

# DESIGN AND VLSI IMPLEMENTATION OF ANTI-COLLISION ENABLED ROBOT PROCESSOR USING RFID TECHNOLOGY

Joyashree Bag, Rajanna K M and Subir Kumar Sarkar

Dept. Of Electronics and Telecommunication Engg, Jadavpur University, Kolkata

joyashree\_bag@yahoo.co.in

rajanna\_km@ieee.org, subirsarkar@ieee.org

## ABSTRACT

*RFID is a low power wireless emerging technology which has given rise to highly promising applications in real life. It can be employed for robot navigation. In multi-robot environment, when many robots are moving in the same workspace, there is a possibility of their physical collision with themselves as well as with physical objects. In the present work, we have proposed and developed a processor incorporating smart algorithm for avoiding such collisions with the help of RFID technology and implemented it by using VHDL. The design procedure and the simulated results are very useful in designing and implementing a practical RFID system. The RTL schematic view of the processor is achieved by successfully synthesizing the proposed design.*

## KEYWORDS

*Navigation, RFID, RTL schematic, Simulation, Synthesis, VHDL*

## 1. INTRODUCTION

RFID stands for Radio Frequency Identification, a term that describes any system of identification, wherein an electronic device that uses radio frequency or magnetic field variations to communicate is attached to an item. The two most talked- components of an RFID system are the tag, which is the identification system attached to the item to be tracked and the reader, which is a system unit that can recognize the presence of RFID tags and read the information stored on them. The reader can then inform another system about the presence of the tagged items. The system with which the reader communicates usually runs on software that stands between readers and applications. This software is called RFID middleware.

In a typical RFID system [1], passive tags are attached to an object such as goods, vehicles, human beings, animals and shipments, while a vertical/circular polarization antenna is connected to the RFID reader. The RFID reader and tag can radio-communicate (whether mobile or fixed) with each other using a number of different frequencies and currently most RFID systems use unlicensed spectrum. The common frequencies used are low frequency (125 KHz), high frequency (13.56 MHz), ultra high frequency (860–960 MHz) and microwave frequency (2.4 GHz). The typical RFID readers are able to read or detect the tags of only a single frequency. But multimode readers are becoming cheaper and popular which are capable of reading the tags of different frequencies [2].

The factors that influence detectability of the mobile RFID tag with fixed RFID reader includes [25], [26]:

- Reader type
- Contents of the object
- position, type and direction of tag
- Moving speed of mobile tag
- Angle of antenna, Power, type, gain, frequency range and number of antennas
- Working environment of the RFID system, etc

In multi-robot environment, robots are fitted with RFID readers and tags. The power received by the tags from the RFID readers can give information about the location of robots. When the robots are in operation, there is a possibility of physical collision of robots with other robots as well as with physical objects viz., walls, pillars etc. There are some fixed tags on the walls. Different techniques to measure the range of robots from the RFID readers are described in literatures [3], [4]. In this paper, we describe a novel technique for anti-collision of robots and its implementation in real time environment and we consider the power response of mobile tag as influence factor. The proposed algorithm is implemented in VHDL and tested using Xilinx simulation software. The synthesizable module of the processor is achieved after successful synthesis [18]-[20].

This paper is organized as: Section 2 describes the previous work and literature review. Section 3, introduces the proposed scheme. Section 4 discusses the design of robot processor and test bench simulation. Section 5 gives the applications of the proposed scheme and finally we conclude with remarks in Section 6.

## 2. LITERATURE REVIEW

In the field of mobile robot navigation using RFID technology numerous researches have been carried to develop novel and efficient algorithms [3], [4]. Different types of navigation techniques are available like dead-reckoning-based, vision-based, behavior-based and landmark-based techniques [4]-[7]. Dead-reckoning navigation system has small precision errors and sensor drifts which may lead to increasing cumulative errors in the robot's position and orientation [8]. Bing Jiang et. al. [9] have taken a novel approach in detecting the motion of passive RFID tags within the range of the antenna. The work of Myungsik Kim et. al.[10] highlights another novel improvement in the estimation of the Direction Of Arrival (DOA) of signals from a transponder. This enables the robot to continuously monitor and control the overall system and ensures reliable operation. Sunhong Park and Shuji Hashimoto have proposed a method which effectively estimates both the location and the pose of a mobile robot during navigation [11]. The proposed algorithm uses only passive RFID is able to estimate the robot's location and orientation more precisely by using trigonometric functions and the IC tags Cartesian coordinates in a regular grid-like pattern. In the field of physical anti-collision using RFID, research is still on, with contributions from Todd M. Ruff and Drew Hession-Kunz, with their work on a collision warning system designed using RFID technologies [12]. The location measurement of the robots plays a vital part in the prevention of collisions in the robots. We can easily mention the work of Toshihiro Hori et. al. [13], where they proposed a new position estimation method of RFID tags by using a probabilistic approach. In this method, mobile objects (person, robot, etc.) with RFID readers estimate the positions of RFID tags with plural communication ranges. A novel approach in estimating the location of a robot is given in the works of Hyung Soo Lim et. al.[14]. They proposed an error-compensation algorithm based on the relations of the localization error to the gap between the tags and the velocity of the robot. Md. Suruz Miah and Wail Gueaieb have proposed an RFID-based robot navigation system with a Customized RFID Tag Architecture [15]. The RFID system with anti-collision algorithm is described with various aspects in the literature by Yu Song-sen et.al. [16] and Narek Pezeshkian et.al.[17]. S. M. A. Motakabber et.al has presented the VLSI design of an anti-collision protocol in the article [21]. We have developed an algorithm for the proposed system based on these approaches and techniques.

