" I hereby declare that I have read this thesis and acknowledge it has achieved the scope and quality for the award of the Degree of Electronics Industrial."

Signature : ....

Supervisor : PUAN SITI HUZAIMAH BINTI HUSIN

Date : MAY 2006

DESIGN THE CAR PARK CONTROL USING GRAFCET

VIGNESH KUMAR S/O SUBRAMANIAM

This report is submitted in partial fulfillment of requirements for the Bachelor Degree of Engineering (Industrial Electronics)

KOLEJ UNIVERSITI TEKNIKAL KEBANGSAAN MALAYSIA

MEI 2006

" I hereby declare that I have complete this thesis and acknowledge. References of information obtained from other are specially quoted, otherwise the rest of the information presented through this study is solely worked and experimentally carried out by the author "

Signature : ....

Student : VIGNESH KUMAR S/O SUBRAMANIAM

Date : MAY 2006

ACKNOWLEGDEMENT

First and foremost, I would like to take this opportunity to extend my deepest gratitude to all these who have assisted me in making his project a success. A special thanks to my respected and supportive supervisor, Puan Siti Huzaimah Binti Husin for her patience and guidance throughout the course of this project.

I would also like to take this opportunity to thanks all the KUTKM lecturers for their support and guidance throughout the course of the project.

Also not forgetting to thank my lovely supportive family and all my beloved friends for their help, support and guidance.

THANK YOU.

ABSTRACT

The title of my project is design the car park control using Grafcet as programming tool. This project is chosen because it exposes myself to few field of studies for example Grafcet and PLC. Furthermore this project also is very practically feasible to construct, practically applicable and useful and has the potential to be marketed in addition it is different from other existing similar in the market.

The Grafcet is the design tool for the PLC was chosen since it gives more structured approach rather than conventional way. This project use the PLC to control the parking area. This simple car park control system allows a certain number of cars parking space. Every time a car comes in, the PLC will add one and subtract one when a car goes out. When a determined number of cars are registered, the car park full sign will be lighted to inform oncoming vehicles not to enter. Basically this system allows a car when there is a space in the car park and show sign (light) to inform when the car park is full.

The final product of this project is a fully functional hardware that achieves the initial objectives of this project and at the same time illustrates the practical feasibility of our idea. However since the main objectives is only to illustrate the practical feasibility of our idea, therefore the specification and quality of certain components have been compromised to reduce the project cost. Therefore in real life application, the specifications of certain components have to be changed for optimum and satisfactory performance.

TABLE OF CONTENT

CHAPTER I

INTRODUCTION

1.0 Introduction

The aim of this final project is to design the car park control using the Grafcet and PLC to control the parking area. Basically we must know what is PLC. A PLC (Programmable Logic Controller) is a device that was invented to replace the necessary sequential relay circuits for machine control. The PLC works by looking at its inputs and depending upon their state, turning on/off its outputs. The user enters a program, usually via software, that gives the desired results.

PLCs are used in many "real world" applications. If there is industry present, chances are good that there is a plc present. If you are involved in machining, packaging, material handling, automated assembly or countless other industries you are probably already using them. If you are not, you are wasting money and time. Almost any application that needs some type of electrical control has a need for a plc.

For example, let's assume that when a switch turns on we want to turn a solenoid on for 5 seconds and then turn it off regardless of how long the switch is on for. We can do this with a simple external timer. But what if the process included 10 switches and solenoids? We would need 10 external timers. What if the process also needed to count how many times the switches individually turned on? We need a lot of external counters.

As you can see the bigger the process the more of a need we have for a PLC. We can simply program the PLC to count its inputs and turn the solenoids on for the specified time.

This site gives you enough information to be able to write programs far more complicated than the simple one above. We will take a look at what is considered to be the "top 20" plc instructions. It can be safely estimated that with a firm understanding of these instructions one can solve more than 80% of the applications in existence. That's right, more than 80%! Of course we'll learn more than just these instructions to help you solve almost ALL your potential plc applications.

1.1 Problem Statement

Among the analyzed problem is:

- Do not need manpower when the car park system can function automatically because there is a sensor to detect when cars approach.
- Most of the car parking area do not have counter and full sign display to show the users whether there is a space for their vehicle can be parked or not. It is very difficult when there is no counter display when a car comes in and out. So when a determined number of cars are registered, the car park full sign will be lighted to inform oncoming vehicles not to enter.
- Since some car parking system doesn't equipped with programmable signage to inform whether the parking is full or not, the car users will waste their time and petrol consumption to find a free parking lot.
- The data collection from the systems can be used for future development

1.2 Project Objectives

This project is consisted of two parts. Part I, the objectives of this work is to familiarize with the hardware itself, the function and the controlling elements. Since the PLC as the controller, the CX Programmer as the programming tool will also be learned in this part.

