Car Crash Detector System

Mohd Shahril bin Mat Saad
Faculty of Electrical Engineering, Universiti Teknologi MARA,
40450 Shah Alam, Selangor.

Abstract—This paper propose a new technology of the safety system that called car crash detector. The sensor will detect the impact of collision and send the signal to the system. The system will automatically turn off the engine if it met the condition which is in danger situation. This can avoid the car from burning or get exploded. The system also preventing the airbag from activated if the low impact collision occurs where the situation is not dangerous to driver then the airbag is no need to be activated. Programmable Interface Controller (PIC) is use as an interface between the input (sensor) and output (engine system). The system use distance detector or distance sensor type to detect the distance changes within interval of 3cm between radiator and front of the car's engine or can also shown between backs of the car's engine and sensor. The change of the distance that varies by speed is considered to make sure the correct programming code can be downloaded into the PIC. This paper also discuss on the programming using the assembly language and the flow of the system. The programming code more concern on the delay operation of the system and the Analog-to-Digital Converter (ADC). The delay used in this system is 1.08ms and this delay operates to delay operation of reading the signal from input (sensor). Hardware for this project designed base on considering the distance by 8cm, 11cm, and 14cm. The result of this project is discussed in three cases which are low, high, and very high impact collision.

Keywords—PIC, distance sensor, engine system, ADC.

I. INTRODUCTION

A great number of people die every year in car accidents. In most cases, these accidents happen because of the mistake of one driver overtaking the other vehicle, or driving the car in fast speed. The other factors like bad condition of the road, mountainous regions, slippery road, or snow falls also contribute a great deal in causing accidents [1]. So we cannot say the accident is only happen when the people are careless.

Nowadays the technologies become more important to our life and we cannot deny that. The technologies help us to get the more advance in something like the electric device and many more. In case of the technology in device, the high cost car is included with several of the accident avoidance system. This technology also can be applied to all of car and this will become more important thing today to avoid the accident to happen. The safety device is such as the vehicle's headlamps, reflectors, lights and signals, vehicle's mirror, vehicle's brake, steering, and suspension system [2] [3]. The vehicle also included with the driver assistance system which is guide the driver to control the vehicle more efficient such as traction control system, tire pressure monitoring system, deflection detection system, reverse backup sensor, electronic stability control, anti-lock braking system, cornering brake control and forward collision warning system.

While considering all the possibilities before car get collide, the factor after the accident also must be consider to safe the driver. For an example if some cars occur in accident, most of the time the engine of the car must be effect by impact of the crash and it might be dangerous. The engine of the car maybe can burned, but the worse thing is the engine of the car exploded and people in the car that might be panic stuck according the damage of the front car. So, this problem must be solved by some advance system that can control to off the car's engine when it involves the high damage in front of the car. The car crash detector come with the new technology that detect the distance between the radiator to engine of the car in certain condition to give an output action whether to off the engine or let it on and also whether to activate the airbag or not. The system used the distance as the parameter to consider the output action.

II. METHODOLOGY

A. *Measuring distance using distance sensor*

The first step is to measure the distance by using distance sensor. The distance is set to be 8cm, 11cm, and 14cm and this value means the distance from the back of the car's engine to the sensor. The value output that must be identified or measured is in voltage. Distance sensor give an output voltage and this voltage level can change by varying the distance from sensor to the object.

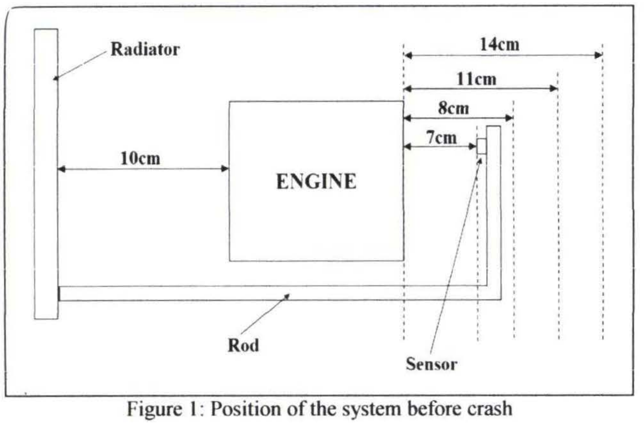

Figure 1: Position of the system before crash

Figure 1 shows the application of the system on car's engine. The distance sensor attach to the end of the rod and it place against on the back of the engine. The initial distance between the back of car's engine to the sensor is set to 7cm. The end of the front rod is touched to the radiator and the distance between the fronts of the car's engine to radiator is 10cm. As mention earlier, the three distances are set at the back of the engine. These distances are set according to the output voltage reading which are 2.6V, 2.0V, and 1.6V respectively.

