

Original Research Article

Hardware Efficient Scheme for Indoor Environment Using Grid Mapping

ABSTRACT: Robotic Navigation is a key aspect in mobile robotics. This paper presents about the grid mapping algorithm for navigation of a robot which is controlled by an FPGA. The environment is divided into the grids where Landmarks are considered as grid points in the environment. The proposed path planning is also developed with obstacle avoidance mechanism to overcome the obstacles in robust environment. The robot navigation efficacy improves with the landmarks and hardware scheme. The hardware scheme is developed with an NI lab view and implemented on Spartan 3e FPGA. The FPGA's are best in parallel processing comparatively with other control devices.

Aim: Navigation of FPGA based Autonomous robot efficiently towards the target using grid mapping technique.

Study design: The study of path planning methods. Among that efficient path planning is proposed with landmark by using grid mapping. Developed the hardware scheme for proposed algorithm.

Place and Duration of Study: Padamasri Dr B V Raju Institute of Technology, Narasapur, Medak (Dist), Telangana, 502313. Duration of study in 2014-15.

Methodology: Hardware scheme for autonomous robot navigation.

Results: Simulation results of proposed algorithm and experimental results with snapshots

Conclusion: Navigation of Autonomous fpga based robot with a hardware scheme by parallel processing and using grid mapping methods.

Keywords— Path Planning, Hardware, Landmark, Obstacle avoidance, FPGA.

1. INTRODUCTION

Robotics is one of the most important fields in automation sciences, where assistance is improved in different areas of mankind like health, agriculture, transportation, service industries and so on. The mobile robots play vital role in automation and also in providing services. The researchers have proposed different navigation methods or algorithms for the mobile robots in both indoor and outdoor environments for last three decades. The importance of navigation with minimal sensing has increased in present scenarios. The mostly known path planning algorithms are as like D*, A* algorithms are used for navigation of robots.

Many algorithms have come into existence for computing the shortest path which has some merits and demerits. Palatino's graph growth algorithm implemented with two queues (TWO-

22 Q) algorithm which is useful for path routing on road networks. This algorithm spends a lot of
23 time on useless work when the shortest path is relatively short. An improved version of
24 TWO-Q is Mild-TWO-Q algorithm which reduces the time to reach the shortest path [1].

25 Though conventional quad tree-based path planning approach is simple, robust and efficient
26 has some limitations. Dijkstra's shortest path searching algorithm, a fast path planning
27 overcomes the limitations of the quad tree by optimization of a path graph for both efficiency
28 and accuracy is proposed in [2].

29 In real time applications, path planning in known or unknown environments is a difficult job.
30 The D * algorithm is implemented for path planning in partially known environments with the
31 help of sensor information. For effective path planning, improved version of D* algorithm
32 based grids is studied in [3]. The path planning is also incorporated with the voronoio (GVD)
33 [18] and visibility graphs [7] & [14]. Thomas M Howard et al [14] was mentioned the
34 developments of robot motion and planning in all terrains by preferring the graphical
35 methods.

36 RFID-assisted indoor localization and the impact of interference on its performance [4].
37 Performance-based evaluation of RFID-based indoor location sensing solutions for the built
38 environment is studied in [5]. Improving location awareness in indoor spaces using RFID
39 technology [6]. Das, A et al [17] rely on the mapping to explore the unknown environment.

40 To find out the appropriate shortest paths in a convex polygonal environment, by reducing
41 the visibility graph a new parallel algorithm is developed and its FPGA implementation is
42 studied in [7]. Several practical applications that require high-speed shortest-path
43 computations are developed by field programmable gate array (FPGA) -based accelerator to
44 achieve high performance at low cost. Bellman-Ford algorithm adapted to solving the single-
45 source shortest-path problem in a fast and efficient manner [8]. The proposed algorithm is
46 carried out on FPGA (Spartan 3E) due to its high features like less power consumption and
47 parallel processing nature [9]. The authors [11] are mentioned the shortest path planning
48 algorithm upto the simulation results. In similar lines Abdelmoula C, et al [12] are explained
49 the importance of FPGA's in shortest path planning using Altera's devices.

50 Navigation of mobile robots also developed by using the fuzzy and Neuro controlling
51 methods. Algabri et al [10] was described about the mobile robot navigation in an unknown
52 environment with fuzzy logic approaches. Authors [10] are concluded as genetic fuzzy is
53 best for travelling time and average speed. PSO-fuzzy and Neuron -fuzzy are efficient in
54 distance travelled. Baturone, I et al [13] et al proposed about the robot navigation with
55 Neuro-fuzzy using FPGA embedded controller, but it is limited to obstacle avoidance
56 precision.

57 Obstacle avoidance mechanism is another key aspect in the navigation of mobile robots. In
58 the early 1990's Lumelsky et al are mentioned about the bug algorithms [15]. . All these
59 individual mobile robot algorithms can develop in PSO (practical swarm optimization).
60 Supakar ,N et al [16] has been developed the Obstacle avoidance for PSO's.

61

62 Presently, services for humans are provided by the robots, but all these robots prefers the
63 path planning based on the particular identity like track. Track type of path planning has its
64 own limitations like less self intelligence which fails to extend their services in all directions
65 of the indoor environment. The Landmark determination method overcomes the track issues
66 by proposing algorithm in service robots, by estimating its current position and improves self

67 intelligence. When the robot is assigned to a desired task, it establishes a relevant path from
68 the source to the destination with landmarks in the mapping environment. Path planning
69 without track can increase the abilities of robot with minimum resources and fulfils the task in
70 short span of time. The proposed path planning is also developed with obstacle avoidance
71 mechanism to overcome the obstacles in robust environment.

72 1.1 Background

73 Dutch computer scientist Edsger Dijkstra proposed Dijkstra's algorithm in 1956 and
74 published in 1959.[20-21] it is a graph search algorithm that solves the single source
75 shortest path tree. This algorithm is often used in robot path planning to accomplish the task.
76 The simple mathematical modelling preferred, weighted graph $G = \{E, V\}$ and source vertex
77 $v \in V$, such that all edge weights are nonnegative. The findings of Dijkstra's algorithm
78 shortest paths (or the shortest paths themselves) from a given source vertex $v \in V$ to all other
79 vertices. This algorithm hardware scheme proposed by the authors [6], but not mentioned
80 real time node point with any device or component. In this context, this research work
81 developed a new VLSI-efficient scheme with landmarks for real time scenarios.

