On Modeling the Factory Dynamics

George A. Rovithakis and Manolis A. Christodoulou

Dept. of Electronic & Computer Engineering
Technical University of Crete
73100 Chania, Crete, GREECE

Abstract

In this paper a continuous time state space approach to the problem of modeling the factory dynamics is proposed. Several manufacturing structures are examined and an illustrative example is presented, to highlight the applicability of our modeling technique.

1 Introduction

The purpose of this paper[1] is to propose a new approach for the modeling of a factory plant. The problem can be stated as follows:

Factory Dynamics Modeling Problem: Find a mathematical model that will describe the unknown factory dynamics with a prescribe degree of accuracy; in other words the error between the actual dynamics and these of the proposed model should rely within a "small" neighborhood of zero. The dimension of the above mentioned neighborhood is a design requirement which will strongly affect the behavior of a future developed control law.

Until now, the modeling of factory dynamics has been based upon the folowing three methodologies: (i) Queuing networks [1]-[3], (ii) Activity cycle diagrams [4] and (iii) Petri nets [5]-[7] However, the above methodologies, basicaly lead up to quite involved procedures and moreover lack the property of adaptation which is crusial in order to deal with problems like accomodation of disturbances created by the plant facilities, by the environment, or by variable working times and plant failures.

In this paper, we propose a new model which we believe shows great promise in the direction of on line dynamic modeling of manufacturing systems. Crusial to its development is the notion of "basic module" which describes a single operation, performed by a machine, with its corresponding output buffer. Thus our manufacturing system model can be viewed as a general interconnection of "basic modules". These "basic modules" can adapt in order to "catch" the various operations that may take place in a plant and/or to cover for variable working times.

In the paper, we shall present the model in more detail by deriving mathematical relationships for some basic structures like Single Input Single Output series, or MISO series, MIMO and parallel interconnections, just to name a few. Moreover, an illustrative example will be given.

2 The Model

A manufacturing system can be viewed as a general interconnection of machines (M) and buffers (B). In order to better understand the operation of such a system it is usefull to divide it into basic

[1]This work is supported by the HIMAC-ESPRIT project 8141

modules. In this context the term "basic module" is used to describe a machine with its corresponding output buffer.

As it can be seen from figure 1, each "basic module" is set in operation, (it is controlled), through the switch S. Obviously, if $S = 1$ then the "basic module" is engaged, otherwise it is disengaged. It is obvious, that from the moment the switch S is closed, $S = 1$, a time T is needed for the machine M to make a product which is automatically placed on buffer B. Thus, if one measures the content of the buffer, he will notice a behaviour of the form shown in Fig. 2, which is a delayed unit step function. Hence, another way of representing it, is as a unit step response of a first order delay element which as it is well known from basic control systems theory it can also take the form:

$$\begin{aligned} \dot{x} &= -ax + au \\ y &= x \end{aligned} \qquad (1)$$

where a is a positive constant with

$$u = \begin{cases} 1 & \text{if } S = 1 \\ 0 & \text{if } S = 0 \end{cases}$$

and y demotes the content of the buffer. Thus graphically we obtain the approximation shown in Fig. 3. Note, however, that all intermidiate values between 0 and 1 do not have any physical interpretation. Equation (3.1) is meaningfull for the mathematical analysis and for the development of the "apropriate" control law. The buffer will take a product after the delay element has reached it is steady state value (x_{ss}) which is designed to be equal to one and the time needed to achieve such a goal is approximately equal to $5a$. To verify that $x_{ss} = 1$ take $\dot{x}|_{ss} = 0$ thus $-ax_{ss} + au = 0$ or $x_{ss} = u$, $a > 0$. However, since u is a unit step function, we have that $u = 1$ hence $x_{ss} = 1$. Moreover, the time required for the switch S to be on, is at least equal to $5a$, otherwise the delay element will fail to reach its steady state thus leading to modeling imperfections. When such a situation appears we say that the switch S is "too fast". Figure 4 illustrates this problem. The solid line presents the actual behaviour, while the dotted line the model behaviour. In fig. 4 the switch which closes at $t = 0, (S = 1)$ opens at time $t = T_0 < T$. Thus the switching frequency f_s should be at least equal to $1/T$, $f_s \leq \frac{1}{T} = \frac{1}{5a}$. In what follows we will discuss some fundamendal interconections of "basic modules" and we shall derive their corresponding mathematical models.

2.1 Series Single Input Single Output Interconnection

In this subsection we will discuss the case where we have two machines, M_1, M_2 with three buffers including one input and one output buffer. In this structure B_1 is the input buffer with infinite capacity buffer B_2 is assumed to have a maximum capacity b_{20} and buffer B_3, (output buffer), is also assumed to be upper bounded by b_{30}. In this interconection we have two "basic modules" connected in serries. From the above we can come up with the following mathematical model:

$$\begin{aligned} \dot{x}_1 &= -a_1 x_1 + a_1 u_1 \\ \dot{x}_2 &= -a_2 x_2 + a_2 u_2 \\ y_1 &= x_1 - u_2 \\ y_2 &= x_2 \end{aligned}$$

where y_1, y_2 are the contents of buffers B_2, B_3 respectively. Along with the above, the following constraints also hold a) $0 \leq y_1 \leq b_{20}$, b) $0 \leq y_2 \leq b_{30}$. The above SISO series interconnection can be expanded to the case where we have n "basic modules" as follows:

$$\begin{aligned} \dot{x} &= -Ax + Au \\ y &= Ix - Eu \end{aligned}$$

where $x = [x_1, x_2 \ldots, x_n]^T$, $y = [y_1, y_2 \ldots, y_n]^T$, $A = diag[a_1, a_2 \ldots, a_n]$

$$E = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

and I is the identity matrix. Again $0 \leq y_i \leq b_{i+1\,0}$, $i = 1, 2, \ldots, n$