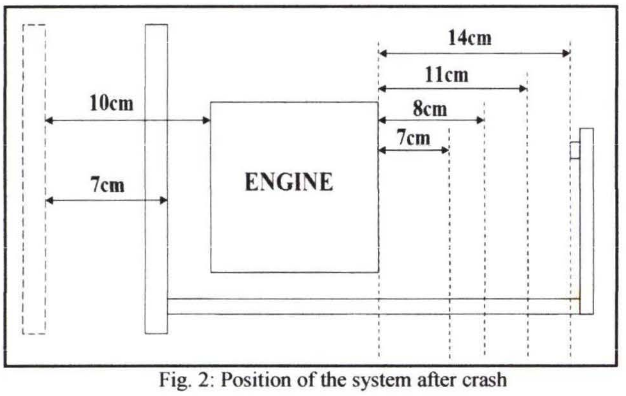

Fig. 2: Position of the system after crash

Figure 2 shows the position of the radiator, rod and all distance. The rod will move backward causes by the impact of an accident when the car was hit by another car or it hit to the wall.

The output voltage level from distance sensor can be explained by Figure 3. The distance for this system is selected above the 5cm, this is because the voltage is rapidly decreased below the distance of 5cm by short distance interval. The output voltage level decrease by increasing of the distance from object to the sensor. The operation of the distance sensor actually is using the optical parameter which is transmitting, reflected and receiving the infrared signal.

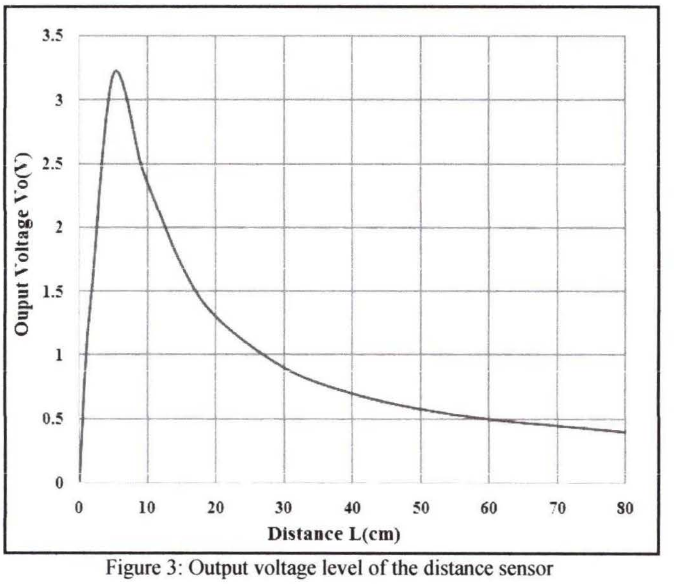

Figure 3: Output voltage level of the distance sensor

B. *Programming using Assembly language on PIC*

The Programmable Interface Controller (PIC) or microcontroller can be program by using several languages such as assembly code or C language. This project is using MPLAB software to program the microcontroller. The assembly language has lots of different from the C language because it is using simple command compare to the assembly language. Before write the programming code, the understanding about PIC is important to make sure all command can operate smoothly without any error. The understanding can be achieved by doing some study on the PIC datasheet. Instruction set can be categories into four groups which are:

1. Byte-oriented operations
2. Bit-oriented operations
3. Literal operations
4. Control operations

The microcontroller consists of the analog-to-digital converter (ADC) which converts the analog signal (voltage level) to be in digital signal. This project will use the output voltage level produce by the distance sensor as the input for microcontroller. This voltage level will convert by ADC build-in the microcontroller. The register that include in the operation of ADC is ADRESH, ADRESL, ADCON0, ADCON1, and ADCON2.