82 This research work summarizes in different sections like Section 1 introduces the
83 introduction of robotics path planning and also the importance of landmark determination.
84 Section 2 presents the proposed algorithm with necessary environments and environments.
85 Section 3 gives VLSI architectures for proposed algorithm with the explanation of individual
86 blocks. Section 4 deals with experimental setup and results. Section 5 concludes with
87 conclusions and future scope.

88

89

90 2. PROPOSED PATH PLANNING ALGORITHM

91

92 The interest of authors is to develop assistive methods to reduce the human efforts by using
93 robots in service oriented indoor environments. The main theme of this chapter is to develop
94 an autonomous FPGA based robot of self intelligence which can assist in the service
95 environment to perform various tasks by sensing its location with the help of Landmark
96 (RFID) tags. The environment is mapped initially with defined parameters like distances
97 between one location to another location. When a robot receives a task by the external
98 world, it assesses all the possible probabilities for different paths to reach indeed
99 destination. Among all probabilities it looks attentively at the low weight, distance path as
100 the shortest route by comparing the distances from present location to the desired task
101 destination using FPGA hardware.

102 The obstacle detection and obstacle avoidance is one of the major issues to reach the
103 destination successfully. It can easily detect all the obstacles through ultrasonic sensors and
104 avoids by obstacle mechanism which is developed in FPGA control unit. The mobile robot
105 equipped with eight ultrasonic sensors and RFID reader is configured to the robot such that
106 it can be capable of estimating particular location and move towards the destination. With
107 the help of these devices (ultrasonic sensor and RFID) which are of being low cost, the robot
108 can easily accomplish any kind of navigation task.

109 An RFID system is a pair of RFID Reader and RFID Tag. A radio frequency identification
110 reader (RFID reader) is a device used to collect information from an RFID tag. All RFID tags
111 are unlike to each other and contains the unique ID about a particular location

112 RFID system works under the phenomenon of electromagnetic. RFID reader has an antenna
113 and creates a electromagnetic field in its surrounding areas. When the robot moves close to

the RFID tag, the tag gets activated by the emission of electromagnetic field emitted by an RFID reader. The RFID reader reads the tag's unique value which is under the electromagnetic field and stores in stack memory. It gathers all the tag data sequentially in the given environment and stores in a stack memory. After receiving all the available tags it evaluates its current position and take appropriate action to reach the next step. In this manner, it moves step by step based on the collected data.

Ultrasonic sensors are used to detect the obstacles and avoid the obstacles by easy method of distance measurement (PWM). The sensor continually transmits an ultrasonic wave. When the ultrasonic wave hits the obstacle, it gets reflected to the sensor as echo signal. By measuring the echo pulse width, the distance of the obstacle can be easily calculated. By knowing the distance value the robot changes its direction with the help of stepper motors into the path where there are no obstacles. Finally, it traverses the path towards the destination.

2.1 INDOOR ENVIRONMENT

The indoor environment represents in Fig1, it consists of six different Halls like Hall A, Hall B and so on Hall F. Respective RFID tags are placed at each hall. The current position of the robot is identified by using RFID tags. One of the experimental exemplar is followed by considering the desired task as Hall 'F', when the robot is localized at Hall 'A'.

The weights of the path distance of different path probabilities between starting Hall 'A' to destination Hall 'F' are evaluated. The comparison circuit between path distances concludes the shortest path. The same path is defined as the trajectory of the robot, in this example shortest path is A-D-E-F. All these computations are done in control unit of the FPGA.

Robot continually checks for RFID tag values, then moves towards the destination with respect to the path planning algorithm. The each Hall is having subset Landmarks for example Hall A by A1, A2, A3 in the localization of the robot. This subset values is used for accurate path planning by the correction and prediction module.

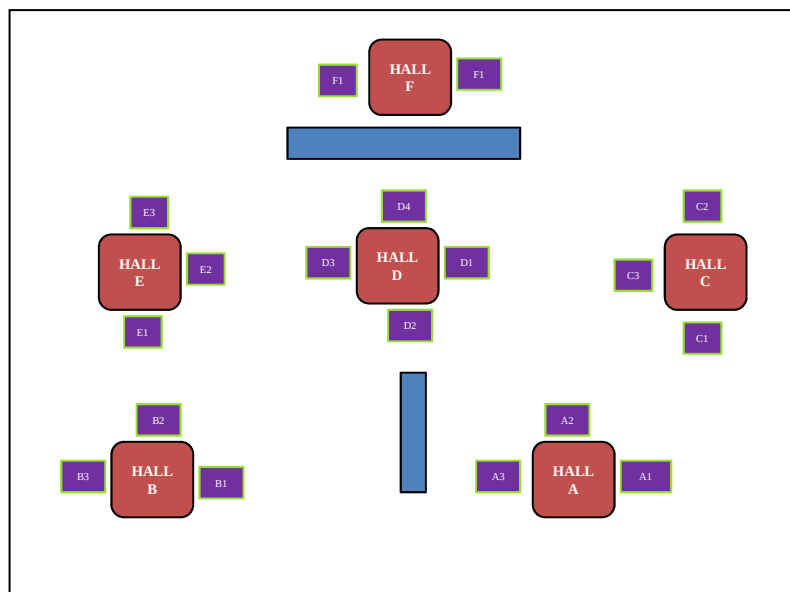


Fig 1: Indoor environment

2.2 PSEUDO CODE FOR THE PROPOSED ALGORITHM

Algorithm Steps:

STEP 1: The Localization of the robot is performed by capturing the landmark by RFID system.

STEP 2: The next level of algorithm is assigned by desired task. The Probability Path Analyzer module gives the shortest path between source to destination node as shown in Fig 2.