### 3. PROPOSED SCHEME

In a multi robot environment, a number of robots are moving in a same work space. The work space may be a closed room or an open space. To ensure these robots do not collide either with each other or with static objects while they are in moving. Different types of robot processor are commercially available, but our aim is to modify the design effectively combining the merits of VLSI design with the RFID technology which detects tagged item within a fraction of second and without line of sight.

A special processor designed is fitted with the robots. Writable active tags are fitted on the robots. These tags on the robots have a special type of identity as shown in Fig. 1. For the tags fitted with static objects, 'State bit' is "00" and for dynamic tag 'State bit' is "01" and Reader identifies these bits and processes accordingly. The robot is also fitted with an RFID reader.

Fig. 2 shows the basic structure of the robot along with the tag, reader and the processor integrated on it. The robots are cylindrical in shape and fitted with wheels to facilitate its dynamic movement.

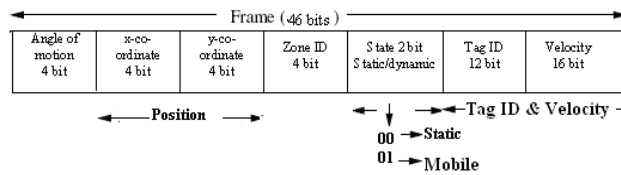


Fig.1. ID Scheme of the tag on robot.

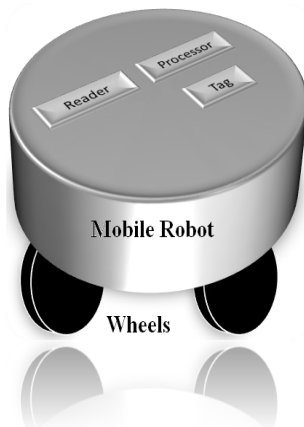


Fig. 2. The robot Configuration

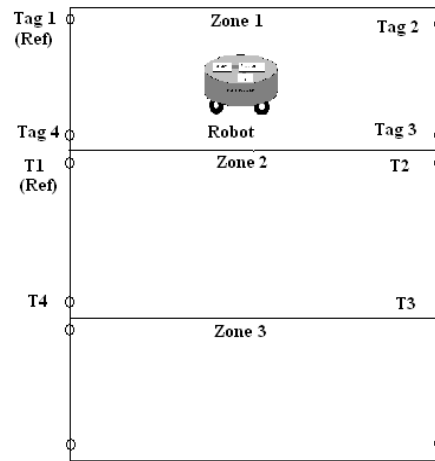


Fig. 3. The Zone Configuration

The area through which the robots move is divided into different zones. These zones are configurable depending on the site situation. In every zone, a geographical region configured as per application is fitted with a number of active RFID tags. These tags are fitted on the walls or static objects like pillars etc. within that zone. The zone configuration and the arrangement of tags are shown in Fig 3. Out of these tags, a particular tag is marked as the reference tag. The zone identity of a tag is contained in 'Zone Id' which is of 4 bits. The 'State bit' is to indicate whether the tag is static or dynamic (Static tags 00, tags on robots 01). The reader continuously reads the tag ID and power received from them. This data is transferred to the processor. The processor

determines the zone Id and processes the data frame as per the design. Similarly, the position coordinates is determined. The processor determine the distance of a tag from the reader through Friis free-space propagation equation:

$$P_{reader} = EIRP \times G_{reader} (\lambda/4\pi r^2)$$

Where, EIRP- Equivalent isotropically radiated power.

Greader- Gain of reader antenna

$\lambda$ - Wavelength

r- Distance between transmitting and receiving antennas

Now, the processor stores the tag ID, the determined distance of it from the reader and the zone ID. The zone with reference static tag indicates the zone ID of the robot/reader. The processor measures the position coordinates (x1, y1) of the mobile robot with respect to the reference tag of that zone.

From the Fig.4, Let the distances of robot R1 from reference nodes (Tags) RT1 and RT2 be  $\rho_1$  and  $\rho_2$  respectively. The distance from the static tags of its zone will be used by it to determine its location for the given distance. As in the Fig.4, we consider RT1 and RT2 as two centers of imaginary circles of radii  $\rho_1$  and  $\rho_2$  respectively. They represent the locations of active tags of a zone. R1 is on their common tangent and represents the location of reader of robot1

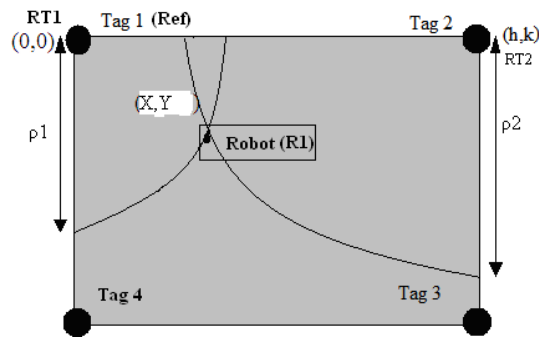


Fig.4. Co-ordinates of the Zone considered

From the equation of circle, we have

$$x^2 + y^2 = \rho_1^2 \tag{1}$$

$$(x-h)^2 + (y-k)^2 = \rho_2^2 \tag{2}$$

Let the intersecting point of the circles (which is the location of robot1) be (x1, y1)

Putting the co-ordinates in eqns (1) and (2) we get

$$x_1^2 + y_1^2 = \rho_1^2 \tag{3}$$

$$x_1^2 + y_1^2 - 2x_1h - 2y_1k + h^2 + k^2 = \rho_2^2 \tag{4}$$

Substituting eqn (3) in eqn (4), we get

$$\rho_1^2 - 2x_1h - 2y_1k + h^2 + k^2 = \rho_2^2 \tag{5}$$

Now, the values of  $\rho_1$ ,  $\rho_2$ , h and k are known. So we get x1 in terms of y1 and from eqn (3) we get x1 and y1.

