VHDL Modelling of Traffic Signal Controller

AbobkrHassen.R. Mohamed
Higher Institute of Sciences and Technology Aljoufra,
Electronic Engineering department, Zella, Libya
abobkr.mohamed316@gmail.com

Abstract-Recently, the electronic devices enter our daily life in many places, to make it more convenient. However, the aim of this work is to model, the design and test traffic signal controller using VHDL, which can be easily used to describe and simulate the operation of digital circuits. Furthermore, it has been utilized commonly in industry since it became an IEEE standard. The traffic signal controller has two traffic lights, and each traffic light has three lights red, green and amber as outputs. Also it has two sensors as inputs; additionally, initialisation signal to reset the controller to the first state of the controller. The output of the controller changes according to the input sensory data received. The design done implemented by using Altera Max Plus II software package; in addition to that, it done inspected and verified by simulation waveform the work done verified by using a simulation waveform in order to make sure that all of the design requirement is satisfied.

I. Introduction

The monitoring and control of traffic congestion have been an issue in many cities around the world. Even before automobiles, the movement of horses, buggies and pedestrians had been controlled for safe and orderly, by police officers, who wore white gloves and blew whistles [1]. Subsequently, a traffic light controller was created to solve the problem. Traffic lights are electronic devices used to control and organise the movement of vehicles and pedestrians, in order to keep vehicles and pedestrians safe[2].

In General, a traffic signal system has three lights are controlled by using microcontroller [2].

In traditional traffic lights, incandescent halogen bulbs were used; however, currently they are being replaced with LED traffic lights. This is because the LEDs are brighter,and consumes less energy than the incandescent bulbs and has longer life span[2]. However, some traffic light controllers include other components such as vehicle detection systems, inductive loops or sensors[3].

The traffic signals in a number of cities over the world can basically operate in three ways as follows:

1. Fixed controller: In this kind of traffic, the signal cycles continuously, and the period for each cycle is constant regardless of real traffic desire.
2. Semi- actuated: In this sort of traffic controller, it has detection system; however, it is for minor cross Street only and the major street is not.
3. Actuated controller: In this type of the traffic controller, there is detection system for each road. It provides green light on demand, or only when the vehicles are detected.[4]

II. Electronic design automation (EDA)

Electronic design automation (EDA) is plainly identified as a package of software tools used in the design **of** electronic devices such as printed circuit boards and integrated circuits (ICs). There are many languages that are used in this field like the VHDL and virology VDL. In 1950s and 1960s electronic circuits were designed by hand and laid out manually. However, large chips or systems are too complicated to be designed by hand; thus, EDA were invented. The usage of EDA programs has been gradually increased by the modern industries; as a result of incessant requirement of semiconductor technology[4]. Additionally, the EDA software tools aid the design and evaluation of the complex digital circuits within thebuilt-in computer work station, without the need of the hardware simulation. Moreover, EDA programs can detect the errors simply, without going through the costly stage of prototype construction. [5,6].

III. Definition of VHDL

HSIC is an abbreviation for Very High Speed Integrated Circuit. VHDL is an acronym for VHSIC Hardware Description Language. It is a computer programming language used to model, synthesis and describe the behaviour of electronic circuits [5], also the performance can be described in many electronic components in many areas such as simple logic gates and microprocessors by the VHDL. Furthermore, it can be utilized to depict

different level of abstraction. It is as C, which is used to describe, model and implement a solution to a problem [12]; moreover, it looks like Java, it can be easily designed a circuit before knowing which sort of device it will be implemented on [7].

IV. The specification of the traffic signal controller

It is required to design traffic signal controller with certain specifications, in order to process a problem in many cities around the world, sometimes occurs in rush hour traffic. In intersections, people who are at the traffic have to wait at the traffic lights a period of time until the ending of red light's cycle, and then they acquire opportunity to pass it (during the cycle of green light). Even though there are no cars are coming from the other way, people have to wait; as a result of that, they waste their time. Therefore, the VHDL will be used to model a traffic signal controller, to demonstrate the design technique of a real digital system. However, it is particular for traffic at intersection of two roads, and each road is only one-way as shown in fig.(1) beneath.[8,11]

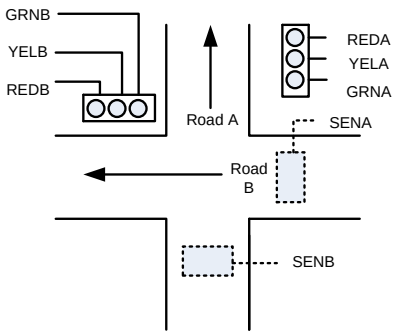

Figure (1)the intersection, where the traffic light can be installed

This unique traffic light consists of two traffic lights one traffic light for each one –way street (traffic A and traffic B) , and two sensors as inputs (SEN A and SEN B). The SENA is linked to REDA, GRNA and YELA. And SENB is connected to REDB, GRNB and YELB. The purpose of the controller is to let the traffic passes one way and stop the other way as the associated sensors are activated by cars.

