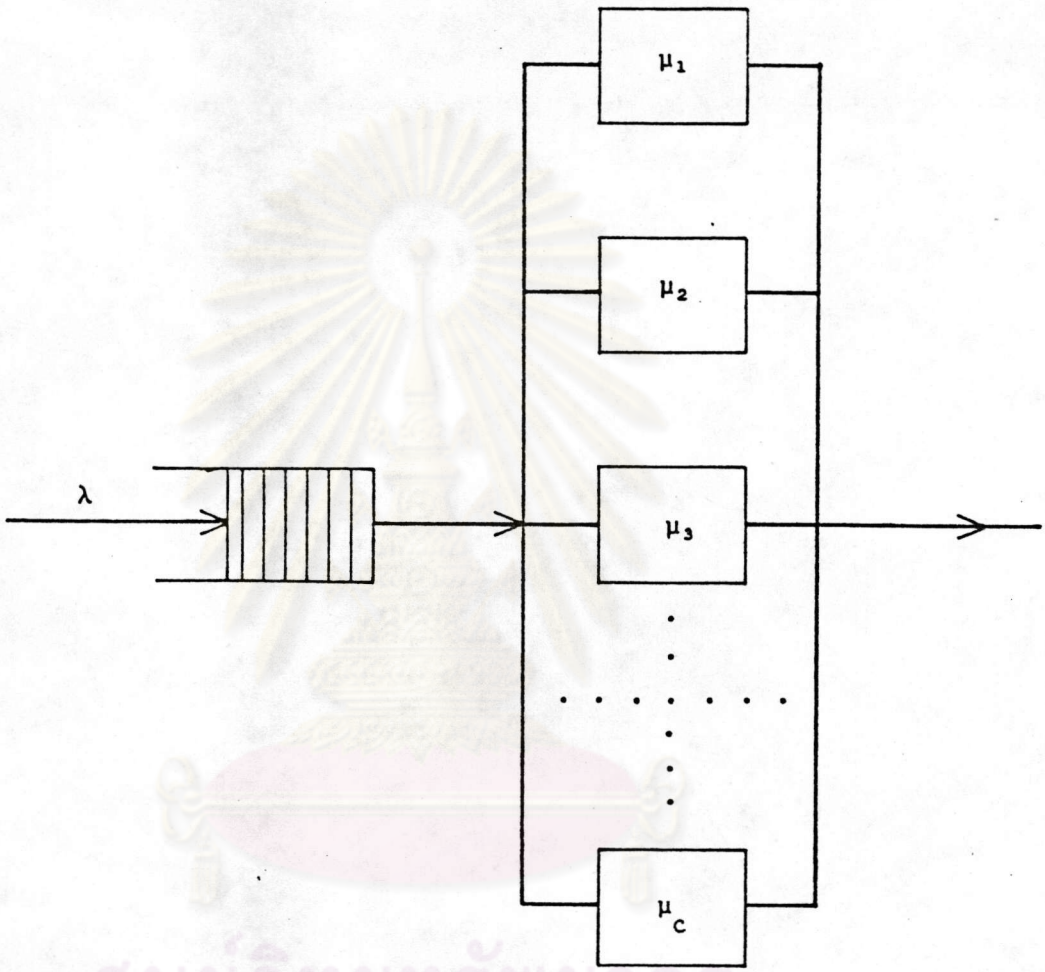

Fig 3.2 Queueing model (M/M/c with unidentical service rate) for the multiple-microprocessor system.

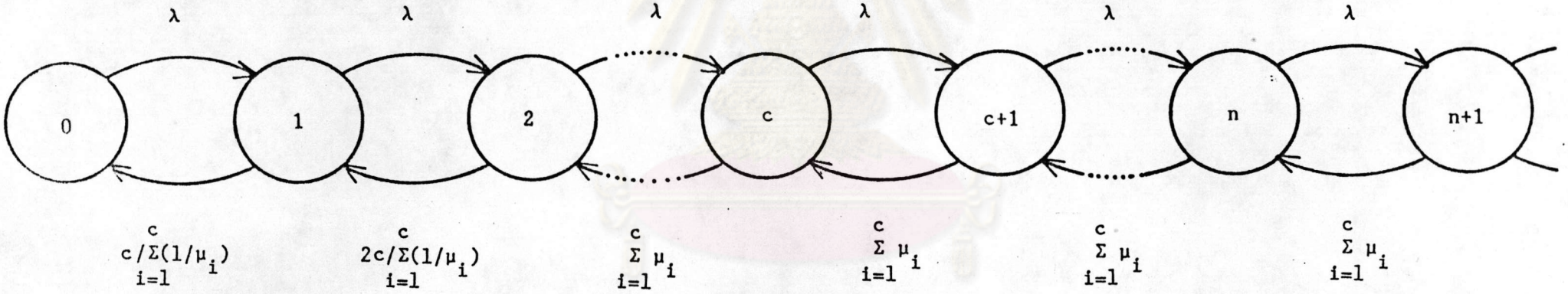

Fig 3.3 State diagram of the queueing model in Fig. 3.2

is shown in Fig.3.3, where P_n denotes the probability that the system is in state n.

