International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9, Issue-7, May 2020

Development of Gaiting Analysis using Footwear

M. C. Chinnaiah, A. Murali Krishna, C. V Dheeraj, D. Abhishek

Abstract: ***An intelligent hybrid project is developed for assisting the partially impaired and physically challenged people. The Fall Detection is proposed to ensure the safety of these people. Force Sensing Resistor (FSR) sensors are attached to the footwear. For each in a pair of shoes four FSR sensors are attached at the precise locations in the shoe. In this method, the user's pressure from the specified points is calculated in real time using an integrated force sensory system, which comprises FSR sensors in the shoe. When the pressure from these specified points cross the boundary limit of the specified safety value, i.e., the maximum pressure, it is assessed that the user is going to fall down. The data retrieved from these FSR sensors is used for computing the gaiting analysis. We have used Arduino microcontrollers for data collection from the pair of shoes incorporated with sensors. The data gathered here is transmitted via Bluetooth protocol to the core FPGA where the main Gaiting Analysis is performed and fall detection is achieved here. The novelty in this paper is we have used the FPGA board for Fall Detection because it has high performance, less power consumption and parallel processing support. So using FPGA is the best option in this project. This project can be broadly applied across Healthcare and Military applications. The detected fall can be further improved and it can be prevented by Fall Prevention prototypes. In this study, we have successfully achieved the desired results, and all the proposed methods were finally verified through simulations and experiments.***

Index Terms: Fall detection, FSR sensors, Bluetooth, Arduino, FPGA.

I. INTRODUCTION

WALKING is a basic ability that helps people engage in their daily lives, and it is one of the most important abilities that allows us to stay healthy throughout our lives. Walking involves a repetitive procession of the limb movement to guide the body forward while maintaining stable posture synchronously. The leg which leaves the ground swings forward from the hip during forward motion. Then the leg strikes the heel on the ground, and rolls to the toe.

Walking is described by the gait [1]. Gait analysis is a process or study of human walking behavior,

Revised Manuscript Received on May 30, 2020.
***** Correspondence Author

Dr. M. C. Chinnaiah*, Electronics and Communication Engineering, B. V. Raju Institution of Technology, Narsapur, HYD, India. chinnaaiah@bvrit.ac.in.

A. Murali Krishna, Electronics and Communication Engineering, B. V. Raju Institution of Technology, Narsapur, HYD, India. muralikrishna1698@gmail.com**.**

C. V Dheeraj, Electronics and Communication Engineering, B. V. Raju Institution of Technology, Narsapur, HYD, India. cvdheeraj977@gmail.com.

D. Abhishek, Electronics and Communication Engineering, B. V. Raju Institution of Technology, Narsapur, HYD, India. 17215A0401@bvrit.ac.in.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

With the help of instruments which can measure the body movements, mechanics. A standard gait requires one foot to be placed forward with the second having the same distance to the first. A normal individual gait, often referred to as the normal gait, is a very effective gait pattern in terms of power and walking speed so a human can walk effortlessly for a considerable amount of time. In addition, the normal gait allows the human to remain light-footed so that she/he can climb and go downstairs effortlessly, change walking directions and avoid obstacles instantly.

Falls are a major problem for older people. There are many reasons why older people are more likely to fall. Some of the reasons like chronic health conditions, partial impairments or physically challenged. So to prevent the fall there are many fall detection devices, such as walking-aid cane robots [2], smart helmets etc. These falls detection devices can detect the fall prior to the occurrence of a fall. Also there are different kinds of approach towards the fall detection like leg-motion based fall detection, center of pressure fall detection [2].

Unfortunately, there are quite a number of people who lose their ability to walk because of injuries, illnesses or disabilities. As mankind ages, the number of the elderly suffering from walking dysfunctions is growing rapidly. The growing elderly population also leads to a shortage of young people to look after them. Therefore, there is a serious need for development of walking-aid robots that can partially replace nurses and therapists. FPGA boards are used in different applications due to its high performance onboard processing unit with less power consumption and reconfigurable characteristics. It also has a parallel processing support which is very useful in obtaining an efficient output when there are so many input and output terminals. Only few of them have designed the Fall Detection using FPGA. This is the novelty and these all are the reasons for picking up the FPGA board in our project.

The rest of the paper is organized as follows: The related work done by other researchers is referred in the section II. The control design of the proposed project described in Section III. The fall detection Prototype based on the FSR sensor values is discussed in Section IV. In Section V we discussed the verification of the effectiveness of the proposed model and methods through simulations and experiments, respectively. The conclusion is presented in section VI.

II. RELATED WORK

Until now, many researchers have proposed several kinds of walking support systems for assisting elderly and handicapped people. Hirata *et al.* developed a passive-type intelligent prototype walking support system known as RT Walker.

