A MULTIRATE SAMPLED-DATA CONTROLLER WITH MULTIPLEXED INPUTS AND OUTPUTS

BY

EKACHIDD CHUNGCHAROEN

Submitted in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in the
Electrical Engineering
Program

Robert W Foulkes Jr 10-1-90

Advisor Date

Sally M. Hotchkiss October 3, 1990

Dean of the Graduate School Date

YOUNGSTOWN STATE UNIVERSITY

DECEMBER, 1990

ABSTRACT

A MULTIRATE SAMPLED-DATA CONTROLLER WITH MULTIPLEXED INPUTS AND OUTPUTS

Ekachidd Chungcharoen

Master of Science in Electrical Engineering

Youngstown state University, 1990

In this thesis, the multirate sampled-data controller is extended to include multiplexed inputs and outputs. The controller detects the i^{th} plant output N_i times respectively during a period T_o with uniform sampling period T and changes the plant inputs once during T_o. Two designs to obtain the state transition and the input gain matrices of the controller are presented. It will be shown that if a plant satisfies three conditions of being controllable, observable, and having no zeros at origin, this controller can be made equivalent to the state variable feedback control law and the state transition matrix of the controller itself can be set arbitrarily in order to meet the controller's stability. The effects of disturbances to this controller are discussed and it will be explained how to choose the design parameters to minimize the disturbance effects. Control algorithms for this controller are developed and implemented on an IBM PC/AT microcomputer which has a multiplexed A/D and D/A converter board as an interfacing unit. Finally, three application examples are then presented to demonstrate the system performance.

ACKNOWLEDGEMENTS

I dedicate this thesis to my parents, Mr. Chitdee and Mrs. Chavewon Chungcharoen, for their love, understanding, encouragement and support which made my studies at Youngstown State University possible.

I would like to express my sincere appreciation to Dr. Robert H. Foulkes, Jr., my advisor, for his valuable advice and for his contribution to the theoretical developments presented in this thesis. Without his guidance, this thesis could never have been accomplished.

A word of appreciation is also given to Dr. Salvatore R. Pansino, chairman of the Electrical Engineering department, and Prof. Samual Skarote for their support and for being on my committee.

Special thanks are given to Dr. Raymond and Mrs. Martha Shuster, to Dr. Howard and Mrs. Marion Fritz, to Mrs. Anna Mae Serrecchio, to all the faculty and the Graduate school members, and to all my friends for their friendship, help and warm support during my two years of studies.

TABLE OF CONTENTS

PAGE

LIST OF SYMBOLS

SYMBOLS	DEFINITION
A,B,C,D	Coefficient matrices of the state-space model for continuous time system.
A'	Transpose of the matrix A.
$\bar{C},\bar{G}$	Coefficient matrices of the multirate sampling mechanism.
C_i	The i^{th} row of the output matrix C.
F	State variable feedback gain matrix.
H,H_{min}	Input gain matrix of the multirate sampled-data controller.
M	State transition matrix of the multirate sampled-data controller.
MIMO	Multiple-Input and Multiple-Output.
N_i	The i^{th} output multiplicity.
N_r	Feedforward gain matrix.
OIV	Observability Index Vector.
P	Observability Index matrix.
Q	Observability matrix.
R_x,R_u	Submatrices of $[\bar{C}\ \ \bar{G}]^{-1}$.
$r(kT_o)$	Reference input vector of the system.
T,T_i	Uniform sampling period during T_o.
T_o	Frame period.
$tr(A)$	Trace of the square matrix A.
u_i	The i^{th} input vector.
$u(t)$	Continuous-time input vector.
$u(kT_o)$	Discrete-time input vector.
$v(kT_o)$	Previous value of the input vector.

SYMBOLS	DEFINITION
$x(t)$	Continuous-time state vector.
$x(kT_o)$	Discrete-time state vector.
$\bar{Y}(kT_o)$	Multirate sampled-data output vector.
$y(t)$	Continuous-time output vector.
$y(kT_o)$	Discrete-time output vector.
Φ_i, Γ_i	Coefficient matrices of the discrete-time state equivalent model.
α	Vector integral of the state transition matrix of the continuous-time system.

LIST OF FIGURES

FIGURE PAGE

FIGURE PAGE

FIGURE **PAGE**

LIST OF TABLES

TABLES **PAGE**

CHAPTER I

INTRODUCTION

1.1 Background

In the design of a controller using a state-space approach, the procedure consists of two independent steps. The first step, called the state variable feedback control law design, assumes that all states are available for feedback purposes. Since typically not all states can be measured, the second step, called the observer law design, is used to estimate the entire state vector from given measurements of the portion of the state. The final control algorithms will consist of the control law and the observer law combined where the control law calculations are based on the estimated states rather than the actual states. The advantage of using this procedure exists in the fact that the whole design procedure is simplified. However, there are two clear disadvantages which accompany the use of the observer design: increase of the order of the system, and possibility of producing an unstable controller, which is undesirable from the viewpoint of stability.

To solve the above problem, using a digital computer, T. Hagiwara and M. Araki introduced a new type of controller called "multirate output sampled-data controller", which can be regarded as a special type of multirate sampled-data controller, as presented in an

article which appeared in the IEEE Transactions on Automatic Control [1]. The controller detects the i^{th} plant output N_i times during a period T_o and changes the plant inputs once during T_o. This controller has the following advantages. First, it has the same ability as the state variable feedback in adjusting the closed-loop characteristics of the control system. Second, it has the ability of choosing the arbitrary state transition matrix of the controller itself. Third, it can apply to the wide class of plants which are controllable and observable, have at least as many outputs as inputs, and do not have invariant zeros at origin. Fourth, calculations required in the design are almost the same as those required for the state variable feedback controller.

1.2 Objective

The controller in reference [1] requires specific hardware implementation such as a multi-processor computer and several A/D and D/A converters interfacing boards since the controller requires a non-uniform sampling period for each output, simultaneously sampling some outputs, and simultaneously updating all control inputs. These requirements present some difficulties in finding suitable hardware to implement the controller in the control laboratory.

The objective of this thesis is to modify the multirate output sampling mechanism from reference [1] to

include multiplexing of plant input and output signals, to present two designs of the multirate sampled-data controller based on theorems in reference [1], and to develop the multirate sampled-data control algorithms and implement the algorithms on a microcomputer which has a multi-purpose multiplexed A/D and D/A converter board as an interfacing unit.

1.3 Overview

Chapter II reviews a multirate output sampling mechanism and the theorems of the multirate sampled-data controller of reference [1]. Then, two designs of the multirate sampled-data controller with multiplexed outputs and multiplexed both inputs and outputs are presented. Explanations of how to choose the design parameters to minimize the disturbance effects are given.

Chapter III describes the appropriate hardware and software used for real-time implementation.

Chapter IV applies the two designs discussed in Chapter II to three application examples. The results of Computer Simulations are presented in comparison with the results of Real-time implementations.

Chapter V discusses results and problems. Finally, a conclusion and some suggestions for future work are included.

CHAPTER II

MULTIRATE SAMPLED-DATA THEORY

2.1 Introduction

In this chapter, the multirate output sampled-data controller of reference [1] is extended to include multiplexed inputs and outputs. Section 2.2 reviews the Observability Index Vector, the multirate output sampling mechanism and the multirate sampled-data theorems based on reference [1]. Section 2.3 modifies the multirate output sampling mechanism to include the multiplexing of output samples and presents two designs of the multirate sampled-data controller. In section 2.4, the design for a system that requires both multiplexed inputs and outputs is then presented.

2.2 Observability Index Vector and Multirate Output Sampling Mechanism

Consider the state-space model of the linear continuous time-invariant plant :

$$\frac{dx}{dt}(t) = Ax(t) + Bu(t) \qquad (2.2.1)$$

and

$$y(t) = Cx(t) \qquad (2.2.2)$$

where $x(t)$ is an $n \times 1$ state vector,

A is an $n \times n$ matrix,

$u(t)$ is an $m \times 1$ input vector,

B is an n x m input matrix,

y(t) is an p x 1 output vector,

and C is an p x n output matrix.

The fundamental assumption is that this plant is completely controllable and observable, and does not have invariant zeros at origin. Concerning the observable pair (A,C), the pn x n observability matrix

$$Q = \begin{bmatrix} C \\ CA \\ CA^2 \\ \cdot \\ CA^{n-1} \end{bmatrix} \qquad (2.2.3)$$

has full rank, which means Q has n linearly independent rows. The **Observability Index** of the system is defined as the smallest integer σ for which the matrix

$$Q(\sigma) = \begin{bmatrix} C \\ CA \\ \cdot \\ \cdot \\ CA^{\sigma-1} \end{bmatrix} \qquad (2.2.4)$$

has rank n. Generally, for a multiple-output system $\sigma \leq n$. Assuming y(t) has p independent components, the C matrix in equation (2.2.2) is composed of p independent rows $(C_1,\ldots,C_p)$. Therefore, it is always possible to make a selection of the n linearly independent vectors which comprise the rows of the matrix $\overline{Q}(\sigma)$ of the form

$$\bar{Q}(\sigma) = \begin{bmatrix} C_1 \\ C_1A \\ \vdots \\ C_1A^{n_1-1} \\ C_2 \\ C_2A \\ \vdots \\ C_2A^{n_2-1} \\ \vdots \\ C_p \\ \vdots \\ C_pA^{n_p-1} \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} n_1 \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} n_2 \\ \vdots \\ \left.\begin{matrix} \\ \\ \\ \end{matrix}\right\} n_p \end{matrix} \qquad (2.2.5)$$

where

$$\sum_{i=1}^{p} n_i = n.$$

In this case, a set of the smallest p integers $(n_1,\ldots,n_p)$ for which the matrix $\bar{Q}(\sigma)$ has full rank is called **Observability Index Vector** (abbreviated as OIV) [2]. The Observability Index Vector plays a significant role in the theory of the multirate sampled-data controller as will be seen next.

From the plant in equation (2.2.1), if a sampler and a zero-order hold circuit is connected to each plant input, then

$$u(t) = u(kT_o) \quad ; \quad (kT_o \le t < kT_o+T_o).$$

Therefore, for any T such that $0 \le T \le T_o$,

$$x(kT_o+T) = \exp(AT)x(kT_o) + \left(\int_o^T \exp(At)B\,dt\right)u(kT_o). \qquad (2.2.6)$$

From equation (2.2.2), if the i^{th} plant output is detected at every T_i, the sampled-data output value becomes

$$y_i(kT_o+\mu T_i) = C_i x(kT_o+\mu T_i) \qquad (2.2.7)$$

where $\mu = 0,\ldots,N_i-1$ and $i = 1,\ldots,p$.

Here C_i is the i^{th} row of the output matrix C, and T_i and T_o are related by

$$T_i = T_o/N_i \quad ; N_i = \text{positive integer.} \qquad (2.2.8)$$

Equation (2.2.8) implies that the i^{th} plant output is detected N_i times during T_o. T_o is referred to as the input sampling period or **the frame period**, T_i is referred to as the i^{th} output sampling period, and $N_1,N_2,\ldots,N_p$ are referred to as the output multiplicities. The above sampling mechanism is the multirate output sampling mechanism introduced in [1]. An example of this multirate output sampling mechanism is given in Figure 2.2.1 on the next page.

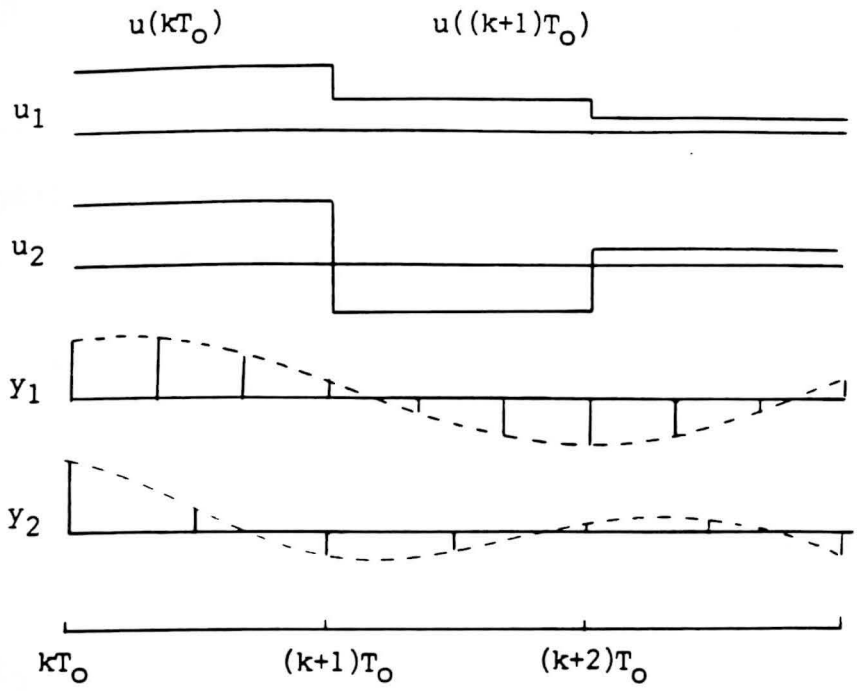


Figure 2.2.1 Multirate output sampling mechanism for a 2 inputs and 2 outputs plant with $N_1=3$ and $N_2=2$ introduced by [1].

Letting $T = T_o$ in equation (2.2.6), a zero-order hold equivalent model becomes

$$x(kT_o+T_o) = \exp(AT_o)x(kT_o) + \left(\int_o^{T_o} \exp(At)B\,dt\right)u(kT_o). \quad (2.2.9)$$

Letting $\mu T_i=T$ in equation (2.2.6), a zero-order hold equivalent model for each sampling period μT_i becomes

$$x(kT_o+\mu T_i) = \exp(A\mu T_i)x(kT_o) + \left(\int_o^{\mu T_i} \exp(At)B\,dt\right)u(kT_o). \quad (2.2.10)$$

Substituting equation (2.2.10) into equation (2.2.7) gives

$$y_i(kT_o+\mu T_i) = C_i\exp(A\mu T_i)x(kT_o) + C_i\left(\int_o^{\mu T_i}\exp(At)B\,dt\right)u(kT_o) \tag{2.2.11}$$

for $i=1,\ldots,p$ and $\mu=0,\ldots,N_i-1$.

The equation (2.2.11) is expressed in terms of vector-matrix form as shown below.

$$\bar{Y}(kT_o) = \bar{C}x(kT_o) + \bar{G}u(kT_o) \tag{2.2.12}$$

where

$$\bar{Y}(kT_o) = \begin{bmatrix} y_1(kT_o) \\ \vdots \\ y_1(kT_o+(N_1-1)T_1) \\ \vdots \\ y_p(kT_o) \\ \vdots \\ y_p(kT_o+(N_p-1)T_p) \end{bmatrix}, \tag{2.2.13}$$

$$\bar{C} = \begin{bmatrix} C_1 \\ \vdots \\ C_1\exp(A(N_1-1)T_1) \\ \vdots \\ C_p \\ \vdots \\ C_p\exp(A(N_p-1)T_p) \end{bmatrix}, \tag{2.2.14}$$

and

$$\bar{G} = \begin{bmatrix} 0 \\ \vdots \\ C_1 \int_0^{(N_1-1)T_1} \exp(At)B\,dt \\ \vdots \\ 0 \\ \vdots \\ C_p \int_0^{(N_p-1)T_p} \exp(At)B\,dt \end{bmatrix}. \qquad (2.2.15)$$

Therefore the vector $\bar{Y}(kT_o)$ is composed of the multirate sampled-data of the outputs in a single frame period T_o and its relation to the input and the states during a frame period is given by equation (2.2.12). In order that the multirate output sampled-data controller can be realized, the coefficient matrix $\bar{C}$ in equation (2.2.12) must have full rank. This can be achieved by selecting the output multiplicities N_i large enough, as is shown next.

Lemma 1 [1]: The matrix $\bar{C}$ given by equation (2.2.14) will have full rank(=n) for almost every frame period T_o if the output multiplicities $(N_1,N_2,\ldots,N_p)$ satisfy

$$N_i \geq n_i \qquad (i=1,\ldots,p) \qquad (2.2.16)$$

where $(n_1,\ldots,n_p)$ is the OIV of the pair (A,C) founded in equation (2.2.5).

Proof:

Since the rank of a matrix is not changed by elementary row operations, to compute the rank of $\bar{C}$, elementary row

operations are applied to the matrix $\bar{C}$ as shown below. Recall equation (2.2.14):

$$\bar{C} = \begin{bmatrix} C_1 \\ \vdots \\ C_1\exp(A(N_1-1)T_1) \\ \vdots \\ C_p \\ \vdots \\ C_p\exp(A(N_p-1)T_p) \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \bar{C}_1 \\ \\ \\ \\ \bar{C}_p \\ \\ \end{matrix}$$

Applying the following row operations to $\bar{C}_i$ for i=1,...,p in equation (2.2.14) gives

$$\begin{aligned} 1^{st}\ \text{row} &= 1^{st}\ \text{row} = C_i \\ 2^{nd}\ \text{row} &= (2^{nd}\ \text{row} - 1^{st}\ \text{row})/T_i \\ &= C_i\{\exp(AT_i)-I\}/T_i \\ 3^{rd}\ \text{row} &= (3^{rd}\ \text{row} - 2(2^{nd}\ \text{row}) + 1^{st}\ \text{row})/T_i^2 \\ &= C_i\{\exp(AT_1)-I\}/T_i^2 \\ &\vdots \\ N_i^{th}\ \text{row} &= C_i\{\exp(AT_i)-I\}/T_i^{N_i-1}. \end{aligned}$$

Therefore, the obtained matrix is

$$\bar{C}^+ = \begin{bmatrix} C_1 \\ \vdots \\ C_1\{\exp(AT_1)-I\}^{N_1-1}/T_1^{N_1-1} \\ \vdots \\ C_p \\ \vdots \\ C_p\{\exp(AT_p)-I\}^{N_p-1}/T_p^{N_p-1} \end{bmatrix}. \qquad (2.2.17)$$

Deleting appropriate rows in $\bar{C}^{+}$ gives

$$\bar{C}^{++} = \begin{bmatrix} C_1 \\ \cdot \\ C_1\{\exp(AT_1)-I\}^{n_1-1}/T_1^{n_1-1} \\ \cdot \\ \cdot \\ C_p \\ \cdot \\ C_p\{\exp(AT_p)-I\}^{n_p-1}/T_p^{n_p-1} \end{bmatrix}. \qquad (2.2.18)$$

The definition of the matrix exponential $\exp(AT_i)$ is given by

$$\exp(AT_i) = I + (AT_i) + \frac{1}{2!}(AT_i)^2 + \frac{1}{3!}(AT_i)^3 + \ldots \qquad (2.2.19)$$

Substituting equation (2.2.19) into equation (2.2.18), as T_o goes to 0, the limit of equation (2.2.18) becomes

$$\bar{C}^{++} = \begin{bmatrix} C_1 \\ \cdot \\ C_1A^{n_1-1} \\ \cdot \\ \cdot \\ C_p \\ \cdot \\ C_pA^{n_p-1} \end{bmatrix}. \qquad (2.2.20)$$

Therefore, the determinant of equation (2.2.18), as T_o goes to zero, goes to that of equation (2.2.20), which is nonzero since the plant is observable. Since the determinant of equation (2.2.18) is a continuous function of its entries and its limit is nonzero as T_o goes to zero, it is nonzero for sufficiently small T_o. By using Analytic

Function theorem [11], since all entries in equation (2.2.18) are analytic function of T_o, it follows that the determinant of equation (2.2.18) is also analytic and therefore nonzero except at isolated values of T_o. Therefore, $\bar{C}$ also has full rank. This completes the proof of Lemma 1.

Extending to use the pair $[\bar{C} \quad \bar{G}]$ gives the next result.

Lemma 2 [1]: Suppose that

$$\text{rank of} \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} = n+m. \tag{2.2.21}$$

Then, the matrix $[\bar{C} \quad \bar{G}]$ given by equation (2.2.14) and equation (2.2.15) has full rank (=n+m) for almost every frame period T_o if the output multiplicities $(N_1,\ldots,N_p)$ satisfy

$$N_i \geq m_i \tag{2.2.22}$$

where $(m_1,\ldots,m_p)$ is an OIV of the augmented system

$$\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, [C \quad 0] \right). \tag{2.2.23}$$

Proof: Observing that

$$\exp\left(\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} \right) T = \begin{bmatrix} \exp(AT) & \int_o^T \exp(At)B\,dt \\ 0 & I \end{bmatrix}.$$

One can see that the matrix $[\bar{C} \quad \bar{G}]$ has the same structure as the matrix $\bar{C}$ if (A,C) is replaced by the pair (2.2.23) of the coefficient matrices of the augmented system. The result follows from Lemma 1 [1]. It should be noted that "for almost every frame period T_o" means the assertion fails only at isolated value of T_o.

Next, consider the multirate sampled-data controller introduced by [1]. It includes the multirate sampling mechanism $\bar{Y}(kT_o)$ given in equation (2.2.13) as follows:

$$u(kT_o+T_o) = Mu(kT_o) - H\bar{Y}(kT_o) + N_r r(kT_o) \qquad (2.2.24)$$

where M is the m x m state transition matrix,

H is the m x n input gain matrix,

and N_r is the m x r feedforward gain matrix.

This equation means that the control inputs for the $(kT_o+T_o)^{th}$ frame period are determined based on the values of the control inputs for the kT_o^{th} frame period and the sampled-data outputs obtained during the kT_o^{th} frame period (and the reference input for tracking purposes). The above controller is also interpreted as an m^{th} order discrete-time system whose state is $u(kT_o)$. Therefore, the stability of the controller is determined by the eigenvalues of the state transition matrix M.

Concerning the matrix $\bar{C}$, the multirate sampled-data control law in equation (2.2.24) can be made equivalent to the

state variable feedback control law by the following theorem:

Theorem 1 [1]: Suppose that (A,C) is an observable pair and the output multiplicities $(N_1,\ldots,N_p)$ satisfy

$$N_i \geq n_i \qquad (i=1,\ldots,p)$$

where $(n_1,\ldots,n_p)$ is an OIV of the pair (A,C). Then for almost every frame period T_o, one can make the control law in equation (2.2.24) equivalent to the state variable feedback control law by suitable choice of the matrics H and M.

Proof:

Consider the state variables feedback control law given for the kT_o^{th} sampling period by

$$u(kT_o) = - Fx(kT_o) + N_r r(kT_o) \qquad (2.2.25)$$

where F is the feedback Gain matrix.

For the next frame period, the control law becomes

$$u(kT_o+T_o) = - Fx(kT_o+T_o) + N_r r(kT_o+T_o). \qquad (2.2.26)$$

From equation (2.2.9), the discrete-time system for a frame period T_o can be written as

$$x(kT_o+T_o) = \Phi x(kT_o) + \Gamma u(kT_o) \qquad (2.2.27)$$

where $\Phi = \exp(AT_o)$ and $\Gamma = \left(\int_o^{T_o} \exp(At)dt\right)B$.

Substituting equation (2.2.27) into equation (2.2.26) gives

$$u(kT_o+T_o) = -F[\Phi x(kT_o)+\Gamma u(kT_o)] + N_r r(kT_o+T_o)$$

$$= -F\Phi x(kT_o) - F\Gamma u(kT_o) + N_r r(kT_o+T_o). \quad (2.2.28)$$

Substituting equation (2.2.12) into equation (2.2.24) gives

$$u(kT_o+T_o) = Mu(kT_o) - H[\bar{C}x(kT_o)+\bar{G}u(kT_o)] + N_r r(kT_o)$$

$$= Mu(kT_o) - H\bar{C}x(kT_o) + \bar{G}u(kT_o) + N_r r(kT_o+T_o). \quad (2.2.29)$$

Assuming the reference input is constant for tracking, $r(kT_o) = r(kT_o+T_o)$. Then equation (2.2.28) is equivalent to equation (2.2.29) if and only if

$$-F\Phi x(kT_o) - F\Gamma u(kT_o) = Mu(kT_o) - H\bar{C}x(kT_o) - H\bar{G}u(kT_o). \quad (2.2.30)$$

Writing equation (2.2.30) in matrix form gives

$$H \begin{bmatrix} \bar{C} & \bar{G} \end{bmatrix} \begin{bmatrix} x(kT_o) \\ u(kT_o) \end{bmatrix} = \begin{bmatrix} F\Phi & F\Gamma+M \end{bmatrix} \begin{bmatrix} x(kT_o) \\ u(kT_o) \end{bmatrix}. \quad (2.2.31)$$

Therefore, the control law in equation (2.2.24) will be equivalent to the state variable feedback control law in equation (2.2.26) if the matrix H satisfies

$$H\bar{C} = F\Phi \quad (2.2.32)$$

and the matrix M satisfies

$$H\bar{G} = F\Gamma + M. \quad (2.2.33)$$

Since Lemma 1 implies that the matrix $\bar{C}$ has full rank,

there exists the matrix H which satisfies equation (2.2.32) for any specified feedback gain matrix F corresponding to the desired state feedback. The resulting closed-loop system using the multirate sampled-data controller in equation (2.2.24) is shown in Figure 2.2.2.

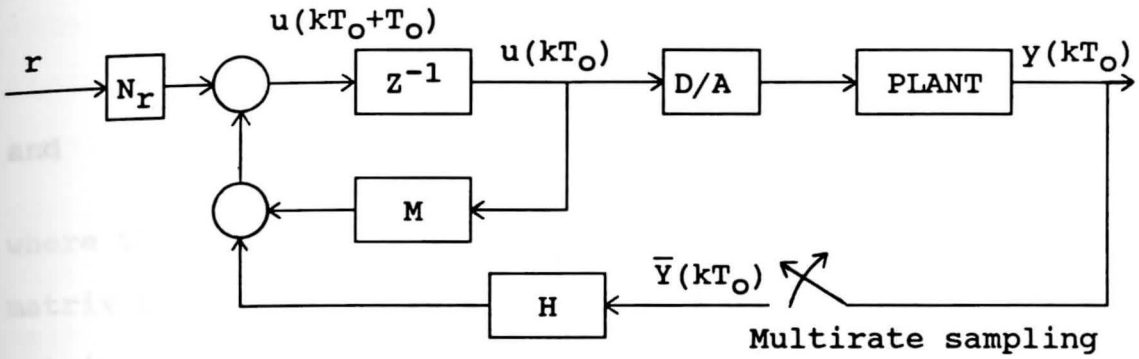


Figure 2.2.2 Closed-loop configuration using a multirate sampled-data controller.

Concerning the matrix $[\bar{C} \quad \bar{G}]$, the multirate sampled-data control law in equation (2.2.24) can also be made equivalent to the state variable feedback control law by the following theorem:

Theorem 2 [1]: Suppose that (A,C) is an observable pair and that

$$\text{rank} \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} = n+m.$$

Further suppose that the output multiplicities $(N_1,\ldots,N_p)$

satisfy

$$N_i \geq m_i \qquad (i=1,\ldots,p)$$

where $(m_1,\ldots,m_p)$ is an OIV of the augmented system in equation (2.2.23). Then, for almost every frame period T_o, there exists the matrix H such that

$$H\bar{C} = F\Phi \tag{2.2.34}$$

and

$$H\bar{G} = F\Gamma + M \tag{2.2.35}$$

where the matrix F is an arbitrary specified feedback gain matrix corresponding to the desired state feedback, and the matrix M is an arbitrary specified matrix corresponding to the desired transition matrix of the controller itself.

Proof:

Equation (2.2.34) and equation (2.2.35) are equivalent to

$$H\,[\bar{C} \quad \bar{G}] = [\, F\Phi \quad F\Gamma+M \,]. \tag{2.2.36}$$

By Lemma 2, the matrix $[\bar{C} \quad \bar{G}]$ has full rank, therefore equation (2.2.36) has a matrix solution H.

Note that it is desirable to use a stable controller from the viewpoint of sensitivity to disturbances. Since the stability of the controller itself is determined by matrix M, it is appropriate to select eigenvalues of M within a unit circle to obtain a stable controller.

2.3 Multirate sampled-data controller with multiplexed outputs

Since the controller mentioned in section 2.2 requires simultaneous sampling of some outputs, simultaneous updating of all control input signals and non-uniform sampling, the hardware implementation must allow for these requirements. By multiplexing input and output samples, these difficulties can be overcome. Therefore, the hardware implementation of the multirate sampled-data controller used in this thesis is mainly composed of a microcomputer which is a single-processor computer, and a multiplexed A/D and D/A converter interfacing unit. The details of the hardware implementation are given in Chapter III. For the above reason, the multirate output sampling mechanism in the last section cannot be used. In this section, the multirate output sampling mechanism is modified to include multiplexing of output samples for a single input and multiple outputs plant; two designs of the multirate sampled-data controller are presented based on the multirate sampled-data theorems in reference [1]. The multirate output sampling mechanism for both multiplexed inputs and outputs will be discussed in the next section.

The modified multirate output sampling mechanism in this case involves detecting the 1^{st}, 2^{nd},...., p^{th} plant outputs $N_1, N_2, \ldots, N_p$ times respectively every uniform sampling period T. Therefore, the sampled-data outputs are

given by

$$\begin{aligned}
y_1(kT_o) &= C_1x(kT_o) \\
&\vdots \\
y_1(kT_o+\mu_1T) &= C_1x(kT_o+\mu_1T)
\end{aligned} \Bigg] \; N_1 \text{ times}$$

$$\begin{aligned}
y_2(kT_o+(\mu_1+1)T) &= C_2x(kT_o+(\mu_1+1)T) \\
&\vdots \\
y_2(kT_o+\mu_2T) &= C_2x(kT_o+\mu_2T)
\end{aligned} \Bigg] \; N_2 \text{ times}$$

$$\vdots$$

$$\begin{aligned}
y_p(kT_o+\mu_pT) &= C_px(kT_o+\mu_pT) \\
&\vdots \\
y_p(kT_o+(N-1)T) &= C_px(kT_o+(N-1)T)
\end{aligned} \Bigg] \; N_p \text{ times}$$

or

$$y_i(kT_o+\mu T) = C_ix(kT_o+\mu T) \tag{2.3.1}$$

where $i = 1,\ldots,p$

and $\mu = 0,\ldots,\mu_1,\mu_1+1,\ldots,\mu_2,\mu_2+1,\ldots,\mu_p,\ldots,N-1.$

In this case, T and T_o are related by

$$T = T_o/N \tag{2.3.2}$$

and

$$N = \sum_{i=1}^{p} N_i. \tag{2.3.3}$$

The above mechanism is a multirate output sampling mechanism for multiplexed outputs. A typical situation is shown in Figure 2.3.1 as an example.

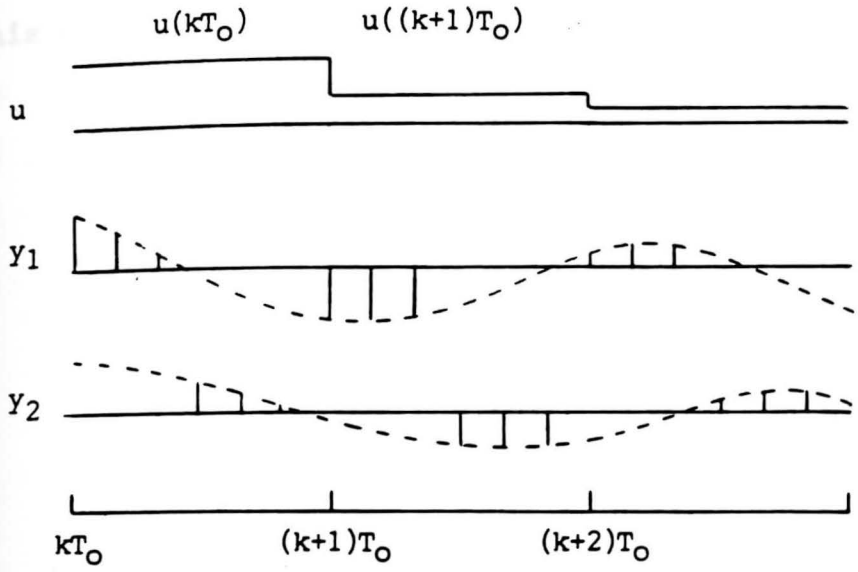


Figure 2.3.1 Multirate output sampling mechanism for a 1 input and 2 outputs plant with $N_1=3$ and $N_2=3$.

Putting $\mu T=T$ into equation (2.2.6), the state at the μT^{th} sampling period is

$$x(kT_o+\mu T) = \exp(A\mu T)x(kT_o) + \left(\int_o^{\mu T} \exp(At)B\,dt\right)u(kT_o). \quad (2.3.4)$$

Substituting equation (2.3.4) into equation (2.3.1) yields

$$y_i(kT_o+\mu T) = C_i\exp(A\mu T)x(kT_o) + C_i\left(\int_o^{\mu T} \exp(At)B\,dt\right)u(kT_o) \quad (2.3.5)$$

where i=1,...,p. Writing equation (2.3.5) in matrix form

gives

$$\bar{Y}(kT_o) = \bar{C}x(kT_o) + \bar{G}u(kT_o). \tag{2.3.6}$$

In this case

$$\bar{Y}(kT_o) = \begin{bmatrix} y_1(kT_o) \\ \vdots \\ y_1(kT_o+\mu_1 T) \\ \vdots \\ y_p(kT_o+\mu_p T) \\ \vdots \\ y_p(kT_o+(N-1)T) \end{bmatrix}, \tag{2.3.7}$$

$$\bar{C} = \begin{bmatrix} C_1 \\ \vdots \\ C_1 \exp(A\mu_1 T) \\ \vdots \\ C_p \exp(A\mu_p T) \\ \vdots \\ C_p \exp(A(N-1)T) \end{bmatrix}, \tag{2.3.8}$$

and

$$\bar{G} = \begin{bmatrix} 0 \\ \vdots \\ C_1 \int_o^{\mu_1 T} \exp(At) B dt \\ \vdots \\ C_p \int_o^{\mu_p T} \exp(At) B dt \\ \vdots \\ C_p \int_o^{(N-1)T} \exp(At) B dt \end{bmatrix}. \tag{2.3.9}$$

The above equation is a basic formula of multirate sampling mechanism for multiplexed outputs. The multirate sampled-data controller with multiplexed outputs can also be realized by using Theorem 1 and Theorem 2 if the matrix $\bar{C}$ in equation (2.3.8) and the pair of $[\bar{C} \quad \bar{G}]$ still satisfies Lemma 1 and Lemma 2 in the last section. By applying elementary row operations as in the last section to the matrix $\bar{C}$ in equation (2.3.8), the result becomes

$$\bar{C}^{+} = \begin{bmatrix} C_1 \\ C_1\{\exp(AT)-I\}/T \\ \vdots \\ C_1\{\exp(AT)-I\}^{N_1-1}/T^{N_1-1} \\ \vdots \\ C_p\exp(A\mu_p T) \\ C_p\exp(A\mu_p T)\{\exp(AT)-I\}/T \\ \vdots \\ C_p\exp(A\mu_p T)\{\exp(AT)-I\}^{N_p-1}/T^{N_p-1} \end{bmatrix} . \qquad (2.3.10)$$

Deleting appropriate rows, as T_o goes to zero, the limit of equation (2.3.10) becomes

$$\begin{bmatrix} C_1 \\ \vdots \\ C_1A^{n_1-1} \\ \vdots \\ C_p \\ \vdots \\ C_pA^{n_p-1} \end{bmatrix}$$

which is in the same form as equation (2.2.5) and satisfies Lemma 1. Note that the result for the pair $[\bar{C} \quad \bar{G}]$ in

equation (2.3.8) and equation (2.3.9) is similar to Lemma 2.

In this thesis, two designs which use the multirate output sampling mechanism for multiplexed outputs are presented by using the Theorem 1 and Theorem 2 respectively as follows:

Design 1: Suppose that the output multiplicities $(N_1,\ldots,N_p)$ are set to minimum values satisfying (2.2.16) which are equal to OIV of the system as shown below.

$$N_i = n_i \qquad (i=1,\ldots,p) \tag{2.3.11}$$

In this case, the matrix $\bar{C}$ becomes a nonsingular square matrix and the matrix H satifying equation (2.2.32) is uniquely determined by

$$H = F\Phi\bar{C}^{-1}. \tag{2.3.12}$$

Therefore, the matrix M can be found from substituting equation (2.3.12) into equation (2.2.33) as shown below.

$$\begin{aligned} M &= H\bar{G} - F\Gamma \\ &= F\Phi\bar{C}^{-1}\bar{G} - F\Gamma. \end{aligned} \tag{2.3.13}$$

Observe that the stability of the matrix M in the above equation depends on the choice of the matrix F which is obtained from an arbitrary pole assignment or optimal control method because other parameters in equation (2.3.13) are unchanged for selected sampling period T_o.

Design 2: Suppose that the output multiplicities $(N_1,\ldots,N_p)$ are set larger than the minimum values as

$$N_i > n_i$$

and their values satisfy the OIV of the augmented system as follows:

$$N_i = m_i \tag{2.3.14}$$

Then matrix $[\bar{C} \quad \bar{G}]$ becomes a nonsingular square matrix and the matrix H is uniquely determined from equation (2.2.36) by

$$H = [F\Phi \quad F\Gamma+M][\bar{C} \quad \bar{G}]^{-1}. \tag{2.3.15}$$

In this case, the matrix H depends on the matrix M and the matrix F which can be chosen arbitrarily corresponding to the desired state transition matrix and the desired state feedback respectively.

Note that, even though these two designs are used for single input and multiple-output plants due to the hardware implementation in this thesis, they can be used for multiple-input and multiple-output plants if the hardware implementation of a controller can update all control input signals to the plant simultaneously.

