EE208:CONTROL ENGINEERING LAB 12

Transfer between digital states.

Course Instructor : Dr. Sanjay Roy

Date : 14th April 2022

Group Number - 14

Raj Kumar Deb	2020eeb1025
Tushar Sharma	2020eeb1325
Tiya Jain	2020eeb1213

TABLE OF CONTENTS

1. OBJECTIVE

In this experiment we consider a given three-variable digital system with parameters susceptible to arbitrariness of values (or for that matter, adjustable by the operator), and check out the scope and limits of possible deadbeat type performance.

2. GIVEN INFORMATION

The following three variable system has arbitrary values or settings possible for parameters a and b:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a & b & \frac{-a}{b} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} .u(k)$$

Conditions needed to be satisfied:

1. With initial state $x(0) = [1\ 1\ 1]^T$, find out if $x(3) = 0$ is achievable.

2. With initial state $x(0) = 0$, find out if $x(3) = [1\ 1\ 1]^T$ is achievable.

3. APPROACH

In this experiment , we start by solving the equations for the given conditions.

- We begin our approach by trying to satisfy the second condition, i.e. we took $x_1(0) = x_2(0) = x_3(0) = 0$. And using the matrix equations to consecutively derive the expressions for next states, we obtained $x_1(3) = u(1)$; $x_2(3) = b.u(0) + u(2)$; $x_3(3) = b.u(1)$.
- Next step was to equate all the values of these states to 1, and conclude whether this would be possible or not.

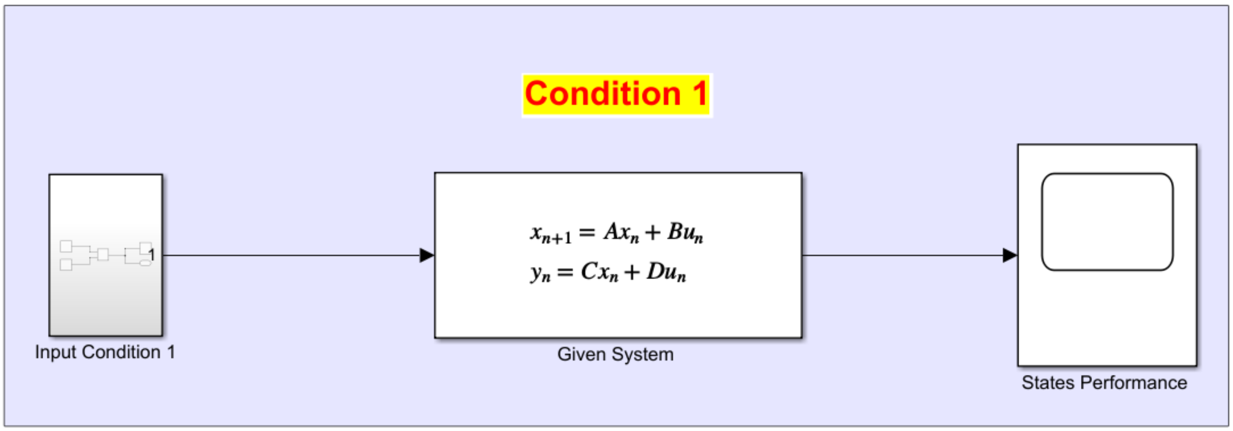
  So, we get $u(1) = 1$; $b.u(0) + u(2) = 1$; $b.u(1) = 1$.

- Similarly, we then try to satisfy the first condition by taking **b = 1 itself** (in order to check if the given system itself can be used to satisfy both the conditions simultaneously).

- Again taking $x_1(0) = x_2(0) = x_3(0) = 1$, and continuing in the same way, we obtain:

  $x_1(3) = 1 + u(1);$ $x_2(3) = 1 + u(0) + u(2);$ $x_3(3) = 1 + u(1),$

These observations are obtained through manual calculations. The same has been done further using the Simulink & MATLAB model in order to verify the conditions.



