Chapter 4

SIMULATION EXPERIMENTS

E-laboratory provides facilities, for example user interface, graphical presentation and the text file available to be downloaded. Once the users place the value of parameters he want the system to execute and submit it, the graphical result will be presented. This graphic is a plot function result of the plant response. From this page we also provide link to download the text file of that graphic. Fig. 4.1 shown a web page in our application which is became the gate to enter to all experiments.

Figure 4.1: Main page of the experiments.

E-laboratoray is a virtual laboratory as well as a remote laboratory. It can conduct simulation and actual experiments. Simulations are prepared to enable the users understand the plant clearly. In the simulation, we use simulink and represent the real plant by a transfer function or state space. Beside can be used to perform experiments in the absence of real laboratories, simulation also believed as a suitable complement in achieving aim to teach the techniques and possible pitfalls of theoretically based design method when applied in practice.

Types of simulation experiments can be held are system analysis, system identification

and controller design. Those three experiments in simulation is applied to three respective plant platforms. Only flexible link is prepared with real experiments support which also able to run those three type of experiments. The following sections are presented to discuss and to explain the results of the development.

4.1 Flexible Link

4.1.1 System Analysis

The first type of experiment that can be conducted for every plants is system analysis experiment. This experiment is specially intended to give explanation to the users that assumed did not know about the plant yet. We also certainly provide the desription and it is necessary for them to read those files. Nevertheless, special aim of this experiments is to enable the users understand the plant by letting them observe the responses and the behaviour of the plant by simulation. To conduct the system analysis, we provided pre-defined controller for each plant to be explored. Using this controller, users can have some experiments to do, for example by tuning the gain, merely by changing some parameters and references to observe the response of the system.
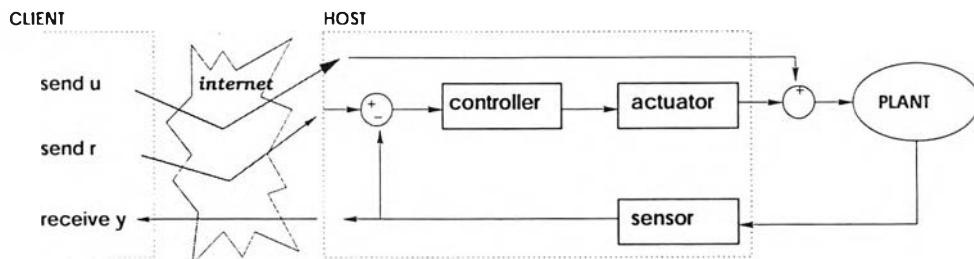

Figure 4.2: System analysis control design.

System analysis is designed as the diagram shown in Fig. 4.2. We can notice from this figure that user can send their command to the host computer and afterward, host computer will send the reference or input signal to the system simulations or experiment. The flow of the experiment is shown in Fig. 4.3.

Before we get into the experiment deeper, we may need to observe some behavior of that plant. The behavior of the plant which will be discussed here are as follow.

- Open-loop response

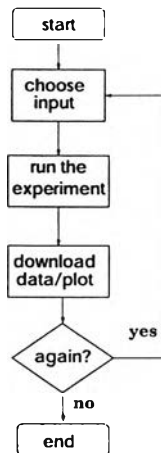- Sytem stability, controllability and observability

- Closed-loop response

Figure 4.3: Flow chart of system analysis.

Open-loop response

The system analysis is started by analysing the open-loop response. We recall the state representation of the flexible link, those are the matrices A and B as shown by the following matrices. For the time domain analysis, step response of the system is shown in Fig. 4.4.

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 566.46 & -37.022 & 0 \\ 0 & -921.77 & 37.022 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 65.113 \\ -65.113 \end{bmatrix}$$
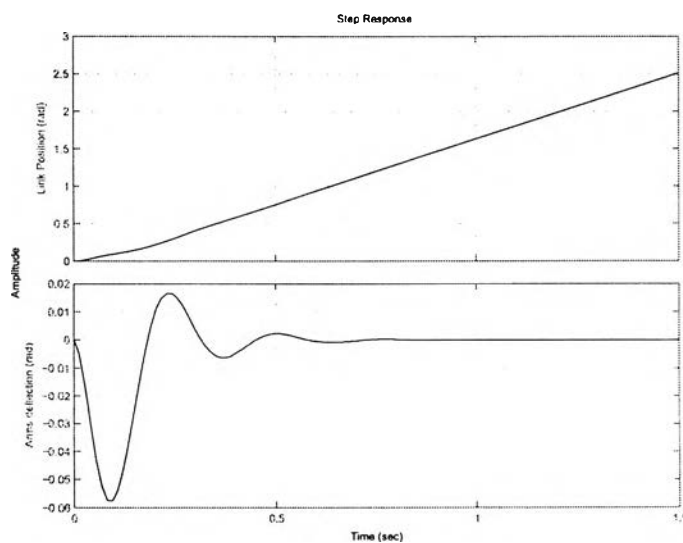

Figure 4.4: Step response of flexible link .

The plant has eigenvalues 0, -7.4138+23.1887i, -7.4138-23.1887i, and -22.1942. One of its eigen values is not located in the left hand side, which make the plant is not stable. This is also proved by the step response of the system. The above part of Fig. 4.4 shows the angle position of the link which is increasing as time goes by.

Subsequently, we need to design a controller that can stabilise the plant. First we have to check the controllability matrix of the plant. The controllability matrix, C_m is as follows.

$$C_m = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix}$$
$$= \begin{bmatrix} 0 & 65.113362837 & -2410.612148431 & 0.52361040781011 \\ 0 & -65.113362837 & 2410.612148431 & -0.29225888013643 \\ 65.113362837 & -2410.612148431 & 52361.040781011 & -5.72983963009677 \\ 65.113362837 & 2410.612148431 & -29225.888013643 & -2.83520414691549 \end{bmatrix}$$

The matrix C_m is full rank which is mean that the plant is controllable and we can design a controller to control the plant.

We also analyse the flexible link system in term of frequency domain. The frequency response of a system can be viewed two different ways via the Bode plot or via the Nyquist diagram. A Bode diagram consist of two graphs. One is a plot of the logarithm of the magnitude of a sinusoidal transfer function and the other is a plot of the phase angle. Both are plotted against the frequency in logaritmic scale.

In flexible link plant, we have two measured output, the position of the link and the deflection arm. We will observe the Bode Plot of those two states. From the state space representation, we can transform to transfer function, yields

$$\frac{\theta(s)}{u(s)} = \frac{-4.974 \times 10^{-14}s^3 + 65.11s^2 - 6.73 \times 10^{-11}s + 23140}{s^4 + 37.02s^3 + 921.8s^2 + 13150s}$$
$$\frac{\alpha(s)}{u(s)} = \frac{-3.553 \times 10^{-14}s^3 - 65.11s^2 - 4.002 \times 10^{-11}s}{s^4 + 37.02s^3 + 921.8s^2 + 13150s}$$

(4.1)

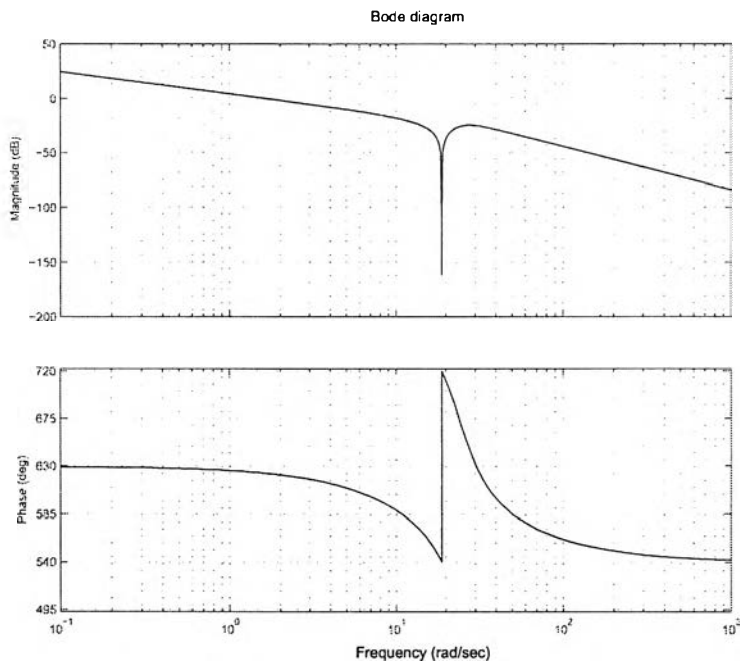The bode diagram of those two states are shown in Fig. 4.5 and Fig. 4.6
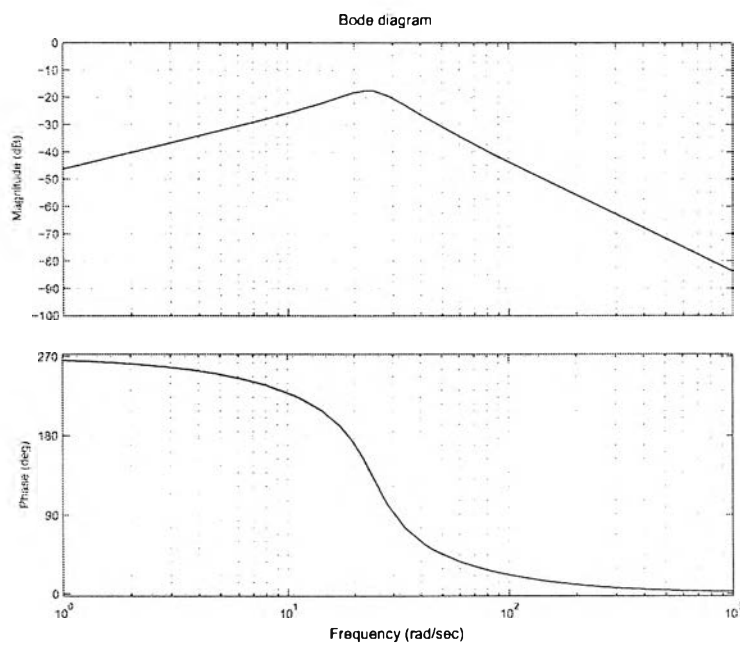
Figure 4.5: Frequency response of theta.

Figure 4.6: Frequency response of alpha.

Closed-loop response

As the sequence of the our analysis, we will observe what are the closed-loop response of the flexibe link plant. A simple, arbitrarily chosen propotional gain is applied in closed-loop

structure as shown in Fig. 4.7. Let us choose the gain ;
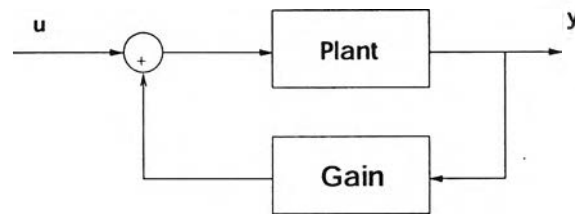
$$k = -\begin{bmatrix} 20 & 20 & 20 & 20 \end{bmatrix} \qquad (4.2)$$

Figure 4.7: Closed-loop condition.