2.4 Multirate sampled-data controller with Multiplexd Inputs and Outputs

Recall the controller equation (2.2.24):

$$x(kT_o+T_o) = Mu(kT_o) - H\bar{Y}(kT_o) + N_r r(kT_o)$$

When a plant has multiple inputs and outputs (abbreviated as MIMO plant), not only multiplexing of output samples is needed, but multiplexing of control input signals is also necessary since the control input signals cannot be updated simultaneously. This increases the complexity of the discrete-time equivalent model of the system. This section discusses the multirate output sampling mechanism for both multiplexed inputs and outputs and presents the design based on Theorem 2 in section 2.2.

Consider the state-space model of the analog plant given by equation (2.2.1) and equation (2.2.2):

$$\frac{dx}{dt}(t) = Ax(t) + Bu(t)$$

and

$$y(t) = Cx(t).$$

In order to multiplex both inputs and outputs, the multirate sampling mechanism for each single frame period T_o involves updating each control input with uniform sampling period T, followed by detecting each output N_i times respectively with the same uniform sampling period T. In this case, T and T_o are related by

$$T = T_o/S \qquad (2.4.1)$$

where $$S = m + \sum_{i=1}^{p} N_i \quad ; \quad S = \text{positive integer.} \qquad (2.4.2)$$

To be more specific and make this concept easier to understand, consider a fourth order analog plant model with two inputs and two outputs. It is assumed that the plant is controllable and observable, and

$$\text{rank} \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} = 4+2 = 6$$

which satisfies Lemma 2. The output multiplicities are selected to be $N_1=3$ and $N_2=3$. Therefore, S is determined by equation (2.4.2) as shown below.

$$S = 2 + (3+3) = 8.$$

This gives the relation between the uniform sampling period T and and the frame period T_o. From equation (2.4.1), therefore

$$T = T_o/8 \quad \text{or} \quad T_o = 8T. \qquad (2.4.3)$$

Figure 2.4.1 illustrates the sampling mechanism with multiplexed inputs and outputs.

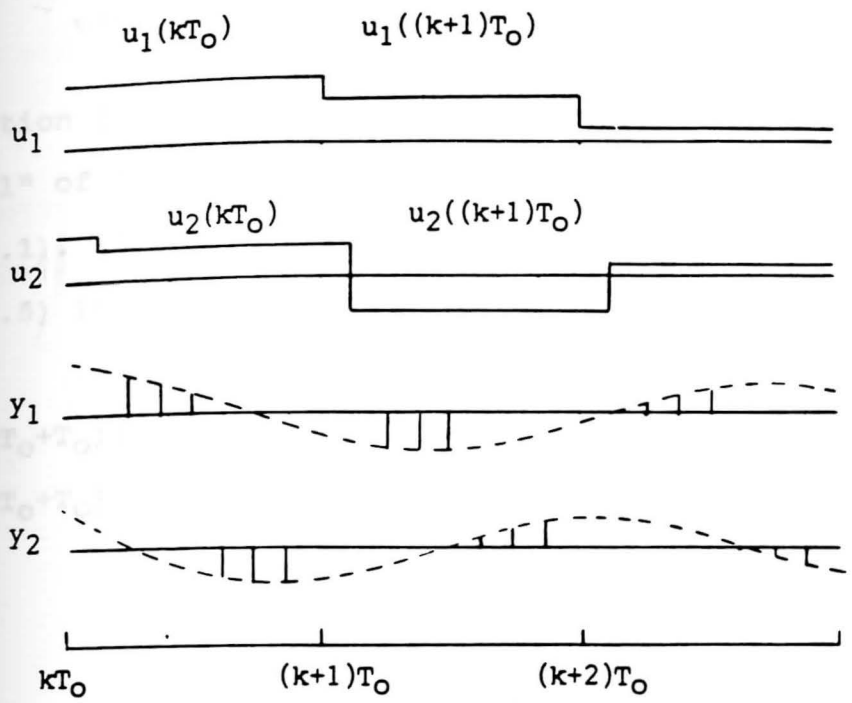


Figure 2.4.1 Multirate output sampling mechanism for a 4th order plant with 2 inputs and 2 outputs with $N_1=N_2=3$.

A zero-order hold equivalent model with a frame period T_o can be developed as follows:

$$x(kT_o+T_o) = \Phi x(kT_o) + \alpha(T_o)b_1u_1(kT_o)$$

$$+ \alpha(7T)b_2u_2(kT_o) + \Phi_{7T}\alpha(T)b_2v(kT_o) \qquad (2.4.4)$$

where $\Phi = \exp(AT_o)$, $\Phi_{iT} = \exp(AiT)$, $\alpha(T) = \int_o^T \exp(At)dt$,

and $v(kT_o)$ is defined as the previous value of $u_2(kT_o)$,

i.e,

$$v(kT_o+T_o) = u_2(kT_o). \tag{2.4.5}$$

Equation (2.4.4) describes the "augmented discrete-time model" of the original discrete-time plant in equation (2.2.1). The matrix form of equation (2.4.4) and equation (2.4.5) is given by

$$\begin{bmatrix} x(kT_o+T_o) \\ v(kT_o+T_o) \end{bmatrix} = \begin{bmatrix} \Phi & \Phi_{7T}\alpha(T)b_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(kT_o) \\ v(kT_o) \end{bmatrix}$$

$$+ \begin{bmatrix} \alpha(T_o) & \alpha(7T)b_2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(kT_o) \\ u_2(kT_o) \end{bmatrix}. \tag{2.4.6}$$

It can be shown that if the analog plant is controllable then this augmented discrete plant is also controllable for almost all sampled periods [9]. From Figure 2.4.1, sampled-data outputs are given by

$$\begin{aligned} y_1(kT_o+2T) &= C_1x(kT_o+2T) \\ y_1(kT_o+3T) &= C_1x(kT_o+2T+T) \\ &= C_1\Phi_T x(kT_o+2T) + C_1\alpha(T)\cdot B\cdot u(kT_o) \\ y_1(kT_o+4T) &= C_1x(kT_o+2T+2T) \\ &= C_1\Phi_{2T}x(kT_o+2T) + C_1\alpha(2T)\cdot B\cdot u(kT_o) \\ y_2(kT_o+5T) &= C_2x(kT_o+2T+3T) \\ &= C_2\Phi_{3T}x(kT_o+2T) + C_2\alpha(3T)\cdot B\cdot u(kT_o) \end{aligned}$$

$$y_2(kT_o+6T) = C_2x(kT_o+2T+4T)$$
$$= C_2\Phi_{4T}x(kT_o+2T) + C_2\alpha(4T)\cdot B\cdot u(kT_o)$$
$$y_2(kT_o+7T) = C_2x(kT_o+2T+5T)$$
$$= C_2\Phi_{5T}x(kT_o+2T) + C_2\alpha(5T)\cdot B\cdot u(kT_o).$$

Therefore, a basic formula of multirate output sampling mechanism for multiplexed inputs and outputs becomes

$$\bar{Y}(kT_o) = \bar{C}x(kT_o+2T) + \bar{G}u(kT_o) \tag{2.4.7}$$

where

$$\bar{Y}(kT_o) = \begin{bmatrix} y_1(kT_o+2T) \\ y_1(kT_o+3T) \\ y_1(kT_o+4T) \\ y_2(kT_o+5T) \\ y_2(kT_o+6T) \\ y_2(kT_o+7T) \end{bmatrix}, \tag{2.4.8}$$

$$\bar{C}(kT_o) = \begin{bmatrix} C_1 \\ C_1\Phi_T \\ C_1\Phi_{2T} \\ C_2\Phi_{3T} \\ C_2\Phi_{4T} \\ C_2\Phi_{5T} \end{bmatrix}, \tag{2.4.9}$$

and

$$\bar{G}(kT_o) = \begin{bmatrix} 0 \\ C_1\Gamma_T \\ C_1\Gamma_{2T} \\ C_2\Gamma_{3T} \\ C_2\Gamma_{4T} \\ C_2\Gamma_{5T} \end{bmatrix}. \tag{2.4.10}$$

In equation (2.4.9), $\Phi_{iT} = \exp(AiT)$; and in equation (2.4.10), $\Gamma_{iT} = \alpha_{iT}B$. Note that the vector of output

samples $\bar{Y}(kT_o)$ is expressed in terms of the state at time kT_o+2T because of the delay in taking samples. This will affect the multirate sampled-data control law as shown next.

The state variables feedback control law for the augmented system is

$$u(kT_o) = - [F \quad G] \begin{bmatrix} x(kT_o) \\ v(kT_o) \end{bmatrix} + N_r r(kT_o) \qquad (2.4.11)$$

where [F G] is the state variable feedback gain matrix obtained by pole assignment or optimal control method. From equation (2.4.11), the control for the next frame period becomes

$$u(kT_o+T_o) = - Fx(kT_o+T_o) - Gv(kT_o+T_o) + N_r r(kT_o+T_o) \qquad (2.4.12)$$

since
$$v(kT_o+T_o) = u_2(kT_o)$$

$$= \beta u(kT_o) \qquad (2.4.13)$$

where β=[0 1].

It is appropriate to express $x(kT_o+T_o)$ in term of $x(kT_o+2T)$ because all control inputs are held constant from this time until the next frame period. Therefore

$$x(kT_o+T_o) = \Phi_{6T} x(kT_o+2T) + \Gamma_{6T} u(kT_o). \qquad (2.4.14)$$

Assume the reference input is constant, $r(kT_o+T_o)=r(kT_o)$.

Hence, substituting equation (2.4.13) and equation (2.4.14) into equation (2.4.12) gives

$$\begin{aligned} u(kT_o+T_o) &= -F\{\ \Phi_{6T}x(kT_o+2T) + \Gamma_{6T}u(kT_o)\ \} - G\beta u(kT_o) \\ &\quad + N_r r(kT_o) \\ &= -F\Phi_{6T}x(kT_o+2T) - \{\ F\Gamma_{6T} + G\beta\ \}u(kT_o) \\ &\quad + N_r r(kT_o). \end{aligned} \tag{2.4.15}$$

Now, substituting equation (2.4.7) into the multirate sampled-data control law in equation (2.2.24) yields

$$\begin{aligned} u(kT_o+T_o) &= Mu(kT_o) - H\{\ \bar{C}x(kT_o+2T) + \bar{G}u(kT_o)\ \} \\ &\quad + N_r r(kT_o) \\ &= Mu(kT_o) - H\bar{C}x(kT_o+2T) - H\bar{G}u(kT_o) \\ &\quad + N_r r(kT_o). \end{aligned} \tag{2.4.16}$$

As before, equation (2.4.15) is equivalent to equation (2.4.16) if and only if

$$\begin{aligned} -F\Phi_{6T}x(kT_o+T_o) - (F\Gamma_{6T}+G\beta)u(kT_o) &= Mu(kT_o) - HCx(kT_o+2T) \\ &\quad - HGu(kT_o) \end{aligned}$$

for every $x(kT_o+2T)$ and $u(kT_o)$. In terms of matrices,

$$H[\bar{C}\quad \bar{G}] = [\ F\Phi_{6T} \qquad F\Gamma_{6T}+G\beta+M\ ]. \tag{2.4.17}$$

Since $[\bar{C}\quad \bar{G}]$ is a nonsingular square matrix by using Design 2, therefore the matrix H is determined by

$$H = [\ F\Phi_{6T} \qquad F\Gamma_{6T}+G\beta+M\ ][\bar{C}\quad \bar{G}]^{-1} \tag{2.4.18}$$

where the matrix M is an arbitrarily specified matrix

corresponding to the desired transition matrix of the controller itself. Note that the final form of the multirate sampled-data controller equation is the same form as in equation (2.2.24).

2.5 Minimizing disturbance effects

Practically, when the controller is implemented, it is unavoidable to have disturbances which are not accessible for control. Examples of disturbances are errors caused by hardware equipment or delays in command execution, and noise from a plant or quantization. These are undesirable inputs that will disturb the controller and degrade the system performance. This section discusses how to choose the control parameters of the multirate sampled-data controller that will minimize the disturbance effects to the system performance.

Again, consider the multirate sampled-data control law in equation (2.2.24). Since the matrix H is the input gain matrix of the controller, the sampled-data output vector $\bar{Y}(kT_o)$ is amplified by the matrix H. Disturbances will also be amplified into the controller. Ordinarily, disturbance signals are small and may not affect the system performance. However, if entries in the matrix H are significantly large, disturbances are enlarged and become severe to the system performance. Therefore, it is desirable to select control parameters of the designs

presented in sections 2.3 and 2.4 to obtain a suitable small H matrix to minimize the disturbance effects as explained below.

In Design 1, since the matrix H is directly determined by equation (2.3.12), it will depend directly on the choice of the state variable feedback gain matrix F. Therefore, not only is the matrix F chosen to obtain the desired state transition matrix M, it also must be chosen in order to keep the matrix H small.

In Design 2, since selections of the state transition matrix M and the state variable feedback gain matrix F affect directly the matrix H obtained from equation (2.3.15) or equation (2.4.18), these two parameters should be considered carefully. For a given matrix F, a choice of the suitable matrix M to minimize the matrix H can be considered by the following procedure.

Recall equation (2.3.15):

$$H = [F\Phi \quad F\Gamma+M][\bar{C} \quad \bar{G}]^{-1}. \tag{2.5.1}$$

Letting

$$[\bar{C} \quad \bar{G}]^{-1} = \begin{bmatrix} R_x \\ R_u \end{bmatrix}_{(n+m)\times(n+m)} \tag{2.5.2}$$

where R_x is the submatrix containing the first n rows,

R_u contains the last m rows.

Note that n and m are the number of states and inputs of the plant, respectively. Substituting equation (2.5.2) into equation (2.5.1) gives

$$H = F\Phi R_x + (F\Gamma+M)R_u$$

$$= (F\Phi R_x + F\Gamma R_u) + MR_u$$

$$= H_o + MR_u. \qquad (2.5.3)$$

Observe that $H = H_o$ when $M = 0$. To find the minimum H matrix, consider the trace of a square matrix and some of its properties. The trace of a square is defined to be the sum of the entries on the main diagonal, that is, for a square matrix A

$$tr(A) = \sum_{i=1}^{n} a_{ii}.$$

Some properties of trace of any square matrix are as follows:

1. If A is a square matrix, then

$$tr(A) = tr(A') \qquad (2.5.4)$$

where A' is the transpose of a matrix A.

2. If A and B are square matrices, then

$$tr(A+B) = tr(A) + tr(B). \qquad (2.5.5)$$

3. If A is m x n matrix and B is n x m matrix, then (AB) is m x m matrix,

and

$$tr(AB) = \sum_{i=1}^{m} (AB)_{ii}$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij}b_{ji} \qquad (2.5.6)$$

where a_{ij} and b_{ij} are entries in A and B, respectively.

4. The partial derivative of the trace with respect to the matrix A is [13]

$$\frac{\delta}{\delta A} tr(AB) = B. \qquad (2.5.7)$$

From equation (2.5.6), since H is m x (n+m) matrix, therefore

$$tr(HH') = \sum_{i=1}^{m} (HH')_{ii} = \sum_{i=1}^{m} \left(\sum_{j=1}^{n+m} h_{ij}h_{ij} \right)$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{n+m} h_{ij}^2$$

$$= \text{Sum of squares of elements of H} \qquad (2.5.8)$$

where H' is the transpose of the matrix H.

Next, consider the linear algebraic equation (2.5.3) and find M to minimize tr(HH') in order to obtain the minimum H matrix. Since

$$HH' = (H_o + MR_u)(H_o' + R_u'M')$$

$$= H_oH_o' + MR_uH_o' + H_oR_u'M' + MR_uR_u'M'$$

$$= H_oH_o' + MR_uH_o' + (MR_uH_o')' + MR_uR_u'M', \tag{2.5.9}$$

therefore

$$\begin{aligned} tr(HH') &= tr(H_oH_o' + MR_uH_o' + (MR_uH_o')' + MR_uR_u'M') \\ &= tr(H_oH_o') + tr(MR_uH_o') + tr(MR_uH_o')' \\ &\quad + tr(MR_uR_u'M') \\ &= tr(H_oH_o') + 2\cdot tr(MR_uH_o') + tr(MR_uR_u'M'). \end{aligned} \tag{2.5.10}$$

Defining f as the scalar function of the (HH') matrix and

$$f = tr(HH'),$$

therefore

$$f = tr(H_oH_o') + 2\cdot tr(MR_uH_o') + tr(MR_uR_u'M'). \tag{2.5.11}$$

The partial derivative of f with respect to the M matrix is given by

$$\begin{aligned} \frac{\delta f}{\delta M} &= \frac{\delta}{\delta M} tr(HH') \\ &= \frac{\delta}{\delta M} tr(H_oH_o') + 2\frac{\delta}{\delta M} tr(MR_uH_o') + \frac{\delta}{\delta M} tr(MR_uR_u'M'). \end{aligned} \tag{2.5.12}$$

Considering the first term of equation (2.5.12), $tr(H_oH_o')$ is a constant value respect to M, therefore

$$\frac{\delta}{\delta M} tr(H_oH_o') = 0. \tag{2.5.13}$$

Considering the second term of equation (2.5.12), the property in equation (2.5.7) gives

$$2\frac{\delta}{\delta M}\mathrm{tr}\{M(R_u H_o')\} = 2R_u H_o'. \qquad (2.5.14)$$

Considering the third term of equation (2.5.12) and by given Q as

$$Q = R_u R_u', \qquad (2.5.15)$$

therefore

$$\mathrm{tr}(MR_u R_u' M') = \mathrm{tr}(MQM'). \qquad (2.5.16)$$

Since it can be derived that

$$\frac{\delta}{\delta M}\mathrm{tr}(MQM') = (Q + Q')M' \qquad (2.5.17)$$

and

$$Q = Q',$$

therefore

$$\frac{\delta}{\delta M}\mathrm{tr}(MQM') = 2QM'$$

$$= 2R_u R_u' M'. \qquad (2.5.18)$$

Substituting equation (2.5.13), equation (2.5.14), and equation (2.5.18) into equation (2.5.12) gives

$$\frac{\delta f}{\delta M} = 2R_u H_o + 2R_u R_u' M'. \qquad (2.5.20)$$

The tr(HH') is minimized when $\frac{\delta f}{\delta M} = 0$,

therefore

$$0 = 2\{R_u H_o' + R_u R_u' M'\}$$

or
$$R_u R_u' M' = -R_u H_o'. \tag{2.5.21}$$

The transpose of the above equation is written as

$$M R_u R_u' = -H_o R_u'. \tag{2.5.22}$$

Multiplying $(R_u R_u')^{-1}$ to the right of both sides in equation (2.5.22) gives

$$M = - H_o R_u' (R_u R_u')^{-1}. \tag{2.5.23}$$

Thus, the matrix M obtained from equation (2.5.23) minimizes tr(HH'), which is the sum of the squares of the elements of matrix H, and therefore keeps the controller gains small. Finally, the matrix H minimum can be calculated from equation (2.2.15) by using the result of the matrix M from equation (2.5.23). Note that the result of the matrix M to minimize the matrix H in section 2.4 is the same form as equation (2.5.23). The matrix H minimum is then calculated by equation (2.4.18).

CHAPTER III

REAL TIME IMPLEMENTATION

3.1 Introduction

The multirate sampled-data controller with multiplexed inputs and outputs is implemented on the computer hardware and software available in the control laboratory. Section 3.2 presents an overview of the computer hardware which is composed of the analog computer used to simulate plant models and the digital computer used to implement control algorithms. Also, the interfacing unit is discussed in some detail. Section 3.3 explains the computer software used to implement control algorithms including flow charts.

3.2 Computer Hardware

The hardware implementation in this thesis is shown in Figure 3.2.1. The following subsections give brief explanations of various parts of the hardware.

3.2.1 Analog computer

The COMDYNA GP-6 analog computer is used to simulate plant models. The details of operator functions and operating procedures can be found in GP-6 Analog Computer Manual [14].

3.2.2 Digital computer

The EVEREX 286 (IBM PC/AT compatible) computer is used

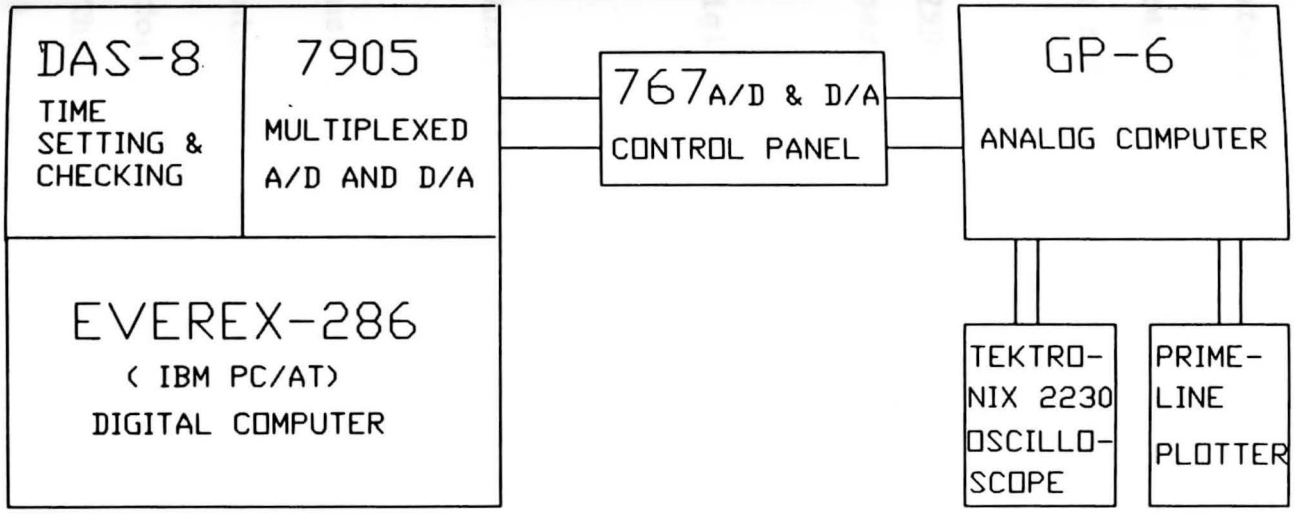


Figure 3.2.1 Block diagram of hardware implementation.

to run program CC for purposes of design and to write control programs in the BASIC language. It is also used as a digital controller to control the analog plant built on GP-6. The details of The EVEREX 286 computer can be found in the Reference and User Manual [15].

3.2.3 Interfacing Unit

This section presents three separate parts in some detail.

3.2.3.1 7905 AD/DA Interface board has available functions used to perform multiplexed A/D and D/A conversions as follows:

Analog/Digital Conversion...12-bits sucessive-approximation converter with eight channels, multiplexed inputs.

Digital/Analog Conversion...Three 12-bit Digital to Analog converters.

Logic Sense.....Three input logic sense lines.

Logic Control.....Four latched output logic control lines.

Performing A/D or D/A conversion requires two data bytes. The high byte, data bits D0-D7 are A/D or D/A data bits 4-11. The low byte, data bits D4-D7 are A/D or D/A data bits 0-3. Address locations of 7905 board begin at 310 Hex, where A4, A9 and AEN (Address enable) are fixed as the board code, address bits A0-A3 and the instruction,

either an input (INP) or output (OUT), determines the function to be executed as shown in Table 3.2.1.

TABLE 3.2.1

Function mode locations of 7905 board.

A3	A2	A1	A0	Location Hex(Dec)	I/O	Description
0	0	0	0	310(784)	OUT	Set Mux only.
0	0	0	1	311(785)	OUT	Set Control byte only.
0	0	1	0	312(786)	OUT	Set Mux and starts ADC.
0	0	1	1	313(787)	OUT	also start ADC.
0	1	0	0	314(788)	OUT	Set low byte of LDAC.
0	1	0	1	315(789)	OUT	Set LDAC only.
0	1	1	0	316(790)	OUT	Set l. byte & starts ADC.
0	1	1	1	317(791)	OUT	Set LDAC & starts ADC.
1	0	0	0	318(792)	OUT	Set low byte of RDAC.
1	0	0	1	319(793)	OUT	Set RDAC only.
1	0	1	0	31A(794)	OUT	Set l. byte & start ADC.
1	0	1	1	31B(795)	OUT	Set RDAC & start ADC.
1	1	0	0	31C(796)	OUT	Set low byte of VDAC.
1	1	0	1	31D(797)	OUT	Set VDAC only.
1	1	1	0	31E(798)	OUT	Set l. byte & start ADC.
1	1	1	1	31F(799)	OUT	Set VDAC & start ADC.
0	0	1	0	312(786)	IN	Read ADC data low byte.
0	0	1	1	313(787)	IN	Read ADC data high byte.

In mode sense, data bits D0-D2 in the low byte are logic sense bits C0-C2. In this case, C2 is fixed to monitor the analog computer mode logic. Data bit D3 is End of Conversion (EOC) sense logic. When EOC is low, the A/D conversion is in progress and ADC data is not ready. When EOC is high, ADC data is ready. In mode control, the logic control bit C3, when C5 is low, is used to pull the GP-6 OP bus from an operation state to an initial condition state. Table 3.2.2 describes logic conditions.

TABLE 3.2.2

Logic conditions of GP-6.

GP-6 Control Push Button	OP bus state	Logic control C5	Logic control C3	Logic sense C2
OP	IC	low	high	low
OP	OP	low	low	high

Note that GP-6 control push button must be in the OP (operation) position to control GP-6 modes from the 7905 board. The details of the 7905 board can also be found in the Reference Manual [16].

3.2.3.2 COMDYNA 767 A/D & D/A Control Panel organizes the control system by interconnecting its component parts as seen from Figure 3.2.1. Operation of the 767 requires the connection of two cables; 7905 cable and GP-6 cable. Figure 3.2.2 shows a schematic of interconnections between GP-6, 767 panel and 7905 board. The following covers necessary details of the 767 panel as seen in Figure 3.2.3.

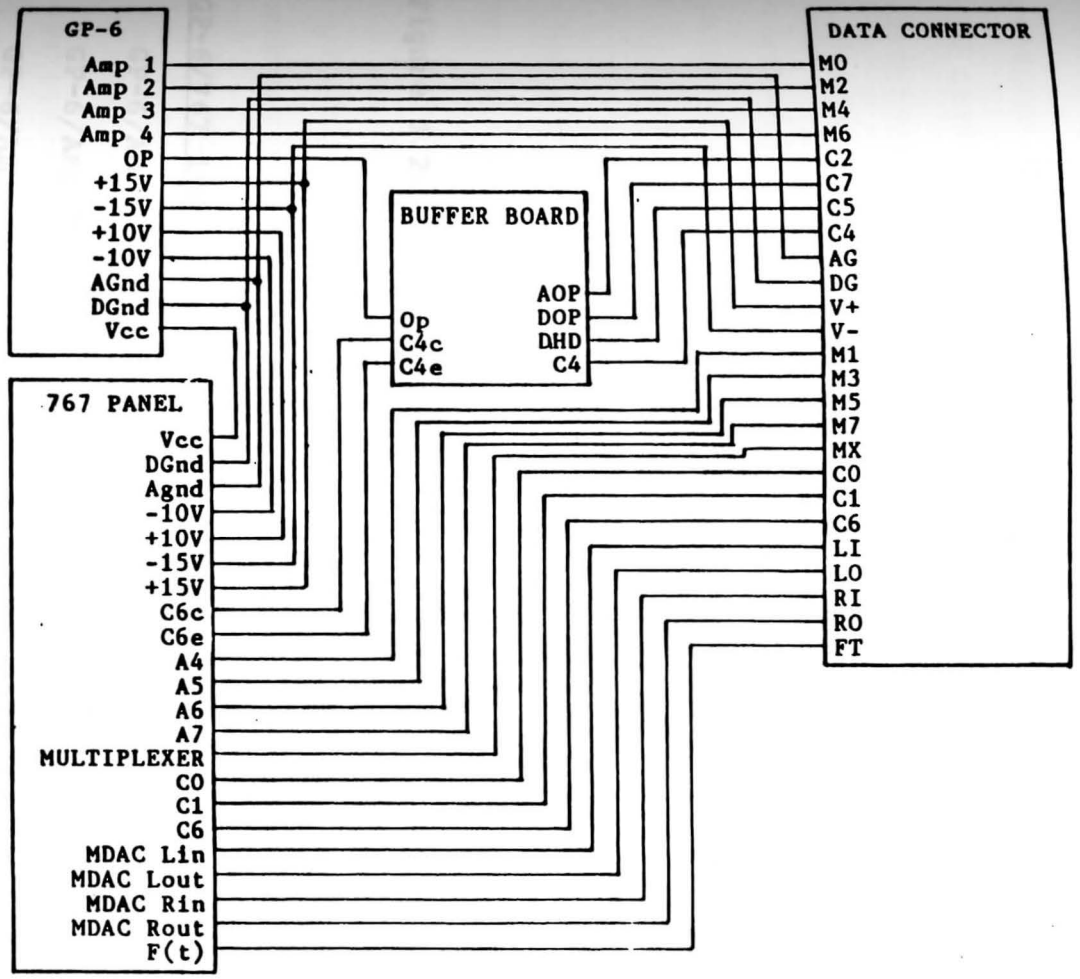


Figure 3.2.2 Schematic diagram of hardware interconnections [16].

A/D Conversion..... Table 3.2.3 shows relations of the 7905 multiplexer addresses, 7905 connector terminations and the 767 inputs, which consist of four trunked GP-6 amplifier outputs and four 767 patch panel inputs.

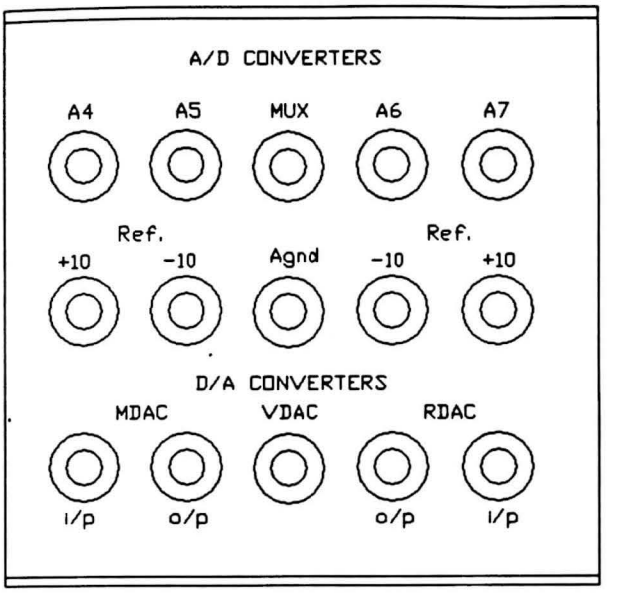


Figure 3.2.3 A/D and D/A section of 767 control panel.

TABLE 3.2.3

Multiplexer addresses of 7905.

GP-6/767 locations	Multiplexer Addr.	7905 connector
GP-6/Amp#1	00	M0
GP-6/Amp#2	02	M2
GP-6/Amp#3	04	M4
GP-6/Amp#4	06	M6
767 A4	08	M1
767 A5	0A	M3
767 A6	0C	M5
767 A7	0E	M7

Multiplexer..... The multiplexer is the output of the 7905 multiplexer (MX at the data connector in Figure 3.2.2) which is also input to the A/D converter.

D/A Conversion..... "LDAC" and "RDAC" are multiplying D/A converters. Each attenuates a patched analog input and produces an analog output that is the input multiplied by the digital data word setting. "VDAC" has the same circuitry as LDAC and RDAC only the input is fixed at a +10 volts reference rather than a patched variable. In this thesis, +10 volts reference is also used for LDAC and RDAC patched inputs.

3.2.3.3 DAS-8 Interface board is chosen to perform sampling time setting and to check whether the sampling time is over or not, which are required to run the control algorithms in real time. Four operation modes are used as follows:

1. Mode 0 (Initialize DAS-8 board) is used to set base address 300Hex (as default) for DAS-8 board. This mode has to be set before using other modes.

2. Mode 10 (Configure DAS-8 timer/counter) is used to configure the DAS-8 timer/counter which is the advanced Intel 8254 timer/counter providing 3 x 16-bit count down registers.

The counter #0 and counter #2 are set to operate configuration 0 and 3 respectively. Brief explanations for these two configurations are given as follows:

configuration 0 .....Pulse on terminal count.

After loading the counter, the output goes low. Counting is enabled when the gate input is high (forces the output to go low) and continues until the count reaches zero. Then the output will go high and remain high until the counter is reloaded by a programmed command.

configuration 3 .....Square wave generator (N count).

After loading the counter, the output goes high for half the count and low for the other half. If N is even, a symmetrical square wave output is obtained. If N is odd, the output is high for (N+1)/2 counts and low for (N-1)/2 counts.

3. Mode 11 (Load timer/counter) is used to start the selected timer counter from mode 10.

4. Mode 13 (Read digital inputs IP1-3) is used to read state of digital inputs. By wiring the output of counter #0 to the IP1 and using programmed commands, sampling time checking can be achieved.

Concerning the sampling time setting, counter #2 configuration 3 is set using mode 10 and started to generate a square wave by loading mode 11 with count number. It is appropriate to set count number = 378 to obtain frequency ≈ 10 kHz which is fast enough to use as an input clock for counter #0. Then the counter #0

configuration 0 in mode 10 is set and started by loading mode 11 with counter number suitable for each desired sampling period. The output of counter #0 is forced to go low, the counter #0 starts to count down as soon as the next coming clock input arrives and it continues until the count reaches zero, then the output of counter #0 goes high. Mode 11 is reloaded again with the same count number to start the next sampling period.

Concerning sampling time checking, after calculations in each sampling period, IP1 is checked by program commands. If it goes high before calculations are finished, sampling time is too short. The count number for counter #0 configuration 0 has to be reset.

The timing diagram for sampling time setting is shown in Figure 3.2.4. Figure 3.2.5 shows the wiring diagram required for the above operation modes. More details of operation modes can also be found in DAS-8 User's Manual [17].

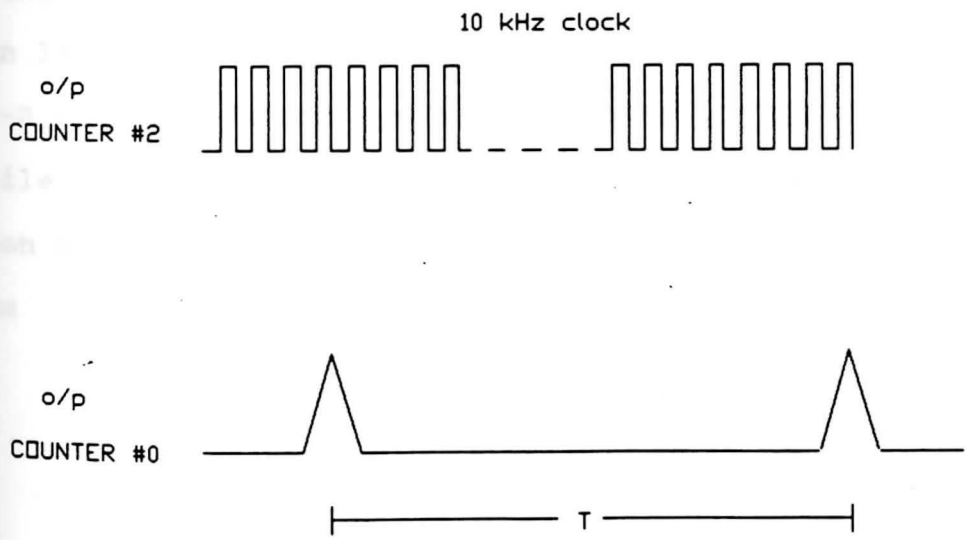


Figure 3.2.4 Timing diagram of uniform sampling period T.

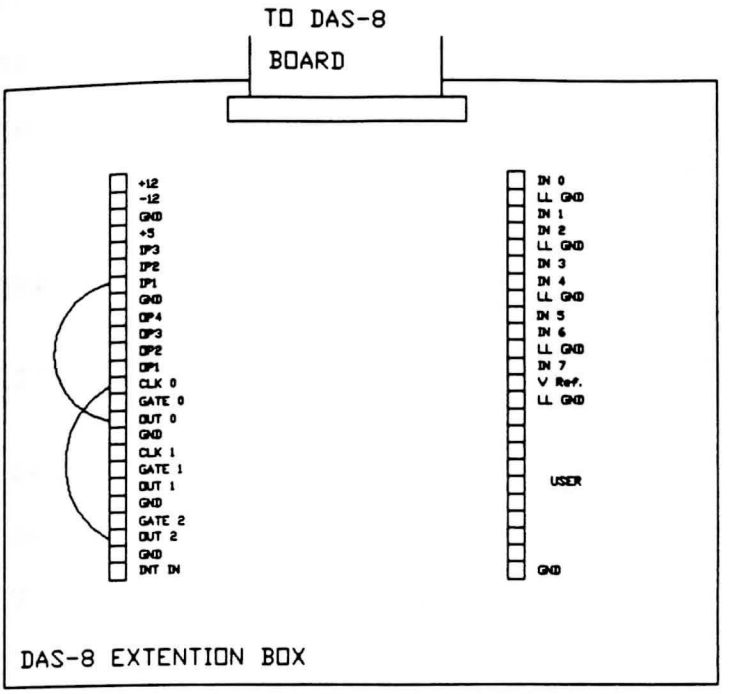


Figure 3.2.5 Wiring diagram required for DAS-8.

3.3 Computer Software

The control program is written in Micro-soft Advanced BASIC language (BASICA), compiled by the BASIC compiler, and then linked to "DAS8.OBJ" Assembly program in order to use DAS-8 operation modes by CALL statements. Explanations of compile and link programs can be seen in MS-DOS Operation Manual [19]. The format of CALL statements is in the form

```
CALL DAS8 (MD%,IP%,FLAG%)
```

where MD% is mode number,

IP% is data such as count number,

and Flag% is error flag.

Operation modes of the 7905 board is selected by OUT or INP statement with specified address given in section 3.2.3.1. Brief explanations for the control program are in steps as follows:

Main program

Step 1. Initialize DAS-8 set address to 300 Hex.