The values of  $\rho_1$  and  $\rho_2$  are known. This can be converted into the polar plot using simple calculations as shown in Fig 4. The polar coordinates  $(\rho, \psi)$  are defined in terms of Cartesian coordinates as shown in Fig. 5 and is given by:

$$x = \rho \cos \psi \tag{6}$$

$$y = \rho \sin \psi \tag{7}$$

$$\rho = (x^2 + y^2)^{1/2} \tag{8}$$

$$\psi = \tan^{-1} (y/x) \tag{9}$$

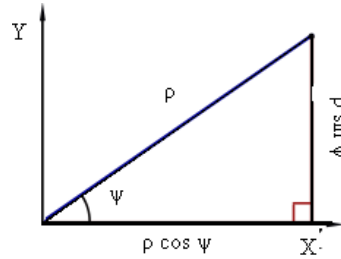


Fig.5. Polar co-ordinate system

Where,  $\rho$  is the radial distance from the origin, and  $\psi$  is the counter clockwise angle from the positive X-axis. After some pre-assigned time 't' the signal strengths of tags are measured at readers and accordingly the polar coordinate of R1 can be calculated. Let the polar coordinates of R1 initially be  $(\rho_1, \psi_1)$  and after time interval 't' is  $(\rho_2, \psi_2)$  as shown in Fig. 6.

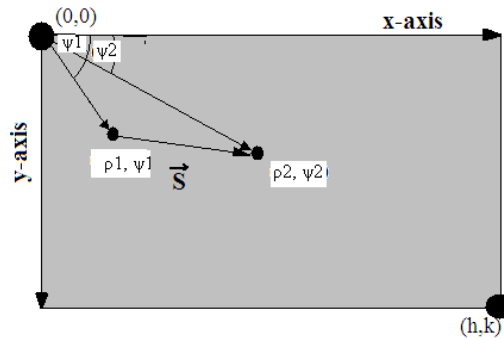


Fig.6. Vector diagram of the displacement vector

The displacement of R1 is given as the vector quantity 's' and is given by,

$$s = (\rho_2, \psi_2) - (\rho_1, \psi_1) \tag{10}$$

$$\text{Where } |s| = \rho_1^2 + \rho_2^2 + 2\rho_1\rho_2\cos [180^\circ - (\psi_1 - \psi_2)] \tag{11}$$

$$\text{Angle, } s = (\rho_2 \cos \psi_1 - \rho_1 \cos \psi_2) / |s| \tag{12}$$

Using the value of the displacement over a fixed time interval, the instantaneous velocity of the robots will be calculated.

$$(|v| = |s|/t) \tag{13}$$

Now, each robot will store its Zone ID, velocity and direction in its tag memory and will transmit this information to the other robots. Moreover, in its memory it will store its instantaneous location coordinates and its destination co-ordinates.

### 3.1 ALGORITHM:

For this algorithm we choose the robot R1 in the Fig 1. The term ‘Safe Distance’ means a minimum distance ‘f’ that must be maintained between two robots or between robot and static object. The processor of the robot will process some steps for successful collision avoidance operation with static as well as dynamic objects. Each robot has one RFID tag and processor attached with it. The processor will follow the algorithm described here.

*Step 1: Calculate ID ( $T_{ID}$ ) and distance ( $T_{dist}$ ) of all tags in range.*

*Step 2: Get Zone ID from  $T_{ID}$ .*

*Step 3: Extract bits 29 and 28 (state bits) from  $T_{ID}$*

*Step 4: If 00 (static) → Step 5*

*01 (dynamic) → Step 12*

*Step 5: Get  $\rho_1, \psi_1$  → Calculate  $x_1, y_1$ .*

*Step 6: If ' $\rho_1$ ' < ' $f$ ,' goto Step 7.*

*else goto Step 8.*

*Step 7: Turn left, move forward. Loop back to Step 6*

*Step 8: Move forward. After ' $t$ ' seconds, derive  $x_2, y_2$  from*

*$\rho_2, \psi_2$ .*

*Step 9: Calculate  $\cos \psi, v, x_2, y_2$ . Generate data frame and write to tag.*

*Step 10: Move forward*

*Step 11: Go to step 1*

*Step 12: Determine  $\cos \psi$*

*Step 13: If  $\cos \psi =$  own angle of movement,*

*goto Step 14*

*else*

*goto Step 15.*

*Step 14: Turn left, move forward, goto step 1*

*Step 15: Measure distance ' $d_1$ ' of other moving robot from it*

*Step 16: If ' $d_1$ ' < ' $f$ '*

*Begin*

*Turn left*

*Move forward*

*goto step 1*

*End*

*Step 17: goto step 1.*

Notations used in the algorithm:

- The distances of robot R1 (Fig. 4) from reference nodes (Tags) RT1 and RT2 be  $\rho_1$  and  $\rho_2$  respectively.
- The polar coordinates of R1 initially be  $(\rho_1, \psi_1)$  and after time interval  $t$  is  $(\rho_2, \psi_2)$  as shown in Fig. 4
- Displacement of R1 is given as the vector quantity 's' eqn. no.(11).
- The instantaneous velocity 'v' of the robots will be calculated from eqn. no.(13)  
( $|v| = |s|/t$ )
- Distance from other moving robots is denoted as ' $d_1$ '.

