“I hereby declare that I have read through this report entitle “Analysis of Performance for Cart Gantry Crane System using Scheduling Algorithm via Controller Area Network (CAN)” and found that it has comply the partial fulfillment for awarding the degree of Bachelor of Electrical Engineering (Control, Instrumentation and Automation)”

Signature : ……………………………………………

Supervisor’s Name : En. Hazriq Izzuan Bin Jaafar

Date : 23th June 2016

ANALYSIS OF PERFORMANCE FOR CART GANTRY CRANE SYSTEM USING SCHEDULING ALGORITHM VIA CONTROLLER AREA NETWORK (CAN)

MOHAMAD ROSZULFADLI BIN MOHAMAD @ MUDA

This Report Is Submitted In Partial Fullfillment Of Requirements For The Bachelor Of Electrical Engineering (Control, Instrumentation, and Automation)

Fakulti of Electrical Engineering
UNIVERSTI TEKNIKAL MALAYSIA MELAKA

2016

I declare that this report entitle "Analysis of Performance for Cart Gantry Crane System using Scheduling Algorithm via Controller Area Network (CAN)" is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : Mohamad Roszulfadli Bin Mohamad @ Muda

Date : 23th June 2016

ACKNOWLEDGEMENT

Firstly, I would like to express my gratitude and appreciation to all those who gave me the possibility to complete this report. Deepest thanks to my supervisor, Mr. Hazriq Izzuan Bin Jaafar for his invaluable guidance, stimulating suggestions and encouragement to complete this project. During under his supervision, lot of valuable information is obtained in order to coordinate and complete this project and thesis in the time given.

Special thanks and higher appreciation to my parents, family and special mate of mine for their cooperation, constructive suggestion and also supports from the beginning until the ends during the period of the project. Also thanks to all my friends and others, that has been contributed by supporting and helps myself during the final year project progress till it is fully completed.

Last but not least, greater appreciation to BEKC classmates and Electrical Engineering Faculty UTeM for great commitment and cooperation during my Final Year Project. Besides, I would like to appreciate the guidance given by the panels especially in our project presentation that has improved our presentation skills by their comment and tips.

ABSTRACT

Gantry Crane System (GCS) is one of the huge machinery that carrying a heavy load from one place to another. If the load could not handle properly, it can be effected to the workers at the working area. In order to prevent this issue, the Deadline Monotonic Priority Assignment (DMPA) and the Earliest Deadline First (EDF) are implemented for this GCS. In addition, the presence of Controller Area Network (CAN) is also implemented for the model for Multi-Cart GCS analysis. The analysis is based on the Settling Time (Ts) and Overshoot (%OS). PID Controller is used to the implementation of this project that serves as a basic requirement of control system. All conducive to the implementation of the simulation results and the impact of input-output is the same environment that will be implemented in MATLAB. Based on the results, the implementation of DMPA and EDF scheduling algorithm via CAN is successfully analyzed whereby the execution time could not exceed to the period time to make ensure the GCS in safe condition.

ABSTRAK

Sistem Gantri Kren (SGK) adalah salah satu daripada jentera besar yang membawa beban berat dari satu tempat ke tempat lain. Jika beban tidak dapat dikendalikan dengan baik, ia boleh menjadi kesan kepada pekerja di kawasan kerja. Untuk mengelakkan masalah ini, *Deadline Monotonic Priority Assignment (DMPA)* dan *Earliest Deadline First (EDF)* dilaksanakan untuk SGK ini. Di samping itu, kehadiran *Controller Area Network (CAN)* juga dilaksanakan untuk model. Analisis ini adalah berdasarkan kepada Masa Penyelesaian dan Masa Lonjakan. *PID Controller* digunakan untuk pelaksanaan projek ini yang berfungsi sebagai keperluan asas sistem kawalan. Semua kondusif untuk pelaksanaan keputusan simulasi dan kesan masukkan-keluaran adalah persekitaran yang sama yang akan dilaksanakan dalam MATLAB. Berdasarkan keputusan, pelaksanaan DMPA dan EDF algoritma penjadualan melalui CAN berjaya dianalisis di mana masa pelaksanaan tidak boleh melebihi tempoh sistem untuk memastikan SGK dalam keadaan selamat.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABRREVIOTIONS

GCS	**-**	Gantry Crane System
DMPA	**-**	Deadline Monotonic Priority Assignment
PID	**-**	Proportional, Integrator and Derivative
EDF	**-**	Earliest Deadline First
CAN	**-**	Controller Area Network

CHAPTER 1

INTRODUCTION

This chapter will explain descriptions of Gantry Crane System (GCS) that will be discussed to provide knowledge about the project. The objectives, statements of issues, scope and project outlines for the project as a whole is clearly stated in this chapter.

1.1 Gantry Crane System

Gantry Crane is a vehicle that is also known as cart. The cart used to carry loads from one place to another place in time. Gantry Crane Systems (GCS) need a good control system because GCS heavily involved in the work environment and must be controlled precisely to avoid any unwanted accidents in [1]. The GCS is the most popular in the form of container gantry cranes. It is used for loading containers used to load and transport hub in the outer container ships. It can be categorized as huge gantry crane "full", capable of carrying the heaviest in the world. The small gantry crane normally carries out the work of lifting loads less severe as the engine to be removed or inserted into the vehicle.

The main purpose of this study was to perform a good system of gantry crane, where it gives merit to the work done on the gantry crane without disruption to the load swing. it also aims to put some gantry crane on a common platform. Figure 1.1 shows one example of a gantry crane used in the industry in Malaysia.

Figure 1.1 : Example of a Gantry Crane System

1.2 Objectives

This study embarks on the following objectives :

1. To design TRUETIME simulation environment for analyzing Single Cart Gantry Crane System (GCS) via Deadline Monotonic Priority Assignment (DMPA) and Earliest Deadline First (EDF) scheduling techniques.

2. To analyze the DMPA and EDF for three model of Cart GCS running concurrently in terms of transient response performance.

iii. To evaluate the DMPA and EDF scheduling techniques by using Controller Area Network (CAN) on the global schedule performances.

1.3 Motivation

In the work done by the GCS is primarily concerned with safety. This is because the work performed by GCS very dangerous to the environment and can kill workers who were around. Operating conducted by GCS is associated with heavy materials and may have dangerous substances like explosives or other. The main objective of this project is to give attention to the position of the cart at GCS. In addition, this project will to the favorable impact on the speed of GCS without much effect on load swing.

1.4 Problem Statement

Gantry crane is a system for carrying loads from one place to another. Gantry cranes facilitate the work carrying heavy loads. Gantry cranes have to pay for the speed of its tasks to save time without giving too much power to the load swing. One had accidents is human error, which operates in the state of GCS manual. Furthermore, the percentage of accidents is high if the load is bigger and heavier. Ranking trolley must immediately follow the requirements of the system to stop the trolley in the current move. Scheduling methods are very important for GCS to control the position and speed of the trolley is good and in accordance with the requirements of the system.

1.5 Scopes

The scope on this project, state as below:

I. Analyze the system of Deadline Monotonic Priority Assignment (DMPA) and Earliest Deadline First (EDF) in Single Trolley Gantry Crane System (GCS).
II. The scheduling algorithm and simulation purpose via MATLAB environment.
III. Implement the Controller Area Network (CAN) into the system to analyze the DMPA and EDF scheduling algorithm.

1.6 Report outlines

There are report outlines, as below:

Chapter 1 is the introduction to the project. The objectively, problem statement and scope of the project is obviously in this report.

Chapter 2 is the study of GCS through some paper work that has been done by the researchers of this wipeout, scheduling algorithms, and network control method.

Chapter 3 is about the methodology of the entire project, which will tell you the steps from the beginning of the project until the end of the project. Additionally, it will tell you the materials used during this project, such as Matlab Software will be used to perform simulations on the project.

Chapter 4 is showing the results of the implementation of the project. As a result of the implementation of this project divides into three parts, of which the first part is a result of the Single Trolley GCS to implement scheduling algorithms. Part two is the Multi-Cart GCS and the top three were Cart GCS with scheduling algorithms in the Controller Area Network (CAN). The outcome this time, Single Trolley GCS only executed.

Chapter 5 is the end of the evaluation and discussion of the overall project for future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, it discusses previously associated in this research. The method used in this research scheduling algorithm through the Control Area Network (CAN) and some other methods used in the implementation of GCS. The main purpose of this chapter is to understand and delve more deeply into the methods used.

2.2 Control System of GCS

In 2012, researchers have used genetic Fuzzy System to control the gantry crane [2]. This method is a form of gain scheduling control system. This technique is appropriate and broad based multi-parameter controller by exogenous variables. In addition, this technique is aimed at finding the minimum linear scheduled set by the fuzzy controller interpolation schema. But the researchers are just focusing sheets y axis on which anti-swing crane. The GCS has a range of effects to make the work go smoothly and safely.

In 2011, a researcher has focused on the effects of heavy load on the system response GCS. In his research [3], he uses feedback control schemes for GCS. To achieve this result were evaluated with different weights in algorithm. Three different control strategies used by researchers, namely LQR, PD controller for DFS and simulation exercises in Matlab, and the results were compared with the uncontrolled system. Many researchers have used the term open loop optimal strategy as discussed in [4,5]. However, they come out with results that are not good for open-loop strategy because it is sensitive to parameters (eg rope long) and cannot recompense for wind noise. Singhose [6] introduced the importance of open-loop strategy is the input form. However input form the method is still open loop approach. But others with feedback control, it is less sensitive to noise and parameter variation [7] is also used to control the GCS.

2.3 Deadline Monotonic Priority Assignment (DMPA)

DMPA is the inverse of the deadline and it is similar in concept to rate monotonic priority. Where priority is given to the process is inversely proportional to the length of the deadline. Short deadlines or quickly given the highest priority and due date of the oldest of the lowest priority. The patient is the priority order when the next rate monotonic order to match the deadline (period = the deadline). DMPA is a static priority scheme optimal for fast sharing of critical process. This was stated as Theorem 2.4 in [8] “Reverse-deadline optimum priority assignment for the processors”. In [9] states that any set of processes whose characteristics appropriate time to rate monotonic analysis will also be accepted by virtue of static theory that justifies the deadline and within different processes.

Generally, where a deadline-monotonic scheme who are not working due to lack of sufficient schedulability test. The rate-monotonic scheduling schedulability test can be used to reduce the length of the individual so that the same process with deadlines. This situation obviously test it will not be optimal because the workload on the processor will exceed the estimate. The priority assignment is optimal for a set of tasks that are given priority according to their deadline, the task with a short deadline given the highest priority. Optimal assignment policy priorities for a set of tasks on a regular or occasional strictly compliant system model like, all tasks have deadlines of less than or equal to the minimum their time between the arrival (or periods) and all tasks have the worst-case execution times (WCET) of less than or equal to their end date. All tasks are independent and so do not prevent their implementation. This is because there is a duty voluntarily suspends itself. There is a point in time, the so-called critical instant, where all tasks to be ready to execute simultaneously and when scheduling overhead is zero. Over scheduling is intended to switch from one task to another. All tasks have zero emissions jitter (time of assignment arrived for it to be ready to implement).

Besides that, for the example in [10] if the restriction is lifted 7, then "deadline less jitter" monotonic priority assignment is optimal. If the restrictions lifted to allow the deadline to be greater than the last, then Audsley optimum priority assignment algorithm can be used to find the optimum priority assignment.

2.4 Earliest Deadline First (EDF)

The Earliest Deadline First (EDF) is a dynamic scheduling algorithm used in real-time operating system to put a process in the priority queue. The priority queue is an abstract data type as a stack data is unusual, but in which every element has an additional "priority" associated with it. In the priority queue, an element with priority served before the element with low priority. If two elements have the same priority, they are served according to their order in the queue. Although priority queue are often implemented with a stack, they are conceptually different from the stack.

The priority queue is an abstract concept such as “list” or “map”. It is only as a list can be implemented with a linked list or array. In addition, the priority queue can be implemented with a stack or various other method such as multiple unordered. Whenever an event occurs such as a completed task scheduling, the line will be searched for the closest to the deadline. This process is next scheduled for execution. EDF is an optimal scheduling algorithm in uniprocessors advance. It means, if the collection of independent and each has features that moment arrived, the implementation requirements and deadlines, it can be scheduled with any algorithm in a way that ensures all work completed by their deadlines. EDF will schedule the collection so that they all complete the work according to their deadline. EDF can guarantee all the deadlines met provided that the total CPU usage is more than 100%. EDF can ensure that all deadlines in the system of higher loading compared with fixed priority scheduling techniques such as scheduling classes monotonic [11].

In [12], the researchers have used the EDF for automotive applications. Generally, automotive applications are cyber ordinary physical systems, where it performs real-time processing of continuous data using a variety of sensors and communication on board from outside the vehicle. However, transmission of data outside-the-vehicles often has problems when introducing data rate fluctuations are large, where the arrival time can vary or not be guaranteed. The researchers have determined the flow and have been using EDF Scheduling. These techniques can be used not only EDF-PStream, but also for general data processing flow-based EDF. EDF-Tstream is a method that uses rescheduling by EDF preemtable data stream. The design method preemptable data flow task by their characteristics, and the researchers confirmed the effectiveness of this method is based on analysis and experiments [12]. It showed improved efficacy in a real-time implementation constraints, reduction of vehicle accidents, and vehicle positional accuracy, compared to the scheduling method based on data flow. existing real-time scheduling cannot handle the data flow out of order queuing, and search out-of-order queue by EDF performance degrades as frequently accessing queues. Weighted Fair Queueing (WFQ) [13] and the earliest Deadline First (EDF) [14] have been extensively studied in recent years. WFQ and EDF using dynamic priority mechanisms in the packet and both can provide end-to-end delay limits for regulated traffic flow. In a packet network, the scheduling is an important mechanism for realizing the Quality-of-Service (QoS) as it directly controls the packet delay. WFQ has good properties for the protection of traffic, while EDF is known to be the optimum delay in providing limits on single node [15]. If in the case of end-to-end EDF overcome if traffic WFQ each node to form implemented [16].

2.5 Control Area Network (CAN)

A Controller Area Network (CAN) is a vehicle bus standard designed to allow microcontrollers. CAN is also for the tools to communicate with each other in the application without a host computer. It is a message-based protocol, originally designed for a multiplex electrical wiring in cars, but also used in many other contexts. Car accidents or vehicle on the road is often the case. One of the reasons is because drivers are often sleepy when driving. Therefore, in [17] the researchers have been using CAN as control Advanced RISC Machines (ARM). The system is safe for drivers who often sleepy when driving. Both ARM controller, Master and Slave are connected to the CAN bus protocol for exchanging information and for commu-nication. The CAN be used for faster and reliable Communi-cations.