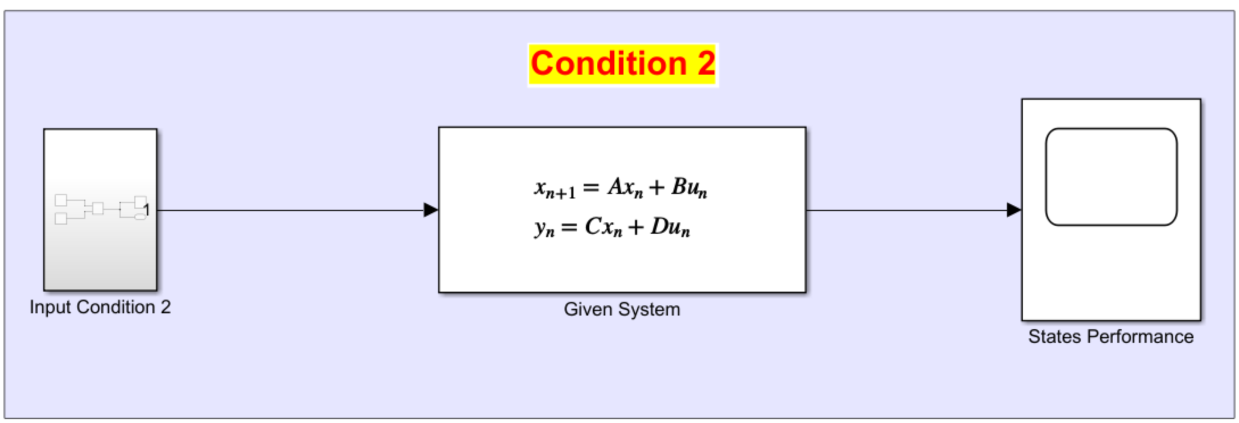


Figure 3.1 Simulink Block Diagram for both conditions

We were given two conditions and we did preliminary inspection and some calculations to find out if the condition is achievable or not and the values of a, b and the input *u(k)* required.

We took the C matrix to be an identity matrix and D to be a null matrix and get the states x_n as our output to monitor the response in scope.

4. OBSERVATIONS

4.1 Condition No. 1

For values of a, b parameters and input u(k), assuming the condition x(3) = 0 as true we solve these step by step below:

$$x_1(3) = a + b - \frac{a}{b} + u(1) = 0$$

$$x_2(3) = a + b\,[(1 + u(0)] - \frac{a}{b}\,(a + b - \frac{a}{b}) + u(2) = 0$$

$$x_3(3) = ab + b^2 - a + bu(1) + \frac{a^3}{b^2} - \frac{a^3}{b^3} = 0$$

4.1.1 MATLAB Code

```
clc;
close all;

% Constant Values
a = 0.5;
b = 1;

U = zeros(10);
X = zeros(3,10);

% Defining A,B matrix
A = [
    [0 1 0];
    [0 0 1];
    [a b -a/b];
    ];
B = [0; 1; 0;];
C = [[1 0 0];[0 1 0];[0 0 1];];
D = [0;0;0;];
```

```
% Case #1
U(2)=-1;U(3)=-1;
X(:,1) = [1;1;1;];

% Iteration
for i=2:10
    X(:,i) = A*X(:,i-1) + B*U(i-1);
end

X
```

4.1.2 Scope and limits of possible deadbeat type performance on MATLAB

4.1.2.1 Input given to the system

As $a \in Real$, b = 1, u(0) + u(2) = -1 and u(1) = -1, assuming u(0) = p, we get u(2) = -1 - p. The range of our input will be of the format $u(k) = [p, \; -1, \; -1-p, \; 0, \; 0, \; 0, \; 0, \; ...]$ where p is an arbitrary real constant value. For simplicity, here we took p = 0.