## 4. THE DESIGN AND SIMULATION OF THE ROBOT PROCESSOR CIRCUIT

During the hardware realization, the simulation outputs are obtained by running the block level model in fixed point quantization with stored integer (SI) data type format [18], [19]. Then, the VHDL code is generated from the subsystems [20], [21]. The VHDL code is simulated in Xilinx 9.2i with appropriate input signals for each operational block. All the operational blocks are then

incorporated into a single processor and simulated [22], [23]. After the successful synthesis of the processor design, we achieve the RTL schematic view and HDL synthesis report as well as the device utilization chart of the processor. The RTL representations and technology views of the proposed system is needed for the proposed hardware design. The operational flowchart is given in Fig.7.

For the proposed robot processor design, we have divided the processor into two sections as part1 and part2. We have taken the locations of the readers at two ends of a side. These may be fitted at two diametrically opposite corners as well. The block diagram is given in Fig. 8.

Part1: It is the 46 bit data-frame generator and it is a part of the robot. It performs operations like:

- Detect the static tags and decode the frame of 46 bits. Determines its existing zone.
- Detect reference tag present in each zone. Determine its position vector in polar coordinates ( $\rho_1, \psi_1$ ).
- After 't' seconds, derive  $x_2, y_2$  from  $\rho_2, \psi_2$ .
- Calculate  $s$  and  $\psi$  as in eqns. 11 and 12.
- Generate data frame as in Fig 1.

Part2: Decision making part of the robot. It performs operations like:

- Detection process is continuous. As soon as a static tag is detected, check the distance ' $\rho_1$ '.
- If  $\rho_1 < f$   
Turn left, move forward. Go to Step 2.
- If a dynamic tag is detected, first decode the frame of 46 bits.
- Detect the zone position, tag ID, velocity and angle of movement of the other robot.
- If  $\cos \psi = \text{own angle of movement}$   
turn left,  
else  
move forward.
- Calculate distance of other robot,  $d_1$ .
- If ' $d_1 \leq f$ ',  
Turn left  
else  
Move forward.

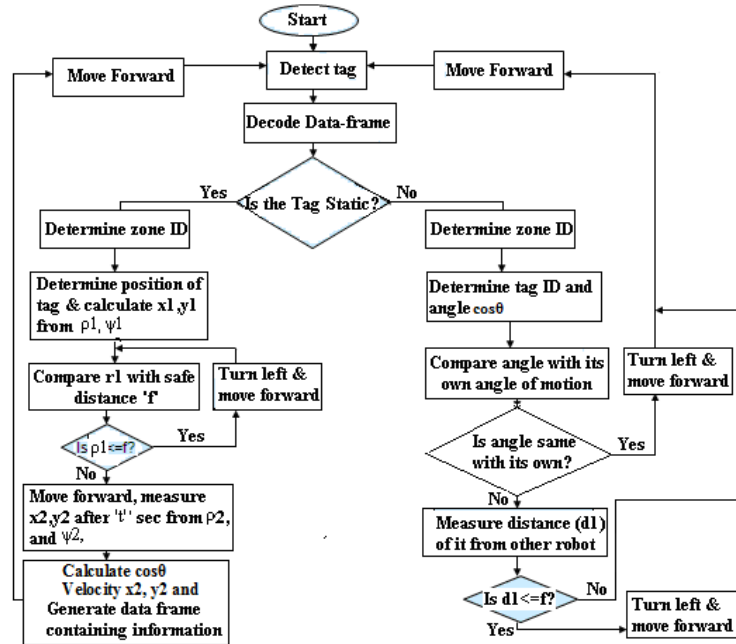


Fig. 7. Flow chart for operation of proposed system

Test bench simulation results of the processor are shown in Fig.9 and Fig.10. Fig. 9 shows the inputs given to the processor of the robot. Tag1 is the tag ID of the robot itself whereas tag2 is the reference tag ID detected by the robot which determines the existing Zone ID of the robot. Here, the state “01” indicates that the robot is in dynamic nature.