In a study [18], a system developed using ARM controller as the main control unit and CAN bus in a car. ARM is used to get high performance. CAN makes use of high-speed communication in control networks. Besides, it also helps the sharing of data between all nodes resulting increase their collaborative work. Vehicle reliability is largely influenced by the complexity of the circuits used in control systems [19, 20] in an increasing number of electronic controllers and instruments in the modern automotive industry. Maintenance is difficult to carry out and from the point of view of the layout, electrical systems using point-topoint traditional single communication approach, which inevitably will lead to the big problem of cable pets. Thus, CAN is used as a high quality vehicle for the bus system to connect all the controllers in the system to achieve unified management [21]. The CAN lead to easier data sharing and interoperability between different control systems. Due to the complexity of the vehicle, for example, the entire vehicle with a sensor assigned to various standard and automotive systems is the data in various formats such as complex data, heterogeneous data etc. [22].

Vehicle systems need information to the maintainer and driver. It is necessary to design an efficient, reliable gateway and data processing systems. The gap between different systems is difficult to meet is a concoction. This problem can be solved by the entrance bridge to connect multiple CAN buses with different speed ratio.

2.6 Conclusion

There are many systems that use techniques and CAN scheduling algorithm implemented by the researchers of this wipeout. Similarly the DMPA and EDF, where researchers study the past, have used it to perform in their system. According to the paper work done by the researchers, the technique uses an algorithm scheduling has advantages and disadvantages. However, it also depends on agreement on a system to be implemented.

CHAPTER 3

METHODOLOGY

This chapter will discuss the methods that were used to complete this project. Gantry Crane System (GCS) will be used in this project as a plant. GCS system scheduling algorithm uses both algorithms which DMPA and EDF via CAN.

3.1 Project Planning

Figure 3.1 shows the flowchart for the entire project. The flowchart is shows overall the project for this analysis. The flowchart shows the step by step to make this analysis. Figure 3.2 shows the K-Chart for graphically explain the flow method and requirements use in overall of this project. The blue color boxes represented the method and requirements that will use in this project.

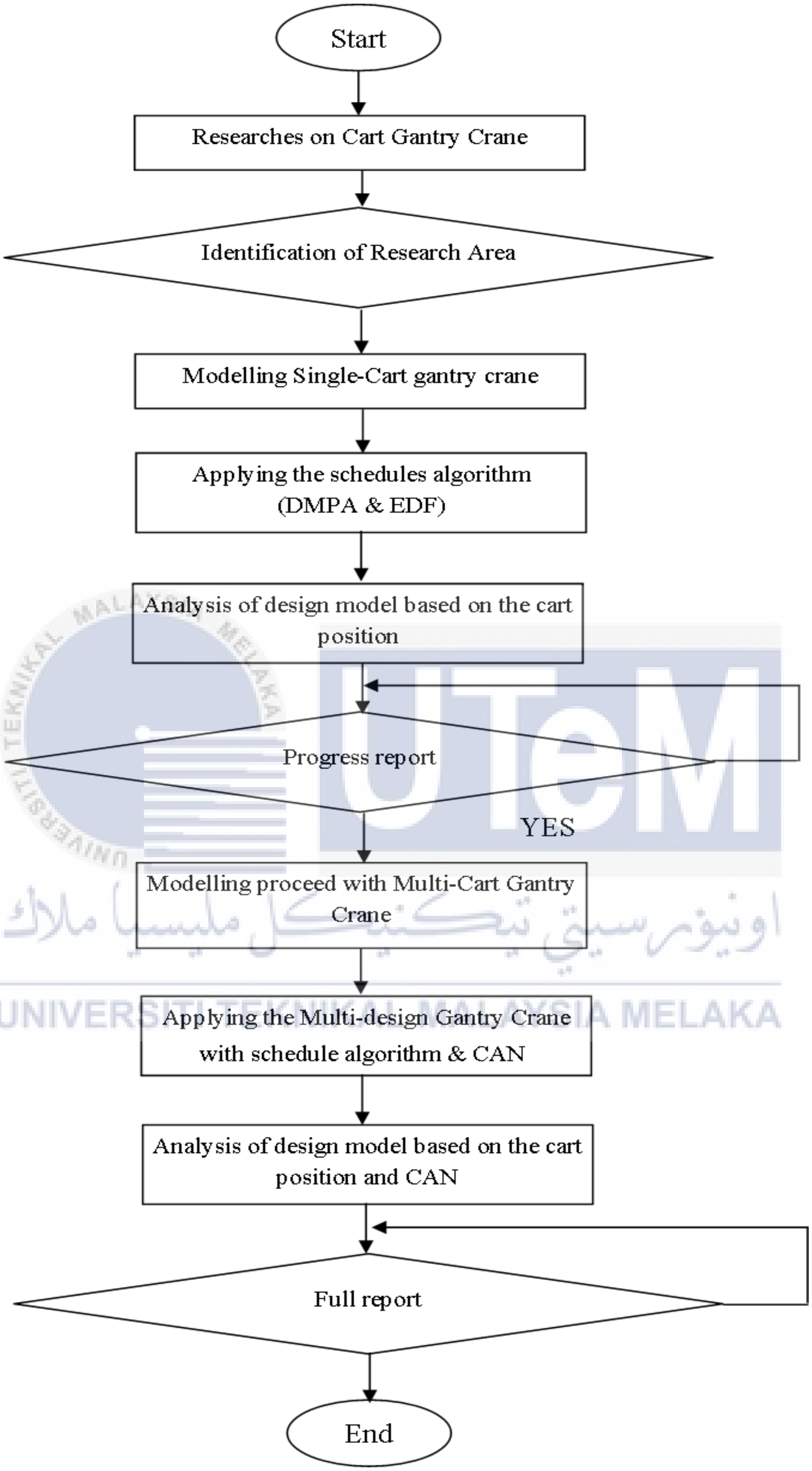


Figure 3.1 : Flowchart of project

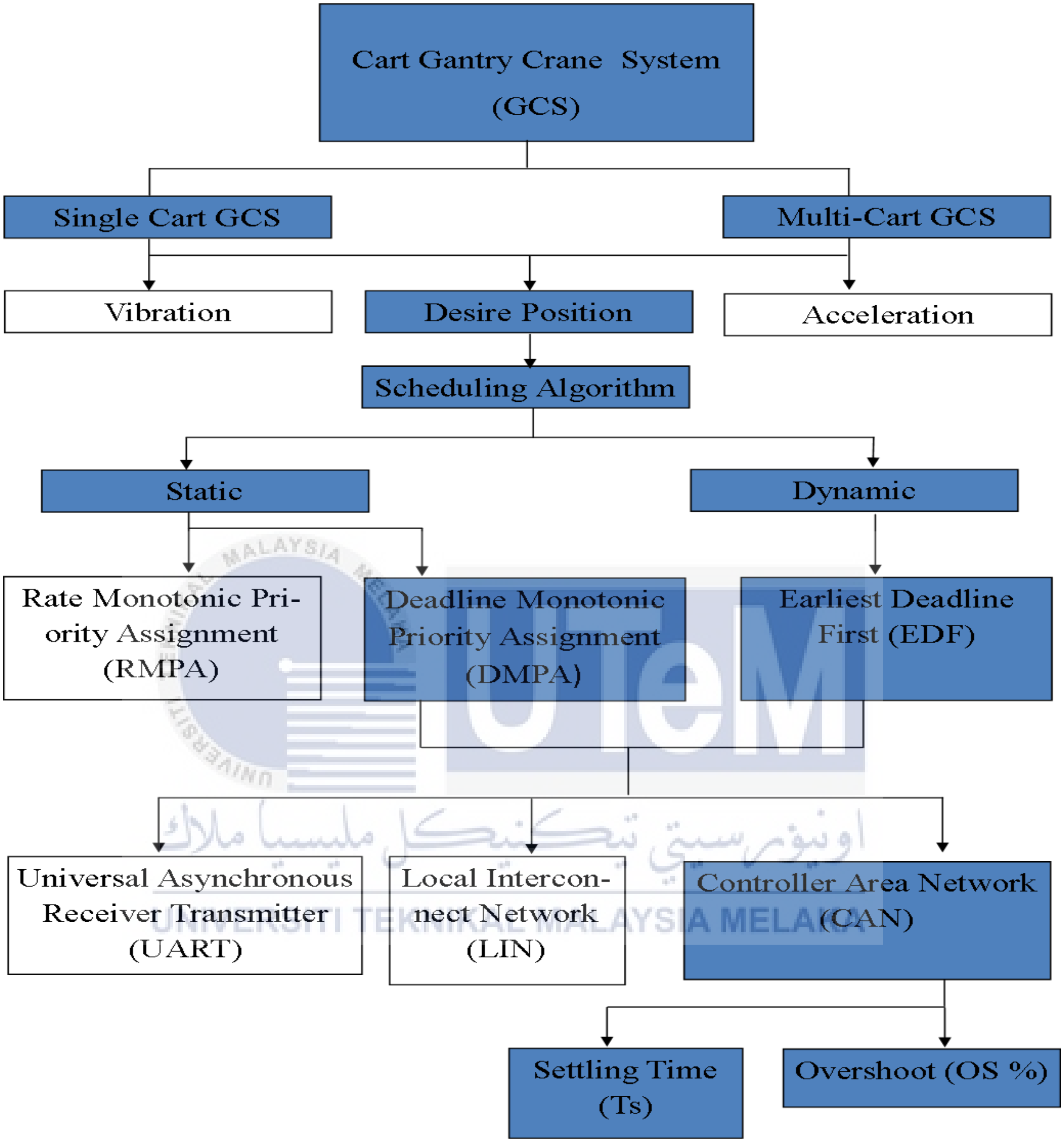


Figure 3.2: K-Chart

3.2 Gantry Crane System Model

Figure 3.3 shows a schematic diagram in which the GCS is a system that will be implemented in this study. GCS parameters are shown as follows:

- Mass of payload = m_1
- Mass of trolley = m_2
- Cable length = l
- Trolley horizontal position = x
- Swing angle = T
- Torque = T
- Each driving force = F

However, in this study, it will not involve research on the state of the charge oscillations and nonlinear models of GCS modeled by Simon et al. [23-24]. In addition, it will assume as a rigid body and without mass to reduce a number of problems in terms of modeling the cable and the cable load. Table 3.1 indicates the values of the other parameters.

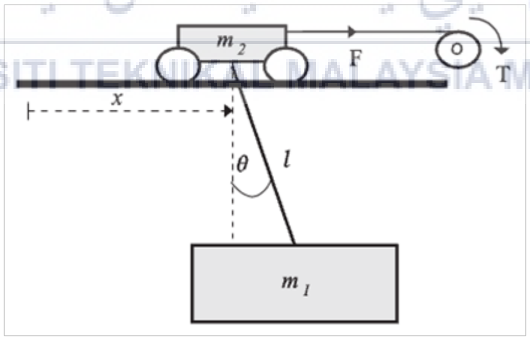


Figure 3.3: Schematic Diagram of a Gantry Crane System [23]

Table 3.1: System Parameter Value [23]

Parameters	**Value**
Payload mass *(m_1)*	1 kg
Trolley mass *(m_2)*	5 kg
Cable length *(l)*	0.75 m
Gravitational *(g)*	9.81 m/s^2
Damping Coefficient *(B)*	12.32 Ns/m
Resistance *(R)*	2.6 Ω
Torque constant *(K_T)*	0.007 Nm/A
Electric constant *(K_E)*	0.007 Vs/rad
Radius of pulley *(r_P)*	0.02 m
Gear ratio *(z)*	15

3.3 Mathematical Modeling

It is important to understand the mathematical model the behavior of GCS. Use mathematical language to describe the system or process. In addition, a mathematical model was designed to optimize system behavior and identify the parameters that optimize system performance.

The differential equation of the GCS can be obtained as :

$$V=\left[\frac{RBr_P}{K_T z}+\frac{K_E z}{r_P}\right]+\left[\frac{Rr_P}{K_T z}\right](m_1 l)\left[\ddot{\theta}\cos\theta-\dot{\theta}^2\sin\theta\right]+\left[\frac{Rr_P}{K_T z}\right](m_1+m_2)\ddot{x} \tag{3.1}$$

$$m_1 l^2\ddot{\theta}+m_1 l\ddot{x}\cos\theta+m_1 gl\sin\theta=0 \tag{3.2}$$

3.4 Simulation of Single Cart GCS

Figure 3.4 shows the simulated environment TRUETIME, it is GCS process where simple PID controller is used as an introduction to the basics. The TRUETIME kernal will control process performed by the controller task. Declarations will be made in MATLAB and TRUETIME to implement the scheduling algorithms. Observation about DMPA and EDF scheduling algorithms will be performed and will analyze the different input-output for the implementation of GCS.

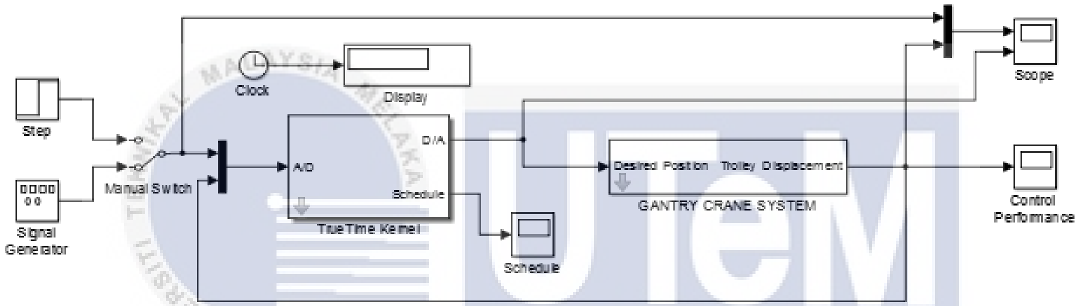


Figure 3.4: Single Cart Gantry Crane System

In TRUETIME kernel of the main block GCS shown in Figure 3.3 has a block as shown in Figure 3.5. The block diagram as shown in Figure 3.4 will be compressed at a subsystem which represents a major block for GCS.

