DIGITAL COUNTER FOR TRAFFIC LIGHT

MOHD HAFIZI YUSOF

A FINAL YEAR PROJECT REPORT
SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF BACHELOR OF ELECTRICAL
AND ELECTRONICS ENGINEERING

SCHOOL OF ENGINEERING AND
INFORMATION TECHNOLOGY
UNIVERSITI MALAYSIA SABAH
2008

UNIVERSITI MALAYSIA SABAH

BORANG PENGESAHAN STATUS TESIS@

JUDUL: DIGITAL COUNTER FOR TRAFFIC LIGHT

IJAZAH: IJAZAH SARJANA MUDA KEJURUTERAAN ELEKTRIK DAN ELEKTRONIK

SAYA MOHD HAFIZI YUSOF (HURUF BESAR)

SESI PENGAJIAN: 2003

*mengaku membenarkan tesis (LPSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Universiti Malaysia Sabah dengan syarat-syarat kegunaan seperti berikut:-

1. Tesis adalah hakmilik Universiti Malaysia Sabah.
2. Perpustakaan Universiti Malaysia Sabah dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institutsi pengajian tinggi.
4. Sila tandakan (/)

□ SULIT (Mengandungi maklumat yang berdarjah keselamatan atau Kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

□ TERHAD (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

☑ TIDAK TERHAD

Disahkan Oleh

(TANDATANGAN PENULIS)

(TANDATANGAN PUSTAKAWAN)

Alamat Tetap: 355 KG BANGGOL
22100 BDR PERMAISURI
TERENGGANU

AHMAD MUKIFZA HARUN
Nama Penyelia

Tarikh: 04/06/08

Tarikh: ______

CATATAN:- *Potong yang tidak berkenaan.
**Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa /organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT dan TERHAD.
@Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara penyelidikan atau disertai bagi pengajian secara kerja kursus dan Laporan Projek Sarjana Muda (LPSM).

DECLARATION

I declare that this writing is the sole work of mine except for the quotation, summary of references, which I have explained the source of each one of them.

MOHD HAFIZI YUSOF
HK2003-2205
18 APRIL 2008

CERTIFIED BY

Mr. Ahmad Mukifza Harun

(SUPERVISOR)

b/p

Mr. Liawas Barukang

(EXAMINER)

b/p

Mr. Khairul Anuar Mohamad

(CHAIRMAN)

ACKNOWLEDGEMENT

First of all, I would like to thank the most graceful my Lord Allah SWT for His love and guidance that give me strength and confidence to complete my project. Also to my beloved family and friends for their trust, support and patient to me, their prayers kept me going.

I would like to express my deepest gratitude to my supervisor, Mr. Ahmad Mukifza Harun for his guidance, invaluable advice, support and positive encouragement that he has given to me throughout the course of this project.

Thank you to all my friends in UMS especially my course mate for giving their full support and advices to me. This thesis will not be finish and complete without their help.

Thank you very much.

Mohd Hafizi Yusof

ABSTRACT

DIGITAL COUNTER FOR TRAFFIC LIGHT

Traffic light is the one important device where it is using for the street user. There were so many traffic lights with multi function. In this thesis, the digital counter for traffic light to display the desire count up timer is constructing. The C++ language and Visual C++ are using in this term. Every changing the color light has setting timer and it will display the count up number. The important fields which are discussing about counter and software where used for produce the Digital Counter for traffic light.

ABSTRAK

Lampu isyarat merupakan peralatan yang digunakan untuk kemudahan pengguna jalan raya. Terdapat pelbagai jenis lampu isyarat dengan pelbagai fungsi pengoperasian. Dalam tesis ini, lampu isyarat yang memaparkan nombor dengan bilangan masa secara menaik dihasilkan. Penghasilannya melibatkan perisian C++ dan juga Visual C++. Pertukaran setiap warna lampu adalah mengikut ketetapan masa yang telah ditetapkan dan setiap lampu yang menyala akan memaparkan kitaran bilangan nombor secara menaik. Isi utama yang diwacanakan adalah berkaitan dengan pembilang dan juga bahasa perisian yang digunakan untuk menghasilkan isyarat berdigit bagi lampu isyarat.

