Implementation of Algorithm for Vehicle Anti-Collision Alert System in FPGA

Aiman Zakwan Jidin, Lim Siau Li, Ahmad Fauzan Kadmin
Faculty of Engineering Technology, Universiti Teknikal Malaysia Melaka, Malaysia

Article Info	ABSTRACT
	Vehicle safety has becoming one of the important issues nowadays, due to the fact the number of road accidents, which cause injuries, deaths and also damages, keeps on increasing. One of the main factors which contribute to these accidents are human's lack of awareness and also carelessness. This paper presents the development and implementation of an algorithm to be utilized for vehicle anti-collision alert system, which may be useful to reduce the occurrence of accidents. This algorithm, which is to be deployed with the front sensors of the vehicle, is capable of alerting any occurrence of sudden slowing or static vehicles ahead, by sensing the rate of distance change. Furthermore, it also triggers an alert if the driver is breaching the safe distance from the vehicle ahead. This algorithm has been successfully implemented in Altera DE0 FPGA and its functionality was validated via hardware experimental tests.

Corresponding Author:

Aiman Zakwan Jidin,
Faculty of Engineering Technology,
Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
Email: aimanzakwan@utem.edu.my

1.   INTRODUCTION

Nowadays, we have seen an increase in the number of traffic accidents. For example, in Malaysia, there were about 65,883 accident cases on Malaysian roads involving car drivers and motorcyclists which were recorded in 2014. This number is in fact 5.4 percent higher than the 62,519 cases recorded in the previous year [1]. Traffic accidents occur due to several reasons, such as the heavy traffic and weather conditions, vehicle and road conditions, and notably the human error [2].

According to an article in [3], there were 90% of road accidents were caused by bad driving behavior like driving too fast and recklessly, driving under alcohol influence, changing the lanes without signaling, and breaching the safe distance between two vehicles. The latter may cause a severe accident especially when a vehicle ahead brake suddenly, where the driver unable to brake in time and thus, the collision may be inevitable. Besides, many accidents also occur due to the driver's failure to recognize the danger. For example, they may not notice when a vehicle in front of them is suddenly braking or changing to their lanes. Sometimes, their driving awareness may be degraded when they have been driving for too long for their journey, due to sleepiness and fatigue factors. Therefore, an alert system is required to alert the driver with a warning when the system detects that there is a possibility of collision and also to remind the driver to keep a safe distance with the vehicle ahead.

There were many ideas and researches which have been proposed to reduce the occurrence of road accidents especially accidents caused by human error, such as an anti-collision system with BLINDER laser detector [4], an anti-collision and intercommunication system using communication protocol [5] and an anti-collision system using electromagnet and ultrasonic sensor [6]. While the article in [7] proposed the implementation of vehicle front sensor system, which is based on Arduin, by using the ultrasonic sensor.

Most of these anti-collision systems were controlled by using the microcontroller. However, for an application or a system which need a very high processing speed, the microcontroller can be replaced by a higher performance device like FPGA, due to the former's inability to perform very fast computations since all of the instructions are executed sequentially inside the microcontrollers [8], [9].

Field programmable gate array (FPGA) is a digital integrated circuit that is designed to perform a variety of tasks such as combinational functions, or simply logic. It can be configured or reprogrammed by customers or designer engineers after manufacturing [10]. FPGA hardware design is increasingly popular in applications like graphic processing [11], digital signal filtering [12], cryptography [13], and also the electric motor drives [8], [9]. This is owing to its flexibility, very fast computation time with low latency, low power consumption and its continual improvement of the technology, which make it even better in the future.

The need for a very fast computation is essential for this system because the slower the system detects a possible collision, the later the driver will react and thus, the effect of collision will be more severe. For example, if two cars, which are initially 30 m from each other, are moving fast at 100 km/h on a highway. Then, imagine if the car at the front suddenly slows down to 70 km/h. Within one second, if the driver of the second car doesn't apply any braking, he can find himself just 3 m behind the car at the front.

This paper presents the development of an algorithm for the vehicle anti-collision alert system, which is implemented in FPGA. The proposed system is able to detect a possibility of collision by detecting the rate of change of distance between two vehicles and thus, triggering the alert. Furthermore, it also capable of alerting the driver in the case where the safe distance has been breached. The process of detection and notification is done within a very short time, owing to FPGA hardware ability to perform a very fast computation. For the system test purpose, an ultrasonic range sensor is utilized in order to detect the distance between two vehicles.