Further the RT Walker is included with the environmental information which provides obstacle/ step avoidance and gravity compensation functions [3]-[5]. Huang *et al.* developed a three-wheeled Omni-directional cane robot with a vision-based fall detection method. The fall model is created using the head and leg position information recorded simultaneously [6]. Kong *et al* proposes the tendon-driven exoskeletal assistive device, which consists of a wearable exoskeleton [7]. Kiguchi *et al* proposed a robotic exoskeleton for human upper-limb motion assist [8]. Mohammed *et al* addresses the latest
Advanced research in actuated lower-limb wearable robots and, particularly, the knee joint active orthosis [9].

Various methods have been proposed on gait analysis such as Mazhar *et al* developed Real-Time gait phase detection with the help of wearable sensors [10]. Ma *et al* developed a real-time gait modeling and analysis. It is evaluated by the method based on ground reaction forces (GRF's) determined by a couple of smart insoles [11]. Agostini *et al* proposed a general purpose algorithm for classifying gait cycles, which can be used independently of the pathology considered in clinical gait analysis, provided that a reliable foot-switch signal can be collected [12]. Khandelwal *et al* proposes a novel approach which shows how domain knowledge of human gait can be incorporated in time-frequency analysis to develop a robust algorithm that can detect gait events from long-term accelerometer signals [13].

There are few researches dealing with fall detection, Yan *et al* proposed a cane-type walking-aid robot with a ball joint is developed to help walk older people with a new concept of Human-Robot Coordination Stability (HRCS) [14]. Merrouche *et al* proposed a fall detection method using depth cameras. This method combines human shape analysis, head tracking and center of mass detection by exploiting the advantages of Kinect [15]. Wu *et al* proposed novel fall detection techniques were developed based on time-frequency analysis [16]. Nguyen *et al* analysed a threshold based fall detection algorithm for two Smartphone data collected [17]. Although falling is a major cause of fatal injury for the elderly, fall detection and prevention methods have not been adequately developed for current walking-aid robots. There is still great space for advancement of present-day walking-aid robots.

III. CONTROL DESIGN FLOW

From the figure 1 we can analyse the control design flow of our proposed project. Our main focus is to perform the gaiting analysis and detect the fall of the user. Generally a user's walking status can be classified into two types, “normal walking” and “abnormal walking”. Whenever the user’s walking pattern deviates from normal walking, fall is almost possible and we have to detect that fall direction by using gaiting analysis. The main “Fall Detection” function module is analysed at the FPGA side based upon the real time walking status of the user. This module is illustrated in Section IV. Whenever the power supply is given all the modules interconnected start functioning. We have divided our project flow into separate modules & parts and worked on it for better understanding and analysing purposes.

A. Sensor Module:

In this module we will have the entire sensors used in our project. The wearable sensor system we have used here comprises Force Sensing Resistor (FSR 402) sensors. Four FSR sensors each are incorporated at the specific pre-assumed spots at the insole of the footwear (as shown in figure 4). These spots are the locations where the foot and the ground are made in contact with each other. From these sensors we obtain the real time foot-ground reaction force values.