From the state diagram, the steady state solution for P_n is summarized as follow:

$$P_n = \begin{cases} \frac{1}{n!}\delta^n P_o & 1 < n < c \\ \frac{\delta^{c-1}\rho^{n-(c-1)}}{(c-1)!} P_o & c \leqslant n \end{cases} \qquad (3.1)$$

where

$$\rho = \lambda / \sum_{i=1}^{c} \mu_i$$

$$\delta = \lambda / \frac{c}{\sum_{i=1}^{c} \frac{1}{\mu_i}}$$

(see Appendix A)

The summation of all probabilities is 1 then

$$P_o + \sum_{n=1}^{c-1} \frac{1}{n!}\delta^n P_o + \sum_{n=c}^{\infty} \frac{\delta^{c-1}}{(c-1)!} \rho^{n-(c-1)} P_o = 1 \qquad (3.2)$$

$$P_o = \left[\sum_{n=0}^{c-1} \frac{1}{n!}\delta^n + \sum_{n=c}^{\infty} \frac{\delta^{c-1}}{(c-1)!} \rho^{n-(c-1)} \right]^{-1}$$

Since

$$\sum_{m=1}^{\infty} \rho^m = \frac{\rho}{1-\rho}$$

then

$$P_o = \left[\sum_{n=0}^{c-1} \frac{1}{n!}\delta^n + \frac{\delta^{c-1}}{(c-1)!} \left(\frac{\rho}{1-\rho}\right) \right]^{-1} \qquad (3.3)$$

The expected number of transaction in the system now can be calculated.

$$E[N] = \sum_{n=0}^{\infty} nP_n$$

$$= \left[\sum_{n=1}^{c-1} \frac{n}{n!} \delta^n + \sum_{n=c}^{\infty} \frac{\delta^{c-1}}{(c-1)!} n\rho^{n-(c-1)} \right] P_o$$

Since

$$\sum_{n=c}^{\infty} n\rho^m = \sum_{m=1}^{\infty} m\rho^m + (c-1) \sum_{m=1}^{\infty} \rho^m$$

$$= \frac{\rho}{(1-\rho)^2} + \frac{(c-1)\rho}{(1-\rho)}$$

$$= \frac{\rho^2+c\rho-c\rho^2}{(1-\rho)^2}$$

where $m = n - c + 1$ then

$$E[N] = \left[\sum_{n=1}^{c-1} \frac{\delta^n}{(n-1)!} + \frac{\delta^{c-1}}{(c-1)!} \frac{(\rho^2+c\rho - c\rho^2)}{(1-\rho)^2} \right] P_o \qquad (3.4)$$

The expected response time of the system can be calculated from Little's theorem [28].

$$E[R] = \frac{E[N]}{\lambda} \qquad (3.5)$$

If the processing rate of all the microprocessor are identical then eq. (3.4) becomes the solution for standard M/M/c queueing model.

$$E[N] = \delta + \frac{\delta^c}{c!} \frac{\rho}{(1-\rho)^2} P_o$$

$$= c\rho + \frac{(c\rho)^c}{c!} \frac{\rho}{(1-\rho)^2} P_o \qquad (3.6)$$

When there are only two microprocessors in the system, eq.(3.4) is simplified to be :

$$E[N] = \frac{\lambda\,(\mu_1 + \mu_2)^3}{[2\mu_1\mu_2\,(\mu_1 + \mu_2 - \lambda) + \lambda(\mu_1 + \mu_2)^2]\,(\mu_1 + \mu_2 - \lambda)} \qquad c=2 \tag{3.7}$$

If the capacity of the shared memory is limited and it can store up to k transactions in the queue, eq. (3.2) becomes

$$p_o = \left[\sum_{n=o}^{c-1} \frac{1}{n!}\,\delta^n + \sum_{n=c}^{k} \frac{\delta^{c-1}}{(c-1)!}\,\rho^{n-(c-1)} \right]^{-1}$$