2.2 Series Multiple Input Single Output Interconnection

In this subsection we extend our proposed model to include the case where machine 1 has multiple inputs. Figure 7, presents schematically the case. However in the above structure only one input is active at a time. Thus $\sum_{i=1}^{n} SI_i \leq 1$. As previously we assume the infinite capacitance of the input buffers, while buffers B_1 and B_2 are constained by b_{10} and b_{20} repectively $0 \leq b_i \leq b_{i0}$, $i = 1, 2$. From the above we can derive the following mathematical model

$$\dot{x}_1 = -a_1 x_1 + a_1 \sum_{i=1}^{n} v_i$$

where $v_i = SI_i$, $i = 1, 2, \ldots, n$ and $a_1 > 0$.

$$\begin{aligned} \dot{x}_2 &= -a_2 x_2 + a_2 u_2 \\ y_1 &= x_1 - u_2 \\ y_2 &= x_2 \\ u_2 &= S_1 \end{aligned}$$

with

$$\sum_{i=1}^{n} v_i \leq 1$$

$$0 \leq b_i \leq b_{i0} \ , \ i = 1, 2$$

Again we can extented the above formulation to include n-"basic modules".

$$\begin{aligned} \dot{x} &= -Ax + Au \\ y &= Ix - Eu \end{aligned}$$

where I is the identity matrix, $A = diag[a_1\ a_2 \ldots a_n]$, $x = [x_1\ x_2 \ldots x_n]^T$, $y = [y_1\ y_2 \ldots y_n]^T$, $u = [\sum_{i=1}^{n} u_i\ u_2 \ldots u_n]^T$

$$E = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

Again $0 \leq y_i \leq b_{i+1\ 0}$, $i = 1, 2, \ldots, n$.

2.3 Multi Input Multi Output Interconnection

Whenever we have Multi Input Multi Output interconnection the following model can be applied.

In Fig. 9 we have one machine with two mutually exclusive inputs and two corresponding outputs. Thus we have a Two Input Two Output setting. As we did in the previous subsections we can impose constraints on the buffers like: B_1, B_2 to have infinite capacity while B_3, B_4 are bounded by b_{30} and b_{40} respectively. Hence the following mathematical model can be derived

$$\begin{aligned} \dot{x}_1 &= -a_1 x_1 + a_1 u_1 \\ \dot{x}_2 &= -a_2 x_2 + a_2 u_2 \\ y_1 &= x_1 \\ y_2 &= x_2 \end{aligned}$$

with $u_1 + u_2 \leq 1$, $0 \leq y_i \leq b_{i0}$ $i = 1, 2$ where y_1 and y_2 are the contents of the buffers B_3, B_4 respectively. The above model can be extended to include n-inputs, n-outputs.

$$\begin{aligned} \dot{x} &= -Ax + Au \\ y &= x \end{aligned}$$

where $x = [x_1\ x_2 \ldots x_n]^T$, $y = [y_1\ y_2 \ldots y_n]^T$, $A = diag[a_1\ a_2 \ldots a_n]$ Again y is the buffer content.

2.4 Parallel Interconnection

The parallel interconnection of machines is nothing more than a number of SISO series structures together with a MISO interconnection model. For example take the case where we have two machines M_1, M_2 in parallel. Their corresponding output buffers constitute the inputs on machine M_3 which is in a MISO form. As we did in the previous subsections we assume infinite capacity for the buffers B_1, B_3 while buffers B_2, B_4, B_5 are uperbounded by a positive constant. However, now we do not assume that the inputs of the MISO basic module are mutually exclusive. On the contrary, all inputs are needed in order for machine M_3 to produce an element. From the above discussion we can derive the following mathematical model:

$$\dot{x}_1 = -a_1 x_1 + a_1 u_1$$

$$
\begin{aligned}
\dot{x}_2 &= -a_2 x_2 + a_2 u_2 \\
\dot{x}_3 &= -a_3 x_3 + a_3 (\frac{u_3 + u_4}{2}) \\
y_1 &= x_1 - u_3 \\
y_2 &= x_2 - u_4 \\
y_3 &= x_3
\end{aligned}
$$

$$0 \leq y_i \leq b_{0i} \ , i = 1, 2, 3$$

Observe that the denominator in the expression for $\dot{x}_3$ must be equal to the total number of parallel inputs, since the parallel module operates only if all corresponding inputs are set to one. In this way we can view $\frac{u_3+u_4}{2}$ as another control input, for which we know that its admisible values are either zero or one. The above can be easily extended to include n parallel branches.

$$
\begin{aligned}
\dot{x} &= -Ax + au_1 \\
y &= Ix - Eu_2
\end{aligned}
$$

where I is the identity matrix and $x = [x_1 \ x_2 \ldots x_{n+1}]^T$, $y = [y_1 \ y_2 \ldots y_{n+1}]^T$, $A = diag[a_1 \ a_2 \ldots a_n a_{n+1}]$, $u_1 = [u_{11} \ u_{21} \ldots u_{n1} \ u_{n+1\ 1}]^T$, $u_2 = [u_{12} \ u_{22} \ldots u_{n2}]^T$. with $u_{n+1\ 1} = \sum_{i=1}^{n} (\frac{u_{i2}}{n})$ and E is a $(n+1 \times n)$ matrix of the form:

$$
E = \begin{bmatrix}
1 & 0 & 0 & \cdots & 0 \\
0 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
0 & 0 & 0 & \cdots & 0
\end{bmatrix}
$$

where again y is the buffer content.