Part II, the objective is to write the programming using the GRAFCET as a graphical tool in order to develop the car park system. In further, the objective is to show the GRAFCET can be used as a design tool that can helps the designers to come out with the structured, safety, working and readability control program.

The objective in producing such a project is to come up with something that has never been ventured into and also that looks out for technology, safety and law abiding. This is where, I came up with an idea of create a display counter in the car park area to inform oncoming vehicles to enter or not to enter. Every time a car comes in, the PLC will add one and subtract one when a car goes out. When a determined number of cars are registered, the car park full sign will be lighted. Basically this system allows a car when there is a space in the car park and show sign (light) to inform when the car park is full. The advantage of this system is convenient to people because it saves time, petrol, cuts down manpower and help to improve the technology in certain places. This system can also used in shopping complex, campus and private car park

Internally, this project tests our knowledge in our area of specialty that is an electronic and electric theoretically as well as practically. In order to conduct our project, a lot of research and study has to be done on many things namely the counter, the working concept of a motor for parking bar, components and many more.

The project has also assigned because it makes us very important decision like time dividing among many things and the project, builds our character where we have to be really hardworking and have the perseverance.

Basically this is the objectives of the project that is going to be done in order to achieve a reasonable well one.

1.3 Project scopes of work

The scope of the work for this project is basically must familiarize with the PLC programming. Study about the PLC programming tool, "GRAFCET". Actually Grafcet is a flow chart that we plan how the system functions. In this project, we mainly concentrate on design using GRAFCET. Design a complete GRAFCET diagram of this module and transfer the designation of GRAFCET on to ladder diagram. Next, must familiarize with the equipments, components, and the things that we need to be used to create the car parking system.

Finally plan how to create or develop the model of the car parking system. I planned to use few equipments for the project such as sensors to detect the parking bar when car approach but then it is to expensive and costly. Therefore I planned to use micro switch as replacing the sensor for the parking bar in the project.

CHAPTER II

LITERATURE STUDY / REVIEW

2.0 Introduction

The cost of developing software for the logic controller in the manufacturing world becoming increasing day by day. Furthermore, in designing the representation of the logic controller such as "PLC", many companies using a different languages. In the desire to improve the capability of this logic controller, these languages become more complex. This is in contradiction with the main goals of the PLC's, which is that he "PLC" is a very simple system because it is very important that "PLC" must be reliable. There obviously needs to be a consistent philosophy for development reusable code for manufacturing controller systems

In today's economic context, the design of these control applications is of a great impact in terms of productivity and production costs. Because of these costs, of the complexity of the control system and the multiple hardware/software combinations, the designer has to take the safety of this system into account. In this context, it is necessary to provide the designer with verification methods that ensure the safety and live ness of the control system. In deep, the verification methods will help the designer to troubleshoot the programs while the error occurs in the production. One way to ensure the safety of the PLC programs is by using "framework" or standard while programming process began.

Thus, the usage of GRAFCET is a very good tool for logic controller specification has been approved. It allows modeling of concurrency and synchronization. Above all, the input-output behavior is specified without doubt. When some parts of logic controller can be describe separately, one can use macro steps to simplify the model. In addition, the comparison was made between State Table, Petri Net, RLL and GRAFCET, and it make an evident that the GRAFCET applications was the easiest among all.

The strength of the GRFACET is the graphical nature of the language makes GRAFCET easy to learn and use. The ability to test the different ideas quickly has been very useful in determining the final design. The GRAFCET helps the designer determine: modularization of the code, functions that can be performed in parallel, communication between parallel processes, and problems in control flow. The further extension of the GRAFCET usage which touches on how these tools can be applied to the reduction of a model in a specific context such as for the model that has typically over fifty steps and transitions. The GRAFCET also has been successful in order to converting existing RLL programs into the form that considerably easier to understand and it's also helps to modify the programs. For the good solution, the GRAFCET can be implemented to avoid the damages and system failures during the plant operation, due to interactions between human operators and plant. This statement was fully supported shown that the GRAFCET language has a particular characteristics that support supervision of external actions over a process. The results demonstrated shown that GRAFCET implementation over a PLC can avoid human errors and can indicate on a set of outputs, the parts and variables of the plant ha do not satisfy the interaction demand conditions. At the end, to make GRAFCET more efficient and applicable, it must introduce two different techniques to make proofs on the properties of this language