Since

$$\sum_{m=1}^{r} \rho^m = \frac{\rho\,(1-\rho^r)}{(1-\rho)}$$

then

$$p_o = \left[\sum_{n=o}^{c-1} \frac{1}{n!}\,\delta^n + \frac{\delta^{c-1}}{(c-1)!}\,\frac{\rho(1-\rho)}{(1-\rho)}^{k-c+1} \right]^{-1} \tag{3.8}$$

The expected number of transactions in the system now can be calculated from the following equation

$$E[N] = \left[\sum_{n=1}^{c-1} \frac{n}{n!}\,\delta^n + \sum_{n=c}^{k} \frac{\delta^{c-1}}{(c-1)!}\,n\rho^{n-(c-1)} \right] p_o$$

Since

$$\sum_{m=1}^{r} m\rho^m = \frac{1-(r+1)\rho+r\rho^{r+1}}{(1-\rho)^2}$$

then

$$E[N] = \left[\sum_{n=1}^{c-1} \frac{\delta^n}{(n-1)!} + \frac{1}{(1-\rho)^2}\,(1-A\rho^B + B\rho^A) + (c-1)\,\frac{(1-\rho^B)}{(1-\rho)} \right] p_o$$

where $A = k + c + 2$ (3.9)

$B = k - c + 1$

The system response time can then be calculated by using Little's theorem as in the previous section.

3.2 The Unsymmetrical Multiple-Microprocessor System with Processor Priority.

The multiple-microprocessor system considered in this section is shown in Fig. 3.4 . There are two microprocessors sharing the same resources. They are shared memory and input-output units. Each microprocessor has its local memory for program and private data storage. An external data is sent into the system through the shared input unit. The data processing rate of microprocessor-1 is higher than that of microprocessor-2. The input data transaction is scheduled to be processed by microprocessor-1 if both microprocessors are ready. The input data transaction will be processed by microprocessor-2 only if microprocessor-1 is busy. The processed data transaction is then sent back to the external device through the shared output unit. If both microprocessors are busy,the input data transaction is queued in the shared memory until either of the microprocessors is ready. This unsymmetrical multiple-microprocessor system becomes the symmetrical system if both microprocessors have the same data processing rate.

3.2.1 System Model.

The performance of the multiple-microcessor system with processor priority is analyzed by using queueing model. The performance considered in this section is the system response time. In order to simplify the queueing model, the following conditions are assumed.

1. Microprocessor-1 has a higher processing rate than that of microprocessor-2. The arriving data transaction is first scheduled to be processed by microprocessor-1. The arriving data is