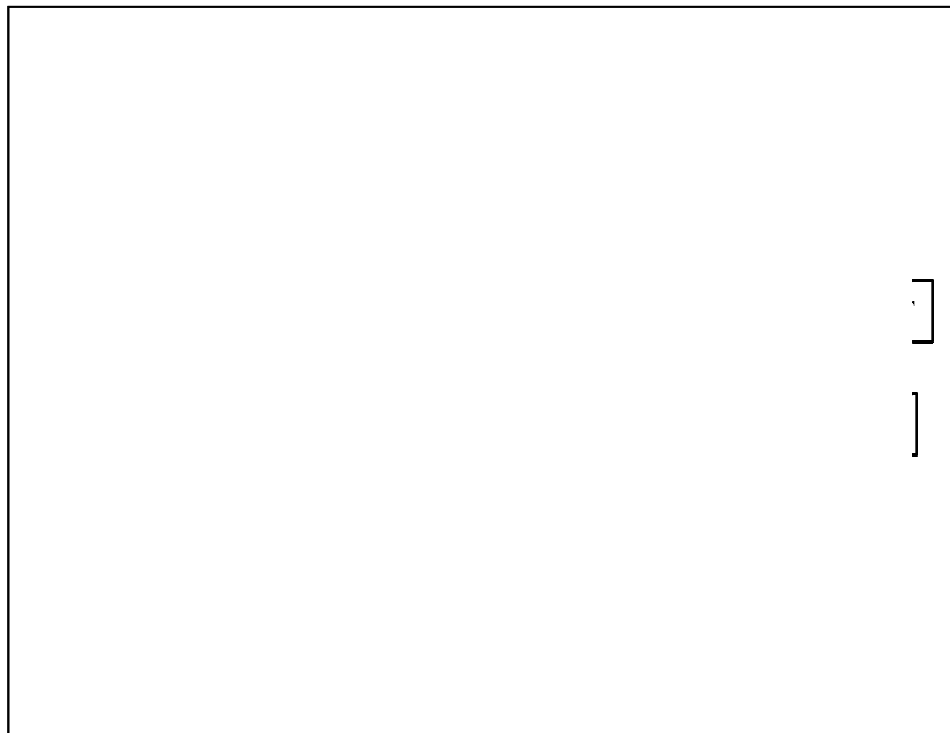
STEP 3: The robot starts at source point and traverse towards the destination point by using the RFID landmark. Each Hall has few subset landmarks, it provides perfect direction to traverse towards indeed destination.

STEP 4: In trajectory of the robot, it also visits the other Halls to reach the final destination node. In this process the correct module internally having computation method which always checks the visited Hall and gives the next Hall of the trajectory to achieve proper path planning. The path traversing has shown with flow chart in Fig 3.

STEP 5: when the robot reaches for final Hall, it compares with destination Hall and concludes as a goal.

Remark 1: After evaluation of different path probabilities at step 2, one of the examples Hall 'A' considered as start and Hall 'F' as the destination. The shortest trajectory of the robot between A to F presents as A- D -E – F path as shown in Fig 2.

Remark 2: According to the example the robot start from 'A' and feed forward up to 'D' landmark. If any subset of 'D' landmark is achieved, it considered as D and corrects the next direction of the path as shown in Fig 3. This action performed by correction and verification module. It repeats similar method for all Halls.