Step 2. Set counter #2 with configuration 3 for generating square wave used as a real time clock for step 3. Then, load the counter with count number equal to 378 to start clock at frequency ≈ 10 kHz.

Step 3. Set counter #0 with configuration 0 preparing for the output sampling period count.

Step 4. Initialize the control inputs, set GP-6 to initial condition mode and Set A/D input address to the 1^{st} output y_1 of the plant.

Step 5. Set parameters in mode 11 and mode 13 and select a count number for the output sampling period T. Also, select the reference input if required.

Step 6. Set GP-6 to operation mode and perform the control algorithm by calling MRC subroutine.

Note that the program is loop running at step 6 until the "F1" key is pushed to end the program.

Subroutine

1. **MRC**: This subroutine is composed of control algorithms for the multirate sampled-data control law in equation (2.2.24),

$$u(kT_o+T_o) = M \cdot u(kT_o) - H \cdot \bar{Y}(kT_o) + N_r \cdot r(kT_o).$$

For the multiplexed output mentioned in section 2.3, MRC has algorithms started from the beginning of the kT_oth frame period as shown below.

kT_oth
- Start counter #0, config. 0 by using Call DAS8 (mode 11) to count for sampling period T.
- Update control input and perform A/D conversion simutaneously. (It is done by one OUTPUT command mentioned in section 3.2.3.1).
- Obtain the sampled-data output $y_1(kT_o)$.
- Transform into 2's complement word and scale to ± 10 volts range by using subroutine SCALE-DATA.
- Multiply $y(kT_o)$ to the related element of the H matrix,

$$Y_k = h_{11} * y_1(kT_o).$$

- Check whether there is sufficient time between samples or not by calling subroutine CHECK-TIMING.

(kT_o+T)th
- Start counter #0 for sampling period T.
- Perform A/D conversion.

- Obtain $y_1(kT_o+T)$.
- Call SCALE-DATA.
- Multiply $y_1(kT_o+T)$ to the related element of the H matrix and add to the last subinterval period,

$$Y_k = h_{12} * y_1(kT_o+T) + Y_k.$$

- Call CHECK-TIMING.

$(kT_o+2T)^{th}$ to $(kT_o+(N-2)T)^{th}$: Obtain $y_i(kT_o+\mu T)$ for i=1,...,p by using the same algorithm as at $(kT_o+T)^{th}$ sampling period.

Note that for multiple outputs plant, the A/D input address is set to the next output before calling subroutine CHECK-TIMING during the last subinterval of the present output.

$(kT_o+(N-1)T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_i(kT_o+(N-1)T)$.
- Call SCALE-DATA.
- $Y_k = h_{1N} * y_i(kT_o+(N-1)T) + Y_k$.
- Calculate the control input for the $(kT_o+T_o)^{th}$ frame period,

$$U_k = M \cdot U_k - Y_k + N_r \cdot r_k.$$

- Separate U_k to high and low bytes.
- Set A/D input address to the first channel.
- Call CHECK-TIMING.

For multiplexed inputs and outputs mentioned in section 2.4, MRC has algorithms shown below.

$(kT_o)^{th}$

- Start counter #0 for sampling period T.
- Update control input u_1.
- Calculate control input u_2,

$$U_{2k} = m_{21} \cdot U_{1k} + m_{22} \cdot U_{2k} - Y_{2k} \cdot$$

- Separate u_2 to high and low bytes.
- Call CHECK-TIMING.

$(kT_o+T)^{th}$

- Start counter #0 for sampling period T.
- Update control input u_2.
- Set A/D input channel to the first address.
- Call CHECK-TIMING.

$(kT_o+2T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_1(kT_o+2T)$.
- Call SCALE-DATA.
- Multiply $y(kT_o+2T)$ to the related element of the H matrix,

$$Y_{1k} = h_{11} * y_1(kT_o+2T)$$

$$Y_{2k} = h_{21} * y_1(kT_o+2T).$$

- Call CHECK-TIMING.

$(kT_o+3T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_1(kT_o+3T)$.

- Call SCALE-DATA.
- Multiply $y_1(kT_o+3T)$ to the related element of the H matrix and add to the prior value,

$$Y_{1k} = h_{12}*y_1(kT_o+3T) + Y_{1k}$$

$$Y_{2k} = h_{22}*y_1(kT_o+3T) + Y_{2k}.$$

- Call CHECK-TIMING.

$(kT_o+4T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_1(kT_o+4T)$.
- Call SCALE-DATA.
- Multiply $y_1(kT_o+4T)$ to the related element of the H matrix and add to the prior value,

$$Y_{1k} = h_{13}*y_1(kT_o+4T) + Y_{1k}$$

$$Y_{2k} = h_{23}*y_1(kT_o+4T) + Y_{2k}.$$

- Set A/D input channel to the 2^{nd} output address.
- Call CHECK-TIMING.

$(kT_o+5T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_2(kT_o+5T)$.
- Call SCALE-DATA.
- Multiply $y_2(kT_o+5T)$ to the related element of the H matrix and add to the prior value,

$$Y_{1k} = h_{14} * y_2(kT_o+5T) + Y_{1k}$$

$$Y_{2k} = h_{24} * y_2(kT_o+5T) + Y_{2k}.$$

- Call CHECK-TIMING.

$(kT_o+6T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_2(kT_o+6T)$.
- Call SCALE-DATA.
- Multiply $y_2(kT_o+6T)$ to the related element of the H matrix and add to the prior value,

$$Y_{1k} = h_{15} * y_2(kT_o+6T) + Y_{1k}$$

$$Y_{2k} = h_{25} * y_2(kT_o+6T) + Y_{2k}.$$

- Call CHECK-TIMING.

$(kT_o+7T)^{th}$

- Start counter #0 for sampling period T.
- Perform A/D conversion.
- Obtain $y_2(kT_o+7T)$.
- Call SCALE-DATA.
- Multiply $y_2(kT_o+7T)$ to the related element of the H matrix and add to the prior value,

$$Y_{1k} = h_{16} * y_2(kT_o+7T) + Y_{1k}$$

$$Y_{2k} = h_{26} * y_2(kT_o+7T) + Y_{2k}.$$

- Calculate control input u_1,

$$U_{1k} = m_{11}U_{1k} + m_{12}U_{2k} - Y_{1k}.$$

- Separate U_{1k} into high and low bytes.
- Set A/D input channel to the first output address.

- Call CHECK-TIMING.

2. **SCALE-DATA**: The result obtained from A/D coversion both high and low bytes are transformed to 2's complement word and scaled to ±10 volts range.

3. **CHECK-TIMING**: Mode 13 of DAS-8 functions is loaded to check whether setting time T is sufficient to perform calculations or not.

Flow charts of the main program and subroutines are shown in Figures 3.3.1-3.3.5.

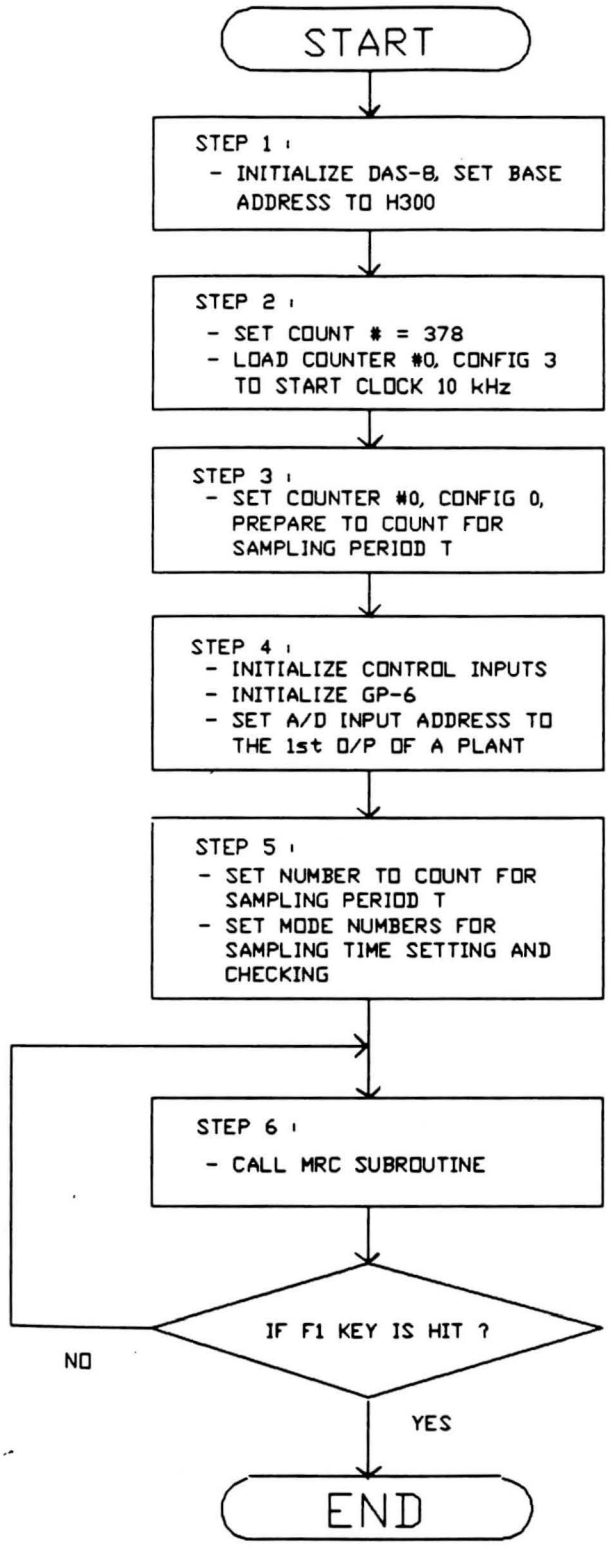


Figure 3.3.1 Flow chart of the main program.

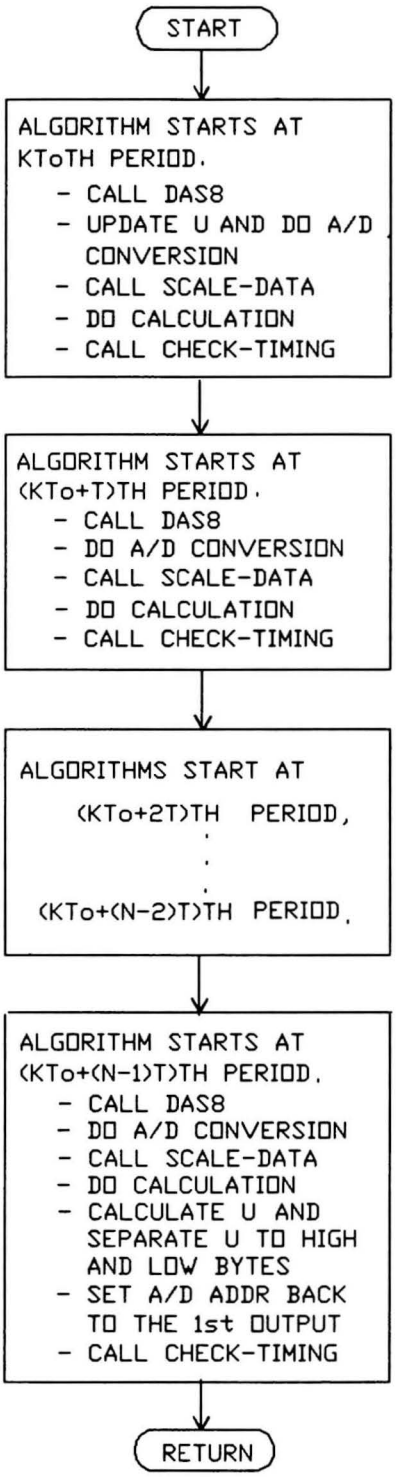


Figure 3.3.2 Flow chart of MRC subroutine for multiplexed outputs.

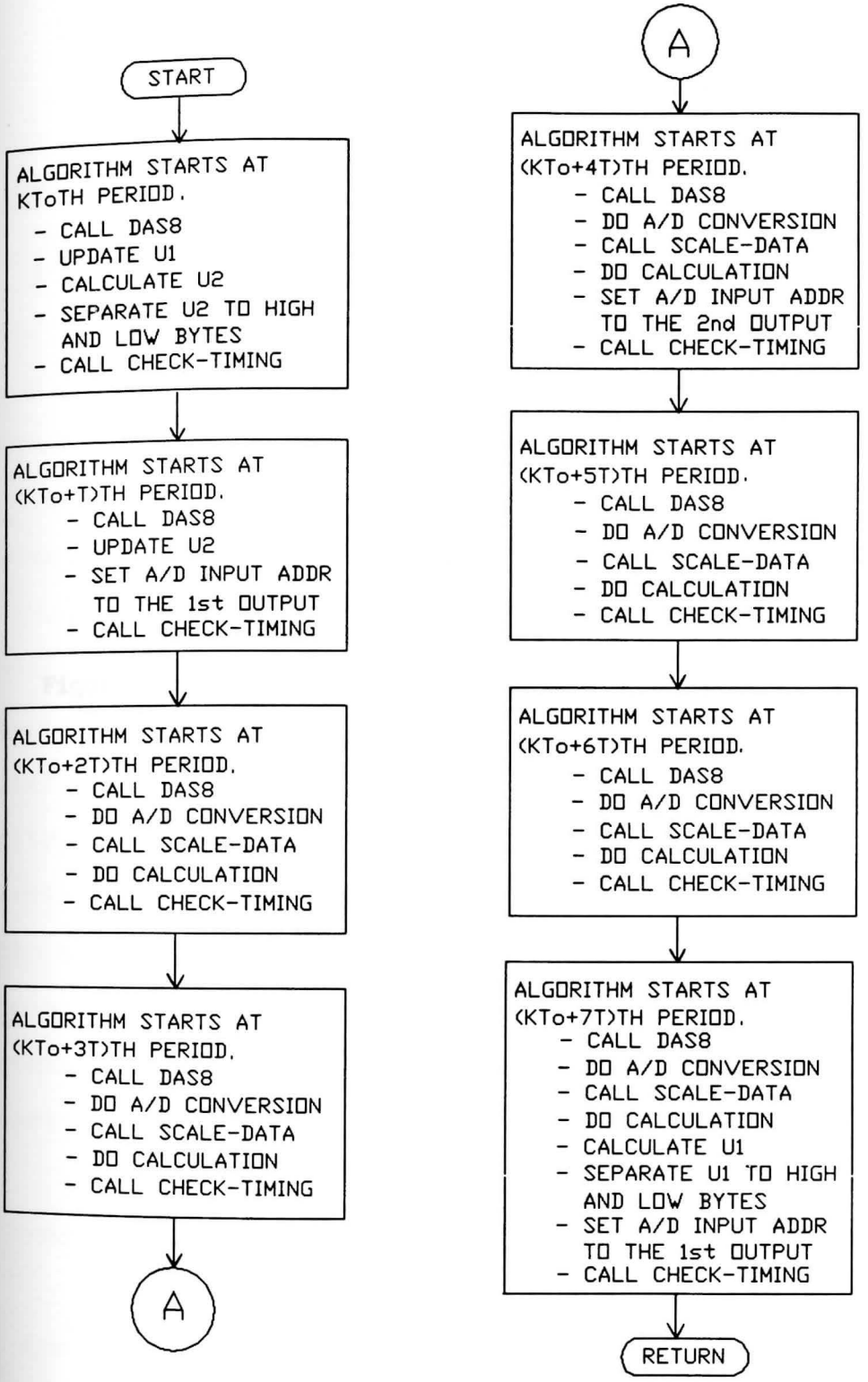


Figure 3.3.3 Flow chart of MRC subroutine for multiplexed inputs and outputs.

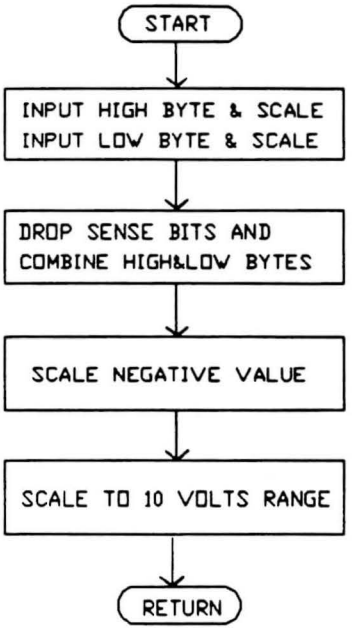


Figure 3.3.4 Flow chart of SCALE-DATA subroutine.

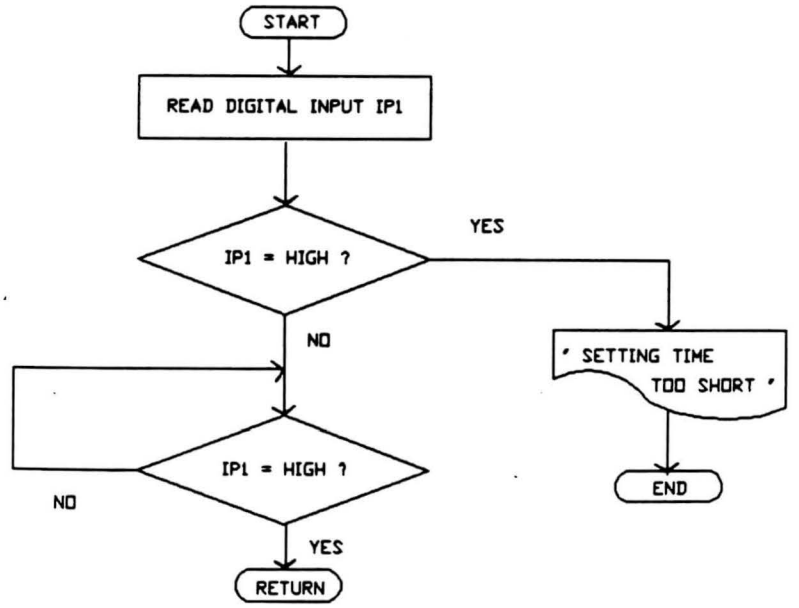


Figure 3.3.5 Flow chart of CHECK-TIMING subroutine.

CHAPTER IV

APPLICATION EXAMPLES

4.1 Introduction

In this chapter, three application examples using multirate sampled-data control law are presented in section 4.2, 4.3 and 4.4 respectively. These are the 2^{nd} order plant with single input single output, the 4^{th} order plant with one input and two outputs, and the 4^{th} order plant with two inputs and two outputs. The first two examples use the multirate sampling mechanism and the designs presented on section 2.3 while the last example uses the mechanism and the design presented in section 2.4 in Chapter II. In section 4.5, the Computer Simulation results of all examples are shown and compared with the Real-time results. An Interactive Computer-aided Control Design software, Program CC [18], is used to help in design calculations and to simulate the results. Listings of CC programs and control programs for all designs are given in Appendices A-E.

4.2 Example 1: The 2nd order plant with 1 input and 1 output

Consider state-space coefficient matrices of the controllable and observable analog plant as follows:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$C = [\ 1 \quad 0\ ], \quad \text{and} \quad D = [0].$$

The opened-loop transfer function of this plant is given by

$$\frac{1}{s(s+1)}.$$

It shows that this plant has no invariant zero at origin and has opened-loop poles at 0 and -1 in the s-plane. Suppose that the servo design for this plant is to be obtained. Design parameters are chosen as follows:

Frame period T_o = .09 second

Desired closed-loop poles = $-1 \pm j1$ (s-plane).

Transform to the discrete-time closed-loop poles (on z-plane), therefore

$$\text{C.L. Poles} = \exp(sT_o)$$

$$= .910232261 \pm j.082142809$$

where s = s-plane poles.

The coefficient matrices of the zero-order hold equivalent model for the plant with a frame period .09 second are

$$\Phi = \begin{bmatrix} 1 & 0.0860688 \\ 0 & 0.9139312 \end{bmatrix}, \qquad \Gamma = \begin{bmatrix} 0.0039312 \\ 0.0860688 \end{bmatrix},$$

$C = [\ 1 \qquad 0\ ]$, and $D = [0]$.

The result of the feedback gain matrix obtained by pole assignment at desired pole locations is

$$F = [\ 1.911350 \quad 0.9986520\ ].$$

Design 1 : Since the plant has OIV = (2), let the output multipliciity $N_1=2$. From equation (2.3.2), the uniform output sampling period is

$$T = .09/2 = .045 \quad \text{second.}$$

The multirate output sampling mechanism of this plant is shown in Figure 4.2.1.

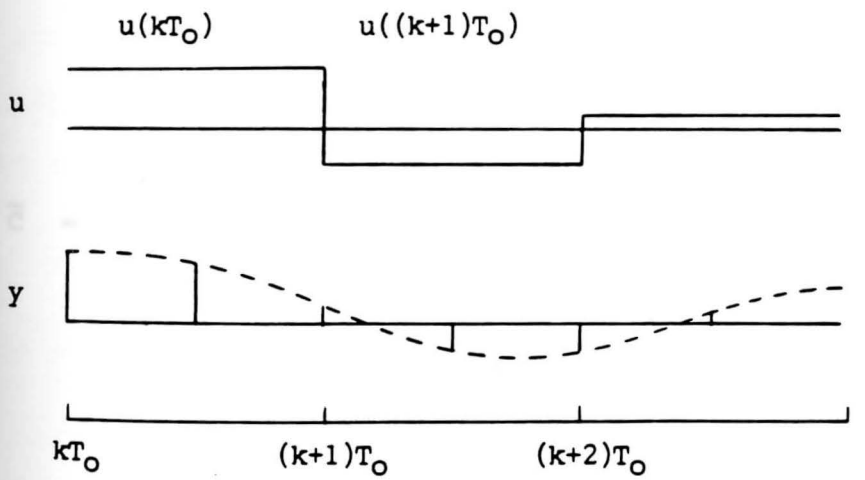


Figure 4.2.1 Multirate output sampling mechanism for a single input and single output plant with $N_1=2$.

As seen from Figure 4.2.1, the multirate output sampling mechanism is

$$y1(kT_o) = Cx(kT_o)$$

$$y1(kT_o+T) = Cx(kT_o+T) = C\Phi_1 x(kT_o) + C\Gamma_1 u(kT_o).$$

In this case, the coefficient matrices of the zero-order hold equivalent model with output sampling period T = .045 second is given by

$$\Phi_1 = \begin{bmatrix} 1 & 0.0440025 \\ 0 & 0.9559975 \end{bmatrix}, \quad \Gamma_1 = \begin{bmatrix} 0.0009975 \\ 0.0440025 \end{bmatrix}.$$

From a basic formula in equation (2.3.6), the results of $\bar{Y}(kT_o)$, $\bar{C}$ and $\bar{G}$ are

$$\bar{Y}(kT_o) = \begin{bmatrix} y_1(kT_o) \\ y_1(kT_o+T) \end{bmatrix},$$

$$\bar{C} = \begin{bmatrix} C \\ C\Phi_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 0.0440025 \end{bmatrix},$$

and

$$\bar{G} = \begin{bmatrix} 0 \\ C\Gamma_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.00099748 \end{bmatrix}.$$

Therefore

$$\bar{C}^{-1} = \begin{bmatrix} 1 & 0 \\ -22.72597 & 22.72597 \end{bmatrix}.$$

Then, the results of the matrix H and M are obtained from equation (2.3.12) and (2.3.13) respectively as follows:

$$H = [\ -22.56922 \quad 24.48057\ ]$$

$$M = [\ -0.0690478\ ].$$

Next, consider a dc gain N_r for tracking a reference step input. The output of this system is

$$y = x_1$$

and there is no steady state error for type 1 system. From the state variable feedback control law in equation (2.2.25) at steady state, both the control input and the state x_2 go to zero. Then the control law becomes

$$0 = - f_1 x(kT_o) + N_r r(kT_o).$$

Since $x(kT_o) = r(kT_o)$ is desired, then N_r must be chosen as

$$N_r = f_1 = 1.91135.$$

Design 2 : Since

$$\text{rank} \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & 0 \end{bmatrix} = 2+1 = 3,$$

the augmented system has OIV = (3). Therefore let the output multiplicities $N_1 = 3$. The uniform output sampling period T is

$$T = T_o/N = .09/3 = .03 \text{ second.}$$

The multirate output sampling mechanism in this case is shown in Figure 4.2.2.

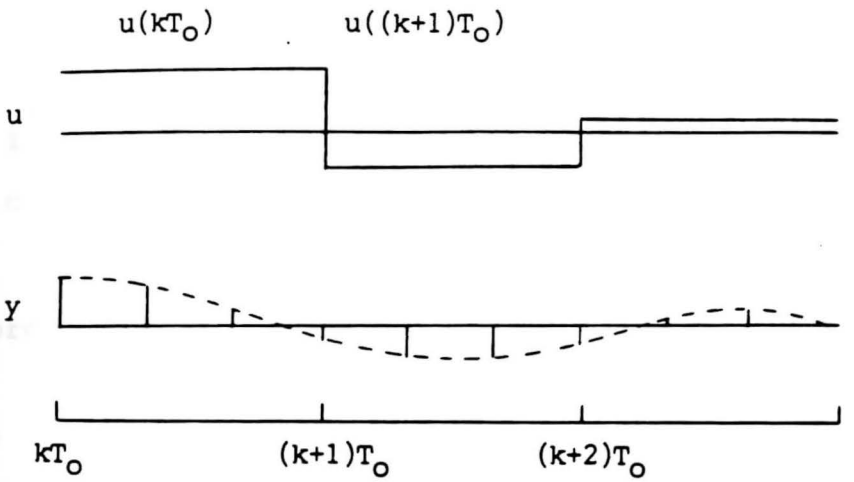


Figure 4.2.2 Multirate output sampling mechanism for a single input and single output plant with $N_1=3$.

From Figure 4.2.2, the multirate output sampling mechanism is

$$y(kT_o) = Cx(kT_o)$$

$$y(kT_o+T) = Cx(kT_o+T) = C\Phi_1 x(kT_o) + C\Gamma u(kT_o)$$

$$y(kT_o+2T) = Cx(kT_o+2T) = C\Phi_2 x(kT_o) + C\Gamma u(kT_o).$$

The coefficient matrices of the zero-order equivalent models for this plant with output sampling period T (.03 second) and 2T (.06 second) are obtained respectively as follows:

$$\Phi_1 = \begin{bmatrix} 1 & 0.0295545 \\ 0 & 0.9704455 \end{bmatrix}, \quad \Gamma_1 = \begin{bmatrix} 0.0004455 \\ 0.0295545 \end{bmatrix},$$

and

$$\Phi_2 = \begin{bmatrix} 1 & 0.0582355 \\ 0 & 0.9417645 \end{bmatrix}, \quad \Gamma_2 = \begin{bmatrix} 0.0017645 \\ 0.0582355 \end{bmatrix}.$$

Therefore,

$$\bar{y}(kT_o) = \begin{bmatrix} y(kT_o) \\ y(kT_o+T) \\ y(kT_o+2T) \end{bmatrix},$$

and

$$[\bar{C} \quad \bar{G}] = \begin{bmatrix} C & 0 \\ C\Phi_1 & C\Gamma_1 \\ C\Phi_2 & C\Gamma_2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0.0295545 & 0.0004455 \\ 1 & 0.0582355 & 0.0017645 \end{bmatrix}.$$

Then

$$[\bar{C} \quad \bar{G}]^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -50.335840 & 67.33834 & -17.00250 \\ 1000.0945 & -2.222.389 & 1127.861 \end{bmatrix}.$$

The minimum H matrix is obtained by the following steps:

Substituting M = [0] into equation (2.3.15) gives

$$H_o = [\ 49.99110 \quad -135.1820 \quad 87.10220\ ].$$

Since

$$[\bar{C} \quad \bar{G}]^{-1} = \begin{bmatrix} R_x \\ R_u \end{bmatrix},$$

therefore

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ -50.33584 & 67.33834 & -17.00250 \end{bmatrix}$$

and

$$Ru = [\ 1094.528 \quad -2222.389 \quad 1127.861\ ].$$

Substituting H_o and R_u into equation (2.5.23) gives

$$M = [-0.0611929\ ].$$

The matrix H minimum is obtained by using equation (2.3.15) again.

Therefore

$$H_{min} = [-16.98622 \quad 0.8124516 \quad 18.08512\ ].$$

4.3 Example 2: The 4th order plant with 1 input and 2 outputs

Consider state-space coefficient matrices of the controllable and observable analog plant as follows:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 2 & -1 & 0 & 0 \\ -1 & 0 & -3 & 0 \\ 1 & 0 & 0 & -2 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 2 \\ -1 \\ 1 \end{bmatrix},$$

$$C = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \text{and} \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

The opened-loop transfer functions of this plant are given by

$$\frac{Y_1}{u} = \frac{(s+.999)(s+5)(s+2)}{s(s+1)(s+2)(s+3)}$$

and

$$\frac{Y_2}{u} = \frac{(s+3)(s+.999)(s+1)}{s(s+1)(s+2)(s+3)}.$$

It shows that this plant has no invariant zeros at origin and has opened-loop poles at 0, -1, -2, -3. Suppose that the regulator design is to be obtained (reference input=0).

Choose: frame period T_o = .3 second,

desired closed-loop poles = -.5, -1, -2, -3.

Therefore,

desired discrete poles = .904837418, .818730753,

.670320046, and .548811636.

The coefficient matrices of the zero-order hold equivalent model with frame period T_o = .3 second for this plant are

$$\Phi = \begin{bmatrix} 1.000300 & 0 & 0 & 0 \\ 0.5184452 & 0.7408182 & 0 & 0 \\ -0.1978442 & 0 & 0.4065697 & 0 \\ 0.2256314 & 0 & 0 & 0.5488116 \end{bmatrix}$$

and

$$\Gamma = \begin{bmatrix} 0.3000450 \\ 0.6000084 \\ -0.2318771 \\ 0.2628010 \end{bmatrix}.$$

The result of the state variable feedback gain matrix obtained by the pole assignment method at desired poles becomes

F = [0.4652377 -2.839783E-07 1.504320E-09 5.755672E-10]

Design 1 : Since the plant has OIV=(2,2), let output multiplicities be $N_1 = N_2 = 2$. Then obtain the uniform output sampling period T from equation (2.3.2),

$$T = T_o/(N_1+N_2)$$

$$= .3/(2+2) \quad = .075 \text{ second.}$$

Figure 4.3.1 shows the multirate output sampling mechanism for this case.

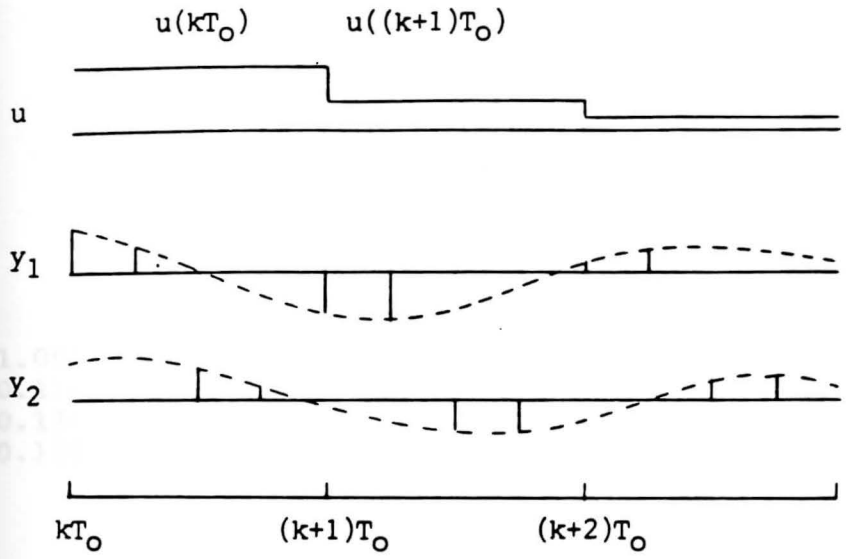


Figure 4.3.1 Multirate output sampling mechanism for a 1 input and 2 outputs plant with $N_1=N_2=2$.

As seen from Figure 4.3.1, the multirate output sampling mechanism can be written as follows:

$$y_1(kT_o) = C_1x(kT_o)$$

$$y_1(kT_o+T) = C_1x(kT_o+T) = C_1\Phi_1x(kT_o) + C_1\Gamma_1u(kT_o)$$

$$y_2(kT_o+2T) = C_2x(kT_o+2T) = C_2\Phi_2x(kT_o) + C_2\Gamma_2u(kT_o)$$

$$y_2(kT_o+3T) = C_2x(kT_o+3T) = C_2\Phi_3x(kT_o) + C_2\Gamma_3u(kT_o)$$

The Φ and Γ matrices for the zero-order hold equivalent models with sampling period T, 2T, and 3T respectively are

$$\Phi_1=\begin{bmatrix} 1.0000750 & 0 & 0 & 0 \\ 0.1445185 & 0.9277435 & 0 & 0 \\ -0.0671639 & 0 & 0.7985162 & 0 \\ 0.0696487 & 0 & 0 & 0.8607080 \end{bmatrix},$$

$$\Gamma_1 = \begin{bmatrix} 0.0750028 \\ 0.1500001 \\ -0.0697742 \\ 0.0723231 \end{bmatrix},$$

$$\Phi_2=\begin{bmatrix} 1.0001500 & 0 & 0 & 0 \\ 0.2786055 & 0.8607080 & 0 & 0 \\ -0.1208004 & 0 & 0.6376282 & 0 \\ 0.1296011 & 0 & 0 & 0.7408182 \end{bmatrix},$$

$$\Gamma_2 = \begin{bmatrix} 0.1500113 \\ 0.3000011 \\ -0.1305276 \\ 0.1397960 \end{bmatrix},$$

$$\Phi_3=\begin{bmatrix} 1.0002250 & 0 & 0 & 0 \\ 0.4030146 & 0.7985162 & 0 & 0 \\ -0.1636350 & 0 & 0.5091564 & 0 \\ 0.1812078 & 0 & 0 & 0.6376282 \end{bmatrix},$$

and

$$\Gamma_3 = \begin{bmatrix} 0.2250253 \\ 0.4500036 \\ -0.1840780 \\ 0.2030947 \end{bmatrix}.$$

Therefore from equation (2.3.6),

$$\bar{y}(kT_o) = \begin{bmatrix} y1(kT_o) \\ y1(kT_o+T) \\ y2(kT_o+2T) \\ y2(kT_o+3T) \end{bmatrix}, \quad \bar{C} = \begin{bmatrix} C_1 \\ C_1\Phi_1 \\ C_2\Phi_2 \\ C_2\Phi_3 \end{bmatrix} \text{ and } \bar{G} = \begin{bmatrix} 0 \\ C_1\Gamma_1 \\ C_2\Gamma_2 \\ C_2\Gamma_3 \end{bmatrix},$$

where

$$\bar{C} = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0.0773546 & 0.9277435 & 0.7985162 & 0 \\ 0.1296011 & 0 & 0 & 0.7408182 \\ 0.1812078 & 0 & 0 & 0.6376282 \end{bmatrix}$$

and $\bar{G} = \begin{bmatrix} 0 \\ 0.0802259 \\ 0.1397960 \\ 0.2030947 \end{bmatrix}$. Then

$$C^{-1} = \begin{bmatrix} 0 & 0 & -12.356000 & 14.355620 \\ -6.179162 & 7.738305 & 7.396222 & -8.593184 \\ 7.179162 & -7.738305 & -7.396222 & 8.593184 \\ 0 & 0 & 3.511456 & -2.511418 \end{bmatrix}$$

The results of matrix H and matrix M calculated by equation (2.3.12) and equation (2.3.13) respectively are

H = [1.304340E-6 -1.632689E-6 -5.750199 6.680778]

M = [0.4133836].

Design 2 : Since

$$\text{rank} \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 2 & -1 & 0 & 0 & 2 \\ -1 & 0 & -3 & 0 & -1 \\ 1 & 0 & 0 & -2 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} = 4 + 1 = 5,$$

the augmented system has OIV=(3,2). Let the output multiplicities be $N_1 = 3$ and $N_2 = 2$. The uniform output sampling period becomes

$$T = .2/(3+2) = .04 \text{ second.}$$

Figure 4.3.2 shows the multirate output sampling mechanism for this case.

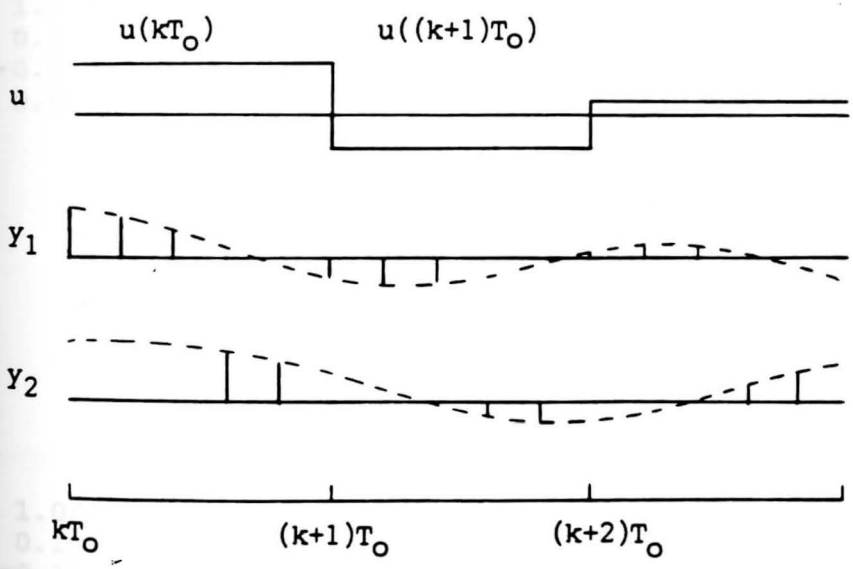


Figure 4.3.2 Multirate output sampling mechanism for a 1 input and 2 outputs plant with N_1=3 and N_2=2.

