

## FPGA Realization of a Fuzzy Based Wheelchair Controller

<sup>1</sup>Md. Shabiul Islam, <sup>1</sup>M.S. Bhuyan and <sup>2</sup>Sawal H. Md. Ali

<sup>1</sup>Institute of Microengineering and Nanoelectronics (IMEN),  
Universiti Kebangsaan Malaysia (UKM), 43600 UKM Bangi, Selangor, Malaysia

<sup>2</sup>Department of Electrical, Electronic and Systems Engineering,  
Faculty of Engineering and Built Environment, 43600 UKM Bangi, Selangor, Malaysia

---

**Abstract:** This study describes a FPGA realization of a Fuzzy Logic Controller (FLC) algorithm for designing a Wheelchair Controller (WC). The controller enables the movement of wheelchair and makes brake in an unstructured environment by the WC sensor to avoid any encountered obstacles. The WC is found to be able to react to the environment appropriately during its navigation to avoid crashing with obstacles by turning to the proper angle while moving. To design the controller unit, a speed sensor and a distance sensor, etc. are placed in front of the wheelchair for its functionality. The numerous data is used to evaluate the algorithm which control an output signal for the brake-power using by the input signals of speed sensor and distance sensor. The FLC has proven a commendable solution in dealing with certain control problems when the situation is ambiguous. One of the main difficulties faced by conventional control systems is the inability to operate in a condition with incomplete and imprecise information. As the complexity of a situation increases, a traditional mathematical model will be difficult to implement. Fuzzy logic is a tool for modeling uncertain systems by facilitating common sense reasoning in decision-making in the absence of complete and precise information. In this study, the WC is designed based on the theories of fuzzy logic (such as fuzzifier, fuzzy rule base, inference mechanism and defuzzifier) and then simulated in MATLAB platform. The designed codes of WC also have written in VHDL language for implementing the hardware blocks of the separate modules of the WC. The verified VHDL code of the WC has been synthesized using Quartus II tool in Altera environment. Finally, the hardware designed codes have downloaded into FPGA board (APEX 20K200EF484) for the circuit's functionality verification. From the timing analyzer report during implementation into FPGA board, it is observed that the longest delay from the source pin "sel3" to destination pin "LED7SEG1" is 24.019 nsec. Hence, the maximum clock speed (fmax) of the wheelchair is 41.63 MHz. During the experiment, researchers have set the hardware working frequency in 40 MHz to be confirmed the reliability in working condition.

**Key words:** Wheelchair controller, fuzzy logic rules, VHDL, dynthesis, FPGA

---

### INTRODUCTION

In the last decades, mobile robots have been a subject of major interest cause of the wide range of possible applications. The navigation system is decomposed into the two sub-tasks: Path Tracking (PT) and Obstacle Avoidance (OA). The control system moves the robot along the path while handling disturbance, i.e., PT behavior and sudden obstacles, i.e., OA behavior (Luis *et al.*, 2002). In recent years, several autonomous wheelchair systems were developed. It focused on improving the autonomy, intelligence, safety and navigation of the wheelchair. The french acronym for autonomous vehicle for people with motor disabilities (Kuo *et al.*, 2003) has developed the autonomous wheelchair to help the disabled people who are

unpractical to drive a conventional powered wheelchair. The wheelchair control architecture is categorized as the physical, local and global levels. The Bremen Autonomous Wheelchair (BAW) (Kuo *et al.*, 2003) has developed a "Rolland" wheelchair and it is implemented by following the skills of the driving and route assistants. The "Rolland" is accomplished of smooth speed control, turning around in an open space, turning around in a corner, obstacle avoidance and doorway passage. It provides the mobility assistance for the handicapped people. Luis *et al.* (2002) from Portugal have developed fuzzy control based architecture for PT and OA. They combined the PT and OA in the wheelchair mobile robot by using a fuzzy tracking controller. A mobile robot developed for corridor navigation (Ono *et al.*, 2004) that uses a robot equipped with a minimal set of sensors such