213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265

Fig 2: Flowchart for identification of shortest path

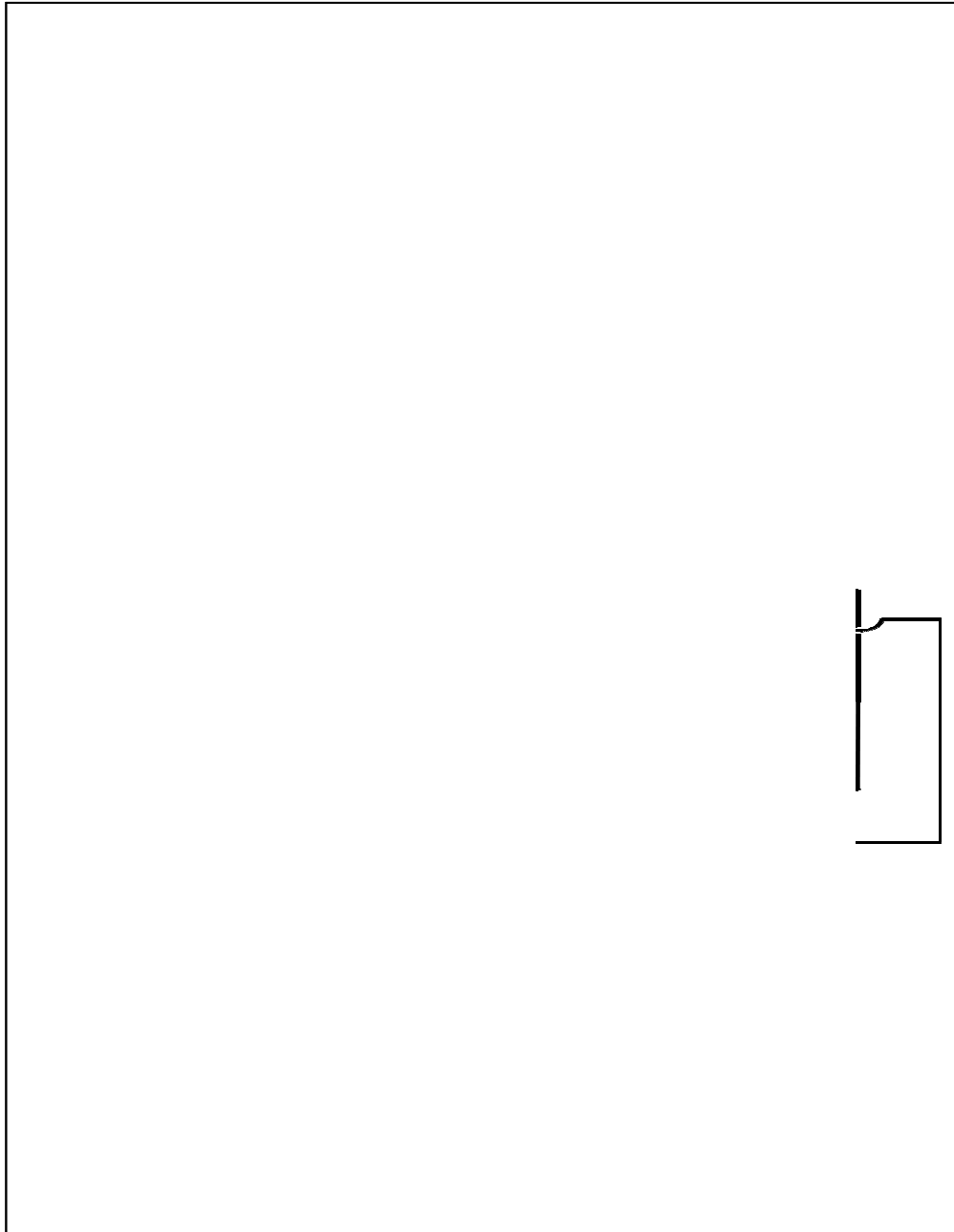


Fig 3: ASM Chart for Path Navigation

3. VLSI ARCHITECTURE OF PROPOSED ALGORITHM

The Proposed algorithm architecture is deployed on FPGA and navigated successfully. The sensor modules are used to estimate the distance with pulse width modulation technique. Minimum distance finder exhibits output for each sensor. The angles are known in path planning by digital compass module with UART protocol. The CORDIC IP CORE module has been used for trigonometric computations. The soft core odometer module is a special technique developed for computing odometer distance without external devices. The CU also bridges the communication between the FPGA and external devices like sensor and compass. It performs parallel processing in sensing the information, odometer and angle computations and, other than these, the motors are driven by motor logic control, the actions required are synchronized with device clock frequency.

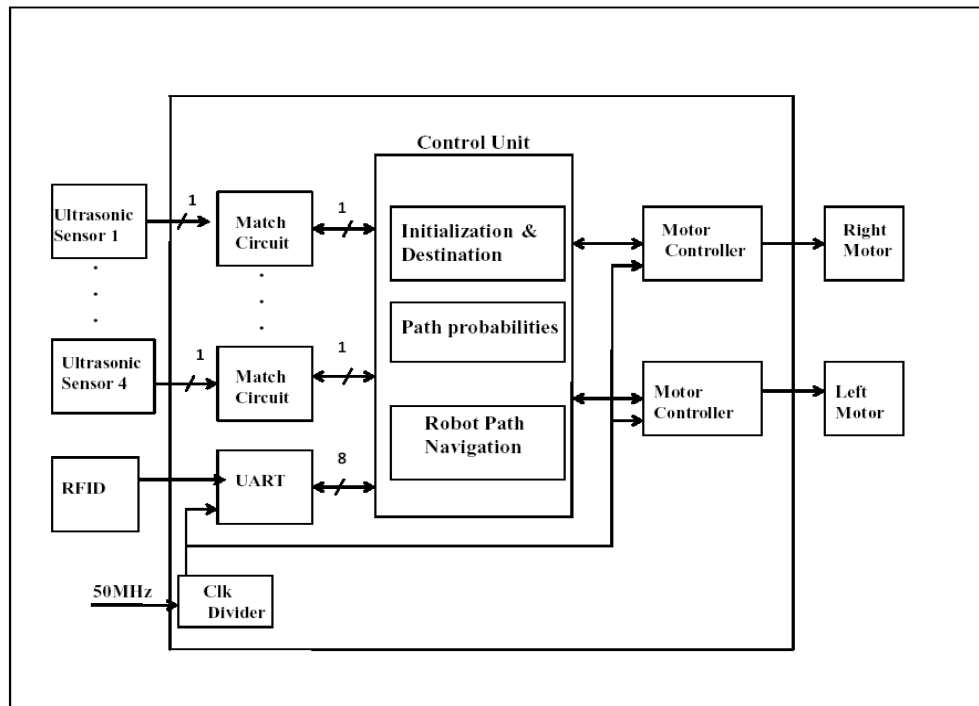


Fig 4: Overall VLSI Architecture of Proposed algorithm in indoor environments.

3.1 FPGA (Control Unit):

Here the control unit is Xilinx board XC3S2000 (FPGA Spartan 3E). In this research FPGAs are preferred due to its parallel processing nature. The control unit controls all the devices and navigates the robot to reach the destination. The architecture is developed by the behavioural modelling. The Xilinx CORDIC soft core is incorporated for angle evaluation and

317 computations. Control unit was integrated with the following modules internally like 1)
318 Initialization and Destination Module 2) Path Probabilites Module 3) Robot Navigation
319 Module.

320

321 3.1.1 INITIALIZATION AND DESTINATION MODULE:

322

323 This module comprises of very first job whenever a task is given to the robot. This module
324 particularly helps to know its initial position, with the help of RFIDs. Initially the robot gets a
325 two tag values which indicate a source and destination. Among them it considers first Tag
326 value as source point and second as destination point. All the tag values which are under
327 given environment are stored in a stack memory. When robot receives two tag values it is
328 compared to the tag values which are already in the stored in the memory and comes to the
329 conclusion with a particular Halls as a source and destination.
330

331 3.1.2 PATH PROBABILITIES MODULE:

332

333 This module is followed by initialization and the destination. By receiving the source and
334 destination, it checks for all probabilities and chooses a feasible path which is of low weight
335 age using comparator circuit.
336

337 3.1.3 ROBOT PATH NAVIGATION:

338

339 This module comprises of correction circuit. From the above module it gets the shortest path
340 to navigate towards the destination. Correction circuit helps the robot to choose appropriate
341 subsets by taking required angle and enters into next Hall.

342 3.2 MATCHING CIRCUIT:

343

344 The matching circuit comprises of the comparator and FIFO. The signals from the ultrasonic
345 sensors are converted into pulses. With these pulses the distance of an object can be
346 obtained. Now, the obtained distance value is compared with matching circuit whether to
347 focus the obstacle or to discard the obstacle. Similarly, all the RFID Tag values are stored in
348 the FIFO and gives the appropriate signals to the control unit to perform the required action.

349 3.3 UART MODULE

350

351 The UART is a serial communication protocol used to interface RFID to the robot. The 12
352 byte data on RFID Tags are transmitted to control unit through the UART protocol.

353 3.4 MOTOR CONTROLLER MODULE:

354

355 Depending on RFID & Ultrasonic inputs, the control unit produces specified signals to motor
356 controller module. Which helps the robot to move in all required directions like left action,
357 right action etc.