The value that set into register ADCON0 is 0x00H or in the clear state which is all bit in register is zero. Bit (7-6) is the unimplemented and it set to be zero. The bit (5-2) is the analog channel select bits and it choose to be all zero which means the channel 0 (AN0) is selected. Bit (1) is the A/D conversion status bit where it will be detect when running the program code whether '0' or '1'. The bit (0) will be set to enable the ADC and for the programming it initially at clear condition '0'. For the register ADCON1, the value that set into it is 0x0EH. Bit (7-6) is unimplemented and is choose in clear condition. Bit (6-5) is in clear condition '00' and this is because to set Vdd as (Vref+) and Vss as (Vref-). Bit (3-0) is an A/D port configuration control bit and these bits are choosing to be '1110' which means the AN0 is use as the analog input. For the register ADCON2, bit (7) is A/D result format select bit that clear to'0' which means the result of A/D in the left justified. Bit (6) is an unimplemented which is clear to be '0'. Bit (5-3) is A/D acquisition time select bits which are set to '101' that give the $12T_{AD}$. Bit (2-0) is A/D conversion clock select bits that set to be '010'. After all the programming code has been done, the PIC needs to be program using the software MPLAB. The process to program the PIC is called "burning" process. The assembly code must be converted into machine code before apply the burning process.

C. *Hardware of the system*

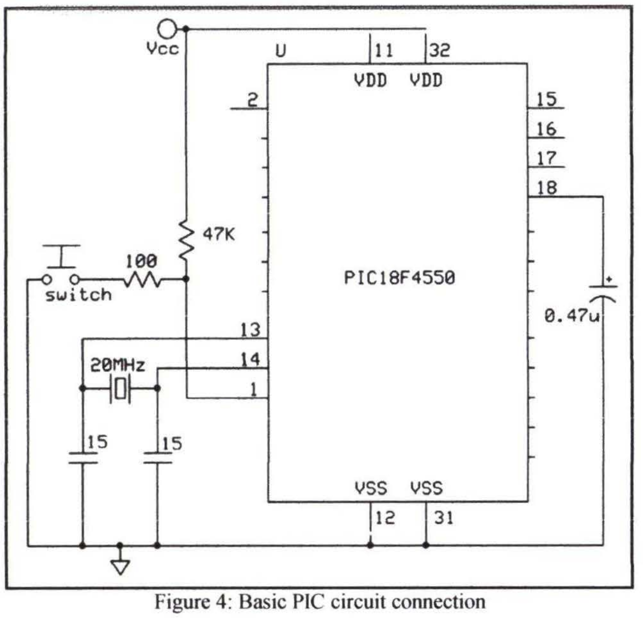

Figure 4: Basic PIC circuit connection

The microcontroller that used in this project is PIC18F4550 which having 40-pin (35-pin I/O). It has port A, B, C, D and E, the input port which connected to the sensor is set to PORTA and output port which connected to engine (DC motor), airbag (LED), and buzzer is set to PORTC. Figure 4 shows the basic connection of the component such as resistor, capacitor, resonator, and switch to the microcontroller pin. Vcc is connected to the pin 11 and pin 32 while Vss connected to pin 12 and pin 31. The pin 2 is RA0 (AN0) where the output signal (Vo) from distance sensor will be connected to the input of microcontroller. The pins (15-17) are the output from microcontroller that sent the output signal as digital signal to DC motor, LED, and buzzer. Pin 15 will be input signal to the DC motor, pin 16 is input to the LED and pin 17 is input to the buzzer. Figure 5 below shows the distance sensor with the configuration of the pin. Pin 1 is output signal from the sensor that will be connected to input of microcontroller, pin 2 is ground and pin 3 is connected to the Vcc. The Vcc is set to +5V and use to supply the input signal to the sensor.

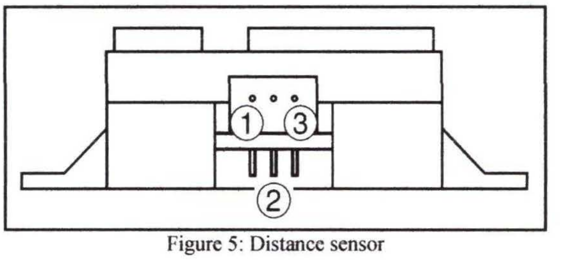

Figure 5: Distance sensor

C. *Real case crash calculation*

The real time calculation for collision of the car is important because from there, the displacement front of the car can be known by the certain time with speed of the car. To calculate this real time and displacement, one case is selected and the case take one car for a certain speed collides with the wall. The calculation for this case is simple by using the physic theory and equation.

$$v = x / t \qquad (1)$$

Where;

v = velocity
x = displacement
t = time

$$100\text{km/h} = 3\text{cm} / t$$

$$t = 1.08 \times 10^{-3}\ \text{s}$$

The calculation above give the answer $t = 1.08 \times 10^{-3}$ s which means if the car with speed of 100km/h collide with the wall, the time take to travel within 3cm displacement is 1.08×10^{-3} s.