Then we checked the step response of the formed closed-loop. The result is shown in figures 4.8. We can see that the closed-loop system responses the step input well, although not in a good performance.
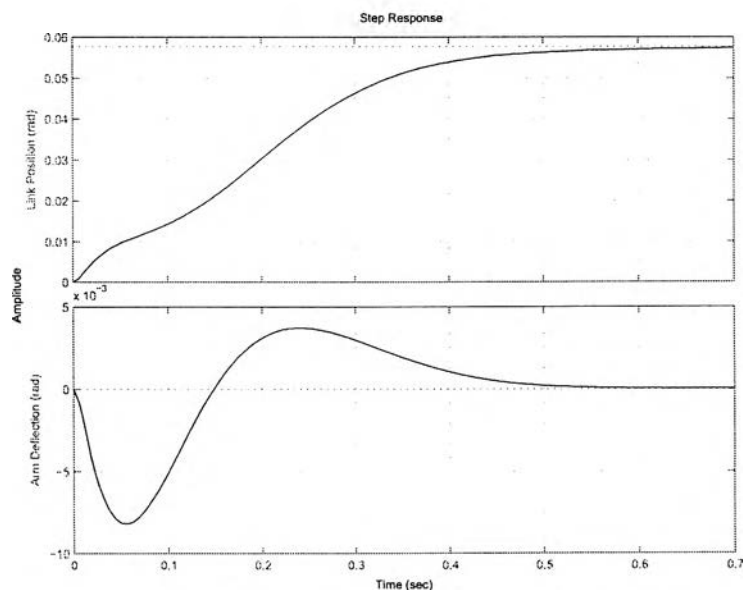

Figure 4.8: Closed-loop step response using simple gain.

A more specific examination is conduct by checking the response of closed-loop system to a reference. A square wave with amplitude equal to 30° is applied and we want the link position, θ, to follow this reference. Fig. 4.9 and 4.10 show the simulation result of link position and arm deflection respectively. Both figure show a very poor performance of the system that provided by the choosen gain. A better performance will be obtained using control regulator as LQR or pole placement as discussed in the next sections.
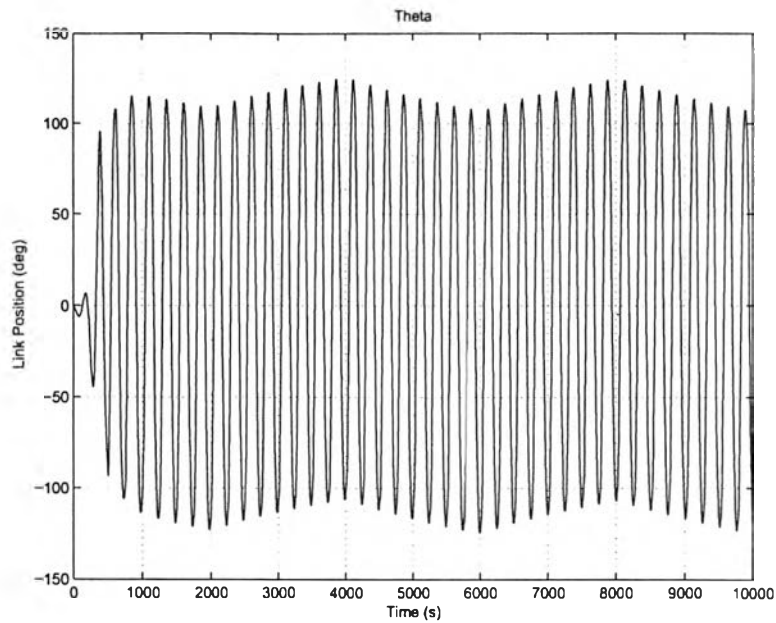
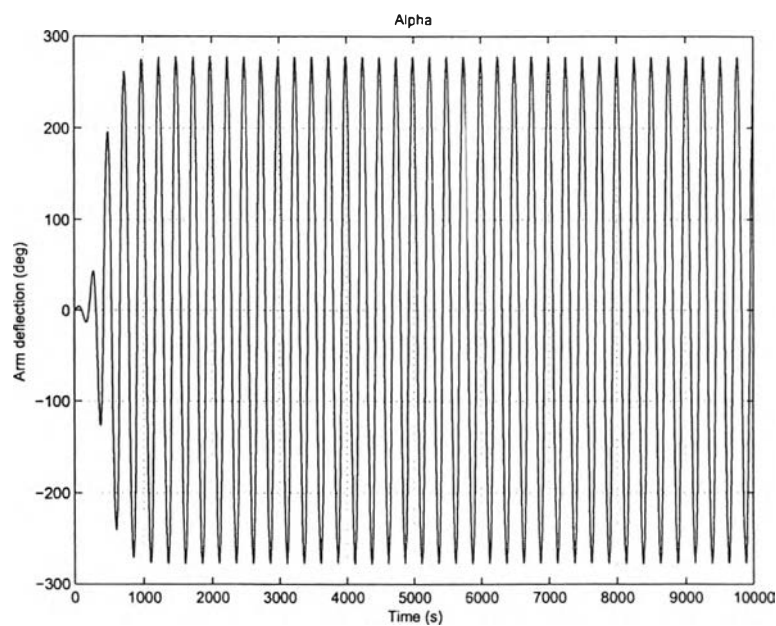Figure 4.9: Closed-loop response using simple gain to the reference of 30°.

Figure 4.10: Arm deflection to the reference of 30°.

In the experiment, the the pre-defined gain k controller of flexible link plant is design using LQR. The result can be tuned by filling the form that already provided or just leave it with the default values. The default values at the form is obtained from calculation using

MATLAB in order to direct the user not to fill a very different values from the optimal one.
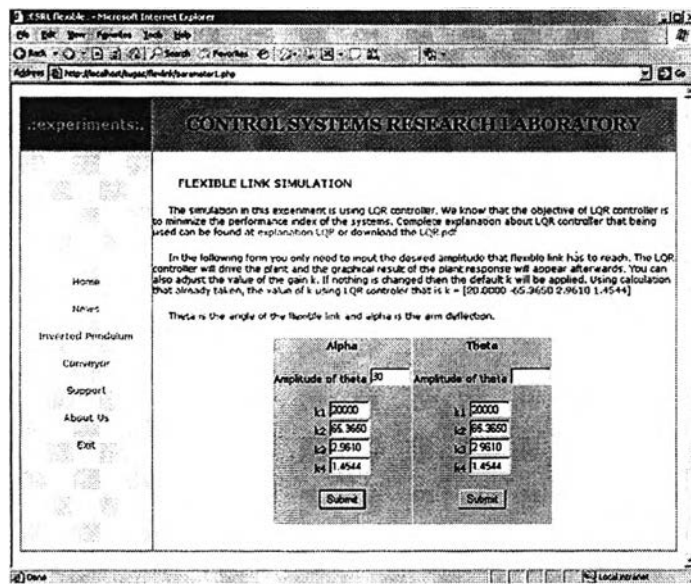

Figure 4.11: Web page for flexible link system analysis.

The procedure of system analysis experiment of flexible link plant is described as follows. At first user will access the interface as shown in Fig. 4.11. From this pages the users can starts to read the informations about the flexible link. What we write at this page is very basic information. For complete information, a link to a more comprehensive information page is provided. Another possibility is offered since we provided a pdf file that can be downloaded.

Notice that in the web page of this experiment we provide two columns of forms. Actually they are choices and therefore user cannot submit the value of those column in the same time. The left column is pointed for the user who want to know the response of the plant in term of vibration, α. The right column is pointed to users who request for angle of the flexible link, θ. But for all of those experiments, a compulsory parameter that should be filled is the desired amplitude of theta. Once the value is submitted, then the requested α or θ figure will be plotted and presented to the web page. Fig. 4.12 shows the web page presentation from a request by applying 30° as amplitude or angle movement of the flexible link, θ.

If the request responded succesfully, then we will have a link from the page which present the plot result of flexible link experiment to download the text file version. Using the downloaded file, users will be able to observe the response offline from their own computer to make any decision about the plant. Figure 4.13 shows the result of requesting text file from the host.
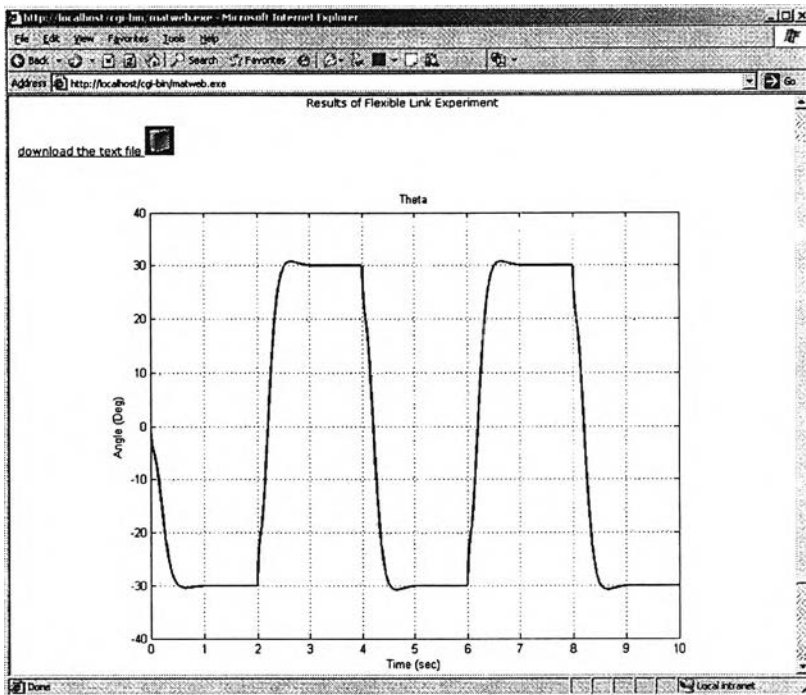
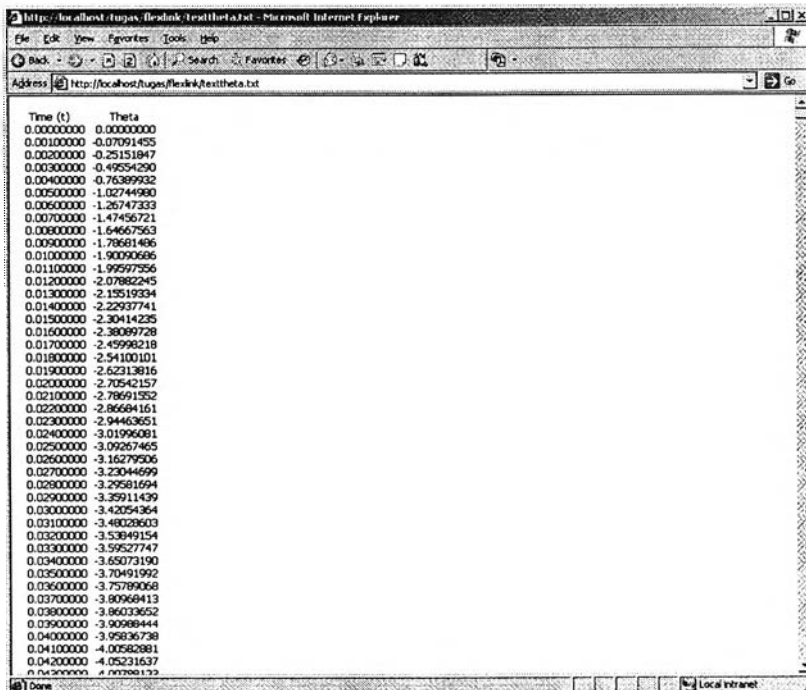Figure 4.12: Simulation result of the flexible link system analysis.

Figure 4.13: Simulation result of the flexible link system analysis in text file.