2.5 Example

In this example we consider the structure shown in figure 13. The above example contains two MIMO "basic modules" M_1, M_2. Define $u_1 = S_1, u_2 = S_4, u_3 = S_2, u_4 = S_3, u_5 = S_5$ and $y_1 =$ content of $B_2, y_2 =$ content of $B_6, y_3 =$ content of $B_3, y_4 =$ content of $B_5, y_5 =$ content of B_4 Constraints:

- B_1, B_7 have infinite capacity
- B_i are upper bounded by b_{i0} , $i = 2, 3, \ldots, 6$ respectively.

Employing the above we can derive the following mathematical model:

$$
\begin{aligned}
\dot{x}_1 &= -a_1 x_1 + a_1 u_1 \\
\dot{x}_2 &= -a_2 x_2 + a_2 u_2 \\
\dot{x}_3 &= -a_3 x_3 + a_3 u_3 \\
&\vdots \\
\dot{x}_5 &= -a_5 x_5 + a_5 u_5 \\
y_1 &= x_1 - u_3 \\
y_2 &= x_2 \\
y_3 &= x_3 - u_5 \\
y_4 &= x_4 - u_2 \\
y_5 &= x_5
\end{aligned}
$$

with

$$0 \leq y_i \leq b_{i0}, \ i = 1, 2, \ldots, 5$$

$$
\begin{aligned}
u_1 + u_2 &\leq 1 \\
u_3 + u_4 &\leq 1
\end{aligned}
$$

or in matrix form

$$
\begin{aligned}
\dot{x} &= -Ax + Au \\
y &= Ix - Eu
\end{aligned}
$$

$$0 \leq y_i \leq b_{i0}, \ i = 1, 2, \ldots, 5$$

$$
\begin{aligned}
u_1 + u_2 &\leq 1 \\
u_3 + u_n &\leq 1
\end{aligned}
$$

$$
\begin{aligned}
x &= [x_1 \ x_2 \ldots x_5]^T \\
y &= [y_1 \ y_2 \ldots y_5]^T \\
A &= diag[a_1 \ a_2 \ldots a_5]
\end{aligned}
$$

where I is the identity matrix and E is a 5×5 matrix of the form.

$$
E = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

3 Conclusions

The purpose of this report is to propose a new model for describing the evolution of a factory plant. Our proposal is simple and at the same time it can be seen to work satisfactorily. However, we should always have in mind that we build models to control a factory. Hence wheather a model is "good" or "bad" is staightforwardly connected to showing that the model under consideration leads relative easy to the development of suitable control laws. On the basis of the above statement our model shows great promise.

References

[1] E. Gelenbe and G. Pujolle, *Introduction to Queuing Networks*, J. Wiley & Sons, Chichester, 1987.

[2] S. Lavenberg (Ed.), *Computer Performance Modelling Handbook*, Academic press, Orlando, 1983.

[3] N. Deul, B. Grabowski & J. Kaltwasser, "Queuing Network Analysis of Flexible Manufacturing Systems", In G Menga & V. Kempe (Eds.), *Information in Manufacturing Automation, Proc. of the 5th Bilateral Workshop GDR-Italy*, Dresden, pp. 169-186, 1987.

[4] A. Carrie, *Simulation of Manufacturing Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1987.

[5] W. Reisig, *Petri Nets: An Introduction*, Springer-Verlag, Berlin, 1982.

[6] J. Kaltwasser, G. R. Friedrich, B. Muller & T. Vieweg, "Modelling and Simulation by Timed Petri Nets", In G Menga & V. Kempe (Eds.), *Information in Manufacturing Automation, Proc. of the 5th Bilateral Workshop GDR-Italy*, Dresden, pp. 222-246, 1987.

[7] E. Canuto & M. Vallauri, "Analysis of Complex Systems with Petri Nets", In M.H. Hamza (Ed.), *Proc. IASTED Int. Symp. on Modelling, Identification and Control, MIC'89*, pp. 5-10, 1989.

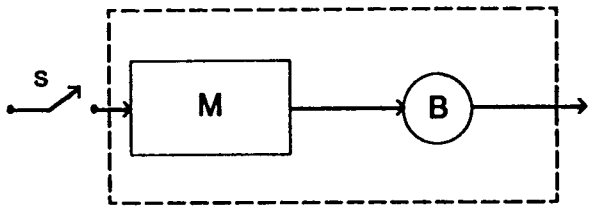

Figure 1: The "basic module"