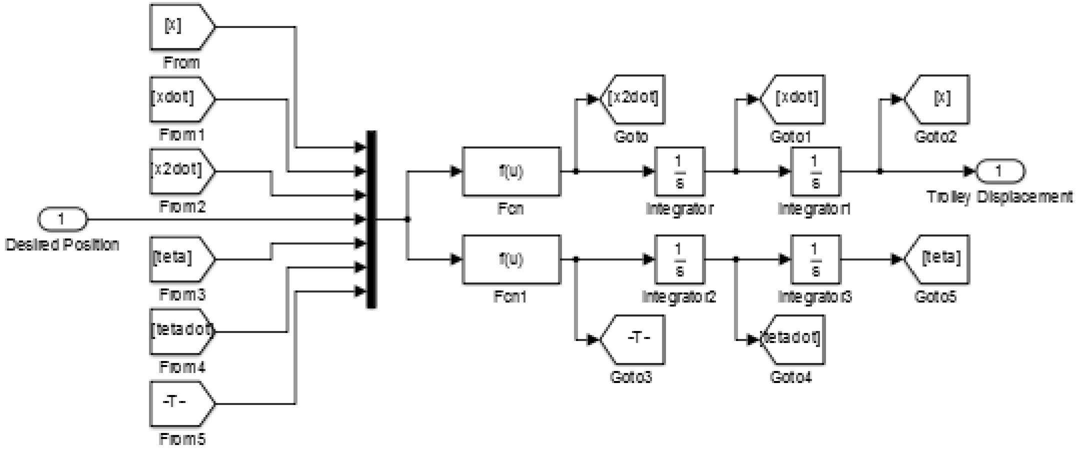
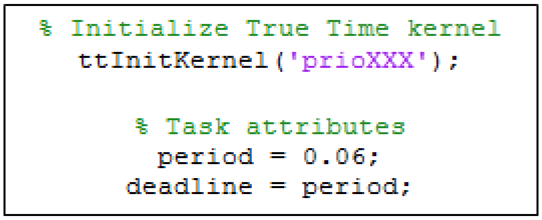


Figure 3.5: Block Diagram inside the Main Block of GCS

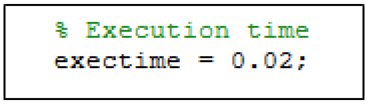
For this analysis, algorithm DMPA and EDF will be implemented. With the implementation of a different time, there were four cases to be simulated in this configuration. Figure 3.6 and Figure 3.7 is an example of the pattern of change for the execution time of the declaration of the state of the task for an initial period of system set as 0.06 s and execution time is set as 0.02 s.



```
% Initialize True Time kernel
ttInitKernel('prioXXX');

% Task attributes
period = 0.06;
deadline = period;
```

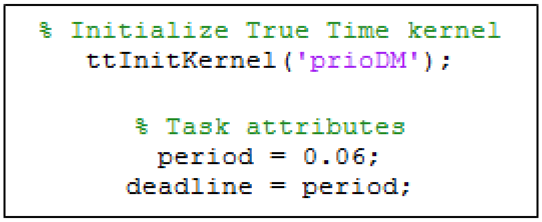
Figure 3.6: Initialize True Time Kernel and Task Attributes



```
% Execution time
exectime = 0.02;
```

Figure 3.7 : Initialize True Time Kernal and Task Attributes fo Execution Time

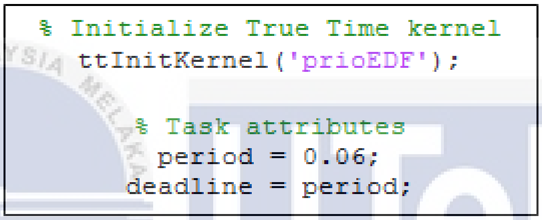
Figure 3.8 show the declaration for DMPA and Figure 3.9 show the declaration for EDF by using MATLAB. It is the priority call for DMPA and EDF. This coding have a related with the GCS model.



```
% Initialize True Time kernel
ttInitKernel('prioDM');

% Task attributes
period = 0.06;
deadline = period;
```

Figure 3.8: DMPA Initialize True Kernel and Task Attributes



```
% Initialize True Time kernel
ttInitKernel('prioEDF');

% Task attributes
period = 0.06;
deadline = period;
```

Figure 3.9: EDF Initialize True Time Kernal and Task Attributes

3.5 Analysis of Multi-Cart GCS Based on DMPA and EDF Algorithm

This analysis uses a simple PID control for controlling three GCS on the same CPU as shown in the Figure 3.10. The three cases of PID are running at the same time to control the movement or the work done by the same three GCS. In this analysis, the scheduling algorithm is the same but the difference is only increasing the number of GCS only. It aims to demonstrate the EDF and DMPA algorithm applied to the GCS is suitable or not when the number of GCS is increased. in this case, where three cart GCS used in a one load only. Usually the number cart GCS increases when the parameters of load is increases, so that the burden of larger and heavier load increases, where the single GCS was cannot able to do alone.

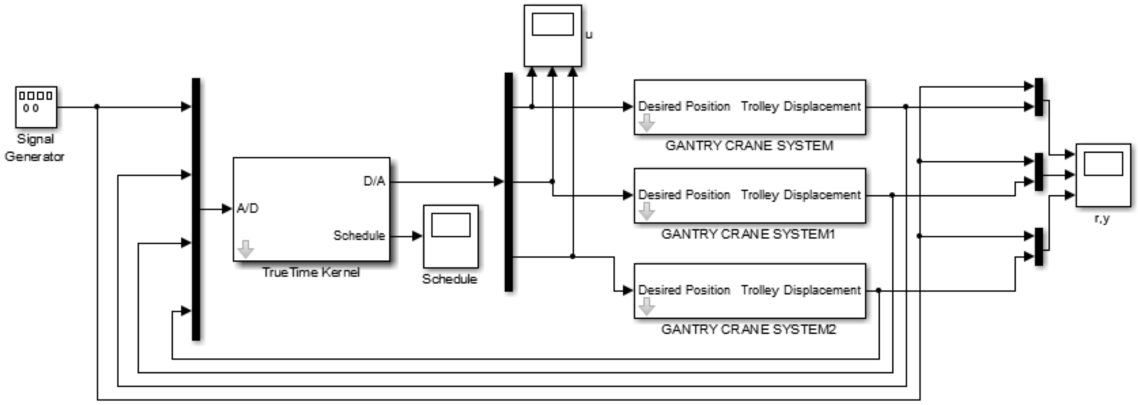


Figure 3.10: Three Gantry Crane Systems (GCS)

3.6 Scheduling Algorithm Via Controller Area Network (CAN)

The Figure 3.10 shown the CAN implement at GCS with DMPA and EDF scheduling algorithm. The CAN implement at GCS with two algorithm when the modeling proceed with Multi-Cart GCS or Multi-Cart GCS. Usually, the CAN is used to control more than one cart GCS in this project.

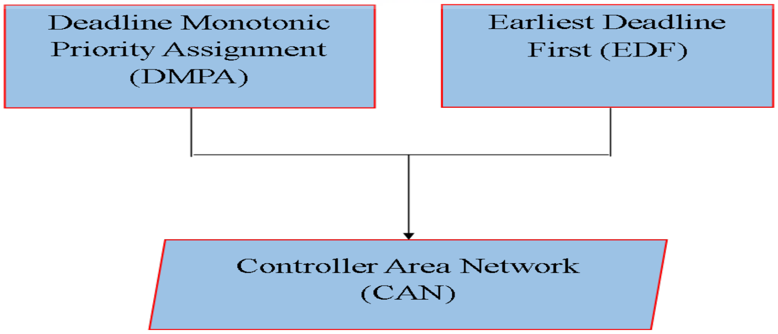


Figure 3.11: Implement DMPA and EDF via CAN

3.6.1 Analysis of GCS Based on DMPA and EDF Algorithm via CAN

Figure 3.12 shows Simulink block diagram GCS with CAN. Two types of algorithms DMPA and EDF will be analyzed in this analysis. To carry out CAN simulation, it can be done using TRUETIME Kernal Block and TRUETIME Network Block is available from True Time Library.

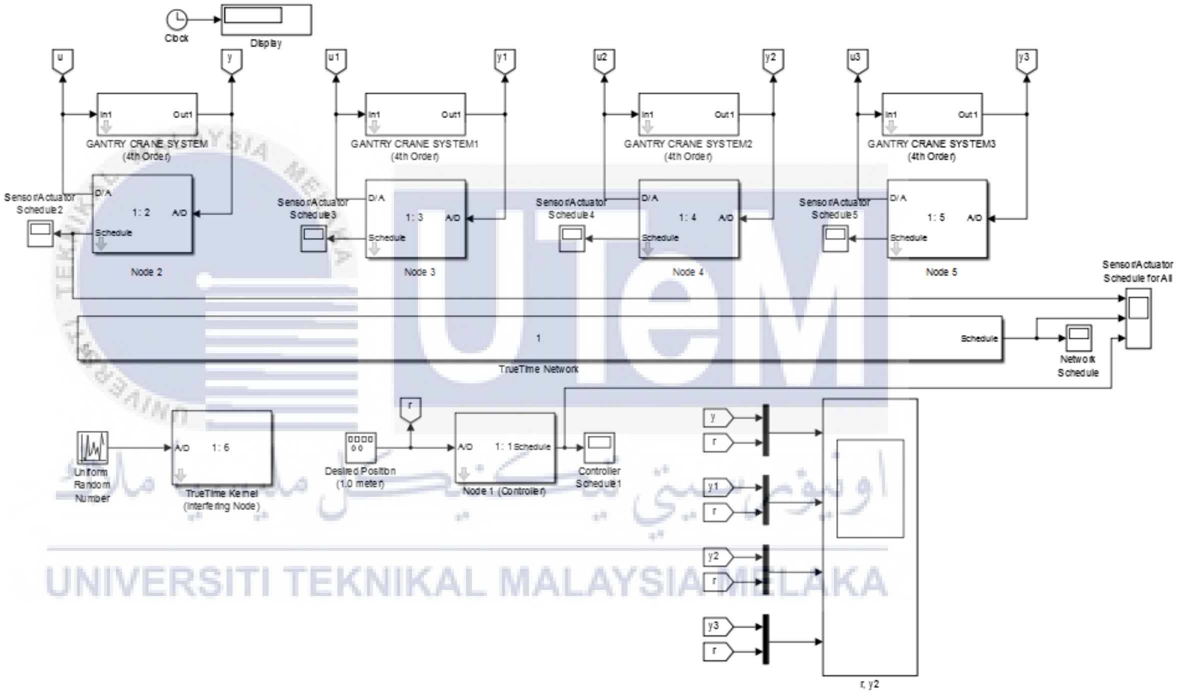
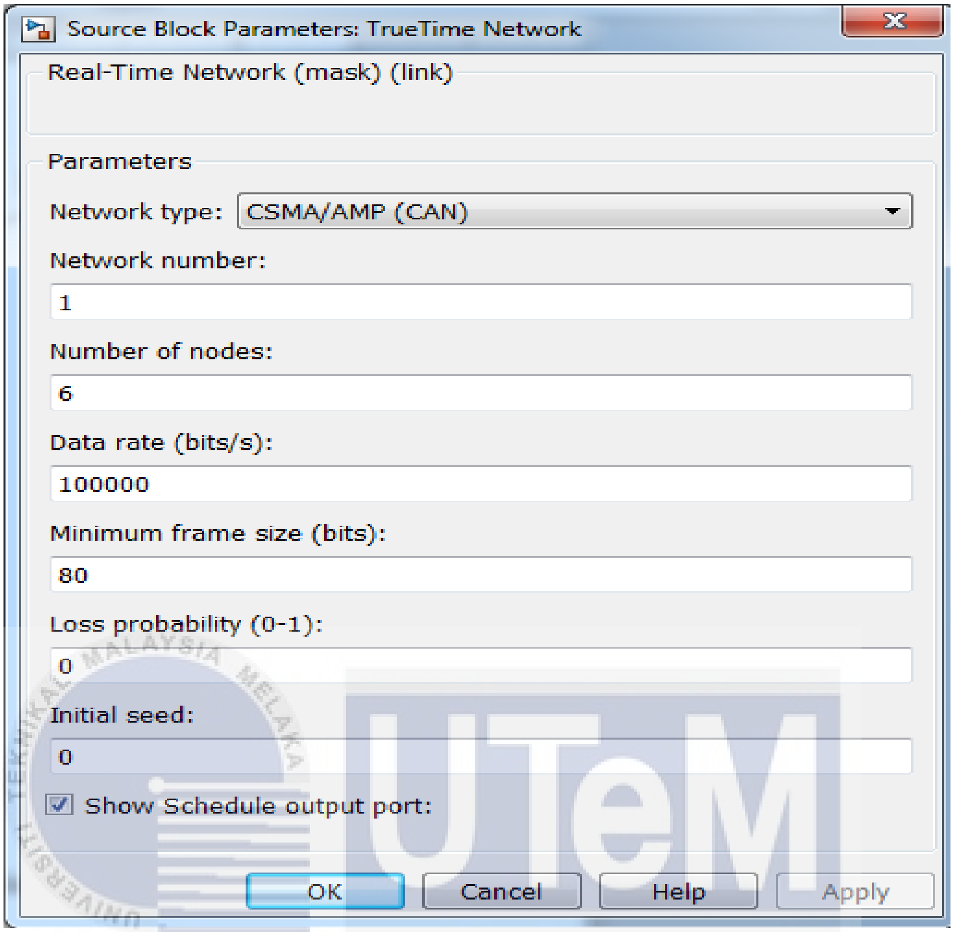


Figure 3.12: Simulink Block Diagram GCS with CAN

Figure 3.13 show the TRUETIME network parameter. The network number is set only 1 because this system just used only one network. The number of nodes is 6, four node for gantry crane, one node for TrueTime Kernal and last one node for controller. The number of node is for defines the number of kernel elements in the network. Data rate is set 100000 bits/s and for minimum frame size is 80 bits. Actually for CAN network, it can be operate with bit rate up to 1M bits/sec.