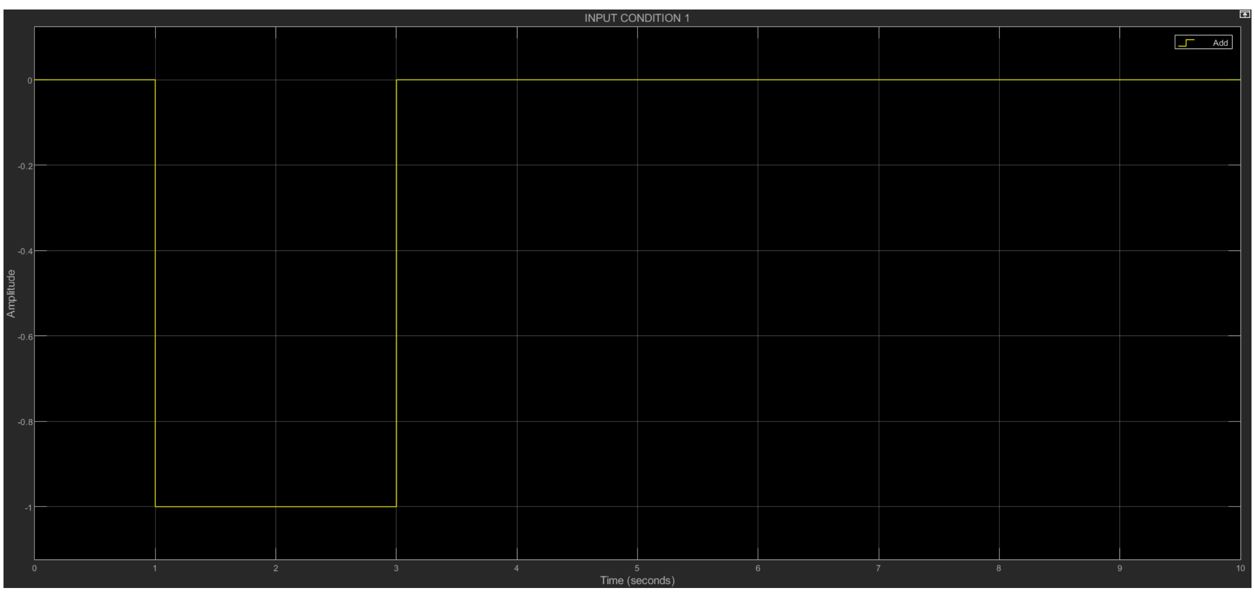


Figure 4.1.2.1.1 Input for system for condition1

4.1.2.2 Simulink output representation

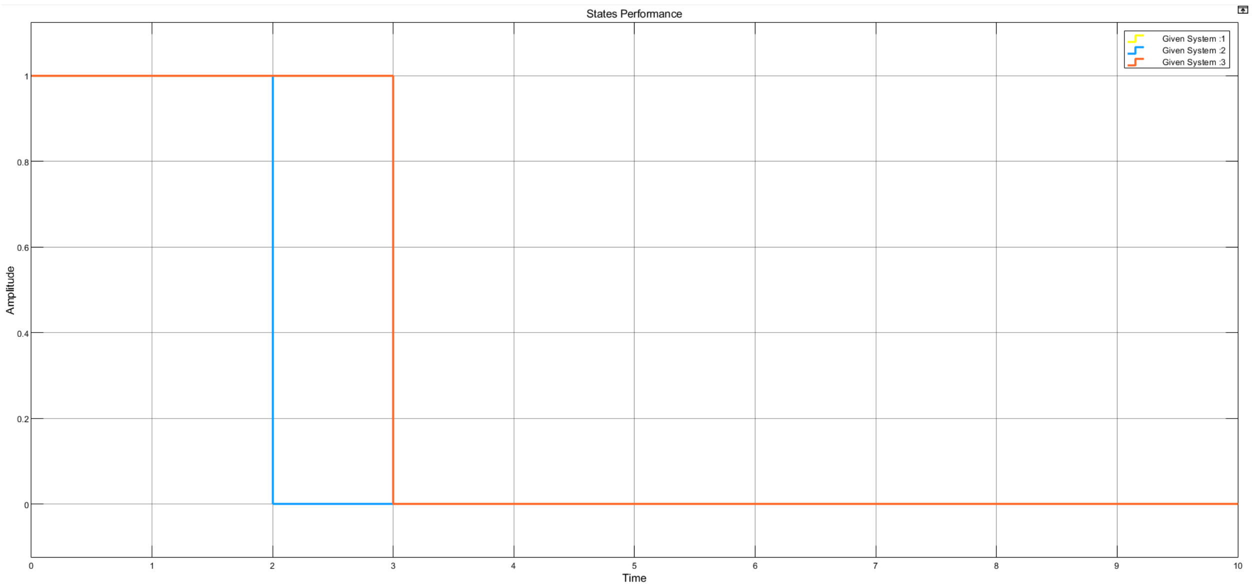


Figure 4.1.2.2.1 Simulink output representation for condition 1

4.1.2.3 MATLAB Output Representation

```
STATE_VALUES = 3×10

     1     1     1     0     0     0     0     0     0     0
     1     1     0     0     0     0     0     0     0     0
     1     1     1     0     0     0     0     0     0     0
```

4.1.3 Observation & Explanation

- In order to change the value of states from higher to lower values,we required a negative input (initial state of [1 1 1] to [0 0 0]). Once the desired value of x_3 is achieved, we stop the supply of any input i.e. u(k) is equal to zero (for k greater than or equal to 3). Hence, we observed that all the states settle down at its value at x_3.
- As the states settle at their final values instantly and there is no ripple effect, the **deadbeat type response** is achieved.