CONTENTS

TITLE **PAGE**

LIST OF FIGURES

FIGURE **PAGE**

LIST OF TABLES

LIST OF SYMBOLS

>	more than
==	equal
int	integer
char	character
NS	North South
EW	Earth West
1	ON
0	OFF
!	Bar
SOP	Sum-of-Products
X	don't care
N	North
S	South
E	East
W	West

CHAPTER 1

INTRODUCTION

1.1 Overview

This project is to build a program for digital counter to display the timer of traffic light. Before design the program we must know and understand about traffic light, it defined as road signal for directing vehicular traffic by means of colored light, typically red for stop, green for go and yellow for proceed with caution. A traffic light, also known as traffic signal, stop light, or semaphore, is a signaling device positioned at a road intersection, pedestrian crossing, or other location in order to indicate when it is safe to drive, ride, or walk using a universal color code.

A traffic signal is typically controlled by a controller inside a cabinet mounted on a concrete pad. Although some electro-mechanical controllers are still in use, modern traffic controllers are solid state. The cabinet typically contains a power panel, to distribute electrical power in the cabinet, a detector interface panel, to connect to loop detectors and other detectors, detector amplifiers, the controller itself, a conflict monitor unit, flash transfer relays, a police panel, to allow the police to disable the signal, and other components.

Traffic controllers use the concept of phases, which are directions of movement lumped together. For instance, a simple intersection may have two phases: North/South, and East/West like in front of our university. A 4-way intersection with independent control for each direction and each left-turn will have eight phases. Controllers also use *rings*; each ring is an of independent timing sequences. For example, with a dual-ring controller, opposing left-turn arrows may turn red independently, depending on the amount of traffic. Thus, a typical controller is an 8-phase, dual ring control.

Many traffic light systems operate on a timing mechanism that changes the lights after a given interval. A traffic light system senses the presence or absence of vehicles and reacts accordingly. The idea behind digital counter for traffic light is that drivers will not spend unnecessary time waiting for the traffic lights to change. We need to understand the function of traffic signals so that we can improve driving habits by controlling the speed in order to reduce the number of associated traffic accidents.

1.2 Literature Review

From the information that I have been found, the traffic signals have become a valuable tool for ensuring traffic flow throughout the community. The traffic signal must be instructed when to change phase. They can also be coordinated so that the phase changes called for occur in the relationship with nearby signals. Actually, the traffic signal came with more technique construction. Some streets using traffic with sensor, timer, voice, and other function. Traffic signal phase changes are based on one of three systems: pre-timed, semi-actuated, and fully-actuated.

The simplest control system uses a timer (fixed-time): each phase of the signal lasts for a specific duration before the next phase occurs; this pattern repeats itself regardless of traffic. Many older traffic light installations still use these, and timer-based signals are effective in one way grids where it is often possible to coordinate the traffic lights to the posted speed limit. They are however quite disadvantageous when the signal timing of an intersection would profit from being adapted to the dominant flows changing over the time of the day.

Some of the signals are timed to work with other traffic signals to allow continuous flow from one signal to another. Some are timed so that if driver are driving at the posted speed limit, they will receive a green signal at each intersection as they progress down the street. Others are on preset timers where the red, green and yellow indications will cycle for a specified time period. The most important thing is that, this technology needs a new brand idea comes along with combination of methodologies in order to improve the performance of traffic light solution.

The current design of a traffic light system in terms of mechanical, electrical, logic and instrumentation aspects takes full advantage of the application of counter in the real life situation of traffic flow by optimizing the time between light changes. This process goes on in a cycle from the North lane, followed by the East lane, South lane and lastly the West lane.

In a simple traffic light, each state is timed. It is appropriate that our traffic light run continuously and that every state be timed appropriately. The traditional way to make a timer is to set a counter to the number of units to be counted and then count down to zero but in this project we changed the count from zero to up. So it is proposed that a suitable timer can be implemented by making a up-counter which can initialized (loaded) with the number of units to count. For a traffic light, a resolution of one second would appear to be appropriate. I should be noted, however, that if zero is the terminal state, and if the counter is reloaded following zero, then the timed interval will in fact be N + 1 seconds, where N is the number loaded into the down-counter. Accordingly, to make N the number of seconds counted, the terminal state must be a count of one.

1.3 Project Description

The main goal of this project is to build a program of digital counter display for traffic signal, where it counts the timer so that the cycles between green, yellow and red states are display. The project using C++ language to create the coding and Visual C++ is simulated to display a specific time at the traffic light. Hence when the street is normal the counter is set to function timer for green, yellow and red light and when there is detected some error or pedestrian press push button, the counter is set so that there is a increase the display of red light.