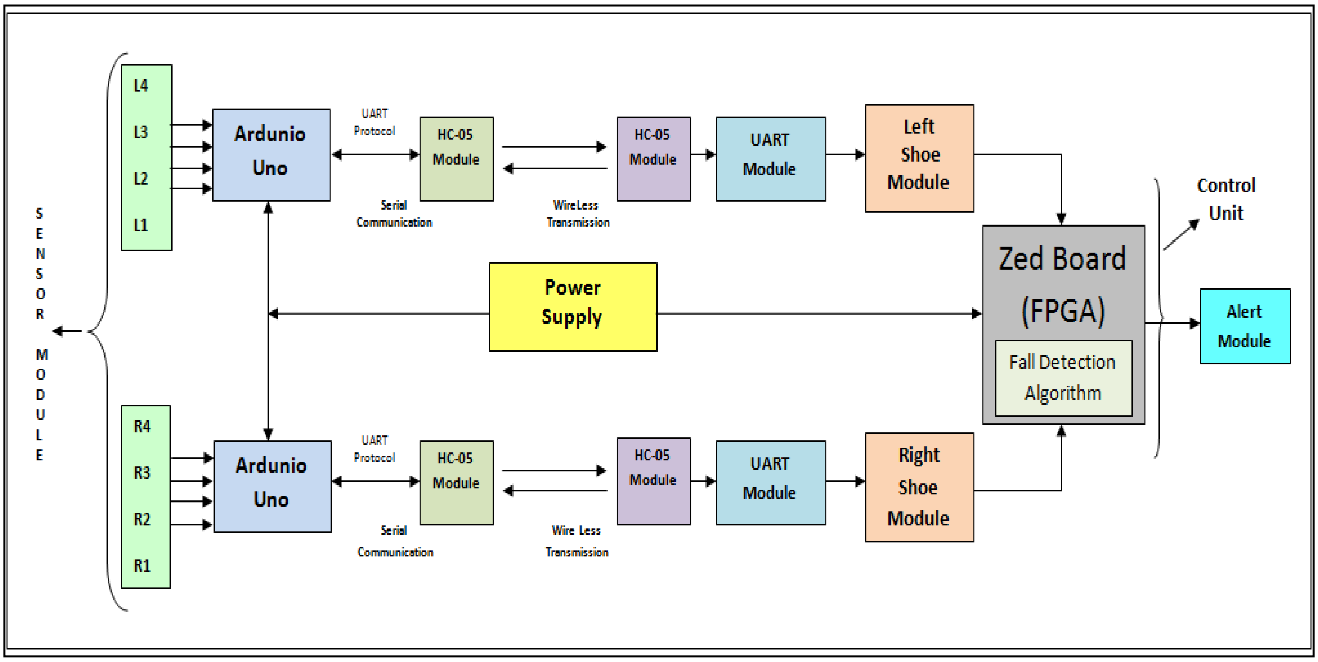

Fig. 1. Block Diagram

Force sensing resistors, or FSRs, area unit sturdy chemical compound thick film (PTF) devices that exhibit a decrease in resistance with increase operative applied to the surface of the detector. This force sensitivity is optimized to be used in human bit management of electronic devices like automotive physical science, medical systems, and in industrial and artificial intelligence applications. From these FSR sensors we are able to retrieve the bottom reaction force of its several feet for sleuthing the autumn of the user. The FSR detector physically incorporated onto the footwear sole is shown physically within the below figure 3.

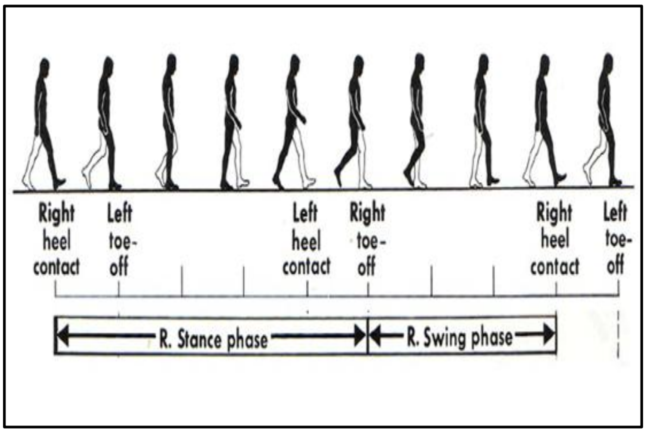

Fig. 2. Human Standard Walking Pattern

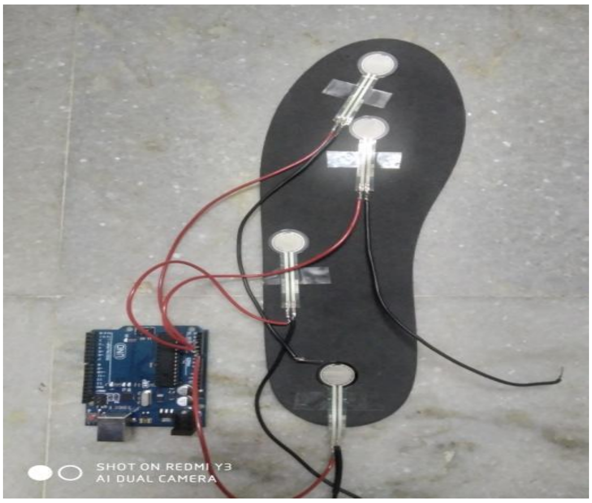

Fig. 3. Insole FSR Sensor System

B. Arduino Module:

These FSR sensor values of corresponding feet are collected at the Arduino UNO microcontroller. An Arduino microcontroller is used to retrieve the analog voltage data from the four sensors attached to its analog pins. Arduino has an inbuilt ADC (Analog to Digital Converter) which is helpful to directly convert the analog values obtained from the sensors to digital values. These digital values can be obtained from the digital pins of the Arduino board. The FSR and Arduino interfacing is shown physically in the figure 3. The functioning of this Arduino module is to just collect the data and transmit it to the ZedBoard FPGA. We have used wireless communication protocol to transfer the data from Arduino to FPGA. Arduino works on serial communication. It has an inbuilt UART protocol in it. So whenever we interface any of the external devices, it is very easy to build a communication.

C. Wireless Transmission Module:

These sensor values obtained from the Arduino are wirelessly transmitted to Xilinx Zed-Board FPGA using HC-05 Bluetooth module. HC-05 Bluetooth module is used to initialise serial communication with Arduino and other microcontrollers using serial port (USART). It can be set as both Master and Slave for data transmission wirelessly. In simpler words, it is a replacement for serial port communication. We have established a connection for the two HC-05 modules in advance. The connection is established in such a way that the HC-05 bluetooth module near to the Arduino is set as the Master module. The HC-05 bluetooth module near to the FPGA is set as the Slave module.

Once the connection is established between the two modules, whenever we give the power supply automatically they both will get bound and the data transmission takes place between them. The data is transmitted bit by bit here. The HC-05 bluetooth module at the FPGA side cannot be directly connected to FPGA. Because like Arduino, FPGAs do not have any in-built UART protocol. So that is why we have used the UART module to interface and establish a successful communication with HC-05 Bluetooth module and Zedboard FPGA.