The actual delay that has been set into programming code is not 1.08ms, but is 21.6µs. This time delay have been set because consider to the input reading which means the frequently of data or signal been read. For example, if the delay is too long the reading maybe not accurate and it will affect the output result. The calculation below shows how to get the time delay of 21.6µs.

$$\begin{aligned} Delay_{actual} &= Delay_{old} / 50 \\ &= 1.08ms / 50 \\ &= 21.6\mu s \end{aligned}$$

The old delay which is 1.08ms is divided with 50 and gets the actual delay 21.6μs. Value of 50 is the counter that set as comparison in the programming code. If the sequence repeat 50 times, it will get 1.08ms total time delay which indicated that the 3cm distance interval has been covered with the speed of 100km/h.

D. *Flow chart of the system and program code*

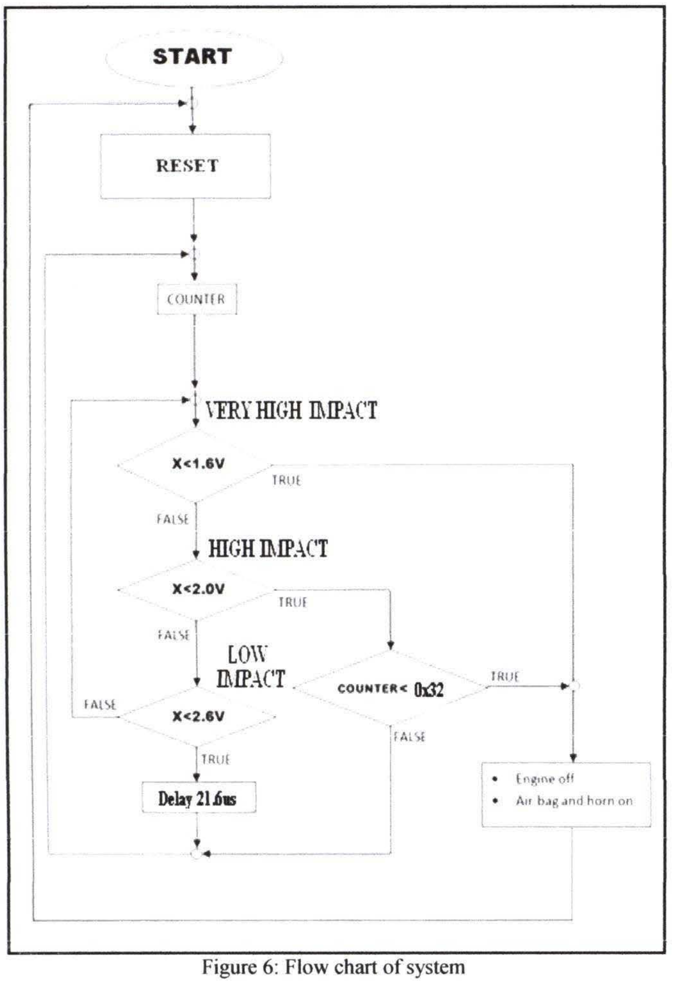

Figure 6: Flow chart of system

Figure 6 show the flow chart of the Car Crash Detector system. To explain the flow chart of the system, the three cases of high impact collision is selected. Firstly let say the low impact collision is occur. When the car's engine started, the flow of the process will go to the block counter. This counter will count the numbers of time it exceed the low impact collision state. Then, the flow will go to the condition that detects the Very High Impact Collision. At this state, the condition will compare the result of Analog-to-Digital conversion with the value of +1.6V. The result from A/D for low impact collision is greater than 2.6V, meaning that for this state the condition is false. The flow will go to next condition where it check for high impact collision and compare the A/D result with 2.0V. At this state the condition also is false and the flow will go to the last condition where is to check for low impact collision. The condition of this state true which means the result from A/D (x) is greater than 2.6V. The flow will go to delay block where the delay operation of 21.6μs executed and the flow is repeated again.

For high impact collision, the flow is same as low impact collision until it reach the second condition where to check the x and compare it with 2.0V. The high impact collision will give the A/D result lower than 2.0V and for this case, the second condition is true where the x lower than 2.0V. The flow then goes to condition to check and compare the counter. The counter is less than 32_{hex} which means equal to 50_{dec}. For this situation the condition is true, the flow will go to block operation that will deactivate the car's engine and activate the air-bag and horn.