as a camera and infrared sensors. Their multi-agent based control system is built to tackle various problems encountered during corridor navigation. A novel design and development on bio-impedance based wheelchair control for disable people and the elderly presented (Yunfei *et al.*, 2010). Researchers have use three electrodes to measure two channels of bio-impedance from the trapezius muscle. Bio-impedance changes when there is a movement in the segment of trapezius muscle. They can classify six types of motions resulting in six operation capabilities for wheelchair control based on six types of shoulder movements, i.e., left shoulder up, right shoulder up and both shoulder up for short time and long time. The system is composed of the modified Howland current bridge circuit which supplies the 0.5 mA ac current to the measurement system at the frequency of 50 kHz. NI PCI-6250 DAQ board was adopted to collect the data and LabView 8.2 was used to implement the signal processing and control system. Algorithms applied in the system are an automatic threshold value adjustment which adapt its value to the measured signal. A novel driving control scheme of electric power-assisted wheelchairs for higher energy efficiency driving based on fuzzy algorithm described by Tanohata *et al.* (2010). Electric power-assisted wheelchair which assists the driving force by electric motors is expected to be widely used as a mobility support system for elderly and disabled people. However, if the batteries have run out, the wheelchair imposes a strain on human operator. Therefore, this study proposes a novel fuzzy algorithm based assisted torque generation method according to the human input torque in order to realize higher energy efficiency driving. Fuzzy rules are designed from the consumption energy and driving distance in order to improve the energy efficiency. Some driving experiments confirm the effectiveness of the proposed control system (Islam *et al.*, 2011).

In the project, firstly researchers have investigated an autonomous wheelchair algorithm based on fuzzy logic and lastly realized the circuit functionality using FPGA board in the experiment. The developed wheelchair algorithm in MATLAB package has designed as hardware blocks of the WC using VHDL language under Altera environment. Then, the blocks of WC architecture have been implemented using FPGA board.

### WHEELCHAIR ARCHITECTURE USING FUZZY LOGIC ALGORITHM

The WC is simple but FLC algorithm makes it an intelligent. The FLC based wheelchair will be navigated through an unstructured environment without human intervention. In the process of moving forward, it is necessary for the wheelchair to have the information on its current position and speed where the intake of this imperative information through the two sensors is needed on its body. Figure 1 shows the physical view of the wheelchair. The structure of the wheelchair is made by two wheels and controlled by its remote buttons. A significant of the intelligent control wheelchair would be controlled by FLC algorithm, instead of human intelligence. The sensors are positioned in such a way that the wheelchair is able to detect its distance and speed from obstacles. The brake power has a maximum power of 60 pu (power unit). If the brake is not applied on the wheelchair, it will moves in a constant velocity and the brake power is controlled by the FLC. In project (Islam *et al.*, 2011), the intelligent wheelchair has been performed by using the following rules of the FLC algorithm. The block diagram of the fuzzy logic controller architecture is shown in Fig. 2 (Luis *et al.*, 2002).

Fuzzy controller takes the inputs (referred to as the crisp input) from the sensors and spreads it into degrees