D. UART Module:

UART stands for "Universal Asynchronous receiver-transmitter". Data transmission takes place within the sort of information packets, starting with a begin bit, wherever the commonly high line is force to ground. When the beginning bit, five to nine information bits transmit in what's called the packet's information frame, followed by Associate in nursing non-mandatory bit to verify correct information transmission. Finally, one or a lot of stop bits square measure transmitted wherever the road is about to high. This ends a packet. Here the Verilog code for UART module is written and instantiated further in the shoe modules (that are left shoe and right shoe modules). Here we will obtain the string of data because the data is received through serial communication that is bit-by-bit.

E. Left & Right Shoe Modules:

Now the data which is in the string format is subdivided into their corresponding variables. That means the four sensor values from the Arduino are clubbed together and concatenated in the form of a string and finally a string set of data is received at the shoe module side. The concatenation takes place by clubbing together the ASCII values of each and every digit of four sensor values. The same happens in the other Arduino also. Therefore both the set of string values are obtained at the left and shoe modules respectively. Now the string set of data contains the each 4 sensor values. So these 4 sensor values are mapped and segregated with their respective variables. Now ultimately the 4 sensor values are ready at each of the left and right shoe modules. Now further for performing the proposed algorithm we will be sending those values in the top module of FPGA (Control Unit) by instantiating into them.

F. Control Unit:

This is the main controlling part of our proposed project. As already mentioned above, the 4 sensor values of both left and right foot are ready with the left shoe module and right shoe module. Our algorithm is designed here. Gaiting Analysis is performed here and an algorithm is proposed here for Fall Detection. Then both the foot sensor values are clubbed together and the user walking status is analysed. Whenever the user's normal walking status is deviated, we consider it to fall under the user's abnormal walking status. Here the fall detection module comes into picture. Our proposed method for fall detection by using gaiting analysis is discussed further clearly in Chapter IV. ZedBoard may be a affordable development board for the Xilinx Zync-7000 All Programmable SoC. This board contains everything necessary to form a UNIX operating system, Android, Windows or different OS/RTOS-based style. All the devices' working frequencies are matched using a clock divider code.

G. Alert Module:

This is the final module of our proposed project. We will be ready with the output with this module. From the previous control unit we have obtained the fall detection direction. So that fall detection status is further extended as an alert to the user. The alert is given to the user in advance before falling occurs to prevent the falling. It can be helpful to the user and also in the health care services while looking after the patient's case.

IV. FALL DETECTION

In this study, initially we analysed the human walking motion (as shown in figure 2). We have incorporated the FSR sensors at the insole at the specific locations (as shown in the figure 4 below) which mainly from the point of contact for foot and the ground.

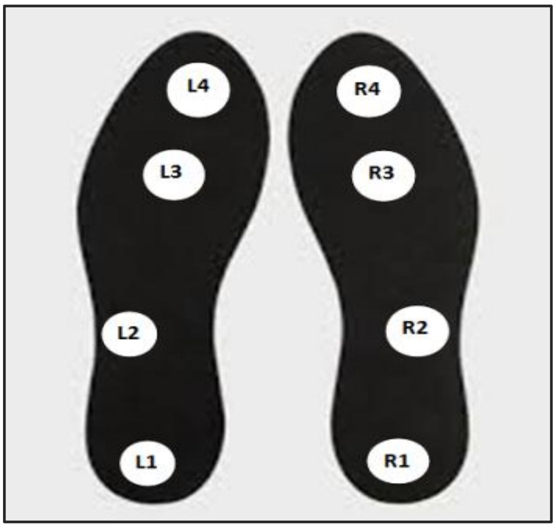

Fig. 4. Assumed Sensor Locations

All the sensors on the right foot are indicated as R1, R2, R3 and R4. Similarly the sensors on the left foot are indicated as L1, L2, L3 and L4. From these sensors we get the force-ground reaction values. From these values we can determine the walking status of the user. For instance we have considered that whenever the sensor makes a point of contact with the ground we assume it to be '+'. Else we assume it to be '-'. Based on these assumptions we have constructed a standard table from the human walking motion as shown in Table 1.

Table-I Standard Cases of Human Motion

S. no	L1	L2	L3	L4	R1	R2	R3	R4
1	-	-	+	+	+	-	-	-
2	-	-	-	+	+	+	+	+
3	-	-	-	-	+	+	+	+
4	-	-	-	-	+	+	+	+
5	+	-	-	-	-	-	+	+
6	+	+	+	+	-	-	-	+
7	+	+	+	+	-	-	-	-
8	+	+	+	+	-	-	-	-

The above standard table indicates the "normal walking" type of user's walking status. We consider this table as a reference for analysing the further cases of fall detection.

Whenever the user's walking status deviates from the normal walking pattern, then we assume it to come under fall case. These deviations may occur if in case the user gets stumbled or gets pushed by some external forces. So we have constructed a real time table by comparing the standard table. It consists of the sensor data whenever the user gets deviated from normal walking status. From this deviation we can identify the direction of fall of the user.