Real-Time Network (mask) (link)

Parameters

Network type: CSMA/AMP (CAN)

Network number: 1

Number of nodes: 6

Data rate (bits/s): 100000

Minimum frame size (bits): 80

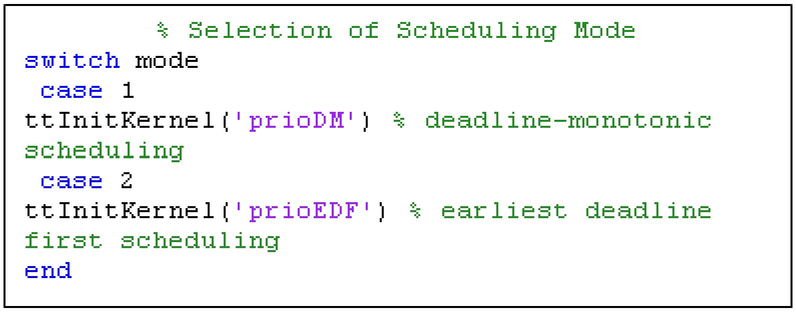
Loss probability (0-1): 0

Initial seed: 0

☑ Show Schedule output port:

Figure 3.13: TRUETIME Network Parameter Configuration

This system used two scheduling algoritm DMPA and EDF via CAN. The block of simulink for this system are same but it need choose only one algorithm with CAN. So the coding to choose the mode for scheduling algoritm is made such as Figure 3.14.



```
        % Selection of Scheduling Mode
switch mode
 case 1
ttInitKernel('prioDM') % deadline-monotonic
scheduling
 case 2
ttInitKernel('prioEDF') % earliest deadline
first scheduling
end
```

Figure 3.14: Selection of Scheduling Mode

3.7 Conclusions

In this chapter show overall the procedure to make this analysis from start to end. The Matlab Software is used to make the simulation for this project. In this project have three part or task, where the first task is Single-Cart GCS. After this task is done, the second task is run for Multi-Cart GCS and the last task for this project is implement the CAN at GCS with DMPA and EDF scheduling algorithm. For task 1 and 2 just only used the DMPA and EDF scheduling algorithm in this GCS.

CHAPTER 4

RESULTS AND DISCUSSIONS

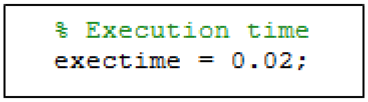
There were four cases of declarations with different implementation periods with fixed implementation time as 0.02 s, 0.12 s, 0.40 s and 0.90 s for both two algorithms. This analysis will discuss the results of the four simulated cases for DMPA and EDF algorithm simulation model.

4.1 Result of Simulation for a Single-Cart GCS

CASE 1 :

The execution time for DMPA and EDF is set as Figure 4.1 and Figure 4.3. The Figure 4.2 is graph for DMPA and the Figure 4.4 is graph for EDF. The both of graph exactly show the same output when the overshoot around 6% and settling time is 3.1 s where the green line is the trolley displacement output system. However, the cart GCS is stable for this case because the system meet the deadline.

DMPA



```
% Execution time
exectime = 0.02;
```

Figure 4.1 : Execution time is 0.02 s

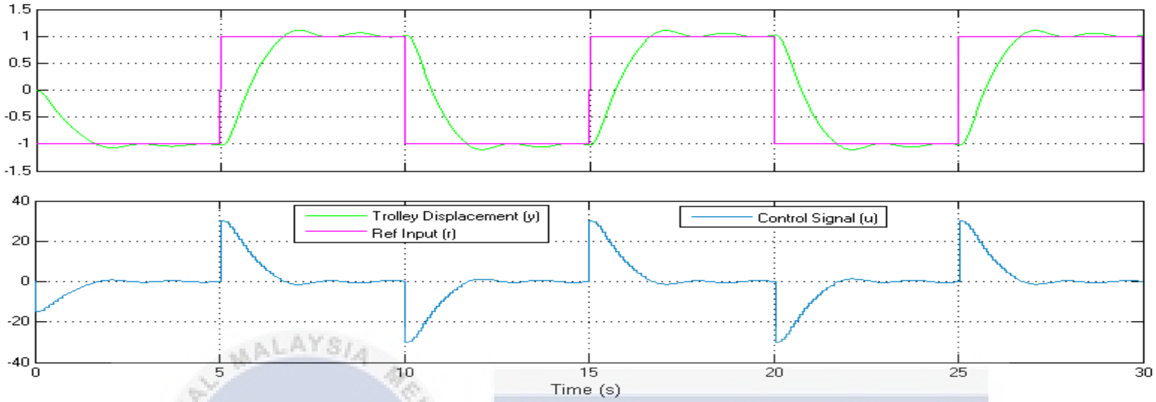
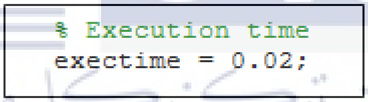


Figure 4.2: Reference Input (r), Trolley Displacement (y) and Control Signal (u)

EDF



```
% Execution time
exectime = 0.02;
```

Figure 4.3 : Execution time is 0.02 s

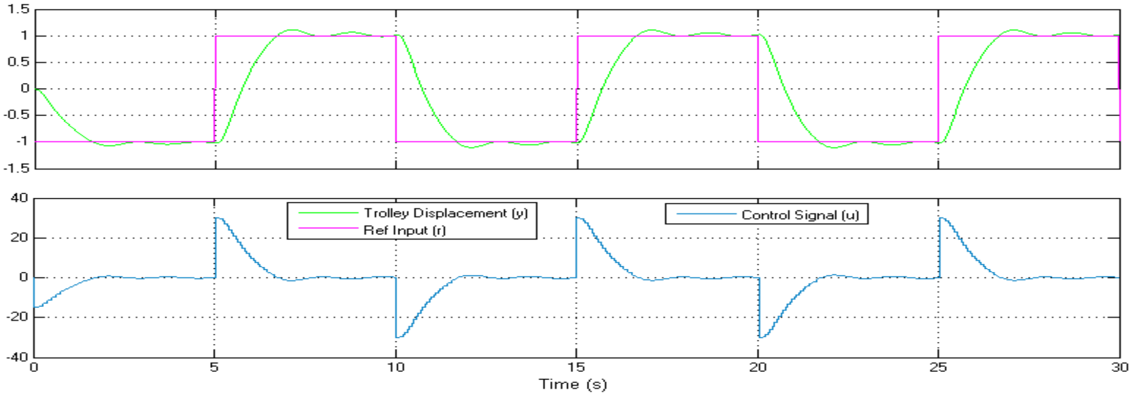
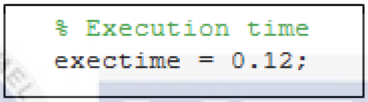


Figure 4.4 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

CASE 2 :

The execution time for DMPA and EDF is set as shown in Figure 4.5 and Figure 4.7. The Figure 4.6 is graph for DMPA and the Figure 4.8 is graph for EDF. From two graph for both algorithm is same pattern. But for this case, the overshoot (23%) is increased and settling time (3.3 s) also increased. However, the performance of cart GCS is still stable. The blue line is the control signal that show it is still smooth even though the overshoot and settling time for both graph of DMPA and EDF are increased.

DMPA



```
% Execution time
exectime = 0.12;
```

Figure 4.5: Execution time is 0.12 s

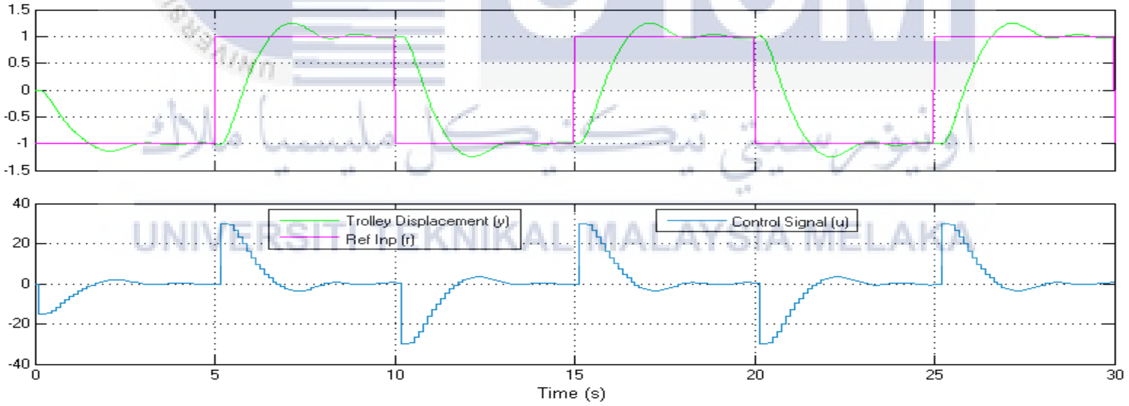
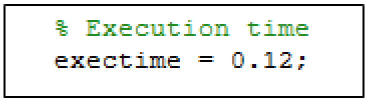


Figure 4.6 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

EDF



```
% Execution time
exectime = 0.12;
```

Figure 4.7 : Execution time is 0.12 s

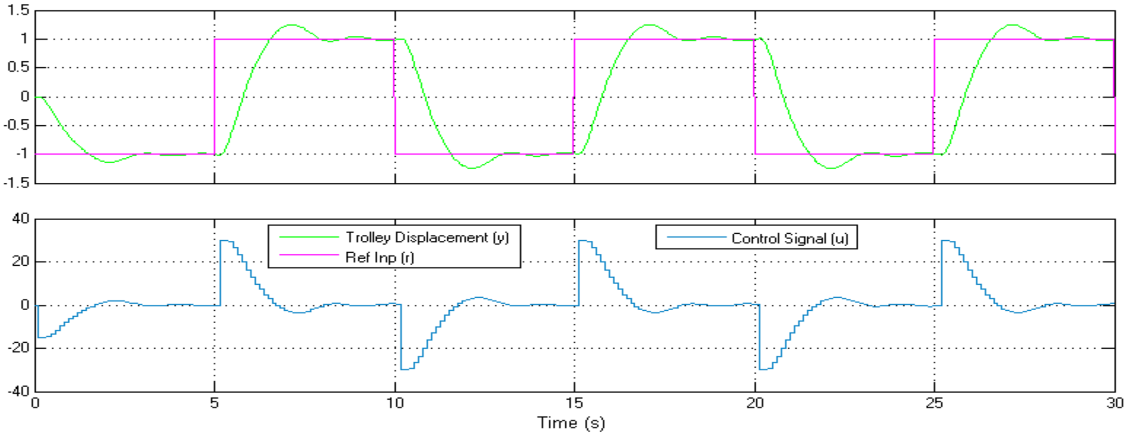
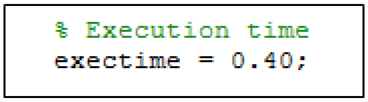


Figure 4.8 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

CASE 3

Figure 4.9 and Figure 4.11 shows the execution time for both algorithm. While the graph at Figure 4.10 for DMPA and Figure 4.12 for EDF. For this case, the both graph also shows the same pattern with DMPA and EDF algorithm. But the way, this case for both algorithm have increased the overshoot (50%) and settling time (>5) compared the previously task. The control signal show the system is toward unstable. So, overall the output of trolley displacement system still acceptable but not as required.

DMPA



```
% Execution time
exectime = 0.40;
```

Figure 4.9 : Execution time is 0.40 s

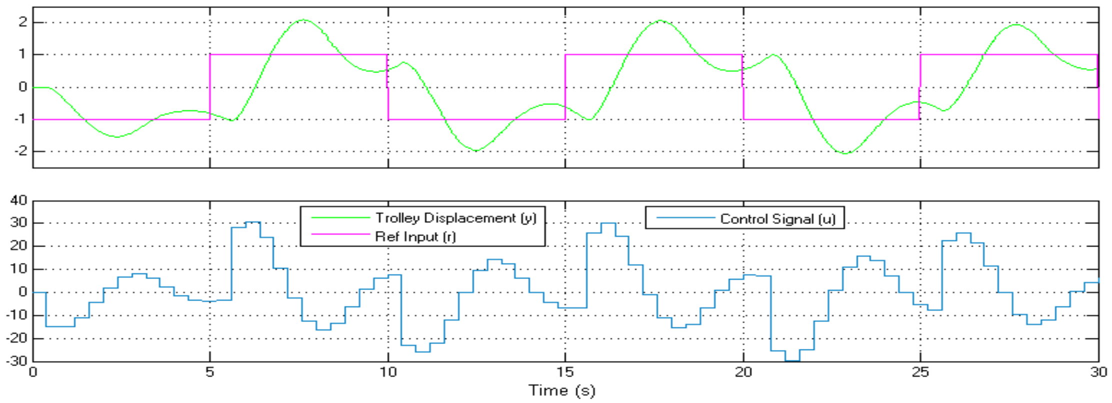
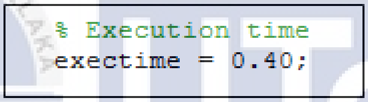


Figure 4.10 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

EDF



```
% Execution time
exectime = 0.40;
```

Figure 4.11 : Execution time is 0.40 s

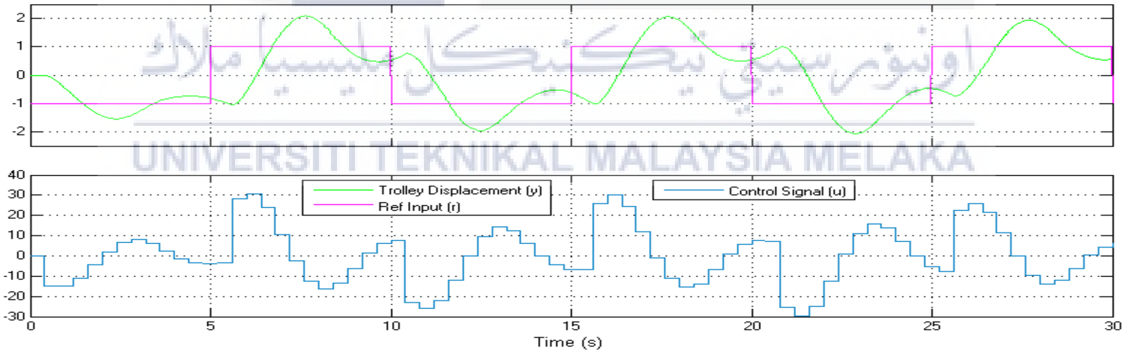
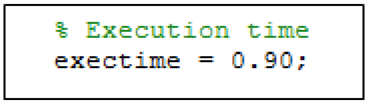


Figure 4.12 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

CASE 4 :

Case 4 the execution time for DMPA and EDF as shown in Figure 4.13 and Figure 4.15. The Figure 4.14 is graph for DMPA and the Figure 4.16 is graph for EDF. For this case the systems are not stable. It is because the overshoot for the output of trolley displacement at green line is not same in each new cycle and the settling time for the output is not stable also. For the control signal at blue line, the oscillation is not same in each new cycle. So, the overall system for this case is unstable and not as required.