The inputs to the controller are the SENA, SENB and an initialization signal to make sure that the lights are synchronized in the proper sequence to start with. And six outputs,three for each traffic light, which are REDA, GRNA, YELA, REDB, GRNB and YELB.

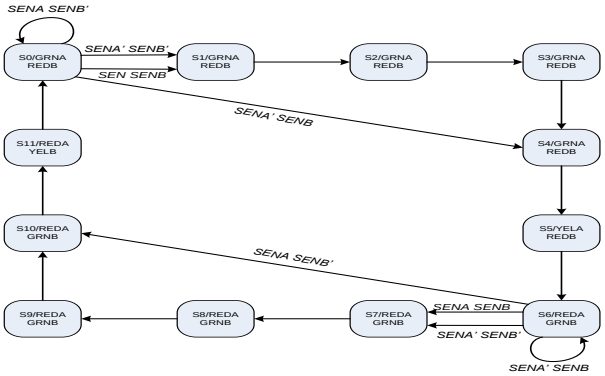

Figure (2) The state transition of the traffic signal controller.

V. Results and Discussion of traffic signal controller

Following to the code, which describes the operation of the traffic controller with required specifications, compiling by compiler to make sure that the code is written correctly(see appendix A). seven waveforms have been achieved, to represent and cover all the required state movement of the controller and initialisation action, in accordance with state transition. The rising edge of the clock at each 80 ns moves the states of the controller to all the needed states according to the state table [9,10].

In the graph (3) the initialisation action moved the controller from State (s11) to state (s0). However; this transition state is not a necessary movement and it does not include in the movements of state table or in the required movements, which demonstrate the behaviour of the controller, due to the initialisation action condition. Moreover, in simulation result, the initialisation is at the beginning to make sure that the program starts from first state (s0); additionally, the initialisation can be used at the end of states to make sure that the program will restart from the first state immediately.

Figure (3) illustrates the state transition diagram to describe the relationship between the input sensory data and the output signals. And also state transition table, which is according to the state transition diagram, is shown in fig.(2)[10].

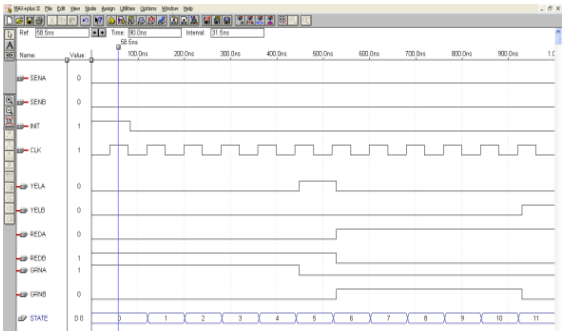

Figure (3)the state movement of the controller when SENA and SENB are (0) on S0 and S6, and initialisation signal is (1)

As it can be clearly seen from the above graph that, we could achieve the first condition, which is reset of the controller to state S0 by initialisation signal. Furthermore, the second condition has been achieved in graph (3), which is the once the SENA, SENB are not activated by cars in state S0 and also in state S6, the movement of states is respectively starts from state S0 to state S11. The output of traffic light A is from state S0 to S4 the GRN A is (1) and the other lights are (0), in state S5 the YELA is (1) and the other lights are (0) and from state S6 to S11 the REDA is (1) and others lights are (0).similarly, the output of traffic light B is from state S0 to S5 the REDB is (1) and other lights are (0), from state S6 to state S10 the GRN b is (1) and the other lights are (0) and the output of the YEL B in state S11 is (1) and the other lights are (0). Moreover, GRN A and GRN B took long cycles, which are four clock cycles as mentioned in the required specification.

The graph (4) below illustrates the output of the controller when the value of San A and SENB is (1) in the state S0 (when the output of the controller GRN A and RED B is (1)). In this case, the output of the controller is exactly same as the output of the controller once the value of SENA and SENB is (0) as in above graph at the same state.