The last case is for very high impact collision and for this case the A/D result is less than 1.6V. When the car is start, the flow still go to reset and counter block similar as low impact and high impact collision, but in the first condition where is to check and compare the A/D result with 1.6V, it is true and the flow will directly go to block operation that will deactivate the car's engine and activate the air-bag and horn.

Process will go to the block counter. This counter will count the numbers of time it exceed the low impact collision state. Then, the flow will go to the condition that detects the Very High Impact Collision. At

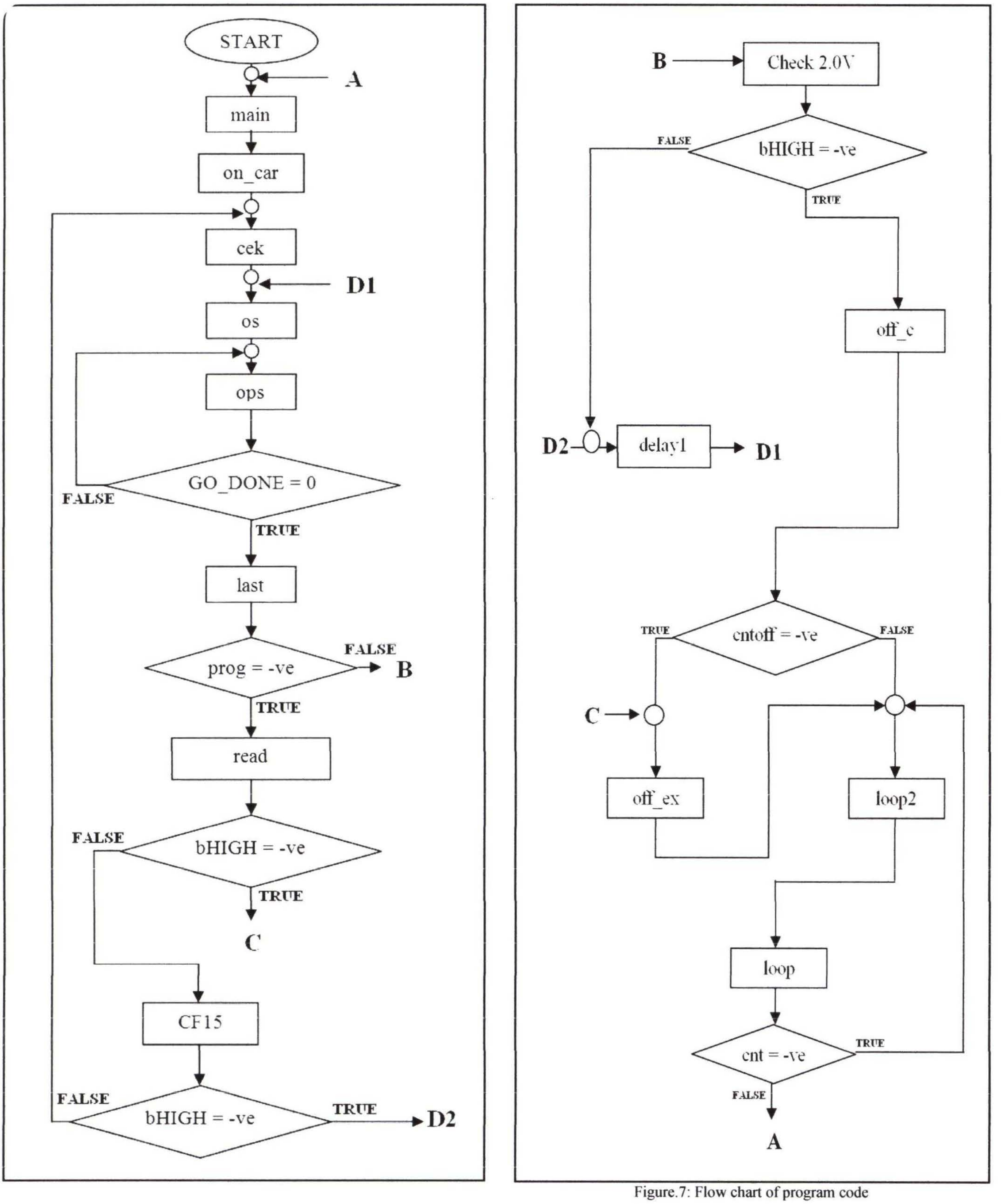

Figure.7: Flow chart of program code

Flow chart in Figure 7 shows the flow of programming process when switch of the car was activated. To explain the flow chart, let take an example which the collision give the very high impact to car's engine. Once the car was activated, the program in microcontroller run and the first process of it is initializing the port within the pin of microcontroller whether input or output. It also initialize the analog-to-digital setting which convert the analog input in form of voltage level to digital signal. The A/D process take a time to fully convert the data and bit GO_DONE is checked whether it '0 = A/D done' or '1 = A/D still in progress'. If the A/D was done, the program go to block "last" and this block function to compare "prog" with 0x01H. In this case the result become negative, the flow then go to block "read" and the program code in this block is compare the A/D result. The A/D result is negative which means the result of A/D is low from 1.6V (x > 14cm), the flows then go to block "off_ex". This block will deactivate the DC Motor, and activate the LED and Buzzer. This means the engine of car will switch off and the air-bag and horn is activated.

E. *Block Diagram of the System*

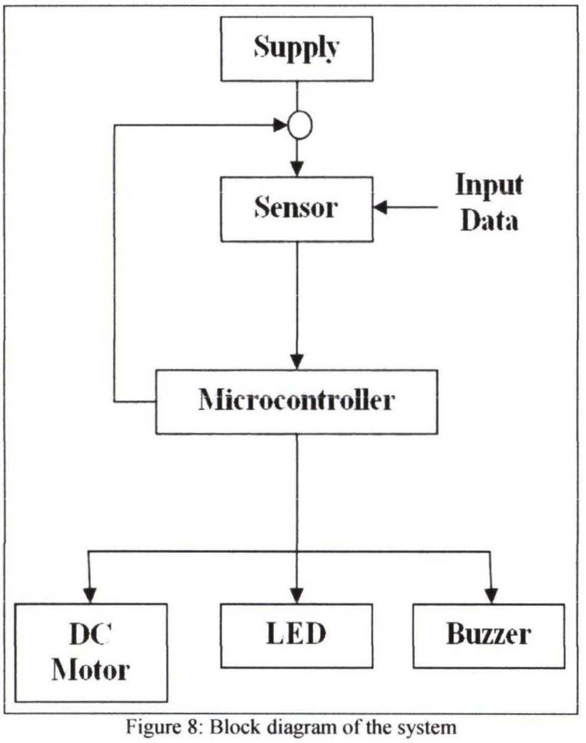

Figure 8: Block diagram of the system