2.   RESEARCH METHOD
2.1. System Architecture

Figure 1 shows the block diagram of the system developed. As can be seen from this figure, it consists of two main subcomponents: detector and comparator. The detector initiates the detection of the existence of any objects ahead, whereas the comparator will use the data received from the detector in order to determine if there are any occurrence of slowing vehicles or static objects ahead, or the safe distance between two vehicles has been breached.
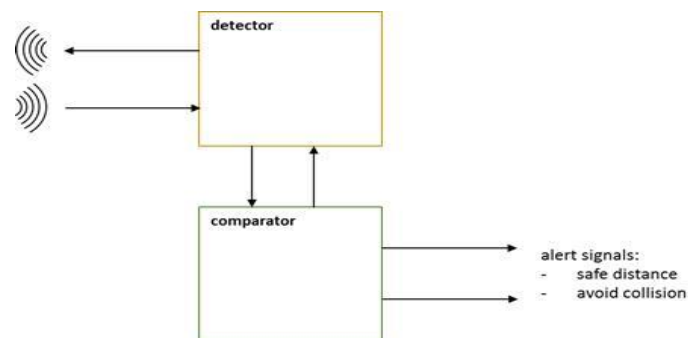

Figure 1. The block diagram of the proposed vehicle anti-collision alert system

In this research, for testing purpose, the detector deployed the ultrasonic sensor range detection mechanism. An ultrasonic sensor sends out a sonic burst at 40 KHz and its velocity is the same as the speed of sound, which is approximately 340 ms^{-1}. This signal will be reflected back by the object ahead and thus, the signal traveling time, which is the time interval between the trigger signal is sent and the echo signal is received, is calculated. Therefore, the distance (in meter) between two objects can be determined by using the following formula:

distance = travelling time * 340 / 2

In the calculation, the traveling time needs to be divided by 2, since the traveling time includes the time it took from the ultrasonic sensor to reach the object ahead and also the time the echo signal needed to reach back at the sensor.

2.2. Detection and Comparison Process

The states machine in Figure 2 illustrates the detection mechanism which has been operated by the detector. It consists of five different states: IDLE, INIT, WAIT, RECEIVE, and STOP. In IDLE state, the machine is waiting for the comparator to be ready to acquire a new comparison first, which is indicated by **the compare_done** flag. Next, in INIT state, the detector will send a high-level trigger signal to the ultrasonic sensor, in order to initiate the sonic burst. This trigger signal is maintained at high level for at least 10 µs. After the WAIT state, the sonic burst should be ready and transmitted. During the RECEIVE state, there is a counter inside the detector which counts the number of clock cycles between the moment the sonic burst is transmitted and the time when the sensor receives the echo (when **echo** flag is becoming low). However, in the case of the inexistence of any objects ahead, it will directly transit to the STOP state but no comparison will be performed afterward.

Finally, during the STOP state, the counter value is sent to the comparator together with a **cnt_done** flag, indicating that the counting process is completed and the comparison can be commenced. Before returning back to the IDLE state, the detector will stay in the same state until the **end_60ms** flag is high, signaling that the detector has reached its complete measurement cycle, which is set at 60 ms. This measurement cycle period, which is the time interval between two consecutive trigger signal, had been recommended by the sensor manufacturer, in order to prevent trigger signal to the echo signal [14].
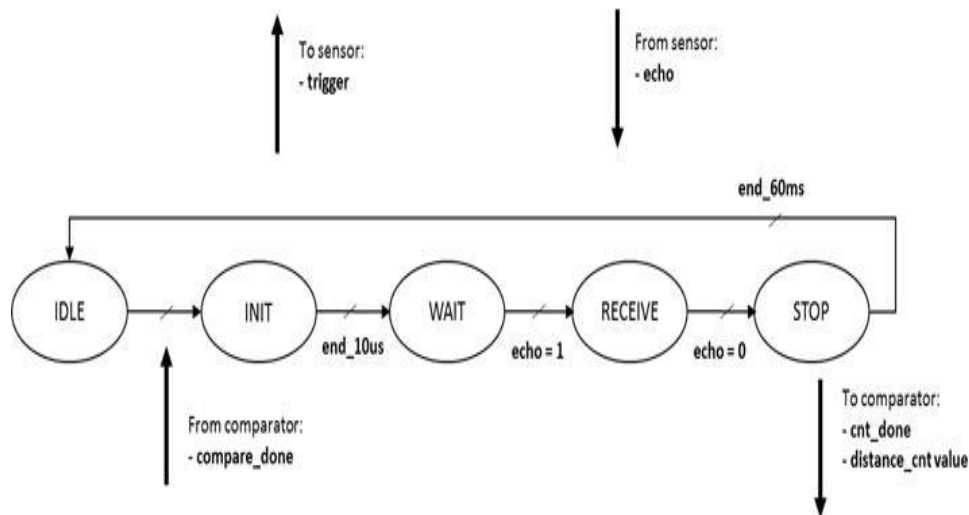