2.1 Historical Development

In 1975, the working group called "Logical Systems" from AFCET (Association Francaise de Cybernetique Economicque et Tecnichque) create the standardization of a requirement representation of a logical automated system. This group trying to define a simple formalism, accepted by everyone and well-adapted for the representation of the sequential evolutions of a system understandable by designers as well as by users and providing potentially easiness for the implementation with hardware and/or software solution. In December 1997, the Grafcet which means functional graph of a step- transition command was derived as a tool of state graph. After Grafcet had been introduced in higher and technical education, it was supported by the arrival of the first programming language. Allowing implementation of Grafcet specification models on industrial logic controllers; it became an AFNOR standard in 1982known under the reference NF c03190.

Aware of the necessity to precise how to implement a Grafcet specification with hardware and/or software, a synthesis document on recommended interpretation for the transformation from the Grafcet specification model to a specific realization. This though lead group to propose in 1987 in an AFCET. Synthesis document, a certain number of Grafcet extensions to meet users' expectation. So as to strengthen the diffusion of Grafcet, and thanks to the efficient action of Paul BRARD (la Telemecanique Electrique) Grafcet became an international standard in 1988 under name: "Sequential Function Chart" (SFC).

In 1989-1996, the study was carried out of the extension and moved towards a strengthening of the Grafcet theoretical basis. As a matter of fact, the growing demand of Grafcet users regarding a good understanding of the complexity and the dynamic behaviors of the models, regarding the prove of characteristics for the development of safe operational system, or regarding the prospective use of the Grafcet model for further modeling tools, did not match the Grafcet definitions that were only textual.

In 1997 the group organized a presentation stand at EDUCATEC'97 from December 2nd to December 5th, 1997(CNIT –Paris La Defense) to mark the 20yh birthday of Grafcet. This event gave a complete review of the action led in education, industry, standardization, and research and to show the progress mate during the last 20 years. In June 1998 an invited session at IFAC INCOME'98 drew attention to foreign laboratories which carry on with researches on Grafcet. The theoretical works on the Grafcet in model are increased by taking into account the characteristics of Grafcet in relation to the characteristics of the system which describe the behavior.

In this sense, the current orientations aim at better defining the properties of the frontier separating Grafcet from its environment (interactions by input- outputs, topology of these interactions exploiting the definition of both temporal scales, receptivities, actions, events, the taking into account of the semantics of the modelized system as a guide for the elaboration of Grafcet. I the future , the Grafcet group will certainly imagine other actions of the same kind in order to enable Grafcet to remind a model with a rich past and a promising future.

The Guts Inside

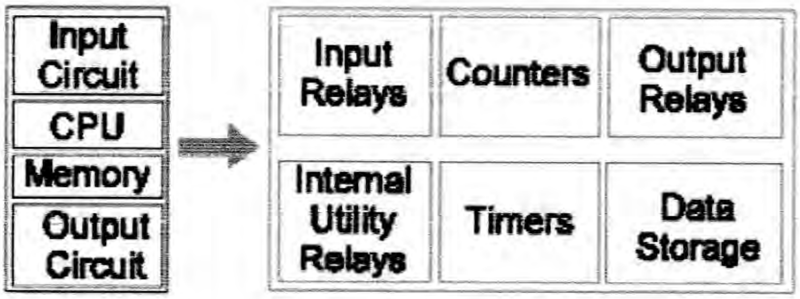

Figure 2.0: PLC CPU

The PLC mainly consists of a CPU, memory areas, and appropriate circuits to receive input/output data as shown in Figure 2.0. We can actually consider the PLC to be a box full of hundreds or thousands of separate relays, counters, timers and data storage locations. Do these counters, timers, etc. really exist? No, they don't "physically" exist but rather they are simulated and can be considered software counters, timers, etc. These internal relays are simulated through bit locations in registers.

What does each part do?