As seen in Figure 4.3.2, the multirate output sampling mechanism can be written as follows.

$$y_1(kT_o) = C_1x(kT_o)$$

$$y_1(kT_o+T) = C_1x(kT_o+T) = C_1\Phi_1x(kT_o) + C_1\Gamma_1u(kT_o)$$

$$y_1(kT_o+2T) = C_1x(kT_o+2T) = C_1\Phi_2x(kT_o) + C_1\Gamma_2u(kT_o)$$

$$y_2(kT_o+3T) = C_2x(kT_o+3T) = C_2\Phi_3x(kT_o) + C_1\Gamma_3u(kT_o)$$

$$y_2(kT_o+4T) = C_2x(kT_o+4T) = C_2\Phi_4x(kT_o) + C_2\Gamma_4u(kT_o).$$

The Φ and Γ matrices of the zero-order hold equivalent models for this plant with output sampling period T, 2T, 3T and 4T respectively are

$$\Phi_1 = \begin{bmatrix} 1.0000600 & 0 & 0 & 0 \\ 0.1164745 & 0.9417645 & 0 & 0 \\ -0.0549116 & 0 & 0.8352702 & 0 \\ 0.0565415 & 0 & 0 & 0.8869204 \end{bmatrix},$$

$$\Gamma_1 = \begin{bmatrix} 0.0600018 \\ 0.1200001 \\ -0.0566067 \\ 0.0582699 \end{bmatrix},$$

$$\Phi_2 = \begin{bmatrix} 1.0001200 & 0 & 0 & 0 \\ 0.2261730 & 0.8869204 & 0 & 0 \\ -0.1007810 & 0 & 0.6976763 & 0 \\ 0.1066927 & 0 & 0 & 0.7866279 \end{bmatrix},$$

$$\Gamma_2 = \begin{bmatrix} 0.1200072 \\ 0.2400006 \\ -1.1718330 \\ 0.1133433 \end{bmatrix},$$

$$\Phi_3 = \begin{bmatrix} 1.0001800 & 0 & 0 & 0 \\ 0.3294901 & 0.8352702 & 0 & 0 \\ -0.1390976 & 0 & 0.5827483 & 0 \\ 0.1511763 & 0 & 0 & 0.6976763 \end{bmatrix},$$

$$\Gamma_3 = \begin{bmatrix} 0.1800162 \\ 0.3600019 \\ -0.1527235 \\ 0.1655818 \end{bmatrix},$$

$$\Phi_4 = \begin{bmatrix} 1.0002400 & 0 & 0 & 0 \\ 0.4267975 & 0.7866279 & 0 & 0 \\ -0.1711056 & 0 & 0.4867523 & 0 \\ 0.1906330 & 0 & 0 & 0.6187834 \end{bmatrix},$$

and

$$\Gamma_4 = \begin{bmatrix} 0.2400288 \\ 0.4800043 \\ -0.1940570 \\ 0.2153062 \end{bmatrix}.$$

Therefore from equation (2.3.6),

$$\bar{y}(kT_o) = \begin{bmatrix} y_1x(kT_o) \\ y_1x(kT_o+T) \\ y_1x(kT_o+2T) \\ y_2x(kT_o+3T) \\ y_2x(kT_o+4T) \end{bmatrix} \quad \text{and} \quad [\bar{C} \quad \bar{G}] = \begin{bmatrix} C_1 & 0 \\ C_1\Phi_1 & C_1\Gamma_1 \\ C_1\Phi_2 & C_1\Gamma_2 \\ C_2\Phi_3 & C_2\Gamma_3 \\ C_2\Phi_4 & C_2\Gamma_4 \end{bmatrix},$$

where

$$[\bar{C} \quad \bar{G}] =$$

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0.0615628 & 0.9417645 & 0.8352702 & 0 & 0.0633934 \\ 0.1253920 & 0.8869204 & 0.6976763 & 0 & 0.1328173 \\ 0.1511763 & 0 & 0 & 0.6976763 & 0.1655818 \\ 0.1906330 & 0 & 0 & 0.6187834 & 0.2153062 \end{bmatrix},$$

and $$[\bar{C} \quad \bar{G}]^{-1} =$$

$$\begin{bmatrix} -1178.486 & 2662.264 & -1498.150 & -391.4482 & 441.3566 \\ 93.82704 & -220.2887 & 129.2484 & 41.48431 & -46.77343 \\ -92.82704 & 220.2887 & -129.2484 & -41.48431 & 46.77343 \\ 24.27880 & -54.84713 & 30.86441 & 12.57307 & -12.56003 \\ 973.6601 & -2199.551 & 1237.765 & 310.4552 & -350.0373 \end{bmatrix}.$$

The minimum H matrix is obtained as follows:

Substituting M = [0] into equation (2.3.18) gives

$$H_o = [-412.5253 \quad 931.9170 \quad -524.4225 \quad -138.8340 \quad 156.5349]$$

From equation (2.5.2),

$$R_x = \begin{bmatrix} -1178.49 & 2662.264 & -1498.150 & -391.4482 & 441.357 \\ 93.8270 & -220.2887 & 129.2484 & 41.48431 & -46.7734 \\ -92.8270 & 220.2887 & -129.2484 & -41.48431 & 46.7734 \\ 24.2788 & -54.84713 & 30.86441 & 12.57307 & -12.5600 \end{bmatrix}$$

and

$$R_u = [973.660 \quad -2199.551 \quad 1237.765 \quad 310.4552 \quad -350.0373].$$

Substituting H_o and R_u into equation (2.5.23) gives

$$M = [.424].$$

Applying M into equation (2.3.18) gives

$$H_{min} = [\ .3065341 \quad -.6924752 \quad .3896791 \quad -7.200939 \quad 8.119036\ ].$$

To see disturbance effects when the matrix H is significantly large, choose the state transition matrix

$$M = [.44].$$

The result of H matrix obtained from equation (2.3.15) is

$$H = [\ 15.88510 \quad -35.88528 \quad 2.019391 \quad -2.233655 \quad 2.518439\ ]$$

The results of using H_{min} in comparison with the results of using H are given in section 4.5 as Example 2 Design 2 and Example 2 Design 2A, respectively.

4.4 Example 3: The 4^{th} order plant with 2 inputs and 2 outputs

The regulator design is to be obtained for the controllable and observable analog plant which has state-space coefficient matrices as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 3 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 \\ 0 & -2 & 0 & 0 \end{bmatrix}, \qquad B = [B1 \quad B2] = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix},$$

$$C = \begin{bmatrix} C1 \\ C2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \text{ and } D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

The opened-loop transfer functions of this plant are given by

$$\frac{y_1}{u_1} = \frac{1}{(s^2+1)},$$

$$\frac{y_1}{u_2} = \frac{2}{s(s^2+1)},$$

$$\frac{y_2}{u_1} = \frac{-2}{s(s^2+1)},$$

and

$$\frac{y_2}{u_2} = \frac{(s+1.732051)(s-1.732051)}{s^2(s^2+1)}.$$

This plant has no zeros at origin and has opened-loop poles at 0, 0, and ±j1.

Choose : frame period T_o = .2 second

desired closed-loop poles = -.833±j.164, -1.364±j1.590.

In this case, these poles are obtained from the analog optimal control design by using the Linear Quadratic Regulator (LQR) design for which the cost function is

$$J = \int_0^\infty (x^T Q x + \tau u^T R u)\, dt$$

where $Q = [I_4]$, $R = [I_2]$ and $\tau = 1$.

Since Program CC does not have the discrete-time optimal

control command available, the pole assignment technique is used for the discrete-time design by transforming the optimal analog poles to discrete-time poles. Therefore,

the desired discrete-time poles = .846096645±j.027705397
and .72306452±j.237997776.

The Φ and Γ matrices of the zero-order hold equivalent model of this plant with frame period T_o = .2 second become

$$\Phi = \begin{bmatrix} 1.0598000 & 0.1986693 & 0 & 0.0398668 \\ 0.5960080 & 0.9800666 & 0 & 0.3973387 \\ -0.0079840 & -0.0398668 & 1 & 0.1946773 \\ -0.1196005 & -0.1397339 & 0 & 0.9202663 \end{bmatrix}$$

and

$$\Gamma = \begin{bmatrix} 0.0199334 & 0.0026613 \\ 0.1986693 & 0.0398668 \\ -0.0026613 & 0.0197337 \\ -0.0398668 & 0.1946773 \end{bmatrix}.$$

Since a multivariable pole-placement algorithm was not available, the state-variable feedback gain matrix F was found by using a Weighted Least-Squares Approximation method. The details of this method can be seen in Appendix F. The result gives the approximated value of the matrix F as shown below.

$$F = \begin{bmatrix} 3.589307 & 1.725144 & -0.649064 & 1.519300 \\ 0.120262 & -1.688656 & 3.245327 & 2.403051 \end{bmatrix}$$

From equation (2.4.6), the Φ and Γ matrices of the zero-order hold equivalent model of the augmented discrete-time for this plant becomes

$$\Phi_A = \begin{bmatrix} 1.059800 & 0.1986693 & 0 & 0.0398668 & 0.0008776 \\ 0.5960080 & 0.9800666 & 0 & 0.3973387 & 0.0093199 \\ -0.0079840 & -0.0398668 & 1 & 0.1946773 & 0.0045773 \\ -0.1196005 & -0.3973387 & 0 & 0.9202663 & 0.0232448 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and

$$\Gamma_A = \begin{bmatrix} 0.0199334 & 0.0017837 \\ 0.1986693 & 0.0305469 \\ -0.0026613 & 0.0151563 \\ -0.0398668 & 0.1714326 \\ 0 & 1 \end{bmatrix}.$$

To find the feedback gain [F G] for the augmented system, use another approximation by

$$G = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Therefore, the matrix [F G] becomes

$$\begin{bmatrix} 3.5893070 & 1.7251440 & -0.6490643 & 1.5193000 & 0 \\ 0.1202620 & -1.6886560 & 3.2453270 & 2.4030510 & 0 \end{bmatrix}.$$

Design 2 : Only Design 2 is presented for Example 3. Since T_o=.2 second, therefore the output sampling period T is obtained by

$$T = .2/8 = .025 \text{ second.}$$

The multirate output sampling mechanism of this plant is shown in Figure 4.4.1.

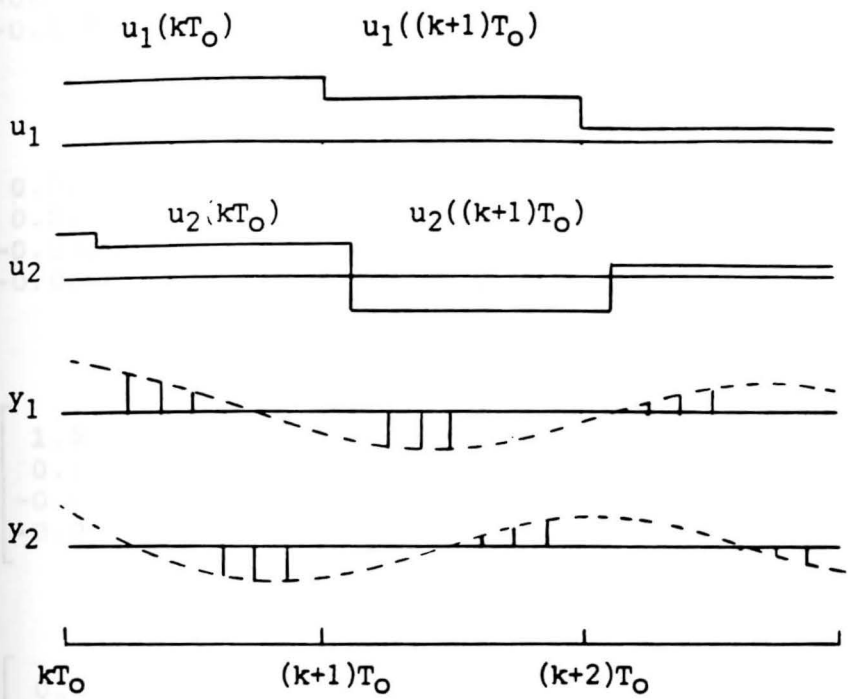


Figure 4.4.1 Multirate output sampling mechanism for a 2 inputs and 2 outputs plant with $N_1=N_2=3$.

As seen from Figure 4.4.1, a basic formula of the multirate output sampling mechanism is

$$\begin{bmatrix} y_1(KT_o+2T) \\ y_1(kT_o+3T) \\ y_1(kT_o+4T) \\ y_2(kT_o+5T) \\ y_2(kT_o+6T) \\ y_2(kT_o+7T) \end{bmatrix} = \begin{bmatrix} C_1 \\ C_1\Phi_T \\ C_1\Phi_{2T} \\ C_2\Phi_{3T} \\ C_2\Phi_{4T} \\ C_2\Phi_{5T} \end{bmatrix} x(kT_o+2T) + \begin{bmatrix} 0 \\ C_1\Gamma_T \\ C_1\Gamma_{2T} \\ C_2\Gamma_{3T} \\ C_2\Gamma_{4T} \\ C_2\Gamma_{5T} \end{bmatrix} u(kT_o).$$

The Φ and Γ matrices of the zero-order hold equivalent models of the plant at output sampling period T, 2T, 3T, 4T, and 5T respectively are

$$\Phi_T=\begin{bmatrix} 1.0009370 & 0.0249974 & 0 & 0.0006250 \\ 0.0749922 & 0.9996875 & 0 & 0.0499948 \\ -0.0000156 & -0.0006250 & 0 & 0.0249896 \\ -0.0018749 & -0.0499948 & 0 & 0.9987501 \end{bmatrix},$$

$$\Gamma_T=\begin{bmatrix} 0.0003125 & 0.0000052 \\ 0.0249974 & 0.0006250 \\ -0.0000052 & 0.0003124 \\ -0.0006250 & 0.0249896 \end{bmatrix},$$

$$\Phi_{2T}=\begin{bmatrix} 1.0037490 & 0.0499792 & 0 & 0.0024995 \\ 0.1499375 & 0.9987503 & 0 & 0.0999583 \\ -0.0001250 & -0.0024995 & 0 & 0.0499167 \\ -0.0074984 & -0.0999583 & 0 & 0.9950010 \end{bmatrix},$$

$$\Gamma_{2T}=\begin{bmatrix} 0.0012497 & 0.0000417 \\ 0.0499792 & 0.0024995 \\ -0.0000417 & 0.0012490 \\ -0.0024995 & 0.0499167 \end{bmatrix},$$

$$\Phi_{3T}=\begin{bmatrix} 1.0084340 & 0.0749297 & 0 & 0.0056224 \\ 0.2247891 & 0.9971888 & 0 & 0.1498594 \\ -0.0004218 & -0.0056224 & 0 & 0.0747188 \\ -0.0168671 & -0.1498594 & 0 & 0.9887553 \end{bmatrix},$$

$$\Gamma_{3T}=\begin{bmatrix} 0.0028112 & 0.0001406 \\ 0.0749297 & 0.0056224 \\ -0.0001406 & 0.0028072 \\ -0.0056224 & 0.0747188 \end{bmatrix},$$

$$\Phi_{4T}=\begin{bmatrix} 1.0149880 & 0.0998334 & 0 & 0.0099917 \\ 0.2995002 & 0.9950042 & 0 & 0.1996668 \\ -0.0009995 & -0.0099917 & 0 & 0.0993337 \\ -0.0299750 & -0.1996668 & 0 & 0.9800167 \end{bmatrix},$$

$$\Gamma_{4T}=\begin{bmatrix} 0.0049958 & 0.0003332 \\ 0.0998334 & 0.0099917 \\ -0.0003332 & 0.0049833 \\ -0.0099917 & 0.0993337 \end{bmatrix},$$

$$\Phi_{5T}=\begin{bmatrix} 1.0234070 & 0.1246747 & 0 & 0.0156047 \\ 0.3740242 & 0.9921977 & 0 & 0.2493495 \\ -0.0019516 & -0.0156047 & 0 & 0.1236989 \\ -0.0468140 & -0.2493495 & 0 & 0.9687907 \end{bmatrix},$$

and

$$\Gamma_{5T}=\begin{bmatrix} 0.0078023 & 0.0006505 \\ 0.1246747 & 0.0156047 \\ -0.0006505 & 0.0077718 \\ -0.0156047 & 0.1236989 \end{bmatrix}.$$

Therefore, $[\bar{C} \quad \bar{G}]$ is

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1.00094 & .024997 & 0 & .000625 & .000313 & .000005 \\ 1.00375 & .049979 & 0 & .002410 & .001250 & .000042 \\ -.000422 & -.005622 & 1 & .074719 & -.000141 & .002807 \\ -.000995 & -.009992 & 1 & .099334 & -.000333 & .004983 \\ -.001952 & -.015605 & 1 & .123699 & -.000651 & .007772 \end{bmatrix}$$

and $[\bar{C} \quad \bar{G}]^{-1}$ is

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -59.910 & 79.794 & -19.884 & .67414 & -1.3483 & .67414 \\ .50592 & -1.0137 & .50781 & 10.051 & -15.101 & 6.0505 \\ -15.823 & 31.697 & -15.873 & -181.331 & 322.662 & -141.33 \\ 1617.6 & -3238.3 & 1617.7 & 281.77 & -483.537 & 201.77 \\ 202.12 & -484.90 & 282.79 & 1617.86 & -3235.71 & 1617.9 \end{bmatrix}.$$

From equation (2.4.15), the Φ and Γ matrices of the zero-order hold equivalent model with output sampling period 6T is

$$\Phi_{6T} = \begin{bmatrix} 1.0336870 & 0.1494381 & 0 & 0.0224578 \\ 0.4483144 & 0.9887711 & 0 & 0.2988763 \\ -0.0033712 & -0.0224578 & 1 & 0.1477525 \\ -0.0673735 & -0.2988763 & 0 & 0.9550843 \end{bmatrix},$$

and

$$\Gamma_{6T} = \begin{bmatrix} 0.0112289 & 0.0011237 \\ 0.1494381 & 0.0224578 \\ -0.0011237 & 0.0111657 \\ -0.0224578 & 0.1477525 \end{bmatrix}.$$

The matrix H minimum can be determined by the following steps:

Letting $M = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$,

the result of H_o obtained from equation (2.4.19) is

$$\begin{bmatrix} 345.95 & -776.96 & 434.61 & 136.10 & -332.31 & 195.57 \\ -316.57 & 701.80 & -385.13 & 103.84 & -307.04 & 206.45 \end{bmatrix}$$

In this case , R_x =

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -59.910 & 79.794 & -19.884 & 0.6741 & -1.34827 & 0.67414 \\ 0.5059 & -1.0137 & 0.5078 & 10.0510 & -15.1011 & 6.05050 \\ -15.823 & 31.697 & -15.8732 & -181.330 & 322.6615 & -141.331 \end{bmatrix}$$

and R_u =

$$\begin{bmatrix} 1617.62 & -3238.32 & 1617.69 & 281.770 & -483.540 & 201.770 \\ 202.116 & -484.902 & 282.786 & 1617.86 & -3235.71 & 1617.86 \end{bmatrix}.$$

Substituting H_o and R_u into equation (2.5.23) gives

$$M = \begin{bmatrix} -0.229871 & -0.0685321 \\ 0.236488 & -0.1303905 \end{bmatrix}$$

which has its eigenvalues = -.1801308±j.1171876 (within a unit circle). Substituting M into equation (2.4.18) gives the result of the matrix H minimum. Therefore,

H_{min} =

$$\begin{bmatrix} -39.752 & .66438 & 43.366 & -39.551 & .588604 & 38.3129 \\ 39.648 & -.80162 & -39.435 & -40.482 & .515322 & 43.2118 \end{bmatrix}$$

Analog models of all three examples are implemented on the GP-6 analog computer, and their circuit diagrams are shown in Appendix G. The M, H, and H_{min} matrices obtained from all designs are applied in the multirate sampled-data control law in equation (2.2.24),

$$u(kT_o+T_o) = Mu(kT_o) - H\bar{Y}(kT_o) + N_r r(kT_o),$$

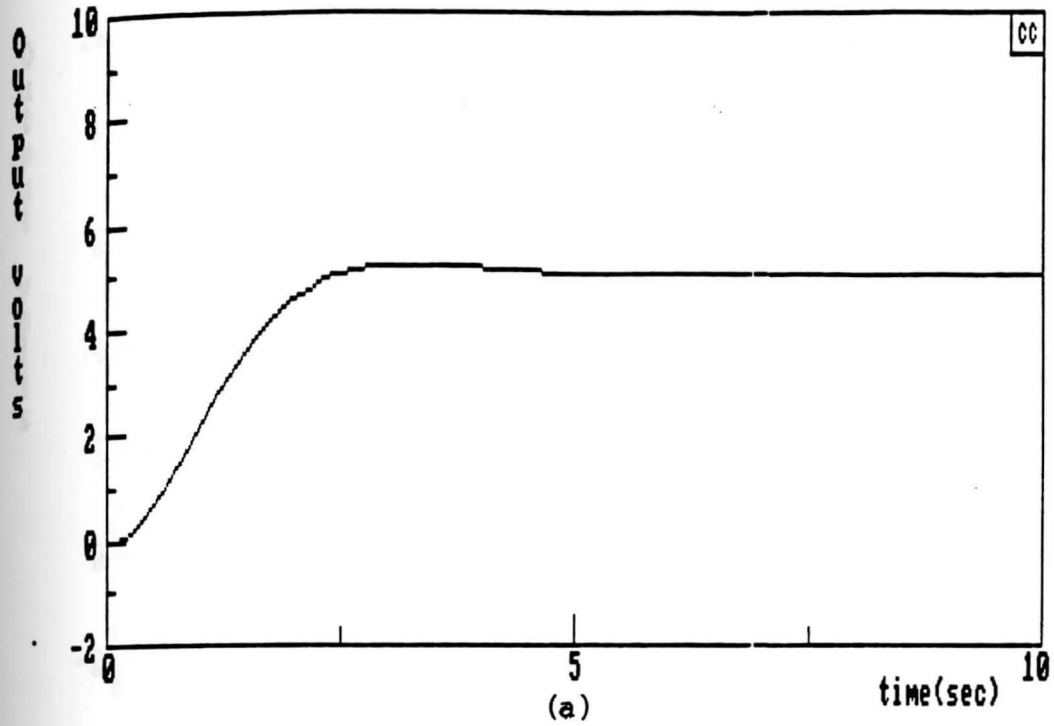
for implementing the multirate sampled-data controller. The results from Computer Simulation and Real-time implementation of all designs are presented in section 4.5.

4.5 Computer Simulation and Real-time Results

This section shows the Computer Simulation results in comparison with the Real-time results.

Figures 4.5.1-4.5.3 show the results of Example 1 Design 1. Figures 4.5.4-4.5.6 show the results of Example 1 Design 2. Figures 4.5.7-4.5.9 show the results of Example 2 Design 1. Figures 4.5.10-4.5.12 show the results of Example 2 Design 2. Figures 4.5.13-4.5.15 show the Real-time results of Example 2 Design 2 and compared with the Real-time results of Example 2 Design 2A (on page 79) to see the effects of disturbances when H_{min} and H are used. Figures 4.5.16-4.5.19 show the results of Example 3 Design 2.

Brief explanations of two options in program CC that are used for simulating the results of each design are given in Appendix H.



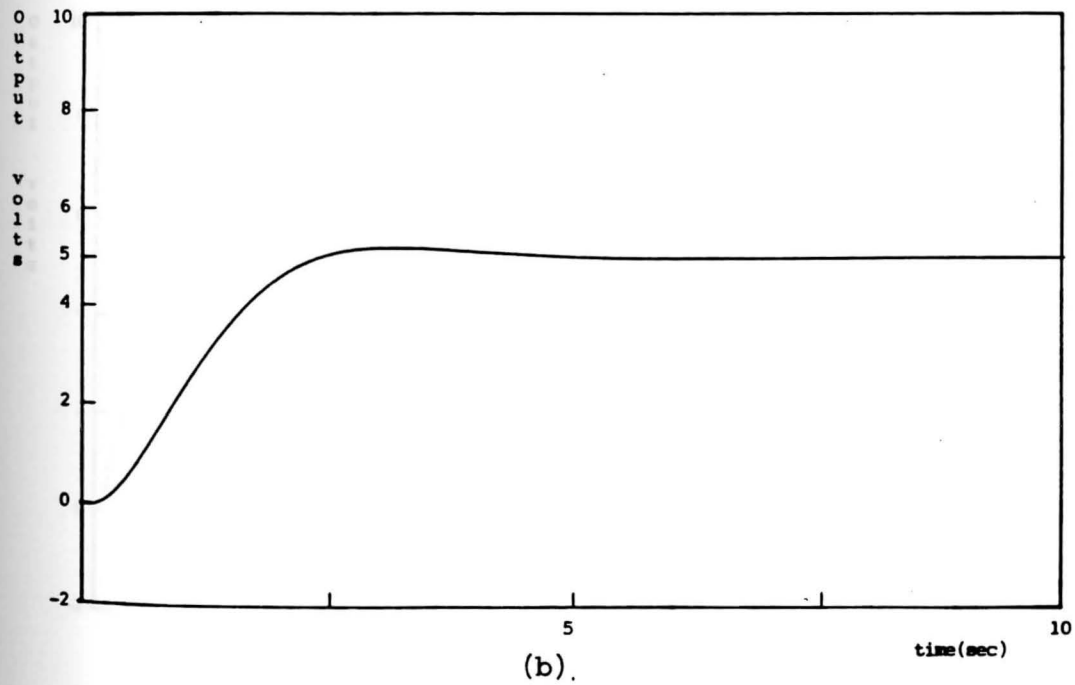
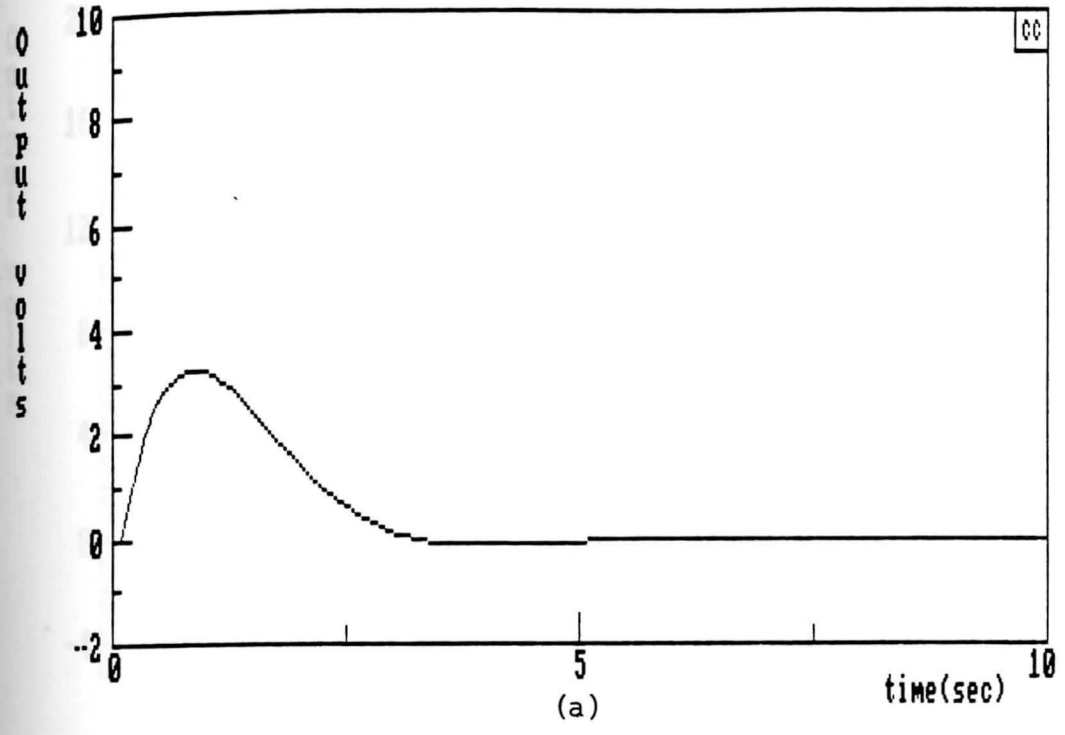
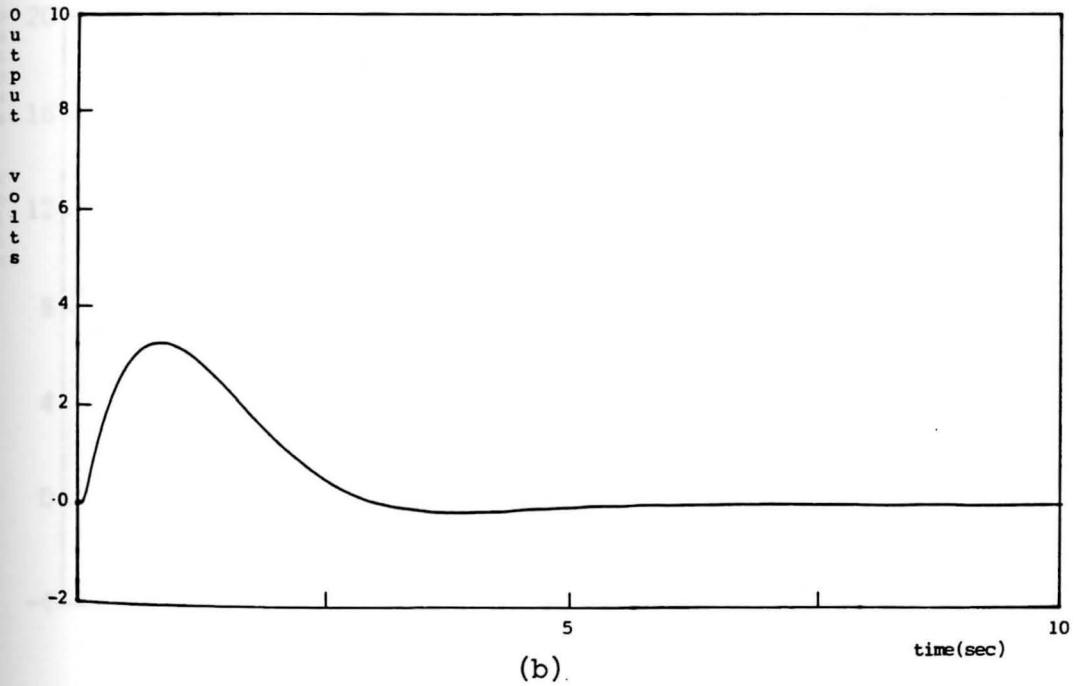


Figure 4.5.1 Output signals of Example 1 Design 1. (a) Simulated output signal; (b) Real-time output signal.

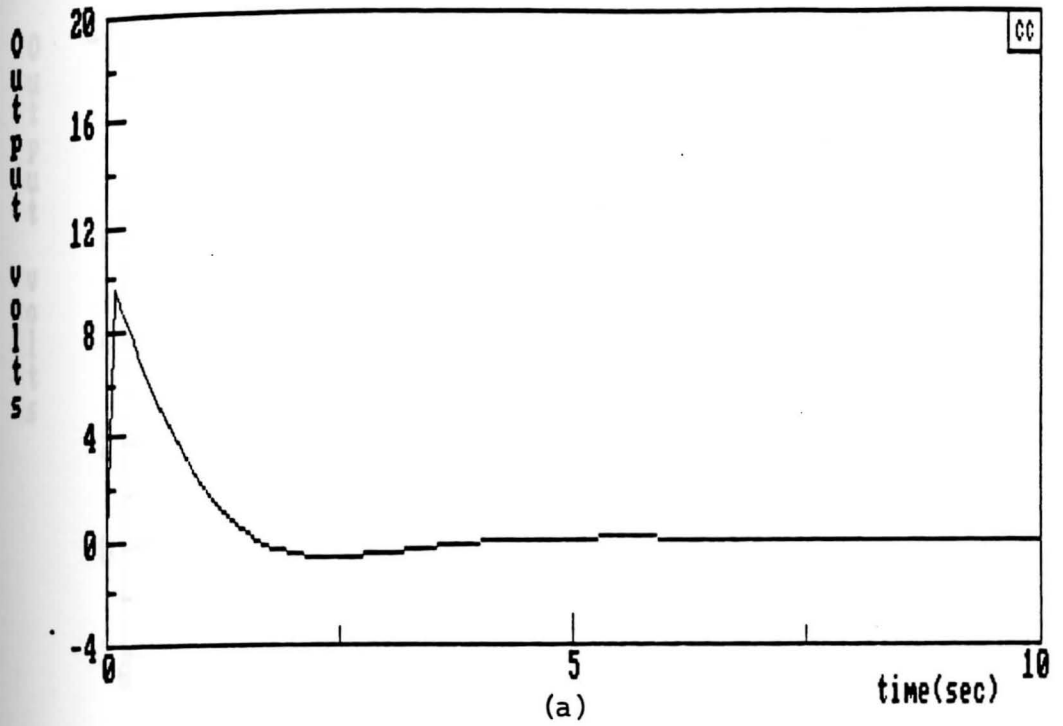


(a)

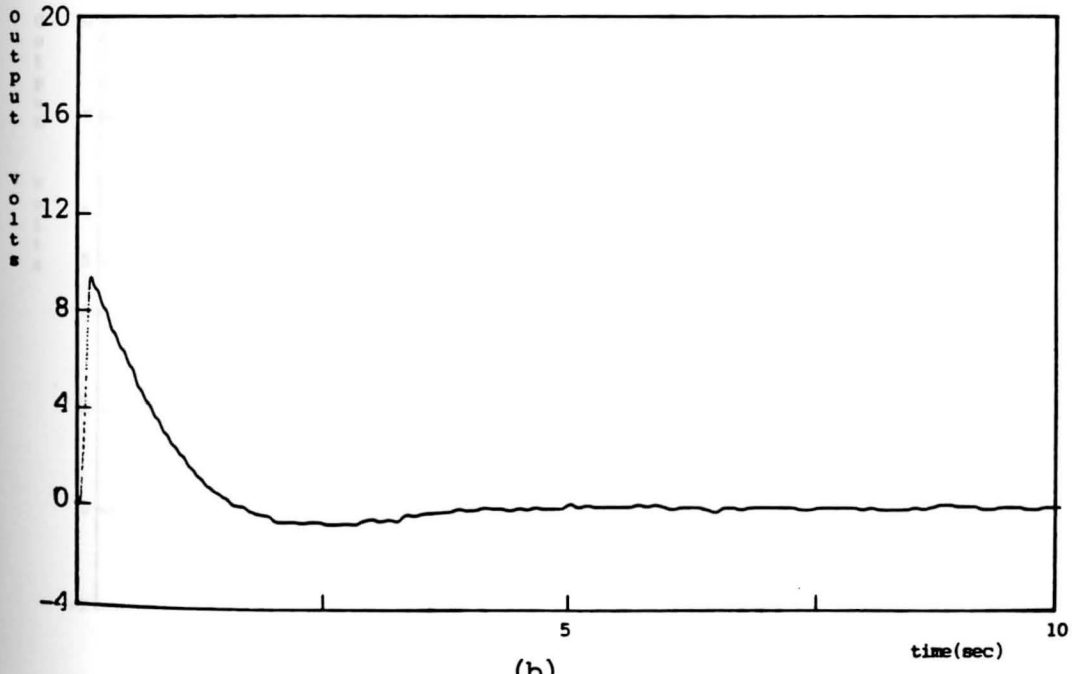


(b)

Figure 4.5.2 State x_2 signals of Example 1 Design 1. (a) Simulated state x_2 signal; (b) Real-time state x_2 signal.