Figure 2. States machine which represent the detector's operation

Figure 3 shows the states machine which represents the operations executed by the comparator. Initially, it remains in the IDLE state until it receives a high-level **cnt_done** flag, which indicates that the detection process is done in the detector. Next, in the READ state, the new distance value counted inside the detector is updated inside the comparator. At this stage, this newly updated value is compared to the distance value from the previous measurement cycle. At the COMPARE state, it will decide to perform one of the following actions:
a. If the difference between the new distance value and the previous distance value is significant, then it will transit to TRIGGER_COLL in order to activate the collision alert
b. If the difference between the new distance value and the previous distance value is small and the collision alert is currently activated, then it will transmit to UNTRIG_COLL state to deactivate the collision alert
c. If the new distance value is greater than the safe distance threshold, then it will transmit to the TRIGGER_PROX state in order to activate the safe distance alert
d. If the new distance value is lesser than the safe distance threshold and the safe distance alert is currently activated, then it will transit to the UNTRIG_PROX state to deactivate the safe distance alert
e. If none of the above cases, it will directly transit to the WAIT state.

After the WAIT state, it will return back to the IDLE state. At this stage, the comparator sends a compare_done flag to the detector, in order to indicate the latter that the comparison is completed and it may now proceed with the new detection.

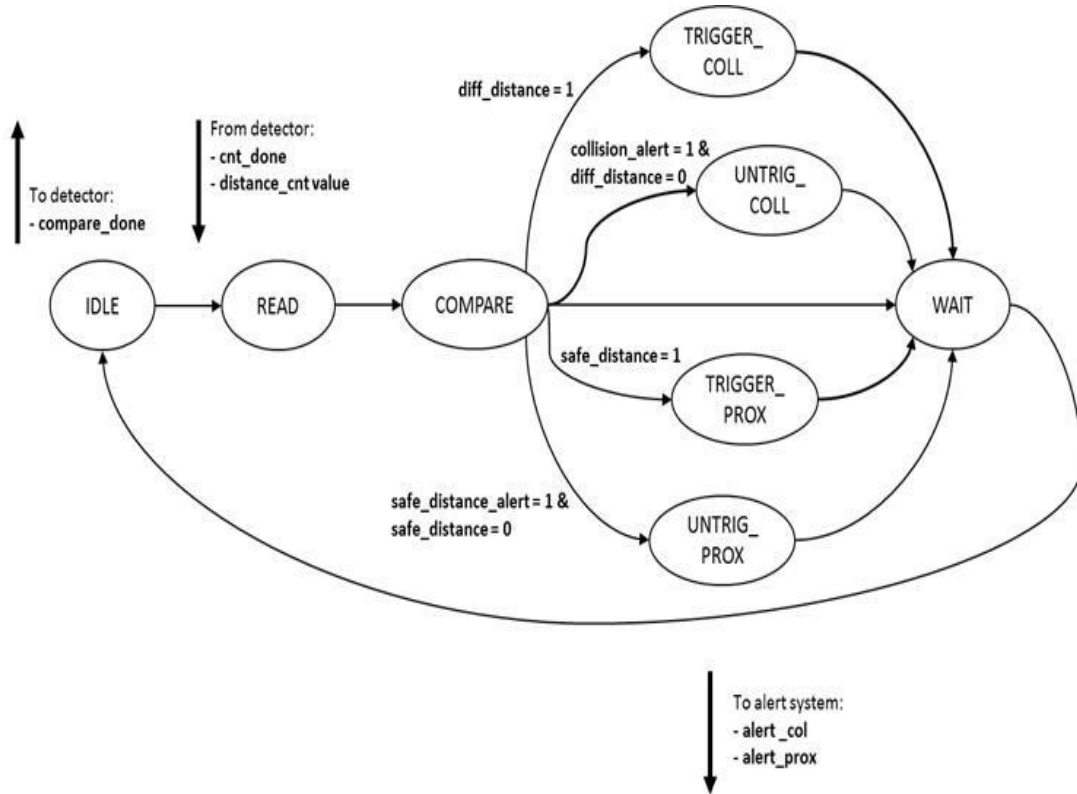Figure 3. States machine which represent the comparator's operation