We have two types of experiments for flexible link plant. The first type of experiments utilyses a model in SIMULINK as plant representation. To find the model of the flexible link plant, we do the following steps.

We know that in the real experiment the flexible link plant has two encoders. One encoder is used to measure the angular position of the servomotor, θ, and another encoder is used to measure the arm deflection, α. We must use two observers to reconstruct the state variables $\dot{\theta}$ and $\dot{\alpha}$. Remembering that we cannot implement a pure derivative we use low-pass band- limited observer.

$$\frac{150s}{s + 150} \tag{4.3}$$

This filter is tuned to reduce the noise introduced in the numerical derivative. In SIMULINK, we apply this by put a transfer function with value as shown in equation 4.3 to obtain the $\dot{\theta}$ at the branch of the line from the angular position encoder. Another low-pass band limited filter is added to obtain $\dot{\alpha}$ as shown at equation 4.4. This transfer function block is placed after the arm deflection encoder.

$$\frac{50s}{s + 50} \tag{4.4}$$

This SIMULINK model then combined with M-files to be used in both simulation and real experiments.

4.1.2 Identification

The aim of the system identification is to find a suitable model which approximates the real plant. This experiment is performed by exciting the system and observing its input and output over a time interval. Before one can conduct identification experiments, he should already has pre- identification knowledge. The prior knowledge is about an identification technique and a performance index used as a criterion to calculate the model. The significant aspects of the experiment design are how to choose the input signal and sampling period so that the collected data are informative and suitable for identification process. This knowledge is important because it will decide wether the result from the identification experiments are useful or not. In this application, we provide plants or model to be identified with the final decision is on user.

A procedure to follow by the user to obtain the data for identification is shown in Fig. 4.14. The procedure is used by all identification experiments in E-laboratory. There are three inputs can be used, with the information results we provided are in text files and graphics. The three of inputs are choosen for the reason of its benefits since they are signal that most likely used in identification. To start the experiment, the users should access or
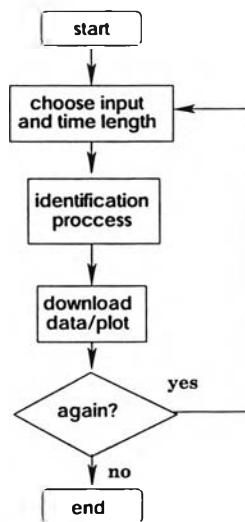
Figure 4.14: Procedure to obtain data for identification.

open the web page and from the experiments page, the user can choose the *Identification* because every plants are equiped with that link.

In practise, no matter how well the analytical modelling is performed, some identifications are always needed. Important fact that make real hardware and simulation experiments are simply different locate on the unmodelable effects and the sensitivities of real hardware. Sometimes they cannot be modelled. Therefore bring this experiment along with the E-laboratory could deliver useful experiences to the users.

An example of identification techniques is presented here. We will identify using a pair of information consist of PRBS input and its plant response. Identification process will use MATLAB and the procedure is as follows. At first measured information is taken from the measurement, in this manner we download from the website of identification result. Some analysis will be conducted here except the transient analysis due to unappropriate used data.

From system analysis section we know that the system is unstable. This fact mean that any measurement from open-loop configuration will be useless. We must measure the data from the closed-loop system. Furthermore, after having been collected from the experiment, the data should be preprocessed because they may not be ready to be used instantly. For example, if there are signal offsets, they should be removed or estimated explicitly. Prefiltering the input and output data by using the same filter can remove undesired disturbances of high or low frequencies. This will not change the input-output relation for a linear system.

Non parametric method

Transient analysis

Transient analysis was conducted by applying step input to the plant. Nevertheless, we already know from system analysis section that the system is not stable so that we cannot use this method to identify the system.

Correlation analysis

By using correlation analysis, we will find the impulse and step response of the plant. This following command is applied in MATLAB to obtain the figures 4.15, and 4.16.

```
x=data(:,1);
y=data(:,2);
mtr=iddata(y,x,0.001);
mtr.InputName='Vin';
mtr.OutputName='Vout';
mtr.InputUnit='Volt';
mtr.OutputUnit='Volt';
ze=mtr(1:600);
plot(ze(100:200));
ze=detrend(ze);
impulse(ze,'sd',3,'fill')
```
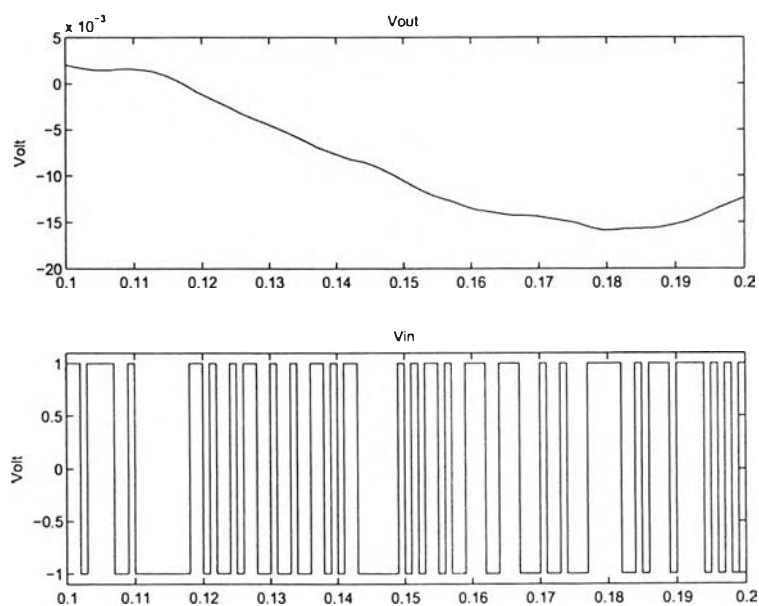

Figure 4.15: Model data plot.
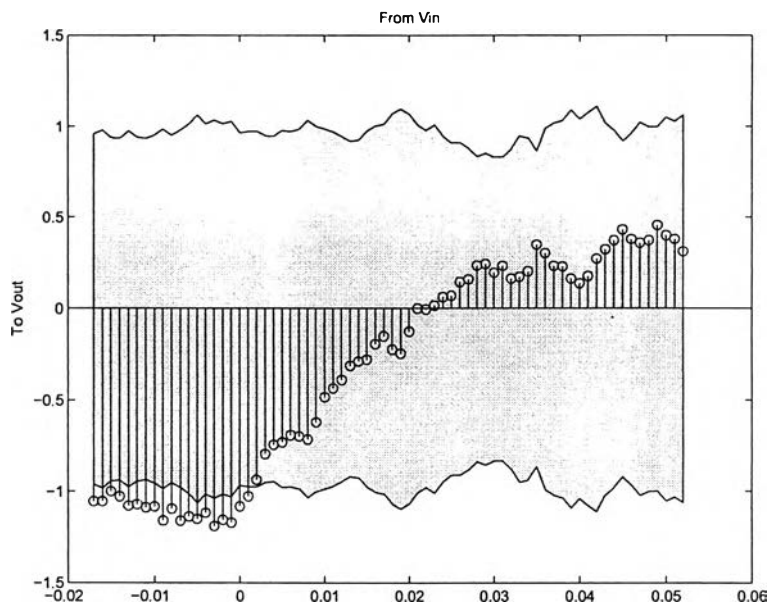
Figure 4.16: Impulse response obtained from correlation analysis.

By using the impulse response, we can obtain a plot with fill region marking a confidence region. Unfortunately we cannot find such a confidence value to decide whether theres a time delay or not. To check this, later we try another method to find out the time delay using n4sid and arxstruc command.

Spectral analysis

Principally, the spectral analysis can be applied with rather arbitrary inputs. the transfer function is obtained in form of a Bode plot. To obtain the figure needed in spectral analysis, the following commands are used and yield figures 4.17 and 4.18.

```
g=spa(ze);
bode(g);
ffplot(g);
```
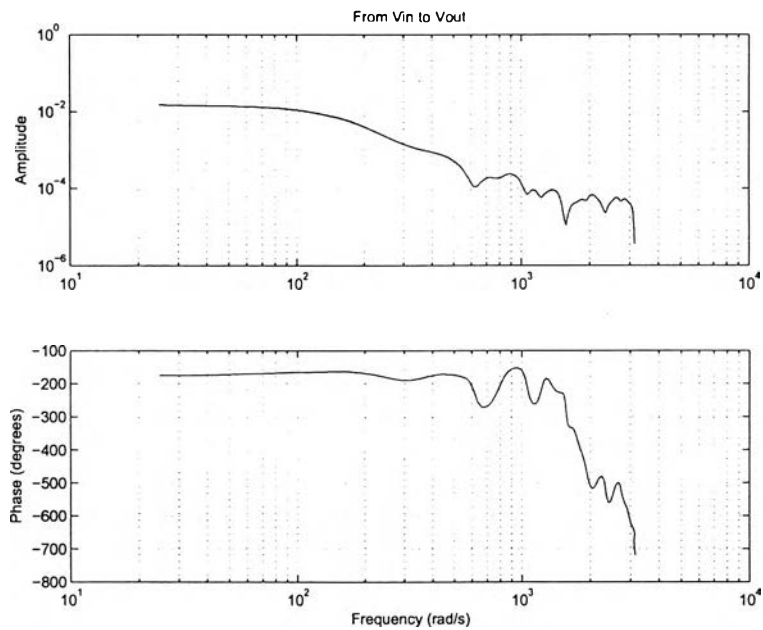
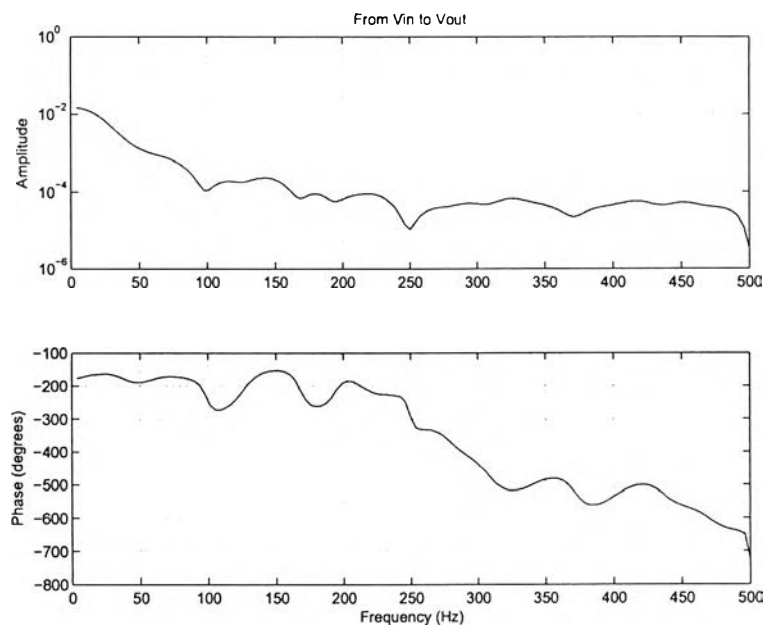Figure 4.17: Bode plot for spectral analysis in rad/s.

Figure 4.18: Ffplot for spectral analysis in frequency (Hz).