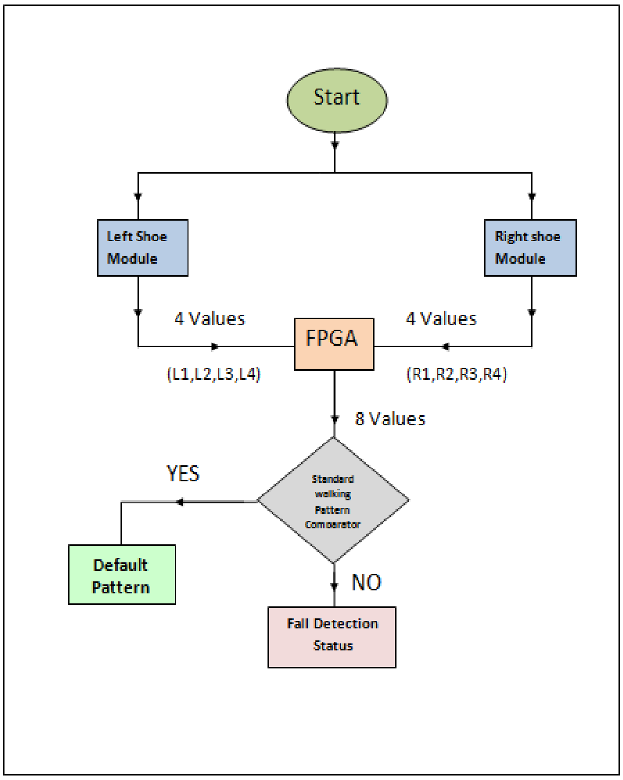

Fig. 5. Flow Chart for Proposed Algorithm

Above is the figure 5 which indicates the flow of our proposed algorithm. Based upon the Gaiting Analysis (standard human walking motion pattern and standard table) done before we should apply them in real time cases. Sometimes falling occurs due to some external forces or stumbling. That falling pattern is entirely different from the standard human walking pattern.

- Whenever the power supply is given to the circuit the start module is initialized.
- From the flow chart it is clear after the start we will be ready with the left shoe and right shoe modules. They will be with-holding the four FSR sensor values each of their corresponding feet.
- Then the both shoe module's data will be transmitted to the main control unit by performing instantiation. In the control unit we will be writing the main proposed algorithm.
- If the 8 FSR sensor values combinations deviate from the standard human walking pattern then we have moved to display the status of the Fall Detection and pass some alert to the user.
- If the real time 8 sensor values match with the standard human walking pattern, then we will be considering it as a standard case of walking (default pattern).
- But we are focused with the fall cases, so the fall detected status is indicated below. By following the same flow chart we have obtained the 10 real time cases when compared with standard walking cases.

We have considered a common critical/cut-off pressure value for the sensors. This value indicates the abnormal pressure value which is more than expected value. So fall is assumed in the particular sensor whenever it crosses the critical pressure value. Then combining together the entire sensor values of both left and right foot, we can determine the direction of fall of the user.

In our Verilog code we have considered **220mv** as the Critical Pressure reading for FSR sensors. If the corresponding sensor value crosses the value 220mv, the fall is detected on that particular sensor and we have noted '↑' for that respective sensor, else '↓'. So based on the above real time table we have written an algorithm for Fall Detection. Below table indicates all the possible cases for falling off a person. Based upon the values we have obtained from the sensors we have analyzed the Fall Direction of a person/user.

Table- II Real Time Table

S. no	Fall status	L1	L2	L3	L4	R1	R2	R3	R4
1	Left	↓	↑	↓	↓	↑	↓	↓	↓
2	Front	↓	↓	↓	↑	↓	↓	↓	↓
3	Front	↓	↓	↓	↓	↓	↓	↑	↑
4	Right	↓	↓	↓	↓	↓	↑	↓	↓
5	Back	↓	↓	↓	↓	↑	↓	↓	↓
6	Right	↑	↓	↓	↓	↓	↑	↓	↓
7	Front	↓	↓	↓	↓	↓	↓	↓	↑
8	Front	↓	↓	↑	↑	↓	↓	↓	↓
9	Left	↓	↑	↓	↓	↓	↓	↓	↓
10	Back	↑	↓	↓	↓	↓	↓	↓	↓

Note:

- Here ↑ indicates sensor value greater than 220.
- Here ↓ indicates sensor value lesser than 220.