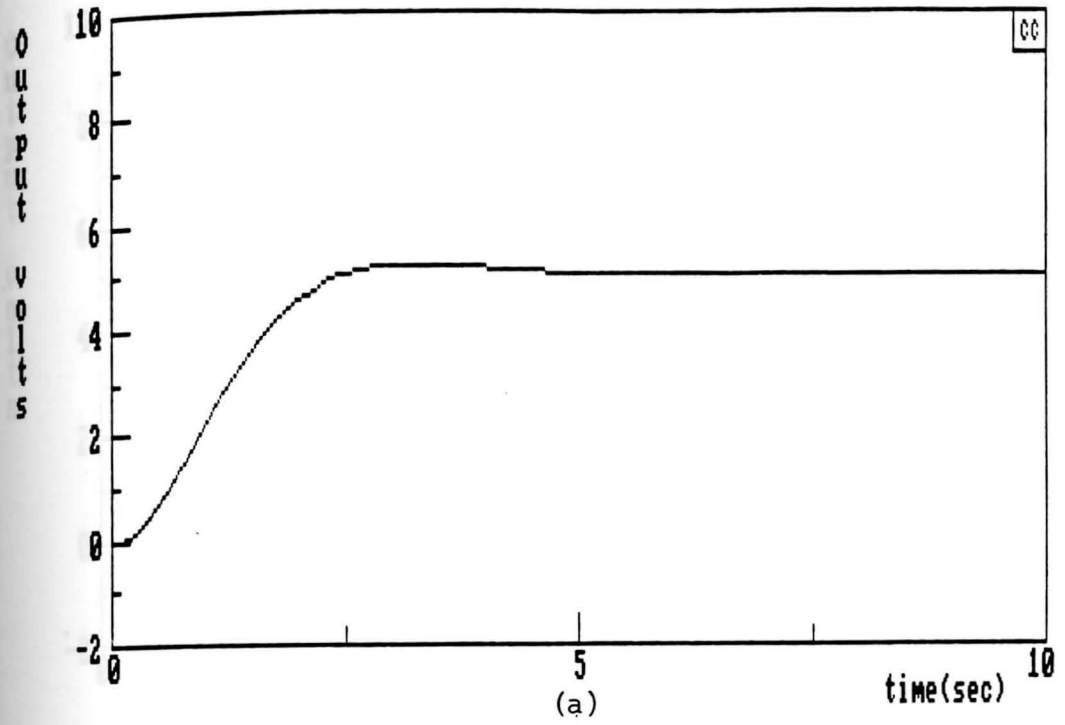


(a)

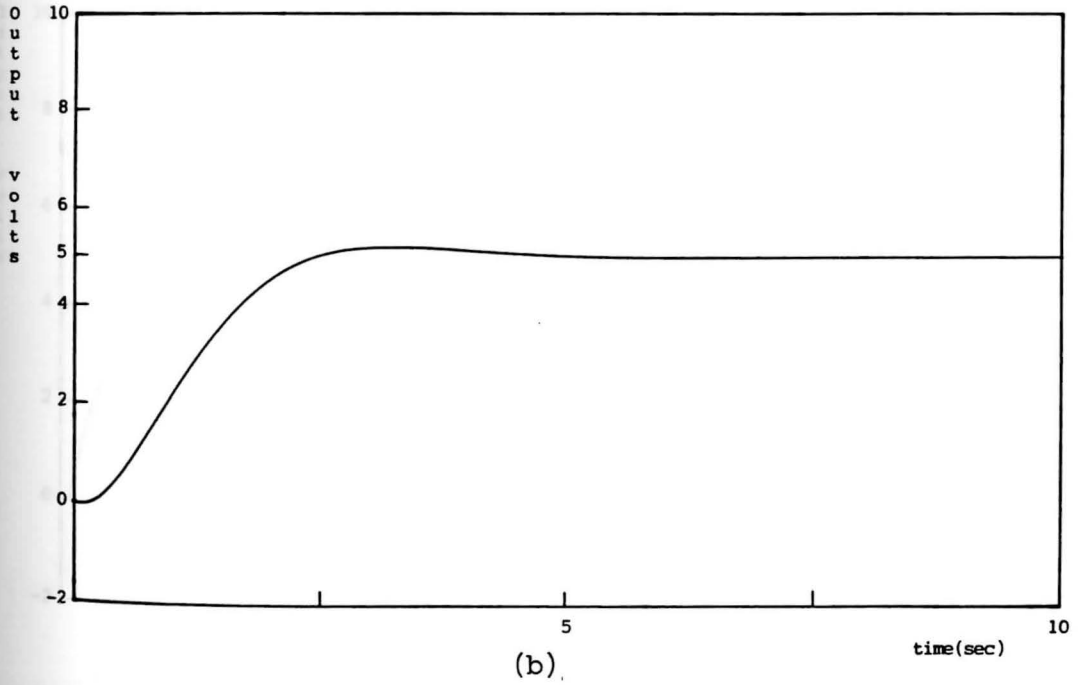


(b)

Figure 4.5.3 Control input signals of Example 1 Design 1. (a) Simulated control input signal; (b) Real-time control input signal.

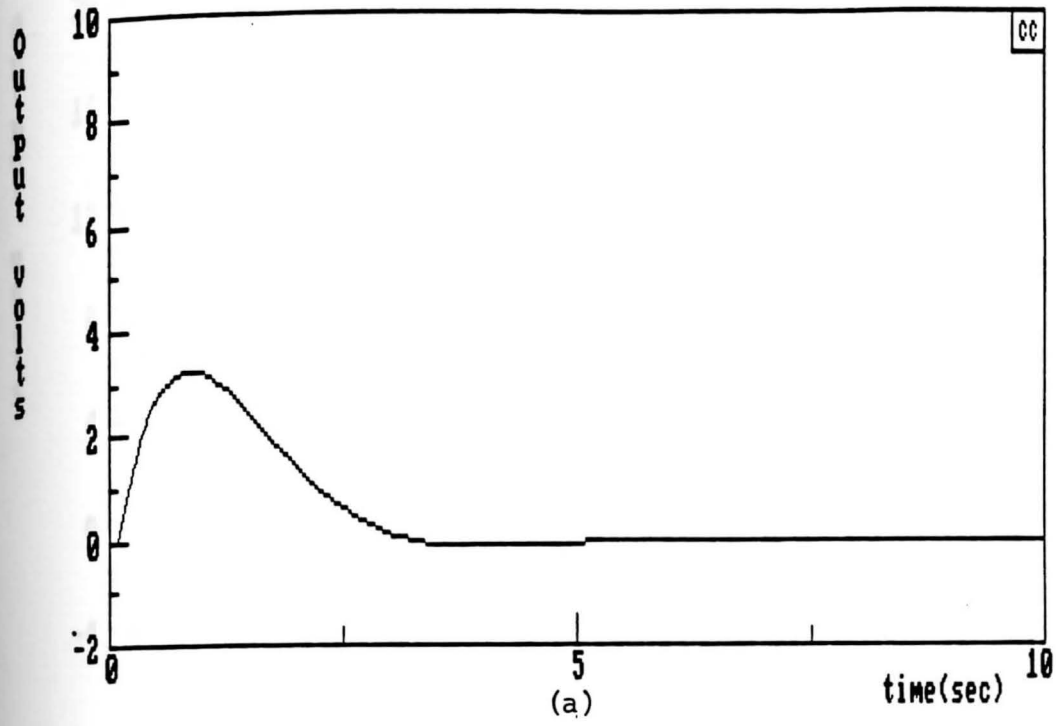


(a)



(b)

Figure 4.5.4 Output signals of Example 1 Design 2. (a) Simulated output signal; (b) Real-time output signal.



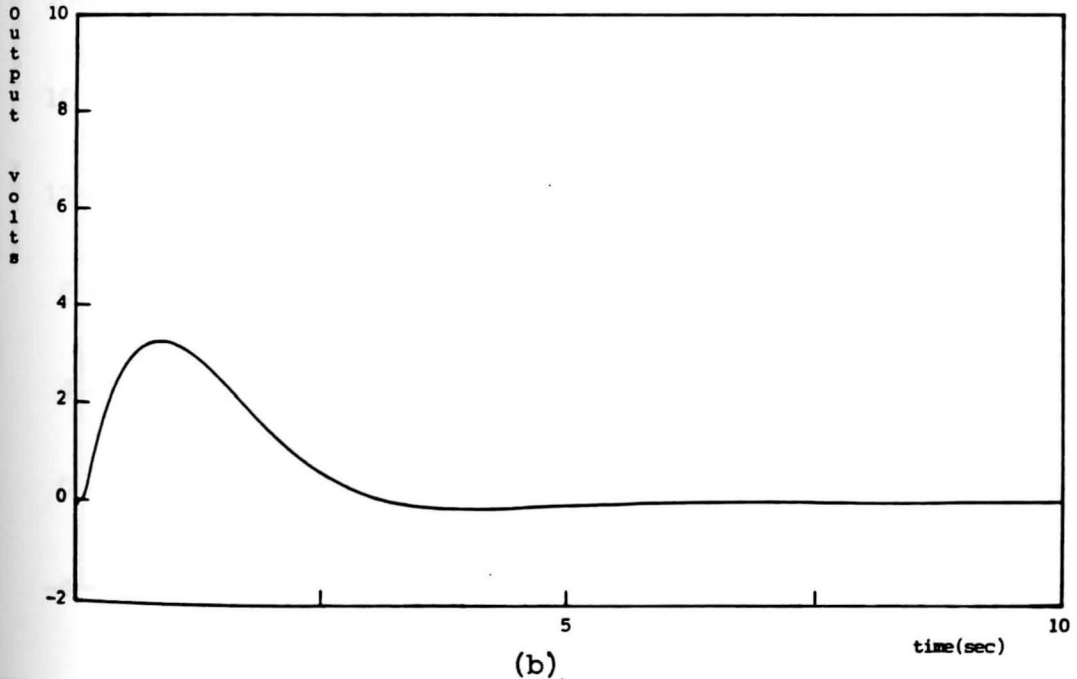
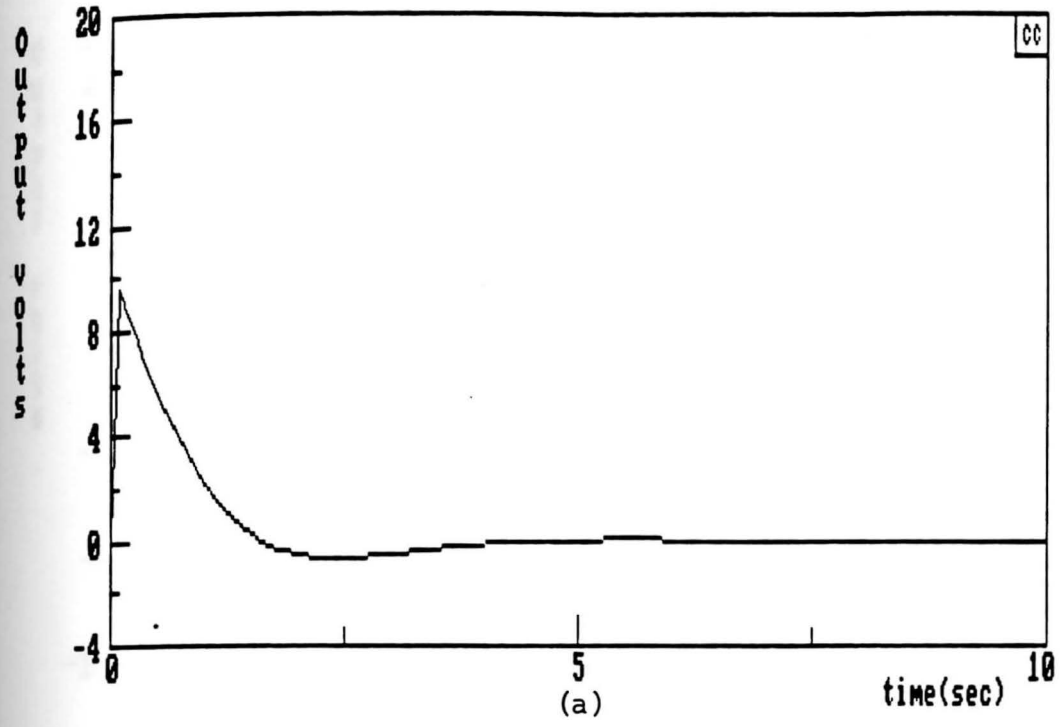


Figure 4.5.5 State x_2 signals of Example 1 Design 2. (a) Simulated state x_2 signal; (b) Real-time state x_2 signal.



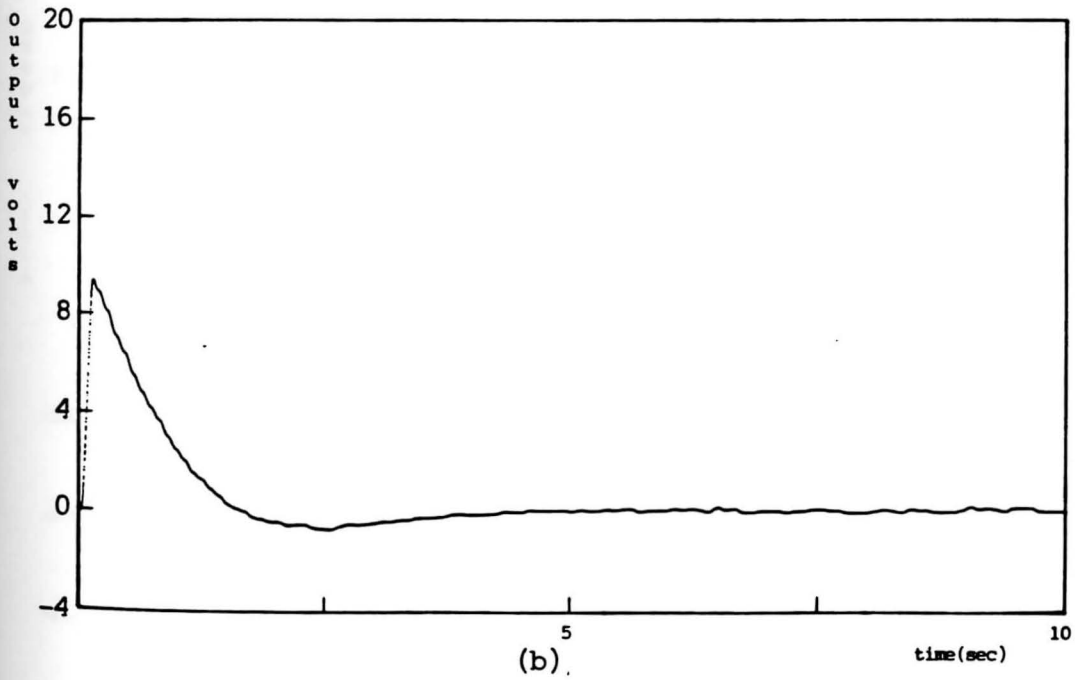
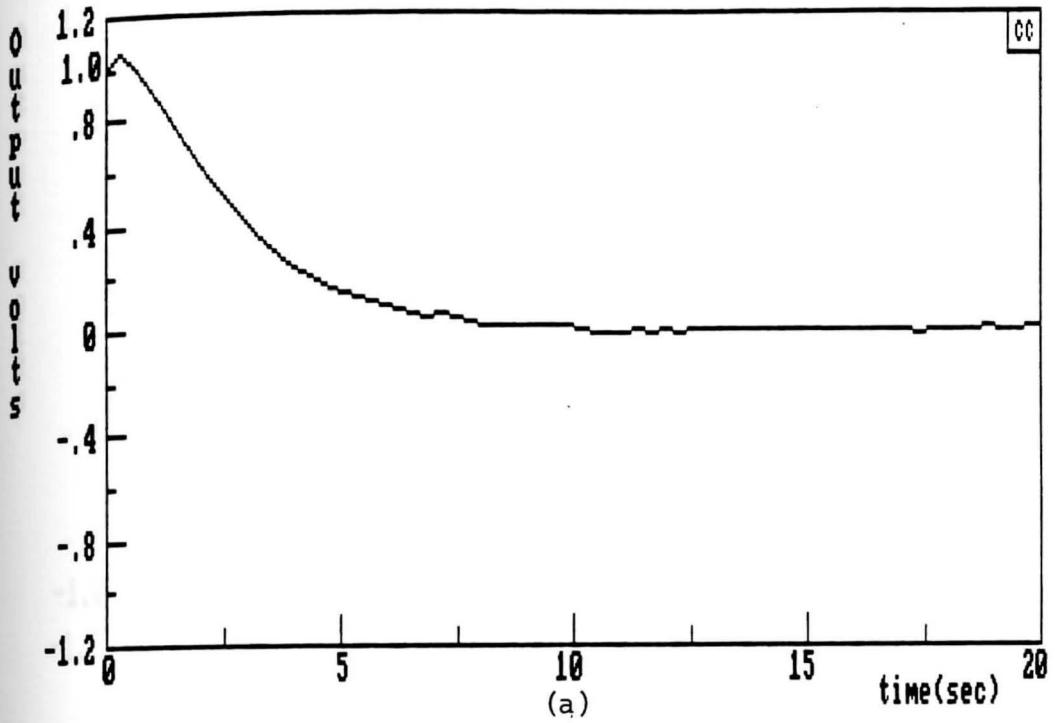


Figure 4.5.6 Control input signals of Example 1 Design 2. (a) Simulated control input signal; (b) Real-time control input signal.



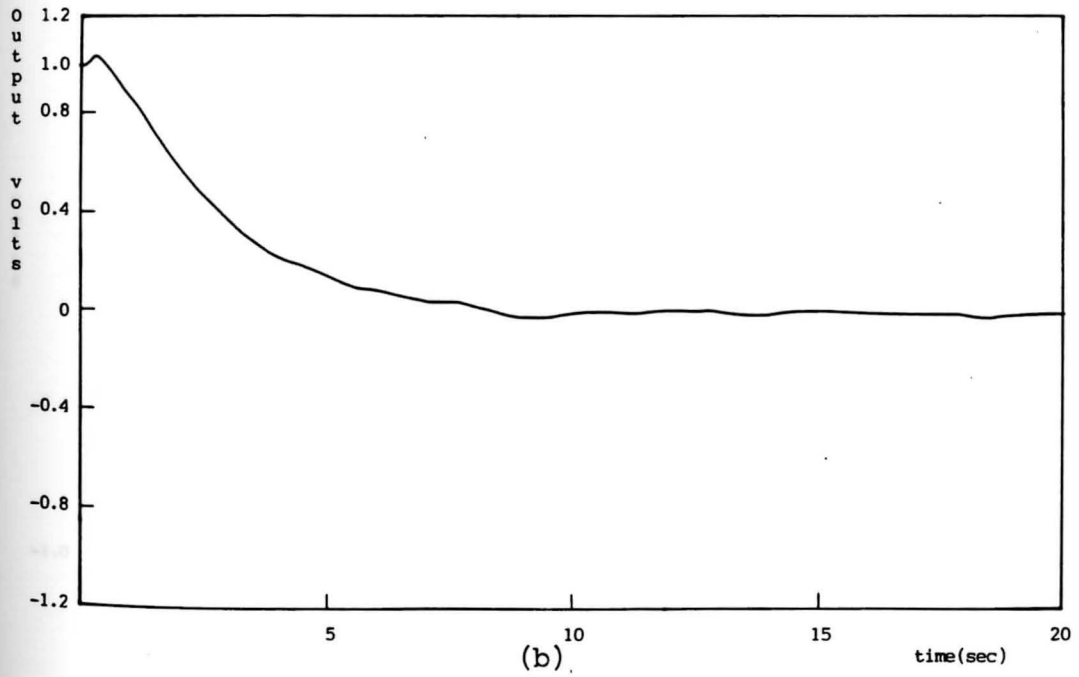
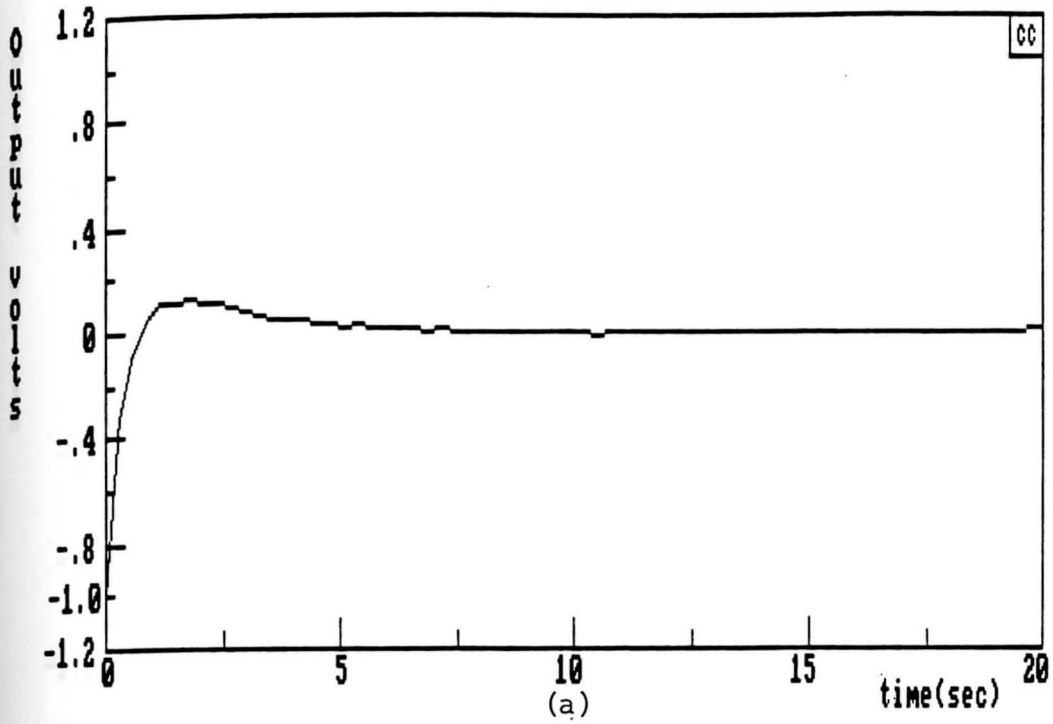


Figure 4.5.7 Output y_1 signals of Example 2 Design 1. (a) Simulated output y_1 signal with noise input; (b) Real-time output y_1 signal.



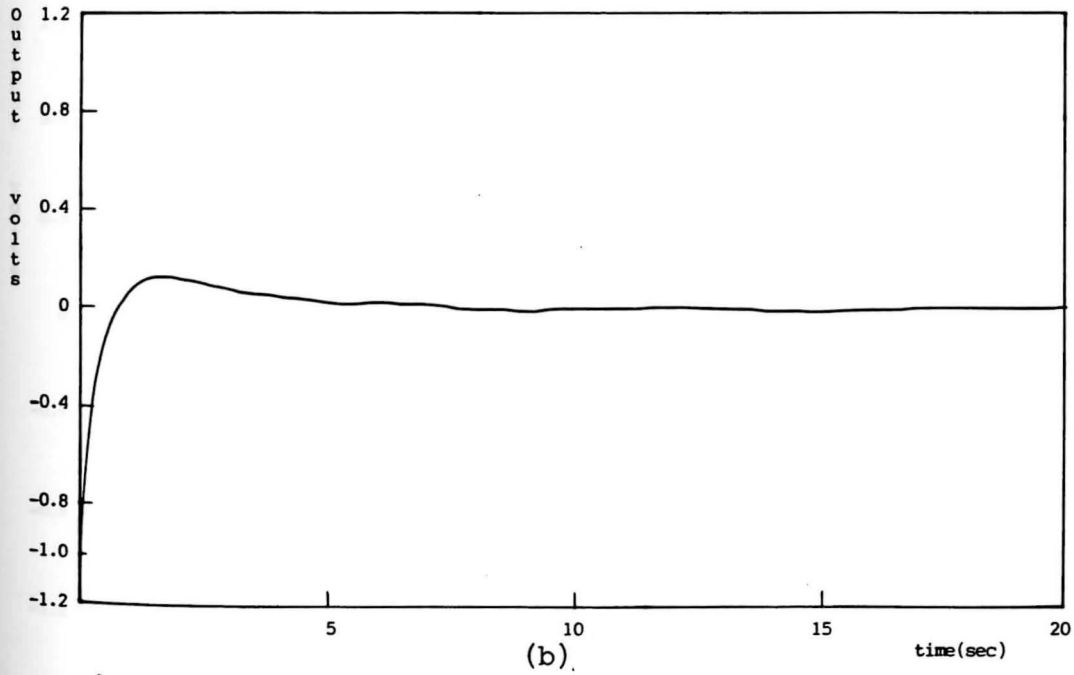
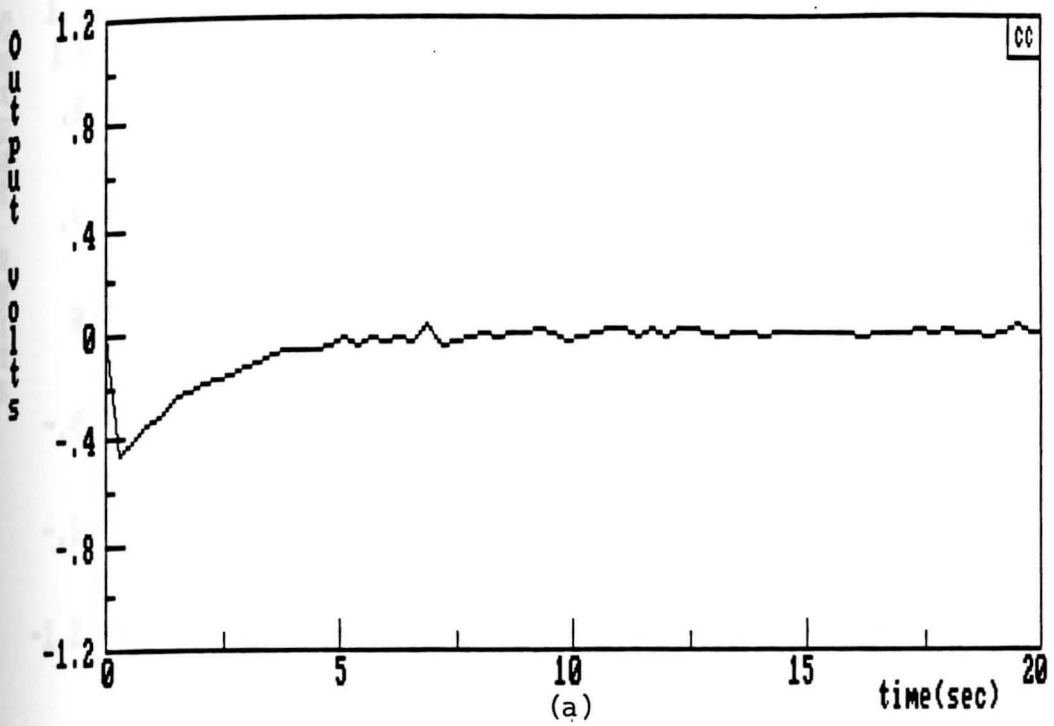
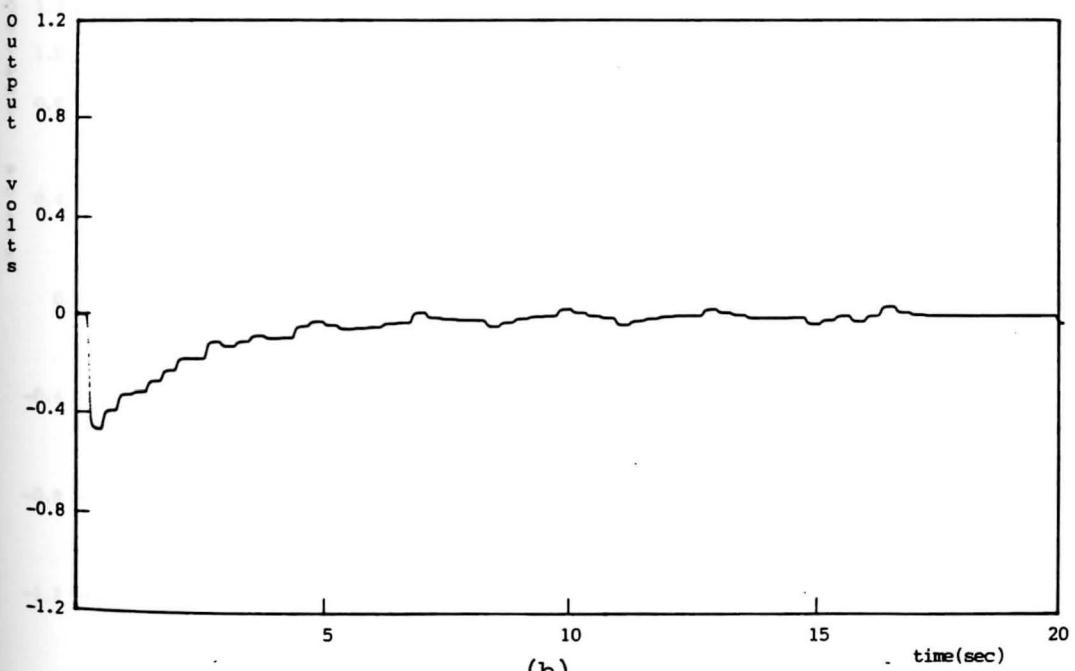


Figure 4.5.8 Output y_2 signals of Example 2 Design 1. (a) Simulated output y_2 signal with noise input; (b) Real-time output y_2 signal.

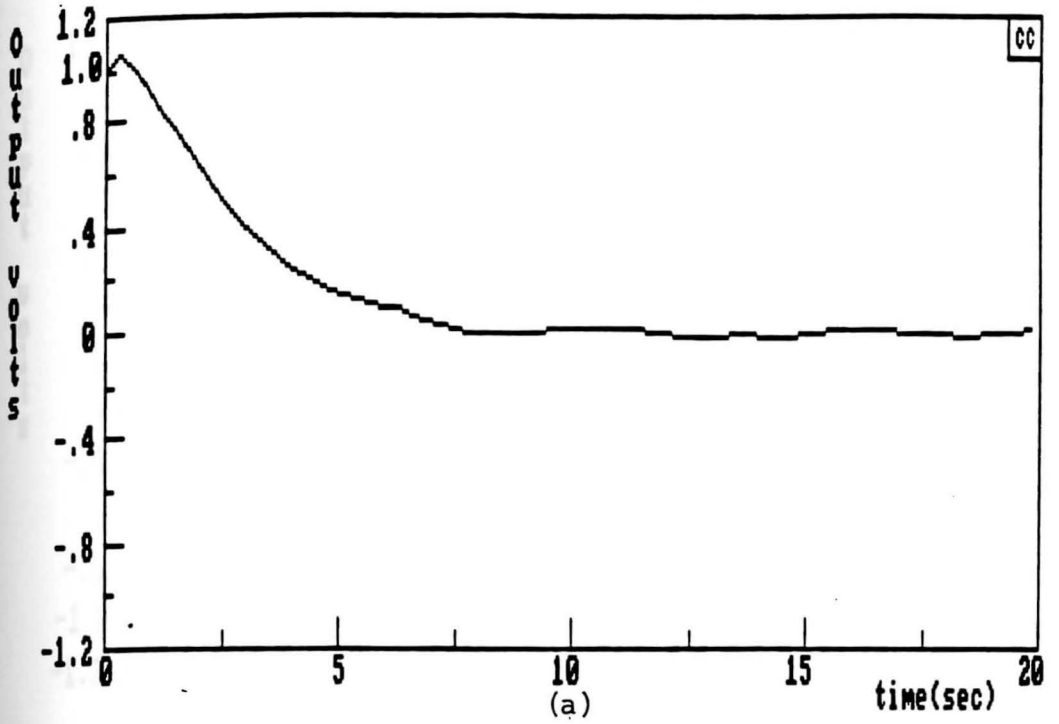


(a)



(b)

Figure 4.5.9 Control input signals of Example 2 Design 1.
(a) Simulated control input signal with noise input;
(b) Real-time control input signal.



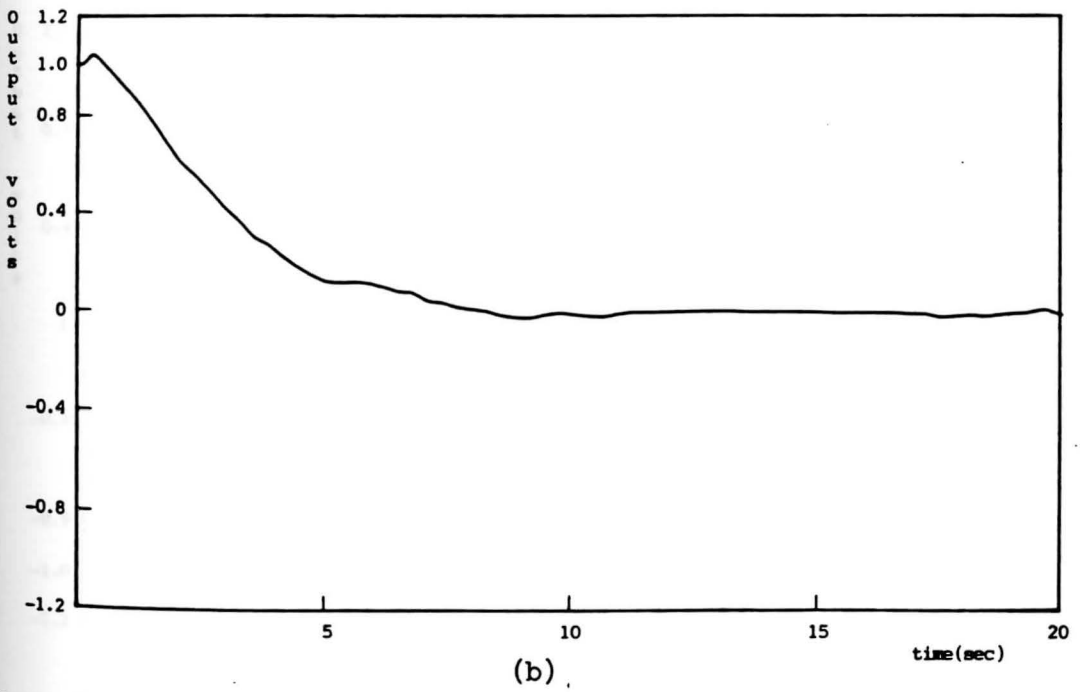
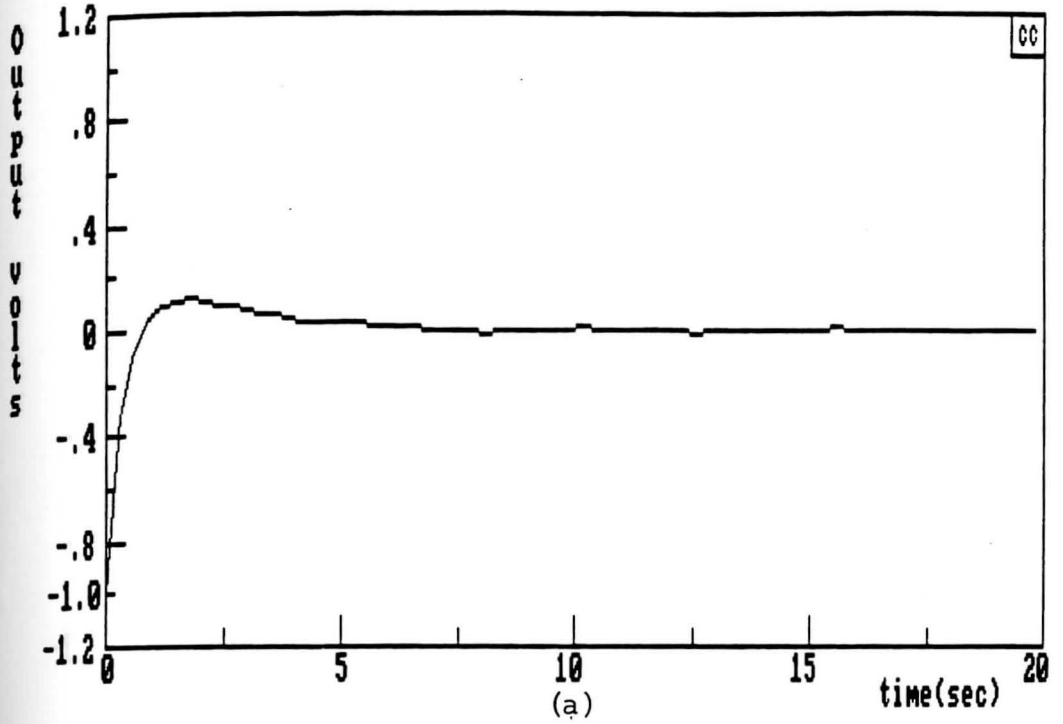
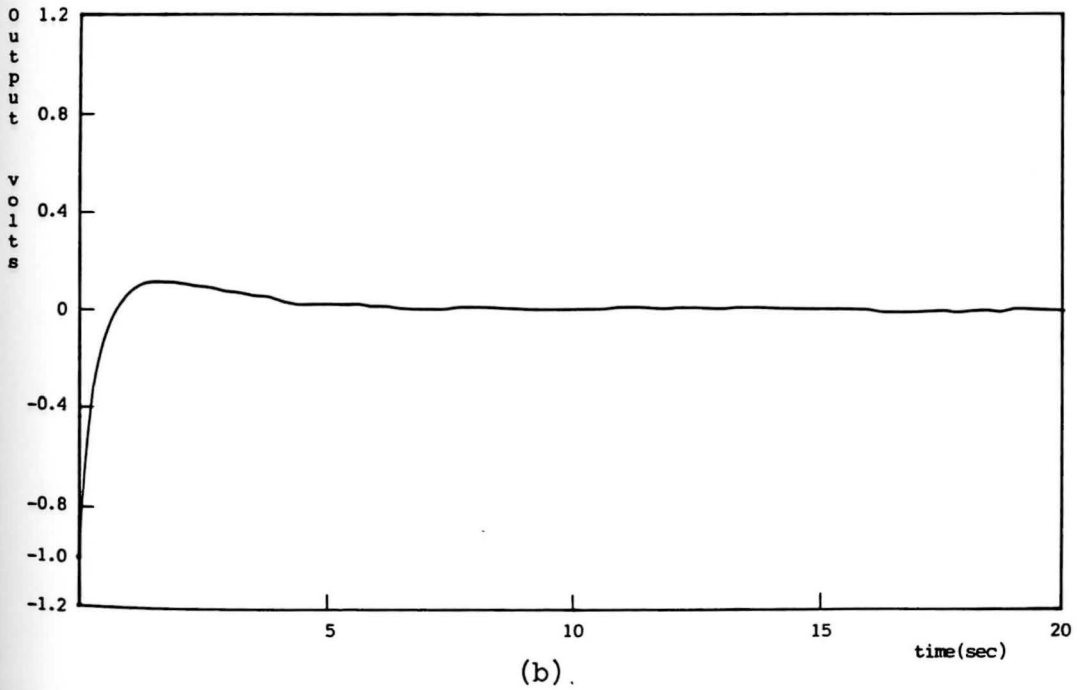


Figure 4.5.10 Output y_1 signals of Example 2 Design 2. (a) Simulated output y_1 signal with noise input; (b) Real-time output y_1 signal.