Frequency analysis

The input to the plant is sinusoidal with frequency value. One sinusoidal inputed mean that one pair of relation in a frequency is obtained. In order to have a complete estimation of plant,

we need to collect as many as possible pair between input and output in many frequency. By then we able to get the data of magnitude and angle of the transfer function $G(s)$ versus the frequency. We can plot this to obtain the bode plot of the transfer function $G(s)$. An example of a pair of input and output at frequency = 1 rad/sec is shown in Fig. 4.19.
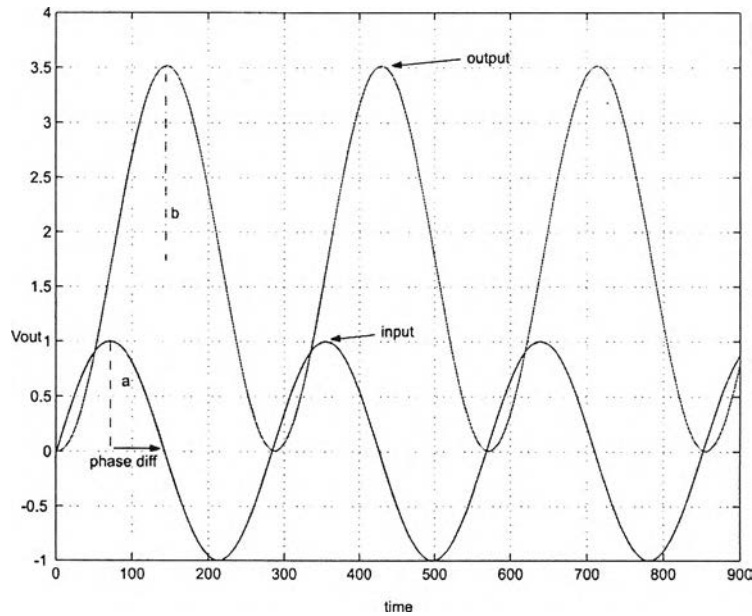

Figure 4.19: Output and input of the plant with f = 1 rad/sec .

Recalling equation (3.17), combining with this figure, we aproximate that the parameter of $G(s)$ for f = 1 rad/s is

$$|G(i\omega)| = 3.5$$
$$\arg|G(i\omega)| = 73^o$$

Parametric method

Using parametric method, well try to find the best model among these models:

1. State space model

2. ARX model

3. ARMAX model

State space model

We do the state space identification by continuing   the previous command with the following command and the result is showed in Fig. 4.20.
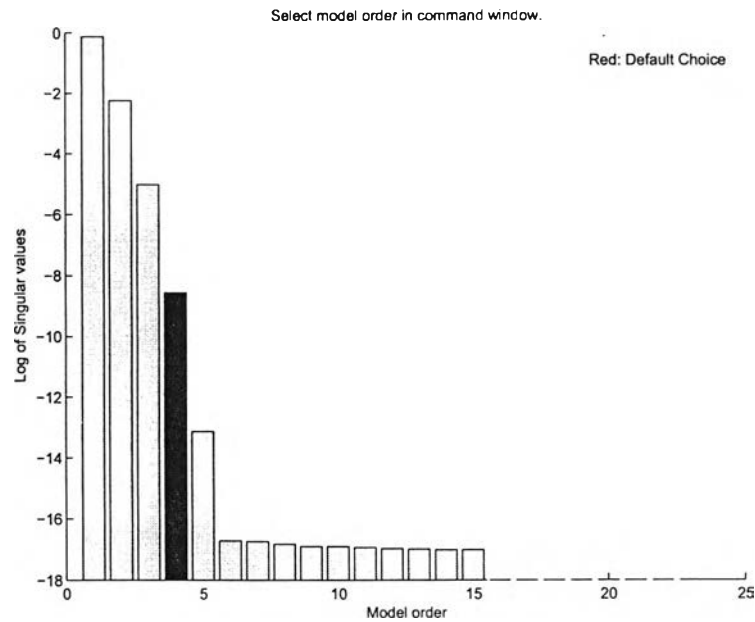
```
ml=n4sid(ze,1:15)
```


Figure 4.20: Comparison of different order of state space.

The red color bar is the default choice by MATLAB and we can use this as our choice as well. Applying this we get the following result.

$$
A = \begin{bmatrix} 0.99419 & -0.024575 & -0.0017074 & -6.1197 \times 10^{-6} \\ -0.0093377 & 0.97223 & -0.017245 & 0.0014465 \\ 0.0043269 & 0.030154 & 1.0001 & 0.014872 \\ -0.0019076 & -0.010584 & -0.016704 & 1.0005 \end{bmatrix} \quad B = \begin{bmatrix} 0.0003567 \\ -0.013294 \\ -0.0048945 \\ 0.0032317 \end{bmatrix}
$$

$$
C = \begin{bmatrix} 0.18971 & 0.0014987 & 0.0031299 & -8.1636e-005 \end{bmatrix} \quad D = \begin{bmatrix} 0 \end{bmatrix}
$$

ARX model

ARX model is obtained by using these commands.

```
zr=mtr(800: 950);
zr=detrend(zr);
```

```
V=arxstruc(ze,zr,struc(2,2,1:10));
nn=selstruc(V,0)
```

The interactif response from MATLAB will give

```
nn =    2    2    1
```

At the following step, we use that result to be involved in the command below and yield the model as written by equation (4.5).

```
m2=arx(ze,[2 2 1])
```

$$A(q)y(t) = B(q)u(t) + e(t) \quad \text{with}$$
$$A(q) = 1 - 1.964q^{-1} + 0.9641q^{-2} \tag{4.5}$$
$$B(q) = 3.246 \times 10^5 q^{-1} + 3.203e - 005q^{-2}$$

ARMAX model

To choose the best order model from ARMAX model, first we build ARMAX model of different order. We compared the data fit of measurement and output of our model (at each order) start from 1st to 10th order. The procedure obtain ARMAX model

$$y(t) = \frac{3.218 \times 10^{-5}q^{-1} + 3.178 \times 10^{-5}q^{-2}}{1 - 1.964q^{-1} + 0.9644q^{-2}}u(t) + \frac{1 + 1.548q^{-1} + 1.02q^{-2}}{1 - 1.964q^{-1} + 0.9644q^{-2}}e(t)$$

The last step is to validate the model. We wil compare the results from state space model and ARX model with the original model. The command is as follow, and yields Fig. 4.21. The comparison show that state space and ARX model identification have the fittest model compared with validation data.
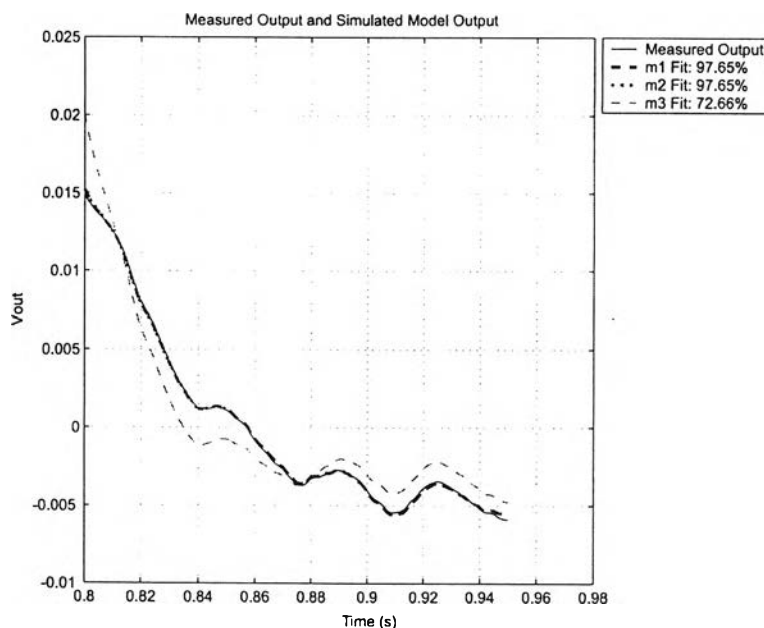
```
compare(zr,m1,m2,m3)
```

Figure 4.21: Comparison between measured output and simulated model output of state space (m1), ARX model (m2) and ARMAX model (m3).

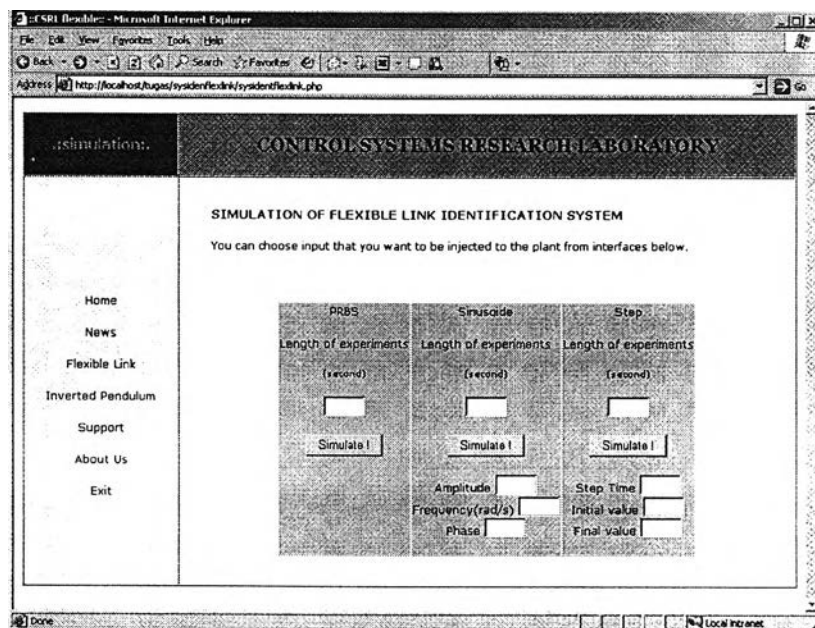The interface for the identification experiment of flexible link plant is shown in Fig. 4.22.

Figure 4.22: Web page for flexible link identification.

From this interface the input to the process can be selected. Also the parameters of the input can be selected among a set of predefined inputs, including the length of the experiment. They are pseudo random binary sequences, sinusoidal, and step. Users are enabled to change the length of the experiments to decide how long the identification will be taken place to obtain enough data for him. An example of simulation result when prbs signal is applied can be seen at Fig. 4.23.
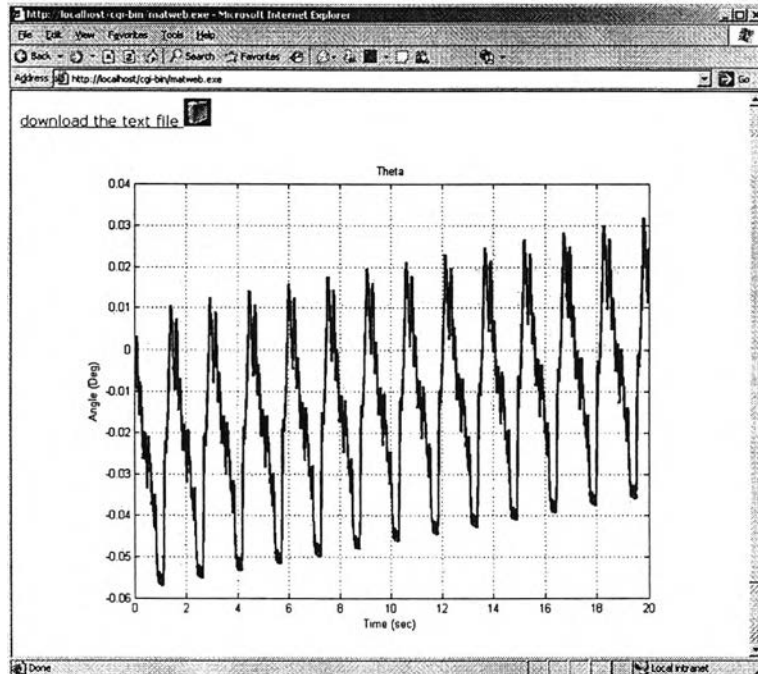

Figure 4.23: Simulation result of flexible link identification.