4. EXPERIMENTAL SETUP & RESULTS

4.1 Experimental Setup:

The fig 5 shows the experimental setup or architecture of prototype to perform the given task. The heart of this setup is an FPGA (SPARTAN 3E), it fetches signals from the ultrasonic sensors and RFIDs for navigation of robot. Control unit (FPGA) drives the stepper motors through the driving circuits. The Power supply equipped with 12v and 3000mah. The voltage regulator is on-board available to power up the sensors.

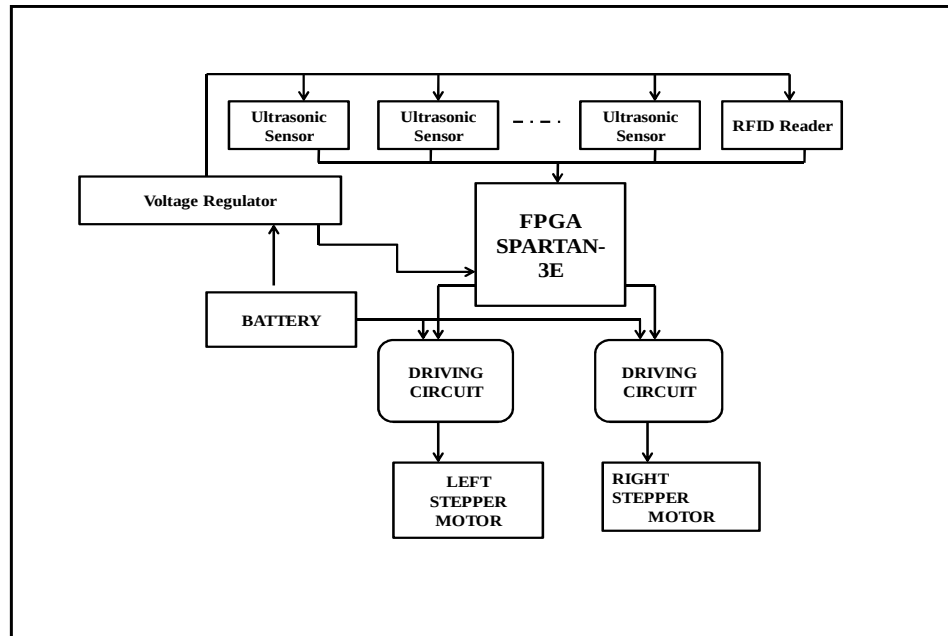


Fig 5: Block Diagram of Robot with FPGA

4.2 SIMULATION RESULTS:

394

395

4.2.1 Initialization module

396



Fig 6: Simulation results of initialization module

Table 1: Parent and the child node's RFID values

Source Location (Parent, Node)	A – 001	B - 010	C - 011	D - 100	E – 101	F - 110
RFID TAGS (Sub nodes)	A1 -51 A2 -52 A3 -53	B1 -54 B2 -55 B3 -56	C1 -57 C2 -58 C3 -59	D1 -60 D2 -61 D3 -62 D4 - 63	E1 -64 E2 -65 E3 -66	F1 -67 F2 -68

The above simulation results illustrates the initial position of a robot. In the given indoor environment in Fig:6 there are seven different halls like HALL _A to HALL _F with their respective sub nodes. In Fig: 6 A, B, C, D, E and F are parent nodes and the notation for these halls are given in binary as 001,010,011,100,101,110 respectively. A1, A2, A3, B1, B2.....F1, F2 are sub nodes having different RFID Tag values.

When RFID Reader reads the sub node values, Robot concludes its corresponding parent node as initial position. For e.g., if Robot reads any one of 51 or 52 or 53 it identifies A-001 as its initial position. In the above simulation robot reads the 54 tag values in between 200ms and 400ms and concludes B as initial position.

4.2.2 DESTINATION MODULE:



Fig 7: Simulation results of destination module

In the above waveform, robot evaluates the destination from external source. Here Hall A, Hall B..... Hall F is the different Halls in a given environment, having different number of RFID tags. When an RFID reader reads the data from the RFID tags the control unit checks the data is from which node and gives the appropriate Hall as a destination.

In the above simulation, robot gets destination node from Zigbee module. Here, at 0-200ms the Zigbee data is 51. Since 51 is a sub node of A and A is treated as destination point. Similarly, at 600ms the Zigbee data is 68. Since 68 is a sub node of F and F is treated as destination point.

4.2.3 PATH PROBABILITIES:

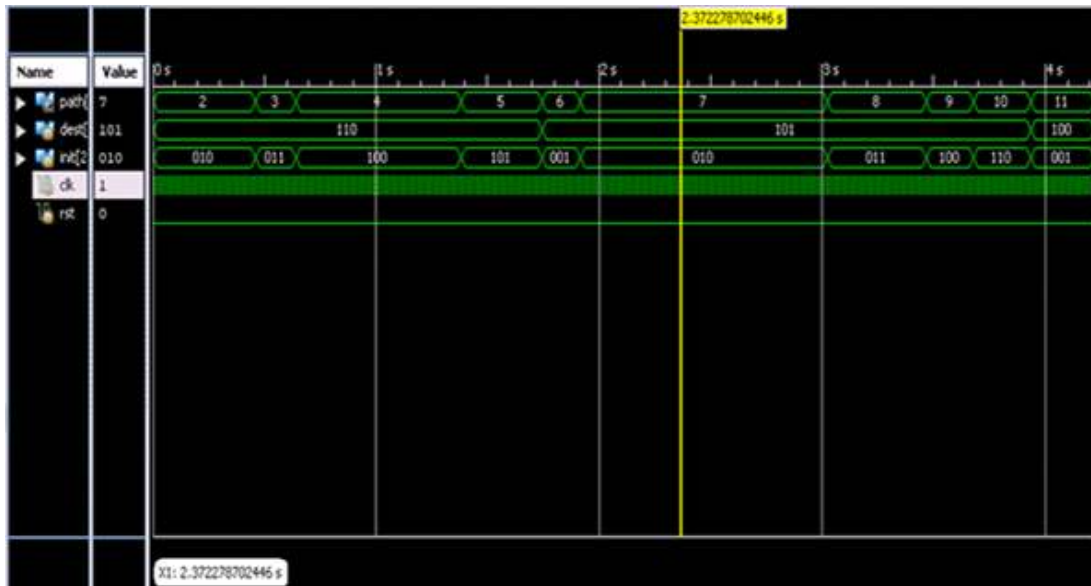


Fig 8: Simulation results of Path Probability module

In the above waveform, the robot checks all the path probabilities for giving initial and destination locations. Among all the path probabilities, robot compares for the shortest path and navigates through that path.