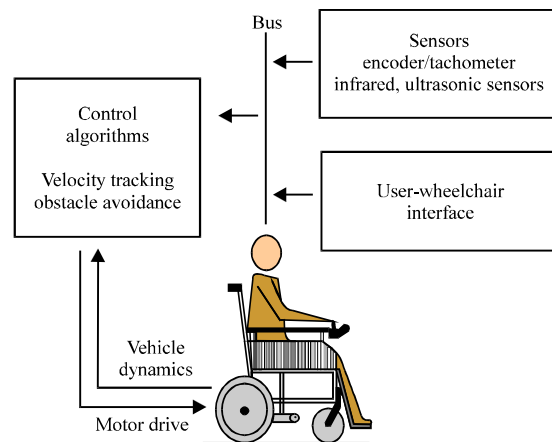


Fig. 1: Physical view of the wheelchair

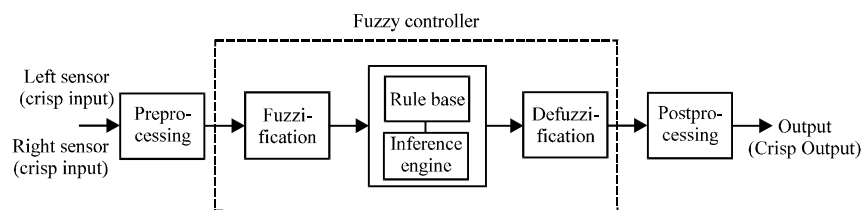


Fig. 2: Fuzzy Logic Controller (FLC) architecture

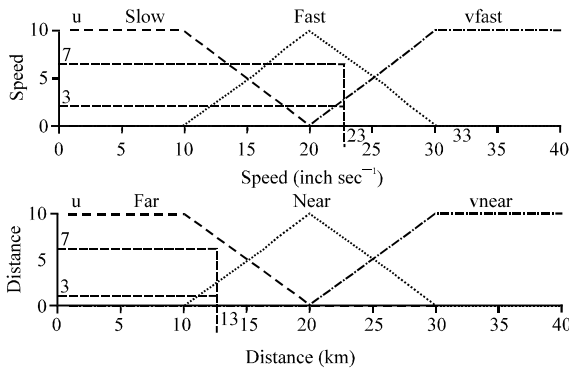


Fig. 3: a) Fuzzification of input parameters (speed sensor is 23); b) Fuzzification of input parameters (distance sensor is 13)

Table 1: Rule strength

Rules	$\mu_{slow}$ (speed)	$\mu_{fast}$ (speed)	$\mu_{vfast}$ (speed)
$\mu_{far}$ (distance)	$\alpha_1$	$\alpha_4$	$\alpha_7$
$\mu_{near}$ (distance)	$\alpha_2$	$\alpha_5$	$\alpha_8$
$\mu_{vnear}$ (distance)	$\alpha_3$	$\alpha_6$	$\alpha_9$

of membership between zero and one (the fuzzy input) as a function of the difference between the crisp input and the position of the member called the membership function.

Avoid the fuzzification in duddy controller loop-up table where the crisp input range is mapped to stored values for the fuzzy set's output. Fuzzy set refers a set or collection of membership functions from the same crisp input.

The fuzzy inputs are fed into the inference engine which contains a rule-base describing the controller's course of action, generating another set of fuzzy values called fuzzy output which are defuzzified to produce the controller's real, crisp output.

**Fuzzification:** Fuzzification is the process of transforming crisp values into degree of membership for linguistic terms of fuzzy sets. The measurement values of both the speed sensor and distance sensor are the input parameters which have to be translated to proper terms of the corresponding linguistic variables. As an example when the input of the speed sensor is 23, the input of the distance sensor is 13 as shown in Fig. 3a and b.

Table 1 shows the rule strength such as  $\alpha_1$  to  $\alpha_9$ , corresponds to rule 1 to rule 9, respectively. Here, researchers shall refer  $\alpha$  as strength for simplicity. It is sometimes referred to as strength of rule. Example:  $\alpha = \mu$  (Speed)  $\wedge$   $\mu$  (Distance) =  $\min(\mu$  (Speed),  $\mu$  (Distance)). As an example for speed = 23 and distance = 13, the rule strength is given in Table 2. From Table 2, it can be noticed that only four cells contain non-zero value which

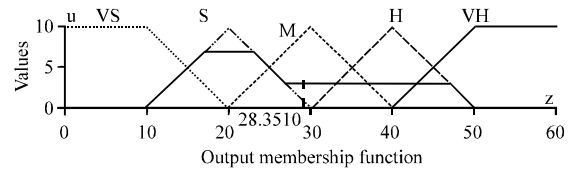


Fig. 4: Center of Gravity (COG)