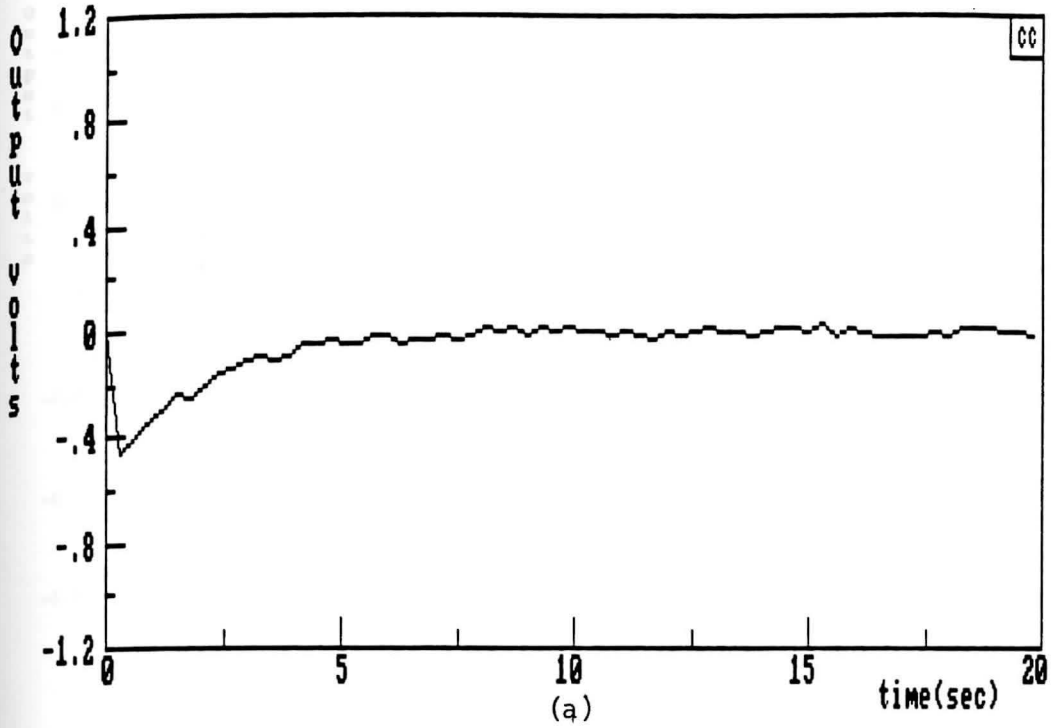


(a)

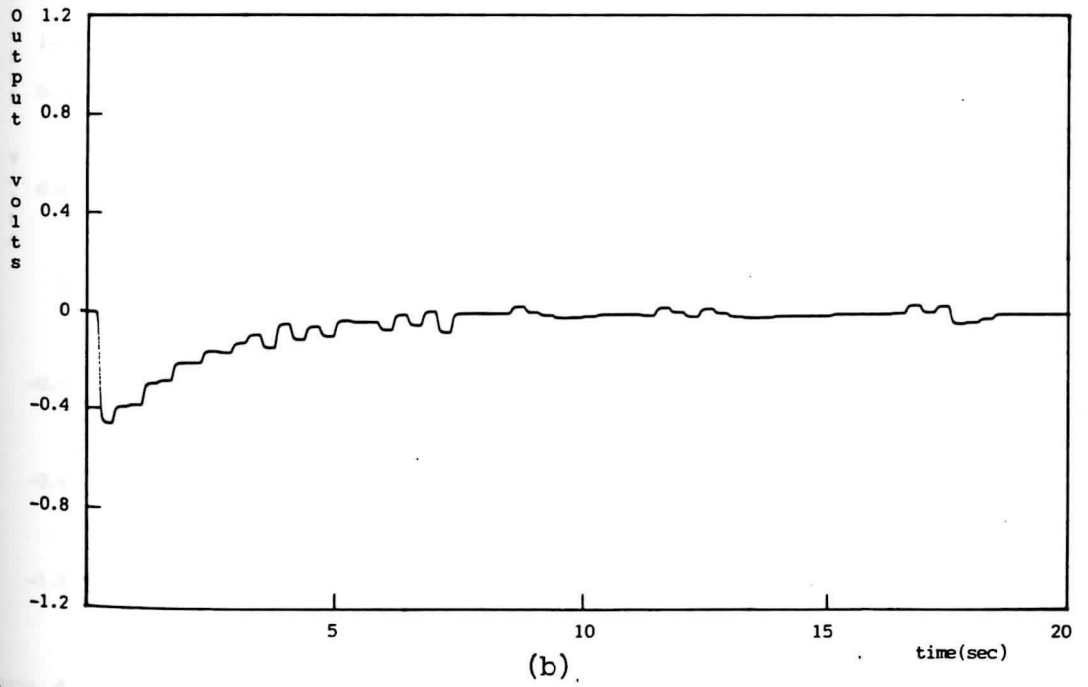


(b)

Figure 4.5.11 Output y_2 signals of Example 2 Design 2. (a) Simulated output y_2 signal with noise input; (b) Real-time output y_2 signal.

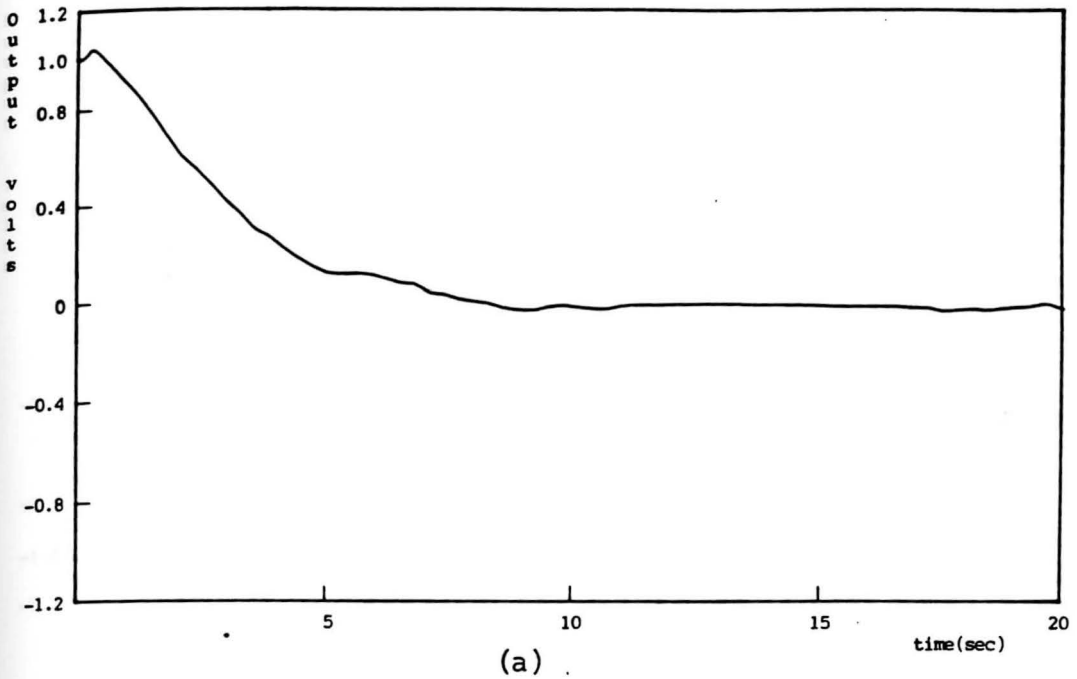


(a)

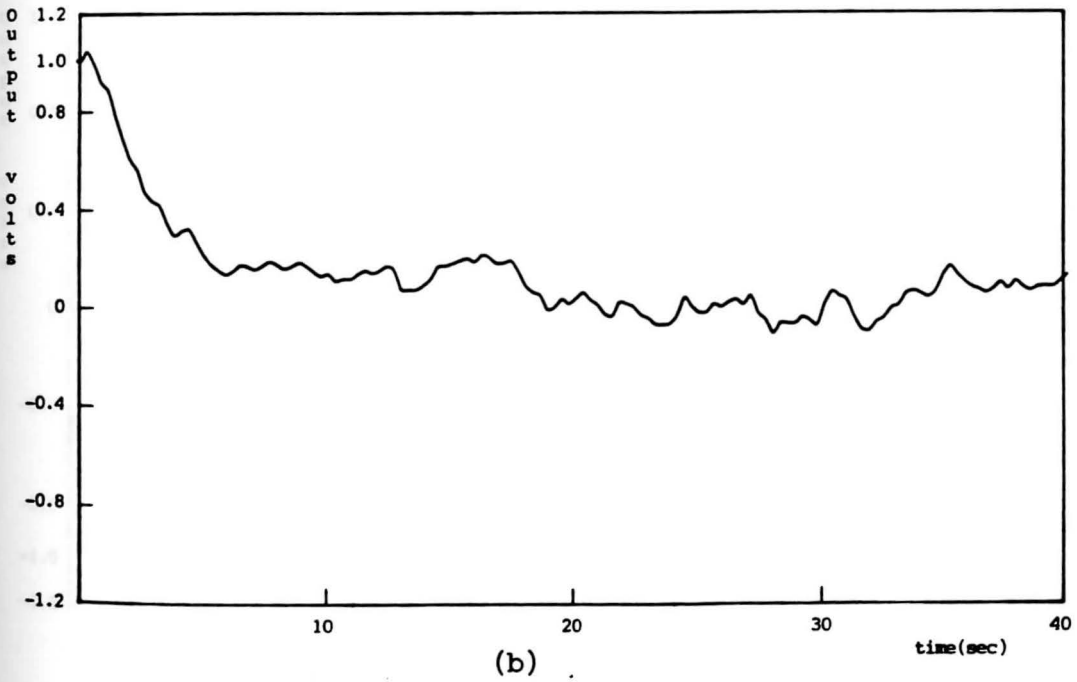


(b)

Figure 4.5.12 Control input signals of Example 2 Design 2.
(a) Simulated control input signal with noise input;
(b) Real-time control input signal.

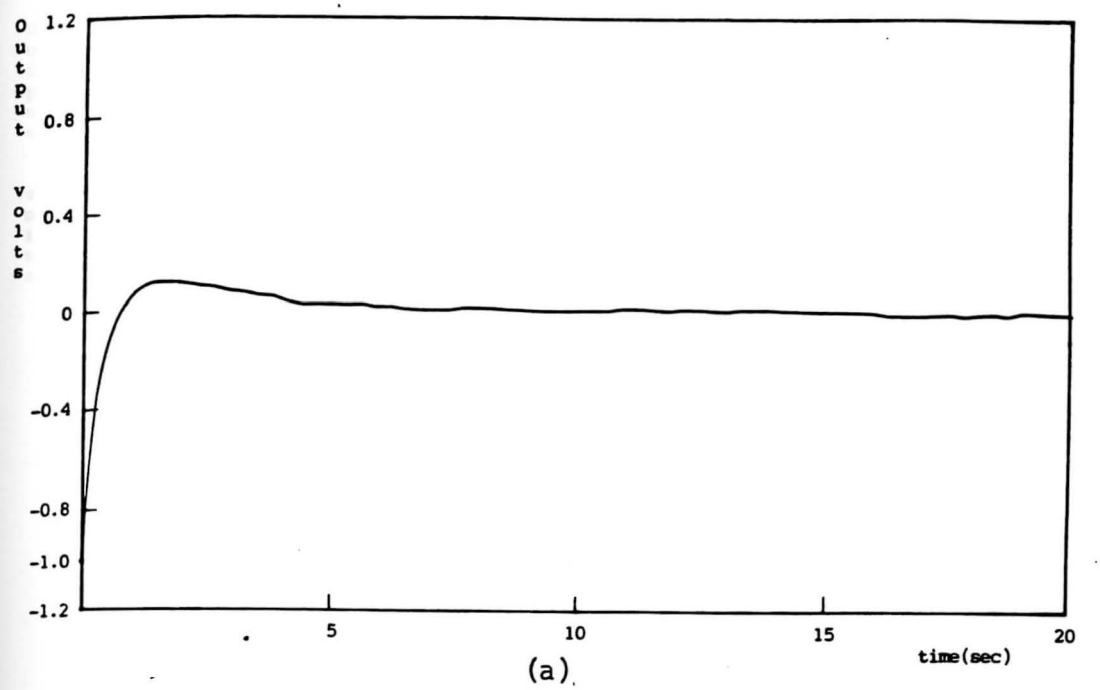


(a)

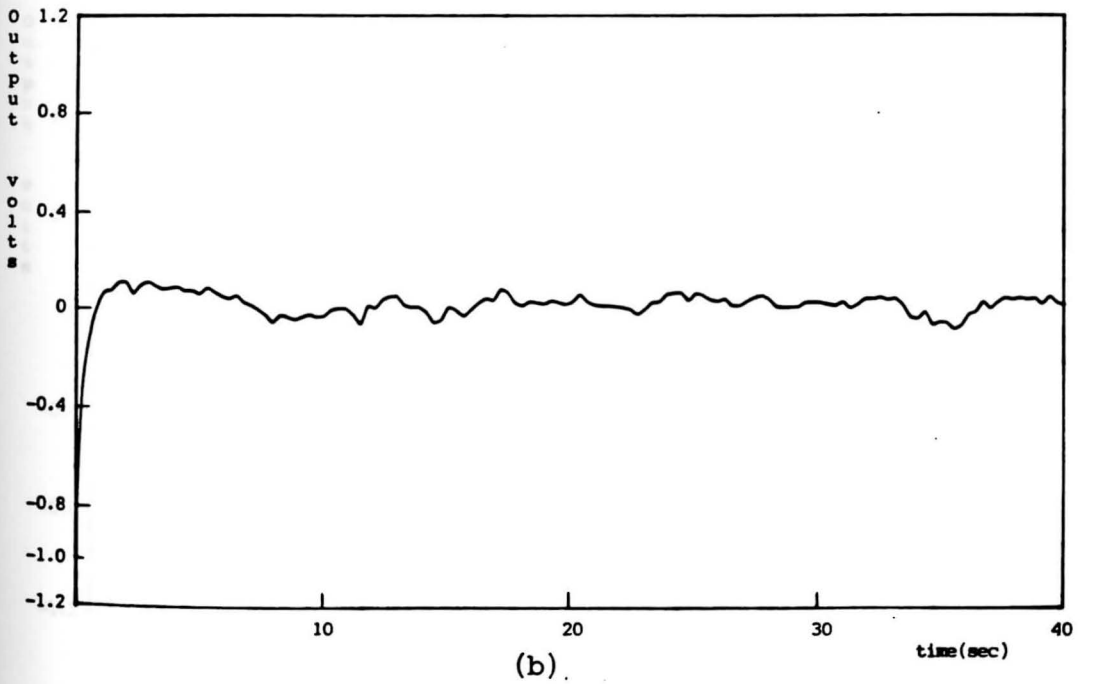


(b)

Figure 4.5.13 (a) Real-time output y_1 signal of Example 2 Design 2;
(b) Real-time output y_1 signal of Example 2 Design 2A.

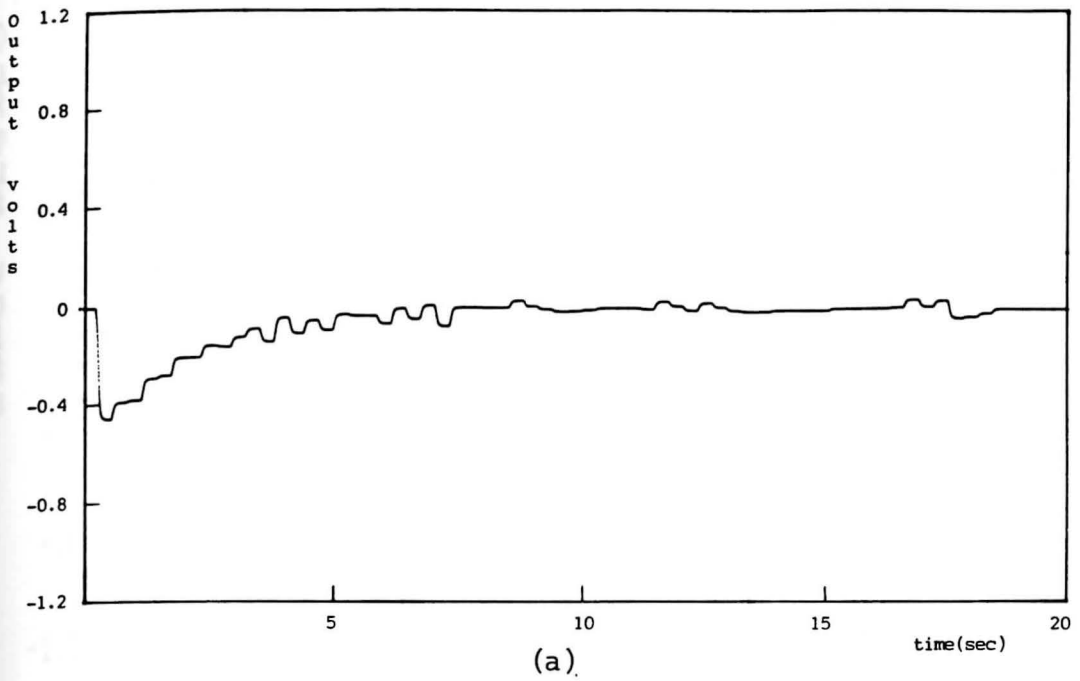


(a)

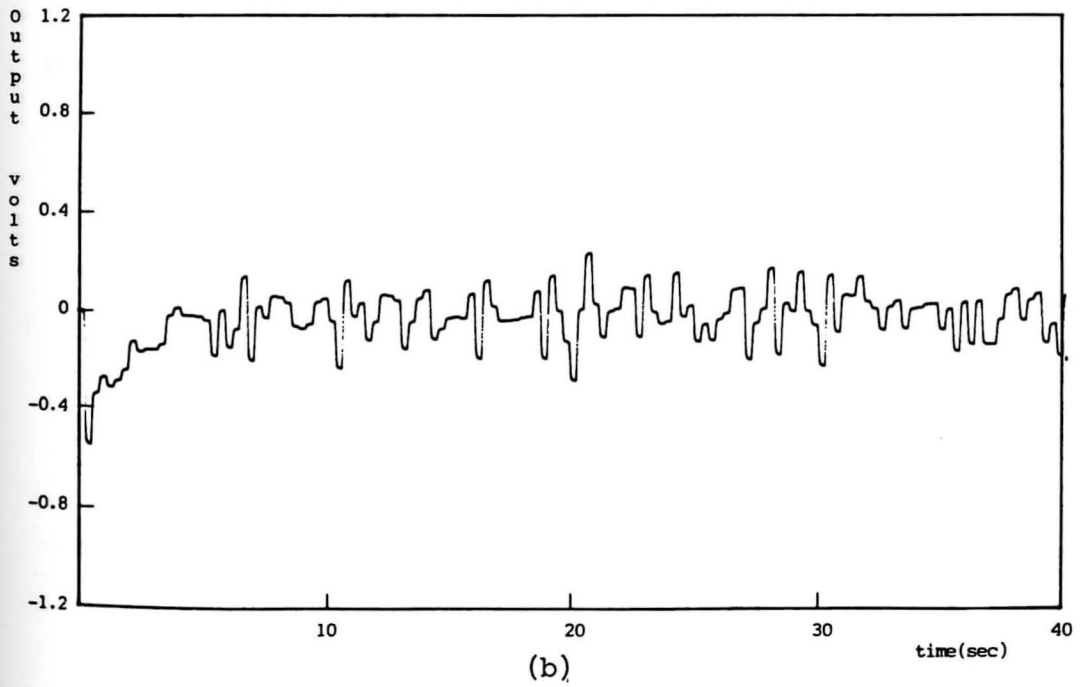


(b)

Figure 4.5.14 (a) Real-time output y_2 signal of Example 2 Design 2;
(b) Real-time output y_2 signal of Example 2 Design 2A.

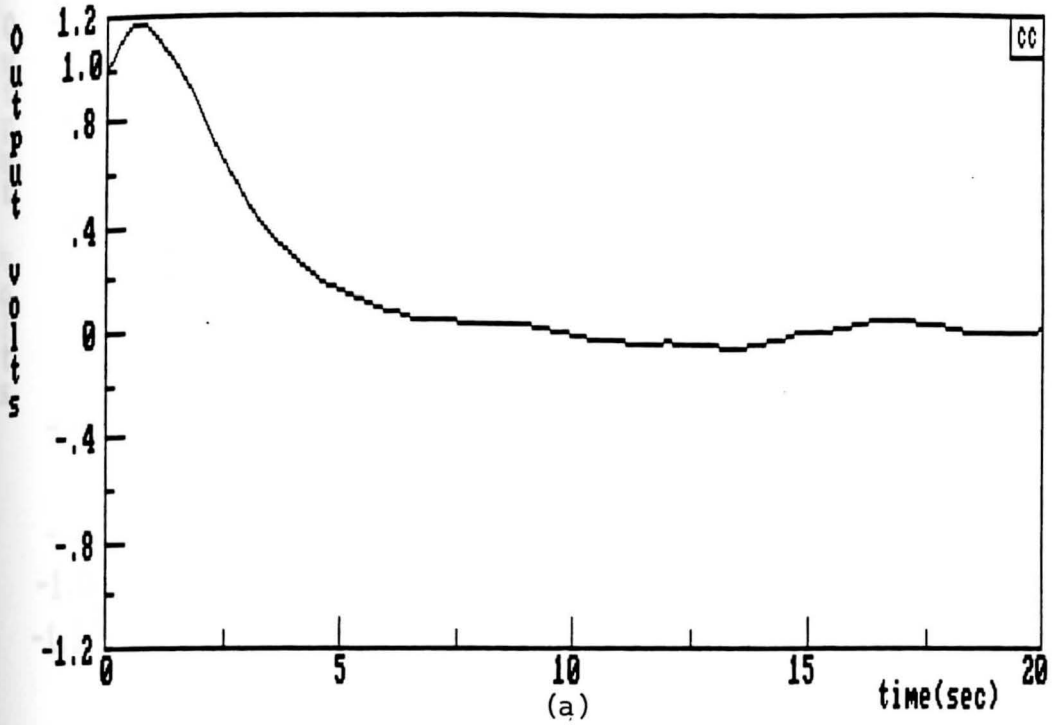


(a)



(b)

Figure 4.5.15 (a) Real-time control input signal of Example 2 Design 2; (b) Real-time control input signal of Example 2 Design 2A.



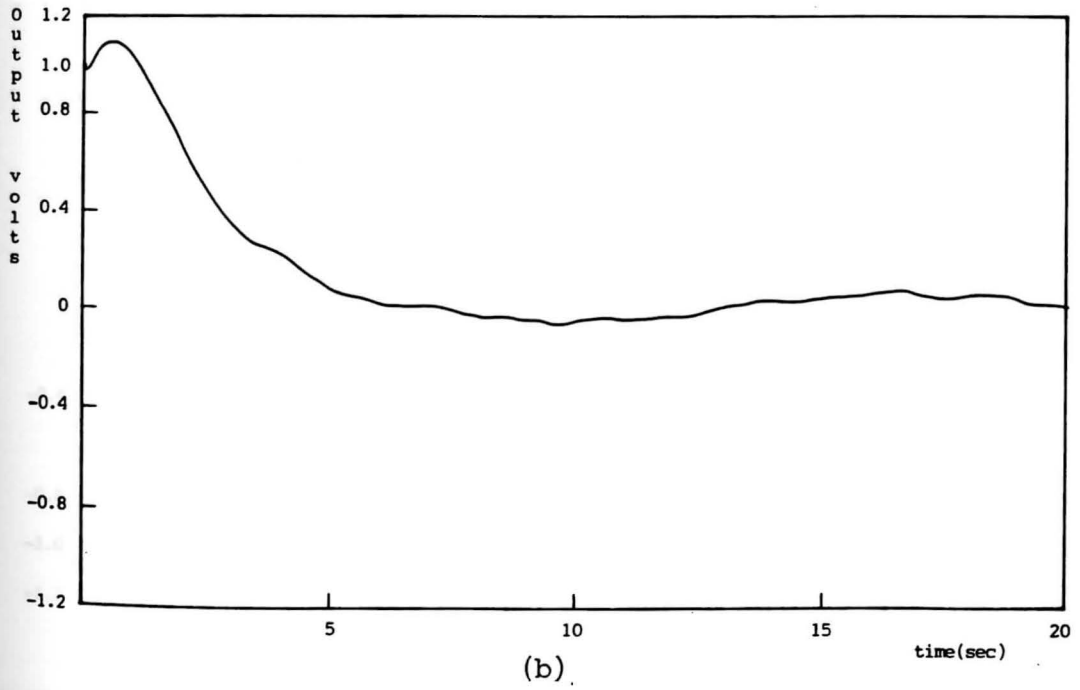
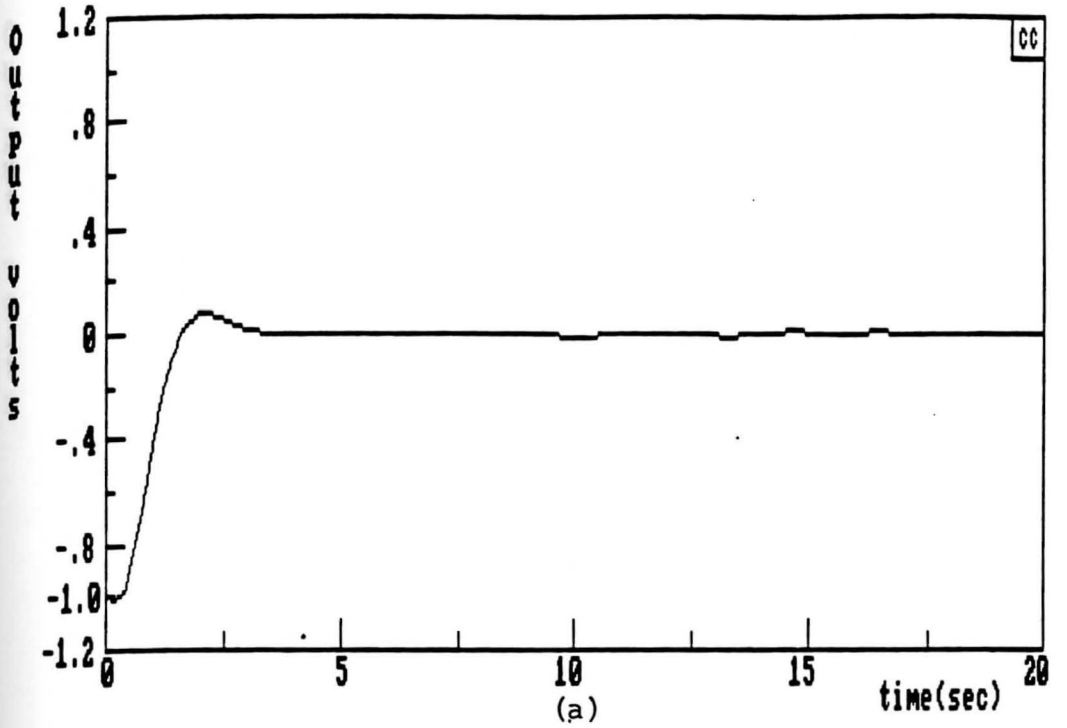


Figure 4.5.16 Output y_1 signals of Example 3 Design 2. (a) Simulated output y_1 signal with noise input; (b) Real-time output y_1 signal.



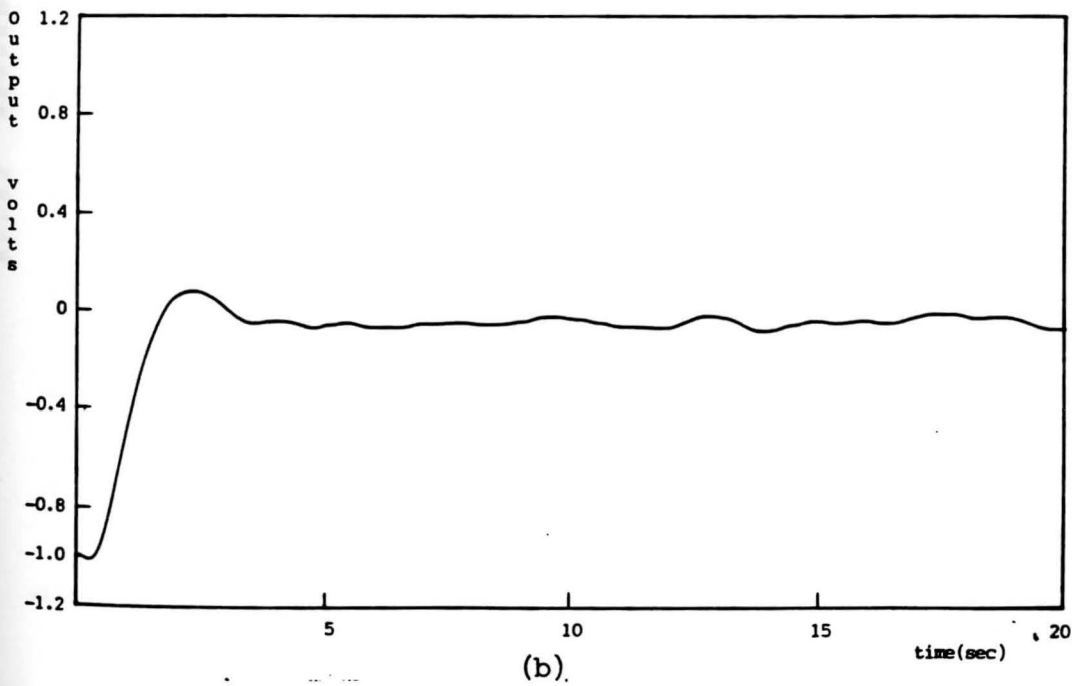
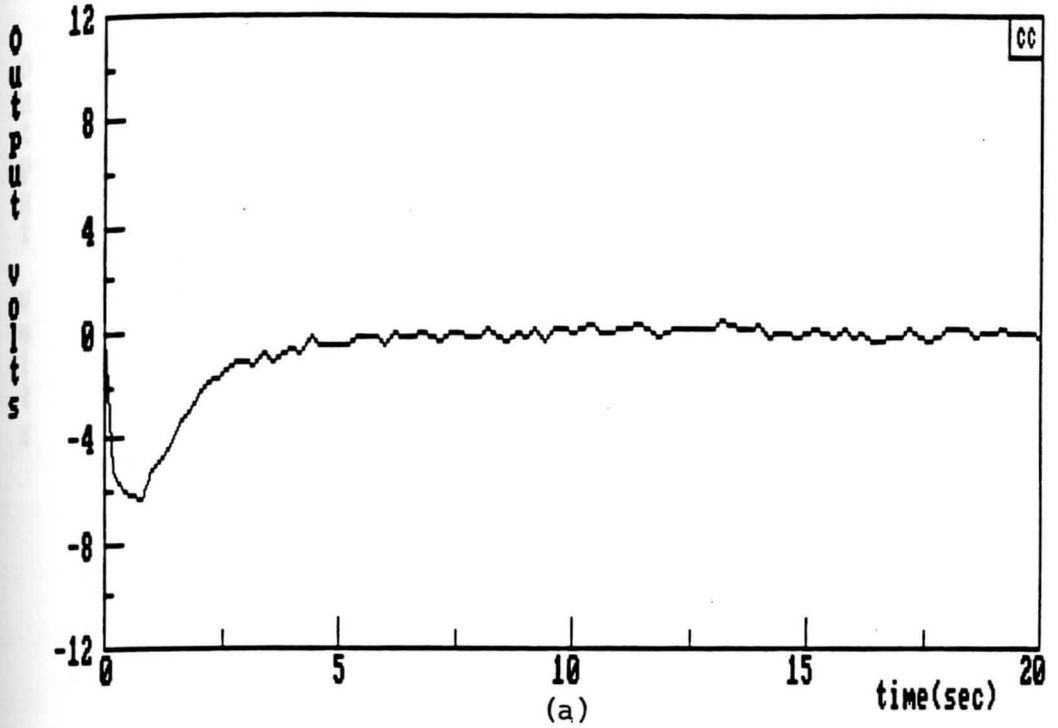


Figure 4.5.17 Output y_2 signals of Example 3 Design 2. (a) Simulated output y_2 signal with noise input; (b) Real-time output y_2 signal.



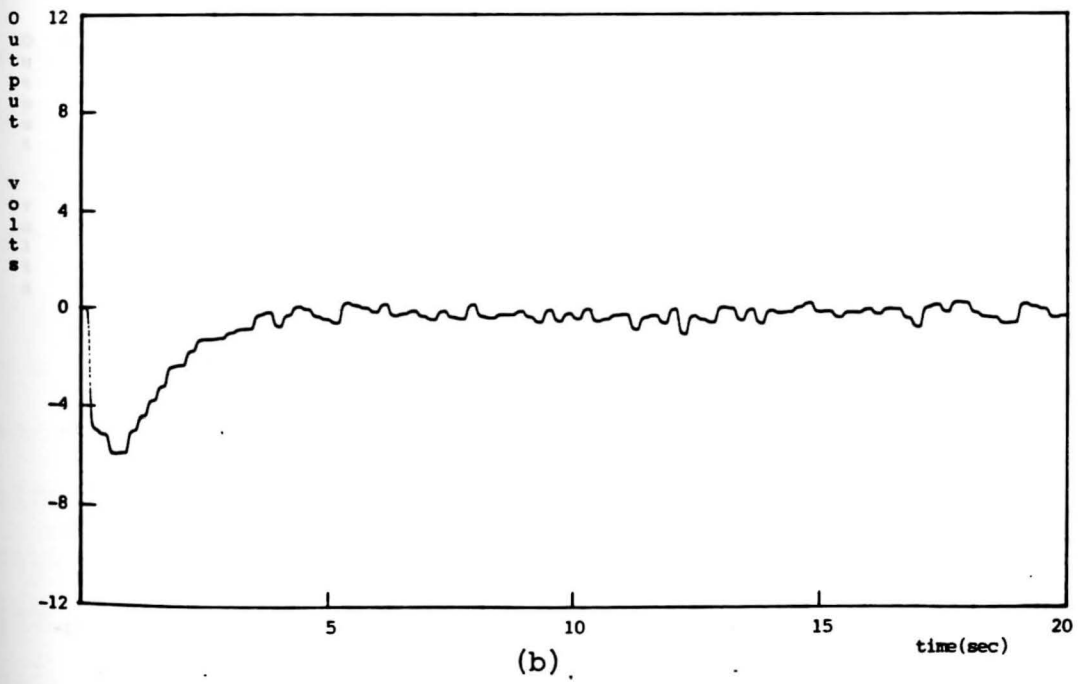
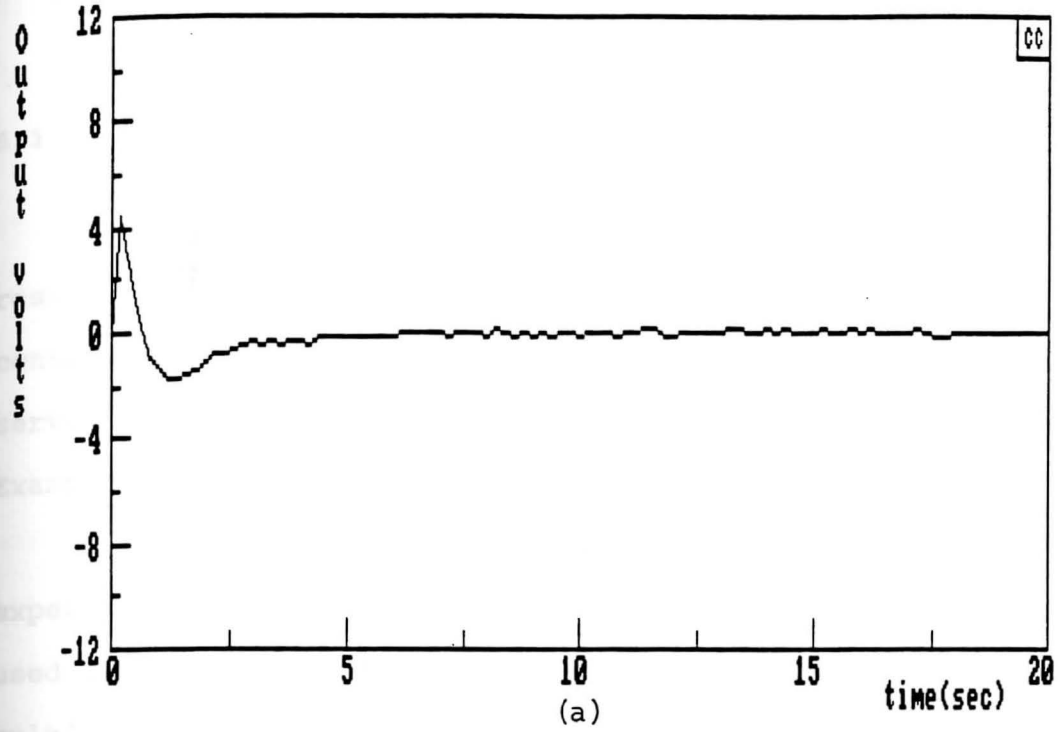
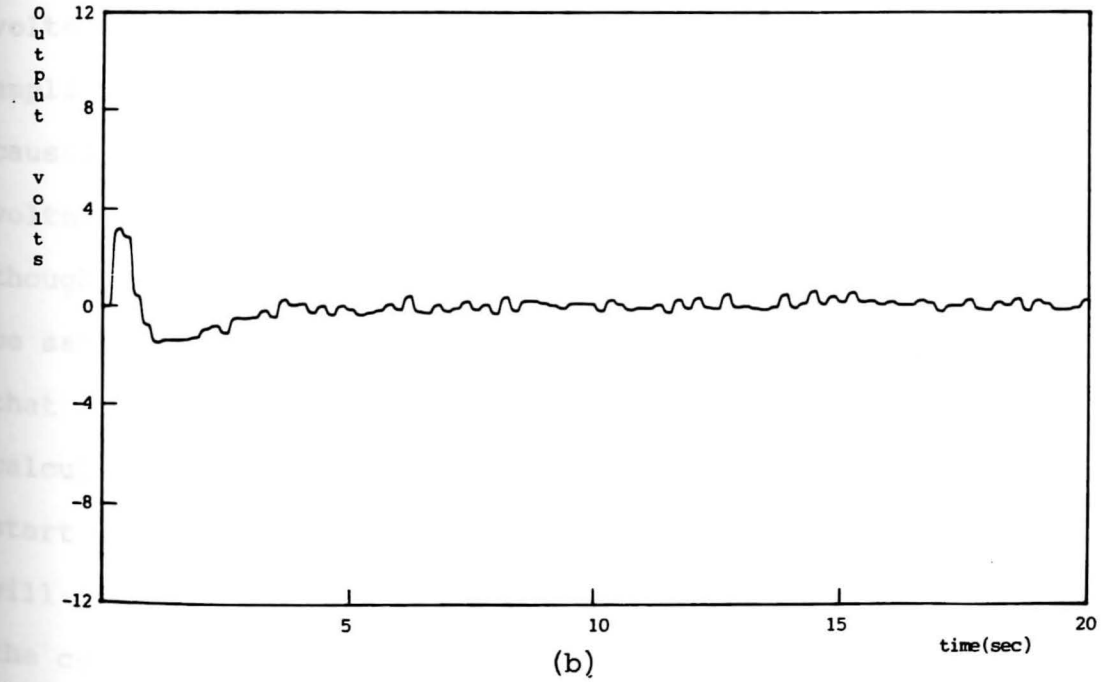


Figure 4.5.18 Control input u_1 signals of Example 3 Design 2.
(a) Simulated control input u_1 signal with noise input;
(b) Real-time control input u_1 signal.