The block diagram shown in Figure 8 explains the flow of the system which is from input data to the output that is DC Motor, LED, and buzzer. Battery 9V is use as a supply for the sensor and sensor detect the input data which in form of optical signal. The output of the sensor will be connected to the microcontroller that decide and generate the system. Microcontroller is always connected to the sensor which means that it will give signal to sensor to take the input data in certain time interval. After it get the data require, the microcontroller will process the result and make decision to send the signal to DC Motor, LED, and Buzzer.

III. Results and Discussion

The result for this project have been taken from three cases which are for the low impact of car collision, high impact of car collision, and very high impact of car collision. The component such as DC motor represents the engine, LED represents the air-bag, and buzzer represents the horn of the car.

A. *Low impact collision*

Figure 9 and Figure 10 shows the position of the system after crash in case of the low impact collision. When car collide with the wall, the front body of the car decline to the back and push the rod that touched at the radiator. The sensor that attach to the end of the rod detect the changes of distance from back of engine and translate it as the voltage level output to the microcontroller. The microcontroller compare the result with parameter that set to it, if the voltage level is exceed 2.6V which is the distance at 8cm it count the delay within distance interval of 3cm from 8cm to 11cm. The total time delay is set to be 0.00108second. If the condition does not meet which is the counter is not exceed the time delay, the output digital signal that sent to engine is '1' which means the engine does not deactivate because the collision give the low impact and cannot harm the engine. In this case also, the air-bag and buzzer is not activates due to low impact collision.

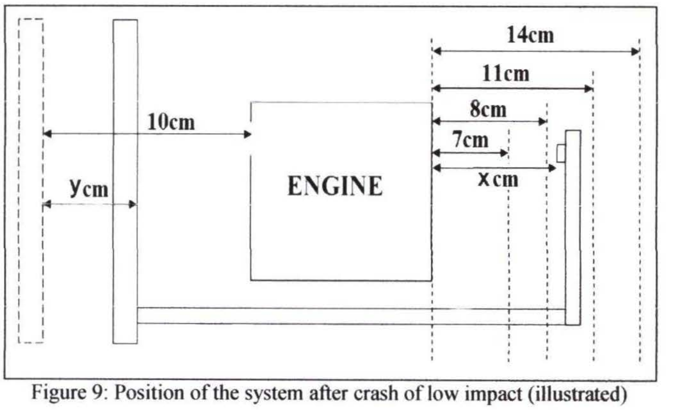

Figure 9: Position of the system after crash of low impact (illustrated)