4.1.3 Controller Design

Controller design experiment is an expriments which enable the users to upload their controller file. To do this the experiment should be at stopped position and then user uploads their controller. To deal with different input that users may need for their controller, there will be a specific design of closed-loop schema as the master design. Controller design system that being developed in E-laboratory can be shown in Fig. 4.24. As stated before, Flexible Link is equipped with pre-defined controller. Instead of using this controller, user are allowed to design their own controller. How user can use their controller is explained in a procedure as follows.

1. Observe Fig. 4.25 to find a link to download a template file. This template file is a model of the plant which developed using SIMULINK.
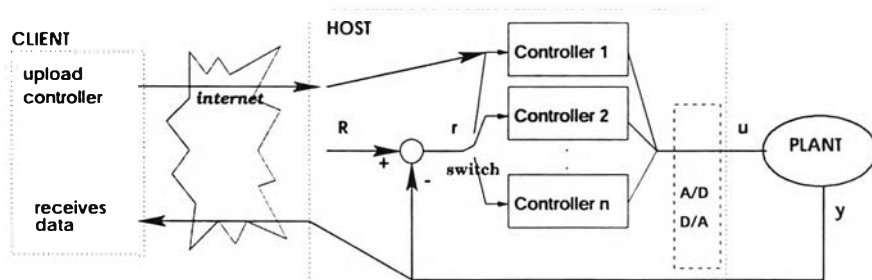
Figure 4.24: Switching the controller.

2. Once template model is downloaded, user may edit and change parameter inside the model.

3. The template model file consist of two main part, the plant and the controller. It is can be seen from Fig. 4.26. User need only to change the parameter that are in the controller subsystem.

4. User analyze and try the controller in his computer, computer where client or browser located.

5. The finished controller design by user can be uploaded to the server.

6. User then executes their new controller and get the data from the experiment.

An example of computing the controller design using LQR is showed as follows. Some value of the parameters that employed in our calculation during designing the controller system are the weighting matrices, they are Q and R

$$Q = \begin{bmatrix} 400 & 0 & 0 & 0 \\ 0 & 10000 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \text{ and } R = 1$$

These parameters bring optimal feedback gain k

$$k = \begin{bmatrix} 20 & -93.5614 & 3.3621 & 0.9623 \end{bmatrix} \tag{4.6}$$

This gain then applied as our default predefined controller. This is the value of the default controller that is presented in the web page. User then can try some reasonable value of gain and enter it in the web page interface and submit it to see what are plant responses.

Using MATLAB we can use this value to generate figure which shows the position of flexible link at each time instant. The generated value from the simulation is depicted in
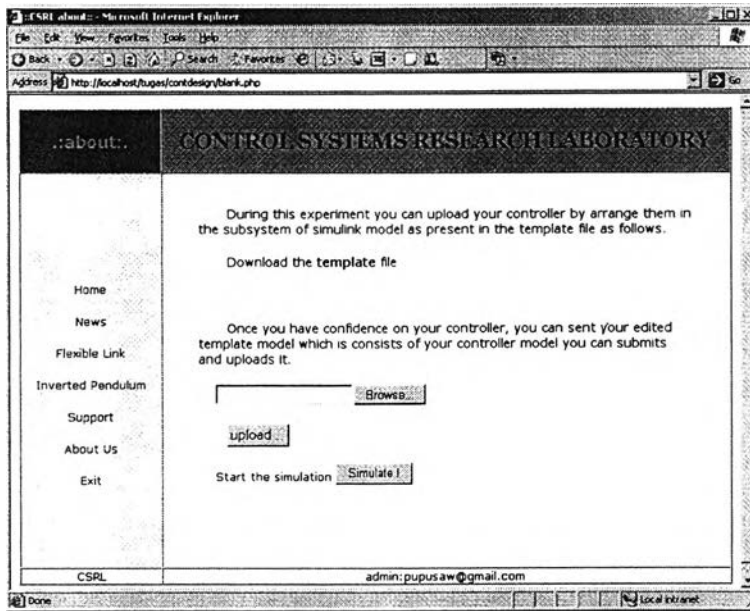
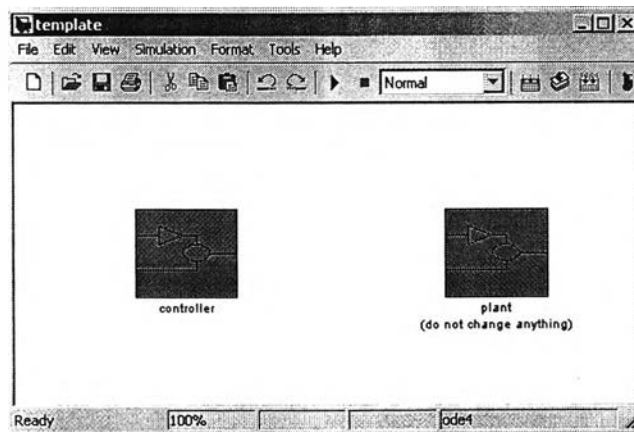Figure 4.25: Web interface to upload user controller.

Figure 4.26: Template model file.

Fig. 4.27. We can compare between result from this MATLAB plot and result from web page to know whether there are any differences.

The solid line is the plot of the angular position and the solid and dash line is the plot of arm deflection. We see that the response of the plant is quite slow as it will reach the reference given almost at half time of its peak periode. Considering this, user can optimize the problem by using, for example, pole placement control strategy.
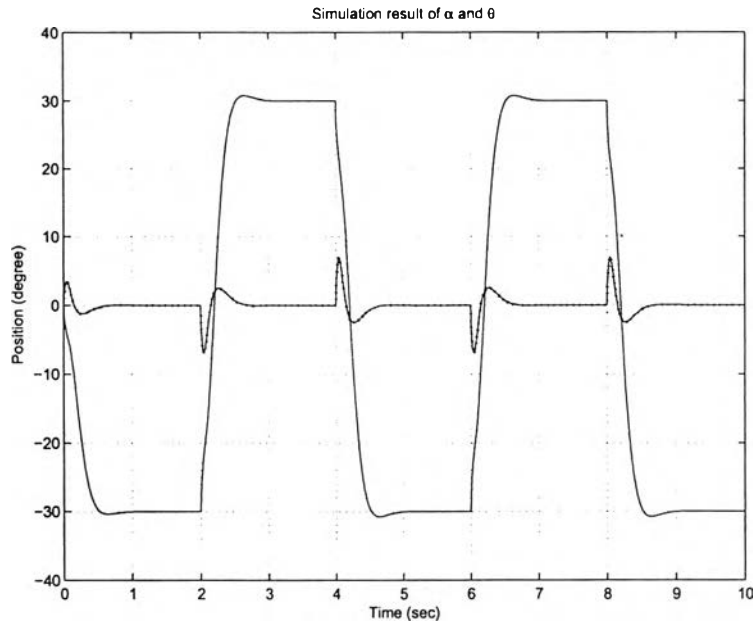
Figure 4.27: Simulation result using MATLAB.

4.2 Rotary Inverted Pendulum

4.2.1 System Analysis

Open loop analysis

To conduct the system analysis on this system, we recall the state space representation of the plant as shown in equation (4.7).

$$
A = \begin{bmatrix}
0 & 0 & 0 & 1.0000 & 0 & 0 \\
0 & 0 & 0 & 0 & 1.0000 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.0000 \\
0 & 4.2017 & 1.7791 & -0.3247 & -0.0222 & 0.0095 \\
0 & 28.1858 & 0.8197 & -0.1496 & -0.1486 & 0.0044 \\
0 & -2.9100 & -40.6937 & 0.2249 & 0.0153 & -0.2162
\end{bmatrix}, \quad
B = \begin{bmatrix}
0 \\
0 \\
0 \\
0.0302 \\
0.0139 \\
-0.0209
\end{bmatrix}
$$

$$\tag{4.7}$$

$$
C = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix}, \quad
D = \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
$$

The system's eigen values are 0, 5.2220, -5.3922, -0.1128±6.3752i, and - 0.2938. It has three main transfer functions which come from its measured variable, they are the angular

displacement of the rotating disc (α), the 1st pendulum angular displacement respect to the vertical axis (β_1), and the 2nd pendulum angular displacement respect to the vertical axis (β_2). By transforming the state space representation, to the transfer function, yields

$$\frac{\alpha(s)}{u(s)} = \frac{0.0302s^4 + 0.01051s^3 + 0.3998s^2 - 0.001291s - 31.28}{s^6 + 0.6895s^5 + 12.65s^4 + 4.252s^3 - 1145s^2 - 336.3s}$$

$$\frac{\beta_1(s)}{u(s)} = \frac{0.01391s^4 + 0.002917s^3 + 0.549s^2}{s^6 + 0.6895s^5 + 12.65s^4 + 4.252s^3 - 1145s^2 - 336.3s} \tag{4.8}$$

$$\frac{\beta_2(s)}{u(s)} = \frac{-0.02091s^4 - 0.002894s^3 + 0.549s^2}{s^6 + 0.6895s^5 + 12.65s^4 + 4.252s^3 - 1145s^2 - 336.3s}$$

As time response test is held, we can see that the system is not stable. It is shown by the step response of each measured variable, as shown by Fig. 4.28. The system has one input that is voltage. All of the state tend to go to infinity as the time goes by. Their respective bode diagram is shown in Fig. 4.29 a, b and c.
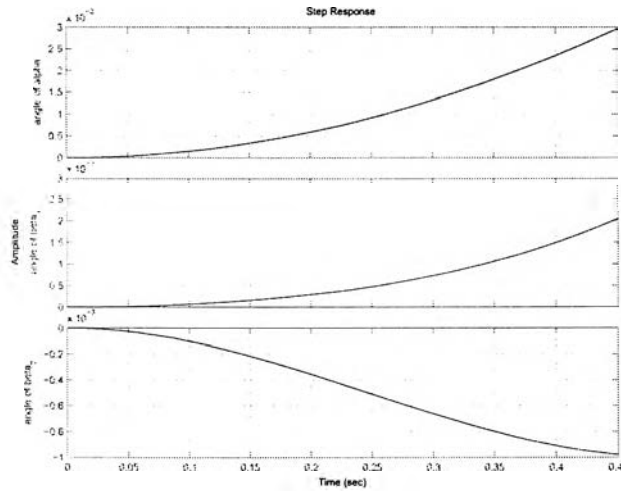

Figure 4.28: Step response of the measured variables of rotary inverted pendulum.