Table 2: Probabilities of Shortest Path Planning

Initial and Destination	A – 001 & F - 110	B – 010 & F - 110	C – 011 & F - 110	D – 100 & F - 110
Path	A-D-E-F	B-D-E-F	C-D-E-F	D-E-F

In the above simulation, initial and destination nodes are given by RFID and Zigbee modules. Here when the robot gets an input as (A and F), (B and F).... so on. For every input there will be a number of different paths having different weight ages. Among those paths robots uses the comparison module between all paths and gets the shortest path as a desired path. Here the above simulation gives required paths for different inputs.

461 4.2.4 ROBOT PATH NAVIGATION

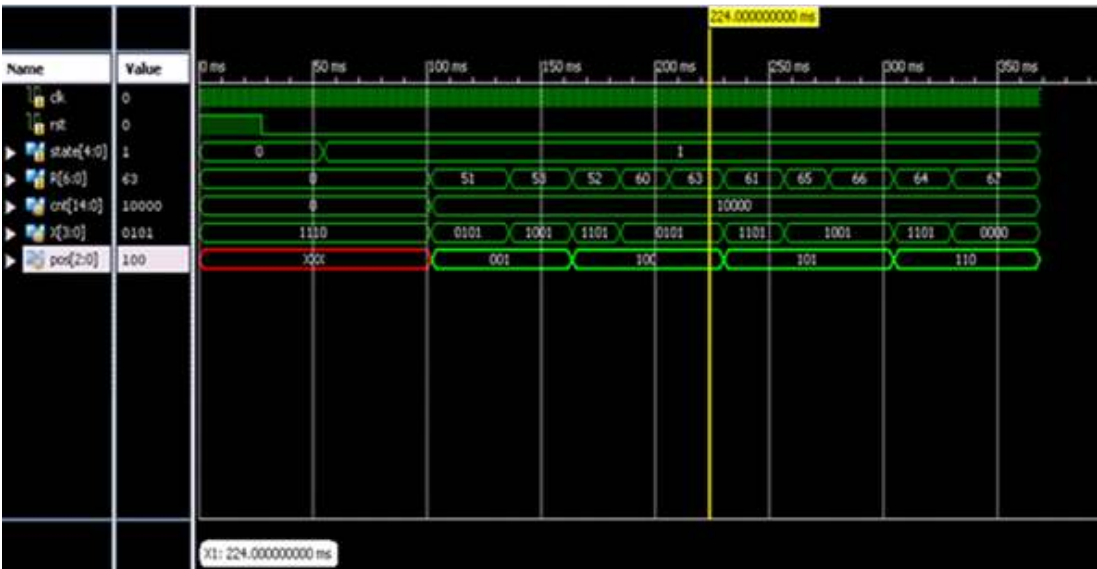


Fig 9: Simulation results of Navigation module

The above simulation gives the Robot navigation after getting initial and destination nodes. This simulation shows how the robot navigates towards destination for the input (A and F). From the path probability simulation robot gets desired path as A-D-E-F. Here at the time period 100ms when it reads 51tag it enters into the A (001) node. At 200ms it reads the 60th tag and enters into D node. Similarly, it enters into E and F when it reads them sub nodes respectively.

472 4.2.5 OBSTACLE AVOIDANCE:

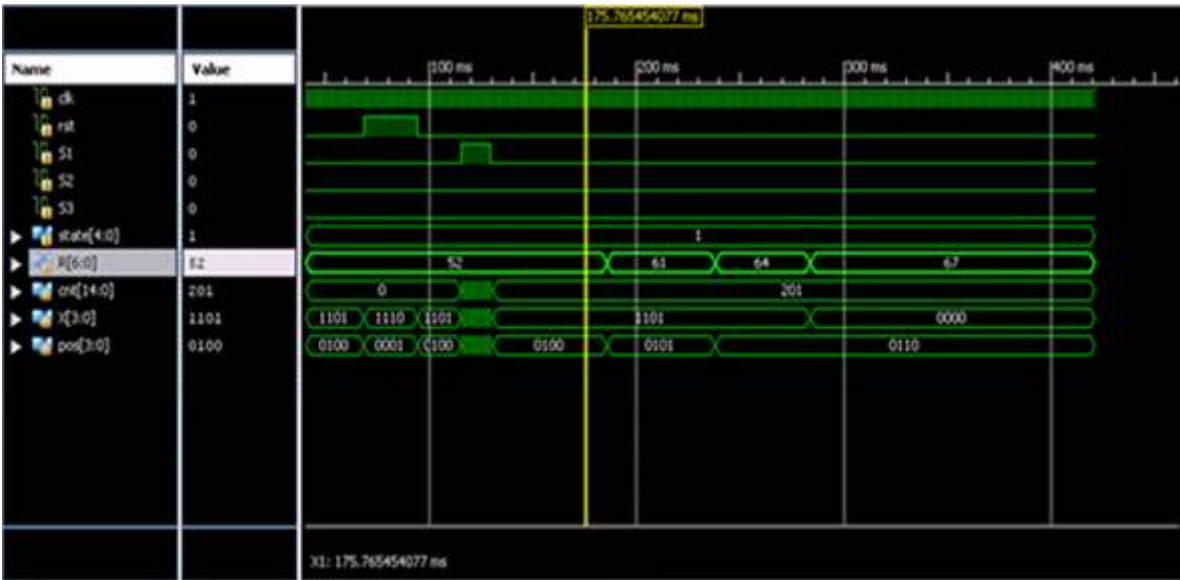


Fig 10: Simulation results of obstacle module

477
478
479
480
481
482
483
484
485
486
487
488
489

490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509

In the above waveform, robot overcomes the obstacle and continues travelling on the desired shortest path. Here robot gets an obstacle between the subsets A&D, It avoids the obstacle and moves in through D, E & F nodes. This simulation shows how the robot avoids the obstacle in its desired path. In the fig. The robot gets an obstacle between the nodes A and D. Here at the time period 120ms (S1=1 i.e. it indicates obstcale) and performs obstacle avoidance and enters into D node at 170ms.