The current design of a traffic light system in terms of mechanical, electrical, logic and instrumentation aspects takes full advantage of the application of counter in the real life situation of traffic flow by optimizing the time between light changes.

1.4 The Scope of The Project

The main scope of this project is in Digital electronic because it was count the cycle of rotation based on the traffic light cycle using the timer and counter. However since it designing the traffic light, counter and timer are most involved. This project has five main objectives:

- Understand about traffic light and their system.
- Describe how the counter was count the timer to display as a traffic light system.
- Enhance knowledge on C++ and Visual C++ program and running the digital counter to display time accurately according to traffic condition.
- To make a sequence of the counter to display desired time at the traffic light whether to increase or decrease the time according to the traffic condition.
- Make a good communication between Supervisor and student.

CHAPTER 2

METHODOLOGY

2.0 Introduction

In this chapter, according to the project specification, the counter must count up and it has programmable, that is to say, it should be able to count up from any given value. The justification of the software that will be applied through this project is described briefly. The Visual C++ Version 6.0 software is use to insert and display the output to prove the results as a consul and it is known the experiment was successful. Generally, this project contains two main parts, construct the coding and proving the results. In the other hand, construct the coding is the actual procedure to make an identify the function of traffic light. Therefore, for make functioning of the traffic light, toolboxes from the Visual C++ 6.0 are used to display the output.

The digital counter for traffic light considered a variety of issues. These issues included site specific characteristics such as counter type which is the suitable using where it related to this project. The numbers for count each light for lit such as green yellow and red must be declare. The evaluation process entailed is making a determination whether detailed traffic counts were required (either 45 second for green. 6 seconds for yellow and 60 seconds for red and add 20 seconds for red if have disturbance (pedestrian crossing)). However in my project don't have disturbance or pedestrian crossing.

2.1 Coding Construction

The designing a coding that best represents to set the function for traffic light that means that it requires displaying the counter timer regarding to the traffic flow. Therefore, to create coding for traffic light counter, design schemes a simple algorithm is needed to display the results.

A C program contains *functions* and *variables*. The functions specify the tasks to be performed by the program. The ``main" function establishes the overall logic of the code. It is normally kept short and calls different functions to perform the necessary sub-tasks. All C codes must have a ``main" function. The first statement ``#include < iostream.h>" includes a specification of the C I/O library. All variables in C must be explicitly defined before use: the ``.h" files are by convention ``header files" which contain definitions of variables and functions necessary for the functioning of a program, whether it be in a user-written section of code, or as part of the standard C libraries. The directive ``#include" tells the C compiler to insert the contents of the specified file at that point in the code. The ``< ...>" notation instructs the compiler to look for the file in certain ``standard" system directories. The void proceeding ``main" indicates that main is of ``void" type--that is, it has *no* type associated with it, meaning that it cannot return a result on execution. The ``;" denotes the end of a statement. Blocks of statements are put in braces {...}, as in the definition of functions. All C statements are defined in free format, i.e., with no specified layout or column assignment. White space (tabs or spaces) is never significant, except inside quotes as part of a character string.

The main technical contribution is an exploration of one simple but useful contract, the program counter must count up, where the information of counter which is related to traffic condition to put the values to program the counter at each step of the computation. The coding which is related for the program where it is creates and should be compiling to run. So to make it the functioning, it must be clean from the error. However, notice that can see the entire control flow behavior of the program, so it will run as a role of the count down up.

The Traffic light class is responsible for timing the light changes and signaling the current color. This behavior emulates gates at toll booths - a rather drastic method for enforcing the traffic rules. However, it greatly simplifies the changing of timer. To initialize the timing of the traffic light, we need to know the duration of the three light cycles. For the light to function properly, the duration of the red light should be long than the green and yellow light. Knowing that the yellow light has a fixed duration of the user only needs to specify the duration of green light in two directions.

2.2 Interfacing

The traffic light which is created using the C++ language for make a count number of timer to change each of the light setting and it will be converting to Visual C++ for make an output in console. After output will do, the coding will be assembling to binary format to make interface at memory which is EPROM. It actually is one of the part to make how the functioning of traffic counter when it combine with the hardware.