- We observed that no feedback was required during the above process. The condition was achieved as we stopped the input at the 3rd sample. The same states can be achieved earlier at **2nd sample** and settle there if the value of p = -1.

4.2 Condition No. 2

For values of a, b parameters and input u(k), assuming the condition x(3) = 0 as true we solve these step by step below:

$$x_1(3) = u(1) = 1$$

$$x_2(3) = b.u(0) + u(2) = 1$$

$$x_3(3) = b.u(1) = 1$$

4.2.1 MATLAB Code

```
clc;
close all;

% Constant Values
a = 0.5;
b = 1;

U = zeros(10);
X = zeros(3,10);

% Defining A,B matrix
A = [
    [0 1 0];
    [0 0 1];
    [a b -a/b];
    ];
B = [0; 1; 0;];
C = [[1 0 0];[0 1 0];[0 0 1];];
D = [0;0;0;];
```

```
% Case #2
U(2)=1;U(3)=1;
X(:,1) = [0;0;0;];

% Iteration
for i=2:10
    X(:,i) = A*X(:,i-1) + B*U(i-1);
end

X
```

4.2.2 Scope and limits of possible deadbeat type performance on MATLAB

4.2.2.1 Input given to the system

As u(0) + u(2) = 1 and u(1) = 1, assuming u(0) = p, we get u(2) = 1 - p. The range of our input will be of the format $u(k) = [p, 1, 1 - p, 0, 0, 0, 0, \ldots]$ where p is an arbitrary real constant value. For simplicity, here we took p = 0.