DMPA



```
% Execution time
exectime = 0.90;
```

Figure 4.13: Execution time is 0.90 s

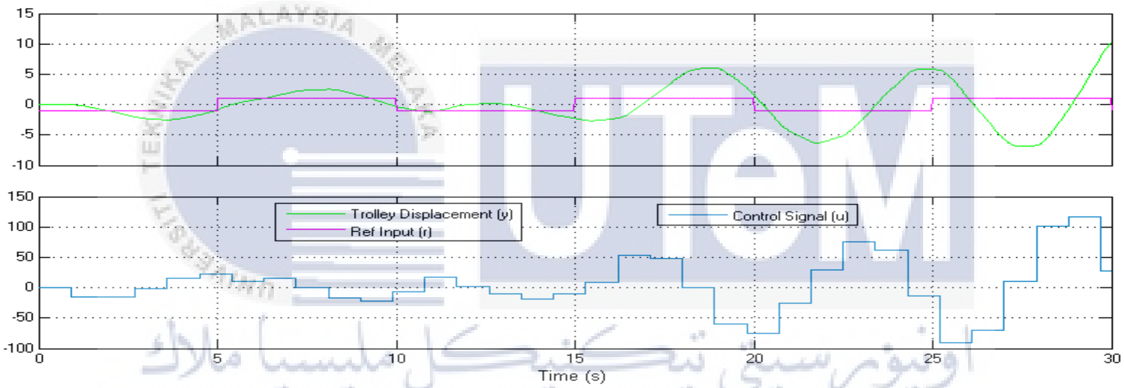
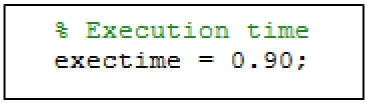


Figure 4.14: Reference Input (r), Trolley Displacement (y) and Control Signal (u)

EDF



```
% Execution time
exectime = 0.90;
```

Figure 4.15 : Execution time is 0.90 s

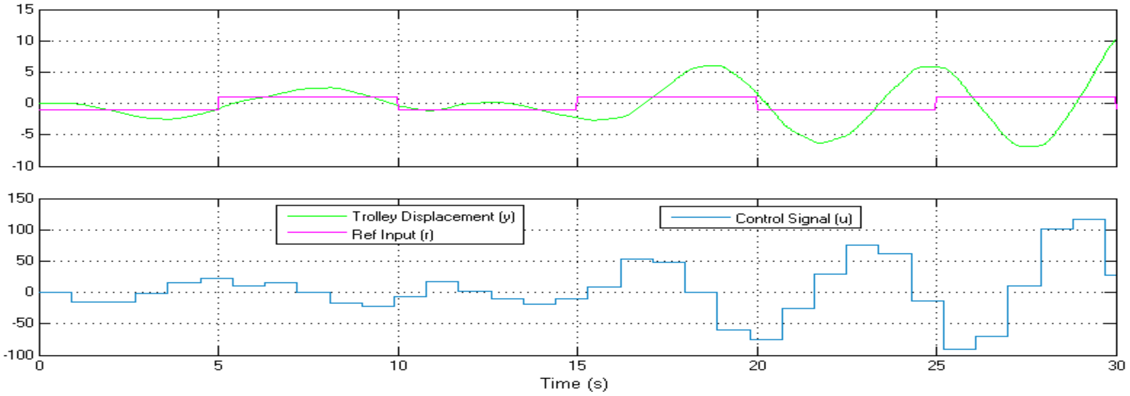
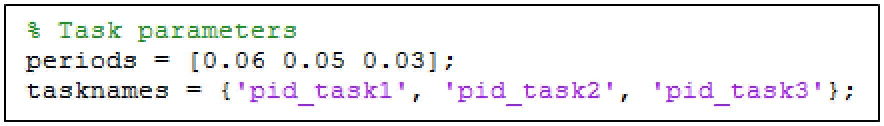


Figure 4.16 : Reference Input (r), Trolley Displacement (y) and Control Signal (u)

4.2 Result of Simulation for a Multi-Cart GCS

Figure 4.20 show the task parameter declaration for DMPA and EDF scheduling algorithm system. This parameter to make the sets point for every PID task system that involved in the process. That has three task means for three cart GCS. So, one task for one cart GCS in this proses. The period for Task is 0.03 s, Task 2 is 0.05 s, and Task 3 is 0.03 s.



```
% Task parameters
periods = [0.06 0.05 0.03];
tasknames = {'pid_task1', 'pid_task2', 'pid_task3'};
```

Figure 4.17: Task Parameter Declaration for EDF

4.2.1 Analysis of DMPA Algorithm

Figure 4.18 shows the graph control performance for DMPA scheduling algorithm. This graph show for Task 1 and Task 2 is similarity pattern. If the system only consider at Task 1 and Task 2, the system will be stable. But, for this analysis the system is unstable because the Task 3 there is no pattern shows on the graph. Therefore, for the Task 3 is no have Overshoot (OS) and Settling Time (T_s).

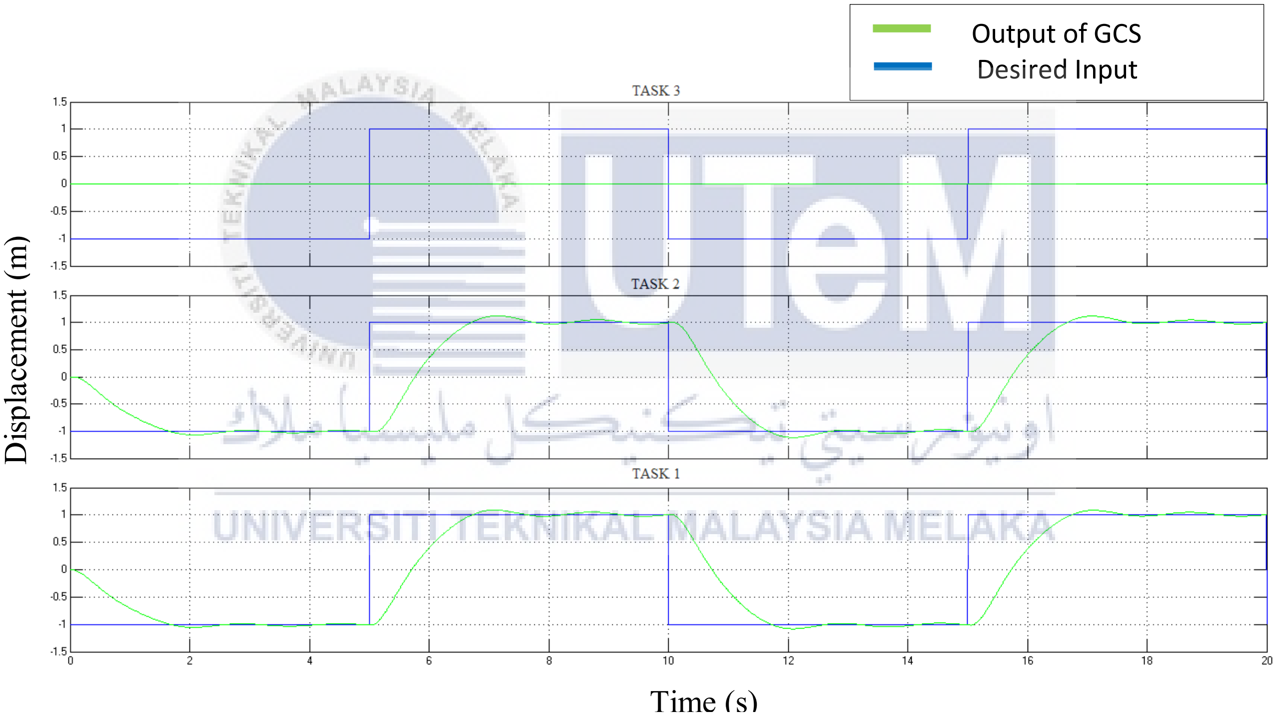


Figure 4.18: Control performance for DMPA analysis

Figure 4.19 shows the behavior for Task 1, Task 2 and Task 3 from computer schedule for DMPA analysis. The execution time for all tasks it same and unchangeable where the execution time is 0.02 s. But the period for all tasks is not same, where the period for Task 1 is 0.06 s, Task 2 is 0.05 s and 0.03 s for Task 3. The graph show for Task 1 is misses all it deadlines. However, for the Task 2 and Task 3 meet the deadlines.

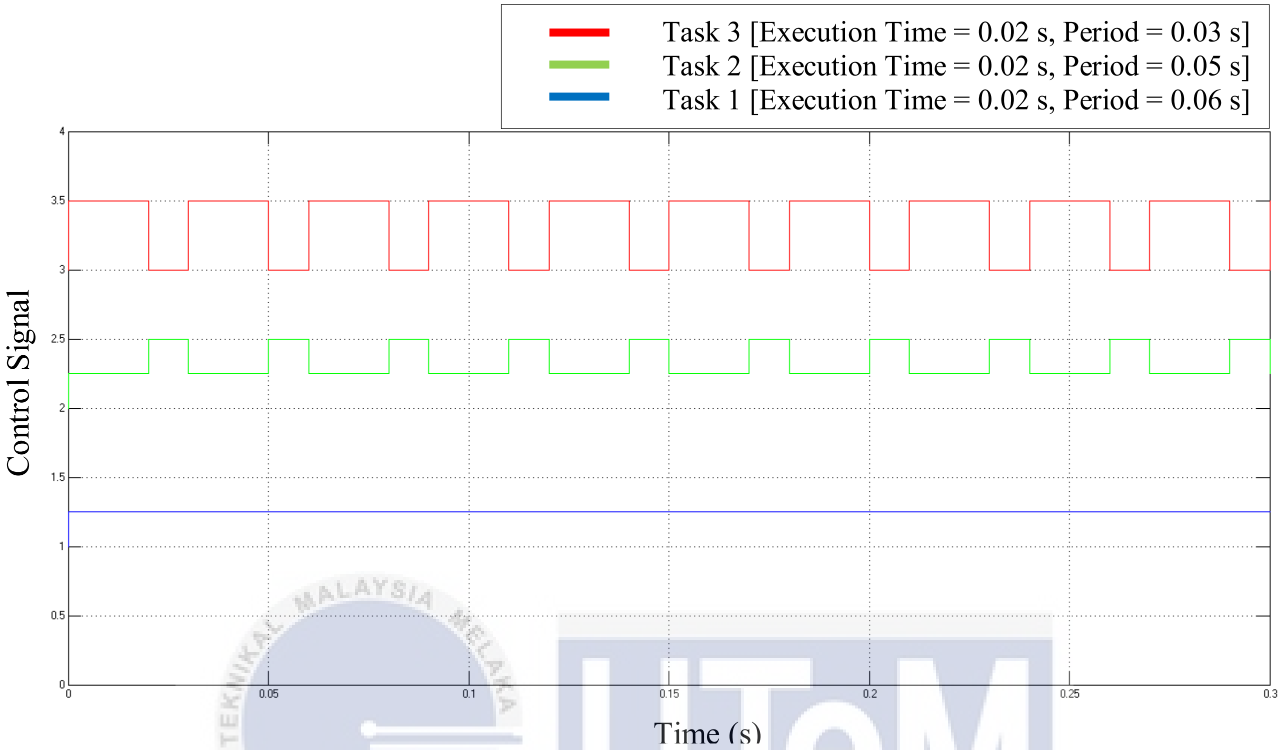


Figure 4.19: Computer schedule for DMPA analysis

4.2.2 Analysis of EDF Algorithm

Figure 4.20 shows the graph control performance for EDF scheduling algorithm. This graph show for Task 1, Task 2, and Task 3 is similarity pattern. From this graph, the trolley displacement for all task have Overshoot (OS) and Settling Time (T_s). However, the value OS and T_s for all tasks is not higher. Therefore, the overall performance for EDF algorithm is still satisfactory and considerable stability for corresponding control loop of the system for EDF scheduling algorithm.

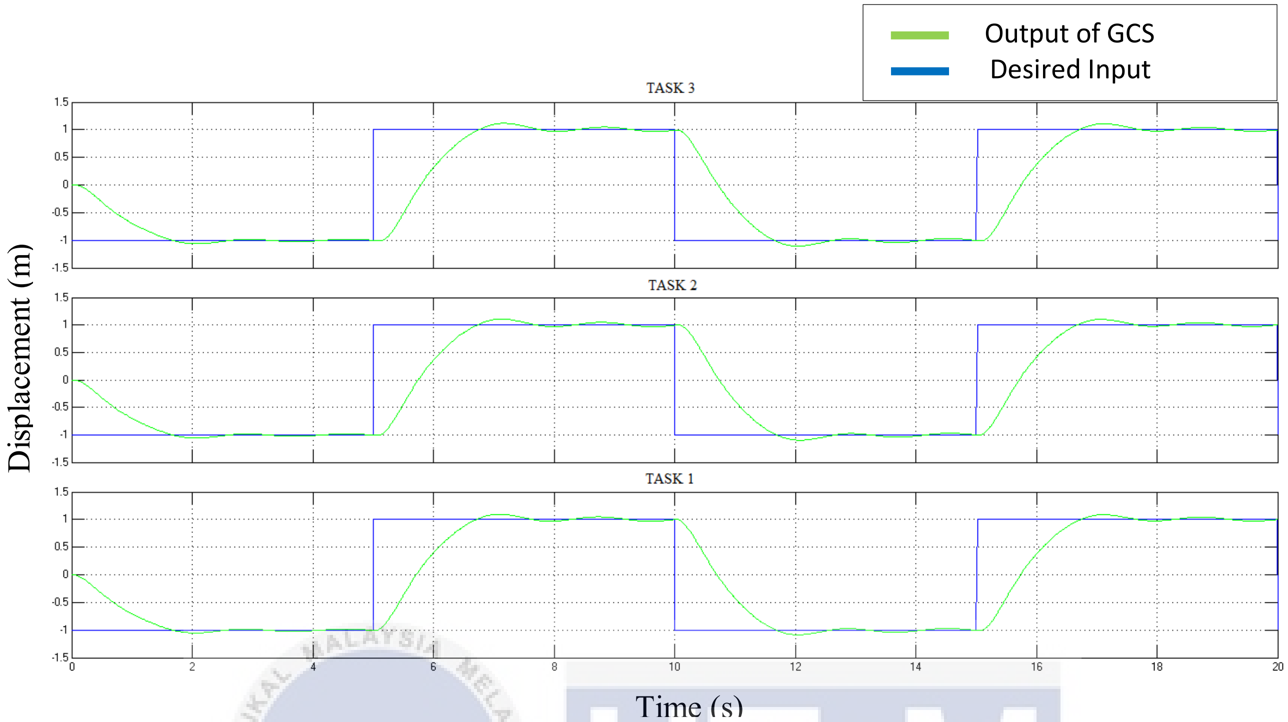


Figure 4.20: Control performance for EDF analysis

Figure 4.21 shows the behavior for Task 1, Task 2 and Task 3 from computer schedule for EDF analysis. The execution time for all tasks it same and unchangeable where the execution time is 0.02 s. But the period for all tasks is not same, where the period for Task 1 is 0.06 s, Task 2 is 0.05 s and 0.03 s for Task 3. The graph show for Task 1 meet the deadline and it mean the Task 1 on smooth system. The Task 2 has one cycle miss the deadline and Task 3 have four cycle miss the deadline. From this analysis, it show when the period is higher than execution time, the miss deadline is increasesd.