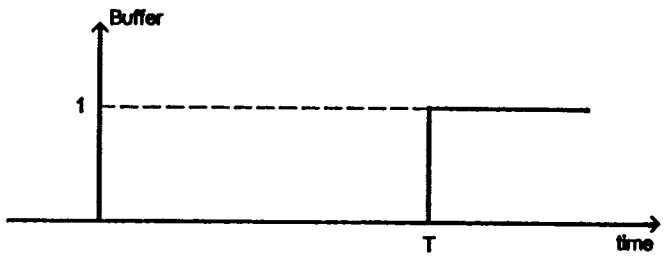

Figure 2: The delayed step function

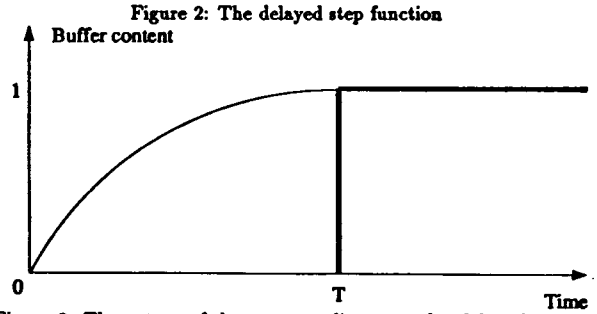

Figure 3: The output of the corresponding 1st order delay element

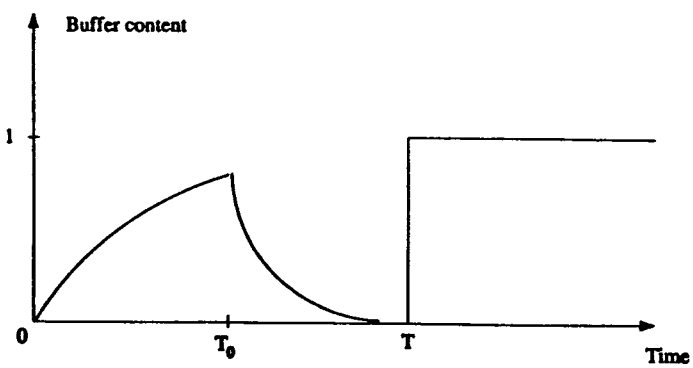

Figure 4: Fast switching problems

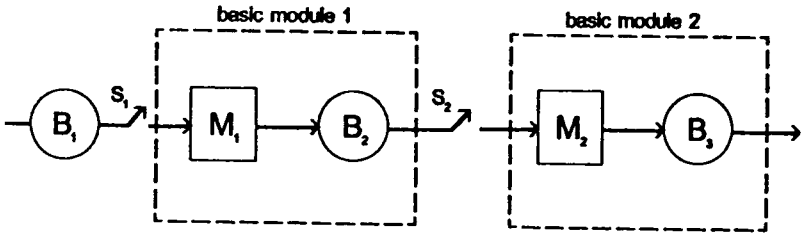

Figure 5: SISO Interconnection

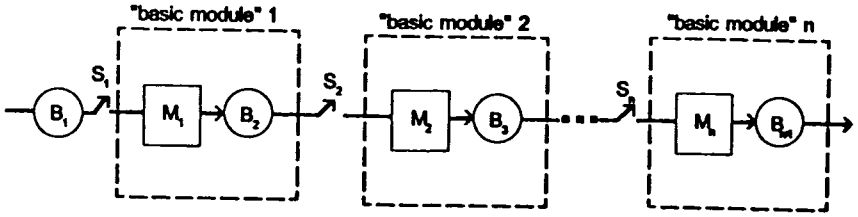

Figure 6: SISO Interconnection (n-"basic modules")

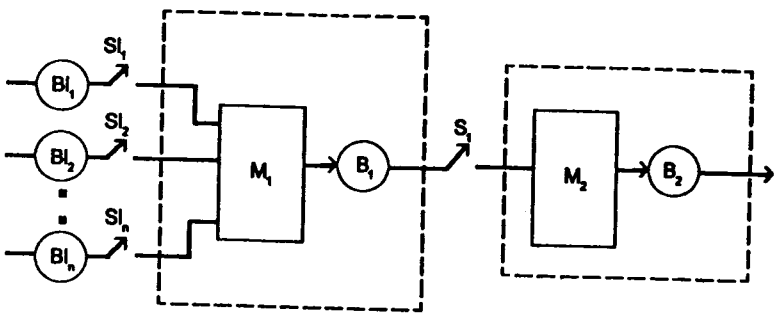

Figure 7: MISO Series Interconnection

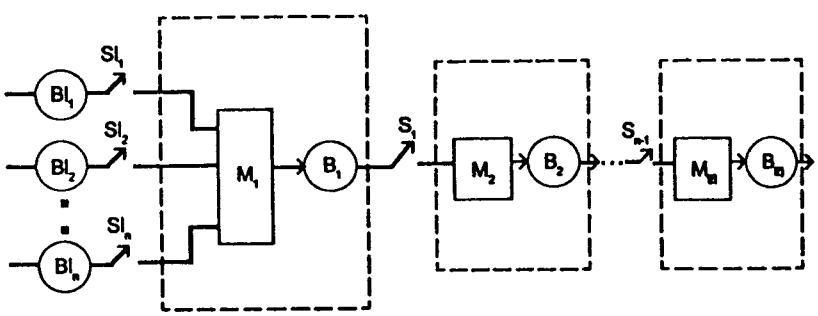

Figure 8: MISO Series Interconnection (n-"basic modules")