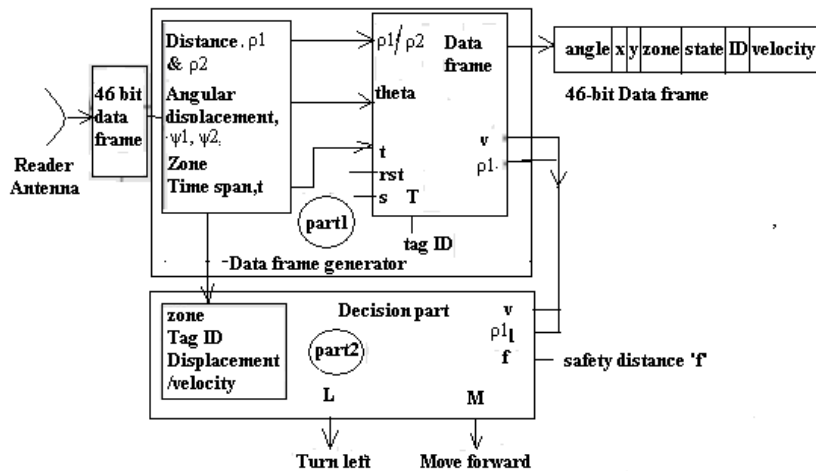


Fig. 8. Block diagram of the proposed processor blocks



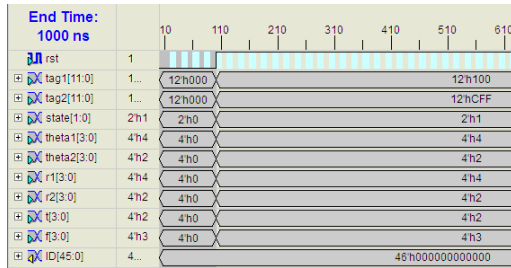


Fig. 9. Input to part1 of the processor

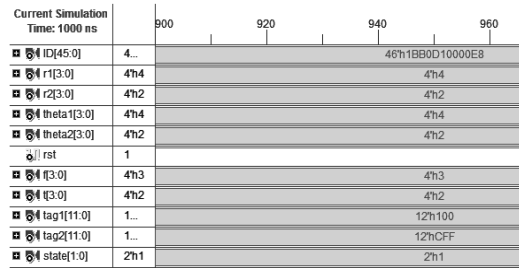


Fig. 10. Output of part1 of the processor

The inputs theta1, theta2, r1, r2 indicates the angles and distances of the robot ( $\psi_1, \psi_2, \rho_1, \rho_2$ ) from the reference tag after a fixed interval of time 't'. Another processor determines these angles and distances from the power received by the antenna of the reader. Input 'f' is the predetermined data indicating the safe distance to avoid collision. This part of the processor will generate 46 bit Data frame ID containing the information shown in Fig. 11.

Angle of motion 4-bit	X-co-ordinate 4-bit	Y-co-ordinate 4-bit	Zone ID 4-bit	State 2-bit static/dynamic	Tag ID 12-bit	Velocity 16-bit
--------------------------	------------------------	------------------------	------------------	-------------------------------	------------------	--------------------

Fig.11. Data frame format

Fig. 10 shows the output obtained after simulation of input test bench. Now the Data frame is generated and transmitted by the robot as well as it stores the data in memory for use in part2 of the processor for taking decision to move forward or turn left and detect other tags and move avoiding collision by continuously checking the distance of other objects and comparing with safe distance 'f' as shown in the flow chart, Fig. 7.

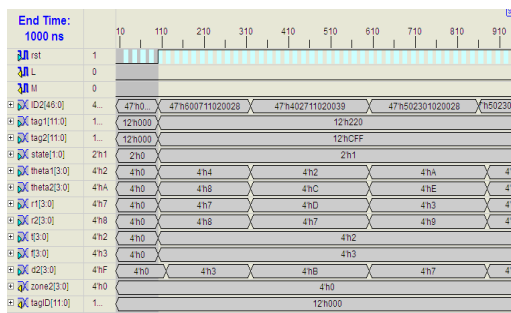


Fig. 12. Input to the processor

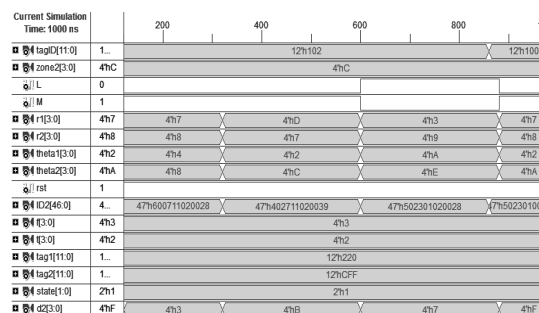


Fig. 13. Output of the processor

Fig. 12 and Fig. 13 show the input and output of the final processor including part1. In part2, there is an input terminal which detects 46-bit data frame transmitted by other robots (ID2). After decoding the frame, it identifies the zone ID, tag ID, angle of motion, velocity etc. of the detected tag of other robot. Now, it compares the data with its own and takes proper decision which is required to avoid collision. Two output terminals 'L' and 'M' helps for this purpose. When 'L' is high and 'M' is low, the robot turns left and try to detect other tags. Similarly, when 'L' is low and 'M' is high, the robot moves forward without any turn. In Fig. 13 we observe that for first two detected tags, 'L' is low and 'M' is high, means the robot will move forward without collision. For the third detection,  $\rho_1$ , the distance of other robot is equal to the safe distance 'f' and 'L' is high, indicating the robot to turn left. The process of detection and checking is continuous. The other cases of checking are also carried out and its result is satisfying. The IC

and RTL view of the final processor is shown in Fig. 14 and Fig. 15. The components used in final processor and their RTL views are shown in Fig. 16 and Fig. 17.

Fig. 16 is the RTL view of the 46 bit Data frame generator part of the processor. It has major components, like zone detector, polar co-ordinate or position detector, velocity determinator etc. Fig. 17 shows the RTL view of the position detector which generates polar co-ordinates ( $x = \rho \cos \psi$ ;  $y = \rho \sin \psi$ ).