2.3. Parameters Setting

As the matter of fact, there are several parameters which need to be set for this system. For example, in the detector, a parameter called MAX_CNT is used to set the maximum count value by the counter; beyond this value, we can assume that there are no objects or vehicles which exist in front of us. If the maximum detection is set at d (in meter), the MAX_CNT shall be set equal to $2d / (340 * T_{clk})$, where Tclk is the period of the clock utilized for the counter. For example, if d = 10 m and T_{clk} = 20 ns, then:

$$MAX\_CNT = (2*10) / (340 * 20x10^{-9}) \approx 2941176.$$

In the comparator, there are two parameters that need to be determined, which are SAFE_DISTANCE_THOLD and DIFF_DISTANCE_THOLD. The former is the safe distance threshold for the counted distance value; if the counted value is lesser than the threshold, it indicates that the safe distance is breached. While the latter is the minimum difference between the current distance value and the previous value. If their differences are bigger than the threshold value, then the system will trigger the collision alert.

Both of these parameters can be determined by using the same formula as the MAX_CNT determination. For example, if the safe distance is fixed at 2 m, then

$$SAFE\_DISTANCE\_THOLD = (2*2) / (340 * 20x10^{-9}) \approx 588235.$$

Meanwhile, if the minimum distance count difference is set at 0.5 m every measurement cycle, then

$$DIFF\_DISTANCE\_THOLD = (0.5*2) / (340 * 20x10^{-9}) \approx 147059$$

3.   RESULTS AND ANALYSIS

The proposed algorithm was successfully implemented on Altera DE0 Development Board, which equipped with Cyclone III FPGA Device. It had been designed by using Verilog HDL code. Figure 4 shows the proposed system's hardware test setup. It can be seen that the designed system in FPGA is connected to the HC-SR04 ultrasonic sensor via input-output ports. Moreover, a ruler is put at the sensor, in order to easily fix the distance between the sensor and the obstacle during the test.
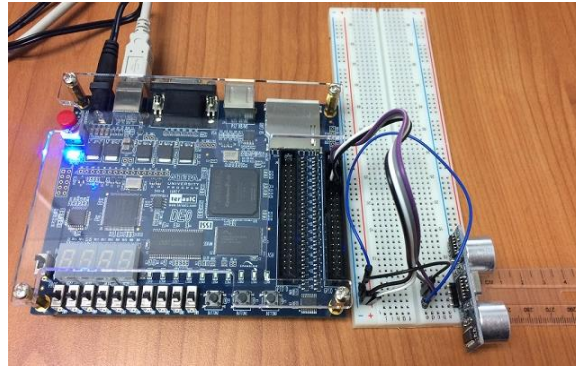
Figure 4. FPGA hardware test setup

For the test in the laboratory purpose, the safe distance threshold is 5 cm while the distance difference threshold is set to 2 cm. Therefore, SAFE_DISTANCE_THOLD and DIFF_DISTANCE_THOLD are fixed at 14700 and 5882, respectively. Meanwhile, the maximum detection range is set at 30 cm (MAX_CNT = 88235).