Table 2: Rule strength for speed = 23 and distance = 13

Rules	$\mu_{slow}$ (speed) = 0	$\mu_{fast}$ (speed) = 7	$\mu_{vfast}$ (speed) = 3
$\mu_{far}$ (distance) = 7	0	7	3
$\mu_{near}$ (distance) = 3	0	3	3
$\mu_{vnear}$ (distance) = 0	0	0	0

are  $\alpha_4$ ,  $\alpha_5$ ,  $\alpha_7$  and  $\alpha_8$ . This indicates that the only active rules are rule 4, 5, 7 and 8. After obtaining the rule strength table, the Control Outputs (CO) has to be obtained:

- CO of rule 4:  $\alpha_4 \wedge \mu P$  (S) =  $\min(\alpha_4, \mu P$  (S))
- CO of rule 5:  $\alpha_5 \wedge \mu P$  (M) =  $\min(\alpha_5, \mu P$  (M))
- CO of rule 7:  $\alpha_7 \wedge \mu P$  (M) =  $\min(\alpha_7, \mu P$  (M))
- CO of rule 8:  $\alpha_8 \wedge \mu P$  (H) =  $\min(\alpha_8, \mu P$  (H))

The control outputs of the two rules have to be combined or aggregated in order to produce an output with membership function  $\alpha_{agg}(z)$  called the aggregated output. It is also expressed in below:  $\alpha_{agg}(z) = \max(\text{CO1, CO2, CO3, CO4, CO5, CO6, CO7, CO8, CO9}) = \max(\text{CO4, CO5, CO7, CO8}) = \max\{(\alpha_4 \wedge \mu P$  (S)),  $(\alpha_5 \wedge \mu P$  (M)),  $(\alpha_7 \wedge \mu P$  (M)),  $(\alpha_8 \wedge \mu P$  (H))\} =  $\max\{\min(\alpha_4, \mu P$  (S)),  $\min(\alpha_5, \mu P$  (M)),  $\min(\alpha_7, \mu P$  (M)),  $\min(\alpha_8, \mu P$  (H))\}.

**Defuzzification:** Upon receiving the inputs, the values are presented in fuzzy terms by the fuzzification process. In order to produce a control action which is a crisp value z (a non-fuzzy term), defuzzification has to be applied. The defuzzification used to calculate the Center of Gravity (COG) Method: the COG can be represented graphically in Fig. 4.

$$\text{COG} = \frac{\int z * \mu \text{Agg}(z) / \int \mu \text{Agg}(z)}{\int \mu \text{Agg}(z)} = \frac{\sum z k * \mu \text{Agg}(z k) / \sum \mu \text{Agg}(z k)}{\sum \mu \text{Agg}(z k)} = 28.3510$$
 The result of the defuzzification is the output of the FLC which is the brake power of the wheelchair. The black line is the defuzzification area. There are five fuzzy sets in the output membership function which are VS (Very Soft) in blue line, S (Soft) in red line, M (Medium) in green line, H (Hard) in yellow line and VH (Very Hard) of the last one. For this case when the input of the speed sensor is 23 and the input of the distance sensor is 13, the corresponding output is approximately 28 which fall under the S and M fuzzy sets.

The fuzzy logic based WC has been simulated using MATLAB package for a few cases with different values.

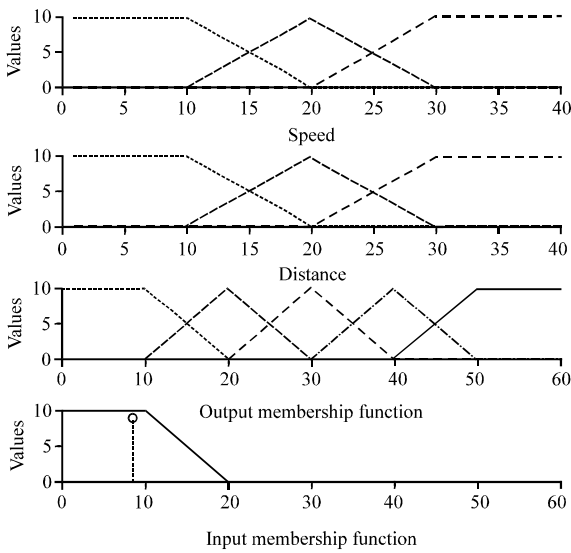


Fig. 5: Graphical simulation result in MATLAB (Case No. 1)

Here, the simulated result in MATLAB for one case only (as an example) is shown in Fig. 5, as partially graphical representation. The graph consists of the input membership functions (one for the speed sensor and another for the distance sensor), graph of the output membership function and the graph of the aggregated output (final output).