V. EXPERIMENTAL RESULTS

A. Simulation Results:

Thus we have obtained the simulation results for our project as shown in the figure 6 below. For better analysis of our algorithm and to improve the ease in understanding the results, we have displayed one of the cases of fall

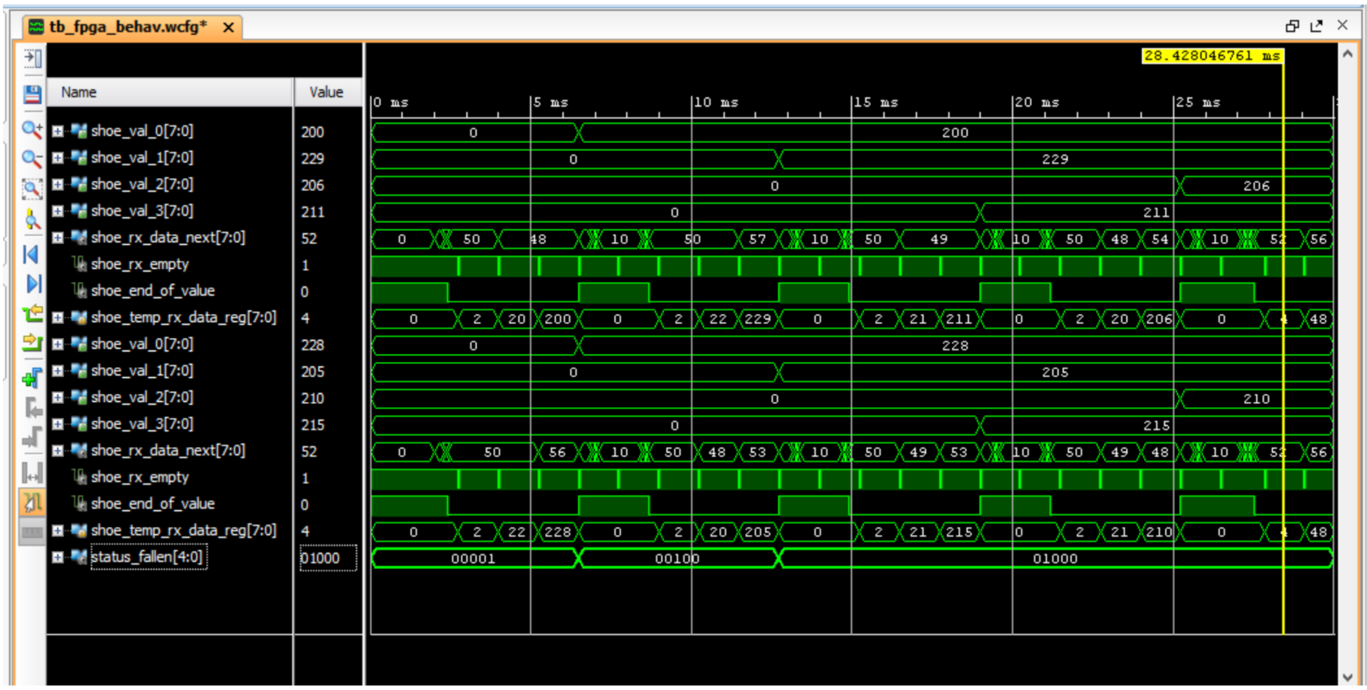

Fig. 6. Simulation Results

Detection. The 8 sensor values that are 4 from each foot have been received continuously to the FPGA from Arduino via Bluetooth protocol. These values are approaching the FPGA via UART protocol in a string format. In UART protocol data is converted into ASCII format and further transferred using string format. Each sensor data is attached with start and stop bits. So whenever a new line character is approached we assume that one sensor value is transmitted completely. In our case at a time four sensor values are transmitted from each Arduino to FPGA via Bluetooth using UART protocol. So here a set of data means we are transmitting 4 sensors data at a time as a group.

Then after some slight manual delay we will obtain the next set of data.

The 4 sensors data is entirely digit by digit converted into ASCII format and sent as a whole concatenated new set of string.

ASCII values:

0 → 48; 1 → 49 ; 2 → 50 ; 3 → 51…… 8 → 56 ; 9 → 57
Carriage Return -> 13
New Line Character ->10

For understanding let us explain this transmission using a real time example. Let us assume the four sensor values are 20, 21, 22, 23 respectively. So these 4 values represent the set of data to be transmitted as a string. Below is the fashion in which it is transmitted. So the above 4 values are transmitted as a string indicated below:
String= {50, 48, 13, 10, 50, 49, 13, 10, 50, 50, 13, 10, 50, 51, 13, 10}

The above is the string containing 4 sensor values from a foot. Similarly from another foot another 4 sensor values are also achieved in the form of a string which is indicated in the above example. So finally by considering the 2 sets of data comprising 8 sensor values we will analyse the Fall Detection zone of the user as per our proposed algorithm. Similarly with slight manual delay from the Arduino we will be receiving the 8 sensor values to the FPGA continuously whenever the power supply is given. So for each case we will be detecting the Fall zone of the user and give him an alert signal before he falls unexpectedly.

Now let us analyse our simulation results. From the below figure, the starting four values "shoe_val_0, shoe_val_1, shoe_val_2, and shoe_val_3" are 4 sensor values for the Left Shoe. Similarly the middle four values "shoe_val_0, shoe_val_1, shoe_val_2, and shoe_val_3" are 4 sensor values for the Right Shoe. Here **shoe_rx_data_next [7:0]** (2 variables i.e., one for left foot and another for right foot) represents the variable capable of storing the Strings coming from the Arduino board. For each foot the 4 sensor values are transmitted in the form of a string as already discussed above to **shoe_rx_data_next [7:0]**. Once the complete string is received to the FPGA side, then further bifurcations are performed and the corresponding sensor values are reallocated to their respective register variables (i.e., shoe_val_0, shoe_val_1, shoe_val_2, and shoe_val_3). Similarly the other foot values are also divided and reallocated to their respective register variables. So finally we have obtained the 8 sensor values from both the feet.