(a)



(b)

Figure 4.5.19 Control input u_2 signals of Example 3 Design 2.
(a) Simulated control input u_2 signal with noise input;
(b) Real-time control input u_2 signal.

CHAPTER V

CONCLUSION

5.1 Discussion of results

Comparing the simulated results to the real-time results indicates that the multirate sampled-data controllers will perform well in the designs used for the servo problem in Example 1 and the regulator problems in Examples 2 and 3.

However, there are some problems discovered during experiments which are due to limitations of the hardware used in this thesis that degrade the capability of the multirate sampled-data controller as follows:

1. Since the GP-6 analog computer restricts the maximum voltage output of each amplifier to ±10 volts reference, an amplifier overload will occur when any control input signal causes an amplifier output voltage to be greater than ±10 volts. The controller will fail to control a plant even though simulated results using the same design come out to be satisfactory. After many experiments, it is observed that if the first control input signal obtained from calculations (the control input signal 0 volt is sent at start of a control process) is greater than ±10 volts, it will cause an amplifier output to overload and will cause the controller to fail to control a plant. Since the state variable feedback gain matrix F directly affects the H and

M matrices in both designs, desired pole locations of the closed loop system must be carefully selected in order not to force the system to go to a steady state too fast.

2. The controllers perform well for specific plants. For a stable plant, there is more flexibility to adjust the feedback gain matrix F for the desired pole locations since the obtained matrix F is not large enough to cause the above problem if the closed-loop poles are not moven too far. For an unstable plant, large gains are required to move unstable poles to the left half plane (or within a unit circle when transformed to discrete-time system) in order to make the closed-loop system stable. This will cause control input signals to be significantly large, and make the amplifier output of GP-6 overload.

3. As mentioned in section 2.5, there are disturbances while implementing the controller. First, an error occurs in measurement of $\bar{Y}(kT_o)$ due to a sampling time error. A sampling time error is made by mis-counting the clock pulses or delays in execution commands even though the counter in the Intel 8254 programmable interval timer used in DAS-8 board is considered to be an accurate counter and the control program compiled by the BASIC compiler is considered to be real-time software. Second, there is a quantization error associated with the conversion. For the 7905 board, an Analog Device Integrated Circuit ADC 574 chip, which is a 12-bits successive approximation

converter, is used. It allows the maximum percent error to be .0244% or 2.44 mV for 10 volts span [21]. This error is interpreted as random noise introduced into the controller.

Observe that the effect of disturbances to the system performance is clearly seen in the results of Example 2 and Example 3 by choices of the H matrix obtained from calculations. It requires time-consuming effort to adjust control parameters in both designs to obtain good system performance.

5.2 Summary

In this thesis, a multirate output sampling mechanism was modified to include multiplexed outputs and multiplexed both inputs and outputs. Two designs to obtain the state transition matrix and the input gain matrix of the multirate sampled-data controller were presented based on theorems in reference [1]. Control algorithms for the multirate sampled-data controller were developed and successfully implemented on an EVEREX 286 microcomputer, an IBM PC/AT compatible, which has a 7905 multiplexed A/D and D/A converter board as an interfacing unit and a DAS-8 data acquisition board as a real-time clock generator. Finally, three application examples were investigated and it was shown that the multirate sampled-data control law can be made equivalent to the state variable feedback control law from a single input single output system to a multiple inputs and outputs system.

5.3 Suggestions for future work

More application examples should be further investigated by using the multirate sampled-data control law, especially the MIMO plants, when hardware which would permit simultaneously updating of all control input signals becomes available. It is also possible to modify the present hardware used in this thesis to allow the update of control input signals simultaneously and use designs presented in section 2.3 for the MIMO plant instead of the design in section 2.4 which requires complicated modifications of the discrete-time state-space dynamic model.

To improve the system performance, the stochastic disturbance modeling could be the subject of another study for noise disturbances to the multirate sampled-data controller together with choosing control parameters to minimize disturbance effects presented in this thesis.

APPENDIX A

The listing of a CC program for Example 1 Design 1.

```
state
pen
  p1                              'Analog plant
exp,p1,p2,.045                    'Discrete-time model at T
unpack,p2,p3,p4,,                 'p3=Φ1, p4=Γ1
analog
exp,p1,p5,.09                     'Discrete-time model at To
unpack,p5,p6,p7,p8,p9             'p6=Φ, p7=Γ, p8=C, p9=[0]
mult,p8,p3,p10
para,p8,p10,p11,2                 'p11 = C̄
inv,p11,p12                       'p12 =   C̄-1
mult,p8,p4,p13
para,p9,p13,p14,2                 'p14 = Ḡ
poleplace,p5,p15                  'p15 = F matrix
mult,p15,p6,p16
mult,p16,p12,p17                  'H = FΦC̄-1
mult,p17,p14,p18
mult,p15,p7,p19
subtr,p18,p19,p20                 'M = FΦC̄-1Ḡ - FΓ
```

To simulate the result.

```
pack,p6,p7,p11,p14,p55        'Plant model with Ȳ(kTo)
pen
  p56                         'p56 = [-H   Nr]
pen
  p57                         'p57 = [1]
pen
  p58                         'p58 = [0   0   0]
pack,p20,p56,p57,p58,p60      'Controller model
para,p55,p60,p61,1
sel,p61,p62,output,3,1,2
feedback,6,p62,p63,3
unpack,p63,p64,p65,,
pen
  p66                         'p66 = C matrix of C.L. system
pen
  p67                         'p67 = D matrix of C.L. system
pack,p64,p65,p66,p67,p68
dsim                          'Digital simulation
  p68
  1,1,3                       'option 1, 1st i/p, Ref= 3v.
  1                           'zero initial condition
  30,1                        'maximum sample
plot,p68.y,a,a
```

The listing of a control program for Example 1 Design 1.

```
10 REM ##########################################################################
20 REM    EXAMPLE I: DESIGN I                                    updated 06/02/90
30 REM       STATE-SPACE MODEL:  .       0    1          0
40 REM                           X   =    0   -1  X   +    1  U
89 REM ##########################################################################
90 REM
100 REM ----------STEP 1:INITIALIZE DAS-8 WITH MODE 0-----------------------
110 MD% = 0
120 BASADR% = &H300              'base address
130 FLAG% = 0
140 CALL DAS8 (MD%,BASADR%,FLAG%)
150 REM-----------STEP 2:SET AND LOAD COUNTER 2,CONFIG. 3 ----------------
160 DIM DIO%(2)
170 MD% = 10                     'mode 10   set config.
180 DIO%(0) = 2                  'select counter 2
190 DIO%(1) = 3                  'set config.3  square wave generator
200 CALL DAS8 (MD%,DIO%(0),FLAG%)
210 MD% = 11                     'mode 11  load timer counter 2
220 DIO%(1) = 378                'count number to generate 10 kHz clock
230 CALL DAS8 (MD%,DIO%(0),FLAG%)
240 REM-----------STEP 3:SET UP COUNTER 0 AND CONFIG. 0---------------------
250 MD% = 10                     'mode 10   set config.
260 DIO%(0) = 0                  'select counter 0
270 DIO%(1) = 0                  'set config 0  pulse high on terminal count
280 CALL DAS8 (MD%,DIO%(0),FLAG%)
290 REM-----------STEP 4:INITIALIZE CONTROL INPUT & GP-6------------------
300 KEY(1) ON:ON KEY(1) GOSUB 440       'to exit use F1
310 X=0:Y=0                      'initialize controller state
315 OUT 798,Y:OUT 799,X
320 OUT 784,2                    'set mux ch AMP#2
322 OUT 785,8                    'set c3 high --> GP-6  IC mode
323 WAIT 786,4,4                 'wait for c2 is high   IC mode
330 REM-----------STEP 5:SET PARAMETERS MODE 11 & 13----------------------
340 MDL% = 11                    'mode 11 load timer counter 0
345 MD% = 13: IP% = 0: FLAG% = 0        'mode 13 check digital i/p
360 DIO%(0) = 0:DIO%(1) = 450    'output sampling period 45 msec
370 INPUT"ENTER REFERENCE VALUE(-10 TO +10 volts)";RK
380 PRINT"INITLAL VALUE OF CONTROL = 0 volt"
390 REM-----------STEP 6:PERFORM CONTROL ROUTINE--------------------------
400 OUT 785,0                     'set GP-6  OP mode
410 WAIT 786,4                    'wait until op mode set
420 GOSUB 1000
430 GOTO 420
440 END
```

```
1000 REM--------------SUBROUTINE  MRC---------------------------------------
1005 CALL DAS8 (MDL%,DIO%(0),FLA6%)
1010 OUT 798,Y                    'set low byte VDAC & start ADC
1020 OUT 797,X                    'set high byte VDAC
1030 WAIT 786,8                   'wait until ECH
1040 GOSUB 3000
1080 YK=(-22.56922*AA)
1085 PRINT " Y1 = ";AA
1090 GOSUB 2000
1095 CALL DAS8 (MDL%,DIO%(0),FLA6%)
1097 OUT 786,2                     'set aux amp#2 & start adc
1098 WAIT 786,8                    'wait until ECH
1100 GOSUB 3000
1110 YK=(24.48057*AA) + YK
1115 PRINT " Y2 = ";AA
1250 UK=(-6.904775E-02*UK)-YK+(1.91135*RK)
1260 PRINT "                              UK = ";UK
1500 BB=(UK*2047)/10              'from binary data word
1510 IF BB < 0 THEN BB=4095+BB    'form 2's complement word
1520 BC=INT(BB/16)                'seperate h.byte
1530 BD=BB-(BC*16)                'seperate l.byte
1540 BD=BD*16                     'shift left 4 bits
1550 X=BC:Y=BD                    'prepare to send DAC
1560 IF Y > 255 THEN Y=255
2000 REM--------------SUBROUTINE CHECK-TIMING-------------------------------
2020 CALL DAS8 (MD%,IP%,FLA6%)
2030 IF IP% = 7 THEN PRINT "SETTING TIME TOO SHORT":END
2040 CALL DAS8 (MD%,IP%,FLA6%)
2050 IF IP% <> 7 THEN GOTO 2040
2060 RETURN
3000 REM --------------SUBROUTINE SCALE-DATA---------------------------------
3010 QA=INP(787)*16                 'get data h.byte & scale
3020 QB=INP(786)/16                 'get data l.byte & scale
3030 AA=QA+INT(QB)                  'drop sense bits & combine H&L
3040 IF AA > 2047 THEN AA=AA-4095   'scale negative data
3050 AA=(AA/2047)*10                'scale to 10 v.ref
3060 RETURN
```

APPENDIX B

The listing of a CC program for Example 1 Design 2.

```
state
pen
p1                              'Analog plant
exp,p1,p2,.03                   'Discrete-time model at T
unpack,p2,p3,p4,,               'p3=Φ1, p4=Γ1
analog
exp,p1,p5,.06                   'Discrete-time model at 2T
unpack,p5,p6,p7,,               'p6=Φ2, p7=Γ2
analog
exp,p1,p8,.09                   'Discrete-time model at To
unpack,p8,p9,p10,p11,p12        'p9=Φ, p10=Γ, p11=C, p12=[0]
mult,p11,p3,p13
mult,p11,p6,p14
para,p11,p13,p15,2
para,p15,p14,p16,2              'p16 = C̄
mult,p11,p4,p17
mult,p11,p7,p18
para,p11,p17,p19,2
para,p19,p18,p20,2              'p20 = Ḡ
para,p16,p20,p21,3              'p21 =  [C̄   Ḡ]
inv,p21,p22                     'p22 =        [C̄   Ḡ]^-1
poleplace,p8,p23                'p23 =  F matrix
mult,p23,p9,p24
mult,p23,p10,p25
pen
p26                             'Select desired M
add,p25,p26,p27
para,p24,p27,p28,3
mult,p28,p22,p29                'H = [FΦ  FΓ+M][C̄   Ḡ]^-1
```

To simulate the result.

```
pack,p9,p10,p16,p20,p55     'Plant model with Ȳ(kTo)
pen
  p56                       'p56 = [-H   Nr]
pen
  p57                       'p57 = [1]
pen
  p58                       'p58 = [0  0  0  0]
pack,p26,p56,p57,p58,p60    'Controller model
sel,p61,p62,output,4,1,2,3
feedback,6,p62,p63,4
unpack,p63,p64,p65,,
pen
  p66                       'p66 = C matrix of C.L. system
pen
```

```
  p67                          'p67 = D matrix of C.L. system
pack,p64,p65,p66,p67,p68
dsim
  p68                          'same option as design 1
plot,p68.y,a,a
```

The listing of a control program for Example 1 Design 2.

```
10 REM ##########################################################################
20 REM    EXAMPLE I: DESIGN II                                    updated 06/02/90
30 REM       STATE VARIABLE MODEL:  .    0    1                  0
40 REM                              X=   0   -1   X   +    1     U
89 REM ##########################################################################
90 REM
100 REM ----------STEP 1:INITIALIZE DAS-8 WITH MODE 0----------
110 MD% = 0
120 BASADR% = &H300              'base address
130 FLAG% = 0
140 CALL DAS8 (MD%,BASADR%,FLAG%)
150 REM-----------STEP 2:SET UP COUNTER 2 AND CONFIG. 3---------------------
160 DIM DIO%(2)
170 MD% = 10                     'mode 10   set config.
180 DIO%(0) = 2                  'select counter 2
190 DIO%(1) = 3                  'set config.3  square wave generator
200 CALL DAS8 (MD%,DIO%(0),FLAG%)
210 MD% = 11                     'mode 11  load timer counter 2
220 DIO%(1) = 378                'count number to generate 10 kHz clock
230 CALL DAS8 (MD%,DIO%(0),FLAG%)
240 REM-----------STEP 3:SET UP COUNTER 0 AND CONFIG. 0---------------------
250 MD% = 10                     'mode 10   set config.
260 DIO%(0) = 0                  'select counter 0
270 DIO%(1) = 0                  'set config 0  pulse high on terminal count
280 CALL DAS8 (MD%,DIO%(0),FLAG%)
290 REM-----------STEP 4:INITIALIZE CONTROL INPUT &GP-6---------------------
300 KEY(1) ON:ON KEY(1) GOSUB 440     'to exit program, use F1
310 X=0:Y=0:X1K=0:X2K=0               'initialize controller state
315 OUT 798,Y:OUT 799,X
320 OUT 784,2                         'set mux ch AMP#2
322 OUT 785,8                         'set c3 high --> GP-6  IC mode
323 WAIT 786,4,4                      'wait for c2 is high   IC mode
330 REM-----------STEP 5:SET PARAMETERS MODE 11 & 13-----------------------
340 MDL% = 11                         'mode 11 load timer counter 0
345 MD% = 13: IP% = 0: FLAG% = 0      'mode 13 check digital i/p
360 DIO%(0) = 0:DIO%(1) = 300         'output sampling period = 30 msec
370 INPUT"ENTER REFERENCE VALUE(-10 TO +10 volts)";RK
380 PRINT"INITLAL VALUE OF CONTROL = 0 volt"
390 REM-----------STEP 6: PERFORM CONTROL ROUTINE--------------------------
400 OUT 785,0                         'set GP-6  OP mode
410 WAIT 786,4                        'wait until op mode set
420 GOSUB 1000
430 GOTO 420
440 END
```

```
1000 REM---------------SUBROUTINE MRC---------------------------------------
1005 CALL DAS8 (MDL%,DIO%(0),FLA6%)
1010 OUT 798,Y                    'set low byte VDAC & start ADC
1020 OUT 797,X                    'set high byte VDAC
1030 WAIT 786,8                   'wait until ECH
1040 GOSUB 3000
1080 YK=(-16.98622*AA)
1085 PRINT " Y1 = ";AA
1090 GOSUB 2000
1095 CALL DAS8 (MDL%,DIO%(0),FLA6%)
1097 OUT 786,2                    'set mux amp#2 & start adc
1098 WAIT 786,8                   'wait until ECH
1100 GOSUB 3000
1110 YK=(.8124516*AA) + YK
1115 PRINT " Y2 = ";AA
1120 GOSUB 2000
1130 CALL DAS8 (MDL%,DIO%(0),FLA6%)
1140 OUT 786,2                    'set mux amp#2 and start adc
1150 WAIT 786,8                   'wait until ECH
1160 GOSUB 3000                   'get sampled-data
1170 YK=(18.08512*AA) + YK
1180 PRINT " Y3 = ";AA
1250 UK=(-6.119289E-02*UK)-YK+(1.91135*RK)
1260 PRINT "                              UK = ";UK
1500 BB=(UK*2047)/10              'from binary data word
1510 IF BB < 0 THEN BB=4095+BB    'form 2's complement word
1520 BC=INT(BB/16)                'seperate h.byte
1530 BD=BB-(BC*16)                'seperate l.byte
1540 BD=BD*16                     'shift left 4 bits
1550 X=BC:Y=BD                    'prepare to send DAC
1560 IF Y > 255 THEN Y=255
2000 REM---------------SUBROUTINE CHECK-TIMING------------------------------
2020 CALL DAS8 (MD%,IP%,FLA6%)
2030 IF IP% = 7 THEN PRINT "SETTING TIME TOO SHORT":END
2040 CALL DAS8 (MD%,IP%,FLA6%)
2050 IF IP% <> 7 THEN GOTO 2040
2060 RETURN
3000 REM---------------SUBROUTINE SCALE-DATA--------------------------------
3010 QA=INP(787)*16                'get data h.byte & scale
3020 QB=INP(786)/16                'get data l.byte & scale
3030 AA=QA+INT(QB)                 'drop sense bits & combine H&L
3040 IF AA > 2047 THEN AA=AA-4095  'scale negative data
3050 AA=(AA/2047)*10               'scale to 10 v.ref
3060 RETURN
```

APPENDIX C

The listing of a CC program for Example 2 Design 1.

```
state
pen
  p1                          'Analog plant model
exp,p1,p2,.075                'Discrete-time model at T
unpack,p2,p3,p4,,             'p3 = Φ1, p4 = Γ1
analog
exp,p1,p5,.150                'Discrete-time model at 2T
unpack,p5,p6,p7,,             'p6 = Φ2, p7 = Γ2
analog
exp,p1,p8,.225                'Discrete-time model at 3T
unpack,p8,p9,p10,,            'p9 = Φ3, p10 = Γ3
analog
exp,p1,p11,.3                 'Discrete-time model at To
unpack,p11,p12,p13            'p12 = Φ, p13 = Γ
pen
  p14                         'p14 = C1
pen
  p15                         'p15 = C2
pen
  p16                         'p16 = [0]
mult,p14,p3,p17
mult,p15,p6,p18
mult,p15,p9,p19
para,p14,p17,p20,2
para,p20,p18,p21,2
para,p21,p19,p22,2            'p22 = C̄
mult,p14,p4,p23
mult,p15,p7,p24
mult,p15,p10,p25
para,p16,p23,p26,2
para,p26,p24,p27,2
para,p27,p25,p28,2            'p28 = Ḡ
inv,p22,p29                   'p29 =  C̄-1
poleplace,p11,p30             'P30 = F
mult,p30,p12,p31
mult,p31,p29,p32              'p32 = FΦC̄-1 = H
mult,p32,p28,p33
mult,p30,p13,p34
subtr,p33,p34,p35             'p35 = FΦC̄-1Ḡ - FΓ = M
```

To simulate the result.

```
pen
  p55                         'Discrete plant model includes
                               noise input
pen
```

```
  p60                          'controller model
para,p55,p60,p61,1
sel,p61,p62,input,1,3,4,5,6,2  'Prepare inputs to be
                                feedback
sel,p62,p63,output,5,1,2,3,4   'Prepare outputs to be
                                feedback
feedback,6,p63,p64,6
unpack,p64,p65,p66,,
pen
  p67                          'p67 = C matrix of C.L.  system
pen
  p68                          'p68 = D matrix of C.L.  system
pack,p65,p66,p67,p68,p69
dsim
  p69
  4,1,σ,0                      'Guassian noise i/p with zero
                                mean, standard deviation σ and
                                random seed
  2,p100                       'Set intial condition
  30,1                         'Set simulation time
plot,p69.y,a,a
```

The listing of a control program for Example 2 Design 1.

```
10 REM ##########################################################################
20 REM    EXAMPLE II: DESIGN I.  FRAME  PERIOD=.3 ,  updated 06/04/90
30 REM       STATE-SPACE MODEL:     .  .001  0   0   0         1
40 REM                              X=   2 -1   0   0  X  +    2   U
50 REM                                  -1   0  -3   0        -1
60 REM                                   1   0   0  -2         1
65 REM
70 REM                  Y =   0   1   1   0    X
75 REM                        0   0   0   1
89 REM ##########################################################################
90 REM
100 REM -----------STEP 1:INITIALIZE DAS-8 WITH MODE 0----------------------
110 MD% = 0
120 BASADR% = &H300              'base address
130 FLAG% = 0
140 CALL DAS8 (MD%,BASADR%,FLAG%)
150 REM------------STEP 2:SET UP COUNTER 2 AND CONFIG. 3---------------------
160 DIM DIO%(2)                                        .
170 MD% = 10                     'mode 10   set config.
180 DIO%(0) = 2                  'select counter 2
190 DIO%(1) = 3                  'set config.3  square wave generator
200 CALL DAS8 (MD%,DIO%(0),FLAG%)
210 MD% = 11                     'mode 11  load timer counter 2
220 DIO%(1) = 378                'count number to generate 10 kHz clock
230 CALL DAS8 (MD%,DIO%(0),FLAG%)
240 REM------------STEP 3:SET UP COUNTER 0 AND CONFIG. 0---------------------
250 MD% = 10                     'mode 10   set config.
260 DIO%(0) = 0                  'select counter 0
270 DIO%(1) = 0                  'set config 0  pulse high on terminal count
280 CALL DAS8 (MD%,DIO%(0),FLAG%)
290 REM------------STEP 4:INITIALIZE CONTROL INPUT & GP-6--------------------
300 KEY(1) ON:ON KEY(1) GOSUB 440     ' to exit program, use F1
310 X=0:Y=0                      'initialize controller state
315 OUT 798,Y:OUT 799,X
320 OUT 784,8                    'set mux ch A4
322 OUT 785,8                    'set c3 high --> GP-6  IC mode
323 WAIT 786,4,4                 'wait for c2 is high   IC mode
330 REM------------STEP 5:SET PARAMETERS MODE 11 & 13------------------------
340 MDL% = 11                         'mode 11 load timer counter 0
345 MD% = 13: IP% = 0: FLAG% = 0      'mode 13 check digital i/p
360 DIO%(0) = 0:DIO%(1) = 750         'subinterval period = 75 msec
370 INPUT"PRESS [ENTER] TO START-------------->";R1
380 PRINT"INITLAL VALUE OF CONTROL = 0 volt"
390 REM------------STEP 6:PERFORM CONTROL ROUTINE---------------------------
400 OUT 785,0                     'set GP-6  OP mode
410 WAIT 786,4                    'wait until op mode set
420 GOSUB 1000
430 GOTO 420
440 END
```

```
1000 REM------------SUBROUTINE MRC-----------------------------------------
1005 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1010 OUT 798,Y                    'set low byte VDAC & start ADC
1020 OUT 797,X                    'set high byte VDAC
1030 WAIT 786,8                   'wait until ECH
1040 GOSUB 3000
1080 YK=(1.30434E-06*AA)
1085 PRINT "Y1=";AA
1090 GOSUB 2000
1095 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1097 OUT 786,8                     'set mux A4 & start adc
1098 WAIT 786,8                    'wait until ECH
1100 GOSUB 3000
1110 YK=(-1.632689E-06*AA)+YK
1115 PRINT "Y2=";AA
1117 OUT 784,10                    'set mux ch A5 for next sampling
1120 GOSUB 2000
1130 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1140 OUT 786,10                   'set mux A5 and start adc
1150 WAIT 786,8                   'wait until ECH
1160 GOSUB 3000                   'get sampled-data
1170 YK=(-5.750199*AA)+YK
1171 PRINT "Y3=";AA
1175 GOSUB 2000
1180 CALL DAS8(MDL%,DIO%(0),FLAG%)
1185 OUT 786,10                   'set mux A5 and start adc
1190 WAIT 786,8
1192 GOSUB 3000
1195 YK=(6.680778*AA)+YK
1196 PRINT "Y4=";AA
1250 UK= (.4133836*UK)-YK
1300 PRINT "                            UK=";UK
1500 BB=(UK*2047)/10              'from binary data word
1510 IF BB < 0 THEN BB=4095+BB    'form 2's complement word
1520 BC=INT(BB/16)                'seperate h.byte
1530 BD=BB-(BC*16)                'seperate l.byte
1540 BD=BD*16                     'shift left 4 bits
1550 X=BC:Y=BD                    'prepare to send DAC
1560 IF Y > 255 THEN Y=255
1570 OUT 784,8                     'set mux ch A4 prepare for next loop
2000 REM------------SUBROUTINE CHECK-TIMING --------------------------------
2020 CALL DAS8 (MD%,IP%,FLAG%)
2030 IF IP% = 7 THEN PRINT "SETTING TIME TOO SHORT":END
2040 CALL DAS8 (MD%,IP%,FLAG%)
2050 IF IP% <> 7 THEN GOTO 2040
2060 RETURN
3000 REM ------------SUBROUTINE SCALE-DATA ---------------------------------
3010 QA=INP(787)*16               'get data h.byte & scale
3020 QB=INP(786)/16               'get data l.byte & scale
3030 AA=QA+INT(QB)                'drop sense bits & combine H&L
3040 IF AA > 2047 THEN AA=AA-4095   'scale negative data
3050 AA=(AA/2047)*10                'scale to 10 v.ref
3060 RETURN
```

APPENDIX D

The listing of a CC program for Example 2 Design 2.

```
state
pen
  p1                          'Analog plant model
exp,p1,p2,.06                 'Discrete-time model at T
unpack,p2,p3,p4,,             'p3 = Φ1, p4 = Γ1
analog
exp,p1,p5,.12                 'Discrete-time model at 2T
unpack,p5,p6,p7,,             'p6 = Φ2, p7 = Γ2
analog
exp,p1,p8,.18                 'Discrete-time model at 3T
unpack,p8,p9,p10,,            'p9 = Φ3, p10 = Γ3
analog
exp,p1,p11,.24                'Discrete-time model at 4T
unpack,p11,p12,p13,,          'p12 = Φ4, p13 = Γ4
analog
exp,p1,p14,.30                'Discrete-time model at To
unpack,p14,p15,p16,,          'p15 = Φ, p16 = Γ
pen
  p17                         'p17 = C1
pen
  p18                         'p18 = C2
pen
  p19                         'p19 = [0]
mult,p17,p3,p20
mult,p17,p6,p21
mult,p18,p9,p22
mult,p18,p12,p23
para,p17,p20,p24,2
para,p24,p21,p25,2
para,p25,p22,p26,2
para,p26,p23,p27,2            'p27 = C̄
mult,p17,p4,p28
mult,p17,p7,p29
mult,p18,p10,p30
mult,p18,p13,p31
para,p19,p28,p32,2
para,p32,p29,p33,2
para,p33,p30,p34,2
para,p34,p31,p35,2            'p35 = Ḡ
para,p27,p35,p36,3            'p36 =  [C̄   Ḡ]
inv,p36,p37                   'p37 =        [C̄   Ḡ]^-1
poleplace,p14,p38             'p38 = F
mult,p38,p15,p39
mult,p38,p16,p40
pen
  p41                         'p41 = desired M matrix
add,p40,p41,p42
```

```
para,p39,p42,p43,3
mult,p43,p37,p44                'p44 = [FΦ  FΓ+M][C̄  Ḡ]^-1
```

To simulate the result.

```
pen
  p55                           'Discrete plant model includes
                                 noise input
pen
  p60                           'Controller model
para,p55,p60,p61,1
sel,p61,p62,input,1,3,4,5,6,7,2
sel,p62,p63,output,6,1,2,3,4,5
feedback,6,p63,p64,6
unpack,p64,p65,p66,,
pen
  p67                           'p67 = C matrix of C.L.  system
pen
  p68                           'p68 = D matrix of C.L.  system
pack,p65,p66,p67,p68,p69
dsim                            'Digital simulation
  p69
  4,1,σ,0                       'Guassian noise i/p with zero
                                 mean, standard deviation σ and
                                 random seed
  2,p100                        'Set intial condition
  30,1                          'Set simulation time
plot,p69.y,a,a
```

The listing of a control program for Example 2 Design 2.

```
10 REM ##########################################################################
20 REM    EXAMPLE II: DESIGN II  frame period=.3     updated  06/04/90
30 REM       STATE-SPACE MODEL:      . .001  0   0   0         1
40 REM                              X=   2  -1   0   0  X  +   2   U
50 REM                                  -1   0  -3   0        -1
60 REM                                   1   0   0  -2         1
65 REM
70 REM                  Y =   0   1   1   0    X
75 REM                        0   0   0   1
89 REM ##########################################################################
90 REM
100 REM ----------STEP 1:INITIALIZE DAS-8 WITH MODE 0----------------------
110 MD% = 0
120 BASADR% = &H300              'base address
130 FLAG% = 0
140 CALL DAS8 (MD%,BASADR%,FLAG%)
150 REM-----------STEP 2:SET UP COUNTER 2 AND CONFIG. 3---------------------
160 DIM DIO%(2)
170 MD% = 10                     'mode 10   set config.
180 DIO%(0) = 2                  'select counter 2
190 DIO%(1) = 3                  'set config.3  square wave generator
200 CALL DAS8 (MD%,DIO%(0),FLAG%)
210 MD% = 11                     'mode 11  load timer counter 2
220 DIO%(1) = 378                'counter number to generate 10 kHz clock
230 CALL DAS8 (MD%,DIO%(0),FLAG%)
240 REM-----------STEP 3:SET UP COUNTER 0 AND CONFIG. 0---------------------
250 MD% = 10                     'mode 10   set config.
260 DIO%(0) = 0                  'select counter 0
270 DIO%(1) = 0                  'set config 0  pulse high on terminal count
280 CALL DAS8 (MD%,DIO%(0),FLAG%)
290 REM-----------STEP 4:INITIALIZE CONTROL INPUT & GP-6 --------------------
300 KEY(1) ON:ON KEY(1) GOSUB 440     ' to exit program, use F1
310 X=0:Y=0                      'initialize controller state
315 OUT 798,Y:OUT 799,X
322 OUT 785,8                    'set c3 high --> GP-6  IC mode
323 WAIT 786,4,4                 'wait for c2 is high   IC mode
330 REM-----------STEP 5:SET PARAMETERS MODE 11 & 13------------------------
340 MDL% = 11                    'mode 11 load timer counter 0
345 MD% = 13: IP% = 0: FLAG% = 0 'mode 13 check digital i/p
360 DIO%(0) = 0:DIO%(1) = 600    'subinterval period = 60 msec
370 INPUT"PRESS [ENTER] TO START-------------->";R1
375 PRINT"FRAME PERIOD = .3 SEC"
380 PRINT"INITLAL VALUE OF CONTROL = 0 volt"
390 REM-----------STEP 6:PERFORM CONTROL ROUTINE---------------------------
400 OUT 785,0                     'set GP-6  OP mode
410 WAIT 786,4                    'wait until op mode set
415 OUT 784,8                     'set mux ch A4
420 GOSUB 1000
430 GOTO 420
440 END
```

```
1000 REM-----------SUBROUTINE MRC-----------------------------------------
1005 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1010 OUT 798,Y                    'set low byte VDAC & start ADC
1020 OUT 797,X                    'set high byte VDAC
1030 WAIT 786,8                   'wait until ECH
1040 GOSUB 3000
1080 YK=(.3065341*AA)
1085 PRINT "Y11=";AA
1090 GOSUB 2000
1095 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1097 OUT 786,8                     'set mux A4 & start adc
1098 WAIT 786,8                    'wait until ECH
1100 GOSUB 3000
1110 YK=(-.6924752*AA)+YK
1115 PRINT "Y12=";AA
1120 GOSUB 2000
1122 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1123 OUT 786,8                    'set mux A4 and start adc
1124 WAIT 786,8                    'wait until ECH
1125 GOSUB 3000
1126 YK=(.3896791*AA)+YK
1127 PRINT "Y13=";AA
1128 OUT 784,10                    'prepare mux ch A5
1129 GOSUB 2000
1130 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1140 OUT 786,10                   'set mux A5 and start adc
1150 WAIT 786,8                   'wait until ECH
1160 GOSUB 3000                   'get sampled-data
1170 YK=(-7.200939*AA)+YK
1171 PRINT "Y21=";AA
1175 GOSUB 2000
1180 CALL DAS8(MDL%,DIO%(0),FLAG%)
1185 OUT 786,10                   'set mux A5 and start adc
1190 WAIT 786,8
1192 GOSUB 3000
1195 YK=(8.119036*AA)+YK
1196 PRINT "Y22=";AA
1250 UK= (.424*UK)-YK
1300 PRINT "                  UK= ";UK
1500 BB=(UK*2047)/10              'from binary data word
1510 IF BB < 0 THEN BB=4095+BB    'form 2's complement word
1520 BC=INT(BB/16)                'seperate h.byte
1530 BD=BB-(BC*16)                'seperate l.byte
1540 BD=BD*16                     'shift left 4 bits
1550 X=BC:Y=BD                    'prepare to send DAC
1560 IF Y > 255 THEN Y=255
1570 OUT 784,8                     'prepare mux ch A4 for next loop
2000 REM-----------SUBROUTINE CHECK-TIMING-------------------------------
2020 CALL DAS8 (MD%,IP%,FLAG%)
2030 IF IP% = 7 THEN PRINT "SETTING TIME TOO SHORT":END
2040 CALL DAS8 (MD%,IP%,FLAG%)
2050 IF IP% <> 7 THEN GOTO 2040
2060 RETURN
```

```
3000 REM ----------SUBROUTINE SCALE-DATA-----------------------------------
3010 QA=INP(787)*16              'get data h.byte & scale
3020 QB=INP(786)/16              'get data l.byte & scale
3030 AA=QA+INT(QB)              'drop sense bits & combine H&L
3040 IF AA > 2047 THEN AA=AA-4095   'scale negative data
3050 AA=(AA/2047)*10               'scale to 10 v.ref
3060 RETURN
```

APPENDIX E

The listing of a CC program for Example 3 Design 2.