E-laboratory provide facility for user to change this gain. By using existing LQR controller as its pre-defined controller, user can conduct some experiments by changing the value of gain k. As explained in the book of instruction manual, [15], the rotary inverted pendulum plant has six variables of interest. The values that already inside the form are the default value from calculation. Users can try to change the values of these gain component to see what will happen to the plant. Fig. 4.30 shows the web page of inverted pendulum analysis experiment.
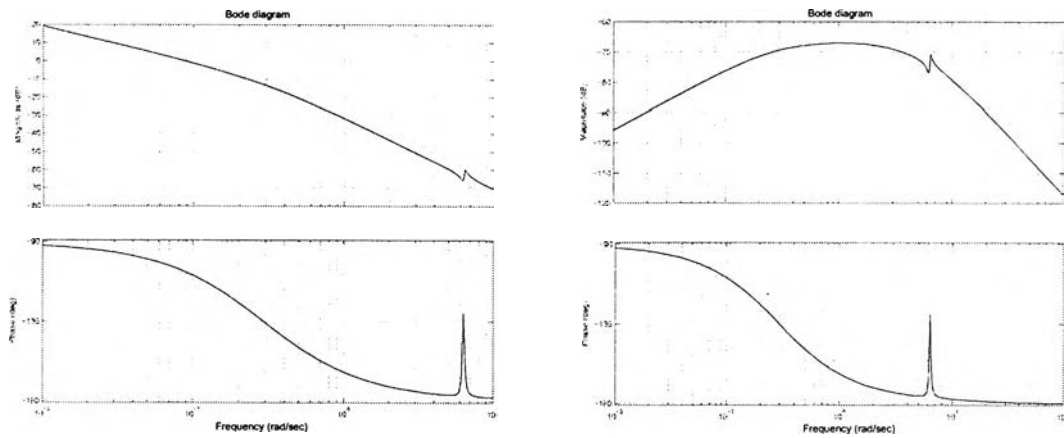
(a) The angular displacement of the rotating disc.

(b) The 1st pendulum angular displacement respect to the vertical axis.
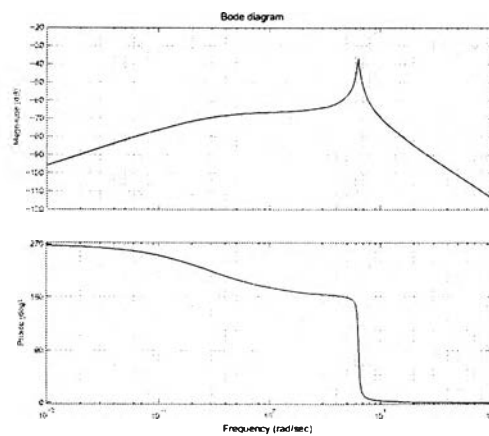
(c) The 2nd pendulum angular displacement respect to the vertical axis.

Figure 4.29: Frequency response.

Closed-loop analysis

Closed-loop analysis carried out by the same diagram as shown in Fig. 4.7. Recall the state space representation with the value of A and B as stated in equation 3.10 to be our model. It is also necessary to check the controllability matrix of the plant to check whether it is
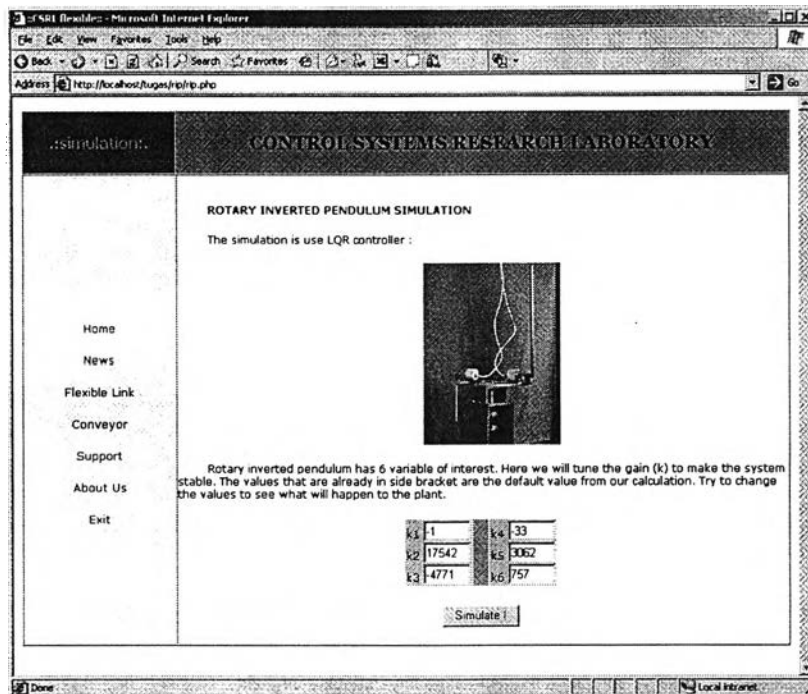
Figure 4.30: Web page for rotary inverted pendulum system analysis.

possible to design a controller which able to stabilize the plant. The controllability matrix is

$$C_m = \begin{bmatrix} B & AB & A^2B & A^3B & A^4B & A^5B \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0.0302 & -0.0103 & 0.0249 & -0.0164 & 3.0246 \\ 0 & 0.0139 & -0.0067 & 0.3776 & -0.2350 & 11.3374 \\ 0 & -0.0209 & 0.0115 & 0.8057 & -0.6125 & -33.7604 \\ 0.0302 & -0.0103 & 0.0249 & -0.0164 & 3.0246 & -3.6295 \\ 0.0139 & -0.0067 & 0.3776 & -0.2350 & 11.3374 & -9.4107 \\ -0.0209 & 0.0115 & 0.8057 & -0.6125 & -33.7604 & 33.7607 \end{bmatrix}$$

The matrix C_m is full rank and certainly means that the plant is controllable. Then here we tried a gain k which arbitrarily choosen with

$$k = \begin{bmatrix} 10 & 10 & 10 & 10 & 10 & 10 \end{bmatrix}$$

The step response of the plant when this gain applied is as shown by Fig. 4.31. The system is not stable since neither position of rotating disc and both pendulum converge to some values. To have the stable plant we have to use a control design as discussed in section 4.2.3

Another thought that should also be considered is about the initial position of the pendulum. Since our system is linearized around $\beta_1 = 0^o$ and $\beta_2 = 180^o$, we cannot put our
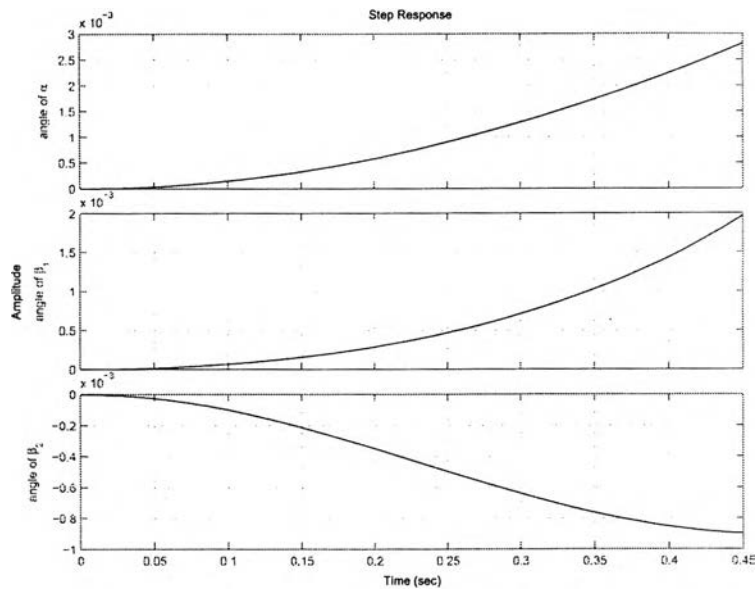
Figure 4.31: Simulation of double inverted pendulum result using arbitrary gain.

pendulum to far from that position. We limit the initial of the 1st pendulum not more than 10° around 0°. The initial position of 2nd pendulum should be not be more than 30° around 180°. We can compare the figure that generated in the web page and one which created from
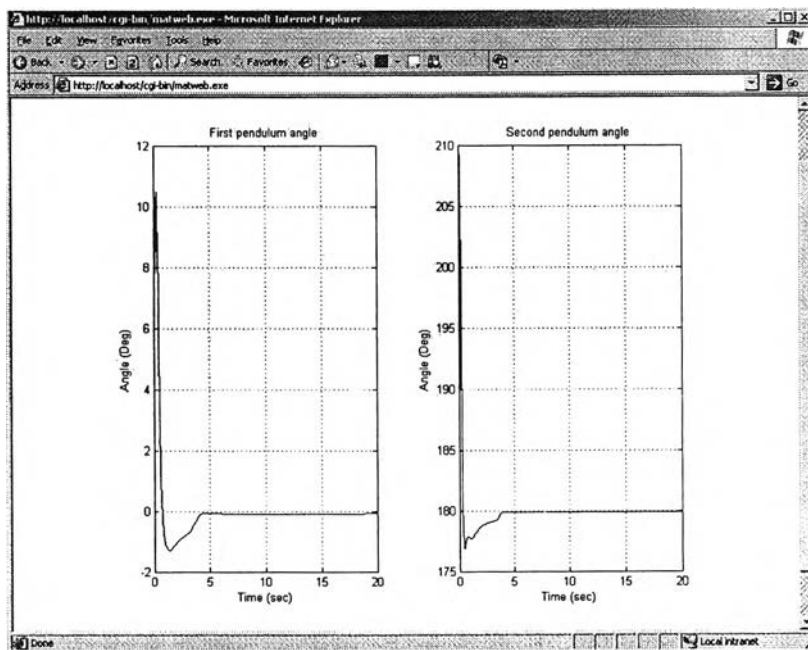

Figure 4.32: Simulation result of the rotary inverted pendulum system analysis.

the simulation in MATLAB as shown in Fig. 4.33. Both figures are identical and prove that the desired condition for both position of the pendulum is reached.
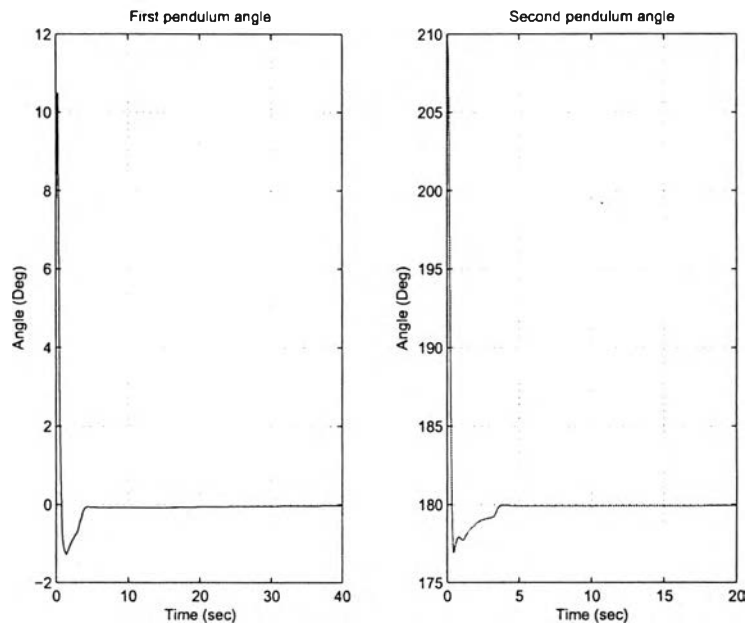
Figure 4.33: Simulation of double inverted pendulum result using MATLAB.

4.2.2 Identification

Rotary inverted pendulum is a quite complex plant. To identify this plant we need some steps of identification. In fact in our system, for others experiment we use a linearized model which in some way do not able to show the genuine condition. That is why in our application, we do not provide the simulation experiments of rotary inverted pendulum since it would not represent the real condition.