- INPUT RELAYS-(contacts) these are connected to the outside world. They physically exist and receive signals from switches, sensors, etc. Typically they are not relays but rather they are transistors.
- INTERNAL UTILITY RELAYS-(contacts) these do not receive signals from the outside world nor do they physically exist. They are simulated relays and are what enables a PLC to eliminate external relays. There are also some special relays that are dedicated to performing only one task. Some are always on while some are always off. Some are on only once during power-on and are typically used for initializing data that was stored.

- COUNTERS-These again do not physically exist. They are simulated counters and they can be programmed to count pulses. Typically these counters can count up, down or both up and down. Since they are simulated they are limited in their counting speed. Some manufacturers also include high-speed counters that are hardware based. We can think of these as physically existing. Most times these counters can count up, down or up and down.
- TIMERS-These also do not physically exist. They come in many varieties and increments. The most common type is an on-delay type. Others include off-delay and both retentive and non-retentive types. Increments vary from 1ms through 1s.
- OUTPUT RELAYS-(coils) these are connected to the outside world. They physically exist and send on/off signals to solenoids, lights, etc. They can be transistors, relays, or triacs depending upon the model chosen.
- DATA STORAGE-Typically there are registers assigned to simply store data. They are usually used as temporary storage for math or data manipulation. They can also typically be used to store data when power is removed from the PLC. Upon power-up they will still have the same contents as before power was removed. Very convenient and necessary

PLC Operation

A PLC works by continually scanning a program. We can think of this scan cycle as consisting of 3 important steps. There are typically more than 3 but we can focus on the important parts and not worry about the others. Typically the others are checking the system and updating the current internal counter and timer values.

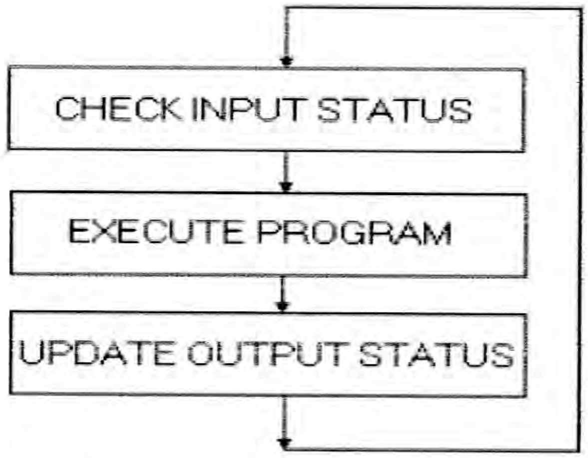

Figure 2.1 Scan Cycle

Step 1-CHECK INPUT STATUS-First the PLC takes a look at each input to determine if it is on or off. In other words, is the sensor connected to the first input on? How about the second input? How about the third. It records this data into its memory to be used during the next step.

Step 2-EXECUTE PROGRAM-Next the PLC executes your program one instruction at a time. Maybe your program said that if the first input was on then it should turn on the first output. Since it already knows which inputs are on/off from the previous step it will be able to decide whether the first output should be turned on based on the state of the first input. It will store the execution results for use later during the next step.

Step 3-UPDATE OUTPUT STATUS-Finally the PLC updates the status of the outputs. It updates the outputs based on which inputs were on during the first step and the results of executing your program during the second step. Based on the example in step 2 it would now turn on the first output because the first input was on and your program said to turn on the first output when this condition is true.

After the third step the PLC goes back to step one and repeats the steps continuously. One scan time is defined as the time it takes to execute the 3 steps listed above.

CHAPTER III

METHODOLOGY

Referring to Figure 3.0, there are few stages that will be involved in order to work out the aimed objectives in this project. The figure was intended to feature the entire project process. Initially, the first stage highlighting the preparation and familiarization level.

The second stage is regarding the study and familiarization of the GRAFCET as the design tool to represent the logic controller. Since the available references using higher technical standard jargon, the author tried to produce a guideline that can helps the lecturers and students in future usage, aimed to represent the GRAFCET in a simple approach. Some experimental workings were developed in order to get more understanding how the GRAFCET itself.

This importance part in this project is to develop the programming design the car park system using the GRAFCET as a tool design. The CX-Programmer, the programming tool for OMRON PLC is applied to create the programs of the car park system.