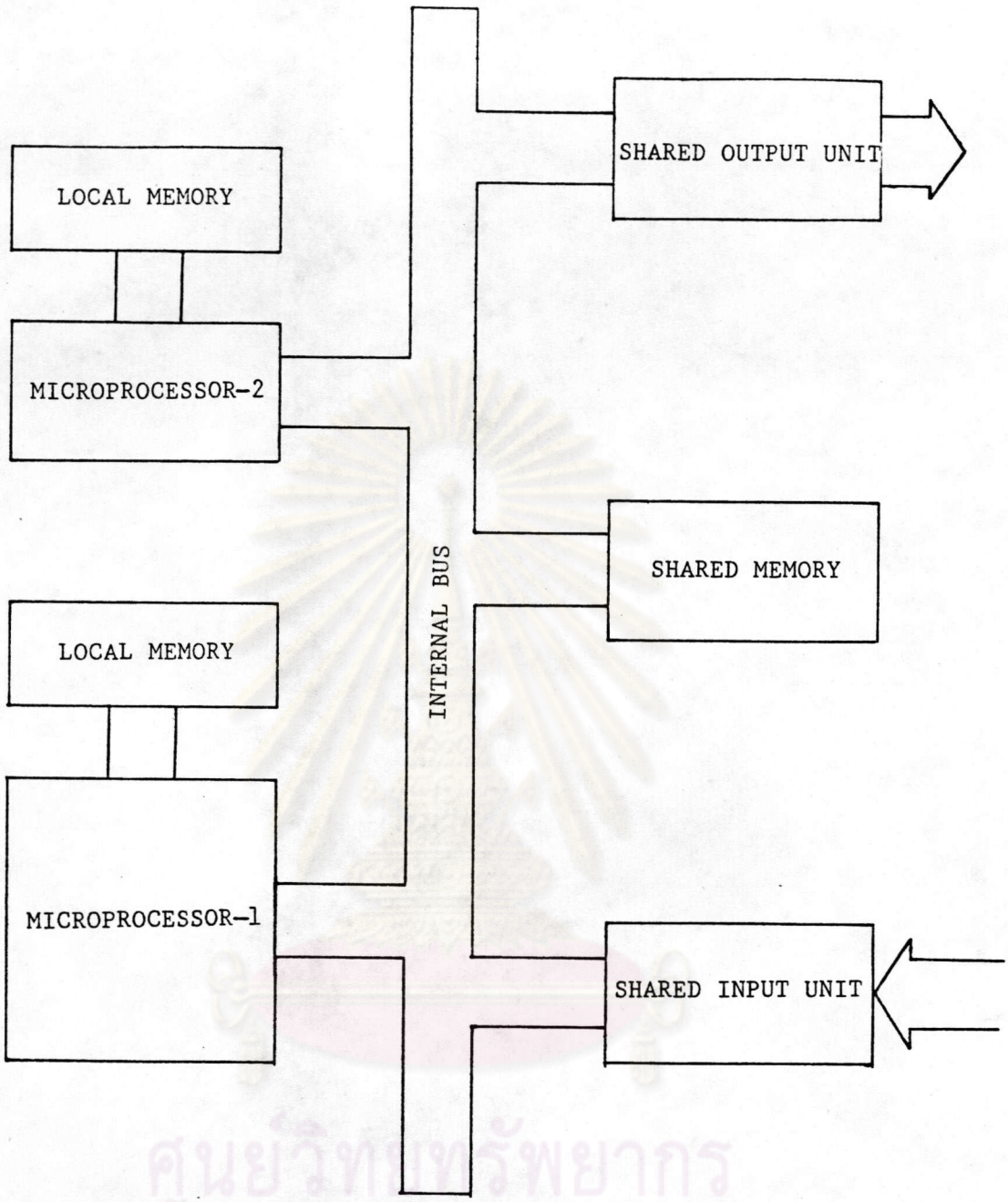

Fig 3.4 The unsymmetrical multiple-microprocessor system with processor priority.

processed by microprocessor-2 only when microprocessor-1 is busy.

2. The response times of the shared memory, the input unit, the output unit, and the internal bus are very short compared with the processing time of the microprocessors.

3. The shared memory capacity is very high compared with the amount of the incomming data.

4. The data transaction arrival rate is a random variables observing Poisson distribution with a mean of λ .

5. The processing time of the microprocessor is a random variable with exponential distribution. The mean processing rates of microprocessor-1 and microprocessor-2 are μ_1 and μ_2 ,respectively, and $\mu_1 > \mu_2$.

3.2.2 Analysis of the Model.

If n_1 denotes the number of data transactions in the queue including the transaction being processed at microprocessor-1, and n_2 denotes the number of data transactions at microprocessor-2 , due to assumption 1, then n_1 can be any integer starting from 0 to infinity while n_2 is either 0 or 1 .

The queueing model for the multiple-microprocessor system in Fig.3.4 is shown in Fig. 3.5.

The state of the system can be defined to be the tuple (n_1,n_2) and the state diagram of the queueing model is shown in Fig. 3.6; where $P(n_1,n_2)$ denotes the probability that the system is in state (n_1,n_2) .

Since our interest is in the steady state, balance equations can be written by equating the rate of flow into a state to the flow out of that state .

$$
\begin{aligned}
(\lambda+\mu_1+\mu_2)\,P(n,1) &= (\mu_1+\mu_2)\,P(n+1,1) + \lambda P(n-1,1)\\
(\lambda+\mu_1+\mu_2)\,P(1,1) &= (\mu_1+\mu_2)\,P(2,1) + \lambda P(0,1) + \lambda P(1,0)\\
(\lambda+\mu_2)\,P(0,1) &= \mu_1\,P(1,1)\\
(\lambda+\mu_1)\,P(1,0) &= \mu_2\,P(1,1) + \lambda P(0,0)\\
\lambda\,P(0,0) &= \mu_1\,P(1,0) + \mu_2\,P(0,1)
\end{aligned}
\qquad (3.10)
$$