Now we have the 8 sensors data in our hand in the form of shoe_val_0, shoe_val_1, shoe_val_2, and shoe_val_3 for left and right feet. So we should look into our proposed algorithm based upon these 8 sensor values. On deep analysis, we have assumed that for approximately every **6ms** we are getting each sensor value from the Arduino to the FPGA. So finally around **24ms** we will get the both left and right foot data (total 8 sensor values). The output value before 24ms is an unnecessary output value and we do not focus on that output. We only concentrate on the output which is occurring around every **24ms**. This output indicates the direction of the fall of the user.

In the above figure 5.1 we have received the 8 sensor values as shown below:

Left Shoe Sensor values:

shoe_val_0 = 200 (L1) ; shoe_val_0 = 229 (L2) ; shoe_val_0 = 206 (L3) ; shoe_val_0 = 211 (L4)

Right Shoe Sensor values:

shoe_val_0 = 228 (R1) ; shoe_val_0 = 205 (R2) ; shoe_val_0 = 210 (R3) ; shoe_val_0 = 215 (R4)

We have designed our output in Verilog as shown below:

```
Parameter STATE_DEFAULT    = 5'b00001;
Parameter STATE_FRONT      = 5'b00010;
Parameter STATE_BACK       = 5'b00100;
Parameter STATE_LEFT       = 5'b01000;
Parameter STATE_RIGHT      = 5'b10000;
```

According to our Real Time Cases table since the L2 and R1 are greater than 220 millivolt so the output should be LEFT Fall (5'b01000). The same output we have obtained in the Simulation Results, as shown in the above figure (i.e., in the last line of simulation **status_fallen [4:0] around 24ms** we have got the Left Fall output). So the 1st case of a real time table is obtained successfully. Similarly we can obtain all the 10 cases by using our Verilog code.

So, we have successfully achieved the simulation results and the algorithm which we have proposed here is absolutely working fine. Finally whenever the user's fall is detected in a particular direction an alert is given to the user in advance to avoid the fall due to some external forces or stumbling nature. Partially impaired people have a lot of usage with our proposed project.

B. Device Utilization Report:

We have obtained the Device Utilization Summary report after the synthesis. The utilization percentage of the logic devices is indicated in the table III given below.

Table-III Device Utilization Report

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	169	106400	0.16%
Number of Slice LUTs	158	53200	0.30%
Number of bonded IOBs	10	200	5.00%
Number of BUFGCTRLs	2	32	6.25%

1. Out of 106400 **Slice Registers**, 169 have been utilised in our project and the utilization is **0.16%**.
2. Out of 53200 **Slice LUT's**, 158 have been utilised in our project and the utilization is **0.30%**.
3. Out of 200 **Bonded IOBs**, 10 have been utilised in our project and the utilization is **5.00%**.
4. Out of 32 **BUFGCTRLs**, 2 have been utilised in our project and the utilization is **6.25%**.

Below is the graph between utilization percentage and logic devices in our project after performing the synthesis report.

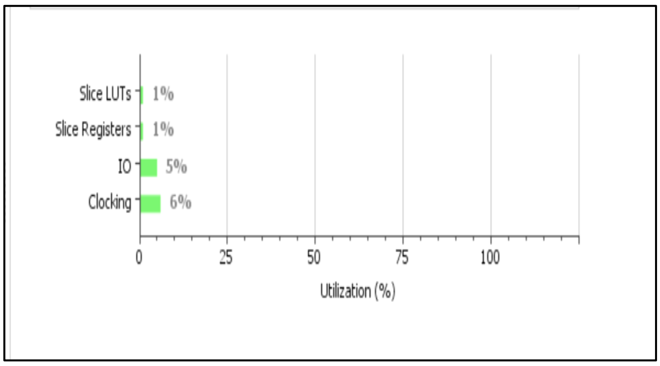

Fig. 7. Device Utilization Summary

VI. CONCLUSION

In this study, Gaiting Analysis is performed successfully based upon the FSR sensor values obtained from the footwear. Whenever the user deviates from the normal walking pattern i.e., if the sensor value crosses the pre-assumed cut off value, fall is detected. The detected fall is given as an alert to the user in advance to ensure the safety of the person. From this continuous inspection of human standard walking patterns based on our proposed algorithm, Fall Detection is finally achieved. It is very much useful to partially disabled people. Protection is must for these kinds of paralyzed people. So keeping in mind these people, we have designed this project. We have scrutinized our proposed algorithm through simulation and finally we have effectively obtained the desired results. It can be applicable and widely used in military, healthcare applications.