**MODELING OF THE WHEELCHAIR IN VHDL**

The structure of the WC controller based on FLC algorithm has modeled using VHDL language which consists of three modules namely `wc_evabot.m`, `wc_strength.m` and `wc_cog.m` as shown in Fig. 6. Firstly, the module of `wc_evabot.m` consists of two input ports and one output port. The inputs are the actual sensor’s inputs (Porrero and Ballabio, 1998) which are the speed of the wheelchair and the distance between obstacles and wheelchair. The output is the brake power of the wheelchair. The design in VHDL does not show the data of degree of membership and rule strengths as outputs because these data are not necessary to be probed as the actual pin outs of the hardware. These data can be declared as internal signals to eliminate the waste of usage of pins on the FPGA. As `evabot.vhd` is the top-level entity, it calls the other lower level modules by declaring them as components. It declares three components which are two `str` modules and one `cog` module. Secondly, the module of the `str.vhd` implements the `wc_strength.m` with most of its function but not generating the rule strengths. Instead, it only computes the degree of membership. As VHDL has the reuse

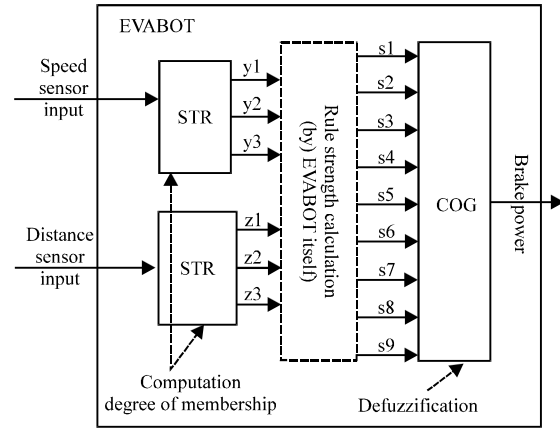


Fig. 6: Structure of the wheelchair controller based on FLC in VHDL

Table 3: Comparison of results between VHDL and MATLAB

Cases	A (speed) (inch/sec)	B (distance) (inch)	Output in MATLAB (pu)	Output in VHDL (pu)
1	5	8	8.52	8
2	13	23	28.35	28
3	10	35	31.00	31
4	18	11	19.50	19
5	28	8	28.59	28
6	33	13	34.35	34
7	33	23	45.10	45
8	33	33	53.48	53
9	21	29	41.17	41

capability, so one `str` module is created only because the rules compute the degree of membership for both of the speed sensor and the distance sensor as similar. This reuse capability is useful for design systems that have repeated module. Third, the module of the `cog.vhd` implements all the functionalities of the `wc_cog.m`. It implements the defuzzification process using the COG Method.

The “two `str`” modules separately receive speed sensor input and distance sensor input and then produce the degree of membership as `y1`, `y2`, `y3` and `z1`, `z2` and `z3`. The top level `evabot` module produces the rule strengths as `s1`, `s2`, `s3`, `s4`, `s5`, `s6`, `s7`, `s8` and `s9`. The rule strengths are sent to `COG` for defuzzification and the final output is produced as the brake power of the wheelchair. However, the simulation in VHDL only produces one output which is the brake power of the wheelchair, ‘`o1`’. Because it is unnecessary to have the degree of membership and the rule strengths as actual pin outs of the chip. Table 3 shows the brake power that simulated by VHDL. “`i1`” is the input of speed sensor, “`i2`” is the input of distance sensor and “`o1`” is the brake power produced by the fuzzy logic WC controller. The simulated waveform (example: Case No. 1) is shown in Fig. 7.