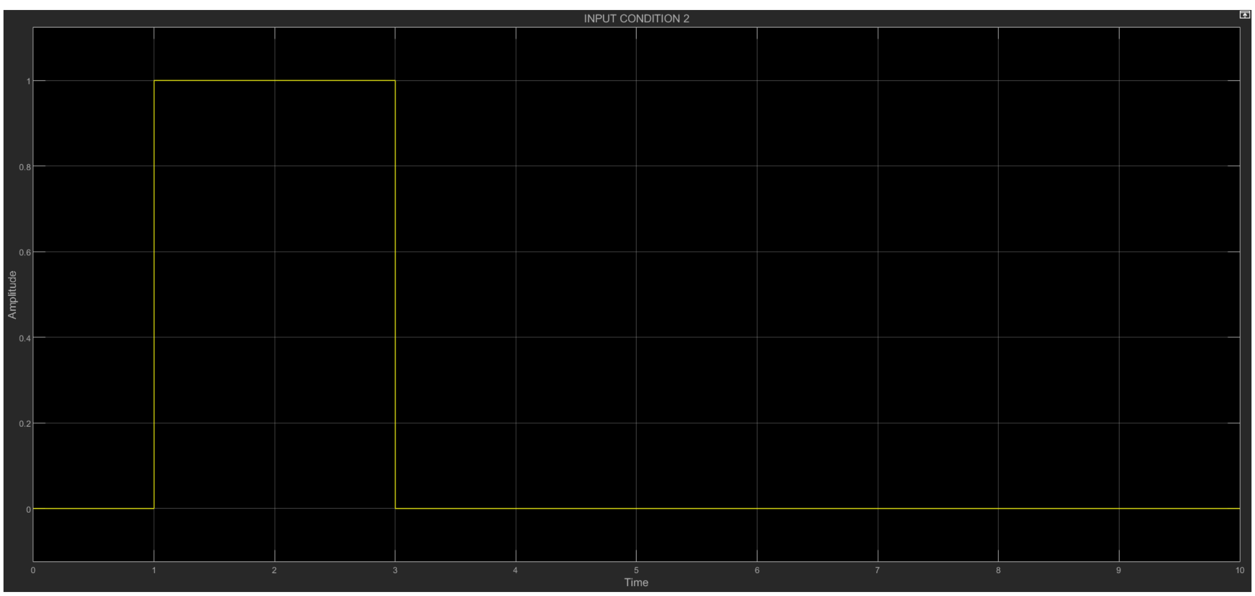


Figure 4.2.2.1.1 Input for system for condition 2

4.2.2.2 Simulink output representation

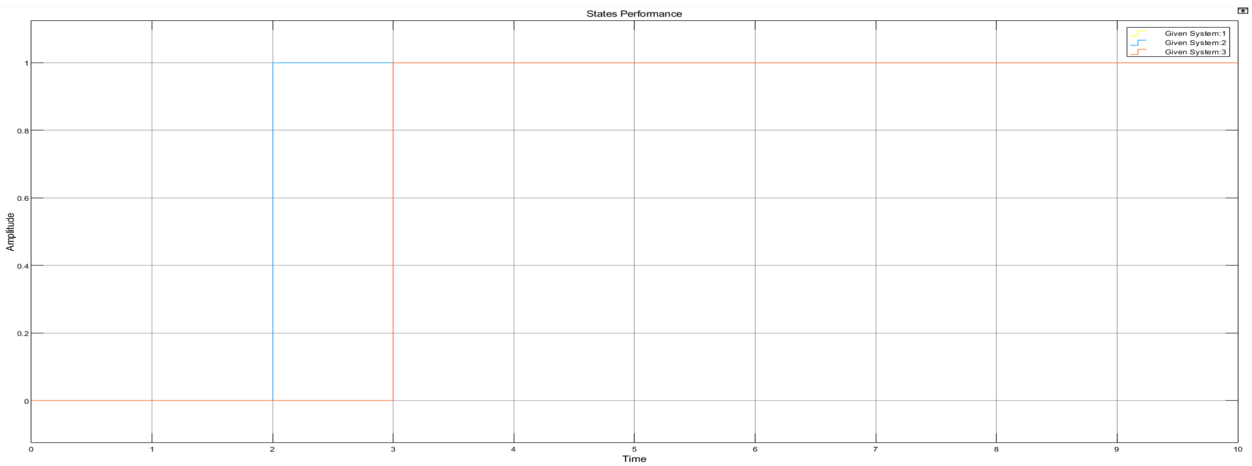


Figure 4.2.2.2.1 Simulink output representation for condition 2

4.2.2.3 MATLAB Output Representation

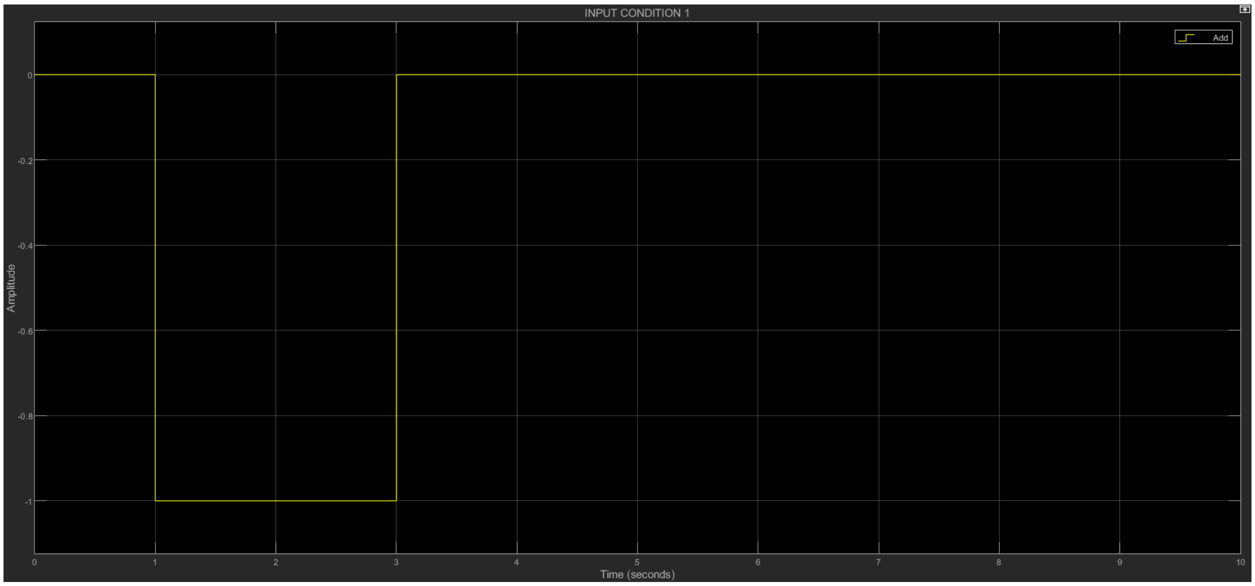
```
STATE_VALUES = 3×10
     0     0     0     1     1     1     1     1     1     1
     0     0     1     1     1     1     1     1     1     1
     0     0     0     1     1     1     1     1     1     1
```