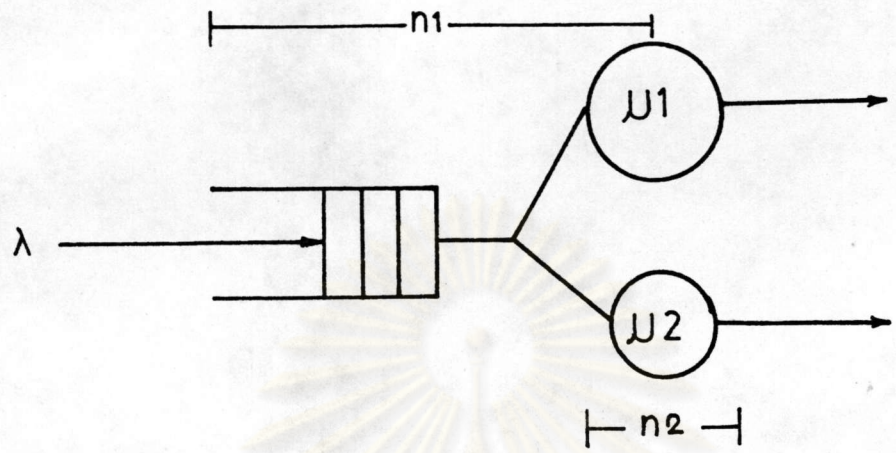

Fig 3.5 Queueing model of the unsymmetrical multiple-microprocessor system with processor priority.

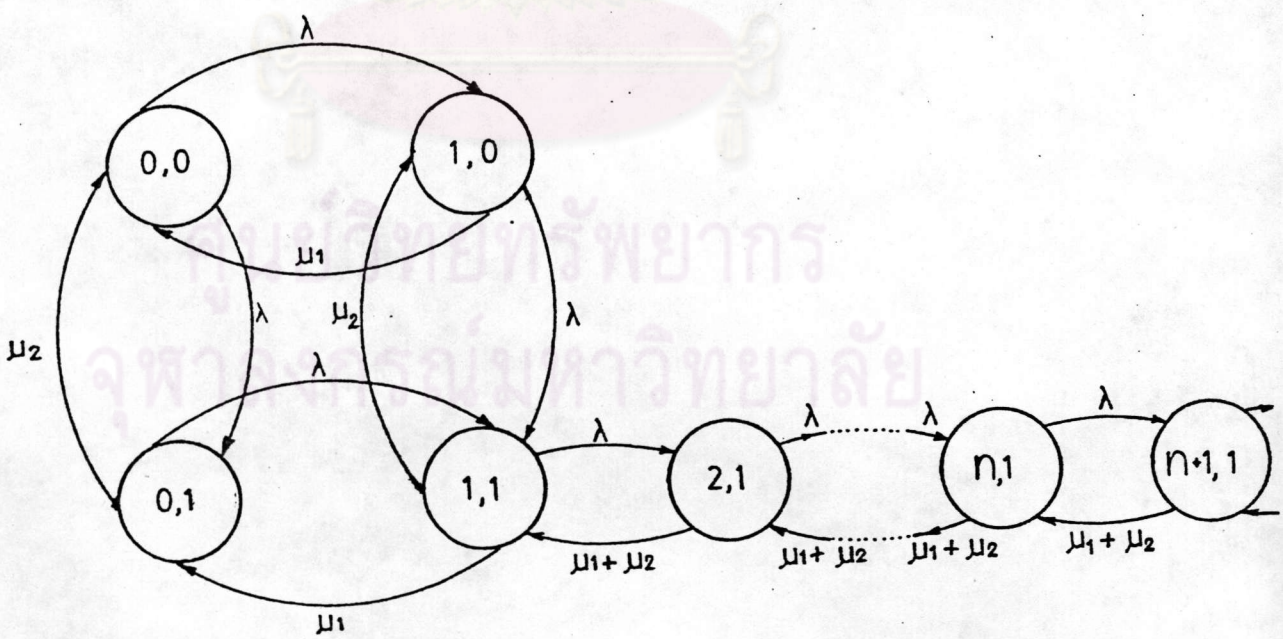

Fig 3.6 State diagram of the queueing model in Fig 3.5

Rewriting P(0,1),P(1,0) and P(1,1) in terms of P(0,0) from eq.(3.10) , we have

$$P(0,1) = \frac{\rho}{1+2\rho}\ \frac{\lambda}{\mu_2}\ P(0,0)$$

$$P(1,0) = \frac{1+\rho}{1+2\rho}\ \frac{\lambda}{\mu_1}\ P(0,0)$$

$$P(1,1) = \frac{\rho}{1+2\rho}\ \frac{\lambda(\lambda+\mu_2)}{\mu_1\mu_2}\ P(0,0) \qquad (3.11)$$