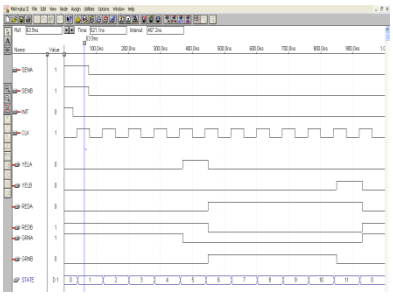

Figure (4) controller state movement of once SENA and SENB are (1) on state S0.

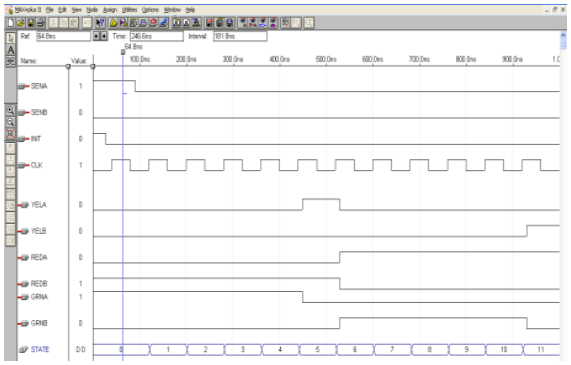

Figure (5) the controller output when only SENA is (1) on state S0

From the above graph it can be shown that, the controller stayed in state S0 because the value of the SENA is (1), and SENB is (0) at the same time with clock rising edge; therefore, the controller will stay at state S0 until the input of the controller changes before the rising adage coming again.

The graph (6) is shown below demonstrates the movement of the controller states form state S0 to state S4 when SENA is (0) and SENB is (1). In this case the controller will take short cycle for GRN A, which is one clock cycle, to give priority for GRN B.

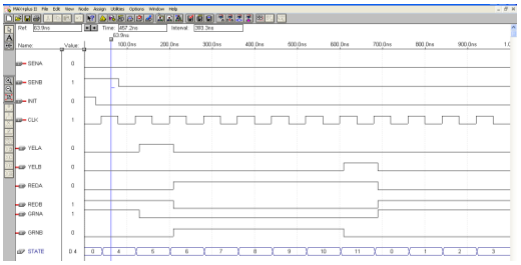

Figure (6) the controller out if SENB is (1) on state S0.

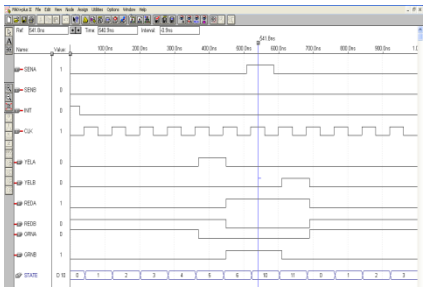

Figure (7) state movement of the controller form state S6 when SENA is (1).

In above graph (7) it can be obviously seen the behaviour of the controller once SENA is (1), and SENB is (0) at the state S6. The controller moved from state S6 to state S (10), which also took short cycle for GRN B.
The beneath graph shows that, the controller state moves from state S0 to S11 respectively even the value of SENA and SENB is (1) at the state S6 and the output of the controller is such as graphs (3 ,4)

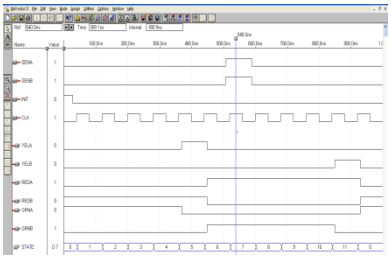

Figure (8) the state movement of the controller from state S6 once SENA, SENB are (1).

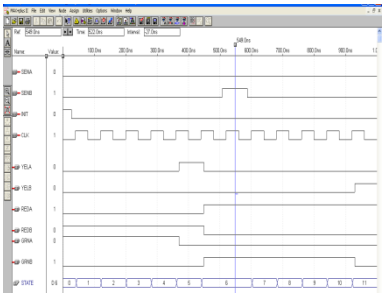

Figure (9) the behaviour of the controller when SENB is (1) on state S6.