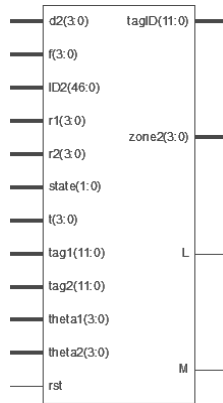


Fig. 14. IC view of the Processor

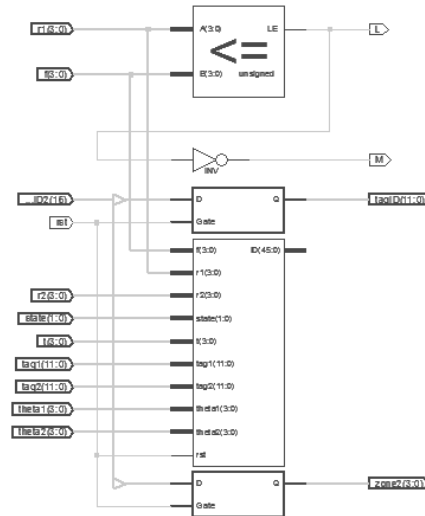


Fig. 15. RTL view of the Processor including part1

For this generator we need two other components,  $\cos \psi$  generator and  $\sin \psi$  generator, which we have achieved using look-up-table (LUT) and the RTL views are shown in Fig. 18 and Fig. 19. Fig. 18 shows the component named as 'velocity2'. This unit generates the displacement vector of the moving robot, as  $||s|| = \rho_1^2 + \rho_2^2 + 2 \rho_1 \rho_2 \cos(\psi)$ . For this purpose we have used another unit named 'velocity' and s1, s2, s3 generator in Fig.22, which generates the signals:  $\rho_1^2$ ,  $\rho_2^2$  and  $2\rho_1\rho_2\cos(\psi)$ . Therefore, we get the displacement of the robot for a predetermined time.

Fig.19 shows the RTL view of the velocity determinator unit named as 'divider1', which divides the displacement 's' by the time 't' and generates the output velocity 'v'.

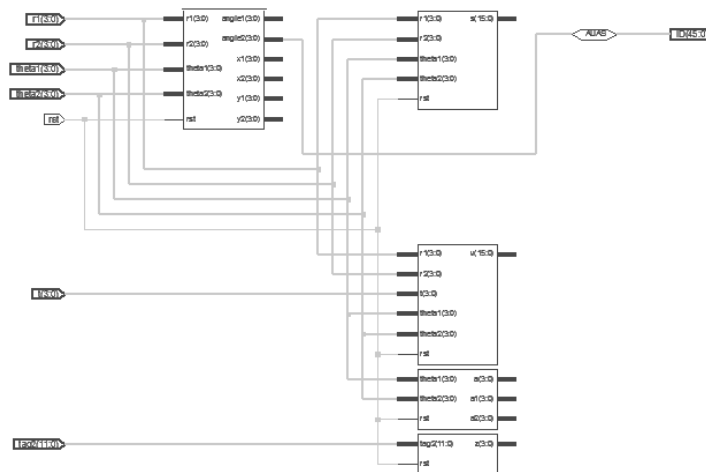


Fig. 16. RTL view of the part1 (data generator part)

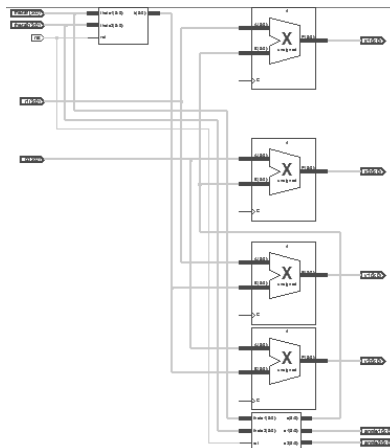


Fig. 17. RTL view of the component 'Position' of the Processor

#### 4.1 SYNTHESIS REPORT FOR THE PROCESSOR [INCLUDING PART1]:

The synthesis report obtained from successful synthesis of processor design module using Xilinx simulation tools, which can be implemented on reconfigurable FPGA kit for real time verification. The advanced synthesis report and device utilization chart are tabulated in Table1

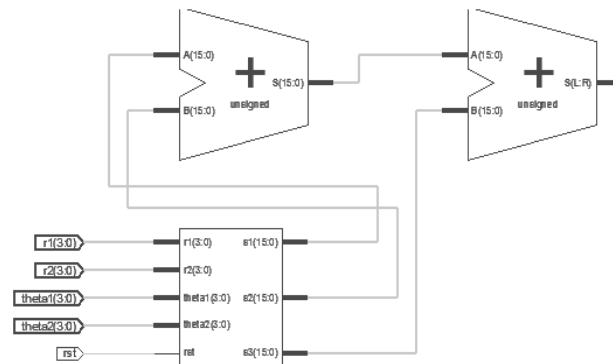


Fig. 18. RTL view of the component 'velocity2'.

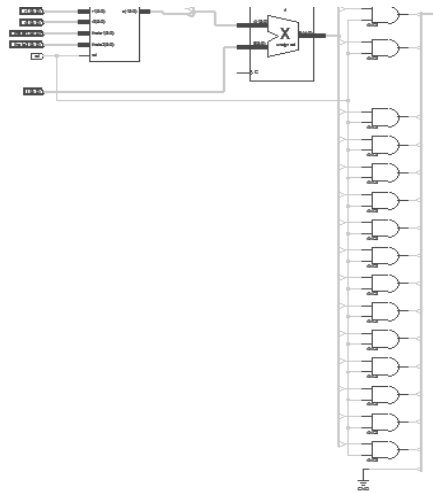


Fig. 19. RTL view of the component 'divider1' of the Processor

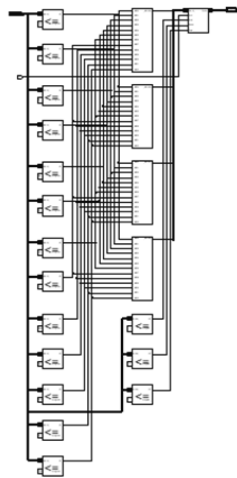


Fig. 20. RTL view of 'cos (ψ).generator'