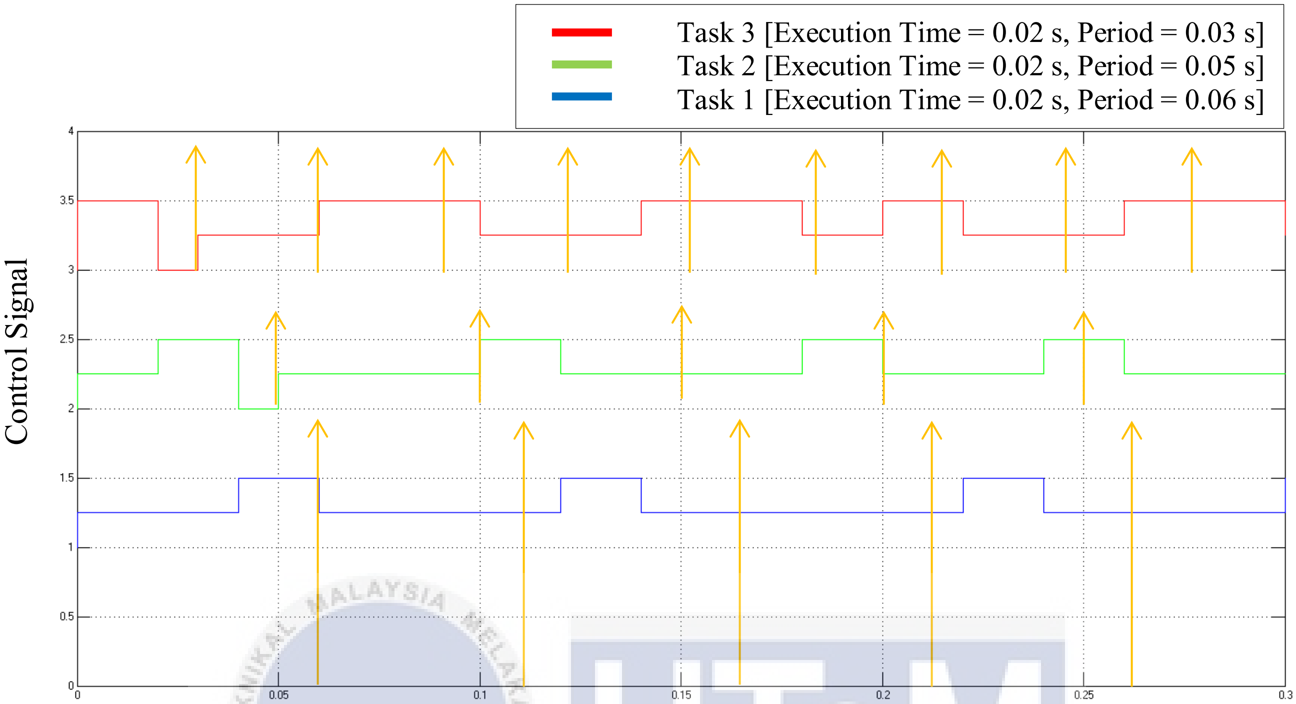


Figure 4.21: Computer schedule for EDF analysis

4.3 Result of GCS on DMPA and EDF Algorithm via CAN

4.3.1 Analysis DMPA and EDF Algorithm via CAN for CASE 1

Table 4.1 show the period and execution time is was set to Network Node. In this case, the all period is higher than execution time. But the period of controller is small than all sensor and actuator.

Table 4.1 : Period and Execute Time for Network Node

	Network Node	Period (s)	Execute Time (s)
1	Controller	0.01	0.005
2	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004
3	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004
4	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004
5	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004

Figure 4.22 show the performance of scheduling at network node and controller for DMPA and EDF scheduling algorithm analysis. The graph show the all sensor send message to Network. For this case the scheduling at network node and controller for DMPA and EDF scheduling algorithm are same pattern. The graph scheduling at network show the network received a message from the sensor does not exceed the stipulated period. Therefore, it is meet the deadline or the data is not miss the schedule. It is because the execute time all sensor not exceed the period.

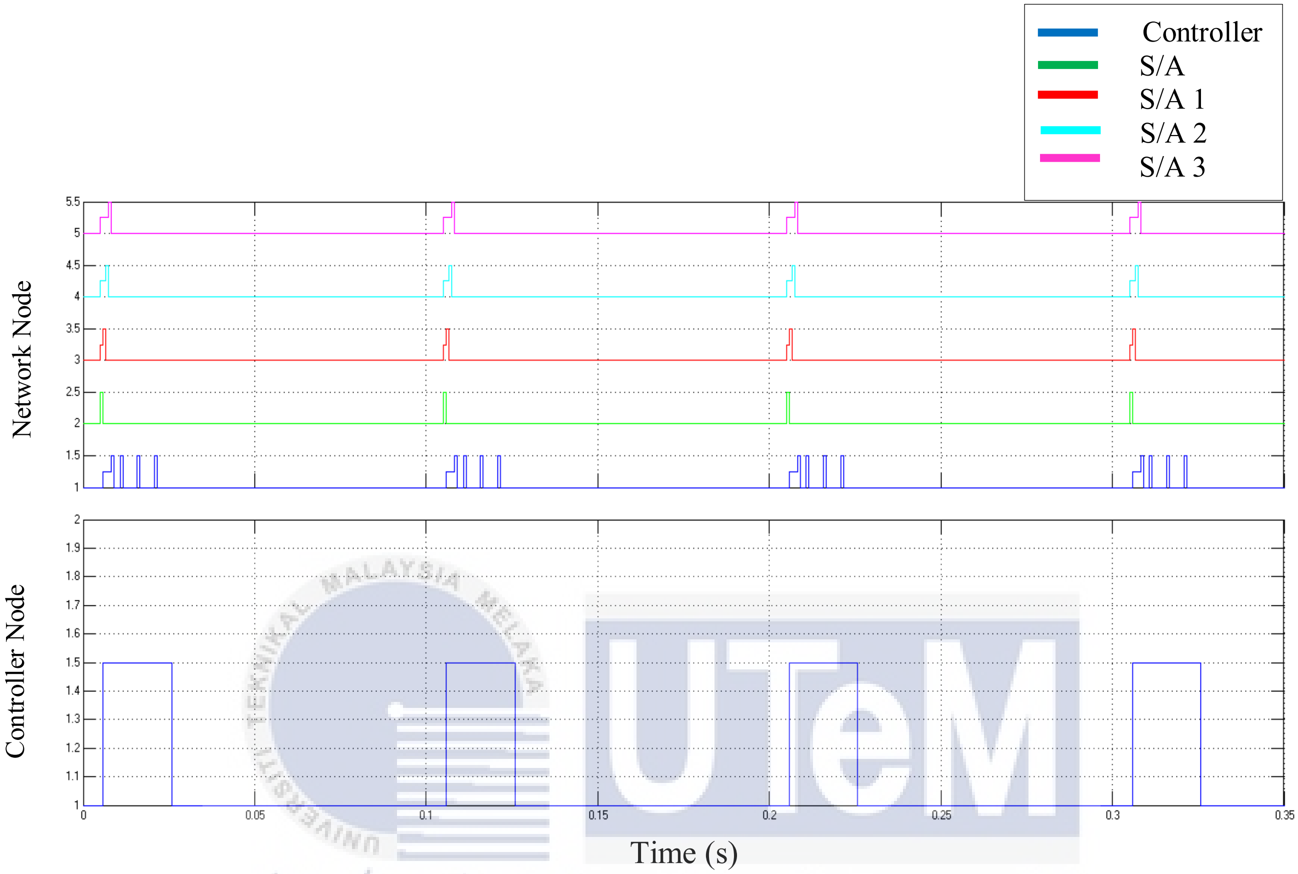


Figure 4.22: DMPA and EDF Schedule Analysis of Gantry Crane System with CAN

Figure 4.25 shows the performance DMPA and EDF analysis of Gantry Crane System (GCS) with CAN. From this analysis, the graph show output of four GCS with DMPA schedule algorithm is same when scheduling method is changes to EDF. This result shows that the output of Gantry Crane (y), Gantry Crane 1 (y1), Gantry Crane 2 (y2), and Gantry Crane 3 (y3) are stable and as required.

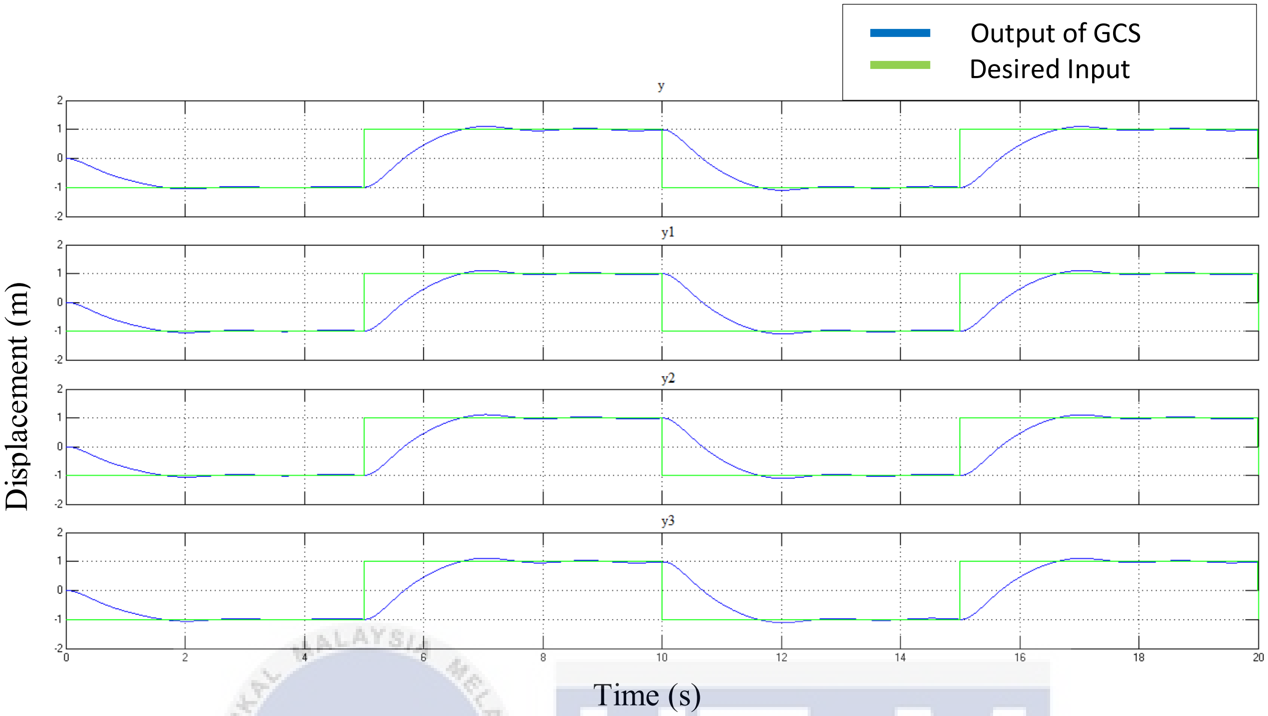


Figure 4.23: DMPA and EDF Analysis of Gantry Crane System with CAN

4.3.2 Analysis DMPA and EDF Algorithm via CAN for CASE 2

Table 4.2 show the period and execution time is was set to Network Node. In this case, the all period is higher than execution time. But the execution time for Sensor 2 and Sensor 3 is higher than Sensor 4 and Sensor 5.

Table 4.2 : Period and Execute Time for Network Node

	Network Node	Period (s)	Execute Time (s)
1	Controller	0.01	0.005
2	Sensor	0.1	Get Data : 0.05
			Send Msg : 0.05
	Actuator	0.1	0.004
3	Sensor	0.1	Get Data : 0.05
			Send Msg : 0.05
	Actuator	0.1	0.004
4	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004
5	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004

Figure 4.24 show the performance of scheduling at network node and controller for DMPA and EDF scheduling algorithm analysis. The graph show the all sensor send message to Network. For this case the scheduling at network node and controller for DMPA and EDF scheduling algorithm are same pattern. The graph scheduling at network show the network received a message from the sensor does not exceed the stipulated period. Therefore, it is meet the deadline or the data not miss the schedule. It is because the execute time all sensor not exceed the period.