4.2.3 Controller Design

Generally, the interfaces to conduct the controller design between three plants are similar. The difference located on the template file that needed to be downloaded by clients. Each of them was developed dependent to what kind of controller are going to be used. Web page presentation for control design experiment for rotary inverted pendulum is shown by Fig. 4.34.
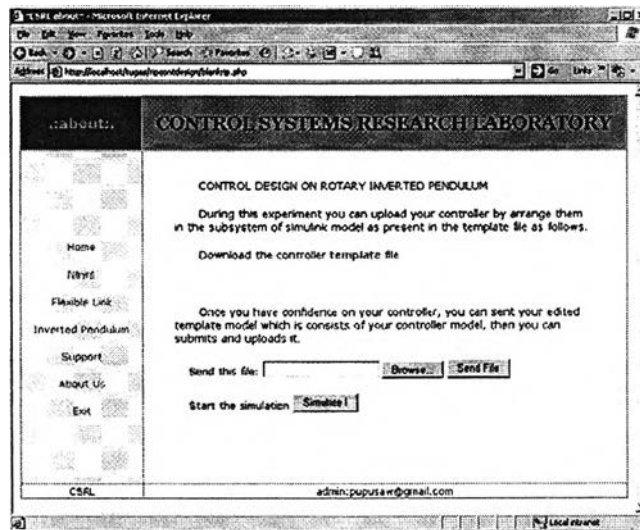

Figure 4.34: Web page for rotary inverted pendulum control design.

As an example, a closed-loop gain controller obtained from LQR design is presented here. This controller use weighting matrices Q and R which actually can be changed arbitrarily, depending on the performance specification. In this simulation we use Q and R as follows.

$$Q = \begin{bmatrix} 100 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5.e9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.e9 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad R = 100;$$

Using those weighting matrices, we have the optimal feedback gain k as follows.

$$k = \begin{bmatrix} -1 & 17542 & -4771 & -33 & 3062 & 757 \end{bmatrix}$$

Applying the gain k to the closed-loop system, we obtain that the plant is stable. It can be checked by the time response analysis using step input as shown by Fig. 4.35 and the frequency response from three measured variables, α, β_1, and β_2 as shown by Fig. 4.36.
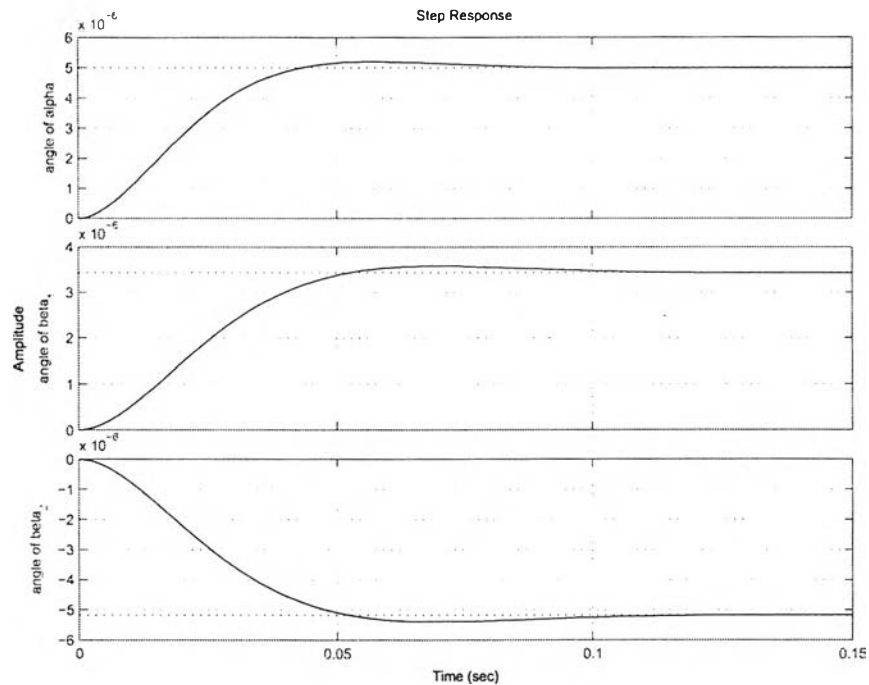
Figure 4.35: Step response of closed-loop system.

4.3   Belt Conveyor

4.3.1   System Analysis

The belt conveyor system is represented in the experiment using transfer function. The model constructed in closed-loop condition as depicted in Fig. 3.3. During the experiment this model is used as the platform and combined with PID controller as its pre-defined controller.

Firstly we checked the step response of the system. As can be seen from the section 3.1.3, the conveyor belt plant is constructed in a closed-loop system. The step response itself show that the plant is stable, as shown in fig. 4.38. However, the performance is not satsfying as it has slow response. We will design a controller afterward which can reach such a specified performance.

The belt conveyor is the only plant in our system to be given PID controller design as its pre-defined controller. Notice at the web page interface for belt conveyor analysis experiment as shown by Fig. 4.37. Two columns, the left side is for PID gain parameters and the right side is for input parameters, are presented. Using this interface we can tune the value of PID gain in accordance with the objective. We can observe what is the behavior of the plant if we apply a step input into the plant.
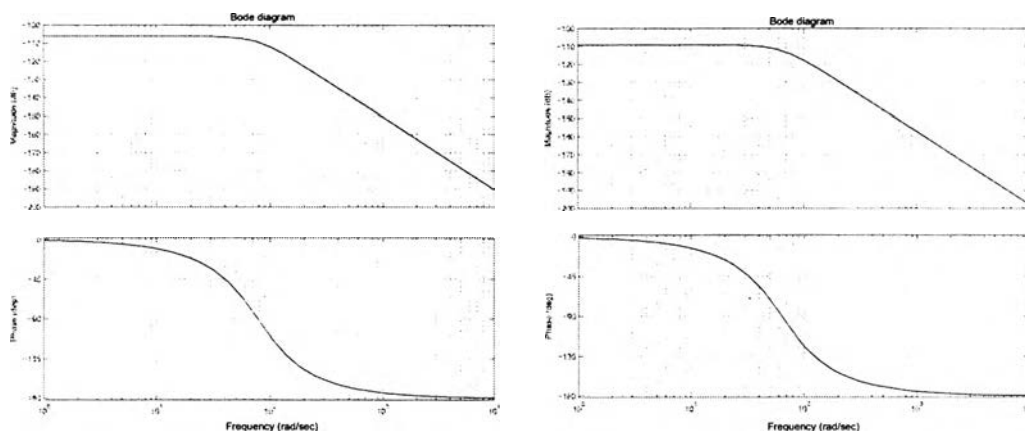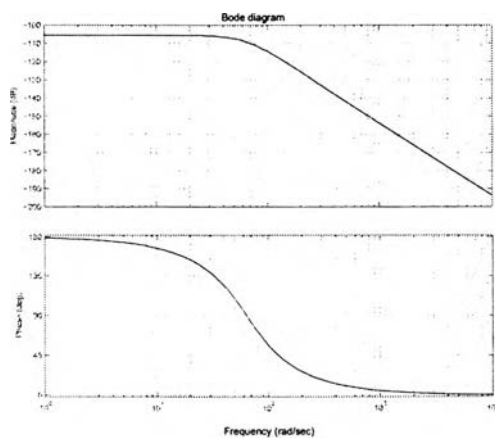
(a) Response from α.

(b) Response from β_1.

(c) Response from β_2.

Figure 4.36: Closed-loop frequency response.

4.3.2 Identification

The web page of belt conveyor identification is shown in Fig. 4.39. In this experiments, we also have three inputs that can be generated to obtain the data, they are prbs, sinusoide and step signal. The length of data can be decided from the length of experiments since for this plant we use sample time 0.01ms. Also for this simulation experiment, we use plant with load in closed-loop. The load used is modeled as

$$\frac{2s + 6}{s^2 + 20s + 125} \tag{4.9}$$

An example result of identification that already done using prbs input is shown in the following explanation. First we have to obatin the pair of input and output data using
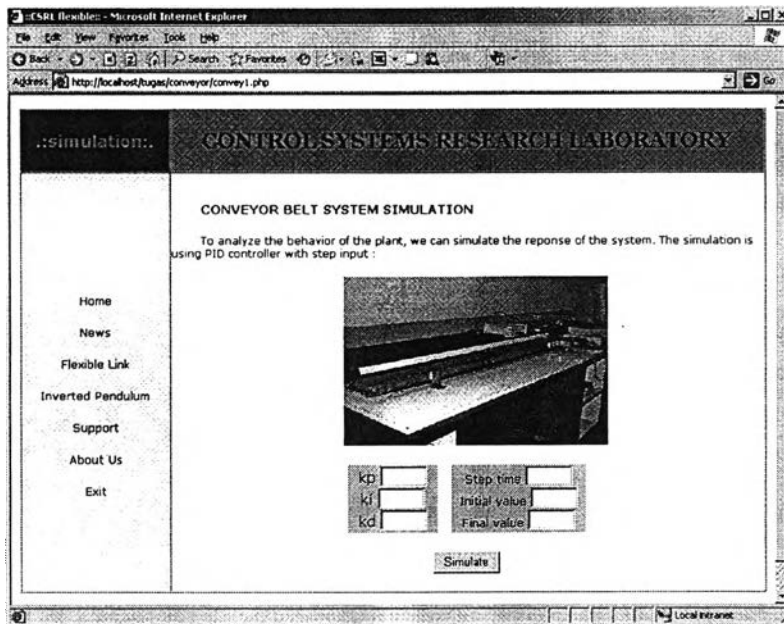
Figure 4.37: Web page for belt conveyor system analysis.
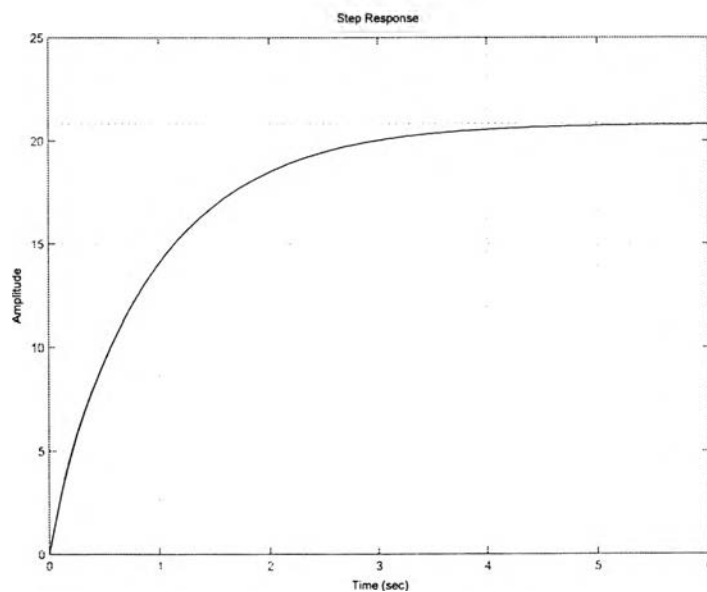

Figure 4.38: Step response of belt conveyor.

interface on the web page. We use prbs to be examined. The direct figure once the data was obtained is shown by Fig. 4.40