3.1. Hardware Experimental Results

Figure 5 shows an obstacle, which represents a vehicle in front, is set at a distance of 7 cm from the sensor. The result of the detection in FPGA is observed in the SignalTap II window, as shown in Figure 6. In this case, the detector counted a value equal to 20520, which is corresponding to 7 cm, approximately. As can be seen here, the **alert_prox** signal remains LOW, as expected since the distance is greater than the safe distance set.
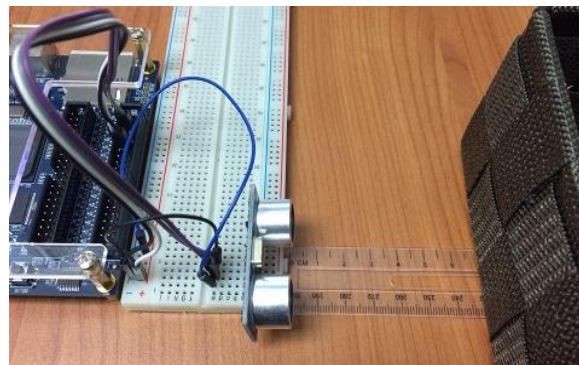

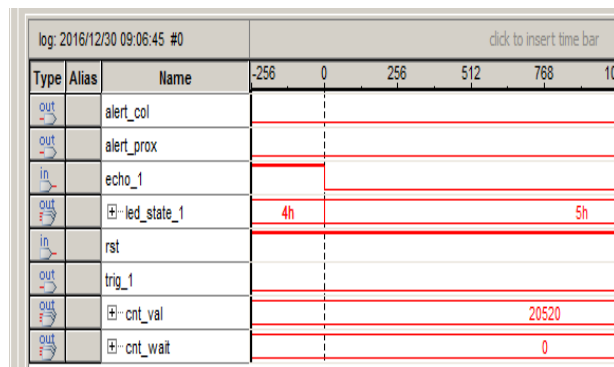Figure 5. An obstacle is set at 7 cm from the sensor

Figure 6. Waveforms observed in SignalTap II window when an obstacle is set at 7 cm from the sensor

Meanwhile, Figure 7 shows that the obstacle had been moved closer to the sensor, within a distance of 4.5 cm. The counter value produced inside the detector is 13432, which is corresponding to a distance equal to 4.5 cm, approximately. In this case, since the distance is smaller than the safe distance set, then the **alert_prox** signal is set to HIGH, as can be observed in Figure 8.
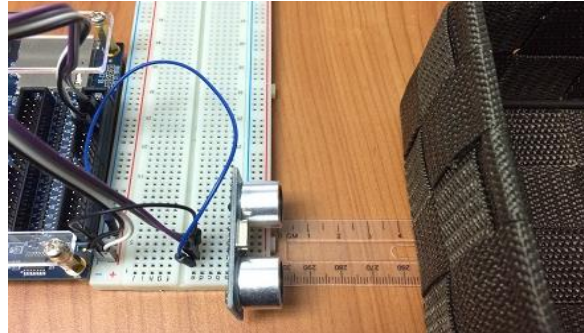

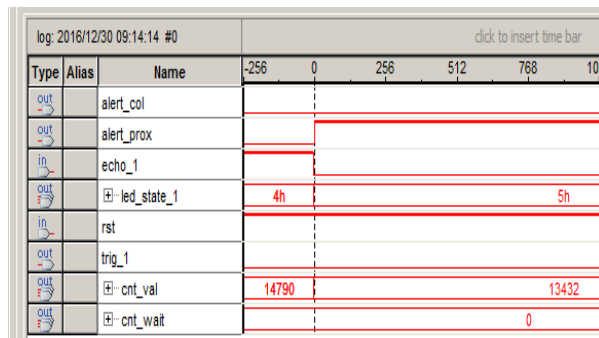Figure 7. An obstacle is set at 4.5 cm from the sensor

Figure 8. Waveforms observed in SignalTap II window when an obstacle is set at 4.5 cm from the sensor