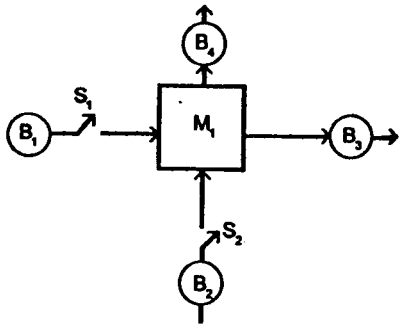

Figure 9: Two Input Two Output case

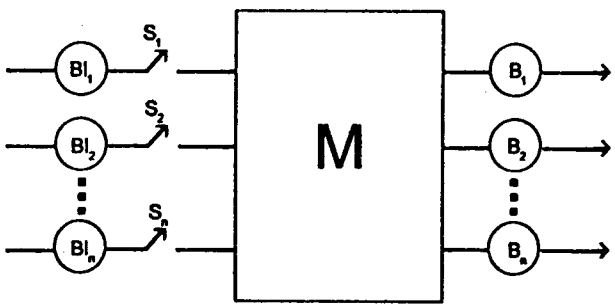

Figure 10: n-Input n-output case

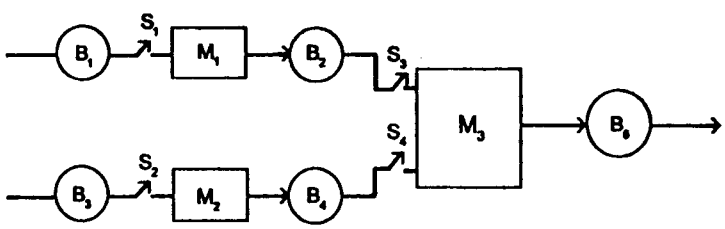

Figure 11: Parallel Interconnection

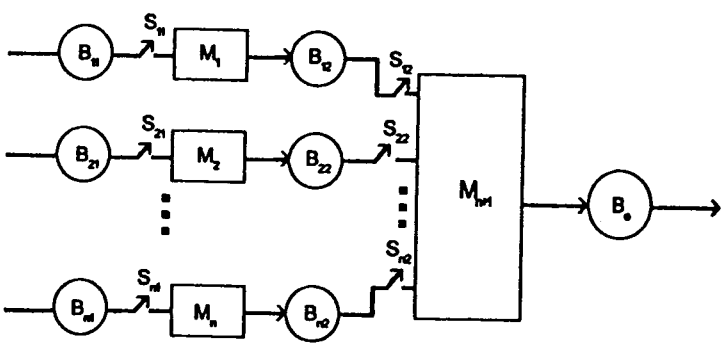

Figure 12: Parallel Interconnection of n series branches

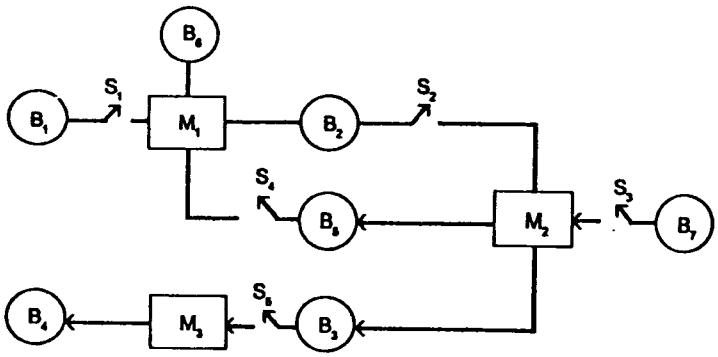

Figure 13: An example

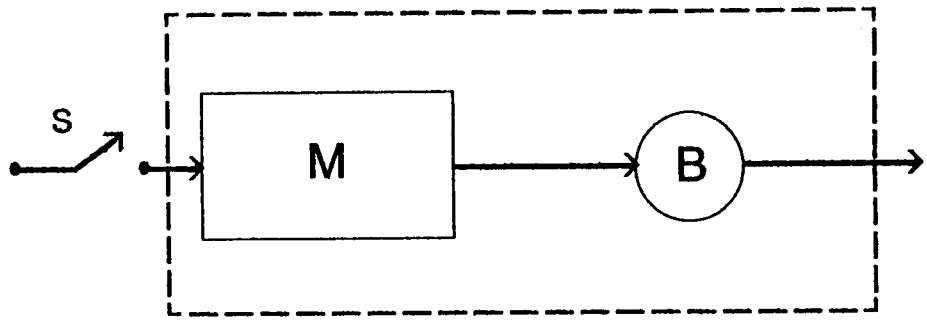

Figure 1: The "basic module"