As it can be clearly exposed in above graph at state S6, the controller stayed at state S6 because the clock rising edge came at the same time once SENA was (0), SENB was (1) and it would be at state S6 until the controller input is changed, more details is in appendix B. From the above graphs it can be observed that the actions of the controller at state S0, and at state S6 are exactly the same. In addition to that, all the previous graphs present the values of the inputs and the outputs according to the transition table. Thus, all the required specifications have been achieved.Since any logic gate has a time delay between its inputs and output. By using timing analyser, which is one of Max-Plus II feature, we found the time delay inabove graphs, which describe the behaviour of the controller between clock andeach output, is 9.5ns as bellow . [7,8,11,12]

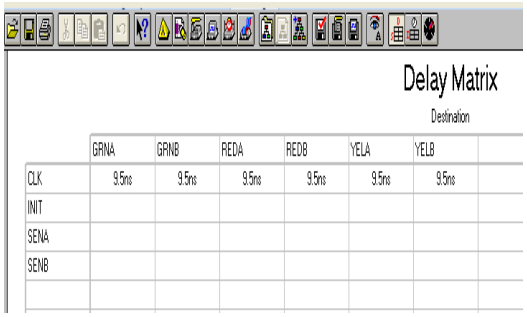

Figure (10) shows the time delay between the clock and each output of traffic controller.

VI. Conclusion and future work

The traffic signal controller has been properly designed.
It can be clearly illustrated that the, design has been carried out with the highest level of the abstraction, which describes the behaviour of the design instead of elucidating its components and interconnection between them.
-it can be realised that all controller is acted in a fixed time. However, it stays on state S0 if the value of SENA is greater than the SENB value. Also on state S6 when the value of SENA is smaller than SENB value until their values are changed. Similarly, the improved controller stays on state S0 once SENA value is (1) and other inputs are (0), and it also stays on state S7 when SENB is (1) and the others are (0).
-The VHDL code, which depicts the traffic signal controller, is reusable because after alteration the number of its inputs, outputs and states, it becomes a device with high efficiency in controlling the traffic.
- In the future work, it is possible to study how the average of the waiting time in intersections can be reduced. In specific, to improve the performance of the controller; particularly, to make it appropriate for using in intersections, where the roads are two- way and allowing the cars to turn left or right, for making the controller is close to reality.

References:

1-History of traffic signal design, www.kbrhorse.net, available at:http://www.kbrhorse.net/signals/history01.html, (accessed 03/03/2017).
2- 2010, traffic lights, Ezine articles, available at:http://ezinearticles.com/?Traffic-Lights&id=409709, (accessed 04/03/2017).
3- Anguse p. Davol, 1997, Modeling of Traffic Signal Control and Transit Signal Priority Strategies in a Microscopic Simulation Laboratory, web.mit.edu, Available at: http://web.mit.edu/its/papers/TRAFF.PDF , (accessed 06/03/2017).
4- Wang.L, chang.W, ching.K, 2009, Electronic design automation :synthesis, verification, and test, google books, available at:http://books.google.com/books?id=3XBe7dLb

5NEC&printsec=frontcover&dq=inauthor:%22Kwang-Ting+Cheng%22&hl=en&ei=D0GATLHMBZSOjAfk_vF6&sa=X&oi=book_result&ct=book-thumbnail&resnum=4&ved=0CDsQ6wEwAw#v=onepage&q&f=false, (accessed 08/03/2017).

5- Pedroni .V, 2004, Circuit Design with VHDL, USA.

6- Birnbaum.M, 2004, essential electronic design automation (EDA), google books, available at: http://books.google.com/books?id=NYnphhDUN5cC&printsec=frontcover&dq=essential+electronic+design+automation&hl=en&ei=PEKATO-8KJbKjAexktXqAQ&sa=X&oi=book_result&ct=book-thumbnail&resnum=1&ved=0CDMQ6wEwAA#v=onepage&q&f=false, (accessed 08/03/2017).

7- Chang, K, 1997, Digital Design and Modeling with VHDL and Synthesis, IEEE Computer Society Press, USA.

8- 2004, Accolade VHDL reference guide, www. acc-eda.com, available at: **http://ece.wpi.edu/~wrm/Courses/EE3810/geninfo/Welcome%20to%20the%20VHDL%20Language.pdf**, (accessed 08/03/2017).

9- Ashenden .J, 2004, VHDL tutorial, www.ashenden.com.au [online]. Available at: http://www.ashenden.com.au/ashenden-bio.html, (Accessed 15/08/ 2010)

10- AU.A, Tyan.V,Supmonchai.B,Szymanski.T, 1999, MAX+PLUS II Tutorial Manual, availble at : http://ee.sharif.edu/~logiclab/Maxplus2_Tut_v3.0.pdf , (accessed 10/03/2017).

11- Roth.C , John.J, 2008, Digital system design using VHDL, USA.

12- Wakerly. J, 2002, Digital Design Principle& Practices, 3rd edition, Tom Robbins, USA.