4.3 SYNTHESIS REPORT:

TABLE 3:Area consumption on Xilinx FPGA board Spartan XC3S2000

Block	No.of Slices	No.of 4 i/p LUTS
Matching Circuit	84	176
Control Unit	652	1304
US Sensor Module	67	114
UART module	293	584
Motor Logic Control	86	166
Obstacle avoidance module	184	264
Total	1366	2608 (5.6%)

Fig 11: Analysis of Area consumption on FPGA board

Fig 11 represents area consumption, it relied on the table 3 Xilinx FPGA board Spartan XC3S2000. The basic units of the area have depended on CLB's occupied with LUT's.

4.4 EXPERIMENTAL RESULTS:

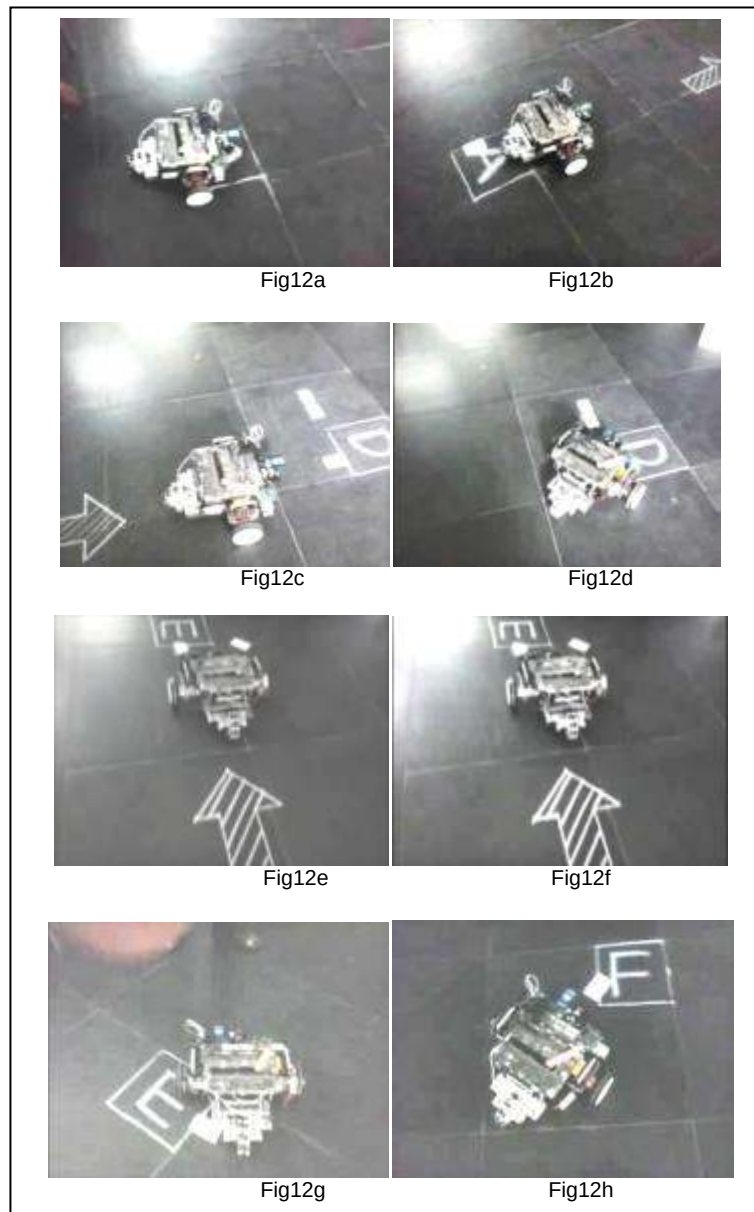


Fig12a-12h: Snapshots of Proposed algorithm's experiment results

Fig 12a- h are represented proposed hardware based robot navigation with grid mapping as mentioned in the section 2. The Fig 12a is localization of the robot and remaining snapshots represents the navigation of the robot from the A grid point to the F grid point (intended destination).

4.5 Comparison results:

561

562 The other robot path planning algorithms like D*, A* is not perfectly fitted towards
563 service environment. Hardware based D* algorithm shortest path planning [5] is limited to
564 the navigation computations. The novelty of proposed algorithm is perfectly drive the robot
565 path by landmark determination and comparatively [6] it best up to our knowledge. The
566 proposed algorithm mainly depended on landmarks (visited nodes). Precision navigation can
567 be done by using the Landmark or grid mapping.

568

569

570 TABLE 4:Area consumption on Xilinx FPGA board Spartan XC3S2000 by visiting node

571

RFID's in the Environment (nodes)	BRAM
16 visited nodes	18 (45 %)
24 visited nodes	27 (67.5%)

572

573 In comparison with [6] (for 64 nodes occupied the 124 BRAM), the proposed algorithm
574 occupies 72 BRAM (for 64 nodes) in an FPGA device. The other papers [13] used Fuzzy
575 controller and Micro blaze it occupies more device space. The efficiency is improved in
576 driving the path by the hardware scheme (FPGA), it made parallel processing with low
577 power consumption [19]. We preferred the Lower FPGA device (Spartan 3e 2000) than
578 vertex II [13].

579

580 5. CONCLUSIONS

581

582 This Paper presents path planning in service robot in a robust environment. In Taiwan
583 and Japan the restaurants are preferred to use the track system navigation to deliver the
584 food. We developed the hardware efficient algorithm, it can be incorporated in industrial and
585 service environments (hotel). In this research work, the usage of landmarks increased the
586 abilities of a robot in knowing its current position and navigate to the intended goal point. Its
587 precision is improved in navigation and device utilization than [6] & [13] .Its performance is
588 validated by experiments in the defined environment have been done successfully in the
589 laboratory. The FPGA energy consumption is only 9.64 Joules during obstacle avoidance
590 and path planning for this experiment.