PROJECT METHODOLOGY

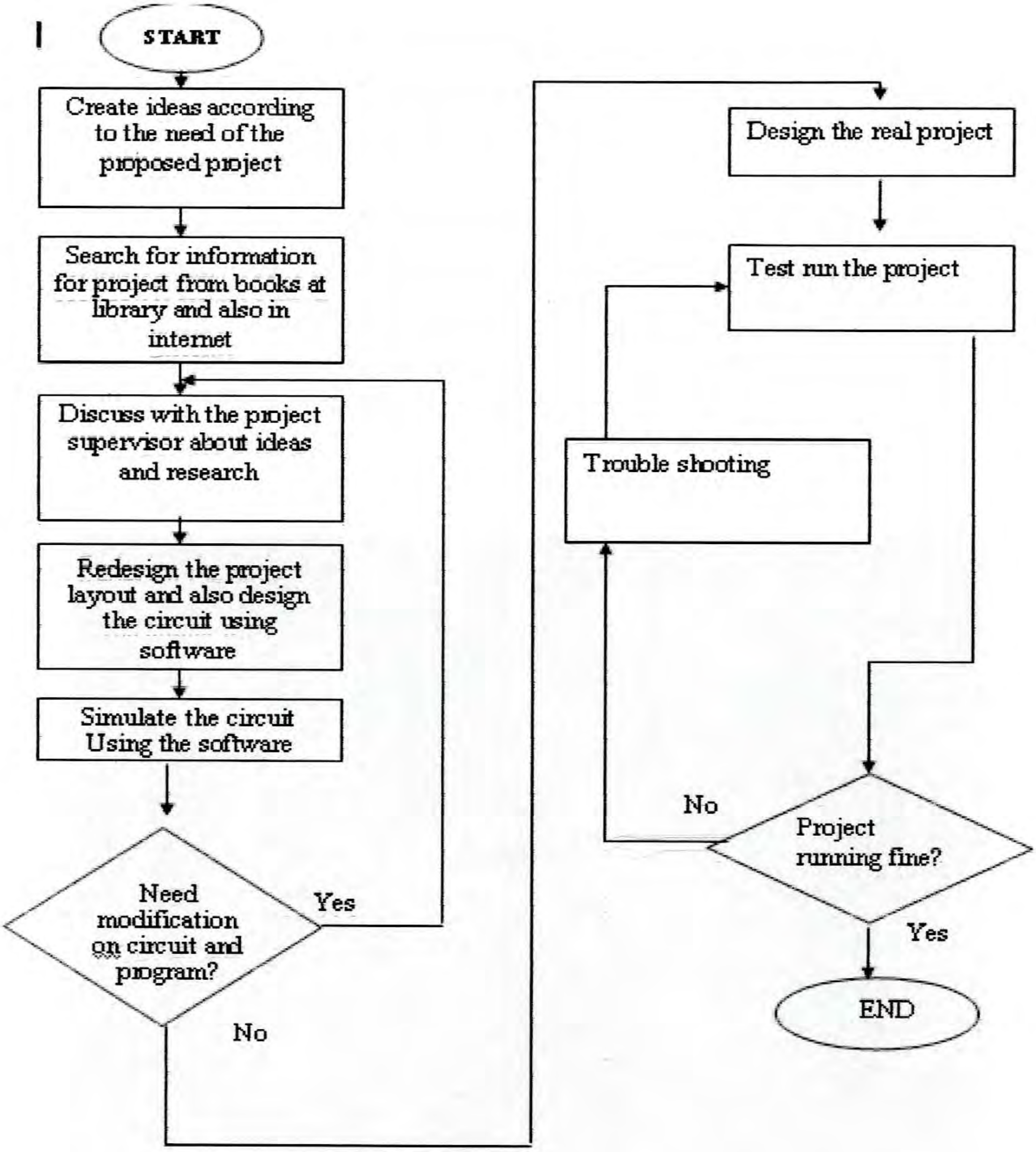

Figure 3.0: Methodology Flow Chart

3.1 Introduction

The producers and methods that been used to achieve the project is plan how to design a smart car parking system. So to gain some knowledge I have done a survey throughout Malacca and Johor Bahru (Southern zone) to find out what type of car parking system have been using.