The result of identification using parametric method is shown in Table 4.1 and the comparison between obtained model are shown in Fig 4.41.
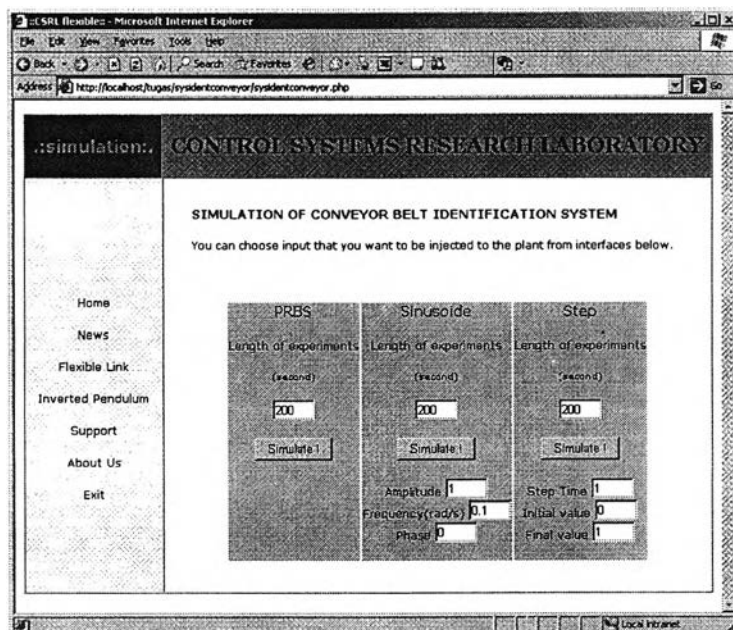
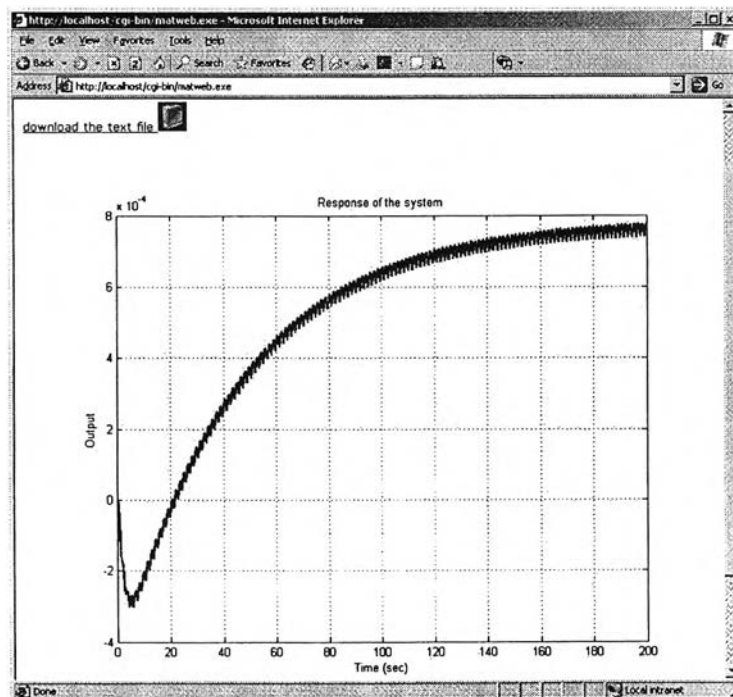Figure 4.39: Web page for belt conveyor identification.

Figure 4.40: Simulation result of belt conveyor identification using prbs input.

Table 4.1: Identification result of belt conveyor using parametric method.

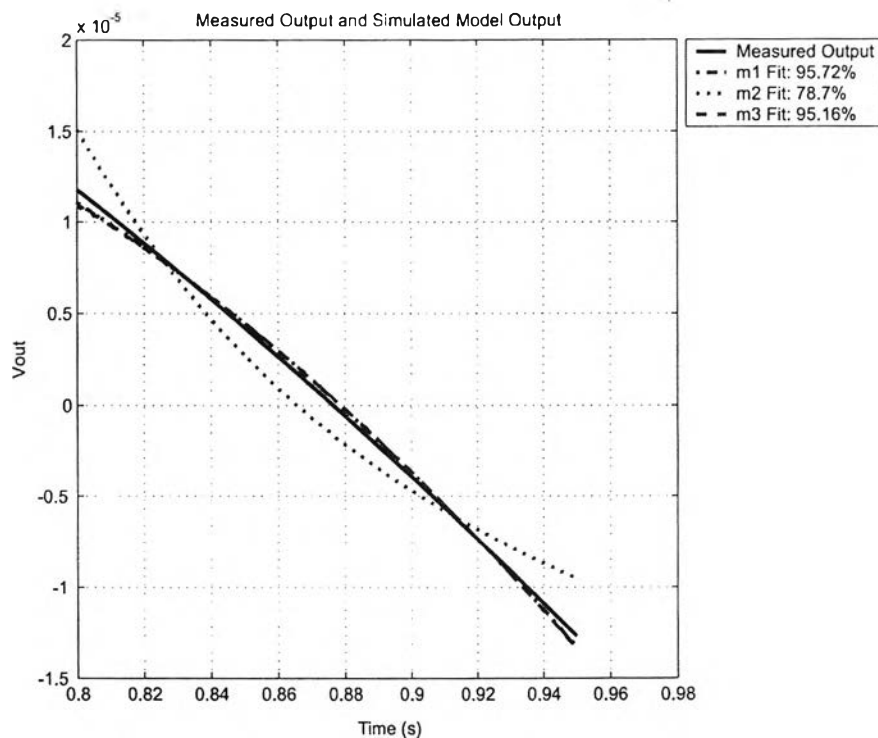| Model type	Model
ARX	$y(t) = \frac{1.41 \times 10^{-10} q^{-1}}{1 - 1.99 q^{-1} + 0.9902 q^{-2}} u(t) + \frac{1}{1 - 1.99 q^{-1} + 0.9902 q^{-2}} e(t)$
State Space (n4sid)	$y(t) = \frac{(-0.156s + 0.162) \times 10^{-8}}{1 s^2 - 2s + 1}$
ARMAX	$y(t) = \frac{-3.353 \times 10^{-10} q^{-1} + 7.224 \times 10^{-11} q^{-2} + 3.298 \times 10^{-10} q^{-3}}{1 - 1.154 q^{-1} - 0.6999 q^{-2} + 0.8536 q^{-3}} u(t)$ $+ \frac{1 - 0.7327 q^{-1} - 0.7115 q^{-2} + 0.526 q^{-3}}{1 - 1.154 q^{-1} - 0.6999 q^{-2} + 0.8536 q^{-3}} e(t)$

Figure 4.41: Comparison between state space(m1), ARX(m2), and ARMAX(m3) model.

4.3.3 Controller Design

The control design experiment in belt-conveyor plant based on the transfer function representation. Controller template that was made is prepared for PID predefined controller design. The procedure of experiment is as shown is section 4.1.3.

For example a quick response to the reference is the main criteria of the performance we want to achieve. An example of combination here use $K_P = 12$, $K_I = 0.49$ and $K_D = 0.5$. The result of the simulation from the webpage and MATLAB is plotted as shown in the Fig. 4.43 and Fig. 4.44.
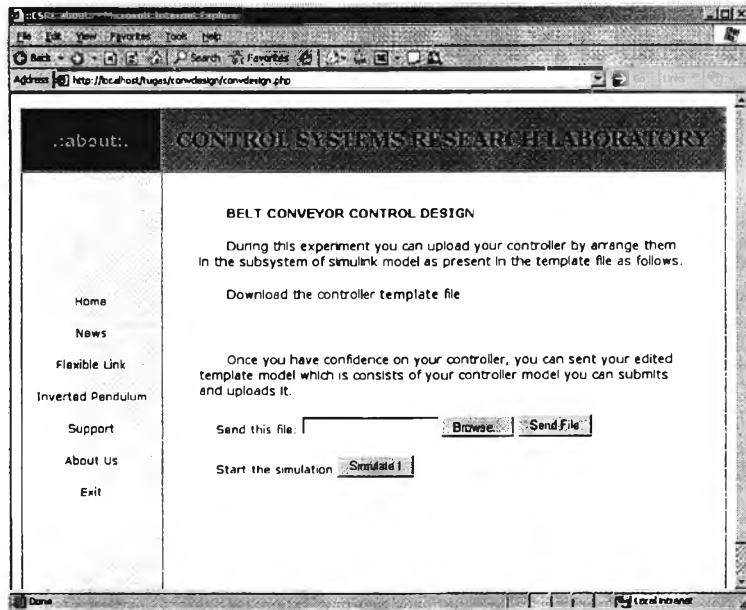
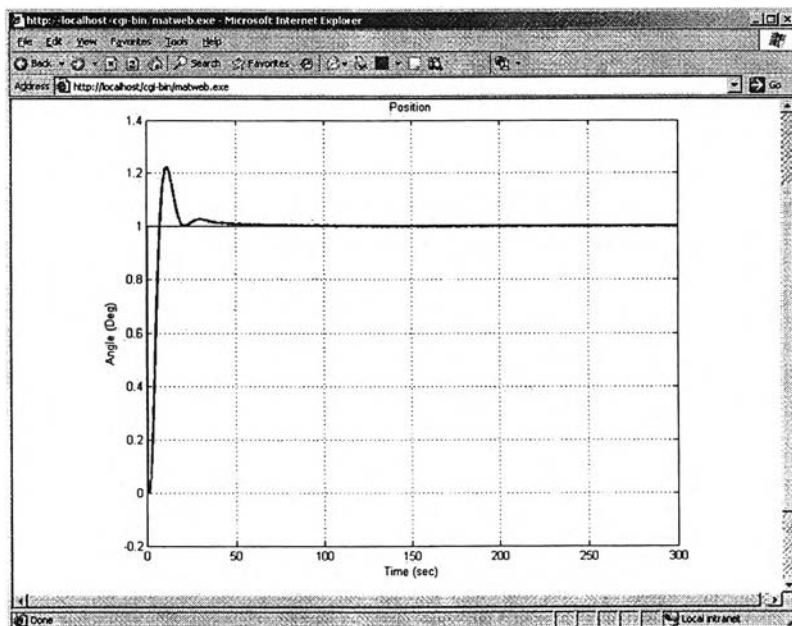Figure 4.42: Web page for belt conveyor control design.

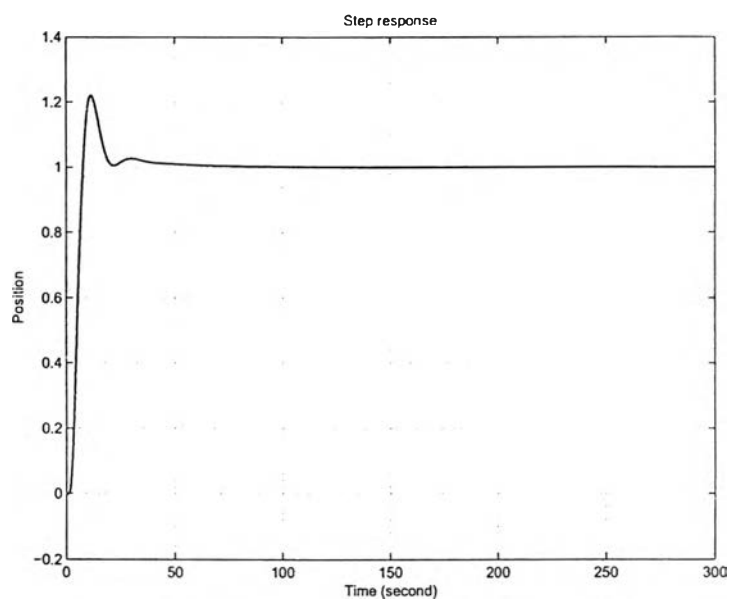Figure 4.43: Web page presentation of PID controller simulation.

Figure 4.44: PID controller simulation using MATLAB.