4.2.3 Observation & Explanation

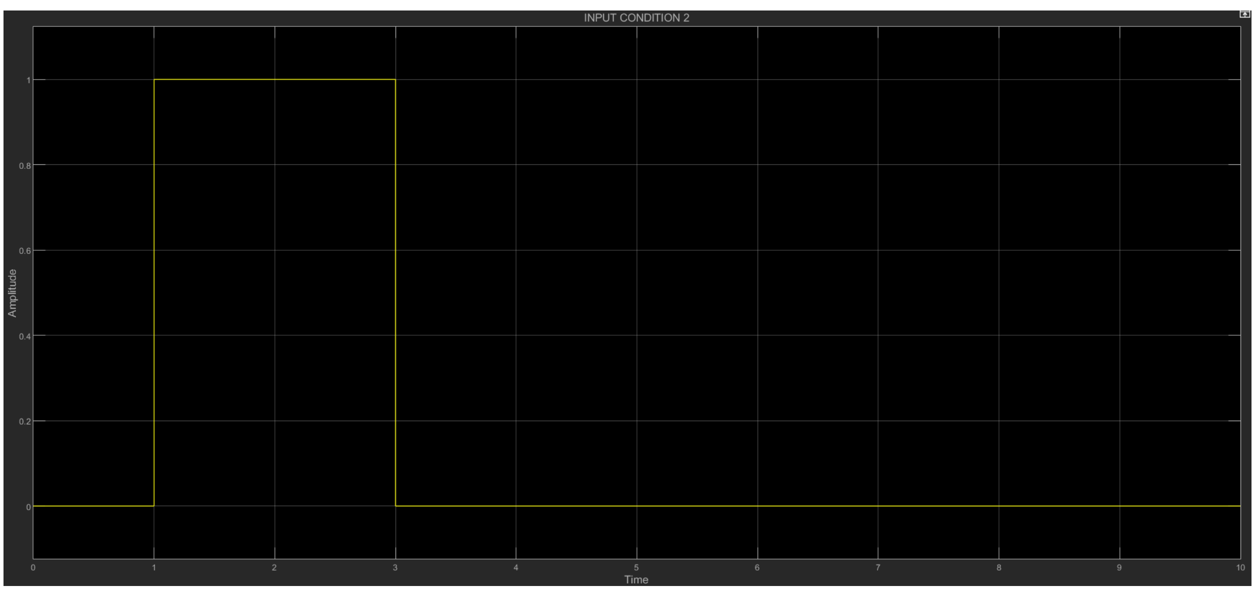
- In order to change the value of states from lower to higher values, we required a positive input(initial state of [0 0 0] to [1 1 1]). Once again in this case, as we keep u(k) = 0 for k greater than or equal to 3, the value of state settles again.
- As the states settle at their final values instantly and there is no ripple effect, the **deadbeat type response** is achieved again.
- We observed that no feedback was required for this condition also. The condition was achieved as we stopped the input at the 3rd sample. The same states can be achieved earlier at 2nd sample and settle there if the value of p = 1.

5. CONCLUSION

Condition 1 parameter & input :



Condition 2 parameters & inputs :



We simulated the state space matrix (assuming $C = I_3$ and $D = 0$, just for simulation purpose) and plotted the Simulink scope graphs (States v/s Sampling time) for both the conditions, and concluded that the calculations obtained manually are in total sync with the simulations result.

Hence we conclude the following from the experiment –

- For the first case, in order to satisfy the given condition of x(3) = 0.
  - The range of the parameters will be $a \in Real$ & $b = 1$.
  - The range of inputs can be $u(1) = -1$, $u(2) = p$, $u(0) = -1 - p$ otherwise $u(k) = 0$.
- For the second condition in order to satisfy the given condition x(3) = 1.
  - The range of the parameters equated were found to be $a \in Real$ & $b = 1$.
  - The range of inputs can be $u(1) = 1$, $u(2) = p$, $u(0) = 1 - p$ otherwise $u(k) = 0$.

So according to our requirements, once we obtain the given state values of x_3, we set the input u(k) to zero again for k greater than or equal to 3, to restrict the states from changing further. The system we chose **doesn't depend on the value of parameter a and the value of parameter b is 1**.