The PR-875B is a universal programmer which works via a parallel port of your PC, enabling to program, read, copy or check any DIL device with up to 48 pins without the need for adapters. This software accepts more than 3000 different devices, including logic devices (PAL, GAL, CEPAL, PEEL, FPLA, EPLD, FPGA), memories (PROM, EPROM, EEPROM, Flash, and PROM series) and single-chip microcontrollers. The following features stand out from among its characteristics:

- Ultra-fast programming speed

The intelligent control system of the PR-875B reduces the complexity of the system to a minimum. The PR-875B is much faster than its competitors (it only takes 8.5 seconds to program a 1 Mbit EPROM), and so is much more productive with today's high density devices.

- Checking the insertion and contact of the device

The PR-875B carries out a check on the insertion of the device before proceeding to program it. It checks that the device is not badly defined (the actual number of pins differs from that of the device selected), that the insertion is correct (not displaced or inverted), that the connections are correct and that the device is not faulty. This feature acts as a precaution against costly breakdowns caused by human error or faulty contacts, the latter often being due to aged bases, difficult to detect by other means.

➢ Detection of the identifier of EPROM and Flash memories

Many EPROM and Flash memories have a burnt-in device identifier and manufacturer identifier. The PR-875B can read these identifiers with the aim of determining the manufacturer and the reference of the device. This characteristic automates the selection of EPROM and Flash memories and is especially useful in the identification of devices which have their code accidentally (or intentionally) erased.

➢ Automatic programming

In order to satisfy production requirements, the PR-875B incorporates new technologies both in its hardware and in its software. In the Mass Production Mode, the operator inserts a device in the ZIF socket. An LED in the PR-875B indicates when the device has been satisfactorily programmed, and the operator then removes the device and replaces it with another. The ease of this operation eliminates the need for specialized training, saving time and money. The keyboard and the mouse are deactivated in the Mass Production Mode, eliminating the possibility of involuntary errors.

➢ Storage of the working file

The PR-875B allows the saving of the working configuration file, which contains the selected device, the buffer data and all the configuration options of the program. This file can be loaded for future use without the need to reselect the configuration options.

- Auto-increment function

When the devices programmed require individual serial numbers, the PR-875B has an auto-increment function: this function increases the serial number whenever a new device is inserted.

- Programming and checking voltages

The PR-875B provides two checking processes: one process with just VDC checking, or two processes with VDC ±5% and VDC ±10%. This characteristic ensures that the device has been properly programmed, preventing faults due to programming errors and ensuring the storage of the data.

In order to do this project, I have to find some information about the basic for traffic light especially digital counter for timer then study some example circuit to understand how it works so that I will be able to do the designing program, it actually more to imitation but there will be some modification made. After completion of the designing program, comes to the construction how there will be display the results.

CHAPTER 3

DIGITAL COUNTER

3.1.1 Introduction

In this chapter, the overall progress of the project will be discussed and the preliminary results that have been obtained. The method in the previous chapter has been used to complete the analysis and the most important thing we should know about digital counter or digital logic. In digital logic and computing, a counter is a device which stores and sometimes displays the number of times a particular process has occurred, often in relationship to a clock signal. There are two types of counters, up counters which increase or increment in value and down counter which decrease or decrement in value. In this project I only focus on up counter.

3.1.2 Binary Counter

Before starting with counters there is some vital information that needs to be understood. The most important is the fact that since the outputs of a digital can only be in one of two states, it must use a different counting system than to accustom. Normally when using a decimal counting system, meaning that each digit in a number is represented by one of 10 characters (0-9). In a binary system, there can only be two characters, 0 and 1.

When counting up in a decimal system, it started with the first digit. When that digit 'overflows', it means gets above 9, so set it to 0 and add one to the next digit over. The same goes for a binary system. When the count goes above 1, must add one to the next digit over and set the first digit to 0. The table below is shown convert decimal to binary.

Decimal	**Binary**
0	000
1	001
2	010
3	011
4	100
5	101
......	......

Table 3.1: Decimal to Binary

To convert a binary number to a decimal, we use a simple system. Each digit or 'bit' of the binary number represents a power of two. All here, need to do to convert from binary to decimal is add up the applicable powers of 2. Here, the binary number 10110111 is equal to 183. The diagram also shows that eight bits make up what is called a byte. Nibbles are the upper or lower four bits of that byte. Referring to nibbles and bytes are useful when dealing with other number systems such as hexadecimal which a base 16 instead of base 10 or base two.