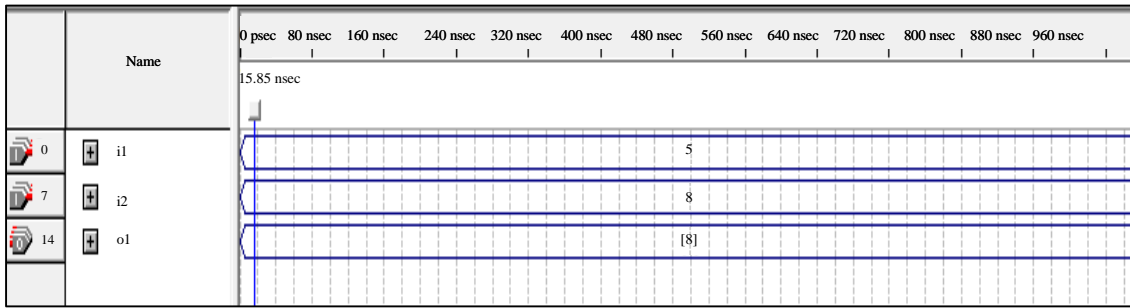


Fig. 7: Simulation result in VHDL (Case No. 1)

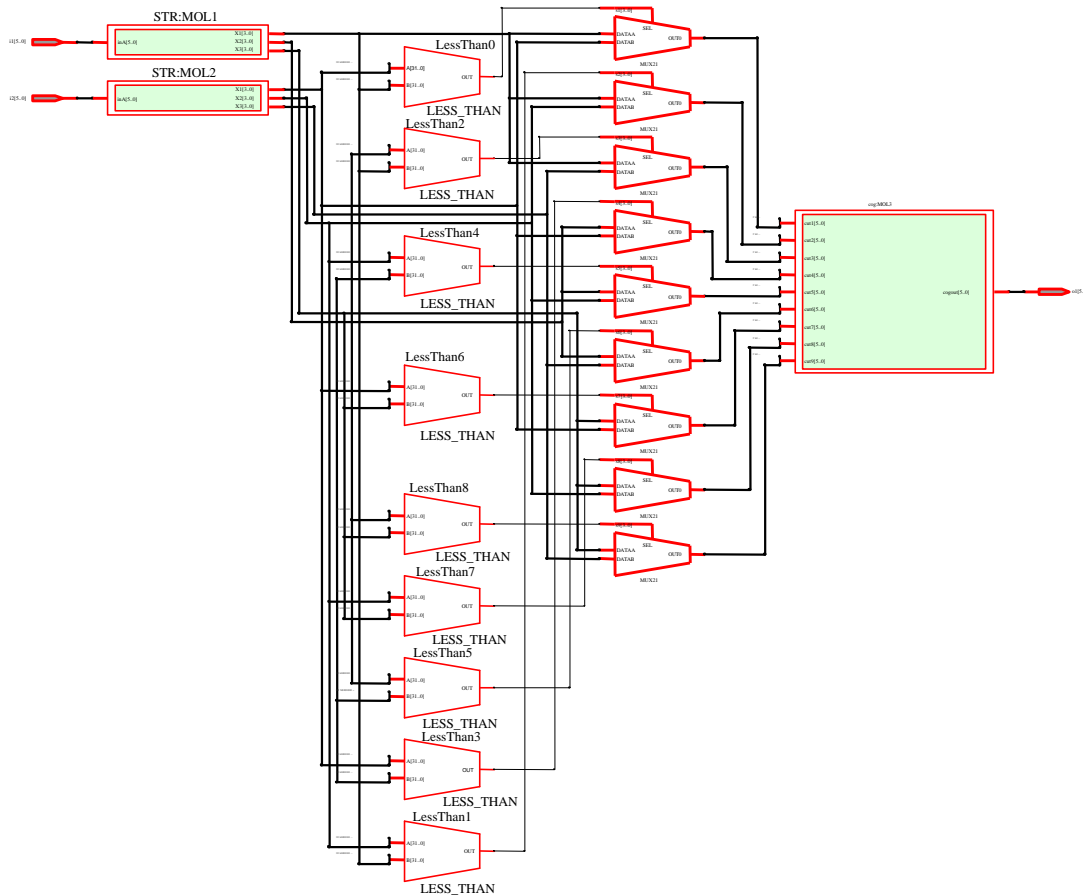


Fig. 8: RTL view of the “EVABOT” block (WC controller)