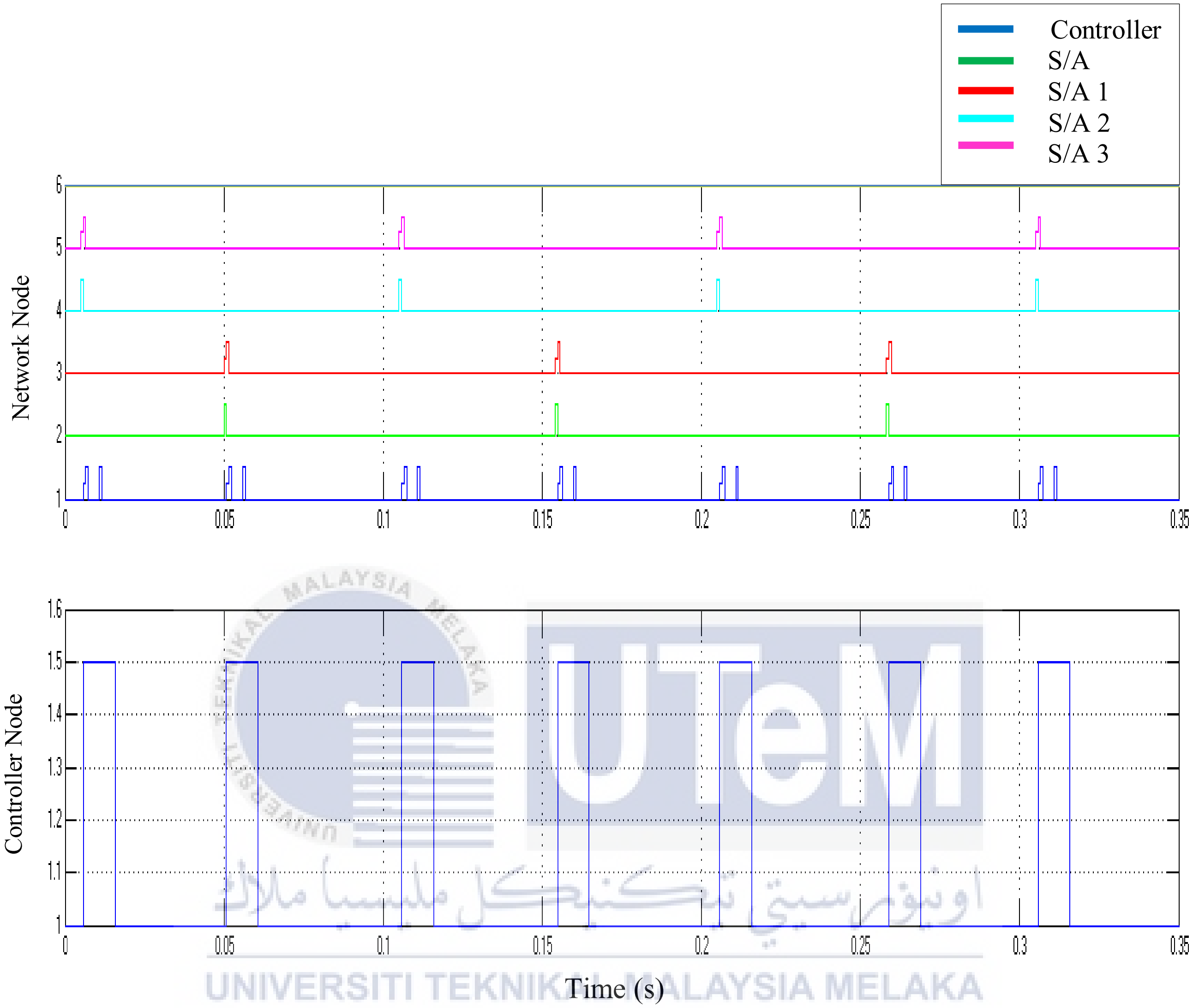


Figure 4.24: DMPA and EDF Schedule Analysis of Gantry Crane System with CAN

Figure 4.25 shows the performance DMPA and EDF analysis of Gantry Crane System (GCS) with CAN. From this analysis, the graph show output of four GCS with DMPA schedule algorithm is same when scheduling method is changes to EDF. This result shows that the output of Gantry Crane (y) and Gantry Crane 2 (y1) are towards unstable state. It is because, the overshoot at second loop for y and y1 is increased and the settling time is higher. However, the output for Gantry Crane 2 (y2) and Gantry Crane 3 (y3) is stable and as required.

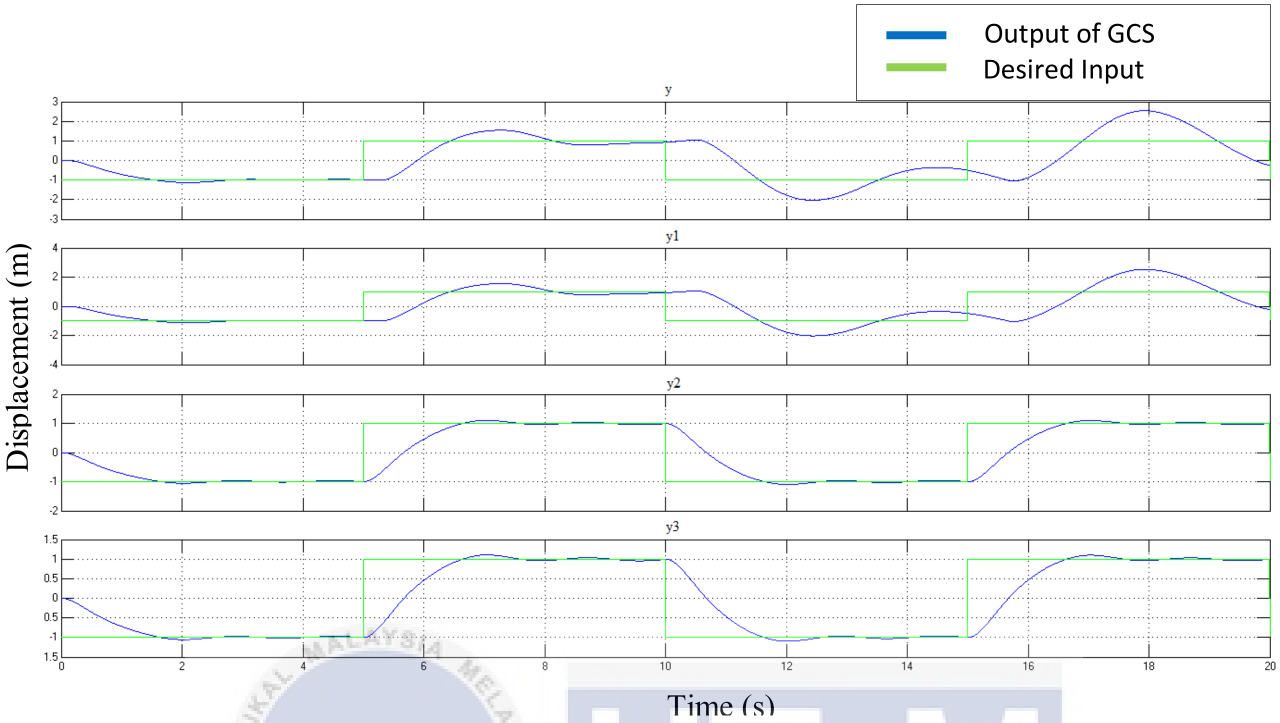


Figure 4.25: DMPA and EDF Analysis of Gantry Crane System with CAN

4.3.3 Analysis DMPA and EDF Algorithm via CAN for CASE 3

Table 4.3 show the period and execution time is was set to Network Node. In this case, the period for node 2, node 3 and node 4 are was changed. The execution time for node 2 and node 3 also change at get data and send massage.

Table 4.3 : Period and Execute Time for Network Node

	Network Node	Period (s)	Execute Time (s)
1	Controller	0.01	0.005
2	Sensor	0.05	Get Data : 0.1
			Send Msg : 0.02
	Actuator	1	0.004
3	Sensor	0.05	Get Data : 0.04
			Send Msg : 0.1
	Actuator	0.1	0.004
4	Sensor	0.05	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004
5	Sensor	0.1	Get Data : 0.005
			Send Msg : 0.004
	Actuator	0.1	0.004

Figure 4.26 show the performance of scheduling at network node and controller for DMPA and EDF scheduling algorithm analysis. The graph show the all sensor send message to Network. For this case the scheduling at network node and controller for DMPA and EDF scheduling algorithm are same pattern. For this case, the graph scheduling at network show the network received a message from the Sensor/Actuator and Sensor/Actuator 1 does exceed the stipulated period. Therefore, it is miss the deadline or the data miss the schedule. It is because the execution time for Sensor/Actuator get data is higher than period and execution time for Sensor/Actuator 1 send message is higher than period. But, the graph scheduling at network show the network received a message from the Sensor/Actuator 2 and Sensor/Actuator 3 does not exceed the stipulated period. Therefore, it is meet the deadline or the data not miss the schedule. It is because the execute time not exceed the period.

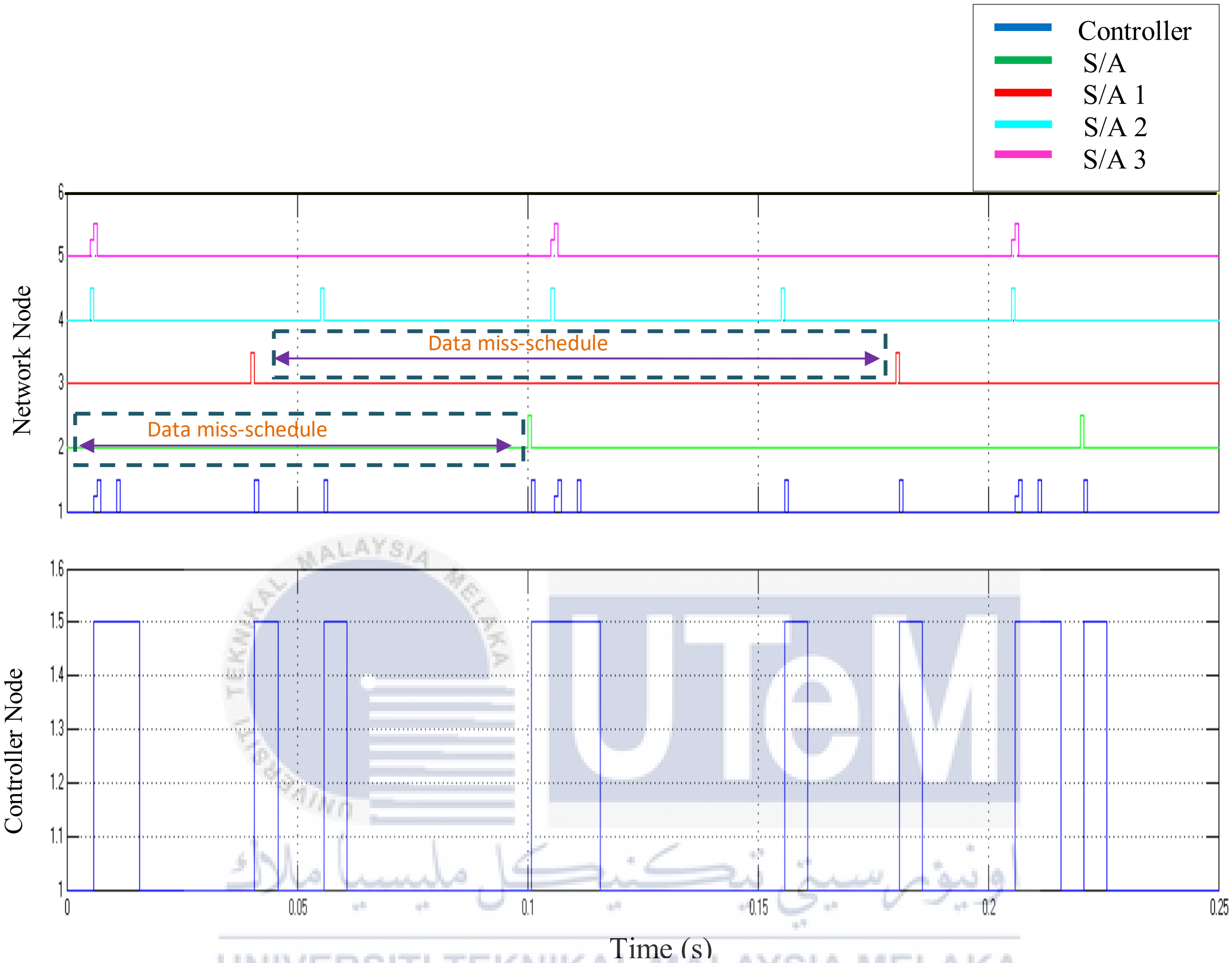


Figure 4.26: DMPA and EDF Schedule Analysis of Gantry Crane System with CAN

Figure 4.27 shows the performance DMPA and EDF analysis of gantry crane system (GCS) with CAN. From this analysis, the graph show output of four GCS with DMPA schedule algorithm is same when scheduling method is changes to EDF. This result shows that the output of Gantry Crane (y) and Gantry Crane 2 (y1) is unstable. The Gantry Crane and Gantry Crane 1 does not meet the output requirement. However, the output for Gantry Crane 2 (y2) and Gantry Crane 3 (y3) is stable and as required.

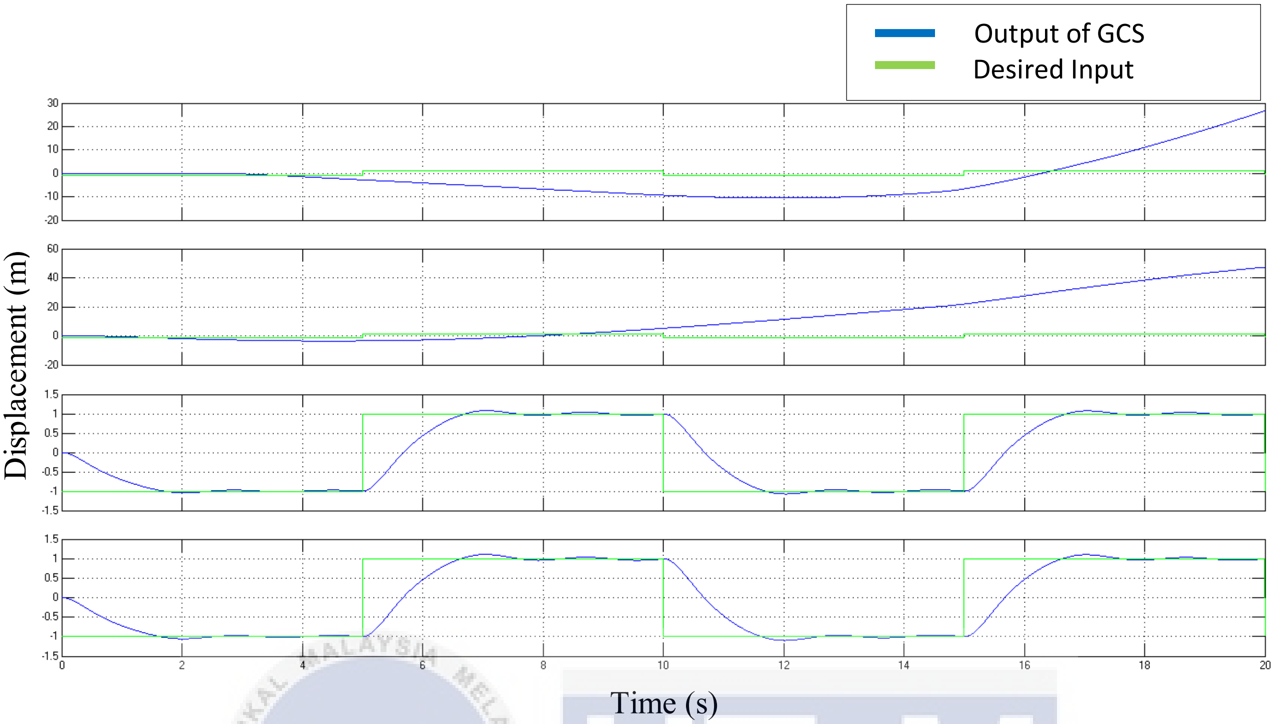


Figure 4.27: DMPA and EDF Analysis of Gantry Crane System with CAN

4.4 Discussion

4.4.1 Single-Chart GCS Analysis Discussion (DMPA and EDF)

Based on the results of the simulation Single-Chart GCS of DMPA and EDF, the output pattern of the two algorithms are the same. The result at table 4.1 show the effect of the control of the two algorithms that DMPA and EDF. Looking at Table 4.1, it indicates that the corresponding output is stable despite the implementation of the limitation period of time. The system is still stable despite execution time was increased to 0.12 s but the percentage overshoot is increased and settling time also increased, but not too much. When the execution time increased to 0.90s, the system becomes unstable and can be seen in the case of four that the graph of the two algorithms shows the output not smooth.