$$\text{Where}\quad \rho = \frac{\lambda}{\mu_1+\mu_2}$$

From eq. (3.11), the system is a Birth-Death process in the state ranging from state (1,1) to state (n,1), hence

$$P(n,1) = \frac{\lambda}{\mu_1+\mu_2}\ P(n-1,1)$$

$$= \rho\ P(n-1,1) \qquad (3.12)$$

$$= \rho^{n-1}\ P(1,1) \qquad n>1$$

The total probability is equal to 1; then

$$\sum_{n=1}^{\infty} P(n,1) + P(0,0) + P(0,1) + P(1,0) = 1$$

$$\sum_{n=1}^{\infty} \rho^{n-1}\ P(1,1) + P(0,0)\ [1+\frac{\rho}{1+2\rho}\ \frac{\lambda}{\mu_2} + \frac{1+\rho}{1+2\rho}\ \frac{\lambda}{\mu_1}] = 1$$

$$P(0,0)\ [(\frac{1}{1-\rho})(\frac{\rho}{1+2\rho})(\frac{\lambda+\mu_2}{\mu_1\mu_2})\ \lambda + 1 + \frac{\rho}{1+2\rho}\ \frac{\lambda}{\mu_2} + \frac{1+\rho}{1+2\rho}\ \frac{\lambda}{\mu_1}] = 1 \qquad (3.13)$$

$$P(0,0) = \frac{\mu_1\mu_2(1+2\rho)(1-\rho)}{\mu_1\mu_2(1+2\rho)(1-\rho) + \lambda(\lambda+\mu_2)}$$

The average number of data transactions being processed (E[N]) can be calculated from the following equation.

$$E[N] = \sum_{K=0}^{\infty} KP(K,0) + \sum_{K=0}^{\infty} (K+1)\ P(K+1,1)$$

$$= 1-P(0,0) + \frac{P(1,1)}{(1-\rho)^2} \qquad (3.14)$$

Substituting P(0,0), P(1,1) from eq.(3.11)and eq.(3.12) to eq.(3.14),we have

$$E[N] = \frac{\lambda(\lambda+\mu_2)}{\mu_1\mu_2(1-\rho)^2(1+2\rho) + \lambda(1-\rho)(\lambda+\mu_2)} \qquad (3.15)$$

From Little's theorem, the response time of the multiple-microprocessor system (E[R]) is

$$E[R] = \frac{\lambda+\mu_2}{\mu_1\mu_2(1-\rho)^2(1+2\rho) + \lambda(1-\rho)(\lambda+\mu_2)} \qquad (3.16)$$

If the shared memory can store only upto k transactions in the queue, eq. (3.13) becomes

$$\sum_{n=1}^{k} p(n,1) + p(1,0) + p(0,1) + p(0,0) = 1$$

$$p(0,0) = \frac{\mu_1\mu_2(1-\rho)(1+2\rho)}{\mu_1\mu_2(1-\rho)(1+2\rho) + \lambda(\lambda+\mu_2)(1-\rho)^{k+1}} \qquad (3.17)$$

Following the same procedure as in the previous derivation, the system response time can be calculated from the following equation.

$$E[R] = \frac{(\lambda+\mu_2)(1-A\rho^B + B\rho^A)}{\mu_1\mu_2(1+2\rho)(1-\rho)^2 + \lambda(\lambda+\mu_2)(1-\rho)(1-\rho^B)} \qquad (3.18)$$

where $A = k + 2$

$B = k + 1$

3.3 Experimental Verification.

The multiple-microprocessor system presented in chapter 3 is reconfiqurated as shown in Fig.3.7 for experiment. The experiment is set up to investigate the relation between our model and the actual system implementation. The system consists of three Z-80 microprocessors; one microprocessor is assigned to function as data transaction generator and the other two function as data transaction processors. Hence the experimental system simulates a situation of the