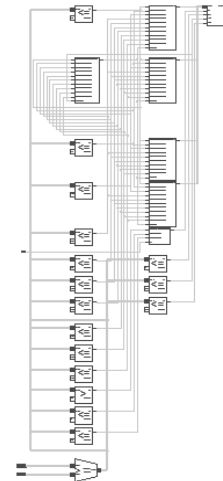


Fig. 21. RTL view of 'sin (ψ).generator'

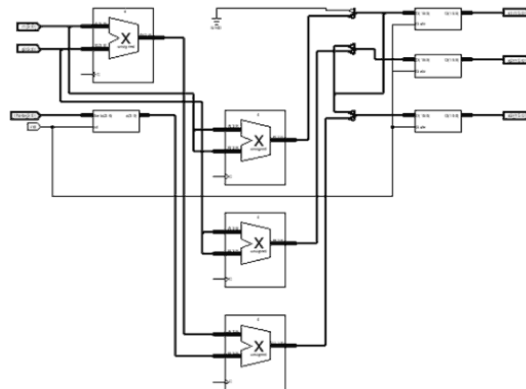


Fig. 22. RTL view of s1, s2, s3 generator  
 $[s1 = \rho_1^2; s2 = \rho_2^2; s3 = 2\rho_1 \rho_2 \cos(\psi)]$ .

TABLE I Synthesis report

Particulars	Number of particulars
16x4-bit multiplier	1
4x4-bit multiplier	10
8x4-bit multiplier	2
16-bit adder	4
4-bit subtractor	5
1-bit latch	16
16-bit latch	6
4-bit latch	14
12-bit comparator less equal	4
4-bit comparator greater	1
4-bit comparator less equal	196
Number of Slice Flip Flops	16
Number of 4 input LUTs	4
Number of bonded IOBs	43
IOB Flip Flops	16
Max. combinational path delay	9.968ns
Total memory usage	164424 kilobytes

## 5. PRACTICAL APPLICATION OF OUR PROPOSED SCHEME

There are manifold applications of this scheme since it helps to create a multiple robot environment which is completely collision free. Due to its flexible nature it finds in numerous applications. A few of the application area are:

**Ware House:** In ware houses, if each of the goods carrier are robots then the proposed system can be adopted to make sure that there are no possibilities of collision between two robots, one robot and boundary, and between a robot and obstacle. Even if there increases the numbers of robots or obstacles, collision can be prevented by slightly modifying the original database. The address of a specific item in the stock can be easily detected by its tag number.

**Super Market:** In super markets we find numerous products which are placed in different locations. The proposed scheme can be efficiently used in transporting goods to it's corresponding stalls in the shortest path without any collision from other robots.

**Institution for Visually Challenged Persons:** Institutions for visually challenged people are also a domain where application of this scheme can create a safe and accident free zone. Smart wheelchairs, having this technology can carry patients to their predefined destinations in a safe and collision free.

Military and Space: Networked robots are frequently required to be deployed in military applications such as battlefield and during disaster rescue operations such as nuclear leakages. In such situations, collisions among robots can have catastrophic consequences.

## 6. CONCLUSION

We have proposed, designed and implemented an anti-collision enabled robot processor incorporating a smart anti-collision algorithm for avoiding collisions with mobile robots as well as with physical objects in multi robot environment. The RTL schematic view of the processor has been achieved from the test bench simulation followed by successful synthesis of the design. RFID technology has been used in the present work to make the system efficient with respect to the power consumption and faster detection of tag IDs to enhance the overall system speed.

## ACKNOWLEDGEMENT

Subir Kumar Sarkar thankfully acknowledges the financial support obtained from UGC vide File No: 36-100/2008(SR).