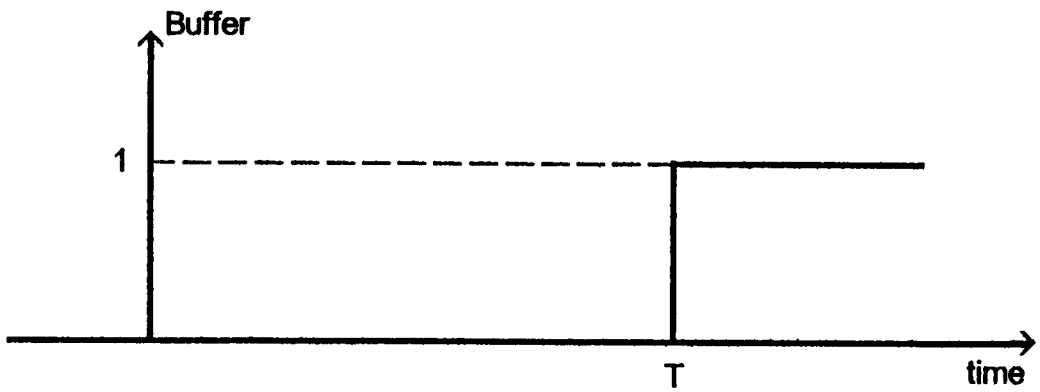

Figure 2: The delayed step function

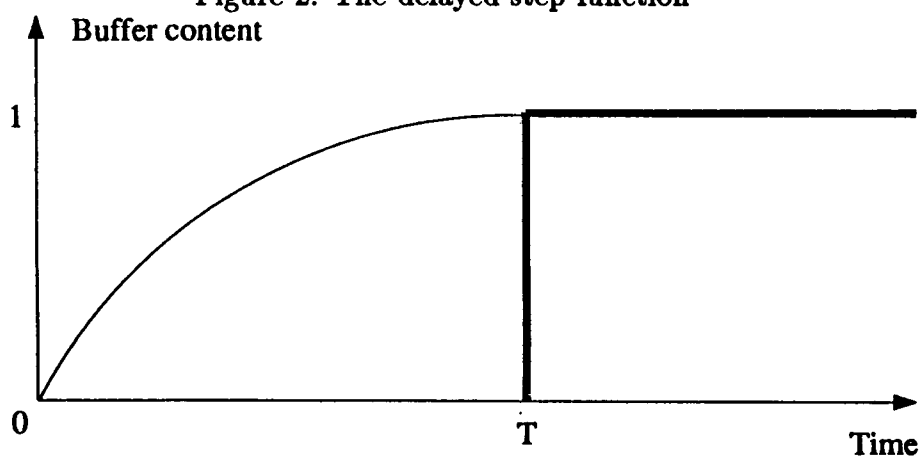

Figure 3: The output of the corresponding 1st order delay element

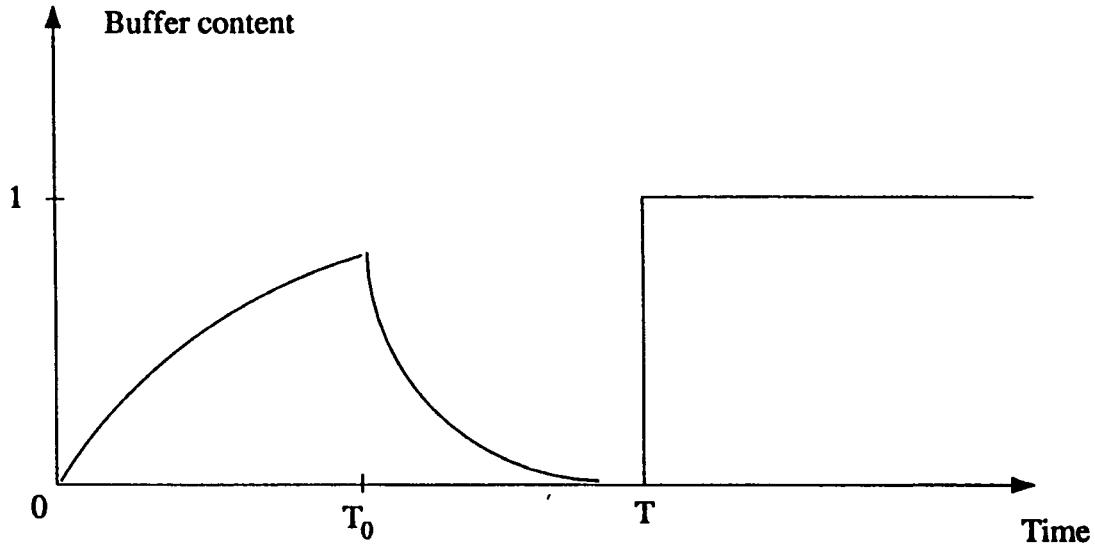

Figure 4: Fast switching problems

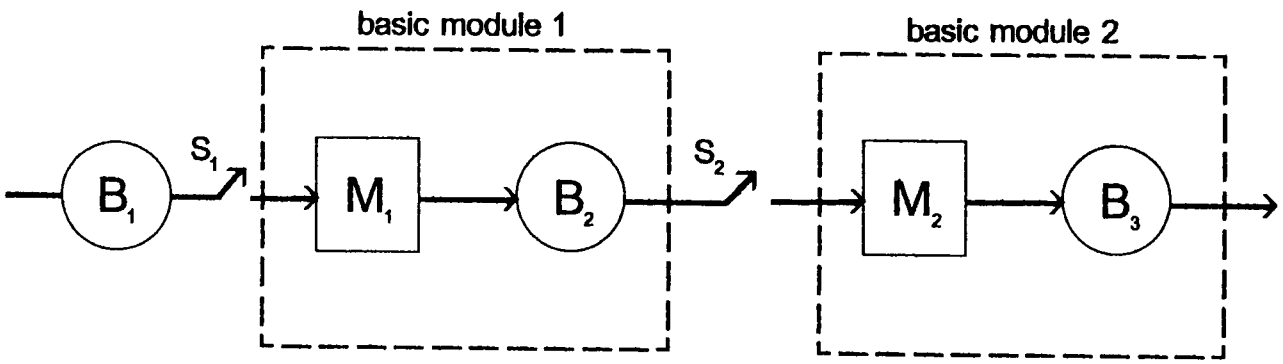

Figure 5: SISO Interconnection

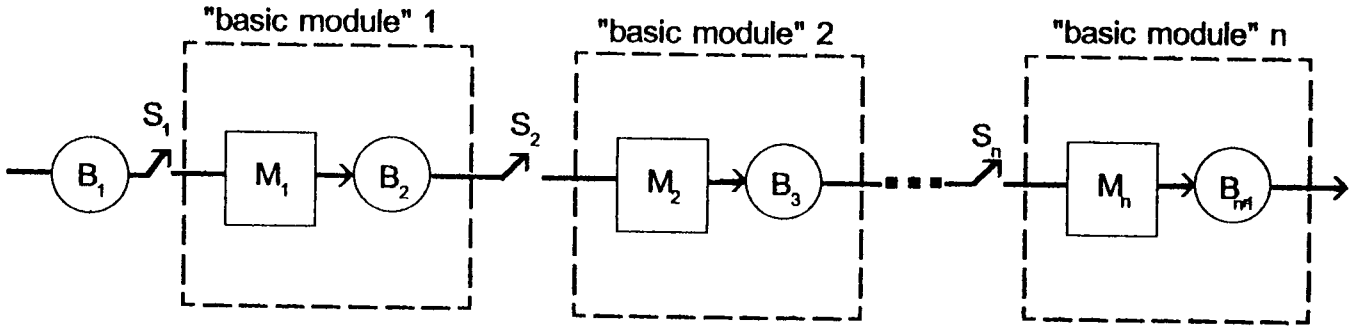

Figure 6: SISO Interconnection (n-"basic modules")