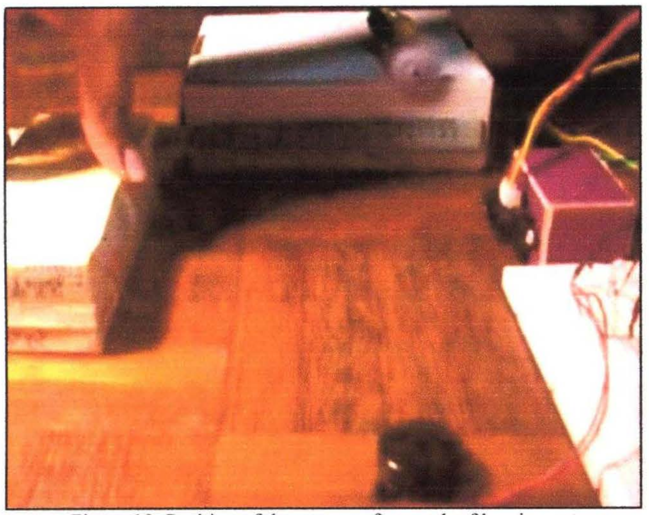

Figure 10: Position of the system after crash of low impact

B. *High impact collision*

Figure 11 and Figure 12 shows the position of the system after crash in case of the low impact collision. When car collide with the wall, the front body of the car decline to the back and push the rod that touched at the radiator. The sensor that attach to the end of the rod detect the changes of distance from back of engine and translate it as the voltage level output to the microcontroller. The microcontroller compare the result with parameter that set to it, if the voltage level is exceed 2.6V which is the distance at 8cm it count the delay within distance interval of 3cm from 8cm to 11cm. The total time delay is set to be 0.00108second. If the condition does meet which is the counter is exceed the time delay, the output digital signal that sent to engine is '0' which means the engine is deactivate because the collision give the high impact and can harm the engine. In this case also, the air-bag and buzzer is activates due to high impact collision.

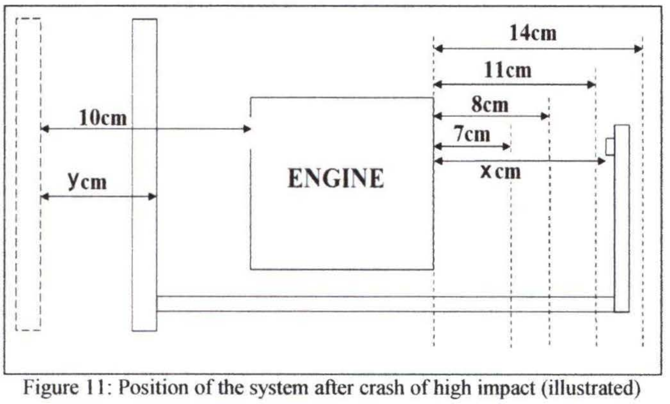

Figure 11: Position of the system after crash of high impact (illustrated)

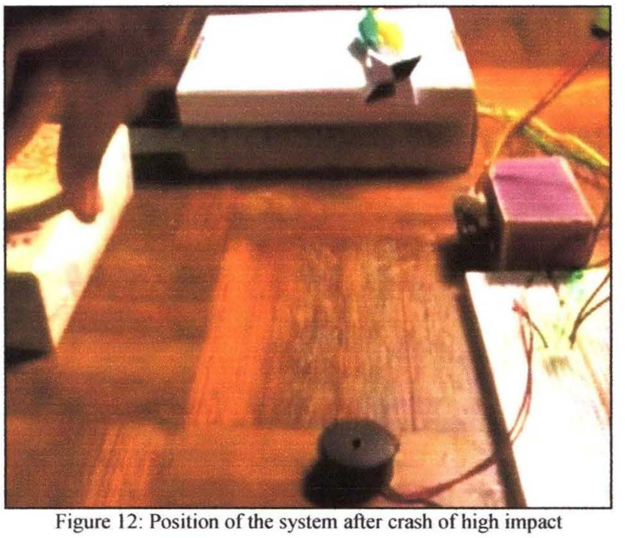

Figure 12: Position of the system after crash of high impact

C. Very *high impact collision*

Figure 13 and Figure 14 shows the position of the system after crash in case of the low impact collision. The third case is different from the two cases above because this time the collision is very high and it reach the very high speed and damage to the engine. When car collide with the wall, the front body of the car decline to the back and push the rod that touched at the radiator. The sensor that attach to the end of the rod detect the changes of distance from back of engine and translate it as the voltage level output to the microcontroller. The microcontroller compares the result with parameter that set to it which is the voltage level of 1.6V which is the distance at 14cm. If the condition does meet, the output digital signal that sent to engine is '0' which means the engine is deactivate because the collision give the high impact and can harm the engine. In this case also, the air-bag and buzzer is activates due to very high impact collision.