Figure 9 shows the results observed in the SignalTap window when the obstacle ahead had been moved very quickly and thus, its distance from the sensor was drastically changed between two consecutive measurements. As can be seen from the results shown, the distance was reduced from 9.6 cm (corresponding to cnt_val =28348) to 6.7 cm (corresponding to cnt_val =19907). It means that the distance between the obstacle and the sensor was reduced by 2.9 cm within two consecutive measurements. Consequently, the **alert_col** signal is set to HIGH in order to trigger the anti-collision alert.
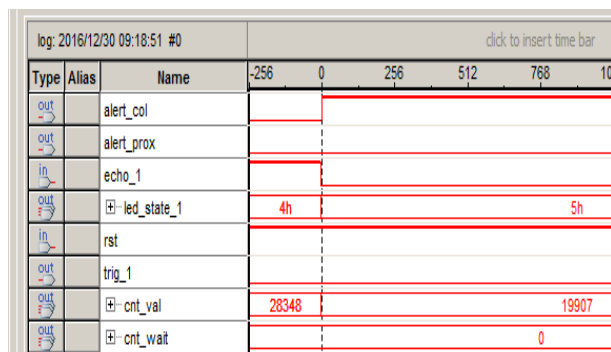

Figure 9. Waveforms observed in SignalTap II window when the distance between the sensor and the obstacle changed drastically

In Figure 10, it shows that the system took only 5 clock cycles from the reception of the echo signal has been completed to the alert_col signal is triggered. Since the 50 MHz clock is used for this test (clock period is equal to 20 ns), the time taken for this detection is 100 ns.
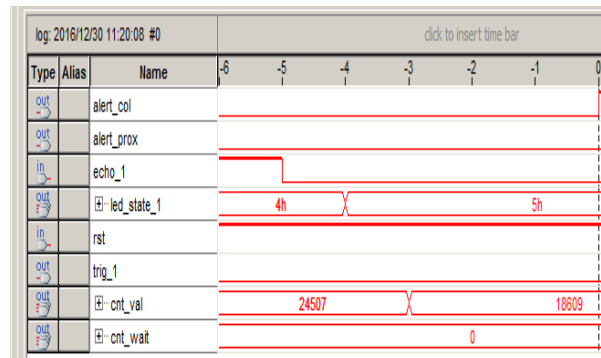

Figure 10. Waveforms observed in SignalTap II window showing the number of clock cycles it took to trigger the collision alert

When comparing the performance of the proposed system with the performance of the microcontroller-based system [7], it clearly shows that the detection of the proposed system is almost 400 time faster than the detection of the latter, which was measured at 40 us. This performance can be significantly improved if the usage of the ultrasonic sensor is replaced by the laser range detector due to the greater speed of light, as proposed in [4].

3.2. Design Performance Analysis

Table 1 shows the hardware resources utilization and also the analysis on the timing and power consumption of the implemented algorithm in FPGA. These data are obtained from the Quartus II compilation report, while the timing performance and the power dissipation are extracted from the TimeQuest timing analysis report and PowerPlay Analyzer report, respectively.

Table 1. The overall performance of the implemented system in FPGA

Logic Elements	Fmax (MHz)	Power Dissipation (mW)
1399	72.4	128.93

Based on the figures in Table 1, the whole system only consumed 1399 from the total 15408 logic elements which are available in the FPGA used. In terms of the timing performance, the maximum clock frequency which can be used to operate this system is 72.4 MHz. Thus, the computation time of the proposed system can be improved by replacing the 50 MHz clock with a faster clock, which frequency is up to the stated Fmax.

In fact, the power consumption is not the main focus of this paper. However, it can be improved in the future by implementing several power management techniques such as the clock gating, or by changing the power-optimized state machine encoding like Gray encoding instead of Binary encoding.

4. CONCLUSION

This paper has presented an implementation of the algorithm for vehicle anti-collision alert system, which had been successfully implemented in Altera DE0 FPGA Board. The proposed system is capable of detecting a decelerating or even a static object ahead and thus, alerting a possible collision. Besides, it also has the ability to remind drivers to keep the distance between two vehicles within a safe range. The experimental tests were performed by using the targeted FPGA hardware, and the results from these tests had validated the functionality of the proposed system. Furthermore, the data from the analysis shows that it can perform a very fast detection while consuming low hardware resources. This result may contribute to

improving the performance of systems which are considered as time-critical sytems and which require very high-speed execution and very low computing time.