So far, I have realized that only place using the smart car park system that I plan to design for my final year project is at KLIA (Kuala Lumpur International Airport). Since KLIA have implemented this car park design I take an initiative to go and have a look on the car park system. I have snap few photos around the car parking area at KLIA. Below are the pictures:

Figure 3.1: KLIA Car Parking Areas

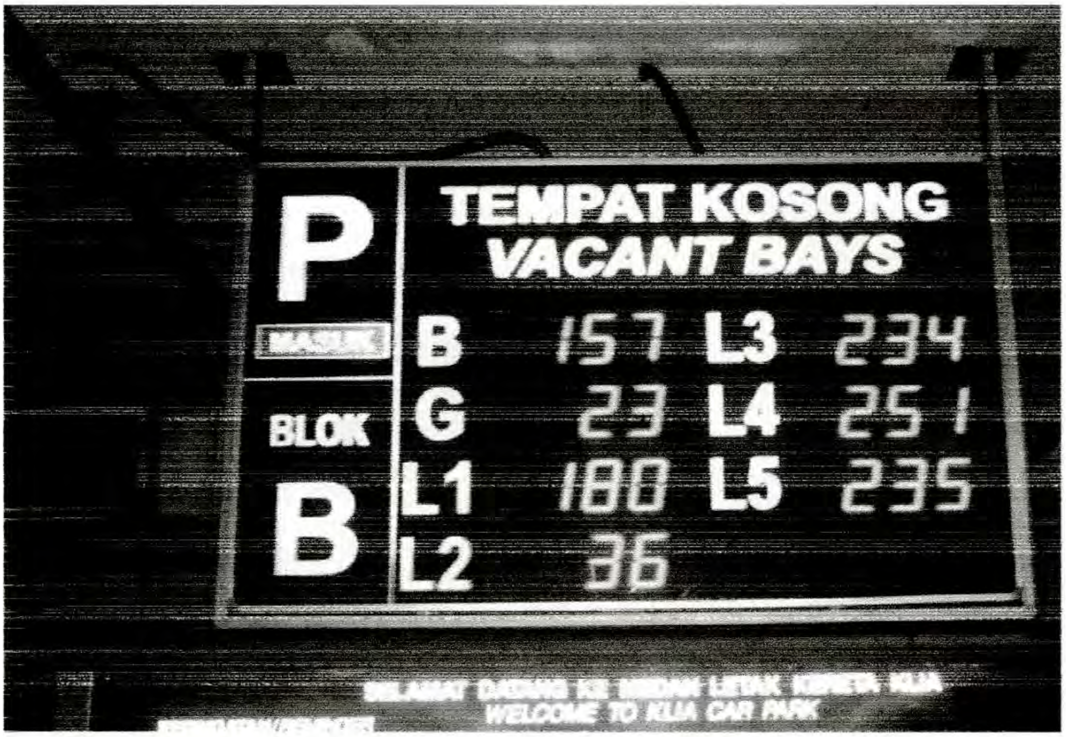

Figure 3.2: Displays for Vacant Bays at KLIA

Figure 3.3: Display and Counter at KLIA

3.2 Research and Analysis on car parking system at Malacca.

From the research and observation that I have done in few places at Malacca shows that there is no any car park system with car in and out counter display and full sign display. Below shows few figure at car parking area in Malacca.

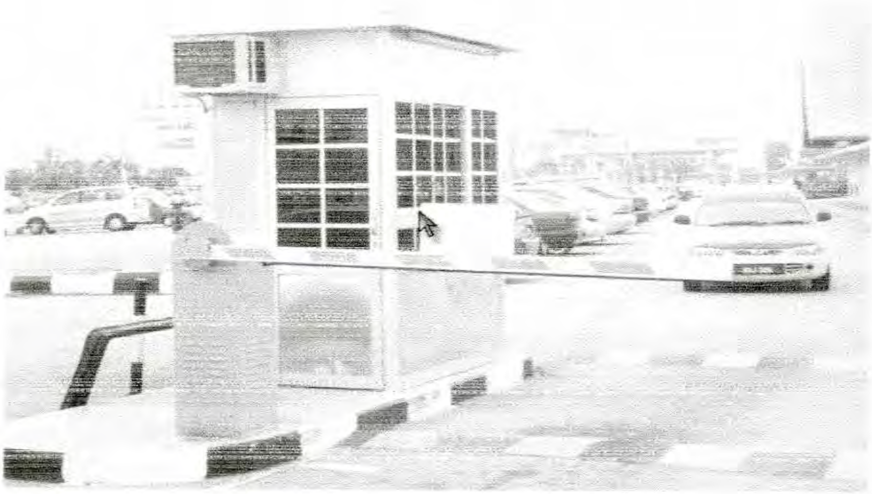

Figure 3.4: Malacca Central car parking system.

Figure 3.5: Malacca Central car parking system.