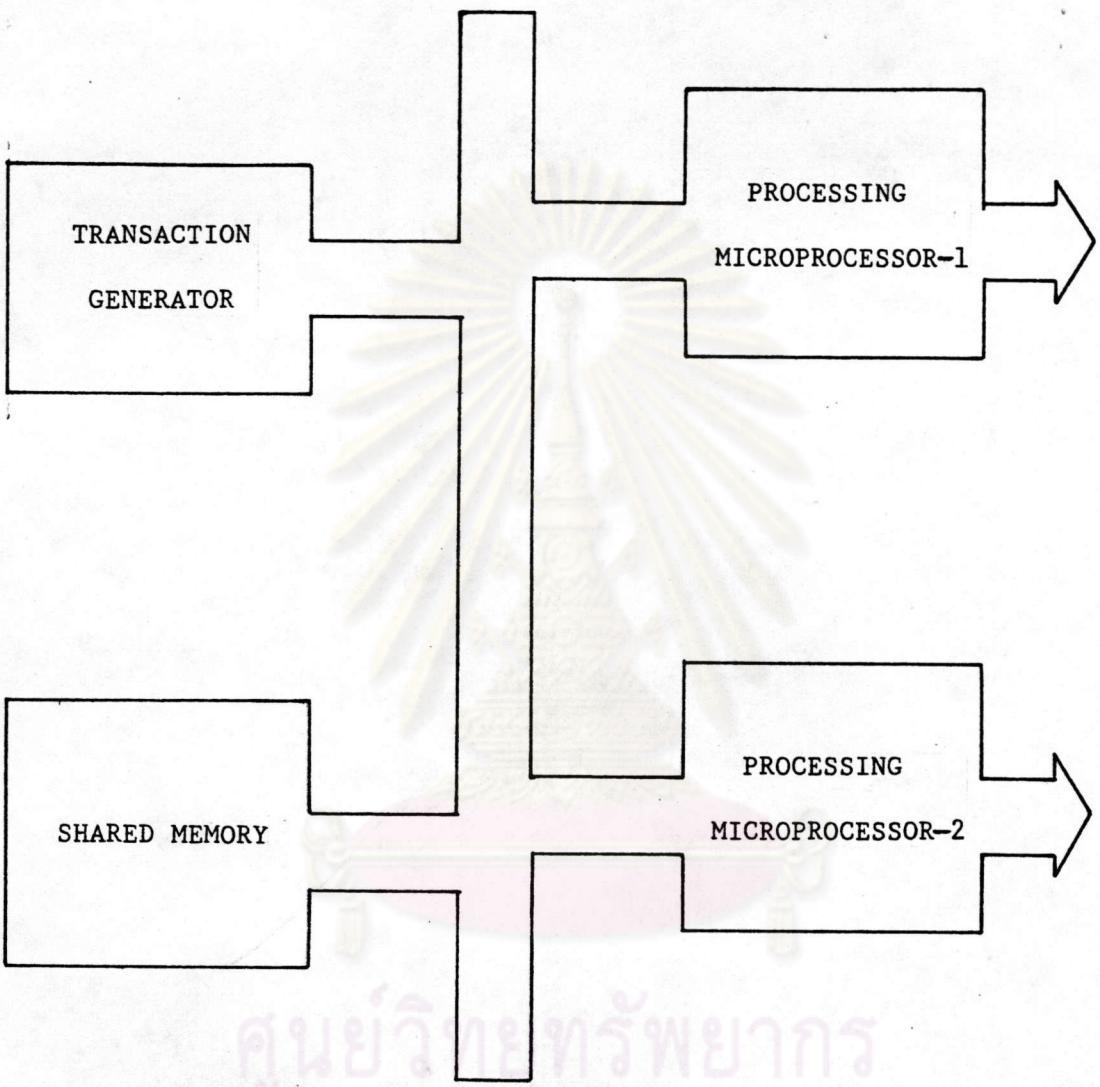

Fig 3.7 Block diagram of the experimental multiple-microprocessor system .

multiple-microprocessor system with two microprocessors. The data transactions are generated randomly according to Poisson distribution with an average transaction rate of 1 transaction per second. Each transaction is one-byte in length. The generated transactions are sent into the shared memory of 2K Byte. This ensures that there is no transaction loss from the queue in the shared memory. The transaction is then processed by either microprocessor-1 or microprocessor-2 . The processing rate of the microprocessor-1 is kept constant at 1 transaction per second for all experiments. Some delay loops are added to the control program of the microprocessor-2 in order to vary the processing rate from 0.1 to 0.9 transaction per second for each experiment. The average response time of the system in each experiment is measured and converted to that of the system with exponential distribution service time according to the equation from [29].

$$\left[\begin{array}{l}\text{Nonexponential}\\ \text{Response time}\end{array}\right] = \left[\begin{array}{l}\text{Exponential}\\ \text{Response time}\end{array}\right] \times \left[\tfrac{1}{2}\left|1 + \frac{\delta^2}{\mu}\right|\right]$$

where μ = Average service rate

δ = Standard deviation

The average response time of the system,measured in the experiment, are compared with that of the calculation from our model. The results are shown graphically in Fig. 3.8 . The results indicate that the model gives the same trend as the actual system but the model has a smaller response time. The difference between the response time calculated by our model and that measured from the experiment comes from the fact that there is some delay time in transfering data between each microprocessor through the shared memory.