Table 4.4 : Effect of Control Performances

Single-Chart GCS Analysis Table (DMPA & EDF)				
Case	**1**	**2**	**3**	**4**
Execute time (s)	0.02	0.12	0.40	0.90
Period (s)	0.06	0.06	0.06	0.06
OS (%)	6	23	50	Unstable
Ts (s)	3.1	3.3	>5	Unstable

4.4.2 Multi-Chart GCS Analysis Discussion (DMPA and EDF)

Table 4.5 shows the DMPA Multi-Chart GCS have no miss the deadlines when the period 0.03 s and 0.05 s. All the deadline was missed when the period 0.06 s. The overall system for DMPA schedule algorithm cannot perform when the period very higher than execution time. It is not suitable to use the higher period.

For the EDF Multi-Chart GCS, the Table 4.2 show the miss deadline is happened when the period 0.03 s and 0.05 s. But, the number of miss deadline is increased when the period become smaller, where for period 0.03 s is 4 miss deadlines and for period 0.05s is 1 miss deadline. When the period is bigger as 0.06 s, the miss deadline was not happened. So, the result show the EDF schedule algorithm is suitable when the period is higher than execution time. The overall system for EDF schedule algorithm can perform better when the period is higher than execution time.

Table 4.5 : Effect of Control Performances

Multi-Chart GCS Analysis Table (DMPA)			
Task	**1**	**2**	**3**
Execute time (s)	0.2	0.2	0.2
Period (s)	0.6	0.05	0.03
OS (%)	4	6	-
Ts(s)	1.7	3.2	-
Schedule Performance DMPA	Miss All	No Miss	No Miss
Multi-Chart GCS Analysis Table (EDF)			
Task	**1**	**2**	**3**
Execute time (s)	0.2	0.2	0.2
Period (s)	0.6	0.05	0.03
OS (%)	4	4	4
Ts(s)	1.7	1.7	1.7
Schedule Performance DMPA	No Miss	1 Miss	4 Miss

4.4.3 Analysis Discussion DMPA and EDF Algorithm via CAN

Table 4.7 shows the effect of control performance for analysis DMPA and EDF algorithm via CAN. The overall performance is same output with DMPA and EDF algorithm in this system. The result show from this analysis, the all data is not miss-schedule. However, the overshoot for Gantry Crane and Gantry Crane 1 is unstable. Therefore, the output for Gantry Crane and Gantry Crane 1 are toward unstable. It is because the total execution time for get data and send message is same with period time. So, it mean the system with both algorithm of DMPA and EDF are not suitable when the execution time same with period time.

Table 4.6 : Effect of Control Performances

Analysis DMPA and EDF Algorithm via CAN for CASE 1					
Network Node		**Gantry Crane**	**Gantry Crane 1**	**Gantry Crane 2**	**Gantry Crane 3**
Period Sensor (s)		0.1	0.1	0.1	0.1
Execution Time Sensor (s)	**Get Data**	0.005	0.005	0.005	0.005
	Send Msg	0.004	0.004	0.004	0.004
Period Actuator (s)		0.1	0.1	0.1	0.1
Execution Time Actuator (s)		0.004	0.004	0.004	0.004
OS (%)		9.4	9.4	9.4	9.4
Ts (s)		2.8	2.8	2.8	2.8
Schedule Performance DMPA/EDF		No Miss	No Miss	No Miss	No Miss

Table 4.7 shows the effect of control performance for analysis DMPA and EDF algorithm via CAN. The overall performance is same output with DMPA and EDF algorithm in this system. The result show from this analysis, the all data is not miss-schedule. However, the overshoot for Gantry Crane and Gantry Crane 1 is unstable. Therefore, the output for Gantry Crane and Gantry Crane 1 are toward unstable. It is because the total execution time for get data and send message is same with period time. So, it mean the system with both algorithm of DMPA and EDF are not suitable when the execution time same with period time.

Table 4.7 : Effect of Control Performances

Analysis DMPA and EDF Algorithm via CAN for CASE 2					
Network Node		**Gantry Crane**	**Gantry Crane 1**	**Gantry Crane 2**	**Gantry Crane 3**
Period Sensor (s)		0.1	0.1	0.1	0.1
Execution Time Sensor (s)	**Get Data**	0.05	0.05	0.005	0.005
	Send Msg	0.05	0.05	0.004	0.004
Period Actuator (s)		0.1	0.1	0.1	0.1
Execution Time Actuator (s)		0.004	0.004	0.004	0.004
OS (%)		Unstable	Unstable	9.4	9.4
Ts (s)		Unstable	Unstable	2.8	2.8
Schedule Performance DMPA/EDF		No Miss	No Miss	No Miss	No Miss

From Table 4.8, shows the Gantry Crane and Gantry Crane 1 is miss-schedule when the execution time sensor get data for Gantry Crane is higher than period sensor and the execution time sensor send message for gantry 1 is higher than period. Therefore, the overshoot for Gantry Crane and Gantry Crane 1 is unstable. It mean, the displacement for

Gantry Crane and Gantry Crane 1 is unstable. It is because when the execution time is higher than period, the system of cart GCS will become unstable condition and is not as required. From this analysis, the DMPA and EDF schedule algorithm is same output for this GCS.

Table 4.8 : Effect of Control Performances

Analysis DMPA and EDF Algorithm via CAN for CASE 3					
Network Node		**Gantry Crane**	**Gantry Crane 1**	**Gantry Crane 2**	**Gantry Crane 3**
Period Sensor (s)		0.05	0.05	0.05	0.1
Execution Time Sensor (s)	**Get Data**	0.1	0.04	0.005	0.005
	Send Msg	0.02	0.1	0.004	0.004
Period Actuator (s)		1	0.1	0.1	0.1
Execution Time Actuator (s)		0.004	0.004	0.004	0.004
OS (%)		Unstable	Unstable	9.4	9.4
Ts (s)		Unstable	Unstable	2.8	2.8
Schedule Performance DMPA/EDF		Miss	Miss	No Miss	No Miss

4.5 Conclusion

This chapter provides more understanding based on the results shown by both the scheduling algorithm, it able to demonstrate how these two performance algorithms for difference case or task are executed. From input-output performance by DMPA and EDF could explain the difference between these two scheduling algorithms and the behavior of the schedule algorithm performance can be analyzed.

CHAPTER 5

CONCLUSIONS

5.1 Conclusion

As all know, GCS is a system that carries a heavy burden from one place to another. This system helps people from outside the human physical of strength to move large loads and heavier. Accidents often occur because the GCS need a better system for perfect operation. So, proper task execution is important to avoid accidents and perfect for work performed by GCS.

In this research, DMPA and EDF scheduling algorithms have been used to upgrade the GCS for perfect execution. The use of DMPA and EDF scheduling algorithms at Single-Chart GCS, that was only focused on the trolley displacements. It is has proved they are safe to the environment and the system is better. It is safe because when the task execution not exceed the period time, the model system is stable.

From the second objective, the Multi-Chart GCS was analyzed with implementation DMPA and EDF algorithm. This task, the analysis provides the overall behavior for the control performance and computer schedule behavior for DMPA and EDF algorithm. It conclude about the schedule performance of the system that is either miss or satisfied the schedule requirement.

Furthermore, the last objective is implementing the DMPA and EDF algorithm via CAN. From this analysis, it show the behavior the control performance cart GCS with CAN. Usually, the CAN is used to show to control the cart GCS more than one. When the sensor is used for send message to network and the network send data to actuator for run the cart GCS. So, the CAN can be analyzed that the data schedule is miss or not.

5.2 Future Work

This research about analysis of performance for Cart GCS using DMPA and EDF scheduling algorithm via Controller Area Network (CAN). This analysis just focused about displacement of chart GCS. For the future, it is recommended to involve with controlling the swing of cart GCS when carrying the various load of mass due to accidents avoidance.

REFERENCES

[1] H. I. Jaafar, Z. Mohammed, J. J. Jamian, A. F. Z. Abidin, A. M. Kassim, and Z. A. Ghani, "Dynamic Behaviour of a Nonlinear Gantry Crane System," Procedia Technology, vol. 11, pp. 419-425, 2013.

[2] Jaroslaw Smoczek, Janusz Szpytko''Design of Gain Scheduling Anti-Sway Crane Controler Using Genetic Fuzzy System''

[3] M. A. Zawawi, W. M. S. W. Zamani, M. A. Ahmad, M. S. Saealal, and R. E. Samin, "Feedback Control Schemes for Gantry Crane System incorporating Payload," IEEE Symp. Ind. Electron. Appl. (ISIEA2011), Langkawi, Malaysia, pp. 370–375, 2011.

[4] Manson, G.A., "Time-optimal control of and overhead crane model", Optimal Control Applications & Methods, Vol. 3, No. 2, 1992, pp. 115- 120.

[5] Auernig, J. and Troger, H. "Time optimal control of overhead cranes with hoisting of the load", Autornatica, Vol. 23, 1987, pp. 437-447.

[6] W. E. Singhose, L. J. Porter, and W. P. Seering, "Input shaped control of a planar gantry crane with hoisting," *Proc. 1997 Am. Control Conf. (Cat. No.97CH36041)*, vol. 1, no. June, pp. 97–100, 1997.

[7] R. J. Vaccaro, M. Hill, and P. R. Belanger, "tabilitook," 1995.

[8] N. C. Audsley, "Deadline-Monotonic Scheduling," no. September, 1990.

[9] C. M. Lewandowski, "HARD REAL-TIME SCHEDULING: THE DEADLINE-MONOTONIC APPROACH1," *Eff. Br. mindfulness Interv. acute pain Exp. An Exam. Individ. Differ.*, vol. 1, 2015.

[10] Alan Burns and Andy Wellings (2001). *Real Time Systems and Programming Languages (3rd Edition)*. AddisonWesley. p. 485. ISBN 0201729881.

[11] Earliest Deadline First Scheduling of Preemptable Data Streams Issues Related to Automotive Applications," *2015 IEEE 21st Int. Conf. Embed. Real-Time Comput. Syst. Appl.*, vol. 1, pp. 257–267, 2015.

[12] R. G. Gallager and A. C. Smith, "A Generalized Processor Sharing Approach to Flow Control In Integrated Services Networks," *Word J. Int. Linguist. Assoc.*, no. 3, 1992.

[13] J. Liebeherr, D. E. Wrege, and D. Ferrari, "Exact admission control for networks with a bounded delay service," *IEEE/ACM Trans. Netw.*, vol. 4, no. 6, pp. 885–901, 1996.

[14] L. Georgiadis, R. Guerin, and A. Parekh, "Optimal multiplexing on a single link: delay and buffer requirements," *IEEE Trans. Inform. Theory*, vol. 43, no. 5, pp. 1518–1535, 1997.

[15] L. Georgiadis, R. Guérin, V. Peris, and K. N. Sivarajan, "Efficient network QoS provisioning based on per node traffic shaping," *IEEE/ACM Trans. Netw.*, vol. 4, no. 4, pp. 482–501, 1996.

[16] N. Audsley, a Burns, M. Richardson, K. Tindell, and a J. Wellings, "Applying New Scheduling Theory To Static Priority Preemptive Scheduling," no. September, 1993.

[17] P. G. Student, T. C. E. Stes, and E. Vadgaon, "A Review : Control Area Network (CAN) based Intel- ligent Vehicle System for Driver Assistance using Advanced RISC Machines (ARM)," vol. 00, no. c, pp. 3–5, 2015.

[18] J. Hu and C. Xiong, "Study on the Embedded CAN Bus Control System in the Vehicle," *2012 Int. Conf. Comput. Sci. Electron. Eng.*, pp. 440–442, 2012.

[19] Kumar, M., A. Verma, and A. Srividya, Response-Time Modeling of Controller Area Network (CAN). Distributed Computing and Networking, 2009: p. 163-174.

[20] Tindell, K., A. Burns, and A.J. Wellings, Calculating controller area network (CAN) message response times. Control Engineering Practice, 1995. 3(8): p. 1163-1169.

[21] Marwedel, P., Embedded System Design: Embedded Systems Foundations of Cyber Physical Systems2010: Springer Verlag.

[22] Estrada, B. and P. Mariano, Development of uclinux Platform for Cal Poly Super Project, 2008, the California Polytechnic State University.

[23] Mahmud Iwan Solihin, Wahyudi, M.A.S. Kamal and Ari Legowo, “Fuzzy-tuned PID Anti-swing Control of Automatic Gantry Crane ,” Journal of Vibration and Control, pp. 127-145, 2010.

[24] M. I. Solihin, Wahyudi, M.A.S Kamal and A. Legowo, Optimal PID Controller Tuning Of Automatic Gantry Crane Using PSO Algorithm,” *Proceeding of the 5th International Symposium on Mechatronics and its Applications (ISMA08)*, Amman, Jordan, May 27 29, pp. 1-5, 2008

APPENDIX A

Project Planning

Month Activities	SEPT		OCT				NOV				DEC				JAN
FYP 1 (SEPT 2015- JAN 2016)															
Choosing a title	■														
Gather the information		■	■												
Research				■	■	■	■								
Analysis						■	■	■	■	■					
Preliminary result							■	■	■	■	■				
Report				■	■	■	■	■	■	■	■	■	■	■	■
Presentation													■		
FYP 2 (FEB 2016- JUN 2016)	FEB		MAC				APR				MAY				JUN
Simulation	■	■	■	■											
Troubleshooting				■	■	■	■	■	■						
Result						■	■	■	■	■	■	■	■	■	
Report	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Presentation															■