## REFERENCES

- [1] C.M.Roberts, "Radio frequency identification (RFID)", *Computers and Security* 25(2006) 18--26.
- [2] Konstantinos Domdouzis, Bimal Kumar and Chimay Anumba, Radio-frequency Identification (RFID) applications: A brief introduction, *Advanced Engineering informatics* 21(2007) 350—355.
- [3] L. Peters, M. Pauly, and K. Beck, "Servicebots mobile robots in cooperative environments," in *ERCIM News*, no. 42, July 2000.
- [4] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning: Sensors and techniques," *Robotic Systems*, J, vol. 14, no. 4, pp. 231–249, April 1997.
- [5] G.N.Desouza and A.C.Kak "Vision for Mobile robot navigation: A survey", *IEEE trans.Pattern Analysis and Machine Intelligence*, vol.24, no.2, pp-237-267, February 2002.
- [6] P.Rusu, E.M.Petriu, T.E.Whalen, A.Cornel and H J W Spoilder, "Behavior-based neuro fuzzy controller for mobile robot navigation", *IEEE Trans. Instrumentation and Measurement*, vol 52, no.4, pp-1335-1340, August-2003.
- [7] Hallmann and B. Siemiatkowska, "Artificial landmark navigation system," in *Int. Sym.. Intelligent Robotic Systems*, July 2001.
- [8] L.Ojeda, D. Cruz, G. Reina, and J. Borenstein, "Current-based slippage detection and odometry correction for mobile robots and planetary rovers," *IEEE Trans. Robotics*, vol. 22, no. 2, pp. 366–378, April 2006.
- [9] Bing Jiang, Kenneth P. Fishkin, Sumit Roy and Matthai Philipose, "Unobtrusive Long-Range Detection of Passive RFID Tag Motion", *IEEE Trans. Instrumentation and Measurement*, Vol. 55, No. 1, February 2006.
- [10] Myungsik Kim and Nak Young Chong, "Direction Sensing RFID Reader for Mobile Robot Navigation", *IEEE Trans. Automation science and Engineering*, vol. 6, no. 1, January 2009.
- [11] Sunhong Park and Shuji Hashimoto, "Autonomous mobile robot navigation using passive RFID in indoor environment", *IEEE Trans. Industrial Electronics*, vol. 56, no. 7, July 2009, pp 2366-2373.
- [12] Todd M. Ruff and Drew Hession-Kunz, "Application of Radio-Frequency Identification Systems to Collision Avoidance in Metal/Nonmetal Mines", *IEEE Trans. Industry Applications*, vol.-37, no.-1, January/February 2001.
- [13] Toshihiro Hori, Tomotaka Wada, Yuuki Ota, Norie Uchitomi, Kouichi Mitsuura, Hiromi Okada, "A Multi-Sensing-Range Method for Position Estimation of Passive RFID Tags", *IEEE Int. Conf. Wireless and Mobile Computing Networking and Communications* (2008), pp. 208-213.
- [14] Hyung Soo Lim, Byoung Suk Choi and Jang Myung Lee, "An Efficient Localization Algorithm for Mobile Robots based on RFID System", *Int. Jt. Conf. SICE-ICASE*, Oct. 18-21, 2006, Bexco, Busan, Korea. pp:5945-5950.
- [15] Md. Suruz Miah and Wail Gueaieb, "RFID-Based Robot Navigation System with a Customized RFID Tag Architecture", *IEEE Trans. Automation Science and Engineering*, Jan. 2009, pp.44-54.

- [16] Yu Song-sen, Zhan Yi-ju, Wang Yong-hua, "RFID Anti-collision algorithm Based on Bi-directional Binary Exponential Index", IEEE Int. Conf. Automation and Logistics, 2007, 18-21 Aug. 2007, pp. 2917 – 2921.
- [17] Narek Pezeshkian, Hoa G. Nguyen, Aaron Burmeister, "Unmanned ground vehicle radio relay deployment system for non-line-of-sight operations", Proc. 13th IASTED Int. Conf. Robotics and Applications.2007,pp. 501-506.
- [18] J.Bhasker: A VHDL synthesis Primer, BS Publication
- [19] Wayne Wolf :Modern VLSI Design; 4th edition; PHI Learning Private Limited
- [20] Stephen Brown and Zvonko Vranesic: Digital Logic design; Tata McGraw Hill Publication.
- [21] S. M. A. Motakabber, Mohd Alauddin Mohd Ali, Nowshad Amin:"VLSI Design of an Anti-Collision Protocol for RFID Tags": European J.Scientific Research ISSN 1450-216X Vol.28 No.4 (2009), pp.559-565 .
- [22] www.vhdl.org
- [23] www.edaboard.com
- [24] Liu Jing and Po Yang , "A Localization Algorithm for Mobile Robots in RFID System" IEEE Int. Conf. Wireless Communications Networking and Mobile Computing (2007),pp. 2109-2111.
- [25] Minh Jo, Chang-Gyoon Lim, Emory W. Zimmers, "RFID tag detection on a water content using a back-propagation learning machine", KSII Transactions on Internet and Information Systems / Dec, 2007.
- [26] Minh Jo, Hee Yong Youn, Si-Ho Cha and Hyunseung Choo, "Mobile RFID Tag Detection Influence Factors and Prediction of Tag Detectability", IEEE SENSORS JOURNAL, VOL. 9, NO. 2, FEBRUARY 2009,pp.112-119.

#### AUTHORS

**Joyashree Bag** born in Kolkata, West Bengal, received MTech Degree in VLSI Design and Microelectronics Engineering from Jadavpur University. She is with the Department of Electronics and Communication Engineering, Meghnad Saha Institute of Technology, Kolkata, India, currently pursuing her research leading to PhD at Jadavpur University as a Research fellow. Her research field is VLSI Design and development of RFID technology based systems/ASICs and hardware realization upto RTL schematic level using VHDL code and system level Implementation using FPGA.



**Rajanna .K .M** received BE and ME Degrees in Electronics and Communication and Digital communication in 1992 and 2000. He is with Department of Telecommunication Engineering, Dr. Ambedkar Institute of Technology, Bangalore, India, since 2003, currently pursuing his research leading to PhD on deputation at the Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India. He is a senior member of IEEE. His research areas include Embedded System Design, RFID, Low power system realization using RFID technology and its simulations.



**Dr. Subir Kumar Sarkar** is with the Department of Electronics and Telecommunication Engineering as Professor and Head of the Department. He has numerous research papers in archival journals and peer reviewed conferences. Twenty six students have been awarded PhD degree under his guidance so far. As a Principal Investigator he has successfully completed six sponsored R&D projects. He is a senior member of IEEE. His research area includes RFID Applications and system realization, Ad hoc and Sensor networks, Digital water marking and data hiding, Electron device simulation and Low power VLSI circuits.