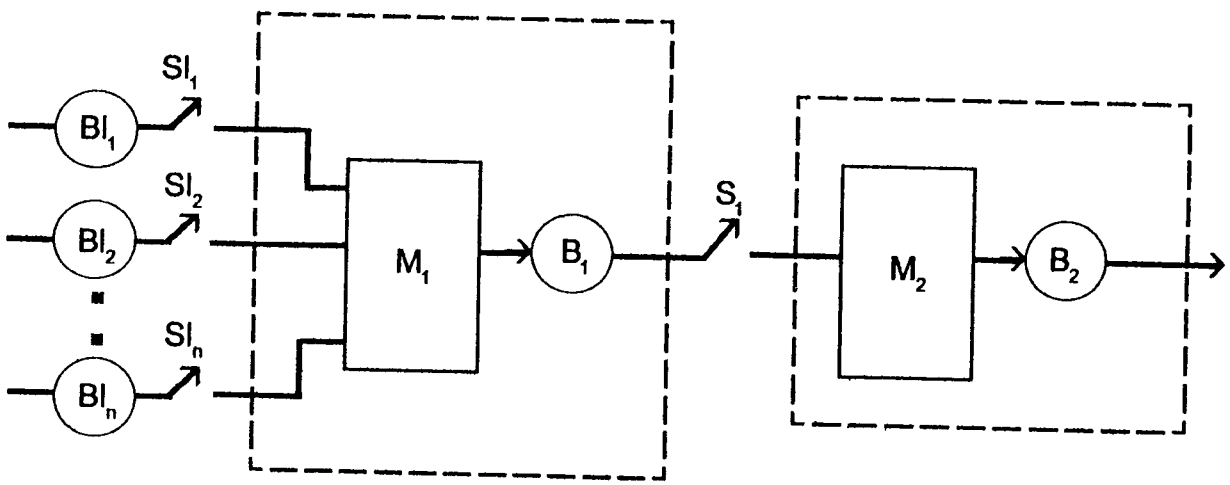

Figure 7: MISO Series Interconnection

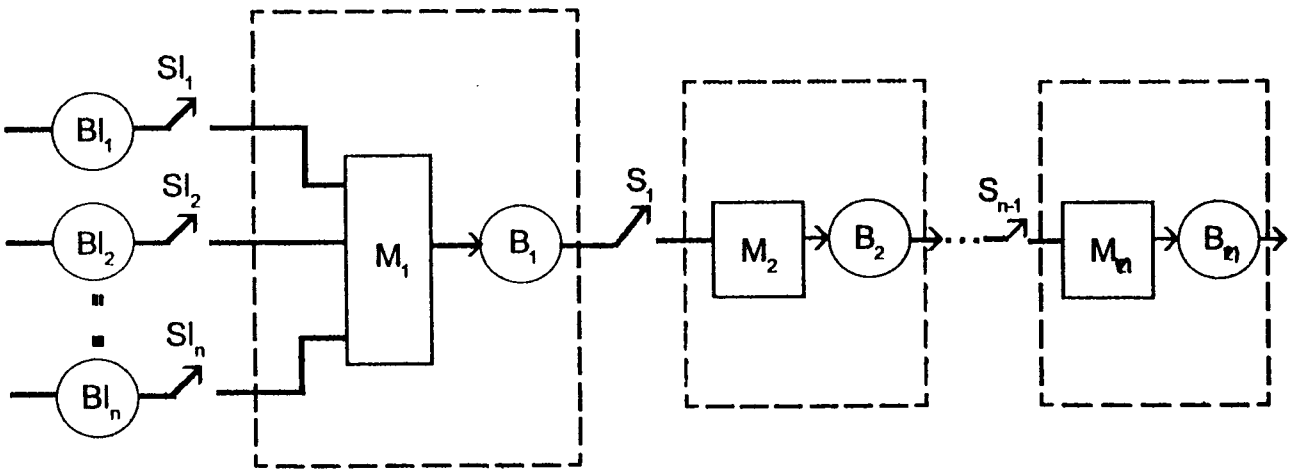

Figure 8: MISO Series Interconnection (n-"basic modules")