REFERENCES
[1]   Willy A. "5 common causes of road accidents in Malaysia", [Internet]. Motorme - Malaysia Cars Bikes Trucks One Stop Automotive Website. 2015. Available from: http://www.motorme.my/5-common-causes-of-road-accidents-in-malaysia/
[2]   S. Gothane and M. V. Sarode, *"Analyzing Factors, Construction of Dataset, Estimating Importance of Factor, and Generation of Association Rules for Indian Road Accident"*, 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, 2016, pp. 15-18.
[3]   "Human Error Accounts For 90% Of Road Accidents", [Internet] International News. 2011. Available: http://channel.staging.alertdriving.com/home/fleet-alert-magazine/international/human-error-accounts-90-road-accidents
[4]   Kumar A, Jaiswal A, Jaiswal N, Sharma R. Vehicles Anti-collision System. *International Journal of Computer Applications*. 2014; 99(19): 7-9.
[5]   Triveni Shinde and Prof. B.V. Pawar. "Car Anti-Collision and Intercommunication System using Communication Protocol (A Prototype Model)". *International Journal of Engineering Sciences & Research Technology*, 2(8): 25-32, 2013.
[6]   Shival Dubey and Abdul Wahid Ansari. "Design and Development of Vehicle anti-collision System using Electromagnet and Ultrasonic Sensors". *International Journal on Theoretical and Applied Research in Mechanical Engineering (IJTARME),* Volume 2, Issue 1, 2013.
[7]   Zulkifli AS, Arduino Based Automobile Front Sensor, Universiti Teknikal Malaysia Melaka, 2015.
[8]   Sutikno T, Idris NR, Jidin AZ, Jidin A. A Model of FPGA-based Direct Torque Controller. *Indonesian Journal of Electrical Engineering and Computer Science*. 2013 Feb 1; 11(2): 747-53.
[9]   M.A. Kacou, F. Ghaffari, O. Romain and B. Condamin, *"Influence of high-power electric motor on an FPGA used in the drive system of electric car"*, IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 2016, pp. 4796-4801.
[10] Maxfield C. FPGAs. 1st ed. Amsterdam: Newnes; 2008.
[11] D. Chen and D. Singh, *"Parallelizing FPGA Technology Mapping Using Graphics Processing Units (GPUs)"*, *2010* International Conference on Field Programmable Logic and Applications, Milano, 2010, pp. 125-132.
[12] C. Zhao and Z. Zhang, *"Digital filter design and performance analysis of dynamic temperature signal denoise based on FPGA"*, 2016 10th International Conference on Sensing Technology (ICST), Nanjing, China, 2016, pp. 1-7.
[13] G. Werner, S. Farris, A. Kaminsky, M. Kurdziel, M. Lukowiak and S. Radziszowski, *"Implementing authenticated encryption algorithm MK-3 on FPGA"*, MILCOM 2016 - 2016 IEEE Military Communications Conference, Baltimore, MD, USA, 2016, pp. 1225-1230.
[14] Ultrasonic Ranging Module HC - SR04 [Internet]. Available from: http://www.micropik.com/PDF/HCSR04.pdf

BIOGRAPHIES OF AUTHORS

Aiman Zakwan Jidin obtained his MEng in Electronic and Microelectronic System Engineering from ESIEE Engineering Paris France in 2011. He has 2 years of working experience in designing digital IC and digital system in FPGA at Altera Corporation Malaysia (now part of Intel), before joining Universiti Teknikal Malaysia Melaka (UTeM) as lecturer and researcher, in Electronics and Computer Engineering. His research interests include FPGA Design and Digital System Design.

Lim Siau Li is currently a final year student of Bachelor of Computer Engineering Technology (Computer Systems) at Universiti Technical Malaysia Melaka (UTeM), Malaysia. She had recently completed her final year project on developing the algorithm for the vehicle anti-collision alert system by using FPGA and Altera Quartus II software. She is currently undergoing an engineering internship at Intel Microelectronics (M) Sdn. Bhd.

Ahmad Fauzan B. Kadmin has over 13 years of experience in computer & communication engineering field with technical expert in mobile computing and wireless communication system. He Graduated with MEng in Computer & communication Engineering from Universiti Kebangsaan Malaysia. Previously, he worked as an Engineer with Megasteel Sdn. Bhd., Samsung SDI(M) Sdn. Bhd. and Agensi Angkasa Negara. Currently attached with UTeM as lecturer.