591 REFERENCES

- 592 [1]. JinFu Leng; Wen Zeng, "An Improved Shortest Path Algorithm for Computing One-to-
593 One Shortest Paths on Road Networks," *Information Science and Engineering (ICISE), 2009*
594 *1st International Conference on*, vol., no., pp. 1979,1982, 26-28 Dec. 2009.
- 595 [2]. Joe Young Hwang; Jun Song Kim; Sang Seok Lim; Kyu-Ho Park, "A fast path planning
596 by path graph optimization," *Systems, Man and Cybernetics, Part A: Systems and Humans*,
597 *IEEE Transactions on*, Vol. 33, no. 1, pp. 121,129, Jan. 2003.
- 598 [3]. Jianming Guo; Liang Liu; Qing Liu; Yongyu Qu, "An Improvement of D* Algorithm for
599 Mobile Robot Path Planning in Partial Unknown Environment," *Intelligent Computation*
600 *Technology and Automation, 2009. ICICTA '09. Second International Conference on*, Vol. 3,
601 no., pp. 394,397, 10-11 Oct. 2009 doi: 10.1109/ICICTA.2009.561.

602 [4] Chakravarthy, N.; Jizhong Xiao, "FPGA-based Control System for Miniature Robots,"
603 2006 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol., no., pp.
604 3399, 3404, 9-15 Oct. 2006.

605 [5] Sridharan, K.; Priya, T.K.; Kumar, P.R., "Architecturally-efficient computation of shortest
606 paths for a mobile robot," *IEEE Symposium on Industrial Electronics & Applications*, 2009.
607 *ISIEA 2009*, Vol. 1, no., pp. 277, 282, 4-6Oct.2009

608 [6] Sridharan, K.; Priya, T. K.; Kumar, P.R., "Hardware architecture for finding shortest
609 paths," *IEEE Region 10 Conference TENCN 2009 - 2009*, vol., no., pp. 1, 5, 23-26
610 Jan.2009.

611 [7]. Priya, T. K.; Sridharan, K., "An efficient algorithm to construct reduced visibility graph
612 and its FPGA implementation," *VLSI Design, 2004. Proceedings. 17th International
613 Conference on*, vol., no., pp. 1057,1062, 2004.

614 [8]. Jagadeesh, G.R.; Srikanthan, T.; Lim, C.M., "Field programmable gate array-based
615 acceleration of shortest-path computation," *Computers & Digital Techniques, IET*, Vol. 5, no.
616 4, pp. 231,237, July 2011.

617 [9] Chinnaiah, M.C.; Sanjay, D.; Kumar, P.R.; Savithri, T.S., "A Novel approach and
618 Implementation of Robot path planning using Parallel processing algorithm," *International
619 Conference on Communications, Devices and Intelligent Systems (CODIS)*, 2012, vol., no.,
620 pp.345,348, 28-29 Dec.2012.

621 [10] Algabri, M., Mathkour, H., Ramdane, H., Alsulaiman, M. ,"Comparative study of soft
622 computing techniques for mobile robot navigation in an unknown environment" (2015)
623 *Computers in Human Behavior*, 50, pp. 42-56.

624 [11] Chinnaiah, M.C., Divyavani, G., Satyasavithri, T., Rajeshkumar, P. "Implementation of
625 Shortest path planning algorithm without track using FPGA robot: A new approach", 2014
626 *International Conference on Advances in Electrical Engineering*, ICAEE 2014.
627

628 [12] Abdelmoula, C., Chaari, F., Masmoudi, M. "Real time algorithm implemented in Altera's
629 FPGA for a newly designed mobile robot: Autonomous navigation and parallel parking"
630 (2014), *Multidiscipline Modelling in Materials and Structures*, 10 (1), art. no. 17112943, pp.
631 75-93.
632

633 [13] Baturone, I., Gersnoviez, A., Barriga, Á. "Neuro-fuzzy techniques to optimize an FPGA
634 embedded controller for robot navigation" (2014) *Applied Soft Computing Journal*, 21, pp.
635 95-106.
636

637 [14] Howard, T., Pivtoraiko, M., Knepper, R.A., Kelly, A. "Model-predictive motion planning:
638 Several key developments for autonomous mobile robots" (2014) *IEEE Robotics and
639 Automation Magazine*, 21 (1), pp. 64-73.
640

641 [15] Lumelsky, V.J.; Mukhopadhyay, S.; Sun, K., "Dynamic path planning in sensor-based
642 terrain acquisition," *Robotics and Automation, IEEE Transactions on*, Vol. 6, no. 4, pp. 462,
643 472, Aug 1990
644

645 [16] Supakar, N., Senthil, A. "PSO obstacle avoidance algorithm for robot in unknown
646 environment" (2014) 2013 *International Conference on Communication and Computer
647 Vision*, ICCCV 2013.
648

649 [17] Das, A., Diu, M., Mathew, N., Scharfenberger, C., Servos, J., Wong, A., Zelek, J.S.,
650 Clausi, D.A., Waslander, S.L."Mapping, planning, and sample detection strategies for
651 autonomous exploration" (2014) *Journal of Field Robotics*, 31 (1), pp. 75-106.

652

653 [18] Vachhani, L.; Sridharan, K., "Hardware-Efficient Prediction-Correction-Based
654 Generalized-Voronoi-Diagram Construction and FPGA Implementation," *Industrial*
655 *Electronics, IEEE Transactions on*, Vol. 55, no. 4, pp. 1558,1569, April 2008.

656

657 [19] Yongguo Mei; Yung-Hsiang Lu; Hu, Y.C.; Lee, C. S G, "A case study of mobile robot's
658 energy consumption and conservation techniques," *Advanced Robotics*, 2005. ICAR '05.
659 *Proceedings.*, 12th International Conference on, vol., no., pp. 492,497, 18-20 July 2005.

660

661 [20] Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numer. Math.* **1**, 269–
662 271 (1959). .

663

664 [21] Dijkstra, E.W., Misa, T.J. (eds.): An Interview with Edsger W. Dijkstra. *Commun.*
665 *ACM* **53**(8), 41–47 (2010).

666

667