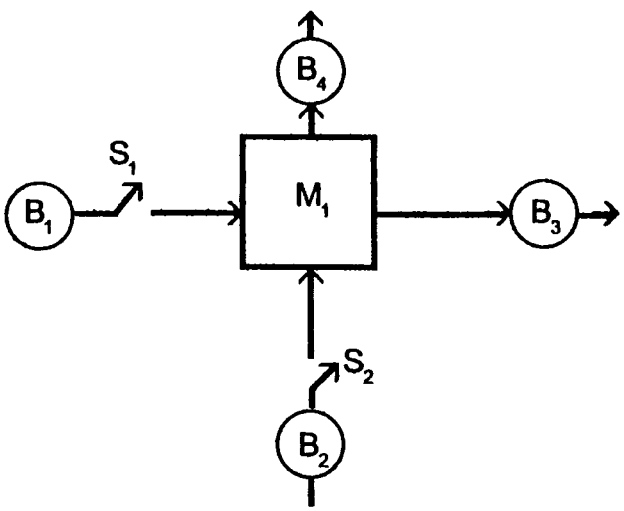

Figure 9: Two Input Two Output case

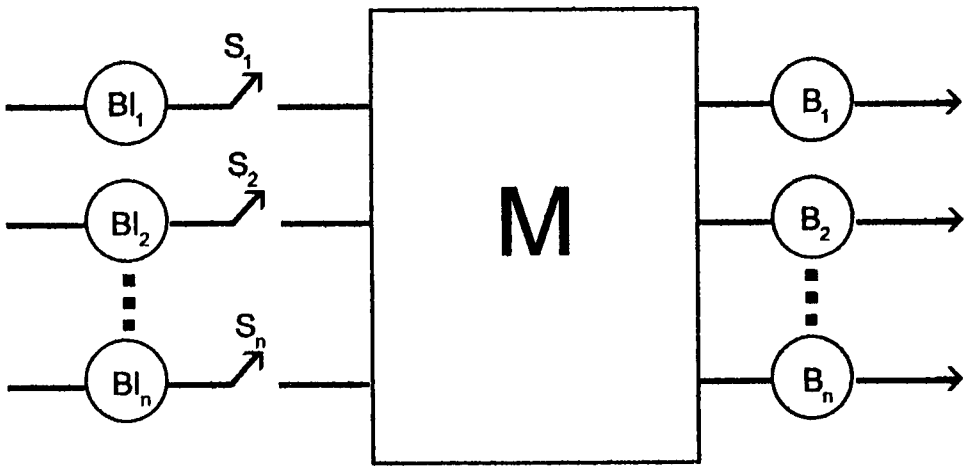

Figure 10: n-Input n-output case

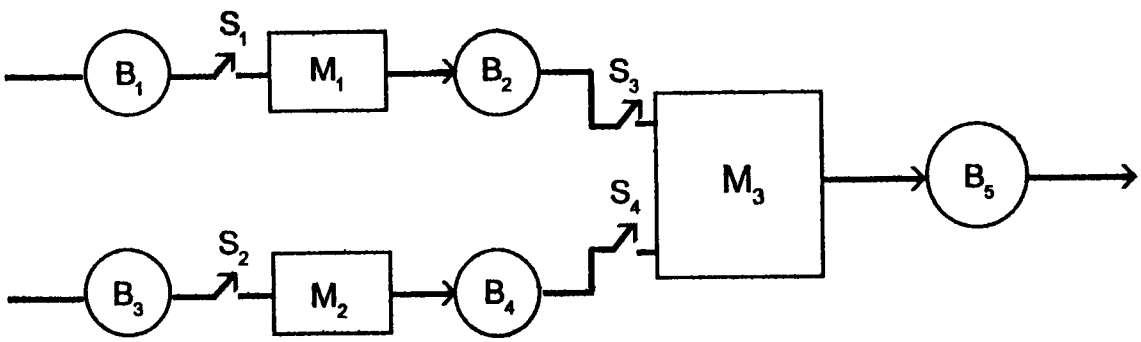

Figure 11: Parallel Interconnection

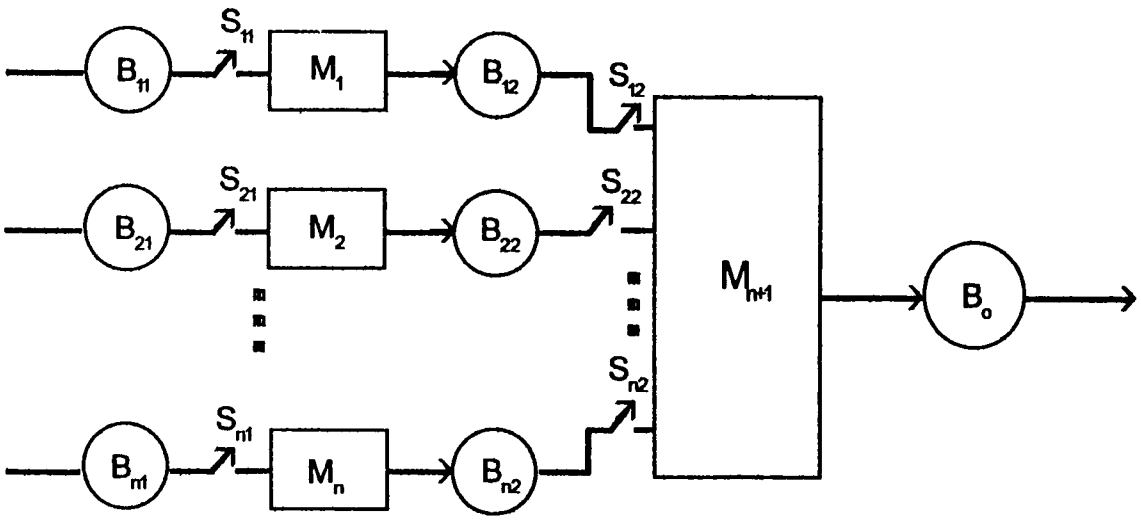

Figure 12: Parallel Interconnection of n series branches

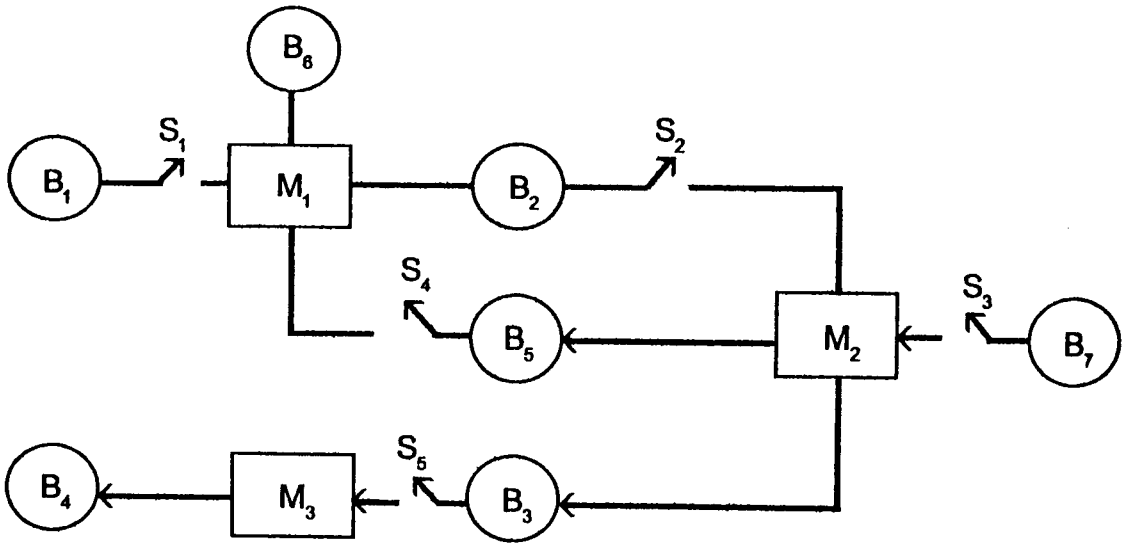

Figure 13: An example