It is generally believed that analytical models can provide estimates of average response time to within 30 percent accuracy [13]. The results from Fig.3.8 show that our analytical and experimental

results are agreeable with this limit. Hence, it is possible to use our model to predict the performance of the unsymmetrical multiple-microprocessor system in various configurations.

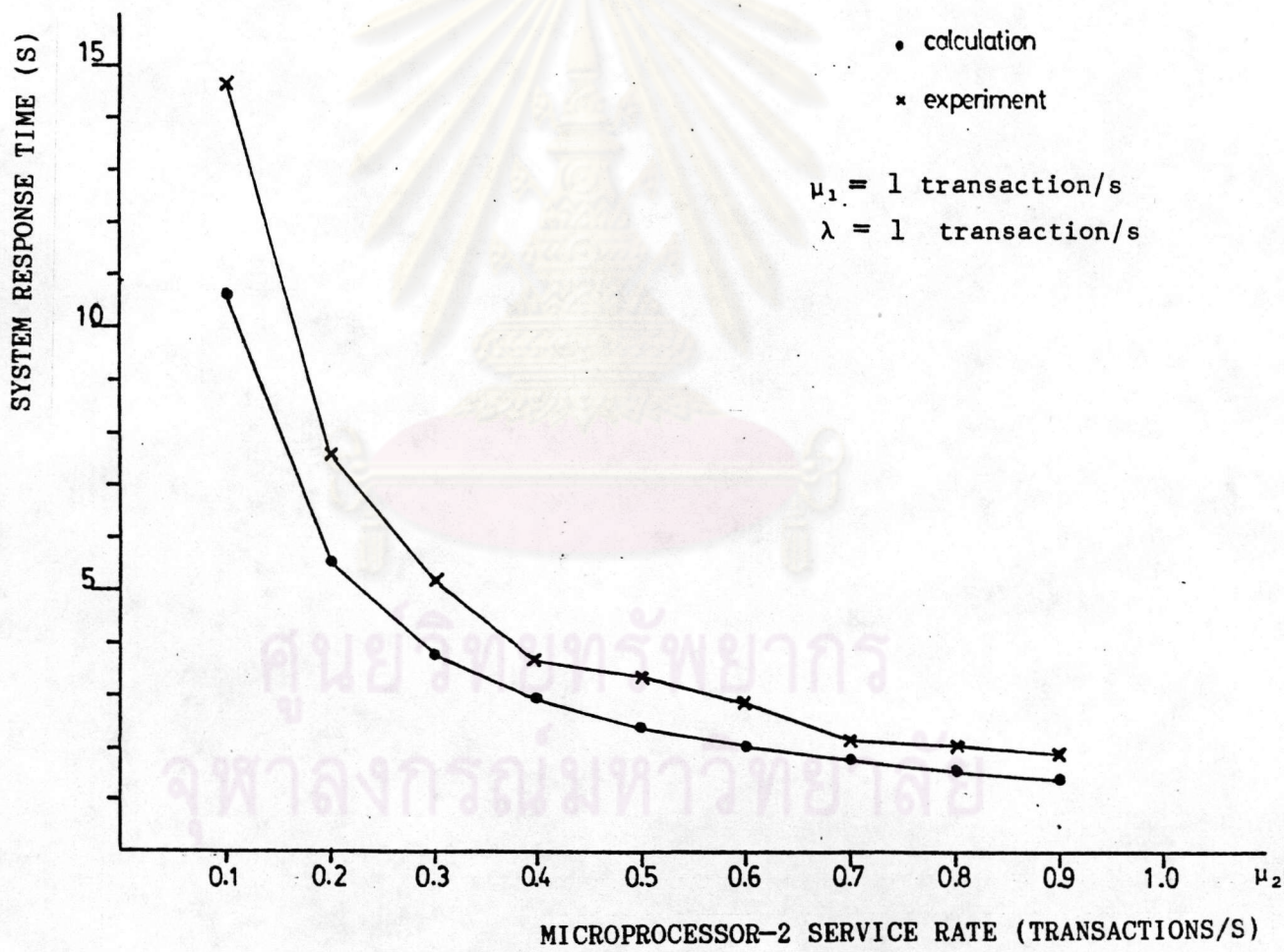

Fig 3.8 Response time of the experimental multiple-microprocessor system compared with that of the calculation from the queueing model.