Disk 1

```
state
pen
  p1                          'Analog model
exp,p1,p2,.025                'Discrete-time model at T
unpack,p2,p3,p4,,             'p3 = Φ_T, p4 = Γ_T
analog
exp,p1,p5,.050                'Discrete-time model at 2T
unpack,p5,p6,p7,,             'p6 = Φ_2T, p7 = Γ_2T
analog
exp,p1,p8,.075                'Discrete-time model at 3T
unpack,p8,p9,p10,,            'p9 = Φ_3T, p10 = Γ_3T
analog
exp,p1,p11,.100               'Discrete-time model at 4T
unpack,p11,p12,p13,,          'p12 = Φ_4T, p13 = Γ_4T
analog
exp,p1,p14,.125               'Discrete-time model at 5T
unpack,p14,p15,p16,,          'p15 = Φ_5T, p16 = Γ_5T
pen
  p17                         'p17 = C_1
pen
  p18                         'p18 = C_2
pen
  p19                         'p19 = [0   0]
mult,p17,p3,p20
mult,p17,p6,p21
mult,p18,p9,p22
mult,p18,p12,p23
mult,p18,p15,p24
para,p17,p20,p25,2
para,p25,p21,p26,2
para,p26,p22,p27,2
para,p27,p23,p28,2
para,p28,p24,p29,2            'p29 = C̄
mult,p17,p4,p30
mult,p17,p7,p31
mult,p18,p10,p32
mult,p18,p13,p33
mult,p18,p16,p34
para,p19,p30,p35,2
para,p35,p31,p36,2
para,p36,p32,p37,2
para,p37,p33,p38,2
para,p38,p34,p39,2            'p39 = Ḡ
para,p29,p39,p40,3            'p40 =   [C̄   Ḡ]
inv,p40,p41                   'p41 =          [C̄   Ḡ]^-1
exp,p1,p42,.15                'Discrete-time model at 6T
unpack,p42,p43,p44,,          'p43 = Φ_6T, p44 = Γ_6T
```

```
analog
digital To=.2
pen
  p45                          'Augmented dicrete-time model
pen
  p46                          'F matrix
pen
  p47                          'G matrix
pen
  p48                          'β matrix
mult,p46,p43,p49               'FΦ6T
mult,p46,p44,p50               'FΓ6T
mult,p47,p48,p51               'Gβ
add,p50,p51,p52                'FΓ6T+Gβ
pen
  p53                          'Desired M matrix
add,p52,p53,p54
para,p49,p54,p55,3
mult,p55,p41,p56               'H=[FΦ6T   FΓ6T+Gβ+M][C̄   Ḡ]-1
```

To simulate the result.

```
pen
  p70                          'p70=α(T)b2=the second column
                                of ΓT matrix
mult,p3,p70,p71                'p71 = ΦTα(T)b2
para,p6,p71,p72,3              'p72 = [Φ2T   ΦTα(T)b2]

mult,p29,p72,p73               'p73 = C̄[Φ2T   ΦTα(T)b2]
pen
  p74                          'p74=α(2T)b1=the first column
                                of Γ2T
para,p74,p70,p75,3             'p75 = [α(2T)b1   α(T)b2]

mult,p29,p75,p76               'C̄[α(2T)b1   α(T)b2]

add,p76,p39,p77                'C̄[α(2T)b1   α(T)b2] + Ḡ
unpack,p45,p78,p79,,
pack,p78,p79,p73,p77,p80       'Augmented discrete-time with
                                multirate o/p sampling model
pchange
  p80                          'Augmented model with noise i/p
pen
  p84                          'Controller model
para,p80,p84,p85,1
sel,p85,p86,input,1,2,4,5,6,7,8,9,3
sel,p86,p87,output,7,8,1,2,3,4,5,6
feedback,6,p87,p88,8
unpack,p88,p88,,,
sel,p85,p89,input,1,2
unpack,p89,,p89,,
pen
```

```
  p90                          'p90 = C matrix of C.L.  system
pen
  p91                          'p91 = D matrix of C.L.  system
pack,p88,p89,p90,p91,p92
dsim                           'Digital simulation
  p92
  4,1,σ,0
  2,p100
  30,1
plot,p92.y,a,a
```

Disk 2 **To build the augmented discrete-time plant.**

```
pen
  p1                           'Analog plant model
exp,p1,p2,.2                   'Discrete-time model at T_o=8T
unpack,p2,p3,,,                'p3 = exp(A8T)
pen
  p4                           'p4=α(8T)b_1 =the first column
                                of Γ matrix
pen
  p5                           'p5=α(T)b_2 = the second column
                                of Γ_T matrix
analog
exp,p1,p6,.175                 'Discrete-time model at 7T
unpack,p6,p7,,,
pen                            'p7 = exp(7T)
  p8                           'p8=α(7T)b_2 = the second column
                                of Γ_7T matrix
pen
  p9                           'p9 = [0   0   0   0]
pen
  p10                          'p10 = [0]
pen
  p11                          'p11 = [0   1]
mult,p7,p5,p12
para,p3,p9,p13,2
para,p12,p10,p14,2
para,p13,p14,p15,3             'p15 = Φ_Aug
para,p4,p8,p16,3
para,p16,p11,p17,2             'p17 = Γ_Aug
pack,p15,p17,,,p18             'p18 = augmented discrete-time
                                      state model
```

Note that "plpd" and "pd" commands for printing all results are not included in programs.

The listing of a control program for Example 3 Design 2.

```
10 REM ##########################################################################
20 REM    EXAMPLE III: DESIGN II    frame period=.2      updated 06/04/90
30 REM       STATE-SPACE MODEL:     .    0   1   0   0         0   0
40 REM                              X=   3   0   0   2  X  +   1   0    U
50 REM                                   0   0   0   1         0   0
60 REM                                   0  -2   0   0         0   1
65 REM
70 REM                  Y =   1   0   0   0    X
75 REM                        0   0   1   0
89 REM ##########################################################################
90 REM
100 REM -----------STEP 1:INITIALIZE DAS-8 WITH MODE 0----------------------
110 MD% = 0
120 BASADR% = &H300              'base address
130 FLAG% = 0
140 CALL DAS8 (MD%,BASADR%,FLAG%)
150 REM-----------STEP 2:SET UP COUNTER 2 AND CONFIG. 3---------------------
160 DIM DIO%(2)
170 MD% = 10                     'mode 10   set config.
180 DIO%(0) = 2                  'select counter 2
190 DIO%(1) = 3                  'set config.3  square wave generator
200 CALL DAS8 (MD%,DIO%(0),FLAG%)
210 MD% = 11                     'mode 11  load timer counter 2
220 DIO%(1) = 378                'count number to generate 10 kHz clock
230 CALL DAS8 (MD%,DIO%(0),FLAG%)
240 REM-----------STEP 3:SET UP COUNTER 0 AND CONFIG. 0---------------------
250 MD% = 10                     'mode 10   set config.
260 DIO%(0) = 0                  'select counter 0
270 DIO%(1) = 0                  'set config 0  pulse high on terminal count
280 CALL DAS8 (MD%,DIO%(0),FLAG%)
290 REM-----------STEP 4:INITIALIZE CONTROL INPUT & GP-6--------------------
300 KEY(1) ON:ON KEY(1) GOSUB 440     ' to exit program, use F1
310 X=0:Y=0:U1=0:U2=0:YU2=0
315 OUT 788,Y:OUT 789,X              'initialize U1
320 OUT 792,Y:OUT 793,X              'initialize U2
322 OUT 785,8                        'set c3 high --> GP-6  IC mode
323 WAIT 786,4,4                     'wait for c2 is high   IC mode
330 REM-----------STEP 5:SET PARAMETERS MODE 11 & 13------------------------
340 MDL% = 11        'mode 11 load timer counter 0
345 MD% = 13: IP% = 0: FLAG% = 0     'mode 13 check digital i/p
360 DIO%(0) = 0:DIO%(1) = 250        'subinterval period = 25 msec
370 INPUT"PRESS [ENTER] KEY TO START------->";R
380 PRINT"INITLAL VALUE OF CONTROL = 0 volt"
390 REM-----------STEP 6:PERFORM CONTROL ROUTINE---------------------------
400 OUT 785,0                        'set GP-6  OP mode
410 WAIT 786,4                       'wait until op mode set
420 GOSUB 1000
430 GOTO 420
440 END
```

```
1000 REM-----------------SUBROUTINE MRC-------------------------------------
1005 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1010 OUT 788,Y                        'set low byte LDAC
1020 OUT 789,X                        'set high byte LDAC
1022 PRINT "                                  U1 = ";U1
1025 U2 = (.2364881*U1)+(-.1303905*U2)-YK2     'prepare control i/p U2
1026 UK=U2
1027 GOSUB 1500
1040 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1050 OUT 792,Y                        'set low byte RDAC
1060 OUT 793,X                        'set high byte RDAC
1065 PRINT "                                  U2 = ";U2
1070 OUT 784,2                        'prepare for mux channel Amp#2
1080 GOSUB 2000
1090 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1100 OUT 786,2                        'set mux Amp#2 & start ADC
1110 WAIT 786,8                       'wait ECH
1120 GOSUB 3000
1130 YK1 = (-39.75176*AA)
1135 YK2 = (39.64745*AA)
1137 PRINT "             Y1 = ";AA
1140 GOSUB 2000
1150 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1160 OUT 786,2                        'set mux Amp#2 & start ADC
1170 WAIT 786,8                       'wait ECH
1180 GOSUB 3000
1190 YK1 = (.6643795*AA)+YK1
1195 YK2 = (-.801614*AA)+YK2
1197 PRINT "             Y2 = ";AA
1200 GOSUB 2000
1210 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1220 OUT 786,2                         'set mux Amp#2 & start ADC
1230 WAIT 786,8                        'wait ECH
1240 GOSUB 3000
1250 YK1 = (43.3663*AA)+YK1
1255 YK2 = (-39.43504*AA)+YK2
1260 OUT 784,6                         'prepare for channel Amp#4
1265 PRINT "             Y3 = ";AA
1270 GOSUB 2000
1280 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1290 OUT 786,6                         'set mux Amp#4 & start ADC
1300 WAIT 786,8                        'wait ECH
1310 GOSUB 3000
1320 YK1 = (-39.55061*AA)+YK1
1325 YK2 = (-40.48182*AA)+YK2
1327 PRINT "             Y4 = ";AA
1330 GOSUB 2000
1340 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1350 OUT 786,6                         'set mux Amp#4 & start ADC
1360 WAIT 786,8                        'wait ECH
1370 GOSUB 3000
1380 YK1 = (.5886037*AA)+YK1
1385 YK2 = (.5153218*AA)+YK2
```

```
1387 PRINT "            Y5 = ";AA
1390 GOSUB 2000
1400 CALL DAS8 (MDL%,DIO%(0),FLAG%)
1410 OUT 786,6                          'set mux Amp#4 & start ADC
1420 WAIT 786,8                         'wait ECH
1430 GOSUB 3000
1435 PRINT "            Y6 = ";AA
1440 YK1 = (38.31294#AA)+YK1
1445 YK2 = (43.21182#AA)+YK2
1450 U1=(-.229871#U1)+(-6.853208E-02#U2)-YK1   'prepare control i/p U1
1460 UK=U1
1500 REM-------------Prepare data to send-----------------------------------
1505 BB=(UK#2047)/10              'from binary data word
1510 IF BB < 0 THEN BB=4095+BB    'form 2's complement word
1520 BC=INT(BB/16)                'seperate h.byte
1530 BD=BB-(BC#16)                'seperate l.byte
1540 BD=BD#16                     'shift left 4 bits
1550 X=BC:Y=BD                    'prepare to send DAC
1560 IF Y > 255 THEN Y=255
2000 REM-----------------SUBROUTINE CHECK-TIMING----------------------------
2020 CALL DAS8 (MD%,IP%,FLAG%)
2030 IF IP% = 7 THEN PRINT "SETTING TIME TOO SHORT":END
2040 CALL DAS8 (MD%,IP%,FLAG%)
2050 IF IP% <> 7 THEN GOTO 2040
2060 RETURN
3000 REM ----------------SUBROUTINE SCALE-DATA-----------------------------
3010 QA=INP(787)#16               'get data h.byte & scale
3020 QB=INP(786)/16               'get data l.byte & scale
3030 AA=QA+INT(QB)                'drop sense bits & combine H&L
3040 IF AA > 2047 THEN AA=AA-4095   'scale negative data
3050 AA=(AA/2047)#10                'scale to 10 v.ref
3060 RETURN
```

APPENDIX F

Find state variable feedback gain matrix using Weighted Least-Squares Approximation [25].

Consider the discrete-time state model of the plant:

$$x(kT_o+T_o) = \Phi x(kT_o) + \Gamma u(kT_o) \tag{1}$$

where

$$\Phi = \begin{bmatrix} \Phi_{11} & \Phi_{12} & \Phi_{13} & \Phi_{14} \\ \Phi_{21} & \Phi_{22} & \Phi_{23} & \Phi_{24} \\ \Phi_{31} & \Phi_{32} & \Phi_{33} & \Phi_{34} \\ \Phi_{41} & \Phi_{42} & \Phi_{43} & \Phi_{44} \end{bmatrix}, \text{ and } \Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \\ \Gamma_{31} & \Gamma_{32} \\ \Gamma_{41} & \Gamma_{42} \end{bmatrix}.$$

The control law is given by

$$u(kT_o) = - Fx(kT_o) \tag{2}$$

where

$$F = \begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{14} \\ f_{21} & f_{22} & f_{23} & f_{24} \end{bmatrix}.$$

Since the plant is not in canonical form and has two inputs, there are eight entries in the gain matrix to be selected and the specification of four closed loop poles will clearly leave many possible values of F which will meet the specification. This can make it difficult to apply the pole assignment method to the system.

By decoupling, which assumes that the model in equation (1)

is of the form

$$x(kT_o+T_o) = \bar{\Phi}x(kT_o) + \bar{\Gamma}v(kT_o) \tag{3}$$

where

$$\bar{\Phi} = \begin{bmatrix} \Phi_{11} & \Phi_{12} & 0 & 0 \\ \Phi_{21} & \Phi_{22} & 0 & 0 \\ 0 & 0 & \Phi_{33} & \Phi_{34} \\ 0 & 0 & \Phi_{43} & \Phi_{44} \end{bmatrix}, \quad \bar{\Gamma} = \begin{bmatrix} \Gamma_{11} & 0 \\ \Gamma_{21} & 0 \\ 0 & \Gamma_{32} \\ 0 & \Gamma_{42} \end{bmatrix},$$

the control law will become

$$v(kT_o) = -\bar{F}x(kT_o)$$

$$= -\begin{bmatrix} f_{11} & f_{12} & 0 & 0 \\ 0 & 0 & f_{23} & f_{24} \end{bmatrix} x(kT_o). \tag{4}$$

The decoupled form in equation (3) permits an easy gain calculation by partitioning it into two subsystems. The single-input pole assignment method can be applied separately to each subsystem. Then, results of the two feedback gain matrices are combined to obtain the F matrix.

Actually, a decoupled model in equation (3) is an approximate model of an original model in equation (1). It is desirable to make the model in equation (3) as close to the original model as possible. If they match closely enough, the relationship between two models can be written as

$$\Phi x(kT_o) + \Gamma u(kT_o) = \bar{\Phi}x(kT_o) + \bar{\Gamma}v(kT_o). \tag{5}$$

Rewriting equation (5) gives

$$-\Gamma u(kT_o) = (\Phi - \bar{\Phi})x(kT_o) - \bar{\Gamma}v(kT_o). \qquad (6)$$

Observe that equation (6) is a linear equation and the statement represented by equation (5) is true if and only if one can find a solution $u(kT_o)$ of equation (6). Premultiplying equation (6) by $-\Gamma^{-1}$ gives

$$u(kT_o) = -\Gamma^{-1}(\Phi - \bar{\Phi})x(kT_o) + \Gamma^{-1}\bar{\Gamma}v(kT_o). \qquad (7)$$

Since Γ is 4 x 2 matrix which is not a square matrix , one cannot find Γ^{-1}. Therefore a solution $u(kT_o)$ in equation (7) for equation (6) is not correct. The approximate solution of equation (6) can be obtained by using Weighted Least-Squares Approximation which gives a solution of equation (6) as follows:

$$u(kT_o) = -(\Gamma' Q^{-1}\Gamma)^{-1}\Gamma' Q^{-1}(\Phi-\bar{\Phi})x(kT_o) + (\Gamma' Q^{-1}\Gamma)\Gamma' Q^{-1}\bar{\Gamma}v(kT_o)$$

$$= - M\, x(kT_o) + Nv(kT_o) \qquad (8)$$

where Q is a symmetric, nonsingular and diagonal matrix. In this thesis Q is selected to be I_4.
Substituting $v(kT_o) = -\bar{F}x(kT_o)$ into equation (8) gives

$$u(kT_o) = - Mx(kT_o) + N(-\bar{F})x(kT_o)$$

$$= - (N\bar{F} + M)x(kT_o). \qquad (9)$$

Comparing equation (2) and equation (9), the state variable feedback gain matrix of an original model in equation (1)

can be obtained by

$$F = N\bar{F} + M. \tag{10}$$

The relation of an original model and an approximate model using Weighted Least-Squares Approximation method is shown in figure F.1. More details can be seen in reference [6].

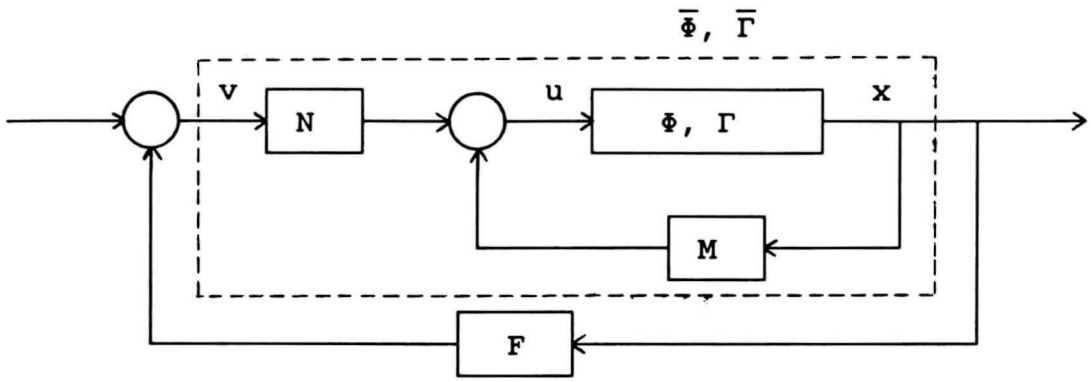


Fig. F.1 Block diagram of the closed-loop system using Weighted Least-Squares Approximation.

The listing of a CC program for the Weighted Least-squares Approximation method.

```
pen
  p1                          'Analog plant model
exp,p1,p2,T_o                 'Discrete-time model at T_o
unpack,p2,p3,p4,,             'p3=Φ and p4=Γ
scale,p2,p5,1
pchange
  p5
unpack,p6,p7,,                'p6=Φ̄ and p7=Γ̄
pen
  p10                         'p10 = Q , in this case Q=I_4
transpose, p4,p11
mult,p11,p10,p12
mult,p12,p4,p13
inv,p13,p14
mult,p14,p11,p15
mult,p15,p10,p16
mult,p16,p7,p17               'p17 = N = (Γ'QΓ)^-1 Γ'Q̄Γ
subtr,p3,p6,p18
mult,p16,p18,p19              'p19 = M = (Γ'QΓ)^-1 Γ'Q(Φ-Φ̄)
sel,p5,p20,state,1,2          'Obtain subsystem 1
sel,p20,p20,inp,1
poleplace                     'Use desired discrete poles
  p20,p21                     'p21 = feedback gain matrix
sel,p5,p22,state,3,4          'Obtain subsystem 2
sel,p22,p22,inp,2
poleplace                     'Use desired disdrete poles
  p22,p23                     'p23 = feedback gain matrix

para,p21,p23,p24,1            'p24 = F̄

mult,p17,p24,p25              'p25 = NF̄

add,p25,p19,p26               'p26 = F
```

APPENDIX G

Circuit diagrams of Example 1, 2, and 3.

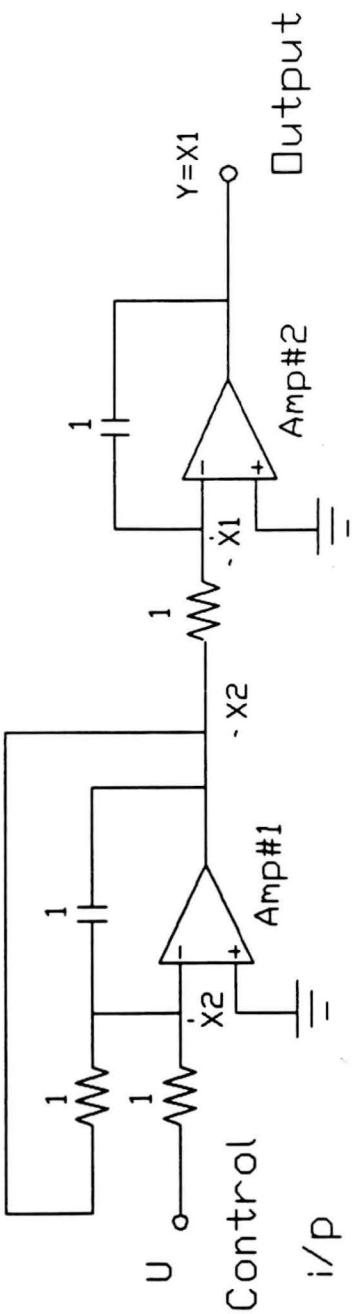


Figure G.1 Analog plant circuit built on GP-6 for Example 1.

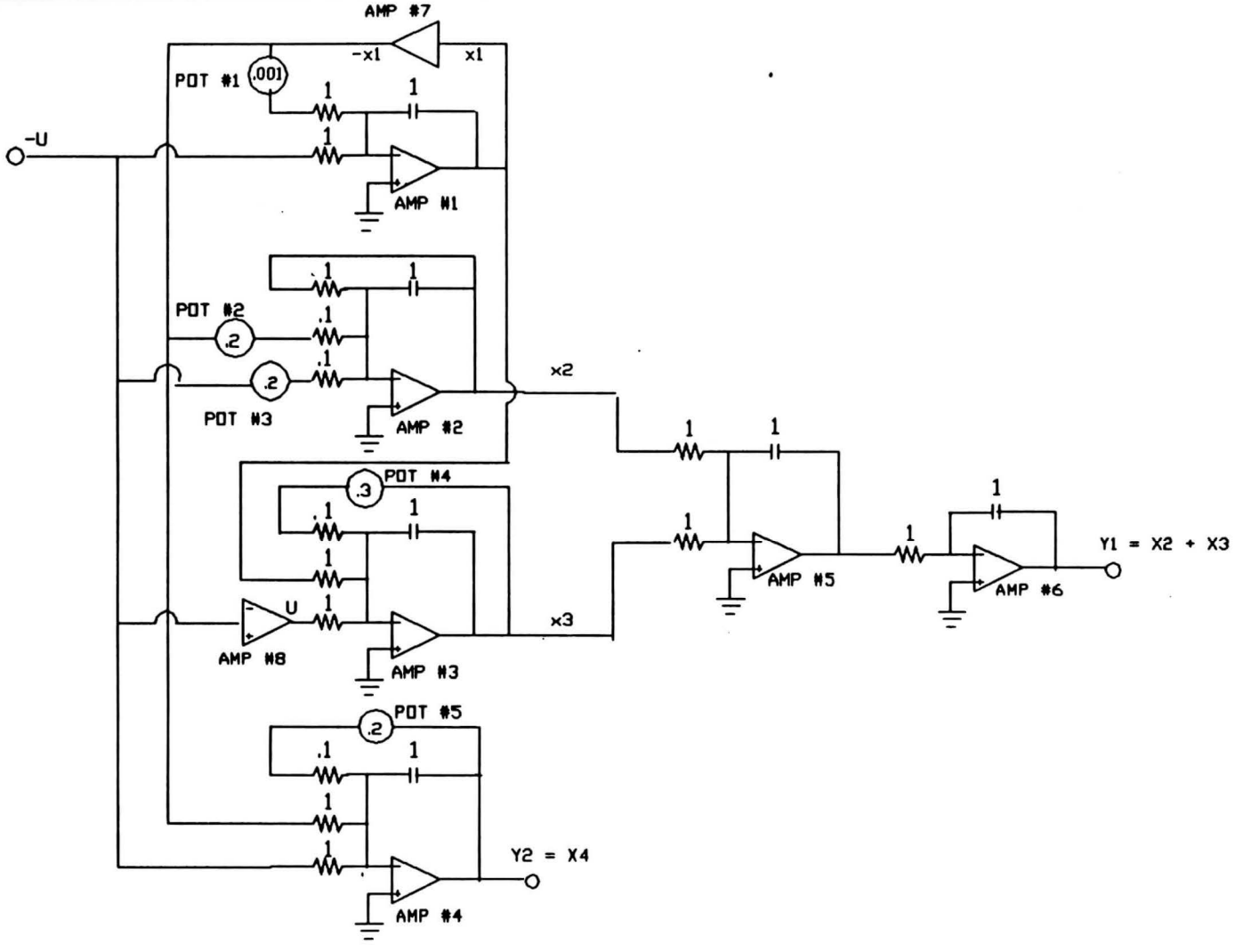


Figure G.2 Analog plant circuit built on GP-6 for Example 2.

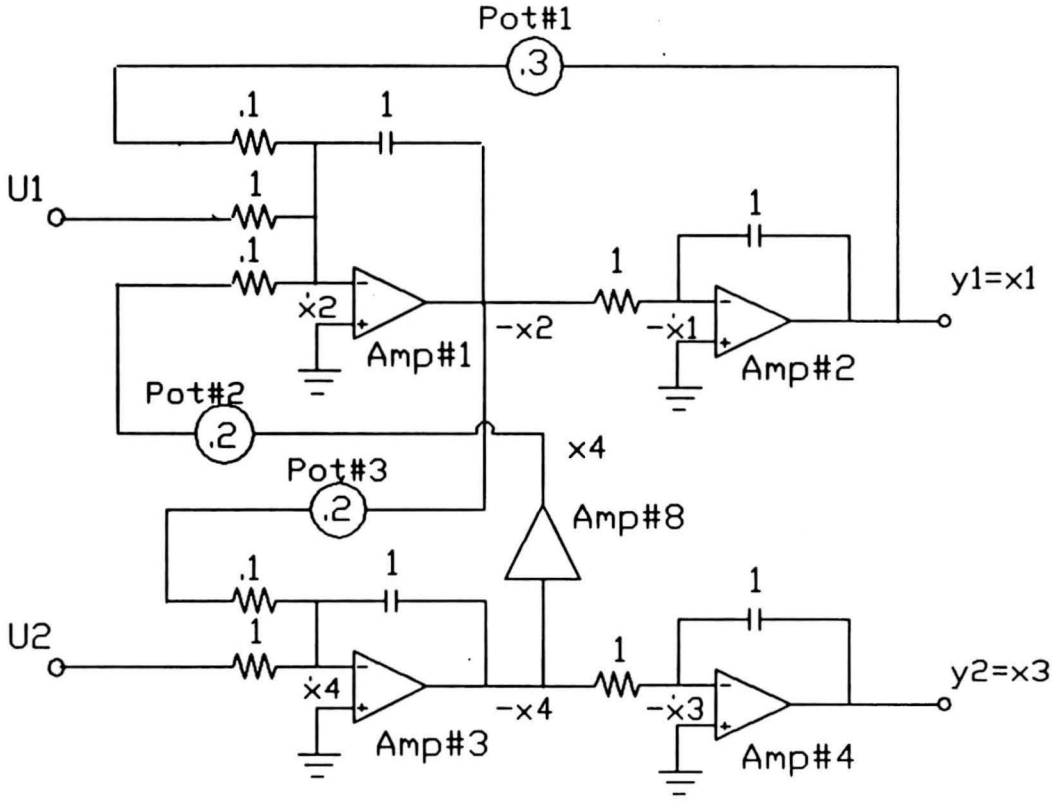


Figure G.3 Analog plant circuit built on GP-6 for Example 3.

APPENDIX H

Brief explanations of two options in program CC that are used for simulating the results.

Simulating the results with reference input for Example 1.

Consider the discrete-time equivalent model of the plant with a multirate output sampling mechanism at frame period T_o given by

$$\left.\begin{aligned} x(kT_o+T_o) &= \Phi x(kT_o) + \Gamma u(kT_o) \\ Y(kT_o) &= \bar{C}x(kT_o) + \bar{G}u(kT_o). \end{aligned}\right\} \quad (1)$$

The multirate sampled-data controller model, the outputs of which are the control signals of the plant, is given by

$$\left.\begin{aligned} u(kT_o+T_o) &= Mu(kT_o) - H\bar{Y}(kT_o) + N_r r(kT_o) \\ Y(kT_o) &= u(kT_o). \end{aligned}\right\} \quad (2)$$

The procedure is explained in the following steps:

1. Build the discrete-plant model from equation (1).
2. Build the controller model from equation (2).
3. Parallel two models together using option 1. The result is shown in Figure H.1.

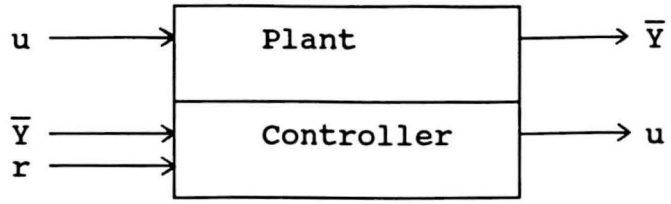


Figure H.1 Block diagram of the two models after parallel by using CC commands.

4. Select outputs for full state feedback by order for matching inputs order.
5. Do a full state feedback option 6 with number of outputs. The result is given in Figure H.2.

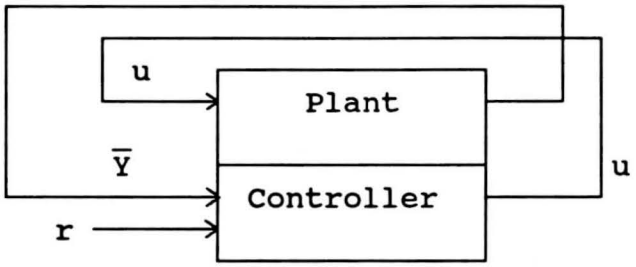


Figure H.2 Block diagram of the closed-loop system by using CC commands.

6. Add the output matrix which is composed of the original outputs and the control input of the plant to the closed-loop system. The result is shown in Figure H.3.

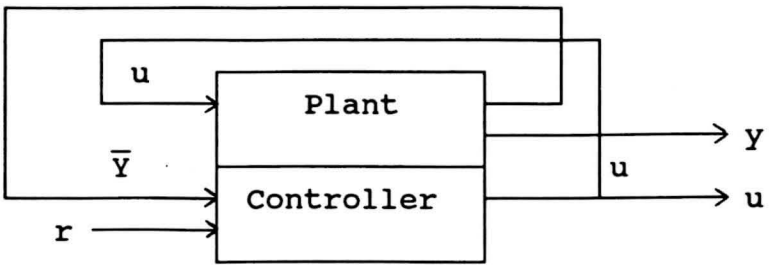


Figure H.3 Block diagram of the closed-loop system with monitored outputs by using CC commands.

7. Do a digital simulation option 1 with a +5 volts reference input.
8. Plot the results.

Simulating the results with measurment noise for Examples 2 and 3.

In Example 2, to simulate responses of the system in case disturbances occur in measurments, an additional vector is added to the model to represent noise input. The plant model in equation (1) becomes

$$\left.\begin{aligned} x(kT_o+T_o) &= \Phi x(kT_o) + \Gamma u(kT_o) + W_1 v(kT_o) \\ \bar{Y}(kT_o) &= \bar{C}x(kT_o) + \bar{G}u(kT_o) + W_2 v(kT_o) \end{aligned}\right\} \quad (3)$$

where $v(kT_o)$ is noise input vector,

W_1 is n x 1 matrix which has all 0's,

and W_2 is (N_1+N_2) x 1 matrix which has all 1's.

The multirate sampled-data controller model with no reference input is written as

$$\left.\begin{aligned} u(kT_o+T_o) &= Mu(kT_o) - H\bar{Y}(kT_o) \\ Y(kT_o) &= u(kT_o). \end{aligned}\right\} \quad (4)$$

The procedure is explained in the following steps:

1. Build the plant model from equation (3).
2. Build the controller model from equation (4).
3. Parallel models from step 1 and 2. The result is shown in Figure H.4.

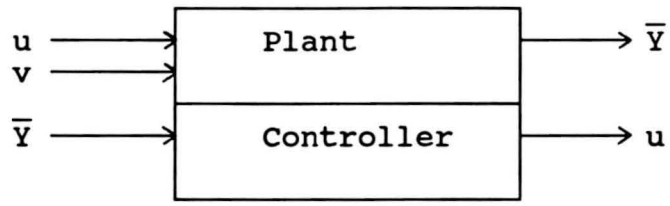


Figure H.4 Block diagram of the two models after parallel with noise input by using CC commands.

4. Select inputs to be feedback.
5. Select outputs to be feedback by matching to the inputs.
6. Do a full state feedback option 6.
7. Add output matrix which is composed of the original outputs and the control input of the plant to the closed-loop system. The result is shown in Figure H.5.

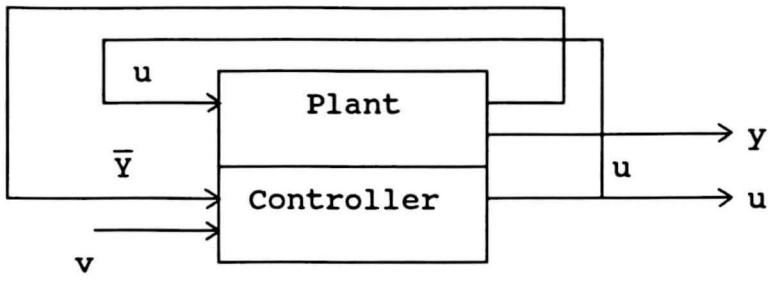


Figure G.5 Block diagram of the closed-loop system with noise input by using CC commands.

8. Do a digital simulation option 4 to generate random Guassian noise input with zero mean and standard deviation = a x (1 quantization level), where a is an integer number. Since
the maximum voltage allowed for conversion = + 10 volts,
the minimum voltage allowed for conversion = - 10 volts,
and resolution = 12 bits,
therefore

$$1 \text{ quantization level} = \frac{20}{2^{12}} = .004882813 \text{ volts.}$$

Note that random noise input with zero mean and standard deviation = 3 quantization level (.014648438 volts) is generated for the Computer Simulation results to compare with the Real-time results.

9. Plot the results by using an initial condition

$$x(0) = [\ 1 \quad 1 \quad 0 \quad -1]^T.$$

In Example 3, consider the augmented discrete-time model of the plant with noise input at frame period T_o given by equation (2.4.6),

$$\begin{bmatrix} x(kT_o+T_o) \\ v(kT_o+T_o) \end{bmatrix} = \begin{bmatrix} \Phi & \Phi_{7T}\alpha(T)b_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(kT_o) \\ v(kT_o) \end{bmatrix}$$

$$+ \begin{bmatrix} \alpha(8T) & \alpha(7T)b_2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(kT_o) \\ u_2(kT_o) \end{bmatrix} + W_1 v(kT_o) \qquad (5)$$

and a multirate output sampling mechanism given by equation (2.4.7),

$$\bar{Y}(kT_o) = \bar{C}x(kT_o+2T) + \bar{G}u(kT_o) + W_2 v(kT_o). \qquad (6)$$

Writing $x(kT_o+2T)$ in equation (6) in term of $x(kT_o)$ and $v(kT_o)$ gives

$$x(kT_o+2T) = \Phi_{2T}x(kT_o) + \alpha(2T)b_1u_1(kT_o) + \alpha(T)b_2u_2(kT_o)$$

$$+ \Phi_T\alpha(T)b_2v(kT_o). \qquad (7)$$

Substituting equation (7) into equation (6), the multirate sampled-data outputs is written as

$$Y(kT_o) = \bar{C}[\Phi_{2T} \quad \Phi_T\alpha(T)b_2] \begin{bmatrix} x(kT_o) \\ v(kT_o) \end{bmatrix}$$

$$+ [\bar{C}[\alpha(2T)b_1 \quad \alpha(T)b_2] + \bar{G}] \begin{bmatrix} u_1(kT_o) \\ u_2(kT_o) \end{bmatrix} + W_2 v(kT_o). \qquad (8)$$

Equation (5) and (8) are used to represent the augmented

discrete-time plant model with noise input in step 1. The procedures in step 2 through step 9 are similar to Example 2 except the random noise input with zero mean and standard deviation = 5 quantizations (=.024414063 volts) is generated for comparing the results, and initial condition is set to be

$$x(0) = [1 \quad 0 \quad -1 \quad 0]^T.$$

More details can be seen in The Program CC Reference Manual [18].

REFERENCES

[1] T. Hagiwara and M. Araki. Design of a Stable State Feedback Controller Based on the Multirate Sampling of the Plant Output. IEEE Transactions on Automatic Control. Vol. 33, No. 9, pp. 812-819, September 1988.

[2] D. G. Luenberger. Canonical Forms for Linear Multivariable System. IEEE Transactions on Automatic Control. Vol. AC-12, pp.290-293, June 1967.

[3] D. G. Luenberger. Observers for Multivariable Systems. IEEE Transactions on Automatic Control. Vol. AC-11, pp. 190-197, April 1966.

[4] Gene F. Franklin and J. David Powell. Digital Control of Dynamic Systems. New York: Addison-Wesley Publishing Company., 1980.

[5] Gene H. Hosttetter, Clement J. Savant, Jr. and Raymond T. Stefani. Design of Feedback Control Systems. New York: Holt, Rinehart and Winston, Inc., 1988.

[6] Wiliam L. Brogan. Modern Control Theory. New Jersey: Prentice-Hall, 2nd edition., 1985.

[7] Benjamin C. Kuo. Digital Control Systems. New York: Holt, Rinehart and Winston, Inc., 1980.

[8] Friedland, Bernard. Control System Design. New York: McGraw-Hill, 1986.

[9] Chi-Tsong Chen. Linear System Theory and Design. New York: Holt, Rinehart and Winston, Inc., 1984.

[10] Karl J. Astrom and Bjorn Wittenmark. Computer Controlled Systems. New Jersey: Prentice-Hall, 1984.

[11] Ruel V. Churchill. Complex Variables and Applications. New York: McGraw-Hill, 1960.

[12] Gene H. Hostetter. Digital Control System Design. New York: Holt, Rinehart and Winston, Inc., 1988.

[13] Ben Noble and James W. Daniel. Applied Linear Algebra. New Jersey: Prentice-Hall, 3rd edition, 1988.

[14] The GP-6 Analog Computer Operators and Maintainance Manual. Illinois: Comdyna Inc.

[15] The STEP 286/16 Reference and User Manual. Fremont, CA: EVEREX System Inc., 1988.

[16] Model 767 Analog/Digital Positional Control Panel Operators and Maintainance Manual. Illinois: Comdyna, Inc.

[17] DAS-8 and DAS-8PGA User's Manual. Taunton MA: MetraByte Corperation., 1988.

[18] Peter M. Thompson. Program CC User's Guide Version 3. Hawthorne, CA: System Technology, 1985.

[19] Microsoft MS-DOS User's Guide. Version 3.3. Fremont, CA: EVEREX Systems Inc., 1988.

[20] Microsoft GW-BASIC User's Guide. Fremont, CA: EVEREX Systems Inc., 1988.

[21] Analog Devices. The Data-Acquisition Databook. Norwood, MA: Analog Devices, volume 1., 1984.

[22] Bassil Ibrahim. Implementation of a Digital Control Algorithm with Multiplexed State Variable Feedback. Masters Thesis Youngstown State University., 1988.

[23] Nai-Chian Guo. Comparision of Parameter Estimation Algorithms Implemented on a Personel Computer. Masters Thesis Youngstown State University., 1988.

[24] Jyan-Bang Chen. A Two-parameter Compensator for Set-point Control and Noise Reduction. Masters Thesis Youngstown State University., 1989.

[25] Montakef Ali. A Design Technique for Multivariable Servo-compensators with an Application to Flight Control. Masters Thesis Youngstown State University., 1987.