Here, “o1” is the output of the FLC while “i1” and “i2” are the inputs from the speed sensor and distance sensor, respectively. The brake power that produced is 8 pu when the speed is 5 inches/sec and distance is 8 inches. As an example, speed (i1): 5 and distance (i2): 8. The results of the final output are the brake power of the wheelchair which simulated in VHDL under Quartus-II Software, Altera. The comparison of simulation results in MATLAB and VHDL are given in Table 3. The different values are due to cause of VHDL simulator (in integer form) and MATLAB simulator (in floating-point form), so

the difference in between two simulators are minor values of decimal point. As a result, there is no problem for synthesizing the VHDL design codes of the WC architecture for realization using FPGA board as well as VLSI implementation.

**SYNTHESIS OF THE WHEELCHAIR CONTROLLER**

Synthesis is the process of transforming one representation in the design abstraction hierarchy to another representation. Synthesis process has performed

using synthesis tool for synthesizing the compiled VHDL design codes into gate level schematics. In order to get the Register Transfer Level (RTL) view of the WC architecture, the VHDL codes are synthesized using Quartus-II. The technology mapping has chosen in this project from Altera (FLEX 20K200EF484). The technology view of the various modules for WC architecture has been carried out. The output is a set of logic expressions that describe the logic functions needed to realize the circuit. The RTL description is converted into a gate netlist which makes downloading the codes into a target device. The design codes of the controller will be downloaded into FPGA board. This board consists of 8320 Logic Elements (LE), 106496 memory blocks and 2 Phase Lock Loops (PLL). As an example, the top level of RTL views of the WC is shown in Fig. 8. The flattened technology view of Evabot block consisting of different functional logic components of primitive cells are in this figure which describe the functionality at the behavioral level of the WC algorithm.

**FPGA IMPLEMENTATION**

The generated synthesized netlist of the WC controller has been downloaded into FPGA (APEX 20K200EF484) board from Altera environment for verification the correctness of the algorithm functionality. Note that the FPGA board contains a built in 8 bit DIP switch, a dual-digit seven-segment display and three expansion slots, each with 42 I/O pins and 7 global pins as shown in Fig. 9. To make compatible the fuzzy based WC controller inputs with I/O pins in FPGA board, the 8 bit DIP switches as manual inputs and 8 bit of two 7 segment LEDs as outputs are chosen for verifying the correct functionality of the WC controller for VLSI implementation. Next, the timing analyzer shows the delay of data path and gives report the performance of the design in terms of longest delay time. The inversion of the delay time is the  $f_{max}$ . From the analyzer report, it is observed that the longest delay from the source pin “sel3” to destination pin “LED7SEG1” is 24.019 nsec. Hence, the  $f_{max}$  of the wheelchair is 41.63 MHz. During the experiment, researchers have set the hardware working frequency in 40 MHz, so the circuit has performed in working condition. In this experiment (as example, case no 6 from Table 4), the input of the speed sensor (33 inches/sec) and the input of the distance sensor (13 inches) are verified by the evabot module of the WC controller for FPGA implementation through the str and cog modules as shown in Fig. 9. Figure 9 shows the FPGA output of 34 which is the brake power of the WC controller needed to apply on the wheelchair when the speed is 33 inches/sec and the distance between the

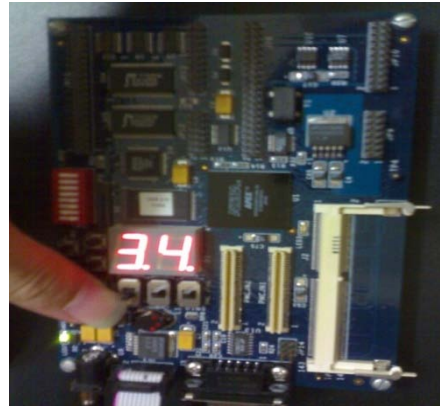


Fig. 9: FPGA implementation of the speed sensor (33 inches/sec) and the distance sensor (13 inches) of the WC controller