BYTE								
Upper Nibble				Lower Nibble				
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Binary number
1	0	1	1	0	1	1	1	
2^7 +	2^6 +	2^5 +	2^4 +	2^3 +	2^2 +	2^1 +	2^0	**=183**

Table 3.2: Binary number

3.2 COUNTER

According to the project description, the counter must count up and it has to be programmable, that is to say, it should be able to count down from given input value. Here, I used basic counter to introduce to obtain the required of the project.

To generate a traffic light sequence, here, for the traffic light, I prefer to use the down counter because for its binary contents decrease as the number of pulses increases. For example, a typically sequence of counts in three-stage up counter 001, 010, 011, 100, 101, 110,111,.... One use of up counters is in applications where it is necessary to display the amount of time until the execution of some event. A necessary state diagram is shown below.

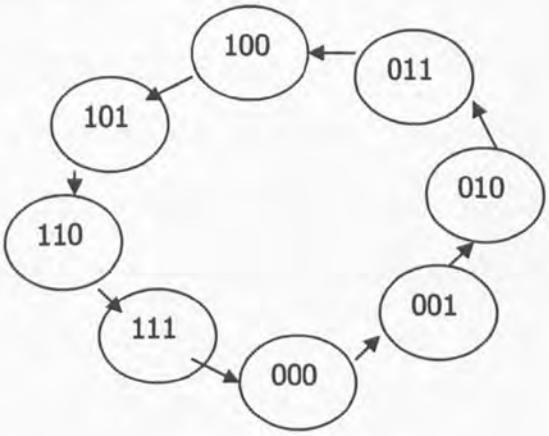

Figure 3.1: State diagram for up counter

From the state diagram above, the next state table or true table is derived and is shown in table below.

REFERENCES

1. Theodore F Bogart, Jr. 1992. *Introduction to Digital Circuit.* Glenceo Division of Macmillian/McGraw – Hill Publishing Company.

2. Thomas L. Floyd. 2003. *Digital Fundamental.* Eight Edition, International Edition, prentice Hall.

3 Ivor Horton's. 1998. *Beginning Visual C++ 6.* Wrox Press.

4. Steven C. Chapra, Raymond P. Canale. International Edition 2006. *Numerical Methods for Engineering.* Fifth Edition. Mc Graw – Hill.

5.Goldberg, Bar-Giora.1996. *Digital Techniques in Frequency Synthesis.* New York: McGraw-Hill.

6. Philip Daum, Brent Fukuda, Nick Jensen, 4th year students, Garwin Hancock and Carmen Ciubotariu. *Virtual Versus Real Design Of A Traffic Light/Voice Controller.* Department of Electrical and Computer Engineering, University of Calgary. Calgary, Alberta, Canada.

7. . Howard Boyet, Ron Katz. 1997. *MA-2, Microcomputer Applications: Applications, Experiments with an 8080.* A&L Instruments.

8. Roland E. Thomas & Albert J. Rosa, Wiley. *The Analysis and Design of Linear Circuit.* Fourth edition.

9. Stephen Randy Davis. 2000. *C++ for Dummies.* 4th Edition. IDG Books Worldwide.

10. H S Kalsi 1995. *Electronic Instrumentation.* Second Edition. McGraw- Hill.

11. Michael J. Young. 1998. *Mastering Visual C++ 6.* Sybex.

12. . Joanne Penner Ricardo Hoar Christian Jacob. *Swarm-based Traffis Simulation With Evaluationaty Traffic Light Adaptation.* Department of Computer Science, University of Calgary 2500 University Drive N.W, Calgary, T2N 1N4, CANADA..

13. William Schweber, Schweber. *Electronic Communication Systems*. Third Edition.

14. Tim Williams, Elsevier. *The Circuit Designer's Companion.* Second Edition.

15. Viera K. Proulx. *Traffic Simulation: A Case Study for Teaching Object Oriented Design.* Northeastern University. College of Computer Science. Boston, MA 02115, USA.

16. Avinson, John. 2001. *The World of Physics.* 2nd Editions. Melbourne, Australia

17. Tierney, J., Rader, C.M., and Gold, B. A Digital Frequency Synthesizer. *IEEE Transactions on Audio and Electroacoustics* **AU-19**:1, March 1971, 48-56.

18. http://en.wikipedia.org/wiki/Direct_digital_synthesis

19. www.physicsforums.com/archive/index.php/t-55833.htmm

20. www.ibiblio.org/kuphaldt/socratic/output/proj_ctrt.pdf

21. www.sunrom.com/s-seven-segemnt-display-interfacing.html

22. http://ieeexplore.ieee.org