In this study, a prototype was designed for analyzing our proposed concept. Furthermore, our concept can be extended to the Fall Prevention prototype. The fall which is detected, should also be prevented immediately to ensure the safety of the user. To improve the quality of the production, the sensors can be replaced with cheaper, robust and highly accurate ones. FSR sensors are cheaper and efficient sensors, but regarding precision, their measurements are +/- 10%. In future, we can consider more cases for Fall Detection and enhance the quality of the project.

REFERENCES

1. J. Perry, Gait Analysis. Thorofare, NJ: SLACK, Inc., 1992.
2. P. Di et al., "Fall Detection and Prevention Control Using Walking-Aid Cane Robot," in IEEE/ASME Transactions on Mechatronics, vol. 21, no. 2, pp. 625-637, April 2016.
3. Y. Hirata, A. Hara, and K. Kosuge, "Passive-type intelligent walking support system 'RT walker'," in Proc. IEEE/RSJ Int. Conf. Intell. Robots.
4. Y. Hirata, A. Hara, and K. Kosuge, "Motion control of passive intelligent walker using servo brakes," IEEE Trans. Robot., vol. 23, no. 5, pp. 981– 990, Oct. 2007.
5. Y. Hirata, A. Muraki, and K. Kosuge, "Motion control of intelligent passive-type walker for fall-prevention function based on estimation of user state," in Proc. IEEE Int. Conf. Robot. Autom, 2006, pp. 3498–3503.
6. J. Huang, P. Di, K. Wakita, T. Fukuda, and K. Sekiyama, "Study of fall detection using intelligent cane based on sensor fusion," in Proc. IEEE Int. Symp. Micro-Nano Mechatronics Human Sci., 2008, pp. 495–500. Syst., 2004, vol. 4, pp. 3871–3876.
7. K. Kong and D. Jeon, "Design and control of an exoskeleton for the elderly and patients," IEEE/ASME Trans. Mechatronics, vol. 11, no. 4, pp. 428–432, Aug. 2006.
8. K. Kiguchi, T. Tanaka, and T. Fukuda, "Neuro-fuzzy control of a robotic exoskeleton with EMG signals," IEEE Trans. Fuzzy Syst., vol. 12, no. 4, pp. 481–490, Aug. 2004.
9. S. Mohammed, Y. Amirat, and H. Rifai, "Lower-limb movement assistance through wearable robots: State of the art and challenges," Adv. Robot., vol. 26, nos. 1/2, pp. 1–22, 2012.
10. O. Mazhar, A. Z. Bari and A. A. M. Faudzi, "Real-time gait phase detection using wearable sensors," 2015 10th Asian Control Conference (ASCC), Kota Kinabalu, 2015, pp. 1-4.
11. H. Ma and W. Liao, "Human Gait Modeling and Analysis Using a Semi-Markov Process With Ground Reaction Forces," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 25, no. 6, pp. 597-607, June 2017.
12. V. Agostini, G. Balestra and M. Knaflitz, "Segmentation and Classification of Gait Cycles," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 22, no. 5, pp. 946-952, Sept. 2014.
13. S. Khandelwal and N. Wickström, "Gait Event Detection in Real-World Environment for Long-Term Applications: Incorporating Domain Knowledge Into Time-Frequency Analysis," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 24, no. 12, pp. 1363-1372, Dec. 2016.
14. Q. Yan, J. Huang and Z. Luo, "Human-robot coordination stability for fall detection and prevention using cane robot," 2016 International Symposium on Micro-NanoMechatronics and Human Science (MHS), Nagoya, 2016, pp. 1-7.
15. F. Merrouche and N. Baha, "Depth camera based fall detection using human shape and movement," 2016 IEEE International Conference on Signal and Image Processing (ICSIP), Beijing, 2016, pp. 586-590.
16. M. Wu, X. Dai, Y. D. Zhang, B. Davidson, M. G. Amin and J. Zhang, "Fall Detection Based on Sequential Modeling of Radar Signal Time-Frequency Features," 2013 IEEE International Conference on Healthcare Informatics, Philadelphia, PA, 2013, pp. 169-174.
17. H. Nguyen, F. Zhou, F. Mirza and M. A. Naeem, "Fall Detection Using Smartphones to Enhance Safety and Security of Older Adults at Home," 2018 Eleventh International Conference on Mobile Computing and Ubiquitous Network (ICMU), Auckland, New Zealand, 2018, pp. 1-2.

AUTHORS PROFILE

Dr. M. C. Chinnaiah, Professor in Electronics and communication department in BVRIT, Narsapur, Hyderabad. He has Ph. D., M.Tech, B.Tech in ECE. He has a teaching experience of 19 years. He has published various research papers in many journals. He is member of IEEE, ISTE-LM.

A. Murali Krishna, Pursuing final year of B.Tech in Electronics and communication engineering, BVRIT, Narsapur, Hyderabad.

C. V Dheeraj, Pursuing final year of B.Tech in Electronics and communication engineering, BVRIT, Narsapur, Hyderabad.

D. Abhishek, Pursuing final year of B.Tech in Electronics and communication engineering, BVRIT, Narsapur, Hyderabad.