Table 4: Results of different cases into FPGA board

Cases	Speed (inch/sec) (input)	Distance (inch) (input)	Brake power (pu) (output)
1	5	8	8
2	13	23	28
3	10	35	31
4	18	11	19
5	28	8	28
6	33	13	34
7	33	23	45
8	33	33	53
9	21	29	41

wheelchair and obstacle is 13 inches. This result is computed by the FLC algorithm through the EVABOT module. Furthermore, Table 4 shows the statistical results on FPGA implementation when applying different inputs of the speed and distance are needed for different types implementation into FPGA board.

**CONCLUSION**

The FPGA realization of a FLC algorithm for the WC controller has been successfully developed in both of MATLAB and VHDL languages from Altera environment. Fuzzy logic is proved to be an excellent choice for control problems in ambiguous or vague environments. The developed algorithm of the WC controller in MATLAB package has been successfully translated into VHDL codes for hardware implementation. In this study, a comparison results in MATLAB and VHDL Software’s of the WC algorithm have presented for VLSI implementation. The VHDL codes then are synthesized using synthesis tool, Quartus-II, Altera. Finally, the WC hardware blocks for VLSI design have been carried out. The designed hardware blocks (i.e., gate-levels) of the WC controller have been downloaded into the FPGA

(FLEX 20K200EF484) board for its circuit functionality verification. The results show that the WC reacts differently according to the environment. It breaks itself with an appropriate power to avoid obstacles within its vicinity while navigating the completely unobstructed environment. A space for improvement in the design is refining the smoothness of the wheelchair. Besides, the distance of its vicinity has to be improved to a larger area and make the speed faster, so that the user can reach the destination with a shorter time. On the other words, the user can reach the destination faster without hitting any obstacles.

### REFERENCES

- Islam, S., M.R. Sarker, H.M. Sawal, M. Othman and Y.M. Burhanuddin, 2011. Design, simulation and synthesis of wheelchair controller using fuzzy logic. Proceedings of the IEEE Regional Symposium on Micro and Nanoelectronics, September 28-30, 2011, Kota Kinabalu, Sabah, Malaysia, pp: 117-122.
- Kuo, C.H., H.L. Huang and M.Y. Lee, 2003. Development of agent-based autonomous robotic wheelchair control systems. *J. Biomed. Eng. Appli. Basis Communi.*, 15: 223-234.
- Luis, C.B., P. Gabriel and N. Urbano, 2002. A behavior based fuzzy control architecture for path tracking and obstacle avoidance. Proceedings of the 5th Portuguese Conference on Automatic Control, September 5-7, 2002, Aveiro, Portugal, pp: 341-346.
- Ono, Y., H. Uchiyama and W. Potter, 2004. A mobile robot for corridor navigation: A multi-agent approach. Proceedings of the 42nd Annual Southeast Regional Conference, ACM, April 2-3, 2004, Huntsville, Alabama, USA., pp: 379-384.
- Porrero, I.P. and E. Ballabio, 1998. Sensors to Improve the Safety for Wheelchair Users. In: *Improving the Quality of Life for the European Citizen: Technology for Inclusive Design and Equality*, Schilling, K., H. Roth, R. Lieb and H. Stutzle (Eds.). IOS Press, Netherlands.
- Tanohata, N., H. Murakami and H. Seki, 2010. Battery friendly driving control of electric power-assisted wheelchair based on fuzzy algorithm. Proceedings of the SICE Annual Conference, August 18-21, 2010, IEEE Press, pp: 1595-1598.
- Yunfei, H., P. Phukpattaranont, B. Wongkittisuksa and S. Tanthanuch, 2010. Wheelchair control based on bioimpedance. *Int. J. Applied Biomed. Eng.*, 3: 13-17.