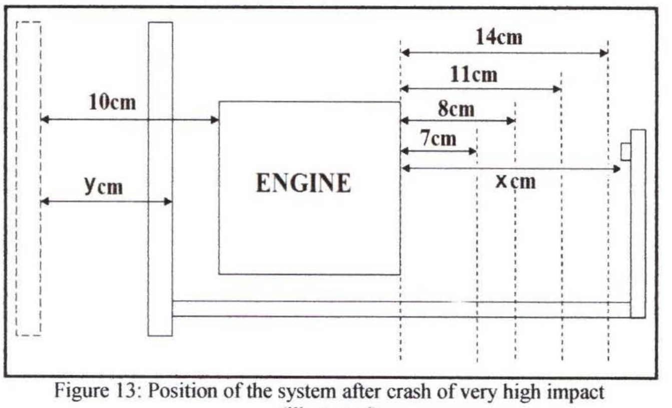

Figure 13: Position of the system after crash of very high impact (illustrated)

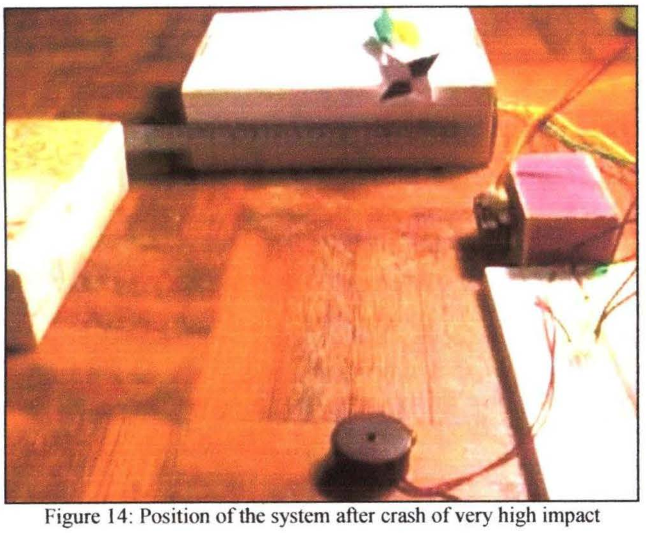

Figure 14: Position of the system after crash of very high impact

IV. CONCLUSION

The functional of the system have been proved by showing the cases of low, high, and very high impact collision. Base on the result from this project, the technology of car safety can be improved by using this system. The system operates accordingly to design of the system which will deactivate the engine if the car collision exceeds the setting speed (100km/hour). It provides the safety design for collision that gives high damage to the car which can blow up the engine.

From the result also, the system is proven can solve the problem exploded of the car's engine when collision happen. This will decrease the death case where the accident occurs. With this system, problem such as the air-bag activated when low impact collision occurs also solved.

V. FUTURE DEVELOPMENT

The results can be improved in future in many ways. It is recommended that the sensor of using the high collecting input which means that system will be improved in term of accuracy. The sensor that can be use is such as ultrasonic sensor. This system also can be modified to use as the automatic car driver which means that the driver do not have to control or handling their car.

ACKNOWLEDGMENT

The author would like to thanks Puan Zurita as supervisor for her guidance and support in conducting the design process of the Car Crash Detector. The author also would like to thanks Prof Uzer for his kind and helpfulness in teaching of MPLAB software and lots of opinions and idea that been thrown in accomplishing the Car Crash Detector Programming design.

REFERENCES

[1] Harry Lum & Jerry A. Reagan (Winter 1995). "Interactive Highway Safety Design Model: Accident Predictive Module

[2] The Seat Belt, Swedish Research and Development for Global Automotive Safety

[3] The Automotive Chassis Engineering Principles - J. Reimpell H. Stoll J. W. Betzler

[4] PIC18F2455/2550/4455/4550 Data Sheet (Microchip)